



RTU-BOX800

实时数字控制器使用指南



用户手册ver20250627

让控制简单高效!

南京瑞途优特信息科技有限公司
Nanjing Rtunit Information Technology Co., LTD

ABOUT RTUNIT

瑞途优特公司简介

南京瑞途优特信息科技有限公司（rtunit®）成立于2016年，是一家专注于电力电子与电机驱动领域的创新型科技企业，致力于成为全球领先的开源设备及解决方案供应商。作为国家认证的高新技术企业，公司始终坚持“自主研发、自主品牌、国产化”的发展理念，以创新驱动为核心，为客户提供高效、可靠的技术支持与产品服务。

公司深耕科研与教育领域，主要服务于高校、科研院所等高端客户群体，凭借深厚的技术积累与丰富的行业经验，已成功开发多款创新型产品，包括快速原型控制器、积木式电力电子功率模块，以及覆盖多种功率等级的驱动器和逆变器。这些产品以模块化设计、灵活扩展、高可靠性、高稳定性为特点，广泛应用于科研实验、技术开发及系统集成等领域。

瑞途优特高度重视技术研发与知识产权保护，目前已拥有多项发明专利，并持续推动技术成果的产业化落地。公司不仅提供标准化产品，更注重为客户量身定制系统化解决方案，助力其在电力电子与电机驱动领域实现高效研发与创新突破。

未来，南京瑞途优特将继续秉承“开放、协作、共赢”的理念，携手合作伙伴，推动国产电力电子技术的进步与行业生态的繁荣，为全球科研与工业发展贡献力量。





致用户

尊敬的用户：

您好！

首先感谢您对我公司的信任，并选用本公司产品。

为了使您尽快掌握本产品的正确使用方法，我们特别为您编写了《实时数字控制器RTU-BOX800使用指南》。我们对产品使用指南的编制力求全面而详尽，从中您可以获得有关本产品的正确操作方法等相关知识。我们强烈建议您在操作本产品前，务必先仔细阅读《实时数字控制器RTU-BOX800使用指南》，这样有助您更好地使用本产品。

《实时数字控制器RTU-BOX800使用指南》是引导操作者正确使用产品的指导性文件，为操作者提供正确操作及使用的信息说明，提供保护操作者、他人和产品安全的使用方法，请操作者严格按照说明书要求进行正确操作，避免因操作失误而产生的风险。同时，对操作者在产品操作过程中可能遇到的问题给予解答，并给出适当的风险警示。

为了维护您的权益，请遵守《实时数字控制器RTU-BOX800使用指南》的相关操作条款，如果您未按本说明书的要求使用本产品可能会造成设备故障及人身伤害安全事故，为了保证产品的使用安全请严格按照本说明书要求进行操作，未经我公司设计部门同意，请勿擅自对产品进行改装及违章违规作业，以免给您带来不必要的损失。

同时，您的需求是我们产品研发和提升的方向。您在使用我公司产品时有任何好的建议及意见，可通过相应渠道及时告诉我们。本公司将以至诚、快捷和有效的服务满足您的需求，为您带来最大的经济效益，助您的事业更进一步。



CONTENTS

目录

1.RTU-BOX800 简介	1
1.1 概述	1
1.2 模块拓扑	2
1.3 产品特点	2
1.3.1 信号	2
1.3.2 安全性	2
1.4 RTU-BOX800	3
1.4.1面板主视图	3
1.4.2产品尺寸与重量	4
2. RTU-BOX800硬件手册	5
2.1 硬件概述	5
2.2 电源	5
2.3 LCD	5
2.4 通信接口介绍	6
2.4.1 RS232	6
2.4.2 RS485	6
2.4.3 CAN	7
2.4.4 ETHERNET	7
2.5 ADC板卡	7
2.6 PWM板卡	9
2.7 ENC板卡	9
2.8 DAC板卡	10
3 .RTU-BOX800开发环境搭建	11
3.1 概述	11
3.2 安装Rtunit Studio 2023	11
3.3 安装RTU-BOX800 Toolbox	14
3.4 CCS运行环境配置	17
3.5 开发环境配置	18

4.RTUNIT STUDIO 2023使用教程	22
4.1 Rtunit Studio 2023	22
4.2 新建工程	22
4.3 创建Logic模型	23
4.4 编译&下载	28
4.5 调试	29
4.6 参数观测	29
4.7 波形观测	30
4.8 查看代码	31
4.9 导出	32
4.10 导入	32
4.11 修改RTU-BOX800 IP 地址	33
4.12 更新Rtunit Toolbox&更新框架	33
5.Rtunit Toolbox模型手册	35
5.1 Rtunit Toolbox	35
5.2 Common Library	35
5.2.1 Pulse Wave 模块	35
5.2.2 Sawtooth Wave 模块	37
5.2.3 Sine wave 模块	38
5.2.4 Tcp2Sim 模块	40
5.3 Power Electronics	41
5.3.1 SVPWM 模块1	41
5.3.2 SVPWM 模块2	42
5.4 RTU BOX	43
5.4.1 Absolute Rotary Encoder 模块	43
5.4.2 ADC 模块	44
5.4.3 DAC 模块	47
5.4.4 DI 模块	48
5.4.5 DO 模块	50
5.4.6 eCAN Rx 模块	51
5.4.7 eCAN Tx 模块	52
5.4.8 Ethernet Configuration 模块	53
5.4.9 Ethernet Receive 模块	54
5.4.10 Ethernet Send 模块	55
5.4.11 FPGA PWM 模块	56
5.4.12 Hall Encoder 模块	59

5.4.13 Incremental Encoder 模块	60
5.4.14 Main Loop Task 模块	61
5.4.15 TIMING 模块	63
5.4.16 Rotary Encoder 模块	64
5.4.17 SCI Receive 模块	66
5.4.18 SCI Transmit 模块	68
5.4.19 SYSCFG 模块	70
5.4.20 Variable 模块	71
5.4.21 WAVE模块	72
5.4.22 WAVE REC模块	73
5.5 Transformations	75
5.5.1 AVG模块	75
5.5.2 Clarke 模块1	76
5.5.3 Clarke 模块2	77
5.5.4 IClarke 模块	78
5.5.5 IPark 模块	79
5.5.6 Park 模块1	79
5.5.7 RMS 模块2	80
5.5.8 SPLL-SOGI 模块	81
5.5.9 SRF-PLL 模块	82
5.5.10 VPP2Vphase 模块	83
RTU-BOX800使用技巧&注意事项	84

1. RTU-BOX800 简介

摘要: 本章是 RTU-BOX800 快速原型系统 (RPS) 概述, 介绍了 RTU-BOX800 各个部分的基本功能, 定义并解释了本手册后文中出现的名词, 建议优先阅读本章。

1.1. 概述

RTU-BOX800 是一种基于模型设计、具有代码自动生成功能的实时数字控制器, 适用于汽车工业与智能制造、工业自动化、高品质电驱动系统、电力系统自动化与智能电网、电力电子、航空航天, 轨道交通等领域, 与现有的快速原型系统 (RPS) 相比, RTU-BOX800 使用简单, 灵活性强、成本更低。

RTU-BOX800 软件控制平台包括底层驱动软件包 RTU.Lib、集成开发环境 Rtunit Studio、Simulink®模型库 Rtunit Toolbox 和实时代码生成组件 RTU-Coder 组成, 如图 1.1 所示。



图 1.1 RTU-BOX800 控制平台

RTU-BOX800 底层驱动软件包 RTU.Lib 是将硬件功能的驱动函数进行了封装, 并经过精心优化设计使其符合实际工程应用需求的程序框架。用户使用时无需关注底层, 只需调用相应的模型或者功能函数, 增强了自动生成代码的可读性, 提高了代码执行效率。

集成开发环境 Rtunit Studio 负责工程的统一管理, 将 Simulink®中的模型文件以及生成的代码文件导入, 并结合框架程序转化成产品级的工程代码, 下载入硬件控制器中运行。通过自定义的图形化界面, 用户可以对程序中的任意变量进行在线观测和调整, 同时还可以观察变量的实时波形、保存并导出波形的数据。

模型库 Rtunit Toolbox 是系统集成于 Matlab®/Simulink®环境中的功能模块库, 是对 Simulink®工具箱的补充和扩展, 提供了系统中所有硬件资源的 Simulink®封装模块。用户能够直接将硬件功能集成到 Simulink®中, 设计硬件控制模型。此外, Rtunit 公司还提供电机驱动和电力电子控制中常用的算法模型与案例, 详情请访问 www.rtunit.com。

RTU-BOX800 控制平台包括一套经过 100%测试的软件库, 一个坚固的金属机箱, 以及各种丰富的使用案例。RTU-BOX800 无疑是科研/工业领域的可靠工具。

1.2. 模块拓扑

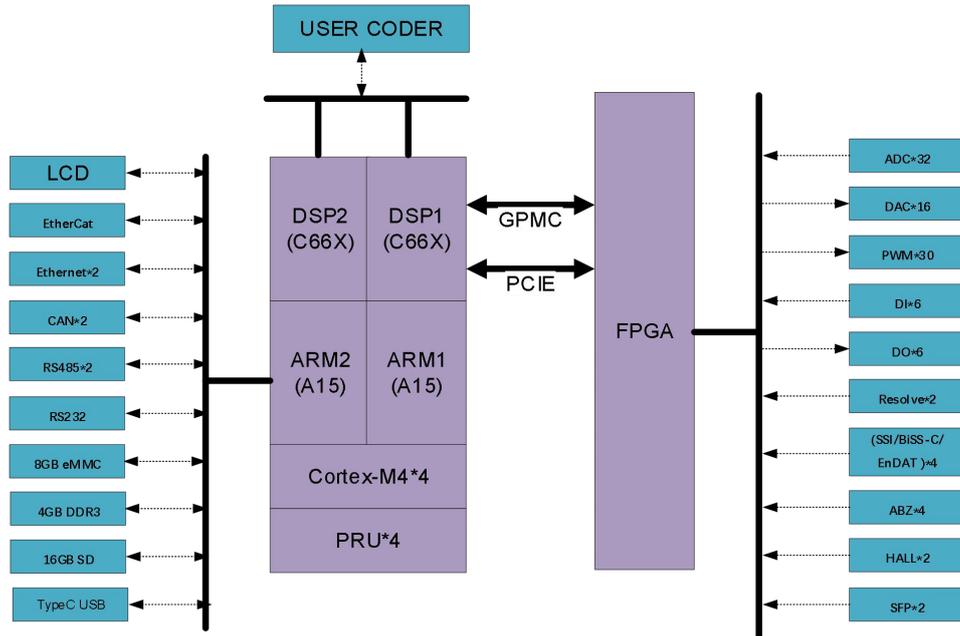


图 1.2 RTU-BOX800 拓扑结构

RTU-BOX800 可以使用的 CPU 核心有 2 个 750MHz 的 C66x DSP, 2 个 1500MHz 的 A15 ARM, 4 个 Cortex-M4, 4 个 PRU, 性能强劲。

1.3. 产品特点

1.3.1. 信号

在科研/工业领域, 为了信号的安全性和完整性, 需要增加电隔离器件。用户一般需要在控制部分和电源部分之间设立特定的接口。

RTU-BOX800 的接口适合几乎任何传感器的多功能模拟前端。除此之外, RTU-BOX800 自带的数字通信接口允许在各种工业和开发环境中集成此系统。

1.3.2. 安全性

RTU-BOX800 能够在程序错误、器件损坏等意外事件发生时主动停止并报警, 从而保障 RTU-BOX800 控制系统的安全性。

用户可以根据需要设定门限, 当检测值超过门限时, 所有的输出信号都会被阻止。这种保护机制的特点在于其完全独立于用户的程序/算法, 因此无论 DSP/FPGA 出现任何 bug, 门限一直存在, 如图 1.3 所示。每当检测值大于门限(高)或小于门限(低)时, RTU-BOX800 就会激活故障状态, 锁住所有的输出。

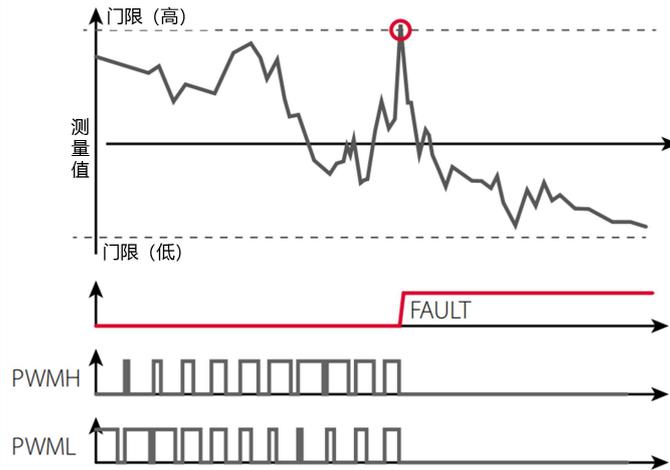


图 1.3 门限设置与保护措施

1.4. RTU-BOX800

1.4.1 面板主视图

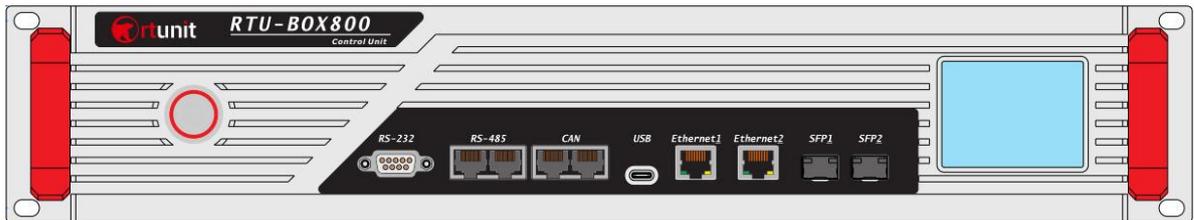


图 1.4 RTU-BOX800 正面主视图

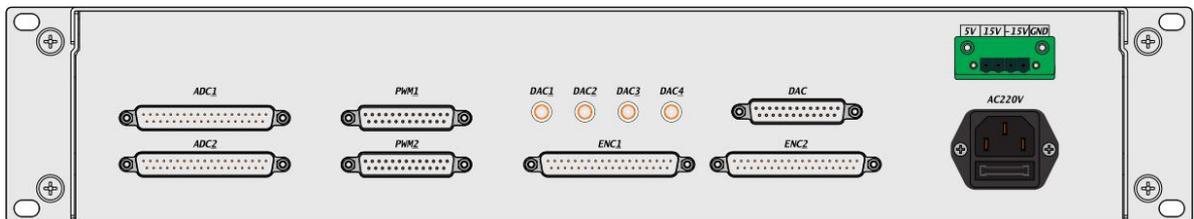


图 1.5 RTU-BOX800 背面主视图

1.4.2 产品尺寸与重量

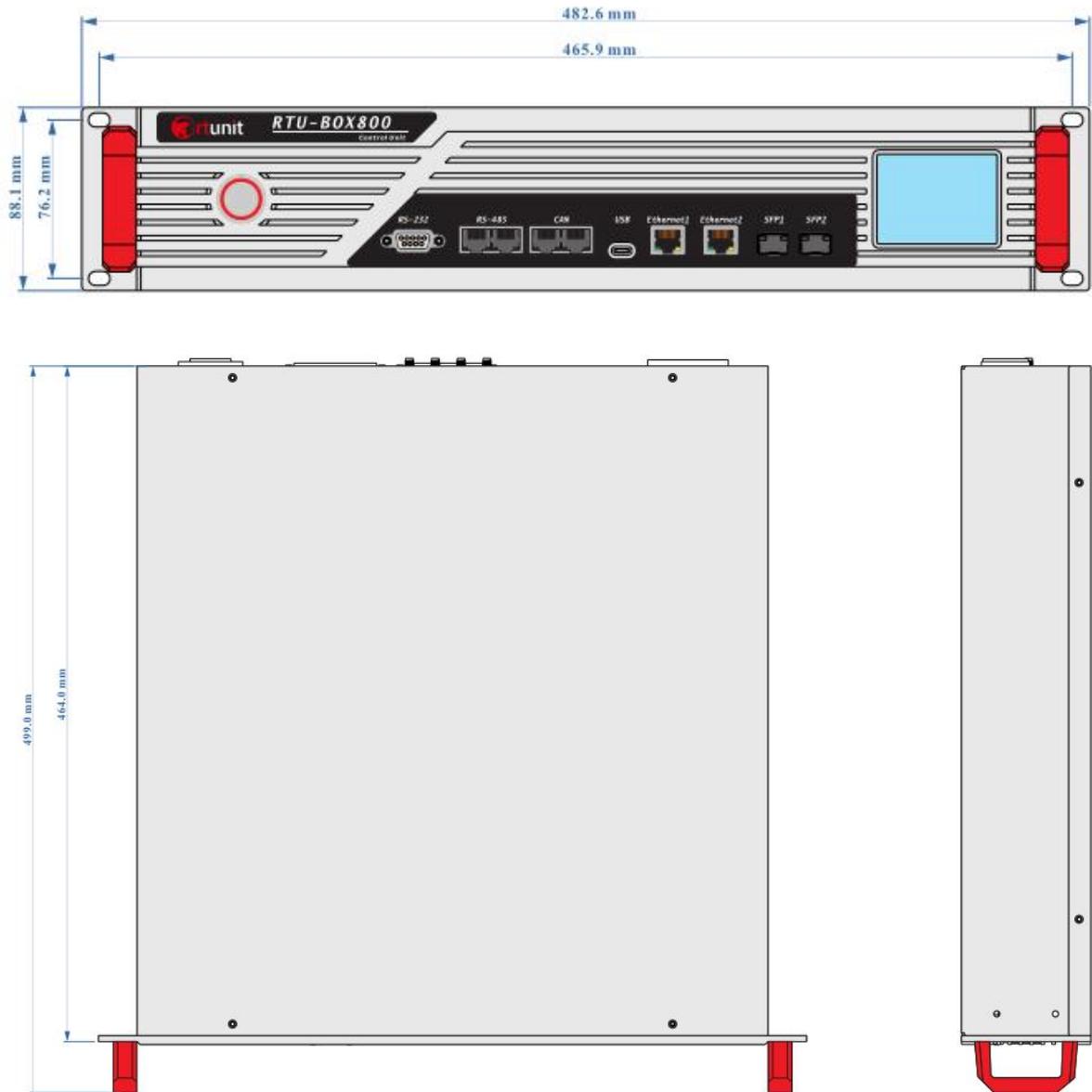


图 1.6 RTU-BOX800 尺寸图

RTU-BOX800 重量为 7.9KG（标准配置）。

2. RTU-BOX800 硬件手册

摘要: RTU-BOX800 的硬件手册，详细介绍了 RTU-BOX800 各个外设的功能与特性。

2.1. 硬件概述

RTU-BOX800 内部集成了 ADC、DAC、PWM、RS232、RS485、CAN、USB、Ethernet、SFP 以及 CAP、SSI、ABZ 编码器等外设。

2.2. 电源

RTU-BOX800 需在背面板接入 220VAC 供电，此外，RTU-BOX800 还可以向外部提供 5V，15V，-15V，各电源功率如表 2.1 所示。电源接口如图 2.1 所示。

表 2.1 各电源参数

类型	电压	最大功率
输入	AC220V	200W
输出	5V	50W
	15V	65W
	-15V	15W



图 2.1 电源接口

2.3. LCD

RTU-BOX800 前面板右侧配备有一个显示屏，用于显示设备运行的状态信息。

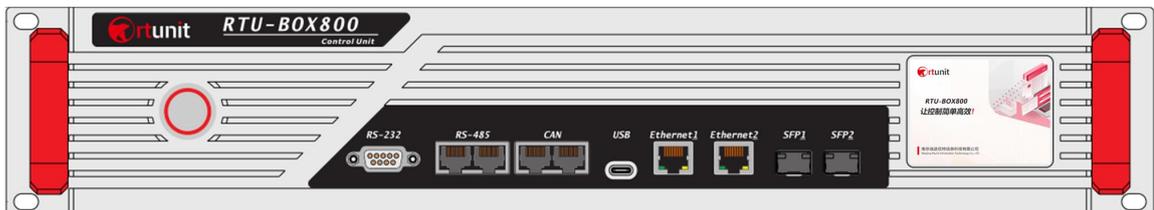


图 2.2 RTU-BOX800 前面板

2.4. 通信接口介绍

RTU-BOX800 前面板配备了 100M/1000M 以太网 ETHERNET, SFP, USB, 隔离的 RS232, 隔离的 RS485, 隔离的 CAN 等通信接口, 用于数据交互。

2.4.1 RS232

RS232 接口特性如表 2.2 所示。

表 2.2 RS232 接口参数

项目	参数	备注
接口数量	1	-
I/O 特性	磁隔离	-
电平标准	RS232C	-
通信速率	2400bps—20kbps	可以设置
隔离电压	2.5kV	-
端子	DB9	公头

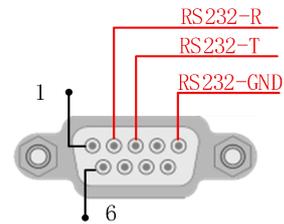


图 2.3 RS232 引脚定义

2.4.2 RS485

RS485 接口特性如表 2.3 所示。

表 2.3 RS485 接口参数

项目	参数	备注
接口数量	2	-
I/O 特性	磁隔离	-
电平标准	RS485	-
通信速率	100KbS—10Mbps	可以设置
隔离电压	2.5kV	-
终端电阻	120 Ω	可旁路
端子	RJ45	

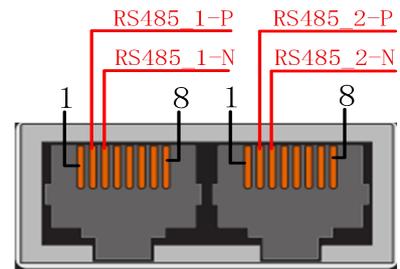


图 2.4 RS485 引脚定义

2.4.3 CAN

CAN 接口特性如表 2.4 所示。

表 2.4 CAN 接口参数

项目	参数	备注
接口数量	2	-
I/O 特性	磁隔离	-
电平标准	CAN2.0B	-
通信速率	125kbps—1Mbps	可以设置
邮箱数量	32	
隔离电压	2.5kV	-
终端电阻	120 Ω	可旁路
端子	RJ45	

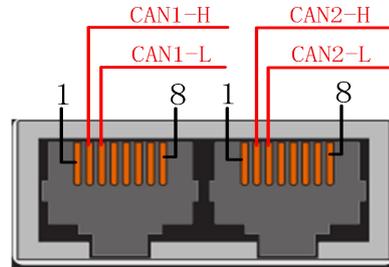


图 2.5 CAN 引脚定义

2.4.4 ETHERNET

ETHERNET 是以太网通信接口，其特性如表 2.5 所示，具备 TCP/IP 通信功能。可以通过 ETHERNET 实现与 Rtunit Studio 软件的数据通信。

表 2.5 ETHERNET 接口参数

项目	参数	备注
接口数量	2	-
I/O 特性	内置网络变压器滤波器	-
LAN 通信	1000Base-T	1000M
IP 地址	可软件设定	
端子	RJ45	-

2.5.ADC 接口介绍

RTU-BOX800 共有两个 DB37 接入的 ADC 接口——ADC1 和 ADC2。ADC 用于采集模拟量信号并转为数字量。引脚定义如图 2.6 所示，性能参数如表 2.6 所示。

表 2.6 ADC 板卡性能参数

项目	参数	备注
接口类型	DB37	×2
信号类型	单端	-
通道数量	32ch	-
转换精度	16bit	-
转换时间	1.25 μs	-
最大采样率	800Kbps	-
输入范围	±10V	-
输入阻抗	1MΩ	-
信噪比	95.5dB	-

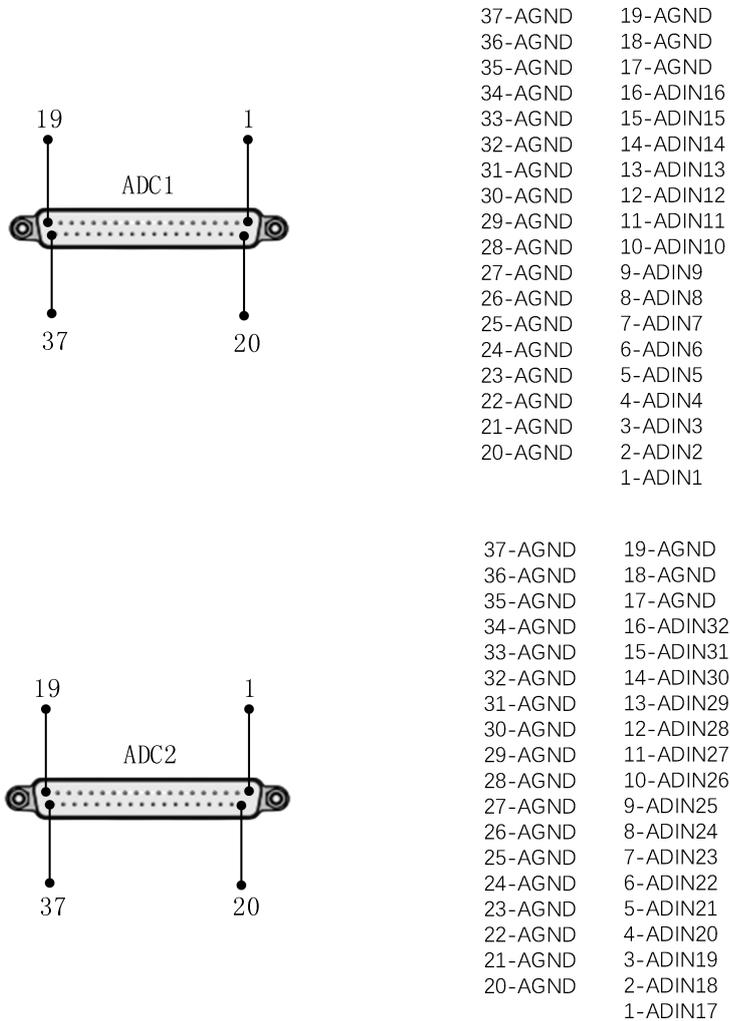


图 2.6 RTU-BOX800ADC 接口引脚定义

2.6. PWM 接口介绍

RTU-BOX800的PWM接口共有30路由FPGA控制的PWM信号,其中PWM25~PWM30与DO复用,同时还有6路DI位于该接口。引脚定义如图2.7所示。

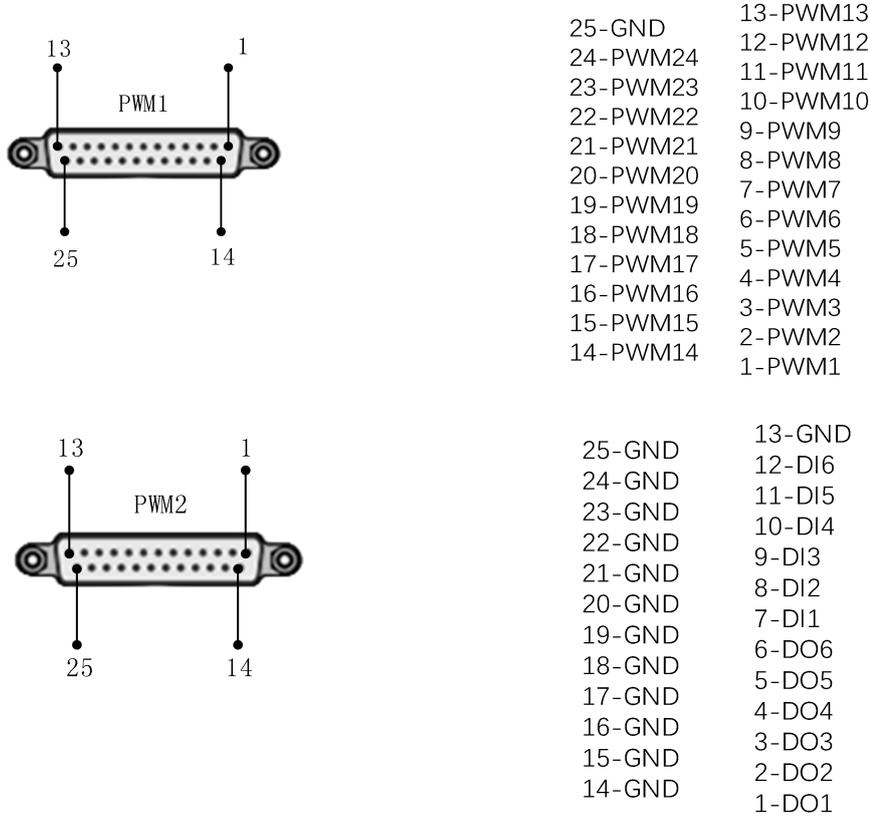
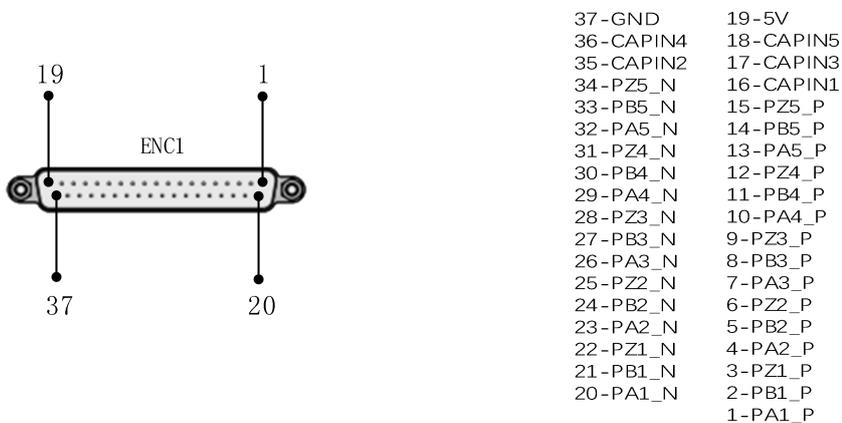


图 2.7 RTU-BOX800PWM 接口引脚定义

2.7. ENC 接口介绍

ENC 口是编码器信号输入接口,可处理增量式编码器、绝对值编码器和旋转变压器信号。其引脚定义如图2.8所示。



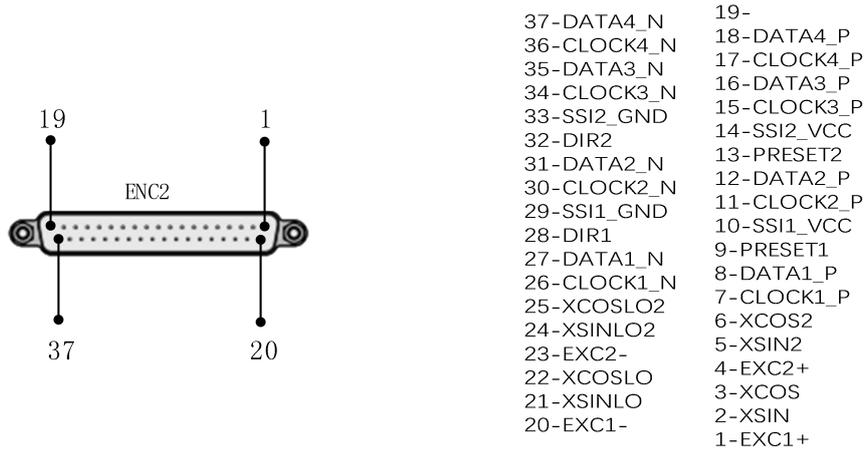


图 2.8 RTU-BOX800ENC 接口引脚定义

2.8. DAC 接口介绍

DAC 板用于将数字信号转换为模拟信号并通过 DAC 口输出，RTU-BOX800 的 DAC 板共有 16 路，可以同时输出 16 路模拟量信号，模拟量的电压范围为 ±10V。其中 DAC1~DAC4 可以通过 SMA 线直接连接到示波器观测。

引脚定义如图 2.9 所示，DAC 性能参数如表 2.7 所示。

表 2.7 DAC 板卡性能参数

项目	参数	备注
接口类型	SMA	内孔，RF 连接器/同轴连接器，阻抗 50Ω
通道数量	16ch	-
转换精度	16bit	-
建立时间	15 μs	TO 1LSB, from 0000h to FFFFh
负载电流	< ±10mA	-
输出范围	±10V	-
偏移量	< ±15mV	-

