**CAPÍTULO 3**

**3 DISEÑO E IMPLEMENTACIÓN**

**3.1 Diagrama de bloques del Proyecto**

El circuito del medidor está constituido por 4 partes principales que realizan sistemáticamente cada operación necesaria para obtener el valor de la capacitancia o inductancia conectada al circuito. A continuación se detalla el diagrama de bloques explicando el mismo:

µControlador

Oscilador LM311

Oscilador 7555

Pantalla LCD

**Figura 7: Diagrama de bloques del proyecto**

**3.2 Funcionamiento del Proyecto**

El programa inicia con la asignación de constantes y de variables así como la definición de la rutina de interrupción externa que se usa aquí, posterior a esto realiza la autocalibración de ambos circuitos de oscilación.

Para calibrar el oscilador con LM311 toma dos frecuencias, una con el capacitor C y la inductancia L y luego cierra el relé que pone en paralelo al capacitor de calibración Ccal, luego de esto calcula el valor de C y L y posteriormente procede a realizar la calibración del circuito 555 misma que procede de la misma manera usando un relay de conmutación con un capacitor de mas capacidad para calcular los valores de C2 y R del circuito, luego de esto se coloca como 000pf.

El siguiente paso con el oscilador LM311 es el valor de la capacitancia; si esta resulta ser menor de la que el circuito resonante necesita para oscilar es decir el equivalente a un capacitor de 20 nF, conmuta este capacitor para ser medido con el circuito oscilador 555, no sin antes haber preguntado por el modo si es Capacitancia o Inductancia, enseguida el 555 calcula su valor de capacitancia en base a la frecuencia, si la frecuencia es menor a un valor equivalente a un capacitor de 10 uf esta se conmuta no a medir la frecuencia sino el tiempo en alto esto se logra por medio de la interrupción externa y un oscilador de 1 Khz externo, para conveniencia y precisión se logra con el PIC de gama baja 12F629.

**3.3 Diagrama de flujo del algoritmo**

Proceso de inicializacion y autocalibracion

C=1?

No

Presento inductancia

Si

Cap>20nf

No

Presento capacitancia

Si

Conmuto a circuito 555

Presento capacitancia

Cap>10uf

SiNo

Conmuto a medidor de tiempo en alto

Presento capacitancia

**3.4 Código de programación PCW CCS**

#include <18F4520.h>

#fuses HS,NOWDT,NOPROTECT,NOLVP

#use delay(clock=20000000)

#bit t1\_overflow=0x0C.0

#include <lcd.c>

#bit TMRIF=0XF9E.0

#byte portb=6

#byte portc=9

int1 flagToggleFlanco=0, flagHayDatos=0;

int16 t1=0x00, t2=0x00, tt=0x00;

int ciclos8, ciclos;

int8 digito0, digito1, digito2, digito3, decimal0, decimal1;

int32 mod1, mod2, mod3, mod4, mod5, mod6;

int32 freq, freq2, cap, cap2, microcap;

float capf, frecf, frecf2, capacitor, microcapac, microinduc;

float inductor, frecal1, frecal2, capacitor2, k1, k2, capcal;

float inducal, capacitorfix, capacitorfix2, resiscal;

int16 freqc\_high;

int16 freqc\_low;

#int\_ext

void handle\_ext\_int(){

if(flagToggleFlanco==0){

t1=get\_timer1();

ext\_int\_edge(0,H\_TO\_L);

flagToggleFlanco=1;}

else {

t2=get\_timer1();

ext\_int\_edge(0,L\_TO\_H);

flagToggleFlanco=0;

set\_timer1(0);

if(flagHayDatos==0)

{flagHayDatos=1;

}} }

void obtener\_freq()

{

ciclos8=0;

ciclos=0;

freqc\_high=0;

TMRIF=0;

set\_timer1(0);

setup\_timer\_1(T1\_EXTERNAL|T1\_DIV\_BY\_1);

while (ciclos!=0xFF)

{

while (ciclos8!=0xFF)

{

if (TMRIF)

{

TMRIF=0;freqc\_high++;

}

else

{

delay\_cycles(5);

}

delay\_cycles(62);

ciclos8++;

}

delay\_cycles(216);

ciclos++;

}

delay\_cycles(211);

setup\_timer\_1(T1\_DISABLED);

if (TMRIF)

{

freqc\_high++;

}

freqc\_low=get\_timer1();

freq=make32(freqc\_high,freqc\_low);

freq2=freq;

frecf=freq;

frecf=frecf/1000;

}

void calibrando()

