



**UNIVERSIDAD DEL AZUAY**

**FACULTAD DE CIENCIA Y TECNOLOGÍA**

**ESCUELA DE INGENIERÍA ELECTRÓNICA**

**“CONSTRUCCIÓN DE UN MEDIDOR DE PH, CONDUCTIVIDAD  
Y OXÍGENO DISUELTO EN AGUA, PARA RÍOS Y LAGUNAS”**

**Trabajo de graduación previo a la obtención del título de Ingeniero  
Electrónico**

**Autores:**

**Pedro Miguel Urigüen Eljuri**

**Juan Patricio González Brito**

**Director:**

**Freddy Gonzalo Pesántez Díaz**

**Cuenca, Ecuador**

**2013**

*Handwritten signature and date:*  
24/04/13

## CONSTRUCCIÓN DE UN MEDIDOR DE PH, CONDUCTIVIDAD Y OXÍGENO DISUELTO EN AGUA, PARA RÍOS Y LAGUNAS.

### RESUMEN

Para diseñar el software, hardware y realizar las pruebas del equipo ya sea en el laboratorio o pruebas de campo; Adquirimos y procesamos señales desde una sonda YSI, empleando un microcontrolador PIC18F4620, el cual visualiza y graba los datos en la memoria MicroSD.

El equipo consta de tres bloques principales: comunicación con la sonda, interacción con el usuario y manejo de archivos en la memoria MicroSD.

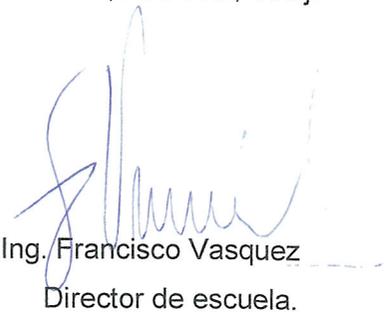
La programación del software fue desarrollada en lenguaje C y el diseño de hardware se desarrolló en Altium Designer. El equipo fue probado y homologado con un equipo YSI 6500, y actualmente está en uso en óptimas condiciones.

### Palabras clave:

Lenguaje C, sonda multiparámetros, PIC18F4620, adquisición de datos, MicroSD, reloj.



Ing. Freddy Pesántez Díaz  
Director de tesis.



Ing. Francisco Vasquez  
Director de escuela.



Juan Patricio González B.  
Autor.



Pedro Urigüen Eljuri  
Autor.

r.

*Handwritten signature:* Juco

## ABSTRACT

### CONSTRUCTION OF A PH, CONDUCTIVITY, AND OXYGEN MEASURING DEVICE FOR RIVERS AND LAKES

In order to design the software, hardware, and to carry out laboratory or field tests, we acquired and processed signals from a YSI probe through a PIC18F4620 microcontroller, which visualizes and records the data in the MicroSD memory.

The equipment consists of three main blocks: communication with the probe, interaction with the user, and management of the files through the MicroSD memory.

The software programming was developed in language C while the design was developed in Altium Designer. The equipment was tested and homologated with YSI 6500 equipment and is currently in use.

**Key Words:** Language C, Multi-parameter probe, PIC18F4620, data gathering, MicroSD, clock.



Ing. Freddy Pesantez Díaz  
Thesis Director



Ing. Francisco Vásquez  
School Director



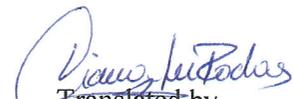
Juan Patricio González B.  
Author



Pedro Uriguen Eljuri  
Author



UNIVERSIDAD DEL  
AZUAY  
DPTO. IDIOMAS



Translated by,  
Diana Lee Rodas

## ÍNDICE DE CONTENIDOS

<b>RESUMEN</b>	ii
<b>ABSTRACT</b>	iii
<b>ÍNDICE DE CONTENIDOS</b>	iv
<b>ÍNDICE DE FIGURAS</b>	vii
<b>INTRODUCCIÓN</b>	1

### **CAPITULO 1: ANÁLISIS DE PROBLEMA Y MATERIALES A USAR**

1.1	Análisis del Problema	2
1.2	Conceptos	3
1.2.1	El pH	3
1.2.2	El electrodo de pH	4
1.2.3	Oxígeno Disuelto	4
1.2.4	Conductividad del agua	5
1.2.4.1	Conductividad eléctrica (EC)	5
1.3	Análisis de Materiales a usar	5

### **CAPÍTULO 2: ANÁLISIS DEL FUNCIONAMIENTO DE LA SONDA Y DISEÑO DE SOFTWARE PARA EL MICRO CONTROLADOR**

2.1	Introducción	8
2.2	Análisis del funcionamiento básico de la sonda	9
2.2.1.1	Efecto de la temperatura en la conductividad	9
2.2.2	Salinidad	10
2.2.3	pH	10
2.2.3.2	Calibración y Efectos de la temperatura	11
2.2.4	Temperatura	12
2.2.5	Oxígeno Disuelto	12
2.2.5.3	Principio de funcionamiento	12
2.3	Diseño de Software	13
2.3.1	Adquisición de datos de la sonda, interacción con el usuario	13
2.3.1.1	Crear un proyecto en MikroC PRO	13
2.3.1.2	Declaración de variables para el uso del LCD	14
2.3.1.3	Declaración de variables de uso general	15
2.3.1.4	Inicialización del segmento principal del programa	16

2.3.1.5 Descripción breve de la comunicación con la sonda	17
2.3.1.6 Conexión con la sonda en el segmento principal	19
2.3.1.7 Ingreso al menú principal de la sonda	19
2.3.1.8 Selección de submenú dentro del menú principal	20
2.3.1.9 Submenú RUN	22
2.3.1.10 Fin del submenú RUN	24
2.3.1.11 Submenú Reporte	24
2.3.1.12 Submenú Calibrar	28
2.3.2 Manejo de la memoria SD	31
2.3.2.1 Introducción a la memoria SD y Conexión	32
2.3.2.2 La memoria SD y la comunicación SPI	33
2.3.2.3 Software para inicializar, crear archivos y escribir en la SD	35
2.3.3 Reloj en tiempo real	41
2.3.3.1 Reloj	44
2.3.3.2 Ajuste reloj	45

### **CAPÍTULO 3: DISEÑO DE HARDWARE PARA LA SONDA**

3.1 Introducción	46
3.2 Creación del proyecto en Altium designer y cargar las librerías	47
3.3 Búsqueda de componentes en las librerías	49
3.4 Posicionamiento de bus de datos y Net labels	51
3.4.1 Poner conectores de tierra o voltaje	53
3.5 Posicionar Net Label	54
3.6 Búsqueda de los FootPrints	55
3.7 Designar nombre de componentes	60
3.8 Creación de PCB	62
3.9 Consideraciones especiales para el hardware de la placa	66

### **CAPÍTULO 4: PRUEBAS DE LABORATORIO Y CAMPO. Y FUNCIONALIDAD**

4.1 Introducción	68
4.2 Pruebas de laboratorio y campo	68
4.3 Funcionalidad	69

<b>CONCLUSIONES</b>	71
---------------------	----

<b>BIBLIOGRAFÍA</b>	72
<b>ANEXOS</b>	74

## ÍNDICE DE FIGURAS

Figura 1 Diagrama de bloques de partes del software en MikroC PRO	8
Figura 2 Creación de un nuevo proyecto en MikroC PRO	13
Figura 3 New Project Wizard.	14
Figura 4 Subrutina de inicialización de la pantalla LCD.	14
Figura 5 Conexión microcontrolador y pantalla LCD.	15
Figura 6 Inicialización de variables a ser utilizadas.	16
Figura 7 Inicialización de puertos.	16
Figura 8 Inicialización del LCD y puerto UART.	17
Figura 9 MikroElectronica USART Terminal.	18
Figura 10 Mensaje entregado por la sonda.	18
Figura 11 Conexión con la sonda en el programa principal.	19
Figura 12 Diagrama de bloques del menú principal de la sonda.	20
Figura 13 Selección de un submenú mediante pulsantes.	21
Figura 14 Diagrama de bloques del menú RUN de la sonda.	22
Figura 15 Envío de datos para el ingreso a Run.	23
Figura 16 Bucle de adquisición de datos.	23
Figura 17 Procesamiento a datos adquiridos.	24
Figura 18 Salida del submenú Run.	24
Figura 19 Diagrama de bloques del menú Reporte de la sonda.	25
Figura 20 Ingreso al submenú Reporte.	26
Figura 21 Subrutina reporte1().	26
Figura 22 Subrutina reporte3().	27
Figura 23 Selección de parámetros.	28
Figura 24 Diagrama de bloques del menú Calibrar de la sonda.	28
Figura 25 Selección de parámetros.	29
Figura 26 Estructura jerárquica del Submenú Calibrar.	29
Figura 27 Inicialización de la subrutina valor().	30
Figura 28 Segmento de la subrutina valor, operación para obtener valor.	31
Figura 29 Distribución de pines de la memoria SD.	32
Figura 30 Pines de la SD	33
Figura 31 Esquema básico de una conexión SPI.	34
Figura 32 Diagrama de bloques del funcionamiento de la SD.	35
Figura 33 Declaración de las variables de la memoria SD.	36
Figura 34 Declaración de las variables de uso general para la SD.	37
Figura 35 Inicio del módulo SPI	37
Figura 36 Comprobación de si esta la SD	38
Figura 37 Estructura If agregar líneas al archivo.	38
Figura 38 Rutina de Graba SD.	38
Figura 39 Bits para el MMC_Fat_Assing.	39
Figura 40 Rutina de M_Open_File_Append();	40
Figura 41 Tabla de caracteres especiales para string.	41
Figura 42. Diagrama de bloques del reloj en tiempo real.	42

Figura 43 Conexión realizada para el reloj en tiempo real. _____	43
Figura 44 Lectura de datos del integrado DS1307. _____	44
Figura 45 Decodificación de los datos obtenidos del DS1307. _____	44
Figura 46 Grabar datos en el integrado DS1307. _____	45
Figura 47 Hardware del equipo. _____	46
Figura 48 Creación de un nuevo proyecto en Altium. _____	47
Figura 49 Como agregar un archivo esquemático al proyecto. _____	48
Figura 50 Ventana para agregar Librerías. _____	49
Figura 51 Búsqueda de componentes en la librería. _____	50
Figura 52 Ventana de búsqueda de componentes. _____	50
Figura 53 Icono de poner un Bus. _____	51
Figura 54 Ejemplo de posicionar bus alrededor de los componentes. _____	52
Figura 55 Poner un cable en el circuito. _____	52
Figura 56 Poner una entrada en el Bus de datos. _____	52
Figura 57 Posicionar conector de tierra. _____	53
Figura 58 Posicionar conector de VCC. _____	53
Figura 59 Ventana para cambiar parámetros del conector. _____	53
Figura 60 Posicionar Nel Label. _____	54
Figura 61 Menú para cambiar el nombre del Net Label. _____	54
Figura 62 Ventana de propiedades del componente. _____	55
Figura 63 Ventana Edit del footprint. _____	56
Figura 64 Lista de Footprints disponibles para elegir. _____	57
Figura 65 Opción de Find Similar ítems. _____	58
Figura 66 Ventana de Find Similar Objects _____	59
Figura 67 Ventana para realizar cambios globales. _____	60
Figura 68 Ventana para realizar el annotate de todos los componentes. _____	61
Figura 69 Primera ventana del Wizard para crear un PCB. _____	62
Figura 70 Elección de tipos de componentes para el PCB. _____	63
Figura 71 Placa PCB sin componentes. _____	64
Figura 72 Placa PCb con los componentes antes de posicionarlos. _____	64
Figura 73 Ventana para el Auto Route. _____	65
Figura 74 Diagrama de conexión de la tarjeta SD. _____	66
Figura 75 Foto de Medición 1 con equipo realizado. _____	68
Figura 76 Foto de Medición 1 con equipo de referencia. _____	68
Figura 77 Foto de medición 2 con equipo realizado. _____	69
Figura 78 Foto de medición 2 con equipo de referencia. _____	69

González Brito Juan Patricio

Urigüen Eljuri Pedro Miguel

Trabajo de Graduación

Ing. Freddy Gonzalo Pesántez Díaz

Mayo de 2013

## **CONSTRUCCIÓN DE UN MEDIDOR DE PH, CONDUCTIVIDAD Y OXÍGENO DISUELTO EN AGUA, PARA RÍOS Y LAGUNAS.**

### **INTRODUCCIÓN**

Dentro de los procesos que realiza la escuela de Biología de facultad de Ciencia y Tecnología de la Universidad del Azuay, está el análisis de líquidos respecto a sus diferentes parámetros como son: oxígeno disuelto, pH, conductividad, salinidad, temperatura. Este análisis permite conocer los niveles de contaminación y diferentes características del líquido analizado. Para este análisis se requiere de la obtención y muestreo de datos.

En este trabajo de graduación se desarrolla un analizador multiparámetros que va a ser un equipo homologado que no cumplirá con estándares; Este adquiere y procesa datos, exclusivamente de una sonda YSI 600R. Además de la adquisición y procesamiento de datos obtenidos con un micro controlador PIC18F4620, es un equipo autónomo energéticamente, ya que cuenta con una batería en su interior; los datos además de denotarlos en la pantalla LCD de 20x4, son guardados en una memoria externa MicroSD en un archivo de texto el mismo que puede ser abierto en cualquier editor de texto.

La problemática vigente en la escuela de Biología, Ecología y Gestión es la necesidad de un oxímetro de campo, para la obtención de datos a diferentes profundidades. Problema que se plantea resolver con la obtención de la sonda YSI 600R y este equipo.

El diseño el software va a ser realizado en lenguaje de programación C, el hardware por su parte, en Altium designer. Además realizar las pruebas del equipo ya sea en el laboratorio o pruebas de campo, son objetivos principales que se pretenden realizar en este trabajo de graduación.

## **CAPITULO 1**

### **ANÁLISIS DE PROBLEMA Y MATERIALES A USAR**

#### **1.1 Análisis del Problema**

La cantidad de oxígeno en los cuerpos de agua da cuenta de las condiciones en las que se encuentra el ecosistema. Cuando las cantidades de oxígeno son bajas seguramente es por la existencia de bacterias que lo consumen para degradar materiales contaminantes que seguramente llegaron a los cuerpos de agua por actividades antrópicas u otras fuentes.

La calidad del agua se ve afectada por diversos factores como el uso del suelo, la producción industrial y agrícola, el tratamiento que se da antes de ser vertida nuevamente, esto significa que el inmenso volumen de líquidos que se vierten en ríos, lagos o mares sin ningún tratamiento previo, ocasiona la contaminación de estos y, en consecuencia, la reducción de agua disponible para el uso de humanos, animales y seres vivientes en general. Un parámetro clave para hacer monitoreo es el pH, la conductividad y el oxígeno disuelto, son parámetros importantes a tomar en cuenta ya que mediante su monitoreo podemos saber si las aguas están contaminadas.

Existen muchas actividades humanas como la minería que afectan mucho los cursos de agua muchas veces acidificando las agua hasta niveles en donde ya no se puede dar la vida y tampoco son aguas que pueden ser usadas; las mediciones de oxígeno, pH y conductividad son parámetros que nos permiten detectar rápidamente si existen afecciones a los cursos de agua y con una rápida detección de afecciones se puede iniciar un proceso para que se tomen las medidas respectivas.

Una problemática vigente en nuestra facultad específicamente en la escuela de Biología, Ecología y Gestión es la necesidad de un oxímetro de campo, para la obtención de datos a diferentes profundidades. Actualmente con los equipos disponibles no se pueden medir a más de 2 metros de profundidad, por lo que no se puede hacer estudios de estos parámetros en profundidades superiores a las ya mencionadas en los lagos o ríos con un gran caudal o profundidad, lo que es un gran problema para cuando se requieren estudios de campo en lugares lejanos ya

que al ser equipos con limitantes de los cables y las sondas al ser de laboratorio son más delicadas que las del campo, por lo que no son la opción óptima para la toma de mediciones de campo. Ahora vamos a explicar con un poco más de detalle cada uno de los parámetros para la mejor comprensión de la importancia que tienen en el agua y en los organismos vivos.

## 1.2 Conceptos

### 1.2.1 El pH

El pH es la abreviatura de "pondus Hydrogenium" que literalmente significa "el peso del hidrogeno" y sirve para indicar la acidez de una sustancia, este se determina por el número de iones de hidrogeno libres ( $H^+$ ) en una sustancia. La acidez esta entre las propiedades más importantes a considerar del agua ya que el agua puede disolver casi todos los iones. El pH sirve para comparar algunos de los iones solubles en el agua. Cuando se mide el pH obtenemos el resultado que es una consideración entre el número de protones (iones de  $H^+$ ) y el número de iones hidroxilo ( $OH^-$ ), cuando estos están en igual proporción el agua es neutra y tendrá su pH alrededor de 7.

El pH del agua puede variar entre 0 y 14 cuando el valor es superior a 7 es una sustancia básica y caso contrario es una sustancia acida, mientras más alejado este el valor de 7 la solución será más acida o más básica, cabe recalcar que el pH no tiene unidad, se puede observar la escala de pH en la imagen de a continuación<sup>1</sup>.

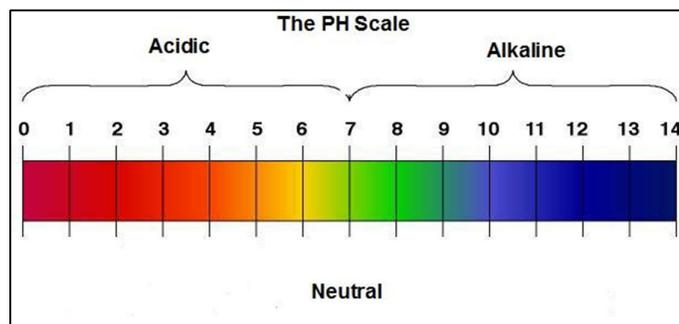


Imagen de la escala del Ph<sup>2</sup>

<sup>1</sup> <http://www.lennotech.es/ph-y-alcalinidad.htm>

<sup>2</sup> <http://3.bp.blogspot.com/-ymsPUp4Tcyg/T26DjukvsPI/AAAAAAAAAEw/bONo9pE4q8o/s1600/PH-Scale.jpg>

### 1.2.2 El electrodo de pH

Un electrodo de pH es un tubo pequeño que se introduce en las sustancias, está unido a un pH-metro por medio de un cable y un fluido que normalmente es “cloruro de potasio 3M” También existen electrodos que utilizan un gel con las mismas propiedades que este fluido.

En este fluido o gel tiene cables de plata y platino. Los iones H<sup>+</sup> y OH<sup>-</sup> entran en el electrodo por una membrana y se crean cargas en los extremos del electrodo y la diferencia entre estas cargas determina el número de iones H<sup>+</sup> y OH<sup>-</sup> y con esto determinado se obtiene el pH, este potencial depende de la temperatura, por eso normalmente un electrodo de pH también mide la temperatura.<sup>3</sup>

### 1.2.3 Oxígeno Disuelto

El oxígeno disuelto proviene de la unión del agua con el aire, que se ocasiona en ciertos casos, por el viento o el oxígeno que liberan las plantas acuáticas. La solubilidad del oxígeno depende de la presión atmosférica en cada sitio, su temperatura y su contenido de sales disueltas es decir que es proporcional a la presión e inversamente proporcional a la temperatura y sales disueltas.

Unos ejemplos de la solubilidad de un gas son el burbujeo de cuando hierve el agua, o cuando se destapa un refresco carbonatado la efervescencia que se produce para equilibrar la presión interna con la externa a la botella. La dependencia de la presión en la solubilidad de un gas en el agua, se puede expresar según la Ley de Henry:

$$C = kP$$

En donde:

C es la concentración molar de oxígeno.

K es una constante de proporcionalidad igual a 0.00035 mol/lit x Atmosfera.

P es la presión del agua a una profundidad determinada.

---

<sup>3</sup> <http://www.lennotech.es/ph-y-alcalinidad.htm>

El oxígeno disuelto es un factor muy importante en el ambiente si hay poco oxígeno en el agua no puede existir vida de peces u otras especies, en la industria mucho oxígeno en el agua es perjudicial ya que causa problemas de corrosión que afectan las partes metálicas por donde circula el agua.<sup>4</sup>

#### 1.2.4 Conductividad del agua

“La conductividad de una sustancia se define como “La habilidad o poder de conducir o transmitir calor, electricidad o sonido”. Su unidad son Siemens o micro Siemens por metro [uS/m].”<sup>5</sup>

##### 1.2.4.1 Conductividad eléctrica (EC)

El movimiento de partículas cargadas eléctricamente resulta en corriente eléctrica, dentro de la mayoría de los sólidos existe un flujo de electrones que provoca una corriente y el flujo de electrones es denominado conducción electrónica. La conductividad depende del número de electrones libres para realizar la conducción, los metales por naturaleza son buenos conductores por su gran número de electrones libres, el agua es un buen conductor, pero el agua destilada en equilibrio con dióxido de carbono del aire tiene un conductividad de  $10 \times 10^{-6} \text{ W}^{-1} \text{ m}^{-1}$  (20dS/m). La corriente eléctrica se transporta por medio de iones en el líquido y esta conductividad aumenta cuando aumentan los iones.<sup>6</sup>

### 1.3 Análisis de Materiales a usar

Para el desarrollo de esta investigación se usan diferentes elementos; pero, los más importantes son los que vamos a mencionar a continuación:

- La sonda, en este caso la sonda es de la marca YSI diseñada específicamente para la realización de mediciones de laboratorio o de campo, ya que tiene un cable con una longitud mayor a la de los 10 metros, y al ser para campo su construcción es más resistente y robusta que la de

---

<sup>4</sup> [http://200.69.103.48/comunidad/grupos/fluoreciencia/capitulos\\_fluoreciencia/calaguas\\_cap16.pdf](http://200.69.103.48/comunidad/grupos/fluoreciencia/capitulos_fluoreciencia/calaguas_cap16.pdf)

<sup>5</sup> <http://www.lenntech.es/aplicaciones/ultrapura/conductividad/conductividad-agua.htm>

<sup>6</sup> <http://www.lenntech.es/aplicaciones/ultrapura/conductividad/conductividad-agua.htm#ixzz1mJ5PHX64>

laboratorio, la sonda se comunica al equipo mediante el protocolo R232 y recibe la alimentación desde el equipo, esta sonda mide todos los parámetros que requerimos directamente es decir que con una sola sonda vamos a obtener los parámetros de pH, Oxígeno, Oxígeno Disuelto, conductividad y temperatura.

- El microcontrolador, es un circuito integrado programable que va a ser el cerebro del equipo por así decirlo ya que vamos a usar un microcontrolador Microchip de la familia 18FXXXX ya que son fáciles de conseguir en nuestro medio y de un costo módico y accesible, estos microcontroladores tienen la ventaja de su facilidad de programación en los lenguajes Ensamblador y C, lo que los vuelve en una de las mejores opciones para el proyecto.
- La placa de entrenamientos de microcontroladores, que va a ser de utilidad para realizar las pruebas de software con facilidad sin tener que modificar el hardware, esta placa es la versión V1.1\_1\_2, esta placa solo va a ser para realizar las pruebas ya que para cuando sea la presentación del equipo se va diseñar un PCB específicamente para el hardware.
- Programador PicKit3, que será usado para transferir el software o el código del programa ya sea en Ensamblador o en C desde el computador hacia el microcontrolador.
- Display matricial, el display matricial será utilizado para visualizar los valores de los parámetros medidos por la sonda, previo al guardado en la memoria de tipo SD o micro SD.
- Una memoria de tipo SD o micro SD, en la que se van a almacenar los datos adquiridos por el equipo para después ser descargados en el computador.
- MicroC, es el programa que nos permite crear los proyectos y programar el microcontrolador en el lenguaje C.
- MplabIDE, es el programa de Microchip para programar el microcontrolador en Ensamblador e importar los archivos creados desde MicroC para enviarlos al pickit y programar el microcontrolador.
- LabVIEW, vamos a usar el programa para instrumentación y control LabVIEW de National Instruments para descifrar los datos que nos envía la sonda y así tratarlos en la computadora, previo el programado del microcontrolador.
- NImyDAQ, se va a usar el DAQ para las pruebas de adquisición de datos, ya que adquiere los datos de una forma mucho más rápida que el

microcontrolador, vamos a usar el DAQ para realizar pruebas de la adquisición.

- Una fuente variable, que será utilizada para la alimentación de la placa de entrenamiento de microcontroladores que posteriormente será sustituida por baterías cuando el equipo este en la etapa de pruebas y mediciones.
- Altium Designer Winter 2009, mediante el uso de este software vamos a diseñar el posicionamiento y ruteo de los componentes en el PCB del equipo, dicho ruteo será enviado a la fábrica para su elaboración previa la presentación final del proyecto.
- Proteus v7.6, será utilizado como soporte para la elaboración y testeo de circuitos previo al armado ya sea en placa impresa, placa perforada o protoboard, esto se utilizará para así evitar posibles errores en los circuitos lo que puede llevar a pérdida de tiempo en búsqueda de dichos errores.
- Multisim, será usado para simulaciones de circuitos y como instrumento virtual para las mediciones ya que al ser de National Instruments hay compatibilidad con el DAQ lo que nos permite tener osciloscopio, voltímetro, amperímetro y otros instrumentos en el computador.
- Componentes varios, aquí hacemos referencia a los componentes de uso común para las pruebas como lo son leds, resistencias, capacitores y borneras como un Protoboard para armar los circuitos de prueba.
- Materiales varios, en estos materiales se incluyen los recursos que son usados diariamente pero que no conforman parte del proyecto como lo son destornilladores planos y de estrella, cables, cautín, computador.

## CAPÍTULO 2

### ANÁLISIS DEL FUNCIONAMIENTO DE LA SONDA Y DISEÑO DE SOFTWARE PARA EL MICRO CONTROLADOR

#### 2.1 Introducción

Este capítulo se basa en dos partes fundamentales, el análisis del funcionamiento básico de la sonda y el diseño de software para el micro controlador usado. El análisis de la sonda es una breve descripción y funcionamiento de cada uno de los parámetros de los cuales está compuesta la sonda, además de las fórmulas utilizadas para medir cada uno de los parámetros.

En la Figura 1. se ilustra un diagrama de bloques de los fragmentos de programa necesarios para realizar el diseño de software final que irá grabado en nuestro microcontrolador, utilizamos MikroC Pro, que es un compilador en C para la programación de microcontroladores, en nuestro caso el microcontrolador PIC 18F4620. Además se utiliza el software MPLAB IDE, el cual nos brindara la posibilidad de importar el programa realizado en MikroC y grabarlo en el pic.

