

第 10 章 单片机应用系统设计与开发

前面介绍了单片机的基本结构、功能及其扩展和基本外围设备的接口技术。从单片机应用系统设计的角度看，这些内容仅使我们掌握了单片机的工作状态，或者说，使我们掌握了单片机所提供的软件和硬件资源，以及怎样合理地使用这些资源。这为单片机应用系统设计奠定了基础。除此之外，一个实际的单片机应用系统除需要进行多种配置及其接口连接外，还会涉及到更为复杂的内容和问题，多种类型的电路结构（模拟电路、伺服驱动电路、抗干扰隔离电路等）。因此，单片机应用系统设计应遵循一些基本原则和方法。从一般应用角度来说，了解单片机应用系统的结构、设计的内容与一般方法，对于单片机应用系统的工程设计与开发有着十分重要的指导意义。

10.1 单片机应用系统结构与设计内容

从系统的角度来看，单片机应用系统是由硬件系统和软件系统两部分组成的。硬件系统是指单片机及扩展的存储器、外围设备及其接口电路等。软件系统包括监控程序和各种应用程序。

10.1.1 单片机应用系统的一般硬件组成

由于单片机主要用于工业测控，因而其典型应用系统用于测控对象的前向传感器输入通道、后向伺服控制输出通道以及基本的人机对话通道。典型单片机应用系统结构框图如图 10-1 所示。

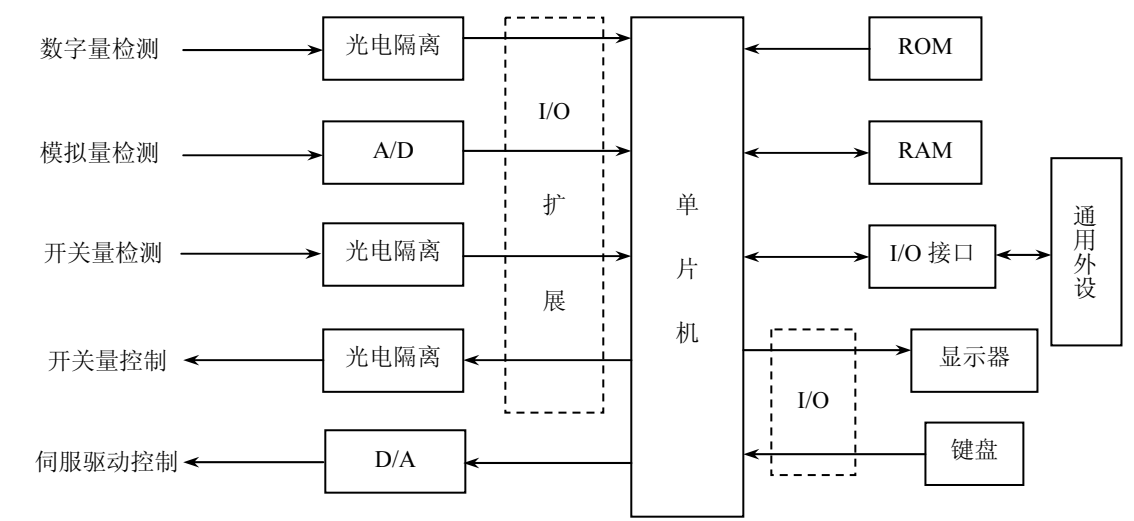


图 10-1 典型单片机应用系统结构框图



1. 前向通道的组成及特点

前向通道是单片机与测控对象相连的部分，是应用系统的数据采集输入通道。

来自被控对象的现场信息是各种各样的，按物理量的特征可分为模拟和数字量、开关量两种。

对于数字量（频率、周期、相位、计数）的采集，输入比较简单。它们可直接作为计数输入、测试输入、I/O 输入或中断输入进行事件计数、定时计数，实现脉冲的频率、周期、相位及计数测量。对于开关量的采集，一般通过 I/O 口线或扩展 I/O 线直接输入。一般被控对象都是交变电流、交变电压、大电流系统，而单片机属于数字弱电系统，因此，在数字量和开关量采集通道中，要用隔离器进行隔离（如光电耦合元器件）。

模拟量输入量通道的结构比较复杂，一般包括变换器、隔离放大器、滤波器、采用保持器、多路电子开关、A/D 转换器及其接口电路，如图 10-2 所示。

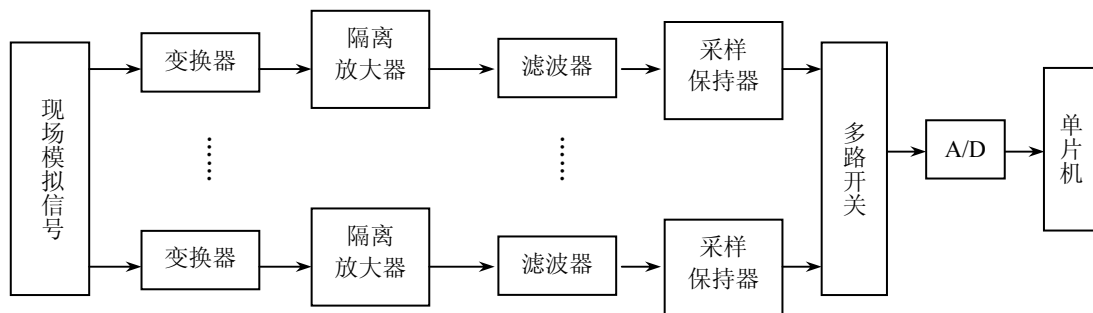


图 10-2 模拟信号采集通道结构图

（1）变换器：是各种传感器的总称，它采集现场的各种信号并将其变换成电信号（电压信号或电流信号），以满足单片机的输入要求，现场信号各种各样，有电信号，如电压、电流、电磁量等，也有非电量信号，如温度、湿度、压力、流量位移量等。对于不同物理量，应选择相应的传感器。

（2）隔离放大与滤波：传感器的输出信号一般是比较微弱的，不能满足单片机系统的输入要求，要经过放大处理后才能作为单片机系统的采集输入信号；现场信息来自各种工业现场，夹带大量的噪音干扰信号，为提高单片机应用系统的可靠性，必须隔离或削减干扰信号，这是整个系统抗干扰设计的重点部位。

（3）采样保持器：向前通道中的采样保持器有两个作用，一是实现多路模拟信号的同时采集，二是消除 A/D 转换器的“孔径误差”。

一般的单片机应用系统都是用一个 A/D 转换器分时对多路模拟信号进行转换并输入给单片机，而控制系统又要求单片机对同一时刻的现场采样值进行处理，否则将产生很大的误差。用一个 A/D 转换器同时对多路模拟信号进行采样是由采样保持器来实现的。采样保持器在单片机的控制下，在某一时刻采样同时，它可接收一路模拟信号的值，并能保持该瞬时值，直到下一次重新采样。

A/D 转换器把一个模拟量转换成数字量总要经过一段时间。A/D 转换器从接通模拟信号开始转换到转换结束，将输出稳定的数字量，这一段时间称为孔径时间。对于一个动态模拟信号，在 A/D 转换器接通的孔径时间里，输入模拟信号值是不确定的，从而会引起输出的不确定性



误差。在 A/D 转换器前加设采样保持器,在孔径时间里,使模拟信号保持某一个瞬时值不变,从而可消除孔径误差。

(4) 多路开关:用多路开关实现一个 A/D 转换器分时对多路模拟信号进行转换。多路开关是受单片机控制的多路模拟电子开关,某一时刻需要对某路模拟信号进行转换,由单片机向多路开关发出路地址信息,使多路开关把该路模拟信号与 A/D 转换器接通,其他路模拟信号与 A/D 转换器不接通,实现有选择的转换。

(5) A/D 转换器:是前向通道中模拟系统与数字系统连接的核心部件。

综上所述,前向通道具有以下特点:

(1) 与现场采集对象相连,是现场干扰进入的主要通道,是整个系统干扰设计的重点部位。

(2) 由于所采集的对象不同,有开关量、模拟量、数字量,而这些都是由安放在测量现场的传感、变换装置产生的,许多参量信号不能满足单片机输入的要求,故有大量的、形式多样的信号变换调节电路,如测量放大器、I/F 变换、A/D 转换、放大、整形电路等。

