

MG-CANEX 协议转换器

用户手册

V1.2

2025.09.26

CANopen 到 Modbus TCP 协议转换器



四川零点自动化系统有限公司

版权©2015 四川零点自动化系统有限公司保留所有权利

版本信息

对该文档有如下的修改：

日期	版本号	修改内容	作者
2015-5-26	V1.0	发布版本	GJ
2020-03-26	V1.1	外观改版	CCL
2020-07-20	V1.2	硬件改版，功能完善	CCL
2025-09-26	V1.2	更新 DF 指示灯说明	YPP

所有权信息

未经版权所有者同意，不得将本文档的全部或者部分以纸质或者电子文档的形式重新发布。

免责声明

本文档只用于辅助读者使用产品，本公司不对使用该文档中的信息而引起的损失或者错误负责。本文档描述的产品和文本正在不断地开发和完善中。四川零点自动化系统有限公司有权利在未通知用户的情况下修改本文档。

软件下载

请登录零点自动化官网 <https://www.odot.cn/support.html>，在对应的产品页面点击下载。

免责条款

产品使用

注意

- 在安装、操作和维护设备时，请勿超过电气特性中指定的任何额定值；
 - 在安装、操作和维护设备时，请勿超过环境特性中指定的任何额定值。请勿在下列场所使用产品：有灰尘、油烟、导电性尘埃、腐蚀性气体、可燃性气体的场所；请勿暴露于高温、结露、风雨的场合；有振动、冲击的场合也会导致产品损坏；
- 未按说明操作则设备提供的保护可能会失效，可能导致轻微身体伤亡或设备损坏。

免责范围

本公司不对以下情况导致的设备损坏或故障承担任何责任：

- 1、运输损坏：由于不当运输或包装造成的设备损坏；
- 2、自然因素：雷击、电压波动、进水或自然灾害（如火灾、洪水等）造成的损坏；
- 3、不当使用：超负荷、不规范操作、私自改装或使用不合格配件导致的损坏；
- 4、未授权维修：未经授权的维修或改动所导致的设备故障；
- 5、其他非产品原因：与设备本身无关的其他原因引起的损坏。

维修服务

- 1、对于上述原因导致的损坏，本公司将根据实际情况收取维修费用。
- 2、保修期外，本公司提供有偿维修服务，费用依据维修情况收取。

风险承担

本公司不承担因设备使用过程中产生的人员伤亡、财产损失或其他相关损失。所有风险由使用者自行承担。

目 录

1 产品概述	6
1.1 产品功能	6
1.2 功能特点	7
1.3 技术参数	8
2 硬件说明	9
2.1 产品外观	9
2.2 指示灯说明	10
2.3 端子定义	11
2.4 系统复位	11
2.5 安装尺寸	12
3 产品应用拓扑图	13
4 产品使用	14
4.1 网关工作原理	14
4.2 数据对象在 Modbus 缓存区的映射	15
4.3 网络功能	15
4.3.1 网络扫描	15
4.3.2 紧急对象	16
4.3.3 NMT 网络管理	18
4.3.4 SDO 服务数据对象	20
5 IO Config 配置软件	23
5.1 软件安装	23
5.2 加载硬件支持包	24
5.3 软件界面	24
5.4 网关参数配置	32
6 固件升级	45
7 产品应用实例	47
7.1 网关与分布式 IO 通讯示例	47
7.2 网关与鸣志伺服驱动器通讯示例	58
附录：CANopen 协议简介	69
1 CANopen 概述	69
2 NMT 网络管理	69
3 服务数据对象 SDO	71
4 SDO 传输协议	71
5 紧急对象 Emergency Object	74
6 过程数据对象 PDO	75

1 产品概述

1.1 产品功能

MG-CANEX 是一款 CANopen 到 Modbus TCP 的协议转换器。设备在 CANopen 网络中作为主站，可连接标准 CANopen 从站设备。数据传输支持 PDO、SDO，错误控制支持 Heartbeat。支持同步、异步报文发送。

设备在 Modbus TCP 网络中作为 TCP 服务器，可供 5 个 TCP 客户端同时访问，可接入 PLC 控制器以及各种上位机组态软件。可连接光端机等设备实现远距离数据传输。

1.2 功能特点

- ◆ 网关自带配置软件，参数配置信息通过网口下载到网关。网关自动保存最新配置信息，网关断电在上电后无需加载配置。
- ◆ 网关在 CANopen 网络上为 CANopen 主站，可接 CANopen 从站设备。
- ◆ 网关在 Modbus TCP 网络上为 Modbus server，支持多达 5 个 TCP 客户机访问。双以太网口，带交换机功能，支持级联。
- ◆ 2KV 网口隔离保护，10M/100Mbps 速率自适应，自动 MDI/MDIX 翻转。
- ◆ 支持地址映射模式，实现对 TCP 客户端请求的快速响应。
- ◆ Modbus TCP 支持功能码：0x01,0x02,0x03,0x04,0x05,0x06,0x0F,0x10。
- ◆ 6KB 超大数据缓存区，数据传输量更大。
- ◆ CAN 接口支持 CANopen 工作模式。
- ◆ CAN 接口波特率：10K~1Mbps。
- ◆ CANopen 协议符合 DS301 V4.02，支持 NMT 主站、PDO、SDO、Heartbeat。
- ◆ 支持一键复位功能，恢复出厂设置。
- ◆ 35mm 标准导轨安装。
- ◆ EMC 符合 EN 55022:2010 & EN55024:2010 国际标准。

1.3 技术参数

本产品相关技术参数如下表所示，请在本产品的参数范围内使用本产品，以便获得更好的性能。

环境参数	
工作温度范围	-40~85℃/-20~70℃ 可选
存储温度范围	-45~125℃
工作湿度范围	5%~95%（无冷凝）
电源参数	
电源端口数量	1 路
输入电压范围	19.2~28.8VDC, 3KV 隔离电压
功耗	Max.110mA@24V
以太网参数	
Modbus TCP 功能码	0x01、0x02、0x03、0x04、0x05、0x06、0x0F、0x10
以太网端口数量	2 个 RJ45, 10M、100M 自适应速率, 带交换机功能
网络协议	ETHERNET、ARP、IP、TCP、ICMP
TCP 连接数量	最大 5 个
Modbus 数据存储区	0xxxx 区（线圈）: 8192 Bit 1xxxx 区（离散量输入）: 8192 Bit 3xxxx 区（输入寄存器）: 2048 Word 4xxxx 区（保持寄存器）: 2048 Word
默认 IP 地址	192.168.1.100
CANopen 参数	
CAN 波特率	10K~1Mbps
CAN 协议	CANopen
支持从站数	16 个站
PDO 功能	支持 TPDO、RPDO 数据传输
SDO 功能	支持最大 4 字节快速 SDO 传输
错误控制	支持 Heartbeat

2 硬件说明

2.1 产品外观



2.2 指示灯说明

设备共有 6 个 LED 状态指示灯，其符号定义及状态说明如表所示。

符号	定义	状态	说明
PWR	电源指示	常亮	电源接通
		熄灭	电源未接通
DF	系统故障指示	闪烁	系统存在故障
		熄灭	设备正常
RUN	以太网运行状态	闪烁	Modbus-TCP 数据收发中
	CAN 运行状态指示	闪烁(2Hz)	预操作状态
		单闪	停止状态
		常亮	操作状态
ERR	以太网错误状态	闪烁(2Hz)	网口连接异常
	CAN 错误状态指示	单闪	CAN 错误帧达到警戒值
		双闪	错误控制事件
		常亮	总线关闭
		熄灭	总线正常
CTX	CAN 发送指示	闪烁	CAN 在发送数据
		熄灭	CAN 未发送数据
CRX	CAN 接收指示	闪烁	CAN 在接收数据
		熄灭	CAN 未接收数据

2.3 端子定义

网关接线采用 10Pin 3.81mm 间距拔插式接线端子，CAN 接口及电源端子定义如表所示。

序号	端子	定义
1	NC	空
2	NC	空
3	NC	空
4	NC	空
5	NC	空
6	NC	空
7	NC	空
8	NC	空
9	CAN_L	CAN_L 信号线
10	CAN_H	CAN_H 信号线
11	SGND	CAN 信号地
12	PE	接地端子
13	NC	空
14	NC	空
15	NC	空
16	NC	空

2.4 系统复位

设备带有系统复位按钮 RESET，当用户忘记设备的 IP 地址及配置端口号而不能连接到网关时，可触发复位按钮。系统参数将恢复出厂设置，复位后可重新下载配置参数。触发复位按钮后，所有 LED 指示灯被点亮，设备复位完成后指示灯熄灭（PWR 除外）。

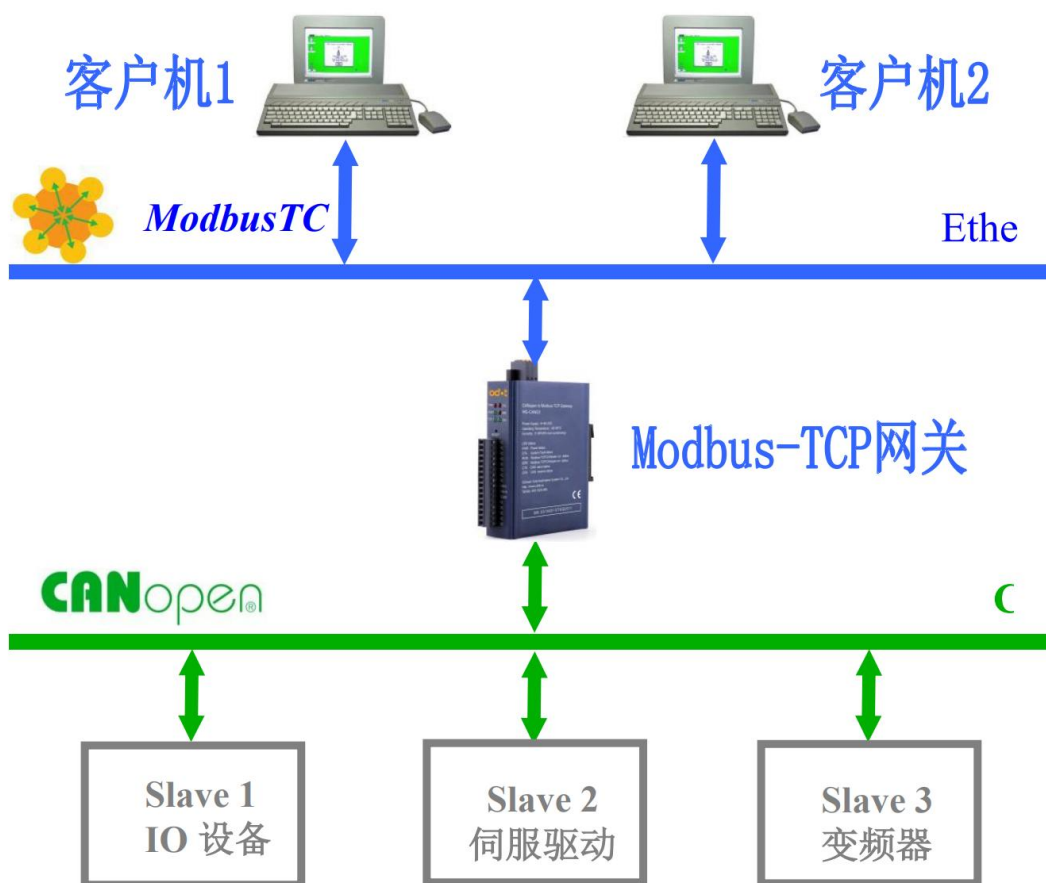


2.5 安装尺寸



3 产品应用拓扑图

产品典型的网络拓扑图如图所示。



4 产品使用

4.1 网关工作原理

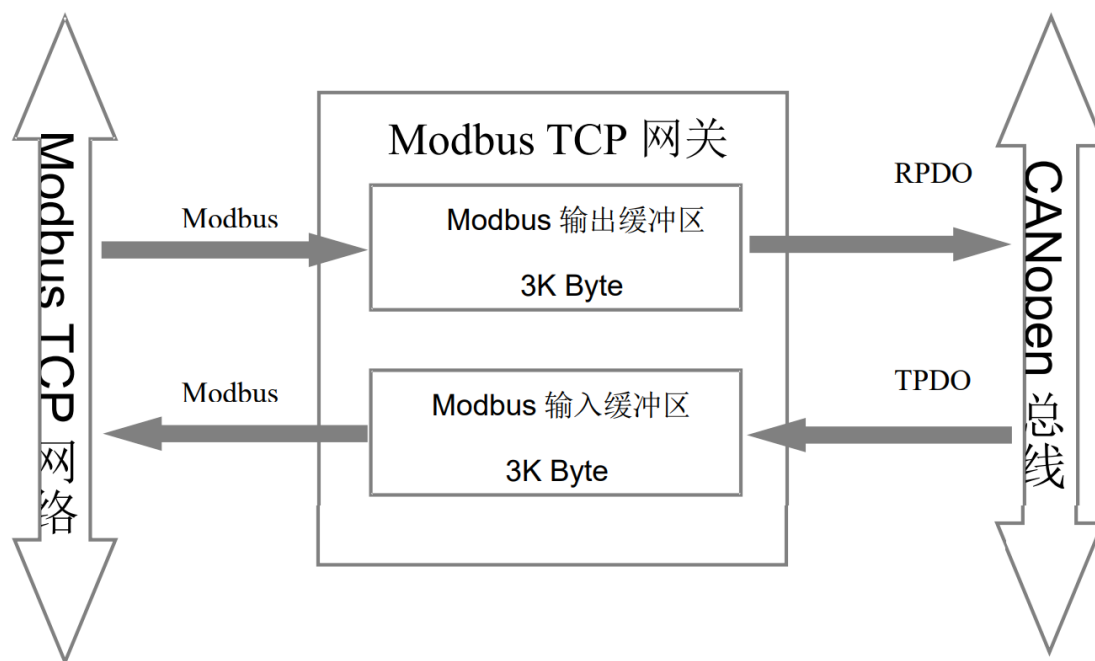
网关协议转换属于地址映射模式，CANopen 网络中所有设备的数据都被映射到 Modbus TCP 数据存储区中。

CANopen 从站设备的 TPDO 被映射到 Modbus 离散量输入和输入寄存器区，RPDO 被映射到 Modbus 线圈和保持寄存器区。

CANopen 从站设备的 SDO 读命令被映射到 Modbus 输入寄存器区（3xxxx），SDO 写命令被映射到 Modbus 保持寄存器区（4xxxx）。

当网关接收到从站的数据后，将数据存储于 Modbus 缓冲区中。当 Modbus 缓冲区中的数据发生改变时，网关将发送从站 RPDO/SDO 写命令的数据到 CANopen 网络中。

网关数据交换原理下图所示。



PDO 采用生产者/消费者模式来传输数据，只有请求没有应答，响应快，适合对响应速度要求高的场合。SDO 采用客户机/服务器模式进行数据传输，有请求也有应答，响应速度慢，但可靠性高。所以对可靠性要求比较高的数据可配置

SDO 命令来传输数据，否则就配置 PDO 来传输数据。

4.2 数据对象在 Modbus 缓存区的映射

数据对象	映射到 Modbus 寄存器区	Modbus TCP 功能码
TPDO (位)	1xxxx	02
TPDO (字)	3xxxx	04
RPDO (位)	0xxxx	5/15
RPDO (字)	4xxxx	6/16
SDO 循环写	4xxxx	6/16
SDO 循环读	3xxxx	04

4.3 网络功能

4.3.1 网络扫描

CANopen 网络上最大 127 个节点，网关本身占一个节点地址。通过网络扫描功能可以初步扫描 CANopen 网络上的从站设备的基本情况。

通过添加网络扫描模块来实现网络扫描功能。网络扫描模块数据地址对应关系如下图：

数据方向	字偏移	描述	高字节								低字节								
		位	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	数据说明
输入	0	扫描状态字	扫描到的网络节点数量								/				扫描中		0→1触发位		蓝色数据位为输出反馈值。
	1	模块信息	节点ID								节点状态机								
	2	模块信息	节点ID								节点状态机								
	.	.	.								.								
	.	.	.								.								
	126	模块信息	节点ID								节点状态机								
输出	0	扫描控制字	/								/				0→1触发位		扫描启动触发位。		

注：绿色：只读；蓝色：反馈

控制流程：

- 1.输出触发位 0->1 上升沿，启动扫描。
- 2.扫描中状态为被置 1，节点数量及模块信息全部清零。
- 3.等待扫描完成，扫描中状态位清零。
- 4.节点数量中存储当前网络扫描到的所有节点数量，模块信息中存储节点 ID 及节点状态机信息。

4.3.2 紧急对象

紧急报文由设备内部出现的致命错误触发，由相关应用设备已最高优先级发送到其它设备。适用于中断类型的错误报警信号。

一个紧急报文由 8 字节组成，格式如下：

sender → receiver(s)

COB-ID	Byte0-1	Byte2	Byte3-7
0x080+Node_ID	错误代码	错误寄存器 (对象 0x1001)	制造商特定的错误区域

16 进制的应急错误代码如下表所示。应急错误代码中‘xx’部分由相应的设备子协议定义。

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 tempearture

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 lenvgth error
8220	Length exceedd
90xx	External error
F0xx	Additional functions
FFxx	Device specific

错误寄存器(Error Register)在设备的对象字典（索引 0x1001）中，表中说明了错误寄存器的位定义。设备可以将内部错误映射到这个状态字节中，并可以快速查看当前错误。

Bit	错误类型
0	Generic
1	Current
2	Voltage
3	Temperature
4	Communication
5	Device profile specific
6	Reserved(=0)
7	Manufacturer specific

制造商特定错误区域可能包含与设备相关的其它的错误信息。紧急报文由设备内部出现的致命错误触发，由相关应用设备已最高优先级发送到其它设备。适

用于中断类型的错误报警信号。

通过添加 EMCY 控制模块子模块来实现紧急报警工能。EMCY 控制模块数据地址对应关系如下图：

字偏移	描述	高字节								低字节								数据说明	
	位	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0		
0	状态字	EMCYDATA_Reset	Counter_Reset	Overflow_Reset	NonEmpty_Reset	/		Overflow	NonEmpty	/	NodeID								
1	溢出计数	Overflow_Couter																	
2	EMCY数据1	Error_Code																	
3	EMCY数据2	Manufacture_Data_0								Error_Register									
4	EMCY数据3	Manufacture_Data_2								Manufacture_Data_1									
5	EMCY数据4	Manufacture_Data_4								Manufacture_Data_3									
0	扫描控制字	EMCYDATA_Reset	Counter_Reset	Overflow_Reset	NonEmpty_Reset	/				/				NodeID					

注：蓝色：反馈；绿色：可清零

控制流程：

- 1.等待输入位 NonEmpty 置 1,说明收到 1 条紧急报文。
- 2.读取紧急报文信息 NodeID, Error_Code, Error_Register, Manufacture_Data, 处理报警信息。
- 3.控制输出位 NonEmpty_Reset 上升沿来清除输入 NonEmpty 标志。
- 4.若输入位 Overflow 被置 1, 表示当前有紧急报文被丢弃, Overflow_Couter 表示了被丢弃的紧急报文数量。
- 5.控制输出位 Overflow_Reset, Counter_Reset 上升沿来清除输入 Overflow, Overflow_Couter。
- 6.可控制输出位 EMCYDATA_Reset 位上升沿来清除紧急报文信息 NodeID, Error_Code, Error_Register, Manufacture_Data。

4.3.3 NMT网络管理

CANopen NMT 网络管理功能可通过对“系统控制区”中 NMT 控制域的读写操作来实现, NMT 控制寄存器地址范围为 0x8000~0x8040。Modbus TCP 客户机可通过 0x03、0x06、0x0F 功能码访问该寄存器组。

NMT 命令字为网络管理控制命令, 有效命令字取值如下：

0x01: 启动远程节点。

0x02: 停止远程节点。

0x80: 进入预操作状态。

0x81: 复位节点。

0x82: 复位通信。

写入其他的 NMT 命令值将会被忽略。触发位由 0 变成 1 时，将启动一次 NMT 命令的发送，NMT 从站地址为远程节点地址，取值为 1-127,0 代表广播地址。

NMT 状态包含当前网络中所有从站的当前状态（要获得有效的从站状态，必须启动从站的错误控制功能 Node Guarding 或 Heartbeat），从站状态内容为只读，写入任何值都将会被忽略，状态值对应的状态如表所示。初始化状态表示主站收到该从站的 Boot-up 启动报文，当主站查询从站状态超时或接收从站心跳包超时，表示从站离线，当收到从站的状态信息时，处于停止、操作、预操作三种状态，当未收到任何从站状态信息时为未知状态。

状态值	节点状态
0x00	初始化状态
0x01	离线状态
0x04	停止状态
0x05	操作状态
0x7F	预操作状态
0x0F	未知状态

通过添加 NMT 网络管理模块子模块来实现控制从站状态功能。NMT 网络管理模块数据地址对应关系如下图：

数据方向	字偏移	描述	高字节								低字节									
		位	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	数据说明	
输入	0	NMT控制字	NMT命令字								0→1触发位	NodeID, 0代表广播地址								输出的NMT控制字反馈
输出	0	NMT控制字	NMT命令字								0→1触发位	NodeID, 0代表广播地址								NMT控制字

注：蓝色:数据反馈

控制流程：

1.NodeID 节点地址赋值，表示要操作的节点 ID，0 代表广播地址。

2.NMT 命令字赋值。

3.触发位 0->1 上升沿触发 NMT 命令发送。

4.3.4 SDO服务数据对象

SDO 的在线读写功能可通过对“系统控制区”中 SDO 控制域的读写操作来实现，SDO 控制寄存器地址范围为 0x8046~0x806B。Modbus TCP 客户机可通过 0x03、0x06、0x0F 功能码访问该寄存器组，其数据具体编码格式如表所示。

描述	高字节								低字节									
位	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	数据说明	
索引	Index																	
节点ID/子索引	/	NodeID								SubIndex								
控制字	Abort_Code_Reset	Done_Reset	Error_Reset	/	SDO_Done	SDO_Error	SDO_Busy	/	Trigger	RW_Type	Byte_Len							
中止代码	Abort_Code_3								Abort_Code_2									
	Abort_Code_1								Abort_Code_0									
SDO数据1	SDO_Data_1																	
.	.																	
.	.																	
SDO数据16	SDO_Data_16																	
索引	Index																	
节点ID/子索引	/	NodeID								SubIndex								
控制字	Abort_Code_Reset	Done_Reset	Error_Reset	/				/	Trigger	RW_Type	Byte_Len							
SDO数据1	SDO_Data_1																	
.	.																	
.	.																	
SDO数据16	SDO_Data_16																	

注：红色:只读；蓝色:反馈；橙色:条件反馈；绿色:可清零

对象索引 Index，子索引 SubIndex 为将要访问的对象参数。SDO 服务器从站号有效地址范围为 1-127。中止代码 Abort_Code 的定义如表所示。

Abort_Code 中止代码	故障说明
0x06010001	试图读只写对象
0x06010002	试图写只读对象
0x06020000	对象字典中对象不存在
0x06040041	在索引处的 PDO 中不可映射数据
0x06070010	传入数据长度错误
0x06070012	数据太长
0x06070013	数据太短

0x06090011	对象字典中子索引不存在
0x06090030	写访问超出参数范围
0x06090031	值范围错误：值过低
0x06090032	值范围错误：值过高
0x05030000	触发位没有交替改变
0x05040000	传输超时
0x05040005	超出存储器范围
0x08000000	通用错误
0x08000021	本地控制错误

控制流程：

A:读流程

- 1.设置对象索引/子索引/节点地址信息 Index/SubIndex/NodeID。
- 2.设置 RW_Type 为 0，表示 SDO 上传。
- 3.设置触发位 Trigger 上升沿，SDO 传输开始，SDO_Busy 位被置 1。
- 4.用户等待 SDO_Done 完成位置 1。
- 5.若 SDO 传输正常 SDO_Error 和 Abort_Code 为 0，Byte_Len 中存储读取的对象数据的字节长度，SDO_Data 中存储对象的值，有效字节长度为 Byte_Len。
- 6.若 SDO 传输失败 SDO_Error 位被置 1，Abort_Code 中存储了中止代码，指示失败原因。Byte_Len 及 SDO_Data 被清空。
- 7.控制 Done_Reset/Error_Reset 位上升沿清除 SDO_Done/SDO_Error 标志位，以便开始下一次传输。
- 8.可以控制 Abort_Code_Reset 位上升沿来清除错误代码 Abort_Code。

B:写流程

- 1.设置对象索引/子索引/节点地址信息 Index/SubIndex/NodeID。
- 2.设置 RW_Type 为 1，表示 SDO 下载，设置输出数据长度及输出数据值 Byte_Len/SDO_Data，输出值将反馈到对应的输入值中。
- 3.设置触发位 Trigger 上升沿，SDO 传输开始，SDO_Busy 位被置 1。

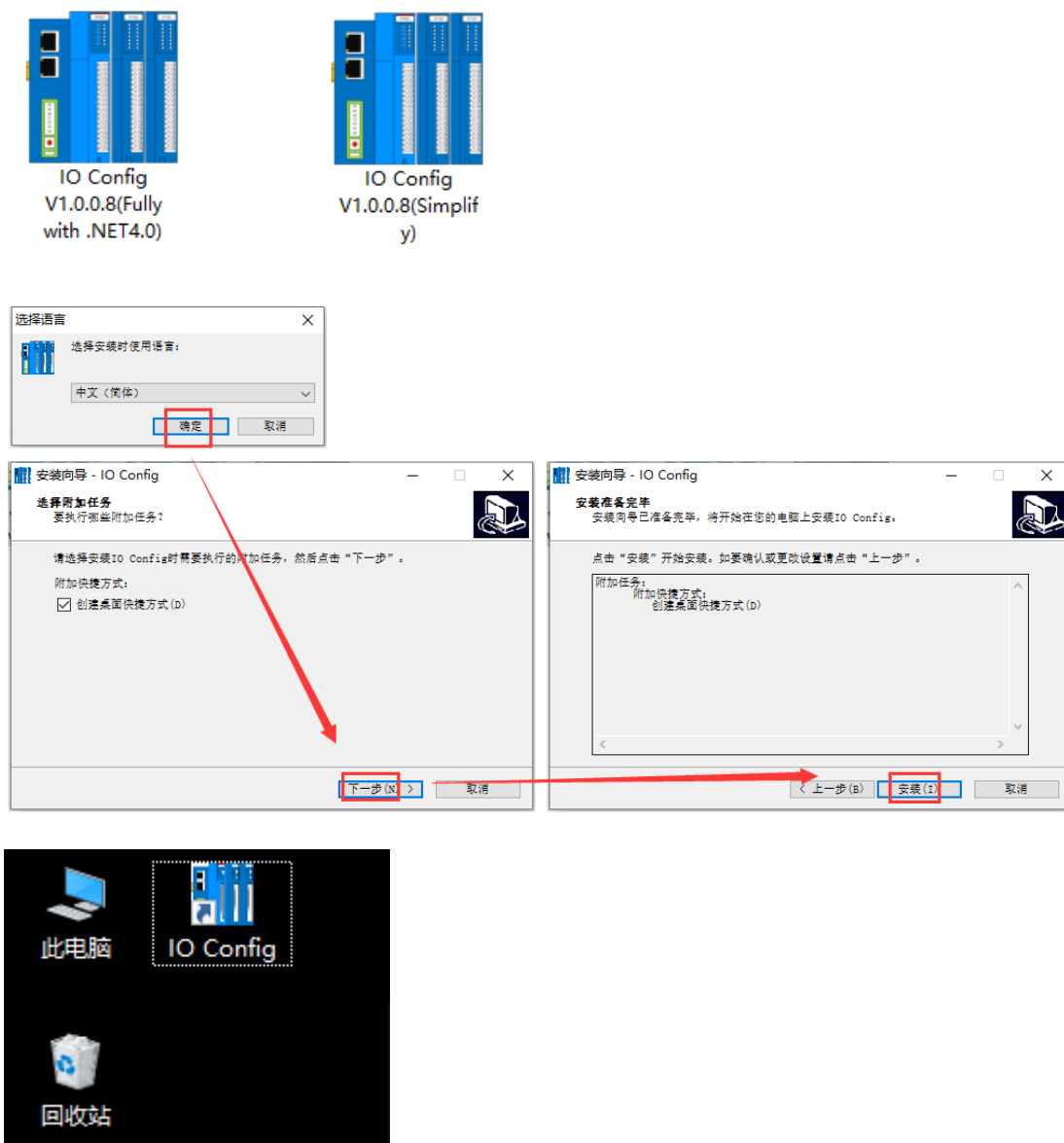
- 4.用户等待 SDO_Done 完成位置 1。
- 5.若 SDO 传输正常 SDO_Error 和 Abort_Code 为 0。
- 6.若 SDO 传输失败 SDO_Error 位被置 1，Abort_Code 中存储了中止代码，指示失败原因。
- 7.控制 Done_Reset/Error_Reset 位上升沿清除 SDO_Done/SDO_Error 标志位，以便开始下一次传输。
- 8.可以控制 Abort_Code_Reset 位上升沿来清除错误代码 Abort_Code。

