



# Sensores Ultrasónicos y Arduino en Experiencias de Laboratorio de Cinemática

## Ultrasonic Sensors and Arduino in Kinematics Lab Experiences

F. BARRALAGA<sup>1</sup>

Recibido: 15 de octubre de 2020 / Aceptado: 18 de diciembre de 2020

<sup>1</sup>Escuela de Física, Facultad de Ciencias,  
Universidad Nacional Autónoma de Honduras  
email: francisco.barralaga@unah.edu.hn

### RESUMEN

Arduino es una plataforma de control automático muy popular, de fácil y rápido uso, con una amplia gama de periféricos disponibles en un mercado creciente que ha permitido que su costo sean relativamente bajo y que se produzca gran cantidad de material didáctico disponible gratuitamente en la Web. Se desarrolló un sensor ultrasónico con el que la medición de la distancia a un obstáculo es inmediata y aceptablemente precisa. Este sensor es controlado a través de una pequeña tarjeta Arduino. La aplicación desarrollada registra la distancia del obstáculo más cercano al sensor y el tiempo de detección. Con esta información la aplicación calcula la rapidez a la que el obstáculo se mueve respecto del sensor. Los datos son opcionalmente enviados a una pantalla para su monitoreo visual inmediato, o se crea un archivo en un formato que permite el procesamiento posterior con sistemas de álgebra simbólica o con alguna hoja electrónica.

### ABSTRACT

Arduino is a popular, easy-to-use, automatic control platform with a wide range of peripherals in a growing market that gives it a relatively low cost, which has allowed the development of a large amount of free educational support material able on the Web. An ultrasonic sensor was developed with which the measurement of the distance to an obstacle is immediate and acceptably accurate. This sensor is controlled through a small Arduino board. The application developed records the distance of the obstacle closest to the sensor and the detection time. With this information the application calculates the speed at which the obstacle moves relative to the sensor. The data is optionally sent to a screen for immediate visual monitoring, or a file is created in a format that can be opened and processed from symbolic algebraical systems or any Spreadsheet.

### PALABRAS CLAVES

Control Automático, Arduino, Sensores Ultrasónicos, Experiencia de Física

### KEYWORDS

Automatic Control, Arduino, Ultrasonic Sensors, Physics Experience

\* Esta obra está bajo una licencia Creative Commons Reconocimiento - NoComercial 4.0 Internacional

\* This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

## I | JUSTIFICACIÓN

La implementación de laboratorios de cinemática requiere contar con dispositivos para la medición y el registro de distancias e intervalos de tiempo cortos. El análisis del movimiento de un cuerpo es una temática obligada en todos los cursos de Física de cualquier plan de estudios de ingeniería y contar con equipo capaz de generar datos de parámetros del movimiento con un mínimo de precisión es imprescindible y en la mayoría de los casos implica una inversión relativamente alta.

El desarrollo de la capacidad de diseño e implementación de dispositivos que operen con tecnología de automatización, es muy importante y se justifica, si los resultados de las mediciones son aceptablemente precisos para la realización de experiencias de laboratorio de nivel superior, y si los costos son apreciablemente menores que los de equipos adquiridos a empresas especializadas de nivel internacional.

## II | INTRODUCCIÓN

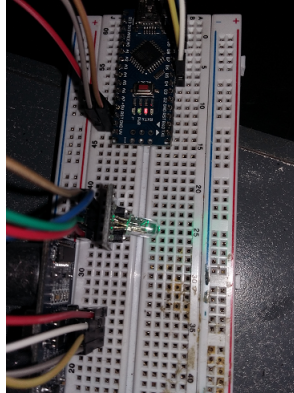
Desde hace varios años, se ha reportado el uso de la plataforma Arduino en la implementación de opciones de control automático a nivel superior en diferentes áreas de ciencias físicas e ingeniería. Estas aplicaciones incluyen la utilización de sensores de señales físicas conectados a tarjetas de control Arduino, para la toma de datos en diversas experiencias de laboratorio de ciencias físicas (Moore, 30 de Julio de 2014). En estas experiencias, los estudiantes fueron auxiliados de una de estas tarjetas controladoras para construir un “*datalogger*” para extraer los datos del estudio de un fenómeno físico en laboratorio, que posteriormente fueron numéricamente analizados y procesados para generar modelos matemáticos usando hojas electrónicas o sistemas de álgebra simbólica.

