Instituto Tecnológico de Estudios Superiores de Monterrey

Laboratorio de microcontroladores

Práctica 5: Parallel Ports (GPIOs)

Integrantes:

Gabriela Natalia Altamirano Cruz | A01411942

Rodrigo Elizondo Arevalo | A00821566

Código

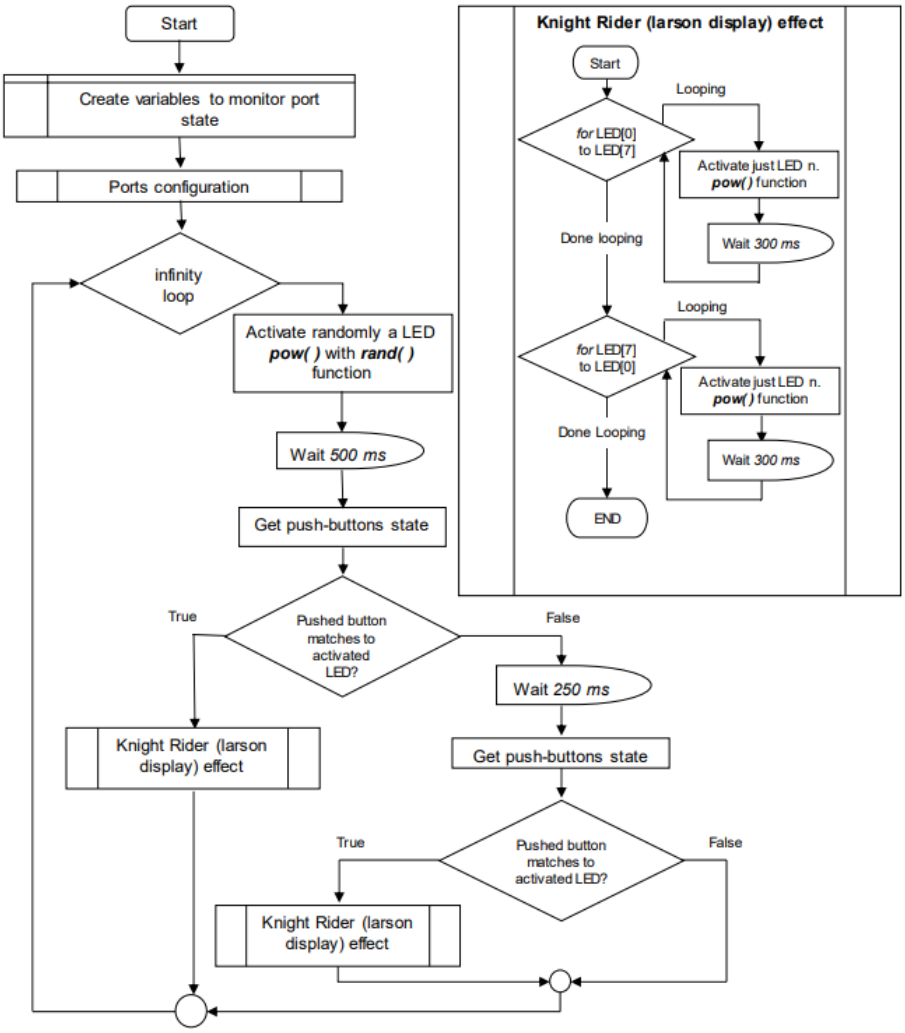

Figure 1. Flow diagram of the Whack-a-Mole game

//Libraries/headers

```
#include "device_config.h"
#include <math.h>

//Directive section

#define _XTAL_FREQ 1000000
#define DELAY_SWEEP 300
```

//Data type section

```
enum exponent{ bbase = 2, limit = 8 };
enum por_dir{ output, input };
```

```
enum por_ACDC { digital, analog };
enum resistor_state { set_ON, res_ON };
enum led_state { led_OFF, led_ON };
enum button_state { pushed, no_pushed };
```

//function declaration/section

```
void portsInit( void ) {
        ANSELB = digital;
        TRISB  = 0x00;
        ANSELD = digital;
        TRISD  = 0xFF;

}

void __interrupt(high_priority)
high_isr(void){

…
}
```

//Main section

```
void main( void ){
   portsInit();
   char molePosition, buttonStatus = 0;
   while(1){
      unsigned char num = rand() % 8 ;
      switch(num) {
         case 0:
            molePosition = 0x01;
            break;
         case 1:
            molePosition = 0x02;
            break;
         case 2:
            molePosition = 0x04;
            break;
         case 3:
            molePosition = 0x08;
            break;
         case 4:
            molePosition = 0x10;
            break;
```

```
            case 5:
                molePosition = 0x20;
                break;
            case 6:
                molePosition = 0x40;
                break;
            default:
                molePosition = 0x80;
                break;
        }

        LATA = molePosition;
        __delay_ms(1000);
        buttonStatus = ~PORTD;
        if (molePosition == buttonStatus)
        {
            for (unsigned char i = 0; i < 7; i++){
                LATA = (unsigned char) pow(2, i);
                __delay_ms(DELAY);
            }
            for (unsigned char i = 7; i > 0; i--){
                LATA = (unsigned char) pow(2, i);
                __delay_ms(DELAY);
            }
            LATA = 0x01;
            __delay_ms(DELAY);
        }
    }
}
```