{

lcd\_putc('\f');

lcd\_putc(" calibrando 311");

delay\_ms(100);

obtener\_freq();

frecal1=freq;

frecal1=frecal1/1000;

output\_bit(PIN\_E0,1);

delay\_ms(100);

obtener\_freq();

frecal2=freq;

frecal2=frecal2/1000;

frecal1=frecal1\*frecal1;

frecal2=frecal2\*frecal2;

output\_bit(PIN\_E0,0);

capcal=(capacitorfix\*frecal2)/(frecal1-frecal2);

inducal=(k1/capacitorfix)\*((1/(frecal2))-(1/(frecal1)));

}

void calibrando2()

{

lcd\_putc('\f');

lcd\_putc("Medidor LC ");

lcd\_putc('\n');

lcd\_putc("ESPOL V 1.0.3");

delay\_ms(100);

obtener\_freq();

frecf2=frecf;

}

void main()

{

delay\_ms(10);

set\_tris\_a(0xff);

set\_tris\_e(0x08);

setup\_adc\_ports(NO\_ANALOGS);

setup\_adc(ADC\_OFF);

setup\_spi(FALSE);

setup\_psp(PSP\_DISABLED);

output\_bit(PIN\_E0,0);

output\_bit(PIN\_E1,1);

output\_bit(PIN\_E2,1);

lcd\_init();

setup\_timer\_1(T1\_EXTERNAL | T1\_DIV\_BY\_1);

port\_b\_pullups(FALSE);

k1=0.02533029;

k2=1.443;

capacitorfix=0.00104;

capacitorfix2=1.00;

calibrando();

output\_bit(PIN\_E1,0);

output\_bit(PIN\_E2,0);

delay\_ms(25);

calibrando2();

output\_bit(PIN\_E1,1);

output\_bit(PIN\_E2,1);

delay\_ms(200);

while (true)

{

disable\_interrupts(int\_ext);

disable\_interrupts(global);

output\_bit(PIN\_E1,1);

if (input(pin\_a1))

{

obtener\_freq();

frecf=frecf\*frecf;

capf=(k1/(frecf\*inducal))-capcal;

capacitor=capf\*1000000;

microcapac=capacitor\*100;

freq=capacitor;

microcap=microcapac;

}

else

{

obtener\_freq();

frecf=frecf\*frecf;

capf=(k1/(frecf\*capcal))-inducal;

inductor=capf\*1000000;

microinduc=inductor\*100;

freq=inductor;

microcap=microinduc;

}

cap=freq;

if (cap>999)

{

if (cap>9999||freq2<1)

{

if (input(pin\_a1))

{

do

{

output\_bit(PIN\_E1,0);

delay\_ms(50);

obtener\_freq();

resiscal=14.1\*frecf;

capf=k2/resiscal;

capacitor2=capf\*100000;

if (capacitor2>99999||freq2<1)

{

if (capacitor2>999999||freq2<1)

{

output\_bit(PIN\_E2,0);

setup\_timer\_1(T1\_EXTERNAL|T1\_DIV\_BY\_1);

set\_timer1(0x0000);

disable\_interrupts(global);

disable\_interrupts(int\_timer1);

disable\_interrupts(int\_rda);

disable\_interrupts(int\_ext);

ext\_int\_edge(0,L\_TO\_H);

flagToggleFlanco=0;

enable\_interrupts(int\_rda);

enable\_interrupts(int\_ext);

enable\_interrupts(global);

do

{

cap=1;

if(flagHayDatos==1)

{

if(t2>t1)

{

tt=t2-t1;

capf=tt;

frecf=capf\*40/(260\*frecf2);

cap=frecf;

mod1=cap/10;

mod2=cap/100;

mod3=cap/1000;

mod4=cap/10000;

mod5=cap/100000;

mod6=cap/1000000;

decimal1=(cap%10)+48;

decimal0=(mod1%10)+48;

digito0=(mod2%10)+48;

digito1=(mod3%10)+48;

digito2=(mod4%10)+48;

lcd\_gotoxy(1,1);

lcd\_putc("C =");

lcd\_putc(digito2);

lcd\_putc(digito1);

lcd\_putc(digito0);

lcd\_putc(decimal0);

lcd\_putc(decimal1);

lcd\_putc(" uf");

}

}

}

while (cap<5300&&cap!=0);

disable\_interrupts(global);

disable\_interrupts(int\_timer1);

disable\_interrupts(int\_rda);

disable\_interrupts(int\_ext);

output\_bit(PIN\_E2,1);

output\_bit(PIN\_E1,1);

}

else

{

freq=capacitor2/1000;

}

}

else

{

freq=capacitor2;

