FreeRTOS车载音响

项目内容要求：

1.背景介绍

1)汽车音响必不可少的两个功能：

①Source 切替:

汽车音响通常都会支持多种音源播放，例如 收音机，CD, DVD, USB,蓝牙，aux等。Source切替就是指切换当前的播放音源。

②Diag ：

汽车音响都会提供一种基本功能检查的方法，该方法通常用来进行一些硬件是否正常的基础检查。这些方法都是不对用户公开的，因此进入方式对用户都是保密的。为了避免用户误操作进入，因此进入方式都会采用一些不易操作的方法。

汽车音响上的按键通常都有两种操作方式，短压 和 长压。

短压：是指在规定时间内完成完成压下及抬起操作。

长压：是指压下操作持续时间超过规定时间。

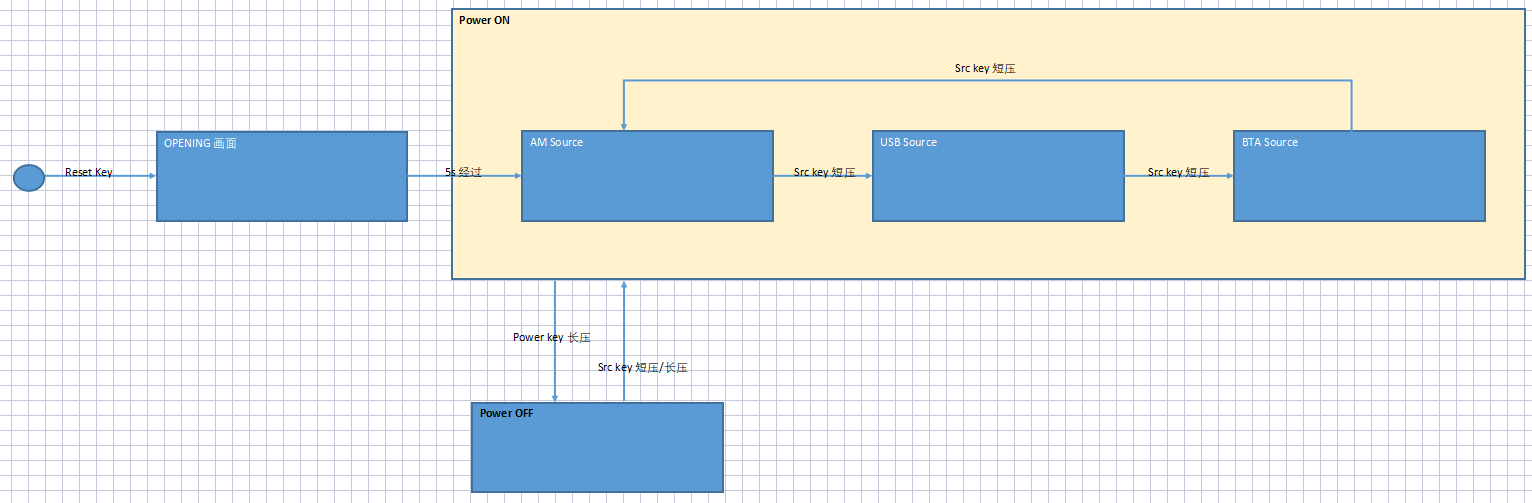
因此对于按键的检测，通常会识别处以下几种状态，压下，短压抬起，长压，长压抬起。

本式样中长压要求时间为 1.7 S。

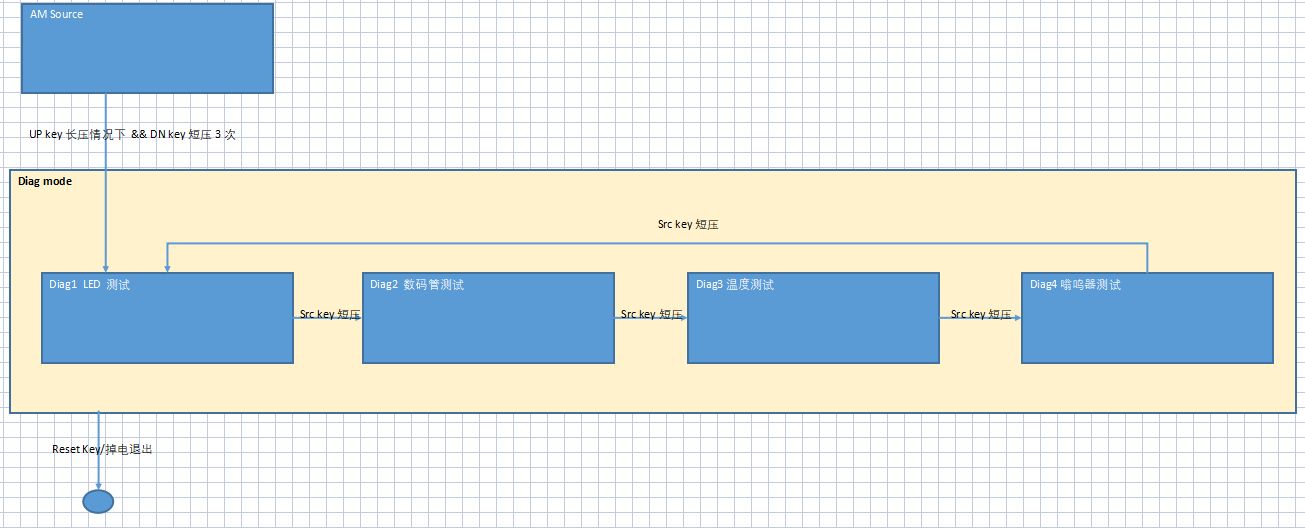
