

PT_9

CONTROLAR EL ESTADO DE UN LED MEDIANTE LA CONEXIÓN BLUETOOTH USANDO APP INVENTOR Y UN MODULO HC-06.

AUTOR: AHMED CHTATOU BOUGDADI

ÍNDICE

1. INTRODUCCIÓN.....	2
2. MATERIALES Y COMPONENTES.....	2
3. DATASHEET MODULO BLUETOOTH HC-06	3
4. ¿QUE ES APP INVENTOR?.....	5
5. DISEÑO DE LA APLICACIÓN, PROGRAMACIÓN E INSTALACION DE LA MISMA EN EL DISPOSITIVO MOVIL.....	7
6. CÓDIGO	12
7. CIRCUITOS EN FRITZING.....	13
8. VINCULAR EL DISPOSITIVO CON EL TELEFONO MOVIL MEDIANTE BLUETOOTH.....	15
9. FOTOS Y VIDEOS DEL MONTAJE.....	15

1. INTRODUCCIÓN

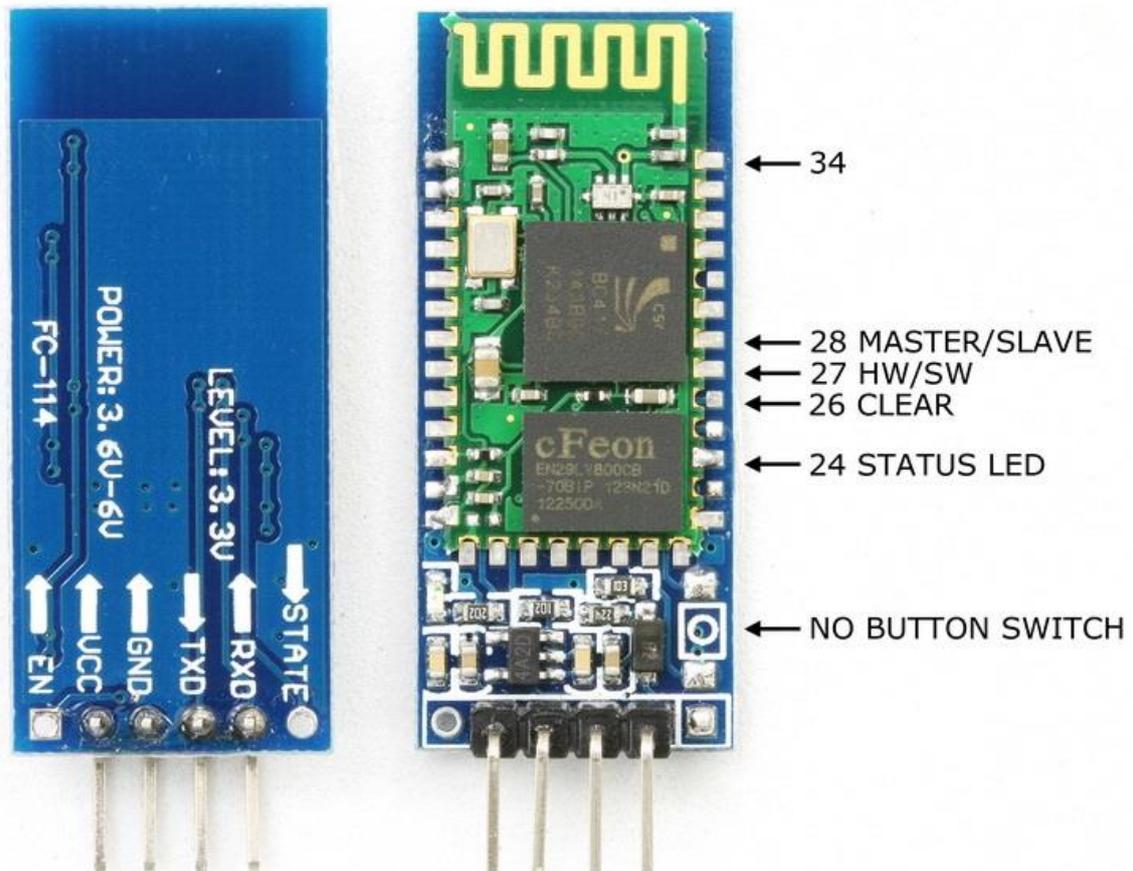
El objetivo de esta práctica es poder usar la comunicación Bluetooth a través de Arduino, la cual consiste en poder controlar desde un dispositivo móvil mediante una app creada desde la web App Inventor y un módulo bluetooth HC-06 el encendido y apagado de un led.

