

第3章

FPGA开发流程

本章致力于深入解析现代硬件开发领域中极为关键的一部分——现场可编程门阵列(Field-Programmable Gate Array,FPGA)的开发过程。理解 FPGA 背后的基本概念、特性、内部结构以及它是如何工作的,对于深入掌握硬件设计和优化至关重要。

我们将深入了解 FPGA 的内部结构,包括它如何通过灵活的逻辑块、可编程的输入/输出以及配置逻辑实现复杂的电子设计。同时,我们会讨论 FPGA 工作的基本原理,解析它是如何在提供高度灵活性的同时保证性能和效率。本章会重点介绍 FPGA 的开发流程及其相关工具。我们将详述从设计思想到实际硬件落地的整个开发过程,包括设计、仿真、编译、配置和测试等关键步骤。我们也将探讨几种主要的 FPGA 开发工具及其特点,为设计师选择最合适的工具提供参考。

3.1 FPGA 的概念

在探讨现场 FPGA 之前,让我们深入了解 FPGA 到底是什么,它的特点,复杂的内部结构,以及它是如何工作的。本部分旨在为读者揭示 FPGA 技术的核心,带领大家一步步探索这一强大的硬件工具如何在电子设计和产品开发中扮演着不可替代的角色。通过这一系列的探讨,读者将能够获得关于 FPGA 基本概念的全面理解,为后续深入学习和应用 FPGA 技术打下坚实的基础。

3.1.1 FPGA 是什么

FPGA 即现场可编程门阵列,是一种可编程的数字逻辑芯片,它是在 PAL、GAL、CPLD 等可编程器件的基础上进一步发展的产物。FPGA 是作为专用集成电路(ASIC)领域中的一种半定制电路而出现的,既解决了定制电路的不足,又克服了原有可编程器件门电路数量有限的缺点。FPGA 芯片示意图如图 3.1 所示。

我们可以通过对 FPGA 编程实现大部分的数字功能。可以说在数字世界里它无所不能,就像乐高的积木一样可以搭建各种不同的功能模块,实现你所希望的各

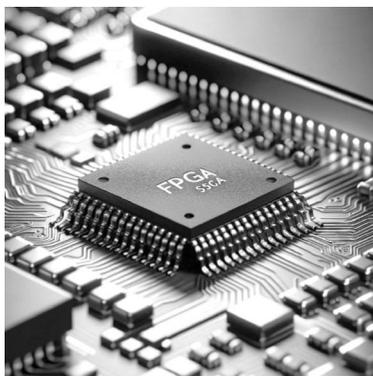


图 3.1 FPGA 芯片示意图

种功能。首先,你已经掌握最基本的数字逻辑知识,学会了一种用来构建各种功能的工具语言,其次,你要动脑(考验的是你的逻辑思维是否清晰),一个优秀的建筑师的作品是在脑子里勾画出来的,而不是拿积木碰运气拼凑出来的。

FPGA 是一种非常灵活的电子器件,它允许你通过编程来定义其硬件功能。想象一下,有一个电路板,上面有成千上万个小型电子开关(逻辑门),这些开关可以通过编程任意连接起来,形成不同的电路。

FPGA 的核心特点是它们可以在不改变物理硬件的情况下进行重新编程。这意味着你可以设计一个电路,将其通过编程方式加载到 FPGA 上,如果需要改变功能或改进设计,只需重新编程 FPGA 即可,而不需要重新制造整个电路板。这使得 FPGA 非常适合用于原型设计、测试新的电子设计或者用于教育和研究。

在电子和计算机工程领域,FPGA 经常被用于模拟复杂的电路系统,例如处理器、信号处理电路或者用于特定应用的定制电路。由于它们的高度可编程性和灵活性,FPGA 成为电子工程师和学生实验新想法的理想选择。简单来说,如果你想设计一个电子系统,但不确定最终需要什么功能,或者你希望能够快速迭代和测试你的设计,那么 FPGA 就是一个非常有用的工具。

FPGA 行业的主要厂商包括 AMD(收购 Xilinx)、Intel(收购 Altera)、Microsemi(现为 Microchip Technology 的一部分),以及 Lattice Semiconductor。以下是这些主要厂商的简要介绍。

- Xilinx: Xilinx 是 FPGA 行业的先驱和领导者之一,总部位于美国加州圣何塞。该公司于 1984 年推出了世界上第一款商用 FPGA,并持续在可编程逻辑设备领域进行创新。Xilinx 的产品广泛应用于通信、工业、科学和军事等领域。2021 年,Xilinx 被 AMD 收购。
- Intel(Altera): Altera 是另一家 FPGA 行业的重要玩家,于 1983 年成立,在硅谷有很大的影响力。2015 年,Intel 收购了 Altera,从而进入 FPGA 市场。Intel FPGA 产品(以前的 Altera 品牌)被广泛应用于数据中心、通信网络和工业系统中。
- Microsemi(Microchip Technology): Microsemi 是一家提供半导体和系统解决方案的公司,特别是在高性能和高可靠性应用方面。它在 FPGA 领域也有一席之地,尤其是在航空航天和国防市场。2018 年,Microchip Technology 收购了 Microsemi。
- Lattice Semiconductor: Lattice Semiconductor 是一家总部位于美国的公司,专注于低功耗、小尺寸和低成本的 FPGA 产品。Lattice 的 FPGA 主要用于消费电子、工业、通信、计算机网络和汽车市场。

国产 FPGA 起步较晚,发展相对国外落后不少,主要的厂商有高云半导体、安路科技、复旦微电子、紫光同创、京微齐力等,总体技术实力落后于国外。

3.1.2 FPGA 的特点

1. FPGA 和 CPLD 的区别

FPGA(现场可编程门阵列)和 CPLD(复杂可编程逻辑器件)是一回事吗?不是的,它们都是可编程的数字逻辑芯片,但是有着不同的特性。它们在结构和适用场景上有一些关键的区别。

1) 结构差异

- FPGA: 拥有更复杂的可编程逻辑块,通常包括查找表(LUTs)、寄存器和逻辑门。FPGA 的互连网络更加灵活和密集,这使得它们能够实现更复杂的逻辑功能和更高的处理速度。
- CPLD: 相比之下,CPLD 结构更为简单。它们通常由一个较小数量的较大的可编程逻辑块组成,这些逻辑块通过一个相对固定的互连阵列连接。CPLD 中的逻辑资源通常基于宏单元或逻辑单元,这些单元包含固定数量的逻辑门和简单的互连结构。

2) 逻辑容量和速度

- FPGA 通常提供更高的逻辑容量和更复杂的逻辑处理能力,适合于处理大型和复杂的设计,如数字信号处理、图像处理 and 自定义处理器逻辑。
- CPLD 由于其简单的结构,适用于较小和相对固定的逻辑应用,如简单的逻辑控制、状态机、接口管理等。

3) 编程和重配置性

- FPGA 支持较为复杂的重配置选项,可以容纳更大规模和更复杂的设计。它们通常通过硬件描述语言编程。
- CPLD 通常用于较简单的逻辑功能,编程相对简单,但其重配置能力和灵活性不如 FPGA。

4) 应用场景

- FPGA 通常用于需要高度灵活性和复杂数据处理的情况,例如在需要高性能计算、视频处理或者复杂的数字系统集成时。
- CPLD 更适合小型和固定逻辑应用,如简单的控制系统、I/O 扩展和固定逻辑替代,尤其是在对功耗和成本有严格要求的情况下。

总的来说,FPGA 提供了更高的灵活性和处理能力,适合复杂和高性能的应用;而 CPLD 则更小、更节能,适合简单的逻辑应用和成本敏感的情况。

2. FPGA 和处理器的区别

FPGA 和微处理器、微控制器是一回事么? 不是的。FPGA 相比于 PC 或单片机(无论是冯·诺依曼结构还是哈佛结构)的顺序操作有很大区别,基于 CPU 架构的微处理器或微控制器会执行预定义的指令(如加法、跳转、数据移动等)。这些指令是在软件层面上编写并存储在内存中的,处理器按顺序或根据需要读取并执行这些指令。FPGA 则是通过硬件描述语言编程的,这种语言描述的是电路的物理行为。FPGA 中的逻辑块和互连资源可以被配置为直接实现特定逻辑功能的电路用于计算和信号处理。

FPGA 的开发相对于传统 PC、单片机的开发有很大不同。虽然 FPGA 的开发需要使用硬件描述语言编程,但是编写的代码在 FPGA 内部实现后是一个个电路模块,各个模块可以并行运算。这导致 FPGA 开发入门较难,因为它需要的不只是编程能力,还有电路设计能力,所以 FPGA 开发需要从顶层设计、模块分层、逻辑实现、软硬件调试等多方面着手。

3. 为什么用 FPGA?

FPGA 具有灵活的开发周期、更低的设计迭代成本、更低的一次性工程费用,易于评

估和实现的可选设计架构,新产品上市时间快。相比于 ASIC 和 MCU, FPGA 具有以下优势:

- 功能强大,有大量并行处理结构。可以实现数字设计领域几乎所有的功能,如组合逻辑、时序逻辑、存储、处理器;现今的 FPGA 芯片集成了更多功能,比如 PLL 时钟产生、分配、驱动,支持各种高速接口规范的可编程 I/O 块,硬核化的 SPI/I²C 总线以及 ARM 内核等,增强的 DSP 单元,Altera 公司(现已被 Intel 收购)的 MAX10 甚至集成了串行 ADC 能够监测环境的温度。
- 开发快,上市时间短,适合原型设计或小批量产品。FPGA 高度灵活,设计实现和后续优化的灵活性可以显著影响项目的进度、设计的复杂度,降低项目的风险,便于更改和升级。
- 重复编程/配置,灵活、快速。集成度高,可以通过选用不同规模的器件实现自己所需要的功能,内部功能模块之间的通信和接口的速度、性能都会较多个分立的芯片之间互连有明显的改善,节省板卡空间,便于调试。

3.1.3 FPGA 的内部结构

FPGA 的关键组件,包括可编程逻辑块 (Logic Blocks)、可编程互连资源 (Programmable Interconnect Resources)、输入/输出块 (I/O 块)、存储元件、时钟管理、配置逻辑、其他特殊功能块。FPGA 内部结构如图 3.2 所示。

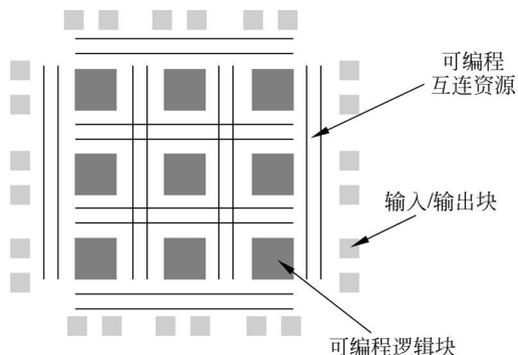


图 3.2 FPGA 内部结构

- 可编程逻辑块: FPGA 的核心部分是逻辑单元,通常称为逻辑单元或逻辑块。这些是小型、可编程的数字逻辑单元,可以执行简单的逻辑操作,如与、或、非等。逻辑单元可以被编程组合来实现更复杂的功能。
- 可编程互连资源: 为了使逻辑单元之间能够相互通信和连接, FPGA 包含了大量的可编程互连资源。这些资源包括一系列的编程线路和开关,可以根据需要配置来创建不同的数据路径。
- 输入/输出块: 这些块连接到 FPGA 的外围引脚,允许 FPGA 与外部世界通信。I/O 块可以被配置为支持各种不同类型的外部接口标准。
- 存储元件: 大多数 FPGA 包含一些形式的内部存储元件,如触发器、存储器块(如 RAM 块)和寄存器。这些元件可用于存储数据和中间计算结果。
- 时钟管理: FPGA 通常包含专用的时钟管理模块,如相位锁定环(PLLs),用于生成

和管理时钟信号。这些时钟信号对于同步 FPGA 内部操作至关重要。

- 配置逻辑：FPGA 在上电时通过一个称为配置过程的过程被编程。配置数据通常存储在外部存储器中，比如闪存，当 FPGA 上电时，这些数据被用来设置逻辑块、互联资源和 I/O 块的功能。
- 其他特殊功能块：一些 FPGA 还包含专用的硬件模块，如数字信号处理(DSP)块、高速串行接口等，这些可以用于特定类型的计算任务。

3.1.4 FPGA 是如何工作的

FPGA 是基于查找表结构，采用了逻辑单元阵列的模式来实现组合逻辑，每个查找表连接到一个 D 触发器的输入端，触发器再来驱动其他逻辑电路或驱动 I/O，由此构成了既可实现组合逻辑功能，又可实现时序逻辑功能的基本逻辑单元模块，这些模块间利用金属连线互相连接或连接到 I/O 块。FPGA 的逻辑是通过向内部静态存储单元加载编程数据来实现的，存储在存储器单元中的值决定了逻辑单元的逻辑功能以及各模块之间或模块与 I/O 块间的连接方式，并最终决定了 FPGA 所能实现的功能，FPGA 允许无限次的编程。

以上内容显得有些晦涩难懂，我们可以这样理解，FPGA 就像一块空白的数字电路板，你可以根据需要来“绘制”上面的电路。当你开机时，它会从一个叫做配置文件的东西中加载设计。这个文件告诉 FPGA 如何连接它内部的小组件，这些组件包括可以执行简单逻辑操作(比如加法或乘法)的逻辑单元，以及连接这些单元的线路。你可以把它想象成乐高积木，你可以用这些积木按照图纸搭建出各种各样的结构。

FPGA 的亮点在于它的灵活性。就像你可以重复使用同一堆乐高积木来构建不同的模型一样，FPGA 可以被重新编程来执行各种不同的任务，从简单的数据处理到复杂的控制系统。

3.2 FPGA 的开发流程与工具

在 FPGA 的开发过程中，熟悉和掌握正确的开发流程与工具是至关重要的。本节将针对 FPGA 的开发流程与工具进行深入讨论，旨在为那些寻求专业知识和技能提升的硬件工程师和开发人员提供宝贵的资源与指南。通过本部分的学习，读者将能够获得关于 FPGA 开发流程和工具的深入理解，学习到如何选择和使用合适的工具，以及如何高效地管理整个开发过程。

3.2.1 FPGA 的开发流程

FPGA 的开发流程通常可以分为几个主要步骤。

- 需求分析与设计规划：明确 FPGA 设计的目标 and 需求，进行初步的设计规划。
- 硬件描述语言编写：使用硬件描述语言(如 VHDL 或 Verilog)编写设计代码，描述硬件的逻辑和功能。
- 代码仿真：在编写代码的过程中进行仿真测试，确保代码按预期工作。
- 综合：将硬件描述语言代码转换成电路网表，这一步骤通常由 FPGA 设计软件自动完成。

- 布局布线(Place & Route): 确定电路中各个元素的物理位置,并进行布线,确保它们能够正确连接。
- 生成比特流: 将布局与布线的结果转换成 FPGA 芯片可以理解的比特流文件。
- 下载配置: 将比特流下载到 FPGA 芯片上,对芯片进行配置。
- 硬件测试与验证: 在实际硬件上测试 FPGA 设计,确保它在真实环境中按预期工作。
- 迭代优化: 根据测试结果对设计进行必要的修改和优化。

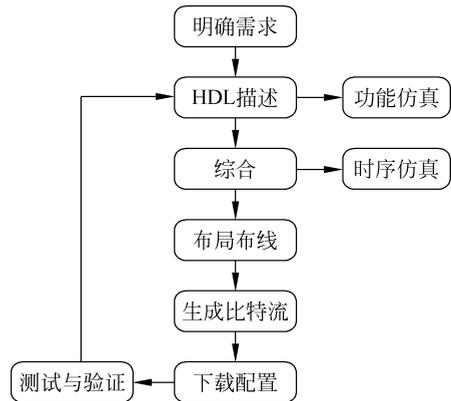


图 3.3 FPGA 开发流程图

FPGA 开发流程图如图 3.3 所示。

3.2.2 FPGA 开发工具

1. EDA 工具

在 FPGA 设计的各个阶段,FPGA 厂商和 EDA 软件公司提供了很多优秀的 EDA 工具,尤其是 FPGA 厂商提供的集成开发环境(IDE)。熟练掌握这些设计工具能够有效提高设计的效率,但是必须明白这些 EDA 软件只是一个工具,核心的 FPGA 设计流程是不变的。

目前主流的 FPGA 厂商 Xilinx(现被 AMD 收购)、Altera、Intel、Lattice,都有独立的开发平台,每个产品系列的工具会有不同,安装时需要选择对应的软件工具。

1) Lattice 公司

支持 Windows 和 Linux 平台的 Lattice Diamond、Lattice Radiant。Lattice Diamond 软件界面如图 3.4 所示。

