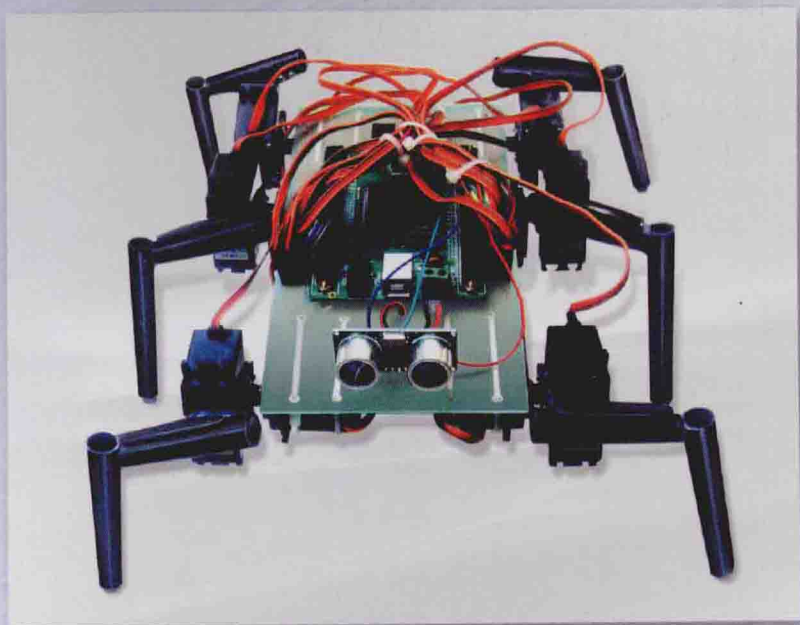


FPGA

硬件软件设计 及项目开发

李裕华 马慧敏 编著

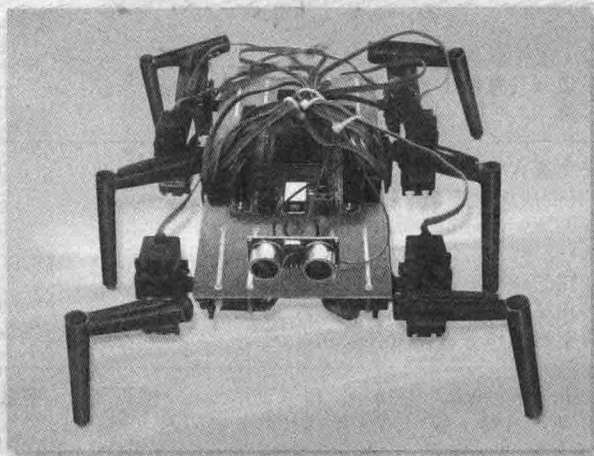


西安交通大学出版社
XI'AN JIAOTONG UNIVERSITY PRESS

FPGA

硬件软件设计 及项目开发

李裕华 马慧敏 编著



西安交通大学出版社
XI'AN JIAOTONG UNIVERSITY PRESS

内容提要

本书是入门级的教材,采用直通车的方式介绍内容,从硬件介绍到项目开发,一条路、一口气完成。本书介绍了 FPGA 的知识,横向剖析和比较了 ROM 与 FPGA 的关系,详细介绍了 Spartan-3E 系列 FPGA 芯片的结构,详细介绍了如何用 DXP 2004(Protel 2004)设计 FPGA 开发板。完全公布书中 FPGA 开发板的全部原理图和 PCB 图,以及生成的过程。手把手介绍开发环境 ISE 和软件语言 VHDL。最后,介绍了用 FPGA 开发的 22 个项目的真实源代码和详细注释。

本书适合作为计算机专业硬件开发课程的教材,也适合 FPGA 开发的爱好者和相关技术人员。

作者可以提供书中硬件设计资料和全部源代码。

作者电话:13709181375, E-Mail: yhli@mail. xjtu. edu. cn, qq: 630972994

qq 群: 290863896, 微信号: liyuhua686994。

图书在版编目(CIP)数据

FPGA 硬件软件设计及项目开发/李裕华,马慧敏编著.

—西安:西安交通大学出版社,2014.8

ISBN 978-7-5605-6640-5

I. ①F… II. ①李… ②马… III. ①可编程程序逻辑
器件—系统设计—教材 IV. ①TP332.1

中国版本图书馆 CIP 数据核字(2014)第 194354 号



书 名 FPGA 硬件软件设计及项目开发
编 著 李裕华 马慧敏
责任编辑 叶 涛

出版发行 西安交通大学出版社
(西安市兴庆南路 10 号 邮政编码 710049)

网 址 <http://www.xjtpress.com>
电 话 (029)82668357 82667874(发行中心)
(029)82668315 82669096(总编办)

传 真 (029)82668280
印 刷 陕西宝石兰印务有限责任公司

开 本 787mm×1092mm 1/16 印张 25.25 插页 1 字数 612 千字
版次印次 2014 年 9 月第 1 版 2014 年 9 月第 1 次印刷
书 号 ISBN 978-7-5605-6640-5/TP·633
定 价 49.90 元

读者购书、书店添货、如发现印装质量问题,请与本社发行中心联系、调换。

订购热线:(029)82665248 (029)82665249

投稿热线:(029)82664954

读者信箱: jdlgy@yahoo. cn

版权所有 侵权必究

序

我们生活在一个快速连接的世界中。目前,全球有 60 亿台移动计算设备相互连接,并且每天都新增 100 万台移动设备;预计到 2020 年,全球移动计算设备总数将达到 300 亿台。随着物联网(IoT)以及万联网(IOE)的发展,海量大数据的存储、传输、处理、挖掘,对技术提出了极大的挑战。从处理响应速度来看,计算的响应速度,从文字时代的秒级,多媒体时代的百毫秒级,视频时代的十毫秒级,会迅速推进到 5G 时代的 1 毫秒级。对海量数据在 1 毫秒内完成处理,将是未来数十年摆在电子信息设计工程师面前的巨大问题。

