

SISTEMA DE REGISTRO Y PESAJE DE GANADO

**JUAN CARLOS PERDOMO FLOREZ
OSCAR JAVIER HERNANDEZ OSPINA**

**UNIVERSIDAD SURCOLOMBIANA
FACULTAD DE INGENIERIA
PROGRAMA INGENIERIA ELECTRONICA
NEIVA
2008**

SISTEMA DE REGISTRO Y PESAJE DE GANADO

**JUAN CARLOS PERDOMO FLOREZ
OSCAR JAVIER HERNANDEZ OSPINA**

**Trabajo de grado para optar al título
de Ingeniero Electrónico**

**Director
AGUSTIN SOTO OTALORA
Docente Programa Ingeniería Electrónica**

**UNIVERSIDAD SURCOLOMBIANA
FACULTAD DE INGENIERIA
PROGRAMA INGENIERIA ELECTRONICA
NEIVA
2008**

Nota de Aceptación

Director

Firma del jurado

Firma del jurado

Neiva, 2 de abril de 2008

Oscar Javier Hernández

A Dios, a mi abuela, mi madre y mis hermanos los cuales han sido mi apoyo durante el transcurso de mi vida dándome consejos y formándome como persona.

Juan Carlos Perdomo Florez

A Dios, mi madre y mis hermanas, que con sus consejos y apoyo incondicional me han guiado y enseñado a vivir la vida.

AGRADECIMIENTOS

Los autores expresan sus agradecimientos a:

Dios.

Nuestros padres y hermanos.

A nuestro director de tesis Agustín Soto por habernos orientado durante todo el proceso de realización del proyecto.

A nuestros profesores por ayudarnos en nuestra formación profesional y personal.

A nuestros compañeros y amigos por acompañarnos, ayudarnos, escucharnos y enseñarnos a lo largo de este proceso.

CONTENIDO

	Pág.
INTRODUCCIÓN.....	15
1. OBJETIVOS.....	17
2. BÁSCULA.....	18
3. CELDA DE CARGA.....	19
3.1 GALGA EXTENSIOMÉTRICA.....	19
3.1.1 Características del sensor.....	20
3.1.2 Configuración física.....	21
3.1.3 Parámetros para la selección de la celda de carga.....	22
3.1.4 Tipos de celdas de carga.....	23
3.1.4.1 Celdas mecánicas.....	23
3.1.4.2 Celdas de carga eléctrica.....	23
4. AMPLIFICADORES DE INSTRUMENTACION.....	24
5. FILTROS ACTIVOS PASA BAJO.....	26
6. MULTIVIBRADOR MONOESTABLE.....	27
7. PIC 16F877A.....	28
7.1 CARACTERISTICAS.....	29
7.2 DESCRIPCIÓN DE LOS PUERTOS.....	29
8. LCD.....	31
9. CIRCUITO INTEGRADO PARA CONVERSIÓN DE NIVELES.....	32
10. ELEMENTOS SOFTWARE.....	33
10.1 MPLAB.....	33
10.1.1 COMPILACIÓN DEL PROGRAMA Y CARGA AL PIC.....	33
10.2 MICROSOFT VISUAL BASIC.NET.....	34
10.2.1 Características generales.....	34
10.2.2 Win32 APIs, APIs de Windows.....	35
11. DISEÑO.....	36
11.1 HARHWARE.....	37
11.1.1 Bloque A: Etapa de adquisición de señal.....	37
11.1.2 Etapas B y C: Adecuación de la señal.....	39
11.1.2.1 Bloque B: Amplificación.....	39
11.1.2.2 Bloque C: Filtro pasa bajo.....	30
11.1.3 Bloque D. Etapa de control de funciones.....	42
11.1.4 Bloque E. Etapa de procesamiento de señal.....	43
11.1.5 Bloque F. Etapa de almacenamiento de datos.....	45
11.1.6 Bloque G. Etapa de visualización.....	46
11.1.7 Fuentes de alimentación.....	47
11.2 SOFTWARE.....	47
12.2.1 Modo LCD.....	49
12.2.2 Modo PC.....	50
13. RESULTADOS.....	60
14. CONCLUSIONES.....	62
15. RECOMENDACIONES.....	63
16. FUENTES DE CONSULTA.....	64
16.1. BIBLIOGRAFIA.....	64
16.2. MEDIOS TELEINFORMÁTICOS.....	64

LISTA DE TABLAS

	Pág.
Tabla 1. Características del PIC 16F877.....	29
Tabla 2. Respuesta de una barra de pesaje de 0 – 500kg alimentada a 10V.....	38
Tabla N3. Comparación de las mediciones de la báscula del Fondo Ganadero del Huila y la báscula diseñada.....	60

LISTA DE FIGURAS

	Pág.
Figura 1. Báscula electrónica.....	18
Figura 2. Galga extensiométrica.....	19
Figura 3. Puente de medida con una galga.....	21
Figura 4. Puente de medida con dos galgas (Medio puente).....	22
Figura 5. Puente de medida con cuatro galgas (Puente completo).....	22
Figura 6. Amplificador de instrumentación.....	25
Figura 7. Filtro activo paso bajo con Amplificador Operacional.....	26
Figura 8. Circuito monoestable y respuesta obtenida.....	27
Figura 9. Display LCD de texto de 16x2 con controlador HD44780.....	31
Figura 10. MAX232 - Diagrama De Pines.....	32
Figura 11. Programador Picstart Plus de Microchip.....	33
Fugura12. Diagrama en bloques Sistema de Registro y Pesaje.....	36
Figura 13. Etapa de adquisición de señal.....	37
Figura 14. a) Imagen de la barra de pesaje b) Conexiones internas.....	37
Figura 15. Curva de Respuesta de una barra de pesaje de 0 – 500kg alimentada a 10V.....	38
Figura 16. Etapa de adecuación de señal.....	39
Figura 17. Circuito Amplificador de Señal.....	40
Figura 18. Circuito Filtros pasa bajos.....	41
Figura 19. Etapa de control de funciones.....	42
Figura 20. Circuito de control.....	43
Figura 21. Etapa de procesamiento.....	44
Figura 22. Circuito de procesamiento de señal.....	45
Figura 23. Etapa de almacenamiento de datos.....	45
Figura 24. Cable DKU-5 – diagrama del circuito.....	46
Figura 25. Etapa de visualización.....	46
Figura 26. LCD YM202-1.....	47
Figura 27. Diagrama circuital fuentes de voltaje.....	47
Figura 28. Diagrama de flujo programa PIC 16F877A.....	48
Figura 29. Inicio de instalación.....	51
Figura 30. Selección de carpeta de instalación.....	52
Figura 31. Confirmación de la instalación.....	52
Figura 32. Pantalla interfase del programa Sistema de Registro y Pesaje de Ganado.....	53
Figura 33. Primer criterio de búsqueda.....	55
Figura 34. Segundo criterio de búsqueda.....	55
Figura 35. Resultados de la consulta.....	56
Figura 36.	
Figura 37. Pantalla conexión con báscula.....	57
Figura 38. Iniciar nueva sesión.....	58
Figura 39. Lector de Código de Barras.....	59
Figura 40. Formulario de Nuevo registro.....	59
Figura 41. Gráfica de las mediciones de la báscula del Fondo Ganadero..	61

LISTA DE ANEXOS

	Pág.
ANEXO A. PROGRAMA PARA EL PIC 16F877A.....	64
ANEXO B. TARJETAS DE CIRCUITO IMPRESO.....	75
ANEXO C. PLANO GENERAL SISTEMA DE REGISTRO Y PESAJE DEGANADO.....	76

GLOSARIO

API: Una API (del inglés Application Programming Interface - Interfaz de Programación de Aplicaciones) es el conjunto de funciones y procedimientos (o métodos si se refiere a programación orientada a objetos) que ofrece cierta librería para ser utilizado por otro software como una capa de abstracción.

Amplificador operacional: Un amplificador operacional (A.O., habitualmente llamado op-amp) es un circuito electrónico (normalmente se presenta como circuito integrado) que tiene dos entradas y una salida. La salida es la diferencia de las dos entradas multiplicada por un factor (G) (ganancia): $V_{out} = G \cdot (V_+ - V_-)$.

CheckBox: El control CheckBox de los formularios Windows Forms indica si una condición determinada está activada o desactivada. Se utiliza habitualmente para presentar al usuario una selección de tipo Sí/No o Verdadero/Falso. Puede utilizar grupos de casillas de verificación para mostrar múltiples opciones entre las cuales el usuario puede elegir una o más.

Cizallamiento: El esfuerzo de cizallamiento se denomina la tensión, que actúa paralelamente al área. El esfuerzo de cizallamiento da origen a una deformación por fractura.

Conversión A/D: Una conversión analógica-digital (CAD) consiste en la transcripción de señales analógicas en señales digitales, con el propósito de facilitar su procesamiento (codificación, compresión, etc.) y hacer la señal resultante (la digital) más inmune al ruido y otras interferencias a las que son más sensibles las señales analógicas.

DESTARE: lo que se rebaja de lo que se ha pesado.

DLL: DLL es el acrónimo de *Dynamic Linking Library* (*Bibliotecas de Enlace Dinámico*), término con el que se refiere a los archivos con código ejecutable que se cargan bajo demanda del programa por parte del sistema operativo. Esta denominación se refiere a los sistemas operativos Windows siendo la extensión con la que se identifican los ficheros, aunque el concepto existe en prácticamente todos los sistemas operativos modernos.

Extensión .ASM: archivo de bloc de notas para compilar en MPLAB.

Extensiométrico: La presión produce un esfuerzo que hace cambios, la longitud del alambre y consecuentemente cambia la resistencia del mismo.

Hardware: refiere a todos los componentes físicos (que se pueden tocar), en el caso de una computadora personal serían los discos, unidades de disco, monitor, teclado, la placa base, el microprocesador, etc.

Label: etiqueta donde se visualiza un texto.

ListBox: Un control ListBox de formularios Windows Forms muestra una lista de elementos de los cuales el usuario puede seleccionar uno o más.

ListView: El control ListView de formularios Windows Forms muestra una lista de elementos con iconos. Puede utilizar una vista de lista para crear una interfaz de usuario similar al panel derecho del Explorador de Windows.

LCD: (*Liquid Crystal Display*) son las siglas en inglés de Pantalla de Cristal Líquido, Se trata de un sistema eléctrico de presentación de datos formado por 2 capas conductoras transparentes y en medio un material especial cristalino (cristal líquido) que tienen la capacidad de orientar la luz a su paso. Cuando la corriente circula entre los electrodos transparentes con la forma a representar (por ejemplo, un segmento de un número) el material cristalino se reorienta alterando su transparencia.

Monoestable: Se llama multivibrador monoestable a un dispositivo capaz de pasar de un estado estable a otro inestable, por los efectos de un impulso, con la particularidad de que el estado inestable tiene una duración que depende de las constantes del dispositivo. Todo vuelve a empezar cuando se produce un nuevo impulso.

OCX: El término OCX hace referencia a instrucciones preprogramadas capaces de ser insertadas en cualquier aplicación que está siendo desarrollada, y permite al programador ahorrar tiempo debido a que en lugar de reprogramar un código anteriormente hecho, ahora únicamente lo insertamos.

PIC: *Peripheral Interface Controller* (Controlador de Interfaz Periférico), Los 'PIC' son una familia de microcontroladores tipo RISC fabricados por Microchip Technology Inc. y derivados del PIC1650, originalmente desarrollado por la división de microelectrónica de General Instruments.

Puente de Wheatstone: Se utiliza para medir resistencias desconocidas mediante el equilibrio de los brazos del puente. Estos están constituidos por cuatro resistencias que forman un circuito cerrado, siendo una de ellas la resistencia bajo medida.

Sallen-Key: El circuito produce un filtro pasa bajo o pasa alto de dos polos usando dos resistencias, dos condensadores y un amplificador. Para obtener un filtro de orden mayor se pueden poner en cascada varias etapas.

Sensor óptico: sensor que utiliza luz para ser excitado.

Errores Sintácticos: producidos por mal escritura de los comandos o instrucciones.

Software: programa, equipamiento lógico o soporte lógico a todos los componentes intangibles de una computadora, es decir, al conjunto de programas y procedimientos necesarios para hacer posible la realización de una tarea específica, en contraposición a los componentes físicos del sistema.

Subutilizar: no utilizar toda la capacidad de un elemento.

TextBox: Muestra un control de cuadro de texto para la entrada de datos del usuario.

Trimer: resistencia variable lineal.

RESUMEN

El objetivo del proyecto era implementar un sistema que permitiera el pesaje de ganado vacuno, así mismo fuera capaz de almacenar y procesar los datos obtenidos, buscando minimizar costos de operación, agilidad y confiabilidad del proceso de registro y pesaje en una ganadería.

El proyecto presenta dos modos de operación, MODO PIC y MODO PC según como el usuario decida trabajar.

El MODO PIC no requiere PC debido a que trabaja de modo autónomo visualizando la medición actual, almacenando el total pesado, el promedio y el número de animales que se han pesado en la sesión.

El MODO PC trabaja con un software desarrollado en Visual Basic.Net que nos permite llevar una base de datos con las características de: lote, número, peso, fecha y observación además mediante una interface gráfica de usuario (GUI) el operario llevar a cabo el proceso de una forma fácil y confiable.

La adquisición de las señales se hizo por medio de celdas de carga tipo SB especiales para este tipo de ambientes donde encontraremos humedad, calor, barro, excremento de animales y todo tipo de condiciones hostiles.

La adecuación de la señal (Amplificación – Filtrado), se implementó por medio de amplificadores de instrumentación AD620 que amplifica la señal de mV obtenidas en los transductores a un rango de 0 – 5V para el conversor AD y amplificadores operacionales TL082 configurados en modo filtro pasa bajos topología Sallen Key.

El procesamiento (Conversión AD – Suma Canales – Almacenamiento – Control y Visualización), fue hecho usando el PIC 16F877A el cual se ajustó a todos los requerimientos de diseño propuesto entre ellos el manejo de LCD YM202-1 con el cual, el usuario visualiza las mediciones.

Para conectar el módulo al PC se utilizo un cable DKU – 5 (Conversor RS232 – USB) y por último, para llevar el registro en la base de datos se usó un lector de códigos de barras el cual identifica el animal que se esta pesando.

ABSTRACT

The project's objective was to implement a system that would allow the weighing of cattle. Also, that it was capable of store and process the data obtained, in that way looking to diminish operation's costs, agility and reliability of the process' registration and weighing of the cattle.

The project shows two operation modes, Mode PIC and Mode PC depending on how the user decides to work .

Mode PIC does not require a PC, due to its mode that Works in a self- sufficient way, visualizing the actual measurement, storing the total weighing and also showing the average and the number of animals that have been weighed at a given time.

Mode PC Works with a software developed with Visual Basic.net. This allows to keep track of a data base containing the following: lote, number, weight, date and observations. Furthermore, using an user's graphic Interface (GUI), the user can carry out processes in an easy and reliable way.

The signals' gathering was made thru SB type charge cells, which are specially used in this kind of environment where humidity, heat, soil, animal's feces and all kind of hostile conditions would be found

The signal's adjustment (amplification – filtration) was made with the use of instrumentation's amplifiers type AD620 that amplify the mV signals obtained in the transducers, with a range of 0 – 5V for the AD converter and operational amplifiers TL082 set in the low pass filter topology Sallen Key.

The process (AD conversion – channel sum – storage – control and visualization) was made using the PIC 16F 877A , which fit the design's requirements proposed, one of those requirements being the use of the LCD YM 202-1, that in this case, aids the user to observe the measurements been made.

In order to connect the scale to the PC, it was used a DKU – 5 cable(RS 232 converter – USB). Finally, to keep track of the records at the data base, it was used a bar code's reader that identifies which animal is been weighed.

INTRODUCCIÓN

Unas buenas instalaciones para el manejo de ganado son la parte esencial en cualquier operación ganadera, las instalaciones bien construidas y funcionales contribuyen a facilitar el manejo seguro y rápido del mismo. Las buenas instalaciones para manejo de ganado no son necesariamente de costo excesivo, resulta bien invertir en productos de calidad en algunas de las áreas una de ellas el pesaje, mayormente si se trata de ganado de ceba.

Los registros o toma de información sobre un animal son muy útiles si se tienen en cuenta los datos relacionados a la producción de este, llevar registros tiene un costo y por lo tanto se justifica llevar sólo los indispensables, sin embargo, llevar registros de producción y analizarlos con el fin de tomar las decisiones a tiempo como son seleccionar los animales superiores y descartar los inferiores, es uno de los procesos más rentables en la empresa ganadera. El descarte elimina individuos inferiores que reducen las utilidades que producen los superiores y además permite obtener progreso genético transmisible a la descendencia lo cual es de vital importancia a la hora de comercializar un lote o animal.

Las necesidades y situaciones de cada ganadero son diferentes, algunos pueden adquirir directamente de catálogos o empresas, otros tendrán que rediseñar el sistema para acomodarlo a sus instalaciones ganaderas y a sus necesidades individuales lo cual presenta unos gastos adicionales y por lo tanto cierto rechazo a la implementación de nuevas tecnologías permaneciendo en la practica convencional del oficio, lo cual funciona pero es menos eficiente y rentable a mediano plazo.

Basados en lo anterior se desarrollo una alternativa económica, práctica y sobretodo confiable en lo que se refiere a pesaje y registro de ganado, de manera que sea mas asequible al ganadero de la región con el fin de garantizar un mayor nivel de competencia en el mercado ganadero.

El libro inicia mostrando una breve introducción con definiciones y marco teórico indispensable para una buena comprensión de este, se tiene la teoría básica de Celdas de Cargas, Amplificadores de Instrumentación, Filtros Activos Pasabajo, Multivibrador Monoestable, el PIC 16F877A, LCDs y MAX232

Una vez expuesta la teoría básica acerca del funcionamiento del hardware se describe los criterios básicos de la programación en Visual Basic.NET 2005 y MPLAB con el cuales se desarrollo la parte de software.

Ya con el Marco teórico evacuado se pasa a la etapa de diseño donde se explica el por que de los dispositivos utilizados y la configuración de los

mismos. Esta parte es de vital importancia puesto que se abordan los problemas y soluciones presentados a lo largo del proyecto.

A continuación se expresa el proyecto por medio de un diagrama de bloques para tener una vista más detallada bloque a bloque y como interactúan los diferentes componentes del proyecto buscando una más precisa y fácil comprensión.

Para la parte de software se muestra un tutorial de manejo del programa desarrollado para el registro y manejo de los datos procesados.

Los resultados obtenidos se presentarán conforme a las pruebas realizadas en el trabajo de campo y confrontados con básculas comerciales de la región, de igual manera se expondrán las conclusiones y recomendaciones a que dieron lugar las pruebas realizadas y el desarrollo del proyecto en sí.

En los anexos se encuentran el programa del PIC y los formularios desarrollados en Visual Basic.NET 2005 y algunos datos importantes del proyecto.

1. OBJETIVOS

General

Implementar un sistema electrónico para realizar la identificación, el registro y el pesaje de ganado vacuno.

Específicos

1. Implementar una plataforma de pesaje para ganado de 2000Kg de capacidad.
2. Construcción de un módulo que permita procesar, visualizar y exportar la información obtenida a través de los sensores (ópticos y de peso).
3. Por medio de sensores ópticos y software identificar el individuo a tratar.
4. Realizar el registro de la información en una base de datos por medio de un software de programación.

2. BÁSCULA

Figura 1. Báscula electrónica



Fuente: <http://img211.imageshack.us/img211/3807/frontalpk5.jpg>

¹La báscula, también llamada en algunos lugares pesa, juntamente con la romana y la balanza son los tres instrumentos u operadores técnicos que se han diseñado e inventado para el peso de masas. La báscula fue el operador que se inventó para pesar las masas que no se podían pesar en las romanas, bien porque fuesen demasiado pesadas o bien porque no se pudiesen colgar de los ganchos de las romanas. Así que lo característico de las básculas es que tienen una plataforma a ras de suelo, donde resulta fácil colocar la masa que se quiere pesar.

