

51单片机 应用开发25例 ——基于Proteus仿真

张新 陈跃琴 编著

背景介绍→设计思路和涉及的基础原理→硬件设计→软件
设计→仿真与总结

免费资源下载:

提供每个实例的Proteus电路及C51应用代码,可登录
www.hxedu.com.cn查找本书下载,直接运行仿真。



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

51 单片机应用开发 25 例

——基于 Proteus 仿真

张 新 陈跃琴 编著

電子工業出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

目前, Keil μ Vision 是应用最广泛的 51 单片机软件开发环境, Proteus 是应用最广泛的硬件仿真环境, 而本书基于 Keil μ Vision 和 Proteus 介绍了 25 个 51 单片机的应用实例, 每个实例都包括背景介绍、设计思路、硬件设计、软件设计以及仿真与总结, 并提供了相应的 Proteus 电路及 C51 应用实例代码。

本书共分 25 章, 包含丰富的单片机内部资源和外围模块的应用实例, 并且都基于 Proteus 仿真, 简单直观, 适合具有初步单片机基础的单片机工程师进阶学习, 也适合高等院校电子类专业学生和单片机爱好者阅读, 还可以作为工程设计的参考手册。

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

51 单片机应用开发 25 例: 基于 Proteus 仿真 / 张新, 陈跃琴编著. —北京: 电子工业出版社, 2013.10
ISBN 978-7-121-21628-2

I. ①5… II. ①张… ②陈… III. ①单片微型计算机—系统仿真—应用软件 IV. ①TP312

中国版本图书馆 CIP 数据核字 (2013) 第 238241 号

策划编辑: 陈韦凯

责任编辑: 桑 昀

印 刷: 北京京师印务有限公司

装 订: 北京京师印务有限公司

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: 787×1 092 1/16 印张: 27.25 字数: 736 千字

印 次: 2013 年 10 月第 1 次印刷

印 数: 3 500 册 定价: 59.00 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, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010) 88258888。

前言

一、行业背景

51 单片机具有体积小、功能强和价格低的特点,在工业控制、数据采集、智能仪表、机电一体化、家用电器等领域有着广泛的应用,其应用可以大大提高生产和生活的自动化水平。近年来,随着嵌入式的应用越来越广泛,51 单片机的开发也变得更加灵活和高效,而 51 单片机的开发和应用也已经成为嵌入式应用领域的一个重大课题。

二、关于本书

目前,Keil μ Vision 是应用最广泛的 51 单片机软件开发环境,Proteus 是应用最广泛的硬件仿真环境,而本书基于 Keil μ Vision 和 Proteus 介绍了 25 个从简单到复杂,从内部资源应用、扩展系统应用到嵌入式操作系统应用的实例。读者从本书中既可以了解该应用系统设计的基础知识、电路模块以及对应的代码,也可以在 Proteus 中进行仿真并且观察仿真结果。

本书各章的实例说明如下:

第 1 章“呼吸灯”是一个实现发光二极管呼吸效果的应用系统。

第 2 章“跑步机启/停和速度控制模块”是一个对跑步机的工作状态进行控制的应用系统。

第 3 章“简易电子琴”是一个可以弹奏的简易电子琴应用系统。

第 4 章“手机拨号模块”是一个手机的拨号界面应用系统,包括键盘和液晶显示模块。

第 5 章“简易频率计”是一个对当前输入频率进行测量的应用系统。

第 6 章“PC 中控系统”是一个实现 PC 对外部系统进行控制的应用系统。

第 7 章“天车控制系统”是天车动作的核心控制模块。

第 8 章“负载平衡监控系统”是一个对当前系统平衡性进行监控的模块。

第 9 章“电子抽奖系统”是一个用 51 单片机实现抽奖的系统。

第 10 章“多点温度采集系统”是使用多个温度传感器对多点温度进行轮询采集的应用系统。

第 11 章“简易波形发生器”是在用户控制下产生简单波形的模型。

第 12 章“数字时钟”是一个可以用数字显示当前时间和日期的应用系统。

第 13 章“模拟时钟”是在液晶模块上模拟钟表指针来显示时间信息的应用系统。

第 14 章“自动打铃器”是根据当前时钟来自动打铃提示上课和下课,并且显示当前时间的应用系统。

第 15 章“手动程控放大器”是根据当前用户选择来对输入信号进行放大的应用系统。

第 16 章“自动换挡数字电压表”是一个根据当前输入电压值来自动切换量程,并且测量当前电压值的应用系统。

第 17 章“货车超重监测系统”是通过压力来检测当前道路上行驶的货车是否超重,并且对相应的数据进行记录的应用系统。

第 18 章“远程仓库湿度监测系统”是一个获得远程的仓库湿度数据的应用系统。

第 19 章“带计时功能的简单计算器”是一个简单的可以显示时间的计算器模型。

第 20 章“密码保险箱”是一个密码保险箱的应用系统,用户可以自行设置密码,并且通过设置好的密码打开保险箱。

第 21 章“SD 卡读卡器”是一个简易的可以读写 SD 卡的读卡器模型。

第 22 章“简易数字示波器”是一个可以对简单波形进行测量，并且将该波形显示到液晶模块上的应用系统。

第 23 章“多功能电子闹钟”是一个有温度显示、时间显示和定时闹铃等功能的电子闹钟模型。

第 24 章“俄罗斯方块”是一个俄罗斯方块的游戏模型。

第 25 章“RTX51 操作系统应用”是一个 RTX51 操作系统在 51 单片机上的应用实例，包括对 RTX51 操作系统的介绍和应用方法，并且给出了一个应用实例。

三、本书特色

(1) 应用实例从简单到复杂，涵盖了 51 单片机从内部资源到用户输入通道、A/D 信号采集、温度/湿度传感芯片、有线通信模块、操作系统等常用资源和常用模型的应用。

(2) 基于 Proteus 硬件开发环境提供了相应的仿真运行实例及其输出结果。

(3) 对于每个应用实例，都按照实例背景介绍、实例设计思路和涉及的基础原理介绍、硬件设计、软件设计及仿真综合与总结来进行组织，条理清晰，便于阅读理解。

(4) 提供了大量的 Proteus 应用电路和 Keil μ Vision 的工程文件，读者可以直接运行仿真。

四、作者介绍

本书由张新、陈跃琴编著。同时，参与本书编写和审定工作的还有孙明、唐伟、王杨、顾辉、李成、陈杰、张霁芬、张计、陈军、张强、杨明、李建、张玉兰等人。

为与 Proteus 软件中的电路图保持一致，本书仿真电路中的部分元件符号（如二极管、电阻、电容等）以及单位（如 10k 未改为 10k Ω ，10 μ F 未改为 10 μ F 等）的不规范处未做标准化处理，在此特加以说明。

由于时间仓促、程序和图表较多，受学识水平所限，错误之处在所难免，请广大读者给予批评指正。

编 著 者

目 录

第1章 呼吸灯	(1)
1.1 呼吸灯应用系统的背景介绍	(1)
1.2 呼吸灯应用系统的设计思路	(1)
1.2.1 呼吸灯应用系统的工作流程	(1)
1.2.2 呼吸灯应用系统的需求分析与设计	(2)
1.2.3 “呼吸”效果的实现原理	(2)
1.2.4 51 单片机简介	(2)
1.2.5 RCL 响应电路	(3)
1.2.6 PWM 控制	(3)
1.2.7 51 单片机的软件开发环境使用	(4)
1.3 呼吸灯应用系统的硬件设计	(11)
1.3.1 呼吸灯硬件系统的模块划分	(12)
1.3.2 呼吸灯硬件系统的电路	(12)
1.3.3 硬件模块基础——发光二极管 (LED)	(13)
1.3.4 硬件模块基础——三极管	(14)
1.3.5 硬件模块基础——电阻、电容和电感	(14)
1.3.6 Proteus 硬件仿真环境的使用	(15)
1.4 呼吸灯应用系统软件设计	(18)
1.4.1 呼吸灯应用系统的软件流程	(19)
1.4.2 呼吸灯应用系统软件的应用代码	(19)
1.5 呼吸灯应用系统的仿真与总结	(21)
第2章 跑步机启/停和速度控制模块	(30)
2.1 跑步机启/停和速度控制模块的背景介绍	(30)
2.2 跑步机启/停和速度控制模块的设计思路	(30)
2.2.1 跑步机启/停和速度控制系统的工作流程	(30)
2.2.2 跑步机启/停和速度控制系统的需求分析与设计	(31)
2.2.3 长按键和短按键检测原理	(31)
2.3 跑步机启/停和速度控制模块的硬件设计	(31)
2.3.1 跑步机启/停和速度控制硬件系统的模块划分	(31)
2.3.2 跑步机启/停和速度控制模块的电路	(32)
2.3.3 硬件模块基础——独立按键	(33)
2.3.4 硬件模块基础——数码管	(34)
2.4 跑步机启/停和速度控制模块的软件设计	(35)
2.4.1 跑步机启/停和速度控制模块的软件模块划分和流程设计	(35)
2.4.2 启/停控制模块设计	(36)
2.4.3 速度控制模块设计	(37)

