

## PSEUDOCÓDIGOS

### Slave 1:

**setup();**

(canal A2)--> analogico

(canales A2, SS-A5, SCK-C3, SDI-C4)→como entrada

(demás canales)→Digitales y como salida

Se activan bits de interrupciones→GIE, PEIE, SSPIE, ADC

Configuración del SPI: (SLAVE con SS enable, DATA Sample Middle, Clock Idle Low , Idle 2 Active)

Confiuracion del ADC: (Fosc/16, canal 2, Left Justified, Internal Ref Vol)

Empiezo conversion

**LoopPrincipal();**

if (ADC\_finish == 1) Reviso bandera pa ver si ya puedo empezar la siguiente conversion

ADC\_finish = 0; Apago la bandera

\_\_delay\_us(400); Hago acquisition time

Empezar siguiente conversion

**Interrupciones();**

if (PIR1bits.ADIF == 1) { /Interrupcion del ADC

PIR1bits.ADIF = 0; Apago bandera del ADC

ADC\_value = ADRESH; //Se cargan los 8 MSB del resultado al registro

ADC\_finish = 1; //Aviso que ya se puede hacer otra conversion

else if (PIR1bits.SSPIF == 1) { /Interrupcion del SPI

OK = spiRead(); /Recibo el dato

spiWrite(ADC\_value); /Envio el dato del ADC

PIR1bits.SSPIF = 0; /Apago la bandera mientras se espera el siguiente dato

## Slave 2:

**setup();**

(Ningún canal)--> analogico

(canales B0, B1, SS-A5, SCK-C3, SDI-C4)→como entrada

(demás canales)→Digitales y como salida

Se activan bits de interrupciones→GIE, PEIE, SSPIE, RBIE

Interrupt con change para bits B0 y B1

Configuración del SPI: (SLAVE con SS enable, DATA Sample Middle, Clock Idle Low , Idle 2 Active)

**LoopPrincipal();**

Envio dato del contador por SPI

**Interrupciones();**

if (INTCONbits.RBIF == 1) { //Interrupcion del interrupt-on-change para los botones

if (Boton de subida == 1) {

PORTD++; //aumento en 1 el puerto D, pues este es mi contador

} else if (Boton de bajada == 1) {

PORTD--; //disminuyo en 1 el puerto D, pues este es mi contador

}

Antirrebote en delay

INTCONbits.RBIF = 0; /Apago bandera del Interrupt on change

}else if (PIR1bits.SSPIF == 1) {

OK = spiRead();

spiWrite(PORTD);

PIR1bits.SSPIF = 0;

### Slave 3:

**setup();**

(Canal A0)--> analogico

(canales A0, Vref+ A3, Vref- A2, SS-A5, SCK-C3, SDI-C4)→como entrada

(demás canales)→Digitales y como salida

Se activan bits de interrupciones→GIE, PEIE, SSPIE, ADC

Configuración del SPI: (SLAVE con SS enable, DATA Sample Middle, Clock Idle Low , Idle 2 Active)

Configuración del ADC: (Fosc/16, canal 0, Left Justified, External Ref Vol)

Empiezo conversion

**LoopPrincipal();**

if (ADC\_finish == 1) Reviso bandera pa ver si ya puedo empezar la siguiente conversion

ADC\_finish = 0; Apago la bandera

\_\_delay\_us(400); Hago acquisition time

Empezar siguiente conversión

If (valor de ADC < valor digital correspondiente a 25 grados)

Encender led Verde, apagar las demas

If (valor de ADC > valor digital correspondiente a 25 grados y < valor digital correspondiente a 36 grados)

Encender led amarilla, apagar las demas

If (valor de ADC > valor digital correspondiente a 36 grados)

Encender led roja, apagar las demás

Se envia dato del ADC por SPI

**Interrupciones();**

if (PIR1bits.ADIF == 1) { /Interrupcion del ADC

PIR1bits.ADIF = 0; Apago bandera del ADC

ADC\_value = ADRESH; //Se cargan los 8 MSB del resultado al registro

ADC\_finish = 1; //Aviso que ya se puede hacer otra conversion

## Master:

**setup();**

(Ningún canal)--> analogico

(canales RX-C4, SDI-C4)→como entrada

(demás canales)→Digitales y como salida

Se activan bits de interrupciones→GIE, PEIE

Configuración del SPI: (MASTER con Fosc / 4, DATA Sample Middle, Clock Idle Low , Idle 2 Active)

Inicializar LCD

Confiuracion de UART

**LoopPrincipal();**

[Para los 3 Salves se hace lo siguiente consecutivamente]

Poner en 0 el SS deseado

Delay

Enviar dato cualquiera

Recibir dato→guardarlo en variable

Delay

Poner en 1 el SS utilizado

[Repetir con los demás]

    Seleccionar primera posición del LCD

    Lcd\_Write\_String(" S1: S2: S3:");

    SendString("S1: "); /Se envia al UART

    WriteNumber(2, 1, Sensor1, 1); //Escribir en la posición 1 de abajo valor del sensor 1, y enviar por el UART valor el sensor 1

    SendString(" S2: ");/Se envia al UART

    WriteNumber(2, 8, Sensor2, 0); //Escribir en la posición 8 de abajo valor del sensor 2, y enviar por el UART valor el sensor 2

    SendString(" S3: ");/Se envia al UART

    WriteNumber(2, 13, Sensor3, 3); //Escribir en la posición 13 de abajo valor del sensor 3, y enviar por el UART valor el sensor 3

for (int i = 0 ; i < 27 ; i++) { //Se hace un for para borrar 27 caracteres de la terminal y poder volver a escribir los nuevos valores cada vez y que la terminal no se llene.

SendChar(BACKSPACE);

[Para función WriteNumber]

/Se toma la fila, columna, valor y formato requerido

/Según formato requerido manipular valor requerido

Para temperatura:

If(valor menor a equivalente digital de 0)

Restar equivalente digital – valor

If(valor mayor a equivalente digital de 0)

Restar valor – equivalente digital

Se separan variables en cifras distintas dividiendo entre 100

Restando lo restante

Dividir entre 10

Restar lo restante

Tomar ultima cifra

Posicionar LCD en fila y columna requerida

Enviar primer cifra

Posicionar LCD en fila requerida y columna requerida + 1

Enviar segunda cifra

[Repetir hasta enviar todo]

[Enviar caracteres extra dependiendo del formato, por ejemplo puntos decimales y “V” como voltaje]

**EN LA SIGUIENTE PAGINA SE ENCUENTRA EL DIAGRAMA DE FLUJO**