El programa realizado en este compilador va a estar dividido en tres partes fundamentales necesarias para realizar todo el programa.



Figura 1. Diagrama de bloques de partes del software en MikroC PRO

## 2.2 Análisis del funcionamiento básico de la sonda

Para la tesis vamos a usar una sonda de tipo YSI 600R que mide los parámetros requeridos, aparte de también tener una comunicación de datos de diversos tipos.

### 2.2.1 Conductividad

La sonda utiliza una celda de cuatro electrodos de níquel puro para medir la conductividad de los líquidos, dos de los electrodos envían una corriente eléctrica mientras que los otros dos electrodos miden la caída de voltaje en dicha corriente, esta caída se convierte en conductancia y su unidad es miliSiemens, para que este valor se convierta en unidades por centímetro se tiene que multiplicar por la constante de la celda que es aproximadamente 5.0/cm, el software de la sonda es la encargada de realizar esta multiplicación y conversiones antes de comunicarse con el equipo, cuando hace el cálculo de la conductancia también hace el cálculo de la conductividad específica.

#### 2.2.1.1 Efecto de la temperatura en la conductividad

La conductividad de las sustancias dependen bastante de la temperatura de dicha sustancia, tanto así que varía un 3% por cada grado Celsius, por lo que podemos decir que el coeficiente de la temperatura es de un 3%/C, pero este coeficiente varía con la naturaleza iónica de la sustancia.

Como normalmente la composición de este medio no es conocida, lo mejor es reportar la conductividad a una temperatura en particular, sin embargo, en muchos casos también es útil compensar la dependencia de la temperatura para determinar si hay cambios bruscos ocurriendo en el contenido iónico del medio con el paso del tiempo.

Por eso esta sonda y la de la serie de los 600 nos da la posibilidad de mostrar los datos de la conductividad sin compensar o compensados, normalmente los datos compensados se hacen en referencia a una temperatura de 25 grados Celsius, si seleccionamos la conductividad específica el cálculo es realizado para 25 grados y es como en la siguiente ecuación.

$$\text{Conductancia específica (25°C)} = \frac{\text{Conductancia}}{1 + TC \times (T - 25)}$$

El cálculo es realizado con un coeficiente de temperatura de 1.91% es decir que  $TC=0.0191$ .

A menos que la sustancia a ser medida solo sea KCl (Cloruro de potasio) en el agua, el valor de la temperatura compensada va a ser inexacto, pero en la ecuación con el valor de  $TC = 0.0191$  nos va a dar una aproximación muy cercana a agua de mar o soluciones con mucha sal como NaCl (Cloruro de Sodio) y  $NH_4Cl$  (Cloruro de Amonio).

### 2.2.2 Salinidad

La salinidad se determina automáticamente de la medida de conductividad de acuerdo a algoritmos usados en el libro "Standard Methods for the Examination of Water and Wastewater" en español "Métodos básicos para el análisis de Agua y Agua de desechos", en base a este algoritmo las medidas se realizan con las mismas referencias que las de la conductividad y el resultado nos da en partes por mil (ppt), la relación entre la salinidad y la conductividad es la siguiente:

$$\frac{2\mu S}{cm} = 1ppm(\text{Parte por millon}) = 1mg/l$$

Los medidores de salinidad tienen compensación manual o automática de temperatura en este caso la compensación esta en la conductividad.

### 2.2.3 pH

La sonda usa un electrodo de pH que puede ser remplazado cuando se termine su vida útil o sufra alguna avería. La sonda usa sensores individuales de vidrio y un electrodo de referencia, ahora vamos a explicar el principio de funcionamiento a breves rasgos como nos lo indica el catálogo de la sonda.

El sensor de vidrio utiliza un protón selectivo que está lleno de un buffer de pH 7 con un alambre de plata recubierto de AgCl. El electrodo de referencia está basado en un par de Ag/AgCl y consiste en un alambre recubierto de cloruro de plata dentro de un gel donde el cloruro de potasio tiene una concentración de 4M. Los protones o iones positivos del Hidrogeno en cada lado del sensor de vidrio reaccionan selectivamente con el vidrio, creando un potencia a través de la

membrana de vidrio, ya que la concentración de iones en el buffer interno es invariante, esta diferencia de potencial se determina con relación al electrodo de Ag/AgCl, en proporción con el pH del medio.

### 2.2.3.2 Calibración y Efectos de la temperatura

La sonda calcula el pH de la relación lineal entre el pH y la salida de milivolts que se define por la ecuación de Nernst:

$$E = E_0 + \frac{2.3RT}{nF} * \text{pH} \text{ Donde } E \text{ es la salida en milivolts.}$$

$E_0$  es la constante asociada al electrodo de referencia.

T es la temperatura medida en grados Kelvin

R, n y F son constantes invariantes.

Si simplificamos esto nos queda que la relación es (salida en mV) = pendiente \* pH, para que se cumpla esta condición, el instrumento debe ser calibrado correctamente usando buffers de valores conocidos de pH. En el proceso de calibración con dos puntos de referencia el software de la sonda hace el cálculo de la pendiente y lo convierte en un valor de pH, por esto la calibración debe realizarse a una misma temperatura, y cuando se va al campo para hacer mediciones en el medio ambiente se debe compensar la temperatura es decir convertir la pendiente e interceptar la gráfica de pH con la de los miliVolts en una  $T_c$  (Temperatura de calibración) y pasarlo a una pendiente que intercepte una  $T_m$  (Temperatura de la medida), para nuestra suerte la ecuación de Nernst nos da las bases para esta conversión.

De acuerdo a la ecuación de Nernst, la pendiente de pH vs miliVolts es directamente proporcional a la temperatura absoluta en grados Kelvin. Además su la pendiente es experimentalmente determinada a ser 59mV/pH a una temperatura de 25°C es decir 298 Kelvin, entonces la pendiente para una temperatura de 313 K (40°K) debe ser  $(313/298) * 59 = 62\text{mv/pH}$  y a 283K (10°C) la pendiente se calcula a ser  $56\text{mv/mV}$  por la misma relación  $(283/298)*59$ . Para establecer el cruce en la nueva grafica el punto en donde el pH y los mV en diferente temperatura se intersectan, es decir en un punto isopotencial, debe ser conocido. El software de la sonda asigna el punto para que sea una medida en mv a pH 7 y calcula ese cruce usando esa suposición, una vez que la pendiente intercepte la gráfica del pH vs. mV se asigna una nueva temperatura y los cálculos de pH bajo la nueva

temperatura estos cálculos son realizados automáticamente por el software de la sonda.

#### **2.2.4 Temperatura**

La sonda viene con un termistor metálico que con el cambio de temperatura varia su resistencia, a esto se le aplica una corriente constante y en base a la variación del voltaje se puede obtener la temperatura mediante la gráfica del sensor y en base a ello se obtiene la relación.

#### **2.2.5 Oxígeno Disuelto**

Para la medida del oxígeno disuelto se usa el método patentado de YSI que consiste sensores electro-químicos de DO (Oxígeno Disuelto), estos sensores son constantemente polarizados con el suficiente voltaje negativo para hacer que el oxígeno sea reducido a un ion de hidróxido en el cátodo que es de plata y el cloruro de plata se va al ánodo, con esto el oxígeno se difunde por la membrana de teflón y la corriente asociada con este proceso es proporcional al oxígeno presente en la muestra fuera de la membrana.

La membrana aísla los electrodos necesarios para la reducción del medio externo, dentro de la fina capa del electrolito para prever que otras sustancias no gaseosas, electroquímicas o especies afecten la medida, como se puede observar esta sonda usa tres electrodos, un ánodo, cátodo y un electrodo de referencia, además la geometría del sensor consiste en una pequeña línea de un cátodo de oro ubicado entre dos rectángulos de plata que sirven como ánodo y electrodo de referencia.

##### **2.2.5.3 Principio de funcionamiento**

Como se indicó antes los sensores se polarizan con voltajes negativos que hace que el oxígeno de la membrana se reduzca a un ion de hidróxido en el cátodo y cloruro de plata en el ánodo, y como se dijo antes la corriente asociada con este proceso es proporcional al oxígeno presente fuera de la membrana, pero como es una reacción electroquímica en el proceso el oxígeno es consumido en el medio dando como resultado una disminución de la medida, para evitar esto, la sonda se polariza y despolariza rápidamente durante la medida permitiendo así que la medida no se va comprometida ya que hace que el consumo de oxígeno en el proceso sea mínimo.

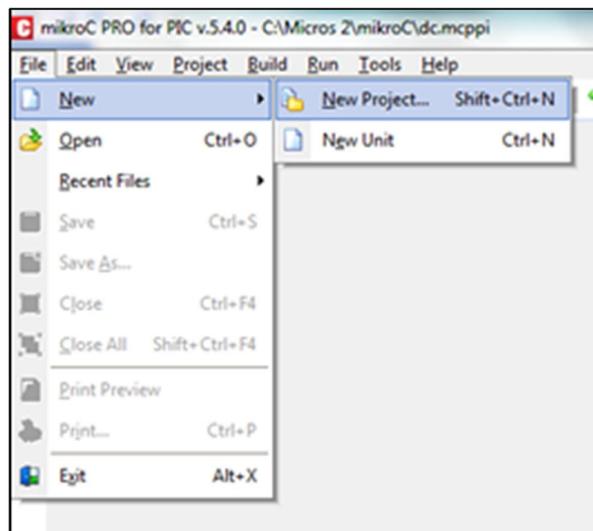
## 2.3 Diseño de Software

### 2.3.1 Adquisición de datos de la sonda, interacción con el usuario

El desarrollo de un programa en MikroC PRO, está dividido en diferentes partes, entre ellas están: Creación de un proyecto, la cabecera, declaración de las variables, subrutinas de inicialización para diferentes puertos y programa en general que se indicarán a continuación.

#### 2.3.1.1 Crear un proyecto en MikroC PRO

Para realizar el programa en el compilador MikroC, se procede a realizar un número definido de pasos necesarios para la inicialización de MikroC PRO, a continuación detallamos cada uno de los mismos. Para crear un nuevo proyecto a realizar damos click en File – New – New Project, como se ilustra en la Figura 2.



**Figura 2.** Creación de un nuevo proyecto en MikroC PRO

Al realizar esta acción se nos abrirá una nueva ventana denominada New Project Wizard, que será el editor de un nuevo proyecto, en esta pantalla seleccionamos Next, como se muestra en la Figura 3, lo cual nos enviará a otra ventana; siguiendo los pasos indicados y seleccionando el microcontrolador a usar que en nuestro caso va a ser el 18F4620 y el clock device, para nuestro proyecto utilizaremos un cristal de 16Mhz para el reloj interno a ser usado en el micro controlador.



Figura 3. New Project Wizard.

Los siguientes puntos los dejamos por defecto con lo cual se seleccionarán todas las librerías de MikroC PRO. Con lo cual se ha creado un proyecto. Las partes de las cuales va a estar compuesto el software en este compilador son las siguientes:

### 2.3.1.2 Declaración de variables para el uso del LCD

Crearemos una subrutina de inicialización de las variables utilizadas en el LCD, para esto vamos a utilizar una LCD matricial de 4x20, es decir va a tener 4 filas y 20 columnas o matrices como indicadores.

```

. //-----
.
. //Subrutina de inicializacion de pines utilizados para el LCD
. sbit LCD_RS at RA4_bit; //Utilizaremos el pin RA4 como RS necesario para el LCD
20 sbit LCD_EN at RC1_bit; //Utilizaremos el pin RC1 como Enable necesario para el LCD
. sbit LCD_D4 at RD4_bit; //Utilizaremos los pines del puerto D del 7 al 4 como salidas para el LCD
. sbit LCD_D5 at RD5_bit;
. sbit LCD_D6 at RD6_bit;
. sbit LCD_D7 at RD7_bit;
.
26 sbit LCD_RS_Direction at TRISA4_bit; //se seleccionan y activan de los pines para su uso con el LCD
. sbit LCD_EN_Direction at TRISC1_bit;
. sbit LCD_D4_Direction at TRISD4_bit;
. sbit LCD_D5_Direction at TRISD5_bit;
30 sbit LCD_D6_Direction at TRISD6_bit;
. sbit LCD_D7_Direction at TRISD7_bit;
. // Termino de seleccion de pines para el LCD
. //-----

```

Figura 4. Subrutina de inicialización de la pantalla LCD.

En MikroC existe una subrutina previamente descrita la cual proporciona los pines necesarios para la comunicación del microcontrolador y el LCD matricial, en la Figura 4, se ilustra esta subrutina.

Los pines a ser utilizados por el LCD para la comunicación de datos son los siguientes:

- RA4 RS.
- RC1 Enable.
- RC2 Backlight.
- RD4-RD7 Comunicación LCD.

Y la conexión de los mismos se ilustra en la Figura 5.

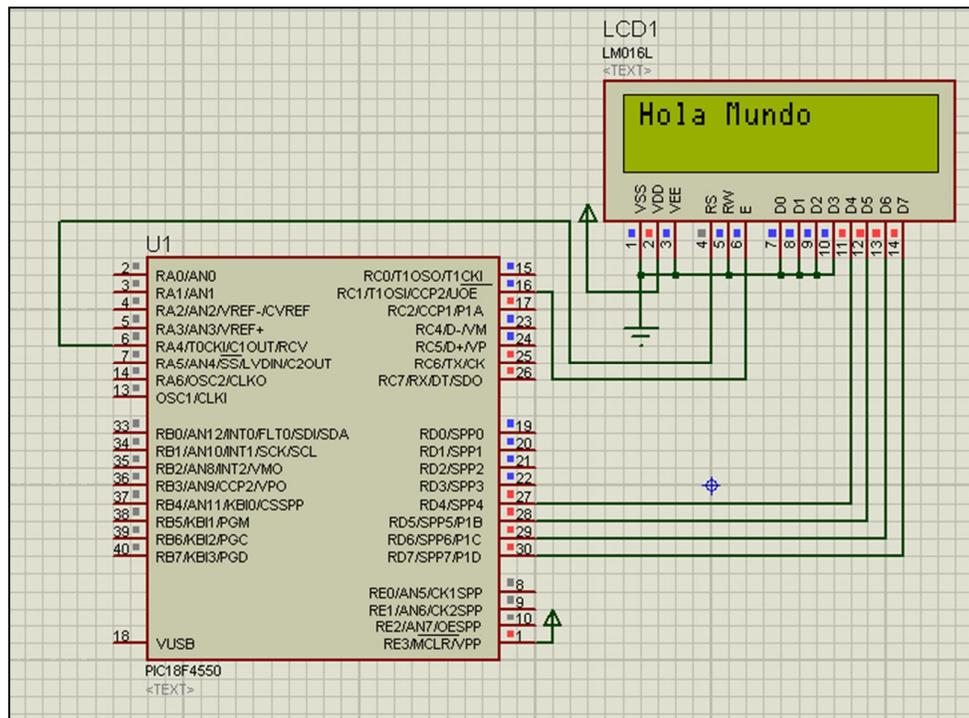


Figura 5. Conexión microcontrolador y pantalla LCD.

### 2.3.1.3 Declaración de variables de uso general

En este punto vamos a colocar la declaración de variables a ser utilizadas, estas son variables de diferentes tipos como se muestra en la Figura 6, en cada una de las mismas se indica su función en las subrutinas posteriores

```

//Declaración de las variables a ser utilizadas
int b;           //Se usara esta variable para la prueba de los puertos a necesarios para la
double texte;   //Esta variable es utilizada para adquirir los datos q llegan del puerto USA
int i = 0;       //Bandera utilizada para reducir espacios en la lectura de datos obtenidos
int j = 0;       //Bandera utilizada para reducir espacios en la lectura de datos obtenidos
int m = 1;       //Variable utilizada en la sección de Menu para seleccionar el camino a seg
char g, byte_read; //Esta variable es utilizada para adquirir los datos q llegan del puerto
int Fil,Col;     //Declaración de las variables a ser utilizadas como fila y columna
#define MAX_COL 20 //Se define el número máximo de columnas del LCD matricial en este caso
#define MAX_FIL 4  //Se define el número máximo de filas del LCD matricial en este caso son
//-----

```

Figura 6. Inicialización de variables a ser utilizadas.

### 2.3.1.4 Inicialización del segmento principal del programa

Después de generadas todas estas las subrutinas anteriores, se procede a realizar subrutinas dentro del segmento principal, que facilitarán el desarrollo del mismo; para realizar esto se divide en algunas partes, una de ellas es la asignación de puertos a ser utilizados dentro del micro controlador.

Como se muestra en la Figura 7, el segmento de programa principal se empieza por "void main( void ) {" , y a continuación la declaración de puertos y el uso de los mismos ya que unos van a ser utilizados como entrada digital y otros como salida analógica para las funciones que pretendemos realizar.

```

. //Subrutina Principal en la que se encuentra el desarrollo del código a ser realizado
80 void main( void ){
. //.....
. TRISB = 0b00001111; //Se seleccionan los pines de ingreso y salida del puerto B, en este
. TRISD =0;           //Se seleccionan los pines de ingreso y salida del puerto D, en este
. PORTD=0;           //Se inicializa todo el puerto D al valor de 0
. TRISC =0;         //Se seleccionan los pines de ingreso y salida del puerto C, en este
. TRISE =0;
. CMCON = 0;        // desactivar los comparadores
. PORTC.RC0 = 0;    // desactivar los comparadores
. PORTC.RC2 = 1;    //Se activa el Backligh Utilizado en el LCD
90 ADCON1 = 0xF;    // Configure AN pins as digital
. CMCON = 7;        // Disable comparators
. INTCON2 = 0;
. PORTB = 0x7;

```

Figura 7. Inicialización de puertos.

De igual manera, debemos inicializar el LCD y el puerto UART a ser utilizado, MikroC PRO nos brinda gran facilidad para realizar esta función como se muestra en la Figura 8; Para el caso de la inicialización del puerto UART, seleccionamos una velocidad de 600bps a 8 bits y sin paridad. Para los dos casos necesitamos dar una demora para la inicialización de las subrutinas.

```

- //.....
- //Subrutina de inicialización del LCD
- Lcd Init(); //Por defecto en MckroC se utiliza este comando de inic
- Delay_ms(100); //Demora de 100ms para la inicialización del LCD
- Lcd Cmd( LCD CLEAR); //Se borra todo el contenido en la LCD
00 Lcd Cmd( LCD CURSOR OFF); //Se desactiva el cursor
-
- //.....
- //Subrutina de inicialización del protocolo de comunicación Uart
- UART1 Init(600); //Inicialización del protocolo Uart en 600 baudios a
- Delay_ms(100); //Tiempo de demora para la inicialización del puerto
-

```

Figura 8. Inicialización del LCD y puerto UART.

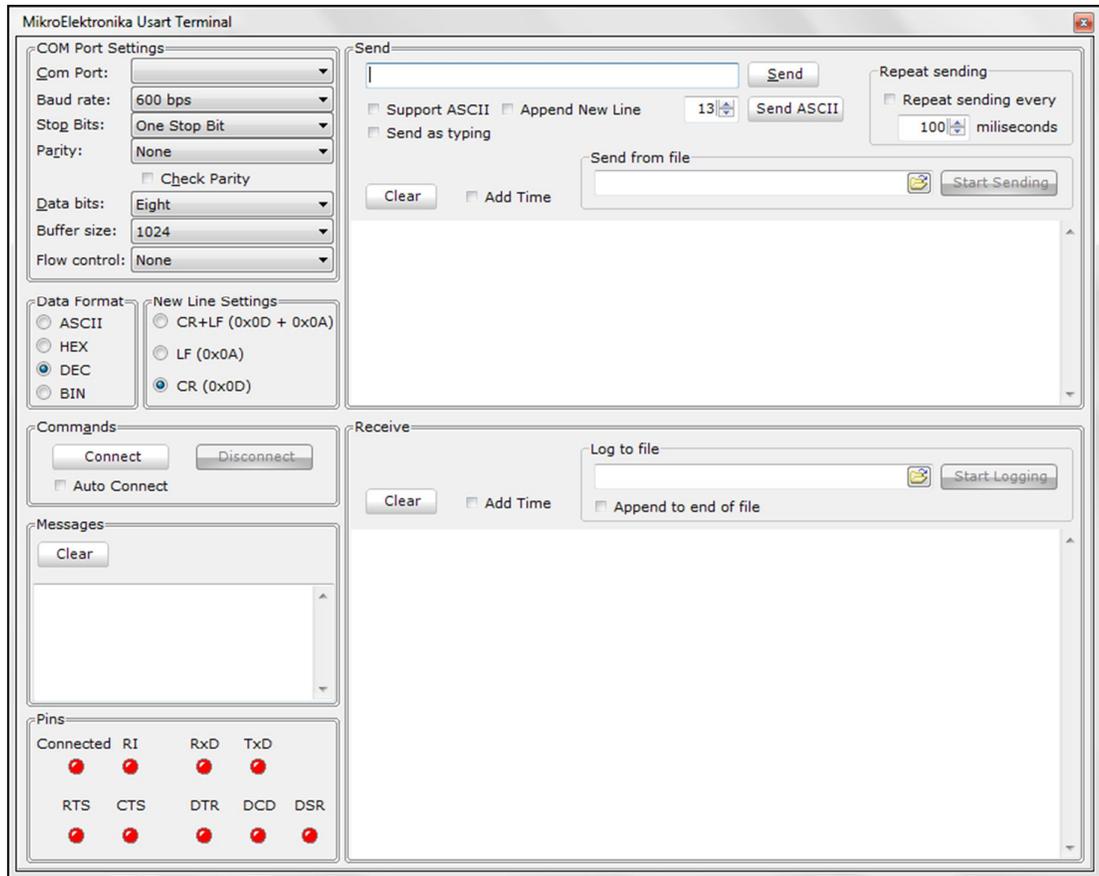
### 2.3.1.5 Descripción breve de la comunicación con la sonda

La comunicación con la sonda se va a dar mediante el puerto UART del micro controlador, para las pruebas realizadas utilizamos una herramienta que nos brinda el software de programación MikroC PRO y esta herramienta es USART Terminal, la misma a la que podemos acceder dando click en la barra de herramientas en Tools – USART Terminal o pulsando Ctrl+t; con esto ingresamos a una ventana que es el terminal remoto, como se ve en la Figura 9.

Para esta comunicación en nuestro caso requerimos de un convertor de RS232C a USB, ya que nuestro computador no cuenta con el puerto RS232C; en este terminal seleccionamos las características que necesitamos para nuestra conexión como son:

Com Port: Com4(el lugar en donde nos detecta que conectamos el convertor RS232C-USB)

Baud rate: 600bps.



**Figura 9.** MikroElektronika USART Terminal.

Una vez conectada la fuente de poder a la sonda y los cables conectados de manera correcta a nuestro puerto, damos click en Connect, ahora estamos conectados con la sonda, y esta nos enviará un mensaje que lo recibiremos en ASCII, este mensaje se muestra en la Figura 10, una vez conectado a la sonda debemos enviar los caracteres exactos para entrar al menú y controlar su funcionamiento. Los datos a enviarse y recibir siempre estarán en ASCII.

```

YSI Water Quality Systems
Version 1.10
Copyright (c) 1992-1996 by YSI Incorporated.
All rights reserved.
•
Type MENU<Enter> for instrument menu.
#

```

**Figura 10.** Mensaje entregado por la sonda.

### 2.3.1.6 Conexión con la sonda en el segmento principal

Para conocer cuando la sonda se conecta al equipo o en su caso al micro controlador lo que hacemos es enviar ENTER cada cierto tiempo; cuando hacemos esto con la sonda, esta nos comienza a responder con "#", es decir que si la sonda no está conectada no recibimos nada, mientras q si la sonda se conecta, nos empieza a enviar numerales.

Lo que hacemos es preguntar si se recibió un numeral, lo preguntamos en su número decimal como es el 35, y si nos responde afirmativamente, la sonda estará conectada, caso contrario se escribirá el mensaje de sonda no conectada. Este segmento de programa se ilustra en la Figura 11.

```

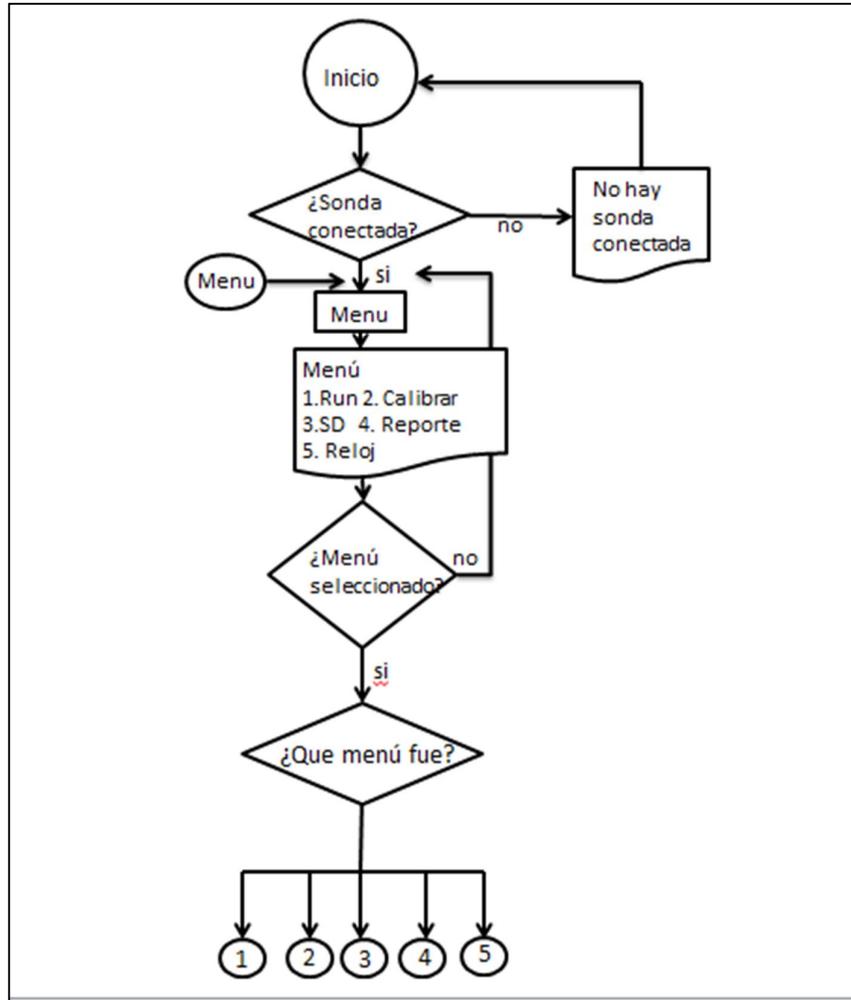
. //-----
. // RUTINA PRINCIPAL DEL PROGRAMA
. while(1)
. {
.   Delay_ms(60);           //Damos un retardo para que se escriba la palabra
.   UART1_Write(13);
.
.   if (UART1_Data_Ready()) //Se pregunta si se recibio algún dato
.   {
.     g = UART1_Read(); //Se lee el datos recibido
.     if ((UART1_Read() == 35) //Si llega # reconocio la sonda y
284 { ...
.     }
.     Lcd_Out(2, 6, "No existe"); //Imprimimos en la Pantalla el mensaje Universidad del
650 Lcd_Out(3, 4, "Sonda Conectada");
.   }
. }

```

Figura 11. Conexión con la sonda en el programa principal.

