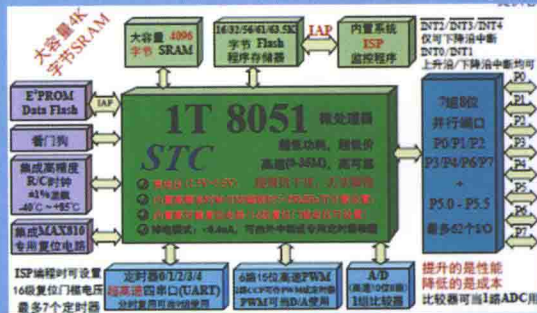


新编高等教育电子信息类规划教材  
STC15系列单片机丛书

# 单片微机原理与接口技术

## ——基于STC15W4K32S4系列单片机

丁向荣 陈崇辉 编 著  
姚永平 主 审



### 教材内容与特色

- ◆ 选用STC最新技术、功能最齐全的STC15W4K32S4系列单片机为教学机型
- ◆ 基于STC官方STC15学习板开发教学例程
- ◆ STC单片机大学推广计划合作教材
- ◆ 向选用教材学校免费提供教学课件、实验实训指导与教学例程程序代码

 中国工信出版集团

 电子工业出版社  
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY  
<http://www.phei.com.cn>

新编高等教育电子信息类规划教材

STC15 系列单片机丛书

# 单片微机原理与接口技术 ——基于 STC15W4K32S4 系列单片机

丁向荣 编著

陈崇辉

姚永平 主审

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

## 内 容 简 介

STC15W4K32S4 系列单片机是 STC 增强型 8051 单片机最新技术结晶, 宽电源电压 (2.4~5.5V), 无须转换芯片可直接与 PC 机 USB 接口进行通信; 增强型 8051 单片机集成了上电复位电路与高精度 R/C 振荡器, 给单片机芯片加上电源就可跑程序; 可在线编程与在线仿真, 一颗芯片既是目标芯片, 又是仿真芯片; 集成了大容量的程序存储器、数据存储器以及 EEPROM, 增加了定时器、串行口等基本功能部件, 集成了 A/D、PCA、比较器、专用 PWM 模块、SPI 等高功能接口部件, 可大大简化单片机应用系统的外围电路, 使单片机应用系统的设计更加简捷, 系统性能更加高效、可靠。本教材以 STC15W4K32S4 系列中的 IAP15W4K58S4 单片机为主线, 系统地介绍了 IAP15W4K58S4 单片机的硬件结构、指令系统与应用编程, 系统地介绍了单片机应用系统的开发流程与接口设计, 基于宏晶科技大学推广计划开发此教材。

本书可作为普通高校计算机类、电子信息类、电气自动化与机电一体化等专业的教学用书, 基础较好的高职高专也可选用本书。此外, 可作为电子设计竞赛、电子设计工程师考证的培训教材。也是传统 8051 单片机应用工程师升级转型的最新参考书籍。

未经许可, 不得以任何方式复制或抄袭本书之部分或全部内容。  
版权所有, 侵权必究。

### 图书在版编目 (CIP) 数据

单片微机原理与接口技术: 基于 STC15W4K32S4 系列单片机 / 丁向荣, 陈崇辉编著. —北京: 电子工业出版社, 2015.5

(新编高等教育电子信息类规划教材. STC15 系列单片机丛书)

ISBN 978-7-121-25832-9

I. ①单… II. ①丁… ②陈… III. ①单片微型计算机—基础理论—高等学校—教材②单片微型计算机—接口—高等学校—教材 IV. ①TP368.1

中国版本图书馆 CIP 数据核字 (2015) 第 071249 号

策 划: 陈晓明

责任编辑: 郭乃明 特约编辑: 范 丽

印 刷: 三河市华成印务有限公司

装 订: 三河市华成印务有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×1092 1/16 印张: 25.75 字数: 659 千字

· 版 次: 2015 年 5 月第 1 版

印 次: 2015 年 5 月第 1 次印刷

印 数: 3 000 册 定价: 55.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 [zltz@phei.com.cn](mailto:zltz@phei.com.cn), 盗版侵权举报请发邮件至 [dbqq@phei.com.cn](mailto:dbqq@phei.com.cn)。

服务热线: (010) 88258888。

# 前 言

STC 系列单片机传承于 Intel8051 单片机，但在传统 8051 单片机框架基础上注入了新鲜血液，焕发出新的“青春”。STC 宏晶科技对 8051 单片机进行了全面的技术升级与创新：全部采用 Flash 技术（可反复编程 10 万次以上）和 ISP/IAP（在系统可编程/在应用可编程）技术；针对抗干扰进行了专门设计，超强抗干扰；进行了特别加密设计，如宏晶 STC15 系列单片机现无法解密；对传统 8051 进行了全面提速，指令速度最快提高了 24 倍；大幅提高了集成度，如集成了 A/D、CCP/PCA/PWM（PWM 还可当 D/A 使用）、高速同步串行通信端口 SPI、高速异步串行通信端口 UART（如宏晶 STC15W4K58S4 系列集成了 4 个串行口）、定时器（STC15W4K58S4 系列最多可实现 7 个定时器）、看门狗、内部高精度时钟（ $\pm 1\%$ 温飘， $-40^{\circ}\text{C}\sim+85^{\circ}\text{C}$ 之间，可彻底省掉外部昂贵的晶振）、内部高可靠复位电路（可彻底省掉外部复位电路）、大容量 SRAM（如 STC15W4K58S4 系列集成了 4K 字节的 SRAM）、大容量 EEPROM、大容量 Flash 程序存储器等。STC 单片机的在线下载编程和在线仿真功能以及分系列的资源配置，增加了单片机型号的选择性，可根据单片机应用系统的功能要求选择合适的单片机，从而降低单片机应用系统的开发难度与开发成本，使得单片机应用系统更加简单、高效，提高了单片机应用产品的性能价格比。