También aprenderemos a cómo crear nuestra aplicación desde app inventor, como programar nuestra placa arduino y como conectarlo correctamente con el dispositivo bluetooth para enlazar la conexión y así llevar acabo nuestro objetivo de encender y apagar el led.

2. MATERIALES Y COMPONENTES

- Protoboard
- Placa Arduino UNO
- Modulo Bluetooth HC-06
- Led (cualquier color)
- Resistencia de 220k Ω
- Jumpers (cables)
- Un dispositivo móvil con Android.

3. DATASHEET MODULO BLUETOOTH HC-06



Características:

Especificación bluetooth v2.0 + EDR (Enhanced Data Rate)

Modo esclavo (Solo puede operar en este modo)

Puede configurarse mediante comandos AT (Deben escribirse en mayúscula)

Chip de radio: CSR BC417143

Frecuencia: 2.4 GHz, banda ISM

Modulación: GFSK (Gaussian Frequency Shift Keying)

Antena de PCB incorporada

Potencia de emisión: ≤ 6 dBm, Clase 2

Alcance 5 m a 10 m

Sensibilidad: ≤ -80 dBm a 0.1% BER

Velocidad: Asíncrona: 2 Mbps (max.)/160 kbps, síncrona: 1 Mbps/1 Mbps

Seguridad: Autenticación y encriptación (Password por defecto: 1234)

Perfiles: Puerto serial Bluetooth

Módulo montado en tarjeta con regulador de voltaje y 4 pines suministrando acceso a VCC, GND, TXD, y RXD

Consumo de corriente: 30 mA a 40 mA

Voltaje de operación: 3.6 V a 6 V

Dimensiones totales: 1.7 cm x 4 cm aprox.

Temperatura de operación: -25 °C a +75 °C

Aplicaciones:

Comunicación inalámbrica entre microcontroladores

Comunicación inalámbrica entre computadoras y microcontroladores

Comunicación inalámbrica entre teléfonos móviles o tabletas y microcontroladores

Advertencias:

La comunicación Bluetooth entre dos módulos debe realizarse entre un módulo configurado como maestro y otro como esclavo. Para la comunicación Bluetooth con computador, teléfono, PDA, tableta, etc., el módulo debe ser esclavo.

comandos AT y observar la respuesta del módulo. Como este monitor emplea la comunicación serial que el Arduino utiliza para comunicarse con la computadora en los pines 0 y 1 digitales, se crea un puerto serial por software para pasar los datos al módulo Bluetooth empleando los pines digitales 10 y 11.

También se podría hacer con un sketch mas sencillo, directamente conectando el módulo al puerto serie de la placa Arduino (Pines digitales 0 y 1), pero se tendría que remover el microcontrolador del Arduino primero.

Por supuesto también se pueden enviar los comandos AT desde cualquier microcontrolador sin ayuda de computadoras.

Configuración:

El módulo suele venir configurado con velocidad de transmisión serial de 9600 bps, 1 bit de parada, y sin bit de paridad, nombre: linvor, password: 1234

Para su configuración se puede conectar a el viejo puerto serial RS232 de la computadora a través de un convertidor TTL a RS232, o mejor empleando un conversor USB a serial TTL y utilizando el Hyperterminal de Windows u otro programa con funciones de terminal serial para enviar los comandos AT (Por ej. el SSCOM32, PuTTY, etc.). (A partir de Win Vista el hyperterminal ya no está incluido en el SO)

Con Arduino también se puede hacer fácilmente y sin ningún convertidor con un pequeño sketch que utiliza el monitor serial del IDE de Arduino para escribir los comandos AT y observar la respuesta del módulo. Como este monitor emplea la comunicación serial que el Arduino utiliza para comunicarse con la computadora en los pines 0 y 1 digitales, se crea un puerto serial por software para pasar los datos al módulo Bluetooth empleando los pines digitales 10 y 11.

