



Xilinx 大学合作计划指定教材

EDA Principle and
Implementation of VHDL

EDA原理及 VHDL实现

何 宾 编著
He Bin

清华大学出版社

Xilinx 大学合作计划指定教材

EDA Principle and Implementation of VHDL
EDA原理及VHDL实现

何 宾 编著

He Bin

清华大学出版社
北京

内 容 简 介

本书是为高等学校信息类和其他相关专业而编写的教材。本书共分为 14 章,主要介绍了数字系统 EDA 设计概论、可编程逻辑器件设计方法、VHDL 语言基础、数字逻辑单元设计、数字系统高级设计技术、基于 HDL 语言设计输入、基于原理图设计输入、设计综合和行为仿真、设计实现和时序仿真、设计下载和调试、数字时钟设计及实现、通用异步接收发送器设计及实现、数字电压表设计及实现、软件处理器 PicoBlaze 的原理及应用。

本书根据数字系统 EDA 课程的教学要求和笔者的实际教学实践体会,系统地介绍了数字系统 EDA 设计理论和方法,同时在书中给出了大量的设计实例,将理论和实践相结合。

本书可作为大学本科生和研究生教材,也可供从事 Xilinx 可编程逻辑器件设计的设计人员参考使用,同时也可作为 Xilinx 相关培训班的授课教材。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

EDA 原理及 VHDL 实现/何宾编著. —北京:清华大学出版社,2011.9

ISBN 978-7-302-26195-7

I. ①E… II. ①何… III. ①电子电路—计算机辅助设计—应用软件 IV. ①TN702

中国版本图书馆 CIP 数据核字(2011)第 137162 号

责任编辑:盛东亮

责任校对:焦丽丽

责任印制:何 芊

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62795954, jsjje@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者:北京市人民文学印刷厂

装 订 者:三河市金元印装有限公司

经 销:全国新华书店

开 本:185×260 印 张:22.75 字 数:511 千字

版 次:2011 年 9 月第 1 版 印 次:2011 年 9 月第 1 次印刷

印 数:1~3000

定 价:35.00 元

产品编号:042316-01

随着半导体技术的飞速发展、新电子产品上市周期的缩短以及数字化处理技术的不断提高,基于可编程逻辑器件设计逐渐成为电子系统设计中一个重要的研究方向和应用领域。在现阶段,必须依赖于高性能电子设计自动化技术,才能完成复杂数字系统的设计。采用可编程逻辑器件比采用专用集成电路和专用标准部件的成本要低,基于电子设计自动化技术完成复杂数字系统设计,大大缩短了设计周期,并且适应市场对产品竞争力的要求。

随着可编程逻辑器件复杂度的不断增加和电子设计自动化软件开发平台的不断完善,数字系统自动化设计手段越来越受到设计人员的重视。基于硬件描述语言、原理图、知识产权核等混合设计方法成为现代数字系统设计中采用的主要方法。当前,数字系统设计已经发展到了片上可编程系统阶段,要求设计人员能够实现软件和硬件的协同设计和调试。

本书在《EDA 原理及应用》(清华大学出版社,2009 年出版)一书的基础上,增加了包括片上可编程系统在内的很多新的内容,力图更加全面系统地介绍现代数字系统的设计原理和方法,使读者能够系统地掌握设计复杂数字系统的方法和技巧。

本书包含以下内容。

(1) 数字系统 EDA 设计概述,该部分主要介绍数字系统 EDA 技术的发展历史、数字系统设计方法和 HDL 语言。

(2) 可编程逻辑器件设计方法,该部分主要介绍可编程逻辑器件工艺,并重点介绍复杂可编程逻辑器件 CPLD 和现场可编程逻辑阵列 FPGA 的原理和内部结构,还对 Xilinx 芯片的性能和结构进行了比较详细的介绍。