STC 作为中国本土 MCU 的领航者，从 2006 年诞生起，现已发展了 STC89/90 系列、STC10/11 系列、STC12 系列、STC15 系列。2014 年 4 月，宏晶科技重磅推出了 STC15W4K32S4 系列单片机，宽电源电压范围，能在 2.4~5.5V 电压范围内工作；无须转换芯片，STC15W4K32S4 单片机可直接与 PC 机的 USB 接口相连进行通信；集成了更多的数据存储器、定时/计数器以及串行口；集成了更多的高功能部件（如比较器、专用 PWM 模块）；开发了功能强大的 STC-ISP 在线编程软件工具，除可进行在线编程以外，还包括在线仿真器制作、脱机编程工具的制作、加密传输、项目发布、各系列单片机头文件的生成、串行口波特率的计算、定时器定时程序的设计、软件延时程序的设计等工具，给学习者或单片机应用的生产者带来了极大的便捷与高效。

本教材选用 STC 最新系列 STC15W4K32S4 系列的 IAP15W4K58S4 单片机作为主讲机型，系统地介绍了 IAP15W4K58S4 单片机的硬件结构、指令系统与应用编程。

STC 系列单片机的指令系统和标准的 8051 内核完全兼容，因此，原来讲解 8051 单片机的师资力量可以充分发挥以前讲解单片机原理及应用课程的经验；对于具有 8051 单片机知识的读者，也不存在转型困难的问题。

教材力求实用性、应用性与易学性，以提高读者的工程设计能力与实践动手能力为目标。本书具有以下几方面的特点：

(1) 单片机机型贴近生产实际：STC 单片机是我国 8 位单片机应用中市场占有率最高的，更难能可贵的是，STC 单片机是我国本土的 MCU。

(2) 采用“双”语言编程：在绝大多数应用程序的编程中，采用汇编语言和 C 语言（C51）对照编程。采用汇编语言程序学习更有利于加强对单片机的理解；而 C51 在功能、结构上以及可读性、可移植性、可维护性都有非常明显的优势。

(3) 理论联系实际, 在学习单片机指令系统前的第 3 章就专门介绍了单片机应用的开发工具, 贯穿程序的编辑、编译、下载与调试。强化单片机知识的应用性与实践性, 不论是一条指令, 或若干条指令, 或一个程序段都可以用开发工具进行仿真调试或在线联机调试。

(4) 强化单片机应用系统的概念, 学习单片机就是为了能开发与制作有具体意义的单片机应用系统, 第 15 章介绍了单片机基本的外围接口技术与典型单片机应用系统的设计与开发。

(5) 在本教材的编写中, 直接与 STC 单片机的创始人姚永平先生进行密切沟通与交流, 姚永平先生亲自担任本教材的主审, 确保了教材内容的系统性与正确性。

(6) 本教材例题基于 STC 官方 STC15 学习板开发, 本教材是宏晶科技 STC 单片机大学推广计划的合作教材, 也是全国信息技术应用水平大赛“STC”杯单片机系统设计大赛的推荐用书。

本书由丁向荣、陈崇辉编著, 其中, 第 1 章~第 10 章、附录 1~附录 7 由丁向荣编写, 第 11~第 15 章由陈崇辉编写。深圳宏晶科技有限公司技术部工程人员在技术上给予了大力支持和帮助, STC 单片机创始人姚永平先生担任主审, 对全书进行了认真审阅, 并提出了宝贵意见。在此。对所有提供帮助的人表示感谢!

由于编者水平有限, 书中定有疏漏和不妥之处, 敬请读者不吝指正! 书中相关勘误或信息也会动态地公布在 STC 官网上: [www.stcmcu.com](http://www.stcmcu.com)。您有什么建议, 以及教学中需要程序源代码的, 可发电子邮件到: [dingxiangrong65@163.com](mailto:dingxiangrong65@163.com), 与作者进一步沟通与交流。

编 者

2014.12 于广州



## 序

21 世纪全球全面进入了计算机智能控制/计算/通信（物联网）时代，而其中的一个重要方向就是以单片机为代表的嵌入式计算机控制/计算。由于最适合中国工程师/学生入门的 8051 单片机已有 30 多年的应用历史，绝大部分工科院校均有此必修课，有几十万名对该单片机十分熟悉的工程师可以相互交流开发/学习心得，有大量的经典程序和电路可以直接套用，从而大幅降低了开发风险，极大地提高了开发效率，这也是 STC.宏晶科技/南通国芯微电子电子有限公司生产基于 8051 指令系列单片机产品的巨大优势。