### 2.3.1.7 Ingreso al menú principal de la sonda

La Figura 12, ilustra mediante un diagrama de bloques el ingreso al menú de la sonda y sus posteriores submenús.



**Figura 12.** Diagrama de bloques del menú principal de la sonda.

Una vez que detectamos la conexión con la sonda, lo que hacemos es enviar la palabra “menu”, seguida de un ENTER (13 en decimal) con un retardo de tiempo cada una, y mostramos en la pantalla “Conectando..”, “YSI 600R”, mensajes que indican que se está realizando la conexión.

Después de esta acción nos encontraremos en el menú principal de las sonda por lo que imprimiremos en la pantalla después de limpiarla un menú que es el que nosotros crearemos para la comunicación con la sonda.

### 2.3.1.8 Selección de submenú dentro del menú principal

Para realizar esta acción debemos utilizar un pulsante, ya que lo que queremos es desplazarnos por la pantalla, el puerto a utilizar es el puerto B y el pin es el 3, la forma de desplazamiento por la pantalla es la impresión de “\*” en el lugar en donde

se encuentre el submenú a seleccionar. Para seleccionar dicho menú se va a necesitar de otro pulsante, en este caso sería el pulsante de ingreso, y va a ser utilizado mediante el pin 2 del puerto B.

Para realizar el testeo del pulsante se va a ingresar a un bucle infinito, como se muestra en la Figura 13, esto se realiza mediante el uso de `do{.....}while(1)`, y lo siguiente es testear cada cierto tiempo, en este caso cada 150ms, si se ha realizado un cambio de valor en el pin 3 del puerto B. Si esto sucede la bandera `m` aumenta su valor con lo que "\*" cambia de posición; cuando `m` tenga el valor de 5, le asignamos el valor de 1, con lo que seguimos en este menú sin importar las veces que se pulse.

Cuando nos encontremos en un submenú, podemos pulsar el pin 2 con lo que ingresamos a este submenú y realizamos su función específica, esto funciona de la misma manera que el pulsante anterior, es decir con una demora de tiempo para testear el momento en el que se pulsa.

```
do {
  delay_ms(150);
  if (Button(&PORTB, 2, 1, 0)) { // (puerto, #de pin,
    m++;
    delay_ms(100);
  }
  if(m == 1){ //Run [...]
    if(m == 2){ //Calibrar
      Lcd_Out(2, 1, " ");
      Lcd_Out(2, 12, "***");
    }

    if(m == 3){ //Sistema
      Lcd_Out(2, 12, " ");
      Lcd_Out(3, 1, "***");
    }

    if(m == 4) // Reporte
    { [...]

    if(m == 5) {
      Lcd_Out(3, 12, " ");
      m = 1;
    }
  }
}while(1);
```

Figura 13. Selección de un submenú mediante pulsantes.

### 2.3.1.9 Submenú RUN

La Figura 14, ilustra mediante un diagrama de bloques el ingreso al submenú Run y su funcionamiento.

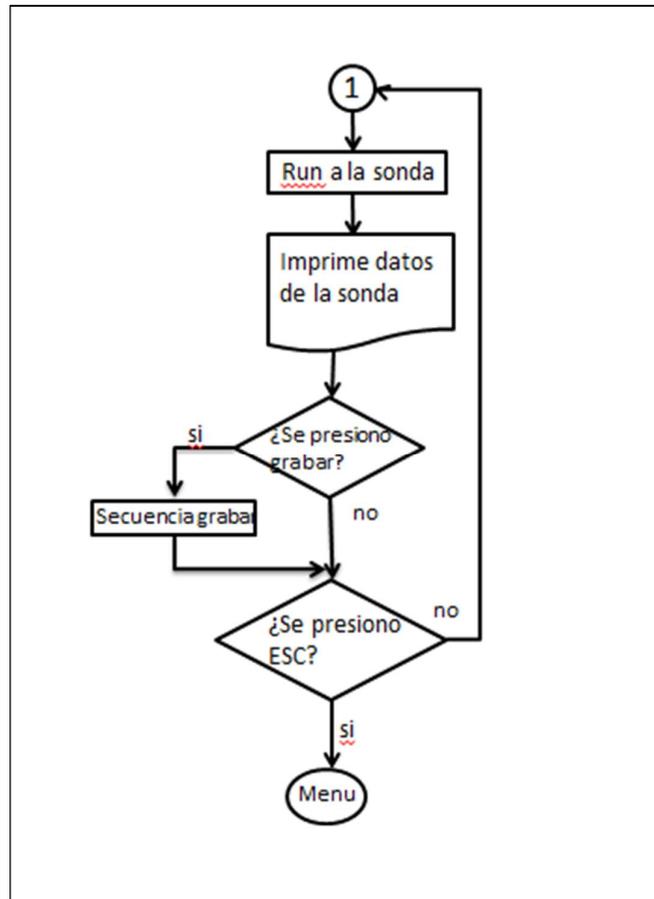


Figura 14. Diagrama de bloques del menú RUN de la sonda.

Después de haber seleccionado el submenú Run, pulsamos el pulsante asignado al pin 2, con lo que ingresaremos a una subrutina la cual es la subrutina de Run.

Para ingresar a este submenú dentro de la sonda lo que debemos realizar es enviar “1”, una demora y “1” en decimal, mediante el puerto USART, así ingresaremos al submenú Run dentro de la sonda. En la Figura 15 se muestra este segmento de programa. Además debemos inicializar las variables a ser utilizadas y borrar la pantalla ya que ingresaremos datos. Con las acciones realizadas la sonda nos comenzará a enviar datos.

```

Delay_ms(100);          //Demora de testeo para el RBI
if (Button(&PORTB, 1, 1, 0)) //Testeo del estado del pin RBI
//Submenú RUN, aquí se realiza la adquisición de datos directa desde la sonda
{
  UART1_Write_Text("1"); //Envío de 1 a la sonda para ingresar a un submenu
  Delay_ms(400);
  UART1_Write_Text("1"); //Envío de 1 a la sonda para iniciar la posición de Run
  Lcd_Cmd(_LCD_CLEAR); //Borrar la pantalla del LCD
  Fil = 3; //Inicialización de variables a usarse
  Col = 1;
  i=0;
  j=0;
}

```

**Figura 15.** Envío de datos para el ingreso a Run.

Después de reconocer los parámetros a ser escritos, entramos en un bucle, el cual va a obtener datos de la sonda, hasta que se pulse el botón de salida; es decir va a ser un bucle que terminara con el botón de salida, este botón lo asignamos al puerto B, y es el pin 0, en la Figura 16, se muestra el bucle.

```

while (PORTB.F0 == 1) //Mientras no se pulse RBO , los datos se
{
  //seguirán actualizando automáticamente.
  if (UART1_Data_Ready()) //Se pregunta si se recibió algún dato
  { ...
}

```

**Figura 16.** Bucle de adquisición de datos.

Dentro de este bucle se van a adquirir datos los cuales van a ser de alguna manera procesados, ya que los vamos a imprimir en la pantalla del LCD en una posición predefinida para cada valor, y vamos a preguntar por si llega un Enter, esto significa que se van a sobrescribir los datos, por lo tanto debemos tener un conteo exacto de las posiciones a las cuales se van a asignar cada parámetro.

Como se muestra en la Figura 17, esto se consigue con el uso de banderas para la selección de cada uno de los espacios a ser utilizados, ya que utilizado las variable Fil y Col, podemos elegir la posición exacta para la impresión de datos.

```

        if ((UART1_Read() == 13)) //Si se lee un cambio de linea en hexadecimal
        {
            //se vuelve al origen y no se escribe nada
            g=" ";
        }else {
            Lcd_chr(Fil, Col++, g); //Se Muestran los datos en la posición seleccionada
        }
        if(Col > MAX_COL) //Si los datos llegan al borde de la pantalla se
        { //sobreescriben al inicio de la misma linea
            Col = 1;
            Fil=Fil+2;
            if(Fil > MAX_FIL) //Si los datos llegan al borde de la pantalla se
            {
                Fil=2;
            }
        }
    }
}

```

Figura 17. Procesamiento a datos adquiridos.

### 2.3.1.10 Fin del submenú RUN

Como se muestra en la Figura 18 cuando queremos salir del submenú Run, pulsamos el RB0, con lo que saldremos del bucle, al salir el programa realiza un segmento para regresar al menú principal, el cual es enviar "0", seguido de "0" con sus retardos de tiempo respectivos y así volvemos al menú principal de la sonda y por lo tanto imprimimos en la sonda el menú principal y así podemos elegir cualquiera de los submenús principales.

```

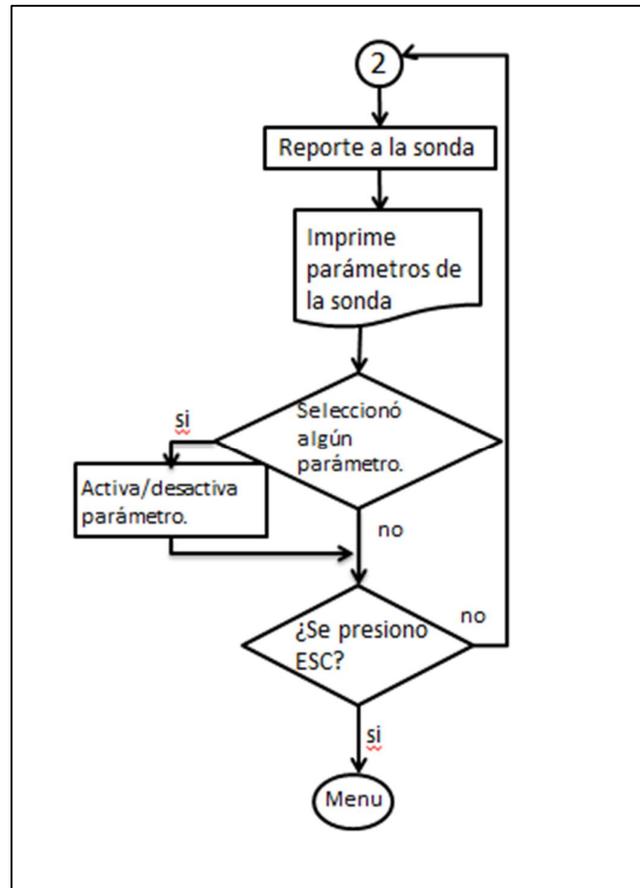
    }
    Lcd_Cmd( LCD_CLEAR);
    UART1_Write_Text("0"); //Envío de 0 a la sonda para regresar al menu
    Delay_ms(600); //Demora de envio de datos
    UART1_Write_Text("0"); //Envío de 0 a la sonda para regresar al menu
    Delay_ms(600); //Demora de envio de datos
    Lcd_Out(1, 8, "MENU"); //Imprimir el menu principal en el LCD
    Lcd_Out(2, 2, "Run Calibrar");
    Lcd_Out(3, 2, "Sistema Reporte");
    i=0;
    j=0;
}

```

Figura 18. Salida del submenú Run.

### 2.3.1.11 Submenú Reporte

La Figura 19, ilustra mediante un diagrama de bloques el ingreso al submenú Reporte y su funcionamiento.



**Figura 19.** Diagrama de bloques del menú Reporte de la sonda.

Lo que se pretende indicar con este submenú, son los parámetros que se encuentran activos en la sonda y por lo tanto se obtendrán valores de los mismos; Los parámetros que están disponibles y que vamos a utilizar son:  
Temperatura.

Conductividad.

Total de sólidos disueltos (TDS).

Oxígeno disuelto (en porcentaje %).

Oxígeno disuelto (en mg/l).

Ph.

Estos parámetros son con los cuales nuestro equipo va a trabajar. Los mismos van a poder ser seleccionados, y deseleccionados, con lo que se tendrá la posibilidad de elegir los parámetros a trabajar. Para ingresar en el submenú Reporte, en el menú principal debemos posicionar el "\*" en la posición que se encuentre Reporte, esto lo hacemos pulsando el botón de selección o en su defecto el RB3. Realizado esto pulsamos el botón de ENTER o RB2, con lo que ingresamos a este submenú.

Al igual que para el ingreso al submenú RUN, se asigna un valor a la variable “m” y de acuerdo a esta se ingresa al submenú correspondiente.

Lo que el programa va a realizar es enviar un código en ASCII, que en este caso sería “4” en decimal, como se muestra en la Figura 20, esta figura también indica que después de dado el pulso de ENTER el programa va a realizar tres subrutinas diferentes.

```

if(m == 4) // Reporte
{
  Lcd_Out(3, 1, " ");
  Lcd_Out(3, 12, "");
  Delay_ms(100); //Demora de testeo para el RB1
  if (Button(&PORTE, 1, 1, 0)) //Testeo del estado del pin RB1
    //Se leen los datos seleccionados y se escriben los que se quieran
    {
      UART1_Write_Text("4"); //Envio de 4 a la sonda para ingresar a un submenu
      reporte1();
      reporte2();
      reporte3();
    }
}

```

Figura 20. Ingreso al submenú Reporte.

La primera subrutina a realizar es reporte1(), en la Figura 21 se ilustra esta subrutina mencionada, y su función es imprimir en la pantalla que los datos se están cargando y además inicializar las variables a ser utilizadas en la siguiente subrutina.

```

198 void reporte1(){
.
.
200   Lcd_Cmd( LCD_CLEAR); //Borrar la pantalla del LCD
.   Lcd_Out(1, 7, "REPORTE");
.   Lcd_Out(3, 6, "cargando");
.   Lcd_Out(4, 7, "datos");
.
.
-   Fil = 1; //Inicialización de variables a usarse
.   Col = 1;
.   i=0;
.   j=0;
.   e=0;
210 }

```

Figura 21. Subrutina reporte1().

La segunda subrutina es reporte2(), esta sonda nos indica que parámetros son los que están activos dentro de la sonda. La tercera subrutina a realizar es reporte3(), su principal función es escribir los parámetros que se encuentran activos para que





De la misma manera en la que ingresamos a los submenús RUN y Reporte, ingresaremos al submenú calibrar, es decir seleccionando la opción calibrar con el pulsante de selección y pulsando ENTER. Para la comunicación con la sonda, debemos enviar el valor "2", como se muestra en la Figura 25, en esta figura también se ilustra la impresión de este submenú.

```

if(m == 2){ //Calibrar
  Lcd_Out(2, 1, " ");
  Lcd_Out(2, 12, "");

  Delay_ms(100); //Demora de testeo para el RBI
  if (Button(&PORTB, 1, 1, 0)) //Testeo del estado del pin RBI
    //Se leen los datos seleccionados y se escriben los que se quieren
    {
      UART1_Write_Text("2"); //Envio de 4 a la sonda para ingresar a un submenu
      Lcd_Cmd(_LCD_CLEAR);
      Lcd_Out(1, 7, "CALIBRAR"); //Colocamos la palabra Calibrar
      Lcd_Out(2, 2, "Conductividad");
      Lcd_Out(3, 2, "Oxígeno D");
      Lcd_Out(4, 2, "pH");
      delay_ms(300);
      i=1;
    }
}

```

Figura 25. Selección de parámetros.

Para calibrar un parámetro de la sonda, vamos a constar de un conjunto de submenús. En la siguiente imagen (Figura 26), se indica en una estructura jerárquica, en la cual se muestra la estructura del submenú calibrar.

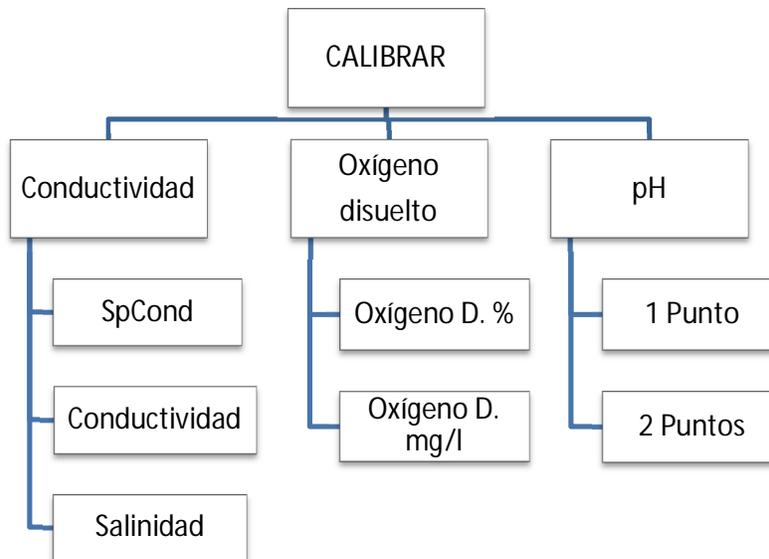


Figura 26. Estructura jerárquica del Submenú Calibrar.

De esta manera pretendemos construir un camino de la misma forma que utilizamos para llegar a los submenús, mediante el pulsante de selección nos movemos dentro de la pantalla de cualquier submenú seleccionado y con el pulsante ENTER, ingresamos al mismo.

Cuando estemos en el parámetro, el cual vamos a calibrar, lo que debemos hacer es ingresar un valor, para esto vamos a utilizar una subrutina que nos ayudará a ingresar el valor. El nombre de la subrutina es valor(), y como se muestra en la Figura 27, este segmento de código, nos muestra que con el pulsante ENTER, el valor de cada dígito va a cambiar, con lo que pulsando el botón de selección vamos cambiando cada uno de los dígitos del valor a calibrarse. Además de esto, inicializamos las variables a ser utilizadas por nuestra subrutina

```

void valor(){
    //Se va a imprimir 0000.00 y se va a cambiar el valor con el pulsante3
    //Con el pulsante dos pasara al siguiente y con el 1 saldrá y grabara el
    i=0;
    j=0;
    m=1;
    val=0;
    v=10000;
    Col=5;
    Fil=3;
    Lcd_Out(Fil, Col, "0000.000");
    //Lcd_Cmd(_LCD_UNDERLINE_ON);
    while (PORTB.F0 == 1){
        delay_ms(120);
        if (Button(&PORTB, 1, 1, 0)) { // (puerto, #de pin, Pasa de 1, a 0)
            m++;
            delay_ms(10);
        }
    }
}

```

Figura 27. Inicialización de la subrutina valor().

En la Figura 28, se ilustra el segmento de código, el cual es la continuación de la subrutina valor(). En este se muestra la operación que se realiza para obtener el valor deseado, ya que la finalidad de esta subrutina es obtener un valor. Para obtener este valor, se va a cambiar dígito por dígito mediante los pulsantes de selección y ENTER; con el pulsante de selección se va a escoger el dígito a editar, este puede ser centenas, decenas, unidades, etc. Mientras que con el pulsante ENTER se cambiara el valor de este dígito seleccionado.

La operación que se realiza es multiplicar cada uno de estos dígitos por su valor asignable, es decir que si editamos el valor de las centenas a por ejemplo 4, este valor se multiplicará por 100 y se sumará al resto de dígitos.

```

if(m == 2){
  delay_ms(80);
  Lcd_Out(Fil, Col, "1");
  if (Button(&PORTB, 2, 1, 0)) { // (puerto, #de pin, Pasa de 1, a 0)
    Col++;
    m=1;
    j=1;
    if (Col == 9){
      Col=10;
    }
    v=v/10;
    val=val+1*v;
    if (Col > 12){
      Col=5;
      j=10000;
    }
  }
}

```

**Figura 28.** Segmento de la subrutina valor, operación para obtener valor.

Después de haber obtenido el valor, debemos enviarlo a la sonda, para realizar esta acción, primero debemos convertir nuestro valor de float a String, MikroC PRO nos brinda un comando con el cual se hace muy sencilla esta conversión, el comando es FloatToStr(val, h), siendo val el valor flotante y h un string a cual se le va a escribir el dato.

Una vez enviado el dato debemos esperar que se inicialice el dato con fin de que los datos estén estables y calibrarlos. Los datos de los valores, después de enviar el valor de calibración van a fluctuar, durante un tiempo hasta estabilizarse, estos datos van a ser mostrados en la pantalla utilizando la subrutina RUN, previamente descrita en la sección 2.3.1.9.

Por otra parte si el dato a calibrar no está en los límites del parámetro se mostrará un mensaje el cual indicará que el valor a ser ingresado para la calibración es ilegal. Todo este procedimiento de calibración se lo va a realizar para cada uno de los submenús de acuerdo al orden jerárquico, es decir va a ser un procedimiento repetitivo por lo que se realizaron subrutinas con el fin de ser llamadas y así que no ocupen gran cantidad de memoria en el micro controlador.

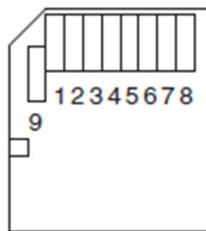
### 2.3.2 Manejo de la memoria SD

Para el manejo de la memoria MMC (Multi Media Card) en la que se encuentra la memoria SD, debemos comprender bien su funcionamiento, la memoria.

### 2.3.2.1 Introducción a la memoria SD y Conexión

La memoria SD es una memoria de tipo Flashes es decir que tienen gran capacidad, no es volátil y se puede sobrescribir cuantas veces se quiera, es usada normalmente en dispositivos móviles para almacenar los datos como en las cámaras, computadoras, laptops, sistemas de posicionamiento global, etc. Las memorias SD se dividen a su vez en mini, micro y la versión estándar que vamos a usar en el dispositivo para almacenar los datos que adquirimos de la sonda.

Las características básicas de la memoria SD es que su tamaño es de 32 x 24 x 2.1 mm, y su voltaje funcional es en el rango de 2.7 a 3.6 voltios y usa una interfaz de transferencia de datos SPI o SD y tienen 9 pines de comunicación, tiene velocidades de transferencia de 4x, 16x, 40x, 66x.



**Figura 29.** Distribución de pines de la memoria SD.

Según la Figura 29, podemos ver la distribución de los pines y a continuación en la Figura 30 vamos a indicar la función de cada pin.

Pin	Nombre	Modo SD	Modo SPI
1	Cd/Dat3/CS	Línea de datos 3	Chip Select
2	CMD/Data in	Comando/Respuesta	Comunicación, comandos y datos
3	VSS	Tierra	Tierra
4	VDD	Voltaje	Voltaje
5	CLK	Clock	Clock
6	VSS2	Tierra	Tierra
7	DAT0	Línea de datos 0	Tarjeta hacia el host de datos y status
8	DAT1	Línea de datos 1	Reservado
9	DAT2	Línea de datos 2	Reservado

**Figura 30.** Pines de la SD

Como podemos ver la distribución de los pines es realmente simple, y como en el micro controlador se utiliza una comunicación de datos de tipo SPI, debemos tener en cuenta los pines en dicha configuración, de los cuales vamos a usar:

- Datos de salida
- Datos de entrada
- Clock
- Chip Select

Que también pueden ser llamados:

- MOSI – Master output, slave input
- MISO – Master input, slave output
- SCLK – Serial Clock
- SS – Slave select

Aparte tenemos que conectar los pines 4 y 6 a tierra y el pin 3 a un voltaje de 3.3

### 2.3.2.2 La memoria SD y la comunicación SPI

Antes de continuar con las bases de la comunicación de la memoria SD y su comunicación mediante el protocolo SPI vamos a indicar las bases del protocolo SPI.

#### 2.3.2.2.1 SPI

El protocolo de comunicación SPI (Serial Peripheral Interface) es un estándar de comunicaciones que normalmente se utiliza para transferir datos entre circuitos

integrados y un equipo electrónico, este protocolo básicamente se encarga de enviar un flujo de bits que está regulado por el clock del mismo, como vimos antes tiene 4 pines o líneas de comunicación importantes que son las de Dato de salida para enviar los datos al equipo, Dato de entrada por donde recibe el flujo de datos, CS(Chip Select) que se encarga de conectar o desconectar el dispositivo, y finalmente el Clock que regula los tiempos de las transmisiones de datos.

Su funcionamiento básico es el siguiente:

El clock marca la sincronización, en cada pulso del reloj se lee un bit o se envía un bit, el MOSI recibe los datos del microcontrolador en este caso, el MISO envía los datos hacia el micro controlador y el SS, le dice a la SD que los datos son para ella.

El flujo de bits es enviado de manera síncrona con los pulsos del reloj, con cada pulso el Master envía un bit, para que funcione bien se tiene que enviar a 0 el SS y con esto el dispositivo SD se activa para funcionar y empieza la transmisión de datos.

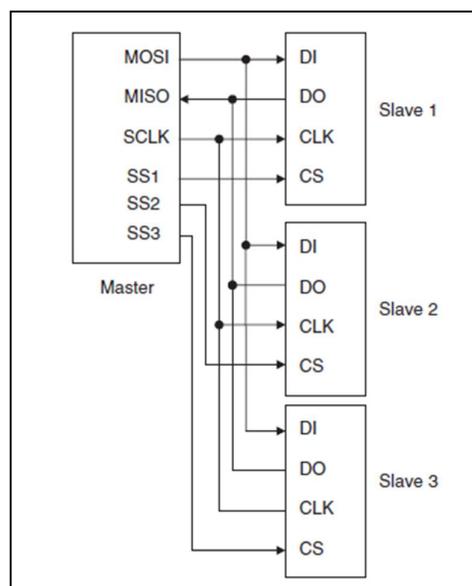


Figura 31. Esquema básico de una conexión SPI.

#### 2.3.2.2.2 Comunicación con la memoria SD

Como vimos antes para que la SD funcione debemos enviar a 0 lógico el Chip select (CS o SS), esto se realiza cuando conectamos la SD a la alimentación, para que se active la SD y cambie de modo SD a modo SPI, esto lo realizamos automáticamente con las librerías para MMC o FAT 16 de MikroC for Pic.

Una vez que está en modo SPI se requieren manejar los cuatro registros de la Sd que nos dan la información acerca de su estado, estos registros son:

- **CID**(Card identification register) Registro de la tarjeta de identificación
- **CSD**(Card specific data register) Registro de datos específicos de la tarjeta
- **SCR**(SD configuration register) Registro de configuración de la SD
- **OCR**(Operation control register) Registro de control de operaciones

### 2.3.2.3 Software para inicializar, crear archivos y escribir en la SD

En la figura 32 se ilustra el diagrama de bloques que se utiliza para grabar datos en la SD y en el cual está basado el programa descrito a continuación(Figura 32).

