SISTEMAS BASADOS EN MICROPROCESADOR **Segundo parcial 2025-2026**

Apellidos:.....

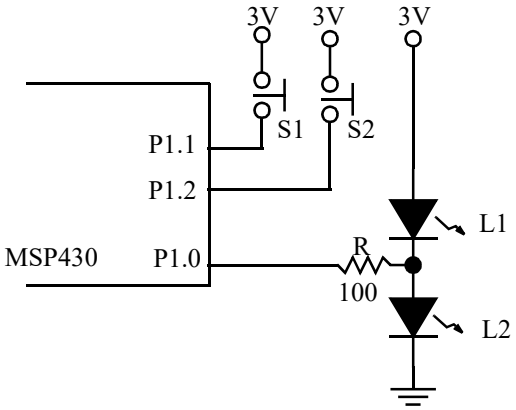
Nombre:.....

P1		P2	

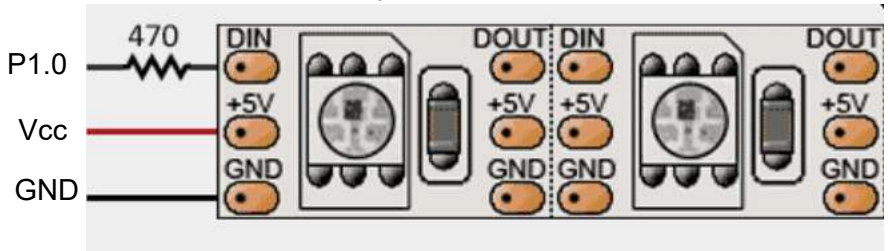
Duración 2:00 horas

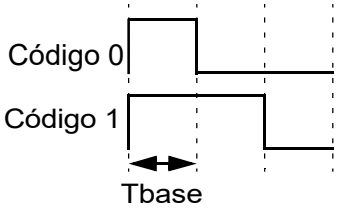
1.- Considere el circuito de la figura. La tensión directa de los leds es Vf=2V, por lo que no es posible encender los dos leds simultáneamente con una tensión de alimentación de 3V. Explique y haga un programa en ensamblador del MSP430 que gestione ***por multitarea cooperativa*** y en el modo de menor consumo el encendido/apagado de los leds en función de los pulsadores. Inicialmente ambos leds están apagados, pero el led activo es el L1 y el modo de pulsación. Cada vez que se pulsa S2, se cambia el led activo. En modo de pulsación, cada vez que se pulsa S1 se enciende el led activo y se apaga cuando se suelta. En modo de conmutación, cada pulsación de S1 hace que se invierta el estado del led activo. Se cambia de modo cuando se pulsa S2 estando S1 pulsado.

Nota: Considere los pulsadores libres de rebotes, el perro guardián desactivado, los puertos desbloqueados y la pila inicializada. Dispone de la librería `st.asm` y la función `cmp32`.



2.- El WS2812B es un módulo RGB inteligente que integra lógica de comunicaciones, tres controladores PWM, un led rojo, uno verde y otro azul dentro de cada píxel, permitiendo iluminar individualmente con hasta 256 niveles de brillo cada led y dando 2^{24}=16,777.216 colores. Los píxeles se conectan en cascada usualmente en tiras o paneles, usando un solo pin de datos; funcionan a 5V y son populares para proyectos de iluminación personalizados debido a su flexibilidad y capacidad de crear efectos visuales complejos.





La comunicación se hace por una única línea de datos. Se envían 24 bits por píxel, un byte por color, empezando por el msb en el orden RGB (rojo, verde, azul). Cada píxel se queda con los primeros 24 bits que le llegan y retransmite los demás, permitiendo que la información llegue al resto de leds de la tira. La transmisión de un 0 consiste en poner la línea en alto durante `Tbase`, y en bajo durante `2*Tbase`. La transmisión de un 1 mantiene la línea en alto durante `2*Tbase`, y en bajo `Tbase` (ver figura). Además, un código de `reset` consiste en dejar la línea a 0 durante al menos 50us. Hace que cada píxel actualice el color según la información recibida y que espere una nueva recepción. `Tbase=500ns`.

Realice un programa en ensamblador del MSP430 que configure el puerto P1.0 en su función primaria (TA0.1) para controlar una tira de 64 píxeles WS2812B con la ayuda del TA0 y por

intrerrupciones. El programa mantendrá un vector llamado `Leds` con la información de color de los píxeles y los enviará por la línea cada 20ms (es decir, a 50 fps o frames per second).

Con objeto de que la CPU no se vea sobrecargada en exceso, aumente la frecuencia de la misma a 16MHz. Minimice también el número de interrupciones del TA0 configurándolo para que genere los códigos a enviar por la línea usando el modo PWM (sólo una IRQ por bit en lugar de 2). Al final de la secuencia de bits deberá generar un código de `reset` para actualizar los leds y dejarlos listos para el siguiente ciclo.

Minimice el consumo del sistema (optimice el código, use los modos de bajo consumo,...). Considere el perro guardián desactivado, los puertos desbloqueados y la pila inicializada.