

河海大学物联网工程学院（常州）

课程设计报告

题 目 八音盒

专业、学号

授课班号

学生姓名

指导教师

完成时间

目 录

摘要	3
第一章、绪论	4
1.1 课程设计背景.....	4
1.2 课程设计目的.....	4
1.3 课程设计任务.....	4
1.3.1 基本部分要求.....	4
1.3.2 提高部分要求.....	4
1.4 课程设计意义.....	5
第二章、系统设计	5
2.1 系统设计原理.....	5
2.1.1 声音的产生	5
2.1.2 计数初值的计算.....	5
2.1.3 音调.....	6
2.1.4 节拍.....	7
2.2 系统硬件电路设计	8
2.2.1 单片机最小系统设计	8
2.2.2 键盘接口电路.....	9
2.2.3 数码管显示电路	10
2.2.4 拨码开关电路.....	11
2.3 系统软件设计.....	12
2.3.1 系统软件方框图	12
2.3.2 单片机初始化.....	12
2.3.3 音乐播放函数.....	12
2.3.4 模式切换	14
第三章、实验结果	16
3.1 调试过程.....	16
3.2 调试结果.....	16
3.3 创新点	17
第四章、总结	17
4.1 收与心得	17
4.2 改善建议.....	18

参考文献	19
附录一：程序代码	20

摘要：此次课程实践利用STC90C516RD单片机设计了一个八音盒，实现目标为：按下按键可以演奏预先设定的歌曲旋律，并可以切换到琴键模式发出诸如do、re、mi等音调的频率。通过这次设计，我们可以学习MCS-51系列单片机定时器、外部中断、按键扫描、数码管显示以及蜂鸣器控制等的程序设计，从而加深对课程内容的理解。

根据课设基本要求，我们设定的此八音盒的基本功能为：

- 1、使用一位数码管显示当前正在播放的歌曲编号，编号为从1到8。
- 2、通过8个独立按键K20-K27来选择播放的曲目，同时可以实现在不同歌曲之间任意切换的功能，并设定K20用来播放《茉莉花》。
- 3、根据实验提高部分的要求，K20-K27同时可以用来弹奏出do、re、mi等音调的频率，以实现琴键的功能。

关键词： MCS-51 定时器 外部中断 按键扫描 数码管显示

Abstract

This design uses the STC90C516RD+ MCU design a music box and pressing the button can play the pre-set song melody. Through this design, we can learn MCS-51 series microcontroller timer and external interruption, key scanning, digital tube display, version control and program design, so as to deepen the understanding of the content of the course.

Its Basic functions: 1.using one digital tube displays the currently playing song number, from one to eight.

2、Using eight independent press including K20-K27to select the playlist, especially the button K20 was used to play the song 《Arabian jasmine flower》.

3.Based on the requirements,keys can be used to play music itself. By repeatedly commissioning, the features are accurate in achieving both completed the course design tasks,also deepened understanding of single-chip microcomputer principle curriculum, achieve the intended purpose and effect.

Key words: external interrupts 、 MCS-51、 timer 、 key scan 、 digital tube display

第一章、绪论

1.1 课程设计背景

随着电子集成技术的发展，单片机在生活中的应用也越来越广泛。它的出现和发展极大地丰富了市场上商品的种类，给我们的生活带来了方便、快捷和乐趣。同时《单片机》作为一门技术性、应用性很强的学科，在我们的专业课中也占有举足轻重的地位。近年来随着课程改革的进一步实施，课程实践的地位也显得愈加重要。对于我们学生来说，实践给我们提供了一个很好的理论联系实践的平台，使我们能够在课程实践这个不断改进、不断完善的过程中，进一步夯实理论知识，同时使能力得到提升。而且到目前为止，我们在课堂上已经学习了两种编程语言，C语言和汇编语言。可根据自己对于这两种语言的掌握程度任意选取一种进行相关编程。

1.2 课程设计目的

为了使同学们将课堂上学到的理论知识与实际应用结合起来，通过在课程实践中完成一小系统的设计，不但能加深对电子电路、电子元器件等方面的认识，同时在相关仪器设备的使用技能等方面也会得到较全面的锻炼和提高。而且有助于同学们在实践中提高自己分析问题、解决问题的能力，为今后能够独立进行某些单片机应用系统的开发设计工作打下一定的基础。

1.3 课程设计任务

1.3.1 基本部分要求

利用蜂鸣器和独立按键K20-K27制作一个八音盒，内存八首乐曲，当按下不同的按键时，会奏出不同的乐曲。其中《茉莉花》是必选曲目，其他任意。例如：按下K20奏出《茉莉花》，（提示：利用定时器产生不同的音调需要查do，re，mi等音调的频率）

1.3.2 提高部分要求

把其他按键当作琴键，可以弹奏乐曲。

1.4 课程设计意义

通过自己动手设计单片及其组成的器件，在软件与硬件上进一步了解单片机的原理，深入了解单片机 C51 语言的实际应用，能帮助同学们更深入地理解课堂上所学的内容，巩固 C 语言程序编写，体会作为一名程序设计人员所需要具备的各种能力。同时通过一个实践的完成帮助同学们更好的树立自信心，灵活运用理论知识联系实际，提高发现问题、分析问题、解决问题和实践动手的能力。督促同学们在课余实践中勇于尝试，把自己真正培养成一个创新型人才。

