

```

1:  *****
2:  *****
3:  /* Microcontroller: PIC16F877A @ 8 Mhz + MICROGENIOS --> OK! */
4:  *****
5:  /* PROGRAMA EXEMPLO: RTC. OBJETIVO: Aprender a utilizar o RTC para leitura de
6:  Datas e Hora. MICROCONTROLADOR: PIC18F4520.
7:  PARA MAIS INFORMACOES REPORTAR-SE A FOLHA DE DADOS DO CI RTC --> DS1307.
8:  SOFTWARE: MikroC PRO for PIC; CRISTAL: 8Mhz. --> CHAVES DE FUNÇÃO:
9:  -----
10: |GLCD\LCD ( 1) = ON |DIS1 ( 1) = OFF |
11: |SCK ( 6) = ON |RESIS ( 6) = OFF |
12: |SDA ( 7) = ON |TEMP ( 7) = OFF |
13: |RTC ( 8) = ON |VENT ( 8) = OFF |
14: -----
15: OBS: - HABILITAR BIBLIOTECA LCD. - HABILITAR BIBLIOTECA I2C.
16: *****
17: // INCLUDES AND DEFINES
18: #include "lcdMicroGenios.h"
19: #define BUZZER PORTC.RC1
20: *****
21: // VARIAVEIS GLOBAIS
22: unsigned char ucHora; // ARMAZENA HORA.
23: unsigned char ucMinuto; // ARMAZENA MINUTO.
24: unsigned char ucSegundo; // ARMAZENA SEGUNDO.
25: unsigned char ucDia; // ARMAZENA DIA.
26: unsigned char ucMes; // ARMAZENA MES.
27: unsigned char ucAno; // ARMAZENA ANO.
28: unsigned char ucDia_Semana; // ARMAZENA DIA DA SEMANA.
29: char *ucTexto; // PONTEIRO PARA LEITURA DE STRINGS.
30: unsigned int i = 0, nTimes = 0, tHigh = 0, tLow = 0; // BeepBeep FUNCTION VARS
31: *****
32: // PROTOTIPOS DE FUNÇÃO.
33: void Display_LCD();
34: void Encontra_Dia_Semana();
35: void Grava_RTC();
36: void Leitura_RTC();
37: void Converte_BCD (unsigned char ucLinha,
38: unsigned char ucColuna,unsigned char ucValor);
39: *****
40: // FUNCTION TO SOUNDS BEEP N WITH TIMES THIGH AND TLOW
41: void BeepBeep(unsigned int nTimes, unsigned int tHigh, unsigned int tLow){
42: for (i = 0; i < nTimes; i++){
43: BUZZER = 0; VDelay_ms(tHigh); BUZZER = 1; VDelay_ms(tLow);}}
44: *****
45: void main(){
46: ADCON1= 0x07; // TORNA TODOS OS PINOS A/D COMO I/O DE USO GERAL
47: TRISA = 0; // DEFINE PORTA,B,C,D,E
48: TRISB = 0; TRISC = 0; TRISD = 0; TRISE = 0;
49: TRISC.RC1 = 0; BUZZER = 1; // BUZZER LOW ACTIVE
50: Lcd_Init(); // INICIALIZA LCD NO MODO 4 BIT
51: Lcd_Cmd(_LCD_CLEAR); // APAGA MOSTRADOR
52: Lcd_Cmd(_LCD_CURSOR_OFF); // DESLIGA CURSOR
53: Grava_RTC();
54: BeepBeep(3,15,150); // TOCA BUZZER NO INICIO
55: while(1){
56: Display_LCD(); // ESCRIVE NO MOSTRADOR LCD O VALOR
57: Leitura_RTC(); // EFETUA LEITURA DE SEGUNDO, MINUTO E HORAS DO DS1307
58: Delay_ms(200);}} // AGUARDA 200 milisegundos
59: *****
60: // ROTINA DE CONVERSÃO DE DADOS PARA BCD
61: void Converte_BCD (unsigned char ucLinha,
62: unsigned char ucColuna,unsigned char ucValor) {
63: unsigned char ucValor1, ucValor2;
64: ucValor1 = (ucValor >> 4 ) + '0';
65: // CONVERTE O PRIMEIRO NIBBLE EM BCD E APÓS EM STRING
66: Lcd_Chr(ucLinha,ucColuna,ucValor1); // ESCRIVE CARACTERE NO LCD
67: ucValor2 = (ucValor & 0x0F) + '0';
68: // CONVERTE O SEGUNDO NIBBLE EM BCD E APÓS ISSO, EM STRING

```

```

69:  Lcd_Chrcp(ucValor2);} // ESCRIBE CARACTERE NO LCD
70:  /*****
71:  void Leitura_RTC() { // ROTINA DE LEITURA DO DS1307
72:      I2C1_Start(); // INICIA COMUNICAÇÃO I2C
73:      I2C1_Wr(0xD0);
74:  // END. FIXO PARA DS1307: 1101000X, ONDE X = 0 É PARA GRAVAÇÃO
75:      I2C1_Wr(0);
76:  // END. ONDE COMEÇA A PROGRAMAÇÃO DO RELÓGIO, END DOS SEGUNDOS.
