**PRÁCTICA 3 EC**

**Apartado 1: Funciones simples del display LCD**

LCD\_Clr () {

Borrado de la pantalla

Delay de 1.52ms

Return

}

|  |  |  |
| --- | --- | --- |
| Borrado de pantalla | Comando para borrar | R0 |
| Registro de control | Comando para borrar | R1 |
| Tiempo de delay | Espera para que borre todo | R0 |
| Posición del cursor | Guardar en memoria | R0 |
| Dirección de la posición del cursor | Guardar en memoria | R1 |

LCD\_SetCursor (línea, posición) {

Si (línea < 1 y línea > 2) {

Return

}

Si (posición < 1 y posición > 40) {

Return

}

Posición--

Si (línea == 2) {

Posición += 0x40

}

Poner cursor en posición indicada

Delay 37micros

Return

}

|  |  |  |
| --- | --- | --- |
| Línea | Parámetro de entrada | R0 |
| Posición | Parámetro de entrada | R1 |
| Línea | Mantener registro | R4 |
| Posición | Mantener registro | R5 |
| Dirección de la posición | Guardar en memoria | R0 |
| Posicionar cursor | Comando para DDRAM | R0 |
| Registro de control | Comando para DDRAM | R1 |
| Tiempo de delay | Espera para que posicione | R0 |

LCD\_Shift (sentido) {

Si (sentido == derecha) {

Cargo derecha

Si (posición == 40) {

Return

}

}

Si (sentido == izquierda) {

Cargo izquierda

Si (posición == 0) {

Return

}

}

Muevo en sentido cargado

Delay 37micros

Return

}

|  |  |  |
| --- | --- | --- |
| Sentido de desplazamiento | Parámetro de entrada | R0 |
| Dirección de la posición de cursor | Sacar de memoria | R6 |
| Posición del cursor | Sacar de memoria | R5 |
| Desplazar display | Comando para desplazar | R0 |
| Límite del display | Comprobar que no desplaza demás | R4 |
| Registro de control | Comando para desplazar | R1 |
| Tiempo del delay | Esperar a que desplace | R0 |

Test\_1 {

Inicializar el LCD

Colocar la posición del primer mensaje

Escribir el primer mensaje

Colocar la posición del segundo mensaje

Escribir el segundo mensaje

Desplazar a derecha

Desplazar a izquierda

Esperar a pulsar tecla

Borrar pantalla

Colocar la posición del tercer mensaje

Escribir el tercer mensaje

Colocar la posición del cuarto mensaje

Escribir el cuarto mensaje

Desplazar a derecha

Esperar a pulsar tecla

Borrar pantalla

}

|  |  |  |
| --- | --- | --- |
| Dirección del mensaje | Dirección base para coger los caracteres | R4 |
| Desplazamiento del carácter | Offset del carácter | R5 |
| Carácter del mensaje | Carácter a escribir | R6 |
| Carácter que se escribe | Parámetro de entrada de LCD\_WChr | R0 |
| Línea donde escribir | Parámetro de entrada de LCD\_SetCursor | R0 |
| Posición donde escribir | Parámetro de entrada de LCD\_SetCursor | R1 |
| Número de veces a desplazar | Veces que se va a desplazar el display | R4 |
| Sentido del desplazamiento | Parámetro de entrada de LCD\_Shift | R0 |
| Tiempo de delay | Espera para que haga un desplazamiento que se lea el mensaje | R0 |

**Elección de mensajes de prueba y posiciones:**

Mensaje 1: “La practica funciona” Línea 1 posición 5, llega hasta la posición 25, al desplazar a la derecha se ve solo la última a.

Mensaje 2: “No lo ves?” Línea 2 posición 10, llega hasta la posición 20, al desplazar a la derecha queda totalmente oculto.

Mensaje 3: “Esta funcionando” Línea 1 posición 7, llega hasta la posición 23, al desplazar a la derecha queda totalmente oculto.

Mensaje 4: “Creo que ya se ha comprobado” Línea 2 posición 1, llega hasta la posición 28, al desplazar a la derecha se ve bado.