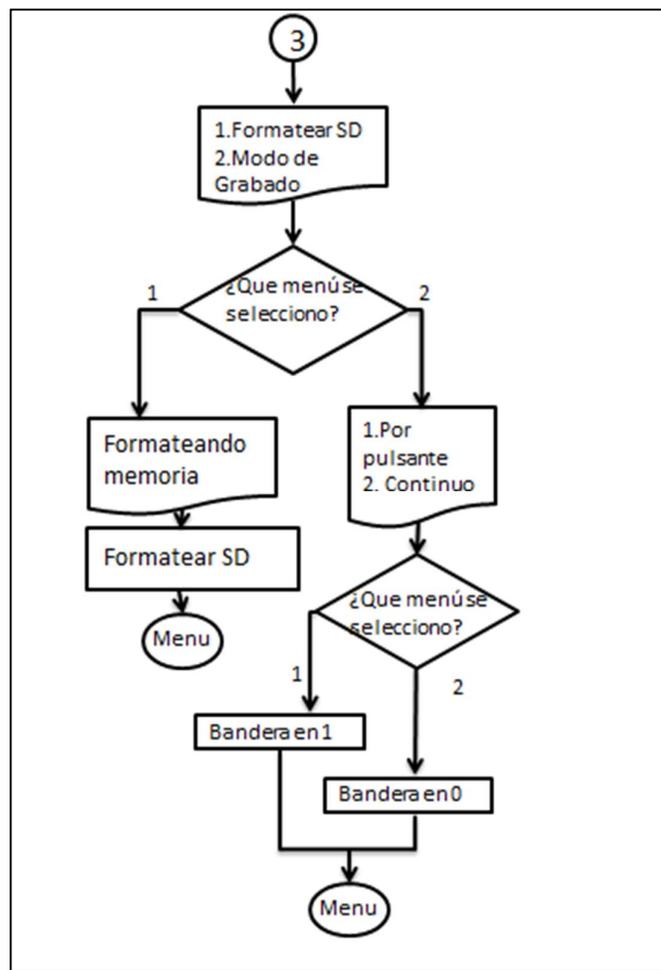


Figura 32. Diagrama de bloques del funcionamiento de la SD.

El software para la parte de la memoria SD va a estar conformado de 3 partes como la declaración de variables, subrutina de inicialización de la SD, y programa general.

### 2.3.2.3.1 Declaración de las variables

En esta parte vamos a declarar las variables que se requieren para el funcionamiento correcto de la SD, como se indica en la Figura 33.

```
// MMC module connections  
  
sfr sbit Mmc_Chip_Select at Ra0_bit;  
sfr sbit Mmc_Chip_Select_Direction at TRISa0_bit;
```

Figura 33. Declaración de las variables de la memoria SD.

Realizada esta declaración indicamos que el pin del Chip Select va a conectarse al puerto RA0, aparte como es un protocolo SPI la conexión externa de los pines es la siguiente:

- RA0 CS.
- RC7 Dato salida.
- RB0 Dato entrada.
- RB1 Clock.

Con estas variables declaradas y la conexión correcta podemos hacer uso de las librerías y subrutinas que existen en MikroC.

### 2.3.2.3.2 Declaración de variables de uso general para la SD

En estas variables se encuentran las variables del nombre del archivo a crear, o el mensaje que vamos a grabar, ya que esto cambia constantemente es necesario que sean declaradas después de las variables de la memoria SD, como se ve en la Figura 34.

```

· int contador = 0;
- char num[5];
· char txt[]="Grabar SD";
· char      mmc_error;
· char      filename[14]; // = "MIKRO00x.TXT";
· char      *resultado;
30 char      mensaje[] = "\n\r*Prueba número:\r\n";

```

Figura 34. Declaración de las variables de uso general para la SD.

### 2.3.2.3.3 Programa principal de la SD

A continuación vamos a explicar básicamente el programa principal para el control, grabado y creación de archivos de una memoria Sd.

Como se ve en la Figura 35, el comando de SPI1\_Init y sus parámetros respectivos da inicio a la comunicación SPI en la velocidad más baja posible por el oscilador externo, empezando así la comunicación con la SD.

```
SPI1_Init_Advanced(_SPI_MASTER_OSC_DIV64, _SPI_DATA_SAMPLE_MIDDLE, _SPI_CLK_IDLE_LOW, _SPI_LOW_2_HIGH);
```

Figura 35. Inicio del módulo SPI

### 2.3.2.3.4 Comprobación si está conectada la SD y se inició correctamente

Para comprobar si se inició correctamente la SD y que esta formateada en formato FAT utilizamos la subrutina de la librería de MMC\_FAT, llamada Mmc\_Fat\_Init() que aparte de iniciar la SD ya no el módulo SPI nos devuelve un valor de 0 si fue correcta la inicialización o un valor de 1 a 255 si ocurrió un error, como se puede ver en la Figura 36, si no se inicializo mostramos un mensaje de No SD y ahí termina el programa ya que al no haber SD no hay como continuar; Pero si existe una memoria SD procedemos a llamar a las subrutinas de inicializar el módulo SPI a una velocidad superior ya que si nos comunicamos a la SD podemos aumentar su velocidad para no hacer más lento el programa principal y creamos el primer archivo y le agregamos una línea al mismo.

```

if (!Mmc_Fat_Init()) {
    SPI1_Init_Advanced(_SPI_MASTER_OSC_DIV4, _SPI_DATA_SAMPLE_MIDDLE, _SPI_CLK_IDLE_LOW, _SPI_LOW_2_HIGH);
    GrabaSD();
    M_Open_File_Append();
    PortE.f0=0;
}
else {
    PortE.f1 =1;
    PortE.f0 =1;
    Lcd_Out(1,1, "No SD");
    Lcd_Out(2,1, "Reinicie");
}
}

```

Figura 36. Comprobación de si esta la SD

### 2.3.2.3.5 Estructura del programa para agregar líneas al archivo creado

Como se puede apreciar en la Figura 37, el programa es muy sencillo simplemente preguntamos si el puerto RB3 está en 0 es decir presionando el pulsante y llamamos a la rutina de M\_Open\_File\_Append() que nos agrega líneas al archivo.

```

}
else if(PortB.f3 == 0){
    M_Open_File_Append();
    PortE.f2 =1;
    PortE.f1 =0;
}
}

```

Figura 37. Estructura If agregar líneas al archivo.

### 2.3.2.3.6 Rutinas de Graba SD

Esta rutina se encarga de abrir o crear un archivo nuevo si no existe en la SD en el directorio raíz, como se ve en la Figura 38.

```

void GrabaSD () {
    int bitsnombre;
    IntToStr(contador, num);
    resultado = strcat(num, ".txt", 4);
    bitsnombre=strlen(resultado);
    filename[bitsnombre] = resultado;
    Mmc_Fat_Assign(filename, 0xA0);
    Delay_ms(550);
}

```

Figura 38. Rutina de Graba SD.

Como se puede ver primero inicializamos una variable local llamada bitsnombre que contiene la longitud en bits del nombre del archivo que vamos a crear, de ahí del contador que teníamos lo volvemos un string con la función IntToStr que nos convierte un numero en una cadena de caracteres y este resultado lo guardamos en otra variable llamada num.

Y al inicio del programa principal definimos una variable resultado esta variable va a tener el nombre del archivo a crear ya con su extensión .txt o la que nosotros deseemos ya que tenemos el nombre del archivo como string solo le pones la extensión “.txt” con la función strcat. Es decir si es la primera vez que vamos a crear un archivo, nuestra variable contador va a estar en 0, lo que al convertirlo en una cadena de caracteres va a hacer que num tenga el valor de “0” y que resultado tenga esto “0.txt”.

Ahora vamos a contar la longitud de bits de resultado y lo almacenamos en la variable local que definimos al inicio es decir en bitsnombre, que para este ejemplo va a tener 5 bits y pasamos este resultado a la variable de filename que va a ser el nombre final del archivo a crear. Para crear el archivo vamos a usar la función Mmc\_Fat\_Assing(nombre, bits), con la que creamos una asignación del archivo que queremos crear, si no existe este archivo lo crea, pero todo esto depende de cómo estén los bits, la distribución de los bits se puede ver en la Figura 39.

Bit	Mask	Description
0	0x01	Read Only
1	0x02	Hidden
2	0x04	System
3	0x08	Volume Label
4	0x10	Subdirectory
5	0x20	Archive
6	0x40	Device (internal use only, never found on disk)
7	0x80	File creation flag. If file does not exist and this flag is set, a new file with specified name will be created.

**Figura 39.** Bits para el MMC\_Fat\_Assing.

Como se puede ver en este caso al asignar el archivo ponemos de bits 0xA0 que corresponde en binario a 10100000, lo que quiere decir que vamos a tener un archivo por el bit 5 que está en 1 y que si el archivo no existe lo crea por el bit 7 en 1. Y así es como creamos un archivo nuevo si no existe o abrimos el archivo si ya existe, si quisiéramos borrar el contenido del archivo abierto solo tendríamos que usar la instrucción

MMC\_Fat\_Rewrite() que nos borra el contenido del archivo.

### 2.3.2.3.7 M\_Open\_File\_Append()

Esta rutina se encarga de agregar un mensaje de texto al archivo que hemos asignado con el MMC\_Fat\_Assing de la rutina de GrabaSD(), en la Figura 40 se puede ver esta rutina y vamos a explicarla a más detalle ahora.

```
void M_Open_File_Append() {
    int Bmensaje;
    Mmc_Fat_Set_File_Date(2010, 4, 19, 9, 20, 0);
    Mmc_Fat_Append();
    Bmensaje=strlen(mensaje);
    // Prepare file for append
    Mmc_Fat_Write(mensaje,Bmensaje );
}
```

Figura 40. Rutina de M\_Open\_File\_Append();

Iniciamos la rutina definiendo una variable local llamada Bmensaje que va a tener los bits del mensaje, usamos la función Mmc\_Fat\_Set\_File\_Date para ponerle una fecha de modificación al archivo, esto no es necesario pero es útil por si tenemos un reloj interno para poner la fecha y hora correcta en la que modificamos el archivo, de ahí usamos Mmc\_Fat\_Append que se encarga de preparar el archivo para recibir nuevas líneas de texto, es decir con este comando abrimos el archivo y nos posicionamos en el último bit de escritura.

Ahora obtenemos la longitud en bits del mensaje para escribir solo el mensaje ya que si ponemos una cantidad errónea de bits no se puede escribir completo el mensaje o podemos insertar caracteres no deseados al final del mensaje, esta longitud la almacenamos en la variable Bmensaje que definimos antes y ahora finalmente usamos el comando Mmc\_Fat\_Write con los datos del mensaje y de los bits, esto nos escribe el mensaje al final del archivo como deseábamos.

Como el mensaje es un string es bueno que conozcamos algunos caracteres especiales que son útiles al momento de escribir el mensaje, como lo son el dar un cambio de línea, escribir una tabulación y otros, estos caracteres se pueden apreciar en la Figura 41. Que tiene la tabla de todos los caracteres de escape como también son conocidos de las cadenas de caracteres.

Sequence	Value	Char	What it does
<code>\a</code>	0x07	BEL	Audible bell
<code>\b</code>	0x08	BS	Backspace
<code>\f</code>	0x0C	FF	Formfeed
<code>\n</code>	0x0A	LF	Newline (Linefeed)
<code>\r</code>	0x0D	CR	Carriage Return
<code>\t</code>	0x09	HT	Tab (horizontal)
<code>\v</code>	0x0B	VT	Vertical Tab
<code>\\</code>	0x5C	\	Backslash
<code>\'</code>	0x27	'	Single quote (Apostrophe)
<code>\"</code>	0x22	"	Double quote
<code>\?</code>	0x3F	?	Question mark
<code>\O</code>		any	O = string of up to 3 octal digits
<code>\xH</code>		any	H = string of hex digits
<code>\XH</code>		any	H = string of hex digits

**Figura 41.** Tabla de caracteres especiales para string.

### 2.3.3 Reloj en tiempo real

El reloj en tiempo real se lo va a realizar siguiendo el diagrama de bloques ilustrado en la Figura 42.

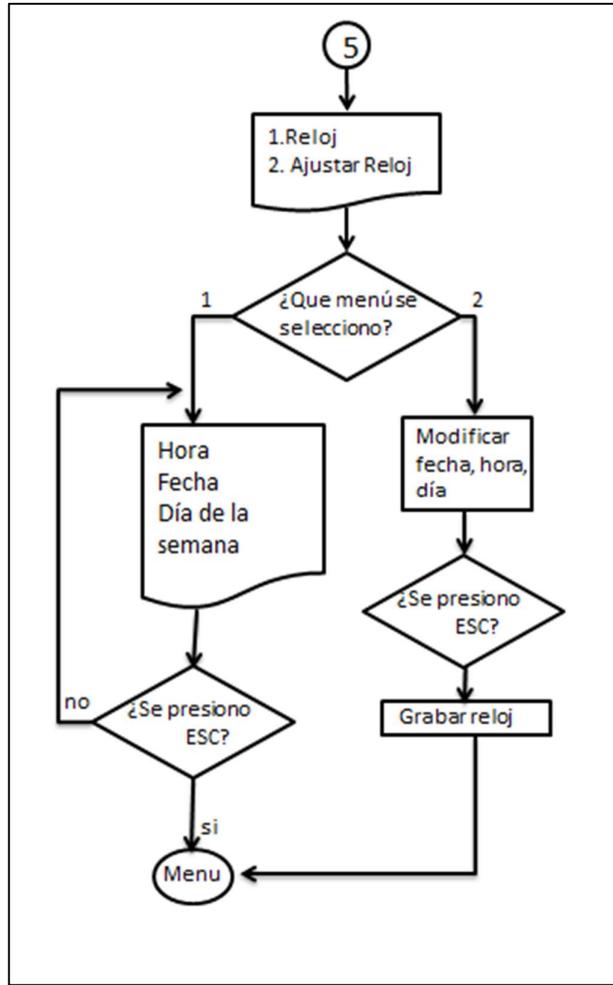


Figura 42. Diagrama de bloques del reloj en tiempo real.

El circuito a realizar para obtener el reloj en tiempo real está dispuesto en la Figura 43, aquí se indican los componentes a utilizar, ya que se trata de un integrado DS1307 y un oscilador de 32.768 KHz, además de resistencia. En esta figura también se explica la conexión realizada hacia el micro controlador y al LCD respectivamente, además de los pulsantes para la configuración del tiempo.

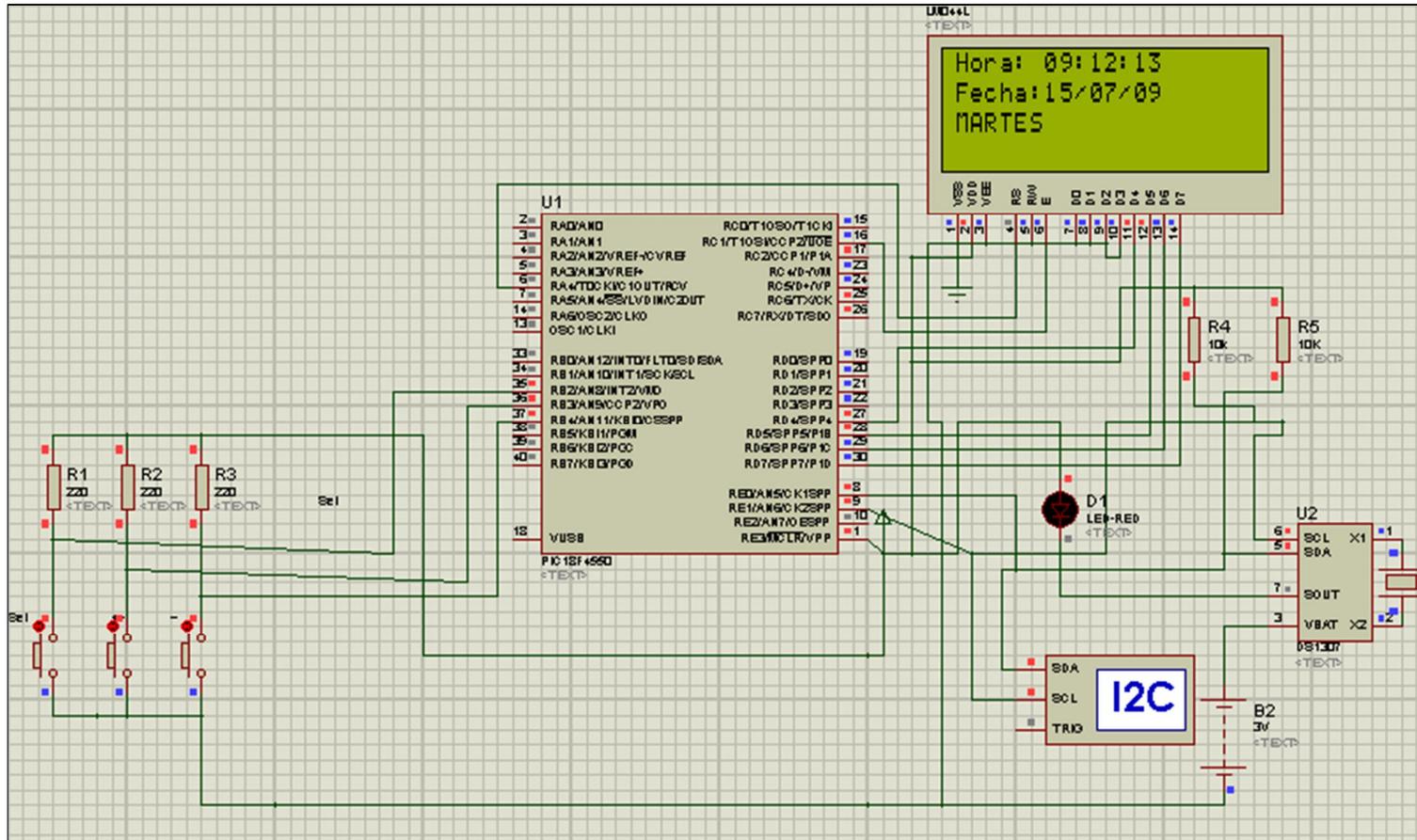


Figura 43. Conexión realizada para el reloj en tiempo real.

Al igual que en los menús anteriormente mencionados, el ingreso a el menú Reloj, se da pulsando el botón “Ent” sobre el previamente submenú reloj, seleccionado.

Este menú, como el diagrama de bloques indica, consta de dos submenús, los cuales detallaremos a continuación.

### 2.3.3.1 Reloj

En esta subrutina lo que se indica es el tiempo exacto que el microcontrolador, recibe del integrado DS1307, para realizar esta acción necesitamos de ciertas subrutinas para obtener y decodificar este valor.

```

· void Leitura_RTC() {
·     //En esta subrutina se leen los datos desde el DS1307 y se los asigna a cada variable
·     Soft_I2C_Start(); // se envía señal de Start
180 Soft_I2C_Write(0xD0); // Dirección del DS1307
·     Soft_I2C_Write(0); // Se empieza en la dirección 0
·     Soft_I2C_Start(); // Se envía una señal repetida
·     Soft_I2C_Write(0xD1); // Dirección de R/W
·     ucSegundo = Soft_I2C_Read(1); // Se lee el valor de segundos
·     ucMinuto = Soft_I2C_Read(1); // Se lee el valor de (minutos)
·     ucHora = Soft_I2C_Read(1); // Se lee el valor de (horas)
·     ucDia_Semana = Soft_I2C_Read(1); // Se lee el valor del día de la semana
·     ucDia = Soft_I2C_Read(1); // Se lee el valor de día
·     ucMes = Soft_I2C_Read(1); // Se lee el valor del mes
190 ucAño = Soft_I2C_Read(0); // Se lee el valor de año
·     Soft_I2C_Stop(); // Envía señal de stop
· }

```

Figura 44. Lectura de datos del integrado DS1307.

Como se muestra en la Figura 44, la lectura de datos se da mediante el protocolo de comunicación I2C, en este caso es por software, ya que los pines de uso directo, ya están ocupados. En esta subrutina se leen todos los datos, como año, día, mes, hora, etc. Para la decodificación, utilizamos la subrutina indicada en la Figura 45, esta decodifica el valor que obtenemos anteriormente.

```

// Rutina de conversión de datos para BCD
void Converte_BCD(unsigned char ucLinha, unsigned char ucColuna, unsigned char ucValor) {
    volatile unsigned char ucValor1, ucValor2;
    ucValor1 = (ucValor >> 4) + '0'; // Converte el primer nibble en BCD y despues en
    Lcd_Chr(ucLinha, ucColuna, ucValor1); // Escribe el caracter en el LCD
    ucValor2 = (ucValor & 0x0F) + '0'; // Converte el primer nibble en BCD y despues en
    Lcd_Chr_CP(ucValor2); // Escribe el caracter en el LCD
}

```

Figura 45. Decodificación de los datos obtenidos del DS1307.

Una vez obtenidos los datos simplemente se muestran en la pantalla del LCD, como ya lo indicamos en el circuito de la Figura 43.

### 2.3.3.2 Ajuste reloj

Para ajustar el reloj, tendremos dos rutinas importantes, una para ingresar los datos y la otra para escribirlos en el DS1307. Para ingresar los datos, contamos con la subrutina **valck**, con esta subrutina obtenemos un valor, y este valor lo enviamos mediante el protocolo I2C, con la subrutina **Grava\_RTC**, como se ilustra en la Figura 46, inicializamos la comunicación y enviamos todos los datos para su sobre escritura.

```

• void Grava_RTC(){
- //En esta subrutina se envian datos de edición al clock, para ajustar la hora exacta
• I2C1_Init(100000); // Inicializa I2C con frecuencia de 100KHz
• Soft_I2C_Start(); // Inicializa a comunicación I2c
• Soft_I2C_Write(0xD0); // Dirección del DS1307
• Soft_I2C_Write(0); // Comienza comunicación para enviar primero segundos.
- Soft_I2C_Write(seg1); // Inicializa segundos.
200 Soft_I2C_Write(min1); // Inicializa minutos.
• Soft_I2C_Write(hora1); // Inicializa la hora (formato 24 horas).
• Soft_I2C_Write(semanal); // Inicializa el día de la semana
• Soft_I2C_Write(dial); // Inicializa día
- Soft_I2C_Write(mes1); // Inicializa mes
• Soft_I2C_Write(año1); // Inicializa año
• Soft_I2C_Stop();
• }

```

Figura 46. Grabar datos en el integrado DS1307.

## CAPÍTULO 3

### DISEÑO DE HARDWARE PARA LA SONDA

#### 3.1 Introducción

El hardware del analizador está compuesto de una tarjeta de control que tiene como parte fundamental el microcontrolador y sus periféricos integrados para la comunicación con la sonda, el usuario y manejo de la memoria SD, en la Figura 47 se puede apreciar el diseño final del hardware. También vamos a explicar cómo crear un proyecto en Altium Designer 2009, y la creación del proyecto desde el circuito esquemático hasta el diseño de la placa impresa, para el uso de la sonda vamos a usar un conector MS – 8 conectado como si fuera un puerto RS232 normal y las consideraciones especiales de hardware.

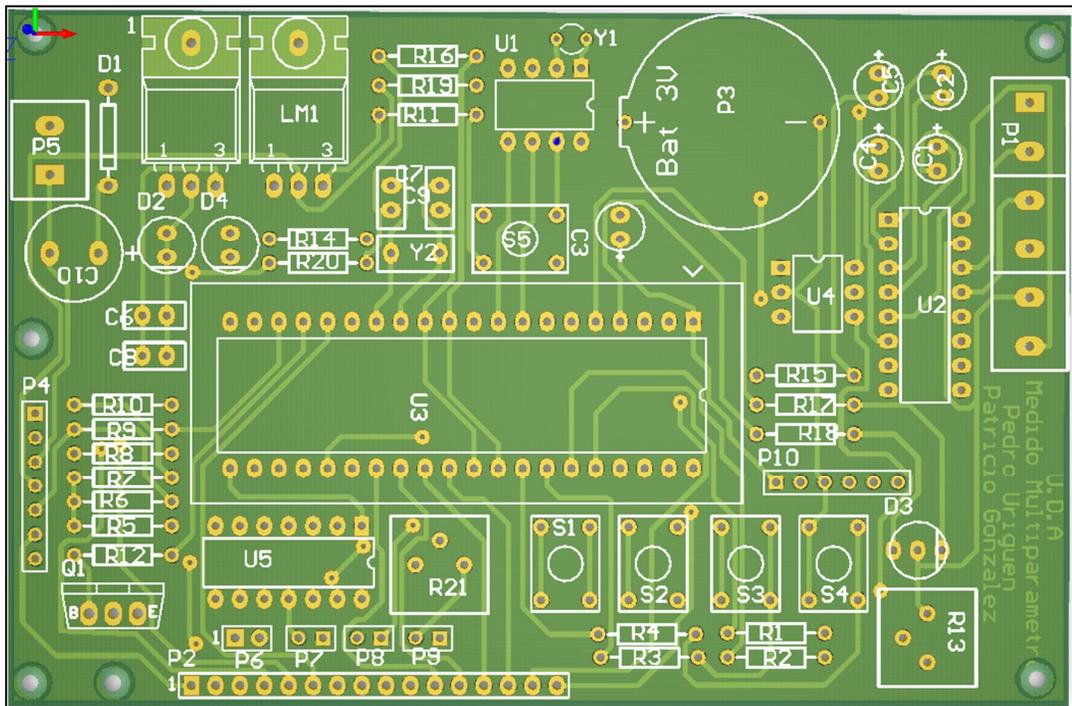


Figura 47. Hardware del equipo.

### 3.2 Creación del proyecto en Altium designer y cargar las librerías

Primero en Altium vamos a File – New – Project y seleccionamos un proyecto de tipo PCB como se puede ver en la Figura 48.