}

cap2=freq;

mod1=cap2/10;

mod2=cap2/100;

mod3=cap2/1000;

mod4=cap2/10000;

decimal1=(cap2%10)+48;

decimal0=(mod1%10)+48;

digito0=(mod2%10)+48;

digito1=(mod3%10)+48;

digito2=(mod4%10)+48;

lcd\_putc('\f');

lcd\_putc("C =");

lcd\_putc(digito2);

lcd\_putc(digito1);

lcd\_putc(digito0);

lcd\_putc(".");

lcd\_putc(decimal0);

lcd\_putc(decimal1);

if (capacitor2>99999)

{

lcd\_putc(" uf");

}

else

{

lcd\_putc(" nf");}

output\_bit(PIN\_E1,0);

delay\_ms(20);

}

while (capacitor2>1000);

output\_bit(PIN\_E1,1);

}

else

{

if (cap>399999)

{

lcd\_putc('\f');

lcd\_putc("Fuera de rango");

}

else

{

mod2=cap/100;

mod3=cap/1000;

mod4=cap/10000;

mod5=cap/100000;

mod6=cap/1000000;

decimal0=(mod1%10)+48;

digito0=(mod2%10)+48;

digito1=(mod3%10)+48;

digito2=(mod4%10)+48;

digito3=(mod5%10)+48;

lcd\_putc('\f');

lcd\_putc("L =");

lcd\_putc(digito3);

lcd\_putc(digito2);

lcd\_putc(digito1);

lcd\_putc(".");

lcd\_putc(digito0);

lcd\_putc(decimal0);

lcd\_putc(" mh");

}

}

}

else

{

mod2=cap/100;

mod3=cap/1000;

mod4=cap/10000;

mod5=cap/100000;

mod6=cap/1000000;

decimal0=(mod1%10)+48;

digito0=(mod2%10)+48;

digito1=(mod3%10)+48;

digito2=(mod4%10)+48;

digito3=(mod5%10)+48;

lcd\_putc('\f');

if (input(pin\_a1))

{

lcd\_putc("C =");

}

else

{

lcd\_putc("L =");

}

lcd\_putc(digito3);

lcd\_putc(digito2);

lcd\_putc(digito1);

lcd\_putc(".");

lcd\_putc(digito0);

lcd\_putc(decimal0);

if (input(pin\_a1))

{

lcd\_putc(" nf");

}

else

{

lcd\_putc(" mh");

}

}

}

else

{

if (input(pin\_a1))

{

mod1=cap/10;

mod2=cap/100;

decimal1=(cap%10)+48;

decimal0=(mod1%10)+48;

digito0=(mod2%10)+48;

lcd\_putc('\f');

lcd\_putc("C =");lcd\_putc(digito0);

lcd\_putc(decimal0);

lcd\_putc(decimal1);

lcd\_putc(" pf");

}

else

{

mod1=microcap/10;

mod2=microcap/100;

mod3=microcap/1000;

mod4=microcap/10000;

decimal1=(microcap%10)+48;

decimal0=(mod1%10)+48;

digito0=(mod2%10)+48;

digito1=(mod3%10)+48;

digito2=(mod4%10)+48;

lcd\_putc('\f');

lcd\_putc("L =");

lcd\_putc(digito2);

lcd\_putc(digito1);

lcd\_putc(digito0);

lcd\_putc(".");

lcd\_putc(decimal0);

lcd\_putc(decimal1);

lcd\_putc(" uh");

}

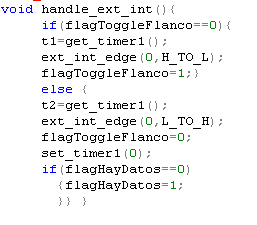
}

}

}

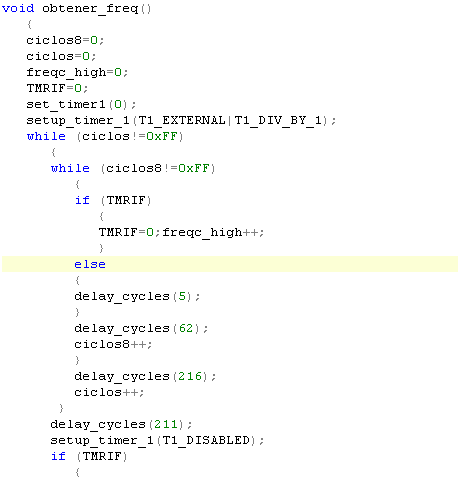
**3.4.1 Funciones usadas y dinámica del programa**

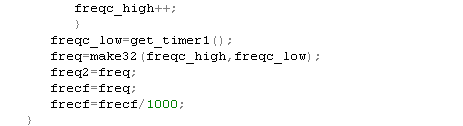
**3.4.1.1 Manejo de la interrupción externa**

****

La primera de las funciones encontradas es la de manejo de la interrupción externa por parte de la pata RB0 la cual es usada en el modo de conteo de tiempo en alto y sirve para determinar el tiempo en alto de la señal emitida por el oscilador 7555, lo que concretamente hace esta función es cambiar el flanco de detección de la interrupción externa.

