**Ejercicio 16121601**

Modulo Bluetooth HC-05 V2 con capacidad para gestionar el modo master y el modo slave por configuración. Es un módulo sencillo e ideal para pequeños proyectos en los que buscas una comunicación fácil entre tu móvil y Arduino u otros microcontroladores.

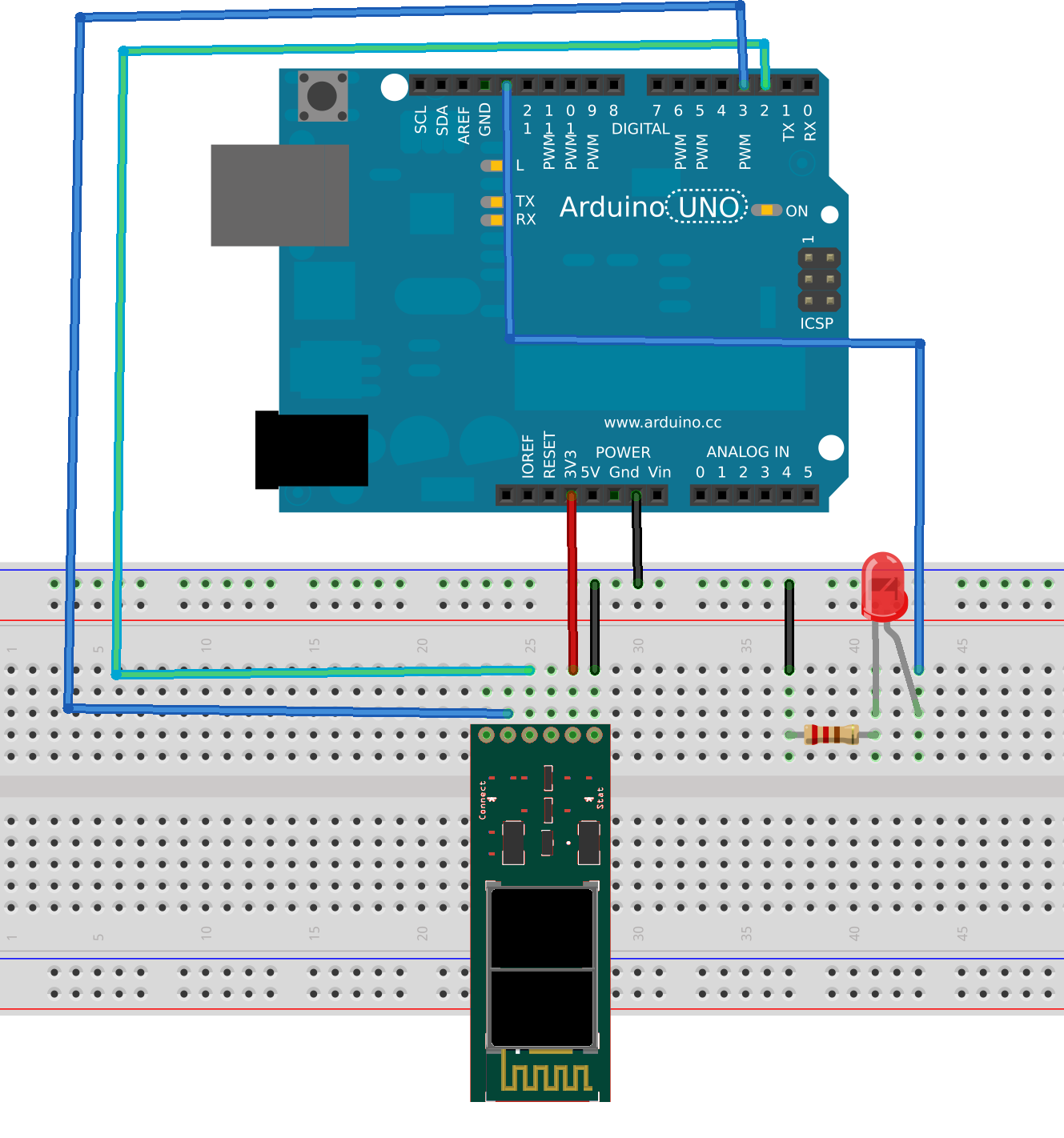
Tenemos varias sesiones presentando este módulo y sus hermanos HC-06 (Que solo acepta el modo slave): y HC-08, HC-10 que acepta la norma Bluetooth LE V4.0:

Configuración Bluetooth Esclavo

Hardware Requerido:

* Tarjeta Arduino Uno
* Resistencia (220 ohm)
* LED
* Módulo Serial Bluetooth(SLAVE)
* Protoboard

Circuito:



Bluetooth es una especificación tecnológica para redes inalámbricas que permite la transmisión de voz y datos entre distintos dispositivos mediante una radiofrecuencia segura (2,4 GHz). Esta tecnológica permite las comunicaciones sin cables ni conectores y la posibilidad de crear redes inalámbricas domésticas para sincronizar y compartir la información que se encuentra almacenada en diversos equipos.

Los comandos AT son instrucciones codificadas que conforman un lenguaje de comunicación con un terminal módem, en este caso el módulo BT. Los comandos AT se denominan así por la abreviatura de atención.

Para la configuración del Esclavo Bluetooth, se usan los comandos:

AT: Prueba de comunicación

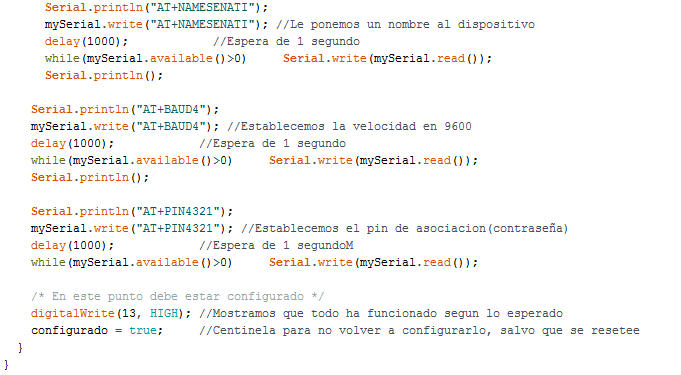
AT+NAME:Cambia el nombre del dispositivo BT

AT+BAUD:Cambia la comunicación serial del BT

AT+PIN:Cambia la contraseña asociada al módulo BT

*// RX, TX*





AT : Ceck the connection.  
AT+NAME : See default name  
AT+ADDR : see default address  
AT+VERSION : See version  
AT+UART : See baudrate  
AT+ROLE: See role of bt module(1=master/0=slave)  
AT+RESET : Reset and exit AT mode  
AT+ORGL : Restore factory settings  
AT+PSWD: see default password

#include <SoftwareSerial.h>  
#define rxPin 11  
#define txPin 10

SoftwareSerial mySerial(rxPin, txPin);  
char myChar ;

void setup() {  
 Serial.begin(9600);   
 Serial.println("AT");  
 mySerial.begin(38400);  
 mySerial.println("AT");  
}

void loop() {  
 while (mySerial.available()) {  
 myChar = mySerial.read();  
 Serial.print(myChar);  
 }

while (Serial.available()) {  
 myChar = Serial.read();  
 Serial.print(myChar);   
 mySerial.print(myChar);  
 }  
}

#include <SoftwareSerial.h>  
#define rxPin 11  
#define txPin 10

SoftwareSerial mySerial(rxPin, txPin);  
char myChar ;

void setup() {  
 Serial.begin(9600);   
 Serial.println("AT");  
 mySerial.begin(38400);  
 mySerial.println("AT");  
}

void loop() {  
 while (mySerial.available()) {  
 myChar = mySerial.read();  
 Serial.print(myChar);  
 }

while (Serial.available()) {  
 myChar = Serial.read();  
 Serial.print(myChar);   
 mySerial.print(myChar);  
 }  
}