5 IO Config 配置软件

5.1 软件安装

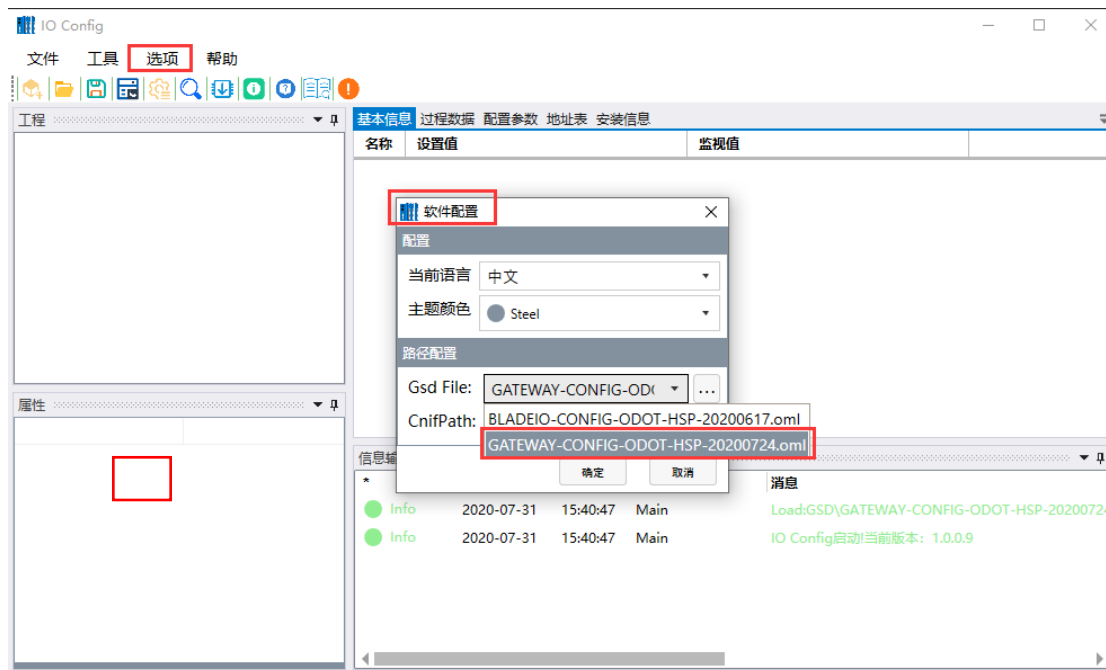
公司提供给客户 IO Config V x.x.x.x (Fully with .NET4.0)（完整版的安装文件带.net 4.0 环境）和 IO Config V x.x.x.x (Simplify)（简洁版的不带.net 4.0 环境）配置软件。

客户收到配置软件后双击图标点击安装，在弹出的窗口依次点击“确定”——“下一步”——“安装”。勾选创建桌面快捷方式，安装完成后，会在桌面生成 IOConfig 快捷图标。

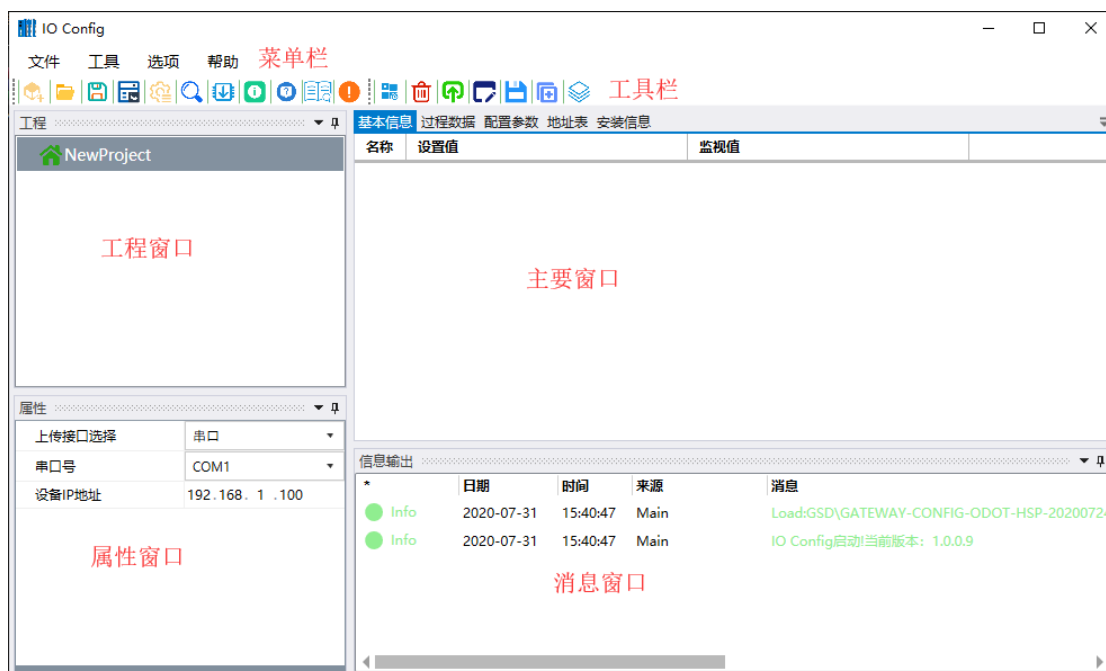


5.2 加载硬件支持包

安装完成后，需要加载网关 MG-CANEX 的硬件支持包。打开软件，点击“选项”——“配置”，Gsd File 路径配置的文件.oml 选择“GATEWAY-CONFIG-ODOT-HSP-20200724.oml”。点击“确定”。



5.3 软件界面



菜单栏：IOConfig 软件的菜单

工具栏：常用的用户菜单。

工程窗口：树型显示当前激活的工程。

属性窗口：显示当前项的具体参数。网关（模块名称、模块号、模块描述、设备版本、模块个数、接口选择、设备 IP 地址、串口号、在线刷新周期），从站设备（模块名称、模块号、模块描述、子模块个数）。

主要窗口

基本信息：可查看网关的模块名称、模块号、硬件版本、软件版本、模块描述、电流消耗。

过程数据：可以用于在线监控通道数据。

配置参数：模块可修改的模块参数。

地址表：Modbus 映射地址表。

安装信息：可查看模块描述、电流消耗、模块尺寸、剩余电流，产品图片。

消息窗口：输出当前操作的实时信息，显示新建工程、上传、下载、配置参数修改等操作日志

主菜单

◇文件

菜单	子菜单	描述
工程	新建工程	创建新的工程
	打开工程	打开已保存的工程
	全部保存	保存当前工程
	另存为	保存当前工程作为新的工程
退出		关闭工程

工具

菜单	描述
搜索设备	弹出新的窗口，用于 MODBUS 通讯搜索设备
在线升级	弹出新的窗口，用于网关固件升级

选项

菜单	描述
----	----

配置	可修改软件显示语言，软件界面显示颜色，硬件支持包文件路径
----	------------------------------

帮助

菜单	描述
关于	可查看配置软件版本号
异常帮助	弹出新的窗口，异常退出提醒，WIN7 sp1/XP 系统以下版本请安装微软补丁。

工具栏

菜单常规快捷图标



图标	名称	菜单	描述
	新建工程	文件-工程-新建工程	创建新的工程
	打开工程	文件-工程-打开工程	打开已保存的工程
	全部保存	文件-工程-全部保存	保存当前工程
	另存为	文件-工程-另存为	保存当前工程作为新的工程
	配置	选项-配置	可修改软件显示语言，软件界面显示颜色，设备库描述文件路径
	搜索设备	工具-搜索设备	弹出新的窗口，用于 MODBUS 通讯搜索设备
	在线升级	工具-在线升级	弹出新的窗口，用于网关固件升级
	关于	帮助-关于	可查看配置软件版本号
	帮助文档	帮助-帮助文档	弹出新的窗口，IOConfig 软件的使用手册
	硬件手册	帮助-硬件手册	弹出新的窗口，所有 IO 模块的硬件手册
	异常帮助	帮助-异常帮助	弹出新的窗口，异常退出提醒，WIN7 sp1/XP 系统以下版本请安装微软补丁。

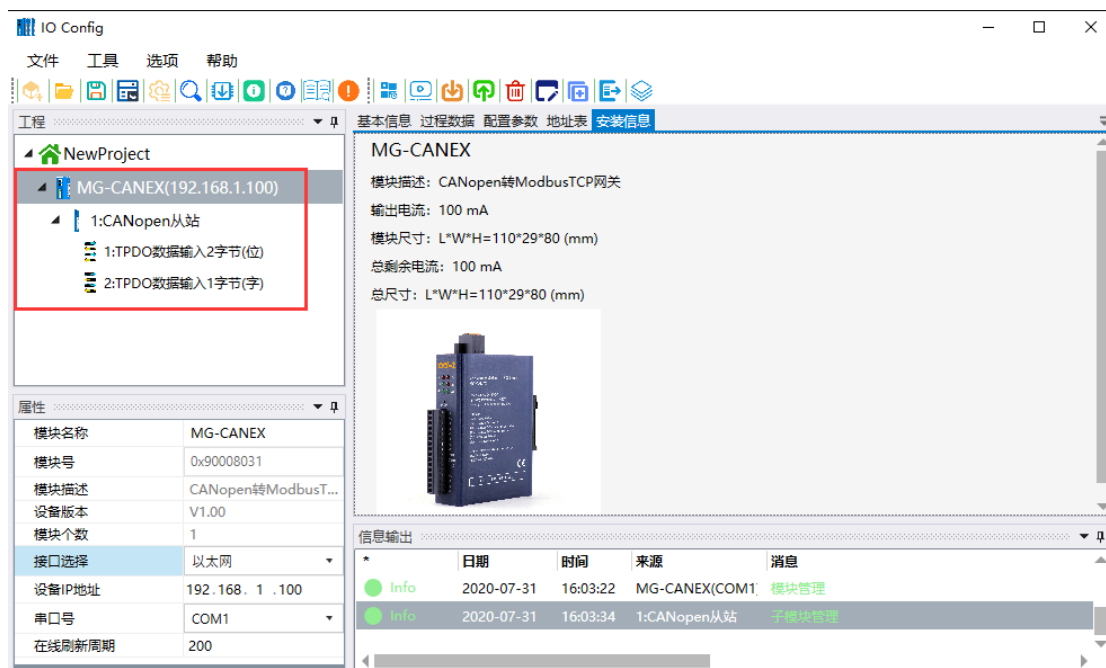
适配器常规快捷图标



图标	名称	菜单	描述
	模块管理	网关-模块管理	添加 CANopen 从站
	在线	网关-在线	网关在线监控。
	下载 IO 参数	网关-下载 IO 参数	下载网关所有参数
	上传 IO 参数	网关-上传 IO 参数	上载网关所有参数
	删除	网关-删除	删除当前网关
	重命名	网关-重命名	网关重新命名
	复制	网关-复制	复制网关
	导出地址表	网关-导出地址表	导出网关地址对应关系
	导出文档	网关-导出文档	导出网关及从站配置所有信息，包括地址表、从站配置参数、尺寸图。

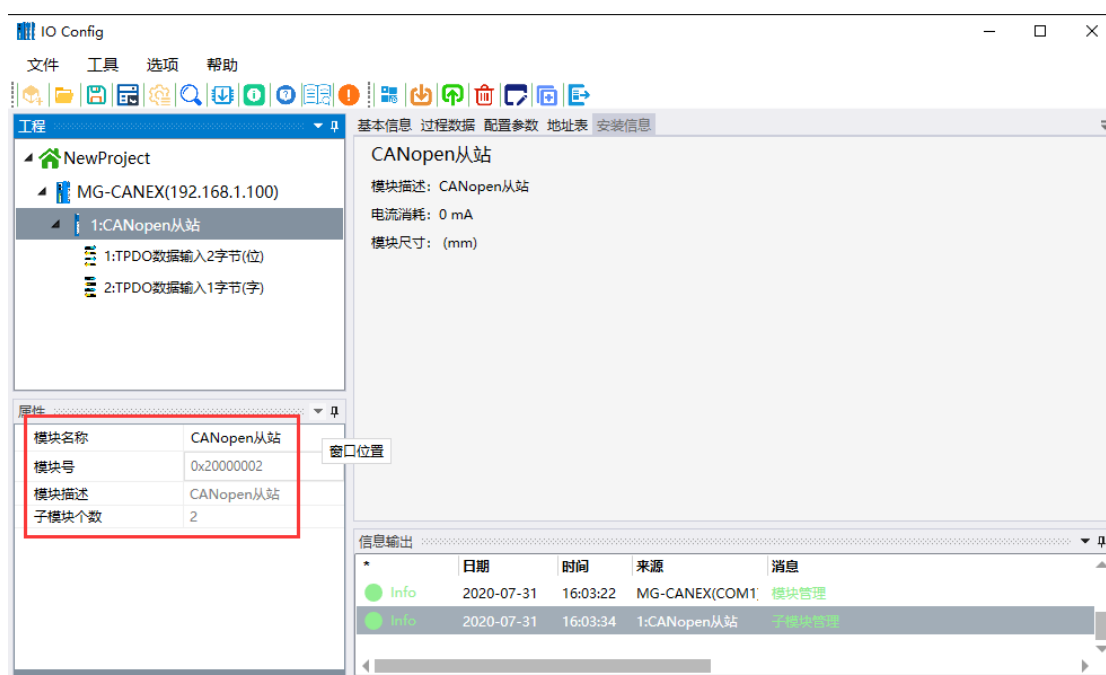
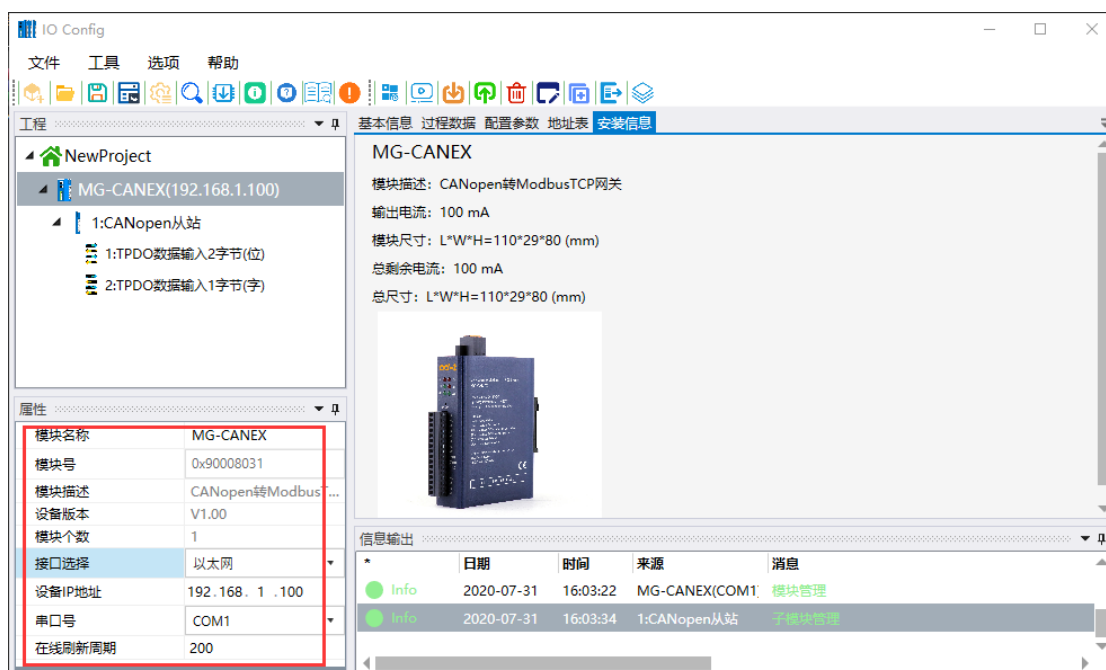
工程窗口

以树型形式显示当前激活的项目，



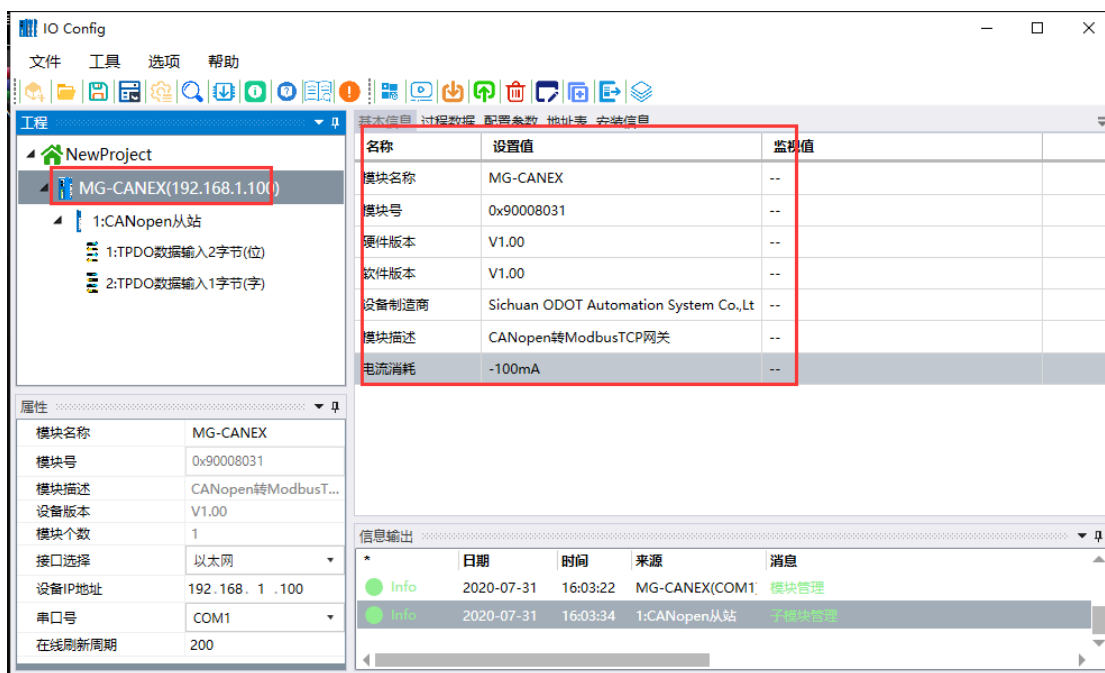
属性窗口

属性窗口显示当前项的具体参数。适配器模块（模块名称、模块号、模块描述、设备版本、模块个数、接口选择、设备 IP 地址、串口号、在线刷新周期），IO 模块（模块名称、模块号、模块描述、子模块个数）

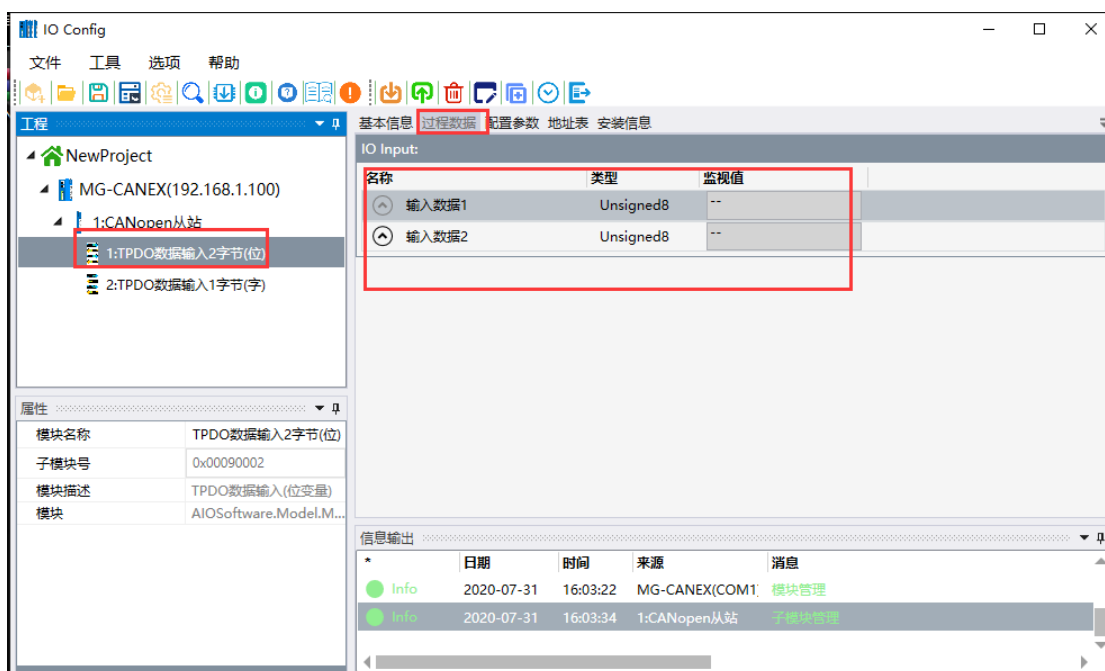


主要窗口

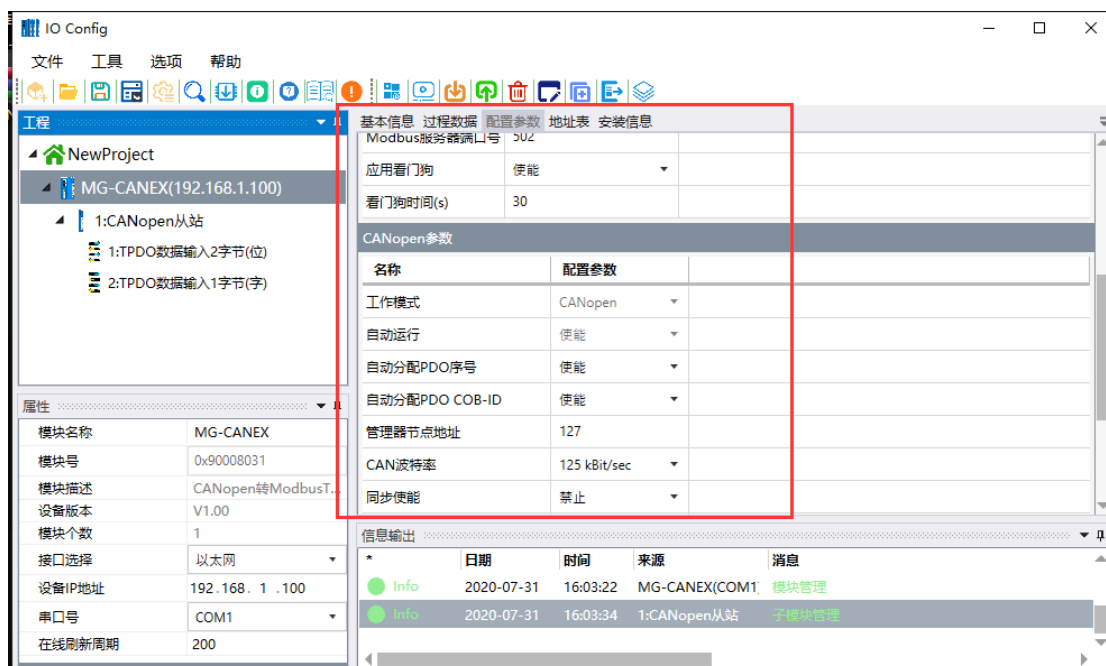
基本信息：可显示网关及 CANopen 从站设备名称、模块号、硬件版本、软件版本、模块描述、电流消耗、设备制造商。