(3) 前向通道是一个模拟、数字混合电路系统,其电路功耗小,一般没有功率驱动要求。

2. 后向通道的组成与特点

后向通道是应用系统的伺服驱动通道。作用于控制对象的控制信号通常有两种:一种是开关量控制信号,另一种是模拟控制信号。开关量控制信号的后向通道比较简单,只需采用隔离器进行隔离及电平转换。模拟控制信号的后向通道需要进行 D/A 转换,隔离放大、功率驱动等处理。

后向通道具有以下特点:

(1) 后向通道是应用系统的输出通道,大多数需要功率驱动。

(2) 靠近伺服驱动现场,伺服控制系统的大功率负荷易从后向通道进入单片机系统,故后向通道的隔离对系统的可靠性影响很大。

(3) 根据输出控制的不同要求,后向通道电路多种多样,如模拟电路、数字电路、开关电路等,输出信号形式有电流输出、电压输出、开关量输出及数字量输出等。

3. 人机通道的结构及其特点

单片机应用系统中的人机通道是用户为了对应用系统进行干预(如启动、参数设置等)以及了解应用系统运行状态所设置的对话通道,主要有键盘、显示器、打印机等通道接口。

人机通道具有以下特点:

(1) 由于通常的单片机应用系统大多是小规模系统,因此,应用系统中的人机对话通道以及人机对话设备的配置都是小规模,如微型打印机、功能键、拨码盘、LED/LCD 显示器等。若需高水平的人机对话配置,如通用打印机、CRT、硬盘、标准键盘等,则往往将单片机应用系统通过总线与通用计算机相连,享用通用计算机的外围人机对话设备。

(2) 在单片机应用系统中,人机对话通道及接口大多采用内总线形式,与计算机系统扩展密切相关。

(3) 人机通道接口一般都是数字电路,电路结构简单,可靠性好。

4. 相互通道及其特点

单片机应用系统中的相互通道是解决计算机系统间相互通信的接口。在较大规模的多机测控系统中,就需要设计相互通道接口。

相互通道设计中需要考虑的问题如下:



(1) 中、高档的单片机大多设有串行口,为构成应用系统的相互通道提供了方便。

(2) 单片机本身的串行口只为相互通道提供了硬件结构及基本的通信方式,并没有提供标准的通信规程。故利用单片机串行口构成相互通道时,要配置比较复杂的通信软件。

(3) 在很多情况下,采用扩展标准通信控制芯片来组成相互通道。例如,用扩展 8250、8251、SIO、8273、MC6850 等通用通信控制芯片来构成相互通信接口。

(4) 相互通信接口都是数字电路系统,抗干扰能力强,但大多数都需要远距离传输,故需要解决长线传输的驱动、匹配、隔离等问题。

10.1.2 单片机应用系统的设计内容

单片机应用系统设计包含硬件设计与软件设计两部分。硬件设计又包括单片机系统扩展和配置。具体的设计内容包括:

1. 单片机系统设计

单片机应用系统包含强大的功能,但往往不能满足一个实际应用系统的功能要求。有些单片机本身就缺少一些功能,如 MCS-51 系列中的 8031、8032 片内无程序存储器,所以要通过系统控制来构成一个完善的计算机系统,它是单片机应用系统中的核心部分。系统扩展方法、内容、规模与所用的单片机系列及供应状态有关。单片机具有较强的外部扩展通信能力,能方便地扩展至应用系统所要求的规模。在单片机应用系统中,单片机系统控制的设计内容如下:

(1) 最小系统设计:给单片机配以必要的器件构成单片机最小的系统。如 MCS-51 系列单片机片内有程序存储器的机型,只需在片外配上电源、复位电路、振荡电路,这样便于对单片机系统进行测试与调试。

(2) 系统扩展设计:在单片机最小系统的基础上,再配置能满足应用系统要求的一些外围功能器件。

2. 通道与接口设计

由于通道大都是通过 I/O 口进行配置的,与单片机本身的联系不甚紧密,因此大多数接口电路都能方便地移植到其他类型的单片机应用系统中去。

3. 系统干扰设计

抗干扰设计要贯穿到应用系统设计的全过程。从具体方案、器件选择到电路系统设计,从硬件系统设计到软件系统设计,都要把抗干扰设计列为一项重要工作。

4. 应用软件设计

应用软件设计是根据系统功能要求,采用汇编语言或高级语言进行设计。

10.2 单片机应用系统的一般设计方法

10.2.1 确定系统的功能与性能

由需求调查可以确定出单片机应用系统设计的目标,这一目标包括系统功能和系统性能。

系统功能主要有数据采集、数据处理、输出控制等。每一个功能又可细分为若干个子功



能。比如数据采集可分为模拟信号采样与数字信号采样,模拟信号采样与数字信号采样在硬件支持与软件控制上是有明显差异的。数据处理可分为预处理、功能性处理、可干扰等子功能,而功能性处理还可以继续划分为各种信号处理等。输出控制按控制对象的不同可分为各种控制功能,如继电器控制、D/A 转换控制、数码管显示控制等。

在确定了系统的全部功能之后,就应确定每种功能的实现途径,即哪些功能由硬件完成,哪些功能由软件完成,这就是系统软、硬件功能的划分。

系统性能主要由精度、速度、功耗、体积、重量、价格、可靠性等技术指标来衡量。系统研制前,要根据需求调查结果给出的上述各指标来衡量系统性能。一旦这些指标被确定下来,整个系统将在这些指标限度下进行设计。系统的速度、体积、重量、价格、可靠性等指标会影响系统软、硬件功能的划分。系统功能尽可能用硬件完成,这样可提高系统的工作速度,但系统的体积、重量、功耗、硬件成本都相应地增大,而且还增加了硬件所带来的不可靠因素。可用软件功能尽可能代替硬件功能,以使系统体积、重量、功耗、硬件成本降低,并可提高硬件系统的可靠性,但可能会降低系统的工作速度。因此,在进行系统功能的软、硬件划分时,一定要依据系统性能指标综合考虑。

10.2.2 确定系统基本结构

单片机应用系统结构一般是以单片机为核心的。在单片机外部总线上要扩展连接相应功能的部件,配置相应外部设备和通道接口。因此,系统中单片机选型、存储器分配、通道划分、输入/输出方式及系统中硬、软件功能划分等都对单片机应用系统结构有直接影响。

1. 单片机选型

不同系列、不同型号的单片机内部结构、外部结构特征均不同,而应用系统中的单片机系列或型号直接决定其总体结构。因此,在确定系统基本结构时,首先要选择单片机的系列或型号。

选择单片机应考虑以下几个主要因素:

(1) 单片机性价比。应根据应用系统的要求和各种单片机的性能,选择最容易实现产品技术指标的机型,而且能达到较高的性能价格比。性能选得过低,将给组成系统带来麻烦,甚至不能满足要求;性能选得过高,就可能大材小用,造成浪费,有时还会带来问题,使系统复杂化。

(2) 开发周期。选择单片机时,要考虑具有新技术的新机型,更应考虑应用技术成熟、由较多软件支持、能得到相应单片机开发工具的比较成熟的机型。这样可借鉴许多现成的技术,移植一些现成软件,可以节省人力、物力,缩短开发周期,降低开发成本,使所开发的系统具有竞争力。

在此需要特别指出的是:在选择单片机芯片时,一般选择内部不含 ROM 的芯片比较合适,通过外部扩展 EPROM 和 RAM 即可构成系统,这样不需专门的设备即可固化应用程序。另外,可以选择内部有 Flash 存储器的机型,Flash 存储器在不加电的情况下能长期保存信息,又能在线进行擦除和重写。当设计的应用系统批量比较大时,可选择带 ROM、EPROM、OTPROM 或 EEPROM 等的单片机,这样可使系统更加简单。通常的做法是在软件开发过程中采用 EPROM 型或 Flash 存储器芯片,而最终产品采用 OTPROM 型芯片(一次性可编程 EPROM 芯片),这样可以提高产品的性能价格比。



总之,对单片机芯片的选择绝不是传统意义上的器件选择,它关系到单片机应用系统的整体方案、技术标准、功耗、可靠性、外设接口、通信方式、产品价格等。所以,设计人员必须反复推敲,慎重选择。

2. 存储空间分配

存储空间分配既影响单片机应用系统的硬件结构,也影响软件的设计及系统调试。

不同的单片机具有不同的存储空间分布。8051单片机的程序存储器与数据存储器空间相互独立,工作存储器、特殊功能存储器与内部数据存储器共享一个存储空间,I/O端口则与外部数据存储器共享一个空间。8098单片机的片内RAM程序存储区、数据存储区、I/O端口全部使用同一个存储空间。总的来说,大多数单片机都存在不同类型的器件共享同一个存储空间的问题。因此,在系统设计时就要合理地给系统中的各种部件分配有效的地址空间,以便简化译码电路,并使CPU能准确地访问到指定部件。