2.4.4 跑步机启/停和速度控制模块的软件综合	(40)
2.5 跑步机启/停和速度控制模式的应用系统仿真与总结	(41)
第3章 简易电子琴	(43)
3.1 简易电子琴应用系统的背景介绍	(43)
3.2 简易电子琴应用系统的设计思路	(43)
3.2.1 简易电子琴应用系统的工作流程	(43)
3.2.2 简易电子琴应用系统的需求分析与设计	(44)
3.2.3 51 单片机播放音乐	(44)
3.3 简易电子琴应用系统的硬件设计	(45)
3.3.1 简易电子琴的硬件系统模块划分	(45)
3.3.2 简易电子琴的硬件系统电路	(46)
3.3.3 硬件模块基础——独立按键	(47)
3.3.4 硬件模块基础——蜂鸣器	(48)
3.4 简易电子琴应用系统的软件设计	(48)
3.4.1 简易电子琴应用系统的软件流程	(48)
3.4.2 简易电子琴的软件应用代码	(48)
3.5 简易电子琴应用系统的仿真与总结	(52)
第4章 手机拨号模块	(54)
4.1 手机拨号模块的背景介绍	(54)
4.2 手机拨号模块的设计思路	(54)
4.2.1 手机拨号模块的工作流程	(54)
4.2.2 手机拨号模块的需求分析与设计	(54)
4.2.3 手机拨号模块的工作原理	(55)
4.3 手机拨号模块的硬件设计	(55)
4.3.1 手机拨号模块的硬件划分	(55)
4.3.2 手机拨号模块的电路图	(55)
4.3.3 硬件模块基础——行列扫描键盘	(56)
4.3.4 硬件模块基础——1602 液晶模块	(57)
4.4 手机拨号模块的软件设计	(59)
4.4.1 软件模块的划分和流程	(59)
4.4.2 行列扫描键盘的软件驱动模块设计	(60)
4.4.3 1602 液晶的软件驱动模块设计	(61)
4.4.4 手机拨号模块的软件综合	(63)
4.5 手机拨号模块的应用系统仿真与总结	(64)
第5章 简易频率计	(66)
5.1 简易频率计的背景介绍	(66)
5.2 简易频率计的设计思路	(66)
5.2.1 简易频率计应用系统的工作流程	(66)
5.2.2 简易频率计应用系统的需求分析与设计	(67)
5.2.3 频率测量原理	(67)

5.3	简易频率计的硬件设计	(67)
5.3.1	简易频率计的硬件模块划分	(67)
5.3.2	简易频率计的电路图	(68)
5.3.3	硬件模块基础——多位数码管	(68)
5.4	简易频率计的软件设计	(69)
5.4.1	简易频率计的软件模块的划分和流程	(69)
5.4.2	频率测量和计算模块的设计	(70)
5.4.3	显示驱动模块设计	(71)
5.4.4	简易频率计的软件综合	(72)
5.5	简易频率计的应用系统仿真与总结	(73)
第6章	PC 中控系统	(76)
6.1	PC 中控系统的背景介绍	(76)
6.2	PC 中控系统的设计思路	(76)
6.2.1	PC 中控系统的工作流程	(76)
6.2.2	PC 中控系统的需求分析与设计	(76)
6.2.3	PC 和 51 单片机应用系统的通信方式	(77)
6.3	PC 中控系统的硬件设计	(79)
6.3.1	硬件系统模块划分	(79)
6.3.2	硬件系统的电路图	(79)
6.3.3	硬件模块基础——51 单片机的串口模块	(80)
6.3.4	硬件模块基础——MAX232	(84)
6.3.5	硬件模块基础——光电隔离器	(85)
6.3.6	硬件模块基础——继电器	(85)
6.4	PC 中控系统的软件设计	(86)
6.4.1	软件模块划分和流程设计	(86)
6.4.2	软件综合	(86)
6.5	PC 中控系统的仿真与总结	(88)
第7章	天车控制系统	(92)
7.1	天车控制系统的背景介绍	(92)
7.2	天车控制系统的设计思路	(93)
7.2.1	天车控制系统的工作流程	(93)
7.2.2	天车控制系统的需求分析与设计	(93)
7.2.3	天车控制系统的工作原理	(93)
7.3	天车控制系统的硬件设计	(94)
7.3.1	天车控制系统的硬件模块划分	(94)
7.3.2	硬件系统的电路	(94)
7.3.3	硬件模块基础——直流电动机	(95)
7.3.4	硬件模块基础——H 桥	(95)
7.3.5	硬件模块基础——步进电动机	(96)
7.3.6	硬件模块基础——ULN2003A	(97)

