Descripción breve

Elementos, cosas y pasos que necesitamos para poder realizar la parte de esta materia en el proyecto integrado

Proyecto integrador 4 semestre

Sistemas Embebidos

Equipo 4:

Amayelli Itzel Silva Contreras

José Luis Garza Gallegos

Kevin Roberto Gómez Peralta

Edsel Barbosa Gonzalez

**Introducción:**

Actualmente en la Universidad de Colima se nos ha pedido desarrollar un sistema embebido el cual calcule el caudal de agua que corre en la Facultad de Telemática, así mismo mandara esta información a un servidor FTP o HTTP para después ser mostrado en una página web, aunado a esto si el caudal de agua muestra una anomalía se enviara una alerta mediante correo electrónico al encargado del sistema.

Los micro controladores son un componente que se ha vuelto esencial en las últimas décadas, incluso a pesar de que ya existen computadoras mucho más avanzadas, los micro controladores siguen siendo necesarios porque hay aparatos que no necesitan más capacidades que las que éstos brindan, por eso siguen siendo la mejor opción. Realizamos esta práctica con el fin de comenzar a usar un microntrolador y probarlo con un pequeño conjunto de instrucciones que se introducirán con una computadora.

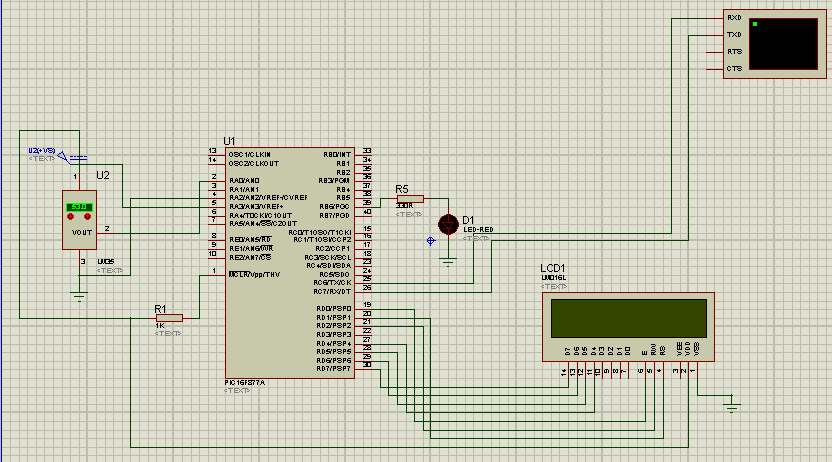
Para lo cual optamos por el desarrollo de este sistema en protoboard donde implementamos el circuito, al ser baratas y confiables; además de sensores no industriales pero confiables, de procedencia china.

