



电子信息与电气学科规划教材

单片机系统

设计与开发教程

张文祥 李志军 张子红 张小清 编著



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

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

单片机系统设计与开发教程

张文祥 李志军 张子红 张小清 编著

電子工業出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

本书以单片机的单元电路设计和应用实例为主线,对涉及的相关理论进行介绍,共分为基础篇、设计篇和应用篇三部分。基础篇包括单片机应用系统的设计与开发、KeilV808A 和 Proteus 软件的使用、单片机内部资源及其 C 语言编程。设计篇主要包括 LED 数码管、矩阵键盘、定时器中断、液晶显示、串口通信、红外接收、A/D 和 D/A 转换等单元电路设计。应用篇包括数字温度计设计、无线数据传输系统设计、超声波测距仪设计、铁路限速标志设计。

本书内容丰富,实用性强,可作为高等院校信息与通信工程及相关专业的本科生教材,还可作为相关领域的工程技术人员的参考书。

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

单片机系统设计与开发教程 / 张文祥等编著. —北京:电子工业出版社, 2011.5
ISBN 978-7-121-13361-9

I. ①单… II. ①张… III. ①单片微型计算机—C 语言—程序设计—教材 IV. ①TP368.1 ②TP312

中国版本图书馆 CIP 数据核字(2011)第 073102 号

责任编辑:董亚峰 特约编辑:史 涛

印 刷:北京丰源印刷厂

装 订:三河市鹏成印业有限公司

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:16 字数:498 千字

印 次:2011 年 5 月第 1 次印刷

印 数:4 000 册 定价:29.80 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 zlts@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

目前,单片机技术在测控系统、智能仪表、机电一体化、机器人、家用电器等领域均得到了广泛的应用,极大地推动了电子产业的发展。2006年,单片机设计师也正式成为我国的一种新职业。鉴于单片机在工业领域和日常生活中的应用日趋广泛和深入,以及社会对单片机应用人才的大量需求,单片机技术已成为电子、通信、计算机、信息、电气、自动化、机电和数控等各工科专业学生必须掌握的一门基本技能。目前高等院校各工科专业普遍将单片机系统设计与开发设置为必修课。

本书根据作者多年应用实践和授课经验,从一个单片机初学者的角度出发,介绍了单片机应用系统的设计方法和开发过程,使初学者在脑海中对学习和使用单片机有一个整体的认识。针对初学者尤其是在校广大学生资金有限,不能一次性投入太多的情况,引入了 Proteus 软件,它能够很好地帮助初学者在不能购买单片机学习开发板的情况下,仍然可以进行单片机应用系统的设计与开发,并通过使用 Proteus 仿真软件来完成应用实例的验证。调试通过后,还可以依据原理图,搭建实际的硬件实验电路,对软硬件进行联机调试,在调试过程中进一步加深对单片机应用系统软硬件设计与开发的理解。

本书共分为 8 章,其中第 1~3 章为基础篇,主要介绍了单片机应用系统的设计与开发、KeilV808A 和 Proteus 软件的使用、单片机内部资源及其 C 语言编程。第 4 章为设计篇,主要介绍了 LED 数码管、矩阵键盘、定时器中断、液晶显示、串口通信、红外接收、A/D 和 D/A 转换等单元电路的设计。第 5~8 章为应用篇,主要介绍了数字温度计的设计、无线数据传输系统的设计、超声波测距仪的设计和铁路限速标志的设计。读者可以紧跟作者的思路,在设计中学会思考,在制作中学会设计,活学活用,直到将所学的单片机知识能够熟练运用并能够解决生产生活中遇到的实际问题。使得读者在使用本教程后,在短时间内成为单片机领域基础理论丰富、设计与开发能力超强的应用型人才。

本书第 1 章、第 3 章由张子红编写,第 2 章由张小清编写;第 4 章由李志军编写;第 5~8 章由张文祥编写。

本书在编写过程中,得到了谢子殿教授和郭继坤教授的大力帮助,他们提供了一些宝贵资料及建议,并指导了部分章节的编写工作,在此表示感谢。对本书所列文献作者,在此一并表示感谢。

