

【博客藏经阁丛书】

那些年， 我们拿下了FPGA

苏 阳 蒋银坪 邢培飞 编著



北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS

【博客 赢经网 丛书】

· 014009700

TP332.1
131

内容简介

那些年， 我们拿下了FPGA

苏 阳 蒋银坪 邢培飞 编著



TP332.1
131



北航 C1695927



北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS

内 容 简 介

本书着眼于目前发展火热的 FPGA 技术,由浅入深地介绍了 FPGA 技术的学习过程和学习方法。首先,从应用场景到可用资源,从提出问题到拿出解决方案,多角度解读 FPGA;其次,在描述数字系统设计方法的基础上,以环环相扣的方式详细介绍 FPGA 的开发流程;然后,结合开发工具,以独有的章节穿插方式,图文并茂地介绍 Verilog 语法的方方面面;最后,寄娱乐于学习,举例说明 FPGA 在控制音乐及液晶显示器方面的应用设计。作者结合自己的经验,在书中或以仿真举例,或以截图的形式,总结了许多在语法学习过程中需要注意的细节以及工具使用的技巧,希望能帮助读者全面掌握 FPGA 知识。

本书适合一切关注 FPGA 的人,特别是初学者,以及即将从事 FPGA 工作的人士。同时,本书对于已经具有比较丰富的设计经验但又想全面梳理 FPGA 知识的读者来说,也是一本很好的参考书。

图书在版编目(CIP)数据

那些年,我们拿下了 FPGA / 苏阳,蒋银坪,邢培飞
编著. — 北京:北京航空航天大学出版社,2013.10
ISBN 978-7-5124-1232-3

I. ①那… II. ①苏… ②蒋… ③邢… III. ①可编程
序逻辑器件—系统设计 IV. ①TP332.1

中国版本图书馆 CIP 数据核字(2013)第 196749 号

版权所有,侵权必究。

那些年,我们拿下了 FPGA

苏 阳 蒋银坪 邢培飞 编著

责任编辑 王静竞

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(邮编 100191) <http://www.buaapress.com.cn>

发行部电话:(010)82317024 传真:(010)82328026

读者信箱:emsbook@gmail.com 邮购电话:(010)82316936

涿州市新华印刷有限公司印装 各地书店经销

*

开本:710×1 000 1/16 印张:22.25 字数:474 千字

2013 年 10 月第 1 版 2013 年 10 月第 1 次印刷 印数:4 000 册

ISBN 978-7-5124-1232-3 定价:49.00 元

若本书有倒页、脱页、缺页等印装质量问题,请与本社发行部联系调换。联系电话:(010)82317024

前言

开始的开始,也就是5年前,现在的你也许和我一样,学习起来像青春生活一样,迷茫!于是,我带着这些迷茫穿梭于图书馆的各个楼层,寻找着机缘,也寻找着可以倚仗过生活的手艺。为了在将来过得好一些,我努力地学习各种热门技术,网站、单片机、ARM等都尝试着学习过。唯有FPGA,陪我度过了5年多的时间,伴我走到迷茫的尾巴。点滴过程,记录于此,纪念那些在一起的日子,也祭奠即将逝去的迷茫!

这是一本再思考的回忆录,越是基础的东西就越是值得再思考、再研究、再掌握。基础是求大发展的资本,学习更是如此。本记录对可编程逻辑器件FPGA的设计重新进行了思考,虽然是初级知识,但其中再思考的过程还是会引起共鸣。所以本书既适合初学者,也适合中高级设计者,也许你能从中找到当年的你。

这是一本文艺青年的书,以面对面的语言进行陈述,其中没有需要琢磨的技术名词。特别是FPGA基础知识和语法,相比其他同类书里各种晦涩难懂的名词罗列,更多的是风趣形象的比喻和直白清新的描述。学员会有读小说那样的轻松,从而更加有兴趣、有效率和有信心地学习。

这是一本谙熟学习之道的书,打破以往技术类的编排方式,以再次学习的角度,带着和初学者一样的问题,来进行章节的编排。本书从FPGA的历史说起,分别介绍了它是什么,可以用来干什么,怎么使用这几个部分。同时在怎么使用的环节中,依次讲解了整体开发流程、基本语法、开发工具和高级语法。这样一来,做到边学边练,避免了枯燥的语法学习。

本书稿部分语法章节的初稿由邢培飞和蒋银坪完成。

感谢潘丹博士、邱望洁博士及安宏勇在百忙之中抽出时间参与部分章节的审稿工作。

最后的最后,感谢北京航空航天大学出版社的大力支持,感谢那些在书稿写作困难的时候给予动力的人和事,感谢迷茫青春,感谢FPGA!

