

- 1.- Realizar un programa que encienda y apague un LED cada 500ms
- 2.- Realizar un programa que encienda el LED correspondiente a la tecla que se mantiene presionada. SW1 – LED1, SW2 – LED2, SW3 – LED3, SW4 – LED4.
- 3.- Realizar un programa que encienda y apague cada vez que se presione la tecla el LED correspondiente. SW1 – LED1, SW2 – LED2, SW3 – LED3, SW4 – LED4.
- 4.- Realizar un programa que cuando se presione la tecla SW1, enciendan los LEDs en secuencia, esto es: primero LED1, luego LED2, LED3, LED4, LED1, y así sucesivamente, cada LED debe permanecer encendido un tiempo t . Si se presiona SW2, los LED se enciende en secuencia pero esta vez en el otro sentido, primero LED4, luego LED3, LED2,
Para detener la secuencia y apagar todos los LEDs se debe presionar la tecla SW3.
Con SW4 se podrá cambiar el tiempo t que se mantiene cada LED encendido. Por defecto el tiempo comienza en $t = 500ms$, luego cada vez que se presiona SW4, el tiempo se duplica, llegando a un máximo de $t = 2seg$, luego el tiempo vuelve a comenzar en $t = 250ms$.
- 5.- Realizar el ejercicio 4, sin utilizar *delay()* en el bucle principal (*loop()*), y utilizando la función *millis()* o *micros()*.
- 6.- Sin usar *delay()* en el bucle principal (*loop()*), realice un programa que simule la cerradura digital de una caja fuerte, el código de la caja consta de 6 dígitos y cada uno de estos se corresponde con la tecla según su numeración (1 = SW1, 2 = SW2, 3 = SW3, 4 = SW4).
Cuando inicia el programa la caja se encuentra abierta y la cerradura no tiene asignada ninguna contraseña, en esta situación el LED_BUILTIN (**D13**) destella encendiéndose y apagándose cada 250ms.
Al ingresar una contraseña de 6 dígitos, la caja pasa a estar cerrada, y el LED_BUILTIN comienza a destellar encendiéndose 100ms cada 1Seg, indicando que la caja esta cerrada.
Para abrirla ahora debe ingresarse el último código de 6 dígitos con el que se cerro. Si el código es correcto los cuatro LEDs parpadean 3 veces, indicando que la contraseña fue correcta, pasando ahora a estar la caja abierta, con lo cual el LED_BUILTIN pasa destellar cada 250ms. Si la contraseña ingresada es incorrecta los cuatro LEDs se encienden destellando cada 250ms durante 3seg, indicando que la contraseña es incorrecta, manteniéndose así la caja cerrada.
Para indicar que la tecla fue presionada, cada vez que se presione alguna de estas se deberá encender el LED correspondiente (SW1 = LED1, ...) durante 250ms.
- 7.- Sin usar *delay()* en el bucle principal (*loop()*), realice un programa que simule el juego **Simón dice**, el mismo consiste en que uno de los jugadores realiza un movimiento y los demás jugadores deben copiarlo. Si alguno de los jugadores no lo imita de forma adecuada o demora su respuesta, pierde. La versión electrónica del mismo consta de cuatro teclas y cuatro LEDs, los cuales se encienden utilizando una secuencia aleatoria.

Para esto considere:

- a) El juego se realizará considerando un total de 10 niveles, por lo que la secuencia final se corresponderá con el encendido de 10 LEDs en un orden particular al que deberá responderse con la misma secuencia de pulsación en las teclas.

- b) Si el jugador gana deberá parpadear el LED1 y si pierde el LED4, 5 veces a intervalos de 250 ms.
- c) Previo al inicio del juego se deberá representar una secuencia lumínica en los LEDs.
- d) Al presionar una tecla y luego de una pausa de 500 ms, deberá comenzar el juego encendiendo el primer led y esperando se presione el pulsador asociado.
- e) En cada pulsación se deberá encender el led asociado al pulsador.