由于水平时间有限,错误不当之处在所难免,敬请读者批评指正。

编 者

2011 年 4 月

目 录

基 础 篇

第 1 章 单片机应用系统的设计与开发	3
1.1 单片机应用系统的设计方法	3
1.1.1 系统总体方案的确定	4
1.1.2 应用系统的硬件设计	5
1.1.3 应用系统的软件设计	8
1.1.4 应用系统的抗干扰设计	9
1.2 单片机应用系统的开发过程	10
1.2.1 单片机的开发与开发工具	10
1.2.2 单片机开发系统的功能	11
1.2.3 单片机应用系统的调试、运行与维护	13
第 2 章 KeilV808A 和 Proteus 软件的使用	18
2.1 uVision3 集成开发环境	18
2.2 KeilV808A 的使用	21
2.2.1 创建第一个 KeilV808A 的应用程序	21
2.2.2 程序文件的编译与链接	25
2.3 调试仿真功能的使用	27
2.4 Protues ISIS 设计与仿真平台	28
2.4.1 界面简介	29
2.4.2 Proteus 文件操作	34
2.5 Proteus 库	35
2.5.1 Proteus 库分类	35
2.5.2 部分模型举例	38
2.5.3 库规则	40
2.6 VSM 源程序编辑器和代码生成工具	45
2.7 单片机系统的 Proteus 设计与仿真基础	46
2.7.1 Proteus 设计与仿真流程	46
2.7.2 Proteus 电路设计	48
2.7.3 源程序设计	53
2.7.4 生成目标代码文件	55
2.7.5 加载目标代码文件、设置时钟频率	55
2.7.6 单片机系统的 Proteus 交互仿真	56
2.7.7 Proteus 7.1 与 Keil 8.0 的联调方法	56

2.8 单片机系统的 Proteus 源代码调试仿真	58
2.8.1 存储器窗口	58
2.8.2 观察窗口应用	59
第 3 章 单片机内部资源及其 C 语言编程	62
3.1 中断系统	62
3.1.1 中断系统介绍	62
3.1.2 C51 编写中断服务程序	66
3.1.3 共用中断	69
3.1.4 外部中断的扩充	70
3.2 定时器/计数器	71
3.2.1 定时器/计数器结构	72
3.2.2 定时器/计数器的控制寄存器	73
3.2.3 定时器/计数器工作模式	73
3.2.4 定时器/计数器的初始化	75
3.2.5 定时器/计数器综合应用	76
3.3 并行 I/O 口	77
3.3.1 并行 I/O 口	77
3.3.2 编程实例	82
3.4 串行口及其通信	83
3.4.1 8051 单片机的串行口结构	83
3.4.2 串行口应用	86

设 计 篇

第 4 章 单片机的实用单元电路设计	91
4.1 USTH-51S 单片机学习板简介	91
4.2 独立 I/O 口的单元电路设计	93
4.2.1 硬件原理分析	93
4.2.2 软件编程	94
4.3 流水灯单元电路设计	96
4.3.1 硬件原理分析	96
4.3.2 软件编程	97
4.4 LED 数码管静态显示单元电路设计	98
4.4.1 硬件原理分析	98
4.4.2 软件编程	100
4.5 LED 数码管动态显示单元电路设计	101
4.5.1 硬件原理分析	101
4.5.2 软件编程	102
4.6 蜂鸣器发声单元电路设计	104