Intel 8051 技术诞生于 20 世纪 70 年代，不可避免地面临着落伍的危险，如果不对其进行大规模创新，我国的单片机教学与应用就会陷入被动局面。为此，STC.宏晶科技对 8051 单片机进行了全面的技术升级与创新，经历了 STC89/90、STC10/11、STC12、STC15 系列，累计发布上百种产品：全部采用 Flash 技术（可反复编程 10 万次以上）和 ISP/IAP（在系统可编程/在应用可编程）技术；针对抗干扰进行了专门设计，超强抗干扰；进行了特别加密设计，如 STC15 系列现无法解密；对传统 8051 进行了全面提速，相同时钟频率指令平均快 7 倍，最快指令速度提高了 24 倍，时钟从传统的 12MHz 提高到最快可达 30MHz；大幅提高了集成度，如集成了 A/D、CCP/PCA/PWM（PWM 还可当 D/A 使用）、高速同步串行通信端口 SPI、4 个高速异步串行通信端口 UART、5 个定时器/计数器、看门狗、内部高精度时钟（ $\pm 1\%$  温飘， $-40^{\circ}\text{C}\sim+85^{\circ}\text{C}$  之间，可彻底省掉外部昂贵的晶振）、内部高可靠复位电路（可彻底省掉外部复位电路）、大容量 SRAM、大容量 EEPROM、大容量 Flash 程序存储器等。针对大学教学，现 STC15 系列一个单芯片就是一个仿真器（IAP15W4K58S4），定时器改造为支持 16 位自动重载（学生只需学一种模式），串行口通信波特率计算改造为[系统时钟/4/(65536-重装数)]，极大地简化了教学，针对实时操作系统 RTOS 推出了不可屏蔽的 16 位自动重载定时器（定时器 0 的模式 3），并且在最新的 STC-ISP 烧录软件中提供了大量的贴心工具，如范例程序/定时器计算器/软件延时计算器/波特率计算器/头文件/指令表/Keil 仿真设置等。

封装也从传统的 PDIP40 发展到 DIP8/DIP16/DIP20/SKDIP28，SOP8/SOP16/SOP20/SOP28，LQFP32/LQFP48/LQFP64S/LQFP64L，TSSOP20/TSSOP28，DFN8/QFN28/QFN32/QFN48/QFN64，芯片的 I/O 口从 6 个到 62 个不等，价格从 0.89 元到 5.9 元不等，极大地方便了客户选型和设计。

2014 年 4 月，STC 宏晶科技重磅推出了 STC15W4K32S4 系列单片机，宽电压工作范围，不需任何转换芯片，STC15W4K32S4 系列单片机可直接通过电脑 USB 接口进行 ISP 下载编程，集成了更多的 SRAM（4K 字节）、定时器 7 个（5 个普通定时器+CCP 定时器 2）、串口（4 个），集成了更多的高性能部件（如比较器、带死区控制的 6 路 15 位专用 PWM 等）；开发了功能强大的 STC-ISP 在线编程软件，包含了项目发布、脱机下载、RS-485 下载、程序加密后传输下载、下载需口令等功能，并已申请专利。

IAP15W4K58S4 一个芯片就是一个仿真器（OCD，ICE），是全球第一个实现一个芯片就可以仿真的（彻底抛弃了 J-Link/D-Link），一个仿真器售价仅 5.6 元，有 SOP28/SKDIP28/LQFP32/PDIP40/LQFP44/LQFP48/LQFP64S/LQFP64L 等封装型式。

## STC 大学计划

STC 全力支持我国的单片机/嵌入式系统教育事业，STC 大学计划正如火如荼地进行中，陆续开展向普通高等学校电子信息/自动化/物联网等相关专业赠送可仿真的 STC15 系列实验箱<仿真芯片 IAP15W4K58S4>，共建 STC 高性能单片机联合实验室，本教材为 STC 大学计划的合作教材，也是 STC 杯单片机系统设计大赛的推荐教材。

部分已建和在建的高校：上海交通大学、西安交通大学、浙江大学、武汉大学、华中科技大学、中山大学、吉林大学、山东大学、哈尔滨工业大学、天津大学、同济大学、湖南大学、兰州大学、东北大学、西北农林科技大学、中国海洋大学、北京航空航天大学、南京航空航天大学、北京理工大学、南京理工大学、华东理工大学、太原理工大学、东华理工大学、哈尔滨理工大学、哈尔滨工程大学、北京化工大学、北京工业大学、东华大学、苏州大学、江南大学、扬州大学、南通大学、宁波大学、深圳大学、杭州电子科技大学、桂林电子科技大学、西安电子科技大学、成都电子科技大学、华北电力大学、南京邮电大学、西安邮电大学、天津工业大学、中国石油大学、中国矿业大学等国内著名 985/211/及电类本科高校，以及广东轻工职业技术学院、深圳信息职业技术学院、深圳职业技术学院等著名的职业高校。

### 对大学计划与单片机教学的看法

STC 大学计划有步骤地向前推进中：第九届“STC 杯单片机系统设计大赛”刚成功落幕，全国数百所高校的近 1100 支队伍参赛；在国内上百所大学建立了联合实验室；上海交通大学、西安交通大学、浙江大学、山东大学、哈尔滨工业大学、成都电子科技大学等著名高校的多位知名教授也正在基于 STC 1T 8051 创作全新的教材。多所高校每年都有用 STC 单片机进行的全校创新竞赛，如杭州电子科技大学、湖南大学、山东大学等。

现在学校的学生到底应该先学 32 位的微控制器好还是 8 位的 8051 单片机好？我觉得应该是从 8 位的 8051 单片机入门比较合适。因为现在大学嵌入式课程一般只有 64 个学时，甚至只有 48 个学时，学生能把 8051 单片机学懂，真正能做出产品，工作以后就能触类旁通了。但如果也只给 48 个学时去学 ARM，学生不能完全学懂，最多只能搞些函数调用，没有意义，培养不出真正能动手的人才，嵌入式第一门课就将学生吓倒，可能他终生也不会再碰嵌入式开发了，所以我们要培养学生的信心，而不是唱高调，伤害了他们。所以大家反思说，大一大二还是应该先以 8 位单片机入门，大三时学有余力的学生再选修 32 位的嵌入式课程。C 语言要与 8051 单片机融合教学，大一第一学期就要开始学，现在有些中学的课外兴趣小组多在学 STC 的 8051 + C 语言。工科非计算机专业的学生不要在大一时全是数学、物理、英语...，学生一进大学的门，跟高中一样，又不知道自己本专业能干啥，就放羊啊！所以 C 语言和单片机要提前学。