**Apartado 2: Escritura de mensajes largos en el LCD**

LCD\_BorradoLinea (línea) {

Si (línea != 1 y línea != 2) {

Return

}

LCD\_SetCursor(línea, 1)

Posición = 40

While (posición != 0) {

Borra carácter (meter espacio en el carácter)

Posición--

}

Return

}

|  |  |  |
| --- | --- | --- |
| Línea a borrar | Parámetro de entrada | R0 |
| Número de posiciones a borrar | Índice del bucle de borrado de posiciones | R4 |
| Posición por la que empezar a borrar | Parámetro de entrada de LCD\_SetCursor | R1 |
| Borrado de letra | Espacio para que se vea vacío, parámetro de entrada de LCD\_WChr | R0 |
| Posición del cursor | Guardar en memoria | R1 |
| Dirección de la posición del cursor | Guardar en memoria | R0 |

LCD\_WString (dirección del string) {

Desplazamiento = 0

Línea = 1

Posición = 0

CopiaPosicion

While (carácter [dirección+deplazamiento] != 0) {

Si (carácter == 0xD) {

CopiaPosicion = posición

Espero medio segundo

Si (línea == 1) {

Línea = 2

Posición = 0

LCD\_SetCursor(línea, posición + 1)

} Else {

Espero dos segundos

Línea = 1

LCD\_BorradoLinea(línea)

Posición = 0

LCD\_SetCursor(línea, posición + 1)

}

CopiaPoscicion -= 16

While (CopiaPosicion != 0) {

LCD\_Shift(izquierda)

CopiaPosicion--

}

}

Si (carácter=0x8 y posición != 0) {

Espero medio segundo

LCD\_SetCursor(línea, posición)

LCD\_WChr(espacio)

LCD\_SetCursor (línea, posición)

LCD\_Shift (izquierda)

Posición--

} Else {

Posición++

LCD\_WChr (carácter)

Espero medio segundo

Si (posición != 41 y posición >= 17) {

LCD\_Shift (derecha)

}

CopiaPosicion = Posición

Si (posición == 40) {

Si (línea == 1) {

Línea = 2

LCD\_BorradoLinea (línea)

Posición = 0

LCD\_SetCursor (línea, posición + 1)

} Else {

Espera dos segundos

Línea = 1

LCD\_BorradoLinea (línea)

Posición = 0

LCD\_SetCursor (línea, posición + 1)

}

CopiaPoscicion -= 16

While (CopiaPosicion != 0) { (para no repetir código y ya que ensamblador lo permite solo se ha escrito una vez de las dos que aparece en el pseudocódigo y se ha llamado desde los dos sitios)

LCD\_Shift (izquierda)

CopiaPosicion--

}

}

}

Desplazamiento++

}

Return

}

|  |  |  |
| --- | --- | --- |
| Dirección del string | Parámetro de entrada | R0 |
| Desplazamiento del carácter | Offset del carácter | R4 |
| Dirección del string | Copia de la dirección para no perderla, dirección base del carácter | R5 |
| Línea en la que estamos | Conocer la línea en la que estamos para controlar los saltos | R6 |
| Posición en la que estamos | Conocer la posición en la que estamos para controlar la escritura | R7 |
| Copia de la posición | Para controlar los desplazamientos | R8 |
| Carácter a escribir | Carácter que tenemos que conocer y escribir | R0 |
| Tiempo de delay | Espera para que se vean los caracteres y se lea, 0,5s para la escritura, 2s para borrado de primera línea… | R0 |
| Línea en la que posicionarse | Parámetro de entrada de LCD\_SetCursor | R0 |
| Posición en la que posicionarse | Parámetro de entrada de LCD\_SetCursor | R1 |
| Sentido del desplazamiento | Parámetro de entrada de LCD\_Shift | T0 |
| Línea que borrar | Parámetro de entrada de LCD\_BorradoLinea | R0 |
| Número de posiciones a desplazar | Desplazamientos necesarios para regresar a la primera posición del display | R8 |

