

4.按键控制LED跑马灯

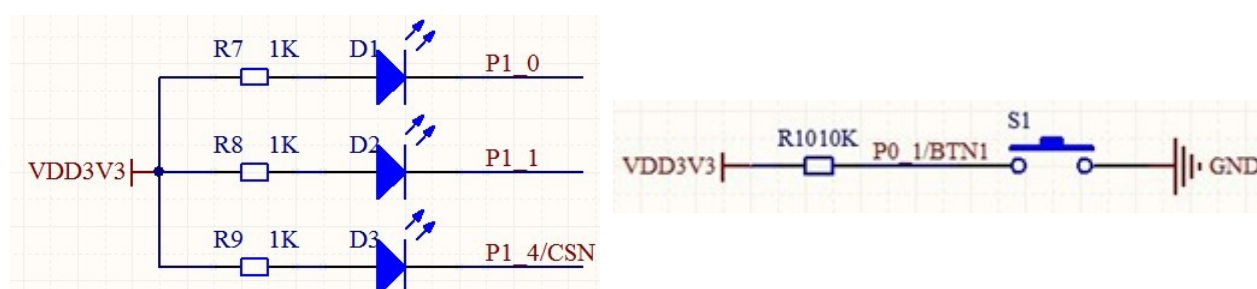
1. 实验目的

- 1) 通过实验掌握 CC2530 芯片 GPIO 的配置方法
- 2) 掌握 Led 驱动电路及开关 Led 的原理
- 3) 掌握按键的使用，实现人机交互

2. 实验设备

- 硬件：PC 机 一台
EB2530（底板、核心板、仿真器、USB线） 一套
- 软件：2000/XP/win7 系统，IAR 8.10 集成开发环境

3. 实验相关电路图



由于发光二极管单向导电特性，即只有在正向电压（二极管的正极接正，负极接负）下才能导通发光。所以 P1.0、P1.1、P1.4 引脚输出低电平 LED 亮，引脚输出高电平 LED 熄灭。当 P0_1 引脚为低电平时说明按键被按下，高电平时为抬起状态。

4. 实验相关寄存器

操作GPIO口需要了解的寄存器，P0、P1相同以下只列出P1的寄存器。如下表所示（更详细的介绍请参考“EB2530\相关资料与软件\cc2530-datasheet.pdf, 英文不好的可参考CC2530中文数据手册完全版.pdf，翻译有250页并非官方中文，有个别地方有错，请以英文为主哦”）：

寄存器	作用	描述
P1 (0x90)	端口 1	端口 1。通用 I / O 端口。可以从 SFR 位寻址。
P1SEL (0xF4)	端口 1 功能选择	P1.7 到 P0.0 功能选择 0: 通用 I / O 1: 外设功能
P1DIR (0xFE)	端口 1 方向	P1.7 到 P1.0 的 I/O 方向 0: 输入 1: 输出
P1INP (0xF6)	端口 1 输入模式	P1.7 到 P1.2 的 I/O 输入模式。由于 P1.0 和 P1.1 没有上拉/下拉功能，P1INP 暂时不需要配置，了解一下为后面的实验打下基础 0: 上拉/下拉(见 P2INP (0xF7) - 端口 2 输入模式) 1: 三态

按照表格寄存器的内容，对 P1.0、P1.1、P1.4 口进行配置，简化配置指令如下：

```
P1DIR |= 0x13; // P1.0、P1.1、P1.4 定义为输出
```

按键 S1 配置如下：

```
POSEL &= ~0x01; //设置 P0.1 为普通 IO 口
PODIR &= ~0x01; //按键接在 P0.1 口上，设 P0.1 为输入模式
P0INP &= ~0x01; //打开 P0.1 上拉电阻
```

5. 源码分析

```
/*
 * 文件 名: main.c
 * 作    者: Andy
 * 修    订: 2013-01-08
 * 版    本: 1.0
 * 描    述: 按下按键 S1 控制 LED1. LED2. LED3 实现跑马灯效果
 */
#include <ioCC2530.h>

typedef unsigned char uchar;
typedef unsigned int  uint;

#define LED1 P1_0      // 定义 P1.0 口为 LED1 控制端
#define LED2 P1_1      // 定义 P1.1 口为 LED2 控制端
#define LED3 P1_4      // 定义 P1.4 口为 LED3 控制端
#define KEY1 P0_1      // 定义 P0.1 口为 S1 控制端
#define ON      0
#define OFF     1

/*
 * 名    称: DelayMS()
 * 功    能: 以毫秒为单位延时，系统时钟不配置时默认为 16M(用示波器测量相当精确)
 * 入口参数: msec 延时参数，值越大，延时越久
 * 出口参数: 无
 */
void DelayMS(uint msec)
{
    uint i, j;

    for (i=0; i<msec; i++)
        for (j=0; j<535; j++);
}
```

```

}

/*****
* 名    称: LedOnOrOff()
* 功    能: 点亮或熄灭所有 LED 灯
* 入口参数: mode 为 0 时 LED 灯亮  mode 为 1 时 LED 灯灭
* 出口参数: 无
*****/
void LedOnOrOff(uchar mode)
{
    LED1 = mode;
    LED2 = mode;
    LED3 = mode; //由于 P1.4 与仿真器共用, 必须拔掉仿真器的插头才能看到 LED3 的变化
}

/*****
* 名    称: InitLed()
* 功    能: 设置 LED 相应的 IO 口
* 入口参数: 无
* 出口参数: 无
*****/
void InitLed(void)
{
    P1DIR |= 0x13;  // P1.0、P1.1、P1.4 定义为输出
    LedOnOrOff(1);  // 使所有 LED 灯默认为熄灭状态
}

/*****
* 名    称: InitKey()
* 功    能: 设置按键相应的 IO 口
* 入口参数: 无
* 出口参数: 无
*****/
void InitKey(void)
{
    POSEL &= ~0x02;    //设置 P0.1 为普通 IO 口
    PODIR &= ~0x02;    //按键接在 P0.1 口上, 设 P0.1 为输入模式
    POINP &= ~0x02;    //打开 P0.1 上拉电阻

```

```

}

/*****
* 名称: KeyScan()
* 功能: 读取按键状态
* 入口参数: 无
* 出口参数: 0 为抬起 1 为按键按下
*****/
uchar KeyScan(void)
{
    if (KEY1 == 0)
    {
        DelayMS(10); //延时 10MS 去抖
        if (KEY1 == 0)
        {
            while(!KEY1); //松手检测
            return 1; //有按键按下
        }
    }

    return 0; //无按键按下
}

/*****
* 程序入口函数
*****/
void main(void)
{
    InitLed(); //设置 LED 灯相应的 IO 口
    InitKey(); //设置按键 S1 相应的 IO 口

    while(1)
    {
        if (KeyScan()) //扫描按键当前状态, 按下时执行跑马灯效果
        {
            LED1 = ON; //点亮 LED1
            DelayMS(500);
            LED1 = OFF; //熄灭 LED1

```

```
        LED2 = ON;  
        DelayMS(500);  
        LED2 = OFF;  
        LED3 = ON;  
        DelayMS(500);  
        LED3 = OFF;  
    }  
}  
}
```

6. 实验步骤

编译下载程序后按下按键 S1 控制 LED1.LED2.LED3 实现跑马灯效果