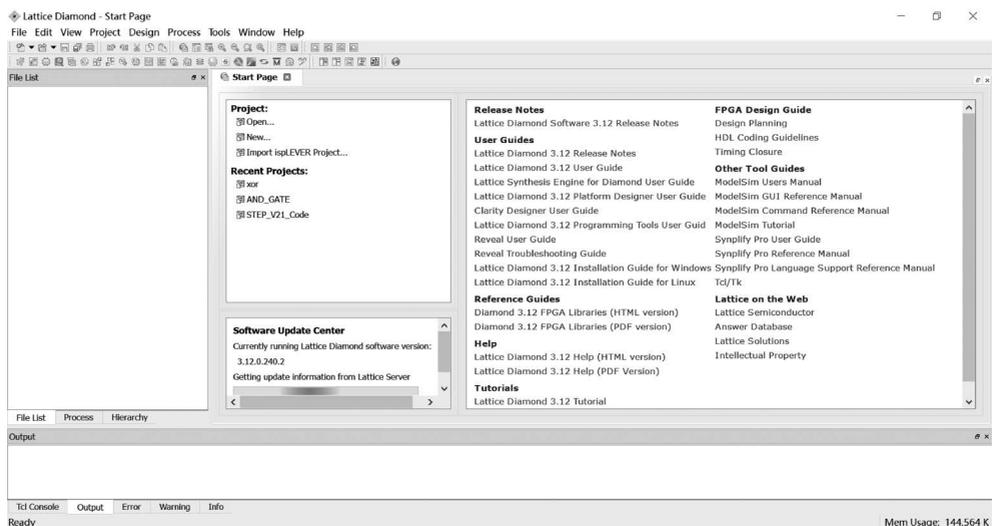


图 3.4 Lattice Diamond 软件界面

2) Intel FPGA

支持 Windows 和 Linux 平台的 Quartus Prime Lite Edition。Quartus Prime Lite Edition 软件界面如图 3.5 所示。

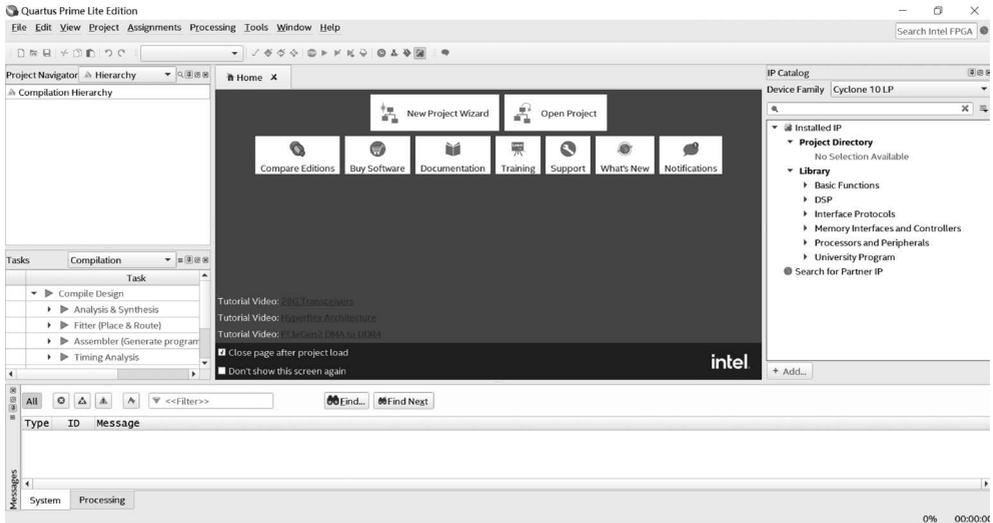


图 3.5 Quartus Prime Lite Edition 软件界面

3) Xilinx 公司的 Vivado Design Suite

Vivado Design Suite 软件界面如图 3.6 所示。

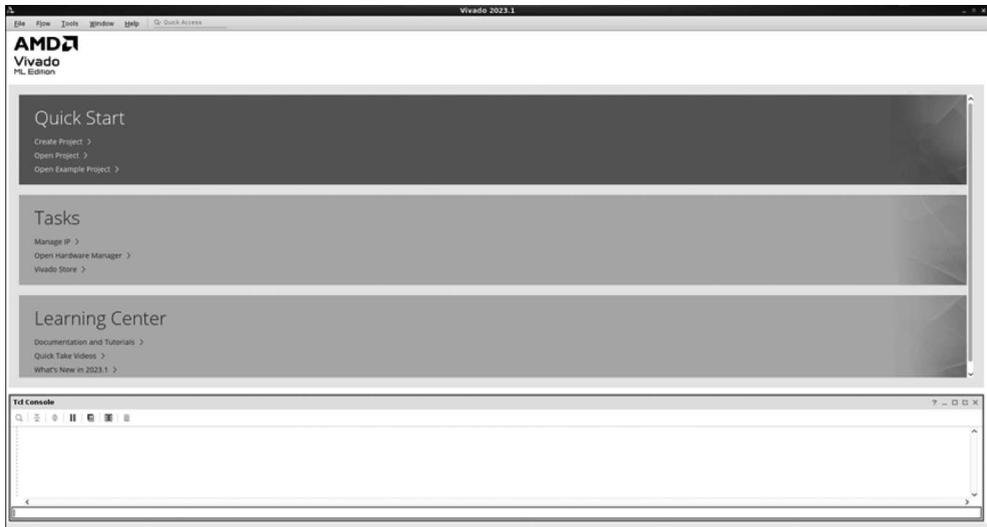


图 3.6 Vivado Design Suite 软件界面

4) 在线编译工具

如苏州思得普科技有限公司开发的“小脚丫”在线 FPGA 综合设计平台,在浏览器里运行,因此可以支持任何一种操作系统,且能够支持 Lattice 的 XO2 系列 FPGA 和 Altera 的 MAX10 系列 FPGA,适合初学者使用。“小脚丫”在线 FPGA 综合设计平台界面如图 3.7 所示。



图 3.7 “小脚丫”在线 FPGA 综合设计平台界面

除原厂的集成开发环境之外,许多第三方的专业工具也可以用来做 FPGA 开发,最常用的是数字电路的综合工具和仿真工具。比如综合工具 Synplify 和仿真工具 Modelsim。

5) 综合工具 Synplify Premier

Synplify Premier 软件界面如图 3.8 所示。



图 3.8 Synplify Premier 软件界面

Synplify Premier 是一款由 Synopsys 公司开发的行业领先的 FPGA 综合工具。综合是将高级硬件描述语言(如 VHDL 或 Verilog)编写的设计转换为门级网表的过程,这个网表随后用于 FPGA 或 ASIC 的布局与布线。

Synplify FPGA 综合工具以其能对 FPGA 设计产生高性能和低成本而成为业界的标准工具。Synplify 软件支持最新的 VHDL 和 Verilog 语言结构,包括 System Verilog 和 VHDL-2008。该软件也支持多种不同的 FPGA 架构,如 Altera、Achronix、Lattice、Microsemi and Xilinx。

Synplify Premier 综合过程包括三方面内容。

(1) 对 HDL 源代码进行编译, Synplify Premier 将输入的 HDL 源代码翻译成 boolean 表达式。

(2) 对编译的结果优化, 通过逻辑优化消除冗余逻辑和复用模块, 这种优化是针对逻辑关系的, 与具体器件无关。

(3) 对优化的结果进行逻辑映射与结构层次上的优化, 最后生成网表; Synplify 将编译生成的逻辑关系映射成 FPGA 的底层硬件模块和原语(primitive), 生成网表并优化。

6) ModelSim

ModelSim 软件界面如图 3.9 所示。

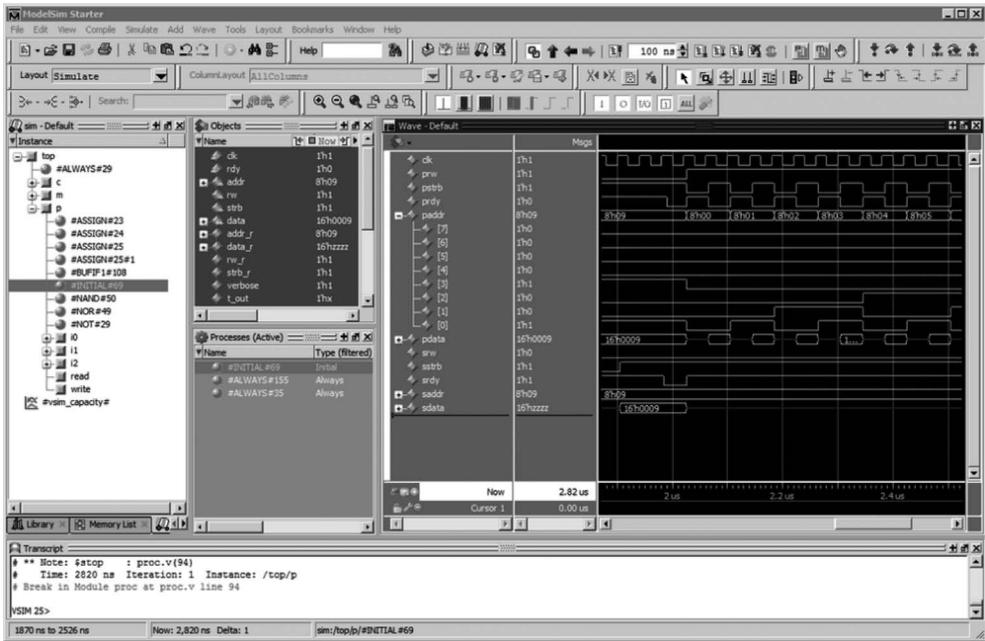


图 3.9 ModelSim 软件界面

ModelSim 是由 Mentor Graphics(现为 Siemens EDA 的一部分)开发的一款行业领先的硬件描述语言(HDL)仿真软件。它被广泛用于验证 FPGA 和 ASIC 设计的正确性。以下是 ModelSim 的一些关键特点。

(1) 多语言支持: ModelSim 支持主流的硬件描述语言, 包括 VHDL、Verilog 和 SystemVerilog, 使其适用于多种设计流程。

(2) 高效的仿真性能: ModelSim 提供高效的仿真性能, 这对于减少复杂设计的验证时间尤为重要。

(3) 用户友好的界面: 它具有直观的用户界面, 包括代码编辑器、波形查看器和调试工具, 使得设计师可以轻松地编写、测试和分析他们的 HDL 代码。

(4) 强大的调试功能: ModelSim 提供了广泛的调试功能, 包括断点、单步执行和变量跟踪, 这些功能使得定位和修复设计错误更加高效。

(5) 波形分析: 软件能够生成详细的波形输出, 帮助设计师可视化信号在时间上的变化, 从而更容易理解和调试设计。

(6) 集成开发环境(IDE): ModelSim 通常与其他 EDA 工具集成,如综合工具和时序分析工具,形成一个完整的设计和验证流程。

(7) 广泛的适用性: 从学术研究到工业应用,ModelSim 在电子设计自动化(EDA)领域内有着广泛的应用。

ModelSim 的这些特点使其成为了 FPGA 和 ASIC 设计师进行 HDL 代码验证和仿真的首选工具之一。它不仅适用于大型复杂设计的验证,也适合小型或学术项目,使其在电子设计领域内极为受欢迎。

2. 小脚丫(STEP)FPGA 学习开发平台

FPGA 的学习门槛相比单片机要高,不管是硬件成本还是设计难度,对于初学者来说都不够友好。小脚丫(STEP)FPGA 开发板是苏州思得普信息科技有限公司专门针对 FPGA 初学者打造的一款性价比高、学习门槛低的学习模块系列。该系列中所有板子的大小兼容标准的 DIP40 封装,尺寸只有 52mm×18mm,非常便于携带;并通过 USB 端口进行供电和下载,板上选用的芯片兼具了 FPGA 和 CPLD 的优点,瞬时上电启动,无需外部重新配置 FPGA,是学习数字逻辑绝佳的选择。而且能够直接插在面包板上或以模块的方式放置在其他电路板上,即插即用,大大简化了系统的设计。小脚丫(STEP)FPGA 开发板如图 3.10 所示。

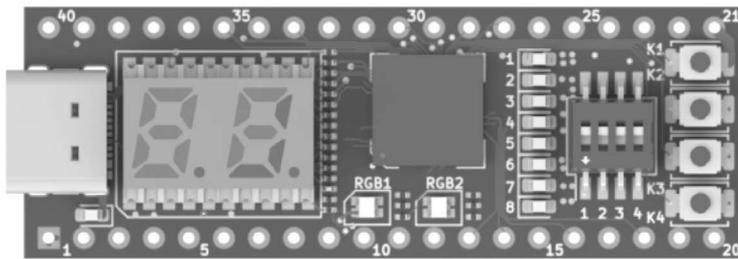


图 3.10 小脚丫(STEP)FPGA 开发板

其主要产品有两个系列: 基于 Lattice XO2-4000HC FPGA 的 STEP MXO2 开发板和基于 Intel(Altera) MAX10M02/08 FPGA 的 STEP MAX10 开发板。两个产品系列除核心的 FPGA 芯片不一样以外,其他板载资源和外观都是相同的。

两款 FPGA 都是小容量的,逻辑资源在 10KB 以下,并且内部集成了 Flash,而一般的 FPGA 都需要用户单独外配一片 Flash 用于存储配置文件,FPGA 在上电时从外部 Flash 加载配置文件并运行。STEP FPGA 系列开发板所选的两款 FPGA 无需用户外配 Flash,配置文件之间下载到芯片内部,集成度较高。

Lattice 系列开发板小脚丫 STEP-MXO2 硬件资源介绍如图 3.11 所示。

硬件资源如下。

(1) 核心器件: Lattice LCMXO2-4000HC-4MG132。

- 132 脚 BGA 封装,引脚间距 0.5mm,芯片尺寸 8mm×8mm;
- 上电瞬时启动,启动时间<1ms;
- 4320 个 LUT 资源,96Kb 用户闪存,92Kb RAM;
- 2+2 路 PLL+DLL;
- 嵌入式功能块(硬核): 一路 SPI、一路定时器、两路 I²C;

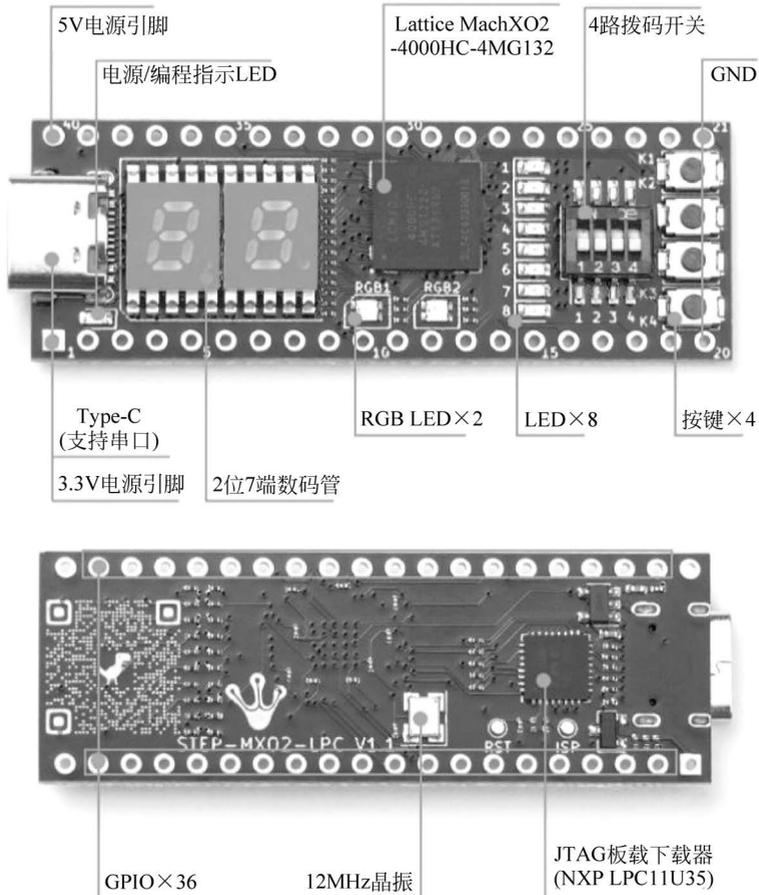


图 3.11 小脚丫 STEP-MX02 开发板硬件资源介绍

- 支持 DDR/DDR2/LPDDR 存储器；
 - 104 个可热插拔 I/O 接口；
 - 内核电压 2.5~3.3V。
- (2) 板载资源。
- 两位 7 段数码管；
 - 两个 RGB 三色 LED；
 - 8 路用户 LED；
 - 4 路拨码开关；
 - 4 路按键；
 - 外部时钟频率为 12MHz；
 - 36 个用户可扩展 I/O 接口(其中包括一路 SPI 硬核接口和一路 I²C 硬核接口)；
 - 支持的开发工具 Lattice Diamond；
 - 一路 Micro USB 接口；
 - 板上集成 FPGA 编程器；
 - 支持 MICO32/8 软核处理器；
 - 板卡尺寸 52mm×18mm。

