

TAREFA 1 – Microcontrolador PIC

PROGRAMAÇÃO GERAL E COM INTERRUPTO EXTERNA

Analise os códigos a seguir (todos com algum tipo de bug) e identifique o problema que o impede de rodar como esperado. Faça a correção adequada.

1. A Figura 1 deve ser usada para resolver os itens (a), (b) e (c).

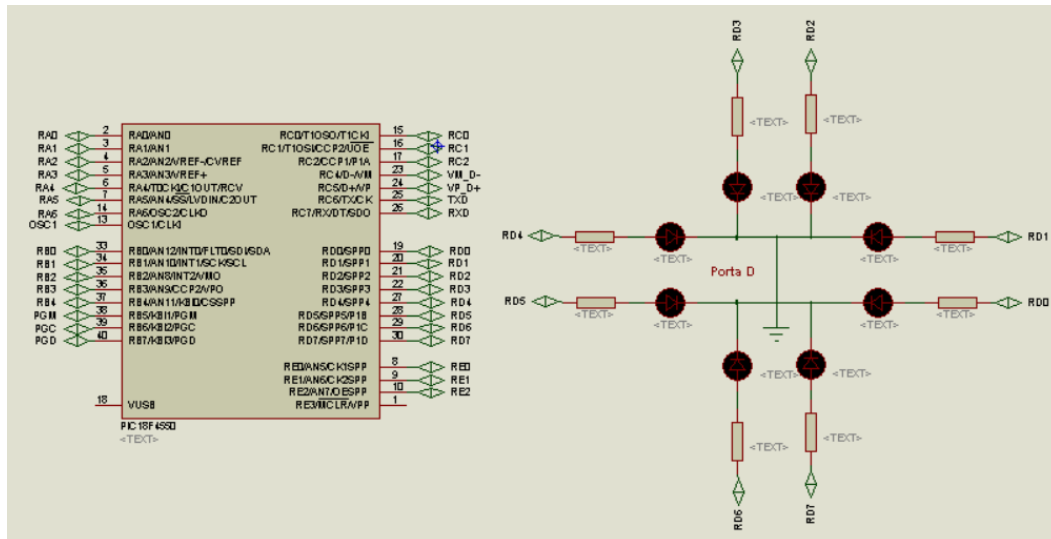


Figura 1: Rotação de Leds na Porta D

- (a) No código a seguir o comportamento esperado é que os Leds rotacionem no modo bate-e-volta. Ao iniciar o programa, os Leds começam a rotacionar normalmente, mas, depois de várias rotações, o processo volta para o primeiro Led.

```
#define _XTAL_FREQ 4000000
#pragma config FOSC = INTOSC_EC
#pragma config WDT = ON
#pragma config MCLRE = OFF

void main(void) {
    FOSC = 0b01100000;
    TRISD = 0b00000000;
    LATD = 1;
    while(1) {
        while(LATD != 0b00000001) {
            LATD = (LATD >> 1 | LATD << 7);
            __delay_ms(100);
        }
        while(LATD != 0b10000000) {
            LATD = (LATD << 1 | LATD >> 7);
            __delay_ms(100);
        }
    }
}
```

- (b) No código a seguir o comportamento esperado é que os Leds rotacionem no modo bate-e-volta. Ao iniciar o programa, nada acontece.

```
#define _XTAL_FREQ 4000000
#pragma config FOSC = INTOSC_EC
#pragma config WDT = OFF
#pragma config MCLRE = ON

void main(void) {
    FOSC = 0b01100000;
    TRISD = 0b00000000;
    LATD = 1;
    while(1) {
        while(LATD != 0b00000001) {
            LATD = (LATD >> 1 | LATD << 7);
            __delay_ms(100);
        }
        while(LATD != 0b10000000) {
            LATD = (LATD << 1 | LATD >> 7);
            __delay_ms(100);
        }
    }
}
```

- (c) No código a seguir o comportamento esperado é que os Leds rotacionem no modo bate-e-volta. Ao iniciar o programa, nada acontece. A tensão nos pinos de saída dos LEDs não mostram nenhum valor bem definido.

```
#define _XTAL_FREQ 4000000
#pragma config FOSC = INTOSC_EC
#pragma config WDT = OFF
#pragma config MCLRE = OFF

void main(void) {
    FOSC = 0b01100000;
    LATD = 1;
    while(1) {
        while(LATD != 0b00000001) {
            LATD = (LATD >> 1 | LATD << 7);
            __delay_ms(100);
        }
        while(LATD != 0b10000000) {
            LATD = (LATD << 1 | LATD >> 7);
            __delay_ms(100);
        }
    }
}
```

