



INSTITUTO POLITÉCNICO NACIONAL  
ESCUELA SUPERIOR DE CÓMPUTO

APPLICATION DEVELOPMENT FOR MOBILE  
DEVICES

“PROYECTO 01: ARDUINO Y ANDROID”

GRUPO: 3CV16

BOBADILLA SEGUNDO DAN ISRAEL

PROFESOR: CIFUENTES ALVAREZ ALEJANDRO SIGFRIDO

FECHA DE ENTREGA: 08/11/2021



# INTRODUCCIÓN

La conexión de dispositivos es la parte central de nuestra tecnología, como lo es el WIFI, NFC y Bluetooth, todas estas tecnologías son tan importantes que si un celular no contara con ellas quedaría completamente obsoleto. En el presente trabajo se mostrará como conectar un Arduino con un dispositivo móvil utilizando Bluetooth con el objetivo de adentrarse al mundo de las comunicaciones.

Además, tiene como objetivo principal ser una guía para el usuario que se este adentrando en el mundo de la programación de dispositivos móviles.

## OBJETIVOS

- Crear una conexión dinámica para cualquier Modulo de Arduino
- Crear un sensor que pueda medir la capacitancia de un capacitor en caso de desconocer de que valor es
- Crear un sensor que pueda medir las revoluciones por minuto de una hélice
- Entender el funcionamiento de los sockets en Android
- Crear una herramienta que sea fácil de entender para el usuario
- Utilizar mas a fondo los Intent para realizar peticiones de servicios

## DESARROLLO

El presente trabajo se dividirá en dos partes, únicamente el armado o la parte del servidor (Arduino) será diferente, debido a que el programa esta diseñado para ambos usos. Primero comenzaremos con la parte de la programación en Android.

Nota: El módulo que utilizo es el HC-05 FN-114 el cual necesita un divisor de voltaje debido a que no soporta el voltaje de 5V que ofrece Arduino, si se utiliza con los pines 0 y 1 digitales es posible que tenga un problema al cargar el programa, si remueve estos pines al cargar el programa no tendrá problema, sin embargo, recomiendo utilizar los pines 2 y 3 (RX, TX) ya que no nos creara ningún tipo de problema, aun así puede configurar los pines en la salida 4 y 5 para la utilizar el "Tacómetro" si lo desea, ya que para utilizar el tacómetro los pines de intervención del Arduino UNO son 2 y 3.

## CODIGO JAVA

Como primera pantalla tenemos una interfaz sencilla, donde se le pedirá al usuario que encienda su Bluetooth presionando el botón (Ilustración 1).

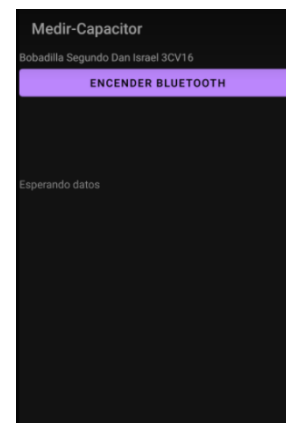


Ilustración 1

Al encender el Bluetooth se le mostrará una alerta de diálogo para indicarle que deberá tener en su lista de dispositivos conocidos al módulo HC (Ilustración 2).

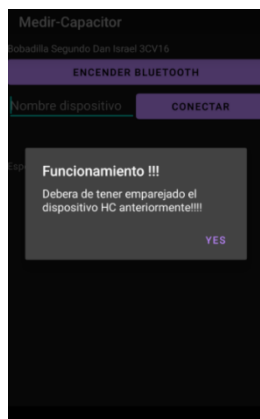


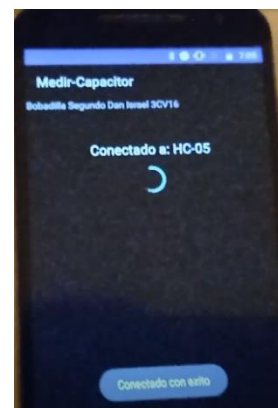
Ilustración 2

Ahora el usuario únicamente deberá de ingresar el nombre del modulo HC y al presionar “CONECTAR” este será conectado (Ilustración 2).

