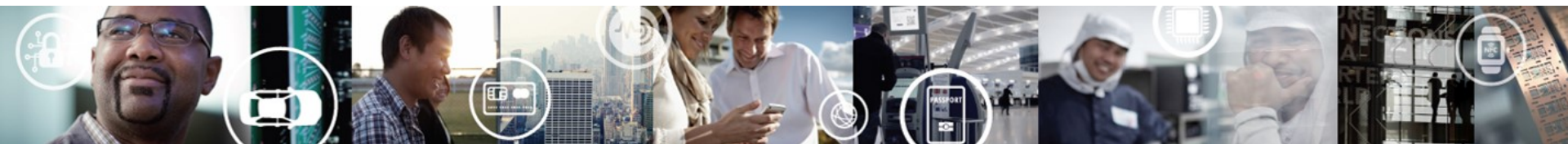


LPC82X 培训资料

通用输入输出接口 动手实验

MAY, 2016



EXTERNAL USE



SECURE CONNECTIONS
FOR A SMARTER WORLD

动手实验1

GPIO口控制LED状态

内容

- 实验简介（目的，内容，结果）
- 软/硬件环境搭建
- 实验步骤
- 相关底层驱动APIs说明

实验简介

- **目的**：通过本实验，理解和掌握LPC82x GPIO关于以下几点：
 - 如何初始化GPIO口
 - 如何设置GPIO口的输入/输出状态
 - 如何控制GPIO的输出电平
- **描述**：本实验实现GPIO口控制LED闪烁状态
- **结果**：板上LED闪烁

