

学习资源
见书中
学习说明

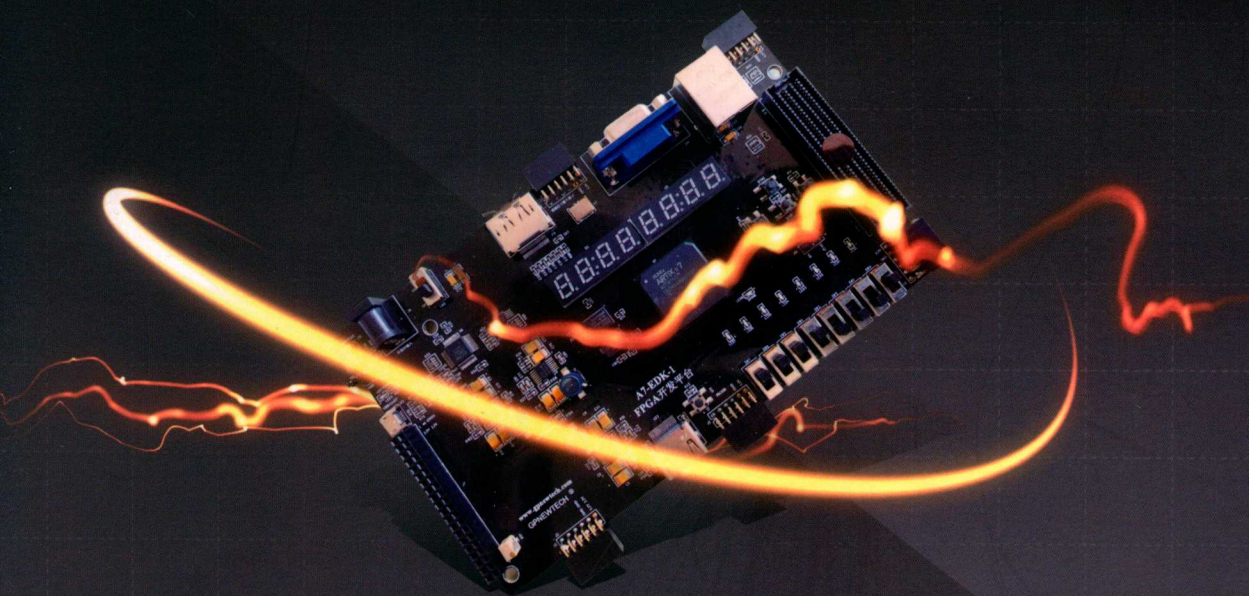
电子系统EDA新技术丛书

Xilinx Vivado

数字设计权威指南

从数字逻辑、Verilog HDL、嵌入式系统
到图像处理

◎ 何 宾 编著



- ★ 采用流行的SPICE仿真工具讲解数字逻辑和电路知识
- ★ 采用Vivado工具多角度分析Verilog HDL语法
- ★ 采用ARM Cortex-M1 IP核在FPGA上实现嵌入式系统
- ★ 包含视频图像采集和处理系统设计的知识



中国工信出版集团



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

电子系统 EDA 新技术丛书

Xilinx Vivado 数字设计 权威指南

从数字逻辑、Verilog HDL、嵌入式系统到图像处理

何 宾 编著

電子工業出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书以 Xilinx 公司的 Vivado 2018 集成开发环境作为复杂数字系统设计的平台,以基础的数字逻辑和数字电路知识为起点,以 Xilinx 7 系列可编程逻辑器件和 Verilog HDL 为载体,详细介绍了数字系统中基本逻辑单元 RTL 描述方法。在此基础上,实现了复杂数字系统设计、数模混合系统设计和基于 Cortex-M1 处理器软核的片上嵌入式系统设计。全书共 10 章,内容主要包括数字逻辑基础、数字逻辑电路、可编程逻辑器件原理、Vivado 集成开发环境设计流程、Verilog HDL 语言规范、基本数字逻辑单元 Verilog HDL 描述、复杂数字系统设计和实现、数模混合系统设计、片上嵌入式系统的构建和实现,以及图像采集、处理系统的构建和实现。