对我的

1. 画面迁移要求

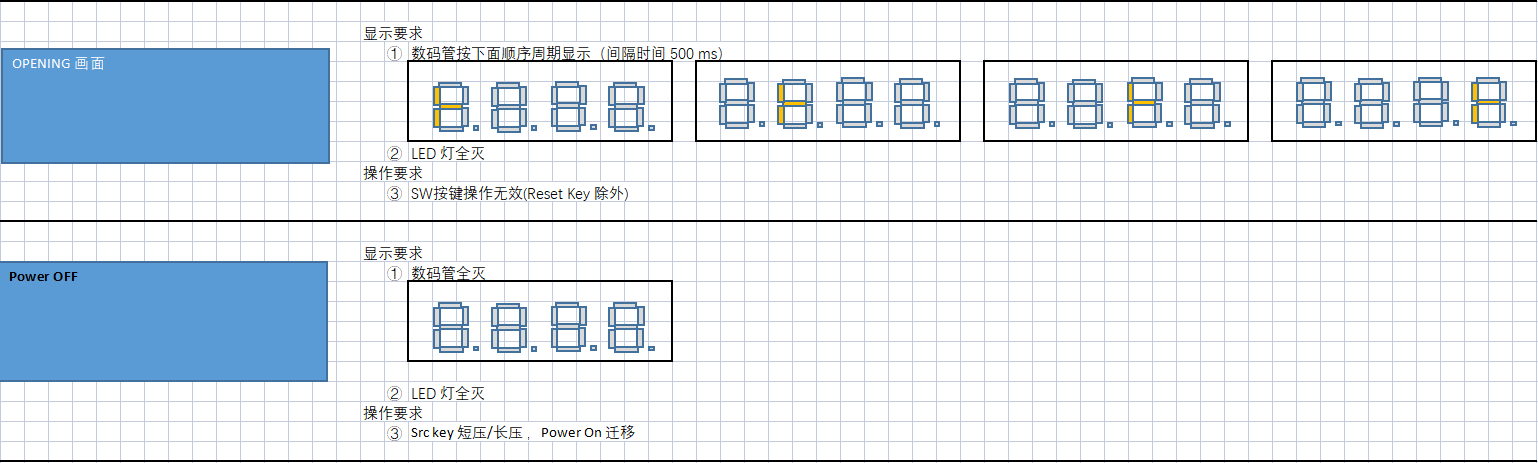
1)非DIGA模式画面迁移

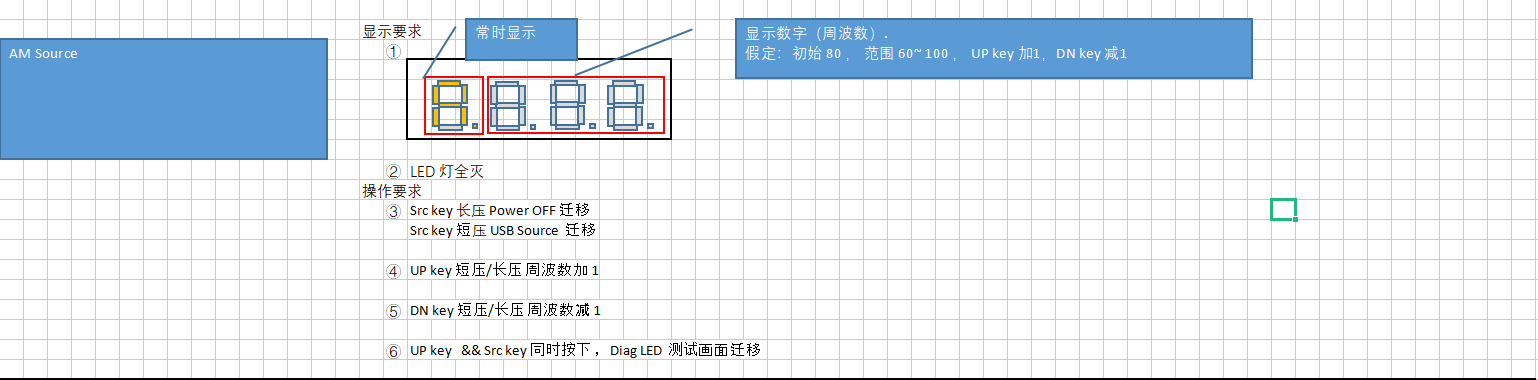


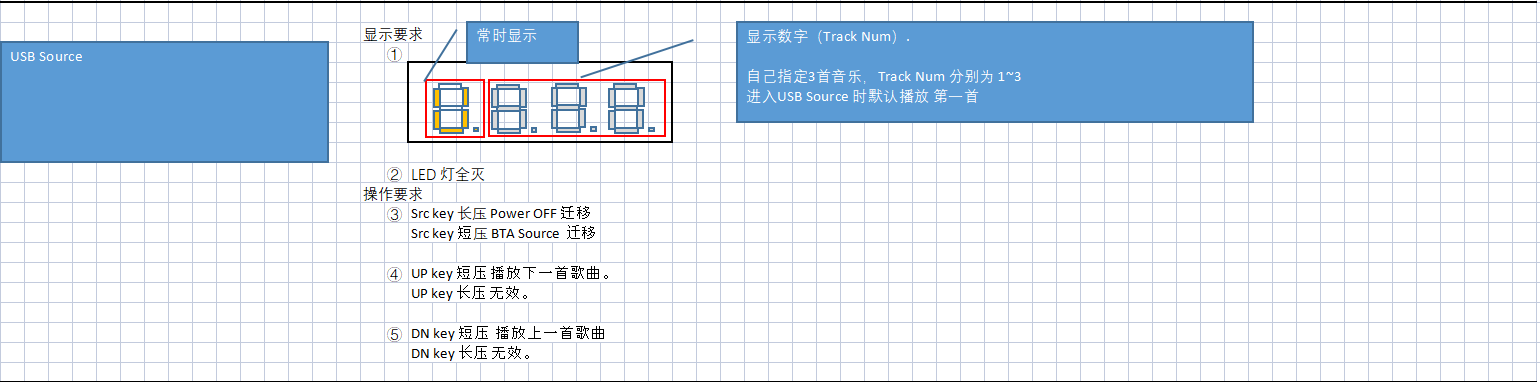
2)DIGA模式画面迁移

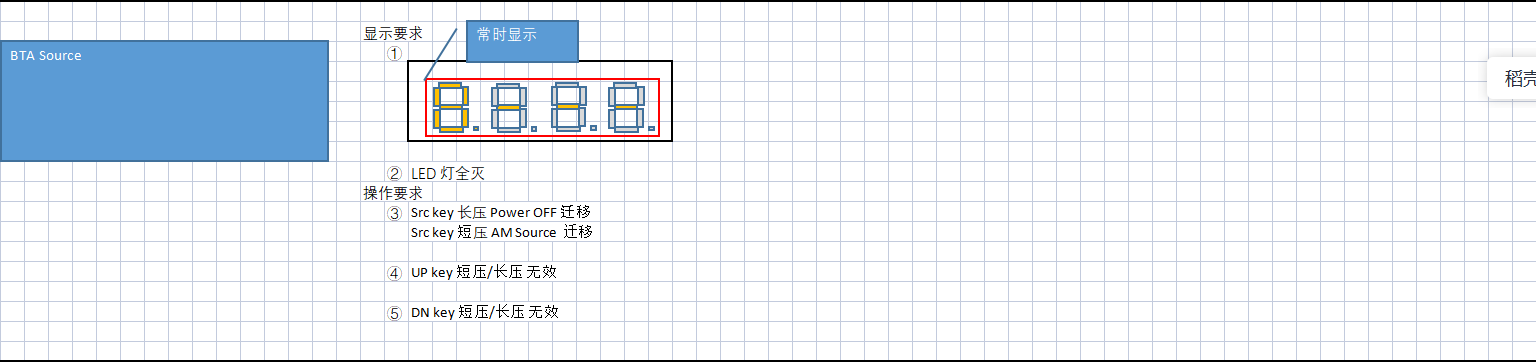


