



**Módulo de instrumentación con interfaz HMI y software libre  
para prácticas de medición de masa con celda de carga**  
*Instrumentation module with HMI interface and free software for load  
cell mass measurement practices*

Landy Inga Bertha Ivelia<sup>1</sup>

Ing. Damian Barahona Zambrano, Mg.<sup>3</sup>



0000-0003-2298-1754

<sup>1</sup> Instituto Superior Tecnológico Tsa'chila, Ecuador [berthalandy@tsachila.edu.ec](mailto:berthalandy@tsachila.edu.ec)

<sup>2</sup> Instituto Superior Tecnológico Tsa'chila, Ecuador [damianbarahona@tsachila.edu.ec](mailto:damianbarahona@tsachila.edu.ec)

**Recepción:** 22 de febrero de 2022

**Aceptación:** 12 de marzo de 2022

**Publicación:** 29 de abril de 2022

**Citación/como citar este artículo:** Landy, B. y Barahona, D. (2022). Módulo de instrumentación con interfaz HMI y software libre para prácticas de medición de masa con celda de carga. Ideas y Voces, 2(1), 13-34.



## **Resumen**

El presente trabajo de integración curricular pretende realizar el diseño de un módulo para instrumentación utilizando una celda de carga en conjunto con el transductor HX711 y una placa de arduino uno, los mismos que permitirán la toma de datos dentro del software libre (MyOpenLab), para realizar la práctica y toma de datos se implementó los dispositivos dentro de un prototipo el cual se debe calibrar con un peso neto, esto permitirá trabajar con datos en tiempo real, cabe recalcar que para este trabajo de investigación se deberá extraer información sobre el puente de Wheatstone ya que este es el principio fundamental de funcionamiento de la celda de carga.

## **Palabras clave**

Prototipo, software libre, instrumentación, arduino

## **Abstract**

The present work of curricular integration intends to carry out the design of a module for instrumentation using a load cell in conjunction with the HX711 transducer and an arduino uno board, the same ones that will allow the data collection within the free software (MyOpenLab), for carry out the practice and data collection, the devices were implemented within a prototype which must be calibrated with a net weight, this will allow working with data in real time, it should be noted that for this research work information on the bridge must be extracted Wheatstone as this is the fundamental operating principle of the load cell.

## **Keywords**

Prototipo, free software, instrumentation, arduino

## **Introducción**

La instrumentación industrial se encuentra actualmente en la cúspide del campo de la automatización y evolucionando a tal punto que los datos emitidos se pueden almacenar en una base de datos, haciendo más fácil realizar un análisis y tomar decisiones con mayor exactitud. La implementación de módulos en el área de instrumentación permite adquirir el conocimiento necesario sobre el uso de instrumentos, los mismos que pueden ser utilizados para medir diferentes variables, como (temperatura, caudal, presión, masa, etc.).

El poder utilizar tecnología de punta de código abierto, es una gran posibilidad que acerca a todos (estudiantes o empíricos) a una puesta a punto de procesos que conllevan la interacción hombre – máquina, haciendo de cada persona, con un poco de conocimiento y curiosidad un potencial solucionador de problemas en su emprendimiento o en automatización de procesos semi industriales.

## **Metodología**

Para alcanzar los objetivos propuestos se creó un diseño metodológico basado en estos elementos de vital importancia para el desarrollo de cualquier proceso investigativo:

Como meta principal se planteó:

Diseñar un módulo para prácticas de instrumentación aplicando celda de carga con simulación en pantalla HMI con software libre.

Otros pasos más específicos para cumplir el objetivo fueron:

- Analizar el funcionamiento del módulo HX711 de arduino en conjunto con la celda de carga YXB183-5K
- Elaborar un prototipo para la toma de datos con referencia a la masa
- Realizar una práctica mediante el software libre MyOpenLab



El enfoque investigativo aplicado fue el cualitativo ya que considera que la realidad se modifica constantemente, y que el investigador, al interpretar la realidad, obtendrá resultados subjetivos. (Bryman, 2004:20).

El tipo de investigación fue la aplicada porque se pretende dar solución al problema que se plantea de forma inmediata en tiempo real obteniendo datos concretos. Según Behar (2008) la investigación aplicada realiza el estudio de problema en circunstancias y características concretas, siendo aplicada a la brevedad para obtener resultados inmediatos.

## Resultados

### Objetivo 1: Análisis hx711 y yxb183-5k

Tabla 1 Tipos de celda de carga.