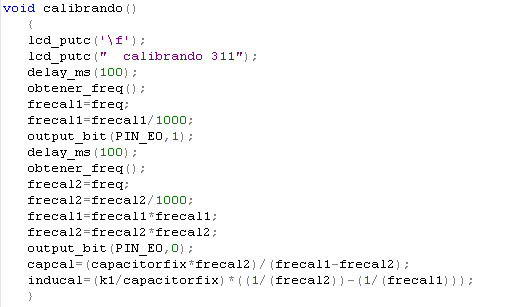
**3.4.1.2 La función Obtener frecuencia**

****

****

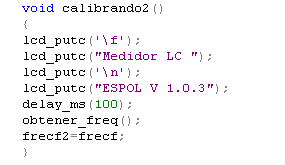
La función obtener frecuencia es la base de todo el programa pues calcula la frecuencia en base al contador del timer 1 misma que está gobernada por un lazo de un segundo de duración entregando la frecuencia en hercios para freq y en kilohercios para frecf.

**3.4.1.3 La función Calibrando**



Esta función es la que calcula dos frecuencias, una de oscilación inicial con los componentes L y C un inductor y un capacitor de tolerancia aceptable y luego otra, puesta en paralelo con un capacitor de alta calidad que sirve para realizar los cálculos para obtener los valores de L y C que se usaran posteriormente en el programa inicial.

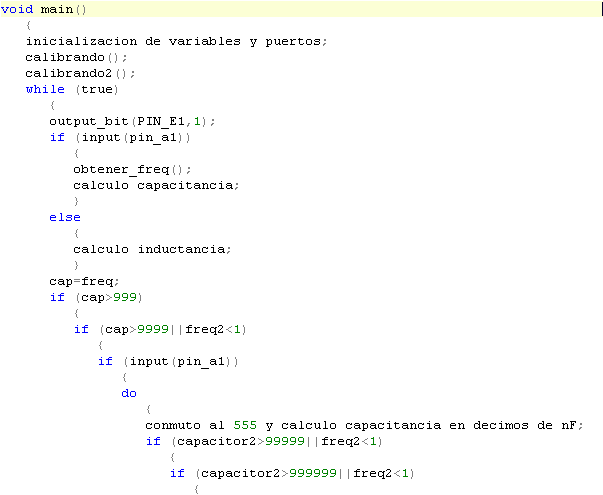
**3.4.1.4 La función Calibrando 2**

****

Esta función logra calcular con precisión el oscilador que da la cuenta de tiempo en alto del oscilador 7555 cuando se usa en modo interrupción externa es decir ajusta el reloj de 1 Khz en caso de que no fuera así guardando este valor en frecf2.

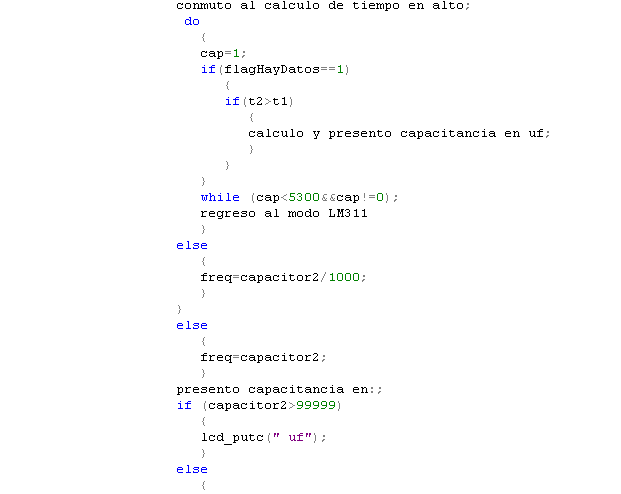
**3.5 El programa principal**

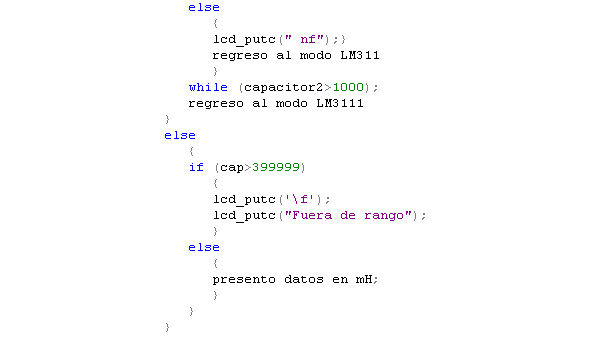
Vamos a centrarnos en analizar las partes más importantes del programa principal expuesto anteriormente en orden desimplificar su comprensión y dado que arriba ya está expuesto vamos a resumir rutinas con nombres.



