



首本使用汇编和C双语言双实例解读8051单片机系统的书籍。

本书概念清晰，实例丰富，循序渐进引导读者边做边学。

作者系此领域知名作者，是多本畅销教材的合著者，权威性毋庸置疑。



The 8051 Microcontroller  
A Systems Approach

# 深入理解 8051单片机系统

穆罕默德·阿里·马齐迪 (Muhammad Ali Mazidi)

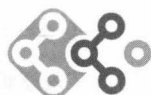
[美] 詹尼斯·吉利斯皮·马齐迪 (Janice Gillispie Mazidi) 著

罗兰 D. 麦金利 (Rolin D. McKinlay)

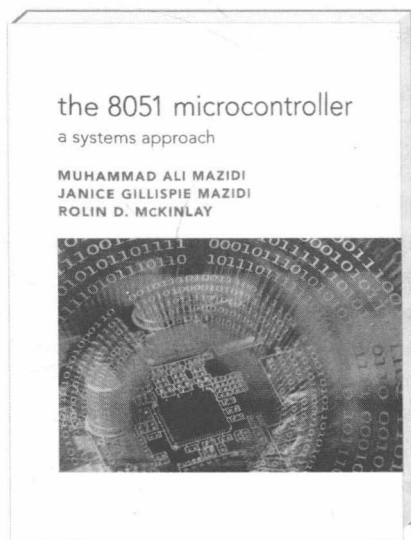
张红英 译



机械工业出版社  
China Machine Press



电子与嵌入式系统  
设计译丛



The 8051 Microcontroller  
A Systems Approach

# 深入理解 8051单片机系统

穆罕默德·阿里·马齐迪 (Muhammad Ali Mazidi)  
[美] 詹尼斯·吉利斯皮·马齐迪 (Janice Gillispie Mazidi) 著  
罗兰 D. 麦金利 (Rolin D. McKinlay)

张红英 译



机械工业出版社  
China Machine Press

## 图书在版编目(CIP)数据

深入理解 8051 单片机系统 / (美) 马齐迪 (Mazidi, M. A.), (美) 马齐迪 (Mazidi, J. G.), (美) 麦金利 (McKinlay, R. D.) 著; 张红英译. —北京: 机械工业出版社, 2015.11

(电子与嵌入式系统设计译丛)

书名原文: The 8051 Microcontroller: A Systems Approach

ISBN 978-7-111-51797-9

I. 深… II. ①马… ②马… ③麦… ④张… III. 单片微型计算机 IV. TP368.1

中国版本图书馆 CIP 数据核字 (2015) 第 251264 号

本书版权登记号: 图字: 01-2012-7773

Authorized translation from the English language edition, entitled *The 8051 Microcontroller: A Systems Approach*, 9780135080443 by Muhammad Ali Mazidi, Janice Gillispie Mazidi, Rolin D. Mckinlay, published by Pearson Education, Inc., Copyright © 2013.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc.

Chinese simplified language edition published by Pearson Education Asia Ltd., and China Machine Press Copyright © 2016.

本书中文简体字版由 Pearson Education (培生教育出版集团) 授权机械工业出版社在中华人民共和国境内 (不包括中国台湾地区和香港、澳门特别行政区) 独家出版发行。未经出版者书面许可, 不得以任何方式抄袭、复制或节录本书中的任何部分。

本书封底贴有 Pearson Education (培生教育出版集团) 激光防伪标签, 无标签者不得销售。

## 深入理解 8051 单片机系统

出版发行: 机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码: 100037)

责任编辑: 张国强

责任校对: 殷 虹

印 刷: 三河市宏图印务有限公司

版 次: 2016 年 1 月第 1 版第 1 次印刷

开 本: 186mm × 240mm 1/16

印 张: 31.75

书 号: ISBN 978-7-111-51797-9

定 价: 119.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010) 88378991 88361066

投稿热线: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: hzjsj@hzbook.com

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光 / 邹晓东

# 前 言

8051 微控制器的生产商众多，架构简单，因此得到了广泛的应用。本书适用于大学微控制器课程和嵌入式系统课程教学。它不仅可以奠定汇编语言程序设计的基础，而且可以为工程专业的学生设计 8051 接口提供综合解决方案。以此为基础，可以进一步探索基于微控制器的嵌入式系统的设计技术和接口技术。技术人员、硬件工程师、计算机科学家和业余爱好者都可使用本书。本书不仅可以为独立系统及采集数据到 PC 的系统提供设计思想，还可以为联网的分布式系统提供理想的参照。

