



Instituto Politécnico Nacional.

Escuela Superior de Cómputo.



**Proyecto Final**  
**“Sensores con ADC y Arduino”.**

Integrantes:

- Frías Mercado Carlos Elliot – 2016630119.
- Gómez Ramírez Oswaldo – 2016630149.
- Hernández Castro Karla Beatriz – 2016630173.

Grupo:  
3CM4

Fecha de entrega:  
04/12/2018

Profesor:  
Ing. Juan Carlos Téllez Barrera.

# PROYECTO FINAL

## “Sensores con ADC y arduino”

### Introducción:

Arduino es una plataforma de código abierto utilizada para construir proyectos de electrónica. Cada circuito de Arduino consta de una placa de circuito física programable (a menudo denominada microcontrolador) y una pieza de software, o IDE (Entorno de desarrollo integrado) que se ejecuta en la

El hardware y software Arduino fue diseñado para artistas, diseñadores, aficionados, hackers, novatos y cualquier persona interesada en crear objetos o entornos interactivos. Arduino puede interactuar con botones, LED, motores, parlantes, unidades de GPS, cámaras, Internet e incluso su teléfono inteligente o televisor. Esta flexibilidad combinada con el hecho de que el software de Arduino es gratuito, las placas de hardware son bastante baratas y que tanto el software como el hardware son fáciles de aprender, ha llevado a una gran comunidad de usuarios que han contribuido con código y han publicado instrucciones para una gran variedad de Proyectos basados en arduino.

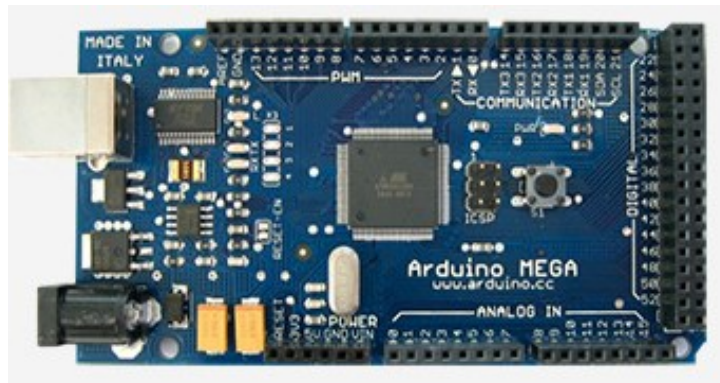


Imagen 1: Placa arduino - Arduino Mega

EL modulo Bluetooth HC-05 compatible con arduino viene configurado de fábrica como esclavo, pero dicha configuración puede cambiarse para que trabaje como maestro, además al igual que el sensor Bluetooth HC-06, se puede cambiar el nombre, código de vinculación velocidad y otros parámetros más.

Modulo Bluetooth HC-05 esclavo:

Cuando está configurado de esta forma, espera que un dispositivo Bluetooth maestro se conecte a este, generalmente se utiliza cuando se necesita comunicarse con una PC o Celular, pues estos se comportan como dispositivos maestros.

Modulo Bluetooth HC-05 Maestro:

En este modo, EL HC-05 es el que inicia la conexión. Un dispositivo maestro solo se puede conectarse con un dispositivo esclavo. Generalmente se utiliza este modo para comunicarse entre módulos Bluetooth, pero es necesario antes especificar con que dispositivo se tiene que comunicar.

El módulo HC-05 viene por defecto configurado de la siguiente forma:

- Modo o role: Esclavo
- Nombre por defecto: HC-05
- Código de emparejamiento por defecto: 1234
- La velocidad por defecto (baud rate): 9600

EL Modulo HC-05 tiene 4 estados:

1. Estado Desconectado:

Entra a este estado tan pronto se alimenta el modulo, y cuando no se ha establecido una conexión Bluetooth con ningún otro dispositivo. EL LED del módulo en este estado parpadea rápidamente

En este estado a diferencia del HC-06, el HC-05 no puede interpretar los comandos AT

2. Estado Conectado o de comunicación

Entra a este estado cuando se establece una conexión con otro dispositivo Bluetooth. El LED hace un doble parpadeo.

Todos los datos que se ingresen al HC-05 por el Pin RX se transmiten por Bluetooth al dispositivo conectado, y los datos recibidos se devuelven por el pin TX.

3. Modo AT 1

Para entrar a este estado después de conectar y alimentar el modulo es necesario presionar el botón del HC-05.

En este estado, es posible enviar comandos AT, pero a la misma velocidad con el que está configurado. EL LED del módulo en este estado parpadea rápidamente igual que en el estado desconectado.

4. Modo AT 2

Para entrar a este estado es necesario tener presionado el botón al momento de alimentar el modulo, es decir el modulo debe encender con el botón presionado, después de haber encendido se puede soltar y permanecerá en este estado.

Para enviar comandos AT es necesario hacerlo a la velocidad de 38400 baudios. EL LED del módulo en este estado parpadea lentamente.

