

Desarrollo y optimización de la interfaz para la medición e interpretación de las pruebas de tensión en máquinas universales, mediante el uso de la instrumentación virtual

Development and optimization of the interface for the measurement and interpretation of stress tests in universal machines, through the use of virtual instrumentation.

Diana Isabel Valdés Guajardo

Universidad tecnológica del norte de Coahuila, México

nana_valdes@hotmail.com

Resumen

En la actualidad existen una cantidad de Máquinas Universales muy limitada, su principal uso es la aplicación de la prueba de destrucción en probetas de diferentes materiales la cual puede ser aplicada para el análisis de 3 tipos de ensayos para el comportamiento y propiedades de los materiales: el ensayo de tracción, mejor conocido como ensayo de ingeniería, el ensayo de compresión y el ensayo de flexión.

El problema principal es que algunas de las Maquinas Universales carecen de algún elemento en la interfaz gráfica para el control de la prueba, debido a esto, se propuso como objetivo el elaborar una interfaz, mediante el uso de la instrumentación virtual, utilizando para el desarrollo de la interfaz un software cuyo lenguaje de programación es

"La creación de instrumentos virtuales, permite que las maquinas tradicionales y sus funciones vayan a la par del desarrollo de las nuevas tecnologías".

gráfica, ya que para medir e interpretar los datos obtenidos al desarrollar dicha prueba, en los materiales que se encuentran bajo los estándares de las diferentes normas, es muy sencilla su utilización.

Palabras Clave: instrumentación virtual, labview, máquina universal, probetas, materiales.

Abstract

At present there is a very limited amount of Universal Machines, its main use is the application of the test of destruction in test pieces of different materials which can be applied for the analysis of 3 types of tests for the behavior and properties of the materials: the tensile test, better known as the engineering test, the compression test and the bending test. The main problem is that some of the Universal Machines lack some element in the graphical interface for the control of the test, due to this, it was proposed as an objective the elaboration of an interface, through the use of virtual instrumentation, using for the development of the interface a software whose language of programming is graphical, since to measure and to interpret the data obtained when developing this test, in the materials that are under the standards of the different standards, its use is very simple.

Keywords: virtual instrumentation, labview, universal machine, probes, materials.

Fecha Recepción: Enero 2017

Fecha Aceptación: Julio 2017

Introducción

El desarrollo tecnológico de los materiales ha permitido grandes avances en todos los campos de la ingeniería; es por esto que los materiales, sus propiedades y cómo mejorarlas han sido motivo de investigación. En dichas investigaciones, se realizan pruebas de destrucción mecánicas a probetas de diferentes materiales, de acuerdo con la característica mecánica que se desee conocer, como pueden ser la dureza, ductilidad, fragilidad y resiliencia entre otras.

Los resultados de tales ensayos han contribuido directamente al mejoramiento del diseño de elementos de máquinas universales. Los avances tecnológicos actuales, principalmente dentro de la industria, y las normas exigen que los resultados de dichos ensayos sean confiables. Es por esto que, para garantizar la repetitividad en las pruebas mecánicas que se deban de realizar, el uso y la correcta interpretación de los resultados se ha optado por el uso de la instrumentación virtual en los equipos y máquinas que las efectúan.

La Máquina Universal de Ensayos Dinámicos de Tensión a Probetas, que se encuentra presente en la Universidad Tecnológica del Norte de Coahuila, es un dispositivo mecánico, dotado de indicadores de aguja y dial, de resolución limitada, tanto en su escala de ángulo como en su escala de energía, esta

máquina carece de algunos elementos necesarios para el desarrollo óptimo de las pruebas, en su interfaz gráfica, ya que cuenta con algunos limitantes al momento de realizar la medición e interpretación de los datos obtenidos en la aplicación de las pruebas.

Con el fin de mejorar las capacidades del equipo se ha realizado la instrumentación electrónica del mismo, por medio de la adaptación de dispositivos de medición electrónicos a la máquina. Los valores medidos van a un computador, para ser mostrados en una aplicación de LabView, teniendo entonces los resultados en magnitud de la energía requerida en la ruptura de la probeta. Finalmente como verificación, éstos resultados se comparan con los obtenidos por el instrumento de medida original del equipo.

DEFINICIONES

Una maquina universal es una aparato que nos permite realizar una serie de pruebas destructivas a probetas de diferentes materiales, estos ensayos nos ayudan a poder analizar el comportamiento de los mismos a la aplicación de esfuerzos medidos y controlados, de tensión, compresión o deflexión.

Un sistema de instrumentación virtual es una aplicación para computadora que permite que un operador pueda manejar y/o controlar un instrumento físico, que se encarga de monitorear variables importantes para el desarrollo y/o análisis de un proceso por medio de una interfaz.

Un sistema de adquisición de datos (DAQ) es el proceso de medir con una PC un fenómeno eléctrico o físico como voltaje, corriente, temperatura, presión o sonido. Un sistema DAQ consiste de sensores, hardware de medidas DAQ y una PC con software programable. Comparados con los sistemas de medidas tradicionales, los sistemas DAQ basados en PC aprovechan la potencia del procesamiento, la productividad, la visualización y las habilidades de conectividad de las PCs estándares en la industria proporcionando una solución de medidas más potente, flexible y rentable.



Figura 1.- Componentes de un sistema de adquisición de datos.

Los elementos básicos de un sistema de medida son el sensor, el circuito de acondicionamiento de señal y el sistema de visualización. Sin embargo, muchos sistemas de medida son más complejos e incluyen una conversión analógico-digital, procesamiento de señal y transmisión de datos. Dicha transmisión puede ser antes o después de la digitalización.

De acuerdo con la naturaleza de la variable se tienen diversas señales características sobre las cuales habrá que realizar diferentes tareas de adecuación primaria como son amplificación, reducción, linealización, eliminación de offset de cero, etc.

Los ensayos destructivos en materiales, tienen una gran importancia dentro de la industria, ya sea, para desarrollar nuevos productos y/o diseñar nuevos materiales que cumplan con las necesidades específicas requeridas para alguna aplicación. Uno de los ensayos más utilizados para el análisis de las propiedades mecánicas en un material, es el ensayo destructivo de tensión, el cual se realiza en probetas del material que se quiere analizar.

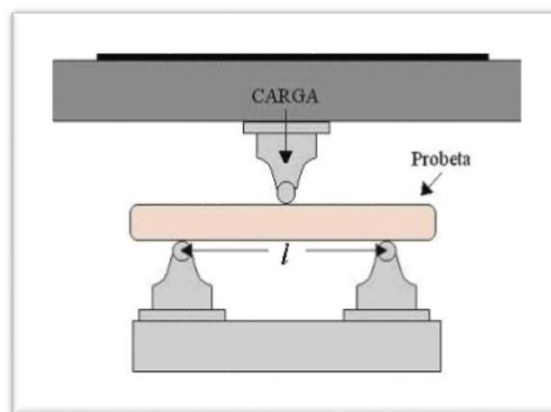


Figura 2.- Ensayo de Tensión.