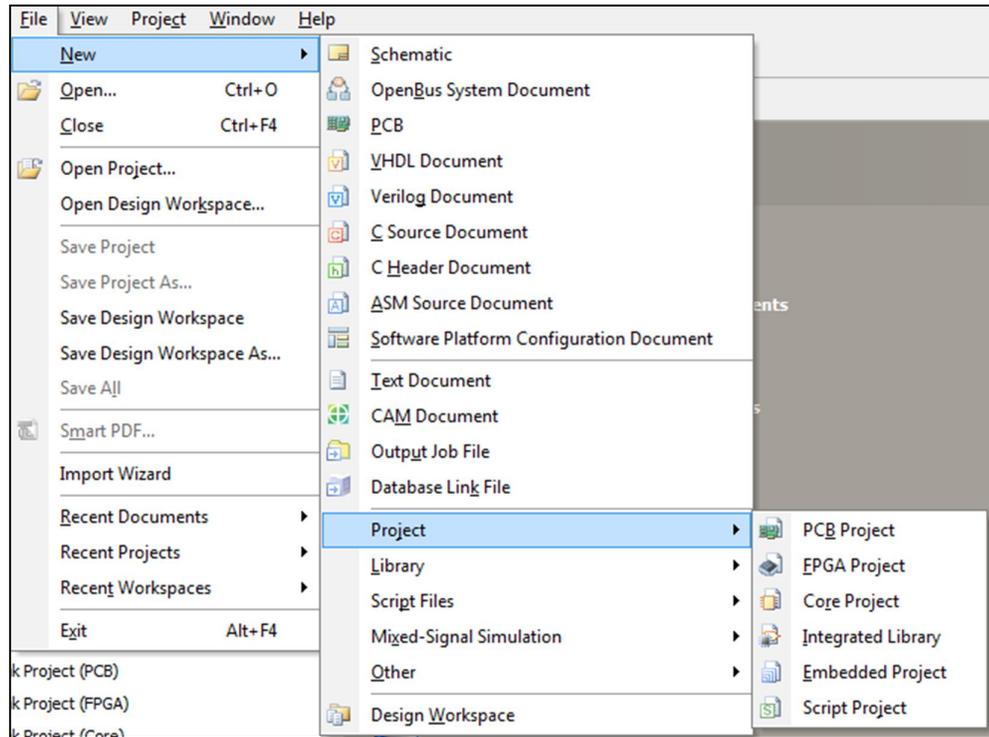
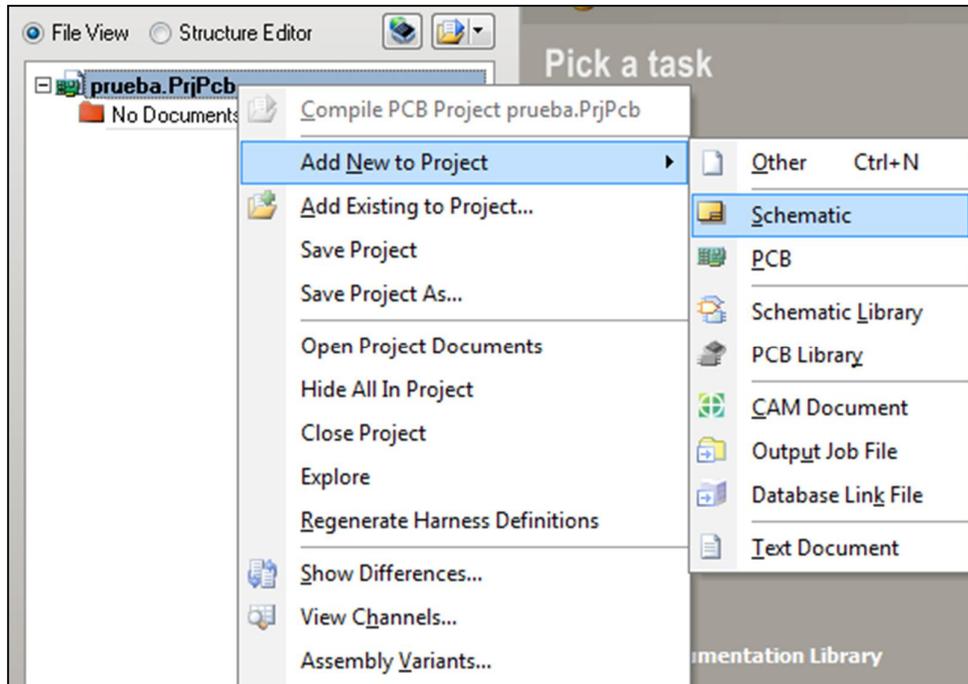


Figura 48. Creación de un nuevo proyecto en Altium.

Y a continuación se nos ha creado el proyecto, de ahí vamos a new – save Project as... y grabamos el proyecto con el nombre que queremos, y agregamos un archivo esquemático al proyecto, dando click derecho en el proyecto que creamos y elegimos agregar archivo al proyecto, como vemos en la Figura 49.



**Figura 49.** Como agregar un archivo esquemático al proyecto.

Con el archivo esquemático agregado, vamos a agregar la librerías que vamos a usar en el proyecto en este caso van a ser las de conectores misceláneos, componentes misceláneos y el de Microchip, para agregar las librerías vamos a ir a librerías en el lado izquierdo de la ventana y damos click en libraries y nos aparecerá una ventana como la que vemos en la Figura 50.

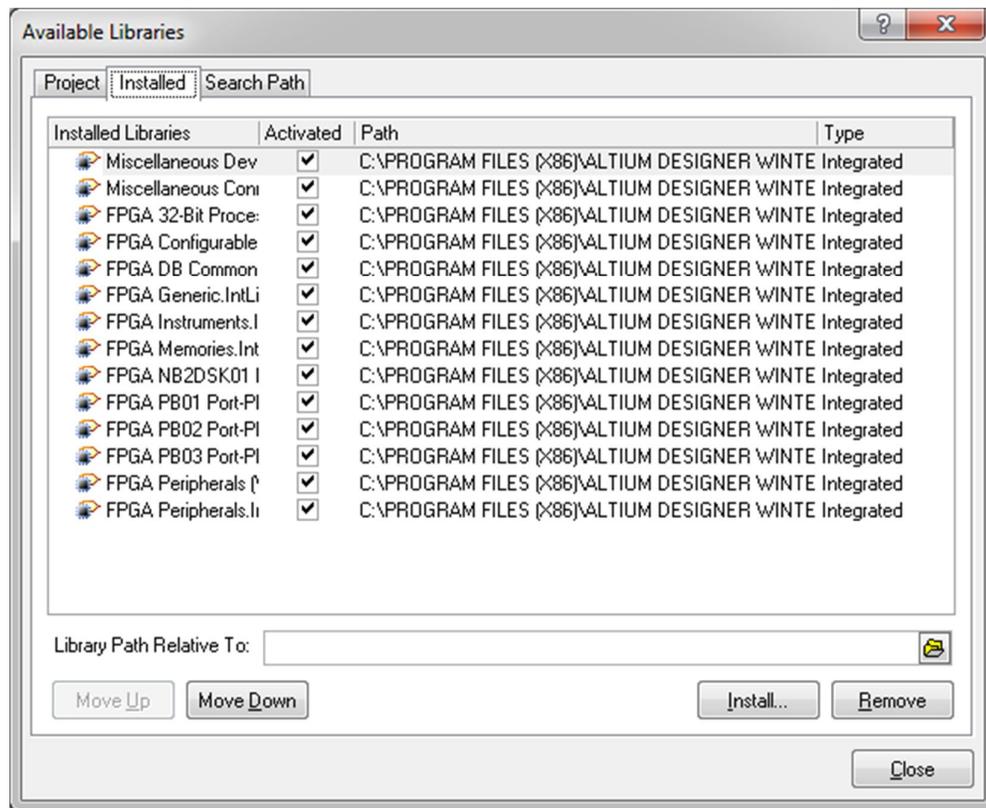


Figura 50. Ventana para agregar Librerías.

Y vamos a instalar y se nos abrirá una ventana de explorador de Windows en las que vamos a seleccionar las bibliotecas que vamos a usar y les damos en el check box de activated para que se nos actualicen.

### 3.3 Búsqueda de componentes en las librerías

En la ventana de librerías seleccionamos la librería que queremos usar y ponemos entre \* las iniciales del componente que queremos buscar por ejemplo “\*res” para resistencia como se ven en la Figura 51.

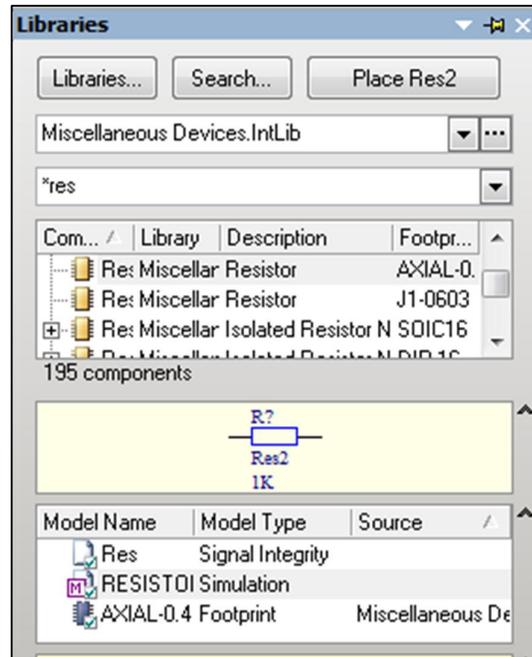


Figura 51. Búsqueda de componentes en la librería.

Y damos en place para posicionar el componente en la hoja esquemática, este proceso lo repetimos para todos los componentes que necesitemos o conectores ya que es de forma análoga, en el caso de que no encontremos un componente en específico usamos la opción de Search... de las librerías que nos abre una ventana como la de la Figura 52.

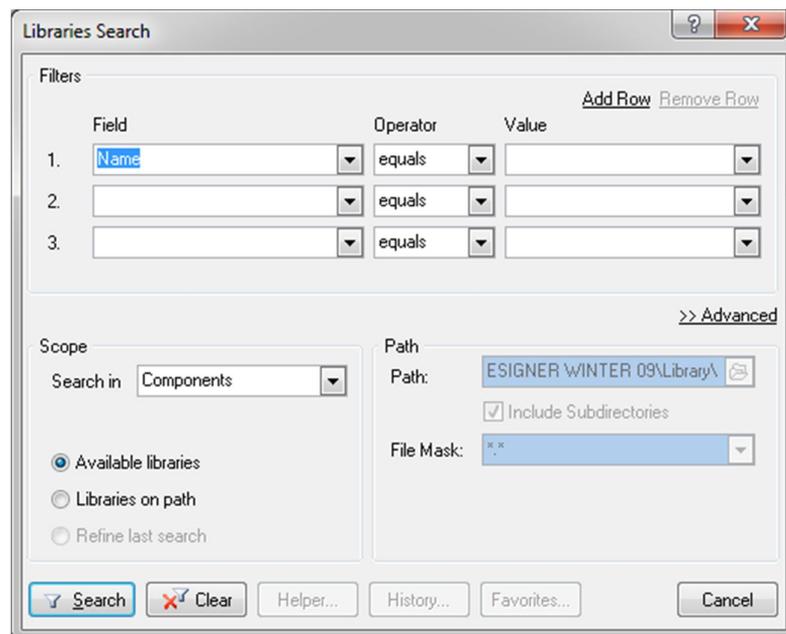
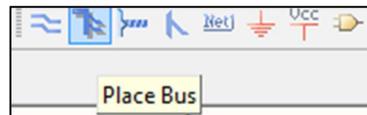


Figura 52. Ventana de búsqueda de componentes.

Y introducimos el nombre del componente que queremos buscar en el campo que dice Field, en Operator elegimos si queremos que busque por nombre igual, o que solo contenga parte del nombre que vamos a ingresar en value y esto nos devolverá como resultado los componentes que cumplan los parámetros que introducimos y lo posicionamos en la el circuito esquemático.

### 3.4 Posicionamiento de bus de datos y Net labels

Altium nos facilita la interconexión de los componentes con Netlabels y buses de datos, para poner un bus de datos vamos a las barras de iconos el de place bus como en la Figura 53 y ponemos un bus de datos alrededor de los componentes como en la Figura 54.



**Figura 53.** Icono de poner un Bus.

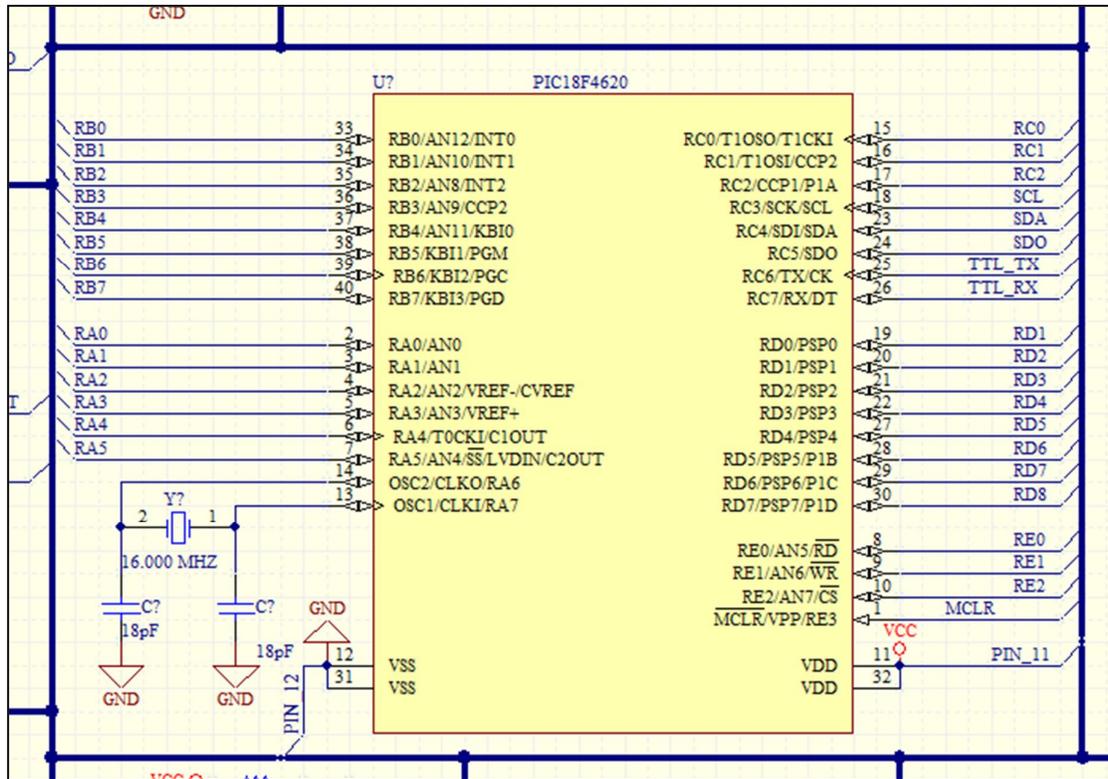


Figura 54. Ejemplo de posicionar bus alrededor de los componentes.

E interconectamos con un cable o alambre desde el componente hacia el bus, para crear un cable usamos el icono de la Figura 55.

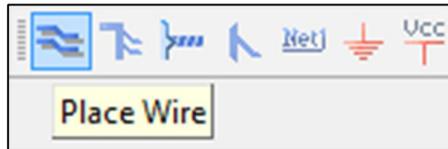


Figura 55. Poner un cable en el circuito.

Para interconectar el bus con el cable usamos el icono de Place Bus Entry como el de la Figura 56.

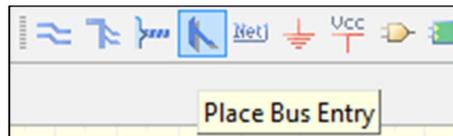


Figura 56. Poner una entrada en el Bus de datos.

Este proceso lo realizamos con todos los componentes hasta que todos los componentes estén conectados a un bus o a hacia un terminal de tierra o Vcc.

### 3.4.1 Poner conectores de tierra o voltaje

Para poner un conector de tierra vamos al icono de tierra y lo seleccionamos y lo ponemos en el componente que deseamos, hacemos el mismo proceso para los conectores de Vcc, si presionamos la tecla de TAB podemos cambiar parámetros como la figura o el color del conector, como se ve en las Figura 57, Figura 58 y Figura 59.

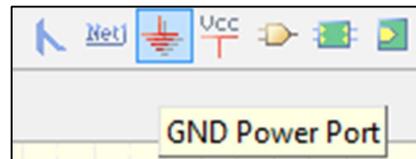


Figura 57. Posicionar conector de tierra.

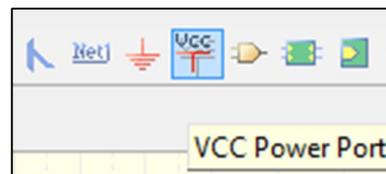


Figura 58. Posicionar conector de VCC.

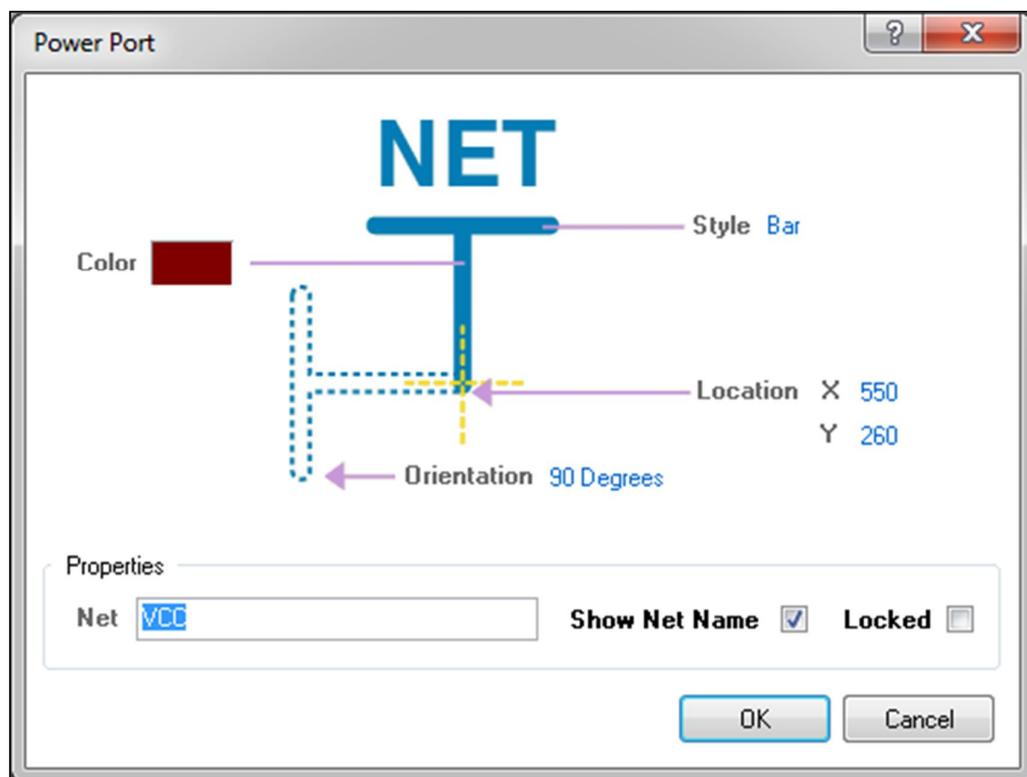


Figura 59. Ventana para cambiar parámetros del conector.

### 3.5 Posicionar Net Label

Los Net Label son para interconectar componentes y así evitar cruces y un lío de cables entre los componentes para posicionar un Net Label vamos al icono de la Figura 60. Y con la tecla de TAB entramos en el menú para cambiar el nombre del Net Label(Figura 61), un net label une todas las etiquetas con un mismo nombre el momento de rutear la placa.

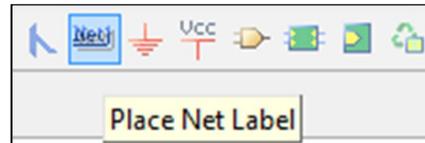


Figura 60. Posicionar Nel Label.

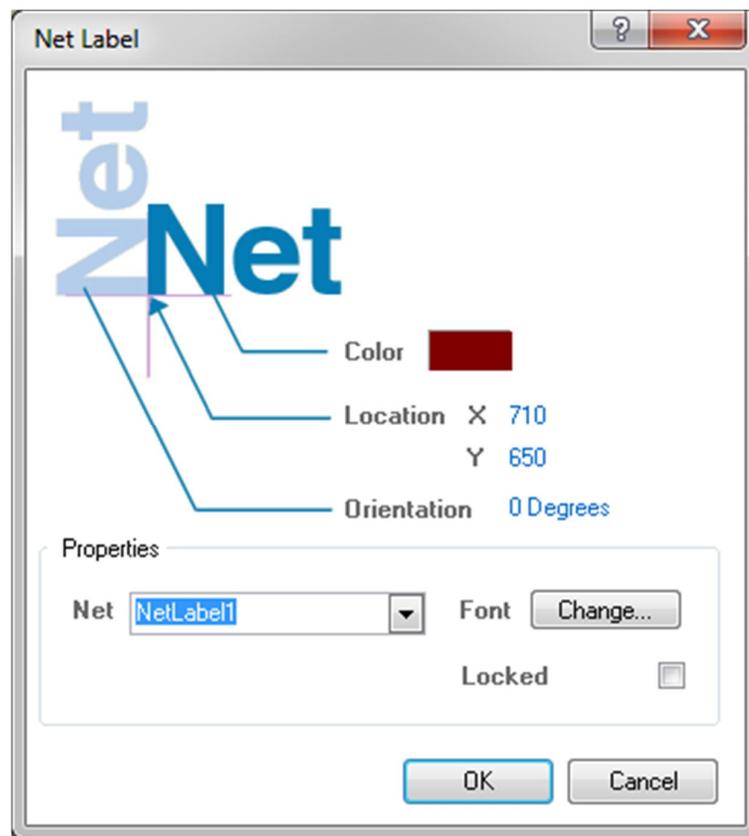


Figura 61. Menú para cambiar el nombre del Net Label.

Posicionados todos los componentes y con conexiones para Net Labels, podemos empezar a buscar los footprints para los componentes previo al diseño de la placa impresa.

### 3.6 Búsqueda de los FootPrints

En Altium ya vienen la mayoría de los footprints para cada componente, pero nunca está de más revisar que los footprints estén bien ya que un foot print erróneo nos puede llevar a que la placa este mal diseñada y no funcione correctamente o genere un cortocircuito. Para buscar un footprint primero vamos al componente y vemos sus propiedades con click derecho, propiedades y nos muestra una ventana como la de la Figura 62.

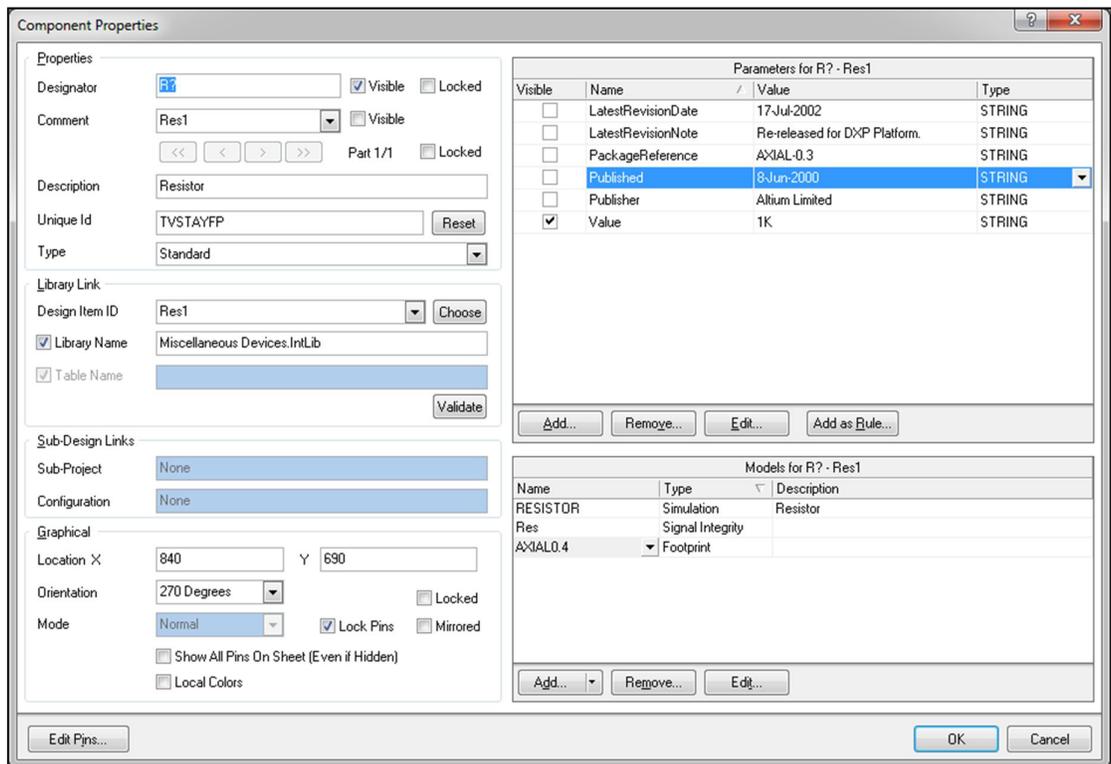


Figura 62. Ventana de propiedades del componente.

Y en la ventana en la parte inferior derecha, donde dice footprint seleccionamos y damos en Edit, que nos abre una nueva ventana para los footprints como se puede apreciar en la Figura 63.