2. A Figura 2 deve ser usada para resolver os itens (d) e (e).

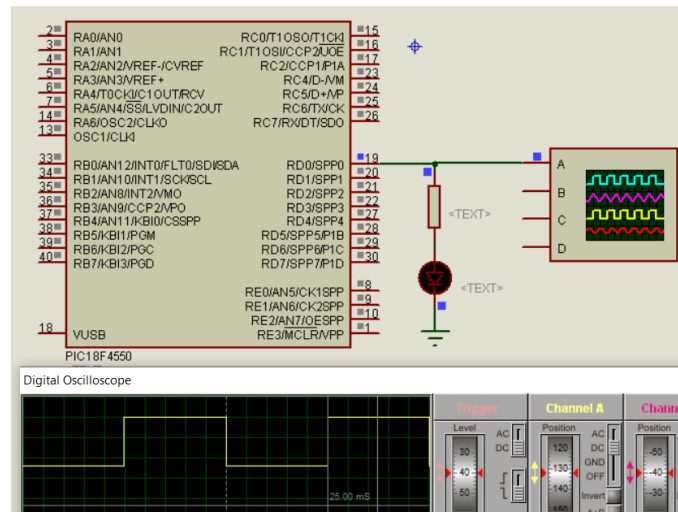


Figura 2: Led do pino RD0 piscando

- (d) No código a seguir o comportamento esperado é que o Led do pino RD0 pisque a cada 100 ms. No entanto, o comportamento observado no osciloscópio é o Led piscando a cada 25 ms.

```
#define _XTAL_FREQ 1000000
#include <xc.h>
#pragma config FOSC = INTOSC_EC
#pragma config WDT = OFF
#pragma config MCLRE = OFF

void main(void) {
    OSCCON = 0b01100000;
    TRISDbits.TRISD0 = 0b00000000;

    while(1) {
        LATDbits.LATD0 = !LATDbits.LATD0;
        __delay_ms(100);
    }
}
```

- (e) No código a seguir o comportamento esperado é que o Led do pino RD0 pisque a cada 100 ms. No entanto, o comportamento observado é o Led ligado o tempo todo.

```
#define _XTAL_FREQ 1000000
#include <xc.h>
#pragma config FOSC = INTOSC_EC
#pragma config WDT = OFF
#pragma config MCLRE = OFF

void main(void) {
    OSCCON = 0x01100000;
    TRISDbits.TRISD0 = 0b00000000;

    while(1) {
        LATDbits.LATD0 = !LATDbits.LATD0;
        __delay_ms(100);
    }
}
```

3. Na Figura 3 o comportamento esperado é o Led no pino RD0 ligar/desligar a cada vez que o botão em RB0 for pressionado. No entanto, nada acontece ao se pressionar o botão em RB0.

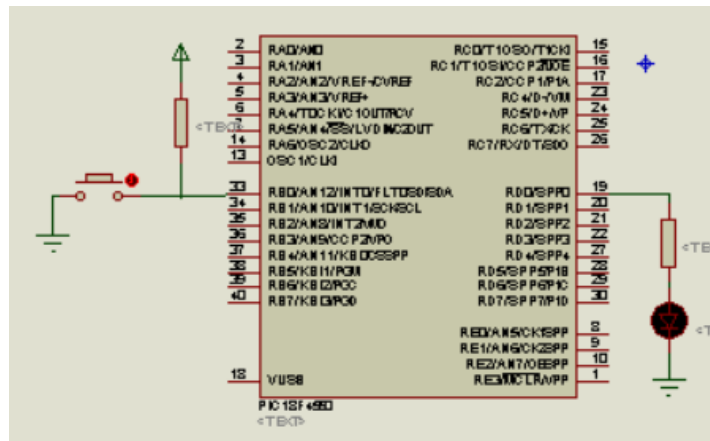


Figura 3: Led acionado através da interrupção 0

```
#define _XTAL_FREQ 4000000
#include <xc.h>

#pragma config FOSC = INTOSC_EC
#pragma config WDT = OFF
#pragma config MCLRE = OFF

void setupInt(void) {
    GIE = 1;
    INTOIE = 1;
    INTOF = 0;
    INTEDG0 = 1;
}

void interrupt interrupcao(void) {
    if (INT0F) {
        LATDbits.LD0 = !LATDbits.LD0;
        INTOF = 0;
    }
}

void main(void) {
    OSCCON = 0b01100000;
    TRISD = 0x00;
    TRISB = 0x01;
    setupInt();

    while(1) {
    }
}
```