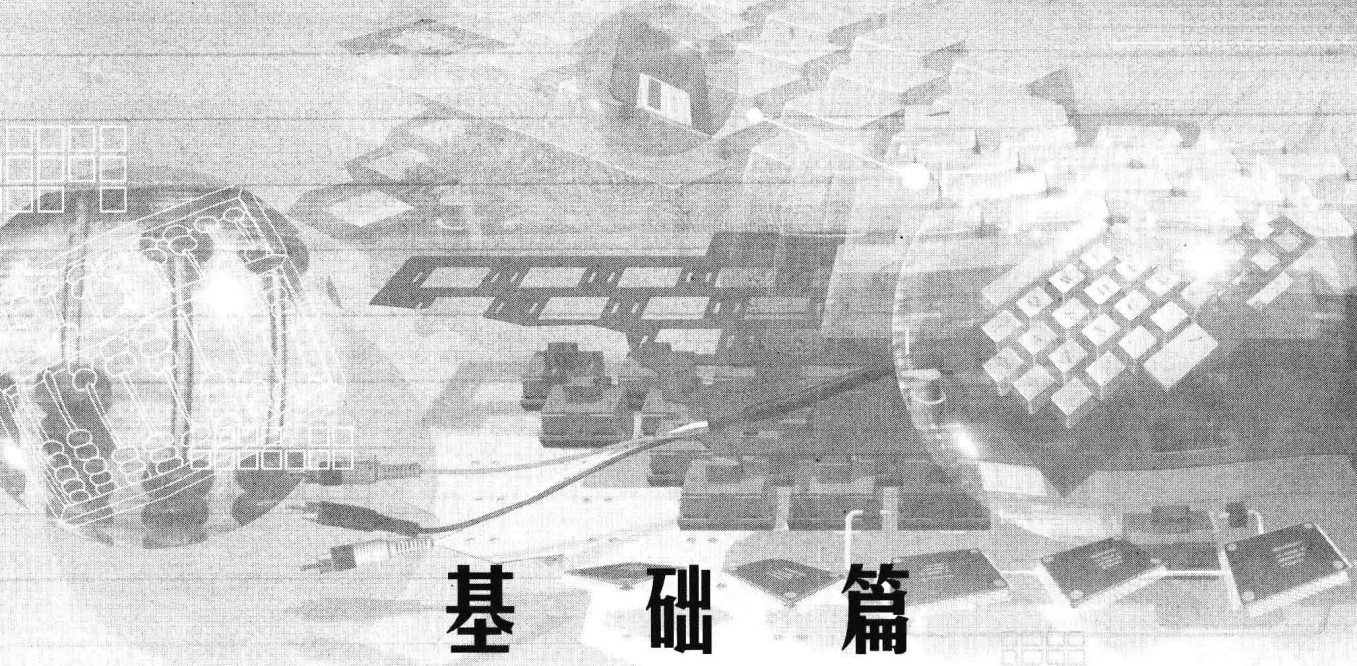
4.6.1	硬件原理分析	105
4.6.2	软件编程	105
4.7	矩阵键盘单元电路设计	107
4.7.1	硬件原理分析	107
4.7.2	软件编程	108
4.8	定时器中断单元电路设计	111
4.8.1	硬件原理分析	111
4.8.2	软件编程	115
4.9	模数转换 ADC0804 单元电路设计	120
4.9.1	硬件原理分析	120
4.9.2	软件编程	122
4.10	数模转换 DAC0832 单元电路设计	125
4.10.1	硬件原理分析	125
4.10.2	软件编程	128
4.11	1602 字符液晶显示单元电路设计	129
4.11.1	硬件原理分析	129
4.11.2	软件编程	132
4.12	12864 图形点阵液晶显示单元电路设计	135
4.12.1	硬件原理分析	135
4.12.2	软件编程	137
4.13	带有 I2C 总线的 AT24C04 单元电路设计	143
4.13.1	硬件原理分析	144
4.13.2	软件编程	146
4.14	串行口通信单元电路设计	150
4.14.1	硬件原理分析	150
4.14.2	软件编程	155
4.15	LED 点阵显示单元电路设计	157
4.15.1	硬件原理分析	157
4.15.2	软件编程	157
4.16	单片机控制继电器单元电路设计	159
4.16.1	硬件原理分析	159
4.16.2	软件编程	160
4.17	红外接收单元电路设计	161
4.17.1	硬件原理分析	161
4.17.2	软件编程	162