<b>Tipos de celdas de carga</b>	<b>Norma técnica de homologación</b>	<b>Marca</b>	<b>Tipo</b>	<b>Peso</b>	<b>Material</b>
 Celda de carga de cizallamiento	OIML	SIEMENS	cizallamiento	100kg / 10.000 kg	Acero inoxidable
 Celda de carga de aluminio	OIML NPEP ATEX	METTLE R TOLEDO	extensiométrica	3 kg / 750kg	Aluminio








 <p>Celda de carga de flexión</p>	OIML	Minebea intec	Flexión	10kg/200kg	Acero inoxidable
 <p>Celda de carga de tracción</p>	OIML	Minebea intec	S	60kg/5000kg	Acero inoxidable
 <p>Celda de carga de dobles cizallamiento</p>	OIML	logic	doble cizallamiento	45kg	Acero inoxidable
 <p>Celda de carga de viga</p>	UNE-EN 60529:2018	YXB183- 5K	Viga	5Kg	Aluminio

Tabla 2 Sensor HX711

sensor HX711	Precisión	Conectividad	Voltaje de operación
	24 bits	Fácil celda de carga transductor dos excitadores y dos de señal	5 DC

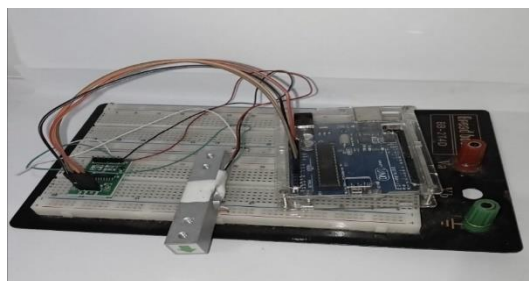
Según el análisis planteado sobre el objetivo uno se presenta como resultado final la selección de la celda de carga de viga en conjunto con el sensor transductor HX711 compatible con Arduino mostrado en la siguiente tabla.

Tabla 3 Selección

Celda de carga de viga YXB183-5K	Sensor de Arduino HX711
	
Celda de carga de viga YXB183-5K.	Transductor HX711
4 cables que conforma el puente de wheatstone	Conexión fácil
Protección media de las galgas	Alta precisión 24 bits
Material de construcción aluminio	protocolo de conexión tipo serial
Norma regente: UNE 60529:2018	mediante 2 pines (Clock y Data).
Largo 8.5 cm	Frecuencia de lectura: 80 Hz.
Ancho 1.3cm	Medidas 4.5 cm de largo
Precio bajo	Ancho 3cm
Instalación fácil	Precio bajo
	Conexión fácil

## Objetivo dos: preensamblado del prototipo para pesado

Como se puede observar en la figura X la conexión de Arduino en conjunto con la celda de carga y el sensor HX711 de Arduino se la realiza primariamente en una protoboard como circuito primario.



*Figura 1 Preensamblado para comprobar el funcionamiento de los dispositivos.*

La conexión de la celda de carga y el sensor hx711 se la realiza de la siguiente manera:

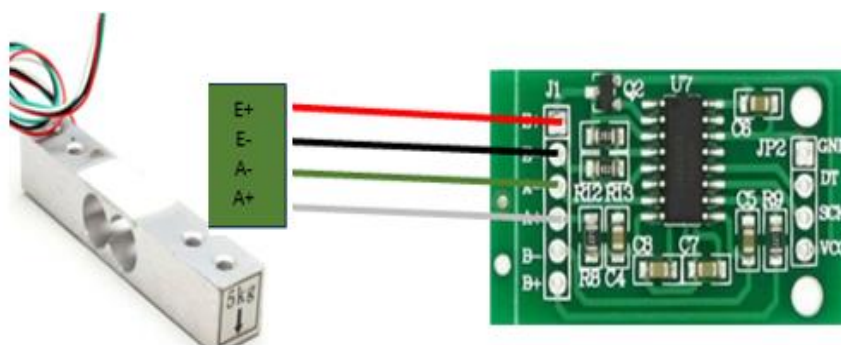
Primer paso: Para conectar la celda de carga con el sensor HX711 de Arduino se utiliza cables de conexión utilizando los siguientes pines;

E+: Se conecta el cable rojo (positivo) perteneciente a la primera galga extensiométrica.

E-: Se conecta el cable negro (negativo) perteneciente a la segunda galga extensiométrica

A+: Se conecta el cable blanco (gris) se utiliza este color ya que el color blanco no se nota. Pero generalmente en una celda de carga viene con un color blanco.

A-: Se conecta el cable de color verde perteneciente a la segunda salida



*Figura 2 Conexión de la celda de carga con el sensor HX711*

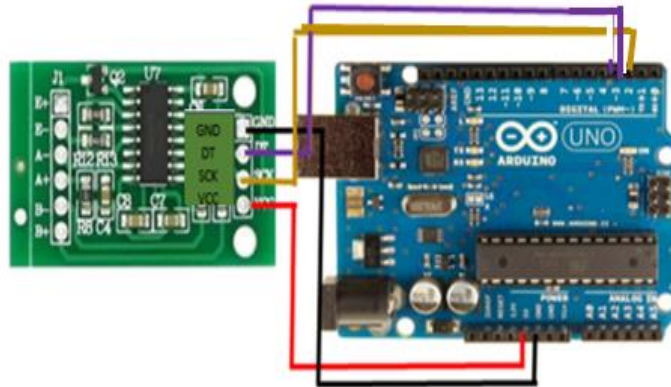
Paso dos: conexión del sensor HX711 en conjunto con la placa de Arduino uno

Gnd: se conecta el cable negro desde el sensor HX711 al pin GND del Arduino uno

DT: data emite la señal digital

SCK: clock emite la señal digital

VCC: se conecta el cable rojo desde el sensor hasta el pin 5V de Arduino



*Figura 3 Conexión de salidas digitales con el arduino.*

Tercer paso: se conecta el Arduino al pc mediante el cable USB (Universal Serial Bus) que ya viene predeterminado desde la empresa, al conectar la placa Arduino uno al computador se procede a cargar la programación para la toma de datos preliminar dentro del programa de Arduino.