第二章、系统设计

2.1 系统设计原理

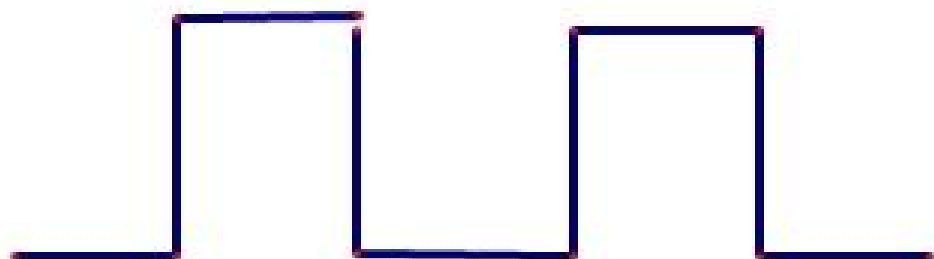
2.1.1 声音的产生

声音是一种音频振动的效果，若振动的频率高，则为高音，反之频率低，则为低音。音频的范围为 20Hz 到 200KHz 之间，人类耳朵比较容易辨认的声音范围为 0-20KHz。一般音响电路是以正弦波信号驱动喇叭，即可产生悦耳的音乐；在数字电路里，则是以脉冲信号驱动蜂鸣器，以产生声音。

在单片机中要发出不同频率的声音，只要产生不同频率的脉冲，再通过喇叭等播放出来即可。又由于方波容易用定时器产生，故一般单片机使用方波脉冲。

单片机产生不同频率脉冲信号的原理：

要产生音频脉冲，只要算出某一音频的脉冲(1/频率)，然后将此周期除以 2，即为半周期的时间，利用定时器计时这个半周期的时间，每当计时到后就将输出脉冲的 I/O 反相，然后重复计时此半周期的时间再对 I/O 反相，就可以在 I/O 脚上得到此频率的脉冲。

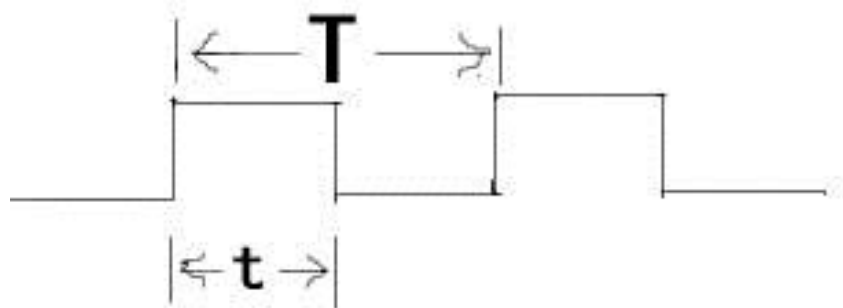


以 8051 单片机为例（8051 单片机的定时器每次计数时间为 1us）。利用 8051 的内部定时器使其工作在计数器模式 MODE2 下，改变计数值 TH0 及 TL0 以产生不同频率。

2.1.2 计数初值的计算

这里以标准音高 A 为例来详细说明，A 的频率 $f = 440 \text{ Hz}$ ，其对应的周期为：

$$T = 1/f = 1/440 = 2272 \quad \mu\text{s}$$



由上图可知，单片机上对应蜂鸣器的 I/O 口来回取反的时间应为：

$$t = T/2 = 2272/2 = 1136 \quad \mu\text{s}$$

这个时间 t 也就是单片机上定时器应有的中断触发时间。一般情况下，单片机奏乐时，其定时器为工作方式 1，它以振荡器的十二分频信号为计数脉冲。设振荡器频率为 f_0 ，则定时器的予置初值由下式来确定：

$$t = 12 * (T_{\text{ALL}} - T_{\text{HL}}) / f_0$$

式中 $T_{\text{ALL}} = 2^{16} = 65536$, T_{HL} 为定时器待确定的计数初值。因此定时器的高低计数器的初值为：

$$TH = T_{\text{HL}} / 256 = (T_{\text{ALL}} - t * f_0 / 12) / 256$$

$$TL = T_{\text{HL}} \% 256 = (T_{\text{ALL}} - t * f_0 / 12) \% 256$$

将 $t = 1136 \mu\text{s}$ 代入上面两式（注意：计算时应将时间和频率的单位换算一致），即可求出标准音高 A 在单片机晶振频率 $f_0 = 12 \text{ Mhz}$ ，定时器在工作方式 1 下的定时器高低计数器的予置初值为：

$$TH_{440\text{Hz}} = (65536 - 1136 * 12/12) / 256 = \text{FBH}$$

$$TL_{440\text{Hz}} = (65536 - 1136 * 12/12) \% 256 = 90\text{H}$$

2.1.3 音调

若以频率来表示声音，既抽象又无趣。在音乐中，通常是以 Do、Re、Mi、Fa、So、La、Si、Do 分别来代表某一个频率的声音，我们称之为“音调”，即 Tone。如图所示为 C 调音阶表，包括三个音阶（低音、中音与高音）每个音阶为八音度，其中细分为 12 个半音，而每个音阶之间的频率相差一倍。