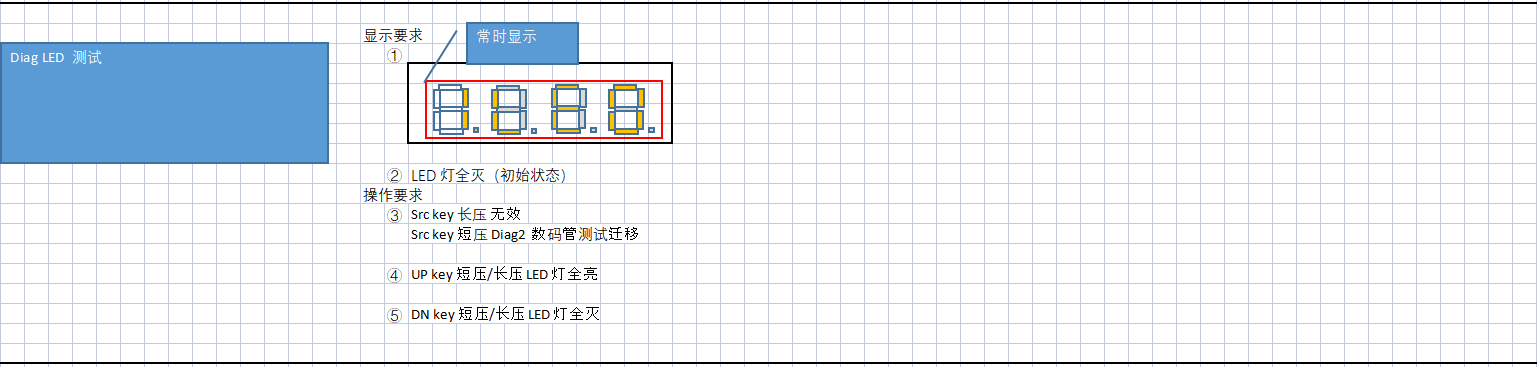
1. 显示及操作要求

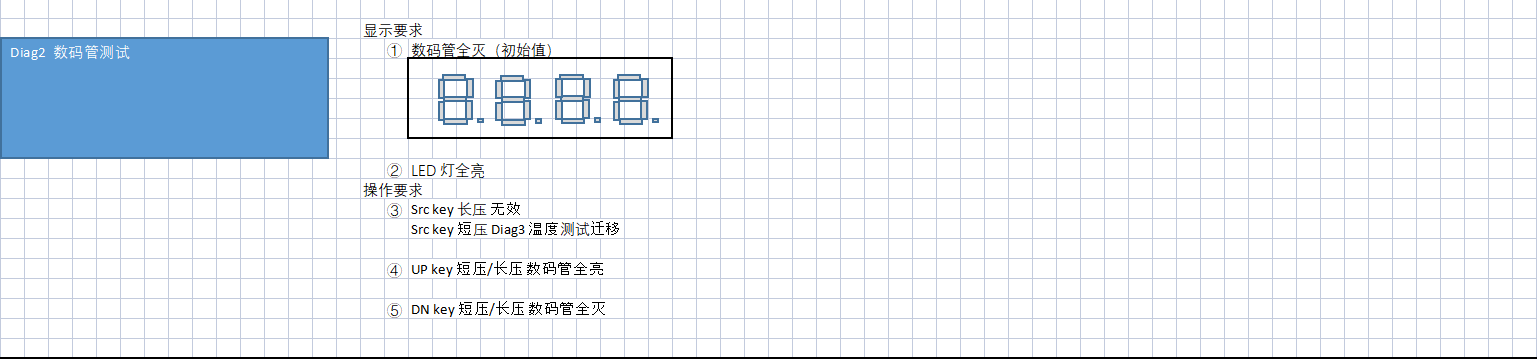


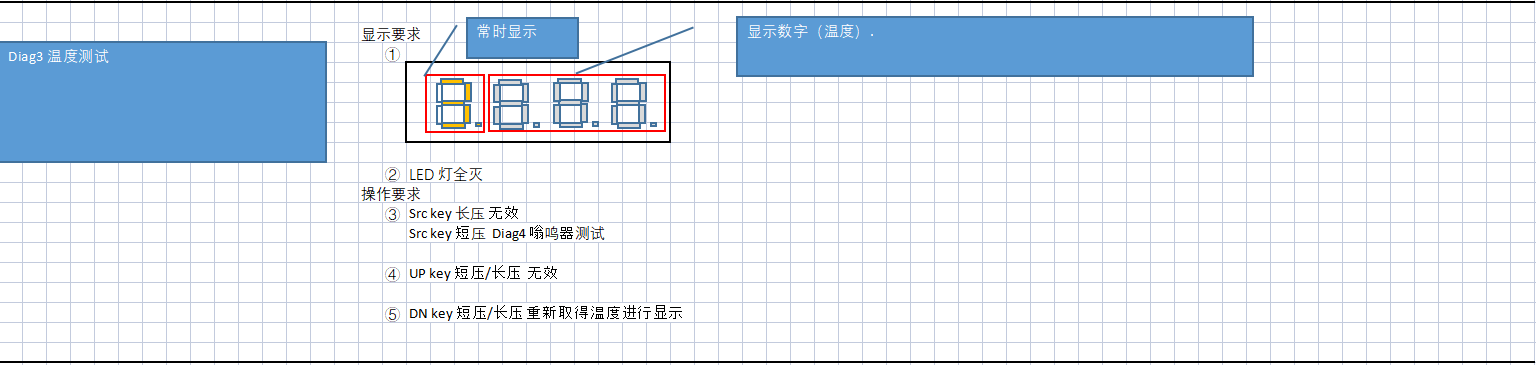


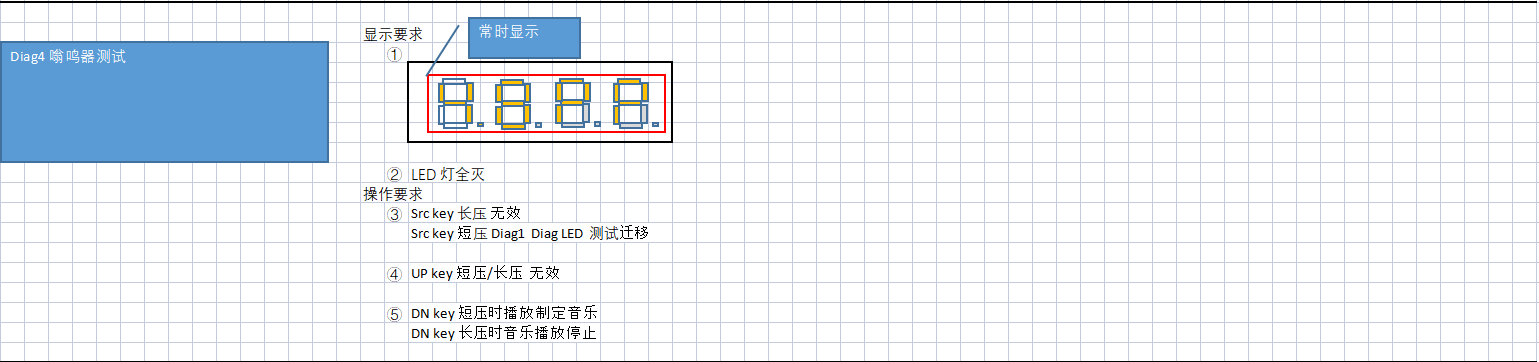




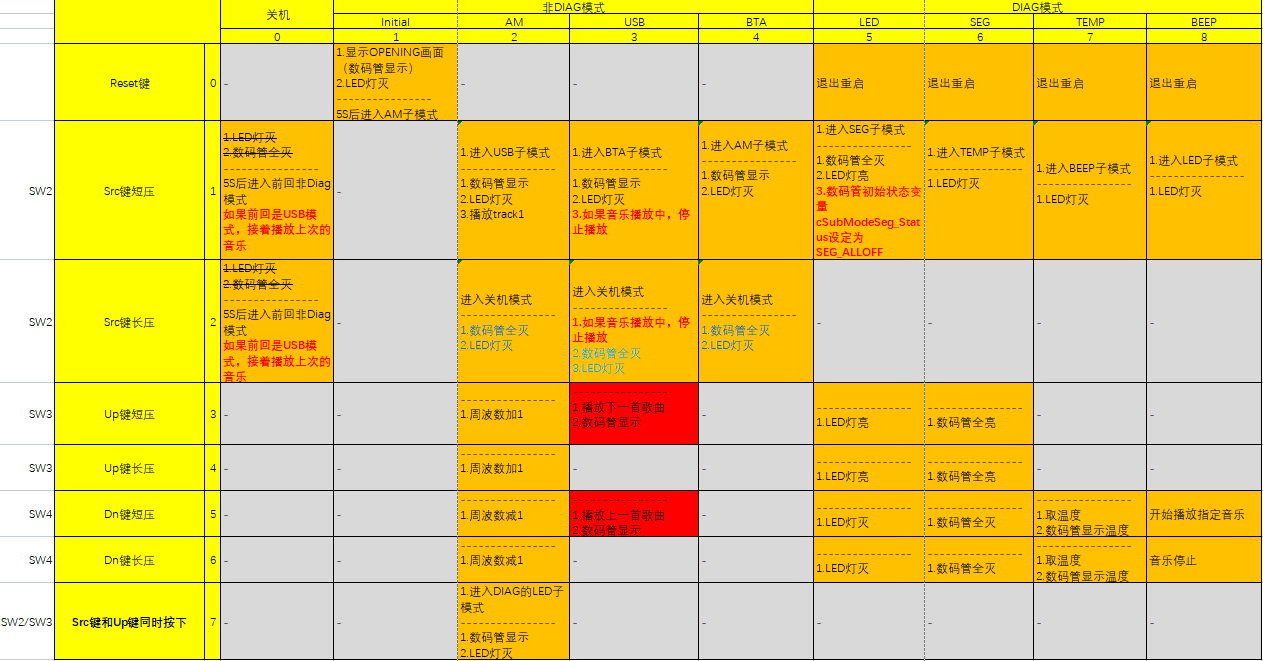




