

SISTEMAS EMPOTRADOS

3º Grado en Ingeniería Informática

PRÁCTICA 6: Introducción al puerto serie

6.1.– Introducción

Independientemente de que la aplicación final en un sistema empotrado no tenga interfaz de usuario, en otras fases del desarrollo es necesario tener una vía de comunicación con el usuario para depurar la aplicación, para depurar el hardware, para mantenimiento o diagnóstico del sistema, o para informar de sucesos.

Si el sistema no dispone de un interfaz visual, hay que dotar al hardware de un interfaz económico y que no consuma demasiado tiempo de CPU. La comunicación serie a través de una UART (*Universal Asynchronous Receiver/Transmitter*) integrada en el sistema o una UART virtual utilizando un puerto de entrada/salida digital, es la más económica y utilizada. Mediante esta comunicación serie tenemos una manera sencilla para mostrar o aceptar información del usuario si al otro lado de la línea serie conectamos un terminal.

6.2.– Objetivos

El objetivo de esta práctica será realizar una pequeña aplicación que se comunice con el usuario utilizando uno de los puertos serie de la placa MCB2300 y conectando al otro lado de la línea un ordenador personal que simule un terminal, por ejemplo, con el *Hyperterminal* de Windows XP.

6.3.– Material necesario:

- Ordenador personal con Windows XP.
- Placa de desarrollo MCB2300 de Keil.
- Adaptador USB–JTAG de la familia ULINK para depurar programas.
- Dos cables USB A–B.
- Cable serie RS232.

6.4.– Desarrollo de la práctica:

Se creará un programa que realice el producto de dos números BCD de un dígito. El resultado será un número de dos dígitos BCD. El interfaz de usuario que se mostrará en la simulación del Keil μ Vision 5 tendrá un aspecto similar a este:

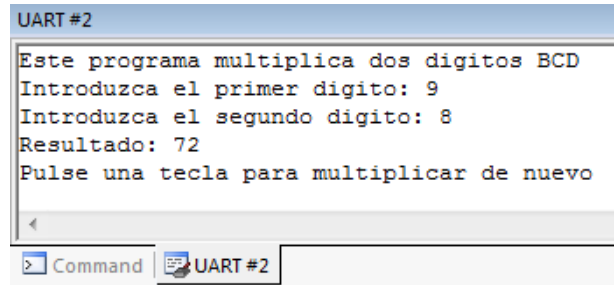


Figura 6-1: Interfaz de usuario a conseguir.

Se aceptarán valores de un dígito y se descartarán las teclas distintas de un valor BCD. Hay que tener en cuenta que los códigos ASCII de los números del 0 al 9 están comprendidos entre 0x30 y 0x3A, por lo que al valor introducido por el teclado habrá que restarle 0x30.

6.5.– Indicaciones:

Necesitaremos los ficheros de retarget.c y serial.c de alguna aplicación conocida (por ejemplo, Blinky de MCB2300). El manejador de proyectos tendrá un entorno como éste:

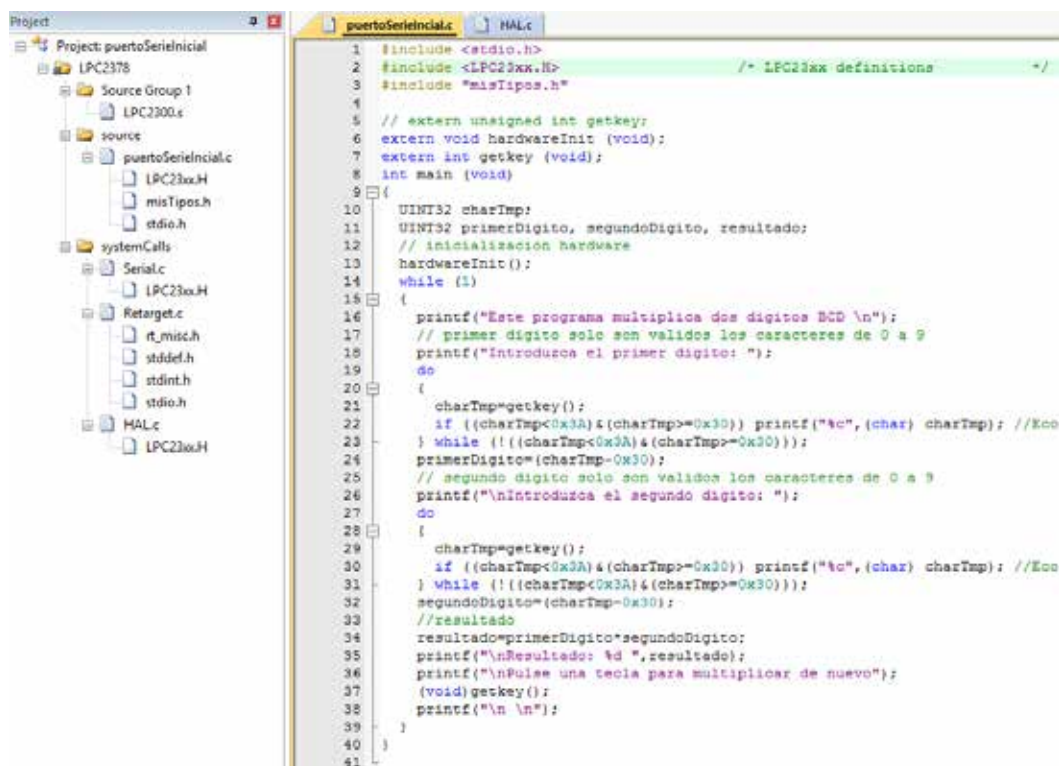
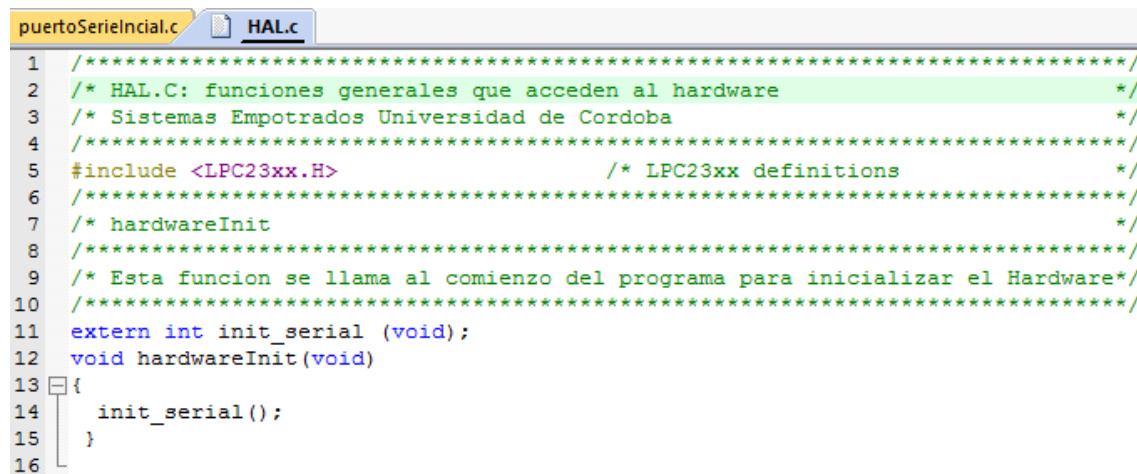


Figura 6-2: Programa principal de la aplicación.

En HAL.c solamente es necesario introducir la función `init_serial` que a su vez llama al `serial.c`, donde se definen las características del puerto serie del microcontrolador LPC2378 de nuestra placa.