(3) 硬件描述语言 VHDL,该部分介绍了 VHDL 语言的结构和风格、VHDL 语言的数据类型和运算符、VHDL 语言的主要描述语句等。在介绍这些内容时,为了便于学习,本书给出了大量单元模块的 VHDL 设计代码。

(4) 数字逻辑单元设计,该部分的介绍分成简单逻辑单元设计和复杂逻辑单元设计两个部分。简单逻辑单元设计部分重点介绍了组合逻辑电路设计和时序电路设计,复杂逻辑单元设计部分重点介绍了存储器设计、运算单元设计和有限自动状态机设计。

(5) VHDL 高级设计技巧,该部分重点介绍了提高 HDL 设计性能的几种常用方法,包括逻辑复制和复用、并行和流水技术、系统同步和异步单元及逻辑结构设计方法。通过该部分的学习,读者可以熟练运用 HDL 语言进行设计。

(6) IP 核设计技术,该部分介绍了 IP 核的分类、优化、生成和应用。通过该部分内容的学习,读者能更好地了解 IP 核设计技术的各个方面,为从事 IP 核设计打下良好基础。

(7) EDA 设计流程,该部分基于 Xilinx 的 ISE11 软件平台,介绍了基于 HDL 设计输入、基于原理图设计输入、设计综合和行为仿真、设计实现和时序仿真以及设计下载和调试。在介绍这些内容时,采用了混合设计方法,有利于读者掌握整个设计流程。该部分也是本书最重要的内容之一。

(8) 数字时钟设计及实现,该部分详细介绍了数字时钟的功能要求、数字时钟系统模块结构和数字时钟的具体实现方法,并提供了全部设计代码。

(9) 通用异步收发器的设计及实现,该部分详细介绍了通用异步接收/发送器的功能要求、模块结构和具体实现方法,并提供了全部设计代码。

(10) 数字电压表的设计及实现,该部分详细介绍了数字电压表的功能要求、模块结构和具体实现方法,并提供了全部设计代码。

(11) 微控制器 PicoBlaze 原理及应用部分,该部分以 Xilinx 的 PicoBlaze 的 8 位软核处理器为例,介绍了片上可编程系统的基本理论和设计方法,内容包括片上可编程系统概述、PicoBlaze 处理器原理及结构分析、PicoBlaze 处理器指令集、PicoBlaze 处理器汇编程序和基于 PicoBlaze 的 PWM 控制实现等内容。

为了让读者更好地掌握相关内容,本书还给出了大量设计示例程序和习题。在讲授和学习本书内容时,可以根据教学课时数和教学侧重点不同,适当将相关章节的内容进行调整和删减。本书不仅可以作为大学信息类专业 EDA 相关课程的教学用书,也可供从事 EDA 教学和科研工作的人员参考使用。

在本书的编写过程中引用和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,在此向他们表示衷心的感谢。同时,王刚领、彭勃、常晓磊负责了本书部分章节的编写与校对工作,王刚领负责第 6~10 章,彭勃负责第 1~4 章和第 11~13 章内容,常晓磊负责第 14 章内容;北京交通大学电子信息工程学院的陶丹老师也负责了本书第 1~2 章的编写工作,在此一并向他们表示感谢。在本书的出版过程中,得到了 Xilinx 公司大学合作计划和美国 Digilent 公司的大力支持和帮助,他们为本书的再次出版提供了宝贵的软件和硬件资源,同时也得到了清华大学出版社各位编辑的帮助和指导,在此也表示深深的谢意。