3. I/O通道划分

单片机应用系统中通道的数目及类型直接决定系统结构。设计中应根据包括对象所要求的输入/输出信号的数目及类型,确定整个应用系统的通道数目及类型。

4. I/O方式的确定

采用不同的输入/输出方式,对单片机应用系统的硬、软件要求是不同的。在单片机应用系统中,常用的I/O方式主要有无条件传送方式(程序同步方式)、查询方式和中断方式。这三种方式的硬件要求和软件要求结构各不相同,且存在着明显的优缺点差异。在一个实际应用中,选择哪一种I/O方式,要根据具体的外设工作情况和应用系统的性能技术指标综合考虑。一般来说,无条件传送方式只适用于数据变化非常缓慢的外设,这种外设的数据可视为常态设计;中断方式处理器效率较高,但硬件结构稍复杂一些;询问方式硬件价格较低,但处理器效率比较低,速度比较慢。在一般的应用系统中,由于速度要求不高,控制对象也较少,此时,大多采用询问方式。

5. 软、硬件功能划分

同一般的计算机系统一样,单片机应用系统的软件和硬件在逻辑上是等效的。具有相同功能的单片机应用系统,其软、硬件功能可以在很宽的范围内变化。一些硬件电路的功能可以由软件来实现,反之亦然。在应用系统设计中,系统的软、硬件功能划分要根据系统的要求而定,多用硬件来实现一些功能,这样可以提高利用率和系统的灵活性和适应性。相反,若用软件来实现某些硬件功能则可以节省硬件开支,提高灵活性和适应性,但速度要下降,软件设计费用和所需存储容量要增加。因此,在总体设计时,必须权衡利弊,仔细划分应用系统中的硬件和软件的功能。

10.2.3 单片机应用系统硬件、软件的设计原则

1. 硬件系统设计原则

一个单片机应用系统的硬件电路设计包括两部分内容:一是单片机系统扩展,即单片机内部的功能单元(如程序存储器、数据存储器、I/O、定时计数器、中断系统等)的容量不能满足应用系统的要求时,必须在片外进行扩展,选择适当的芯片、设计相应的扩展连接电路;二是系统配置,即按照系统功能要求配置外围设备,如键盘、显示器、打印机、A/D转换器、D/A转换器等,设计合适接口电路。



系统扩展和配置设计应遵循下列原则：

(1) 尽可能选择典型通用的电路，并符合单片机的常规用法，为硬件系统的标准化、模块化奠定良好的基础。

(2) 系统的扩展与外围设备配置的水平应充分满足应用系统当前的功能要求，并留有适当余地，便于以后进行功能的扩充。

(3) 硬件结构应结合应用软件方案一并考虑。硬件结构与软件方案会相互影响，考虑的原则是：软件能实现的功能尽可能由软件实现，即尽可能地以软件代替硬件，以简化硬件结构，降低成本，提高可靠性。但必须注意，由软件实现的硬件功能，其响应时间要比直接用硬件长。因此，某些功能选择以软件代替硬件实现时，应考虑系统响应速度、实时要求等相关的技术指标。

(4) 整个系统中相关的器件要尽可能做到性能匹配。例如，选用的晶振频率较高时，存储器的存取时间就短，应选择允许存取速度较快的芯片；选择 CMOS 芯片单片机构成低功耗系统时，系统中的所有芯片都应该选择低功耗产品。如果系统中相关的器件性能差异很大，则系统综合性能将降低，甚至不能正常工作。

(5) 可靠性及抗干扰设计是硬件设计中不可忽视的一部分，具体包括芯片、器件选择、去耦滤波，印刷电路板布线、通道隔离等。如果设计中只注重功能实现，而忽视可靠性及抗干扰设计，则到头来只能是事倍功半，甚至造成系统崩溃。

(6) 单片机外接电路较多时，必须考虑其驱动能力。解决的办法是增强驱动能力，增加总线驱动器或者减少芯片功耗，降低总线负载。

2. 应用软件设计的特点

应用系统中的应用软件是根据系统功能设计的，应可靠地实现系统的各种功能。应用系统种类繁多，应用软件各不相同，但是一个优秀的应用系统的软件应具有以下特点：

(1) 软件结构清晰、简洁，流程合理。

(2) 各功能程序实现模块化、系统化，这样既便于调试、连接，又便于移植、修改和维护。

(3) 程序存储区、数据存储区规划合理，既能节约存储容量，又能给程序设计与操作带来方便。

(4) 运行状态实现标志化管理。各个功能程序运行状态、运行结果以及运行需求都设置状态标志以便查询，程序的转移、运行、控制都可以通过状态标志条件来控制。

(5) 经过调试修改后的程序应进行规范化，除去修改“痕迹”。规范化的程序便于交流、借鉴，也为今后的软件模块化、标准化打下基础。

(6) 实现全面软件抗干扰设计。软件抗干扰设计是计算机应用系统提高可靠性的有力措施。

(7) 为了提高运行的可靠性，在应用软件中设置自诊断程序，在系统运行前先运行自诊断程序，用以检查系统各特征参数是否正常。

10.2.4 硬件设计

单片机应用系统的硬件设计是围绕着单片机功能扩展和外围设备配置及其接口而展开的。硬件设计主要包括下面几部分。

1. 程序存储器

若单片机内无程序存储器或存储容量不够，则需外部扩展程序存储器。外部扩展存储器



通常选用 EPROM 或 EEPROM。EPROM 集成度高、价格便宜,EEPROM 则编程容易。当程序容量较小时,使用 EEPROM 较便宜;当程序量较大时,采用 EPROM 更经济。

2. 数据存储器

数据存储器利用 RAM 构成,大多数单片机都提供了小容量的片内数据存储区,只有当片内数据存储区不够用时,才扩展外部数据存储器。

存储器的设计原则是,在存储容量满足的前提下,尽可能减少存储芯片的数量,建议使用大容量的存储芯片以减少存储器芯片数目,但应避免盲目地扩大存储器容量。

3. I/O 接口

由于外设多种多样,使得单片机与外设之间的接口电路也各不相同,因此,I/O 接口常常是单片机应用系统中设计最复杂也是变化最多的部分之一。

I/O 接口大致可归类为并行接口、串行接口、模拟采集通道(接口)、模拟输出通道(接口)等。目前有些单片机已将上述各接口集成在单片机内部,使 I/O 接口的设计大大简化。系统设计时,可以选择含有所需接口的单片机。

4. 译码电路

当需要外部扩展电路时,就需要设计译码电路。译码电路要尽可能简单,这就要求存储空间分配合理,译码方式选择得当。

考虑到修改方便与保密性,译码电路除了可以使用常规的门电路、译码器来实现外,还可以利用只读存储器与可编程门阵列来实现。

5. 总线驱动器

如果单片机外部扩展的器件较多,负载过重,就要考虑设计总线驱动器。比如,51 单片机的 P0 口负载能力为 8 个 TTL 芯片,P2 口负载能力为 4 个 TTL 芯片,如果 P0、P2 实际连接的芯片数目超过上述定额,则必须在 P0、P2 口增加总线驱动器来提高它们的驱动能力。P0 口应使用双向数据总线驱动器(如 74LS245),P2 口可使用单向总线驱动器(如 74LS244)。

6. 抗干扰电路

针对可能出现的各种干扰,应设计抗干扰电路。在单片机应用系统中,一个不可缺少的抗干扰电路就是抗电源干扰电路。其最简单的实现方式是在系统弱电部分(以单片机为核心)的电源入口对地跨接一个大电容(100 μ F 左右)与一个小电容(0.1 μ F 左右),在系统内部芯片的电源端对地跨接一个小电容(0.01~0.1 μ F)。

