

 <b>UNIVERSIDAD</b> <b>POLITÉCNICA</b> <b>DE MADRID</b> <b>ETIS Telecomunicación</b>			<b>APELLIDOS:</b>					
			<b>NOMBRE:</b>			<b>DNI:</b>		
			<b>MICROPROCESADORES</b>					
			<b>Grupo Lab.:</b>			<b>Puesto lab.:</b>		
<b>Fecha</b>			<b>Curso</b>		<b>Calificaciones Parciales</b>			<b>Cal. Final</b>
6	6	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()` y tampoco se permite el uso de la librería `sw_tick_serial`, pero sí puede reutilizar el código que haya escrito anteriormente durante las prácticas (puede serle interesante el ejercicio 9 de la práctica 3). No obstante lo anterior, **el código que presente para evaluar deberá estar libre de líneas sobrantes procedentes de código antiguo.**

### Hito 1 (40 puntos): MEDIDA DE DISTANCIA EN PULGADAS

El sistema mostrará, en el *display* de 7 segmentos de la izquierda, la distancia, en pulgadas a la que se encuentra el objetivo (1 pulgada = 2.54 cm). Si la distancia es superior a 9 pulgadas, en el *display* izquierdo se mostrará el mensaje «r». La medida de distancia se realizará una vez cada dos segundos. El *display* de la derecha mostrará constantemente el símbolo «-».

El sistema debe contar el número de veces que la medida es estrictamente inferior a 4 pulgadas en una variable **no global** (que será del tipo adecuado para contener un número de 16 bits con signo) llamada `cnt_near`. Cuando muestre este hito al profesor se le pedirá que active un punto de ruptura (*breakpoint*) que permita conocer el valor de la variable.

#### CRITERIOS:

Vº. Bº:

- ☐ Se llama a `wait()` o se incluye o *linka* `sw_tick_serial` (-40)
- ☐ Los mensajes mostrados en el *display* no son los indicados (-15)
- ☐ La multiplexación de los *displays* no es correcta (-15)
- ☐ Las medidas se realizan a una velocidad distinta a la indicada (-10)
- ☐ No sabe poner un punto de ruptura (-20)
- ☐ No sabe ver el valor de una variable (-20)
- ☐ La variable cuenta mal el número de veces que la medida es inferior a 4 pulgadas (-10)
- ☐ La variable no es del tipo adecuado o es global (-10)
- ☐ Los pulsadores influyen en el funcionamiento o los LED se encienden (-10)
- ☐ El brillo del *display* varía (-10)
- ☐ Hay código sobrante (-10)
- ☐ Otros (a valorar):

## Hito 2 (40 puntos): CONTROL DEL BRILLO DEL *DISPLAY* DERECHO

Modifique el código del hito anterior para que el brillo del *display* derecho (y solo de él, es el que muestra el símbolo «-») varíe en función de la luz incidente sobre la LDR (a mayor luz incidente, mayor brillo y viceversa). La medida de luz y el brillo del *display* derecho se actualizarán una vez cada dos segundos. La restante funcionalidad del hito anterior debe conservarse inalterada.

### CRITERIOS:

Vº. Bº:

- ☐ Se llama a `wait()` o se incluye o `linka sw_tick_serial` (-40)
- ☐ La multiplexación de los *displays* no es correcta, siéndolo en el hito anterior (-20)
- ☐ El brillo del *display* izquierdo varía (-20)
- ☐ El brillo del *display* derecho, ante una iluminación intensa, es inferior al del otro *display* (-15)
- ☐ El brillo del *display* derecho, ante una iluminación muy débil, no es muy inferior al del otro *display* (-15)
- ☐ El brillo del *display* derecho se actualiza a una frecuencia distinta a la indicada (-10)
- ☐ La restante funcionalidad del hito 1 se ha modificado o degradado (-10)
- ☐ Hay código sobrante (-10)
- ☐ Otros (a valorar):

## Hito 3 (20 puntos): LED DE PROXIMIDAD

Modifique el código del hito anterior para que el LED derecho parpadee a una frecuencia de 10 Hz. El ciclo del trabajo del LED se actualizará cada dos segundos y será función de la distancia medida. Así: para una distancia de 9 pulgadas, el ciclo de trabajo será del 10 %; para una distancia inferior a 1 pulgada, el ciclo de trabajo será del 100 %; entre ambos extremos el ciclo de trabajo variará linealmente con la distancia. En caso de que el resultado de la medida sea superior a 9 pulgadas (*display* derecho mostrando «r»), el ciclo del trabajo será del 0 %. La restante funcionalidad de los hitos anteriores debe conservarse inalterada.

### CRITERIOS:

Vº. Bº:

- ☐ Se llama a `wait()` o se incluye o `linka sw_tick_serial` (-20)
- ☐ Se emplea un LED distinto del derecho (-20)
- ☐ El LED no parpadea a la frecuencia indicada (-10)
- ☐ El ciclo de trabajo del LED no evoluciona según lo esperado (-10)
- ☐ La restante funcionalidad de los hitos anteriores se ha modificado o degradado (-5)
- ☐ Hay código sobrante (-5)
- ☐ Otros (a valorar):