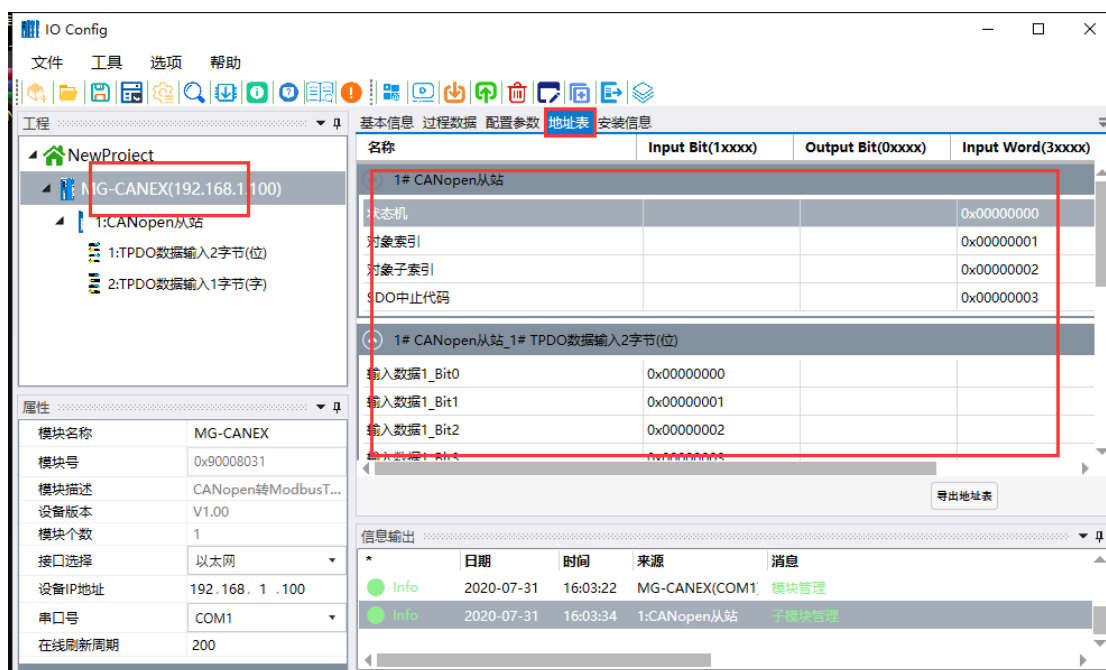
过程数据：显示 CANopen 从站的读写数据，用于在线监控通道数据



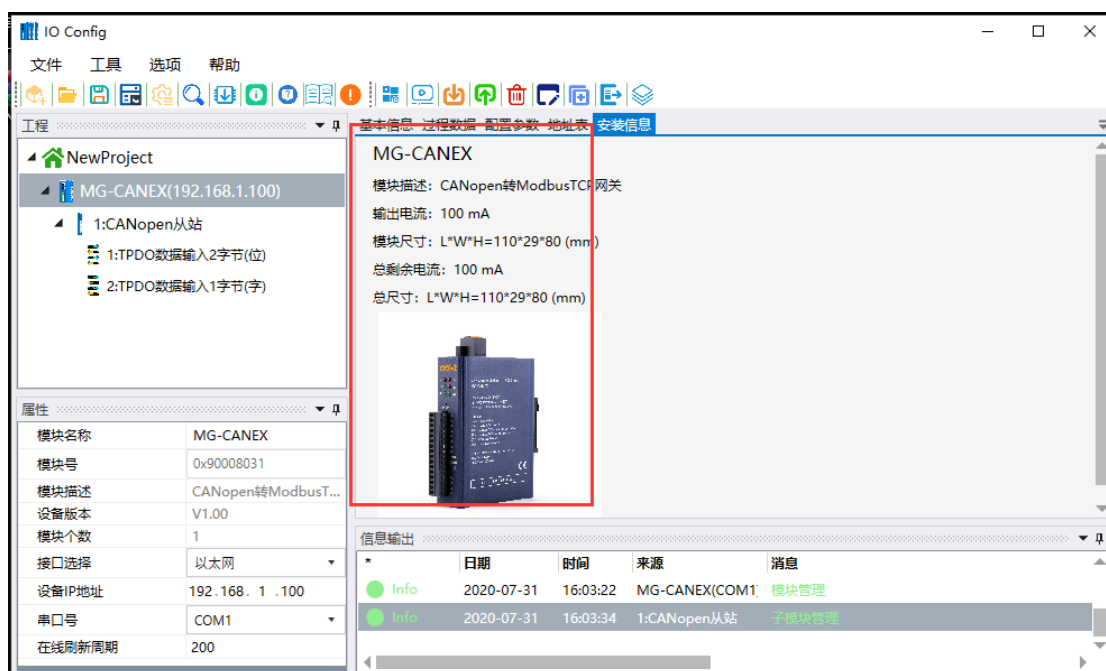
配置参数：显示网关和 CANopen 的设置参数，可修改。



地址表：显示网关的地址映射表。

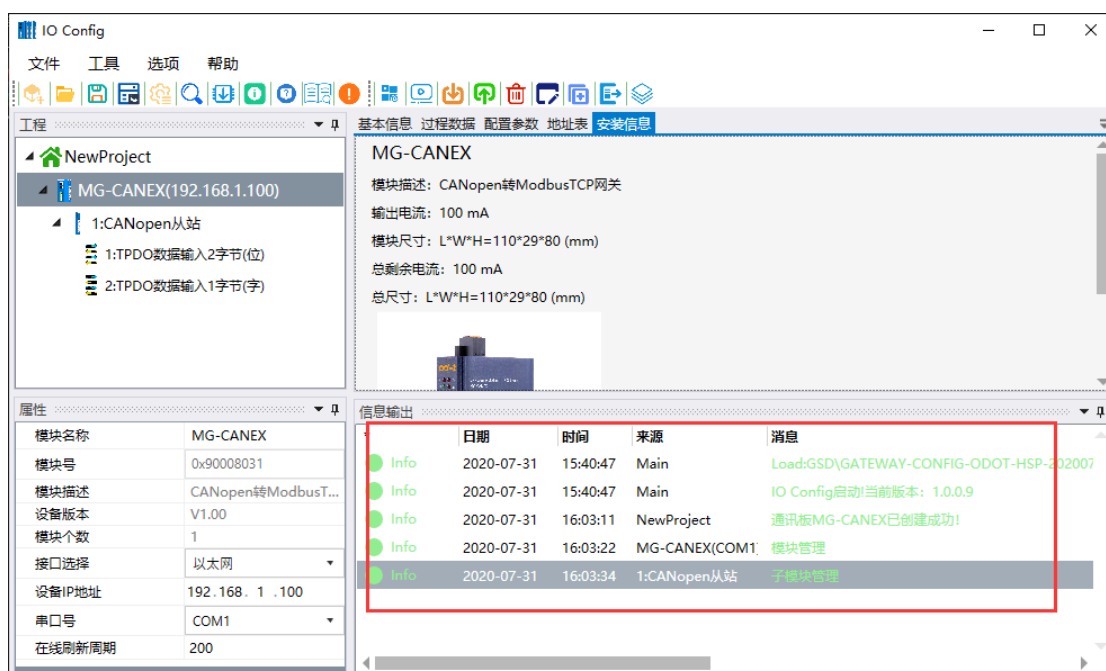


安装信息：可显示网关模块描述、电流消耗、模块尺寸、剩余电流，产品图片。



消息窗口

显示当前操作的实时信息，显示新建工程、上传、下载、配置参数修改、复制粘贴输出等所有的操作日志记录。



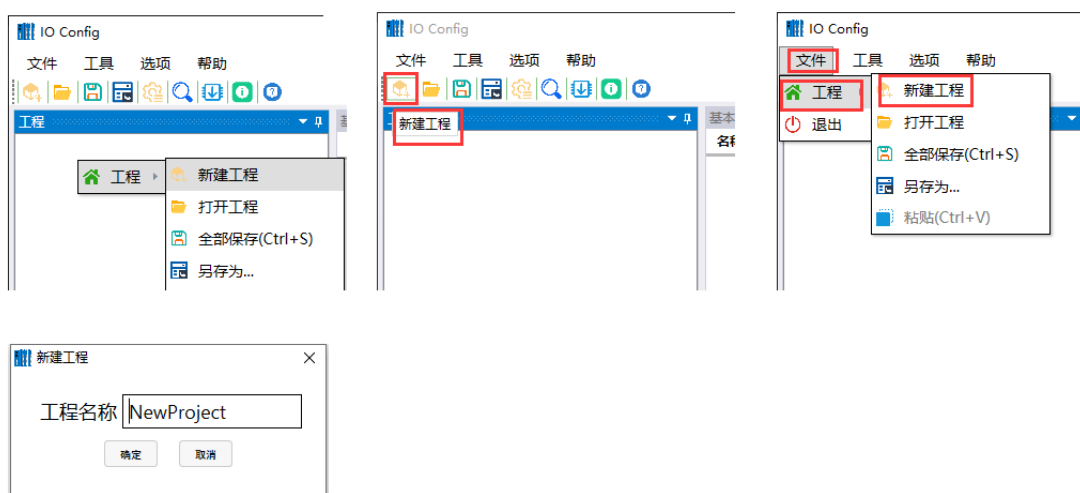
快捷键

快捷键	菜单	描述
F1		进入帮助文档查看远程 IO 硬件手册
Ctrl+C	工程/适配器-复制	复制工程、CN&CT 模块
Ctrl+V	工程/适配器-粘贴	粘贴工程、CN&CT 模块
Delete	工程/适配器-删除	删除工程、CN&CT 模块
Ctrl+S	文件-工程-全部保存	保存配置工程
Ctrl+M	适配器-导出地址表	导出 CN&CT 地址表

5.4 网关参数配置

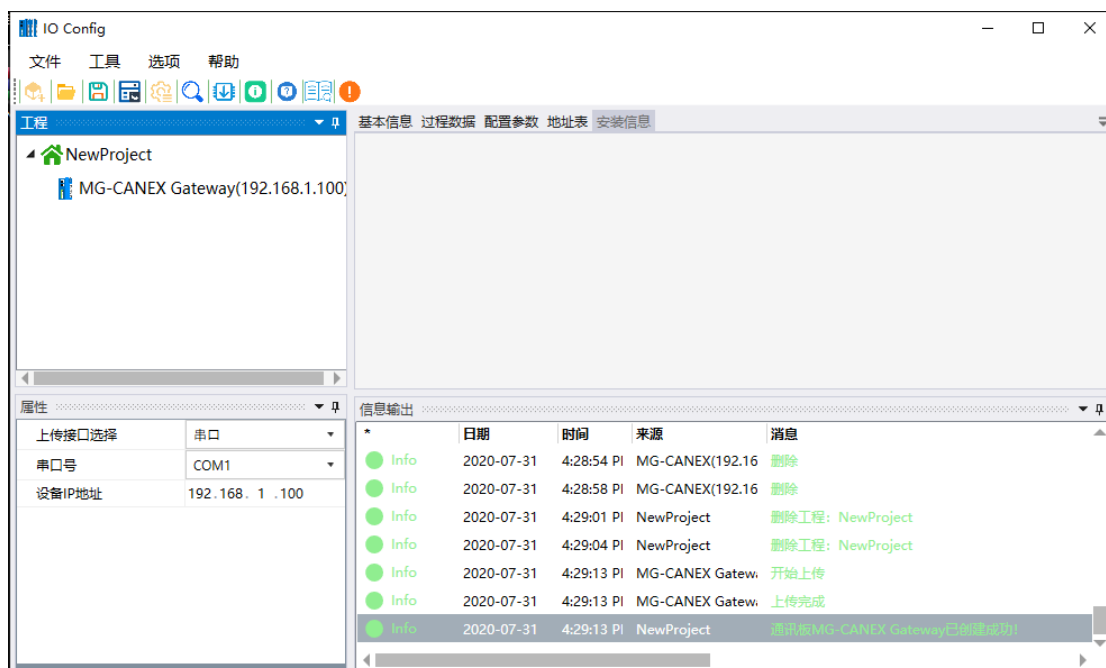
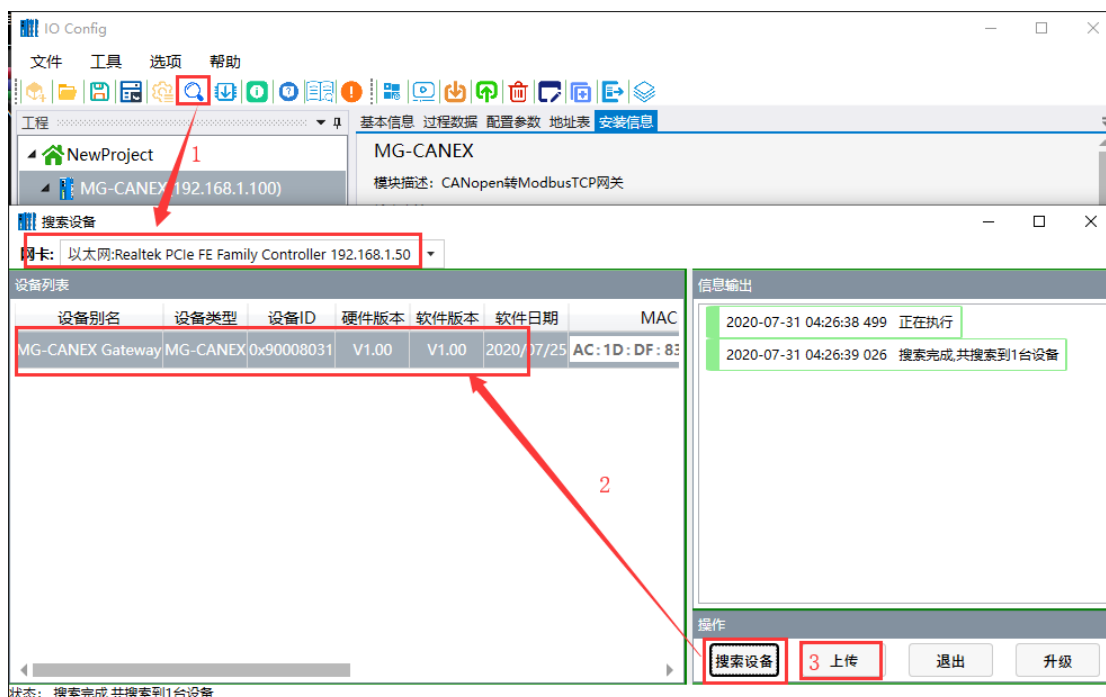
新建工程

双击 IOConfig 快捷图标，新建一个工程，在工程目录栏右键工程-新建工程或点击快捷键或菜单栏文件-工程-新建工程，手动录入工程名称。



搜索网关

将电脑本机网卡设置成 192.168.1.网段，点击快捷键 ，在弹出的界面选择本机网卡。点击右下角“搜索设备”，会在设备列表扫描出网络上的网关。点击上传可以建立一个网关工程项目。



网关参数

选中网关，可点击查看网关的基本信息、过程数据、配置参数、地址表参数。

【基本信息】：选中网关或从站可以查看模块名称，硬件版本，软件版本等相关信息。

【过程数据】：当选中网关后点右键，选择“在线”可以在线监测从站模

块，查看在线监测值。

【配置参数】：选中对应的模块，可以设置模块的通讯参数。

【地址表】：可以查看从站配置的子模块，在 Modbus TCP 网络中对应的地址。

点击配置参数可以查看网关的设备参数、MODBUS TCP 参数、CANopen 参数。



设备参数：

端口镜像使能：禁止、使能可选，默认禁止。

镜像端口选择：LAN1、LAN2 可选，默认 LAN1.当端口镜像功能使能时，此端口用于监视适配器网络报文数据默认

Modbus TCP 参数：

MAC 地址：只读属性。

IP 地址：网关的 IP 地址。

子网掩码：

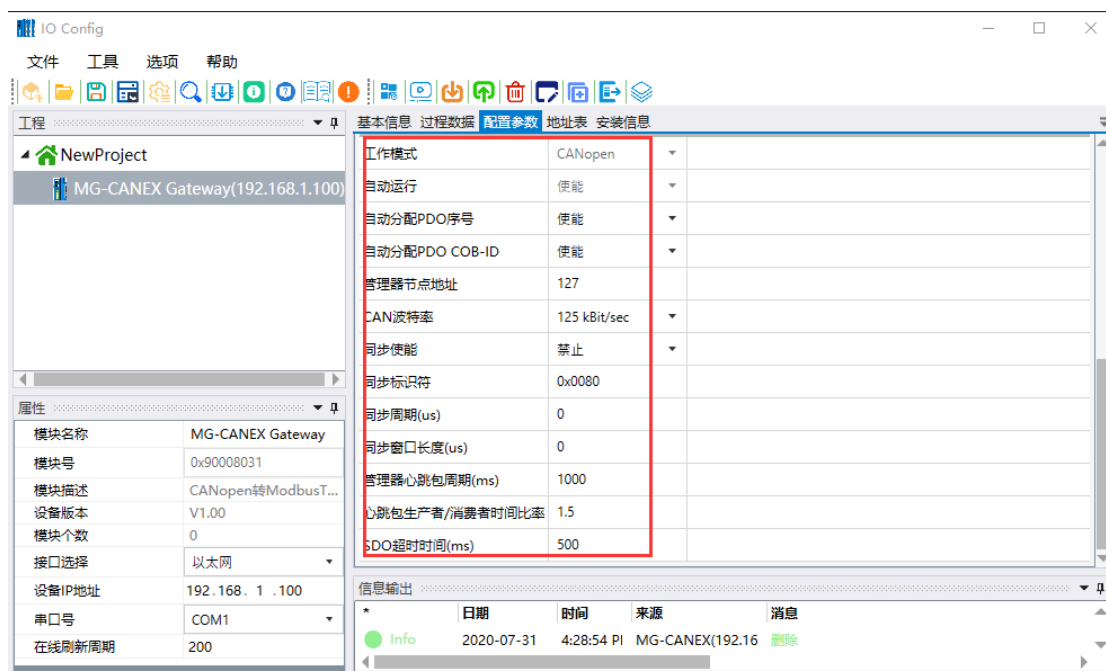
网关：网关地址

Modbus 服务器端口号：502

应用看门狗：禁止、使能可选，默认使能

看门狗时间：当看门狗使能后，若此时间周期内 TCP 连接上没有 Modbus 数

据交换，该 TCP 连接将被断开(其他有数据交换 TCP 连接正常保持)。(默认值：30)



CANopen 参数

工作模式：CANopen

自动运行：使能

自动分配 PDO 序号：使能、禁止可选，默认：使能。

自动分配 PDO COB-ID：使能、禁止可选，默认：使能。

管理节点地址：设置网关在 CANopen 网络中做为主站的节点号，默认：127。

CAN 波特率：设置 CANopen 网络通讯波特率。10k、20k、50k、100k、125k、250k、500k、800k、1MBit/sec 可选，默认 125kBit/sec。

同步使能：禁止、使能可选，默认：禁止。

同步标识符：默认 128。

同步周期 (us)：自定义

同步窗口长度(us)：自定义

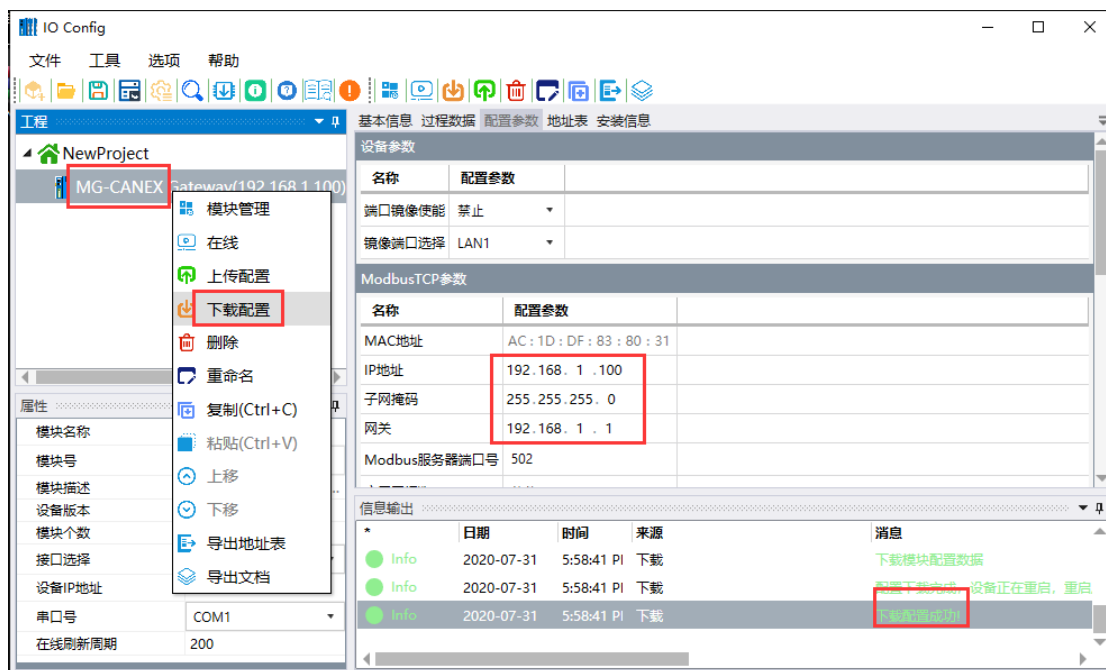
管理器心跳包周期(ms)：默认 1000

心跳包生产者/消费者时间比率：1.5 以上

SDO 超时时间(ms)：默认 500

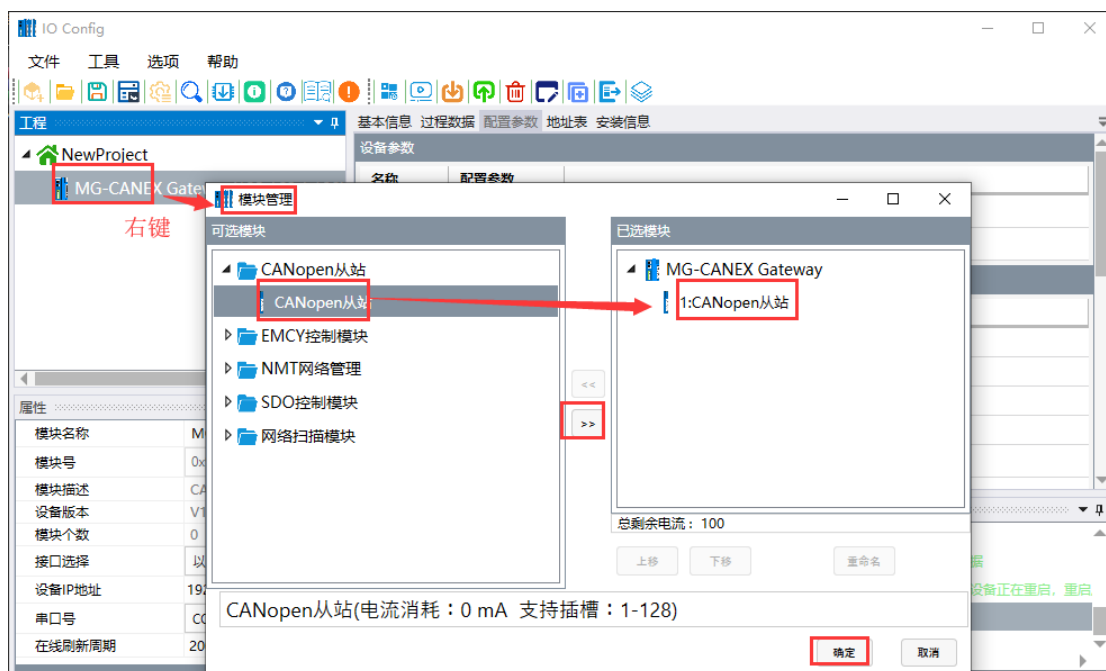
修改网关 IP 地址

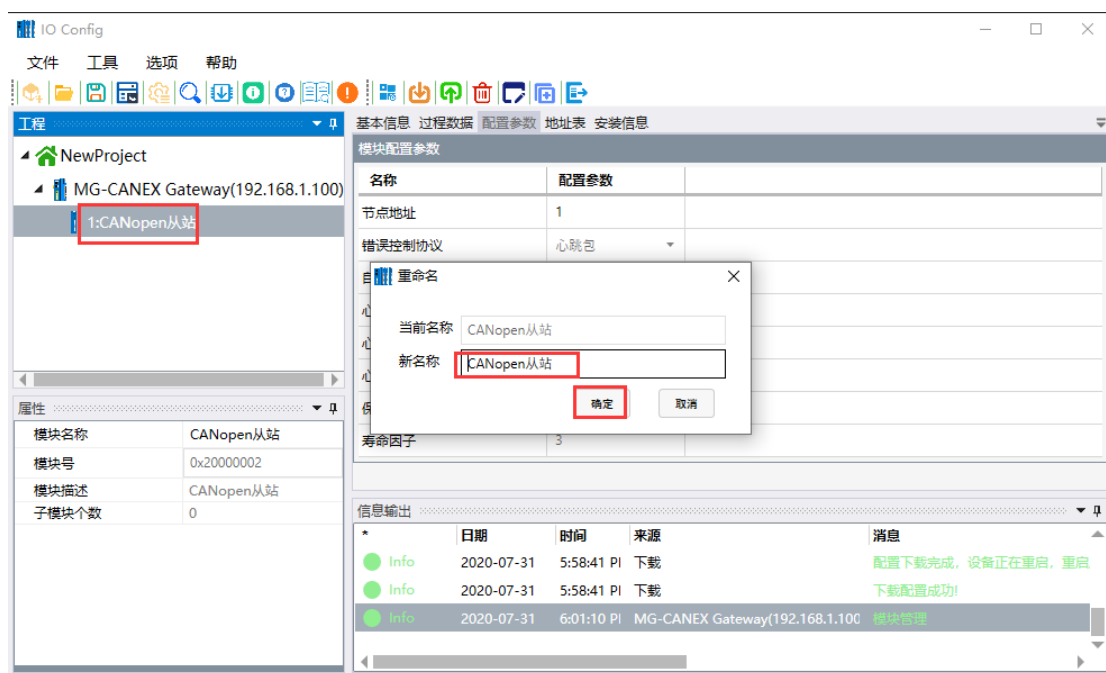
将网关配置参数里的 Modbus TCP 参数里的 IP 地址和网关参数修改成客户需要的 IP 地址。右键网关点击下载配置，即可修改网关的 IP 地址



CANopen 从站参数

右键点击 MG-CANEX Gateway—“模块管理”，在弹出的界面选择 CANopen 从站，点击确定。添加从站后，右键从站点击重命名可以修改从站名称。





选中 CANopen 从站设备，点击主要窗口右侧配置参数，可查看修改设备的配置参数。

节点地址：是 CANopen 从站的站号。

错误控制协议：心跳包

自动运行：使能

心跳包生产者周期（ms）：CANopen 从站发送心跳包时间间隔,默认 1000。

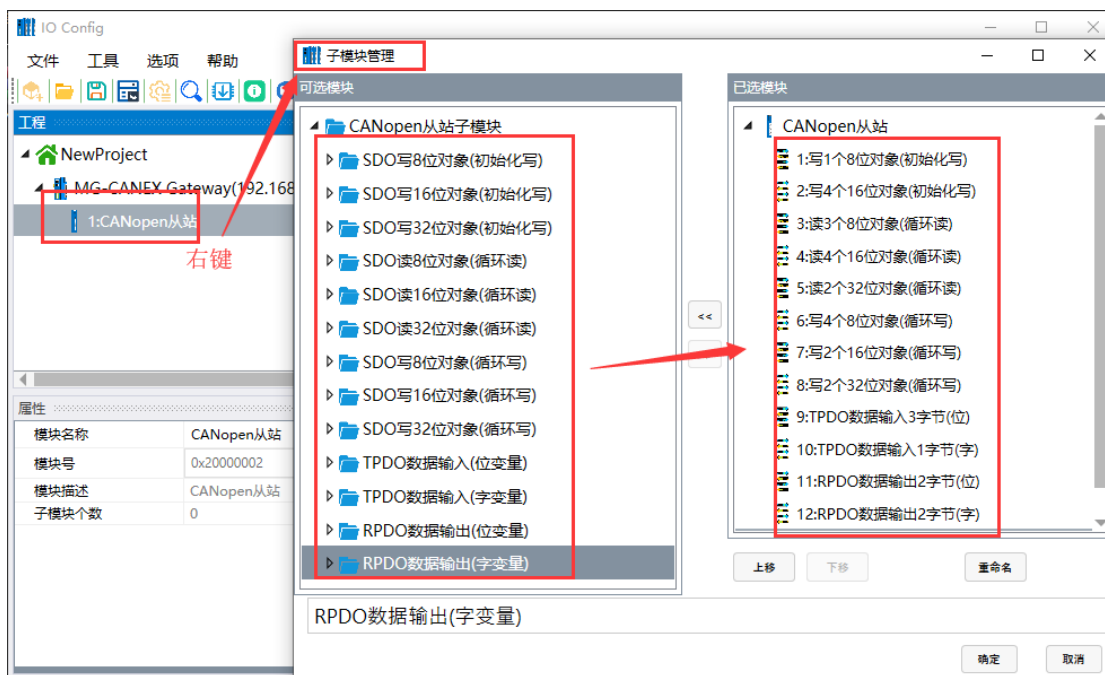
心跳包消费者（节点 ID）：设置由那个站点来检测本从站的心跳包，一般设置为主站的节点号。