7.4	天车控制系统的软件设计	(97)
7.4.1	天车控制系统的软件模块划分和流程设计	(98)
7.4.2	直流电动机驱动模块设计	(98)
7.4.3	步进电动机驱动模块设计	(99)
7.4.4	天车控制系统的软件综合	(99)
7.5	天车控制应用系统的仿真与总结	(100)
第8章	负载平衡监控系统	(102)
8.1	负载平衡监控系统的背景介绍	(102)
8.2	负载平衡监控系统的设计思路	(102)
8.2.1	负载平衡监控系统的工作流程	(102)
8.2.2	负载平衡监控系统的需求分析与设计	(102)
8.2.3	51 单片机应用系统的通信模型和 RS-422 协议	(103)
8.3	负载平衡监控系统的硬件设计	(103)
8.3.1	负载平衡监控系统的硬件划分	(103)
8.3.2	负载平衡监控系统的硬件电路	(104)
8.3.3	硬件模块基础——SN75179	(105)
8.3.4	硬件模块基础——拨码开关	(105)
8.4	负载平衡监控系统的软件设计	(106)
8.4.1	负载平衡监控系统的软件模块划分和流程设计	(106)
8.4.2	负载平衡监控系统的软件综合	(106)
8.5	负载平衡监控应用系统的仿真与总结	(109)
第9章	电子抽奖系统	(111)
9.1	电子抽奖系统的背景介绍	(111)
9.2	电子抽奖系统的设计思路	(111)
9.2.1	电子抽奖系统的工作流程	(111)
9.2.2	电子抽奖系统的需求分析与设计	(112)
9.2.3	单片机系统的随机数产生原理	(112)
9.3	电子抽奖系统的硬件设计	(113)
9.3.1	电子抽奖系统的硬件划分	(113)
9.3.2	抽奖系统的硬件电路	(114)
9.3.3	硬件模块基础——51 单片机的外部中断	(115)
9.3.4	硬件模块基础——51 单片机的定时器/计数器	(116)
9.3.5	硬件模块基础——74HC595	(118)
9.4	电子抽奖系统的软件设计	(119)
9.4.1	电子抽奖系统的软件模块划分和流程设计	(119)
9.4.2	74HC595 的驱动函数模块设计	(120)
9.4.3	电子抽奖系统的软件综合	(123)
9.5	电子抽奖应用系统的仿真与总结	(126)
第10章	多点温度采集系统	(128)
10.1	多点温度采集系统的背景介绍	(128)