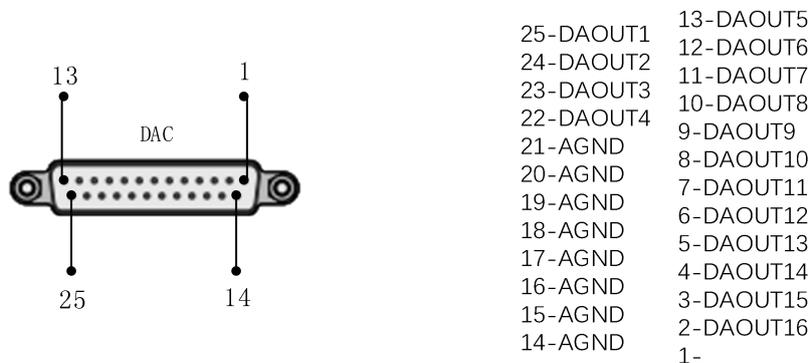


图 2.9 RTU-BOX800DAC 接口引脚定义

3. RTU-BOX800 开发环境搭建

摘要： 在您使用 RTU-BOX800 之前，首先需要按照本章说明搭建开发环境。

3.1. 概述

在使用 RTU-BOX800 之前，首先需要确认您的电脑操作系统是 Windows10 专业版，需安装 Code Composer Studio V9.1.0（以下简称 CCS）或更高级版本、Matlab® R2016b 或更高级版本、Rtunit Studio2023（以下简称 RTUS2023），然后在 CCS 根目录下安装 processor_sdk_rtos_am57xx_09_03_00_00 库文件（建议版本保持一致），在 Matlab/Simulink 中安装 RTU-BOX800 Toolbox，最后在 RTUS2023 中配置 CCS 与 Matlab。本手册不提供 CCS 与 Matlab 软件安装教程，若您需要视频指导，可访问官网 www.rtunit.com 查看具体操作视频。

注意事项：

- 1) 请使用正版软件。为了能够得到更好的体验、服务与支持，请使用我司的正版软件；
- 2) 计算机操作系统请使用非家庭中文版；
- 3) 我们不提供也不出售 Matlab 软件，请向 Math Works 公司购买正版 Matlab 软件；
- 4) RTUS 需与 USB 密钥配合使用；
- 5) 请在 Ti 官网下载 CCS；
- 6) 各软件安装文件所在的目录不能有中文字符；（用户名也不能有中文请检查 C 盘用户下的账户文件夹是否为中文;如果是，请改为英文）
- 7) 各软件安装路径中不能有中文字符，也不能有空格，比如“Program Files”；
- 8) 计算机上建议不要运行杀毒软件，比如 360 安全卫士，存在误删软件的风险。

由于在产品使用过程中需要运行专业软件，建议电脑运行内存大于等于 16G。

3.2. 安装 Rtunit Studio2023

Rtunit Studio 是南京瑞途优特信息科技有限公司（rtunit®）自主研发，供 RTU-BOX 系列产品使用的集成开发环境。Rtunit Studio2023 为绿色免安装软件，使用时需在 USB 接口插入密钥，安装教程如下：

- 1、解压 RTUS2023 压缩包至 RTUS2023 文件夹；

名称	修改日期	类型	大小
 RTUS2023	2022-11-10 16:39	文件夹	
 RTUS2023	2023-01-31 14:18	WinRAR 压缩文件	134,575 KB

图 3.1 RTUS2023 安装过程图 1

2、文件路径：RTUS2023/bin/RtunitStudio，如图 3.2，双击打开 RtunitStudio 即可打开软件，如图 3.3；

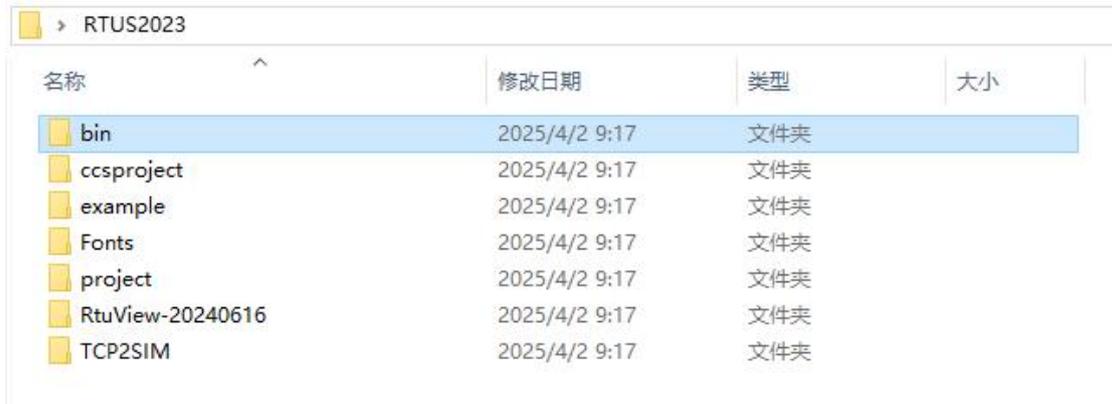


图 3.2 RTUS2023 安装过程图 2

msvcp140_1.dll	2025/4/2 9:17	应用程序扩展	29 KB
msvcp140_2.dll	2025/4/2 9:17	应用程序扩展	171 KB
msvcp140_codecvt_ids.dll	2025/4/2 9:17	应用程序扩展	26 KB
msvcr100.dll	2025/4/2 9:17	应用程序扩展	753 KB
RtuCompress.dll	2025/4/2 9:17	应用程序扩展	96 KB
rtunit.ini	2025/4/2 9:28	配置设置	1 KB
RtunitStudio.exe	2025/4/2 9:17	应用程序	7,573 KB
RtunitView.exe	2025/4/2 9:17	应用程序	6,374 KB
RtuOcs.dll	2025/4/2 9:17	应用程序扩展	71 KB
RtuProject.dll	2025/4/2 9:17	应用程序扩展	252 KB
RtuProjectView.dll	2025/4/2 9:17	应用程序扩展	250 KB
RtuRecord.dll	2025/4/2 9:17	应用程序扩展	60 KB

图 3.3 RTUS2023 安装过程图 3

3、在弹窗“系统配置”界面配置 Matlab 和 CCS 软件，如图 3.4。若不小心删除此弹窗，则点击左上角图标打开系统配置界面，如图 3.5；

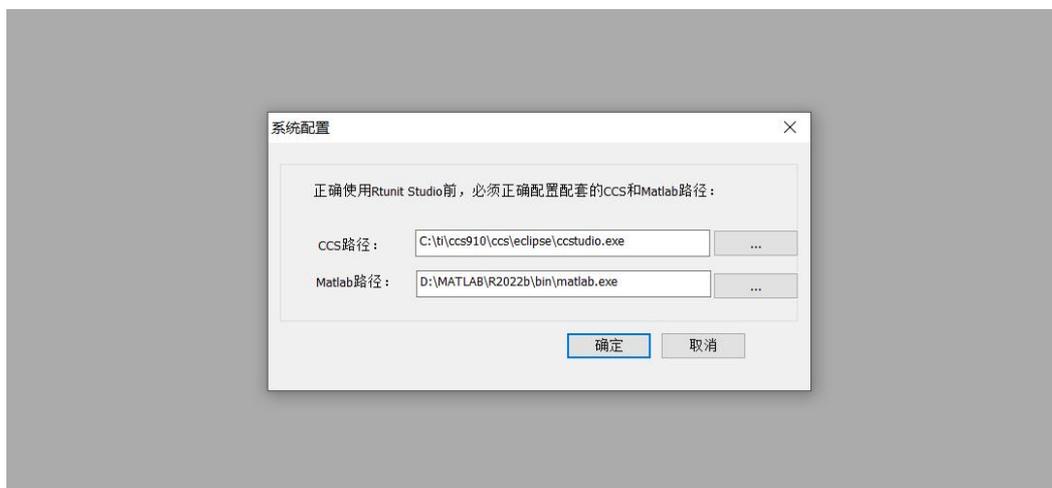


图 3.4 RTUS2023 安装过程图 4

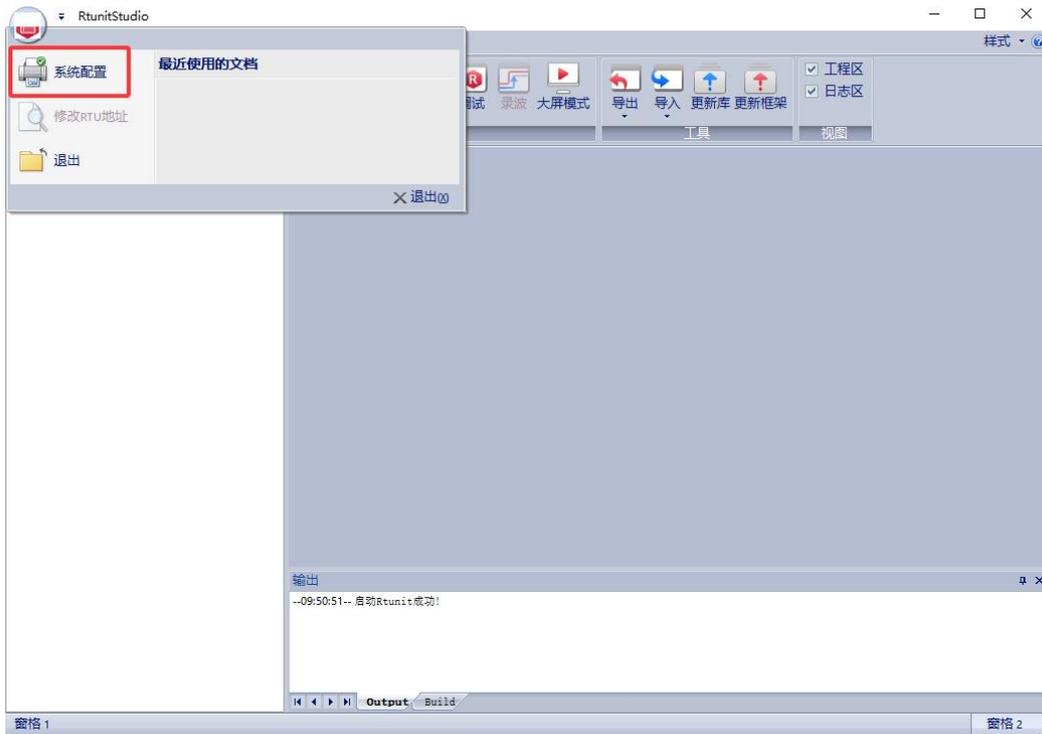


图 3.5 RTUS2023 安装过程图 5

- 4、在系统配置界面中添加 Matlab、CCS 的路径，点击浏览找到 Matlab 与 CCS 安装路径下的图标位置，分别点击“打开”后点击“确认”，即配置完成；

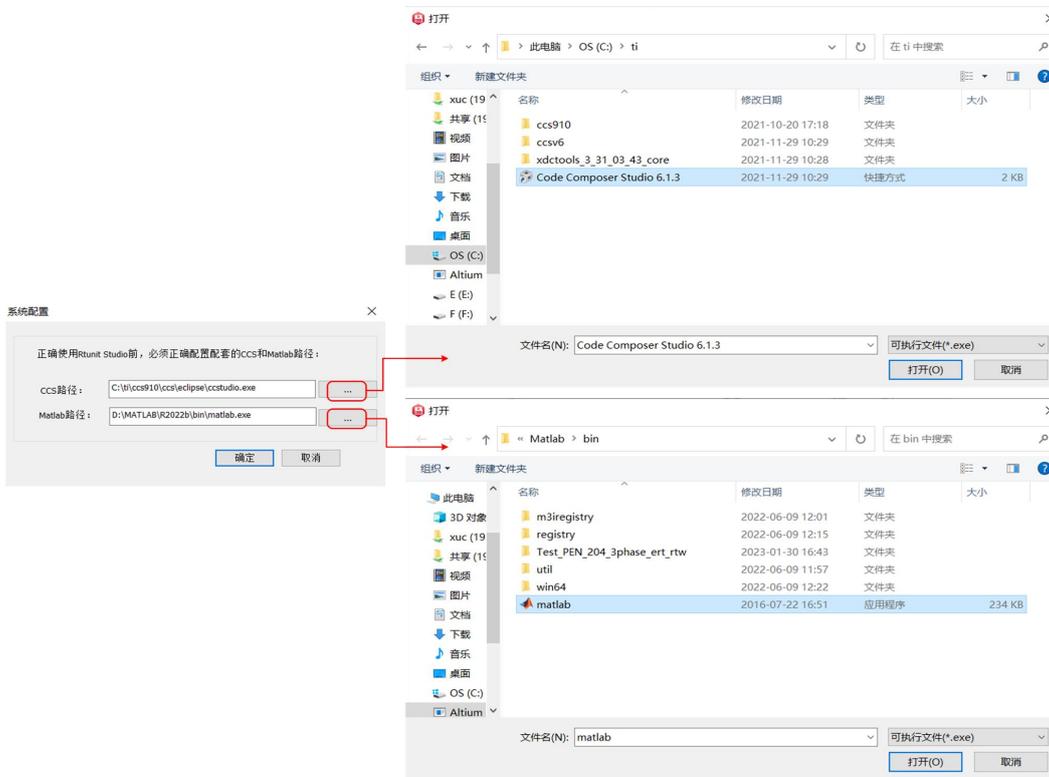


图 3.6 RTUS2023 安装过程图 6

3.3. 安装 RTU-BOX800 Toolbox

RTU-BOX800 Toolbox 中包含 Simulink 中的 Rtunit Toolbox 库文件以及框架程序。其中，RTU-BOX800 Toolbox 库用于存放与 RTU-BOX800 硬件功能配套使用的 Simulink 模型，框架程序是 RTU-BOX800 的底层硬件框架，与自动生成的 C 代码结合组成完成工程。安装教程如下：

1、打开 Matlab 软件，在图标处打开文件夹 RTUS2023/bin/Rtunit_Lib；

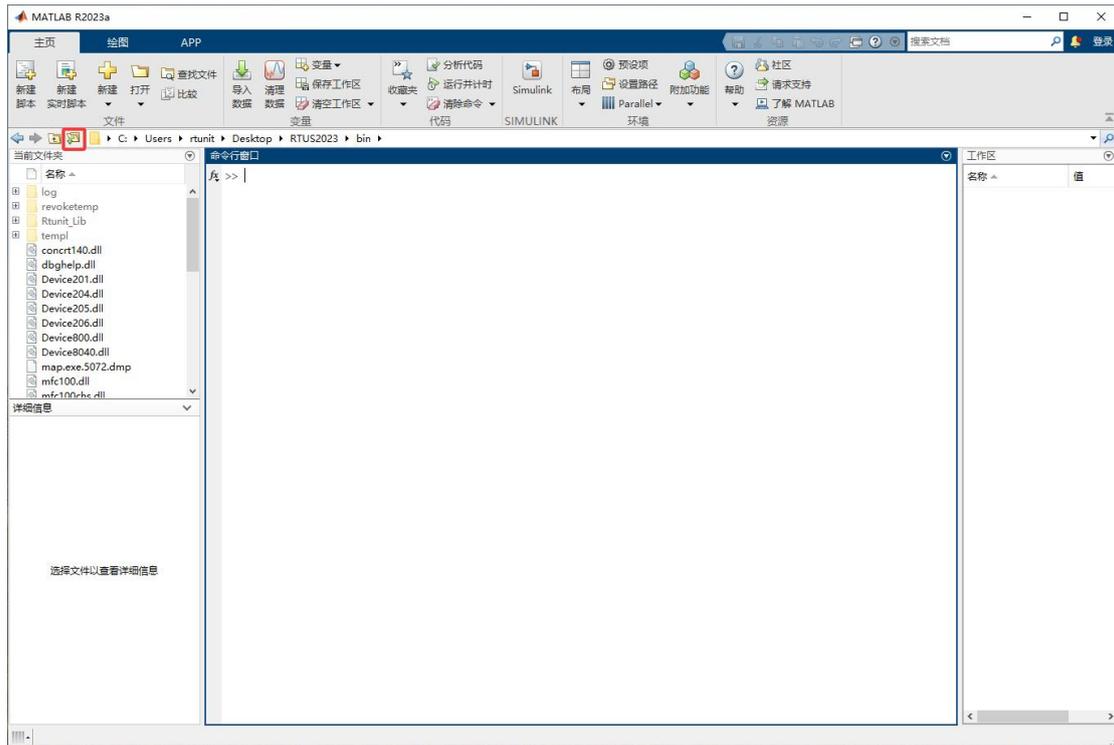


图 3.7 Rtunit_Lib 安装过程图 1

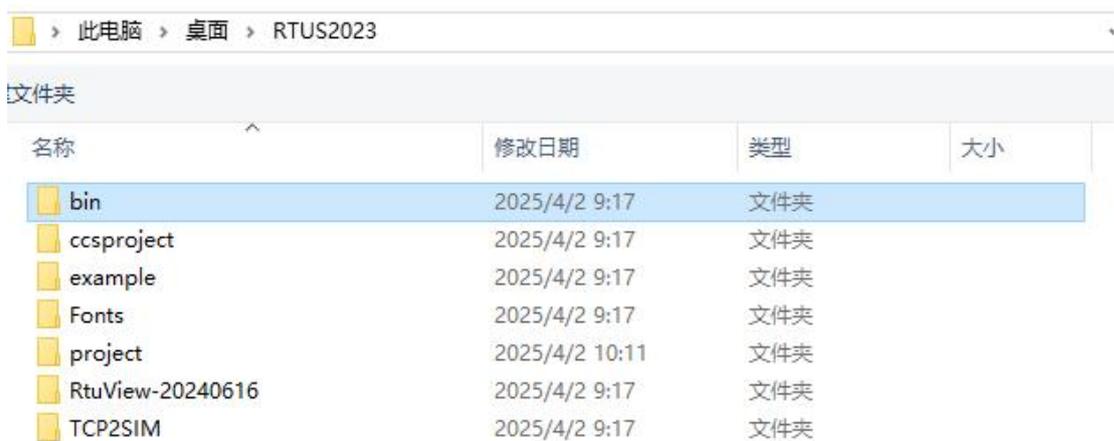


图 3.8 Rtunit_Lib 安装过程图 2

2、打开 Rtunit_Lib 文件夹，选中 800 库文件，点击“选择文件夹”；

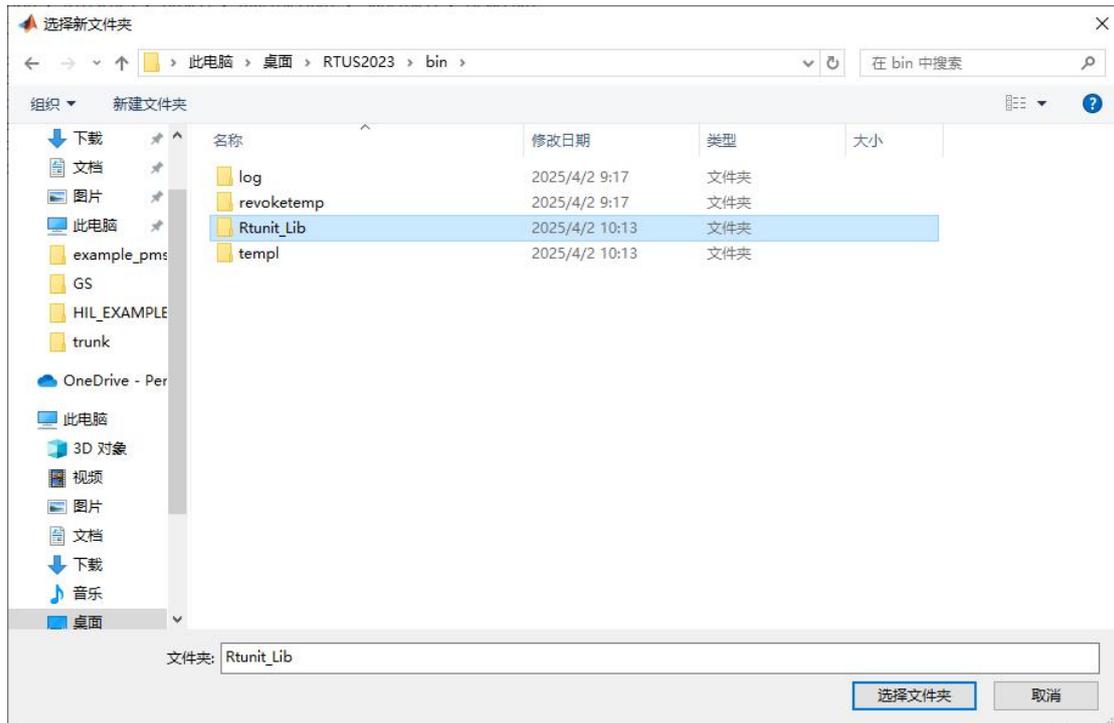


图 3.9 Rtunit_Lib 安装过程图 3

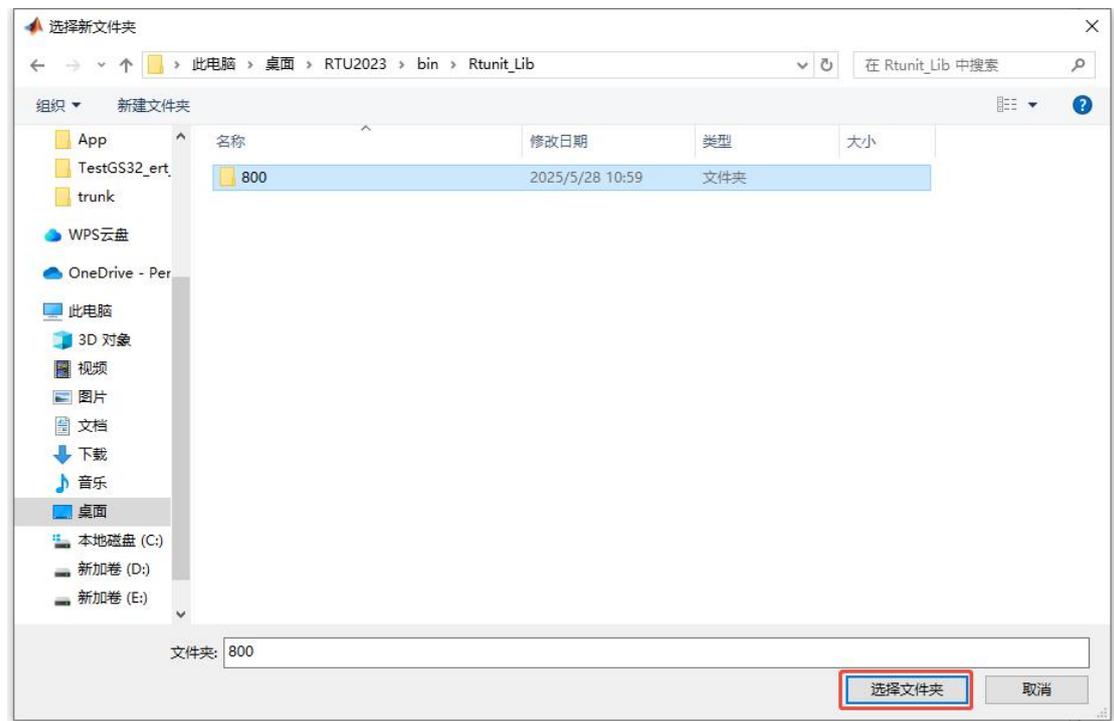


图 3.10 Rtunit_Lib 安装过程图 4

3、在 800 文件夹中找到 LibSetup.p 文件，右键点击运行，安装完成后在命令行窗口可以看到安装成功的提示，此外打开设置路径可以看到库文件所在路径已成功添加；

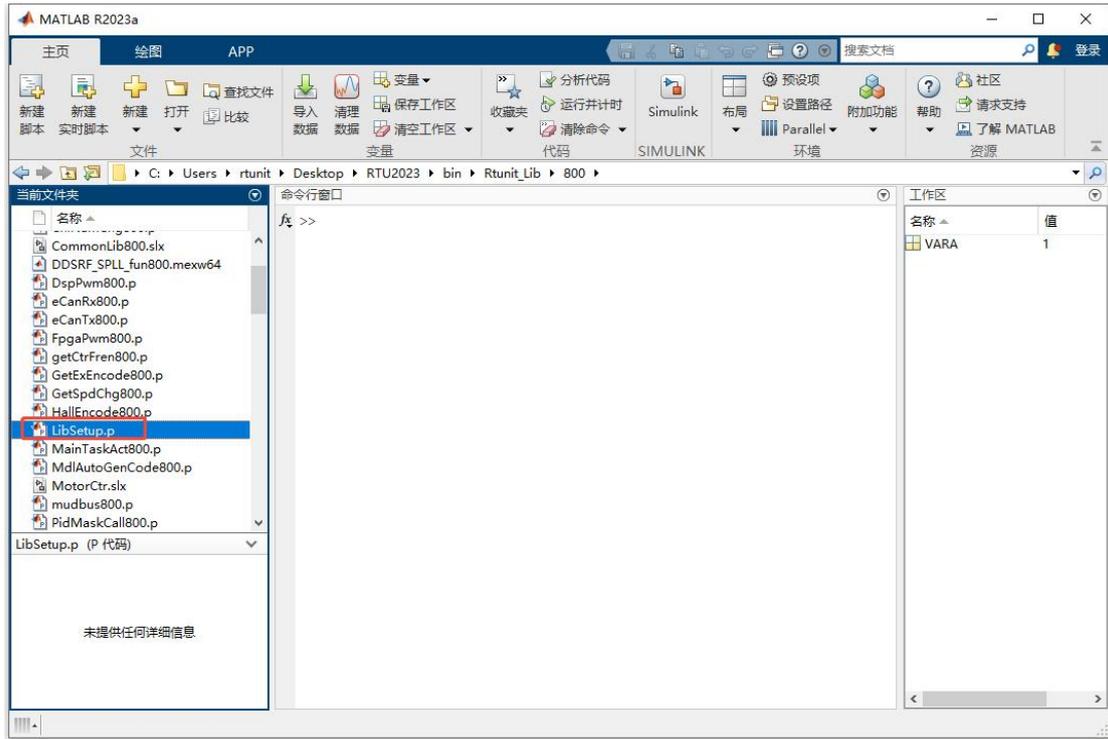


图 3.11 Rtunit_Lib 安装过程图 5

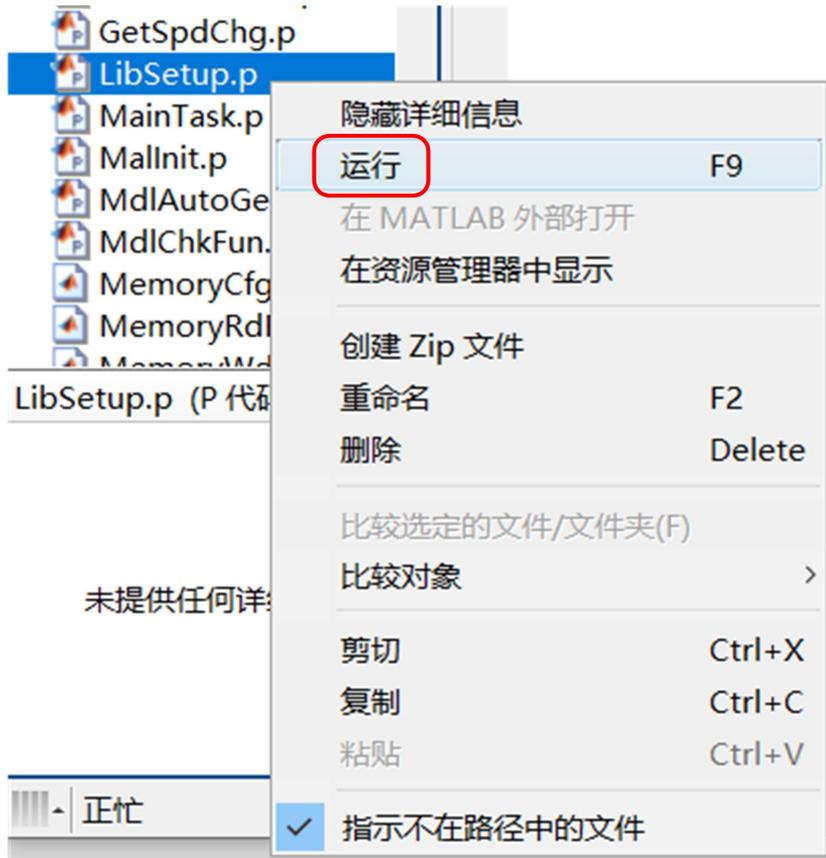


图 3.12 Rtunit_Lib 安装过程图 6

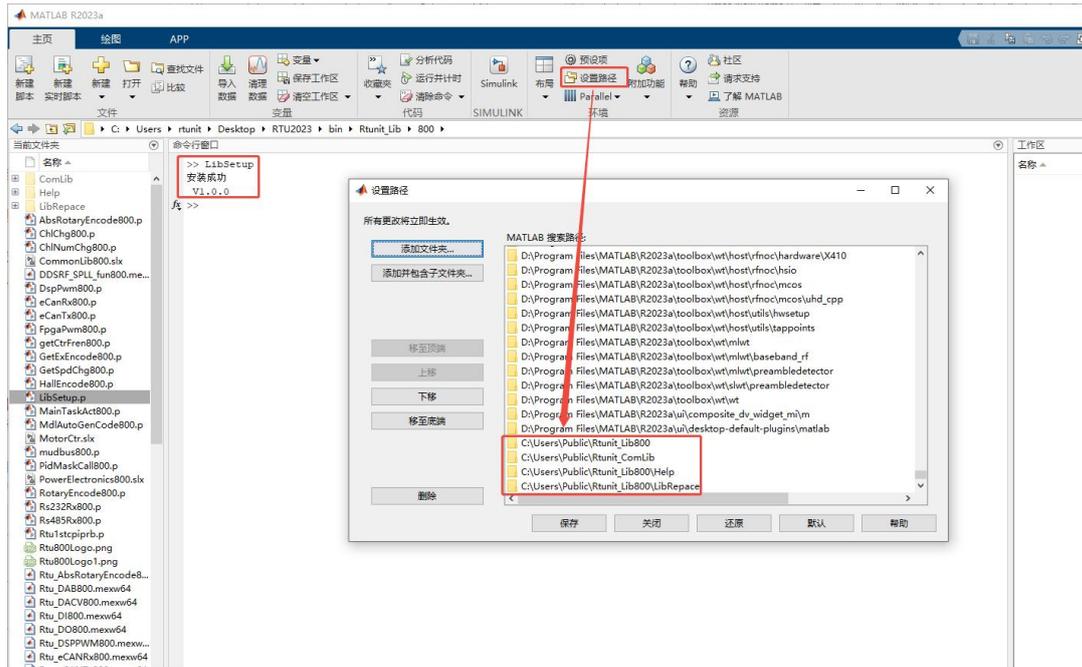


图 3.13 Rtunit_Lib 安装过程图 7

3.4. CCS 运行环境配置

RTU-BOX800 在使用时，需要在 C:/ti 目录下安装 processor_sdk_rtos_am57xx 库文件(可从 TI 官网直接下载)；

注意：根目录并非软件安装目录；

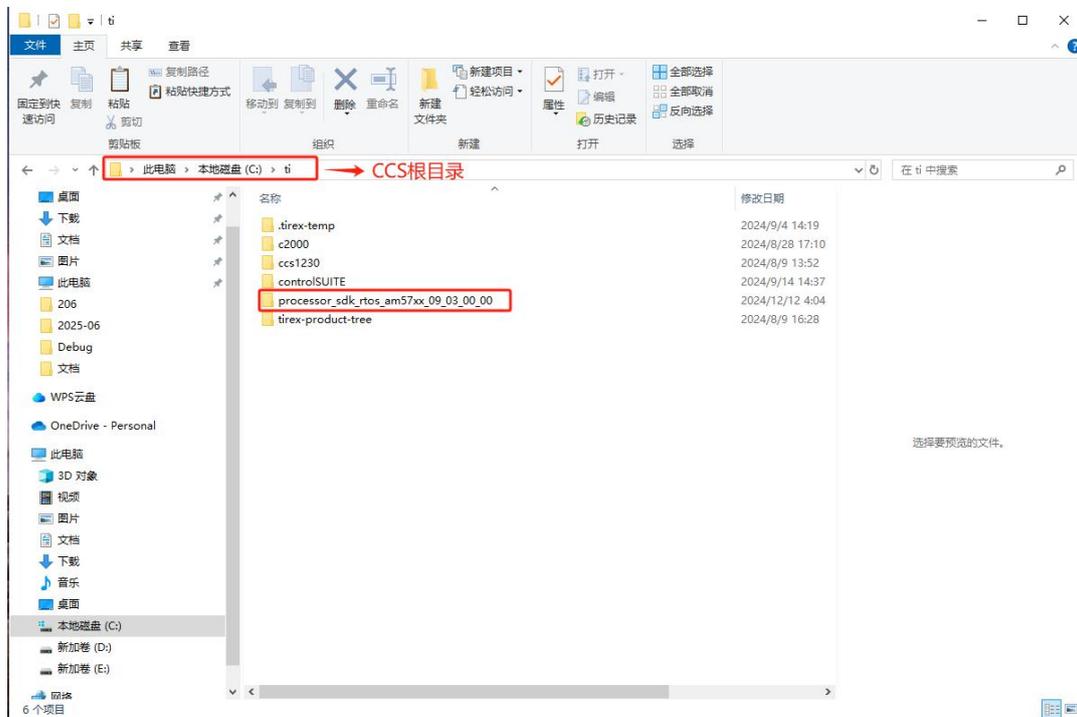


图 3.14 CCS 运行环境配置图 1

3.5. 开发环境配置

RTU-BOX800 在使用时，不论是更新程序还是传输数据，都是通过以太网与计算机进行通信。首先需要对计算机网络进行相应的配置（以 Windows 10 系统为例）。