心跳包生产者/消费者时间比率：1.5 以上

保护时间（ms）:1000

寿命因子：3

右键 CANopen 从站，点击子模块管理，可添加 SDO 读命令、SDO 写命令、TPDO、RPDO。添加完成后点击确定。

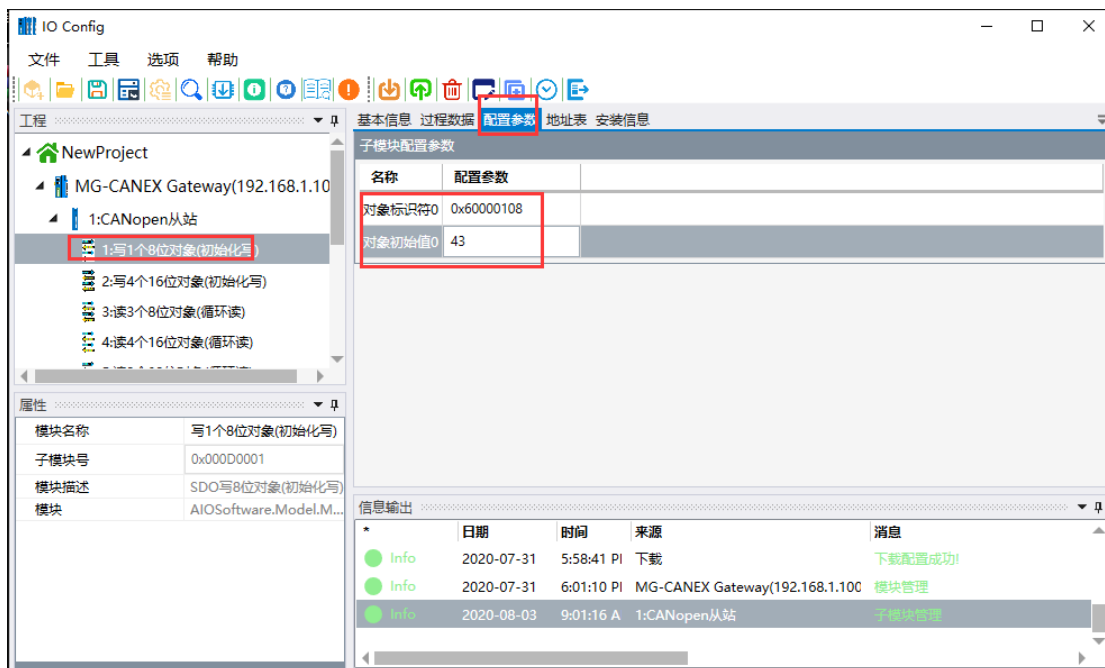


所有的选择最大支持 8 个，即 8 个 8 位对象，8 个 16 位对象，8 个 32 位对象，8 字节（位）、8 字节（字）。多余 8 个后重复调用指令。

服务数据对象 SDO

SDO 写 1 个 8 位对象（初始化写）

SDO 初始化写没有对应的映射地址和过程数据。



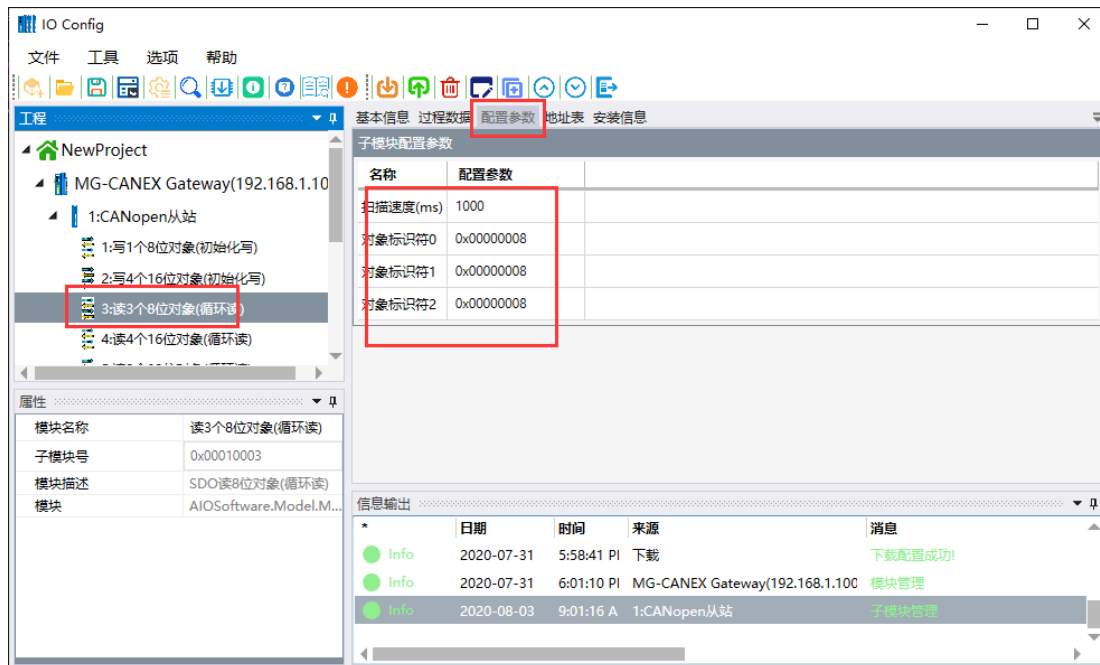
配置参数：

对象标识符：对象索引+对象子索引+位长度

对象初始值：非 0 值。

SDO 读 3 个 8 位对象（循环读）

SDO 读指令映射到 Modbus 的寄存器 3 区地址和过程数据。



配置参数：

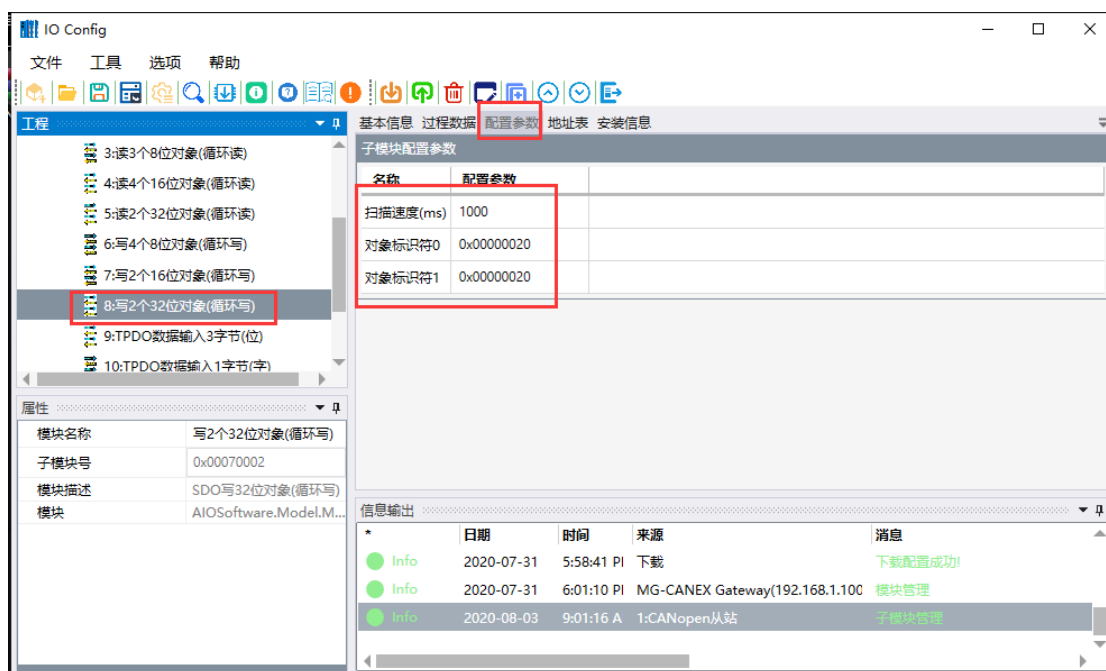
扫描速度（ms）:SDO 读数据的扫描周期，默认 1000

对象标识符 0：对象 0 索引+对象 0 子索引+位长度

对象标识符 1：对象 1 索引+对象 1 子索引+位长度

对象标识符 2：对象 2 索引+对象 2 子索引+位长度

SDO 写 2 个 32 位对象（循环写）



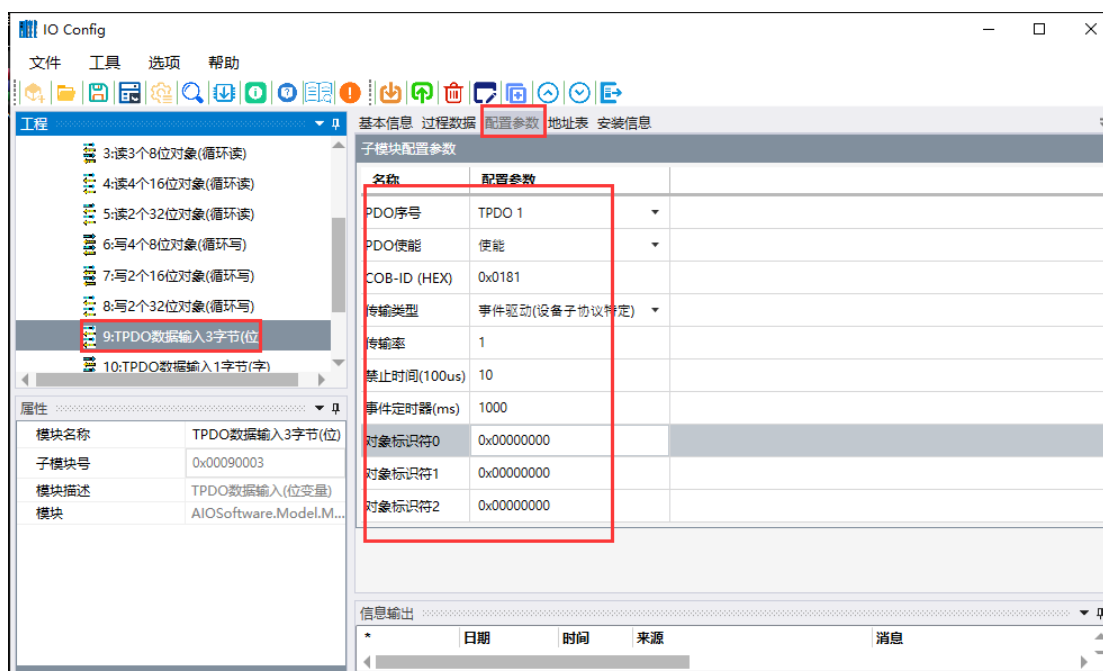
配置参数:

扫描速度 (ms): SDO 读数据的扫描周期, 默认 1000

对象标识符 0: 对象 0 索引+对象 0 子索引+位长度

对象标识符 1: 对象 1 索引+对象 1 子索引+位长度

TPDO 数据输入 3 字节 (位)



配置参数:

PDO 序号: TPDO1-TPDO64, 自定义, 不能重复调用。

PDO 使能：使能、禁止可选，默认：使能。

COB-ID(HEX)：0X180h+\$NODEID、0X280h+\$NODEID、0X380h+\$NODEID、0X480h+\$NODEID、0X580h+\$NODEID.....。不能重复调用，与 PDO 序号配合使用

传输类型：同步（非循环）、同步（循环），事件驱动（制造商特定）、事件驱动（设备自协议特定）可选，默认事件驱动（设备自协议特定）

传输率：自定义

禁止时间（100us）：自定义

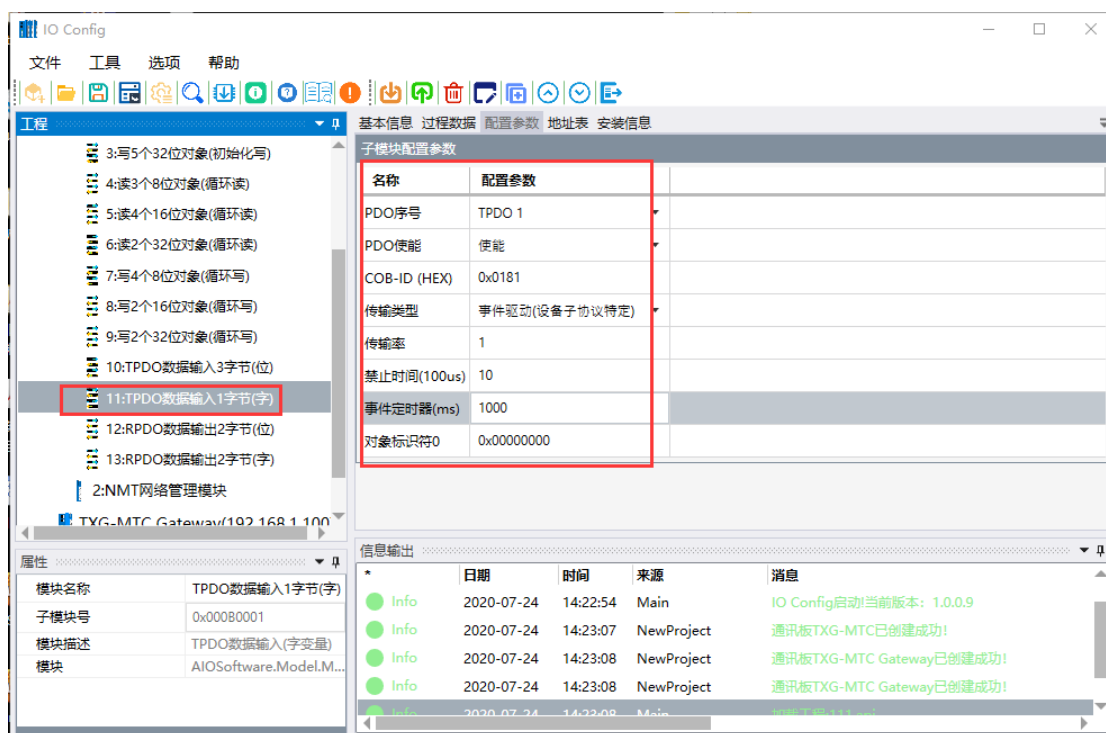
事件定时器（ms）：自定义

对象标识符 0：对象 0 索引+对象 0 子索引+位长度

对象标识符 1：对象 1 索引+对象 1 子索引+位长度

对象标识符 2：对象 2 索引+对象 2 子索引+位长度

TPDO 数据输入 1 字节（字）



配置参数：

PDO 序号：TPDO1-TPDO64,自定义，不能重复调用。

PDO 使能：使能、禁止可选，默认：使能。

COB-ID(HEX)：0X180h+\$NODEID、0X280h+\$NODEID、

0X380h+\$NODEID、0X480h+\$NODEID、0X580h+\$NODEID.....。不能重复调用，与 PDO 序号配合使用

传输类型：同步（非循环）、同步（循环），事件驱动（制造商特定）、事件驱动（设备自协议特定）可选，默认事件驱动（设备自协议特定）

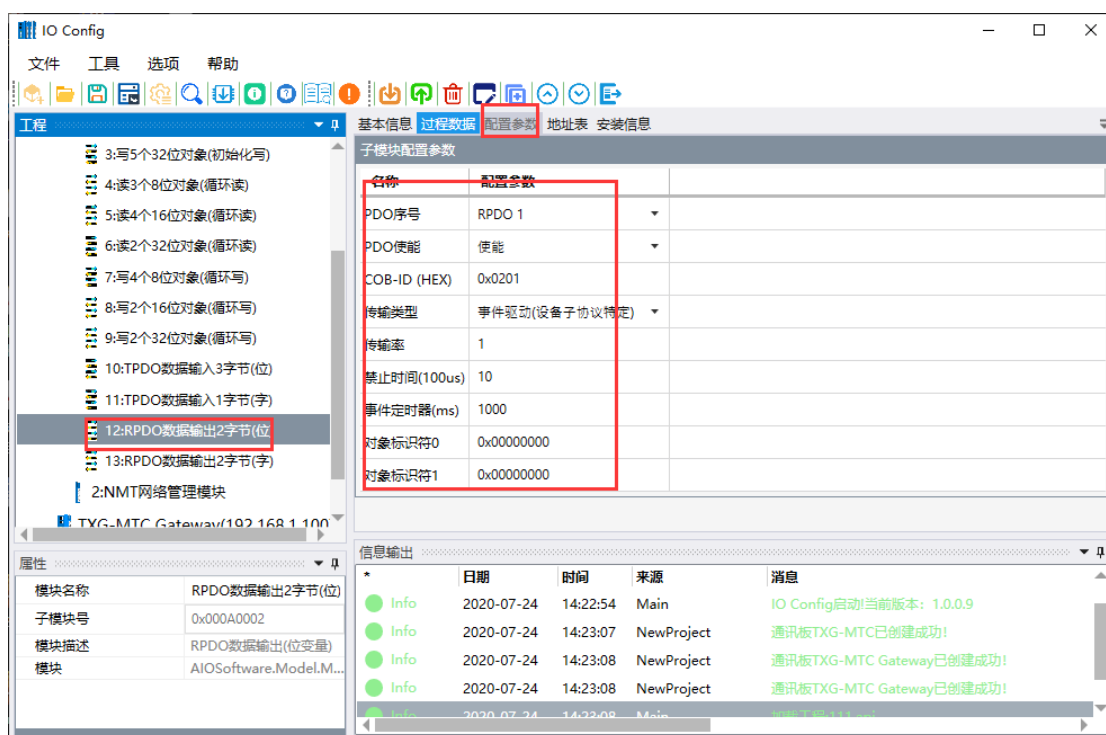
传输率：自定义

禁止时间（100us）：自定义

事件定时器（ms）：自定义

对象标识符 0：对象 0 索引+对象 0 子索引+位长度

RPDO 数据输出 2 字节(位)



配置参数：

PDO 序号：RPDO1-RPDO64,自定义，不能重复调用。

PDO 使能：使能、禁止可选，默认：使能。

COB-ID(HEX)：0X200h+\$NODEID、0X300h+\$NODEID、0X400h+\$NODEID、0X500h+\$NODEID、0X600h+\$NODEID.....。不能重复调用，与 PDO 序号配合使用

传输类型：同步（非循环）、同步（循环），事件驱动（制造商特定）、事件

驱动（设备自协议特定）可选，默认事件驱动（设备自协议特定）

传输率：自定义

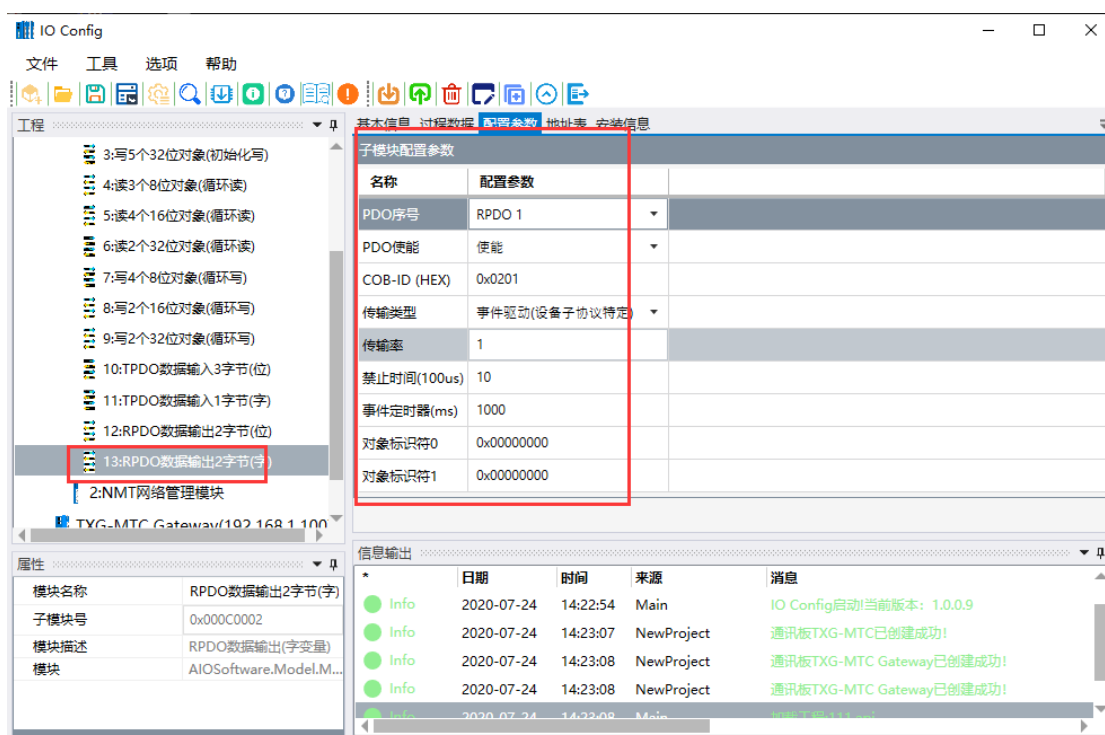
禁止时间（100us）：自定义

事件定时器（ms）：自定义

对象标识符 0：对象 0 索引+对象 0 子索引+位长度

对象标识符 1：对象 1 索引+对象 1 子索引+位长度

RPDO 数据输出 2 字节(字)



配置参数：

PDO 序号：RPDO1-RPDO64,自定义，不能重复调用。

PDO 使能：使能、禁止可选，默认：使能。

COB-ID(HEX)：0X200h+\$NODEID、0X300h+\$NODEID、0X400h+\$NODEID、0X500h+\$NODEID、0X600h+\$NODEID.....。不能重复调用，与 PDO 序号配合使用

传输类型：同步（非循环）、同步（循环），事件驱动（制造商特定）、事件驱动（设备自协议特定）可选，默认事件驱动（设备自协议特定）

传输率：自定义

禁止时间 (100us): 自定义

事件定时器 (ms): 自定义

对象标识符 0: 对象 0 索引+对象 0 子索引+位长度

对象标识符 1: 对象 1 索引+对象 1 子索引+位长度

配置下载、上传

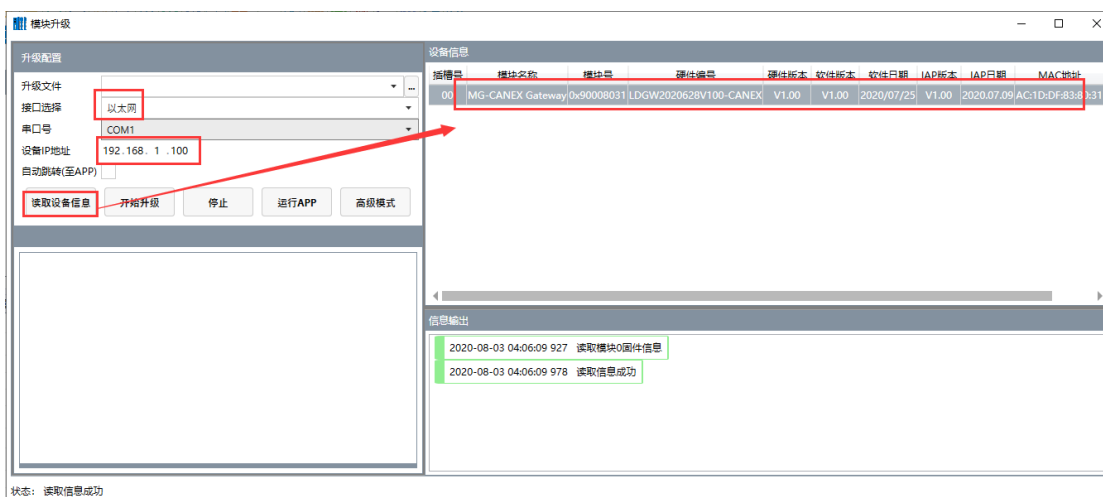
配置参数修改完成后, 右键网关, 点击“下载配置”, 就可以将所有配置下载到网关中。下载成功后右键网关, 点击“上传配置”, 可以做到分配 PDO 号和 COB-ID 号, 也可以上传上 CANOPEN 从站设备自带的映射配置。

数据采集控制可以通过访问 Modbus 地址, 直接控制 CANopen 从站设备。

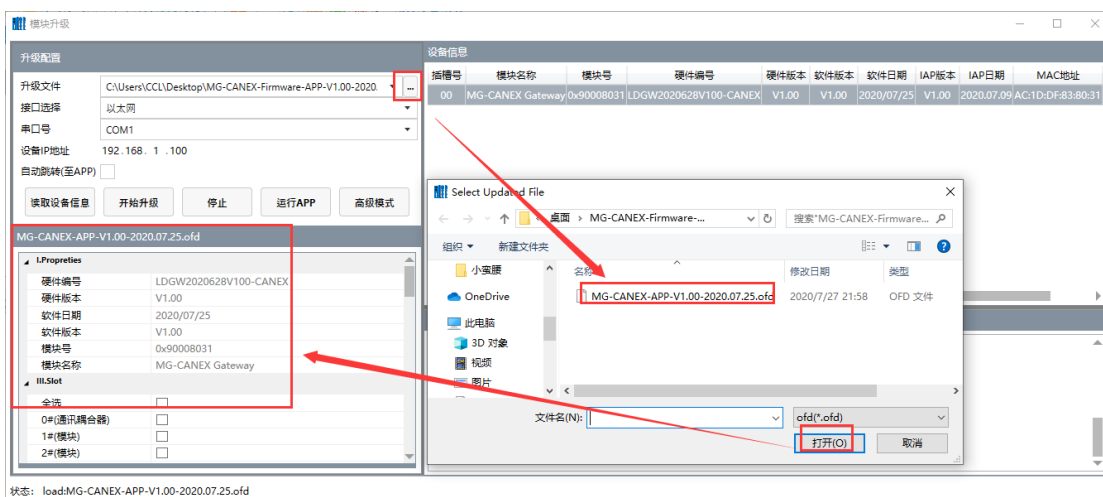
6 固件升级

当模块固件更新，需要给模块固件升级。

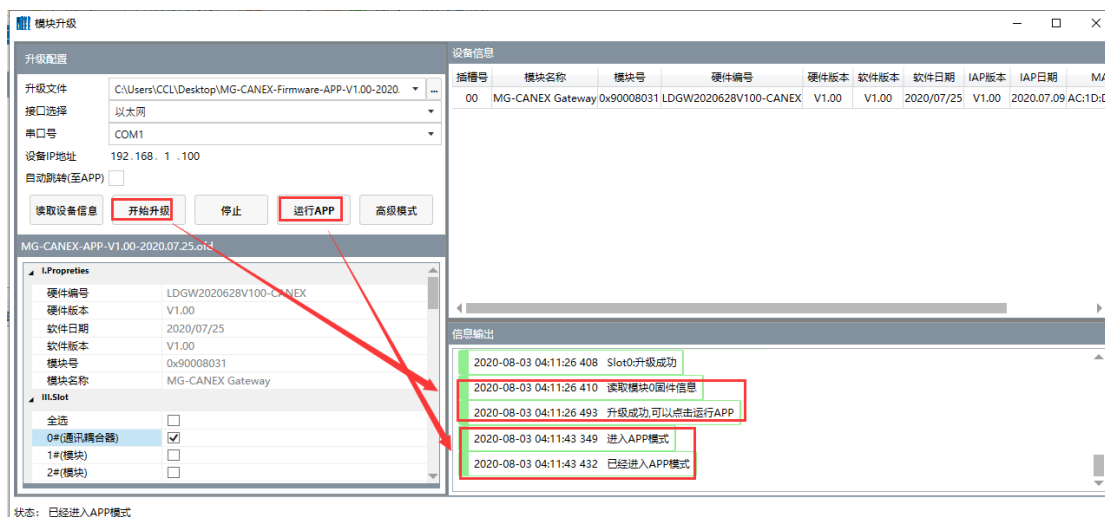
打开 IO Config 软件，点击工具在线升级或点击快捷键 ，接口选择以太网，设备 IP 地址，点击读取设备信息，可以读取到网关内部固件信息。