Para conectar el dispositivo con el modulo HC debemos de crear una Lista de dispositivos donde en cada uno buscaremos el nombre que tienen y si es el mismo que el usuario haya escrito obtendremos su dirección MAC la cual nos servirá para crear un Socket (Ilustración 3).

```
private void conectarDispositivo(String name){  
    pairedDeviceList = mBlueAdapter.getBondedDevices();  
    for (BluetoothDevice pairedDevice: pairedDeviceList) {  
        if(pairedDevice.getName().equals(name)){  
            address = pairedDevice.getAddress();  
            Log.d( tag: "PRUEBA", address);  
        }  
    }  
}
```

Ilustración 3



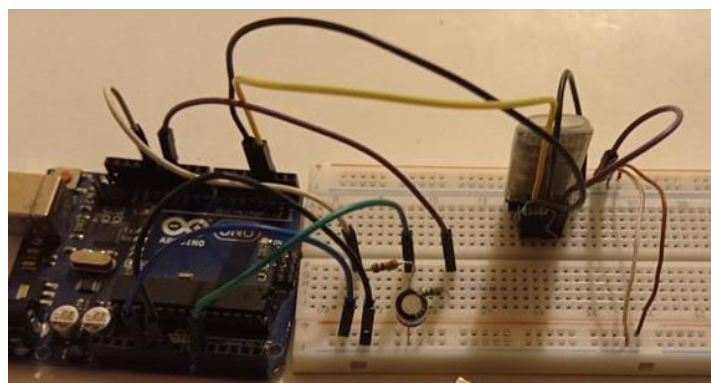
Obtendremos el Socket utilizando la address, para usar el Socket es el mismo principio que se utiliza si trabajáramos con Java nativo, debemos de poner en escucha al cliente (Android) y obtener estos datos en forma de byte estos deben ser casteados a String y mandarlos a imprimir. El código se omite debido a que estos temas son enseñados en POO. Por último el celular se pondrá a escuchar hasta que llegue algún tipo de información del HC (Ilustración 3)

# ARDUINO MEDIDOR DE CAPACITANCIA

## MATERIALES

- 1 Arduino UNO
- 1 Resistencia 10k Ohm (No necesaria)
- 1 Resistencia 220k Ohm
- 1 Capacitor 10uF
- 1 Modulo HC-05

La conexión se muestra en la siguiente imagen (Ilustración 4).



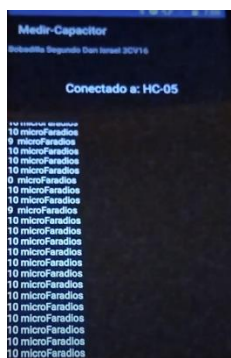
*Ilustración 4*

Para conectar el módulo con el Arduino tuvimos que encontrar la velocidad adecuada en este caso es 9600, para medir la capacitancia lo conectamos en un pin análogo que nos permite medir el voltaje y gracias a ello podemos ir obteniendo el PWD del capacitor, para la parte del envío de datos, llamamos al BT para escribir los datos y utilizamos Serial para leer los datos que el Arduino nos manda (Ilustración 5).

```
BTserial.println((long)microFarads);  
BTserial.write(Serial.read());  
delay(500);
```

*Ilustración 5*

Ahora podremos visualizar la información enviada desde el HC-05 a la aplicación (Ilustración 6).



*Ilustración 6*

# ARDUINO MEDIDOR DE RPM

## MATERIALES

- 1 Arduino UNO
- 1 Resistencia 10k Ohm
- 1 Resistencia 1k Ohm (Varía dependiendo de la distancia)
- 1 Led IR
- 1 Fotodiodo
- 1 Modulo HC-05

La conexión se muestra en la siguiente imagen (Ilustración 7).

