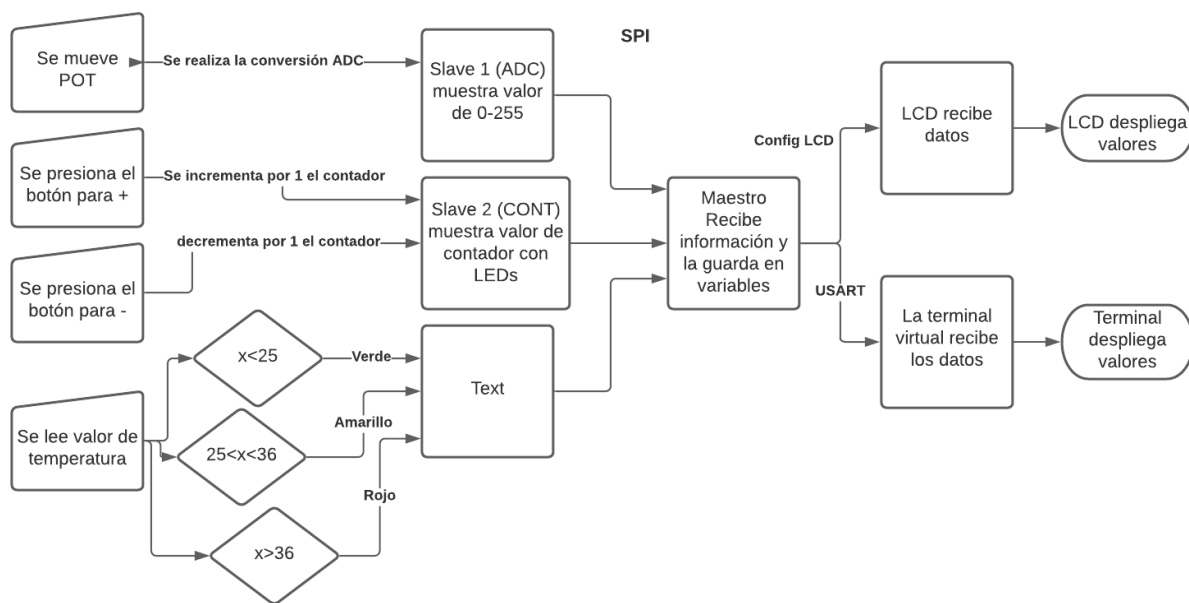


# Mini Proyecto #1: Comunicación SPI

Diagrama de Flujo:



## Pseudo Código:

### Slave 1:

Configurar pin RA0 como entrada analógica, los demás bits como salidas.

Habilitar interrupciones de SPI, configurar el slave select y bits de SSPSTAT y SSPCON para que funcione como esclavo.

Se utiliza la librería ADC ya creada para inicializar y utilizar interrupciones

Loop principal:

Comenzar conversión

Delay adc

Go bit = 1

PORTD = valor adc

Interrupción:

Revisar SSPIF

Spi write (valor adc)

SSPIF = 0

Revisar ADIF

Valor adc = ADRESH

ADIF = 0

### **Slave 2:**

Configurar RB0 y RB1 como entradas de los botones.

Habilitar interrupciones de SPI, configurar el slave select y bits de SSPSTAT y SSPCON para que funcione como esclavo.

Se utiliza la librería INT\_B ya creada para habilitar interrupciones puerto B y seleccionar bits.

Loop principal:

PORTD = contador

Interrupción:

Revisar SSPIF

Spi write (contador)

SSPIF = 0

Revisar RBIF

Si se presionó RB0, incrementar al contador

Si se presionó RB1, decrementar al contador

### **Slave 3:**

Configurar pin RA0 como entrada analógica, los demás bits como salidas.

Habilitar interrupciones de SPI, configurar el slave select y bits de SSPSTAT y SSPCON para que funcione como esclavo.

Se utiliza la librería ADC ya creada para inicializar y utilizar interrupciones

Cambiar el voltaje de referencia de 5V a 1.5V

Loop principal:

Comenzar conversión

Delay adc

Go bit = 1

Si el valor adc es menor a 25, prender RD0 (led verde)

Si el valor adc está entre 25 y 36, prender RD1 (led amarillo)

Si el valor adc es mayor a 36, prender RD2 (led rojo)

Interrupción:

Revisar SSPIF

Spi write (valor adc)

SSPIF = 0

Revisar ADIF

Valor adc =  $ADRESH * 150 / 255$  (porque se modificó voltaje de referencia)

ADIF = 0

### **Master:**

Solamente se tiene el bit RC7 como entrada, para la comunicación UART, todos los demás son salidas. No hay entradas analógicas. Se prenden los bits RC0, RC1 y RC2 porque se utilizarán como slave select y deben estar siempre en 1, y se rotan estar en 0 para realizar comunicación.

Bits de SSPSTAT y SSPCON para que funcione como maestro. Se configura con  $F_{osc}/4$ , que el idle sea low y que mande en flancos positivos.

Se utilizan las librerías de UART y LCD ya creadas para poder desplegar los datos. SPI para recibir datos.

Loop principal:

Se manda string en UART de los labels de los valores

Se cambia de línea

Se despliegan los valores recibidos de adc, contador y temperatura

Se cambia de línea

Se limpia la LCD

Se setea el cursor en 1,1

Se manda la string de labels

Se setea el cursor en 2,1

Se despliegan los valores recibidos de adc, contador y temperatura

Se realiza un delay de 500 ms

Se apaga sólo el SS1

spiWrite(1)

guardar valor del contador

Se apaga sólo el SS2

spiWrite(1)

guardar valor del adc

Se apaga sólo el SS3

spiWrite(1)

guardar valor de la temperatura