

哈尔滨工业大学（深圳） 工程训练（电子工艺实习）	文档编号	产品版本	密级
	GX-DG-2019-09-02-01	V 1.0	/
	产品名称：哈工大校庆软件设计说明书		共 14 页

# 软件设计说明书

(仅供内部使用)

小组成员：李木晗 林俊锟 杨丰源 高颂恩      日期：9/2/2019

哈尔滨工业大学（深圳）

版权所有 不得复制

---

文件状态：	文件标识：	哈工大校庆软件设计说明书
<input type="checkbox"/> 草稿	当前版本：	1.0
<input checked="" type="checkbox"/> 正式发布	作 者：	李木晗 林俊锟
<input type="checkbox"/> 正在修改	完成日期：	2019-9-2

软件综合设计

---

## 目 录

一 引 言.....	1
1.1 编写目的.....	1
1.2 读者对象.....	1
1.3 软件项目概述.....	1
1.4 文档概述.....	1
1.5 定义.....	1
二 软件设计约束.....	2
2.1 设计目标和原则.....	2
2.2 设计约束.....	2
三 软件设计描述.....	3
3.1 总体设计.....	3
3.1.1 功能说明.....	3
3.1.2 功能框图.....	3
3.2 接口协议.....	4
3.3 程序编写.....	4
3.3.1 主程序.....	4
3.3.2 子程序.....	6
3.4 源文件列表.....	11
3.5 相关参考资料及文档.....	11

# 一 引 言

## 1.1 编写目的

更好地展现小组作品，阐述系统结构和设计思路，清晰说明系统设计中每个部分的原理，方便读者更加了解我们的设计系统。

## 1.2 读者对象

系统使用者；在校师生；MAX7219 点阵的使用者；编程初学者等

## 1.3 软件项目概述

为实现课程要求，以工大百年校庆为主题，使用单片机和点阵实现灯光秀的效果，运用课程所学知识，共同迎接工大百年的到来。开发目的为以下：

- (1) 学习 MAX7219 的硬件和程序编程
- (2) 学习点阵字模的设计
- (3) 学习 SPI 通信协议的原理
- (4) 掌握嵌入式系统开发环境的建立和使用
- (5) 掌握嵌入式系统基本驱动、应用程序的设计调试和编译移植方法

## 1.4 文档概述

本文档整体分为引言、软件设计约束、软件设计描述，从三方面对所搭建的嵌入式系统进行详细说明。包括软件的设计目的、设计思路、功能介绍、流程图、程序源代码、实验效果等。

## 1.5 定义

SPI: 串行外设接口

ASCII ((American Standard Code for Information Interchange): 美国信息交换标准代码) 是基于拉丁字母的一套电脑编码系统，主要用于显示现代英语和其他西欧语言。

## 二 软件设计约束

### 2.1 设计目标和原则

课程要求为：

1. 使用 8\*8 的 led 点阵 1-4 块，实现"哈尔滨工业大学百年校庆，规格严格，功夫到家"汉字滚动显示。
2. 自行设计多个图片或图形动态显示(不少于两个)，如心形,礼花等，显示过程中可添加文字，表达对母校百年校庆的祝福，实现"灯光秀"的效果。
3. 添加背景音乐，利用蜂鸣器或者扬声器奏响哈尔滨工业大学校歌，使用开发板的 LED1-LED6 显示音高柱

本系统完成了要求 1、2 和 3，使用 4 块 8\*8LED 点阵，设计了心形、礼花的动态显示，并在其中加入可滚动显示的文字，并且可以同时播放校歌为背景音乐。

在设计过程中，遵循的一般原则：

- 编码的数组采用二进制，方便修改和查看效果
- 根据功能划分程序，规划程序结构
- 添加注释，方便使用者理解程序

### 2.2 设计约束

- 硬件平台
  - MSP430F5529 实验开发板
  - 4 块 8\*8 点阵 LED
  - 两个面包板
  - 杜邦线
- 开发语言：C 语言
- 开发工具：Code Composer Studio

