

SISTEMAS EMPOTRADOS

3º Grado en Ingeniería Informática

PRÁCTICA 9: Aplicación integrando un sistema operativo *Instant-Up*

9.1. Introducción

Vamos a realizar un pequeño ejercicio para utilizar el sistema operativo *Instant-Up* en un programa que active una señal periódica por el puerto P4.24.

9.2. Material necesario:

- Ordenador personal con Windows 7.
- Placa de desarrollo MCB2300 de Keil.
- Adaptador USB-JTAG de la familia ULINK para depurar programas.
- Dos cables USB A-B.

9.3. Desarrollo de la práctica:

Crearemos una carpeta nueva para la realización de la práctica donde copiaremos los ficheros `serial.c`, `retarget.c`, `misTipos.h`, y `LPC2300.s` de cualquiera de las prácticas anteriores. Los ficheros en que realizaremos modificaciones son los siguientes:

- `Practica_SO_GPIO.c`: En este fichero se realizará el programa principal de la aplicación, desde donde llamaremos al resto de las funciones.
- `HAL.c`: Fichero para definir el *hardware* que utilizaremos en la aplicación, en esta práctica sólo el GPIO P4.24 y el *timer* 1.

El fichero fuente del sistema operativo está disponible en la plataforma Moodle.

Al crear un nuevo proyecto deberemos de incluir estos archivos de manera que el manejador de proyectos quede, por ejemplo, como se indica en la figura 9-1.

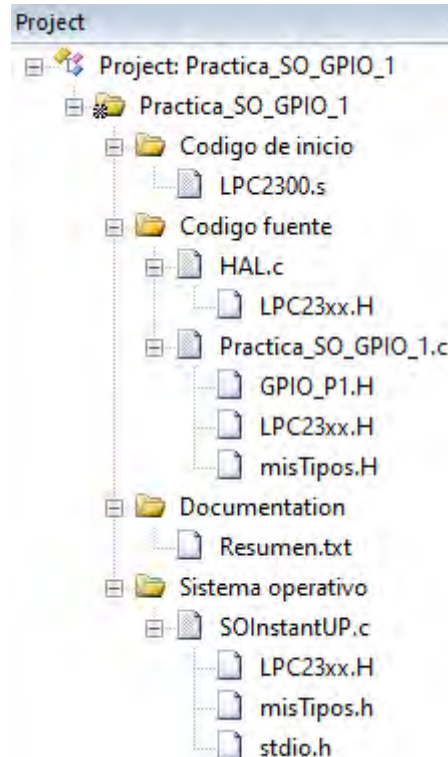
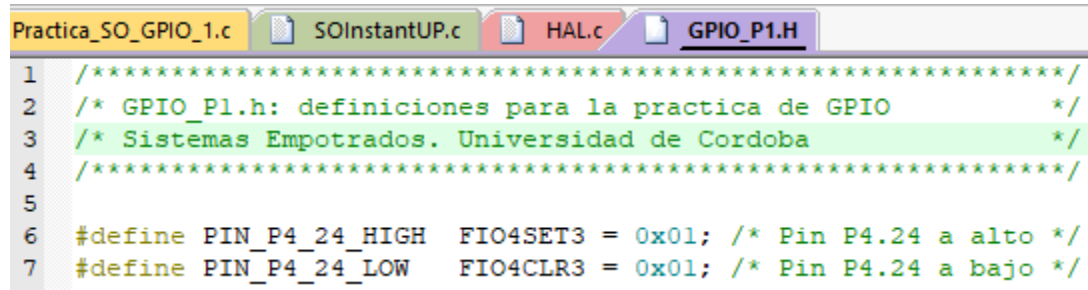


Figura 9-1: Vista previa del manejador de proyectos (*Project Manager*).

```
Practica_SO_GPIO_1.c | SOInstantUP.c | HAL.c | GPIO_P1.H
1  /*****
2  /* HAL.C: funciones generales que acceden al hardware.          */
3  /* Capa de abstracción del hardware (Hardware Abstract Layer)  */
4  /* Sistemas Empotrados Universidad de Cordoba                  */
5  *****/
6  #include <LPC23xx.H> /* LPC23xx definitions */
7  /*****
8  /* pinesSignalInit: Esta función configura los pines          */
9  /*                   P4.24 y P4.25 como salida                */
10 *****/
11 void pinesSignalInit(void)
12 {
13     PINSEL9 = 0x00000000;
14     PINMODE9 = 0x00000000;
15     FIO4DIR3 = 0x03;
16 }
17 /*****
18 /* hardwareInit: Esta funcion se llama al comienzo del programa */
19 /*               para inicializar el Hardware                    */
20 *****/
21 void hardwareInit(void)
22 {
23     pinesSignalInit(); // Configura los pines del circuito
24 }
```

Figura 9-2: Fichero de la capa de abstracción del hardware (HAL).



```
1  /*****  
2  /* GPIO_P1.h: definiciones para la practica de GPIO          */  
3  /* Sistemas Empotrados. Universidad de Cordoba              */  
4  *****/  
5  
6  #define PIN_P4_24_HIGH  FIO4SET3 = 0x01; /* Pin P4.24 a alto */  
7  #define PIN_P4_24_LOW   FIO4CLR3 = 0x01; /* Pin P4.24 a bajo */
```

Figura 9-3: Fichero cabecera (*header*).