



UNIVERSIDAD DEL AZUAY
FACULTAD DE CIENCIA Y TECNOLOGÍA
ESCUELA DE INGENIERÍA ELECTRÓNICA

**“Diseño y construcción de un prototipo para automatización
de muestreo y monitorización de la señal generada por un
espectrómetro de absorción atómica”**

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

Autores:

César Rolando González Brito
Gerónimo Andrés Molina Abril

Director:

Leonel Pérez Rodríguez

Cuenca, Ecuador

2012



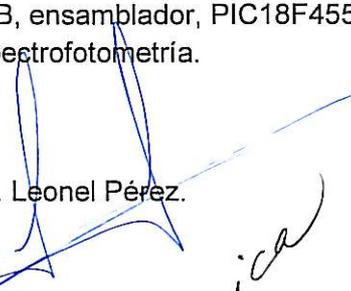
Diseño y construcción de un prototipo para automatización de muestreo y monitorización de la señal generada por un espectrómetro de absorción atómica

Resumen

Este Trabajo de Grado se orientó a la automatización de una máquina de muestreo utilizada para el análisis espectrofotométrico por absorción atómica utilizando un soporte mecánico en desuso y a la implementación un Sistema de Adquisición de Datos para la monitorización de la señal generada a partir de dicho análisis. La máquina de muestreo fue automatizada empleando un microcontrolador PIC18F4550, encargado de controlar los elementos electromecánicos apoyándose en un Interface de Potencia completamente basada en componentes de estado sólido. El Sistema de Adquisición de Datos de la señal generada por la máquina de espectrometría marca PERKIN-ELMER consta de dos bloques principales: uno para acondicionar la señal analógica y otro basado en un microcontrolador PIC18F4553 para digitalizarla y transmitirla vía USB al ordenador. La programación del micro controlador fue desarrollada en ensamblador. El sistema ha superado una etapa de pruebas y se encuentra actualmente en uso.

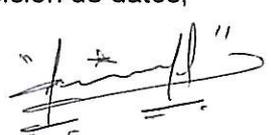
Palabras clave:

USB, ensamblador, PIC18F4553, automatización, muestreo, adquisición de datos, espectrofotometría.



Ing. Leonel Pérez.

César González B.



César González, Gerónimo Molina

Revisado
Junta Académica
09/05/2012.

Handwritten signature and date: 21/05/12

Design and construction of a prototype automated sampling and monitoring the signal generated by an atomic absorption spectrometer

Abstract

This Prom Work was oriented to the automation of a Sampling Machine that helps in samples manipulation during Atomic Absorption Photometric analysis and a Data Acquisition System for monitoring the signal generated by said analysis. The Sampling Machine automation was supported on rebuilding an abandoned mechanic structure and adding intelligent electronics based on a Microchip PIC18f4550 micro controller. The micro controller was programmed to control electro - mechanic components through a solid state technology Power Interface. The Data Acquisition System of the signal generated by the a PERKIN-ELMER spectrophotometer machine consists of two principal blocks: One dedicated to condition the analog signal and other one based in a PIC18F4553 microcontroller used digitalize and transmit the signal to the computer via USB. The firmware for the microcontroller was fully developed in assembler language. The system has overcome the stages of testing and is currently in use.

Keywords:

USB, assembler, PIC18F4553, automation, sampling, data acquisition, spectrophotometric.

Handwritten signature of Ing. Leonel Pérez.

Ing. Leonel Pérez.

*Revisado
Firma
09/05/2012
Académica*

Handwritten signature of César González B.

César González, Gerónimo Molina

ÍNDICE GENERAL

Resumen	ii
Abstract	iii
Índice de Contenidos	iv
Índice de Figuras	vi
Índice de Tablas	viii

INTRODUCCIÓN..... 1**CAPITULO 1. FUNDAMENTO TEÓRICO**

1.1. Conceptos Relacionados.....	3
1.1.1. Espectrofotometría	3
1.1.2. Sistemas A/D.....	5
1.1.3. Elementos de Potencia	8
1.1.4. Generalidades sobre USB	10
1.1.5. Micro controladores	21
1.1.6. Sistemas SCADA.....	28
1.2. Diagnóstico del Sistema Automático de Muestreo.....	29
1.2.1. Introducción	29
1.2.2. Ciclo funcional del equipo	32

CAPITULO 2. HARDWARE

2.1. Máquina de Muestreo.....	33
2.1.1. Introducción	33
2.1.2. Fuente de alimentación para Microcontrolador	34
2.1.3. Tarjeta del Microcontrolador	36
2.1.4. Opto acopladores	49
2.1.5. Fuente de alimentación para motor DC	40
2.1.6. Inversión de giro para motor DC	41
2.2. Sistema de Adquisición de Datos	42
2.2.1. Acoplamiento de la señal.....	42
2.2.2. Digitalización y transmisión de datos	43

CAPITULO 3. FIRMWARE

3.1. Automatización de la máquina de muestreo	45
3.1.1. Inicio	45
3.1.2. Interrupciones	46
3.2. Sistema de Adquisición de datos USB	56
3.2.1. Conversión de señal analógica	57
3.2.2. Comunicación USB.....	57

CAPITULO 4. SOFTWARE

4.1. Introducción.....	63
4.2. Controladores.....	63
4.3. Adquisición de datos y procesamiento de la señal	63
4.3.1. Inicialización	64
4.3.2. Panel Frontal	65
4.3.3. Diagrama de Bloques	66

RESULTADOS	71
-------------------------	-----------

CONCLUSIONES	72
---------------------------	-----------

BIBLIOGRAFÍA.....	74
--------------------------	-----------

ANEXOS	75
---------------------	-----------

ÍNDICE DE FIGURAS

Figura 1.1. Diagrama de Bloques: Adquisición de datos	5
Figura 1.2. Error en la cuantificación	7
Figura 1.3. Digitalización de una señal analógica.....	7
Figura 1.4. Relé de Estado Sólido	9
Figura 1.5. Puente H	10
Figura 1.6. Puertos USB tipo "A" y "B" machos y hembras	11
Figura 1.7. Identificación de dispositivos "Low", "Full" y "High".....	12
Figura 1.8. Conexión de un dispositivo USB a un Host	13
Figura 1.9. Topología USB	14
Figura 1.10. Ejemplo de una transacción USB	16
Figura 1.11. Campos componentes de un paquete	16
Figura 1.12. Mapa de descriptores	20
Figura 1.13. Esquema básico de un microcontrolador	21
Figura 1.14. Esquema del módulo USB y opciones.....	25
Figura 1.15. Lógica funcional de Interrupciones USB	28
Figura 1.16. 1.17 Funcionamiento Motor AC	30
Figura 1.18. Terminal de succión en estado inicial	31
Figura 2.1. Disposición de los componentes en la máquina de muestreo.....	34
Figura 2.2. Puente de Greatz	35
Figura 2.3. Esquema de la tarjeta de Alimentación.....	36
Figura 2.4. Esquema del uC PIC18F4550	37
Figura 2.5. Esquema Circuito MCLR	37
Figura 2.6. Esquema en la tarjeta del uC para LCD y Teclado	38
Figura 2.7. Teclado Matricial de 4x4.....	38
Figura 2.8. LCD matricial de 16x2	39
Figura 2.9. Circuito de control del LED del LCD	39
Figura 2.10. Circuito de control del motor AC	40
Figura 2.11. Esquema usado en la tarjeta "Fuente Variable".....	41
Figura 2.12. Esquema usado Puente H L298	41
Figura 2.13. Encapsulado L298 y Huella en PCB	42
Figura 2.14. Amplificador instrumental basado en el integrado AD620AN.....	43
Figura 2.15. Esquema del módulo de adquisición de datos	43

Figura 2.16. PCB del módulo de adquisición de datos	44
Figura 3.1. Diagrama de flujo principal del firmware "Sampler 1.0"	46
Figura 3.2. Teclado en la Interrupción por desborde de TMR0	47
Figura 3.3. Teclado matricial	48
Figura 3.4. Secuencia temporizada de los motores	49
Figura 3.5. Diagrama de flujo de secuencia de motor	51
Figura 3.6. LCD mientras se ejecuta la secuencia de motores	52
Figura 3.7. Diagrama del Menú o Panel de Configuración.....	53
Figura 3.8. LCD primera pantalla del MENU.....	53
Figura 3.9. LCD segunda pantalla del MENU	54
Figura 3.10. LCD tercera pantalla del MENU	54
Figura 3.11. Submenú "Contador"	55
Figura 3.12. "Muestras a Tomar"	55
Figura 3.13. El contador es reiniciado	56
Figura 3.14. Selección del periodo	56
Figura 3.15. Diagrama de bloques general del Sistema de Adquisición	57
Figura 3.16. Diagrama de flujo, Envío de descriptores de dispositivo	59
Figura 3.17. Diagrama de flujo, Establecimiento de la dirección del dispositivo.....	61
Figura 4.1. Diagrama de flujo del software desarrollado en Labview	64
Figura 4.2. Panel frontal	65
Figura 4.3. Obtención de datos a través de VISA. Diagrama de bloques	66
Figura 4.4. Obtención de datos a través de VISA en una iteración.....	67
Figura 4.5. Acondicionamiento de la señal	67
Figura 4.6. Ajuste de rango	68
Figura 4.7. Filtrado numérico de la señal	68
Figura 4.8. Aplicación de logaritmo natural a la señal.....	69
Figura 4.9. Indexado y archivado de datos	69
Figura 4.10. Diagrama de bloques completo	70
Figura 5.1. Señal obtenida con un DAQ USB-6009 sin amplificar ni filtrar	71
Figura 5.2. Señal obtenida por el sistema de adquisición de datos	71

ÍNDICE DE TABLAS

Tabla 1.1. Diagrama de Bloques: Adquisición.....	27
Tabla 3.1. Error en la cuantificación.....	60

ÍNDICE DE ANEXOS

Anexo 1. Sampler 1.0 (Guía de usuario).....	75
Anexo 2. Dispositivo USB (Guía de usuario).....	80

González Brito César González

Molina Abril Gerónimo Andrés

Trabajo de Graduación

Ing. Leonel Pérez Rodríguez

Mayo de 2012

DISEÑO Y CONSTRUCCIÓN DE UN PROTOTIPO PARA AUTOMATIZACIÓN DE MUESTREO Y MONITORIZACIÓN DE LA SEÑAL GENERADA POR UN ESPECTRÓMETRO DE ABSORCIÓN ATÓMICA

INTRODUCCIÓN

Dentro de los procesos que se realizan en el Laboratorio de Biología de facultad de Ciencia y Tecnología de la Universidad del Azuay, la espectrometría por absorción atómica requiere el análisis de una gran cantidad de muestras tomadas en tiempos periódicos. El proceso de muestreo se lo hace de forma automática para eximir al operador de realizar tareas repetitivas y evitar desfases de tiempo entre una muestra y otra, desfase que sería perjudicial a la hora de procesar los datos.

En este trabajo de graduación se desarrolla una máquina de muestreo automática digital, exclusiva para el análisis espectrofotométrico por absorción atómica que realiza una maquina marca "PerkinElmer" en dicho laboratorio, además la adquisición de los datos obtenidos a partir de esta máquina y transmisión vía USB hacia un ordenador en donde se los monitorea y procesa.

El proceso de muestreo se lo automatiza de forma digital, mediante un microcontrolador "PIC18F4550" de la casa fabricante "Microchip Technology Inc.". El firmware es escrito enteramente en lenguaje ensamblador, y trata básicamente de un menú de configuración y una secuencia de alimentación e inversión de giro de

motores, que corre al mismo tiempo que un programa cronómetro que separa el análisis de una muestra a otra en tiempos exactos.

La parte mecánica de la máquina de muestreo, es decir, motores, engranes y estructura, se han obtenido de otra máquina que ha dejado de funcionar, estos han sido calibrados y acoplados a la etapa digital mediante una etapa de potencia de estado sólido.

La máquina de espectrometría "PerkinElmer" obtiene y analiza muestras de forma continua. Es decir, una muestra es succionada para ser analizada durante un tiempo determinado, antes de que la siguiente muestra sea succionada, se debe limpiar el área de análisis y el tubo de succión durante un segundo tiempo, en el cual, la máquina de espectrometría no deja de hacer el análisis de componentes. Por ese motivo se obtiene como resultado una curva en el tiempo que debe ser procesada y analizada. Todos los datos de esta curva serán monitoreados, procesados, analizados y almacenados en un ordenador.

Es por eso que es necesario comunicar la máquina de espectrometría "PerkinElmer" con el ordenador de una manera eficaz y moderna, por lo que se ha construido un sistema de adquisición de datos vía USB. Este digitaliza la señal analógica que da como resultado la máquina de espectrometría y transmite esos datos al ordenador.

El ordenador debe ser capaz de recibir datos, mostrarlos y separar los datos de utilidad para simplificar la labor del operador realizando las tareas repetitivas que antes él realizaba.

CAPITULO 1

FUNDAMENTO TEÓRICO

1.1. Conceptos relacionados

1.1.1. Espectrofotometría

Proceso por el cual se determina específicamente el analito y su cantidad en una muestra, valiéndose de los fenómenos de emisión y absorción atómica. Dicha muestra es, en general, una solución, siendo el analito el soluto.

Todas las sustancias absorben energía radiante, cuya longitud de onda depende de los elementos químicos que compongan la sustancia, y suele estar en el rango del infrarrojo al ultravioleta, pasando por supuesto por la luz visible.

El proceso de medición por espectrofotometría consiste en determinar los elementos que de una muestra según la onda que sea absorbida. Es necesario de una fuente de luz, cierta cantidad de muestras, y un análisis matemático de los resultados obtenidos para así, obtener como conclusiones el soluto y su cantidad.

1.1.1.1. Principios de la Espectrofotometría de Absorción Atómica

Cuando a un átomo se le aplica cierta cantidad de energía, por ejemplo por calentamiento, excitación eléctrica o por la absorción de un fotón, esta energía es absorbida por electrones de capas externas que pasarán a una condición de inestabilidad, dejando al átomo en un "estado de excitación", en este estado el electrón salta a una capa inferior y cambia sus números cuánticos. Inmediatamente después, el átomo regresa a su "estado fundamental" liberando esta energía adicional, este proceso es conocido como "emisión atómica".

Las longitudes de onda (λ) de la energía emitida por el átomo es de carácter discreto y está dentro del rango de infrarrojo a ultravioleta, por lo que se tiene para un elemento, un espectro de ondas de diferentes λ separados por intervalos iguales llamados líneas

de emisión. Cada electrón irradia su energía para llegar a su estado fundamental de una longitud de onda característica y específica según su nivel o orbita. La intensidad de luz emitida depende del número de átomos excitados.

En el proceso de "absorción atómica", la muestra es expuesta a una fuente de luz, un átomo en estado fundamental, absorbe esta energía radiante mientras entra a su estado de excitación, la cantidad de fotones absorbidos por la muestra hace relación al número de átomos en la misma. Cada elemento cumple con este proceso con una longitud de onda específica.

El haz de luz que ha pasado por la muestra es medido por un detector o un elemento fotosensible, entonces se comparan los resultados con la medición del haz de luz que no ha pasado por muestra alguna y esta diferencia es proporcional a la absorbencia en la muestra.

Se define a absorbencia (A) como:

$$A = \log_{10} \frac{I}{I_0}$$

Donde:

I es la intensidad de luz que pasa por la muestra.

*I*₀ la intensidad de luz que no entra en la muestra.

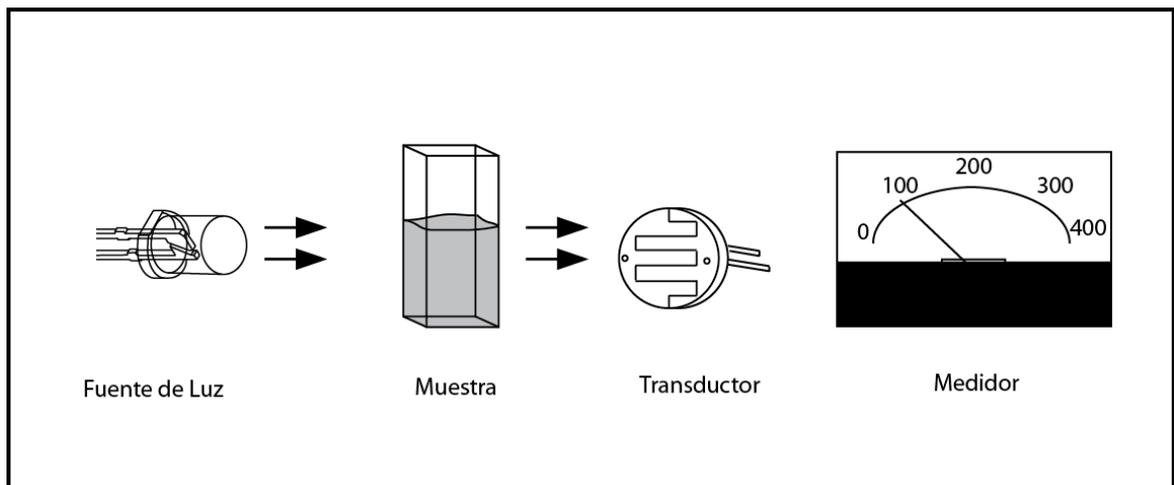
1.1.1.2. Espectrofotómetros de de Absorción Atómica

Un espectro fotógrafo generalmente está compuesto de 4 partes (figura 1.1):

- 1) Una fuente de energía radiante, emite el haz de luz que va a ser absorbida por la muestra. El haz de luz debe ser de una longitud de onda específica, por lo que se la suele crear a partir de gases que generan llamas de colores determinados, según el gas elegido. También, se puede filtrar un haz de luz blanca (de un espectro amplio) con un vidrio o plástico de color, que solamente permita el paso de el haz de longitud de onda deseada.
- 2) Un contenedor en donde se coloca la solución que va a ser analizada, alineado con el haz de luz y el elemento fotosensible.

- 3) Un elemento fotosensible o transductor que convierte el nivel de absorbencia en un nivel de voltaje o de corriente, generalmente se compone de un sensor lumínico adaptado a un circuito electrónico que acondiciona la señal.
- 4) Un sistema de lectura y procesamiento de datos que permite convertir las señales eléctricas provenientes del transductor en datos legibles para el operador. Existen máquinas de espectrofotetría autónomas que muestran los resultados en su pantalla, sin embargo suelen tener la posibilidad de transmitir la señal final a un ordenador por medio de un sistema de adquisición de datos.

Figura 1.1. Diagrama de Bloques: Adquisición de datos.



1.1.2. Sistemas A/D

1.1.2.1. Concepto y partes del sistema A/D

Un sistema de conversión analógico a digital transforma una señal de voltaje analógico a valores binarios para que, entre otras cosas, sean entendidos por una unidad central de procesos. Para ello, se toman muestras de la señal analógica a una frecuencia determinada y se asigna un valor binario correspondiente a un nivel de amplitud de la señal analógica.

Los dos procesos principales en la digitalización de una señal son entonces, el muestreo y la cuantificación, por lo que los convertidores A/D son compuestos de dos partes llamadas "Sample and Hold" (S/H) y "Cuantificador".

El muestreo realizado por el S/H, es la toma de un nivel de voltaje a una frecuencia determinada (figura 1.3b) y la mantención de dicho nivel durante el tiempo necesario (tiempo de muestreo, t_s) para la cuantificación (figura 1.3c); todos los datos correspondientes a la señal que se encuentre entre una muestra y otra se pierden, introduciendo un error de muestreo que se analiza más adelante.

Un nivel de voltaje analógico es cuantificado tomando en cuenta dos voltajes de referencia, generalmente llamados V_{ref+} y V_{ref-} , dando un valor binario correspondiente, desde cero si el nivel de voltaje es V_{ref-} y un valor máximo si el nivel de voltaje es igual a V_{ref+} . El valor binario máximo depende de la cantidad de bits con que se cuantifique el voltaje, este número de bits se denomina resolución. El nivel de voltaje a cuantificar se supone, debe estar dentro del rango V_{ref-} a V_{ref+} , de lo contrario toma valores extremos.

Por ejemplo, para un sistema A/D de 8 bits de resolución, los valores a la salida del cuantificador estarán comprendidos entre 0 y 11111111, lo que en base decimal corresponde a valores de 0 por V_{ref-} hasta 255 para V_{ref+} .

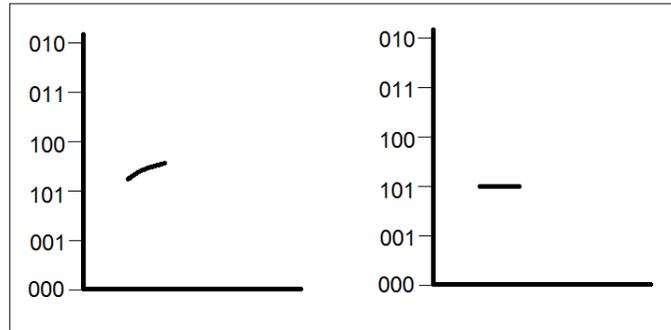
1.1.2.2. Procesos de muestreo y cuantificación óptimos

Por la naturaleza discreta de las señales digitales, una señal analógica al digitalizarse, perderá exactitud, por lo tanto, debe cumplir ciertas normas para garantizar su calidad.

Cada muestra requiere de un tiempo mínimo para ser cuantificada, perdiéndose así toda la información que esté entre una muestra y otra. Mientras mayor sea el tiempo que hay entre una muestra y otra, más información se pierde. Este error se reduce aumentando la frecuencia de muestreo. El teorema de muestreo de Nyquist-Shannon, dice que para poder replicar con mayor exactitud la forma de una onda es necesario que la frecuencia de muestreo sea superior al doble de la máxima frecuencia de la señal a muestrear.

De la misma forma en que se pierden los datos entre una muestra y otra, el nivel de voltaje que esté entre los correspondientes a dos valores binarios no se lo cuantifica de manera exacta.

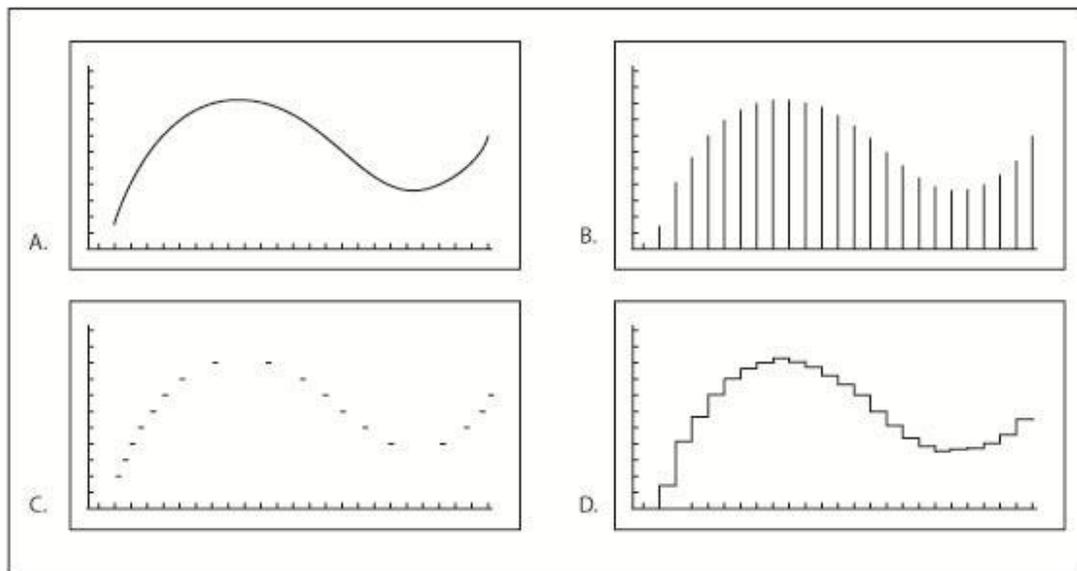
Figura 1.2. Error en la cuantificación.



Este error de cuantificación se reduce al aumentar la resolución, pero nunca se elimina. Se define "Bit menos significativo" (LSB por sus siglas en inglés) al valor de voltaje por encima de cuya mitad (LSB/2) se tiene un número superior a la salida del convertidor. El LSB se puede encontrar mediante la siguiente ecuación.

$$LSB = \frac{V_{ref+} - V_{ref-}}{2^{Resolución} - 1}$$

Figura 1.3. Digitalización de una señal analógica. 4A Señal analógica. 4B Intervalos de tiempo de muestreo. 4C Señal a la salida de S/H. 4C Señal digital.



1.1.3. Elementos de Potencia

Circuito o circuitos intermedios entre la etapa digital y los componentes inductivos o de alto voltaje que requieran corrientes mayores a los que se maneje en dicha etapa digital. Es decir, para controlar una etapa de alta potencia, formada por componentes como motores AC, motores DC, solenoides, etc. por medio de una etapa de baja potencia, se necesita una sistema de acople entre las dos etapas mencionadas; este sistema se llama "Interface de Potencia".

El uso común de interfaces de potencia se debe a que la automatización de procesos mediante micro controladores o cualquier sistema digital que haga la lógica de control, suministra corrientes bajas, por ejemplo un micro controlador suministra alrededor de 20mA de corriente, por lo que sería un error conectar una carga que requiera corrientes mayores para su funcionamiento, directamente a sus puertos.

Por motivos de seguridad para el operario y protección del sistema digital, es deseable que los circuitos de corrientes altas estén aislados eléctricamente de los circuitos de corrientes bajas.

Existen varias formas para acoplar una etapa de potencia a una etapa digital, evitando la conexión física entre ellas. El acoplamiento inductivo, por ejemplo, consiste en alimentar una bobina electroimán que al ser activado en la etapa digital, acciona magnéticamente, uno o varios contactos metálicos en la etapa de potencia, evitando así conexión conductiva entre las dos etapas; la bobina y los contactos vienen encapsulados en un "Relé Electromecánico".