Para este tipo de pruebas es necesario que la probetas tengan un diseño específico y estandarizado (ANSI) para poder tener la certeza de que los datos que se obtengan sean lo más confiables posible. Este ensayo se puede realizar de forma analítica, aplicando métodos matemáticos y también de forma mecánica, sometiendo a estiramiento (tensión), a la probeta por medio de un aparato llamado Máquina Universal, la cual muestra los datos de forma visual mediante

instrumentos analógicos, que nos ayudaran a interpretarlos por medio de la generación de una gráfica llamada Grafica de Esfuerzos, que nos permite conocer la resistencia del material al esfuerzo sometido.

Actualmente los sistemas de instrumentación nos permiten recopilar información generada por diferentes procesos, además de tener la opción de controlarlos y generar gráficos en los cuales se pueden interpretar los datos obtenidos. Esta recopilación de datos se realiza mediante la utilización de sensores y hardware especializado, por ejemplo la NI MyDAQ de National Instrument,

la cual funciona a la par con el software para adquisición de datos LabView, que permite trasladar los datos físicos a datos electrónicos, manejables en una computadora. Este tipo de sistemas debe de contar con una interfaz gráfica para el usuario de la maquina (operador), la cual deberá ser amigable para el desarrollo confiable de la prueba. La interfaz debe de mostrar los datos generales necesarios en forma de tabla y generar la gráfica de esfuerzos de forma automática, para después poder ser comparados con los datos teóricos calculados por el operador.

Es necesario incluir una etapa de potencia para el control de la maquina por parte del sistema de instrumentación que permite proteger en todo momento los componentes del hardware de posibles daños por sobrecarga

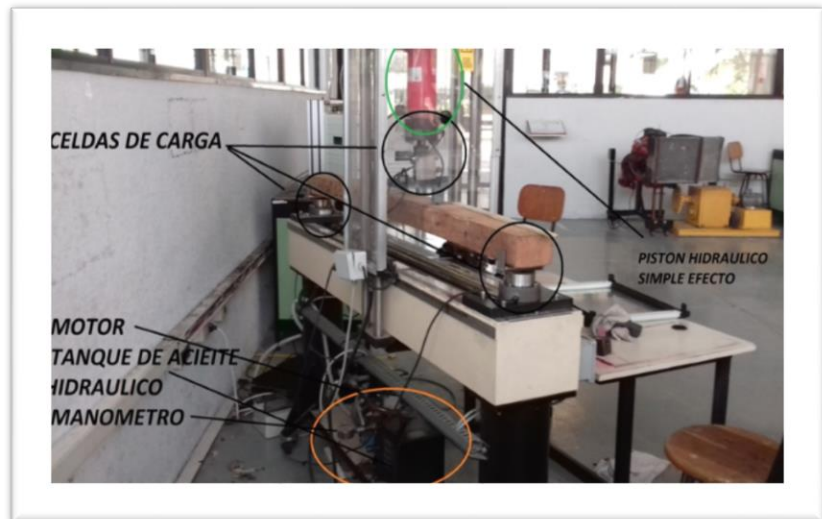


Figura 3.- Elementos de una máquina universal.

MÉTODO

La máquina flexionadora cuenta con tres celdas de carga que tienen las mismas especificaciones técnicas, dos en la parte de debajo que cuentan con un juego de piezas y tornillos para sostener la viga de manera recta y adecuada para que al incidir la carga del pistón sobre ésta los sensores lean de manera correcta el peso que se ejerce sobre la viga. Por tanto podremos conocer el peso ejercido en ambos extremos de la viga; la otra se encuentra en la parte anterior a la punta del vástago con sus respectivas piezas para acoplarse al pistón y medir de forma correcta la carga al ser enfocada al centro del sensor, esta celda servirá para medir la carga aplicada directamente por el pistón y con eso hacer los respectivos cálculos.



Figura 4a.- (Superior e izquierda).-
Celda de carga extremo izquierdo

Figura 4b.- (Derecha).- Celda de
carga para medir la fuerza aplicada.

Así es como empezamos a hacer pruebas de calibración haciendo una sumatoria de fuerzas en los extremos, sabiendo que el resultado debe coincidir con el peso de la lectura del sensor del centro, determinamos la fuerza ejercida por el pistón en base a fórmulas tomando en cuenta algunas áreas así como la presión que la bomba entregaba leída en el manómetro. Las mediciones debían ser precisas por lo que procedimos a utilizar las herramientas adecuadas para este tipo de trabajos y la más importante de ellas fue el micrómetro. Con los datos a la mano y los cálculos rápidos hechos

descubrimos ciertos valores erróneos al comparar lo teórico con lo real, por lo cual hicimos cálculos un ay otra vez así como el ajuste de la lectura de las celdas hasta que determinamos que había ciertos errores en una de las celdas que después de varios intentos decidimos solucionar por software, aunque aun así debíamos tenerlo bajo observación (los valores entregados) por si los datos no eran proporcionales, esto quiere decir que debíamos verificar si el error seguía siendo el mismo conforme aumentaba la carga o si variaba excesivamente.



Figura 5.- Medición de diámetro exterior del cilindro hidráulico de simple efecto.

Lo siguiente que había que lograr era la interfaz gráfica con la que interactuaría el usuario que se dispusiera a utilizar la máquina, haciéndola sencilla, precisa y clara, indicando la función de cada uno de los indicadores y controles, y decidimos hacerlo un tanto moderno. Esta encomienda fue porque, como se había mencionado, la unidad de control y monitoreo de la máquina estaba en mal estado con muchos de sus componentes dañados o inservibles. Dicha interfaz de control debía realizarse en un ordenador mediante el software de National Instrument LabVIEW y éste debía ser capaz de monitorear en tiempo real las variables de entrada directamente de los sensores, además de tener que visualizarlo en la pantalla y así poder tener una idea grafica de lo que la máquina está

haciendo, así como almacenar ciertos valores para después crear una tabla de esfuerzo-deformación de la viga.

Con nuestra propuesta de utilizar la compactDAQ no se requiere de algún acondicionamiento de señal físico en alguno de los sensores, esto se realizará por medio de software debido a que LabVIEW es un potente programa con una diversidad de funciones. Lo que hacemos es calibrar los diferentes sensores introduciendo fórmulas en el programa de diagrama de bloques por aquello de la carga que ejercen las piezas que sostienen la viga, antes mencionadas, y también se tendrán diversos controles para calibrar los sensores así como para indicar si se desprecia o no el peso de la viga, ya que en la teoría se realizan cálculos con y sin éste valor.

Otra de las señales importantes es la entregada por el transductor de presión, por medio del cual conoceremos el valor de la presión hidráulica que existe entre la bomba y el cilindro. El mismo funciona de forma similar a las celdas de carga ya que es un sensor de salida igual a 2mV/V , aunque al contar con un manómetro nos será mucho más fácil el sensor ésta variable ya que sólo leímos el voltaje que entrega y lo comparamos con la lectura del manómetro y como el voltaje de salida del sensor es proporcional a la presión determinaremos ésta última a través de una regla de tres simples tomando en cuenta algunos otros valores para una lectura correcta.

