

# 项目一 交通灯信号指示电路的设计与制作

## ★ 项目描述

在城乡街道的很多路口，特别是十字交叉路口，为了保证行人安全和交通秩序顺畅，一般在每条道路和人行横道上各装有一组红、黄、绿交通信号灯。其中红灯亮，表示该条道路禁止通行；黄灯亮，表示该条道路上未过停止线的车辆禁止通行，已过停止线的车辆继续通行；绿灯亮，表示该条道路允许通行。智能的交通信号灯指挥着各种车辆和行人的安全通行，实现红、黄、绿灯的自动指挥是城乡交通管理现代化的重要课题。

本项目采用常见的发光二极管来模拟城乡街道的交通信号指示灯，利用 51 单片机的输入/输出端口来控制红色、黄色、绿色三种发光二极管的发光状态，模拟实现一个简单的交通指示灯控制电路来自动控制红、黄、绿交通灯的状态转换。



图 1.1 交通信号灯示意图

## ★ 项目分析

### 1. 工作任务

设计一个 51 单片机的最小系统电路,并在此基础上按照项目设计要求构成交通灯信号指示电路,在与本课程配套的实验板上制作出硬件电路并下载相关应用编程软件进行编程调试,使由发光二极管构成的三种指示灯(红色、黄色、绿色)按照要求进行指示。

## 2. 项目任务和要求

(1) 通过学习相关知识,选定合适的 51 单片机型号和其他需要的元器件,设计一个 51 单片机的最小系统电路,在此基础上通过单片机输入/输出端口来控制红、黄、绿三种颜色的发光二极管,构成一个简易十字路口交通灯信号指示电路,分别模拟南北方向和东西方向各个路口的交通灯运行状况。

(2) 实验板的输入电源电压为直流 5 V,红、黄、绿发光二极管均采用  $\phi 5$  mm 的高亮度发光二极管,工作电压 1.8~2.1 V,工作电流 15~25 mA,合理选择其他电路元器件,通过学习相关知识或者查找资料了解所选用的电路元器件的主要性能特点及管脚排列。

(3) 每次绿灯变换为红灯前,要求先亮黄灯 3 s 后,才能变换,以免造成交通安全隐患。

(4) 具有一个启动开关按钮,按下之后进入正常工作模式。

(5) 在正常工作模式下,各条道路的红、黄、绿灯光变换按照表 1.1 要求的 10 种状态进行切换,依次循环。

表 1.1 各个方向的指示灯状态表

方向 状态	东西方向			南北方向			各方向 人行道	时间
	左转	直行	右转	左转	直行	右转		
①	红灯	绿灯	绿灯	红灯	红灯	绿灯	红灯	25 s

②	红灯	黄灯	绿灯	红灯	红灯	绿灯	红灯	3 s
③	绿灯	红灯	绿灯	红灯	红灯	绿灯	红灯	20 s
④	黄灯	红灯	绿灯	红灯	红灯	绿灯	红灯	3 s
⑤	红灯	红灯	绿灯	红灯	绿灯	绿灯	红灯	25 s
⑥	红灯	红灯	绿灯	红灯	黄灯	绿灯	红灯	3 s
⑦	红灯	红灯	绿灯	绿灯	红灯	绿灯	红灯	20 s
⑧	红灯	红灯	黄灯	黄灯	红灯	黄灯	红灯	3 s
⑨	红灯	红灯	红灯	红灯	红灯	红灯	绿灯	15 s
⑩	红灯	红灯	红灯	红灯	红灯	红灯	黄灯	3 s

(6) 设计电路原理图，画出 PCB 版图，看懂项目指导老师给出的装配图。

(7) 在配套的实验板上按照给出的装配图进行电路元件装配焊接，编写软件程序，运用 Keil C51 和 Proteus 软件进行软件程序编译、模拟仿真与调试下载，进行软件、硬件联合调试，并根据测试现象分析故障原因。

(8) 编写相关的技术文档及工艺文档。包括：产品的功能说明，方案选择报告，产品电路原理图及分析，工具、测试仪器仪表、元器件及材料清单，电路板上的电路布局图（参见附录一的实验开发板），电路装配的工艺流程说明，调整测试记录，测试结果分析，现场介绍所需的幻灯演示文稿。

(9) 将实验板上装配的发光二极管模拟城乡街道交通信号灯控制电路上电模拟演示，并现场介绍功能。

## ★ 项目分解与实施

根据以上对项目的分析，依据循序渐进的原则，从能够搭建一个单片机最小系统电路开始，然后到能用 Keil C51 软件对单片机控制发光二极管 (LED) 电路进行软件的编程与调试，再到能用 Proteus 仿真软件进行仿真，最后实现红、黄、绿发光二极管交通信号指示电路的设计与制作。