Esta facilidad para poner masas grandes y pesadas encima de la plataforma es lo que ha hecho posible construir básculas con una capacidad de peso muy grande que son utilizadas para pesar camiones de gran tonelaje.

Con el tiempo las básculas han evolucionado mucho y hoy día ya funcionan con métodos y sistemas electrónicos dando una lectura rápida y directa del peso de la masa.

Las básculas miden la fuerza ejercida por un objeto sujeto a la fuerza de gravedad. Gracias a la relación $F = m \cdot a$, siendo $a =$ la gravedad, es posible calcular la masa. Las básculas se tienen que "calibrar" en donde se vayan a utilizar, debido a las diferencias en la fuerza de gravedad en diferentes partes del planeta. El método utilizado para calibrar es por comparación a estándares o patrones internacionales definidos de masa (el kilogramo, la libra, etc.) La división se hace automáticamente por comparación, ya que se toma teóricamente una fuerza de gravedad constante... si la fuerza de gravedad es constante, entonces la masa es directamente proporcional a la fuerza.

Comercialmente existen dos tipos de básculas: mecánicas o electrónicas. Las básculas mecánicas actúan por medio de relación de palancas. Las básculas electrónicas utilizan un sensor mejor conocido como celda de carga.

¹ Texto tomado de <http://es.wikipedia.org/wiki/B%C3%A1scula>

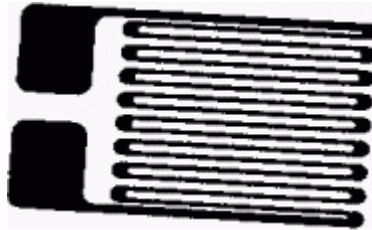
3. CELDA DE CARGA

²El principio básico de una celda de carga esta basado en el funcionamiento de cuatro galgas extensiométricas (strain gauge), dispuestos en una configuración especial.

3.1 GALGA EXTENSIONOMÉTRICA

Una galga extensiométrica es un dispositivo electrónico que utiliza el efecto piezorresistivo para medir deformaciones. Ante un cambio en la estructura del material de la galga se producirá una variación de su resistencia eléctrica. Los elementos que suelen utilizarse para construir galgas son aleaciones de cobre y níquel, platino y silicio. Esta es básicamente una resistencia eléctrica. El parámetro variable y sujeto a medida es la resistencia de dicha galga. Esta variación de resistencia depende de la deformación que sufre la galga.

Figura 2. Galga extensiométrica.



Fuente: <http://www.forosdeelectronica.com/about35.html>

Una galga extensiométrica consiste de un hilo metálico muy fino, o más comúnmente una lamina de papel metálico arreglado en forma de rejilla como se muestra en la figura. Esta forma permite aprovechar la máxima cantidad de material de la galga sujeto a la tensión a lo largo de su eje principal.

La resistencia eléctrica del hilo es directamente proporcional a su longitud, o lo que es lo mismo, su resistencia aumenta cuando éste se alarga. De este modo las deformaciones que se producen en el objeto, en el cual está adherida la galga, provocan una variación de la longitud y, por consiguiente, una variación de la resistencia.

Otro principio de funcionamiento se basa en la deformación de elementos semiconductores. Esta deformación provoca un cambio, tanto en la longitud

² Texto tomado de <http://www.forosdeelectronica.com/about35.html>

como en la sección transversal, este tipo de sensor semiconductor posee un factor de galga más elevado que el construido por hilo metálico.

Su principal ventaja es su linealidad; también presentan una baja impedancia de salida. Su principal desventaja es su dependencia de la temperatura, lo que provoca que, a veces, haya que diseñar circuitos electrónicos para compensar esa dependencia.

3.1.1 Características del sensor. Las principales características de las galgas son las siguientes:

- **Anchura y Longitud:** Estos parámetros proporcionan las características constructivas de la galga. Permite escoger el tamaño del sensor que más se adecue a nuestras necesidades.
- **Peso:** Define el peso de la galga. Este suele ser del orden de los gramos. En aplicaciones de mucha precisión el peso puede influir en la medida de la deformación.
- **Tensión medible:** Es el rango de variación de longitud de la galga, cuando ésta se somete a una deformación. Este rango viene expresado en un tanto por cien respecto a la longitud de la galga.
- **Temperatura de funcionamiento:** Es aquella temperatura para la cual el funcionamiento de la galga se encuentra dentro de los parámetros proporcionados por el fabricante.
- **Resistencia de la galga:** Es la resistencia de la galga cuando ésta no está sometida a ninguna deformación. Es la resistencia de referencia y suele acompañarse de un porcentaje de variación.
- **Factor de galga:** El parámetro que define la sensibilidad de una galga se conoce como factor de galga GF , y es definido como el cociente entre el cambio fraccional de la resistencia eléctrica y la tensión ε . En la figura se aprecia el cálculo del factor de galga. El factor de galga al igual que la tensión ε es adimensional.
- **Coefficiente de temperatura del factor de galga:** La temperatura influye notablemente en las características. A su vez, cualquier variación en estas características influye en el factor de galga. Este coeficiente se mide en $\% / ^\circ\text{C}$, que es la variación porcentual del valor nominal del factor de galga respecto al incremento de temperatura.
- **Prueba de fatiga:** Indica el número de contracciones o deformaciones a una determinada tensión que puede soportar la galga sin romperse.
- **Material de la lámina:** Define el material del que está hecho de hilo conductor o el material semiconductor.

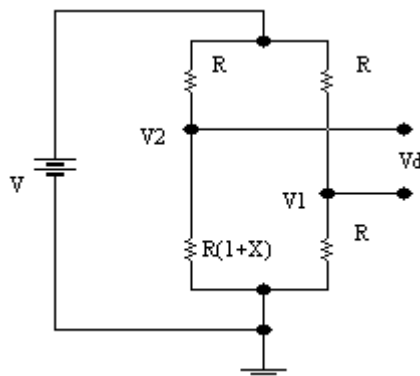
- **Material de la base:** Define el material del que está constituida la base no conductora de la galga.

- **Factor de expansión lineal:** Representa un error que se produce en la magnitud de salida en ausencia de señal de entrada, es decir, en ausencia de deformación. Este error depende de la temperatura ambiente a la que esta sometida la galga.

3.1.2 Configuración física. El montaje más común utilizado para medir deformaciones mediante galgas es el puente de Wheatstone. Existen tres tipos de montajes básicos: con una, dos y cuatro galgas. La medida se suele realizar por deflexión, es decir midiendo la diferencia de tensión existente entre los terminales de salida del sensor.

Las principales diferencias de estos montajes se encuentran en la sensibilidad y la capacidad de compensación del efecto de temperatura. Esta compensación consiste en suprimir los efectos de la temperatura en el valor de la resistencia de la galga; cuando en un puente de medida coinciden dos o cuatro galgas de iguales características, los efectos de la temperatura se les anulan ya que ésta les afecta por igual.

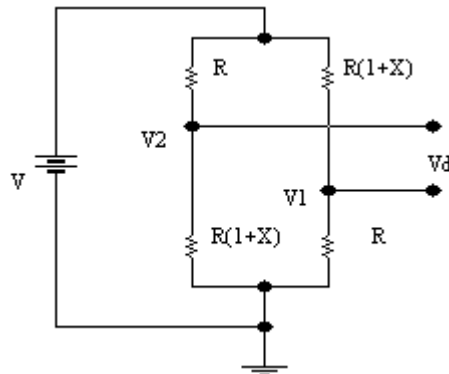
Figura 3. Puente de medida con una galga.



Fuente: <http://www.forosdeelectronica.com/about35.html>

Este puente de medida se caracteriza por una baja sensibilidad. Por otro lado al solo haber una galga esta no está compensada en temperatura.

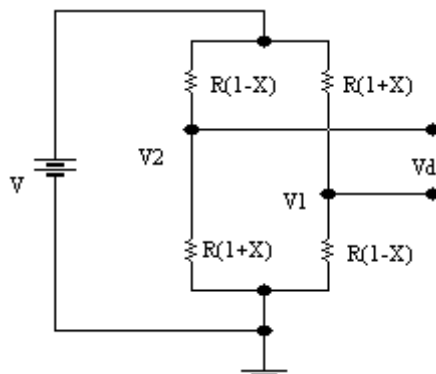
Figura 4. Puente de medida con dos galgas (Medio puente).



Fuente: <http://www.forosdeelectronica.com/about35.html>

Debido a la utilización de dos galgas se consigue duplicar la sensibilidad del puente respecto al anterior. Esto permite que para una misma deformación tengamos una mayor señal de salida para una tensión de alimentación dada. La disposición de las galgas, permiten la compensación en temperatura.

Figura 5. Puente de medida con cuatro galgas (Puente completo).



Fuente: <http://www.forosdeelectronica.com/about35.html>

La utilización de cuatro galgas cuadruplica la sensibilidad del puente respecto al puente de una sola galga. De igual forma que en el caso anterior, las galgas están compensadas en temperatura.

3.1.3 Parámetros para la selección de la celda de carga.

▪ **Capacidad:** Se refiere al peso máximo al que se puede someter la celda de carga. La capacidad real de la carga vendría dada por:

$$\text{Capacidad real} = \text{Peso muerto} + \text{peso neto máximo} + \text{tolerancia}$$

El peso muerto se define como el peso de la plataforma y el soporte y se debe tener en cuenta, puesto que también ejerce una fuerza sobre la plataforma.

- **Sensitividad:** Indica la señal que la celda entrega cuando se esta alimentando con un voltaje de 1 V, y se tiene con el peso máximo al que se puede somete. Viene dada en mV/V.

3.1.4 Tipos de celdas de carga.

3.1.4.1 Celdas mecánicas.

- **Hidráulicas:** Tienen dispositivos que balancean las fuerzas; y el peso se mide como un cambio en la presión del fluido de llenado interno.
- **Neumáticas:** También operan con el principio de fuerzas balanceadas. Estos dispositivos utilizan múltiples cámaras de humedad para dar más exactitud de la que puede dar un dispositivo hidráulico.

3.1.4.2 Celdas de carga eléctrica. Las celdas de carga eléctricas, basadas en galgas extensométricas, convierten la carga o fuerza (peso) en señales eléctricas. Toman muchas formas para adecuarse a la variedad de usos a lo largo de todas las aplicaciones en investigación y en la industria.

- **Tipo viga “S” o “Z”:** Es un diseño de celda de carga simple donde la estructura toma la forma de una “S” o de una “Z” y las galgas extensométricas están conectadas al área central de sensado en configuración puente Wheatstone completo.
- **Bending Beam:** En estas celdas de carga, las galgas extensométricas están conectadas en las secciones planas superior e inferior de la celda, en los puntos de máxima tensión.
- **Shear Beam:** En estas celdas de carga, las galgas extensométricas están conectadas en una reducida parte de la sección transversal de la viga con el objeto de maximizar el efecto de cizallamiento (**shear**). Estan conectadas en ángulos de 45° sobre ambos lados de la viga para medir las tensiones **shear**.
- **Single Point:** son celdas de carga de diseño similar a las **shear beam** pero están diseñadas para utilizarse en aplicaciones que utilizan una sola celda, como las básculas de plataforma pequeñas de uso comercial.

4. AMPLIFICADORES DE INSTRUMENTACIÓN

³El amplificador de instrumentación es un amplificador diferencial de tensión-tensión cuya ganancia puede establecerse de forma muy precisa y que ha sido optimizado para que opere de acuerdo a su propia especificación aún en un entorno hostil. Es un elemento esencial de los sistemas de medida, en los que se ensambla como un bloque funcional que ofrece características funcionales propias e independientes de los restantes elementos con los que interacciona.

Para ello, se le requiere:

- a) Tengan unas características funcionales que sean precisas y estables.
- b) Sus características no se modifiquen cuando se ensambla con otros elementos.

A los amplificadores de instrumentación se les requieren las siguientes características:

- 1) Son amplificadores diferenciales con una ganancia diferencial precisa y estable, generalmente en el rango de 1 a 1000.
- 2) Su ganancia diferencial se controlada mediante un único elemento analógicos (potenciómetro resistivo) o digital (conmutadores) lo que facilita su ajuste.
- 3) Su ganancia en modo común debe ser muy baja respecto de la ganancia diferencial, esto es, debe ofrecer un CMRR muy alto en todo el rango de frecuencia en que opera.
- 4) Una impedancia de entrada muy alta para que su ganancia no se vea afectada por la impedancia de la fuente.
- 5) Una impedancia de salida muy baja para que su ganancia no se vea afectada por la carga que se conecta a su salida.
- 6) Bajo nivel de la tensión de offset del amplificador y baja deriva en el tiempo y con la temperatura, a fin de poder trabajar con señales de continua muy pequeñas.
- 7) Una anchura de banda ajustada a la que se necesita en el diseño.

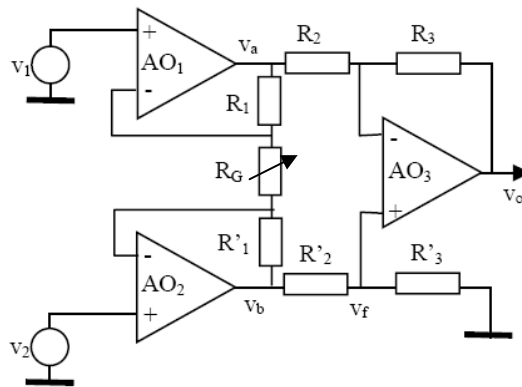
³ Texto tomado de http://es.wikipedia.org/wiki/Amplificador_de_instrumentaci%C3%B3n

8) Un factor de ruido muy próximo a la unidad, Esto es, que no incremente el ruido.

9) Una razón de rechazo al rizado a la fuente de alimentación muy alto.
Fuente Amplificador Carga

La configuración mas utilizada como amplificador de instrumentación está constituida por tres amplificadores operacionales utilizados de acuerdo con el esquema de la figura.

Figura 6. Amplificador de instrumentación.



Fuente:http://es.wikipedia.org/wiki/Amplificador_de_instrumentacion

Los amplificadores de instrumentación han sido desarrollados para ser utilizados en sistemas de instrumentación en los que las características de operación son críticas. Las características de los amplificadores de instrumentación pueden optimizarse si se diseñan como circuitos integrados, ya que en este caso, el fabricante puede garantizar el diseño de los elementos críticos, haciendo que tengan valores precisos y que las relaciones entre las características de elementos emparejados tengan razones muy exactas, justo tal como se requiere en su diseño.

La precisión y estabilidad de los amplificadores de instrumentación se realiza a costa de limitar su flexibilidad. Son amplificadores que han sido diseñados para ser utilizados únicamente como amplificadores, pero a cambio de ello, proporcionan unas características excepcionalmente buenas, y además pueden utilizarse sin necesidad de conocer con detalle su diseño interno y con sólo interpretar su especificación externa.

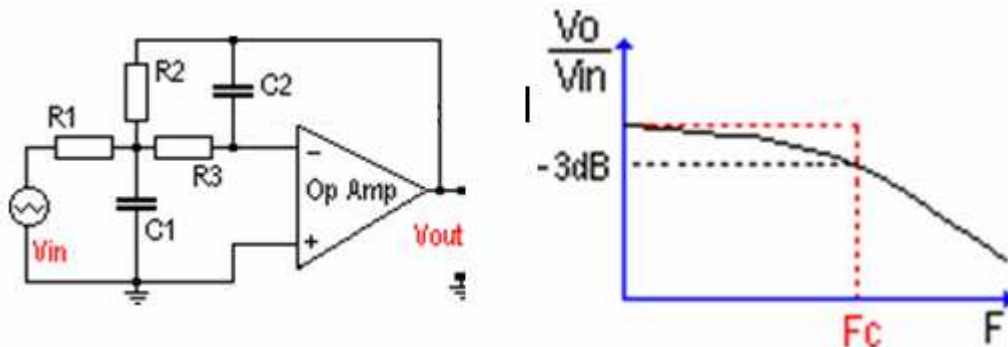
5. FILTROS ACTIVOS PASA BAJO

⁴ Se diferencian de los filtros comunes, en que estos últimos son solamente una combinación de resistencias, capacitores e inductores.

En un filtro común, la salida es de menor magnitud que la entrada en cambio los filtros activos se componen de resistores, capacitores y dispositivos activos como Amplificadores Operacionales o transistores.

En un filtro activo la salida puede ser de igual o de mayor magnitud que la entrada.

Figura 7. Filtro activo paso bajo con Amplificador Operacional.



Fuente: <http://redescolar.ilce.edu.mx/redescolar/cursos/sepacomputo>

Si se seleccionan los capacitores de modo que:

$$C1 = C2 = C \quad \text{y} \quad R1 = R2 = R3 = R$$

El valor de la frecuencia **Fc** (frecuencia de corte) se puede obtener con ayuda de la siguiente fórmula:

$$F_c = 0.0481 / RC.$$

Y la ganancia del filtro (acordarse de que es un amplificador) será:

$$A_v = V_o / V_{in} = R2 / R1.$$

Si se expresa esta ganancia en decibeles:

$$A_v = 20 \text{Log } V_o / V_{in} \quad \text{o} \quad A_v = 20 \text{ log } R2 / R1$$

⁴ Texto tomado de <http://redescolar.ilce.edu.mx/redescolar/cursos/sepacomputo/pdimtex.pdf>

6. MULTIVIBRADOR MONOESTABLE

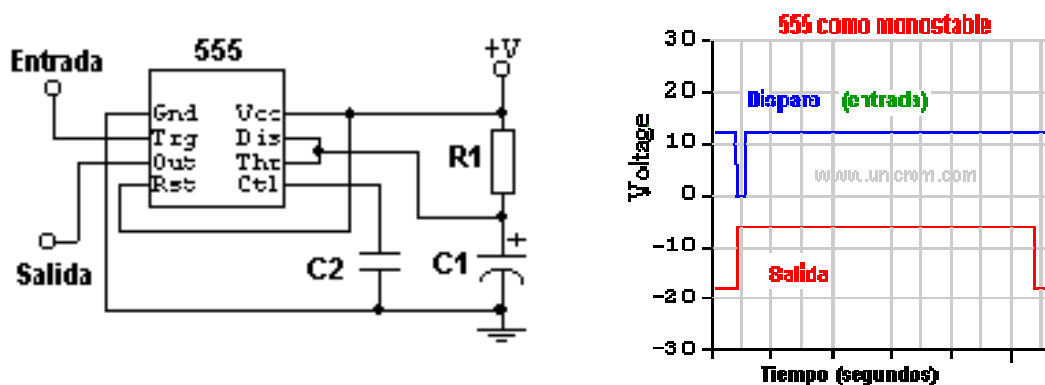
⁵En este caso el circuito entrega a su salida un solo pulso de un ancho establecido por el diseñador (tiempo de duración).

El esquema de conexión es el que se muestra. La Fórmula para calcular el tiempo de duración (tiempo que la salida esta en nivel alto) es:

$$T = 1.1 \times R1 \times C1 \text{ (en segundos).}$$

Observar que es necesario que la señal de disparo, sea de nivel bajo y de muy corta duración en el PIN # 2 del C.I. para iniciar la señal de salida.

Figura 8. Circuito monoestable y respuesta obtenida.



Fuente: http://www.unicrom.com/tut_555.asp

⁵ Texto tomado de http://www.unicrom.com/tut_555.asp

7. PIC 16F877A

⁶Se denomina microcontrolador a un dispositivo programable capaz de realizar diferentes actividades que requieran del procesamiento de datos digitales y del control y comunicación digital de diferentes dispositivos.

Los microcontroladores poseen una memoria interna que almacena dos tipos de datos; las instrucciones, que corresponden al programa que se ejecuta, y los registros, es decir, los datos que el usuario maneja, así como registros especiales para el control de las diferentes funciones del microcontrolador.

Los microcontroladores se programan en Assembler y cada microcontrolador varía su conjunto de instrucciones de acuerdo a su fabricante y modelo. De acuerdo al número de instrucciones que el microcontrolador maneja se le denomina de arquitectura RISC (reducido) o CISC (complejo).