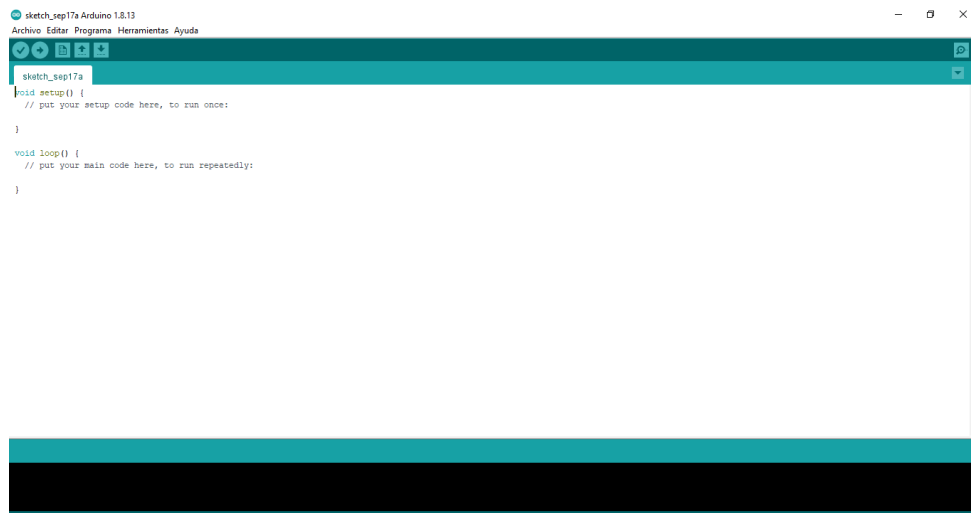


*Figura 4 Arduino conectado a un PC.*

*Obtenido de (Geek Factory. 2017)*

Paso cuatro: abre en la computadora el programa de Arduino que se debe descargar previamente si no aún no se cuenta con el software de Arduino debes ingresar a la

siguiente página web <https://www.arduino.cc/en/Guide> descargar e instalar en el pc, una vez listo el software abre Arduino te aparecerá la siguiente interfaz



*Figura 5 Pantalla inicial de arduino*

Se debe eliminar todo lo que aparece en la zona de trabajo, una vez limpia copiar y pegar el siguiente código:

```
#include <HX711.h>
```

```
// Pin de datos y de reloj
```

```
byte pinData = 3;
```

```
byte pinClk = 2;
```

```
HX711 bascula;
```

```
// Parámetro para calibrar el peso y el sensor
```

```
float factor_calibracion = 20780.0; //Este valor del factor de calibración funciona para mi. El tuyo probablemente será diferente.
```

```
void setup() {
```

```
  Serial.begin(9600);
```

```
  //Serial.println("HX711 programa de calibracion");
```

```
  //Serial.println("Quita cualquier peso de la bascula");
```

```

//Serial.println("Una vez empiece a mostrar informacion de medidas, coloca un peso
conocido encima de la bascula");

//Serial.println("Presiona + para incrementar el factor de calibracion");

//Serial.println("Presiona - para disminuir el factor de calibracion");

// Iniciar sensor

bascula.begin(pinData, pinClk);

// Aplicar la calibración

bascula.set_scale();

// Iniciar la tara

// No tiene que haber nada sobre el peso

bascula.tare();

// Obtener una lectura de referencia

long zero_factor = bascula.read_average();

// Mostrar la primera desviación

//Serial.print("Zero factor: ");

//Serial.println(zero_factor);

}

void loop() {

// Aplicar calibración

bascula.set_scale(factor_calibracion);

// Mostrar la información para ajustar el factor de calibración

//Serial.print("Leyendo: ");

Serial.println(bascula.get_units(), 1);

//Serial.print(" kgs");

//Serial.print(" factor_calibracion: ");

```

```

//Serial.print(factor_calibracion);

Serial.println();

delay(500);

// Obtener información desde el monitor serie

/*if (Serial.available())
{
    char temp = Serial.read();

    if (temp == '+')

        factor_calibracion += 10;

    else if (temp == '-')

        factor_calibracion -= 10;

}*/
}