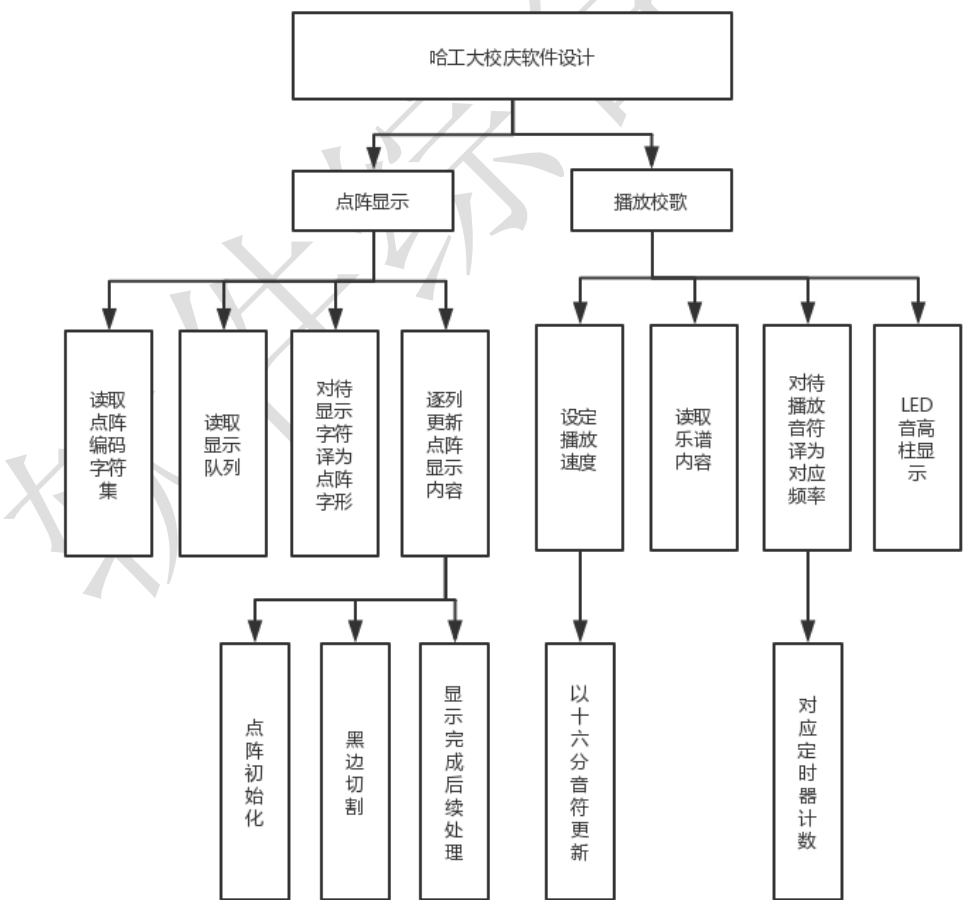
### 三 软件设计描述

#### 3.1 总体设计

##### 3.1.1 功能说明

- 循环滚动显示“哈尔滨工业大学深圳 规格严格功夫到家 1920-2020 百年校庆”；
- 循环显示心形和礼花的动态图片；
- 播放哈工大校歌，显示音高柱。

##### 3.1.2 功能框图



## 3.2 接口说明

- 软件接口

MSP430F5529 单片机上的 P6.0 对应点阵上的 IN 串行数据输入端

P6.1 对应 CS 片选端

P6.2 对应 CLK 时钟序列输入端

I/O 口上的高低电平即对应接口的高低电平

在 CLK 位于上升沿时，数据才能从 IN 被载入内部。MAX7219 连续数据的后 16 位在 CS 端的上升沿时被锁定。

程序设计如下：

```
#define CS_1 P6OUT |= BIT1
#define CS_0 P6OUT &= ~BIT1
#define CLK_1 P6OUT |= BIT2
#define CLK_0 P6OUT &= ~BIT2
#define DIN_1 P6OUT |= BIT0
#define DIN_0 P6OUT &= ~BIT0
```

