

EasyGo DeskSim 软件 实时仿真起步指南

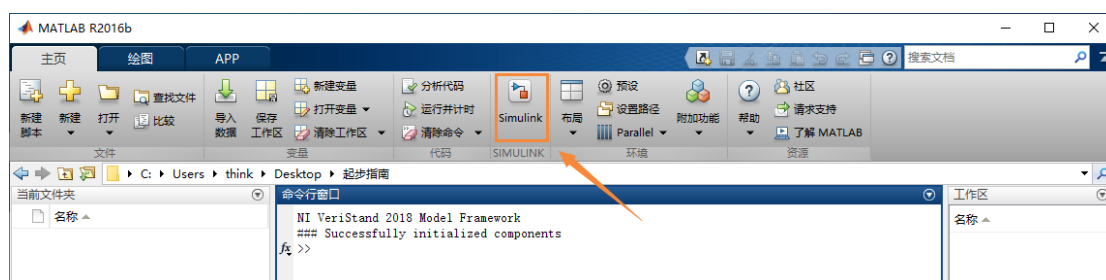
目 录

软件简介	3
仿真机的软件安装	5
实时仿真操作流程	7
Template 1: CPU Alone	8
准备程序	8
导入程序	10
仿真配置	10
添加自定义界面	13
程序运行	15
Template 2: CPU + FPGA	18
准备程序	18
导入程序	21
仿真配置	22
添加自定义界面	25
程序运行	27
Template 3: CPU + FPGAs	31
准备程序	31
导入程序	36
仿真配置	36
添加自定义界面	39
程序运行	42
Template 4: FPGA Alone	45
准备程序	45
导入程序	48
仿真配置	49
添加自定义界面	50
程序运行	52
Template 5: FPGAs	55
准备程序	55
导入程序	59
仿真配置	59
添加自定义界面	60
程序运行	63
EasyGo FPGACoder 使用流程	67
EasyGo Template : FPGACoder	68
准备程序	68
导入程序	73
模型配置	74
添加自定义界面	75
程序运行	78

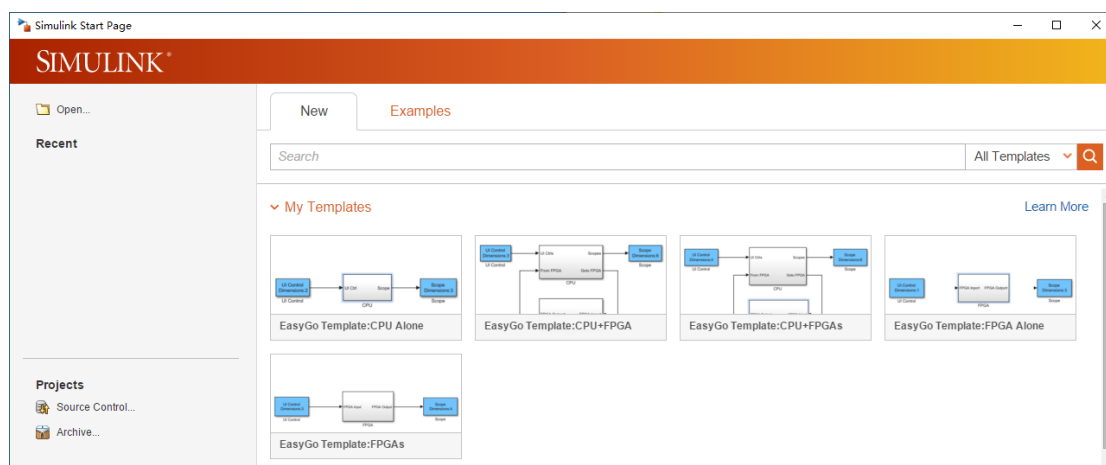
软件简介

EasyGo DeskSim 是一款配置型的软件，使用简单，同时功能强大。能实现 CPU Model 的实时运行，电力电子模型 ns 级别在 FPGA 上实时仿真运行，CPU 模型与 FPGA 联合仿真或实现 RCP 与 HIL 自闭环运行，多 FPGA 的并行仿真等多种复杂的实时仿真功能。

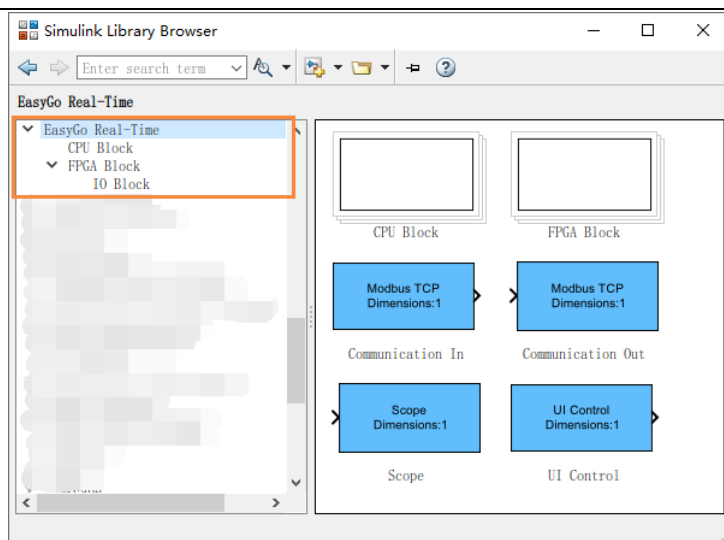
当我们正常安装完我们的 EasyGo DeskSim 软件后，打开 Matlab 软件，点击启动 Simulink，如下图所示：



启动 Simulink 后，弹出“Simulink Start Page”，在该页面中我们可以看到在“My Templates”目录下，会有 5 个“EasyGo Template”命名的模板文件，我们可以根据不同的需求打开不同的模板进行快速模型搭建。



同时我们打开“Simulink Library Browser”页面，在该页面中可以看到“EasyGo Real-Time”函数库，如下图所示：



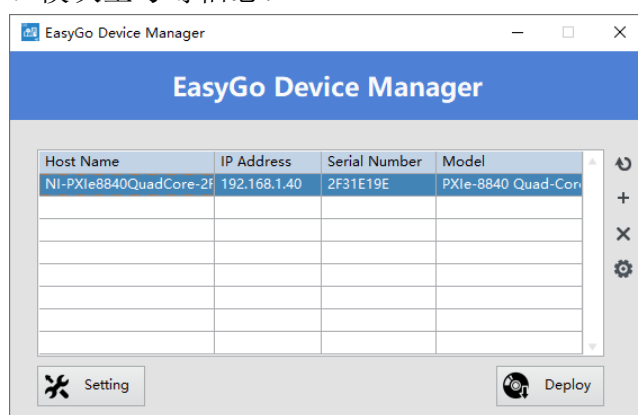
利用 EasyGo Real-Time 函数库和 5 个模板模型可以实现快速搭建模型，具体每个模板模型的功能用途如下：

CPU Alone	利用 EasyGo Real-Time 函数库建立仅用于 CPU 的实时模型框架
CPU + FPGA	利用 EasyGo Real-Time 函数库建立用于 CPU 和单卡 FPGA 联合的实时模型框架
CPU + FPGAs	利用 EasyGo Real-Time 函数库建立用于 CPU 和多块 FPGA 联合的实时模型框架
FPGA Alone	利用 EasyGo Real-Time 函数库建立仅用于单卡 FPGA 的实时模型框架
FPGAs	利用 EasyGo Real-Time 函数库建立仅用于多块 FPGA 联合的实时模型框架

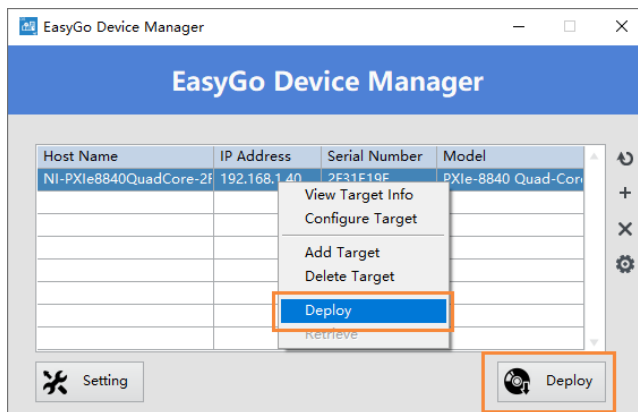
仿真机的软件安装

在进行实时仿真之前，第一次使用实时仿真机时，需要对仿真机进行系统软件安装。（注意：如果用户的实时设备上的软件已经正确安装且正常运行，可忽略这一步操作。）

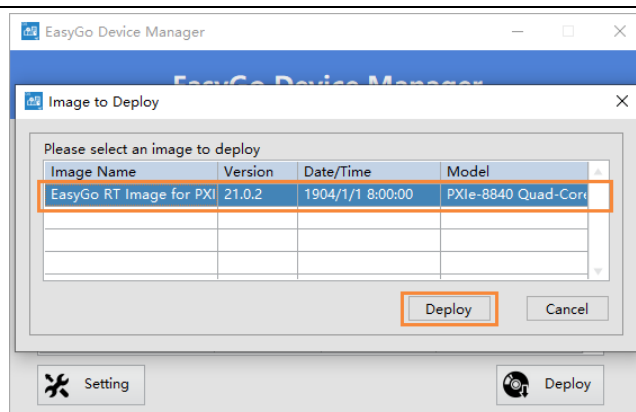
在开始菜单中找到并打开 EasyGo Device Manager，程序启动后会自动寻找网络中的可用设备，找到设备后，如下图所示，会显示当前找到的设备的名称、IP 地址、序列号、模块型号等信息：



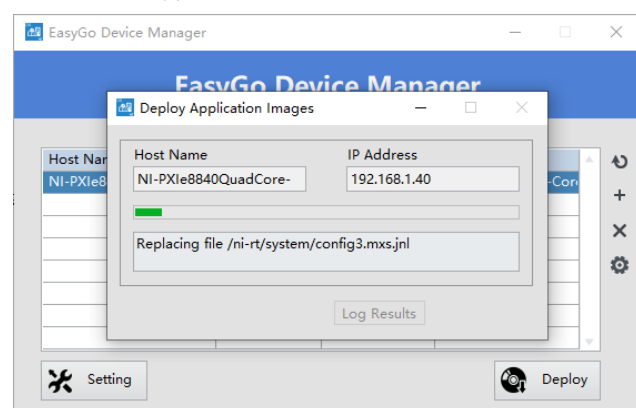
右键点击目标设备，在快捷菜单中选择“Deploy”或在程序右下角点击“Deploy”按钮，如下图所示：



此时会弹出“Image to Deploy”对话框，如下图所示，该页面中显示的是可供用户部署到指定设备的镜像文件：

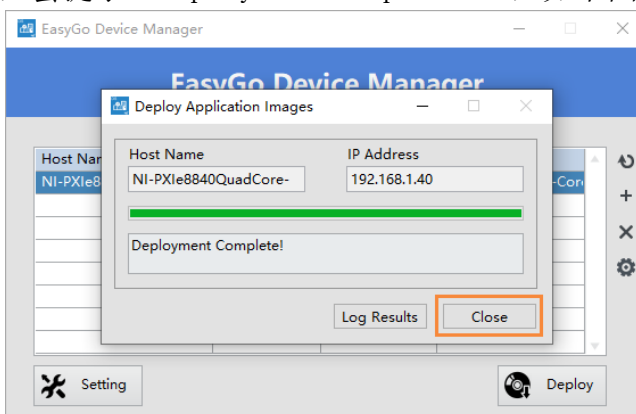


选择需要部署的镜像文件，点击“Deploy”按钮，即可进入部署流程，此时该界面中会实时显示当前软件安装进度，如下图所示：



注意：整个部署过程会持续 1-2 分钟，期间请勿关闭软件或断开设备的连接。在部署过程中，实时设备可能又重启行为，属正常情况。

部署完成后，会提示“Deployment Complete!”，如下图所示：



点击“Close”按钮，退出部署对话框，回到主界面，此时会重新刷新设备列表，刷新完成后，即整个下位机软件安装部署完成。

注：EasyGo Device Manager 工具的详细使用方法可参考 EasyGo DeskSim 的帮助文档中的相关说明。

实时仿真操作流程

完成了实时仿真机的系统软件安装后，后续的实时仿真使用中就不需要每次安装系统软件了。

接下来，利用 EasyGo DeskSim 软件进行实时仿真时，需要从离线仿真程序到使用 EasyGo DeskSim 软件进行实时运行，具体流程如下所示：

- 准备程序
- 导入程序
- 仿真配置
- 添加自定义界面
- 程序运行

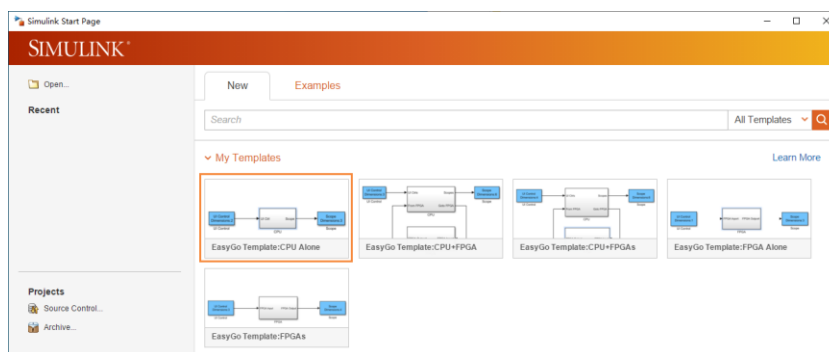
我们根据实际的应用需求，选择合适的框架模型，依照操作流程进行解释说明。

Template 1: CPU Alone

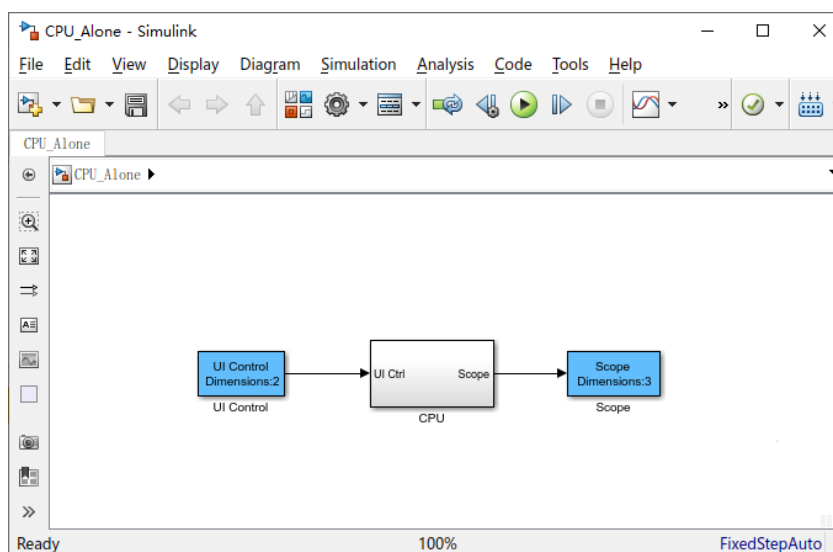
CPU Alone 模型是利用 EasyGo Real-Time 函数库建立仅用于 CPU 的实时模型框架，即在实际仿真过程中只使用 CPU 进行仿真，并将仿真结果上传至上位机观察或交互至工业通讯端口。

准备程序

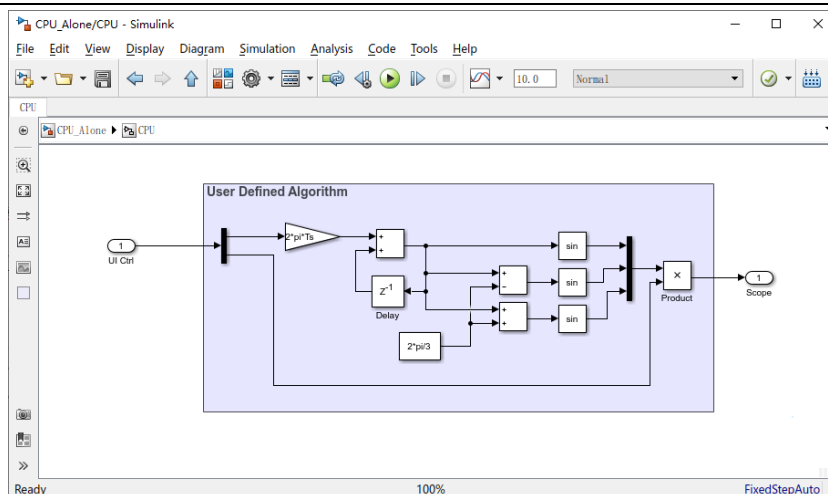
打开 Matlab 软件，点击启动 Simulink，在“Simulink Start Page”界面中点击“EasyGo Template: CPU Alone”创建出 CPU Alone 的框架模型，如下图所示：



新建出来的原始模型是未保存的，我们将其保存成“CPU_Alone.slx”，如下图所示：

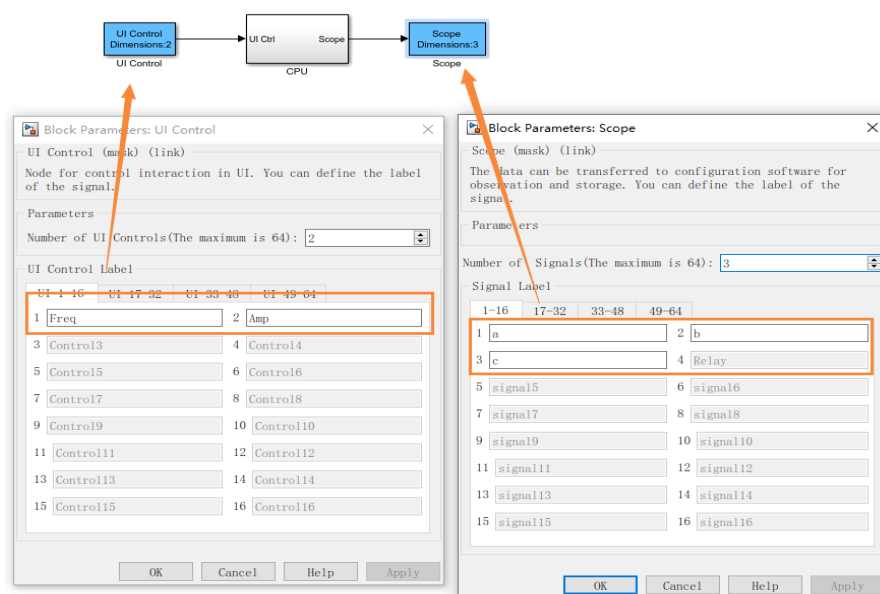


由于是纯 CPU 模型仿真程序，所以核心代码在“CPU”子系统里面，双击打开“CPU”子系统，如下图所示：



我们可以看到是一个简单的三相正弦波生成的程序，返回主程序，我们分别打开输入端“UI Ctrl”和输出端“Scope”的配置窗口，如下图所示：这里我们把所有需要在 DeskSim 软件界面控制的信号全部配置在“UI Control”模块中，把所有需要在 DeskSim 软件界面上观察的信号全部配置在“Scope”模块中，同时在连入子系统后的“UI Control”的信号根据程序的需要解绑连入程序中，将程序输出的信号根据需要按顺序捆绑在一起输出至“Scope”模块中。

注意：程序中所有与主界面交互的数据必须通过“UI Control”模块和“Scope”模块进行交互，程序中存在的其他的输入输出信号均不支持在 DeskSim 软件上进行控制或显示，且同一个程序有且最多支持一个“UI Control”模块或“Scope”模块。（通讯模块使用方式同理，详细说明可参考 [EasyGo RealTime Block 帮助文档](#)）



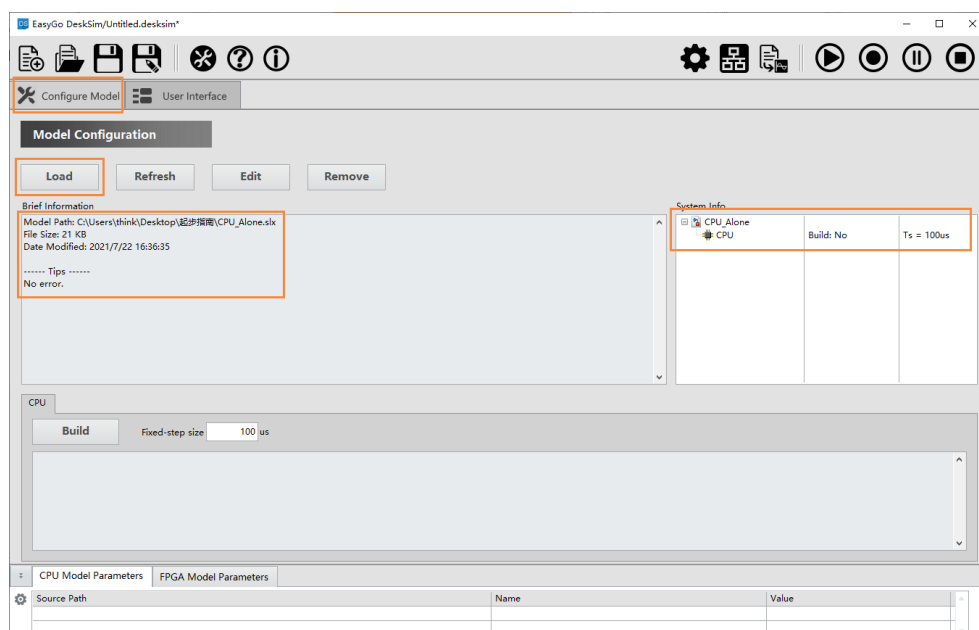
由程序可知，整个程序实现的功能是输入端输入幅值和频率，控制三相正弦

波，输出端为三相波的每个计算周期的瞬时值。

注：用户可以根据自己的 CPU 上的算法修改“CPU”子系统中的代码，以及对应修改输入输出端口的信号名称及数量。

导入程序

打开 EasyGo DeskSim 软件，在 Configure Model 界面中点击“Load”按钮载入程序文件；



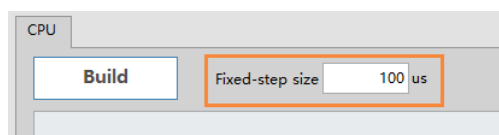
载入文件后，软件会分析得到程序的基本信息。

注意：

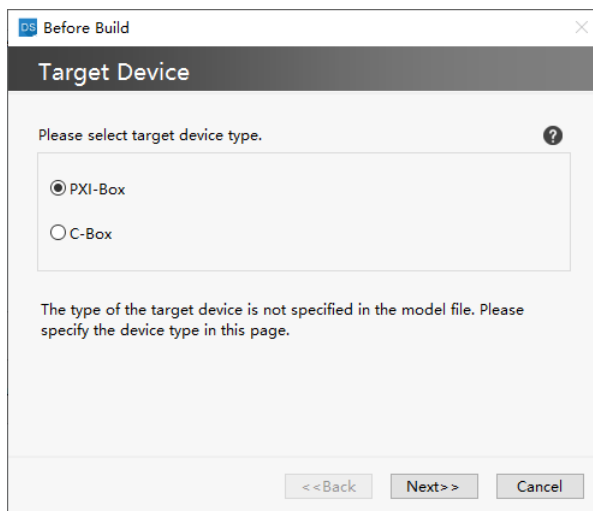
- ◆如果模型中存在错误，用户需要把错误信息全部处理掉，才能进行实时运行；
- ◆如果需要修改或者打开模型文件，可以点击“Edit”按钮打开模型文件进行修改或查看；
- ◆如果用户载入的模型文件发生了修改，在界面中可以点击“Refresh”按钮进行更新。

仿真配置

针对 CPU 模型，我们需要对 CPU 模型进行编译操作，如下图所示，我们先设置程序运行的步长，如 100us，即 10k 的控制频率；



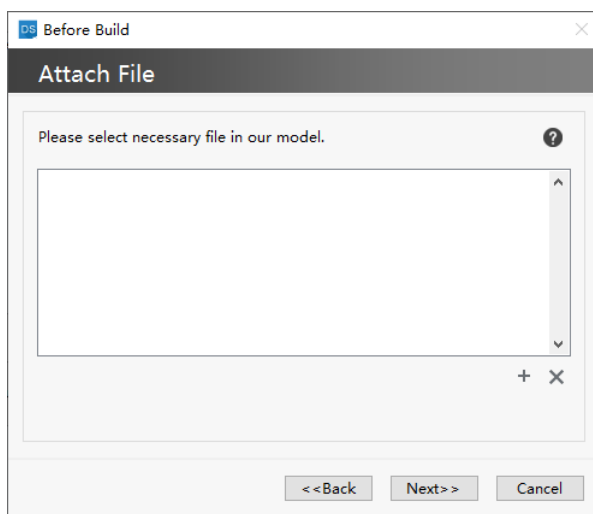
步长设置完成后，我们开始编译 CPU 模型，点击界面中的“Build”按钮，会弹出 Before Build 窗口，供用户对实际的编译操作进行与配置。如下图所示：



在“Target Device”界面中，提示用户确认硬件信息，用户可根据实际需求修改硬件类型。

注意：当用户的模型文件中有较为明确的硬件信息，则该页面中的硬件选择为禁用的，如果用户需要修改模型的硬件信息，用户需要在原始文件中进行修改，并重新载入；当用户的模型文件中没有明确的硬件信息，则用户可以直接在该页面中进行硬件信息选择。

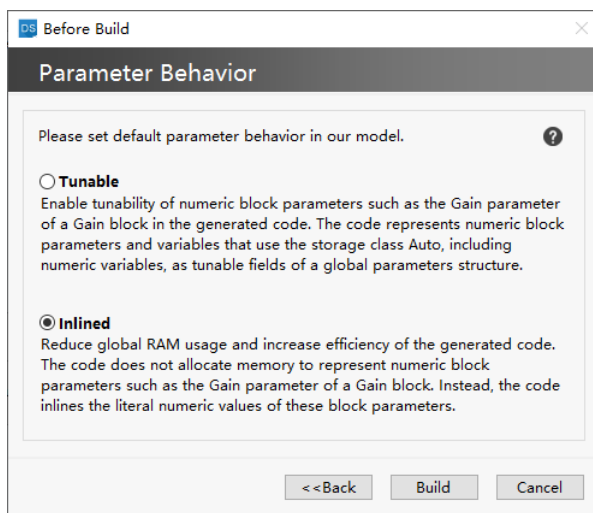
由于我们使用的 CPU 模型中没有任何硬件相关的设置，所以我们需要手动设置硬件类型，此处我们选择“PXI-Box”，硬件类型配置完成后，点击“Next>>”进入下一步配置，如下图所示：