随着摩尔定律走向深纳米时代,在 20nm 以下的工艺节点,每个节点的性价比提高幅度比上一代将会逐步减少,而前期 NRE 费用(即一次性工程费用)投入巨大,服务客户数量稀少,使得专用集成电路 ASIC 以及专用标准集成电路 ASSP 在商业模式上走入绝境。如此直去,赢利的公司数目锐减,直至最后消亡,尚能存活的将是可编程器件。

面对海量的计算任务,多核并行曾是解决方案之一,但受制于算法可并行部分的局限,更多的核并不能带来更高的效率,加速效能也逐渐走到了尽头。而受单颗芯片发热量密度限制,即便芯片上集成的晶体管越来越多,但可同时运行的晶体管数目趋于恒定,其它晶体管沦为暗硅(Dark Silicon)。设计者的目光不得不转向冯诺依曼结构之外的计算构架,比如领域定制化计算(Domain Specific Computing),它可在保持灵活性的同时,发挥每一个晶体管的计算能力,当然,这也离不开可编程技术的长足发展。

在系统级别,大数据、软件定义一切、虚拟化一切的趋势,使得系统构架工程师不得不寻求更灵活、更智慧、更快速、更绿色的解决方案。而这些解决方案的核心,往往与软件、硬件、I/O 均可编程的芯片,赛灵思公司的 All Programmable 芯片相关。

在教育领域,除了需要培养能够应对未来数十年挑战的电子信息工程师之外,本身也充满了变革和机遇。随着大规模网上课程 MOOC 的兴起,通过网络,

在统一平台下,以翻转课堂的方式,打破业界与教育界的壁垒,完成软硬件、实验与课程、年级与院系的全面贯通。这正是很多电子信息类教育工作者的毕生追求。

赛灵思大学计划将不遗余力地帮助教育工作者应对这些变革,将 All Programmable 全可编程技术,全面引入新型知识传播体系中去,培养能够处理下一代电子系统设计挑战的未来卓越工程师和创新者,为将中国制造变成中国智造的梦想,提供充足的智力和人才保障。

西安交通大学城市学院的《FPGA 软件硬件设计及项目开发》一书,通俗易懂,从 FPGA 实现简单 74LS00 逻辑入手,利用开源 PCB 开发板,以小项目的形式,循序渐进掌握 FPGA 的编程方法,是一本适合初学者快速入门的好书。特此作序,以示支持。

谢凯年

赛灵思大学计划中国区经理

前言

FPGA 是一片芯片,中文名为“现场可编程门阵列”,它是最近十几年里发展出来的新型数字电子电路芯片。本书的目的就是教会读者如何从硬件设计到软件开发两个方向上使用这片芯片。

本书瞄准的是赛灵思(Xilinx)公司的 Spartan-3E 系列的 FPGA,型号为 XC3S500E。用到的开发环境为赛灵思公司的 ISE,软件语言为 VHDL。这片 FPGA 内含有 50 万个逻辑门,以及其它资源。FPGA 到底有什么特色?它被称为“半定制”器件。就是说,芯片内部集成的这些硬件资源,50 万逻辑门都没有连接好,提供给用户按照自己的意愿来连接,构建用户自己的独特的电路。这样构建的电路是稳定的,可工业使用的。当然,电路还可以拆除,可以改建,可以刷新重建。这实际上圆了一些工程师的梦,他们曾经梦想自己设计 IC(集成电路芯片)。设想一下这样的场景,在你面前码放着 50 万个微小的逻辑门电路单元,让你自由支配,这是一个什么样的诱惑。有网友把 FPGA 比作橡皮泥,让你自己来拿捏,塑造你自己渴望的塑像。作者却愿意把 FPGA 比作变形金刚,可以演变成无数的变种。一个数据比较可以说明 FPGA 的最大的优越性。如果已经开发出来了一个硬件电路的应用,现在要升级。如果是传统的电路板,这样的升级就要重新设计电路板,制作电路板,元件的焊接等等,一定会花去一个月以上的时间,老板子只能变为电子垃圾。如果是用 FPGA 的电路板,那么这样的硬件升级可能几个小时就能搞定,而且线路板不用从系统上拔下来。这样的优势是当前 FPGA 技术应用快速发展的主要原因。

学习 FPGA 难不难?与学习单片机,学习 32 位嵌入式系统相比,可以这么说,学习 FPGA 入门比较烦,入门后会感觉比较容易。而学习单片机和嵌入式系统正好相反。这有点像儿童学小提琴和学钢琴,同样是乐器中的顶级,一个是乐器皇后,一个乐器之王。学钢琴入门容易,坐姿很舒服,手臂动作很自然,弹下去的音很好听,因为音是准的。大约学一个小时,就能弹出很美好的乐曲《小小星星亮晶晶》,让家长鼓掌叫好。学小提琴则完全不一样,脖子要能夹住琴,左手向前方拐成 45° ,琴轴要和地面平行,琴面要轴向旋转 45° ,右手的 5 指捏弓要虚空而稳定,右小臂要和弓成 90° ,右手腕要自然垂下。最要命是,拉弓要定在琴弦的一个点上,拉弓的运动必须是顺着弓的直线(要知道,人的胳膊的运动从来都是弧线),还要和琴弦保持 90° ,使琴弦产生纯的横向振动。只要拉弓动作不是 90° ,或者弓压在琴弦上的力度不够或太大,琴弦就产生纵向振动,产生的是极难听的噪音。此外,还要把左手指放在琴弦的绝对正确的位置上,这个位置没有标记,否则音不准。初学提琴者,无不在很长的时间里,不断地拉出杀鸡般的声音。这是入门的难易问题。但入门后就反过来了:钢琴发展下去,十个手指在弹奏,每一瞬间要看乐谱的几个音来分配十个指头的动作;小提琴发展下去,每一瞬间只关心一个音和一个手指的动作。只要比较一下小提琴谱子和钢琴谱子就知道谁的信息量大了,可以斗胆地说,小提琴谱子的信息量不及钢琴谱子的信息量的 $1/4$ 。怎么来评判钢琴和小提琴的学习的难易度,就是说钢琴入门容易,继续发展很难,小提琴入门难,继续发展容易。学习 FPGA 大约像学习小提琴,学习单片机和嵌入式系统就像学钢琴。例如,让一个发光二极管(LED)闪

