

Verilog HDL

语言及数字系统设计

李景华 杜玉远 主编

VHDL HDL
YUYAN JI SHUZI
XITONG SHEJI



国防工业出版社

National Defense Industry Press

Verilog HDL 语言及 数字系统设计

李景华 杜玉远 主编

国防工业出版社

· 北京 ·

内 容 简 介

本书在介绍 EDA 技术相关内容的基础上,讲述了 Verilog HDL 语言及采用 Verilog HDL 语言设计并实现数字逻辑电路和数字系统的方法。内容包括典型 CPLD 和 FPGA 器件的结构特点、Verilog HDL 语言、典型 Verilog HDL 程序设计、典型数字系统设计、RISC 处理器设计方法和 QuartusII 10.1 开发系统。全书以 Verilog 语言为基础,结合典型的设计例程和典型的数字系统,由浅入深地讲述了 Verilog 语言的设计方法。通过全面细致地剖析 RISC 处理器的过程,讲述了复杂数字系统的设计方法。

本书可作为高等学校电子信息工程、电子科学与技术、通信工程,计算机科学与技术、电子信息工程、自动控制等专业的本科生和相关专业研究生的教科书或参考教材,也可作为电子系统的工程设计人员和其他相关人员的自学参考书。

本书还可以为授课教师免费提供结合本书内容的、图文并茂的教学课件。

图书在版编目(CIP)数据

Verilog HDL 语言及数字系统设计/李景华,杜玉远

主编. —北京:国防工业出版社,2012.2

ISBN 978-7-118-07903-6

I. ①V... II. ①李... ②杜... III. ①VHDL 语言 - 程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字(2012)第 013101 号

※

国防工业出版社 出版发行

(北京市海淀区紫竹院南路 23 号 邮政编码 100048)

北京奥鑫印刷厂印刷

新华书店经售

*

开本 787×1092 1/16 印张 16¼ 字数 373 千字

2012 年 2 月第 1 版第 1 次印刷 印数 1—5000 册 定价 29.00 元

(本书如有印装错误,我社负责调换)

国防书店:(010)88540777

发行邮购:(010)88540776

发行传真:(010)88540755

发行业务:(010)88540717

前 言

随着电子技术的飞速发展,可编程逻辑器件及其设计技术也快速发展。20 世纪末期,集成电路的制造技术处于深亚微米阶段,其特征尺寸为 $0.35\mu\text{m} \sim 0.18\mu\text{m}$ 。经历了短短的几年时间,集成电路的制造技术已经发展到 $90\text{nm} \sim 65\text{nm}$ 的水平。这就意味着单片可编程逻辑器件可以集成几千万个 PLD 门。

无论是可编程逻辑器件的发展,还是集成电路的技术进步,都促进了从事可编程逻辑器件设计和 IC 设计方面人才的需求增长。

Verilog HDL 作为可编程逻辑器件和 IC 设计的共同适用的开发编程语言,已经成为硬件开发人员必须学习和掌握的设计语言之一。

编者结合自己多年的教学和科研经验,由浅入深地讲述了 Verilog HDL 的语法规格和设计规范,并结合丰富的设计实例,由简至繁地介绍了数字系统设计的方式方法。编者希望本书能够对广大读者学习 Verilog HDL 语言提供一定帮助。在书中第 5 章详细介绍了设计 RISC 处理器的全部过程,希望为进行复杂数字系统设计的广大读者提供一定的借鉴。

为了便于学习和掌握 Verilog HDL,书中在第 6 章以 step by step 方式,详细地介绍了 QuartusII 软件的使用方法。

本书可作为高等院校电子信息工程、计算机应用、通信工程、微电子、自动控制类专业的本科生和研究生学习数字系统设计课的教材,也可以作为电子系统设计工程师的技术参考书。

本书由李景华、杜玉远、李汇明编写。其中第 1 章~第 3 章由李汇明和李景华编写,第 4 章~第 6 章由杜玉远编写。全书由李景华和杜玉远负责内容编排、定稿与修改。书中第 4 章的设计例程由沈鸿媛老师协助编写。