## 3.3 程序编写

### 3.3.1 主程序

**int main(void)**

(1) 功能

输入要显示的内容，并使点阵能够循环显示；

显示部分字符画动画；

同时播放哈工大校歌（进中断），显示音高柱；

(2) 入口参数

```
(char*) character;    // 字符集
(int) pos;            // 记录歌曲播放进度
(char) flag;          // 是否播放歌曲
(int) speed_ctrl;     // 控制播放速度
(char) display_flag_a; // 是否显示字符画 A
(char) display_flag_b; // 是否显示字符画 B
```

(3) 出口参数

无

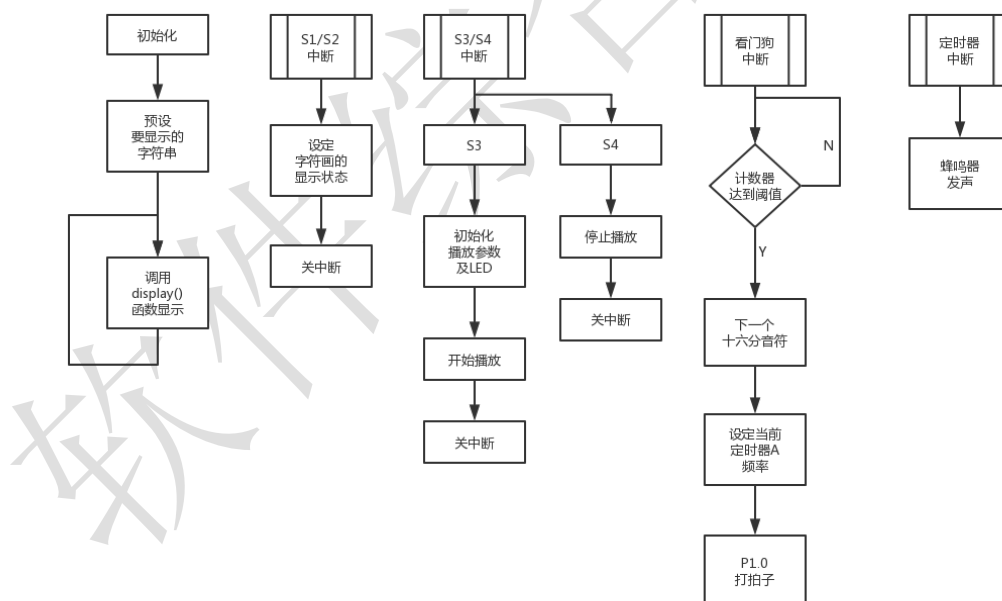
(4) 局部变量

```
(char*) text0;    // 显示的字符串  
(char*) text1;  
(char*) text2;  
(char*) text3;
```

(5) 函数调用

```
init1();          // 点阵初始化  
init2();          // 中断初始化  
display();        // 显示字符串  
Port_1_Key();     // S1/S2 中断，控制字符画  
Port_2_Key();     // S3/S4 中断，控制校歌  
Timer_A0();       // 定时器控制声音频率  
watchdog_timer(); // 看门狗打节拍
```