### 对大学工科非计算机专业 C 语言教学的看法

再讲讲 C 语言，现在工科非计算机专业讲 C 语言的书大多是空中飘着，落不到地，学完之后不知道干什么。以前我们学 BASIC/C，学完用 DOS 系统，也在 DOS 下开发软件。现在学生学完 C，要从 Windows 去返回 DOS 运行，学的 C 也不能在 8051 上运行。嵌入式 C 语言有多个版本，国内 Keil C 流行，现我们也在开发我们中国人自己的 C 编译器。现在学标

准 C 语言，没办法落地了，学完了，PC 上干不了事，单片机上也动不了。我们现在推教学改革将单片机和 C 语言（嵌入式 C，面向控制的 C）放在一门课中讲，在大一的第一学期就讲，学生学完后就知道他将来能干啥了，大二的第二学期再开一门 Windows 下的 C++ 开发，正好我们的单片机 C 语言给他打基础。学生学完模电/数电（FPGA）/数据结构/RTOS（实时操作系统）/传感器原理/自动控制原理/数字信号处理等后，在大三再开一门综合电子系统设计，这样就能循序渐进地培养出真正能动手的人才了。

我们现在主要的工作是推动中国工科非计算机专业高校教学改革，研究成果的具体化，就是大量高校创新教材的推出。丁向荣老师编写的这本教材，就是我们高校教学改革研究成果的具体体现。希望能在我们这一代人的努力下，让我们中国的嵌入式单片机系统设计全球领先。

感谢 Intel 公司发明了经久不衰的 8051 体系结构，感谢丁向荣老师的新书，保证了中国 30 年来的单片机教学与世界同步。

我们将本教材确定为 STC 大学计划推荐教材、STC 单片机系统设计大赛推荐教材，采用本书作为教材的高校将优先免费获得我们可仿真的 STC15 系列实验箱的支持<主控芯片 IAP15W4K58S4>。

最后，为了中华民族的伟大复兴，让我们一起——明知山有虎，偏向虎山行！

STC MCU Limited: Andy.姚

[www.STCMCU.com](http://www.STCMCU.com) [www.GXWMCU.com](http://www.GXWMCU.com)

2015/4/15



# 目 录

<b>第 1 章 微型计算机基础</b> .....	(1)
1.1 数制与编码 .....	(1)
1.1.1 数制及转换方法 .....	(1)
1.1.2 微型计算机中数的表示方法 .....	(3)
1.1.3 微型计算机中常用编码 .....	(5)
1.2 微型计算机原理 .....	(6)
1.2.1 微型计算机的基本组成 .....	(6)
1.2.2 指令、程序与编程语言 .....	(8)
1.2.3 微型计算机的工作过程 .....	(8)
1.2.4 微型计算机的应用形态 .....	(9)
本章小结 .....	(11)
习题 1 .....	(11)
<b>第 2 章 IAP15W4K58S4 单片机增强型 8051 内核</b> .....	(12)
2.1 单片机概述 .....	(12)
2.1.1 单片机的概念 .....	(12)
2.1.2 常见单片机 .....	(12)
2.1.3 STC15W4K32S4 系列单片机 .....	(13)
2.2 IAP15W4K58S4 系列单片机的引脚功能 .....	(16)
2.3 IAP15W4K58S4 单片机的内部结构 .....	(20)
2.3.1 IAP15W4K58S4 单片机的内部结构 .....	(20)
2.3.2 CPU 结构 .....	(21)
2.4 IAP15W4K58S4 单片机的存储结构 .....	(22)
2.5 IAP15W4K58S4 单片机的并行 I/O 口 .....	(26)
2.5.1 IAP15W4K58S4 单片机的并行 I/O 口与工作模式 .....	(26)
2.5.2 IAP15W4K58S4 单片机的并行 I/O 口的结构 .....	(27)
2.5.3 IAP15W4K58S4 单片机并行 I/O 口的使用注意事项 .....	(29)
2.6 IAP15W4K58S4 单片机的时钟与复位 .....	(30)
2.6.1 IAP15W4K58S4 单片机的时钟 .....	(30)
2.6.2 IAP15W4K58S4 单片机的复位 .....	(32)
本章小结 .....	(34)
习题 2 .....	(35)
<b>第 3 章 IAP15W4K58S4 单片机的在线编程与在线仿真</b> .....	(36)
3.1 Keil $\mu$ Vision4 集成开发环境 .....	(36)
3.1.1 概述 .....	(36)
3.1.2 应用 Keil $\mu$ Vision4 开发工具编辑、编译用户程序, 生成机器代码 .....	(37)