本书在编写过程中参考了相关专家和学者的著作,同时也引用了一些公司的相关数据手册,在此深表感谢。由于编者水平有限,在书中可能存在不足或错误,恳请读者批评指正。

所提意见或建议可直接发到编者邮箱 duyuyuan@ise.neu.edu.cn

编 者

2011 年 11 月

目 录

第 1 章 可编程逻辑器件和 EDA 技术	1
1.1 EDA 技术的主要特征	1
1.2 EDA 技术的设计方法	3
1.3 可编程逻辑器件简介	4
1.3.1 从 ASIC 到 FPGA/CPLD	4
1.3.2 CPLD 器件	6
1.3.3 FPGA 器件	12
1.4 可编程逻辑器件的设计	24
1.4.1 可编程逻辑器件的设计流程	24
1.4.2 Xilinx 公司的 ISE 开发工具概述	26
1.4.3 Altera 公司的 Quartus II 开发工具概述	27
1.4.4 Lattice 公司的 IspLever 开发工具概述	28
1.5 可编程逻辑器件的选型	29
1.5.1 选择 CPLD 的方法	29
1.5.2 选择 FPGA 的方法	30
1.6 IP 核简介	31
1.7 EDA 技术的发展趋势	33
1.7.1 可编程逻辑器件的发展趋势	33
1.7.2 EDA 软件开发工具的发展趋势	33
1.7.3 设计输入方式的发展趋势	34
第 2 章 Verilog HDL	36
2.1 Verilog HDL 基本概念	36
2.1.1 Verilog HDL 简介	36
2.1.2 Verilog HDL 模块的概念及结构	36
2.1.3 Verilog 模块的测试与验证	39
2.2 Verilog HDL 的基本要素	42
2.2.1 数据类型	42

2.2.2	运算符	44
2.2.3	编译指令	48
2.3	行为描述	50
2.3.1	行为描述的基本概念和结构	50
2.3.2	结构说明语句	52
2.3.3	过程赋值语句	53
2.3.4	条件语句和循环语句	54
2.3.5	任务和函数	58
2.4	数据流描述	60
2.4.1	数据流描述的基本概念和结构	60
2.4.2	连续赋值语句	61
2.5	结构描述	62
2.5.1	结构描述的概念及方式	62
2.5.2	开关级建模	62
2.5.3	门级建模	63
第3章	典型 Verilog HDL 的设计实例	65
3.1	组合逻辑电路的设计	65
3.1.1	逻辑门电路的设计	65
3.1.2	常用编码器的设计	67
3.1.3	常用译码器的设计	70
3.1.4	数据选择器的设计	72
3.1.5	数据分配器的设计	74
3.1.6	数值比较器的设计	75
3.1.7	算术运算单元电路的设计	77
3.2	时序逻辑电路的设计	79
3.2.1	常用触发器的设计	80
3.2.2	常用数码寄存器的设计	81
3.2.3	常用计数器的设计	84
3.3	有限状态机的设计	86
3.3.1	有限状态机的建模	86
3.3.2	状态编码	89
3.3.3	Mealy 型状态机的设计	91
3.3.4	Moore 型状态机的设计	94
3.4	存储器的设计	97

3.4.1	只读存储器的设计	97
3.4.2	随机存储器的设计	99
3.4.3	顺序存取存储器的设计	100
第 4 章	典型数字系统的分析与设计	102
4.1	数字系统概述	102
4.2	数码管动态显示扫描电路原理及设计	102
4.2.1	数码管动态显示扫描电路原理	102
4.2.2	采用 Verilog HDL 描述的动态显示扫描电路	104
4.3	乘法器的原理及设计	106
4.3.1	乘法器的工作原理	106
4.3.2	采用 Verilog HDL 描述的乘法器	108
4.4	除法器的原理及设计方法	110
4.4.1	除法器的工作原理	110
4.4.2	用 Verilog HDL 描述的除法器	112
4.5	简易 CPU 的工作原理及设计方法	115
4.5.1	简易 CPU 的工作原理	115
4.5.2	采用 Verilog HDL 描述的 ALU	118
4.6	交通信号灯控制器的原理及设计	123
4.6.1	交通信号灯控制器的原理	123
4.6.2	交通信号灯的 Verilog HDL 描述	125
4.7	数字频率计的原理及设计	130
4.7.1	数字频率计的原理	130
4.7.2	数字频率计的 Verilog HDL 描述	132
4.8	数字信号发生器的原理及设计	136
4.8.1	数字信号发生器的原理	136
4.8.2	数字信号发生器的 Verilog HDL 描述	138
第 5 章	RISC 处理器的设计	144
5.1	支持 RISC 处理器的器件	144
5.1.1	Cyclone 系列 FPGA 器件	144
5.1.2	Cyclone II 系列 FPGA 器件	148
5.1.3	Cyclone III 系列 FPGA 器件	155
5.1.4	Stratix II 系列 FPGA 器件	157
5.1.5	Stratix II GX 系列 FPGA 器件	162