因此，按照先简单、后复杂的顺序对本项目进行分解，包括以下四个学习任务：

- (1) 单片机最小系统电路的设计与制作。
- (2) 单片机编译软件 Keil C51 的使用训练。
- (3) 仿真软件 Proteus 的使用训练。
- (4) 交通灯信号指示电路的设计与制作。

## 任务一 单片机最小系统的设计与制作

### 【任务要求】

根据 51 单片机的最小系统电路构成，设计与制作出 51 单片机的最小系统电路主板，并运用单片机的相关理论知识在实验板上对单片机最小系统电路进行调试与检测。

具体任务要求如下：

- (1) 选出适合本项目的单片机芯片以及其他相关电子元器件。
- (2) 根据设计要求，设计单片机的电源电路、时钟电路、复位电路以及其他接口电路。
- (3) 能用装配工具焊接、制作出单片机最小系统电路的电路板。
- (4) 能用万用表、示波器等仪器仪表检测电子元器件和相关电路，会在实验板上调试单片

机最小系统电路。

## 【相关知识】

### 一、为什么要学习单片机

当今社会，应用单片机的电子产品已经渗透到日常生活的各个领域，几乎很难找到哪个领域没有单片机的踪迹。现在，单片机的应用领域已十分广泛，比如智能仪表、实时工控、机电一体化、家用电器、通信设备、导航系统等，都离不开单片机的控制。

早期的大部分电子仪器和设备不是成本太高就是电路复杂，工程师设计它们要花很长的时间，维护它们也要花很多的精力，想要更改或者升级它们的功能就更加困难了。而单片机具有强大的功能和扩展性，完成这些工作得心应手。单片机只用一片集成电路，就可以进行简单的逻辑运算和控制。各种产品一旦用上了单片机，就能使产品升级换代，这些产品往往在名称前冠以“智能型”，如智能型洗衣机等，单片机在整个产品装置中起着核心的作用。

目前，应用单片机技术开发智能型产品已经成为电子设计的一种潮流，因为采用单片机技术设计的电路简单、性价比高、功能强大。单片机的广泛应用也是电子产品向智能化方向发展的必然趋势。

### 二、如何学习单片机

使用单片机需要理解单片机的硬件结构以及内部资源的使用方法，在熟悉编程语言的基础上学会单片机各种接口电路的编程技巧，从而实现对硬件电路各种功能的程序控制。那么，怎样才能学好单片机呢？下面介绍几点技巧。

#### (一) 理论是基础

要掌握理论知识，首先要了解单片机的基本原理以及其寄存器、外部中断、定时器、串口等的功能和特点。建议初学者将 51 单片机作为入门级芯片来学习，因为：

(1) 51 单片机内部结构简单，非常适合初学者学习，而且它的技术资料比较齐全，用户也较多，市场占有率也很大。由于其拥有大量的用户和技术资料，不论是入门还是熟练使用它都相对比较容易，能使初学者很快收到事半功倍的学习效果，并且在学习了 51 内核的单片机之后，再学习使用其他内核的单片机就变得相对简单了。

(2) 51 内核单片机（指具有 MCS-51 CPU 的单片机）是 20 世纪 70 年代由 Intel 公司研发并于 80 年代被广泛应用的单片机，当时 Intel 公司开放了 51 内核的授权，至今众多的授权供应商开发了超过万种的 51 内核或兼容 51 内核的单片机，而且 51 内核相当稳定，指令集比较合理，性价比高，适应性强，这使得大多数研发人员都愿意选用它，市场普及率较高。

## (二) 重视动手实践

单片机这门课是非常重视动手实践的，在掌握理论知识的同时，要积极去动手实践。单片机属于硬件，只有把硬件摆在面前，亲自操作它，才会有深刻的体会，才能掌握它。要把更多的时间放到动手实践中去，在实践过程中有不懂之处再查书，这样记忆才深刻。单片机技术与其说是学出来的，还不如说是做实验练出来的，做实验本身就是一种学习过程。边学边练的学习方法效果特别好。

比如，在单片机开发板上先实现最简单的功能，如做一个流水灯、数码管显示等；实现简单的功能后，再逐步尝试完成稍微复杂的程序，例如矩阵键盘、中断控制、串口通信、液晶显示等；在上述功能都可以实现之后，如果对硬件熟悉，可以自己尝试制作一个多功能的单片机开发板；最后再动手做出一个合格的产品，这是检验自己单片机技能的唯一标准。在这个过程