Los microcontroladores poseen principalmente una ALU (Unidad Lógica Aritmética), memoria del programa, memoria de registros, y pines I/O (entrada y/o salida). La ALU es la encargada de procesar los datos dependiendo de las instrucciones que se ejecuten (ADD, OR, AND), mientras que los pines son los que se encargan de comunicar al microcontrolador con el medio externo; la función de los pines puede ser de transmisión de datos, alimentación de corriente para el funcionamiento de este o pines de control específico.

En este proyecto se utilizó el PIC 16F877A. Este microcontrolador es fabricado por MicroChip familia a la cual se le denomina PIC. El modelo 16F877A posee varias características que hacen a este microcontrolador un dispositivo muy versátil, eficiente y práctico para ser empleado en la aplicación que posteriormente será detallada.

Algunas de estas características se muestran a continuación:

- Soporta modo de comunicación serial, posee dos pines para ello.
- Amplia memoria para datos y programa.
- Memoria reprogramable: La memoria en este PIC es la que se denomina FLASH; este tipo de memoria se puede borrar electrónicamente (esto corresponde a la "F" en el modelo).
- Set de instrucciones reducidas (tipo RISC), pero con las instrucciones necesarias para facilitar su manejo.

⁶ Texto tomado de <http://www.monografias.com/trabajos18/descripcion-pic/descripcion-pic.shtml>

7.1 CARACTERÍSTICAS

En siguiente tabla de pueden observar las características más relevantes del dispositivo:

Tabla 1. Características del PIC 16F877

CARACTERÍSTICAS	16F877
Frecuencia máxima	DX-20MHz
Memoria de programa flash palabra de 14 bits	8KB
Posiciones RAM de datos	368
Posiciones EEPROM de datos	256
Puertos E/S	A,B,C,D,E
Número de pines	40
Interrupciones	14
Timers	3
Módulos CCP	2
Comunicaciones Serie	MSSP, USART
Comunicaciones paralelo	PSP
Líneas de entrada de CAD de 10 bits	8
Juego de instrucciones	35 Instrucciones
Longitud de la instrucción	14 bits
Arquitectura	Harvard
CPU	Risc
Canales Pwm	2
Pila Harware	-
Ejecución En 1 Ciclo Máquina	-

Fuente:<http://www.monografias.com/trabajos18/descripcion-pic>

7.2 DESCRIPCIÓN DE LOS PUERTOS:

▪ Puerto A:

- Puerto de e/s de 6 pines.
- RA0 y RA0 y AN0.
- RA1 y RA1 y AN1.
- RA2 y RA2, AN2 y Vref-.
- RA3 y RA3, AN3 y Vref+.
- RA4 y RA4 (Salida en colector abierto) y T0CKI (Entrada de reloj del modulo Timer0).
- RA5 y RA5, AN4 y SS (Selección esclavo para el puerto serie síncrono).

▪ **Puerto B:**

- Puerto e/s 8 pines.
- Resistencias pull-up programables.
- RB0 Interrupción externa.
- RB4-7 Interrupción por cambio de flanco.
- RB5-RB7 y RB3 programación y debugger in circuito.

▪ **Puerto C:**

- Puerto e/s de 8 pines.
- RC0 y RC0, T1OSO (Timer1 salida oscilador) y T1CKI (Entrada de reloj del modulo Timer1).
- RC1-RC2 y PWM/COMP/CAPT.
- RC1 y T1OSI (entrada osc timer1).
- RC3-4 y IIC.
- RC3-5 y SPI.
- RC6-7 y USART.

▪ **Puerto D:**

- Puerto e/s de 8 pines.
- Bus de datos en PPS (Puerto paralelo esclavo).

• **Puerto E:**

- Puerto de e/s de 3 pines.
- RE0 y RE0 y AN5 y Read de PPS.
- RE1 y RE1 y AN6 y Write de PPS.
- RE2 y RE2 y AN7 y CS de PPS.

▪ **Dispositivos periféricos:**

- Timer0: Temporizador-contador de 8 bits con preescaler de 8 bits.
- Timer1: Temporizador-contador de 16 bits con preescaler que puede incrementarse en modo sleep de forma externa por un cristal/clock.
- Timer2: Temporizador-contador de 8 bits con preescaler y postescaler.
- Dos módulos de Captura, Comparación, PWM (Modulación de Anchura de Impulsos).
- Conversor A/D de 10 bits.
- Puerto Serie Síncrono Master (MSSP) con SPI e I²C (Master/Slave).
- USART/SCI (Universal Synchronous Asynchronous Receiver Transmitter) con 9 bit.
- Puerta Paralela Esclava (PSP) solo en encapsulados con 40 pines.

8. LCD

⁷Solamente permiten visualizar mensajes cortos de texto. Existen algunos modelos estandarizados en la industria, en función de su tamaño medido en número de líneas y columnas de texto. Existen modelos de una, dos y cuatro filas únicamente. El número de columnas típico es de ocho, dieciséis, veinte y cuarenta caracteres.

Figura 9. Display LCD de texto de 16x2 con controlador HD44780.



Fuente: <http://es.wikipedia.org/wiki/LCD>

El controlador **Hitachi HD44780** se ha convertido en un estándar de industria cuyas especificaciones funcionales son imitadas por la mayoría de los fabricantes. Este controlador cuenta con los siguientes interfaces eléctricos:

- **D0-D7:** ocho señales eléctricas que componen un bus de datos.
- **R/W:** una señal que indica si se desea leer o escribir en la pantalla (generalmente solamente se escribe).
- **RS:** una señal que indica si los datos presentes en D0-D7 corresponden bien a una instrucción, bien a sus parámetros.
- **E:** una señal para activar o desactivar la pantalla.
- **V0:** señal eléctrica para determinar el contraste de la pantalla. Generalmente en el rango de cero a cinco voltios. Cuando el voltaje es de cero voltios se obtienen los puntos más oscuros.
- **Vss y Vdd:** señales de alimentación. Generalmente a cinco voltios.

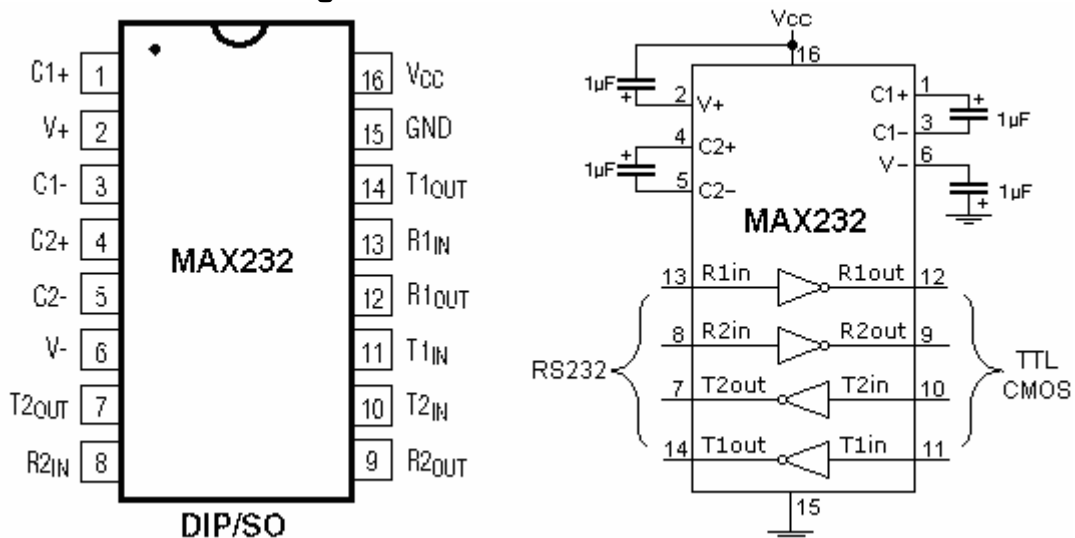
Estas señales son fácilmente controladas desde un Microcontrolador o un computador por medio de puerto paralelo.

⁷ Texto tomado de <http://es.wikipedia.org/wiki/LCD>

9. CIRCUITO INTEGRADO PARA CONVERSIÓN DE NIVELES

⁸ El MAX232 es un circuito integrado que convierte los niveles de las líneas de un puerto serie RS232 a niveles TTL y viceversa. Lo interesante es que sólo necesita una alimentación de 5V, ya que genera internamente algunas tensiones que son necesarias para el estándar RS232. Otros integrados que manejan las líneas RS232 requieren dos voltajes, +12V y -12V.

Figura 10. MAX232 - Diagrama De Pines.



Fuente: [http:// robots-argentina.com.ar/Comunicacion_max232.htm](http://robots-argentina.com.ar/Comunicacion_max232.htm)

El MAX232 soluciona la conexión necesaria para lograr comunicación entre el puerto serie de una PC y cualquier otro circuito con funcionamiento en base a señales de nivel TTL/CMOS.

El circuito integrado posee dos convertidores de nivel TTL a RS232 y otros dos que, a la inversa, convierten de RS232 a TTL.

Estos convertidores son suficientes para manejar las cuatro señales más utilizadas del puerto serie del PC, que son TX, RX, RTS y CTS.

TX es la señal de transmisión de datos, RX es la de recepción, y RTS y CTS se utilizan para establecer el protocolo para el envío y recepción de los datos.

⁸ Texto tomado de [http:// robots-argentina.com.ar/Comunicacion_max232.htm](http://robots-argentina.com.ar/Comunicacion_max232.htm)

10. ELEMENTOS SOFTWARE

10.1 MPLAB

⁹ Es un editor IDE gratuito, destinado a productos de la marca Microchip. Este editor es modular, permite seleccionar los distintos microprocesadores soportados, además de permitir la grabación de estos circuitos integrados directamente al programador.

Es un programa que corre bajo Windows y como tal, presenta las clásicas barras de programa, de menú, de herramientas de estado, etc. El ambiente MPLAB® posee editor de texto, compilador y simulación (no en tiempo real). Para comenzar un programa desde cero para luego grabarlo al μC en MPLAB® v7.XX los pasos a seguir son:

- a) Crear un nuevo archivo con extensión .ASM y nombre cualquiera.
- b) Crear un Proyecto nuevo eligiendo un nombre y ubicación.
- c) Agregar el archivo .ASM como un SOURCE FILE.
- d) Elegir el microcontrolador a utilizar desde SELECT DEVICE del menú CONFIGURE.

Una vez realizado esto, se está en condiciones de empezar a escribir el programa respetando las directivas necesarias y la sintaxis para luego compilarlo y grabarlo en el PIC

10.1.1 COMPILACIÓN DEL PROGRAMA Y CARGA AL PIC

Figura 11. Programador Picstart Plus de Microchip.



Fuente: [dettp://es.wikipedia.org/wiki/MPLAB](http://es.wikipedia.org/wiki/MPLAB)

⁹ Texto tomado [dettp://es.wikipedia.org/wiki/MPLAB](http://es.wikipedia.org/wiki/MPLAB)

Una vez escrito y depurado el programa, se procede a la compilación. Para esto, desde el menú PROYECT se elige la opción BUILD ALL (construir todo) que, si no existen errores, devolverá un mensaje como BUILD SUCCESFULL.

Los errores que muestra el compilador son del tipo sintácticos, es decir que si se llegara a tener un error, se espera que se ponga un bit en "0" y nunca pasa, se estará en un bucle infinito y el compilador compilará perfectamente porque no hay error de sintaxis.

También existen mensajes y advertencias; los mensajes pueden ser, por ejemplo, que se está trabajando en un banco de memoria que no es el bank 0, etc. Las advertencias tienen un poco más de peso, por ejemplo: el PIC seleccionado no es el mismo que esta definido en el programa, etc. En ambos casos, mensajes y advertencias, la compilación termina satisfactoriamente pero hay que tener en cuenta siempre lo que nos dicen estos para prevenir errores.

Terminada la compilación el MPLAB® nos genera un archivo de extensión *.hex* el cual es completamente entendible para el PIC. Es decir, solo resta grabarlo al PIC por medio de una interfaz como por ejemplo el programador Picstart Plus de microchip. Una vez completado esto, se alimenta al mismo y el programa ya se estará ejecutando.

10.2 MICROSOFT VISUAL BASIC.NET

Es un lenguaje de programación desarrollado por Alan Cooper. Es un lenguaje y compilador visual que desciende del lenguaje de programación BASIC. Su primera versión fue presentada en 1991 con la intención de simplificar la programación utilizando un ambiente de desarrollo completamente gráfico que facilitara la creación de interfaces gráficas y en cierta medida también la programación misma.

10.2.1 Características generales. Es un lenguaje de fácil aprendizaje con funciones para programadores principiantes y expertos, guiado por eventos, y centrado en un robusto motor de formularios que facilita el rápido desarrollo de aplicaciones. Su principal innovación, que luego fue adoptada por otros lenguajes, fue el uso de un tipo de DLL, llamado inicialmente VBX y posteriormente OCX, que permiten contener toda la funcionalidad de un control y facilitar su rápida incorporación a los formularios. Al ser un lenguaje desarrollado por Microsoft esta dotado con la capacidad de acceder a las APIs de Windows, permitiéndole usar funciones propias del núcleo del sistema operativo.

OCX. Controles propios del entorno Windows utilizados en programación, que inicialmente fueron denominados OCX (En la actualidad conservan esta extensión como parte del nombre) y ahora conocidos como los ActiveX. Son módulos separados del programa principal pero que se deben distribuir con el mismo.

Tienen la ventaja de ser portables entre unos y otros lenguajes y su contenido puede ser cualquiera, no tienen una utilidad específica para un desarrollo concreto.

Un control ActiveX es un componente que puede ser integrado por otras aplicaciones. Se almacena en un fichero OCX y es lo más parecido a cualquiera de los controles que proporciona Visual Basic, y en base a los cuales se construye una aplicación.

La tecnología ActiveX es el resultado de la evolución natural de otras tecnologías de Microsoft, que permiten al programador de Visual Basic y de otros lenguajes desarrollar rápidamente aplicaciones gracias a la reutilización de los controles o componentes.

10.2.2 Win32 APIs, APIs de Windows. Win32 es un conjunto de APIs (Application Programming Interface - Interfaz de programación de aplicaciones) disponibles para los sistemas operativos Microsoft Windows. Microsoft proporciona un SDK (kit de desarrollo de aplicaciones) en el que se incluyen la documentación y las herramientas necesarias para que los desarrolladores puedan crear sus aplicaciones y aprovechar los recursos del sistema. La primera versión de esta API se llama Win16, y sólo se utilizaba en las versiones de 16 bits de Windows.

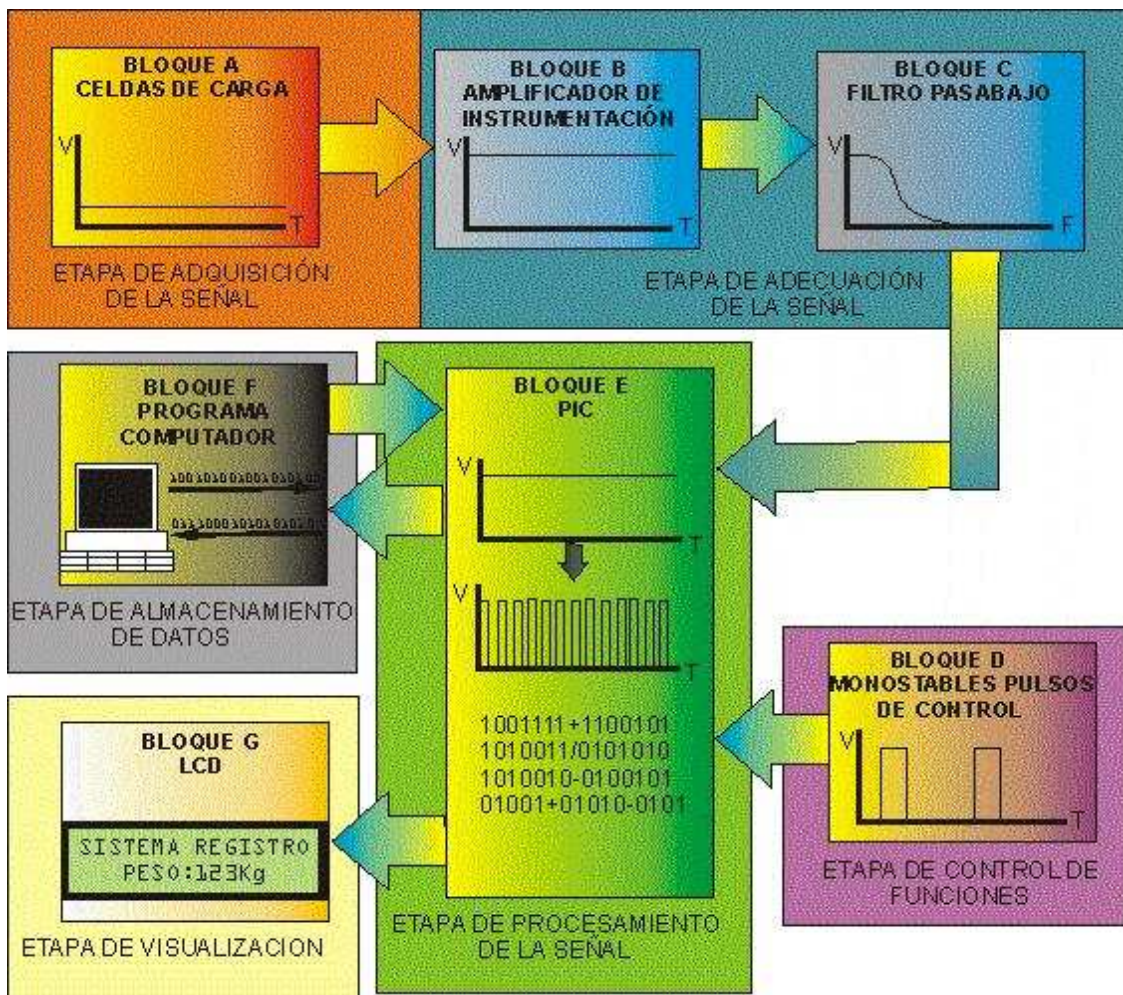
Win32 es la API de 32 bits que se utiliza en todas las versiones modernas de Microsoft Windows. Esta consiste en un conjunto de funciones C que se exponen a las aplicaciones a través de DLLs (Dynamically Linked Library - Biblioteca de enlace dinámico). Las principales DLL que forman Win32 son kernel32.dll, user32.dll.

11. DISEÑO

A continuación describiremos los pasos y los procedimientos que usamos para el desarrollo e implementación del proyecto teniendo en cuenta que se deseaba implementar un sistema electrónico unificado de registro y pesaje de ganado, de fácil manejo, eficiente, más asequible al ganadero común y sobretodo preciso y confiable.

Para una mejor comprensión del sistema en general, a continuación se muestra el diagrama en bloques del diseño propuesto e implementado.

Figura12. Diagrama en bloques Sistema de Registro y Pesaje de Ganado.



Con referencia en el diagrama en bloques expuesto podemos decir que la parte HARWARE esta comprendida en todos los bloques del sistema y la parte de SOFTWARE solo hace referencia a los bloques E y F.

11.1 HARHWARE

Es la parte del Sistema De Registro y Pesaje De Ganado encargada de adquirir, adecuar, controlar, procesar, almacenar y visualizar la señal, de acuerdo a las características de programación definidas en el diseño.

11.1.1 Bloque A: Etapa de adquisición de señal. Las especificaciones de la báscula requieren una capacidad de 2000kg, de igual manera el sistema funcionara en ambientes hostiles donde el lodo, el agua y demás factores pueden deteriorar los sensores que estarán en contacto directo con estos.

Figura 13. Etapa de adquisición de señal.

