**Uso del LCD externo tipo Hitachi HD44780**

**conectado a PIC 18 con el compilador XC8:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Adaptado de MPLAB® C18 C COMPILER LIBRARIES (DS51297F)  por**: Rommel Contreras** - <rommeljose@gmail.com> | | | | |
| **Descripciones de funciones** | | | |
| **BusyXLCD** | |  | |
| **Función:** | | ¿Está ocupado el controlador LCD? | |
| **Incluir:** | | xlcd.h | |
| **Prototipo:** | | unsigned char BusyXLCD( void ); | |
| **Observaciones:** | | Esta función devuelve el estado de la bandera de ocupado del controlador LCD Hitachi HD44780. | |
| **Valor de retorno:** | | 1 si el controlador está ocupado 0 en caso contrario. | |
| **Nombre del archivo:** | | busyxlcd.c | |
| **Ejemplo de código:** | | while( BusyXLCD() ); | |
| **OpenXLCD** | |  | |
| **Función:** | Configurar los pines de E/S del PIC® e inicializar el controlador del LCD. |
| **Incluir:** | xlcd.h |
| **Prototipo:** | void OpenXLCD( unsigned char ***lcdtype*** ); |
| **Argumentos:** | ***lcdtype***  Una máscara de bits que se crea realizando una operación AND a nivel de bits ('&') con un valor de cada una de las categorías enumeradas a continuación. Estos valores se definen en el archivo **xlcd.h.**  **Interfaz de datos:**  FOUR\_BIT Modo de interfaz de datos de 4 bits  EIGHT\_BIT Modo de interfaz de datos de 8 bits **Configuración del LCD:**  LÍNEA\_5X7 5x7 caracteres, visualización de una sola línea  LÍNEA\_5X10 Visualización de 5x10 caracteres  LINEAS\_5X7 5x7 caracteres, visualización de varias líneas |
| **Observaciones:** | Esta función configura los pines de E/S del PIC18 utilizados para controlar el controlador de LCD Hitachi HD44780. También inicializa este controlador. |
| **Nombre del archivo:** | openxlcd.c |
| **Ejemplo de código:** | OpenXLCD( EIGHT\_BIT & LINES\_5X7 ); |
| **putcXLCD** | |  | |

*Ver* **WriteDataXLCD.**

**putsXLCD putrsXLCD**

|  |  |
| --- | --- |
| **Función:** | Escribir una cadena en el controlador LCD Hitachi HD44780. |
| **Incluir:** | xlcd.h |
| **Prototipo:** | void putsXLCD( char \*buffer );  void putrsXLCD( const rom char \*buffer ); |
| **Argumentos:** | ***buffer***  Puntero a los caracteres que deben escribirse en el controlador de la pantalla LCD. |
| **Observaciones:** | Esta función escribe una cadena de caracteres situada en el *búfer* en el controlador LCD Hitachi HD44780. Detiene la transmisión cuando se encuentra un carácter nulo. El carácter nulo no se transmite.  Las cadenas situadas en la memoria de datos deben utilizarse con las versiones "puts" de estas funciones.  Las cadenas situadas en la memoria del programa, incluidas las literales de cadena, deben utilizarse con las versiones "putrs" de estas funciones. |
| **Nombre del archivo:** | putsxlcd.c putrxlcd.c |
| **Ejemplo de código:** | char mybuff [20]; |

putrsXLCD( "Hola Mundo" ); putsXLCD( mybuff );