10.2	多点温度采集系统的设计思路	(128)
10.2.1	多点温度采集系统的工作流程	(128)
10.2.2	多点温度采集系统的需求分析与设计	(129)
10.2.3	单片机应用系统的温度采集方法	(129)
10.2.4	1-wire 总线的工作原理	(130)
10.3	多点温度采集系统的硬件设计	(132)
10.3.1	多点温度采集系统的硬件模块划分	(132)
10.3.2	多点温度采集系统的电路	(132)
10.3.3	硬件模块基础——DS18B20	(133)
10.4	多点温度采集系统的软件设计	(135)
10.4.1	多点温度采集系统的软件模块划分和流程设计	(136)
10.4.2	DS18B20 驱动函数模块设计	(136)
10.4.3	1602 液晶驱动函数模块设计	(139)
10.4.4	多点温度采集系统的软件综合	(140)
10.5	多点温度采集应用系统的仿真与总结	(142)
第 11 章	简易波形发生器	(145)
11.1	简易波形发生器的背景介绍	(145)
11.2	简易波形发生器的设计思路	(145)
11.2.1	简易波形发生器的工作流程	(145)
11.2.2	简易波形发生器的需求分析与设计	(146)
11.2.3	D/A 芯片的工作原理	(146)
11.2.4	I ² C 接口总线工作原理	(147)
11.3	简易波形发生器的硬件设计	(150)
11.3.1	简易波形发生器的硬件模块划分	(150)
11.3.2	简易波形发生器硬件电路图	(150)
11.3.3	硬件模块基础——单刀单掷开关	(151)
11.3.4	硬件模块基础——MAX517	(151)
11.4	简易波形发生器的软件设计	(152)
11.4.1	简易波形发生器的软件模块划分和流程设计	(152)
11.4.2	MAX517 的驱动函数设计	(153)
11.4.3	简易波形发生器的软件综合	(155)
11.5	简易波形发生器的应用系统仿真与总结	(158)
第 12 章	数字时钟	(159)
12.1	数字时钟的背景介绍	(159)
12.2	数字时钟的设计思路	(159)
12.2.1	数字时钟的工作流程	(159)
12.2.2	数字时钟的需求分析与设计	(159)
12.2.3	单片机应用系统的时间获取方法	(159)
12.3	数字时钟的硬件设计	(160)
12.3.1	数字时钟的硬件模块划分	(160)

12.3.2	数字时钟的硬件的电路	(160)
12.3.3	硬件模块基础——DS12C887	(161)
12.4	数字时钟的软件设计	(165)
12.4.1	数字时钟的软件模块划分和流程设计	(165)
12.4.2	DS12C887 的驱动函数模块设计	(165)
12.4.3	1602 液晶显示驱动函数模块设计	(166)
12.4.4	数字时钟应用系统的软件综合	(168)
12.5	数字时钟应用系统的仿真与总结	(169)
第 13 章	模拟时钟	(171)
13.1	模拟时钟的背景介绍	(171)
13.2	模拟时钟的设计思路	(171)
13.2.1	模拟时钟的工作流程	(171)
13.2.2	模拟时钟的需求分析与设计	(171)
13.2.3	模拟时钟的时间获取方法	(172)
13.3	模拟时钟的硬件设计	(172)
13.3.1	模拟时钟的硬件模块划分	(172)
13.3.2	模拟时钟硬件系统的电路	(172)
13.3.3	51 单片机的地址-数据总线扩展方法	(173)
13.3.4	硬件模块基础——外部 RAM 芯片 62256	(175)
13.3.5	硬件模块基础——12864 液晶模块	(176)
13.4	模拟时钟的软件设计	(177)
13.4.1	模拟时钟的软件模块划分和流程设计	(177)
13.4.2	时间信息算法模块的设计	(178)
13.4.3	12864 液晶模块的驱动函数设计	(179)
13.4.4	模拟时钟系统的软件综合	(190)
13.5	模拟时钟应用系统的仿真与总结	(191)
第 14 章	自动打铃器	(194)
14.1	自动打铃器的背景介绍	(194)
14.2	自动打铃器的设计思路	(194)
14.2.1	自动打铃器的工作流程	(194)
14.2.2	自动打铃器的需求分析与设计	(194)
14.2.3	单片机串行端口字符串输出	(195)
14.3	自动打铃器的硬件设计	(197)
14.3.1	自动打铃器的硬件模块划分	(197)
14.3.2	自动打铃器的硬件电路	(197)
14.3.3	自动打铃器的硬件模块基础——时钟芯片 PCF8563	(198)
14.4	自动打铃器的软件设计	(202)
14.4.1	自动打铃器软件的工作流程设计	(202)
14.4.2	PCF8563 基础驱动函数模块设计	(202)
14.4.3	1602 液晶驱动函数模块设计	(206)