烁,大约就像《小小星星亮晶晶》乐曲,是单片机,嵌入式系统和 FPGA 的第一个实验。在单片机上和嵌入式系统上,几句 C 语言,编译,下载,很快就完成了,初学者的成就感油然而生。在 FPGA 上,编写代码后,要有一系列电脑操作才能实现在电路板上的 LED 闪烁起来。有点像学小提琴,准备动作太多,包括各种术语,开发环境的层次和多项操作。FPGA 看上去好像入门门槛比较高,其实不然,而是前期的预备知识有点繁琐。比较多的术语,比较多的操作步骤,较多的辅助工具。这些前期的预备知识分散了新手的精力。但是一旦进入 FPGA,以后的发展会容易,因为 FPGA 虽然基本电路单元数量庞大,但毕竟资源种类并不多。希望读者中的新手会有勇气坚持学下去。

本书的第 1 章很简单地介绍了一些 FPGA 的知识,横向地作了些比较,在发展历史里特别地剖析了 ROM 和 FPGA 的关系。

本书第 2 章详细地介绍了 Spartan-3E 系列的 FPGA 芯片的结构,这些资料来源于 Xilinx 公司的官方资料^[1]。这部分内容对初学者会有点困难,会感觉枯燥。对有些读者建议可以简单了解,缺乏这些知识并不影响后面几章的学习。等到掌握了 FPGA 的源代码编写技术以后,再返回来学习这一章,就会恍然大悟。如果掌握了这一章的内容,编写源代码时会技高一筹。

本书第 3 章详细地介绍了如何用 DXP 2004(Protel 2004)来设计本书中用到的 FPGA 开发板,公布全部原理图和 PCB 图,以及生成的过程。如果完全按照本书的内容走,一定能设计和制作出来的。作者的这个设计共修改了 3 次,才感觉到可能是无错的 PCB 了。这样的硬件设计细节介绍是其它 FPGA 书籍中没有的内容。市场上能看到很好的 FPGA 开发板,FPGA 的硬件设计在国内已有相当的水平,为什么没有人来写具体的硬件设计呢?估计是知识产权的问题,当内容接近实际产品时,通常工程师和老板们不会愿意完整公开内容的。所以技术书籍,或教材中几乎看不到 PCB 图,但这里作者愿意与读者共享自己的设计。

本书第 4 章介绍开发环境 ISE 和软件语言 VHDL。软件语言从来都是教学环节上的软肋,没有深度和逻辑,像字典一样地罗列,教师教课无法有激情,学生学得模模糊糊。软件语言应该在使用中学,这是大部分人都能接受的。就好像手机的使用说明书,很少有人认真地一点一点地仔细读一遍,然后再开机使用,差不多所有的人都是在使用中学习的。所以也建议按照手机的学习方法,先做第 5 章的项目,通过使用、拷贝、修改等方法掌握 VHDL 软件语言。开发环境 ISE 的学习也是同样的,在第 5 章的学习过程中掌握 ISE。特别提出第 4 章的 4.3 节,其中介绍了从创建一个新项目开始,直到完成烧写并运行项目的全部操作过程。这一节的内容类似于手把手的教学,虽然死板一点,但相信会帮助新手轻松入门。

本书第 5 章包含 22 个项目的真实源代码和它们的解释。根据过去的教学经验,不说“前面学过了,后面就不讲了”这句课堂的套话。而是尽可能做到每个项目的源代码都反复给予解释,希望直到读者都烦了,这才到达作者的目的。因为这样,新手才会将这些代码深深地刻在脑中。

通常的 FPGA 的书籍介绍的内容会很丰富,很完美,涉及的面会很多。但本书作为入门级的教材,希望的是保持读者的热情,想让读者学会了再说。为了不使读者分散精力,本书采用直通车的方式介绍内容。从硬件到最终的项目完成,一条路,一口气地完成。这是本书和其它有关 FPGA 书不一样的特点。FPGA 发展到现在,出现了很多辅助工具,很多仿真工具,也就是说,在从开始设计到项目完成的一条路上有很多分叉,通过这些分叉也许可以更优化地抵

达目标。但是本书略去了这些发展的分叉,使读者的精力聚焦在一条主线上。让本书成为初学者的第1本引进门型的教材,这是作者的初衷。

第5章中项目开发的前3个题目分别是仿照74LS00,仿照74LS138,仿照74LS164芯片的功能,这是作者引进读者入门的一条经验。这3个题目的设计是基于这样的想法,学习过数字电路的读者对于这3个经典的芯片是很熟悉的,所以项目的需求就会很清晰。74LS00是与非门,74LS138是3—8选一的解码器,74LS164是串入并出的移位寄存器。这相当于把过去的简单的熟记的知识拿来作为新知识的垫脚石,使它们作为新老知识过渡的桥梁。用传统逻辑芯片作为FPGA的第一个实践项目是作者的创新,因为差不多所有的FPGA教材都是以计数器作为FPGA的第一个实践项目。这是思维方式的不同。后者考虑的是如何在第一个实践项目中立刻体现FPGA的特点,而作者的意图是第一个实践项目如何能帮助读者很快地完美实现FPGA的全过程。在实现74LS00的与非门的项目中,读者不用再考虑功能需求是什么,因为它太明白了,而是把注意力集中到了如何实现的过程中了。