3.1.3	应用 Keil $\mu$ Vision4 集成开发环境调试用户程序	(44)
3.2	STC 系列单片机在线编程与在线仿真	(48)
3.2.1	STC 系列单片机在线可编程 (ISP) 电路	(48)
3.2.2	单片机应用程序的下载与运行	(50)
3.2.3	Keil $\mu$ Vision4 与 STC 仿真器的在线仿真	(52)
3.2.4	STC-ISP 在线编程软件的其他功能	(54)
	本章小结	(55)
	习题 3	(55)
<b>第 4 章</b>	<b>IAP15W4K58S4 单片机的指令系统与汇编语言程序设计</b>	<b>(57)</b>
4.1	IAP15W4K58S4 单片机的指令系统	(57)
4.1.1	概述	(57)
4.1.2	数据传送类指令 (29 条)	(62)
4.1.3	算术运算类指令 (24 条)	(68)
4.1.4	逻辑运算类与循环移位类指令 (24 条)	(74)
4.1.5	控制转移类指令 (17 条)	(77)
4.1.6	位操作类指令 (17 条)	(83)
4.2	汇编语言程序设计	(88)
4.2.1	汇编语言程序设计基础	(88)
4.2.2	基本程序结构与程序设计举例	(93)
	本章小结	(102)
	习题 4	(102)
<b>第 5 章</b>	<b>C51 与 C51 程序设计</b>	<b>(106)</b>
5.1	C51 基础	(106)
5.1.1	C51 数据类型	(108)
5.1.2	C51 的变量	(110)
5.1.3	8051 单片机特殊功能寄存器变量的定义	(111)
5.1.4	8051 单片机位寻址区 (20H~2FH) 位变量的定义	(112)
5.1.5	函数的定位	(113)
5.1.6	中断服务函数	(113)
5.1.7	函数的递归调用与再入函数	(114)
5.1.8	在 C51 中嵌入汇编	(115)
5.2	C51 程序设计	(115)
5.2.1	C51 程序框架	(115)
5.2.2	C51 程序设计举例	(119)
	本章小结	(121)
	习题 5	(121)
<b>第 6 章</b>	<b>IAP15W4K58S4 单片机的存储器与应用编程</b>	<b>(123)</b>
6.1	IAP15W4K58S4 单片机的程序存储器	(123)
6.2	IAP15W4K58S4 单片机的基本 RAM	(125)

6.3	IAP15W4K58S4 单片机的扩展 RAM (XRAM)	(126)
6.4	IAP15W4K58S4 单片机的 EEPROM (数据 Flash)	(130)
	本章小结	(138)
	习题 6	(139)
<b>第 7 章</b>	<b>IAP15W4K58S4 单片机的定时/计数器</b>	<b>(140)</b>
7.1	IAP15W4K58S4 单片机定时/计数器 (T0/T1) 的结构和工作原理	(140)
7.2	IAP15W4K58S4 单片机定时/计数器 (T0/T1) 的控制	(141)
7.3	IAP15W4K58S4 单片机定时/计数器 (T0/T1) 的工作方式	(143)
7.4	IAP15W4K58S4 单片机定时/计数器 (T0/T1) 的应用举例	(147)
7.4.1	IAP15W4K58S4 单片机定时/计数器 (T0/T1) 的定时应用	(147)
7.4.2	IAP15W4K58S4 单片机定时/计数器 (T0/T1) 的计数应用	(149)
7.4.3	T0、T1 的综合应用	(150)
7.5	IAP15W4K58S4 单片机的定时器 T2	(155)
7.5.1	IAP15W4K58S4 单片机的定时器 T2 的电路结构	(155)
7.5.2	IAP15W4K58S4 单片机的定时/计数器 T2 的控制寄存器	(156)
7.6	IAP15W4K58S4 单片机的定时器 T3、T4	(157)
7.6.1	IAP15W4K58S4 单片机的定时器 T3、T4 的电路结构	(157)
7.6.2	IAP15W4K58S4 单片机的定时/计数器 T3、T4 的控制寄存器	(157)
7.7	IAP15W4K58S4 单片机的可编程时钟输出功能	(158)
7.7.1	IAP15W4K58S4 单片机 T0、T1、T2、T3、T4 的可编程时钟输出	(159)
7.7.2	IAP15W4K58S4 单片机可编程时钟的应用举例	(160)
	本章小结	(161)
	习题 7	(162)
<b>第 8 章</b>	<b>IAP15W4K58S4 单片机中断系统</b>	<b>(163)</b>
8.1	中断系统概述	(163)
8.1.1	中断系统的几个概念	(163)
8.1.2	中断的技术优势	(164)
8.1.3	中断系统需要解决的问题	(164)
8.2	IAP15W4K58S4 单片机的中断系统	(165)
8.2.1	IAP15W4K58S4 单片机的中断请求	(165)
8.2.2	IAP15W4K58S4 单片机的中断响应	(172)
8.2.3	IAP15W4K58S4 单片机中断应用举例	(175)
8.3	IAP15W4K58S4 单片机外部中断的扩展	(180)
	本章小结	(183)
	习题 8	(183)
<b>第 9 章</b>	<b>IAP15W4K58S4 单片机的串行口</b>	<b>(185)</b>
9.1	串行通信基础	(185)
9.2	IAP15W4K58S4 单片机的串行口 1	(188)
9.2.1	串行口 1 的控制寄存器	(188)