Intel(Altera)系列开发板如图 3.12 所示。

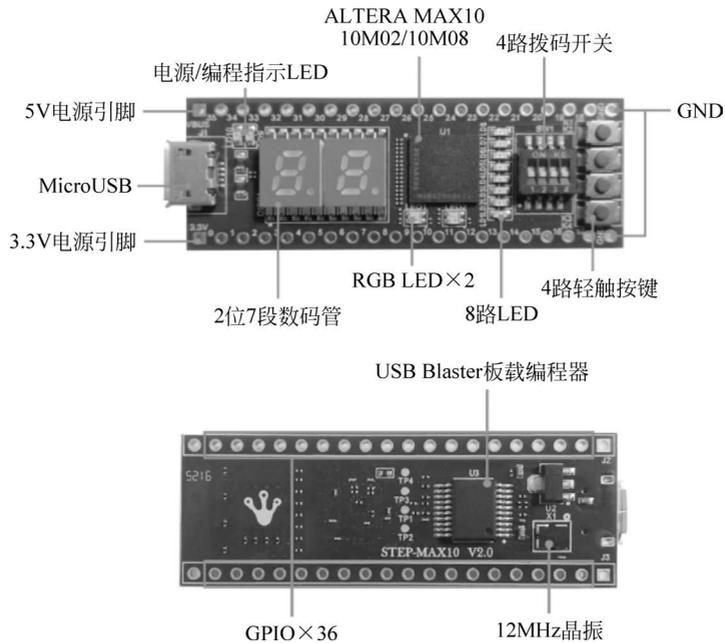


图 3.12 Intel(Altera)系列开发板

硬件资源如下。

(1) 核心器件：Intel(Altera) 10M08SCM153。

- 153 脚 BGA 封装,引脚间距 0.5mm,芯片尺寸 8mm×8mm;
- 上电瞬时启动;
- 8000 个 LE 资源,最大 172KB 用户闪存,378Kb RAM;
- 2 路 PLL;
- 24 路硬件乘法器;
- 支持 DDR2/DDR3L/DDR3/LPDDR2 存储器;
- 112 个用户 GPIO 接口;
- 3.3V 电压供电。

(2) 板载资源。

- 两个 RGB 三色 LED;
- 2 路用户 LED;
- 4 路拨码开关;
- 2 路按键;
- 36 个用户可扩展 I/O 接口;
- 支持的开发工具 Intel Quartus Prime;
- 一路 Micro USB 接口;
- 一个 10 引脚的 JTAG 编程接口;
- 板卡尺寸 52mm×18mm。

3.3 FPGA 开发流程示例

3.3.1 Lattice Diamond 开发 FPGA 实例(以 STEP MXO2 开发板为例)

下面我们开始可编程逻辑的开发,以控制 LED 交替闪烁为例,完成自己的第一个程序。

1. Lattice Diamond 软件的安装

Lattice Diamond 软件的安装注册激活过程我们不再赘述,读者可以在 Lattice 官网注册登录之后获取 Diamond 的安装包,通过邮箱和计算机的网络地址免费申请许可证,打开软件激活以后即可使用。下面我们重点讲解使用 Diamond 开发 FPGA 的流程。

2. 新建工程

1) 新建 Project

双击运行 Diamond 软件,首先新建工程:选择 File→New→Project→Next 选项。Diamond 软件新建 FPGA 工程页面如图 3.13 所示。

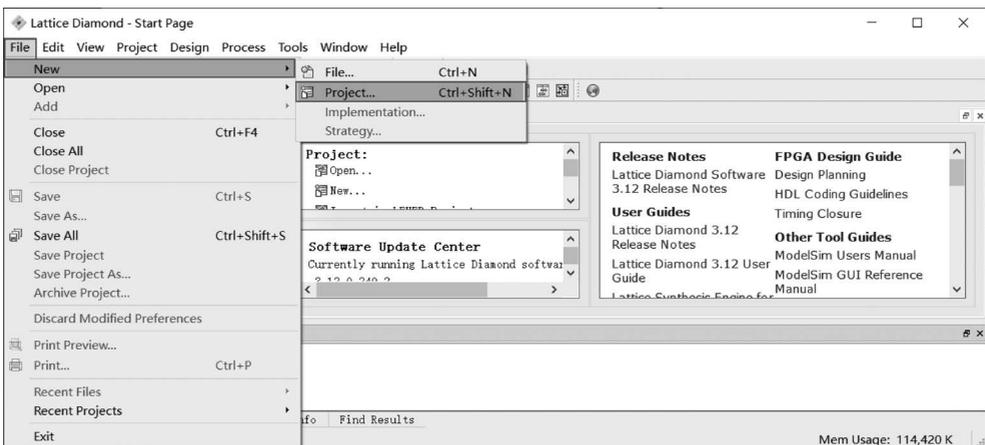


图 3.13 Diamond 软件新建 FPGA 工程页面

2) 工程命名

我们将新工程命名为 LED_shining,选择工程保存目录。Diamond 软件新建工程命名-保存目录页面如图 3.14 所示。

3) 添加设计文件

下一步可以添加相关设计文件或约束文件(如果已经有设计文件和约束文件,我们可以选择添加进工程):这里我们新建工程,没有相关文件,不需添加,直接单击 Next 按钮。

4) 器件选择

按照 STEP FPGA 开发板器件 LCMXO2-4000HC-4MG132C 配置(器件型号必须确认正确,否则在引脚设置时会报错)。Diamond 软件新建 FPGA 工程-选择器件型号页面如图 3.15 所示。

5) 选择综合工具

Synplify Pro(第三方)和 Lattice LSE(原厂)都可以,我们就使用 Lattice LSE。

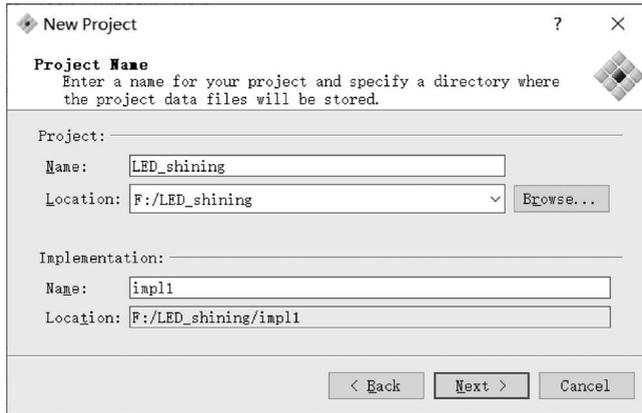


图 3.14 Diamond 软件新建工程命名-保存目录页面

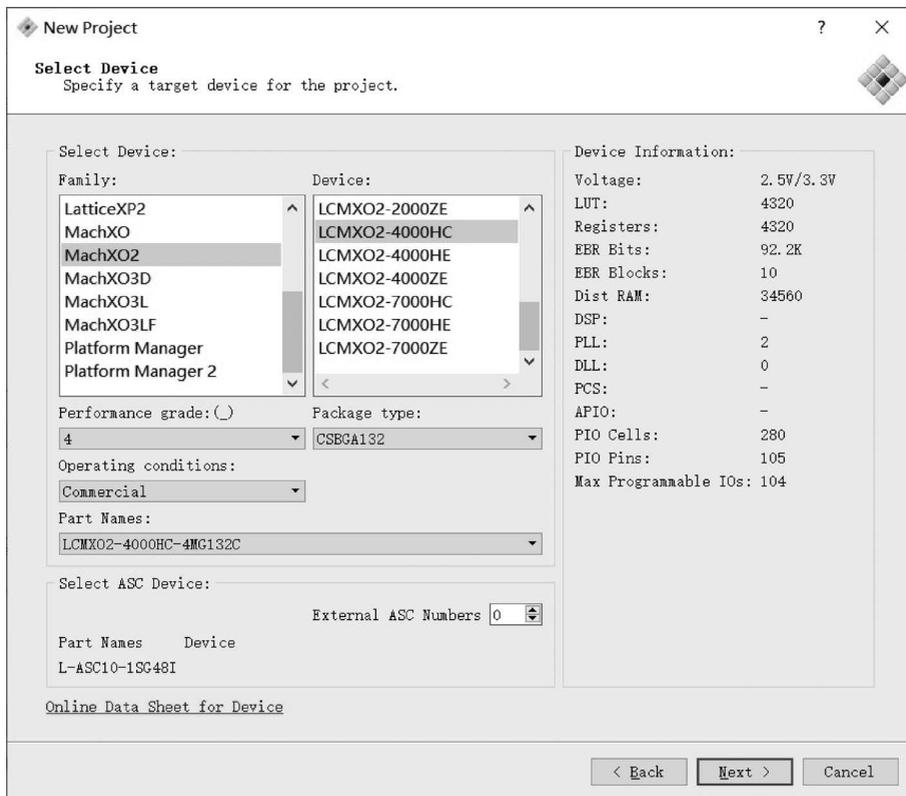


图 3.15 Diamond 软件新建 FPGA 工程-选择器件型号页面

Diamond 软件新建 FPGA 工程-选择综合工具页面如图 3.16 所示。

6) 工程信息确认

上面选择的所有信息都在这里, 确认没有问题, 直接单击 Finish 按钮。Diamond 软件新建 FPGA 工程-完成新建页面如图 3.17 所示。

3. 添加设计文件

工程已经建好, 我们下面添加设计文件, 选择 File→New→File 选项。选择 Verilog

Files(选择自己使用的硬件描述语言),Name 填写 LED_shining,然后单击 New 按钮,这样我们就创建了一个新的设计文件 LED_shining.v,然后我们就可以在设计文件中进行编程了。Diamond FPGA 开发-添加设计文件页面如图 3.18 所示。

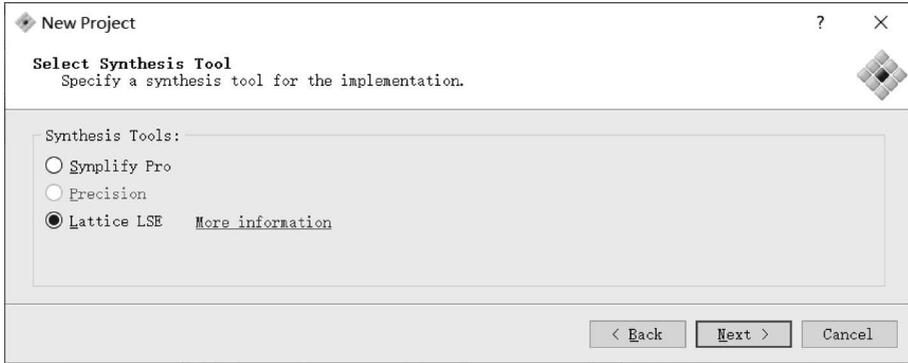


图 3.16 Diamond 软件新建 FPGA 工程-选择综合工具页面

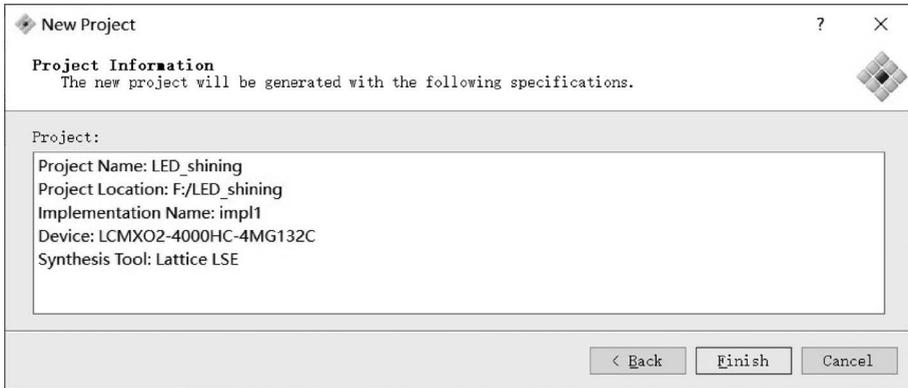


图 3.17 Diamond 软件新建 FPGA 工程-完成新建页面

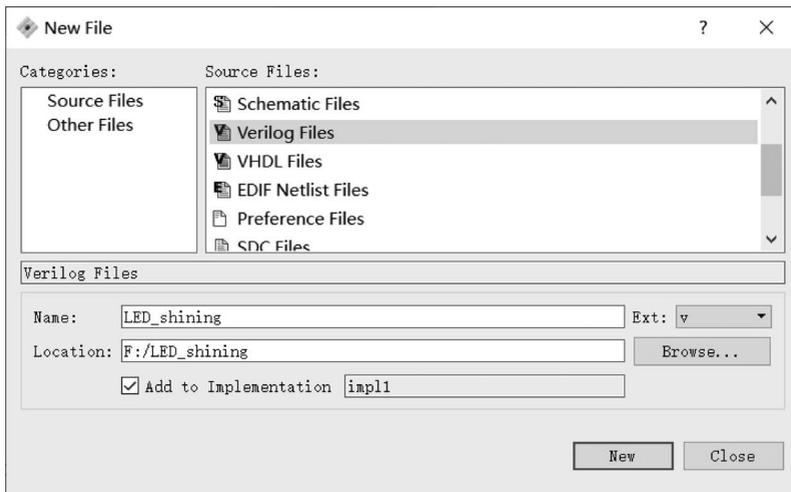


图 3.18 Diamond FPGA 开发-添加设计文件页面

程序源码如代码 3.1 所示,将代码复制到设计文件 LED_shining.v 中,并保存。

代码 3.1 LED_shining 模块 Verilog 代码

```

module LED_shining (
    input clk,        //clk = 12mhz
    input rst_n,     //rst_n, active low
    output led1,     //led1 output
    output led2     //led2 output
);

parameter CNT_1S = 12_000_000 - 1;    //time 1S
parameter CNT_05S = CNT_1S >> 1;    //time 0.5S

reg [23:0] cnt;
always @(posedge clk or negedge rst_n) begin
    if (!rst_n) cnt <= 1'b0;
    else if (cnt >= CNT_1S) cnt <= 1'b0;
    else cnt <= cnt + 1'b1;
end

wire clk_div = (cnt > CNT_05S)? 1'b1 : 1'b0;

assign led1 = clk_div;
assign led2 = ~clk_div;

endmodule

```

4. 综合

程序编写完成,需要综合,在软件左侧 Process 栏,选择 Process,双击 Synthesis Design,对设计进行综合,综合完成后 Synthesis Design 显示对勾(如果显示叉号,说明代码有问题,根据提示修改代码),Diamond FPGA 开发-综合页面如图 3.19 所示。

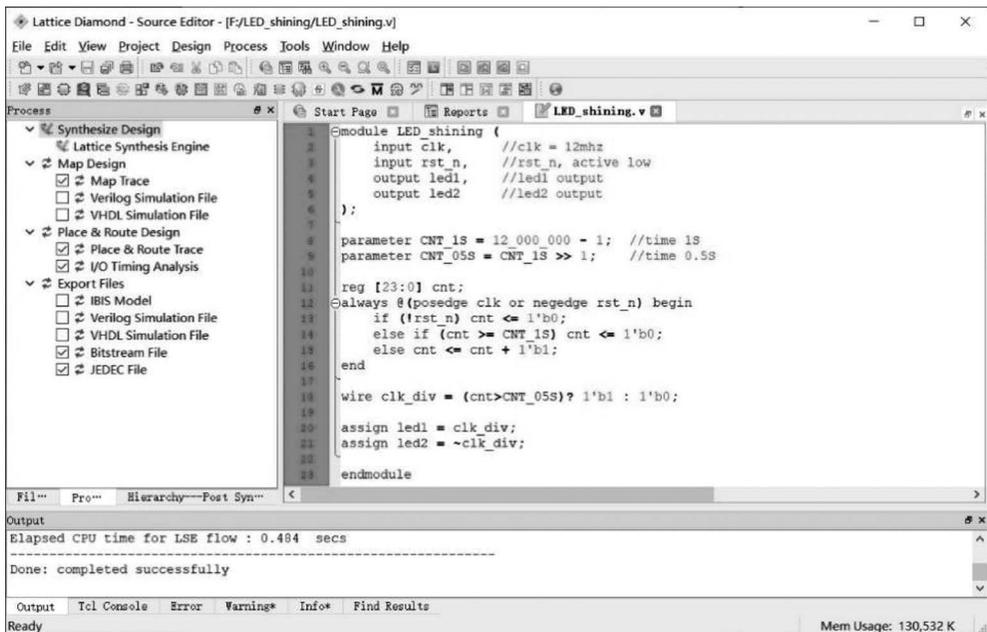


图 3.19 Diamond FPGA 开发-综合页面

通过综合工具,我们的代码就被综合成了电路,生成的具体电路,我们可以通过选择 Tools→Netlist Analyzer 查看(仅限 Lattice 的综合工具,第三方综合工具无法查看),Diamond FPGA 开发-查看综合结果页面如图 3.20 所示。

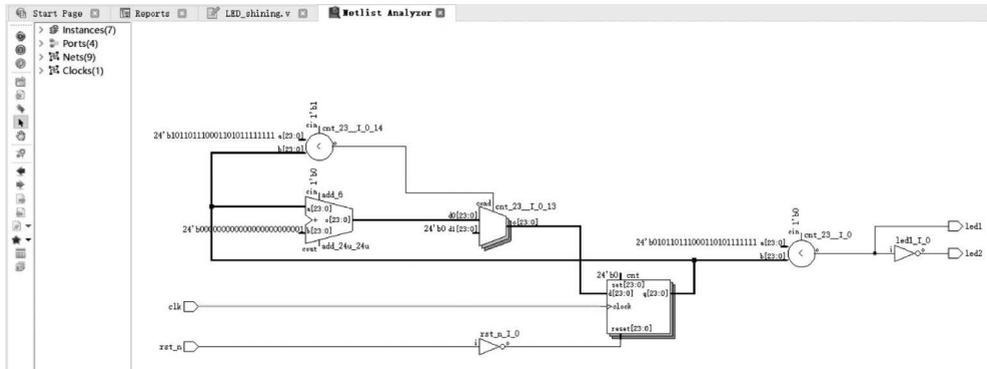


图 3.20 Diamond FPGA 开发-查看综合结果页面

5. 分配引脚和布局布线

综合生成电路后,分配引脚,选择 Tools→Spreadsheet View 选项,按照图 3.21 分配 FPGA 引脚,然后设置 IOTYPE 为 LVC MOS33,保存。Diamond FPGA 开发-分配引脚页面如图 3.21 所示。

The screenshot shows the 'Spreadsheet View' of the pin assignments. The table below represents the data shown in the spreadsheet.