También se podría hacer con un sketch más sencillo, directamente conectando el módulo al puerto serie de la placa Arduino (Pines digitales 0 y 1), pero se tendría que remover el microcontrolador del Arduino primero.

Por supuesto también se pueden enviar los comandos AT desde cualquier microcontrolador sin ayuda de computadoras.

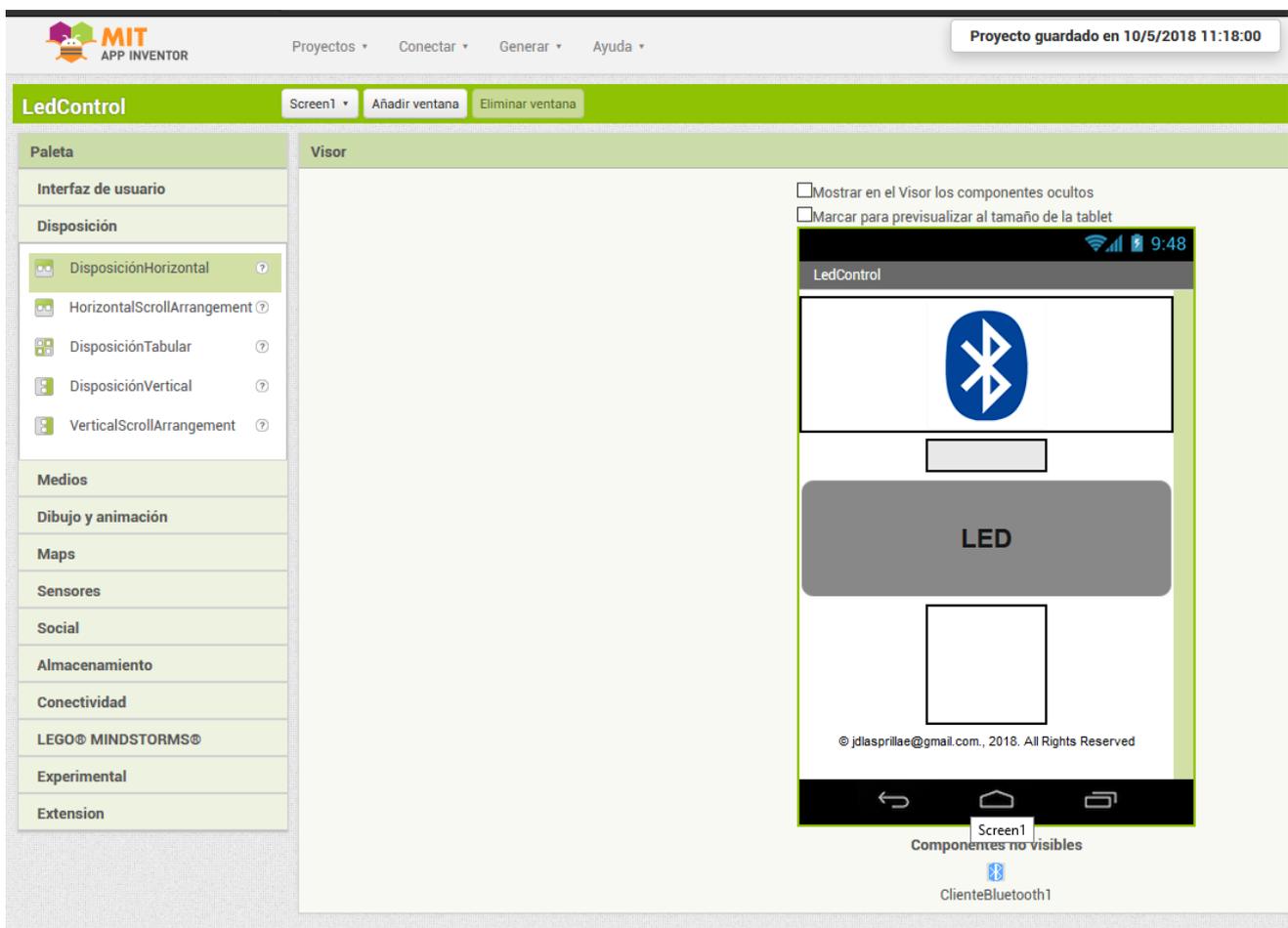
4. ¿QUE ES APP INVENTOR?