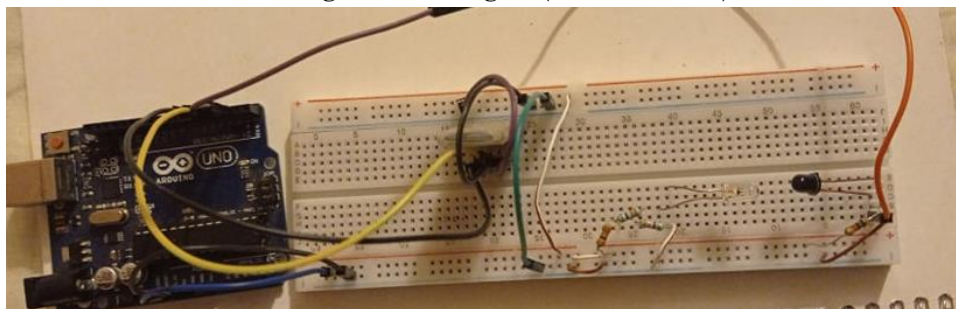


Ilustración 7

Como podemos ver tenemos un conjunto de resistencias, debido a que no se contaba con una de 1k decidí poner en serie varias resistencias para obtener 1k, es la misma conexión que se mostro anteriormente con el único cambio que movimos los pines del modulo como se indico al principio del documento. Para medir las revoluciones, dividimos 60 segundos / número hélices y el resultado será multiplicado por el contador y esto nos dará las RPM (Ilustración 8).

```
BTserial.println(contador*30);  
BTserial.write(Serial.read());
```

Ilustración 8

Por último el resultado se muestra a continuación (Ilustración 9).

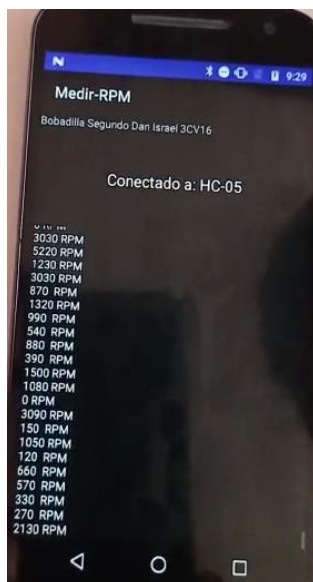


Ilustración 9



## CONCLUSIONES PERSONALES

En lo personal nunca había trabajado con lo que es Arduino, al principio tuve que leer documentación y ver un curso de Arduino, tuve muchos problemas con mi Modulo HC-05 debido a que en especial tiene un problema con el voltaje, incluso la misma placa presenta un problema si se utilizan los pines digitales 0 y 1. También a la hora de trabajar con la conexión de Arduino y el Modulo para el envío de datos tuve bastantes problemas al establecer las velocidades. Sin embargo, logré solucionar todas las problemáticas que tuve. Por otro lado, en la parte de Android no tuve tantos problemas ya que tenía en mente que debía de hacer, eh trabajado con Sockets únicamente en lenguaje C (Linux) por lo que todo lo que hice en C tuve que pasarlo a Java y además ya tenía un poco de experiencia trabajando con Sockets en java, no me resulto difícil poder obtener los datos, creo que el mayor reto fue utilizar Arduino.

## CONCLUSIONES OBJETIVAS

Logramos todos los objetivos que habíamos mostrado al principio del reporte, trabajar en la conexión de Bluetooth es algo diferente y nos abre un sinfín de posibilidades, una de ellas es que ahora podremos realizar la conexión entre dos dispositivos por medio de Bluetooth. Además, podemos utilizar esta información para realizar nuevos proyectos que impliquen utilizar Arduino para realizar una tarea en específico.

## BIBLIOGRAFIA

[ 1] Gironés, Jesús. *El gran libro de Android*. Barcelona: Marcombo, 2018. Print.

[ 2] (S/f). *Android.com*. Recuperado el 1 de noviembre de 2021, de <https://developer.android.com/guide?hl=es-419>

[ 3] *Obtuve información de diferentes foros para programar en Arduino por lo que sería complicado citar todos.*