5.2 RISC 处理器的原理	165
5.2.1 RISC 处理器的特性	165
5.2.2 ALU 的工作原理	166
5.2.3 寄存器组	167
5.2.4 总线接口单元	168
5.2.5 流水线结构	168
5.2.6 中断系统	169
5.2.7 存储器地址分配	170
5.2.8 RISC 处理器的总线时序	171
5.2.9 RISC 处理器引脚	173
5.3 RISC 处理器指令系统	174
5.3.1 RISC 处理器指令格式	174
5.3.2 RISC 处理器指令集	174
5.4 RISC 处理器的设计	175
5.4.1 RISC 处理器主程序	175
5.4.2 RISC 处理器 ALU 的设计	179
5.4.3 RISC 处理器寄存器组及总线接口的设计	185
5.4.4 RISC 处理器控制器的设计	194
第 6 章 Quartus II 10.1 开发系统	209
6.1 Quartus II 10.1 开发系统简介	209
6.1.1 Quartus II 10.1 开发系统的特性	209
6.1.2 Quartus II 10.1 开发系统的安装	210
6.1.3 Quartus II 10.1 开发系统的软件许可配置	213
6.1.4 Quartus II 10.1 开发系统的设计流程	215
6.2 设计输入	215
6.2.1 建立设计工程	216
6.2.2 原理图设计文件	219
6.2.3 VHDL 设计文件	222
6.2.4 设计约束文件	223
6.3 综合与编程	225
6.3.1 综合参数控制	225
6.3.2 RTL 查看器和状态机查看器	227
6.3.3 渐进式综合	228
6.3.4 多样化编程	234

6.4	设计仿真	235
6.4.1	仿真波形文件.....	236
6.4.2	仿真	238
6.5	SignalTap II 逻辑分析器	239
6.5.1	设置和运行 SignalTap II 逻辑分析器	239
6.5.2	渐进式编译使用 SignalTap II 逻辑分析器	242
6.5.3	分析 SignalTap II 数据	242
6.6	设计实例	242
6.6.1	建立设计工程.....	243
6.6.2	建立源文件	244
6.6.3	编译设计	246
6.6.4	引脚锁定	247
6.6.5	仿真设计	248
6.6.6	编程和配置	251
参考文献		252

第 1 章 可编程逻辑器件和 EDA 技术

EDA(电子设计自动化)技术是以计算机为工作平台,以融合了应用电子技术、计算机技术、智能化技术最新成果而研制成的电子 CAD 通用软件包为开发环境,以电子系统设计为应用方向的电子产品自动化设计过程。EDA 主要包含三方面的设计工作,即 IC 设计、电子电路设计和印制电路板(PCB)设计。本章主要介绍 EDA 技术的主要特征和发展方向,常用的复杂可编程逻辑器件 CPLD 和现场可编程门阵列 FPGA 的工作原理,并对常用的 EDA 工具和设计技术进行简单介绍。

1.1 EDA 技术的主要特征

20 世纪 90 年代后,电子系统已经从电路板级系统集成发展成为包括 ASIC、FPGA 和嵌入系统的多种模式,EDA 产业已经成为电子信息类产品的支柱产业。在过去几十年内,IC 设计方法经历了从手工设计、电路仿真、原理图输入和逻辑仿真、布局和布线到综合这几个阶段。近年来,微电子技术以惊人的速度发展,其工艺水平已达到纳米级,在一个芯片上可集成数百万乃至上千万个晶体管,工作速度可达到 Gb/s 的数量级,这为制造出规模更大、速度和信息容量更高的芯片系统提供了基础条件。集成电路设计技术的进步也对 EDA 技术提出了更高的要求,大大地促进了 EDA 技术的发展。

以高级语言描述、系统仿真和综合技术为特征的 EDA 技术,代表了当今电子设计技术的最新发展方向。采用 EDA 设计技术的基本流程是设计者按照“自上而下”的设计方法,对整个系统进行方案设计和功能划分。电子系统的关键电路一般用一片或几片专用集成电路(ASIC)实现,采用硬件描述语言(HDL)完成系统行为级设计,最后通过综合器和适配器生成最终的目标器件。这种高层次的电子设计方法,不仅极大地提高了系统的设计效率,而且使设计者摆脱了大量的辅助性工作,将精力集中于创造性的方案与概念的构思上。近年来的 EDA 技术主要有以下特点:

- (1) 采用行为级综合工具,设计层次由 RTL 级上升到了系统级;
- (2) 采用硬件描述语言描述大规模系统,使数字系统的描述进入抽象层次;
- (3) 采用 Floor Planning 技术,使得复杂 IC 的描述规范化,做到在逻辑综合早期设计阶段就考虑到物理设计的影响;
- (4) 将可测性电路结构做在可编程 ASIC 芯片上,开发了扫描插入、内建自测试、边界扫描(JTAG)等可测性设计工具,并已集成到 EDA 系统中;
- (5) 为带有嵌入式 CPU 核的 ASIC 设计提供软件、硬件协同设计工具。