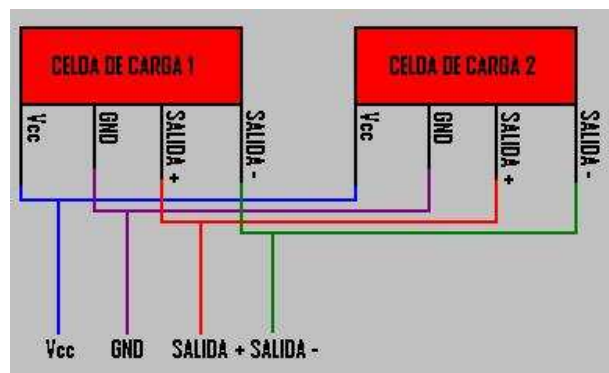


La adquisición de la señal se realizó por medio de 2 barras de pesaje de la empresa Básculas Thomas, cada una de ellas con dos celdas de carga tipo Shear Beam con salidas de 2mV/V, con protección contra polvo y agua IP67, con capacidad para 500Kg $\pm 10\%$ y alimentación máxima de 10V.

Figura 14. a) Imagen de la barra de pesaje b) Conexiones internas.



(a)



(b)

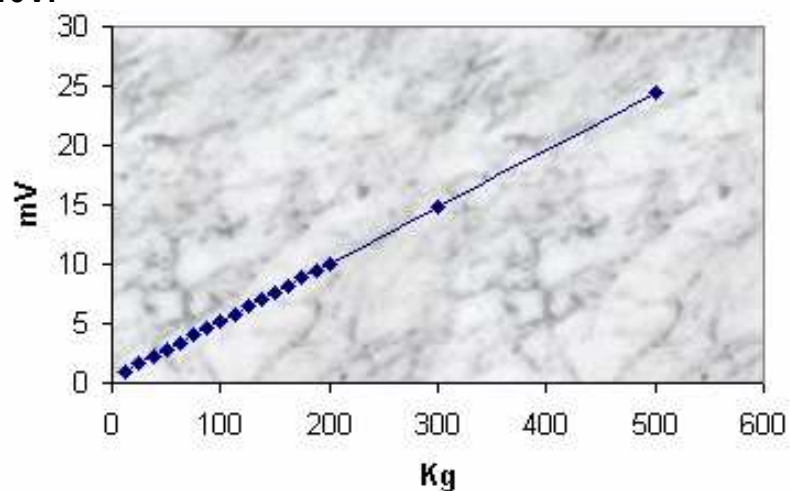
Las barras de pesajes fueron escogidas basados principalmente en tres aspectos: primero y lo más importante están diseñadas especialmente para pesaje de ganado lo cual implica que serán expuestas al ambiente ya mencionado, segundo la capacidad cumple con los requerimientos del proyecto y tercero el precio fue el mas asequible en el mercado.

Una vez seleccionado el tipo de sensor a utilizar se procedió a confirmar las características de dicho dispositivo para conocer su comportamiento obteniendo los siguientes resultados.

Tabla 2. Respuesta de una barra de pesaje de 0 – 500kg alimentada a 10V.

Peso (Kg)	mV
12,5	0,9
25	1,6
37,5	2,2
50	2,8
62,5	3,4
75	4
87,5	4,6
100	5,2
112,5	5,8
125	6,4
137,5	7
150	7,6
162,5	8,2
175	8,8
187,5	9,4
200	10
300	14,8
500	24,4

Figura 15. Curva de Respuesta de una barra de pesaje de 0 – 500kg alimentada a 10V.

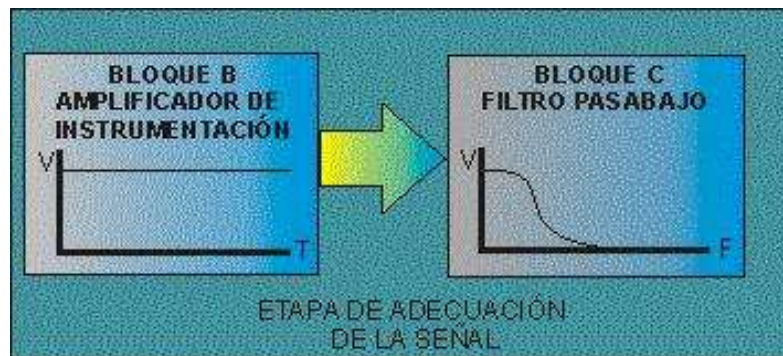


Como se puede observar a partir de 25Kg la respuesta de la barra es completamente lineal lo cual garantiza un tratamiento mas preciso de la señal sin tener que recurrir a correcciones adicionales para linealizar el sistema.

11.1.2 Etapas B y C: Adecuación de la señal. La señal obtenida tiene las características de ser una señal de magnitud muy reducida, de voltaje DC y en consecuencia es muy vulnerable a los ruidos, debido a esto se diseñaron dos etapas, una para amplificar la señal y la otra para filtrarla.

Cuando se manejan celdas de carga las señales a tratar se suman y amplifican al mismo tiempo por medio de amplificadores operacionales o amplificadores de instrumentación. En este caso se optó por tratar cada canal independientemente para aprovechar el conversor AD del PIC16F877A para poder mostrar una resolución en libras y no tener que recurrir a un conversor AD externo, subutilizando este dispositivo. La conversión se realizan en el PIC multiplexando los dos canales, esta operación dura alrededor de 100msg con lo cual se garantiza que el peso no alcance a desplazarse de una celda a la otra afectando la medición, luego se suman las dos señales digitalmente en el mismo PIC por medio de operaciones entre registros.

Figura 16. Etapa de adecuación de señal.



11.1.2.1 Bloque B: Amplificación. La señal adquirida es del orden de los mV en un rango entre 0mV – 40mV flotante la cual para poder procesarla en el PIC debe tener un rango de 0 – 5V referenciada a tierra por ello la ganancia es de $5V/40mV = 125$.

El AD620 es un amplificador de instrumentación de analog devices con un rango de ganancia de 1 a 1000 Voltaje de alimentación de $\pm 2.3V$ a $\pm 18V$, un voltaje de offset de alrededor de $50\mu V$ y lo mas importante puede amplificar y referenciar una señal de voltaje flotante a tierra, características mas que suficientes para cumplir con los requerimientos del diseño, fuera de esto tiene la ventaja por ser un amplificador de instrumentación de ajustar la ganancia por medio de una sola resistencia sin mencionar que es comercialmente muy asequible.

Para el cálculo de la ganancia se tubo en cuenta la ecuación dada por el fabricante:

$$G = 1 + (49.4/R_G)$$

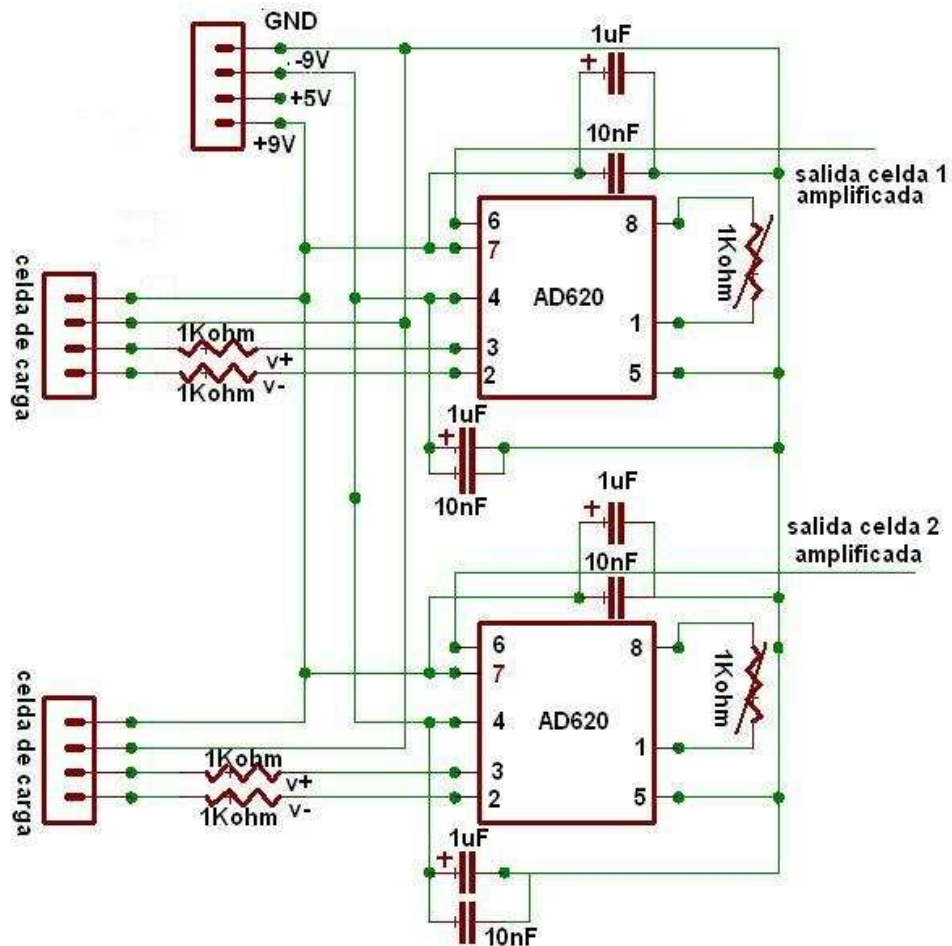
Donde G es la ganancia y R_G es la resistencia que fija la ganancia para este caso:

$$R_G = 49.4/(G-1) = 49.4/(125-1) = 398.78\Omega$$

Debido a la tolerancia de las celdas, a los factores físicos que pueden alterar sus características y a la calibración que se tiene que hacer con respecto a un patrón de pesaje la resistencia R_G se fijará por medio de un trimer.

A continuación se muestra el circuito implementado para el diseño de la etapa de amplificación.

Figura 17. Circuito Amplificador de Señal.



11.1.2.2 Bloque C: Filtro pasa bajo. La señal obtenida de la amplificación es un voltaje DC por lo tanto el filtro propuesto fue un filtro activo pasa bajo con frecuencia de corte de 3Hz de segundo orden, topología Sallen-Key, con la

finalidad de eliminar todos los ruidos de media y alta frecuencia generados por el funcionamiento del sistema y factores externos.

Se seleccionan los capacitores de modo que:

$$C1 = C2 = C \quad \text{y} \quad R1 = R2 = R3 = R$$

El valor de la frecuencia $F_c = 3\text{Hz}$ (frecuencia de corte) y fijando $C = 470\text{nF}$ y basados en la fórmula:

$$F_c = 0.0481 / RC.$$

Tenemos que:

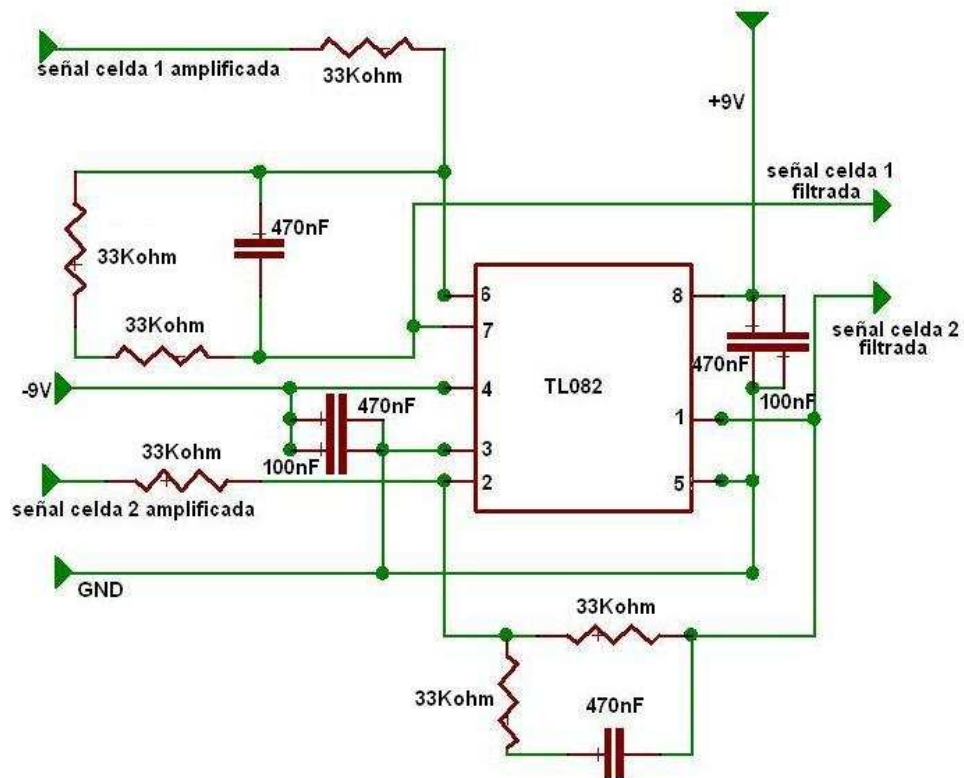
$$R = 0.0481 / (C \cdot F_c) = 0.0481 / (470\text{nF} \cdot 3\text{Hz}) = 34.1\text{K}\Omega$$

Y la ganancia del filtro (acordarse de que es un amplificador) será:

$$A_v = V_o / V_{in} = R2 / R1 = 1$$

Los filtros fueron implementados con el amplificador operacional TL082 con un offset corregido y viene con dos amplificadores en el mismo integrado, como se manejan dos canales por cada barra de pesaje este amplificador resulta ser una buena elección.

Figura 18. Circuito Filtros pasa bajos.



11.1.3 Bloque D. Etapa de control de funciones. Esta sección se diseñó con el fin de entregar un pulso a las entradas de control del PIC lo suficientemente confiables para que el programa funcione de la manera adecuada.

Figura 19. Etapa de control de funciones.



Fue implementada con temporizadores 555 en modo monoestable con un tiempo de ancho de pulso de:

$$T = 1.1 \times R1 \times C1 \text{ (en segundos).}$$

Donde:

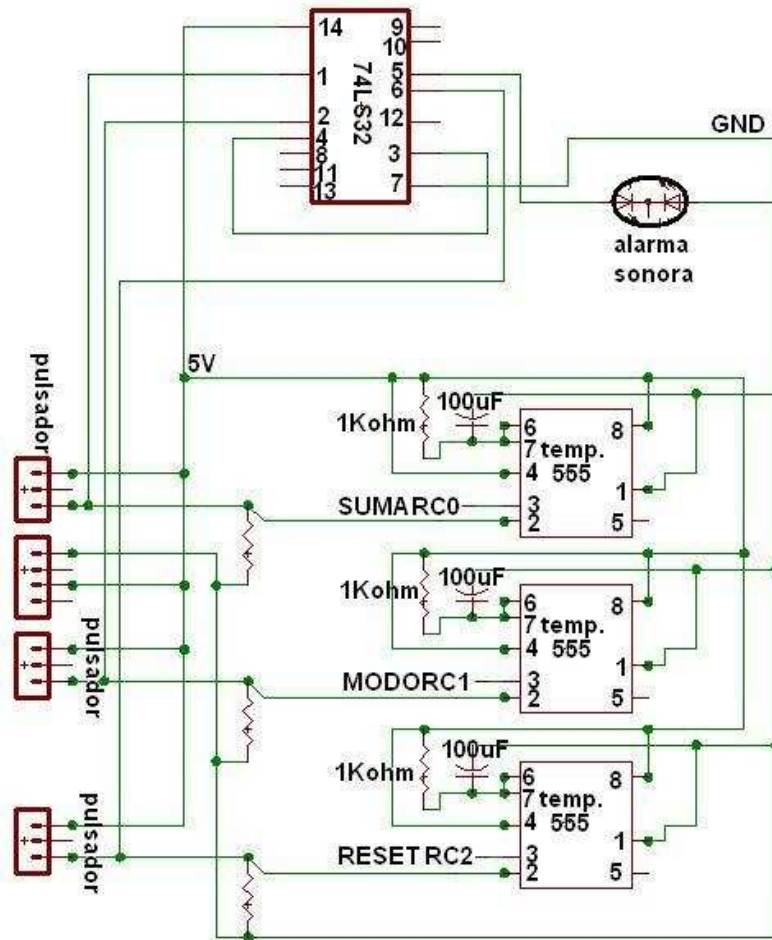
$$R1 = 1K\Omega \text{ y } C1 = 100\mu F$$

$$T = 1.1 * 1K\Omega * 100\mu F = 110ms$$

Tiempo suficiente para que el PIC desarrolle la rutina y encuentre el pulso de condición.

Adicionalmente diseñamos un sistema de alarma sonora para indicarle al usuario cuando presione un pulsador de control es decir los tres pulsos de control entran a una compuerta digital OR 74LS32 cuya salida excita dicha alarma sonora.

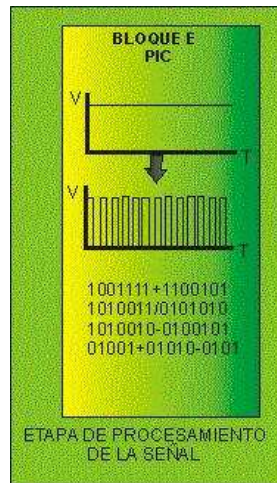
Figura 20. Circuito de control.



11.1.4 Bloque E. Etapa de procesamiento de señal. Una vez teniendo las dos señales de los sensores amplificadas y filtradas se pasa a digitalizarlas para posteriormente procesarlas y hacer las visualizaciones del caso.

Debido a que la frecuencia de muestreo en la conversión A/D para este caso no es un factor determinante, ya que la señal muestreada son voltajes DC, se utilizó para dicho fin el PIC 16F877A reduciendo carga de hardware, pues con este mismo se realiza la conversión, procesamiento, visualización y transmisión al computador, resultando bastante práctico y funcional el sistema implementado al no subutilizar dispositivos.

Figura 21. Etapa de procesamiento.



El PIC está polarizado con una fuente de 5V a tierra con una frecuencia de reloj de 4MHz proporcionada por un oscilador de cristal.

Las entradas de las señales de las celdas son aplicadas a los pines:

- **RA0 (canal de conversión A/D 1) = Señal barra de pesaje 1.**
- **RA2 (canal de conversión A/D 3) = Señal barra de pesaje 2.**

Las señales de control provenientes de los monostables están alambreadas al PIC de la siguiente manera:

- **RC0 = Pulso 1 = SUMA.**
- **RC1 = Pulso 2 = MODO (TOTAL – PROMEDIO – PESO ACTUAL).**
- **RC2 = Pulso 3 = RESET.**

Para el manejo del LCD se utilizaron los siguientes pines:

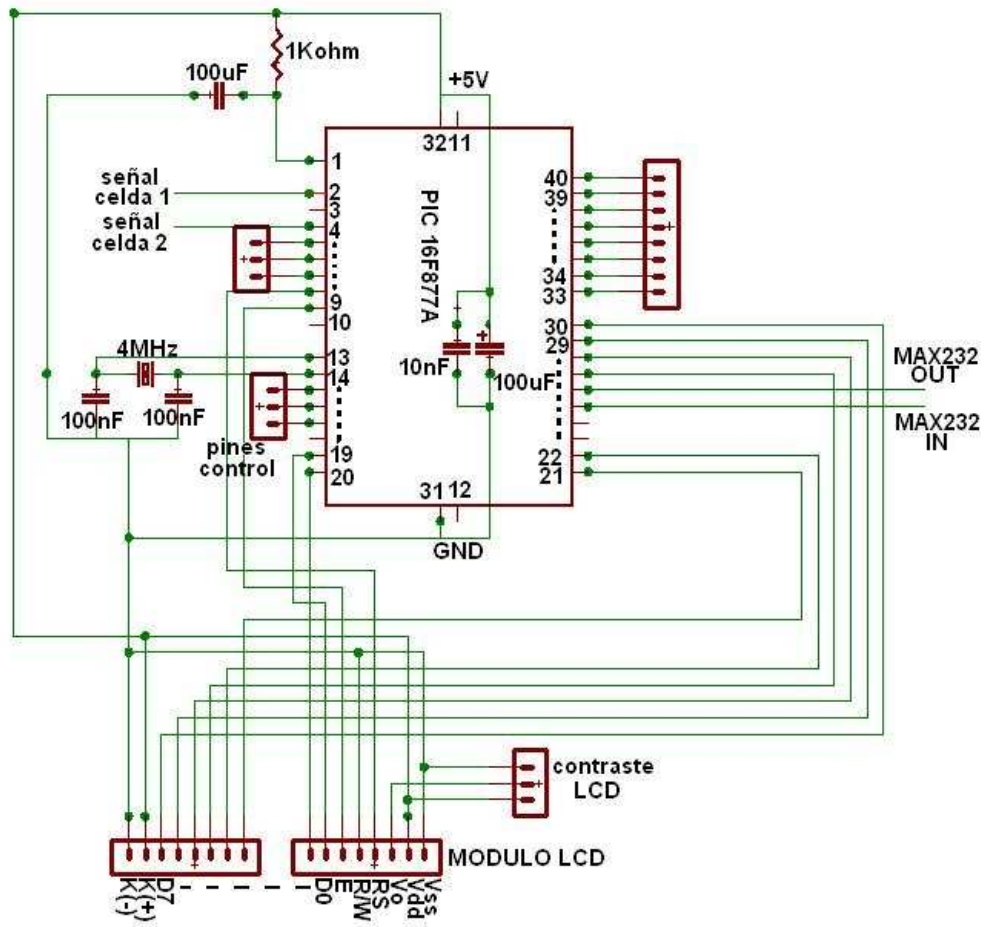
- **RE0 = RS (señal de control LCD).**
- **RE1 = E (enable LCD).**
- **RD0 – RD7 = D0 – D7 (bus de datos LCD)**

El PIC se comunica con el MAX232 por medio:

- **RC7 = RX (señal proveniente del MAX232)**
- **RC6 = TX (señal a transmitir al MAX232)**

A continuación se muestra el circuito implementado.

