

# Diseño con microcontroladores

## (Informática industrial 2012/2013)

### Módulo LCD\_I2C

El módulo LCD\_I2C es un módulo LCD paralelo estándar que tiene conectado un expansor I2C PCF8574. Este expansor recibe datos de 8 bits en serie por el bus I2C y los transfiere en paralelo por su puerto de 8 bits quasi-bidireccional. Asimismo, cuando recibe una orden de lectura por el bus I2C, lee del puerto un dato de 8 bits que convierte en serie y transmite por el bus I2C.

La figura 1 muestra el esquema de conexión del módulo fabricado por YwRobot, que es el que se utiliza en las prácticas. En ella se puede observar que los pines P4..P7 están conectados a los terminales D4..D7 del módulo LCD, P0, P1 y P2 a los terminales RS, R/ $\bar{W}$  y E y que P3<sup>1</sup> (BT) controla el backlight (led's de iluminación posterior) del módulo.

#### 1.- Protocolo de comunicación con el módulo LCD

Para enviar un dato –C: C<sub>7</sub>C<sub>6</sub>C<sub>5</sub>C<sub>4</sub>C<sub>3</sub>C<sub>2</sub>C<sub>1</sub>C<sub>0</sub>– al módulo LCD, ya sea una orden o el siguiente carácter a escribir, se ha de enviar la siguiente secuencia de datos al expansor I2C:

1. E(C<sub>7</sub>C<sub>6</sub>C<sub>5</sub>C<sub>4</sub>BT00RS): 4 bits superiores de C con R/ $\bar{W}$  y E a 0
2. E(C<sub>7</sub>C<sub>6</sub>C<sub>5</sub>C<sub>4</sub>BT10RS): 4 bits superiores de C con E a 1 para generar el pulso de enable
3. E(C<sub>7</sub>C<sub>6</sub>C<sub>5</sub>C<sub>4</sub>BT00RS): 4 bits superiores de C con E a 0 para terminar el pulso de enable
4. E(C<sub>3</sub>C<sub>2</sub>C<sub>1</sub>C<sub>0</sub>BT00RS): 4 bits inferiores de C con R/ $\bar{W}$  y E a 0
5. E(C<sub>3</sub>C<sub>2</sub>C<sub>1</sub>C<sub>0</sub>BT10RS): 4 bits inferiores de C con E a 1 para generar el pulso de enable
6. E(C<sub>3</sub>C<sub>2</sub>C<sub>1</sub>C<sub>0</sub>BT00RS): 4 bits inferiores de C con E a 0 para terminar el pulso de enable

BT y RS deben permanecer en el mismo estado, 0 o 1, durante toda la secuencia y RS determina si C es un carácter o una orden.

Para leer un dato –L: L<sub>7</sub>L<sub>6</sub>L<sub>5</sub>L<sub>4</sub>L<sub>3</sub>L<sub>2</sub>L<sub>1</sub>L<sub>0</sub>– del módulo LCD hay que configurar los pines P4..P7 en modo entrada. Para ello hay que escribir un 1 en dichos pines (Esto es así porque los pines del expansor tienen una salida quasi-bidireccional) y llevar a cabo el siguiente protocolo de escrituras y lecturas:

---

<sup>1</sup>La forma de controlar el transistor Q de este circuito no es adecuada ya que genera una corriente de salida de P3 y de la unión base-emisor del transistor muy elevada que reduce la vida útil de ambos componentes. Para corregirlo deberían añadir una resistencia que limite la corriente entre P3 y la base del transistor, tal y como se ha explicado en clase.