14.4.4	自动打铃器系统的软件综合	(208)
14.5	自动打铃器应用系统仿真与总结	(212)
第 15 章	手动程控放大器	(214)
15.1	手动程控放大器的背景介绍	(214)
15.2	手动程控放大器的设计思路	(214)
15.2.1	手动程控放大器的工作流程	(214)
15.2.2	手动程控放大器的需求分析	(214)
15.2.3	单片机应用系统的信号放大	(215)
15.2.4	手动程控放大器的实现方法	(217)
15.3	手动程控放大器的硬件设计	(218)
15.3.1	手动程控放大器的硬件系统模块	(218)
15.3.2	手动程控放大器的硬件系统电路	(218)
15.3.3	硬件模块基础—— $\mu A741$	(220)
15.3.4	硬件模块基础——CD4066	(220)
15.3.5	硬件模块基础——MAX7219	(221)
15.4	手动程控放大器的软件设计	(225)
15.4.1	软件模块划分和工作流程	(225)
15.4.2	MAX7219 驱动模块设计	(225)
15.4.3	手动程控放大器的软件综合	(226)
15.5	手动程控放大器应用系统仿真与总结	(229)
第 16 章	自动换挡数字电压表	(230)
16.1	自动换挡数字电压表的背景介绍	(230)
16.2	自动换挡数字电压表的设计思路	(230)
16.2.1	自动换挡数字电压表的工作流程	(230)
16.2.2	自动换挡数字电压表的需求分析	(231)
16.2.3	自动换挡数字电压表的换挡原理	(231)
16.3	自动换挡数字电压表的硬件设计	(231)
16.3.1	自动换挡数字电压表的硬件模块	(231)
16.3.2	自动换挡数字电压表的电路	(231)
16.3.3	硬件模块基础——LM324	(233)
16.3.4	硬件模块基础——ADC0809	(233)
16.4	自动换挡数字电压表的软件设计	(234)
16.4.1	自动换挡数字电压表的软件模块划分和工作流程	(234)
16.4.2	1602 液晶驱动模块函数设计	(234)
16.4.3	自动换挡数字电压表的软件综合	(236)
16.5	自动换挡数字电压表应用系统仿真与总结	(239)
第 17 章	货车超重监测系统	(241)
17.1	货车超重监测系统的背景介绍	(241)
17.2	货车超重监测系统的设计思路	(241)
17.2.1	货车超重监测系统的工作流程	(241)

17.2.2	货车超重监测系统的需求分析	(242)
17.2.3	货车超重监测系统的工作原理	(242)
17.3	货车超重监测系统的硬件设计	(242)
17.3.1	货车超重监测系统的硬件模块	(242)
17.3.2	货车超重监测系统的电路	(243)
17.3.3	硬件模块基础——压力传感器 MPX4115	(244)
17.3.4	硬件模块基础——A/D 芯片 ADC0832	(245)
17.3.5	硬件模块基础——E ² PROM 芯片 24C04A	(246)
17.4	货车超重监测系统的软件设计	(247)
17.4.1	货车超重监测系统的软件模块划分和 workflow	(247)
17.4.2	A/D 转换模块函数设计	(247)
17.4.3	E ² PROM 读写模块函数设计	(249)
17.4.4	货车超重检测系统的软件综合	(253)
17.5	货车超重监测应用系统仿真与总结	(255)
第 18 章	远程仓库湿度监测系统	(257)
18.1	远程仓库湿度监测系统的背景介绍	(257)
18.2	远程仓库湿度监测系统的设计思路	(257)
18.2.1	远程仓库湿度监测系统的工作流程	(257)
18.2.2	远程仓库湿度监测系统的需求分析	(257)
18.2.3	远程仓库湿度监测系统的工作原理	(257)
18.3	远程仓库湿度监测系统的硬件设计	(258)
18.3.1	远程仓库湿度监测系统的硬件模块	(258)
18.3.2	远程仓库湿度监测系统的电路	(258)
18.3.3	硬件模块基础——湿度传感器 SHT11	(259)
18.3.4	硬件模块基础——RS-485 芯片 MAX487	(261)
18.4	远程仓库湿度监测系统的软件设计	(262)
18.4.1	远程仓库湿度监测系统的软件模块划分和 workflow	(262)
18.4.2	湿度采集模块函数设计	(262)
18.4.3	1602 液晶驱动模块函数设计	(264)
18.4.4	远程仓库湿度监测系统的软件综合	(265)
18.5	远程仓库湿度监测应用系统仿真与总结	(268)
第 19 章	带计时功能的简单计算器	(269)
19.1	带计时功能的简单计算器的背景介绍	(269)
19.2	带计时功能的简单计算器的设计思路	(270)
19.2.1	带计时功能的简单计算器的工作流程	(270)
19.2.2	带计时功能的简单计算器的需求分析	(270)
19.2.3	带计时功能的简单计算器的工作原理	(270)
19.3	带计时功能的简单计算器的硬件设计	(271)
19.3.1	带计时功能的简单计算器的硬件模块	(271)
19.3.2	硬件系统的电路图	(271)