Figura 22. Circuito de procesamiento de señal



11.1.5 Bloque F. Etapa de almacenamiento de datos. El PIC esta configurado para funcionar de dos maneras, modo PC modo LCD, cuando trabaja en modo PC se realiza una comunicación serial.

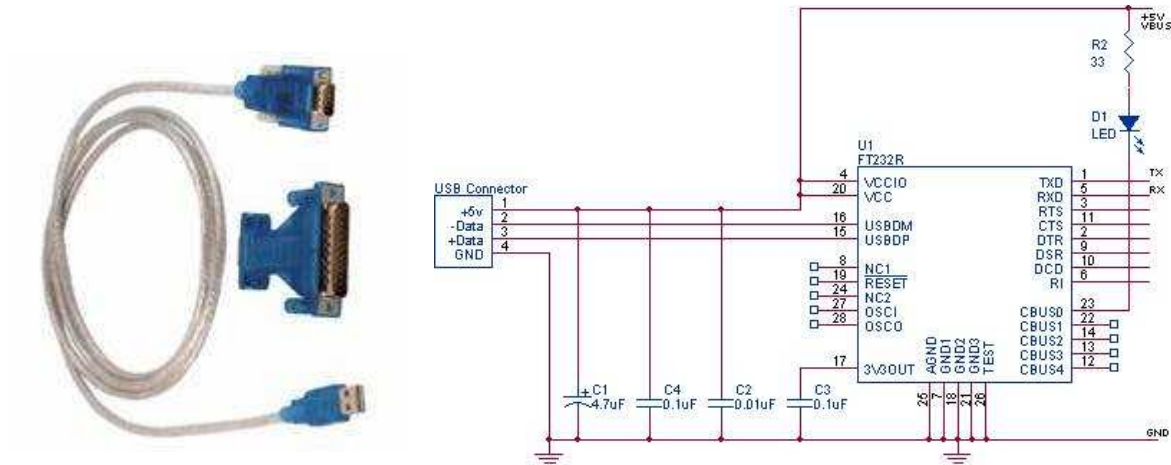
En este proyecto la idea trabajar en un portátil por que un computador de escritorio no resulta conveniente instalarlo en un corral de ganado, además la idea fundamental es que el sistema tenga movilidad.

Figura 23. Etapa de almacenamiento de datos.



Por otra parte se sabe que los portátiles actuales solo manejan puerto USB, razón por la cual se optó por el cable DKU-5 que es un adaptador de RS232 a USB.

Figura 24. Cable DKU-5 – diagrama del circuito



Fuente: http://www.hosenose.com/toystore/images_small/usb-rs232.gif

Este cable tiene la característica de crear un puerto virtual USB para esto se tiene que instalar el controlador en el portátil que se use.

11.1.6 Bloque G. Etapa de visualización.

Figura 25. Etapa de visualización.



El PIC seleccionado en modo LCD permite el uso de la báscula sin necesidad del PC, para esto se comunica con un LCD YM202-1 con 2 líneas de visualización de 20 caracteres cada una.

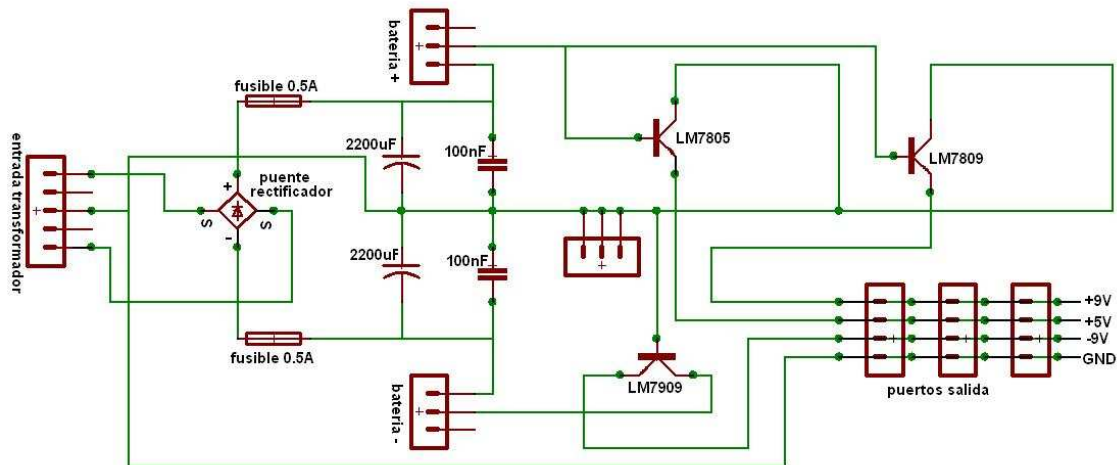
Figura 26. LCD YM202-1



Fuente: <http://www.pi4zlb.nl/Zelfbouw/CWreader/LCD-CQ.JPG>

11.1.7 Fuentes de alimentación. Los anteriores circuitos necesitan ser alimentados y polarizados con diferentes valores de voltaje, tal como se aprecian en los diagramas circuitales de cada etapa. Por esta razón se diseñaron 3 fuentes de voltajes con valores de +9V, +5V y -9V, teniendo en cuenta que la regulación de voltaje sea buena ya que en el caso de las celdas de carga es vital importancia un voltaje fijo pues si este varía el sistema total no funcionará de una forma precisa.

Figura 27. Diagrama circuital fuentes de voltaje.

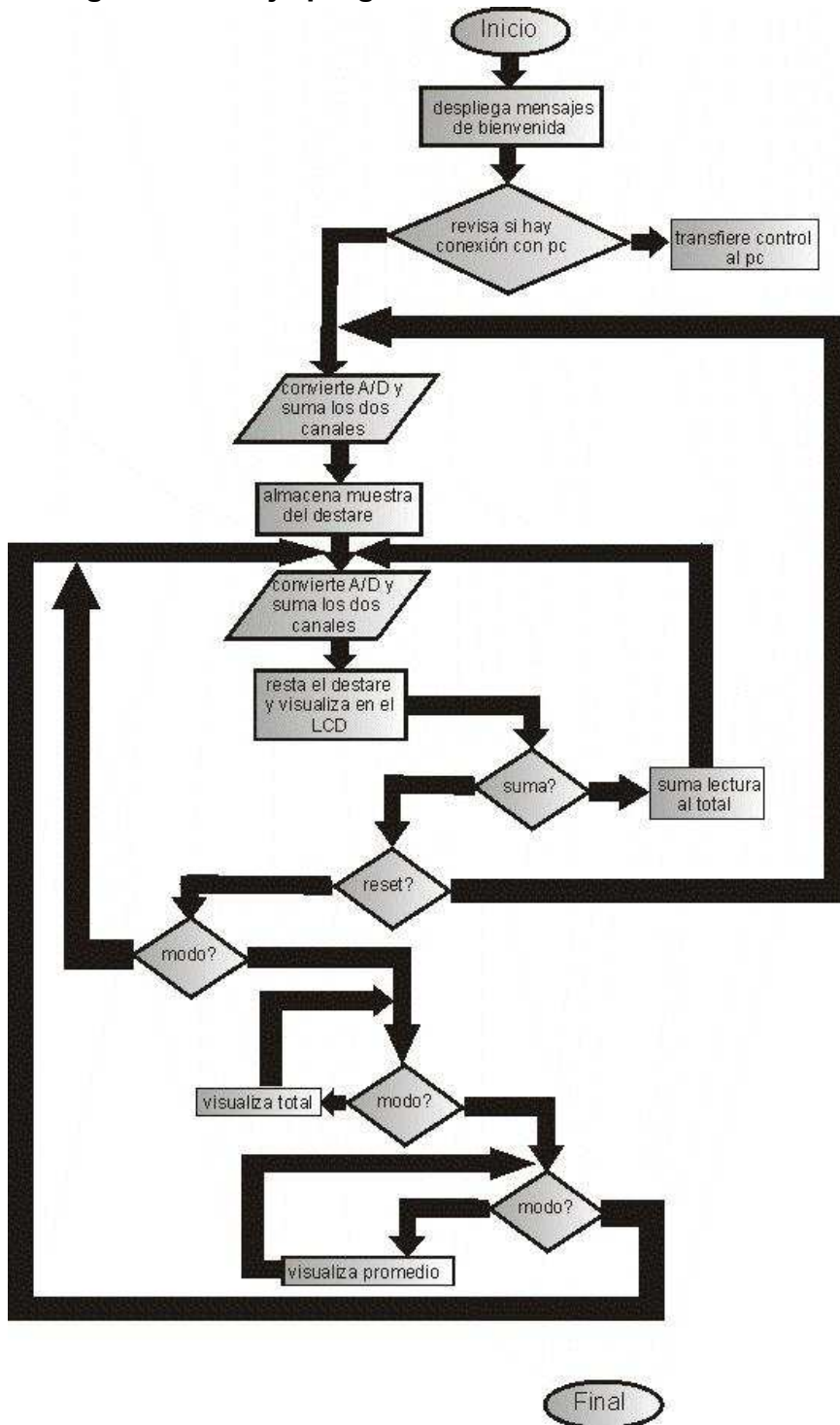


Estas fuentes se implementaron con reguladores LM7805 (+5V), LM7809 (+9V) y LM7909 (-9V), de la mejor calidad posible para tener una óptima regulación.

11.2 SOFTWARE

El Sistema de Registro y Pesaje de Ganado cuenta con dos modos de trabajo, modo PC, y modo LCD, a continuación se muestra el diagrama de flujo del programa del PIC16F877A que fue desarrollado para este proyecto el cual se explicará paso a paso más detalladamente.

Figura 28. Diagrama de flujo programa PIC 16F877A



Al encender el dispositivo, este despliega en un LCD un mensaje de bienvenida:

**“SISTEMA ELECTRONICO
DE PESAJE DE GANADO”**

**”UNIVERSIDAD SURCOLOMBIANA
ING ELECTRONICA”**

Este mensaje y todo lo que se muestra en el LCD se despliega manejando los puertos E y D del PIC16F877A con transmisión a 8bits y en modo de solo lectura (ver código fuente Anexo 1).

Al continuar el programa despliega el mensaje:

**“PESO ACTUAL
Conectando con PC”**

En este punto es donde el usuario decide el modo en que trabajar el sistema.

12.2.1 Modo LCD. Este modo se selecciona por defecto cuando el sistema no encuentra conexión con el PC, se optó por establecer este modo para situaciones en las cuales el usuario no cuenta con un computador o simplemente quiere llevar los registros a mano ya sea por que va a pesar poco ganado o simplemente por practicidad del proceso.

Una vez seleccionado este modo el LCD el PIC realiza la función de conversión A/D, que consiste en tomar los dos canales A0 (señal barra de pesaje 1) y A2 (señal barra de pesaje 2) consecutivamente para luego ser sumados y almacenados en los registros BAJO y ALTO (ver código fuente Anexo 1). Esta primer medición se almacena en el registro DESTARE y representa el peso de la plataforma el cual tiene que ser descontado en los pesajes para medir el peso neto del animal ya que:

Peso Total = peso plataforma + peso animal

Luego de tener el destare se despliega en la pantalla del LCD el mensaje:

**“PESO ACTUAL:
N:001 000000,0 Kg”**

Que nos indica que el sistema esta listo para empezar una medición.

El programa empieza a hacer un ciclo de conversión, suma de canales, resta de destare, visualización peso, testeo de pulsos de control (Pines RC0 = SUMA, RC1 = MODOS y RC2 = RESET.), y vuelve a iniciar.

Mientras el programa esta ejecutándose el usuario por medio de los pulsos de control puede realizar las siguientes acciones.

- Si se oprime el pulsador SUMA el peso que se este visualizando en ese momento será acumulado en los registros TOTALMAX, TOTALALTO y TOTALBAJO así como el regístrito NUMCICLOS que nos indica el total de animales pesados, se incrementará en 1.
- Si se oprime el pulsador MODO por primera vez se visualizara en la pantalla el total acumulado (registros TOTALMAX, TOTALALTO y TOTALBAJO) y el número de animales que se han pesado hasta el momento (NUMCICLOS-1), indefinidamente esperando que el usuario vuelva a oprimir MODO.

**“PESO ACUMULADO:
N:003 0000700,0 Kg”**

- Si se oprime por segunda vez el pulsador MODO se visualizara en la pantalla el peso promedio (registros TOTALMAX, TOTALALTO y TOTALBAJO / NUMCICLOS-1) el número de animales, indefinidamente esperando que el usuario vuelva a oprimir MODO.
- Si se oprime por tercera vez pulsador MODO el sistema vuelve a visualizar:

**“PESO ACTUAL:
N:001 0000000,0 Kg”**

Retornando al ciclo inicial.

- Estando en esta pantalla el usuario puede reiniciar el sistema pulsando el botón de RESET, borrando toda la información almacenada.

12.2.2 Modo PC. Como se había mencionado antes, el PIC16F877A opta automáticamente por este modo si al iniciar el dispositivo encuentra conexión con el PC, esta “conexión” radica en el hecho que el PIC envía un dato (C0 en este caso) y da un tiempo de espera para una respuesta del PC (A0), si esta respuesta no llega, el PIC pasa a modo LCD, pero si por lo contrario la recibe, entonces inmediatamente transfiere el control de la báscula al PC.

La aplicación de comunicación con el computador esta desarrollada en Visual Basic.Net 2005 y básicamente lo que hace es establecer la comunicación entre PIC y PC, administrar la base de datos y visualizar lo que el usuario requiera. Se escogió este lenguaje de programación por los elementos que posee para manejar el puerto serial y sobretodo la gestión de base de datos que es punto clave en este proyecto.

La base de datos esta hecha en Access 2003 esta constituida por cinco columnas sin llave principal, dichas columnas son:

- **lote:** campo numérico para especificar el número del lote la que pertenece un animal.
- **número:** campo numérico que indica el número del animal dentro del lote.
- **peso:** campo numérico donde se almacena el peso del animal registrado.
- **fecha:** campo alfanumérico para especificar la fecha del pesaje en cualquier formato.
- **observaciones:** campo de alfanumérico que almacena observaciones que el usuario crea convenientes, ejemplo: nombre del dueño del lote, origen del lote, etc.

La aplicación esta compilada de manera que genere un instalador con lo cual queda empaquetada lista para usar sin necesidad de tener el Visual Studio.Net 2005 instalado en nuestro computador, al correr el instalador se tiene la siguiente secuencia:

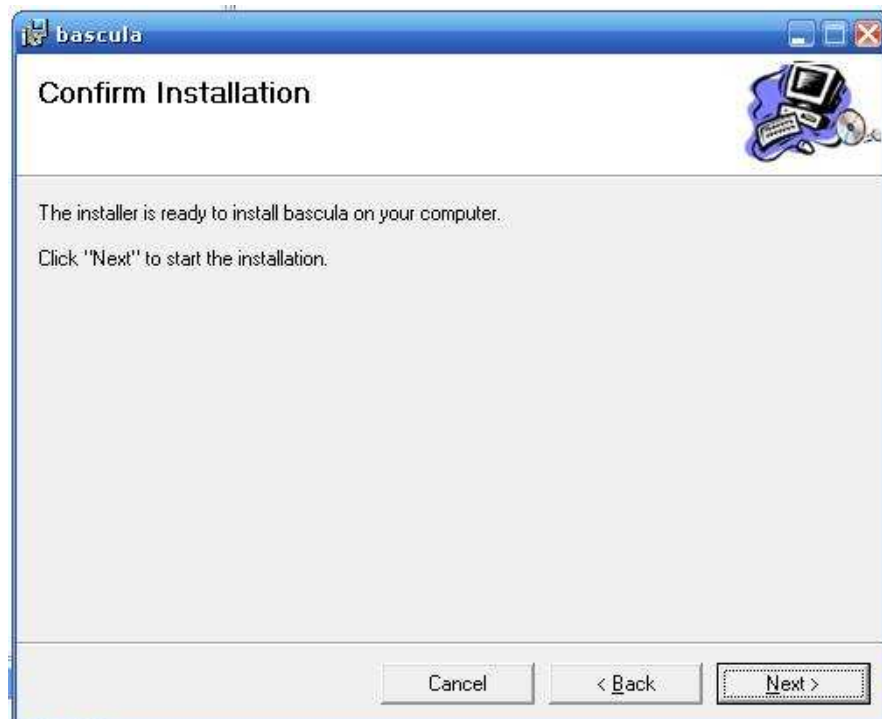
Figura 29. Inicio de instalación.



Figura 30. Selección de carpeta de instalación.



Figura 31. Confirmación de la instalación.



Una vez instalada la aplicación se puede proceder a utilizarla por medio del ejecutable que se genera en la carpeta de instalación, la pantalla del programa muestra datos y botones de función que se muestran y explican a continuación:

Figura 32. Pantalla interfase del programa Sistema de Registro y Pesaje de Ganado.



1. **ListBox1 Buscar por...:** sirve para seleccionar el criterio de búsqueda N° 1 (el menú se compone de: lote, número, peso, fecha y observaciones, que son las columnas de la base de datos).
2. **TextBox1:** caja de texto donde se digita la información a discriminar.
3. **CheckBox1:** al seleccionarse, la consulta se realiza según dos criterios de búsqueda, si no esta seleccionado, la consulta se realiza solo con un criterio de búsqueda.
4. **ListBox2 Buscar por...:** cumple la misma función que le ListBox1 pero se activa solo cuando el CheckBox1 esta seleccionado.
5. **TextBox2:** cumple la misma función que le TextBox1 pero se activa solo cuando el CheckBox1 esta seleccionado.
6. **Botón Mostrar datos:** inicia la consulta según los criterios especificados en los ListBoxes y los TextBoxs.
7. **ListView:** es una tabla que muestra los datos obtenidos de la consulta, esta constituida por las cinco columnas de la base de datos: lote, numero, peso, fecha y observaciones.