Por medio de éste sabremos cual es la presión hidráulica que se genera al momento en que el pistón aplica la carga sobre la viga que se encuentra entre las dos celdas de carga.

Para regular la carga del pistón se utiliza una válvula reguladora pilotada, controlada por medio de un potenciómetro al variar de 0 a 10 volts la señal de entrada al circuito de control de la misma, el cual da en su salida la señal equivalente para la carga deseada. La variación de la señal que controlaremos puede ser con el potenciómetro (de forma manual) como ya mencionamos o por medio de la salida analógica de alguna de las tarjetas o módulos de NI para hacerlo por medio del panel de control a través del ordenador. Con la primera propuesta haremos uso del módulo NI 9263, módulo de 4 salidas analógicas de la clase C, para manipular ese voltaje que regulará la carga a través del panel de control. En caso de utilizar una myRio o MyDaq utilizaremos alguna de sus salidas analógicas de 0 a 10 v con las que cuenta. Para esta parte del sistema también se debe monitorear la señal del potenciómetro para saber el porcentaje de la carga máxima ejercida, lo cual se hará solo si se utiliza una myRio o MyDaq ya que cuentan con entradas analógicas que van desde -10 hasta 10v.

Por otra parte, también se nos indicó el cambio de sensor que medía la deflexión, ya que éste contaba con un potenciómetro lineal de 10k, el cuál debía hacer contacto directamente con la viga para medir

la deflexión de forma correcta, lo cual suponía un desgaste mayor y probablemente una duración en buenas condiciones considerablemente mayor a alguno que pudiese medir ésta variable a distancia, sin contacto físico por lo cual decidimos utilizar un sensor ultrasónico modelo HC SR04, junto con el que debimos implementar un microcontrolador (Arduino uno o nano), esto para enviar el pulso de 10us a través del pin “TRIGGER” para que el circuito del sensor envíe 8 pulsos a 40kHz y leer el tiempo en μS que permanece en alto el pin “ECHO” hasta que las ondas de sonido regresen al parlante y lo ponga en nivel lógico 0 para así tomar el valor del tiempo en la escala de microsegundos y determinar la distancia (mm) en base a la velocidad del sonido (340 m/s).

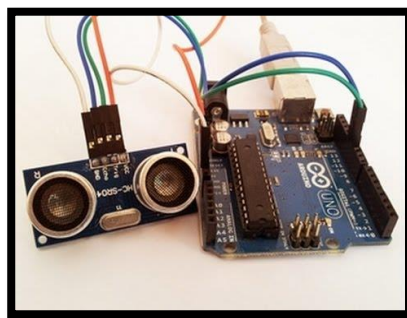


Figura 6 - Arduino UNO y sensor ultrasónico HC SR04

El uso del arduino se hará para aislar la programación del sensor de todo lo demás en el VI, utilizando VISA para leer el dato que el μC nos entregará, además de que se debe convertir el dato a entero para poder trabajar con él (el dato en tipo String no nos serviría para nuestro propósito). En la siguiente figura se encuentra capturada la parte en LabVIEW que se encargará de leer y convertir el dato de entrada.

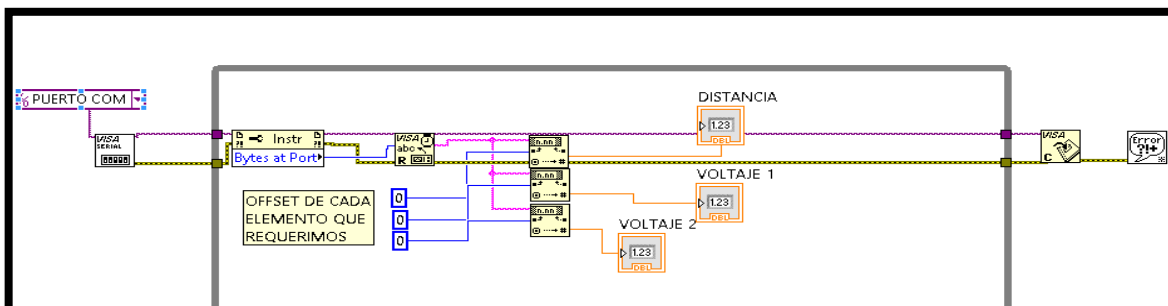


Figura 7.- Parte del VI encargada de leer el dato del Arduino y convertirlo a tipos de datos numéricos de doble precisión para leerlos.

En la figura 7 se puede observar tres números para el offset en la conversión de datos, los cuales modificaremos al finalizar el VI de acuerdo con el dato de entrada (cadena de caracteres) para identificar las posiciones en las que empiezan los valores que leeremos, así

```

archivo Editar Programa Herramientas Ayuda
sketch_aug12b $

const int TriggerPin = 2; // declaramos la entrada del pin TRIGGER
const int EchoPin = 3; // y del ECHO

long Duracion = 0; //variable para almacenar el tiempo

void setup(){
  pinMode(TriggerPin, OUTPUT); // Pin del TRIGGER lo declaramos como salida
  pinMode(EchoPin, INPUT); // Pin del ECHO lo declaramos como entrada

  Serial.begin(9600); //activamos la comunicacion serial a 9600 baudios
}

void loop() {
  float Dist=distancia(); //llamamos a la funcion distancia y almacenamos
                        // el resultado en la variable Dist
  Serial.print("Distancia= "); //Imprimimos el mensaje de
  Serial.print(Dist); //Salida junto con el dato para saber
  Serial.println(" mm"); // la distancia y la escala
  delay(100); //esperamos para hacer perceptible el
                //mensaje y la distancia
}

float distancia() {
  digitalWrite(TriggerPin, LOW); // ponemos el pin TRIGGER en estado bajo
  delayMicroseconds(2); // esperamos 2 microsegundos
  digitalWrite(TriggerPin, HIGH); // ponemos el pin TRIGGER en HIGH
  delayMicroseconds(10); // mantenemos activo el pin durante 10 microsegundos
  digitalWrite(TriggerPin, LOW); // ponemos el pin del TRIGGER en LOW

  Duracion = pulseIn(EchoPin, HIGH, 5000); //Contamos el tiempo(uS) que tarda en regresar
                        //el ultrasonido emitido
  float distancia_mm=(duracion/2.9)/2; //hacemos el calculo correspondiente
  return distancia_mm; //retornamos el valor tipo float a la funcion
}

```

Figura 8.-Programa para lectura de sensor ultrasónico.

LabVIEW convertirá el dato tipo cadena (parte del mismo) en valores numéricos para así introducirlos a las fórmulas para determinar la deflexión y demás variables que demanda el proyecto para el μC .

El programa encargado de hacer la lectura del sensor ultrasónico es el que se muestra en la siguiente imagen, programado bajo la IDE de Arduino el cual viene comentado para la comprensión del mismo.

El programa anterior es la parte de lectura del sensor ultrasónico.

La distancia se calcula midiendo el tiempo que tarda en regresar el sonido y multiplicándole por la velocidad del mismo (aprox. 340m/s).