软/硬件环境搭建

- 硬件

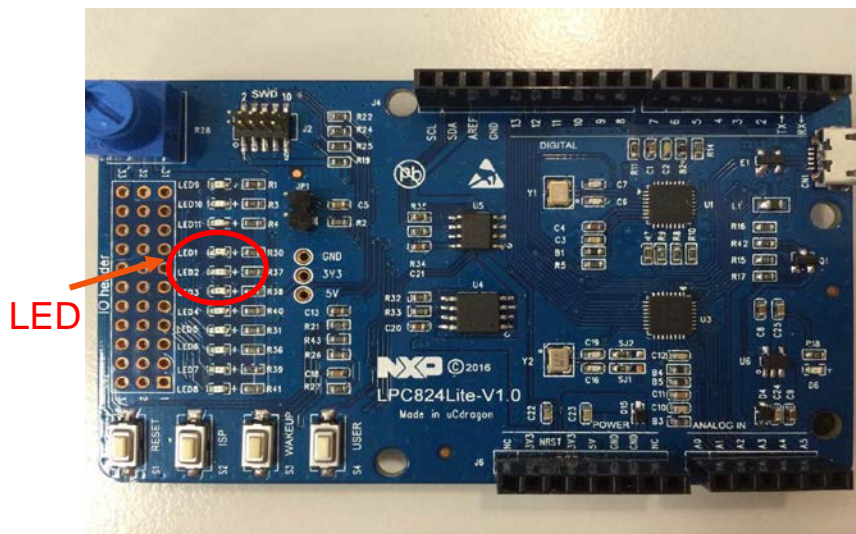
- 评估板：LPC824Lite-V1.0

- 工程位置

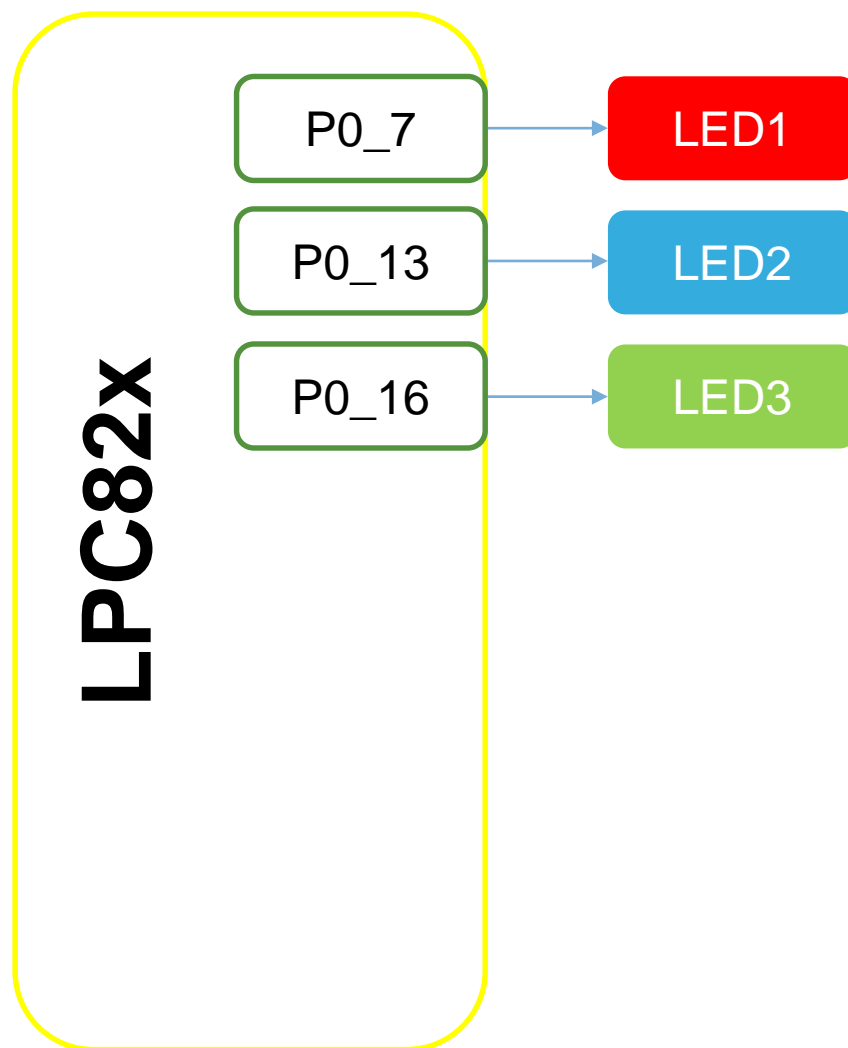
- ..\peri_example\gpio\gpio_hands_on_led\project_gpio_handson1.uvprojx

- 硬件连接

LED1	LED2	LED3
P0_7	P0_13	P0_16



原理图分析



实验步骤

- 第一步 – 打开实验例程工程
- 第二步 – 搭建好硬件环境
- 第三步 – 编译下载程序，运行
 - LED1/2/3灯不断闪烁

相关底层驱动APIs说明 - 1

1.初始化GPIO

*STATIC INLINE void Chip_SPI_Init(LPC_SPI_T *pSPI)*

2.关闭GPIO功能

*STATIC INLINE void Chip_GPIO_DeInit(LPC_GPIO_T *pGPIO)*

3.设置对应port和pin脚的输入输出状态(isOutput, false为输入, true为输出)

*STATIC INLINE void Chip_GPIO_PinSetDIR(LPC_GPIO_T *pGPIO, uint8_t port, uint8_t pin, bool isOutput)*

4. 获取对应port和pin脚的输入输出状态(false为输入 , true为输出)

*STATIC INLINE bool Chip_GPIO_PinGetDIR(LPC_GPIO_T *pGPIO, uint8_t port, uint8_t pin)*

5. 翻转对应port和pin脚的输入输出状态

*STATIC INLINE void Chip_GPIO_PinToggleDIR(LPC_GPIO_T *pGPIO, uint8_t port, uint8_t pin)*

6. 设置对应port和pin脚的输出电平状态setting(false低电平, true高电平)

*STATIC INLINE void Chip_GPIO_PinSetState(LPC_GPIO_T *pGPIO, uint8_t port, uint8_t pin, bool setting)*

相关底层驱动APIs说明 - 2

7. 获得对应port和pin脚上的电平状态(false为低电平, true为高电平)

```
STATIC INLINE bool Chip_GPIO_PinGetState(LPC_GPIO_T *pGPIO, uint8_t port, uint8_t pin)
```