8. **TextBox Total pesado:** muestra el total acumulado de los registros que se muestren en el ListView provenientes de la consulta.
9. **TextBox Número de vacas:** muestra el número de registros que se muestren en el ListView provenientes de la consulta.
10. **TextBox Promedio:** muestra el promedio de pesaje de los registros que se muestren en el ListView provenientes de la consulta.
11. **TextBox Máximo:** muestra el peso máximo almacenado en los registros que se muestren en el ListView provenientes de la consulta.
12. **TextBox Mínimo:** muestra el peso mínimo almacenado en los registros que se muestren en el ListView provenientes de la consulta.
13. **Botón Conectar con báscula:** despliega el formulario de conexión a la báscula, para realizar la comunicación serial.
14. **Botón Iniciar mediciones:** comienza el bucle de visualización del peso en el Label1 en Kg.
15. **Label1 Peso:** muestra el peso adquirido desde el PIC16F877A.
16. **Botón Iniciar nueva sesión:** activa los CheckBox: Fecha predefinida, Lote predefinido y Observación predefinida, así como limpia el ListView para empezar a registrar datos.
17. **TextBox y CheckBox Fecha predefinida:** guarda una fecha para toda la sesión ahorrando tener que escribirla cada vez que se tiene un nuevo registro.
18. **TextBox y CheckBox Lote predefinido:** guarda un número de lote para toda la sesión ahorrando tener que escribirla cada vez que se tiene un nuevo registro.
19. **TextBox y CheckBox Observación predefinida:** guarda una observación para toda la sesión ahorrando tener que escribirla cada vez que se tiene un nuevo registro.
20. **Botón Nuevo registro:** despliega el formulario de Nuevo registro, para almacenar un nuevo elemento en la base de datos.
21. **Botón Guardar:** guarda los últimos registros realizados, pues la administración de la base de datos es offline: se carga, se modifica y se guarda.
22. **Botón Cerrar:** cierra la aplicación.

Ahora de manera más práctica e ilustrada se mostrará el manejo de la aplicación comenzando por la búsqueda de un lote pesado en una fecha por ejemplo, primero se selecciona el criterio y se escribe la información a buscar.

Figura 33. Primer criterio de búsqueda.

SISTEMA DE REGISTRO Y PESAJE DE GANADO

Buscar por... [peso] y... [] Mostrar datos

lote	numero	peso	fecha	observaciones
1	1	123	12-12-07	juan
1	2	134	12-12-07	juan
1	3	243	12-12-07	juan
1	4	213	12-12-07	juan
1	5	432	12-12-07	juan

Conectar con Bascula
 Iniciar mediciones
 Iniciar nueva sesion
 Fecha predefinida
 Lote predefinido
 Observacion predefinida

Peso actual: 0

Nuevo registro Guardar Cerrar

Luego se selecciona el CheckBox Y, para enlazar la búsqueda a dos criterios y se escoge el segundo criterio escribiendo luego la información a buscar.

Figura 34. Segundo criterio de búsqueda.

SISTEMA DE REGISTRO Y PESAJE DE GANADO

lote [1] y... [] Buscar por... [fecha] Mostrar datos

lote	numero	peso	fecha	observaciones
1	1	123	12-12-07	juan
1	2	134	12-12-07	juan
1	3	243	12-12-07	juan
1	4	213	12-12-07	juan
1	5	432	12-12-07	juan

Total pesado
 Numero de vacas
 Promedio
 Máximo
 Mínimo

Conectar con Bascula
 Iniciar mediciones
 Iniciar nueva sesion
 Fecha predefinida
 Lote predefinido
 Observacion predefinida

Peso actual: 0

Nuevo registro Guardar Cerrar

Luego al pulsar el botón Mostrar datos, se realiza la búsqueda y se despliegan los resultados obtenidos, así como los datos adicionales de Total pesado, Número de vacas, Promedio, Máximo y Mínimo.

Figura 35. Resultados de la consulta.

The screenshot shows the application interface with the following data:

Search criteria: lote: 1, fecha: 12-12-07. Button: **Mostrar datos**

Summary statistics:

- Total pesado: 1145 Kg
- Numero de vacas: 5
- Promedio: 229 Kg
- Máximo: 432 Kg
- Mínimo: 123 Kg

Table of results:

lote	numero	peso	fecha	observaciones
1	1	123	12-12-07	juan
1	2	134	12-12-07	juan
1	3	243	12-12-07	juan
1	4	213	12-12-07	juan
1	5	432	12-12-07	juan

Additional interface elements: **Conectar con Bascula**, **Iniciar mediciones** (Peso actual: 0), **Iniciar nueva sesion**, checkboxes for **Fecha predefinida**, **Lote predefinido**, and **Observacion predefinida**. Buttons at the bottom: **Nuevo registro**, **Guardar**, **Cerrar**.

En el caso para el cual la consulta no arroje resultados se tendrá el siguiente mensaje:

Figura 36. Consulta nula.

The screenshot shows the application interface with the following data:

Search criteria: numero: 13123123, Buscar por... Button: **Mostrar datos**

Summary statistics (all zeros):

- Total pesado: 0
- Numero de vacas: 0
- Promedio: 0
- Máximo: 0
- Mínimo: 0

Table of results: Empty table.

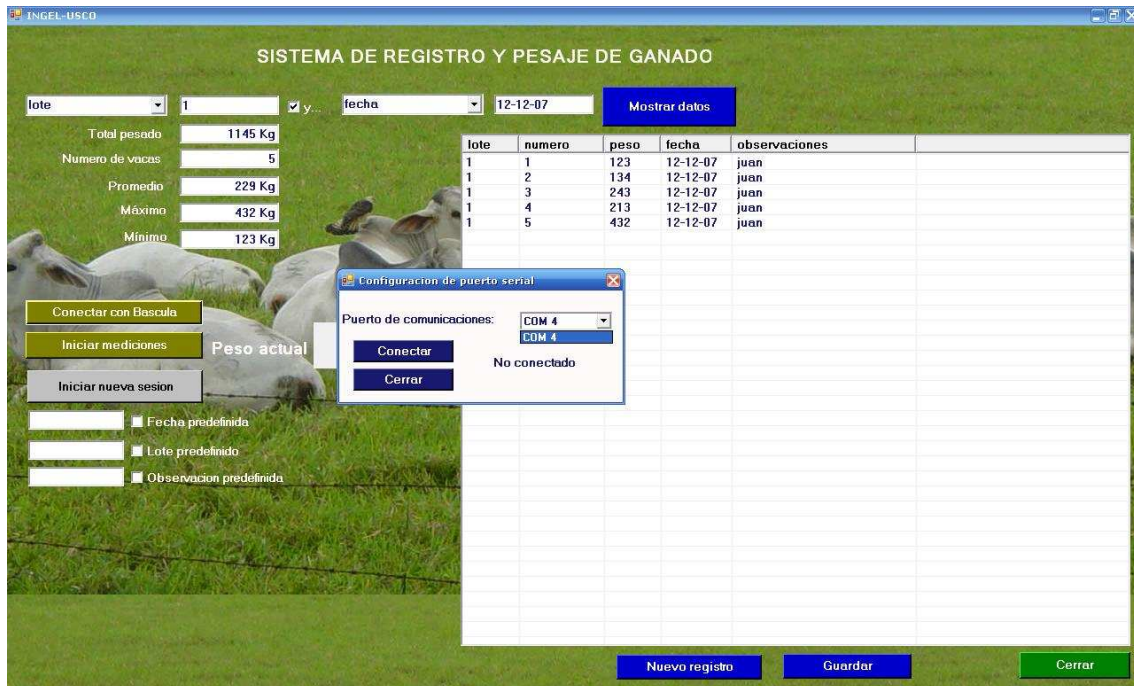
Dialog box 'PlantillaBas' message: "No hay registros que cumplan los criterios de búsqueda, Intente nuevamente." Button: **Aceptar**

Additional interface elements: **Conectar con Bascula**, **Iniciar mediciones** (Peso actual: 0), **Iniciar nueva sesion**, checkboxes for **Fecha predefinida**, **Lote predefinido**, and **Observacion predefinida**. Buttons at the bottom: **Nuevo registro**, **Guardar**, **Cerrar**.

Las consultas se pueden realizar sin necesidad de estar conectado a la báscula, pero el pesaje y el registro de nuevos datos deben realizarse con la conexión a la misma establecida, para realizar la conexión el proceso es el siguiente:

Primero se oprime el botón Conectar con báscula el cual nos despliega la siguiente pantalla:

Figura 37. Pantalla conexión con báscula.



En este formulario se procede a escoger el puerto a utilizar (COM2, COM3,...) que automáticamente se generan en el ListBox de acuerdo a los puertos disponibles para la conexión, una vez escogido, se presiona el botón Conectar, inmediatamente se enciende la báscula para que el PIC16F877A establezca la comunicación con el PC, una vez desaparece el formulario de Conectar, esto indica que la conexión es exitosa y se procede a pulsar el botón Iniciar mediciones para visualizar el peso el Label1 Peso.

Ahora si lo que se va a realizar es una sesión de pesaje, se pulsa el botón Iniciar nueva sesión y se llenan los datos correspondientes a Fecha predefinida, Lote predefinido y Observación predefinida, así:

13. RESULTADOS

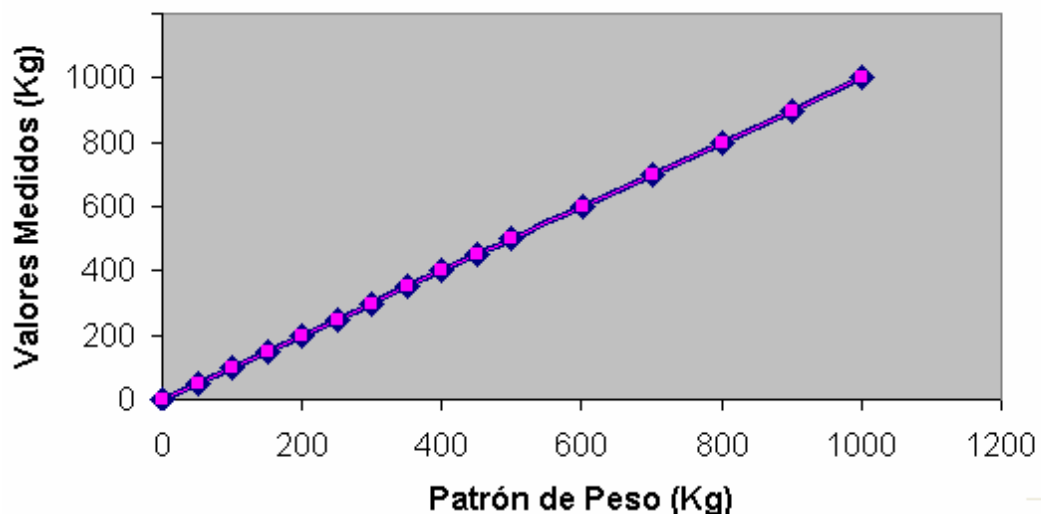
Al concluir el montaje de la báscula y la aplicación software, el siguiente paso fue la calibración la cual se realizó de acuerdo a una báscula comercial FX15 de Iconix con capacidad para 2000 Kg, de propiedad del Fondo Ganadero Del Huila con certificación del departamento de metrología de la Superintendencia De Industria y Comercio según la norma técnica NTC – 2031:2002 que regula las especificaciones para básculas de 0 a 5 toneladas (básculas clases III y IV).

Obteniendo los siguientes resultados.

Tabla 3. Comparación de las mediciones de la báscula del Fondo Ganadero del Huila y la báscula diseñada.

Patrón de Peso (Kg)	Valor Medido Báscula Fondo Ganadero (Kg)	Valor Medido Báscula Diseñada (Kg)
0	0	0
50	51	50
100	100	100,5
150	150	150
200	200	200
250	250	250
300	300	300
350	350	350
400	400	400
450	450	450
500	500	500
600	600	599,5
700	700	699
800	800	800
900	900	900
1000	1000	1000

Figura 41. Grafica de las mediciones de la báscula del Fondo Ganadero del Huila y la báscula diseñada.



14. CONCLUSIONES

- Esta báscula resulta de forma general universal, pudiendo ser utilizada para mediciones en cualquier tipo de rango, según el sensor (celda de carga) utilizada y realizando solo unos pequeños ajustes en la ganancia de los amplificadores de instrumentación así como la modificación del programa del PIC en tan solo una línea. Esto se puede dar por el hecho que la señal de la celda es adquirida y el amplificador de instrumentación lo que hace acomodar esta al rango de conversión del PIC, este último la convierte y la procesa, para luego visualizarla, ahí es la modificación del programa del PIC, cambiar las letras Kg. del LCD por la unidad que estemos manejando (g ó t por ejemplo).
- La báscula presentada cumple con los requisitos técnicos mínimos exigidos para equipos de esta categoría, con lo cual se puede garantizar la confiabilidad en los resultados obtenidos al realizar sesiones de pesaje.
- Los elementos utilizados para la fabricación de la báscula son bastante económicos y se consiguen fácilmente (a excepción de las celdas), lo que simplifica el montaje de estos equipos en masa y también ofrece ventajas en el caso de la reparación.
- La interfaz de la báscula en el computador resulta bastante sencilla y práctica para el usuario, lo que provoca una gran aceptación y facilidad en la operación del equipo, pues el ambiente donde se va a desarrollar así lo amerita.
- El equipo desarrollado ofrece las ventajas de los equipos tradicionales y otras tales como interface de usuario más amigable, identificación por medio de código de barras, precio mas asequible al ganadero común, que lo hacen una excelente opción para la realización de pesajes de todo tipo.
- Se cumplió con los objetivos fundamentales del proyecto: aumentar funcionalidad y practicidad y la vez minimizar costos, pues una báscula comercial de características similares cuesta alrededor de \$4'000.000 comparada con la diseñada en este proyecto que cuesta alrededor de \$3'000.000.

15. RECOMENDACIONES

- Cuando se trabajen con celdas de cargas utilizar reguladores de voltajes de buena calidad que mantengan un voltaje lo más fijo posible para evitar mediciones erradas.
- Desarrollar la aplicación de software del PC en otros sistemas operativos ejemplo windows pocket o móvil (PAD) para hacer mas versátil y funcional el sistema y por lo tanto mas atractivo a el consumidor.
- Desarrollar una interface de comunicación tipo RF o Bluetooth entre la báscula y el sistema administrador de base de datos con el propósito de dar más movilidad e independencia al sistema.
- Ampliar el software desarrollado mediante la adición de nuevos variables de registro para la base con el fin que el usuario tenga un historial más amplio y detallado de sus animales.

16. FUENTES DE CONSULTA

16.1. BIBLIOGRAFIA.

ANGULO USATEGUI José M and ROMERO YESA Susana,
Microcontroladores <<PIC>> Diseño Práctico de Aplicaciones. MC Graw Hill.

16.2. MEDIOS TELEINFOMÁTICOS.

<http://es.wikipedia.org>

<http://www.forosdeelectronica.com>

www.unicrom.com

<http://www.monografias.com>

<http://www.robots-argentina.com.ar>

<http://www.hosenose.com>

<http://www.pi4zlb.nl>

<http://msdn2.microsoft.com>

<http://www.microchip.com>

ANEXO A. PROGRAMA PARA EL PIC 16F877A

LIST p=16F877A
include P16F877A.INC
RADIX HEX

Entry	EQU	0X23		
Inc		EQU	0X24	
MEDIO		EQU	0X25	;DICE SI HAY MEDIO KILOGRAMO.
BAJO		EQU	0X26	;PARTE BAJA DEL DATO.
ALTO		EQU	0X27	;PARTE MEDIA DEL DATO.
BCD0		EQU	0X28	;ULTIMO PAR DE DIGITOS EN BCD.
BCD1		EQU	0X29	;PENULTIMO PAR DE DIGITOS EN BCD.
BCD2		EQU	0X2A	;ANTEPENULTIMO PAR DE DIGITOS EN BCD.
BCD3		EQU	0X2B	;TRASANTEPENULTIMO PAR DE DIGITOS EN BCD.
CONTADOR		EQU	0X2C	;CONTADOR PARA CONVERSION BINTOBCD.
TMP		EQU	0X2D	;POSICION TEMPORAL CONVERSION BINTOBCD
CIFRA1	EQU	0X2E		;C.
CIFRA2	EQU	0X2F		;I.
CIFRA3	EQU	0X30		;F.
CIFRA4	EQU	0X31		;R.
CIFRA5	EQU	0X32		;A.
CIFRA6	EQU	0X33		;S L.
CIFRA7	EQU	0X34		;C.
CIFRA8	EQU	0X35		;D.
CONTA1	EQU	0X36		;CONTADOR DEL RETARDO.
CONTA2	EQU	0X37		;COTADOR RETARDO ENTRE CARACTERES.
CONTA3	EQU	0X38		;CONTADOR DE CARACTERES LCD.
CONTA4	EQU	0X39		;CONTADOR SELECCIONAR CARACTERES.
VARIABLES				
DECIMA	EQU	0X3A		;GUARDA LA DECIMA DEL PROMEDIO.
DESTAL	EQU	0X3B		;GUARDA EL PESO DE LA PLATAFORMA.
TOTALALTO	EQU	0X3C		;PARTE MEDIA DEL TOTAL PESADO.
TOTALBAJO	EQU	0X3D		;PARTE BAJA DEL TOTAL PESADO.
TOTALMAX	EQU	0X3E		;PARTE ALTA DEL TOTAL PESADO.
NUMCICLOS	EQU	0X3F		;NUMERO DE PESAJES PARA PROMEDIO.
RDEC	EQU	0X40		;AUXILIAR DE DECIMALES.
COCI	EQU	0X41		;RESIDUO DE LA DIVISION.
RESU1	EQU	0X42		;PARTE ALTA DEL PROMEDIO TEMPORAL.
RESU2	EQU	0X43		;PARTE BAJA DEL PROMEDIO TEMPORAL.
PESOMAX	EQU	0X44		;AUXILIAR PARTE ALTA TOTAL.
PESOALTO	EQU	0X45		;AUXILIAR PARTE MEDIA TOTAL.
PESOBAJO	EQU	0X46		;AUXILIAR PARTE BAJA TOTAL.
DIVI	EQU	0X47		;AUXILIAR NUMERO DE PESAJES.
CELDA1L	EQU	0X48		;VALOR BAJO CELDA 1.
CELDA1H	EQU	0X49		;VALOR ALTO CELDA 1.
CELDA2L	EQU	0X4A		;VALOR BAJO CELDA 2.
CELDA2H	EQU	0X4B		;VALOR ALTO CELDA 2.
TRASBAJO	EQU	0X4C		;REGISTRO DE TRASMISION.
TRASALTO	EQU	0X4D		;REGISTRO DE TRASMISION.
CONTA5	EQU	0X4E		
SELECT	EQU	0X4F		;SELECCION DE TABLA.
CIFRA05	EQU	0X50		;DEFINE SI ES DATO ENTERO O NO.
MAX	EQU	0X51		;PARTE ALTA DEL DATO.
CONTAL	EQU	0X52		
MOD0	EQU	0X53		;SELECCION DE FUNCION A REALIZAR.
CIFRA1P	EQU	0X54		;CIFRA 1 DE UMEMO DE VACAS PESADAS.
CIFRA2P	EQU	0X55		;CIFRA 2 DE UMEMO DE VACAS PESADAS.
CIFRA3P	EQU	0X56		;CIFRA 3 DE UMEMO DE VACAS PESADAS.
TBAJO	EQU	0X57		;REGISTRO AUXILIAR PARA EL PROMEDIO.
TALTO	EQU	0X58		;REGISTRO AUXILIAR PARA EL PROMEDIO.
TMAX	EQU	0X59		;REGISTRO AUXILIAR PARA EL PROMEDIO.
ATOTALBAJO	EQU	0X5A		
ATOTALALTO	EQU	0X5B		
ATOTALMAX	EQU	0X5C		
NUMVACAS	EQU	0X5D		
NUMPROM	EQU	0X5E		;INDICA EL NUMERO DE DATOS A PROMEDIAR.
BPROM	EQU	0X5F		
APROM	EQU	0X60		

```

RESET  ORG          0x00
;el vector de reset es la posición 00h.
      GOTO  INICIO
;se salta al inicio del programa.
      ORG   0x10
;el programa empieza en la posición.
      GOTO  INICIO
;de memoria de programa 05H.

;TABLAS DE DESPLIEGUE

TABLAD MOVF  CONTA4,W
      ADDWFPCL,1
      RETLW "N"
      NOP
      RETLW " ."
      NOP
      MOVF  CIFRA1P,W
      RETURN
      MOVF  CIFRA2P,W
      RETURN
      MOVF  CIFRA3P,W
      RETURN
      RETLW " "
      NOP
      RETLW " "
      NOP
      MOVF  CIFRA1,W
      RETURN
      MOVF  CIFRA2,W
      RETURN
      MOVF  CIFRA3,W
      RETURN
      MOVF  CIFRA4,W
      RETURN
      MOVF  CIFRA5,W
      RETURN
      MOVF  CIFRA6,W
      RETURN
      MOVF  CIFRA7,W
      RETURN
      MOVF  CIFRA8,W
      RETURN
      RETLW " ."
      NOP
      MOVF  CIFRA05,W
      RETURN
      RETLW " "
      NOP
      RETLW "K"
;termina segunda línea.
      NOP
      RETLW "g"
      NOP
      retlw  " "

;tablas de despliegue mensaje de bienvenida.

TABLA0 MOVF  CONTA3,W
;tabla de inicio 1.
      ADDWF PCL,1
      RETLW "S"
;comienza de primera línea.
      RETLW "I"
      RETLW "S"
      RETLW "T"
      RETLW "E"
      RETLW "M"
      RETLW "A"
      RETLW " "

RETLW "E"
RETLW "L"
RETLW "E"
RETLW "C"
RETLW "T"
RETLW "R"
RETLW "O"
RETLW "N"
RETLW "I"
RETLW "C"
RETLW "O"
RETLW " "

;termina primera línea.
TABLA1 MOVF  CONTA3,W
;tabla de inicio 2.
      ADDWFPCL,1
      RETLW "D"
;comienza segunda línea.
      RETLW "E"
      RETLW " "
      RETLW "P"
      RETLW "E"
      RETLW "S"
      RETLW "A"
      RETLW "J"
      RETLW "E"
      RETLW " "
      RETLW "D"
      RETLW "E"
      RETLW " "
      RETLW "G"
      RETLW "A"
      RETLW "N"
      RETLW "A"
      RETLW "D"
      RETLW "O"
      RETLW " "
;termina segunda línea.
TABLA2 MOVF  CONTA3,W
      ADDWF PCL,1
      RETLW "U"
;comienza de primera línea.
      RETLW "N"
      RETLW "I"
      RETLW "V"
      RETLW " ."
      RETLW " "
      RETLW "S"
      RETLW "U"
      RETLW "R"
      RETLW "C"
      RETLW "O"
      RETLW "L"
      RETLW "O"
      RETLW "M"
      RETLW "B"
      RETLW "I"
      RETLW "A"
      RETLW "N"
      RETLW "A"
      RETLW " "
;termina primera línea.
TABLA3 MOVF  CONTA3,W
      ADDWFPCL,1
      RETLW " "
;comienza segunda línea.
      RETLW " "
      RETLW "I"