中，遇到不懂的问题要先思考，自己查找硬件和软件上的原因，实在找不到解决办法时，可以请教老师和项目组其他同学，也可以通过在网上众多的单片机学习论坛中和其他人交流学习来获得经验。

### (三) 选择合适的编程语言

简单地说，使用单片机实际上就是编写软件去控制单片机的各个功能寄存器以及单片机的相应引脚，使其按照时序要求输出高、低变化的电平，并由这些高、低变化的电平来控制外围接口电路，以实现人们需要的各种功能。

单片机编程用 C 语言或汇编语言都可以，建议用 C 语言比较好。如果同学们有 C 语言的基础，则学起来会更快；如果没有 C 语言基础，也可以边学单片机边学 C 语言。C 语言的可读性好，移植容易，易学易用，是普遍使用的一种计算机语言。汇编语言写程序代码效率高，但相对难度较大，而且很繁琐，尤其是遇到算法方面的问题时编程非常麻烦。现在单片机的主频在不断提高，人们完全不需要那么高效率的代码，同时单片机的 ROM 空间也在不断提高，足够装下用 C 语言编写的复杂代码，C 语言的资料又多又好找，可移植性非常好，所以建议初学者使用 C 语言进行单片机编程。

## 三、单片机以及 51 单片机概述

### (一) 什么是单片机

你应该熟悉并经常使用个人电脑吧？个人电脑的另一个名字是微型计算机，微型计算机系统一般包括中央处理单元（CPU）、存储器（Memory）以及输入/输出设备（I/O）三大部分。CPU 控制整个系统的运行；存储器存放运行的程序和数据；输入/输出设备是微型计算机与外部接口沟通和联系的通道。

单片机实际上就相当于一个微型计算机，它的中央处理单元、存储器、输入/输出设备等全部集成在一个芯片里，外面再加上几个电阻器、电容器、晶体管等，构成一个完整的微型计算机系统。图 1.2 所示就是单片机的外形及内部的结构示意图。

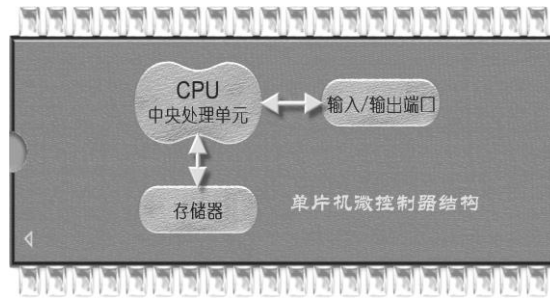


图 1.2 单片机内部的外形及结构示意图

所以说，单片机实际上就是一块集成芯片，但这块集成芯片具有强大的特殊功能，可以通过将编写的软件代码（编程）写进它的存储器来控制其输入/输出端口，进而控制与单片机各个引脚相连的外围电路的电气状态，从而达到控制外围电路功能的目的。

## （二）51 单片机的标号信息与封装

MCS-51 单片机是指由美国 Intel 公司生产的一系列单片机的总称。Intel 公司的 MCS-51 系列单片机是 8 位的，这一系列单片机包括了很多品种，如 8051、8052、8951、8952 等。其中 8051 是 MCS-51 系列单片机中最早、最典型的产品，该系列其他单片机都是在 8051 的基础上进行功能增加、改变而来的，所以人们习惯用 8051 来称呼 MCS-51 系列单片机。

Intel 公司将 MCS-51 的核心技术授权给了很多其他公司，所以目前有很多公司在生产以 8051 为内核的单片机，当然功能或多或少有些改变，以满足不同的需求。比如宏晶科技公司开发生产的 STC89C51 系列芯片是这几年在我国比较流行的 8 位单片机，本项目将用 STC89C51RC（或者 STC89C52RC）来完成一系列项目的设计和制作。



有很多芯片生产厂商都在生产 51 单片机，这些主要生产厂商及其生产的 51 单片机芯片如表 1.2 所示。

表 1.2 51 单片机的主要生产厂商及产品

公 司	产 品
STC	STC89C51RC, STC89C52RC, STC89C53RC, STC89LE51RC, STC89LE52RC, STC12C5412AD 等
Atmel	AT89C51, AT89C52, AT89C53, AT89C55, AT89LV52, AT89S51, AT89S52 等
Philips	P80C54, P80C58, P87C54, P87C58, P87C524, P87C528 等
Winbond	W78C54, W78C58, W78E54, W78E58 等
Intel	i87C54, i87C58, i87L54, i87L58, i87C51FB, i87C51FC 等
Siemens	C501-1R, C501-1E, C513A-H, C503-1R, C504-2R 等
Silicon Labs	C8051F020, C8051F040, C8051F310, C8051F320, C8051F340, C8051F410 等