苏 阳

2013年2月于洱海虫二客栈

目 录

第 1 章 从春天走来	1
1.1 撒下一把种子	1
1.2 繁芜苏醒	2
1.3 鹤立鸡群	5
1.4 丰 满	6
第 2 章 看透你的心	10
2.1 灵活,并不是善变——SRAM	10
2.1.1 动静 RAM 之间	10
2.1.2 SRAM 是如何构成 FPGA 的	11
2.1.3 往细处讲讲查找表	12
2.1.4 关于 LUT 现状的讨论	13
2.1.5 灵活归一码,也有缺点	14
2.2 想回到过去,难也——反熔丝	15
2.2.1 何为熔丝,何为反熔丝	15
2.2.2 回不去,得从编程机理说起	15
2.2.3 以小见大——最小单元 MUX	16
2.2.4 回不去的痛	17
2.3 灵魂,一块住进去——FLASH	18
2.3.1 FLASH,灵魂的驻扎地	18
2.3.2 一小步一个大家族	18
2.3.3 挺好的了,只是时候未到	18
第 3 章 走进生活,生生不息	20
3.1 你存在,眼睛的世界	20
3.1.1 概 述	20
3.1.2 视频编解码	20

3.1.3	目标识别	21
3.1.4	图像处理支持资源	21
3.2	你，使我们走近	24
3.2.1	有线通信	24
3.2.2	无线通信	25
3.2.3	通信领域支持资源	26
3.3	你，承载着“0”和“1”的世界	29
3.3.1	概 述	29
3.3.2	数字处理系统模型	29
3.3.3	五花八门的方式	30
3.3.4	其实挺有优势	31
3.3.5	DSP 支持资源	32
3.4	你，可以像大脑一样思考	33
3.4.1	概 述	33
3.4.2	创新挑战和机遇	34
3.4.3	嵌入式支持资源	35
3.5	可以考虑这些解决方案	36
第4章	像绘画一样设计系统	38
4.1	抽出系统设计模型	38
4.2	最美不过过程	39
4.2.1	系统分析	39
4.2.2	方案设计	39
4.2.3	top-down	40
4.2.4	硬件化	40
4.2.5	平台选择	41
第5章	开发流程走一遍	42
5.1	设计输入	43
5.1.1	设计输入方式	43
5.1.2	输入方式使用探讨	45
5.2	综 合	46
5.2.1	编 译	47
5.2.2	映 射	47
5.3	布局布线	48
5.3.1	布 局	48

5.3.2	布 线	49
5.4	约 束	49
5.4.1	综合约束	49
5.4.2	位置约束	50
5.4.3	时序约束	50
5.5	开发仿真	51
5.5.1	测试平台	51
5.5.2	RTL 级仿真	52
5.5.3	静态仿真	52
5.5.4	时序仿真	53
5.6	静态时序分析	54
5.7	在线调试	54
5.8	配置及固化	55
5.8.1	FPGA 配置过程	55
5.8.2	举例——Altera FPGA 配置全过程	55
5.8.3	配置模式	56
5.8.4	模式选择	58
5.9	开发工具总结	59
第 6 章 Verilog HDL or VHDL?		61
6.1	Verilog HDL 基本结构	61
6.1.1	端口列表定义	62
6.1.2	I/O 说明	63
6.1.3	内部信号声明	64
6.1.4	模块功能的定义	64
6.2	VHDL 基本结构	65
6.2.1	程序包说明	65
6.2.2	实体说明	66
6.2.3	结构体说明	66
6.3	民主的 Verilog	67
6.4	Verilog 升级 SystemVerilog	67
第 7 章 拿下常量数据类型		69
7.1	我该如何表达整数	69
7.1.1	Verilog 中整数长啥样	69
7.1.2	负 数	70

7.1.3	X 和 Z 之间的事	70
7.1.4	变与不变的参量型	72
7.1.5	我和整数的约定	73
7.2	实数常量是几许人物	74
7.3	字符串是怎么回事	75
第 8 章	拿下变量数据类型	76
8.1	多变的变量	76
8.1.1	网络(net)连接类型	76
8.1.2	寄存器(Register)类型	85
8.2	wire 和 reg 才是精干	86
8.2.1	wire 型	86
8.2.2	reg 型	87
8.3	不要以为 reg 就是触发器	88
8.4	memory 其实也是 reg	89
8.4.1	该怎么声明 memory 型变量	89
8.4.2	该怎么对 memory 型变量赋值	89
8.5	inout 该怎么定义类型	91
第 9 章	拿下运算符	92
9.1	林林总总的运算符	92
9.1.1	逻辑运算符	92
9.1.2	按位运算符	94
9.1.3	缩减运算符	96
9.1.4	关系运算符	96
9.1.5	等式运算符	97
9.1.6	移位运算符	99
9.1.7	位拼接运算符	100
9.1.8	三目运算符	101
9.2	给它们排个序吧	102
9.3	赋值语句	102
9.3.1	阻塞赋值	103
9.3.2	非阻塞赋值	104
9.3.3	塞还是不塞	105