另外,可以采用隔音放大器、光电隔离器件抗共地干扰;采用差分放大器抗共模干扰;采用平滑滤波器抗白噪声干扰;采用屏蔽手段抗辐射干扰等。

要注意的是,在系统硬件设计时,要尽可能充分利用单片机的片内资源,使设计的电路向标准化、模块化方向靠拢。

硬件设计后,应画出硬件电路原理图,并编写硬件设计说明书。

10.2.5 软件设计

整个单片机应用系统是一个整体,在进行应用系统总体设计时,软件设计和硬件设计应统一考虑,相结合进行。当系统的硬件电路设计定型后,软件的任务也就明确了。

一个应用系统中的软件一般是由系统的监控程序和应用程序两部分构成的。其中,应用程序是用来完成诸如测量、计算、显示、打印、输出控制等各种实质性功能的软件;系统监控



程序是控制单片机系统按预定操作方式运行的程序,它负责组织调度各应用程序模块,完成系统自检、初始化、处理键盘命令、处理接口命令、处理条件触发和显示等功能。此外,监控程序还用来监视系统的运行状态。单片机应用系统中的软件一般是用高级语言与汇编语言混合编写的,编写程序时常常与输入、输出接口设计和存储器扩展交织在一起。因此,软件设计是系统开发过程中最重要也是最困难的任务,它直接关系到系统功能和性能的实现。

系统软件设计时,应根据系统软件功能要求,将系统软件分成若干个相对独立的部分,并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。通常在编制程序前,先根据系统输入和输出变量建立起正确的数学模型,然后画出程序流程框图,流程框图应结构清晰、简洁、合理。画流程框图时,还要对系统资源作具体的分配和说明。编制程序时,一般采用自顶向下的程序设计技术,先设计监控程序,再设计各应用程序模块。多功能程序应模块化、子程序化,这样不仅便于测试和连接,还便于修改和移植。

10.2.6 资源分配

合理地资源分配关系到能否充分发挥单片机的性能,能否有效、正确地编制应用程序。

一个单片机应用系统所拥有的硬件资源分片内和片外两部分。片内资源是指本身包括的中央处理器、程序存储器、数据存储器、定时器/计数器、中断系统、I/O 接口以及程序通信接口等。这部分硬件资源的种类很多,不同公司、不同类型的单片机之间差别很大,当设计人员选定某种型号的单片机进行系统设计时,应充分利用片内的各种硬件资源。但是在应用中,若片内的这些硬件资源不够使用,就需要在片外加以扩展,通过系统扩展,单片机应用系统具有了更多的硬件资源,因而有了更强的功能。

由于定时器/计数器、中断器等资源的分配比较容易,因此下面只介绍 ROM/EPROM 资源和 RAM 资源的分配。

1. ROM/EPROM 资源的分配

ROM/EPROM 用于存放程序和数据表格。按照 51 单片机的复位及中断入口的规定,002FH 以前的地址单元作为中断、复位入口地址区。在这些单元中,一般都设置了转移指令,用于转移到相应的中断服务程序或复位启动程序。当程序存储器中存放的功能程序及子程序数量较多时,应尽可能为它们设置入口地址表。一般将常数、表格集中设置在表格区,二次开发扩展区则尽可能放在高位地址区。

2. RAM 资源的分配

RAM 分为片内 RAM 和片外 RAM,片外 RAM 的容量比较大,通常用来存放批量大的数据,如采样结果数据;片内 RAM 容量较少,应尽量重叠使用,比如数据暂存区与显示、打印缓冲区重叠。

对于 8051 单片机来说,片内 RAM 是指 00H~7FH 单元,这 128 个单元的功能并不完全相同,分配时应注意发挥各自的特点,做到物尽其用。

00H~1FH 这 32 个字可以作为工作寄存器组,在工作寄存器的 8 个单元格中,R0 和 R1 具有指针功能,是编程的重要角色,应充分发挥其作用。系统上电复位时,置 PSW=00H,当前工作寄存器为 0 组,而工作寄存器组 1 为堆栈,并向工作寄存器组 2、3 延伸。若在中断服务器中也要使用 R1 寄存器且不将原来的数据冲掉,则可在主程序中先将堆栈空间设置在其他位置,然后在进入中断服务器程序后选择工作寄存器组 1、2 或 3,这时若再执行诸如“MOV



R1, #00H”指令, 就不会冲掉 R1 (01H 单元) 中原来的内容, 因为这时 R1 的地址已变为 09H、11H 或 19H。在中断服务程序结束时, 可重新选择工作寄存器组 0。因此, 通常可在应用程序中安排主程序及调用的之程序使用工作寄存器组 0, 而安排定时器溢出中断、外部中断、串行口中断使用工作寄存器组 1、2 或 3。

10.3 单片机应用系统的开发

10.3.1 单片机应用系统的仿真

单片机应用系统经过预研、总体设计、软/硬件开发、制版、元器件安装和代码下载(固化)后, 系统就可以运行了。但是一次成功几乎是不可能的, 多少会出现一些硬件、软件上的错误, 这都需要调试来发现错误并纠正。通常, 单片机程序调试都需要借助被称为仿真系统或开发系统的专用工具来进行。一个单片机在线仿真器应具备的功能有:

- (1) 能输入和修改应用程序。
- (2) 能对用户系统硬件电路进行检查与诊断。
- (3) 将程序代码编译为目标码并固化或下载到系统中去。
- (4) 能以单步、断点和连续方式运行程序, 正确反映用户程序执行的中间结果。
- (5) 最好不占用用户单片机资源。
- (6) 提供足够的仿真 RAM 空间作为用户的程序存储器, 并提供足够的 RAM 空间作为用户的数据存储器。

- (7) 齐全的软件开发工具, 如交叉汇编、连接、固化和下载, 甚至反编译等。

系统仿真调试的目标是检测并排除故障, 检测并修正模块化软件。

对于一些小系统, 也可以不使用专门的仿真器, 而是直接采用写入装置, 将目标码写入到系统的程序存储器中。如果采用具有 Flash 存储器和支持 ISP 的单片机芯片, 甚至只需要一个编程/下载电缆, 利用专门的下载软件, 就可以通过 ISP 插座将目标码下载到具有 Flash 存储器的单片机芯片中, 然后通过直接运行判断硬件/软件的正确性。

10.3.2 单片机应用系统的制版

单片机应用系统的制版, 其实质就是将设计的硬件电路通过 EDA (如 Protel 等) 绘制原理图, 并形成 PCB 制版图, 最后检验无误后将 PCB 图交给制版公司, 加工制造电路板。

10.3.3 单片机应用系统的调试

单片机应用系统的调试包括硬件和软件调试两部分, 但是它们并不能完全分开。一般的方法是先排除明显的硬件故障, 在进行综合调试, 排除可能的软件、硬件故障。

1. 静态硬件调试

将器件安装/焊接到电路板, 进行系统调试前, 应做好以下几方面的工作:

- (1) 拿到电路板后, 先要检查加工质量, 并确保没有任何制造方面的错误, 如短路和断路, 尤其要避免电源短路。



(2) 元器件在安装前要逐一检查。

(3) 完成焊接后，应先空载上电（芯片座上不插芯片），并检查各引脚的电位是否正确。若一切正常，方可在断电的情况下将芯片插入，再次检查各引脚的电位及逻辑关系。

2. 系统的调试

硬件的静态调试后，就可以进行系统的调试，一个经典的调试方案是：把整个应用系统按功能进行分块，如系统控制模块、A/D 模块和监控模块等。针对不同的功能模块，用测试程序并借助万用表、示波器及逻辑笔等来检测硬件电路的正确性。

系统硬件的调试可采用监控命令法和程序调试法 2 种，调试的内容包括以下几方面：

- (1) 外部数据存储器 RAM 的调试。
- (2) 程序存储器的调试。
- (3) 输出功能模块的调试。
- (4) I/O 接口芯片的调试。
- (5) 外部中断与定时器的调试。
- (6) 键盘的调试。
- (7) 显示器的调试。
- (8) A/D 和 D/A 的调试。
- (9) 串行通信口的调试。

10.3.4 单片机应用系统的编程、下载与运行

(1) 应用程序编程。可采用多种形式编写应用程序的源程序，如采用文本编写器、KeilC 等。

(2) 源程序的汇编。可采用编译程序和交叉编译程序进行汇编，并将其转换为目标码。

(3) 目标码的下载与运行。可借助仿真器、写入器或是利用 ISP 的配套软件进行。调试完成后，将单片机应用系统设置到运行状态，程序就开始自行运行了。

10.4 单片机在温度数据采集系统中的应用

在科研、生产和人们的日常生活中，模拟量的测量和控制是很常见的。为了对温度、压力、流量、速度、位移等物理量进行测量和控制，都是通过传感器把上述物理量转换成能模拟物理量的信号，即模拟电信号。将模拟电信号经过处理并转换成计算机能识别的数字量，送进计算机，这就是数据采集。计算机将采集来的数字量根据需要进行不同的辨识、运算，得出所需要的结果，这就是数据处理。数据处理结果显示于显示器或屏幕上，或由打印机打印在纸上，以便对某些物理量进行监视，再将数据处理结果的数字量转换成模拟信号去控制某些物理量，这就是监控。这一数据采集、计算机处理、数—模转换的监控系统，就是一种数据采集处理系统。下面主要介绍一种基于 AD590 的温度巡回检测单片机控制系统。

