



**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**

**FACULTAD DE INGENIERÍA**

---

---

**DESARROLLO DE FIRMWARE E INTERFAZ DE  
USUARIO PARA UN DEFLECTÓMETRO LIGERO**

**TESIS**

**QUE PARA OBTENER EL TÍTULO DE INGENIERO  
EN COMPUTACIÓN**

**PRESENTA:  
CÉSAR OMAR HERNÁNDEZ CRUZ**

**ASESOR: ING. ENRIQUE RAMÓN GOMEZ ROSAS**

**CIUDAD UNIVERSITARIA NOVIEMBRE/2014.**



# Desarrollo de firmware e interfaz de usuario para el manejo de un deflectómetro ligero

## Contenido

Objetivo .....	5
Hipótesis .....	5
1 INTRODUCCION .....	6
1.1 Pruebas no destructivas .....	6
1.2 Fundamentos del deflectómetro ligero.....	6
1.4 Características del modelo mecánico .....	7
1.2 Fenómeno físico e interpretación.....	8
2 Desarrollo del sistema electrónico.....	10
2.1 Modelo general.....	10
2.2 La fuente de poder .....	12
2.3 Transductores .....	12
2.3.1 Celda de carga .....	13
2.3.2 Geófono.....	13
2.3.3 Calibración .....	14
2.4 Acondicionamiento de las señales: El amplificador operacional .....	14
2.5 Muestreo analógico y conversión digital: Convertidores ADS y DAC .....	15
2.6 El módulo de posicionamiento global: Modulo GPS.....	17
2.7 Hardware de control de muestras: El PIC y la memoria RAM .....	18
2.8 Sistema de comunicación inalámbrica: El modulo Bluetooth.....	20
2.9 Análisis final del modelo electrónico .....	21
3 El sistema de control de muestras (Firmware).....	22
3.1 Diagrama de procesos.....	23
3.2 Herramientas de desarrollo .....	24
3.3 Protocolos de comunicación .....	24
3.3.1 El bus SPI.....	24
3.3.2 Comunicación serial EUSART .....	25
3.3 Entradas y salidas esperadas.....	26

3.4	Algoritmos.....	26
3.5	Detalles de implementación y programación.....	38
4	La interfaz de usuario.....	39
4.1	Diagrama de procesos.....	39
4.2	Herramientas de desarrollo.....	40
4.3	Características generales del dispositivo inalámbrico.....	40
4.4	Entradas y salidas esperadas.....	41
4.5	Algoritmos.....	43
4.6	Detalles de implementación y programación.....	50
4.7	Análisis del flujo de procesos.....	51
4.8	Análisis del acoplamiento del firmware e interfaz de usuario.....	52
5	Guía de usuario.....	53
5.1	Manual del firmware.....	53
5.2	Manual de la interfaz de usuario.....	61
5.2.1	Controles del PIC.....	62
5.2.2	Controles de Datos.....	63
5.2.3	Controles del GPS.....	64
5.2.4	Controles de los cálculos.....	65
5.2.5	Muestreo de forma manual.....	66
5.2.6	Muestreo de forma automática.....	73
5.2.7	Funciones adicionales.....	79
	Aplicación.....	83
	Primera prueba.....	85
	Segunda prueba.....	86
	Tercera prueba.....	87
	Cuarta prueba.....	88
	Quinta prueba.....	89
	Sexta prueba.....	90
	Conclusiones.....	91
	Bibliografía/ Referencias.....	92
	Apéndice.....	94
I	Código del Firmware.....	94

Main.c .....	94
1_parametros.h.....	102
2_dac.h.....	103
3_ads.h.....	103
4_memoria.h.....	105
6_GPS.h .....	111
7_bluetooth.h .....	116
II Diagramas de la Interfaz de usuario (LabView 2012) .....	118



## Objetivo

El presente trabajo tiene por objetivo presentar de forma clara y concisa la metodología específica que se siguió para el desarrollo del firmware y software, así como dar detalles sobre la fabricación e implementación de un defleómetro ligero. Se dará una introducción sobre la teoría de funcionamiento y fabricación, y se abordará de manera formal la parte lógica del sistema.

## Hipótesis

El presente trabajo básicamente tendrá la finalidad de responder la siguiente pregunta: ¿Se puede crear un sistema de adquisición de datos para implementarse en un defleómetro ligero? Aunado a esto se debe de comprobar que el diseño electrónico provisto por el Instituto de Ingeniería () puede ser utilizado con un firmware especializado para generar las actividades requeridas para operar un defleómetro ligero.

# 1 INTRODUCCION

El uso del parámetro de deflexión es universalmente usado para conocer de forma certera la capacidad estructural de varios tipos de suelos. La deflectometría de impacto (FWD) es una excelente forma de obtener el Número Estructural asociado a los tipos de suelos sin dañarlo, ya que está catalogada como una prueba no destructiva. Las pruebas a realizar consisten en una serie de ensayos con los cuales mediante la aplicación de una carga se puede medir la deformación en cierto tipo de transductores producidas por el efecto del impacto. De esta manera, el ingeniero será capaz de realizar las respectivas correcciones del terreno para poder hacer uso de él a conveniencia.

## 1.1 Pruebas no destructivas

Se denomina ensayo no destructivo a cualquier tipo de prueba que sea practicada sobre algún tipo de material, hecha de tal forma que no altere las propiedades físicas, químicas, mecánicas o dimensionales de la misma. Esto implica un daño imperceptible o nulo.

En general, las pruebas no destructivas ofrecen información menos precisa que las pruebas destructivas, por lo que para hacer uso de este tipo de metodología, es necesario tener un estudio base, o realizarlas de forma complementaria con pruebas destructivas.

Este tipo de ensayos se dividen fundamentalmente en tres tipos: Defectología, caracterización y metrología. Para el caso del uso del deflectómetro ligero, nos centraremos en el grupo de caracterización, que es el estudio de las características estructurales y mecánicas de los materiales.

## 1.2 Fundamentos del deflectómetro ligero

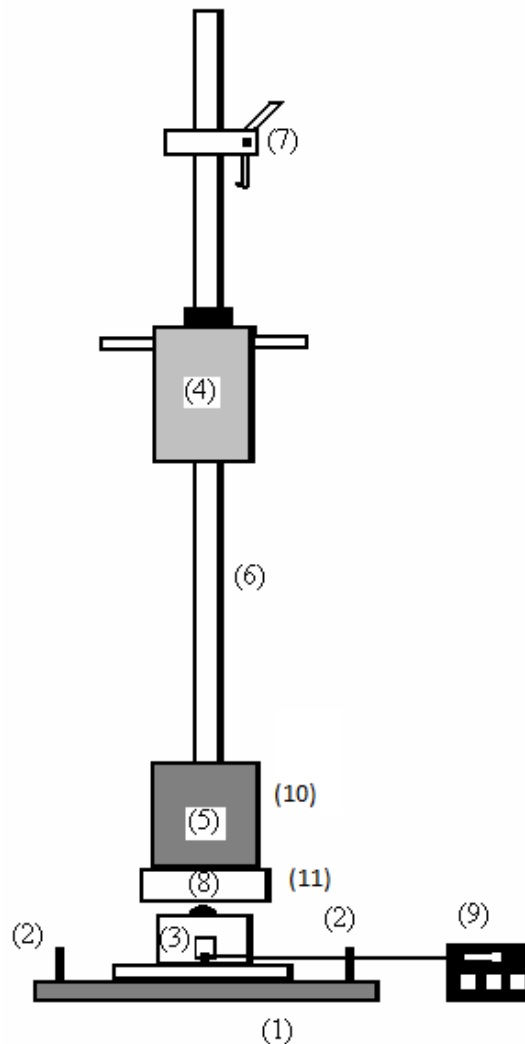
El uso de deflectómetros para conocer las características de los suelos data de los años sesenta. Estos dispositivos en un inicio fueron diseñados en respuesta a la demanda de realizar estudios de diferentes tipos de terreno de forma no destructiva y accesible para los ingenieros encargados de los mismos.

Si bien estos instrumentos podían realizar los ensayos solicitados, eran dispositivos de grandes dimensiones y peso, por lo que resultaban muy difíciles de maniobrar e inclusive necesitaban de vehículos tractores especiales para poder ser transportados y operados. Ante esta problemática se diseñaron los llamado deflectómetros ligeros, los cuales implementando la tecnología electrónica moderna pudieron disminuir notablemente los costos y el tamaño necesario de operación. De este modo, ya no era necesario tener un

operador especial para el manejo del dispositivo, ni tampoco un vehículo remolcador para su transporte.

En el Instituto de Ingeniería de la UNAM, la coordinación de electrónica se dió a la tarea de proponer la manufactura de un deflectómetro ligero diseñado y ensamblado totalmente en México, con el objetivo de desarrollar tecnología que estuviese a la vanguardia de los adelantos mundiales actuales. Concebido como un proyecto a etapas, se requirió de la participación de múltiples profesionistas para el desarrollo del mismo. Para el caso de este trabajo escrito, nos centraremos fundamentalmente en el desarrollo del software y firmware del sistema, no sin antes dar una breve explicación del dispositivo en general.

#### 1.4 Características del modelo mecánico



El modelo mecánico comprende una serie de componentes listados a continuación:

- Placa de carga (1)
- Asas (2)
- Medidor de asiento (3)
- Masa suspendida o maza (4)
- Conjunto de muelles (5)
- Barra guía (6)
- Mecanismo de liberación de la masa (7)
- Rótula (8)
- Equipo electrónico de control (9)
- Transductores (10)
- Tarjeta de adquisición (11)

Si bien el modelo físico diseñado por el IINGEN tiene una serie de medidas específicas, el desarrollo del software ha sido diseñado para poder funcionar con diferentes combinaciones de tamaños. De esta forma, valores de magnitudes físicas del deflectómetro como el diámetro de la placa, o peso de la maza no están condicionados por software, por lo que pueden ser variables. De cualquier forma, para las pruebas y ensayos aquí estudiados se utilizó una maza de 10kg, una barra guía de 5kg (incluidos los componentes ligados a la misma) y un diámetro de placa de 300mm, los cuales representan los valores fundamentales para el estudio.

## 1.2 Fenómeno físico e interpretación

El funcionamiento del deflectómetro ligero se basa principalmente en el módulo dinámico de deformación del terreno a estudiar con el objetivo de conocer el comportamiento de las capas terraplenes. Se basa en la medición de la energía producida por el efecto de una deformación una vez que cesa el esfuerzo aplicado sobre esta. Esta energía puede ser calculada mediante una serie de fórmulas que ayudan a comprender el estado y fiabilidad del terreno para posibles usos.

Para poder obtener los valores totales de dicho estudio, se utilizan los 2 transductores implementados en el modelo electrónico: la celda de carga y el geófono. Con estos dispositivos se pueden conocer los valores necesarios para realizar el ensayo.

La celda de carga podrá capturar la fuerza del impacto de la masa al momento de realizar el ensayo. El geófono funcionará a manera de velocímetro, el cual nos ayudará a calcular los valores de medición de compactación asociados al ensayo.

El equipo electrónico utiliza la señal de los sensores para determinar el asiento  $S$  de la placa al realizar el ensayo. Una vez realizado el ensayo, es labor del equipo electrónico almacenar el asiento máximo de cada impacto y el módulo de deformación dinámico  $E_{vd}$  calculado.

El equipo electrónico de control se divide en dos módulos principales: El sistema de adquisición de datos y la interfaz de control. Debido a las altas velocidades que maneja el sistema de adquisición de datos, se optó por realizar el muestreo de los voltajes producidos por los transductores en este módulo, para después obtener el módulo de deformación dinámico y el asiento máximo en la interfaz de control. El sistema de adquisición almacena los datos obtenidos del ensayo para después enviarlos a petición a la interfaz de control.

Para realizar los cálculos necesarios en cada ensayo, se necesitan los siguientes parámetros:

- Fuerza  $F$  (Obtenida de la celda de carga)
- Diámetro del plato  $d$
- Valor máximo medio del ensayo (velocidad obtenida del geófono)  $S$
- radio del plato  $r$
- Área del plato  $A$

El módulo dinámico de deformación es calculado mediante la ecuación (I) en función de la máxima tensión normal (calculada mediante  $\text{Sigma} = F/A$ ) bajo la placa de carga, el diámetro de la placa y el valor medio de los tres asientos medidos.

$$(I) \quad E_{vd} = (1.5)(r)(F/A) / S$$

Las medidas asociadas al modelo físico son:

Diámetro de la placa = 300mm

Masa total de la maza = 10 kg

Máxima tensión normal teórica =  $0.1 \text{ MN/m}^2$

Cabe destacar que el ensayo será válido únicamente si no existe un movimiento lateral de la placa al momento del golpe o si el ángulo de inclinación no tiene un error mayor del 6%. De cualquier manera, el sistema está diseñado para poder almacenar los datos de tres ensayos consecutivos, y así sacar la media de los mismos, con el objetivo de brindar mayor fiabilidad a los resultados.

## 2 Desarrollo del sistema electrónico

A lo largo del desarrollo del firmware del PIC, se utilizó un componente electrónico desarrollado en el IINGEN, el cual tiene la función de recibir una serie de muestras de voltaje obtenidas a través de transductores, para luego codificarlas, almacenarlas en una memoria y acceder a ellas cuando sea necesario. Dicha placa electrónica tiene la capacidad de poder manejar varios componentes adicionales, como es el caso de un módulo GPS, y en caso de ser necesario, tiene la capacidad de manejar hasta 4 transductores diferentes.

Dicho modelo físico fue ideado en primera instancia para resolver la necesidad de desarrollar un sistema de adquisición de datos para el deflectómetro ligero de impacto, que hace uso de 2 tipos diferentes de transductores. Estos son el geófono y la celda de carga, de los cuales ya hablamos con anterioridad.

### 2.1 Modelo general

El diagrama electrónico del sistema de adquisición de 4 canales se puede apreciar en la **imagen 1**. Como se puede observar, varios de los componentes de dicho modelo no están relacionados de forma directa con el PIC que contendrá el firmware del control del sistema, por lo que para motivos de comprensión, se verán como bloques de procesos y/o entradas al sistema, según sea el caso. En cuanto a los dispositivos que tendrán una comunicación constante con el PIC, y por extensión con el firmware del sistema, serán analizados de forma más exhaustiva en el siguiente tópico.



## 2.2 La fuente de poder

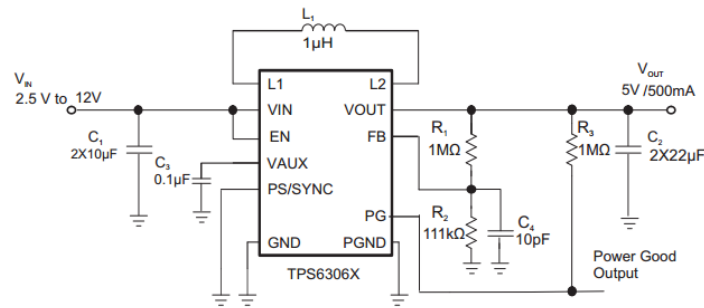


Figure 21. 5V and 500mA from 1 or 2 cell Li-Ion

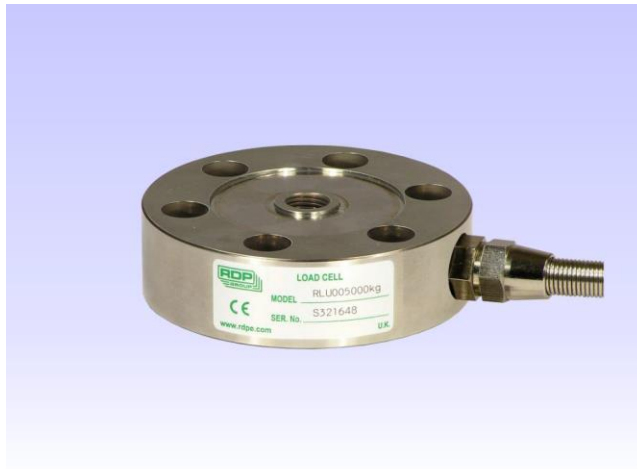
La fuente de poder ha sido diseñada para poder ser alimentada directamente con rangos de voltaje entre 2.5 y 12 volts. Esto permite poder alimentar al dispositivo con múltiples tipos baterías sin que se afecte el funcionamiento del dispositivo en general. Utiliza un circuito integrado TPS63060, el cual es un convertidor buck-boost, el cual a la salida puede mantener un voltaje estable sin importar que a la entrada exista un potencial mayor o menor a este. El voltaje de salida del convertidor buck-boost es de 5 volts ideales, y después de este se tiene un circuito LF33CV el cual a su salida regula el voltaje a 3.3 V. Es necesario tener dos salidas de voltaje dedicadas debido a que si bien prácticamente todo el dispositivo trabaja a 5 volts, el modulo bluetooth añadido funciona únicamente a 3.3 V.

## 2.3 Transductores

Los transductores asociados al sistema de adquisición se encargan de procesar la energía liberada por los ensayos al momento de realizar el impacto con la masa, para después convertirla en señales de voltaje que serán de utilidad para conocer los valores del módulo de deformación dinámico. La interpretación del voltaje obtenido de los transductores depende de las características generales de los mismos, por lo que a continuación se dará una breve explicación de los transductores utilizados.



### 2.3.1 Celda de carga



El modelo de celda de carga utilizado es el RLU02500kg, el cual puede llegar a soportar rangos de  $\pm 2500$  kg. Este rango de valores es suficiente para realizar los ensayos de impacto con la masa, e incluso puede tolerar una mayor fuerza a razón de poder contar con un mayor espectro de tolerancia dentro del uso del deflectómetro. El rango de voltajes de operación puede variar entre 0 y 15V, operando en la tarjeta de adquisición a 5V.

La salida de la celda está en un rango de  $2\text{mV/V} \pm 10\%$ . Tomando en cuenta que el máximo voltaje se obtiene cuando la celda es estresada a su máxima tolerancia, se puede notar que los voltajes entregados por los ensayos serán relativamente bajos, por lo que es necesaria la implementación de los amplificadores operacionales señalados en el diagrama electrónico.

### 2.3.2 Geófono



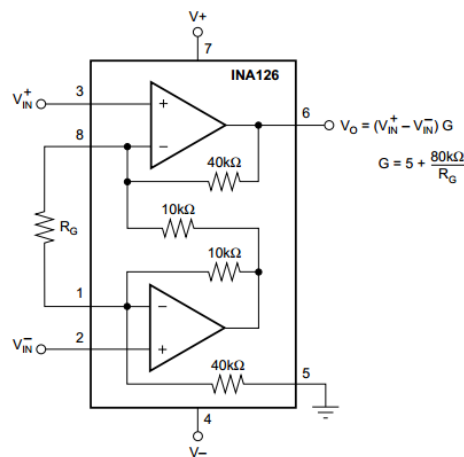
Con el objetivo de poder censar la velocidad característica a la cual se realiza el ensayo del deflectómetro se utiliza un geófono electromagnético (SENSOR Nederlandb.v. serie SM) el cual al momento de realizar el ensayo genera una curva de voltaje característica. Debido a los niveles de voltaje de salida ( $28.8 \text{ V/m/s}$ ) que maneja este transductor, es necesario crear un divisor de voltaje para que pueda ser sensado de forma apropiada por el amplificador operacional. Una vez que ha sido capturada la señal representativa de la velocidad, esta

deberá ser integrada para obtener la curva característica de la aceleración. En capítulos posteriores se ahondará sobre este tema.

### 2.3.3 Calibración

En la práctica el geófono, debido a sus características, no necesita de alguna calibración inicial específica. En contraste, la celda de carga necesita una rutina inicial de nivelación de voltaje para asegurarse de que las mediciones realizadas son correctas. Esto debido a que pueden existir deformaciones en la celda a causa de posibles cambios físicos causadas por múltiples ensayos, o por factores tan simples como el cambio de temperatura. Ante esto, el puente de wheatstone interior de la celda debe ser ajustado para lograr mantener la estabilidad en las condiciones naturales del ensayo a realizar. Para poder realizar la nivelación se utiliza el convertidor digital-analógico integrado en la tarjeta de adquisición de datos. El método implementado, así como más detalles del modelo se darán más adelante del capítulo.

## 2.4 Acondicionamiento de las señales: El amplificador operacional

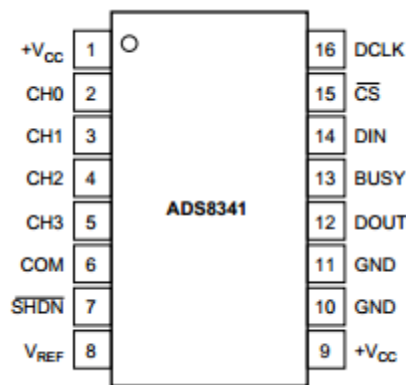


Como puede observarse en la imagen del sistema de adquisición, el modelo electrónico cuenta con una serie de amplificadores operacionales. Dichos dispositivos tienen la importante labor de generar un aumento de los niveles de voltaje que se perciben directamente de los transductores a utilizar sin tener una interferencia (ruido) considerable. Para este propósito se utiliza el circuito integrado INA126, cuya lógica interna se puede apreciar en la imagen anterior. La ganancia que puede tener este amplificador viene dada directamente por la resistencia  $R_G$  acoplada en la parte externa, pudiendo tener un umbral de 5V/V hasta 10000 V/V. La resistencia  $R_G$  utilizada viene definida directamente por los tipos de transductores que se estén empleando. Para el caso de la celda de carga, el

amplificador operacional asociado tiene una resistencia de 980 Ohms, mientras que para el amplificador asociado al geófono, la resistencia es de 1kohm.

La ganancia generada por estas resistencias permite que los voltajes a la salida del amplificador operacional estén en un umbral de entre 0 y 5 volts. El voltaje de referencia usado para los amplificadores es de 2.048 V; esto ha sido pensado para evitar el uso de una fuente de poder con voltajes negativos a su salida, lo que simplifica el diseño y disminuye los costos de fabricación. Finalmente las salidas de estos amplificadores son conducidas al convertidor analógico digital para su procesamiento.

## 2.5 Muestreo analógico y conversión digital: Convertidores ADS y DAC



El convertidor analógico/digital tiene la función de hacer la conversión de las muestras de voltaje de los canales de adquisición, para luego enviar estos valores al PIC a petición. Los voltajes de entrada de cada uno de los canales no provienen directamente de los transductores, sino que son tratados por una serie de amplificadores operacionales, con el fin de que los valores leídos sean lo suficientemente altos como para que el convertidor pueda operar.

El proceso de conversión analógica a digital se realiza mediante aproximaciones sucesivas, haciendo uso de un reloj interno (aunque puede usarse una señal externa), y este proceso puede alcanzar velocidades de hasta 2.4Mhz. El canal que será muestreado depende de la instrucción que se le mande al convertidor, pudiendo incluso realizar muestras diferenciales entre los canales. Cada canal es de 16 bits.

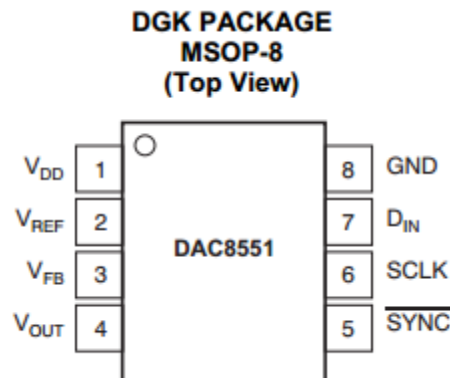
La comunicación entre el convertidor y el micro controlador se realiza mediante el protocolo SPI, siguiendo el siguiente modelo de funcionamiento y tomando en cuenta que el PIC funciona como dispositivo maestro:

- Se envía una instrucción de 8 bits al convertidor.

-El convertidor toma un periodo máximo de 8 $\mu$ s para realizar las aproximaciones sucesivas de los niveles de voltaje del canal escogido.

-El convertidor envía al micro controlador los 16bits de información concernientes a los niveles de voltaje del canal escogido.

Finalmente, el ADS8341 tiene un habilitador de comunicación que se denota como CS el cual nos servirá para elegir si el dispositivo acepta o no señales de control. Esto debido a que 4 componentes del sistema se comunican vía SPI, y solo hay una canal de comunicación.



El DAC tiene una función crucial al momento de realizar la calibración del dispositivo. Es bien sabido que los transductores mecánicos no conservan sus propiedades a lo largo del tiempo, por lo que pueden presentar valores diferentes a lo esperado en los ensayos realizados. Para evitar que el sistema presente deformaciones en las lecturas, así como para eludir cambiar los transductores de forma constante, se ha implementado un mecanismo de realimentación el cual medirá de forma constante los valores del ADS para saber si son adecuados. En caso que no sea así, el micro controlador enviará un comando al DAC para intentar nivelar los voltajes obtenidos del canal de captura. Debido a las necesidades del dispositivo, el único canal que está habilitado para la corrección del voltaje es el canal 0, correspondiente a la celda de carga del sistema del deflectómetro.

El método de comunicación entre el micro controlador y el DAC es vía SPI, y de la misma forma que con el convertidor ADS, el PIC actúa como maestro en la comunicación, mandando una instrucción de 8 bits.

La velocidad máxima que soporta el componente es de 30Mhz y se necesitan 24 ciclos de reloj para poder establecer la señal correcta. Como nota final, este componente también cuenta con un habilitador denotado como SYNC.

## 2.6 El módulo de posicionamiento global: Módulo GPS



El módulo GPS LR9552 es un componente exterior al sistema principal de adquisición, el cual puede ser accedido mediante comunicación serial. Dicha comunicación se realiza siguiendo los protocolos de comunicación NMEA.

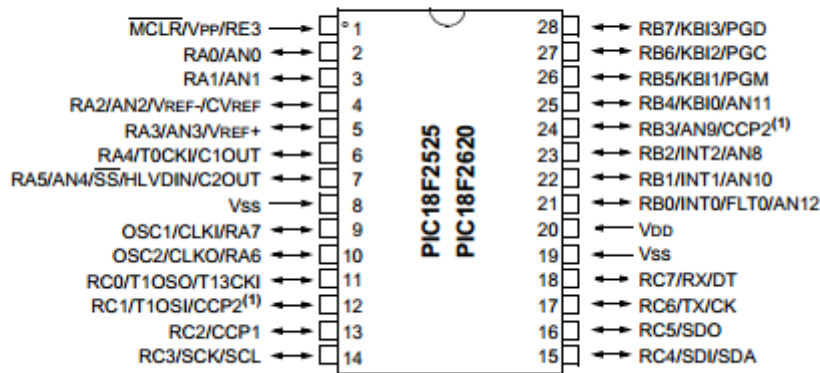
Las principales ventajas de este componente son que puede funcionar con voltajes TTL, y que la interpretación de la información satelital se puede realizar de forma transparente. La información satelital que arroja a través de los canales de comunicación es la siguiente:

- GGA(1sec)
- GSA(5sec)
- GSV(5sec)
- RMC(1sec)
- VTG(1sec)

El baudrate que puede manejar el módulo GPS es de 4800 hasta 57600, siendo 9600 el valor por default. Además, se pueden realizar diferentes configuraciones al dispositivo haciendo uso del protocolo de comandos SiRF-proprietarybinaryprotocols.

## 2.7 Hardware de control de muestras: El PIC y la memoria RAM

### 28-Pin SPDIP, SOIC

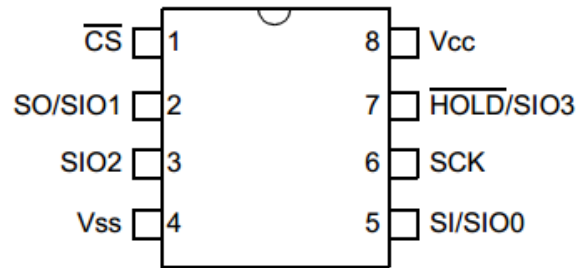


Este componente es la parte fundamental del sistema, ya que representa el cerebro del mismo. Es aquí donde residirá el firmware necesario para el control absoluto del dispositivo y será el controlador de la comunicación con la interfaz de usuario.

Funciona con un cristal de cuarzo de 40Mhz, lo cual permite obtener tasas de muestreo del firmware de hasta 8000 #m/sec. Consta de 3 puertos de comunicación paralela, de los cuales solo se utilizarán 2 para el control de los dispositivos con los cuales mantiene comunicación. Además de esto, tiene dos mecanismos fundamentales de comunicación serial que se usarán en el sistema: la comunicación serial, que se realizará con los pines 17 y 18, y la comunicación SPI, que corresponde a los pines 14,15 y 16.

Si bien este micro controlador tiene muchas características, nos enfocaremos principalmente a las ya citadas con el objetivo de no derivar en conocimiento no relacionado al sistema.

### SOIC/TSSOP/PDIP



La función de la memoria RAM es la de guardar todas las muestras que sean capturadas del convertidor ADS. La implementación de esta memoria es necesaria debido a que el proceso de captura de los valores de voltaje de los canales se realiza a altas velocidades, por lo que si la transmisión de la información se hiciera de forma directa, se perderían valores de los ensayos.

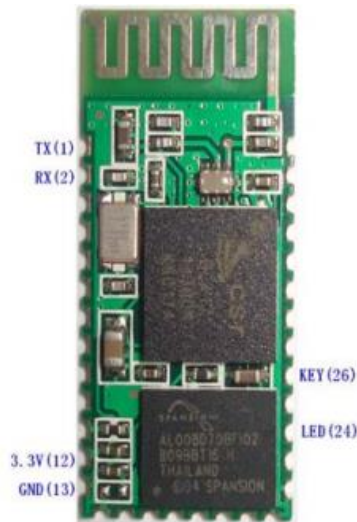
Esta memoria consta de 1Mbit de localidades, de 8 bits cada una. Puede trabajar a frecuencias de hasta 20Mhz y presenta un recorrido de localidades en forma de anillo, por lo que si se alcanza el límite de direcciones en el apuntador, el apuntador se resetea y vuelve al principio.

La memoria tiene 3 modos fundamentales de operación: por Byte, por página y acceso secuencial. Para el desarrollo del firmware, se optó por escoger el modo de acceso por paginación; esto debido a que presenta tiempos de respuesta más óptimos respecto a los otros 2 métodos de acceso.

Debido al tamaño de cada localidad de memoria y tomando en cuenta que los valores de voltaje enviados por el ADS son de 16 bits, se necesitan al menos 2 localidades de memoria para almacenar los valores de cada canal de adquisición.

El método de comunicación con el micro controlador es vía SPI, y se tiene un habilitador del componente, denotado por CS.

## 2.8 Sistema de comunicación inalámbrica: El módulo Bluetooth



El micro controlador realiza la comunicación con la interfaz de usuario haciendo uso de este módulo bluetooth. Gracias a sus características, la información que recibe de sus líneas de comunicación puede ser enviada de forma directa, sin necesidad de alguna configuración especial.

Tiene la capacidad de establecer una contraseña para reforzar la seguridad en el envío de datos, así como se le puede asignar un nombre y el rol de maestro o esclavo. Puede establecer comunicación con otro dispositivo y guardar su dirección para evitar que otro dispositivo pueda acceder al canal de comunicación (aunque también puede trabajar en modo multiusuario).

En cuanto a las características internas, estas pueden ser configuradas mediante el uso de comandos AT. Se tiene un led indicador del estatus entre el circuito y el dispositivo y tiene una línea dedicada al reset del dispositivo. Las velocidades de baudrate que puede manejar son de 1200 a 115200, siendo 9600 el default.



## 2.9 Análisis final del modelo electrónico

La teoría de funcionamiento del dispositivo se puede resumir de la manera siguiente: Al momento de realizar un ensayo en la placa de presión del deflectómetro, se genera un cambio de voltaje en los transductores. La celda de carga y el geófono reaccionan enviando una señal de voltaje a los amplificadores operacionales, los cuales le aplican una ganancia basada en  $R_g$ . El nuevo voltaje amplificado pasa al convertidor analógico-digital, el cual envía la señal digital con el valor del canal seleccionado directamente al PIC para su procesamiento. Una vez la señal ha sido procesada, esta se almacena en la memoria RAM, y se mantiene a la espera de que el usuario final realice la consulta de dicha información.

El proceso de selección de canales viene dado directamente por el firmware del PIC, el cual se verá en capítulos posteriores. El sistema ha sido diseñado para poder almacenar las muestras adquiridas en la memoria RAM incorporada, con el objetivo de aumentar la tasa de muestreo de los canales de adquisición.

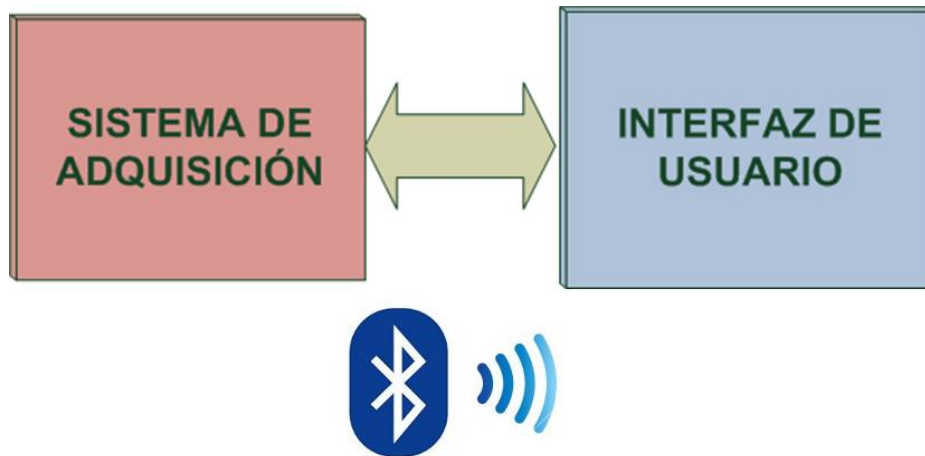
El PIC puede realizar peticiones a voluntad de los datos entregados por el módulo GPS y puede entablar conexión con el módulo bluetooth para realizar la comunicación con el dispositivo inalámbrico a utilizar.

Finalmente, el convertidor digital-analógico tiene la importante labor de nivelar el voltaje obtenido de la celda de carga. Esto es necesario debido a que las condiciones de humedad y temperatura, así como posibles deformaciones físicas del transductor pueden ocasionar medidas erróneas dentro del sistema.

En el siguiente capítulo se abordará la manera en la que el firmware realiza la interacción con los componentes electrónicos para generar el sistema de control de muestras.

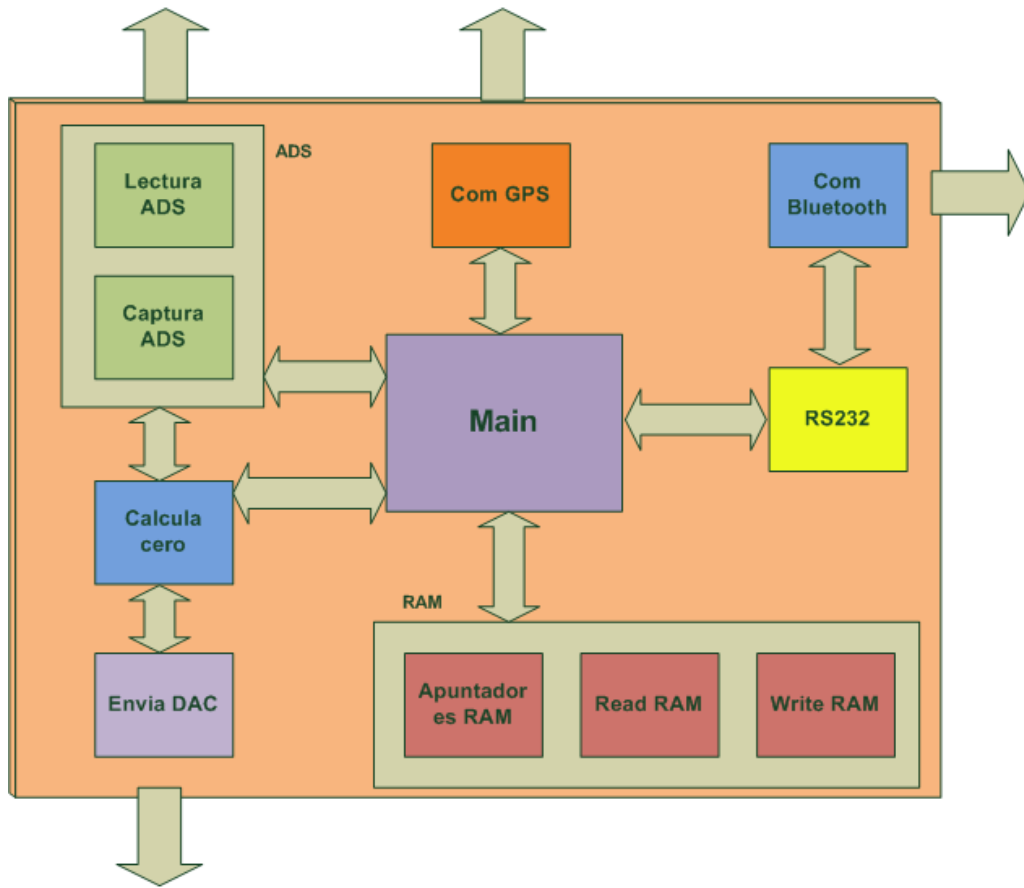
### 3 El sistema de control de muestras (Firmware)

El desarrollo del sistema de control de las muestras para su posterior interpretación y manejo puede dividirse en dos bloques fundamentales: La interfaz de usuario y el sistema de adquisición, ambos comunicados entre sí de forma inalámbrica.