```

```

RETLW "N"
RETLW "G"
RETLW "."
RETLW " "
RETLW "E"
RETLW "L"
RETLW "E"
RETLW "C"
RETLW "T"
RETLW "R"
RETLW "O"
RETLW "N"
RETLW "I"
RETLW "C"
RETLW "A"
RETLW " "
RETLW " "

```

;termina segunda línea.

```

TABLA4 MOVF   CONTA3,W
      ADDWF  PCL,1
      RETLW "P"

```

;comienza de primera línea.

```

RETLW "E"
RETLW "S"
RETLW "O"
RETLW " "
RETLW "A"
RETLW "C"
RETLW "T"
RETLW "U"
RETLW "A"
RETLW "L"
RETLW " "
RETLW "."
RETLW " "
RETLW " "
RETLW " "
RETLW " "
RETLW " "
RETLW " "
RETLW " "
RETLW " "
RETLW " "

```

;termina primera línea.

```

TABLA5 MOVF   CONTA3,W
      ADDWFPCL,1
      RETLW "P"

```

;comienza de primera línea.

```

RETLW "E"
RETLW "S"
RETLW "O"
RETLW " "
RETLW "P"
RETLW "R"
RETLW "O"
RETLW "M"
RETLW "E"
RETLW "D"
RETLW "I"
RETLW "O"
RETLW " "
RETLW "."
RETLW " "
RETLW " "
RETLW " "
RETLW " "
RETLW " "
RETLW " "

```

;termina primera línea.

```

TABLA6 MOVF   CONTA3,W
      ADDWFPCL,1

```

;comienza de primera línea.

```

RETLW "P"
RETLW "E"
RETLW "S"
RETLW "O"
RETLW " "
RETLW "A"
RETLW "C"
RETLW "U"
RETLW "M"
RETLW "U"
RETLW "L"
RETLW "A"
RETLW "D"
RETLW "O"
RETLW " "
RETLW "."
RETLW " "
RETLW " "
RETLW " "
RETLW " "

```

;termina primera línea.

```

TABLA7 MOVF   CONTA3,W
      ADDWFPCL,1

```

;comienza de primera línea.

```

RETLW "C"
RETLW "o"
RETLW "n"
RETLW "e"
RETLW "c"
RETLW "t"
RETLW "a"
RETLW "n"
RETLW "d"
RETLW "o"
RETLW " "
RETLW "c"
RETLW "o"
RETLW "n"
RETLW " "
RETLW "P"
RETLW "C"
RETLW " "
RETLW " "

```

;termina primera línea.

;RUTINA DE RETARDO USADO EN LAS SEÑALES DE CONTROL Y DATOS DEL LCD.

```

RETARDO      MOVLW 0x30
              MOVWF CONTA1
DECRE DECFSZ      CONTA1,1
;decremento CONTA1 salta cuando sea cero.

```

```

          GOTO  DECRE
          RETLW 0

```

;vuelve a donde iba ejecutándose el programa, con W=0.

```

RETARDATE      MOVLW 0x0C
              MOVWF CONTA5
DECRE0 MOVLW 0x90
              MOVWF CONTA1

```

```

DECRE1 DECFSZ CONTA1,1
;decrementa CONTA1 salta cuando sea cero

```

```

GOTO  DECRE1
decfsz  CONTA5,f
GOTO  DECRE0
RETLW  0
;vuelve a donde iba ejecutándose el programa, con
W=0

```

```

;RUTINA PARA LIMPIAR EL LCD.
LIMPIA      CLRf  CONTA5
;limpia el contador de caracteres.
LIMPI      MOVLW  " "
; carga con espacios w.
CALL      DATO
;envía carácter " " a display de modo que quede con "
".
INCF      CONTA5,1
;incrementa el registro CONTA3.
MOVLW    0x50
;hay 80 locaciones de ram en el display
XORWF    CONTA5,W
;comprara si ya llego a 50h=80d.
BTFSS    STATUS,Z
;envió los 80 caracteres? salte si z=1 (W=0).
GOTO     LIMPI
;si no repita el ciclo de envío de carácter " ".
RETLW    0
;vuelve a donde iba ejecutándose el programa, con
w=0.

```

```

;RUTINA DE CONTROL PARA ENVIAR DATO DE CONTROL O CARACTER, EL PIN RS DEL DISPLAY CONTROLA SI LO QUE SE CARGA EN EL PUERTO ES CONTROL O UN DATO, DE MODO QUE RS=1 DATO Y RS=0 CONTROL.
CONTROL    BCF      PORTE,0
;para enviar dato de control(RS=0).
GOTO     DATO2
DATO       BSF      PORTE,0
;para enviar dato carácter (RS=1).
DATO2     MOVWF   PORTD
; carga el puerto del LCD con lo que hay en PORTD.
BSF      PORTE,1
;habilitar display (E=1).
CALL     RETARDO
;llamamos retardo para asegurar la lectura del puerto.
BCF      PORTE,1
;deshabilita el LCD.
RETLW    0
;vuelve a donde iba ejecutándose el programa, con
w=0.

```

```

;RUTINA DE DESTALE.
CALIBRA   BTFSS   PORTC,0
;testeo bit de destare.
RETLW    0
;si RC0=0 vuelve programa, ya hubo destare.
MOVf     BAJO,W
;si RC0=1
MOVWF    DESTALE
;almacenamos peso plataforma.
RETLW    0

```

```

;RUTINA DE CONVERSION A/D.
CONVER1   BCF      STATUS,RP1
;primera celda.

```

```

STATUS,RP0  BSF      ;BANCO 1
CLRf      ADRESL
BCF
STATUS,RP0  ;BANCO 0

CLRf      ADRESH
MOVLW    0x40
MOVWF    ADCON0
;configuración de conversión.
BCF      PIR1,6
BSF      ADCON0,0
CALL     RETARDO
CALL     RETARDO
CALL     RETARDO
BSF      ADCON0,GO
CALL     RETARDO
CALL     RETARDO

CONV1      BTFSC   ADCON0,GO
GOTO     CONV1
BSF
STATUS,RP0

;banco 1.
MOVf     ADRESL,W
BCF
STATUS,RP0

;banco 0.
MOVWF
CELDA1L
MOVf
ADRESH,W
MOVWF
CELDA1H

RETURN

;segunda celda.
CONVER2    BSF
STATUS,RP0

;banco 1
CLRf      ADRESL
BCF
STATUS,RP0

;banco 0.
CLRf      ADRESH
MOVLW    0x50
MOVWF    FADCON0
;configuración de conversión.
BCF
PIR1,6
BSF
ADCON0,0
CALL     RETARDO
CALL     RETARDO
CALL     RETARDO
BSF
ADCON0,GO
CALL     RETARDO
CALL     RETARDO
CONV2      BTFSC   ADCON0,GO
GOTO     CONV2
BSF
STATUS,RP0

;banco 1.
MOVf     ADRESL,W
BCF
STATUS,RP0

;banco 0.
MOVWF
CELDA2L
MOVf
ADRESH,W

```

```

                MOVWF
                CELDA2H
                RETURN
*****
;SUMA DE REGISTROS.
SUMA           MOVF   CELDA1L,W
                ADDWF  BAJO,f
                BTFSS  STATUS,C
                GOTO   OTRO1
                MOVLW 0X01
                ADDWF  ALTO,f
OTRO1          MOVF   CELDA1H,W
                ADDWF  ALTO,f
                RETURN

```

**;RUTINA DE CONVERSION DE BINARIO A BCD
 QUEDA EXPRESADO EN Kg.**

```

BINBCD MOVLW 0x18 ;Cargamos 24 en el
contador.
                MOVWF CONTADOR
                CLRF   BCD3
                CLRF   BCD2
                CLRF   BCD1
;inicial de las posiciones.
                CLRF   BCD0
;finales.
DESPLAZT      BCF   STATUS,C ;ponemos a
cero el carry antes de rotar.
                RLF   BAJO,F
                RLF   ALTO,F
;rotación total.
                RLF   MAX,F
;desde el byte bajo.
                RLF   BCD0,F
;hasta el byte más alto.
                RLF   BCD1,F
;de los datos finales.
                RLF   BCD2,F
                RLF   BCD3,F
                DECF  SZCONTADOR,F
;Si el contador es cero.
                GOTO  AJUSTET
;ya van 16 desplazamientos.
                RETURN
;y retornamos.
AJUSTET       MOVLW  BCD0
;Empleamos direccionamiento indirecto.
                MOVWF FSR
;para llamar al subprograma de ajuste.
                CALL  AJBCDT
;decimal de cada byte, primero con BCD0.
                MOVLW  BCD1
;Lo mismo con BCD1.
                MOVWF FSR
                CALL  AJBCDT
                MOVLW  BCD2
;Lo mismo con BCD2.
                MOVWF FSR
                CALL  AJBCDT
                MOVLW  BCD3
;Lo mismo con BCD3.
                MOVWF FSR
                CALL  AJBCDT
                GOTO  DESPLAZT
;volvemos a las rotaciones.

```

**; SUBPROGRAMA PARA AJUSTE BCD DE CADA
 BYTE**

```

AJBCDT        MOVLW 3 ;sumamos 3 a la
                posición a la que apunta FSR.
                ADDWF INDF,W
;el contenido queda en W.
                MOVWF TMP ;exploramos si en
el primer dígito.
                BTFSC  TMP,3 ;resultado es
mayor que 7.
                MOVWF INDF
;si es así lo corregimos almacenando ese valor
                MOVLW 30 ;hacemos lo mismo
con el dígito BCD superior.
                ADDWF INDF,W
                MOVWF TMP ;exploramos
sumando 30 al byte completo
                BTFSC  TMP,7
;y si el dígito superior es mayor que 7
                MOVWF INDF
;lo almacenamos para corregir
                RETLW 0
;retorno desde el subprograma AJBCD

```

**;RUTINA DE CONVERSION DE BCD A ASCII DE
 LOS DIGITOS DEL PESO.**

```

BCD2ASCII     MOVF           BCD0,W
                ANDLW 0X0F
                MOVWF CIFRA8
                MOVLW 0X30
                ADDWF CIFRA8,F
                MOVF           BCD0,W
                ANDLW 0XF0
                MOVWF CIFRA7
                SWAPF CIFRA7,F
                MOVLW 0X30
                ADDWF CIFRA7,F
                MOVF           BCD1,W
                ANDLW 0X0F
                MOVWF CIFRA6
                MOVLW 0X30
                ADDWF CIFRA6,F
                MOVF           BCD1,W
                ANDLW 0XF0
                MOVWF CIFRA5
                SWAPF CIFRA5,F
                MOVLW 0X30
                ADDWF CIFRA5,F
                MOVF           BCD2,W
                ANDLW 0X0F
                MOVWF CIFRA4
                MOVLW 0X30
                ADDWF CIFRA4,F
                MOVF           BCD2,W
                ANDLW 0XF0
                MOVWF CIFRA3
                SWAPF CIFRA3,F
                MOVLW 0X30
                ADDWF CIFRA3,F
                MOVF           BCD3,W
                ANDLW 0X0F
                MOVWF CIFRA2
                MOVLW 0X30
                ADDWF CIFRA2,F
                MOVF           BCD3,W
                ANDLW 0XF0
                MOVWF CIFRA1
                SWAPF CIFRA1,F
                MOVLW 0X30
                ADDWF CIFRA1,F
                RETLW 0

```

;RUTINA SUMA PESO TOTAL E INCREMENTO DE NUMERO DE CICLOS.

```
SUMATOTAL    MOVF    PESOBAJO,W
              ADDWF
              TOTALBAJO,f
              BTFSS   STATUS,C
              GOTO    OTROT
              MOVLW   0X01
              ADDWF
              TOTALALTO,f
              BTFSS   STATUS,C
              GOTO    OTROT
              MOVLW   0X01
              ADDWF
              TOTALMAX,f
OTROT    MOVF    PESOALTO,W
              ADDWF
              TOTALALTO,f
              BTFSS   STATUS,C
              INCF    TOTALMAX,1
              INCF    NUMCICLOS,1
              RETURN
```

;RUTINA DE VISUALIZACION DEL PESO KILOGRAMOS COMPLETO.

```
VISUAL    CLRF    CONTA3
;borra contador de caracteres.
              CLRF    CONTA4
ACTIV1    MOVF    CONTA3,W
              MOVWF   CONTA4
              BCF     STATUS,C
              RLF     CONTA4,F
;INICIA SELECT CASE
              BTFSS   SELECT,3
              GOTO    ntxND
              CALL    TABLAD
              GOTO    NTX
ntxND     MOVLW   0x01
              XORWF   SELECT,w
              BTFSS   STATUS,Z
              GOTO    NXT2
              CALL    TABLA1
              GOTO    NTX
NXT2     MOVLW   0x02
              XORWF   SELECT,w
              BTFSS   STATUS,Z
              GOTO    NXT3
              CALL    TABLA2
              GOTO    NTX
NXT3     MOVLW   0x03
              XORWF   SELECT,w
              BTFSS   STATUS,Z
              GOTO    NXT4
              CALL    TABLA3
              GOTO    NTX
NXT4     MOVLW   0x04
              XORWF   SELECT,w
              BTFSS   STATUS,Z
              GOTO    NXT5
              CALL    TABLA4
              GOTO    NTX
NXT5     MOVLW   0x05
              XORWF   SELECT,w
              BTFSS   STATUS,Z
              GOTO    NXT6
```

```
CALL    TABLA5
GOTO    NTX
```

```
NXT6     MOVLW   0x06
              XORWF   SELECT,w
              BTFSS   STATUS,Z
              GOTO    NXT7
              CALL    TABLA6
              GOTO    NTX
```

```
NXT7     MOVLW   0x07
              XORWF   SELECT,w
              BTFSS   STATUS,Z
              GOTO    NTX0
              CALL    TABLA7
              GOTO    NTX
```

```
NTX0     CALL    TABLA0
NTX      CALL    DATO
;llama rutina de envío de caracteres al LCD.
              CALL    RETARDOTE
              CALL    RETARDOTE
              CALL    RETARDOTE
```

;leer próximo carácter.

```
              INCF    CONTA3,1
;incrementa el contador de caracteres.
              MOVLW   0X14
;comparar con 15h=20d numero de caracteres de primera.
```

```
              SUBWF   CONTA3,W
              BTFSS   STATUS,C
;terminó primera línea?
              GOTO    ACTIV1
```

si no repita ciclo

```
              RETURN
```

; CONFIGURACION PIC

```
INICIO    SF     STATUS,RP0;
inicio de programa.
              BCF     STATUS,RP1
;ubica en el banco1
              BCF     INTCON,GIE
              MOVLW   0x82
              MOVWF   ADCON1
;A0, A1, A4 entradas analogas ; Vref=GND y
Vref+=Vcc.
              MOVLW   0XFF
              MOVWF   TRISA
;RA0-RA5 entradas, RA0-RA2 parte alta del conversor
A/D.
              MOVLW   0X8F
              MOVWF   TRISC
;RC0-RC7 entradas, configuración externa.
              MOVLW   0x00
              MOVWF   TRISD
;RD0-RD7 salidas, bus de datos LCD.
              MOVLW   0x00
              MOVWF   TRISE
;RE0-RE2 salidas, control LCD.
;configuración transmisión serial.
              MOVLW   0x24
              MOVWF   TXSTA
              MOVLW   0x19
;velocidad de transmisión con el pc 9600 baudios.
              MOVWF   SPBRG
              BCF     STATUS,RP0
;banco 0.
              CLRF   PORTA
;borrado de port A..
```

```

CLRF PORTB
;borrado de port B.
CLRF PORTC
;borrado de port C.
CLRF PORTD
;borrado de port D.
CLRF PORTE
;borrado de portE.
MOVLW 0x90
MOVWF RCSTA

;configuración del display.
MOVLW 0x38
;display a 8 bits y 2 líneas.
CALL CONTROL
MOVLW 0x0C
;activa display.
CALL CONTROL
MOVLW 0x06
;mensaje fijo.
CALL CONTROL

*****

; PROGRAMA PRINCIPAL
CEROS MOV LW 0X30
MOVWF C1FRA1
MOVWF C1FRA2
MOVWF C1FRA3
MOVWF C1FRA4
MOVWF C1FRA5
MOVWF C1FRA6
MOVWF C1FRA7
MOVWF C1FRA8
MOVWF C1FRA05
MOVLW 0X01
MOVWF NUMCICLOS
CLRF TOTALBAJO
CLRF TOTALALTO
CLRF TOTALMAX
CLRF SELECT
RETINI MOV LW 0xFF
MOVWF CONTAL
DECREI call RETARDOTE
DECFSZ CONTAL,1
;decremento CONTA1 salta cuando sea cero.

GOTO DECREI
;100*20 ms
MOVLW 0x80
;dato control para ir a primera fila.
CALL CONTROL
CALL VISUAL
INCF SELECT,F
MOVLW 0xC0
;dato de control para ir a segunda línea.
CALL CONTROL
CALL VISUAL
INCF SELECT,F

RETINI2 MOV LW 0xC8
MOVWF CONTAL
DECREI2CALL RETARDOTE
DECFSZ CONTAL,1
;decremento CONTA1 salta cuando sea cero.

GOTO DECREI2
;100 ms
MOVLW 0x80
;dato de control para ir a primera línea.
CALL CONTROL
CALL LIMPIA
MOVLW 0x80