## 必备条件

读者需学习过入门级的数字系统课程。汇编语言知识对阅读本教程有帮助，但非必需。虽然本书为无汇编语言背景的读者设计，但对有汇编语言经验的学生来说，更能快速掌握 8051 架构，并立即启动项目。对于本书的 8051 C 编程部分，需掌握 C 语言编程的基础知识。

## 本书结构

本书运用系统且循序渐进的方法覆盖 8051 C 和汇编语言编程以及接口编程的各个方面。给出了许多例程和示例以澄清概念，并为学生提供边做边学的机会。每节末尾提供了习题以加强对要点的理解。

第 0 章涵盖了数制（二进制、十进制、十六进制），并介绍了基本逻辑门和存储器的术语。该章还探讨了存储器的概念和 I/O 地址解码。

第 1 章论述了 8051 的历史以及 8031、8751、89C51、DS5000 和 DS89C4x0 等其他 8051 系列成员的特性。该章还提供了 8051 芯片生产商的列表。

第 2 章论述了 8051 的内部架构，解释了如何使用 8051 汇编器创建可执行程序。该章还探讨了栈和标记寄存器，讨论了 RISC 的概念和 CISC 架构。

第 3 章探讨了循环、跳转和调用指令，并附有许多编程实例。

第 4 章探讨了 I/O 端口。通过学习，参与项目的学生可以开始使用 8051 I/O 接口，快速启动项目。

第 5 章涵盖了 8051 的寻址方式，解释了如何使用 8051 的代码空间来存储数据，以及如何访问数据。

第 6 章重点介绍了运算、逻辑指令和程序。

第7章涵盖了8051的C语言编程。

第8章探讨了8051芯片的硬件接线。

第9章描述了8051定时器，以及如何将其用作事件计数器。

第10章着重介绍8051的串行数据通信及其与RS232的接口。该章还展示了8051与x86计算机COM端口的通信。此外，该章还涵盖了DS89C4x0的第二个串口的使用。

第11章详细介绍了8051的中断以及中断处理程序的编程。

第12章展示了8051与外设的接口，如LCD和键盘。

第13章展示了8051与其他器件的接口，如DAC芯片、ADC芯片及传感器。

第14章涵盖了8031/8051与外部存储器的接口，包含ROM和RAM。

第15章讨论了光隔离器、继电器和步进电机。

第16章展示了如何连接DS12887实时时钟芯片以及如何对其编程。

第17章展示了直流电机的基本接口以及使用PWM的基础知识。

第18章展示了I2C和SPI协议的基本概念。

附录用于提供本书中涉及的知识点所需的各种参考资料。附录A详细介绍了每个8051指令并举例，也提供了指令的时钟数、8051寄存器图及RAM存储器映射。附录B介绍了绕线的基础。附录C涵盖了IC技术以及逻辑系列，包括8051的I/O端口和扇出，请确保你在学习这些之前已将8051与外部装置连接到了一起。附录D探讨了流程图和伪代码的使用。附录E针对那些熟悉x86架构并且想快速过渡到8051架构的读者。附录F提供了ASCII字符表。附录G列出了汇编器软件以及电子器件的资源。

## 作者简介

Muhammad Ali Mazidi 拥有南卫理公会大学和得克萨斯大学的硕士学位。他现在是南卫理公会大学电子工程系博士，同时，他也是多部教材的合著者，包括《The x86 IBM PC》《PIC Microcontroller and Embedded Systems》《HCS12 Microcontroller and Embedded Systems》《AVR Microcontroller and Embedded Systems》等，这些教材均由 Prentice Hall 出版社出版。他在德锐大学讲授如何设计基于微控制器的系统。他也是网站 [www. MicroDigitalEd. com](http://www.MicroDigitalEd.com) 的创办者。

Janice Gillispie Mazidi 拥有北得克萨斯州大学的计算机科学硕士学位。她不仅是一名软件工程师，而且拥有多年的教学经验。她也是多部教材的合著者，如 Prentice Hall 出版社的《The x86 PC》。目前，她在得克萨斯大学中讲授编程课程。

Rolin D. McKinlay 拥有德锐大学电子工程技术博士学位。现是德拉斯公司的嵌入式系统工程师和 FPGA 设计顾问。如果你有任何建意或疑问，请通过以下邮箱联系作者：[mdebooks@yahoo. com](mailto:mdebooks@yahoo.com)。