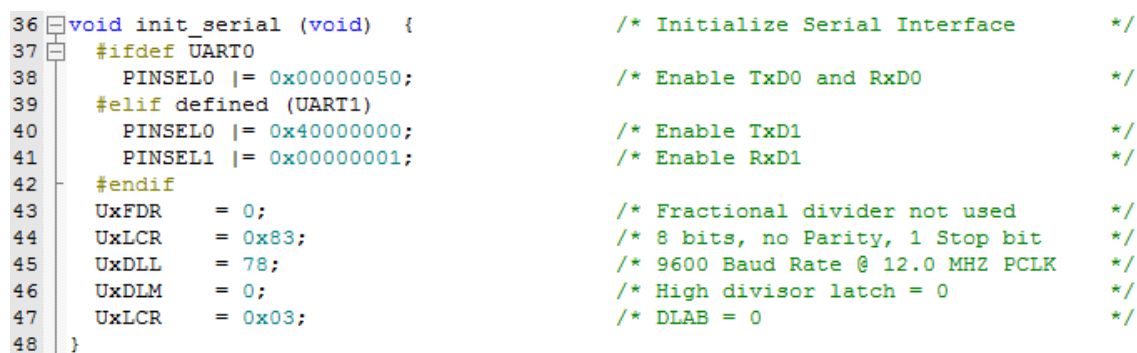


```

1  /*****
2  /* HAL.C: funciones generales que acceden al hardware */
3  /* Sistemas Empotrados Universidad de Cordoba */
4  /*****
5  #include <LPC23xx.H> /* LPC23xx definitions */
6  /*****
7  /* hardwareInit */
8  /*****
9  /* Esta funcion se llama al comienzo del programa para inicializar el Hardware*/
10 /*****
11 extern int init_serial (void);
12 void hardwareInit(void)
13 {
14     init_serial();
15 }
16
  
```

Figura 6-3: Fichero de capa de abstracción del hardware (HAL).

La parte de código que tiene el fichero `serial.c` para configurar el puerto serie del microcontrolador es la siguiente:



```

36 void init_serial (void) { /* Initialize Serial Interface */
37     #ifdef UART0
38         PINSEL0 |= 0x00000050; /* Enable TxD0 and RxD0 */
39     #elif defined (UART1)
40         PINSEL0 |= 0x40000000; /* Enable TxD1 */
41         PINSEL1 |= 0x00000001; /* Enable RxD1 */
42     #endif
43     UxFDR = 0; /* Fractional divider not used */
44     UxLCR = 0x83; /* 8 bits, no Parity, 1 Stop bit */
45     UxDLL = 78; /* 9600 Baud Rate @ 12.0 MHZ PCLK */
46     UxDLM = 0; /* High divisor latch = 0 */
47     UxLCR = 0x03; /* DLAB = 0 */
48 }
  
```

Figura 6-4: Configuración del puerto serie.

Se observa que en este fichero se ha definido el reloj de 12 MHz del LPC2378 y una velocidad de transmisión de 9600 baudios. Por tanto, nuestra conexión del *hyperterminal* tendremos que definirla para 9600 baudios, 8 bits, sin paridad, un bit de parada y sin control de flujo.

El generador de velocidad de transmisión es un pre-escalador de dieciséis bits que divide el PCLK para generar una señal de reloj en la UART igual a 16 veces la velocidad de transmisión. Por tanto la fórmula utilizada para calcular la velocidad de transmisión de la UART es:

$$\text{Divisor} = \text{Pclk} / (16 \times \text{BAUD})$$

En el caso de 12 MHz

$$\text{Divisor} = 12.000.000 / (16 \times 9600) = (\text{approx}) 78 \text{ o } 0x4E$$

Con el valor aproximado de 78 se obtiene una velocidad de transmisión real de 9615 baudios. Normalmente no es posible obtener una velocidad de transmisión exacta para la UART, sin embargo, se trabaja con un error máximo del 5% en la temporización.

Tendremos que conectar con un cable serie RS232 nuestra placa por su COM1 a un puerto serie del ordenador, y observar el menú a través del hyperterminal de Windows.