1.1.3.1. Relé de Estado Sólido

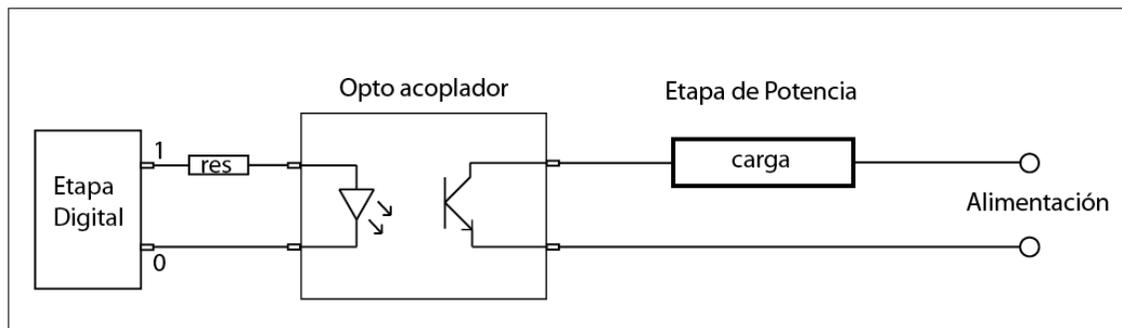
Un "Relé de Estado Sólido" o "Relé SSR" por sus siglas en inglés, es un encapsulado que usa transistores, tiristores o triacs en vez de contactos metálicos, y generalmente acoplamiento óptico.

El acoplamiento óptico, consiste en alimentar un LED (diodo emisor de luz) usualmente infrarrojo o de luz visible en la parte digital, y un dispositivo sensible a la luz en la etapa de potencia, que cuando es activado permite el paso de corriente disparando el elemento que cerrará el circuito de potencia.

Un Relé SSR es más liviano, silencioso, rápido, confiable y resistente al desgaste que un Relé Electromecánico, pero a cambio es más costoso. Sin embargo se puede simular un relé SSR mediante circuitos análogos con la desventaja de que no esté encapsulado.

Entonces, los elementos a considerar son los "Opto acopladores" que cuyo elemento sensible a la luz pueden ser un fototransistor, un fotodarlington, un fotoSCR, un fotoTRIAC, un fotoFET, etc.

Figura 1.4. Circuito básico de un Relé de Estado Sólido.

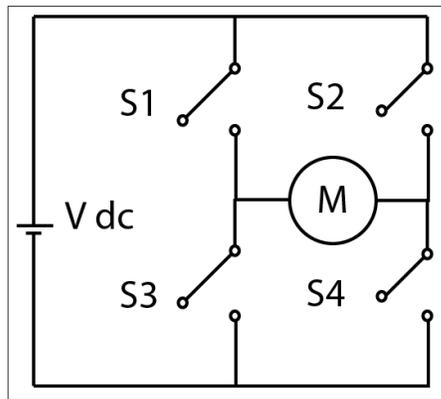


1.1.3.2. Puentes "H"

Un "Puente H" es un circuito de cuatro conmutadores que permite que un Motor DC gire en ambos sentidos al ser accionados de dos en dos. Un "Puente H" con transistores como conmutadores resulta una interface de potencia para un motor DC que simplifica la labor de la etapa digital.

La figura 1.5, muestra la conexión de los conmutadores de un Puente H simplificado, se puede ver que al accionar S1 y S4 se proporciona una tensión positiva al motor haciéndolo girar en un sentido, por el contrario cuando si se activan S2 y S3, el motor gira en sentido inverso. Es entonces, posible controlar el avance o retroceso de un motor DC con solo dos niveles lógicos por parte del sistema digital.

Figura 1.5. Puente H.



1.1.4. Generalidades sobre USB

1.1.4.1. Introducción

USB (Universal Serial Bus) es un protocolo de comunicación serial desarrollado en conjunto por varias empresas de tecnología importantes a nivel mundial, entre ellas Intel®, COMPAQ, IBM®, Microsoft®, por lo que su uso se ha vuelto un estándar a nivel mundial. Su aceptación se debe en parte por lograr transferencias de datos a gran velocidad, por la facilidad de manejo que se brinda al consumidor final, la amplia gama de posibilidades para el desarrollo de dispositivos de diferentes tipos, la apertura de recursos de código hacia los desarrolladores de dispositivos con los que se cuenta, entre otros.

El Protocolo USB nace de la necesidad por parte de las compañías que en ese entonces abarcaban el mercado de la tecnología, de contar con una norma genérica de comunicación entre el ordenador y los dispositivos que se conecten a este. Esto ha permitido que toda persona o entidad que lo desee pueda crear aparatos con soporte de tecnología USB; por otra parte, ha logrado que esta plataforma se popularice en todo el mundo, y que evolucione de una forma rápida.

1.1.4.2. Hardware del interface USB

- Cables y conectores.

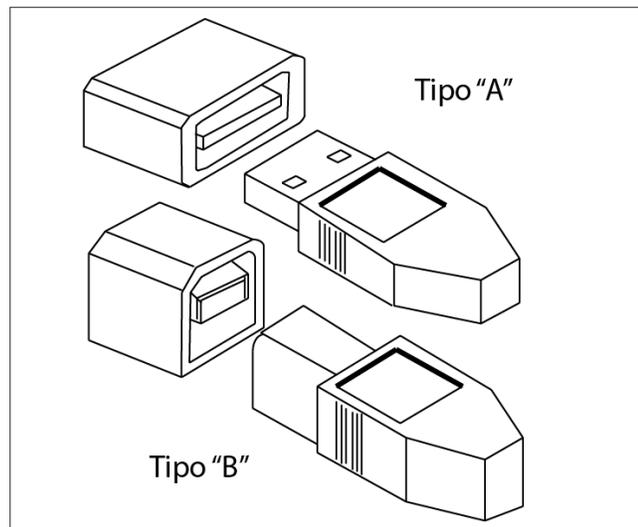
El bus USB es formado por 4 cables apantallados, dos sirven para alimentación (Vusb y Gnd) y dos para la transferencia de Datos (D+ y D-).

Existen conectores USB de diferentes formas, adaptándose de manera especial al "dispositivo", "Hub" o "Host" que lo incluya, sin embargo se los puede clasificar en dos tipos:

Conector tipo "A" orientado siempre hacia el ordenador. En la figura 1.6 se muestra un conector tipo "A" hembra que es el que posee los ordenadores, y un conector tipo "A" macho que es el del bus o cable.

Conector tipo "B" orientado hacia el dispositivo. La figura 1.6 muestra dos conectores tipo "B", uno hembra y uno macho de la misma forma.

Figura 1.6. Puertos USB tipo "A" y "B" machos y hembras.



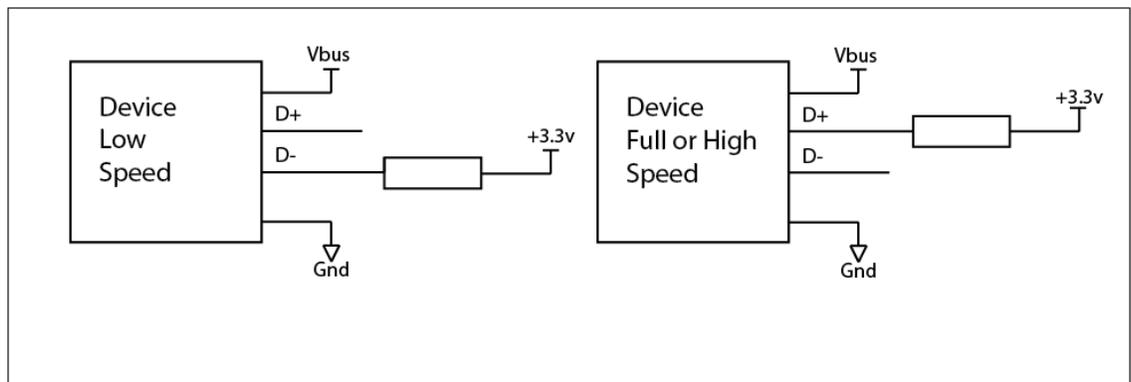
- Identificación de dispositivos.

Cuando un dispositivo es conectado a un Host, el primer proceso realizado por el host es la identificación de la velocidad a la cual trabaja el dispositivo. Actualmente existen tres estándares de dispositivos clasificados por su velocidad de transmisión de datos:

"Low Speed" (hasta 1,5 Mb/s), "Full Speed" (hasta 12 Mb/s) y "High Speed" (hasta 480 Mb/s).

La identificación de la velocidad es de forma física; el Host reconoce si la línea D- se conecta a una fuente de 3.3v a través de una resistencia de "Pull Up" y lo enlista como dispositivo "Low Speed", o si la línea D+ se conecta a la fuente de 3.3v, con la misma resistencia de "Pull Up", y determina que el dispositivo es "Full Speed"; en el proceso de enumeración, el Host comprueba si el dispositivo si es "Full Speed" o "High Speed".

Figura 1.7. Identificación de dispositivos "Low", "Full" y "High".



- Alimentación en el bus.

Como se ha visto, el protocolo USB se ajusta a las necesidades de los dispositivos en todos sus aspectos y simplifica su manejo por parte de los usuarios. Un puerto USB del Ordenador suministra corrientes de 100mA a 500mA, esto permite a los desarrolladores crear dispositivos alimentados solo por el puerto USB ("Bus Power Only") como memorias flash, teclados, ratones, etc. evitando la conexión de alguna fuente externa que resultaría molesta; sin embargo, si el dispositivo necesita más corriente, un dispositivo es "Self Power Mode" cuando tiene su propia fuente de poder y realizar la comunicación de datos con el ordenador sin consumir corriente de aquel.

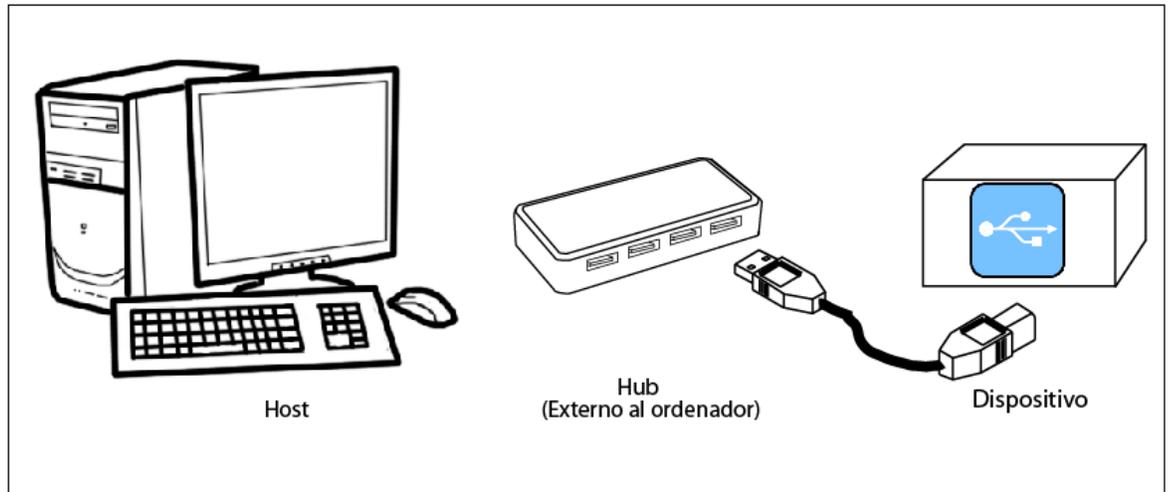
La identificación del tipo de alimentación que requiere el dispositivo, lo hace el Host en el proceso de enumeración a través de dos descriptores de configuración, uno indica si es "Self Power" o "Bus Power", y el otro descriptor indica la corriente que este necesita si es "Self Power".

1.1.4.3. Comunicación en el Bus

- Conceptos introductorios.

Los 3 componentes principales para establecer una comunicación USB son un Host, un Hub y un Dispositivo.

Figura 1.8. Conexión de un dispositivo USB a un Host a través de un Hub externo.

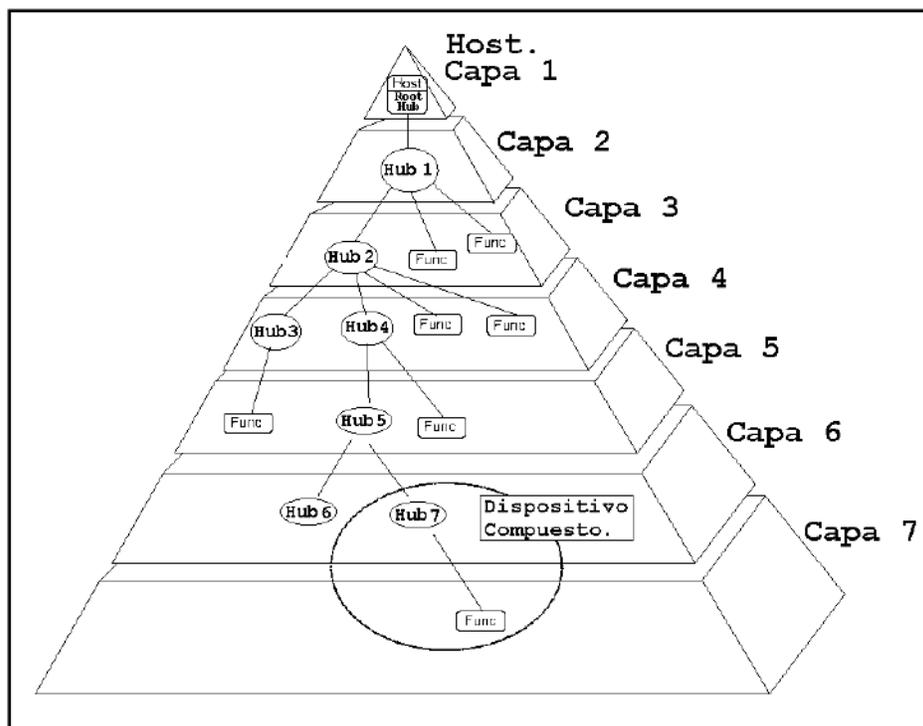


Host o anfitrión USB es en general un ordenador con capacidad USB, aunque hoy en día también puede ser una Tablet, un reproductor mp3, etc. Es entonces la máquina anfitriona de uno o varios dispositivos USB que maneja la parte más compleja de la comunicación; el Host es encargado de enumerar a un dispositivo, iniciar una transferencia de entrada, salida o configuración, tomar datos del dispositivo, entre otras cosas; El desarrollo de los dispositivos es por eso relativamente simple.

Un Hub o un concentrador es un aparato con dos o más puertos USB interno o externo al Host, cuya función es centralizar y ampliar la capacidad del Host de recibir dispositivos USB hasta un máximo de 127 dispositivos. Todo Host dispone de un Hub principal donde se puede conectar uno o varios dispositivos o Hubs secundarios. Un Hub secundario, a su vez, tiene la capacidad de albergar dispositivos o hubs, permitiendo así una topología en estrella por capas (figura 1.9).

Figura 1.9. Topología USB.

(Fuente: Arquitectura de Ordenadores. Lecturas Complementarias_8.)



Un dispositivo es un elemento que se conecta al Host vía USB y proporciona una o varias funciones a este. Un dispositivo compuesto es el que a más de proporcionar una función al Host tiene integrado un Hub. Un dispositivo es multifunción cuando posee varias interfaces que realizan diferentes actividades entre ellas, por ejemplo una impresora con escáner USB.

Ya que el Host es, en todos los casos, el que maneja la comunicación, los datos que van desde el Dispositivo hacia el Host se denominarán siempre datos de entrada y los datos que van desde el Host al Dispositivo, datos de salida. Estos términos se manejan tanto dentro del ordenador como dentro de los dispositivos.

Para entender cómo se realiza la comunicación USB entre el Host y un dispositivo se necesita entender ciertos conceptos que se describe a continuación:

Endpoint (EP) es un pequeño espacio de memoria que se encuentra tanto en el dispositivo como en el Host y es donde se almacena momentáneamente los datos que serán transferidos o que han sido recibidos. Un EP es de "entrada" cuando almacena

datos que serán recibidos por el Host y es de "salida" cuando almacena datos que serán enviados por el Host hacia el dispositivo. Un EP puede ser tipo "entrada/salida" si se divide de tal manera que una parte siempre sea de entrada y otra siempre de salida.

Pipe es el camino establecido entre el Host y el Dispositivo por donde fluirá la comunicación. Es establecido por el Host en seguida de la conexión de un Dispositivo durante el proceso de enumeración, para determinar la asociación entre los EPs del dispositivo y el controlador del Host, a demás de darle su dirección única dentro del concentrador.

Descriptores de Buffer o **Buffer Descriptor (BD)** son conjuntos de información específica de cada EP necesario para la comunicación, aquí se detalla la dirección del EP, su tamaño, si es de entrada, salida o ambos, etc.

- Base de la comunicación: transferencias.

USB es un protocolo de comunicación bastante amplio en el sentido de que cualquier aparato que extienda las funciones de un ordenador puede ser un dispositivo USB; esto hace del protocolo algo complejo para los desarrolladores, debido a la innumerable cantidad de posibilidades, a demás de la sincronización de envío o recibo de datos, verificación de datos, corrección de errores, etc. Es por eso que USB se divide en cuatro capas o niveles de programación, y los desarrolladores solo deben realizar los niveles más altos, dejando los más bajos a Controladores USB ya desarrollados.

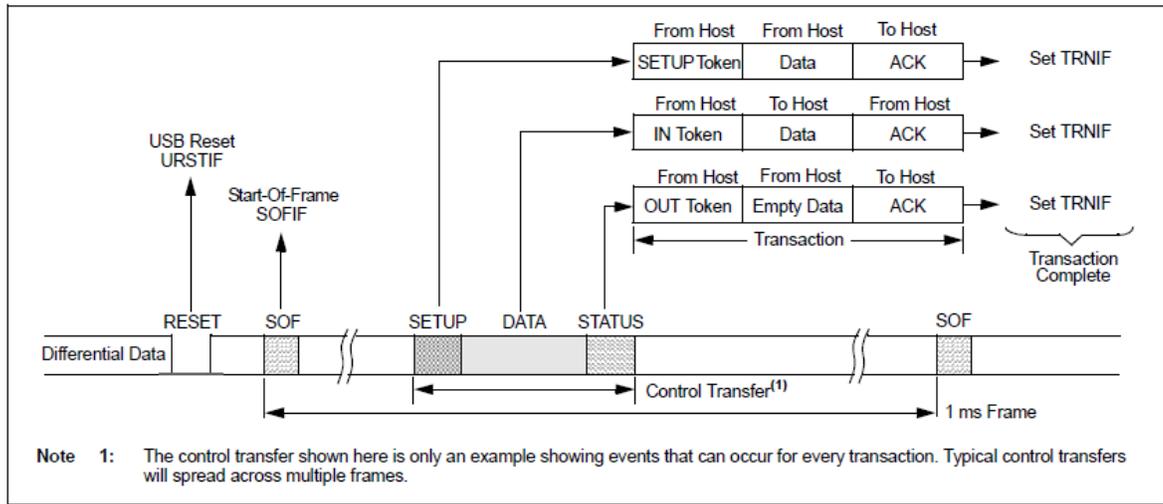
La capa superior de programación es la que corresponde a la Transferencias.

- Estructura de una transferencia.

La comunicación USB se compone de cuatro Capas que son de la más externa a la más interna: Transferencias, Transacciones, Paquetes y Campos. En la figura 1.10 se puede ver la interacción entre cada una de ellas.

Figura 1.10. Ejemplo de una transacción USB y eventos de interrupción en un PIC18F2455/2550/4455/4550.

(Fuente: Hoja de Datos MICROCHIP PIC18F2455/2550/4455/4550)



Un campo es un conjunto de bits que porta información para cumplir una función específica. Las funciones de los campos son: sincronizar el reloj del receptor y el transmisor (campo "Sync"); identificar el tipo de paquete que se transmite (campo "PID"); especifica el dispositivo al que corresponde el paquete según su dirección en el Hub (campo "ADDR"); A que EP corresponde el dato (campo " ENDP"); comprobar errores mediante un código de redundancia cíclica (campo "CRC"), entre otros. Los campos que forman un paquete tipo "TOKEN" se muestran en la figura 1.11.

Figura 1.11. Campos componentes de un paquete tipo "TOKEN".

(Fuente: USB en la Nutche)

Sync	PID	ADDR	ENDP	CRC5	EOP
-------------	------------	-------------	-------------	-------------	------------

Un paquete está conformado por campos y es el elemento primario en la comunicación USB. Un paquete completo viaja del dispositivo hacia el host o del host hacia el dispositivo. Los paquetes se clasifican por la información que llevan en: paquetes tipo "Token", paquetes de Datos, paquetes "Handshake" y paquetes de "Inicio de Trama". Los paquetes "Token" siempre van del host al dispositivo e indica el tipo de transferencia que se va a realizar ("In", "Out" o "Setup"); un paquete de datos es el que uno de sus campos es el dato que se envía o recibe, puede viajar del host al dispositivo o al revés; los paquetes "Handshake" sirven para reportar la correcta

recepción del dato o caso contrario indicar que hubo error; por último un paquete de "Inicio de Trama" o "Start Of Frame" (SOF) es enviado cada cierto tiempo por el Host para comenzar una transferencia.

Toda comunicación la debe iniciar el host, así que el dispositivo no puede enviar paquetes sin que antes el host le informe que está dispuesto a recibirlo, y tampoco puede recibir datos sin que el host le informe que está próximo a enviarlos.

Una Transacción es el envío de un paquete "token" por parte del host donde indica si quiere recibir o enviar el próximo paquete al dispositivo, el posterior envío de un paquete de datos y finaliza con un paquete "Handshake" por parte del receptor indicando si la recepción fue correcta o tuvo algún error.

Una Transferencia es el conjunto de transacciones realizadas, y es la capa más amplia de comunicación USB.

- Tipos de transferencias.

Hay cuatro tipos de transferencia definidos en la especificación USB 2.0 que son: de control, de volumen, por interrupción e Isocrónica

Las **transferencias de Control** son usados normalmente en operaciones de comando y de estado. Son esenciales para configurar un dispositivo USB con todas las funciones de enumeración. Los paquetes son iniciados por el Host y la longitud del paquete de control depende de la velocidad del dispositivo. En dispositivos low speed es de 8 bytes, en dispositivos full speed es de 64 bytes y dispositivos high speed permiten tamaños de paquete de 8, 16, 32 o 64 bytes

En una transferencia de control se distinguen tres etapas:

Transacción "SETUP" donde la solicitud por parte del Host es enviado. Consiste en 3 paquetes: El primero que se envía es un "SETUP TOKEN", que contiene la dirección y el numero de EP. El paquete enviado después es el "DATA PACKET", este siempre tiene un campo "PID" tipo "DATA0" e incluye un "SETUP PACKET" que se detalla en la solicitud. El último paquete es "handshake" usado para acreditar una recepción exitosa o para indicar un error.

Transacción "DATA" es opcional y consiste en una o múltiples transferencias IN o OUT. la solicitud en el SETUP indica la cantidad de datos a ser transmitidos en esta etapa. Si excede el máximo tamaño del paquete, el dato será enviado en varios paquetes, es decir, se divide para que si no puede entrar en el tamaño de un paquete le manda en dos o tres.

Transacción "STAGE" reporta el estado de todas las solicitudes y siempre es hecho por el dispositivo.

Las **transferencias de Volumen** son utilizadas en comunicaciones que requieran grandes cantidades de flujo de datos. Ejemplos de esto son los datos enviados a una impresora o una imagen generada desde un scanner. Las transferencias de Volumen proveen una corrección de error en forma de un campo CRC16 del dato cargado y mecanismos de detección de error-retransmisión que aseguran que el Dato sea transmitido y recibido sin error.

Las transferencias de Volumen son solo soportados por dispositivos Full y High speed. Para Full speed, el tamaño de paquetes es de 8, 16, 32 o 64 bytes; para High speed, es de hasta 512 bytes. Si el dato cargado es más pequeño que el tamaño máximo del paquete, no necesita ser llenado con ceros. Una transferencia de Volumen se la considera completa cuando se ha transferido la cantidad exacta de Datos requeridos o transferido un paquete de longitud cero.

En las **transferencias por Interrupción** el dispositivo genera una interrupción, sin embargo, para ser atendido debe esperar a que el Host lo encuente. La máxima longitud de carga de datos para Low Speed es de 8 bytes, para Full Speed es de 64 bytes y para High Speed es de 1024 bytes.

Las **transferencias Isocrónicas** ocurren continua y periódicamente. Contienen típicamente información sensible al tiempo como Audio y flujo de video. Si algún dato de audio que se trasmite es errático, el ritmo ya no tendría sincronía, sin embargo, si un paquete fuera dado continua y periódicamente, es menos probable que sea notado por el que escucha.

La máxima longitud de la carga es especificada en el descriptor del EP de un "EP Isocrónico", que puede ser hasta un máximo de 1023 bytes para Full Speed y 1024

para High. Como la longitud de la carga tiene relación directa con el ancho de banda del bus, es una buena práctica especificar una longitud de carga conservadora.

Las transacciones Isocrónicas no tienen etapa de Handshaking.

- Lado del dispositivo: descriptores.

Un dispositivo tiene que ser reconocido por el host para que este regularice la comunicación USB, esto se hace mediante descriptores que el host pide al dispositivo en la etapa de enumeración. Todos los dispositivos USB tienen una jerarquía de descriptores que describen al host información como: que dispositivo es, quién lo hizo, que versión de USB soporta, de cuantas formas puede ser configurado, el número y tipo de EPs que tiene , etc.