DISCUSIONES.

Una vez concluidas las pruebas de los sensores, actuadores y salidas de la propuesta uno, decidimos investigar acerca de otras formas de llevar a cabo la lectura de las celdas así como de la manipulación de las señales de salida requeridas por el proyecto, al hacerlo encontramos la alternativa de utilizar amplificadores de instrumentación para el acoplamiento de las señales de las celdas de carga así como de una myRio para la lectura de los mismos, con la ventaja de que sería un sistema de control más compacto y al requerir de menos elementos (en este caso módulos) supondría un coste mucho menor a la propuesta primaria.

Para el acoplamiento de las señales se utilizarán varios IA AD620 (4) que son amplificadores de instrumentación con alto rechazo al modo común por lo cual supone una señal con menor variación en su salida, que se utilizan para medir tensiones de nivel muy bajo en presencia de señales indeseadas (ruidos), así como también cuenta con un arreglo de resistencias interno con el cual se designan un par de pines para que por medio de una resistencia de entrada se pueda ajustar el factor de amplificación que viene dado por la fórmula $G = \frac{49.4\text{k}\Omega}{R_G} + 1$ y que nos servirá para determinar la ganancia que se tendrá para después obtener la relación de la carga con las señal amplificada.

Se dispondrá de un amplificador para celda de carga con una resistencia de ganancia de 100Ω para obtener una ganancia de 500 (no exactamente, pero sí un valor muy cercano), con lo cual la señal de la celda variará hasta 5V, lo que supondría una variación de 1mV por cada kg fuerza ejercido sobre el sensor. A continuación se muestra el cálculo para la ganancia:

$$G = \frac{49.4\text{k}\Omega}{R_G} + 1 \quad \therefore \quad R_G = \frac{49.4\text{k}\Omega}{G-1}$$

Sustituyendo:

$$\frac{49.4k\Omega}{(500) - 1} = \frac{49.4k\Omega}{499} = 98.99\Omega$$

El valor de la resistencia estándar más cercano es 100Ω que son las que utilizaremos con una tolerancia de 2% a ½ watt

Dentro del mismo circuito de los IA se utilizarán amplificadores operacionales del tipo LM358 (5) para el ajuste del offset (ajuste de cero) que se implementará para ingresar un voltaje de referencia al pin Ref de cada IA que se ajustará por medio de un trimpot de 10kΩ conectado en serie a una resistencia de 1kΩ para introducir voltaje al amplificador que trabaja como buffer para evitar errores por la impedancia.

Así mismo dentro de la misma placa en la que se pondrán todos los elementos para la manipulación de entradas llevará un par de reguladores de tensión de modelos 7805 y 7809 para suministrar el voltaje adecuado a los amplificadores y a unos cuantos relevadores que también se montarán sobre la placa para controlar las salidas y poder hacer el encendido de la máquina desde el ordenador, cada uno con su respectivo transistor, diodo y resistencia para así activarlo por medio de una salida digital de la myRio. Un relevador está designado a encender la bomba, y se utilizarán otros dos para activar la carga, los cuales se activarán al mismo tiempo indicándoselos por software, y uno último que será para tener un paro de emergencia en el panel de control.

También se diseñó un placa que será en la que se conecten las celdas y ésta enviará la señal a la placa principal, esta placa contiene, además, un relevador para la selección de la señal que se utilizará para controlar o regular el pistón (conmutar) y un arduino nano para leer e interpretar la señal del sensor.

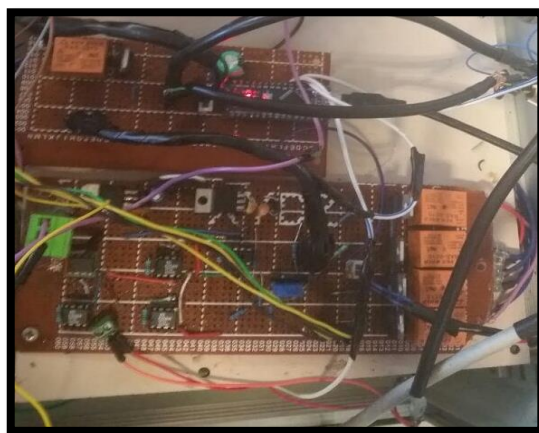


Figura 9.- Circuito de control y adquisición de señales.

El circuito de los amplificadores para las celdas es algo similar a lo que se muestra en la siguiente imagen, en la cual se muestra la simulación del mismo en el software de NI Multisim, en el cual comprobamos si es que, en teoría, el AD620 detectaría y amplificaría valores muy bajos y una vez que la simulación funcionó como esperábamos procedimos a hacer pruebas conectando todos los elementos en una tablilla de pruebas (protoboard) hasta estar seguros del correcto funcionamiento de cada IA. Una vez que el amplificador funcionó de forma adecuada procedimos a montar el circuito sobre la placa fenólica.