点击 , 在弹出的界面，选择新的固件文件，点击打开，会在左下角显示新固件信息。



选中 0#耦合器，打“√”，点击开始升级，完成后点击运行 APP。或者选中自动跳转（至 APP）点击开始升级。



进入 APP 模式时，所有指示灯会闪烁亮一次。

7 产品应用实例

7.1 网关与分布式 IO 通讯示例

本例介绍将我司的协议转换器，把带 CANopen 协议的远程 IO 和带 Modbus TCP 协议的上位机进行通讯。

本例需要的硬件：

CANopen 转 Modbus TCP 协议转换器一台、带 CANopen 协议的远程 IO、24V 开关电源、笔记本一台

本例需要的软件：

上位机软件 Modbus poll，协议转换器调试软件 IO Config，EDS 文件阅读软件：edseditor20de，远程 IO 的 EDS 文件。

硬件接线

将本网网关口和电脑网口连接，并确保电脑在 192.168.1 网段。网关接上 24V 电源，远程 IO 接上电源，将网关的 CAN_L 与远程 IO 的 CAN_L 连接、CAN_H 与 CAN_H、GND 与 CAN_GND 连接。确认无误后上电。

远程 IO 地址查询

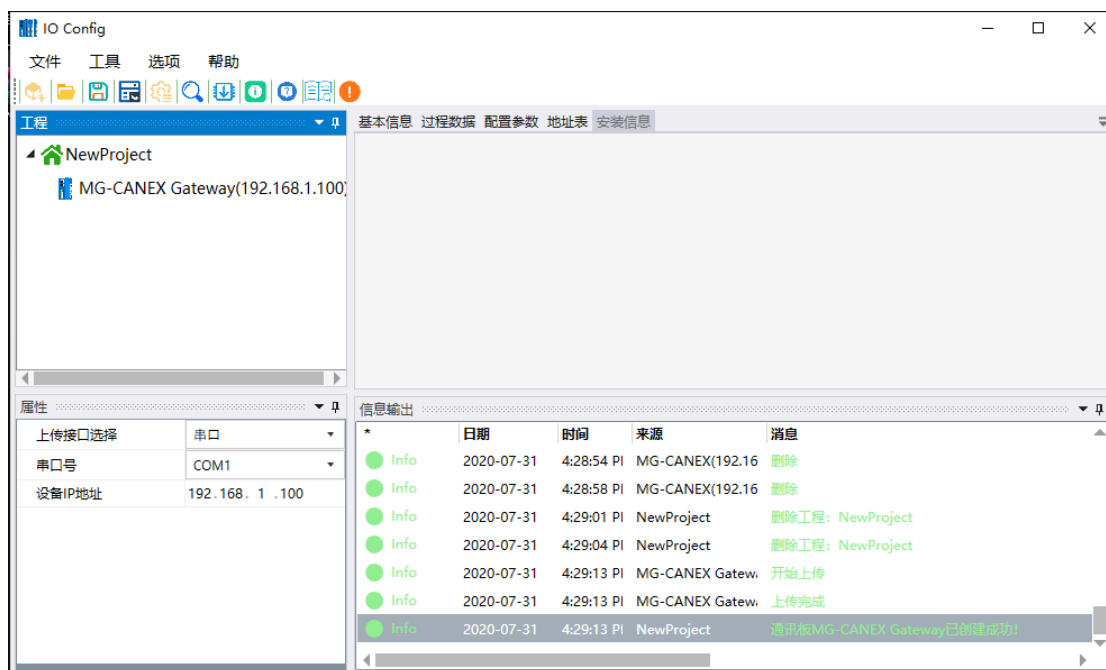
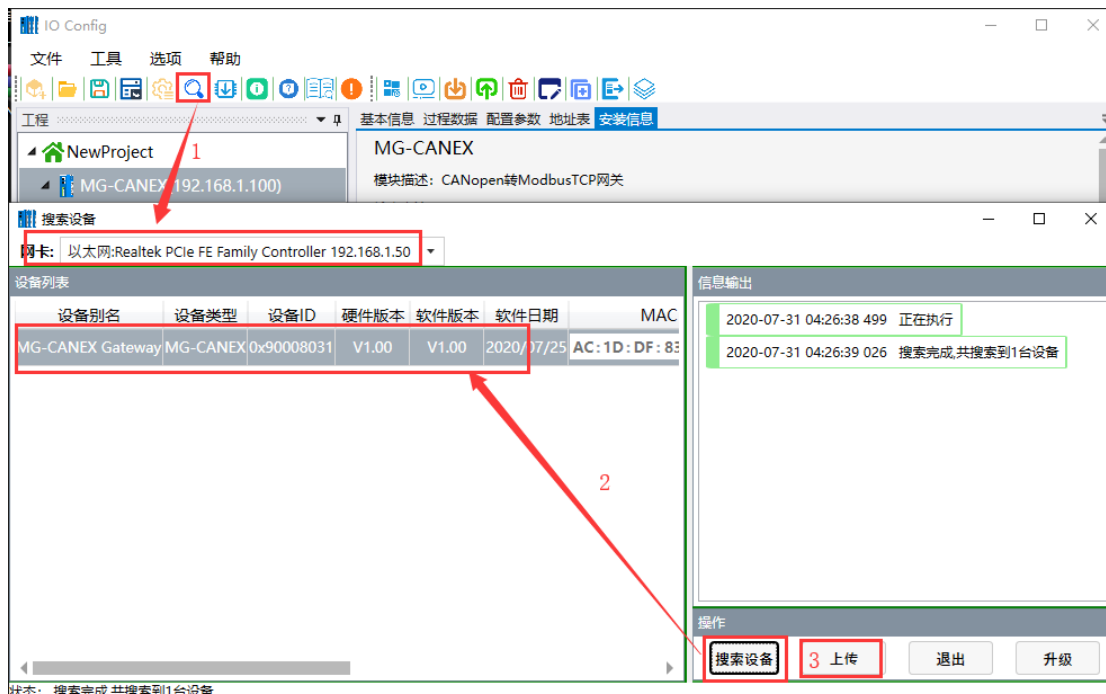
本例远程 IO 的硬件组态为：一个远程 IO 适配器，一个 8DI 数字量输入，一个 8DO 数字量输出，一个 4AI 模拟量输入，一个 4AO 模拟量输出

安装并打开软件：edseditor20de，用软件打开远程 IO 的 EDS 文件，查询“对象字典”得到相关地址如下：（对象标识符）

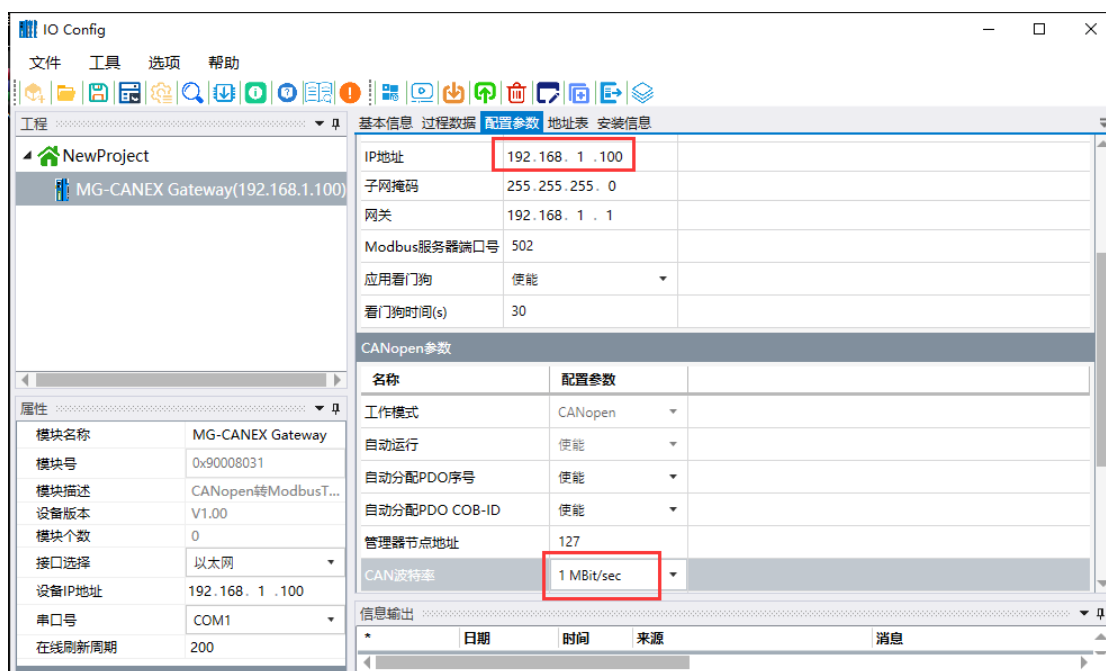
对象索引	对象子索引	位长度	属性	名称	数据类型
0x6000	0x01	0x08	RO	8DI 数字量输入	Unsigned8
0x6401	0x01	0x10	RO	模拟量输入 1	integer16
0x6401	0x02	0x10	RO	模拟量输入 2	integer16
0x6401	0x03	0x10	RO	模拟量输入 3	integer16
0x6401	0x04	0x10	RO	模拟量输入 4	integer16
0x6200	0x01	0x08	RW	8DO 数字量输出	Unsigned8
0x6411	0x01	0x10	RW	模拟量输出 1	integer16
0x6411	0x02	0x10	RW	模拟量输出 2	integer16
0x6411	0x03	0x10	RW	模拟量输出 3	integer16
0x6411	0x04	0x10	RW	模拟量输出 4	integer16

网关配置

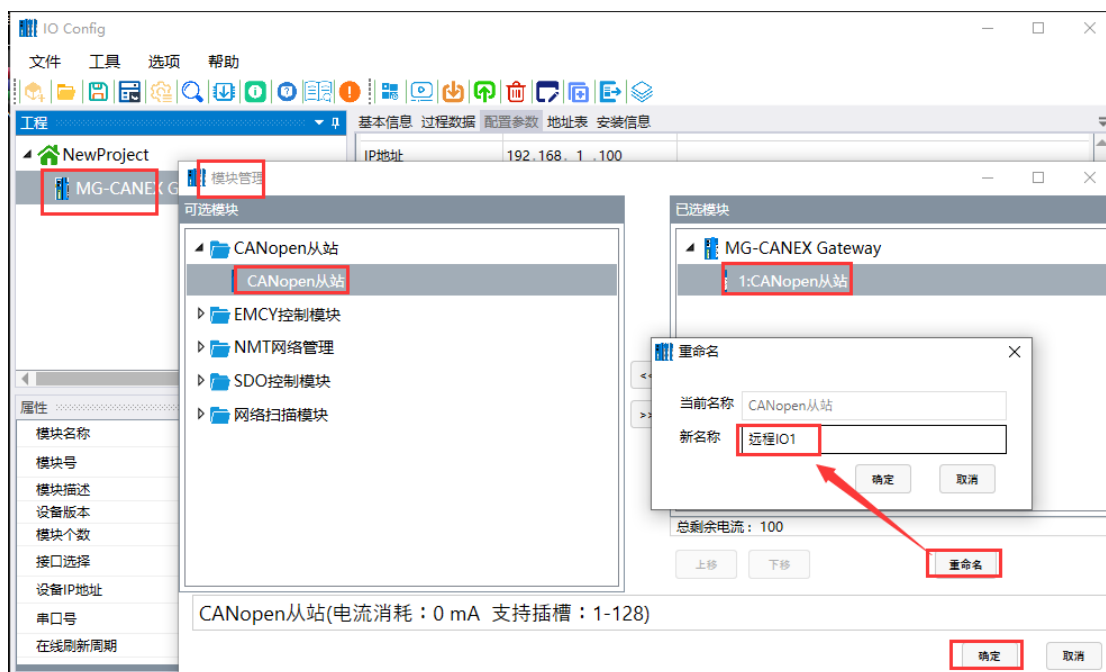
打开《IO Config》配置软件.点击搜索，在弹出的界面选择本机网卡，点击搜索设备，会在设备列表扫描出网关设备，点击上传。

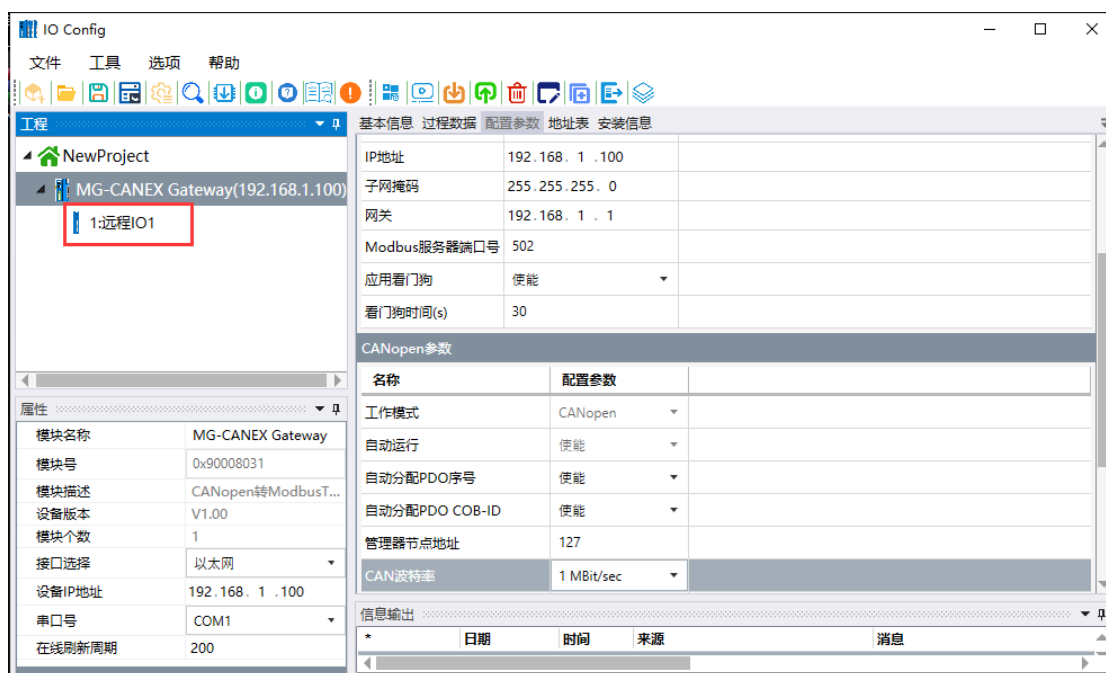


(3)选中添加的网关在右边窗口的工具栏选择【配置参数】对网关进行参数设置。IP 地址：192.168.1.100，CANopen 波特率选 1M bit/sec。

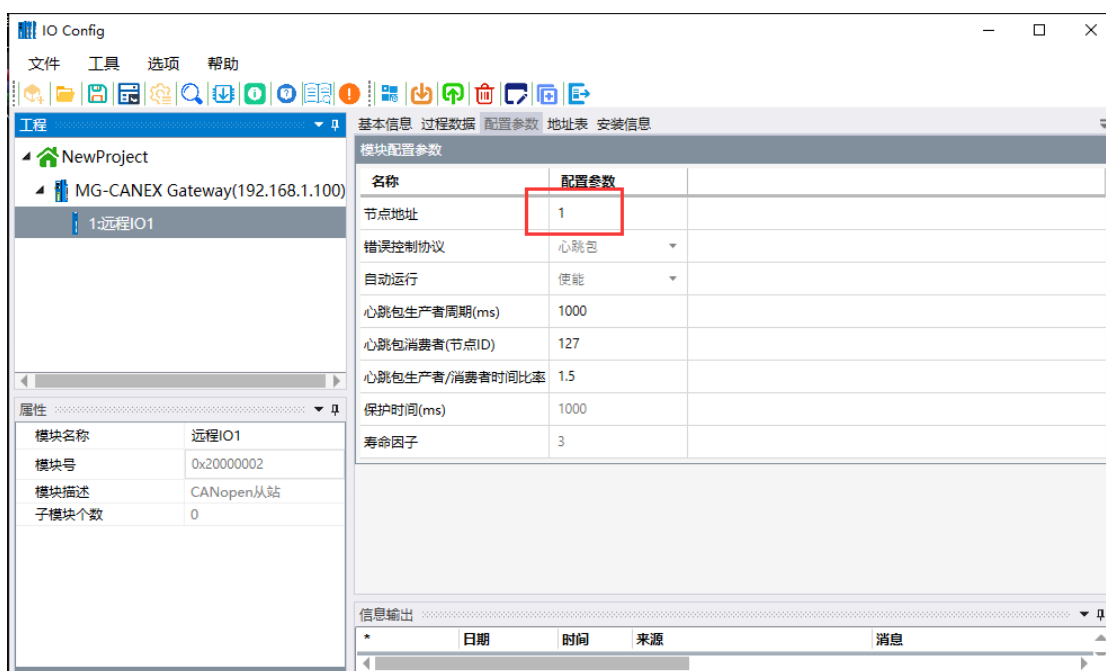


(4) 选中添加的网关，右键选择【模块管理】双击或点击  添加 CANopen 从站。并修改从站名称为：远程 IO1。





(5)选中添加的从站“远程 IO1”，在右边窗口的点击【配置参数】将从站节点地址设置为 1，其余默认。



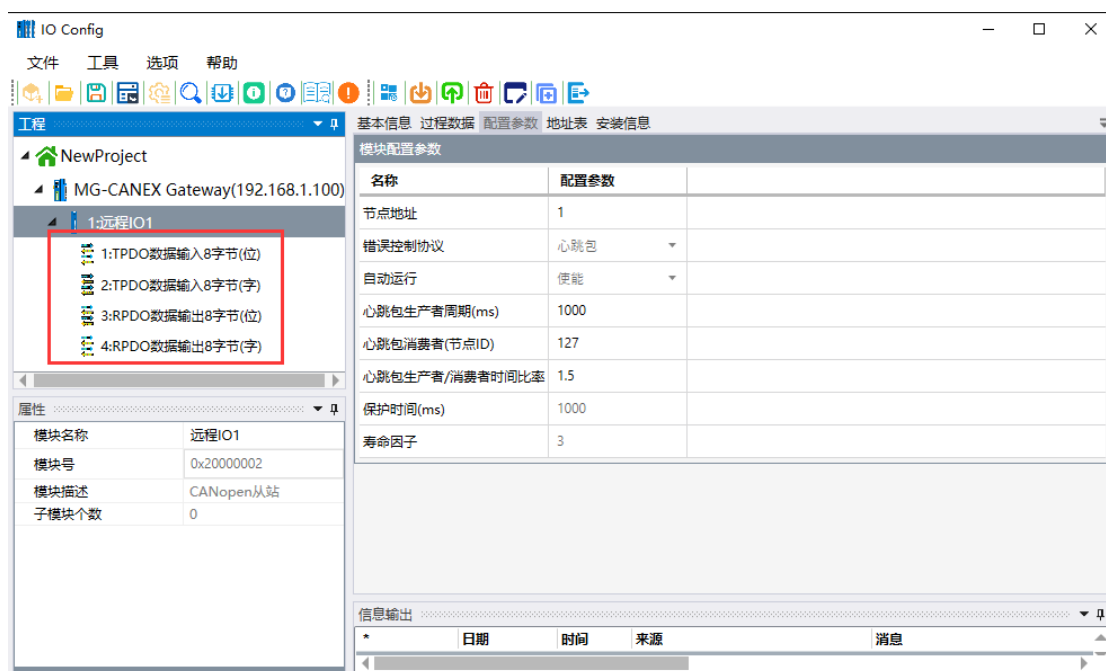
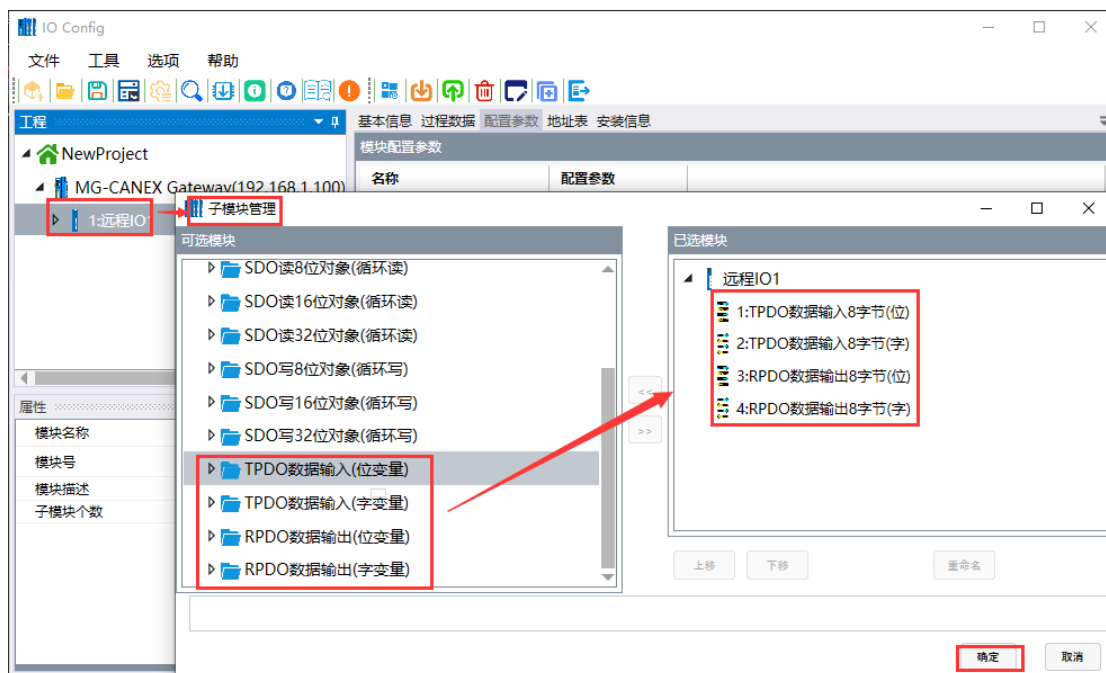
(6)选中添加的从站，右键选择【子模块管理】，双击或点击  添加 CANopen 从站读指令：已选模块选择完成后，点击确定。