本书第5章的项目部分是马慧敏老师参与开发的,本书的很多内容是作者两人在西安交通大学城市学院计算机系里教学实践中获得的。在此感谢西安交大城市学院的系领导和院领导,以及陆丽娜教授对FPGA教学的支持和帮助。感谢上过FPGA课的同学们的支持和鼓励,他们中有些人在工作岗位上,FPGA应用水平已经超过了作者,学生超过了自己正是作者最感欣慰的事件。张喆同学在FPGA开发方面的意见,曾给过作者一些启发,这里对他也表深深谢意。写作过程中得到家人的鼓励和热情的支持,藉此对我的妻子和孩子们表示深情的谢意。还要感谢西安交通大学出版社的叶涛老师,对本书的道义上的支持和实质上的帮助。本书有一部分内容直接取自李舫先生的著作,对此作者表示深深的谢意,祝他在远方他国幸福。

作者这里还要特别感谢谢凯年博士先生。他作为赛灵思公司的大学计划中国区经理,于2011年6月帮助和支持在西安交通大学城市学院成立了FPGA联合实验室。正是在这个大学计划和实验室的支持下,作者开设了FPGA的课程。根据这些年的课程教学经验,才成就了本书的基本素材。本书的提出和写成得到了谢凯年博士的支持和指导,并且在百忙中欣然为本书撰写序言。在这篇高水平的序言中,谢博士介绍了可编程芯片的最新发展趋势,预言了可编程芯片对后计算机时代的贡献,为我们展示了大数据IT的解决方案,并大力支持采用MOOC方式将可编程芯片推向业界。这篇序言值得与可编程芯片相关的工程师和有兴趣的学习者一读。

本书中技术上一定有种种谬误、纰漏、不优化之处,观点上一定有狭隘、偏见之处,望读者君子给予指出、指正、批判。对此,作者抱着诚意向你们致敬,并表衷心谢意。书中的电路板是作者制作过和测试过的,还剩余一些,可提供给读者。书中第5章的全部源代码电子版,可提供给读者。

购买了本书的读者可以和作者联系,作者提供书中硬件设计资料和全部源代码。作者电话:13709181375,E-Mail:yhli@mail.xjtu.edu.cn,qq:630972994,qq群:290863896,微信号:liyuhua686994。

李裕华

2014.08

目 录

序

前 言

第 1 章	FPGA 概述	(1)
1.1	什么是 FPGA	(1)
1.2	FPGA 与 32 位处理器的比较	(2)
1.3	FPGA 和 ASIC 的关系	(3)
1.4	FPGA 的发展	(3)
1.5	代码语言 VHDL 和开发工具 ISE	(6)
1.6	FPGA 的应用	(7)
第 2 章	XC3S500E 和 XCF04S 的基本结构	(9)
2.1	XC3S500E 的主要特点	(9)
2.2	内部结构总论	(10)
2.3	配置操作	(11)
2.4	芯片封装标记	(12)
2.5	输入输出块(IOB)	(12)
2.5.1	IOB 内部结构	(12)
2.5.2	输入延时功能	(14)
2.5.3	存储单元功能	(15)
2.5.4	倍速数据传输	(16)
2.5.5	支持多种信号标准	(17)
2.5.6	片内差分端电阻(On_Chip Differential Termination)	(19)
2.5.7	上拉电阻和下拉电阻	(20)
2.5.8	边沿切换速率(Slew Rate)控制和驱动强度(Drive Strength)控制	(20)
2.5.9	IOB 的区(Bank)划分和规则	(20)
2.5.10	IOB 的供电	(21)
2.5.11	上电和配置过程时 I/O 引脚行为	(21)
2.5.12	JTAG 边界扫描功能	(22)
2.6	可配置逻辑块(CLB)和切片(Slice)资源	(22)
2.6.1	CLB 总论	(22)
2.6.2	切片总论	(23)
2.7	块 RAM(Block RAM)	(31)
2.8	专用乘法器(Dedicated Multipliers)	(39)
2.9	数字时钟管理器(DCM,Digital Clock Manager)	(42)
2.9.1	总论	(42)

2.9.2	延锁环(Delay-Locked Loop, DLL)	(43)
2.9.3	数字频率合成器(Digital Frequency Synthesizer,DFS)	(49)
2.9.4	相移器(Phase Shifter,PS)	(50)
2.9.5	状态逻辑(Status Logic)	(52)
2.10	时钟基本结构	(53)
2.11	芯片内功能单元互连(Interconnect)	(57)
2.12	配置(Coufiguration)	(59)
2.12.1	配置过程	(59)
2.12.2	配置数据大小	(61)
2.12.3	配置时钟信号 CCLK 的设计要点	(61)
2.12.4	HSWAP,M[2:0]和 VS[2:0]引脚线路设计要点	(61)
2.12.5	主机串口模式(Master Serial Mode)	(62)
2.12.6	其它的主机配置模式	(65)
2.12.7	从机串口模式(Slave Seral Mode)	(65)
2.12.8	JTAG 模式	(68)
2.12.9	配置过程总结	(69)
2.13	Spartan-3E 系列 FPGA 芯片引脚说明	(76)
2.13.1	引脚类型	(76)
2.13.2	三种封装的引脚	(77)
2.14	Platform Flash PROM	(81)
2.14.1	Flash PROM 的特点和对 FPGA 的支持	(81)
2.14.2	Flash PROM 内部结构	(82)
2.14.3	PROM 参与 FPGA 配置	(83)
2.14.4	PROM 的在系统烧写(In-system Programming)	(83)
2.14.5	PROM 的指标参数和引脚	(84)
第 3 章	核板和外围模块硬件设计	(86)
3.1	最小系统和外围器件	(86)
3.2	基于 FPGA 的电子产品分类	(86)
3.3	硬件系统的需求分析和项目方案	(87)
3.4	线路板集成设计工具 Protel DXP 简介	(89)
3.5	核板原理图设计	(91)
3.6	核板 PCB 设计	(141)
3.7	模块 Bank1_Md1(液晶和 8xLED 模块)设计	(161)
3.7.1	接口模块 Bank1_Md1 设计	(161)
3.7.2	8xLED 模块设计	(164)
3.7.3	RS232 通信模块设计	(165)
3.8	模块 Bank0_Md1(单数码管模块)设计	(167)
3.9	模块 Bank1_Md2(4 连数码管模块)设计	(169)
3.10	并口 JTAG 下载器设计	(172)