```

Una vez cargado el código de Arduino la pantalla se presentará de la siguiente forma

```

CELDA_DE_CARGA Arduino 1.8.13
Archivo Editar Programa Herramientas Ayuda

CELDA_DE_CARGA
#include <HX711.h>

// Pin de datos y de reloj
byte pinData = 3;
byte pinCLK = 2;

HX711 bascula;

// Parámetro para calibrar el peso y el sensor
float factor_calibracion = 20780.0; //Este valor del factor de calibración funciona para mí. El tuyo probablemente será diferente.

void setup() {
    Serial.begin(9600);
    //Serial.println("HX711 programa de calibracion");
    //Serial.println("Quita cualquier peso de la bascula");
    //Serial.println("Una vez empiece a mostrar información de medidas, coloque un peso conocido encima de la bascula");
    //Serial.println("Presiona + para incrementar el factor de calibracion");
    //Serial.println("Presiona - para disminuir el factor de calibracion");

    // Iniciar sensor
    bascula.begin(pinData, pinCLK);

    // Aplicar la calibración
    bascula.set_scale(1);
    // Iniciar la tara
    // No tiene que haber nada sobre el peso
    bascula.tare();

    // Obtener una lectura de referencia

```

Figura 6 Primer código utilizado para verificar el funcionamiento del prototipo.

Se debe verificar los siguientes pasos antes de comprobar el código:

1. Verificar el puerto al cual está conectado Arduino
2. Verificar el código

### 3. Abrir el monitor serial



Figura 7 Pasos para trabajar con arduino

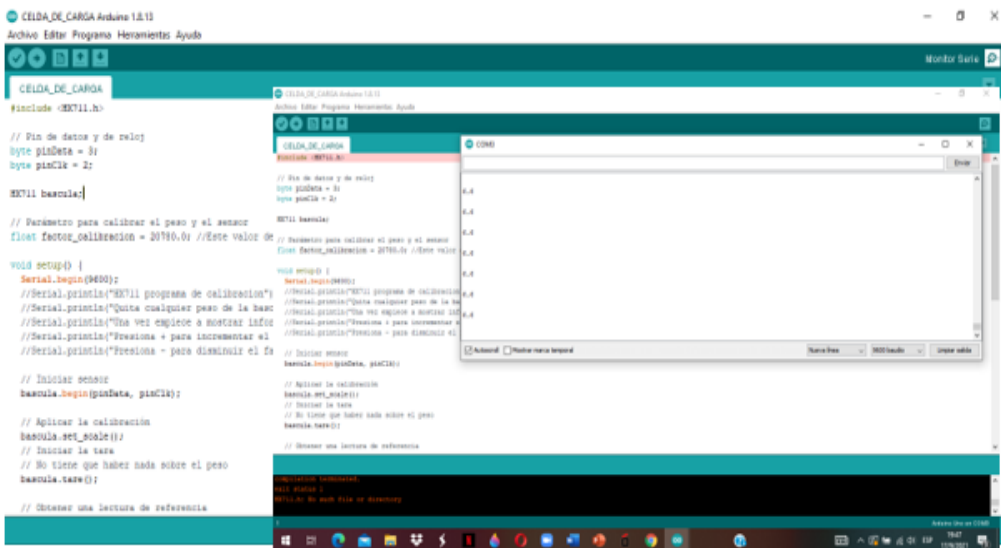


Figura 8 Primera toma de datos con el dispositivo pre ensamblado.

### Objetivo tres: practica

Una vez terminada las pruebas en la pre-implementación del prototipo se procedió a implementar el prototipo final como se ve a continuación, en la figura 20 se observa la conexión de la tarjeta arduino con el transductor HX711.

Como se observa en la figura 8 se realizó la conexión de la celda de carga con el transductor HX711



*Figura 9 Conexión de transductor HX711 con la celda de carga y arduino*

La celda de carga se mantiene suspendida 2cm de la base y 2 cm del plato de pesado, así mismo el Transductor HX711 se encuentra ubicado en una esquina de lado izquierdo, este dispositivo se encuentra aislado por una placa de plástico pues la base que se utilizó para estabilizar la celda de carga es de aluminio, tómesese en cuenta que la celda de carga se encuentra sujeta a la base de una extremo y el otro extremo se encuentra sosteniendo el disco de pesado, se realiza de esta forma ya que al momento de poner la masa esta provoque una tensión que activa las galgas extensiométricas que se encuentran ubicadas en los dos lados de la celda, estas galgas a su vez emiten una señal análoga que forma el puente YXB183-5K. que es recibida por el módulo HX711 el cual se encarga de interpretar dicha señal y retransmitir a la placa de Arduino como una señal digital.