在 51 单片机的芯片封装上都印有关于芯片的标号信息。下面以 STC 单片机芯片为例介绍其芯片标号信息，其他类型的单片机大同小异。比如，STC 单片机芯片上的全部标号为：STC 89C52RC 40I-PDIP 1308H4W377.90C。其标识意义如下：

STC——前缀，表示芯片为 STC 公司生产的产品。其他前缀还有 AT、i、W、SST 等。

8——表示该芯片为 8051 内核芯片。

9——表示内部含 Flash E<sup>2</sup>PROM 存储器。还有如 80C51 中的 0 表示内部含 Mask ROM (掩膜 ROM) 存储器；87C51 中的 7 表示内部含 EPROM 存储器 (紫外线可擦除 ROM)。

C——表示该器件为 CMOS 产品。还有如 89LV52 和 89LE58 中的 LV 和 LE 都表示该芯片为低电压产品 (通常为 3.3 V 电压供电)；而 89S52 中的 S 表示该芯片含有可串行下载功能的 Flash 存储器，即具有 ISP 可在线编程功能。

5——固定不变。

1——表示该芯片内部程序存储空间的大小，1 为 4 KB，2 为 8 KB，3 为 12 KB，即该数乘

上 4 KB 就是该芯片内部程序存储空间的大小。程序存储空间的大小决定了一个芯片能装入多少执行代码。

RC——STC 单片机内部的 RAM ( 随机存取存储器 ) 为 512 B。

40——表示芯片外部晶振最高可接入 40 MHz。对于 AT 单片机其数值一般为 24，表示其外部晶振最高为 24 MHz。

I——产品级别，表示芯片的使用温度范围。I 表示工业级，温度范围为  $-40 \sim +85^{\circ}\text{C}$ ；C 表示商业级，温度范围为  $0 \sim +70^{\circ}\text{C}$ ；M 表示军品级，温度范围为  $-55 \sim +150^{\circ}\text{C}$ 。另外还有一些器件为汽车工业级 A，温度范围为  $-40 \sim +125^{\circ}\text{C}$ 。

PDIP——产品封装型号。PDIP 表示双列直插式。

1308——表示本批芯片生产日期为 2013 年第 8 周。

H4W377.90C——表示芯片制造与处理工艺。

89C51 单片机的器件封装方式有 PDIP40 封装、QFP 封装、PLCC 封装三种，而 89S51 的器件封装方式除了这三种外，还有 PDIP42 器件封装。下面对封装方式进行说明。

89C51/89S51 的常见封装为 40 个引脚双列并排直插的 PDIP40，如图 1.3 所示。这种封装与 Intel 公司的 MCS-51 封装完全兼容，PDIP40 刚好可插在面包板或标准 40pin 的底座上。PDIP40 与 PDIP42 除引脚数量不同外，尺寸差异也很大。

除此之外，还有 QFP 和 PLCC 封装的单片机，如图 1.4 所示。

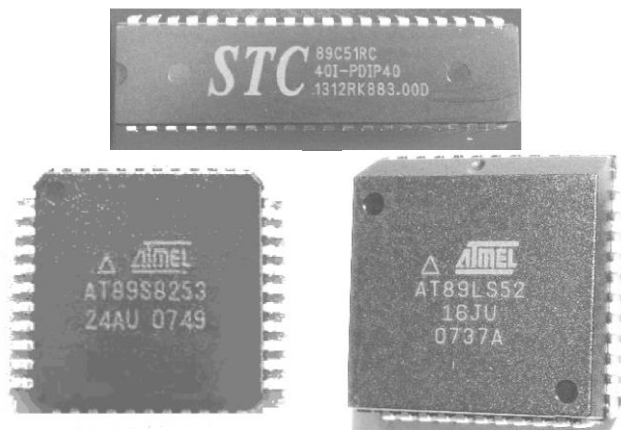


图 1.3 PDIP40 封装的单片机

(T2)P1.0	1	40	VCC
(T2 EX)P1.1	2	39	P0.0(AD0)
P1.2	3	38	P0.1(AD1)
P1.3	4	37	P0.2(AD2)
P1.4	5	36	P0.3(AD3)
(MOSI)P1.5	6	35	P0.4(AD4)
(MISO)P1.6	7	34	P0.5(AD5)
(SCK)P1.7	8	33	P0.6(AD6)
RST	9	32	P0.7(AD7)
(RXD)P3.0	10	31	E $\bar{A}$ /VPP
(TXD)P3.1	11	30	ALE/PROG
(INT0)P3.2	12	29	PSEN
(INT1)P3.3	13	28	P2.7(A15)
(T0)P3.4	14	27	P2.6(A14)
(T1)P3.5	15	26	P2.5(A13)
(WR)P3.6	16	25	P2.4(A12)
(RD)P3.7	17	24	P2.3(A11)
XTAL2	18	23	P2.2(A10)
XTAL1	19	22	P2.1(A9)
GND	20	21	P2.0(A8)