Los descriptores más comunes son: Descriptores del Dispositivo, Descriptores de Configuración, Descriptores de Interface, Descriptores de EP, Descriptores String.

Los Dispositivos USB tienen un solo descriptor de dispositivo, que incluye información acerca de que versión USB soporta, el identificador de Producto y Vendedor usados para cargar los controladores apropiados y el posible numero de configuraciones que el dispositivo pueda tener. El numero de configuraciones indican cuantas ramas de descriptores de Configuración hay para seguir.

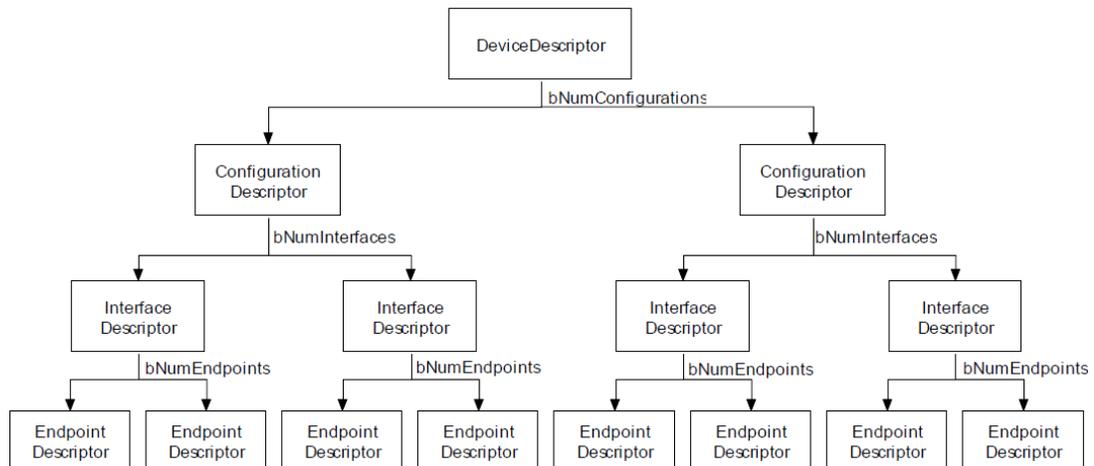
Los descriptores de Configuración especifican valores como la corriente que esta configuración en particular utiliza, si el dispositivo requiere de alimentación externa, y el numero de interfaces que tiene.

Los descriptores del interface podrían ser vistos como un encabezados o una agrupación de los EPs en un grupo funcional realizando una sola función del dispositivo. Por ejemplo , un multifunción "fax/scanner/impresora", el descriptor de Interface 1 podría describir los EPs del "fax", el descriptor de interface 2 la del "scanner" y la 3 la de la "impresora". A diferencia del descriptor de configuración, no hay limitación en tener un interface habilitado a la vez, un dispositivo podría tener uno o varios descriptores de interface habilitados al mismo tiempo.

Cada descriptor de EP es usado para especificar el tipo de transferencia, dirección, intervalo de encuesta y tamaño máximo de paquete. El EP 0 es un EP de control por defecto, se le asume siempre como EP de control y jamás como un descriptor.

La flexibilidad del protocolo USB permite que un dispositivo pueda tener una o más configuraciones, y que cada configuración llegue a tener uno o más interfaces, por lo que pueden existir varios descriptores de Configuración e Interface, esto no pasa en el caso de los descriptores de Dispositivo, ya que aunque existan varias configuraciones, la información del dispositivo será siempre la misma.

Figura 1.12. Mapa de descriptores
(Fuente: Especificación USB 2.0)



- Diseños basados en clases de dispositivos.

Se define como clase al conjunto de dispositivos que realizan tareas similares y por lo tanto mantienen los mismos protocolos de comunicación y requerimientos de transporte de datos entre ellos.

Diferentes dispositivos de una misma clase utilizan un mismo controlador por parte del host para poder comunicarse. Las clases son definidas por la especificación 2.0.

Las clases identifican el comportamiento de un dispositivo, por ejemplo si es de almacenamiento masivo, de interface humana, etc.

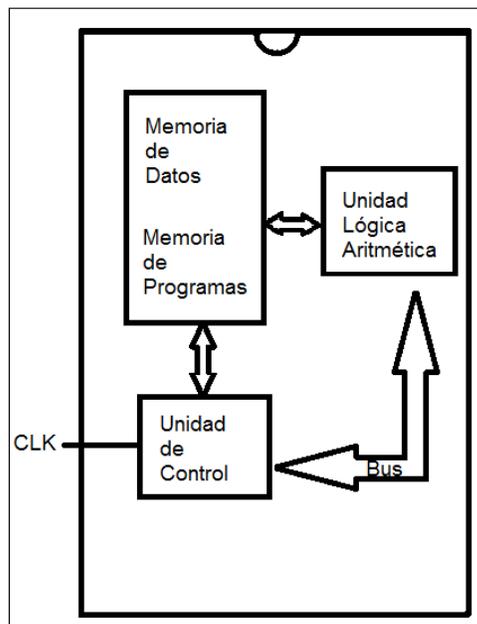
1.1.5. Micro controladores

1.1.5.1. Concepto general

Un microprocesador es un circuito integrado programable capaz de asumir, en principio, cualquier función digital. Un "microcontrolador" (uC) incorpora las características de un microprocesador más dispositivos periféricos adicionales, tales como puertos de entrada/salida, memoria, temporizadores, convertidores analógicos digitales, entre otros.

Esquemáticamente, un uC en su forma básica, se divide en: Memoria de datos, Memoria de programas. Unidad de Control, Unidad Lógica Aritmética y Buses (figura 1.13).

Figura 1.13. Esquema básico de un microcontrolador.



La Memoria de Datos, es un pequeño espacio de memoria donde se almacenan los registros que ocupan las demás partes del uC necesarios para su funcionamiento. Los uC suelen poseer dos tipos de memorias de datos: la memoria volátil (por ejemplo memoria RAM) cuyos datos permanecen si el sistema está funcionando, si se corta la alimentación al uC los datos en este tipo de memoria se pierden; y la memoria permanente (EEPROM) donde los registros permanecen aunque el uC no esté alimentado.

Un registro es un número en binario o, visto de otro modo, un conjunto de bits de un tamaño dado; este tamaño denominado "tamaño de palabra" determina la capacidad de las operaciones matemáticas y lógicas del uC; los tamaños de palabra más comunes son 8, 12, 14, 16, 32 y 64 bits; los registros son de "Propósito Específico" cuando son indispensables para el correcto funcionamiento del uC o de "Propósito General" cuando son usados solamente en algún por un firmware específico.

La Memoria de Programas es siempre permanente, y es el espacio de memoria destinado a guardar las instrucciones que el uC debe ejecutar mientras esté funcionando.

La Unidad de control se encarga de la decodificación y ejecución de las instrucciones, manejo de periféricos, acceso y manipulación de los registros, etc. La Unidad de Control coordina todas las actividades dentro del uC, incluyendo la propia unidad, todo el flujo de datos a través de los buses y los procesos que realice la unidad lógica aritmética

Por último, la Unidad Lógica Aritmética (ULA) realiza todas las operaciones matemáticas y booleanas entre los registros, un uC se basa totalmente en estas operaciones, a partir de esta unidad se crean un sin fin de aplicaciones específicas, sin contar con el hecho de la natural evolución de los microcontroladores, que a la fecha integran una variedad de módulos especializados en ciertos procesos electrónicos.

Microchip Technology Inc. es una empresa fabricante que provee una amplia gama de micro controladores PIC (*Peripheral Interface Controller* ó controlador de interfaz periférico) muy fiables y de bajo costo. Integra en su arquitectura gran cantidad de módulos periféricos reduciendo el número de pines del circuito integrado.

1.1.5.2. Especificaciones generales del PIC18F4553

La familia PIC18FXXXX es una familia moderna de microcontroladores PIC de 8 bits de tamaño de palabra; entre los cuales, los PIC18F2455/2550/4455/4550 son uC de alto rendimiento, flash mejorada y poseen periférico USB.

El PIC18F4550 es un integrado de 40 pines, tiene una memoria de programas de 32Kbytes de tamaño. Posee múltiples fuentes de interrupción que pueden ser configurados como de alta o de baja prioridad, puertos de entrada o salida, cuatro

módulos temporizadores, modulo compara/captura/PWM, modulo convertidor analógico a digital (A/D) de 10 bits de resolución, entre otros.

Otra ventaja del PIC18f4550 es que posee arquitectura "Harvard", esto es, dispone de diferentes buses para la memoria de datos y la memoria de programas, para así accederlas simultáneamente. Se puede entonces, ejecutar una instrucción que requiera un dato mientras se lee la memoria de programa para ejecutar la siguiente instrucción.

El PIC18F4553 es análogo al PIC18F4550 salvo la resolución de convertidor A/D que en este caso es de 12 bits.

1.1.5.3. Convertidor A/D integrado a los micros PIC18F4553

El convertidor A/D del uC PIC18F4553 tiene 13 entradas analógicas que pueden ser multiplexadas entre ellas de tal forma que no sea significativa la perdida de datos en este proceso. El resultado es entregado con una resolución de 12 bits en los registros ADRESH y ADRESL. El módulo es controlado por tres registros llamados ADCON0, ADCON1 y ADCON2.

En el registro **ADCON0** se tiene tres funciones, la primera es escoger el canal o la entrada que el módulo A/D va a usar, la segunda prender o apagar el módulo y la tercera es empezar la conversión. La función de empezar la conversión se hace mediante un bit ($ADCON0<1>$) que además funciona como bandera, al limpiarse automáticamente el momento en que el convertidor acabe su tarea.

En el registro **ADCON1** se escoge el voltaje de referencia para el convertidor, V_{ref+} y V_{ref-} , que puede ser un voltaje externo (conectado a los pines AN2 y AN3) o el voltaje de alimentación del PIC18F4553 (V_{ss} y V_{dd}). También se configura los pines como entradas/salidas digitales o como entradas analógicas (todas las entradas que use el módulo deben ser configuradas como entradas analógicas.)

Por último en el registro **ADCON2** se escoge el formato en que se presentan los resultados, si se justifica a la derecha o izquierda (esto permite usar un sólo registro de resultado, ADRESH, si se justifica a la izquierda perdiendo los 4 bits de resolución menos significativos), y se selecciona el tiempo de adquisición. (Sección 2.0 de la hoja de datos del PIC18F4553)

1.1.5.4. Módulo USB integrado a los micros 18F4553

El microcontrolador PIC18F4553 puede comunicarse con el ordenador mediante USB a velocidades Full Speed y Low Speed a través de un transmisor-receptor interno que cuenta con ingeniería de Interface Serial. (SIE por sus siglas en inglés). Este módulo realiza toda la conformación de campos y paquetes permitiendo al diseñador que realice el programa a nivel de transacciones.

- Estructura general (hardware)

El SIE puede ser conectado directamente al puerto USB de un host, activando un transceptor interno o puede conectarse a través de un transceptor externo. Del mismo modo, el PIC18F4553 incluye un regulador de voltaje a 3.3v interno que puede ser activado evita el uso de uno externo. Por último, dispone de dos resistencias internas entre el voltaje de salida del regulador y cada una de las líneas de datos D+ y D-, que sirven para configurar al dispositivo como "Low" o "High Speed". En la figura 1.14 se observa el esquema interno relacionado al módulo USB.

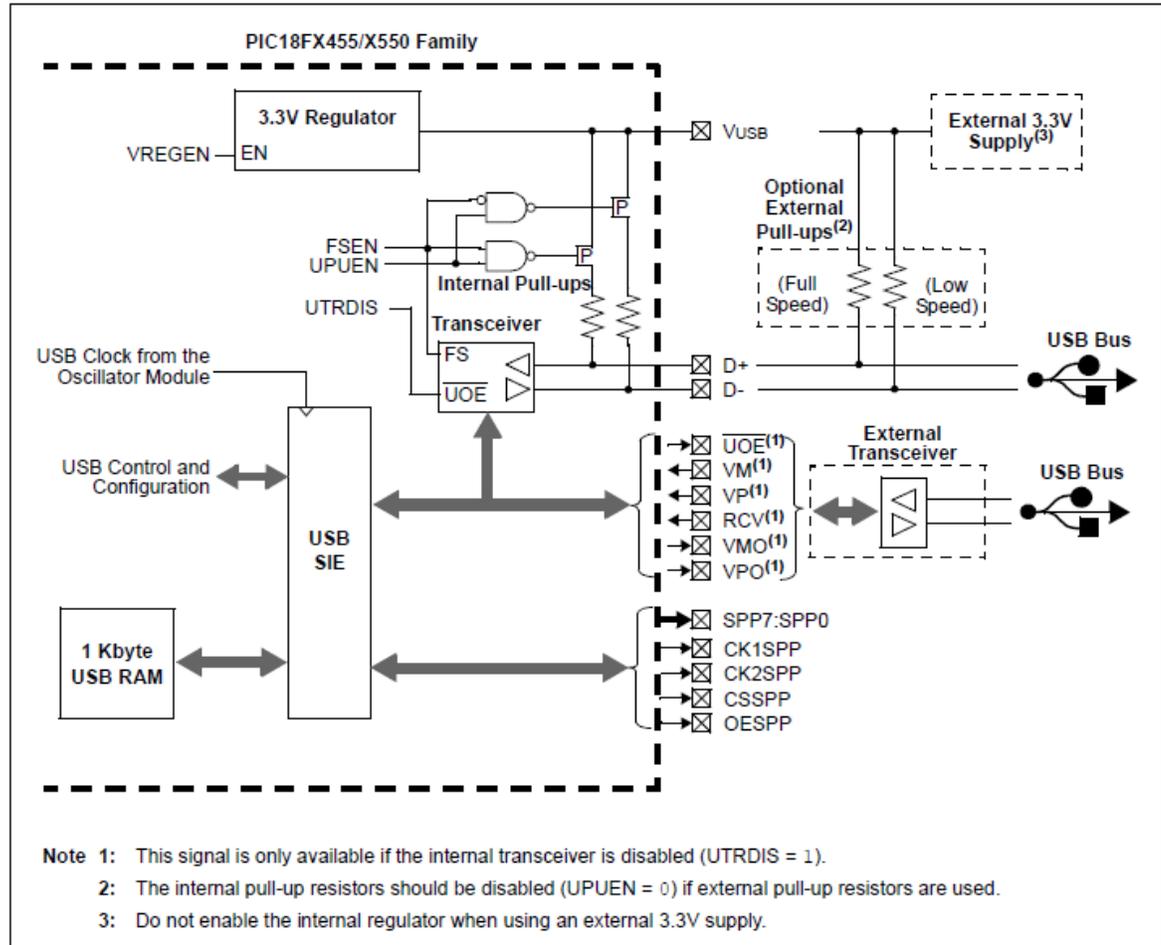
En el registro UCFG se configuran todos estos aspectos.

- Memoria USB.

El Microcontrolador utiliza un espacio específico de la memoria RAM para la comunicación comprendido entre los bancos 4 y 7 de la RAM. El banco 4 es destinado a descriptores de Buffer o "Buffer Descriptors" (BD) pero puede a demás tener Endpoints (EP). Los bancos del 5 al 7 están orientados a contener únicamente EP.

Un BD se compone por cuatro registros que indican el estado, configura el tamaño y la dirección en la que comienza el EP en cuestión. Un EP puede tener hasta dos BD si es configurado como EP de entrada/salida. Por ejemplo si el EP0 de entrada comienza en la dirección hex500 y es de 8 bytes de tamaño y EP0 de salida comienza en hex508 y es de 64 bytes, cada parte del EP necesita su propio BD.

Figura 1.14. Esquema del módulo USB y opciones
(Fuente: Hoja de Datos MICROCHIP PIC18F2455/2550/4455/4550)



Los datos que son enviados hacia el host se colocan en un EP de entrada (tomar en cuenta que toda transacción hacia el host se denomina de entrada), posteriormente se configura el BD correspondiente a su estado y el host los tomará. Del mismo modo, todos los datos que lleguen se almacenan en un EP de salida.

- Registros para controlar el módulo USB.

Existen tres registros principales que controlan el módulo USB, son: registro de Control USB (UCON), registro de Configuración USB (UCFG) y registro de Estado de Transferencia USB (USTAT).

El registro de Control USB (UCON), controla el comportamiento del módulo durante las transferencias. En este registro se enciende o apaga el módulo, se lo deja en modo de

suspensión (aquí el SIE está inactivo), se habilita la recepción de paquetes, entre otras funciones.

El registro de Configuración USB (UCFG), configura propiedades del Hardware Interno del módulo, además de la velocidad de transferencia USB, este, a diferencia otros registros de control, no se debe modificar una vez que el modulo este prendido. Para ver las posibilidades de configuración del Hardware Interno referirse a la hoja de especificaciones de PIC18F2455/2550/4455/4550, sección 17.2.2

El registro de Estado de Transferencia USB (USTAT), reporta el estado de las transacciones que se han producido en el módulo USB, como la dirección de la última actividad, o el EP que realizó la última actividad.

Existen otros registros relacionados con la comunicación USB pero que no inciden en el funcionamiento del módulo, estos son:

Registro de dirección de Dispositivo USB (UADDR) que contiene la dirección del dispositivo que después de la fase de enumeración el host le asigna.

Registros de control de EP que los habilita y controla, y los descriptores de Buffer que indican el estado, configura el tamaño y la dirección de cada EP.

- Sistema de Interrupciones USB.

El módulo USB tiene varias fuentes de interrupción que son manejados por los registros UIR y UIE. El registro UIR es un registro exclusivamente de banderas que se fijan en 1 si cierta actividad se produjo en el módulo. Los bits del registro UIE establecen los eventos que producirán una interrupción. En la tabla 1.1 se especifica las fuentes de interrupción, y los bits correspondientes a los registros UIR y UIE.

Tabla 1.1. Eventos en el modulo USB que pudieran producir una interrupción

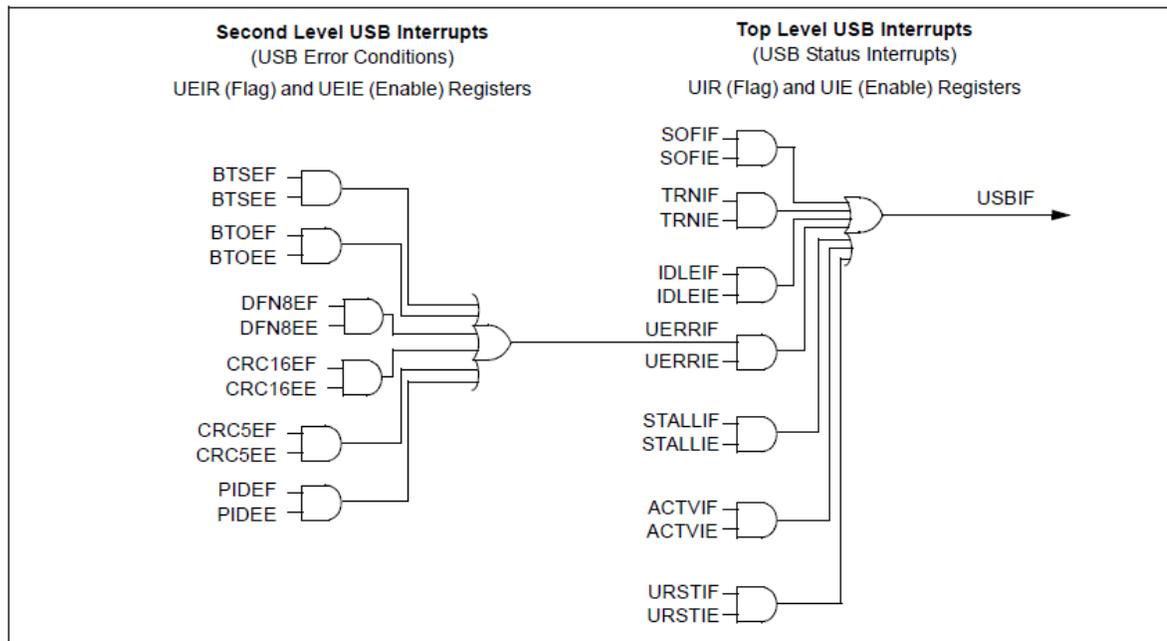
Evento	UIR	UIE
El SIE ha recibido un token de inicio de trama (SOF)	bit 6	bit 6
El SIE ha enviado un error tipo STALL	bit 5	bit 5
Se ha detectado una condición de Idle	bit 4	bit 4
Se ha completado una transacción	bit 3	bit 3
Se ha detectado actividad en las líneas D+/D-	bit 2	bit 2
Ha ocurrido un error	bit 1	bit 1
Ha ocurrido un reinicio (reset)	bit 0	bit 0

No es necesario que la interrupción esté habilitada para que la bandera correspondiente reaccione si se ha producido el evento; por este motivo no es necesario que el programa funcione por interrupciones, sin embargo se debe encuestar todo el tiempo acerca de la actividad que ha ocurrido.

Hay dos bits generales relacionados con las interrupciones USB, un bit de habilitación USBIE (PIE2<5>) que permite que los distintos eventos creen una interrupción y USBIF (PIR2<5>) que indica si hubo interrupción USB. El módulo USB posee una estructura lógica propia, que puede generar múltiples condiciones de interrupción, esta debe ser encuestada cada vez que el bit USBIF este en uno.

En la figura 1.15 se aprecia la lógica de interrupciones proporcionado por Microchip:

Figura 1.15. Lógica funcional de Interrupciones USB.
(Fuente: Hoja de Datos MICROCHIP PIC18F2455/2550/4455/4550)



1.1.6. Sistemas SCADA

1.1.6.1. Concepto

Sistema basado en computador para la adquisición, control y monitoreo de datos. SCADA es el acrónimo de Supervisión, Control y Adquisición de Datos (Supervisory Control And Data Acquisition).

SCADA es un sistema de arquitectura centralizada que permite supervisar y controlar variables a distancia dentro de un proceso automático. A continuación se describe el proceso típico de un sistema SCADA

Los componentes principales de un Sistema SCADA son: HMI(Human Machine Interface), MTU(Master terminal Unit), PLC(Programable Controler Logic) y RTU (Remote Terminal Logic).

1.1.6.2. Teoría sobre SCADA

La variable física que se desea medir, debe ser acoplada para ser entendida por el sistema, generalmente, un sensor o un transductor transforma la variable a medir en

una variable eléctrica, ya sea voltaje, resistencia, corriente, capacitancia, etc. Por ejemplo, si el fenómeno físico a medir es la temperatura, un sensor LM32 la transformará en voltaje, una termo resistencia, en variación de resistencia, etc.

Una vez que la señal pasa a ser una variable que el sistema pueda entender, esta debe ser acoplada o acondicionada, por lo que se la limita su rango por medio de amplificadores o atenuadores, además de filtros y aislación eléctrica para proteger al sistema.

A continuación, se digitaliza la señal por medio de circuitos convertidores analógico a digital para trasladar todos los datos correspondientes a la señal hacia el ordenador, por cualquier tipo de comunicación que sea elegido.

Se procesan los datos correspondientes a la señal y se los muestra en escala real y en tiempo real. Según esta señal, se toma decisiones de manera manual (por iniciativa del operario) o automáticamente (Control ON/OFF, PID, etc.) para realizar una acción de control sobre el proceso.

La salida de control, hecha desde el ordenador debe ser capaz de manejar uno o varios dispositivos dentro del proceso, por lo que esta señal digital debe ser convertida en señal analógica y acoplada para el manejo del dispositivo en cuestión.

1.2. Diagnóstico del Sistema Automático de Muestreo

1.2.1. Introducción

El Laboratorio de Biología de la facultad de Ciencia y Tecnología dispone de una máquina electromecánica de muestreo automático totalmente funcional, marca Continflo, y en bodega varias máquinas disfuncionales que se pretenden recuperar. El sistema de muestreo que se propone armar, usará los elementos mecánicos de dichas máquinas, por lo que en este apartado se explica su funcionamiento.

- Análisis y descripción de la estructura electromecánica del equipo.

La máquina de muestreo Continflo, funciona con dos elementos activos por lo que se puede dividir el análisis de la máquina en dos partes:

Motor AC de 220 voltios, con un sistema de engranes que distribuyen el movimiento, piñones y cadenas, para el movimiento secuencial de los recipientes contenedores de las muestras.

El motor AC es alimentado para hacer rotar las muestras en la bandeja superior. Las características de este motor son: voltaje de alimentación 220 voltios en corriente alterna, corriente de 4,5 amperios.

El eje del motor AC, traslada su movimiento a una central de engranes que tiene por salida el movimiento síncrono de dos ejes con velocidad reducida y torque aumentado; estos ejes a su vez mueven dos piñones que a través de dos cadenas, sostenidas en los extremos por dos piñones más, y con tres varillas sobresalientes en cada cadena, rota contenedores alargados de 10 muestras por contenedor.

Figura 1.16. Funcionamiento Motor AC, vista frontal.