1、在开始菜单栏里点击“设置”——点击“网络和 Internet”；

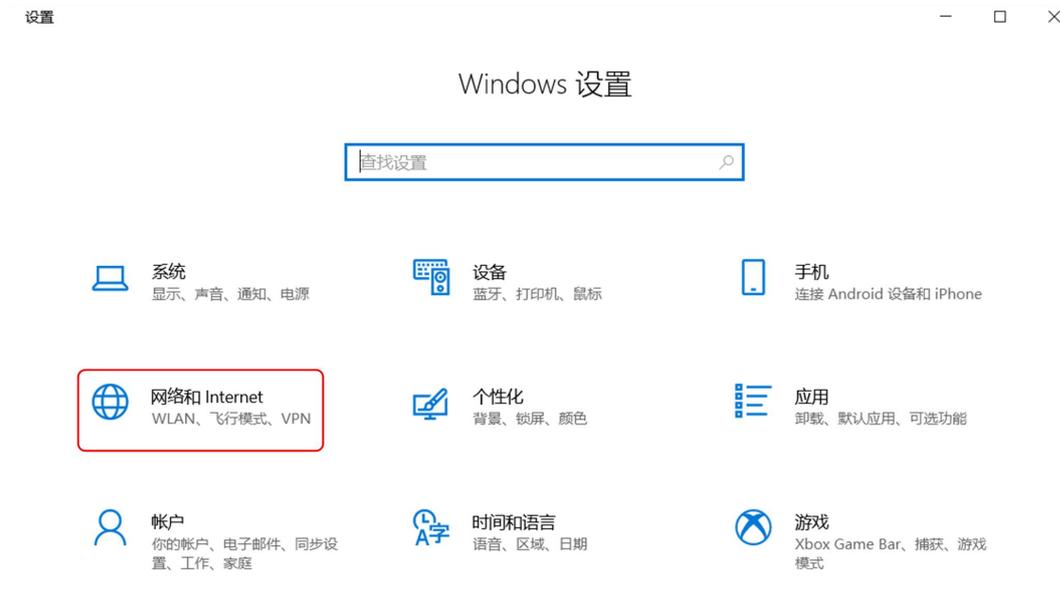


图 3.15 开发环境配置图 1

2、点击“更改适配器选项”；



图 3.16 开发环境配置图 2

3、点击“以太网”，打开以太网“属性”设置界面；

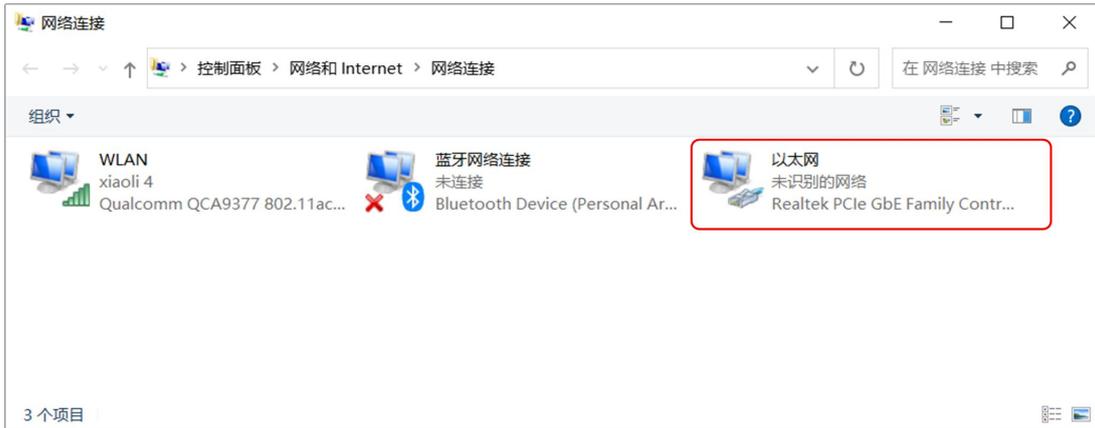


图 3.17 开发环境配置图 3

4、点击“Internet 协议版本 4 (TCP/IPv4)”；



图 3.18 开发环境配置图 4

5、设置计算机的 IP 地址，子网掩码和默认网关：

IP 地址：192.168.16.XX。RTU-BOX800 的 IP 地址预设为 192.168.16.10，因此 XX≠10；

子网掩码：255.255.255.0；

默认网关：192.168.16.1；

说明：当有多个网卡在配置 IPv4 时，IP 地址不能相同。



图 3.19 开发环境配置图 5

6、测试 RTU-BOX800 与计算机之间的通信是否正常：

用以太网线连接 RTU-BOX800 的网络接口与计算机，打开电源开关，在计算机任务栏搜索“cmd.exe”或（快捷键 WIN+R 输入 cmd 后回车），打开命令程序；



图 3.20 开发环境配置图 6

在 cmd.exe 中输入“ping 192.168.16.10”，RTU-BOX800 的 IP 地址为 192.168.16.10。若看到图 3.21 所示反馈结果，则说明以太网连接正确，RTU-BOX800 与计算机之间的通信正常。

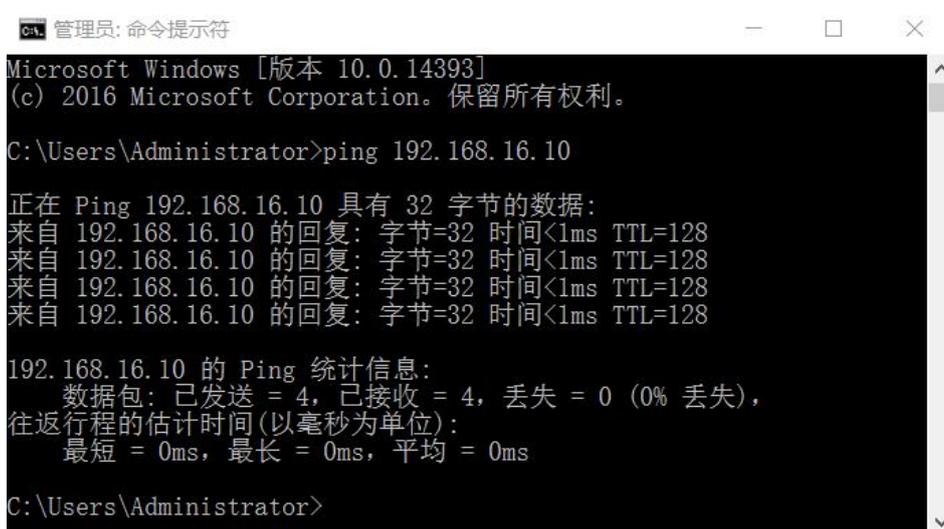


图 3.21 开发环境配置图 7

至此，RTU-BOX800 开发环境配置完成，请在后续章节的指导下进行相关开发工作。

4. Rtunit Studio2023 使用教程

摘要: Rtunit Studio2023 是与 RTU-BOX800 配套的集成开发环境。在您开始开发/科研工作之前请仔细阅读本章内容，本章将帮助您更好地使用 Rtunit Studio2023。

4.1. Rtunit Studio2023

Rtunit Studio2023 是 rtunit®自主研发的与 RTU-BOX800 配套的集成开发环境，与 Matlab 无缝衔接。在 Rtunit Studio2023 中，您可以新建及管理工程、升级 Simulink 库与框架程序、将 Simulink 模型转换为 C 语言程序、编译及下载程序、在程序运行的过程中可以实时修改参数、观测波形、导出数据等。Rtunit Studio2023 界面如图 4.1 所示。

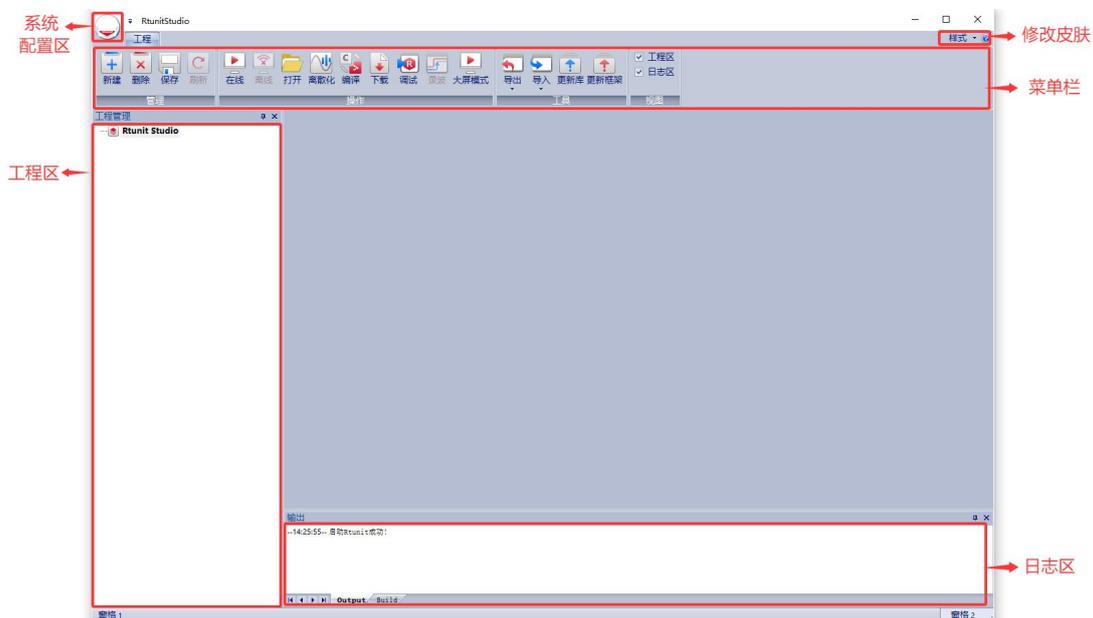


图 4.1 Rtunit Studio2023 界面

Rtunit Studio2023 界面由菜单栏、工程区、日志区、系统配置区四个主要部分组成。您可以根据喜好修改界面皮肤，工程区和日志区可以拖动至您习惯的地方。本章将以一个实例介绍如何使用 Rtunit Studio2023 开发与调试（Rtunit Studio2023 以下简称 RTUS）。

4.2. 新建工程

单击 RTUS 界面菜单栏“新建”按钮，弹出如图 4.2 所示菜单，填入工程项目名称。

工程名称命名规则：

- 1、文件名命名要用英文字符，第一个字符不能是数字和下划线。
- 2、命名尽量不要是简单的英文单词，最好是由大小写英文、数字、下划线等组成。
- 3、文件命名不能有空格，如 three phase，应该写成 three_phase 或者 ThreePhase。

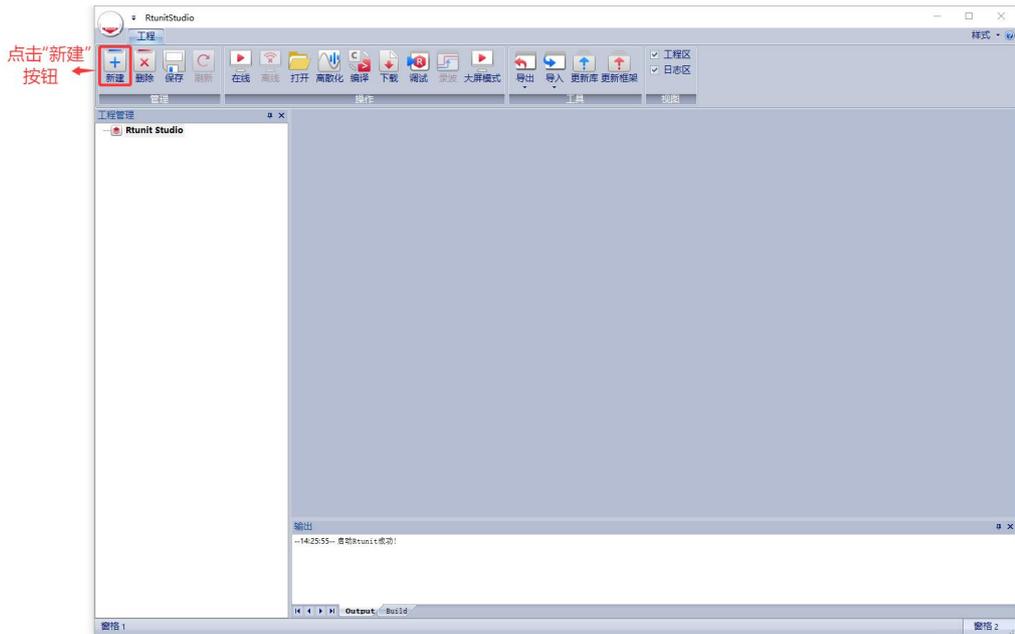


图 4.2 创建工程项目

4.3. 创建 Logic 模型

1、双击工程区域新建的 TEST1 工程，使其处于 Active Project 模式；

注意：RTUS 菜单栏的所有操作仅对当前处于 Active Project 模式下的工程有效。



图 4.3 创建 Logic 模型 1

2、右键点击 Device，新建一个设备工程，填入工程名称、设备类型以及 IP 地址相关信息，如图 4.4 所示；

注意：在同一个工程下，不同设备的 IP 地址必须互不相同，不能全部默认为 192.168.16.10

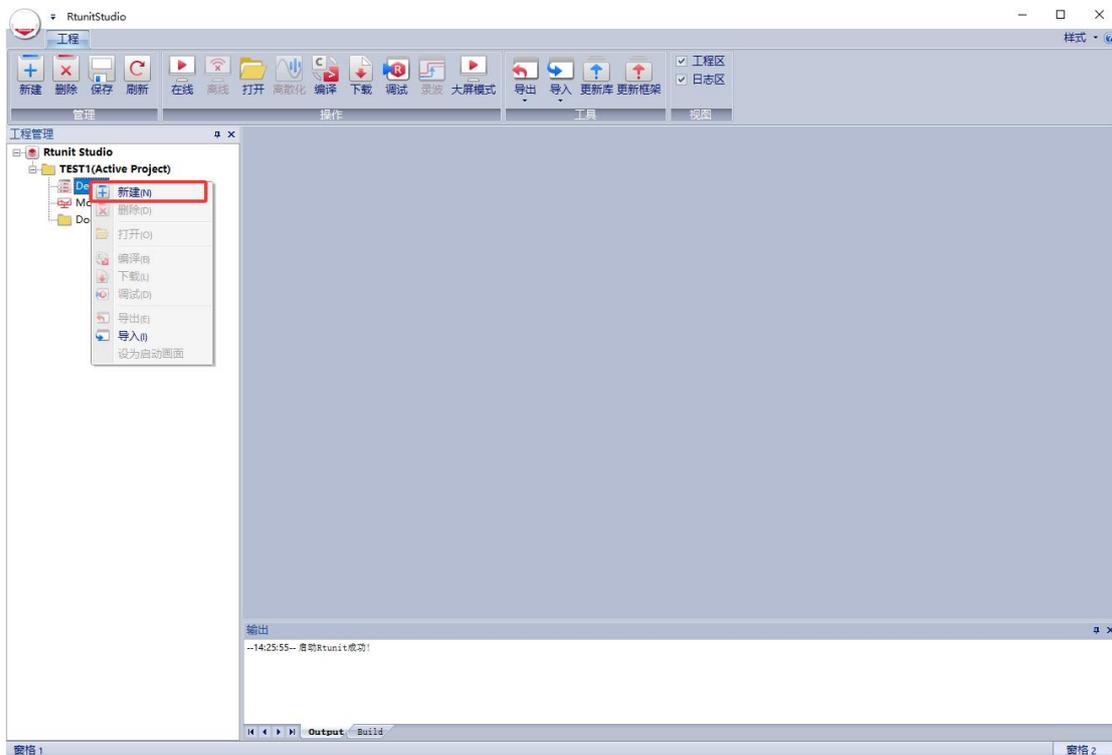




图 4.4 创建 Logic 模型 2

3、点击+展开 Device 目录以及目录下的工程 Device001，如图 4.5 所示；

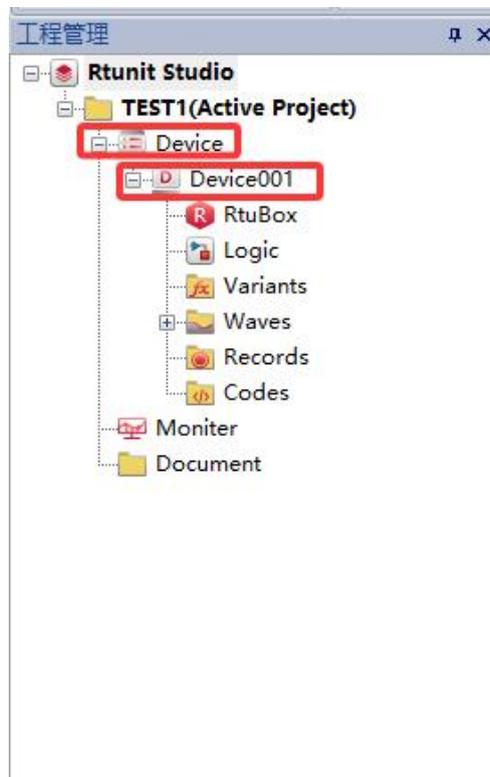


图 4.5 创建 Logic 模型 3

4、双击 Logic 按钮进入 Matlab/Simulink 界面，如图 4.6 所示；

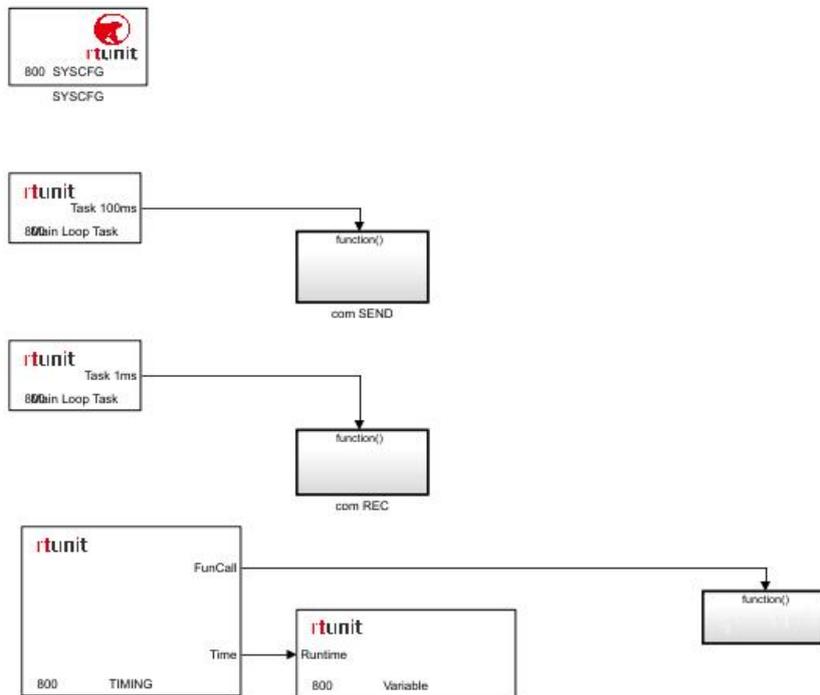


图 4.6 创建 Logic 模型 4

5、在此界面构建模型，如图 4.7 所示。请您参照第 5 章的 Rtunit Toolbox 相关介绍搭建模型。模型搭建完成之后先进行保存，然后单击图 4.7 中 RtuBox 按钮下的 800 Code Generate 生成代码，Logic 模型进入代码生成阶段；

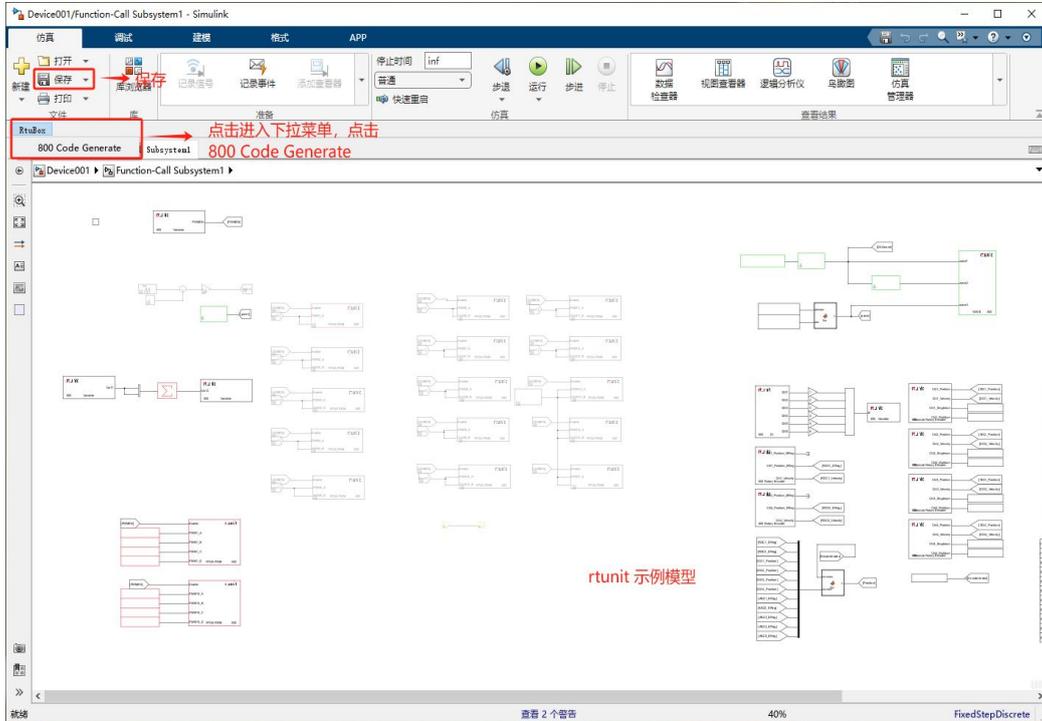


图 4.7 创建 Logic 模型 5

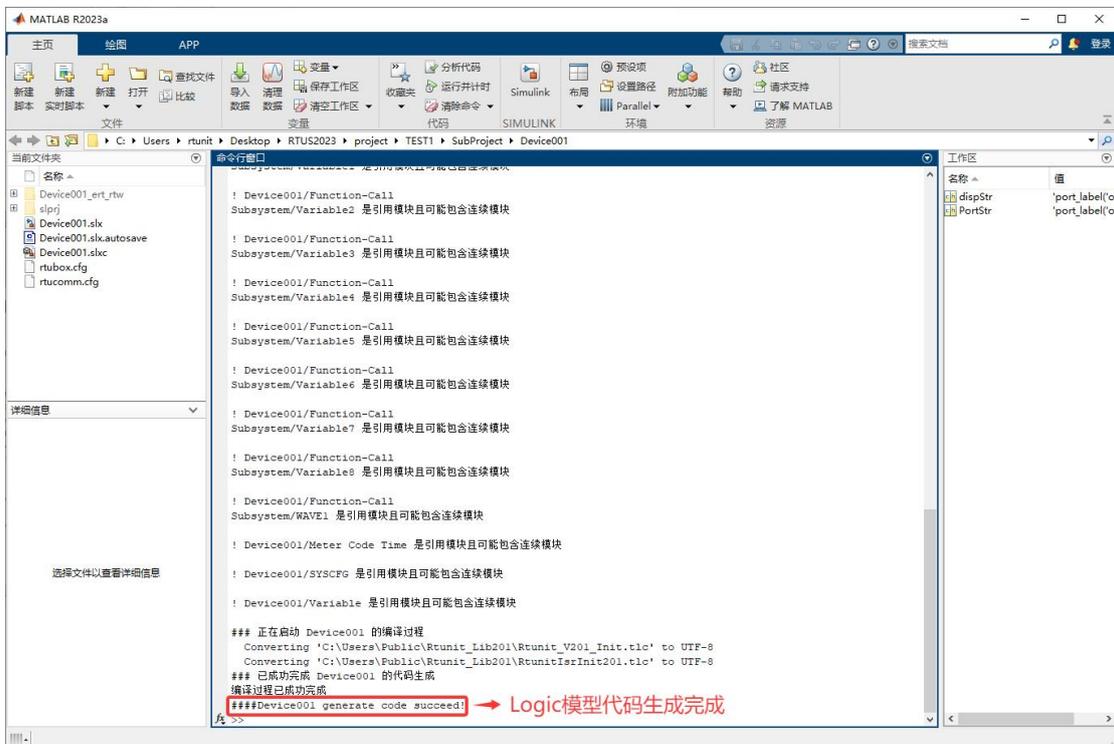


图 4.8 创建 Logic 模型 6

4.4. 编译&下载

1、Logic 模型代码生成完成后，在 RTUS 界面单击编译按钮，进入编译状态，待 RTUS 日志区显示编译成功后，完成编译；

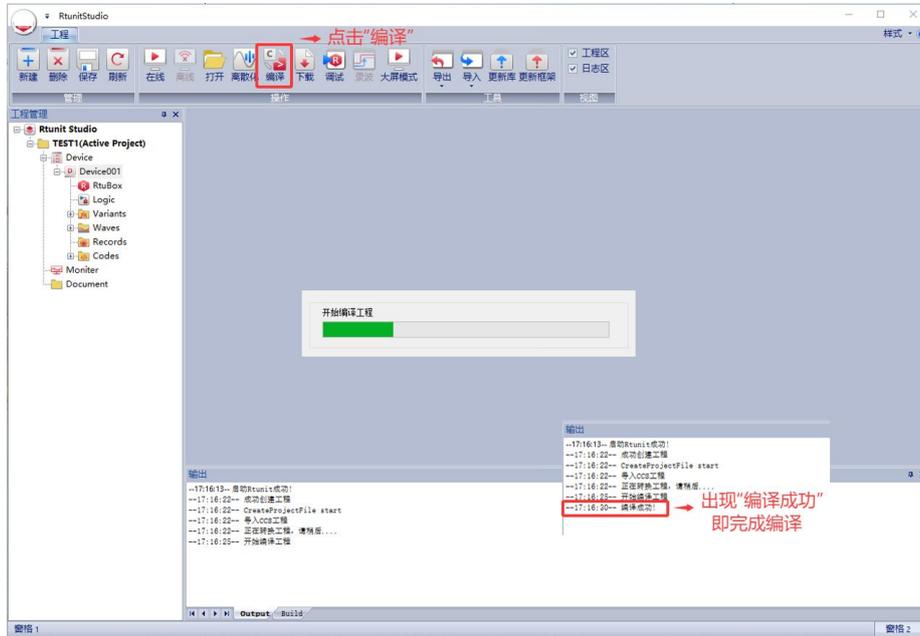


图 4.9 编译&下载 1

2、编译完成后，通过以太网线连接 RTU-BOX800 与计算机，单击 RTUS 菜单栏中的下载按钮将程序下载至 RTU-BOX800 中。程序成功完成更新，输出窗口显示烧写成功。

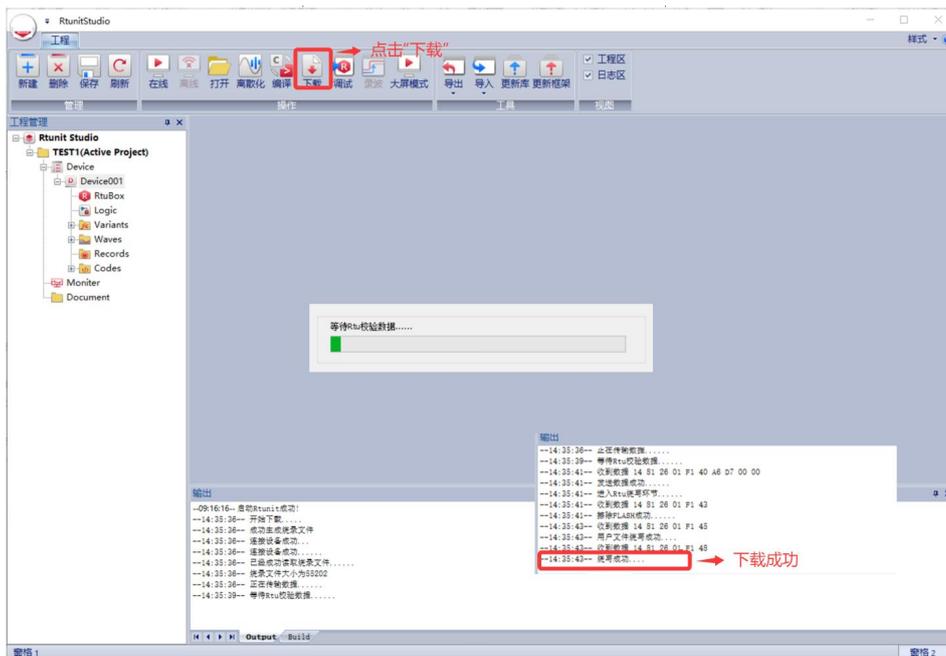


图 4.10 编译&下载 2

4.5. 调试

本章 4.3 节介绍了生成代码的 Logic 模型将经过编译&下载两步进入 RTU-BOX800 中, 您也可以通过调试按钮实现一键编译&下载。

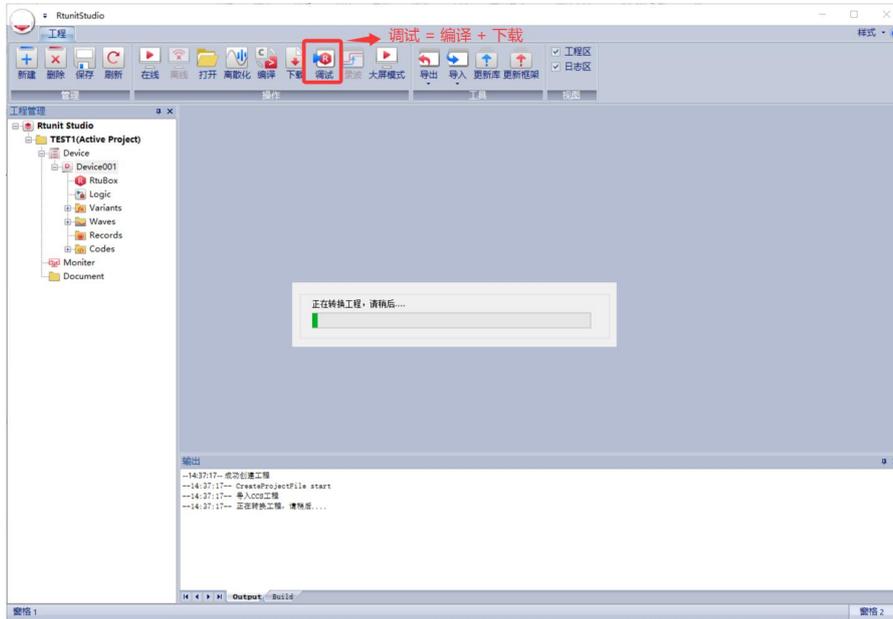


图 4.11 调试

4.6. 参数观测

您可以将想要调试/观测的参数(如电压 U1~U3)通过 Variable 模块(详见第五章)显示在 RTUS 界面上。双击 Variants 按钮弹出变量列表, 单击菜单栏中的在线按钮观测/调试, **注意: 待 RTU-BOX800 显示正常运行后, 方可单击在线按钮。**