本书适合于需要系统掌握 Verilog HDL 和 Vivado 集成开发环境基本设计流程的初学者,同时也适合于需要掌握嵌入式系统软件和硬件设计方法的嵌入式开发工程师。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有,侵权必究。

图书在版编目(CIP)数据

Xilinx Vivado 数字设计权威指南:从数字逻辑、Verilog HDL、嵌入式系统到图像处理/何宾编著. —北京:电子工业出版社,2019.6
(电子系统 EDA 新技术丛书)
ISBN 978-7-121-36495-2

I. ①X… II. ①何… III. ①现场可编程门阵列-系统设计-指南 IV. ①TP331.202.1-62

中国版本图书馆 CIP 数据核字(2019)第 090159 号

策划编辑:张 迪

责任编辑:张 迪(zhangdi@phei.com.cn)

印 刷:三河市良远印务有限公司

装 订:三河市良远印务有限公司

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本:787×1092 1/16 印张:38.5 字数:986 千字

版 次:2019 年 6 月第 1 版

印 次:2019 年 6 月第 1 次印刷

定 价:149.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010)88254888,88258888。

质量投诉请发邮件至 zlts@phei.com.cn,盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式:(010)88254469; zhangdi@phei.com.cn。

学习说明

Study Shows

本书视频课堂地址

书中提及的完整的公共免费高清视频可到爱课程网观看学习。

(1) <http://www.icourses.cn/home/>，在该网站中，通过输入“EDA 原理及应用”搜索视频资源。

(2) <http://www.edawiki.com>，网络课堂栏目。

本书教学课件（PPT）及工程文件下载地址

北京汇众新特科技有限公司维基页面。

<http://www.edawiki.com>

注意：所有教学课件及工程文件仅限于购买本书读者学习使用，不得以任何方式传播！

本书作者联络方式

何宾的网站：<http://www.gpnewtech.com>

何宾的电子邮件：hb@gpnewtech.com

购买本书配套的 **A7-EDP-1** 开发板及配件由北京汇众新特科技有限公司负责
市场及服务支持热线：010-83139076 010-83139176

更多资讯可登录微信公众号

gpnewtech

前 言

现场可编程门阵列 (Field Programmable Gate Array, FPGA) 越来越多地被应用在新技术中, 如物联网、云计算和人工智能等。在这些应用中, FPGA 主要用来对数据进行加速处理。为了应对这些应用, 降低软件工程师应用 FPGA 的难度, Xilinx 公司不断推出新的设计工具, 如高级综合工具 (High Level Synthesis, HLS), 以降低使用 FPGA 实现复杂应用的难度。然而, 很多软件工程师仍然觉得 FPGA 入门较难, 这是因为他们普遍认为 Verilog HDL 比较抽象难懂, 且 FPGA 的内部结构过于复杂。其实, 最根本的原因是软件设计工程师常常以传统软件的思维来看待 FPGA, 他们普遍认为 FPGA 是硬件, 与软件没有太多的交集, 在 FPGA 中实现传统上由软件实现的算法模型难度较大。根据作者长期教学和科研的经验, 数字逻辑和数字电路基础知识是他们入门 FPGA 的“绊脚石”和“拦路虎”。为了帮助广大读者能真正进入 FPGA 设计领域, 尤其是 FPGA 的初学者和那些从事传统软件开发工作的工程师, 作者编写了本书。本书是作者多年从事 FPGA 教学和科研工作的体会与总结, 期望对广大初学者系统掌握 FPGA 的设计方法提供很好的帮助。

