

实验二 基于 S3C2410 的基本接口实验

一、实验目的

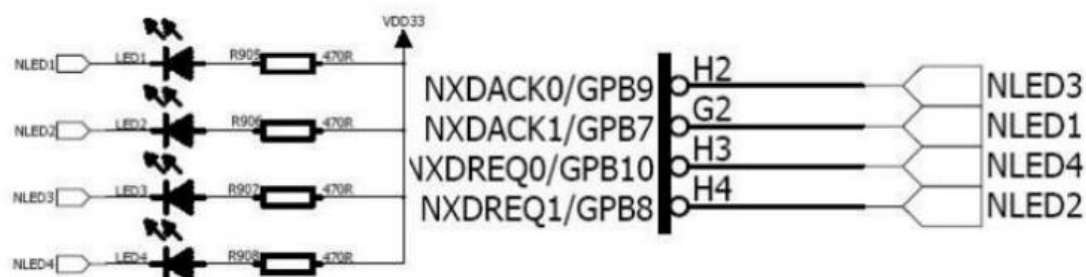
1. 掌握 S3C2410X 芯片的 I/O 控制寄存器的配置；
2. 通过实验掌握 ARM 芯片使用 I/O 口控制 LED 显示；
3. 了解 ARM 芯片中复用 I/O 口的使用方法。

二、实验内容

1. 编写程序，控制实验平台的发光二极管 LED1,LED2,LED3,LED4，使它们有规律的点亮和熄灭；
2. 使用实验箱上的用户键盘，允许接受键盘中断；
3. 利用键盘控制发光二极管的工作模式，按照不同模式点亮。

三、实验原理

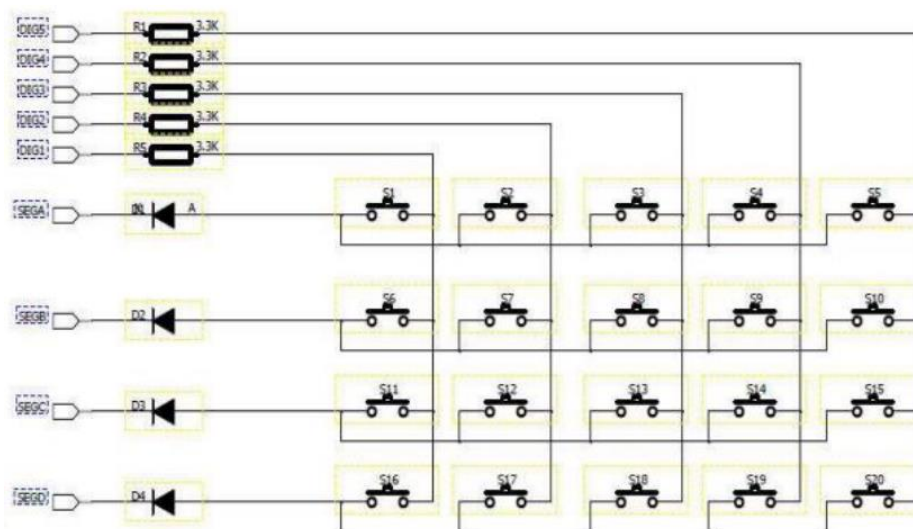
1. LED 基本连接电路



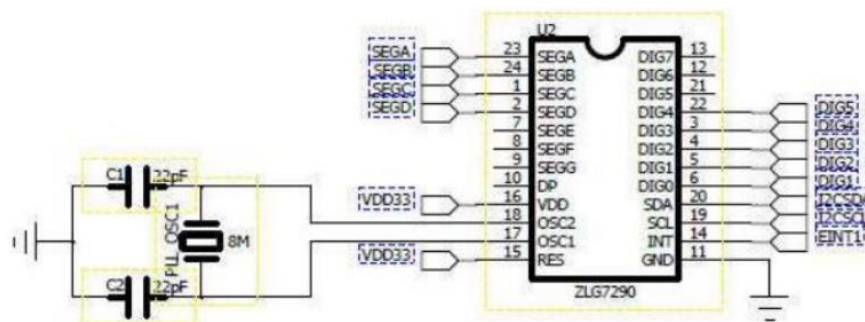
本实验中发光二极管的电路连接模式如上图所示，通过实验手册可知本次实验的发光二极管连接采用共阳极接法，即要求另外一段为低电平二极管才能够导通，因此得出本实验中二极管的工作电平为低电平点亮，高电平熄灭。

2. 用户键盘基本连接电路

➤ 键盘连接电路



➤ 键盘控制电路



键盘动作由芯片 ZLG7290 检测，当键盘按下时，芯片检测到后在 INT 引脚产生中断触发电平通知处理器，EINT1 为处理器 S3C2410 的 I/O 引脚(可软件配置成中断功能)；处理器通过 IIC 总线读取芯片 ZLG7290 键值寄存器(01H)中保存的键值。通过这种工作原理就可以编写键盘中断程序，获取键盘的读入字符，以此控制发光二极管的工作模式。