在“Attach File”页面中，用户可以添加在模型初始化过程中需要用到的

文件，便于程序自动检测文件中所设置的变量；用户可通过右下角的添加或删除按钮进行操作。

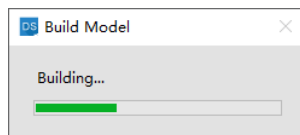
由于我们使用的 CPU 模型比较简单，没有用到相关的文件，所以我们跳过这一步，点击“Next>>”进入下一步配置，如下图所示：



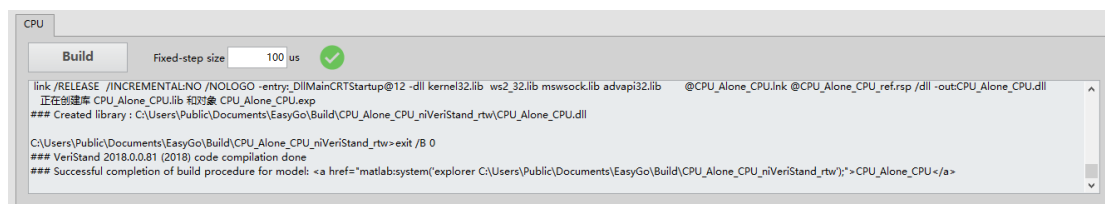
在“Parameter Behavior”页面中，用户可以设置是否支持在线调参功能：

- ◆ Tunable: 可调模式，是指程序中的一些常量或者增益的值是可以在线实时调整的，这个模式下程序运行时间会较长，效率较低。
- ◆ Inline: 不可调模式，是指程序中的一些常量或者增益的值是不可以在线实时调整的，这个模式下程序运行时间会较短，效率更高。

由于我们使用的 CPU 模型比较简单，我们直接选择“Inline”模式，点击“Build”按钮，进入模型编译操作，如下图所示：该过程会持续几分钟时间（具体与模型的复杂程度有关）；



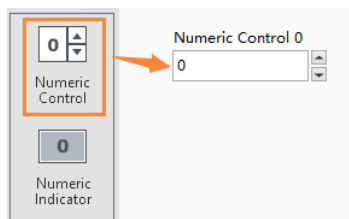
编译完成后，会提示编译结果  编译成功，如下图所示：



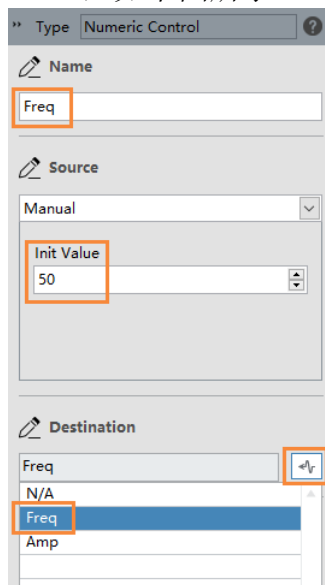
添加自定义界面

添加 UI 界面的控制输入

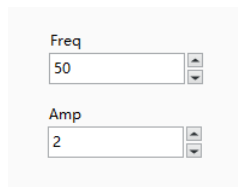
DeskSim 主界面切换至 UI Interface 界面中，在界面左侧的控件选板中，左键点击“Numeric Control”，并将其拖拽到中间的空白编辑区域释放，即完成一个 Numeric Control 控件的添加，如下图所示：



添加完成之后，点击选中这个控件，界面右侧栏会显示这个控件的配置内容，可选择信号绑定的对象（程序的所有输入端），修改名称和初始值；点击信号选择按钮，在弹出的信号列表中选择需要绑定的信号，如“Freq”，选中后，控件名称会同步更新为信号名称，用户可以使用信号名称或手动修改名称，设置完成后，我们对设置初始参数为“50”，如下图所示：

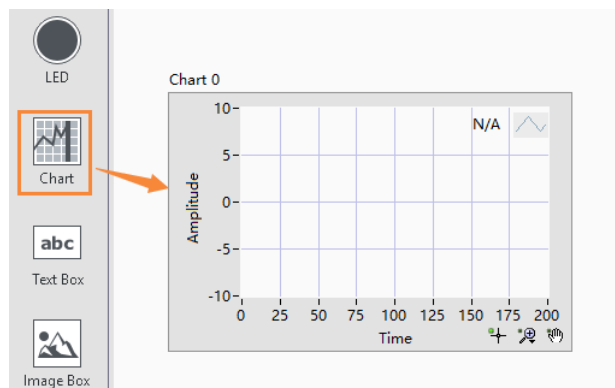


同样的方式，将程序的两个输入端口都绑定到界面的控件之后可以得到：

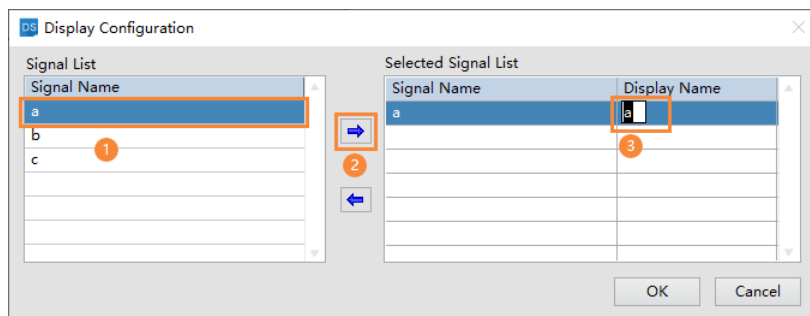
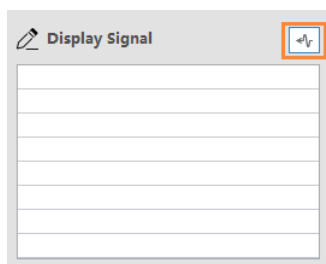


添加 UI 界面的波形显示

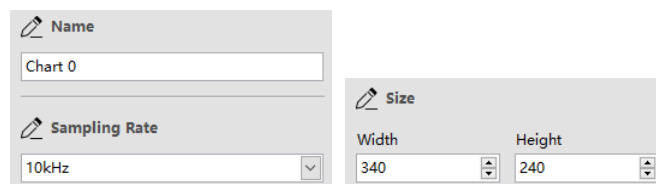
在界面左侧的控件选板中，左键点击“Chart”，并将其拖拽到中间的空白编辑区域释放，即完成一个 Chart 控件的添加，如下图所示：



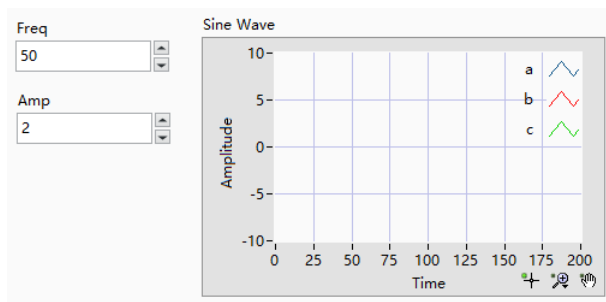
选中 Chart 控件，在右侧显示的配置区域中可以选中提那就显示的信号源（程序的所有输出信号均可显示）。



并可以修改 Chart 的名称、采样率和大小。



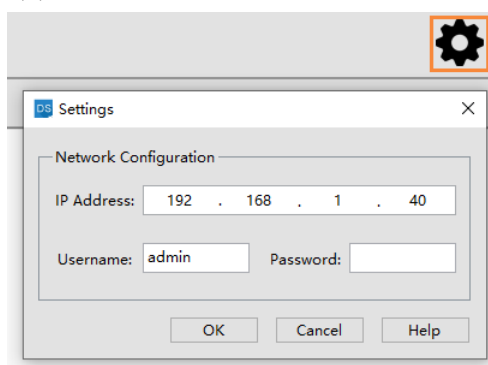
配置完成后，可以得到最终的自定义界面，如下图所示：



程序运行

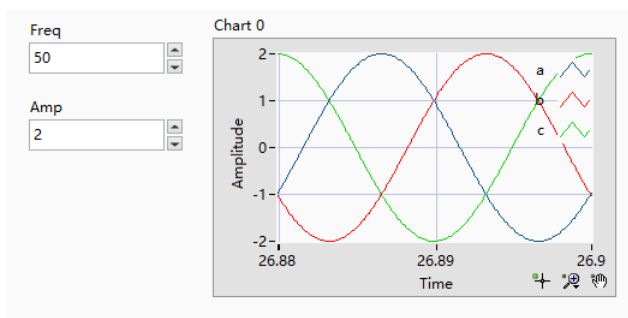
配置仿真机的网络信息

点击 DeskSim 软件工具栏的  按钮，即可打开仿真机的网络配置窗口，设置好 IP 地址、端口号等信息。



运行程序

点击软件工具栏的  按钮即可运行程序（目标仿真机上需要正确激活才能正常使用）。

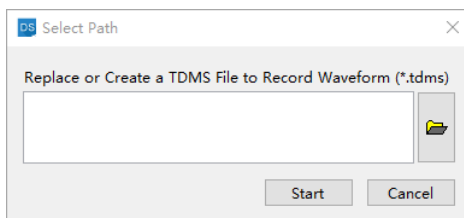


我们可以实时调整波形的幅值和频率来观察实际波形的变化情况。具体现象不再详述。同时运行过程中还可以动态修改附加区中 CPU Model Parameter 的参数，具体参考 EasyGo DeskSim 软件的帮助文档中的相关说明，由于我们的模型

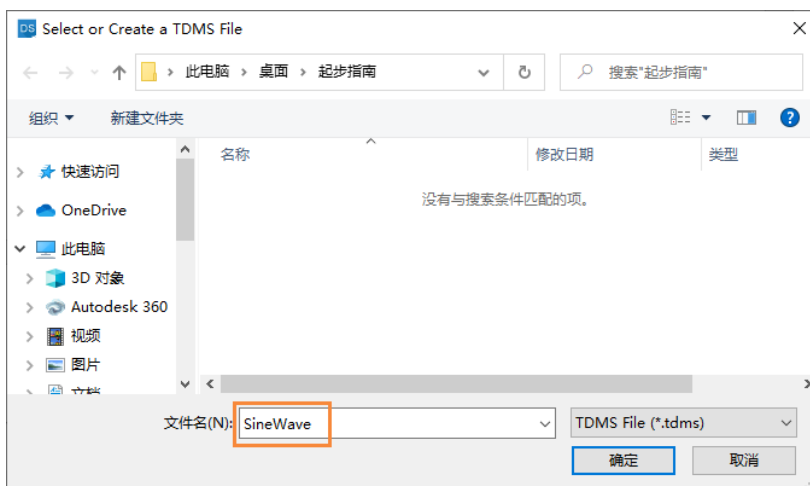
比较简单，所以没有使用动态修改参数功能。

波形录制

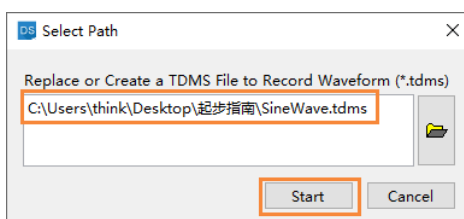
点击软件工具栏的  按钮，弹出录制波形配置界面，如下图所示：



点击路径配置按钮，弹出路径选择对话框，选好存储路径后，配置好波形文件的名称，如下图所示：



点击“确定”按钮完成配置，此时录制波形配置界面配置完成，如下图所示：



点击“Start”按钮，录波功能正式启动，此时我们可以看到界面上的波形录制按钮变成红色的，如下图所示：



在这个录波状态下，程序运行过程中，模型文件中 Scope 模块配置的所有的有效信号均会实时记录下来。

注：录制的波形文件为*.tdms 格式的，具体如何查看波形文件以及波形文件格式转换，

请参考 EasyGo DeskSim 的帮助文档中的工具箱说明部分。

暂停程序

点击软件工具栏的按钮即可随时暂停程序。暂停后，我们可以看到界面
上的暂停按钮变成红色的，如下图所示：



注意：

- ◆ 暂停功能只是暂停主界面的波形更新或数值显示控件的数据更新，便于用户查看暂停瞬时前的一段时间的数据或波形；
- ◆ 暂停功能对实时仿真机中的程序不影响；
- ◆ 暂停功能对波形录制也不影响，上下位机的数据交互始终没有间断；

停止程序

点击软件工具栏的按钮即可停止程序。

保存项目文件

全部运行完成后，为了便于下一次打开直接使用，我们可以对上述所有的配置操作进行保存管理，保存成 DeskSim 软件专属的项目文件 (*.desksim)。

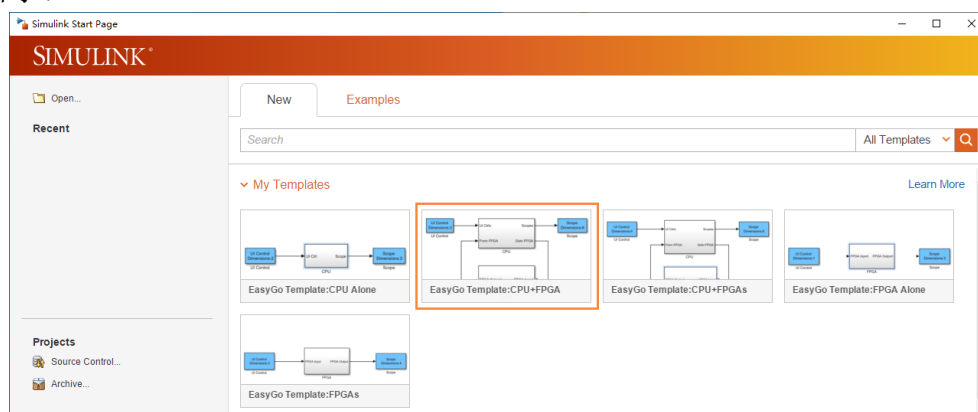
点击工具栏中的保存按钮，选择路径，修改名称，即可完成保存。下次打开时，可以点击软件工具栏中的打开按钮，载入以前保存的项目文件，或者直接双击项目文件，直接打开软件并载入项目文件。

Template 2: CPU + FPGA

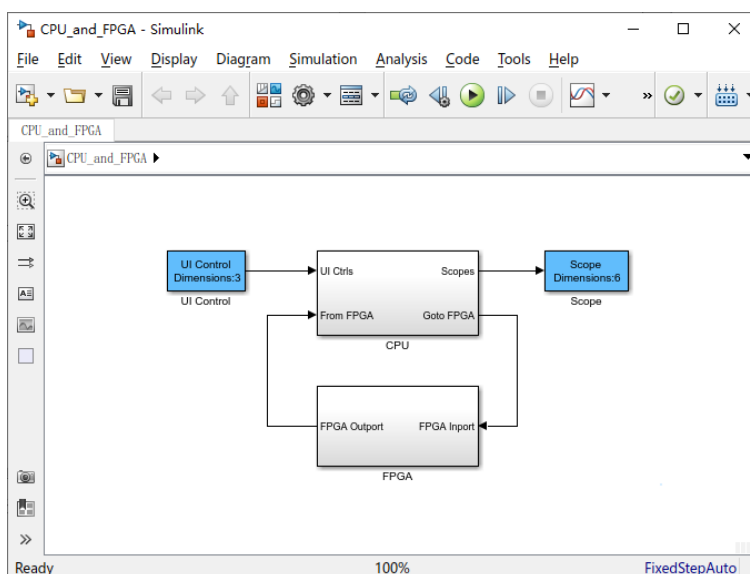
CPU + FPGA 模型是利用 EasyGo Real-Time 函数库建立用于 CPU 和单卡 FPGA 联合仿真的实时模型框架,即在实际仿真过程中使用 CPU 和单卡 FPGA 进行仿真,并将仿真结果上传至上位机观察或交互至工业通讯端口。

准备程序

打开 Matlab 软件, 点击启动 Simulink, 在 “Simulink Start Page” 界面中点击 “EasyGo Template: CPU + FPGA” 创建出 CPU + FPGA 的框架模型, 如下图所示:



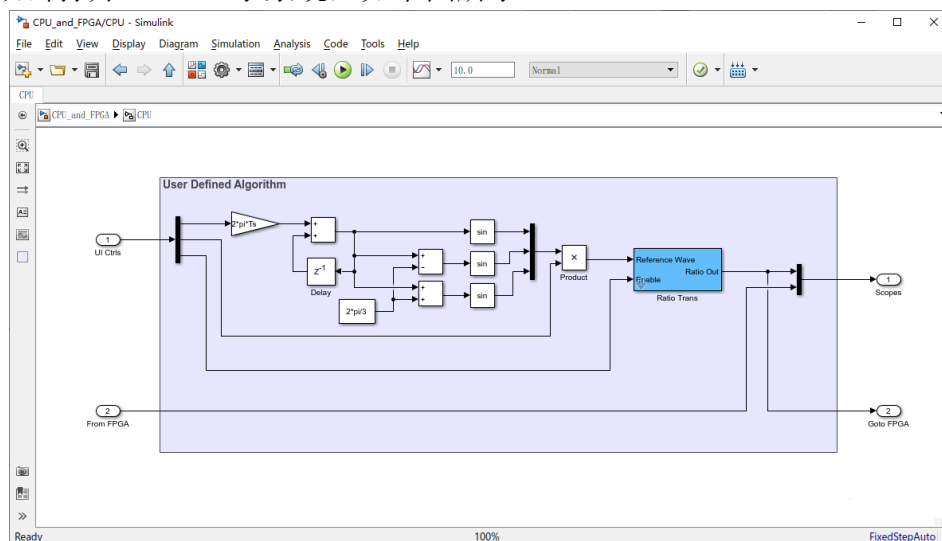
新建出来的原始模型是未保存的, 我们将其保存成 “CPU_and_FPGA.slx”, 如下图所示:



由于是 CPU + FPGA 联合仿真模型仿真程序, 所以核心代码在 “CPU” 子系

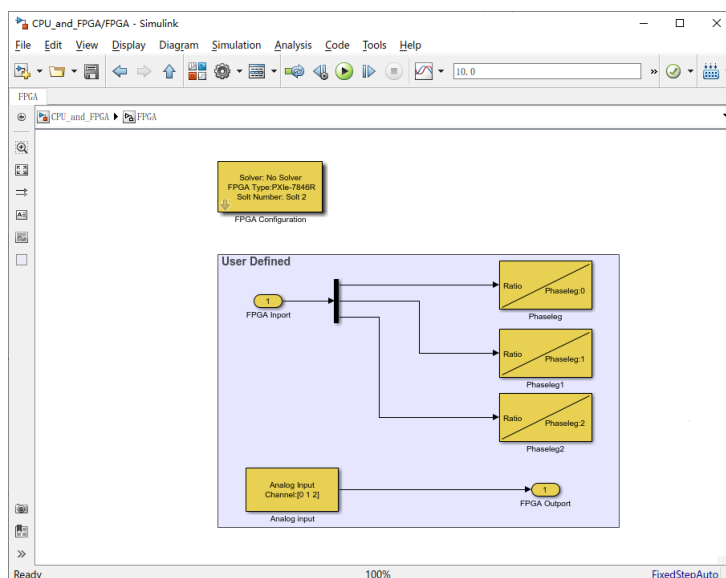
统和 FPGA 子系统里面。

双击打开“CPU”子系统，如下图所示：



在“CPU”子系统中我们可以看到是一个简单的三相正弦波参考波生成的程序，同时附带上传部分 FPGA 数据，

双击打开“FPGA”子系统，如下图所示：

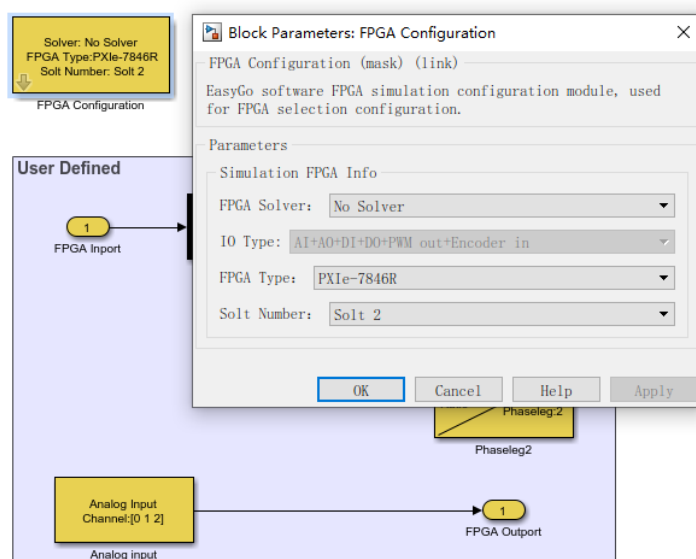
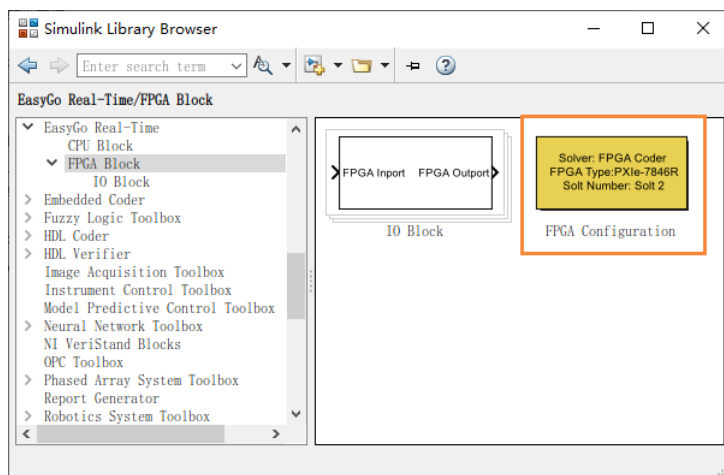


在“FPGA”子系统中我们可以看到 FPGA 程序接收到参考波后生成 PWM 波的程序，同时附带三路模拟信号采集（部分模块的详细说明可参考 EasyGo RealTime Block 帮助文档）。

注意：

- ◆ FPGA 子系统中一个 FPGA 主程序只能在一个 FPGA 板卡上运行，如果 FPGA 程序比较复杂需要用到多块 FPGA 板卡时，可以参考“CPU + FPGAs”或“FPGAs”起步说明。

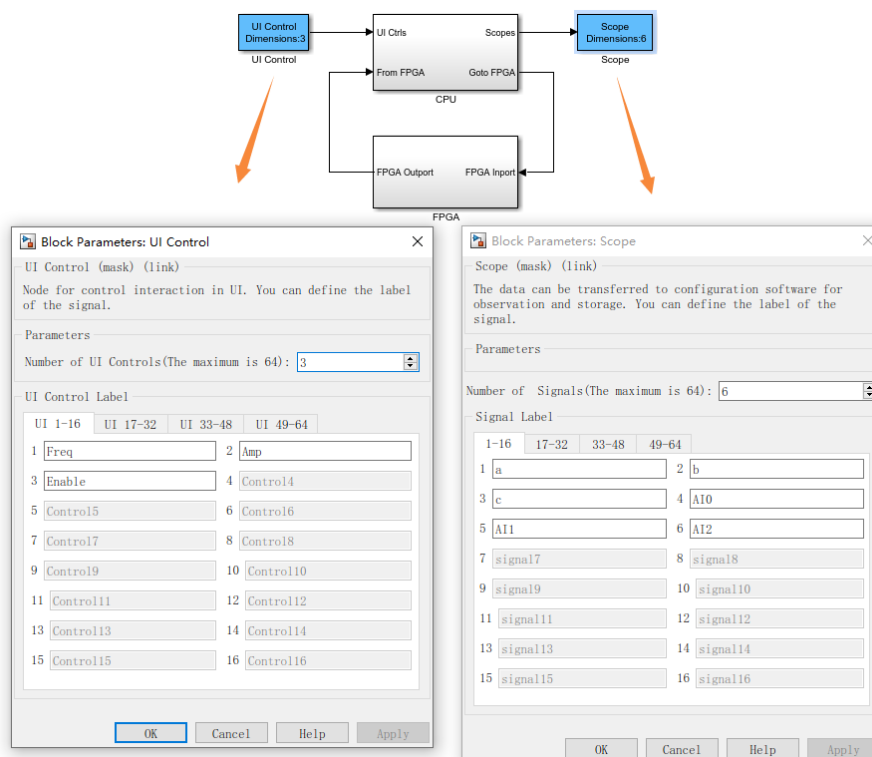
- ◆ 每一个 FPGA 板卡都必须配有一个“FPGA Configuration”模块，如下图所示，在该模块中用户需要根据自己实际的硬件信息及 FPGA 算法的用途进行配置，详细说明可参考 EasyGo RealTime Block 帮助文档。



返回主程序，我们分别打开输入端“UI Ctrl1”和输出端“Scope”的配置窗口，如下图所示：这里我们把所有需要在 DeskSim 软件界面控制的信号全部配置在“UI Control”模块中，把所有需要在 DeskSim 软件界面上观察的信号全部配置在“Scope”模块中，同时在连入子系统后的“UI Control”的信号根据程序的需要解绑连入程序中，将程序输出的信号根据需要按顺序捆绑在一起输出至“Scope”模块中。

注意：程序中所有与主界面交互的数据必须通过“UI Control”模块和“Scope”模块进行交互，程序中存在的其他的输入输出信号均不支持在 DeskSim 软件上进行控制或显示，

且同一个程序有且最多支持一个“UI Control”模块或“Scope”模块。（通讯模块使用方式同理，详细说明可参考 EasyGo RealTime Block 帮助文档）

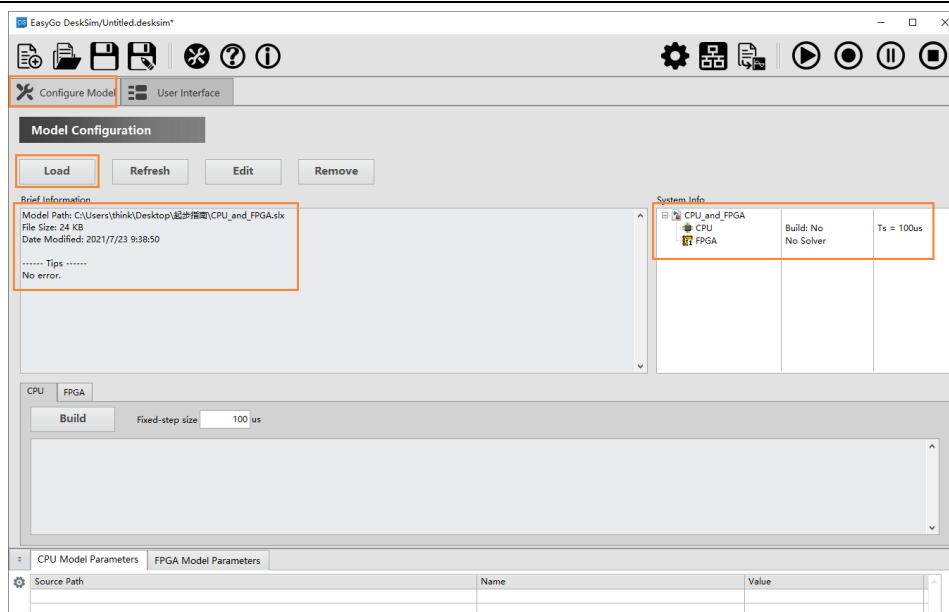


由此可知，整个程序实现的功能是输入端输入幅值、频率及参考波的使能信号，控制三相正弦波参考波，输出端为参考波的每个计算周期的瞬时值及 FPGA 采集到的 AI 信号。

