


| | | | | | | | |
|--|----------|-------------|--------------------------|---------------------------------|--|---------------------|-------------------|
|  UNIVERSIDAD POLITÉCNICA DE MADRID ETSIS Telecomunicación | | | APELLIDOS: | | | | |
| | | | NOMBRE: | | | DNI: | |
| | | | MICROPROCESADORES | | | | |
| | | | Grupo Lab.: | | | Puesto lab.: | |
| Fecha | | | Curso | Calificaciones Parciales | | | Cal. Final |
| 16 | 4 | 2018 | 2 | | | | |

Para la realización de este examen dispone de 50 minutos. Debe crear en el ESCRITORIO de su PC una carpeta por cada hito, cuyo nombre debe tener el formato APELLIDOS_nombre_hitoX. Al finalizar el examen debe comprimir TODAS las carpetas en un único fichero 7-ZIP y subirlo al correspondiente enlace en Moodle.

Para cada hito verá una lista de objetivos que debe cumplir su programa. Si no logra todos ellos no se podrá obtener la máxima calificación. Cuando considere que tiene uno de los hitos funcionando debe levantar la mano para que el profesor pueda ir a validarlo. El profesor evaluará cada hito a cada alumno UNA ÚNICA VEZ. Una vez evaluado un hito no se puede modificar el código evaluado y debe pasar al siguiente hito. En cualquier caso NO SE QUEDE ESPERANDO A QUE LLEGUE EL PROFESOR, continúe con el siguiente hito (EN OTRA CARPETA). El profesor revisará todos los hitos pendientes.

Para la realización de este examen **no** se permite el uso de la función `wait()`, aunque sí se permite el uso de la librería `sw_tick_serial` y puede también reutilizar el código que haya escrito anteriormente durante las prácticas. No obstante lo anterior, **el código que presente para evaluar deberá estar libre de líneas sobrantes procedentes de código antiguo.**

Hito 1 (35 puntos): OFF/ON

El sistema debe emplear el pulsador derecho. Inicialmente en el *display* se mostrará el mensaje «OF». Con cada pulsación del pulsador el mensaje en el *display* pasará de «OF» a «On» y viceversa. Mientras en el *display* se muestre «On», el LED derecho (el conectado al pin *p19* —en LPC1768— o al pin *D11* —en l432kc—) permanecerá encendido, permaneciendo apagado en caso contrario.

El sistema debe contar el número de veces que se pasa de «OF» a «On» en una variable **no global** (que será del tipo adecuado para contener un número de 64 bits sin signo) llamada `cnt_on`. Cuando muestre este hito al profesor este le pedirá que active un punto de ruptura (*breakpoint*) que permita conocer el valor de la variable.

CRITERIOS:

Vº. Bº:

- ☐ Se llama a `wait()` (-35)
- ☐ Los mensajes mostrados en el *display* no son los indicados (-10)
- ☐ La respuesta al pulsador derecho no es la esperada o el sistema pasa de «OF» a «On» o viceversa de forma errática (-15)
- ☐ La multiplexación de los *displays* no es correcta (-10)
- ☐ El funcionamiento del LED derecho no es correcto (-10)
- ☐ No sabe poner un punto de ruptura (-15)
- ☐ No sabe ver el valor de una variable (-15)
- ☐ La variable cuenta mal las veces que se pasa de «OF» a «On» (-10)
- ☐ La variable no es del tipo adecuado o es global (-10)
- ☐ Los demás pulsadores influyen en el funcionamiento o los demás LED se encienden (-10)
- ☐ El brillo del *display* o del LED varía a lo largo del tiempo o hay código sobrante (-5)
- ☐ Otros (a valorar):

Hito 2 (35 puntos): CUENTA DE ENCENDIDOS

Modifique el código del hito anterior para que el *display*, en vez de mostrar el mensaje «On», muestre el valor de la variable `cnt_on` (en decimal con dos cifras). Mientras el *display* muestre «75», se encenderá además el LED izquierdo, permaneciendo apagado en caso contrario. El mensaje «OF» se seguirá mostrando como en el

hito anterior. Cuando muestre este hito al profesor este le pedirá que active un punto de ruptura (*breakpoint*) que permita modificar el valor de la variable `cnt_on`, de modo que no sea necesario realizar 149 pulsaciones sobre el pulsador para ver «75» en el *display* y el LED izquierdo encendido. La restante funcionalidad del hito anterior debe conservarse inalterada. No se preocupe ahora por limitar `cnt_on` a valores inferiores a 100.

CRITERIOS:

Vº. Bº:

- ☐ Se llama a `wait()` (-35)
- ☐ El valor mostrado en el *display* no es correcto (-15)
- ☐ El LED izquierdo no se enciende o apaga cuando corresponde (-10)
- ☐ No sabe modificar el valor de la variable para forzar la visualización del «75» ni el encendido del LED izquierdo (-15)
- ☐ Ya no se muestra el mensaje «OF» (-15)
- ☐ La restante funcionalidad del hito 1 se ha modificado o degradado (-15)
- ☐ Hay código sobrante (-5)
- ☐ Otros (a valorar):

Hito 3 (30 puntos): CUENTA

Modifique el código del hito anterior para que cuando el *display* muestre el valor de la variable `cnt_on`, este valor pueda ser incrementado con cada pulsación del botón central o decrementado con cada pulsación del botón izquierdo. No se permitirá que `cnt_on` sea mayor que 99 ni negativo. La restante funcionalidad de los hitos anteriores debe conservarse inalterada.

CRITERIOS:

Vº. Bº:

- ☐ Se llama a `wait()` (-30)
- ☐ No responde adecuadamente a los pulsadores (-15)
- ☐ El valor de `cnt_on` puede modificarse mediante los pulsadores cuando el *display* muestra «OF» (-10)
- ☐ El valor de `cnt_on` deja de incrementarse con cada paso de «OF» a «On» (-10)
- ☐ La restante funcionalidad de los hitos anteriores se ha modificado o degradado (-10)
- ☐ Hay código sobrante (-10)
- ☐ Otros (a valorar):