

实验 1 - GPIO 输出控制 LED

1. 实验目的

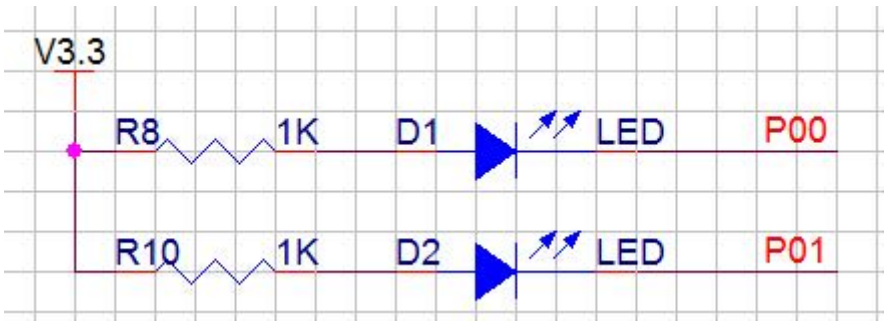
掌握 NRF24LE1 的 GPIO 的配置方式和输出控制。

2. 实验内容

使用 NRF24LE1 的 GPIO 输出控制 LED 的亮灭。  
程序运行后，开发板上的 D1 指示灯以 200ms 的间隔闪烁。

3. 实验原理

3.1. 电路原理



开发板上配置的两个用户指示灯 D1、D2，分别有 GPIO P0.0 和 P0.1 控制，当 GPIO 输出高电平时，LED 两端电压相等，LED 上没有电流流过，LED 处于灭状态，当 GPIO 输出低电平时，LED 两端存在正向压差，电流流过 LED，LED 被点亮。

3.2. GPIO 配置

NRF24LE1 的 GPIO 通过 2 个寄存器来配置：PxDIR 和 PxCON(更详细的内容请查阅 NRF24LE1 数据手册)。

- PxDIR：设置 IO 的方向。
- PxCON：设置 IO 的功能。

表 1：P0DIR 寄存器（地址：0x93，复位值：0xFF）

位	名称	R/W	功能
7~0	方向	R/W	P0.0~P0.7 方向位。输出：dir=0，输入：dir=1。 P0DIR 0 – P0.0 P0DIR 1 – P0.1 P0DIR 2 – P0.2 P0DIR 3 – P0.3 P0DIR 4 – P0.4

			P0DIR 5 – P0.5 P0DIR 6 – P0.6 P0DIR 7 – P0.7
--	--	--	--

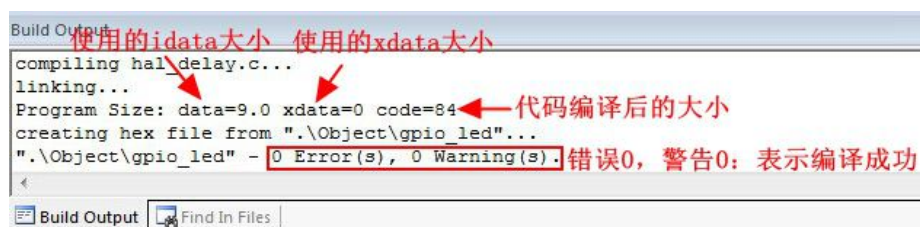
按照上述内容，对 P0.0 进行配置如下：

P0DIR &= ~0x01; //配置 P0.0 为输出

P0CON：采用默认值即可。

## 4. 实验步骤

- 在 Keil uVision4 中打开工程 “gpio\_led.uvproj” 工程；
- 编译工程，注意查看编译输出栏，观察编译的结果，如果有错误，修改程序，直到编译成功为止；

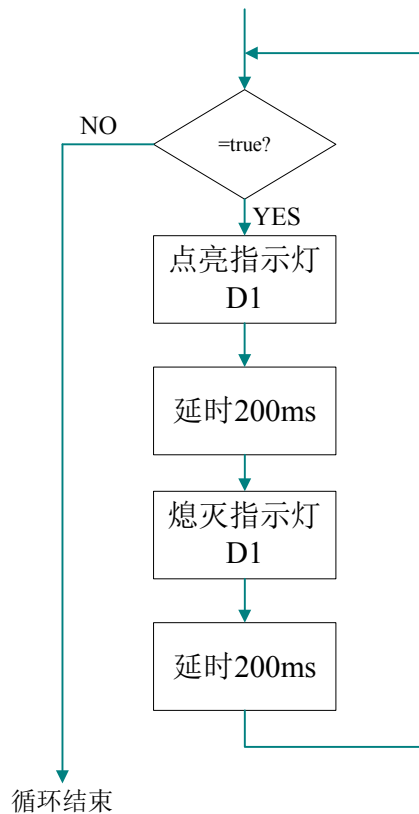


- 将编译生成的 HEX 文件 “key\_led.hex” (该文件位于工程目录下的 “Object” 文件夹中)通过编程器下载到开发板中运行。

## 5. 实验程序

### 5.1. 程序流程

GPIO 输出控制 LED 的程序执行流程如下图所示：



## 5.2. 程序清单

```
#define D1    P00 //开发板上的指示灯 D1
```

```
/******
```

```
*描 述：主函数
```

```
*入 参：无
```

```
*返回值：无
```

```
*****/
```

```
void main(void)
```

```
{
```

```
    P0DIR &= ~0x01;    //配置 P0.0 为输出
```

```
    while(true)
```

```
    {
```

```
        D1 = 0;        //点亮指示灯 D1
```

```
        delay_ms(200); //延时 200ms
```

```
        D1 = 1;        //熄灭指示灯 D1
```

```
        delay_ms(200); //延时 200ms
```

```
    }
```

```
}
```

也可以采用另外一种方式来驱动指示灯的亮灭：

```
while(true)
{
    D1 = ~ D1;      //指示灯 D1 状态取反
    delay_ms(200); //延时 200ms
}
```