第 10 章 不再纠结 if 和 case	107
10.1 if 不同形态	107
10.2 case 不同形态	109
10.3 if 和 case 之间的转换	111
10.4 if 和 case,谁更好	112
10.5 不完全的 if 和 case	115
10.6 关于 case 的约束	117
第 11 章 循环语句	120
11.1 只能仿真的循环	120
11.1.1 repeat 是个定数	120
11.1.2 forever 能永远吗	122
11.1.3 有限的 while	125
11.2 特立独行的 for	126
11.2.1 for 的基本形态	126
11.2.2 可以综合的 for	127
第 12 章 块语句以及过程块	130
12.1 说说 begin_end	130
12.2 说说 fork_join	132
12.3 说说 assign	133
12.4 说说 always	133
第 13 章 抽象级别的描述	136
13.1 不同抽象级别概述	136
13.2 结构级描述	137
13.2.1 门级结构描述及其说明语法	137
13.2.2 用门级结构描述二选一选择器	140
13.3 行为级描述	141
13.3.1 行为级与 RTL 级区别	141
13.3.2 RTL 级描述二选一选择器	142
第 14 章 层次化建设及模块例化	145
14.1 层次化建模概念	145
14.2 模块例化	146

14.3	四位脉动进位计数器	147
第 15 章	建设测试磨坊	149
15.1	编写 Testbench 目的	149
15.2	基本的 Testbench 结构	149
15.3	产生激励的一些描写方式	150
15.3.1	产生时钟的几种方式	150
15.3.2	产生复位信号的几种形式	151
15.4	关于测试的建议	152
第 16 章	仿真工具基本使用	153
16.1	ModelSim 简介	153
16.2	ModelSim HDL 仿真优势	153
16.3	ModelSim 基本使用	154
16.3.1	ModelSim 使用流程	155
16.3.2	简单 2-4 编码器 RLT 仿真实例	155
16.4	高级命令方式仿真	167
16.4.1	常用一些命令	167
16.4.2	其他命令学习方法	169
16.4.3	do 文件的编写与使用	170
16.5	ModelSim 私藏技巧	173
16.5.1	利用 ModelSim 模板建立 Testbench	173
16.5.2	Wave 查看技巧	175
16.5.3	仿真代码的修改	178
第 17 章	组合逻辑专题	179
17.1	组合逻辑描述方法	179
17.1.1	assign 语句实现组合逻辑	179
17.1.2	always 块实现组合逻辑	180
17.2	常用组合电路	183
17.2.1	多路器	183
17.2.2	加法器	185
17.2.3	译码器	186
17.2.4	编码器	188

第 18 章 时序逻辑专题	191
18.1 时序逻辑描述方法	191
18.2 再来两个例子	194
18.2.1 分频电路	194
18.2.2 沿提取电路	197
第 19 章 函数与任务专题	199
19.1 task	199
19.1.1 task 特点及描述	199
19.1.2 task 举例说明	200
19.2 function	201
19.2.1 function 特点及描述	201
19.2.2 function 举例说明	202
19.3 task 和 function 不混淆	203
19.4 系统任务与函数	204
19.4.1 标准输出任务	204
19.4.2 文件管理任务	206
19.4.3 仿真控制任务	209
19.4.4 时间函数	209
19.4.5 其 他	210
第 20 章 编译预处理与可综合性	211
20.1 编译预处理概念	211
20.2 常用编译预处理	211
20.2.1 <code>`define</code> 、 <code>`undef</code>	211
20.2.2 <code>`include</code>	213
20.2.3 <code>`timescale</code>	215
20.2.4 <code>`ifdef</code> 、 <code>`elsif</code> 、 <code>`else</code> 、 <code>`endif</code>	219
20.3 不再烦恼可综合性	221
第 21 章 状态机专题	223
21.1 状态机的概念	223
21.2 状态机的分类	224
21.2.1 Moore 状态机	224
21.2.2 Mealy 状态机	225