App Inventor es un entorno grafico de programación orientado a la educación y aprendizaje, la finalidad de esta aplicación es dar la posibilidad a cualquier persona de realizar aplicaciones para tablets o teléfonos móviles Android sin necesidad de tener muchos conocimientos de programación e informática.

App Inventor Es un software libre y funciona correctamente solo en dispositivos Android. <http://appinventor.mit.edu/explore/>

5. DISEÑO DE LA APLICACIÓN, PROGRAMACIÓN E INSTALACION DE LA MISMA EN EL DISPOSITIVO MOVIL

En la siguiente imagen se muestra cómo quedaría finalmente el diseño de mi app, aquí como bien se ve en la imagen he añadido el símbolo del bluetooth para reemplazar el texto del selector de lista, esto lo he quitado con las opciones para ajustarlo luego. he puesto un espacio añadiendo una disposición horizontal desde la paleta disposición para luego añadir el botón que controlara el estado de nuestro led

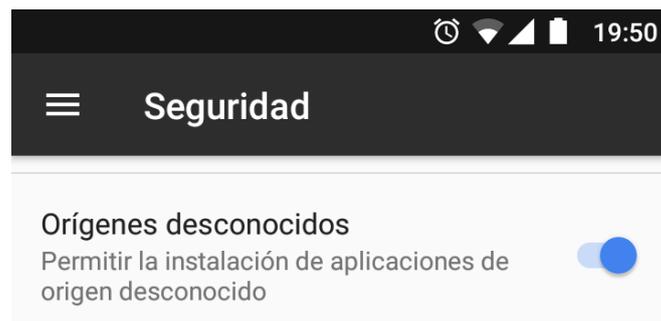


Una vez teniendo el diseño de nuestra app ahora lo que vamos es a configurar su función o programarla.

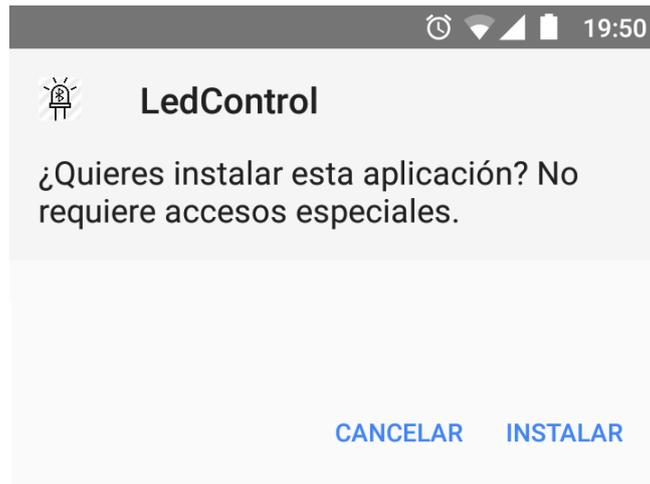
Para esto nos dirigimos a la pestaña de bloques, el cual nos llevará a una ventana donde nos aparecerá la lista de los componentes que conforman nuestra aplicación y unas series de herramientas para ir añadiendo a la función de la app. Como es muy fácil de configurar los bloques solo es ir arrastrándolos hacia el editor y haciendo la función que queremos que realice nuestra app. Para esto dejo la imagen de la programación completa de los bloques.



Una vez terminada nuestra app, lo que vamos hacer a continuación es instalarla en nuestro dispositivo móvil. Exportamos la app en formato .apk esto lo hacemos clicando en la parte superior "generar" y clicamos en guardar archivo apk en mi ordenador, este archivo lo transferimos a nuestro dispositivo móvil mediante un cable USB. Una vez teniendo el archivo en el dispositivo y antes de instalarlo vamos a los ajustes de seguridad de nuestro dispositivo y damos permisos para orígenes desconocidos.



Vamos a la ubicación donde almacenamos nuestra app y la instalamos.