应 用 篇

第 5 章	数字温度计设计	167
5.1	功能要求	167

5.2	总体设计	167
5.3	硬件电路设计	167
5.3.1	温度传感器工作原理	169
5.3.2	DS18B20 与单片机的接口电路设计	170
5.3.3	显示电路设计	171
5.4	软件设计	171
5.4.1	主函数	171
5.4.2	DS18B20 复位函数	171
5.4.3	DS18B20 写字节函数	172
5.4.4	DS18B20 读字节函数	172
5.4.5	温度计算转换函数	173
5.4.6	DS18B20 的主要 ROM 命令	173
5.4.7	温度数据的计算处理方法	174
5.5	源程序清单	174
第 6 章	无线数据传输系统设计	180
6.1	功能要求	180
6.2	总体设计	180
6.3	发射电路硬件设计	181
6.3.1	nRF401 无线收发芯片介绍	181
6.3.2	A/D 转换模块设计	183
6.3.3	单片机控制模块设计	186
6.3.4	无线射频模块设计	187
6.3.5	电源模块设计	188
6.4	接收电路硬件设计	189
6.4.1	单片机控制模块设计	189
6.4.2	显示模块设计	189
6.4.3	接口模块设计	191
6.5	发射电路的软件设计	191
6.6	接收电路的软件设计	196
第 7 章	超声波测距仪设计	207
7.1	设计任务	207
7.2	总体设计	207
7.2.1	超声波测距原理	207
7.2.2	超声波速度的确定	208
7.2.3	超声波测距仪总体设计	208
7.3	硬件电路设计	209
7.3.1	元器件的选择	209
7.3.2	超声波发射电路设计	210

7.3.3 超声波接收电路设计	210
7.3.4 语音播报电路设计	211
7.3.5 LCD 显示电路设计	213
7.3.6 单片机控制电路设计	216
7.4 软件设计	217
7.5 源程序清单	218
第 8 章 铁路限速标志设计	230
8.1 功能要求	230
8.2 总体设计	230
8.3 硬件电路设计	231
8.3.1 CPU 的选择	231
8.3.2 显示器电路设计	232
8.3.3 键盘电路设计	233
8.3.4 位置检测电路设计	233
8.3.5 电动机控制电路设计	235
8.3.6 次数检测电路设计	236
8.4 软件设计	236
8.4.1 主程序设计	236
8.4.2 显示程序设计	237
附录 USTH-51S 单片机学习板原理图	241
参考文献	244



基础篇

单片机技术是一门非常有趣的技术，可以通过软件编程控制单片机的各个功能寄存器，控制单片机引脚输入/输出高低电平，从而实现所需的各个功能。现在使用较多的是 MCS-51 单片机，它的资料比较多，市场也很大。但是许多初学者往往苦于找不到正确的学习方法和合适的学习工具而一直在门外徘徊，那么，怎样才能更快更好地学会单片机技术呢？最有效的办法就是理论与实践相结合，我们先通过基础篇来学习单片机系统设计与开发的基础知识。基础篇包含三章，分别介绍单片机应用系统的设计与开发、Keil V808V 和 Proteus 软件的使用、单片机内部资源及其 C 语言编程。

第 1 章 单片机应用系统的设计与开发

目前，MCS-51 系列单片机以其独特的优越性，在智能仪表、工业测控、数据采集、计算机通信等各个领域得到极为广泛的应用。不同用户根据所要完成任务的不同，进行各种单片机应用系统的设计。本章将对应用系统的软、硬件设计和调试等各个方面进行介绍，并给出具体应用实例，以便设计者能更迅速地完成单片机应用系统的设计与开发。

1.1 单片机应用系统的设计方法

单片机应用系统的设计既是一个理论问题，又是一个实际工程问题。它包括自动控制理论、计算技术、计算方法，还包括自动检测技术与数字电路，是一个多学科的综合运用。

单片机系统设计要具备以下几方面的知识和能力。

首先，必须具有一定的硬件基础知识。这些硬件不仅包括各种单片机、存储器及 I/O 接口，而且还包括对仪器或装置进行信息设定的键盘及开关、检测各种输入量的传感器、控制用的执行装置，以及单片机与各种仪器进行通信的接口、打印和显示设备等。

其次，需要具有一定的软件设计能力。能够根据系统的要求，灵活地设计出所需要的程序，主要有数据采集程序、A/D 转换程序、D/A 转换程序、数码转换程序、数字滤波程序，以及各种控制算法及非线性补偿程序等。

再次，需要具有综合运用知识的能力。必须善于将一台智能化仪器或装置的复杂设计任务划分成许多便于实现的组成部分。特别是对软件、硬件的折中问题能够恰当地运用。设计单片机应用系统的一般原理是先选择和组织硬件，构成最小系统。当硬件、软件之间需要折中协调时，通常解决的办法是尽量减少硬件（以便使仪器的价格减到最低）。接着应满足设计中各方对软件的要求。通常情况下，硬件实时性强，但将会增加仪器成本，且结构复杂；