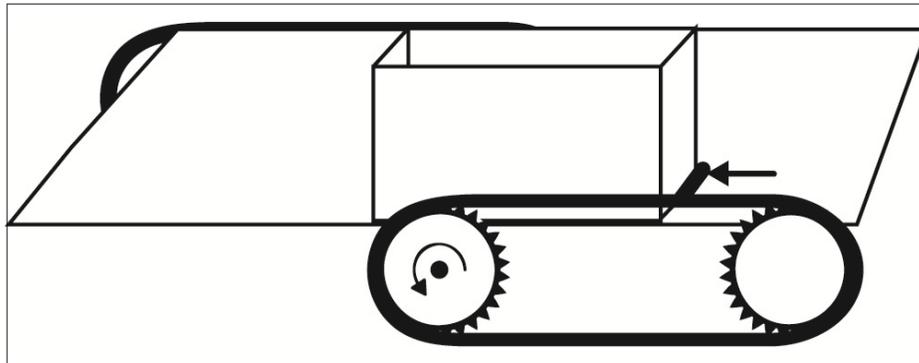
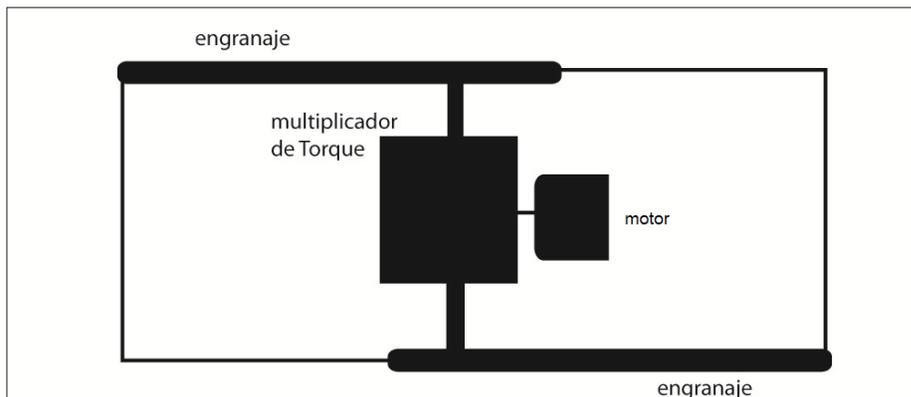


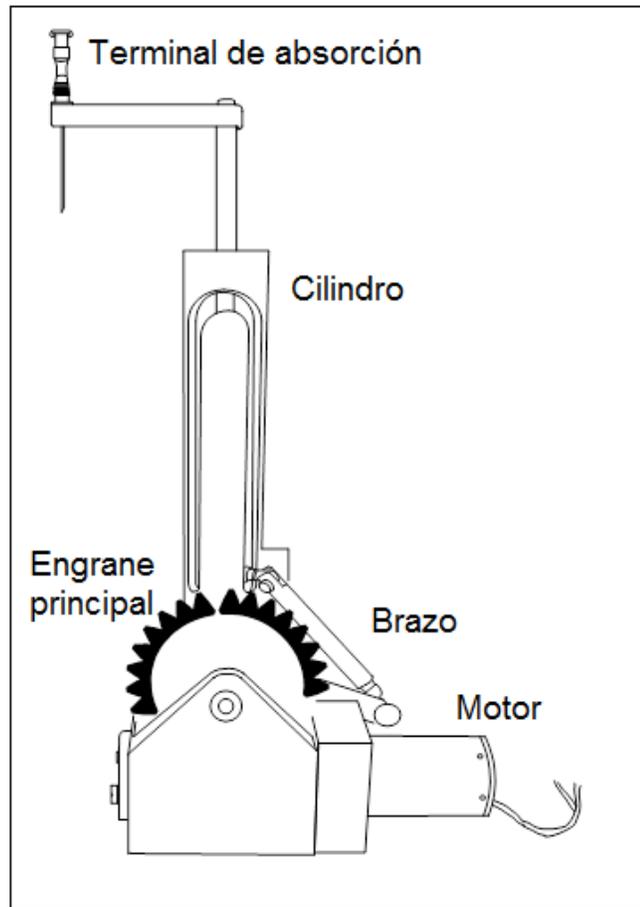
Figura 1.17. Disposición de elementos relacionados al motor AC, vista inferior.



Motor DC de 12 voltios, y sistema de encaminamiento del terminal de succión.

El terminal de succión se mueve por alimentación directa o inversa de un motor DC de 12 voltios y de 0,7 A de corriente, (la corriente máxima del motor es de 1,2A). En la figura 1.18 se ven los componentes mecánicos que transforman el movimiento circular del motor a un movimiento útil del terminal.

Figura 1.18. Terminal de succión en estado inicial



El eje del motor DC mueve al engrane principal (hacia la izquierda en la imagen) cuando al motor se lo alimenta en forma directa, moviendo al brazo, desde su estado inicial, de tal forma que este subirá llevando consigo un émbolo interno que sujeta al terminal de absorción a través de una varilla vertical y una horizontal. Cuando el émbolo está en el punto más alto, debido al movimiento del brazo por el canal del cilindro, éste rota 90 grados permitiendo al terminal de absorción girar hacia adentro, para luego descender por el otro lado del canal insertando así al terminal en la muestra.

Si se invierte la dirección de giro del motor, el engrane principal (que ahora gira a la derecha en la imagen), moverá nuevamente el émbolo interno y por ende el terminal de absorción, para que en el punto más alto, este gire hacia afuera y terminando en la posición inicial.

1.2.2. Ciclo funcional del equipo

La máquina de muestreo funciona de la siguiente manera:

- 1) El terminal de absorción entra al recipiente que contiene la muestra debido a que el motor DC se alimenta normalmente; un sensor "Fin Carrera" detiene al motor. (El movimiento detallado del terminal se explica en la sección 1.2.1).
- 2) Se espera un tiempo determinado para la absorción, atomización y el análisis correcto de la muestra. Este tiempo es configurable.
- 3) El terminal de absorción sale del recipiente de la muestra gracias a la alimentación negativa del motor DC; un segundo "Fin Carrera" detiene al motor.
- 4) La posición de los recipientes rota para que el siguiente se coloque en posición de absorción; esto porque el motor AC es activado hasta que un último "Fin Carrera" lo detiene cuando la nueva muestra esté en posición.
- 5) Se espera nuevamente un tiempo suficiente, que también es programable, para que disminuya el nivel de la muestra, además de limpiar el camino y el terminal de absorción absorbiendo agua.
- 6) Una vez concluido este tiempo, se repite el proceso.

Es imperiosa la necesidad de que los tiempos de espera se cumplan con la mayor precisión posible, ya que son estos los que determinaran las curvas de análisis en el ordenador, y según cada muestra es necesario también que estos tiempos se puedan elegir para satisfacer las particularidades de cada muestra.

CAPITULO 2

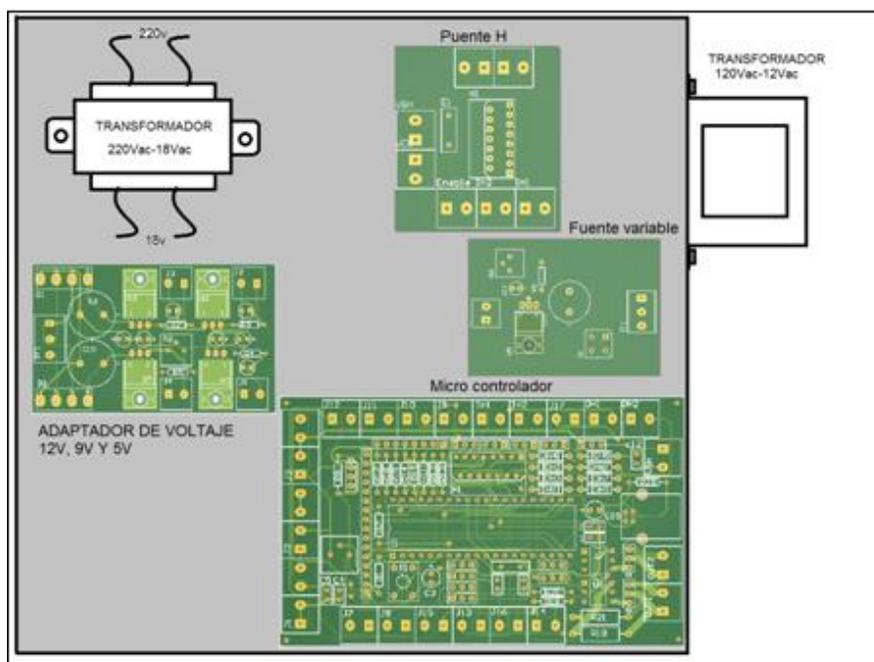
HARDWARE

2.1. Máquina de Muestreo

2.1.1. Introducción

El hardware de la máquina de muestreo está compuesto de cuatro tarjetas y dos transformadores de corriente alterna para su alimentación. En la figura 2.1 se observa la disposición de los elementos que componen el hardware. Dichos elementos son:

- Fuente de alimentación para el uC:
 - Transformador de 120V AC a 12V AC.
 - Adaptador de voltaje de 12V AC a 12V, 9V y 5V DC.
- Fuente de alimentación para motor DC:
 - Transformador de 220V AC a 18V AC.
 - Adaptador de voltaje de 18V AV a variable DC.
- Tarjeta del microcontrolador.
- Tarjeta de potencia "Puente H".

Figura 2.1. Disposición de los componentes en la máquina de muestreo.

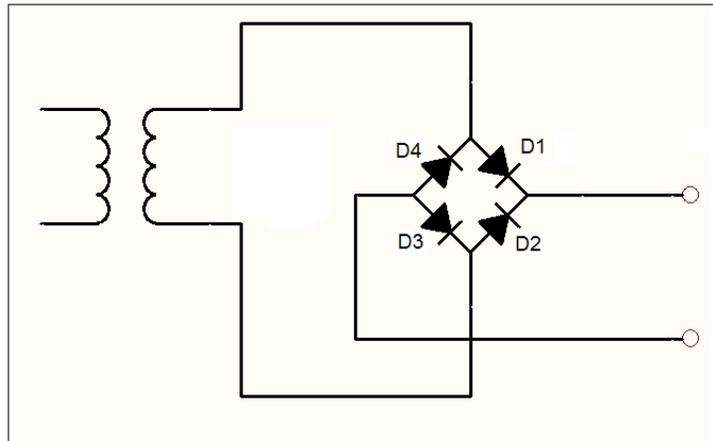
2.1.2. Fuente de alimentación para el uC

La fuente de alimentación para el uC, se compone de un transformador que reduce la corriente de la red de 120V AC a 12V AC y un circuito electrónico que adapta el voltaje a la salida del transformador en voltaje de corriente directa (DC) valiéndose de un rectificador, un filtro y un regulador de voltaje.

Un transformador es un dispositivo que reduce o aumenta el nivel de una onda de corriente alterna conservando, idealmente, su frecuencia y su potencia. Esta conversión se realiza por inducción electromagnética y sin que exista conexión física. La relación entre el voltaje de ingreso y el voltaje de salida es proporcional a la relación entre el número de espiras del devanado primario y el número de espiras del secundario. El transformador utilizado tiene una relación 10/1 (reduce 10 veces el voltaje de entrada).

El circuito para adaptar voltaje AC a DC empieza con un rectificador de onda completa construido con un "Puente de Greatz" que consiste en un arreglo de cuatro diodos. La función de un rectificador de onda completa es convertir la parte negativa de la onda en positiva. En la figura 2.2 se observa un "Puente de Greatz" conectado al secundario de un transformador.

Figura 2.2. Puente de Greatz.



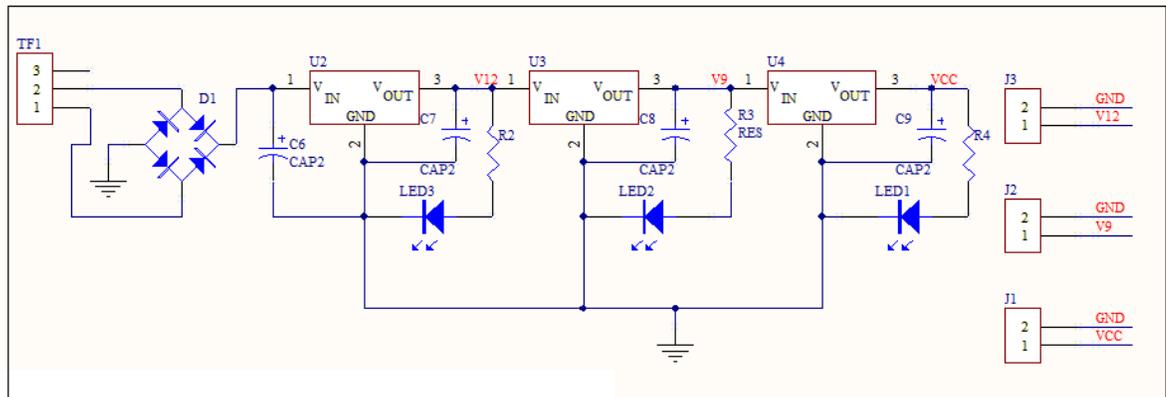
Cuando la onda es positiva, los diodos D1 y D3 permiten su paso sin alterar su polaridad; los diodos D2 y D4 invierten la polaridad de la onda cuando es negativa, por lo que en los terminales de salida se obtiene una onda positiva. La señal de salida es considerada como DC.

A la salida del rectificador de onda completa, se realiza un filtrado de la señal por medio de un capacitor. El objetivo del filtrado es reducir la componente AC de la señal aprovechando la demora que presentan los capacitores para descargar su voltaje, así, cuando el capacitor se esté descargando, la onda nuevamente iniciará su carga.

Un regulador de voltaje es un circuito integrado que estabiliza la señal de entrada a un voltaje DC determinado.

En la figura 2.3 se ve el esquema de la tarjeta "Adaptador de Voltaje" y el proceso de conversión de una onda de corriente alterna de 120v a 60hz (onda de la red) a tres niveles de voltaje de corriente directa. En la bornera TF1 se ha de conectar el devanado secundario del transformador. Las salidas de voltaje en DC se encuentran en las borneras J1 (5v y GND), J2 (9v y GND) y J3 (5v y GND).

Figura 2.3. Esquema de la tarjeta de Alimentación.

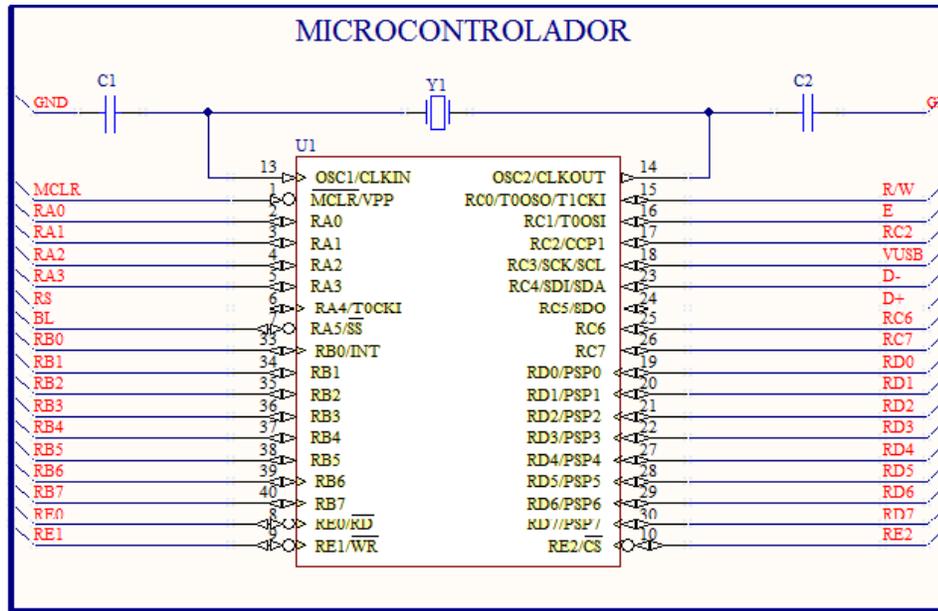


2.1.3. Tarjeta del Microcontrolador

2.1.3.1. Microcontrolador

El control de la máquina de muestreo se realiza mediante un microcontrolador PIC18F4550 de la casa fabricante "Microchip Technology Inc.", este es un integrado de 40 pines, de alto rendimiento y resistencia. El circuito elemental del uC consiste en su alimentación y un oscilador conectados a los pines correspondientes. En la figura 2.4 se observa el circuito esquemático que se aplicó en la máquina de muestreo.

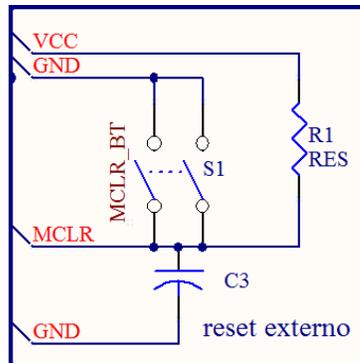
Figura 2.4. Esquema del uC PIC18F4550.



Los valores de VCC y GND son 5V y 0V respectivamente.

Un pin del uC llamado MCLR (pin 1) tiene la facultad de reiniciar el firmware si se lo a conecta a GND. MCLR debe conectarse a VCC por medio de una resistencia de PULL UP (R1 en la figura 2.5) y a un pulsante normalmente abierto (MCLR_BT) conectado a GND.

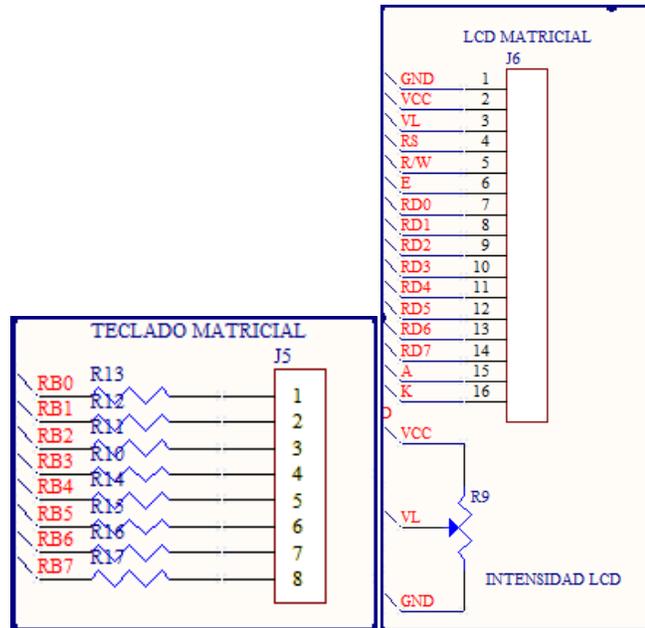
Figura 2.5. Esquema Circuito MCLR.



2.1.3.2. LCD y teclado matricial

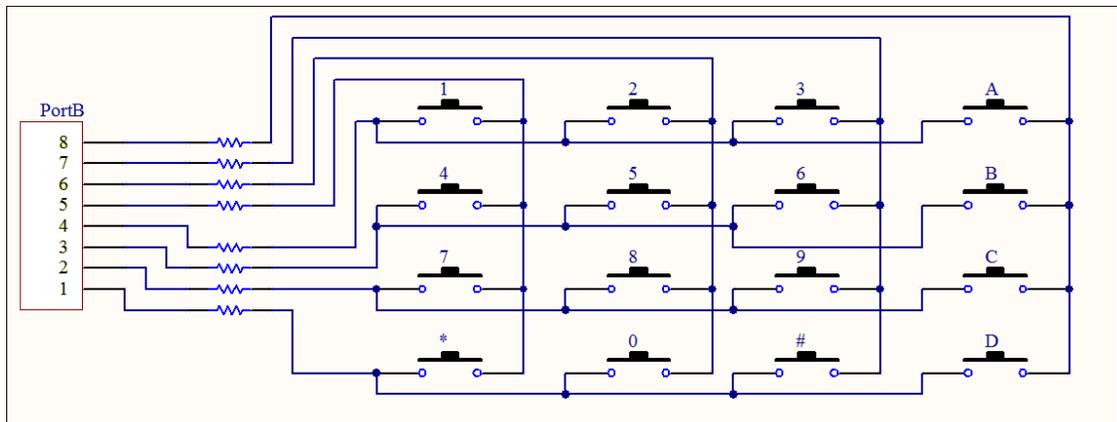
El interface del sistema con el usuario se realiza mediante un teclado matricial de 4x4 y un LCD matricial de 16x2, el primero está conectado al Puerto B del microcontrolador a través de 8 resistencias y el segundo conectado al Puerto D y a los pines 4 del puerto A, y 0 y 1 del puerto.

Figura 2.6. Esquema en la tarjeta del uC para los conectores de LCD y Teclado.



Un **teclado** matricial de 4x4 es constituido por 16 pulsantes distribuidos en 4 filas y 4 columnas conectados entre sí de tal manera que un punto de cada pulsante vaya hacia una línea de fila y el otro hacia una de columna (figura 2.7), permitiendo que el uC reconozca 16 teclas con tan solo 8 pines de control. En el uC se ejecuta un algoritmo para controlar el censado de las teclas, e identificar cual ha sido pulsada.

Figura 2.7. Teclado Matricial de 4x4.

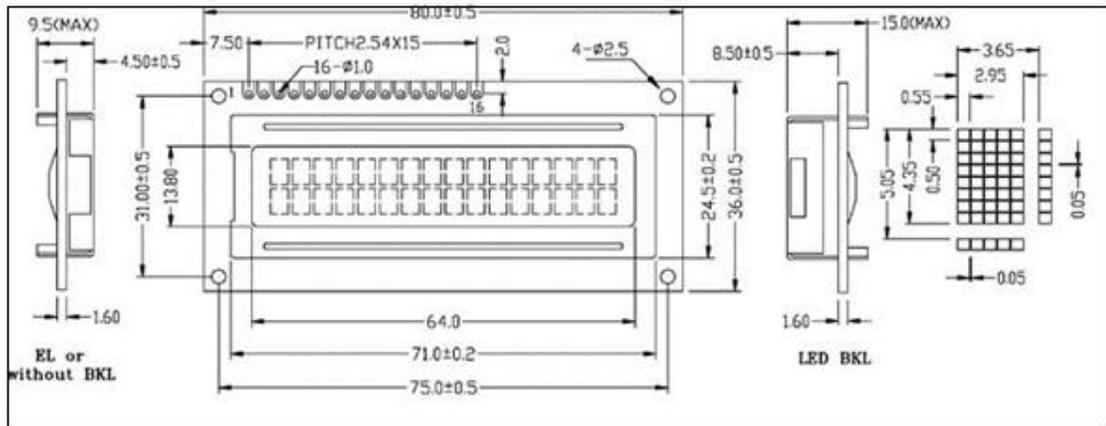


Una pantalla de cristal líquido o **LCD** está formada un arreglo de pixeles, que al ser polarizados absorben la luz en vez de reflejarla, haciendo aparecer en la pantalla un punto oscuro.

Una pantalla LCD matricial de 16x2 está constituida por 2 filas de 16 caracteres, cada carácter es de 5x8 pixeles con los que se puede simbolizar cualquier figura alfanumérica.

Figura 2.8. LCD matricial de 16x2.

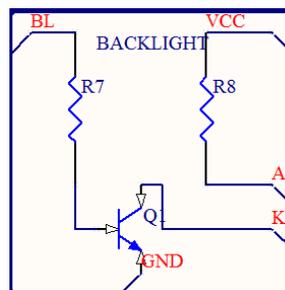
(Fuente: <http://www.planetaelectronico.com>)



El control del LCD se lo realiza mediante tres líneas llamadas RS, R/W y ENABLE y un puerto paralelo de 8 líneas de datos; la conexión de la pantalla LCD al uC es de la siguiente manera: RS se controla por PORTA<4>, R/W por PORTC<0>, ENABLE por PORTC<1> y el puerto de datos por todo PORTD. En la figura 2.6, se ve el esquema de conexión con el uC y el control de intensidad de iluminación de la pantalla.

La pantalla dispone de una fuente de luz LED y adiciona dos líneas para su control: Ánodo y Cátodo. El circuito de control de este LED se muestra en la figura 2.9.

Figura 2.9. Circuito de control del LED del LCD, BL se conecta a PORTA<5> del uC.



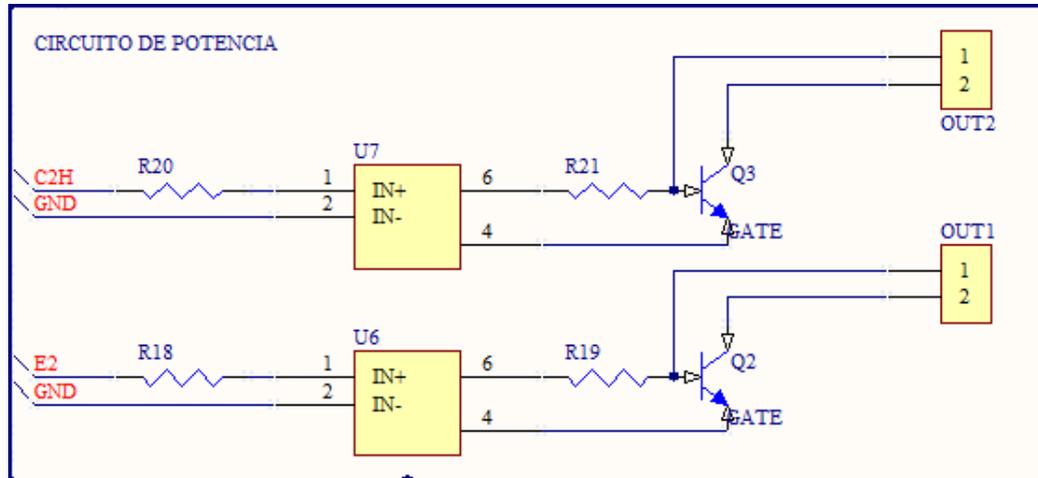
2.1.4. Opto acopladores

El motor de 220V AC será accionado por un relé de estado sólido (Capítulo 1, sección: Elementos de Potencia) aislado ópticamente de la etapa digital. Un opto acoplador es

un integrado que incluye un LED y un elemento fotosensible (U6 y U7 en la figura 2.10). La etapa digital que comanda al motor consiste en el pin PORTE<2> del microcontrolador. PORTC<2> se agregó para un uso futuro.