21.3	状态机实现步骤现场	225
21.3.1	三要素的提取	225
21.3.2	状态转移图	226
21.3.3	状态编码	227
21.3.4	状态机的描述	228
21.4	不同状态机描述方式	228
21.4.1	一气呵成的一段式	228
21.4.2	一看即懂的二段式	231
21.4.3	高效率的三段式	233
21.5	safe 状态机	237
第 22 章	开发工具秘籍	238
22.1	QuartusII 简介	238
22.2	工程实例介绍	241
22.3	准 备	242
22.4	新建工程	243
22.5	设计输入	248
22.6	分析综合	265
22.7	布局布线	269
22.8	下载与固化	273
22.8.1	生成配置和固化文件	273
22.8.2	准备工作	274
22.8.3	下 载	274
22.8.4	固 化	275
22.9	总 结	277
第 23 章	联合仿真和调试	278
23.1	ModelSim 与 Quartus II 联合使用	278
23.1.1	联合准备工作	278
23.1.2	Altera 仿真库的介绍	279
23.1.3	利用 ModelSim 做前仿真	280
23.1.4	利用 ModelSim 做静态仿真	285
23.1.5	利用 ModelSim 做时序仿真	290
23.2	在线调试与 SignalTap II 的使用	292
23.2.1	创建 STP 文件	292
23.2.2	设置采集时钟	292

23.2.3	在 STP 文件中分配信号	295
23.2.4	指定采样点数及触发位置	296
23.2.5	逻辑分析仪触发控制	297
23.2.6	编译嵌入 SignalTap II 逻辑分析仪的设计	300
23.2.7	SignalTap II 分析器件编程	301
23.2.8	查看 SignalTap II 采样数据	301
23.2.9	数据保存	302
第 24 章	IP 核, 拿来主义	304
24.1	IP 核的概念	304
24.2	RAM 核使用	304
24.3	FIFO 使用	308
24.4	PLL 核使用	312
第 25 章	用 FPGA 来搞搞音乐	317
25.1	演奏原理	317
25.2	音调的控制	317
25.3	音长的控制	319
25.4	《天空之城》	320
第 26 章	You Got It!	325
26.1	显示设备 H1602B	325
26.1.1	H1602B 端口与电路连接	325
26.1.2	H1602B 的字符集和指令集	326
26.2	显示实现	328
	参考文献	336

代码块索引

代码块 6-1: Verilog 描述分频器	61
代码块 6-2: VHDL 描述分频器	65
代码块 7-1: 整数表达示例	70
代码块 7-2: 负数表达	70
代码块 7-3: Z 值和 X 值表示示例	70
代码块 7-4: X 值运算示例	71
代码块 7-5: Z 值运算示例	71
代码块 7-6: 参数定义示例	72
代码块 7-7: 参数传递示例	72
代码块 7-8: 下划线示例	73
代码块 7-9: 位宽省略示例一	74
代码块 7-10: 位宽省略示例二	74
代码块 7-11: 实数常量示例	74
代码块 8-1: 多驱动源示例	77
代码块 8-2: 代码块 8-1 双驱动测试	77
代码块 8-3: supply0 与 supply1 示例	79
代码块 8-4: wor 与 trior 示例	80
代码块 8-5: wand 与 triand 示例	81
代码块 8-6: trireg 示例	82
代码块 8-7: 代码块 8-6 测试平台	82
代码块 8-8: tril 及 tri0 示例	84
代码块 8-9: integer 向量访问非法示例	86
代码块 8-10: wire 声明示例	86
代码块 8-11: wire 缺省位宽示例	86
代码块 8-12: wire 型用例	87
代码块 8-13: reg 格式声明	87
代码块 8-14: reg 在 always 中应用	87
代码块 8-15: reg 负值示例	88