Arduino, es una plataforma de reciente creación, utilizada para diseñar modelos o prototipos de dispositivos electrónicos, consistente en una tarjeta de control basado en un microchip Atmel de 8 bits. Dispone de control sobre dispositivos de entrada, usualmente sensores, y de actuadores conectados a sus salidas, que responden con un comportamiento que es especificado por el diseñador a través de código de programación generado en un ambiente basado en C++.

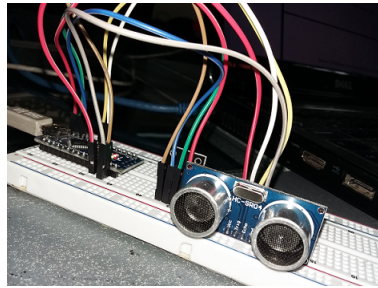
Arduino fue ideada por dos físicos computacionales italianos, David Cuartielles y Massimo Banzi y salió a luz por primera vez en 2005; desde entonces ha crecido enormemente en popularidad, en disposición de material de desarrollo gratuito y de sensores de todo tipo, a la vez que su costo se ha reducido, mejorando la factibilidad de su uso en la implementación de aplicaciones para laboratorios a nivel superior. Como un ejemplo más, el Departamento de Ingeniería Eléctrica de la Universidad Bogor Ibn Khaldun de Indonesia (M. Hariansyah, 2014), ha hecho uso de los modelos AtMega 2560 de Arduino, los más robustos hasta ahora, para la medición de profundidades de drenajes en procesos de inspección.

## III | DESARROLLO

Un sensor ultrasónico HC-SR04 fue utilizado para detectar la distancia del mismo a un obstáculo. El sensor envía un haz de ondas de ultrasonido de 10microsegundos de duración y detecta el haz reflejado en los obstáculos frente al sensor. El sensor envía a la pequeña tarjeta controladora Arduino Nano, el intervalo de tiempo entre el disparo del haz y la recepción de su reflejo.



*Figura 1: Detalle de montaje del sensor. Puede verse en la parte superior de la fotografía a la tarjeta controladora Arduino Nano hacia el centro al LED indicador de cercanía de obstáculo.*



*Figura 2: Sensor ultrasónico HC-SR04. Tiene un alcance máximo detección de 4 m y un mínimo de 2 cm.*

La figura [1] muestra el detalle del montaje del prototipo. Se incluye un LED RGB para notificar visualmente cuando el obstáculo frente al detector está fuera del rango de detección del mismo.

La Universidad de Los Andes, UNIANDES de Colombia, ha explorado y recomendado el uso de las hojas de cálculo como herramienta de soporte para el desarrollo de opciones didácticas en las ciencias (Alvarez, 1996). Aunque existen herramientas más sofisticadas y precisas como Wolfram Mathematica y Matlab, ciertamente estas requieren la compra de licencias para su uso. Por otra parte se cuentan con otras opciones gratuitas y de código abierto, de uso muy extendido entre los estudiantes de las carreras del área físico-matemática, que significan soluciones más expeditas para realizar el análisis de los datos generados en experiencias de laboratorio de física a nivel de pregrado.

Los datos colectados por la tarjeta controladora, son enviados al puerto serial al que la tarjeta está conectada. Desde este puerto, los datos son tomados y vertidos en una hoja electrónica para ser procesados gráfica y estadísticamente. Se hizo uso de *Gobetwino versión 0.5*, un software libre que registra los datos en tiempo real directamente del puerto serial, cuando la tarjeta Nano está recibiendo la señal de salida del sensor ultrasónico.

