

实验 4 GPIO 输出实验

一、实验目的

1. 能够在 PROTEUS 仿真软件运行程序(无操作系统)。
2. 熟悉 LPC2000 系列 ARM7 微控制器的 GPIO 控制。

二、实验设备

硬件：PC 机、EasyARM2103

软件：WindowsXP 系统，PROTEUS 集成开发环境，KEIL C 软件

三、实验内容

控制 ARM2103 开发板上的 LED 闪烁，形成 LED1---LED8 轮流点亮的流水灯效果。

四、实验预习要求

1. 仔细阅读《ARM 嵌入式系统基础教程》第 4 章 4.8 小节的内容，熟悉 GPIO 的设置。
2. 仔细阅读《EasyARM2103》第 2 章的内容，了解 EasyARM2103 的硬件结构，注意 LED 灯的相关控制电路。
3. 了解 PROTEUS 开发环境及应用。

五、实验原理

1. 设置 PINSEL0、IODIR、IOSET、IOCLR、IOPIN 寄存器。
2. 每盏 LED 灯的亮灭时间由软件延时程序确定。
3. 流水灯效果可以通过数组人为定义，也可以通过一定的算法计算。

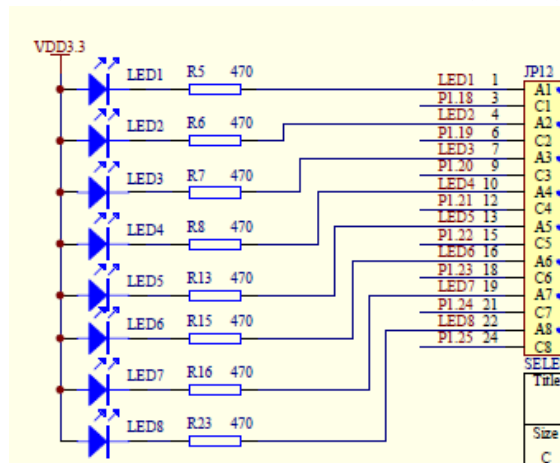


图 1 共阳极 LED 灯接口

六、实验步骤

1. 用工程模板建立一个工程 GPIO_DISPLAY.PRJ。参考《Proteus 仿真 ARM7 实验手册.pdf》
2. 使用 PROTEUS 画出电路图。ARM 处理器的 P0[7:0]连接 LED8~LED1。
3. 在 KEIL 软件中编写 display.C 程序。参考实验例子编写花样流水灯程序。
4. 交叉编译并仿真调试该工程。



七、实验参考例程

选择低速 GPIO，控制 LED 灯闪烁，示例程序如程序清单 1 所示。

程序清单 1 GPIO 控制 LED 闪烁

```
#include "config.h"

/* LED8~LED1 8 个 LED 分别由 P1.25~P1.18 控制 */
const uint32 LEDS8 = (0xFF << 18); // P1[25:18]控制 LED8~LED1，低电平点亮

/*****
** 函数名称 : DelayNS()
** 函数功能 : 长软件延时
** 入口参数 : dly 延时参数，值越大，延时越久
** 出口参数 : 无
*****/

void DelayNS (uint32 dly)
{
    uint32 i;

    for ( ; dly>0; dly--)
        for (i=0; i<50000; i++);
}

/*****
** 函数名称 : main()
** 函数功能 : LED 流水灯显示实验。
** 调试说明 :
*****/

int main (void)
{
    uint8 i;

    //PINSEL1 = 0x00000000; // 设置管脚连接 GPIO
    PINSEL2 = PINSEL2 & (~0x08); // P1[25:16]连接 GPIO
    IO1DIR = LEDS8; // 设置 LED1 控制口为输出

    while (1)
    {
        for (i=0; i<8; i++)
        { /* 流水灯花样显示 */
            IO1SET = ~(LEDS8 & (7 << (18+i)));
            DelayNS(20);
            IO1CLR = LEDS8 & (7 << (18+i));
        }
    }
}
```



```
        DelayNS(20);
    }
}

return 0;
}

/*****
**                                     End Of File
*****/
```

八、思考题

1. 修改流水灯为 LED 数码管，如何在数码管上显示数字“6”？

2. ◎ARM 开发板上的 8 路 LED 分别可选择 P0.0-P0.7 进行控制，试画出 LED 共阳极电路的接线图，并写出控制程序，实现流水灯花样显示，间隔时间为 1 秒。先全部熄灭→再全部点亮→依次逐个点亮→依次逐个叠加→依次逐个递减→两个靠拢后分开→从两边叠加后递减→最后全部熄灭。如此循环。【提示：使用查表法】