*Figura 10 Celda de carga totalmente implementada*

Se procedió a realizar la calibración de la báscula, utilizando para esto siguiente código de Arduino y un peso neto (Peso de la materia) estable

```
#include "HX711.h"

byte pinData = 3;
byte pinClk = 2;
HX711 balanza;

void setup() {
  Serial.begin(9600);
  balanza.begin(pinData, pinClk);
  Serial.print("Lectura del valor del ADC:t");
  Serial.println(balanza.read());
  Serial.println("No ponga ningún objeto sobre la balanza");
  Serial.println("Destarando...");
  balanza.set_scale(); //La escala por defecto es 1
  balanza.tare(20); //El peso actual es considerado Tara.
  Serial.println("Coloque un peso conocido:");
}

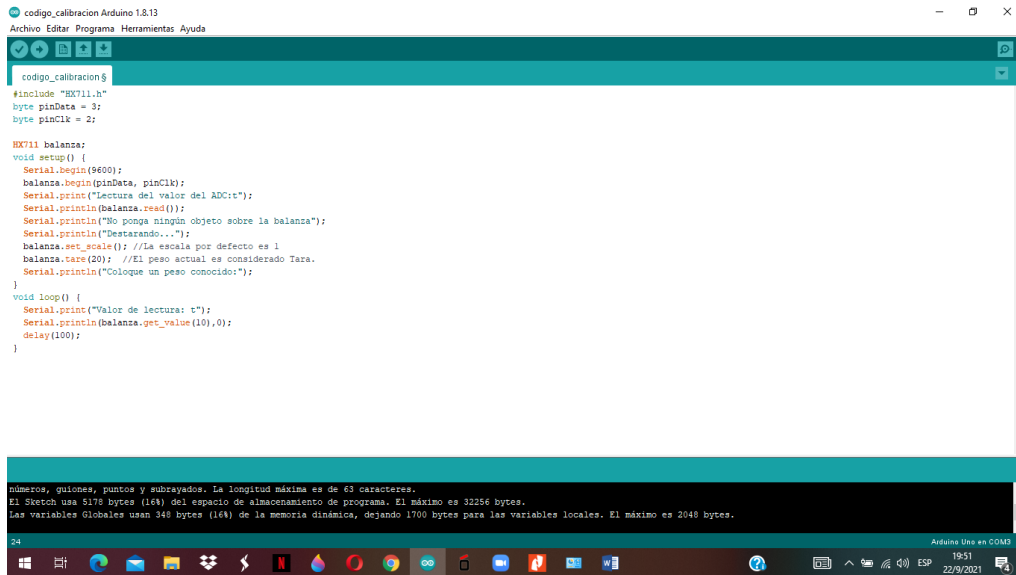
void loop() {
  Serial.print("Valor de lectura: t");
```

```
Serial.println(balanza.get_value(10),0);
```

```
delay(100);
```

```
}
```

Una vez ingresado el código de arduino dentro del software se procede a verificar y a subir el sketch, realizado esto se debe abrir el puerto serial de arduino



```
codigo_calibracion Arduino 1.8.13
Archivo Editar Programa Herramientas Ayuda

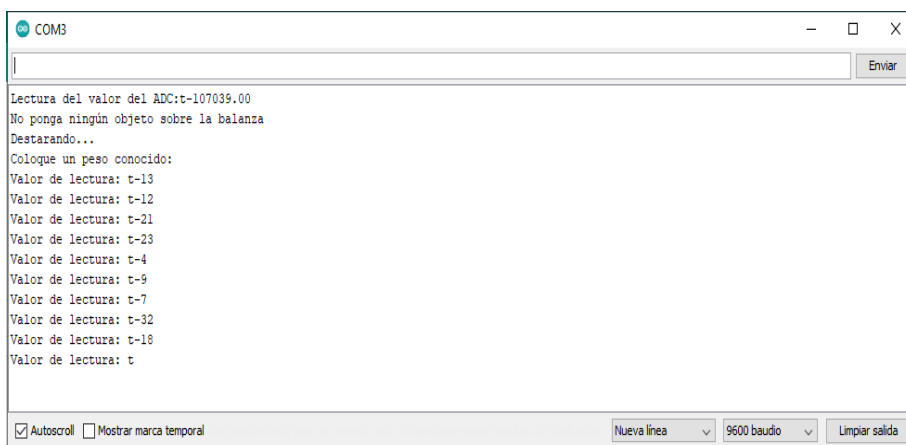
codigo_calibracion $
#include "HX711.h"
byte pinData = 3;
byte pinClk = 2;

HX711 balanza;
void setup() {
  Serial.begin(9600);
  balanza.begin(pinData, pinClk);
  Serial.print("Lectura del valor del ADC:t");
  Serial.println(balanza.read());
  Serial.println("No ponga ningún objeto sobre la balanza");
  Serial.println("Destarando...");
  balanza.set_scale(); //La escala por defecto es 1
  balanza.tare(20); //El peso actual es considerado Tara.
  Serial.println("Coloque un peso conocido:");
}
void loop() {
  Serial.print("Valor de lectura: t");
  Serial.println(balanza.get_value(10),0);
  delay(100);
}

números, guiones, puntos y subrayados. La longitud máxima es de 63 caracteres.
El Sketch usa 5178 bytes (16%) del espacio de almacenamiento de programa. El máximo es 32256 bytes.
Las variables Globales usan 348 bytes (16%) de la memoria dinámica, dejando 1700 bytes para las variables locales. El máximo es 2048 bytes.
```

Figura 11 Código de calibración.