En la etapa de potencia los TRIAC Q2 y Q3, que son gobernados por los elementos fotosensibles de los opto acopladores, cerrarán o abrirán los circuitos de potencia.

Figura 2.10. Circuito de control del motor AC por PORTE<2>, más canal adicional.



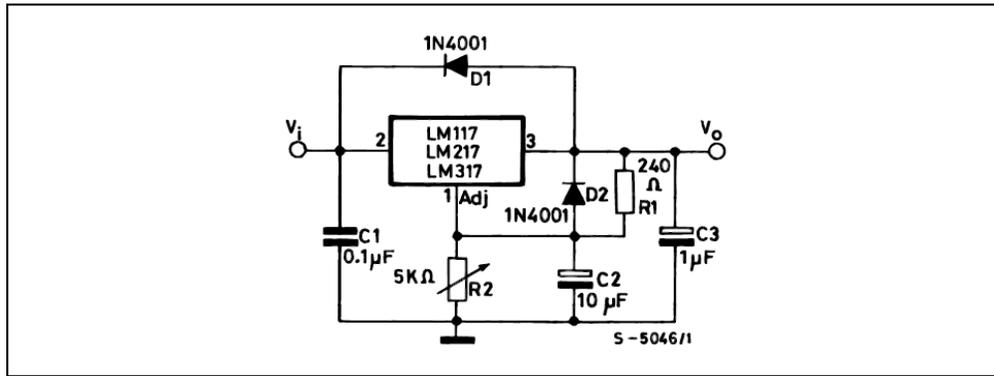
2.1.5. Fuente de alimentación para motor DC

La fuente de voltaje para el motor DC es obtenida de la toma de 220V utilizada para el motor AC. Esta fuente es de carácter variable.

Se utiliza un transformador de 220V AC a 18V AC y un circuito que realiza un proceso análogo al descrito en la sección ("Fuente de alimentación para el uC"), con la diferencia del regulador de voltaje, que en vez de ser fijo, varía según la posición que se le dé a un potenciómetro tipo TRIMMER.

Figura 2.11. Esquema usado en la tarjeta "Fuente Variable".

(Fuente: Hoja de especificación LM317)



2.1.6. Inversión de giro motor DC

El accionamiento del motor DC se realiza a través de uno de los dos circuitos Puente H que vienen en el encapsulado L298. Los objetivos del Puente H son: acoplar una etapa digital a un motor DC de mayor potencia y controlar la dirección de giro de este (Capítulo 1, sección: Elementos de Potencia).

El control del motor DC se realiza por los pines PORTE<0> y PORTE<1> del uC. En la figura 2.12 se ve el esquema usado en la placa.

Figura 2.12. Esquema usado Puente H L298.

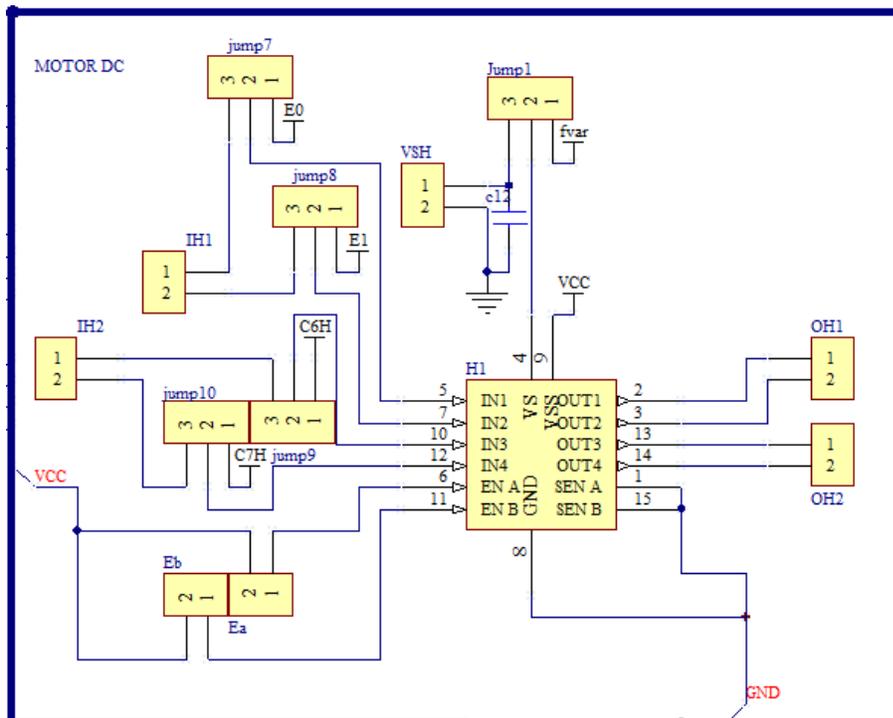
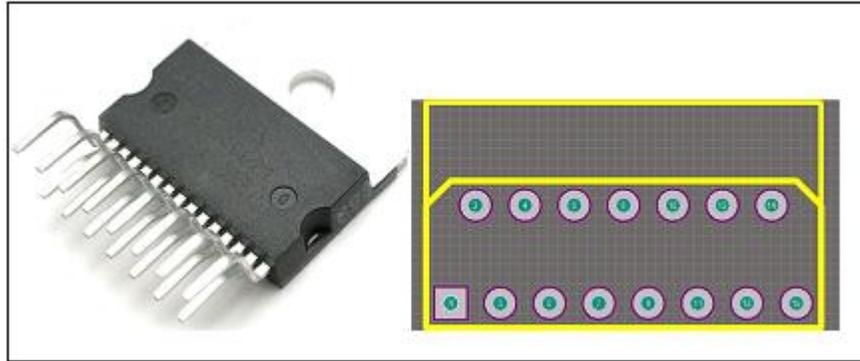


Figura 2.13. Encapsulado L298 y Huella en PCB.



2.2. Sistema de Adquisición de Datos

El módulo de adquisición de datos está construido en base a un microcontrolador PIC18F4553 de "Microchip Technology Inc." Este uC es análogo al PIC18F4550 excepto por la resolución del Convertidor Analógico a Digital, que en este caso es de 12 bits.

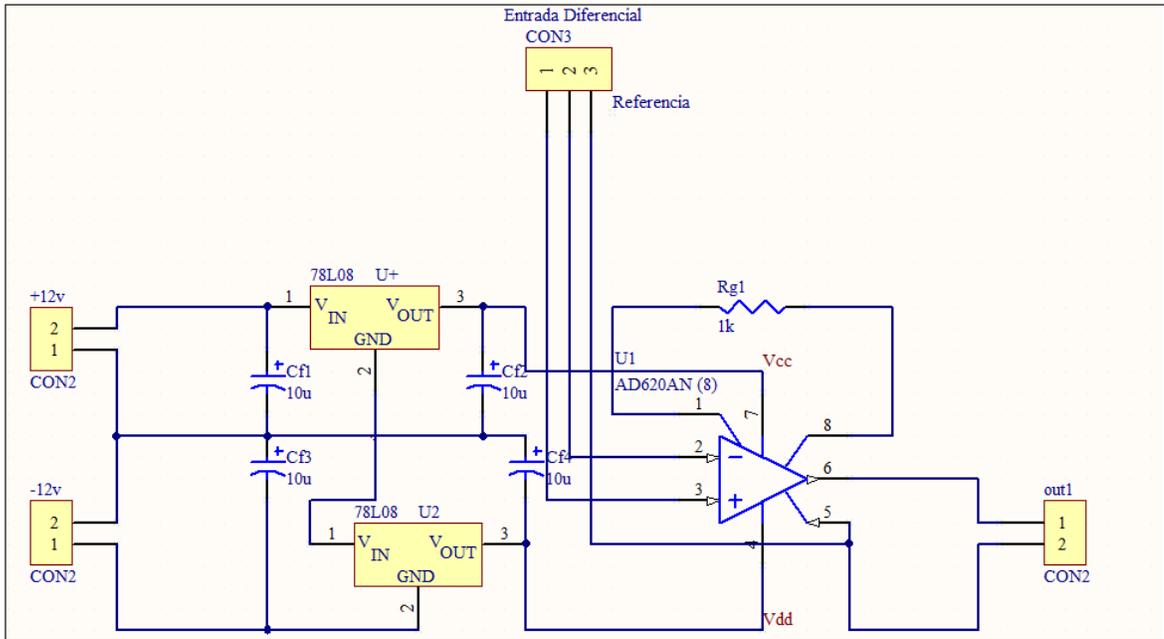
2.2.1. Acoplamiento de la señal

La señal obtenida de la máquina de espectrofotetría Perkin Elmer está entre 0 y 1v, por lo que, mediante un amplificador instrumental, es acondicionada al sistema de adquisición de datos.

Se utilizó un amplificador instrumental integrado AD620AN. Este, es un integrado basado en un amplificador instrumental clásico de tres operacionales, de bajo consumo y alta precisión, y cuya ganancia puede ser configurada entre 1 y 1000. El factor de rechazo al modo común está en el orden de los 100dB para ganancias igual a 10.

Al ser de bajo consumo y requerir una alimentación simétrica, se ha optado por alimentar el circuito mediante dos baterías de 9v que se conectan a las borneras +12v y -12v de la figura 2.14.

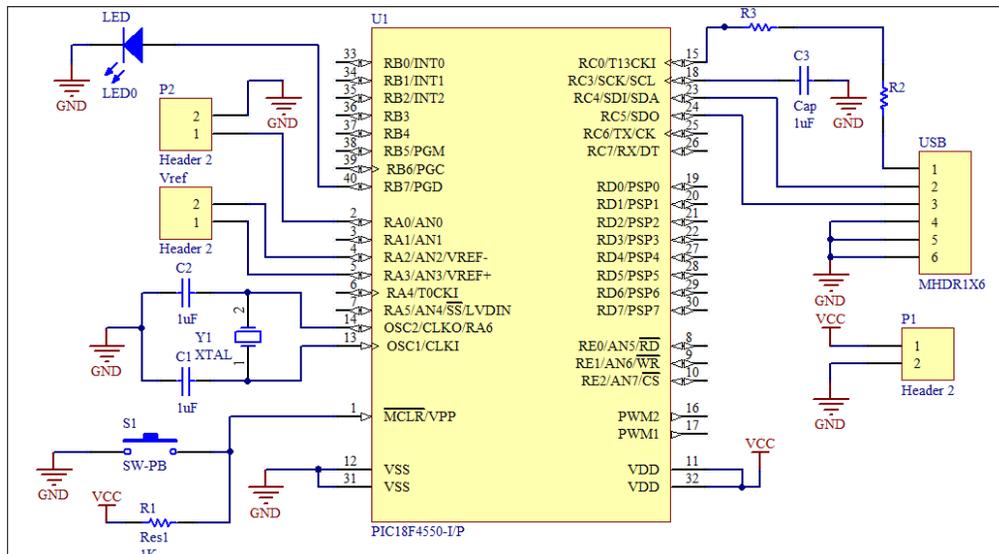
Figura 2.14. Amplificador instrumental basado en el integrado AD620AN.



2.2.2. Digitalización y transmisión de datos

Como se vio en (Capítulo 1, sección: Micro controladores modulo USB), el periférico USB del uC PIC18F4553 posee un transceptor, resistencias de Pull Up con las que el Host determina la velocidad del dispositivo y un regulador de voltaje de 3.3v internos que son activados por firmware, por lo que no es necesario introducir estos elementos en la tarjeta del dispositivo USB.

Figura 2.15. Esquema del módulo de adquisición de datos.

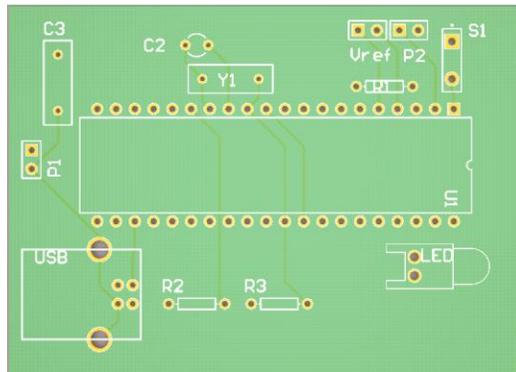


El módulo de adquisición de datos, al ser para una aplicación específica, consta de sólo un canal de adquisición analógico en PORTA<0>, y la posibilidad de establecer el voltaje de referencia para el convertidor A/D desde una fuente externa.

El módulo es un dispositivo de alimentación propia ("Self Power") mediante un adaptador externo. El voltaje que proviene del bus puede ser censado por PORTC<0>.

El conector USB usado es hembra tipo B para dispositivo.

Figura 2.16. PCB del módulo de adquisición de datos.



CAPITULO 3

FIRMWARE

3.1. Automatización de la máquina de muestreo

El firmware para el uC PIC18F4550, encargado del control de la máquina de muestreo, fue realizado en lenguaje ensamblador, en el entorno de desarrollo integrado "MPLAB IDE" proporcionado por "Microchip Technology Inc." para sus productos.

El uC ejecuta la activación o desactivación de los motores de manera secuencial y temporizada, además controla una pantalla LCD de 16x2 y un teclado matricial para el menú de configuración e indicaciones del estado de la máquina.

El uC de la máquina de muestreo trabaja mediante interrupciones producidas por los temporizadores TMR0 y TMR2, varios puertos de entrada y salida para el control de motores, teclado y LCD y memoria no volátil tipo EEPROM.

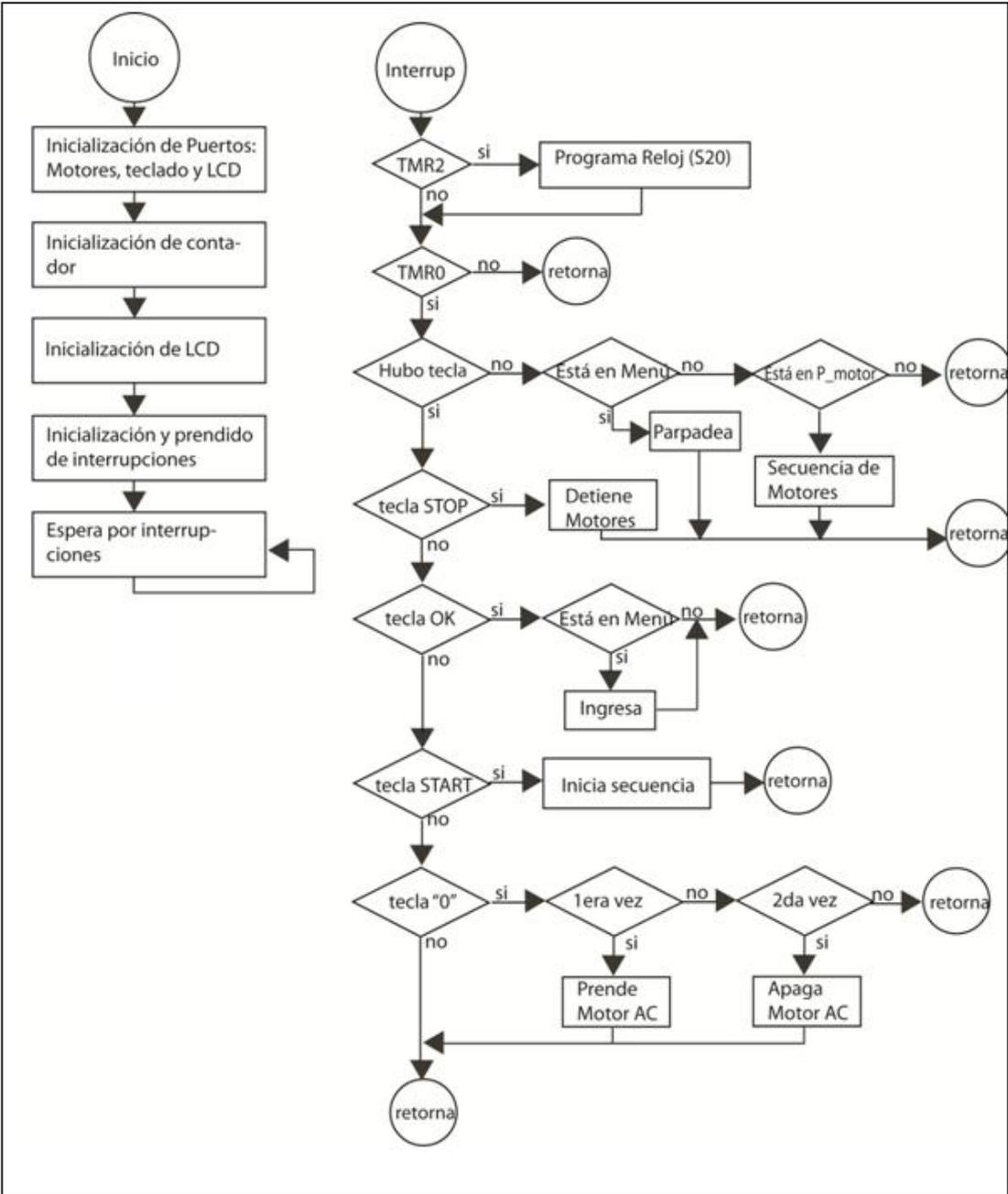
3.1.1. Inicio

Al encender el uC, los siguientes módulos son configurados:

- El puerto PORTE<0:2> como salidas, para accionar los motores y el puerto PORTA<0:2> como entradas, para los sensores de fin carrera.
- El puerto PORTB<0:3> como salida para activar las columnas y PORTB<4:7> para censar filas.
- Los puertos para LCD: PORTA<4:5> (RS y BackLight), PORTC<0:1> (R/W y E) y PORTD (Datos).
- La memoria EEPROM, además son leídos datos necesarios para "Contador".
- Temporizadores e Interrupciones (TMR0 es encendido pero no TMR2)

En la figura 3.1 se ve el diagrama de flujo principal que tiene el inicio hasta el bucle y las interrupciones.

Figura 3.1. Diagrama de flujo principal del firmware "Sampler 1.0".



3.1.2. Interrupciones

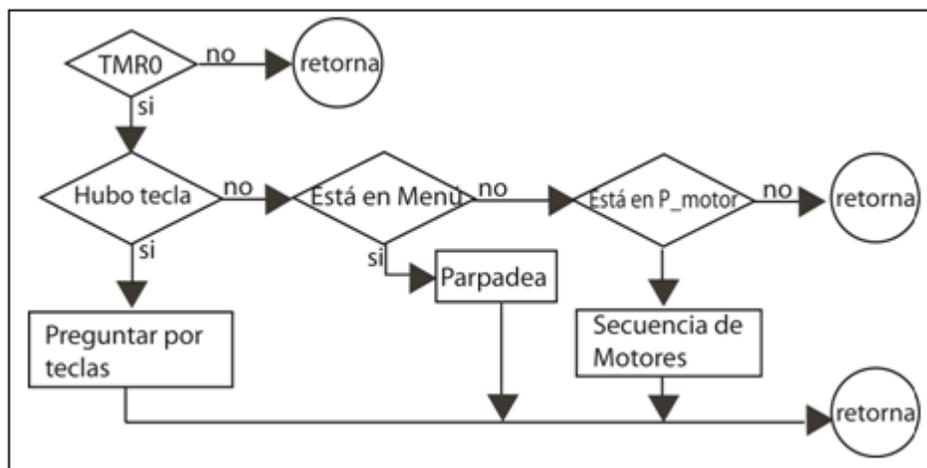
Las acciones que realiza el uC son siempre iniciadas por el operario; por lo que en cada interrupción por desborde de TMR0 se escanea el teclado.

El escaneo del teclado es de la siguiente manera: se alimenta la primera columna y se pregunta por cada una de las cuatro filas, si ninguna de esas teclas fue presionada, la

siguiente columna es alimentada y se repite el proceso. En cada pregunta con respuesta negativa, se incrementa una variable, esta tomará valores de 0 a 15, cada número representa una de las 16 teclas del teclado. Si ninguna tecla ha sido presionada una bandera indica que no hubo tecla.

Si por el contrario, alguna tecla ha sido presionada, se ejecuta una subrutina en donde se pregunta por las teclas a las que se les fueron asignadas ciertas funciones (por ejemplo: iniciar la secuencia de motores).

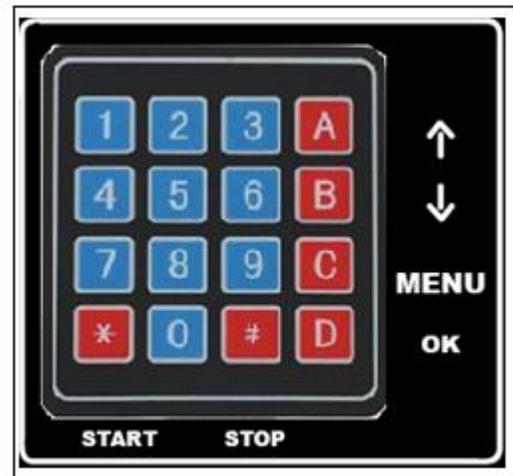
Figura 3.2. Diagrama de bloques, teclado en la Interrupción por desborde de TMR0



Las teclas con funciones asignadas y sus funciones son:

Tecla	Función
"*"	START: Da inicio a la secuencia de motores.
"#"	STOP: Detiene los motores.
"0"	Alimenta el motor AC (para saltar muestras).
"C"	MENU: Accede al panel de configuración
"D"	OK: Ingresa a la opción seleccionada
"A"	UP: Selecciona una opción superior
"B"	DOWN: Selecciona una opción inferior

El teclado matricial se ve en la figura 3.3.

Figura 3.3. Teclado matricial.

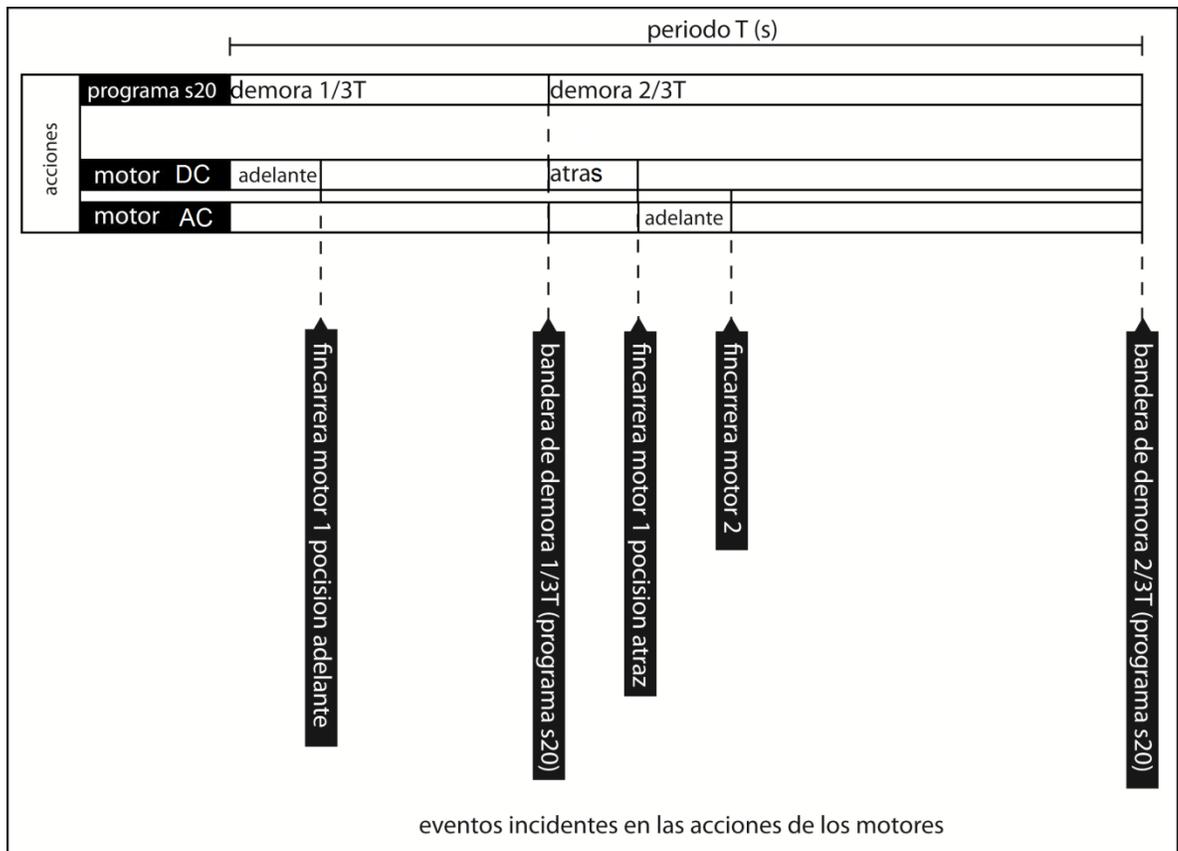
3.1.2.1. Ciclo Funcional

- Secuencia de Motores

Para que la máquina empiece a tomar muestras, el usuario debe presionar la tecla "START", en el uC, en ese momento, TMR2 es encendido, la interrupción es habilitada y se da inicio a la secuencia de motores (Capítulo 1 sección: Ciclo funcional del equipo).

Cuando empieza la secuencia, el motor DC es alimentado de tal manera que el terminal de absorción ingrese en la muestra (el motor va hacia adelante), un fin carrera detiene al motor; después de transcurrida una demora de tiempo, el terminal debe salir de la muestra, para ello el motor DC es alimentado de forma inversa y va hacia atrás, un segundo fin carrera detiene el motor. Cuando el motor DC para, se alimenta el motor AC hasta que un tercer y último fin carrera lo detenga, después de transcurrida una demora igual al doble de la primera, el ciclo vuelve a empezar.

Figura 3.4. Secuencia temporizada de los motores.