**ReadAddrXLCD**

|  |  |
| --- | --- |
| **Función:** | Lea el byte de dirección del controlador LCD Hitachi HD44780. |
| **Incluir:** | xlcd.h |
| **Prototipo:** | unsigned char ReadAddrXLCD( void ); |
| **Observaciones:** | Esta función lee el byte de dirección del controlador LCD Hitachi HD44780. El controlador del LCD no debe estar ocupado cuando se realiza esta operación - esto puede ser verificado usando la función BusyXLCD.  La dirección leída desde el controlador es para la RAM del generador de caracteres o la RAM de datos de la pantalla, ¿dependiendo de la función anterior Set? RamAddr que fue llamada. |
| **Valor de retorno:** | Esta función devuelve una cantidad de 8 bits. La dirección está contenida en los 7 bits de orden inferior y la bandera de estado BUSY en el bit más significativo. |
| **Nombre del archivo:** | readaddr.c |
| **Ejemplo de código:** | char addr; |

while ( BusyXLCD() ); addr = ReadAddrXLCD();

**ReadDataXLCD**

|  |  |
| --- | --- |
| **Función:** | Leer un byte de datos del controlador LCD Hitachi HD44780. |
| **Incluir:** | xlcd.h |
| **Prototipo:** | char ReadDataXLCD( void ); |
| **Observaciones:** | Esta función lee un byte de datos del controlador LCD Hitachi HD44780. El controlador del LCD no debe estar ocupado cuando se realiza esta operación - esto puede verificarse utilizando la función BusyXLCD.  Los datos leídos desde el controlador son para la RAM del generador de caracteres o la RAM de datos de la pantalla, ¿dependiendo de la función anterior Set? RamAddr que fue llamada. |
| **Valor de retorno:** | Esta función devuelve el valor de los datos de 8 bits. |
| **Nombre del archivo:** | readdata.c |
| **Ejemplo de código:** | datos de char; |

while ( BusyXLCD() ); data = ReadAddrXLCD();

**SetCGRamAddr**

|  |  |
| --- | --- |
| **Función:** | Establece la dirección del generador de caracteres. |
| **Incluir:** | xlcd.h |
| **Prototipo:** | void SetCGRamAddr( unsigned char ***addr*** ); |
| **Argumentos:** | ***dirección***  Dirección del generador de caracteres. |
| **Observaciones:** | Esta función establece la dirección del generador de caracteres del controlador LCD Hitachi HD44780. El controlador de LCD no debe estar ocupado cuando se realiza esta operación - esto puede verificarse utilizando la función BusyXLCD. |
| **Nombre del archivo:** | setcgram.c |
| **Ejemplo de código:** | char cgaddr = 0x1F; |

while( BusyXLCD() ); SetCGRamAddr( cgaddr );

**SetDDRamAddr**

|  |  |
| --- | --- |
| **Función:** | Establece la dirección de los datos de la pantalla. |
| **Incluir:** | xlcd.h |
| **Prototipo:** | void SetDDRamAddr( unsigned char ***addr*** ); |
| **Argumentos:** | ***dirección***  Mostrar la dirección de los datos. |
| **Observaciones:** | Esta función establece la dirección de datos de la pantalla del controlador LCD Hitachi HD44780. El controlador de LCD no debe estar ocupado cuando se realiza esta operación - esto se puede verificar utilizando la función BusyXLCD. |
| **Nombre del archivo:** | setddram.c |
| **Ejemplo de código:** | char ddaddr = 0x10; |

while( BusyXLCD() ); SetDDRamAddr( ddaddr );