3.11 模块 Bank2_Mdl(输入模块)设计	(176)
第4章 开发环境 ISE 和 VHDL 语言	(178)
4.1 开发环境 ISE	(178)
4.2 ISE 安装过程	(178)
4.2.1 注册	(178)
4.2.2 ISE 软件下载	(183)
4.2.3 ISE 安装	(189)
4.3 ISE 操作全过程	(199)
4.3.1 ISE 界面介绍	(199)
4.3.2 新项目(New Project)建立	(201)
4.3.3 产生源代码	(203)
4.3.4 源代码综合(Synthesize)	(208)
4.3.5 波形仿真(Simulate)	(211)
4.3.6 产生用户约束文件	(215)
4.3.7 执行设计(Implement Design)	(217)
4.3.8 产生烧写文件(Generate Programming File)	(218)
4.3.9 产生 PROM 文件	(218)
4.3.10 烧写	(224)
4.3.11 运行	(230)
4.4 ISE 各功能模块	(231)
4.4.1 ISE 全过程总结	(231)
4.4.2 参考已有的源代码和约束文件建立新项目	(231)
4.4.3 源代码综合(Synthesize-XST)	(231)
4.4.4 执行设计(Implement Design)	(232)
4.4.5 产生烧写文件(Generate Programming File)	(232)
4.4.6 配置目标器件 (Configure Target Device)	(233)
4.5 VHDL 语言	(233)
4.5.1 VHDL 源代码结构形式	(233)
4.5.2 VHDL 基本定义	(238)
4.5.3 VHDL 语句	(243)
4.5.4 元件模块(Component)	(247)
第5章 项目软件开发和源代码解释	(250)
5.1 简单逻辑门电路	(250)
5.1.1 双输入与非门电路 test00_NAND	(250)
5.1.2 解码器电路 test138_deCoder_1	(254)
5.2 简单时序电路	(257)
5.2.1 LED 闪烁电路 test0_LED	(257)
5.2.2 LED 跑马灯电路 test8LED_HorseRun	(259)

5.2.3	带锁存解码器电路 test138_deCoder_2	(263)
5.2.4	同步解码器电路 test138_deCoder_3	(267)
5.2.5	串入并出同步移位寄存器电路 test164_SIPO	(271)
5.2.6	单数码管电路 testLEDNumber_1	(276)
5.2.7	按键控制的单数码管电路 testLEDNumber_2	(279)
5.3	I/O 口电路	(283)
5.3.1	4 连数码管计数器 NumLEDx4_1	(283)
5.3.2	4 连数码管精密计时器 NumLEDx4_2	(290)
5.3.3	4 连数码管倒计时器 NumLEDx4_3	(296)
5.3.4	4 连数码管定时器 NumLEDx4_4	(297)
5.4	通信电路	(307)
5.4.1	字符液晶显示 LCD1602_disp	(307)
5.4.2	UART 通信发送 UART_transmitter	(315)
5.4.3	UART 通信接收发送 UART_tranceiver	(321)
5.4.4	蓝牙通信 BlueTooth_1	(330)
5.4.5	基于 PWM 的 LED 亮度控制 PWM_2xLED	(345)
5.4.6	双舵机控制 PWM_2xMiniServer	(350)
5.4.7	步进电机控制 Step_Motor1	(358)
5.4.8	带设置的步进电机控制 Step_Motor2	(363)
5.5	DCM 开发	(375)
5.5.1	数字时钟管理器实例 DCM_Test	(375)
5.6	其它项目	(388)
5.6.1	直流电机控制	(388)
5.6.2	继电器控制	(389)
5.6.3	单片机和 FPGA 双机系统	(390)
参考文献		(392)

第 1 章 FPGA 概述

1.1 什么是 FPGA

什么是 FPGA? 下面逐步深入地回答这个问题。

1)FPGA(Field Programmable Gate Array)的全称是“现场可编程门阵列”。它实际上是一片大规模集成电路芯片。本书目的就是学习和应用这片芯片。

2)“现场可编程”是什么意思? 这里有两个单词,“现场”和“可编程”。现场是指 FPGA 运行的工业现场。FPGA 芯片焊在电路板上,电路板安装在设备里,设备在工业现场运行。可编程是指对 FPGA 芯片可以反复修改芯片内部电路。现场可编程就是在工业现场可以直接对芯片内部进行修改。这里的编程是 programming,不是指编程序,而是指下载(也可以称烧写,配置),就是把工程师新做好的设计数据(设置数据)重新烧写到 FPGA 芯片里。这样的结果就是改变了 FPGA 内部的电路。现场可编程,是指在芯片运行的现场就可烧写芯片。要可以在现场做,表明芯片烧写的过程和所使用的工具很简单,很容易操作,而且外部系统可以不断电,在以后学习到第 5 章时就可以体验到这一点。烧写的过程只需要几秒钟时间。烧写的设备只需要一台电脑和一个小型的下载器(烧写器)。