	Name	Group Bv	Pin	BANK	BANK VCC	VREF	IO TYPE	PULLMODE	DRIVE	SLEWRATE	CLAMP
1	All Ports	N/A	N/A	N/A	N/A	N/A	LVC MOS33		N/A	N/A	
1.1	Input	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
1.1.1	Clock	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
1.1.1.1	clk	N/A	C1	5	Auto	N/A	LVC MOS33	DOWN	NA	NA	ON
1.1.2	rst_n	N/A	L14	1	Auto	N/A	LVC MOS33	DOWN	NA	NA	ON
1.2	Output	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
1.2.1	led1	N/A	N13	1	Auto	N/A	LVC MOS33	DOWN	8	SLOW	OFF
1.2.2	led2	N/A	M12	1	Auto	N/A	LVC MOS33	DOWN	8	SLOW	OFF

At the bottom of the window, there are tabs for 'Port Assignments', 'Pin Assignments', 'Clock Resource', 'Route Priority', 'Cell Mapping', 'Global Preferences', and 'Timing Pr'.

图 3.21 Diamond FPGA 开发-分配引脚页面

在软件左侧 Process 栏,选择 Process,直接双击 Place&Route Design 完成布局布线,如果布局布线报错则查看引脚分配和器件型号是否匹配。Diamond FPGA 开发-布局布线页面如图 3.22 所示。

6. 输出配置文件

双击 Export Files,所有布局布线输出依次完成。结束后,所有选项显示对勾。Diamond FPGA 开发-生成配置文件页面如图 3.23 所示。

到此,我们完成了第一个程序流文件的生成,下面可以下载到 FPGA 中。

7. 下载配置到 FPGA

STEP MXO2 开发板的编程芯片已经集成到小脚丫开发板上,因此只需要一根 USB 线和计算机相连,就可以完成供电和编程的功能,驱动安装好以后就可以开始编译下载程序了。

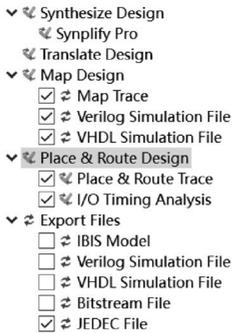


图 3.22 Diamond FPGA 开发-布局布线页面

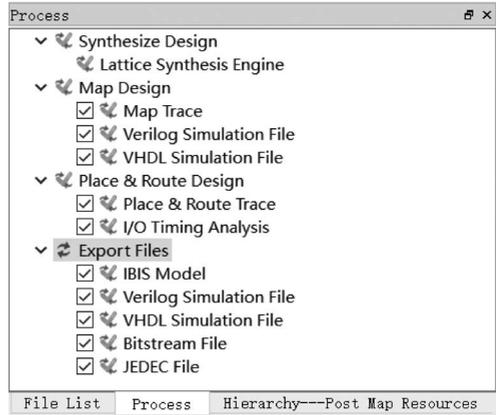


图 3.23 Diamond FPGA 开发-生成配置文件页面

将编译完成的程序下载到开发板,将开发板、下载器和计算机连接。Diamond FPGA 开发板下载连接如图 3.24 所示。



图 3.24 Diamond FPGA 开发板下载连接

选择 Tools→Programmer 选项,选择下载器 HW-USB-2B(FTDI),然后单击 OK,进入 Programmer 界面。在 Programmer 界面,单击右侧 Detect Cable,自动检测 Cable 显示 HW-USB-2B(FTDI),然后单击图 3.25 中的 Program。Diamond FPGA 开发-程序下载页面如图 3.25 所示。

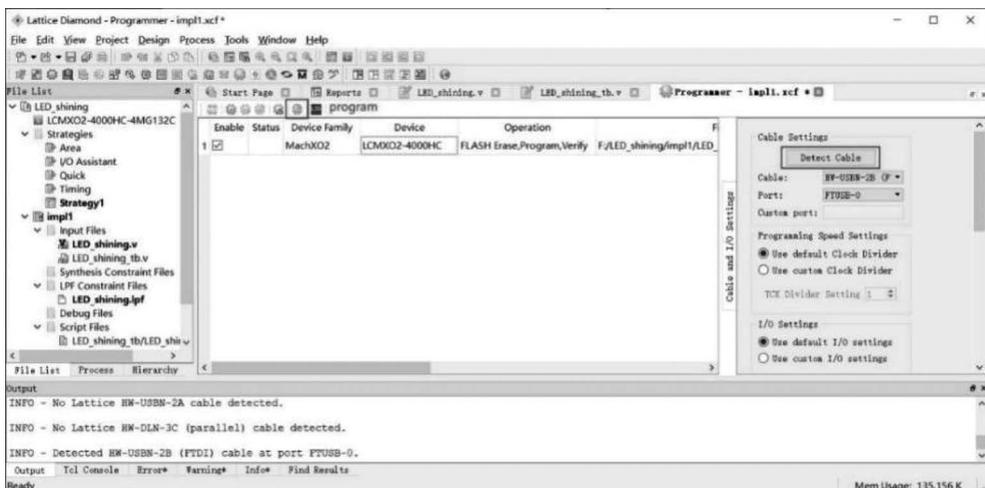


图 3.25 Diamond FPGA 开发-程序下载页面

显示 PASS,加载完成,观察 STEP MXO2 开发板的 LED 交替闪烁,则表示成功了。

8. 设计仿真

上面我们完成了整个工程的开发过程,例程较为简单,对于复杂的工程开发需要预仿真和后仿真等,保证最终的程序设计逻辑和时序符合我们的设计要求。仿真软件很多,这里我们使用软件自带的 Modelsim 软件进行功能仿真:首先我们添加 testbench 文件,和前面添加设计文件一样,选择 File→New→File→Verilog Files 选项,Name 填写 LED_shining_tb,然后单击 New 按钮。FPGA 设计仿真-添加仿真文件页面如图 3.26 所示。

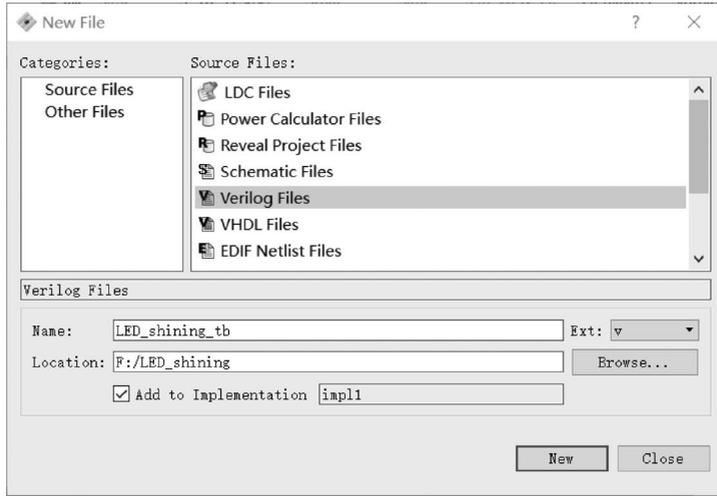


图 3.26 FPGA 设计仿真-添加仿真文件页面

测试源码如下,复制到 LED_shining_tb.v 文件并保存。为了方便仿真,我们在 LED_shining_tb.v 调用 LED_shining 模块时将 CNT1S 重新赋值为 19。LED_shining 模块仿真文件 LED_shining_tb.v 如代码 3.2 所示。

代码 3.2 LED_shining 模块仿真文件 LED_shining_tb.v

```

`timescale 1ns / 100ps
module LED_shining_tb;
parameter CLK_PERIOD = 10;
reg clk;
initial clk = 1'b0;
always # (CLK_PERIOD/2) clk = ~clk;

reg rst_n; //active low
initial begin
    rst_n = 1'b0;
    # 20;
    rst_n = 1'b1;
end

wire led1, led2;
LED_shining # (.CNT_1S ( 19 )) u_LED_shining (
    .clk          ( clk          ),
    .rst_n        ( rst_n        ),

```

```

.led1      ( led1  ),
.led2      ( led2  )
);

endmodule

```

然后在软件左侧 Process 栏,选择 File List,找到 LED_shining_tb.v(必须保存过),右击,选择 Include for→Simulation 选项。FPGA 设计仿真-设置仿真文件属性页面如图 3.27 所示。

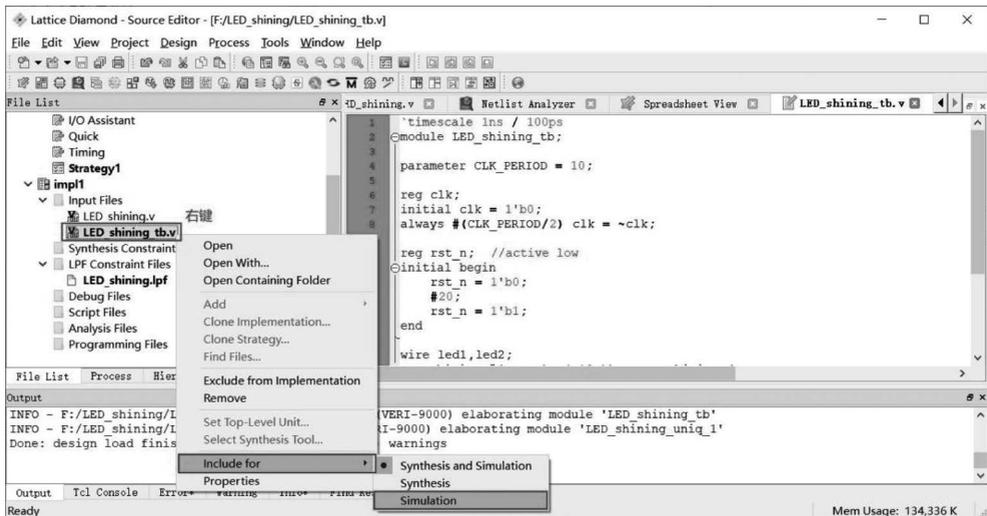


图 3.27 FPGA 设计仿真-设置仿真文件属性页面

准备工作完成,我们选择 Tools→Simulation Wizard→Next 选项,开始建立仿真工程。Lattice Diamond 3.12 版本软件自带 ModelSim 仿真工具,直接调用 ModelSim(默认),工程名称填写 LED_shining_tb,工程路径默认即可,然后单击 Next 按钮。FPGA 设计仿真-选择仿真工具页面如图 3.28 所示。

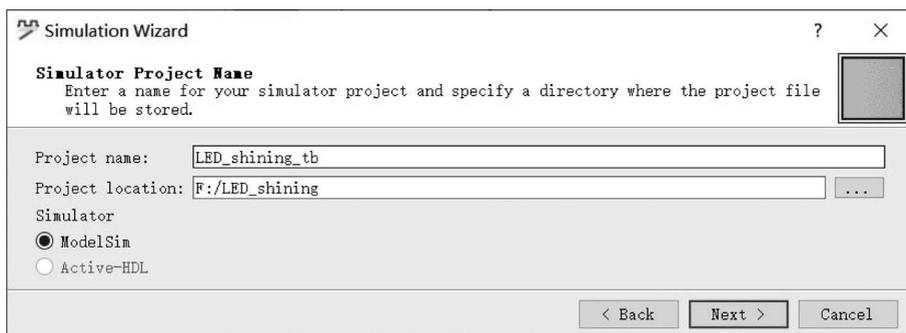


图 3.28 FPGA 设计仿真-选择仿真工具页面

选择 RTL 级仿真。下一步勾选 Copy Source to Simulation Directory,然后单击 Next 按钮,调用 ModelSim 软件。FPGA 设计仿真-选择 RTL 级仿真页面如图 3.29 所示。

启动 ModelSim 软件,可以直接查看 testbench 文件中变量的时序变化,想要看 LED_shining 模块中的变量的时序,可以通过图 3.30 中的步骤添加信号至 WAVE 窗口。FPGA 设计仿真-在 ModelSim 中添加观察信号页面如图 3.30 所示。

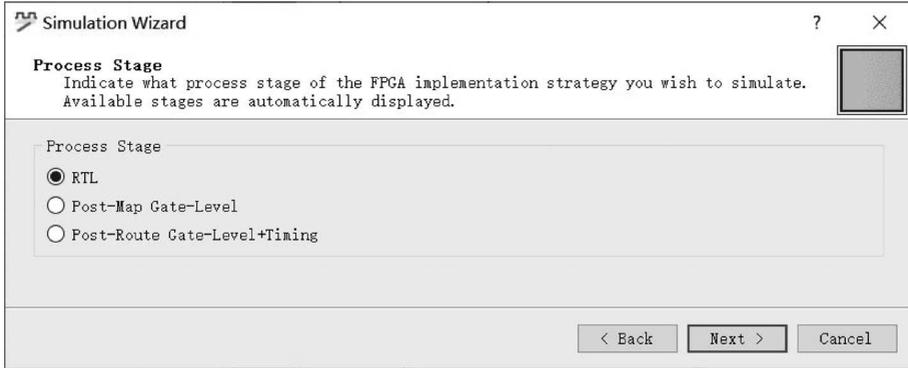


图 3.29 FPGA 设计仿真-选择 RTL 级仿真页面

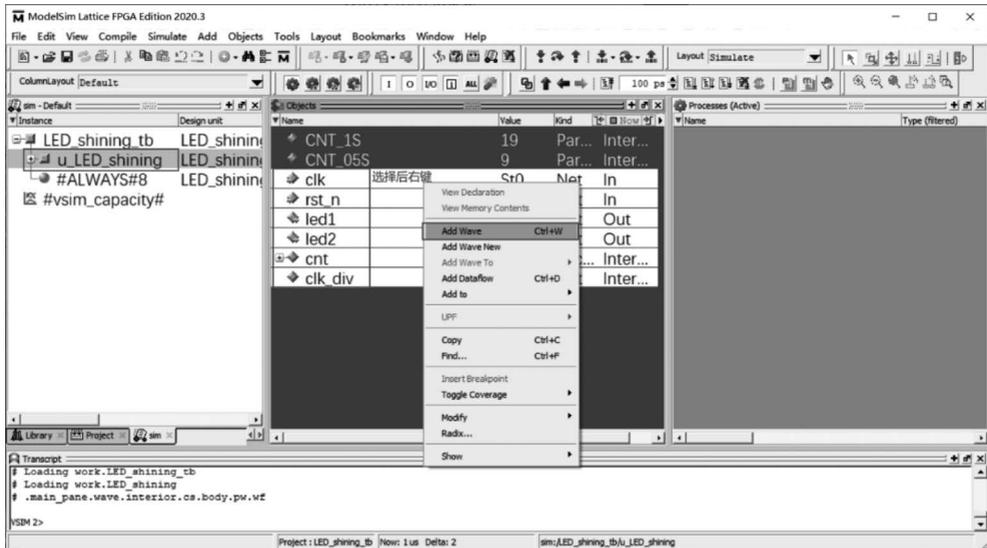


图 3.30 FPGA 设计仿真-在 ModelSim 中添加观察信号页面

在 WAVE 窗口仿真相应的时间长度,观察信号的时序。FPGA 设计仿真-在 ModelSim 中观察信号时序页面如图 3.31 所示。

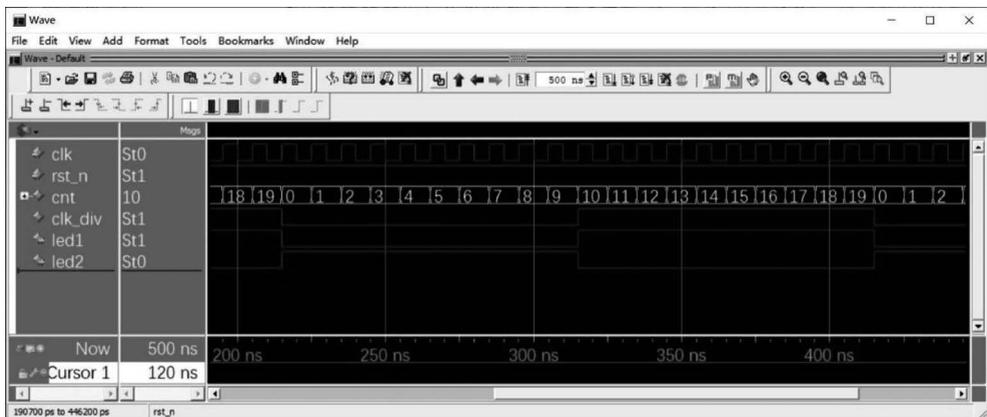


图 3.31 FPGA 设计仿真-在 ModelSim 中观察信号时序页面

至此,我们完成了使用 Diamond 软件进行 FPGA 开发的完整流程。

3.3.2 Intel Quartus Prime 开发 FPGA 实例(以 STEP MAX10 开发板为例)

1. 软件安装

Quartus Prime 是 Intel(原 Altera)公司的综合性 PLD/FPGA 开发软件,设计能力强大,接口直观易用,具有运行速度快、界面统一、功能集中、易学易用等特点。Quartus Prime 有 Lite、Standard 和 Pro 三个版本,不同版本可适用的器件和功能不同,其中 Lite 可用于 MAX10 系列 FPGA 的开发,而且是免费的。