由于编者水平有限,编写时间仓促,书中难免有疏漏之处,敬请读者批评指正。

编 者

2011 年 8 月于北京

第 1 章 数字系统 EDA 设计概述	1
1.1 数字系统 EDA 技术的发展	2
1.1.1 数字系统设计技术发展历史	2
1.1.2 集成电路芯片的发展历史	7
1.2 数字系统设计方法	8
1.2.1 数字系统设计方法概述	8
1.2.2 SOP 和 POS 表达式	9
1.2.3 传统数字系统设计流程	11
1.2.4 计算机最小化技术	12
1.2.5 现代数字系统设计流程	16
1.3 HDL 语言	19
1.3.1 HDL 语言概述	19
1.3.2 HDL 语言的特点和发展	20
习题	22
第 2 章 可编程逻辑器件设计方法	23
2.1 可编程逻辑器件制造工艺	24
2.2 可编程逻辑器件结构	25
2.2.1 CPLD 原理及结构	25
2.2.2 FPGA 原理及结构	28
2.2.3 CPLD 和 FPGA 比较	40
2.3 Xilinx 可编程逻辑器件	41
2.3.1 Xilinx CPLD 芯片介绍	41
2.3.2 Xilinx FPGA 芯片介绍	43
2.3.3 Xilinx PROM 芯片介绍	55
2.4 可编程逻辑器件的选择原则	57
习题	58
第 3 章 VHDL 语言基础	59
3.1 VHDL 程序结构	60
3.1.1 VHDL 程序结构概述	60
3.1.2 VHDL 程序实体	60

目录

3.1.3	VHDL 程序结构体	62
3.2	VHDL 语言的描述风格	64
3.2.1	行为描述	65
3.2.2	数据流描述	65
3.2.3	结构描述	66
3.3	设计资源共享	67
3.3.1	库	67
3.3.2	包集合	69
3.3.3	子程序和函数	70
3.3.4	元件配置	72
3.4	VHDL 语言的文字规则	73
3.4.1	数字型文字	73
3.4.2	字符型文字	74
3.4.3	标识符	75
3.4.4	下标名及下标段名	76
3.5	VHDL 语言的数据对象、类型和属性	76
3.5.1	VHDL 语言的数据对象	76
3.5.2	VHDL 语言的数据类型	78
3.5.3	VHDL 语言的预定义属性	85
3.6	VHDL 语言的操作符	88
3.7	VHDL 语言的顺序描述语句	90
3.7.1	对象赋值语句	90
3.7.2	转向控制语句	92
3.7.3	断言语句	98
3.8	VHDL 语言的并发描述语句	100
3.8.1	进程描述语句	100
3.8.2	并行信号赋值语句	101
3.8.3	条件信号赋值语句	102
3.8.4	选择信号赋值语句	103
3.8.5	并行过程调用语句	103
3.8.6	块语句	104

3.9	VHDL 元件声明及例化语句	105
3.9.1	元件声明	105
3.9.2	元件例化	106
3.9.3	生成语句	108
3.10	VHDL 文件操作	109
	习题	111
第 4 章	数字逻辑单元设计	113
4.1	组合逻辑电路设计	114
4.1.1	门电路设计	114
4.1.2	编码器和译码器设计	114
4.1.3	数据选择器设计	116
4.1.4	数字比较器设计	117
4.1.5	总线缓冲器设计	118
4.2	数据运算单元设计	119
4.2.1	加法器设计	119
4.2.2	减法器设计	120
4.2.3	乘法器设计	120
4.2.4	除法器设计	121
4.2.5	算术逻辑单元设计	123
4.3	时序逻辑电路设计	125
4.3.1	触发器和锁存器的设计	125
4.3.2	计数器设计	128
4.3.3	移位寄存器设计	131
4.3.4	脉冲宽度调制(PWM)设计	137
4.4	存储器设计	139
4.4.1	ROM 设计	139
4.4.2	RAM 设计	140
4.5	有限自动状态机设计	141
4.5.1	FSM 设计原理	142
4.5.2	FSM 的分类及描述	144
	习题	152