注：用户可以根据自己的 CPU 和 FPGA 上的算法修改“CPU”子系统和“FPGA”子系统中的代码，以及对应修改输入输出端口的信号名称及数量。

导入程序

打开 EasyGo DeskSim 软件，在 Configure Model 界面中点击“Load”按钮载入程序文件；



载入文件后，软件会分析得到程序的基本信息。

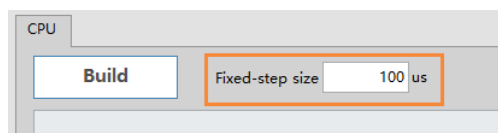
注意：

- ◆如果模型中存在错误，用户需要把错误信息全部处理掉，才能进行实时运行；
- ◆如果需要修改或者打开模型文件，可以点击“Edit”按钮打开模型文件进行修改或查看；
- ◆如果用户载入的模型文件发生了修改，在界面中可以点击“Refresh”按钮进行更新。

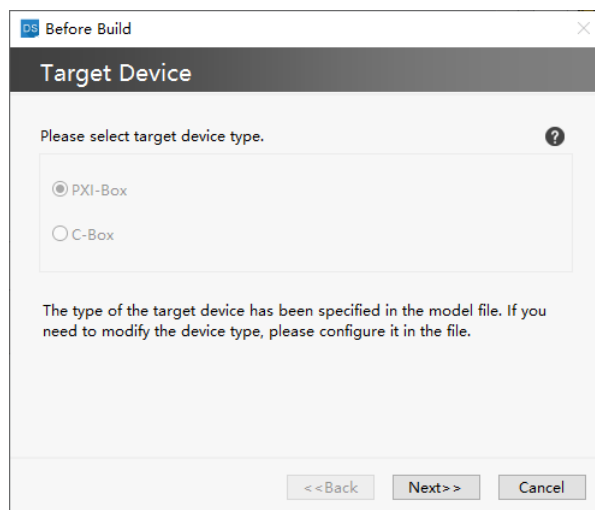
仿真配置

CPU 模型编译

针对 CPU 模型，我们需要对 CPU 模型进行编译操作，如下图所示，我们先设置程序运行的步长，如 100us，即 10k 的控制频率；



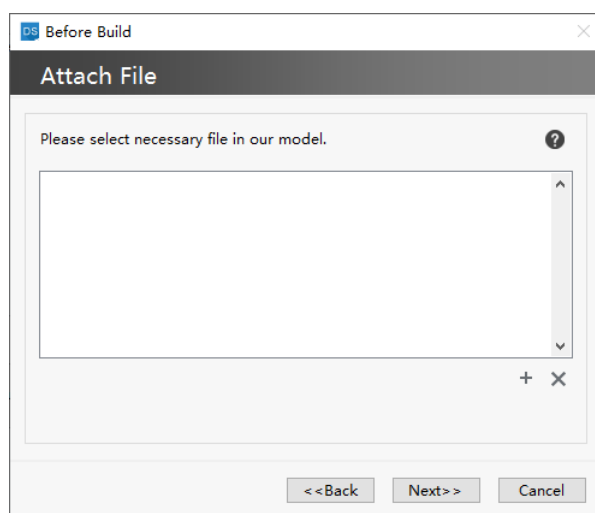
步长设置完成后，我们开始编译 CPU 模型，点击界面上的“Build”按钮，会弹出 Before Build 窗口，供用户对实际的编译操作进行与配置。如下图所示：



在“Target Device”界面中，提示用户确认硬件信息，用户可根据实际需求修改硬件类型。

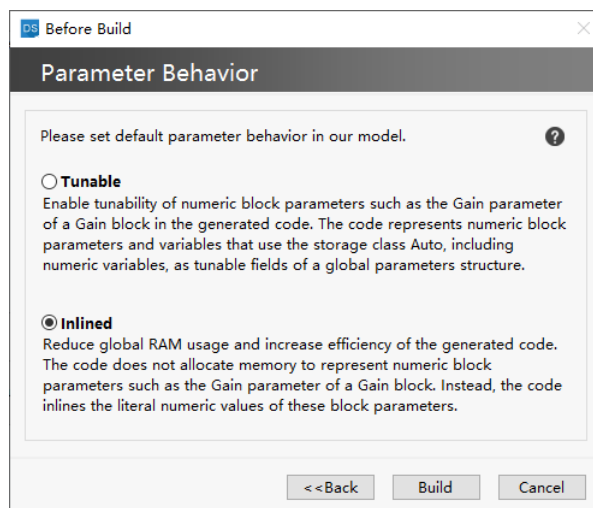
注意：当用户的模型文件中有较为明确的硬件信息，则该页面中的硬件选择为禁用的，如果用户需要修改模型的硬件信息，用户需要在原始文件中进行修改，并重新载入；当用户的模型文件中没有明确的硬件信息，则用户可以直接在该页面中进行硬件信息选择。

由于我们使用的模型中有明确的硬件类型，所以这一步我们可以跳过，点击“Next>>”进入下一步配置，如下图所示：



在“Attach File”页面中，用户可以添加在模型初始化过程中需要用到的文件，便于程序自动检测文件中所设置的变量；用户可通过右下角的添加或删除按钮进行操作。

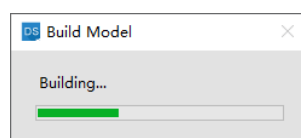
由于我们使用的 CPU 模型比较简单，没有用到相关的文件，所以我们跳过这一步，点击“Next>>”进入下一步配置，如下图所示：



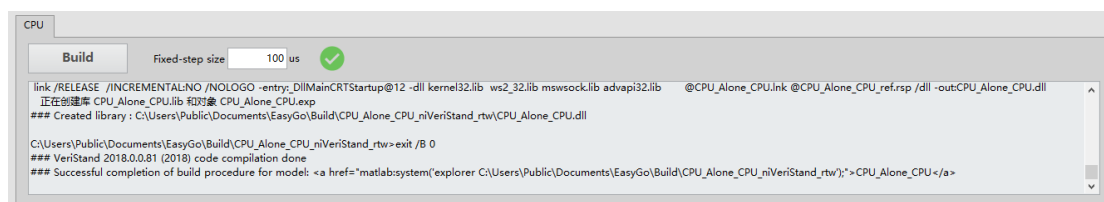
在“Parameter Behavior”页面中，用户可以设置是否支持在线调参功能：

- ◆ Tunable: 可调模式，是指程序中的一些常量或者增益的值是可以在线实时调整的，这个模式下程序运行时间会较长，效率较低。
- ◆ Inline: 不可调模式，是指程序中的一些常量或者增益的值是不可以在线实时调整的，这个模式下程序运行时间会较短，效率更高。

由于我们使用的 CPU 模型比较简单，我们直接选择“Inline”模式，点击“Build”按钮，进入模型编译操作，如下图所示：该过程会持续几分钟时间（具体与模型的复杂程度有关）；



编译完成后，会提示编译结果  编译成功，如下图所示：



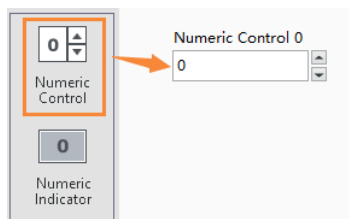
FPGA 模型初始化

对于使用了 FPGA 模型的程序，在 FPGA 模型为电力电子算法模型时，用户可根据实际需要对电力电子算法模型的初始状态进行处理，对于我们现在的 FPGA 模型较为简单，且没有使用电力电子算法模型，所以此处略。

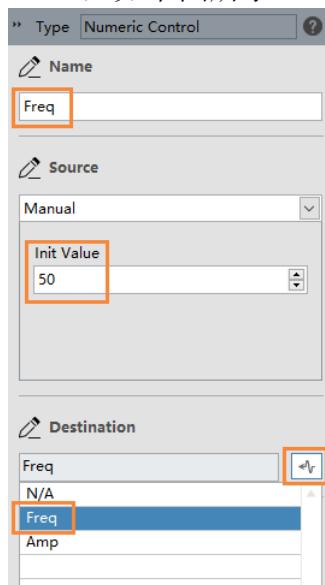
添加自定义界面

添加 UI 界面的控制输入

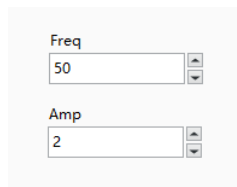
DeskSim 主界面切换至 UI Interface 界面中，在界面左侧的控件选板中，左键点击“Numeric Control”，并将其拖拽到中间的空白编辑区域释放，即完成一个 Numeric Control 控件的添加，如下图所示：



添加完成之后，点击选中这个控件，界面右侧栏会显示这个控件的配置内容，可选择信号绑定的对象（程序的所有输入端），修改名称和初始值；点击信号选择按钮，在弹出的信号列表中选择需要绑定的信号，如“Freq”，选中后，控件名称会同步更新为信号名称，用户可以使用信号名称或手动修改名称，设置完成后，我们对设置初始参数为“50”，如下图所示：

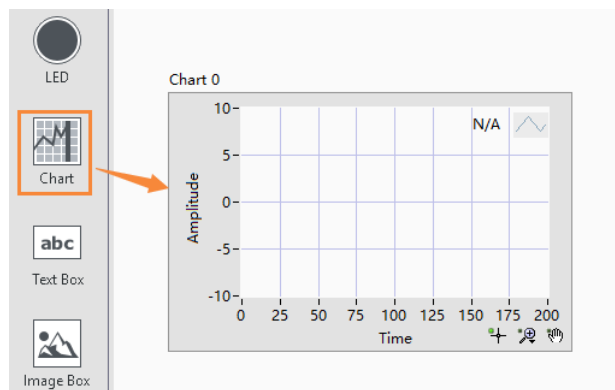


同样的方式，将程序的两个输入端口都绑定到界面的控件之后可以得到：

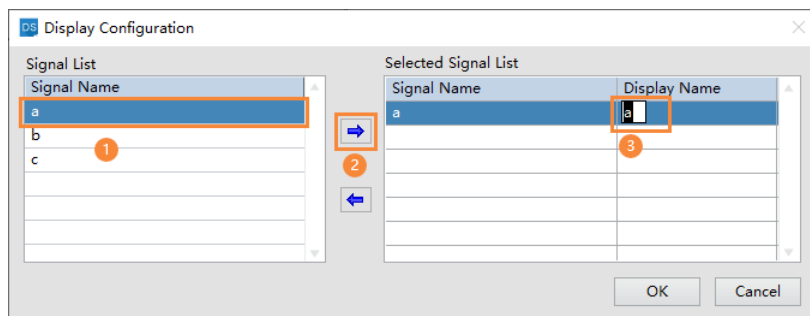
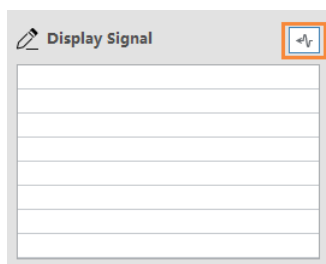


添加 UI 界面的波形显示

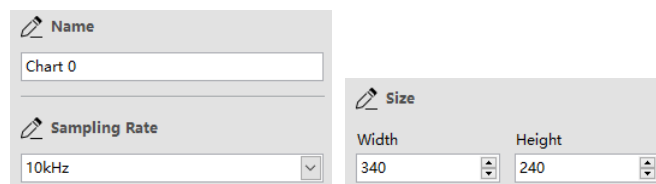
在界面左侧的控件选板中，左键点击“Chart”，并将其拖拽到中间的空白编辑区域释放，即完成一个 Chart 控件的添加，如下图所示：



选中 Chart 控件，在右侧显示的配置区域中可以选中提那就显示的信号源（程序的所有输出信号均可显示）。

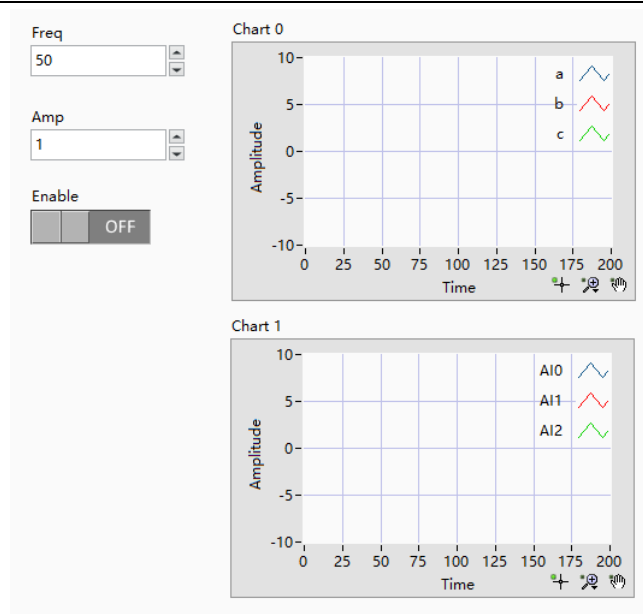


并可以修改 Chart 的名称、采样率和大小。



同样的方式，添加三路 AI 的波形显示。

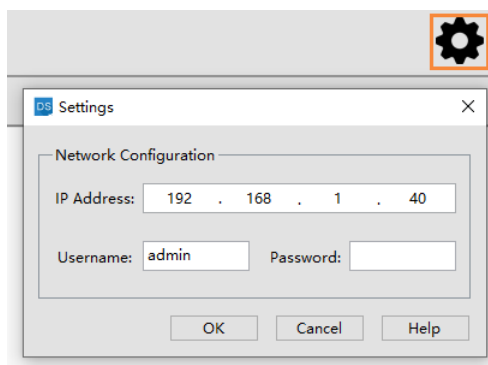
配置完成后，可以得到最终的自定义界面，如下图所示：



程序运行

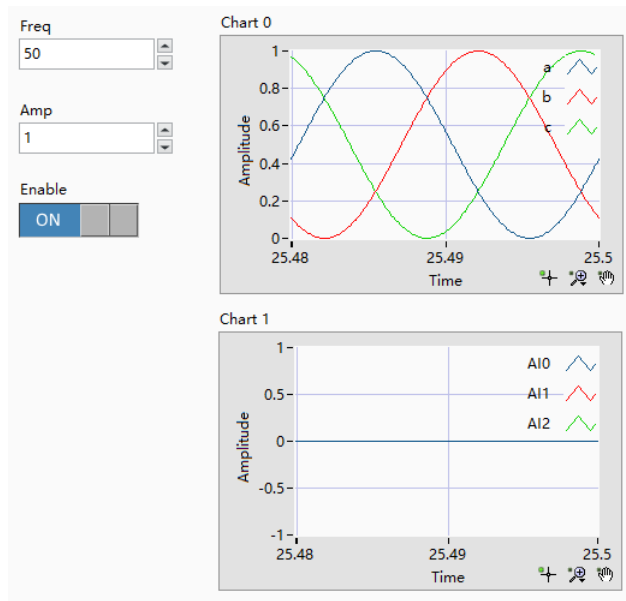
配置仿真机的网络信息

点击 DeskSim 软件工具栏的  按钮，即可打开仿真机的网络配置窗口，设置好 IP 地址、端口号等信息。



运行程序

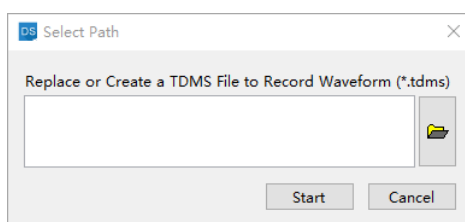
点击软件工具栏的  按钮即可运行程序（目标仿真机上需要正确激活才能正常使用）。



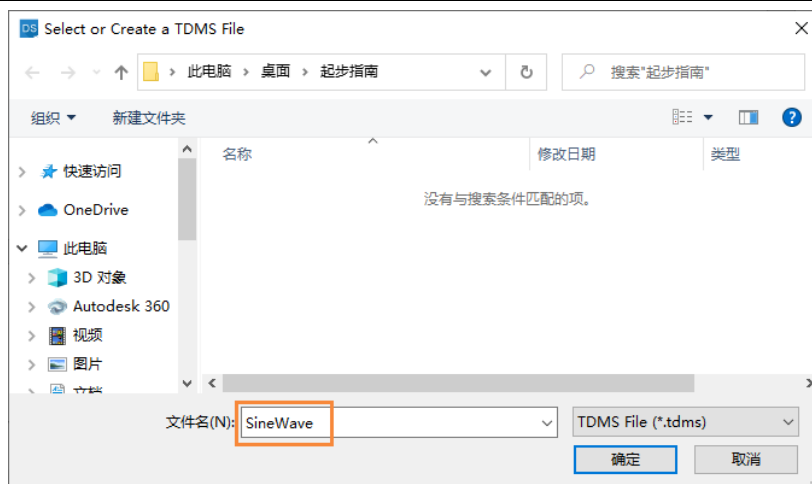
我们可以实时调整波形的幅值和频率来观察实际波形的变化情况。由于实际的 AI 通道是悬空的，所以采集到的信号均为 0V，我们可以外接信号来观察采集信号是否正确。具体现象不再详述。同时运行过程中还可以动态修改附加区中 CPU Model Parameter、FPGA Model Parameter 的参数，具体参考 EasyGo DeskSim 软件的帮助文档中的相关说明，由于 Template 模型比较简单，所以没有使用动态修改参数功能。

波形录制

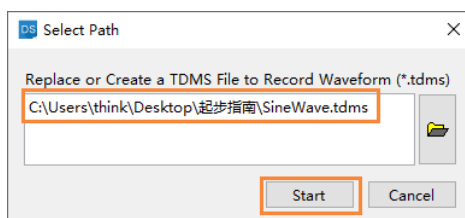
点击软件工具栏的  按钮，弹出录制波形配置界面，如下图所示：



点击路径配置按钮，弹出路径选择对话框，选好存储路径后，配置好波形文件的名称，如下图所示：



点击“确定”按钮完成配置，此时录制波形配置界面配置完成，如下图所示：



点击“Start”按钮，录波功能正式启动，此时我们可以看到界面上的波形录制按钮变成红色的，如下图所示：



在这个录波状态下，程序运行过程中，模型文件中 Scope 模块配置的所有的有效信号均会实时记录下来。

注：录制的波形文件为*.tdms 格式的，具体如何查看波形文件以及波形文件格式转换，请参考 EasyGo DeskSim 的帮助文档中的工具箱说明部分。

暂停程序

点击软件工具栏的  按钮即可随时暂停程序。暂停后，我们可以看到界面上的暂停按钮变成红色的，如下图所示：



注意：

◆ 暂停功能只是暂停主界面的波形更新或数值显示控件的数据更新，便于用户查看暂停瞬时前的一段时间的数据或波形；

- ◆ 暂停功能对实时仿真机中的程序不影响；
- ◆ 暂停功能对波形录制也不影响，上下位机的数据交互始终没有间断；

停止程序

点击软件工具栏的按钮即可停止程序。

保存项目文件

全部运行完成后，为了便于下一次打开直接使用，我们可以对上述所有的配置操作进行保存管理，保存成 DeskSim 软件专属的项目文件 (*.desksim)。

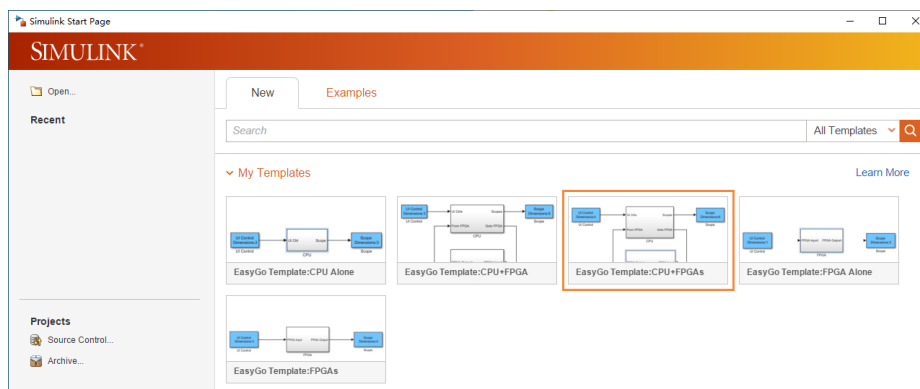
点击工具栏中的保存按钮，选择路径，修改名称，即可完成保存。下次打开时，可以点击软件工具栏中的打开按钮，载入以前保存的项目文件，或者直接双击项目文件，直接打开软件并载入项目文件。

Template 3: CPU + FPGAs

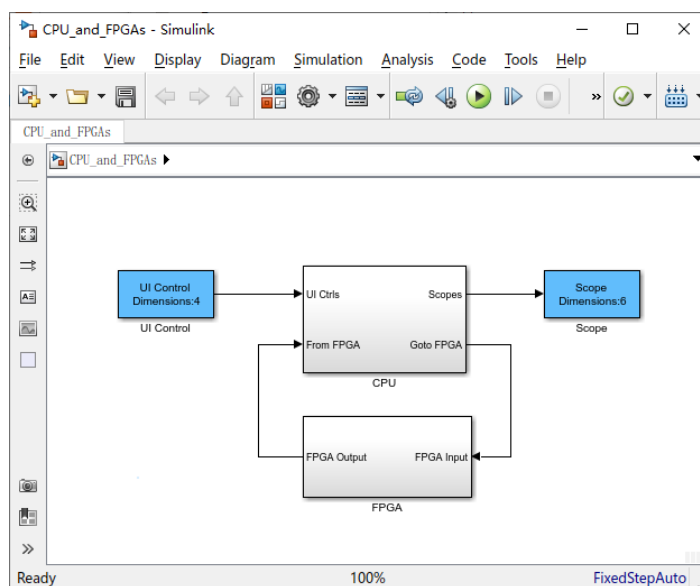
CPU + FPGAs 模型是利用 EasyGo Real-Time 函数库建立用于 CPU 和多块 FPGA 联合仿真的实时模型框架，即在实际仿真过程中使用 CPU 和多块 FPGA 进行仿真，并将仿真结果上传至上位机观察或交互至工业通讯端口。

准备程序

打开 Matlab 软件，点击启动 Simulink，在“Simulink Start Page”界面中点击“EasyGo Template: CPU + FPGAs”创建出 CPU + FPGAs 的框架模型，如下图所示：



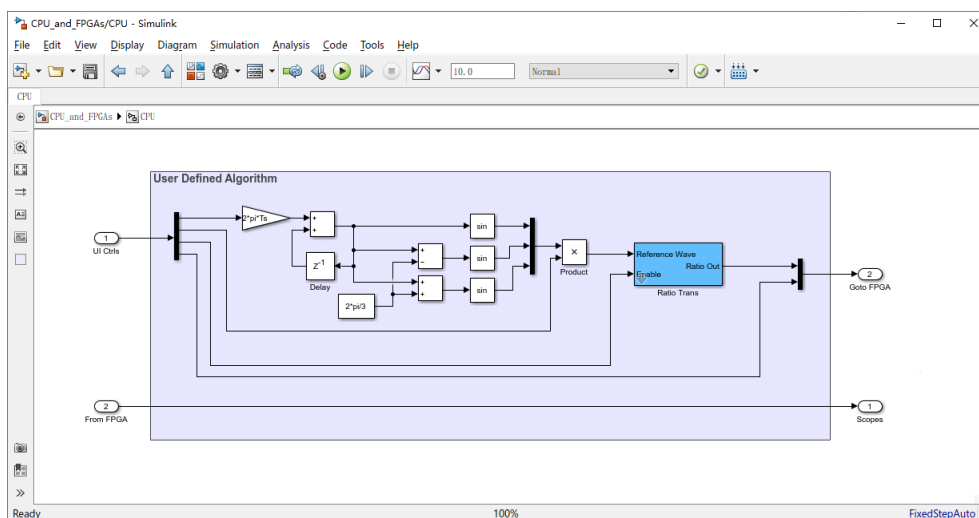
新建出来的原始模型是未保存的，我们将其保存成“CPU_and_FPGAs.slx”，如下图所示：



由于是 CPU + FPGAs 联合仿真模型仿真程序，所以核心代码在“CPU”子系

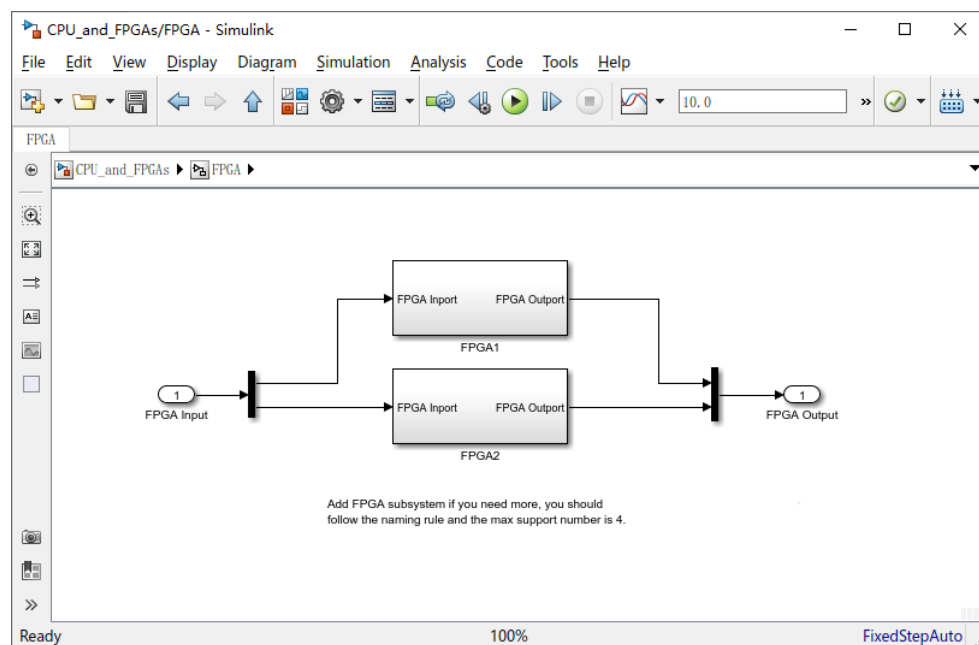
统和“FPGA”子系统里面。

双击打开“CPU”子系统，如下图所示：



在“CPU”子系统中我们可以看到是一个简单的三相正弦波参考波生成的程序，同时附带上传部分 FPGA 数据，

双击打开“FPGA”子系统，如下图所示：



由于我们的程序是 CPU 与多块 FPGA 板卡进行联合仿真的程序，所以在 FPGA 子系统中又包含多个子系统，如上图中显示的：FPGA1、FPGA2，即代表着两块不同的 FPGA 板卡，每个 FPGA 板卡上都有自己的程序。