En este modelo, es el sistema de adquisición el dispositivo que tiene el control de la comunicación, respondiendo solamente a aquellas peticiones validas realizadas desde el dispositivo inalámbrico. A continuación se dará una breve explicación de los diferentes componentes concernientes al sistema.

### 3.1 Diagrama de procesos



El diagrama de procesos mostrado con anterioridad muestra los diferentes bloques que componen al sistema de adquisición de muestras (Firmware), así como su interacción dentro de dicho modelo. El esquema lógico demuestra que las principales salidas del sistema están dedicadas a comunicación y adquisición de datos. Clara omisión de esto sería la comunicación del bloque controlador del DAC, cuya función sería la de realizar nivelaciones de voltaje a uno de los transductores, acorde al modelo electrónico.

El control del circuito correspondiente al ADS viene a través de dos subrutinas principales, las cuales controlan el proceso de lectura y captura de la información obtenida por el dispositivo. La memoria RAM, debido a la importante labor que ejecuta, es considerada directamente como un componente adicional del sistema de adquisición, en vez de un medio de comunicación con un dispositivo. Finalmente el control de cada una de las subrutinas viene siendo controlada de forma directa por el ciclo principal (main), y este a la vez opera de acuerdo al set de instrucciones que hayan sido recibidas por el canal de comunicación serial (RS232) vía bluetooth.

## 3.2 Herramientas de desarrollo

En el desarrollo de este sistema se utilizaron diferentes aplicaciones de software para cada fase del proyecto. Aquí se presentara una lista de dichos programas.

PIC C COMPILER .- Compilador de CCS que puede ser usado en el ambiente de desarrollo integrado MPLAB. Con este programa se realizó la programación del firmware en código C

MPLAB.- Ambiente de desarrollo integrado el cual fue utilizado para la creación del proyecto cuya finalidad seria descargar el código del programa desarrollado en el PIC C COMPILER al microprocesador. Esto lo realiza mediante el dispositivo PickIt 3.

PROTEUS-ISIS.- Simulador de circuitos electrónicos el cual ayudó a realizar pruebas de código sobre un ambiente controlado, con la finalidad de detectar posibles errores de programación y funcionamiento.

## 3.3 Protocolos de comunicación

Los protocolos de comunicación que se implementaron en el Firmware del PIC fueron dos: El protocolo SPI y la interfaz de comunicación serial EUSART. En el desarrollo de este capítulo no se abordará la comunicación inalámbrica bluetooth, esto debido a que el componente HC-06 es el responsable de configurar todos los parámetros necesarios para generar la comunicación inalámbrica.

### 3.3.1 El bus SPI

En cuanto a los enlaces de comunicación controlados por el Firmware, destaca la vía de comunicación SPI. Abreviación de Serial Peripheral Interface, el uso del bus SPI representa cerca del 90% del uso total del uso de recursos y tiempo del procesamiento del PIC. Esto debido a que el sistema de adquisición de muestras fue diseñado para poder manejar capturas consecutivas a lo largo del tiempo, lo que impacta en un alto índice de enlaces de comunicación serial vía SPI para realizar el control del convertidor ADS y el almacenamiento de las capturas obtenidas en la memoria RAM principalmente.

El método de funcionamiento de los dispositivos con capacidad de comunicación SPI ha sido detallado con anterioridad en el capítulo concerniente al sistema electrónico, por lo que en este capítulo hablaremos de las señales de control que deben ser usadas para el manejo de la comunicación. De las 4 señales requeridas, una debe ser dedicada exclusivamente

para el control del habilitador del dispositivo a comunicarse: Comúnmente denominada CS, cada uno de los elementos debe tener asignado un control del PIC con un pin dedicado, con el objetivo de que no hayan colisiones de datos en el bus.

Existe una línea adicional encargada de establecer los pulsos de reloj necesarios para realizar la transmisión de la información (SCLK). El dispositivo maestro (PIC) es el encargado de la transmisión de dichos pulsos para obtener los datos a petición.

Finalmente otras 2 líneas son dedicadas a realizar el envío y recepción de datos (MOSI y MISO respectivamente). Cabe destacar que en este modelo el microprocesador actúa como dispositivo maestro, mientras que todos los demás dispositivos son esclavos a la espera de las señales del control y los pulsos regulados por el PIC.

En cuanto a la manera en la que el tren de bits es recibido por cada dispositivo (tanto esclavo como maestro) depende principalmente de sus características de fabricación, teniendo 4 modelos básicos soportados por el bus SPI:

- Con el flanco de subida sin retraso.
- Con el flanco de subida con retraso.
- Con el flanco de bajada sin retraso.
- Con el flanco de bajada con retraso.

### **3.3.2 Comunicación serial EUSART**

El Enhanced Universal Synchronous / Asynchronous Receiver/Transmitter implementado en el micro controlador se encarga de controlar los puertos en serie y generar las interrupciones al PIC cuando exista una petición de comunicación. Los niveles de voltaje del PIC son TTL, acordes a las necesidades del circuito electrónico, y este cuenta con un registro de almacenamiento de datos entrantes que al recibir los datos genera una interrupción al sistema.

La transmisión y recepción de información se realiza mediante dos pines dedicados (Tx y Rx) y debido a que funciona mediante interrupciones, no necesita una señal de reloj de emparejamiento.

El compilador CCS contiene todas las funciones necesarias para realizar el manejo de la comunicación serial, por lo que se simplifica el control del puerto. Además, una de las características adicionales que nos brinda es el manejo de errores del puerto, con lo cual se puede realizar una limpieza automática del mismo.

El sistema trabaja a una velocidad de 115200 Bauds, sin bit de paridad y con paquetes máximos de 10 caracteres. Y antes de ser recibidos por el usuario, deben ser procesados por el dispositivo HC-06 para su envío bluetooth.

### 3.3 Entradas y salidas esperadas

Las entradas y salidas esperadas se pueden observar en la siguiente tabla:

Puerto Serial			
Entrada	Salida	Instrucción esperada	Comentario
0	"M"   "A"	NULL	Muestra modo actual
1	NULL	calcula_cero(0x7fff);	Ajuste de la celda de carga
2	"%lx %lx\n"	envia_ads8341()	Muestreo de ch0 y ch1 a petición
3	"%lx %lx\n..."   NULL	lectura_ads(), leeGPGGA()	Envío de tren de muestras capturadas, si es que hay un ensayo
4	"%s\n"	leeGPGGA();	Lectura del valor almacenado del GPS
5	NULL	leeGPGGA();	Almacenamiento del valor actual del GPS
6	NULL	m_auto=0	Cambio a modo manual
7	NULL	m_auto=1	Cambio a modo automatico
8	NULL	golpe=1	Activación de muestra manual
9	NULL	m_auto=2	Activación de captura de tiempos

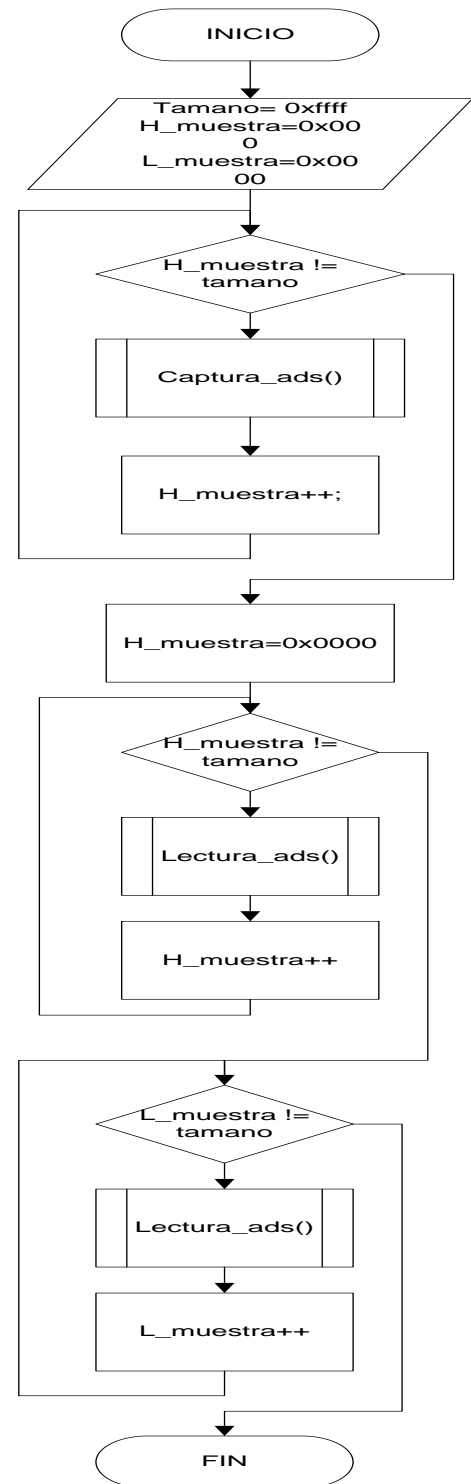
Como se puede analizar, cada instrucción busca tener un estándar específico en cuanto al tamaño y tipo de dato que se envía, con el objetivo de que se puedan desarrollar interfaces de usuario de diferentes Apis de forma intuitiva.

### 3.4 Algoritmos

A continuación se adjuntan los diferentes diagramas de flujo correspondientes a cada uno de los algoritmos de los bloques que componen al sistema junto con una breve explicación del mismo.

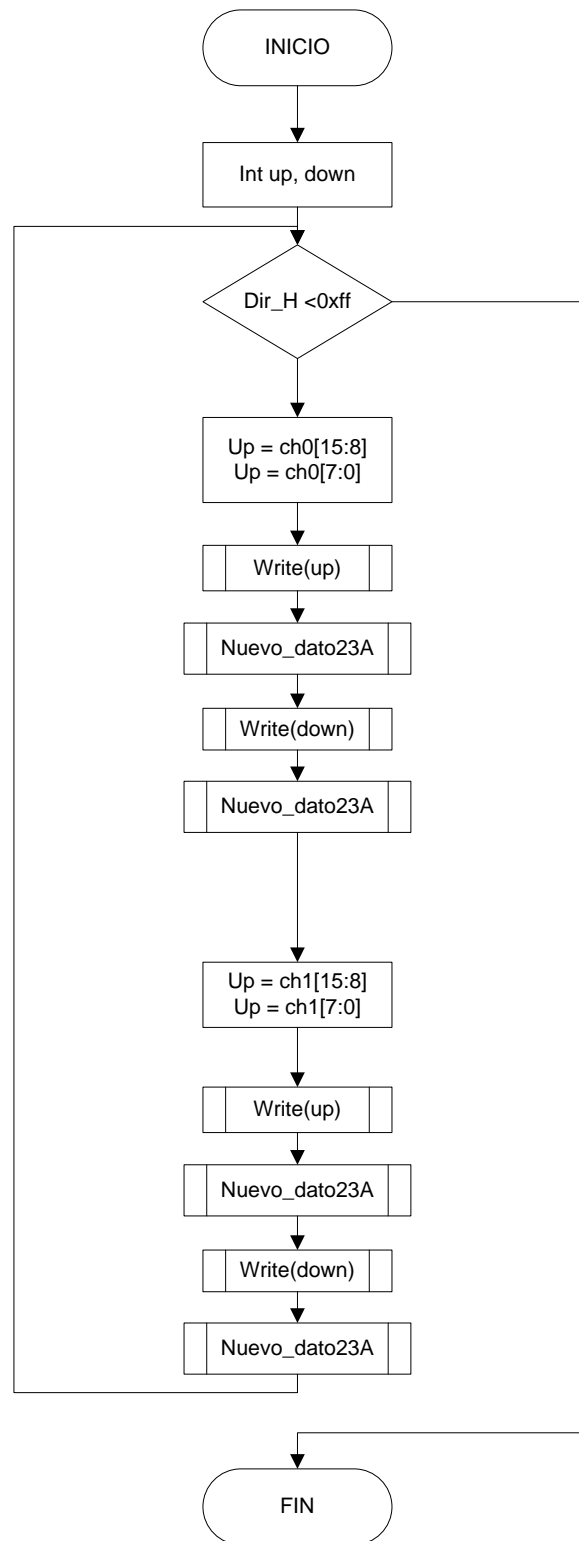
## Captura

El proceso de captura inicia cuando se recibe una muestra con altos niveles de voltaje (modo automático) o cuando se realiza un ensayo a petición (modo manual). Se realiza el proceso de muestreo después del golpe, tomando la mitad del total de muestras requeridas, para después con ayuda de los apuntadores, enviar los datos obtenidos después del ensayo junto con las muestras tomadas antes de la misma.



## Captura ads

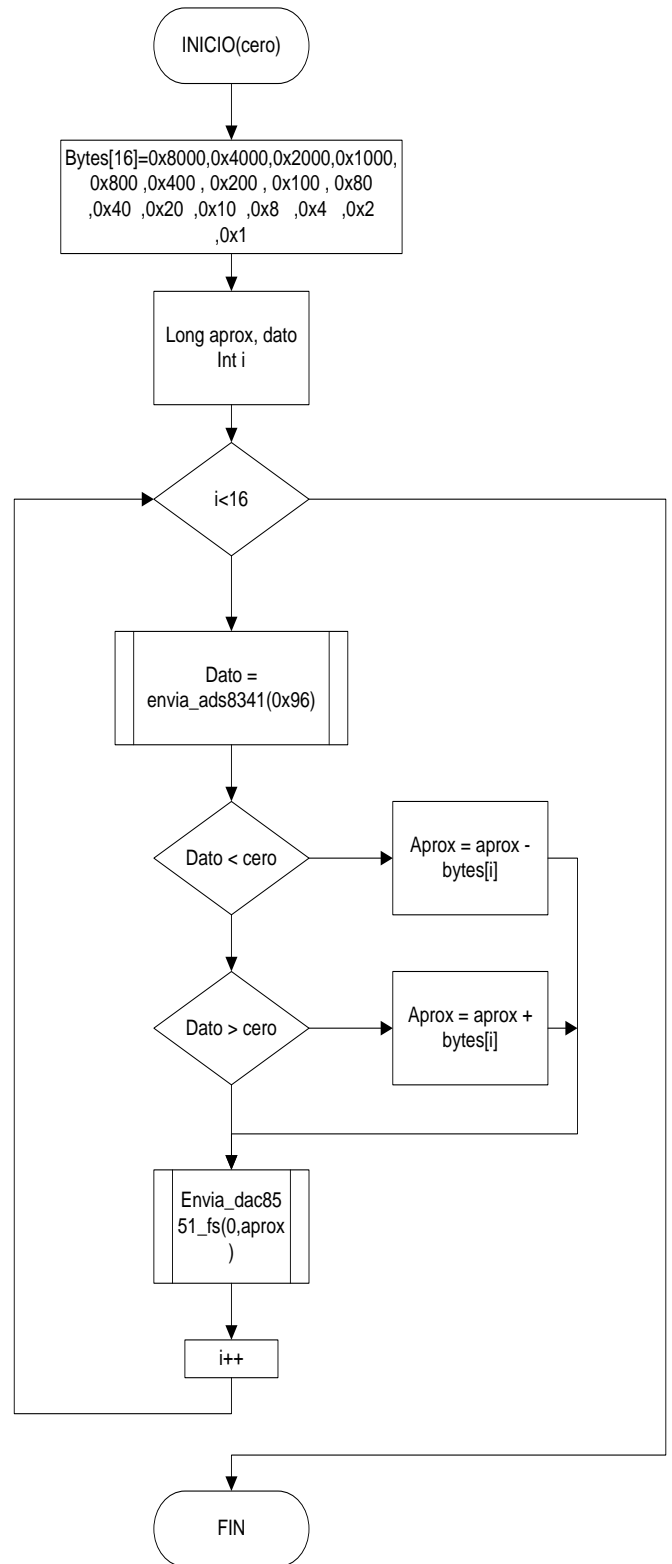
El proceso de captura del convertidor ADS hace uso de las rutinas de manejo de apuntadores para guardar la información recopilada en la memoria RAM del sistema. Primero realiza un muestreo de los canales para después utilizar la dirección actual del apuntador y guardarla en la memoria. Es almacenada en tramas de 8 bits por localidad, usándose un total de 4 localidades para guardar una captura completa.





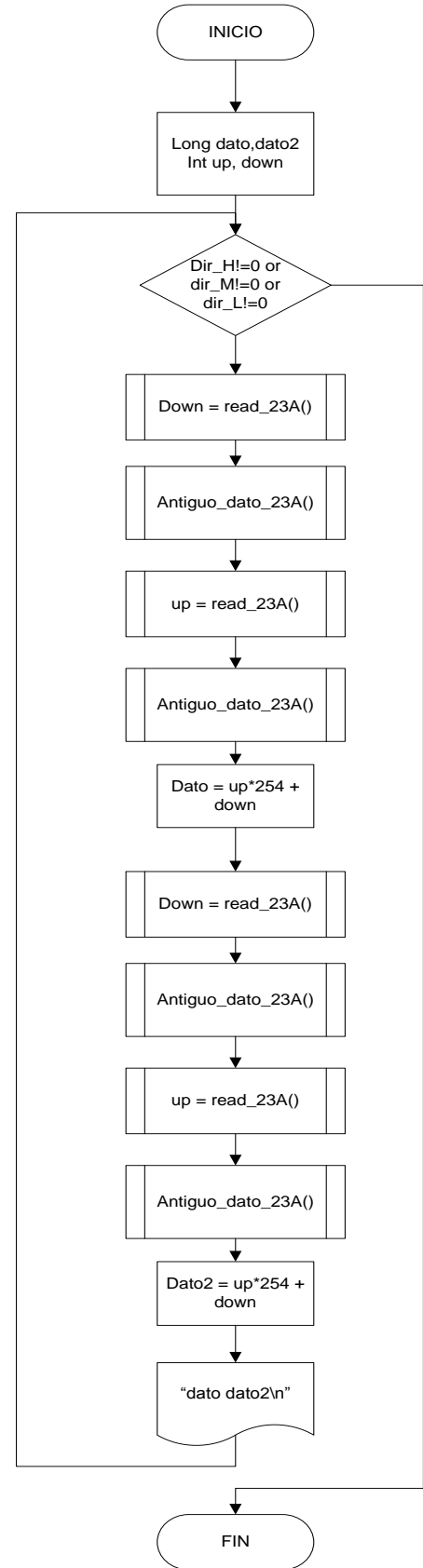
## Ajuste cero

La rutina de ajuste de cero funciona mediante el algoritmo de aproximaciones sucesivas, en el que se hacen modificaciones bit a bit del convertidor DAC hasta lograr estabilizar el sistema. La correcta aproximación depende de las capturas que se realicen del ADS al momento de realizar el ajuste, por lo que el sistema deberá estar en reposo absoluto.



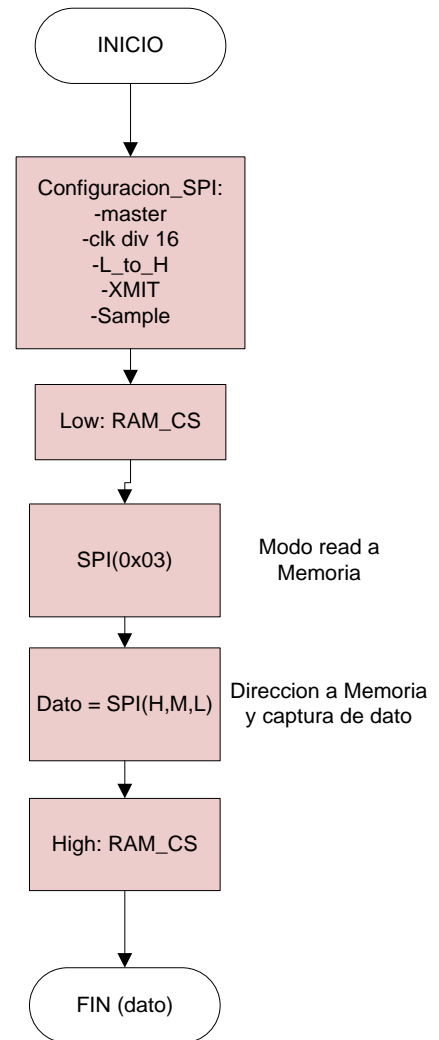
## Lectura ads

La lectura del ADS es el proceso inverso a la escritura. Se analizan los apuntadores y se obtienen los datos almacenados en la memoria de la localidad apuntada. Recordando que por cada proceso de lectura se deben realizar cuatro lecturas a memoria, se debe de actualizar el apuntador constantemente durante cada lectura, para después hacer una concatenación de bits para volver a generar el dato.



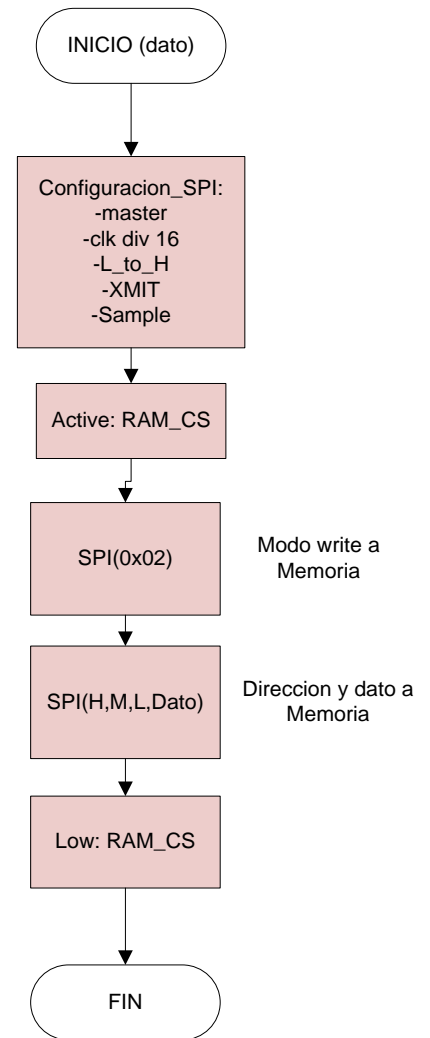
## Lectura a memoria

La rutina mostrada establece la comunicación más básica para realizar las peticiones vía SPI. Implementa los estándares definidos por el fabricante para realizar la comunicación con la memoria RAM, desde activar el habilitador de comunicación SPI del esclavo, hasta el envío del set de instrucciones para el acceso a la memoria.



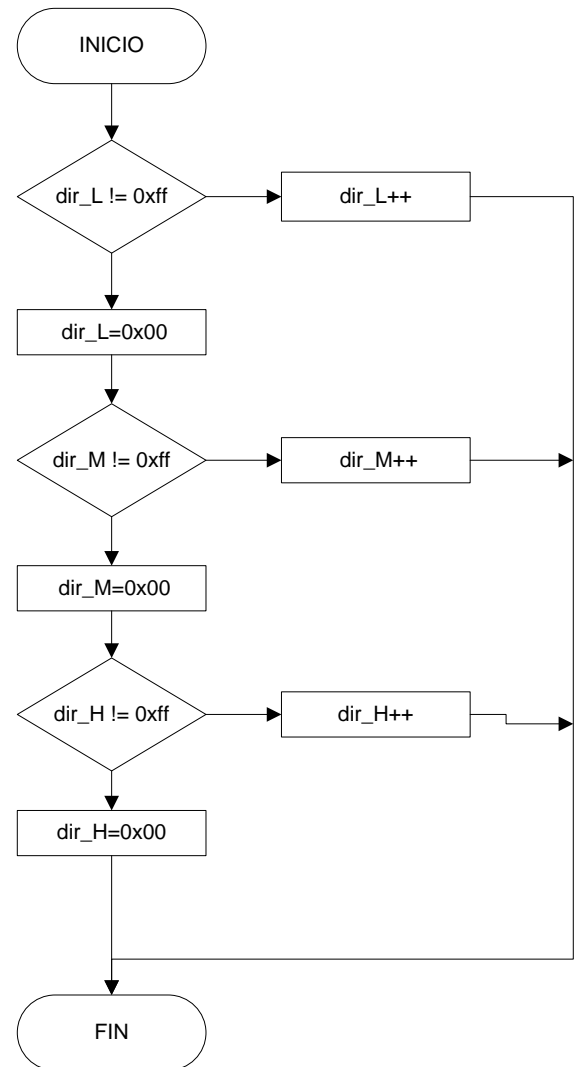
## Escritura a memoria

La escritura a memoria difiere del proceso anterior en el set de instrucciones que se necesitan enviar para obtener la información de una cierta localidad de memoria. Las instrucciones SPI propias del compilador se muestran en diferente color.



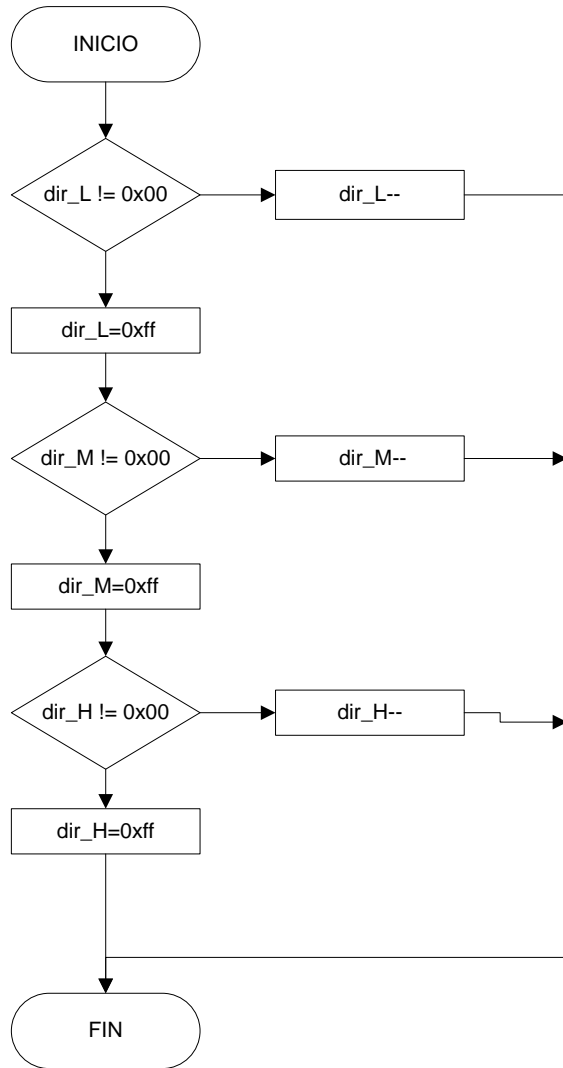
## Nuevo dato (Apuntador)

Rutina básica que se encarga de realizar el incremento del apuntador a memoria. Debido a que la memoria RAM tiene muchas localidades de memoria, es necesario contar con varios bytes concatenados para llevar el control. Cuando los 3 bytes alcanzan el límite de conteos, vuelven a su valor inicial (cero) asemejando el acceso de anillo del dispositivo utilizado.

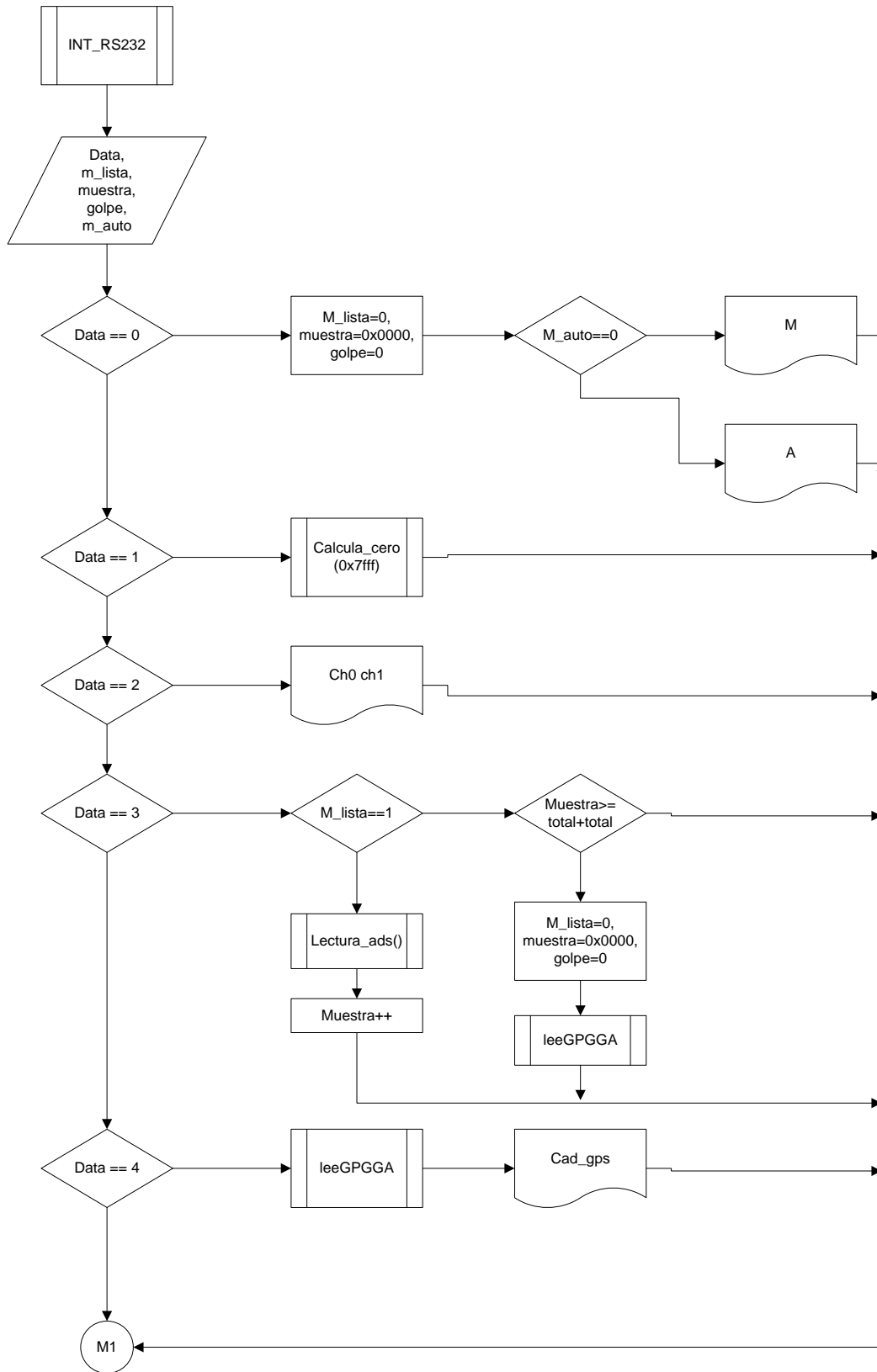


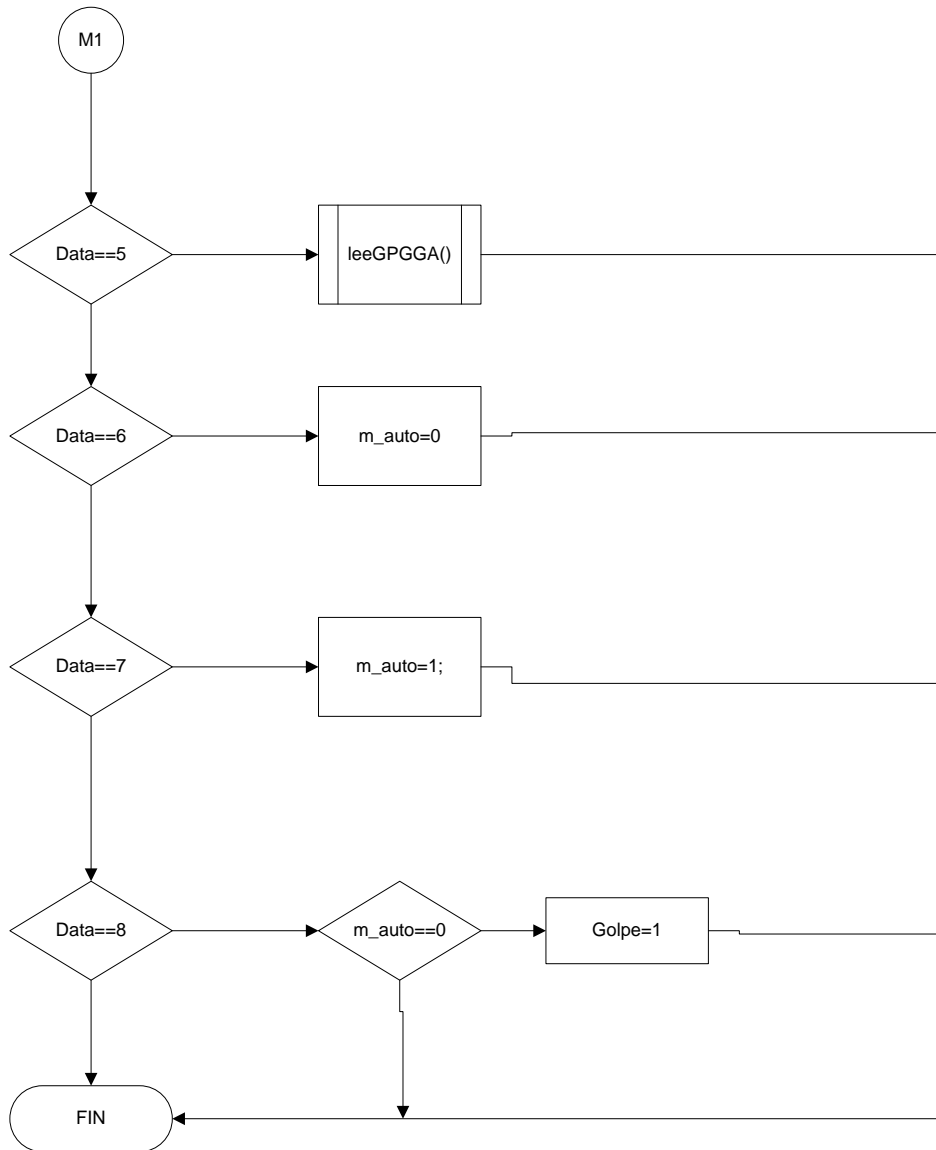
### *Antiguo dato (apuntador)*

Rutina diseñada para decrementar el apuntador a memoria. Esto es usado cuando se recuperan las muestras y se necesita relocalizar el apuntador en el dato anterior. Funciona de una manera similar al algoritmo del nuevo dato, y su recorrido se hace igualmente a manera de anillo.



# Manejo del RS232

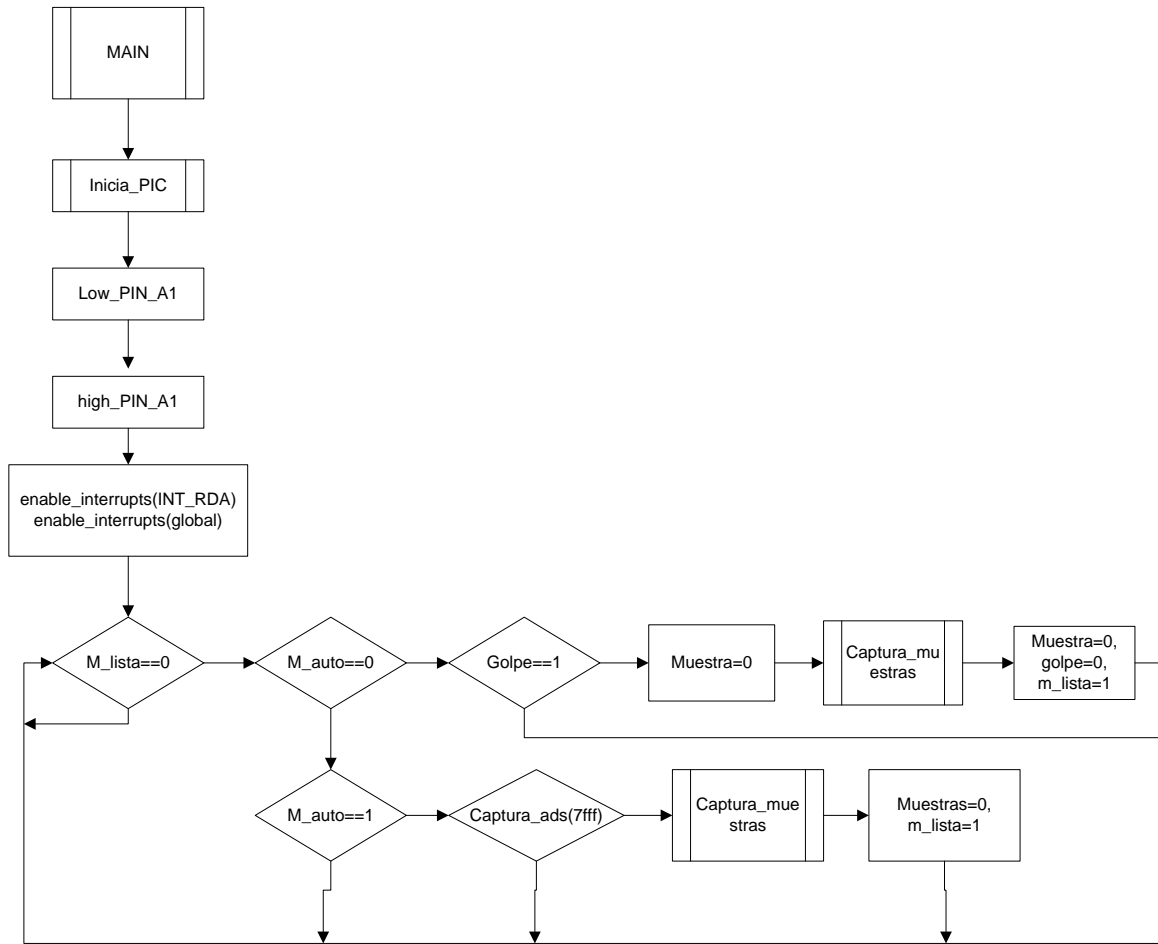




El algoritmo mostrado es implementado una vez que el puerto serial genera la interrupción por una nueva instrucción. Se revisa si la instrucción corresponde con alguna permitida y si es el caso realiza la operación solicitada. Las operaciones varían desde el cambio de alguna bandera de control, hasta la realización de una función elaborada del sistema.