目录

第 5 章 数字系统高级设计技术	153
5.1 VHDL 高级设计技巧	154
5.1.1 逻辑复制和复用技术	154
5.1.2 并行和流水线技术	157
5.1.3 同步和异步单元处理技术	160
5.1.4 逻辑结构处理技术	163
5.2 IP 核设计技术	167
5.2.1 IP 核的分类	167
5.2.2 IP 核的优化	168
5.2.3 IP 核的生成	169
5.2.4 IP 核的应用	169
习题	174
第 6 章 基于 HDL 语言设计输入	176
6.1 ISE 软件开发平台	177
6.2 建立工程	178
6.3 设计原理	182
6.4 添加设计和检查	183
6.5 创建基于 HDL 语言的模块	183
6.6 IP 核的生成和例化	186
6.6.1 timer_preset 模块的生成和例化	186
6.6.2 DCM 模块的生成和例化	189
习题	191
第 7 章 基于原理图设计输入	192
7.1 建立工程	193
7.2 设计原理	194
7.3 创建原理图模块	196
7.3.1 设置原理图编辑器	196
7.3.2 定义模块符号	197
7.3.3 创建模块及相关模块原理图	200

7.3.4 顶层模块原理图的设计	201
7.3.5 分配引脚位置	203
习题	205
第 8 章 设计综合和行为仿真	206
8.1 设计综合的实现	207
8.1.1 设计综合原理	207
8.1.2 设计综合实现	207
8.2 行为仿真的实现	210
8.2.1 测试平台概述	210
8.2.2 仿真环境的设置	211
8.2.3 添加 HDL 测试平台	212
8.2.4 基于 Modelsim 实现行为仿真	212
8.2.5 基于 ISim 实现行为仿真	216
习题	220
第 9 章 设计实现和时序仿真	221
9.1 实现过程概述及约束	222
9.1.1 实现过程概述	222
9.1.2 继续前面的设计	222
9.2 设计实现过程	222
9.3 设置实现属性参数	223
9.4 创建时序约束	224
9.5 设计翻译	225
9.6 设计约束	226
9.6.1 编辑约束	226
9.6.2 使用 PlanAhead 分配 I/O 位置	229
9.7 设计映射及时序分析	232
9.7.1 设计映射	232
9.7.2 使用时序分析评估块延迟	233
9.8 布局布线验证	234
9.8.1 用 FPGA Editor 验证布局布线	235

目录

9.8.2 评估布局后的时序	237
9.8.3 使用 PlanAhead 分析设计	237
9.9 时序仿真实现	238
9.9.1 时序仿真概述	238
9.9.2 使用 ModelSim 进行时序仿真	239
9.9.3 使用 ISim 仿真器进行时序仿真	243
习题	247
第 10 章 设计下载和调试	248
10.1 PLD 配置接口	249
10.1.1 JTAG 模式	249
10.1.2 串行模式	250
10.1.3 SelectMAP 模式	253
10.1.4 主 SPI 模式	255
10.1.5 主 BPI 模式	257
10.2 创建配置数据	258
10.2.1 配置属性	258
10.2.2 创建 PROM 文件	259
10.3 下载实现	261
10.3.1 下载实现设置	261
10.3.2 JTAG 诊断	266
10.3.3 建立 SVF 文件	267
10.4 PLD 调试	269
10.4.1 应用多路复用技术	270
10.4.2 应用虚拟逻辑分析工具	270
10.4.3 应用 ChipScope Pro 调试工具	271
习题	273
第 11 章 数字时钟设计及实现	274
11.1 数字时钟的功能要求和结构	275
11.1.1 数字时钟的功能要求	275
11.1.2 数字时钟的整体结构	275

11.2	模块设计	275
11.2.1	数字时钟控制信号	275
11.2.2	控制模块结构	276
11.3	设计实现	278
	习题	284
第 12 章	通用异步接收/发送器设计及实现	285
12.1	UART 设计原理	286
12.1.1	UART 原理和设计描述	286
12.1.2	接收模块设计	287
12.1.3	发送模块设计	288
12.1.4	UART 的 VHDL 设计代码	289
12.2	UART 设计验证	295
12.2.1	验证原理	295
12.2.2	验证代码	295
	习题	297
第 13 章	数字电压表设计及实现	298
13.1	数字电压表的功能要求和结构	299
13.1.1	数字电压表的功能要求	299
13.1.2	数字电压表的整体结构	299
13.2	模块设计	299
13.2.1	数字电压表控制信号	299
13.2.2	ADC 转换原理	300
13.2.3	控制模块结构	301
13.3	设计实现	302
13.3.1	ADC 控制模块原理及实现	302
13.3.2	显示控制模块原理及实现	304
13.3.3	程序包的设计	306
13.3.4	顶层模块的设计	308
	习题	309