代码块 8-16:非触发器 reg 示例	88
代码块 8-17:memory 参数索引示例	89
代码块 8-18:memory 非法赋值示例	90
代码块 8-19:memory 单元赋值示例	90
代码块 8-20:memory 遍历赋值示例	90
代码块 8-21:inout 使用示例	91
代码块 9-1:逻辑运算符示例	93
代码块 9-2:按位运算示例	95
代码块 9-3:缩减运算符示例	96
代码块 9-4:关系运算符示例	96
代码块 9-5:逻辑等示例	98
代码块 9-6:严格等示例	98
代码块 9-7:逻辑等与严格等区别示例	98
代码块 9-8:移位运算示例	99
代码块 9-9:位拼接运算定义寄存器	101
代码块 9-10:三目运算示例	101
代码块 9-11:三目运算 X 与 Z 值情况示例	102
代码块 9-12:阻塞赋值示例	103
代码块 9-13:非阻塞赋值示例	104
代码块 9-14:阻塞与非阻塞区别	105
代码块 10-1:半句式 if 示例	107
代码块 10-2:全句式 if 示例	107
代码块 10-3:罗列式 if 示例	108
代码块 10-4:if 复合型语句示例	108
代码块 10-5:if 嵌套语句示例	108
代码块 10-6:if 匹配对示例	109
代码块 10-7:casex 优先级示例	110
代码块 10-8:case 方式描述	111
代码块 10-9:if 方式描述	111
代码块 10-10:不完全 if 结构	115
代码块 10-11:不完全 case 结构	116
代码块 10-12:无 default 完全 case 结构	116
代码块 10-13:FULL_CASE 约束示例	117
代码块 10-14:引导语句 case 约束	118
代码块 11-1:带 X 值的 repeat 示例	120
代码块 11-2:带 Z 值的 repeat 示例	121

代码块 11-3:用 repeat 语句实现乘法器	122
代码块 11-4:forever 产生时钟	123
代码块 11-5:forever 后语句不执行示例	123
代码块 11-6:forever 停止示例	124
代码块 11-7:while 示例	125
代码块 11-8:for 的阶乘示例	127
代码块 11-9:可综合 for 的阶乘示例	128
代码块 11-10:用 for 循环统计数据示例	128
代码块 12-1:顺序块示例	130
代码块 12-2:块名引用示例	131
代码块 12-3:并行块示例	132
代码块 12-4:assign 赋值示例	133
代码块 12-5:always 死锁示例	134
代码块 12-6:always 用法举例一	134
代码块 12-7:always 用法举例二	134
代码块 12-8:always 用法举例三	135
代码块 13-1:逻辑块延时声明示例	138
代码块 13-2:逻辑块的强度声明示例	139
代码块 13-3:二选一选择器门级描述	140
代码块 13-4:二选一选择器 RTL 级描述	142
代码块 14-1:模块例化示例	146
代码块 14-2:脉动进位器例化	148
代码块 15-1:forever 产生时钟激励	150
代码块 15-2:always 产生时钟	150
代码块 15-3:repeat 产生时钟	151
代码块 15-4:异步复位激励	151
代码块 15-5:同步复位激励	152
代码块 16-1:2-4 编码器	156
代码块 16-2:2-4 编码器测试模块	156
代码块 16-3:vlib 命令示例	168
代码块 16-4:vmap 命令示例	168
代码块 16-5:vlog 命令示例	168
代码块 16-6:vsim 命令示例	169
代码块 16-7:run 命令示例	169
代码块 16-8:do 命令示例	169
代码块 16-9:do 文件	171

代码块 17-1: assign 连续赋值	179
代码块 17-2: assign 描述三态门	180
代码块 17-3: 举例状态机示例	181
代码块 17-4: 四选一示例	183
代码块 17-5: 加法器示例	185
代码块 17-6: always 实现加法器	186
代码块 17-7: 3-8 译码器	187
代码块 17-8: 8-3 优先编码器	189
代码块 18-1: D 触发器示例	191
代码块 18-2: 清零置位 D 触发器	192
代码块 18-3: 4 分频时序电路	194
代码块 18-4: 6 分频时序关键段	195
代码块 18-5: 4 分频时序电路另一种实现	196
代码块 18-6: 沿提取实现	197
代码块 19-1: task 应用举例	200
代码块 19-2: 函数应用举例	202
代码块 19-3: \$ display 任务	205
代码块 19-4: 特殊字符输出	205
代码块 19-5: 默认打开方式示例	207
代码块 19-6: 输出到文件示例	207
代码块 19-7: 文件读取示例	208
代码块 19-8: 文件读出注意事项举例	208
代码块 19-9: \$ finish 和 \$ stop 示例	209
代码块 19-10: \$ time 示例	209
代码块 19-11: \$ random 示例	210
代码块 20-1: define 宏定义示例	212
代码块 20-2: 宏定义嵌套	212
代码块 20-3: 宏定义陷阱	212
代码块 20-4: include 应用举例	213
代码块 20-5: include 示例	213
代码块 20-6: include 嵌套使用	215
代码块 20-7: 时间精度大于时间单位测试	216
代码块 20-8: 精度舍入示例	217
代码块 20-9: 精度统一示例	217
代码块 20-10: ifdef 示例	219
代码块 21-1: 一段式实现程序	228