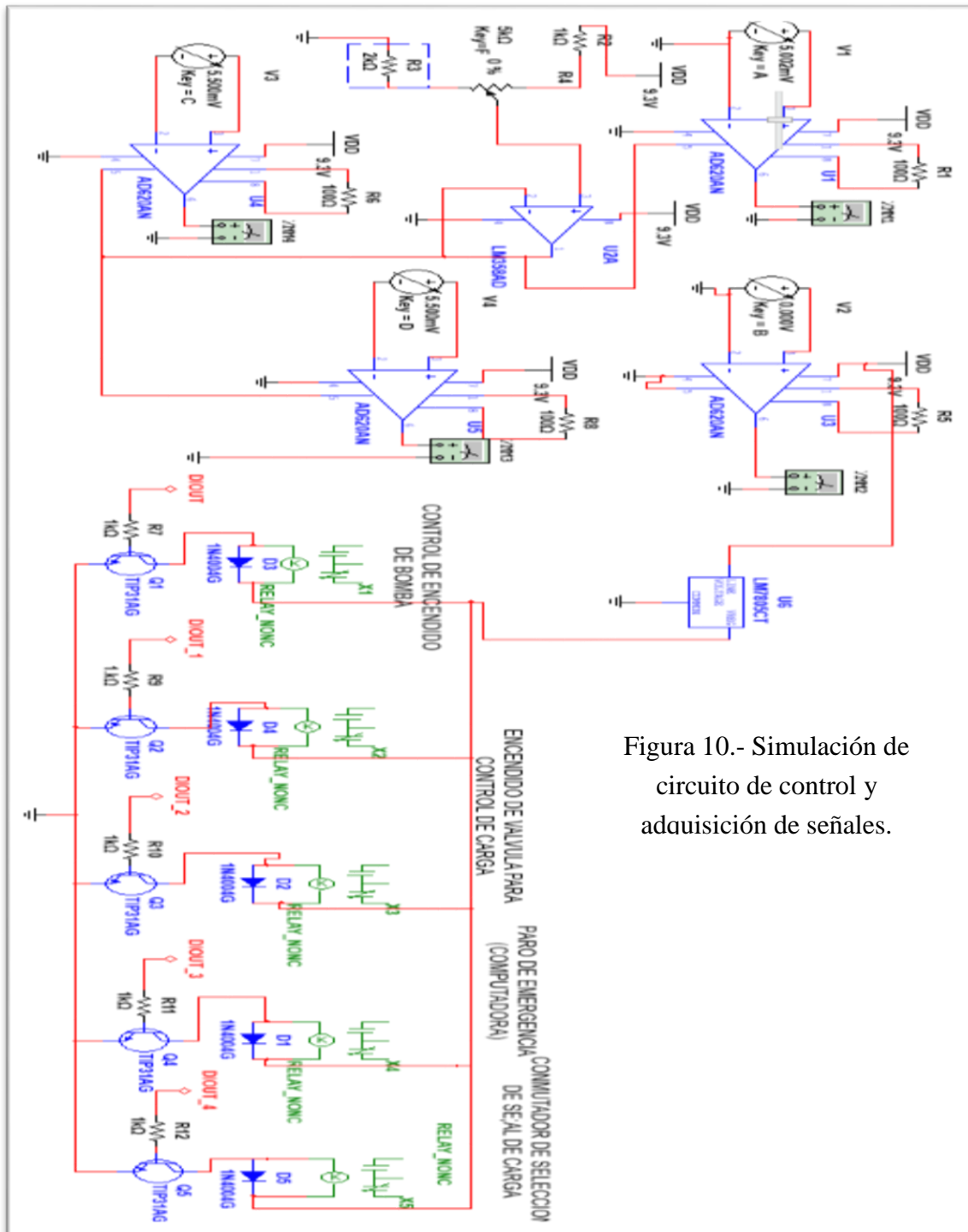


Figura 10.- Simulación de
circuito de control y
adquisición de señales.

Para probar que los circuitos y la programación son correctos nos dimos a la tarea de conectarlo de forma temporal utilizando una myDAQ y un Arduino Nano, la primera para leer la señal de voltaje del potenciómetro que controla el porcentaje de la carga así como también leer la señal de la celda de carga central (la que esta atornillada al pistón) , debido a que es la celda de carga sobre la que se ejercerá más fuerza por tano el voltaje puede variar más y las entradas de la myDAQ soportan hasta 10V de lectura suficientes para evitar un daño en algún otro elemento. En esta conexión se utiliza en Arduino nano no sólo para leer el sensor ultrasónico sino para leer el voltaje de las celdas restantes ya que la myDAQ sólo cuenta con dos entradas analógicas (véase la hoja de datos anexa a la carpeta que contiene el documento), así que utilizaremos 3 entradas analógicas y un par de digitales para la distancia. El Arduino lo usaremos para las otras 2 celdas de carga y el transductor de presión ya que la señal (voltaje) que estos entregan, ya amplificada, es menor a la celda del pistón, lo cual supone que es menos probable que estas superen los 5v que soportan las entradas de nuestro μ C. Lo anterior lo demostramos a continuación:

La carga máxima que podrá ejercer el pistón es de 24000N y sabiendo que la relación de la señal de salida del IA y la carga es de 1mV por cada kgf y que $1\text{kgf}=9.81\text{N}$ el voltaje máximo de salida es de 2.44V por lo que si se tiene un offset de más de 2.6V dañaría la entrada del Arduino.

Dada la necesidad de utilizar más funciones de nuestro microcontrolador el programa que se deberá cargar en éste debe ser ampliado, el mismo se muestra en la imagen 13. Nótese que se conservan las funciones creadas en el programa antes mostrado debido a que sólo ampliaremos el mismo para leer las señales analógicas correspondientes.



ultrasonico \$

```
const int TriggerPin = 2; // pin del TRIGGER
const int EchoPin = 3; // pin del ECHO

long Duracion = 0;

void setup(){
  pinMode(TriggerPin,OUTPUT); // Pin del TRIGGER lo ponemos en output
  pinMode(EchoPin,INPUT); // Pin del ECHO lo ponemos en input
  Serial.begin(9600); // activamos el puerto serie para ver los resultados
}
void loop(){
  float dist=distancia();
  Serial.print(dist);
  Serial.print(" mm");
  Serial.print(volt(14)); //Mandamos a la funcion volt() para leer
  delay(1) //voltaje del pin analoico 0
  Serial.print(volt(15)); //y hacemos lo mismo con AN1
  delay(90);
}
float volt(int pin){
  int suma=0;
  for(int n=0;n<5;n++){
    int x=analogRead(pin);
    suma=suma+x;
    delay(1);
  }
  int valor=suma/5;

  float voltaje=valor*(5/1023);
  return voltaje;
}
float distancia(){
  digitalWrite(TriggerPin, LOW); // ponemos el pin TRIGGER en LOW
  delayMicroseconds(2); // esperamos 2 milisegundos
  digitalWrite(TriggerPin, HIGH); // ponemos el pin TRIGGER en HIGH
  delayMicroseconds(10); // lo tenemos activado durante 10 milisegundos
  digitalWrite(TriggerPin, LOW); // ponemos el pin del TRIGGER en LOW

  Duracion = pulseIn(EchoPin,HIGH,5000); // Esperamos a que el pin del ECHO devuelva

  float distancia=(Duracion/2.9)/2; //hacemos el calculo correspondiente
  return distancia;
  delay(5);
}
```

Guardado.

Figura 11.- Programa que se cargara en Arduino para lectura de señales.

En la siguiente figura podemos observar la conexión de los circuitos antes mencionados a la máquina.

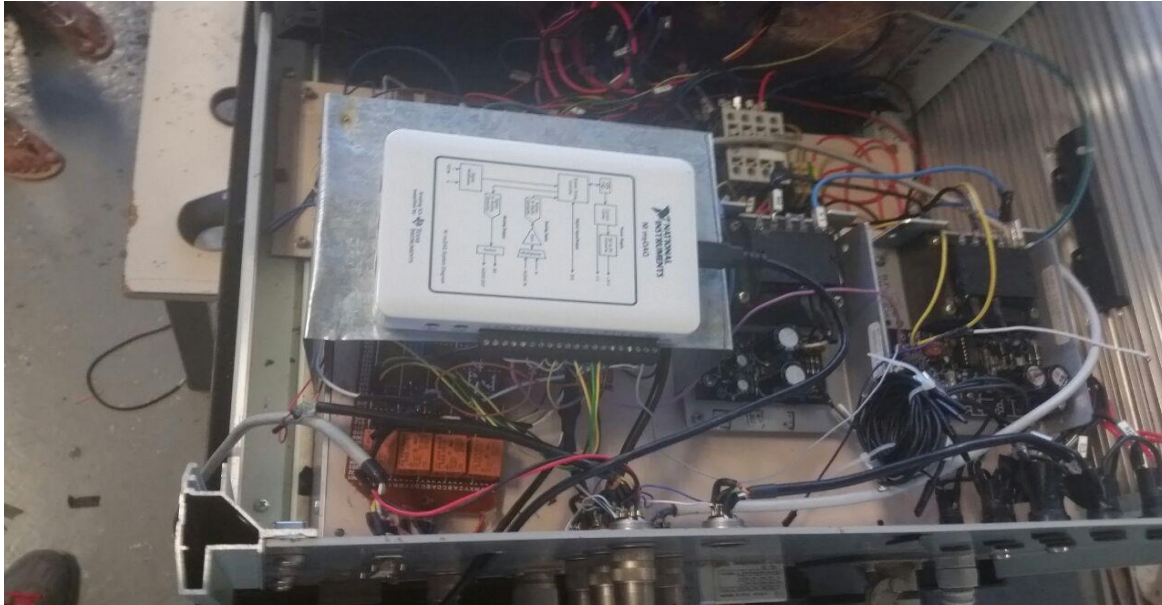


Figura 12.- Montaje de los dispositivos de control en la máquina.