# 目 录

前 言

作者简介

## 第0章 计算机简介/1

0.1 数制与编码系统/2

0.2 数字基础知识/8

0.3 半导体存储器/12

0.4 总线设计与地址译码/22

0.5 I/O 地址解码与设计/29

0.6 CPU 结构/34

总结/37

习题/38

复习题答案/41

## 第1章 8051 微控制器/43

1.1 微控制器与嵌入式处理器/44

1.2 8051 系列概述/47

总结/52

推荐网址链接/52

习题/52

复习题答案/53

## 第2章 8051 汇编语言编程/54

2.1 8051 内部寄存器及作用/55

2.2 8051 汇编编程简介/57

2.3 汇编并运行 8051 程序/59

2.4 8051 中的程序计数器和

ROM 空间/61

2.5 8051 数据类型和指令/64

2.6 8051 标志位和 PSW 寄存器/65

2.7 8051 寄存器组和栈/68

2.8 RISC 体系结构/75

总结/77

习题/78

复习题答案/81

## 第3章 跳转、循环和调用指令/83

3.1 跳转和循环指令/84

3.2 调用指令/88

3.3 各种 8051 芯片的延时/93

总结/98

推荐网址链接/98

习题/99

复习题答案/101

## 第4章 I/O 端口编程/102

4.1 8051 的 I/O 编程/103

4.2 I/O 位操作编程/107

总结/112

习题/113

复习题答案/113

## 第5章 8051 寻址方式/115

5.1 立即寻址和寄存器寻址方式/116

5.2 使用多种寻址方式访问存储器/117

5.3 I/O 和 RAM 的位地址/125

5.4 8052 中额外的 128 字节片上 RAM/134

总结/136

习题/137

复习题答案/139

## 第6章 算术逻辑指令与程序/141

6.1 算术指令/142

6.2 有符号数概念及算术操作/149

6.3 逻辑和比较指令/153

6.4 移位指令和数据序列化/158

6.5 BCD 码、ASCII 码和其他应用程序/163

总结/169

习题/170

复习题答案/173

## 第 7 章 8051 C 语言编程/176

7.1 8051 C 中的数据类型和延时/177

7.2 C 语言 I/O 编程/182

7.3 8051 C 中的逻辑操作/187

7.4 8051 C 中的数据转换程序/190

7.5 8051 C 中的代码 ROM 空间访问/195

7.6 8051 C 中的数据序列化/199

总结/201

推荐网址链接/201

习题/201

复习题答案/203

## 第 8 章 8051 硬件接线及英特尔十六进制文件/205

8.1 8051 引脚概述/206

8.2 DS89C4x0 板的设计和测试/211

8.3 英特尔十六进制文件释义/218

总结/220

推荐网址链接/220

习题/220

复习题答案/222

## 第 9 章 8051 定时器汇编及 C 语言编程/223

9.1 8051 定时器编程/224

9.2 计数器编程/235

9.3 8051 定时器 0 和定时器 1 中的 C 语言编程/240

总结/249

习题/250

复习题答案/252

## 第 10 章 8051 串口汇编和 C 语言编程/253

10.1 串行通信基础/254

10.2 8051 与 RS232 的连接/259

10.3 使用汇编语言编写 8051 串口程序/261

10.4 编写第二个串口程序/271

10.5 使用 C 语言编写串口程序/277

总结/281

习题/281

复习题答案/283

## 第 11 章 中断的汇编和 C 语言编程/285

11.1 8051 中断/286

11.2 定时器中断编程/289

11.3 外部硬件中断编程/290

11.4 串行通信中断编程/297

11.5 8051/52 中断优先级/300

11.6 中断的 C 语言编程/302

总结/306

习题/306

复习题答案/309

## 第 12 章 LCD 和键盘接口/311

12.1 LCD 接口/312

12.2 键盘接口/322

总结/328

推荐网址链接/328

习题/328

复习题答案/329

## 第 13 章 ADC、DAC 和传感器接口/330

13.1 并行和串行 ADC/331

13.2 DAC 接口/351

13.3 传感器接口与信号调理/354

总结/358

习题/358

复习题答案/361

**第 14 章 8051 与外部存储器接口/362**

14.1 8031/51 接口与外部 ROM/363

14.2 8051 数据存储空间/366

14.3 使用 8051 C 语言访问外部数据  
存储器/375

总结/378

习题/378

复习题答案/380

**第 15 章 继电器、光隔离器和  
步进电机/381**

15.1 继电器和光隔离器/382

15.2 步进电机接口/386

总结/393

习题/393

复习题答案/394

**第 16 章 DS12887 RTC 接口与  
编程/395**

16.1 DS12887 RTC 接口/396