(6) 流程图





### 3.3.2 子程序

#### 3.3.2.1 void write\_byte(unsigned char data)

(1) 功能

对 MAX7219 写入一个字节的数据

(2) 入口参数

(unsigned char) data

(3) 出口参数

无

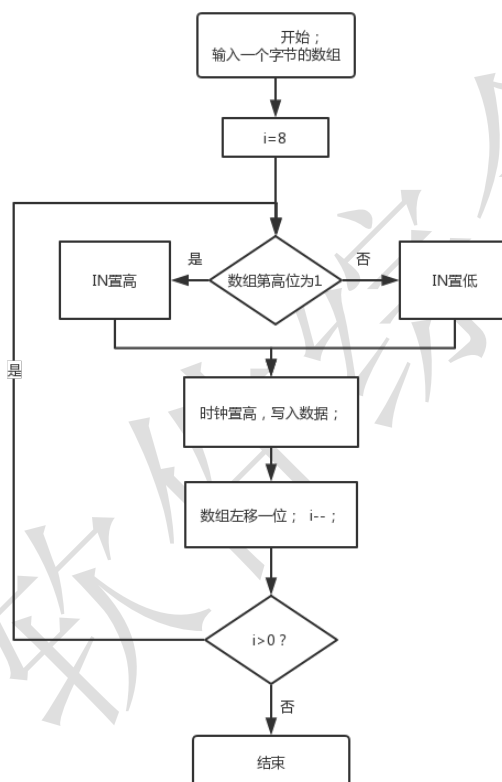
(4) 局部变量

(char) i

(5) 函数调用

无

(6) 流程图



#### 3.3.2.2 void write(unsigned char addr1, unsigned char data1, unsigned char addr2, unsigned char data2, unsigned char addr3, unsigned char data3, unsigned char addr4, unsigned char data4)

(1) 功能

对四个级联的点阵写入数据

(2) 入口参数

(unsigned char) addr1,  
(unsigned char) data1,  
(unsigned char) addr2,  
(unsigned char) data2,  
(unsigned char) addr3,  
(unsigned char) data3,  
(unsigned char) addr4,  
(unsigned char) data4)

(3) 出口参数

无

(4) 局部变量

无

(5) 函数调用

write\_byte(unsigned char data);

(6) 流程图



### 3.3.2.3 void init1(void)

(1) 功能

初始化单片机和 MAX7219

(2) 入口参数

无

(3) 出口参数

无

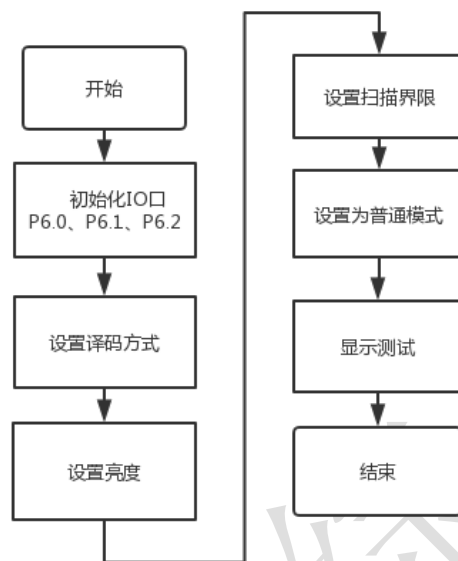
(4) 局部变量

无

(5) 函数调用

```
write(unsigned char addr1, unsigned char data1,  
      unsigned char addr2, unsigned char data2,  
      unsigned char addr3, unsigned char data3,  
      unsigned char addr4, unsigned char data4);  
  
delay_ms();
```