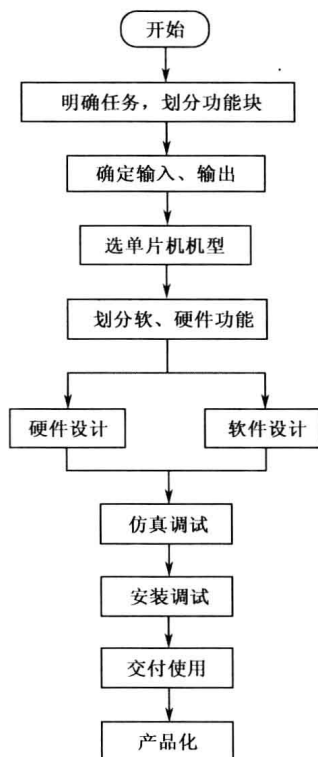


图 1-1 单片机应用系统设计的一般过程

软件可避免上述缺点，但实时性比较差。为保证系统能可靠工作，在软、硬件的设计过程中还应包括系统的抗干扰设计。

最后，还必须掌握生产过程的工艺性能及被测参数的测量方法，以及被控对象的动态、静态特性，有时甚至要建立被控对象的数字模型。

单片机应用系统设计的一般过程如图 1-1 所示。

单片机应用系统设计主要包括下面几方面内容。

(1) 应用系统总体方案设计，包括系统的要求，应用方案的选择，以及工艺参数的测量范围等；

(2) 选择各参数检测元件及变送器；

(3) 建立数学模型及确定控制算法；

(4) 选择单片机，并决定是自行设计还是购买成套设备；

(5) 系统硬件设计，包括接口电路、逻辑电路及操作面板；

(6) 系统软件设计，包括管理、监控程序以及应用程序的设计；

(7) 系统的调试与实验。

本节将就单片机应用系统设计的几个主要方面进行阐述。

1.1.1 系统总体方案的确定

确定单片机应用系统总体方案，是进行系统设计最重要、最关键的一步，总体方案的好坏，直接影响整个应用系统的投资、调节品质及实施细则。总体方案的设计主要是根据被控对象的工艺要求而确定的。由于被控对象多种多样，要求单片机完成的任务也千差万别，所以确定应用系统的总体方案必须根据工艺的要求，结合具体被控对象而定。尽管如此，在总体设计方案中具有一定的共性。

1. 可行性调研

在选择项目时，必须首先进行可行性分析和经济技术论证，基本原则如下：

(1) 技术效果好和经济效益（或社会效益）高；

(2) 技术先进，造价较低；

(3) 可靠性高，维修方便；

(4) 研制周期短；

(5) 操作简便，容易掌握。

可行性调研的目的，是分析完成这个项目的可能性。进行这方面的工作，可参考国内外有关资料，看是否有人进行过类似的工作。若有，则可分析他人是如何进行这方面工作的，有什么优点和缺点，有什么是值得借鉴的；若没有，则需要进一步调研，此时的重点应放在能否实现这个环节，首先从理论上进行分析，探讨实现的可能性，所要求的客观条件是否具备（如环境、测试手段、仪器设备、资金等），然后结合实际情况，确定能否立项的问题。

2. 系统总体方案设计

在进行可行性调研后,如果可以立项,下一步工作就是系统总体方案的设计。首先确定是采用开环系统还是闭环系统,或是数据处理系统。如果是闭环控制系统,则还要确定整个系统是采用直接数字控制(DDC),还是选用计算机监督控制(SCC),或者选用分布式控制(DCS)等。工作的重点应放在该项目的技术难度上,此时可参考这一方面更详细、更具体的资料,根据系统的不同部分和要实现的功能,参考国内外同类产品的性能,提出合理而可行的技术指标,编写出设计任务书,从而完成系统总体方案设计。