## MAIN



La rutina del ciclo main contiene las instrucciones que realizará el PIC en reposo. Como puede observarse, en un inicio genera las directivas con las que trabajará el micro controlador y habilita las interrupciones a utilizar. Luego de esto entra en un ciclo infinito donde dependiendo si está en modo automático o manual, capturará una serie de muestras del sistema.

### 3.5 Detalles de implementación y programación

El desarrollo del código del controlador del sistema de adquisición requirió la aplicación de una serie de algoritmos y esquemas de trabajo, lo cual facilitó la labor de creación del código necesario para operar el sistema.

Para lograr un mejor entendimiento sobre la funcionalidad de cada uno de los módulos desarrollados, en este apartado se procederá a explicar de forma general el modelo de funcionamiento del sistema.

El módulo de control del PIC funciona a manera de servidor, el cual brinda los servicios necesarios para obtener información de la tarjeta de adquisición. Esta solicitud de información o de instrucciones se hace mediante un envío específico de comandos.

En un inicio se debe establecer la comunicación entre la tarjeta de adquisición y el dispositivo inalámbrico vía bluetooth. Para este fin el circuito integrado HC-06 nos ofrece la posibilidad de realizar el emparejamiento con un nombre de dispositivo y una contraseña.

Una vez que el sistema ha realizado el emparejamiento, el modelo está listo para recibir las diferentes instrucciones a petición de la interfaz de usuario. Debido a la modularidad del sistema, las instrucciones pueden ser introducidas desde línea de comandos, haciendo uso del puerto de comunicación dedicado. De cualquier manera, al realizar la comunicación se establece un orden en el envío de datos, donde el sistema permanecerá en reposo hasta recibir una instrucción válida. En caso de que la instrucción tenga validez, el sistema procederá a hacer la operación deseada (especificada en el cuadro de entradas y salidas) para luego, en caso de ser necesario, enviar una confirmación del comando enviado.

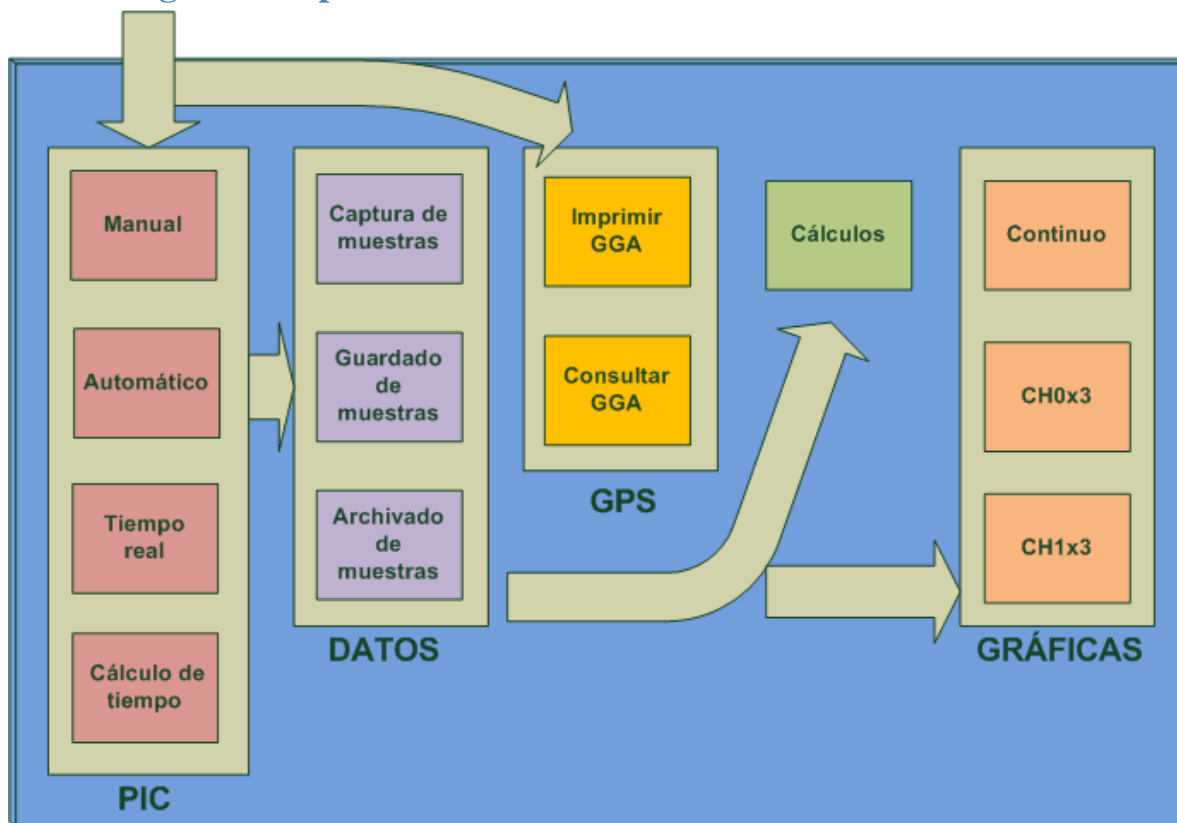
Como se puede observar, el sistema trabaja directamente a petición del usuario final que establece la comunicación, en un esfuerzo por lograr que la persona que opera el dispositivo tenga un control absoluto sobre las acciones del sistema de adquisición.

Finalmente, si se desea analizar de forma personal el código de programación de cada uno de los algoritmos, en el apéndice se podrá observar el código en lenguaje c del sistema.

## 4 La interfaz de usuario

La interfaz de usuario será el software desarrollado para cualquier dispositivo con capacidad bluetooth y con SO Windows. Dicha interfaz tendrá la capacidad de enviar peticiones y recibir información con el sistema de adquisición de muestras una vez que se haya generado un enlace de comunicación entre estos. En la siguiente imagen se puede apreciar el diagrama de bloques de dicho sistema:

### 4.1 Diagrama de procesos



Hay un total de cuatro procesos fundamentales que se encargan de realizar las operaciones de adquisición y control del flujo de datos, así como de generar las peticiones al sistema de adquisición de muestras.

Tanto el módulo de comunicación con el PIC, como el módulo de control del GPS tienen la capacidad de realizar peticiones de información al sistema de adquisición y así generar reportes de status y control del mismo.

El módulo de datos tiene la función principal de recibir las muestras obtenidas de las pruebas de impacto realizadas para su posterior almacenamiento en memoria y/o en archivos de texto planos. Finalmente el módulo de gráficas establece la capacidad de mostrar los datos del muestreo de forma visual para su interpretación.

## 4.2 Herramientas de desarrollo

Las herramientas de desarrollo dedicadas a la realización de la interfaz de usuario han sido las siguientes:

PROTEUS-ISIS.- Simulador de circuitos electrónicos el cual ayudó a realizar pruebas de código sobre un ambiente controlado, con la finalidad de detectar posibles errores de programación y funcionamiento.

VSPE.- Creador de puertos virtuales, con el cual se pudieron realizar pruebas de comunicación entre los módulos del sistema

LABVIEW.- Plataforma de desarrollo en la cual se utiliza lenguaje g para la creación de diferentes trayectorias de procesos.

## 4.3 Características generales del dispositivo inalámbrico

En este apartado se hablará de características que debe tener el dispositivo que albergará la aplicación que fungirá como interfaz de usuario con el sistema de adquisición.

Debido a que el software fue diseñado con la herramienta LabView, el sistema operativo deberá ser Windows. También se deberá contar con el instalador adecuado para los drivers que serán usados. En este caso los correspondientes a la comunicación serial de Labview, llamada VISA y el motor que da soporte a las aplicaciones, el cual es NI-DAQ. Ambos pueden ser provistos por el archivo ejecutable que se obtiene al crear el proyecto correspondiente a la interfaz de usuario.

Finalmente, el último requerimiento del dispositivo será que debe tener soporte para comunicación bluetooth.

Como podrá intuirse, hay una gran cantidad de dispositivos en el mercado que cumplen con los requisitos de compatibilidad del sistema. Para el desarrollo de las pruebas posteriores, se hizo uso de una computadora portátil VAIO y de una tableta ASUS, con SO Windows 7 y 8 respectivamente.

## 4.4 Entradas y salidas esperadas

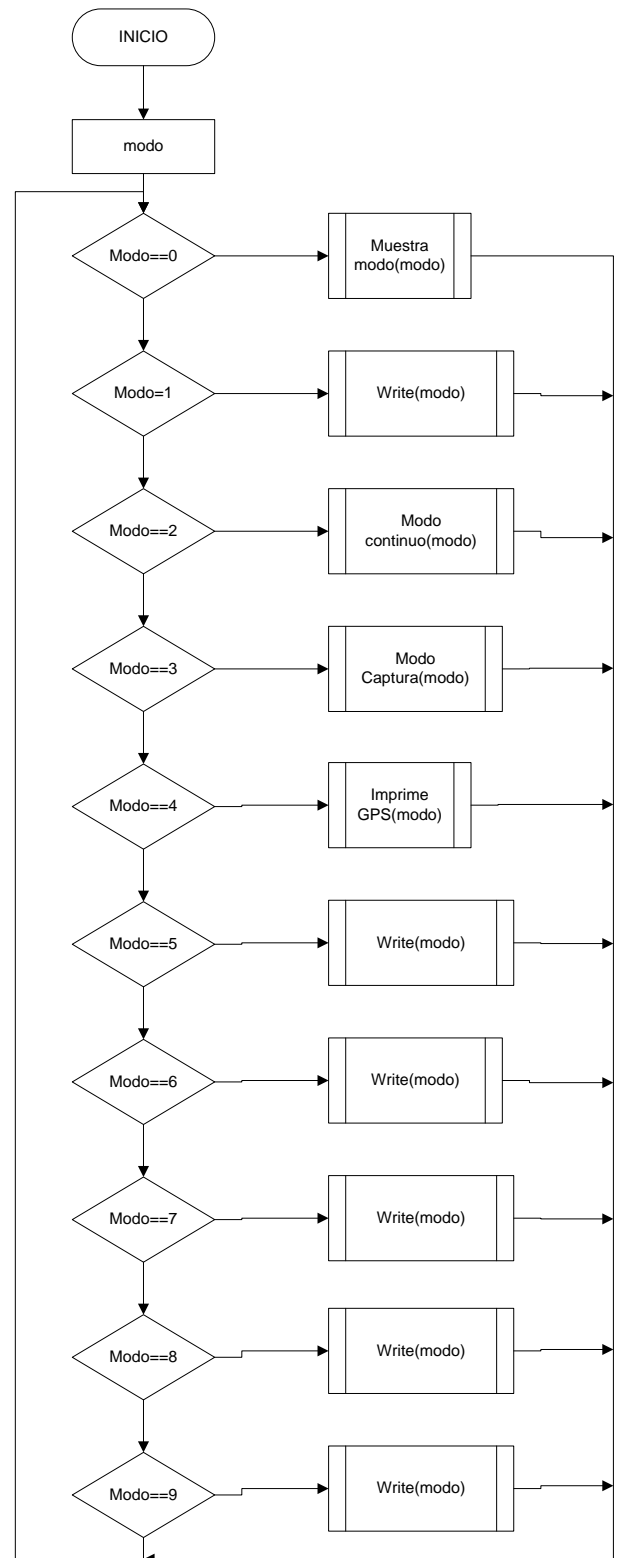
Analizando la interfaz de usuario desde el punto de vista del operador, la siguiente tabla muestra las diferentes entradas que pueden realizar el usuario con el sistema, y el comportamiento que se espera de cada envío de instrucciones.

COMUNICACIÓN CON EL USUARIO				
	Boton/Control	Envio Ser	Accion	Comentario
PIC	Continuo	2	Cambia el modo de funcionamiento a captura continua en tiempo real	La velocidad de captura depende de la velocidad de transferencia
	Manual	6	Cambia el modo de funcionamiento a captura manual	Modo manual necesita la activacion del boton "Activador" para comenzar a capturar muestras
	Auto	7	Cambia el modo de funcionamiento a captura automática	En modo automatico, el pic generará una muestra lista cuando se detecten niveles de voltaje superiores a un rango establecido
	Tiempo	9	Cambia el modo de funcionamiento a captura de muestras por segundo	Realiza el proceso normal de muestreo, registrando el numero de muestras por segundo. Durante este modo no se obtienen muestras del pic
	Activador	8	Habilita el proceso de muestreo (solo para modo manual)	Utilizado unicamente en conjunto con el modo manual
	Ajuste cero	1	Comienza la rutina de ajuste de cero	Se puede realizar en cualquier momento, menos durante el proceso de captura de
	Ver modo	0	Indica si el modo actual es Manual o Automático	Limpia cualquier muestra lista que se haya almacenado en el PIC
	Baud rate	N/A	Establece la velocidad de comunicación	EL valor por default es 9600
	VISA resource name	N/A	Establece el nombre del puerto serial a utilizar	
Datos	Captura	3	Obtiene las muestras de la prueba realizada (debe existir una prueba hecha)	IMPORTANTE: No se debe realizar ninguna otra operación con el PIC mientras este proceso esta en funcionamiento. El PIC debe tener una muestra lista realizada.
	Preparar muestra	N/A	Pone a cero el contador de muestras actual para la muestra seleccionada. Borra toda informacion anterior	Se debe tener cuidado con el uso de este control, ya que si se activa, la muestra actual que esta seleccionada perderá toda su informacion, ademas de que se perderán el número de muestras capturadas (Necesarias para guardar las muestras en los archivos txt)
	Guardar muestra	N/A	Guarda las muestras capturadas en archivos txt ubicados en el directorio de trabajo	Para ser usado, debe existir aun el número de muestras que se obtuvieron en el indicador #muestras. De lo contrario se tendrá que repetir cuando menos una muestra
	#M	N/A	Establece con que muestra se va a trabajar (1-3)	La muestra elegida será sobre la que se apliquen la mayoría de los controles de esta seccion
GPS	Captura Posicion	5	Envia la petición para guardar el valor actual del GPS en el PIC	Capturar posicion no trae a los indicadores el valor capturado
	Imprime dato GPS	4	Muestra el valor almacenado en el PIC del GPS	Imprimir posicion muestra en los indicadores el valor almacenado del GPS, que puede no ser el valor actual
Calculos	Diametro	N/A	Establece el diametro de la placa de pruebas en mm	Dependiente del sistema defléctometro
	Fuerza	N/A	Permite ingresar la fuerza obtenida de forma manual en MN	Dependiente del sistema defléctometro
	Smax	N/A	Permite ingresar de forma manual la media de los valores obtenidos por el geófono	Dependiente del sistema defléctometro
	Graficar	N/A	Habilita la graficación de los valores obtenidos de las muestras	Se obtendrán las gráficas de las muestras, siempre y cuando existan los valores

## 4.5 Algoritmos

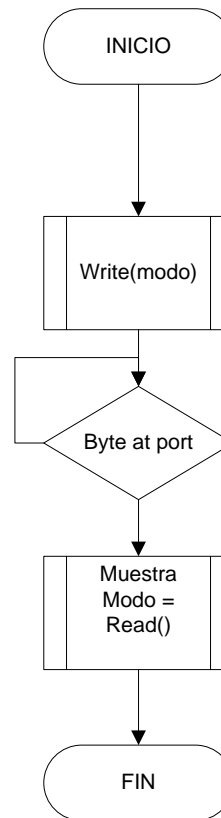
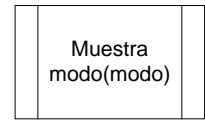
### *PIC (set de instrucciones)*

Rutina principal para la comunicación con el sistema de adquisición. En la interfaz de usuario esta se divide en dos bloques: El manejo del muestreo y la adquisición de los datos emitidos por el modulo GPS. Varios de los envíos del usuario al sistema de adquisición consisten en una instrucción que tiene como objetivo la alteración de alguna bandera de status. Son solo cuatro comandos los cuales realizan peticiones de ejecución de rutinas. Estas son el envío de muestras, la captura de los datos del GPS, solicitud de regeneración de los datos GPS, y la rutina de ajuste de cero.



## *Muestra modo*

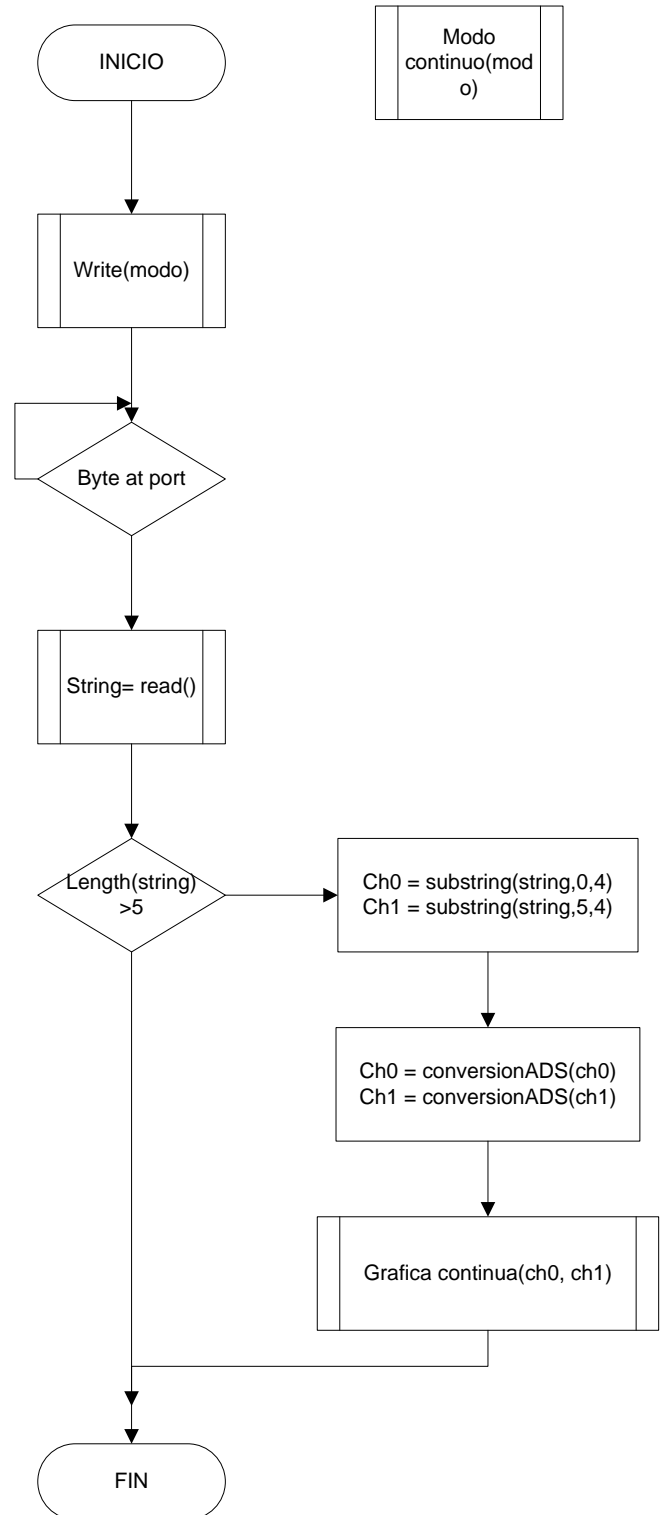
Esta rutina está ligada a la comunicación serial que se realiza con el sistema de adquisición de muestras. En esencia envía una instrucción, y espera la respuesta del PIC para saber si se está operando en modo manual o automático.



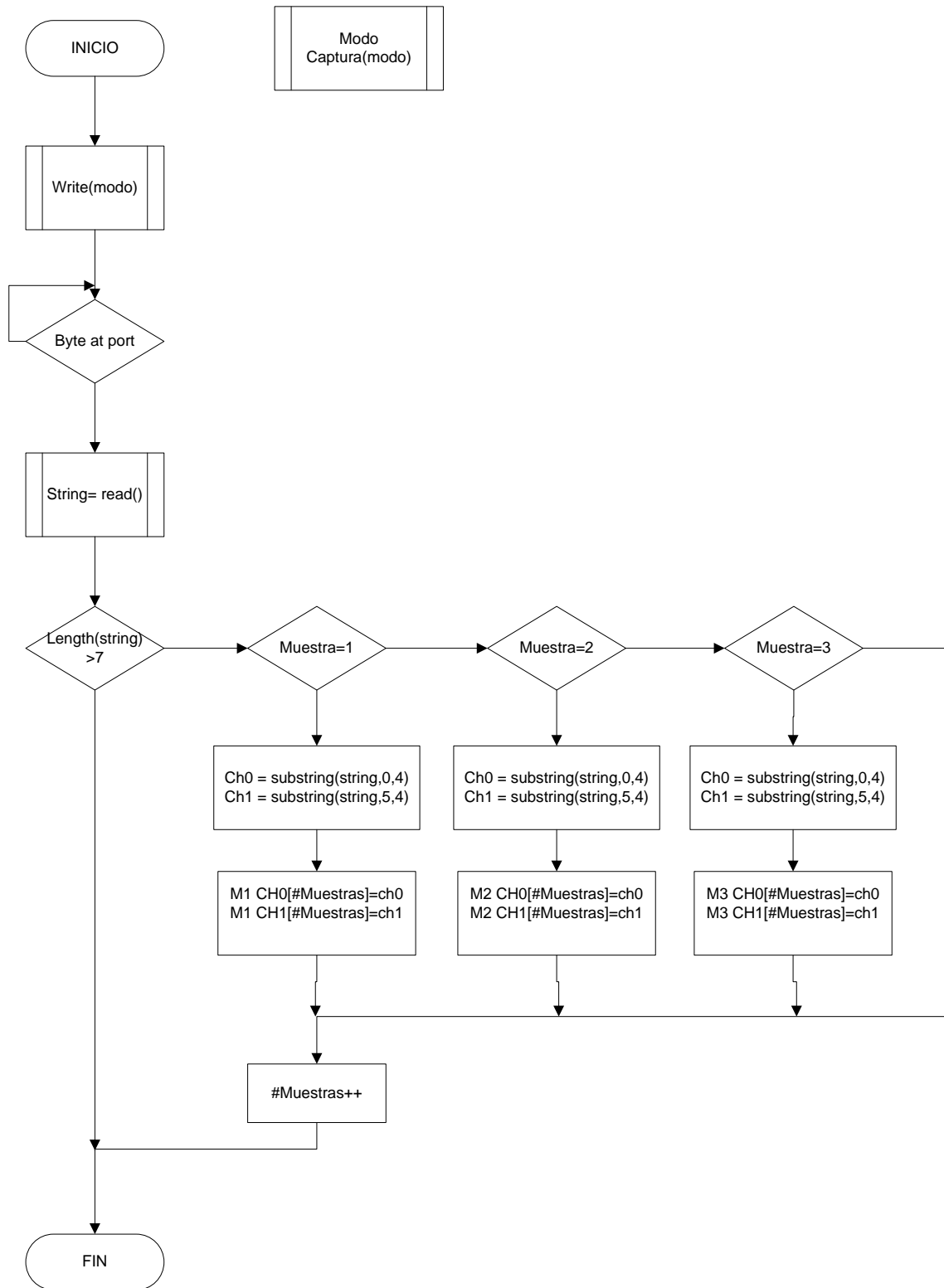


## Modo continuo

El modo continuo de operación genera un ciclo en el cual se envía constantemente una petición de muestreo en tiempo real. Dicha petición es procesada e inmediatamente graficada. Ambos canales son mostrados en la gráfica bajo el nombre de “Continuo”.



## Modo captura



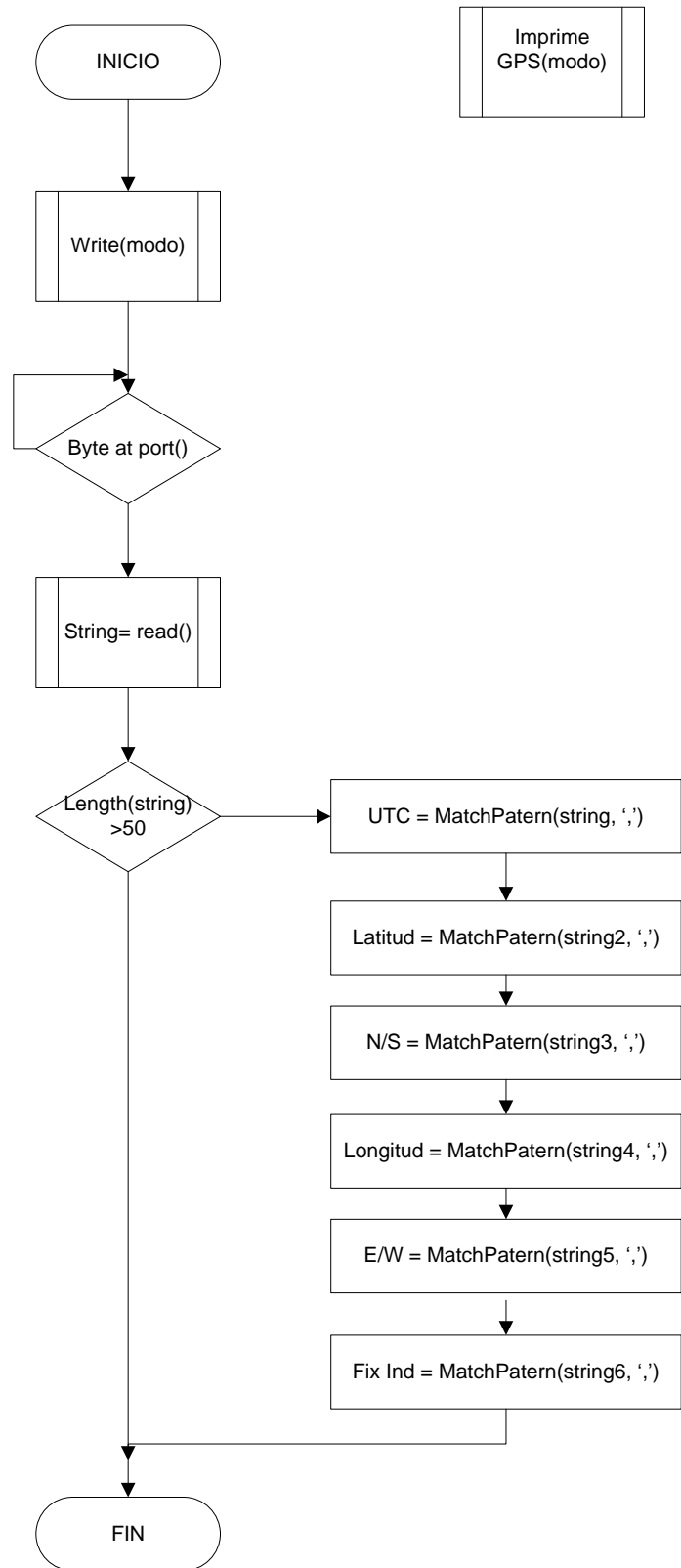
Cuando se activa la rutina de modo captura, se debe estar seguro de que se haya realizado con éxito un ensayo; de lo contrario, el PIC no tendrá ninguna muestra almacenada, por lo que no realizará ningún envío.

En caso de que exista una muestra almacenada, el sistema de adquisición procederá a hacer el envío de una muestra por cada instrucción que sea enviada, logrando así tener el control de cómo se reciben las mismas. Dependiendo de la muestra que sea seleccionada (M1, M2 o M3) estas se irán guardando en un arreglo determinado para su posterior uso.

## Imprime GPS

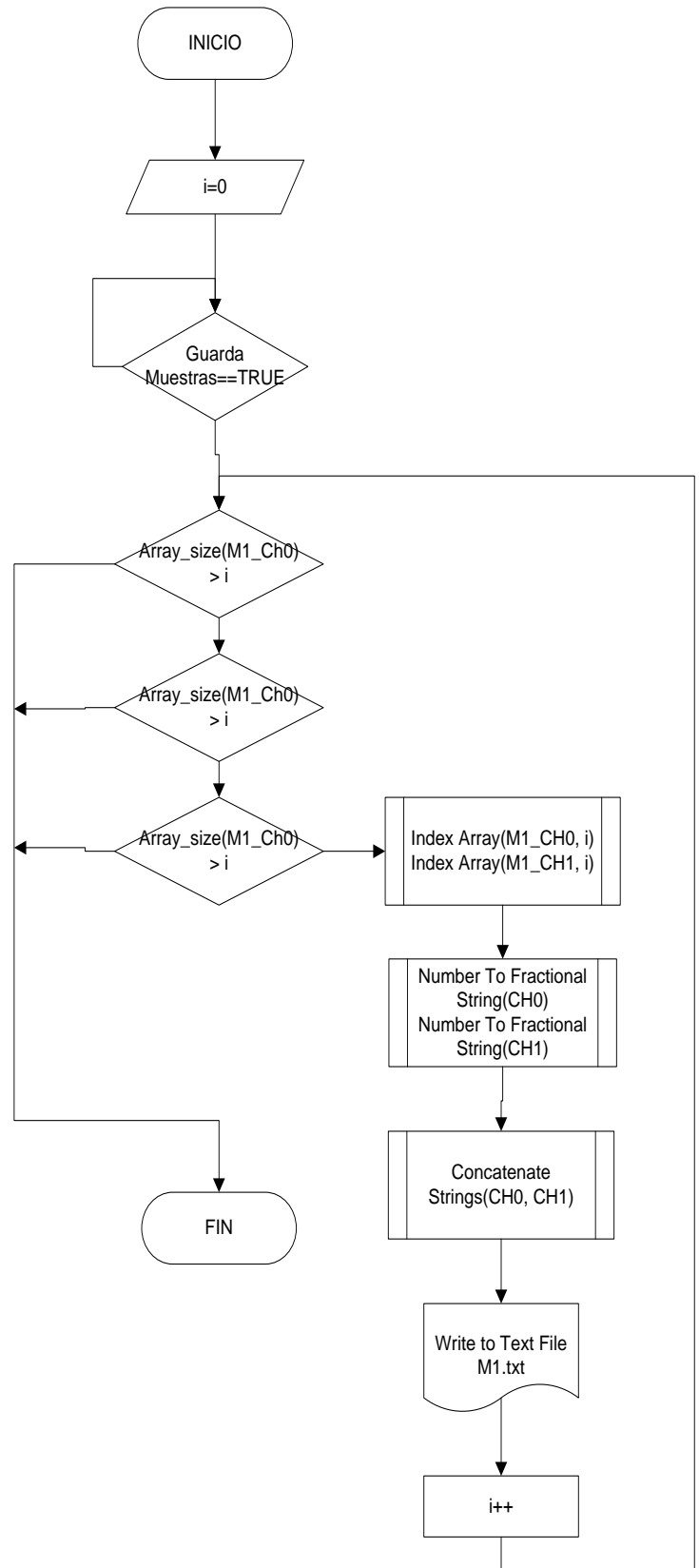
La rutina de impresión de los valores almacenados del GPS se realiza mediante una petición al sistema de adquisición. Cabe destacar que el valor del GPS se almacena dentro de una variable en el PIC, por lo que el valor obtenido al momento de hacer la captura puede que no coincida con la hora actual. Otro detalle de interés es que al momento de realizar un ensayo exitoso, el sistema además de generar las muestras pedidas, también hará una captura de los valores del GPS y los almacenara en el PIC para su posterior consulta.

Una vez que ha sido enviada la petición del valor almacenado del GPS, este es recibido por la interfaz de usuario y se divide la cadena de acuerdo al estándar NMEA. Con esto se obtiene latitud, longitud, UTC, N/S, E/O, e indicador de corrección.



## Datos

Este algoritmo es usado para realizar el procesamiento de los datos que han sido capturados desde el sistema de adquisición. Verifica el tamaño de cada uno de los arreglos obtenidos en los tres ensayos y una vez que la instrucción de almacenaje de los datos es aprobada, se procede a realizar el guardado de las muestras en documentos txt o xml. En este algoritmo solo se muestra el proceso aplicado a la muestra 1, pero el procedimiento es similar para las otras dos muestras.



## *Cálculos del sistema*

Los cálculos del sistema implementan las fórmulas asociadas al estudio del suelo. Dichas fórmulas pueden ser consultadas en la introducción del presente trabajo.

Dentro del sistema de control se ha previsto la capacidad de poder ingresar el diámetro del plato de presión, y la conversión de la muestra de voltaje de la celda de carga a fuerza. También se añadió la opción de generar valores teóricos esperados del ensayo. Lo anterior fue pensado con el objetivo de que el firmware y la interfaz de usuario pudieran ser usados con diferentes tipos de placas de presión y con una masa variable. Mas detalles sobre la implementación pueden ser analizados en el apéndice de este trabajo.

### **4.6 Detalles de implementación y programación**

La interfaz de usuario representa el contacto que el operador tendrá directamente con el sistema de adquisición de muestras. Esta interfaz fue diseñada con el objetivo de brindarle al usuario todas las características necesarias para poder realizar las pruebas de forma fácil e intuitiva.

Ahora bien, hablando del desarrollo de la interfaz, el API de LabView brinda una serie de utilidades para poder implementar diagramas de procesos y de flujo de una forma más vistosa y eficiente, logrando crear grandes proyectos en tiempos considerablemente menores. El uso del lenguaje de programación g nos brinda la oportunidad de realizar una interpretación del software a manera de circuito electrónico, con lo que se logra una visión de una forma más natural de cada uno de los procesos.

Hay un total de cuatro procesos fundamentales, cada uno con una utilidad específica acorde al esquema de bloques visto con anterioridad. Cada uno de estos procesos está corriendo de forma permanente en un ciclo, y todos son totalmente independientes entre sí con el objetivo de no generar conflictos entre ellos. Se tienen arreglos dinámicos de datos de tipo flotante para realizar el almacenamiento en memoria de todas las muestras obtenidas del muestreo del sistema de adquisición de datos (muestras del geófono y de la celda de carga). Son un total de seis arreglos ya que por cuestiones de diseño del deflectómetro, se requieren un total de tres muestras de impacto para tener un estudio completo.

## 4.7 Análisis del flujo de procesos

Debido a que cada uno de los procesos fundamentales tiene un ciclo infinito, se necesita en cada uno de ellos establecer ciertas condiciones para que el sistema no tuviera una alta carga de procesamiento. Es por ello que la interfaz de usuario permanece en un flujo constante y sin realizar alguna operación para garantizar que no se ocupen cantidades innecesarias de procesamiento. Es solamente hasta que se envía una señal de entrada que el sistema redirige el flujo del proceso hacia las diferentes operaciones implementados por LabView para realizar la operación solicitada.

El primer ciclo correspondiente al manejo de la mayoría de instrucciones que se envían al PIC está conformado por dos bloques. El primero de ellos es una estructura case la cual contiene los diferentes procesos a realizar acorde a cada una de las instrucciones que puede introducir el usuario. Cada instrucción cuenta con un botón asociado en la interfaz gráfica del usuario y dependiendo de la acción realizada, cada control actuará de forma diferente. Hay un total de nueve instrucciones válidas para interactuar con el sistema de adquisición y cada una de las acciones correspondientes puede apreciarse en la tabla de entradas y salidas esperadas.

El segundo bloque contiene rutinas de control para manejar el apuntador de los arreglos donde se han de guardar las muestras capturadas por el sistema de adquisición. En tanto el botón que controla la activación de dicho bloque no sea activado, el bloque queda en espera. Cuando el control es activado, dependiendo de la muestra seleccionada, el arreglo elegido es limpiado y el apuntador de muestras colocado en cero. Con esto se garantiza que cada muestra contendrá información totalmente independiente a otras.

El siguiente ciclo de control se encarga exclusivamente del guardado de las muestras adquiridas por el sistema en archivos de texto plano. Una vez que el botón “Guarda muestras” sea activado por el usuario, la sentencia case que está contenida en el ciclo se activará, guardando las tres muestras en tres diferentes archivos. Es recomendable no reiniciar los contadores y hacer las 3 muestras para usar este control.

El ciclo de cálculos corresponde una parte fundamental del sistema, ya que en este módulo se analizaran las muestras y se les aplicarán funciones matemáticas para conocer los diferentes valores claves necesarios para las pruebas del deflectómetro. Opera de manera continua, para analizar los cambios ocurridos en las muestras y acepta varios controles para establecer una mejor interpretación de los datos. De forma adicional tiene un pequeño simulador de valores esperados con el objetivo de hacer la comparativa con las muestras adquiridas y poder conseguir una mayor fiabilidad del sistema.

