

## **INTRO: Development board, IDE, GPIO ports, registers**

Configure the ŠARM development board so that:

- pins P0\_4, P0\_5, P0\_6 and P0\_7 are output (LEDs L0-L3)
- pins P0\_12, P0\_13, P0\_14 and P0\_15 are input (keys T0-T3)

Write a program that turns on/off the LEDs when the corresponding key is pressed/released.

### **Pin configuration:**

GPIO module is described in chapters 7 and 8 of the LPC213x user manual (UM10120.pdf).

GPIO registers for this exercise (all registers are 32 bit long):

PINSEL0: controls the pin functions for pins P0.0...P0.15

PINSEL1: controls the pin functions for pins P0.16...P0.31

IO0DIR: controls the GPIO (general purpose input output) pin direction

IO0SET: writing 1 to bit X will set the output pin P0.X high (writing 0 does nothing)

IO0CLR: writing 1 to bit X will set the output pin P0.X low (writing 0 does nothing)

IO0PIN: reg. hold the state of all GPIO pins

When changing certain bits of the register, all the other bits must be left unchanged.

PINSEL0: xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

- bit0 and bit1 : configure the P0\_0 pin function
- bit2 and bit3 : configure the P0\_1 pin function
- ...
- bit30 and bit31 : configure the P0\_15 pin function

Similar is the PINSEL1 register for pins P0\_16 - P0\_31.

To configure a pin as a GPIO (general purpose input output) pin, write bit combination '00' to the corresponding 2 bits in PINSEL0 (PINSEL1) register. For example: to configure P0\_4 pin as GPIO:

PINSEL0: xxxx xxxx xxxx xxxx xxxx xx00 xxxx xxxx

To configure the direction of a GPIO pin as input/output, write 0/1 to the corresponding bit in IO0DIR register. For example: to configure P0\_4 as output (for LED0):

```
IO0DIR: xxxx xxxx xxxx xxxx xxx0 xxxx xxx1 xxxx
```

To read the state of the i-th GPIO pin, read the state of the i-th bit of the IO0PIN register. For example: to read the state of the pin P0\_12 (key T0, active 0):

```
IO0PIN: xxxx xxxx xxxx xxxx xxx0 xxxx xxxx xxxx :T0 is pressed
IO0PIN: xxxx xxxx xxxx xxxx xxx1 xxxx xxxx xxxx :T0 is released
```

To write 1/0 to the i-th pin of port 0 (P0\_i), write 1 to the i-th bit of the IO0SET/IO0CLR register:

```
e.g. to turn on LED0 (on P0_4):
    IO0SET = 0x00000010
e.g. to turn off LED0 (on P0_4):
    IO0CLR = 0x00000010
```

### **GPIO interface**

Write the functions for GPIO pins:

```
void set_pin_direction( int inputMask, int outputMask ); // configure GPIO pins
void set_pin_value( int pinNum, int value);             // write value to the pin
int get_pin_value( int pinNum );                         // read the value
```

inputMask and outputMask are bit masks for input and output GPIO pins. For example:

```
int inputMask = 0x0000F000; // pins P0_12-P0_15 are input (keys)
int outputMask = 0x000000F0; // pins P0_4-P0_7 are output (LEDs)
```

### **Main program**

Using the GPIO functions, write the program that will continuously read the state of the keys and turn on/off the LEDs whenever the corresponding key is pressed/released.