针对当今 ASIC 的特点(规模大而复杂,数字与模拟电路并存,硬件与软件设计并存;产品上市速度要快),建立并行设计工程框架结构的集成化、系统化设计环境,将不同公

司的优秀工具集成为一个完整的 EDA 系统,各种 EDA 工具在该框架中可以并行使用。

以下是与 EDA 技术密切相关的几个概念。

1. “自上而下”的设计方法

电子设计的最初思路是选用标准集成电路“自底向上”地构造出一个新的系统,这样的设计方法不仅效率低、成本高,而且容易出错。

现代电子系统的设计理念是一种“自上而下”的设计方法,首先从系统设计入手,在顶层对系统进行功能的划分和结构的设计。这一级采用硬件描述语言对高层次的系统进行行为级描述,在系统一级进行验证。然后,用综合优化工具生成具体门电路的网络表,其对应的物理实现通常采用印制电路板或可编程专用集成电路。由于设计的主要仿真和调试过程是在高层次上完成的,从而有利于早期发现结构设计上的错误,避免设计工作的重复,同时减少了逻辑功能仿真的工作量,提高了设计的成功率。

2. ASIC

ASIC 是专用集成电路的简称。ASIC 是随着电子产品的复杂度日益提高应运而生的。一个电子系统若由数万个中小规模 IC 构成,则会产生体积大、功耗大、可靠性差等诸多问题。解决这一问题的有效方法就是采用 ASIC 芯片进行设计。ASIC 按照设计方法的不同可分为全定制 ASIC、半定制 ASIC 和可编程 ASIC(也称为可编程逻辑器件)。

全定制 ASIC 芯片是由设计者定义芯片上所有晶体管的几何图形和工艺规则,最后将设计结果交由 IC 厂家去进行掩模制造,做出产品。这种设计方法的优点是芯片可以获得最优的性能,即面积利用率高、速度快、功耗低,而缺点是开发周期长、费用高,只适合大批量产品开发。目前,模拟电路的设计通常采用该方法。

半定制 ASIC 芯片的版图设计方法分为门阵列设计法和标准单元设计法,这两种方法都是约束性的设计方法,其主要目的就是简化设计,以牺牲芯片性能为代价来缩短开发时间。目前,其在通信领域仍有较大的应用空间。

可编程 ASIC,又称可编程器件。自 20 世纪 70 年代以来,它经历了 PAL、GAL、CPLD、FPGA 几个发展阶段,其中,CPLD/FPGA 属高密度可编程逻辑器件,目前,其集成度已高达 200 万门/片~1000 万门/片。它将掩模 ASIC 集成度高的优点和可编程逻辑器件设计生产方便的特点结合在一起,特别适合于样品研制或小批量产品开发,使产品能以最快的速度上市,而当市场扩大时,它可以很容易地转由掩模 ASIC 实现,因此,开发风险也大为降低。所以,它已成为现代高层次电子设计方法的实现载体。

3. 硬件描述语言

硬件描述语言(Hardware Description Language, HDL)用软件编程的方式来描述电子系统的逻辑功能、电路结构和连接形式。与传统的门级描述方式相比,它更适合大规模电子系统的设计。早期的硬件描述语言,如 ABEL、HDL、AHDL,由不同的 EDA 厂商开发,互不兼容,而且不支持多层次设计,层次间的翻译工作要由人工完成。为了克服以上不足,1985 年,美国国防部正式推出了高速集成电路硬件描述语言 VHDL。1987 年,IEEE 采纳 VHDL 为硬件描述语言标准,命名为 IEEE STD 1076-1987,简称为 87 版。1993 年,进一步修订后,形成了更加完备的 IEEE STD 1076-1993 版本,简称为 93 版。最新的 VHDL 标准是 IEEE STD 1076-2001 版本,目前主流的设计工具可以很好地支持这些版本的 VHDL 设计。

4. EDA 系统框架结构

EDA 系统框架结构(Framework)是一套配置和使用 EDA 软件包的规范。目前,主要的 EDA 系统都建立了框架结构,如 Cadence 公司的 Design Framework、Mentor 公司的 Falcon Framework。这些框架结构都遵守国际 CFI 组织制定的统一技术标准。框架结构能将来自不同 EDA 厂商的工具软件进行优化组合,集成在一个易于管理的统一的环境之下,而且还支持任务之间、设计师之间以及整个产品开发过程中的信息传输与共享,是并行工程和“自上而下”设计方法的实现基础。

1.2 EDA 技术的设计方法

EDA 技术的每一次进步都引起了设计层次上的飞跃。物理级设计主要指 IC 版图设计,它一般由半导体厂家完成,对电子工程师没有太大的意义。电子工程师所关注的内容主要是电路级设计和系统级设计。

1. 电路级设计

电路级设计本质上是基于门级描述的单层次设计。电子工程师接受系统设计任务后,首先确定设计方案,并选择能实现该方案的合适元器件,然后根据具体的元器件设计电路原理图。接着进行第一次仿真,其中包括数字电路的逻辑模拟、故障分析,模拟电路的交直流分析、瞬态分析。在进行系统仿真时,必须要有元件模型库的支持,计算机上模拟的输入输出波形代替了实际电路调试中的信号源和示波器。这一层次仿真主要是检验设计方案在功能方面的正确性。