El último ciclo es el referente al proceso de graficación de las muestras. De igual manera hay un control de usuario dedicado que indica el momento en el que se comenzará a realizar la graficación de las muestras. Hay un total de cuatro graficas: una de ellas es para el proceso de muestreo continuo (con muestras tomadas directamente del puerto serial) y muestra los canales de adquisición correspondientes al geófono y la celda de carga. La segunda y tercera muestran de forma independiente el canal de la celda de carga (Referida como CH0) y el geófono (Referida como CH1). Debido a que se pueden realizar hasta tres ensayos con este sistema, en las gráficas CH0 y CH1 se pueden analizar los resultados de las tres muestras de forma simultánea. La gráfica correspondiente a la integral nos muestra la curva característica que se tiene al aplicar dicho operador sobre el canal CH0 necesario para realizar los cálculos posteriores del sistema.

#### **4.8 Análisis del acoplamiento del firmware e interfaz de usuario**

Llegados a este punto, es conveniente analizar el sistema de control en su totalidad, tomando en cuenta la interacción entre cada uno de los módulos del sistema de forma conjunta.

El ciclo main del sistema de adquisición comienza su operación en cuanto se le suministra el voltaje necesario para que comience a realizar su operación. La primera rutina que realiza es la inicialización de puertos y configuraciones de niveles lógicos de los pines del PIC para el manejo de los diferentes dispositivos electrónicos. Después de esto, hace un reset del módulo bluetooth para asegurar que el emparejamiento entre dispositivos pueda ser realizado de forma libre y con cualquier dispositivo inalámbrico.

Una vez realizado esto, el modulo entra en un ciclo permanente donde está listo para recibir las instrucciones del dispositivo inalámbrico. En un principio el Firmware opera en modo manual, por lo que no estará capturando muestras de forma continua hasta que se cambie dicho modo. Como puede observarse en los diagramas de flujo, este ciclo puede dividirse en las directivas de operación del modo manual, las directivas del modo automático y la operación de conteo de muestras.

Es hasta que el sistema de adquisición esta enlazado con algún dispositivo inalámbrico que se presta atención al manejo de interrupciones del puerto serial. Es en la rutina int\_RS232 donde se capturan las instrucciones, y se descartan o aceptan de acuerdo al formato preestablecido. Cada instrucción realiza algún cambio específico en el sistema y altera el funcionamiento del mismo. De cualquier manera, una vez atendida la interrupción y realizada la petición de la instrucción solicitada, el sistema vuelve a su ciclo continuo, quedando a la espera de alguna nuevas órdenes.



Una vez que el sistema recibe la instrucción de realizar un muestreo (modo manual) o sensa altos niveles de voltaje en los transductores (modo automático) se procede a generar un conteo de muestras específico para su posterior envío a la interfaz de usuario. Cuando se han capturado las muestras necesarias, el sistema queda en suspensión temporal. Si está en modo automático, dejará de tomar muestras hasta que los datos capturados sean enviados al usuario o se baje la bandera correspondiente.

Puede observarse que el sistema de adquisición de muestras actúa a manera de servidor, recibiendo y contestando peticiones de cualquier usuario conectado al sistema. Esta arquitectura da una gran libertad de operación, ya que si no se tuviera inclusive la interfaz de usuario, el sistema podría ser controlado vía consola de comandos. Lamentablemente la interfaz de usuario es la encargada de realizar diferentes operaciones de control sobre los datos enviados por el sistema de adquisición, por lo que todo ese desarrollo tendría que ser realizado de forma manual en caso de que la interfaz no pudiese implementarse.

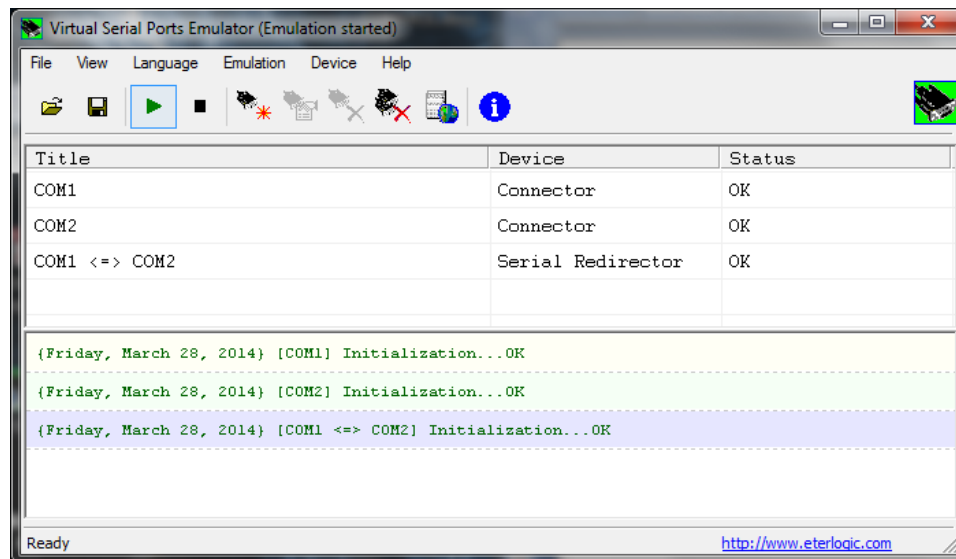
## 5 Guía de usuario

### 5.1 Manual del firmware

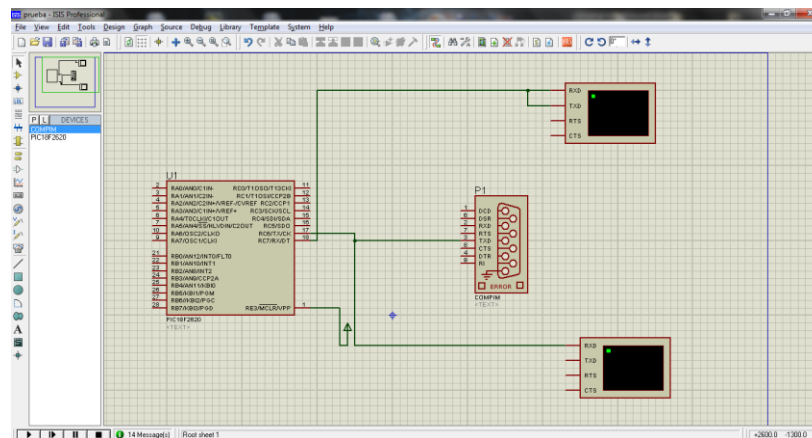
El programa del PIC18F2620 creado en Mplab, tiene una estructura de funcionamiento diseñada especialmente para interactuar con la interfaz de usuario creada en LabVIEW. Sin embargo, debido al manejo en la recepción de instrucciones, el PIC puede admitir instrucciones desde cualquier interfaz de comunicación serial vía bluetooth. Lamentablemente, como se había discutido con anterioridad, muchas de las herramientas de control que ofrece la interfaz de usuario deberían ser implementadas por el usuario, por lo que se recomienda usar el software proporcionado.

Para mostrar la metodología de funcionamiento, se realizarán una serie de simulaciones para comprobar las salidas estándar del sistema de adquisición. Dichas simulaciones serán realizadas con la generación de puertos virtuales, haciendo uso del software VSPE y la API Proteus Isis.

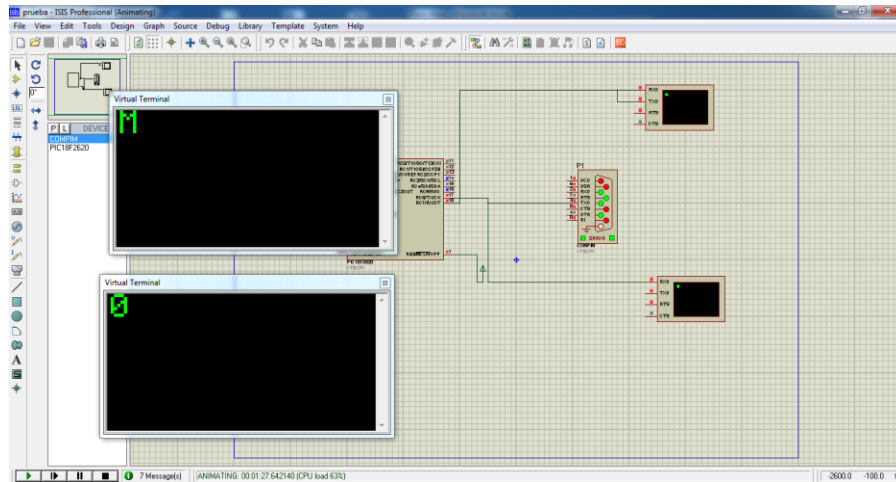
- 1) Configurando un túnel serial virtual con Virtual Port (para la prueba con el software de LabView)



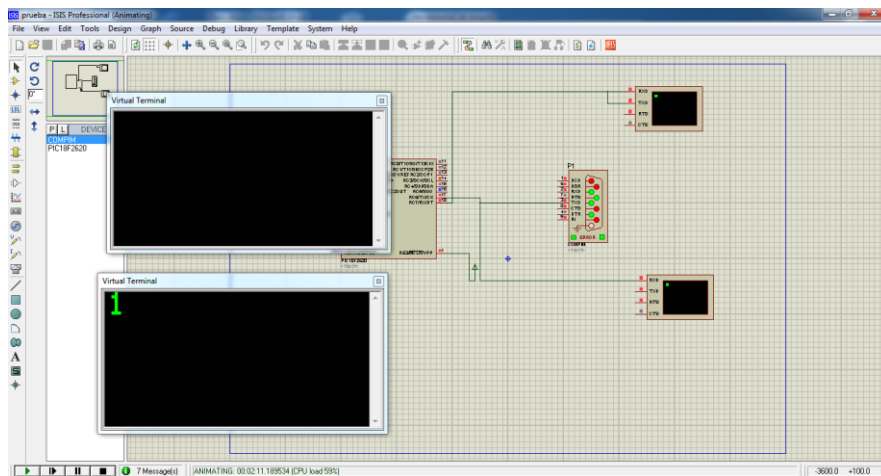
- 2) Circuito básico con el PIC18F2620 para comunicación serial con Isis Proteus



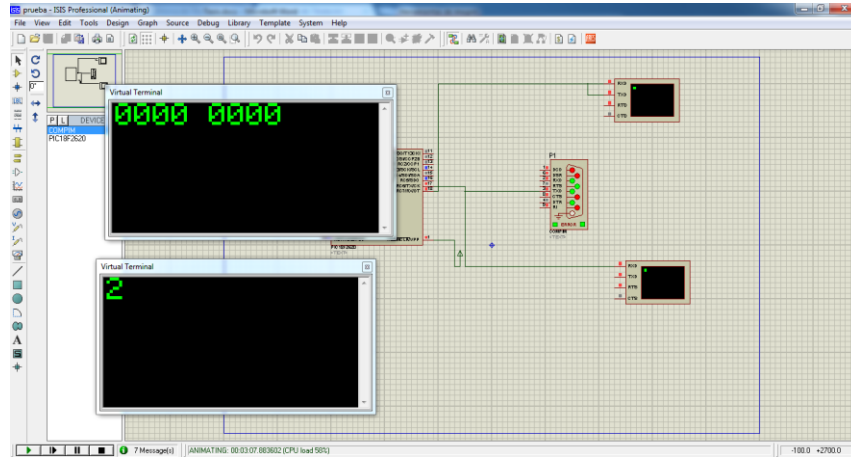
- 3) Comenzando el proceso de simulación, se tienen dos ventanas en las cuales se puede analizar la entrada y la salida del dispositivo (La ventana superior analiza la salida del PIC mientras que la ventana inferior envía la instrucción solicitada)



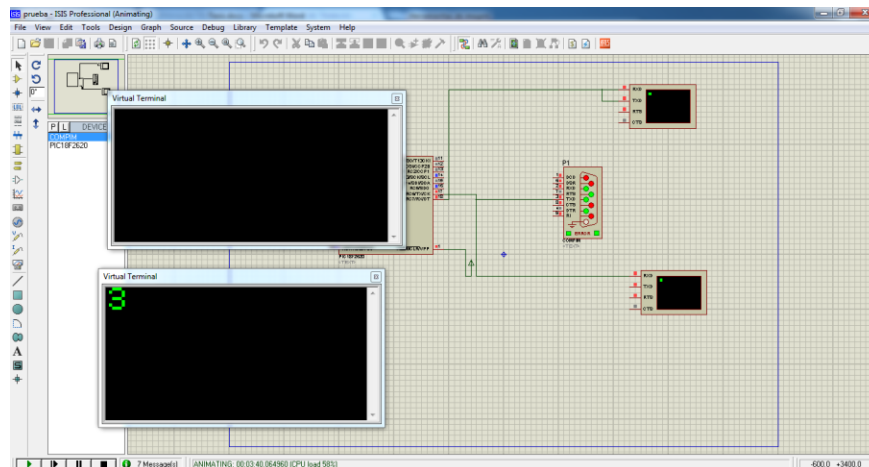
**Envío de instrucción 0:** Obtiene como salida el modo de operación (manual M o automático A)



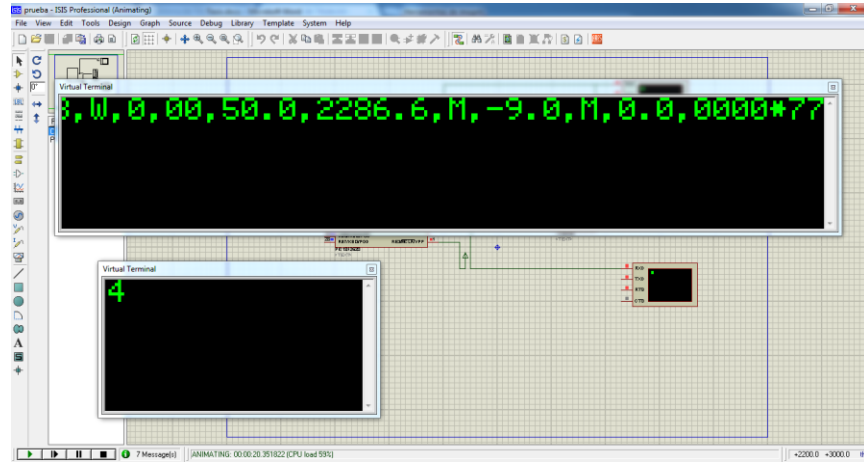
**Envío de instrucción 1:** Realiza el ajuste de cero. No existe salida del sistema



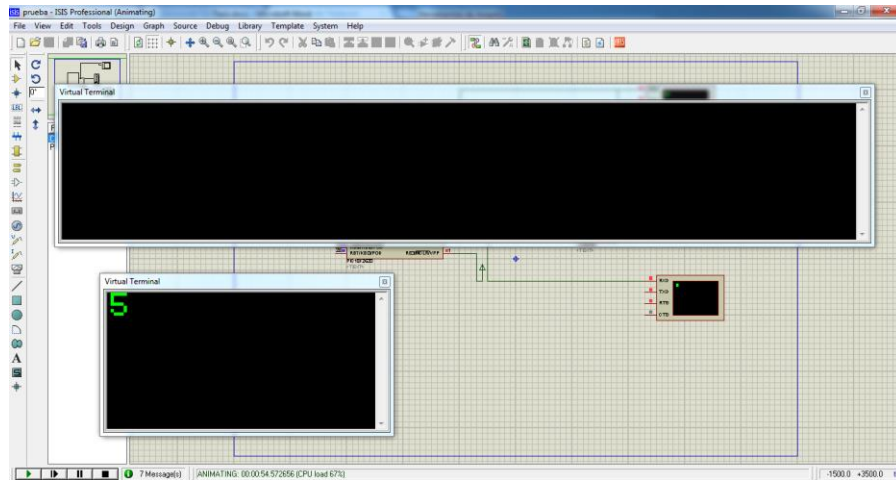
**Envío de instrucción 2:** Realiza muestreos en tiempo real. Se envía una muestra por cada instrucción recibida. Como puede apreciarse, un envío de muestra se compone por dos números hexadecimales (CH0 y CH1) separados por un espacio y con un retorno de carro al final de la cadena.



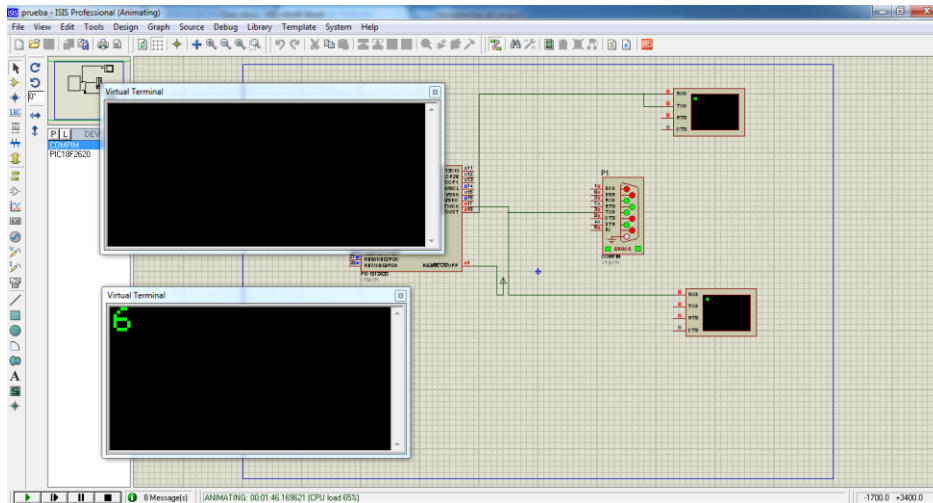
**Envío de instrucción 3:** Esta instrucción envía los resultados del último ensayo realizado, sea en modo manual o automático. En este caso, debido a que aun no se ha realizado algún ensayo, la instrucción no tiene salidas



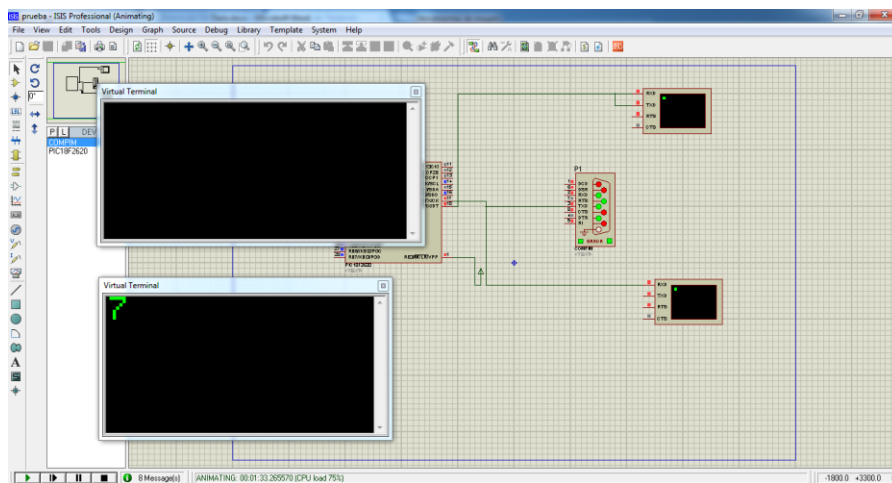
**Envío de instrucción 4:** Da como resultado el valor almacenado dentro del PIC del módulo GPS. La cadena viene acorde a los estándares NMEA y es trabajo de la interfaz de usuario realizar la interpretación.



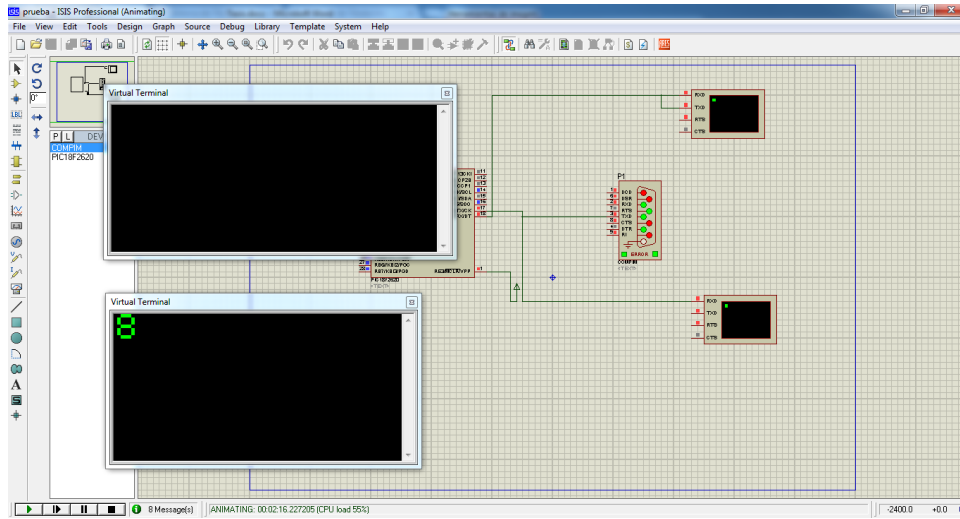
**Envío de instrucción 5:** El PIC realiza una lectura del módulo GPS y la almacena en memoria para su posterior envío. Esta instrucción no realiza el envío a la interfaz de usuario, por lo que la salida esperada es nula.



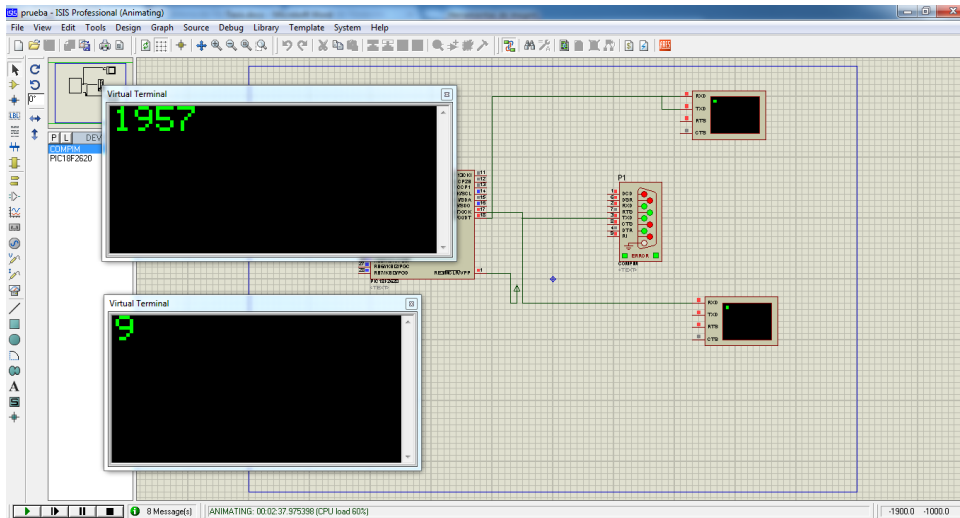
**Envío de instrucción 6:** Intercambia el modo de acción a modo manual. En este modo el sistema está a la espera de recibir la instrucción de activación del golpe para comenzar a realizar el muestreo. No hay salida esperada, pero puede enviarse la instrucción 0 para saber el modo actual de funcionamiento. Una vez que se tiene un ensayo con sus muestras listas, el PIC bloquea todos los muestreos posteriores con el objetivo de conservar los datos hasta que estos sean enviados.



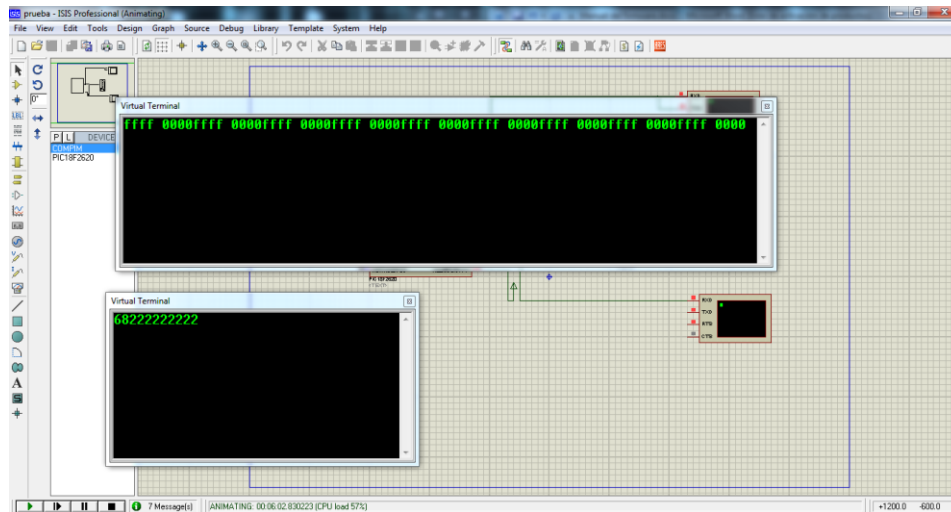
**Envío de instrucción 7:** Intercambia el modo de trabajo a automático. Mientras este activo este modo, se realizará un muestreo de forma continua y una vez que llegue un nivel alto de voltaje se procederá a hacer la captura de las mismas. En este modo se puede realizar una petición para conocer el número de muestras por segundo que se realizan. Una vez que se tiene un ensayo con sus muestras listas, el PIC bloquea todos los muestreos posteriores con el objetivo de conservar los datos hasta que estos sean enviados.



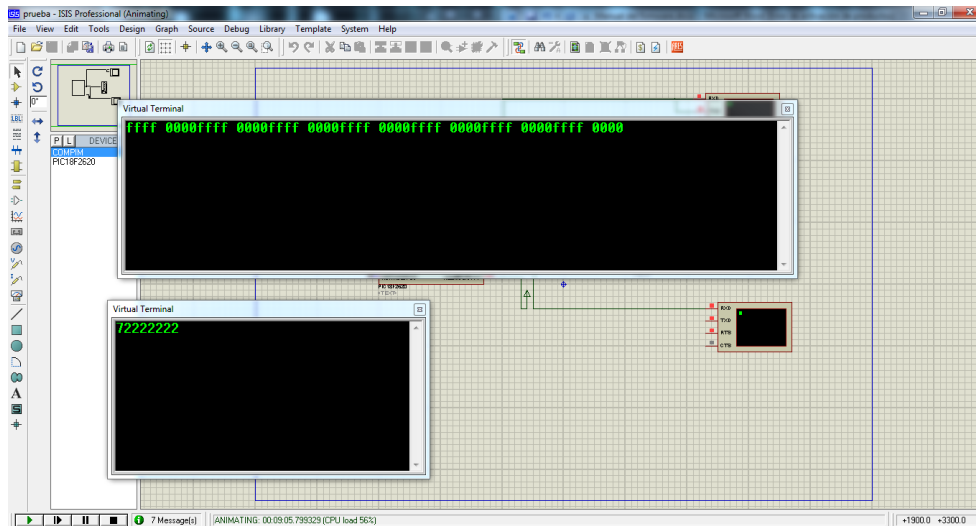
**Envío de instrucción 8:** Realiza la activación de la bandera de golpe. Si el sistema se encuentra en modo manual, se realizará un ensayo completo en el sistema y posterior al almacenamiento de las muestras, el sistema se bloqueará para mantenerlas listas para el envío a la interfaz de usuario.



**Envío de instrucción 9:** Envía el número de muestras que se realizan por unidad de tiempo. Cabe destacar que debido al modo de funcionamiento del reloj, en realidad el conteo es aproximadamente cada 1.125 segundos, por lo que se debe de aplicar un factor de corrección para conocer el tiempo en segundos. El numero enviado está en formato hexadecimal.



Cadena de instrucciones para un ensayo manual: Primero se envía la petición de cambio a modo manual (6) para que funcione con una activación a base de la bandera de golpe. Luego de mandar la señal de golpe (8) se procede a realizar el envío de las muestras haciendo uso del comando de envío de muestras (2) de forma repetida. Cuando se terminen de enviar las muestras el sistema de adquisición se desbloqueará y volverá a su funcionamiento normal.



Para el caso del modo automático, solo se necesita enviar la petición de cambio de modo (7) y una vez que el sistema haya detectado un cambio de voltaje significativo, se procede a hacer la captura de forma automática. Las muestras son enviadas de la misma forma con el comando de envío de muestras (2), y una vez que se envían todas, el sistema vuelve a su modo normal de operación.

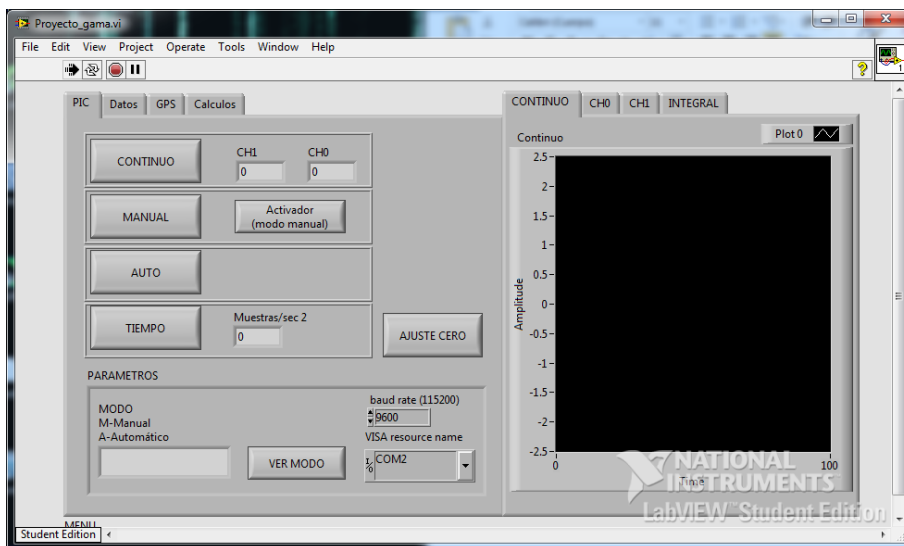


Con esto se han abordado las instrucciones básicas de operación del sistema de adquisición. En el siguiente capítulo se observará el manejo de la interfaz de usuario y su aplicación directa con el sistema visto en este capítulo.

## **5.2 Manual de la interfaz de usuario**

La presente sección tiene por objetivo dar una breve explicación sobre el funcionamiento del programa de LabView que entabla comunicación con el circuito de adquisición de muestras. En una primera instancia se buscará explicar el funcionamiento de cada control, para después mostrar de forma práctica el funcionamiento del sistema completo.

## 5.2.1 Controles del PIC



### Botones

Baudrate: Establece la velocidad de comunicación del puerto.

VISA Resourcename: Establece el Puerto serial a utilizar.

Ver modo: muestra el modo de trabajo (Manual M o Automático A).

Ajuste de cero: Envía la petición al sistema para realizar el ajuste de cero del canal correspondiente a la celda de carga.

Tiempo: Muestra el número de muestras por segundo que es capaz de capturar el sistema (Debio estar activado el modo automático con anterioridad).

Auto: Cambia el modo de operación a Automático.

Manual: Cambia el modo de operación a Manual.

Activador: Activador del ensayo para el modo Manual.

Continuo: Realiza un muestreo continuo en tiempo real.

### Indicadores

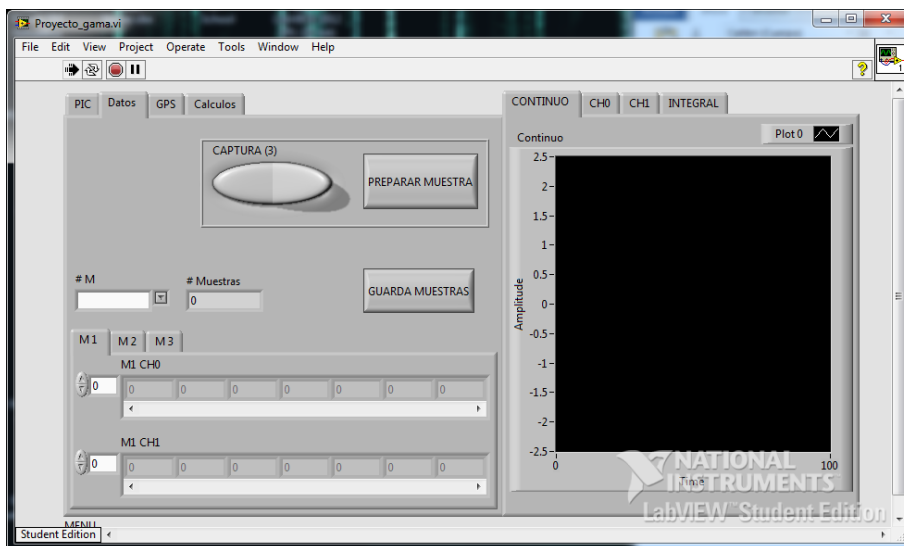
CH0: Valores actuales del muestreo continuo. Contiene la información de la última muestra obtenida del canal CH0.

CH1: Valores actuales del muestreo continuo. Contiene la información de la última muestra obtenida del canal CH1.

Muestras/sec: Muestra el número de muestras que realiza el sistema en un segundo

Modo: Muestra si se trabaja en modo manual o automático.

## 5.2.2 Controles de Datos



### Botones

#M: Establece el arreglo de muestras con el que se va a trabajar (1,2 o 3). Todos los procesos que se realizan en esta pestaña se harán dependiendo de la muestra seleccionada por este control.

Captura: Realiza la petición de muestras y las almacena (Si es que existe un ensayo listo en el sistema de adquisición).

Preparar muestra: Elimina los valores del arreglo seleccionado en #M y limpia los contadores para recibir un nuevo ensayo.

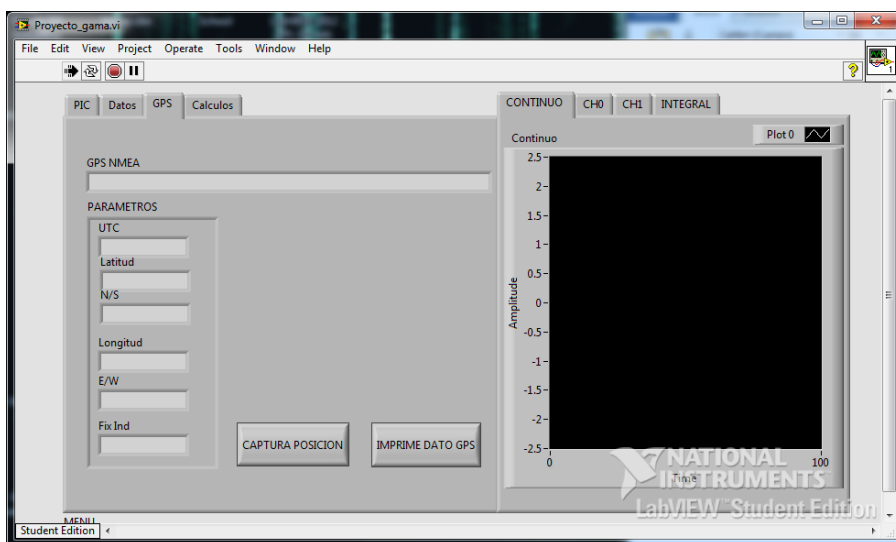
Guarda muestras: Guarda los tres ensayos en archivos de texto planos. Debe existir un valor en el # de muestras para garantizar el correcto almacenamiento de los datos.

### Indicadores

#Muestras: Indica cuantas muestras en total han sido capturadas para cada ensayo.

M1/M2/M3: Nos permite identificar los valores de las muestras capturadas en cada ensayo

## 5.2.3 Controles del GPS



### Botones

Captura posición: Envía la instrucción al sistema de adquisición para que consulte y capture el valor GGA del módulo GPS y lo almacene en memoria para su posterior uso. No es enviado a la interfaz de usuario.

Imprime dato GPS: Solicita el valor almacenado del valor GGA almacenado en el sistema de adquisición y lo muestra en la interfaz de usuario. Nota: El valor mostrado puede no coincidir con el valor teórico actual.

### Indicadores

GPS NMEA: Muestra la cadena GGA obtenida por el Sistema de adquisición sin ningún filtro, en caso que se desee hacer una revisión manual de la misma.

UTC: Muestra el tiempo obtenido del sistema.

Latitud: Muestra la latitud obtenida del sistema.