图 1.5 PDIP40 封装的单片机引脚排列

图 1.4 QFP 和 PLCC 封装的单片机

对于 PDIP40 封装的单片机，它的管脚顺序为：我们正对芯片有标号信息的一面，弧形凹口朝上，左上方有个圆点记号的引脚为第 1 脚，按照逆时针排序，分别为第 2, 3, ..., 40 脚。相邻两个引脚间距为 2.54 mm ( 100 mil )，器件长度为 52.578 mm，而两排引脚的间距为 15.875 mm，器件厚度为 4.826 mm ( 不含引脚)。它特别适合学校培训和行业研发使用。不过，由于针脚式封装体积较大，电路板制作成本较高，目前在功能复杂的产品中使用受到限制。

除了 PDIP42 封装的单片机外，89S51 与 89C51 完全兼容，本书将以 PDIP40 封装的 STC89C51RC 为探讨对象来学习 51 单片机。

PDIP40 封装的 STC89C51RC ( 以下简称 8951 ) 有 40 个引脚，要学习 51 单片机，建议大家记住这 40 个引脚的名称和各自的功能，如图 1.5 所示。以下简要说明。

#### 1. 电源引脚 ( 第 20 脚、第 40 脚 )

几乎所有的集成电路芯片都需要连接电源，而 8951 的电源引脚与大部分数字 IC 的电源引脚类似，右上角接电源 VCC、左下角接地 GND。所以 8951 的第 40 脚为 VCC 引脚，连接 (  $5 \pm 10\%$  )

V 直流电源；第 20 脚为 GND 引脚，必须接地。

### 2. 复位引脚 ( 第 9 脚 )

为了防止微控制器在运行过程中死机或者程序跑飞,几乎所有微控制器都需要复位( Reset )的操作。对于 8951 单片机而言,只要复位引脚接高电平超过两个机器周期(约  $2\mu\text{s}$ ),即可产生复位的操作。而 DIP40 封装的 8951 单片机的复位引脚为第 9 脚。

### 3. 时钟引脚 ( 第 18 脚、第 19 脚 )

微控制器都需要时钟脉冲,对于 8951 单片机而言,在接地引脚上方的两个引脚,即第 19、18 引脚就是时钟脉冲引脚,分别是 XTAL1、XTAL2。

### 4. 程序存储器选择引脚 ( 第 31 脚 )

8951 内部有程序存储器,外部也可接程序存储器。使用内部程序存储器还是外部程序存储器,则需要根据第 31 脚的接法决定。第 31 脚就是  $\overline{\text{EA}}$  引脚,当  $\overline{\text{EA}}=1$  时,系统使用内部程序存储器;当  $\overline{\text{EA}}=0$  时,系统使用外部程序存储器。由于大多数情况下编程人员编写的程序代码不是非常复杂,只需要使用内部程序存储器就够用了,所以通常把第 31 脚直接接 VCC。

### 5. 外部存储器控制引脚 ( 第 29 脚、第 30 脚 )

$\overline{\text{EA}}$  引脚下面的两个引脚(第 29、30 脚)是外部存储器控制引脚,这两个引脚与  $\overline{\text{EA}}$  引脚有点类似,是针对外部存储器的控制引脚。相对于其他引脚,第 29 脚和第 30 脚比较难以说明。不过只要不使用外部存储器,就可以当它们不存在(不接)。

第 29 引脚为  $\overline{\text{PSEN}}$  引脚,即外部程序存储器读选通信号输出端,在从外部程序存储器取指令(或数据)期间,每个机器周期内两次有效。

第 30 引脚 ALE,在系统扩展时用于控制把 P0 口的输出低 8 位地址送入锁存器锁存起来,

以实现低位地址和数据的隔离。ALE 引脚以振荡器频率的 1/6 频率，周期性地发出正脉冲信号。因此，它可用于定时的目的，用作对外输出时钟，也可用于检测单片机的工作是否正常。

#### 6. 32 个输入/输出引脚 ( P0.0 ~ P0.7、P1.0 ~ P1.7、P2.0 ~ P2.7、P3.0 ~ P3.7 )

( 1 ) P0 口 ( P0.0 ~ P0.7 分别对应第 39 引脚 ~ 第 32 引脚 )，是一个 8 位漏极开路型双向 I/O 口，P0 口能以吸收电流的方式驱动 8 个 LS 型 TTL 负载。当 P0 口用作输出功能时，每个管脚外部需要接 10 k $\Omega$  的上拉电阻。在访问外部存储器时，它是分时传送的低字节地址和数据总线。