目录

第 14 章 软核处理器 PicoBlaze 的原理及应用	310
14.1 片上可编程系统概述	311
14.1.1 片上微控制器和专用微控制器的比较	311
14.1.2 片上微控制器和片上逻辑的比较	311
14.2 PicoBlaze 微控制器的原理及结构分析	312
14.2.1 PicoBlaze 微控制器概述	312
14.2.2 PicoBlaze 微控制器应用框架	313
14.2.3 PicoBlaze 微控制器的内部结构及分析	314
14.3 PicoBlaze 微控制器指令集	317
14.3.1 PicoBlaze 指令类型及编码	317
14.3.2 控制程序转移指令	317
14.3.3 中断指令	319
14.3.4 逻辑操作指令	320
14.3.5 算术运算指令	323
14.3.6 循环转移指令	326
14.3.7 输入和输出指令	327
14.4 PicoBlaze 微控制器汇编程序	328
14.4.1 KCPSM3 汇编器原理及操作	328
14.4.2 KCPSM3 编程语法	333
14.4.3 KCPSM3 中断处理	335
14.4.4 KCPSM3 中的 CALL/RETURN 栈	338
14.4.5 KCPSM3 共享程序空间	339
14.4.6 KCPSM3 输出/输入端口的设计	340
14.5 基于 PicoBlaze 微控制器的 PWM 控制	342
14.5.1 系统性能要求和结构	342
14.5.2 系统 PWM 控制原理	344
14.5.3 系统 UART 模块设计	347
习题	349

第1章

数字系统EDA设计概述

本章主要介绍数字系统 EDA 技术发展概述、数字系统设计方法、HDL 语言。数字系统设计技术发展概述部分介绍了数字系统设计技术发展历史和集成电路芯片的发展历史；数字系统设计方法部分介绍了数字系统设计方法概论、SOP 和 POS 表达式、传统数字系统设计流程、计算机最小化技术和现代数字系统设计流程；HDL 语言部分介绍了 HDL 语言的概念、HDL 语言的特点和比较、HDL 语言的最新发展。

通过本章内容的学习，读者可对现代数字系统的设计方法、设计工具和设计芯片有整体的认识，为学习后续章节打下基础。

1.1 数字系统 EDA 技术的发展

EDA(Electronics Design Automation, 电子设计自动化)技术是一门迅速发展起来的新技术。EDA 设计就是设计人员在计算机上通过特定功能的软件开发工具,以全自动或半自动化方式按要求完成电子系统的设计。EDA 技术的不断发展,使得电子系统的设计效率大大提高,设计成本大大降低,并且提高了电子系统的设计可靠性,推动了信息技术的不断发展。

EDA 技术所涉及的领域相当广泛,业界一般将 EDA 技术分成狭义的 EDA 技术和广义的 EDA 技术。

狭义的 EDA 技术,就是指以大规模可编程逻辑器件为设计载体,以硬件描述语言为系统逻辑描述的主要表达方式,以计算机、大规模可编程逻辑器件的开发软件及实验开发系统为设计工具,通过有关的开发软件,自动完成用软件方式设计的电子系统到硬件系统的逻辑编译、逻辑化简、逻辑分割、逻辑综合及优化、逻辑布局布线、逻辑仿真,直至对特定目标芯片的适配编译、逻辑映射、编程下载等工作,最终形成集成电子系统或专用集成芯片的一门新技术,或称为 IES/ASIC 自动设计技术。简单来说,狭义的 EDA 技术也就是使用 EDA 软件进行数字系统的设计,这也是本书所要介绍的内容。