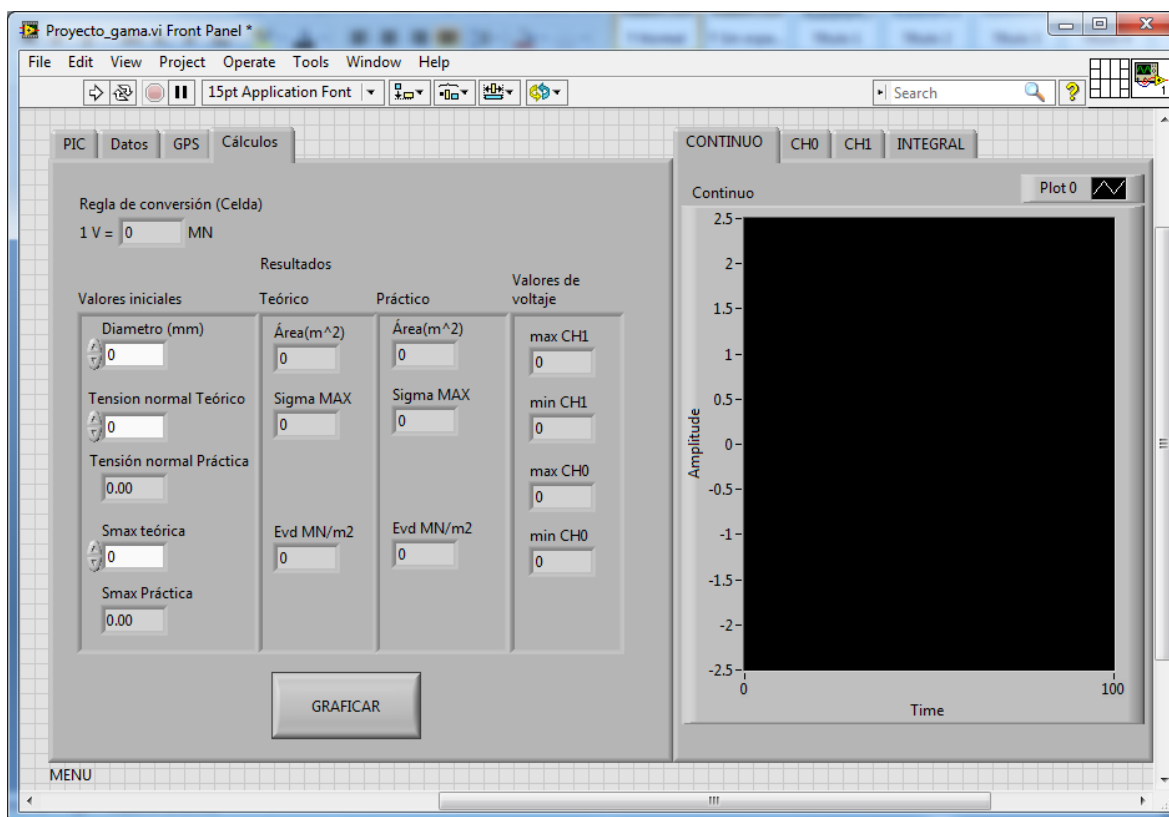
N/S: Muestra el indicador Norte o Sur obtenido por el sistema.

Longitud: Muestra la longitud obtenida del sistema.

E/W: Muestra el indicador Este u Oeste obtenido por el sistema.

FixInd: Muestra si se ha utilizado el arreglo de posición para obtener las coordenadas del módulo GPS. El estándar puede consultarse en la documentación de SIRF.

## 5.2.4 Controles de los cálculos



### Botones

**GRAFICAR:** Realiza el trazado de las gráficas correspondientes a las tres muestras del ensayo.

### Indicadores

**Regla de conversión:** Establece la relación entre el voltaje máximo de la celda de carga y su valor en MN

**Diámetro:** Control para introducir el diámetro del plato de presión

**Tensión Normal:** Tensión normal obtenida del ensayo o introducida manualmente

**Smax:** Valor de la distancia promedio obtenida de la señal resultante del geófono al momento de aplicarle el operador integral.

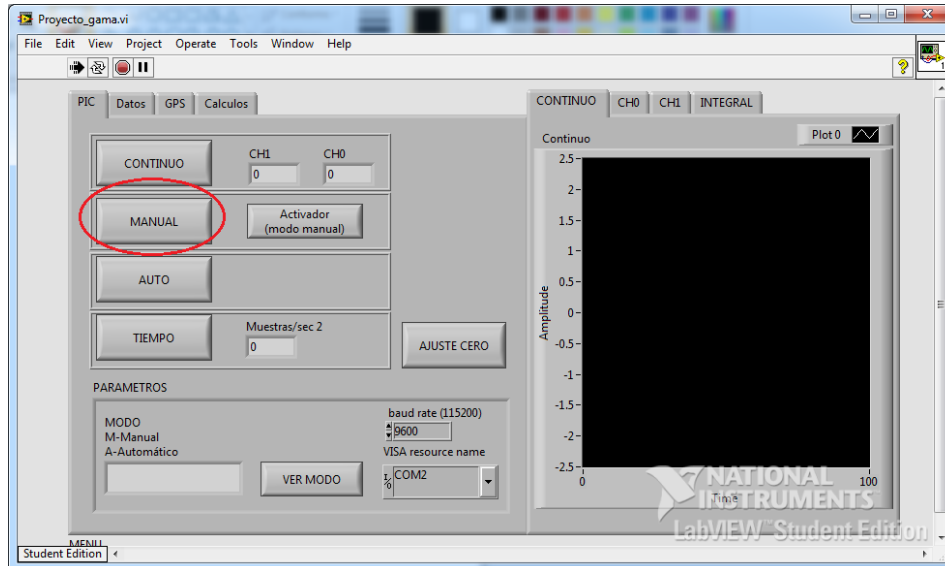
**Área:** Indicador que muestra el cálculo del área del plato de presión.

**Sigma max:** Valor de la sigma máxima calculada como la fuerza obtenida por la celda dividida entre el área del plato de presión.

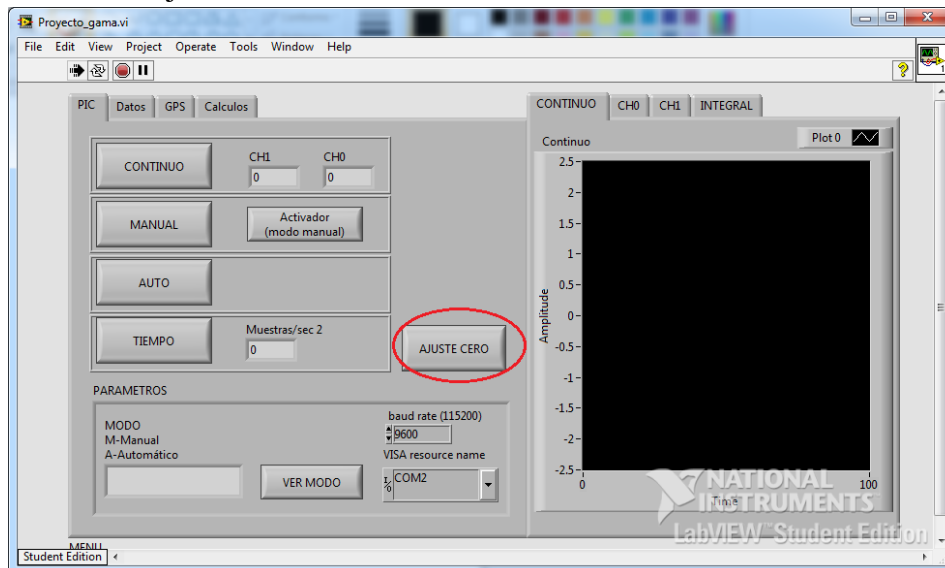
**Evd MN/m<sup>2</sup>:** Valor resultante del modulo resilente de la subrasante.

## 5.2.5 Muestreo de forma manual

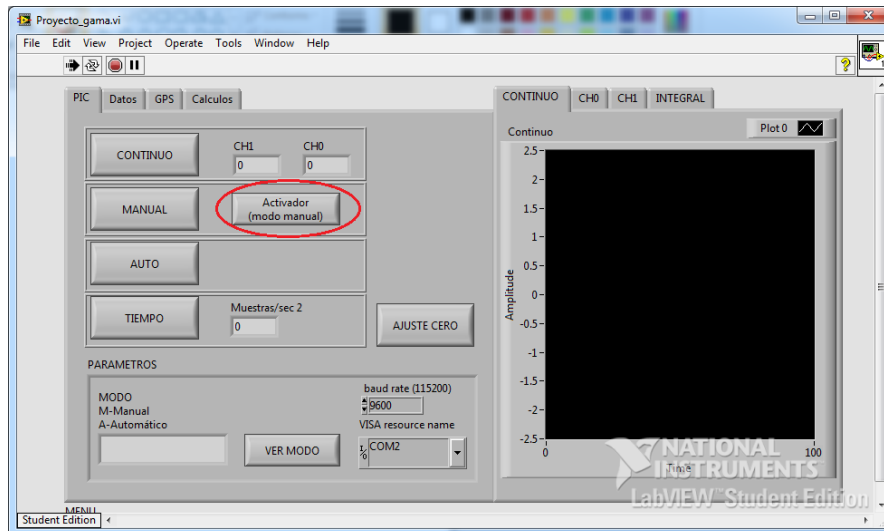
1) En la pestaña del PIC, establecer modo de trabajo



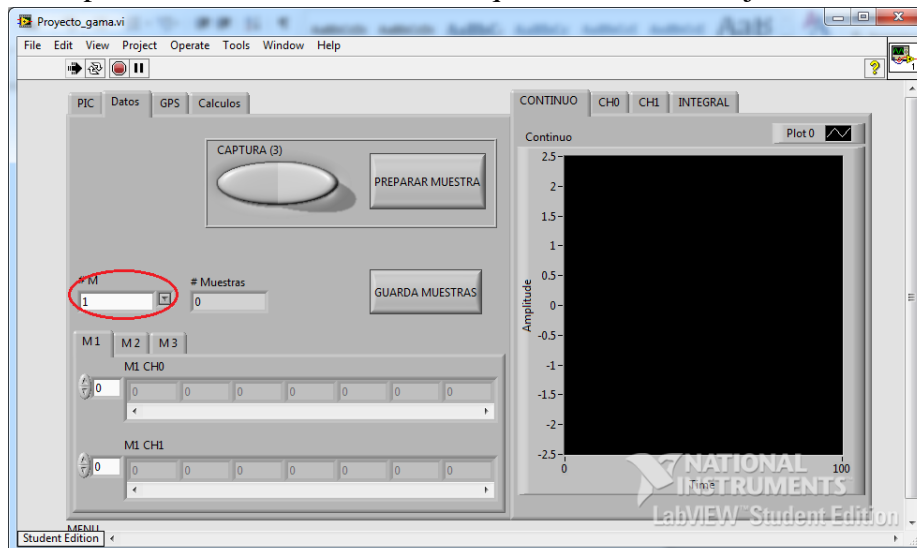
2) Realizar el ajuste de cero



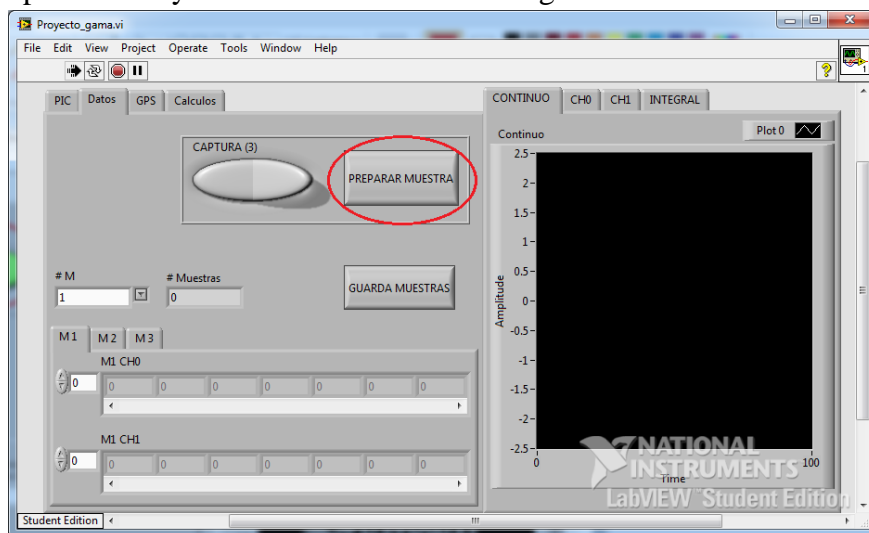
3) Presionar el activador manual



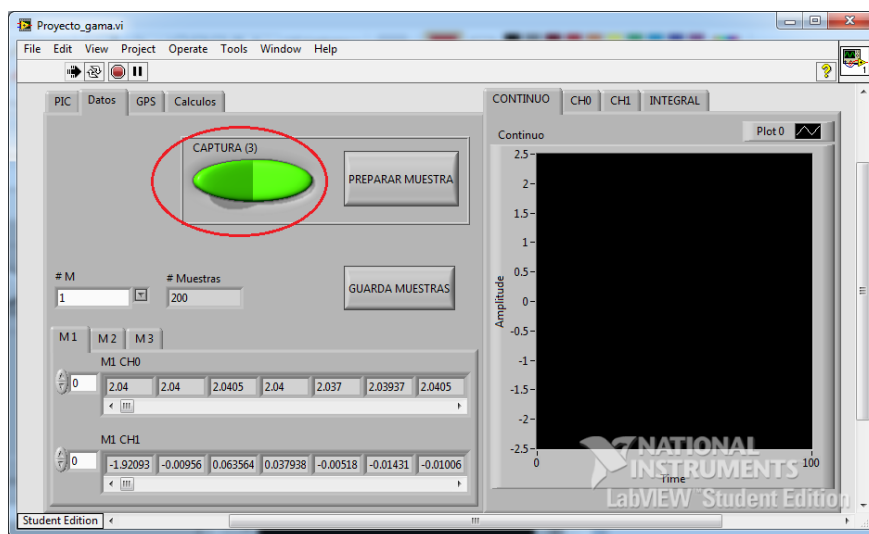
4) En la pestaña Datos, seleccionar con qué muestra se trabajará



- 5) Hacer la preparación de la muestra a utilizar para asegurar que los valores de los apuntadores y la inicialización de los arreglos sean los correctos.

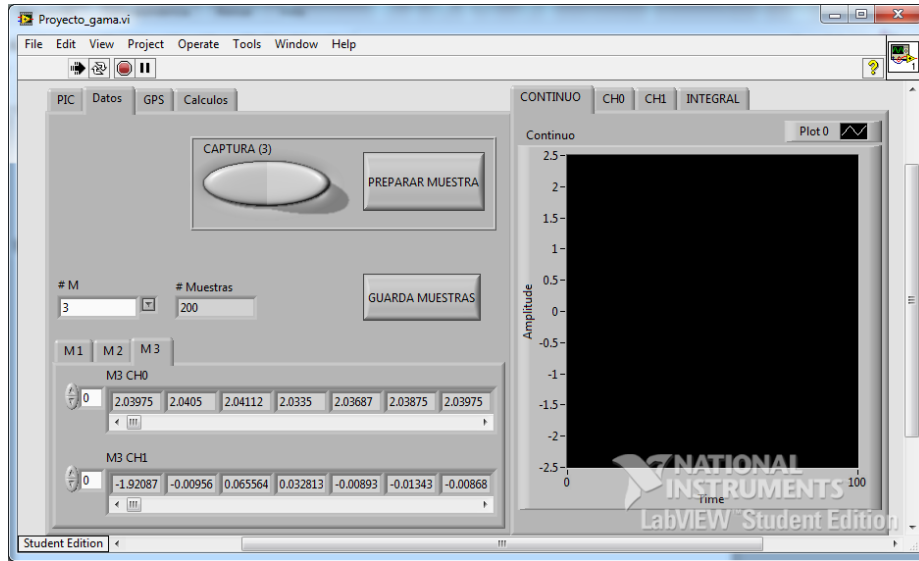


- 6) Realizar el proceso de captura de muestras. Si el ensayo fue exitoso, comenzarán a aparecer los valores de las muestras en el arreglo seleccionado, así como incrementará el contador de muestras. Nota: Es importante dejar trabajar al sistema mientras está haciendo la adquisición de datos, para evitar posibles fallas en el sistema.

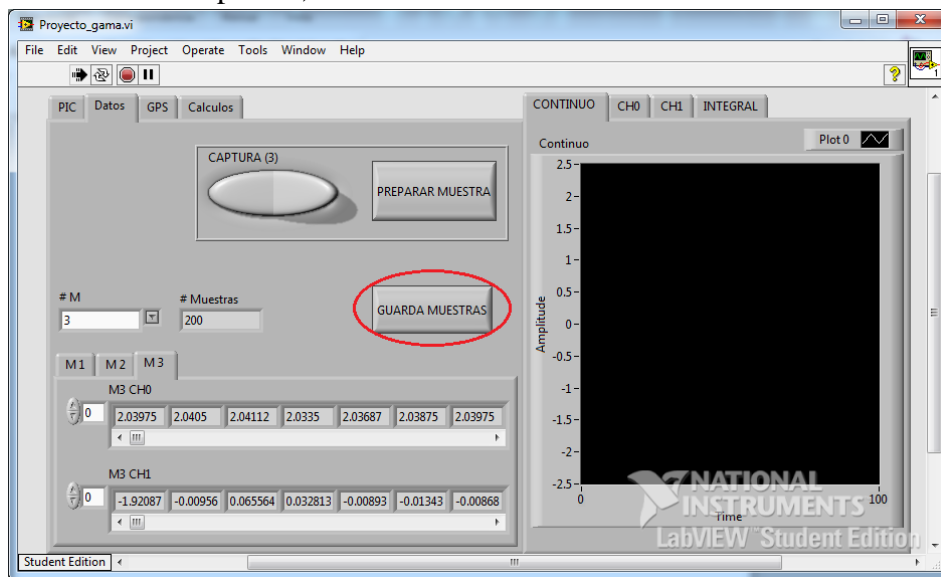


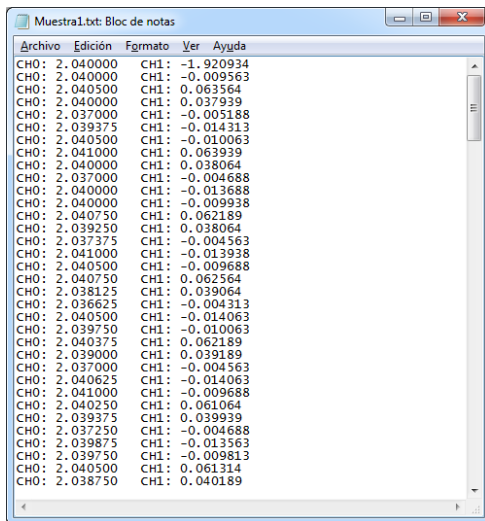
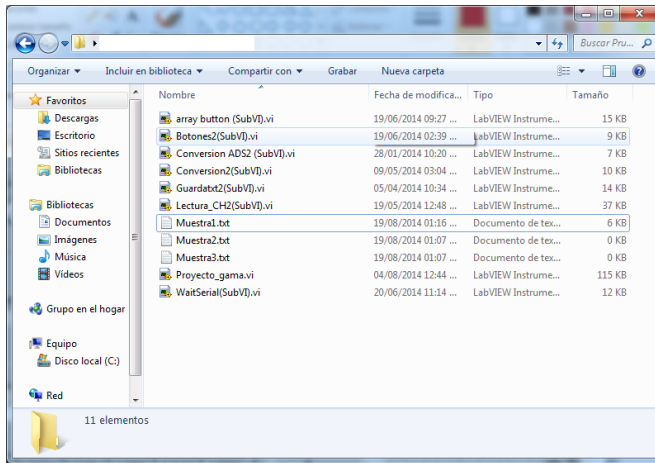


- 7) Repetir los pasos anteriores para los tres ensayos a realizar. Nota importante: En caso de querer almacenar los valores de las muestras en un archivo txt, no se deberán reconfigurar los apuntadores de las muestras ni los arreglos, por lo que se quedarán con los valores de la última muestra analizada (para el caso del ejemplo, la muestra 3)

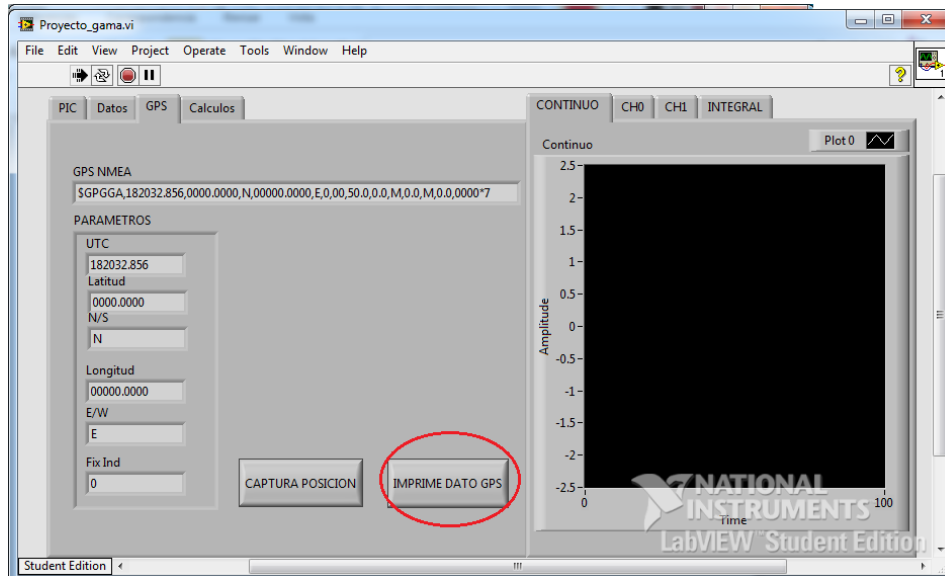


- 8) (Opcional) Guardar las muestras en un archivo txt. Las muestras serán almacenadas en tres archivos planos, ubicados en el directorio actual.

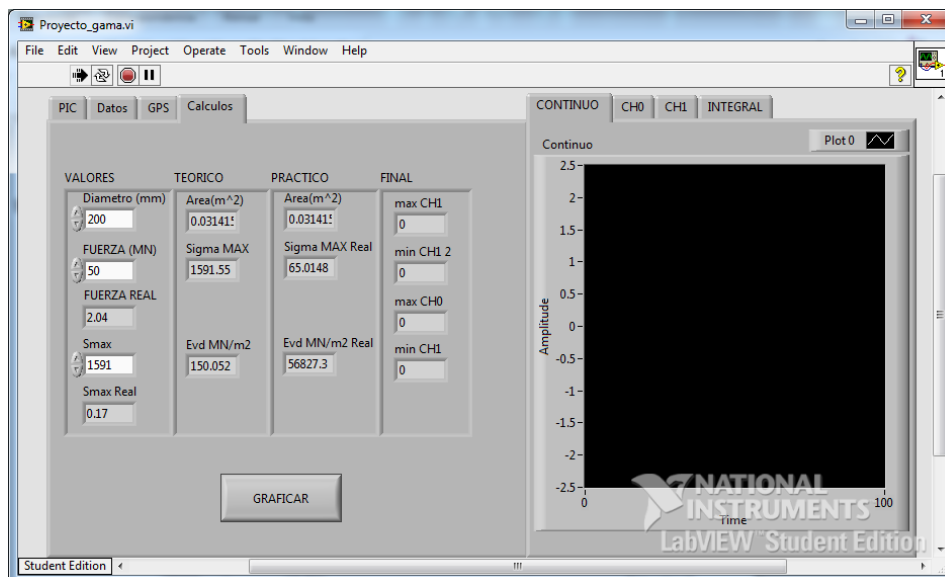




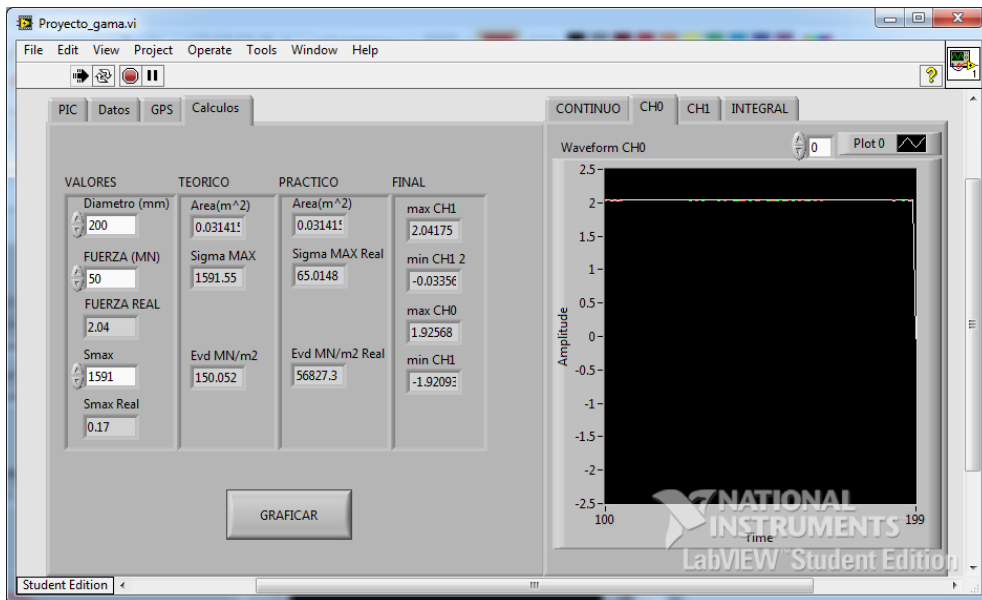
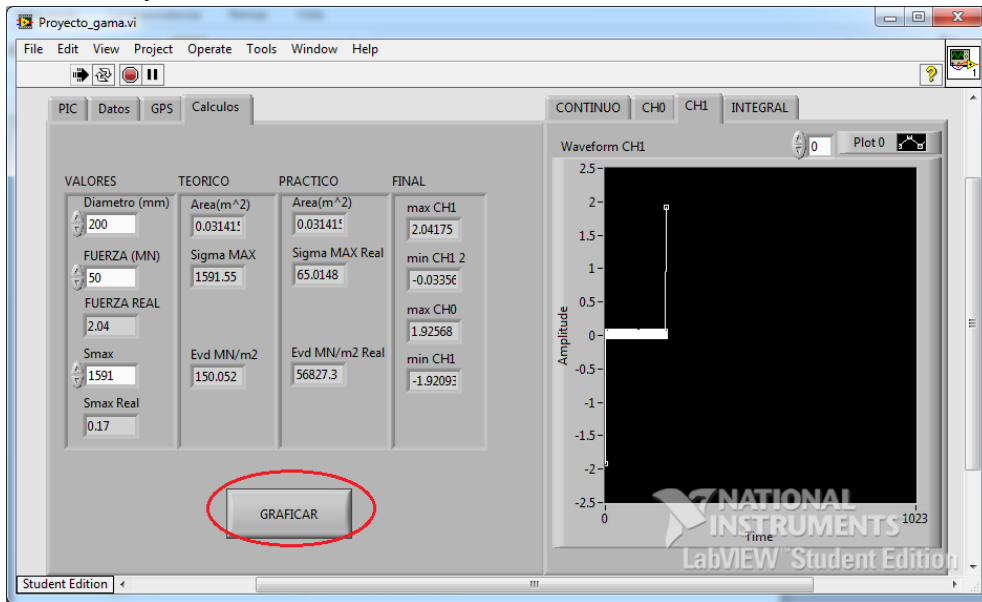
- 9) Obtener la información del módulo GPS. Cuando se realiza un ensayo exitoso, se realiza una consulta automática del parámetro GGA al módulo GPS y se almacena en el sistema. Es por ello que no es necesario hacer una petición de consulta, y se puede proceder solo a imprimir el valor almacenado.



- 10) En la pestaña cálculos se procede a realizar todas las operaciones necesarias para el estudio de los datos.

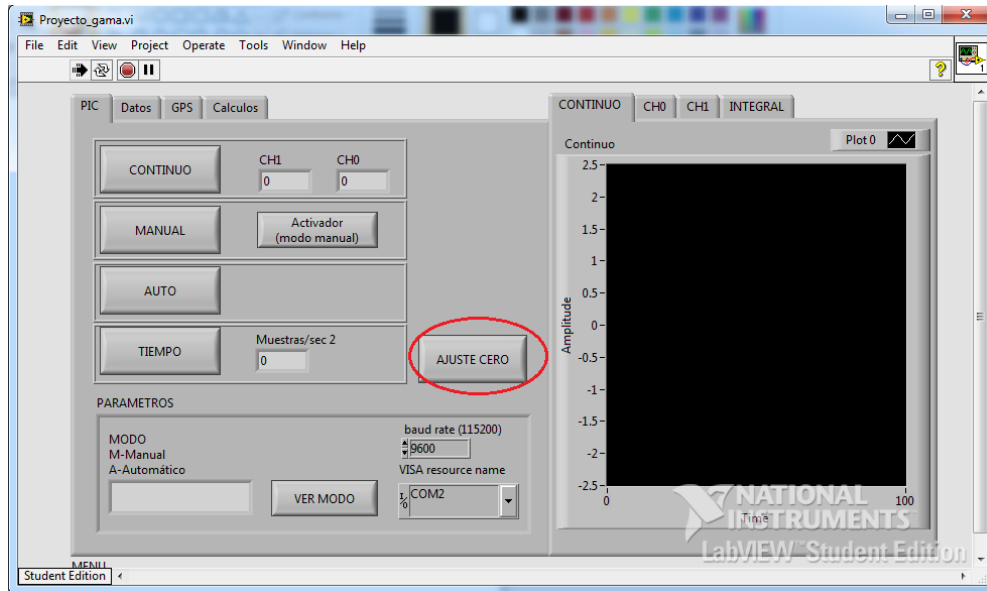


11) Finalmente se realiza la graficación de las muestras. Las gráficas aparecerán en las pestañas CH0 y CH1.

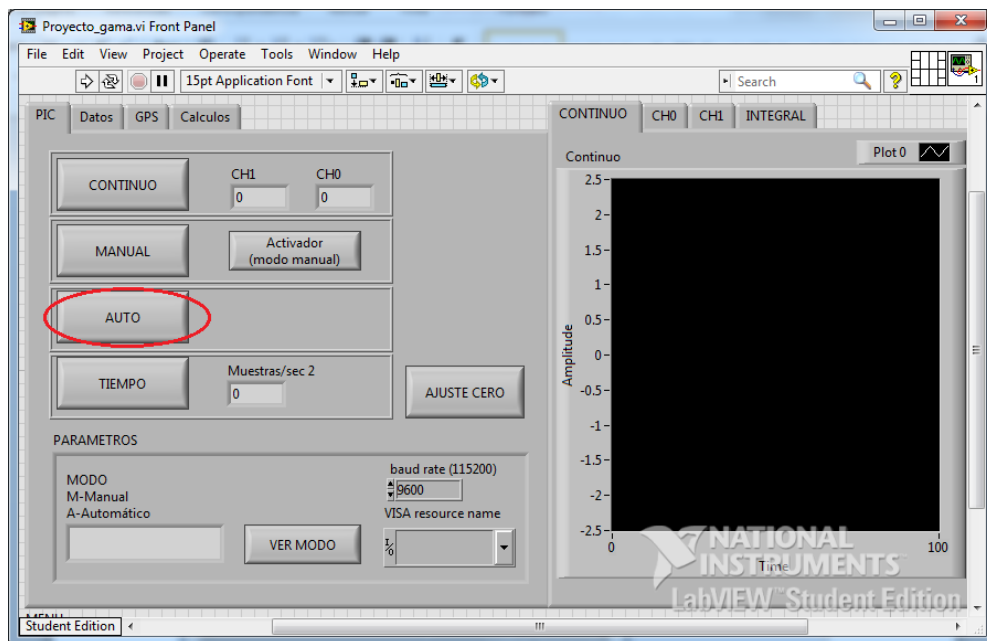


## 5.2.6 Muestreo de forma automática

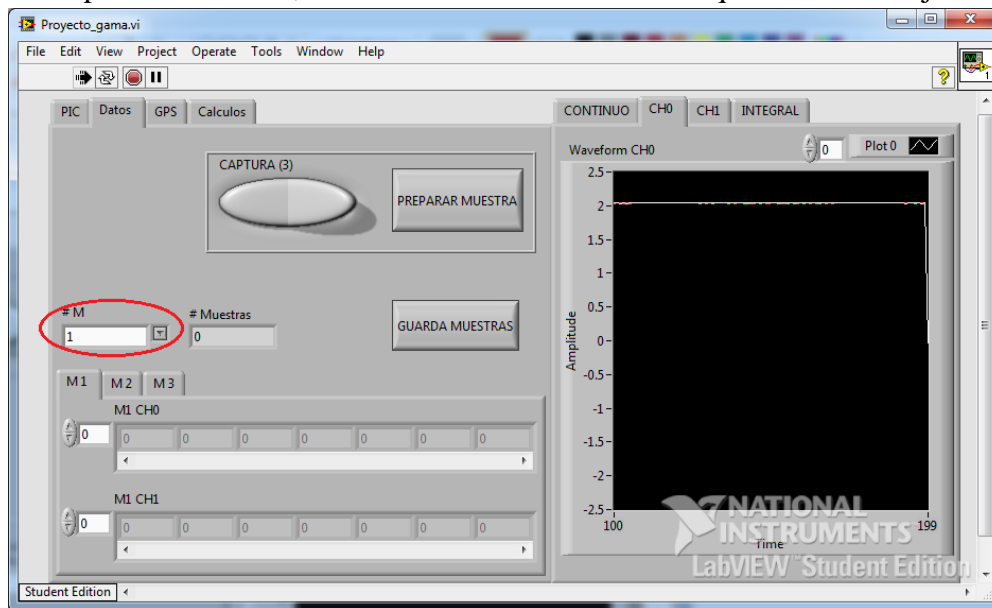
- 1) Se realiza el ajuste de cero.



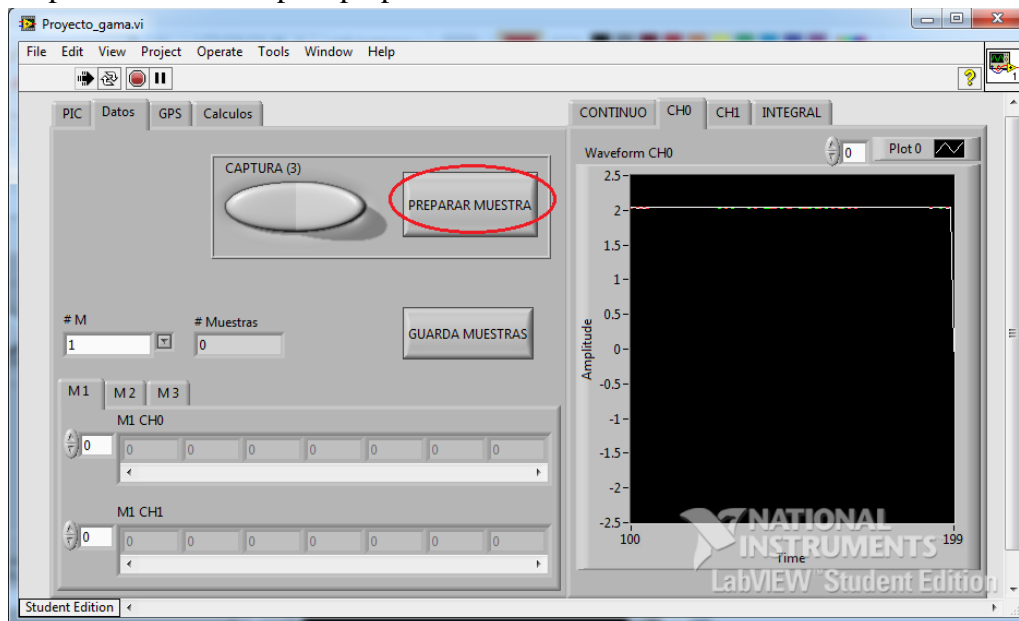
- 2) En la pestaña del PIC, se escoge el modo automático. Una vez activo, en cuanto el sistema detecte un alto voltaje, se iniciará el proceso de captura sin necesidad de un accionador.



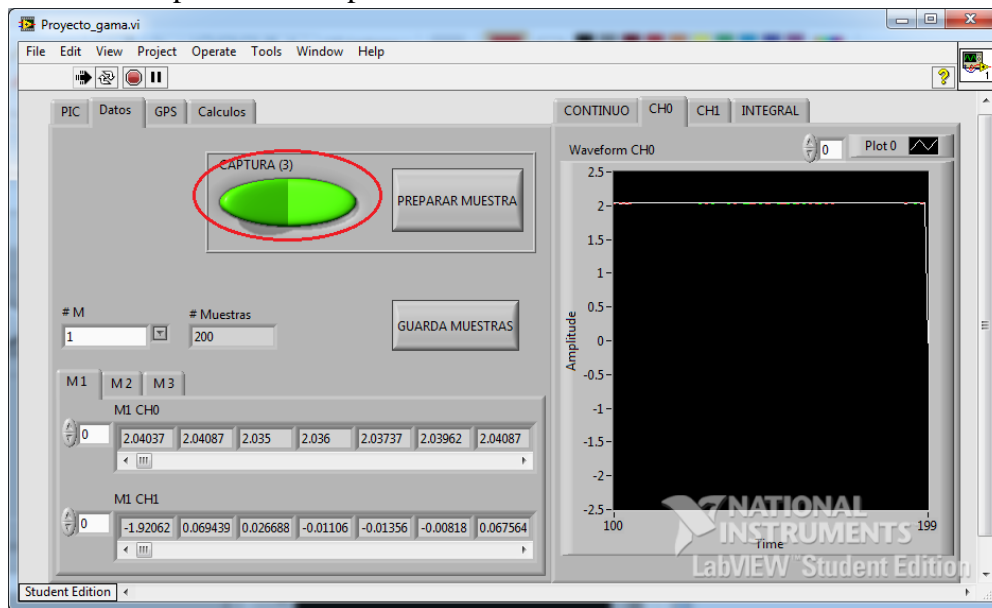
3) En la pestaña de datos, se selecciona la muestra con la que se va a trabajar.