10.4.1 系统硬件设计

基于 AD590 的温度巡回检测系统框图如图 10-3 所示，各点温度由 AD590 温度传感器检



测, 并经屏蔽线传送到模拟开关, 由模拟开关切换后, 经放大器放大, 再由转换器转换成数字信号。8031 单片机读取转换的数字信号并进行必要处理后, 将结果存入内存 RAM 供打印机打印, 同时送 LED 显示器显示。另外, 8031 还将依据由键盘输入的温度上下限报警温度值对所有检测的温度进行越限声光报警处理。

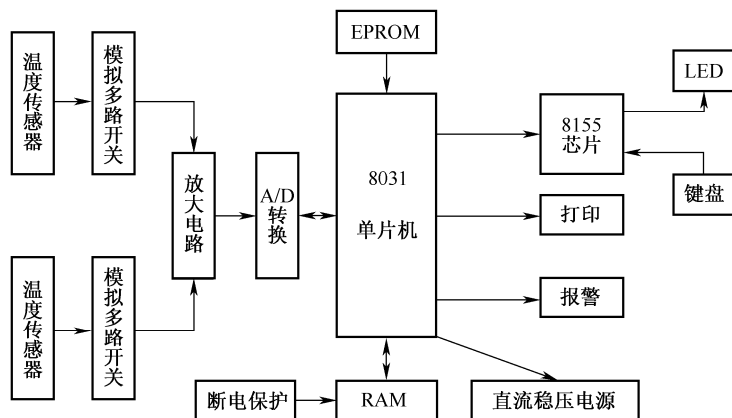


图 10-3 检测系统框图

1. AD590 温度传感器的测温原理

AD590 是一种两端集成电路式半导体温度传感器, 输出电流与它所感受的温度成线性关系, 工作电压可以从 +4V~+30V 范围内选用, 测温范围为 -55℃~150℃, 属于低温传感器。它与其他形式的温度传感器相比, 不存在线性化问题; 与热电阻相比, 不需要设计输入电桥和微弱信号放大器; 与热电偶相比, 不需要进行冷端补偿, 而且它是以高阻抗恒流源形式出入, 传感线上的压降不影响输出电流值, 可以进行远距离传输。因此, 它具有使用方便、抗干扰能力强的特点, 特别适用于较远距离的温度巡回检测系统。本节中, 系统根据待检测温度点分布的特点和要求, 选用 AD590 作为温度传感器。

AD590 是一恒流源器件, 输出的电流值与它所测的绝对温度有精确的线性关系。由于厂家生产时采用激光微调来校正集成电路内的薄膜电阻, 使其在 0℃ (对应绝对温度为 273.2K) 输出电流为 273.2μA, 灵敏度为 1μA/K, 当其感受温度升高或降低时, 输出电流 1μA/K 的速率增大或减小, 从而将被测温度线性转换为电流形式输出, 在测量线路中, 将其电流转换为电压, 则可用电压形式来表示对应的温度大小。由于 AD590 输出电流设计为与开氏温标对应, 而且工作电压范围大, 因此, 在实际应用中应注意以下几个问题。

(1) AD590 在摄氏零度时, 输出电流值为 273.2μA, 它与热力学温度 273.2K 相对应。而人们习惯用摄氏温标表示温度, 摄氏温标与开氏温标的转换关系即 $T(K)=273.2+t(^{\circ}C)$, 在信号处理时, 应将开始温度转换为摄氏温度。

(2) AD590 的工作电压虽可以在 +4V~+30V 的范围内选用, 但某一工作电压一经确定后, 应尽可能使其稳定, 因为工作电压波动将引起 AD590 输出电流在一定程度上的相对漂移, 造成测量误差。

(3) AD590 输出电流在远距离传输时, 虽然它对导线产生的压降不敏感, 但应避免传输导线受电磁干扰影响产生感应电势而导致回路电流变化, 造成测量误差。



2. 采样放大电路

检测系统采样放大电路的设计如图 10-4 所示, 48 个温度传感器 AD590 检测 48 点温度, 6 个 CD4051 模拟开关对应控制采样每一个温度信号, 采样信号通过 CD4051 由同相放大器 741 进行放大, 放大后的信号送 A/D 转换器。模拟开关 CD4051 的选通地址 A0~A2 由 8031 P0 口的低位地址 P0.0~P0.2 控制, 而每片 CD4051 的片选信号 INH 分别由 8031 P1 口 P1.0~P1.5 来控制。AD590 将温度转换为电流后, 经取样电阻 R_1 和 R_{1w} 转换为电压信号, 取样电压 R_p 设计为 $1k\Omega$, 温度每变化一度, 取样电阻上的压降变化 $1mV$, 在 $0^\circ C$ 度时, 取样电压为 $273.2mV$ 。由于后续 A/D 转换器采用 ICL7135, 其差动输入电压量程为 $0 \sim \pm 1.999V$ 。因此, 取样电压在进行 A/D 转换前还需进行放大。放大器采用通用 741 运算放大器, 设计成同相放大, 放大倍数为 10, 以保证在测温范围内输入的电压在 ICL7135 A/D 转换器允许的输入范围内。图 10-4 中的 R_{1w} 调整传感器零点取样电压, R_w 调整放大器的放大倍数, 从而使得在温度为 $0^\circ C$ 时, 放大器输出电压 U_o 为 $2.732V$, 温度为 $100^\circ C$ 时, 放大器输出电压 U_o 为 $3.732V$ 。

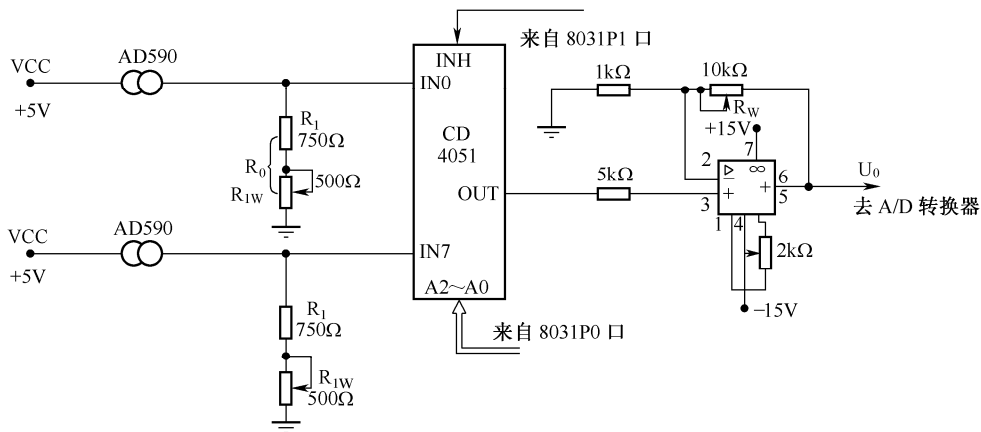


图 10-4 采样放大电路

3. A/D 转换电路

该温度巡回检测系统是采用 $4\frac{1}{2}$ 位双积分式 A/D 转换器 ICL7135 实现模/数转换的。其硬件设计如图 10-5 所示, ICL7135 工作于差动方式, V_{INL} 通过精密稳压管 2CW130 和分压电阻获得 $2.732V$ 的直流电平输入。此举目的之一是为补偿 AD590 在 $0^\circ C$ 时加到 ICL7135 V_{INL} 上的 $2.732V$ 的偏置电压, 使 ICL7135 在 $0^\circ C$ 时, 差动输入电压为零; 目的之二是保证 ICL7135 在测温范围内所接收的差动输入电压在其允许的转换量程范围内。ICL7135 的参考电压 V_{ref} 设定为 $1.000V$, 由稳压管 ICL8069 和分压电阻提供。ICL7135 的转换结果经 74LS244 缓冲器送入单片机 P0 口。74LS244 的输出允许端 1G 和 2G 由来自系统译码器 74LS138 的 Y0 控制。ICL7135 每进行一次 A/D 转换输出 5 位 BCD 码, 即 D5、D4、D3、D2 和 D1。第一次输出的 D5 为最高位, 并且只有一个有效数位 B1 (为 1 或 0)。其他数位溢出信号 OR 送 8031 的 P0.4 端, 欠量程信号 UR 送 8031 的 P0.5, 信号极性 PO1 送 8031 的 P0.6。D5 之后分别输出 D4、D3、D2 和 D1 4 位 BCD 码。8031 的 P1.6 接到 ICL7135 的 $R/\overline{H}$ 端子, 控制 A/D 转换的启动和保持, ICL7135 的输出选通信号 $\overline{STRBOE}$ 接 8031 的外部中断 $\overline{INT0}$, 8031 采用中断方式读取 A/D 转