En la Figura anterior se puede observar la base que se creó para colocar la myDAQ debido a la falta de espacio.

a) Software de Interfaz con el Usuario.

El software de interfaz con el usuario es una de las partes más importantes del sistema. Es en este punto donde se realiza el tratamiento de la información y la toma de decisiones. Las funciones que desempeña el software son:

- Comunicación con el μC para la adquisición de datos.
- Almacenamiento de la información.
- Acceso a los registros.
- Toma de decisiones.
- Despliegue de la información.- La interfaz permite visualizar al usuario en tiempo real la medición de la prueba de tensión.
- Control manual para realizar mantenimiento de los motores.
- Diagramas de flujo.

b) Panel de control.

Todo esto de la lectura de las señales con el Arduino nano y la myDAQ lo podremos observar en un panel frontal de LabVIEW del cual crearemos una aplicación ejecutable (.exe) para el rápido acceso a ello.

Con el uso de la pantalla de la computadora para monitorear y controlar la máquina en el panel de control de la misma sobraban elementos que decidimos retirar de la máquina y con esta acción era necesario crear un nuevo panel físico para el control de la máquina, ya que en el antiguo los orificios le darían una mala imagen a la misma.

Para crear el nuevo panel de control físico tuvimos que sacar medidas de la parte frontal de la caja de control para después cortar una lámina de acero acorde a dichas medidas como se puede observar en las siguientes imágenes. El corte lo hicimos con una cortadora con la que la escuela contaba la cual se puede apreciar en la figura 13.

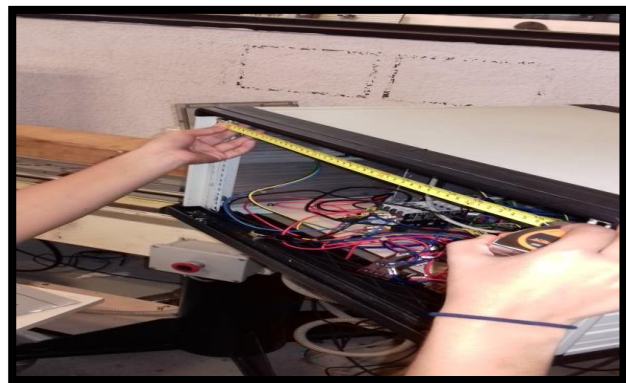


Figura 13.-Corte de lámina del panel. Figura 14.- Medición para la elaboración del panel.

Después de cortar la lámina procedimos a hacer los orificios para los elementos a usar, como lo son los conmutadores, el potenciómetro y un botón. Dichos orificios fueron hechos en el taller de máquinas y herramientas haciendo uso del taladro de banco como podemos ver a continuación.



Figura 15.- Proceso de perforación para colocar los elementos del panel de control.

Una vez perforada la lámina procedimos a pintarla y personalizarla, identificando cada uno de los elementos que contenía.

c) **Panel virtual.**

Para el monitoreo y control de las señales desde la computadora debimos crear una interfaz diferente a la planeada por el uso de los dispositivos mencionados en la última conexión. Dicha interfaz, llamada panel frontal, contiene los elementos para visualizar y controlar la máquina identificados y visualmente parecidos a la máquina.

Para el panel frontal decidimos crear una representación de la máquina en SolidWorks para agregar una imagen que se pareciera a ésta dando como resultado lo que se muestra en la siguiente imagen. Con este modelo 3D de la máquina identificamos en el panel frontal a que sensor pertenece cada una de las señales obtenidas, así como también podemos visualizar si el pistón se encuentra contraído o no, haciendo una representación visual de cuando el vástago sale.

El panel también contiene todos los botones necesarios en el proyecto, uno para controlar el encendido de la carga, otro para la bomba, un paro de emergencia, además de un botón para conmutar la señal de lectura para la regulación de la carga. El botón “cero” tiene la función de poner a cero la carga entregada por las celdas. Además empleamos un control numérico tipo “Knob” para regular la carga. En la siguiente imagen podemos ver los botones mencionados con su respectivo identificador.

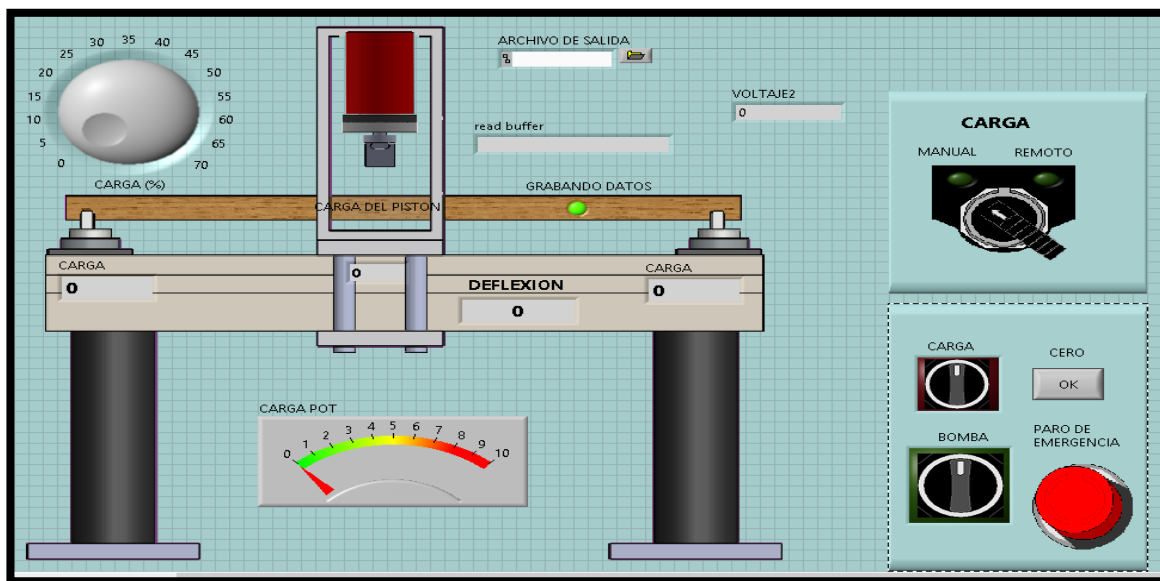


Figura 16.- Panel virtual de monitoreo y control de la máquina.