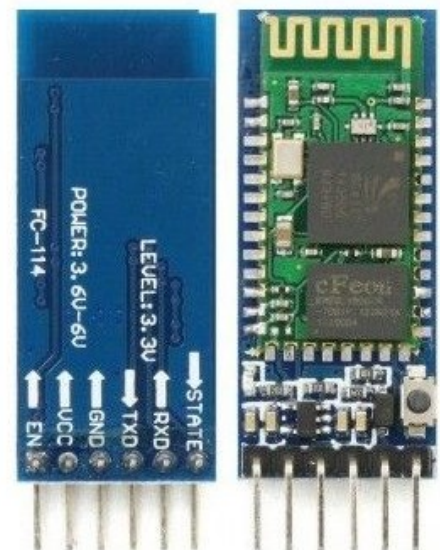


Imagen 2: Módulo Bluetooth HC-05

## Información relevante para el desarrollo de la práctica. Sensor LM35

El LM35 es un sensor de temperatura con una precisión calibrada de 1 °C. Su rango de medición abarca desde -55 °C hasta 150 °C. La salida es lineal y cada grado Celsius equivale a 10 mV, por lo tanto:

150 °C = +1500 mV

25°C = +250 mV

-55 °C = -550 mV

El sensor de temperatura no requiere de circuitos adicionales para calibrarlo externamente. La baja impedancia de salida, su salida lineal y su precisa calibración hace posible que esté integrado sea instalado fácilmente en un circuito de control. Debido a su baja corriente de alimentación se produce un efecto de auto calentamiento muy reducido.

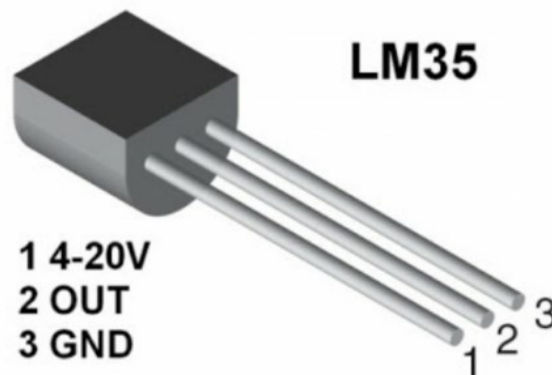


Imagen 3: Configuración de terminales del LM35

El LM35 funciona en el rango de alimentación comprendido entre 4 y 30 voltios.

Especificaciones:

- Resolución: 10mV por cada grado centígrado.
- Tipo de medición. Salida analógica.
- No requiere calibración.
  - Tiene una precisión de  $\pm 1/4^{\circ}\text{C}$ .
  - Esta calibrado para medir  $^{\circ}\text{C}$ .
- Consumo de corriente: 60  $\mu\text{A}$

- Tensión, rango de detección Precisión:  $\pm 0,4 \text{ }^{\circ}\text{C}$
- Lineal. Salida de  $10 \text{ mV}/^{\circ}\text{C}$
- Baja impedancia de salida:  $0.1 \text{ }\Omega$  con carga de  $1 \text{ mA}$
- Temperatura de operación:  $-55 \text{ }^{\circ}\text{C}$  a  $+150 \text{ }^{\circ}\text{C}$

## Sensor resistivo

Un sensor resistivo es aquél que varía una resistencia de acuerdo a cambios físicos en el ambiente, dichos sensores se han clasificado de acuerdo al tipo de variable a medir en los siguientes 5 tipos:

Variable a medir	Sensor resistivo
Mecánica	Potenciómetros y galgas extensométricas.
Térmica	Termorresistencia y termistores.
Magnética	Magneto resistencia.
Óptica	Fotorresistencia.
Química	Higrómetro resistivo.

Tabla 1. Variables físicas

Estos sensores son los más abundantes, debido a la gran cantidad de magnitudes físicas que afectan al valor de la resistencia eléctrica de un material.

Los sensores potenciométricos o potenciómetros son componentes eléctricos cuya resistencia está en función del desplazamiento de un elemento móvil. Consiste en una resistencia fija con un contacto móvil deslizante lineal o giratorio (cursor) que lo divide eléctricamente.

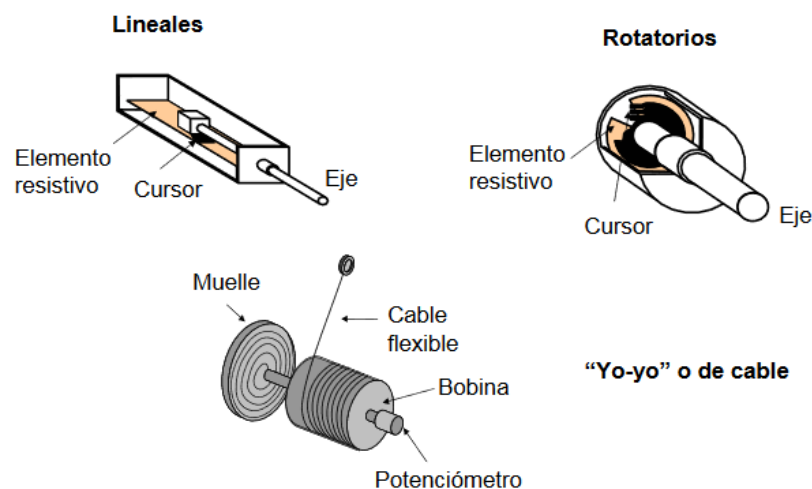


Imagen 4. Tipos de potenciómetros

La aplicación más común del potenciómetro en instrumentación es como sensor de desplazamiento resistivo. El movimiento del cursor origina un cambio en la resistencia medida entre el terminal central y uno cualquiera de los extremos. Este cambio de resistencia puede utilizarse para medir desplazamientos lineales o angulares de una pieza acoplada al cursor.

