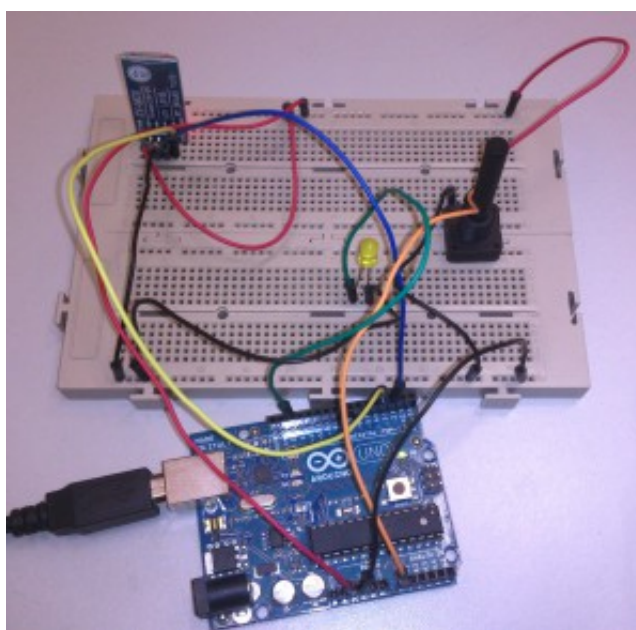


Enviar y recibir datos con módulo Bluetooth para Arduino



Hoy voy a hacer una simple explicación sobre cómo utilizar el módulo bluetooth JY-MCU BT Board V1.04 para transmitir y recibir datos con Arduino.

Módulo puerto serie Arduino Bluetooth JY-MCU



Este módulo se comunica con la placa arduino utilizando una comunicación serie, para conectarla vamos a utilizar la librería "SoftwareSerial.h" que viene incluida en el IDE de Arduino. Con esta librería podremos usar los pines 2 y 3 como RX y TX.

También necesitaremos un LED y un potenciómetro. Las conexiones son muy sencillas:

- Módulo VCC → VCC Arduino
- Módulo GND → GND Arduino

- Módulo RX → PIN 3 Arduino
- Módulo TX → PIN 2 Arduino (En caso de usar Arduino Mega o Leonardo conectar a pin 10 y editar la línea correspondiente en el programa)

El Led lo conectamos al PIN 13 y el potenciómetro al PIN ANALOG 0

ENVIAR DATOS DESDE ARDUINO Y RECIBIRLOS EN EL PC:

Vamos a enviar un valor de 0 a 1023 que obtendremos girando el potenciómetro conectado a la placa Arduino, los valores se enviarán cada segundo. Para ello utilizaremos el siguiente código:

```

1 //Enviar datos con modulo Bluetooth para Arduino http://zygzax.com/
2 #include <SoftwareSerial.h>
3 SoftwareSerial blue(2, 3);
4 int pot=0;
5 void setup(){
6     blue.begin(9600);
7     blue.println("Conectado");
8 }
9 void loop(){
10    pot=analogRead(0);
11    blue.println(pot);
12    delay(1000);
13 }
```



Para conectarnos con el módulo bluetooth desde el ordenador tendremos que añadirlo como si se tratase de un teléfono móvil o cualquier otro dispositivo bluetooth normal y corriente, utilizando la clave de emparejamiento 1234.

En el programa de desarrollo de Arduino seleccionaremos el puerto serie que corresponde a la comunicación bluetooth con este módulo, en mi caso (Sistema operativo Mac) el puerto es: "/dev/tty.HC-07-DevB".

El último paso para leer la información que está transmitiendo nuestra placa Arduino es abrir el monitor serie del programa de desarrollo de Arduino y seleccionar "9600 baud" y ya deberíamos ver la información.

ENVIAR DATOS DESDE EL PC A ARDUINO

Ahora vamos a enviar un "1" para encender el LED conectado a la placa Arduino y un "0" para apagarlo. Si enviamos una cadena de caracteres el programa leerá cada uno de ellos por orden (001101 ->

Apagado-Apagado-Encendido-Encendido-Apagado-Encendido).