Una vez instalada nuestra app la abrimos, y esta sería la aplicación para controlar el estado del led, ya solo nos quedaría cargar el código en nuestra placa y hacer las conexiones de los dispositivos correctamente.



© jdlasprillae@gmail.com., 2018. All Rights Reserved

6. CÓDIGO

El código que usaremos a continuación es uno de los ejemplos que contiene arduino, el cual se llama PhysicalPixel, y lo encontramos en >Archivo >Ejemplos >Communication > PhysicalPixel, hacemos todas las conexiones tal cual como no las dice el ejemplo

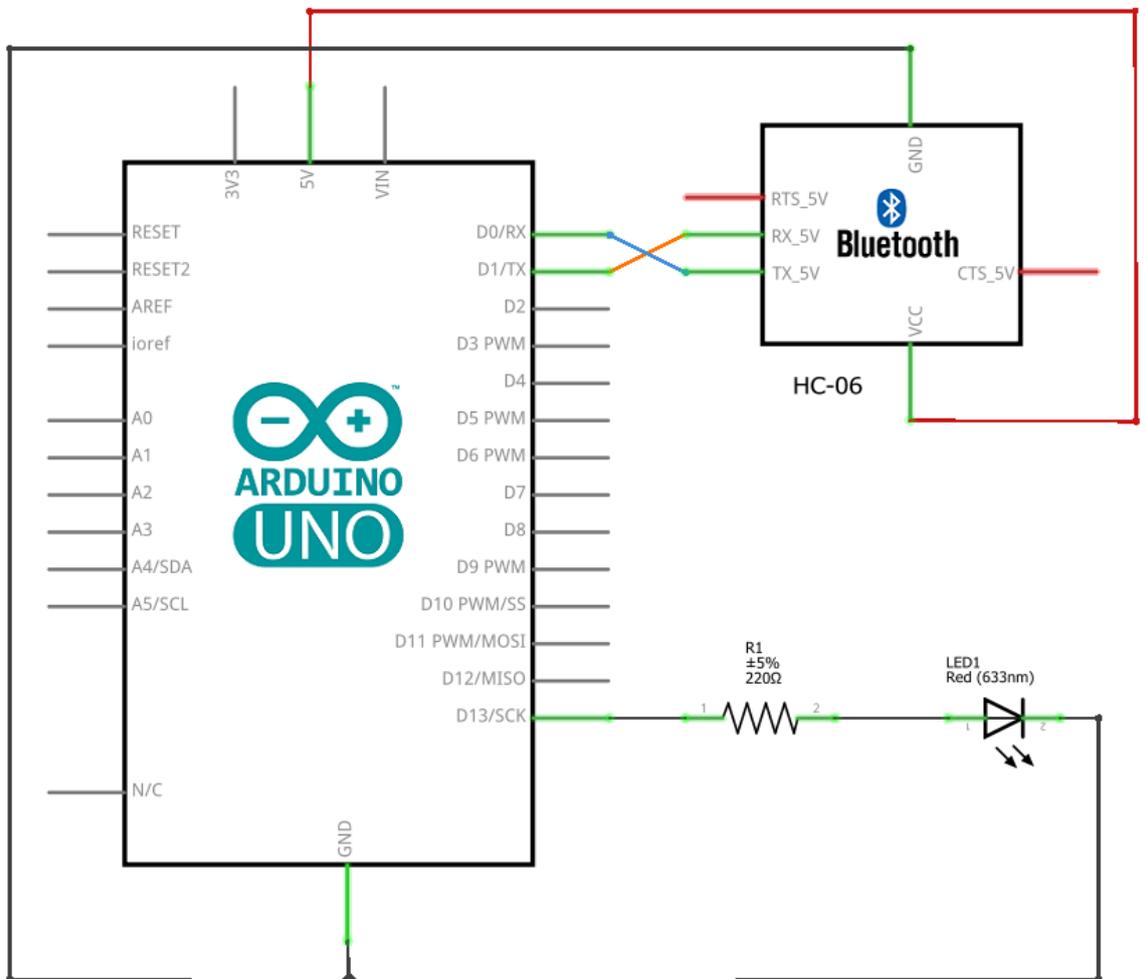
```
const int ledPin = 13; // el pin al que esta conectado el LED
int incomingByte;      // variable para leer los datos entrantes del
dispositivo

void setup() {
  // inicia la comunicacion serie con el dispositivo
  Serial.begin(9600);
  // asignamos el pin del led como salida
  pinMode(ledPin, OUTPUT);
}

void loop() {
  // revisar si hay datos serie entrantes
  if (Serial.available() > 0) {
    // lee el byte mas antiguo en el buffer del serial
    incomingByte = Serial.read();
    51/5000
    // si es una H mayuscula (ASCII 72), enciende el LED:
    if (incomingByte == 'H') {
      digitalWrite(ledPin, HIGH);
    }
    // si es una L mayuscula (ASCII 76) apaga el LED:
    if (incomingByte == 'L') {
      digitalWrite(ledPin, LOW);
    }
  }
}
```