### Aspectos generales de un convertidor analógico-digital.

Un convertidor analógico digital (ADC) es un dispositivo que convierte, generalmente voltaje (señal analógica), a un valor digital que representa la amplitud de dicha señal. La conversión implica una cuantización de la señal de entrada por lo que se puede producir un error al realizar la conversión.

La resolución del convertidor ADC indica la cantidad de valores discretos que puede producir para representar el rango analógico, un ADC de 8bits puede representar 256 niveles de una señal analógica

Hay principalmente dos pasos involucrados en el proceso de conversión.

1. Muestreo y retención (S/H)
2. Cuantización y codificación

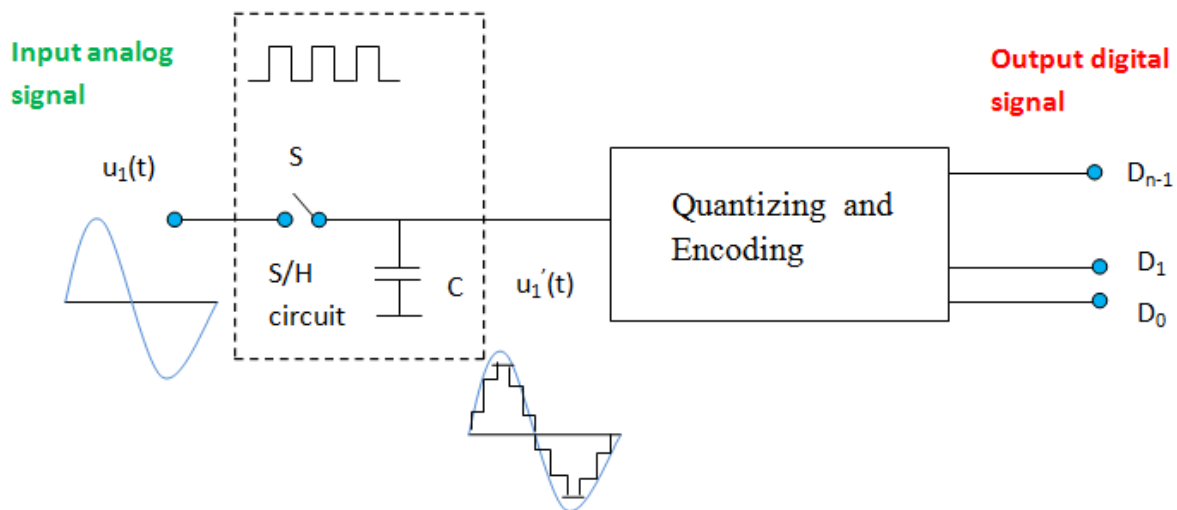


Imagen 5: proceso completo de conversión de una señal analógica a digital

### Muestreo y retención.

En el proceso de muestra y retención (S/H), la señal continua se muestreará y se congelará (retendrá) el valor a un nivel estable durante un período de tiempo mínimo particular. Se hace para eliminar las variaciones en la señal de entrada que pueden alterar el proceso de conversión y, por lo tanto, aumenta la precisión. La frecuencia de muestreo mínima debe ser dos veces la frecuencia máxima de datos de la señal de entrada.

### Cuantización y codificación.

La cuantización es el proceso en el que la señal de referencia se divide en varios cuantos discretos y luego la señal de entrada se corresponde con el cuanto correcto.

Por otra parte, la codificación es, para cada cuanto, se asignará un código digital único y después de eso, la señal de entrada se asignará con este código digital.

### Resolución.

El menor cambio en una señal analógica resultará en un cambio en la señal digital.

$$\Delta V = \frac{V_r}{2^N}$$

Donde:

$V_r$  = rango de voltaje de referencia

$N$  = Numero de bits en salida digital.

$2^N$  = Número de estados.

$\Delta V$  = Resolución

La resolución representa el error de cuantización inherente a la conversión de la señal a formato digital.

Analog signal			Digital o/p
7.5	7	$7\Delta=7V$	111
6.5	6	$6\Delta=6V$	110
5.5	5	$5\Delta=5V$	101
4.5	4	$4\Delta=4V$	100
3.5	3	$3\Delta=3V$	011
2.5	2	$2\Delta=2V$	010
1.5	1	$1\Delta=1V$	001
0.5	0	$0\Delta=0V$	000

Tabla 1: Ejemplo de codificación



## Desarrollo Experimental:

En ésta práctica se va a utilizar el sensor potenciométrico y el de temperatura conectados a un ADC, donde habrá una salida digital de 8 bits la cual será enviada al arduino para procesar su valor, ya sea en centímetros o en grados centígrados para posteriormente enviarlas al dispositivo móvil mediante Bluetooth.