(6) 流程图



### 3.3.2.4 unsigned int char2bit(char m)

(1) 功能

对要显示的内容进行译码

(2) 入口参数

(char) m // 待显示字符

(3) 出口参数

(int) n // 该字符在字符集数组中的位置

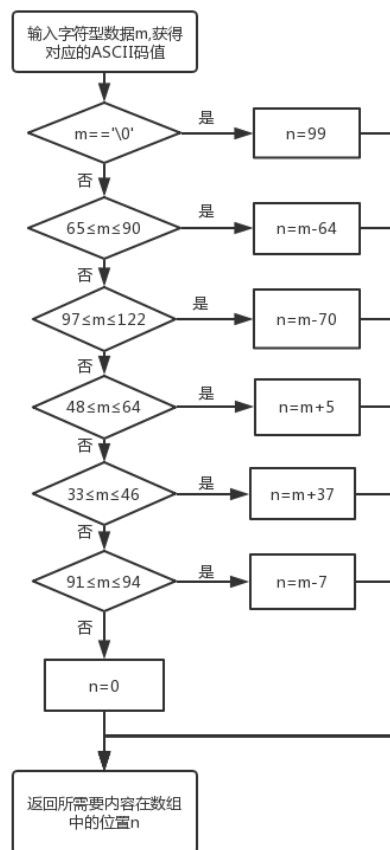
(4) 局部变量

(int) n

(5) 函数调用

无

(6) 流程图



3.3.2.5 display(char\* text)

(1) 功能

使点阵中间滚动显示输入数组译码后的内容  
侧边显示动态心形和礼花的图片

(2) 入口参数

(char\*) text // 待显示内容

(3) 出口参数

无

(4) 局部变量

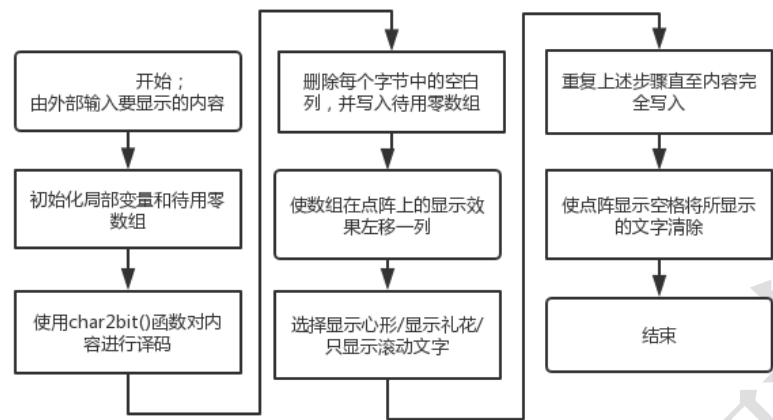
(int) i,j,k,m,n,q

(unsigned char) led[4][8] // 点阵内容

(5) 函数调用

无

(6) 流程图



3.3.2.6 void init2(void)

(1) 功能

初始化 GPIO 和定时器

(2) 入口参数

无

(3) 出口参数

无

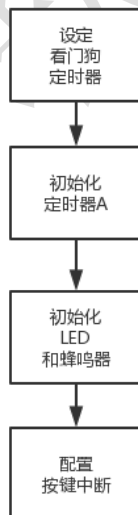
(4) 局部变量

无

(5) 函数调用

无

(6) 流程图



### 3.3.2.7 void set\_frequency (int x)

(1) 功能

设置定时器 A 频率（根据音阶频率换算关系，设定 TA0CCR0 计数个数）

(2) 入口参数

(int) x          // 编码后的简谱音符

(3) 出口参数

无

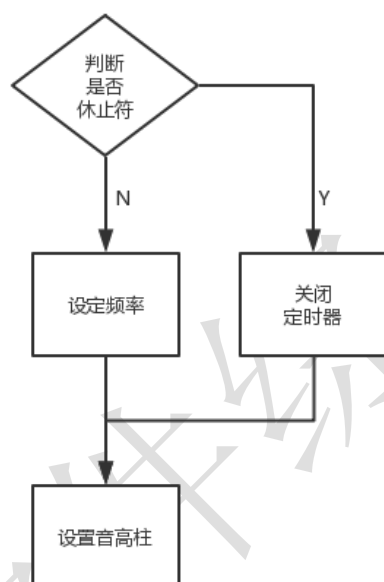
(4) 局部变量

无

(5) 函数调用

led\_off();          // LED 状态初始化

(6) 流程图



## 3.4 源文件列表

[main.c](#)

[HIT\\_LED.mp4](#)

## 3.5 相关参考资料及文档

[MAX7219 中文.pdf](#)

[MAX7219 英文.pdf](#)

[点阵资料.pdf](#)

[MAX7219 点阵原理图.pdf](#)