Microcontroladores PIC – Programación en C con ejemplos ejemplo-12 Home ebooks

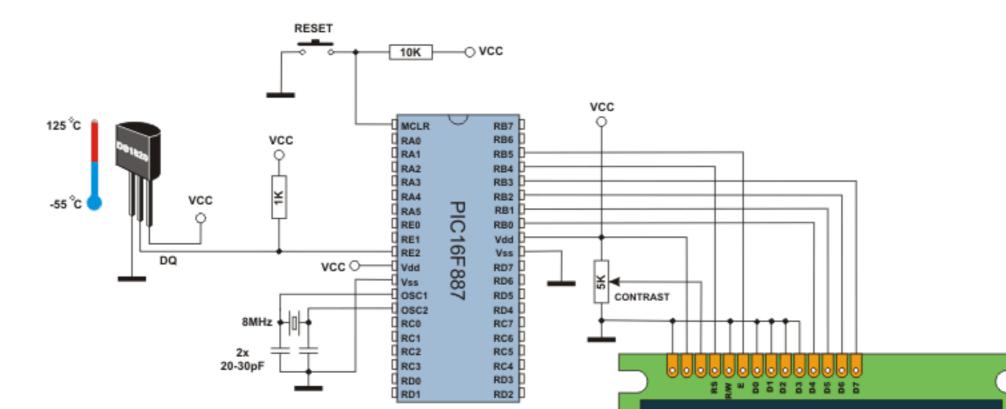
Book: Microcontroladores PIC – Programación en C con ejemplos

## 4.14 Ejemplo 12

## Medición de temperatura por medio del sensor DS1820. Uso del protocolo '1-wire'...

La medición de temperatura es una de las tareas más frecuentes realizadas por el microcontrolador. En este ejemplo, se utiliza un sensor DS1820 para medir. Es capaz de medir en el rango de 55 °C a 125 °C con exactitud de 0.5 °C. Para transmitir los datos al microcontrolador se utiliza un tipo especial de la comunicación serial denominado 1-wire. Debido al hecho de que estos sensores son simples de utilizar y de grandes capacidades, los comandos utilizados para hacerlos funcionar y controlarlos tienen la forma de funciones almacenadas en la librería One\_Wire. En total son las siguientes tres funciones:

- Ow\_Reset se utiliza para reiniciar el sensor;
- Ow\_Read se utiliza para recibir los datos del sensor; y
- Ow\_Write se utiliza para enviar los comandos al sensor



Este ejemplo muestra la ventaja de utilizar librerías con las funciones listas para ser utilizadas. Concretamente, no tiene que examinar la documentación proporcionada por el fabricante para utilizar el sensor. Basta con copiar alguna de estas funciones en el programa. Si le interesa saber cómo se declaran, basta con pulsar sobre alguna de ellas y seleccione la opción Help.

Temperature:

026.5000°C

```
// Conexiones del módulo LCD
sbit LCD_RS at RB4_bit;
sbit LCD_EN at RB5_bit;
sbit LCD_D4 at RB0_bit;
sbit LCD_D5 at RB1_bit;
sbit LCD_D6 at RB2_bit;
sbit LCD_D7 at RB3_bit;
sbit LCD_RS_Direction at TRISB4_bit;
sbit LCD_EN_Direction at TRISB5_bit;
sbit LCD_D4_Direction at TRISB0_bit;
sbit LCD_D5_Direction at TRISB1_bit;
sbit LCD_D6_Direction at TRISB2_bit;
sbit LCD_D7_Direction at TRISB3_bit;
// Final de conexiones del módulo LCD
const unsigned short TEMP_RESOLUTION = 9;
char *text = "000.0000";
unsigned temp;
void Display_Temperature(unsigned int temp2write) {
 const unsigned short RES_SHIFT = TEMP_RESOLUTION - 8;
  char temp_whole;
 unsigned int temp_fraction;
  // comprobar si la temperatura es negativa
 if (temp2write & 0x8000) {
 text[0] = '-';
 temp2write = ~temp2write + 1;
  // extraer temp_whole
  temp_whole = temp2write >> RES_SHIFT ;
  // convertir temp_whole en caracteres
 if (temp_whole/100)
    text[0] = temp whole/100 + 48;
 else
   text[0] = '0';
  text[1] = (temp_whole/10)%10 + 48; // Extraer dígito de decenas
  text[2] = temp_whole%10 + 48;
                                       // Extraer dígito de unidades
  // extraer temp_fraction y convertirlo en unsigned int
  temp fraction = temp2write << (4-RES SHIFT);</pre>
  temp_fraction &= 0x000F;
  temp_fraction *= 625;
  // convertir temp_fraction en caracteres
  text[4] = temp_fraction/1000 + 48;
                                      // Extraer dígito de miles
  text[5] = (temp fraction/100)%10 + 48; // Extraer dígito de centenas
  text[6] = (temp_fraction/10)%10 + 48; // Extraer dígito de decenas
  text[7] = temp_fraction%10 + 48;  // Extraer dígito de unidades
 // Visualizar temperatura en el LCD
 Lcd_Out(2, 5, text);
void main() {
 ANSEL = 0;
                          // Configurar los pines AN como digitales
 ANSELH = 0;
 C10N_bit = 0;
                           // Deshabilitar los comparadores
 C20N_bit = 0;
                          // Inicializar el LCD
 Lcd_Init();
 Lcd_Cmd(_LCD_CLEAR);
                         // Borrar el LCD
 Lcd_Cmd(_LCD_CURSOR_OFF); // Apagar el cursor
 Lcd_Out(1, 1, " Temperatura: ");
 // Visualizar el carácter de grado, 'C' para centígrados
 Lcd_Chr(2,13,223); // distintos visualizadores LCD tienen diferentes códigos
 // de caracteres para el grado
 // si ve la letra griega Alfa, intente introducir 178 en vez de 223
 Lcd_Chr(2,14,'C');
 //--- bucle principal
 do {
   //--- realizar lectura de temperatura
   Ow_Reset(&PORTE, 2);
                           // Señal de reinicio de Onewire
   Ow_Write(&PORTE, 2, 0xCC); // Ejecutar el comando SKIP_ROM
   Ow_Write(&PORTE, 2, 0x44); // Ejecutar el comando CONVERT_T
   Delay_us(120);
   Ow_Reset(&PORTE, 2);
   Ow_Write(&PORTE, 2, 0xCC); // Ejecutar el comando SKIP_ROM
   Ow_Write(&PORTE, 2, 0xBE); // Ejecutar el comando READ_SCRATCHPAD
   temp = Ow_Read(&PORTE, 2);
   temp = (Ow_Read(&PORTE, 2) << 8) + temp;</pre>
   //--- Formatear y visualizar el resultado en el LCD
   Display_Temperature(temp);
   Delay_ms(500);
 } while (1);
```

Para que este ejemplo funcione apropiadamente, es necesario marcar las siguientes librerías en la ventana Library Manager antes de compilar el programa:

- One\_Wire
- LCD