**WriteCmdXLCD**

|  |  |
| --- | --- |
| **Función:** | Escriba un comando en el controlador LCD Hitachi HD44780. |
| **Incluir:** | xlcd.h |
| **Prototipo:** | void WriteCmdXLCD( unsigned char ***cmd*** ); |
| **Argumentos:** | ***cmd***  Especifica el comando a realizar. El comando puede ser uno de los siguientes valores definidos en xlcd.h:  DOFF Apagar la pantalla  CURSOR\_OFF Habilitar la visualización sin cursor  BLINK\_ON Habilitar la visualización con el cursor parpadeante  BLINK\_OFF Activar la visualización con el cursor sin parpadear  SHIFT\_CUR\_LEFT El cursor se desplaza hacia la izquierda  SHIFT\_CUR\_RIGHT El cursor se desplaza a la derecha  SHIFT\_DISP\_LEFT La pantalla se desplaza hacia la izquierda  SHIFT\_DISP\_RIGHT La pantalla se desplaza hacia la derecha  Alternativamente, el comando puede ser una máscara de bits que se crea realizando una operación AND a nivel de bits ('& ') con un valor de cada una de las categorías enumeradas a continuación. Estos valores se definen en el archivo xlcd.h.  **Modo de transferencia de datos:**  FOUR\_BIT Modo de interfaz de datos de 4 bits  EIGHT\_BIT Modo de interfaz de datos de 8 bits  **Tipo de pantalla:**  LÍNEA\_5X7 5x7 caracteres, una sola línea  LÍNEA\_5X10 Visualización de 5x10 caracteres  LÍNEAS\_5X7 5x7 caracteres, varias líneas |
| **Observaciones:** | Esta función escribe el byte de comando en el controlador LCD Hitachi HD44780. El controlador del LCD no debe estar ocupado cuando se realiza esta operación - esto puede verificarse utilizando la función BusyXLCD. |
| **Nombre del archivo:** | wcmdxlcd.c |
| **Ejemplo de código:** | while( BusyXLCD() ); |

WriteCmdXLCD( EIGHT\_BIT & LINES\_5X7 );

WriteCmdXLCD( BLINK\_ON ); WriteCmdXLCD( SHIFT\_DISP\_LEFT );

**putcXLCD WriteDataXLCD**

|  |  |
| --- | --- |
| **Función:** | Escribe un byte en el controlador LCD Hitachi HD44780. |
| **Incluir:** | xlcd.h |
| **Prototipo:** | void WriteDataXLCD( char ***data*** ); |
| **Argumentos:** | ***datos***  El valor de los *datos* puede ser cualquier valor de 8 bits, pero debe corresponder a la tabla de RAM de caracteres del controlador LCD HD44780. |
| **Observaciones:** | Esta función escribe un byte de datos en el controlador LCD Hitachi HD44780. El controlador del LCD no debe estar ocupado cuando se realiza esta operación - esto puede verificarse utilizando la función BusyXLCD.  Los datos leídos desde el controlador son para la RAM del generador de caracteres o la RAM de datos de la pantalla, ¿dependiendo de la función anterior Set? RamAddr que fue llamada. |

**Ejemplo de USO:**

#include "./mcc\_generated\_files/mcc.h"

#include "./Lib\_mias\_all/microchip\_LCD/xlcd.h"

#include "./Lib\_mias\_all/micelaneos.h"

/\* Programa Principal \*/

void main(void)

{

// Initialize the device

SYSTEM\_Initialize();

LCD\_inicio();

while (1)

{

// Add your application code

while (BusyXLCD()); //wait untill LCD controller is busy

SetDDRamAddr(0x80); // Principio area visible LCD

while( BusyXLCD() );

putrsXLCD(" CARIPITO ");

\_\_delay\_ms(2000);

while (BusyXLCD());

SetDDRamAddr(0xC0); // Principio area visible LCD

while( BusyXLCD() );

putrsXLCD(" MONAGAS, Venz. ");

\_\_delay\_ms(2000);

while (BusyXLCD()); // Wait if LCD busy

WriteCmdXLCD(0x01); // Clear display

}

}

/\*\*

End of File

\*/

**Referencias:**

* **MPLAB® C18 C COMPILER LIBRARIES (DS51297F)**

El propósito de este documento es proporcionar información detallada sobre las bibliotecas y los archivos de objetos precompilados que se pueden utilizar con MPLAB de Microchip® C18 C Compiler.

* **MPLAB® IDE Quick Start Guide (DS51281)**

Describe cómo configurar el software IDE de MPLAB y usarlo para crear proyectos y dispositivos de programa.