**Desarrollo**

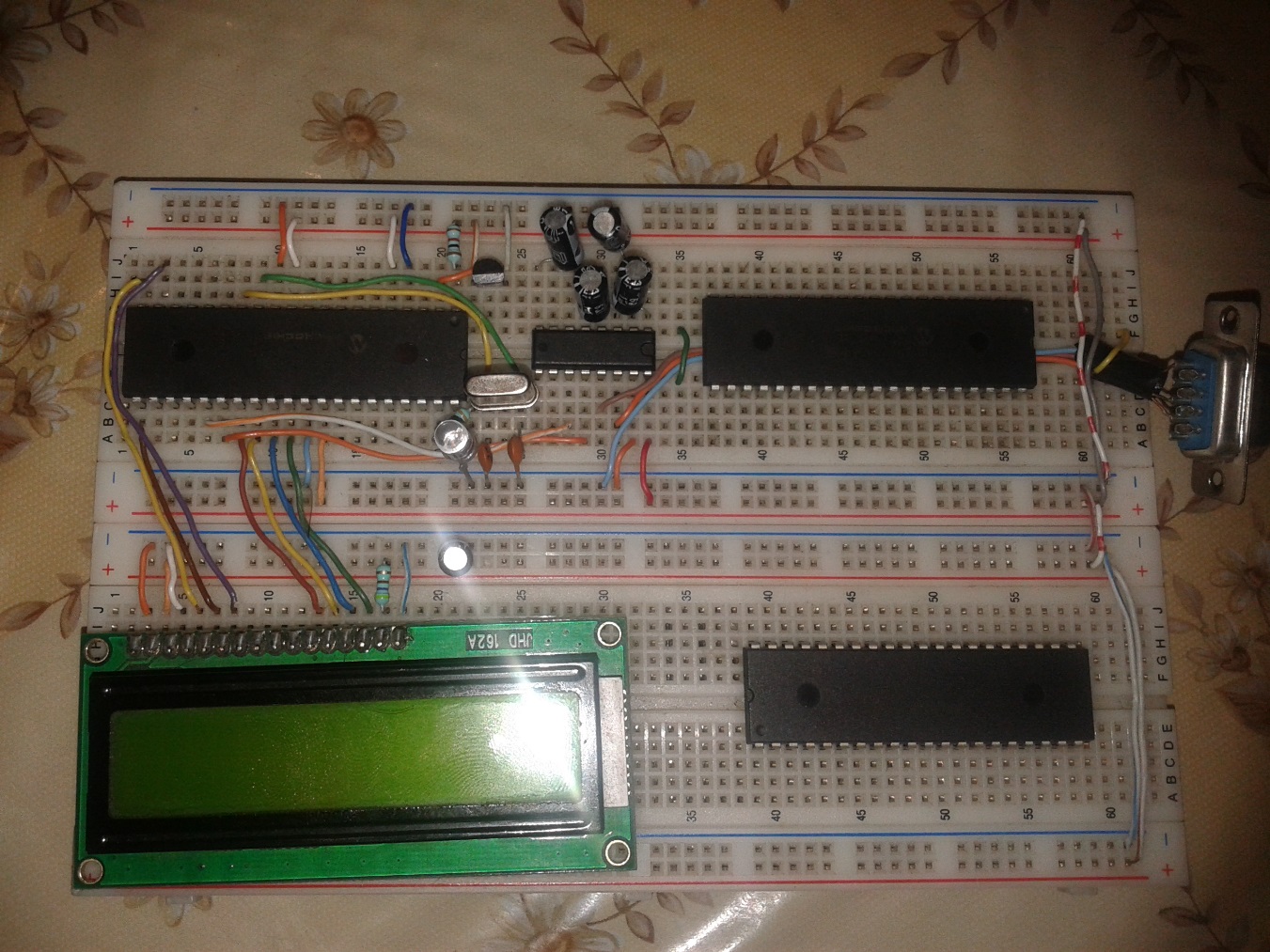
Antes de empezar lo que hicimos fueron pruebas, la cual consistía en cablear la protoboard con los elementos necesarios y todo, como que si ya fuera la entrega final, pero en lugar de sensores de flujo de agua, un sensor de temperatura, para lo cual necesitamos los siguientes materiales:

* PIC 16f
* Protoboard
* 1 metro de cable UTP
* Max 232
* Display JHD 162ª
* 1 led
* Un cristal de cuarzo de 4MHz
* Dos capacitores cerámicos de 27pf o 33 pf
* 1 resistencia de 1 KOhms
* 1 resistencia de 330 Ohms
* 5 capacitores de 40 mf Db9 hembra
* Cable Db9 macho a USB

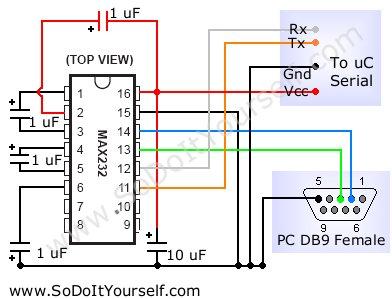
Este fue el cableado que seguimos:



El resultado es el siguiente:



Y para la conexión de Max232 con el db9 hembra es el siguiente:

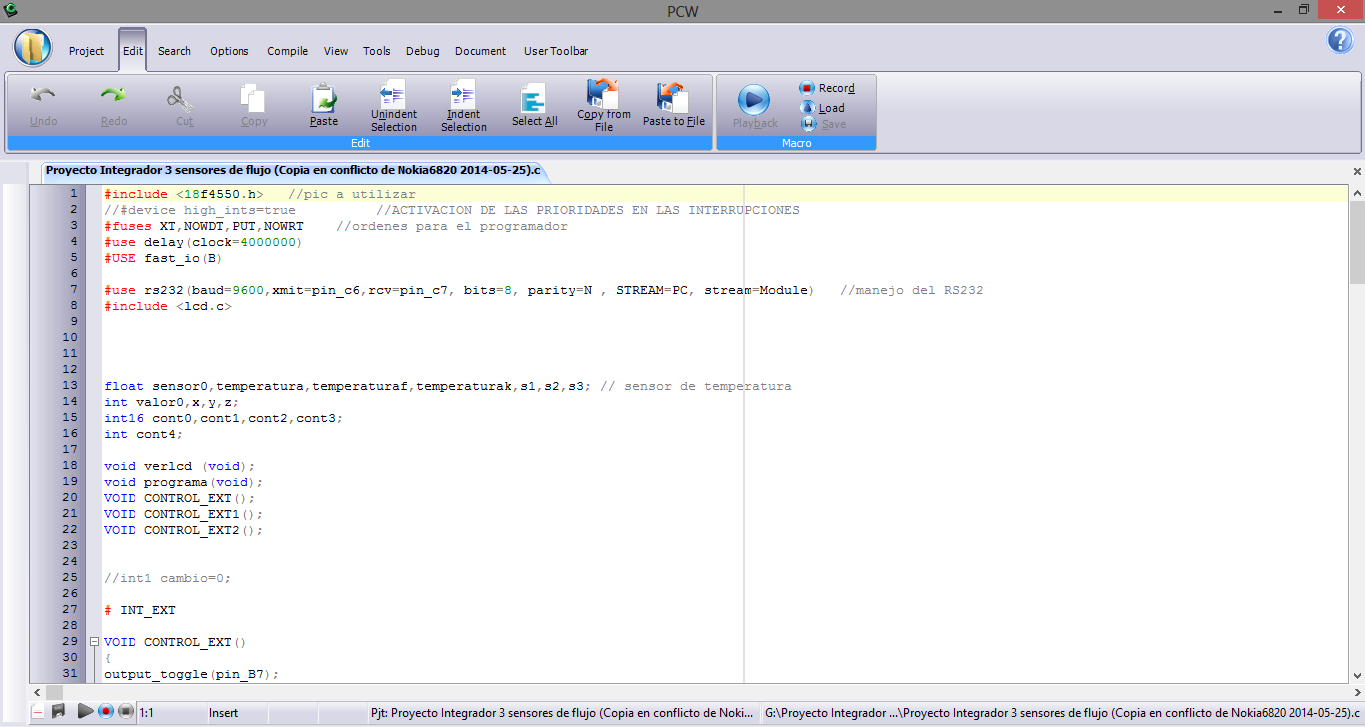


Al principio todo salió correcto con la práctica del sensor de temperatura, los datos eran mostrados en el display y en la computadora.

Después de una semana nos llegaron los sensores de flujo procedentes de china, por lo tanto procedimos a programar el PIC para las lecturas de estos.

Cuando quisimos conectar los 3 sensores notamos un error en el PIC, este no permitía la lectura de más de 1 sensor, por ende tuvimos que comprar otro PIC más caro: “18f4550”.

Una vez comprado este PIC, ahora si lo programamos con sus especificaciones, para la programación lo hicimos en el lenguaje “C”, y para esto necesitamos un programa llamado “C compiler IDE”



#include <18f4550.h> //pic a utilizar

//#device high\_ints=true //ACTIVACION DE LAS PRIORIDADES EN LAS INTERRUPCIONES

#fuses XT,NOWDT,PUT,NOWRT //ordenes para el programador

#use delay(clock=4000000)

#USE fast\_io(B)

#use rs232(baud=9600,xmit=pin\_c6,rcv=pin\_c7, bits=8, parity=N , STREAM=PC, stream=Module) //manejo del RS232

#include <lcd.c>

float sensor0,temperatura,temperaturaf,temperaturak,s1,s2,s3; // sensor de temperatura

int valor0,x,y,z;

int16 cont0,cont1,cont2,cont3;

int cont4;

void verlcd (void);

void programa(void);

VOID CONTROL\_EXT();

VOID CONTROL\_EXT1();

VOID CONTROL\_EXT2();

//int1 cambio=0;

# INT\_EXT

VOID CONTROL\_EXT()

{

output\_toggle(pin\_B7);

delay\_ms (1);

cont0=cont0+1;

if (cont0>450)

{

cont0=0;

}

}

# INT\_EXT1 //FUNCION DE LA INTERRUPCION EXT1, PRIORIDAD ALTA

VOID CONTROL\_EXT1()

{

output\_toggle(pin\_B6);

delay\_ms (1);

cont1=cont1+1;

if (cont1>450)

{

cont1=0;

}

} // Detectar puertos B2, B1, B0.