广义的 EDA 技术,是通过计算机及其电子系统的辅助分析和设计软件,完成电子系统某一部分的设计过程。因此,广义的 EDA 技术除了包含狭义的 EDA 技术外,还包括计算机辅助分析 CAA 技术(如 PSPICE,EWB,MATLAB 等)、印刷电路板计算机辅助设计 PCB-CAD 技术(如 PROTEL,ORCAD 等),以及其他高频和射频设计和分析的工具等。

1.1.1 数字系统设计技术发展历史

1. 数字逻辑和计算机的发展背景

在过去的 60 年中,数字逻辑改变了整个世界,整个世界逐渐朝着数字化方向发展。我们今天所熟悉的计算机在第二次世界大战后才出现在人类世界中。表 1.1 列出了在计算机和数字逻辑发展历史上的重大事件,从该表中可以看出数字逻辑设计技术经过了近 400 年的逐步进化的过程。

公元前 3000 年出现的算盘是最早帮助人类进行计算的工具,直到今天仍然还发挥着其重要的作用。但是直到 16 世纪,人类才开始真正地设计帮助人类进行计算的机器。这些计算机器中,最著名的是 Blaise Pascal 于 19 岁时发明的 Pascaline,当时是用于帮助其父亲进行税务计算工作。Pascal 建立了 5~8 个数字版本的 Pascaline,每个数字和一个表盘、一个轴和一个齿轮关联。该计算机器能进行加法(通过生成减数的 9 位补码进行加法运算)和减法,其运算能力是有限的。

表 1.1 计算机和数字逻辑发展历史上的重大事件

年 份	事 件
公元前 3000 年	巴比伦王国开发了算盘。这个装置使用的线(棍子)上的一系列珠子表示数字,今天仍然在远东地区的一些地方使用,用于执行计算
1614—1617	John Napier, 苏格兰数学家、发明了对数,允许通过加来进行乘法,通过减进行除法。他发明了棒子(即树枝),这样可以通过一种特殊的方法移动棍子来实现对大数的乘或者除运算
1623	Wilhelm Schickard, 德国教授,发明了第一个机械式计算器,称为“计算钟”
1630	William Oughtred, 英国数学家和牧师,发明了计算尺
1642—1644	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	Gottfried Wilhelm Von Leibniz, 德国数学家、外交官、历史学家、法学家和微分的发明家,发明了一个称为步进式计算器的机械计算器。计算器有一个独一的齿轮——莱布尼茨轮,用于机械式的乘法器。这个计算器对未来的机械式计算器的发展产生了深远的影响
1823—1839	Charles Babbage, 英国数学家和发明家,开始在他的差分机上工作,该机器设计用于自动处理对数计算。虽然有大量来自政府的工作和资金,但最终没有完成差分机。1834 年, Babbage 开始在一个功能更强的机器上工作,称为分析机,它被称为第一个通用计算机。由于所要求的精确的机械齿轮,不能在时间上准确的产生,因此没有工作。所以, Babbage 被认为是“计算机之父”
1854	George Boole, 英国逻辑学家和数学家,出版了 <i>Investigation of the Law of Thought</i> , 给出了逻辑数学的基础
1890	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	Lee De Forest, 美国物理学家,发明了三极管。直到 1940 年前,这些管子都没有用于计算机中
1936	Alan M. Turing, 英国逻辑学家,发表了一篇论文 <i>On Computable Numbers</i> , 说明任意的计算都可以使用有限状态机实现。Turing 在第二次世界大战后的英国的早期计算机研制中扮演了重要的角色
1937	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,能进行加、减、乘和除运算
1938	Konrad Zuse, 德国工程师,构建了 Z1——第一个二进制计算机器。1941 年,他完成了 Z3——通用的电子机械式计算机器
1938	Claude Shannon, 基于他在 MIT 的硕士论文,发表了 <i>A Symbolic Analysis of Relay and Switching Circuits</i> , 在该著作中,他说明了符号逻辑和二进制数学如何应用到继电器电路中
1942	John V. Atanasoff, 美国爱荷华州立大学教授,完成了一个简单的电子计算机器
1943	IBM-Harvard Mark I, 一个大的可运行的电子机械式计算器
1944—1945	J. Presper Eckert 和 John W. Mauchly, 在宾夕法尼亚大学的电气工程摩尔学院,设计和建立了 EMIAC。EMIAC 是首个全功能的电子电子计算器
1946	John Von Neumann, ENIAC 项目的顾问,在完成该工程后,写了一个很有影响力的报告,之后在普林斯顿高等研究院开始他自己的计算机项目