“门阵列”是什么? 门是指 4 类逻辑门:与门、或门、非门和触发器类的时序电路。阵列是指很多,而且排列整齐。“现场可编程门阵列”就是可在应用现场对芯片内的逻辑门电路门修改烧写的逻辑电路。这个阵列有多大,有多少逻辑门? 本书要学习的这片 FPGA(型号为 XC3S500E),面积仅为 $32 \times 32 \text{ mm}^2$,其中却集成了 50 万个逻辑门。相比之下,过去熟悉的,最常用的经典芯片 74LS00,却只有 4 个与门加 4 个非门。

3)初步了解了现场和门阵列以后,再深刻认识一下“可编程”(或可再烧写)的意义,这是 FPGA 的精髓所在。FPGA 相当于在用户面前整齐地码放着几十万门逻辑门,这是一大笔丰富的资源。由用户自己来把这些逻辑门在芯片内连接起来,达到用户需要的目的。这个逻辑门的连接可以随时做,经常不断地做,这就是可编程。可以浅显地理解,FPGA 芯片内的逻辑门都是断开的,由用户把需要的、按用户要求的方式烧焊在一起。或者反过来理解,本来这些逻辑门都是连接在一起的,由用户按他自己的要求将某些连接烧断,形成可用的电路。这就是可编程(再烧写)的意义。这样的现场再烧写功能带来的最大的好处在于,可以很容易地修正电路的错误,可以当场升级改版。

大多数现代处理器,例如单片机,32 位嵌入式系统等,都有可再烧写的功能,因为很普遍了,所以对它们都已经不再强调“programmable”,认为是理所当然的。但是 FPGA 的可再烧写和处理器的可编程本质上是完全不同的。处理器再烧写的是程序代码,它写入在 Flash 那样的存储器里的,类似于我们手机的 app 下载,这些程序代码是可以再读出来的二进制的数。FPGA 的可再烧写功能是对芯片内的逻辑门进行再连接,本质上不是写到存储器里的代码,也是不可读出的。

4)FPGA 技术到底是属于硬件范畴还是软件范畴? FPGA 提供了大量数字电路单元供用户自由应用。其中除了大量逻辑门以外,还有一些其它的复杂的硬核资源,所有这些资源都属于硬件。因此可以这样认为,FPGA 的应用开发属于硬件范畴。但是这么复杂和丰富的硬件资源由用户使用,会让用户不知所措。因而 FPGA 制造商和国际上的相关组织为 FPGA 的开发提供了一些规范的开发手段,使得复杂的资源开发变得简单容易,使用户不必关心 FPGA 芯片里具体的连接线路,自动地完成用户的需求。这样的开发工具、环境,要求 FPGA 开发工程师的工作在电脑上进行。开发工程师将较少地关注具体的电路图,更多地关注开发代码怎么写。这又类似于处理器领域的软件工程师工作,所以 FPGA 的开发也被认为属于软件范畴。总之,FPGA 的开发,软件上并没有像处理器领域的软件开发那么复杂,硬件上又没有真实线路设计制作那么繁琐。所以,FPGA 的开发是软硬兼施的。

1.2 FPGA 与 32 位处理器的比较

32 位处理器是当前发展最快的技术之一,它也是一片芯片,称为“嵌入式系统”。手机中的主芯片就是嵌入式系统,可以想象嵌入式系统当前的发展势头有多快多广。FPGA 的市场规模还无法和嵌入式系统相比,FPGA 作为一项高速发展的新技术,必定有其独特的功能和资源是嵌入式系统所不足的和没有的。这里先说明二者的技术行为的不同,再对二者的功能边界给予说明。

32 位处理器,或嵌入式系统都是计算机范畴的,都符合上世纪初计算机之父冯·诺依曼提出的数字计算机思想。体系结构中有作为核心的 CPU,有存储器,程序指令和数据保存在存储器里。运行方式是,指令从存储器里读出,由 CPU 来执行。指令的执行就是对数据加工处理。要注意的是,CPU 是一条指令跟着一条指令执行的。所以计算机领域采用 program 这个词,即“程序”。这个词在计算机发明之前的意思是“节目单”,也就是顺序执行的一个计划。所以计算机的运行行为是时间序的。嵌入式系统领域的软件工程师的工作是编程序,让 CPU 来顺序执行。

FPGA 运行行为是不一样的,它是将大量硬件单元组织在一起,协同起作用,这些硬件单元是一瞬间同时运行的。FPGA 领域不用“程序”这个词,而用“配置”(configuration)一词。FPGA 运行行为是空间状态的。FPGA 工程师的工作是编排配置,让大量片内硬件资源合理地连接起来,协调地运行。好的 FPGA 工程师要完成的是优化的排兵布阵工作。

嵌入式系统和 FPGA 由于运行的行为不同,表现的特性和能力也各不一样。由于 CPU 的数据处理能力强大,发展历史长,并且配套软件完备,使嵌入式系统运算能力大大优于 FPGA。尤其是安装了操作系统的嵌入式系统——例如智能手机,它的多媒体处理能力,汉字显示能力等方面显示了强劲的势头。没有安装操作系统的嵌入式系统,在复杂科学计算方面和控制方面也具有极大的优势。

FPGA 的优势在哪里呢?