换结果, ICL7135 的时钟信号由 8031 的 P1.7 提供, P1.7 输出的方波信号由 8031 内部定时器 T0 通过中断定时产生, 并根据 ICL7135 的特性要求, 使其输出频率为 200kHz。

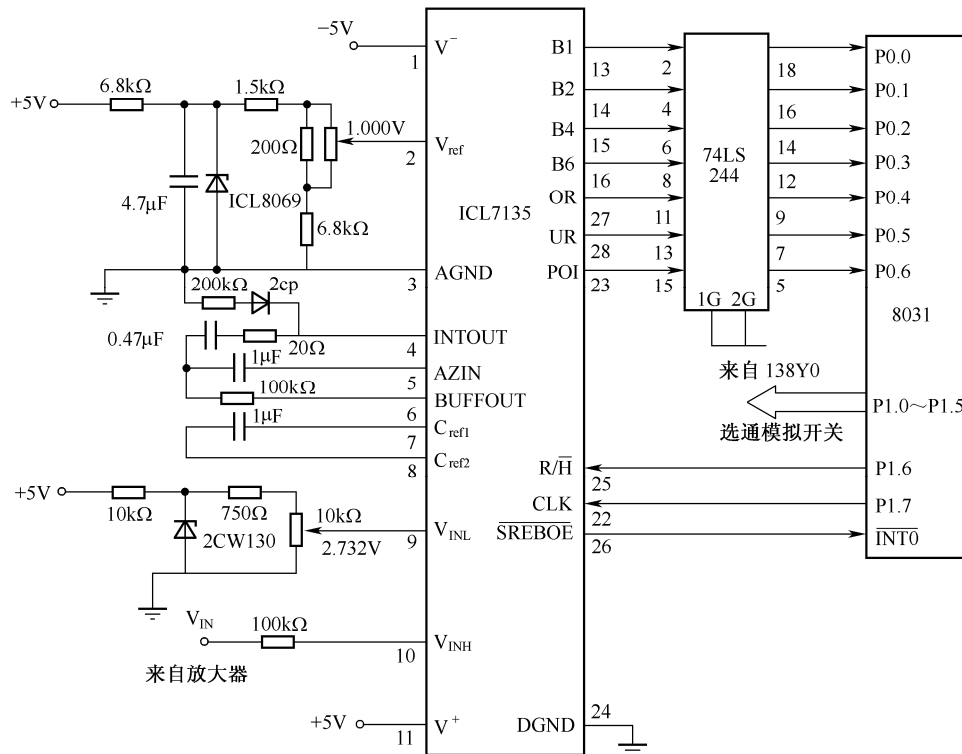


图 10-5 A/D 转换电路

4. 单片机外围电路

该温度巡回监测系统是以 8031 单片机为核心器件, 由 1 片 2764 EPROM、1 片 6116 RAM、1 片 8155 I/O 接口芯片、1 片 74LS138 译码器及一些必要的逻辑器件构成, 其框图如图 10-6 所示。由于检测系统需要对检测的温度数据进行记录保存, 因此, 在单片机的外围电路中设计了 RAM 掉电保护电路, 保证存在 6116 RAM 内的检测结果在断电后不丢失。I/O 接口芯片 8155 用于系统输入/输出接口扩展, 供 LED 显示器和键盘使用。8031 的 P1.0~P1.5 作为位选信号使用, 控制对应的 6 片多路模拟开关 CD4051 的选通。74LS138 译码器的输出 Y7 用于 A/D 转换器的口地址, 去控制 74LS244 缓冲器输出允许端。8031 对采样的温度数据进行处理后, 可送 8155 进行巡回显示或定点显示, 同时, 也可由打印键控制或定时控制, 由打印机随打印输出或定时打印输出。

5. 显示器/键盘接口

检测系统利用 8155 芯片完成系统显示器/键盘的接口任务, 其结构框图如图 10-7 所示。8155 PA 口作为键盘输入口, 系统设置 7 个小键和一个开关。其中, PA0、PA1 分别定义为左移键和加 1 键, 以此 2 键代替 10 个数字键向系统输入有关参数; PA2 定义为上限温度报警器参数设定键; PA3 定义为下限温度报警参数设定键; PA4 定义为定位显示键; PA5 定义为打印键; PA6 定义为停止报警键; PA7 是一个硬开关。PB 口定义为 BCD 码、小数点、符号输出口。其中, PB0~PB3 作为 BCD 码输出位; PB4 作为小数点输出位; PB5 对应符号输出位。当检测温度为负时, 使该位输出低电平, 点亮对应的 1 个发光二极管, 若检测温度为正时, 该位输



出高电平，对应发光二极管不亮。PC 口定义为输出 0，作为 6 位 LED 显示器的位选通输出，6 位 LED 的低 4 位作为温度数值的显示，高 2 位用于通道号数的显示，显示过程采用动态扫描方式。各点温度可以巡回显示，也可由定位显示键控制进行定点显示。

