Lab 09

R. Ferrero, A. C. Marceddu

Politecnico di Torino

Dipartimento di Automatica e Informatica (DAUIN)

Torino - Italy

Exercise 1

- Add to led.h file the prototype:
  `void led4and11_On(void);`
- Add to ‘led’ group the file funct_led.c
- Implement in funct_led.c the function `led4and11_On(void)`, powering on the LEDs 4 and 11 acting on the FIOSET register.
- Note: the state (on/off) of the other LEDs must not be modified.
- Test the function calling it from the main.

Exercise 2

- Add to led.h file the prototype:
  `void led4_Off(void);`
- Implement in funct_led.c the function `led4_Off(void)`, switching off LED 4 acting on FIOCLR register.
- Note: the state (on/off) of the other LEDs must not be modified.
- Test the function calling it from the main.

Exercise 3

- Add to led.h file the prototype:
  `void ledEvenOn_OddOff(void);`
- Implement in funct_led.c the function `ledEvenOn_OddOff(void)`, powering on the LEDs with even index number and powering off odd ones, acting on FIOPIN register.
- Test the function calling it from the main.

Exercise 4

- Add to led.h file the prototype:
  `void LED_On(unsigned int num);`
- Implement in funct_led.c the function `void LED_On(unsigned int num)` powering on the LED corresponding to the parameter passed:
  - num = 0 -> LED 4
  - num = 1 -> LED 5
  - num = 7 -> LED 11
- Test the function calling it from the main.

Exercise 5

- Add to led.h file the prototype:
  `void LED_Off(unsigned int num);`
- Implement in funct_led.c the function `void LED_Off(unsigned int num)` powering off the LED corresponding to the parameter passed: num = 0 -> LED 4
  - num = 1 -> LED 5
  - num = 7 -> LED 11
- Test the function calling it from the main.

Exercise 6

- In the `main`, before entering the endless loop, power on LED 8 using `LED_On`.
- By pressing button KEY1, power off the current LED and power on the LED on the left (when arrived to LED 4, jump to LED 11).
- By pressing button KEY2, power off the current LED and power on the LED on the right (when arrived to LED 11, jump to LED 4).
- By pressing button INT0, get back to original configuration, with LED 8 on.

What LED is on?

- To know which LED is on you can:
  - Read content of `LPC_GPIO2->FIOPIN`
  - Read content of `LPC_GPIO2->FIOSET`
  - *define* a global variable in funct_led.c:

```
unsigned int led_value;
```

`led_value` stores the on LED.

In the other files you can access the variable *declaring*:

```
extern unsigned int led_value;
```