数字逻辑和数字电路的基本理论知识是学习 FPGA 的基础, 不管 FPGA 技术今后如何发展, 始终离不开数字逻辑的基本理论知识。系统深入地掌握以上知识是读者进入 FPGA 设计世界的基石, 特别重要。所以, 在编写本书时特意增加了数字逻辑基础和数字电路两章内容。作者在编写这两章内容时参考了国外大量的设计资料, 希望通过这两章内容的讲解来帮助广大读者准确地把握数字世界的本质, 并且通过 Multisim 内集成的 SPICE 仿真工具对这些知识点进行了直观演示和验证。根据作者多年的教学经验, 认为这些知识难点是入门 FPGA 的最大障碍, 因此通过 SPICE 仿真工具给出的分析结果帮助读者扫清这些学习障碍。

Verilog HDL 是本书最重要的内容之一, 用于对复杂数字系统 (尤其是 FPGA) 进行行为级和寄存器传输级建模。本书严格按照 IEEE Std 1364-2005 规范介绍 Verilog HDL 的词法和句法。在介绍这部分内容时, 将 Verilog HDL 与复杂数字系统 (尤其是 FPGA) 模型之间的对应关系进行系统讲解, 使读者理解 Verilog HDL 的词法和句法在复杂数字系统行为级和寄存器传输级描述中的使用方法。

本书的一大特色是将 Verilog HDL 和 Vivado 集成开发环境进行系统化深度融合, 从不同角度深度解读 Verilog HDL 语言的实现本质。针对 Verilog HDL 中的一些语法难点, 书中通过 Vivado 集成开发环境提供的功能进行演示和说明。在此要特别指出, Vivado 集成开发工具是学习 Verilog HDL 最好的助手, 这是因为在初学者遇到 Verilog HDL 中不理解的地方时, 可以很容易地通过 Vivado 集成开发工具给出的电路结构和仿真结果进行直观的说明。为了帮助读者提高灵活运用 Verilog HDL 构建复杂数字系统模型的能力, 书中给出了大量的基本逻辑单元的寄存器传输级描述, 以及一个复杂数字系统设计实例和数模混合系统设计实例。

本书的另一大特色是引入 ARM 为 Xilinx 现场可编程门阵列最新定制的 Cortex-M1 处理器软核。通过使用 Verilog HDL 构建嵌入式硬件平台和使用 C 语言编写硬件驱动, 以及实现

软件应用，在现场可编程门阵列内实现了真正意义上的片上可编程嵌入式系统。这里的可编程是指使用 Verilog HDL 定制嵌入式系统的硬件，然后使用 C 语言为这个定制的嵌入式硬件平台编写软件驱动和应用，这个设计过程充分体现了在 FPGA 上构建嵌入式系统的灵活性和高效性，同时对广大读者系统学习 ARM 嵌入式的硬件和软件知识提供了很好的帮助。通过对片上嵌入式系统设计流程的详细解读，读者将进一步掌握 C 语言串行执行和 Verilog HDL 并行处理的本质特点。通过在嵌入式系统设计中合理划分软件和硬件的边界，最终实现低成本、高性能的片上嵌入式系统设计。当然，对片上嵌入式系统设计过程的系统讲解也是为了帮助读者理解软件处理的灵活性和硬件处理的高效性，进而使读者进一步理解在新技术中越来越多地使用硬件（FPGA）来实现更复杂的算法的原因。

全书共 10 章，内容主要包括数字逻辑基础、数字逻辑电路、可编程逻辑器件原理、Vivado 集成开发环境设计流程、Verilog HDL 语言规范、基本数字逻辑单元 Verilog HDL 描述、复杂数字系统设计和实现、数模混合系统设计、片上嵌入式系统的构建和实现，以及图像采集、处理系统的构建和实现。为了便于读者自学，本书提供了所有设计实例的完整设计文件和公开教学视频，这些资源可以通过书中学习说明给出的链接地址获取。

在本书编写过程中参考了许多著名学者和专家的研究成果，同时也参考了 Xilinx 公司的技术文档和手册。在编写本书的过程中，Xilinx 多位技术专家解答了作者在设计中所遇到的各种问题，ARM 大学计划提供了 Cortex-M1 及其参考设计在此向他们表示衷心的感谢。在本书编写的过程中，作者的学生孟繁阳负责设计和验证第 1 章和第 2 章的实例，学生周杨参与本书第 10 章实例的设计和文字编写。除此之外，参加本书编写的还有何长有，在此表示感谢。在本书出版的过程中，也得到了电子工业出版社编辑的帮助和指导，在此也表示深深的谢意。