Cuando el motor va hacia adelante, al inicio de la secuencia, se pregunta primero por el fin carrera que lo detiene (en $PORTA<1>$) y después por una bandera del programa "Reloj S20" (la función de esta bandera se explica más adelante). Sin importar el resultado del fin carrera, el programa retorna de la interrupción al bucle principal; sin embargo, queda establecido un punto de ruptura para que de ahora en adelante, en cada interrupción provocada por TMR0, se ejecute la subrutina desde este punto y se encueste nuevamente por el fin carrera y por la bandera de "Reloj S20"; cuando el fin carrera es activado los motores se detienen; cuando la bandera de "Reloj S20" se ha encendido, el punto de ruptura es borrado y la secuencia continua.

Una vez encendida la bandera de "Reloj S20", se definen los valores de $PORTE<0:2>$ para que el motor DC invierta su giro y retorne a su posición inicial, y se pregunta por fin carrera asociado a $PORTA<0>$, si este fin carrera no ha sido activado, el programa retorna de la interrupción al bucle principal y se establece este punto como punto de

ruptura para que en cada desborde de TMR0 se pregunte nuevamente por el fin carrera mencionado.

Cuando se activa el fin carrera que detiene al motor DC, el punto de ruptura es borrado y se configura los valores de PORTE<0:2> para alimentar al motor AC. Del mismo modo, se establece aquí un punto de ruptura para que en cada interrupción de TMR0 se encueste por un último fin carrera conectado a PORTA<2>, y una segunda bandera de "Reloj S20"; el fin carrera detiene los motores, y la bandera inicia la secuencia nuevamente.

- Programa "Reloj S20"

El programa " Reloj S20" se ejecuta por interrupción debida al desborde de TMR2. Cuando la tecla "START" es presionada, TMR2 empieza a correr y las interrupciones son habilitadas, a partir de ese momento, cada vez que exista una interrupción se encuesta por si se desbordó TMR2 y corre "Reloj S20".

La función de "S20" es controlar con exactitud el tiempo en que el motor DC permanece en una posición o en otra. TMR2 se ha configurado para que se desborde cada 0.016 segundos, "Reloj S20" puede entonces, actuar al termino de un tiempo determinado.

"Reloj S20" trabaja con dos registros que por defecto tienen asignado los valores de 125 y 10, respectivamente. Cada vez que se ejecute "Reloj S20", el registro 2 se decrementa hasta llegar a 0, si el valor de este registro es diferente de 0 el programa retorna al bucle principal sin hacer nada; si el registro ha llegado a 0 han transcurrido 0.16 segundos desde que se han activado las interrupciones por desborde de TMR2.

Este proceso se repite el número de veces que indique el registro 1, por defecto 125 veces. Al termino de este tiempo se enciende la bandera que indica que ha transcurrido un tercio del periodo de la secuencia de motores (esta bandera deja pasar la secuencia de motores a su segunda etapa).

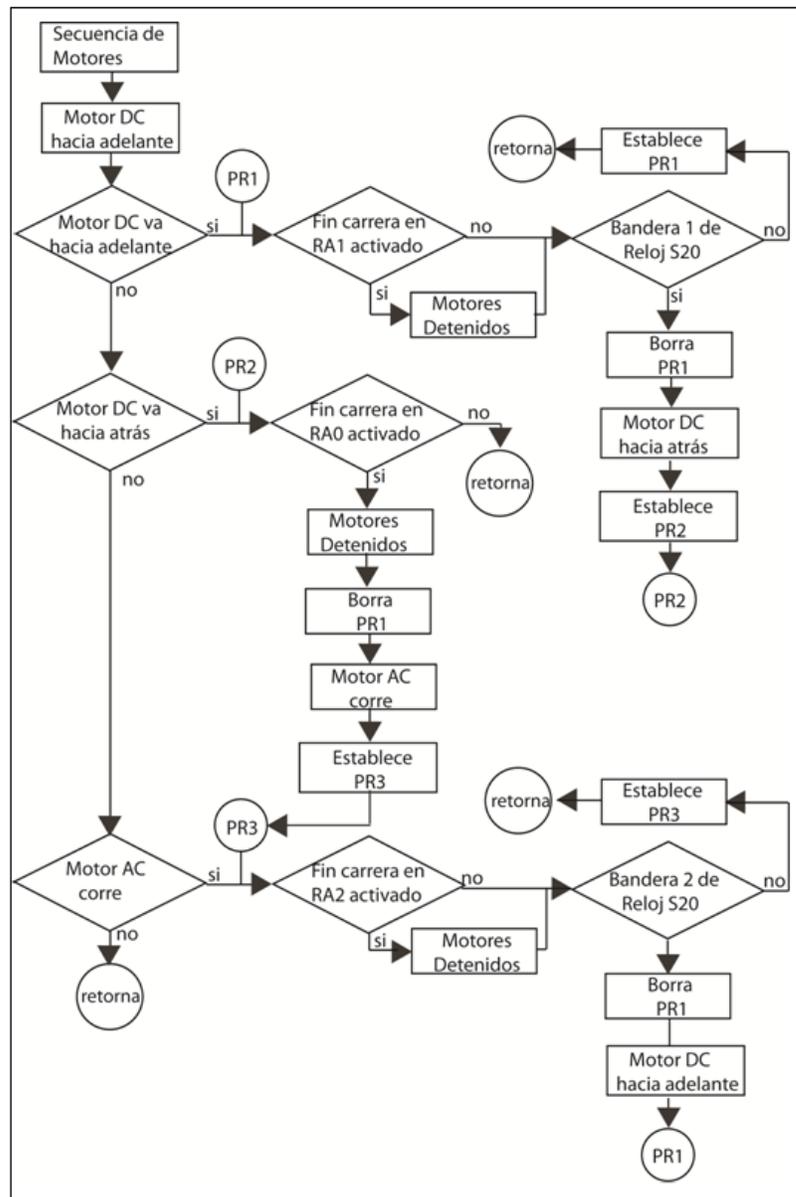
$$t = registro1 * registro2 * 0.016s$$

La ecuación 3.1 representa el algoritmo de demultiplexación de tiempo de "Reloj S20". Por defecto 20s ($125 \cdot 10 \cdot 0.016s = 20s$).

Una segunda bandera se enciende cuando este proceso se ha ejecutado dos veces más, indicando que el periodo de la secuencia ha concluido y que se debe iniciar nuevamente.

Los valores de los registros que son decrementados en "Reloj S20" son configurables con tres opciones de temporización.

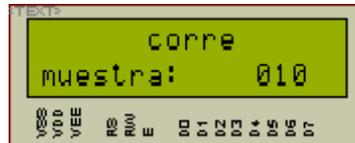
Figura 3.5. Diagrama de flujo de secuencia de motor



- Contador

El número de muestras tomadas por la máquina, son visualizadas en la pantalla LCD mientras la secuencia de motores se esté ejecutando. Cada vez que se termina de tomar una muestra (el motor DC va hacia atrás), se ejecuta el programa "Contador", que indica en pantalla el número de muestras tomadas en un número de tres dígitos.

Figura 3.6. LCD mientras se ejecuta la secuencia de motores. Simulación



Cada vez que se ejecuta el programa "Contador" se incrementa la variable "unidades". Si "unidades" ha llegado al valor de 10, se incrementa la variable "decenas" y "unidades" es nuevamente 0; del mismo modo si "decenas" llega a 10 se incrementa "centenas" y "decenas" se hace 0. En cada cambio de variable se imprime el dígito en la dirección que le corresponda en el LCD.

Estas tres variables son escritas en la memoria EEPROM del uC, para ser inicializadas en el último valor del contador cada vez que se encienda la máquina.

- Tecla "STOP" y "0"

Cuando se está ejecutando la secuencia de motores, el teclado es inhabilitado, a excepción de la tecla "STOP", cuya función es detener la secuencia de los motores y parar el temporizador TMR2 y sus interrupciones.

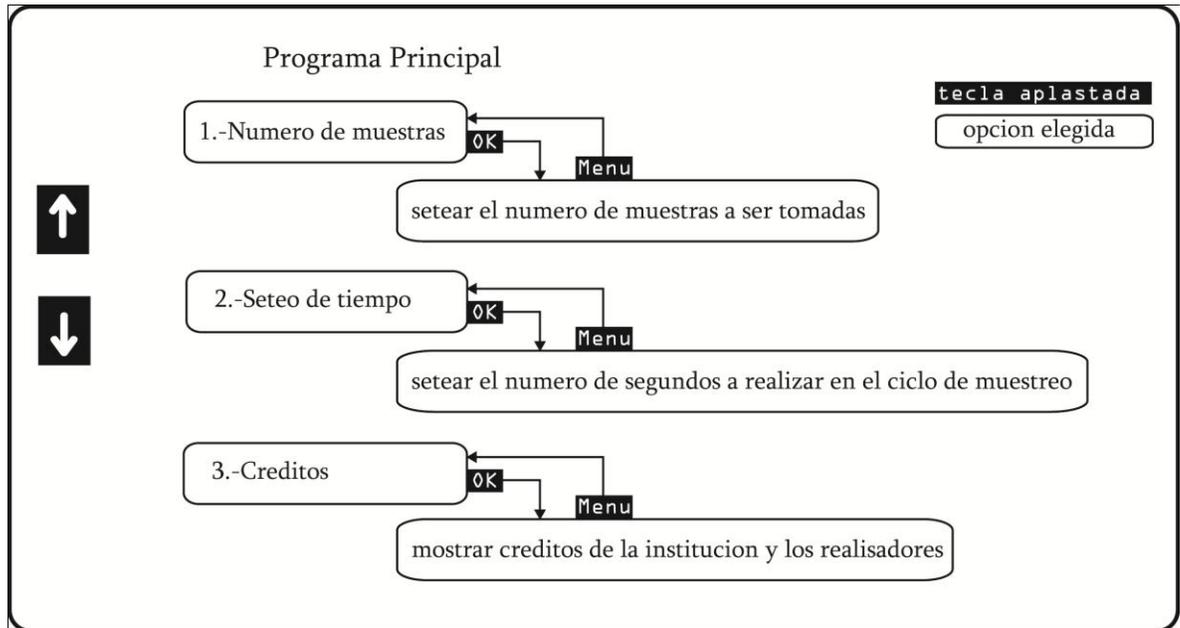
Cuando "STOP" es presionado, es necesario que el motor DC se encuentre en estado inicial para detener por completo la secuencia, se encuesta por el fin carrera correspondiente a dicho estado (En PORTA,1) y si no está activado se definen los valores de PORTE<0:2> para que el motor sea alimentado de forma que regrese a su posición inicial, activando así el fin carrera.

En estado "STOP", la tecla "0" activa al motor AC, sin importar la secuencia o que sea detenido por un fin carrera. Para desactivar el motor se ha de pulsar nuevamente "0".

3.1.2.2. Panel de Configuración

En estado "STOP", la tecla "MENU" permite el acceso al panel de configuración. Este panel está compuesto por las opciones que se ven en la figura 3.7.

Figura 3.7. Diagrama del Menú o Panel de Configuración.



Al acceder por primera vez al panel de configuración, el LCD muestra la pantalla de la figura 3.8 y la variable "opción" es igual a 0 (valor de la variable por defecto. "opción" es usada para acceder a las diferentes opciones del menú), el programa retorna de la interrupción al bucle principal.

Figura 3.8. LCD primera pantalla del MENU. Simulación.



Para que el usuario sepa que opción está seleccionada, en cada interrupción por desbordamiento de TMR0 en la que no se presione ninguna tecla, una variable es incrementada, hasta llegar a 37, entonces la segunda fila de la pantalla es apagada; cuando la variable llegue a 64, se imprime nuevamente la segunda fila y la variable se

establece a su valor inicial 0. Así la opción seleccionada (1.Contador) parpadea con un periodo de:

$$P = 0.016s * (64) = 1.024s$$

TMR0 se desborda cada 0.016s.

La opción permanece apagada durante:

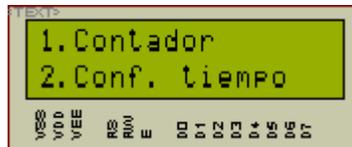
$$0.016s * (64 - 37) = 0.432s$$

y prendida durante:

$$0.016s * (37) = 0.592s$$

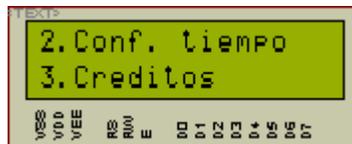
Para acceder a la segunda opción del panel de configuración (opción "Configurar Tiempo"), el operario debe presionar la tecla "DOWN"; el LCD mostrará la pantalla que se ve en la figura 3.9, la variable "opción" será igual a 1 y la opción seleccionada que parpadeará, será la de la segunda fila (2.Conf. tiempo).

Figura 3.9. LCD segunda pantalla del MENU. Simulación.



Si se presiona por segunda vez la tecla "DOWN", el LCD mostrará la tercera pantalla del panel y la variable "opción" será igual a 2. La opción seleccionada y que parpadea es la de la segunda fila (3.Creditos). Si se presiona nuevamente la tecla "DOWN", debido a que "opción" tomó su valor máximo (opción=2), el programa regresa de la interrupción sin hacer nada.

Figura 3.10. LCD tercera pantalla del MENU. Simulación.



La tecla "UP" al ser presionada, decrementa el valor de "opción" hasta un mínimo de 0 y selecciona la opción que se encuentre en la primera fila. Es decir si la pantalla

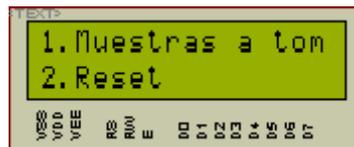
mostrada es la que se ve en la figura 3.10 y se ha presionado la tecla "UP", la variable "opción" será igual a 1 y la primera fila parpadeará (2.Conf. tiempo). Si la tecla "UP" es presionada nuevamente, "opción" será igual a 0 y la pantalla que se imprime es la de la figura 3.9, la primera fila parpadeará (1.Contador). Las próximas veces que la tecla "UP" sea presionada, se considera como si se ingresara por primera vez al panel de configuración.

La última tecla habilitada para el manejo del panel de configuración es la tecla "OK", que al ser presionada ingresa a una de las siguientes subrutinas:

- 1.Contador, si "opción" es igual a 0
- 2. Conf. tiempo, si "opción" es igual a 1
- 3. Créditos, si "opción" es igual a 2

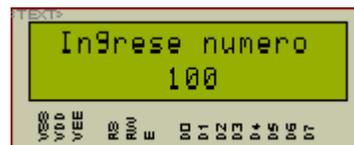
El submenú "Contador", tiene dos opciones que se muestran en el LCD (figura 3.11), y su selección e ingreso es análogo al funcionamiento del menú principal.

Figura 3.11. Submenú "Contador". Simulación.



La primer opción, "Muestras a Tomar", pide al usuario que ingrese un número de máximo tres dígitos para que sea este el número de muestras que la máquina tomará. Una vez aceptado, se reinicia el contador. Cuando la secuencia de los motores se ejecute, cada vez que "Contador" incremente su valor, este se compara con el número previamente establecido, la secuencia se detendrá cuando estos sean iguales.

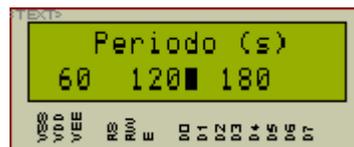
Figura 3.12. "Muestras a Tomar". Simulación.



La segunda opción del submenú contador (2.Reset), establece el valor del contador en 0.

Figura 3.13. El contador es reiniciado. Simulación.

El submenú "Configurar Tiempo", sirve para establecer el periodo de muestreo. Cuando el usuario ingresa en este submenú, puede escoger entre tres opciones de periodo marcadas como 60, 120 y 180; con las teclas "UP" y "DOWN", se imprime un carácter especial al lado de la opción seleccionada. Si el usuario presiona "OK", se aginan a los registros que se decrementan en el programa "RELOJ S20" (Sección: "Reloj S20") valores necesarios para que el periodo seleccionado sea establecido.

Figura 3.14. Selección del periodo. Simulación.

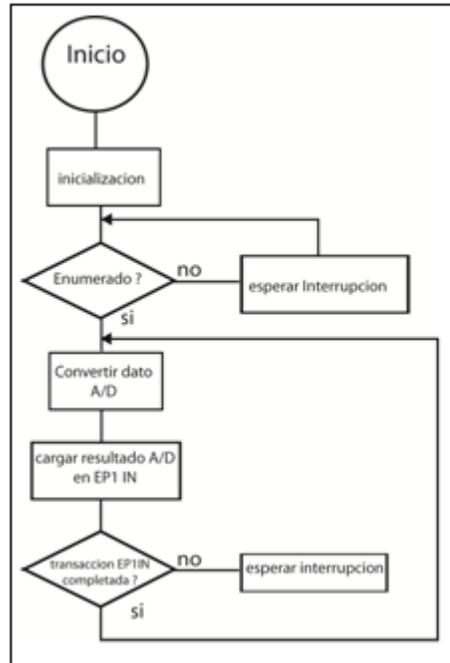
El submenú "Créditos" muestra los créditos del firmware.

3.2. Sistema de Adquisición de Datos USB

Con un uC PIC18F4553 se digitaliza una señal analógica de voltaje y se la transmite al ordenador a través del Puerto USB. El firmware se realizó en lenguaje Ensamblador.

La señal es digitalizada a 12 bits de resolución, y a una frecuencia de muestreo establecida por el ordenador.

El firmware está escrito de tal manera que la comunicación USB se realiza por interrupciones, mientras que la conversión de la señal analógica es hecha en el lazo principal del programa una vez que el dispositivo esté enumerado. El diagrama de flujo del bloque principal del firmware se muestra en la figura 3.15

Figura 3.15. Diagrama de bloques general del Sistema de Adquisición.

3.2.1. Conversión de señal analógica

La digitalización de la señal es realizada por el módulo convertidor A/D de 12 bits de resolución. El tiempo de conversión es menor a 100us.

El módulo está configurado para adquirir datos en un solo canal. El voltaje de referencia es tomado desde una fuente externa.

La etapa de conversión se la realiza en el bucle principal del programa, después que el dispositivo sea conectado y enumerado en el ordenador. En esta etapa, se realiza la conversión del nivel de voltaje analógico en el canal de adquisición y se coloca el resultado en dos bytes de EP1 IN para que sean extraídos por el ordenador. La siguiente conversión es realizada después de que sean enviados los datos.

3.2.2. Comunicación USB

Son múltiples los eventos ocurridos en el módulo USB que producen interrupciones, lo que hace posible que la comunicación con el ordenador sea realizada con un sistema de atención a interrupciones, estos eventos son: inicio de trama, transacción completa, actividad en el bus, errores y el reinicio del dispositivo por parte del Host.

Como la comunicación USB es basada en transferencias (Capítulo 1, sección: Comunicación en el Bus), el módulo permite que las instancias de menor nivel sean transparentes al programador. Las transferencias se clasifican en 2 grupos:

- **Transferencias de control:** manejan datos relacionados a la configuración y enumeración del dispositivo, estas son gestionadas a través de EP0 IN y EP0 OUT.
- **Transferencias de datos:** manejan datos de función del dispositivo, en este caso son los valores obtenidos del convertidor A/D.

La comunicación USB consta de 3 etapas: la inicialización del módulo USB, la enumeración del dispositivo y la transferencia de datos.

3.2.2.1. Inicialización del módulo USB

Previo a la comunicación con el ordenador, se configura el módulo y los EP involucrados de la siguiente manera:

- El módulo USB trabaja a velocidad "FULL SPEED", con transceptor y resistencias de "PULL UP" internas. Las interrupciones por USB son activadas.
- EP0 es usado para transacciones de control; su tamaño es de 72 bytes: 8 bytes configurados como OUT y 64 como IN (tomar en cuenta que la direccionalidad siempre es vista desde el lado del "Host")
- EP1 es usado para transferencia de datos; su tamaño es 4 bytes: 2 bytes como OUT y 2 como IN.

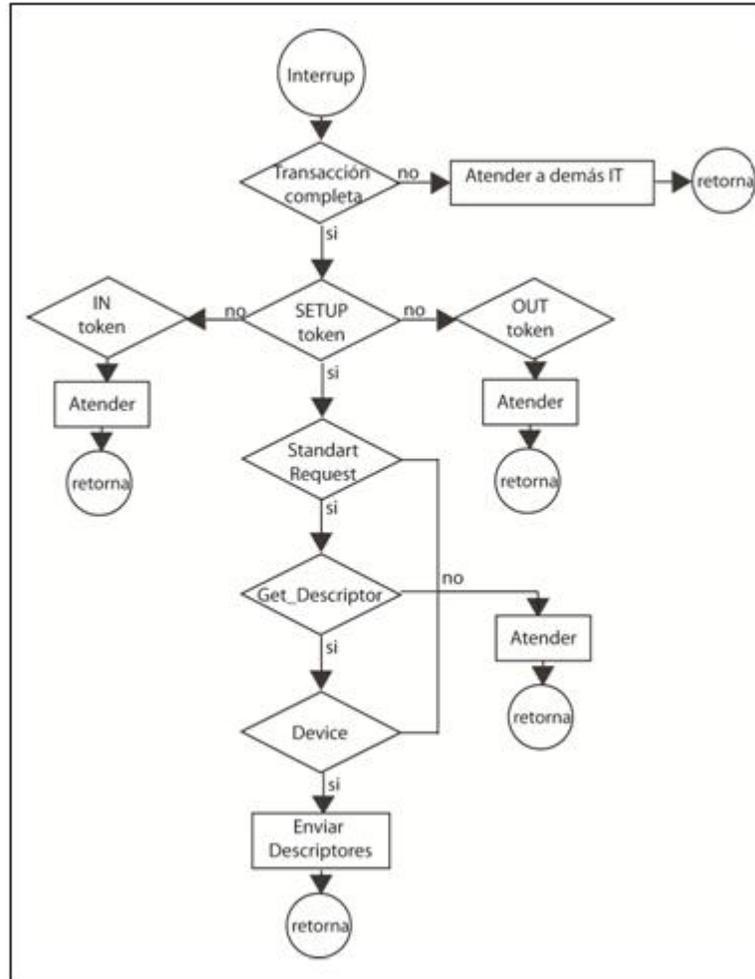
EP0 IN es de 64 bytes (máximo tamaño de transacción en "FULL SPEED") para que el dispositivo envíe descriptores al Host en una sola transacción. EP0 OUT es de 8 bytes, que es el tamaño de las peticiones o "Request" de control enviados por el Host. EP1 IN y EP1 OUT son de 2 bytes cada uno, este es el tamaño de los datos que se transferirán entre el host y el dispositivo.

3.2.2.2. Enumeración USB

El proceso de enumeración se realiza a través de transferencias de control, en donde el Host envía paquetes de datos llamados "Request" y el dispositivo, paquetes llamados "Descriptores".

En la figura 3.16 se ve la ruta del firmware cuando el Host a pedido recibir descriptores de dispositivo.

Figura 3.16. Diagrama de flujo, Envío de descriptores de dispositivo.



Un "Request" es un paquete de datos de 8 bytes de longitud dirigidos a EP0 OUT que contienen información acerca de los descriptores requeridos por el Host. En la tabla 3.1 se detallan los 8 bytes que contiene un Request.

Tabla 3.1 Detalle de los bytes dentro de un Request

(Fuente: Especificación USB 2.0)

Offset	Nombre	Tamaño (bytes)	Descripción
0	bmRequestType	1	Indica direccionalidad y si es solicitud estándar
1	bRequest	1	Especifica la solicitud
2	wValue	2	Depende de la solicitud
4	wIndex	2	Depende de la solicitud
6	wLength	2	Número de bytes a transferir

Un descriptor es una tabla de datos que se envían al host a través de una o varias transferencias de control después de haber recibido un Request.

El orden de los eventos en el proceso de enumeración visto desde un Host con sistema operativo Windows es el siguiente:

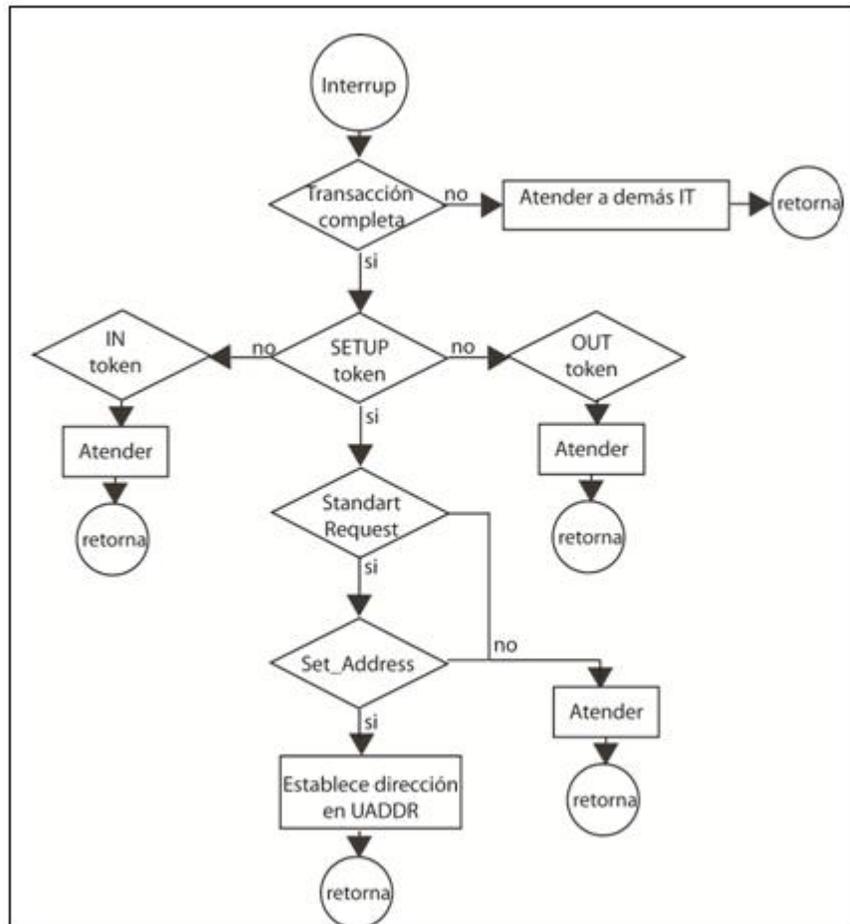
- 1) El Host detecta la conexión del dispositivo y espera 100ms para que establezca la corriente en el dispositivo.
- 2) El Host reinicia al dispositivo.
- 3) El Host solicita los bytes correspondientes a descriptores del dispositivo.
- 4) Después de recibir los primeros 8 bytes, el Host reinicia nuevamente el dispositivo.
- 5) El Host establece la dirección del dispositivo
- 6) El Host pide nuevamente los bytes de descriptores de dispositivo.
- 7) El Host pide los descriptores de configuración.
- 8) El Host pide los descriptores de cadena que se hubieran podido especificar.