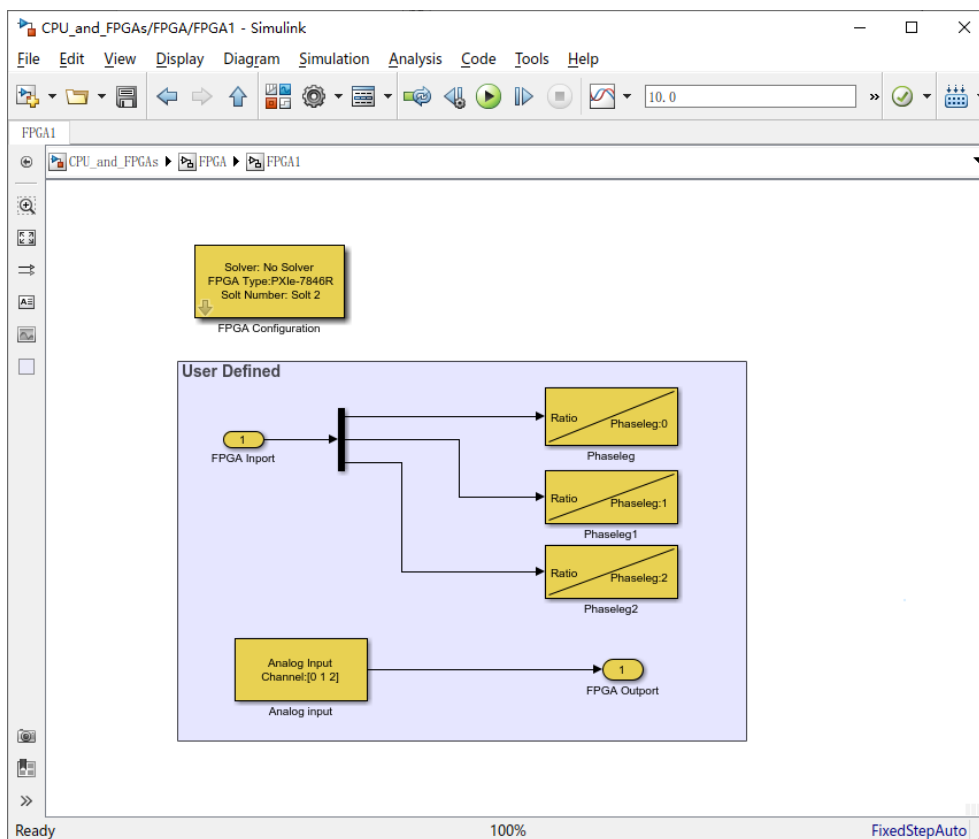
用户在使用多个 FPGA 板卡联合仿真时，模型搭建方面需要注意以下几点：

◆ 如果用户使用多块 FPGA 板卡仿真时，一定要将所有板卡全部放在这个

“FPGA” 子系统里面；且每块板卡上的程序都需要独立在一个子系统里面，子系统命名规则为 “FPGA+序号”；

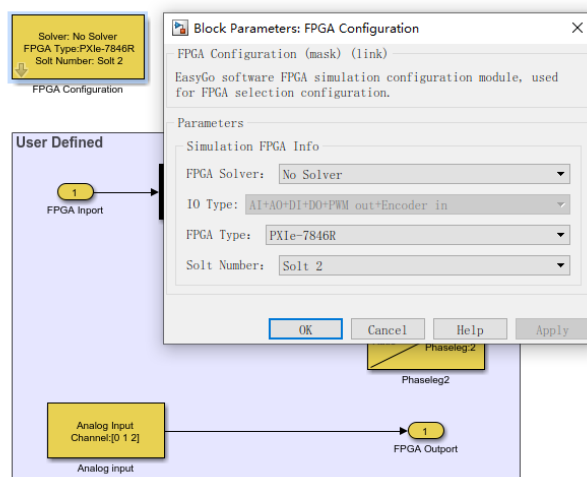
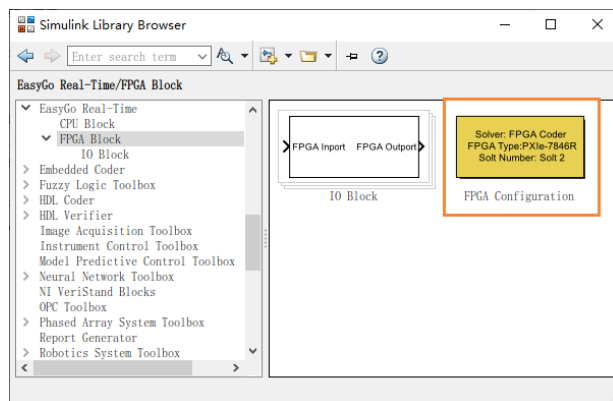
- ◆ 所有板卡的子系统的输入输出信号最终需要全部合并在一个端口上与 CPU 交互或直接与顶层交互模块交互（在没有 CPU 模型的情况下）；

双击打开 “FPGA1” 子系统，如下图所示：

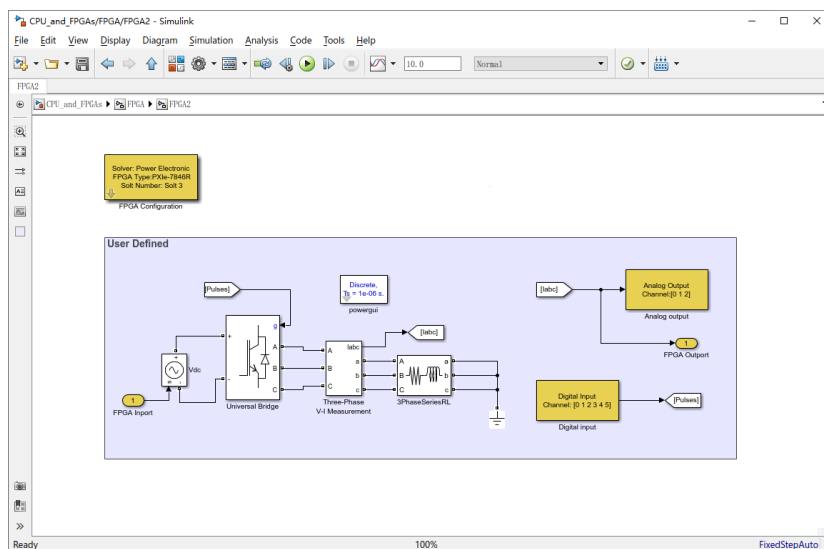


在 “FPGA1” 子系统中我们可以看到 FPGA 程序接收到参考波后生成 PWM 波的程序,同时附带三路模拟信号采集(部分模块的详细说明可参考 EasyGo RealTime Block 帮助文档)。

注意：每一个 FPGA 板卡都必须配有一个 “FPGA Configuration” 模块，如下图所示，在该模块中用户需要根据自己实际的硬件信息及 FPGA 算法的用途进行配置，详细说明可参考 EasyGo RealTime Block 帮助文档。



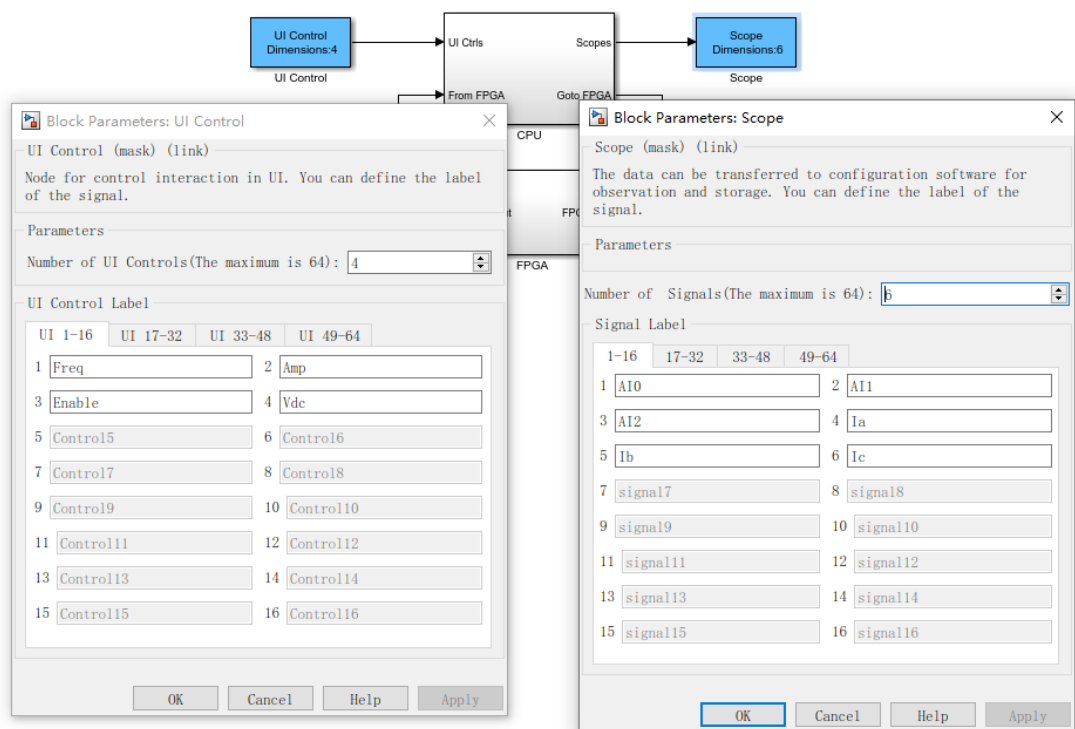
双击打开“FPGA2”子系统，如下图所示：



在“FPGA2”子系统中我们可以看到 FPGA 程序是一个电力电子算法，同时附带硬件 IO 上的数字信号采集及模拟信号输出（部分模块的详细说明可参考 EasyGo RealTime Block 帮助文档）。

返回主程序，我们分别打开输入端“UI Ctrl”和输出端“Scope”的配置窗口，如下图所示：这里我们把所有需要在 DeskSim 软件界面控制的信号全部配置在“UI Control”模块中，把所有需要在 DeskSim 软件界面上观察的信号全部配置在“Scope”模块中，同时在连入子系统后的“UI Control”的信号根据程序的需要解绑连入程序中，将程序输出的信号根据需要按顺序捆绑在一起输出至“Scope”模块中。

注意：程序中所有与主界面交互的数据必须通过“UI Control”模块和“Scope”模块进行交互，程序中存在的其他的输入输出信号均不支持在 DeskSim 软件上进行控制或显示，且同一个程序有且最多支持一个“UI Control”模块或“Scope”模块。（通讯模块使用方式同理，详细说明可参考 EasyGo RealTime Block 帮助文档）

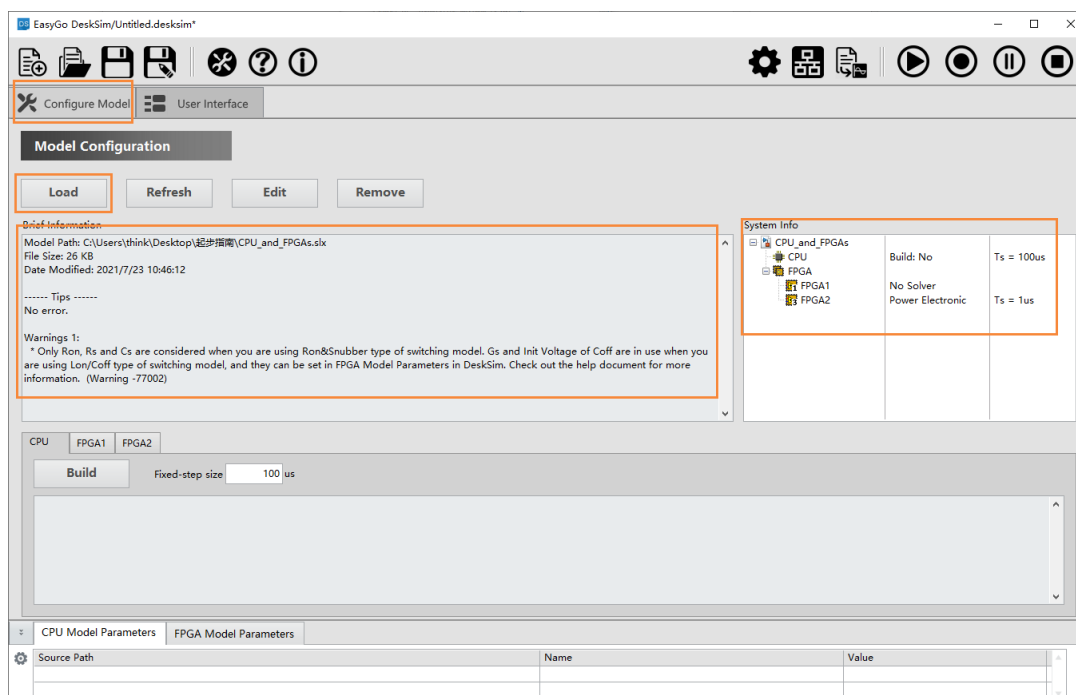


由此可知，整个程序实现的功能是输入端输入幅值、频率、参考波的使能信号以及 Vdc 信号，控制三相正弦波参考波，输出端为 FPGA 采集到的 AI 信号及电力电子算法中三路电流信号。

注：用户可以根据自己的 CPU 和 FPGA 上的算法修改“CPU”子系统和“FPGA”子系统中的代码，以及对应修改输入输出端口的信号名称及数量。

导入程序

打开 EasyGo DeskSim 软件，在 Configure Model 界面中点击“Load”按钮
载入程序文件；



载入文件后，软件会分析得到程序的基本信息。

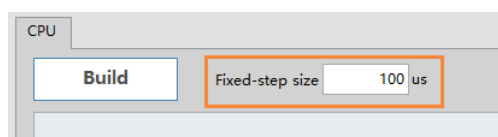
注意：

- ◆如果模型中存在错误，用户需要把错误信息全部处理掉，才能进行实时运行；
- ◆如果需要修改或者打开模型文件，可以点击“Edit”按钮打开模型文件进行修改或查看；
- ◆如果用户载入的模型文件发生了修改，在界面中可以点击“Refresh”按钮进行更新。

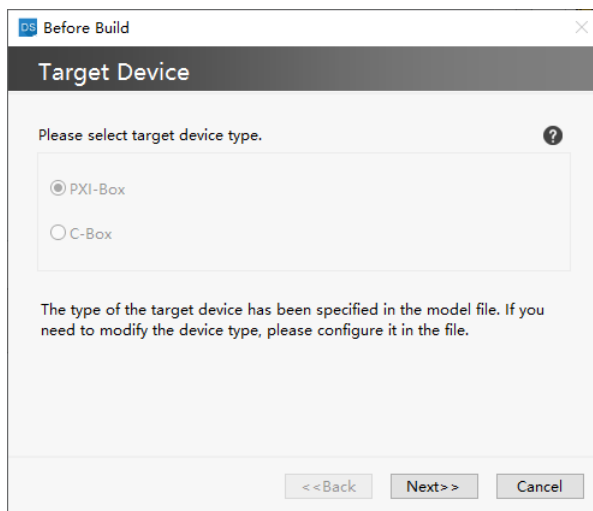
仿真配置

CPU 模型编译

针对 CPU 模型，我们需要对 CPU 模型进行编译操作，如下图所示，我们先设置程序运行的步长，如 100us，即 10k 的控制频率；



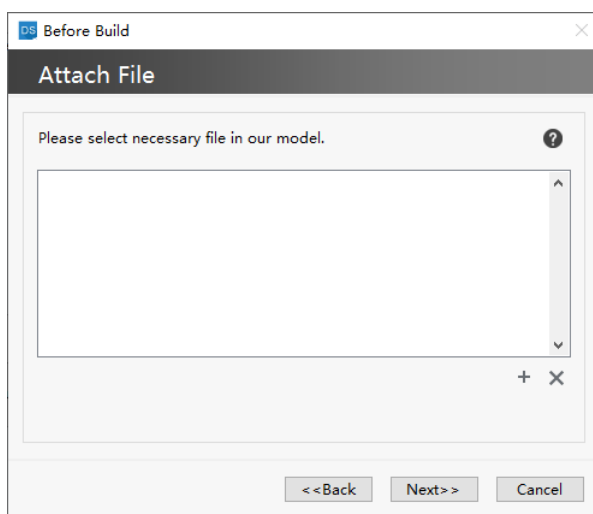
步长设置完成后，我们开始编译 CPU 模型，点击界面中的“Build”按钮，会弹出 Before Build 窗口，供用户对实际的编译操作进行与配置。如下图所示：



在“Target Device”界面中，提示用户确认硬件信息，用户可根据实际需求修改硬件类型。

注意：当用户的模型文件中有较为明确的硬件信息，则该页面中的硬件选择为禁用的，如果用户需要修改模型的硬件信息，用户需要在原始文件中进行修改，并重新载入；当用户的模型文件中没有明确的硬件信息，则用户可以直接在该页面中进行硬件信息选择。

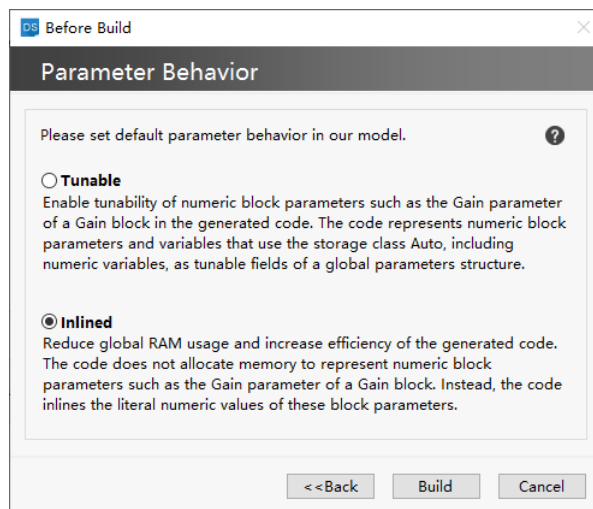
由于我们使用的模型中有明确的硬件类型，所以这一步我们可以跳过，点击“Next>>”进入下一步配置，如下图所示：



在“Attach File”页面中，用户可以添加在模型初始化过程中需要用到的文件，便于程序自动检测文件中所设置的变量；用户可通过右下角的添加或删除

按钮进行操作。

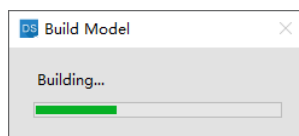
由于我们使用的 CPU 模型比较简单，没有用到相关的文件，所以我们跳过这一步，点击“Next>>”进入下一步配置，如下图所示：



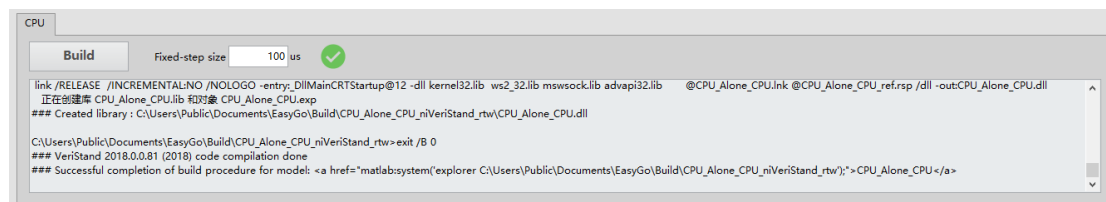
在“Parameter Behavior”页面中，用户可以设置是否支持在线调参功能：

- ◆ Tunable: 可调模式，是指程序中的一些常量或者增益的值是可以在线实时调整的，这个模式下程序运行时间会较长，效率较低。
- ◆ Inline: 不可调模式，是指程序中的一些常量或者增益的值是不可以在线实时调整的，这个模式下程序运行时间会较短，效率更高。

由于我们使用的 CPU 模型比较简单，我们直接选择“Inline”模式，点击“Build”按钮，进入模型编译操作，如下图所示：该过程会持续几分钟时间（具体与模型的复杂程度有关）；



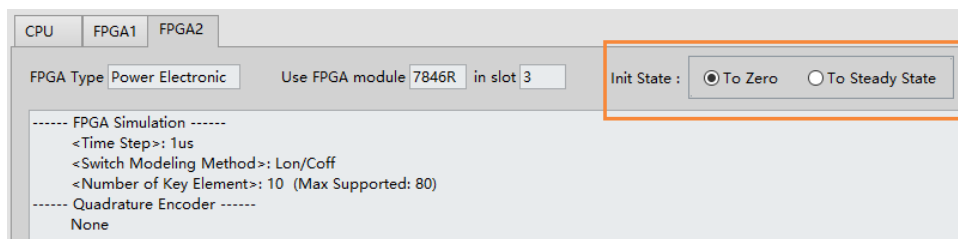
编译完成后，会提示编译结果  编译成功，如下图所示：



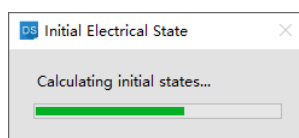
FPGA 模型初始化

对于使用了 FPGA 模型的程序，在 FPGA 模型为电力电子算法模型时，用户可

根据实际需要对电力电子算法模型的初始状态进行处理，如下图所示：



如果需要初始化操作的话，点击“Init State”勾选框中的“To Steady State”选项，之后会弹出确认对话框，确认后开始对模型进行初始化处理，该过程会持续几分钟，如下图所示：

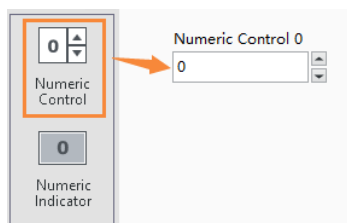


初始化完成后，模型在运行时会预先导入初始状态量进行电力电子拓扑仿真。

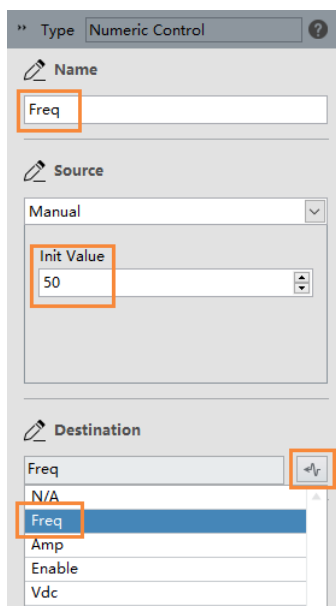
添加自定义界面

添加 UI 界面的控制输入

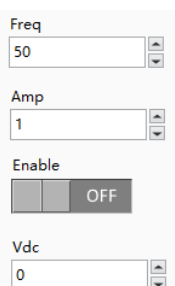
DeskSim 主界面切换至 UI Interface 界面中，在界面左侧的控件选板中，左键点击“Numeric Control”，并将其拖拽到中间的空白编辑区域释放，即完成一个 Numeric Control 控件的添加，如下图所示：



添加完成之后，点击选中这个控件，界面右侧栏会显示这个控件的配置内容，可选择信号绑定的对象（程序的所有输入端），修改名称和初始值；点击信号选择按钮，在弹出的信号列表中选择需要绑定的信号，如“Freq”，选中后，控件名称会同步更新为信号名称，用户可以使用信号名称或手动修改名称，设置完成后，我们对设置初始参数为“50”，如下图所示：

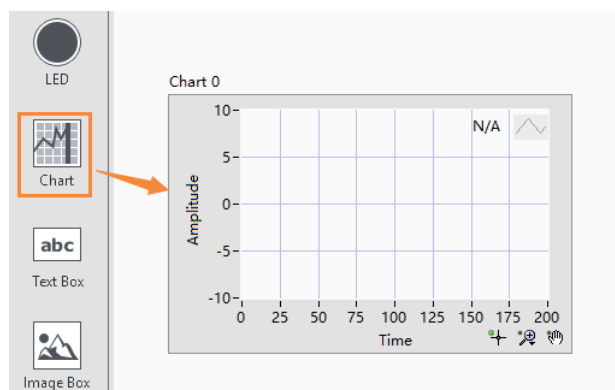


同样的方式，将程序的四个输入端口都绑定到界面的控件之后可以得到：

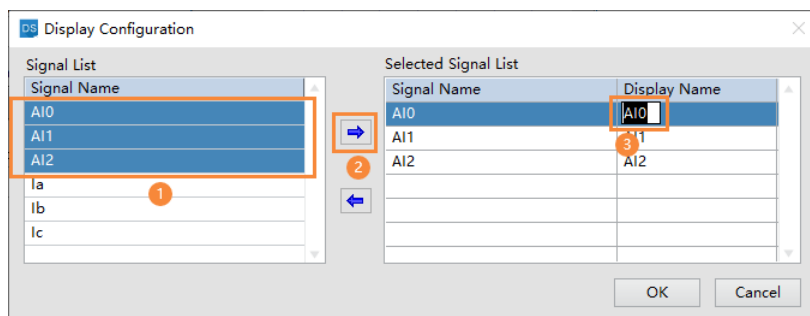
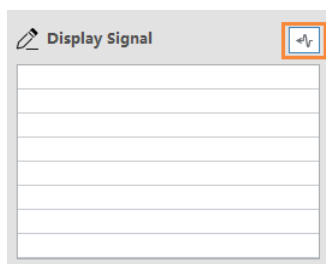


添加 UI 界面的波形显示

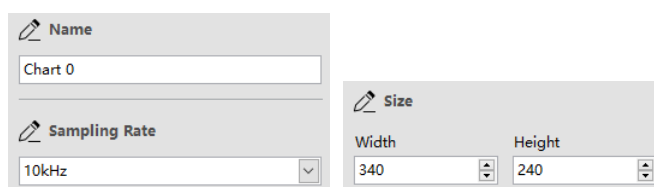
在界面左侧的控件选板中，左键点击“Chart”，并将其拖拽到中间的空白编辑区域释放，即完成一个 Chart 控件的添加，如下图所示：