音阶	n	1	2	3	4	5	6	7	8	9	10	11	12
		Do	Do#	Re	Re#	Mi	Fa	Fa#	So	So#	La	La#	Si
低音	频率	262	277	294	311	330	349	370	392	415	440	464	494
	简谱	1		2		3	4		5		6		7
中音	频率	523	554	587	622	659	698	740	784	831	880	932	988
	简谱	1		2		3	4		5		6		7
高音	频率	1046	1109	1175	1245	1318	1397	1480	1568	1661	1760	1865	1976
	简谱	· 1		· 2		· 3	· 4		· 5		· 6		· 7

根据该表，程序中音阶的定义为：

```
unsigned int code tone[]=
```

```
{1909, 1707, 1520, 1433, 1276, 1136, 1012, 956, 852, 759, 716, 638
```

```
568, 506, 478, 426, 379, 359, 320, 285, 254, 0, 10 };
```

/音阶定义

2.1.4 节拍

若要构成音乐，光有音调是不够的，还需要节拍，让音乐具有旋律（即具有一定节奏的律动）而且还可以调节各个音的快慢速度。“节拍”即 **Beat**。也就是通常意义上所谓的打拍子。简单介绍一下节拍的概念：

在一张乐谱中，我们经常会看到这样的表达式，如 $1=C \frac{4}{4}$ 、 $1=G \frac{3}{4}$ ……等等，这里 $1=C, 1=G$ 表示乐谱的曲调，和我们前面所谈的音调有很大的关联， $\frac{4}{4}$ 、 $\frac{3}{4}$ 就是用来表示节拍的。以 $\frac{3}{4}$ 为例加以说明，它表示乐谱中以四分音符为节拍，每一小结有三拍。比如：

```
1=C 3/4
| 1 2 3 4 5 6 |
```

其中 1、2 为一拍，3、4、5 为一拍，6 为一拍共三拍。1、2 的时长为四分音符的一半，即为八分音符长，3、4 的时长为八分音符的一半，即为十六分音符长，5 的时长为四分音符的一半，即为八分音符长，6 的时长为四分音符长。

在程序中我们定义 6 表示半个节拍，12 为一个完整的节拍。据此依次确定每首歌的节拍。比如《欢乐颂》：


```

Unsigned char code beat3[]={12,12,12,12, 12,12,12,12,
12,12,12,12, 18,6,24,
12,12,12,12, 12,12,12,12,
12,12,12,12, 18,6,24,
12,12,12,12, 12,6,6,12,12,
12,6,6,12,12, 12,12,12,12,
12,12,12,12, 12,12,12,12,
12,12,12,12, 18,6,24
};