(1)它是属于空间状态的,所以它的最强大优势是它的并行运行能力。没有 CPU 的统一调度和管理,各个硬件单元各自为核心地运行。有一个快速的系统时钟统一全体的硬件单元的运行节奏(同步性),以便使各个硬件单元之间的信号准确地传递和作用。例如 30 个不同的变化的 PWM 信号,不间断(non-stop)地由芯片引脚输出,在嵌入式系统上实现很困难,在 FPGA 上则没有问题,这就是 FPGA 的并行运行的能力。在嵌入式系统中,因为 30 个变化的

PWM 的产生要动用 CPU, CPU 在处理某一个 PWM 信号时, 另外的 PWM 信号就无法保持。这是嵌入式系统的 CPU 分时处理的缺点。

(2)FPGA 的另一个优势是运行速度快于嵌入式系统。这符合一般认识, 硬件运行是直接的电子运动, 而嵌入式系统里, CPU 处理是有节拍的, 是按时间序的。所以同样的硅片工艺制造的芯片, FPGA 运行行为本质决定了它跑得比嵌入式系统快。

(3)FPGA 的 IO 接口多于同等级的嵌入式系统。这也是因为同样容量的硅片中, 少了很多 CPU 有关的单元, 使得能有更多的空间留给 IO 接口。

当然, 很多任务用 FPGA 或嵌入式系统都是可以完成得很好的。然而有些任务不是二者都可以完成的。最新的发展, 就是二者往一起靠。Xilinx 公司 2011 年推出的新一代芯片 ZYNQ-7000 系列是 FPGA 和 32 位嵌入式系统的结合(集成了双核的 Cortex-A9, 800 MHz)。这个系列已经很难判断是 FPGA 硬核的基础上集成了嵌入式系统, 还是嵌入式系统架构上集成了 FPGA。当然其性能可以想象是威力无比的。

1.3 FPGA 和 ASIC 的关系

ASIC 全称是 Application Specific Integrated Circuit, 即专用集成电路芯片。对于特定的任务, 有一定的批量, 不采用通用芯片搭建线路, 而制作专用集成电路芯片, 从成本、面积、线路板生产等方面考虑是合理的。这类非通用的芯片称为 ASIC。ASIC 的特点是性能好、成本低(由于批量)、保密性好。缺点是开发成本高, 不灵活。一旦系统要求升级, 芯片的修改是不可能的。如果用 FPGA 替代 ASIC, 可能是个好的设计思想。FPGA 能克服 ASIC 的固有缺点。现场修改和升级正是 FPGA 的长项。相对于 ASIC, FPGA 被称为“半定制”(Semi-product)的, 就是说, 芯片一部分是 FPGA 公司做好了, 还有一部分留给用户自己完成。用户针对应用需求在 FPGA 上进行设计并烧写, 使得 FPGA 转变为 ASIC。用半定制的 FPGA 取代 ASIC 是现代产品开发的趋势之一。

有些情况下, FPGA 替代 ASIC 成本上不可行。例如 LED 圣诞节彩灯, 它的 ASIC 成本极低, 而 FPGA 的成本比其高了十几倍。圣诞节彩灯的批量极大, 这种情形下用 FPGA 替代 ASIC 是不行的。所以 FPGA 替代 ASIC 也是要看具体情况的, 主要还是取决于经济效益。但是用 FPGA 开发板来辅助 ASIC 的开发却是个十分理想的手段。例如用 FPGA 开发板来调试圣诞节彩灯的各种花式, 这里包括了空间状态和时间参数。调试完成后, 从 FPGA 中提取用到的资源和电路原理图, 然后按照这个原理图再设计它的 ASIC。

1.4 FPGA 的发展

FPGA 作为一种可以再烧写的硬件电路, 这种功能思想是起源于早期的可再烧写的逻辑门芯片。上世纪 80 年代以后, 曾经占据了电子电路市场很大份额的是 GAL (Gate Array Logic, 门阵列逻辑) 和 PAL (Programmable Array Logic, 可编程阵列逻辑) 芯片。在双列直插元件的年代里, 它们的面积较小, 设计后可以将芯片里的微熔丝烧断, 变成永久性的硬件电路。由于它们的使用, 使得硬件线路保密性好, 集成度当时相对比较高, 又可重复烧写, 当时很受硬件开发工程师青睐。

后来出现的 PLD(Programmable Logic Device, 可编程逻辑器件)芯片, 其片内逻辑门的集成度更高。直到上世纪 90 年代, 新一代的可编程器件 CPLD(Complex Programmable Logic Device, 复杂的可编程逻辑器件)和 FPGA 开发出来, 在可编程器件领域它们变成了主流, 这个势头一直保持到现在。现代的 CPLD 和 FPGA 有很多共同点, 性能上也不相上下。CPLD 是“粗”粒结构的可编程逻辑器件, 它的基本单元比 FPGA 大, 称为宏单元。FPGA 的片内硬件资源更细小, 更底层。由于结构上的差异, 使得这两类器件适合的应用场合不相同, 但还是有很多任务这两类器件是可以同样完成的。相对来说, CPLD 适合算法和组合逻辑, FPGA 更适合时序逻辑。如果任务中要求特别多触发器, 则 FPGA 更合适。如果任务中乘积项多的话, 用 CPLD 更合适。总的来说, FPGA 的集成度比 CPLD 高, 能实现更复杂的逻辑, 但 CPLD 的开发比 FPGA 要简单。CPLD 还有一个特点是 FPGA 不具备的, CPLD 芯片中集成了 Flash 存储器, 它烧写的代码是不会丢失的。FPGA 中没有 Flash 存储器, 要在外围专门配置 Flash 存储器。

早在 GAL 芯片出现以前, 存储器芯片已经生产出来了。实际上存储器也是一种可编程逻辑器件。无论是 PROM(可编程只读存储器), EPROM(可擦除只读存储器), EEPROM(电可擦除只读存储器)都是可以重烧写的存储器。这是可编程的特点。但是为什么它是逻辑器件? 这可以从两个方面来理解: (1)任何组合逻辑都可以用存储器方式解决; (2)存储器本身是由基本逻辑电路组成的。