下载安装包注意:选择软件版本和操作系统,同时勾选 Quartus Prime、ModelSim-Intel FPGA Edition 和 MAX 10 系列器件支持。大家可以去 Intel FPGA 官网下载,具体下载和安装过程我们不再赘述。

2. 新建工程

1) 创建工程

双击 Quartus 系列软件图标,启动软件,单击 File→New Project Wizard 选项或单击 Home 页面中的 New Project Wizard 图标。Quartus Prime Lite Edition 软件界面如图 3.32 所示。

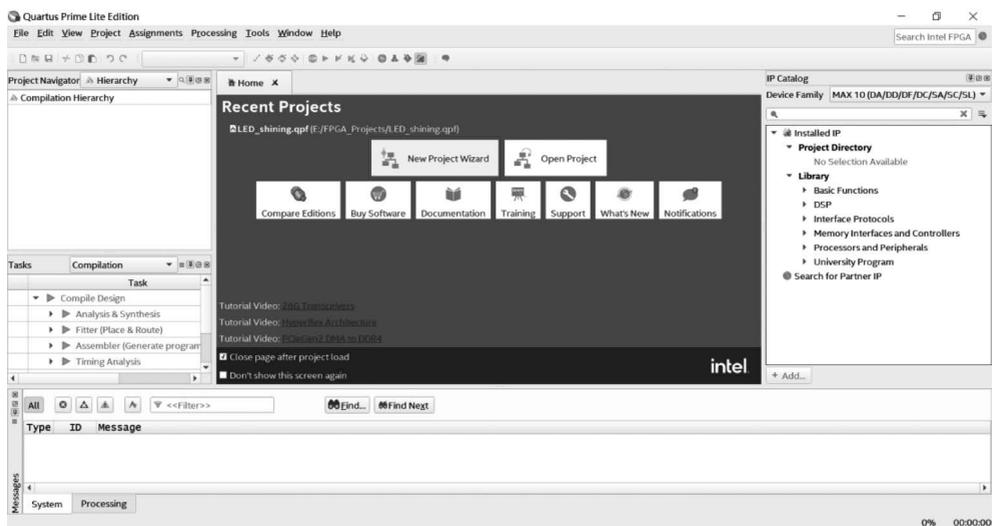


图 3.32 Quartus Prime Lite Edition 软件界面

2) 工程目录、名称填写

(注意:工程目录中不能有汉语、空格等字符)。Quartus Prime 新建工程-设置工程名称和目录页面如图 3.33 所示。

- 工程目录:选择新建工程的目录。
- 工程名称:填写工程名称。
- 顶层模块名称:设计文件中顶层模块名称要跟工程名称相同。
- 工程类型:选择 Empty project。

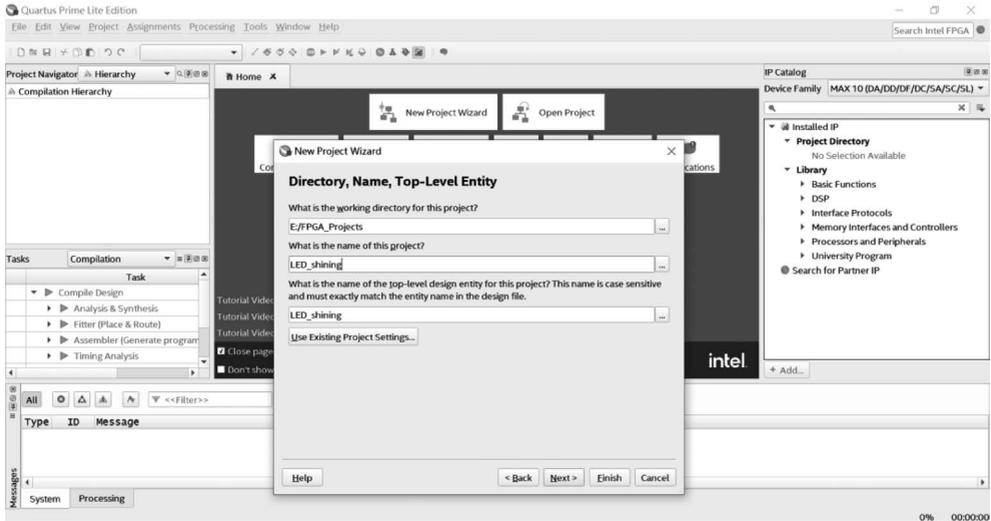


图 3.33 Quartus Prime 新建工程-设置工程名称和目录页面

3) 添加设计文件

如果已有设计文件,在当前页面选择并添加,也可以不添加,工程新建完成后再创建新文件。

4) 器件选择

根据开发平台使用的 FPGA 选择对应器件型号(10M02SCM153I7G/10M08SCM153C8G)页面如图 3.34 所示。

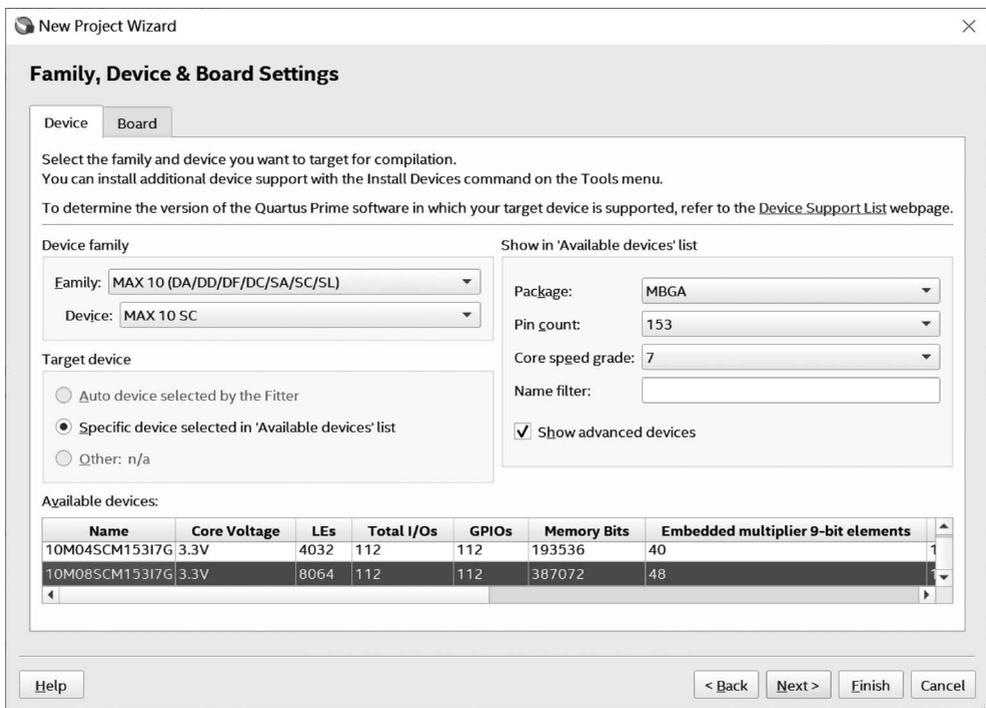


图 3.34 Quartus Prime 新建工程-选择器件型号页面

5) EDA 工具选择

选择第三方 EDA 工具,如果有需要可以选择第三方的综合或仿真工具(第三方工具需要另外安装并设置启动路径),这里我们选择使用 ModelSim 工具仿真。Quartus Prime 新建工程-选择综合和仿真工具页面如图 3.35 所示。

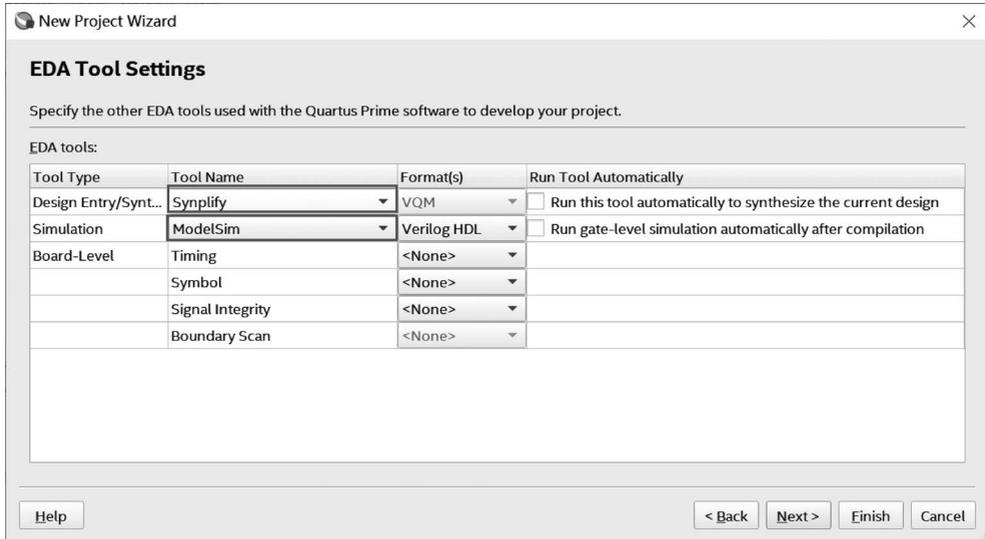


图 3.35 Quartus Prime 新建工程-选择综合和仿真工具页面

6) 工程信息确认

确认工程相应的设置,如需调整单击 Back 按钮返回修改,若确认设置,则单击 Finish 按钮。Quartus Prime 新建工程-工程信息确认页面如图 3.36 所示。

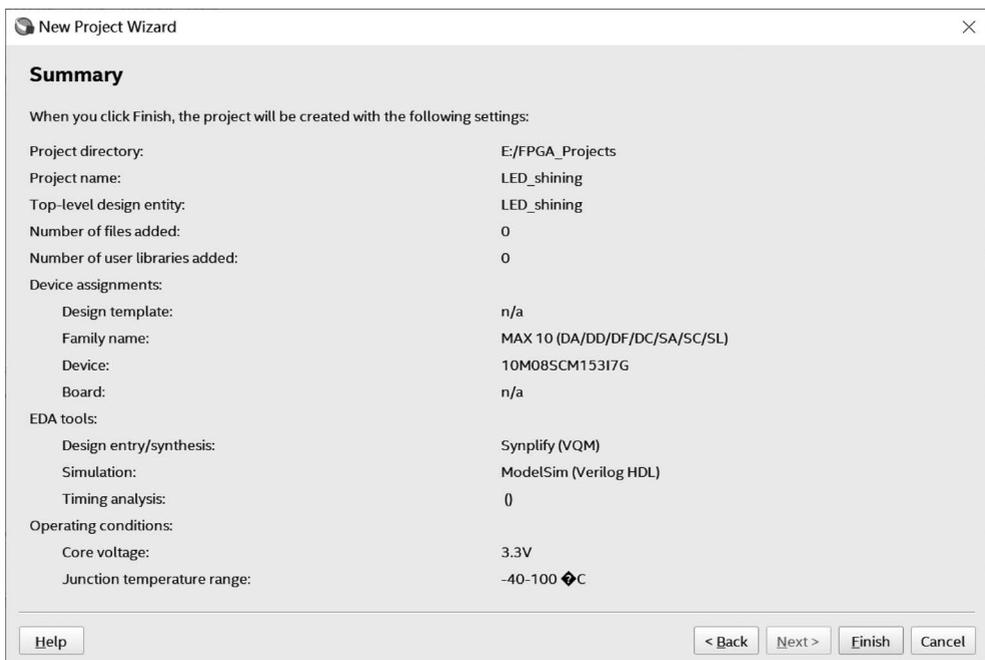


图 3.36 Quartus Prime 新建工程-工程信息确认页面

3. 添加设计文件

(1) 选择 File→New 选项,或单击工具栏中的 New 按钮,选择 Verilog HDL File 文件类型,单击 OK 按钮,Quartus 软件会新建并打开 Verilog 文件。

(2) 在新建的 Verilog 文件中进行 Verilog HDL 代码编写、保存,文件名为 LED_shining.v,程序源码如代码 3.3 所示。Verilog HDL 代码编写、保存页面如图 3.37 所示。

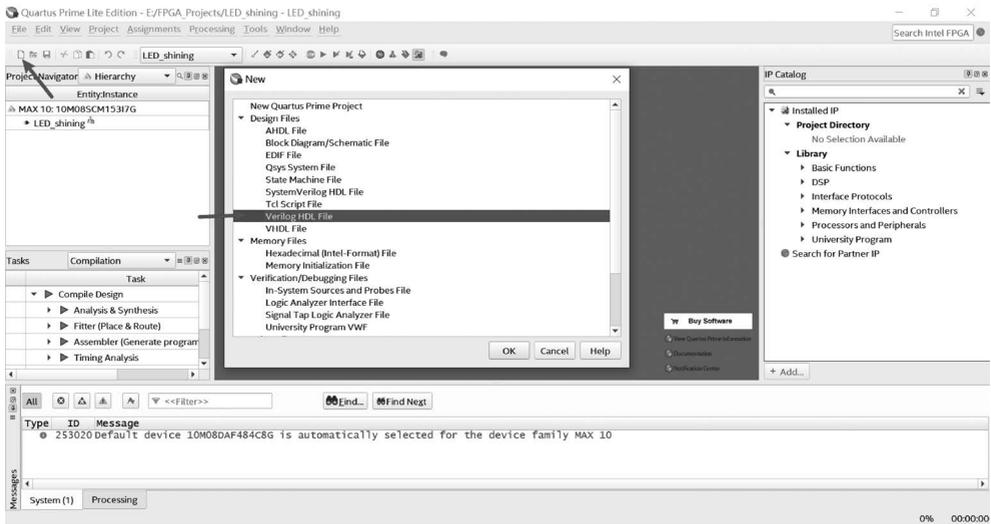


图 3.37 Verilog HDL 代码编写、保存页面

代码 3.3 Intel FPGA 开发案例 LED_shining.v 模块

```

module LED_shining
(
input clk_in,    //clk_in = 12mhz
input rst_n_in, //rst_n_in,低电平有效
output led1,    //led1 输出
output led2     //led2 输出
);

parameter CLK_DIV_PERIOD = 12_000_000;

reg clk_div = 0;

//wire led1,led2;
assign led1 = clk_div;
assign led2 = ~clk_div;

reg[24:0] cnt = 0;
always@(posedge clk_in or negedge rst_n_in) begin
    if(!rst_n_in) begin
        cnt <= 0;
        clk_div <= 0;
    end else begin
        if(cnt == (CLK_DIV_PERIOD - 1)) cnt <= 0;
        else cnt <= cnt + 1'b1;
        if(cnt < (CLK_DIV_PERIOD >> 1)) clk_div <= 0;
    end
end