19.4	带计时功能的简单计算器的软件设计	(272)
19.4.1	带计时功能的简单计算器的软件模块划分和 workflow	(272)
19.4.2	键盘扫描和处理模块函数设计	(273)
19.4.3	计算器功能处理模块函数设计	(274)
19.4.4	计时器功能处理模块函数设计	(280)
19.4.5	显示模块函数设计	(284)
19.4.6	带计时功能的简单计算器的软件综合	(287)
19.5	带计时功能的简单计算器的应用系统仿真与总结	(291)
第 20 章	密码保险箱	(292)
20.1	密码保险箱的背景介绍	(292)
20.2	密码保险箱的设计思路	(292)
20.2.1	密码保险箱的工作流程	(292)
20.2.2	密码保险箱的需求分析与设计	(293)
20.2.3	密码保险箱的工作原理	(293)
20.3	密码保险箱的硬件设计	(293)
20.3.1	密码保险箱的硬件模块	(293)
20.3.2	密码保险箱的电路	(294)
20.4	密码保险箱的软件设计	(295)
20.4.1	密码保险箱的软件模块划分和 workflow	(295)
20.4.2	键盘扫描模块函数设计	(296)
20.4.3	显示驱动模块函数设计	(302)
20.4.4	状态驱动模块函数设计	(303)
20.4.5	报警声驱动模块函数设计	(303)
20.4.6	电动机驱动模块函数设计	(303)
20.4.7	密码保险箱的软件综合	(303)
20.5	密码保险箱应用系统仿真与总结	(306)
第 21 章	SD 卡读卡器	(308)
21.1	SD 卡读卡器的背景介绍	(308)
21.2	SD 卡读卡器的设计思路	(308)
21.2.1	SD 卡读卡器的工作流程	(308)
21.2.2	SD 卡读卡器的需求分析	(309)
21.2.3	SPI 接口总线	(309)
21.2.4	SD 卡读写基础	(311)
21.3	SD 卡读卡器的硬件设计	(315)
21.3.1	SD 卡读卡器的硬件模块	(315)
21.3.2	SD 卡读卡器的电路	(316)
21.3.3	硬件模块基础——SD 卡	(317)
21.4	SD 卡读卡器的软件设计	(317)
21.4.1	SD 卡读卡器软件的工作流程	(317)
21.4.2	SD 卡基础驱动模块设计	(318)

21.4.3	SD 卡读写函数模块设计	(320)
21.4.4	SD 卡读卡器的软件综合	(322)
21.5	SD 卡读卡器应用系统仿真与总结	(323)
第 22 章	简易数字示波器	(325)
22.1	简易数字示波器的背景介绍	(325)
22.2	简易数字示波器的设计思路	(325)
22.2.1	简易数字示波器的工作流程	(325)
22.2.2	简易数字示波器的需求分析	(326)
22.2.3	简易数字示波器的工作原理	(326)
22.3	简易数字示波器的硬件设计	(326)
22.3.1	硬件模块	(326)
22.3.2	简易数字示波器的电路	(326)
22.3.3	硬件模块基础——信号的加法运算	(328)
22.4	简易数字示波器的软件设计	(330)
22.4.1	简易数字示波器的软件模块划分和工作流程	(330)
22.4.2	A/D 转换模块函数设计	(330)
22.4.3	AMPIRE 128×64 液晶模块函数设计	(331)
22.4.4	简易数字示波器的软件综合	(334)
22.5	简易数字示波器应用系统仿真与总结	(340)
第 23 章	多功能电子闹钟	(342)
23.1	多功能电子闹钟应用系统的背景介绍	(342)
23.2	多功能电子闹钟应用系统的设计思路	(342)
23.2.1	多功能电子闹钟的工作流程	(342)
23.2.2	多功能电子闹钟的需求分析	(342)
23.2.3	多功能电子闹钟的工作原理	(343)
23.3	多功能电子闹钟应用系统的硬件设计	(343)
23.3.1	多功能电子闹钟的硬件模块	(343)
23.3.2	多功能电子闹钟的电路	(344)
23.3.3	硬件模块基础——DS1302	(345)
23.4	多功能电子闹钟应用系统的软件设计	(346)
23.4.1	多功能电子闹钟的软件模块划分和工作流程	(346)
23.4.2	温度采集模块函数设计	(347)
23.4.3	时钟芯片驱动模块函数设计	(350)
23.4.4	显示模块驱动函数设计	(352)
23.4.5	时间设置模块驱动函数设计	(353)
23.4.6	闹钟设置模块驱动函数设计	(357)
23.4.7	声音报警模块驱动函数设计	(360)
23.4.8	多功能电子闹钟的软件综合	(360)
23.5	多功能电子闹钟应用系统仿真与总结	(363)