Donde aparecerá el siguiente mensaje se debe dejar durante unos segundos mientras se estabiliza el serial.



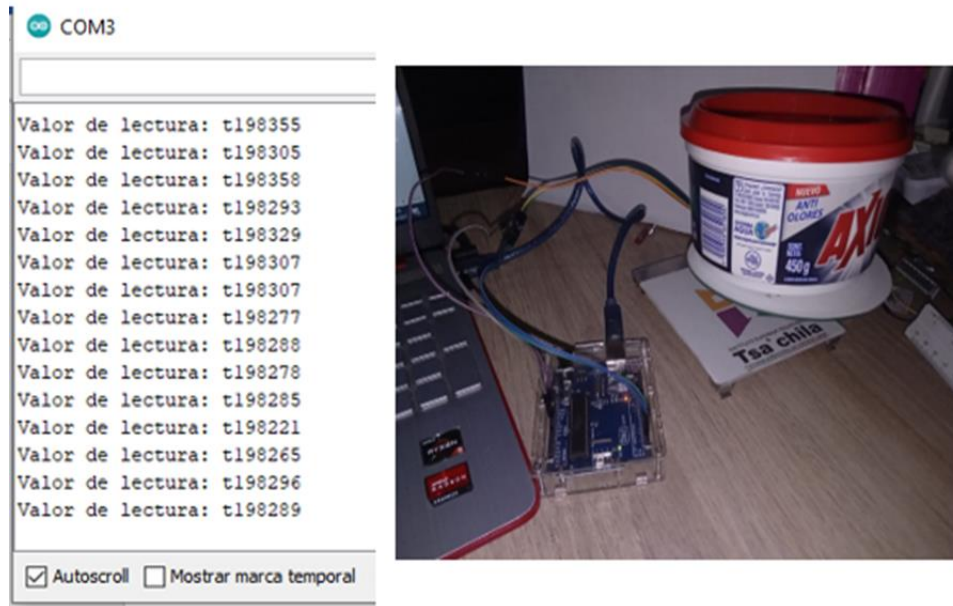
```
COM3
Enviar

Lectura del valor del ADC:t-107039.00
No ponga ningún objeto sobre la balanza
Destarando...
Coloque un peso conocido:
Valor de lectura: t-13
Valor de lectura: t-12
Valor de lectura: t-21
Valor de lectura: t-23
Valor de lectura: t-4
Valor de lectura: t-9
Valor de lectura: t-7
Valor de lectura: t-32
Valor de lectura: t-18
Valor de lectura: t

 Autoscroll  Mostrar marca temporal Nueva línea 9600 baudio Limpiar salida
```

Figura 12 Datos primarios al inicio de la calibración

Una vez estable se procede a colocar el peso neto sobre el plato de pesado, este provocara una variación en los valores que emite el puerto serial como se puede observar a continuación.



*Figura 13 Colocación del peso neto para calibrar, obteniendo los primeros datos con sumas altas.*

Para proceder a calibrar la balanza se debe tomar una de las cifras que se encuentran en el serial en este caso se tomara 198133 y se precede a realizar el cálculo utilizando la siguiente formula.

$$Escala = \frac{\text{valor de lectura}}{\text{peso real}}$$

$$Escala = \frac{198133}{450} = 440.25$$

El total de la división entre el valor de lectura que emite el serial se divide al peso real que ponemos sobre la celda de carga siendo este 440.25, este valor se ingresa al código final que será utilizado para la toma de datos en tiempo real.

#include <HX711.h>

```

byte pinData = 3;

byte pinClk = 2;

HX711 bascula;

float factor_calibracion = 440.25;

void setup() {

  Serial.begin(9600);

  bascula.begin(pinData, pinClk);

  bascula.tare(20);

  long zero_factor = bascula.read_average();

}

void loop() {

  bascula.set_scale(factor_calibracion);

  Serial.println(bascula.get_units(), 1);

  Serial.println();

  delay(500);

}

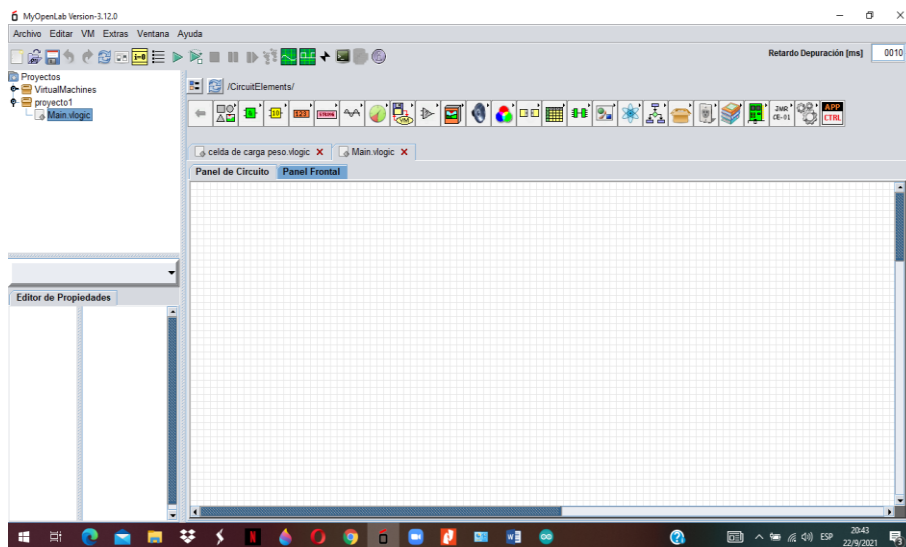
```