由于编著者水平有限，编写时间仓促，书中难免有疏漏之处，敬请读者批评指正。

作者

2019 年 5 月于北京

目 录

第 1 章 数字逻辑基础	1
1.1 数字逻辑的发展史	1
1.2 SPICE 仿真工具基础	4
1.2.1 SPICE 的分析功能	4
1.2.2 SPICE 的分析流程	7
1.3 开关系统	7
1.3.1 0 和 1 的概念	8
1.3.2 开关系统的优势	9
1.3.3 晶体管作为开关	11
1.3.4 半导体物理器件	12
1.3.5 半导体逻辑电路	14
1.3.6 逻辑电路符号	18
1.4 半导体数字集成电路	20
1.4.1 集成电路发展	20
1.4.2 集成电路构成	21
1.4.3 集成电路版图	22
1.5 基本逻辑门及特性	23
1.5.1 基本逻辑门	23
1.5.2 基本逻辑门集成电路	29
1.5.3 逻辑门电路的传输特性	31
1.5.4 不同逻辑门的连接	39
1.6 逻辑代数理论	41
1.6.1 逻辑代数中的运算关系	41
1.6.2 逻辑函数表达式	43
1.7 逻辑表达式的化简	45
1.7.1 使用运算律化简逻辑表达式	47
1.7.2 使用卡诺图化简逻辑表达式	50
1.7.3 不完全指定逻辑功能的化简	56
1.7.4 输入变量的卡诺图表示	58
1.8 毛刺产生及消除	62
1.9 数字码制表示和转换	65
1.9.1 数字码制表示	65
1.9.2 数字码制转换	67

第 2 章 数字逻辑电路	70
2.1 组合逻辑电路	70
2.1.1 编码器	71
2.1.2 译码器	72
2.1.3 码转换器	76
2.1.4 多路选择器	79
2.1.5 数字比较器	80
2.1.6 加法器	83
2.1.7 减法器	86
2.1.8 加法器/减法器	91
2.1.9 乘法器	93
2.2 时序逻辑电路	94
2.2.1 时序逻辑电路类型	95
2.2.2 时序逻辑电路特点	95
2.2.3 基本 SR 锁存器	97
2.2.4 同步 SR 锁存器	98
2.2.5 D 锁存器	99
2.2.6 D 触发器	100
2.2.7 其他触发器	103
2.2.8 普通寄存器	105
2.2.9 移位寄存器	106
2.3 存储器	107
2.3.1 存储器的分类	107
2.3.2 存储器工作原理	108
2.3.3 易失性存储器	109
2.3.4 非易失性存储器	110
2.4 有限状态机	111
2.4.1 有限状态机的原理	111
2.4.2 状态图表示及实现	112
2.4.3 三位计数器的设计与实现	114
第 3 章 可编程逻辑器件原理	119
3.1 可编程逻辑器件发展历史	119
3.2 可编程逻辑器件工艺	120
3.3 可编程逻辑器件结构	124
3.3.1 PROM 结构	124
3.3.2 PAL 结构	124
3.3.3 PLA 结构	125
3.4 复杂可编程逻辑器件结构	125
3.4.1 功能块	126