TPDO 数据输入 8 字节（位）

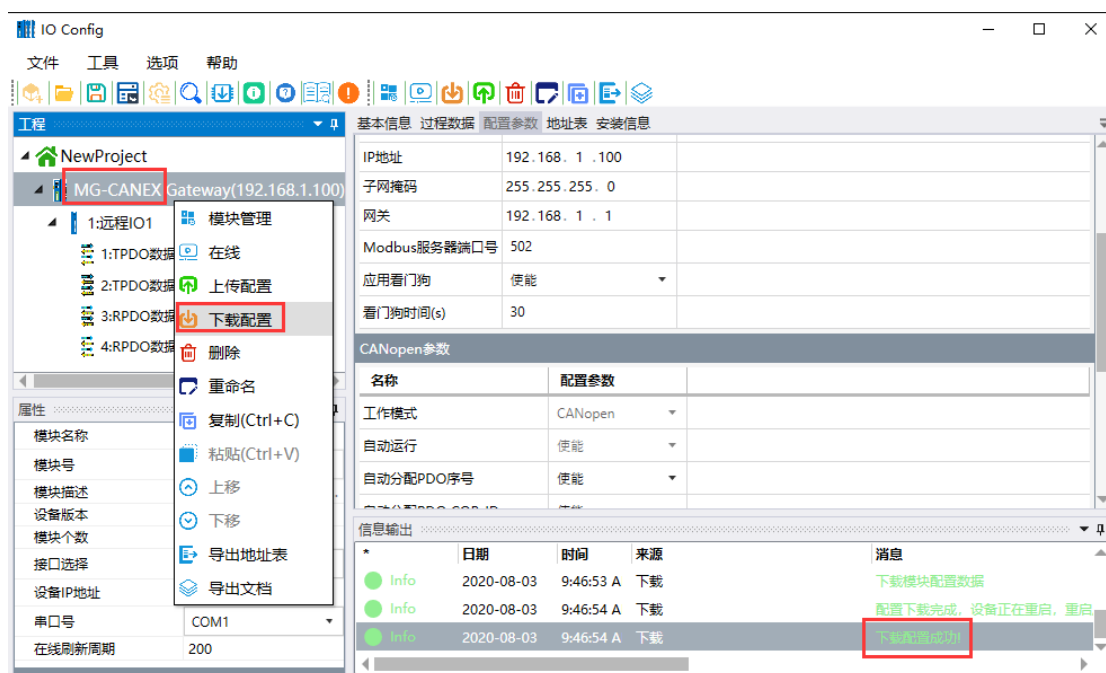
TPDO 数据输入 8 字节（字）

RPDO 数据输入 8 字节（位）

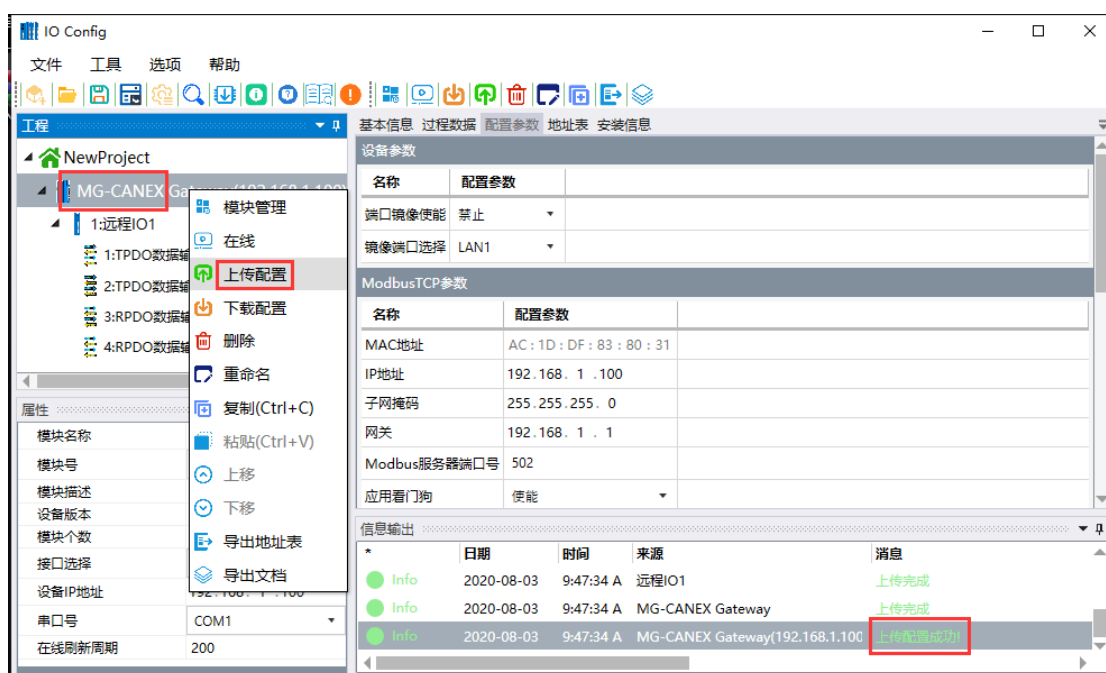
RPDO 数据输入 8 字节（字）

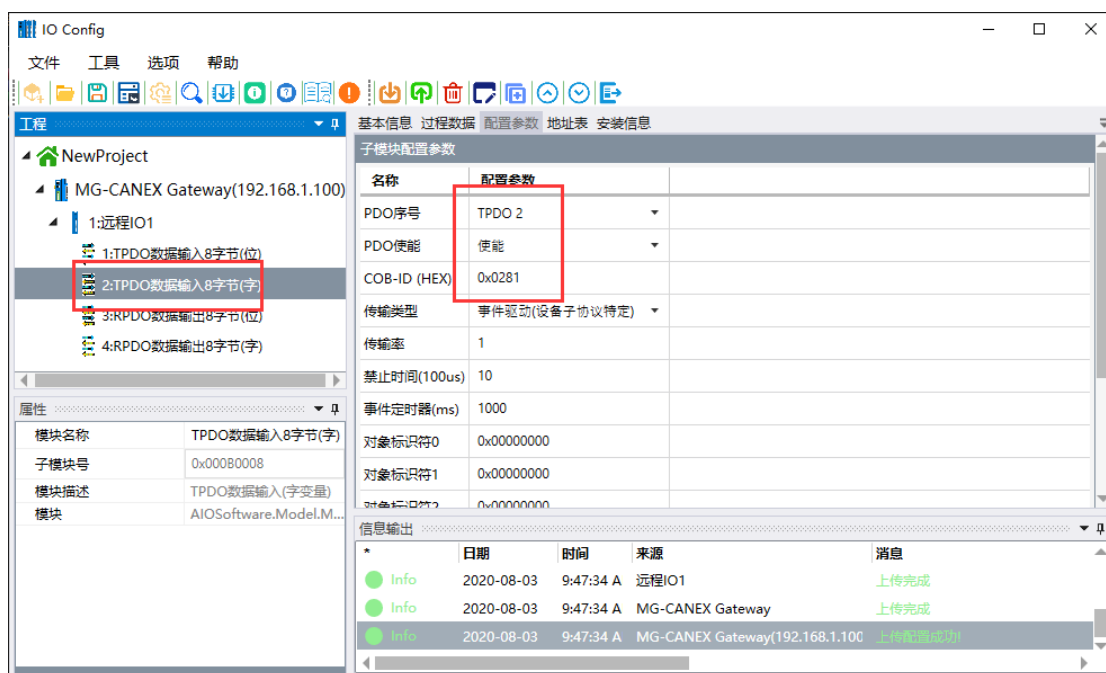


所有配置完成后，右键 MG-CANEX，点击下载配置。

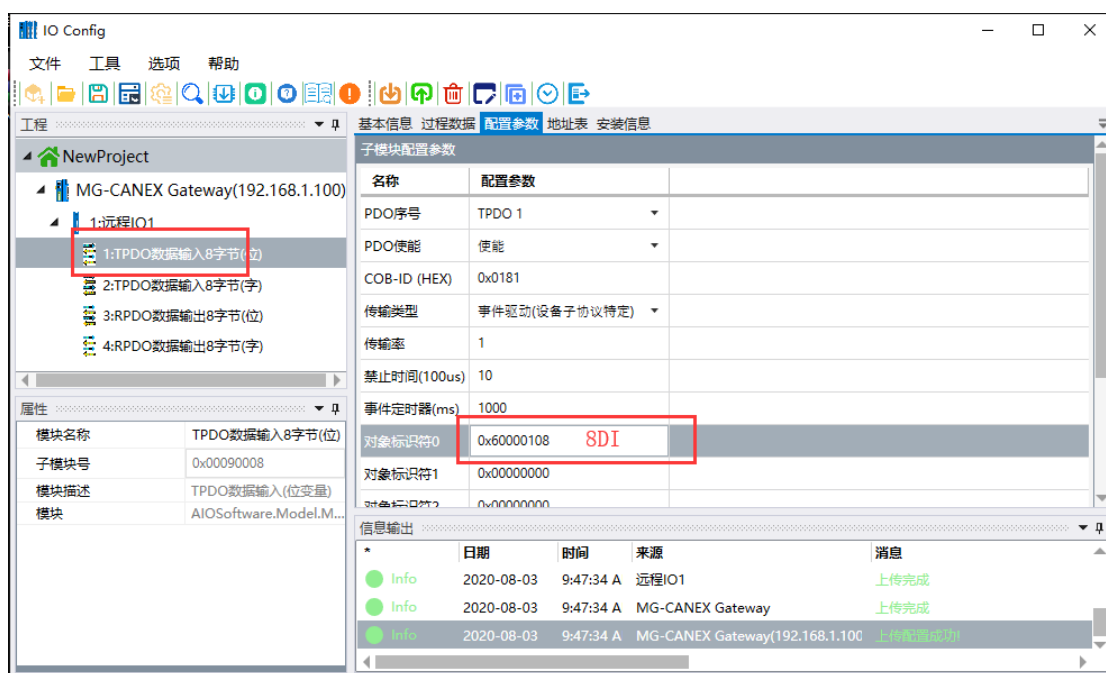


下载完成后，右键 MG-CANEX，点击上传配置，可以自动分配好 PDO 序号和 COB-ID 号。

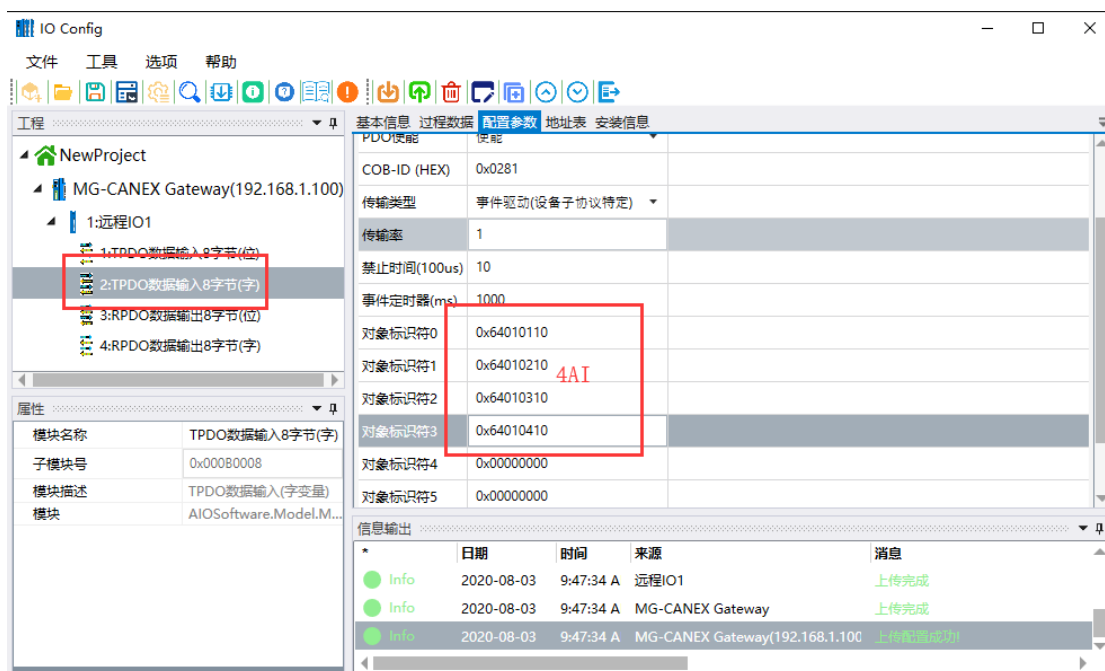




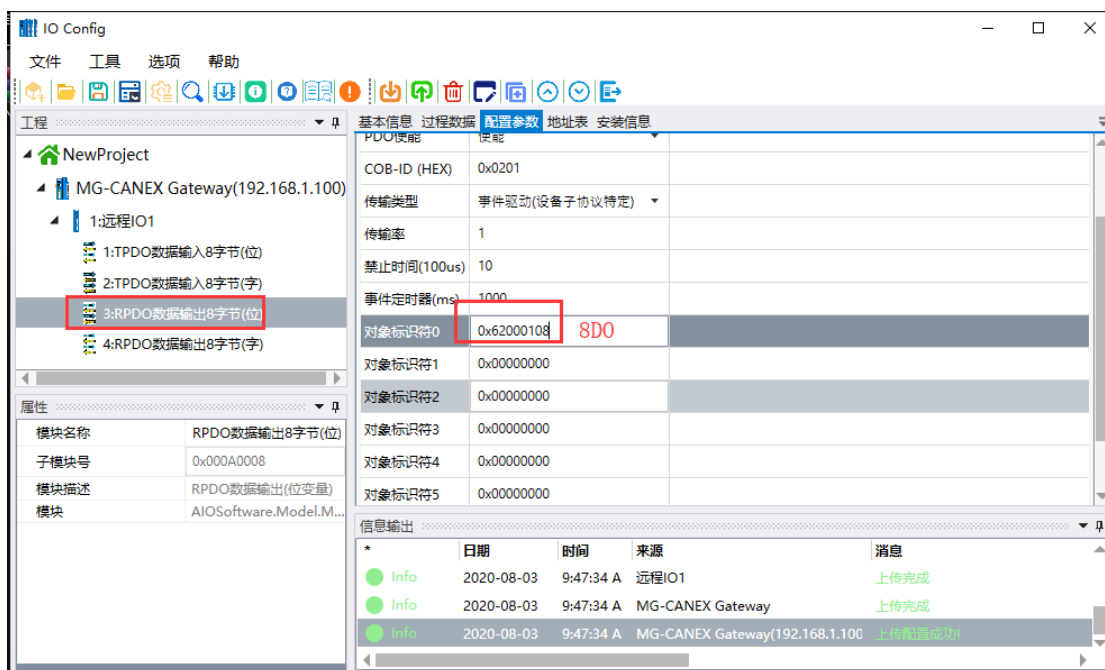
(7) 选中添加的 PDO 指令，点击【配置参数】，配置每个指令的对象标识符参数。



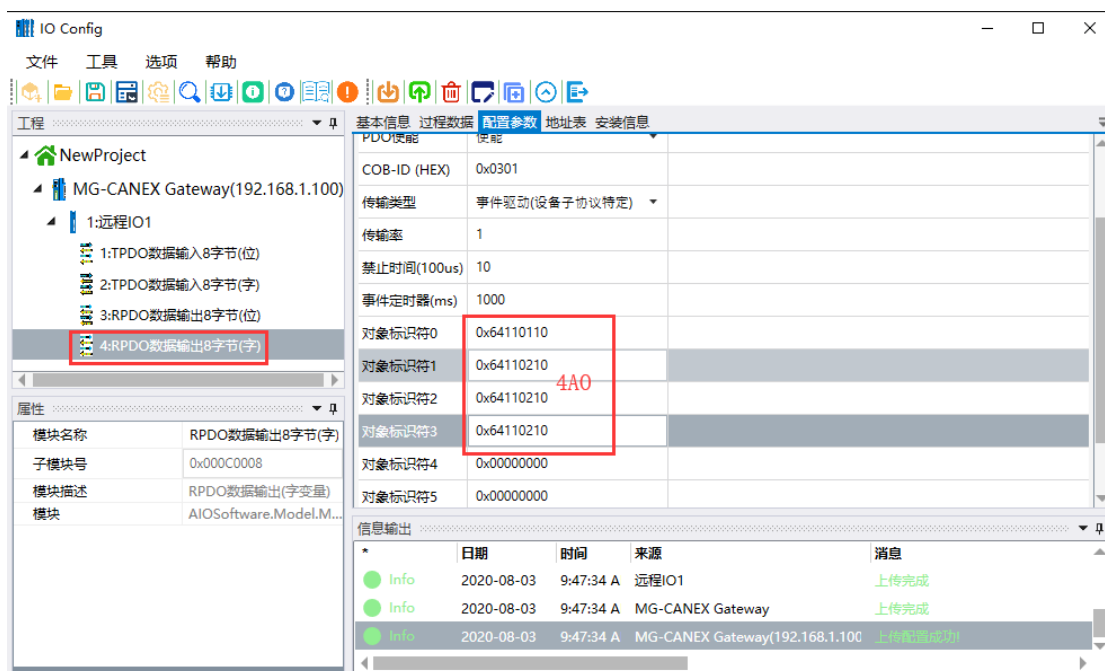
对 TPDO 数据输入 8 字节（字）指令配置



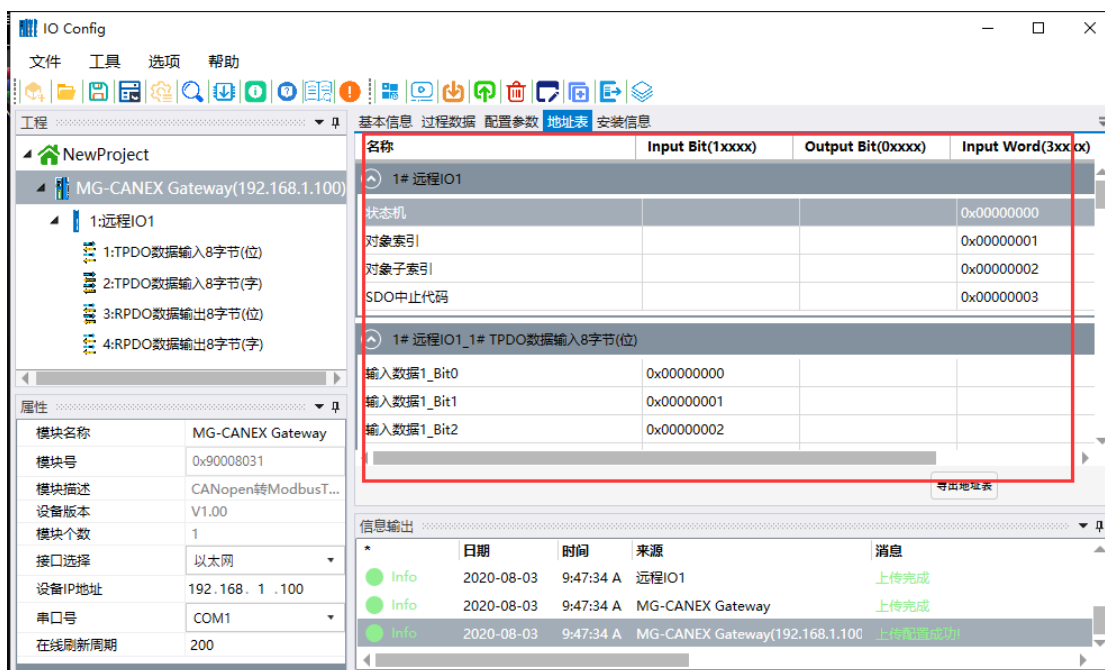
对 RPDO 数据输入 8 字节（位）指令配置



对 RPDO 数据输入 8 字节（字）指令配置



(8) 选中添加的指令选择【地址表】查看读取的参数所对应的 Modbus 地址

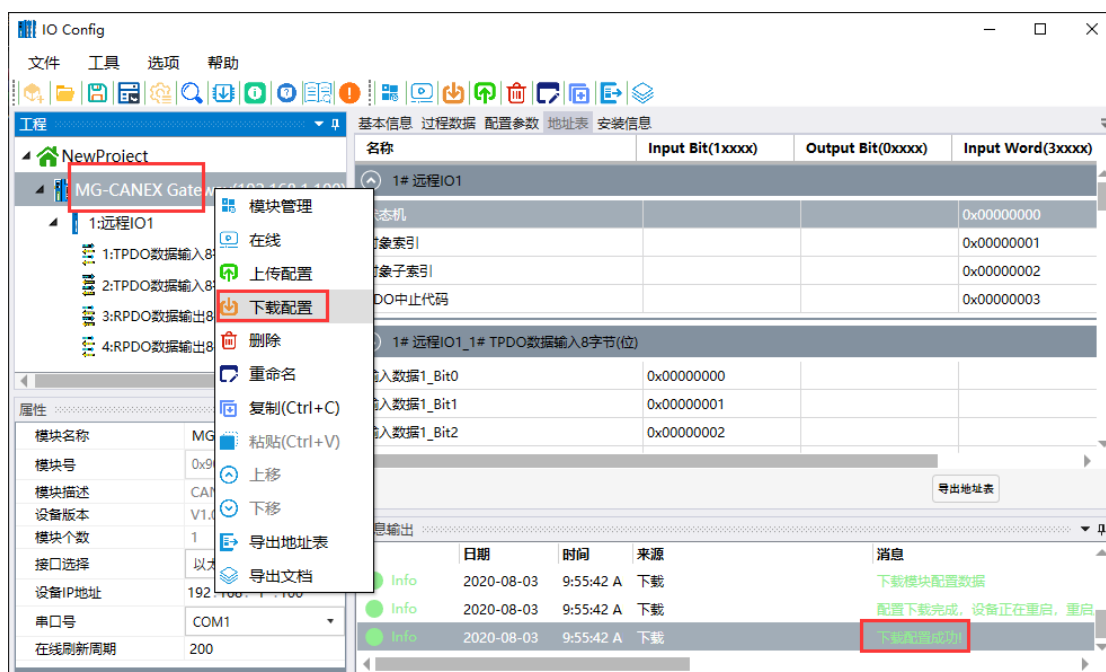


【在线】TXG-MTC_Gateway(192.168.1.100)_地址表_2020-07-28_103016 - 记事本

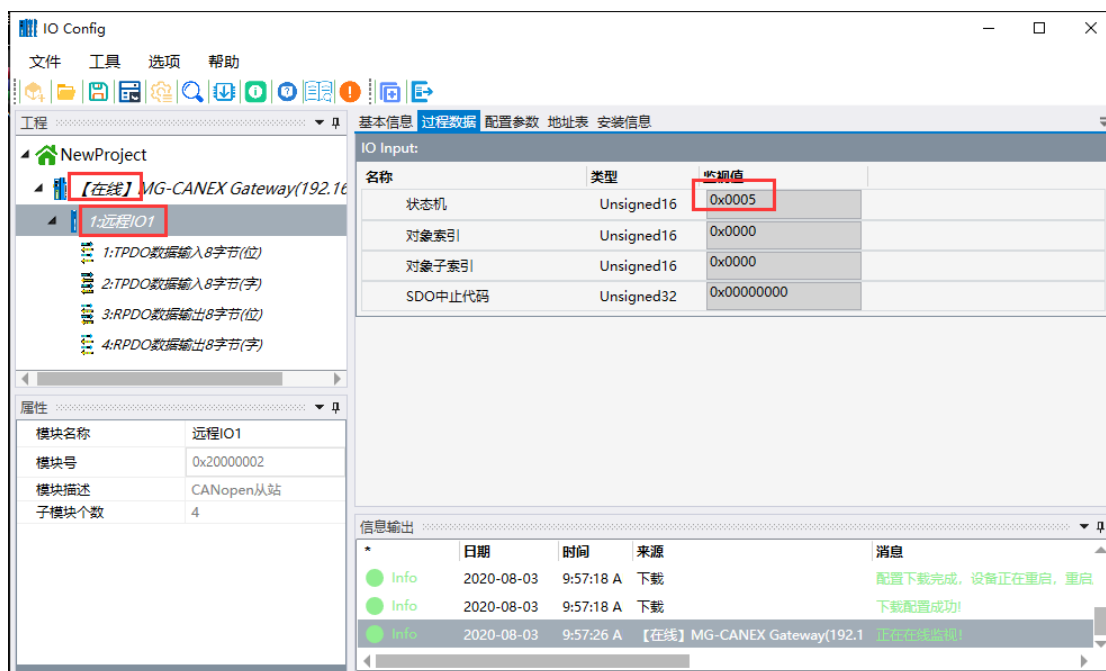
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

1# 远程IO1			
数据名称:状态机	寄存器区域:输入寄存器 (3x)	数据起始地址:0x00000000(16进制)	0(10进制)
数据名称:对象索引	寄存器区域:输入寄存器 (3x)	数据起始地址:0x00000001(16进制)	1(10进制)
数据名称:对象子索引	寄存器区域:输入寄存器 (3x)	数据起始地址:0x00000002(16进制)	2(10进制)
数据名称:SDO中止代码	寄存器区域:输入寄存器 (3x)	数据起始地址:0x00000003(16进制)	3(10进制)
1# 远程IO1_1# TPDO数据输入8字节(位)			
数据名称:输入数据1_Bit0	寄存器区域:离散量输入 (1x)	数据起始地址:0x00000000(16进制)	0(10进制)
数据名称:输入数据1_Bit1	寄存器区域:离散量输入 (1x)	数据起始地址:0x00000001(16进制)	1(10进制)
数据名称:输入数据1_Bit2	寄存器区域:离散量输入 (1x)	数据起始地址:0x00000002(16进制)	2(10进制)
.....			
数据名称:输入数据8_Bit5	寄存器区域:离散量输入 (1x)	数据起始地址:0x0000003D(16进制)	61(10进制)
数据名称:输入数据8_Bit6	寄存器区域:离散量输入 (1x)	数据起始地址:0x0000003E(16进制)	62(10进制)
数据名称:输入数据8_Bit7	寄存器区域:离散量输入 (1x)	数据起始地址:0x0000003F(16进制)	63(10进制)
1# 远程IO1_2# TPDO数据输入8字节(字)			
数据名称:输入数据1	寄存器区域:输入寄存器 (3x)	数据起始地址:0x00000005(16进制)	5(10进制)
数据名称:输入数据2	寄存器区域:输入寄存器 (3x)	数据起始地址:0x00000006(16进制)	6(10进制)
数据名称:输入数据3	寄存器区域:输入寄存器 (3x)	数据起始地址:0x00000007(16进制)	7(10进制)
数据名称:输入数据4	寄存器区域:输入寄存器 (3x)	数据起始地址:0x00000008(16进制)	8(10进制)
1# 远程IO1_3# RPDO数据输出8字节(位)			
数据名称:输出数据1_Bit0	寄存器区域:线圈 (0x)	数据起始地址:0x00000000(16进制)	0(10进制)
数据名称:输出数据1_Bit1	寄存器区域:线圈 (0x)	数据起始地址:0x00000001(16进制)	1(10进制)
数据名称:输出数据1_Bit2	寄存器区域:线圈 (0x)	数据起始地址:0x00000002(16进制)	2(10进制)
.....			
数据名称:输出数据8_Bit5	寄存器区域:线圈 (0x)	数据起始地址:0x0000003D(16进制)	61(10进制)
数据名称:输出数据8_Bit6	寄存器区域:线圈 (0x)	数据起始地址:0x0000003E(16进制)	62(10进制)
数据名称:输出数据8_Bit7	寄存器区域:线圈 (0x)	数据起始地址:0x0000003F(16进制)	63(10进制)
1# 远程IO1_4# RPDO数据输出8字节(字)			
数据名称:输出数据1	寄存器区域:保持寄存器 (4x)	数据起始地址:0x00000000(16进制)	0(10进制)
数据名称:输出数据2	寄存器区域:保持寄存器 (4x)	数据起始地址:0x00000001(16进制)	1(10进制)
数据名称:输出数据3	寄存器区域:保持寄存器 (4x)	数据起始地址:0x00000002(16进制)	2(10进制)
数据名称:输出数据4	寄存器区域:保持寄存器 (4x)	数据起始地址:0x00000003(16进制)	3(10进制)

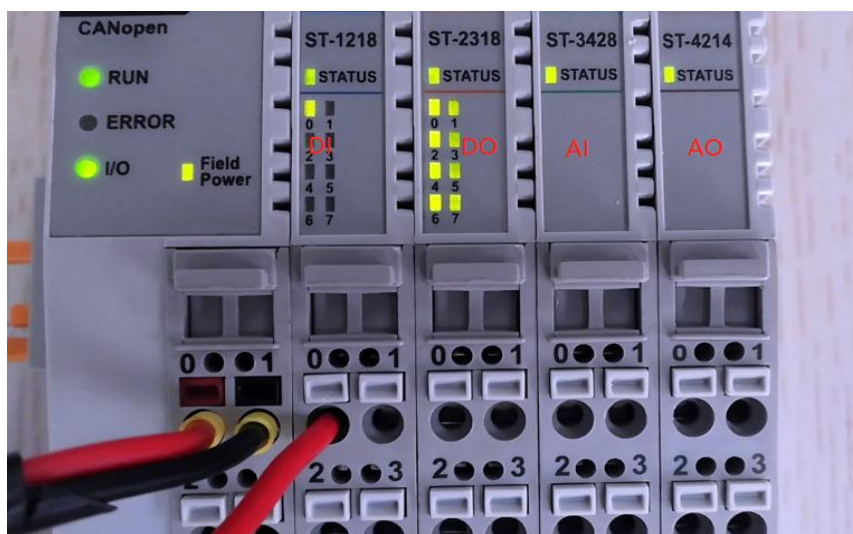
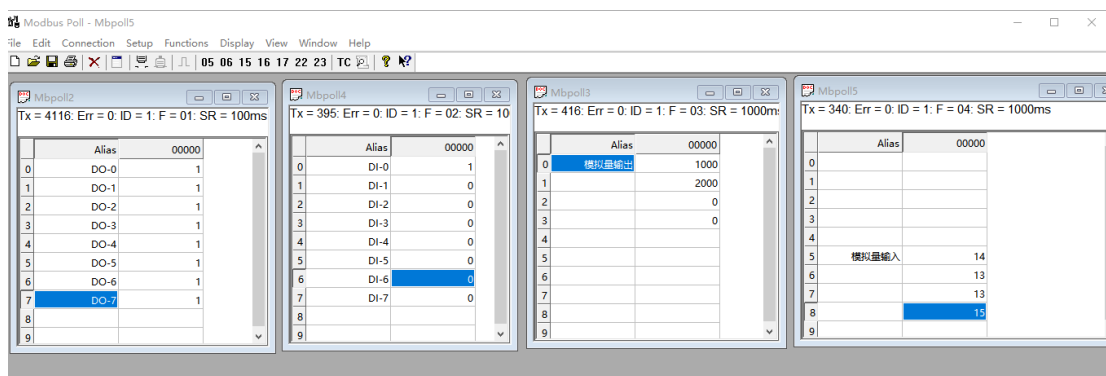
(8) 对象标识符修改完成后, 右键 MG-CANEX, 点击下载配置。



下载成功后选中 MG-CANEX 网关右键可以选择“在线”可以在线监控网关工作状态,“状态机”监视值为: 0x0005 表示通讯正常。



(9) 用 Modbu poll 软件模拟上位机，对相应的 Modbus 地址。



7.2 网关与鸣志伺服驱动器通讯示例

本例介绍将我司的协议转换器，把带 CANopen 协议的 MOONS 伺服驱动器和带 ModbusTCP 协议的上位机进行通讯。

本例需要的硬件：CANopen 转 ModbusTCP 协议转换器一台，MOONS 伺服驱动器一套，24V 开关电源。

本例需要的软件：上位机软件 Modbus poll，协议转换器调试软件 IO Config，

EDS 文件阅读软件：edseditor20de，远程 IO 的 EDS 文件。

硬件接线

将本网网关口和电脑网口连接，并确保电脑在 192.168.1 网段。网关接上 24V 电源，伺服驱动器电源和伺服电机接线 OK，将网关的 CAN_L 与伺服驱动器的 CAN_L 连接 CAN_H 与 CAN_H GND 与 CAN_GND 连接。确认无误后上电。