16.2 DS12887 RTC 的 C 语言编程/403

16.3 DS12887 芯片的报警、SQW 以及  
IRQ 特性/406

总结/413

习题/413

复习题答案/415

**第 17 章 直流电机控制和脉宽调制/417**

17.1 直流电机接口和 PWM/418

总结/427

推荐网址链接/427

习题/427

复习题答案/427

**第 18 章 SPI 和 I2C 协议/428**

18.1 SPI 总线协议/429

18.2 I2C 总线协议/433

总结/439

推荐网址链接/439

习题/439

复习题答案/440

**附录 A 8051 指令、定时器和  
寄存器/441****附录 B 布线基础/472****附录 C IC 技术和系统设计问题/474****附录 D 流程图和伪代码/490****附录 E 8051 入门/494****附录 F ASCII 码/495****附录 G 汇编器、开发资源和  
供应商/497**

# 计算机简介

## 本章目标

完成本章后，你能够

- ◆ 将任意二进制、十进制或十六进制数转换成另外两个进制的数。
- ◆ 描述逻辑操作 AND、OR、NOT、XOR、NAND 以及 NOR。
- ◆ 使用逻辑门画出简单的电路图。
- ◆ 解释位、半字节、字节及字之间的区别。
- ◆ 给出千字节、兆字节、吉字节及兆兆字节的精确数学定义。
- ◆ 描述计算机系统中主要部件的功能。
- ◆ 描述计算机系统中 CPU 的角色。
- ◆ 比较不同类型半导体存储器的容量、结构以及访问时间。
- ◆ 描述位于芯片上的内存位置数、数据引脚数以及芯片存储能力之间的关系。
- ◆ 比较 PROM、EPROM、UV-EPROM、EEPROM、闪存 EPROM 以及掩膜 ROM。
- ◆ 比较 SRAM、NV-RAM 以及 DRAM 存储器。
- ◆ 列出 CPU 存储器地址解码的步骤。
- ◆ 列出计算机中三种类型的总线并介绍每种总线的作用。
- ◆ 描述外围的和存储器映射的 I/O 总线。
- ◆ 设计存储器和 I/O 的地址解码器。
- ◆ 列出 CPU 的主要部件以及每个部件的功能。
- ◆ 理解哈佛体系结构。

为了更好地理解微控制器的软硬件结构，先要理解计算机体系结构的一些基本概念。本章在传统的计算机书籍中称为第0章。在0.1节中讲解数制以及编码系统的基础；在0.2节中介绍逻辑门概述；0.3节中讨论半导体存储器；0.4节讨论存储器与地址解码器之间的连接；0.5节探讨I/O地址解码；最后0.6节介绍哈佛体系结构和冯·诺依曼体系结构的CPU。虽然读者可能已经了解了本章的一些知识，但还是建议通过本章简单复习一下。

## 0.1 数制与编码系统

人类使用的是十进制（decimal）算数，计算机中使用的是二进制（binary）系统。本节将解释怎样将十进制系统转换成二进制系统，反之亦然。二进制的一种更方便的表示法是十六进制（hexadecimal），本节也会涉及。最后将探讨二进制形式的字母数字代码，即ASCII码。

### 0.1.1 十进制和二进制数制系统

尽管有猜测说人类使用十进制系统的最初原因是人类有十根手指，而计算机使用二进制系统的原因却无需猜测。原因是1和0代表两种电压级别，即开和关。十进制中有10个不同的数字符号0, 1, 2, ..., 9，在二进制中却只有两个不同的数字符号，即0和1来构造数字，这两个数字0和1通常称为位。

### 0.1.2 十进制转换成二进制

将十进制转换成二进制的方法之一是十进制数反复除以2，每次保存余数，直到商为0时结束。然后将余数逆着顺序写便可以得到对应的二进制数。

如例0-1所示。

### 0.1.3 二进制转换成十进制

将二进制转换成十进制，关键是理解与每个数位对应的权值。首先，作为一个类比，将十进制数的数位权重举例如下图所示。

|                        |                   |   |        |
|------------------------|-------------------|---|--------|
| 740683 <sub>10</sub> = |                   |   |        |
| 3                      | ↔ 10 <sup>0</sup> | = | 3      |
| 8                      | ↔ 10 <sup>1</sup> | = | 80     |
| 6                      | ↔ 10 <sup>2</sup> | = | 600    |
| 0                      | ↔ 10 <sup>3</sup> | = | 0000   |
| 4                      | ↔ 10 <sup>4</sup> | = | 40000  |
| 7                      | ↔ 10 <sup>5</sup> | = | 700000 |
|                        |                   |   | <hr/>  |
|                        |                   |   | 740683 |