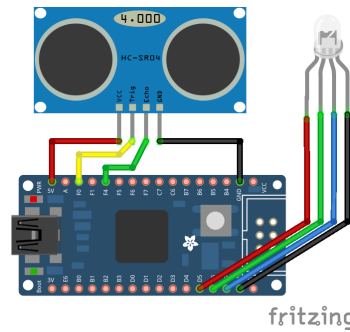


Figura 3: Esquema de montaje de dispositivo. La imagen fue generada con Fritzing

Elemento	Costo (Lps)
Sensor Ultrasónico HC-SR04	\$0.77
Tarjeta Arduino Nano	\$5.00
LED RGB (opcional)	\$0.25
Cable Micro USB	\$0.75

Tabla 1: Insumos necesarios y costo.

## 1 | Código de Control

Arduino cuenta con un ambiente de programación basado en C++. El programa de control de la tarjeta controladora Nano, se realiza desde el editor provisto por Arduino. En el programa de control se incluye código para calcular la velocidad del obstáculo y enviar al puerto serial la corriente de datos generada. El código de control es grabado en la memoria ROM del microcontrolador de la tarjeta. A partir de entonces, el proceso de control lo realiza la tarjeta sin requerir la interacción directa de la computadora; estos sistemas son conocidos como *embedded* por formar parte del dispositivo diseñado y funcionar en forma autónoma. El código necesario para el control del sensor de distancia a través de la tarjeta Nano aparece en la sección VI (Apéndice).

### 1.1 | Esquema de Montaje

El montaje necesario es muy simple. La Figura [3] muestra el detalle.

## 2 | Insumos Necesarios y Costo

La implementación del dispositivo requiere los elementos presentados en la tabla [1].

## IV | RESULTADOS OBTENIDOS

Las figuras [4] y [5] muestran un detalle de los resultados obtenidos.

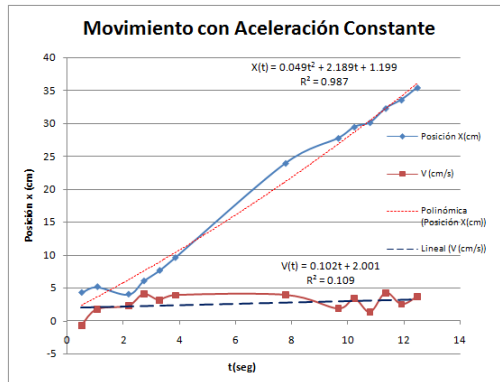


Figura 4: Despliegue gráfico de la distancias sucesivas entre un obstáculo al sensor ultrasónico

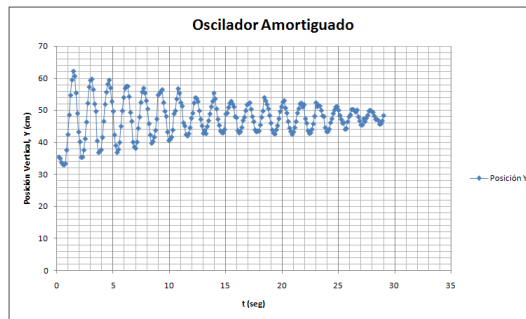


Figura 5: Gráfica de posición vertical de un sensor ultrasónico en movimiento oscilatorio amortiguado. Los datos fueron tomados con una tarjeta controladora Arduino Nano

## V | CONCLUSIONES

1. Es posible construir, con relativa facilidad, sensores para actividades experimentales de nivel superior para tratar tópicos de cinemática de traslación.
2. Arduino es una plataforma adecuada para el desarrollo y diseño de dispositivos electrónicos de control automático, que pueden ser necesitados en experiencias de laboratorio de cinemática de nivel superior.
3. El costo del dispositivo es muy bajo, comparado con el que implica la disponibilidad de este tipo de sistemas de adquisición de datos especializados, tradicionalmente utilizados en laboratorios de sistemas de pregrado.
4. Los datos obtenidos con este dispositivo pueden ser procesados numéricamente con una hoja electrónica popular o de código abierto, de uso extendido, aunque el procesamiento numérico con Matlab o Wolfram Mathematica puede ser deseable si se dispone de las licencias correspondientes.
5. Los resultados de precisión y bajo costo mostrados, justifican el desarrollo de la capacidad de uso de esta tecnología en unidades académicas de pregrado.