### **Procedimiento experimental 1.- Conexión entre el Arduino y el módulo HC - 05.**

Para la conexión, se tomará la salida TX del módulo HC-05 y se conectará a la entrada 19 del Arduino, para este caso, el arduino mega, donde dicha entrada fue configurada como pin de recepción o RX, luego se conectará la salida RX del módulo a la entrada 18 del Arduino que ya fue configurada como pin de transmisión o TX, se conectará a la tierra la salida GND del módulo, y la entrada de VCC será conectada a 5V.

El módulo HC-05 se usará en modo esclavo ya que solo nos importa realizar la transmisión de datos, porque será el dispositivo móvil el que funcionará como dispositivo maestro.

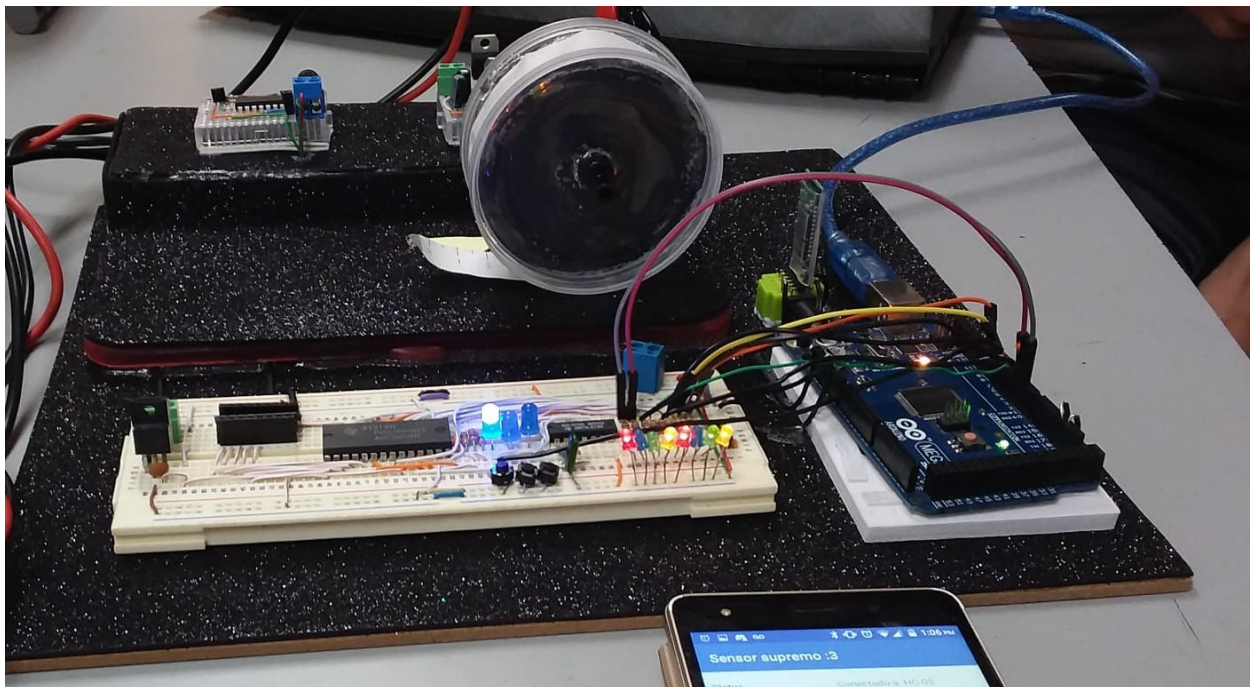


Imagen 6: Proyecto ensamblado con sensor de temperatura, potenciométrico y el módulo Bluetooth y arduino integrado



## Procedimiento experimental 2.- Sensor potenciométrico

La aplicación para el sensor potenciométrico es sencilla, al jalar la cinta métrica se produce una rotación del eje que está unido a ella y a su vez de la perilla del potenciómetro. Al estar la cinta completamente enrollada el potenciómetro tiene un valor de resistencia de 0 ohm, y al estirar la cinta por completo [aprox. 20 cm] la resistencia del potenciómetro es la máxima.