7. CIRCUITOS EN FRITZING

ESQUEMÁTICO.



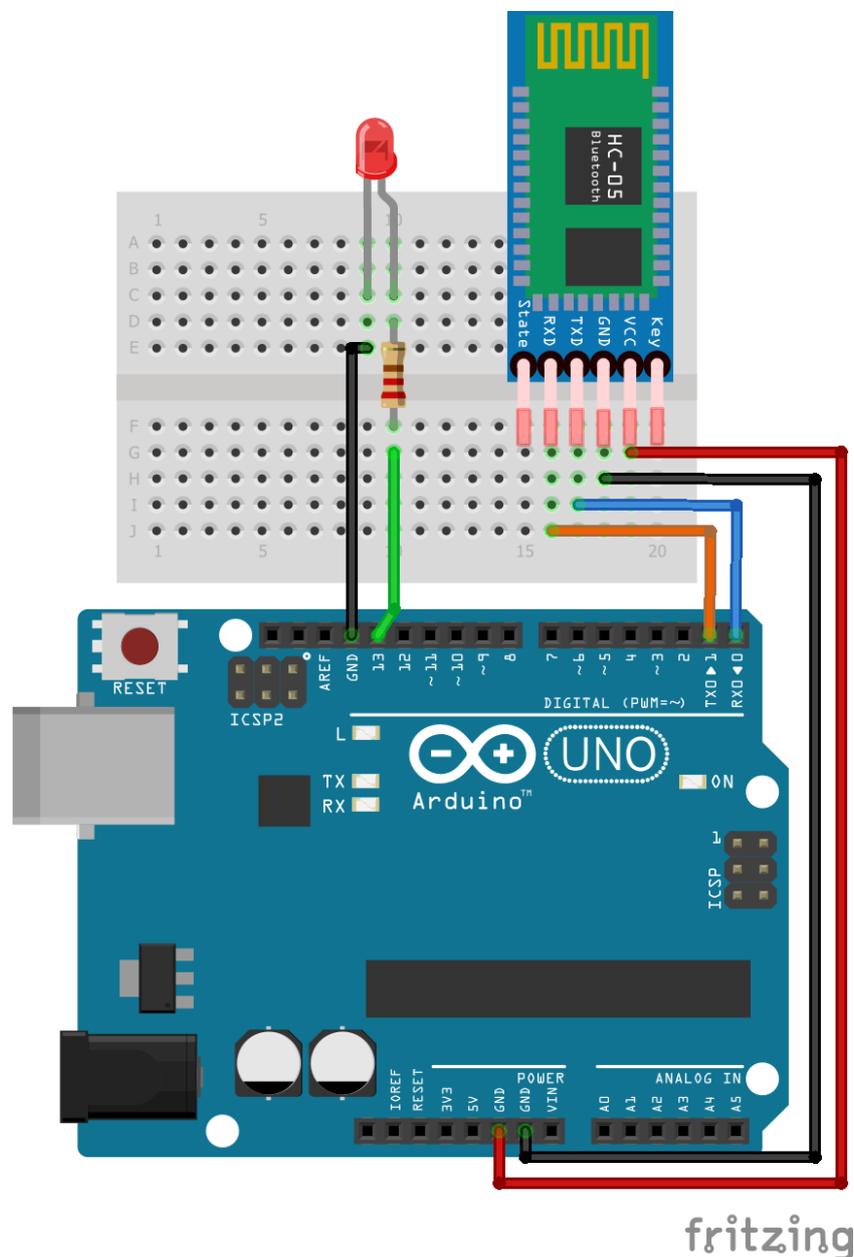
fritzing

El circuito está conectado de la siguiente manera:

Una de las patas del LED está conectada a la salida del pin 13 mediante una resistencia de 220k y la a GND

En el módulo conectamos VCC a 3V de la placa Arduino dependiendo del módulo que tengamos puede ser 5V o 3V. GND lo conectamos a GND de la placa Arduino. El TX del módulo ira conectado al RX de la placa arduino Y el RX del módulo al TX de la placa.

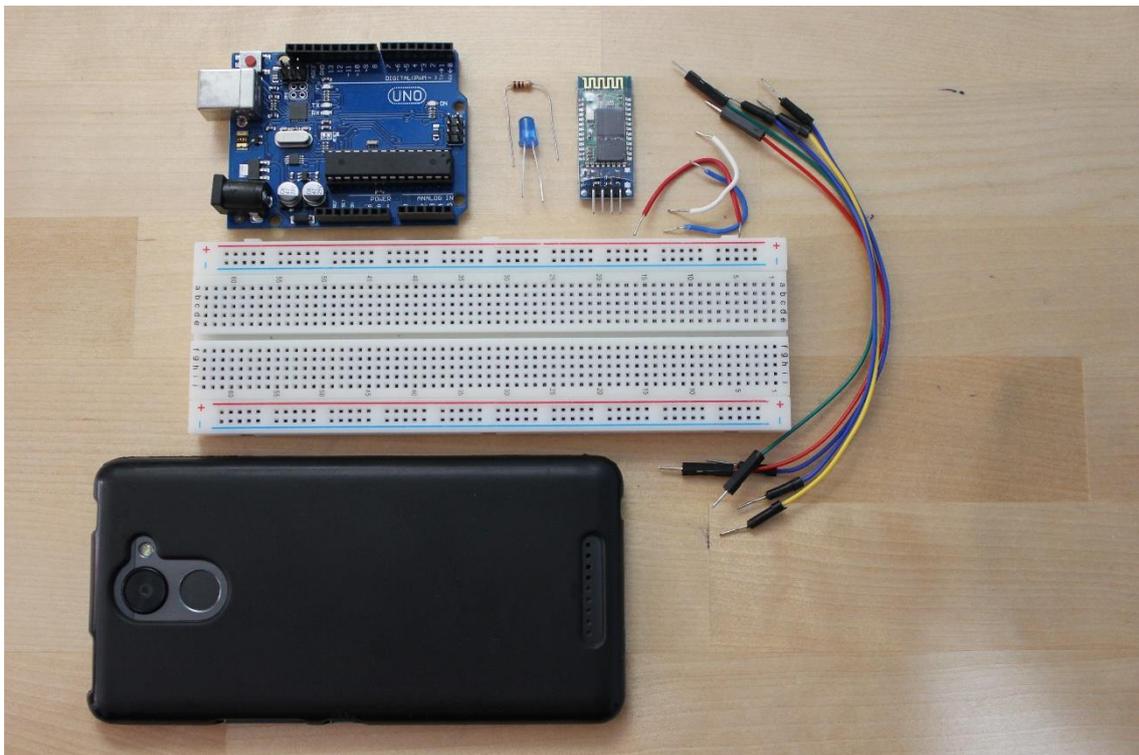
PLACABORAD

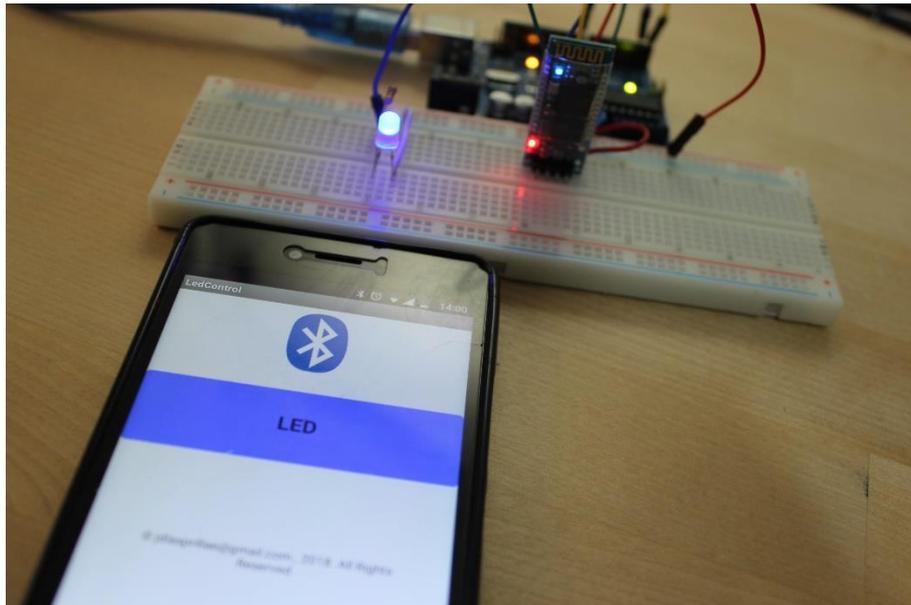


8. VINCULAR EL DISPOSITIVO CON EL TELEFONO MOVIL MEDIANTE BLUETOOTH

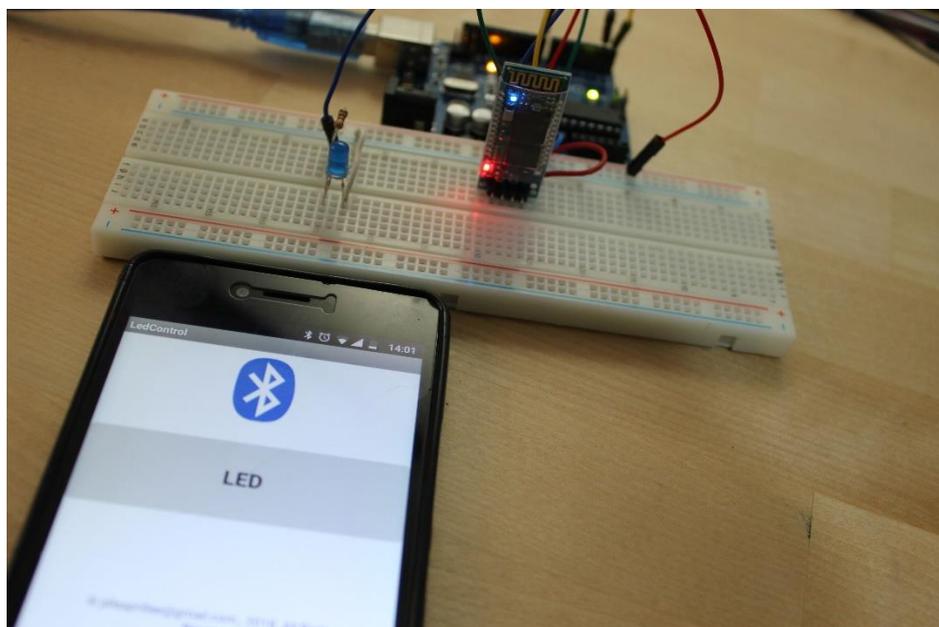
Para vincular nuestro dispositivo bluetooth, activamos el bluetooth en nuestro teléfono móvil y buscamos nuevos dispositivos, a continuación, nos detectará el dispositivo del módulo, en mi caso HC-06, le damos a conectar o simplemente lo seleccionamos y nos pedirá un pin para poder vincular el dispositivo que por defecto es 1234, una vez vinculado el dispositivo ya la aplicación que creamos antes puede acceder a él como veremos en el video del funcionamiento.

9. FOTOS Y VIDEOS DEL MONTAJE





En esta imagen se ve cuando pulsamos el botón LED en el dispositivo móvil se vuelve azul lo cual nos indica que el led esta encendido.



En esta imagen vemos que el botón LED es de color gris el cual nos indica que el LED está apagado.