## I REFERENCIAS

- Evans, B. W. (2007). *Arduino programming notebook*. San Francisco, California:.
- Gerald W. Recktenwald, D. E. (2011). *Using arduino as a platform for programming, design and measurement in a freshman engineering*.
- Gutiérrez, J. M. (2007). *Manual de programación de arduino*. creative commons.
- Holmberg, A. H. (2015). *Measuring mechanical vibrations using an arduino as a slave i/o to an epics control system*.
- M. Hariansyah, S. R. (2014). *The application of ultrasonic sensor and atemega 328 arduino to measure the ploughing depth elevation of drainage channel*. (3.ª ed.).
- Mikmo. (2016). *Gobetwino 0.5 user manual*. (3.ª ed.). Descargado de <http://www.mikmo.dk/gobetwino>
- Moore, A. J. (30 de Julio de 2014). *Using arduino to enhance computer programming*. (W.S.University, Ed. ed.).
- Prometec. (2016a). *Instalación del entorno de trabajo de arduino*.
- Prometec. (2016b). *Mi primer programa*. Descargado de <http://www.prometec.net/nuestro-primer-programa/>

## VI | APÉNDICE

El siguiente es el código necesario para el control del sensor de distancia a través de la tarjeta Nano:

```
int pinEco = 6;
int pinTrigger = 5;
int pinLED_Azul = 10;
int pinLED_Verde = 11;
int pinLED_Rojo = 12;
int count = 0;
float t, dt, dist, v, accel;
int mseg = 50;
int RangoMax = 400;
int RangoMin = 2;
int i = 0;
float x[10];
float buffer[10];

void setup() {

  Serial.begin(9600);
  pinMode(pinTrigger, OUTPUT);
  pinMode(pinEco, INPUT);
  pinMode(pinLED_Azul, OUTPUT);
  pinMode(pinLED_Verde, OUTPUT);
  pinMode(pinLED_Rojo, OUTPUT);

  Shot();
  dt = pulseIn(pinEco, HIGH);
  x[0] = dt/58;
  dist = x[0];
  v = 0;

}
```

```
void loop() {
    v = Vprom(x[0]);
    t = millis()/1000.0;

    Serial.print("#SIDATAUSI[");
    Serial.print(String(t) + "," + String(dist) + "," + String(v));
    Serial.println("]#");

    if (dist >= RangoMax || dist <= RangoMin) {
        Serial.println("Fuera de Rango");
        digitalWrite(pinLED_Rojo, HIGH);
        digitalWrite(pinLED_Verde, LOW);
    }

    else {
        digitalWrite(pinLED_Rojo, LOW);
        digitalWrite(pinLED_Verde, HIGH);
    }
    delay(10);
}

void Shot() {
    digitalWrite(pinTrigger, LOW);
    delayMicroseconds(5);
    digitalWrite(pinTrigger, HIGH);
    delayMicroseconds(10);
    digitalWrite(pinTrigger, LOW);
}

float Vprom(float d) {
    float vel;
    vel = 0;
    for (int i = 0; i <=4; i++){
        vel = vel + CalculeV(x[0]);
    }
    vel = vel/5;
    return vel;
}

float CalculeV(float d) {
    float vel;
    float t1;
    float t2;
    t1 = millis();
    Shot();
    dt = pulseIn(pinEco, HIGH);
    x[1] = dt/58;
    dist = x[1];
}
```

```
t2 = millis ();  
vel = 1000*(x[1] - x[0])/(t2-t1);  
x[0] = x[1];  
return vel;  
}
```