

PROYECTO NO. 1

RELOJ

Instrucciones

Este proyecto se trabajará de forma **individual**. La fecha de entrega es la semana del 21 al 25 de marzo en sus respectivos laboratorios. Deberá hacer un **trabajo escrito** (no se imprime, se sube en PDF a Canvas, antes del **27 de marzo**) donde explique todas sus configuraciones, esquemáticos, cálculos y código debidamente comentado (Enlace a repositorio de GitHub). También deberá subir un **video** a YouTube explicando el funcionamiento de su proyecto.

Requisitos del reloj

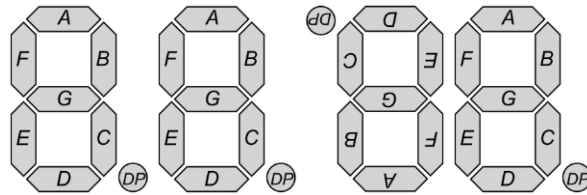
- Reloj digital que muestre la hora y los minutos (formato de 24 horas, al menos x4 displays de 7 segmentos)
- La configuración de la hora debe hacer *overflow* y *underflow*:
 - Si está configurando la hora y el valor de minutos está en 3 se puede hacer la siguiente secuencia: 03 → 02 → 01 → 00 → 59 → 58 → 57
 - La misma funcionalidad debe verse en las horas, días, meses y en la configuración del *timer* en horas y minutos
- Los dos puntos del centro deben parpadear cada 500ms (implementados con LEDs ó con los 7 segmentos)
- Los displays deben estar multiplexados
- Función de fecha:
 - El reloj debe poder desplegar la fecha en formato: 28/02 (28 de febrero) (Recuerde que cada mes tiene distinta cantidad de días)
- Función de *timer*:
 - Se define un tiempo que se quiere contar de forma descendente. Tiene que ser posible configurar minutos y segundos. El máximo tiempo de conteo es de 99 minutos y 59 segundos y el mínimo es de 1 segundo. Al finalizar el *timer* suena una alarma durante 1 minuto o hasta que la apaguemos (ambas formas de apagar la alarma deben estar implementadas)
- Función alarma:
 - Debe ser posible configurar una hora a la que suene una alarma, debe ser posible activarla o desactivarla (debe existir algún indicador para saber si la alarma está activada o no). Al igual que para el *timer*, la alarma suena durante 1 minuto o hasta que la apaguemos (ambas formas de apagar la alarma deben estar implementadas)
- Cuando se esté en modo de configuración (de hora, fecha, alarma o *timer*) deberá haber alguna indicación de que se está en ese modo (LED, parpadeo de los dígitos, etc)

- El reloj debe funcionar de forma similar a un reloj de pulsera, i.e. los botones funcionan de forma *presionar-y-soltar* (antirebotes)
- Tienen que utilizar al menos el *Timer0* del PIC y sus respectivas interrupciones
 - Puede implementar otros *timers* y otras interrupciones si lo desea

MATERIALES

Los materiales obligatorios son:

- Microcontrolador PIC16F887
- x4 displays de 7 segmentos
- x4 transistores NPN 2N3904 ó 2N2222
- x2 LEDs para los segundos (ó puede posicionar los displays 7 segmentos para que los puntos (DP) muestren los segundos)