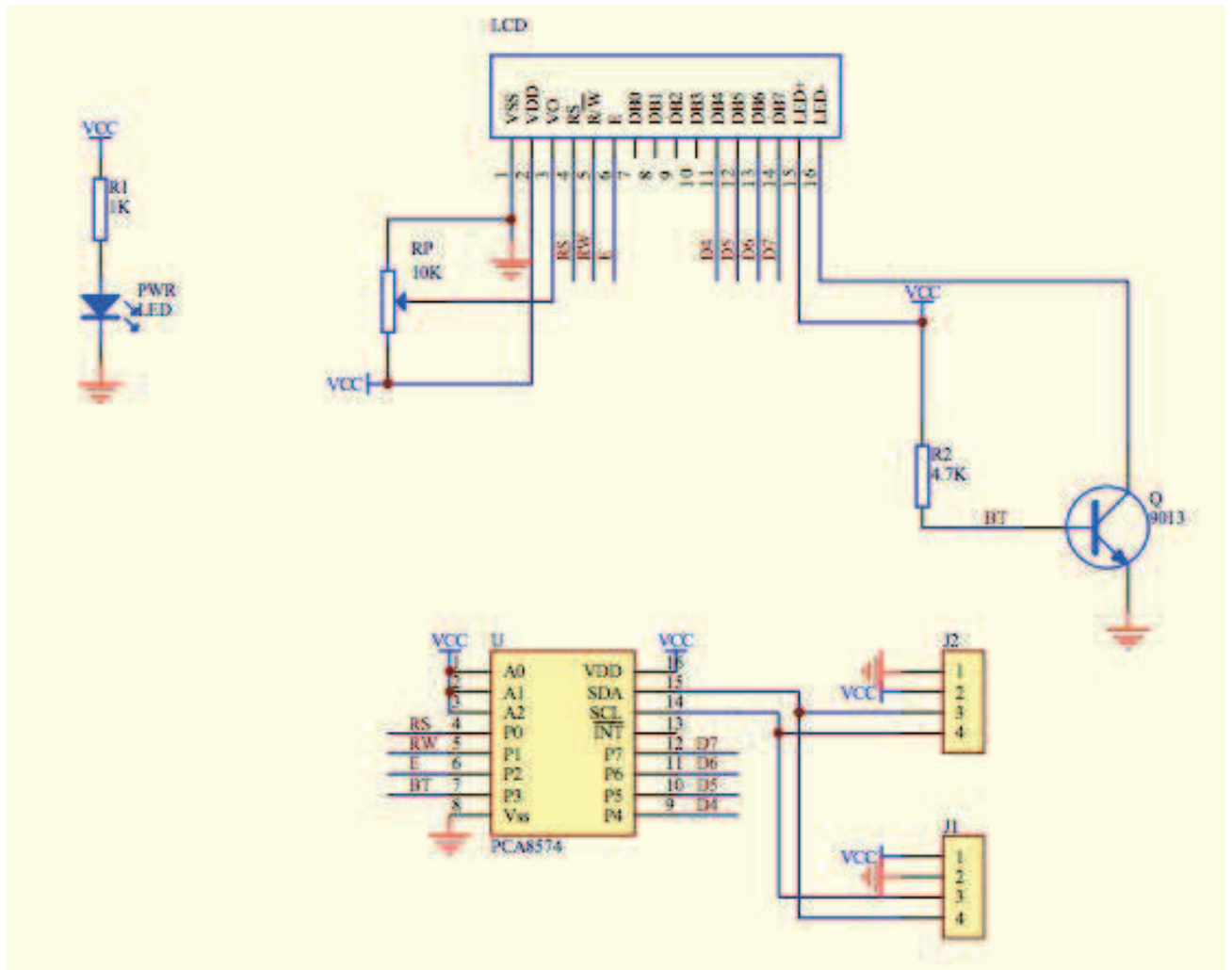


Figura 1: Expansor PCF8574 de YwRobot

1. E(1111BT01RS): 4 bits superiores a 1,  $R/\overline{W}$  a 1 y E a 0
2. E(1111BT11RS): 4 bits superiores a 1,  $R/\overline{W}$  a 1 y E a 1 para generar el pulso de enable
3. L(D7D6D5D4BT11RS): 4 bits superiores de L ( $L_7L_6L_5L_4$ ),  $R/\overline{W}$  a 1 y E a 1
4. E(1111BT01RS): 4 bits superiores a 1,  $R/\overline{W}$  a 1 y E a 0 para terminar el pulso de enable
5. E(1111BT11RS): 4 bits superiores a 1,  $R/\overline{W}$  a 1 y E a 1 para generar el pulso de enable
6. L(D7D6D5D4BT11RS): 4 bits inferiores de L ( $L_3L_2L_1L_0$ ),  $R/\overline{W}$  a 1 y E a 1
7. E(1111BT01RS): 4 bits superiores a 1,  $R/\overline{W}$  a 1 y E a 0 para terminar el pulso de enable