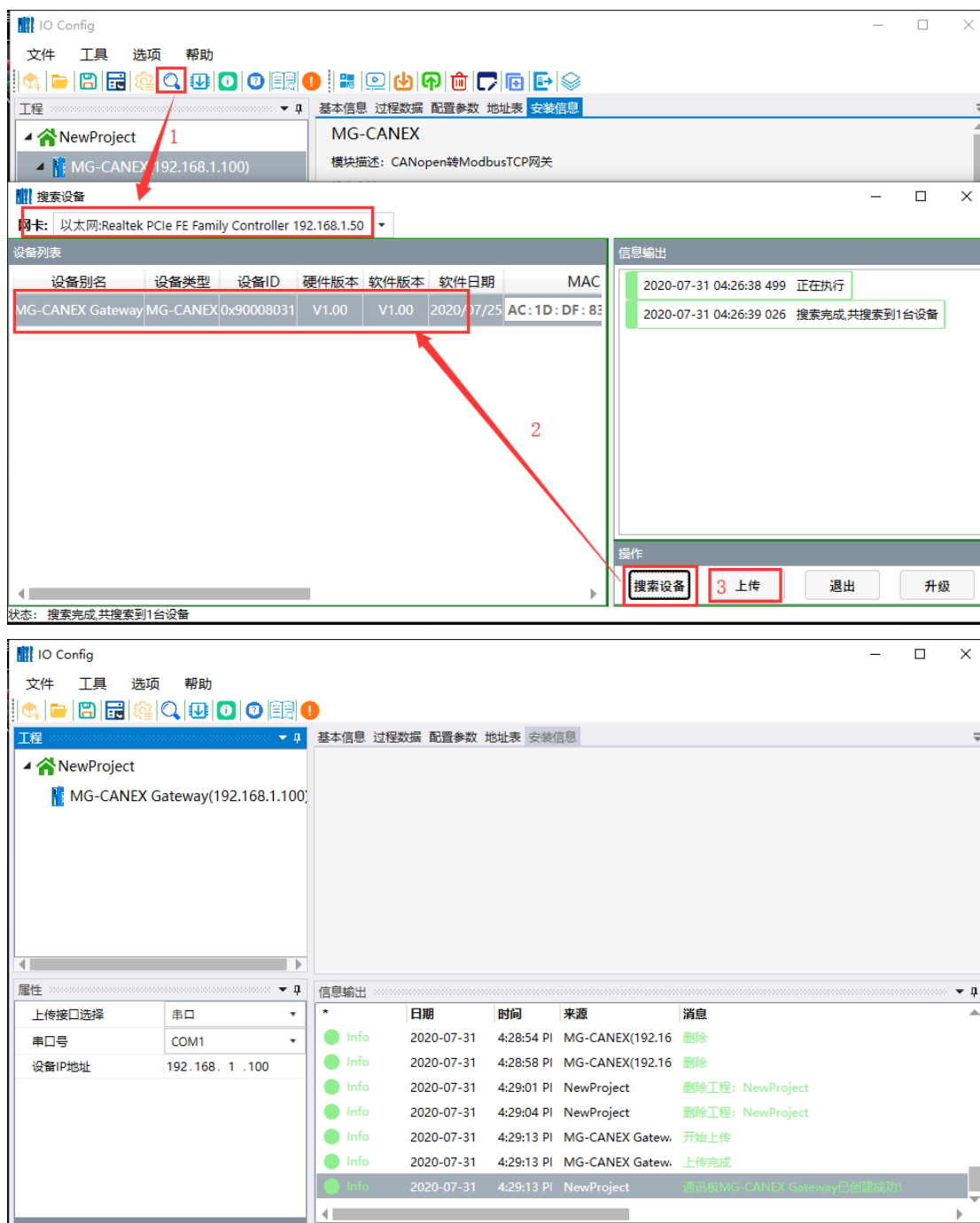
伺服驱动器地址查询

安装并打开软件：edseditor20de，用软件打开伺服驱动器的 EDS 文件，查询“对象字典”得到相关地址如下：（对象标识符）

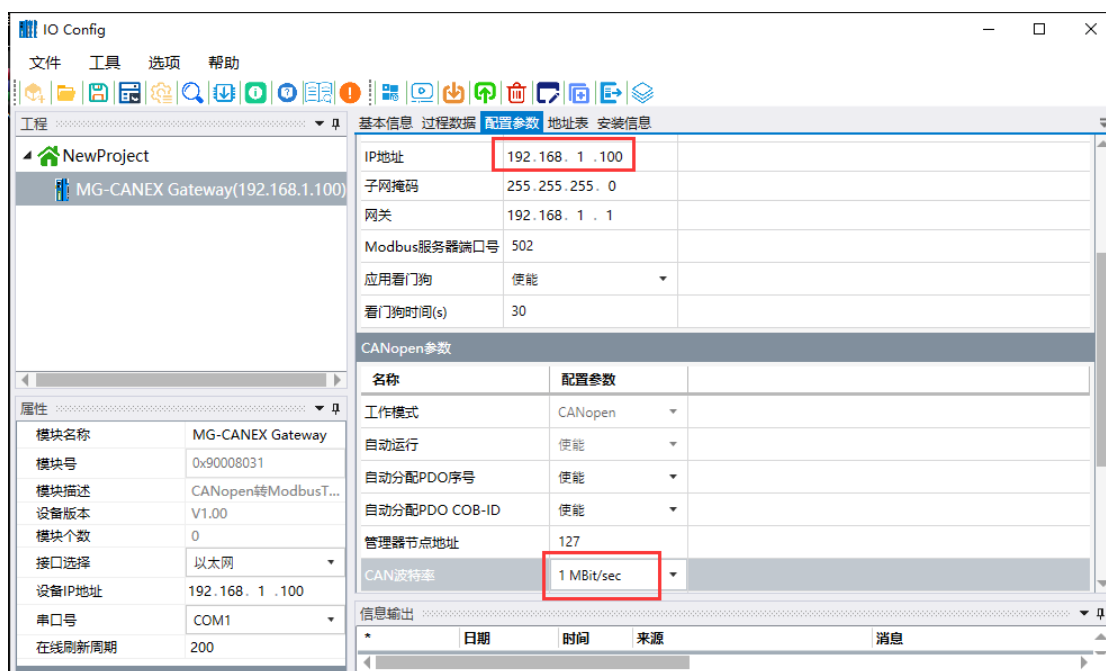
对象索引	对象子索引	位长度	属性	名称	数据类型
0x6040	0x00	0x10	WO	控制字	Unsigned16
0x6060	0x00	0x08	WO	控制模式	Integer8
0x60FF	0x00	0x20	RWW	速度	Integer32
0x6083	0x00	0x20	RWW	加速度	Integer32
0x6084	0x00	0x20	RWW	减速度	Integer32
0x607A	0x00	0x20	RWW	定位位置	Integer32
0x6081	0x00	0x20	RWW	定位速度	Integer32
0x6041	0x00	0x10	RO	状态字	Unsigned16
0x6061	0x00	0x08	RO	控制模式	Integer8
0x606C	0x00	0x20	RO	当前速度	Integer32
0x6064	0x00	0x20	RO	当前位置	Integer32

网关配置

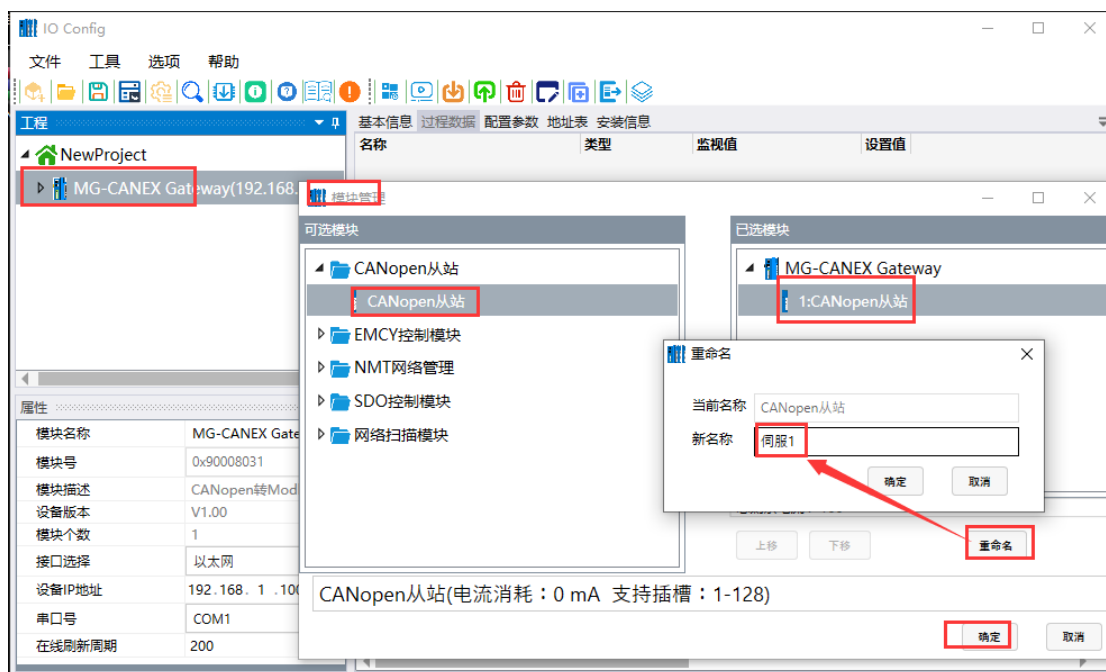
1. 打开《IO Config》配置软件。点击搜索 ，在弹出的界面选择本机网卡，点击搜索设备，会在设备列表扫描出网关设备，点击上传。



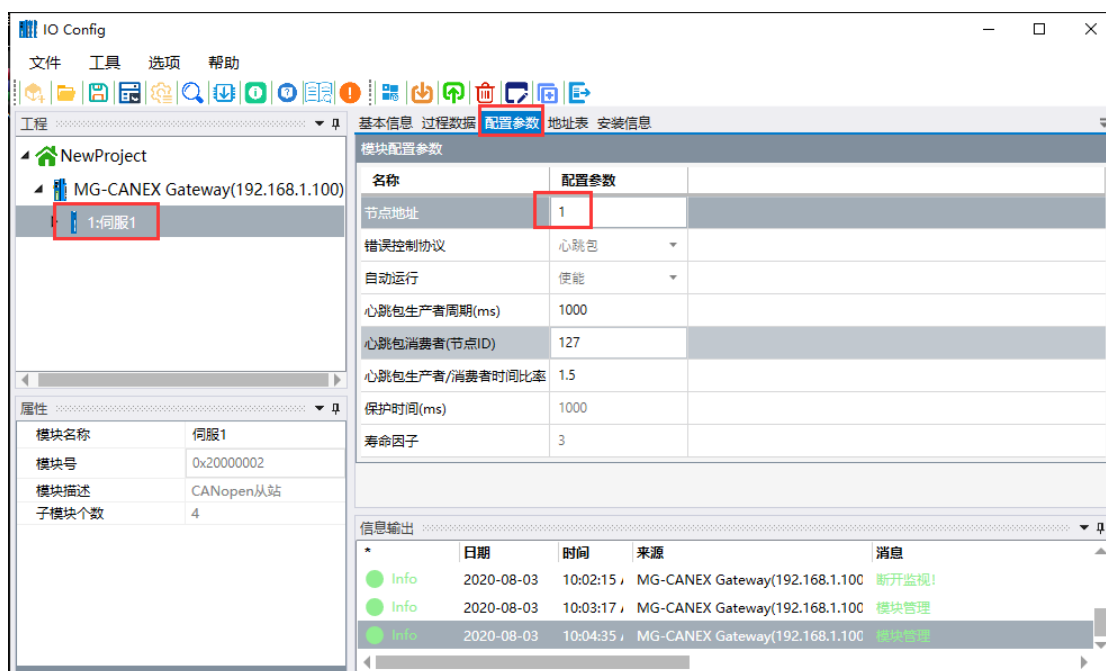
2.选中添加的网关在右边窗口的工具栏选择【配置参数】对网关进行参数设置。IP 地址：192.168.1.100，CANopen 波特率选 1M bit/sec。



3.选中添加的网关，右键选择【模块管理】双击或点击  添加 CANopen 从站。并命名为“伺服 1”



选中添加的从站“伺服 1”,在右边窗口点击【配置参数】，将从站节点地址设置与伺服驱动器节点拨码开关一致，本例为：1 其余默认。



5.右键“伺服 1”，点击【子模块管理】添加 CANopen 从站 PDO 指令：

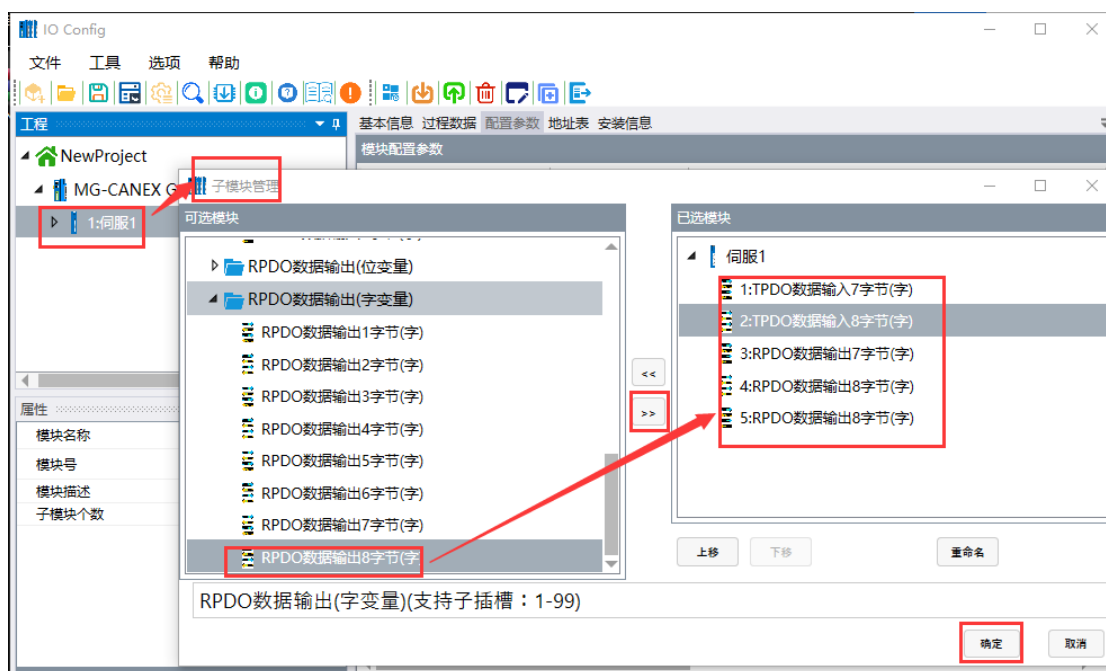
TPDO 数据输入 7 字节（字）：状态字 0x60410010、当前速度 0x606C0020、控制模式 0x60610008；

TPDO 数据输入 8 字节（字）：当前位置 0x60640020

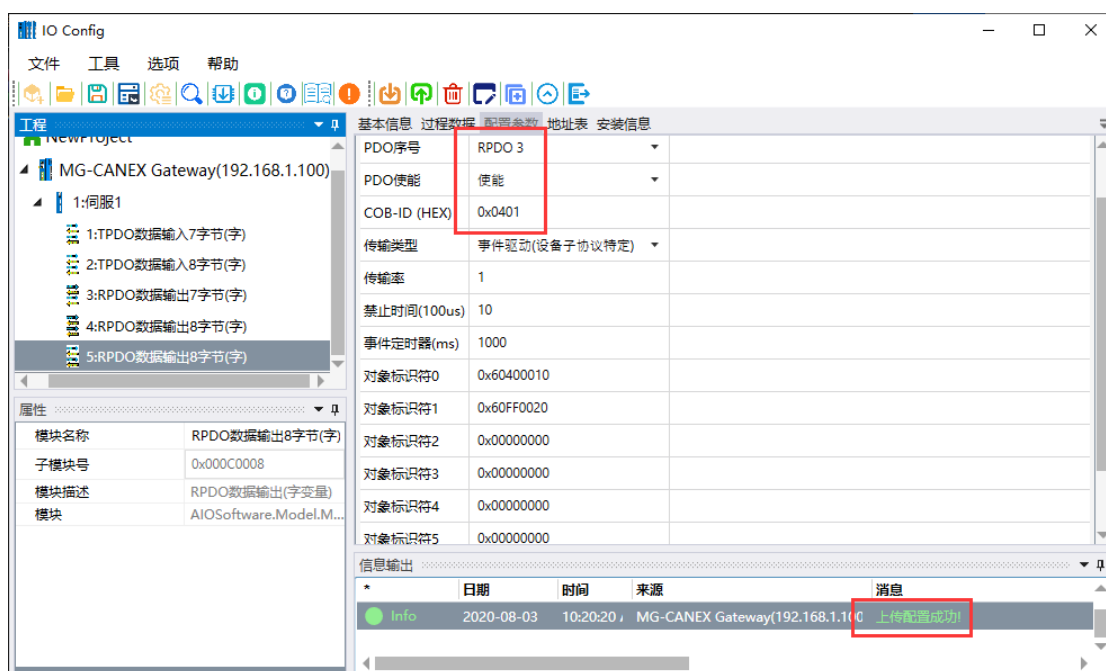
RPDO 数据输出 7 字节（字）：控制字 0x60400010、速度 0x60FF0020、控制模式 0x60600008

RPDO 数据输出 8 字节（字）：加速度 0x60830020、减速度 0x60840020

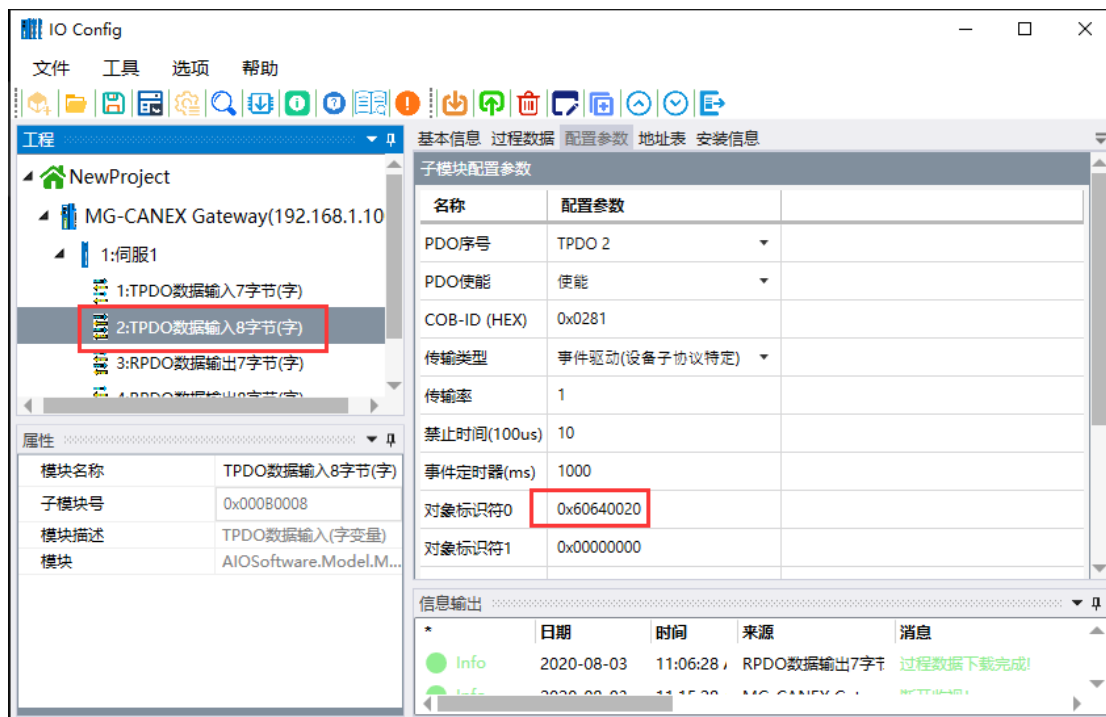
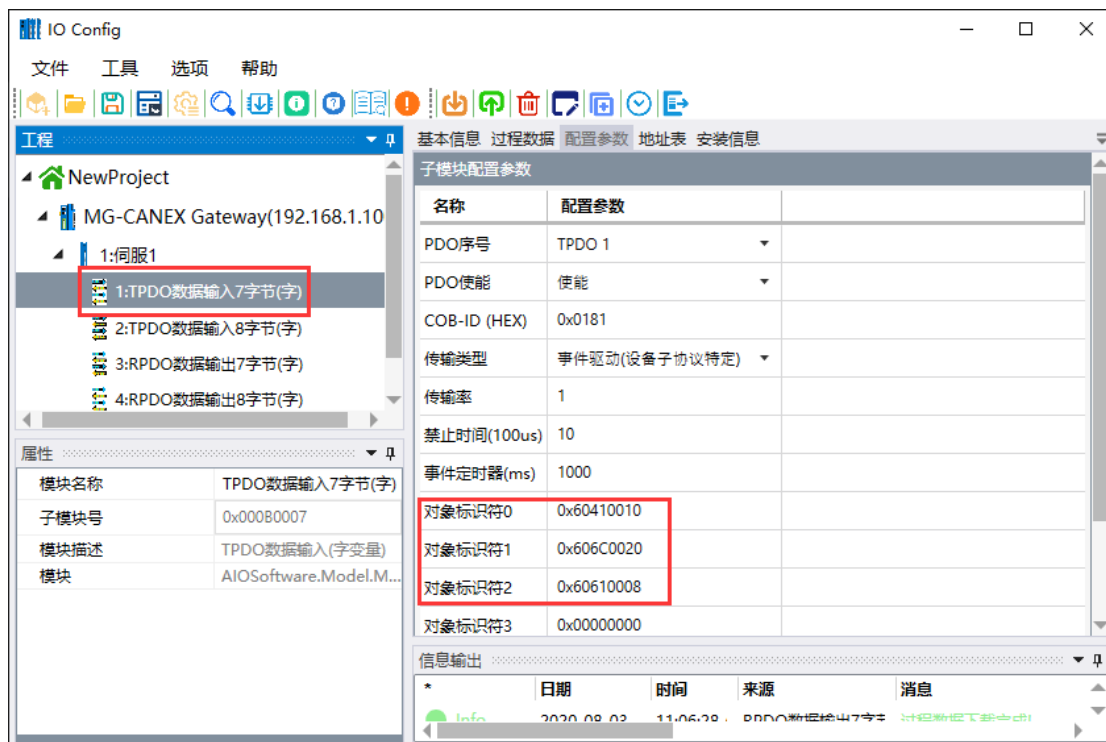
RPDO 数据输出 8 字节（字）：定位位置 0x607A0020、定位速度 0x60810020

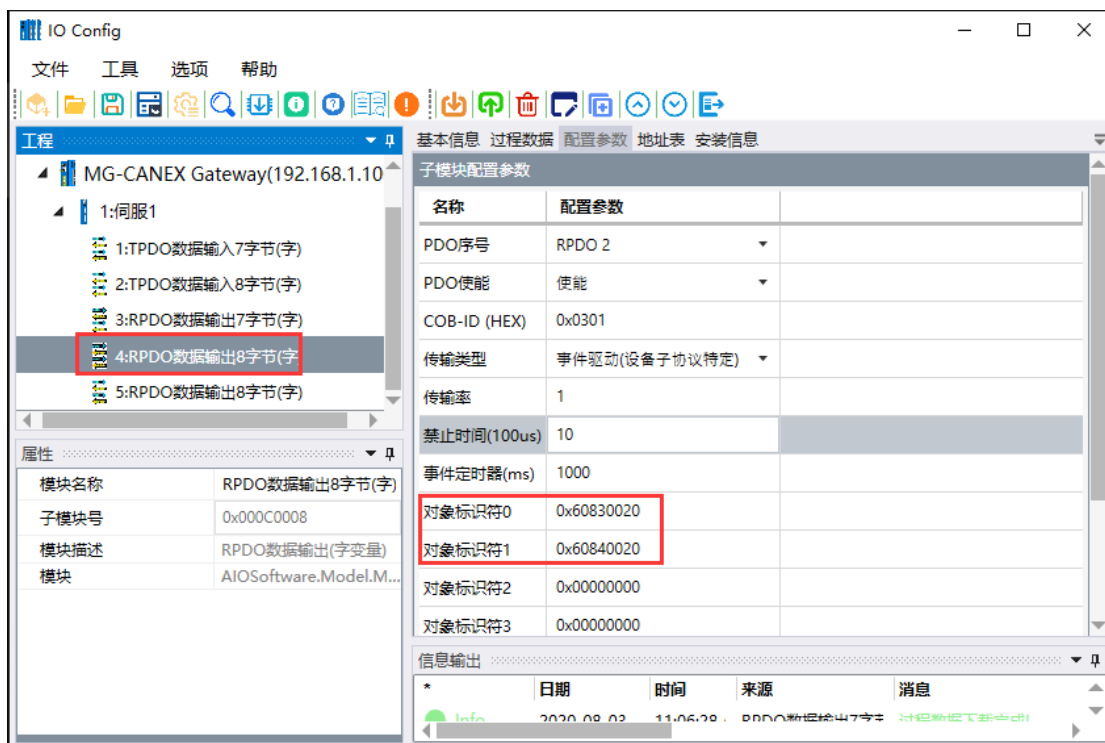
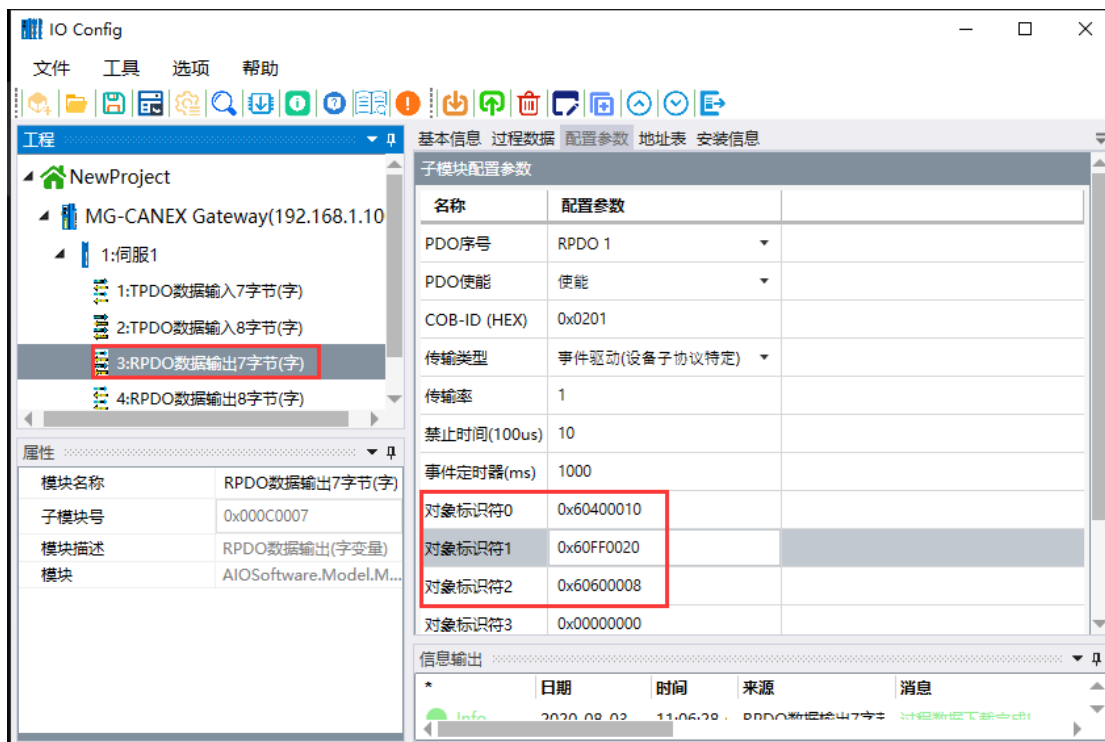


