// Sistemas Embarcados

//Rodrigo Reviglio Weishaupt

//201020656

// funcao utilizada para limpar os caracteres vazios a esquerda do vetor char

// funcao retirada da internet

void TrimLeft(char \*str) {

char count = 1;

while (\*str++ != 0) {

if (\*str != 32) {

\*(str - count--) = \*str;

}

count++;

}

}

// relaciona os pinos do LCD com os pinos do PIC

sbit LCD\_RS at RE2\_bit;

sbit LCD\_EN at RE1\_bit;

sbit LCD\_D4 at RD4\_bit;

sbit LCD\_D5 at RD5\_bit;

sbit LCD\_D6 at RD6\_bit;

sbit LCD\_D7 at RD7\_bit;

// Define a direcao do fluxo de dados

sbit LCD\_RS\_Direction at TRISE2\_bit;

sbit LCD\_EN\_Direction at TRISE1\_bit;

sbit LCD\_D4\_Direction at TRISD4\_bit;

sbit LCD\_D5\_Direction at TRISD5\_bit;

sbit LCD\_D6\_Direction at TRISD6\_bit;

sbit LCD\_D7\_Direction at TRISD7\_bit;

// Fim das conexoes

// declaracao de variaveis

int seg;

int m;

int h;

char txtH[7];

char txtM[7];

char txtS[7];

int flag1;

void main(){

flag1 = -1;

trisb = 7; // Configura todos pinos das portas para digital e

ADCON1 = 0x0F; // desabilita o conversor A/D

// Inicializa o LCD

Lcd\_Init(); //Inicializa o Display

Lcd\_Cmd(\_LCD\_CLEAR); //Limpa o Display

Lcd\_Cmd(\_LCD\_CURSOR\_OFF); //Desabilita o cursor

trisc.rc0 = 0;

portc.rc0 = 0;

// Coloca o simbolo ":" entre as casas

Lcd\_Out(1,3,":");

Lcd\_Out(1,6,":");

while(flag1==-1){ // WHILE que segura o temporizador em modo de configuracao ate que o flag seja estourado

//Loop crescente

for(h=0;h<24;h++){ // For loop para contar as horas

inttostr(h,txtH); // Converte o inteiro hora para string e salva no char relativo a hora

TrimLeft(txtH); // Uso da funcao TrimLeft

Lcd\_Out(1,1,txtH); // Escreve as horas no LCD

for(m=0;m<60;m++){ // For loop para contar os minutos

inttostr(m,txtM); // Converte o inteiro minuto para string e salva no char relativo ao minuto

TrimLeft(txtM); // Uso da funcao TrimLeft

Lcd\_Out(1,4,txtM); // Escreve os minutos no LCD

for(seg=0;seg<60;seg++){ //For loop para contar os segundos

inttostr(seg,txtS); // Converte o inteiro segundo para string e salva no char relativo ao segundo

TrimLeft(txtS); // Uso da funcao TrimLeft

Lcd\_Out(1,7,txtS); // Escreve os segundos no LCD

while (portb.b0 == 1){ // Enquanto Rb0 nao esta acionado, o programa busca a incrementacao de horas e minutos, os segundos

// estao travados e o programa aguarda o disparo do flag que indica que o temporizador foi setado

// No momento em que Rb0 e acionado, os segundos sao incrementados em maneira similar a um relogio

while (portb.b1 == 0){ //WHILE utilizado para somar os minutos

m++;

delay\_ms(500);

inttostr(m,txtM);

TrimLeft(txtM);

Lcd\_Out(1,4,txtM);

// IF utilizado para nao permitir que os minutos passem de 59 durante o ajuste do minutos

// Quando a contagem dos minutos chega a 59, o contador e zerado e o campo dos minutos e limpo

if (m==60){

m = 0;

Lcd\_Out(1,4," ");

Lcd\_Out(1,4,"0");

}

}

while (portb.b2 == 0){ //WHILE utilizado para somar as horas

h++;

delay\_ms(500);

inttostr(h,txtH);

TrimLeft(txtH);

Lcd\_Out(1,1,txtH);

// IF utilizado para nao permitir que as horas passem de 99 durante o ajuste das horas

// Quando a contagem das horas chega a 99, o contador e zerado e o campo das horas e limpo

if (h==99){

h = 0;

Lcd\_Out(1,1," ");

Lcd\_Out(1,1,"0");

}

}

if (portb.b3 == 0){ // IF para acionar o flag que indica que o temporizador esta setado

flag1 = 1;

break; // quebra o WHILE Rb0==1(botao Rb0 nao esta acionado)

}

}

delay\_ms(500);

if (flag1!=-1) // quebra o FOR dos segundos

break;

}

Lcd\_Out(1,8," "); // limpa o campo dos segundos quando o loop dos segundos estoura

if (flag1!=-1) // quebra o FOR dos minutos

break;

}

Lcd\_Out(1,5," "); // limpa o campo dos minutos quando o loop dos minutos estoura

if (flag1!=-1) // quebra o FOR das horas

break;

}

Lcd\_Out(1,2," "); // limpa o campo das horas quando o loop das horas estoura

}

// Loop decrescente, neste momento o temporizador esta configurado

// as variaveis h, m e seg ja estao inicializados com os valor que possuiam no momento que que Rb3 foi pressionado

for(;h>=0;h--){ // For loop para contar as horas

inttostr(h,txtH); // Converte o inteiro hora para string e salva no char relativo a hora

TrimLeft(txtH); // Uso da funcao TrimLeft

Lcd\_Out(1,1,txtH); // Escreve as horas no LCD

for(;m>=0;m--){ // For loop para contar os minutos

inttostr(m,txtM); // Converte o inteiro minuto para string e salva no char relativo ao minuto

TrimLeft(txtM); // Uso da funcao TrimLeft

Lcd\_Out(1,4,txtM); // Escreve os minutos no LCD

for(;seg>=0;seg--){ //For loop para contar os segundos

inttostr(seg,txtS); // Converte o inteiro segundo para string e salva no char relativo ao segundo

TrimLeft(txtS); // Uso da funcao TrimLeft

Lcd\_Out(1,7,txtS); // Escreve os segundos no LCD

// IF para saber quando o botao Rb0 esta apertado

// Se apertado, o delay responsavel por contar os segundo e reduzido,

// logo o tempo passa mais rapido para que o relogio possa ser ajustado

if (portb.b0 == 1)

delay\_ms(1000);

else

delay\_ms(100);

if (seg<11){ // IF utilizado para limpar a segundo digitos dos segundos

Lcd\_Out(1,8," ");

}

}

Lcd\_Out(1,8," "); // limpa o campo dos segundos quando o loop dos segundos estoura

seg = 59;

if (m<11){ // IF utilizado para limpar a segundo digitos dos minutos

Lcd\_Out(1,5," ");

}

}

Lcd\_Out(1,5," "); // limpa o campo dos minutos quando o loop dos minutos estoura

m = 59;

if (h<11){ // IF utilizado para limpar a segundo digitos das horas

Lcd\_Out(1,2," ");

}

}

Lcd\_Out(1,2," "); // limpa o campo das horas quando o loop das horas estoura

// neste momento, apos os tres loops forem executados, temos certeza que as variaveis h, m e seg estao zeradas, logo o rele sera ativado

portc.rc0 = 1; // ativacao do rele

delay\_ms(5000);

portc.rc0 = 0; //desativacao do rele

} /// fim do programa