选中 Chart 控件，在右侧显示的配置区域中可以选中提那就显示的信号源（程序的所有输出信号均可显示）。

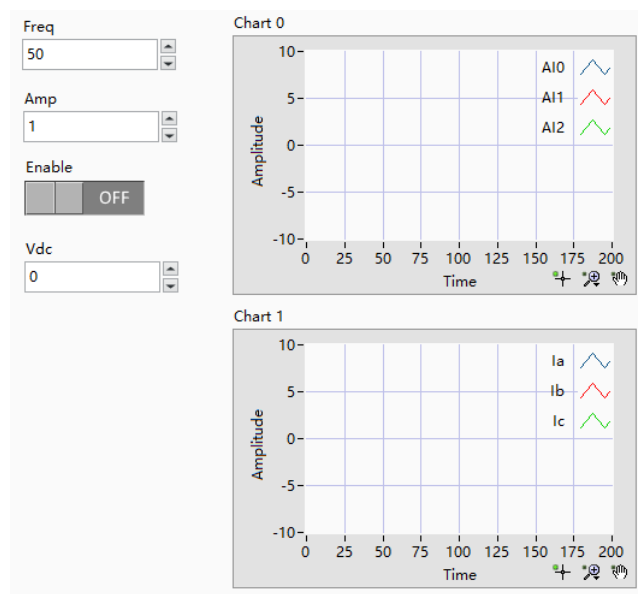


并可以修改 Chart 的名称、采样率和大小。



同样的方式，添加三路电流 Ia，Ib，Ic 的波形显示。

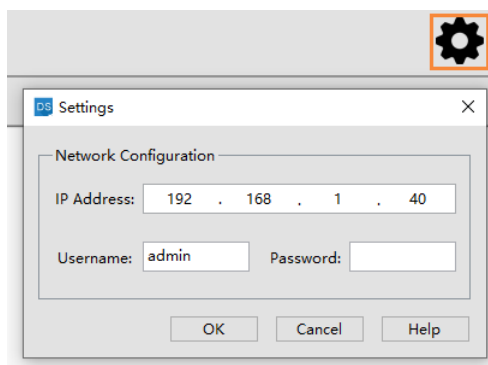
配置完成后，可以得到最终的自定义界面，如下图所示：



程序运行

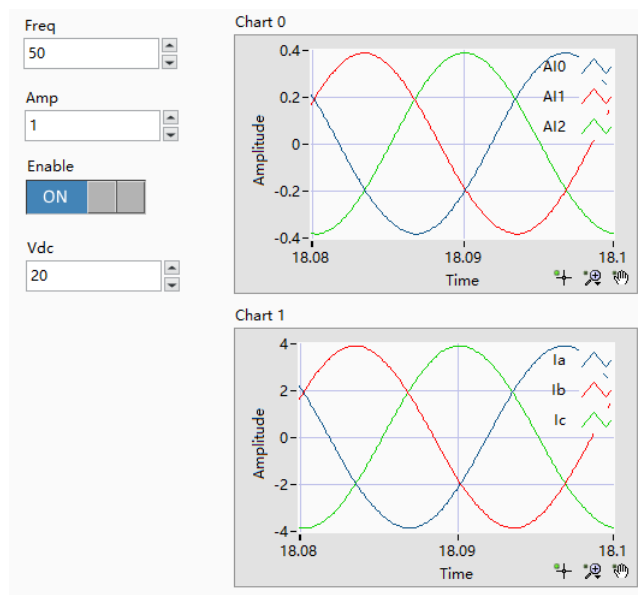
配置仿真机的网络信息

点击 DeskSim 软件工具栏的  按钮，即可打开仿真机的网络配置窗口，设置好 IP 地址、端口号等信息。



运行程序

点击软件工具栏的  按钮即可运行程序（目标仿真机上需要正确激活才能正常使用）。

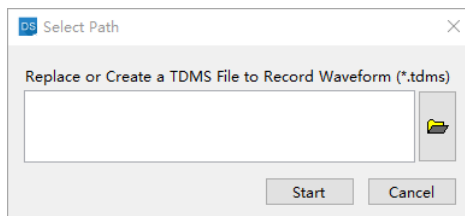


我们可以实时调整波形的幅值和频率来观察实际波形的变化情况。由于实际的 AI 通道是悬空的，所以采集到的信号均为 0V，我们可以外接信号来观察采集信号是否正确。具体现象不再详述。同时运行过程中还可以动态修改附加区中 CPU Model Parameter、FPGA Model Parameter 的参数，具体参考 EasyGo DeskSim

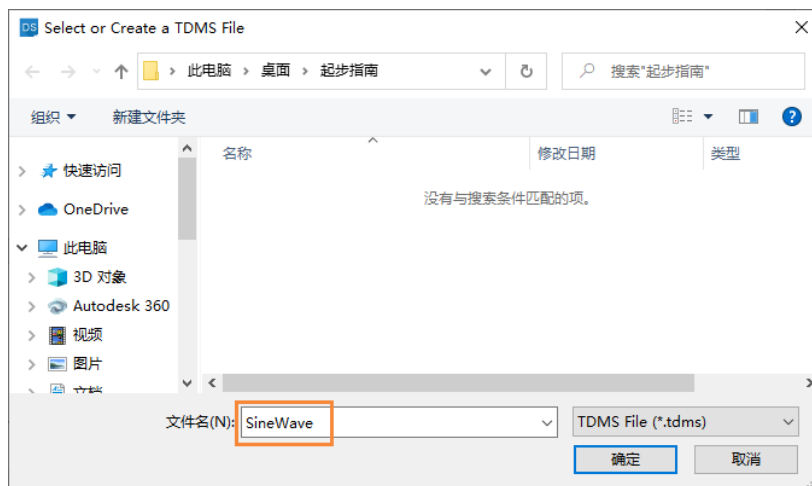
软件的帮助文档中的相关说明，由于 Template 模型比较简单，所以没有使用动态修改参数功能。

波形录制

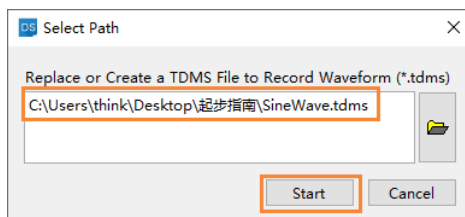
点击软件工具栏的  按钮，弹出录制波形配置界面，如下图所示：



点击路径配置按钮，弹出路径选择对话框，选好存储路径后，配置好波形文件的名称，如下图所示：



点击“确定”按钮完成配置，此时录制波形配置界面配置完成，如下图所示：



点击“Start”按钮，录波功能正式启动，此时我们可以看到界面上的波形录制按钮变成红色的，如下图所示：



在这个录波状态下，程序运行过程中，模型文件中 Scope 模块配置的所有的有效信号均会实时记录下来。

注：录制的波形文件为*.tdms 格式的，具体如何查看波形文件以及波形文件格式转换，请参考 EasyGo DeskSim 的帮助文档中的工具箱说明部分。

暂停程序

点击软件工具栏的按钮即可随时暂停程序。暂停后，我们可以看到界面上的暂停按钮变成红色的，如下图所示：



注意：

- ◆ 暂停功能只是暂停主界面的波形更新或数值显示控件的数据更新，便于用户查看暂停瞬时前的一段时间的数据或波形；
- ◆ 暂停功能对实时仿真机中的程序不影响；
- ◆ 暂停功能对波形录制也不影响，上下位机的数据交互始终没有间断；

停止程序

点击软件工具栏的按钮即可停止程序。

保存项目文件

全部运行完成后，为了便于下一次打开直接使用，我们可以对上述所有的配置操作进行保存管理，保存成 DeskSim 软件专属的项目文件 (*.desksim)。

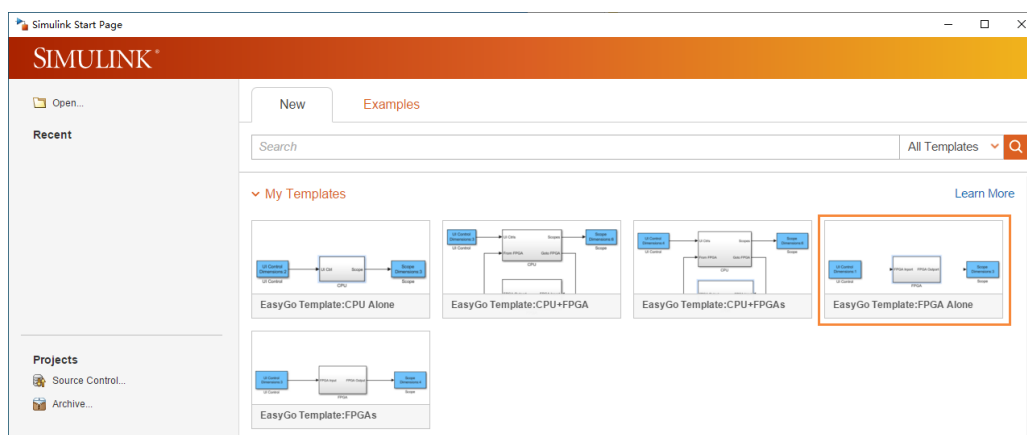
点击工具栏中的保存按钮，选择路径，修改名称，即可完成保存。下次打开时，可以点击软件工具栏中的打开按钮，载入以前保存的项目文件，或者直接双击项目文件，直接打开软件并载入项目文件。

Template 4: FPGA Alone

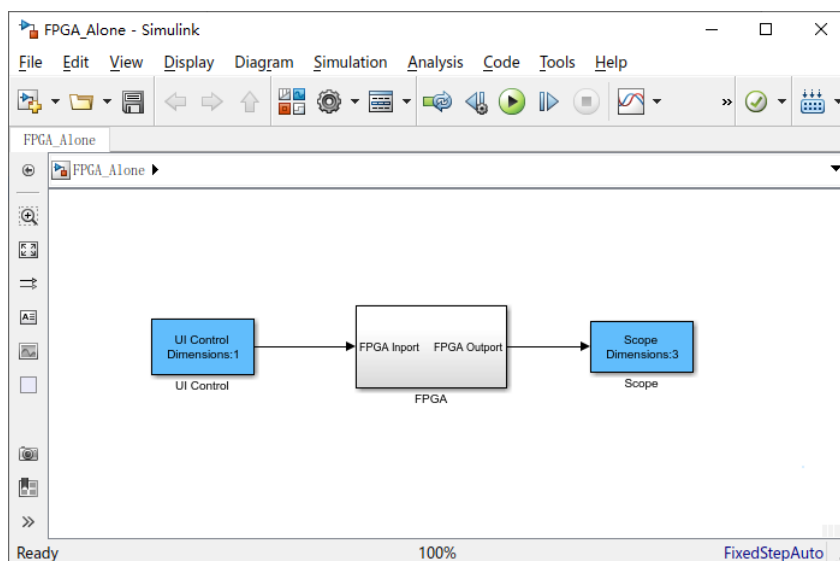
FPGA Alone 模型是利用 EasyGo Real-Time 函数库建立用于单卡 FPGA 实时模型框架，即在实际仿真过程中仅使用一块 FPGA 进行仿真，并将仿真结果上传至上位机观察或交互至工业通讯端口。

准备程序

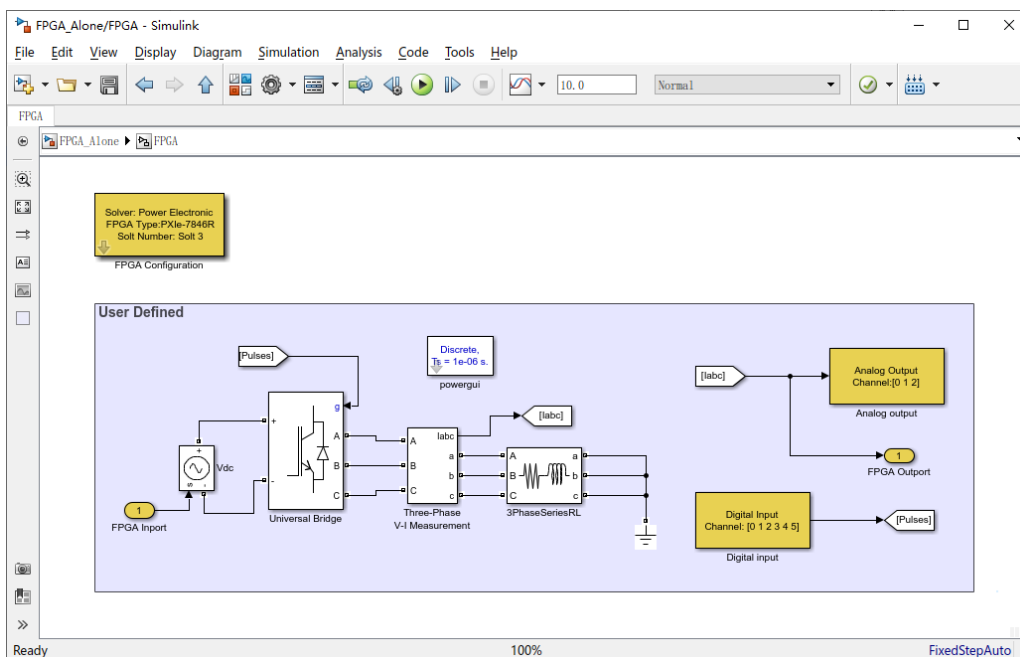
打开 Matlab 软件，点击启动 Simulink，在“Simulink Start Page”界面中点击“EasyGo Template: FPGA Alone”创建出 FPGA Alone 的框架模型，如下图所示：



新建出来的原始模型是未保存的，我们将其保存成“FPGA_Alone.slx”，如下图所示：

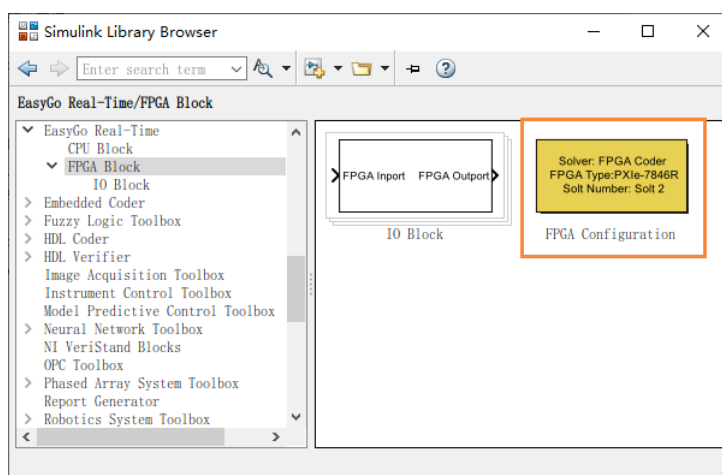


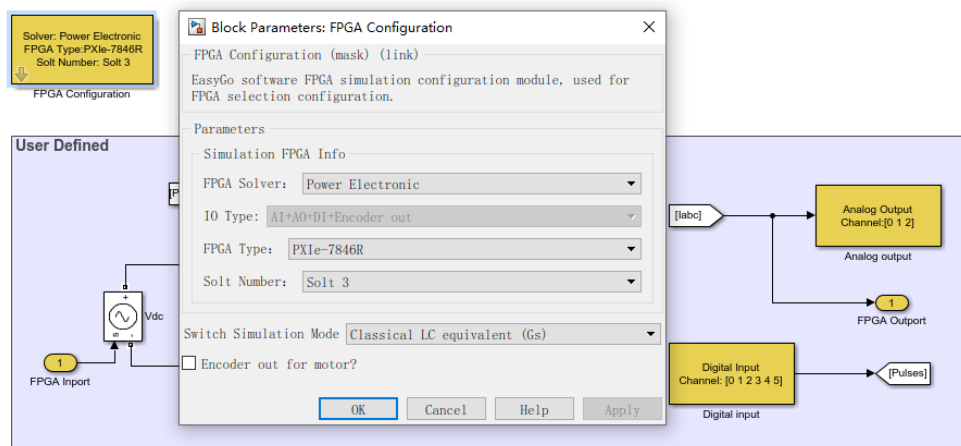
由于是单卡 FPGA 仿真模型仿真程序，所以核心代码在“FPGA”子系统里面。
双击打开“FPGA”子系统，如下图所示：



在“FPGA”子系统中我们可以看到 FPGA 程序是一个电力电子算法，同时附带硬件 IO 上的数字信号采集及模拟信号输出（部分模块的详细说明可参考 EasyGo RealTime Block 帮助文档）。

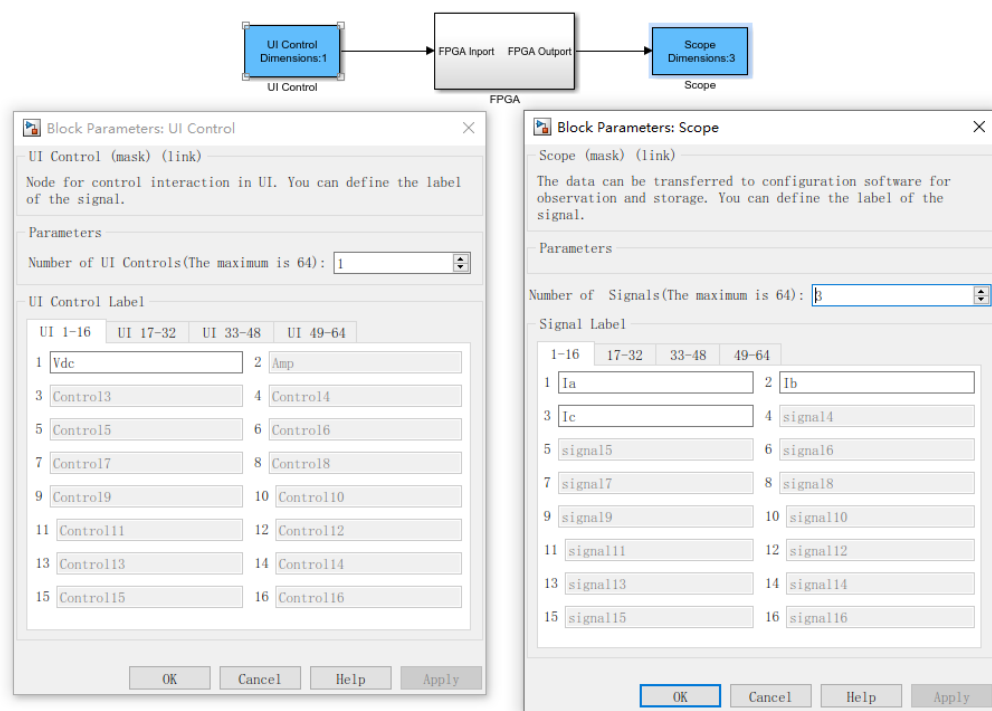
注意：每一个 FPGA 板卡都必须配有一个“FPGA Configuration”模块，如下图所示，在该模块中用户需要根据自己实际的硬件信息及 FPGA 算法的用途进行配置，详细说明可参考 EasyGo RealTime Block 帮助文档。





返回主程序，我们分别打开输入端“UI Ctrl1”和输出端“Scope”的配置窗口，如下图所示：这里我们把所有需要在 DeskSim 软件界面控制的信号全部配置在“UI Control”模块中，把所有需要在 DeskSim 软件界面上观察的信号全部配置在“Scope”模块中，同时在连入子系统后的“UI Control”的信号根据程序的需要解绑连入程序中，将程序输出的信号根据需要按顺序捆绑在一起输出至“Scope”模块中。

注意：程序中所有与主界面交互的数据必须通过“UI Control”模块和“Scope”模块进行交互，程序中存在的其他的输入输出信号均不支持在 DeskSim 软件上进行控制或显示，且同一个程序有且最多支持一个“UI Control”模块或“Scope”模块。（通讯模块使用方式同理，详细说明可参考 EasyGo RealTime Block 帮助文档）

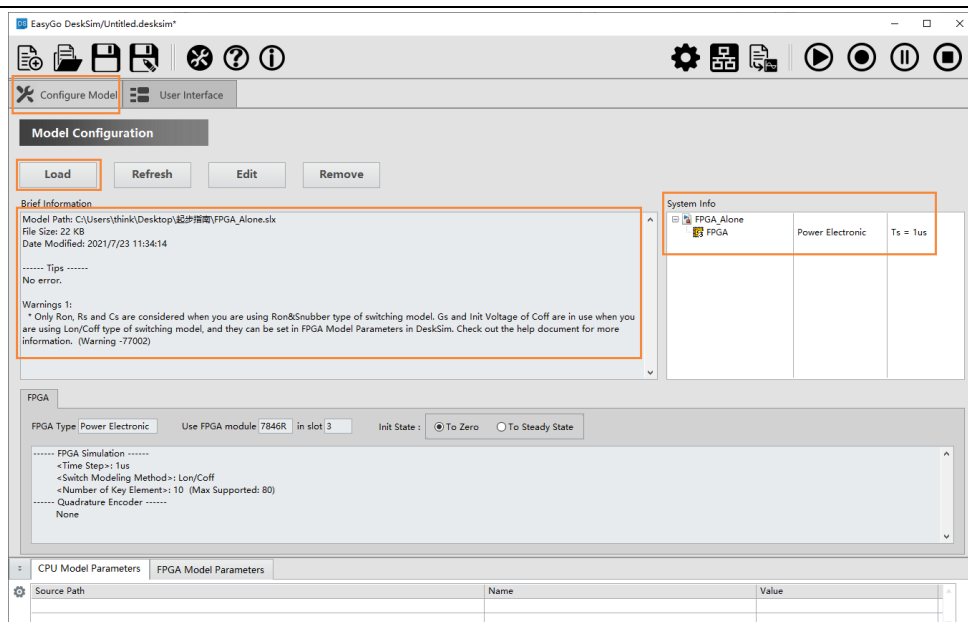


由此可知，整个程序实现的功能是输入端输入 Vdc 信号，输出端为电力电子算法中三路电流信号。

注：用户可以根据自己的 FPGA 上的算法修改 “FPGA” 子系统内的代码，以及对应修改输入输出端口的信号名称及数量。

导入程序

打开 EasyGo DeskSim 软件，在 Configure Model 界面中点击 “Load” 按钮载入程序文件；



载入文件后，软件会分析得到程序的基本信息。

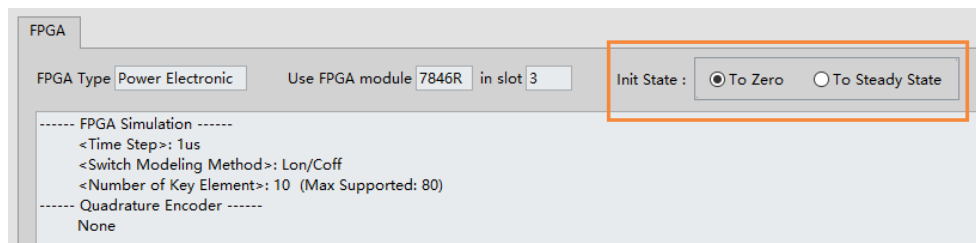
注意：

- ◆如果模型中存在错误，用户需要把错误信息全部处理掉，才能进行实时运行；
- ◆如果需要修改或者打开模型文件，可以点击“Edit”按钮打开模型文件进行修改或查看；
- ◆如果用户载入的模型文件发生了修改，在界面中可以点击“Refresh”按钮进行更新。

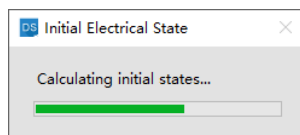
仿真配置

FPGA 模型初始化

对于使用了 FPGA 模型的程序，在 FPGA 模型为电力电子算法模型时，用户可根据实际需要对电力电子算法模型的初始状态进行处理，如下图所示：



如果需要初始化操作的话，点击“Init State”勾选框中的“To Steady State”选项，之后会弹出确认对话框，确认后开始对模型进行初始化处理，该过程会持续几分钟，如下图所示：

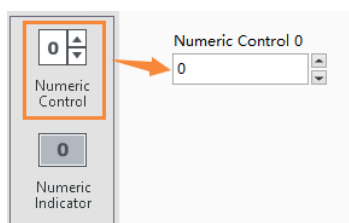


初始化完成后，模型在运行时会预先导入初始状态量进行电力电子拓扑仿真。

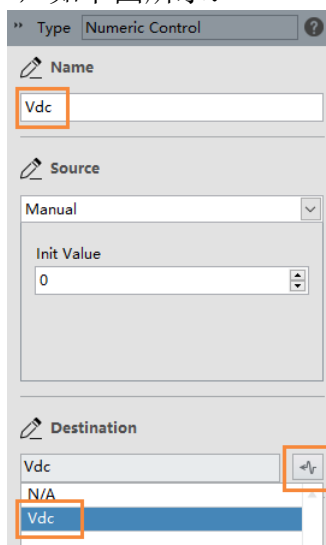
添加自定义界面

添加 UI 界面的控制输入

DeskSim 主界面切换至 UI Interface 界面中，在界面左侧的控件选板中，左键点击“Numeric Control”，并将其拖拽到中间的空白编辑区域释放，即完成一个 Numeric Control 控件的添加，如下图所示：



添加完成之后，点击选中这个控件，界面右侧栏会显示这个控件的配置内容，可选择信号绑定的对象（程序的所有输入端），修改名称和初始值；点击信号选择按钮，在弹出的信号列表中选择需要绑定的信号，如“Vdc”，选中后，控件名称会同步更新为信号名称，用户可以使用信号名称或手动修改名称，设置完成后，我们对设置初始参数为“50”，如下图所示：

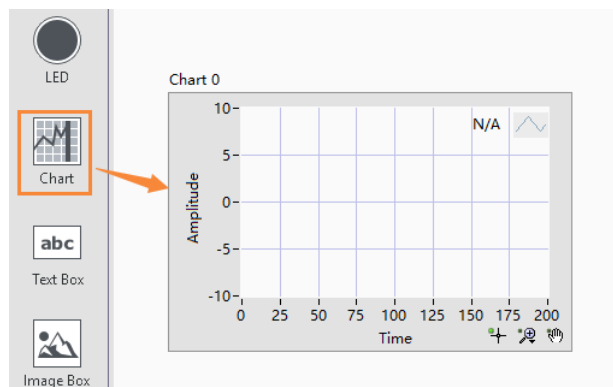


将程序的输入端口绑定到界面的控件之后可以得到：

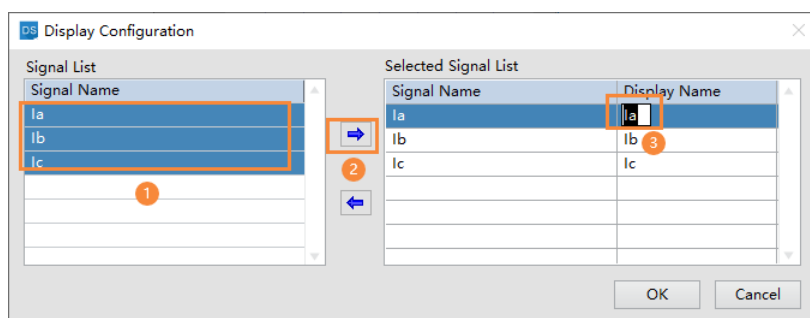
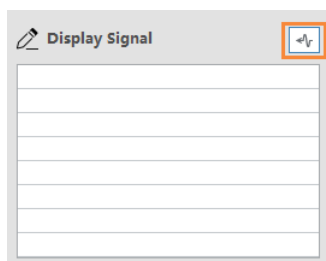