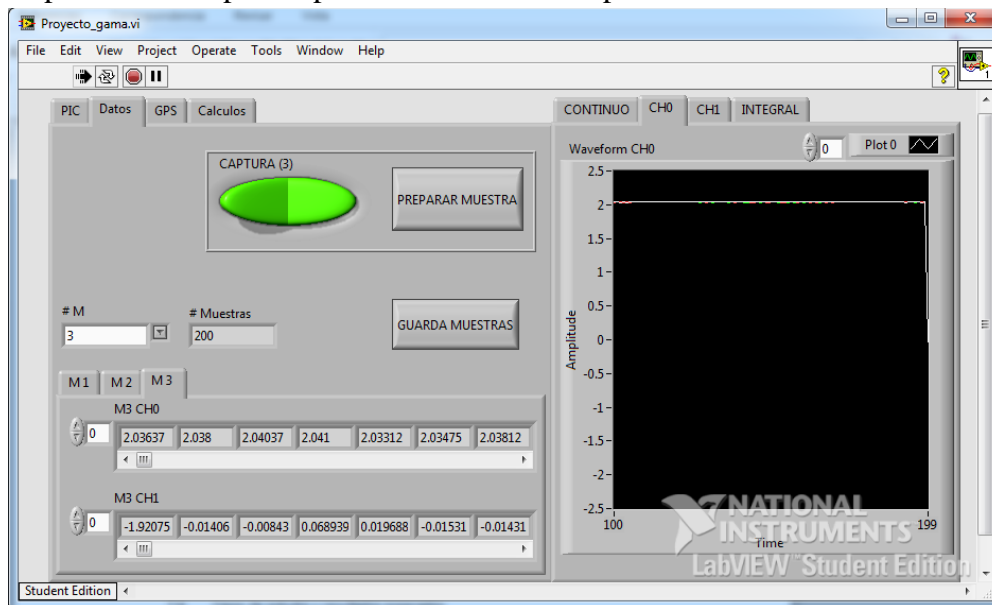
4) Se presiona el botón para preparar la muestra seleccionada.



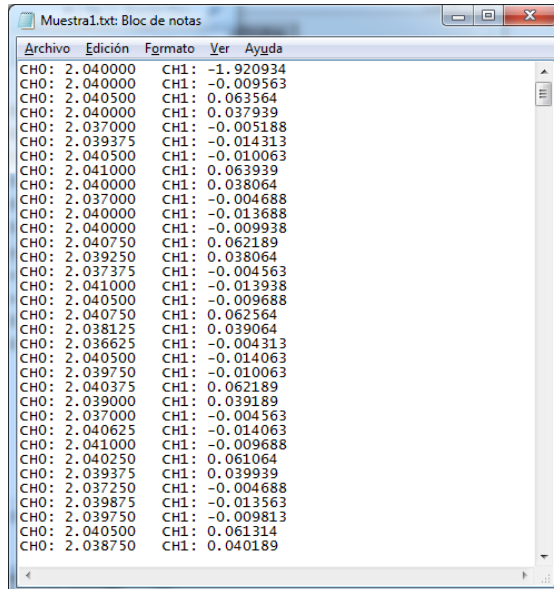
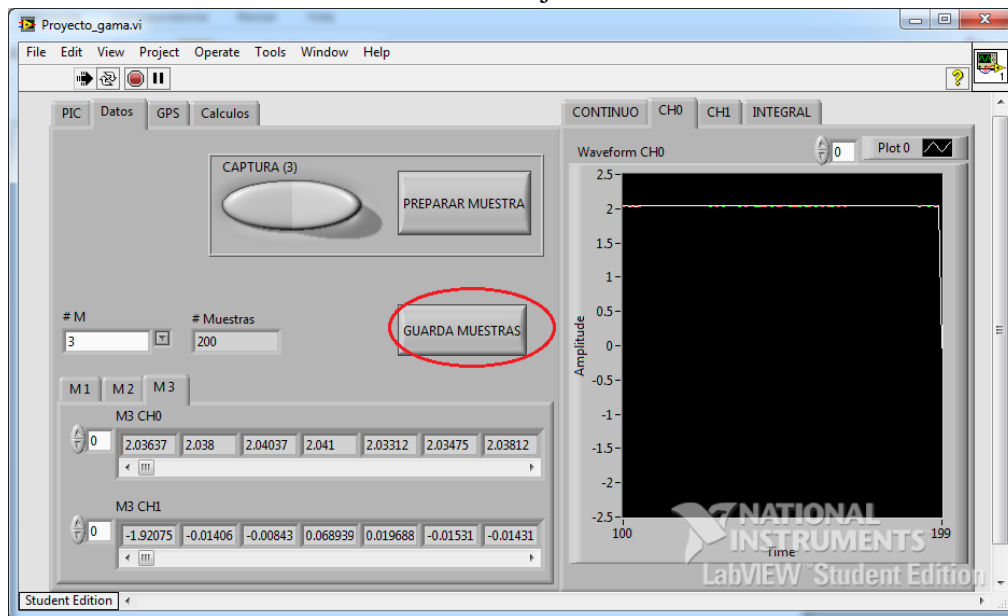
5) Se realiza el proceso de captura de la muestra seleccionada.



6) Repetir desde el paso 3 para cada muestra a capturar

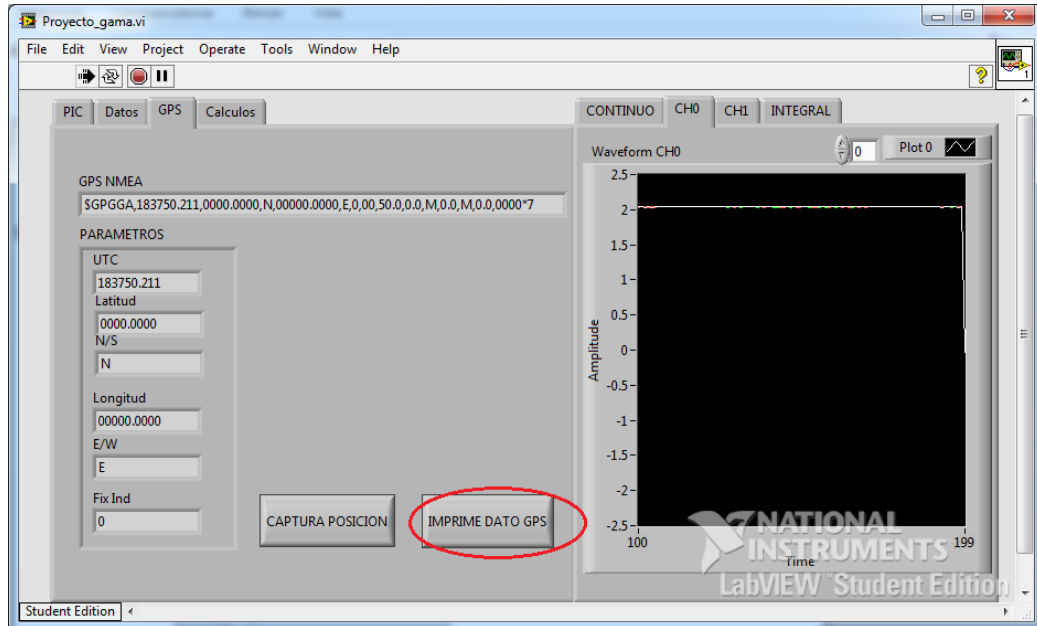


- 7) (Opcional) Guardar las muestras en archivos txt. Recordar que la ubicación de los archivos es en la dirección actual de trabajo.

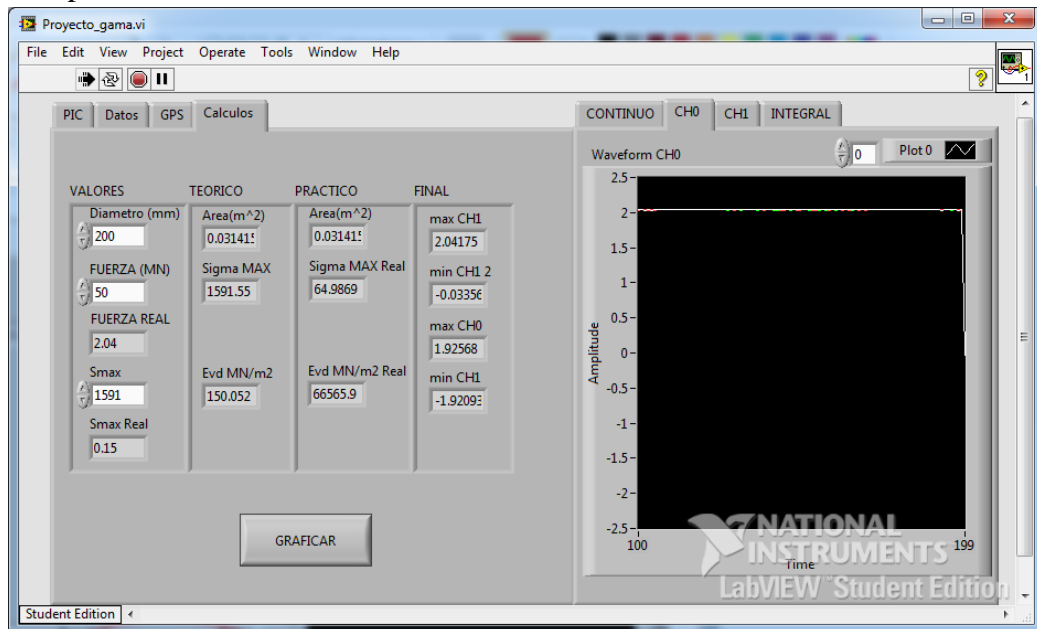




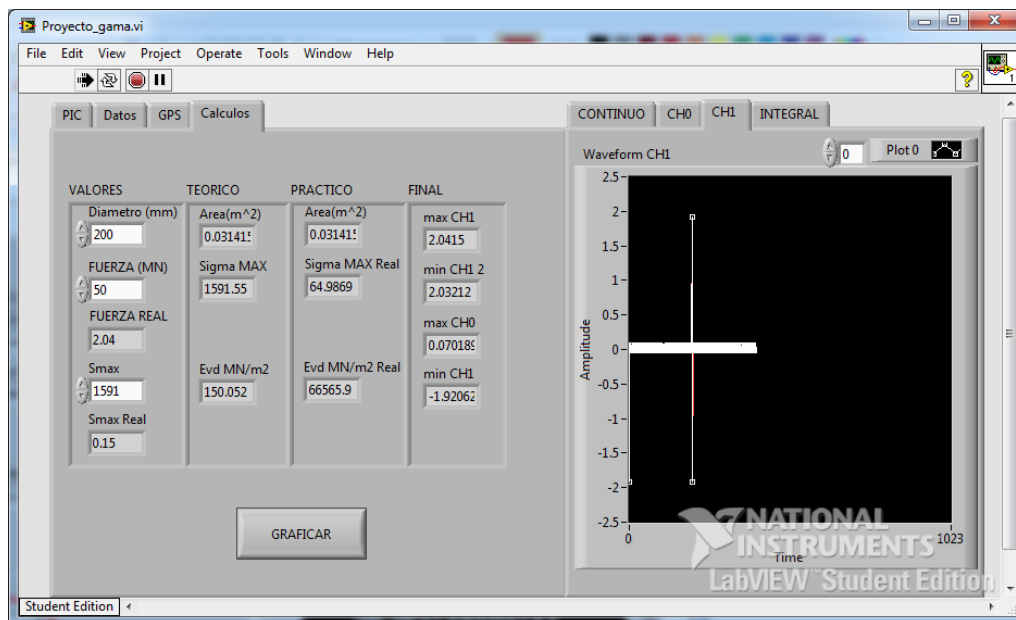
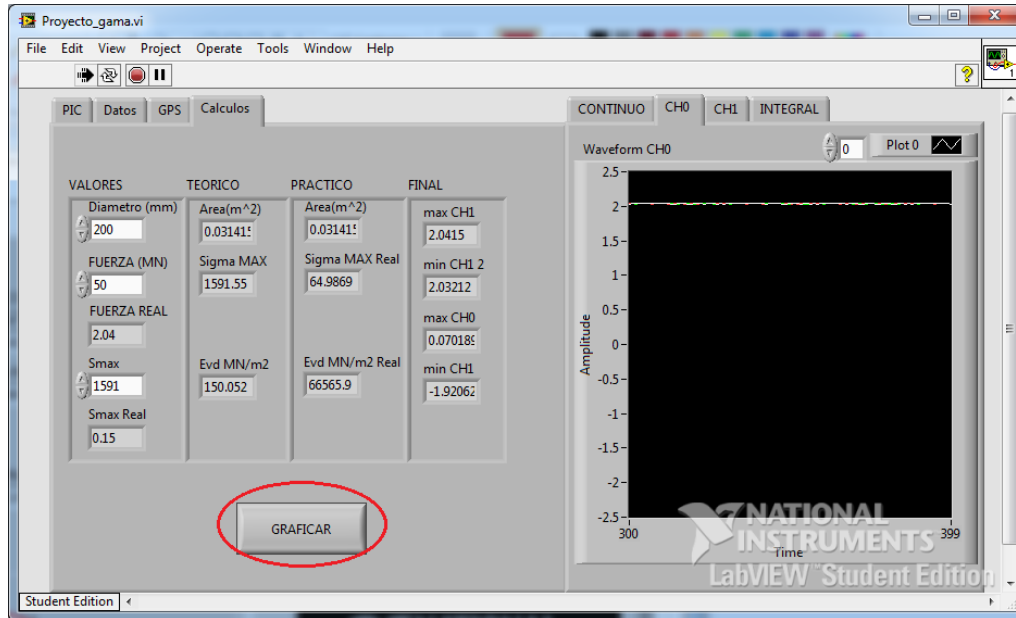
- 8) En la pestaña GPS, Obtener la cadena GPS asociada al último ensayo. Recordar que no es necesario realizar una nueva captura, ya que el sistema la realiza de forma automática.



- 9) En la pestaña de cálculos, realizar todas las operaciones necesarias para la interpretación de los resultados.

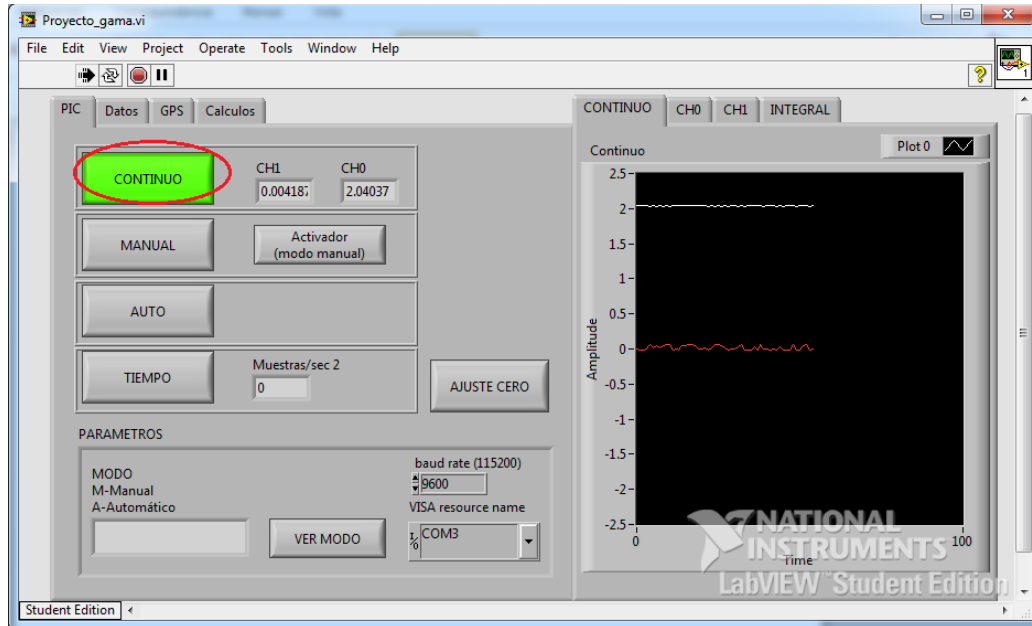


10) Finalmente, para observar las gráficas características, oprimir el botón indicado.

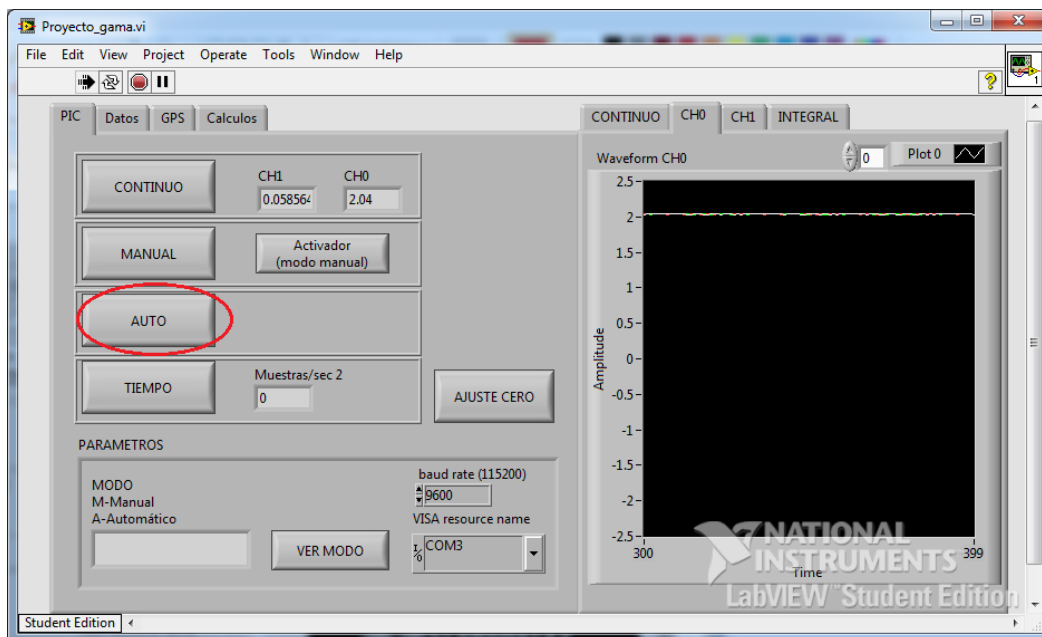


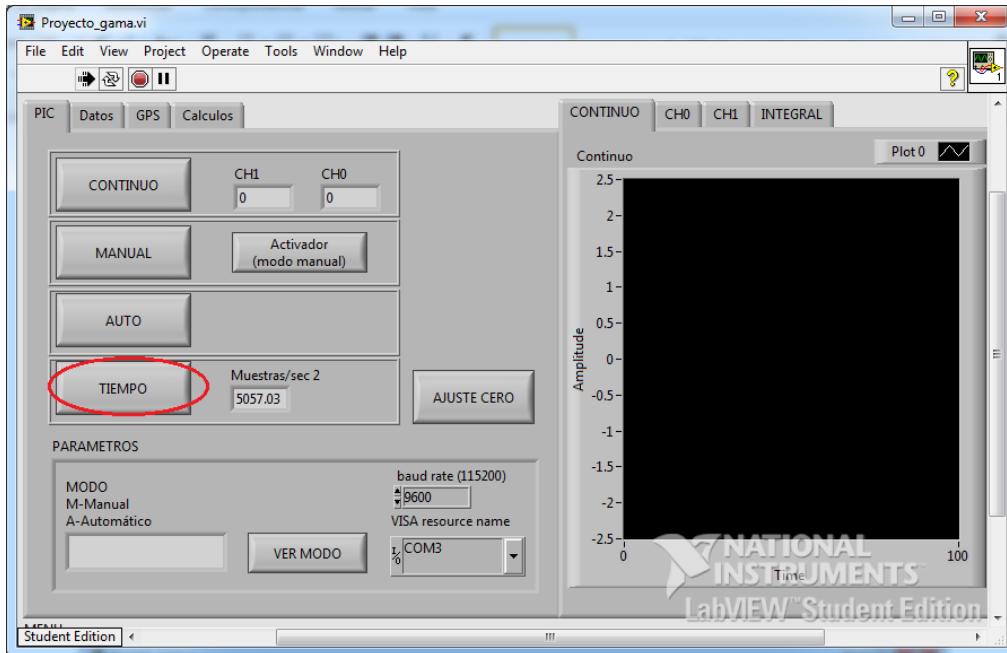
## 5.2.7 Funciones adicionales

**Muestreo continuo:** Para realizar un muestreo en tiempo real, basta con oprimir el botón “Continuo”. La grafica se podrá analizar en la pestaña “Continuo”. El ultimo valor capturado se podrá apreciar en los indicadores CH0 y CH1.

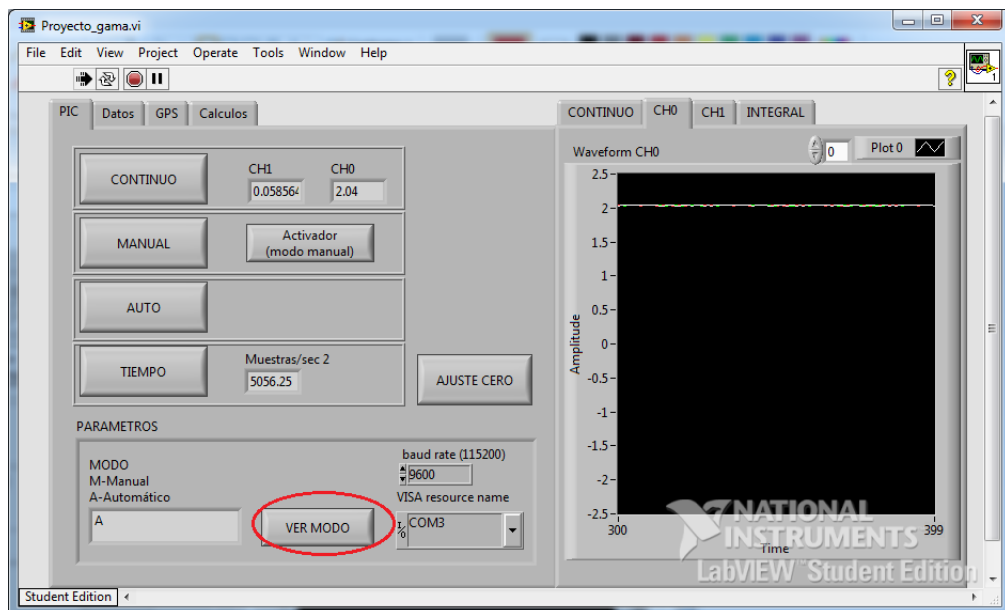


**Número de muestras por segundo:** Para conocer los tiempos de respuesta, basta con colocar el sistema en modo automático unos segundos y después oprimir el botón de tiempo.

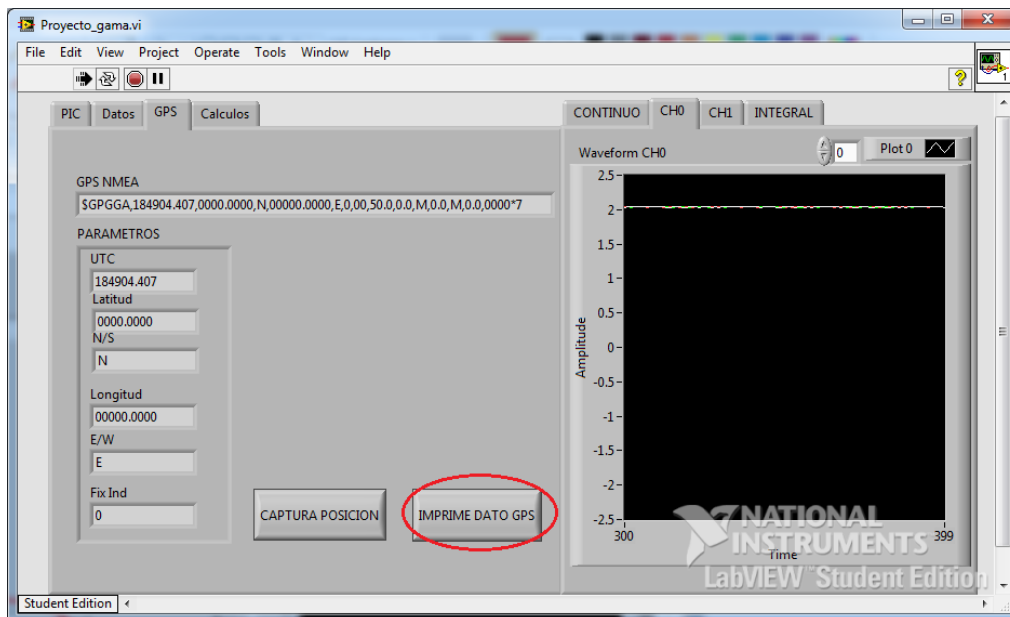
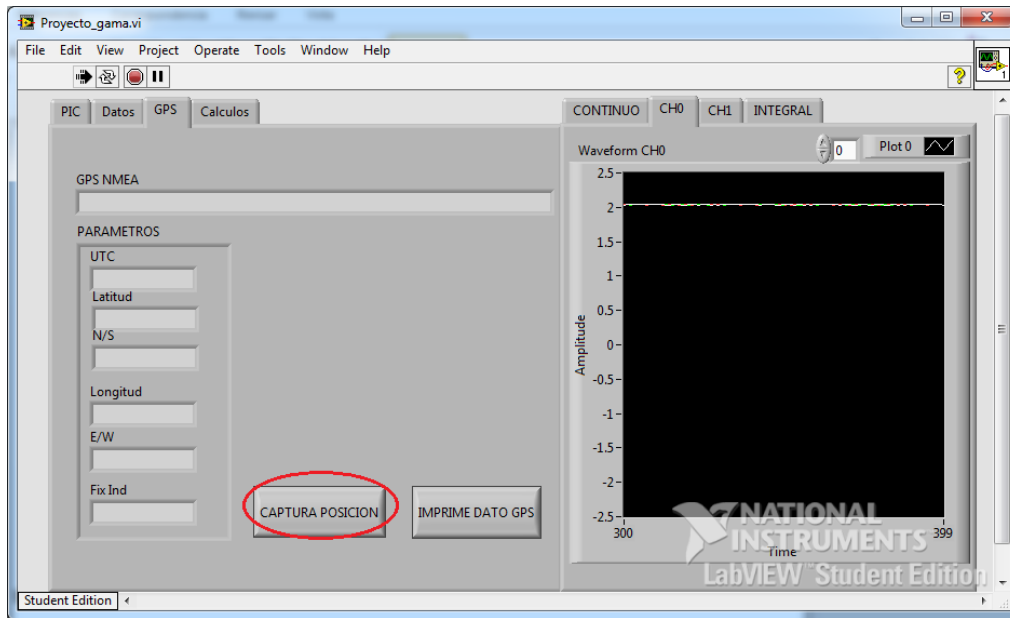




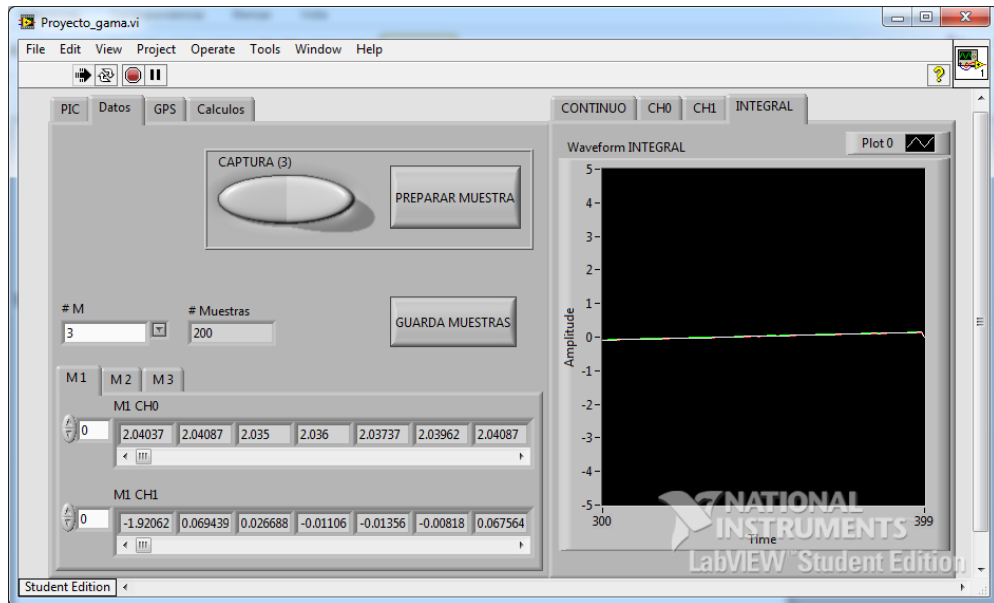
**Modo:** Al oprimir este botón, se podrá observar en que modo esta funcionando el sistema



**Obtener posición actual del GPS:** Para borrar la cadena GGA almacenada y obtener la posición actual, primero se realiza una petición de captura de posición, y luego se manda a imprimir.



**Integral:** La gráfica de la integral de la señal de la celda de carga aparecerá de forma automática en el proceso de graficación tanto del modo manual como del modo automático.



## Aplicación

Para demostrar la funcionalidad del sistema de adquisición en general, se han realizado varios tipos de pruebas en un tipo de suelo terroso con alto nivel de compactación. Es aquí donde se utilizarán diferentes tipos de anillos de goma seleccionados para comprobar que los niveles de voltaje producidos por la excitación de los transductores varían. Estos anillos serán escogidos en base a una serie de ensayos que determinarán que tipo de arreglo es el más adecuado para generar la curva característica para calcular el módulo dinámico de deformación del terreno



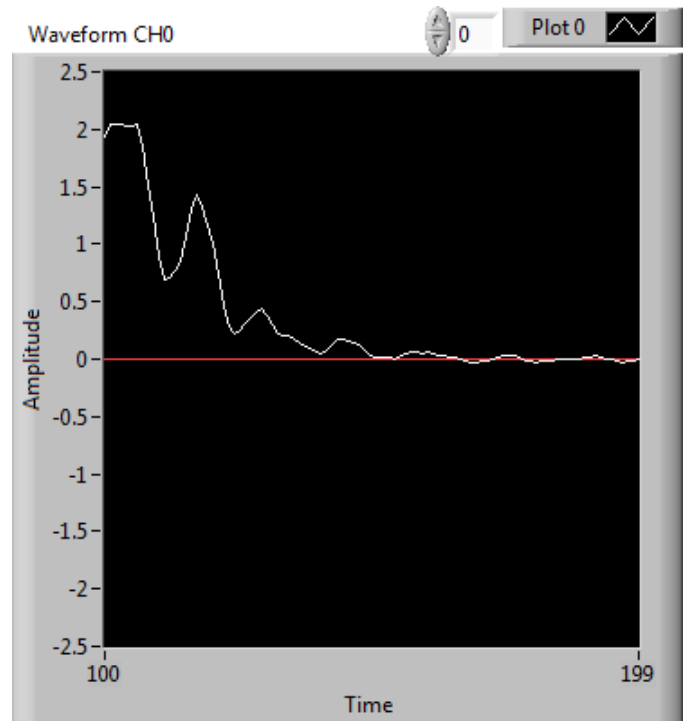
Para el uso del deflectómetro de impacto, se utilizan una serie de arreglos de anillos de goma, las cuales tienen la función de evitar abolladuras en el modelo físico. Dependiendo del número de anillos de goma y de la forma de la misma, se obtendrá una variación en la excitación de la celda de carga.



Cada muestra será obtenida a través de 3 ensayos independientes, dando un impacto con la maza en la placa de presión. En ella se encuentra la celda de carga la cual registrará la fuerza de impacto que se está ejerciendo sobre la placa depresión. El nivel de excitación de la celda dependerá de los anillos que se utilizarán para la prueba. A continuación se muestran las curvas características obtenidas por varios arreglos de anillos.

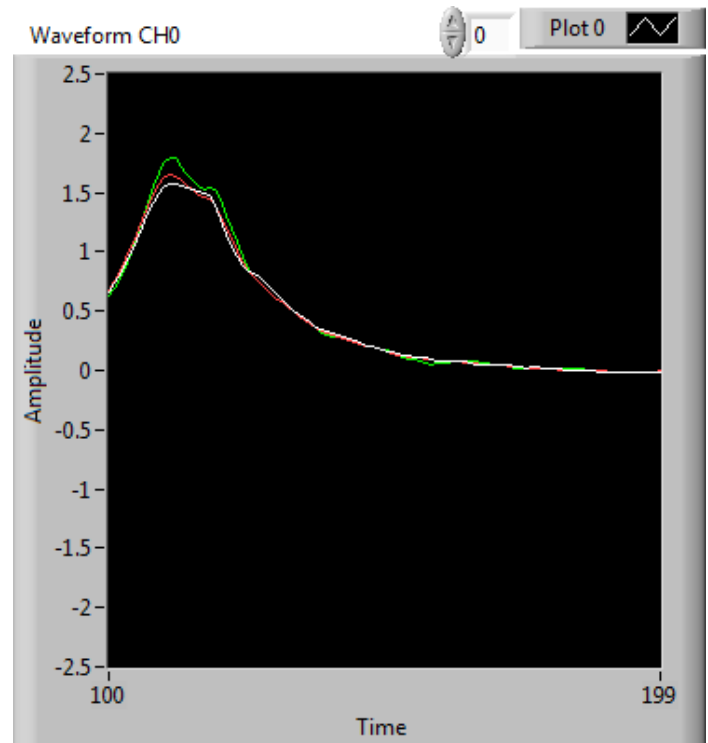


## Primera prueba



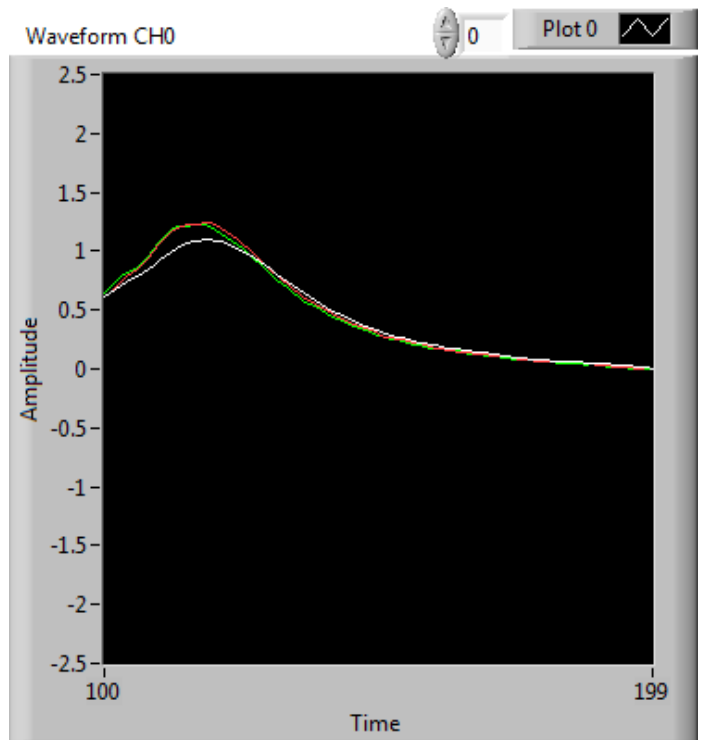
En esta prueba se realizó un impacto con una goma roja. A la salida de la celda de carga se puede observar una curva característica con varias oscilaciones. Recordando que los amplificadores operacionales son saturados a un máximo de 4.096V (2.048V lógicos) se puede concluir que la goma no absorbe una cantidad de energía considerable del impacto, por lo que la salida de la celda de carga presenta una saturación. Las oscilaciones son presentadas debido a que el impacto genera una serie de rebotes en la masa, así como en la placa de presión. De este ensayo puede concluirse que no presenta las características teóricas de la curva característica necesaria para calcular el módulo dinámico de deformación del terreno.

## Segunda prueba



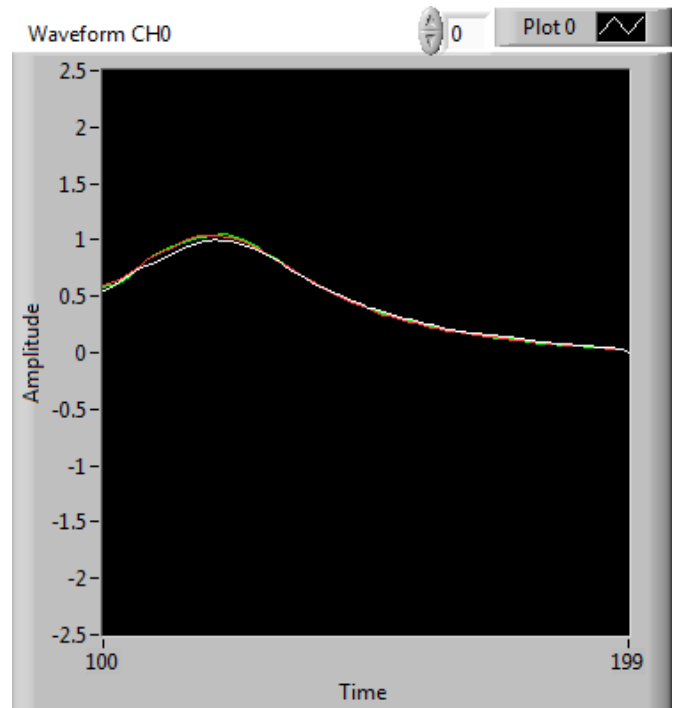
Esta prueba ha sido realizada con dos gomas negras. Si bien para este ensayo el rebote de la masa aumento sensiblemente, este movimiento no afecto a la curva principal del impacto, y con este arreglo se pudo disminuir el movimiento en la placa de presion. Los niveles de voltaje se hallan dentro de los valores aceptables, por lo que con este arreglo puede calcularse el módulo dinámico de deformación del terreno. De cualquier manera, se puede analizar que se puede refinar aún más la curva, por lo que este arreglo de anillos no será utilizado.