仿真通过后,根据原理图产生的电气连接网络表再进行 PCB 板的自动布局布线。在制作 PCB 板之前还可以进行 PCB 后分析,其中包括热分析、噪声及窜扰分析、电磁兼容分析、可靠性分析等,并可将分析后的结果参数反标回电路图,进行第二次仿真,也称为后仿真。后仿真主要是检验 PCB 板在实际工作环境中的可行性。

电路级设计的所有工作(包括设计输入、仿真和分析、设计修改等)都是在基本逻辑门这一层次上进行的。需要电子工程师全面地了解系统的功能特性和物理特性,从而将开发风险消灭在设计阶段。

2. 系统级设计

20 世纪 90 年代以来,电子信息类产品的开发明显呈现两个特点:一是产品复杂程度提高;二是产品上市时间限紧迫。为了适应这种市场的需求,一种高层次的电子设计方法,即系统级设计方法应运而生。

高层次的电子设计针对设计目标进行功能描述。设计者无须关心电路细节,而是将精力放在设计方案的制定和创新上,以高层次描述的形式输入计算机。高层次设计只是定义系统的行为特性,不涉及实现工艺,还可以在厂家综合库的支持下,利用综合优化工具将高层次描述转换为针对某种工艺优化的网络表,可轻易实现工艺优化和系统升级。

系统级设计的流程如下:

- (1) 系统工程师按照“自上而下”的方法进行系统划分。
- (2) 采用 VHDL 语言状态图等方式描述系统,并编译成标准的 VHDL 文件。
- (3) 进行代码级的功能仿真,检验系统功能设计的正确性。

(4) 对VHDL源代码进行综合优化处理,生成门级描述的网络表文件,这是将高层次描述转化为硬件电路的关键步骤。综合的过程要在相应的厂家综合库的支持下才能完成。

(5)利用产生的网络表文件进行适配前的时序仿真。

(6) 将综合后的网络表文件针对某一具体的目标器件进行适配,包括底层器件配置、逻辑分割、逻辑优化、布局布线。

(7) 根据适配后的仿真模型,进行适配后的时序仿真。由于已经得到所描述系统的实际硬件特性(如时延特性),所以,仿真结果能比较精确地预期实现所描述系统的未来芯片的实际性能。如果仿真结果达不到设计要求,就需要修改VHDL源代码或选择不同速度和品质的器件,直至满足设计要求为止。

(8) 将适配产生的器件编程文件通过编程器或下载电缆载入到目标芯片FPGA或CPLD中。

1.3 可编程逻辑器件简介

1.3.1 从 ASIC 到 FPGA/CPLD

1. ASIC

标准集成电路是指具有标准的芯片功能,可以在市场上购买到的通用器件。以下器件都属于标准集成电路。

- (1) 中央处理单元(CPU);
- (2) 存储器(DRAM、SRAM、ROM、EPROM、EEPROM);
- (3) 计算机主板上的南北桥芯片;
- (4) 显示卡上的绘图芯片;
- (5) 运算电路中的加法器、减法器、比较器、数据选择器等;
- (6) USB 收发器。

ASIC(Application Specific Integrated Circuit, 专用集成电路)是设计者根据设计需求所设计的在特殊场合使用的集成电路,是相对标准集成电路而言的,它保证了设计的安全性和系统设计的整合效率。

ASIC 是泛指面向专门用途或特定用户而设计制造的集成电路,是片上系统集成(SOC)的基础。以下器件都属于专用集成电路。

- (1) 全定制;
- (2) 半定制。
 - ① 门阵列;
 - ② 标准单元;
 - ③ 积木块。

ASIC 是 CPLD 和 FPGA 的先驱。这种集成电路的设计是为设计者提供在特殊场合所需的集成电路组件。设计一片 ASIC 芯片,并不需要具有专门的半导体物理或半导体制造工艺方面的知识。ASIC 的厂商会提供一个标准单元库,设计者依据这个单元库进行进一步设计。然后 ASIC 厂商会根据设计者的设计安排半导体芯片原料,生产掩膜,最终制

