

EXPSBMB22019MANANA-SOL.pdf



Anónimo



Sistemas Basados en Microprocesador



3º Grado en Ingeniería Electrónica de Comunicaciones



Escuela Técnica Superior de Ingeniería y Sistemas de Telecomunicación
Universidad Politécnica de Madrid

MÁSTER

**Inteligencia Artificial
& Data Management**

MADRID

Conquista el mundo de la IA
en 10 meses



Ahora
25%
DE DESCUENTO

Aprenderás:

- Datos a IA generativa
- Big Data, ML, LLMs
- MLOps + cloud
- Visión estratégica

EOI Escuela de
organización
industrial



Info y descuentos

Importante

Puedo eliminar la publi de este documento con 1 coin


¿Cómo consigo coins?

Plan Turbo: barato

Planes pro: más coins

perdo
espacio



 UNIVERSIDAD POLITÉCNICA DE MADRID E.T.S.I.S. Telecomunicación	APELLIDOS:			
	NOMBRE:		DNI:	
	SISTEMAS BASADOS EN MICROPROCESADOR Examen Práctico Bloque 2			
	Grupo			
Fecha		Curso	Calificaciones Parciales	Cal. Final
15	11	2019		

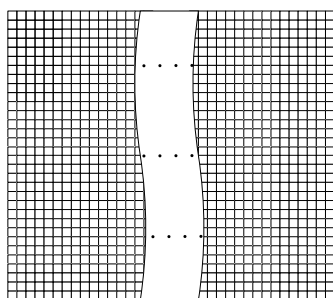
CUANDO COMPLETE CADA UNO DE LOS SIGUIENTES APARTADOS LLAME A SU PROFESOR PARA QUE VERIFIQUE SU FUNCIONALIDAD. UNA VEZ TERMINE TODOS LOS APARTADOS O BIEN CUANDO SU PROFESOR SE LO DIGA, COMPRIMA CADA UNO DE LOS PROYECTOS OBTENIDOS EN UN ÚNICO FICHERO ZIP Y SÚBALO A LA TAREA DE MOODLE CORRESPONDIENTE.

APARTADO A (5 puntos)

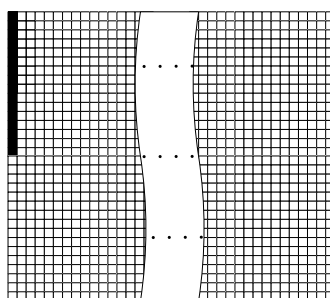
Partiendo de las prácticas realizadas anteriormente y usando CMSIS-RTOS RTX, construya una nueva aplicación para la tarjeta mbed NXP LPC1768 y la mbed Application Board, que tenga la siguiente funcionalidad:

- Inicialmente, tras un RESET, el LCD debe estar en blanco.
- Una pulsación “DERECHA” en el Joystick, hará que se visualice una barra horizontal que irá creciendo en función del número de pulsaciones realizadas. Cada vez que se realice una pulsación la barra irá creciendo en 1 píxel de anchura.
- Una pulsación “IZQUIERDA” en el Joystick decrementará en 1 píxel la anchura de la barra.
- La barra debe ocupar las dos primeras páginas de memoria de las cuatro disponibles (página 0 y 1).
- Si no se realiza ninguna de las dos pulsaciones en 4 segundos, se encenderá el LED1 constantemente.
- En su aplicación deben filtrarse los posibles rebotes del Joystick, utilizando los recursos provistos por el RTOS.
- Deberá visualizar en la ventana de watch, las variables que permiten conocer el número de pulsaciones de ambos gestos y la anchura en pixels de la barra.

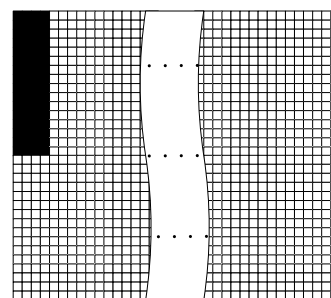
A modo de ejemplo se incluyen las siguientes figuras para aclarar el comportamiento en tres situaciones distintas:



0 pulsaciones



1 pulsación



4 pulsaciones

APARTADO B (3 puntos)

Modifique la aplicación obtenida en el apartado A, de manera que, si no se realiza ninguna de las dos pulsaciones en 4 segundos, se realice un destello de 100 ms. en el LED2.

APARTADO C (2 puntos)

Modifique la aplicación obtenida en el apartado A, que contemple las pulsaciones “ABAJO” del Joystick. Pulsaciones de este tipo permitirán que se utilice bien la parte superior, bien la parte inferior del LCD (página 2 y 3).

WUOLAH

```

void LED(void const *argument);
osTimerDef(one_shot, LED);
osTimerId one_shot_id;

void LED(void const *argument){

    GPIO_PinWrite (1,18,0);

}

void Thread1 (void const *argument) {

    osEvent evento;

    while (1) {

        evento = osSignalWait(0, 4000);

        if (evento.status == osEventTimeout){
            GPIO_PinWrite (1,18,1);
            osTimerStart(one_shot_id, 100);
        }
        else{
            if (evento.value.signals == Signal_S_RIGHT) EscribeBarra();
            if (evento.value.signals == Signal_S_LEFT) BorraBarra();
        }
    }
}

int EscribeBarra(){

    if (posicionBarra<127){
        posicionBarra++;
        buffer[posicionBarra] = 0xFF;
        buffer[posicionBarra+128] = 0xFF;
        copy_to_lcd();
    }
    return 0;
}

int BorraBarra(){

    if (posicionBarra>=0){
        buffer[posicionBarra] = 0x00;
        buffer[posicionBarra+128] = 0x00;
        posicionBarra--;
        copy_to_lcd();
    }
    return 0;
}

```