## 2.- Envío y recepción de datos mediante el bus I2C

Para la transmisión de datos entre el microcontrolador y el expansor I2C se debe usar la librería Wire de Arduino. En primer lugar se debe activar el módulo I2C del microcontrolador mediante la función `Wire.begin()`. Esto se debe hacer sólo una vez y el mejor sitio para ello es en la función `setup()`. El

envío de un dato  $-D-$  que incluye 4 bits del dato  $-C-$  que se quiere enviar al módulo LCD y el estado de las señales BT, E,  $R/\overline{W}$  y RS, se realiza mediante la siguiente secuencia:

1. `Wire.beginTransmission(0x27);`
2. `Wire.write(D);`
3. `Wire.endTransmission();`

A la función `beginTransmission()` se le debe pasar como parámetro la dirección I2C del expansor que, en este tipo de módulos, y en particular en el de YwRobot, es fija: 0x27. Cada una de las operaciones  $E(D)$  descritas en la sección anterior se realiza mediante una secuencia de estas tres funciones.

La lectura de un dato  $-D-$  que incluye 4 bits del dato  $-L-$  proporcionado por el módulo LCD y el estado en que se encuentran las señales BT, E,  $R/\overline{W}$  y RS, se realiza mediante la siguiente secuencia:

1. `Wire.requestFrom(0x27, 1);`
2. `while (!Wire.available()){}`
3. `D = Wire.read();`
4. `Wire.endTransmission();`

Además de la dirección I2C (0x27), la función `requestFrom()` recibe como parámetro el número de bytes que se desea leer que, en el caso del módulo LCD, siempre es 1. Cada una de las operaciones  $L(D)$  descritas en la sección anterior se realiza mediante una secuencia de estas cuatro funciones.

### **Versión optimizada de la comunicación con el módulo I2C**

El protocolo de comunicación con el expansor I2C se puede optimizar agrupando varias operaciones de escritura ( $E(D_i)$ ) consecutivas en una única trama I2C:

1. `Wire.beginTransmission(0x27);`
2. `Wire.write(D1);`
3. `Wire.write(D2);`
- .....
- n-1. `Wire.write(Dm);`
- n. `Wire.endTransmission();`

### 3.- Inicialización del módulo LCD

El controlador hd44780u que maneja el módulo LCD tiene un sistema de reset automático que activa cuando se conecta la alimentación (5 V). Este sistema de reset suele funcionar sin problemas siempre y cuando se cumplan las especificaciones del fabricante. El proceso dura varios milisegundos y se puede monitorizar mediante el flag busy. Tras él, se debe configurar el módulo LCD en modo 4 bits siguiendo las instrucciones de la pg. 209 del manual.

Sin embargo, este sistema de reset automático sólo se puede usar una vez cuando se alimenta el módulo LCD, en nuestro caso a través del arduino. Cuando se resetea el arduino, ya sea pulsando el botón de reset o al cargar un programa a través del usb, el módulo LCD permanece alimentado así que no lleva a cabo ninguna operación de reset interno. Por tanto, se debe usar el sistema de reseteo por software previsto por el controlador hd44780u y descrito en la pg. 213 del manual (pg. 212 si se usa el interfaz de 8 bits), de la que se ha extraído la figura 2. El proceso de reseteo propiamente dicho se puede dar por terminado tras el paso 7 (RS R/W: 00, DB7..DB4: 0010) en que se selecciona el modo de 4 bits y a partir de ahí proceder a la configuración del módulo LCD adecuada a nuestro sistema.