```

2.2 系统硬件电路设计

2.2.1 单片机最小系统设计

如图 2-1 所示，是单片机的最小系统电路，只有保证这部分电路正常工作，整个系统才能运行。

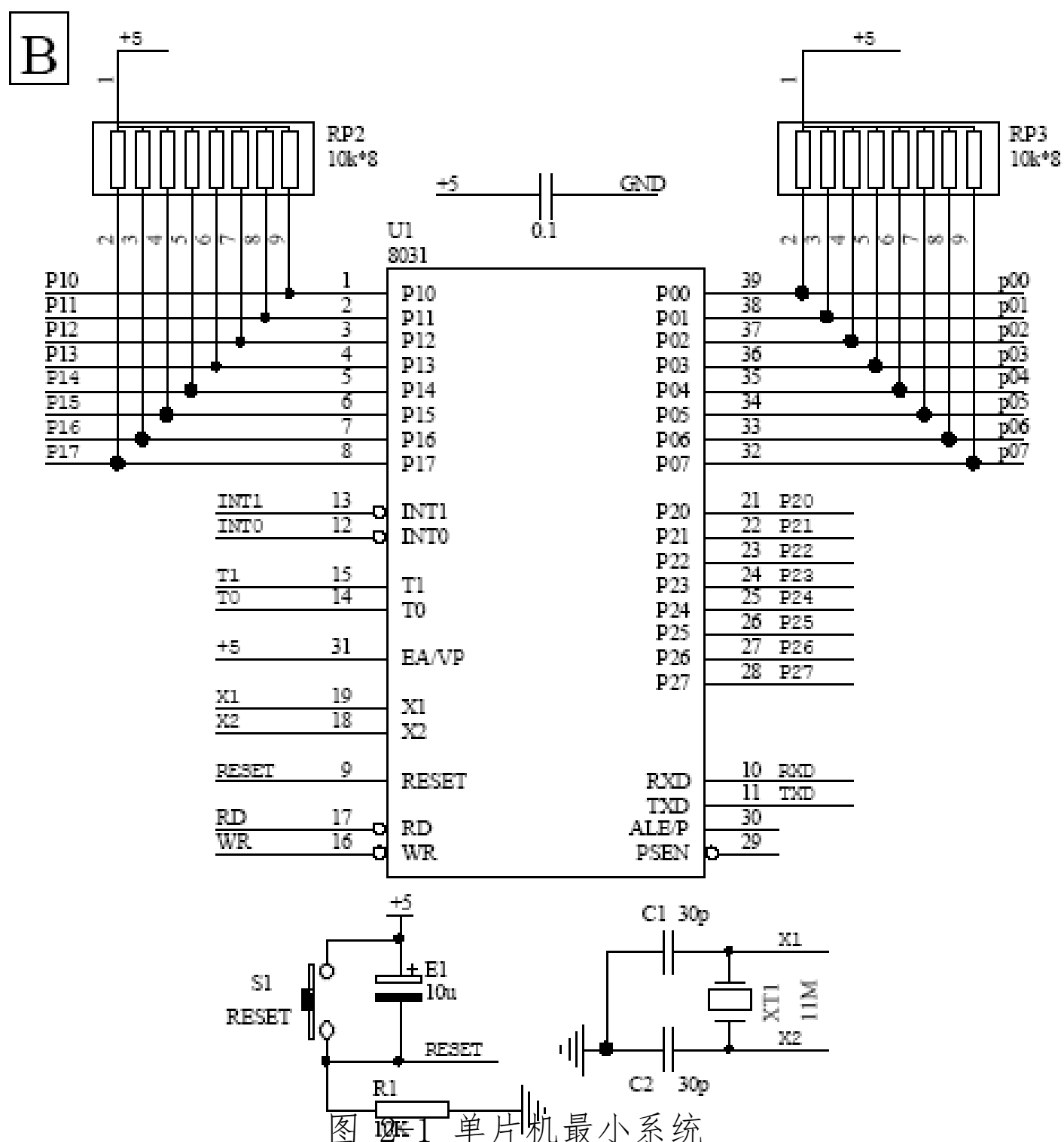


图 2-1 单片机最小系统

8KFLASH程序存储器的 STC89C52RC EA 接高电平；各并行口都加了 10K 的上拉电阻；晶振为 11.0592MHZ 同时设置了上电复位和手动复位电路。S1 为手动复位按钮。

2.2.2 键盘接口电路

如图 2-2 所示，为键盘接口电路，包含行列式键盘和独立式按键输入电路。其中 P20—P23 为行，P24—P27 为列，构成 4×4 行列式键盘。按键编号规律为 S2+行号+列号。例如 S226 表示 P22 行和 P26 列跨接处的按键。