3.4.2	宏单元	127
3.4.3	快速连接开关阵列	128
3.4.4	输入/输出块	128
3.5	现场可编程门阵列结构	129
3.5.1	查找表结构原理	130
3.5.2	可配置的逻辑块	132
3.5.3	时钟管理资源	135
3.5.4	块存储器资源	139
3.5.5	互联资源	141
3.5.6	DSP 切片	143
3.5.7	输入/输出块	143
3.5.8	XADC 模块	145
3.6	Xilinx 7 系列 FPGA 产品	147
第 4 章	Vivado 集成开发环境设计流程	151
4.1	Vivado 集成开发环境	151
4.2	创建新的设计工程	153
4.3	创建并添加一个新的设计文件	156
4.4	详细描述	159
4.4.1	详细描述的原理	159
4.4.2	详细描述的实现过程	160
4.5	设计行为级仿真	161
4.6	设计综合和分析	163
4.6.1	综合过程的关键问题	163
4.6.2	执行设计综合	164
4.6.3	查看综合报告	166
4.7	约束文件对话框	166
4.7.1	约束文件	167
4.7.2	I/O 规划器的功能	168
4.7.3	实现约束	169
4.8	设计实现和分析	173
4.8.1	设计实现原理	173
4.8.2	设计实现及分析	173
4.9	设计时序仿真	175
4.10	生成并下载比特流文件	176
4.10.1	FPGA 配置原理	176
4.10.2	生成比特流文件	178
4.10.3	下载比特流文件	178
4.11	生成并烧写 PROM 文件	180
第 5 章	Verilog HDL 规范	184
5.1	Verilog HDL 发展	184

5.2	Verilog HDL 程序结构	185
5.2.1	模块声明	186
5.2.2	模块端口定义	186
5.2.3	逻辑功能定义	187
5.3	Verilog HDL 描述方式	189
5.3.1	行为级描述	189
5.3.2	数据流描述	190
5.3.3	结构级描述	192
5.3.4	开关级描述	194
5.4	Verilog HDL 要素	195
5.4.1	注释	195
5.4.2	间隔符	196
5.4.3	标识符	196
5.4.4	关键字	197
5.4.5	系统任务和函数	197
5.4.6	编译器指令	197
5.4.7	运算符	197
5.4.8	数字	197
5.4.9	字符串	200
5.4.10	属性	202
5.5	Verilog HDL 数据类型	205
5.5.1	值的集合	205
5.5.2	网络和变量	205
5.5.3	向量	206
5.5.4	强度	206
5.5.5	隐含声明	207
5.5.6	网络类型	207
5.5.7	寄存器类型	213
5.5.8	整型、实数型、时间型和实时时间	214
5.5.9	数组	215
5.5.10	参数	216
5.5.11	Verilog HDL 名字空间	218
5.6	Verilog HDL 表达式	219
5.6.1	操作符	219
5.6.2	操作数	229
5.6.3	延迟表达式	231
5.6.4	表达式的位宽	231
5.6.5	有符号表达式	233
5.6.6	分配和截断	234
5.7	Verilog HDL 分配	234

5.7.1	连续分配	235
5.7.2	过程分配	237
5.8	Verilog HDL 门级和开关级描述	238
5.8.1	门和开关声明	238
5.8.2	逻辑门	242
5.8.3	输出门	243
5.8.4	三态门	244
5.8.5	MOS 开关	246
5.8.6	双向传输开关	248
5.8.7	CMOS 开关	248
5.8.8	pull 门	249
5.9	Verilog HDL 用户自定义原语	249
5.9.1	UDP 定义	249
5.9.2	组合电路 UDP	251
5.9.3	电平触发的时序 UDP	252
5.9.4	边沿触发的时序 UDP	253
5.9.5	边沿和电平触发的混合行为	254
5.10	Verilog HDL 行为描述语句	255
5.10.1	过程语句	255
5.10.2	过程连续分配	261
5.10.3	条件语句	263
5.10.4	case 语句	265
5.10.5	循环语句	268
5.10.6	过程时序控制	270
5.10.7	语句块	274
5.10.8	结构化的过程	276
5.11	Verilog HDL 任务和函数	277
5.11.1	任务和函数的区别	277
5.11.2	定义和使能任务	278
5.11.3	禁止命名的块和任务	281
5.11.4	声明和调用函数	282
5.12	Verilog HDL 层次化结构	285
5.12.1	模块和模块例化	285
5.12.2	覆盖模块参数值	285
5.12.3	端口	290
5.12.4	生成结构	295
5.12.5	层次化的名字	304
5.12.6	向上名字引用	305
5.12.7	范围规则	306
5.13	Verilog HDL 设计配置	307