3. 设计方案细化

一旦总体方案确定下来,下一步的工作就是将该项目细化,即需明确哪些部分用硬件来完成,哪些部分用软件来完成。由于硬件结构与软件方案会相互影响,因此,从简化电路结构、降低成本、减少故障率、提高系统的灵活性与通用性方面考虑,提倡软件能实现的功能尽可能由软件来完成。但也应考虑以软件代硬件的实质是以降低系统实时性、增加处理时间为代价的,而且软件设计费用、研制周期也将增加,因此系统的软件、硬件功能分配应根据系统的要求及实际情况而合理安排,统一考虑。在确定软件、硬件功能的基础上,设计者的工作就开始涉及一系列的具体问题,如仪器的体积及与具体技术指标相对应的硬件实现方案,软件的总体规划等。在确定人员分工、安排工作进度、规定接口参数后,就必须考虑软件、硬件的具体设计问题。系统软件、硬件设计工作可分开进行,同时并进。

在讨论具体设计问题之前,这里还要强调一下,对于一个具体应用系统的设计,上面这几部分工作是必不可少的,否则,可能导致设计方案的整体更改,甚至可能导致方案无法实现,造成人力、物力的浪费。这一点,对于设计者来讲,应加倍注意。

1.1.2 应用系统的硬件设计

一个单片机应用系统的硬件设计包括两大部分内容:一是单片机系统的扩展部分设计,它包括存储器扩展和接口扩展;二是各功能模块的设计,如信号测量功能模块、信号控制功能模块、人机对话功能模块、通信功能模块等,根据系统功能要求配置相应的A/D、D/A、键盘、显示器、打印机等外围设备。

在进行应用系统的硬件设计时,首要问题是确定电路的总体方案,并需要进行详细的技术论证。

所谓硬件电路的总体设计,即是为实现该项目全部功能所需要的所有硬件的电气连线原理图。初次接触这方面工作的设计人员,往往急于求成,在设计总体方案上不愿花时间,过于仓促地开始制板和调试。这种方法不仅不妥当,而且常常得不偿失。因为就硬件系统来讲,电路的各部分都是紧密相关、互相协调的,任何一部分电路的考虑不充分,都会给其他部分带来难以预料的影响,轻则使系统整体结构受破坏,重则导致硬件总体大返工,由此造成的后果是可想而知的。因此,希望设计者不要吝啬在硬件总体方案上花的时间。从时间上看,硬件设计的绝大部分工作量往往是在最初方案的设计阶段,一个好的设计方

案常常会有事半功倍的效果。一旦硬件总体方案确定下来, 下一步工作就能很顺利进行, 即使需要做部分修改, 也只是在此基础上进行一些完善工作, 不会造成整体返工。

在进行硬件总体方案设计时, 所涉及的具体电路可借鉴他人在这方面进行的工作。因为经过别人调试和实验过的电路往往具有一定的合理性(尽管这些电路常与书籍或手册上提供的电路不完全一致, 但这也可能正是经验所在)。在此基础上, 结合自己的设计目的进行一些修改, 是一种简便、快捷的做法。当然, 有些电路还需自己设计, 完全照搬是不太可能的。

在参考别人的电路时, 需对其工作原理有较透彻的分析和理解, 根据其工作机理了解其适用范围, 从而确定其移植的可能性和需要修改的地方。对于有些关键和尚不完全理解的电路, 需要仔细分析, 在设计之前先进行试验, 以确定这部分电路的正确性, 并在可靠性和精度等方面进行试验, 尤其是模拟电路部分, 更需要进行这方面的工作。

为使硬件设计尽可能合理, 根据经验, 系统的电路设计应注意以下几个方面。

(1) 尽可能选择标准化、模块化的典型电路, 提高设计的成功率和结构的灵活性。

(2) 在条件允许的情况下, 尽可能选用功能强、集成度高的电路或芯片。因为采用这种器件可能代替某一部分电路, 不仅元件数量、接插件和相互连线减少, 体积减小, 使系统可靠性增加, 而且成本往往比用多个元件实现的电路要低。

(3) 注意选择通用性强、市场货源充足的器件, 尤其对需大批量生产的场合, 更应注意这方面的问题。其优点是: 一旦某种元器件无法获得, 也能用其他元器件直接替换或对电路稍作改动后用其他器件代替。