Entre los indicadores tenemos los que nos muestran la carga ejercida en cada punto clave, uno tipo “meter” para leer la señal del potenciómetro además de indicadores booleanos para saber si la carga o bomba está encendida y conocer la fuente de señal para regular la carga, así como también para saber si el VI está guardando datos en un archivo para Excel.

d) Exportando datos.

Además de mostrar los resultados de cada elemento, el programa debe ser capaz de almacenar éstos en un archivo que sea legible para Excel, por lo que decidimos utilizar la función “*Write to spreadsheet*” para guardar los datos en el momento en el que el programa lo indique, siendo ese momento un periodo de tiempo definido después de que se ejerza la carga.

Los datos se almacenarán en un archivo de tipo **.csv**, que es un archivo de valores separados por comas de Microsoft Excel en el cual se creará una tabla en la que se indiquen los valores de cada una de las celdas y de la deflexión, mostrando también la hora y fecha en la que fue almacenado. Dicho documento deberá estar presente antes de iniciar la prueba (Nuestro programa no lo creará), solo debemos indicar la ruta en el panel virtual a través de un **Path**, que es un control en LabVIEW para introducir directorios, para saber dónde se almacenarán los datos. Es importante mencionar que el archivo **.csv** no debe estar abierto o usándose por otro programa ya que al momento de iniciar la prueba se generará un error en la misma indicando que no se puede acceder al archivo.

Una vez que el archivo es modificado y los datos se almacenan se debe detener la prueba (el VI) para no alterarlo. El archivo acomodará los datos en forma de tabla (no tiene el formato de tabla solo la distribución de los elementos similar a esta) indicando la fecha y hora de la lectura de las celdas, la deflexión y si se desea, la presión de la bomba.

En la siguiente imagen se muestra la parte del **VI** que se encarga de acomodar estos datos en tipo arreglo de cadena de caracteres para enviarlos al archivo **.csv**. En ella se puede apreciar que utilizamos la función **Elapsed Time**, esto con el fin de activar el envío de datos al documento de Excel al cumplir ciertas condiciones, como lo será el activar la carga principalmente y oras más. La función mencionada también cuenta con una entrada para indicar el reinicio del conteo por lo que al desactivar la carga se reinicia el contador para cuando se vuelva a cumplir la condición dada por defecto. Esta función se utiliza para enviar datos solo por un periodo de tiempo predefinido por nosotros.

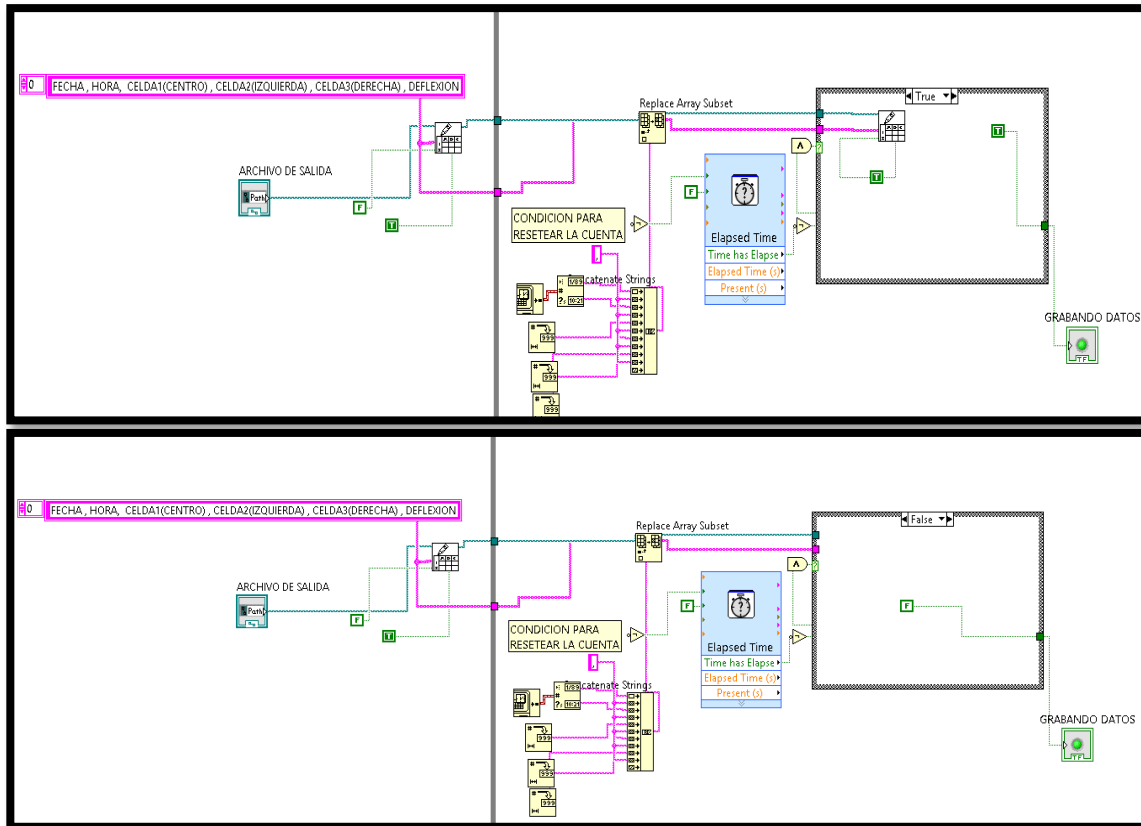


Figura 17a (superior) Diagrama de bloques encargado de exportar los datos a Excel con condición en verdadero.

Figura 17b (inferior) Diagrama de bloques para exportar a Excel con condición en falso.

En las figuras anteriores podemos apreciar el diagrama de bloques encargado de exportar los datos a un archivo legible para Excel. La imagen Xa muestra la condición en verdadero o con el tiempo de grabación transcurrido. La Xb muestra esta condición en falso en la cual se puede apreciar que el programa no hará más que poner en falso el indicador booleano.

Primero se crea un encabezado para cada columna de la tabla, después este se envía al archivo mencionado en el Path de control que se puede ver en las imágenes. Luego de esto entran al ciclo while tanto el Path de salida de la función como el arreglo de cadenas de caracteres y se unen por medio de la función Concatenate Strings que une todos los datos de entrada en una sola cadena de caracteres, antes de esto cada dato numérico también se pasa a este tipo de dato. Una vez convertidos y unidos, los datos se escriben en la hoja de datos de Excel si es que se cumplen ciertas condiciones

y se enciende un indicador para saber si se ejecuta correctamente. En caso de que no se cumpla dicha condición el indicador de “Grabando datos” permanecerá inactivo.

Utilizamos los elementos de DAQmx Data Acquisition para los datos de entrada y salida, utilizando el canal de salida digital controlando las salidas con controles booleanos para activar/desactivar relevadores, así mismo usamos el bloque que crea el canal para leer las entradas analógicas para después descomponerla para leer cada una por separado y así conocer el voltaje que requeríamos saber.