9.2.2	串行口 1 的工作方式 .....	(190)
9.2.3	串行口 1 的波特率 .....	(194)
9.2.4	串行口 1 的应用举例 .....	(195)
9.3	IAP15W4K58S4 单片机与 PC 机的通信 .....	(208)
9.3.1	IAP15W4K58S4 单片机与 PC 机 RS-232 串行通信的接口设计 .....	(208)
9.3.2	IAP15W4K58S4 单片机与 PC 机 USB 总线通信的接口设计 .....	(211)
9.3.3	IAP15W4K58S4 单片机与 PC 机串行通信的程序设计 .....	(211)
9.4	IAP15W4K58S4 单片机串行口 1 的中继广播方式 .....	(214)
9.5	IAP15W4K58S4 单片机串行口 2 .....	(214)
9.6	IAP15W4K58S4 单片机串行口 3 .....	(216)
9.7	IAP15W4K58S4 单片机串行口 4 .....	(218)
9.8	IAP15W4K58S4 单片机串行口功能引脚的切换 .....	(219)
	本章小结 .....	(220)
	习题 9 .....	(221)
<b>第 10 章</b>	<b>IAP15W4K58S4 单片机比较器</b> .....	(222)
10.1	IAP15W4K58S4 单片机比较器的内部结构与控制 .....	(222)
10.2	IAP15W4K58S4 单片机比较器的应用 .....	(224)
	本章小结 .....	(227)
	习题 10 .....	(228)
<b>第 11 章</b>	<b>IAP15W4K58S4 单片机的 A/D 转换模块</b> .....	(229)
11.1	IAP15W4K58S4 单片机 A/D 模块的结构 .....	(229)
11.2	IAP15W4K58S4 单片机 A/D 模块的控制 .....	(230)
11.3	IAP15W4K58S4 单片机 A/D 转换的应用 .....	(233)
	本章小结 .....	(238)
	习题 11 .....	(238)
<b>第 12 章</b>	<b>IAP15W4K58S4 单片机的 PCA 模块</b> .....	(239)
12.1	IAP15W4K58S4 单片机 PCA 模块的结构与控制 .....	(239)
12.2	IAP15W4K58S4 单片机 PCA 模块的工作模式与应用编程 .....	(243)
12.2.1	捕获模式与应用编程 .....	(243)
12.2.2	16 位软件定时器模式与应用编程 .....	(245)
12.2.3	高速输出模式与应用编程 .....	(247)
12.2.4	脉宽调制 (PWM) 模式与应用编程 .....	(249)
12.3	IAP15W4K58S4 单片机 PCA 模块功能引脚的切换 .....	(253)
	本章小结 .....	(254)
	习题 12 .....	(254)
<b>第 13 章</b>	<b>IAP15W4K58S4 单片机的 PWM 模块</b> .....	(255)
13.1	IAP15W4K58S4 单片机 PWM 模块的结构与控制 .....	(255)
13.2	IAP15W4K58S4 单片机 PWM 模块的应用编程 .....	(260)
13.3	IAP15W4K58S4 单片机 PWM 模块功能引脚的切换 .....	(266)

本章小结	(266)
习题 13	(267)
<b>第 14 章 IAP15W4K58S4 单片机的 SPI 接口</b>	(268)
14.1 IAP15W4K58S4 单片机的 SPI 接口的结构与控制	(268)
14.2 IAP15W4K58S4 单片机的 SPI 接口的数据通信	(271)
14.3 IAP15W4K58S4 单片机的 SPI 接口的应用编程	(275)
14.4 IAP15W4K58S4 单片机的 SPI 接口功能引脚的切换	(279)
本章小结	(280)
习题 14	(280)
<b>第 15 章 单片机应用系统的设计</b>	(281)
15.1 单片机应用系统的开发流程	(281)
15.1.1 单片机应用系统的设计原则	(281)
15.1.2 单片机应用系统的开发流程	(282)
15.1.3 单片机应用系统工程报告的编制	(285)
15.2 人机对话接口应用设计	(286)
15.2.1 键盘接口与应用编程	(286)
15.2.2 LED 数码显示与应用编程	(295)
15.2.3 LCD 显示接口与应用编程	(303)
15.3 串行总线接口技术与应用编程	(319)
15.3.1 I <sup>2</sup> C 串行总线接口技术与应用编程	(319)
15.3.2 单总线接口技术与应用编程	(335)
15.4 红外遥控技术与应用编程	(343)
15.4.1 红外遥控发射与接收	(343)
15.4.2 红外遥控应用编程	(346)
15.5 IAP15W4K58S4 单片机的低功耗设计与可靠性设计	(353)
15.5.1 IAP15W4K58S4 单片机的低功耗设计	(353)
15.5.2 IAP15W4K58S4 单片机的可靠性设计	(357)
本章小结	(360)
习题 15	(361)
<b>附录 1 ASCII 码表</b>	(362)
<b>附录 2 STC15W4K32S4 系列单片机指令系统表</b>	(363)
<b>附录 3 C51 常用头文件与库函数</b>	(367)
<b>附录 4 STC-ISP 在线编程软件实用程序简介</b>	(374)
<b>附录 5 STC15 单片机学习板各模块电路</b>	(377)
<b>附录 6 STC15 头文件与 LED 数码管驱动函数</b>	(384)
<b>附录 7 U8 脱机编程器的操作使用</b>	(396)
<b>参考文献</b>	(398)

# 第 1 章 微型计算机基础

## 1.1 数制与编码

数制与编码是微型计算机的基本数字逻辑基础，是学习微型计算机的必备知识。数制与编码的知识一般会在数字逻辑或计算机文化基础中学习，但往往由于数制与编码的知识，与当前课程的关系并非“不可或缺”，又比较枯燥。在微型计算机原理或单片机的教学中，教师普遍感觉到，学生在这方面的知识基础不扎实。在此，以提纲挈领的形式再理一理。

### 1.1.1 数制及转换方法

所谓数制就是计数的方法，通常采用进位计数制。在微型计算机的学习与应用中，主要有十进制、二进制和十六进制三种计数方法。日常生活采用的是十进制；微型计算机只能识别和处理数字信息，微型计算机硬件电路采用的是二进制，但为了更好地记忆与描述微型计算机的地址、程序代码以及运算数字，一般采用十六进制。