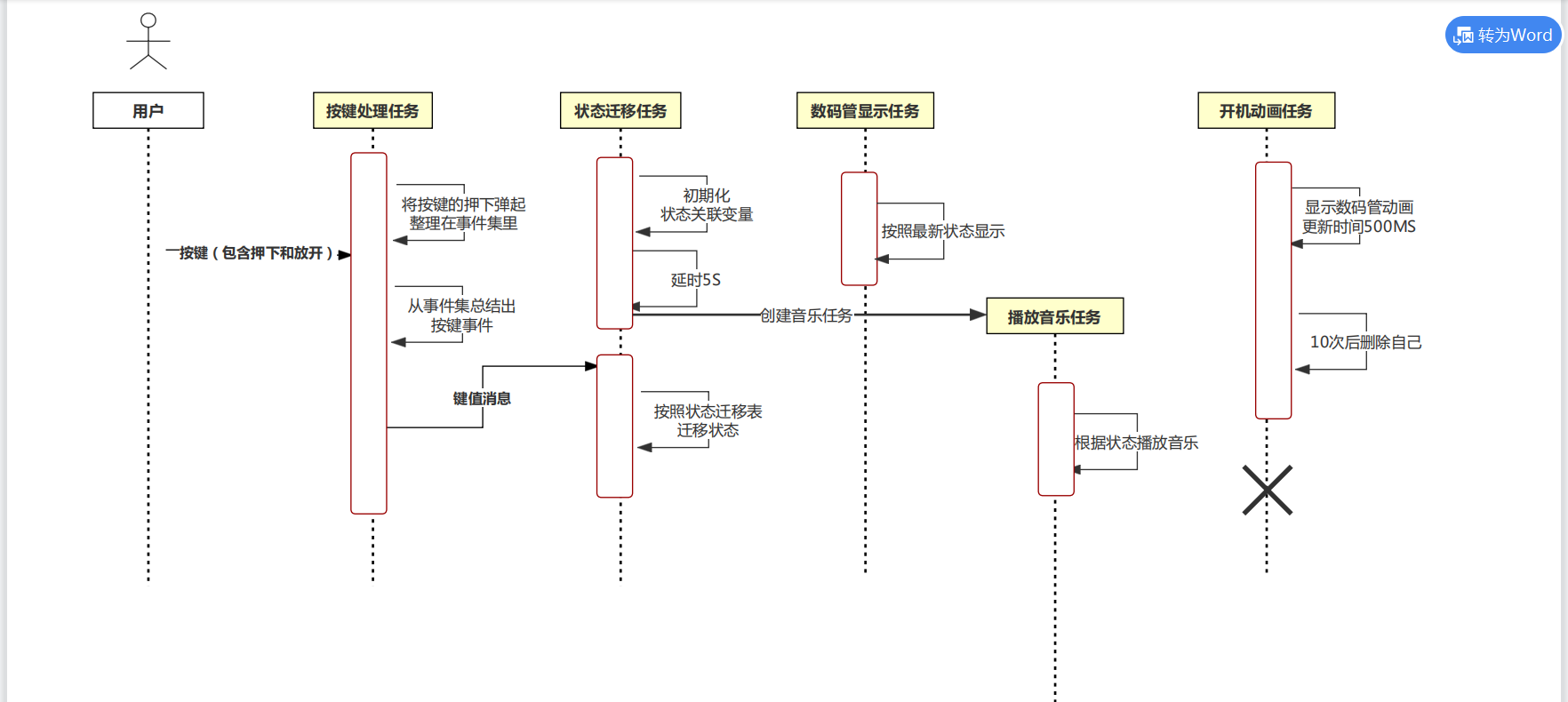




1. 状态迁移表

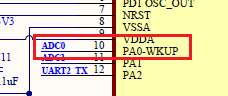
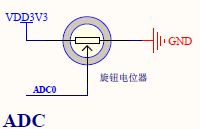


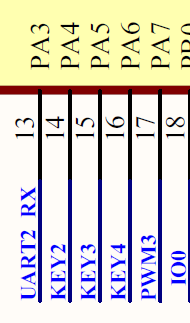
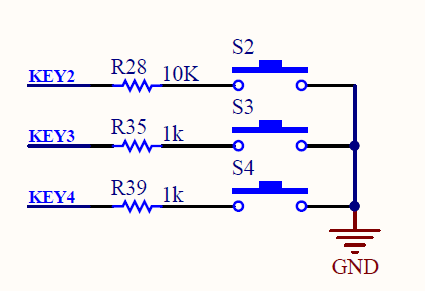
1. SEQUENCE图