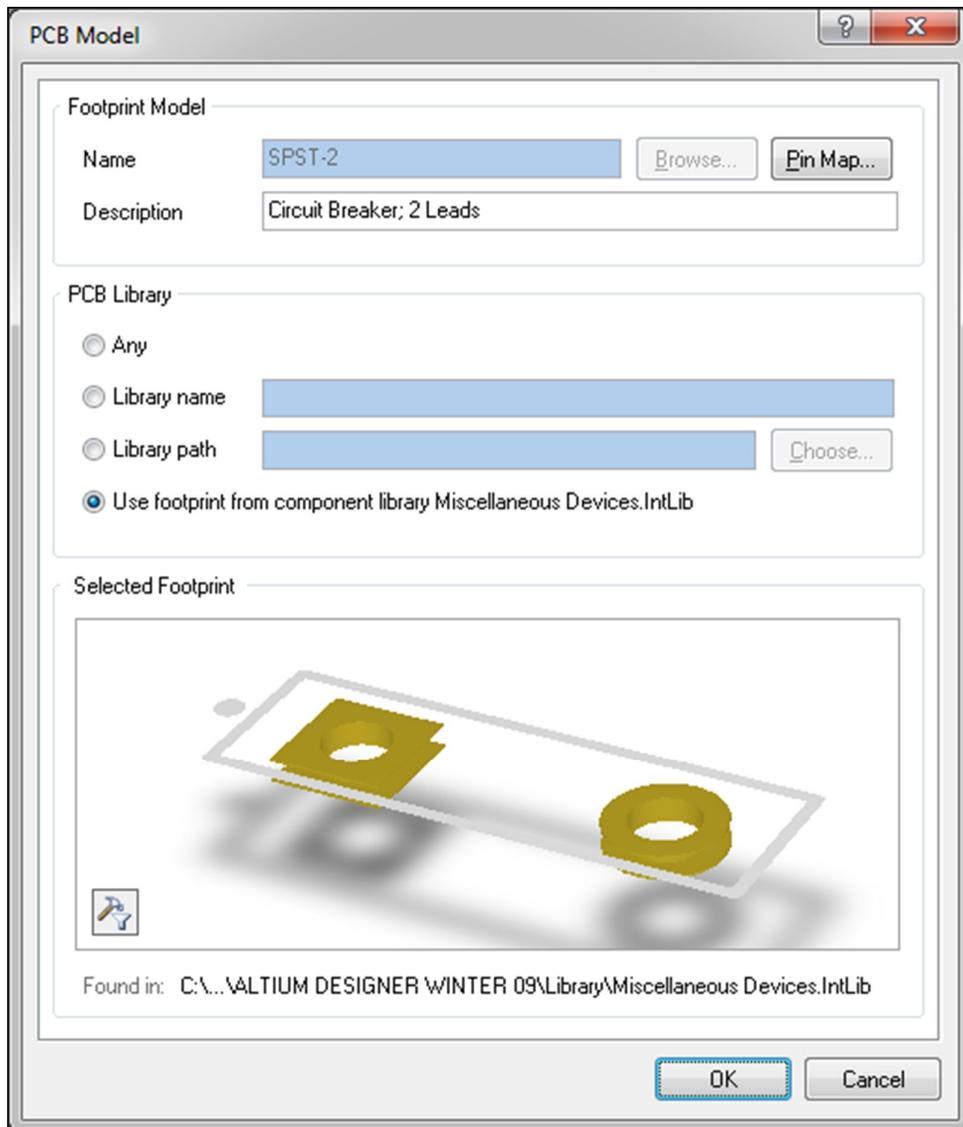
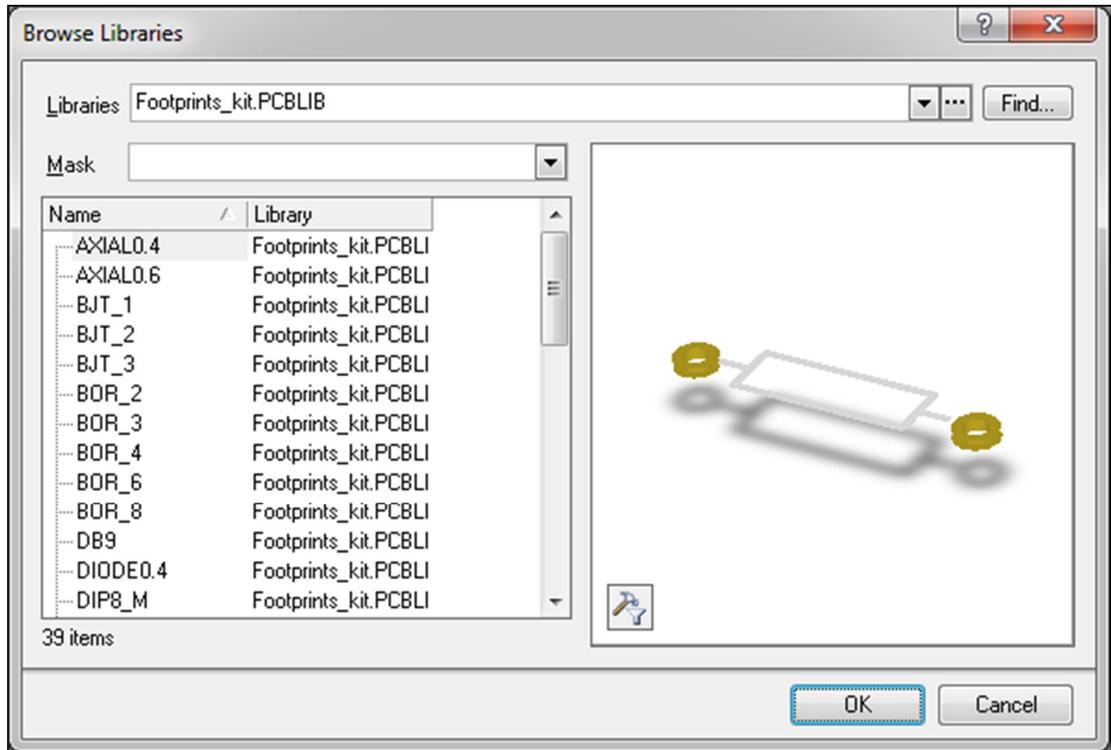


Figura 63. Ventana Edit del footprint.

Si queremos buscar otro footprint seleccionamos la opción Any de PCB Library y de ahí en Browse para ver la lista de Footprints disponibles y elegimos la que deseamos como se ve en la Figura 64.



**Figura 64.** Lista de Footprints disponibles para elegir.

Y para evitar el proceso tedioso de estar cambiando Footprints uno en uno usamos los cambios globales de los componentes, damos click derecho sobre el componente y elegimos find similar ítems como en la Figura 65.

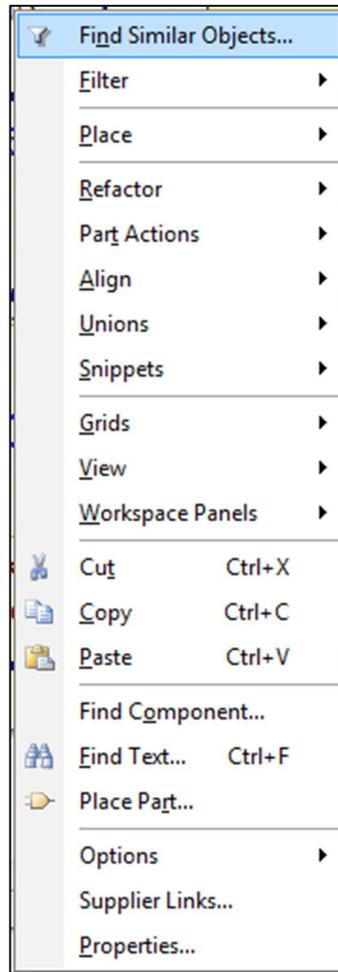


Figura 65. Opción de Find Similar ítems.

Una vez abierta, podemos ver una ventana en la que nos indica el parámetro por el cual queremos encontrar elementos similares en este caso ponemos en vez de “any” “same” como en la Figura 66, y damos en Apply y OK eso nos selecciona todos los componentes iguales a los parámetros que seleccionamos.

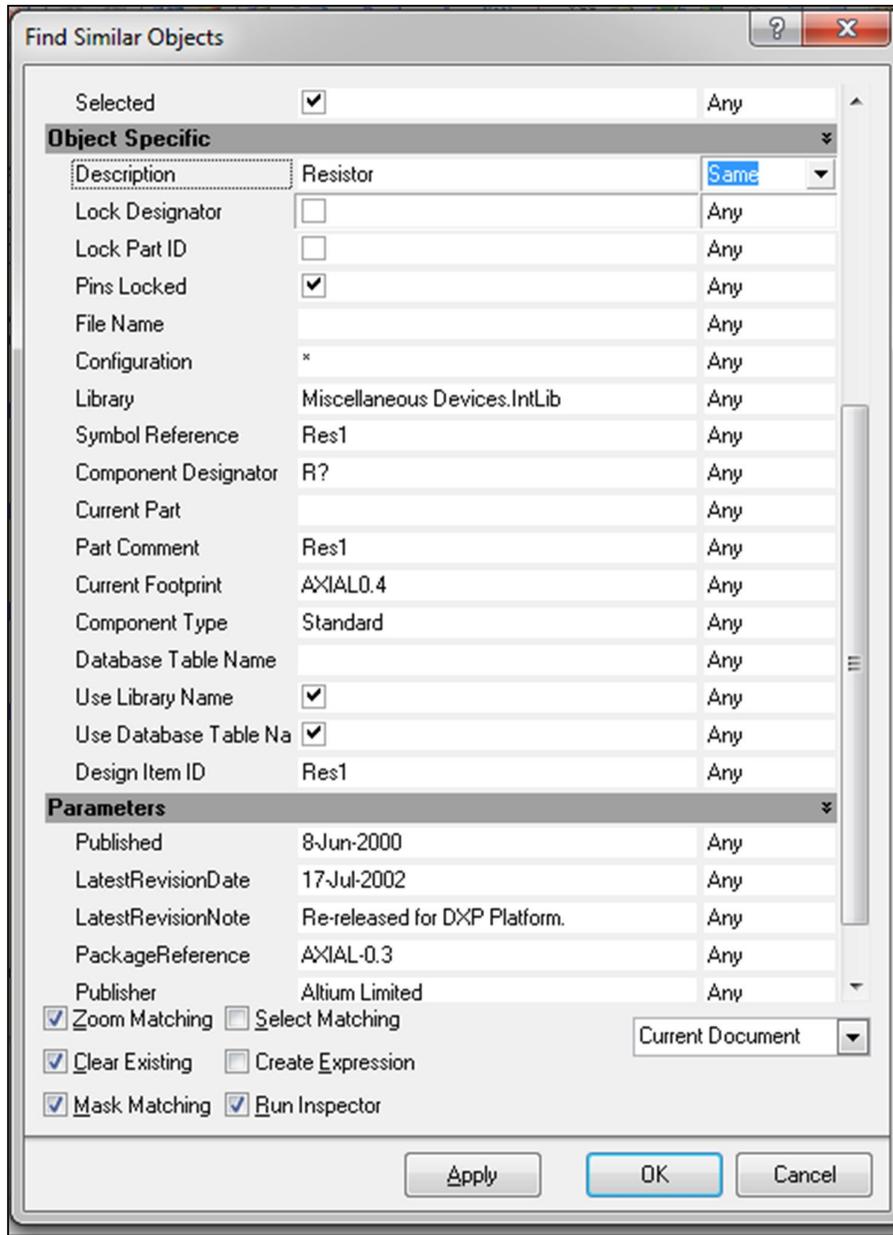


Figura 66. Ventana de Find Similar Objects

Después de que dimos Ok se seleccionan todos los objetos y nos muestra una ventana en el lateral de la ventana de Altium, cualquier cambio realizado en esta ventana se lo realiza a todos los componentes seleccionados y de esta forma cambiamos los footprints de manera global como se ve en la Figura 67.

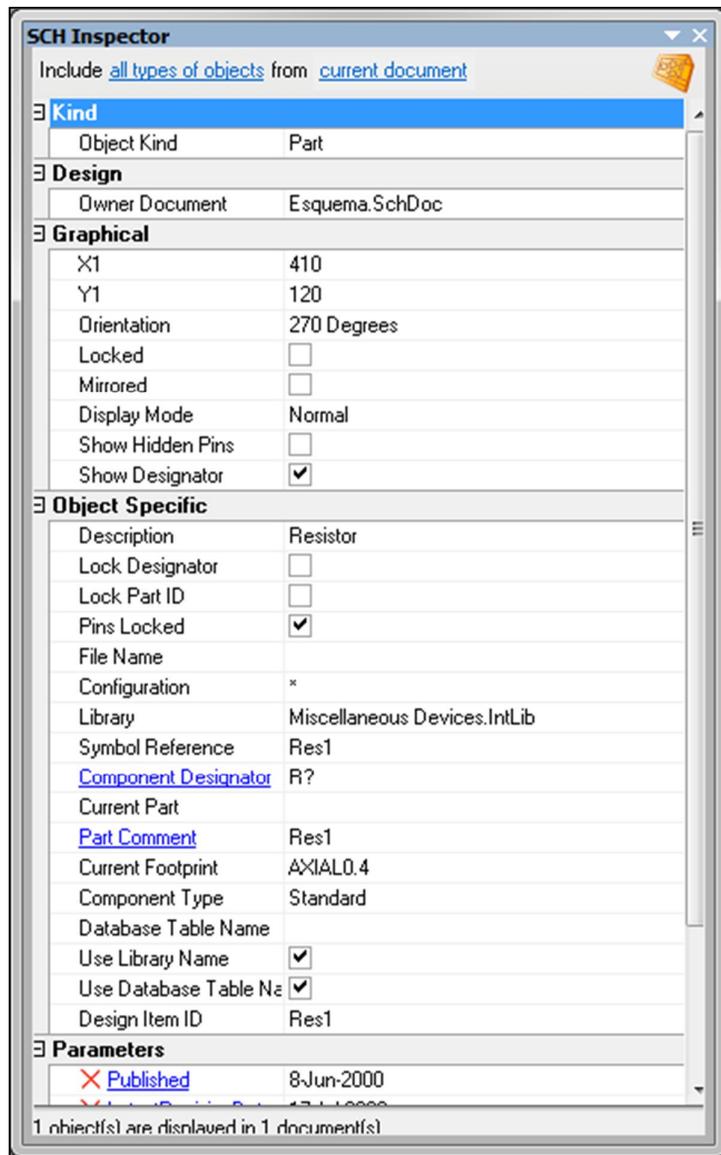


Figura 67. Ventana para realizar cambios globales.

### 3.7 Designar nombre de componentes

Los componentes cuando los ponemos en la hoja esquemática tienen nombres de C? o U?, por lo que hay que cambiar los nombres para que sean iguales a C1, U1, R1, R2, etc.

Pero realizar este cambio es un proceso largo y tedioso en el que si hacemos componente por componente perdemos tiempo para ello vamos a usar la opción de anótate de altium que se encuentra en la barra de menú, tool/annotate schematic y nos aparece una imagen como la de la Figura 68.

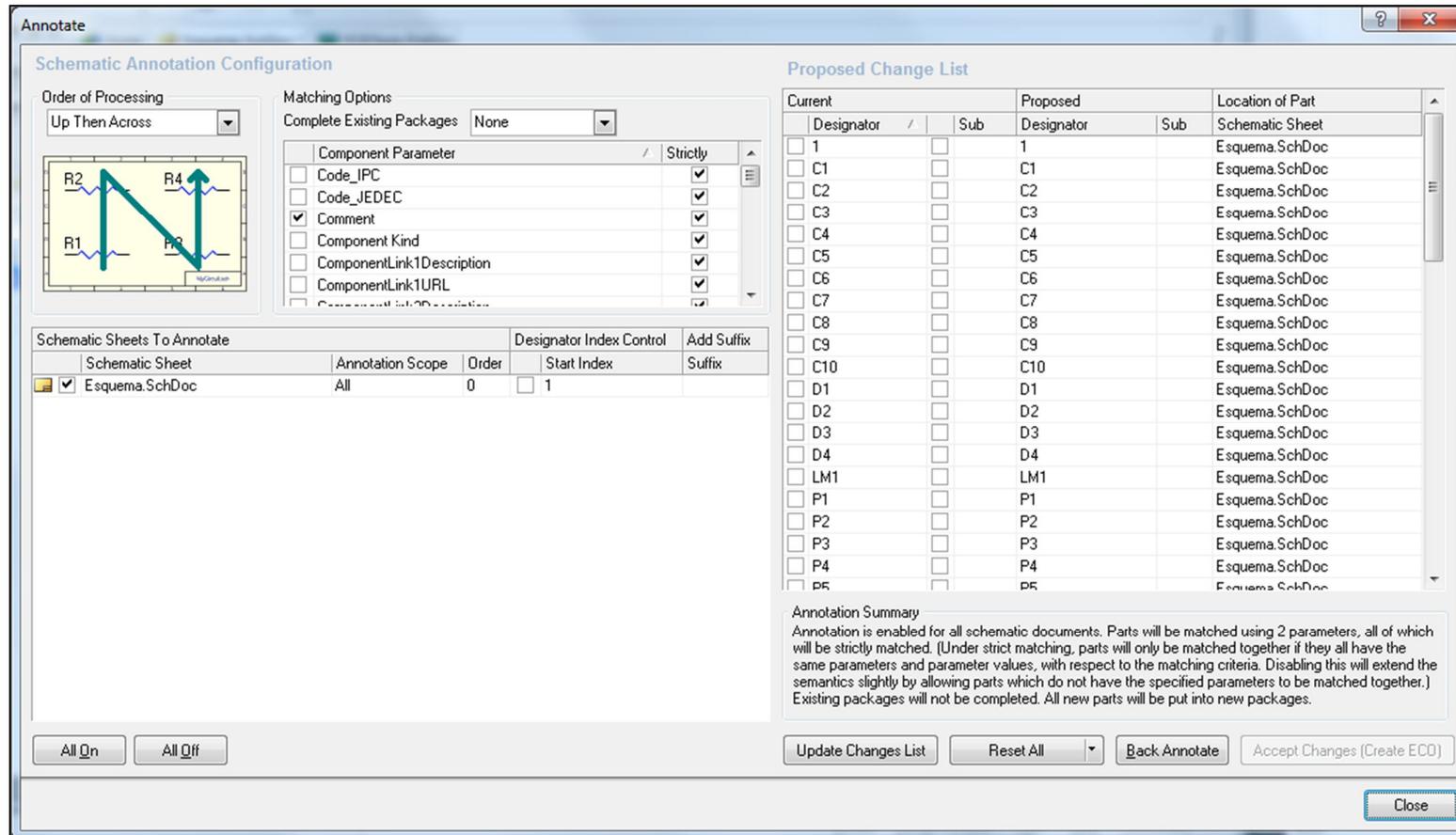


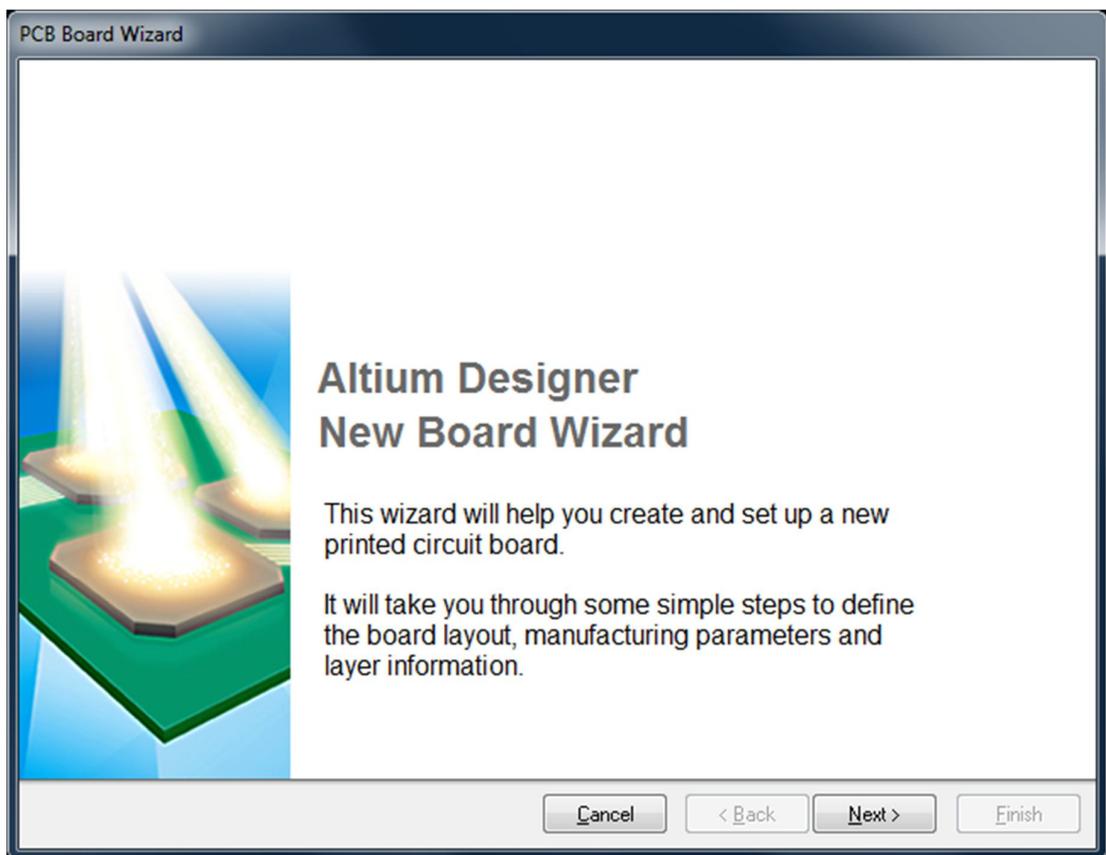
Figura 68. Ventana para realizar el annotate de todos los componentes.

De la ventana de la Figura 68 podemos elegir como queremos que enumere los componentes en qué orden y otros parámetros más seleccionamos a gusto todas esas reglas, y damos click en Update Change List, que nos va a indicar que componentes se van a cambiar y de ahí damos click en Accept Changes (Create ECO). Y de ahí nos aparece otra ventana en la que damos click en Execute Changes y ya tenemos todos los componentes renombrados y podemos seguir a la creación del PCB.

Refiérase al **anexo 3** para ver el esquema completo de todos los circuitos y conexiones del hardware.

### 3.8 Creación de PCB

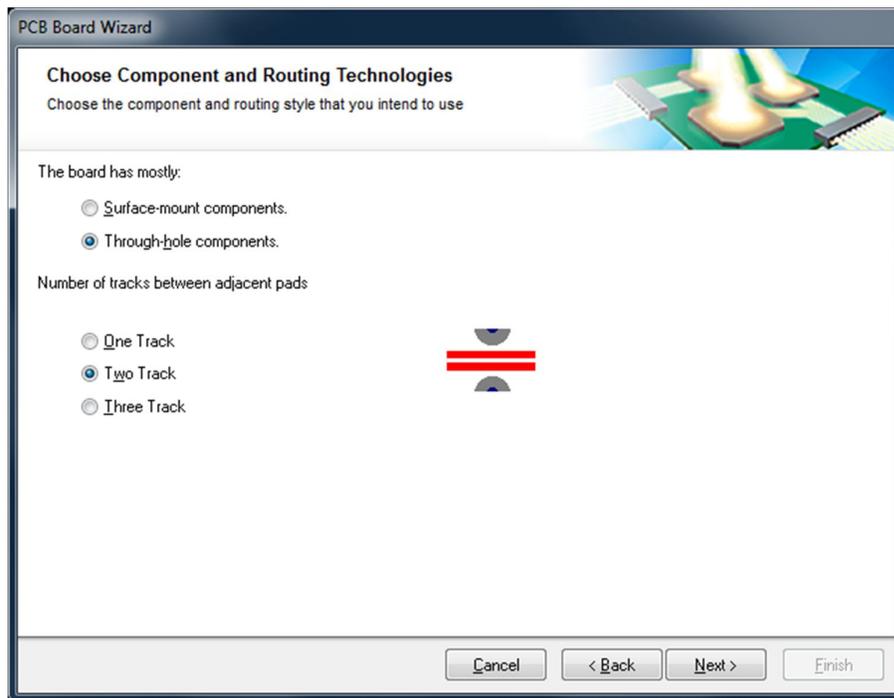
Para crear la placa PCB, vamos a usar el wizard de Altium que nos muestra una ventana como la de la Figura 69.



**Figura 69.** Primera ventana del Wizard para crear un PCB.

En este wizard vamos primero a dar next, y elegimos el sistema métrico y demás siguiente esto es para elegir las unidades que vamos a usar, en la siguiente

ventana elegimos tamaño “Custom” y damos en next, y en esta ventana elegimos el tamaño aproximado que queremos que tenga nuestra tarjeta PCB y damos next y seguimos así dando next hasta la ventana en la que nos pregunta qué tipo de componentes vamos a usar, en ese caso elegimos componentes de tipo through-hole como se aprecia en la Figura 70.



**Figura 70.** Elección de tipos de componentes para el PCB.

Y de ahí seguimos dando next hasta que nos aparezca una placa completamente vacía como se puede ver en la Figura 71. De ahí vamos a grabar este diseño de pcb con un nombre que queramos y en la barra de menú vamos a Desing y damos en import chantes to PCB.

Y nos dará un mini reporte de todos los cambios que se van a importar de ahí damos execute changes y vamos a ver que se nos van a cargar todos los footprints correspondientes a los componentes de nuestro esquema para que los posicionemos en la placa como se ve en la Figura 72. Para posicionar los componentes solo debemos arrastrarlos al espació de la placa y recordando que dando click en la barra espaciadora mientras arrastramos el componente podemos girarlo, repetimos este proceso hasta tener todos los componentes posicionados.

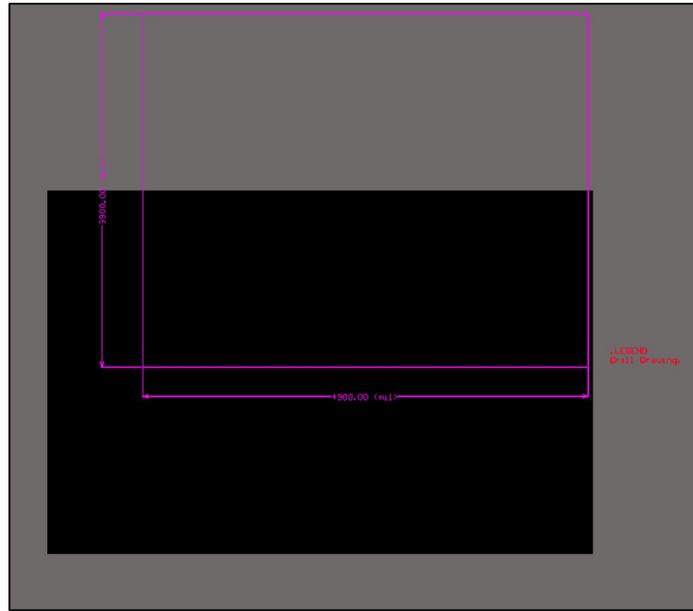


Figura 71. Placa PCB sin componentes.