#### 1. 各种进位计数制及其表示方法（见表 1.1 所示）

表 1.1 二进制、十进制与十六进制的计数规则与表示方法

进位制	计数规则	基数	各位的权	数 码	权值展开式	表 示 法	
						后缀字符	下标
二进制	逢二进一 借一当二	2	$2^i$	0, 1	$(b_{n-1} \cdots b_1 b_0 b_{-1} \cdots b_{-m})_2 = \sum_{i=-m}^{n-1} b_i \times 2^i$	B	$( )_2$
十进制	逢十进一 借一当十	10	$10^i$	0, 1, 2, 3, 4, 5, 6, 7, 8, 9	$(d_{n-1} \cdots d_1 d_0 d_{-1} \cdots d_{-m})_{10} = \sum_{i=-m}^{n-1} d_i \times 10^i$	D 通常缺省表示	$( )_{10}$
十六进制	逢十六进一 借一当十六	16	$16^i$	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F	$(h_{n-1} \cdots h_1 h_0 h_{-1} \cdots h_{-m})_{16} = \sum_{i=-m}^{n-1} h_i \times 16^i$	H	$( )_{16}$

注：i 是各进制数码在数字中的位置，i 值是以小数点为界，往左依次为 0、1、2、3、…，往右依次为 -1、-1、-3、…。

#### 2. 数制之间的转换

任意进制之间相互转换，整数部分和小数部分都必须分别进行。各进制的相互转换关系如图 1.1 所示。

(1) 二进制、十六进制转换为十进制。将二进制、十六进制数按权值展开式展开相加所得数，即为十进制数。

(2) 十进制转换为二进制。十进制转二进制要分成整数部分与小数部分，而且其转换方法是完全不同的。



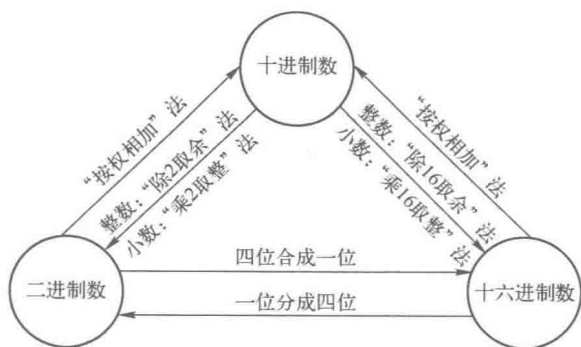


图 1.1 各进制的相互转换关系图

① 十进制整数部分转换成二进制——除 2 取余法，并倒序排列，如下所示：

2	84	余数	二进制数码
2	42	..... 0	b <sub>0</sub>
2	21	..... 0	b <sub>1</sub>
2	10	..... 1	b <sub>2</sub>
2	5	..... 0	b <sub>3</sub>
2	2	..... 1	b <sub>4</sub>
2	1	..... 0	b <sub>5</sub>
	0	..... 1	b <sub>6</sub>

所以  $(84)_{10} = (1010100)_2$

② 十进制小数转换成二进制小数——乘 2 取整法，如下所示：

	0.6875
	× 2
b <sub>-1</sub>	1 ← ..... 1.3750
	× 2
b <sub>-2</sub>	0 ← ..... 0.7500
	× 2
b <sub>-3</sub>	1 ← ..... 1.5000
	× 2
b <sub>-4</sub>	1 ← ..... 1.0000

所以  $(0.6875)_{10} = (0.1011)_2$

将上述两部分合起来，则有， $(84.6875)_{10} = (1010100.1011)_2$

(3) 二进制与十六进制互转。

① 二进制转十六进制。以小数点为界，往左、往右每 4 位二进制数为一组，每 4 位二进制数用 1 位十六进制数表示，往左高位不够用 0 补齐，往右低位不够用 0 补齐，例如，

$$(111101.011101)_2 = (0011 \ 1101.0111 \ 0100)_2 = (3D.74)_{16}$$

② 十六进制转二进制。每位十六进制数用四位二进制数表示，再将整数部分最高位的 0 去掉，小数部分最低位的 0 去掉，例如，

$$(3C20.84)_{16} = (0011 \ 1100 \ 0010 \ 0000.1000 \ 0100)_2 = (11110000100000.100001)_2$$

数制转换工具：利用 PC 机附件中的计算器（科学型）可实现各数制间的相互转换。单击任务栏“开始”按钮，选择“所有程序”→“附件”→“计算器”，即可打开计算器工具，在计算器工具界面“查看”菜单栏中选择“科学型”，计算器界面即为科学型计算器工具界面，如图 1.2 所示。

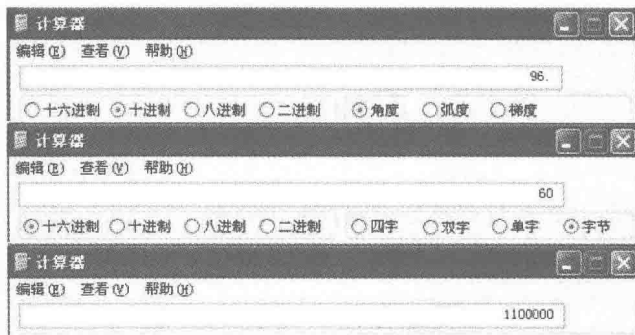


图 1.2 科学型计算器与各进制转换

转换方法：先选择被转换数制的类型，输入转换数字，再选择目标转换数制类型，此时，看到的的就是转换后的数字。如 96 转换为十六进制数、二进制数，先选择数制类型为十进制，如图 1.2 上部所示，在输入框中输入数字 96，然后再选择数制类型为十六进制，此时，显示框中看到的数字即为转换后的十六进制数字 60，如图 1.2 中部所示；再选择数制类型为二进制，此时，显示框中看到的数字即为转换后的二进制数字 1100000，如图 1.2 底部所示。