(4) 在对硬件系统总体结构考虑时, 同样要注意通用性的问题。对于一个较复杂的系统, 设计者常常希望将其模块化, 即对中央控制单元, 输入接口、输出接口、人机接口等分块进行设计, 然后采用一定的连接方式将其组合成一个完整的系统。在这种情况下, 连接方式就显得非常重要, 有时可选用通用接口方式, 因为对于这些总线结构的连接目前应用比较广泛, 不少厂家已开发出适合于这些总线结构的接口板, 如输入板、输出板、A/D 板等。在必要的情况下, 选用现成的模板作为系统的一部分, 尽管成本有些偏高, 但会大大缩短研制周期, 提高工作效率。当然, 在有些特殊情况和小系统的场合, 用户必须自行设计接口, 定义连线方式。此时要注意接口协议, 一旦接口方式确定下来, 各个模块的设计都应遵守该接口方式。

(5) 系统的扩展及各功能模块的设计在满足应用系统功能要求的基础上, 应适当留有余地, 以备将来修改、扩展之需。实际上, 电路设计一次成功而不做任何修改的情况是很少的, 如果在设计之初未留有任何余地, 后期很可能因为一点小小的改动或扩展而被迫进行全面返工。举例来说, 在进行 ROM 扩展时, 尽量选用 2764 以上的芯片, 这样不仅将来升级方便, 成本也会降低; 在进行 RAM 扩展时, 为使系统升级或增加内存方便, 系统的 RAM 空间应留足位置, 哪怕多设计一个 RAM 插座, 不插芯片也好; 在进行 I/O 接口扩展时, 也应给出一定的余量, 这样对临时增加一些测量通道或被控对象就极为方便了。另外在电路板设计时, 可适当安排一些机动布线区, 在此区域中安排若干集成芯片插座和金属化孔, 但不布线, 这样在样机研制过程中, 若发现硬件电路有不足之处, 需增加元器件时, 可在机动布线区临时连线完成, 从而避免整个系统返工。在进行模拟信号处理电路设计时,

尤其要注意这一点。因为在调试这类电路时，经常会增加一些电容、电阻等元器件。当然，一旦试验完成，制作电路板时，可以去掉机动布线区。

(6) 设计时应尽可能多做些调研，采用最新的技术。因为电子技术发展迅速，器件更新换代很快，市场上不断推出性能更优、功能更强的芯片，只有时刻注意这方面的发展动态，采用新技术、新工艺，才能使产品具有最先进的性能，不落后于时代发展的潮流。

(7) 在电路设计时，要充分考虑应用系统各部分的驱动能力。一些经验欠缺者往往忽视电路的驱动能力及时序问题，认为原理上可行就行了，其实不然。因为不同的电路有不同的驱动能力，对后级系统的输入阻抗要求也不一样。如果阻抗匹配不当，系统驱动能力不够，将导致系统工作不可靠甚至无法工作。值得注意的是，这种不可靠很难通过一般的测试手段确定，而排除这种故障往往需要对系统做较大的调整。因此，在电路设计时，要注意增加系统驱动能力或减少系统的功耗。

(8) 工艺设计，包括机箱、面板、配线和接插件等，这也是实际进行系统设计人员容易疏忽但又十分重要的一个问题。在设计时要充分考虑到安装、调试和维修的方便。

(9) 系统的抗干扰设计。这个问题在硬件设计中也是十分重要的，有关这方面的内容，将在 1.1.4 节专门讨论。

除了上述几点之外，在应用系统的硬件设计过程中，还需注意以下几方面。

(1) 选择好被测参数的测量元件，它是影响控制系统精度的重要因素之一。测量各种参数的传感器，如温度、流量、压力、液位、位移、重量、速度等，种类繁多、规格各异，因此，要正确地选择测量元件。

(2) 执行机构是单片机控制系统的重要组成部件之一。执行机构的选择一方面要与控制算法匹配，另一方面要根据被控对象的实际情况决定，常用的执行机构有 4 种：电动执行机构具有响应速度快、与单片机接口容易等优点，成为单片机应用系统的主要执行机构；气动调节阀具有结构简单、操作方便、使用可靠、维护容易、防火防爆等优点，广泛用于石油、冶金和电力系统中；步进电动机可以直接接收数字量，而且具有动作速度快、精度高等优点，所以用步进电动机作为执行机构的控制系统越来越多；液压执行机构（如油缸和油马达）将油液的压力能转换成机械能，驱动负载直线或回转运动，能方便地进行无级调速，且高速范围大，控制和调节简单、方便、省力、易于实现自动控制和过载保护。