Test\_2 {

Inicialización del LCD

Coger dirección del mensaje 1

Mostrar en debugger

Mostrar en LCD

Esperar a pulsar tecla

Borrar LCD

Coger dirección del mensaje 2

Mostrar en debugger

Mostrar en LCD

Esperar a pulsar tecla

Borrar LCD

Coger dirección del mensaje 3

Mostrar en debugger

Mostrar en LCD

Esperar a pulsar tecla

Borrar LCD

}

|  |  |  |
| --- | --- | --- |
| Dirección del mensaje | Parámetro de entrada de LCD\_WString | R0 |

**Elección de mensajes de prueba y posiciones:**

Mensaje 1: “Tengo un mensaje de mas de 40 caracteres sin salto de línea e incluso puede ser de mas de 80 caracteres para comprobar que hace todo bien”

En la primera línea se escribirá “Tengo un mensaje de mas de 40 caracteres” y se empezará a desplazar después de escribir mensaje (palabra completa).

En la segunda línea se deberá ver “ sin salto de línea e incluso puede ser ” y se empezará a desplazar después de escribir la i de línea.

En la primera línea después de un borrado se verá “de mas de 80 caracteres para comprobar q” y se desplaza a partir de la r de la palabra caracteres.

En la segunda línea de nuevo después de borrarla se escribirá “ue hace todo bien” y se desplazará una posición para mostrar la última n.

Mensaje 2: “Aquí hay un salto de linea” 0xD “como este por ejemplo”

En la primera línea se escribe “Aquí hay un salto de línea” y empieza a desplazarse para mostrar la o de salto y las siguientes letras.

En la segunda línea se escribe “como este por ejemplo” y se desplaza después de escribir ej de la palabra ejemplo.

Mensaje 3: “Y aquí voy a borrar el ultimo” 0x8 “ me he quedado sin o”

En la primera línea escribe “Y aquí voy a borrar el ultim me he queda” se empieza a desplazar a la derecha a partir de la primera r de borrar, se desplaza a la izquierda tras borrar la o y después continua desplazando a la derecha hasta el final de la línea.

En la segunda línea escribe “do sin o” y no se desplazará más.

**Preguntas sobre la práctica**

1. **Indica como distingue el controlador del display LCD si se está enviando un comando o un carácter para mostrar en el display.**

Al controlador se le indica con la función LCD\_W\_Byte si lo que se quiere escribir se desea escribir en el registro de control/status o en el registro de datos, por lo que si se escribe en el registro de control, el controlador lo interpretará como un comando, en cambio, si se escribe en el registro de datos, se estará escribiendo un dato para que lo muestre en el display.

1. **Indica la diferencia entre los comandos ‘Selección de modo’ y ‘Desplazamiento del Cursor/Pantalla’ en cuanto al desplazamiento del área visible.**

El comando selección de modo configura como se escribe y se desplaza el cursor por el display, si se incrementará o decrementará la posición según se vaya escribiendo (se escribe a derecha o izquierda), si se hará un scroll automático o no hará scroll la pantalla al escribir en el display, con lo que dependiendo de lo elegido en este comando, se puede mover a la derecha según se escribe de manera automática. Mientras que el comando de desplazamiento del cursor o pantalla, desplaza el cursor o la pantalla una posición a la derecha o la izquierda, según se indique en el comando, por lo que solo se desplazaría cuando se le indique al display.

1. **Razona la elección de los comandos empleados en la práctica.**

Se utilizan los siguientes comandos:

Borrar pantalla: para borrar la pantalla completa lo más rápido es utilizar este comando, que además regresa el cursor a la posición de inicio.

Desplazamiento del Cursor/Pantalla: para mover la pantalla se ha utilizado el movimiento de la pantalla sin contar con el movimiento del cursor, se ha utilizado en la función shift para poder desplazar y ver cómodamente el mensaje que se muestra en el display.

DDRAM: se ha utilizado este comando para colocar el cursor en la posición que queramos, es el más rápido y cómodo, ya que se podría hacer con cursor a inicio y el desplazamiento del cursor hasta la posición deseada pero es mucho menos eficiente y puede haber más errores en la programación.