



Aluna: Tatielen Rodrigues Dutra Pereira

Matricula: 12/0136074

Data: 21/11/2017

Para todas as questões, utilize os LEDs e/ou os botões da placa Launchpad do MSP430.

1. Escreva um código em C que pisca os LEDs ininterruptamente.

```
#include <msp430g2553.h>
volatile int i;
int main(void)
{
    WDTCTL = WDTPW | WDTHOLD; // Stop watchdog timer
    /*Inicializar as saídas em zero*/
    P1DIR = 0x41; /* P1DIR - Quando em 0 é entrada - Quando em 1 é saída
        PIDIR = 01000001
        */
    P1OUT = 0x00;

    //volatile int i;
    for(;;)
    {
        P1OUT^=0x41; // P1OUT = 01000001
        for(i=0;i<0x6000;i++);
    }
}
```

2. Escreva um código em C que pisca os LEDs ininterruptamente. No ciclo que pisca os LEDs, o tempo que os LEDs ficam ligados deve ser duas vezes maior do que o tempo que eles ficam desligados.

```
#include<msp430g2553.h>
volatile unsigned int i;
int main(void)
{
    WDTCTL = WDTPW | WDTHOLD;
    P1OUT = 0;
    P1DIR |= 0x41;
    for(;;)
    {
        P1OUT ^= 0x41;
        for(i=0;i<60000;i++);
        P1OUT ^= 0x41;
    }
}
```



```
for(i=0;i<30000;i++);  
}  
}
```

3. Escreva um código em C que acende os LEDs quando o botão é pressionado.

```
//Ligar LEDS enquanto o  
//botão estiver pressionado  
#include <msp430g2553.h>  
#define BTN BIT3  
#define LED1 BIT0  
#define LED2 BIT6  
int main(void){  
    WDTCTL = WDTPW | WDTHOLD;  
    P1OUT = 0;  
    P1REN |= BTN;  
    P1DIR |= LED1 + LED2;  
    P1OUT |= BTN;  
    for(;;)  
    {  
        if((P1IN & BTN) == 0)  
        {  
            P1OUT |= LED1 + LED2;  
        }  
        else  
        {  
            P1OUT &= ~(LED1 + LED2);  
        }  
    }  
}
```

4. Escreva um código em C que pisca os LEDs ininterruptamente somente se o botão for pressionado.

```
#include <msp430g2553.h>  
#define BTN BIT3  
#define LED1 BIT0  
#define LED2 BIT6  
  
int main( void )  
{  
    volatile unsigned int i;  
    // Stop watchdog timer to prevent time out reset  
    WDTCTL = WDTPW + WDTHOLD;
```



```
P1OUT = 0;
P1REN |= BTN;
P1DIR |= LED1 + LED2;
P1OUT = BTN;
for(;;)
{
    if((P1IN & BTN) == 0)
    {
        P1OUT ^= LED1 + LED2;
    }
    else
    {
        P1OUT &= ~(LED1 + LED2);
    }
    for(i=0; i<30000; i++);
}
return 0;
}
```

5. Escreva um código em C que acende os LEDs quando o botão é pressionado. Deixe o MSP430 em modo de baixo consumo, e habilite a interrupção do botão.

```
#include <msp430g2553.h>
#include <intrinsics.h>
#define BTN BIT3
#define LED1 BIT0
#define LED2 BIT6

void main( void )
{
    // Stop watchdog timer to prevent time out reset
    WDTCTL = WDTPW + WDTHOLD;
    P1OUT |= LED1 + LED2;
    P1DIR |= LED1 + LED2;
    P1DIR &= ~BTN;
    P1REN |= BTN;
    P1OUT |= BTN;
    P1IES |= BTN;
    P1IE |= BTN;
    for(;;)
    {
        _BIS_SR(GIE + LPM4_bits);
    }
}
```

```
}
```

```
#pragma vector = PORT1_VECTOR  
__interrupt void Port_1(void)  
{  
    while((P1IN&BTN)==0)  
    {  
        P1OUT |= LED1 + LED2;  
    }  
    P1OUT &= ~(LED1 + LED2);  
    P1IFG &= ~BTN;  
}
```