**例 0-1** 将  $25_{10}$  转换成二进制数。

|             |   |               |
|-------------|---|---------------|
| 解:          | 商 | 余数            |
| $25/2 = 12$ |   | 1 LSB (最低有效位) |
| $12/2 = 6$  |   | 0             |
| $6/2 = 3$   |   | 0             |
| $3/2 = 1$   |   | 1             |
| $1/2 = 0$   |   | 1 MSB (最高有效位) |

所以,  $25_{10} = 11001_2$ 。

同样地, 二进制中每一数位有相应的权值:

|                                   |     |        |
|-----------------------------------|-----|--------|
| $110101_2 =$                      | 十进制 | 二进制    |
| $1 \times 2^0 = 1 \times 1 = 1$   |     | 1      |
| $0 \times 2^1 = 0 \times 2 = 0$   |     | 00     |
| $1 \times 2^2 = 1 \times 4 = 4$   |     | 100    |
| $0 \times 2^3 = 0 \times 8 = 0$   |     | 0000   |
| $1 \times 2^4 = 1 \times 16 = 16$ |     | 10000  |
| $1 \times 2^5 = 1 \times 32 = 32$ |     | 100000 |
|                                   | 53  | 110101 |

理解了二进制中每一位的权值后, 将它们相加到一块就可以很简单地得到对应的十进制数, 如例 0-2 所示。

**例 0-2** 将  $11001_2$  转换成十进制数。

|    |    |        |       |       |       |               |
|----|----|--------|-------|-------|-------|---------------|
| 解: | 权: | 16     | 8     | 4     | 2     | 1             |
|    | 数: | 1      | 1     | 0     | 0     | 1             |
|    | 和: | $16 +$ | $8 +$ | $0 +$ | $0 +$ | $1 = 25_{10}$ |

知道了二进制中数位相关的权值后, 就可以直接将十进制数转换成二进制数而不需要反复进行除法操作, 如例 0-3 所示。

**例 0-3** 使用权的概念将  $39_{10}$  转换成二进制数。

|    |    |        |       |       |       |       |          |
|----|----|--------|-------|-------|-------|-------|----------|
| 解: | 权: | 32     | 16    | 8     | 4     | 2     | 1        |
|    |    | 1      | 0     | 0     | 1     | 1     | 1        |
|    |    | $32 +$ | $0 +$ | $0 +$ | $4 +$ | $2 +$ | $1 = 39$ |

所以,  $39_{10} = 100111_2$ 。

#### 0.1.4 十六进制系统

基数为 16 的数在计算机术语中称为十六进制 (hexadecimal), 它是一种用来表示二进制的便捷方法。例如, 对于人类来说, 使用 896H 来代表二进制数比使用 100010010110 来代表二进制数就简单得多。二进制数系统有两个数字 0 和 1; 十进制数系统有十个数字 0~9; 而十六进制数系统就有 16 个数字。在十六进制数中, 前面 10 个数 0~9 与十进制数是一样的, 而余下的 6 个数分别使用字母 A、B、C、D、E 和 F 代替。表 0-1 给出了分别表示 0~15 的十进

制数、二进制数以及十六进制数。

表 0-1 十六进制系统

| 十进制 | 二进制  | 十六进制 | 十进制 | 二进制  | 十六进制 |
|-----|------|------|-----|------|------|
| 0   | 0000 | 0    | 8   | 1000 | 8    |
| 1   | 0001 | 1    | 9   | 1001 | 9    |
| 2   | 0010 | 2    | 10  | 1010 | A    |
| 3   | 0011 | 3    | 11  | 1011 | B    |
| 4   | 0100 | 4    | 12  | 1100 | C    |
| 5   | 0101 | 5    | 13  | 1101 | D    |
| 6   | 0110 | 6    | 14  | 1110 | E    |
| 7   | 0111 | 7    | 15  | 1111 | F    |

### 0.1.5 二进制转换成十六进制

将二进制数以十六进制数表示,需从右边开始,4位一组分开,然后每4位以如表0-1所示的十六进制数表示。如果要将十六进制数转换成二进制数,只需将每位十六进制数用4位二进制数表示,如例0-4和例0-5所示。

**例0-4** 使用十六进制数表示二进制数 100111110101。

解:首先,将数每4位一组分开:1001 1111 0101。然后每组用其对应的十六进制表示:

1001 1111 0101  
9 F 5

所以,  $100111110101_2 = 9F5$  (十六进制)。

**例0-5** 将十六进制数 29B 转换成二进制数。

解: 2 9 B  
29B = 0010 1001 1011

将最前面的0去掉得到 1010011011。

### 0.1.6 十进制转换成十六进制

十进制转换成十六进制有两种方法。

1. 先转换成二进制再转换成十六进制。例0-6所示为此方法。
2. 通过多次除以16保存余数的方法直接转换成十六进制。这种方法请读者自行练习。

**例0-6** (a) 将  $45_{10}$  转换成十六进制数。

$\begin{array}{r} 32 \\ 1 \end{array}$   $\begin{array}{r} 16 \\ 0 \end{array}$   $\begin{array}{r} 8 \\ 1 \end{array}$   $\begin{array}{r} 4 \\ 1 \end{array}$   $\begin{array}{r} 2 \\ 0 \end{array}$   $\begin{array}{r} 1 \\ 1 \end{array}$  先转换成二进制。  
 $32 + 8 + 4 + 1 = 45$

$45_{10} = 0010\ 1101_2 = 2D$  (十六进制)

(b) 将  $629_{10}$  转换成十六进制数。

$\begin{array}{r} 512 \\ 1 \end{array}$   $\begin{array}{r} 256 \\ 0 \end{array}$   $\begin{array}{r} 128 \\ 0 \end{array}$   $\begin{array}{r} 64 \\ 1 \end{array}$   $\begin{array}{r} 32 \\ 1 \end{array}$   $\begin{array}{r} 16 \\ 1 \end{array}$   $\begin{array}{r} 8 \\ 0 \end{array}$   $\begin{array}{r} 4 \\ 1 \end{array}$   $\begin{array}{r} 2 \\ 0 \end{array}$   $\begin{array}{r} 1 \\ 1 \end{array}$

$629_{10} = (512 + 64 + 32 + 16 + 4 + 1) = 0010\ 0111\ 0101_2 = 275$  (十六进制)

(c) 将  $1714_{10}$  转换成十六进制数。

$$\begin{array}{r} \underline{1024} \quad \underline{512} \quad \underline{256} \quad \underline{128} \quad \underline{64} \quad \underline{32} \quad \underline{16} \quad \underline{8} \quad \underline{4} \quad \underline{2} \quad \underline{1} \\ 1 \quad 1 \quad 0 \quad 1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0 \\ 1714_{10} = (1024 + 512 + 128 + 32 + 16 + 2) = 0110 \ 1011 \ 0010_2 = 6B2_{16} \text{ (十六进制)} \end{array}$$

### 0.1.7 将十六进制转换成十进制

将十六进制转换成十进制也有两种方法。

1. 先将十六进制转换成二进制，再转换成十进制。例 0-7 对这一方法进行了演示。
2. 通过相加所有数的权直接将十六进制转换成十进制。

**例 0-7** 将下面十六进制数转换成十进制数。

(a)  $6B2_{16} = 0110 \ 1011 \ 0010_2$

$$\begin{array}{r} \underline{1024} \quad \underline{512} \quad \underline{256} \quad \underline{128} \quad \underline{64} \quad \underline{32} \quad \underline{16} \quad \underline{8} \quad \underline{4} \quad \underline{2} \quad \underline{1} \\ 1 \quad 1 \quad 0 \quad 1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0 \\ 1024 + 512 + 128 + 32 + 16 + 2 = 1714_{10} \end{array}$$

(b)  $9F2D_{16} = 1001 \ 1111 \ 0010 \ 1101_2$

$$\begin{array}{r} \underline{32768} \quad \underline{16384} \quad \underline{8192} \quad \underline{4096} \quad \underline{2048} \quad \underline{1024} \quad \underline{512} \quad \underline{256} \quad \underline{128} \quad \underline{64} \quad \underline{32} \quad \underline{16} \quad \underline{8} \quad \underline{4} \quad \underline{2} \quad \underline{1} \\ 1 \quad 0 \quad 0 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 1 \\ 32768 + 4096 + 2048 + 1024 + 512 + 256 + 32 + 8 + 4 + 1 = 40749_{10} \end{array}$$