# INT\_EXT2 //FUNCION DE LA INTERRUPCION EXT1, PRIORIDAD ALTA

VOID CONTROL\_EXT2()

{

output\_toggle(pin\_B5);

delay\_ms (1);

cont2=cont2+1;

if (cont2>450)

{

cont2=0;

}

}

# INT\_TIMER1

void TIMER1S(void)

{

cont4=cont4+1;

if (cont4>1)

{

cont4=0;

set\_timer1 (6072);

}

//delay\_ms (50);

if (cont4==1)

{

output\_toggle(pin\_B4);

x=cont0;

cont0=0;

y=cont1;

cont1=0;

z=cont2;

cont2=0;

}

}

void verlcd (void)

{

set\_adc\_channel (0);

delay\_ms(30); //para que se estabilice

sensor0=read\_adc (0);

delay\_ms(10); //para que se estabilice

temperatura=(sensor0\*.48);

//temperaturaf=((temperatura\*1.8)+32);

//temperaturak=temperatura+273;

// Leer el canal AN2 esta entrada es voltaje de referencia (GND)

set\_adc\_channel (2);

delay\_ms (1);

//medicion2=read\_adc ();

// Leer el canal AN3 esta entrada es voltaje de referencia (VCC)

set\_adc\_channel (3);

delay\_ms (1);

//Lcd\_putc( "\f");

delay\_ms(1);

lcd\_gotoxy(1,1);

printf(lcd\_putc,"\%1.1f dl - %1.1f dl" s1,s2);

printf(lcd\_putc,"\n%1.1f dl" s3);

//printf(lcd\_putc,"\n cont=%lu cont4=%d" cont0,cont4);

delay\_ms(1);

printf("\f%1.1f,%1.1f,%1.1f" ,s1,s2,s3);

}

void main(void)

{

set\_tris\_B (0x07);

output\_low(PIN\_B7);

port\_b\_pullups(TRUE);

enable\_interrupts(int\_ext);

enable\_interrupts(int\_ext1); //INT\_EXT1 HABILITADA

enable\_interrupts(int\_ext2); //INT\_EXT2 HABILITADA

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

ext\_int\_edge(1,L\_TO\_H);

ext\_int\_edge(2,L\_TO\_H);

setup\_timer\_1(T1\_INTERNAL | T1\_DIV\_BY\_4);

set\_timer1 (6072);

enable\_interrupts(INT\_TIMER1);

enable\_interrupts(GLOBAL);

//setup\_adc\_ports(AN0\_AN1\_AN2\_AN4\_AN5\_AN6\_AN7\_VSS\_VREF);

//setup\_adc(ADC\_CLOCK\_INTERNAL);

lcd\_init();

valor0=0;

sensor0=0;

temperatura=0;

cont0=0;

cont1=0;

cont2=0;

cont3=0;

cont4=0;

while (1)

{

//programa();

//delay\_ms (1);

s1=((x/2)\*(60));

s1=s1/45;

s2=((y/2)\*(60));

s2=s2/45;

s3=((z/2)\*(60));

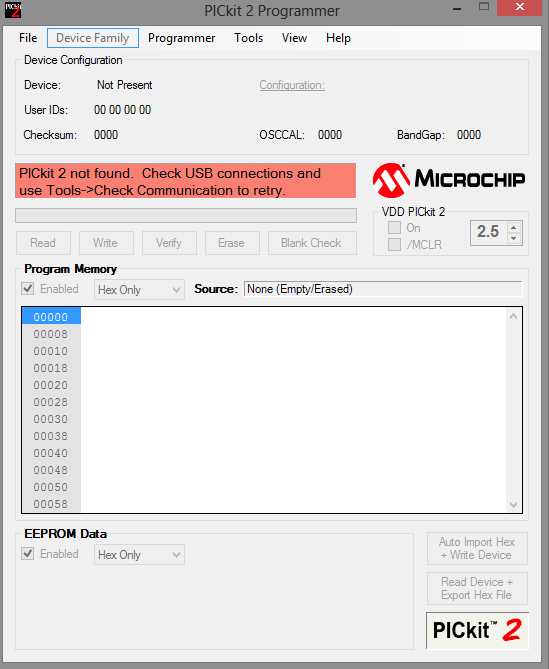
s3=s3/45;

verlcd ();