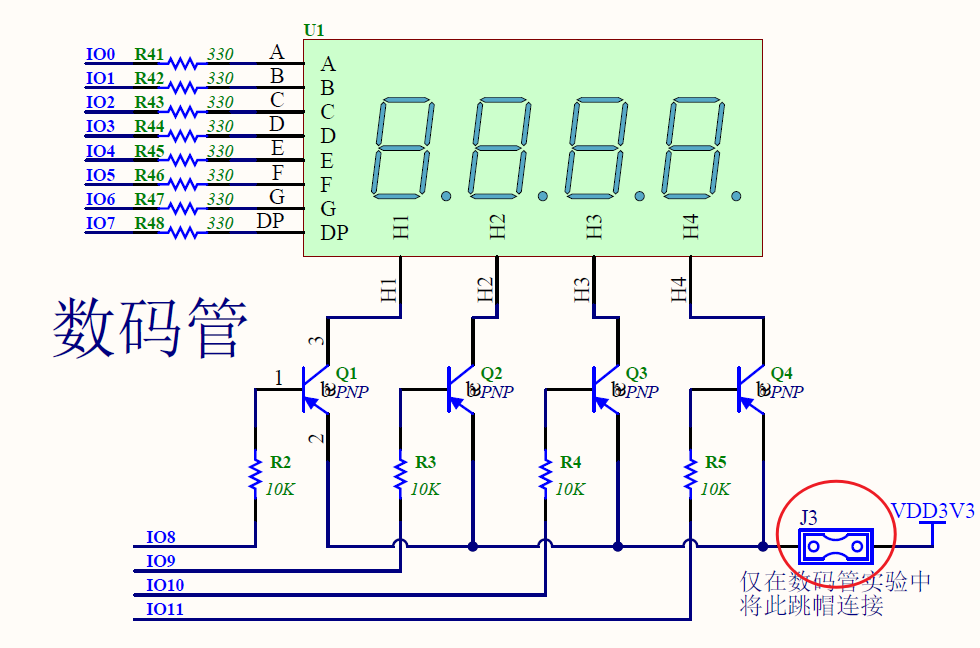


绘图工具：https://www.processon.com/diagrams

1. 电路原理图





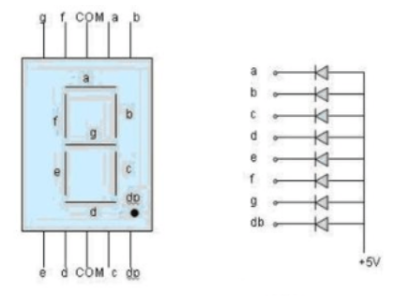


共阳，

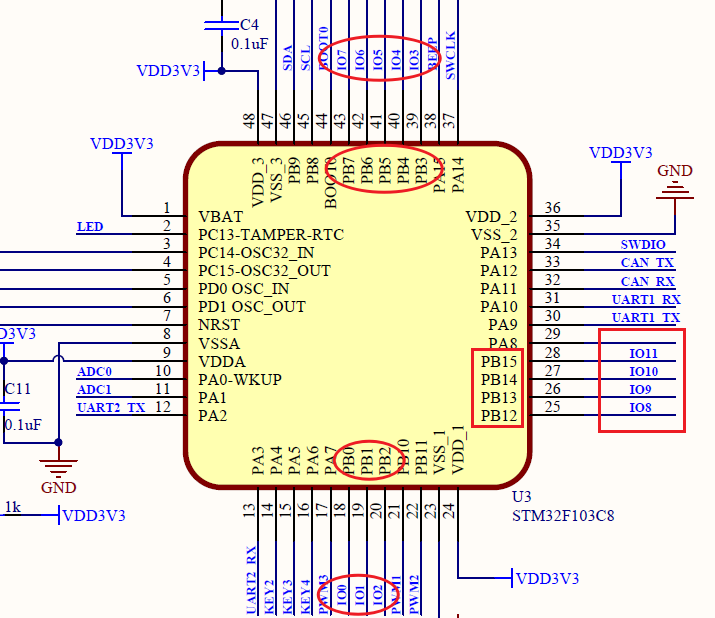
低电平点亮

低电平导通

导通后，高电位

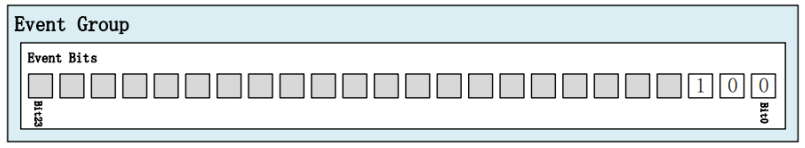


共阳极数码管：八个LED的阳极连在一起。



1. 参考知识及代码：
2. 事件标志组

当configUSE\_16\_BIT\_TICKS 为 0 的时候 TickType\_t 是个 32 位的数据类型，因此 EventBits\_t 也是个 32 位的数据类型。EventBits\_t 类型的变量可以存储 24 个事件位，另外的那高 8 位有其他用。事件位 0 存放在这个变量的 bit0 上，变量的 bit1 就是事件位 1，以此类推。对于 STM32 来说一个事件标志组最多可以存储 24 个事件位，如下图所示：

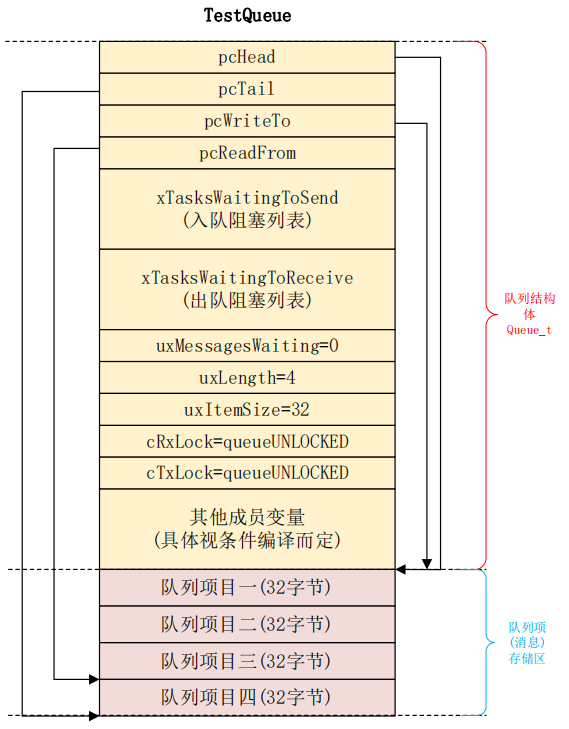


以在任务中可使用的API为例，对部分事件标志组关联API进行说明。

|  |  |  |
| --- | --- | --- |
| **NO** | **API函数名** | **说明** |
| 1 | xEventGroupCreate | 创建一个时间标志组 |
| 2 | xEventGroupClearBits | 将事件标志组中的指定事件位清零 |
| 3 | xEventGroupSetBits | 设置指定的事件位为 1 |
| 4 | xEventGroupWaitBits | 如果任务要等待的事件位还没有准备好(置 1 或清零)的话任务就会进入阻塞态，直到阻塞时间到达或者所等待的事件。 |