5.13.1	配置格式	308
5.13.2	库	309
5.13.3	配置例子	310
5.14	Verilog HDL 指定块	311
5.14.1	模块路径声明	312
5.14.2	为路径分配延迟	317
5.14.3	混合模块路径延迟和分布式延迟	320
5.14.4	驱动布线逻辑	320
5.14.5	脉冲过滤行为的控制	321
5.15	Verilog HDL 时序检查	326
5.15.1	使用一个稳定窗口检查时序	327
5.15.2	时钟和控制信号的时序检查	331
5.15.3	边沿控制标识符	338
5.15.4	提示符：用户定义对时序冲突的响应	338
5.15.5	使能带有条件的时序检查	342
5.15.6	时序检查中的矢量信号	343
5.15.7	负时序检查	344
5.16	Verilog HDL SDF 逆向注解	344
5.16.1	SDF 注解器	345
5.16.2	映射 SDF 结构到 Verilog	345
5.16.3	多个注解	350
5.16.4	多个 SDF 文件	351
5.16.5	脉冲限制注解	351
5.16.6	SDF 到 Verilog 延迟值映射	351
5.17	Verilog HDL 系统任务和函数	352
5.17.1	显示任务	352
5.17.2	文件 I/O 任务和函数	358
5.17.3	时间标度任务	367
5.17.4	仿真控制任务	370
5.17.5	随机分析任务	370
5.17.6	仿真时间函数	372
5.17.7	转换函数	373
5.17.8	概率分布函数	374
5.17.9	命令行输入	375
5.17.10	数学函数	378
5.18	Verilog HDL 的 VCD 文件	379
5.18.1	Vivado 创建四态 VCD 文件	379
5.18.2	Verilog 源创建四态 VCD 文件	381
5.18.3	四态 VCD 文件格式	384
5.19	Verilog HDL 编译器指令	388

5.19.1	`celldefine 和`endcelldefine	388
5.19.2	`default_nettype	388
5.19.3	`define 和`undef	388
5.19.4	`ifdef、`else、`elsif、`endif、`ifndef	390
5.19.5	`include	392
5.19.6	`resetall	393
5.19.7	`line	393
5.19.8	`timescale	393
5.19.9	`unconnected_drive 和`nounconnected_drive	394
5.19.10	`pragma	395
5.19.11	`begin_keywords 和`end_keyword	395
5.20	Verilog HDL (IEEE 1364—2005) 关键字列表	395
第 6 章	基本数字逻辑单元 Verilog HDL 描述	397
6.1	组合逻辑电路 Verilog HDL 描述	397
6.1.1	逻辑门 Verilog HDL 描述	397
6.1.2	编码器 Verilog HDL 描述	398
6.1.3	译码器 Verilog HDL 描述	398
6.1.4	多路选择器 Verilog HDL 描述	401
6.1.5	数字比较器 Verilog HDL 描述	403
6.1.6	总线缓冲器 Verilog HDL 描述	403
6.2	数据运算操作 Verilog HDL 描述	405
6.2.1	加法操作 Verilog HDL 描述	405
6.2.2	减法操作 Verilog HDL 描述	405
6.2.3	乘法操作 Verilog HDL 描述	406
6.2.4	除法操作 Verilog HDL 描述	406
6.2.5	算术逻辑单元 Verilog HDL 描述	407
6.3	时序逻辑电路 Verilog HDL 描述	409
6.3.1	触发器和锁存器 Verilog HDL 描述	409
6.3.2	计数器 Verilog HDL 描述	411
6.3.3	移位寄存器 Verilog HDL 描述	414
6.3.4	脉冲宽度调制 Verilog HDL 描述	419
6.4	存储器 Verilog HDL 描述	421
6.4.1	ROM 的 Verilog HDL 描述	421
6.4.2	RAM 的 Verilog HDL 描述	422
6.5	有限自动状态机 Verilog HDL 描述	423
6.5.1	FSM 设计原理	423
6.5.2	FSM 的应用——序列检测器的实现	428
6.5.3	FSM 的应用——交通灯的实现	432
6.6	算法状态机 Verilog HDL 描述	434
6.6.1	算法状态机原理	435