}

}

Una vez terminado el código, necesitamos de otro programa, con el cual el código que nosotros hicimos lo convertiremos a código máquina y se guarda en el PIC 16f4550



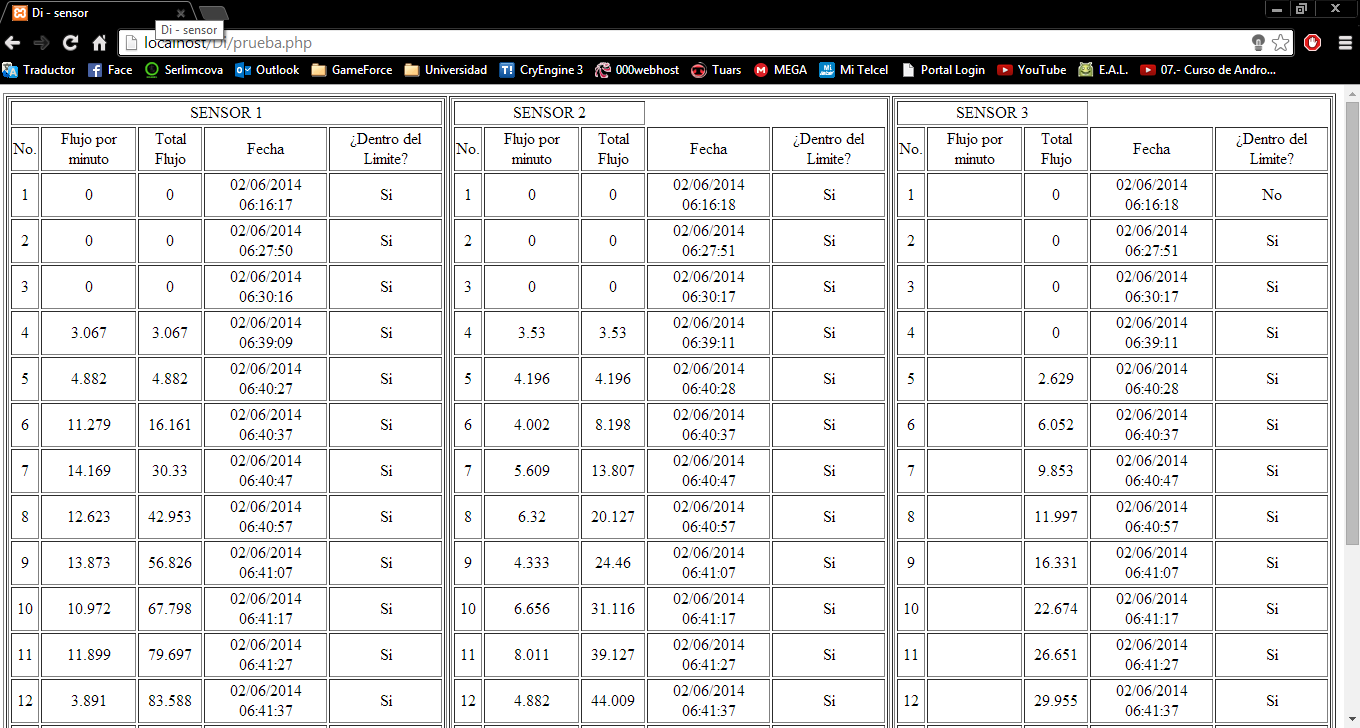
Los sensores son los siguientes y estos se conectan en las patas R0, R1, y R2 del pic:



Posteriormente desarrollamos un programa el cual obtiene los datos del Max232 y guarda dichos valores en una base de datos en MySql:



Y por último, la página web muestra los datos de la base de datos en MySql:



**Conclusión**

Este circuito fue un poco fácil de realizar, una vez que tuvimos la manera de como conectar todos los componentes, además la lectura de los datos en el display nos ayudó mucho al momento de realizar prueba para calar los sensores y antes de empezar toda la operación por el lado de la computadora, servidor y pagina web