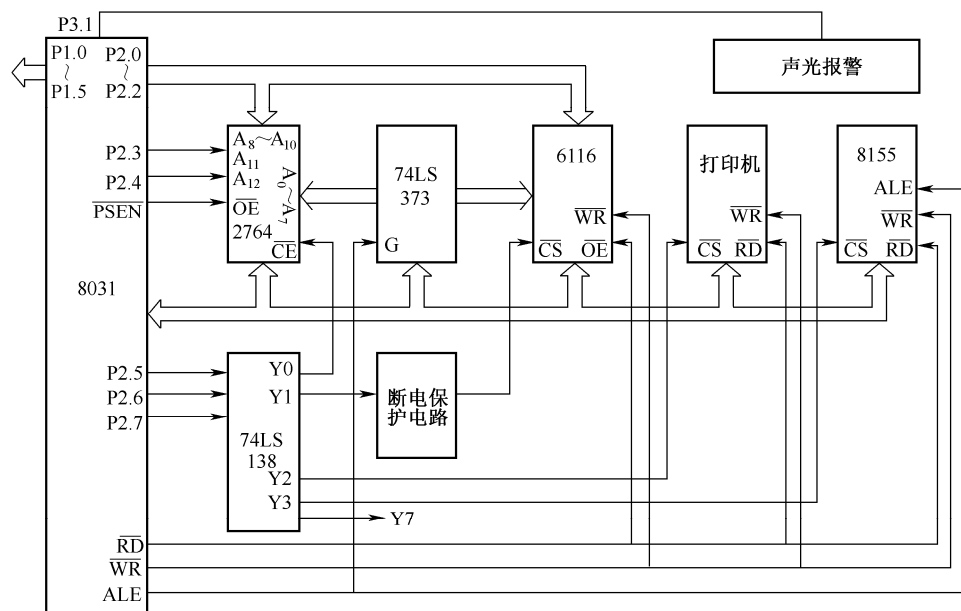


图 10-6 单片机外围电路框图

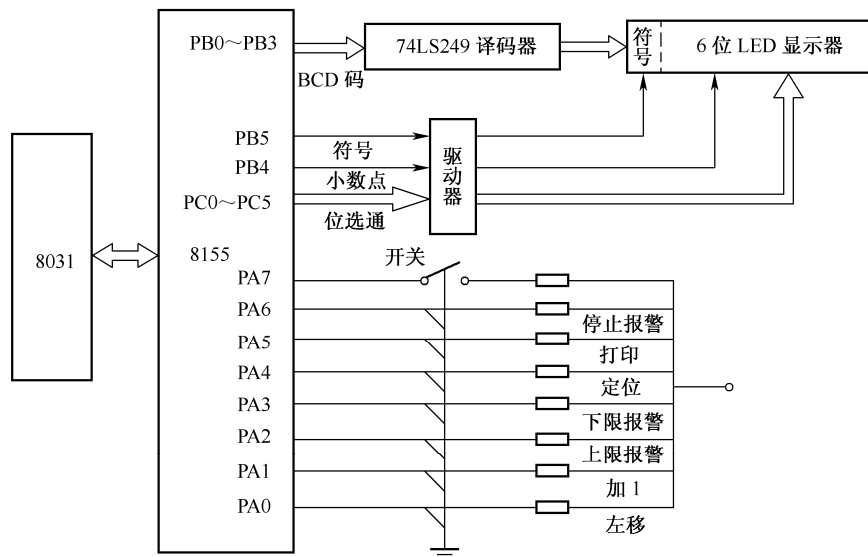


图 10-7 显示器/键盘接口

10.4.2 系统软件设计

系统软件设计采用模块化结构。整个程序由主程序、显示、键盘扫描、A/D 转换等子程序模块组成。



8031 单片机系统中，片内、外 RAM，EPROM 以及 I/O 口存储空间的地址是统一的，地址分配如下。

堆栈栈顶地址设置在片内 RAM 数据缓冲区 60H。

显示缓冲区设在片内 RAM 的 40H~47H 单元。

2764 EPROM 存储区地址为 8000H~9FFFH。

6116 RAM 地址为 3800H~3FFFH。

8155 状态口为 AF00H；RAM AE00H~AEFFH；A 口 AF01H；B 口 AF02H；C 口 AF03H。

ICL7135 口地址为 DFFFH。

系统软件主要由初始化程序、主程序、A/D 转换程序、打印程序及监控程序等组成。初始化程序是对 8031 内部特殊功能寄存器 SFR 及 8155 芯片的 PA 口、PB 口和 PC 口的工作方式进行设定。监控程序完成对键盘输入的扫描及显示器的显示；A/D 转换程序完成对信号的采样和 A/D 转换，主程序对采集的数据进行处理，其中，A/D 转换程序是 8031 响应 ICL7135 A/D 转换器的中断服务程序，打印程序是 8031 响应定时器 T0 的定时中断服务程序，或是响应打印键的键处理程序。下面主要介绍主程序、A/D 转换程序及显示、键盘程序的设计。

1. 主程序

主程序流程图 10-8 所示。

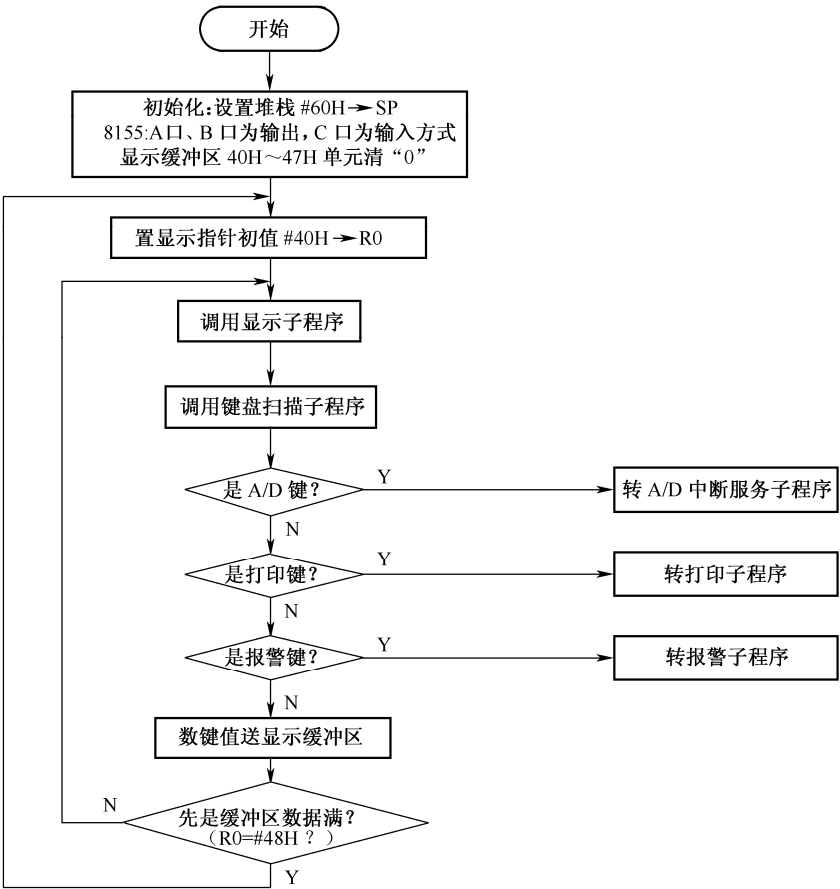


图 10-8 主程序流程



主程序清单如下:

```
                ORG    8000H
MAIN: MOV      SP, #60H           ; 设置堆栈
            MOV    DPTR, #0AF00H ; 8155 A 口、B 口为输出
            MOV    A, #03H       ; C 口为输入方式
            MOV    R0, #40H      ; 显示缓冲区 40H~47H 清零
            MOV    A, #00H
ML0:  MOV     @R0, A
            CJNE   R0, #48H, ML0
ML1:  MOV     R1, #40H
ML2:  ACALL    DIR               ; 调用显示子程序
            ACALL  KEYI          ; 调用键盘扫描子程序
            CJNE   A, #0EH, AD1   ; 如果 A 不等于 0EH 转 AD1
            AJMP   ONE           ; 转到 A/D 工作子程序
AD1:  CJNE   A, #0FH, AD2       ; 如果 A 不等于 0FH 就转 AD2
            AJMP   TWO           ; 如果是打印键, 转打印程序
            AJMP   THREE        ; 转到报警子程序
AD2:  ANL     A, #0FH           ; 数字键, 键值送显示缓冲区
            MOV    @R1, A
            INC    R1            ; 修改显示缓冲区指针
            CJNE   R1, #48H, ML2 ; 显示没完, 转 ML2
            AJMP   ML1
```

2. A/D 转换程序

图 10-9 所示为 A/D 转换程序的框图。

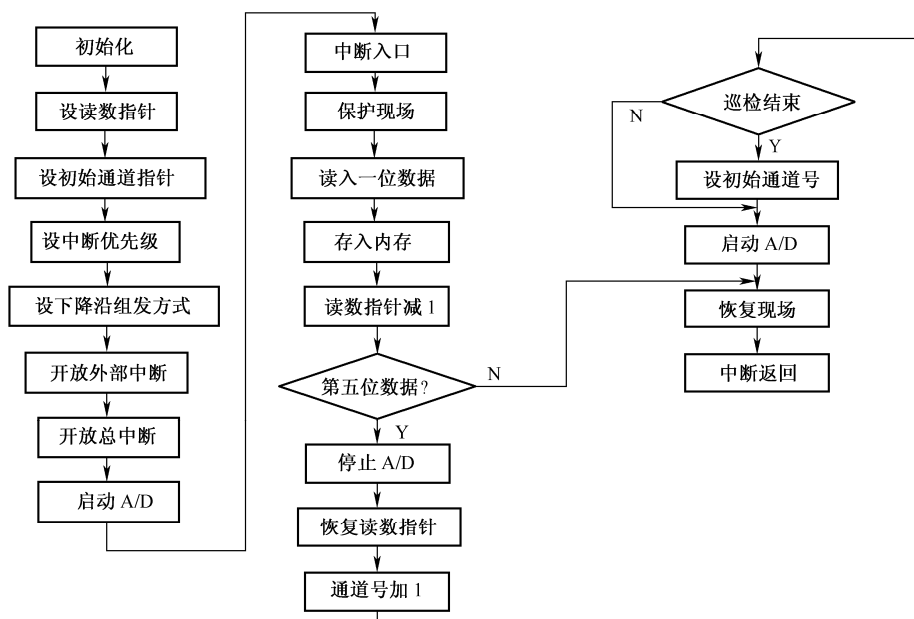


图 10-9 A/D 转换程序框图



A/D 转换程序如下:

```
ONE: MOV    31H, #00H
      MOV    20H, #0FFH
      MOV    21H, #37H
      SETB   IT0
      NOP
      SETB   EA                      ; CPU 开中断
      NOP
      SETB   EX0
      NOP
      MOV    DPTR, #0DFFFH          ; 启动 A/D 转换
      MOV    A, #00H
      MOVBX  @DPTR, A
      MOV    R0, #40H                ; A/D 转换结果送缓冲区
      MOV    A, 30H
      ANL    A, #0FH
      MOV    @R0, A
      INC    R0
      MOV    A, 30H
      ANL    A, #0F0H
      SWAP   A
      MOV    @R0, A
      MOV    42H, #0EEH
      MOV    43H, #0EFH
      MOV    44H, #0DH
      MOV    45H, #0AH
      MOV    46H, #00H                ; 不显示
      MOV    47H, #00H                ; 不显示
      ACALL  DIR                      ; 调用显示子程序
      AJMP   LOOP
```