77:      I2C1_Repeated_Start(); // ISSUE I2C SIGNAL REPEATED START
78:      I2C1_Wr(0xD1);
79:  // END. FIXO PARA DS1307: 1101000X, ONDE X = 1 É PARA LEITURA
80:      ucSegundo = I2C1_Rd(1);
81:  // Lê o PRIMEIRO BYTE(SEGUNDOS), INFORMA QUE CONTINUA LENDO
82:      ucMinuto = I2C1_Rd(1); // Lê o SEGUNDO BYTE(MINUTOS)
83:      ucHora = I2C1_Rd(1); // Lê o TERCEIRO BYTE(HORAS)
84:      ucDia_Semana = I2C1_Rd(1);
85:      ucDia = I2C1_Rd(1);
86:      ucMes = I2C1_Rd(1);
87:      ucAno = I2C1_Rd(0);
88:  // Lê o SÉTIMO BYTE(ANO), ENCERRA AS LEITURAS DE DADOS
89:      I2C1_Stop(); // FINALIZA COMUNICAÇÃO I2C
90:  /*****
91:  // VALORES INICIAIS DO RELOGIO EM TEMPO REAL - RTC
92:  void Grava_RTC(){
93:      I2C1_Init(100000); // INICIA I2C COM FREQUENCIA DE 100 KHZ
94:      I2C1_Start(); // INICIA A COMUNICAÇÃO I2C
95:      I2C1_Wr(0xD0);
96:  // END. FIXO PARA DS1307: 1101000X, ONDE X = 0 É PARA GRAVAÇÃO
97:      I2C1_Wr(0);
98:  // END. ONDE COMEÇA A PROGRAMAÇÃO DO RELÓGIO, END. DOS SEGUNDOS.
99:      I2C1_Wr(0x55); // INICIA COM 55 SEGUNDOS.
100:      I2C1_Wr(0x58); // INICIA COM 58 MINUTOS.
101:      I2C1_Wr(0x23); // INICIA COM 23 h (FORMATO 24 HORAS).
102:      I2C1_Wr(0x03); // INICIA COM TERÇA-FEIRA
103:      I2C1_Wr(0x24); // INICIA COM DIA 24
104:      I2C1_Wr(0x12); // INICIA COM Mês 12
105:      I2C1_Wr(0x14); // INICIA COM ANO 2014
106:      I2C1_Stop(); // FINALIZA COMUNICAÇÃO I2C
107:  /*****
108:  void Encontra_Dia_Semana() {
109:      switch (ucDia_Semana) {
110:          case 01: ucTexto = "DOMINGO"; break; // CASO DIAS_SEMANA = 01
111:          case 02: ucTexto = "SEGUNDA"; break; // CASO DIAS_SEMANA = 02
112:          case 03: ucTexto = "TERÇA"; break; // CASO DIAS_SEMANA = 03
113:          case 04: ucTexto = "QUARTA"; break; // CASO DIAS_SEMANA = 04
114:          case 05: ucTexto = "QUINTA"; break; // CASO DIAS_SEMANA = 05
115:          case 06: ucTexto = "SEXTA"; break; // CASO DIAS_SEMANA = 06
116:          default: ucTexto = "SABADO"; } // SE NÃO FOR NENHUM DESSES
117:  /*****
118:  void Display_LCD() { // MOSTRA VALORES NO LCD E TOCA BUZZER PERIODICAMENTE
119:      Lcd_Out(1,1,"(Hora)");
120:      Converte_BCD(1,7,ucHora); // CONVERSÃO DA VARIÁVEL HORAS PARA BCD
121:      Lcd_Chrcp(':'); // ESCRIBE NO LCD
122:      Converte_BCD(1,10,ucMinuto); // CONVERSÃO DA VARIÁVEL MINUTO PARA BCD
123:      Lcd_Chrcp(':'); // ESCRIBE NO LCD
124:      Converte_BCD(1,13,ucSegundo); // CONVERSÃO DA VARIÁVEL SEGUNDO PARA BCD
125:  if (ucHora == 0x00 && ucMinuto == 0x00 && ucSegundo == 0x00)
126:      BeepBeep(1,50,150); // TOCA BUZZER HORA
127:  if (ucMinuto == 0x00 && ucSegundo == 0x00)
128:      BeepBeep( 1,25,100); // TOCA BUZZER MIN
129:  if (ucSegundo == 0x00) BeepBeep( 1,15, 50); // TOCA BUZZER SEG
130:      Lcd_Out(2,1,"(Data)");
131:      Converte_BCD(2,7,ucDia); // CONVERSÃO DA VARIÁVEL DIA PARA BCD
132:      Lcd_Chrcp('/'); // ESCRIBE NO LCD
133:      Converte_BCD(2,10,ucMes); // CONVERSÃO DA VARIÁVEL MES PARA BCD
134:      Lcd_Chrcp('/'); // ESCRIBE NO LCD
135:      Converte_BCD(2,13,ucAno); // CONVERSÃO DA VARIÁVEL ANO PARA BCD
136:      Encontra_Dia_Semana();
137:      Lcd_Out(3,1,ucTexto); // MOSTRA DIA DA SEMANA
138:  /*****

```