```

```

;dato control para ir a primera fila.
CALL CONTROL
CALL VISUAL
INCF SELECT,F
MOVLW0xC0
;dato de control para ir a segunda línea.
CALL CONTROL
CALL VISUAL
INCF SELECT,F

RETINI3 MOV LW 0xC8
MOVWF CONTAL
DECREI3CALL RETARDOTE
DECFSZ CONTAL,1
;decremento CONTA1 salta cuando sea cero.

GOTO DECREI3
;100 ms
MOVLW 0x80
;dato de control para ir a primera línea.
CALL CONTROL
CALL LIMPIA
MOVLW 0x80
;dato control para ir a primera fila.
CALL CONTROL
CALL VISUAL
INCF SELECT,F
MOVLW 0xC0
;dato de control para ir a segunda línea.
CALL CONTROL

*****
;RUTINA QUE TOMA EL PESO DE LA
PLATAFORMA (DESTALE), PARA INICIALIZAR EL
LCD EN CERO
DESTAL CLR F BAJO
CALL CONVER1
CALL CONVER2
MOV F CELDA1L,W
MOVWF DESTALE
MOV F CELDA2L,W
ADDWF DESTALE,F
;SE TOMA EL PESO, QUITANDO EL DESTALE

*****
BCF STATUS,RP0
MOVLW 0x07
MOVWF SELECT
CALL VISUAL
MOVLW 0xFF
MOVWF INC
step1 MOVLW0x99 ;153
MOVWF TXREG
BSF STATUS,RP0
wtx BTFSS TXSTA,TRMT
GOTO WTTX
BCF STATUS,RP0
CALL RETARDOTE
CALL RETARDOTE
DECFSZ INC,F
GOTO REVSER
GOTO AQUÍ

;operación manual
REVSER BTFSS PIR1,RCIF
GOTO step1
MOVWF RCREG
MOVWF Entry
MOVLW 0xAC
XORWF Entry,w

```

```

step2          BTFSS STATUS,Z
               GOTO          step1
               MOVLW 0xAC
               MOVWF TXREG
               BSF
               STATUS,RP0
wtx2          BTFSS TXSTA,TRMT
               GOTO wtx2
               NOP
               ;automatic operation
RecCon BCF    STATUS,RP0
               BTFSS PIR1,RCIF
               GOTO          RecCon
               MOVFW RCREG
               MOVWF Entry
               MOVLW 0xF0
               ANDWF Entry,w
               MOVWF INC
               MOVLW 0xC0
               XORWF INC,W
               BTFSS          STATUS,Z
               GOTO          RecCon
PCcon         MOVLW 0x0F
               ANDWF Entry,f
               BTFSS          STATUS,Z
               Goto          RecCon
envPC call    Tmues
               MOVLW 0xDA
               MOVWF TXREG
               BSF STATUS,RP0
wtxA         BTFSS TXSTA,TRMT
               GOTO wtxA
               BCF STATUS,RP0
               MOVFWBPROM
               MOVWFTXREG
               BSF STATUS,RP0
wtxB         BTFSS TXSTA,TRMT
               GOTO wtxB
               BCF STATUS,RP0
               MOVFW APROM
               MOVWF TXREG
               BSF STATUS,RP0
wtxC         BTFSS TXSTA,TRMT
               GOTO wtxC
               BCF STATUS,RP0
               GOTO RecCon

*****
Tmues BCF    STATUS,RP0
               CLRF BAJO
               CLRF ALTO
               CLRF MAX
               MOVLW 0X30
               MOVWF CIFRA05
;se alista el numero de vacas pesadas, para mostrar
en el LCD.
               MOVF NUMCICLOS,W
               MOVWF BAJO
               CALL BINBCD
               CALL BCD2ASCII
               MOVF CIFRA6,W
               MOVWF CIFRA1P
               MOVF CIFRA7,W
               MOVWF CIFRA2P
               MOVF CIFRA8,W
               MOVWF CIFRA3P
;hacemos la medición del peso.
               CLRF NUMPROM
               CLRF APROM
               CLRF BPROM
TOMA CALL    RETARDO
               INCF NUMPROM,f
               CLRF BAJO
               CLRF ALTO
               CLRF MAX
               CALL CONVER1
               CALL CONVER2
;sumamos las celdas de carga.
               MOVF CELDA2L,W
               MOVWF BAJO
               MOVF CELDA2H,W
               MOVWF ALTO
               CALL SUMA
               CALL RESTA
;RESTAMOS EL DESTALE PARA INICIAR
EN CERO LA MEDICION
               MOVF BAJO,W
               ADDWF BPROM,f
               BTFSc STATUS,C
               incF APROM,f
OTRO11 MOVF  ALTO,W
               ADDWF APROM,f
               MOVLW 0x0F
               XORWF NUMPROM,w
               BTFSS STATUS,Z
               GOTO TOMA
               BCF STATUS,C
               ;DIVIDIMOS POR 16 PARA PROMEDIAR
LOS RESULTADOS
               RRF APROM,F
               ;nota rrf16= clr C,rrf alto,rrf bajo
               RRF BPROM,F
               BCF STATUS,C
               RRF APROM,F
               RRF BPROM,F
               BCF STATUS,C
               RRF APROM,F
               RRF BPROM,F
               BCF STATUS,C
               RRF APROM,F
               RRF BPROM,F
               call RESTA
;*****
               MOVF BPROM,W
;GUARDAMOS LA
MOVWF PESOBAJO
;INFORMACION DEL PESO
MOVWF BAJO
MOVF APROM,W
;EN OTROS REGISTROS,
MOVWF PESOALTO
;POR SI HAY CONFIRMACION
MOVWF ALTO
BTFSS BAJO,0
;Revisa Si El Numero Es Entero O Hay Libras
GOTO ENTERO
MOVLW 0X35
MOVWF CIFRA05
ENTERO BCF    STATUS,C
;dividimos por 2 para pasar de libras a kilogramos.
RRF ALTO,F
RRF BAJO,F
CALL BINBCD
CALL BCD2ASCII
MOVLW 0x80
;dato de control para ir a la primera línea.
CALL CONTROL
movlw 0x04

```



```

movwf SELECT
CALL VISUAL
MOVLW 0xC0
;dato de control para ir a la segunda línea.
CALL CONTROL
movlw 0x08
movwf SELECT
CALL VISUAL
return
;visualiza peso actual, info guardada en peso.

;MIRA SI HAY CONFIRMACION DE PESO, PARA
AGREGAR AL ACUMULADO

*****
AQUI call Tmues
      BTFSS PORTC,2
      GOTO SIGUEC
      GOTO CERROSS
SIGUEC BTFSS PORTC,1
      GOTO SIGUE
CEROC BTFSC PORTC,1
;no continua el programa hasta que caiga el pulso.
      GOTO CEROC
      CALL SUMATOTAL
;se testea si hay cambio de modo sigue.
      BTFSS PORTC,0
      GOTO AQUI
CEROM BTFSC PORTC,0
;no continua el programa hasta que caiga el pulso.
      GOTO CEROM
;comienza rutina despliegue total pesado se alista el
numero de vacas pesdas, para mostrar en el LCD.
MOVF NUMCICLOS,W
MOVWFNUMVACAS
DECf NUMVACAS,1
MOVF NUMVACAS,W
MOVWF BAJO
CALL BINBCD
CALL BCD2ASCII
MOVf CIFRA6,W
MOVWF CIFRA1P
MOVf CIFRA7,W
MOVWF CIFRA2P
MOVf CIFRA8,W
MOVWF CIFRA3P
CLRF BAJO
CLRF ALTO
CLRF MAX
CLRF CIFRA1
CLRF CIFRA2
CLRF CIFRA3
CLRF CIFRA4
CLRF CIFRA5
CLRF CIFRA6
CLRF CIFRA7
CLRF CIFRA8
CLRF ATOTALBAJO
CLRF ATOTALALTO
CLRF ATOTALMAX
MOVLW 0X30
MOVWF CIFRA05
BTFSS TOTALBAJO,0
;revisa si el numero es entero o hay libras.
GOTO ENTEROT
MOVLW 0X35
MOVWF CIFRA05
ENTEROT MOVF TOTALBAJO,W
MOVWFATOTALBAJO
MOVf TOTALALTO,W
CLRF CIFRA5
CLRF CIFRA6

```

```

MOVWFATOTALALTO
MOVf TOTALMAX,W
MOVWFATOTALMAX
BCf STATUS,C
;multiplicamos por 2 para pasar de libras a kilogramos.
RRF ATOTALBAJO,F
BCf STATUS,C
RRF ATOTALALTO,F
BTFSC STATUS,C
BSF ATOTALBAJO,7
BCf STATUS,C
RRF ATOTALMAX,F
BTFSC STATUS,C
BSF ATOTALALTO,7
MOVf ATOTALBAJO,W
MOVWFBAJO
MOVf ATOTALALTO,W
MOVWFALTO
MOVf ATOTALMAX,W
MOVWFMAX
CALL BINBCD
;convertimos el dato a BCD
CALL BCD2ASCII
;luego a ASCII, para visualizar en el LCD
MOVLW 0x80
;dato de control para ir a primera línea
CALL CONTROL
MOVLW 0x06
MOVWF SELECT
CALL VISUAL
MOVLW 0xC0
;dato de control para ir a segunda línea
CALL CONTROL
MOVLW 0x08
MOVWF SELECT
CALL VISUAL
MOVLW 0xC0
;dato de control para ir a primera línea
CALL CONTROL
MOVLW 0x08
MOVWF SELECT
CALL VISUAL
MOVLW 0xC0
;dato de control para ir a segunda línea
CALL CONTROL
MOVLW 0x08
MOVWF SELECT
CALL VISUAL
UNOT BTFSS PORTC,0
      GOTO UNOT
CEROT BTFSC PORTC,0
      GOTO CEROT
MOVLW 0x80
;dato de control para ir a primera línea
CALL CONTROL
CALL LIMPIA
;comienza rutina despliegue promedio pesado.
MOVFTOTALMAX,W
;movemos los registros.
MOVWFTMAX
;de almacenamiento
MOVf TOTALALTO,W
;a los de trabajo.
MOVWF TALTO

*****
;RUTINA DE DIVISION
MOVf TOTALBAJO,W
;para el promedio.
MOVWF TBAJO
MOVf NUMVACAS,W
MOVWF DIVI
;rutina para obtener el promedio pesado.
CLRF BAJO
CLRF ALTO
CLRF MAX
CLRF CIFRA1
CLRF CIFRA2
CLRF CIFRA3
CLRF CIFRA4
CLRF CIFRA7
CLRF CIFRA8

```

```

        CLRF  RESU1
        CLRF  RESU2
        BCF   STATUS,C
;dividimos por 2 para pasar de libras a kilogramos.
        RRF   TBAJO,F
        BCF   STATUS,C
        RRF   TALTO,F
        BTFSC STATUS,C
        BSF   TBAJO,7
        BCF   STATUS,C
        RRF   TMAX,F
        BTFSC STATUS,C
        BSF   TALTO,7

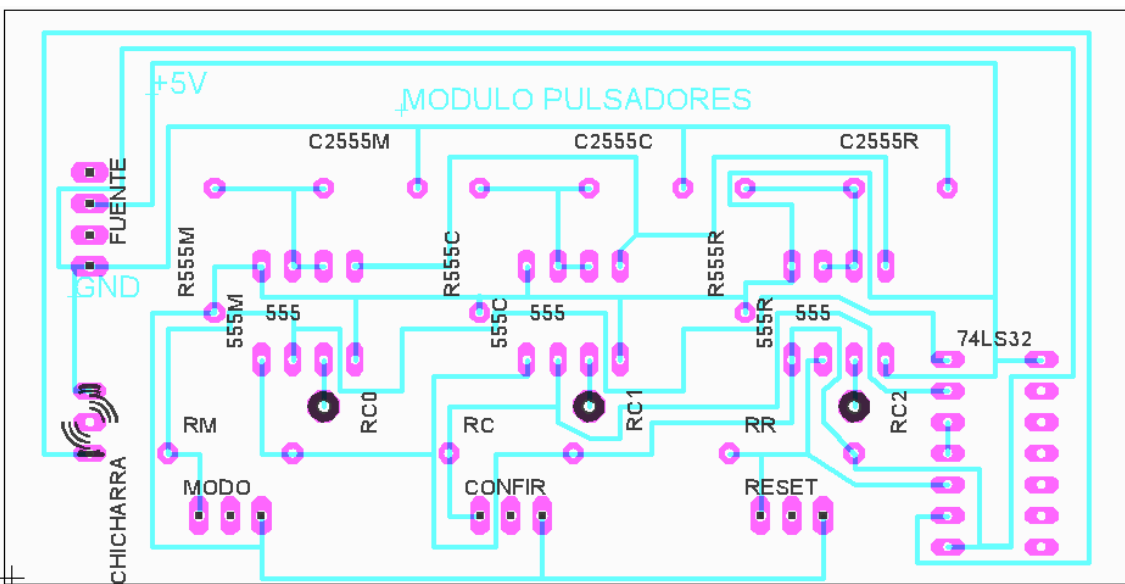
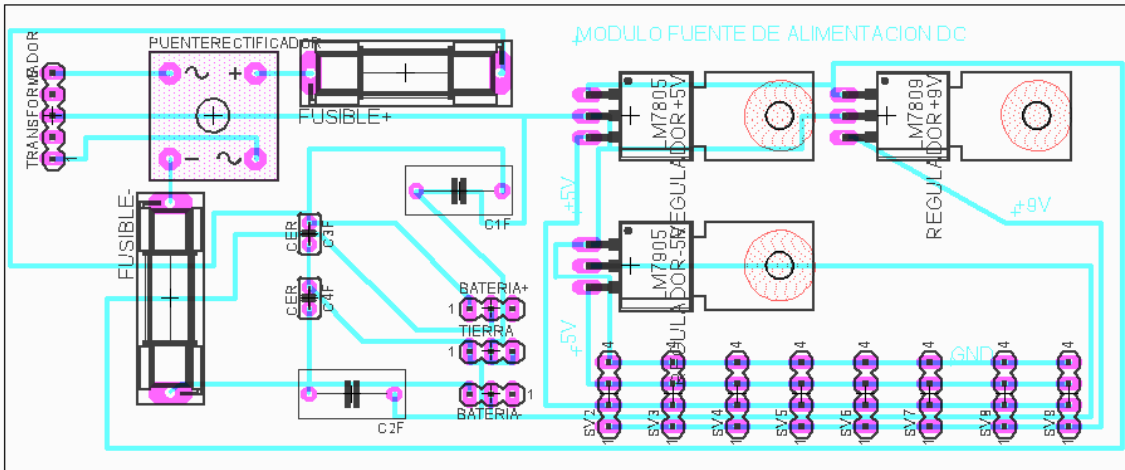
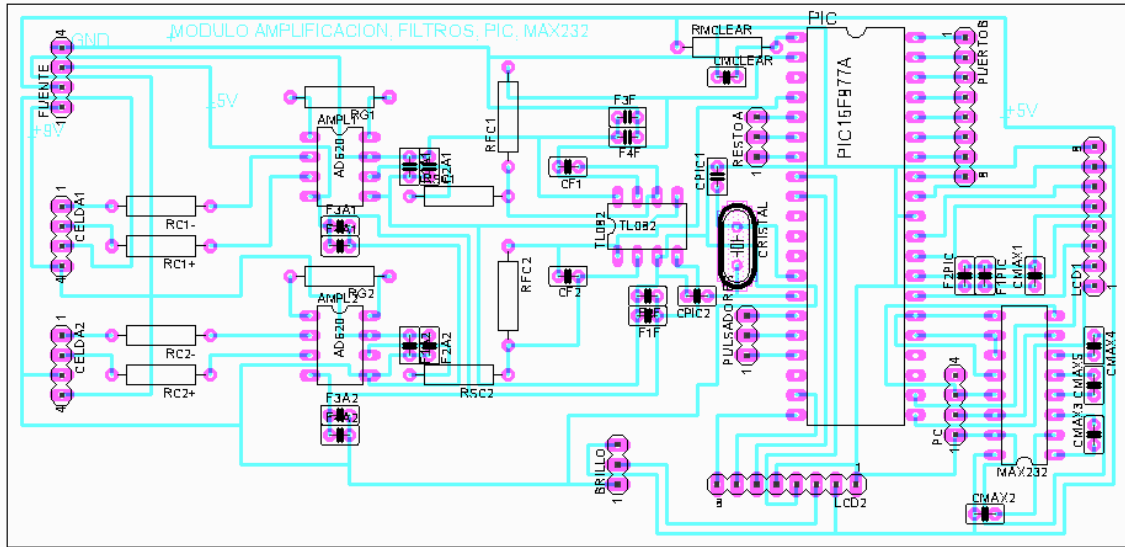
CICLO1 MOVFW TBAJO
        MOVWF COCI
        MOVFW DIVI
        SUBWF TBAJO,F
        BTFSC STATUS,C
        GOTO  SIG1
        MOVLW 0xFF
        ADDWF TALTO,F
        BTFSC STATUS,C
        GOTO  SIG1
        MOVLW 0xFF
        ADDWF TMAX,F
        BTFSS STATUS,C
        GOTO  DECIMAL
SIG1    OVLW  0x01
        ADDWF RESU2,F
        BTFSS STATUS,C
        GOTO  CICLO1
        INCF  RESU1,F
        GOTO  CICLO1
DECIMAL CLRF  TALTO
        MOVFW COCI
        MOVWF TBAJO
        BCF   STATUS,C
        MOVWF BAJO
        CALL  BINBCD
;convertimos el dato a BCD.
        CALL  BCD2ASCII
;luego a ASCII, para visualizar en el LCD
        MOVF  CIFRA8,W
;la cifra decimal.
        MOVWF CIFRA05
        MOVF  RESU1,W
;movemos los registros .
        MOVWF ALTO
;de trabajo de la rutina.
        MOVF  RESU2,W
;de promedio a las de.
        MOVWF BAJO
;conversión y visualización
        CALL  BINBCD
;convertimos el dato a BCD.
        CALL  BCD2ASCII
;luego a ASCII, para visualizar en el LCD

        MOVLW 0x80
;dato de control para ir a primera línea.
        CALL  CONTROL
        MOVLW 0x05
        MOVWF SELECT
        CALL  VISUAL
        MOVLW 0xC0
;dato de control para ir a segunda línea.
        CALL  CONTROL
        MOVLW 0x08
        MOVWF SELECT
        CALL  VISUAL
UNOP   BTFSS PORTC,0

        RLF   TBAJO,F
        RLF   TALTO,F
        RLF   TBAJO,F
        RLF   TALTO,F
        RLF   TBAJO,F
        RLF   TALTO,F
        RLF   TBAJO,F
        RLF   TALTO,F
        CLRF  RDEC
CICLO2 MOVFW IVI
        SUBWF BAJO,F
        BTFSC ATUS,C
        GOTO  G2
        MOVLW 0xFF
        ADDWF TALTO,F
        BTFSS TATUS,C
        GOTO  ONV
SIG2   INCF  RDEC,F
        GOTO  CICLO2
CONV   CLRF  DECIMA
        BTFSS RDEC,3
        GOTO  SIGUE1
        MOVLW 0x32
        ADDWF DECIMA,F
SIGUE1 BTFSS RDEC,2
        GOTO  SIGUE2
        MOVLW 0x19
        ADDWF DECIMA,F
SIGUE2 BTFSS RDEC,1
        GOTO  SIGUE3
        MOVLW 0x0C
        ADDWF DECIMA,F
SIGUE3 BTFSS RDEC,0
        GOTO  FINAL
        MOVLW 0x06
        ADDWF DECIMA,F
FINAL   OVF   DECIMA,W
        GOTO  UNOP
CEROP  BTFSC PORTC,0
        GOTO  CEROP
        MOVLW 0x80
;dato de control para ir a primera línea.
        CALL  CONTROL
        CALL  LIMPIA
        GOTO  AQUÍ
        END

```

ANEXO B. TARJETAS DE CIRCUITO IMPRESO



ANEXO C. PLANO GENERAL SISTEMA DE REGISTRO Y PESAJE DEGANADO