### 0.1.8 使用十进制、二进制和十六进制计数

为展示三个数制之间的关系，表 0-2 给出了 0 ~ 31 的十进制数，并使用了二进制数与十六进制数表示。注意，不管使用哪种数制，当最大的数字再加 1 时就变成 0 且会向更高位进一位，例如，十进制数  $9 + 1 = 0$  就会得到进位，并且进位到更高位中；二进制中  $1 + 1 = 0$  也带有进位；同样十六进制中  $F + 1 = 0$  也带有进位。

表 0-2 使用数制计数

| 十进制 | 二进制   | 十六进制 | 十进制 | 二进制   | 十六进制 | 十进制 | 二进制   | 十六进制 |
|-----|-------|------|-----|-------|------|-----|-------|------|
| 0   | 00000 | 0    | 11  | 01011 | B    | 22  | 10110 | 16   |
| 1   | 00001 | 1    | 12  | 01100 | C    | 23  | 10111 | 17   |
| 2   | 00010 | 2    | 13  | 01101 | D    | 24  | 11000 | 18   |
| 3   | 00011 | 3    | 14  | 01110 | E    | 25  | 11001 | 19   |
| 4   | 00100 | 4    | 15  | 01111 | F    | 26  | 11010 | 1A   |
| 5   | 00101 | 5    | 16  | 10000 | 10   | 27  | 11011 | 1B   |
| 6   | 00110 | 6    | 17  | 10001 | 11   | 28  | 11100 | 1C   |
| 7   | 00111 | 7    | 18  | 10010 | 12   | 29  | 11101 | 1D   |
| 8   | 01000 | 8    | 19  | 10011 | 13   | 30  | 11110 | 1E   |
| 9   | 01001 | 9    | 20  | 10100 | 14   | 31  | 11111 | 1F   |
| 10  | 01010 | A    | 21  | 10101 | 15   |     |       |      |

### 0.1.9 二进制数以及十六进制数的加法

二进制数的加法很简单。表0-3中所示为两位数加法。二进制数的减法本书不再讲解，因为在所有的计算机中都是使用加法来完成减法的。计算机中有加法电路，但却没有减法电路，而将加法器与补码电路一起使用可以实现减法。换句话说，为了实现  $x - y$ ，计算机就必须得到  $y$  的补码，然后再加上  $x$ 。接下来复习补码的概念。例0-8所示为二进制加法。

表 0-3 二进制加法

| A + B | 进位 | 和 |
|-------|----|---|
| 0 + 0 | 0  | 0 |
| 0 + 1 | 0  | 1 |
| 1 + 0 | 0  | 1 |
| 1 + 1 | 1  | 0 |

**例 0-8** 将下列二进制数相加，并与十进制等式进行对照。

解：

|        |     |
|--------|-----|
| 二进制    | 十进制 |
| 1101   | 13  |
| + 1001 | 9   |
| 10110  | 22  |

### 0.1.10 补码

取二进制数的补码，就是先将所有数取反然后再加1。取反所有的位就是简单地将所有的0改成1，将所有的1改成0，这样所得到的二进制数称为反码（complement）。参见例0-9。

### 0.1.11 十六进制数的加法和减法

在学习计算机软硬件时，常会加减十六进制数。

**例 0-9** 求二进制数10011101的补码。

解：

|          |      |
|----------|------|
| 10011101 | 二进制数 |
| 01100010 | 反码   |
| + 1      |      |
| 01100011 | 补码   |

掌握这些技术很重要，随后讨论十六进制加法和减法。

### 0.1.12 十六进制数的加法

本节描述十六进制数的加法，从最低有效位开始，将数字相加，如果结果小于16，就将数字写入相应位置；如果数字大于16，则从数字中减去16得到余下的值，然后进一位，如例0-10所示。

**例 0-10** 完成十六进制数的加法：23D9 + 94BE。

解：

|        |                  |                  |
|--------|------------------|------------------|
| 23D9   | LSD: 9 + 14 = 23 | 23 - 16 = 7 带有进位 |
| + 94BE | 1 + 13 + 11 = 25 | 25 - 16 = 9 带有进位 |
| - B897 | 1 + 3 + 4 = 8    |                  |
|        | MSD: 2 + 9 = B   |                  |

### 0.1.13 十六进制数的减法

两个十六进制数相减，如果第二个数比第一个数大，则从前一位数中借一位，也就是 16，如例 0-11 所示。

**例 0-11** 完成十六进制数的减法：59F - 2B8。

解： 59F    LSD:    8 来自 15 = 7  
     -2B8            11 来自 25 (9 + 16) = 14 (E)  
     2E7            2 来自 4 (5 - 1) = 2