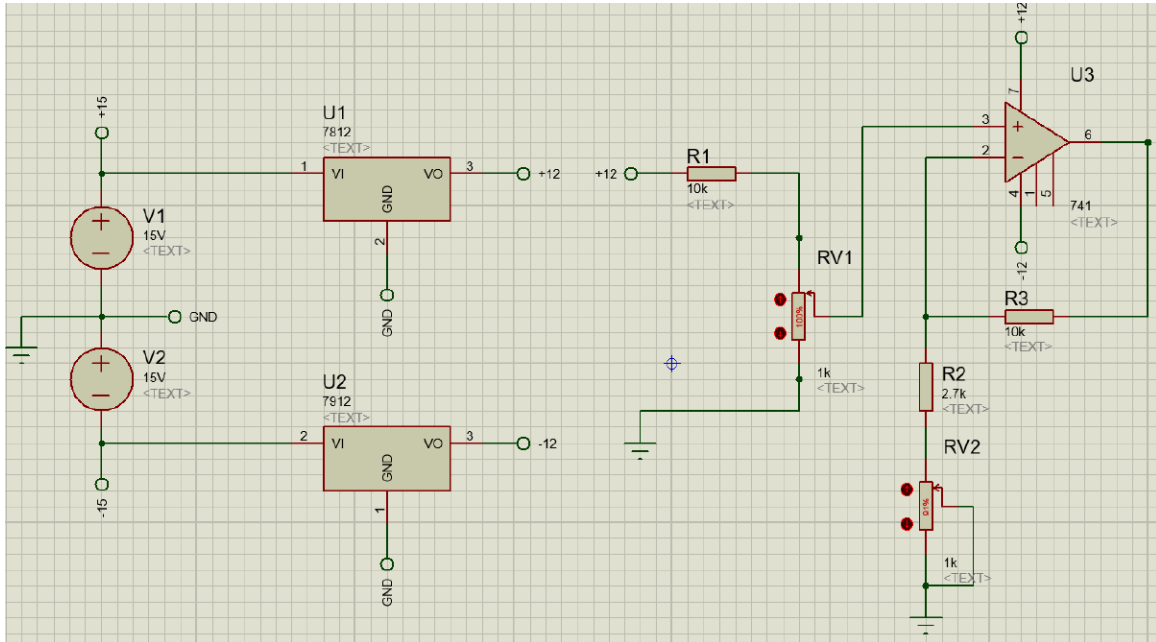


Imagen 7: Esquema del sensor potenciométrico

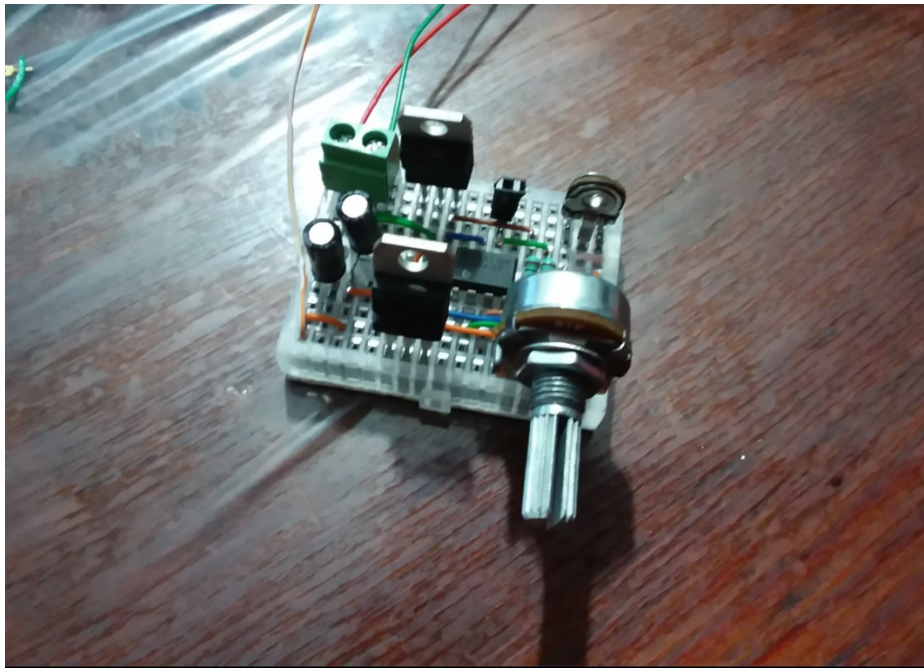


Imagen 8: Sensor potenciométrico antes de ser ensamblado al proyecto

### Procedimiento experimental 3.- Sensor de temperatura

Para el circuito que funcionará como sensor de temperatura se empleó un LM35 y un amplificador operacional, donde mediremos la temperatura ambiente por lo que establecemos que la temperatura máxima será de 50 grados centígrados. De acuerdo con esto, a 50 grados centígrados el voltaje a la salida de la etapa de acondicionamiento será de 5V.

