

实验3 基于 Proteus 及 Keil 的 ARM 仿真实验

一、实验目的

1. 能够在 PROTEUS 仿真软件运行第一个程序(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 机一台

软件: WindowsXP 系统, PROTEUS 环境

三、实验内容

控制 ARM2103 开发板上的 LED 灯每秒闪烁一次。

四、实验预习要求

1. 仔细阅读《ARM 嵌入式系统基础教程(第2版)》第4章4.8小节的内容,熟悉 GPIO 的设置。
2. 仔细阅读《EasyARM2103.PDF》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 PROTESU 开发环境及应用。

五、实验步骤

1. 如何在 EasyARM2103 上运行第一个程序。参考《Proteus 仿真 ARM7 实验手册.pdf》
2. 了解 PROTEUS 及 KEIL 软件。
3. 用工程模板建立第一个工程。使用 PROTEUS 画出电路图(如图1),使用 KEIL 编写程序(程序清单3)。
4. 仿真调试第一个工程

六、电路图

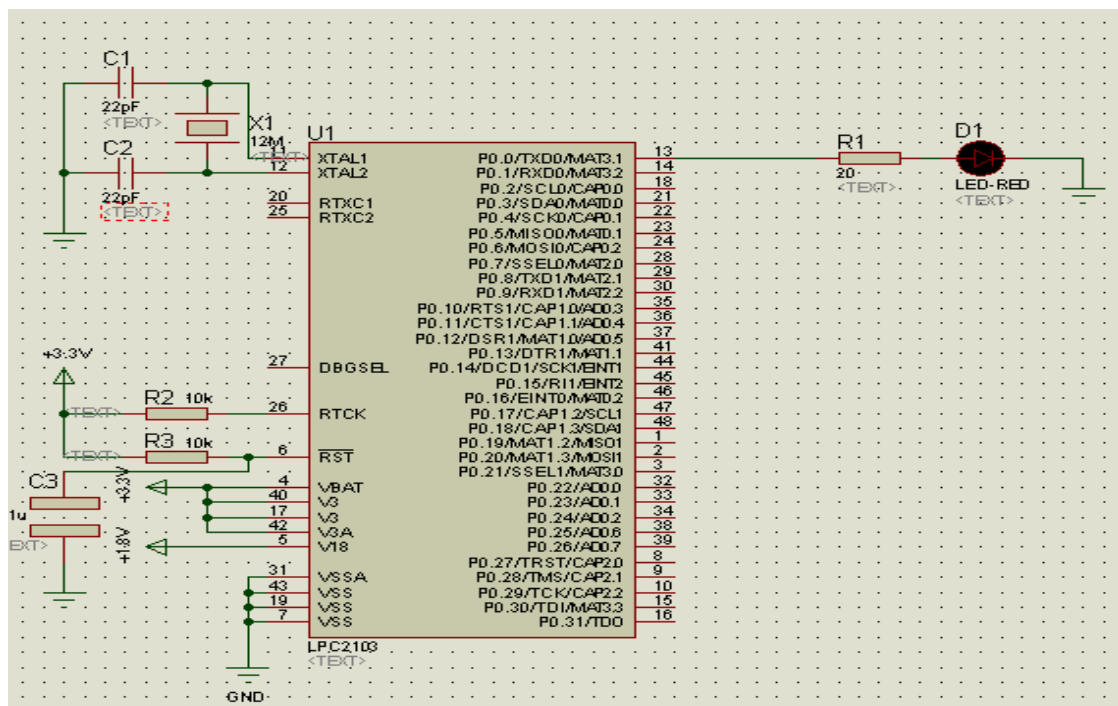


图1 LPC2103 与 LED 灯接口电路图



七、实验参考例子。

选择低速 GPIO，控制 LED 灯闪烁，示例程序如程序清单 3 所示。

程序清单 3 GPIO 控制 LED 闪烁

```
#include "config.h"
#define LED1 1 << 0 /* P0.0 控制 LED1 */
/*****
** 函数名称: DelayNS
** 功能描述: 延时函数
** 入口参数: uiDly 值越大，延时时间越长
** 出口参数: 无
*****/
void DelayNS (uint32 uiDly)
{
    uint32 i;
    for (; uiDly > 0; uiDly--){
        for(i = 0; i < 50000; i++);
    }
/*****
** 函数名称: main
** 功能描述: LED1 闪烁
** 入口参数: 无
** 出口参数: 无
*****/
int main (void)
{
    PINSEL0 = 0x00; /* 将 P0.0 设置为 GPIO */
    IO0DIR = LED1; /* 设置 LED 控制口为输出 */
    IO0SET = LED1; /* LED1 熄灭 */
    while (1) {
        IO0SET = LED1; /* LED1 熄灭 */
        DelayNS(50); /* 延时 */
        IO0CLR = LED1; /* LED1 点亮 */
        DelayNS(50); /* 延时 */
    }
    return 0;
}
```

八、思考题

1. 在实验参考程序中，如何控制 LED 灯亮灭的频率？
2. 修改 LED 灯为共阳极接法，限流电阻阻值有什么变化，显示效果如何？