添加 UI 界面的波形显示

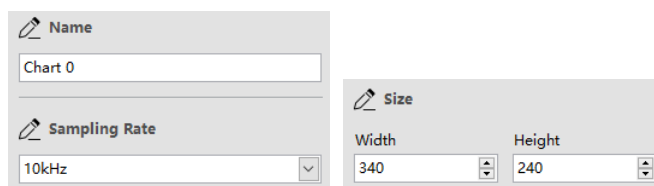
在界面左侧的控件选板中，左键点击“Chart”，并将其拖拽到中间的空白编辑区域释放，即完成一个 Chart 控件的添加，如下图所示：



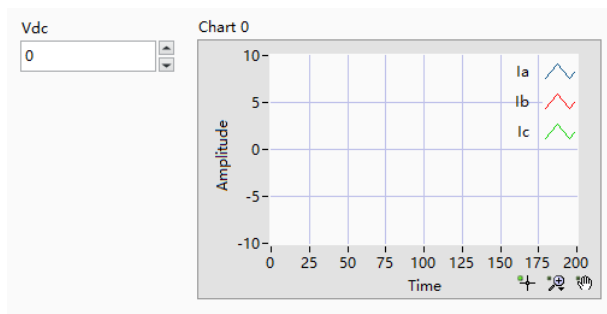
选中 Chart 控件，在右侧显示的配置区域中可以选中提那就显示的信号源（程序的所有输出信号均可显示）。



并可以修改 Chart 的名称、采样率和大小。



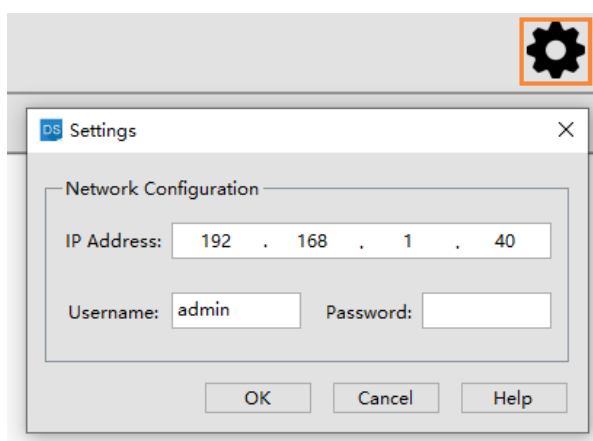
配置完成后，可以得到最终的自定义界面，如下图所示：



程序运行

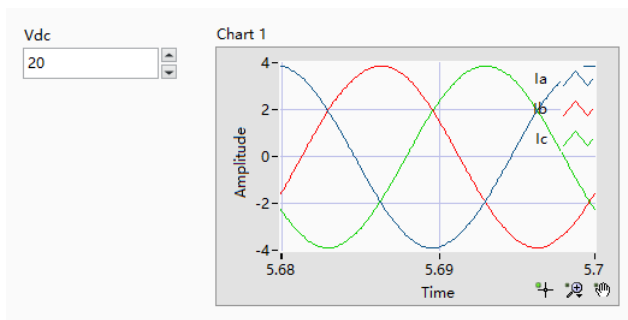
配置仿真机的网络信息

点击 DeskSim 软件工具栏的  按钮，即可打开仿真机的网络配置窗口，设置好 IP 地址、端口号等信息。



运行程序

点击软件工具栏的  按钮即可运行程序（目标仿真机上需要正确激活才能正常使用）。

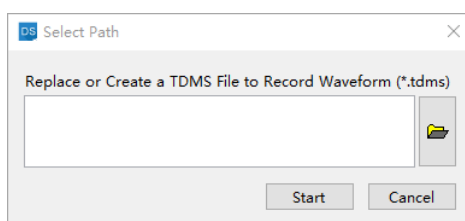


我们可以实时调整 Vdc 的值来观察实际波形的变化情况（由于该程序需要控

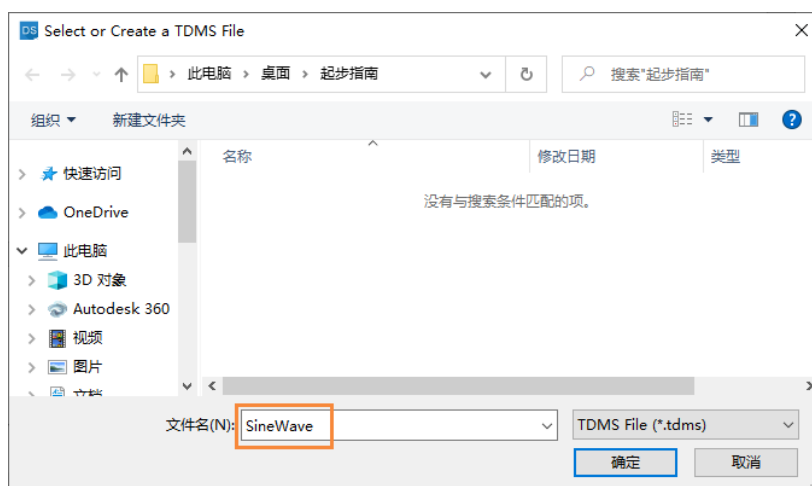
制测发送 PWM 信号来控制电路，所以在实际测试时，需外接控制设备进行测试）。具体现象不再详述。同时运行过程中还可以动态修改附加区中 FPGA Model Parameter 的参数，具体参考 EasyGo DeskSim 软件的帮助文档中的相关说明，由于我们的模型比较简单，所以没有使用动态修改参数功能。

波形录制

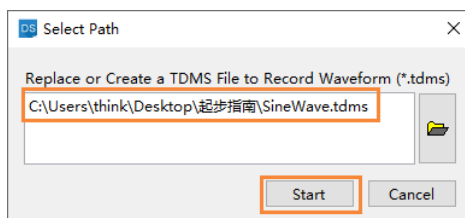
点击软件工具栏的  按钮，弹出录制波形配置界面，如下图所示：



点击路径配置按钮，弹出路径选择对话框，选好存储路径后，配置好波形文件的名称，如下图所示：



点击“确定”按钮完成配置，此时录制波形配置界面配置完成，如下图所示：



点击“Start”按钮，录波功能正式启动，此时我们可以看到界面上的波形录制按钮变成红色的，如下图所示：



在这个录波状态下，程序运行过程中，模型文件中 Scope 模块配置的所有的有效信号均会实时记录下来。

注：录制的波形文件为*.tdms 格式的，具体如何查看波形文件以及波形文件格式转换，请参考 EasyGo DeskSim 的帮助文档中的工具箱说明部分。

暂停程序

点击软件工具栏的  按钮即可随时暂停程序。暂停后，我们可以看到界面上的暂停按钮变成红色的，如下图所示：



注意：

- ◆ 暂停功能只是暂停主界面的波形更新或数值显示控件的数据更新，便于用户查看暂停瞬时前的一段时间的数据或波形；

- ◆ 暂停功能对实时仿真机中的程序不影响；

- ◆ 暂停功能对波形录制也不影响，上下位机的数据交互始终没有间断；

停止程序

点击软件工具栏的  按钮即可停止程序。

保存项目文件

全部运行完成后，为了便于下一次打开直接使用，我们可以对上述所有的配置操作进行保存管理，保存成 DeskSim 软件专属的项目文件 (*.desksim)。

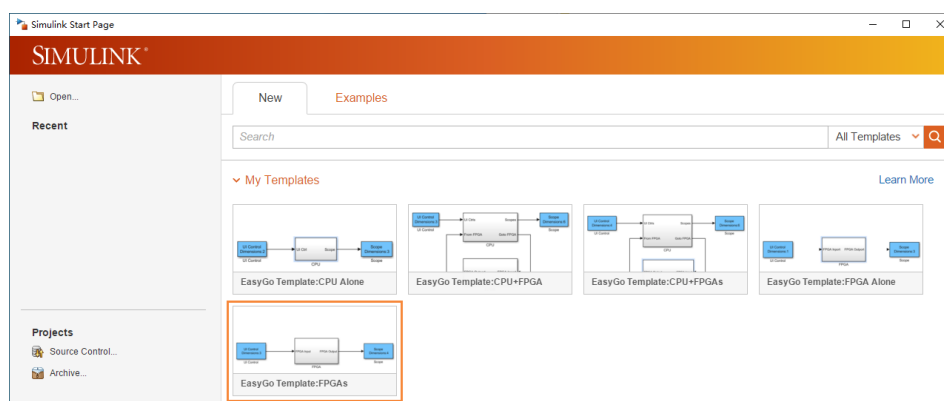
点击工具栏中的  保存按钮，选择路径，修改名称，即可完成保存。下次打开时，可以点击软件工具栏中的  打开按钮，载入以前保存的项目文件，或者直接双击项目文件，直接打开软件并载入项目文件。

Template 5: FPGAs

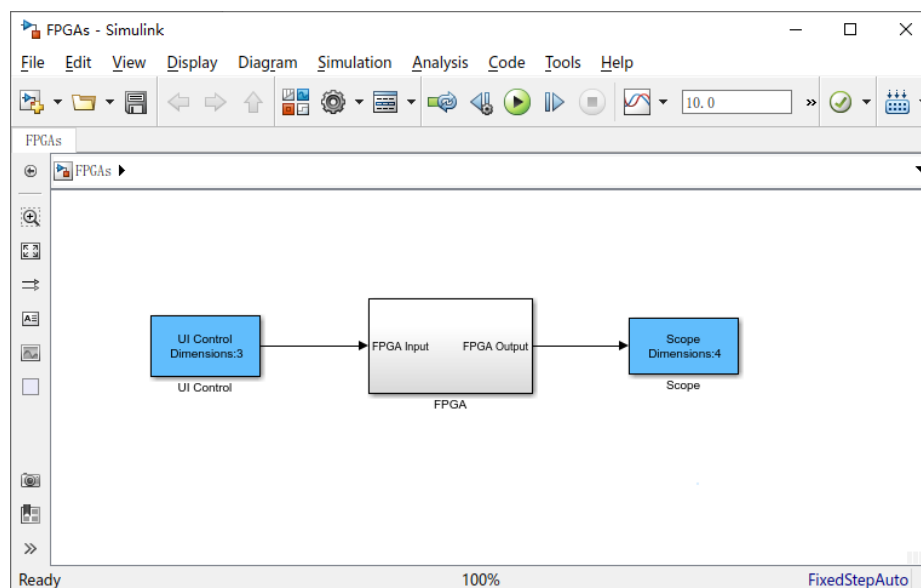
FPGAs 模型是利用 EasyGo Real-Time 函数库建立用于多块 FPGA 联合仿真的实时模型框架，即在实际仿真过程中使用多块 FPGA 板卡进行仿真，并将仿真结果上传至上位机观察或交互至工业通讯端口。

准备程序

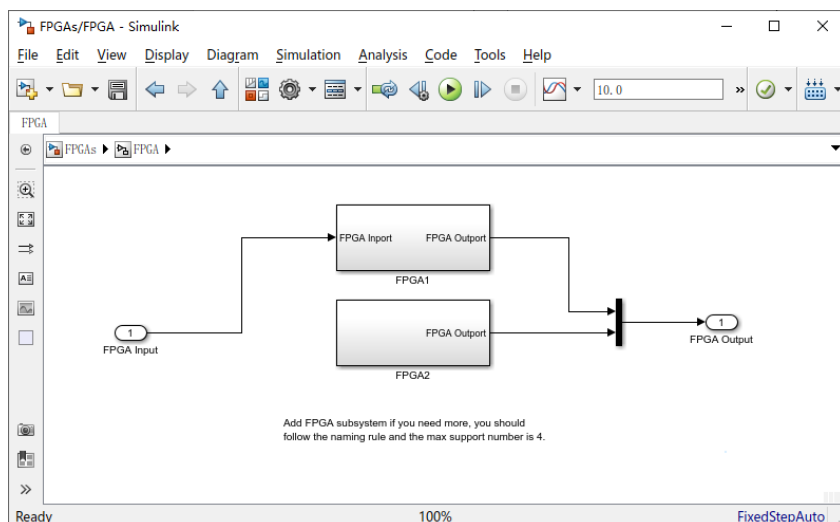
打开 Matlab 软件，点击启动 Simulink，在“Simulink Start Page”界面中点击“EasyGo Template: FPGAs”创建出 FPGAs 的框架模型，如下图所示：



新建出来的原始模型是未保存的，我们将其保存成“FPGAs. slx”，如下图所示：



由于是多卡 FPGA 仿真模型仿真程序，所以核心代码在“FPGA”子系统里面。双击打开“FPGA”子系统，如下图所示：

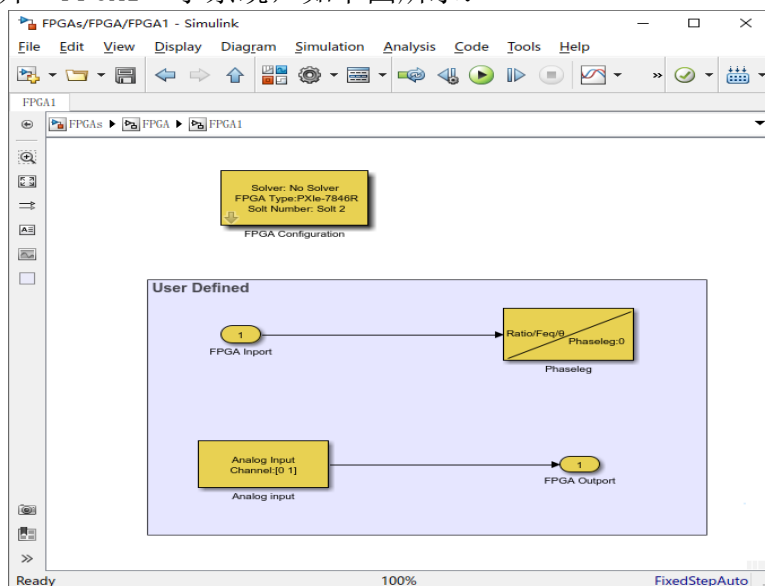


由于我们的程序是多块 FPGA 板卡联合仿真的程序，所以在 FPGA 子系统中又包含多个子系统，如上图显示的：FPGA1、FPGA2，即代表着两块不同的 FPGA 板卡，每个 FPGA 板卡上都有自己的程序。

用户在使用多个 FPGA 板卡联合仿真时，模型搭建方面需要注意以下几点：

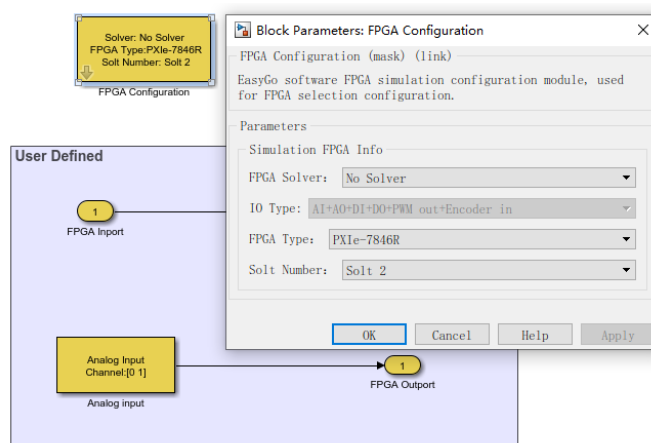
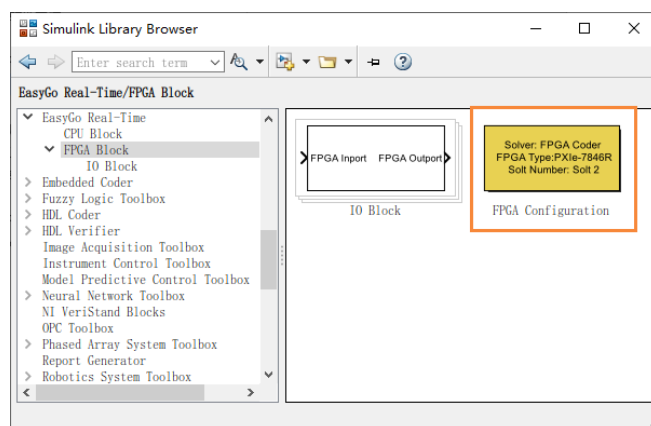
- ◆ 如果用户使用多块 FPGA 板卡仿真时，一定要将所有板卡全部放在这个“FPGA”子系统里面；且每块板卡上的程序都需要独立在一个子系统里面，子系统命名规则为“FPGA+序号”；
- ◆ 所有板卡的子系统的输入输出信号最终需要全部合并在一个端口上与 CPU 交互或直接与顶层交互模块交互（在没有 CPU 模型的情况下）；

双击打开“FPGA1”子系统，如下图所示：

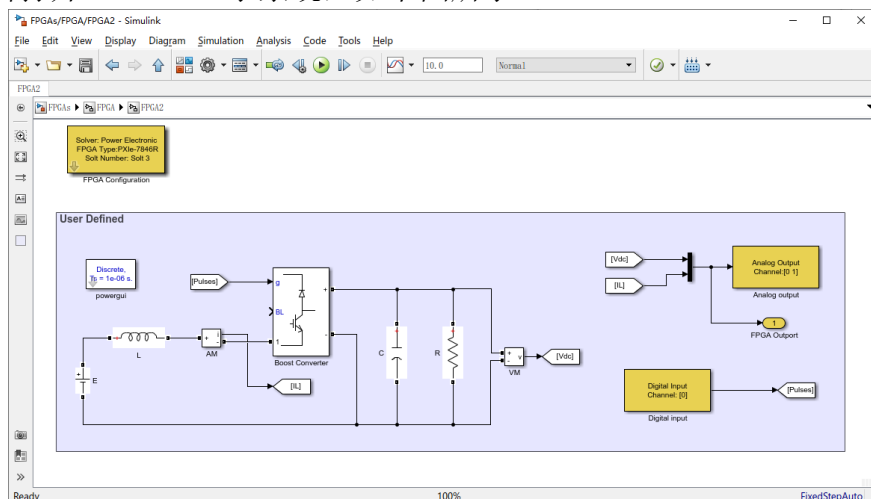


在“FPGA1”子系统中我们可以看到 FPGA 生成 PWM 波的程序，同时附带两路模拟信号采集(部分模块的详细说明可参考 EasyGo RealTime Block 帮助文档)。

注意：每一个 FPGA 板卡都必须配有一个“FPGA Configuration”模块，如下图所示，在该模块中用户需要根据自己实际的硬件信息及 FPGA 算法的用途进行配置，详细说明可参考 EasyGo RealTime Block 帮助文档。



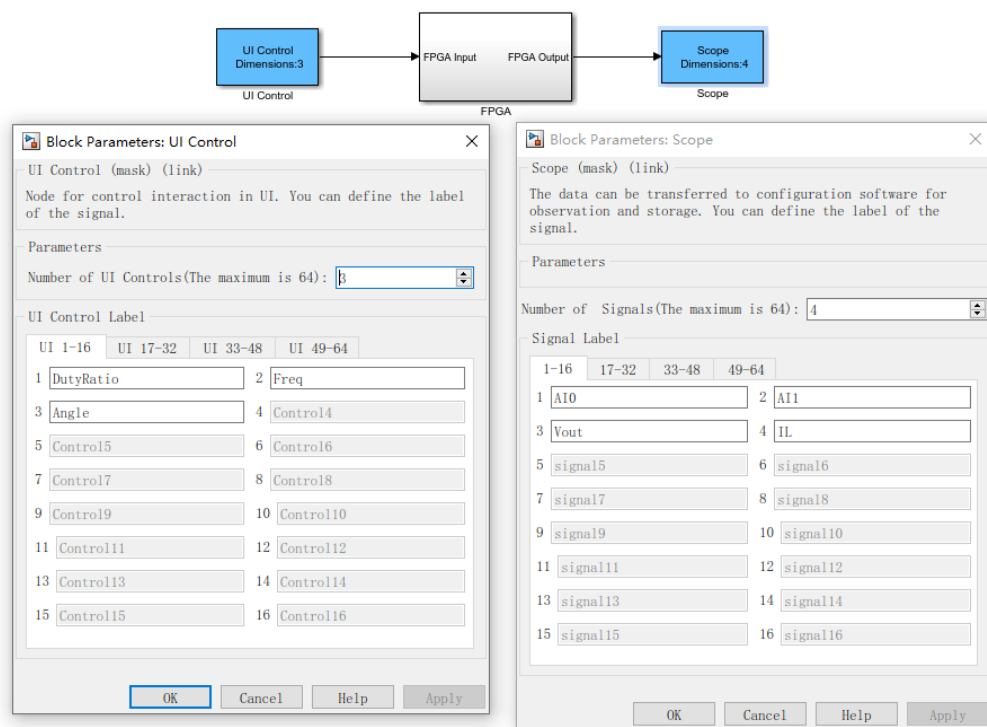
双击打开“FPGA2”子系统，如下图所示：



在“FPGA2”子系统中我们可以看到 FPGA 程序是一个电力电子算法，同时附带硬件 IO 上的数字信号采集及模拟信号输出（部分模块的详细说明可参考 EasyGo RealTime Block 帮助文档）。

返回主程序，我们分别打开输入端“UI Ctrl”和输出端“Scope”的配置窗口，如下图所示：这里我们把所有需要在 DeskSim 软件界面控制的信号全部配置在“UI Control”模块中，把所有需要在 DeskSim 软件界面上观察的信号全部配置在“Scope”模块中，同时在连入子系统后的“UI Control”的信号根据程序的需要解绑连入程序中，将程序输出的信号根据需要按顺序捆绑在一起输出至“Scope”模块中。

注意：程序中所有与主界面交互的数据必须通过“UI Control”模块和“Scope”模块进行交互，程序中存在的其他的输入输出信号均不支持在 DeskSim 软件上进行控制或显示，且同一个程序有且最多支持一个“UI Control”模块或“Scope”模块。（通讯模块使用方式同理，详细说明可参考 EasyGo RealTime Block 帮助文档）

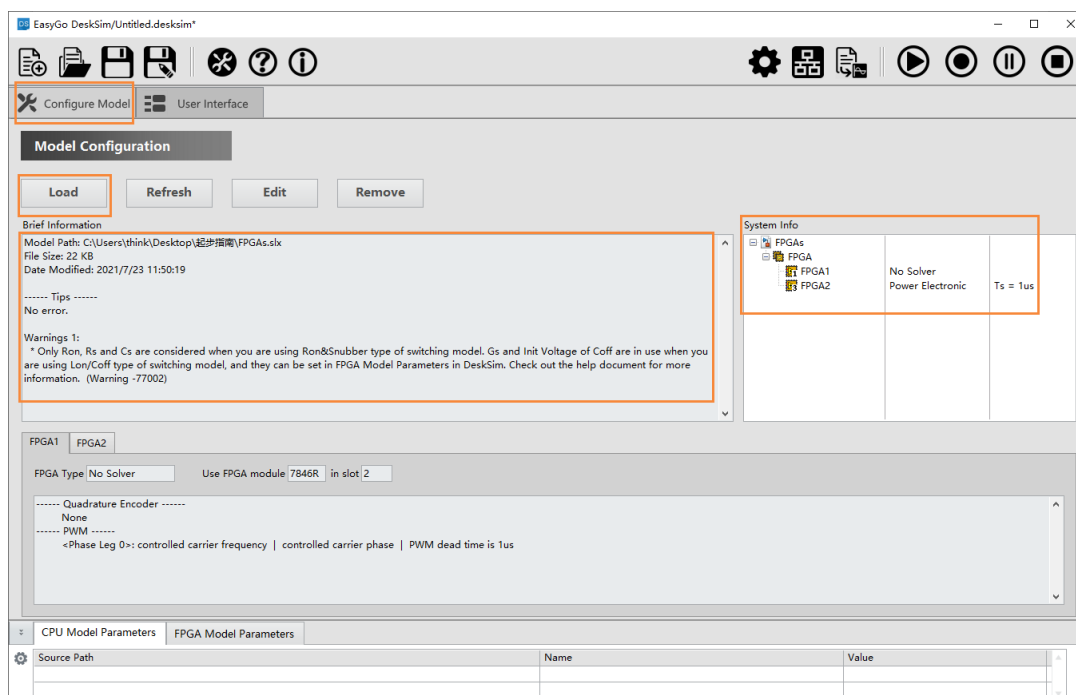


由此可知，整个程序实现的功能是输入端输入占空比、频率、角度信号，输出端为电力电子算法中电压电流及模拟采集信号。

注：用户可以根据自己的 FPGA 上的算法修改“FPGA”子系统内的代码，以及对应修改输入输出端口的信号名称及数量。

导入程序

打开 EasyGo DeskSim 软件，在 Configure Model 界面中点击“Load”按钮
载入程序文件；



载入文件后，软件会分析得到程序的基本信息。

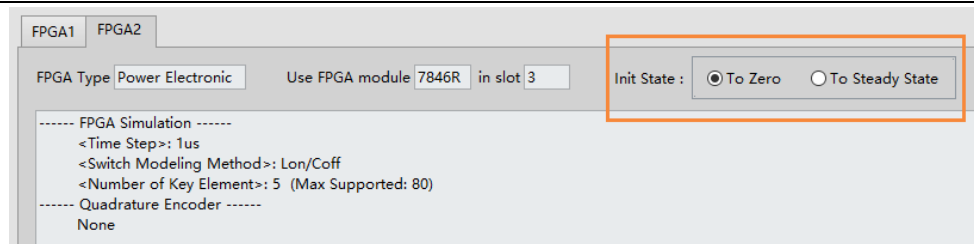
注意：

- ◆如果模型中存在错误，用户需要把错误信息全部处理掉，才能进行实时运行；
- ◆如果需要修改或者打开模型文件，可以点击“Edit”按钮打开模型文件进行修改或查看；
- ◆如果用户载入的模型文件发生了修改，在界面中可以点击“Refresh”按钮进行更新。

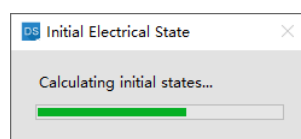
仿真配置

FPGA 模型初始化

对于使用了 FPGA 模型的程序，在 FPGA 模型为电力电子算法模型时，用户可根据实际需要对电力电子算法模型的初始状态进行处理，如下图所示：



如果需要初始化操作的话，点击“Init State”勾选框中的“To Steady State”选项，之后会弹出确认对话框，确认后开始对模型进行初始化处理，该过程会持续几分钟，如下图所示：