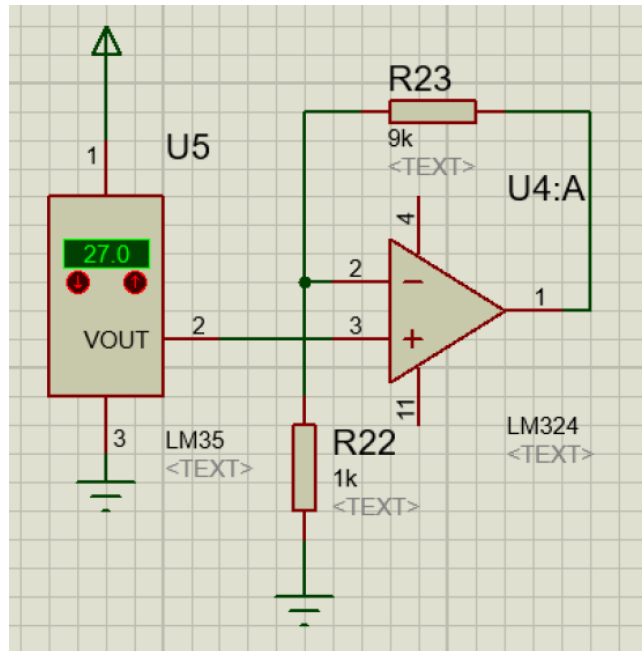


Imagen 9: Esquema del sensor de temperatura

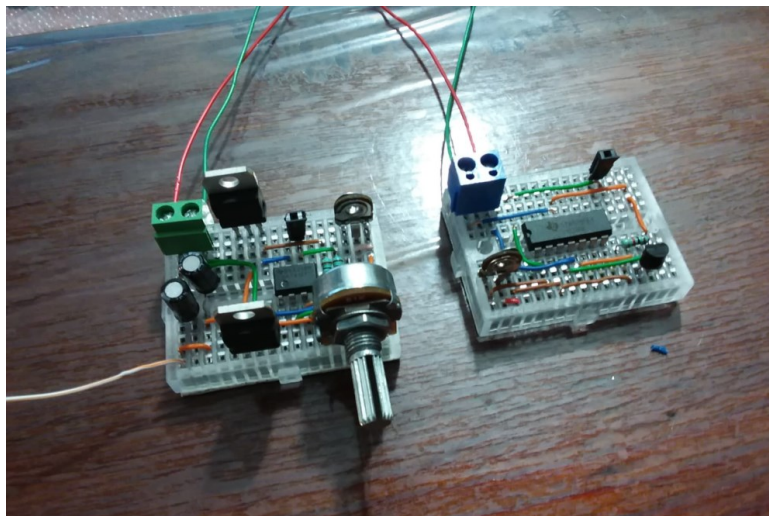


Imagen 10: Sensores potenciométrico (izquierda) y de temperatura (derecha) antes del ensamblado.

#### Procedimiento experimental 4.- ADC

En esta práctica vamos a utilizar el conversor ADC0809 que es un conversor de 8 bits (la señal analógica se convierte en una palabra digital de 8 bits), que tiene la posibilidad de leer 8 señales analógicas. Posee 28 pines de los cuales 8 corresponden a sus canales analógicos de entrada; éste solo puede leer un canal

a la vez y dispone por lo tanto de un selector (multiplexor) de 3 líneas, que permite seleccionar la señal de entrada a convertir, el cual va a ser manipulado mediante el arduino para poder cambiar entre cada uno de los dos sensores de entrada.

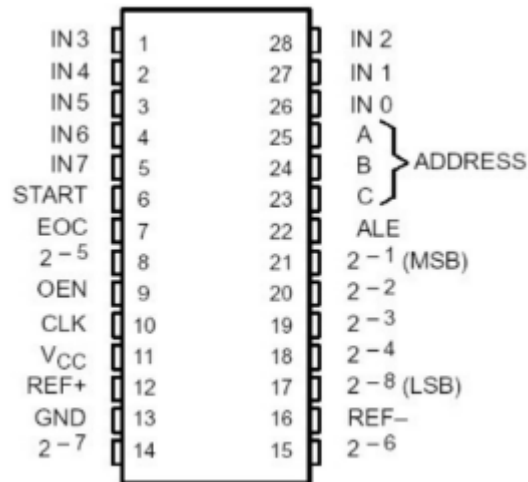


Imagen 11: Configuración del ADC0809

Un tema importante cuando utilizamos un conversor AD es el acondicionamiento de la entrada. Realizaremos el acondicionamiento de una señal proveniente de un sensor de temperatura LM35. Este sensor entrega  $10\text{mV}/^{\circ}\text{C}$ , con lo que a  $100^{\circ}\text{C}$  entrega  $1\text{V}$ .

Supondremos que mediremos la temperatura ambiente por lo que estableceremos la temperatura máxima en 50 grados Centígrados a plena escala. De acuerdo con esto calcule la ganancia y los componentes necesarios para que a 50 grados centígrados el voltaje a la salida de la etapa de acondicionamiento tengamos  $5\text{V}$ .

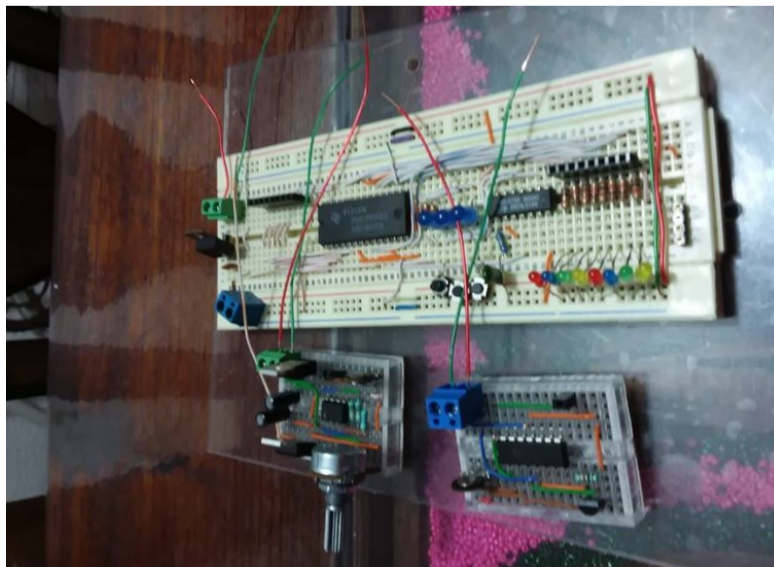


Imagen 12: Circuito del ADC antes de ser ensamblado

## Procedimiento experimental 5.- Conversión de los datos en arduino

Para convertir la entrada digital de 8 bits proveniente del ADC, se tomaron los 8 bits de entrada en el arduino y fueron almacenados en un arreglo de 8 elementos, donde cada posición tiene un índice que va de 0 a 7, y por cada posición que tuviese el valor de 1, se elevaba 2 a la potencia del índice para obtener su valor binario equivalente, dicho valor binario va desde 0 hasta 255.