( 2 ) P1 口 ( P1.0 ~ P1.7 分别对应第 1 引脚 ~ 第 8 引脚 )，是一个带有内部上拉电阻的 8 位准双向 I/O 口，每个引脚能驱动 ( 吸收或输出电流 ) 4 个 LS 型 TTL 负载。

( 3 ) P2 口 ( P2.0 ~ P2.7 分别对应第 21 引脚 ~ 第 28 引脚 )，是一个带有内部上拉电阻的 8 位准双向 I/O 口，每个引脚可以驱动 ( 吸收或输出电流 ) 4 个 LS 型 TTL 负载。在访问外部存储器时，它输出高 8 位地址。

( 4 ) P3 口 ( P3.0 ~ P3.7 分别对应第 10 引脚 ~ 第 17 引脚 )，是一个带有内部上拉电阻的 8 位准双向 I/O 口，每个引脚能驱动 ( 吸收或输出电流 ) 4 个 LS 型 TTL 负载。

## 四、单片机的最小系统电路

所谓单片机的“最小系统电路”，是指单片机电路工作不可或缺的基本电路。所以，掌握单片机的最小系统电路对学习使用单片机是十分必要的。单片机的最小系统电路也称为单片机基本电路，包括四个部分，即电源电路、时钟电路、复位电路及存储器设置电路。

### ( 一 ) 外接电源

8951 单片机电路正常工作首先需要外接电源，将第 40 脚接电源 VCC，也就是 +5 V 电源，第 20 脚接地 GND。

## (二) 时钟电路

8951 内部已具备振荡电路，只要在第 20 引脚 (GND) 上方的两个引脚 (即第 18、19 脚) 上连接简单的石英晶体振荡器和大小合适的电容 (22~38 pF) 即可。目前的 MCS-51 单片机芯片的工作频率已大幅提高，例如 Atmel 公司的 89C51 的工作频率为 0~24 MHz，华邦电子 (Winbond) 的单片机的的工作频率为 0~40 MHz，而 STC89C51RC 的工作频率可达 48 MHz。

第 18、19 引脚在本书多数案例中连接 12 MHz 晶振，在串口通信案例中本书选择的是 11.059 2 MHz 晶振。和晶振相连的两个电容为负载电容，用于晶振的启动。由于单片机的晶振工作于并联谐振状态，可以把这两个电容理解

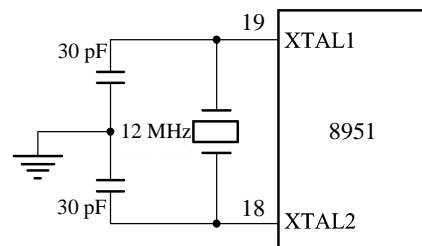


图 1.6 时钟电路

为谐振电容的一部分。两个电容的取值范围为 22~38 pF。两个电容的取值应相同，否则容易造成停振或者不起振。

时钟脉冲采用 12 MHz 晶振时的时钟电路如图 1.6 所示。

## (三) 复位电路

8951 的复位引脚 (Reset, 即 RST) 是第 9 脚，当此引脚连接高电平超过 2 个机器周期 (1 个机器周期包含 12 个时钟脉冲)，即可产生复位的操作。以 12 MHz 的时钟脉冲为例，每个时钟脉冲为  $1/12 \mu\text{s}$ ，2 个机器周期为  $2 \mu\text{s}$ 。因此，我们在第 9 脚上连接一个可让该引脚产生一个  $2 \mu\text{s}$  以上高电平脉冲的复位电路，即可产生复位的操作。复位电路如图 1.7 所示。