(3) 过程通道的选择应考虑以下一些问题：被控对象参数的数量，各输入/输出通道是串行操作还是并行操作，各通道数据的传递速率，各通道数据的字长及选择位数，过程通道的结构形式等。

根据系统的复杂程度，MCS-51 应用系统有 3 种典型结构。

(1) 最小应用系统。

(2) 小规模扩展系统：只扩展少量的 RAM 和 I/O 接口，地址在 00H~0FFH 之间。

(3) 大规模扩展系统：需要扩展大量的 ROM、RAM 和 I/O 接口，连接多片扩展芯片。硬件设计的具体步骤如下。

(1) 确定各输入/输出数据的传送方式是中断方式、查询方式还是无条件方式等。

(2) 根据系统需要确定使用何种结构，确定系统中主要电路是最小系统，还是扩展系

统。除单片机外，系统中还需要哪些扩展芯片、模拟电路等。

(3) 资源分配：各输入/输出信号分别使用哪个并行口、串行口、中断、定时器/计数器等。

(4) 电路连接：根据以上各步完成完整的线路连接图。

1.1.3 应用系统的软件设计

在进行应用系统的总体设计时，软件设计和硬件设计应统一考虑，相结合进行。当系统的电路设计定型后，软件的任务也就明确了。

系统中的应用软件是根据系统功能要求设计的。一般来说，单片机中的软件功能可分为两大类：一类是执行软件，能完成各种实质性的功能，如测量、计算、显示、打印、输出控制等；另一类是监控软件，专门用来协调各执行模块和操作者的关系，充当组织调度角色，也称为 Debug 程序，是最基本的调试工具。开发监控程序是为了调试应用程序。监控程序功能不足会给应用程序的开发带来麻烦，反之，用大量精力研究监控程序会贻误开发应用程序。因此，把监控程序控制在适当的规模是明智的。由于应用系统种类繁多，程序编制者风格不一，因此应用软件因系统而异、因人而异。尽管如此，作为优秀的应用软件还是有其共同特点及其规律的。设计人员在进行程序设计时应从以下几个方面加以考虑。

(1) 根据软件功能要求，将系统软件分成若干个相对独立的部分。根据它们之间的联系和时间上的关系，设计出合理的软件总体结构，使其清晰、简洁、流程合理。

(2) 培养结构化程序设计风格，各功能程序实行模块化、子程序化。既便于调试、链接，又便于移植、修改。

(3) 建立正确的数学模型。即根据功能要求，描述出各个输入和输出变量之间的数学关系，它是关系到系统性能好坏的重要因素。

(4) 为提高软件设计的总体效率，以简明、直观的方法对任务进行描述，在编写应用软件之前，应绘制出程序流程图。这不仅是程序设计的一个重要组成部分，而且是决定成败的关键部分。从某种意义上讲，多花一份时间来设计程序流程图，就可以节约几倍源程序编辑调试时间。

(5) 要合理分配系统资源，包括 ROM、RAM、定时器/计数器、中断源等。其中最关键的是片内 RAM 分配。例如对 8051 来讲，片内 RAM 指 00H~7FH 单元，这 128 个字节的功能不完全相同，分配时应充分发挥其特长，做到物尽其用，在工作寄存器的 8 个单元中，R0 和 R1 具有指针功能，是编程的重要角色，避免作为它用；20H~2FH 这 16 个字节具有位寻址功能，可用来存放各种标志位、逻辑变量、状态变量等；设置堆栈区时应事先估算出子程序和中断嵌套的级数及程序中栈操作指令使用情况，其大小应留有余量。若系统中扩展了 RAM 存储器，应把使用频率最高的数据缓冲器安排在片内 RAM 中，以提高处理速度。当 RAM 资源规划好后，应列出一张 RAM 资源详细分配表，以备编程查用方便。

(6) 注意在程序的有关位置处写上功能注释，提高程序的可读性。

(7) 加强软件抗干扰设计，它是提高单片机应用系统可靠性的有力措施。

实时测控程序一般包括以下几方面。