Ya que se ingresa el valor al código final de Arduino en la línea ( float factor\_calibracion = 440.25; ) se procedió a cargar nuevamente el sketch y a reabrir el serial de Arduino para comprobar que la calibración ha sido correcta, como se observa en la figura 25 algo muy importante es que cada vez que se realice una modificación dentro del código se debe verificar y subir el código (sketch).



*Figura 14 Primera toma de datos reales.*

Una vez comprobada la calibración los valores del peso son reales y que se puede trabajar con este código se debe abrir el software libre MyOpenLab, dentro de esta interfaz se simulara la pantalla HMI y donde se tomaran los datos que emite la celda de carga.



*Figura 15 Primera interfaz de MyOpenLab.*

Cuando se abre MyOpenLab se observará una interfaz en blanco para poder comunicar la celda de carga con el arduino y recibir los datos que emiten los dispositivos antes detallados se debe utilizar la siguiente programación.

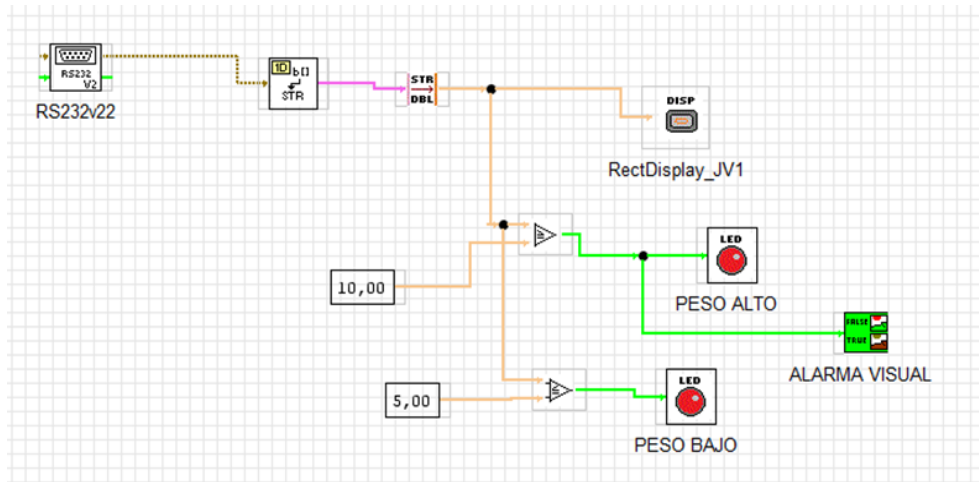


Figura 165 Circuito base para conectar MyOpenLab con arduino.


Para dar inicio a la toma de dato se debe verificar que el puerto de enlace del rs232v22 este correctamente conectado con el puerto de arduino una vez que se a enlazado correctamente se procede a dar clic en  he inmediatamente se desplegara la simulación de la pantalla HMI se debe esperar un poco para poder recibir la señal.



Figura 17 Pantalla de inicio para pesado.

Se procedió a colocar el objeto a pesar conociendo de antemano ya su peso marcado en el mismo, en la figura 17 se observa el peso del objeto igual al peso marcado en la etiqueta.

## Conclusiones

En el desarrollo de la investigación se evidencia la construcción del módulo de prácticas de instrumentación, destinado a la aplicación de una celda de carga simulada en una pantalla HMI utilizando software libre. Con este propósito, se analizó el funcionamiento del módulo HX711 de arduino para incorporar así la parte electrónica en este proceso.

Al utilizar el módulo YXB183-5K como celda de carga se complementó todo el hardware necesario para realizar esta simulación y sobre todo permitir evidenciar un proceso de manera didáctica para las futuras generaciones que pasen por las aulas del ISTT.

La construcción del prototipo fue un poco compleja ya que no estaban claramente marcada la forma de conexión, una vez salvada esta situación todo se desarrolló con normalidad llegando a cumplir los objetivos planteados.