El código es el siguiente:

```

1  //Recibir datos con modulo Bluetooth para Arduino http://zygzax.com/
2  #include <SoftwareSerial.h>
3  char rec;
4  SoftwareSerial blue(2,3);
5  void setup(){
6      pinMode(13,OUTPUT);
7      blue.begin(9600);
8      blue.println("Conectado");
9  }
10 void loop() {
11     if(blue.available()){
12         rec=blue.read();
13         switch(rec){
14             case '0':
15                 digitalWrite(13,LOW);
16                 delay(500);
17                 blue.println("Led off");
18                 break;
19             case '1':
20                 digitalWrite(13,HIGH);
21                 delay(500);
22                 blue.println("Led on");
23                 break;
24             default:
25                 delay(500);
26                 blue.print(rec);
27                 blue.println(" no es una orden valida. Introduzca 0 o 1");
28         }
29     }
30 }

```

El método para conectarnos a través de bluetooth con nuestra placa arduino es exactamente igual al seguido para leer la información pero en este caso no recibiremos ningún mensaje sino que escribiremos un 1 y se lo enviaremos, veremos que nos devuelve un mensaje confirmando que se ha recibido la orden y que se ha encendido el led. Para apagarlo enviaremos un 0.

Compártelo:



Esta entrada se publicó en Arduino y está etiquetada con Arduino, Bluetooth, BT Board, data, datos, enviar, JY-MCU, module, modulo, receive, recibir, send en 8 julio, 2013 [<http://zygzax.com/Enviar-y-recibir-datos-con-modulo-bluetooth-para-arduino/>] por javiersb.

32 pensamientos en "Enviar y recibir datos con módulo Bluetooth para Arduino"



Douglas

23 enero, 2015 en 15:05

no logro ver los codigos que dejas



JavierSB

Autor

23 enero, 2015 en 15:40

Hola, siento el inconveniente.

Los códigos los alojamos en Pastebin y en ocasiones el servicio se satura o se cae por problemas ajenos a nosotros. Tenemos pendiente cambiar el método para almacenar y mostrar los códigos.

Intenta acceder a través del enlace: <http://pastebin.com/Tux4sgGg>



Javier

19 enero, 2015 en 15:52

Hola soy novato en este mundo de arduino, aunque me ha enamorado y estoy leyendo mucho y aprendiendo.

Bueno entrando en mi proyecto es el siguiente: necesito comunicar dos arduino a través de bluetooth para transmitir datos de un comunicador conectado a una máquina por rs232 hacia un ordenador también conectado al puerto serie (rs232).

Ya he conseguido conectar un modulo bluetooth hc-06 en modo esclavo con éxito pero conectándolo con un móvil, aún no he probado conectar modulo con modulo porque estoy esperando que me llega un modulo bluetooth hc-05 para configurarlo como maestro ya que el hc-06 no puede hacerlo.

Una vez que consiga comunicarlo entre ellos no sé como conectarle el comunicador y el rs232 del ordenador, he pensado conectarlo a tx y rx pero no sé si conectarlo directamente o a través de un max232 para evitar conflicto de tensiones. La verdad es que estoy un poco perdido y por eso estoy

escribiendo aquí para que me puedas orientar.

Muchas gracias y si necesitas mas detalles con gusto lo doy.



michelle

25 noviembre, 2014 en 21:54

Hola disculpa es posible controlar el arduino mediante labview utilizando un hc05?, no lo he conseguido



juan fernando cadena

14 noviembre, 2014 en 01:25

hola, excelente pagina pero tengo un pregunta puedo conectar un arduino mega y un lego NXT por bluetooth solo necesito mandar un mensaje al NXT asi se solo un numero te agradecería mucho si me puedes ayudar

gracias



juan

31 julio, 2014 en 01:15

hola excelente aplicación, quisiera saber si el arduino recibiría la información que sale de una impresora térmica serial y mandarla mediante bluethoth a la pc agradecería tu comentario.