1. 队列

比如我们创建一个有 4 个队列项，每个队列项长度为 32 个字节的队列 TestQueue，创建成功的队列如下图所示：



消息指针数组物理结构上仅仅是个数组，可以表现为 FIFO(队列) 或 LIFO(堆栈) 两种形式。

以在任务中可使用的API为例，对部分队列关联API进行说明。

|  |  |  |
| --- | --- | --- |
| **NO** | **API函数名** | **说明** |
| 1 | xQueueCreate | 创建一个队列 |
| 2 | xQueueSend | 向队列中发送消息 |
| 3 | xQueueReceive | 从队列中读取一条消息 |

1. 无源蜂鸣器

有源蜂鸣器内含振荡源，只要一通电就发声，但发生频率固定，音色单一；无源蜂鸣器内部不含振荡源，内部结构相当于电磁场扬声器，可以通过给他输出一定频率的信号才能发声。

人耳能听到的频率范围在20Hz--20kHz之间，通过STM32的GPIO引脚快速切换高低电平输出就能实现无源蜂鸣器的发声，切换的频率不同，发出的音调就不一样。

一段音乐就是不同频率的声音按一定的时间节拍转换发出。所以音乐包含音调和节拍信息。

C调各音符频率如下：

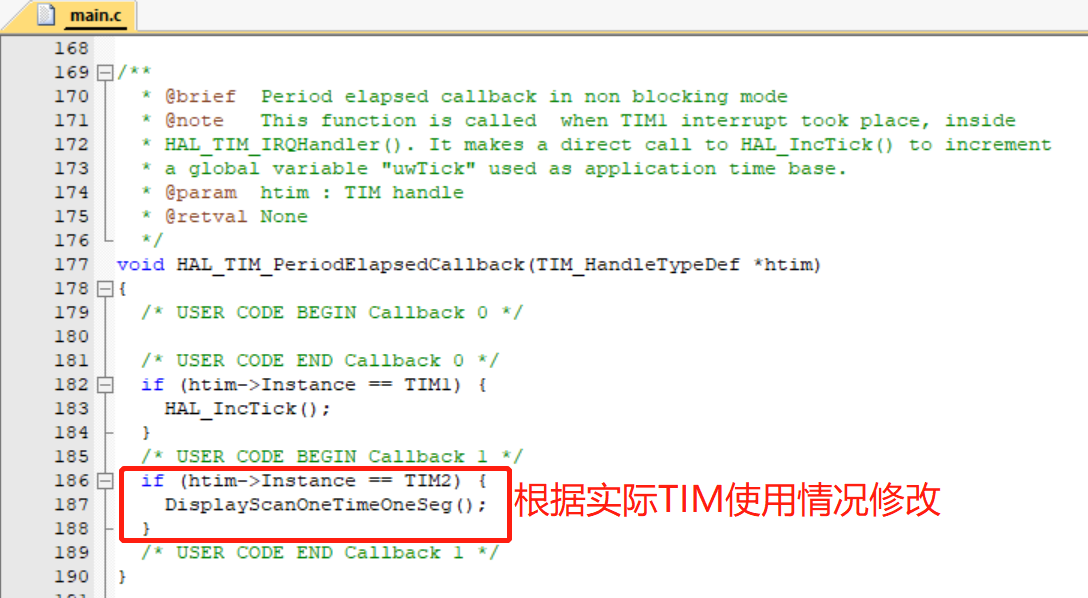


1. 数码管控制代码

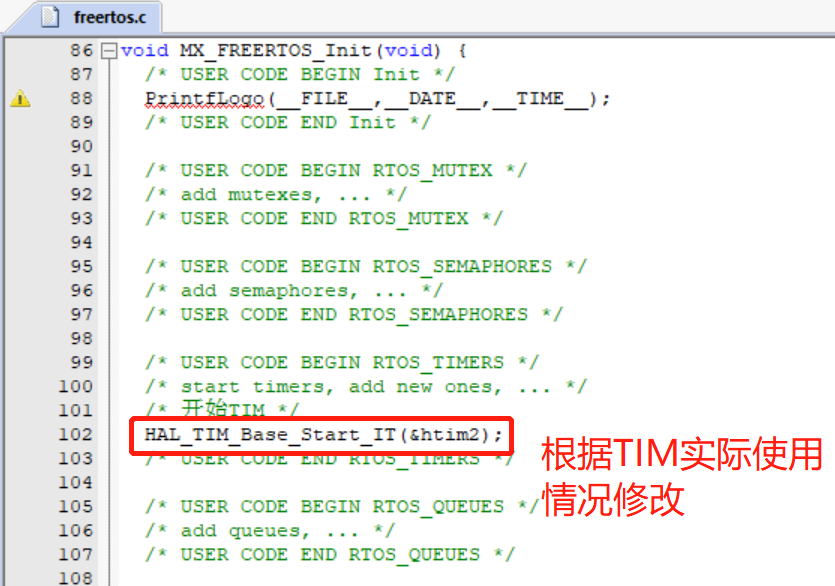
数码管显示文件：

使用方法如下：

1)和TIM结合，刷新数码管显示。



2)开始TIM