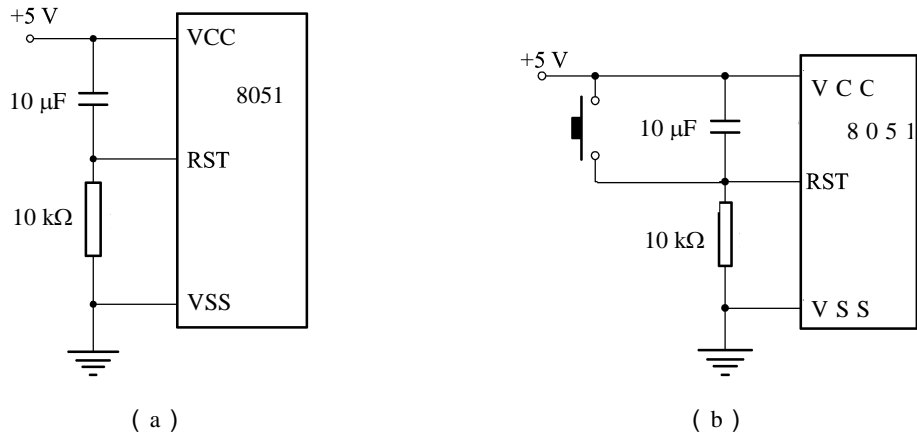


图 1.7 复位电路

单片机加上电源的瞬间，电容上没有电荷，相当于短路，所以第 9 脚直接连接到 VCC，此时 8951 执行复位操作。随着时间的增加，电容上的电压逐渐增加，而第 9 脚上的电压逐渐下降，当第 9 脚上的电压降至低电平时，8951 即恢复到正常工作状态。

(四) 存储器设置

基本电路的最后一个部分是存储器的设置电路。如果把第 31 脚 ( $\overline{EA}$ ) 接 VCC，则采用内部程序存储器；如果把第 31 脚 ( $\overline{EA}$ ) 接地，则采用外部程序存储器。在本书项目里全部采用内部程序存储器，所以把第 31 脚与 VCC 相连接。

单片机最小系统的整个基本电路如图 1.8 所示。

单片机最小系统采用的元器件及其规格如表 1.3 所示。

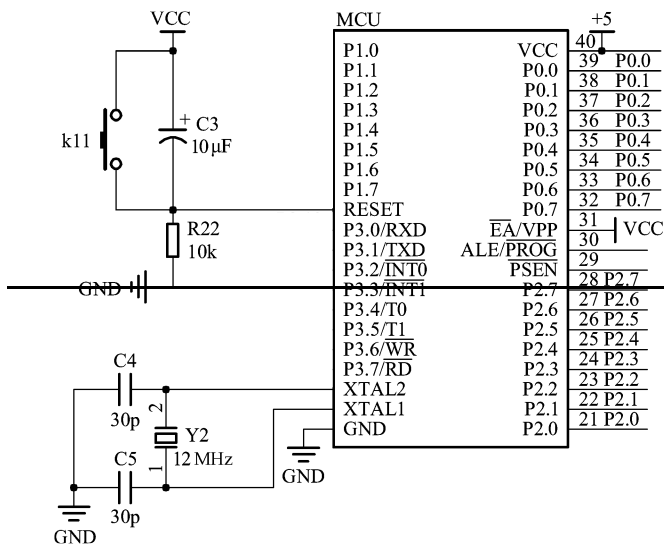


表 1.3 8951 单片机最小系统  
电路的元件清单

名称	规格	数量
STC89C51RC	PDIP40 封装	1
石英振荡晶体	12 MHz	1

图 1.8 单片机最小系统电路

陶瓷电容器	30 pF	2
电容器	10 $\mu$ F/25 V	1
电阻器	10 k $\Omega$	1
按钮开关	a 接点	1

## 五、单片机的内部结构

### (一) 51 单片机的内部结构

51 单片机发展至今，虽然有许多厂商各自开发不同的兼容芯片，但其基本结构并没有多大的变动，标准的 51 单片机的内部结构如图 1.9 所示。其中：

- (1) CPU 为 8 位微控制器。
- (2) 程序存储器 ROM：内部有 4 KB、外部最多可扩展至 64 KB。
- (3) 数据存储器 RAM：内部有 128 B、外部最多可扩展至 64 KB。
- (4) 4 组可位寻址的 8 位输入/输出端口，即 P0、P1、P2 及 P3。
- (5) 1 个全双工串行口，即 UART。
- (6) 两个 16 位定时器/计数器，即定时器 0/计数器 0、定时器 1/计数器 1。
- (7) 5 个中断源，即 INT0、T0、INT1、T1、TXD 和 RXD。