Posteriormente se multiplica el valor binario obtenido por la resolución de cada sensor.

Resolución del sensor de temperatura

$R = 0.2$

Resolución del medidor de distancia

0.085

De igual manera el arduino envía cada 300ms un 1 o un 0 para cambiar rápidamente entre cada una de las salidas del ADC, diferenciando entre sensor y procesando su valor individual

```
#include <SoftwareSerial.h>
SoftwareSerial BTserial(18 , 19); // RX | TX

int bits[8]; //aqui van a ir los 8 bits de entrada
int pinecillos[] = {2, 3, 4, 5, 6, 7, 8, 9}; //estos son los pinecillos de entrada de los 8 bits
int estadoBit = 0;
int osciladorPin = 10; //este va a sacar el valor 0 o 1 para cambiar de sensor
float tempSensor = 0.0;
float centimetros = 0.0;
int sumaBitsTemp = 0, sumaOswi = 0;

int contadorEnvia = 0; //cuando sea 2, ya recibí el valor de ambos sensores y mando al BT

void setup()
{
  for (int i = 0; i < 8; i++) //para leer los 8 bits
  {
    pinMode(pinecillos[i], INPUT); //Declarando los 8 bits como puertos de entrada
  }
  pinMode(osciladorPin, OUTPUT); //la salida de 0 y 1
  Serial.begin (9600);
  BTserial.begin(9600);
```

```

}

void loop()
{
  if (contadorEnvia == 0)
  {
    digitalWrite(osciladorPin, LOW); //Mando un cero para el de temperatura
    contadorEnvia = 1;
  }
  else
  {
    if (contadorEnvia == 1)
    {
      digitalWrite(osciladorPin, HIGH); //Mando un 1 para el de oswi
      contadorEnvia = 2;
    }
    else //ya es 2 y puedo enviar del BT
    {
      Serial.println("Enviando data");

      contadorEnvia = 0;
    }
  }
  for (int i = 0; i < 8; i++) //para leer los 8 bits
  {
    estadoBit = digitalRead(pinecillos[i]); //leo el valor de cada pin de entrada
    if (estadoBit == HIGH)
    {
      bits[i] = 1;
    }
    else
    {
      bits[i] = 0;
    }
  }

  sumaBitsTemp = 0;
  for (int i = 0; i < 8; i++)
  {
    //Serial.print(bits[i]);
    if (bits[i] == 1)
    {
      sumaBitsTemp += pow(2, i); //voy sumando los bits para tener de 0 hasta 255
    }
  }
}

```

```

}

if (contadorEnvia == 1) //recibi los datos del de temperatura y los proceso
{
    tempSensor = sumaBitsTemp * 0.2; //multiplico por la resolución de cada bit
    Serial.println(tempSensor);
    BTserial.print("T");
    BTserial.print(tempSensor);
    sumaBitsTemp = 0;
    tempSensor = 0;
}
else //recibí datos de Oswi
{
    if(contadorEnvia == 2)
    {
        centimetros = sumaBitsTemp * 0.085; //multiplico por la resolución de cada bit
        Serial.println(centimetros);
        BTserial.print("C");
        BTserial.print(20-centimetros);
        sumaBitsTemp = 0;
    }
}

for (int i = 0; i < 8; i++) //para leer los 8 bits
{
    bits[i] = 0;
}

delay(300);
}

```

Tabla 1: Código fuente de la aplicación en Arduino

## Procedimiento experimental 6.- Despliegue en dispositivo móvil



Una vez desarrollada la app para dispositivos con sistema operativo Android, la cual llamamos Sensor Supremo, y una vez establecida la conexión entre el dispositivo móvil y el módulo HC-05, únicamente se procede a desplegar el valor de cada uno de los sensores.

La aplicación únicamente recibe el valor proveniente del módulo Bluetooth, el cual tiene concatenado una letra "C" o una letra "T" al principio antes del valor, con la cual puede distinguir entre sensores y saber a qué campo le corresponde cada valor.

Entre cada medición nuevo hay un intervalo de 300 milisegundos, que fue implementado mediante un simple delay () en el arduino.