造出 ASIC 芯片。

ASIC 可由如下四种方式实现。

1) 门阵列

门阵列 ASIC 是由规则的晶体管排列成的行、列组成的，如图 1-1 所示。外围部分是 I/O 单元。这些单元包含了输入/输出缓冲器、有限的晶体管及压焊盘等。它在阵列内部是基本的逻辑单元或门电路。每个单元均由少量的晶体管组成。门阵列上所有的晶体管在初始时并不连接，布局设计软件根据给定的设计方案，会计算出哪些晶体管需要连接。ASIC 厂商提供了一些未连接的芯片，当设计完成后，厂商只需对参与连接的金属层进行照相掩模，并加到芯片上，就可以生产出自己的芯片。门阵列的优点是内部电路工作速度快、电路密度高(具有百万个门电路密度的门阵列，其时钟频率能达到数百兆赫)。在一个芯片上可集成多个功能块，对需求量较大的产品来说，成本低廉。门阵列的缺点是，ASIC 的厂商需要很长时间去制造和测试电路部件。另外，用户要预先承担大笔的费用，即非重复性工程(Non-Recurring Engineering, NRE) 费用，ASIC 生产厂商用这笔费用去启动整个 ASIC 的生产过程。如果这其中存在某一个错误，那么修改这个错误，并重新生产出新的 ASIC 芯片就需要花费很长的时间和昂贵的生产费用。

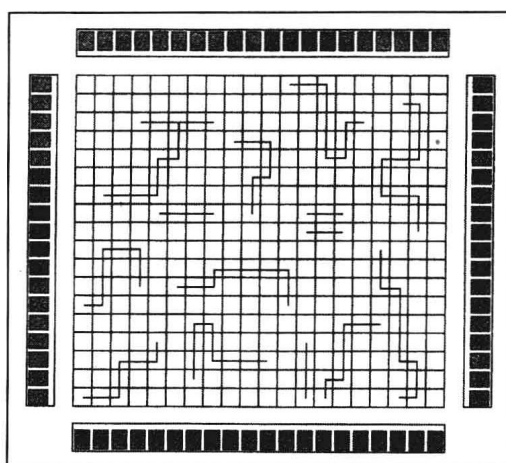


图 1-1 门阵列 ASIC 内部结构

2) 标准单元

这种设计方式是根据 ASIC 厂商提供的标准单元库，进行数字逻辑电路设计的。其优点是设计较为灵活；缺点是设计所需的掩模版层数较门阵列方式多(约 12 层以上)。标准单元设计方式可以使成本降低，但要付出较高的 NRE 费用，同时生产周期较长。

3) 积木式设计

积木式设计是综合了门阵列和标准单元的各自优点来实现器件设计的。即以单元的方式来建立逻辑电路中的不同功能模块，例如，需要改变某个模块的设计时，只需改变相应的模版即可，从而大大节省了设计风险和成本，提高了设计效率。

4) 全定制

该设计方式是按客户要求，以最佳的电路设计为目标，以期得到一个价格合理且性能优良的产品。其缺点是付出高昂的 NRE 费用和大量的设计人工费用以及较长的设计周期。

所以，通常只在关键电路的设计上使用。

用户可以根据所要开发产品的特点，选择某种适合的设计方式完成设计。

2. 可编程逻辑器件

可编程逻辑器件(Programmable Logic Device, PLD)经历了一个相当长的发展过程才达到如今的复杂程度，一个芯片可集成一个完整的系统，即实现系统在片 SOC。

第一代可编程逻辑器件是昂贵的掩模 ROM 的替代品——可编程只读存储器(PROM)，PROM 从一次性编程发展到可重复编程(EPROM、E²PROM、FLASH)。由于 PROM 速度慢，加上工艺的原因，所以，很难将 PROM 集成到逻辑电路中去。

可编程逻辑阵列 PLA 解决了 PROM 的问题，它由可编程“与”阵列和与之相连的“或”阵列构成。PAL 是 PLA 的一个变种，与 PLA 不同的是，内部输入端出现了反相器且内部的或阵列是固定的，添加了多路开关、“异或”门、锁存器和寄存器时钟控制元件等，从而实现了用 PAL 替代大量的标准逻辑器件。PAL 的工作速度非常快，可用 PAL 设计高速控制器。随着 PAL、GAL 等大规模集成电路的广泛应用，一些初级的硬件描述语言 HDL(ABEL、CUPL、PALASM)相继问世，VHDL 和 Verilog HDL 是在这些初级 HDL 语言的基础上发展起来，并得到普及和应用。

随着电子制造工艺的发展，一些芯片制造商逐步推出新的结构的可编程逻辑器件，无论是集成度和编程性能都有了极大进步。按照结构，可编程逻辑器件分为复杂可编程逻辑器件(Complex Programmable Logic Device, CPLD)和现场可编程逻辑门阵列(Field Programmable Gate Array, FPGA)。CPLD 和 FPGA 吸取了 ASIC、PAL 和 GAL 器件各自的优点，成为了名副其实的高速度和高密度的超大规模集成电路 VLSI，它不但可以实现高密度和高速度的电路，为功能转化和低成本设计提供了可能。

下面章节将分别介绍 CPLD 器件和 FPGA 器件的性能特点。

1.3.2 CPLD 器件

CPLD 和 PAL 基于相同的技术制造，使用相同的开发工具，但 CPLD 实现的逻辑规模远大于 PAL。这里以 Altera 公司的 MAX7000S 系列为例，介绍 CPLD 器件的结构和特性。