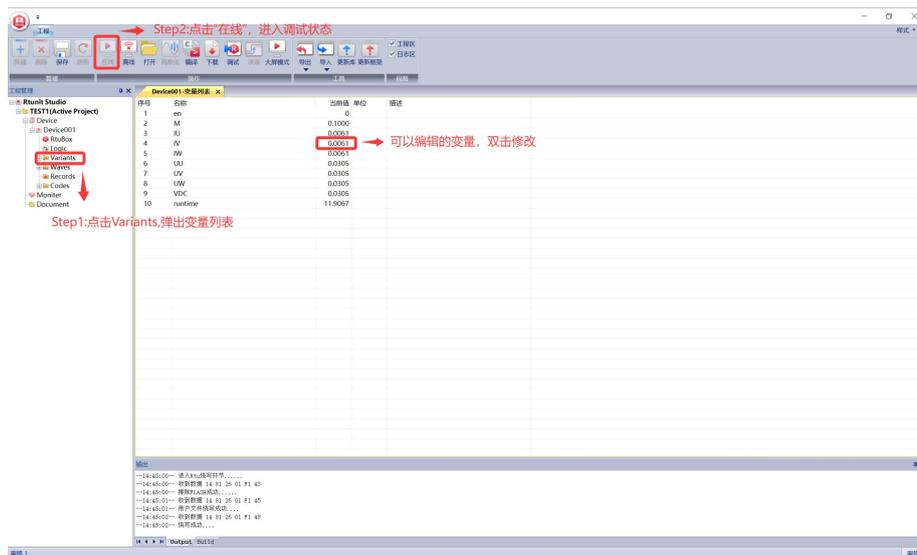


图 4.12 参数观测/调试

4.7. 波形观测

您可以将想要观测的参数波形通过 WAVE 模块（详见第五章）显示在 RTUS 界面上。
双击 Waves 图标，弹出查看波形窗口。

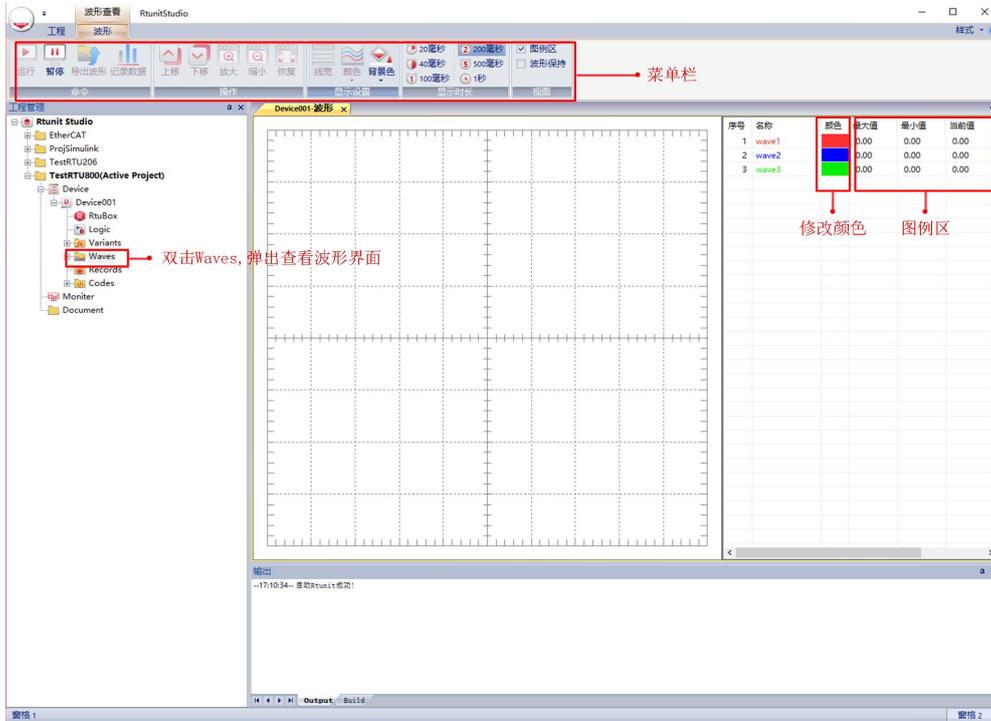


图 4.13 波形观测 1

Waves 菜单栏介绍:



图 4.14 波形观测 2

项目	备注
运行	波形随程序进行实时更新
暂停	波形暂停，不随程序进行更新
导出波形	波形暂停前提下，将波形数据以.csv 格式保存至指定位置
记录数据	将数据实时记录在.csv 文件中，双击 Waves\Data，可查看数据文件
上移	将选中的波形向上移动
下移	将选中的波形向下移动
放大	将选中的波形数值放大
缩小	将选中的波形数值缩小
恢复	将选中的波形恢复至初始状态（初始位置，初始值）
线宽	调整选中波形的线宽
颜色	调整选中波形的颜色
背景色	调整 Waves 的背景颜色
显示时长	调整显示波形窗口波形时长
图例区	显示/隐藏图例区
图形保持	保持 50Hz 整数倍的波形

注意：

- 1、实时波形 Wave 功能每通道每秒记录 2000 个数据；
- 2、导出波形功能仅保存当前暂停画面的数据；记录数据功能可实时保存所有数据。
- 3、波形显示背景和颜色均可以自由设置。

4.8. 查看代码

在 RTUS 中,Logic 模型进行 Code Generate 之后,您可以在 RTUS 界面查看生成的代码。



图 4.15 查看代码

RTU-BOX800 底层框架程序和代码生成机制均按照工程实际应用设计，因此生成的代码的可读性强，执行效率高。在程序调试的过程中，如果需要修改程序，有两种方法：

- 1、回到 Logic 模型修改，修改完成之后再按照 Code Generate→编译→下载的流程进行；
- 2、在 RTUS 的 Codes 中直接修改 C 语言代码，修改之后保存文件。重新编译→下载即可。

4.9. 导出

您可以将 RTUS 中的整个工程导出至指定位置（.rtu 工程文件），也可以将 Logic 模型算法导出至指定位置（.slx 算法文件）。



图 4.16 导出

4.10. 导入

您可以将完整的工程（工程文件）导入 RTUS，也可以只将 Logic 模型（算法文件）导入 RTUS。

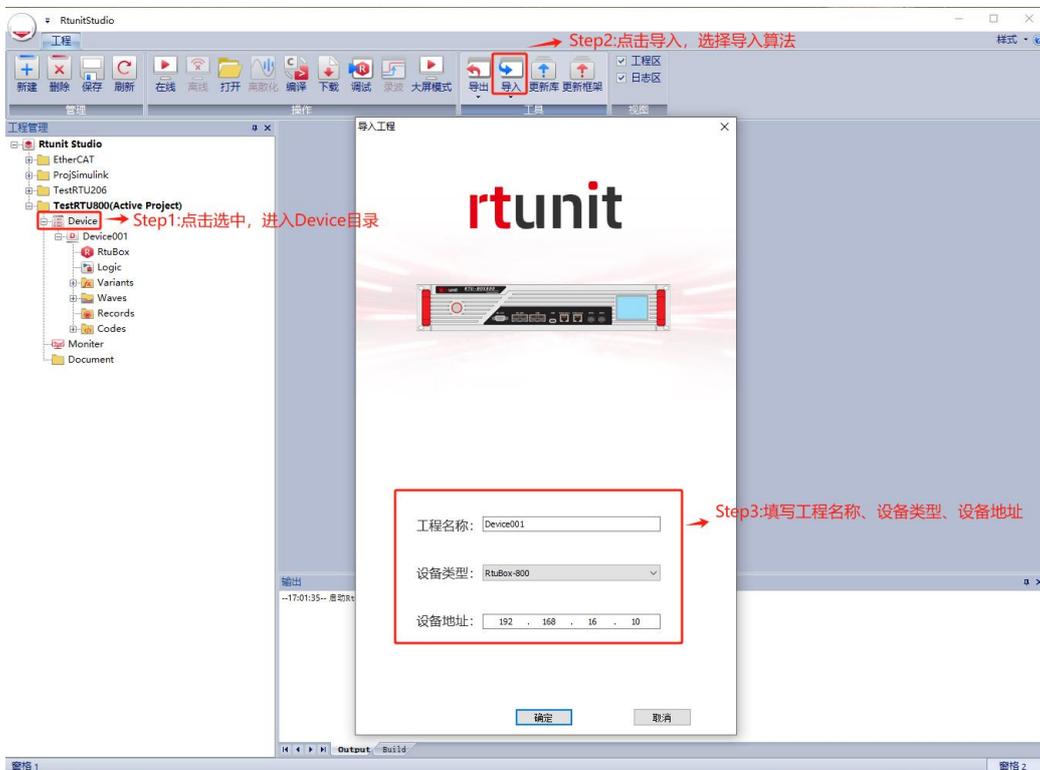


图 4.17 导入 2

4.11. 修改 RTU-BOX800 IP 地址

从 rtunit 出厂的所有 RTU-BOX800 的默认 IP 地址都是 192.168.16.10，您可以根据需要修改设备的 IP 地址。**注意：**RTU-BOX800 的 IP 地址与相连的计算机 TCP/IP 协议中的 IP 地址不能相同。

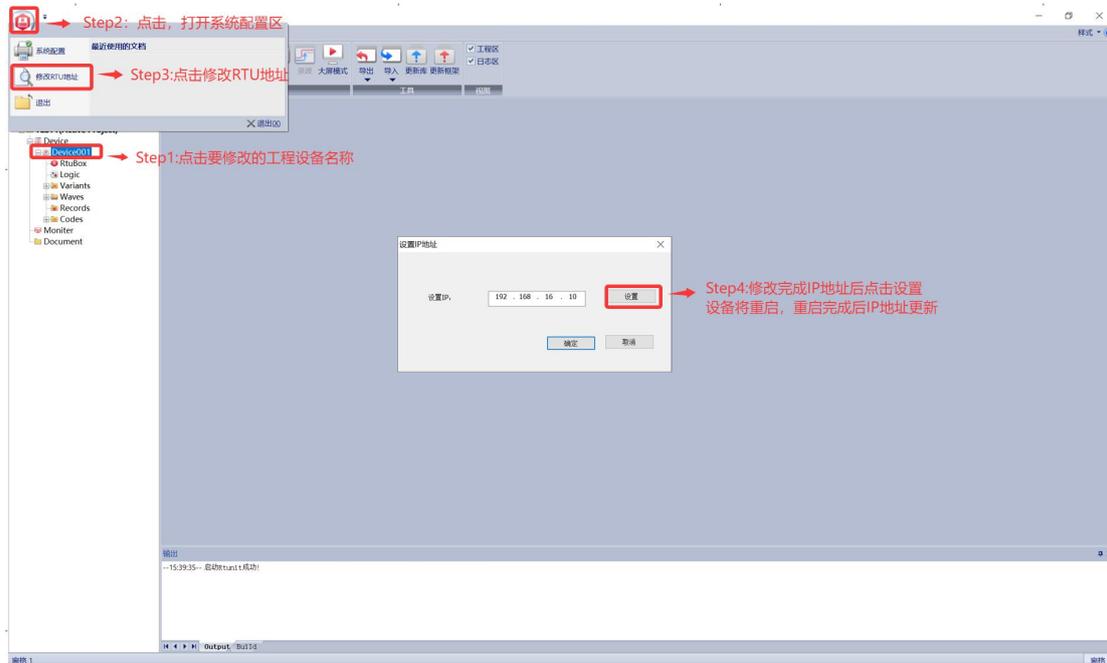


图 4.18 修改 RTU-BOX800 IP 地址

4.12. 更新 Rtunit Toolbox&更新框架

Rtunit Toolbox 是安装到 Simulink 的模型库，包含 RTU-BOX800 的硬件接口模型以及一些常用的算法模型。rtunit 将不断更新模型内容，若您想使用新的模型内容，请访问官网 www.rtunit.com 下载升级 Rtunit Toolbox 模型库。

在 RTUS 菜单栏选择更新库，在弹出的窗口选择 .tool 文件，等待 Matlab Command Window 弹出安装成功的提示。

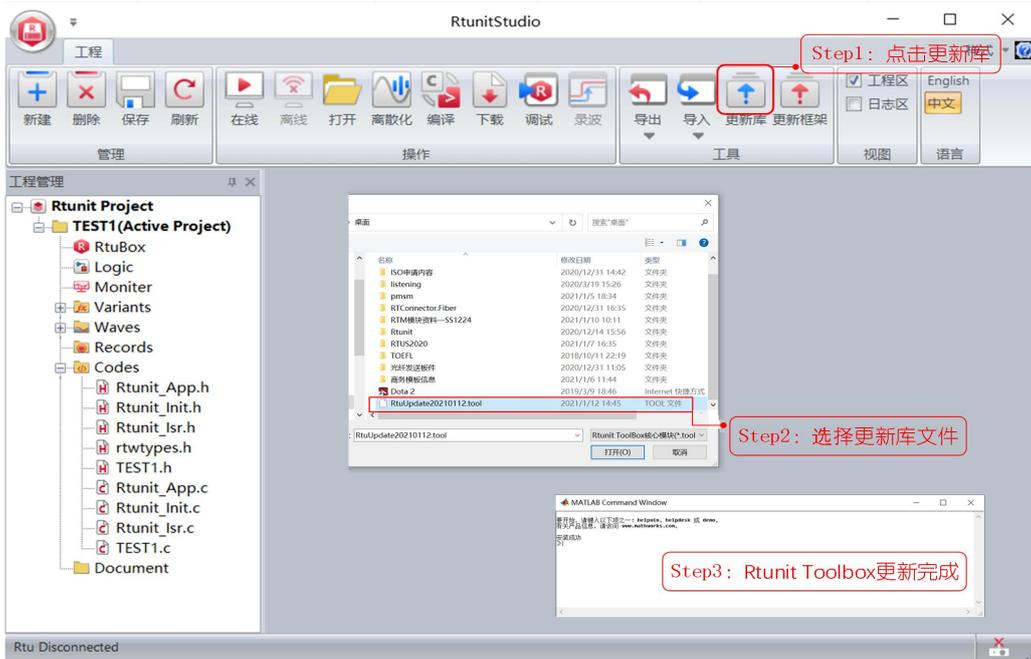


图 4.19 更新 Rtunit Toolbox

框架程序是 RTU-BOX 的底层硬件框架和自动生成的 C 语言代码结合组成完整的工程。随着 Rtunit Toolbox 的更新，框架程序也在更新。若您想使用下载新的框架程序，请访问官网 www.rtunit.com 下载升级 RTU 框架程序。

在 RTUS 菜单栏选择更新框架，在弹出的窗口选择.core 文件。

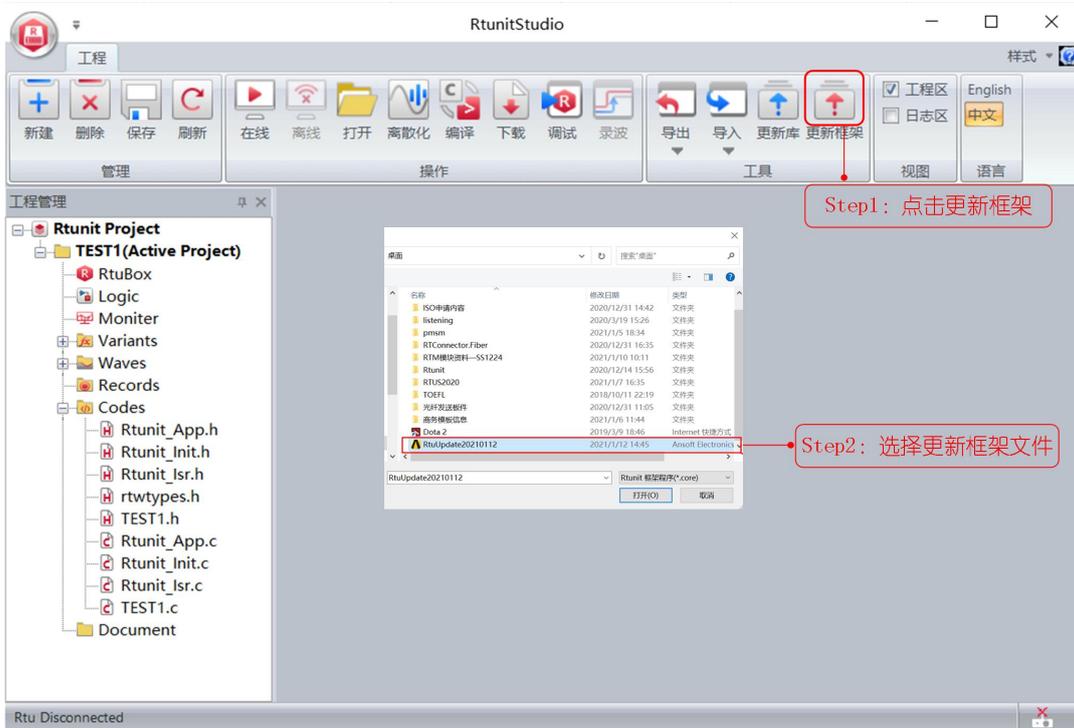


图 4.20 更新框架

5. Rtunit Toolbox 模型手册

摘要: Rtunit Toolbox 是 rtunit 自主研发集成于 Matlab®/Simulink®环境中的功能模块库。Rtunit Toolbox 是对 Simulink®工具箱的补充和扩展，提供了 RTU-BOX800 所有硬件的 Simulink 封装模块，在您搭建 Logic 模型之前，请仔细阅读本章内容。

5.1. Rtunit Toolbox

您在使用 Rtunit Toolbox 之前需要安装 RTU-BOX800 Toolbox，具体安装教程见 3.3。RTU-BOX800 Toolbox 安装在 Matlab/Simulink 库中，主要包括四类模块：Common Library，Power Electronics，RTU BOX，Transformations。

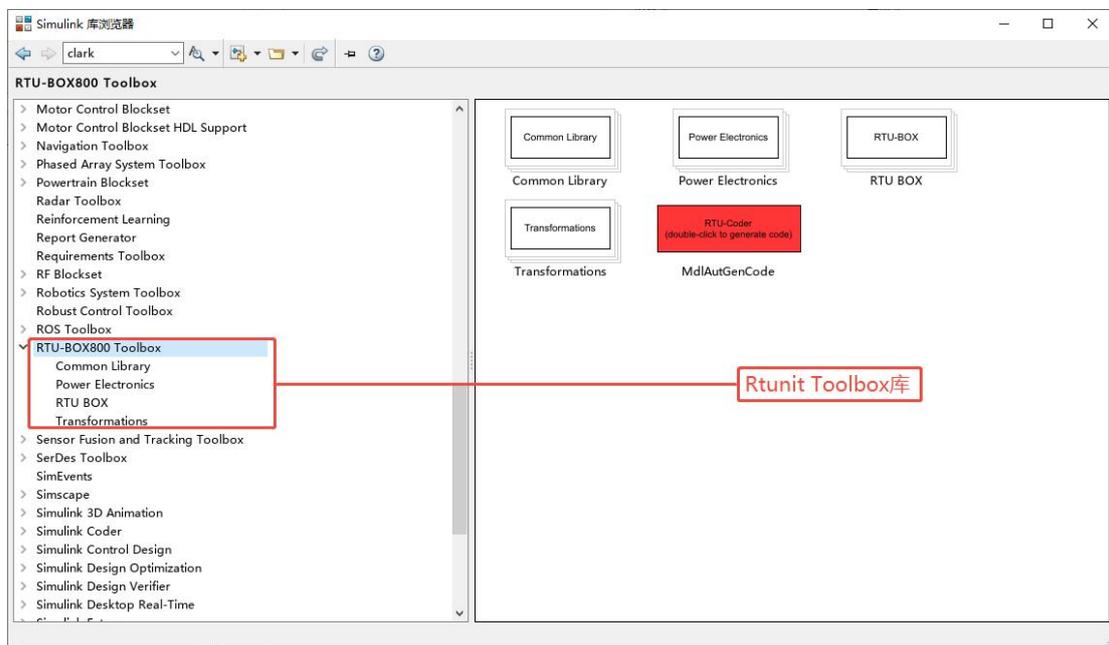


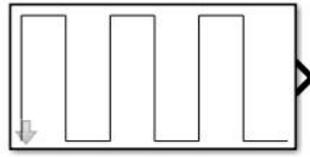
图 5.1 Rtunit Toolbox

5.2. Common Library

RTU BOX 库包含了与硬件接口相关的所有模块，用于实现 RTU-BOX800 的控制功能。本节将详细介绍 RTU-BOX800 库中的各个模块。

5.2.1. Pulse Wave 模块

Pulse Wave 模块用于输出频率、相位、幅值、偏置以及占空比均可设置的方波脉冲，双击 Pulse Wave 模块可进入模块参数设置界面，如 5.2 所示。Simulink 仿真结果如图 5.3 所示。



Pulse Wave



图 5.2Pulse Wave 模块

Pulse Wave 模块参数说明如下表 5.1 所示。

表 5.1Pulse Wave 模块参数说明

项目	名称	参数说明	数据类型
Frequency	频率	脉冲频率设置	-
Phase	相位	脉冲相位设置	-
Amplitude	幅值	脉冲幅值设置	-
Bias	偏置	脉冲偏置设置	-
Duty	占空比	脉冲占空比设置	-
Output	输出	输出值	Single

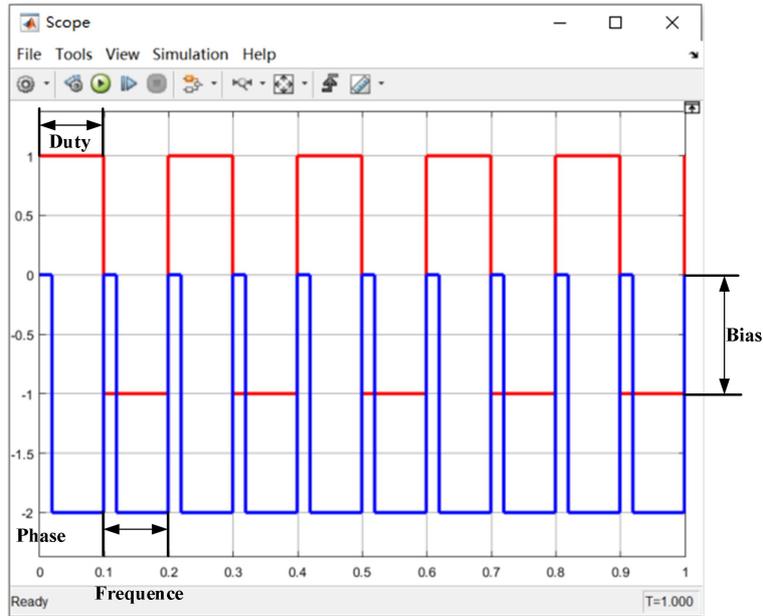


图 5.3 Pulse Wave 模块 Simulink 仿真

5.2.2. Sawtooth Wave 模块

Sawtooth Wave 模块用于输出频率可调的锯齿波，双击 Sawtooth Wave 模块可进入模块参数设置界面，如图 5.4 所示。Simulink 仿真结果如图 5.5 所示。

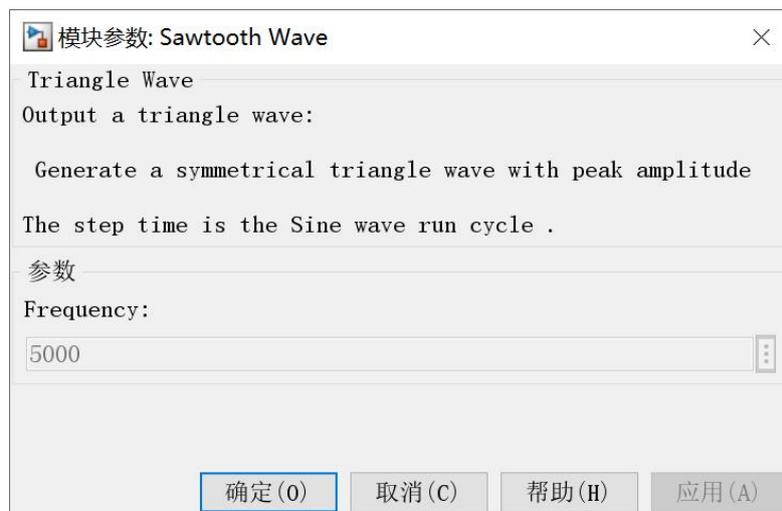
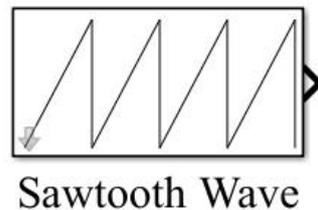


图 5.4 Sawtooth Wave 模块

Sawtooth Wave 模块参数说明如下表 5.2 所示。

表 5.2 Sawtooth Wave 模块参数说明

项目	名称	参数说明	数据类型
Frequency	频率	锯齿波频率设置	-
Output	输出	输出值	Single

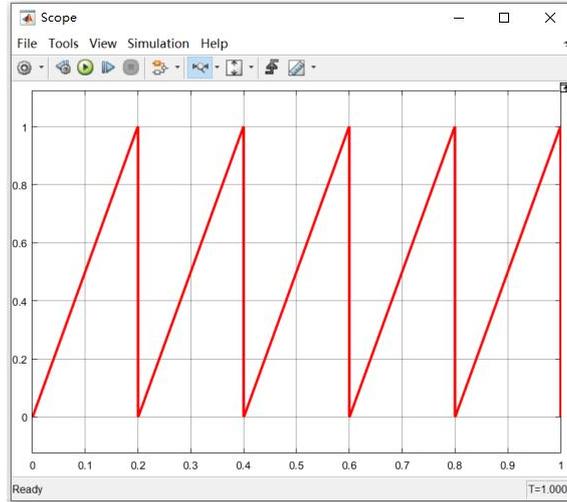
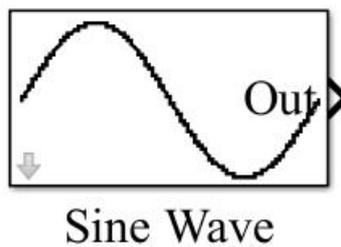


图 5.5 Sawtooth Wave 模块 Simulink 仿真

5.2.3. Sine Wave 模块

Sine Wave 模块用于输出频率、幅值、相位以及偏置均可设置的正弦波。其中，频率、幅值可在线修改，双击 Sine Wave 模块可进入模块参数设置界面，如图 5.6 所示。Simulink 仿真如图 5.7 所示。



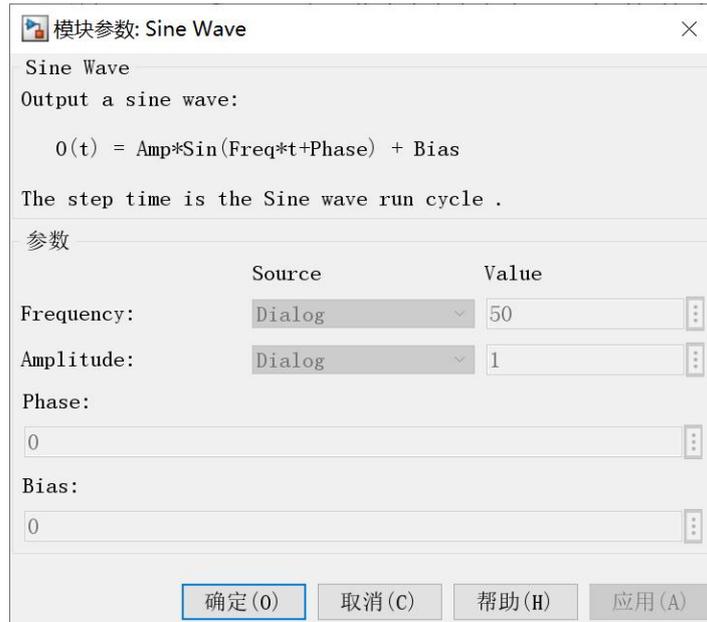


图 5.6 Sine Wave 模块

Sine Wave 模块参数说明如下表 5.3 所示。

表 5.3 Sine Wave 模块参数说明

项目	名称	参数说明	数据类型
Frequency	频率	正弦波频率设置, Input——由 Variable 输入, 在 RTUS 中可在线调节; Dialog——设置为固定值	-
Amplitude	幅值	正弦波幅值设置, Input——由 Variable 输入, 在 RTUS 中可在线调节; Dialog——设置为固定值	-
Phase	相位	正弦波相位设置	-
Bias	偏置	正弦波偏置设置	-
Output	输出	输出值	Single

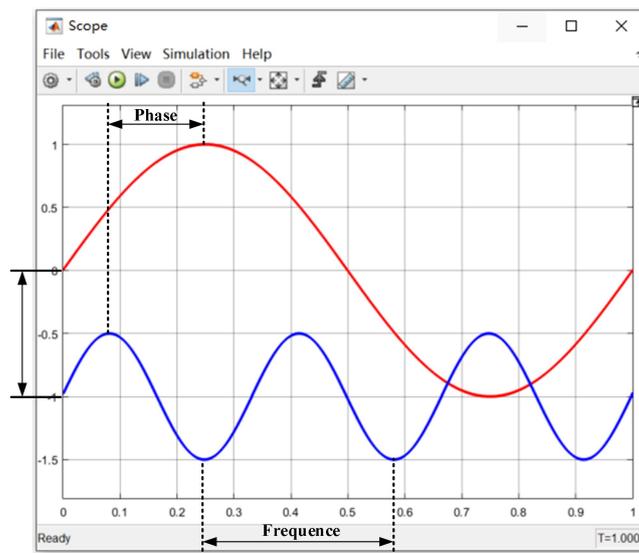


图 5.7 Sine Wave 模块 Simulink 仿真

5.2.4. Tcp2Sim 模块

Tcp2Sim 模块用于 TCP/IP 通讯，双击 Tcp2Sim 模块可进入模块参数设置界面，如图 5.8 所示。

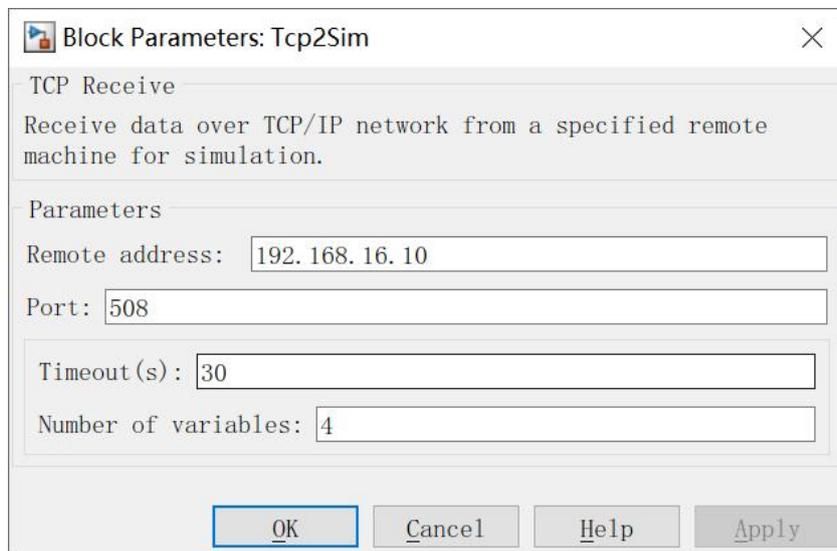
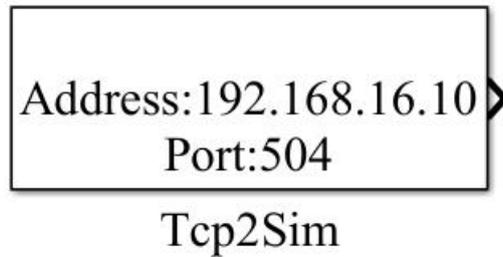


图 5.8 Tcp2Sim 模块

Tcp2Sim 模块参数说明如下表 5.4 所示。

表 5.4 Tcp2Sim 模块参数说明

项目	名称	参数说明	数据类型
Remote address	远程地址	192.168.16.10 (设备地址)	-
Port	端口	508-TCP/IP 以太网通讯	-
Timeout	暂停	超时时间 (秒)	-
Number of variables	变量数	输入的变量数目	-
Output	输出	输出值	Single

5.3. Power Electronics

5.3.1. SVPWM 模块 1

SVPWM 模块 1 的功能是在三相对称正弦波电压供电情况下，三相对称电动机定子以理想磁链圆为参考标准，通过三相逆变器不同开关模式作适当的切换，从而形成 PWM。双击模块可以进入模块介绍界面，如图 5.9 所示。

需要注意的是这里输出的是占空比，用来与 PWM(DSP PWM/FPGA PWM)模块进行调制生成 PWM。

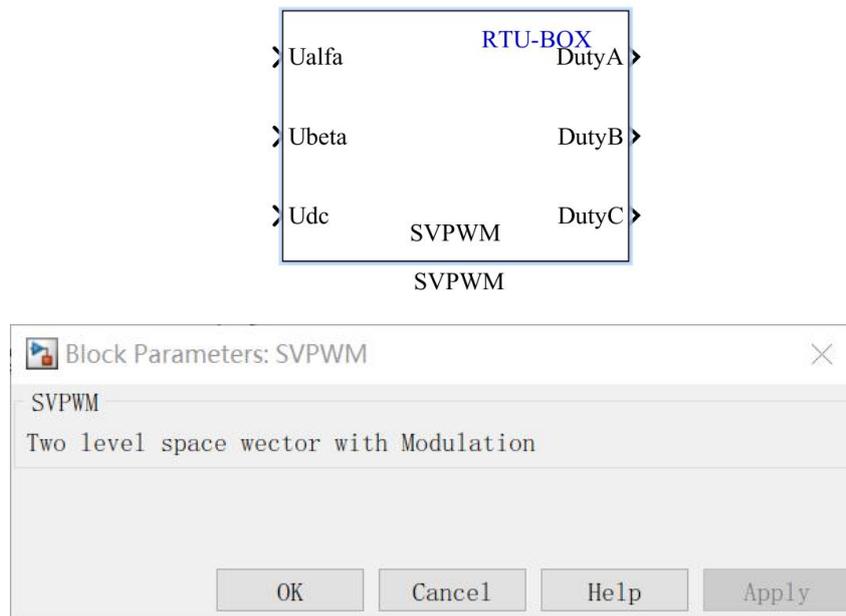


图 5.9 SVPWM 模块

SVPWM 模块 1 参数说明如下表 5.5 所示。

表 5.5 SVPWM 模块 1 参数说明

项目	名称	参数说明	数据类型
U α	输入	α 轴电压	Single
U β	输入	β 轴电压	Single
Udc	输入	直流母线电压	Single
DutyA	输出	输出 A 相脉冲(0-1)	Single
DutyB	输出	输出 B 相脉冲(0-1)	Single
DutyC	输出	输出 C 相脉冲(0-1)	Single