Charly

21 julio, 2014 en 22:37

Hola. Ojalà puedas ayudarme. Estoy realizando un aplicacion para controlar arduino con un android. la app ya la hice, pero al momento de enviar los comandos desde el celular el arduino no los recibe. te muestro los codigos tanto de android. Te agradezco si me ayudas.

ARDUINO LEONARDO

```
char incomingByte;

void setup() {
  pinMode(13, OUTPUT);
  Serial1.begin(9600);
}

void loop() {
  // see if there's incoming serial data:
  if (Serial.available() > 0) {
    // read the oldest byte in the serial buffer:
    incomingByte = Serial1.read();
    // if it's an r, turn the LED on then off
    if (incomingByte == 'r') {
      digitalWrite(53, HIGH);
      delay(500);
      digitalWrite(53, LOW);
    }
  }
  delay(250);
}
```

ANDROID

```
package com.example.bluetooth;

import java.io.IOException;
import java.io.InputStream;
import java.io.OutputStream;
import java.lang.reflect.Method;
import java.util.ArrayList;
import java.util.List;
import java.util.Set;
import java.util.UUID;
```

```

import android.R.string;
import android.os.Build;
import android.os.Bundle;
import android.annotation.SuppressLint;
import android.annotation.TargetApi;
import android.app.Activity;
import android.bluetooth.BluetoothAdapter;
import android.bluetooth.BluetoothDevice;
import android.bluetooth.BluetoothServerSocket;
import android.bluetooth.BluetoothSocket;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.content.IntentFilter;
import android.util.Log;
import android.view.Menu;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.AdapterView;
import android.widget.Button;
import android.widget.ListAdapter;
import android.widget.ListView;
import android.widget.TextView;
import android.widget.Toast;

//@SuppressLint("NewApi")
@TargetApi(Build.VERSION_CODES.ICE_CREAM_SANDWICH_MR1)
public class MainActivity extends Activity {

    protected static final int DISCOVERY_REQUEST = 1;
    private BluetoothAdapter Bluetooth; // BluetoothAdapter gives information about bluetooth status
    public TextView StatusUpdate;
    public Button Search;
    public Button Disconnect;
    public Button Activate;
    public Button Connect;
    public Button Send;
    public String Address;
    protected String toastText;
    public static UUID MY_UUID = UUID.fromString("00001101-0000-1000-8000-00805F9B34FB");
    public static UUID UUID2;

```

```

public BluetoothServerSocket Master;
public BluetoothSocket Slave;
public BluetoothDevice Arduino;
private OutputStream outStream = null;
private InputStream inStream = null;

```

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    SetupUI();
}

```

```

private void SetupUI() {
    // TODO Auto-generated method stub
    final TextView StatusUpdate = (TextView)findViewById(R.id.Result);

```

```

    Button Activate = (Button)findViewById(R.id.Activate);
    Button Search = (Button)findViewById(R.id.Search);
    Button Connect = (Button)findViewById(R.id.Connect);
    Button Send = (Button)findViewById(R.id.Send);
    Button Disconnect = (Button)findViewById(R.id.Disconnect);

```

```

    Bluetooth = BluetoothAdapter.getDefaultAdapter(); //Here we are knowing if the bluetooth is turned on
    or turned off
    if(Bluetooth.disable())
    {
        StatusUpdate.setText("Bluetooth is off");
    }

```

```

    Activate.setOnClickListener(new OnClickListener() {

```

```

@Override
public void onClick(View v) {

```

```

    String Discoverable = BluetoothAdapter.ACTION_REQUEST_DISCOVERABLE;
    startActivityForResult(new Intent(Discoverable),DISCOVERY_REQUEST);
    String Name = Bluetooth.getName();
    String Address = Bluetooth.getAddress();

```



```
StatusUpdate.setText( Name + "n" + Address);
}
```

```
});
```

```
Search.setOnClickListener(new OnClickListener() {
```

```
@Override
```

```
public void onClick(View v) {
```

```
if (Bluetooth.startDiscovery())
```

```
{
```

```
toastText = "Scannig for devices";
```

```
Toast.makeText(MainActivity.this, toastText, Toast.LENGTH_SHORT).show();
```

```
registerReceiver(discoveryResult, new IntentFilter(BluetoothDevice.ACTION_FOUND));
```