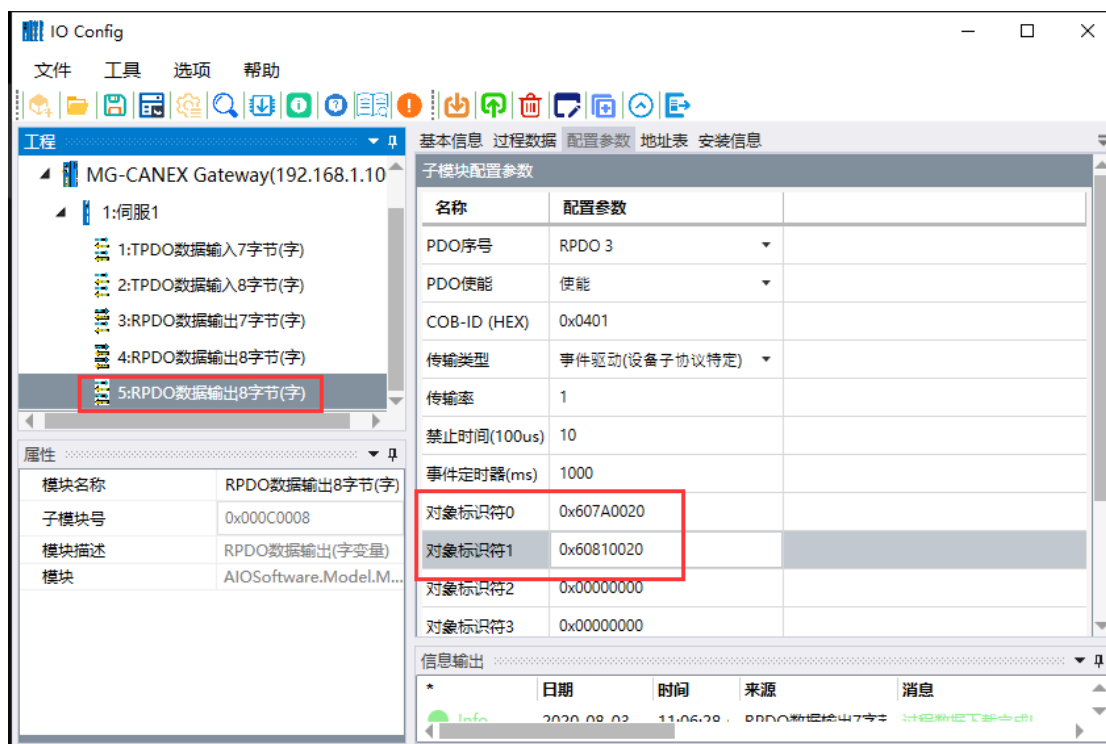
6.右键 MG-CANEX，下载配置，下载成功后，右键 MG-CANEX，上传配置，可以实现从站 PDO 指令自动配置 PDO 号和 COB-ID 号。也可以上传上来伺服自带的映射配置，可以修改默认映射配置。



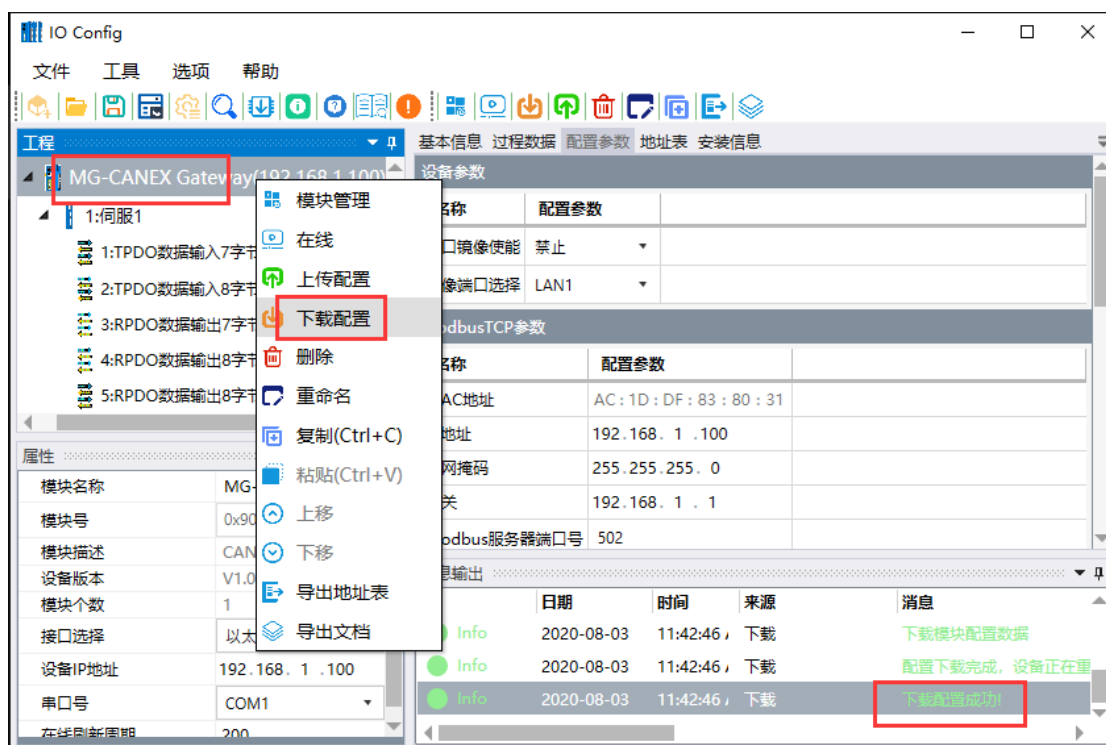
7.根据实际需求修改 PDO 里对象标识符参数。

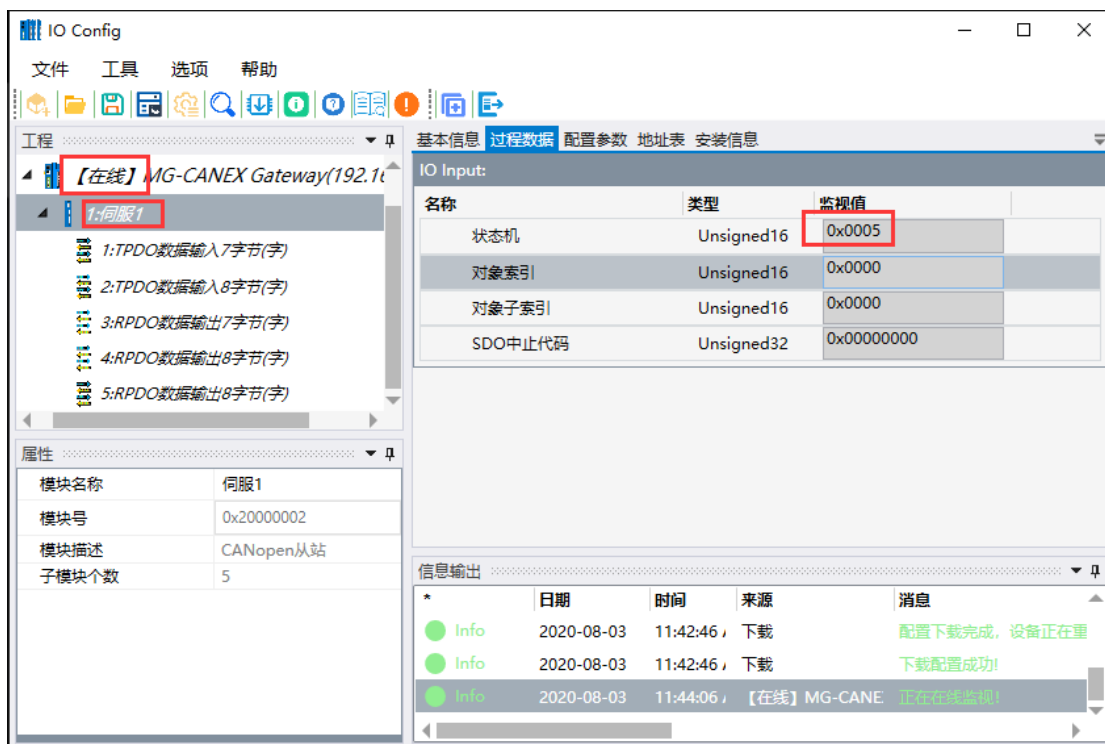




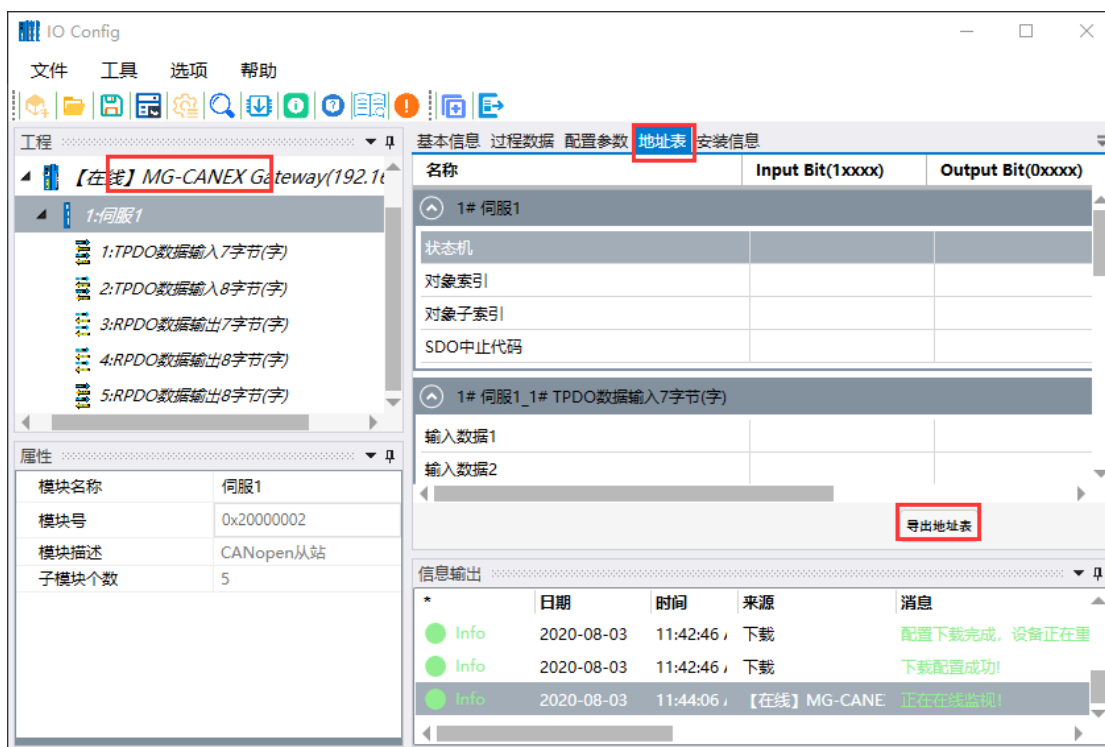


8. 下载网关配置：选中网关 MG-CANEX 然后右键，选择【下载配置】。然后选中 MG-CANEX 网关右键可以选择“在线”可以在线监控网关工作状态，“状态机”监视值为：0x0005 表示通讯正常。





9.选中网关，点击地址表，点击导出地址表。



1_伺服1_地址表_2020-08-03_114640.txt - 记事本				
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)				
1# 伺服1				
数据名称:状态机	寄存器区域:输入寄存器 (3x)	数据起始地址:0x00000000(16进制)	0(10进制)	
数据名称:对象索引	寄存器区域:输入寄存器 (3x)	数据起始地址:0x00000001(16进制)	1(10进制)	
数据名称:对象子索引	寄存器区域:输入寄存器 (3x)	数据起始地址:0x00000002(16进制)	2(10进制)	
数据名称:SDO中止代码	寄存器区域:输入寄存器 (3x)	数据起始地址:0x00000003(16进制)	3(10进制)	
1# 伺服1_1# TPDO数据输入7字节(字)				
数据名称:输入数据1	寄存器区域:输入寄存器 (3x)	数据起始地址:0x00000005(16进制)	5(10进制)	状态字0x60410010
数据名称:输入数据2	寄存器区域:输入寄存器 (3x)	数据起始地址:0x00000006(16进制)	6(10进制)	当前速度0x606C0020
数据名称:输入数据3	寄存器区域:输入寄存器 (3x)	数据起始地址:0x00000007(16进制)	7(10进制)	
数据名称:输入数据4	寄存器区域:输入寄存器 (3x)	数据起始地址:0x00000008(16进制)	8(10进制)	控制模式0x60610008
1# 伺服1_2# TPDO数据输入8字节(字)				
数据名称:输入数据1	寄存器区域:输入寄存器 (3x)	数据起始地址:0x00000009(16进制)	9(10进制)	当前位置0x60640020
数据名称:输入数据2	寄存器区域:输入寄存器 (3x)	数据起始地址:0x0000000A(16进制)	10(10进制)	
数据名称:输入数据3	寄存器区域:输入寄存器 (3x)	数据起始地址:0x0000000B(16进制)	11(10进制)	
数据名称:输入数据4	寄存器区域:输入寄存器 (3x)	数据起始地址:0x0000000C(16进制)	12(10进制)	
1# 伺服1_3# RPDO数据输出7字节(字)				
数据名称:输出数据1	寄存器区域:保持寄存器 (4x)	数据起始地址:0x00000000(16进制)	0(10进制)	控制字0x60400010
数据名称:输出数据2	寄存器区域:保持寄存器 (4x)	数据起始地址:0x00000001(16进制)	1(10进制)	速度0x60FF0020
数据名称:输出数据3	寄存器区域:保持寄存器 (4x)	数据起始地址:0x00000002(16进制)	2(10进制)	
数据名称:输出数据4	寄存器区域:保持寄存器 (4x)	数据起始地址:0x00000003(16进制)	3(10进制)	控制模式0x60600008
1# 伺服1_4# RPDO数据输出8字节(字)				
数据名称:输出数据1	寄存器区域:保持寄存器 (4x)	数据起始地址:0x00000004(16进制)	4(10进制)	加速度0x60830020
数据名称:输出数据2	寄存器区域:保持寄存器 (4x)	数据起始地址:0x00000005(16进制)	5(10进制)	
数据名称:输出数据3	寄存器区域:保持寄存器 (4x)	数据起始地址:0x00000006(16进制)	6(10进制)	减速度0x60840020
数据名称:输出数据4	寄存器区域:保持寄存器 (4x)	数据起始地址:0x00000007(16进制)	7(10进制)	
1# 伺服1_5# RPDO数据输出8字节(字)				
数据名称:输出数据1	寄存器区域:保持寄存器 (4x)	数据起始地址:0x00000008(16进制)	8(10进制)	定位位置0x607A0020
数据名称:输出数据2	寄存器区域:保持寄存器 (4x)	数据起始地址:0x00000009(16进制)	9(10进制)	
数据名称:输出数据3	寄存器区域:保持寄存器 (4x)	数据起始地址:0x0000000A(16进制)	10(10进制)	定位速度0x60810020
数据名称:输出数据4	寄存器区域:保持寄存器 (4x)	数据起始地址:0x0000000B(16进制)	11(10进制)	

10.用 Modbus Poll 软件模拟 Modbus TCP 客户端上位机，对应的 Modbus 地址。

速度模式（控制模式为 3）控制伺服驱动。

控制字 40001：先写 6，然后写入 7，然后写入 F 此时电机按上面设置速度，模式运行。

The screenshot shows the Modbus Poll software interface with two windows, Mbpoll1 and Mbpoll2. Mbpoll1 displays a table of data points with the following values:

Index	Alias	Value
0	6040控制字	15
1	60FF速度	0
2		200
3	6060控制模式	3
4	60836084加速度	0
5		300
6	6084减速度	0
7		300
8	607A定位位置	0
9		0
10	6081定位速度	0
11		0
12		

Mbpoll2 displays a table of data points with the following values:

Index	Alias	Value
0		
1		
2		
3		
4		
5	6041 状态字	(??) 0x0437
6	606C当前速度	0
7		199
8	6061控制模式	3
9	6064当前位置	2
10		-19693
11		
12		

Red annotations include a box around the value 3 in Mbpoll1 index 3, a box around the value 0 in Mbpoll2 index 6, and a box around the value 3 in Mbpoll2 index 8. Red arrows point from the box in Mbpoll1 index 3 to the box in Mbpoll2 index 8, and from the box in Mbpoll2 index 6 to the box in Mbpoll2 index 8.

位置模式（控制模式为3）控制伺服驱动。

控制字 40001：依次写 6—7—F—5F，此时电机按上面设置位置模式运行。

The screenshot shows the Modbus Poll software interface with two windows, Mbpoll1 and Mbpoll2. Mbpoll1 displays a table of data points with the following values:

Index	Alias	Value
0	6040控制字	(?) 0x005F
1	60FF速度	0
2		0
3	6060控制模式	1
4	60836084加速度	0
5		300
6	6084减速度	0
7		300
8	607A定位位置	1
9		0
10	6081定位速度	0
11		250
12		

Mbpoll2 displays a table of data points with the following values:

Index	Alias	Value
0		
1		
2		
3		
4		
5	6041 状态字	(??) 0x1437
6	606C当前速度	0
7		0
8	6061控制模式	1
9	6064当前位置	67
10		-297
11		
12		

Red annotations include a box around the value (?) 0x005F in Mbpoll1 index 0, a box around the value 1 in Mbpoll1 index 3, and a box around the value 1 in Mbpoll2 index 8.

附录：CANopen 协议简介

1 CANopen 概述

CANopen 是一种架构在控制局域网络（control Area Network, CAN）上的高层通讯协定，CANopen 协议簇包括通讯子协议及设备子协议，是工业控制常用的一种现场总线，CANopen 的高实时性使其在伺服系统中得到广泛运用。

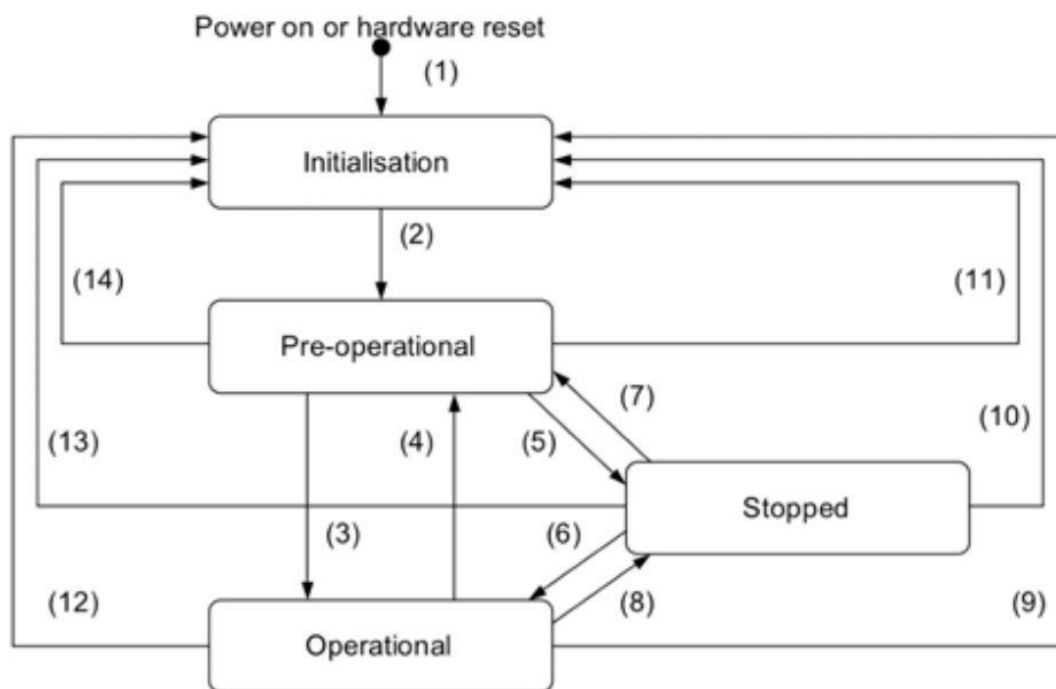
CANopen 网络中的数据包用标识符区分，CANopen 预定义的主从连接集对应的 COB-ID 标识符范围如下：

预定义主/从连接集

通信对象	COB-ID 范围	相应的对象字典
网络管理	000h	
同步报文	080h	1005h, 1006h, 1007h, 1008h
时间戳报文	100h	1012h, 1013h
Emergency（紧急事件）	081h~0FFh	1014h, 1015h
默认 SSDO（tx）	581h~5FFh	1200h
默认 SSDO（rx）	601h~67Fh	1200h
网络管理错误控制	701h~77Fh	1016h,1017h

2 NMT 网络管理

CANopen 网络中数据采用多种传输模式。NMT 网络管理采用 主机-从机 模式，在一个 CANopen 网络中，有一个 NMT 主机，多个 NMT 从机。NMT 主机可通过 NMT 命令控制从机设备的启停状态。NMT 状态图如下：



NMT 状态机

NMT 状态的转变

状态改变路径	状态改变条件
1	上电后自动初始化设备
2	完成初始化之后自动改变
3、6	NMT 主机的启动远程节点指令
4、7	NMT 主机进入预操作状态指令
5、8	NMT 主机进入停止状态指令
9、10、11	NMT 主机复位远程节点指令
12、13、14	NMT 主机复位远程节点通信参数指令

NMT 消息格式如下：

COB-ID	Byte0	Byte1
0x000	CS (命令字)	Node-ID

NMT 命令 COB-ID 固定为 0x000，Node-ID 为需要控制的节点地址。0 为广播地址。

CS 命令字说明：

01h=start_remote_node

81h=reset_node

02h=stop_remote_node

82h=reset_communication

80h=enter_pre-operational

3 服务数据对象 SDO

SDO 通信采用 **客户机-服务器** 模式，SDO 用来访问一个设备的对象字典。访问者被称作客户机 (client)，对象字典被访问且提供所请求服务的 CANopen 设备别称作服务器 (server)。客户机的 CAN 报文和服务器的应答 CAN 报文总是包含 8 字节数据 (尽管不是所有的数据字节都一定有意义)。一个客户的请求一定有来自服务器的应答。

SDO 有 2 种传送机制：

加速传送 (Expedited transfer)：最多传输 4 字节数据

分段传送 (Segmented transfer)：传输数据长度大于 4 字节

4 SDO 传输协议

SDO 中实现了 5 个请求/应答协议：启动域下载、域分段下载、启动域上传、域分段上传和域传送中止。

这些协议的 SDO 命令字 (SDO CAN 报文的第一个字节) 语法和细节在下面部分说明：（‘—’ 表示不相关，应为 0）

（1）启动域下载协议

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

说明：

◆ 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。

(2) 启动域上传协议

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

说明: n, e, s: 与启动域下载相同。

(3) 域分段下载协议

域分段下载								
Bit	7	6	5	4	3	2	1	0
Client→	0	0	0	t	n			c
←Server	0	0	1	t	-	-	-	-

说明:

◆ n : 无意义的数据字节数。如果没有指明段长度, 则为 0。

◆ c : 0 = 有后续分段需要 download, 1 = 最后一个段。

◆ t : 触发位, 后续每个分段交替清零和置位 (第一次传送为 0, 等效于 request/response)。

(4) 域分段上传协议

域分段下载								
Bit	7	6	5	4	3	2	1	0
Client→	0	1	1	t	-	-	-	-
←Server	0	0	0	t	n			c

说明: n, c, t : 与域分段下载相同。

(5) SDO 客户或服务器通过发出如下格式的报文来中止 SDO 传送:

域分段下载								
Bit	7	6	5	4	3	2	1	0
C→/←S	1	0	0	-	-	-	-	-

在域传送中止报文中, 数据字节 1 和 2 表示对象索引, 字节 3 表示子索引, 字节 4 到 7 包含 32 位中止码, 描述中止报文传送原因, 见表所示。

16 进制中止代码表 (字节 4 到 7)

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）

5 紧急对象 Emergency Object

紧急报文由设备内部出现的致命错误触发，由相关应用设备已最高优先级发送到其它设备。适用于中断类型的错误报警信号。

一个紧急报文由 8 字节组成，格式如下：

sender → receiver (s)

COB-ID	Byte0-1	Byte2	Byte3-7
0x080+Node_ID	错误代码	错误寄存器 (对象 0x1001)	制造商特定的错误区域

16 进制的应急错误代码如下表所示。应急错误代码中 ‘xx’ 部分由相应的设备子协议定义。

应急错误代码（16 进制）

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 tempearture
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 lenvgt h error
8220	Length exceedd
90xx	External error
F0xx	Additional functions
FFxx	Device specific

错误寄存器 (Error Register) 在设备的对象字典 (索引 0x1001) 中, 表 3-6 说明了错误寄存器的位定义。设备可以将内部错误映射到这个状态字节中, 可以快速查看当前错误。

8 位错误寄存器位定义

Bit	错误类型
0	Generic
1	Current
2	Voltage
3	Temperature
4	Communication
5	Device profile specific
6	Reserved (=0)
7	Manufacturer specific

制造商特定错误区域可能包含与设备相关的其它的错误信息。

6 过程数据对象 PDO

PDO 对象用来传输实时数据, PDO 对象采用 生产者-消费者 模式。数据从一

个生产者传到多个消费者。数据传送限制在 1-8 个字节（例如：一个 PDO 可以传输最多 64 个数字 I/O 值，或者 4 个 16 位的 AD 值）。PDO 通讯没有额外的协议规定。PDO 有两种类型的使用：即数据发送和数据接收。他们以 TPDO 和 RPDO 区分；

RPDO 通信参数索引=1400h+RPDO 编号-1

TPDO 通信参数索引=1800h+TPDO 编号-1

RPDO 映射参数索引=1600h+RPDO 编号-1

TPDO 映射参数索引=1A00+TPDO 编号-1

PDO 传输模式：

同步传输：同步传输（通过接收同步对象实现同步），同步传输又可分为非周期和周期传输。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：异步传输（由特定事件触发），其触发方式可有两种方式，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，数据变化传输等）。

PDO 通信参数子 02h 为 PDO 的传输类型，其定义了触发 TPDO 传输或处理收到的 RPDO 索引的方法，如表所列：

类型	同步		异步	仅 RTR
	循环	非循环		
0		X		
1~240	X			
241~251	保留			
252	X			X
253			X	X
254、255			X	

四川零点自动化系统有限公司

地址：四川省绵阳市飞云大道 261 号综合保税区 204 厂房

电话：0816-2530577

传真：0816-6337503

邮编：621000

网址：www.odot.cn



零点微信公众号