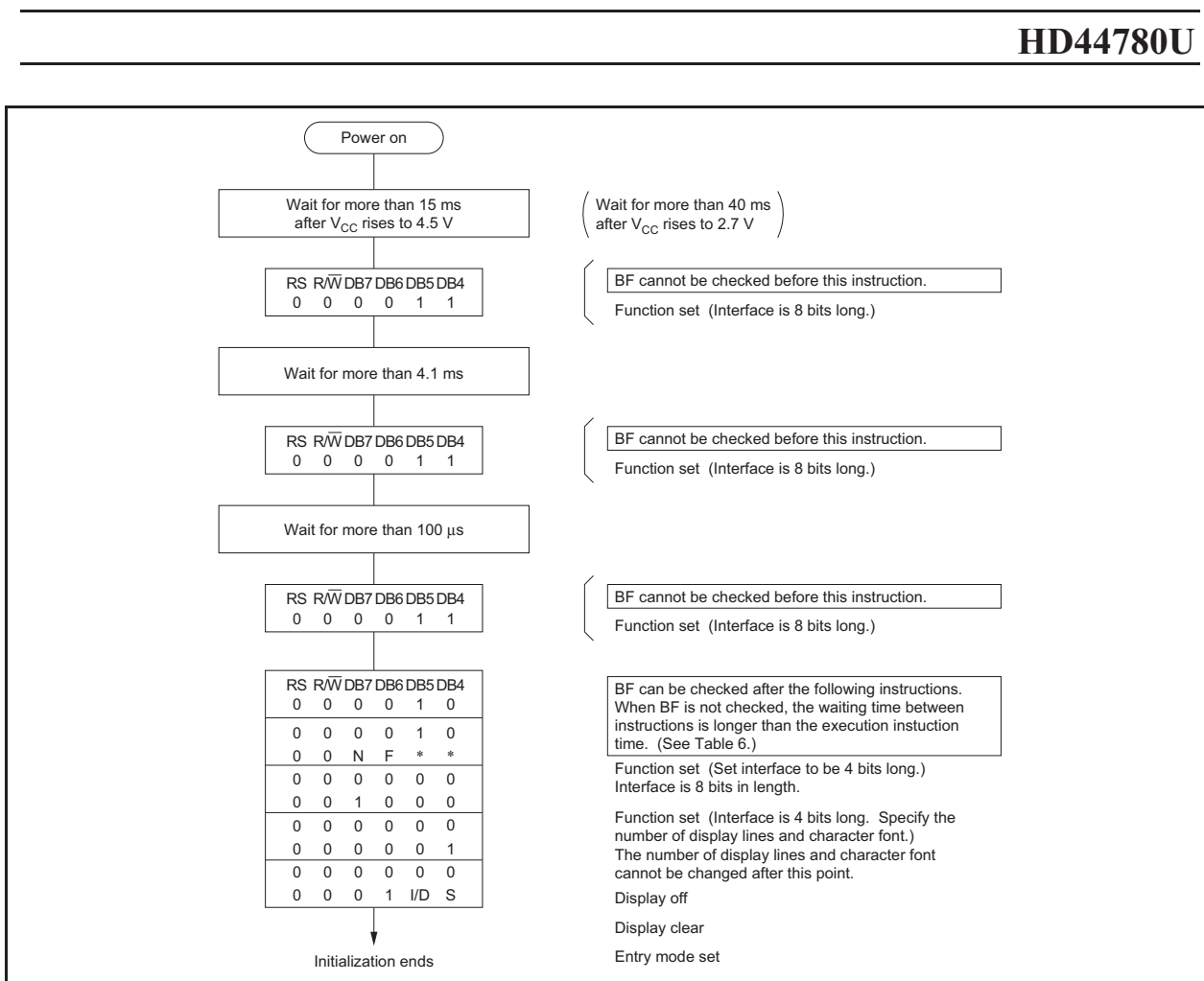


Figure 26 4-Bit Interface

Figura 2: Secuencia de reset del módulo LCD