8. 翻转对应port和pin脚上的电平状态

```
STATIC INLINE void Chip_GPIO_PinToggleState(LPC_GPIO_T *pGPIO, uint8_t port, uint8_t pin)
```

9. 设置一组port管脚的输入输出状态(isOutput, false为输入, true为输出)

```
STATIC INLINE void Chip_GPIO_PortSetDIR(LPC_GPIO_T *pGPIO, uint8_t port, uint8_t pinMask, bool isOutput)
```

10. 获取一组port输入输出状态

```
STATIC INLINE uint32_t Chip_GPIO_PortGetDIR(LPC_GPIO_T *pGPIO, uint8_t port)
```

11. 翻转一组port的输入输出状态

```
STATIC INLINE void Chip_GPIO_PortToggleDIR(LPC_GPIO_T *pGPIO, uint8_t port, uint8_t pinMask)
```

12. 根据value参数设置对应一组port的输出电平状态

```
STATIC INLINE void Chip_GPIO_PortSetState(LPC_GPIO_T *pGPIO, uint8_t port, uint32_t value)
```

相关底层驱动APIs说明 - 3

13.根据bmPins设置对应一组port输出 高电平

```
STATIC INLINE void Chip_GPIO_PortSetOutHigh(LPC_GPIO_T *pGPIO, uint8_t port, uint32_t bmPins)
```

14.根据bmPins设置对应一组port输出 低电平

```
STATIC INLINE void Chip_GPIO_PortSetOutLow(LPC_GPIO_T *pGPIO, uint8_t port, uint32_t bmPins)
```

15.根据mask设置一组port管脚的屏蔽功能

```
STATIC INLINE void Chip_GPIO_PortSetMask(LPC_GPIO_T *pGPIO, uint8_t port, uint32_t mask)
```

16. 获取一组port管脚的屏蔽设置

```
STATIC INLINE uint32_t Chip_GPIO_PortGetMask(LPC_GPIO_T *pGPIO, uint8_t port)
```

动手实验2

获取GPIO口状态

内容

- 实验简介（目的，内容，结果）
- 软/硬件环境搭建
- 实验步骤

实验简介

- **目的：**通过本实验，理解和掌握LPC82x GPIO接口关于以下几点：
 - GPIO初始化
 - GPIO 输入/输出方向设置
 - 同时设置按键和LED的GPIO，利用按键控制LED
- **描述：**本实验实现通过获取按键状态控制LED的功能
- **结果：**
 - 当按键SW1按下时，LED灯点亮
 - 当按键SW1松开时，LED灯熄灭