### 0.1.14 ASCII 码

目前的讨论围绕着数制系统展开。计算机中的所有信息都必须由 0 和 1 表示。

字母和其他字符也需要用二进制表示。在 18 世纪 60 年代，一种标准的表示方法称为 ASCII 码（美国信息交换标准码）出现了。ASCII（发音为“ask-E”）码以二进制表示数字 0~9、所有的英文字母（包括大写和小写）以及许多的控制码和标点符号。这个系统最大的优势是它为众多计算机所使用，信息能在计算机和计算机中共享。ASCII 系统共使用 7 位表示每个符号。例如 100 0001 分配给大写字母“A”，而 110 0001 分配给小写字母“a”。通常情况下，0 放在最高有效位，使得整个码是 8 位。图 0-1 所示为 ASCII 码节选，附录 F 中提供了完整的 ASCII 码列表。ASCII 是一个标准，它不仅使用在美国和其他一些国家的键盘中，而且在打印和显示字符的装置中，如输出装置，打印机和监视器中也广泛使用着它。

| 十六进制 | 符号  | 十六进制 | 符号  |
|------|-----|------|-----|
| 41   | A   | 61   | a   |
| 42   | B   | 62   | b   |
| 43   | C   | 63   | c   |
| 44   | D   | 64   | d   |
| ...  | ... | ...  | ... |
| 59   | Y   | 79   | y   |
| 5A   | Z   | 7A   | z   |

图 0-1 ASCII 码节选

注意，ASCII 码是为了让 ASCII 数据的操作尽量简单而设计的。如数字 0~9 在 ASCII 中用 30~39 来表示，这使一个程序可轻松将 ASCII 转换成十进制，只要将高位的半字节的“3”去掉就行。另外，大写字母与小写字母之间也是有关系的，大写字母由 ASCII 码 41~5A 表示，而小写字母则由 ASCII 码 61~7A 表示。通过观察二进制码可以发现，大写字母“A”与小写字母“a”之间的区别是第 5 个比特位。所以，大写字母与小写字母的转换就是简单地改变 ASCII 码的第 5 个比特位就可。

### 复习题

1. 为什么计算机使用二进制数系统而不使用十进制数系统？
2. 将  $34_{10}$  转换成二进制数和十六进制数。
3. 将  $110101_2$  转换成十六进制数和十进制数。
4. 完成二进制数的加法： $101100 + 101$ 。
5. 将  $101100_2$  用它的补码表示。
6. 对  $36BH + F6H$  求和。
7. 求  $36BH - F6H$  的差。
8. 将“80x86 CPU”写为 ASCII 码（用十六进制形式）。

## 0.2 数字基础知识

本节综述数字逻辑及其设计。首先介绍二进制逻辑运算，然后展示完成这些逻辑操作采用的门，接着将逻辑门放在一起组成简单的逻辑电路，最后讨论微控制器连接中的一些常用逻辑装置。

### 0.2.1 二进制逻辑

如前所述，计算机采用二进制数系统，因为两种电压等级能分别代表数字0和1。信号在数字电路中有两种不同的电压，如定义0 V为逻辑0，而+5 V为逻辑1。图0-2所示为对电压变化有一定容限的系统，该例中的有效数字信号位于两个阴影区域内。

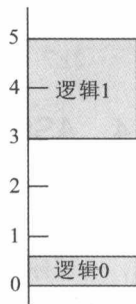


图0-2 二进制信号

### 0.2.2 逻辑门

二进制逻辑门是指具有一个或多个输入信号，而只有一个输出信号的简单电路，下面是一些门的定义。

| 逻辑与功能 |   |         |
|-------|---|---------|
| 输入    |   | 输出      |
| X     | Y | X AND Y |
| 0     | 0 | 0       |
| 0     | 1 | 0       |
| 1     | 0 | 0       |
| 1     | 1 | 1       |



#### 与门

与门（AND）采集一个或多个输入进行逻辑与。参见与门的真值表以及功能图，从中可以注意到，只有当与门的输入都是1的时候输出才会是1，任何其他的组合都会使结果为0。本例所示为两个输入 $x$ 和 $y$ 的情况，逻辑门也可以有多个输入，在与门的情况下，如果输入都是1，则输出就是1；如果输入中有一个是0，则输出就是0。

| 逻辑或功能 |   |        |
|-------|---|--------|
| 输入    |   | 输出     |
| X     | Y | X OR Y |
| 0     | 0 | 0      |
| 0     | 1 | 1      |
| 1     | 0 | 1      |
| 1     | 1 | 1      |