### Tercera prueba



La siguiente prueba ha sido realizada con una goma negra y una roja. Al analizar la gráfica, se puede observar que los niveles de voltaje han disminuido a un nivel intermedio dentro del rango permitido. Se ha reducido prácticamente todo el ruido de rebote tanto de la masa como de la placa de presión, por lo que la curva característica de este ensayo resulta óptima para calcular el módulo dinámico de deformación del terreno.

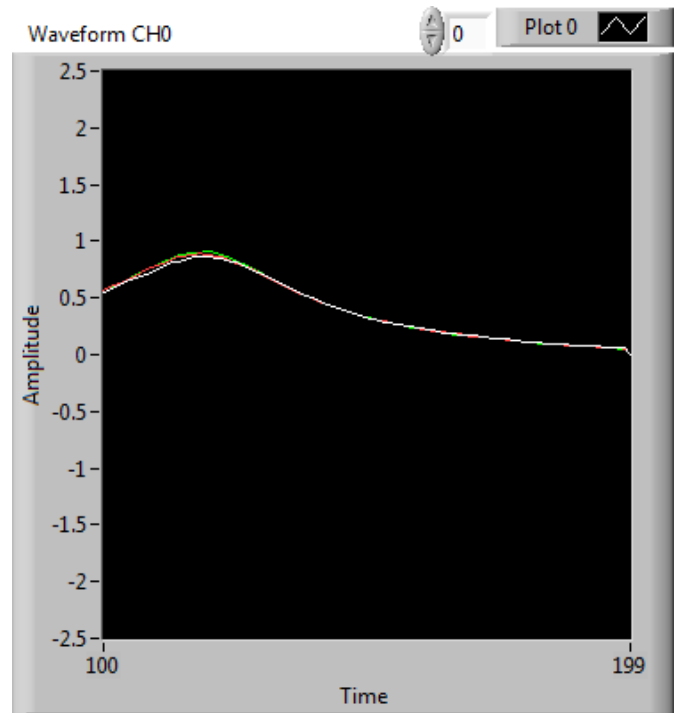
## Cuarta prueba



Para esta prueba se utilizaron en el siguiente orden: una goma roja, una goma negra y una goma roja. Analizando la gráfica se puede concluir que este arreglo puede ser utilizado para calcular el módulo dinámico de deformación del terreno por las características citadas en las pruebas pasadas. Lamentablemente los rangos de voltaje están por debajo del valor medio del rango de valores permitidos, por lo que al repetir el ensayo en cualquier otro tipo de suelo, los resultados podrían verse afectados, llegando incluso a no lograr alcanzar el voltaje de activación del sistema.

Es por esta razón que este arreglo ha sido descartado.

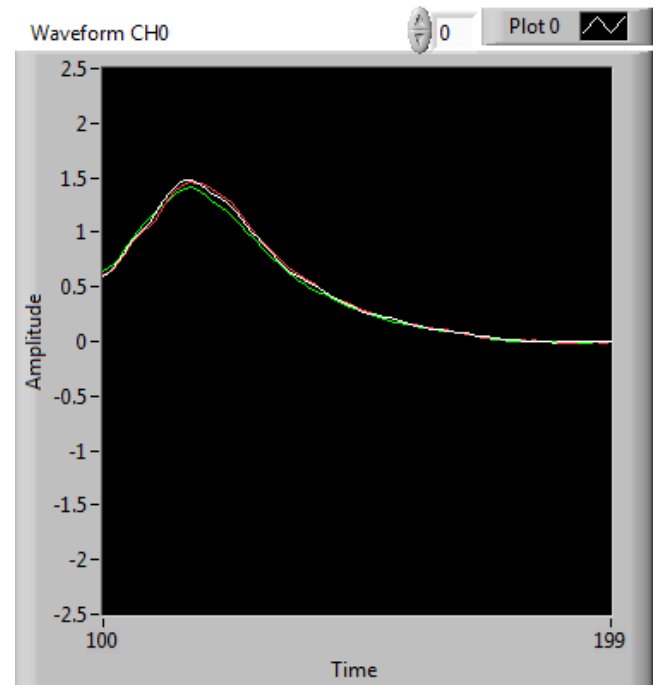
## Quinta prueba



Para esta prueba se usaron dos gomas negras. Si bien el ruido por rebote de la masa y de la placa de presión ha sido suprimido, los niveles de voltaje son aun mas bajos que cualquier otro ensayo realizado. Es por ello que aunque puede obtenerse el módulo dinámico de deformación del terreno para esta prueba en particular, el arreglo queda descartado para otros ensayos en diferentes tipos de suelo.



## Sexta prueba



La siguiente prueba fue realizada con dos gomas rojas. En esta prueba los ensayos mostraron una curva característica con un menor número de oscilaciones y con un nivel de voltaje no saturado. El rebote en la placa de presión también ha sido reducido considerablemente al igual que el rebote en la masa. La prueba ha sido concluyente para considerar este modelo apto para ensayos posteriores en cualquier otro tipo de suelo.

## Conclusiones

Recordando los objetivos principales del presente trabajo, el desarrollo del sistema de adquisición y de la interfaz de usuario ha sido completado en su totalidad, permitiendo con esto estar un paso más adelante en la liberación de la totalidad del proyecto. El sistema de adquisición ha sido exigido en pruebas diversas con diferentes tipos y modelos de transductores, logrando los resultados esperados del mismo. Esto nos indica que la presente tarjeta de adquisición puede ser usada para la captura de diferentes tipos de señales e inclusive el firmware puede ser reutilizado para pruebas posteriores de otros sistemas de adquisición. La organización de la metodología de trabajo en el firmware ha permitido crear una serie de librerías, las cuales fueron diseñadas para poder ser utilizadas por cualquier PIC de 8 o 16 bits. Cada una de dichas librerías controla un bloque específico del sistema, por lo que su implementación puede ser total o parcial.

En cuanto a la interfaz del usuario, se ha logrado generar una interfaz de control amigable y visualmente estética, la cual simplifica en gran medida la operación del sistema, y permite realizar los cálculos y funciones de forma intuitiva y amigable. En las pruebas realizadas se pudo comprobar su eficiencia y tolerancia al fallo, por lo que resulta un diseño de software confiable. De esta forma se comprueba que el presente trabajo ha podido cumplir de forma cabal los objetivos planteados, y aunado a esto el desarrollo del código generado (incluido en el apéndice del presente documento) podrá ser reutilizado para futuros proyectos.

## Bibliografía/ Referencias

- ASTM international, Designation E2835 – 11, “Standard Test Method for Measuring Deflections using a Portable Impulse Plate Load Test Device”
- Pontificia Universidad Católica de Chile, Escuela de Ingeniería 2008, “Metodología de evaluación in-situ del modulo resiliente de capas granulares de pavimentos flexibles con el deflectómetro de impacto liviano”, Alelí Osorio Lird, Guillermo Thenoux, Felipe Halles.
- Boletín de la Asociación de Laboratorios Acreditados de la Comunidad de Madrid, Diciembre 2006, “Ensayo de placa de carga dinámica de 300 mm de diámetro”, Carlos Fernández Tadeo
- Texas Instruments, Enero 1996, SBOS062A INA 126, “Micropower instrumentation amplifier” Data Sheet
- Texas Instruments, Abril 2005, SLAS429B DAC8551, “16-bit, ultra-low glitch, voltage output digital-to-analog converter” Data Sheet
- Texas Instruments, Septiembre 2000, SBAS136D ADS8341, “16-Bit, 4-Channel Serial Output Sampling analog-to-digital converter” Data Sheet
- Texas Instruments, 23A1024/23LC1024, “1Mbit SPI Serial SRAM with SDI and SQI Interface” Data Sheet
- Texas Instruments, Diciembre 2011, SLVSA92A TPS63060, “High input voltage buck-boost converter with 2A switches current” Data Sheet
- Texas Instruments, LF00 SERIES, “Very low drop voltage regulator with inhibit” Data Sheet
- Leadtek, LR9552, “Leadtek SMD GPS module” Data Sheet
- SiRF Technology Inc, December 2007, NMEA Reference Manual
- Wavesen, HC-06 Serial Bluetooth Products, User Instructional Manual
- Microchip, DS39626E, “PIC18F2525/2620/4525/4620” Data Sheet



- Custom Computer Services, May 2014, “CCS C Compiler Manual PCB / PCM / PCH”

## Apéndice

### I Código del Firmware

#### Main.c

Nota: Los acentos y caracteres especiales son omitidos en algunos casos para evitar malfuncionamiento del compilador

```
/**
//directivas
//*****
#include <18F2620.h>
#fuses HS,NOWDT,PROTECT,CPB,CPD,NOPUT,NOBROWNOUT

#use delay(clock=4000000)
#use rs232(baud=9600, xmit=PIN_C6, rcv=PIN_C7, ERRORS, RESTART_WDT)

#include <1_parametros.h>
#include <2_dac.h>
#include <3_ads.h>
#include <4_memoria.h>
#include <6_GPS.h>
#include <7_bluetooth.h>

void int_RS232();

//VARIABLES DEL MUESTREO
int m_auto=0, golpe=0, m_lista=0;
long muestra=0x0000, total=0x0096; //total=0x0096;

//Variable protocolo de comunicacion
int repeticion=0, respuesta=1;

long count=0x0000;
```

```

////////////////////////////////////
//INTERRUPCION LLEGADA DE DATO SERIE
////////////////////////////////////
long x1=0x0000, x2=0x0000, tiempo=0x0000;
#int_rtcc
void contador_timer0()
{
  if(m_auto==1)
  {
    tiempo=x1;
    x1=0x0000;
    x2=0x0000;
  }
}

//Contador de muestras adquiridas
void count_x()
{
  if(x1==0xffff)
  {
    if(x2==0xffff)
    {
      x2=0x0000;
    }
    else
    {
      x2=x2+0x0001;
    }
    x1=0x0000;
  }
  else
  {
    x1=x1+0x0001;
  }
}

```

```

void reset_muestras()
{
////////////////////
//Bloque de reset total
////////////////////
m_auto=0;
golpe=0;
m_lista=0;
muestra=0x0000;
count= total+total;
repeticion=0;
respuesta=1;
}

////////////////////
//MAIN
////////////////////
void main()
{
inicia_PIC();
count= total+total;

output_low(PIN_A1);
delay_ms(500);
output_high(PIN_A1);
delay_ms(500);

enable_interrupts(INT_RDA);
enable_interrupts(global);

```

```

////////////////////////////////////
//CICLO PRINCIPAL
////////////////////////////////////
while(true)
{

if(m_lista == 0)
{
//MODO MANUAL
if(m_auto==0)
{
///ACTIVACION A PETICION
if(golpe==1)
{
muestra=0x0000;
while(muestra < (total+total) )
{
captura_ads_void();
muestra = muestra + 0x0001;
}
muestra=0x0000;
golpe=0;
m_lista=1;
}
} //if auto
//MODO AUTOMATICO
if(m_auto==1)
{
///ACTIVACION CUANDO CH0 CH1 ALCANCEN EL LIMITE
if( captura_ads(0x9fff, 0xafff) )
//if(m_auto==1)
{
while( muestra < total)
{
captura_ads_void();
muestra = muestra + 0x0001;
}
muestra=0x0000;
m_lista=1;
//printf("auto:%i\n",m_lista);

```

```

        }
    } //else auto
} // if m_lista
count_x();
delay_us(10); //5022 m#/s
} //while

} //main

////////////////////////////////////
//INTERRUPCION LLEGADA DE DATO SERIE
////////////////////////////////////
#INT_RDA
void int_RS232()
{
char data;
data = getch();
if(data == '0')
{
reset_muestras();
if(m_auto == 0)
printf("M\n");
else
printf("A\n");
}
if(data == '1') //AJUSTE CERO
{
reset_muestras();
calcula_cero(0x7fff);
//printf("7fff\n"); ejemplo salida
}
if( data== '2') //muestreo EN TIEMPO REAL
{
printf("%lx %lx\n", envia_ads8341(0x96), envia_ads8341(0xD6)); // ch0,ch1
//printf("ffff 0000\n"); ejemplo de salida
}

```



```

}

////////////////////////////////////
//COMANDOS GPS
////////////////////////////////////
if( data=='4')    //ENVIO DE DATOS DEL GPS
{
    //leeGPGGA();
    printf("%s\n",cad_gps);
    //printf("191718.878,1919.6410,N,09910.8603,W,0,00,50.0,2286.6,M,-
9.0,M,0.0,0000*77\n"); Ejemplo de salida
}
if( data=='5')    //ENVIO DEL GPS A PETICION
{
    leeGPGGA();
}

////////////////////////////////////
//MODOS DE CAPTURA
////////////////////////////////////
if( data=='6')    //MODO MANUAL
{
    reset_muestras();
    m_auto=0;
}
if( data=='7')    //MODO AUTOMATICO
{
    reset_muestras();
    m_auto=1;
}
if( data=='8')    //GOLPE MANUAL
{
    respuesta=1;
    repeticion=0;

    if(m_auto==0)
        golpe=1;
}
if( data=='s' || data=='S')

```



```

    {
    respuesta=1;
    repeticion=0;
    }
if( data=='n' || data=='N')
    {
    respuesta=1;
    repeticion=1;
    }
if( data=='d' || data=='D')
    {
    respuesta=1;
    repeticion=0;
    }

////////////////////////////////////
//CONTEO MUESTRAS
////////////////////////////////////
if( data=='9') //CONTEO MUESTRAS
    {
    printf("%lx\n",tiempo);
    }

}

```

## 1\_parametros.h

```
//NOMBRE DE PINES
#define mcu_led    PIN_C0    //led verde
#define RAM_CS     PIN_B0
#define AD_CS      PIN_B1
#define AD_BSY     PIN_B2
#define AD_SHD     PIN_B3
#define DAC8551_SYNC PIN_B4
#define U18_RESET  PIN_A1
#define MUX        PIN_A2

void inicia_PIC()
{
    output_B(0x00);
    SET_TRIS_B(0x04);    //pin_b2= input, else=output
    SET_TRIS_A(0x01);    //pin_b2= input, else=output
    output_high(DAC8551_SYNC); //pin_b3 encendido
    output_high(AD_CS);    //pin_b1 encendido
    output_high(AD_SHD);    //pin_b2 encendido
    output_high(RAM_CS);    //pin_b0 encendido
    output_low(MUX);
    output_high(U18_RESET);    //pin_a0 encendido
    setup_adc_ports(NO_ANALOGS); //evita conversion analoogica

//INICIALIZACION DE LA RAM
setup_spi(SPI_MASTER | SPI_CLK_DIV_16 | SPI_L_TO_H | SPI_XMIT_L_TO_H |
SPI_SAMPLE_AT_END); //CKP=0; CKE=1
    output_low(RAM_CS); //SYNC=0 para habilitar registro
        spi_write(0x01);    // Manda modo write
        //spi_write(0x00);    // Manda modo Byte
        spi_write(0x40);    // Manda modo Secuencial
    output_high(RAM_CS); //RAM_CS=1 para deshabilitar registro
//setup_uart(9600);
}
```

## 2\_dac.h

```
////////////////////////////////////
//MANEJO DAC
////////////////////////////////////

#define DAC8551    PIN_B4

void envia_dac8551_fs(int mode, long data_l)
{
    output_high(DAC8551); //SYNC=1 para deshabilitar registro
    setup_spi(SPI_MASTER | SPI_H_TO_L | SPI_CLK_DIV_16); //Inicializar variables

    mode = (mode&0x03); //Solo m0 y m1 habilitados
    output_low(DAC8551); //SYNC=0 para habilitar registro
    delay_us(5);
    spi_write(mode); // Manda modo
    spi_write(*((int8 *)&data_l + 1)); // manda dato
    spi_write(*((int8 *)&data_l + 0)); // manda dato
    output_high(DAC8551); //SYNC=1 para deshabilitar registro
}
```

## 3\_ads.h

```
#define ADS_CS      PIN_B1
#define ADS_BSY     PIN_B2
#define ADS_SHD     PIN_B3

////////////////////////////////////
//MANEJO ADS
////////////////////////////////////

long envia_ads8341(int mode)
{
    unsigned int data_1,data_2,h;
    long dato;

    setup_spi(SPI_MASTER | SPI_CLK_DIV_16 | SPI_L_TO_H | SPI_XMIT_L_TO_H |
    SPI_SAMPLE_AT_END); //CKP=0; CKE=1

    output_low(ADS_CS); //AD_CS = 0 para habilitar reg I/O del AD
```

```

spi_write(mode);          //Manda DCLK y dato hacia Din del AD

    delay_us(10);        //busy maximo 8 us
    data_1 = spi_read(0); //Manda DCLK y recibe dato del Dout del AD
    data_2 = spi_read(0); //Manda DCLK y recibe dato del Dout del AD
    h = spi_read(0);      //Manda DCLK y recibe ceros del Dout del AD

dato=make16(data_1,data_2)<<1; //junta 2 int en un long, hace corrimiento
//dato=make16(data_1,data_2); //junta 2 int en un long, hace corrimiento

output_high(ADS_CS);      //AD_CS = 1 para deshabilitar reg I/O del AD
return dato;
}
////////////////////////////////////
//AJUSTE DE CERO
////////////////////////////////////
void calcula_cero(long cero)
{
long aprox=0x0000, dato;
long Bytes[16]={0x8000, 0x4000, 0x2000, 0x1000,
                0x0800, 0x0400, 0x0200, 0x0100,
                0x0080, 0x0040, 0x0020, 0x0010,
                0x0008, 0x0004, 0x0002, 0x0001};
int i;
for(i=0; i<16; i++)
{
    dato = envia_ads8341(0x96);
    if( dato < cero)
    {
        aprox = aprox - Bytes[i];
    }
    if( dato > cero)
    {
        aprox = aprox + Bytes[i];
    }
}
envia_dac8551_fs(0, aprox);
delay_ms(10);
//printf(" paso %i  dac:@%lx@  ads: #%lx# \n",i,aprox,dato);
}
}

```

## 4\_memoria.h

```
#include <string.h>
#define RAM      PIN_B0

int dir_H=0x00,dir_M=0x00,dir_L= 0x00;

////////////////////////////////////
//APUNTADO MEMORIA ++
////////////////////////////////////
void nuevo_dato_23A(void)
{
//APUNTADOR DE LAS DIRECCIONES DE LA RAM. dir_H,dir_M,dir_H es el ultimo
valor
//dir_H=0x00,dir_M=0x00,dir_L= 0x00;
if(dir_L != 0xff)
    dir_L++;
else
{
dir_L=0x00;
if(dir_M != 0xff)
{
    dir_M++;
}
else
{
    dir_M = 0x00;
if(dir_H != 0xff)
    dir_H++;
else
    dir_H= 0x00;
}
}
}

////////////////////////////////////
//APUNTADO MEMORIA --
////////////////////////////////////
void antiguo_dato_23A(void)
{
if(dir_L != 0x00)
```

```

    dir_L--;
else
{
dir_L=0xff;
if(dir_M != 0x00)
{
    dir_M--;
}
else
{
    dir_M = 0xff;
if(dir_H != 0x00)
    dir_H--;
else
    dir_H= 0xff;
}
}

}

////////////////////////////////////
//ESCRITURA DEL FLUJO DE ADS EN RAM
////////////////////////////////////
boolean captura_ads(double ch0_M, double ch1_M)
{
int up=0x00, down=0x00;
long ch0,ch1;

    ch0 = envia_ads8341(0x96); //ch0  1ms
    ch1 = envia_ads8341(0xD6); //ch1  1ms

setup_spi(SPI_MASTER | SPI_CLK_DIV_16 | SPI_L_TO_H | SPI_XMIT_L_TO_H |
SPI_SAMPLE_AT_END); //CKP=0; CKE=1

output_low(RAM); //SYNC=0 para habilitar registro

spi_write(0x02); // Manda modo write 0x02
spi_write(dir_H); // manda direccion alta
spi_write(dir_M); // manda direccion media
spi_write(dir_L); // manda direccion baja

```

```

//CH0 dividido a 8x2 bytes
up=make8(ch0,1);
down=make8(ch0,0);

//escritura de CH0 en ram
spi_write(up); //up
nuevo_dato_23A(); //cont++
spi_write(down); //down
nuevo_dato_23A(); //cont++

//CH1 dividido a 8x2 bytes
up=make8(ch1,1);
down=make8(ch1,0);

//escritura de CH1 en ram
spi_write(up); //up
nuevo_dato_23A(); //cont++
spi_write(down); //down
nuevo_dato_23A(); //cont++
output_high(RAM); //RAM_CS=1 para deshabilitar registro

if (ch0 >= ch0_M && ch0_M!=0xffff)
    return true;
if (ch1 >= ch1_M && ch1_M!=0xffff)
    return true;
else
    return false;
}

void captura_ads_void()
{
int up=0x00, down=0x00;
long ch0,ch1;

    ch0 = envia_ads8341(0x96); //ch0 1ms
    ch1 = envia_ads8341(0xD6); //ch1 1ms

setup_spi(SPI_MASTER | SPI_CLK_DIV_16 | SPI_L_TO_H | SPI_XMIT_L_TO_H |
SPI_SAMPLE_AT_END); //CKP=0; CKE=1

```

```

output_low(RAM); //SYNC=0 para habilitar registro

spi_write(0x02); // Manda modo write 0x02
spi_write(dir_H); // manda direccion alta
spi_write(dir_M); // manda direccion media
spi_write(dir_L); // manda direccion baja

//CH0 dividido a 8x2 bytes
up=make8(ch0,1);
down=make8(ch0,0);

//escritura de CH0 en ram
spi_write(up); //up
nuevo_dato_23A(); //cont++
spi_write(down); //down
nuevo_dato_23A(); //cont++

//CH1 dividido a 8x2 bytes
up=make8(ch1,1);
down=make8(ch1,0);

//escritura de CH1 en ram
spi_write(up); //up
nuevo_dato_23A(); //cont++
spi_write(down); //down
nuevo_dato_23A(); //cont++
output_high(RAM); //RAM_CS=1 para deshabilitar registro
}

////////////////////
//LECTURA DEL FLUJO DE ADS EN RAM
////////////////////
void lectura_ads()
{
long dato= 0x0000, dato2= 0x0000;
int up=0x00, down=0x00;

setup_spi(SPI_MASTER | SPI_CLK_DIV_16 | SPI_L_TO_H | SPI_XMIT_L_TO_H |
SPI_SAMPLE_AT_END); //CKP=0; CKE=1

output_low(RAM); //SYNC=0 para habilitar registro

```



```

spi_write(0x03); // Manda modo READ
spi_write(dir_H); // manda direccion alta
spi_write(dir_M); // manda direccion media
spi_write(dir_L); // manda direccion baja

//lectura de CH1
antiguo_dato_23A(); //cont--
down = spi_read(0); //down
antiguo_dato_23A(); //cont--
up = spi_read(0); //up
// dato=make16(up,down);
dato=make16(down,up);

//lectura de CH0
antiguo_dato_23A(); //cont--
down = spi_read(0); //down
antiguo_dato_23A(); //cont--
up = spi_read(0); //up
// dato2=make16(up,down);
dato2=make16(down,up);

printf("%lx %lx\n",dato, dato2);
output_high(RAM); //RAM_CS=1 para deshabilitar registro

}

void lectura_ads_contador(long count, int repeticion)
{
long dato= 0x0000, dato2= 0x0000;
int up=0x00, down=0x00;

////////////////////////////////////
///CAMBIO DE APUNTAORES PARA DATO REPETIDO
////////////////////////////////////

if(repeticion != 0)
{
//SIMULA escritura de CH1 en ram
nuevo_dato_23A(); //cont++

```

```

nuevo_dato_23A(); //cont++
//SIMULA escritura de CH1 en ram
nuevo_dato_23A(); //cont++
nuevo_dato_23A(); //cont++
    count = count+0x0001;
}

setup_spi(SPI_MASTER | SPI_CLK_DIV_16 | SPI_L_TO_H | SPI_XMIT_L_TO_H |
SPI_SAMPLE_AT_END); //CKP=0; CKE=1

output_low(RAM); //SYNC=0 para habilitar registro

spi_write(0x03); // Manda modo READ
spi_write(dir_H); // manda direccion alta
spi_write(dir_M); // manda direccion media
spi_write(dir_L); // manda direccion baja

//lectura de CH1
antiguo_dato_23A(); //cont--
down = spi_read(0); //down
antiguo_dato_23A(); //cont--
up = spi_read(0); //up
// dato=make16(up,down);
dato=make16(down,up);

//lectura de CH0
antiguo_dato_23A(); //cont--
down = spi_read(0); //down
antiguo_dato_23A(); //cont--
up = spi_read(0); //up
// dato2=make16(up,down);
dato2=make16(down,up);

printf("%lx %lx %lx\n",dato, dato2, count);
output_high(RAM); //RAM_CS=1 para deshabilitar registro
}

```

## 6\_GPS.h

```
////////////////////////////////////
//PROGRAMACION GPS
////////////////////////////////////

#define LMUX      PIN_A2

char cad_gps[100];
char dato[20];

void limpia_cad_gps()
{
for(int i=0; i<100; i++)
    {
    cad_gps[i]=0;
    }
}

void limpia_dato()
{
for(int i=0; i<20; i++)
    dato[i]=0;
}

////////////////////////////////////
///VALIDACION DE CADENA GPGGA
////////////////////////////////////
/*
boolean validaGGA(char x)
{
if(x>=48 && x<=57)
    return true;
if(x=='N' || x=='S' || x=='E' || x=='W' || x=='M')
    return true;
if(x=='.' || x==',' || x=='*' || x==' ')
    return true;
return false;
}
}
```

```

*/

////////////////////////////////////
///OBTENCION DE CADENA GPGGA
////////////////////////////////////
void leeGPGGA()
{
  setup_uart(9600);
  limpia_cad_gps();
  output_high(LMUX);
  char c;
  double cont1=0x0000, cont2=0x0000;

  //char cad_gps[100];
  //$GPGGA,191718.878,1919.6410,N,09910.8603,W,0,00,50.0,2286.6,M,-
  9.0,M,0.0,0000*77
  int i=0;
  char formato[10]="$GPGGA,";

  /// VERIFICA QUE SEA EL FORMATO GGA
  while( true )
  {
    if( kbhit() )
    {
      c=getch();
      if(formato[i]==c)
      {
        cad_gps[i]=c;
        i++;
      }
      else
        i=0;
    }
    if(i>=7)
      break;
    cont1 = cont1 + 0x0001;
    if(cont1 >= 0xefff)
      cont2 = cont2 + 0x0001;
    if(cont2 >= 0x8fff)
      break;
  }
}

```

```

cont1=0x0000;
cont2=0x0000;

///CAPTURA TREN DE CARACTERES
//i=0;
while( true )
{
if( kbhit() )
{
c=getch();
if(c!='\n' && c!='\r' && c!='\t')
{
cad_gps[i]=c;
//printf("%c",c);
}
else
break;
i++;
}

if(i>=100)
break;

cont1 = cont1 + 0x0001;
if(cont1 >= 0xefff)
cont2 = cont2 + 0x0001;
if(cont2 >= 0x5fff)
break;
}
output_low(LMUX);
}

////////////////////////////////////
///OBTENCION DE UN DATO ESPECIFICO DE LA CADENA GPS
////////////////////////////////////
void ObtenDato(int num)
{
limpia_dato();
int paro=0;
int num_comas=0;

```

```

int ap_letra=0;

while(paro<100)
{
if(num_comas==num)
{
while( cad_gps[par]!=',' && cad_gps[par]!=0 && paro<100 )
{
dato[ap_letra]=cad_gps[par];
ap_letra++;
paro++;
}
return;
}
if(cad_gps[par]==',' )
num_comas++;
paro++;

} //while exterior
return;
}

```

```

////////////////////
///OBTENCION DE CADENA ALEATORIA DE 5 CARACTERES
////////////////////
void ObtenCadenaAleatoria()
{
double cont1=0x0000, cont2=0x0000;
char c;
int i=0;
limpia_dato();
limpia_cad_gps();

output_high(LMUX);
delay_ms(5);
while( cont2<0x000f)
{
if( kbhit() )
{

```

```
c=getch();
cad_gps[i]=c;
i++;
if( i>= 5)
{
output_low(LMUX);
return;
}
}
cont1+=0x0001;
if(cont1>=0xffff)
{
cont2+=0x0001;
cont1=0x0000;
}
}
output_low(LMUX);
printf("TIMEOUT\n");
}
```

## 7\_bluetooth.h

```
#define MUX_B      PIN_A2
////////////////////
//BLUETOOTH
////////////////////
/*
//baud rate
int count=0;
//setup_uart(9600);

output_low(PIN_A1);
delay_ms(500);
output_high(PIN_A1);

while(count<100)
{
//printf("AT+NAMEalpha\r\n");
printf("AT+BAUD7\r\n");
delay_ms(300);
if(kbhit())
    break;
count++;
}
*/

void baudRate(long int x)
{
output_low(MUX_B);
if(x==1200)
    printf("AT+BAUD1\r\n");
else if(x==2400)
    printf("AT+BAUD2\r\n");
else if(x==4800)
    printf("AT+BAUD3\r\n");
else if(x==9600)
    printf("AT+BAUD4\r\n");
else if(x==19200)
    printf("AT+BAUD5\r\n");
else if(x==38400)
    printf("AT+BAUD6\r\n");
```




```
else if(x==57600)
    printf("AT+BAUD7\r\n");
else if(x==115200)
    printf("AT+BAUD8\r\n");
else
    printf("AT");

output_high(MUX_B);
}

void name(char name[8])
{
    output_low(MUX_B);
    printf("AT+NAME%s\r\n",name);
    output_high(MUX_B);
}

void password(char pass[4])
{
    output_low(MUX_B);
    printf("AT+PIN%s\r\n",pass);
    output_high(MUX_B);
}
```

## II Diagramas de la Interfaz de usuario (LabView 2012)

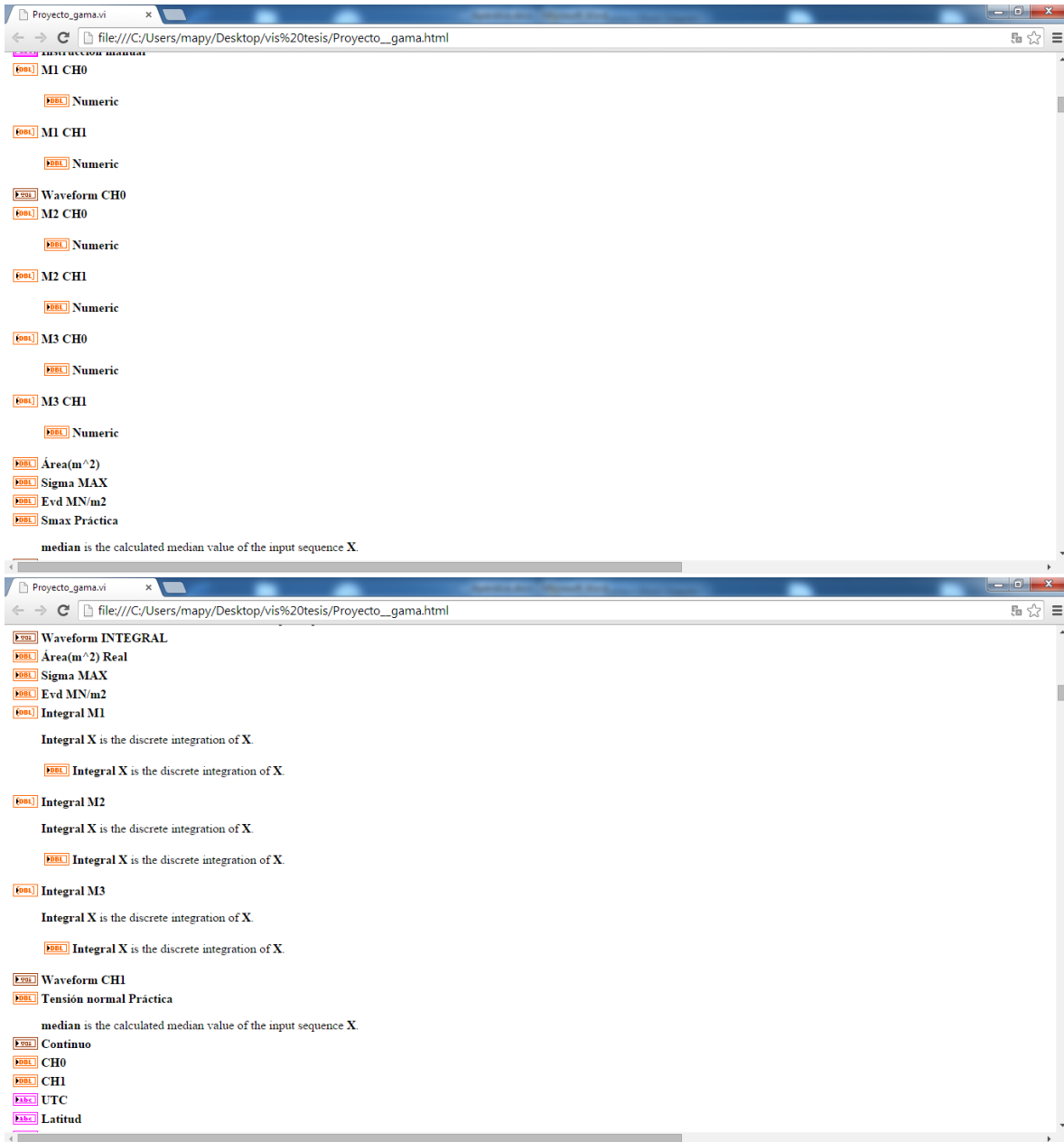


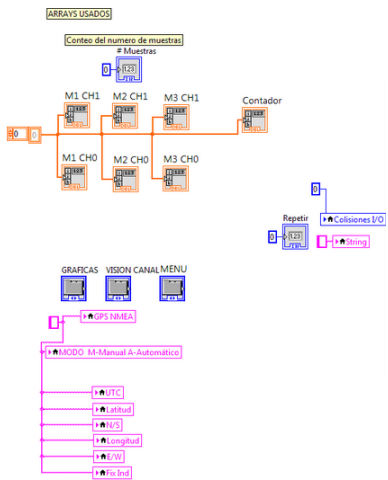
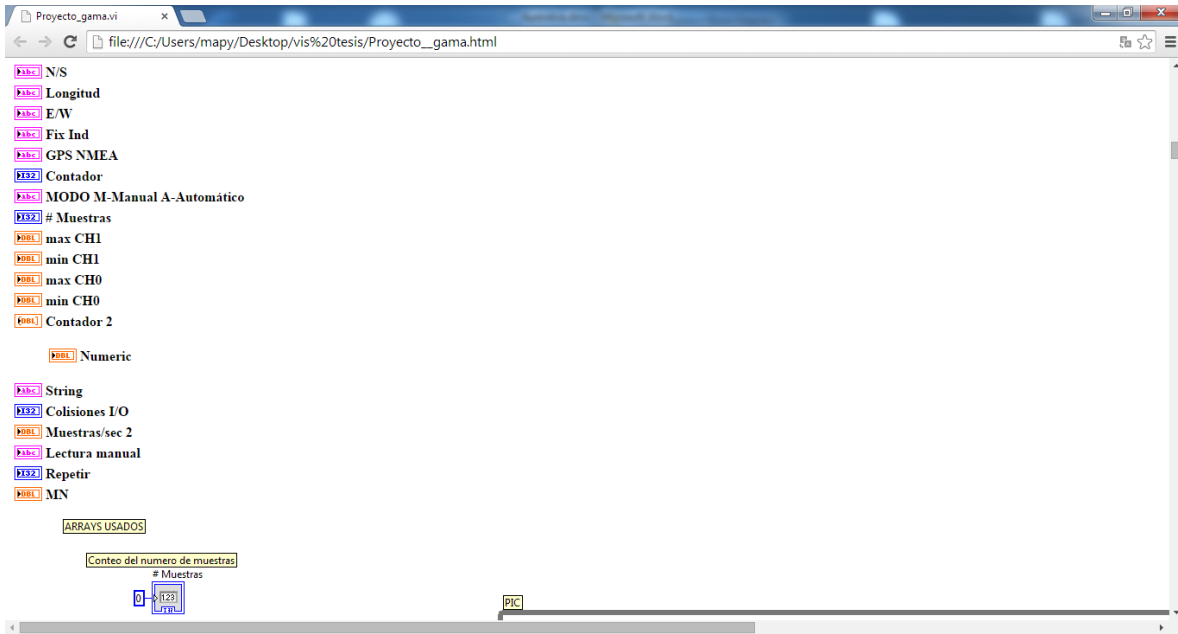
The screenshot displays the LabView interface for 'Proyecto\_gama.vi'. The window title is 'Proyecto\_gama.vi' and the address bar shows the file path: 'file:///C:/Users/mapy/Desktop/vis%20tesis/Proyecto\_gama.html'. The interface is divided into several sections:

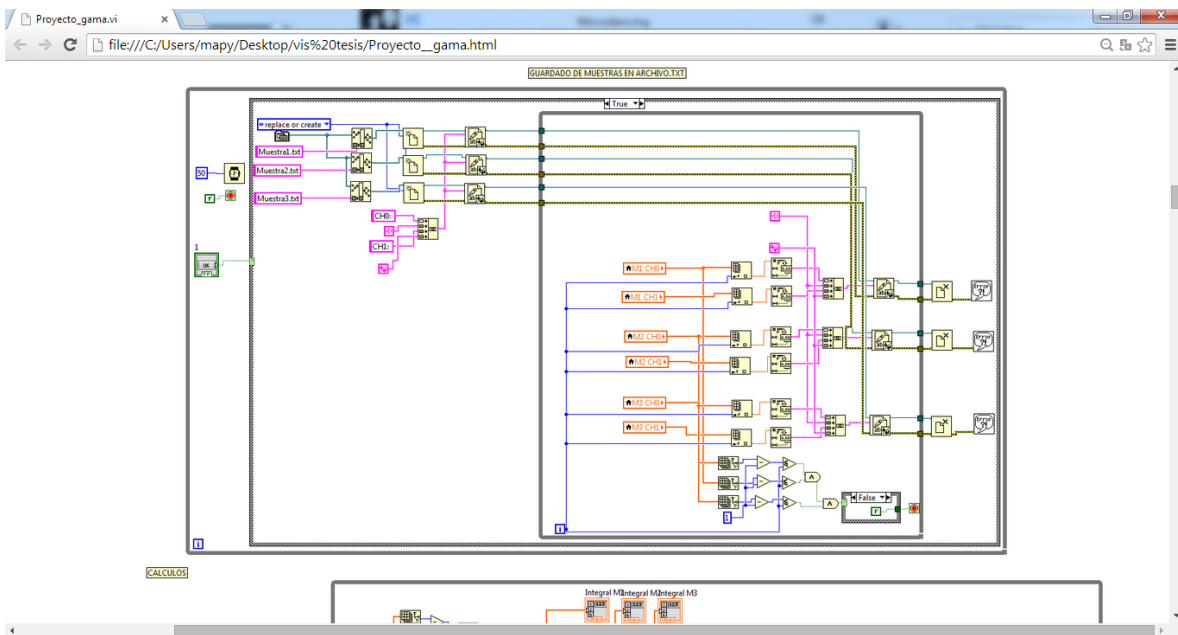
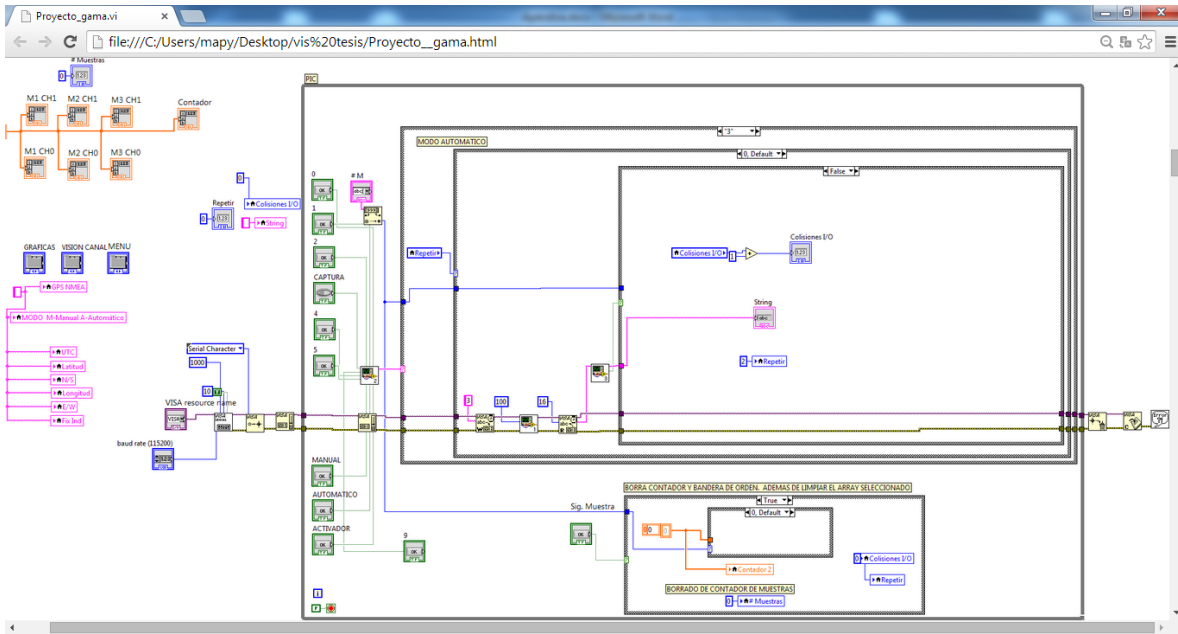
- Left Panel:** A numeric display labeled 'Muestras/sec 2' showing the value '0'.
- Central Panel (PIC):** Contains control buttons for 'CONTINUO', 'MANUAL', 'AUTO', and 'TIEMPO'. It also features a 'Activador (modo manual)' button and an 'AJUSTE CERO' button. Below these are 'PARÁMETROS' including a 'MODO' selector (M-Manual, A-Automático), a 'VER MODO' button, a 'baud rate (115200)' numeric control set to '9600', and a 'VISA resource name' dropdown menu.
- Right Panel:** A plot area with tabs for 'CONTINUO', 'CH0', 'CH1', and 'INTEGRAL'. The plot is titled 'Plot 0' and shows a graph of 'Amplitude' (y-axis, ranging from -2.5 to 2.5) versus 'Time' (x-axis, ranging from 0 to 100). The plot area is currently black, indicating no data is being displayed.

Below the LabView window, a list of controls is shown, each with a small icon and a label:

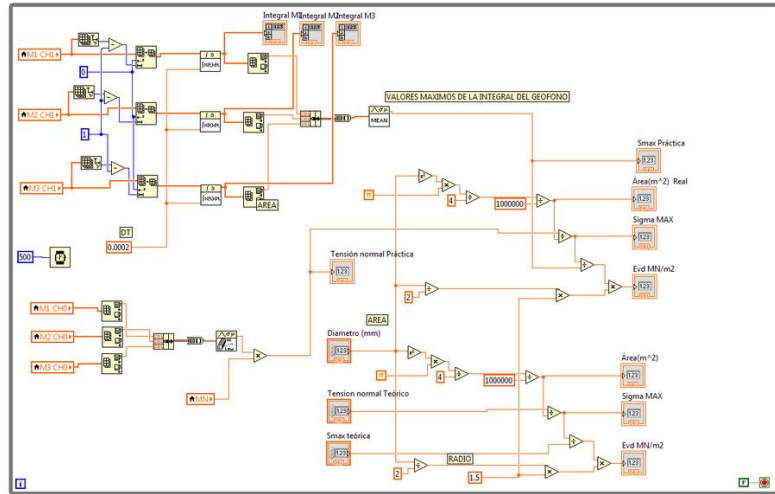
- VISA resource name
- VISA resource name specifies the resource to be opened. The VISA resource name control also specifies the session and class.
- MENU
- Grafica
- Sig. Muestra
- baud rate (115200)
- baud rate is the rate of transmission.
- VISION CANAL
- Diametro (mm)
- Tension normal Teórico
- Smax teórica
- GRAFICAS
- # M
- 1
- 0
- 1 2
- 2
- CAPTURA
- 4
- 5
- 9
- MANUAL
- AUTOMATICO
- ACTIVADOR
- xls
- Instruccion manual





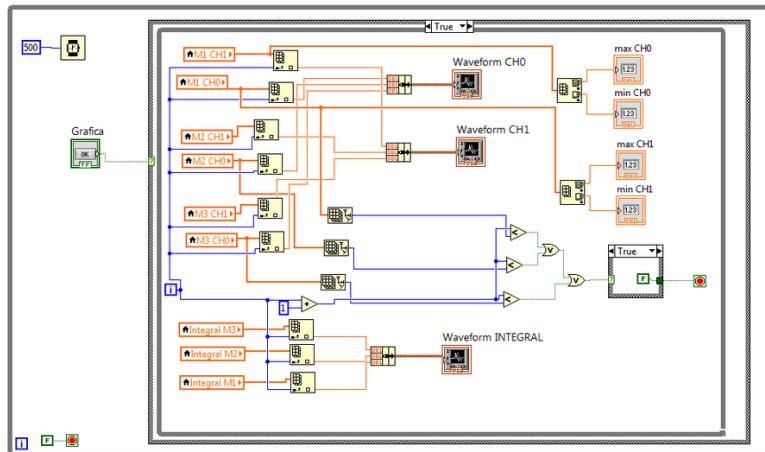


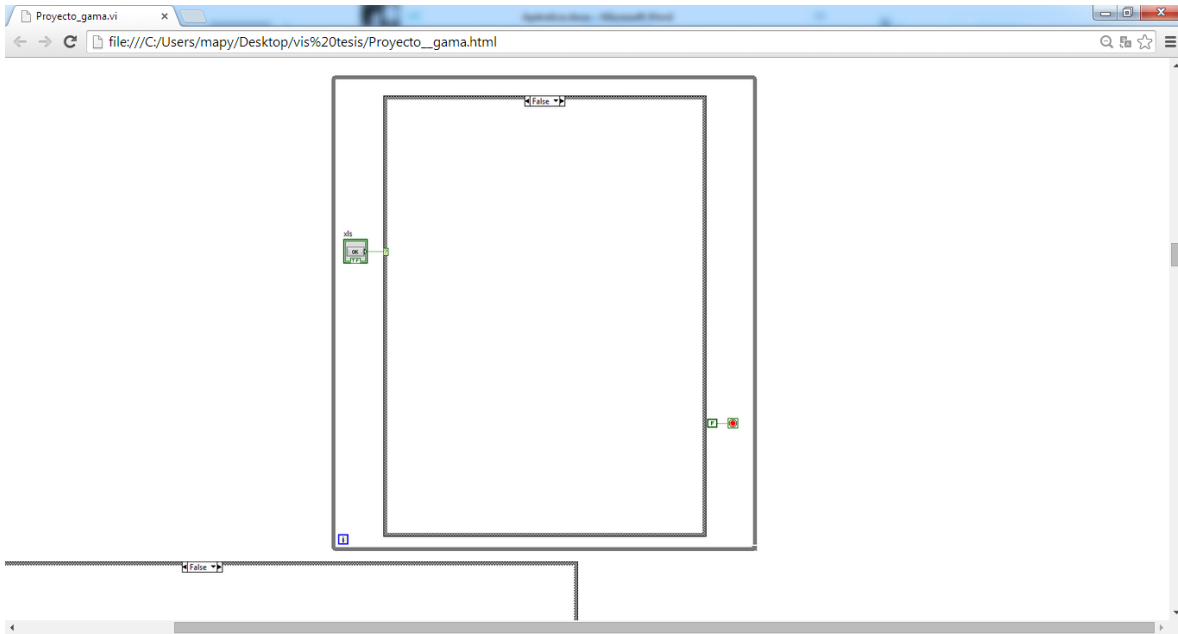
CALCULOS



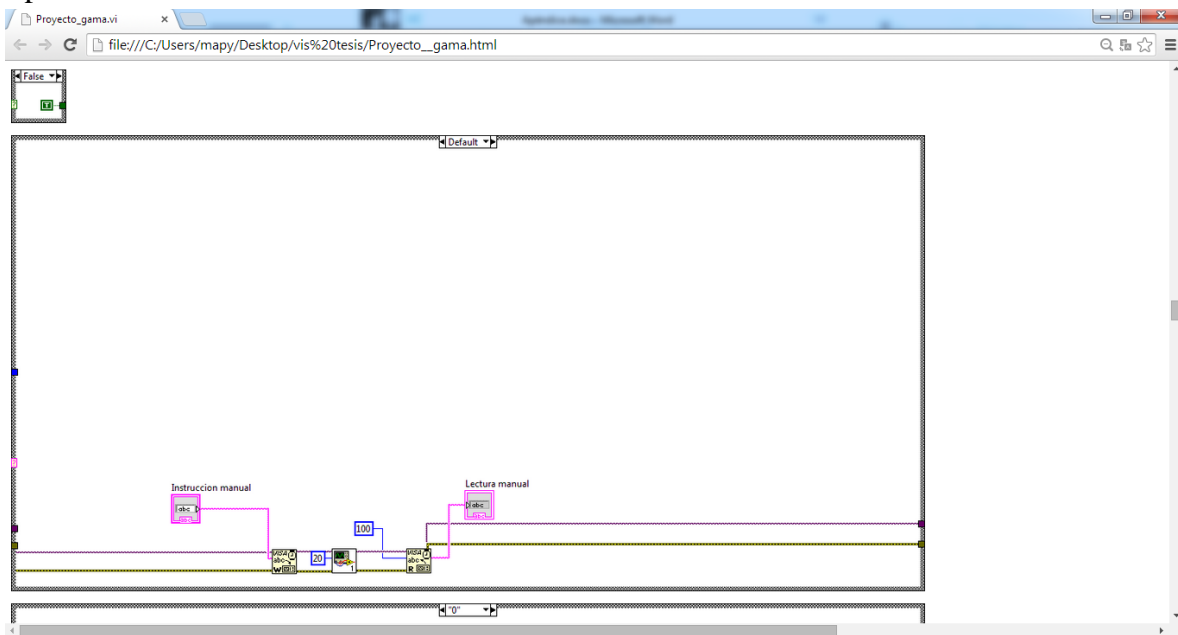
GRAFICA

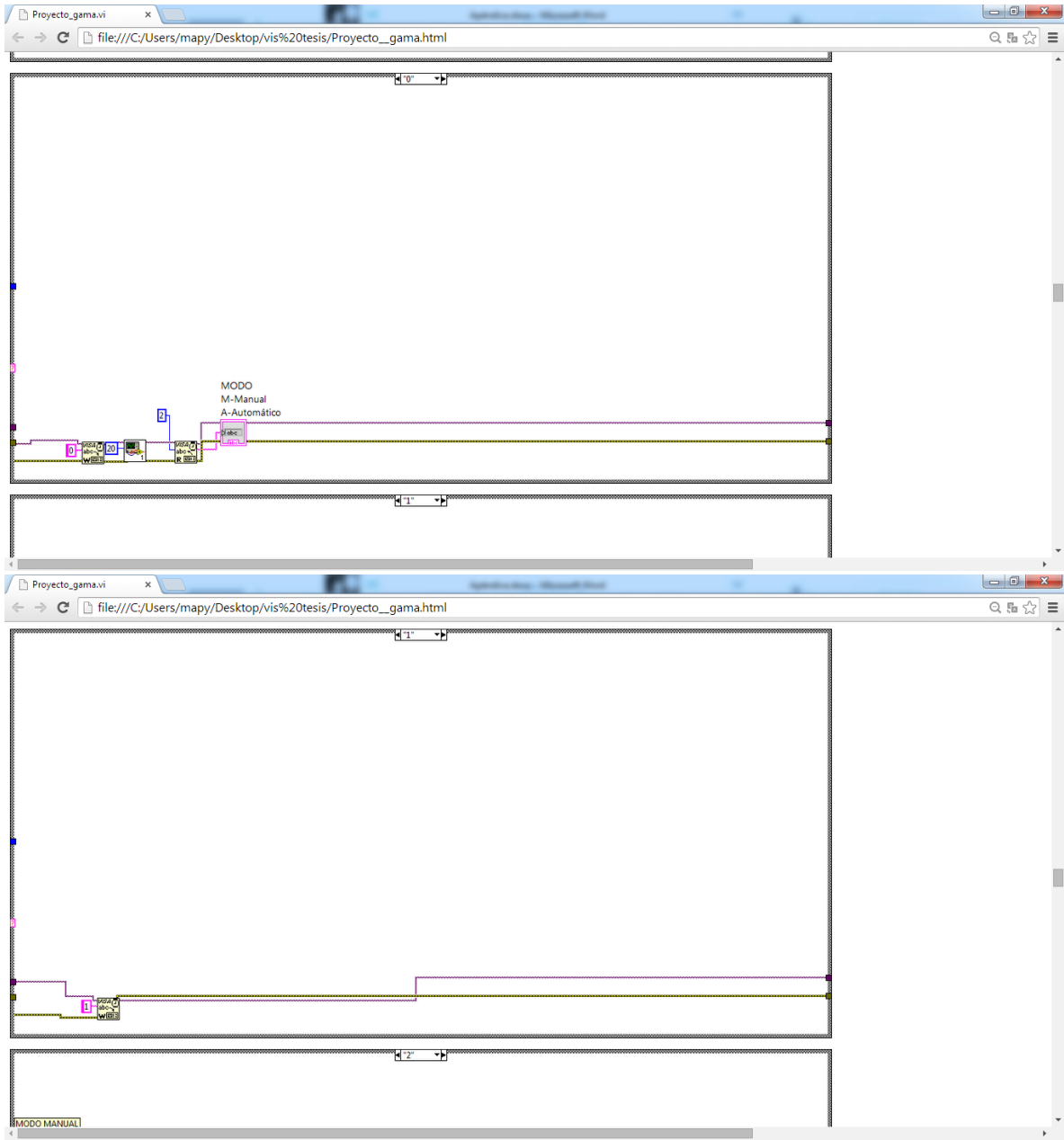
GRAFICA





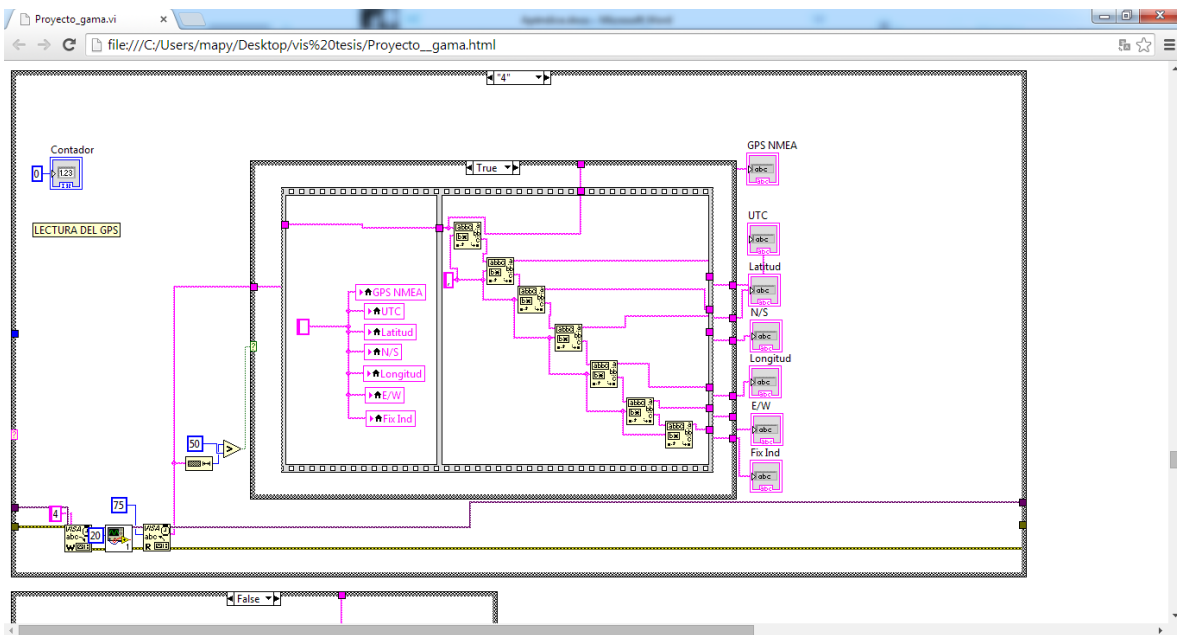
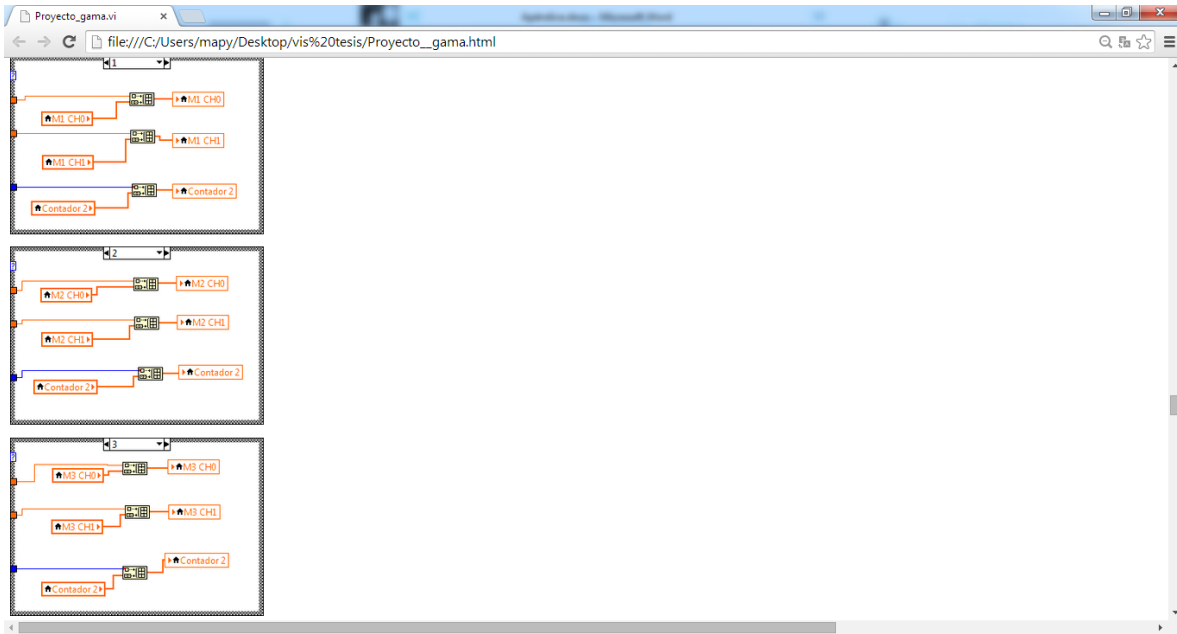
## Opciones de la comunicación con el PIC

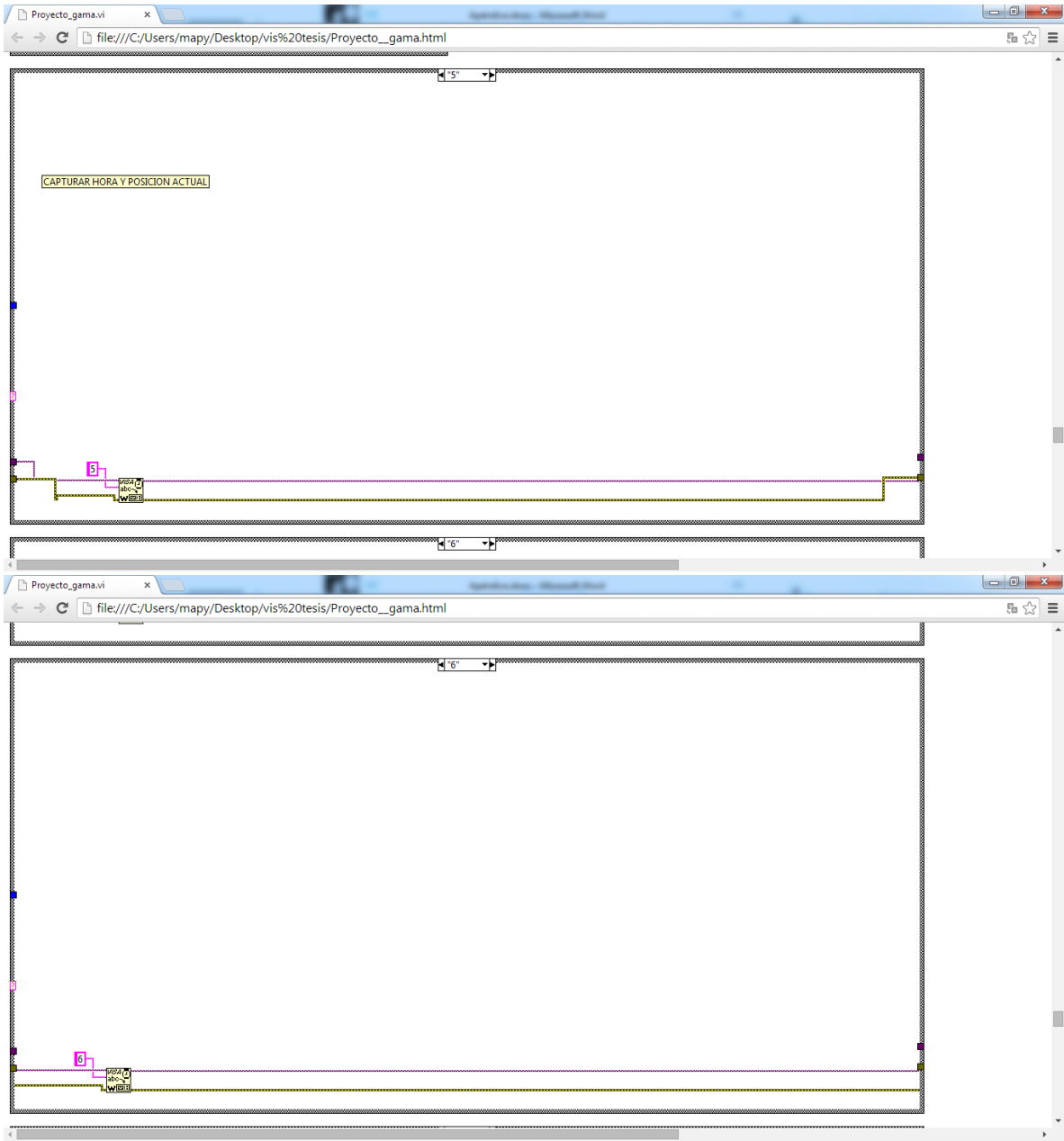


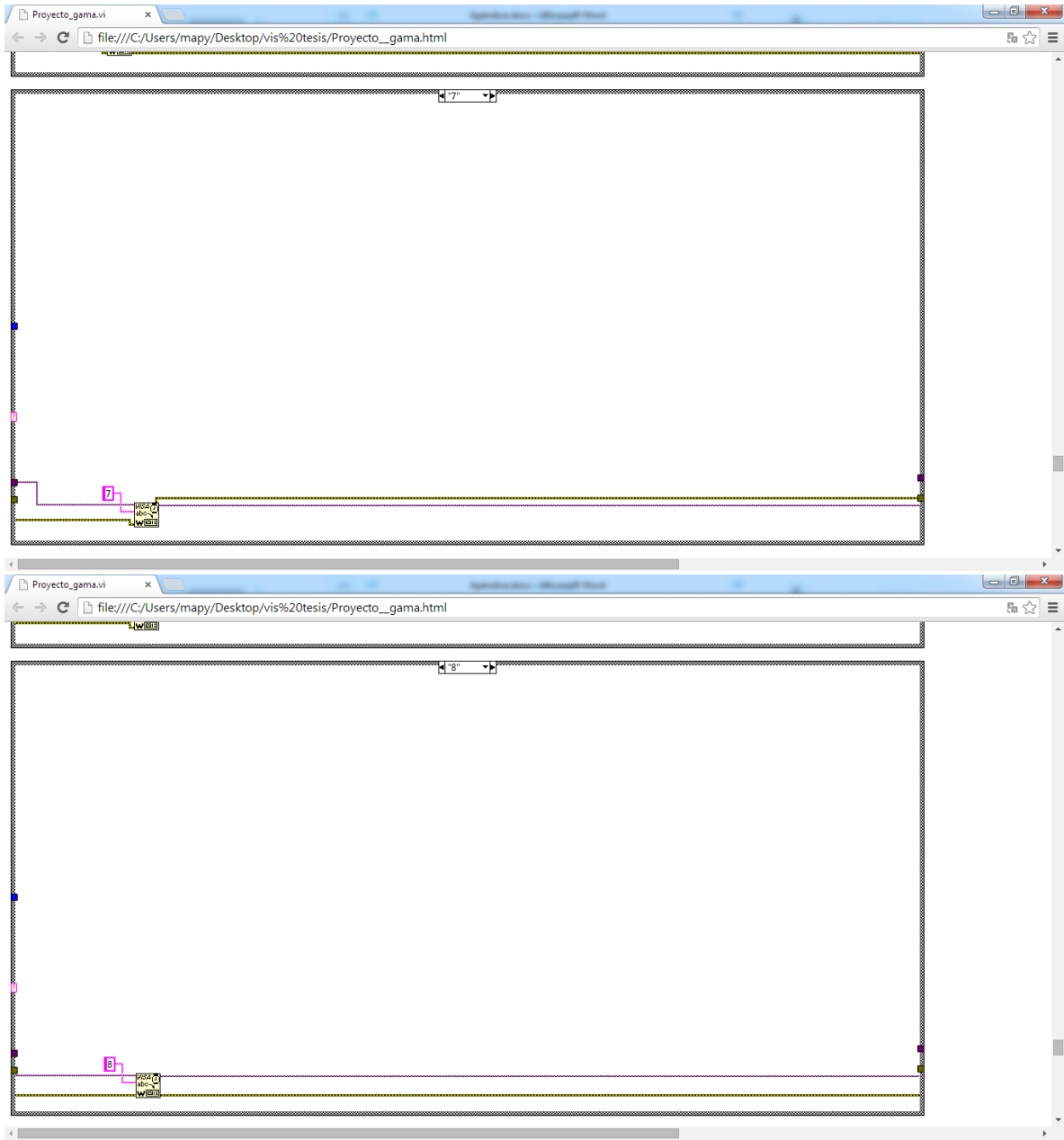


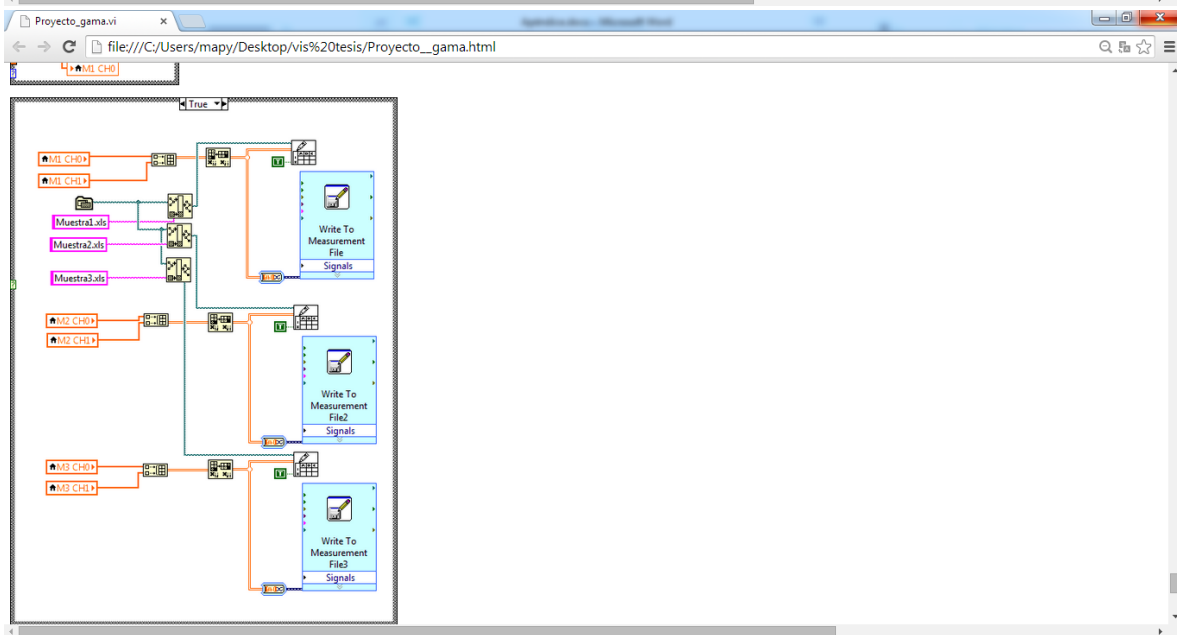
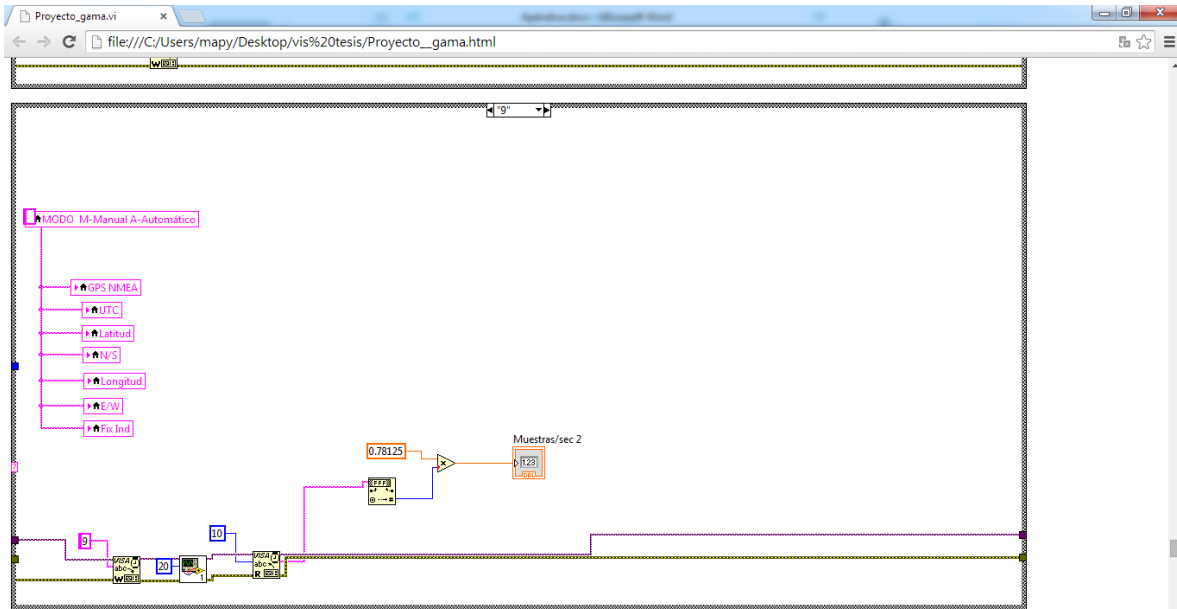


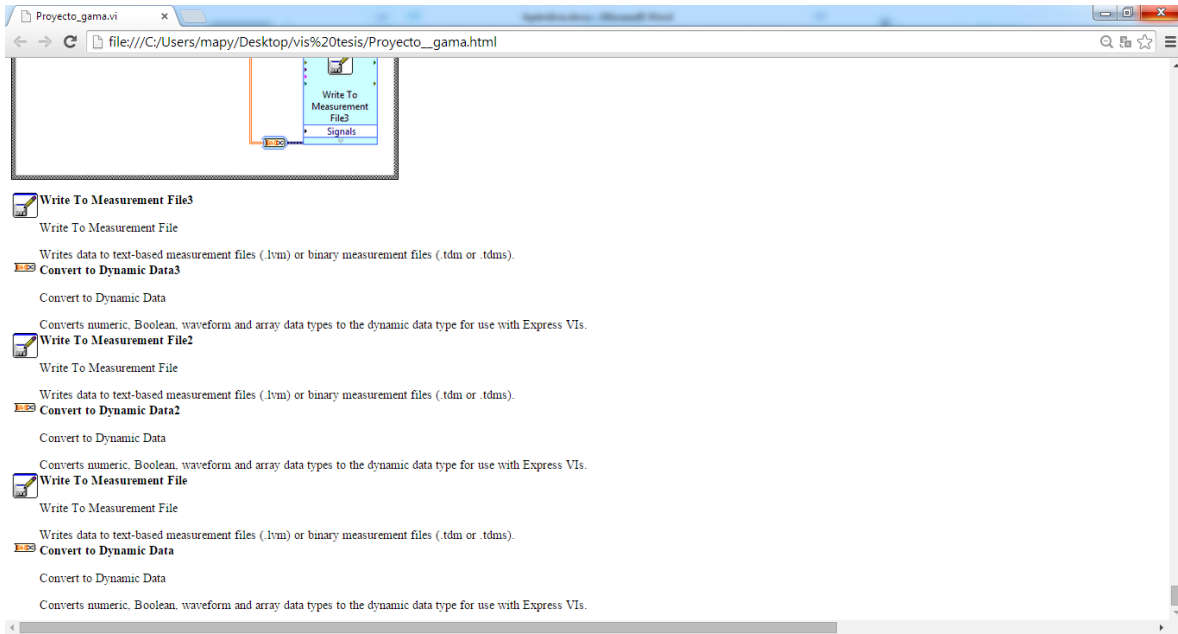












Para mayor información, consultar los archivos incluidos en el disco anexo a este proyecto