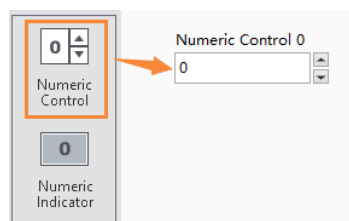


初始化完成后，模型在运行时会预先导入初始状态量进行电力电子拓扑仿真。

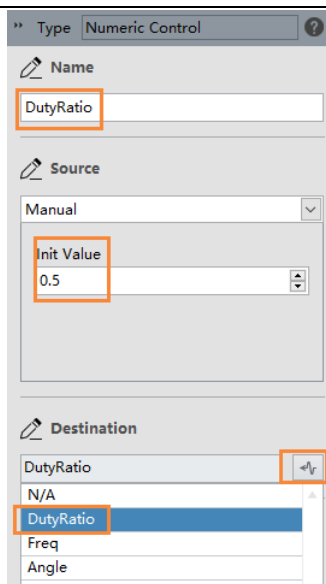
添加自定义界面

添加 UI 界面的控制输入

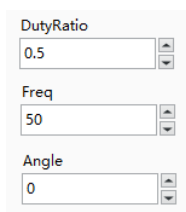
DeskSim 主界面切换至 UI Interface 界面中，在界面左侧的控件选板中，左键点击“Numeric Control”，并将其拖拽到中间的空白编辑区域释放，即完成一个 Numeric Control 控件的添加，如下图所示：



添加完成之后，点击选中这个控件，界面右侧栏会显示这个控件的配置内容，可选择信号绑定的对象（程序的所有输入端），修改名称和初始值；点击信号选择按钮，在弹出的信号列表中选择需要绑定的信号，如“DutyRatio”，选中后，控件名称会同步更新为信号名称，用户可以使用信号名称或手动修改名称，设置完成后，我们对设置初始参数为“0.5”，如下图所示：

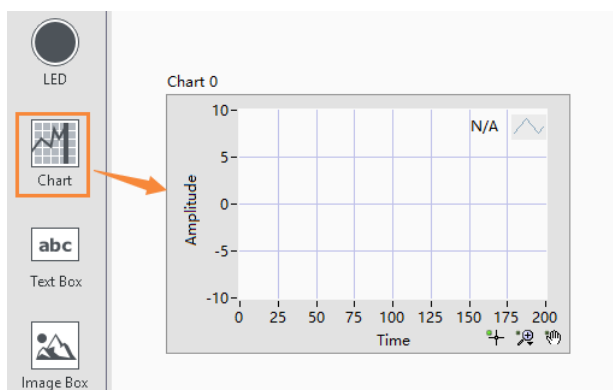


同样的方式，将程序的三个输入端口都绑定到界面的控件之后可以得到：

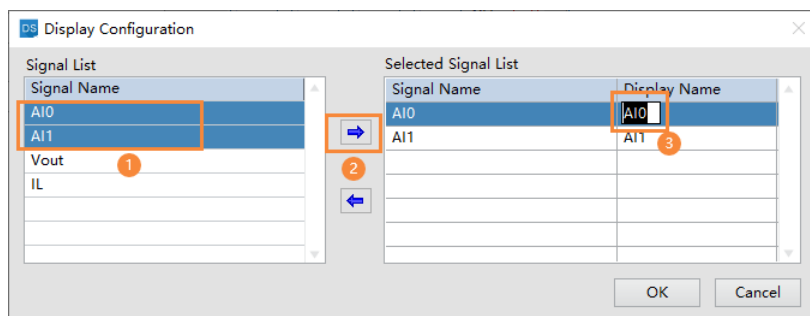
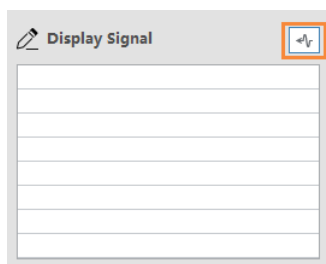


添加 UI 界面的波形显示

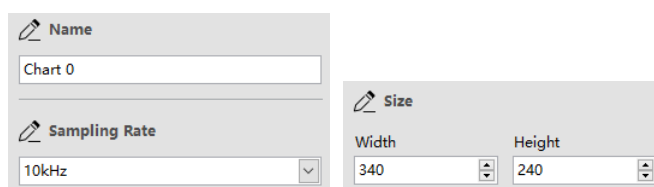
在界面左侧的控件选板中，左键点击“Chart”，并将其拖拽到中间的空白编辑区域释放，即完成一个 Chart 控件的添加，如下图所示：



选中 Chart 控件，在右侧显示的配置区域中可以选中提那就显示的信号源（程序的所有输出信号均可显示）。

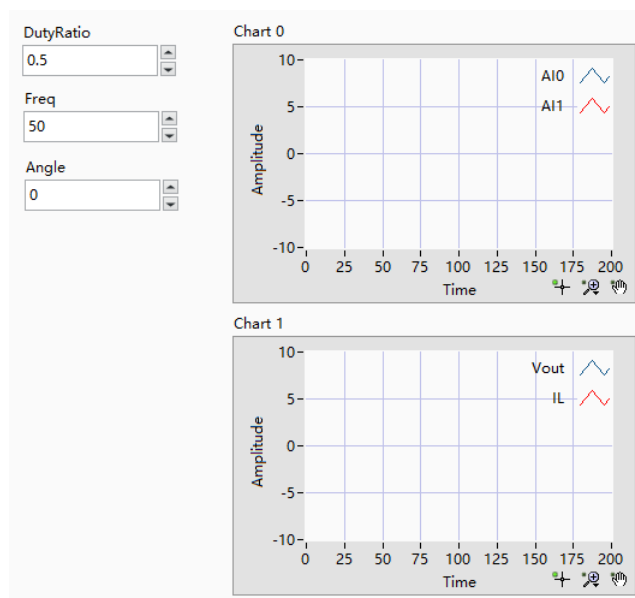


并可以修改 Chart 的名称、采样率和大小。



同样的方式，添加 Vout、IL 的波形显示。

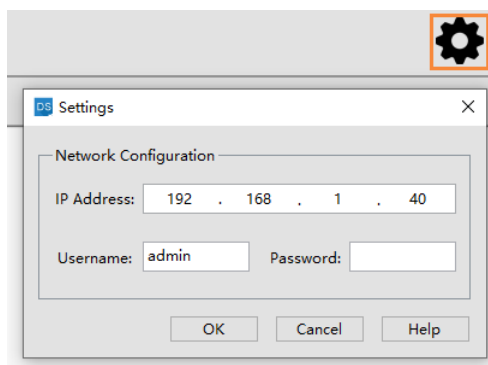
配置完成后，可以得到最终的自定义界面，如下图所示：



程序运行

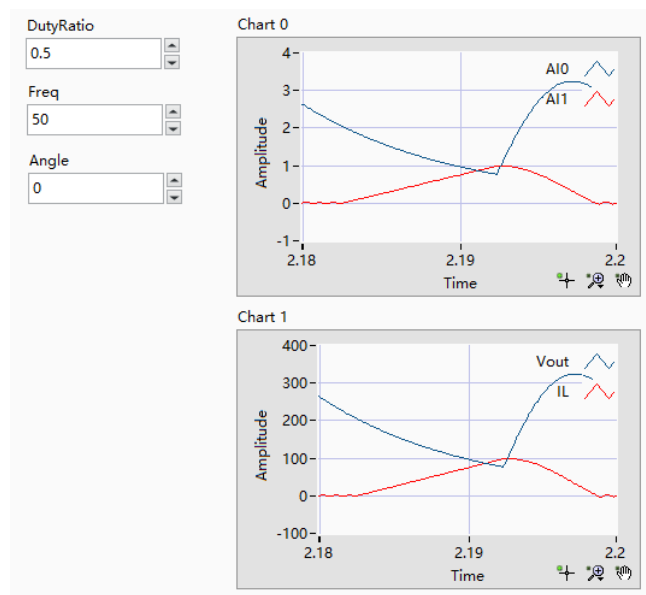
配置仿真机的网络信息

点击 DeskSim 软件工具栏的  按钮，即可打开仿真机的网络配置窗口，设置好 IP 地址、端口号等信息。

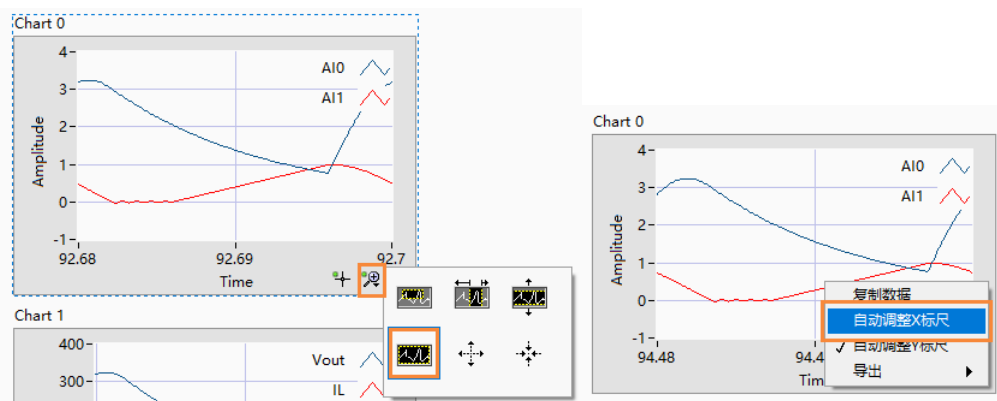


运行程序

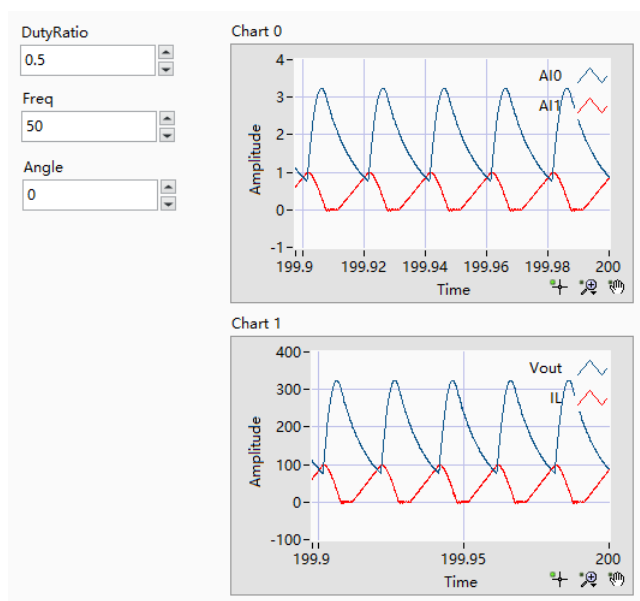
点击软件工具栏的  按钮即可运行程序（目标仿真机上需要正确激活才能正常使用）。



此时波形观察不完整，我们可以通过波形图的工具调整显示范围，或者右键波形图在快捷菜单中选择调整范围标尺，如下图所示：



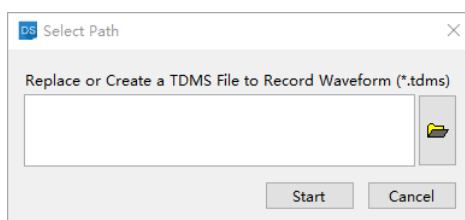
调整完成后，我们可以比较完整的观察到波形，如下图所示：



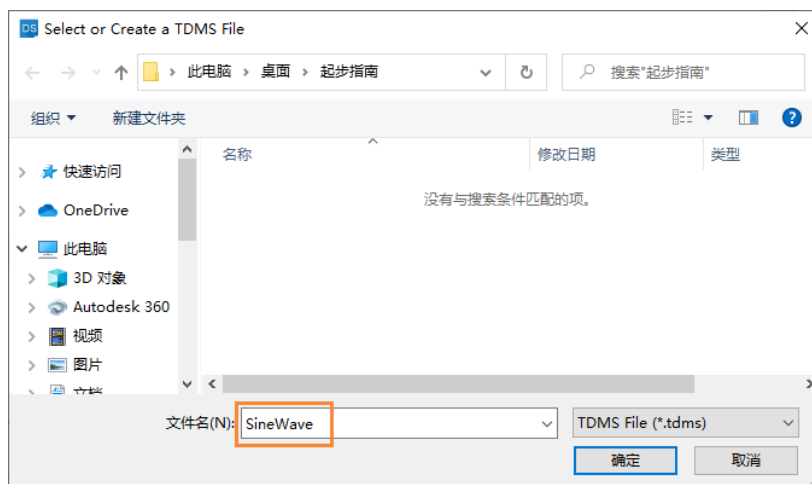
我们可以实时调整占空比的值来观察实际波形的变化情况。具体现象不再详述。同时运行过程中还可以动态修改附加区中 FPGA Model Parameter 的参数，具体参考 EasyGo DeskSim 软件的帮助文档中的相关说明，由于我们的模型比较简单，所以没有使用动态修改参数功能。

波形录制

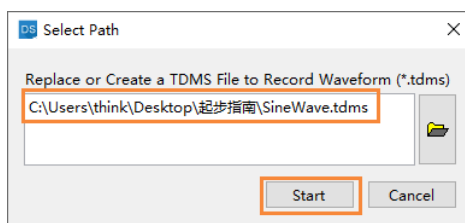
点击软件工具栏的  按钮，弹出录制波形配置界面，如下图所示：



点击路径配置按钮，弹出路径选择对话框，选好存储路径后，配置好波形文件的名称，如下图所示：



点击“确定”按钮完成配置，此时录制波形配置界面配置完成，如下图所示：



点击“Start”按钮，录波功能正式启动，此时我们可以看到界面上的波形录制按钮变成红色的，如下图所示：



在这个录波状态下，程序运行过程中，模型文件中 Scope 模块配置的所有的有效信号均会实时记录下来。

注：录制的波形文件为*.tdms 格式的，具体如何查看波形文件以及波形文件格式转换，请参考 EasyGo DeskSim 的帮助文档中的工具箱说明部分。

暂停程序

点击软件工具栏的  按钮即可随时暂停程序。暂停后，我们可以看到界面上的暂停按钮变成红色的，如下图所示：



注意：

◆ 暂停功能只是暂停主界面的波形更新或数值显示控件的数据更新，便于用户查看暂停瞬时前的一段时间的数据或波形；

◆ 暂停功能对实时仿真机中的程序不影响；

◆ 暂停功能对波形录制也不影响，上下位机的数据交互始终没有间断；

停止程序

点击软件工具栏的按钮即可停止程序。

保存项目文件

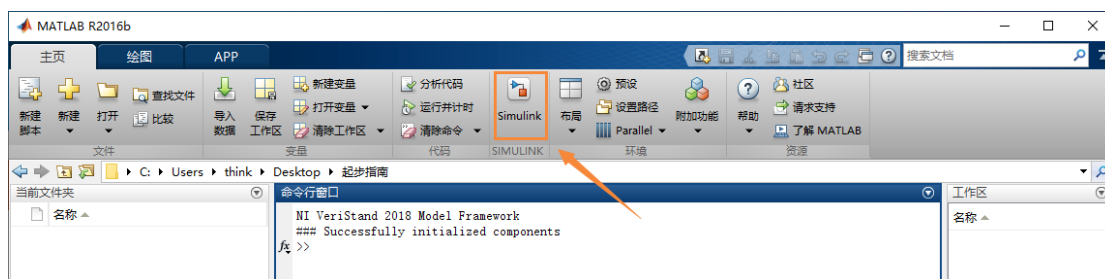
全部运行完成后，为了便于下一次打开直接使用，我们可以对上述所有的配置操作进行保存管理，保存成 DeskSim 软件专属的项目文件 (*.desksim)。

点击工具栏中的保存按钮，选择路径，修改名称，即可完成保存。下次打开时，可以点击软件工具栏中的打开按钮，载入以前保存的项目文件，或者直接双击项目文件，直接打开软件并载入项目文件。

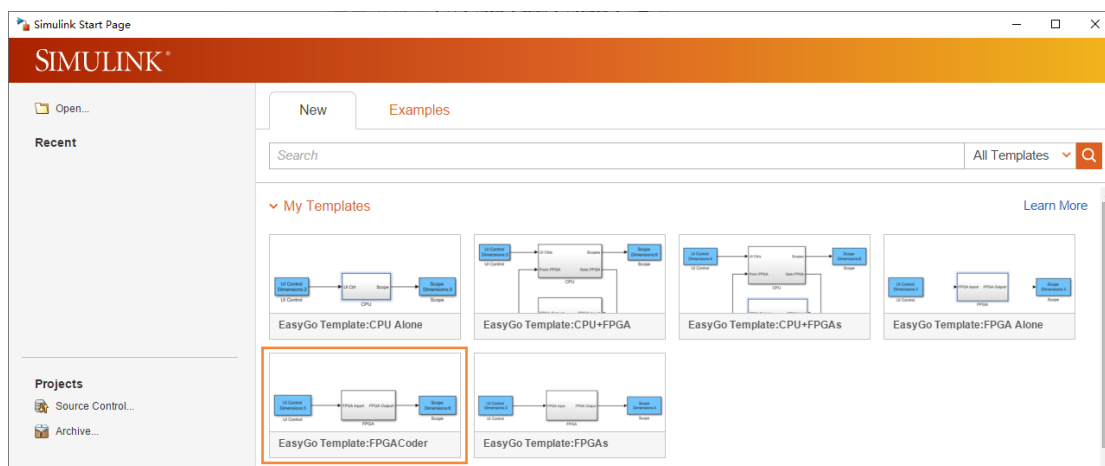
EasyGo FPGACoder 使用流程

EasyGo FPGA Coder 是 EasyGo 公司研发的一个基于 FPGA 的快速算法开发技术，能将用户基于 Simulink 开发的算法快速实现在 FPGA 上，无需经过传统的编译过程，大大增加 FPGA 的开发速率，降低 FPGA 的使用门槛。

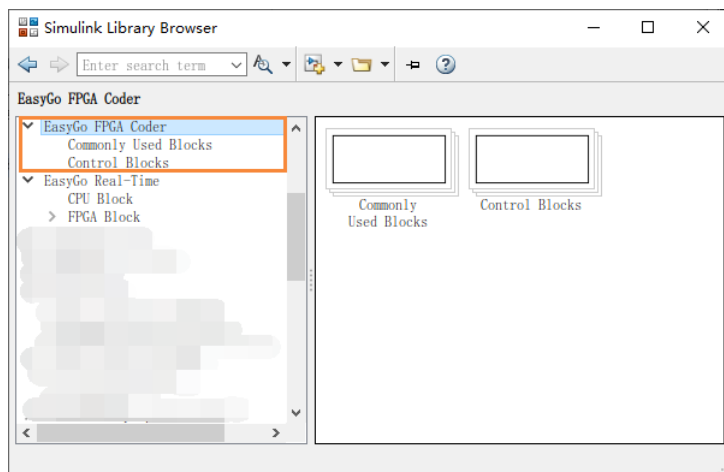
当我们正常安装完我们的 EasyGo FPGACoder 软件后，打开 Matlab 软件，点击启动 Simulink，如下图所示：



启动 Simulink 后，弹出“Simulink Start Page”，在该页面中我们可以看到在“My Templates”目录下，会在原有 5 个“EasyGo Template”基础上多一个“EasyGo Template:FPGACoder”模板，如下图所示。



同时我们打开“Simulink Library Browser”页面，在该页面中可以看到“EasyGo FPGA Coder”函数库，如下图所示：



利用 EasyGo FPGA Coder 函数库配合 EasyGo Real-Time 函数库及提供的 FPGA Coder 的模板模型可以实现快速搭建模型。

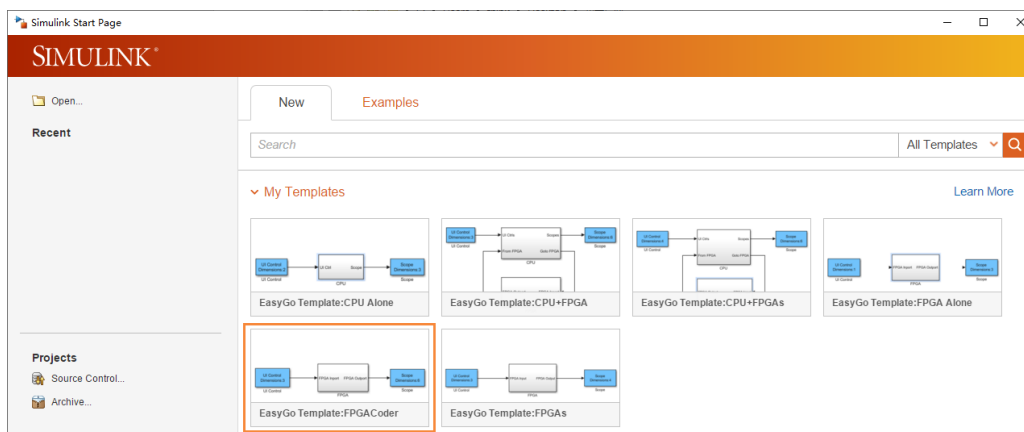
接下来，我们根据 FPGA Coder 框架模型，依照流程进行说明。

EasyGo Template ： FPGACoder

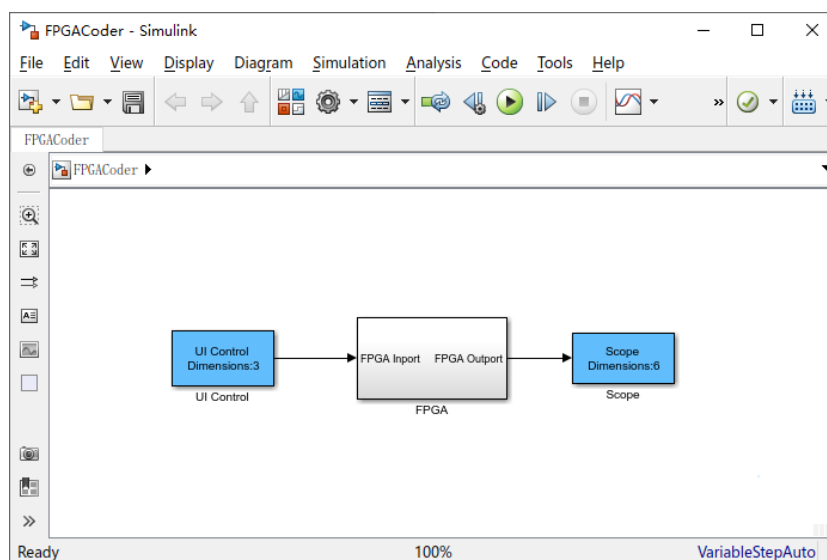
FPGACoder Template 是利用 EasyGo FPGA Coder 函数库配合 EasyGo Real-Time 函数库建立用于纯 FPGA 仿真的实时模型框架，即在实际仿真过程中使用 FPGA 板卡进行仿真，并将仿真结果上传至上位机观察或交互至工业通讯端口。如果用户需要在其他框架下使用 FPGA Coder 的功能，可以将 FPGA 下的程序复制到其他框架的 FPGA 中使用。

准备程序

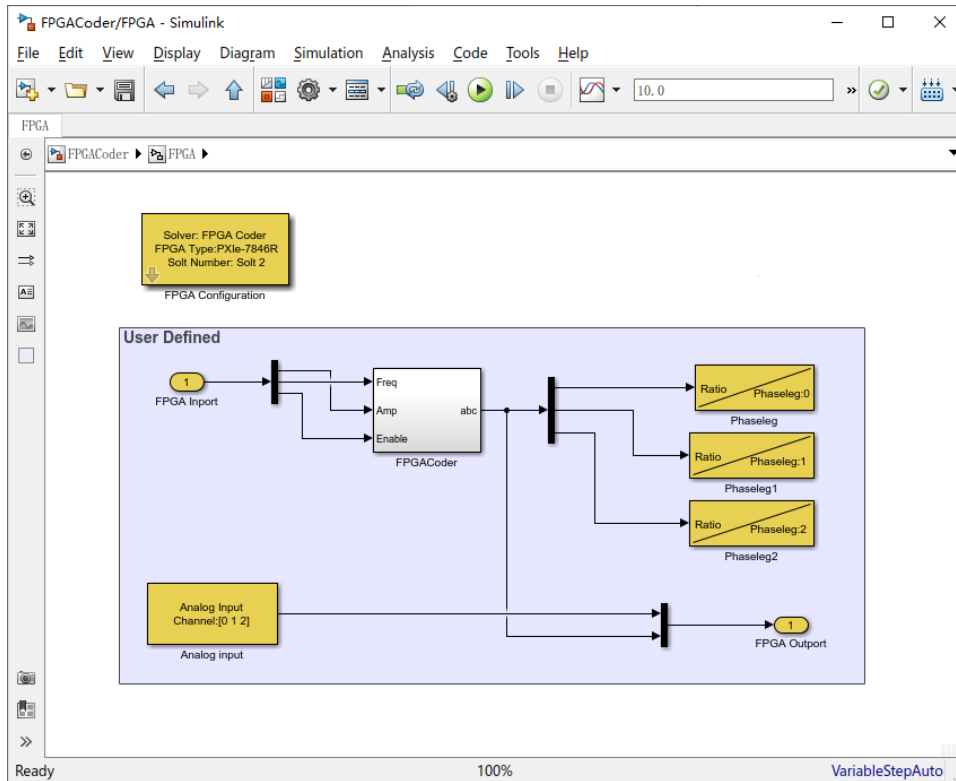
打开 Matlab 软件，点击启动 Simulink，在“Simulink Start Page”界面中点击“EasyGo Template: FPGACoder”创建出 FPGACoder 的框架模型，如下图所示：



新建出来的原始模型是未保存的，我们将其保存成“FPGACoder.slx”，如下图所示：



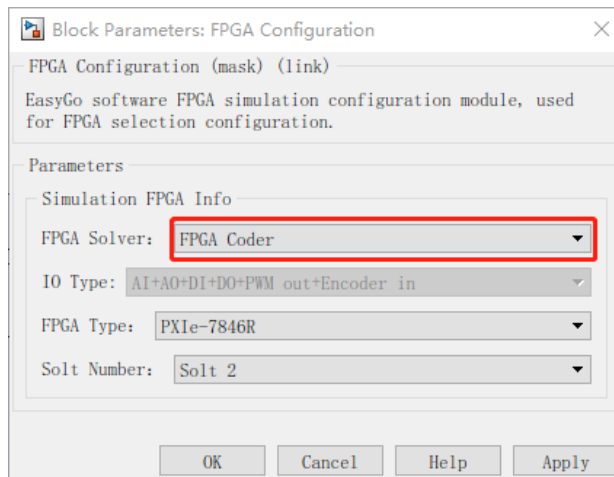
由于是 FPGACoder 仿真模型仿真程序，所以核心代码都在“FPGA”子系统中，双击打开“FPGA”子系统，如下图所示：



