

```
1: /*****
2: **                                     Exercício 03                                     *
3: **                                     *                                     *
4: **      Exercício realizado no laboratório com a finalidade                       *
5: **      de mostrar um relógio no display LCD, ao apertar e                       *
6: **      segurar o pushbutton RB0, é possível alterar os                         *
7: **      segundos, ao segurar o pushbutton RB1, é possível                       *
8: **      alterar os minutos e ao apertar o pushbutton RB2,                       *
9: **      é possível alterar as horas.                                           *
10: **                                     *                                     *
11: ** Arquivo: relógio.c                                                         *
12: ** Compilador: Mikro C PRO PIC v.6.4.0                                       *
13: **                                     *                                     *
14: ** Aluno: Carlos Magno do Nascimento Junior                                   *
15: ** UFLA - Lavras/MG - 06/11/2024                                             *
16: *****/
17:
18:
19: //Bibliotecas necessárias para manipulação de strings
20: #include <stdio.h>
21: #include <string.h>
22:
23: // Conexões LCD do kit PICGenios com 18F
24: sbit LCD_RS at RE2_bit;
25: sbit LCD_EN at RE1_bit;
26: sbit LCD_D4 at RD4_bit;
27: sbit LCD_D5 at RD5_bit;
28: sbit LCD_D6 at RD6_bit;
29: sbit LCD_D7 at RD7_bit;
30:
31: sbit LCD_RS_Direction at TRISE2_bit;
32: sbit LCD_EN_Direction at TRISE1_bit;
33: sbit LCD_D4_Direction at TRISD4_bit;
34: sbit LCD_D5_Direction at TRISD5_bit;
35: sbit LCD_D6_Direction at TRISD6_bit;
36: sbit LCD_D7_Direction at TRISD7_bit;
37: // Fim das conexões
38:
39: // Função para formatar as horas
40: void formatarHora(int horas, int minutos, int segundos, char* horaFormatada) {
41:     char strHoras[7]; // Array para armazenar horas como string (tamanho 7 para
a IntToStr)
42:     char strMinutos[7]; // Array para armazenar minutos como string (tamanho 7 pa
ara IntToStr)
43:     char strSegundos[7]; // Array para armazenar segundos como string (tamanho 7 p
para IntToStr)
44:
45:     // Converte horas, minutos e segundos para strings de 7 caracteres
46:     IntToStr(horas, strHoras);
47:     IntToStr(minutos, strMinutos);
48:     IntToStr(segundos, strSegundos);
49:
50:     // Remove espaços em branco e ajusta para dois dígitos
51:     if (horas < 10) {
52:         horaFormatada[0] = '0';
53:         horaFormatada[1] = strHoras[5];
54:     } else {
55:         horaFormatada[0] = strHoras[4];
56:         horaFormatada[1] = strHoras[5];
57:     }
58:     horaFormatada[2] = ':'; // Adiciona o separador
59: }
```

```
60:     if (minutos < 10) {
61:         horaFormatada[3] = '0';
62:         horaFormatada[4] = strMinutos[5];
63:     } else {
64:         horaFormatada[3] = strMinutos[4];
65:         horaFormatada[4] = strMinutos[5];
66:     }
67:     horaFormatada[5] = ':'; // Adiciona o separador
68:
69:     if (segundos < 10) {
70:         horaFormatada[6] = '0';
71:         horaFormatada[7] = strSegundos[5];
72:     } else {
73:         horaFormatada[6] = strSegundos[4];
74:         horaFormatada[7] = strSegundos[5];
75:     }
76:     horaFormatada[8] = '\\0'; // Adiciona o caractere nulo para finalizar a string
77: }
78:
79: void main()
80: {
81:     int horas = 15, minutos = 5, segundos = 50; //inicialização das variáveis
82:     char horaFormatada[9]; // Array para armazenar "HH:MM:SS" + '\\0'
83:
84:     ADCON0 = 0x00; // Configura todos pinos das portas para digital e
85:     ADCON1 = 0x0F; // desabilita o conversor A/D
86:     trisb = 7;
87:
88:     Lcd_Init(); //Inicializa o Display
89:     delay_ms(100);
90:     Lcd_Cmd(_LCD_CLEAR); //Limpa o Display
91:     delay_ms(100);
92:     Lcd_Cmd(_LCD_CURSOR_OFF); //Desabilita o cursor
93:     delay_ms(100);
94:
95:     //Loop infinito
96:     while(1){
97:         // Ao apertar botão RB0, ajusta os segundos
98:         while (portb.rb0 == 0){
99:             delay_ms(200);
100:             if (segundos < 59){
101:                 segundos += 1;
102:             }
103:             else {
104:                 segundos = 0;
105:                 if (minutos < 59){
106:                     minutos+=1;
107:                 }
108:                 else{
109:                     minutos = 0;
110:                     if(horas < 23){
111:                         horas += 1;
112:                     }
113:                     else{
114:                         horas = 0;
115:                         minutos = 0;
116:                         segundos = 0;
117:                     }
118:                 }
119:             }
120:         }
121:     }
```

```
119:         }
120:         //chama a função para formatar a hora com novos valores
121:         formatarHora(horas, minutos, segundos, horaFormatada);
122:         //Mostra hora no Display LCD
123:         Lcd_Out(1,5,horaFormatada);
124:     }
125:
126:     // Ao apertar botão RB1, ajusta os minutos
127:     while (portb.rb1 == 0){
128:         delay_ms(200);
129:         if (minutos < 59){
130:             minutos+=1;
131:         }
132:         else{
133:             minutos = 0;
134:             if(horas < 23){
135:                 horas += 1;
136:             }
137:             else{
138:                 horas = 0;
139:                 minutos = 0;
140:                 segundos = 0;
141:             }
142:         }
143:         //chama a função para formatar a hora com novos valores
144:         formatarHora(horas, minutos, segundos, horaFormatada);
145:         //Mostra hora no Display LCD
146:         Lcd_Out(1,5,horaFormatada);
147:     }
148:
149:     // Ao apertar botão RB2, ajusta as horas
150:     while (portb.rb2 == 0){
151:         delay_ms(200);
152:         if(horas < 23){
153:             horas += 1;
154:         }
155:         else{
156:             horas = 0;
157:         }
158:         //chama a função para formatar a hora com novos valores
159:         formatarHora(horas, minutos, segundos, horaFormatada);
160:         //Mostra hora no Display LCD
161:         Lcd_Out(1,5,horaFormatada);
162:     }
163:
164:     // Contador de horas normal
165:     delay_ms(1000);
166:     if (segundos < 59){
167:         segundos += 1;
168:     }
169:     else {
170:         segundos = 0;
171:         if (minutos < 59){
172:             minutos+=1;
173:         }
174:         else{
175:             minutos = 0;
176:             if(horas < 23){
177:                 horas += 1;
178:             }
179:             else{
180:                 horas = 0;
```

---

```
181:             minutos = 0;
182:             segundos = 0;
183:         }
184:     }
185: }
186: //chama a função para formatar a hora com novos valores
187: formatarHora(horas, minutos, segundos, horaFormatada);
188: //Mostra hora no Display LCD
189: Lcd_Out(1,5,horaFormatada);
190: }
191: } // fim do programa
192:
```