第 24 章 俄罗斯方块	(365)
24.1 俄罗斯方块应用系统的背景介绍	(365)
24.2 俄罗斯方块应用系统的设计思路	(366)
24.2.1 俄罗斯方块的工作流程	(366)
24.2.2 俄罗斯方块的需求分析	(366)
24.2.3 俄罗斯方块的工作原理	(366)
24.3 俄罗斯方块应用系统的硬件设计	(366)
24.3.1 俄罗斯方块的硬件模块	(366)
24.3.2 俄罗斯方块的电路	(367)
24.4 俄罗斯方块应用系统的软件设计	(368)
24.4.1 俄罗斯方块的软件模块划分和工作流程	(368)
24.4.2 液晶驱动模块函数设计	(369)
24.4.3 游戏操控模块函数设计	(377)
24.4.4 游戏逻辑控制模块函数设计	(378)
24.4.5 俄罗斯方块的软件综合	(384)
24.5 俄罗斯方块应用系统仿真与总结	(390)
第 25 章 RTX51 操作系统应用	(391)
25.1 RTX51 操作系统的基础	(391)
25.1.1 RTX51 占用的资源	(392)
25.1.2 RTX51 的实现机制	(393)
25.1.3 RTX51 的工作原理	(394)
25.1.4 RTX51 的配置	(396)
25.1.5 RXT51 的库函数	(398)
25.1.6 在 RTX51 操作系统下编写用户代码的流程	(404)
25.2 基于 RTX51 操作系统的应用实例——交通灯	(406)
25.2.1 应用实例的 Proteus 电路	(407)
25.2.2 交通灯应用实例的代码	(408)
25.2.3 交通灯应用实例的仿真运行结果和总结	(417)

第1章 呼吸灯

呼吸灯是一种视觉效果，其灯光在 51 单片机的控制下完成由亮到暗的逐渐变化，感觉像是在呼吸，这种效果广泛被应用于数码产品、计算机、音响和汽车等各个领域，能够起到很好的视觉装饰效果。

本章应用实例涉及的知识如下：

- 51 单片机基础；
- RCL 滤波电路原理；
- PWM 控制原理；
- 发光二极管的应用原理；
- 三极管的应用原理；
- 51 单片机应用系统中常用的电阻、电容和电感的应用原理；
- 51 单片机开发环境 μ Vision 的介绍；
- Proteus 硬件仿真环境应用。

1.1 呼吸灯应用系统的背景介绍

呼吸灯最先被应用的应该是在苹果公司的计算机上，其指示标志灯会缓慢由暗变亮，又逐步由亮变暗，过程类似人的呼吸，它由以下两个阶段组成。

- (1) 吸气：灯的亮度曲线上升。
- (2) 呼气：灯的亮度曲线下降。

由于其模拟的是人的呼吸效果，对成人而言，平均每分钟呼吸 16~18 次，对儿童而言，平均每分钟呼吸 20 次，而且在不同的状态下，呼吸的频率是不同的，吸气和呼气的长度也是不同的，所以同样可以使用呼吸灯来指示一个系统的工作状态，如单片机系统在满负荷工作时其“呼吸”频率可以变快，反之变慢。

1.2 呼吸灯应用系统的设计思路

1.2.1 呼吸灯应用系统的工作流程

呼吸灯应用系统的工作流程如图 1.1 所示，其过程相当简单，就是一个“明—暗—明”的过程。