5.3.2. SVPWM 模块 2

SVPWM 模块的功能是在三相对称正弦波电压供电情况下,三相对称电动机定子以理想磁链圆为参考标准,通过三相逆变器不同开关模式作适当的切换,从而形成 PWM。双击模块可以进入模块介绍界面,如图 5.10 所示。

需要注意的是这里输出的是占空比,用来与 PWM(DSP PWM/FPGA PWM)模块进行调制生成 PWM。

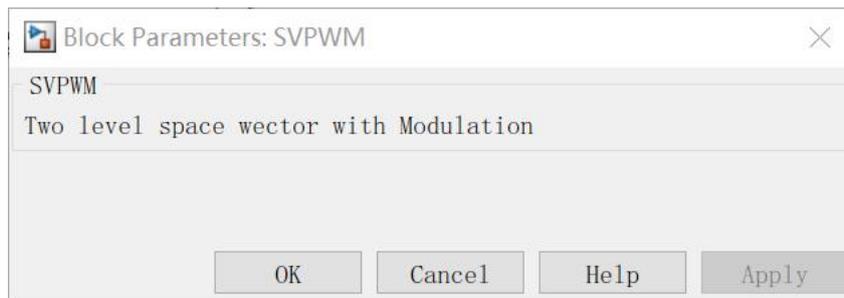
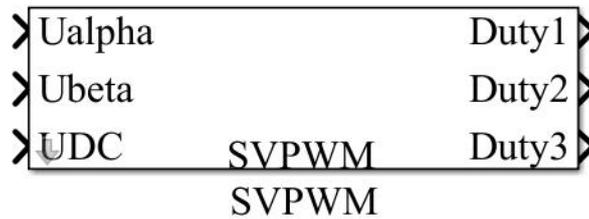


图 5.10 SVPWM 模块

表 5.6 SVPWM 模块 2 参数说明

项目	名称	参数说明	数据类型
U α	输入	α 轴电压	Single
U β	输入	β 轴电压	Single
UDC	输入	直流母线电压	Single
Duty1	输出	输出 A 相脉冲(0-1)	Single
Duty2	输出	输出 B 相脉冲(0-1)	Single
Duty3	输出	输出 C 相脉冲(0-1)	Single

注: 5.3.1 节和 5.3.2 节中的 SVPWM 模块功能一致。5.3.2 中的 SVPWM 模块是通过 Matlab/Simulink 模型搭建的,5.3.1 中的 SVPWM 模块是由 C 语言封装的。5.3.1 中的 SVPWM 模块生成代码更加简洁。

5.4. RTU BOX

5.4.1. Absolute Rotary Encoder 模块

Absolute Rotary Encoder 模块的功能是读取绝对值编码器的单圈和多圈位置。绝对值编码器可以记录编码器在一个绝对坐标系上的位置，每一个位置对应一个确定的数字码。因此绝对值编码器的示值只与测量的起始和终止位置有关，而与测量的中间过程无关。绝对值编码器由机械位置决定的每个位置的唯一性，它无需记忆，无需找参考点，而且不用一直计数。绝对值编码器无需开机找零，抗干扰特性、数据的可靠性大大提高了。另外，绝对值编码器无需判定方向、累计计数，可直接读数，其响应也较增量的快。双击模块可以进入模块介绍界面，如图 5.11 所示。

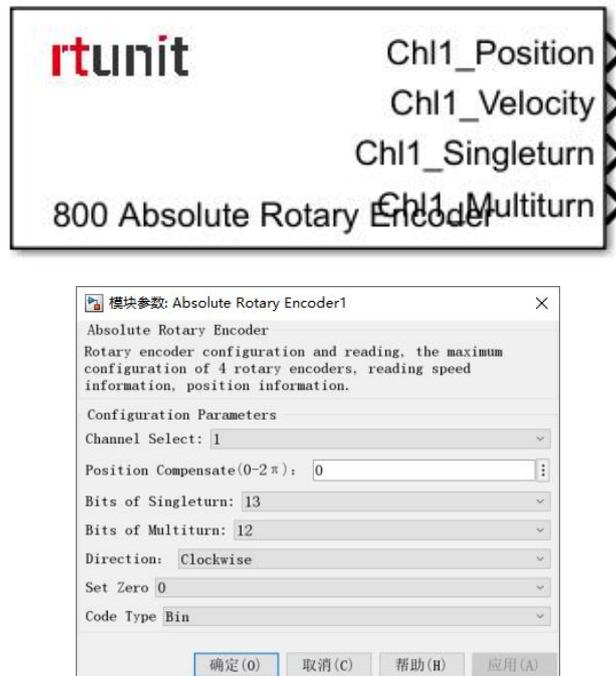


图 5.11 Absolute Rotary Encoder 模块

Absolute Rotary Encoder 模块参数说明如下表 5.7 所示。

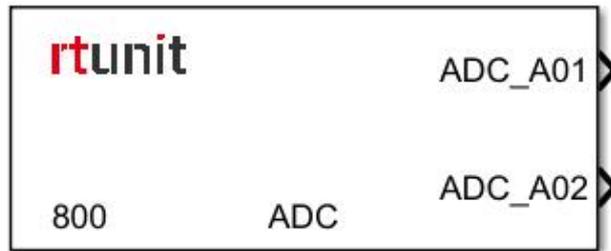
表 5.7 Absolute Rotary Encoder 模块参数说明

项目	名称	参数说明	数据类型
Channel Select	通道选择	设备地址 (1-4)	-
Position Compensate	位置补偿	电机机械零位置和编码器零位置差值 (0-2 π)	-
Bits of Singleturn	单圈位数	编码器单圈位数 (0-32)	-
Bits of Multiturn	多圈位数	编码器多圈位数 (0-32)	-
Direction	参考正方向	顺时针 clockwise; 逆时针 anticlockwise	
Set Zero	初始位置设置	1: 设置初始位置设置; 0: 默认初始位置	
Code Type	编码方式	Bin 二进制编码; Gray 格雷码编码	
Position	位置	机械位置弧度 (0-2 π)	Single
Velocity	速度	机械速度 rpm	Single
Singleturn	单圈位置	单圈位置计数 (0~2 ^{^(Bits of Singleturn)})	Uint32
Multiturn	多圈位置	多圈位置计数 (0~2 ^{^(Bits of Multiturn)})	Uint32

Absolute Rotary Encoder 模块可以匹配 SSI、BISS_C、多摩川、Endata2.2, 由于绝对值编码器类型比较多, 板卡默认 4 个通道为 SSI 编码器。

5.4.2. ADC 模块

ADC 模块的功能是通过采样将模拟量转换为数字量, 双击 ADC 模块可进入模块参数配置界面, 如图 5.12 所示。



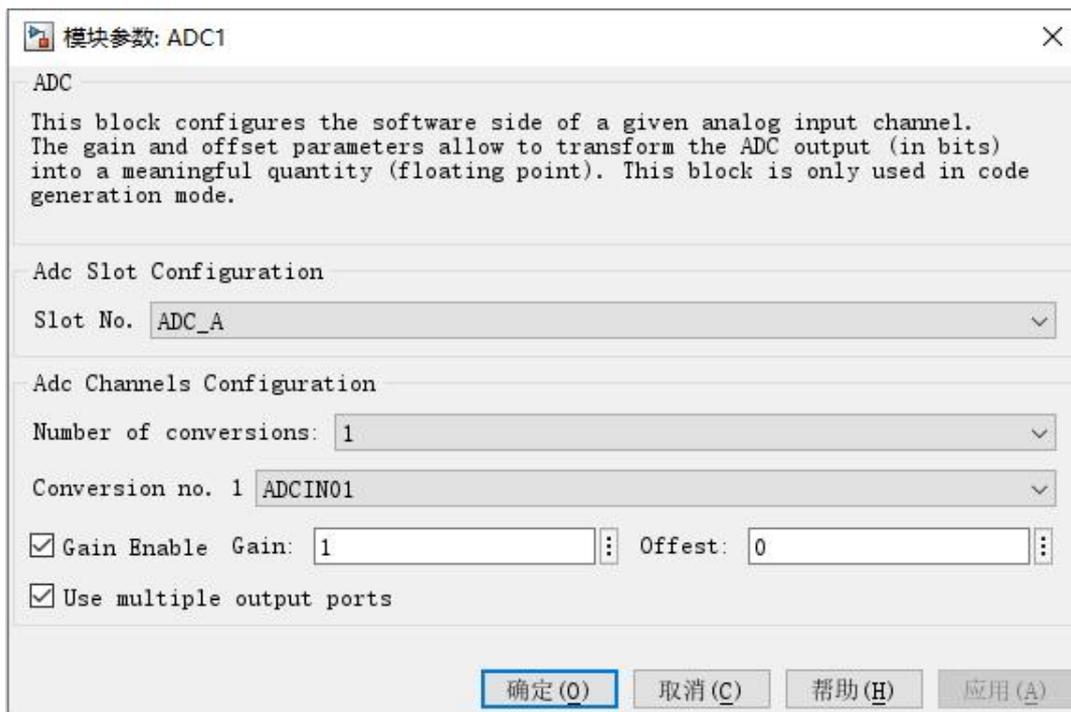


图 5.12 ADC 模块

ADC 模块参数说明如下表 5.8 所示。

表 5.8 ADC 模块参数说明

项目	名称	参数说明	数据类型
Slot No	板卡选择	根据实际情况选择 ADC 板卡，共 2 块：A、B	-
Number of conversions	采样通道数选择	根据实际情况选择使用的 ADC 通道总数，共 16 通道	-
Conversions no.	具体的采样通道	选择实际的采样通道	-
Gain Enable	增益使能	增益功能使能，可配置放大比例和零点偏置	-
Use multiple output ports	多通道输出端口	使能：使用多通道端口输出； 禁止：使用数组端口输出	-
ADC_AN	adc 采样输出	adc 采样输出实时值	Single

说明 1:

Gain Enable 如果不勾选，则 ADC 模块输出的结果为对应硬件端口的实际电压值（范围 -10V~10V）。为了将采样信号与实际信号的值相对应，并且消除外部电路的零漂，可以将 Gain Enable 勾选，则可使能增益和偏置校正功能。Gain 为增益系数，默认为 1；Offset 为偏移量，默认为 0。

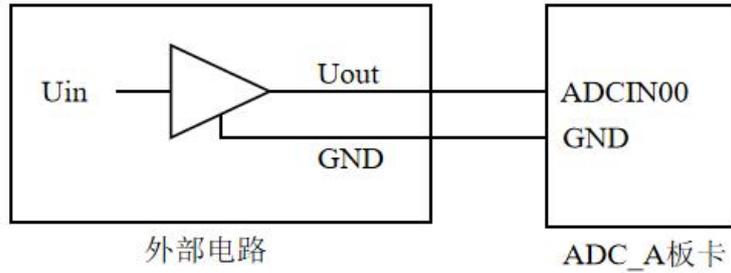


图 5.13 ADC_A 采样电路

参数校正举例：如图 5.13 所示，外部有一霍尔电压传感器采样电路，经过调理后， $U_{in}:U_{out}=100V:1V$ （即实际电压为 100V 对应调理电路输出为 1V， $K=1V/100V=0.01$ ），并将输出 U_{out} 信号接入到 ADC_A 板卡的 ADCIN00 引脚。将 Gain 设置为 1，当外部信号 U_{in} 为 0V 时， U_{out} 理论上也应为 0V。但由于电路中存在零漂，实际输出结果一般不为 0V，这时可以通过 Variable 模块读取 ADCIN00 通道的零漂输出值，假设为 0.1V，则 Offset 应设为 -0.1。

然后调节 U_{in} 的值为 100V，此时 U_{out} 为 1.1V（0.1V 为前述的电路固有零漂电压），经过设置 Offset 校正后 Variable 模块读取 ADCIN00 通道的值应为 $1.1V-0.1V=1V$ ，因此可以得到：

$$Gain=100V / (100V \times 0.01) = 100$$

经过上述设置后，ADC 模块输出的值

说明 2:

Use multiple output ports 如果不勾选，在配置了多通道的情况下，需要配合 Demux 一起使用，如下图所示：

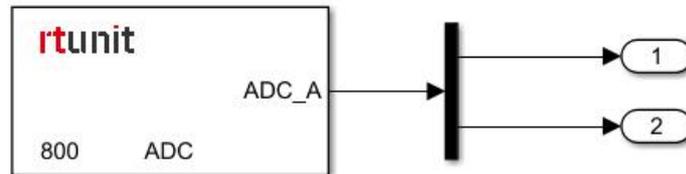


图 5.14 不勾选 Use multiple output ports

Use multiple output ports 如果勾选，在配置了多通道的情况下，则可分别显示，如下图所示：



图 5.15 ADC 通道与实际物理通道对应关系

5.4.3. DAC 模块

DAC 模块用于将数字量转换为模拟量输出，与 DAC 面板对应，接口定义详见 2.5 节。DAC 模块共有 16 路输出口，双击 DAC 模块可进入模块参数配置界面，如图 5.16 所示。

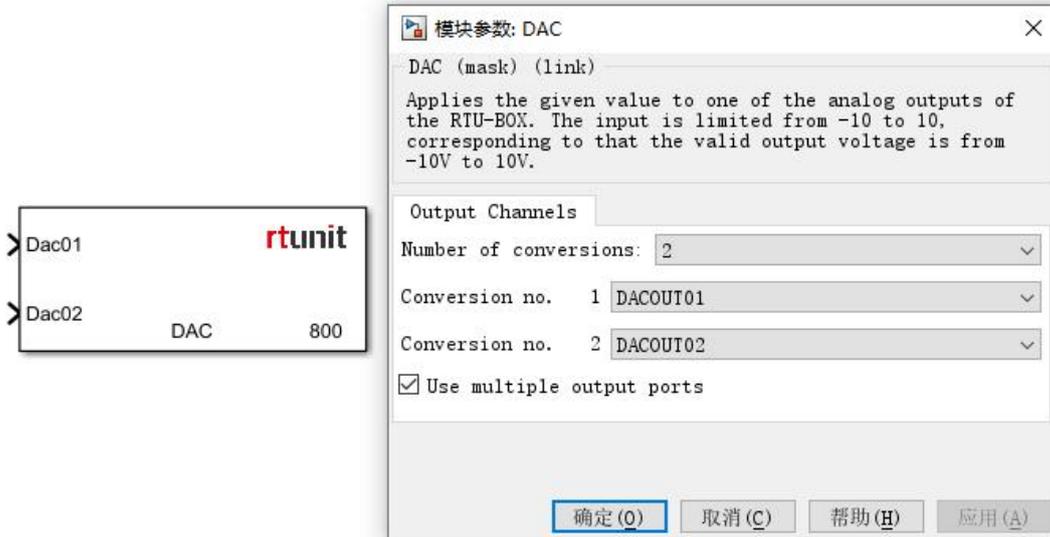


图 5.16 DAC 模块

DAC 模块参数说明如下表 5.9 所示。

表 5.9 DAC 模块参数说明

项目	名称	参数说明	数据类型
Number of Channels	通道总数选择	选择实际使用的 DAC 通道总数	-
Conversions no.	具体的通道	选择实际的转换通道	-
Use multiple input ports	多通道输入端口	使用多通道端口，详见说明 1	-
DacA	输入数据	DAC 通道输入范围-10~10，对应输出电压-10V~10V	Single

说明 1:

Use multiple input ports 如果不勾选，在配置了多通道的情况下，需要配合 Mux 一起使用，如下图所示：

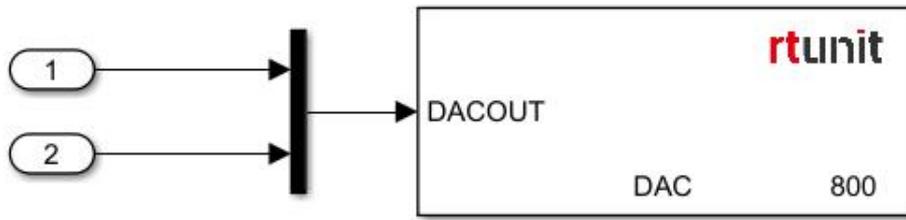


图 5.17 不勾选 Use multiple input ports

Use multiple input ports 如果勾选，在配置了多通道的情况下，则可分别显示，如下图所示：

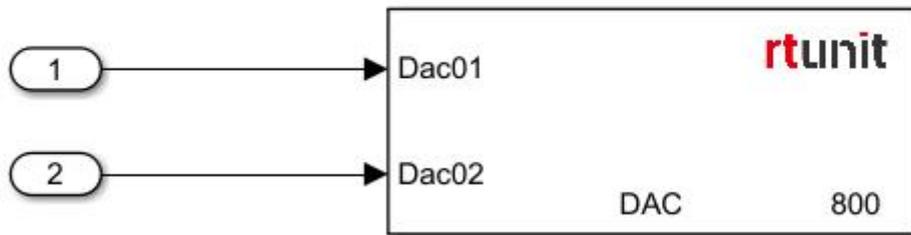


图 5.18 勾选 Use multiple input ports

5.4.4. DI 模块

DI 模块用于将电压信号转化为 boolean 型数据输入，输入电压信号不超过 5V，与 DI 板卡对应，接口定义详见 2.6 节。DI 模块共 6 个通道，双击 DI 模块可进入模块参数配置界面，如图 5.19 所示。

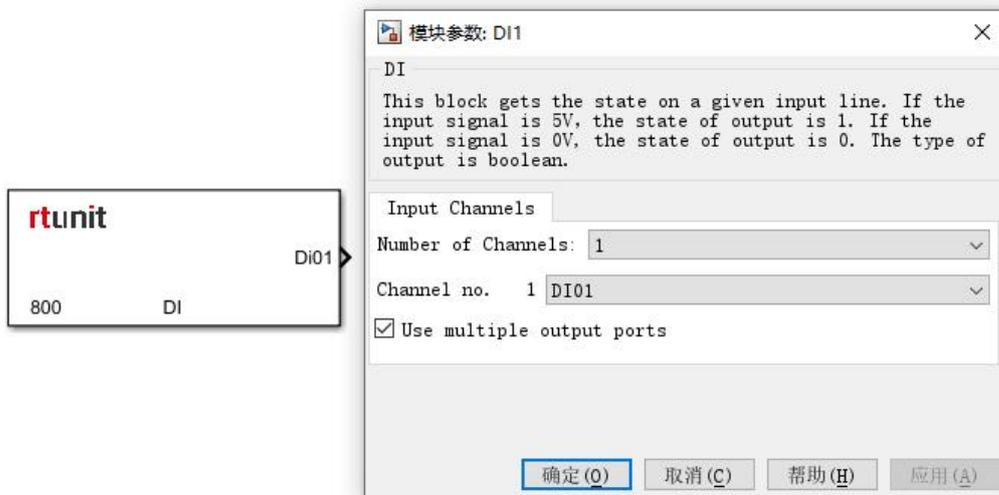


图 5.19 DI 模块

DI 模块参数说明如下表 5.10 所示。

表 5.10 DI 模块参数说明

项目	名称	参数说明	数据类型
Number of Channels	通道总数选择	选择实际使用的 DI 通道总数	-
Channel no.	具体的通道	选择实际的转换通道	-
Use multiple output ports	多通道输出端口	使用多通道端口, 详见说明 1	-
DI	-	返回 0 或者 1	Boolean

说明 1:

Use multiple output ports 如果不勾选, 在配置了多通道的情况下, 需要配合 Demux 一起使用, 如下图所示:

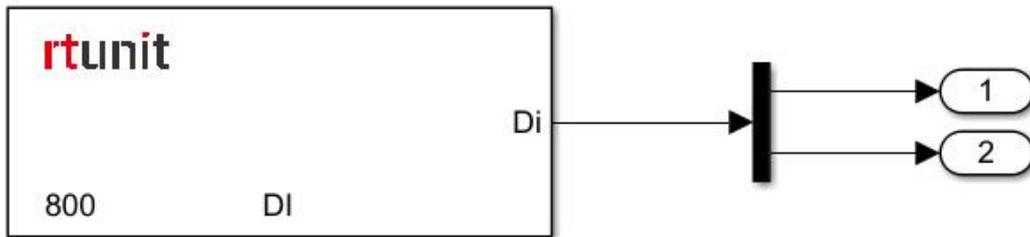


图 5.20 不勾选 Use multiple output ports

Use multiple output ports 如果勾选, 在配置了多通道的情况下, 则可分别显示, 如下图所示:

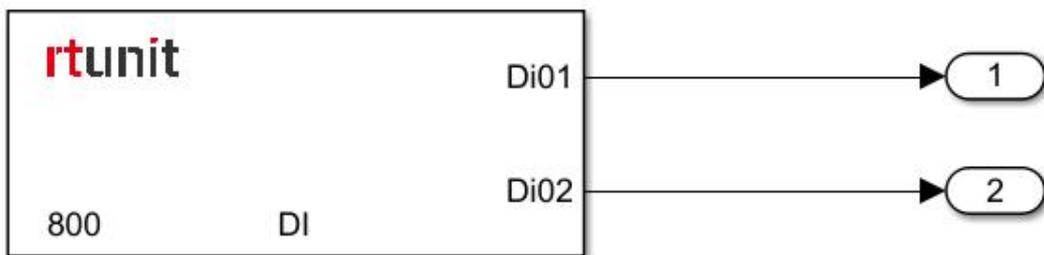


图 5.21 勾选 Use multiple output ports

5.4.5. DO 模块

DO 模块用于将 0/1 数字信号转换为 0/5V 电压信号输出，与 DO 板卡对应，接口定义详见 2.3 节。DO 模块共有 6 个通道，双击 DO 模块可进入模块参数配置界面，如图 5.22 所示。

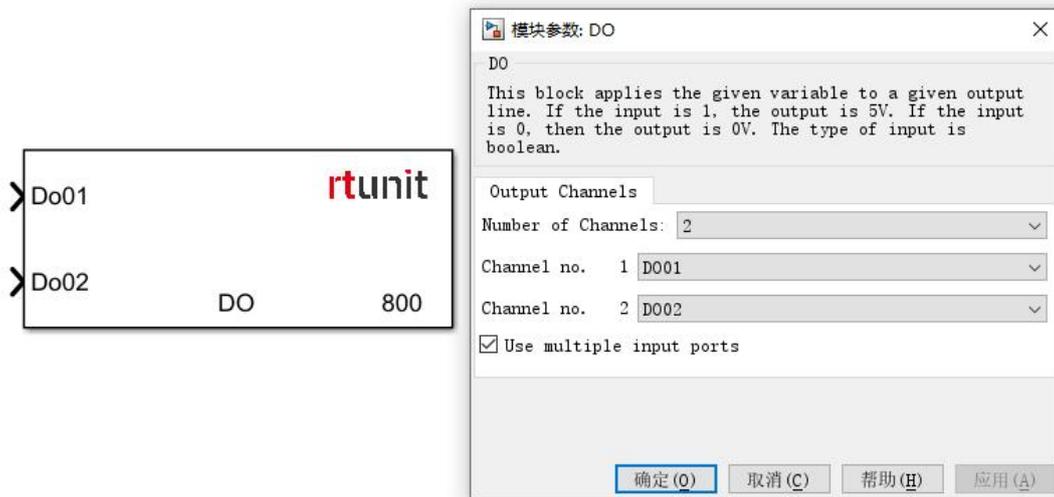


图 5.22 DO 模块

DO 模块参数说明如下表 5.11 所示。

表 5.11 DO 模块参数说明

项目	名称	参数说明	数据类型
Number of Channels	通道总数选择	选择实际使用的 DO 通道总数	-
Channel no.	具体的通道	选择实际的转换通道	-
Use multiple output ports	多通道输出端口	使用多通道端口，详见说明 1	-
DO	-	输入 0 或者 1	Boolean

说明 1:

Use multiple output ports 如果不勾选，在配置了多通道的情况下，需要配合 mux 一起使用，如下图所示：

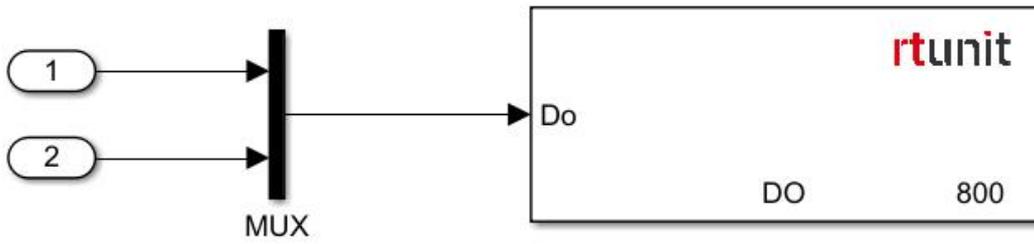


图 5.23 不勾选 Use multiple input ports

Use multiple input ports 如果勾选，在配置了多通道的情况下，则可分别显示，如下图所示：

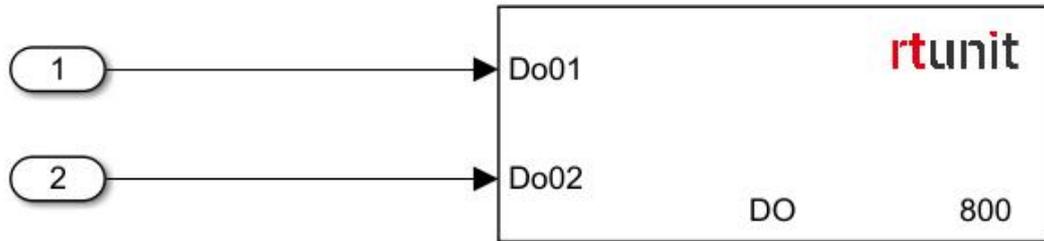


图 5.24 勾选 Use multiple input ports

5.4.6. eCAN Rx 模块

eCAN Rx 模块通过 CAN 通信接收其他设备或上位机发送信息的功能。双击 eCAN Rx 模块可进入模块参数设置界面，如图 5.25 所示。（RTU-BOX800 的 CAN 口在正面板上，如图 2.2 所示）

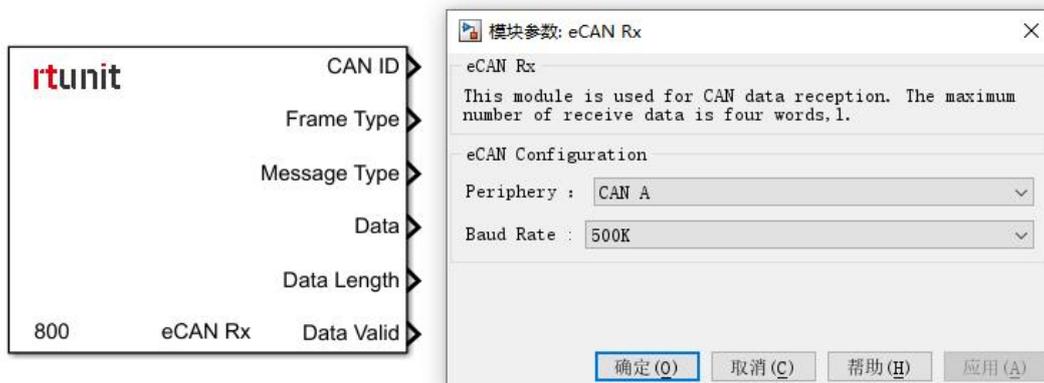


图 5.25 eCAN Rx 模块

eCAN Rx 模块参数说明如下表 5.12 所示。

表 5.12 eCAN Rx 模块参数说明

项目	名称	参数说明	数据类型
CAN ID	CAN ID	外接变量输出 CAN ID, 标准帧 11 位, 扩展帧 29 位	Uint32
Frame Type	帧类型	Ture-标准帧 Flase 扩展帧-	Boolean
Message Type	邮箱类型	Ture-数据帧 Flase-远程帧	Boolean
Data	数据	接受数据的内容, 最多 8 个字节	-
Data Length	数据长度	接受数据的长度, 最多 8 个字节	Uint8
Data Valid	数据校验	1-数据接收完成, 0-数据未接收完成	-
Periphery	外设选择	有两种选择 CAN A 和 CAN B	-
Baud Rate	波特率选择	20K、50K、100K、125K、250K、500K、1000K	-

注意: 使用 CAN 模块通信之前请先确认 CAN 的参数设置, 使用 CAN 通信的两台设备相关参数的设置要一致。

5.4.7. eCAN Tx 模块

eCAN Tx 模块通过 CAN 总线向其他设备或上位机发送信息的功能。双击 eCAN Tx 模块可进入模块参数设置界面, 如图 5.26 所示。(RTU-BOX800 的 CAN 接口在正面板上, 如图 2.2 所示)

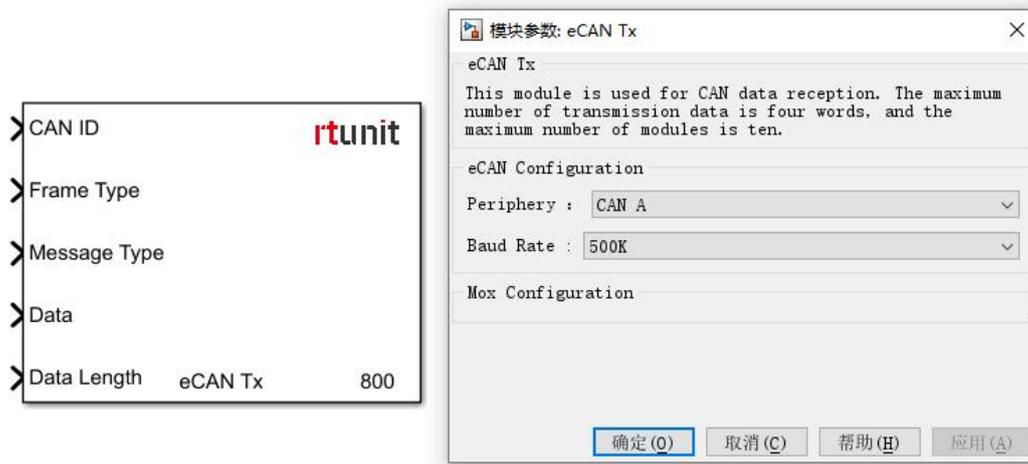


图 5.26 eCAN Tx 模块

eCAN Tx 模块参数说明如下表 5.13 所示。

表 5.13 eCAN Tx 模块参数说明

项目	名称	参数说明	数据类型
CAN ID	CAN ID	外接 Variable 模块输入 CAN ID, 标准帧 11 位, 扩展帧 29 位	Uint32
Frame Type	帧类型	Ture-标准帧 Flase 扩展帧-	Boolean
Message Type	邮箱类型	Ture-数据帧 Flase-远程帧	Boolean
Data	数据	接受数据的内容, 最多 8 个字节	Uint8
Data Length	数据长度	接受数据的长度, 最多 8 个字节	Uint8
Periphery	外设选择	有两种选择 CAN A 和 CAN B	-
Baud Rate	波特率选择	20K、50K、100K、125K、250K、500K、1000K	-

5.4.8. Ethernet Configuration 模块

Ethernet Configuration 模块用于配置 RTU-BOX800 网口地址等信息。双击 Ethernet Configuration 模块可进入模块参数设置界面, 如图 5.27 所示。