四、 实验步骤

1. 验证示例源码

拷贝整个实验例程源码目录到本地磁盘自己的工作目录下；

- 使用 μVision IDE for ARM 通过 ULINK2 仿真器连接实验板。
- 打开实验例程目录 02_led_test 子目录下的 led_test.Uv2 例程，编译链接工程；
- 调试程序，观察运行结果；
- 打开实验例程目录 12_KeyBoardTest 子目录下的 KeyBoardTest.Uv2 例程，编译链接工程
- 调试程序，观察运行结果。

2. 设计实现自己的 I/O 控制程序

拷贝示例实验源码工程；根据实验原理和实验要求设计程序，实现使用键盘控制发光二极管按照不同模式点亮。

五、 实验结果

具体实验结果已经通过实验进行验收，故在此不进行实验箱结果的展示。

六、 程序说明

在本次实验中，结合上述实验原理，基于示例程序我编写了主要的核心代码，其中包括自主设计的 LED 工作模式函数，现就核心代码进行说明：

- LED 工作模式说明：

LED 工作函数	LED 工作模式
led_on_1 ()	一个灯亮
led_on_2 ()	二个灯亮
led_on_3 ()	三个灯亮
led_on_4 ()	四个灯亮
led_on_off()	全部点亮再熄灭
led_on_toRight()	从左向右依次点亮
led_off_toLeft()	从右向左依次熄灭

➤ 键盘控制 LED 工作程序说明：

键盘中断输入	程序中断处理程序说明
键盘输入 1	执行 led_on_1 (), 一个灯亮
键盘输入 2	执行 led_on_2 (), 二个灯亮
键盘输入 3	执行 led_on_3 (), 三个灯亮
键盘输入 4	执行 led_on_4 (), 四个灯亮
键盘输入 5	执行 led_on_off(), 全部点亮熄灭
键盘输入 6	执行 led_on_toRight(), 依次点亮
键盘输入 7	执行 led_off_toLeft(), 依次熄灭

➤ 程序源代码：

```

1. void led_on_1(void)
2. {
3.     int i, nOut;
4.     nOut = 0x0780;
5.     rGPBDAT = nOut & 0x0700;
6.     for(i=0;i<100000;i++);
7. }
8.
9. void led_on_2(void)
10. {
11.     int i, nOut;
12.     nOut = 0x0780;
13.     rGPBDAT = nOut & 0x0600;
14.     for(i=0;i<100000;i++);
15. }
16.
17. void led_on_3(void)
18. {
19.     int i, nOut;
20.     nOut = 0x0780;
21.     rGPBDAT = nOut & 0x0400;
22.     for(i=0;i<100000;i++);
23. }
24. void led_on_4(void)
25. {
26.     int i, nOut;
27.     nOut = 0x0780;
28.     rGPBDAT = nOut & 0x0000;
29.     for(i=0;i<100000;i++);
30. }
31. void led_on_off(void)
32. {
33.     int i;
34.     rGPBDAT=0;

```

```

35.     for(i=0;i<100000;i++);
36.     rGPBDAT=0x0780;
37.     for(i=0;i<100000;i++);
38. }
39. void led_on_toRight(void)
40. {
41.     int i,nOut;
42.     nOut=0;
43.     rGPBDAT = 0;
44.     for(i=0;i<100000;i++);
45.     rGPBDAT = nOut | 0x0380;
46.     for(i=0;i<100000;i++);
47.     rGPBDAT |= nOut | 0x0180;
48.     for(i=0;i<100000;i++);
49.     rGPBDAT |= nOut | 0x0080;
50.     for(i=0;i<100000;i++);
51.     rGPBDAT |= nOut | 0x0000;
52.     for(i=0;i<100000;i++);
53. }
54. void led_off_toLeft(void)
55. {
56.     int i,nOut;
57.     nOut=0;
58.     rGPBDAT = 0;
59.     for(i=0;i<100000;i++);
60.     rGPBDAT = nOut | 0x0000;
61.     for(i=0;i<100000;i++);
62.     rGPBDAT |= nOut | 0x0080;
63.     for(i=0;i<100000;i++);
64.     rGPBDAT |= nOut | 0x0180;
65.     for(i=0;i<100000;i++);
66.     rGPBDAT |= nOut | 0x0380;
67.     for(i=0;i<100000;i++);
68. }

```

七、 心得体会

在这次实验中，我通过学习示例工程中的源代码并结合实验手册，出色地完成了实验内容，这次实验经历让我对 Led 工作原理以及键盘中断控制有了很深刻的理解。通过编写代码实现与外设进行 I/O 交互，我也增加了对计算机与外界进行信息传输的理解，在编程能力上也有了很大的提高，这些宝贵的经验也为后面实验打下了坚实的基础。