```

```

else clk_div <= 1'b1;

end

end

endmodule

```

4. 分析综合

(1) 选择菜单栏中的 Processing→Start→Start Analysis & Synthesis 选项,或单击工具栏中的 Start Analysis & Synthesis 按钮,Quartus Prime 分析综合操作步骤页面如图 3.38 所示。

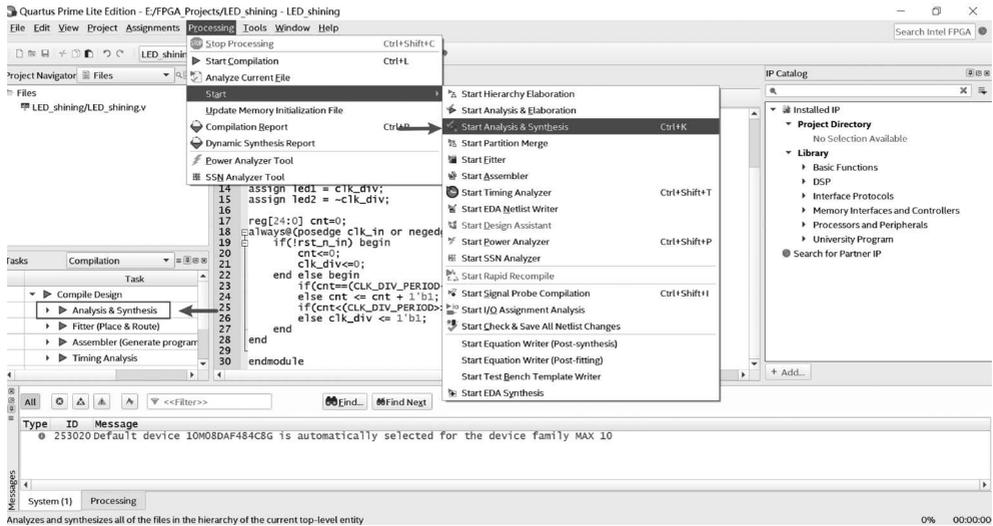


图 3.38 Quartus Prime 分析综合操作步骤页面

(2) Quartus 软件会完成分析综合,若设计没有问题,综合 Tasks 栏中 Analysis & Synthesis 会变成深色,同时左侧出现对勾。Quartus Prime 综合完成显示综合结果页面如图 3.39 所示。

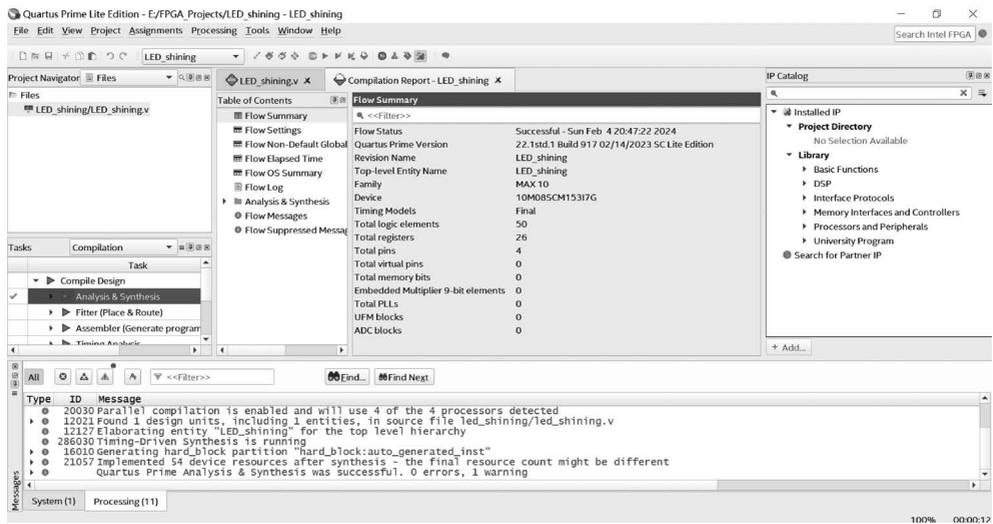


图 3.39 Quartus Prime 综合完成显示综合结果页面

(3) 综合完成后可以选择 Tools→Netlist Viewers→RTL Viewer 查看电路。Quartus Prime 查看综合后电路页面如图 3.40 所示。

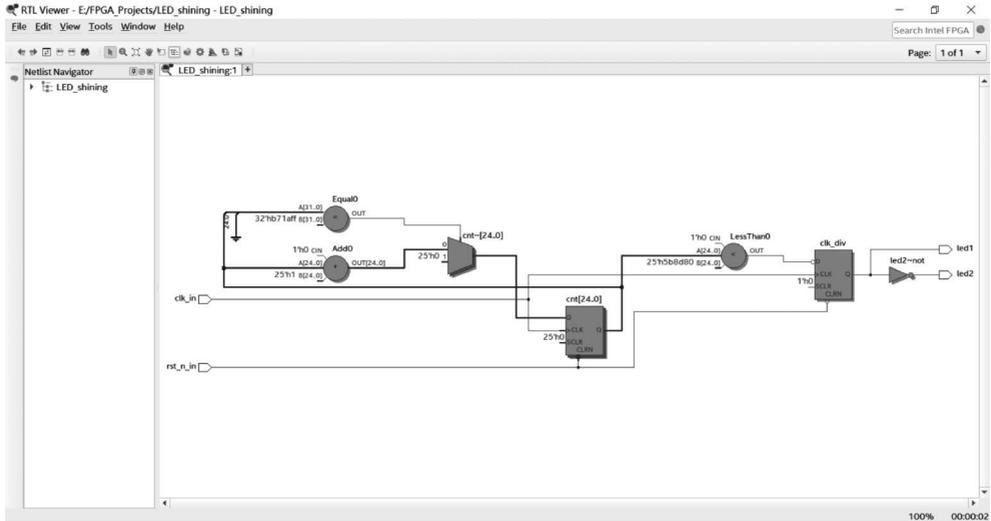


图 3.40 Quartus Prime 查看综合后电路页面

5. 引脚约束

(1) 选择 Assignments→Device 打开器件配置页面,然后单击页面中的 Device and Pin Options...选项打开器件和引脚选项页面。Quartus Prime 配置器件和引脚选项页面如图 3.41 所示。

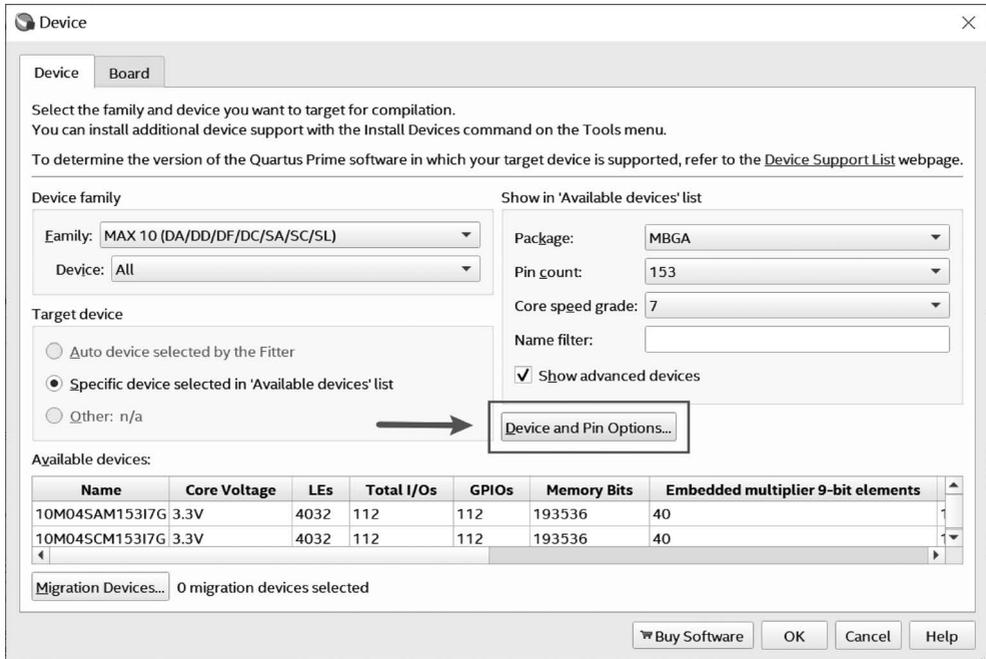


图 3.41 Quartus Prime 配置器件和引脚选项页面

(2) 在 Unused Pins 选项中配置 Reserve all unused pins 为 As input tri-stated 状态。Quartus Prime 配置引脚模式页面如图 3.42 所示。

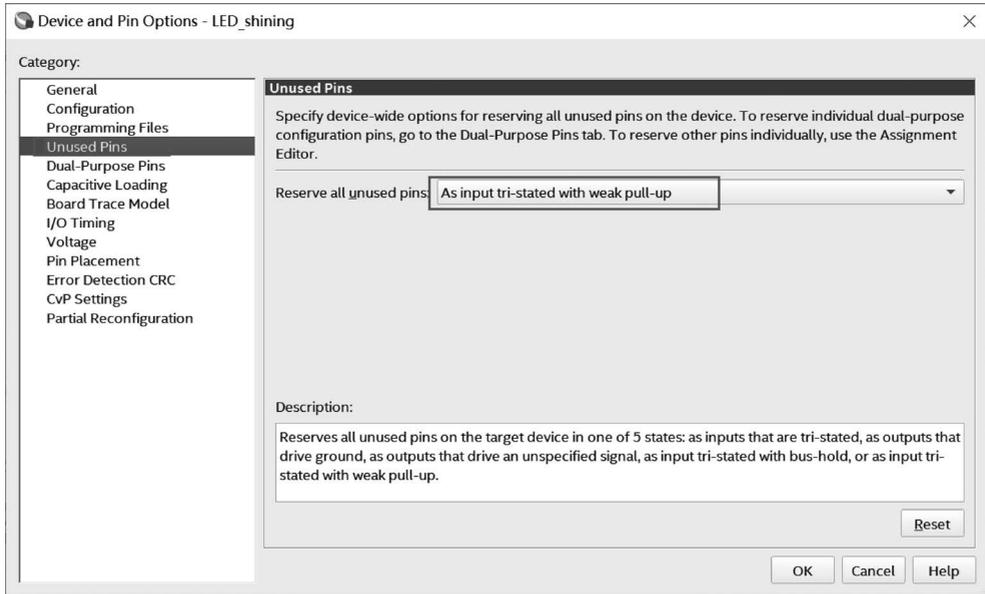


图 3.42 Quartus Prime 配置引脚模式页面

(3) 在 Voltage 选项中配置 Default I/O standard 为 3.3-V LVTTTL 状态。Quartus Prime 配置引脚电平页面如图 3.43 所示。

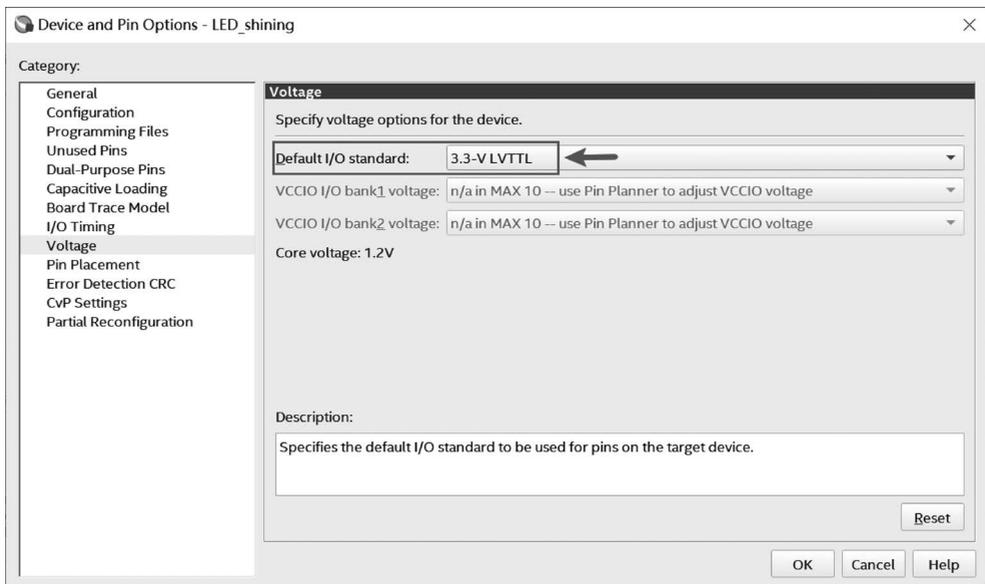


图 3.43 Quartus Prime 配置引脚电平页面

(4) 选择菜单栏中的 Assignments→Pin planner 选项,或单击工具栏中的 Pin planner 图标,进入引脚分配界面。在 Pin Planner 页面中将所有端口分配对应的 FPGA 引脚,Quartus Prime 引脚分配页面如图 3.44 所示,然后关闭(自动保存)。

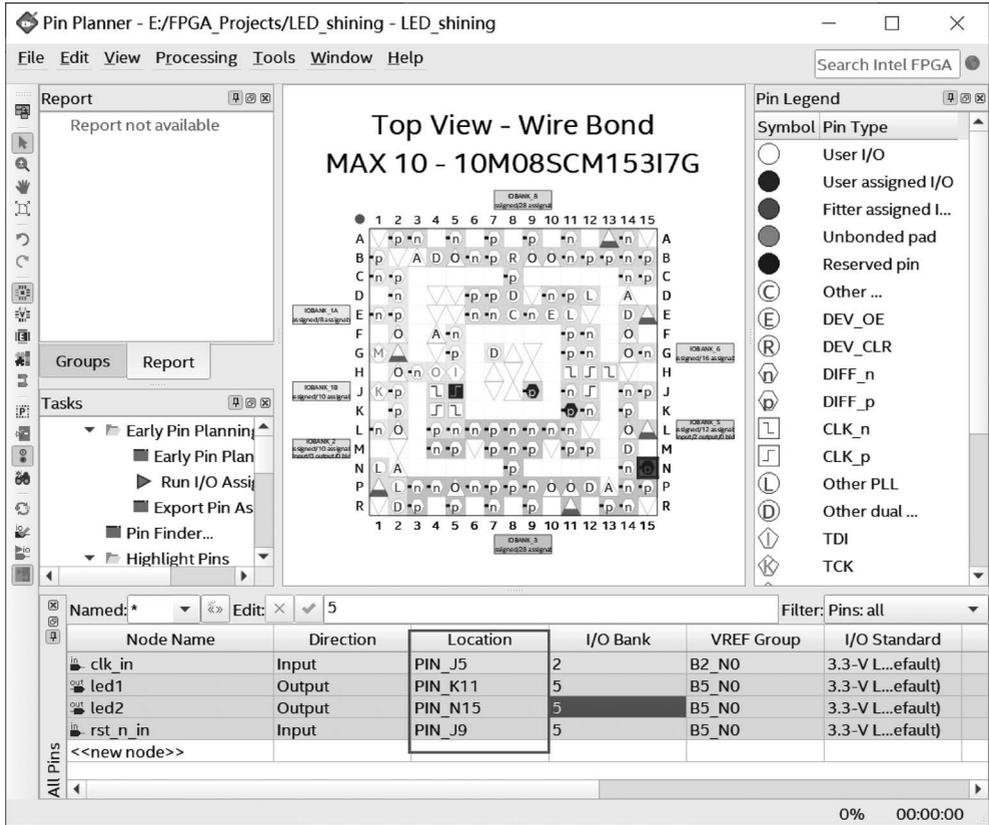


图 3.44 Quartus Prime 引脚分配页面

6. 编译(布局布线&生成配置文件)

选择菜单栏中的 Processing→Start Compilation 选项,或单击工具栏中的 Start Compilation 按钮,开始所有编译,等待 Tasks 列表中所有选项完成,Quartus Prime 编译完成页面如图 3.45 所示。