Imagen 13: Interfaz de Sensor Supremo



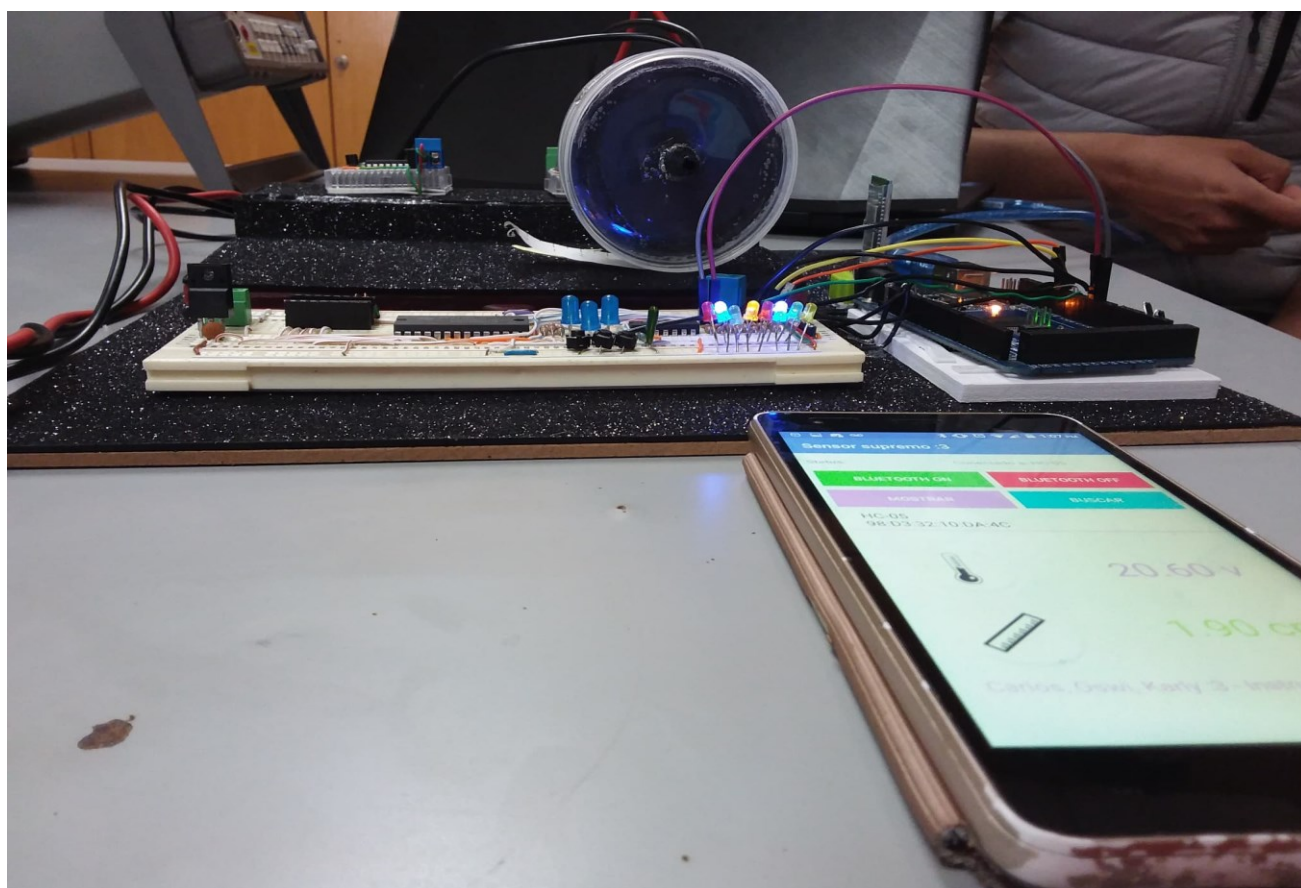


Imagen 14: Funcionamiento completo del circuito con la aplicación, desplegando los valores de cada sensor en la aplicación.

## Conclusiones:

Frías Mercado Carlos Elliot:

Al conectar el Arduino, dispositivo móvil y el circuito nos damos cuenta de las enormes posibilidades que hay con el uso del Bluetooth, ya que no solo se puede enviar un solo dato, sino que múltiples de ellos también al mismo tiempo, los cuales podemos procesar para enviarlos a la manera que nosotros deseamos, pudiendo desplegar esa información en cualquier dispositivo, existiendo la posibilidad del almacenar en una base de datos o generar gráficas.

Gómez Ramírez Oswaldo:

Los instrumentos para medir distancias son de los más comunes, pero el funcionamiento de este distanciómetro fue en especial interesante, pues fue necesario considerar muchas variables del circuito y del entorno. Al término de la práctica pude observar a detalle cómo es que funciona un distanciómetro, desde la emisión de la onda, la calibración, la medición del voltaje y la conversión a distancia.

Hernández Castro Karla Beatriz:

Dentro de la instrumentación una parte fundamental es conocer que hay en nuestro entorno y además entenderlo y con ello poder controlarlo; recordando que la instrumentación es un conjunto de instrumentos que nos ayudan a observar, medir, transmitir, convertir señales o combinaciones de estas mismas. Basándonos en esta idea con el proyecto final aprendimos a combinar dos sensores que miden cosas totalmente distintas como la temperatura y distancias, obteniendo sus señales analógicas y convirtiéndolas en señales digitales y mostrando los datos a través de una aplicación, todo esto me enseñó a que existe un tratamiento diferente por señal, un acoplamiento diferente y un tratamiento de la señal de diferente manera.