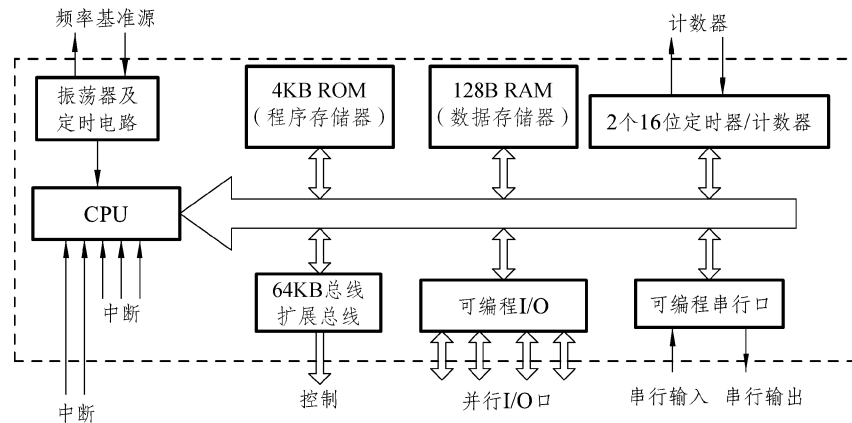


图 1.9 51 单片机的内部基本结构

## (二) 51 单片机的存储器

单片机内部有各种存储器，每个存储器就像一个小抽屉，其中有 8 个小格子，每个小格子就是用来存放“电荷”的，电荷通过与它相连的导线传进来或释放掉，那电荷在小格子里是怎样存储的呢？如果你把导线想象成水管，小格子里的电荷就像是水，就好理解了。存储器中的每个小抽屉就是一个存放数据的地方，我们称之为一个“字节”或者“单元”。

有了存储器就可以开始存放数据了。单片机内部全部采用二进制数据，比如想要给存储器存入一个 2 位的十进制数据“12”，也就是 8 位的二进制数据“00001100”，只要把第二号和第三号小格子里存满电荷，而其他小格子里的电荷给放掉就行了。

存储器按功能可以分为只读存储器和随机存取存储器两大类。所谓只读存储器，从字面上理解就是只能读里面的数据，不能写进去新的数据，它类似于我们的书本，发给我们之后只能读里面的内容，不能随意更改书本上印刷的内容。只读存储器的英文缩写为 ROM (Read Only Memory)。所谓随机存取存储器，即随时能改写，也能读出里面的数据，它类似于我们的黑板，能随时写东西上去，也能用黑板擦擦掉重写。随机存储器的英文缩写为 RAM (Random Access Memory)。这两种存储器的英文缩写一定要记牢。

所谓只读和随机存取，都是针对正常工作情况下而言的，也就是指在使用这块存储器的时候，而不是指对这块芯片烧写程序的时候。不然，只读存储器中的数据是怎么有的呢？其实这个道理也很好理解，书本拿到我们手里是不能改了，但当它还是白纸的时候，当然能由印刷厂印刷文字上去了。

除了无 ROM 型的 8031 和 8032 外，51 单片机的存储器包括程序存储器 (ROM) 和数据存储器 (RAM) 两部分，这两部分在单片机内部是独立的。标准的 8x51 系列具有 4 KB 程序存储器、128 B 数据存储器，而标准的 8x52 系列具有 8 KB 程序存储器、256 B 数据存储器，刚好是 8x51 系列的两倍，不管是 8x51、8031、8x52 还是 8032，其外部扩展的程序存储器或数据存储器最多为 64 KB。51 单片机的存储器结构如图 1.10 所示。

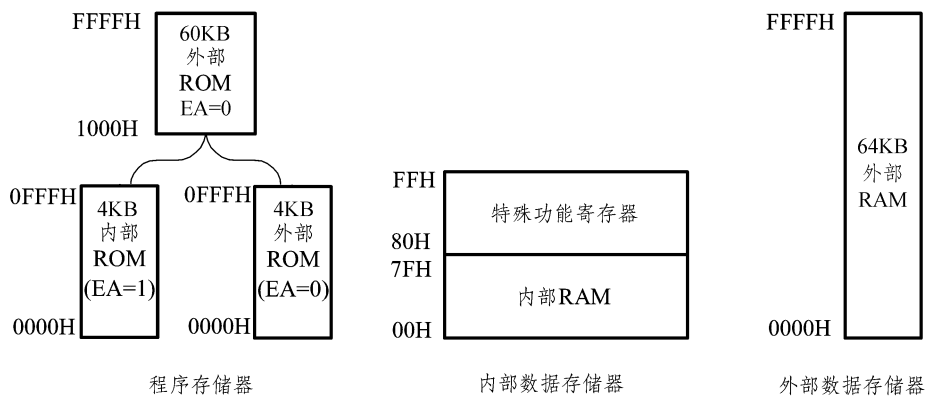


图 1.10 51 单片机的存储器结构

### 1. 程序存储器 (ROM)

一旦用编程器把程序指令写进单片机内部，单片机就可以执行这些指令。如果这些指令保存在单片机内部的只读存储器 ROM (Read Only Memory) 内，则单片机掉电后这些程序指令依然不会丢失。

8951 中的 ROM 是一种电可擦除的 ROM，称为 FLASH ROM，可用编程器在特殊条件下由