6.6.2	ASM 到 Verilog HDL 的转换	436
第 7 章	复杂数字系统设计和实现	439
7.1	设计所用外设的原理	439
7.1.1	LED 驱动原理	439
7.1.2	开关驱动原理	440
7.1.3	七段数码管驱动原理	440
7.1.4	VGA 显示器原理	442
7.1.5	通用异步接收发送器原理	445
7.2	系统中各个模块的功能	447
7.3	创建新的设计工程	449
7.4	Verilog HDL 数字系统设计流程	450
7.4.1	创建 divclk1.v 文件	450
7.4.2	创建 divclk2.v 文件	451
7.4.3	创建 divclk3.v 文件	452
7.4.4	创建 divclk4.v 文件	453
7.4.5	创建 pwm_led.v 文件	454
7.4.6	创建 counter4b.v 文件	456
7.4.7	创建 seg7display.v 文件	457
7.4.8	创建 uart.v 文件	460
7.4.9	创建显示处理文件	462
7.4.10	创建 top.v 文件	468
7.5	添加 XDC 约束	470
7.6	设计下载和验证	474
第 8 章	数模混合系统设计	475
8.1	信号采集和处理的实现	475
8.1.1	XADC 模块原理	475
8.1.2	XADC 原语	476
8.1.3	1602 字符 LCD 模块原理	480
8.1.4	信号采集、处理和显示的实现	485
8.2	信号发生器的实现	496
8.2.1	DAC 工作原理	496
8.2.2	函数信号产生原理	499
8.2.3	设计实现	499
第 9 章	片上嵌入式系统的构建和实现	508
9.1	ARM AMBA 规范	508
9.2	Cortex-M1 内部结构和功能	509
9.2.1	处理器内核及寄存器组	511
9.2.2	Cortex-M1 存储空间及映射	515
9.2.3	系统控制寄存器	517
9.2.4	内核存储器接口	523

9.2.5	嵌套向量中断控制器	525
9.2.6	总线主设备	533
9.2.7	AHB-PPB	534
9.2.8	调试	534
9.3	Cortex-M1 系统时钟和复位	536
9.4	Cortex-M1 嵌入式系统硬件设计	537
9.4.1	建立新的嵌入式设计工程	537
9.4.2	定制七段数码管 IP 核	538
9.4.3	定制按键消抖 IP 核	543
9.4.4	设置 IP 核路径	547
9.4.5	连接 IP 构建嵌入式系统硬件	548
9.4.6	对块设计进行预处理	553
9.5	Cortex-M1 指令系统	557
9.5.1	Thumb 指令集	557
9.5.2	汇编语言格式	558
9.5.3	寄存器访问指令——MOVE	559
9.5.4	寄存器访问指令——LOAD	559
9.5.5	存储器访问指令——STORE	560
9.5.6	多个数据访问指令	561
9.5.7	堆栈访问指令	561
9.5.8	算术运算指令	561
9.5.9	逻辑操作指令	562
9.5.10	移位操作指令	563
9.5.11	逆序操作指令	564
9.5.12	扩展操作指令	564
9.5.13	程序流控制指令	564
9.5.14	存储器屏障指令	565
9.5.15	异常相关指令	565
9.5.16	休眠相关的指令	565
9.5.17	其他指令	565
9.6	Cortex-M1 嵌入式系统软件设计	566
9.6.1	建立嵌入式软件工程	566
9.6.2	设置选项	567
9.6.3	添加汇编文件	567
9.6.4	添加头文件	570
9.6.5	添加主文件	572
9.6.6	生成 HEX 文件	573
9.7	处理并验证设计	573
第 10 章	图像采集、处理系统的构建和实现	575
10.1	图像传感器的原理和驱动	575