## **Bibliografía**

21, A. (2021). Centro de formación técnica para la industria. Obtenido de <https://www.cursosaula21.com/que-es-un-sistema-scada/>

Alzate, O. F. (01 de mayo de 2019). Código Electronica . Obtenido de <http://codigoelectronica.com/blog/modulo-hx711>

CONTAVAL. (2017). Células de carga (teoría básica). Obtenido de <https://www.contaval.es/celulas-carga-teoria/>

electronics, M. (2021). Arduino . Obtenido de <https://arduino.cl/que-es-arduino/>

Fernandez, F. (2015). Buenas Practicas de Medición Balanzas . Messen, <https://es.calameo.com/read/001137461b5732b16827c>.

Gutierrez, P. M., & Iturralde, M. S. (2017). Fundamentos basicos de instrumentación y control. Obtenido de

<https://www.fnmt.es/documents/10179/10666378/Fundamentos+básico+de+instrumentación+y+control.pdf/df746edc-8bd8-2191-2218-4acf36957671>

HOY, C. (2021). Las tres distribuciones de Linux más seguras. Obtenido de <https://computerhoy.com/listas/software/tres-distribuciones-linux-mas-seguras-51680>

inc, O. E. (2021). Introducción a las celdas de carga. Obtenido de <https://mx.omega.com/prodinfo/celdas-de-carga.html>

INTELIGENTE, C. (2020). Pantallas Touch HMI. Obtenido de <https://www.controlinteligente.com.mx/hmi.php>

Java. (2021). Java . Obtenido de <https://www.java.com/es/about/>

Logo, S. (2021). Introduccion a las celdas de carga. Obtenido de <https://mx.omega.com/prodinfo/celdas-de-carga.html>

Mechatronics, N. (2021). TUTORIAL TRASMISOR DE CELDA DE CARGA HX711, BALANZA DIGITAL. Obtenido de [https://naylampmechatronics.com/blog/25\\_tutorial-trasmisor-de-celda-de-carga-hx711-balanza-digital.html](https://naylampmechatronics.com/blog/25_tutorial-trasmisor-de-celda-de-carga-hx711-balanza-digital.html)

Normalización, A. E. (abril de 2018). grados de protección proporcionados por las envolventes (codigo IP). Obtenido de [file:///C:/Users/Usuario/Downloads/EXT\\_KV5jwo5fSSsAzagxSaJP.pdf](file:///C:/Users/Usuario/Downloads/EXT_KV5jwo5fSSsAzagxSaJP.pdf)

Oliver, L. M. (20 de julio de 2017). Diseño y Simulación con MyOpenLab, la alternativa gratuita al Labview. Obtenido de <https://dinglobal.ning.com/profiles/blogs/dise-o-y-simulaci-n-con-myopenlab-la-alternativa-gratuita-al>

ORG, C. (2021). balanza romana . Obtenido de <https://www.cultura10.org/romana/balanza/>

org, s. L. (2021). ¿Que es Windows? Obtenido de <https://softwarelab.org/es/windows-historia/>

S.A., P. N. (2018). MC tpnet . Obtenido de <https://www.muycomputer.com/2021/03/25/mac-os-x-cumple-20-anos/>

S.A.S, E. y. (2021). Historia de la balanza. Obtenido de <https://www.equiposylaboratorio.com/portal/articulo-ampliado/historia-de-la-balanza>

Spectris. (2021 ). omega . Obtenido de Fundamentos de las células de carga: <https://es.omega.com/prodinfo/celulas-de-carga.html>

technology, D. (2008/2021). Nitropc. Obtenido de <https://www.nitro-pc.es/blog/como-tener-la-interfaz-de-windows-7-en-windows-10/>

Tecnologico, C. (2014/2021). TU PROPIO LABORATORIO ELECTRÓNICO DE SIMULACIÓN, ADQUISICIÓN Y CONTROL DE DATOS CON MYOPENLAB. Obtenido de <https://campustecnologicovirtual.es/blogs/tu-propio-laboratorio-electronico-de-simulacion-adquisicion-y-control-de-datos-con-myopenlab-25>

TecnoTrono. (2020). Hardware, videojuegos y programas . Obtenido de <https://tecnotrono.com/software/sistemas-operativos/257/>

TUPUNATRON. (1 de septiembre de 2021). TOTAL AUTOMATIZACION . Obtenido de <https://tupunatron.com/producto/dvx-c-celda-de-carga-digital/>

UtilCell. (s.f.). CIRCUITO ELECTRICO DE UNA CÉLULA DE CARGA. Obtenido de [http://www.utilcell.com.cn/wp-content/uploads/2017/05/Es\\_NT\\_Circuito-Electrico-de-una-C%C3%A9lula-de-Carga\\_Utilcell.pdf](http://www.utilcell.com.cn/wp-content/uploads/2017/05/Es_NT_Circuito-Electrico-de-una-C%C3%A9lula-de-Carga_Utilcell.pdf)

Vasquez Fernandez Pacheco, A. S., Ramos de la Flor, F., Fernández Rodríguez, R., Payo Gutierrez, I., & Adan Óliver, A. (2015). Robótica Educativa. Madrid: Grupo Editorial RA-MA.

Ventures. (septiembre de 2021). SoftwareLab.org. Obtenido de <https://softwarelab.org/es/windows-historia/>