Como vemos al inicio del programa se llama a las funciones de calibración y se realiza la inicialización de las interrupciones y de los puertos, es entonces cuando se llama a la función de obtener frecuencia, con esta frecuencia de oscilación y en función únicamente del pin de selección L/C se calcula bien la capacitancia o la inductancia del circuito ahora bien si este valor llega a ser mayor que 999 es obviamente ahí donde comienza nuestro primer autorango pues luego de esto ya pasa al rango de los nanofaradios o microhenrios según se dé el caso así que comienza nuestro primer lazo; si, luego de esto y tomando en consideración que con un capacitor de 20 nf el oscilador LM311 empieza a presentar problemas de falta de oscilación conmutamos el capacitor al circuito oscilador 7555 que se comporta de mejor manera con capacitores más grandes.

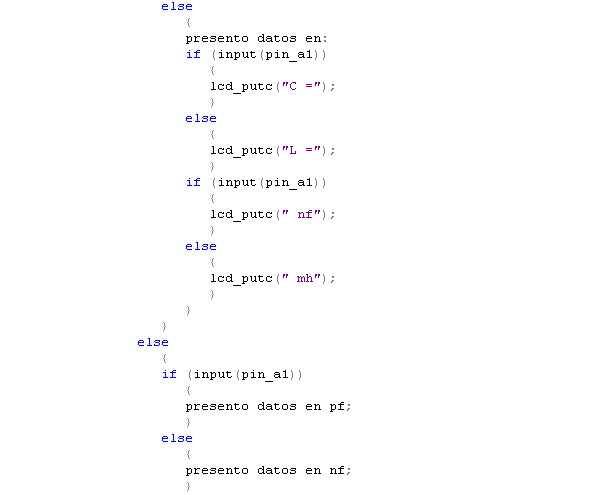
Este cambio de circuito se logra gracias al relé conectado a la salida de la pata E1 el mismo que conmuta el valor. Ahora bien, nuevamente nos topamos con un cálculo de capacitancia pues este es un nuevo oscilador por lo que llamamos a obtener frecuencia nuevamente y calculamos con la nueva fórmula, esta nos lo entrega en décimas de nanofaradio, pues bien si este valor supera los 999.9 nanofaradios tenemos que llamar al segundo rango y pasamos a presentar en microfaradios.





Pues bien resulta que el cálculo de la función obtener frecuencia es válido hasta que la frecuencia sea mayor o cercana a 1 hercio luego de lo cual esta se vuelve completamente inútil para calcular capacitancia, pues no calcula décimas ni centésimas de hercio, si bien el oscilador 7555 no ha parado de oscilar simplemente su frecuencia se ha vuelto indetectable para la función por eso al llegar al valor crítico de 10 uF o 9999.9 nanofaradios llamamos al tercer modo de cálculo.

El nuevo cálculo no realiza conmutación de circuito para el capacitor mas sí lo realiza para el timer 1 que ahora recoge la frecuencia de 1 Khz que da la temporización para que el contador del programa pueda calcular el tiempo en alto por medio de la interrupción externa, esto lo hará hasta un máximo teórico de 5300 uF, si excede este valor o si se vuelve más pequeño que 10 uF volverá al modo principal de oscilación por el LM311, luego de esto se presentarán los datos calculados en el formato que corresponda a si son capacitores y de que rango de eso se encargan los lazos finales.



**3.6 Operación del instrumento**

El instrumento consta de dos interruptores y un botón.

Para encender el instrumento el interruptor marcado como encendido tiene que estar hacia la derecha, el botón etiquetado como 3 pondrá al circuito a calibrar teniendo que encenderse siempre en el modo C del interruptor L/C o en caso contrario si el interruptor se encuentra en el modo L los lagartos de medición deben estar cortocircuitados para poder ser calibrado.

****

**Figura 8: Interruptores de selección del circuito**

Para proceder con la autocalibración del circuito nuevamente se presiona el botón marcado como calibración botón 3 que pondrá el circuito a calibrarse tomando en consideración que el interruptor LC debe ser manejado como en el encendido.