软/硬件环境搭建

- 硬件

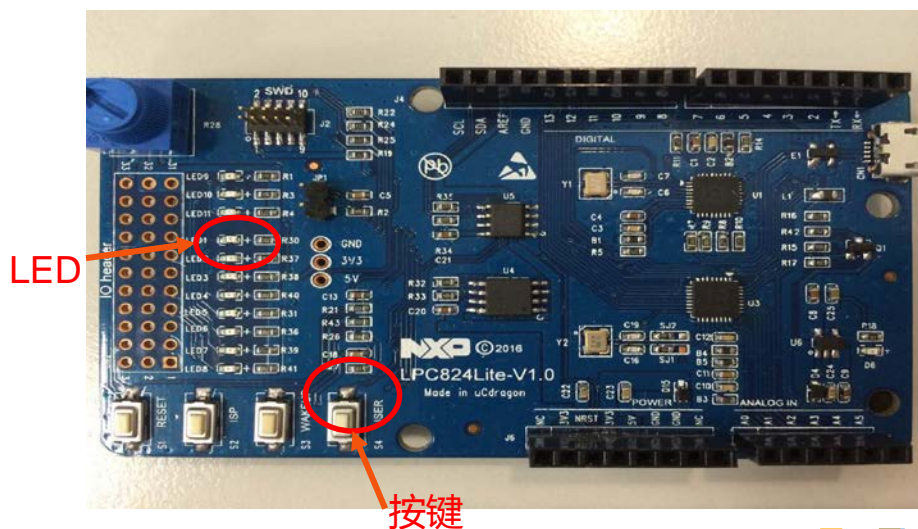
- 评估板：LPC824Lite-V1.0

- 工程位置

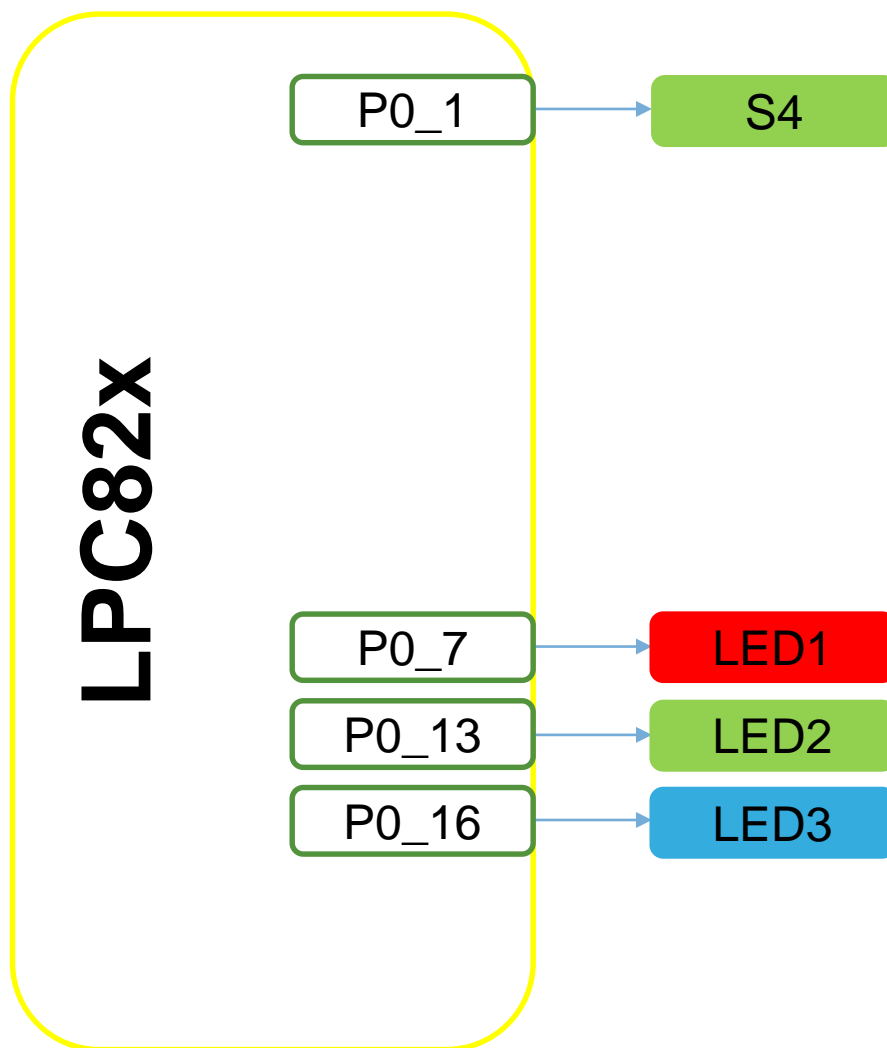
- ..\peri_example\gpio\gpio_hands_on_key\project_gpio_handson2.uvprojx

- 硬件连接

LED1	LED2	LED3	S4
P0_12	P0_16	P0_17	P0_1



原理图分析



实验步骤

- 第一步 – 打开实验例程工程，动手添加代码
- 第二步 – 搭建好硬件环境
- 第三步 – 编译下载程序，运行
 - 当按键S4按下时，LED1灯点亮
 - 当按键S4松开时，LED1灯熄灭



SECURE CONNECTIONS
FOR A SMARTER WORLD