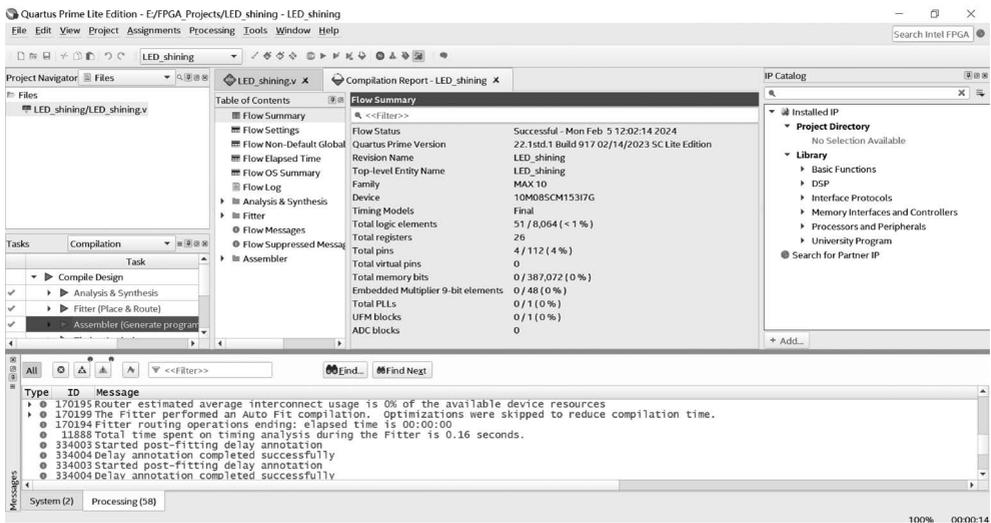


图 3.45 Quartus Prime 编译完成页面

7. 下载到 FPGA

(1) 使用 micro-usb 线将 STEP-MAX10 开发板连接至计算机 USB 接口, 选择菜单栏中的 Tools→Programmer 选项, 或单击工具栏中的 Programmer 按钮, 进入烧录界面。选择 Hardware Setup 选项, 选择硬件驱动为 USB-Blaster[USB-0]。Quartus Prime 编程下载及驱动设置页面如图 3.46 所示。

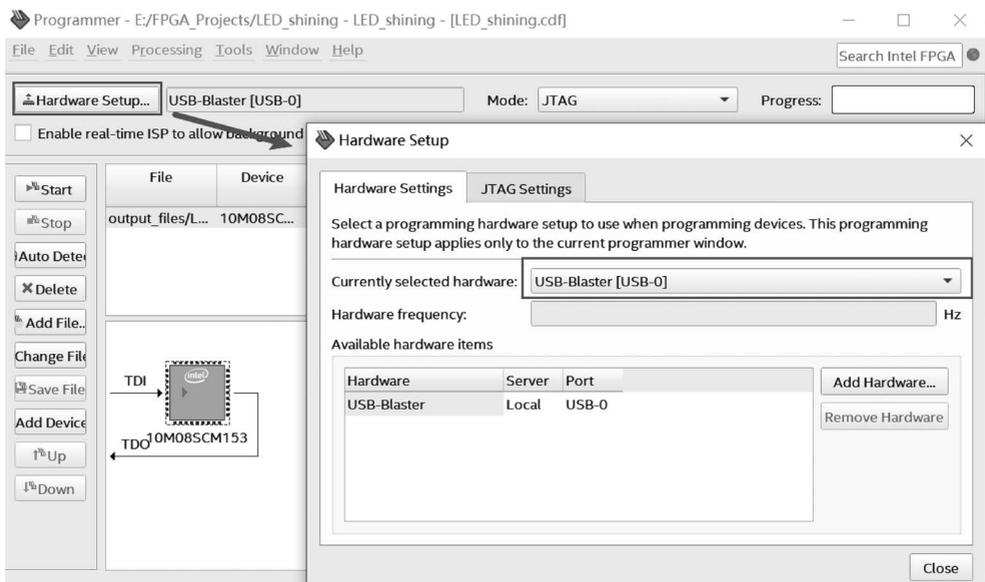


图 3.46 Quartus Prime 编程下载及驱动设置页面

(2) 确认驱动为 USB-Blaster[USB-0]后, 选择 Add File 添加工程输出文件中的 pof 格式文件, 勾选 Program 列和 Verify 列, 单击 Start 按钮进行 FPGA 加载。Quartus Prime 选择编程文件并启动下载页面如图 3.47 所示。

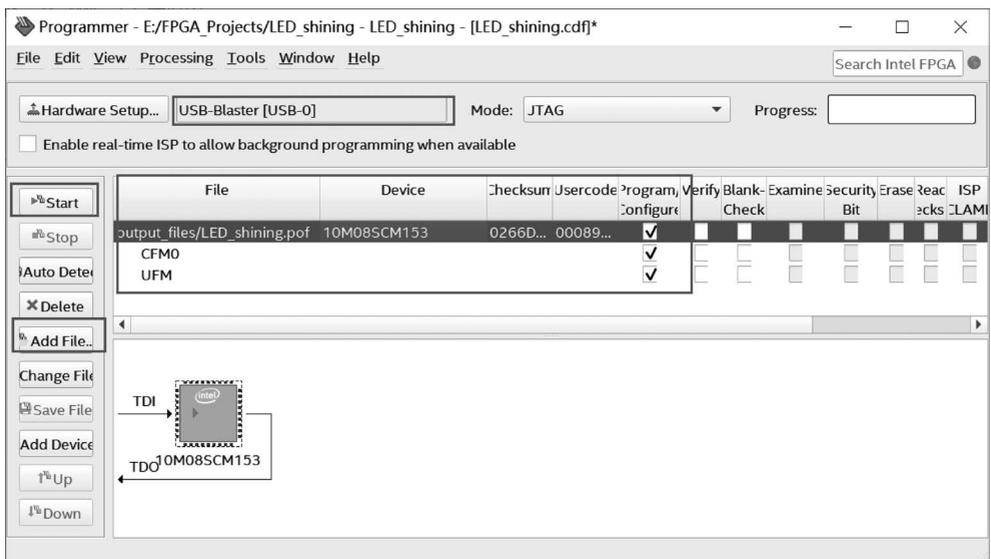


图 3.47 Quartus Prime 选择编程文件并启动下载页面

(3) FPGA 加载完成,界面中 Progress 状态显示 100%(Successful)。
至此,我们完成了 LED_shining 示例的开发流程,可以观察开发板现象。

8. 设计仿真

Quartus Prime 在安装时可选择仿真软件是否一起安装,最新的版本 Quartus Prime 23.1 可一起安装适用 Intel FPGA 的 QuestaSim,之前的版本可选择安装 ModelSim-Intel FPGA。当然我们也可以自己安装第三方仿真软件,除 QuestaSim 和 ModelSim 以外还可以安装 Active-HDL、Riviera-PRO 等仿真软件。所以仿真软件是独立存在的,Quartus Prime 只是预留了启动接口。

(1) 仿真文件:提前准备测试文件(Textbench)LEDshiningtb.v,测试文件源码见代码 3.2。

(2) 选择菜单栏中的 Assignments→Settings 选项,或单击工具栏中的 Settings 按钮,进入设置界面。选择菜单栏中的 Simulation 选项,单选 Compile test bench,单击 Test Benches 按钮。Quartus Prime 仿真设置界面如图 3.48 所示。

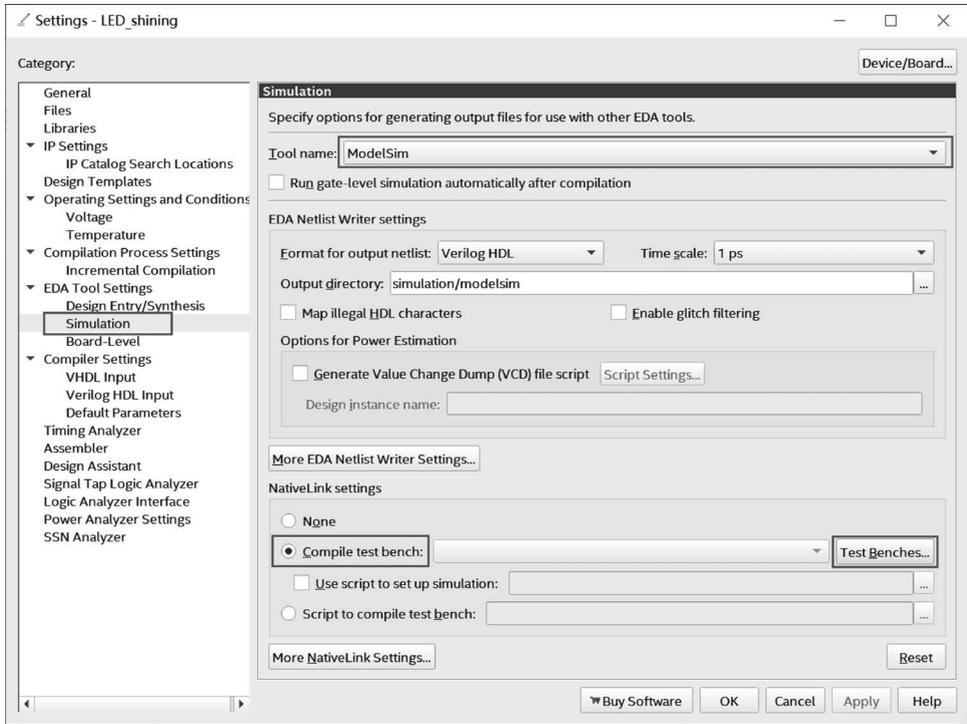


图 3.48 Quartus Prime 仿真设置界面

(3) 单击 Test Benches 按钮后,在弹出的对话框中单击 New 按钮,填写 Test bench name,按照目录添加测试文件,如图 3.49 标识顺序,最后单击 OK 按钮回到设计界面。Quartus Prime 仿真设置-添加测试文件界面如图 3.49 所示。

(4) 选择菜单栏中的 Tools→Run Simulation Tool→RTL Simulation 选项,或单击工具栏中的 RTL Simulation 按钮,Quartus 软件会自动启动 Modelsim 软件。Quartus Prime 仿真设置-启动 RTL 仿真界面如图 3.50 所示。

(5) ModelSim 软件启动后自动完成代码编译,ModelSim 仿真-启动界面如图 3.51 所示。

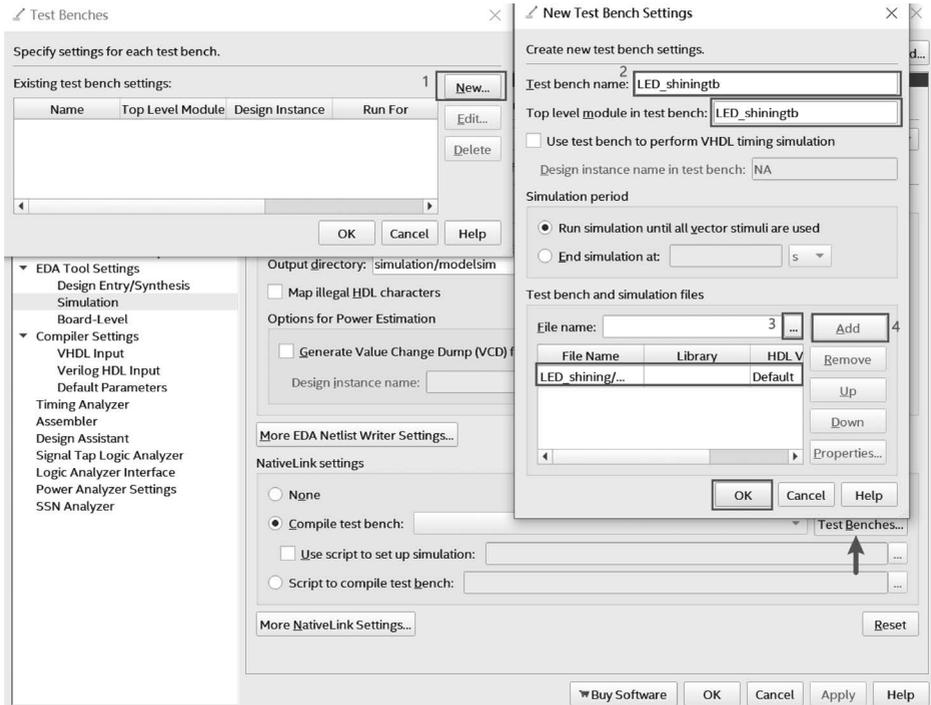


图 3.49 Quartus Prime 仿真设置-添加测试文件界面

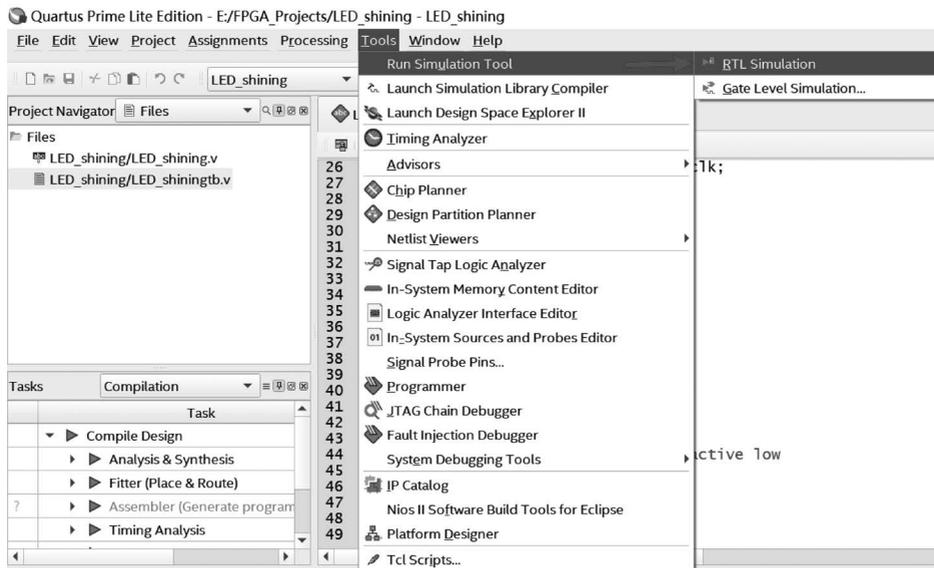


图 3.50 Quartus Prime 仿真设置-启动 RTL 仿真界面

(6) 选择需要观察波形的信号, 右击, 在弹出的快捷菜单中选择 Add Wave 选项, 这样就将对应信号添加至 Wave 窗口。ModelSim 仿真-添加观察信号界面如图 3.52 所示。

(7) ModelSim 仿真-设置仿真时间启动仿真界面如图 3.53 所示。在弹出的 Wave 窗口中单击工具栏中的 Restart 按钮, 在弹出的 Restart 窗口单击 OK 按钮, 复位仿真 Wave 窗口。修改工具栏中仿真时间, 单击工具栏中的 Run 按钮, 进行仿真, 仿真波形如图 3.53 所示, 仿真完成。

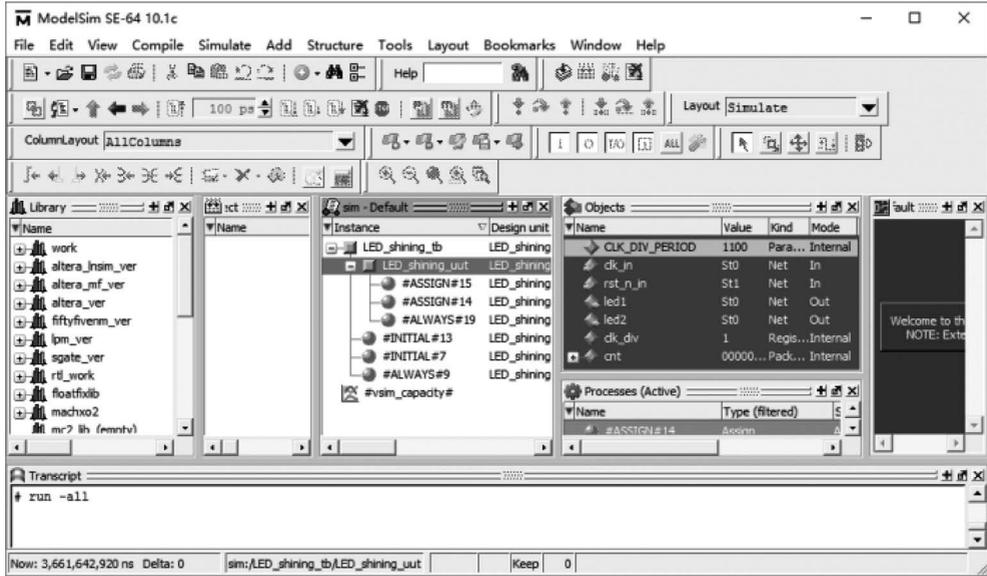


图 3.51 ModelSim 仿真-启动界面



图 3.52 ModelSim 仿真-添加观察信号界面

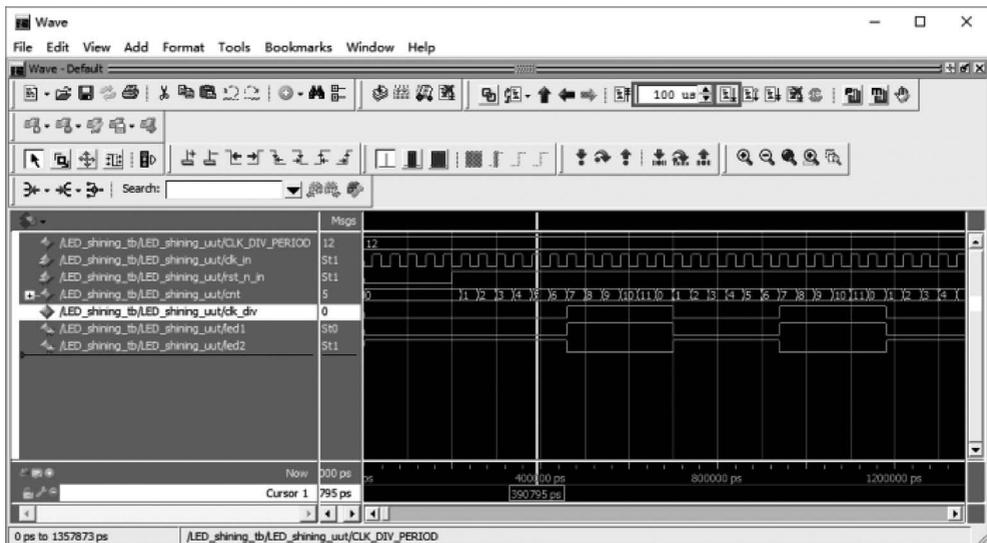


图 3.53 ModelSim 仿真-设置仿真时间启动仿真界面

3.3.3 小脚丫 FPGA(STEP FPGA)线上开发平台

FPGA 的集成开发软件(IDE)往往都非常庞大,所需安装空间少则几 GB 多则几十 GB,运行时对内存和 CPU 也有要求。针对以上问题,小脚丫 FPGA 开发了一款 FPGA 线上开发平台,将 IDE 安装至服务器端,用户在网页端完成所有的开发操作和信息交互,目前支持 STEP MXO2 和 STEP MAX10 FPGA 开发板的开发设计。