1.3.2.1 MAX7000S 系列器件的内部结构

MAX7000S 的内部结构主要包括逻辑阵列块(Logic Array Block, LAB)、宏单元、扩展乘积项(Expander Product Terms, EPT)、可编程连线阵列(Programmable Interconnect Array, PIA)和 I/O 控制块(I/O Control blocks, IOC)。图 1-2 给出了 MAX7000S 器件的内部结构。另外，在 MAX7000S 系列器件的内部结构中还包括全局时钟输入和全局输出使能的控制线，这些线在不用作此功能时，可以作为一般的输入使用。

1. LAB

LAB 是 MAX7000S 系列器件中的最大的逻辑单元，由图 1-2 可以看出每个 LAB 由 16 个宏单元构成，每个 LAB 与各自相对应的 I/O 控制块相连，四个逻辑阵列块通过 PIA 和全局总线连接在一起，全局总线由所有的专用输入、I/O 引脚和宏单元反馈构成。利用这些连线可以实现不同 LAB 之间的连接，用以实现更复杂的逻辑功能。每个 LAB 有如

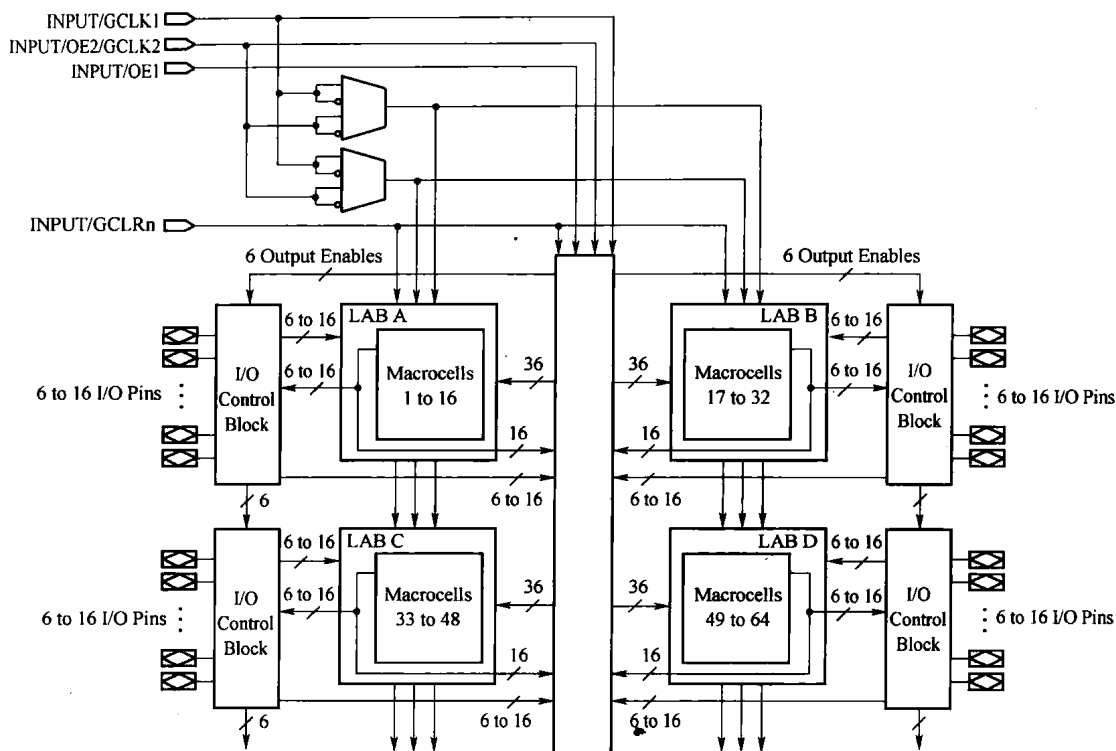


图 1-2 MAX7000S 器件的内部结构框图

下输入信号：

- (1) 来自通用逻辑输入的 PIA 的 36 个信号；
- (2) 用于寄存器辅助功能的全局控制信号；
- (3) 从 I/O 引脚到寄存器的直接输入通道，用以实现 MAX7000E 器件和 MAX7000S 器件的快速建立时间。

2. 宏单元

宏单元是 MAX7000 系列器件的具体逻辑单元，用来实现各种具体的逻辑功能。宏单元由逻辑阵列、乘积项选择矩阵和可编程触发器构成，其结构如图 1-3 所示。逻辑阵列用来实现组合逻辑函数，每个宏单元提供 5 个乘积项。通过乘积项选择矩阵实现这 5 个乘积项的逻辑函数，或者使这 5 个乘积项作为宏单元的触发器的辅助输入(清除、置位、时钟和时钟使能)。每个宏单元的一个乘积项还可以反馈到逻辑阵列。宏单元中的可编程触发器可以被单独编程为 D、T、JK 或 SR 触发器。可编程触发器还可以被旁路掉，用以实现纯组合逻辑方式工作。每个可编程触发器可按以下 3 种不同时钟方式进行控制。