Después que se ha estabilizado la corriente en el dispositivo y este es reiniciado, el Host solicita los descriptores de dispositivo.

Cuando el Host a tomado los primeros 8 bytes, este reinicia al dispositivo para luego tomar nuevamente todos los descriptores de dispositivo. En la interrupción por reinicio se inicializan todos los registros correspondientes a los EP que se describen en la etapa de inicio del dispositivo.

Antes de enviar todos los descriptores del dispositivo, el Host solicita establecer la dirección única que el dispositivo ocupará a lo largo de la comunicación. En la figura 3.17 se ve el camino del programa para establecer dicha dirección única. Cuando el Host envía una solicitud "SET_ADDRESS" envía además un byte de dirección que el dispositivo debe cargar en un registro de propósito específico llamado "UADDR".

Figura 3.17. Diagrama de flujo, Establecimiento de la dirección del dispositivo.



El dispositivo envía al Host descriptores indicando que responde a la versión 2.0 de USB, el tamaño de paquete máximo es 64 bytes, dispone de sólo una configuración, sólo una interface y dos EP. Por último envía descriptores tipo cadena que especifican

el nombre del fabricante y del dispositivo en código ASCII. Al finalizar este proceso el dispositivo queda enumerado.

3.2.2.3. Transmisión de datos al ordenador

La transmisión de datos empieza una vez que la conversión analógica haya sido realizada, en este punto, los 2 bytes del resultado de la conversión son cargados en EP1 hasta que sean tomados por el ordenador cuando este envíe una solicitud "In Token".

El formato de datos enviados al ordenador es de 2 números de 8 bits de longitud en donde el primer byte contiene los 4 bits menos significativos y el segundo los 8 más significativos. Este formato es acondicionado en la etapa de software. Capítulo 4 sección: (Acondicionamiento de la señal).

CAPITULO 4 SOFTWARE

4.1. Introducción

La señal obtenida de la máquina de espectrometría por absorción atómica por el dispositivo USB es procesada, visualizada y almacenada en un ordenador con sistema operativo "Microsoft Windows XP" en el entorno de desarrollo "LabView 2009 STUDENT EDITION" de la empresa "National Instruments".

4.2. Controladores

Una vez que Windows ha enumerado y establecido el Pipe, este buscará el controlador que haga funcionar al dispositivo dentro del sistema operativo.

Labview cuenta con una herramienta para comunicarse con buses de instrumentación llamada VISA. Driver Wizard. Este es un programa que es instalado en Windows junto con VISA y crea controladores para dispositivos USB, PXI/PCI y FireWire.

la guía para crear e instalar el controlador publicada por National Instruments se encuentra en la siguiente dirección: <http://zone.ni.com/devzone/cda/tut/p/id/9019>. Las especificaciones del dispositivo son:

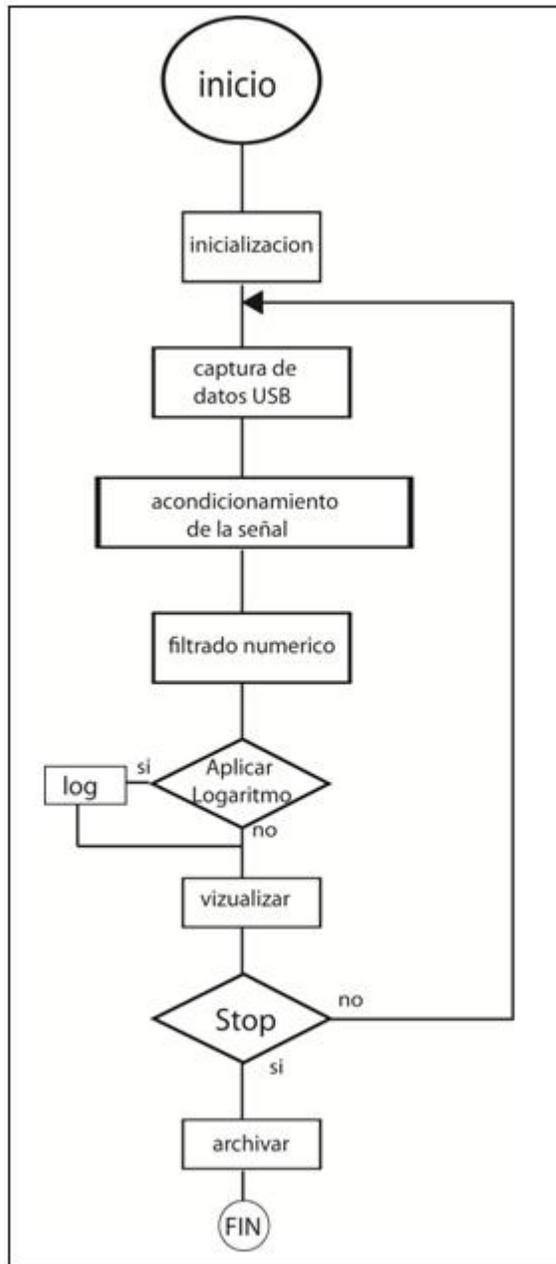
- El dispositivo usado no cumple con el protocolo USBTMC, por lo que se instalará como un dispositivo USB RAW.
- idVendor es 0x1234.
- idProduct es 0x0001.

4.3. Adquisición de Datos y procesamiento de la señal

El dispositivo USB

Los procesos que realiza el software se ilustra en la figura 4.1.

Figura 4.1. Diagrama de flujo del software desarrollado en Labview



4.3.1. Inicialización

Los parámetros de inicialización son:

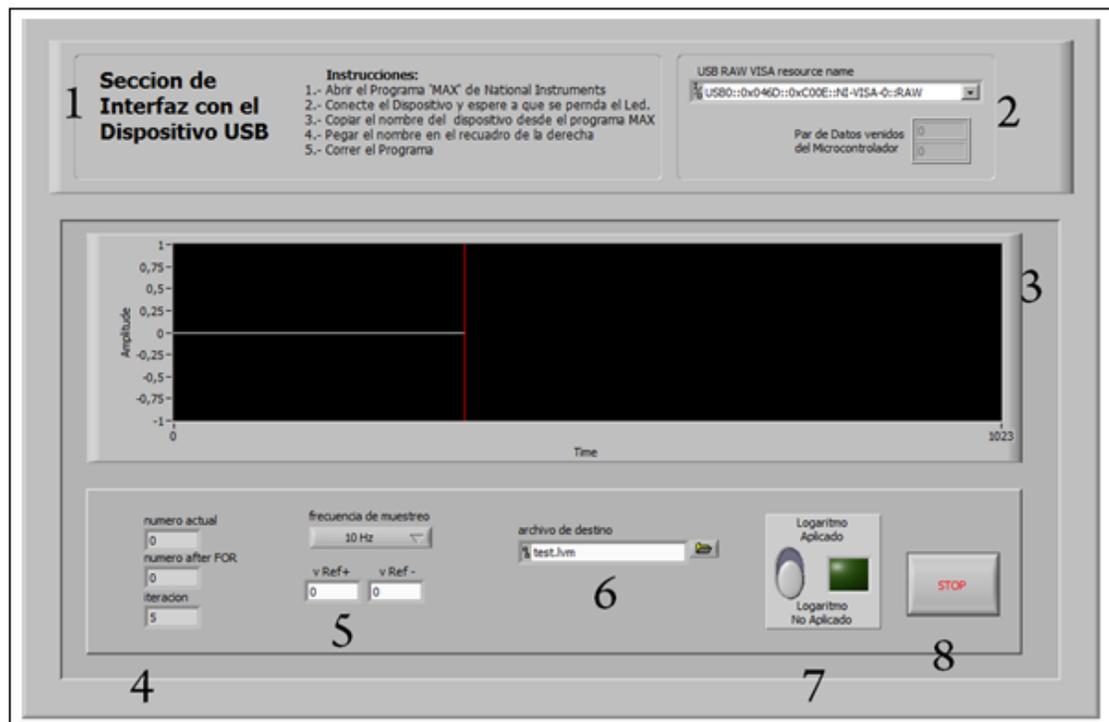
- **El tiempo de demora de cada iteración:** Es el tiempo en que se procesa un dato y está establecido en 10 ms. Es decir cada 10ms se obtiene y decodifica un dato del dispositivo.

- **La frecuencia de muestreo:** Frecuencia con la que se obtiene un dato válido después de ser filtrado, el usuario puede escoger los valores de 1 y 10hz en el panel frontal.
- **Voltaje de Referencia:** Vref+ y Vref- necesarios para el acondicionamiento de la señal, estos son establecidos por el operario en el panel frontal.
- **Recurso VISA RAW:** Nombre del dispositivo USB del cual se obtiene la señal; se lo toma del programa "MAX" de "National Instruments".
- **Nombre del archivo de destino:** Archivo donde se guardarán los datos después de ser captados por el programa.

4.3.2. Panel Frontal

Los elementos del panel frontal están distribuidos de la siguiente manera:

Figura 4.2. Panel frontal



1. Titulo e Instrucciones de conexión del dispositivo.
2. Recurso VISA RAW y par de datos adquiridos del uC. VISA requiere que se especifique el dispositivo USB RAW que trabaja.
3. Osciloscopio

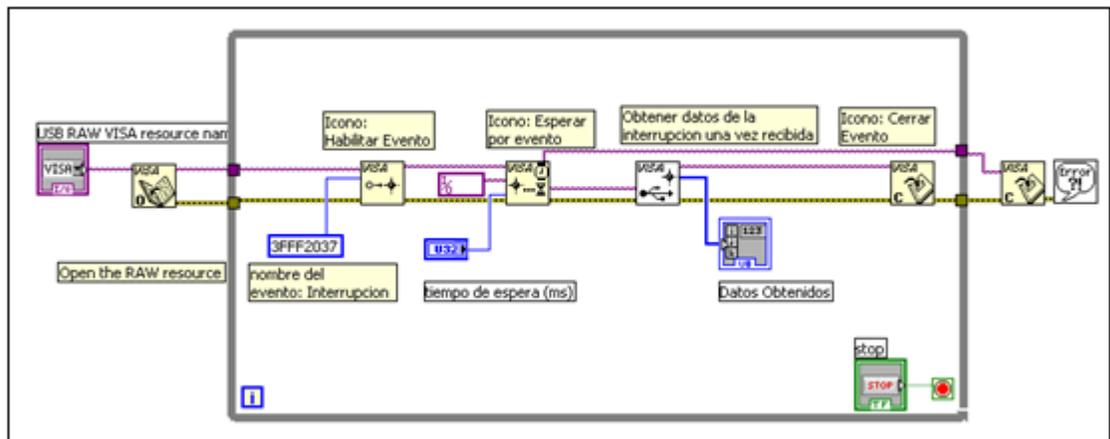
4. Indicadores de iteración, amplitud instantánea filtrada, y amplitud sin filtrar.
5. Frecuencia de Muestreo y Voltaje de Referencia
6. Archivo de destino.
7. Activador de visualización Logarítmica
8. Botón STOP

4.3.3. Diagrama de Bloques

4.3.3.1. Captura de datos USB

Los datos del dispositivo son obtenidos por el programa a través de los recursos VISA con los que cuenta "LabView", la figura 4.3 ilustra la secuencia de funcionamiento de VISA.

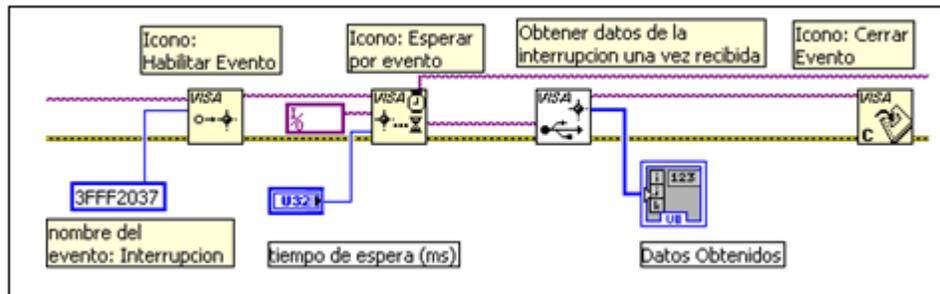
Figura 4.3. Obtención de datos a través de VISA. Diagrama de bloques.



La comunicación con VISA es abierta antes de ingresar al bucle de comunicación. En cada iteración, se habilita el evento de obtención de dato (en este caso, interrupción por USB), se lee el dato y se deshabilita el evento.

En la figura 4.4 se ve las actividades que se realiza dentro del bucle principal.

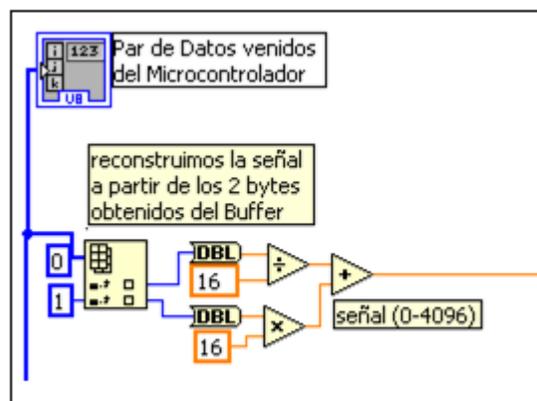
Figura 4.4. Obtención de datos a través de VISA en una iteración.



4.3.3.2. Acondicionamiento de la Señal

Después de obtener el dato con VISA, éste es colocado en un arreglo de dos bytes con los que se forma un número de 16 bits. Sin embargo, para la reconstrucción del número, hay que tomar en cuenta que la resolución del convertidor es de 12 bits; la señal está entre 0x000 y 0xFFF. El byte menos significativo tiene en su nibble más bajo los 4 bits que se deben eliminar, para eso, se divide al número para 0x010 (ó 16 en decimal), lo que en código binario significa rotar el byte 4 veces a la derecha; el byte menos significativo alcanza un valor máximo de 0x00F. Para que se puedan sumar los dos bytes, el byte más significativo se ha de multiplicar por el número más alto que puede tomar el byte menos significativo más uno (0x010 ó 16 en decimal).

Figura 4.5. Acondicionamiento de la señal.

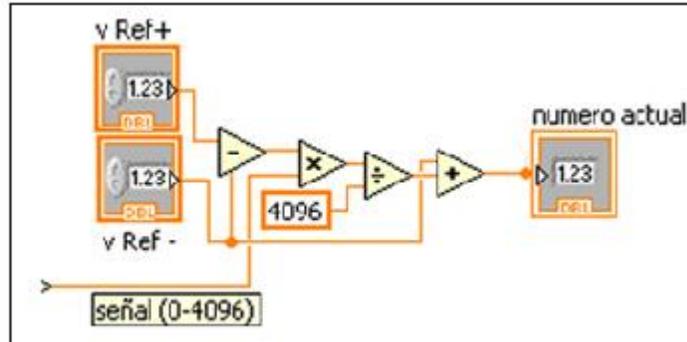


4.3.3.3. Ajuste de rango

Como se vio en el punto anterior, la señal toma valores de 0x000 a 0xFFF, sin embargo, el osciloscopio debe mostrar el valor del voltaje medido. Para adaptar esta

señal a los valores de voltaje establecido (entre los valores de Vref+ y Vref-), se multiplica al dato por el rango y se lo divide para la resolución.

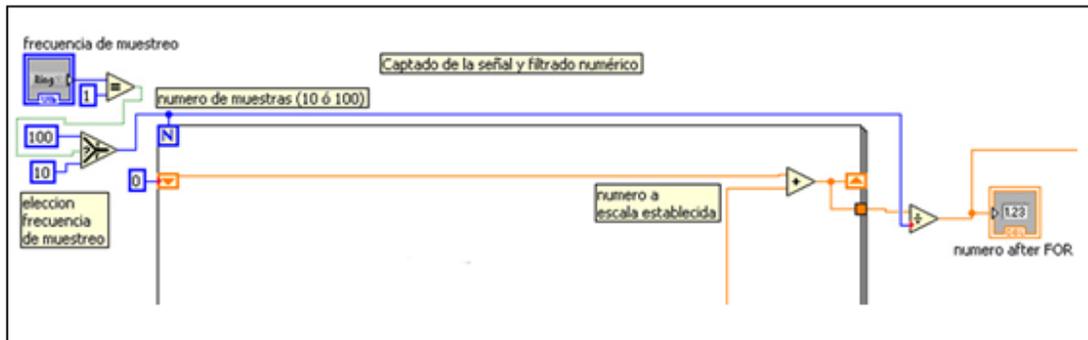
Figura 4.6. Ajuste de rango.



4.3.3.4. Filtrado numérico

Un dato es obtenido, acondicionado y ajustado al rango cada 10ms (Sección: Inicialización), la frecuencia de muestreo es entonces 100hz. El filtrado numérico es un promedio hecho cada 10 ó 100 muestras tomadas, según lo establezca el usuario. Una vez filtrado el número este es visualizado en el osciloscopio.

Figura 4.7. Filtrado numérico de la señal



La frecuencia de visualización es:

$$F = \frac{1}{10 * 10ms} = 10hz$$

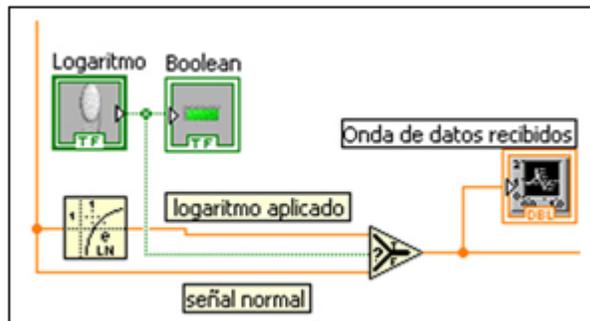
ó

$$F = \frac{1}{100 * 10ms} = 1hz$$

4.3.3.5. Logaritmo natural de la señal

Para que la señal sea visualizada en forma de logaritmo, se dispone de un control booleano en el panel frontal que habilita o deshabilita la función "Logaritmo Natural" en el diagrama de bloques.

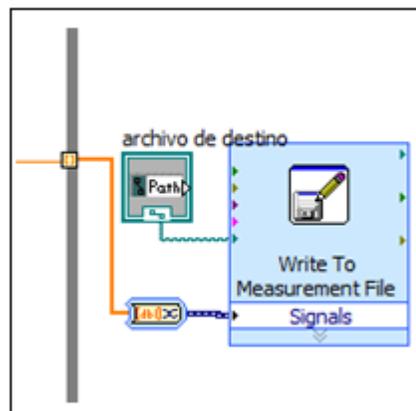
Figura 4.8. Aplicación de logaritmo natural a la señal



4.3.3.6. Indexado de datos

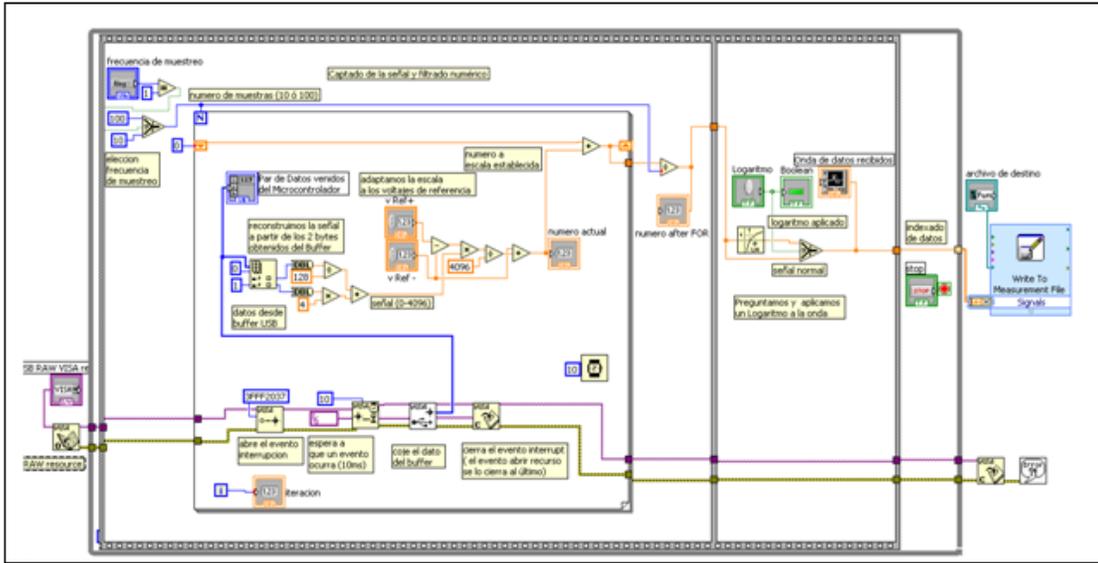
Cada dato filtrado, ya sea visualizado en forma de logaritmo o no, es colocado en un arreglo de datos, para ser guardados en un archivo de texto de extensión .lvm que puede ser leído por otras aplicaciones del entorno de National Instruments.

Figura 4.9. Indexado y archivado de datos



En la figura 4.10. se observan los procesos descritos a lo largo de este capítulo.

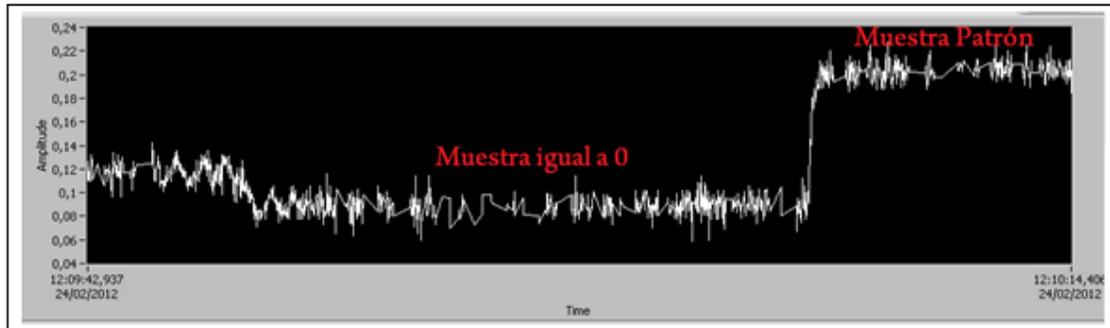
Figura 4.10: Diagrama de bloques completo.



RESULTADOS

En la figura 5.1 se observa la señal obtenida mediante un sistema de adquisición de datos NI USB-6009 de National Instruments sin ser amplificada ni filtrada.

Figura 5.1. Señal obtenida con un DAQ USB-6009 sin amplificar ni filtrar.



En la figura 5.2 se observa la señal amplificada por hardware y filtrada por software que ha sido obtenida por el sistema de adquisición de datos realizado en este trabajo.

Figura 5.2. Señal obtenida por el sistema de adquisición de datos.



Es absolutamente necesario amplificar y filtrar la señal para que el sistema de adquisición de datos de este trabajo funcione adecuadamente, caso contrario, el ruido es mayor que la señal útil de la máquina de espectrometría.

CONCLUSIONES

El análisis de soluciones mediante espectrofotometría por absorción atómica determina el soluto y su cantidad en cada muestra. Para que el laboratorista tenga conclusiones válidas, es indispensable el análisis de un determinado número de muestras.

La automatización en la toma de muestras simplifica la labor del operario, permitiendo el análisis de una onda continua en el tiempo, donde periódicamente se tiene datos que pertenecen a cada muestra, seguidos de datos de menor importancia debidos a la limpieza del área de análisis.

Una vez concluido el análisis de cada muestra realizado por el espectrofotómetro marca PerkinElmer, un sistema de adquisición de datos permite la visualización y procesamiento de la señal.

Las características de la máquina de muestreo y sistema de adquisición de datos desarrollados en este trabajo son:

- La máquina de muestreo responde correctamente a las exigencias presentadas en el laboratorio, desde el circuito digital, el interface de potencia y los actuadores.
- La secuencia y el recorrido de los motores de la máquina de muestreo son correctos.
- Los errores que se han dado en la etapa de pruebas son de carácter mecánico, y han sido solucionados calibrando los elementos móviles en la máquina.
- Para la visualización y procesamiento de la onda resultante, el sistema de adquisición de datos basado en un uC PIC18F4553 con 12 bits de resolución, cuenta con un sistema de comunicación moderno y eficiente como es USB 2.0, capaz de enviar datos al ordenador en tiempo real.

Los alcances del sistema son:

- Los tiempos de la máquina de muestreo pueden variar a un rango más idóneo para el laboratorio, pues actualmente se utilizan muestras llamadas 'blancos', donde en vez de analizar una muestra válida, se limpia la máquina para tomar una nueva muestra. Con otros tiempos de muestreo se pueden elevar la eficiencia del muestreo.