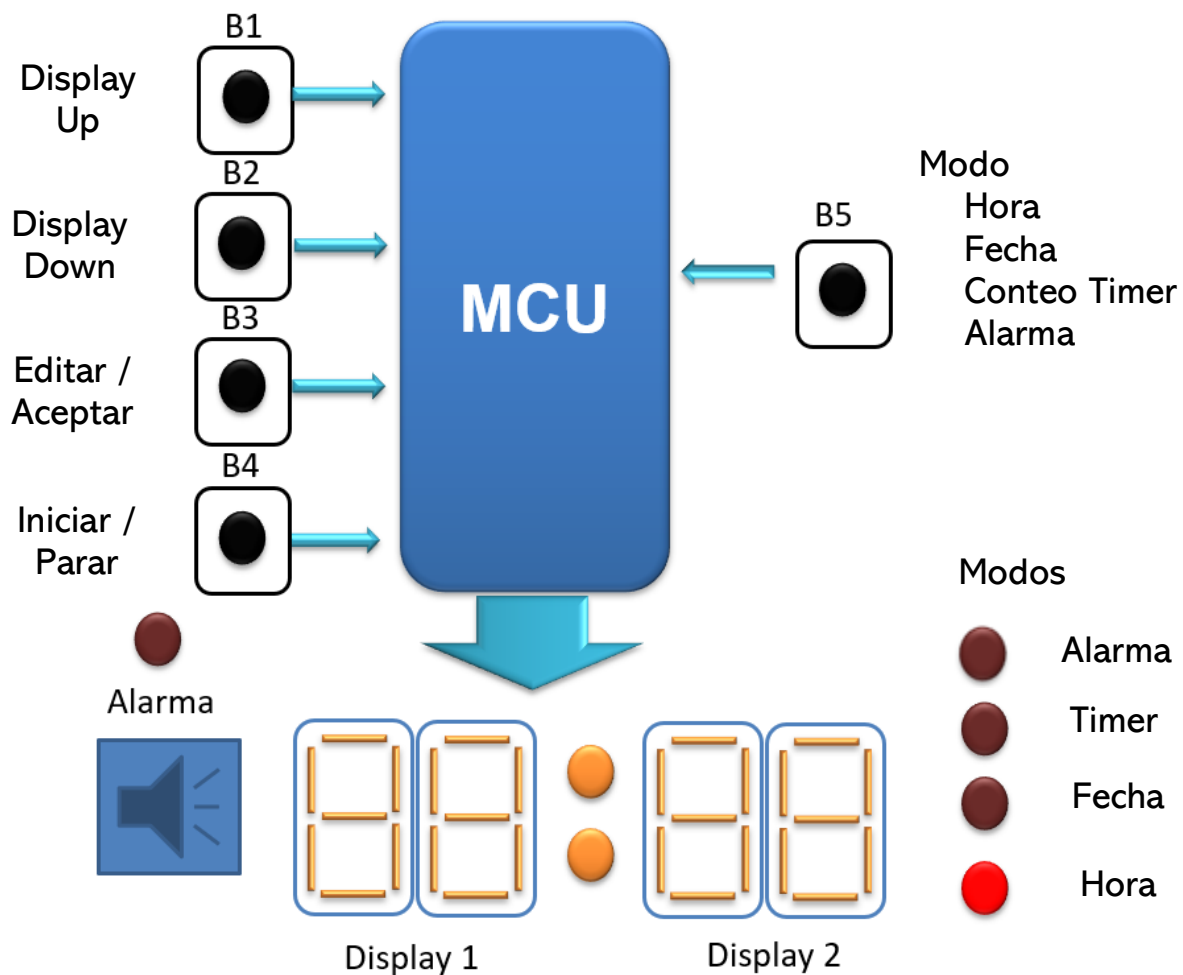
- Al menos x3 pushbuttons (Up/Down y modo)
 - Máximo 5 pushbuttons
- Resistencias para los LEDs, 7 segmentos, *pull-ups*, *pull-downs* y transistores

Algunos materiales **sugeridos** son:

- Base DIP40 para microcontrolador
- Regulador de voltaje 7805
- Batería de 9V
- Broche batería 9V
- 1 cristal con sus respectivos capacitores
- 1 Buzzer
- Placa de cobre / placa perforada para realizar la placa del circuito PCB / Protoboard
- Herramientas de soldadura, pinzas
- Velcro, tape de doble lado, cinchos plásticos
- Antirebotes implementado con un circuito RC

ESQUEMA GENERAL CIRCUITO

El circuito **sugerido** es el siguiente (No por fuerza tiene que seguir este diseño. Usted puede modificar el diseño si así lo desea: cantidad de displays de 7 segmentos, cantidad de pushbuttons, cantidad de LEDs, etc):



EXTRAS (SÓLO SON VÁLIDOS SI EL PROYECTO COMPLETO FUNCIONA ADECUADAMENTE)

- Cronómetro: Modo extra. Debe ser posible intercambiar de modos mientras el cronómetro esté activo y aún estando en algún otro modo, este debe seguir contando. Al regresar al modo cronómetro, debe verse reflejado todo el tiempo transcurrido estando en los otros modos.
- Implementación en PCB