- (1) 全局时钟。这种方式能够实现最快的时钟控制。
- (2) 带高电平使能的全局时钟。这种方式能够实现具有使能控制的触发器，并能够实现最快的时钟控制。
- (3) 来自乘积项的时钟。这种方式，触发器由来自隐含宏单元或 I/O 引脚的信号进行时钟控制，它一般具有较慢的时钟控制。

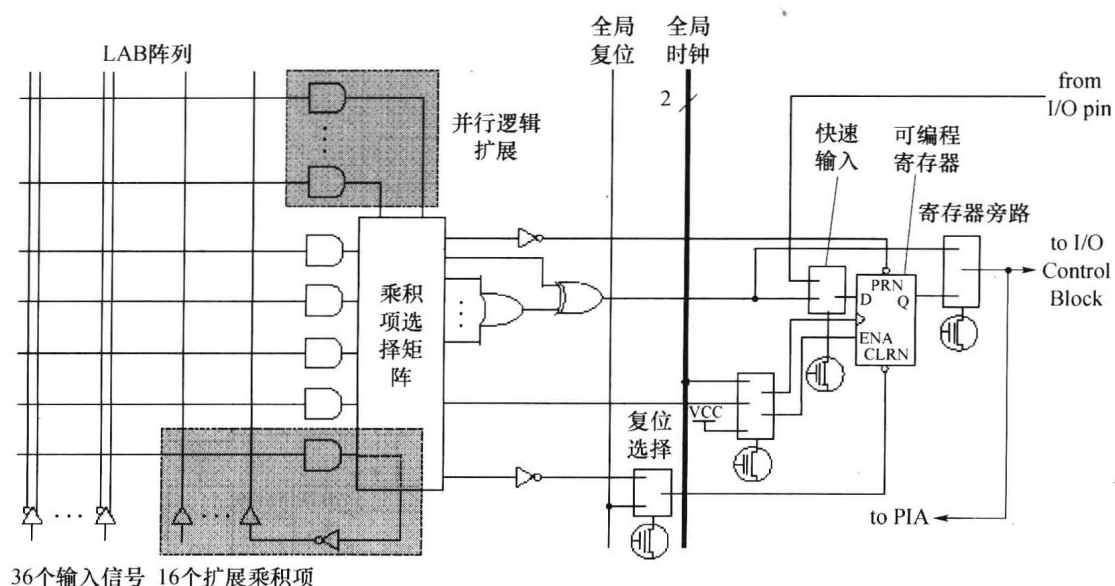


图 1-3 MAX7000 系列的宏单元结构图

MAX7000 系列器件有不同数量的全局时钟信号。有的器件有一个全局时钟信号，有的器件有两个全局时钟信号。例如，EMP7032、EMP7064、EMP7096 有一个全局时钟信号 GCLK，而 MAX7000E 和 MAX7000S 都有两个全局时钟信号 GCLK1、GCLK2

MAX7000E 和 MAX7000S 的 I/O 引脚都有一个宏单元寄存器的快速输入通道；它能够旁路 PIA 和组合逻辑，也允许将触发器作为具有快速建立时间的输入寄存器。

3. EPT

大多数逻辑函数用宏单元中的 5 个乘积项就可实现，但有些更复杂逻辑函数的实现则需要更多的乘积项。为提供所需的逻辑资源，就要用到另一个宏单元内部的逻辑单元的逻辑资源。MAX 7000 结构也允许利用共享和并联扩展乘积项(扩展项)作为附加的乘积项直接送到同一 LAB 的任意宏单元中。利用扩展项实现逻辑综合时，可节省逻辑资源，提高系统的工作速度。

1) 共享扩展项

每个 LAB 有多达 16 个共享扩展项。每个宏单元可提供一个未投入使用的乘积项作为共享扩展项，并将它们反相后反馈到逻辑阵列块，便于集中使用。每个共享 EPT 可被 LAB 内任何一个宏单元或全部宏单元使用和共享，以实现更为复杂的逻辑函数。采用共享扩展项后会增加一个短的延时 t_{SEXP} 。图 1-4 示出了共享扩展项是如何实现多个宏单元之间的连接的。

2) 并联扩展项

并联扩展项是宏单元中没有使用的乘积项，这些乘积项可分配到邻近的宏单元去实现快速复杂的逻辑函数。并联扩展项允许多达 20 个乘积项直接馈送到宏单元的“或”逻辑，其中，5 个乘积项是由宏单元本身提供的，15 个并联扩展项是由 LAB 中邻近宏单元提供的。