在小脚丫网站 www.stepfpga.com 注册账号后就可以体验使用线上设计工具,基于浏览器端的开发环境,无需下载 FPGA 设计工具到本地计算机,使用方便简单。STEP FPGA 线上开发平台主页如图 3.54 所示。



图 3.54 STEP FPGA 线上开发平台主页

1. 新建项目

单击新建项目会进入创建新项目界面,左侧是项目文件组成,可以在这里查看项目所需文件以及软件生成的日志等。在上方是 FPGA 设计的流程图,可以按顺序创建源文件、综合、分配引脚、产生编译文件、仿真以及下载工程和编译文件。

填写项目的一些基本信息,例如项目名称、设备、项目标签、描述等。STEP FPGA 线上开发平台-新建项目页面如图 3.55 所示。

2. 创建源文件

提交创建项目后进入编辑界面,单击界面上的“+”号可以新建源文件(也可以从其他项目复制文件)。STEP FPGA 线上开发平台-新建文件页面如图 3.56 所示。

3. 设置顶层文件

源文件创建完成后需要确认顶层文件,STEP FPGA 线上开发平台-设置顶层文件页面如图 3.57 所示,单击左侧栏文件名称后的箭头图标。设置完成后文件名称后的箭头图标消失,说明该文件目前是顶层文件。如果项目中包含多个源文件,也进行同样设置。

4. 逻辑综合

设置好顶层文件后单击“逻辑综合”,开始进行综合。综合完成后显示日志,如果有错误



图 3.55 STEP FPGA 线上开发平台-新建项目页面



图 3.56 STEP FPGA 线上开发平台-新建文件页面

的话会显示报错信息。STEP FPGA 线上开发平台-逻辑综合页面如图 3.58 所示。

5. 引脚分配

综合通过后进行引脚约束。这里主要对 FPGA 引脚的编号进行分配。STEP FPGA 线上开发平台-引脚分配页面如图 3.59 所示,小脚丫板卡可配置的引脚分为板上部分 (Internal) 和扩展部分 (External)。板子上的数码管、拨码开关、按键、LED 和三色灯属于板上部分。板子两侧扩展的 IO 引脚属于扩展部分。



图 3.57 STEP FPGA 线上开发平台-设置顶层文件页面

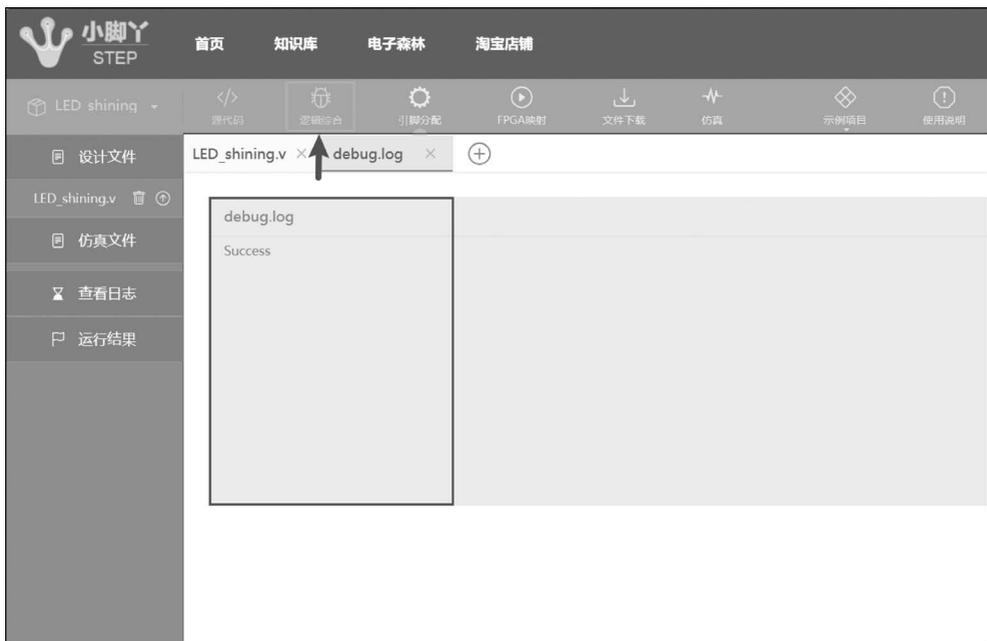


图 3.58 STEP FPGA 线上开发平台-逻辑综合页面

分配引脚过程很简单,单击想要分配的外设,会自动弹出所有的输入输出信号,选择确定就可以。所有信号分配完毕单击保存按键,如果分配错误可以选择重置。板上外设被分配使用后会以黄色来标记。

6. 映射生成流文件

分配完成后单击“FPGA 映射”按钮来产生 FPGA 的配置文件,完成后生成日志,如果

有错误会显示报错信息。STEP FPGA 线上开发平台-FPGA 映射页面如图 3.60 所示。

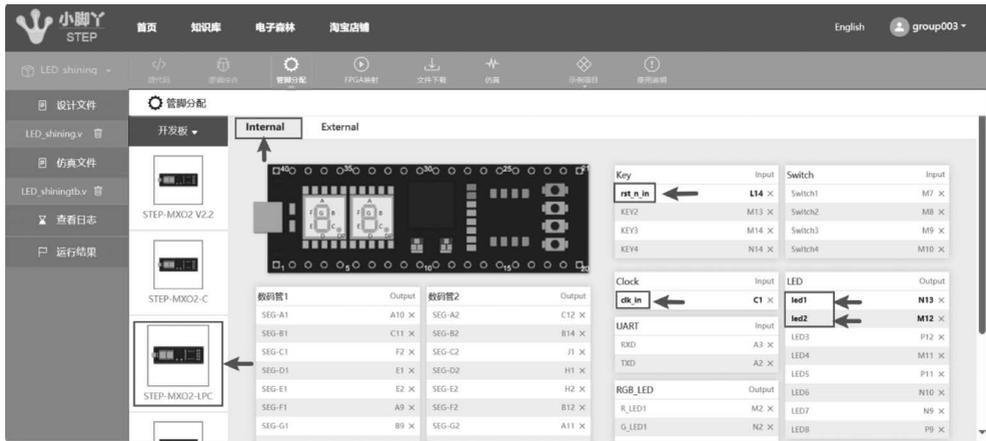


图 3.59 STEP FPGA 线上开发平台-引脚分配页面

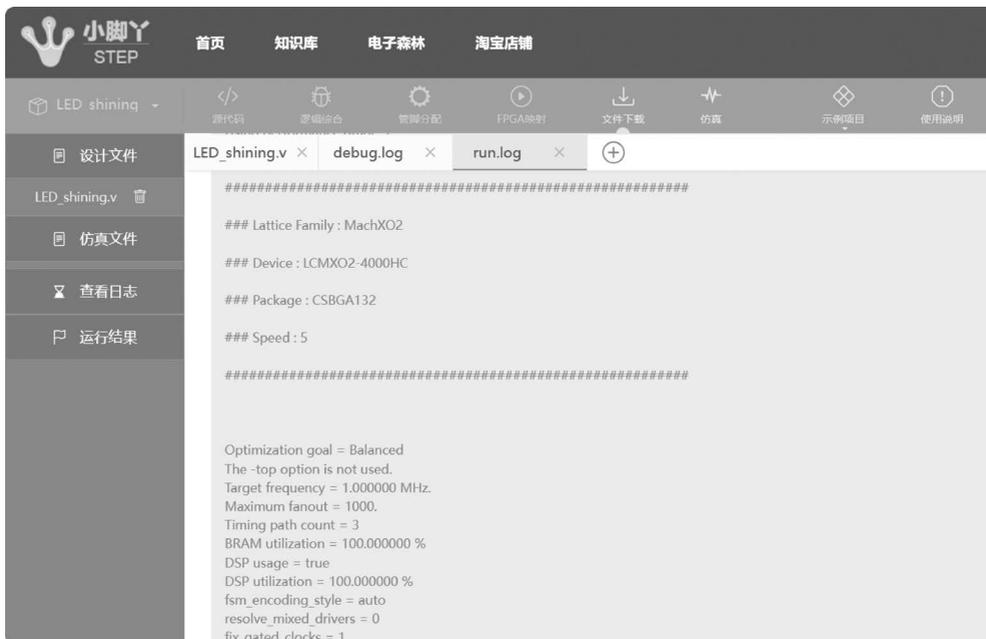


图 3.60 STEP FPGA 线上开发平台-FPGA 映射页面

7. 下载文件

可以在下载界面下载打包整个工程,也可以只下载需要的 FPGA 编程流文件(Lattice 小脚丫产生 jed 文件,Intel 小脚丫产生 pof 和 sof 文件)。STEP FPGA 线上开发平台-下载文件页面如图 3.61 所示。

8. 仿真工具使用

小脚丫线上设计工具也可以进行在线功能仿真操作,方便验证设计。

1) 新建仿真文件

单击新建文件,会弹出窗口,定义文件名称,然后记得勾选“设为仿真文件”选项。

STEP FPGA 线上开发平台-新建仿真文件页面如图 3.62 所示。

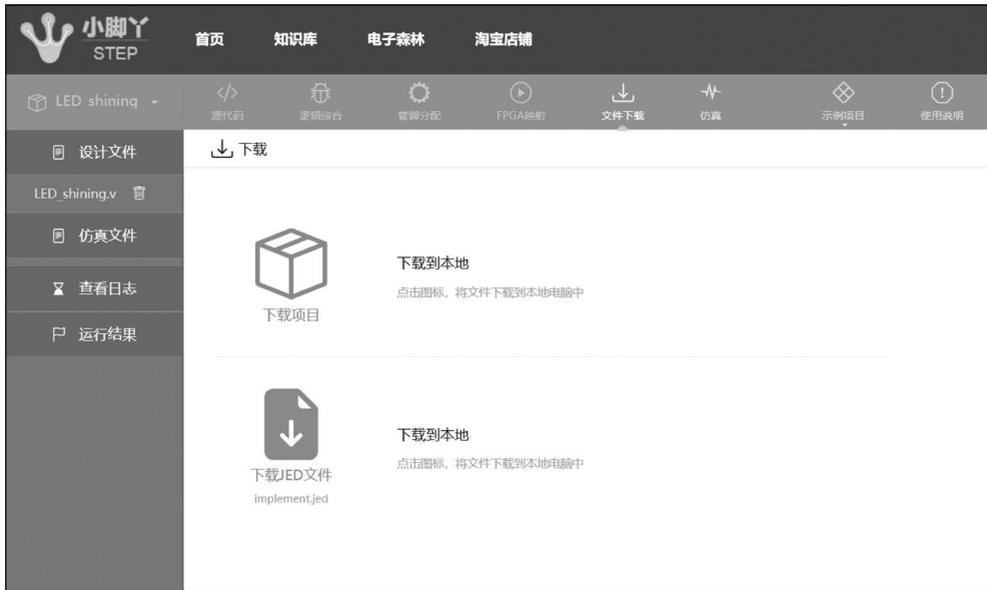


图 3.61 STEP FPGA 线上开发平台-下载文件页面



图 3.62 STEP FPGA 线上开发平台-新建仿真文件页面

2) 编辑仿真文件

编辑仿真文件然后保存。STEP FPGA 线上开发平台-编辑仿真文件页面如图 3.63 所示。

3) 仿真并查看波形

单击“开始仿真”，结束后可以看到输出波形，如果出错请查看出错信息。STEP FPGA 线上开发平台-查看仿真波形页面如图 3.64 所示。

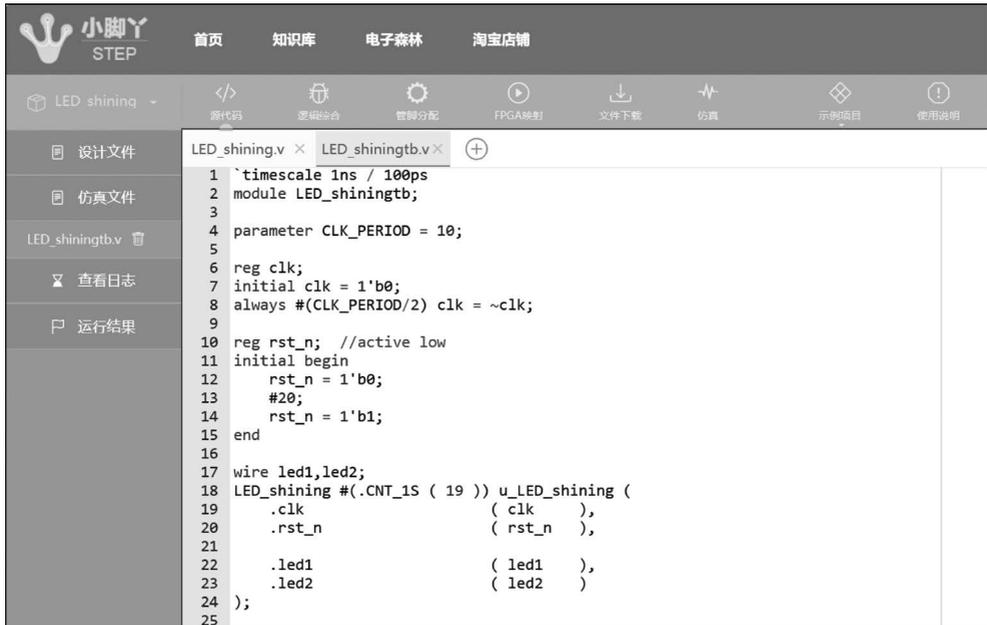


图 3.63 STEP FPGA 线上开发平台-编辑仿真文件页面

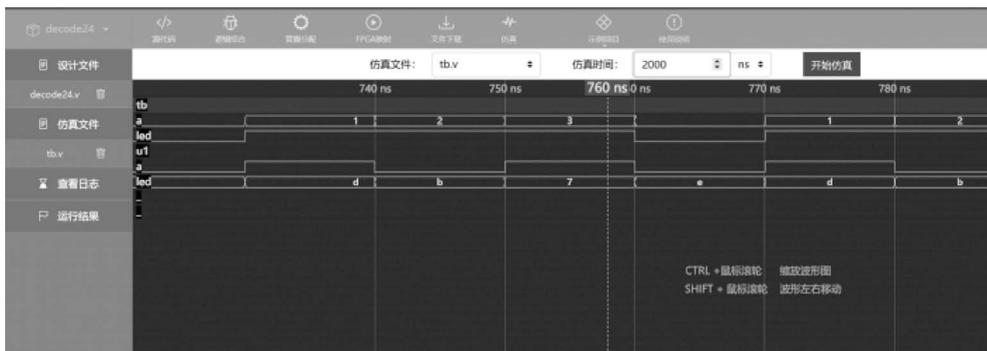


图 3.64 STEP FPGA 线上开发平台-查看仿真波形页面