K20—K27 构成独立式按键，分别接在 P20-P27 上，当按键按下时，对应的口线输入置 0，同时点亮相应的 LED

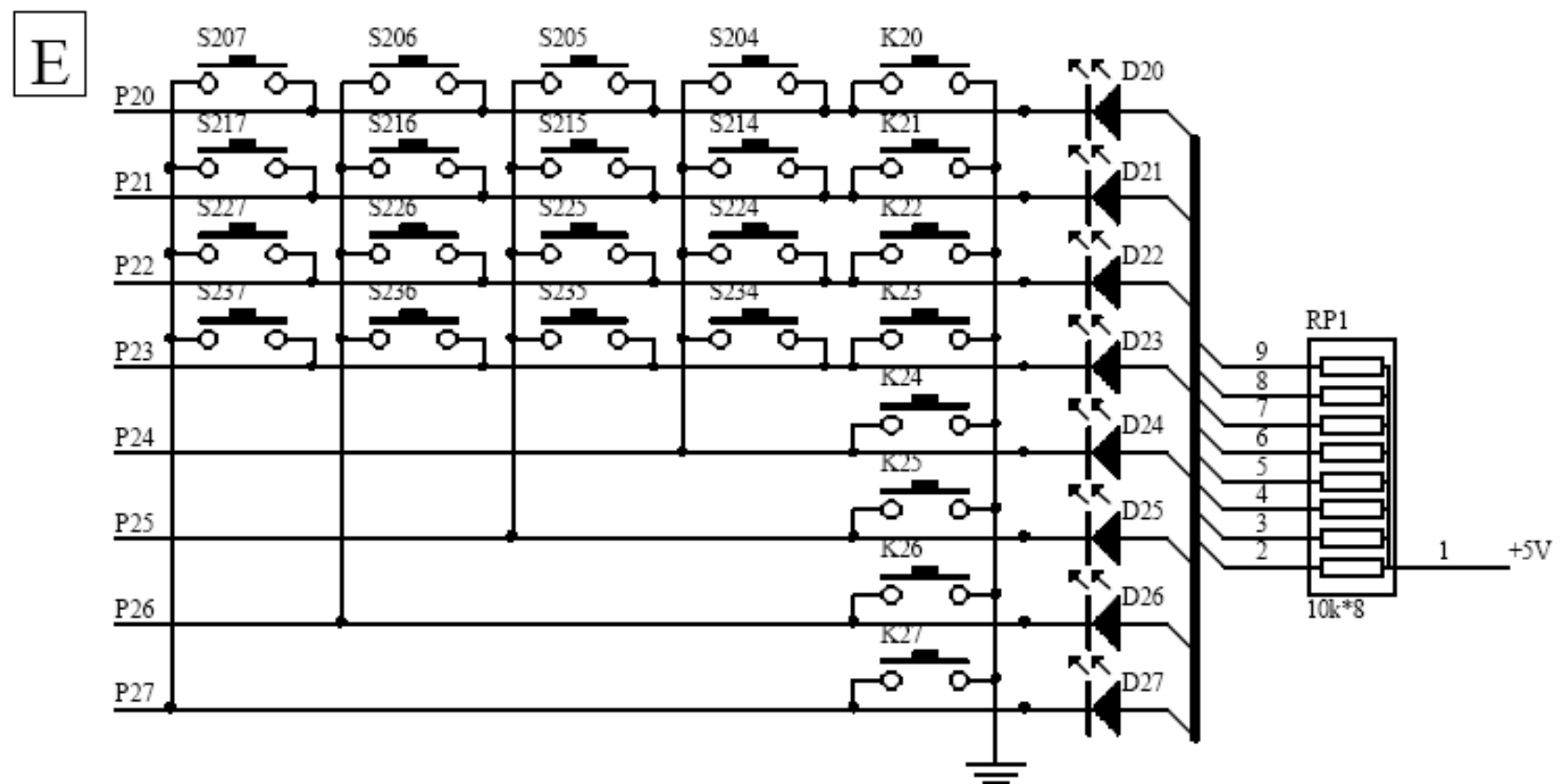


图 2-2 键盘接口路

D20—D27构成 P20—P27 的状态指示，当某口线输出 0 时，相应的 LED 点亮。有 4 个用途：1、当行列式键盘逐行扫描时，可做为正在被扫描的行的指示。

2、当行列式键盘逐行扫描时，可做为已按下键的指示。例：当 S215 键按下，扫到 P21 行时，D21 和 D25 会被点亮。

3、可做为独立式按键被按下的指示。

4、当 P2 口不做为键盘输入时，D20—D27 可做为输出器件，实现流水灯等功能。

数码管显示路

如图 2-3 所示,为数码管显示电路。这部分电路由 4 位一体的数码管 LED1、单个数码管 LED2 和 8 个独立发光二极管 L1-L7 组成,这3个部分都是共阴结构,并联在一起,连接在 P0 口上。

数码管段选定义

sbit smg_2 = P2^6;

sbit smg_3 = P2^5;

sbit smg_4 = P2^4;

其中 LED1 用于完成 LED 动态显示实验,各个位选线为 P24—P27,段码由 P0 口输出; LED2 的 COM 端接地,完成 LED 静态显示实验; L1—L7,可以方便地读出 P0 输出的二进制数据,亦可做流水灯实验。