4. Na Figura 4 o comportamento esperado é uma mensagem de “Display LCD!” na primeira linha do display e uma contagem crescente na segunda linha; e um LED piscando no pino RD0 da Porta D. No entanto, o comportamento verificado é o LED piscando e nenhuma mensagem no LCD.

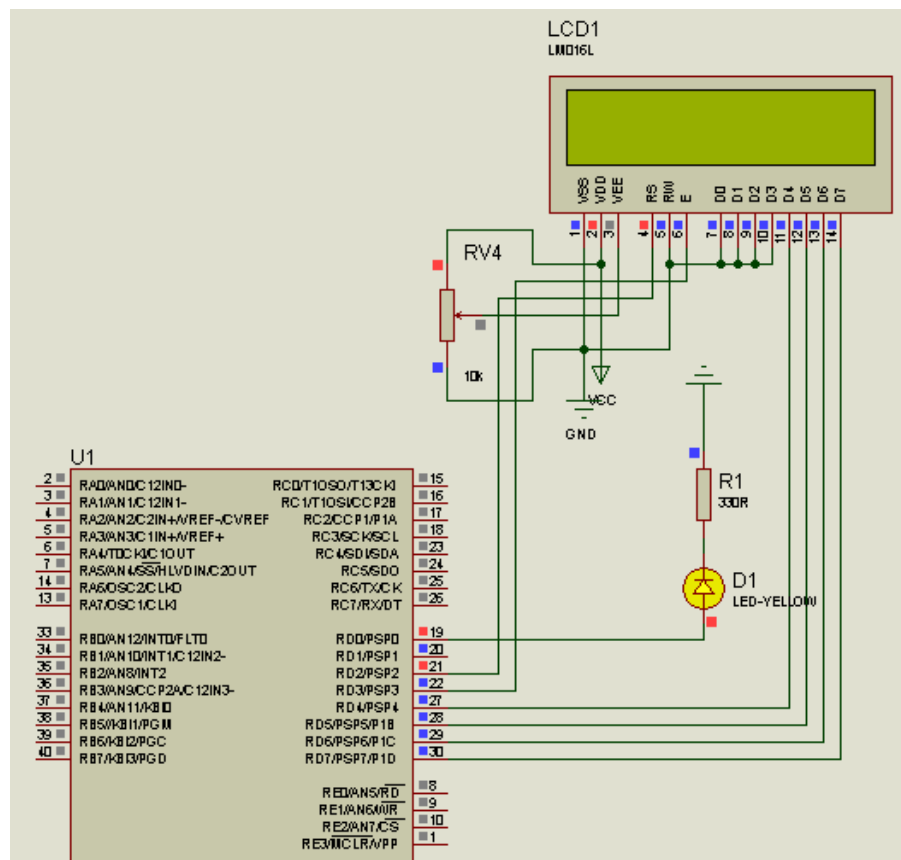


Figura 4: LED piscando e uso do LCD

```
#define _XTAL_FREQ 4000000
```

```
#include <xc.h>
```

```
#define RS LATD2
```

```
#define EN LATD3
```

```
#define D4 LATD4
```

```
#define D5 LATD5
```

```
#define D6 LATD6
```

```
#define D7 LATD7
```

```
#pragma config FOSC = INTOSC_EC
```

```
#pragma config WDT = OFF
```

```
#pragma config MCLRE = OFF
```

```
#include "lcd.h"
```

```
#include <stdio.h>
```

```
char linha1[16];
```

```
char linha2[16];
```

```
int contador = 0;
```

```
void main(void) {
    Lcd_Init();
    TRISD = 0;

    sprintf(linha1, "Hello world! ");
    Lcd_Set_Cursor(1,1);
    Lcd_Write_String(linha1);

    while(1) {
        sprintf(linha2, "Contador: %i ",contador);
        contador ++;
        Lcd_Set_Cursor(2,1);
        Lcd_Write_String(linha2);
        LATDbits.LATD0 = !LATDbits.LATD0;
    }
}
```