3. 中断服务程序

中断服务程序如下:

```
PINTI: PUSH  PSW                      ; 保护现场
      PUSH  ACC
      PUSH  DPL
      PUSH  DPH
      INC    31H
```



```
MOV A, 31H
CJNE A, #0FFH, CCT
MOV 31H, #00H
MOV DPTR, #0DFFFFH ; A/D 转换结果送入 RAM
MOVX A, @DPTR
MOV 30H, A
MOV DPL, 20H ; 修改 RAM 地址
MOV DPH, 21H
INC DPTR
MOV A, 30H
MOVX @DPTR, A
MOV 20H, DPL
MOV 21H, DPH
MOV A, 21H
RLC A
RLC A
RLC A
RLC A
CJNE A, 1, CCT ; 如果 A 不等于 1 就转 CCT
CLR EA
INC A
AJMP ONE
CCT: POP DPH ; 出栈返回
POP DPL
POP ACC
POP PSW
```

A/D 转换程序是系统软件的主要组成部分，它是利用 8031 外部中断 INT0 来读取 A/D 转换数据的，程序包括对 8031 中断的初始化、模拟通道的切换、A/D 转换器的启动及数据读取。程序框图如图 10-9 所示，初始化程序使 8031CPU 具有响应 INT0 外部中断的条件，并且选通模拟开关的 000H 通道，再通过对 8031 的 P1.6 置位，启动 ICL7315 开始将 000H 通道信号进行 A/D 转换。当 ICL7315 一次转换结束后，随即通过 STRBOE 发出 5 个负脉冲，分别对应 5 位 BCD 码的输出，8031 INT0 外部中断每接收 1 个 STRBOE，将向 CPU 发出 1 次中断请求，CPU 在响应中断请求时进入终端服务。在终端服务程序中，除了读取相应位的 BCD 码外，还要进行模拟通道切换的调整工作，从而实现巡回检测和响应通道的 A/D 转换。

4. 显示、键盘处理程序

测量系统采用 6 位 LED 动态扫描方式显示所检测的温度，高 2 位显示对应检测点的通道号，低 4 位为该点的温度值。温度值的 BCD 码在主程序进行数据处理时装入显示缓冲单元字



节的低 4 位, 即 D0~D3 位, 而温度的小数点及极性装配在 D4、D5 位。它们都从 8155 的 PB 口输出到 LED, 8155 PC 口的每一位对应 1 个 LED 的位选通, 通过扫描方式分时每次选通 1 个 LED 点亮, 从而实现动态显示。

在设计键盘处理程序时, 首先考虑到 8155 芯片的 PA 口与键盘输入是采用一键一位对应方式设计的, 因此, CPU 查询是否有键动作或判断是哪个键动作都很方便, 其次, 由于仅用左移键和加 1 键代替 10 个数字键, 这就要求 2 键处理程序密切配合, 而且尽可能使操作者在计数时直观方便。键盘扫描程序框图如图 10-10 所示, 左移键处理程序的主要任务是为操作者提供直观的显示和要修改的 LED 的位。当它被按 1 次, 6 位 LED 的某位闪烁, 以表示该位可以进行新的数字设置, 该位数字设定完以后表示该位可以进行新的数字设置。该位数字设定完成后若需设定另一位 LED 的数字, 则可再按左移键, 闪烁位将由低位向高位左移到所要设定的位。通过重复按左移键可实现 7 位循环 (第 7 位是不闪烁的), 加 1 键是在左移键配合下, 使闪烁的 LED 位内容加 1。当左移键使某位 LED 闪烁后, 再按加 2 键, 则该位的显示数加 1, 若重复按加 1 键, LED 内容累计加 1, 当加到 9 后, 再按加 1 键, 则 LED 显示器内容为零。CPU 在使 LED 内容加 1 的同时判断按键标志, 以确定当前所修改数据是属于哪个按键的要求, 并将修改后的数存入对应键的缓冲单元, 达到所要修改参数的目的。PA 口的 PA7 作为硬开关输入位, 当键扫描程序检测该位为低电平时, 内部设置标志, 表示系统将定时打印检测数据, 若检测系统为高电平, 内部设置另一标志, 表示系统由打印键控制随机打印检测数据。键扫描程序检测到报警参数键被按下时, 则设置相应标志, 表示下面要修改的数据是报警参数, 并将修改后的参数存入相应单元。键扫描程序检测到定位显示键按下时, 设置相应标志, 表示下面要修改显示通道号, 并将要显示的通道号存入相应单元, 使 CPU 固定显示所指定的通道温度。当打印键或消除报警键按下后 CPU 将执行相应的动作, 完成相应的功能。系统键扫描是利用系统提供的基准时间进行定时扫描的, 其定时扫描时间设置为 100ms。

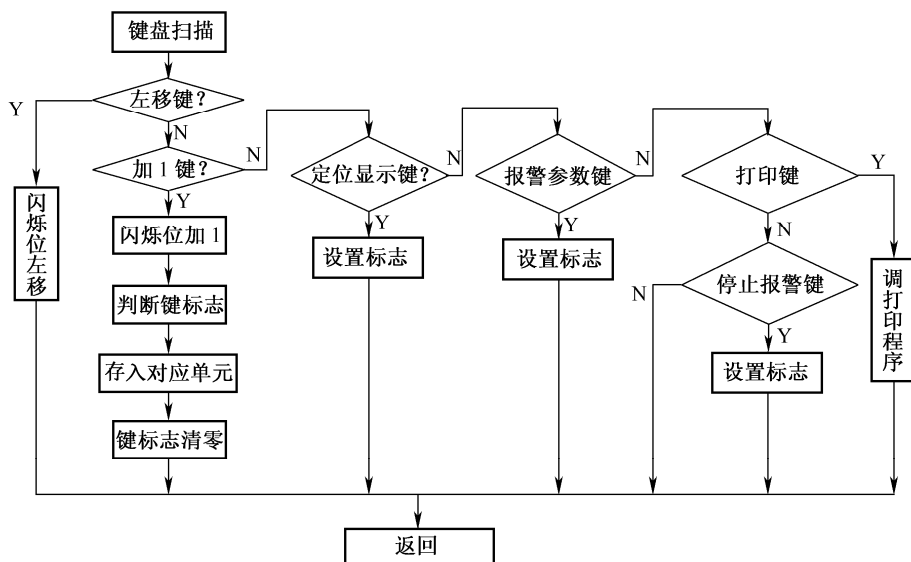


图 10-10 键盘扫描程序框图



本章总结

本章主要让读者对单片机应用系统的开发过程、开发环境、工具有初步了解，然后介绍了温度采集系统实例。单片机应用系统的设计采取软件和硬件相结合的方法，通过对系统的目标、任务、指标要求等要求的分析，确定功能技术指标的软硬件分工方案是设计的第一步；然后分别进行软硬件设计、制作、编程；软件和硬件相结合对系统进行仿真调试、修改、完善是系统设计的关键，也是提高单片机应用水平的重要途径。

习 题

10-1 单片机应用系统设计包括哪些内容？

10-2 单片机应用系统的前向通道包括哪些组成部分？有哪些特点？

10-3 单片机应用系统后向通道有什么特点？

10-4 单片机应用系统硬件设计中应遵循的原则是什么？

10-5 为什么要进行单片机应用系统的调试？如何完成调试工作？

10-6 设计一个步进电机控制器。要求能从键盘上输入步进电机转数，控制步进电机的正、反转及启动。

10-7 设计一个空调的温度控制系统。要求如下：

(1) 能测量并显示实时环境温度；

(2) 当温度高于设定温度，压缩机工作，室温降低；当室温达到设定温度，压缩机停止工作；

(3) 通过键盘输入设定温度。设定温度过程中显示设定温度值，设定完毕后，该位显示当前环境温度。