7407 在此起驱动作用。

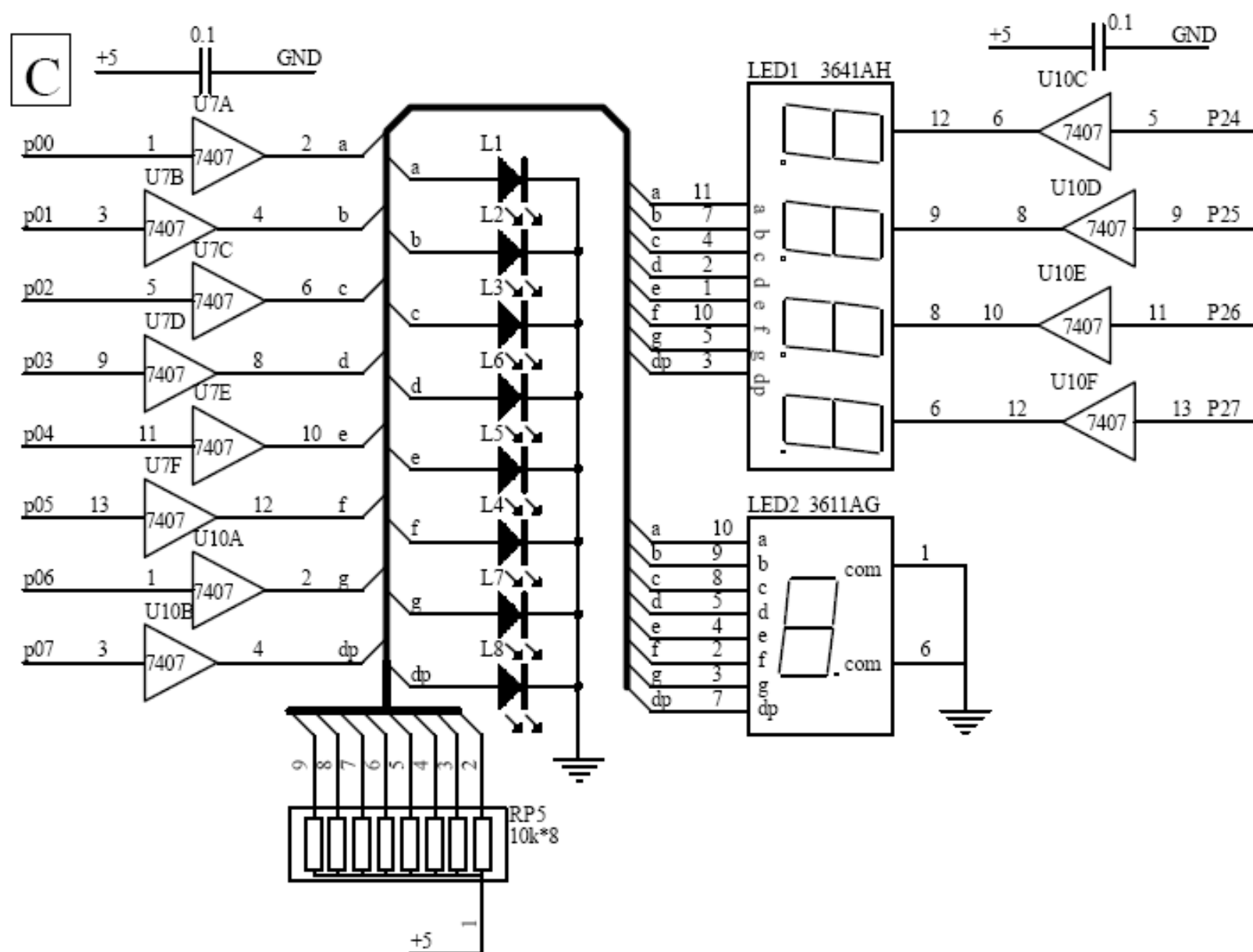


图 2-3 数码管显示电路

外部中断0触发电路

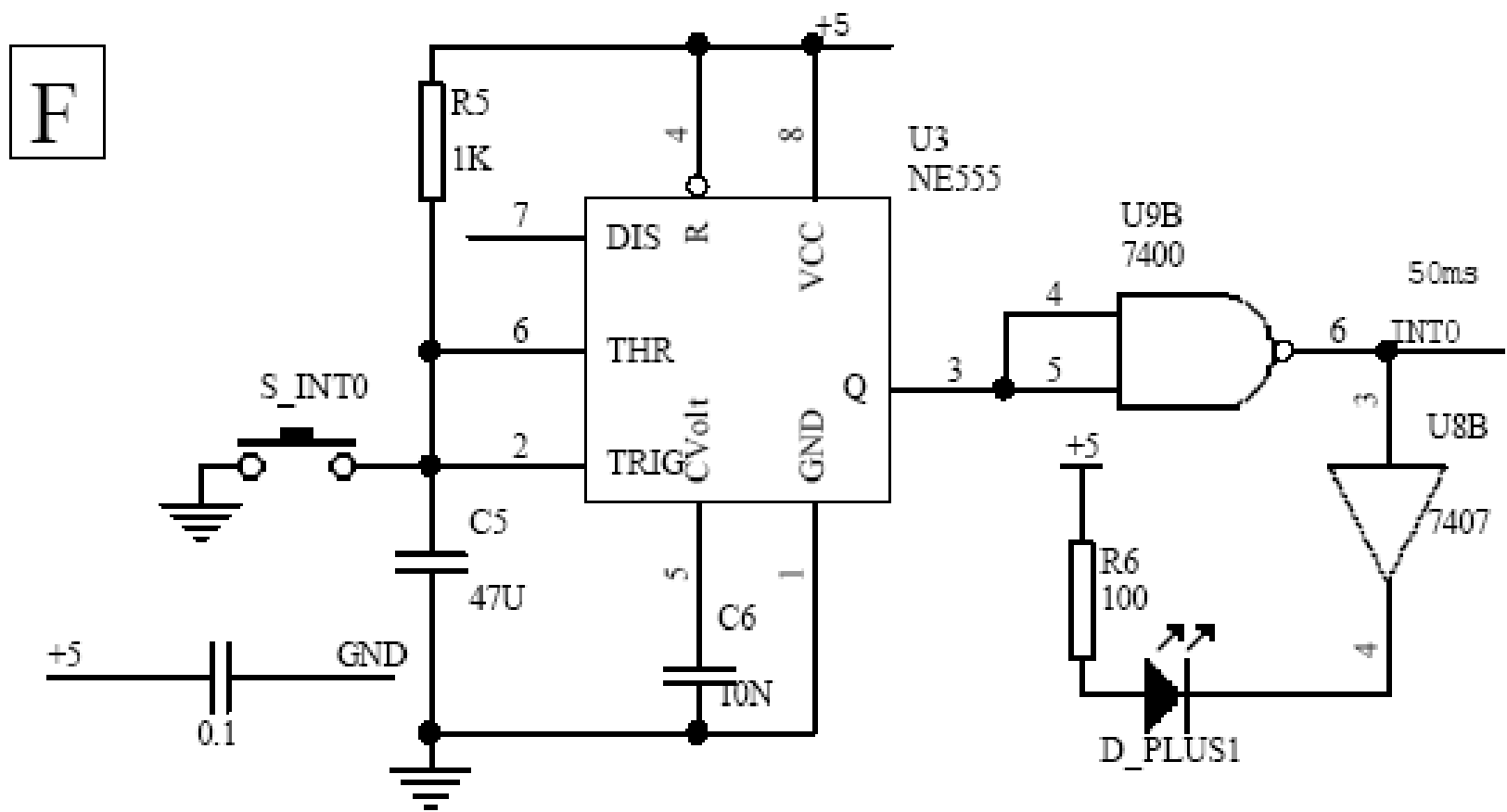


图 2-4 外部中断0触发

注：图 2-4 所示的电路为 NE555 和电容组成的单稳电路。

当 S_INT0 按钮按下后产生一个低电平单稳信号送单片机的INT0，同时 D_PLUS1亮一次，作为外部中断请求信号。

2.2.5 拨码开关电路

如图 2-5 所示，为拨码开关电路。该电路由一个 8 位拨码开关 (U11) 和 8 个 LED (D00—D07) 组成。连在 P1 口上完成基本输入输出功能。