随意举例, 一个组合逻辑如图 1-1 所示。它由与门、或门、非门, 3 个逻辑门组成。它有 3 个输入(A, B, C), 2 个输出(Q1, Q0)。通过设置输入很容易得到这个组合逻辑的真值表(表 1-1)。现在抛开图 1-1 原理图, 将表 1-1 的真值表理解成 ROM 数据表。C、B、A 理解成地址线 A2、A1、A0; 输出 Q1、Q0 理解成数据 D1、D0, 由此形成了一个 ROM 数据表(表 1-2)。ROM 数据表的地址有 3 位, 数据只有 2 位。如果将这个数据写入到一片 EEPROM(例如 7864A 存储器, 8 位数据, 12 位地址)中, 那么这片 7864A 就能完成图 1-1 的组合逻辑。从硬件上连线, 让 7864A 的最低 3 位地址线 A2、A1、A0 认为是图 1-1 的输入线 C、B、A。让 7864A 的最低 2 位数据线 D1、D0 认为是图 1-1 电路的输出线 Q1、Q0。当把数据表 1-2 烧写到 7864A 中, 就实现了用存储器替代组合逻辑电路的变换。3 根地址线有 8 个地址, 每个地址对应一个数据输出, 这个输出的数据值和数据表 1-2 是一致的, 因为存储器就是按照这个数据表烧写的。同时这样的输出值也同真值表 1-1 一致。因为逻辑电路(图 1-1)和存储器的输入输出关系是完全一致的, 说明了存储器就是一种可编程逻辑器件, 经过烧写它可以演变成各种组合逻辑电路。编写一个数据表就相当于编写一个可编程逻辑器件的烧写代码。7864A 这样的 EEPROM 有 12 位地址, 8 位数据。它可以被认为是具有 12 个输入 8 个输出的组合逻辑电路的可编程逻辑器件。这是上面提到的第(1)个方面的理解。

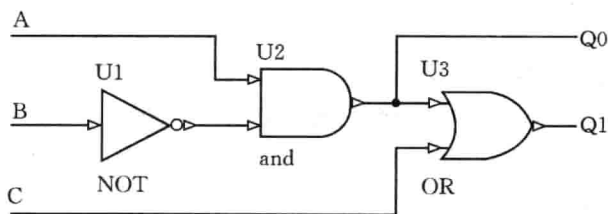


图 1-1 一个组合逻辑原理图

表 1-1 真值表

C	B	A	Q0	Q1
1	1	1	0	1
1	1	0	0	1
1	0	1	1	1
1	0	0	0	1
0	1	1	0	0
0	1	0	0	0
0	0	1	1	1
0	0	0	0	0

表 1-2 ROM 数据表

地址	数据
7	2
6	2
5	3
4	2
3	0
2	0
1	3
0	0

要说明第(2)个方面的理解,可以将 EEPROM 这样的存储器设计成图 1-2 所示的基本逻辑门电路。对它进行烧写,就可以将数据保存,它还能演变成一个可编程逻辑器件。以下对图 1-2 分析,图 1-2 是一个具有 3 位地址,1 位数据的存储器原理图,它可以用来解释 EEPROM 的结构。其中的 S 形的元件(F1 ~ F8)是可恢复熔丝。熔丝打×的符号表示烧写时熔丝被烧断。图 1-2 的烧写是按照表 1-2 的数据完成的。熔丝烧断,形成开路。形成开路的线由于下拉电阻都是低电平。如果对 EEPROM 芯片擦洗,就是恢复芯片的所有已烧断的熔丝。图中 A2, A1, A0 是 3 位地址线,是输入。Q0 是输出数据线。图中有 8 个 3 输入与门(U4~U11),和 3 个非门(U1~U3)。这部分逻辑门和按图中的连线实际上组成了一个 3-8 选 1 的电路单元。举例说,当 A2, A1, A0 都是 1 时,3 输入与门 U4 的输入是 3 个 1,它的输出将是 1。其余 7 个 3 输入与门输出都是 0。这其余 7 个与门的 3 输入中至少会有一个是非门连接着的,由于 A2, A1, A0 输入都是 1,所以 3 个非门的输出都是 0。例如 U5,它的第 3 个输入是和 A0 的非门 U3 连接着。这时非门的输出是 0,所以 U5 的输出是 0。图 1-2 的左边部分构成了一个 3-8 选一的电路,3 根地址线无论输入什么值,这 8 个 3 输入与门(U4~U11)中一定有一个的输出是 1。这 8 个与门的输出串联着 8 个可恢复熔丝(F1~F8)。然后各自连接一个下拉电阻,再作为输入连接到一个 8 输入或门 U12 上。8 输入或门的输入只要有一个或一个以上的 1,则输出就是 1。只有在 8 个输入都是 0 时输出才是 0。从图 1-2 的熔丝烧断的情况来看,有 2 个输入没有被烧断,它们是 F3 和 F7。所以只有在 3-8 选一的这 2 根线被选上时,8 输入或门的输出 Q0 才是 1,其余的选通 Q0 是 0。F3 和 F7 的选通地址是 101 和 001。001 对应的地址是 A2=0, A1=0, A0=1。101 对应地址是 A2=1, A1=0, A0=1。这个结果和表 1-2 的值是一样的,也和表 1-1 的值是一样的。这表示这个烧写是正确的。烧断了熔丝的 EEPROM 可以长久地保持这个结构,从而保持这个数据。从图 1-2 还能看出,没有烧写过的新的 ROM,或者擦洗过的 ROM,它的熔丝全部都恢复,这时无无论什么样的选通,8 输入或门总有一个输入是 1,所以数据输出总是 1。这符合 EEPROM 的特性。

这个例子说明了三个问题:(1)EEPROM 本身就是可编程逻辑器件;(2)各种变化的组合逻辑,可以用类似图 1-2 这样的统一的、万能的线路来解决;(3)复杂组合逻辑电路可以简化地理解为是数据存储器,电路的输入线是地址线,电路的输出线是数据线,组合逻辑电路的变化演变成存储器里的数据的变化。