- El Sistema de adquisición de datos puede ser más genérico, tener más canales de adquisición e incluso generar ondas ordenadas desde el computador.
- El firmware del sistema de adquisición, puede ser modificado para responder al estándar USBTMC.

Se recomienda:

- Dar un mantenimiento mecánico a la máquina de muestreo, aceitar, revisar los componentes mecánicos cada cuatro meses.
- Leer completamente el Manual de usuario adjunto a este documento antes de usar el sistema.

BIBLIOGRAFÍA

REFERENCIAS BIBLIOGRÁFICAS:

1. BERNHARD, Welz; SPERLING, Michael. *Atomic Absorption Spectrometry*. Third Completely Revised Edition. Germany: WILEY-VCH Verlag, 2005. Impreso
2. *OPTOISOLATORS TRIAC DRIVER OUTPUT*. Fairchild Semiconductor Corporation, 2002. Digital.
3. PÉREZ RODRÍGUEZ, Leonel. *Interface USB en la IBM_PC*. Ecuador: Escuela de Ingeniería Electrónica. Universidad del Azuay. 2008. Digital.
4. PÉREZ RODRÍGUEZ, Leonel. *Introducción a los micro controladores*. Ecuador: Escuela de Ingeniería Electrónica. Universidad del Azuay. 2008. Digital.
5. PÉREZ RODRÍGUEZ, Leonel. *Convertor Analógico/Digital*. Ecuador: Escuela de Ingeniería Electrónica. Universidad del Azuay. 2009. Digital.
6. *PIC18F2455/2550/4455/4550 Data Sheet*. Microchip Technology Inc, 2009. Digital.
7. *PIC18F2458/2553/4458/4553 Data Sheet*. Microchip Technology Inc, 2009. Digital.
8. RASHID, Muhammad H. *Electrónica de Potencia*. Tercera edición. México: PEARSON Educación de México, 2004. Impreso.
9. RODRÍGUEZ PENIN, Antonio. *Sistemas SCADA*. Segunda edición. México: MACROMBO Ediciones Técnicas, 2007. Impreso.

REFERENCIAS ELECTRÓNICAS:

1. Compaq; HP; Intel; Lucent; Microsoft; NEC (2000); Philips "*Universal Serial Bus Specification Revision 2.0*". [En línea]. Disponible en: <http://www.usb.org> [Accesado el día 10 de Septiembre del 2011].
2. National Instruments (2009). "*Guía sobre Control de Instrumentos por USB*." [En línea]. Disponible en: <http://www.ni.com/white-paper/4478/es>. [Accesado el día 7 de Enero del 2012].

ANEXO 1

SAMPLER 1.0

(Guía de Usuario)

1. Encendido

Para encender la máquina de muestreo, conectar los dos tomas de corriente a la fuente de poder respectiva y poner el interruptor en la posición ON.

2. Inicio

Una vez encendido el aparato aparece en la pantalla de inicio en el LCD. En la figura 1 se ve la pantalla de inicio.

Figura 1. Pantalla de Inicio.

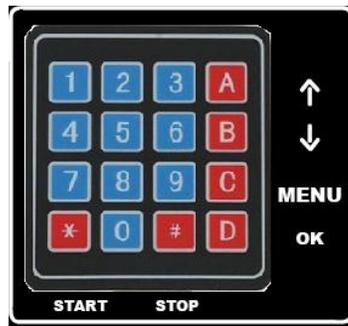


3. Operación

Las teclas con funciones asignadas y sus funciones son:

Tecla	Función
"*"	START: Da inicio a la secuencia de motores.
"#"	STOP: Detiene los motores.
"0"	Alimenta el motor AC (para saltar muestras).
"C"	MENU: Accede al panel de configuración
"D"	OK: Ingresa a la opción seleccionada
"A"	UP: Selecciona una opción superior
"B"	DOWN: Selecciona una opción inferior

Figura 2. Teclado Matricial.



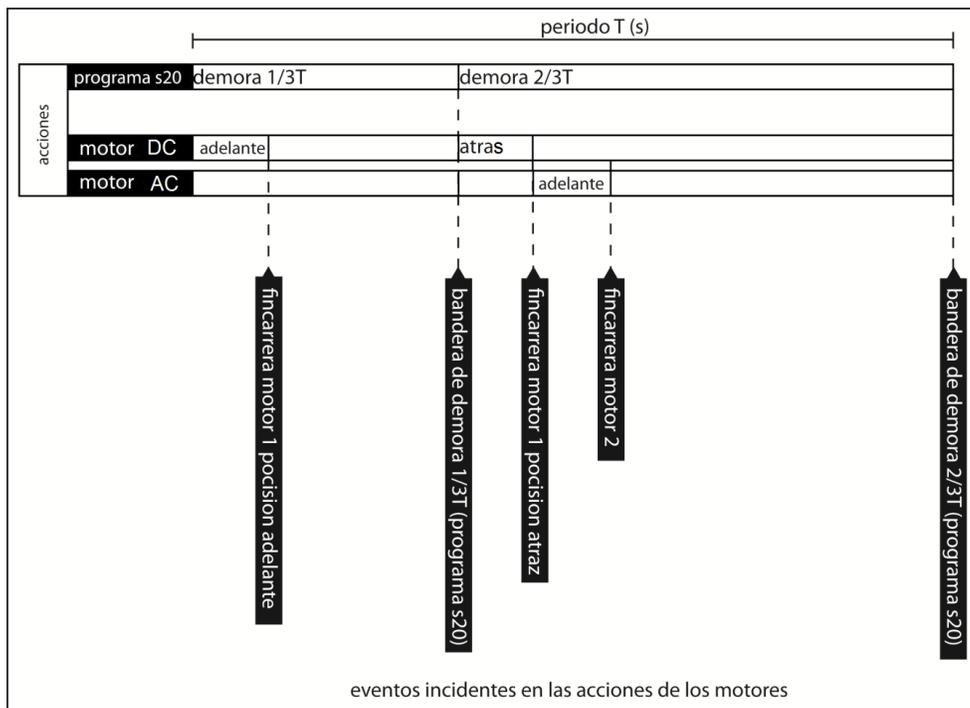
Tecla START. Ciclo Funcional

- Secuencia de Motores

Para que la máquina empiece a tomar muestras, el usuario debe presionar la tecla "START",

Cuando empieza la secuencia, el motor DC es alimentado de tal manera que el terminal de absorción ingrese en la muestra (el motor va hacia adelante), un fin carrera detiene al motor; después de transcurrida una demora de tiempo, el terminal debe salir de la muestra, un segundo fin carrera detiene el motor. Cuando el motor DC para, se alimenta el motor AC hasta que un tercer y último fin carrera lo detenga, después de transcurrida una demora igual al doble de la primera, el ciclo vuelve a empezar.

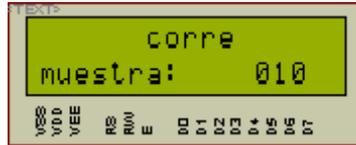
Figura 3. Secuencia temporizada de los motores.



- Contador

La cantidad de muestras tomadas por la máquina, son visualizadas en un número de tres dígitos en la pantalla LCD mientras la secuencia de motores se esté ejecutando.

Figura 4. LCD mientras se ejecuta la secuencia de motores. Simulación



Tecla "STOP" y "0"

Cuando se está ejecutando la secuencia de motores, el teclado es inhabilitado, a excepción de la tecla "STOP", cuya función es detener la secuencia de los motores.

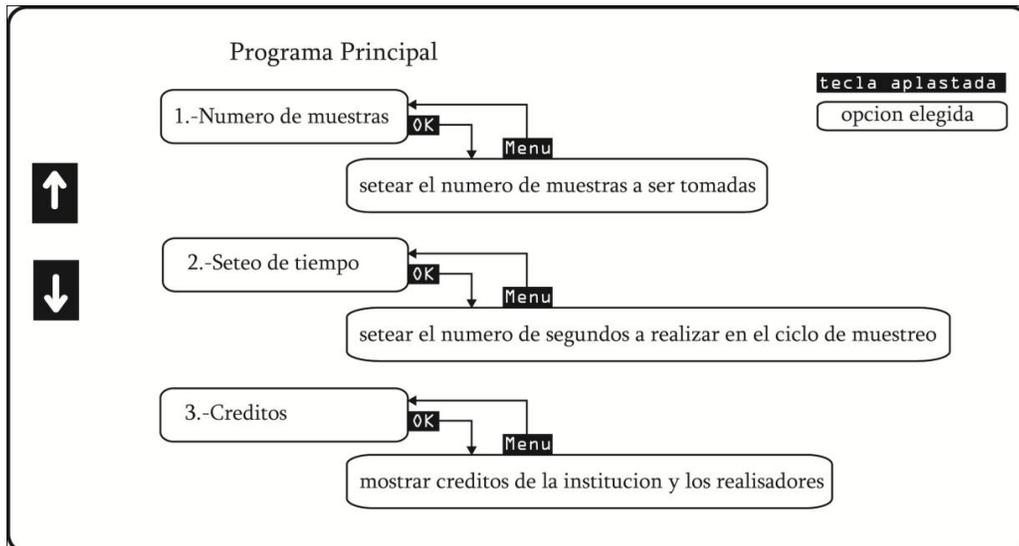
Cuando "STOP" es presionado. El motor AC es detenido y el motor DC va hacia su posición inicial antes de parar.

En estado "STOP", la tecla "0" activa al motor AC, sin importar la secuencia o que sea detenido por un fin carrera. Para desactivar el motor se ha de pulsar nuevamente "0".

Tecla Menú. Panel de Configuración

En estado "STOP", la tecla "MENU" permite el acceso al panel de configuración. Este panel está compuesto por las opciones que se ven en la figura 5.

Figura 5. Diagrama del Menú o Panel de Configuración.



Al acceder por primera vez al panel de configuración, el LCD muestra la pantalla de la figura 5.

Figura 6. LCD primera pantalla del MENU. Simulación.



Teclas OK, UP y DOWN

Con las teclas UP y DOWN, se selecciona una de las tres opciones que dispone el panel de configuración. La opción señalada parpadea.

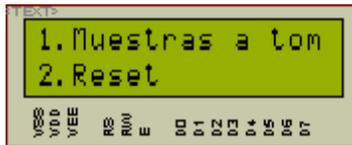
La tecla OK permite el ingreso a cada opción.

- 1. Contador, si "opción" es igual a 0
- 2. Conf. tiempo, si "opción" es igual a 1
- 3. Créditos, si "opción" es igual a 2

1. Contador

El submenú "Contador", tiene dos opciones que se muestran en el LCD (figura 7), y su selección e ingreso es análogo al funcionamiento del menú principal.

Figura 7. Submenú "Contador". Simulación.



La primer opción, "Muestras a Tomar", pide al usuario que ingrese un número de máximo tres dígitos para que sea este el número de muestras que la máquina tomará. Una vez aceptado, se reinicia el contador. Cuando la secuencia de los motores se ejecute, se detendrá cuando el contador llegue al número definido en esta opción.

Figura 8. Submenú "Contador". Simulación.



La segunda opción del submenú contador (2.Reset), establece el valor del contador en 0.

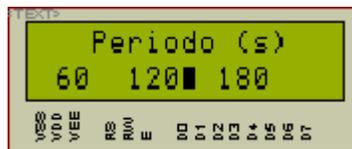
Figura 9. Reinicio de contador. Simulación.



2. Configurar Tiempo

El submenú "Configurar Tiempo", sirve para establecer el periodo de muestreo. Cuando el usuario ingresa en este submenú, puede escoger entre tres opciones de periodo marcadas como 60, 120 y 180; con las teclas "UP" y "DOWN", se imprime un carácter especial al lado de la opción seleccionada.

Figura 10. Selección del periodo. Simulación.



3. Créditos

El submenú "Créditos" muestra los créditos del firmware.

4. Notas

La secuencia de motores no inicia si el programa está en el panel de configuración; para salir del mismo, presionar la tecla MENU.

5. Apagado de la máquina

El apagado de la máquina se debe realizar con los motores detenidos y fuera del panel de configuración. Poner el interruptor en posición OFF.

ANEXO 2

DISPOSITIVO USB

(Guía de Usuario)

1. Instalación

- **Requisitos:**

- Sistema operativo Windows XP o mayor
- LabView que cuente con una versión de NI-VISA 3.0 o mayor.

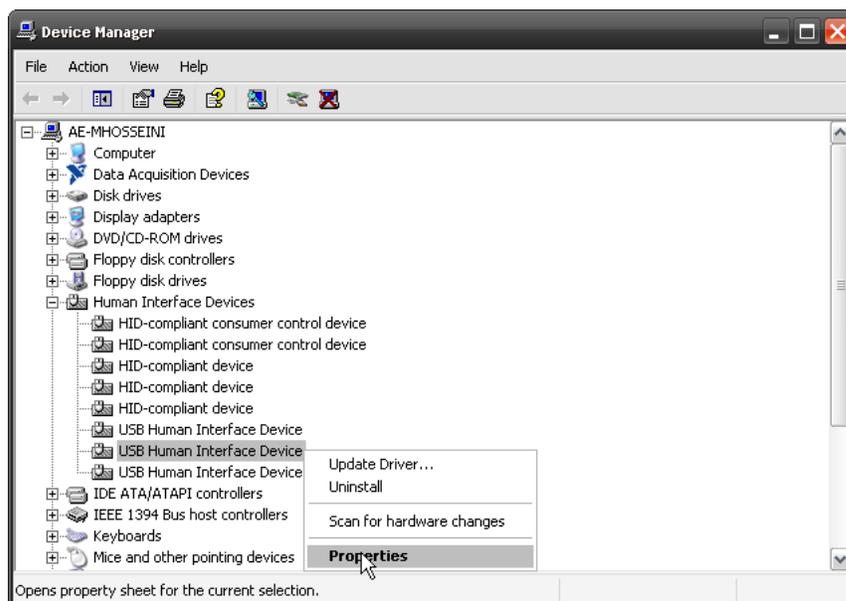
- **Driver:**

Antes de conectar el dispositivo, instalar el archivo DRIVER.INF adjunto en el ordenador.

Para instalarlo, seguir los siguientes pasos descritos en el documento NI-Tutorial 9019 emitido por National Instruments:

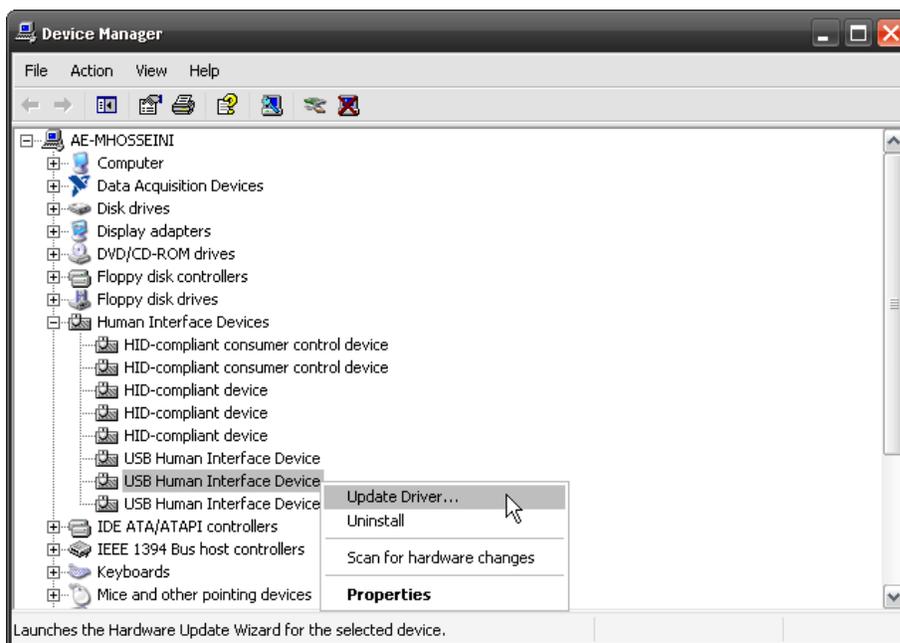
1. Copiar el archivo INF a la carpeta INF. En Windows XP, esta carpeta generalmente se encuentra ubicada en **C:\WINDOWS\INF**. Esta carpeta puede estar escondida, por lo que podría necesitar cambiar las opciones de carpeta para ver los archivos escondidos.
2. Hacer clic derecho con el mouse en el archivo INF en **C:\WINDOWS\INF** y hacer clic en **Instalar**. Este proceso crea un archivo PNF para su dispositivo.
3. Conectar el dispositivo USB. Debido a que USB es insertable durante el funcionamiento, Windows debería detectar su dispositivo USB, y el Asistente de Agregar Nuevo Hardware debería aparecer automáticamente tan pronto como el dispositivo esté conectado al puerto USB. Seguir las instrucciones en la pantalla del asistente. Cuando se le solicite que seleccione un controlador para este dispositivo, buscar la carpeta INF y seleccionar el archivo INF que se generó utilizando el DDW.

Figura 1. Encontrando la Instancia Correcta para el Dispositivo de Interfaz Humana USB para el Dispositivo USB (fuente National Instruments)



Una vez que usted ha encontrado el "Dispositivo de Interfaz Humana de USB" cuyo VID y PID coinciden con el del Dispositivo USB, hacer clic derecho y seleccionar la opción Actualizar Controlador del menú desplegable, como se muestra en la Figura 2.

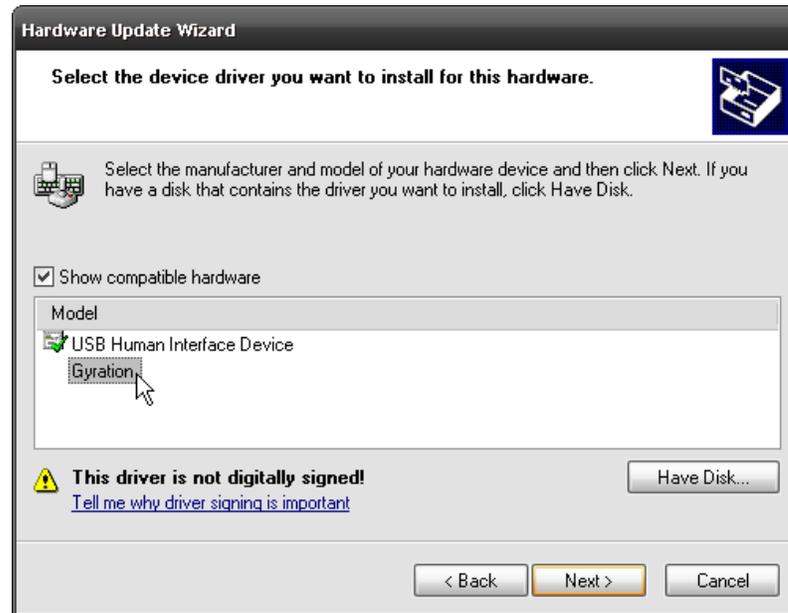
Figura 2. Actualizar el Controlador del Dispositivo USB. (Fuente: National Instruments)



En la primer pantalla, seleccionar "Por ahora no" y después hacer clic en Siguiente. En la segunda pantalla, seleccionar "Instalar desde una lista o una ubicación específica"

(Avanzado)" y después hacer clic en Siguiente. En la tercer pantalla, seleccionar "No buscar. Elegiré el controlador para instalar." La cuarta pantalla será similar a la que se muestra en la Figura 3, en la cual el controlador marcado es el controlador que se creó. Una vez que ya haya seleccionado aquel controlador, hacer clic en Siguiente. Cuando el controlador se haya terminado de instalar, hacer clic en Terminar.

Figura 3. Seleccione el Controlador de su Dispositivo USB (fuente: National Instruments)



2. Conexión

Una vez instalado el driver del dispositivo este se puede conectar y ser identificado para su posterior uso, para esto se deben seguir los siguientes pasos:

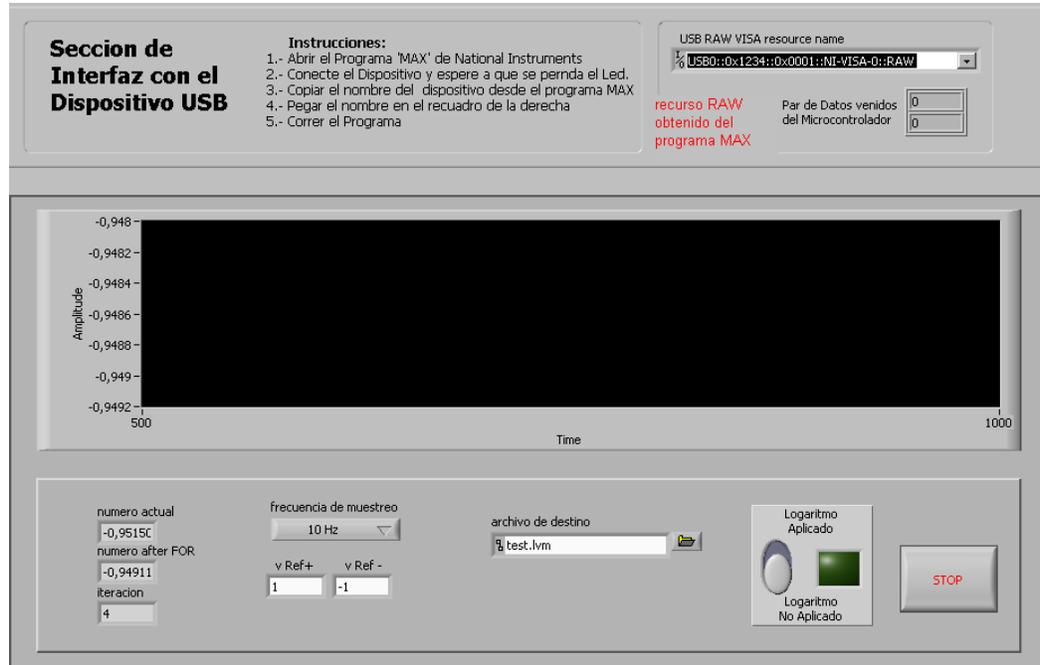
- Conectar el dispositivo al puerto USB del ordenador.
- Conectar el adaptador del dispositivo a la red de alimentación .El LED indicador del dispositivo deberá prenderse por un segundo, luego apagarse por algunos segundos más y finalmente prenderse definitivamente. Si esto no llegara a pasar, reconectar el dispositivo a la toma de voltaje de la red.
- Abrir el programa MAX (Measurement & Instrumentation) que se encuentra dentro del paquete de programas de National Instruments.
- Abrir el subdirectorio USB Devices de Devices & Interfaces, en la lista de Dispositivos de ese subdirectorio.
- Copiar el nombre del recurso del dispositivo que se encuentra en donde indica la figura

3. Ejecución del programa

Para obtener los datos desde el dispositivo a través Software de Labview, seguir los siguientes pasos.

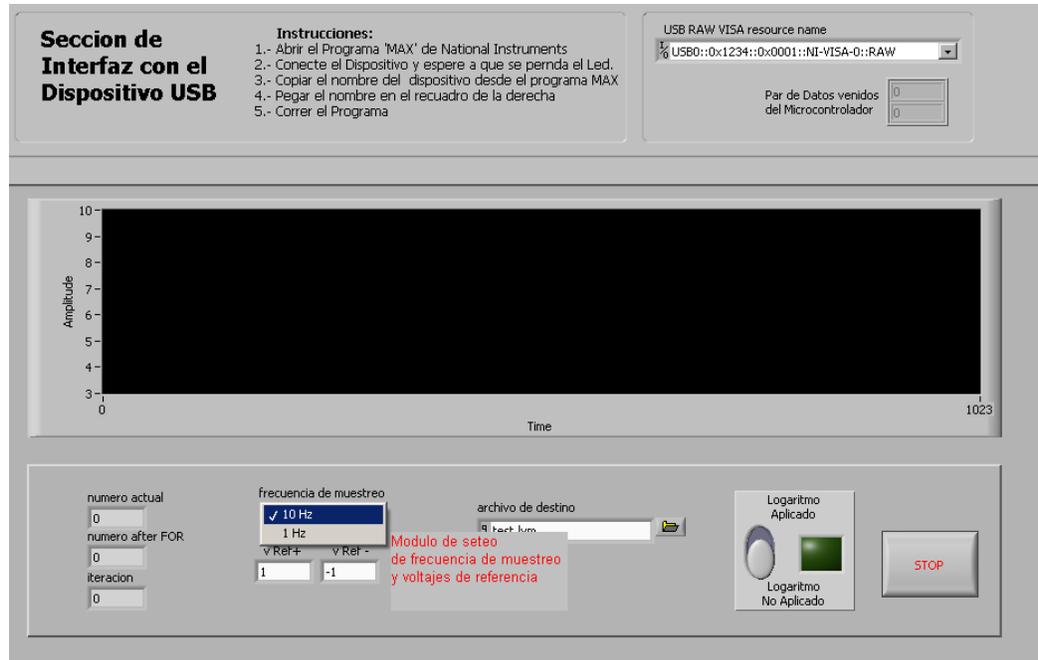
- Copiar el nombre del recurso RAW obtenido del programa MAX y pegarlo en el control que indica el panel frontal:

Figura 4. Elección del recurso RAW en el panel frontal de Labview



- Ingresar los voltajes de referencia y frecuencia de muestreo

Figura 5. Elección de frecuencia de muestreo y de voltajes de referencia en el panel frontal de Labview



- Correr el programa el tiempo que se desee.
- Parar el programa con el botón STOP que se encuentra en el panel frontal y elegir el directorio en donde se requiere que se guarde el achivo.lvm con os datos obtenidos.

Figura 5. Seleccionar archivo en el que se quiera guardar los datos obtenidos por el programa.