Las entradas digitales también fueron utilizadas para leer un botón que se encuentra en el panel de control.

El bloque de salida digital fue utilizado para regular la carga del pistón como ya se había mencionado.

Cada bloque de lectura/escritura de datos genera un error si encuentra que algo está mal en la conexión o el tiempo de ejecución y cierra el programa, así como también si no hay respuesta del puerto serie del Arduino.

e) Comparación teoría/práctica.

Una vez conectado todo y creado el VI completo pudimos empezar a probar la máquina prácticamente terminada, con lo que decidimos hacer el cálculo que nos diría si los sensores arrojaban los datos correctos. Hicimos el cálculo de la deflexión a través de la siguiente fórmula:

$$\delta_c = \frac{PL^3}{48EI}$$

Donde δ_c es la deflexión de la viga al centro.

P Es la fuerza ejercida sobre la viga en N, Lb o kgf

L Es el claro de la barra en m o pulgadas.

E Módulo de elasticidad en Pa o PSI.

I Momento polar de inercia (plg⁴ o m⁴).

El momento polar de inercia se calcula por medio de la fórmula

$$I = \frac{bh^3}{12}$$

El sensor nos dio una fuerza aplicada en la celda del centro de 1080 kgf y marcó una deflexión de 8mm. La barra tiene una longitud de 1.6m con 3in de base por 4.437in de altura.

Primero convertimos los kgf a N.

Si $1\text{kgf}=9.81\text{N}$ determinamos que:

$$P = (1080\text{kgf})\left(\frac{9.81\text{N}}{1\text{kgf}}\right) = 10594\text{N}$$

Después calculamos el momento polar de inercia.

$$I = \frac{(3\text{in})(4.5\text{in})^3}{12} = 12.78\text{in}^4$$

Convertimos a sistema métrico

$$\text{Dado que } (1\text{m})^4 = (39.37\text{in})^4 \therefore 1\text{m}^4 = 2\,402\,490.39\text{in}^4$$

Por ende para calcular el $(12.78\text{in}^4)\left(\frac{1\text{m}^4}{2\,402\,490.39\text{in}^4}\right)$:

$$\delta_c = \frac{(10594\text{N})(1.6\text{m})^3}{48(11\text{GPa})(9.48\text{E}^{-6}\text{m}^4)}$$

$$\delta_c = 8.66\text{mm}$$

La pantalla del ordenador indicó una deflexión de 8mm para esa carga y de igual forma hicimos otros cálculos con carga máxima dando una deflexión de 10mm en lo real y en lo calculado.

Conclusiones y recomendaciones

Para optimizar el desempeño del sistema de monitoreo de la Máquina Universal en la prueba de flexión a probetas, se pueden atender a las siguientes recomendaciones:

- Utilizar una computadora industrial en lugar de una PC convencional, por su mayor y más rápido análisis.
- Construir una interfaz multiplataforma, para que sea compatible con sistemas operativos diferentes a Windows como Linux, Unix o Mac, mejorando así la versatilidad del equipo.
- Es posible incrementar la distancia entre el área de medición de la prueba y la computadora mediante comunicación inalámbrica entre los sensores.
- Sustituir el protocolo de comunicación RS-232 por comunicación Ethernet, para tener acceso al programa desde internet.
- Agregar un sistema de control manual en los motores para situaciones de emergencia.

Bibliografía

- Gutiérrez, J., Orozco, A., & Pérez, S. (1970). Instrumentación para un sistema electromecánico.
- Nolasco, J. J. M., López, L. A. R., Arana, J. D. J. Z., Lesso, A. V., & Guerra, J. F. Instrumentación de Variables Biométricas para el Análisis de Pruebas de Esfuerzo utilizando LabVIEW.
- Humberto, M. A., Fernando, M. C., & Diego, C. G. INSTRUMENTACIÓN DE UNA MÁQUINA UNIVERSAL DE PRUEBAS BALDWIN.
- Ziegler, D., Insausti, J., Lucaioli, A., Iurman, L., Cuenca, L., & Benedetti, P. INSTRUMENTACIÓN DE UNA PRENSA PARA REALIZAR ENSAYOS SWIFT.
- Niño, P., Avilés, O., Saavedra, J., Orjuela, M., & De La Hoz, M. (2001). Módulo de adquisición para prueba de esfuerzo cardiovascular (MAPEC). In Memorias II Congreso Latinoamericano de Ingeniería Biomédica. La Habana: Editorial UH.
- Villarroel Tovar, J. M. (2015). Artículo Científico-Diseño e implementación de un sistema automático de ensayos de tensión para la máquina universal Tinius Olsen de la empresa Cedal SA.
- Alvarez-Alvarez, J. C., Álvarez-Anton, J. C., Blanco, C., Campo, J. C., Ferrero, F. J., González, M.,... & Viera, J. C. (1970). Curso de instrumentación electrónica integrado en una experiencia docente interdisciplinar.
- Meneses Benavides, G. A. (2008). Aplicaciones avanzadas de transferencia de datos de medición con software de instrumentación virtual. revista virtual universidad católica del norte, (24).
- Swain, N.K., Anderson, J.A., Ajit Singh, Swain, M., Fulton, M., Garrett, J., Tucker, O., “Remote data acquisition, control and analysis using LabVIEW front panel and real time engine” SoutheastCon, 2003. Proceedings. IEEE, 4-6 April 2003, pp. 1 – 6.
- National Instruments, “Getting Started with LabVIEW”, Evaluation Version 7.0”, National Instruments, Abril 2003.
- Belletti, A., Borromei, R., Ingletto, G.: EQVAPSIM: A Vapor–Liquid Equilibria of Binary Systems, Computer Simulation by LabVIEW, J. Chem. Educ., Vol. 85 No.6. (2008).
- Lajara, J., Pelegri, J.: LabVIEW: Entorno gráfico de programación. Alfaomega, México D. F. (2007).

Lázaro, A., Río, J.: LabVIEW: Programación gráfica para el control de instrumentación. Thomson, Madrid. (2005).

National Instruments Corporation. DAQ NI 6013/6014 User Manual. 2002.

Meneses Benavides, G. A. (2008). Aplicaciones avanzadas de transferencia de datos de medición con software de instrumentación virtual. revista virtual universidad católica del norte, (24).

Méndez, a. H., & espinosa, m. J. A. M. Electrocardiógrafo para pruebas en esfuerzo basado en instrumentación virtual.

Vargas-Alas, L. G., Castillo-Barahona, R., & Loría-Salazar, L. G. (2012). ENSAYOS DE TENSIÓN AXIAL AL ACERO DE UN PUENTE MODULAR TIPO BAILEY. Programa Infraestructura del Transporte (PITRA), LanammeUCR.

Ruiz Suárez, O. V. (2014). Ensayos de tensión y flexión sobre una nueva propuesta de junta de pilotes hincados (Doctoral dissertation, Escuela Colombiana de Ingeniería Julio Garavito).

Torres, J., & Redondo, J. (2009). Reparación y automatización de una máquina universal de ensayos. Ciencia e Ingeniería, 30(2), 171-179.