输入功能：当某开关闭合后，对应 P1 口线上输入 0，同时相应的 LED 点亮。否则，对应 P1 口线上输入 1，同时相应的 LED 熄灭。例如：第一位开关闭合，D10 点亮，同时 P1.0 输入 0。LED 可以作为开关是否闭合的直观指示。

输出功能：当某开关打开时，用 P1 口线作为输出引脚使用。当输出 1 时，相应的 LED 熄灭，当输出 0 时，相应的 LED 点亮。当某开关闭合时，无法做输出功能使用。

A

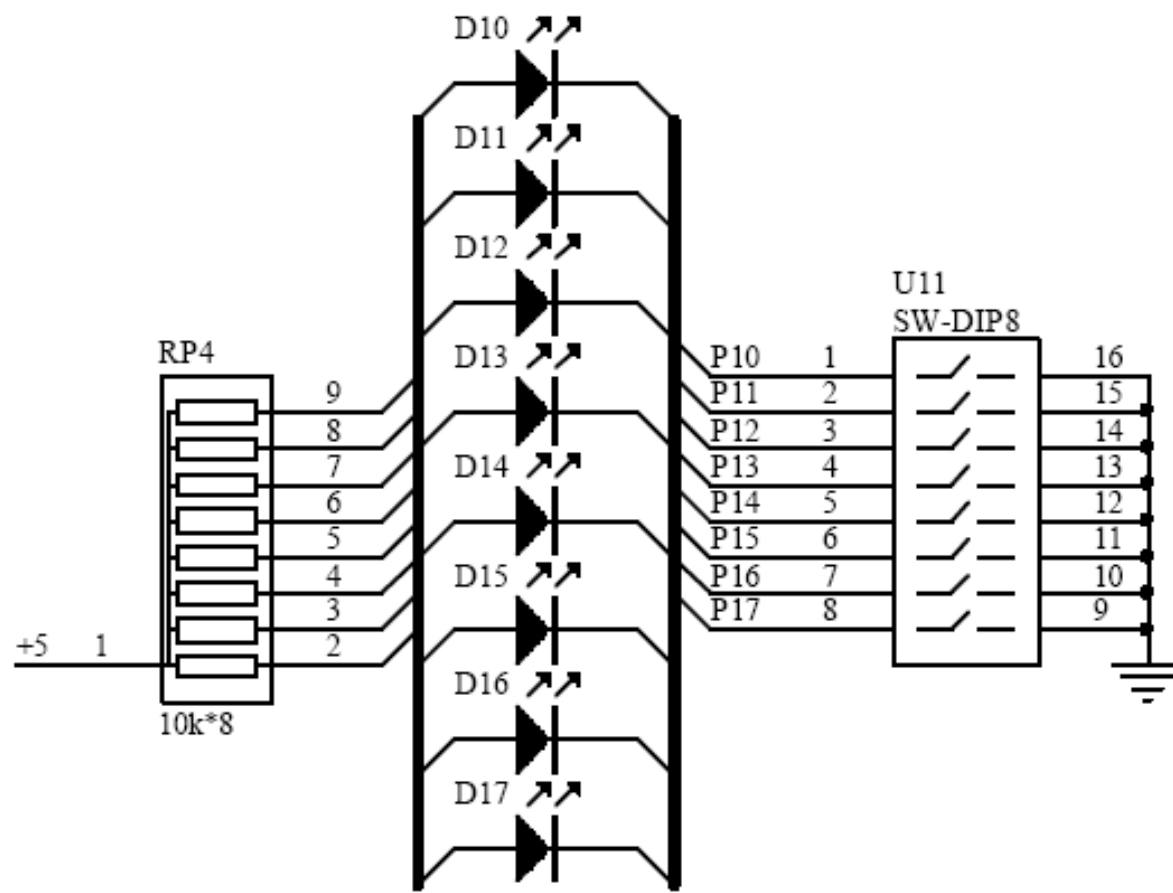
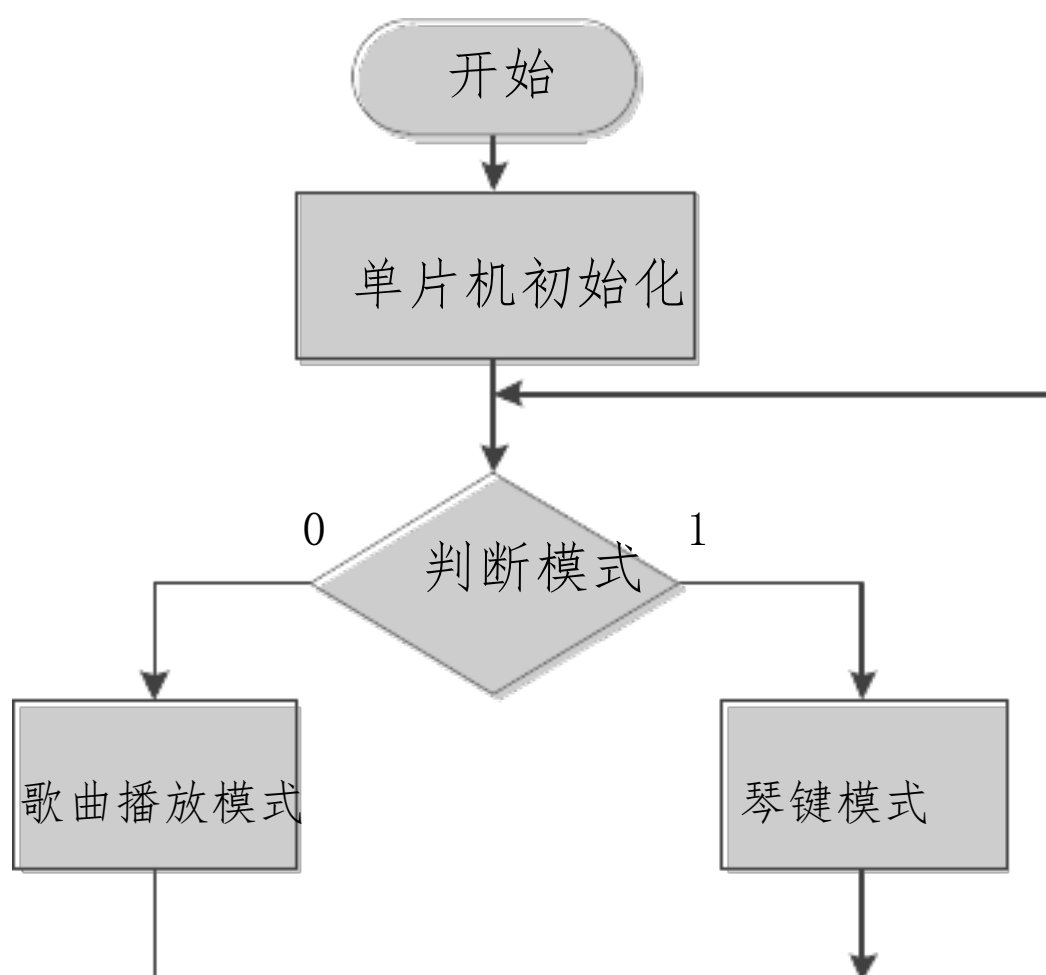