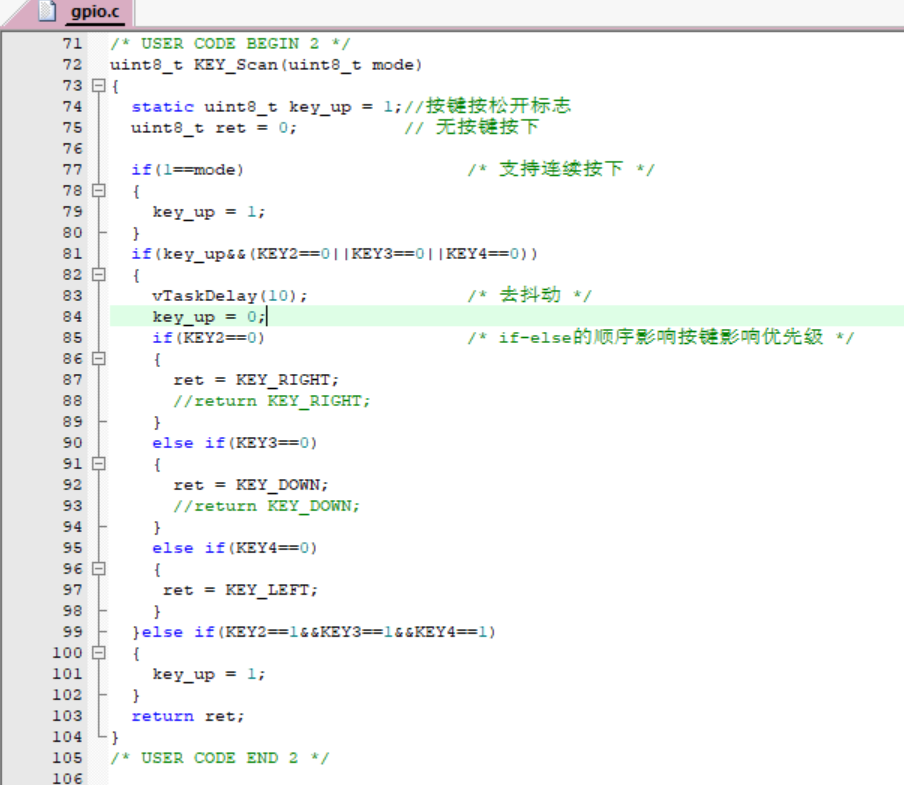
3)显示数据用API如下：

|  |  |  |
| --- | --- | --- |
| **NO.** | **API函数名** | **说明** |
| 1 | LEDDisplayString | 显示指定字符 |
| 2 | LEDDisplay2Half | 显示内容2部分：字符和数字 |
| 3 | LEDDisplayNumber | 显示指定数字 |

1. 按键处理

方式1：

1)按键扫描如下：



2)使用方法如下：

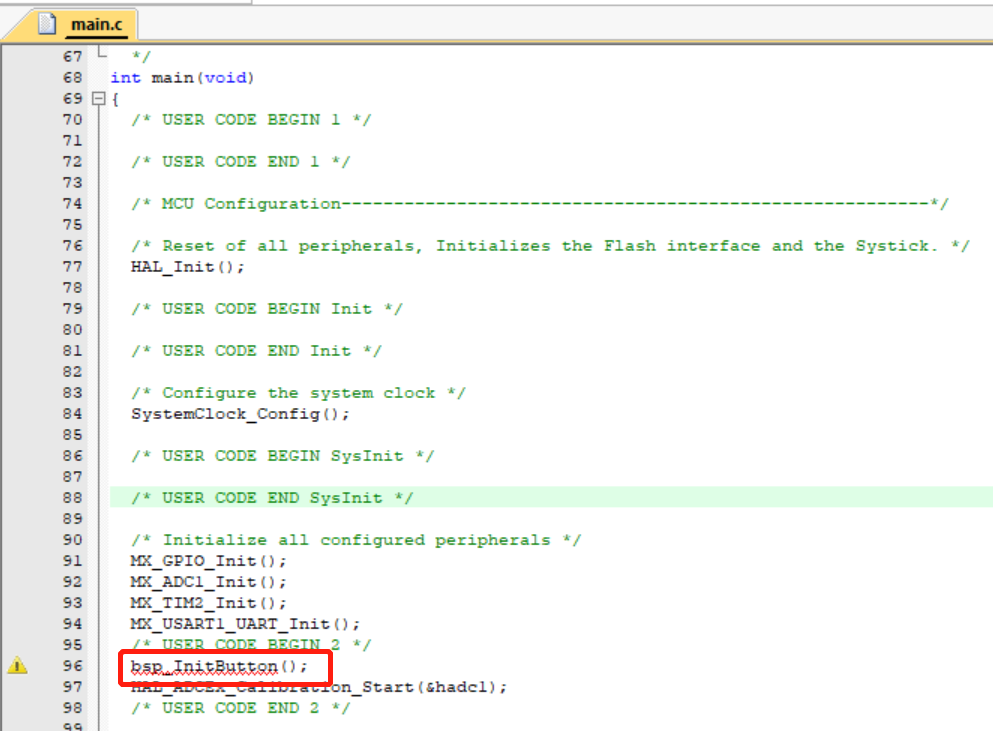


方式2：

按键扫描处理文件：

使用方法如下：

1)按键初期化



2)具体使用方式



六、项目步骤：

1. 正确安装 MDK 软件

2. 运行FreeRTOS 模板程序

3. 创建任务，编译下载至开发板观察实验结果。

七、供参考程序：

1、事件标志组：见教材 P281 页

2、消息队列：见教材 P185页

3、ADC模数转换、按键的检测、数码管显示以及蜂鸣器鸣响的驱动代码，见模板程序

八、项目考核：

满分：5分

评分标准：

（1）内容完整性：是否能够完成实验要求的所有内容 1分

（2）KT点：是否能够掌握本实验相关的核心KT点 1分

（3）代码：是否能够正确说明本实验相关的核心代码 1分

（4）学习态度：能否独立完成 1分

（5）实验报告：内容完整，报告整洁、逻辑清晰 1分

十、课后任务

将实验结果（程序截图、串口输出Log截图）撰写到电子版实验报告上面