- 10.1.1 传感器结构和功能 575
- 10.1.2 传感器引脚功能定义 577
- 10.1.3 SCCB 接口驱动时序 578
- 10.1.4 SCCB 接口驱动的实现 581
- 10.2 Sobel 算子基本原理和实现方法 584
- 10.3 RGB444 数据捕获原理及实现 588
- 10.4 系统整体结构和子模块设计 590
 - 10.4.1 Vivado 中的系统整体结构 591
 - 10.4.2 时钟发生器的配置 596
 - 10.4.3 片内 RAM 模块的配置 596
 - 10.4.4 VGA 驱动模块 597
 - 10.4.5 行缓存模块 598

第 1 章 数字逻辑基础

本章主要介绍了数字逻辑的发展史、SPICE 仿真工具基础、开关系统、半导体数字集成电路、基本逻辑门电路及特性、逻辑代数理论、逻辑表达式的化简、毛刺产生及消除，以及数字码制表示和转换。

为了降低读者的学习难度，在介绍本章重要的知识点时引入了 SPICE 仿真实例，从多个角度对数字逻辑基础知识进行了解读。

本章内容是学习数字逻辑电路的基础，读者必须理解并掌握本章的内容，为后续学习数字逻辑电路知识打下坚实的基础。

1.1 数字逻辑的发展史

在过去的 60 年中，数字逻辑改变了整个世界，整个世界朝着数字化方向发展。今天我们所熟悉的计算机是在第二次世界大战后才出现在人类世界中的。表 1.1 给出了计算机和数字逻辑发展历史中的重大事件，从该表可以看出数字逻辑设计技术经过了近 400 年逐步演化的过程。

表 1.1 计算机和数字逻辑发展历史中的重大事件

年 份	事 件
1614—1617 年	John Napier，苏格兰数学家，发明了对数，允许通过加法实现乘法运算，以及通过减法实现除法运算
1623 年	Wilhelm Schickard，德国教授，发明了第一个机械式计算器，称为计算钟
1630 年	William Oughtred，英国数学家、牧师，发明了计算尺
1642—1644 年	Blaise Pascal，法国数学家、物理学家和宗教哲学家，发明了第一个机械计算器 Pascaline
1672—1674 年	Gottfried Wilhelm Von Leibniz，德国数学家、外交官、历史学家、法学家、微分的发明家，发明了一个称为步进式计算器的机械计算器。计算器有一个齿轮（莱布尼茨轮），用于机械式的乘法器。尽管没有使用这个计算器，但是该设计对未来机械式计算器的发展产生了深远的影响
1823—1839 年	Charles Babbage，英国数学家、发明家，开始设计差分机，该机器设计用于自动地处理对数计算，但由于各种因素最终没有完成差分机。1834 年，Babbage 开始设计一个功能更强的机器，称为分析机，它被称为第一个通用计算机。所以，Babbage 被认为是“计算机之父”
1854 年	George Boole，英国逻辑学家、数学家，出版了《Investigation of the Law of Thought》，奠定了数字逻辑的基础
1890 年	Herman Hollerith，美国发明家，使用打孔卡片制表，用于 1890 年的普查。1896 年，他成立了打卡机公司，最终于 1924 年演变成了 IBM 公司
1906 年	Lee De Forest，美国物理学家，发明了三极管，即具有 3 个电极的真空管。在 1940 年之前，这些管子并没有用于计算机中
1936 年	Alan M. Turing，英国逻辑学家，发表了论文“On Computable Numbers”，说明任意的计算都可以使用有限状态机实现。在第二次世界大战后英国早期的计算机研制中，Turing 扮演了非常重要的角色
1937 年	George Stibitz，贝尔电话实验室的一个物理学家，使用继电器建立了二进制电路，实现了加法、减法、乘法和除法运算