图 2-5 拨码开关电路

2.3 系统软件设计

2.3.1 系统软件方框图



单片机初始化

单片机初始化主要包括定时器初始化和中断初始化。定时器初始化主要包括配置定时器 0、1 的工作方式以及开中断。中断初始化主要包括开中断、配置中断触发方式。具体如下：

```
{
    TMOD|=0x11;
    IE |= 0x8a;
}
void INT_Init(void)
{
    IE|=0x81;
    IP|=0x01;
    IT0 = 0;
}
```

2.3.3 音乐播放函数

音乐播放函数主要用来实现音乐的播放,同时在音乐切换或者模式切换时能快速实现终止当前播放的音乐。具体编程如下：

```
void play(unsigned char* song,unsigned char* beat )
{
    i=0; //
    while(song[i]!=30 && !song_over) // while 循环开始
    {
        tone_H=(65536-tone[song[i]])/256; //读取音阶计数量之高八位元
        tone_L=(65536-tone[song[i]])%256; //读取音阶计数量之低八位元
        TH0=tone_H; //填入音阶计数量之低八位元
        TL0=tone_L; //填入音阶计数量之低八位元
        TH1 = beat_H;
```

```

TR0=1;
TR1=1;          //    Timer 0
beat_flag = 0;
beat_times = beat[i];
while(beat_flag == 0);

i++;           //

    }

}

one_bit_static_smg(~SMG[choose_song + 1]);
switch(choose_song) /选择播放曲目

{
case 0: play(song1,beat1);
        break;
case 1: play(song2,beat2);
        break;
case 2: play(song3,beat3);
        break;
case 3: play(song4,beat4);
        break;
case 4: play(song5,beat5);
        break;
case 5: play(song6,beat6);
        break;
case 6: play(song7,beat7);
        break;
case 7: play(song8,beat8);
        break;
default:play(song1,beat1);

        break;

}

}

```

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/986205024124010214>