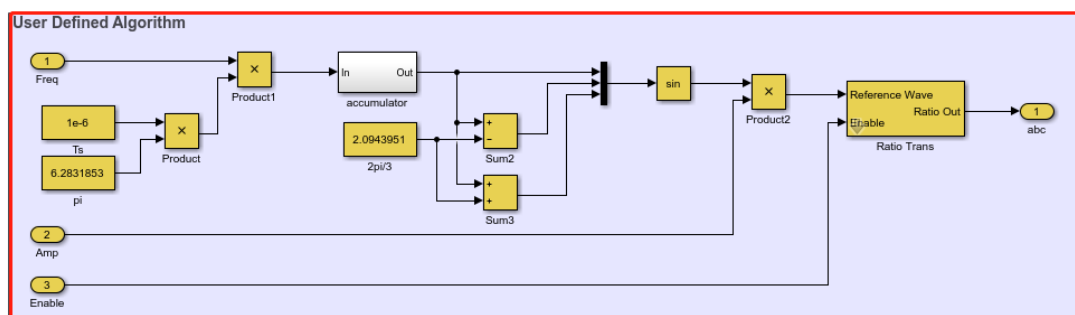
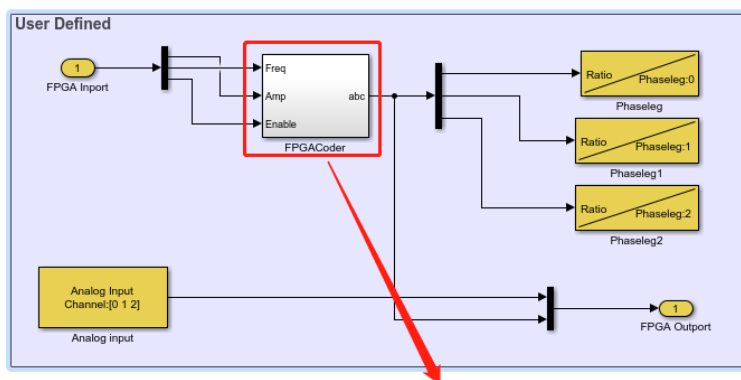
由于我们的程序是使用的单块 FPGA 板卡作为 FPGACoder 仿真的程序，所以只有一个 FPGA 子系统，如果用户使用多 FPGA 仿真，可以参考“FPGAs”框架来调用 FPGACoder 仿真程序。在该“FPGA”子系统中我们可以看到还包含一个“FPGACoder”子系统及外围交互的模型代码，在调用 FPGACoder 建立算法时，模型必须要放在这个“FPGACoder”子系统内。除了核心的“FPGACoder”子系统外，外围只支持配置模块及基础的 IO 模块。

用户在使用 EasyGo FPGA Coder 仿真时，模型搭建方面需要注意以下几点：

◆ FPGA Configuration 中的 FPGA Solver 需要选择为 FPGA Coder。



- ◆ FPGA Coder 的算法需要放在 FPGACoder 的子系统内，输入输出的个数和维数可以自己定义。

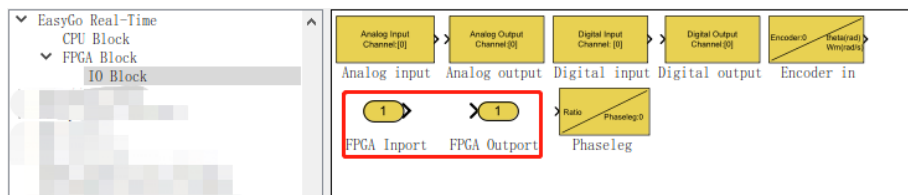


- ◆ 在一个 FPGA 子系统里只能有一个 FPGACoder 子系统。
- ◆ FPGACoder 子系统所在的 FPGA 子系统中，除了核心“FPGACoder”子系统外，只支持配置模块及基础的 IO 模块。

Template 中提供了一个简单的三相正弦波的生成算法，用户可以根据自己的需要进行任意搭建。编辑 FPGA Coder 算法有以下几个注意事项：

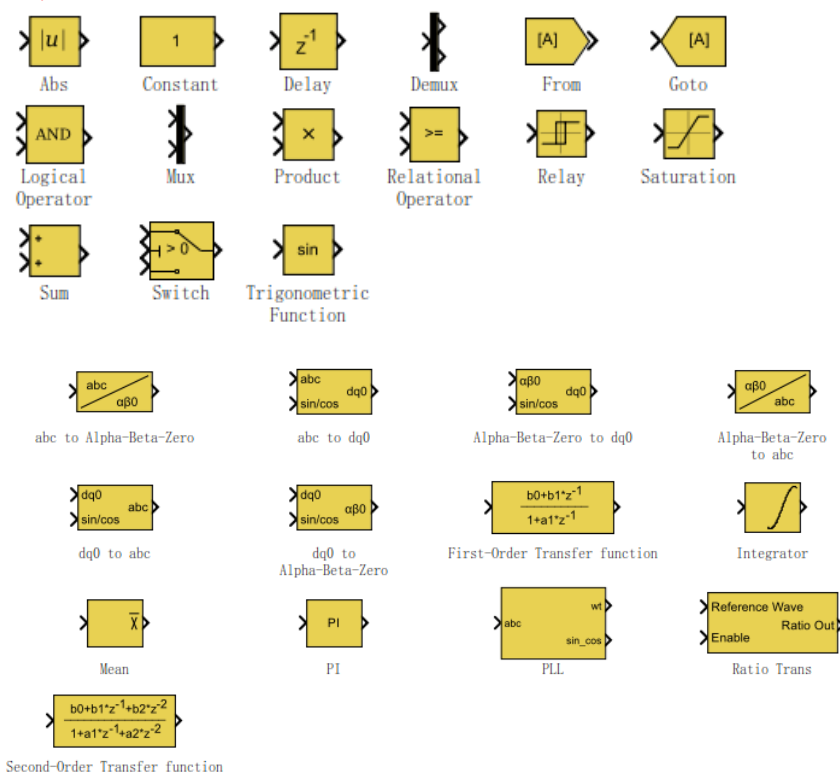
输入输出端口

- ◆ 在 FPGACoder 最上层的输入输出端口，需要用 EasyGo Real-Time>>FPGA Block>>IO Block 下的 FPGA Inport 和 FPGA Outport 这两个端口，并需要正确设置端口的维数。如果是 FPGACoder 里面的子系统，就不需要用这两个端口模块，也不需要设置维数信息。



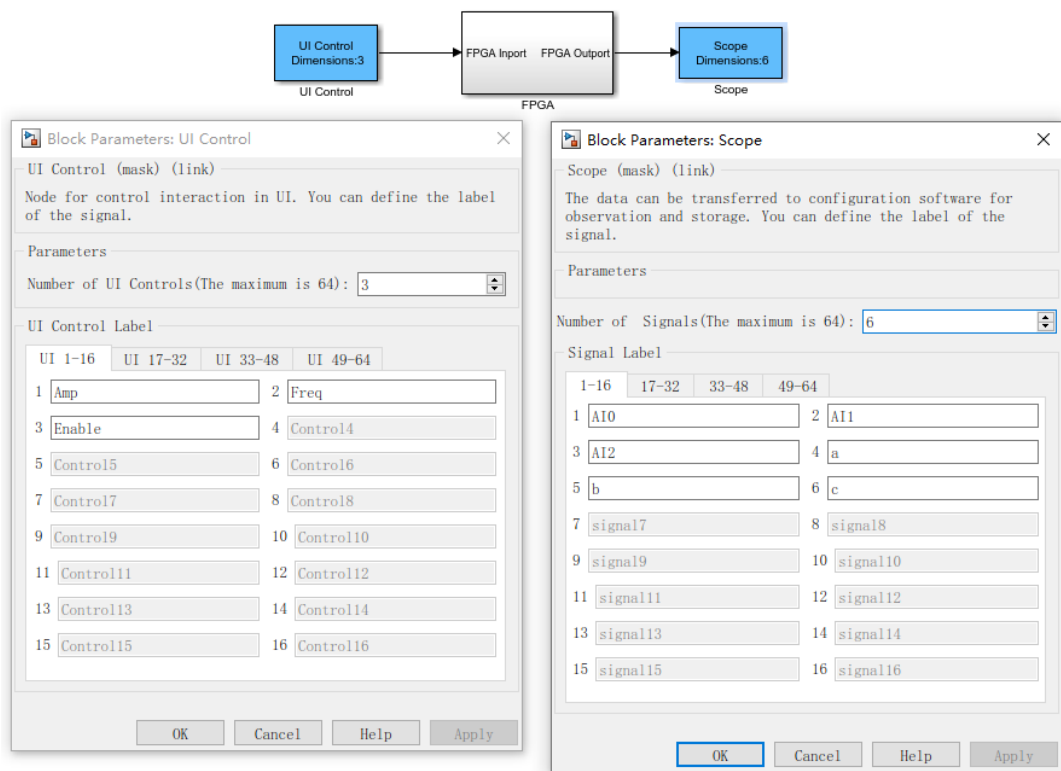
函数模块

◆ 在 FPGACoder 子系统下的函数模块，都需要用 EasyGo FPGA Coder 库下的函数。



返回主程序，我们分别打开输入端“UI Ctrl1”和输出端“Scope”的配置窗口，如下图所示：这里我们把所有需要在 DeskSim 软件界面控制的信号全部配置在“UI Control1”模块中，把所有需要在 DeskSim 软件界面上观察的信号全部配置在“Scope”模块中，同时在连入子系统后的“UI Control1”的信号根据程序的需要解绑连入程序中，将程序输出的信号根据需要按顺序捆绑在一起输出至“Scope”模块中。

注意：程序中所有与主界面交互的数据必须通过“UI Control1”模块和“Scope”模块进行交互，程序中存在的其他的输入输出信号均不支持在 DeskSim 软件上进行控制或显示，且同一个程序有且最多支持一个“UI Control1”模块或“Scope”模块。（通讯模块使用方式同理，详细说明可参考 EasyGo RealTime Block 帮助文档）

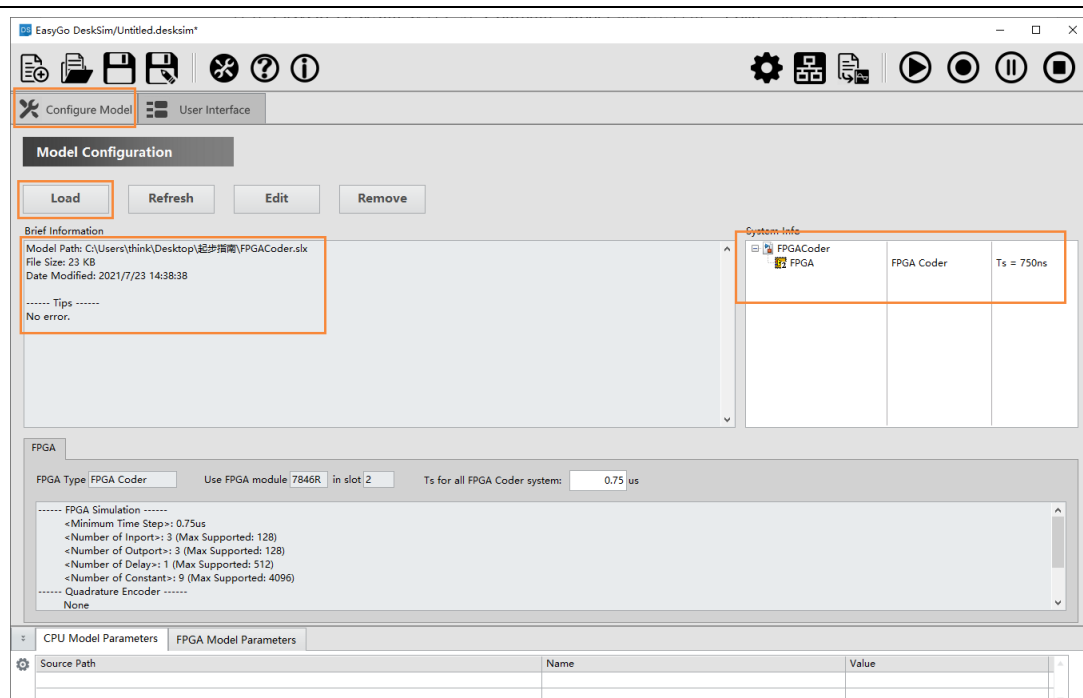


由此可知，整个程序实现的功能是输入端输入幅值、频率、使能信号，输出端为三路模拟采集及 FPGACoder 算法生成的三相波的瞬时值。

注：用户可以根据自己的 FPGA 上的算法修改 “FPGA” 子系统中的代码，以及对应修改输入输出端口的信号名称及数量。

导入程序

打开 EasyGo DeskSim 软件，在 Configure Model 界面中点击 “Load” 按钮载入程序文件；



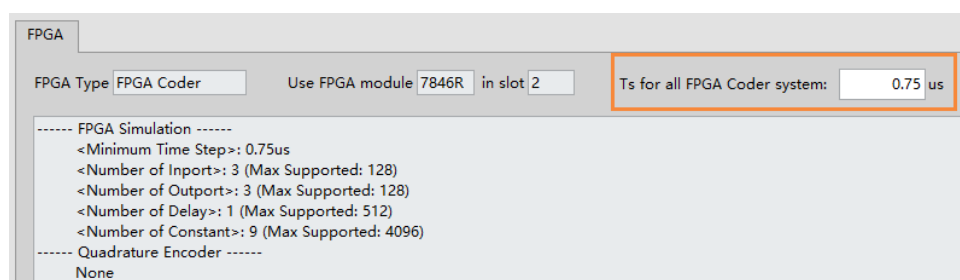
载入文件后，软件会分析得到程序的基本信息。

注意：

- ◆ 如果模型中存在错误，用户需要把错误信息全部处理掉，才能进行实时运行；
- ◆ 如果需要修改或者打开查看模型文件，可以点击“Edit”按钮打开模型文件进行修改或查看；
- ◆ 如果用户载入的模型文件发生了修改，在界面中可以点击“Refresh”按钮进行更新。

模型配置

由于我们只是单纯的使用了 FPGACoder 功能，所以这里的工作只有调整 Ts，如下图所示：



FPGA Coder Solver 仿真规模条件如下：

Minimum Time Step	最小的运行步长
Number of Inport	输入的数量，最大支持 128 个
Number of Outport	输出的数量，最大支持 128 个
Number of Delay	Delay 的数量，最大支持 512 个
Number of Constant	常数的数量，最大支持 4096 个

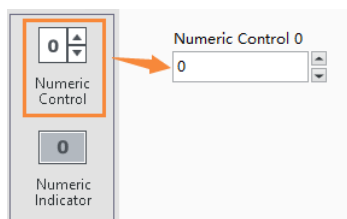
这里 DeskSim 默认显示的都是 FPGA Coder 算法在 FPGA 板卡上最小能够运行的步长，用户可以根据自己的需要修改 Ts 值，这里我们保持默认即可。

载入文件后，对于 FPGA Coder 的部分，不需要经过编译的过程就可以直接运行。

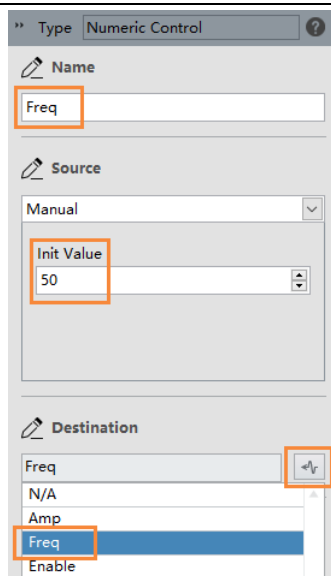
添加自定义界面

添加 UI 界面的控制输入

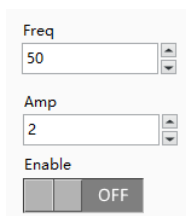
DeskSim 主界面切换至 UI Interface 界面中，在界面左侧的控件选板中，左键点击“Numeric Control”，并将其拖拽到中间的空白编辑区域释放，即完成一个 Numeric Control 控件的添加，如下图所示：



添加完成之后，点击选中这个控件，界面右侧栏会显示这个控件的配置内容，可选择信号绑定的对象（程序的所有输入端），修改名称和初始值；点击信号选择按钮，在弹出的信号列表中选择需要绑定的信号，如“Freq”，选中后，控件名称会同步更新为信号名称，用户可以使用信号名称或手动修改名称，设置完成后，我们对设置初始参数为“50”，如下图所示：

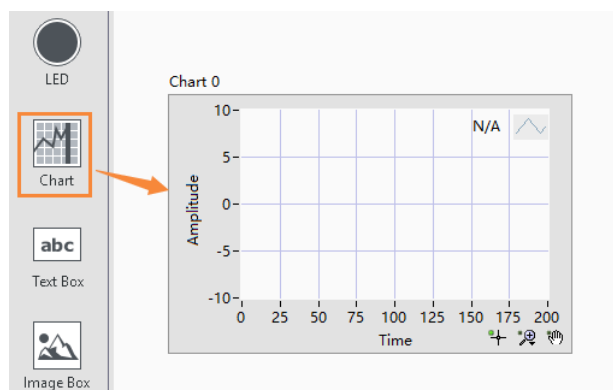


同样的方式，将程序的三个输入端口都绑定到界面的控件之后可以得到：

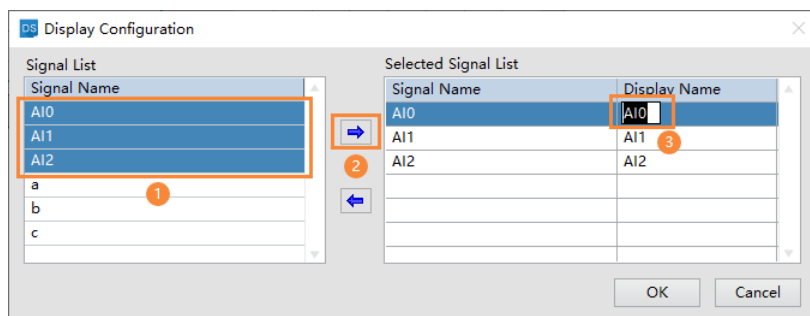
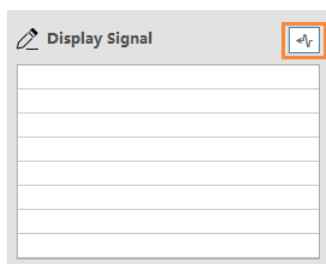


添加 UI 界面的波形显示

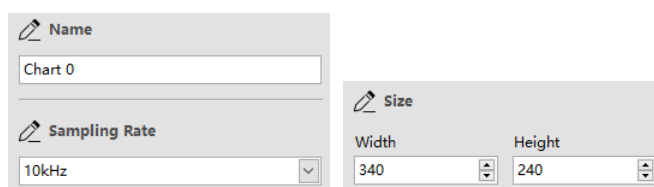
在界面左侧的控件选板中，左键点击“Chart”，并将其拖拽到中间的空白编辑区域释放，即完成一个 Chart 控件的添加，如下图所示：



选中 Chart 控件，在右侧显示的配置区域中可以选中提那就显示的信号源（程序的所有输出信号均可显示）。

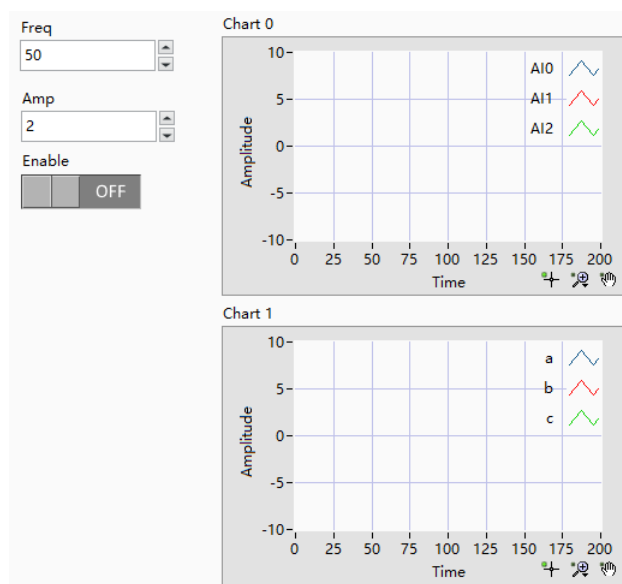


并可以修改 Chart 的名称、采样率和大小。



同样的方式，添加 a，b，c 的波形显示。

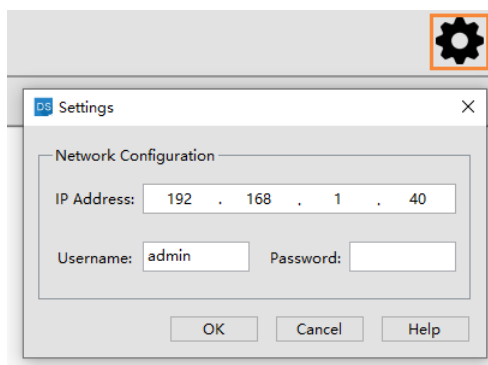
配置完成后，可以得到最终的自定义界面，如下图所示：



程序运行

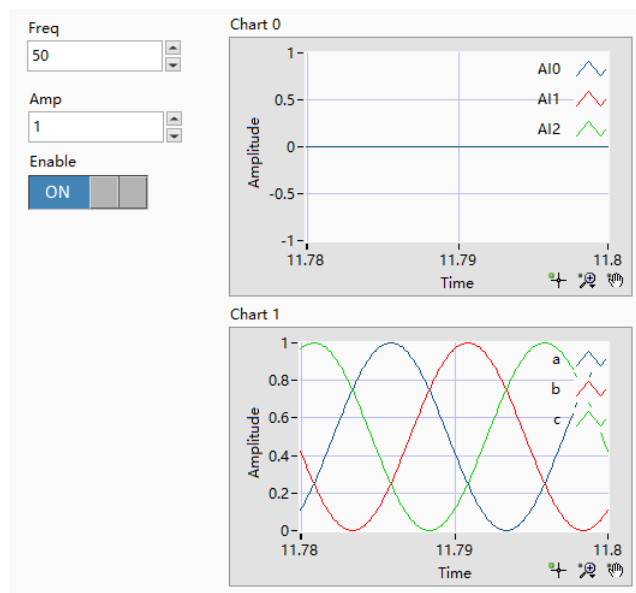
配置仿真机的网络信息

点击 DeskSim 软件工具栏的  按钮，即可打开仿真机的网络配置窗口，设置好 IP 地址、端口号等信息。



运行程序

点击软件工具栏的  按钮即可运行程序（目标仿真机上需要正确激活才能正常使用）。

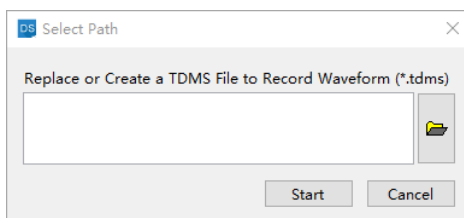


我们可以实时调整 Amp 的值来观察实际波形的变化情况。由于实际的 AI 通道是悬空的，所以采集到的信号均为 0V，我们可以外接信号来观察采集信号是否正确。具体现象不再详述。同时运行过程中还可以动态修改附加区中 FPGA Model Parameter 的参数，具体参考 EasyGo DeskSim 软件的帮助文档中的相关

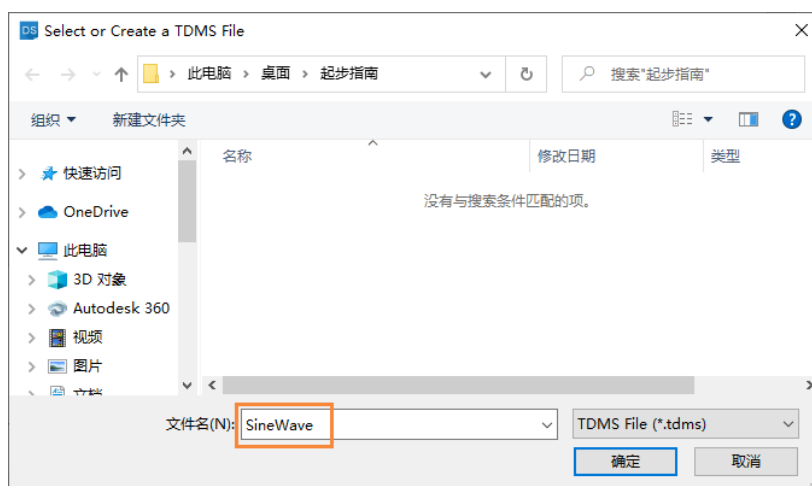
说明，由于我们的模型比较简单，所以没有使用动态修改参数功能。

波形录制

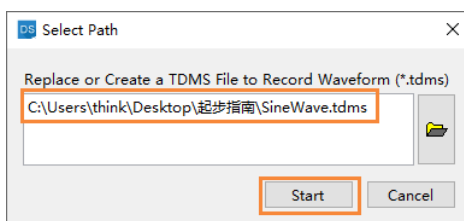
点击软件工具栏的  按钮，弹出录制波形配置界面，如下图所示：



点击路径配置按钮，弹出路径选择对话框，选好存储路径后，配置好波形文件的名称，如下图所示：



点击“确定”按钮完成配置，此时录制波形配置界面配置完成，如下图所示：



点击“Start”按钮，录波功能正式启动，此时我们可以看到界面上的波形录制按钮变成红色的，如下图所示：



在这个录波状态下，程序运行过程中，模型文件中 Scope 模块配置的所有的有效信号均会实时记录下来。

注：录制的波形文件为*.tdms 格式的，具体如何查看波形文件以及波形文件格式转换，

请参考 EasyGo DeskSim 的帮助文档中的工具箱说明部分。

暂停程序

点击软件工具栏的  按钮即可随时暂停程序。暂停后，我们可以看到界面
上的暂停按钮变成红色的，如下图所示：



注意：

- ◆ 暂停功能只是暂停主界面的波形更新或数值显示控件的数据更新，便于用户查看暂停瞬时前的一段时间的数据或波形；
- ◆ 暂停功能对实时仿真机中的程序不影响；
- ◆ 暂停功能对波形录制也不影响，上下位机的数据交互始终没有间断；

停止程序

点击软件工具栏的  按钮即可停止程序。

保存项目文件

全部运行完成后，为了便于下一次打开直接使用，我们可以对上述所有的配置操作进行保存管理，保存成 DeskSim 软件专属的项目文件 (*.desksim)。

点击工具栏中的  保存按钮，选择路径，修改名称，即可完成保存。下次
打开时，可以点击软件工具栏中的  打开按钮，载入以前保存的项目文件，或
者直接双击项目文件，直接打开软件并载入项目文件。