### 3. 二进制数的运算规则

- (1) 加法运算规则。0+0=0, 0+1=1, 1+1=0 (有进位)
- (2) 减法运算规则。0-0=0, 1-0=1, 1-1=0, 0-1=1 (有借位)
- (3) 乘法运算规则。0×0=0, 1×0=0, 1×1=1

## 1.1.2 微型计算机中数的表示方法

### 1. 机器数与真值

数学中的正负用符号“+”和“-”表示，计算机中是如何表示数的正负呢？在计算机中数据存放在存储单元内，而每个存储单元则由若干二进制位组成，其中每一数位或是 0 或是 1。刚好数的符号或为+号或为-号，这样就可用一个数位来表示数的符号。在计算机中规定用“0”表示“+”，用“1”表示“-”。用来表示数的符号的数位被称为“符号位”（通常为最高数位），于是数的符号在计算机中就数码化了，但从表示形式上看符号位与数值位毫无区别。

设有两个数  $x_1, x_2$ ,  $x_1=+1011011\text{ B}$ ,  $x_2=-1011011\text{ B}$ ，它们在计算机中分别表示为（带下画线部分为符号位，字长为 8 位）：

$$x_1=\underline{0}1011011\text{ B}; x_2=\underline{1}1011011\text{ B}$$

为了区分这两种形式的数，我们把机器中以数码形式表示的数称为机器数（即上例中  $x_1=\underline{0}1011011\text{ B}$  及  $x_2=\underline{1}1011011\text{ B}$ ），而把原来一般书写形式表示的数称为真值（即  $x_1=+1011011\text{ B}$  及  $x_2=-1011011\text{ B}$ ）。

若一个数的所有数位均为数值位，则该数为无符号数；若一个数的最高数位为符号位，而其他数位为数值位，则该数为有符号数。由此可见，对于同一存储单元，它存放的无符号

数和有符号数所能表示的数值范围是不同的（如存储单元为 8 位，当它存放无符号数时，因有效的数值位为 8 位，故该数的范围为 0~255；当它存放有符号数时，因有效的数值位为 7 位，故该数的范围（补码）为-128~+127）。

## 2. 原码

对于一个二进制数，如用最高数位表示该数的符号（“0”表示“+”号，“1”表示“-”号），其余各数位表示其数值本身，则称为原码表示法：

若  $x = \pm x_1 x_2 \cdots x_{n-1}$ ，则  $[x]_{\text{原码}} = x_0 x_1 x_2 \cdots x_{n-1}$

其中， $x_0$  为原机器数的符号位，它满足：

$$x_0 = \begin{cases} 0 & (x \geq 0 \text{ 时}) \\ 1 & (x < 0 \text{ 时}) \end{cases}$$

## 3. 反码

$[x]_{\text{原}} = 0x_1 x_2 \cdots x_{n-1}$ ，则  $[x]_{\text{反}} = [x]_{\text{原}}$

$[x]_{\text{原}} = 1x_1 x_2 \cdots x_{n-1}$ ，则  $[x]_{\text{反}} = 1\bar{x}_1 \bar{x}_2 \cdots \bar{x}_{n-1}$

也就是说，正数的反码与其原码相同（反码=原码），而负数的反码为保持原码的符号位不变，数值位按位取反。

## 4. 补码

（1）补码的引进。首先以日常生活中经常遇到的钟表“对时”为例来说明补码的概念，假定现在是北京标准时间八时整，而一只表却指向十时整。为了校正此表，可以采用倒拨和顺拨两种方法：倒拨就是反时针减少 2 小时（把倒拨视为减法，相当于  $10-2=8$ ），时针指向 8；还可将时针顺拨 10 小时，时针同样也指向 8，把顺拨视为加法，相当于  $10+10=12$ （自动丢失） $+8=8$ ，这自动丢失的数（12）就叫做模（mod）。上述的加法称为“按模 12 的加法”，用数学式可表示为：

$$10+10=12+8=8 \pmod{12}$$

因为时针转一圈会自动丢失一个数 12，故  $10-2$  与  $10+10$  是等价的。称 10 和 -2 对模 12 互补，10 是 -2 对模 12 的补码。引进补码概念后，就可将原来的减法  $10-2=8$  转化为加法  $10+10=12$ （自动丢失） $+8=8 \pmod{12}$ 。

（2）补码的定义。通过上面的例子不难理解计算机中负数的补码表示法。设寄存器（或存储单元）的位数为  $n$  位，则它能表示的无符号数最大值为  $2^n-1$ ，逢  $2^n$  进 1（即  $2^n$  自动丢失）。换句话说，在字长为  $n$  的计算机中，数  $2^n$  和 0 的表示形式一样。若机器中的数以补码表示，则数的补码以  $2^n$  为模，即

$$[x]_{\text{补}} = 2^n + x \pmod{2^n}$$

若  $x$  为正数，则  $[x]_{\text{补}} = x$ ；若  $x$  为负数，则  $[x]_{\text{补}} = 2^n + x = 2^n - |x|$ ，即负数  $x$  的补码等于模  $2^n$  加上其真值或减去其真值的绝对值。

在补码表示法中，零只有唯一的表示形式：0000...0。

（3）求补码的方法。根据上述介绍可知，正数的补码等于原码。下面介绍负数求补码的