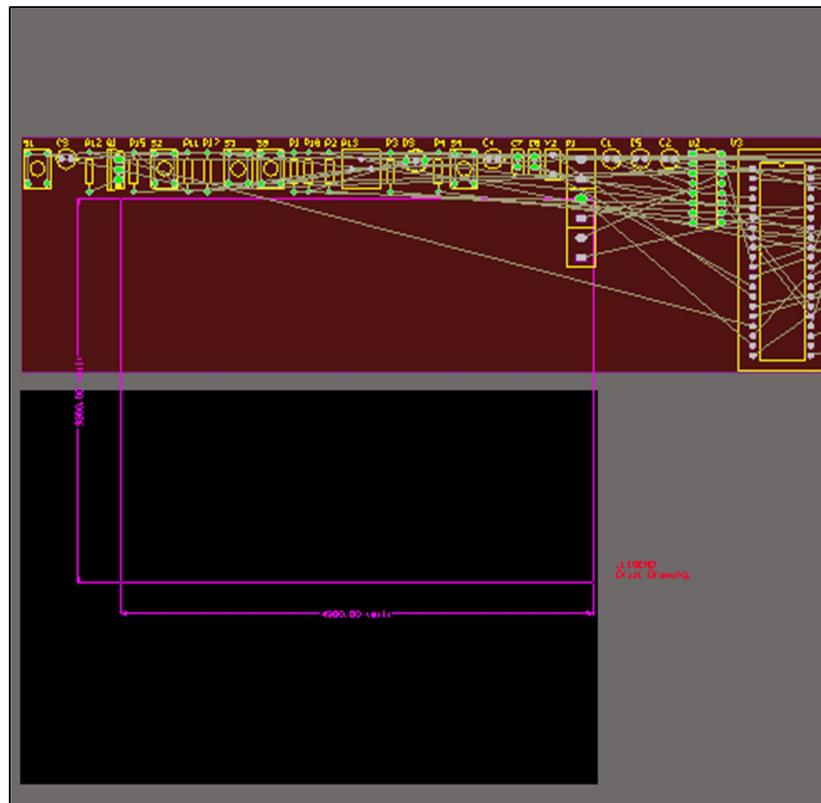


Figura 72. Placa PCB con los componentes antes de posicionarlos.

Como consideración para la ubicación de los componentes debemos poner los componentes que van cerca, reunidos entre sí evitar que estén muy separados los

componentes pero siempre teniendo en cuenta que cuando los soldemos si están demasiado juntos eso va a ser un impedimento.

Después de posicionar todos los componentes si nos sobro espacio en la placa podemos redefinir el tamaño de la placa en la barra de menús, desing/board shape y ahí tenemos las opciones de que si queremos redefinir la forma, cambiar las esquinas, borrar las primitivas o especificar los cortes en la placa.

Finalmente vamos a “rutear” la placa que es conectar con líneas los componentes que van unidos entre sí según el circuito esto lo hacemos automáticamente con la barra de menús en autorute y elegimos all en donde nos aparece una ventana como la de la Figura 73.

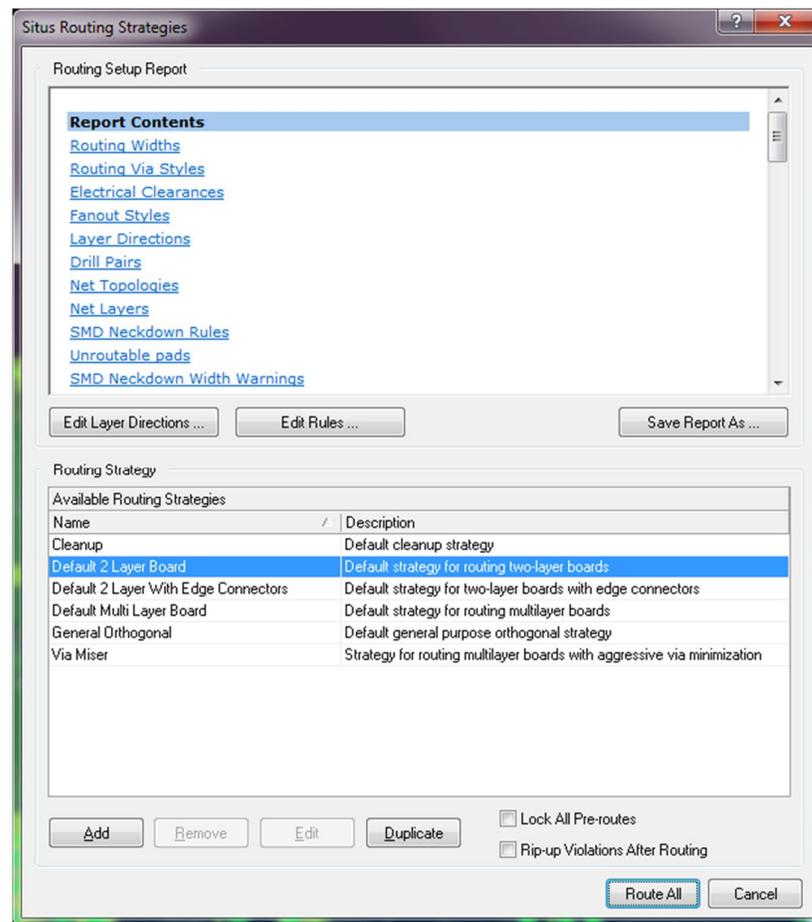


Figura 73. Ventana para el Auto Route.

Y damos click en el la primera opción de Routing Widths en la que elegimos el ancho de la pista para la placa en esta caso como es una placa de poca potencia el ancho de la pista puede variar de entre 0.3 a 1mm de ancho y de ahí damos click en route all en donde ya nos va a rutear la placa y nos va a mostrar errores si existen, y con

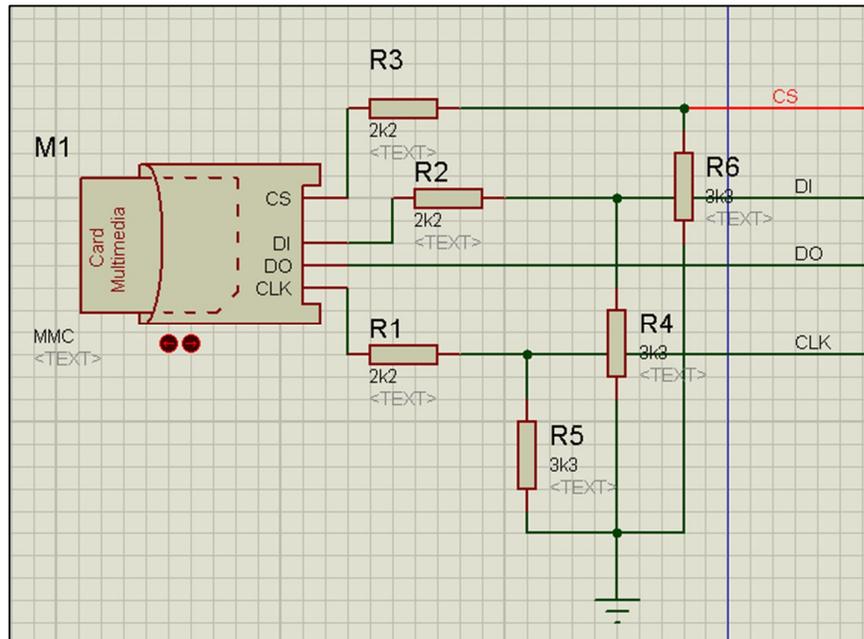
este diseño de la placa podemos ir a la fábrica para que la creen y nos la entreguen.

Refiérase al **anexo 2** para ver la placa del PCB completa como fue entregada a la fábrica.

### 3.9 Consideraciones especiales para el hardware de la placa

Aquí vamos a hablar de algunas consideraciones que tenemos que tener en cuenta en el hardware de la placa como son las siguientes:

Para que funcione correctamente la memoria SD debemos realizar un partidor de tensión de la salida del microcontrolador hacia la tarjeta SD ya que esta usa un voltaje de 3.3 voltios, como se puede ver en la Figura 74.



**Figura 74.** Diagrama de conexión de la tarjeta SD.

Como se puede ver en la Figura 74, se realiza un partidor con resistencias de 2.2K y 3.3K para bajar el voltaje de High TTL es decir de 5V a un nivel de TTL de 3V que aún se considera un 1 lógico, los cálculos del partidor están a continuación:

Si tenemos un Voltaje de ingreso de 5 voltios, necesitamos una salida de 3Voltios.

$$V_s = V_i \times \frac{R_1}{R_1 + R_2}$$

En donde R1 es la resistencia de 3.3K y R2 es la de 2.2k de donde obtenemos:

$$V_s = 5V \times \frac{3.3K}{2.2K + 3.3K}$$

$$V_s = 5V * 0.6$$

$$V_s = 3V$$

Y como podemos ver a la salida del partidor cuando enviamos un 1 lógico desde el microcontrolador que es de 5 V, se va a bajar el voltaje a 3V para que no dañe la memoria SD.

Otra consideración especial para el hardware es que tenemos que poner los cristales osciladores cerca del microcontrolador para el cristal de 16 000 MHZ y lo mismo para el cristal que usamos en el reloj DS1307.

## CAPÍTULO 4

### PRUEBAS DE LABORATORIO Y CAMPO. Y FUNCIONALIDAD.

#### 4.1 Introducción

En este capítulo se van a realizar las pruebas de laboratorio y campo en las que se van a contrastar los valores entre el equipo desarrollado y el equipo de referencia (YSI 6500), como se va a discutir la funcionalidad del equipo.

#### 4.2. Pruebas de laboratorio y campo

Para realizar estas comparaciones o mejor dicho comprobar que los valores que se indican en el display del equipo son exactos a los del equipo de referencia vamos tomar las medidas en diferentes entornos como lo son lagunas, ríos, el laboratorio, a continuación se encuentran las imágenes de las medidas tomadas en diferentes entornos para demostrar que son medidas correctas se hicieron 2 mediciones en cada entorno una con el equipo de referencia y la otra con el equipo elaborado como se aprecia en las Figuras 75,76,77,78.

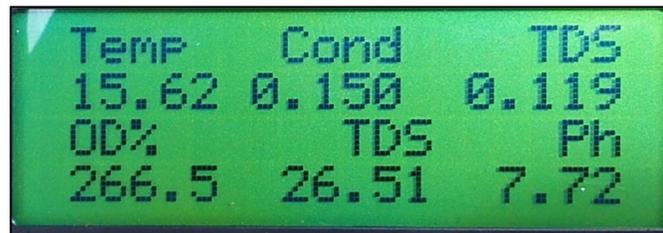


Figura 75. Foto de Medición 1 con equipo realizado.



Figura 76. Foto de Medición 1 con equipo de referencia.



Figura 77. Foto de medición 2 con equipo realizado.



Figura 78. Foto de medición 2 con equipo de referencia.

Como se ha podido ver en las imágenes anteriores, se puede apreciar que las medidas son exactamente las mismas entre los equipos sin importar las condiciones climáticas o el entorno en las que se tomaron, con lo que podemos concluir que nuestro equipo está homologado con uno de referencia reconocido.

#### 4.3. Funcionalidad

Después de realizar pruebas se puede decir que este equipo es funcional por los siguientes parámetros:

Es de tamaño moderado lo que lo convierte en portátil ya que se puede llevar tranquilamente en una mochila o en las manos mientras se camina, es totalmente portátil ya que tiene una batería interna de 12 V que nos permite un funcionamiento continuo por periodos prolongados de tiempo con solo una carga de la batería.

El display matricial de 20 x4 nos da un contraste correcto de los caracteres y no presenta problemas contra los cambios de temperatura bruscos, el menú del equipo es de fácil manejo ya que es intuitivo y el desplazamiento por el mismo se realiza mediante el empleo de 4 pulsantes de tamaño medio.

Tiene un indicador led que dos colores para indicar si la batería esta baja o no cambiando de un tono verde a un tono rojo cuando requiere carga.

La extracción de datos que muestra el equipo es sencilla ya que solo se requiere extraer la memoria de tipo microSD y conectarla a cualquier ordenador, teléfono que tenga un editor de texto que abra archivos de tipo .TXT lo que facilita la lectura de datos como se puede apreciar en la figura....

Los datos se guardan en un archivo que tiene de nombre el día, mes, año todo unido con extensión .txt como se mencionó antes, esto es gracias a que el equipo tiene un reloj de tiempo real que no se desajusta aunque la batería principal se descargue ya que tiene una pila de 3V interna que permite el funcionamiento continuo del reloj solamente, dicho reloj no se desajusta y puede ser modificada su fecha y hora por si en algún momento es necesario el remplazar la pila.

Aparte de las características antes mencionadas, el equipo está protegido contra la humedad del ambiente es decir que la humedad no se va a filtrar dentro del equipo pudiendo ocasionar daños a los componentes internos.

## CONCLUSIONES

Mediante la investigación de la sonda y elementos para adquirir parámetros de lagunas y ríos se construyó un equipo totalmente funcional para la medición de pH, conductividad y oxígeno disuelto. El equipo fue probado y homologado con un equipo YSI 6500.

El equipo es totalmente portable permitiendo así su fácil transportación para las zonas de muestreo. En medio de las mediciones el equipo después de recibir los datos de la sonda YSI y visualizarlos se permite la opción de almacenado de los mismos en una memoria de tipo MicroSD.

Las características del equipo son:

- El equipo responde correctamente a las exigencias presentadas en el laboratorio y en el campo de estudio de análisis.
- La conexión y comunicación con las sonda responden eficazmente.
- La interacción del usuario con el menú del equipo es fácil e intuitiva.
- El equipo es autónomo energéticamente ya que no requiere una conexión a red eléctrica.

Los alcances del equipo son:

- Calibración de los diferentes parámetros de la sonda.
- El equipo puede ser modificado para adicionar más parámetros de los mencionados anteriormente.
- El equipo guarda los datos mostrados en una memoria de tipo microSD para que luego sean leídos en un ordenador.

Se recomienda:

- Dar un mantenimiento al equipo analizador cada 4 meses, revisas los componentes electrónicos.
- Revisar el estado de la batería antes de cualquier uso.
- Leer completamente el manual de usuario adjunto, antes de usar el equipo.

## BIBLIOGRAFÍA

### Referencias bibliográficas:

1. PÉREZ RODRÍGUEZ, Leonel. (2008). *Introducción a los micro controladores*. Ecuador: Escuela de Ingeniería Electrónica. Universidad del Azuay. Digital.
2. *PIC18F2455/2550/4455/4550 Data Sheet*. Microchip Technology Inc, (2009). Digital.
3. *PIC18F2458/2553/4458/4553 Data Sheet*. Microchip Technology Inc, (2009). Digital.
4. *PIC18F2525/2560/4525/4620 Data Sheet*. Microchip Technology Inc, (2008). Digital.
5. *DS1307 Data sheet*. Dallas Semiconductor, (2008). Digital
6. *MOC 3020/MOC3021/MOC3022/MOC3023. Data Sheet*. Opto electronics. Digital.
7. YSI. *600 Multi-parameter Water Quality Monitor Instruction Manual*. (2004)
8. YSI. *6500 Enviromental Monitoring System Instruction Manual*. (2005)
9. RASHID, Muhammad H. (2004). *Electrónica de Potencia*. Tercera edición. México: PEARSON Educación de México. Impreso.
10. *OPTOISOLATORS TRIAC DRIVER OUTPUT*. Fairchild Semiconductor Corporation, (2002). Digital.

### Referencias electrónicas:

1. MikroElektronika; Ejemplos de programación de Microcontroladores; [En línea] Disponible en:  
<http://www.mikroe.com/chapters/view/82/capitulo-4-ejemplos/>  
 [Accesado el día 9 de Septiembre de 2012]
2. CLAVIJO MENDOZA, Juan Ricardo. Diseño y Simulación de Sistemas microcontrolados Lenguaje C. Primera Edición, 2011 [En línea] Disponible en:  
[http://www.mikroe.com/download/eng/documents/publications/other-books/libro\\_simulacion\\_mikroc.pdf](http://www.mikroe.com/download/eng/documents/publications/other-books/libro_simulacion_mikroc.pdf) [Accesado el día 10 de Octubre de 2012]

3. Microchip; Página Oficial de los microcontroladores PIC [En línea]  
<http://www.microchip.com/> [Accesado el día 15 de Agosto de 2012]
4. MikroElektronica; Página Oficial del programa compilador en Lenguaje C [En línea] <http://www.mikroe.com> [Accesado el día 15 de Agosto de 2012]

## ANEXOS

### ANEXO 1

# “Manual de Usuario del medidor multiparámetros para ríos y lagunas”



## 1.- Manual del Usuario

- Favor de leer cuidadosamente este Manual de Usuario antes de operar el equipo.
- Guarde este manual en un sitio seguro para consultarlo en el futuro.

## 2.- DESCRIPCIÓN DEL EQUIPO

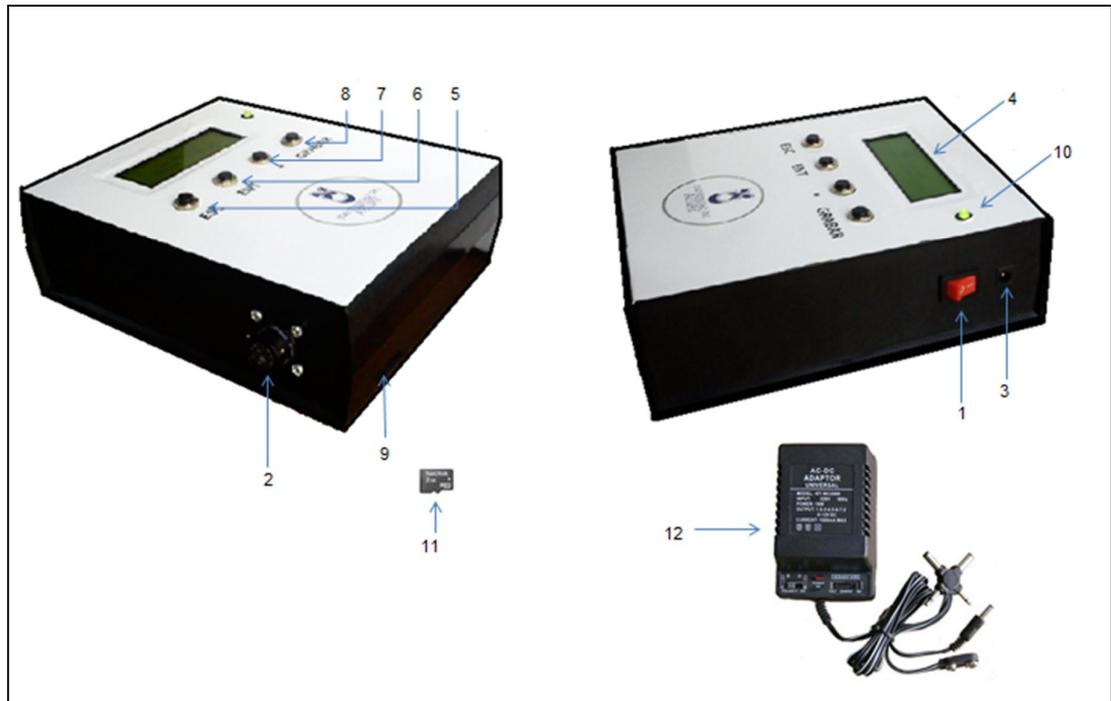


Fig. 2.1 Partes que conforman el analizador multiparámetros.

1. Interruptor de encendido del equipo.
2. Conector MS-8 para la comunicación con la sonda.
3. Conector de batería.
4. Pantalla LCD.
5. Botón "ESC".
6. Botón "ENT".
7. Botón "\*".
8. Botón "Grabar".
9. Ranura para la tarjeta SD.
10. Led indicador de encendido y estado de la batería.
11. Tarjeta micro SD.
12. Cargador de batería.

### 3.- Especificaciones

---

Modelo	V 1.1
Voltaje nominal	12V
Voltaje mínimo de funcionamiento	9V
Voltaje de conexión del cargador	110V
Temperatura de funcionamiento	-5°C a 65°C

### 4.- Características

---

- El equipo cuenta con una pantalla LCD matricial de 4x20, que proporciona la información al usuario.
- La batería de la cual está compuesta brinda gran maniobrabilidad al equipo, que además cuenta con su cargador.
- Graba archivos de texto en su tarjeta SD proporcionada. Estos archivos pueden ser leídos con toda facilidad en cualquier computador, gracias a su adaptador.
- Utiliza simplemente cuatro botones para la interacción con el usuario, además de su interruptor de encendido.
- El conector utilizado para conectar la sonda al equipo, es de uso militar, lo que brinda gran resistencia en el mismo.

## 5.- Descripción de los botones

---



El botón “\*”, es un botón de selección, este nos permite movilizarnos en los menús y submenús, además es utilizado en la configuración de valores, con lo cual permite movilizarse de un dígito a otro, como lo es de centenas a decenas por ejemplo.



El botón “Ent”, abreviación de Enter. Este nos permite ingresar a cualquier menú o submenú previamente seleccionado con el símbolo “\*”, además este botón es utilizado en la configuración de valores, este botón nos permite cambiar el valor de un dígito, es decir cambia el valor de 0 a 1 o de 1 a 2 y así sucesivamente. Para la configuración de valores, cada dígito solo subirá su valor.



El botón “ESC”, Abreviación de Escape. Este como su nombre lo indica, sirve para salir de un menú o submenú, al igual que la configuración de algún valor.



El botón “Grabar”, como su nombre lo indica guarda los valores dentro de la memoria SD, este botón funciona únicamente en el Submenú RUN. Dependiendo de la selección puede grabar los valores de forma continua o dato por dato.

## 6.- Encendido del equipo

---

Para encender el equipo se debe accionar el interruptor ubicado en la parte lateral derecha del equipo, como se indica en la figura 6.a, al realizar esta acción la pantalla LCD se encenderá, al igual que el led indicador de encendido.



Fig. 6.a Accionamiento del interruptor de encendido.

### 6.1- Led indicador

Este led indica cuando se haya encendido el equipo, además es un led bicolor que nos indica el estado de la batería que se esta utilizando, cuando la batería tenga un voltaje mayor a 9 Voltios, el led indicará un color verde, mientras que cuando se encuentre por debajo de este valor, el led indicará un color rojo, lo que significa que la batería necesita ser cargada. La comparación de voltajes se realizará cada que se encienda el equipo.

### 6.2- Encendido de la pantalla LCD

Cuando se encienda el equipo se encenderá la pantalla LCD, esta en un inicio nos mostrará un mensaje que indica "Universidad del Azuay", como se indica en la figura 6.2.1. Este mensaje desaparecerá después de un corto tiempo, después de transcurrido este tiempo, el LCD mostrará un mensaje que indicará que la sonda no está conectada, como se muestra en la figura 6.2.2.

Si la sonda se encuentra conectada, este mensaje desaparecerá inmediatamente y en su lugar indicará el mensaje "Conectando con YSI 600R" y si la sonda no está conectada, deberá serlo con lo que aparecerá el mismo mensaje.



Fig. 6.2.1 Impresión del mensaje "Universidad del Azuay", en el LCD.



Fig. 6.2.2 Impresión del mensaje "No existe sonda conectada", en el LCD.

## 7.- Menú principal

---

El menú principal aparecerá unos segundos después de que la sonda se haya detectado y cuando desaparezca el mensaje "Conectando sonda YSI 600R". El menú principal va a constar de 5 submenús, los cuales se ilustran en la imagen 7.1, para deslizarse en este menú, se debe pulsar el botón "\*" que es el botón de selección. Una vez seleccionado el submenú al cual se desea ingresar se presiona el botón "ENT".

Cada submenú tiene su propia función por lo cual es necesario la lectura de este manual con el fin de conocer todas las características que cuenta el equipo.



Fig. 7.1 Impresión del menú principal en el LCD.

## 8.- Submenú Reporte

El submenú reporte, es un aplicación que nos permite elegir los parámetros a ser calculados por la sonda. En la figura 8.a se ilustran todos los parámetros que la sonda tiene la posibilidad de medir.

( )TempC	Temperatura en grados centígrados.
( )Cond mS/cm	Conductividad del líquido en mili Siemens por centimetro
( )TDS g/L	Total de sólidos disueltos medido en gramos por litro.
( )DOsat %	Oxígeno disuelto, medido en porcentaje de saturación.
( )DO mg/L	Oxígeno disuelto, medido en miligramos por litro.
( )pH	Nivel de pH del líquido a ser medido.

Fig. 8.a Parámetros que la sonda puede leer.

### 8.1- Selección de parámetros a ser leídos

Cuando se ingresa en este submenú aparecerá la pantalla que se muestra en la figura 8.1.1, la cual indica que los datos se están cargando.

Una vez que los datos sean cargados se mostrará en la pantalla todos los parámetros como se indica en la figura 8.1.2, cada uno de los parámetros va a estas acompañado entre paréntesis por un "#". Lo que indica este numeral es que el parámetro está activo. Para desactivar o en su defecto activar uno u otro parámetro lo que debemos hacer es deslizarnos entre cada uno de los mismos y colocarnos en el que queramos realizar la acción, pulsamos el botón "ENT" y volverá a mostrar la pantalla de carga de datos de la figura 8.1.1. Una vez que desaparezca esta pantalla, se identificará que el "#", del parámetro en el que se realizó la acción, desaparecerá o aparecerá dependiendo de la acción realizada.

Para salir de este submenú antes o después de realizar cualquier acción, se deberá pulsar el botón “ESC”, con lo que regresaremos al menú principal.



Fig. 8.1.1 Impresión en la pantalla “Cargando datos”.



Fig 8.2.1 Submenú Reporte.

## 9.- Submenú RUN

Este es el submenú principal del equipo, ya que nos indica los valores de cada parámetro.

En el menú principal, posicionamos el “\*”, con el botón de selección en el submenú RUN, y pulsamos el Botón “ENT”; ingresaremos al submenú RUN, con lo cual aparecerá la ventana “cargando datos” de la figura 8.1.1; una vez que estos datos sean cargados, se imprimirá en la pantalla cada parámetro y bajo el mismo se imprimirá el valor actual de ese parámetro, como se indica en la figura 9.1.1.

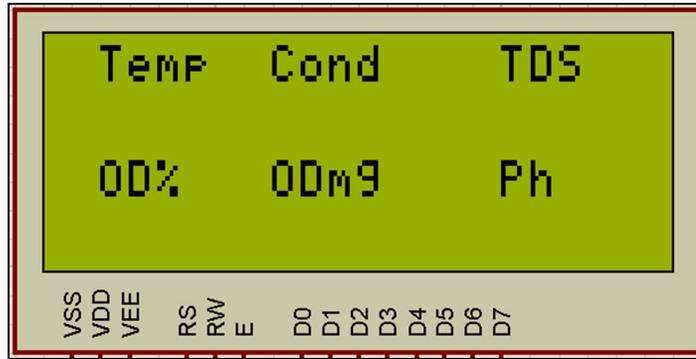


Fig. 9.1.1 Submenú RUN.

Los valores de cada parámetro van a ser actualizados cada 0.6 segundos.

Los datos que se mostrarán en esta ventana están ligados directamente con el submenú reporte; ya que los parámetros seleccionados en el submenú Reporte, van a ser indicados en la pantalla de RUN.

#### 9.1.- Grabar datos en la SD

Dentro del submenú RUN, tenemos la posibilidad de guardar los datos, estos datos se grabarán en la memoria SD conectada al equipo.