续表

年 份	事 件
1947	Walter Brattain, John Bardeen 和 William Shockley 在贝尔实验室发明了晶体管
1948	在英国,在 Manchester Mark I 电子计算机上运行第一个存储程序
1951	发布第一个商业制造的计算机, Ferranti Mark I 和 UNIVAC
1953	IBM 发布了一个电子计算机——701
1958	Kack kilby, 得州仪器公司的一名工程师, 建立了一个可移相的振荡器, 作为第一个集成电路(Integrated Circuit, IC)
1959	Robert Noyce, 1958 年所建立的仙童半导体公司的联合创始人, 生产了第一个集成电路平面工艺。这使得实际大规模生产可靠的集成电路变为可能。1968 年, Noyce 成立了 Intel 公司
1963	数字设备公司 DEC 生产了首个小型计算机
1964	IBM 生产了 System/360 系列电脑主机
1965	在电子杂志上, Gordon Moore 预测一个集成芯片上的元件数量会在每一年翻一倍。这就是著名的“摩尔定律”。在 1975 年, 他修正了该定律, 元件数量开始每两年翻一倍
1969	IBM 的研究人员开发出第一个片上可编程逻辑阵列(Programmable logic array, PLA)
1971	Marcian E. Hoff Jr., Intel 公司的工程师, 发明了第一个微处理器
1975	Intersil 公司生产了第一片现场可编程逻辑阵列(Field Programmable logic array, FPLA)
1978	单片存储器引入了 PLA
1981	IBM 个人电脑诞生。 美国国防部开始开发 VHDL。VHDL 中的 V 表示 VHSIC(Very High Speed Integrated Circuit, 超高速集成电路), HDL 代表 Hardware Description Language
1983	Intermetric, IBM 和 TI 授权开发 VHDL
1984	Xilinx 成立, 并发明了现场可编程门阵列(Field Programmable Gate Array, FPGA)。 Gateway 设计自动化公司引入了硬件描述语言 Verilog
1987	VHDL 成为 IEEE 标准(IEEE 1076)
1990	Cadence Design System 收购 Verilog
1995	Verilog 成为 IEEE 标准

Charles Babbage 被认为是“计算机之父”。他努力生产了一种机器,能用于计算数学表(比如对数),并于 1822 年建立了差分机的工作模型。这个机器通过差分方法来计算在一个表中的 6 位数字。Babbage 为全面的差分机制定了详细的计划,能计算到 20 位,并且生产了一个金属盘用于打印表格。这个机器依靠蒸汽运行,10 英尺高,10 英尺宽,5 英尺深。通过英国政府的资助,Babbage 和他的主要的机械工程师一起工作,尝试建立差分机器,但这里有很多技术和个人问题(当前的机器工具不能满足 Babbage 的精度要求,她妻子的病逝,和机械师的意见不一致),一直妨碍着这个机器的建成。在 1834 年, Babbage 设想了一种更强的分析机,它能解决数学问题。政府于 1842 年终止了这个项目。Babbage 尽管知道在那个时代不可能建立分析机,但是他余生致力于设计这种机器。他给出了有关分析机的大量注释,包含成百个轴和上千个齿轮和轮子;还有今天计算机的很多元件,包括存储器和 CPU。打孔卡用于机器的外部编程。