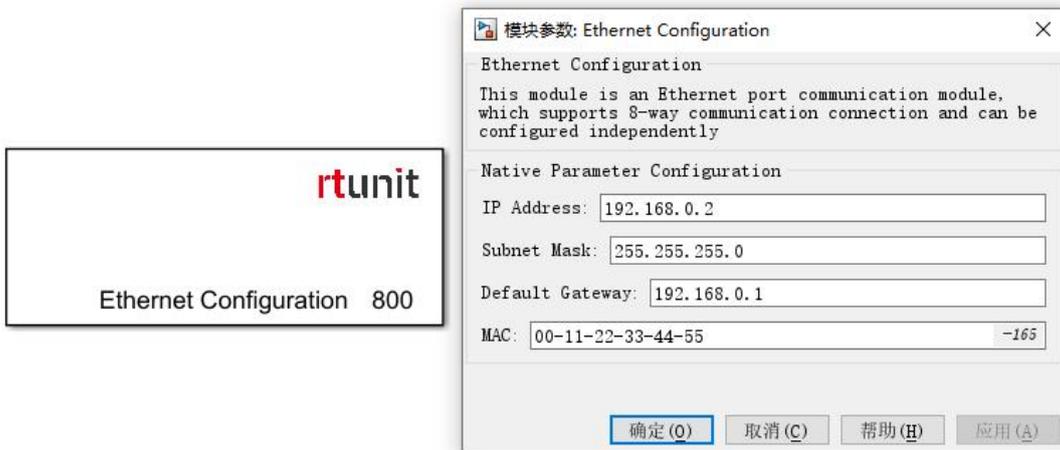


图 5.27 Ethernet Configuration 模块

Ethernet Configuration 模块参数说明如下表 5.14 所示。

表 5.14 Ethernet Configuration 模块参数说明

项目	名称	参数说明	数据类型
IP Address	IP 地址	IP 地址默认为: 192.168.16.10	-
Subnet Mask	掩码地址	掩码地址: 255.255.255.0	-
Default Gateway	默认网关	网关默认为 192.168.16.1	-
MAC	机器码	设备 MAC 地址	-

5.4.9. Ethernet Receive 模块

Ethernet Receive 模块用于实现 RTU-BOX800 通过网口接收其他设备或上位机发送信息的功能。双击 Ethernet Receive 模块可进入模块参数设置界面，如图 5.28 所示。

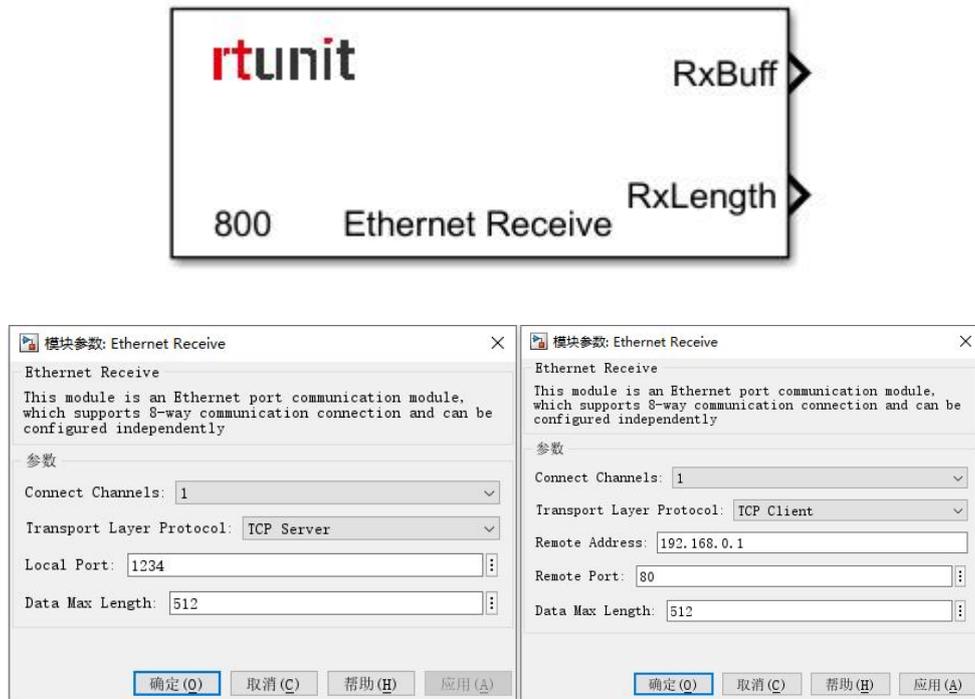


图 5.28 Ethernet Receive 模块

Ethernet Receive 模块参数说明如下表 5.15 所示。

表 5.15 Ethernet Receive 模块参数说明

项目	名称	参数说明	数据类型
Connect Channels	连接通道	通道: 0-7	-
Transport Layer Protocol	传输层协议	TCP Server:传输控制协议服务器 TCP Clinet:传输控制协议客户端 UDP: 用户数据报协议	-
Remote Address	远程地址	服务端的 IP 地址	-
Local Port	本地端口	本地端口号	-
Remote Port	远程端口	服务端端口号	-
Data Max Length	数据最大长度	512	-

5.4.10. Ethernet Send 模块

Ethernet Send 模块用于实现 RTU-BOX800 通过网口向其他设备或上位机发送信息的功能。双击 Ethernet Send 模块可进入模块参数设置界面，如图 5.29 所示。

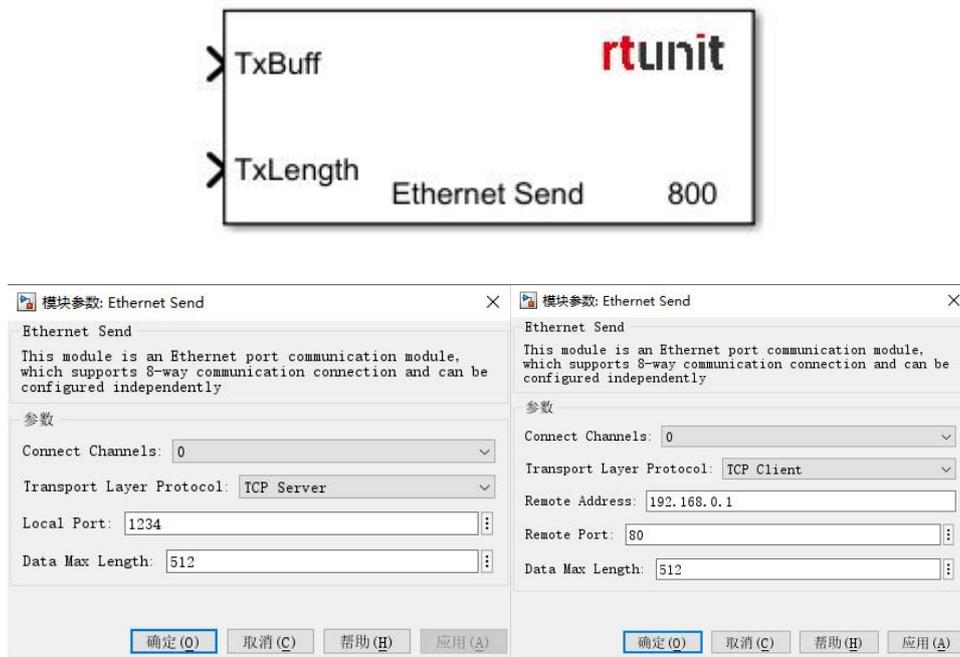


图 5.29 Ethernet Send 模块

Ethernet Send 模块参数说明如下表 5.16 所示。

表 5.16 Ethernet Send 模块参数说明

项目	名称	参数说明	数据类型
Connect Channels	连接通道	通道: 0-7	-
Transport Layer Protocol	传输层协议	TCP Server:传输控制协议服务器 TCP Clinet:传输控制协议客户端 UDP: 用户数据报协议	-
Remote Address	远程地址	服务端的 IP 地址	-
Local Port	本地端口	本地端口号	-
Remote Port	远程端口	服务端端口号	-
Data Max Length	数据最大长度	512	-

5.4.11. FPGA PWM 模块

FPGA PWM 模块的 PWM 由 FPGA 产生，双击 FPGA PWM 模块可进入模块参数设置界面，如图 5.30 所示。

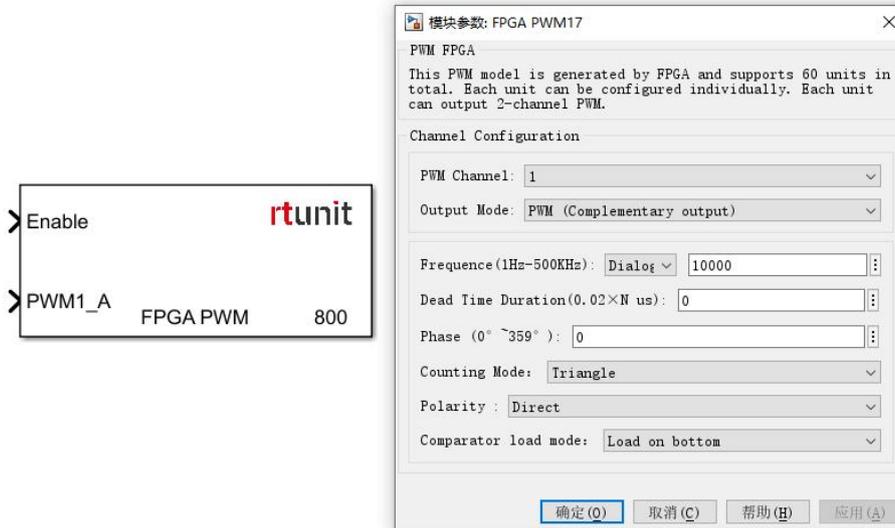


图 5.30 FPGA PWM 模块

FPGA PWM 模块参数说明如下表 5.17 所示。

表 5.17 FPGA PWM 模块参数说明

项目	名称	参数说明	数据类型
Pwm Channel	通道选择	PWM Channel PWM 通道选择使用的 PWM 通道，可选择 1—15 通道	-
Output Mode	输出模式	PWM(single output)-独立输出 PWM 信号 PWM(Complimentary output)-互补输出 PWM 信号 DO(Complimentary output)-互补输出 DO 信号 DO(single output)-独立输出 DO 信号 PWM(Complimentary User define)-互补 Free PWM 信号	-
Frequence	频率设置	频率设置，范围 1Hz~500KHz	-
Dead time Duration	死区时间	死区时间，单位 us	-
Phase	相位	相位参数，可设置范围为 0~359	-
Counting Mode	计数模式	计数模式有 Triangle、Invtriangle，详见说明 3	-
Polarity	极性	输出极性。Direct: 直接输出；Inverse: 反相输出	-
Comparator load mode	比较器加载模式	Load on bottom-底部装载 Load on peak-顶部装载 Load on peak or bottom-顶部或者底部装载	-

说明 1:

FPGA PWM 模块有三种输出模式: PWM 模式、DO 模式以及 Free PWM 模式, PWM 模式根据给定的占空比由 FPGA 发出 PWM; DO 模式是根据给定的输入指令输出 0 或者 1 状态; Free PWM 模式根据给定的 PWM1_A、PWM1_B、PWM1_C、PWM1_D (四段占空比之和必须小于 1, 发波原理参考说明 5) 占空比由 FPGA 发出 PWM。若选择了 DO 模式, 则相关功能选择被禁用, 如图 5.31。

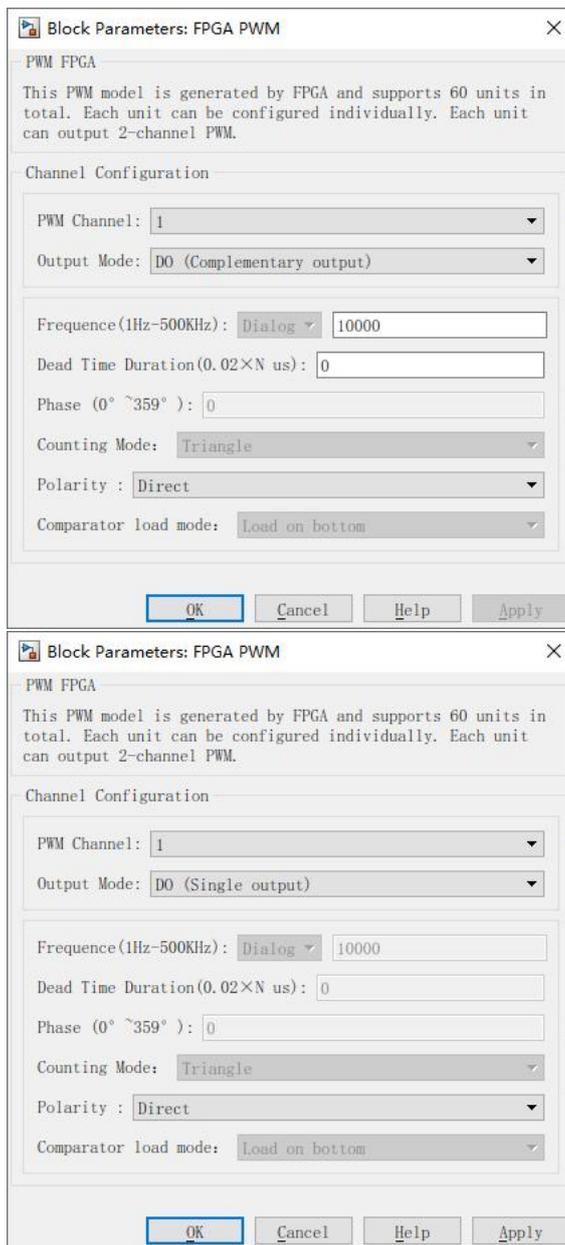


图 5.31 FPGA PWM 模块 DO 通道

说明 2:

RTU-BOX800 的 FPGA PWM 一共有 30 路, 最多输出 15 路互补的 PWM 信号。

说明 3:

FPGA PWM 模块支持两种载波形式: Triangle、Invtriangle, 分别如图 5.32、5.33 所示, 通过载波生成 PWM 的工作原理如图 5.34 所示。

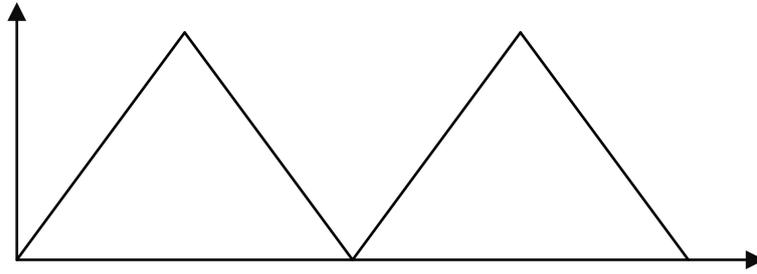


图 5.32 Triangle 载波形式

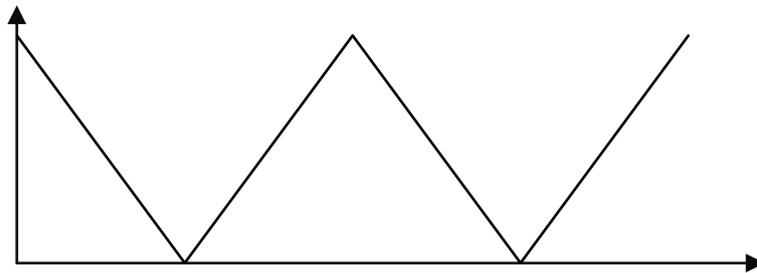


图 5.33 Invtriangle 载波形式

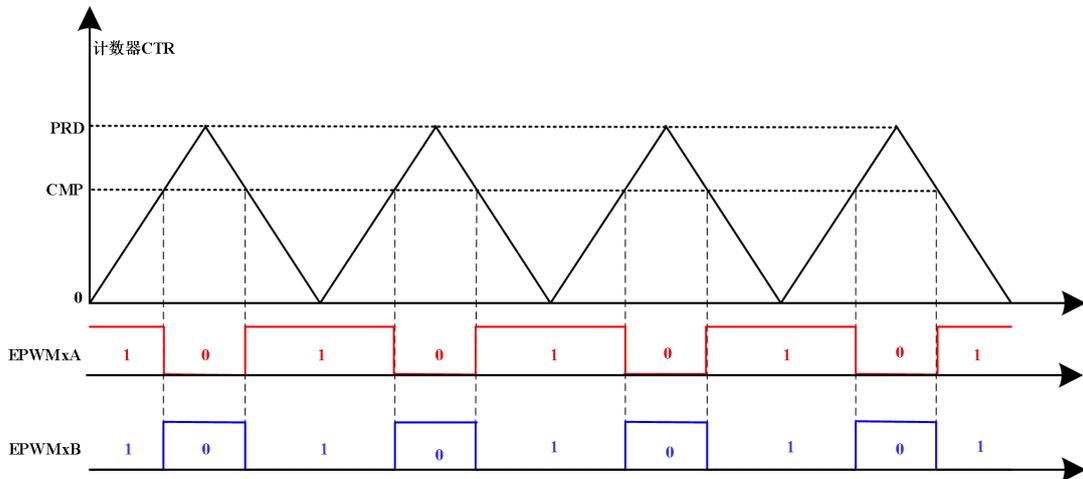


图 5.34 载波: Triangle, 输出模式: Complimentary Output 发波原理

说明 4:

您需要设置死区时间, 死区时间单位为 us。

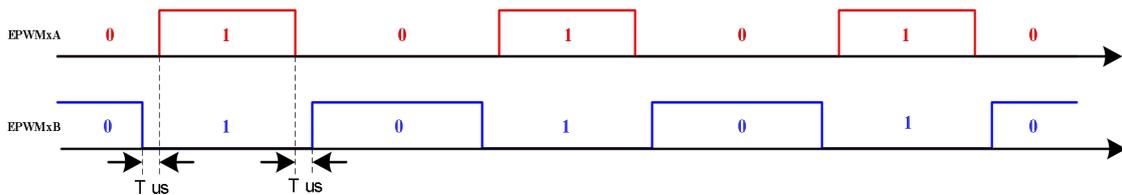


图 5.35 死区时间示意图

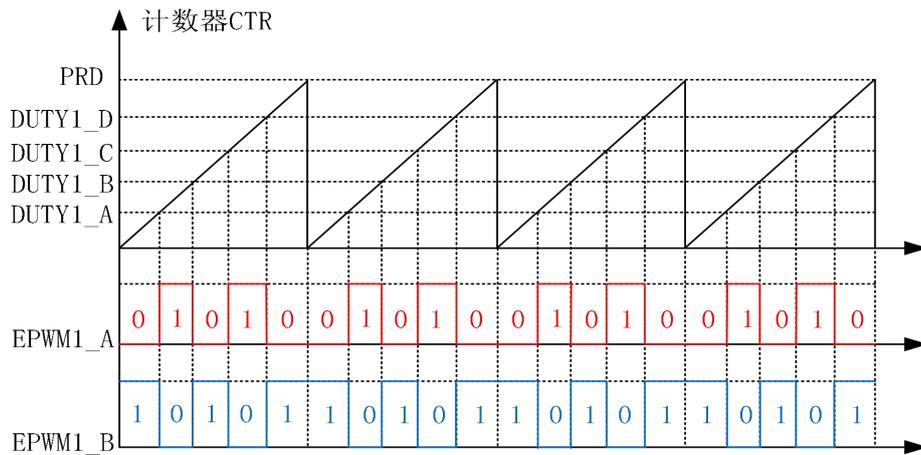


图 5.36 Free PWM 生成原理

说明 5:

FPGA PWM 模块在 Free PWM 模式下，载波模式固定为锯齿波。对应 Free PWM 生成原理如图 5.36 所示。

5.4.12. Hall Encoder 模块

Hall Encoder 模块用于解码 Hall 光电编码器，以获得电机转子的旋转方向、速度和位置信息，与 Hall Encoder 接口对应，接口定义详见 2.4 节。双击 Hall Encoder 模块可进入模块参数设置界面，如图 5.37 所示。

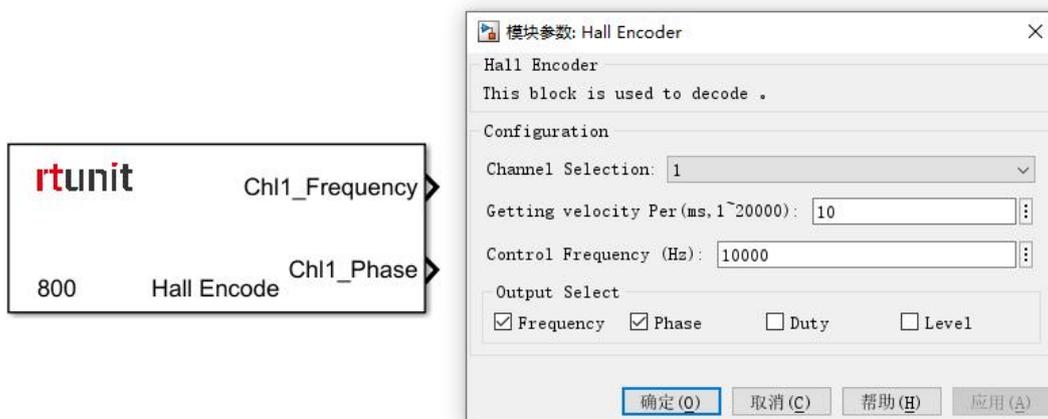


图 5.37 Hall Encoder 模块

Hall Encoder 模块参数说明如下表 5.18 所示。

表 5.18 Hall Encoder 模块参数说明

项目	名称	参数说明	数据类型
Channel Selection	通道选择	1-通道 1 2-通道 2 3-通道 3	-
Getting velocity Per	速度计算周期	默认 10ms	-
Control Frequency	系统控制频率	与系统控制频率同步, 默认 10000Hz	-
Output Select	输出选项	可选择脉冲频率、相位、占空比、电平状态输出	-

5.4.13. Incremental Encoder 模块

Incremental Encoder 模块用于解码正交光电编码器, 以获得电机转子的旋转方向、速度和位置信息, 与 Incremental Encoder 接口对应, 接口定义详见 2.4 节。双击 Incremental Encoder 模块可进入模块参数设置界面, 如图 5.38 所示。

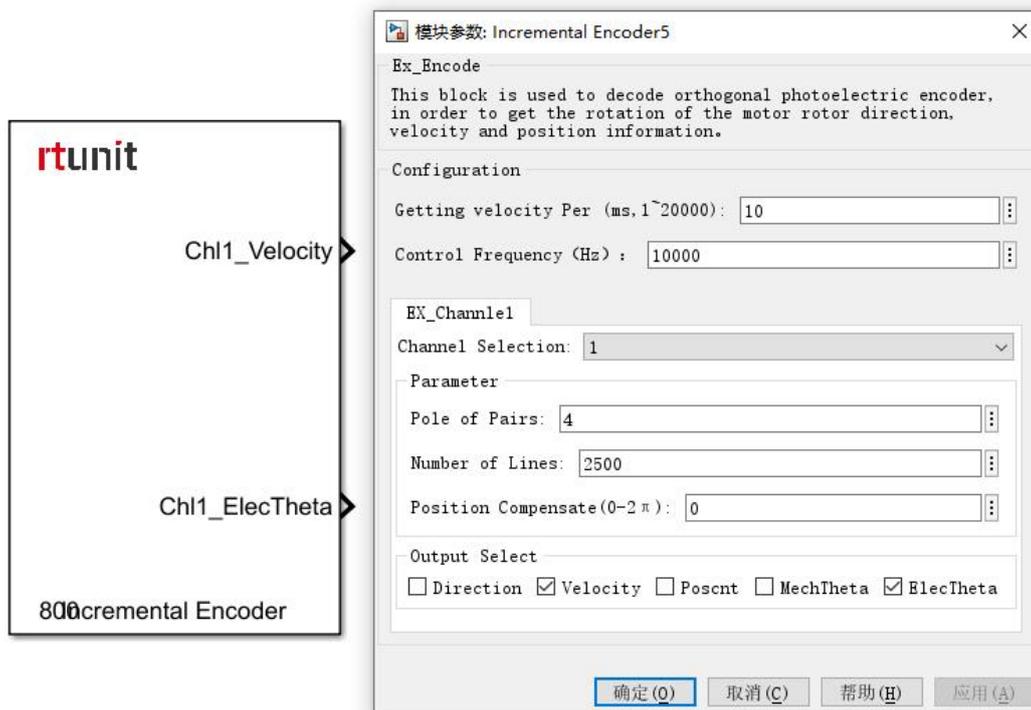


图 5.38 Incremental Encoder 模块

Incremental Encoder 模块参数说明如下表 5.19 所示。

表 5.19 Incremental Encoder 模块参数说明

项目	名称	参数说明	数据类型
Getting velocity Per	速度计算周期	默认 10ms	Uint16
Control Frequency	系统控制频率	与系统控制频率同步, 默认 10000Hz	Uint32
Channel Selection	通道选择	1-通道 1, 2-通道 2, 3-通道 3, 4-通道 4	-
Pole of Pairs	极对数	默认为 4	Uint16
Number of Lines	编码器线数	默认 2000 线	Uint16
Position Compensation	初始补偿位置	默认为 0, 范围[0, 2π]	Float

5.4.14. Main Loop Task 模块

Main Loop Task 模块用于实现执行定时任务, 生成的代码在主循环中执行, 需配合 Function-Call Subsystem 模块一起使用(该模型为 Simulink 工具箱自带), 双击 Main Loop Task 模块可进入模块参数设置界面, 如图 5.39 所示。

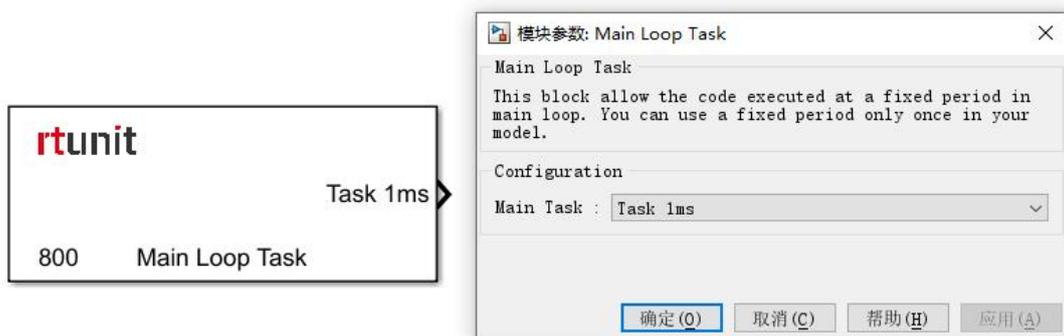


图 5.39 Main Loop Task 模块

Main Loop Task 模块参数说明如下表 5.20 所示。

表 5.20 Main Loop Task 模块参数说明

项目	名称	参数说明	数据类型
Main Task	任务时间	定时任务时长, 可配置 1ms/5ms/10ms/20ms/50ms/100ms/500ms/1000ms	-

说明 1:

Main Loop Task 模块可实现每隔一定时间循环执行一次代码的功能, 同一 simulink 模型中可以使用多个不同时间周期 Main Loop Task 模型, 如图 5.40 所示。

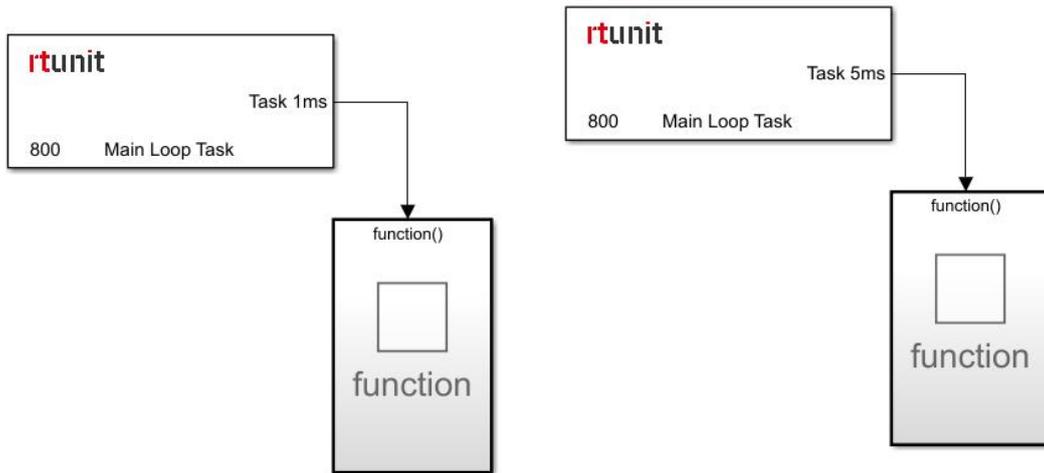


图 5.40 Main Loop Task 模块使用示例

说明 2:

Main Loop Task 模块调用 Function-Call Subsystem 模块，配置如图 5.41 所示，采样时间类型是周期性，采样时间为-1，继承上一层的调用周期。

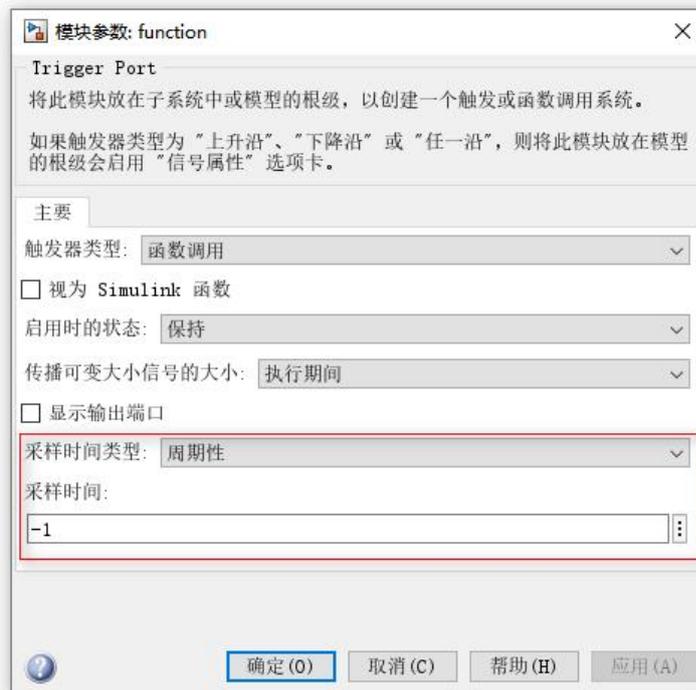
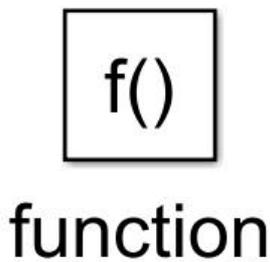


图 5.41 Function-Call Subsystem 模块设置示例

5.4.15. TIMING 模块

TIMING 模块用于测量所包含代码在硬件控制器中运行的时间，模块介绍如图 5.42 所示。

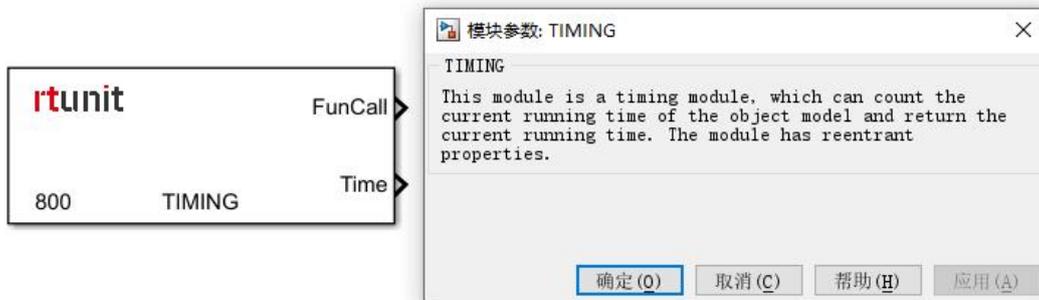


图 5.42 TIMING 模块

TIMING 模块没有参数需要配置，但其需要和 Function-Call Subsystem（该模型为 Simulink 工具箱自带）以及 Variable 模块一起使用，搭建模型如图 5.43 所示，用户需要将待测试的模型放入 Function-Call Subsystem 模块中，代码运行时长通过 TIMING 模块的 Time 口输出，单位为 us，可通过 Variable 模块在 RTUS 中观察。

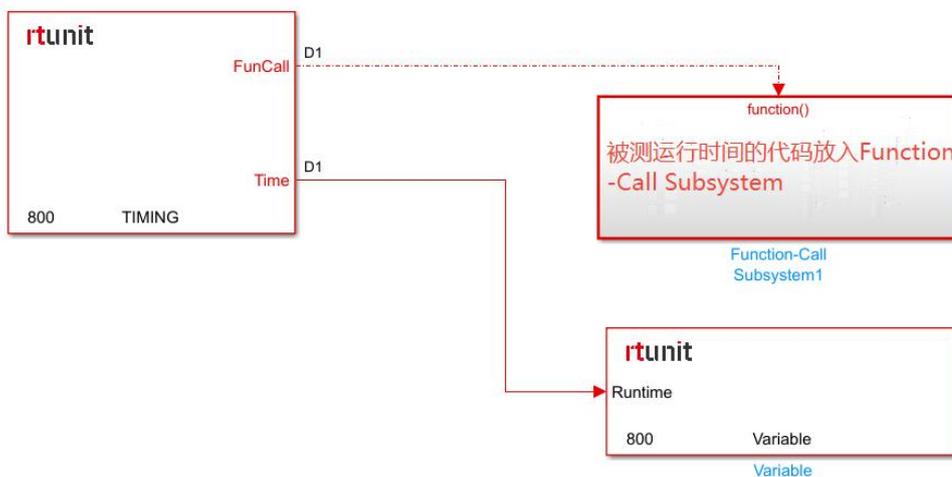


图 5.43 Meter Coder Time 模块使用示例

说明 1:

Meter Coder Time 模块可以只测部分程序，例如只测控制算法的运行时间，可搭建模型如图 5.44 所示。

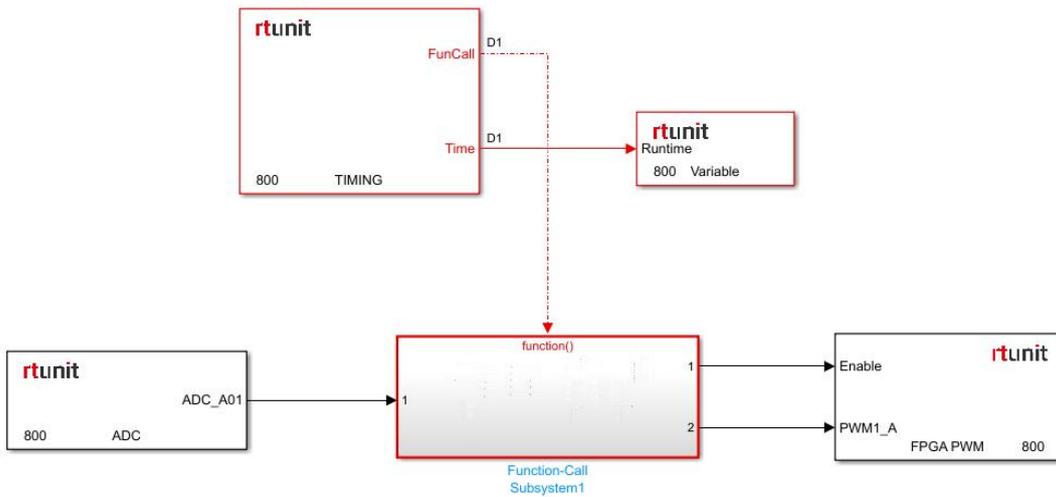


图 5.44 Meter Coder Time 模块使用示例

5.4.16. Rotary Encoder 模块

旋转编码器 (rotary encoder) 也称为轴编码器, 是将旋转的机械位移量转换为电气信号, 对该信号进行处理后检测位置速度等信号的传感器。Rotary Encoder 模块用于解码该类型编码器, 以获得电机转子的旋转方向、速度和位置信息, 与 Rotary Encoder 接口对应, 接口定义详见 2.7 节。双击 Rotary Encoder 模块可进入模块参数设置界面, 如图 5.45 所示

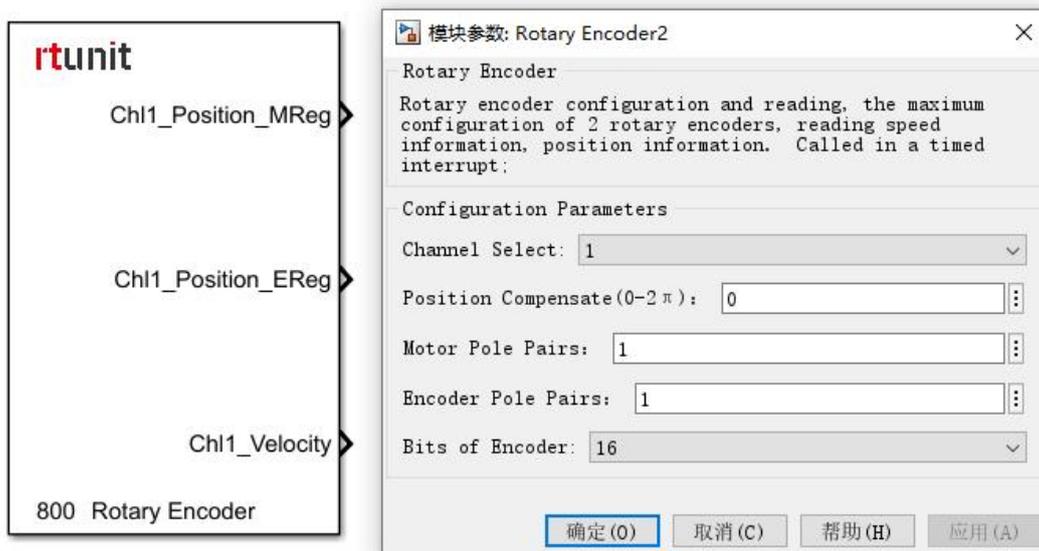


图 5.45 Rotary Encoder 模块

Rotary Encoder 模块参数说明如下表 5.21 所示。

表 5.21 Rotary Encoder 模块参数说明

项目	名称	参数说明	数据类型
Channel Select	通道选择	本设备有 2 个 Rotary 编码器接口, 默认为 1	-
Position Compensate	位置补偿	电机机械零位置和编码器零位置插值 (0-2π)	-
Motor Pole Pairs	电机极对数	默认值 1	-
Encoder Pole Pairs	编码器极对数	默认值 1	-
Bits of Encoder	编码器位数	默认 16 位	-

说明 1:

转子在物理上转过一个周期对应的角度 θ 为机械角度, 当极对数部位 1 时, 一个正余弦周期内转过的角度换算成 0 到 2π 的角度 ϕ 称为电气角度。下面是几种电机和其编码器的角度寻找方法如下:

- 1) 旋变 1 对极的编码器电机 P 对极的情况下, 那么旋变输出的就是:

机械角度 θ_M ,

电气角度 $\theta_E = \theta_M * P$,

速度 Speed 就是机械速度 $Speed_M$,

电气速度 $Speed_E = Speed_M * P$;

- 2) 旋变 P 对极的编码器电机 P 对极的情况下, 那么旋变输出的就是电气角度 θ_E ,

因为电机的电气位置和旋变的机械位置是一一对应的;

输出的是电气速度 $Speed_E$,

机械速度 $Speed_M = Speed_E / P$;

注意: 此情形下, 当旋变极对数较大时, 速度测量范围会变小, 如果实际转速范围较大, 需要适当降低编码器精度以扩大测量范围, 具体示例如下。

旋变 10 对极的编码器电机 10 对极的情况下:

编码器位数为 16bit 时, 编码器所能测量的最大机械速度

$$Speed_M = 32768 * 0.229 / 10 = 750;$$

编码器位数为 14bit 时, 编码器所能测量的最大机械速度

$$Speed_M = 32768 * 0.229 / 10 * 2 = 1500;$$

- 3) 旋变 m 对极的编码器, 电机 n 对极的情况下, $n = k * m$ (电机对极是旋变编码器的整数倍, k 为整数)。假设此时读出的位置信号为 θ , 此时输出的:

电气角度 $\theta_E = k * \theta$,

机械角度是 θ / m 或 (θ_E / n) ,

模块的算法逻辑是 0-2PI 之间的换算, 不取余; 假定读出的速度为 speed,

电气速度为 $k * Speed$,

机械速度为 $k \cdot \text{Speed}/n$ 或 Speed/m ;

4) 旋变 m 对极的编码器,电机 n 对极的情况下, $m=k \cdot n$, (不建议这么使用)。

5.4.17. SCI Receive 模块

SCI Receive 模块用于实现 SCI 通信串口,接收其他设备或上位机发送信息的功能。双击 SCI Receive 模块可进入模块参数设置界面,如图 5.46 所示。

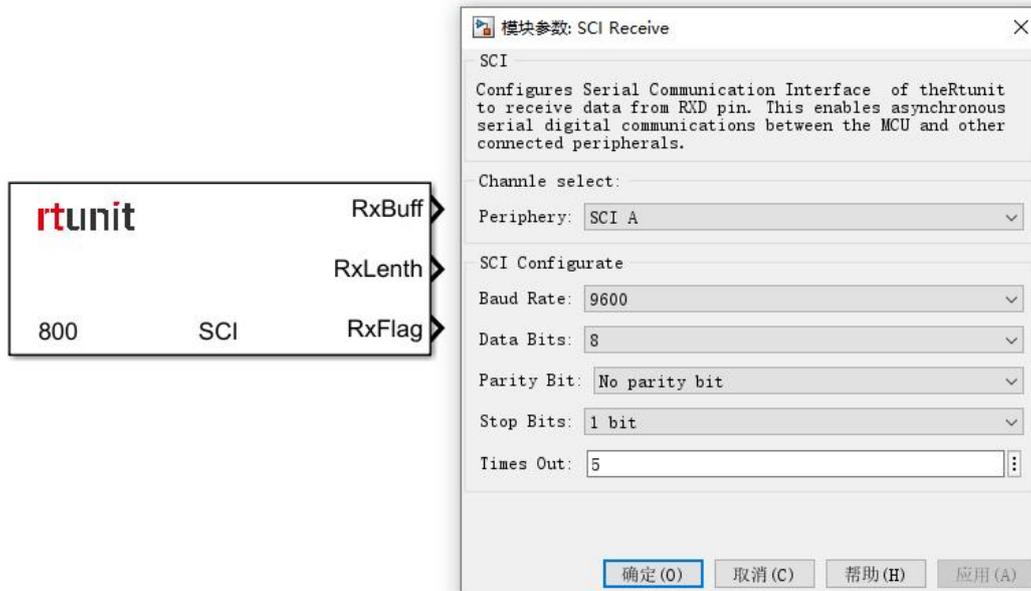


图 5.46 SCI Receive 模块

SCI Receive 模块参数说明如下表 5.22 所示。

表 5.22 SCI Receive 模块参数说明

项目	名称	参数说明	数据类型
Data	数据	接受数据, 长度 255 个字节	-
Length	数据长度	实际接受数据长度	-
RxFlag	接受标志位	1: 数据接受完成; 0: 没有接收到数据	boolean
Periphery	外设选择	有 2 种选择 SCIA、SCIB	-
Baud Rate	波特率选择	支持 2400, 4800, 7200, 9600, 14400, 19200, 38400, 57600, 115200, 128000	-
Data Bits	数据长度	数据长度, 支持 8 字节	-
Parity Bit	校验位	校验位, 可选择奇/偶校验或无校验	-
Stop Bits	停止位	停止位, 可选择 1 位/2 位	-

以下为具体使用参考案例:

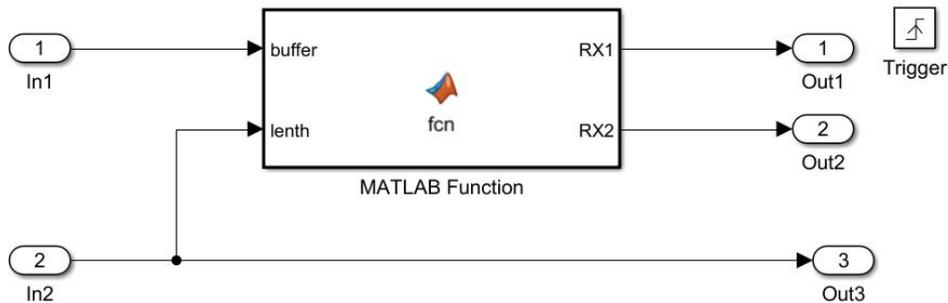
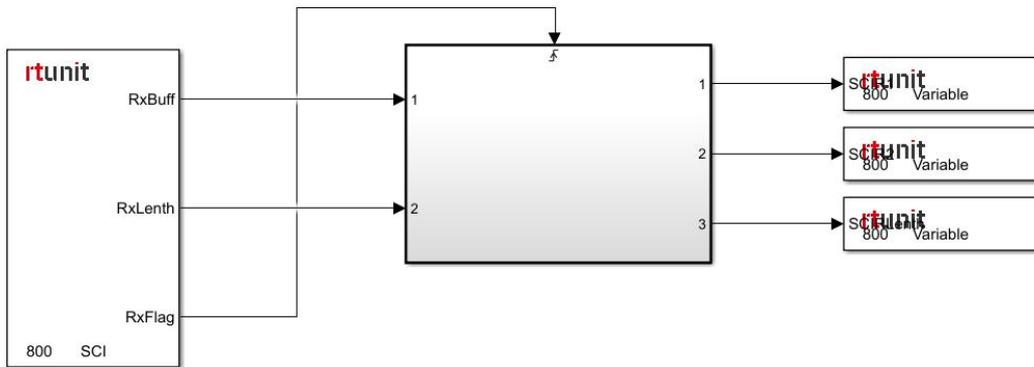


图 5.47 RS485 Receive 模型实例

```

function [RX1, RX2] = fcn(buffer, lenth)
    %对输出的变量进行单精度初始化
    RX1 = uint16(0); %服务请求的地址
    RX2 = uint16(0); %请求的服务

    %按照数据格式解析
    if lenth > 0 %判别是否为数据帧
        RX1 = buffer(4);
        RX2 = buffer(5);
    end
    
```

图 5.48 SCI Receive MATLAB Function 代码

5.4.18. SCI Transmit 模块

SCI Transmit 模块用于实现 RTU-BOX800 通过 SCI 通信串口向其他设备或上位机发送信息的功能。双击 SCI Transmit 模块可进入模块参数设置界面，如图 5.49 所示。(RTU-BOX800 的 SCI 通信串口在 CPU 板上，见 2.4 节)

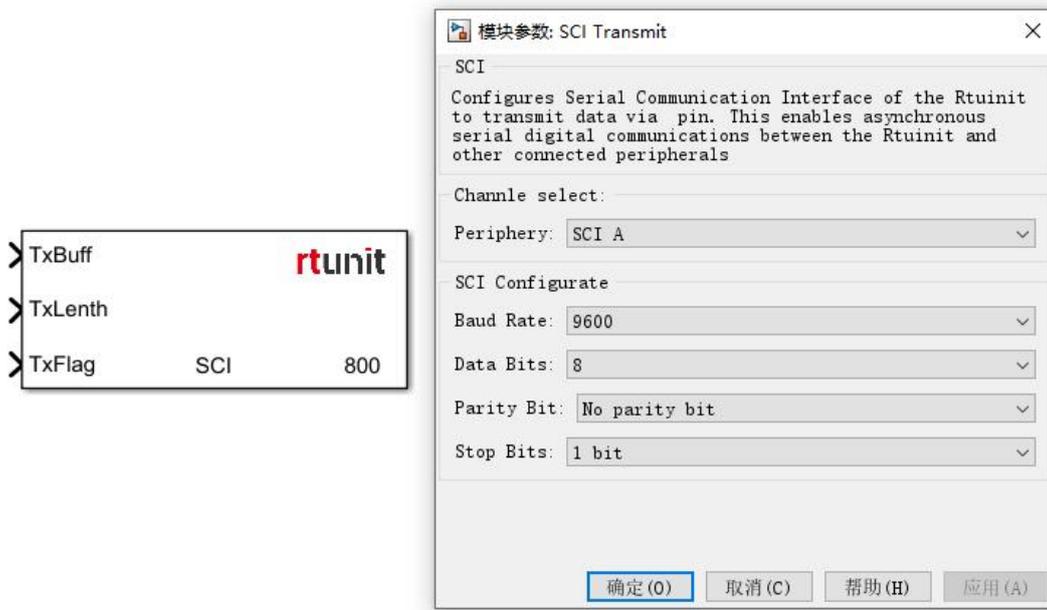


图 5.49 SCI Transmit 模块

SCI Transmit 模块参数说明如下表 5.23 所示。

表 5.23 SCI Transmit 模块参数说明

项目	名称	参数说明	数据类型
Data	数据	发送数据，长度 255 个字节	-
Length	数据长度	实际发送数据长度	-
TxFlag	发送标志位	1: 使能数据发送; 0: 禁止数据发送	boolean
Periphery	外设选择	有 2 种选择 SCIA、SCIB	-
Baud Rate	波特率选择	支持 2400, 4800, 7200, 9600, 14400, 19200, 38400, 57600, 115200, 128000	-
Data Bits	数据长度	数据长度，支持 8 字节	-
Parity Bit	校验位	校验位，可选择奇/偶校验或无校验	-
Stop Bits	停止位	停止位，可选择 1 位/2 位	-

以下为具体使用参考案例：

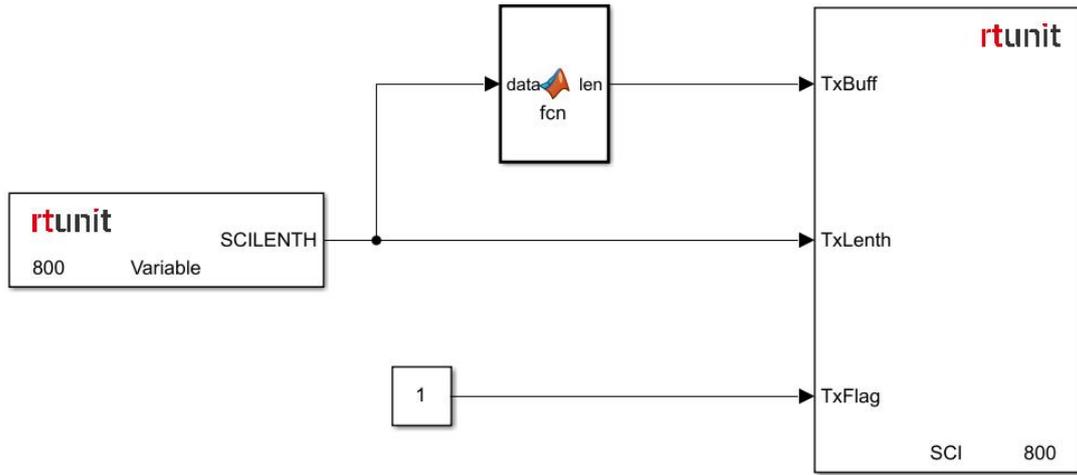


图 5.50 RS485 Transmit 模型实例

```
function y = fcn(~)
    data =uint16( 1:1:255);
    y = data';
```

图 5.51 RS485 Transmitx MATLAB Function 代码

5.4.19. SYSCFG 模块

SYSCFG 模块用于配置系统控制频率，双击 SYSCFG 模块可进入模块参数设置界面，如图 5.52 所示。



图 5.52 SYSCFG 模块

SYSCFG 模块参数说明如下表 5.24 所示。

表 5.24 SYSCFG 模块参数说明

项目	名称	参数说明	数据类型
Frequency	数据	控制频率，详见说明 1	-

说明 1:

控制频率的概念和仿真步长的概念是相同的，即每隔多长时间精准的执行一次控制程序。设定范围为 1KHz~50KHz，推荐设置为 10KHz，即 10000。设定的频率越低，则执行的步长越长，能够执行的代码也就越多越复杂；设定的频率越高，则执行的步长越小，能够执行的代码也就越少。如果代码执行的时间超过了设定的步长，RTU-BOX800 会提示超时，解决方法有两个：一是降低控制频率；二是简化控制模块。

说明 2:

采样时间 Sample time 设置的值与控制步长同步。如果控制频率为 10kHz，则采样时间变为 0.0001。其他模块的频率自动同步为系统频率。

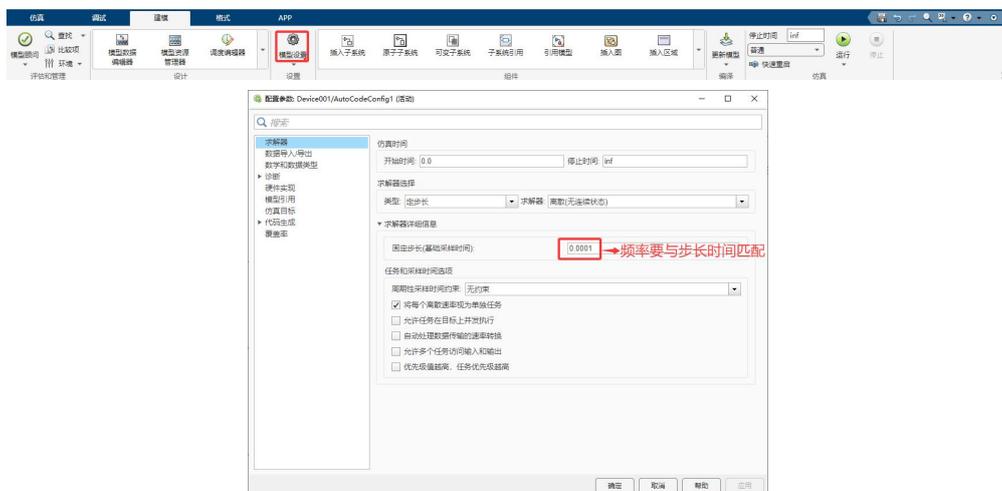


图 5.53 设置 Sample time

5.4.20. Variable 模块

Variable 模块用于观测/修改工程的变量，双击 Variable 模块可进入模块参数配置界面，如图 5.54 所示。在同一 Logic 模型中，最多支持 256 个 Variable 模块。

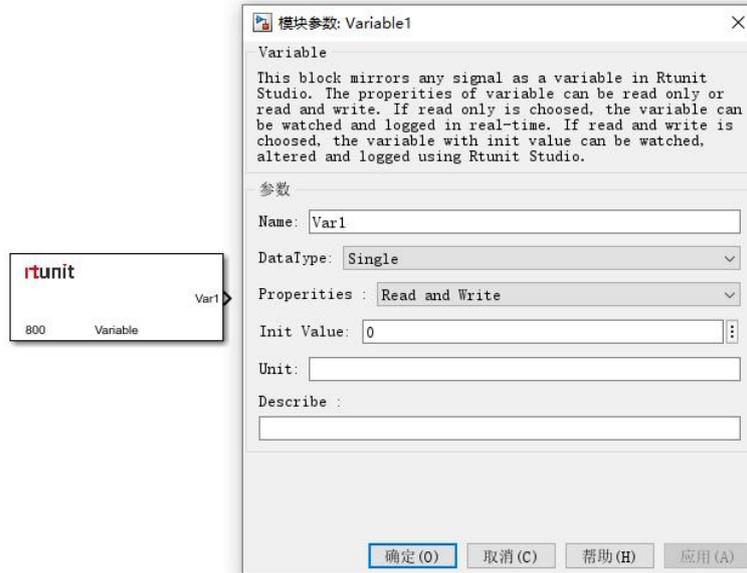


图 5.54 Variable 模块

Variable 模块参数说明如下表 5.25 所示。

表 5.25 Variable 模块参数说明

项目	名称	参数说明	数据类型
Name	变量名	变量名称，见说明 1	-
DataType	数据类型	数据类型，支持包括：Single、Int32、UInt32、Int16、UInt16、Int8、UInt8、Boolean	-
Properties	变量属性	变量属性，只读或者可读可写，见说明 2	-
Init Value	初始值	初始值（针对可读可写变量）	-
Unit	单位	变量单位，在 RTUS 中显示，非必须项	-
Describe	描述	描述，在 RTUS 中显示，非必须项	-

说明 1:

在同一个程序中，变量不能重名。变量名可由字母、下划线、阿拉伯数字构成。

说明 2:

只读（Read only）类型的变量只能在 RTUS 中观测值，不能写值。可读可写（Read and Write）类型的变量可以在 RTUS 中写值。

说明 3:

该模块主要作为变量实时调参的接口；可以用来在上位机上对变量进行显示和修改。

5.4.21. WAVE 模块

WAVE 模块用于将 Logic 模型中的变量以波形曲线的方式在 RTUS 中观察，曲线数据的采样率为 2kHz（每通道每秒采集 2000 个数值）。WAVE 模块最多支持 32 个通道曲线。双击 WAVE 模块可进入模块属性界面，如图 5.55 所示。在同一 Logic 模型中，最多使用 1 个 WAVE 模块。

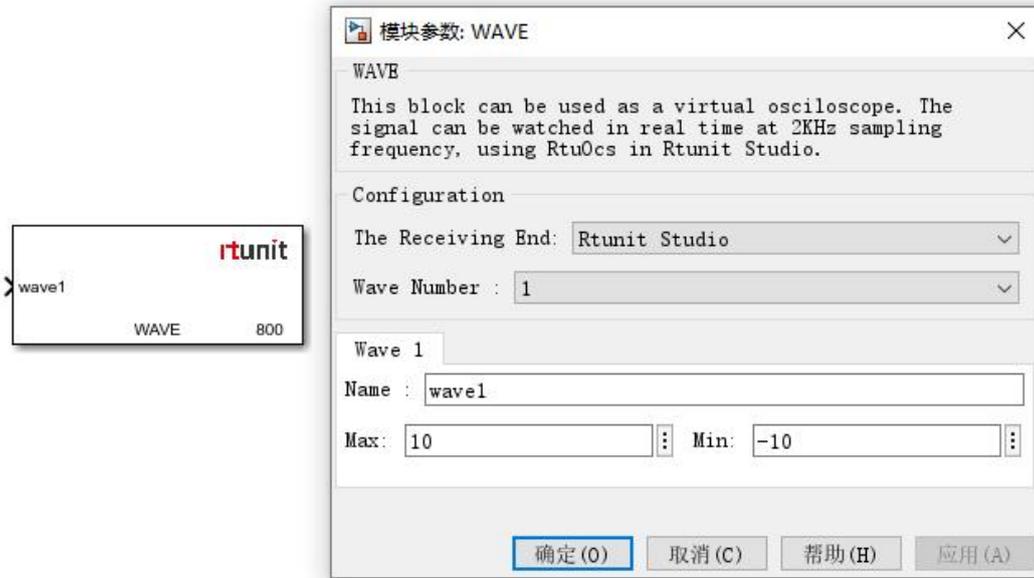


图 5.55 WAVE 模块

WAVE 模块参数说明如下表 5.26 所示。

表 5.26 WAVE 模块参数说明

项目	名称	参数说明	数据类型
The Receiving End	接收终端	数据传至 RTUS/Simulink，见说明 1	-
Wave Number	波形数量	选择输入到 WAVE 模块的通道数量	-
Name	波形名称	波形的名称	-
Max	波形最大值	波形在 RTUS 中坐标显示的上限	-
Min	波形最小值	波形在 RTUS 中坐标显示的下限	-

说明 1:

数据接口选择 Rtunit Studio，则 WAVE 模块数据传输至 RTUS；数据接口选择 Simulink，则 WAVE 模块数据传输至 Simulink，若选择 Simulink 接口，需要与 Tcp2Sim 模块配套使用。

说明 2:

该模块可以将某一变量在上位机上进行波形显示。需要将变量连接到 WAVE 接口上。

5.4.22. WAVE REC 模块

WAVE REC 模块用于将 Logic 模型中的需要记录的变量保存为文件，在 RTUS 的 Records 图标下可读取该文件，单次记录的数据总量为 80000。双击 WAVE REC 模块可进入模块属性界面，如图 5.56 所示。同一 Logic 模型中，最多使用一个 WAVE REC 模块。

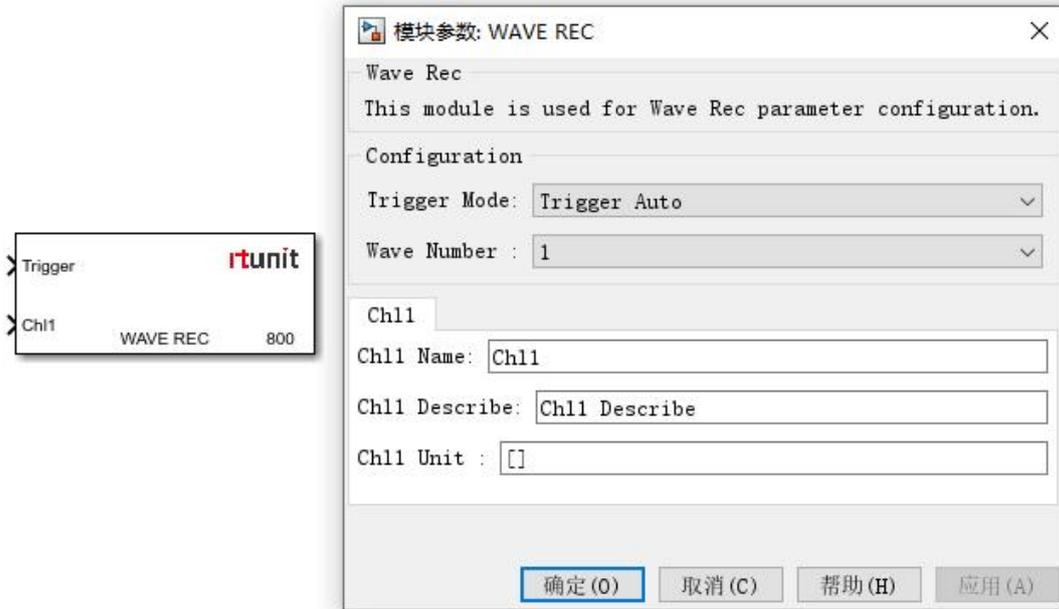


图 5.56 WAVE REC 模块

WAVE REC 模块参数说明如下表 5.27 所示。

表 5.27 WAVE 模块参数说明

项目	名称	参数说明	数据类型
Trigger Mode	触发模式	触发模式，见说明 1	-
Wave Number	录波数量	录波数量，见说明 2	-
Chl Name	波形名称	波形名称，不能重名	-
Chl Describe	描述	波形描述，非必需项	-
Chl Describe	单位	单位，非必需项	-
Trigger 输入端口	-	触发条件，仅适用于条件触发，见说明 1	Boolean
Ch1 输入端口	-	需要录波的数据输入	Single

说明 1:

触发模式支持条件触发（Trigger Auto）和手动触发（Rtunit Studio）。条件触发模式需要在输入端口 Trigger 设置触发条件，Trigger 由 0 变为 1 时，启动录波。录波完成后，再次由 0 变为 1，再次启动录波，如图 5.57 所示。

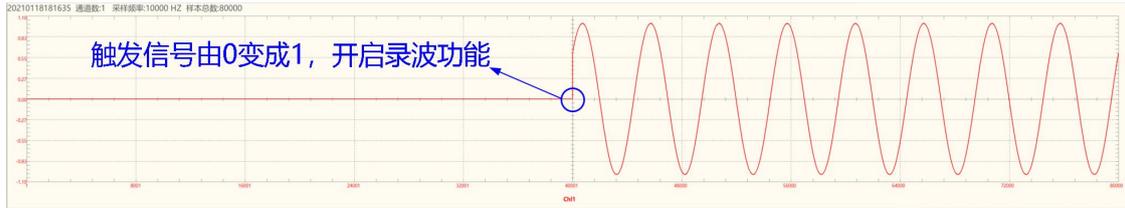


图 5.57 WAVE REC 模块 Trigger Auto 模式

手动触发模式直接在 Rtunit Studio 中按下录波按钮即可启动录波，录波完成后可再次启动，如图 5.58 所示。

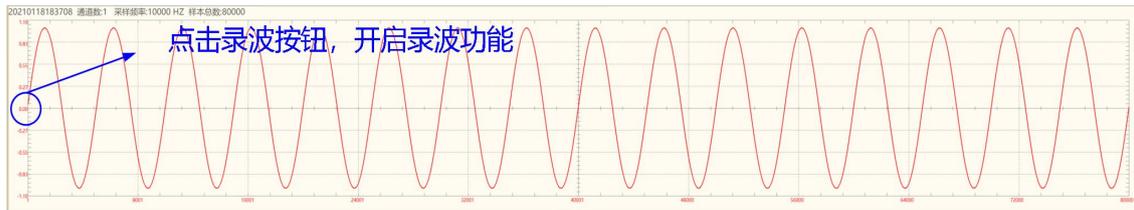


图 5.58 WAVE REC 模块 Rtunit Studio 模式

说明 2:

录波一共支持 8 条波形，数据总量为 8 万点。如果只录 1 条波形，则该条波形可录 8 万点。如果同时录制 8 条波形，则每条波形可录制 1 万点。程序运行过程中的每一个点都会录下来。

5.5. Transformations

Transformations 库中收录了一些电力电子相关算法模块,方便用户使用 RTUS 进行开发。
(Transformations 库将不断完善,请您关注 rtunit 公司公众号获取最新信息。)

5.5.1. AVG 模块

AVG 模块用于计算每个周期内信号的平均值,每周期末尾更新 AVG 计算结果,双击 AVG 模块可进入模块参数设置界面,如图 5.59 所示。

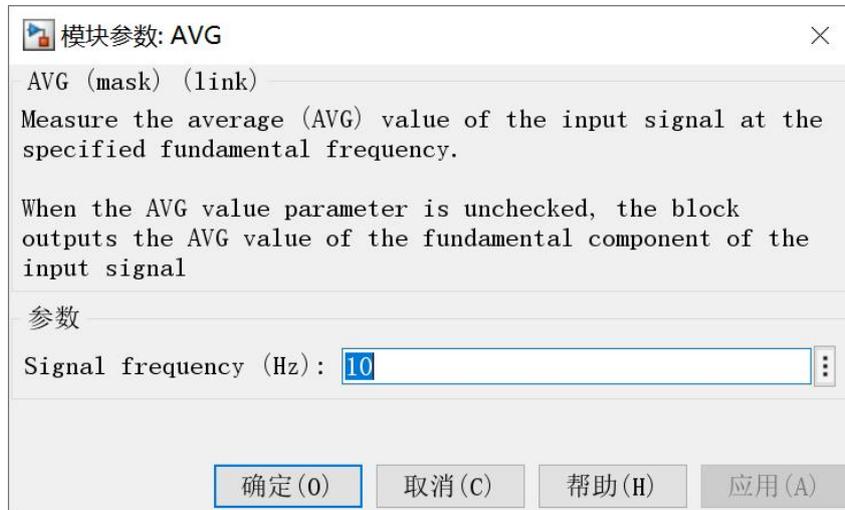
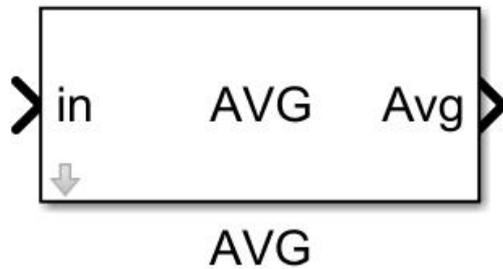