Para realizar esta acción, dentro del submenú RUN y mientras los datos se estén actualizando, se debe pulsar el botón "Grabar" como se indica en la figura 9.1.1.



Fig. 9.1.1 Botón "Grabar".

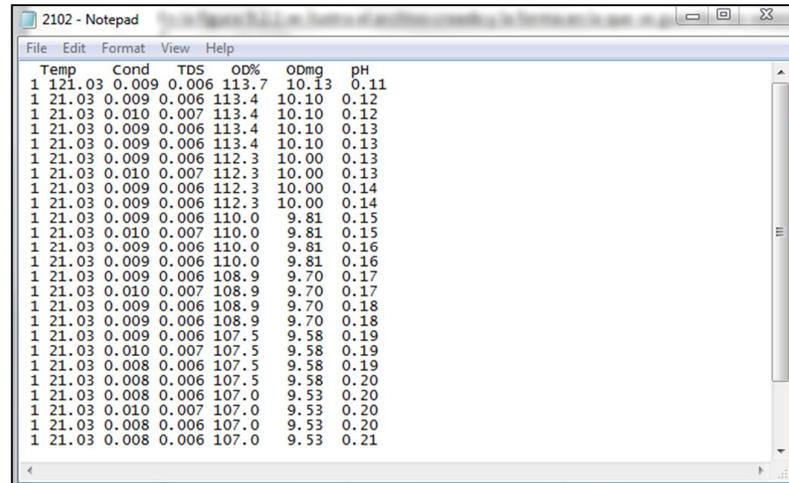
Existen dos diferentes formas para grabar datos en la SD, estas se especifican en el numeral 11.2.- tipos de grabado.

#### 9.2.- Archivo que contiene datos del submenú RUN

El archivo que se crea es un archivo de texto ".txt", que puede ser abierto en cualquier computador con editores de texto como son: Word, Wordpad, Textpad, etc.

El nombre de este archivo está ligado exclusivamente a la fecha y hora exacta en la que se realizó la medición, para así tener una referencia de cual medida corresponde a qué lugar.

En la figura 9.2.1 se ilustra el archivo creado y la forma en la que se guardan los valores en el archivo de texto.



Temp	Cond	TDS	OD%	ODmg	pH
1 21.03	0.009	0.006	113.7	10.13	0.11
1 21.03	0.009	0.006	113.4	10.10	0.12
1 21.03	0.010	0.007	113.4	10.10	0.12
1 21.03	0.009	0.006	113.4	10.10	0.13
1 21.03	0.009	0.006	113.4	10.10	0.13
1 21.03	0.009	0.006	112.3	10.00	0.13
1 21.03	0.010	0.007	112.3	10.00	0.13
1 21.03	0.009	0.006	112.3	10.00	0.14
1 21.03	0.009	0.006	112.3	10.00	0.14
1 21.03	0.009	0.006	110.0	9.81	0.15
1 21.03	0.010	0.007	110.0	9.81	0.15
1 21.03	0.009	0.006	110.0	9.81	0.16
1 21.03	0.009	0.006	110.0	9.81	0.16
1 21.03	0.009	0.006	108.9	9.70	0.17
1 21.03	0.010	0.007	108.9	9.70	0.17
1 21.03	0.009	0.006	108.9	9.70	0.18
1 21.03	0.009	0.006	108.9	9.70	0.18
1 21.03	0.009	0.006	107.5	9.58	0.19
1 21.03	0.010	0.007	107.5	9.58	0.19
1 21.03	0.008	0.006	107.5	9.58	0.19
1 21.03	0.008	0.006	107.5	9.58	0.20
1 21.03	0.008	0.006	107.0	9.53	0.20
1 21.03	0.010	0.007	107.0	9.53	0.20
1 21.03	0.008	0.006	107.0	9.53	0.20
1 21.03	0.008	0.006	107.0	9.53	0.21

Fig. 9.2.1 Archivo ".txt" creado que contiene datos de la sonda.

**NOTA:** Una vez que se desee abandonar este submenú, se deberá pulsar el botón "ESC", cuando esto suceda, y si se han guardado datos, la sonda se reseteará. Caso contrario, si no se han guardado datos y se pulsa el botón "ESC", Una vez que se presione el botón, el equipo se reseteará.

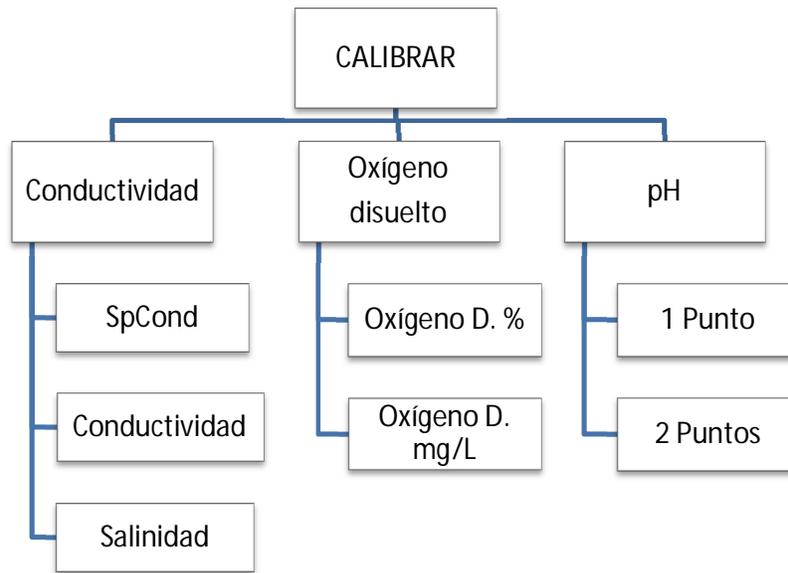
## 10.- Submenú Calibrar

---

Esta opción que posee el equipo, nos permite como su nombre lo indica, calibrar la sonda parámetro por parámetro.

Para ingresar en este submenú, posicionamos el "\*", con el botón de selección en el submenú Calibrar, y pulsamos el Botón "ENT"; con lo cual aparecerá la ventana "cargando datos" de la figura 8.1.1;

Este submenú, cuenta a su vez con diferentes submenús que nos permiten modificar sus valores, en la figura 10.a, se ilustra la estructura jerárquica que se utiliza para la calibración de cada uno de los parámetros.



**Fig. 10.a** Estructura jerárquica del Submenú Calibrar.

En él la figura 10.b se indica la pantalla del submenú reporte, la cual consta de los diferentes submenús.

### 10.1.- Conductividad

Una vez que se ingrese al submenú Conductividad, debemos escoger cada una de las opciones para configurar los parámetros.

#### SpCond

Para configurar SpCond, ingresamos en el mismo y la pantalla mostrará un mensaje como se indica en la figura 10.1.1. En esta pantalla, se muestra el valor por defecto que tiene este parámetro, el cual es 1.432 ms/cm, indicado en la parte superior derecha.

Para modificar el valor que se encuentra en 0, ingresaremos el valor de izquierda a derecha, el botón “\*”, será utilizado para moverse de dígito, siempre de derecha a izquierda. El botón “ENT”, será utilizado para incrementar el dígito en uno. No se puede disminuir sino solo aumentar. La figura 10.1.2 es una explicación gráfica de esta edición.



Fig. 10.1.1 Pantalla de calibración de SpCond.

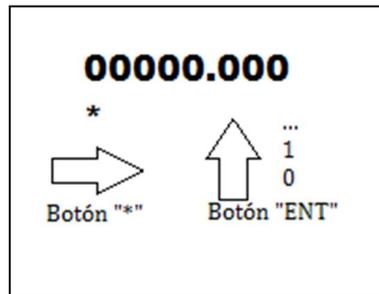


Fig. 10.1.2 Edición de valores.

Una vez que el valor se haya editado, se procede a pulsar el botón “ESC”, con lo que el valor será guardado. Una pantalla aparecerá indicando que el valor va a ser cambiado y que cuando se establezcan los datos, se presione el botón “ESC”. La figura 10.1.3 indica este mensaje.

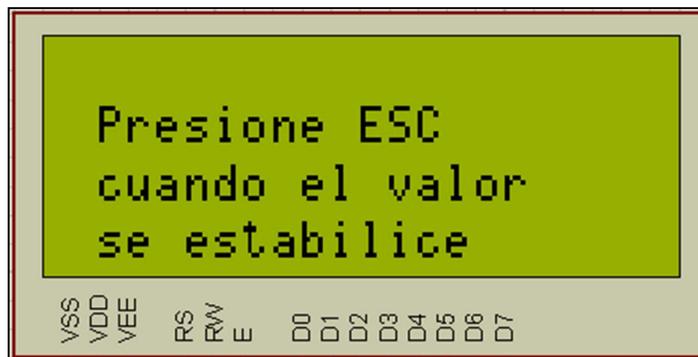


Fig. 10.1.3 Impresión en la pantalla de mensaje de “ESC”.

Aparecerá una pantalla muy parecida al menú RUN, pero en esta pantalla se mostrarán todos los valores independientemente, de los que se hayan escogido en el menú Reporte.

Una vez que el dato esté estable, se pulsa el botón "ESC" y el equipo se reseteará.

### Conductividad

Para configurar la Conductividad, realizamos el mismo procedimiento que el realizado en SpCond. La pantalla mostrará un mensaje como se indica en la figura 10.1.4. En esta pantalla, se muestra el valor por defecto que tiene este parámetro, el cual es 1.413 ms/cm, indicado en la parte superior derecha.

Para modificar el valor realizamos los mismos pasos, siguiendo la figura 10.1.2 que es una explicación gráfica de edición.



Fig. 10.1.4 Pantalla de calibración de Conductividad.

Una vez que el valor se haya editado, se procede a pulsar el botón "ESC", con lo que el valor será guardado.

Aparecerá una pantalla muy parecida al menú RUN, pero en esta pantalla se mostrarán todos los valores independientemente, de los que se hayan escogido en el menú Reporte.

Una vez que el dato esté estable, se pulsa el botón "ESC" y el equipo se reseteará.

### Salinidad

Para configurar la Salinidad, realizamos el mismo procedimiento que el realizado en SpCond. La pantalla mostrará un mensaje como se indica en la figura 10.1.4. En esta pantalla, se muestra el valor por defecto que tiene este parámetro, el cual es 1.413 PPM, indicado en la parte superior derecha.

Para modificar el valor realizamos los mismos pasos, siguiendo la figura 10.1.2 que es una explicación gráfica de edición.



Fig. 10.1.4 Pantalla de calibración de Salinidad.

Una vez que el valor se haya editado, se procede a pulsar el botón “ESC”, con lo que el valor será guardado.

Aparecerá una pantalla muy parecida al menú RUN, pero en esta pantalla se mostrarán todos los valores independientemente, de los que se hayan escogido en el menú Reporte.

Una vez que el dato esté estable, se pulsa el botón “ESC” y el equipo se reseteará.

## 10.2.- Oxígeno Disuelto

Una vez que se ingrese al submenú Oxígeno disuelto, debemos escoger cada una de las opciones para configurar los parámetros.

### Oxígeno Disuelto %

Para configurar el oxígeno disuelto en porcentaje, ingresamos en el mismo y la pantalla mostrará un mensaje como se indica en la figura 10.2.1. En esta pantalla, se muestra el valor por defecto que tiene este parámetro, el cual es la presión barométrica de 567.69 mmHg, indicado en la parte superior derecha.

Para modificar el valor que se encuentra en 0, ingresaremos el valor de izquierda a derecha, el botón “\*”, será utilizado para moverse de dígito, siempre de derecha a izquierda. El botón “ENT”, será utilizado para incrementar el dígito en uno. No se

puede disminuir sino solo aumentar. La figura 10.2.2 es una explicación gráfica de esta edición.



Fig. 10.2.1 Pantalla de calibración de OD %.

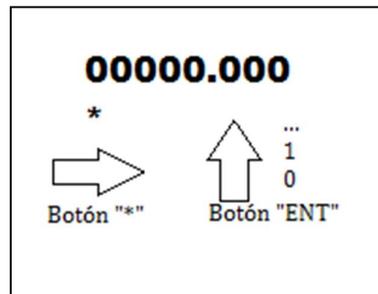


Fig. 10.2.2 Edición de valores.

Una vez que el valor se haya editado, se procede a pulsar el botón "ESC", con lo que el valor será guardado. Una pantalla aparecerá indicando que el valor va a ser cambiado y que cuando se estabilicen los datos, se presione el botón "ESC".

Aparecerá una pantalla muy parecida al menú RUN, pero en esta pantalla se mostrarán todos los valores independientemente, de los que se hayan escogido en el menú Reporte.

Una vez que el dato esté estable, se pulsa el botón "ESC" y el equipo se reseteará.

### Oxígeno Disuelto mg/L

El oxígeno disuelto expresado en mg/L. La pantalla mostrará un mensaje como se indica en la figura 10.2.3. En esta pantalla, se muestra el valor por defecto que tiene este parámetro, el cual es 20.64 mg/L, indicado en la parte superior derecha.

Para modificar el valor realizamos los mismos pasos, siguiendo la figura 10.2.2 que es una explicación gráfica de edición.



**Fig. 10.2.3** Pantalla de calibración de OD mg/L.

Una vez que el valor se haya editado, se procede a pulsar el botón “ESC”, con lo que el valor será guardado.

Aparecerá una pantalla muy parecida al menú RUN, pero en esta pantalla se mostrarán todos los valores independientemente, de los que se hayan escogido en el menú Reporte.

Una vez que el dato esté estable, se pulsa el botón “ESC” y el equipo se reseteará.

### 10.3.- pH

Una vez que se ingrese al submenú de pH, debemos escoger entre las dos opciones que brinda el equipo para calibrar este parámetro.

#### pH 1 punto

Para configurar el pH mediante solo un punto, ingresamos en el mismo y la pantalla mostrará un mensaje como se indica en la figura 10.3.1.

Para modificar el valor que se encuentra en 0, ingresaremos el valor de izquierda a derecha, el botón “\*”, será utilizado para moverse de dígito, siempre de derecha a izquierda. El botón “ENT”, será utilizado para incrementar el dígito en uno. No se puede disminuir sino solo aumentar. La figura 10.3.2 es una explicación gráfica de esta edición.



Fig. 10.3.1 Pantalla de calibración de pH de 1 punto.

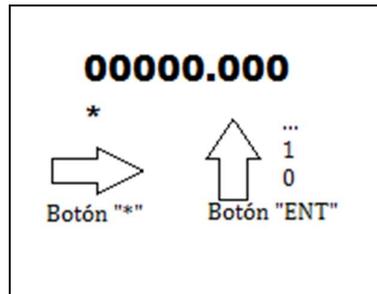


Fig. 10.3.2 Edición de valores.

Una vez que el valor se haya editado, se procede a pulsar el botón “ESC”, con lo que el valor será guardado. Una pantalla aparecerá indicando que el valor va a ser cambiado y que cuando se estabilicen los datos, se presione el botón “ESC”.

Aparecerá una pantalla muy parecida al menú RUN, pero en esta pantalla se mostrarán todos los valores independientemente, de los que se hayan escogido en el menú Reporte.

Una vez que el dato esté estable, se pulsa el botón “ESC” y el equipo se reseteará.

### pH 2 puntos

Para configurar el pH mediante dos puntos, ingresamos en el mismo y la pantalla mostrará un mensaje como se indica en la figura 10.3.3, indicando que se ingrese el primer punto.

Para modificar el valor que se encuentra en 0, ingresaremos el valor de izquierda a derecha, el botón “\*”, será utilizado para moverse de dígito, siempre de derecha a izquierda. El botón “ENT”, será utilizado para incrementar el dígito en uno. No se puede disminuir sino solo aumentar. La figura 10.3.2 es una explicación gráfica de esta edición.



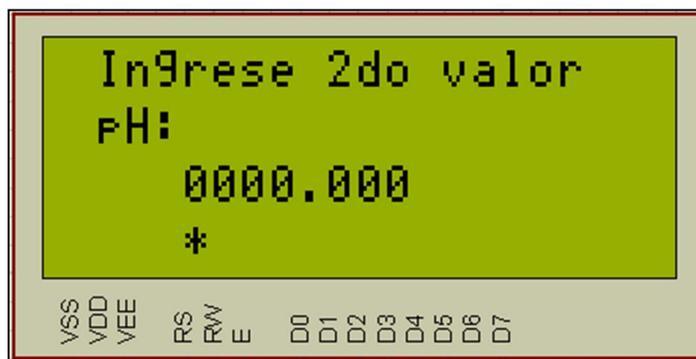
**Fig. 10.3.3** Pantalla de calibración de pH de 2 puntos (primer punto).

Una vez que el valor se haya editado, se procede a pulsar el botón "ESC", con lo que el valor será guardado. Una pantalla aparecerá indicando que el valor va a ser cambiado y que cuando se estabilicen los datos, se presione el botón "ESC".

Aparecerá una pantalla muy parecida al menú RUN, pero en esta pantalla se mostrarán todos los valores independientemente, de los que se hayan escogido en el menú Reporte.

Una vez que el dato esté estable, se pulsa el botón "ESC".

Una vez realizada esta acción, deberemos ingresar el segundo valor, en la imagen 10.3.4, se muestra el mensaje expresado en la pantalla, el cual debemos ingresar el valor de la misma manera que se lo realizó para el primer punto.



**Fig. 10.3.4** Pantalla de calibración de pH de 2 puntos (segundo punto).

Una vez que el valor se haya editado, se procede a pulsar el botón “ESC”, con lo que el valor será guardado. Una pantalla aparecerá indicando que el valor va a ser cambiado y que cuando se estabilicen los datos, se presione el botón “ESC”.

Aparecerá una pantalla muy parecida al menú RUN, pero en esta pantalla se mostrarán todos los valores independientemente, de los que se hayan escogido en el menú Reporte.

Una vez que el dato esté estable, se pulsa el botón “ESC” y el equipo se reseteará.

## 11.- Submenú SD

---

Este submenú está compuesto de dos opciones, las cuales son “Formatear Tarjeta” y “Modo de grabar”, como se muestra en la figura 11.a.



Fig. 11.a Pantalla del submenú SD.

### 11.1.- Formatear tarjeta

Esta opción permite al usuario borrar todos los datos y archivos que se encuentran dentro de la SD, por lo tanto es una opción la cual se debe tener cuidado al usarse, ya que se pueden perder datos importantes. Una vez ingresado en este submenú, aparecerá un mensaje de notificación como se demuestra en la figura 11.1.1, y este mensaje preguntará si se está seguro sobre borrar todos los datos.

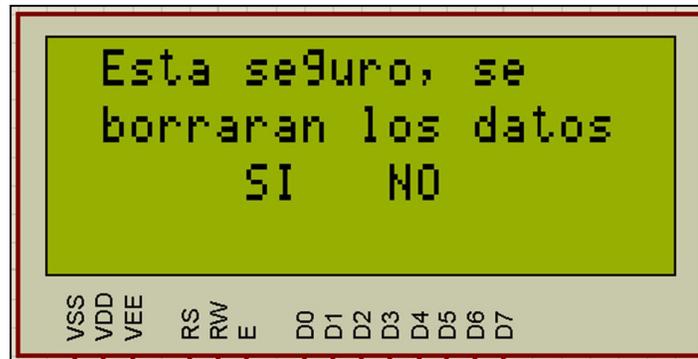


Fig. 11.1.1 Pregunta sobre si se está de acuerdo en borrar todos los datos.

Si se selecciona la opción “NO”, se regresará al submenú anterior, mientras que si se seleccionó la opción “SI”, Aparecerá un mensaje como se indica en la figura 11.1.2, indicando que se está formateando la tarjeta, esto demora alrededor de un minuto. Una vez realizada esta acción se regresará al menú anterior.



Fig. 11.1.2 Formateando la tarjeta SD.

## 11.2.- Modo de grabado

Permite grabar en dos opciones las cuales son datos continuos o dato por dato, como se indica en la figura 11.2.1. Cuando se seleccione “Datos Continuos”, los datos serán guardados uno tras otro pasando cortos periodos de tiempo, mientras que cuando se seleccione “Dato por Dato”, se guardara uno por uno.

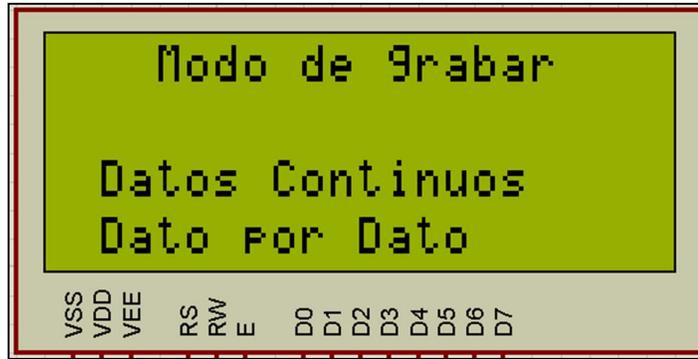


Fig. 11.2.1 Modo de grabado.

## 12.- Submenú Reloj

---

Este submenú está compuesto de dos opciones, las cuales son “Reloj” y “Ajustar Reloj”, como se muestra en la figura 12.a.



Fig. 12.a Pantalla del submenú Reloj.

### 12.1.- Reloj

Esta opción simplemente permite observar la hora actual a la que se encuentra el reloj de tiempo real, el usuario utilizará esta opción simplemente para comprobar que la hora del equipo es la misma que la que se tiene. En la figura 12.1.1 se indica el reloj en que se muestran parámetros como fecha hora y día de la semana.

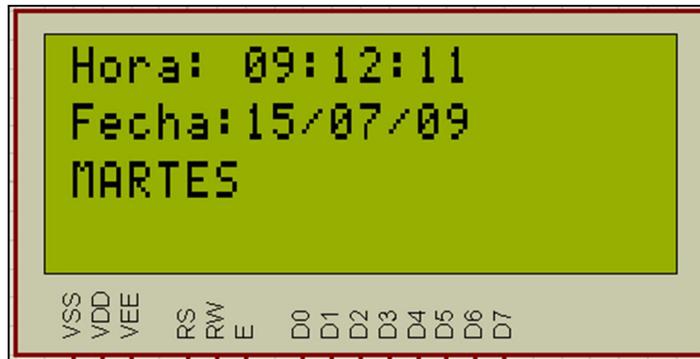


Fig. 12.1.1 Imagen del reloj de tiempo real.

## 12.2.- Ajustar Reloj

Esta opción permite al usuario modificar la fecha, hora y día del reloj, cuando este no este correcto. Una vez ingresado en este submenú aparecerá una pantalla como se muestra en la figura 12.2.1.



Fig. 12.2.1 Ajuste del reloj de tiempo real.

Para modificar el valor de hora y fecha que se encuentra en 0, ingresaremos el valor de izquierda a derecha, en la selección de hora se editará: “HORA:MINUTOS:SEGUNDOS”, mientras que en la selección de fecha se editará: “DIA/MES/AÑO”, luego se tendrá un número solo, el cual será el día de la semana siendo 0 domingo, 1 lunes y así sucesivamente.

El botón “\*”, será utilizado para moverse de dígito, siempre de derecha a izquierda. El botón “ENT”, será utilizado para incrementar el dígito en uno. No se puede disminuir sino solo aumentar. La figura 10.3.2 es una explicación gráfica de esta edición.

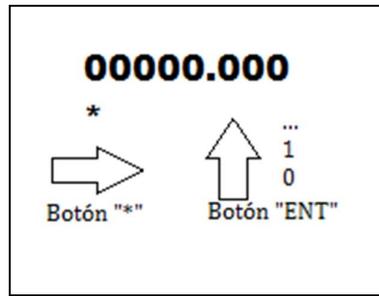
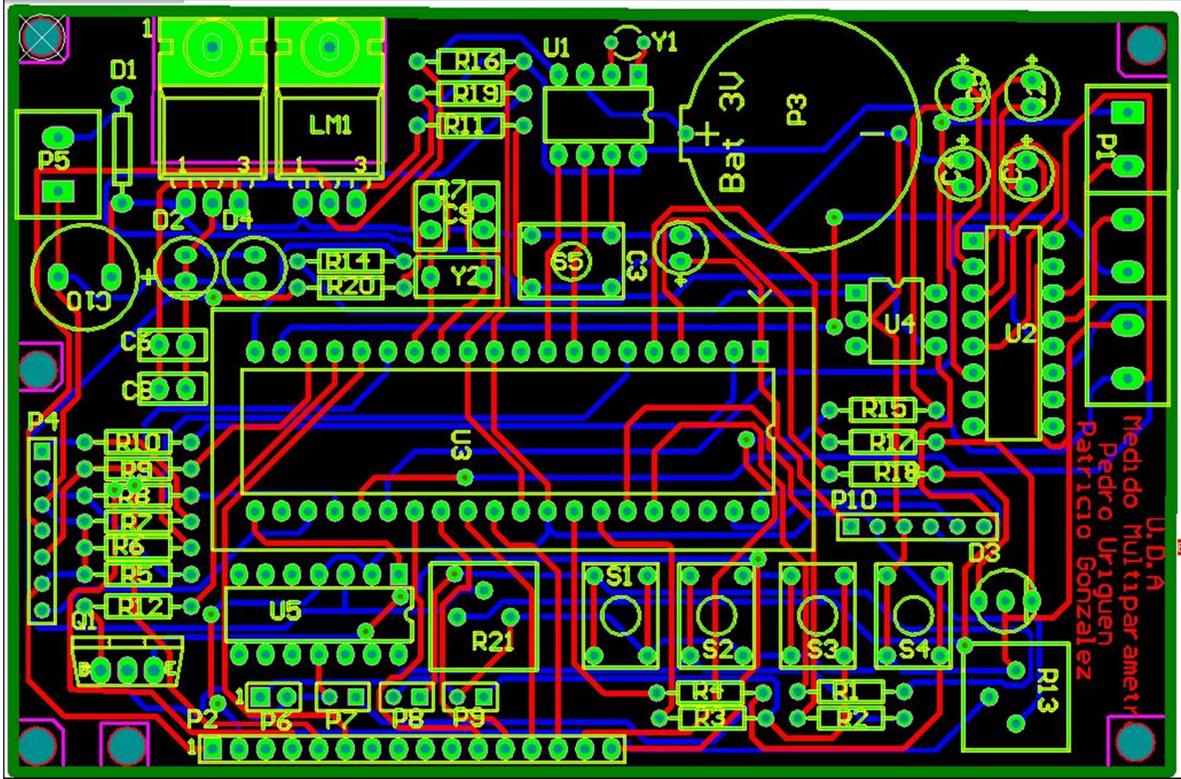


Fig. 12.2.2 Edición de valores.

**ANEXO 2**

Placa impresa.



**ANEXO 3**

Esquema del circuito de la placa