图 5.59 AVG 模块

AVG 模块参数说明如下表 5.28 所示。

表 5.28 AVG 模块参数说明

项目	名称	参数说明	数据类型
Signal frequency (Hz)	信号频率	信号频率, Freq=50; 取值范围: 1-1000HZ	Uint16

5.5.2. Clarke 模块 1

Clarke 模块的作用是将基于三轴二维的定子静止坐标系的物理量变换到二轴的定子静止坐标系中，简称为 3S/2S 变换。该模块用于三相对称的系统。双击 Clarke 模块可进入模块参数设置界面，如图 5.60 所示。

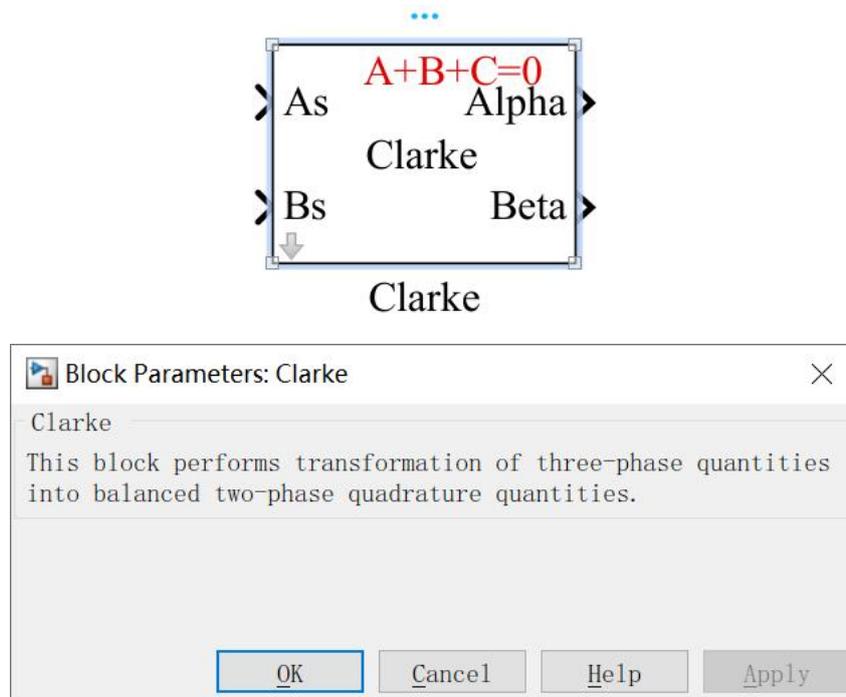


图 5.60 Clarke 模块

说明 1:

设三相电流 ABC 分别为 i_A , i_B , i_C ，根据基尔霍夫电流定律满足以下公式:

$$i_A + i_B + i_C = 0$$

设静止坐标系 $\alpha\beta$ ， α 轴的电流分量为 i_α , i_β ，则 Clarke 变化满足以下公式:

$$i_\alpha = i_A$$

$$i_\beta = \frac{1}{\sqrt{3}} * i_A + \frac{2}{\sqrt{3}} * i_B$$

5.5.3. Clarke 模块 2

Clarke 模块的作用是将基于三轴二维的定子静止坐标系的物理量变换到二轴的定子静止坐标系中, 简称为 3S/2S 变换, 该模块用于三相负载不对称的系统。双击 Clarke 模块可进入模块参数设置界面, 如图 5.61 所示。

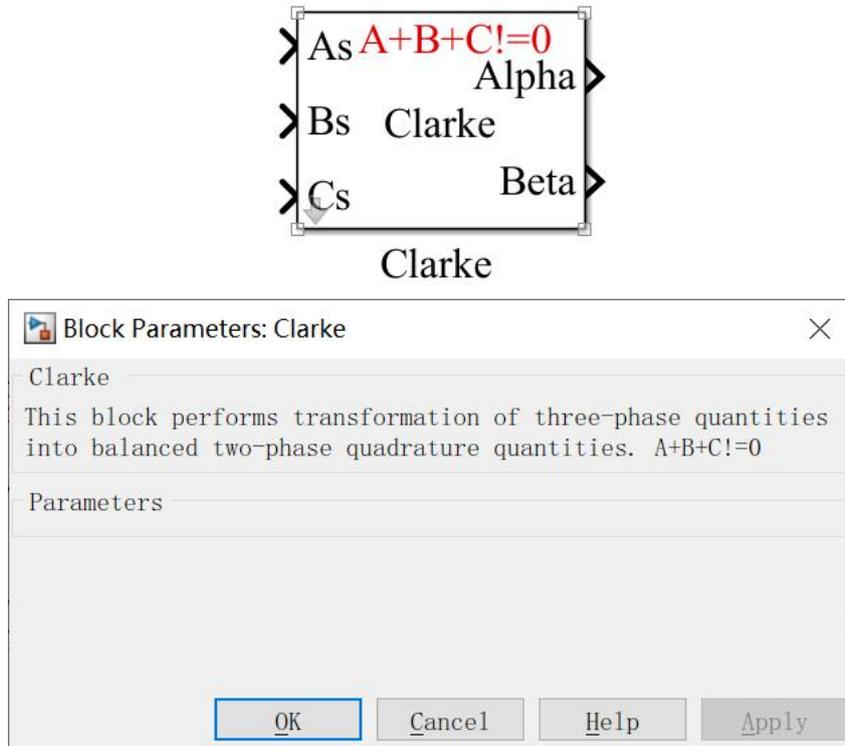


图 5.61 Clarke 模块

说明 1:

设三相电流 ABC 分别为 i_A , i_B , i_C , 当在以下情况下:

$$i_A + i_B + i_C \neq 0$$

设静止坐标系 $\alpha\beta$, α 轴的电流分量为 i_α , i_β , 则 Clarke 变化满足以下公式:

$$i_\alpha = \frac{2}{3} * i_A - \frac{1}{3} * i_B - \frac{1}{3} * i_C$$

$$i_\beta = \frac{1}{\sqrt{3}} * i_B - \frac{1}{\sqrt{3}} * i_C$$

5.5.4. IClarke 模块

IClarke 模块的作用是将静止坐标系 α - β 变换到自然坐标系 ABC。双击 IClarke 模块可进入模块参数设置界面，如图 5.62 所示。

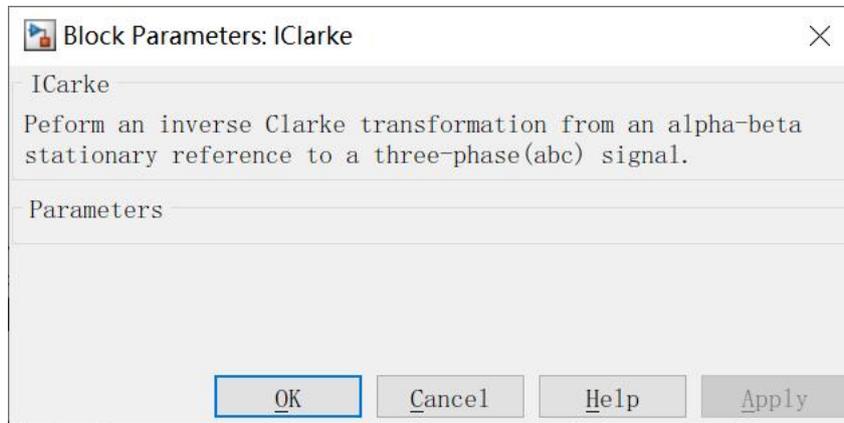
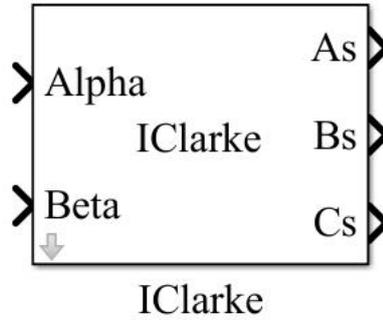


图 5.62 IClarke 模块

IClarke 模块表达式为:

$$i_A = i_\alpha$$

$$i_B = -\frac{1}{2} * i_\alpha + \frac{\sqrt{3}}{2} * i_\beta$$

$$i_C = -\frac{1}{2} * i_\alpha - \frac{\sqrt{3}}{2} * i_\beta$$

5.5.5. IPark 模块

IPark 模块的功能是将同步旋转坐标系 d-q 变换到静止坐标系 α - β ，双击 IPark 模块可进入模块参数设置界面，如图 5.63 所示。

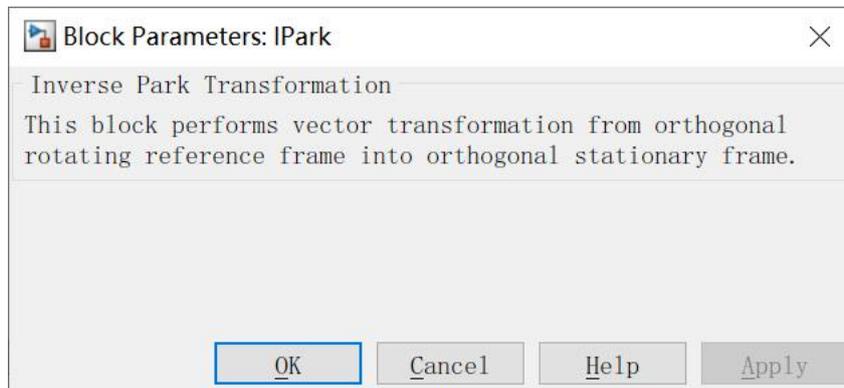
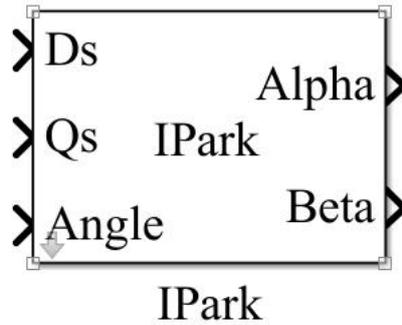


图 5.63 IPark 模块

IPark 模块表达式为：

$$i_{\alpha} = i_d * \cos\theta - i_q * \sin\theta$$

$$i_{\beta} = i_d * \sin\theta + i_q * \cos\theta$$

5.5.6. Park 模块

Park 模块的作用是将静止的 abc 三相绕组中的物理量变为旋转的 dq0 等值绕组中的物理量。双击 Park 模块可进入模块参数设置界面，如图 5.64 所示。

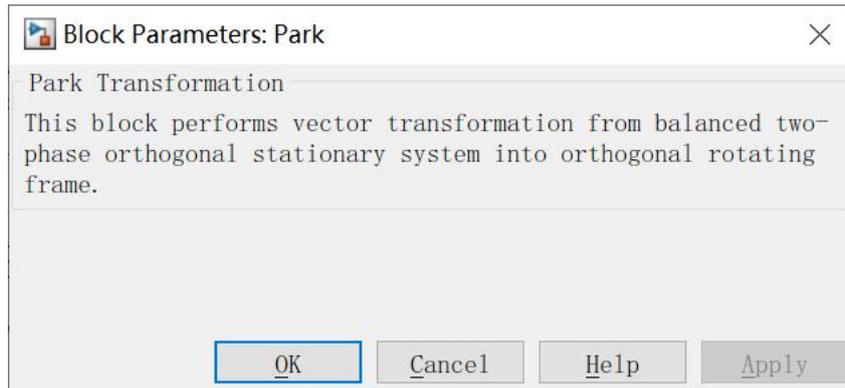
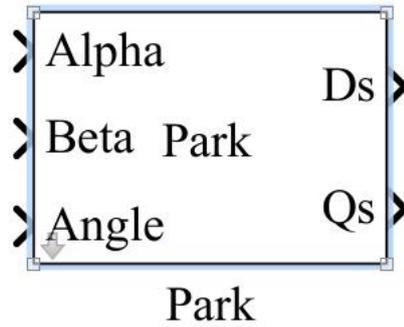


图 5.64 Park 模块

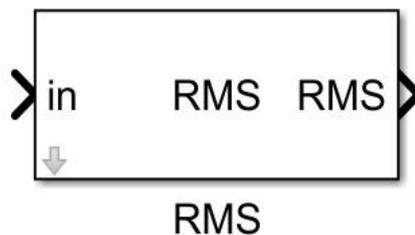
Park 模块表达式为:

$$i_d = i_\alpha * \cos \theta + i_\beta * \sin \theta$$

$$i_q = -i_\alpha * \sin \theta + i_\beta * \cos \theta$$

5.5.7. RMS 模块

RMS 模块用于计算每个周期内信号的均方根有效值，每周期末尾更新 RMS 计算结果，双击 RMS 模块可进入模块参数设置界面，如图 5.65 所示。



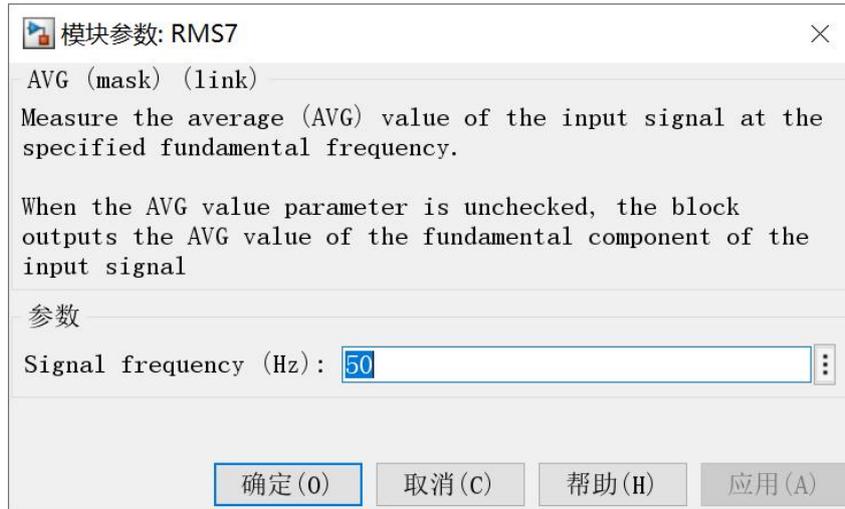


图 5.65 RMS 模块

RMS 模块参数说明如下表 5.29 所示。

表 5.29 RMS 模块参数说明

项目	名称	参数说明	数据类型
Signal frequency (Hz)	信号频率	信号频率, Freq=50; 取值范围: 1-1000HZ;	Uint16

说明 1:

设某个信号一个周期内数据为 X_1, X_2, \dots, X_n , 均方根有效值的计算公式为:

$$\text{RMS} = \sqrt{\frac{1}{N} \sum_{n=1}^N (X_n)^2}$$

5.5.8. SPLL-SOGI 模块

基于二阶广义积分器的单相并网系统锁相 (SPLL-SOGI) 计算模型锁电网电压 V_a 的相角。双击 SPLL-SOGI 模块可进入模块参数设置界面, 如图 5.66 所示。



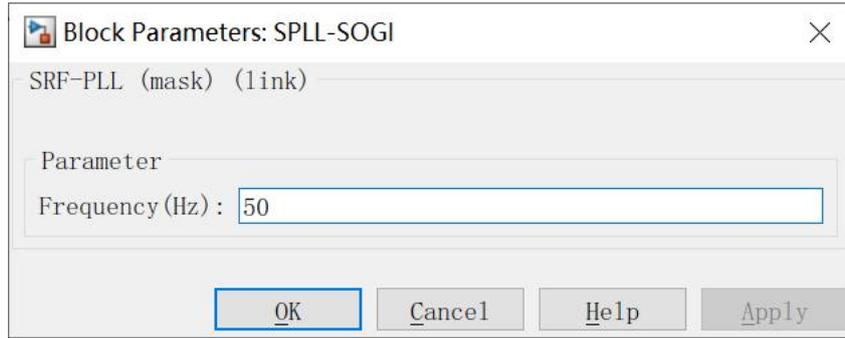


图 5.66 SPLL-SOGI 模块

SPLL-SOGI 模块参数说明如下表 5.30 所示。

表 5.30 SPLL-SOGI 模块参数说明

项目	名称	参数说明	数据类型
Ua	相电压	相电压输入	single
theta	相角	输出范围 $[0, 2\pi]$	single
Frequency	频率	单相电压的基准频率	-

5.5.9. SRF-PLL 模块

SRF-PLL 模块是基于 dq 同步旋转坐标变换的三相锁相环 (SRF-PLL) 计算模型锁电网电压 V_a 的相角。双击 SRF-PLL 模块可进入模块参数设置界面, 如图 5.67 所示。

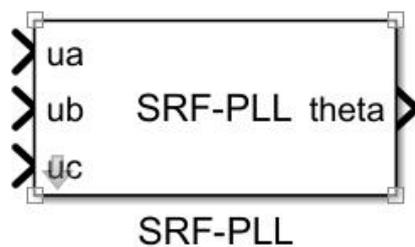


图 5.67 SRF-PLL 模块

SRF-PLL 模块参数说明如下表 5.31 所示。

表 5.31 SRF-PLL 模块参数说明

项目	名称	参数说明	数据类型
ua, ub, uc	相电压	相电压输入	single
theta	相角	输出范围[0, 2π]	single

5.5.10. VPP2Vphase 模块

VPP2Vphase 模块用于将采样的线电压转相电压，双击 VPP2Vphase 模块可进入模块参数设置界面，如图 5.68 所示。

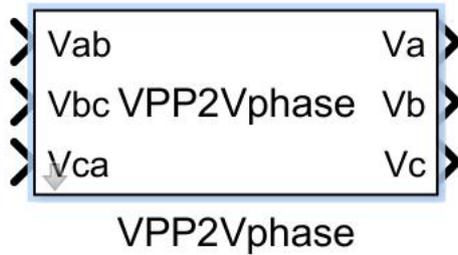


图 5.68 VPP2Vphase 模块

设三相线电压瞬时值为 V_{ab} ， V_{bc} ， V_{ca} ，则相电压计算公式为：

$$V_a = \frac{V_{ab} - V_{ca}}{3}$$

$$V_b = \frac{V_{bc} - V_{ab}}{3}$$

$$V_c = \frac{V_{ca} - V_{bc}}{3}$$

RTU-BOX800 使用技巧&注意事项

摘要: 本章凝结了众多使用 RTU-BOX 系列控制器进行开发或者科研的工程师们的心血。如您是第一次使用 RTU-BOX800 进行开发或者科研,相信这章会让您更快速的入门。若您在使用过程中有自己独到的见解,请联系我们或发布在 rtunit 论坛中,我们将会择优加入本章。

Tips 1:

Logic 模型中所有的模块必须是离散状态的数据,不能使用连续状态的数据。因此在使用 Logic 模型时,首先需要将 S 域模块离散成 Z 域再进行后续代码生成——编译——下载操作。

Tips 2:

搭建 Logic 模型时,除了使用 RTU-BOX800 自带的硬件模型库 Rtunit Toolbox 之外,强烈建议使用 Simulink 基本库,如图 6.1 所示。

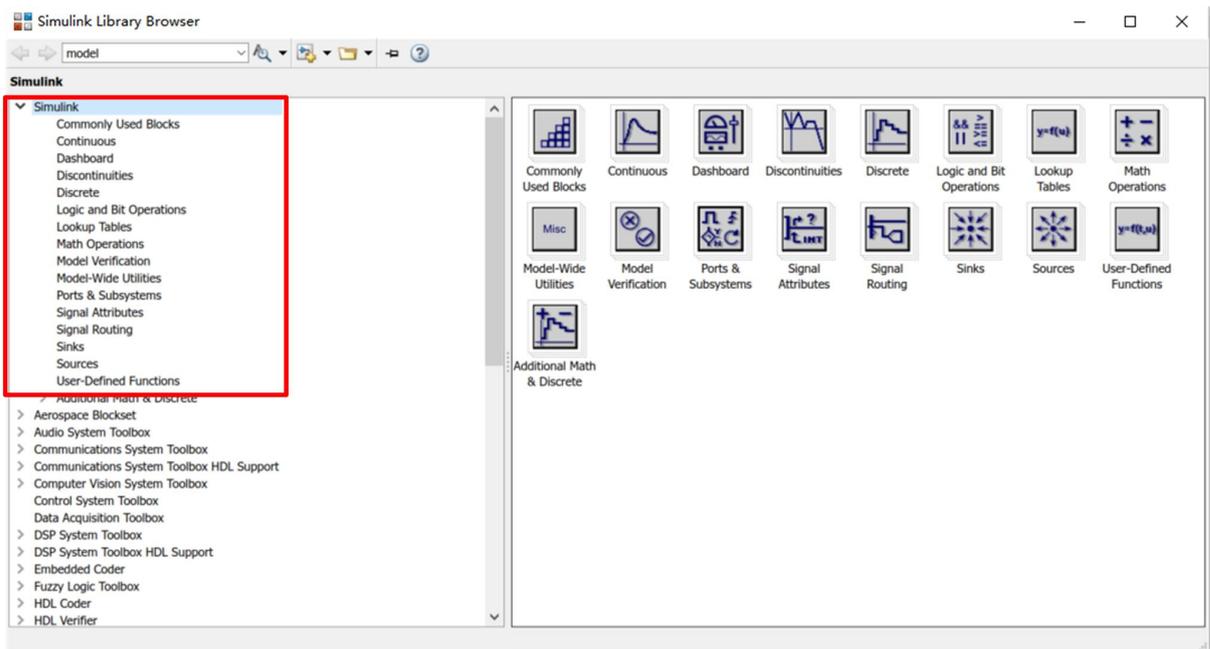


图 6.1 Simulink 基本库

Tips 3:

Simulink 基本库中,包括以下但不限于这些模块不支持代码生成,如图 6.2 所示。

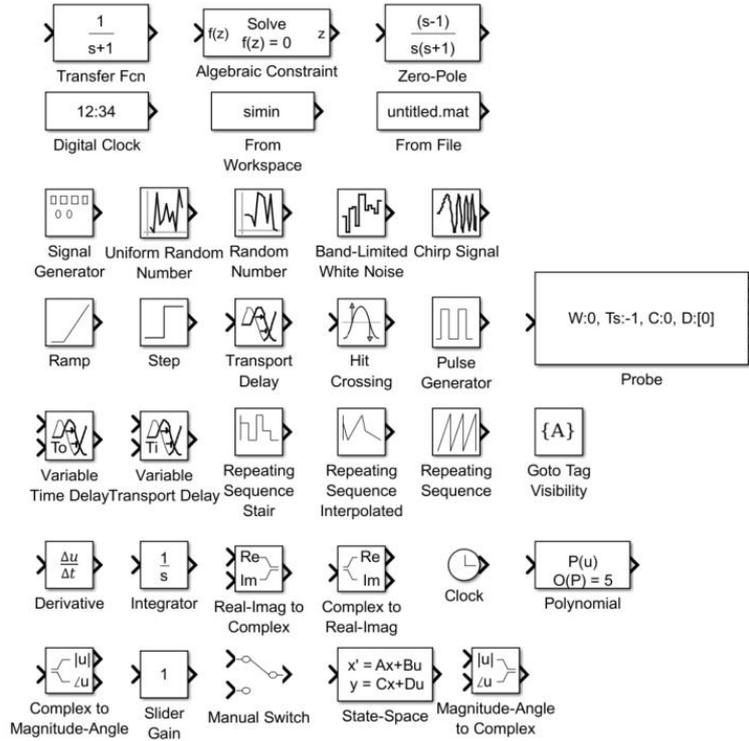


图 6.2 Simulink 基本库中不支持代码生成的模块

Tips 4:

Simulink 支持代码生成的模块都可以在 Logic 模型中正常使用。Logic 模型支持 MATLAB Function 的代码自动生成，但不支持 s Function 直接自动生成代码，s Function 需要另外编写相匹配的 tlc 文件。

Tips 5:

为了避免代码生成出错，常数 Constant 模块的采样时间参数 Sample time 推荐设置为-1。

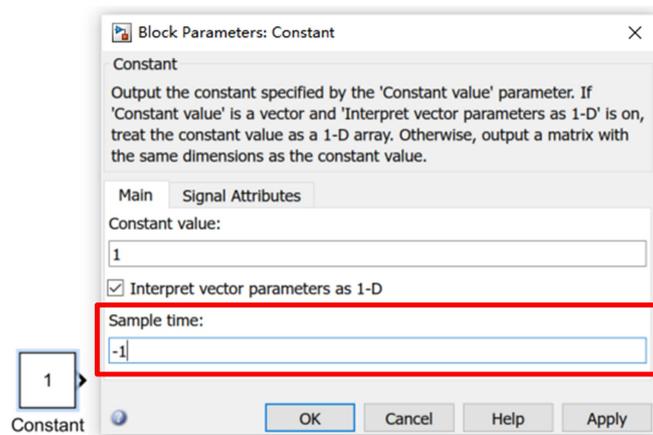


图 6.3 Sample time 设置为-1

Tips 6:

在使用 Simulink 自带的离散 PID、积分器以及滤波器时，首先需要在触发模型中设置控制器的步长。采样时间类型 Sample time type 选择 periodic，采样时间 Sample time 设置的值和控制步长一致。如，控制频率设置为 10kHz，则采样时间设置为 0.0001。

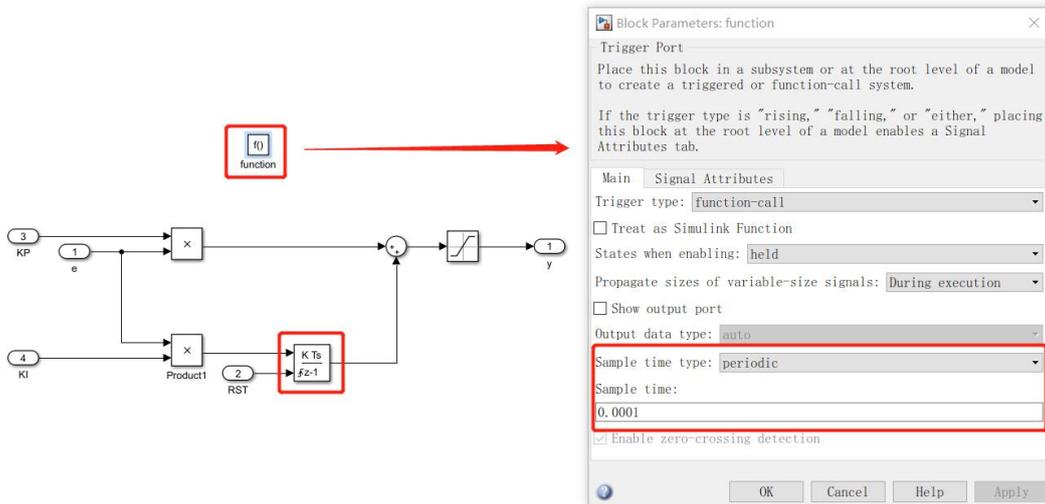


图 6.4 使用积分器时触发模型中设置控制器步长

Tips 7:

在使用 Simulink 自带的 Sine Wave 模型时，需要先在触发模型中设置控制器的步长。采样时间类型 Sample time type 选择 periodic，采样时间 Sample time 设置的值与控制步长一致。如果控制频率为 10kHz，则采样时间设为 0.0001。

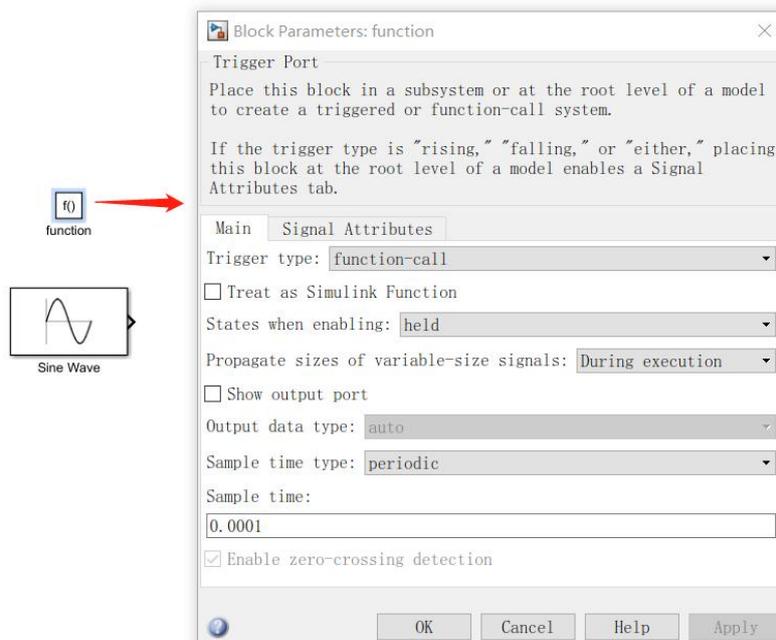


图 6.5 使用 Sine Wave 模块在触发模型中设置控制器步长

Tips 8:

Simulink 模型中不允许闭环反馈回路的存在，即“代数环”。若必须使用闭环反馈回路，比如锁相环，可以在反馈回路中串联 z^{-1} 打断回路（代价是产生一个控制周期的延迟）。因此，最好的方法是先对模型进行处理，将反馈回路简化掉。

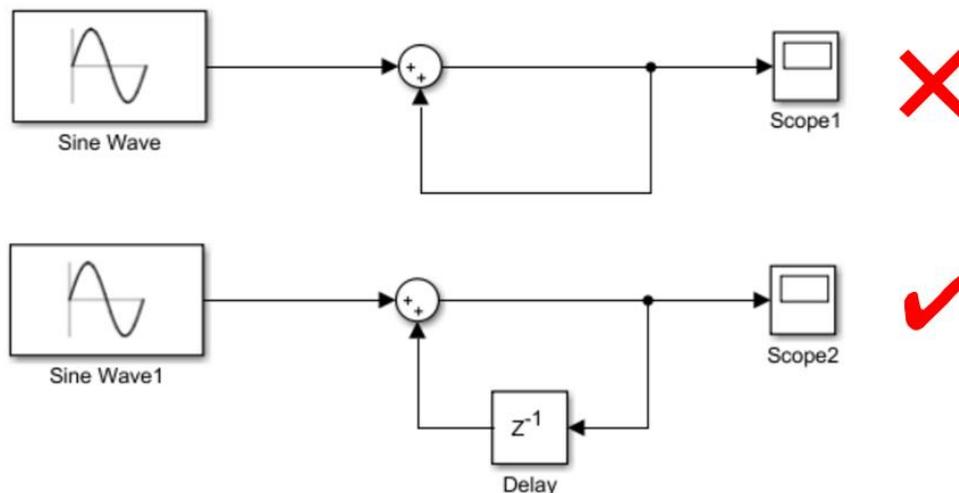


图 6.6 代数环问题的方案

Tips 9:

观察静态变量或者直流变量推荐使用 Variable 模块；观察动态变量或者交流变量推荐使用 Wave 模块。但是 Wave 模块的输出波形并不适合放在论文中，因为 Wave 模块的采样率为 2kHz，波形失真会比较明显。想要提取 RTU 的内部信号，可以选用 Record 模块或者用 DAC 模块输出到示波器。

Tips 10:

DAC 输出同步信号，配合示波器的 single 模式，能够极大提升波形采样的效率。

Tips 11:

若想做 PWM 调制，直接使用 Rtunit Toolbox 库中的 PWM 类模块即可。不需要搭一个三角载波然后与调制波比较，这是仿真和实物的区别。

Tips 12:

推荐使用 Stateflow 完成底层模型，用以解决含时序和状态的问题。不建议将 Stateflow 用于上层，因为这样会降低模型的可读性。

Tips 13:

更换 Matlab/Simulink 软件记得重新安装 Rtunit Toolbox。



让控制简单高效!

CONTACT US



www.rtunit.com



025-52458092



sales@rtunit.com



南京市江宁开发区
铺岗街381号德茂
大厦5F

免责声明:

永远使用由原始制造商为此机器生产的备件。如果使用了非原始备件,南京瑞途优特信息科技有限公司对机器的任何损坏或损失不承担责任。

本资料版权归南京瑞途优特信息科技有限公司所有,未经本公司书面许可,不允许对此出版物的任何部分通过任何方法以任何形式进行复制或使