```
}
```

```
}
```

```
});
```

```
Connect.setOnClickListener(new OnClickListener() {
```

```
@Override
```

```
public void onClick(View v) {
```

```
// TODO Auto-generated method stub
```

```
try {
```

```
EstablishConection();
```

```
} catch (IOException e) {
```

```
// TODO Auto-generated catch block
```

```
toastText = "It was not possible to establish the connection";
```

```
Toast.makeText(MainActivity.this, toastText, Toast.LENGTH_SHORT).show();
```

```
}
```

```
}
```

```
private void EstablishConection() throws IOException {
```

```
// TODO Auto-generated method stub
```

```
if(BluetoothAdapter.checkBluetoothAddress(Address))
```

```
{
```

```
Arduino = Bluetooth.getRemoteDevice(Address);
```

```
// Get a BluetoothSocket to connect with the given BluetoothDevice
try {
    UUID2 = Arduino.getUuids()[0].getUuid();
    Slave = Arduino.createRfcommSocketToServiceRecord(UUID2);
    //Master = Bluetooth.listenUsingInsecureRfcommWithServiceRecord(null, UUID2);
    Bluetooth.cancelDiscovery();
```

```
    Slave.connect();
    //Slave = Master.accept();
    toastText = "Connection done ";
    outputStream = Slave.getOutputStream();
    inputStream = Slave.getInputStream();
    Toast.makeText(MainActivity.this, toastText, Toast.LENGTH_SHORT).show();
}
catch (IOException connectException) {
    // Unable to connect; close the socket and get out
    try {
        toastText = "Connection failed ";
        Toast.makeText(MainActivity.this, toastText, Toast.LENGTH_SHORT).show();
        Slave.close();
    } catch (IOException closeException) { }
    return;
}
}
```

```
else
{
    toastText = "Insert a correct device address";
    Toast.makeText(MainActivity.this, toastText, Toast.LENGTH_SHORT).show();
}

}
});
```

```
Send.setOnClickListener(new OnClickListener() {
```

```
@Override
public void onClick(View v) {
    // TODO Auto-generated method stub
    String on = "r";
    write(on);
```

```

}

private void write(String on) {
// TODO Auto-generated method stub
try {
byte[] Send = on.getBytes();
outStream.write(Send);
toastText = "Message send";
Toast.makeText(MainActivity.this, toastText, Toast.LENGTH_SHORT).show();
} catch (IOException e) {
toastText = "It was not possible to send the message";
Toast.makeText(MainActivity.this, toastText, Toast.LENGTH_SHORT).show();
}
}
});

```

```

Disconnect.setOnClickListener(new OnClickListener() {

```

```

@Override
public void onClick(View v) {
// TODO Auto-generated method stub
if(Slave.isConnected())
{
try {
Slave.close();
outStream.close();
inStream.close();
} catch (IOException e) {
// TODO Auto-generated catch block
e.printStackTrace();
}
}
Bluetooth.disable();
unregisterReceiver(discoveryResult);
StatusUpdate.setText("Bluetooth is off");
}

});

```

```

}

BroadcastReceiver discoveryResult = new BroadcastReceiver()

```

```

{

@Override
public void onReceive(Context context, Intent intent) {
// TODO Auto-generated method stub
String RemoteDeviceName = intent.getStringExtra(BluetoothDevice.EXTRA_NAME);
BluetoothDevice remoteDevice;
remoteDevice = intent.getParcelableExtra(BluetoothDevice.EXTRA_DEVICE);
toastText = "Discovered " + RemoteDeviceName;
Toast.makeText(MainActivity.this, toastText, Toast.LENGTH_SHORT).show();
if(remoteDevice.getName().equals("HC-05"))
{
Address = remoteDevice.getAddress();
}

}

};

protected void OnDestroy() {
super.onDestroy();
unregisterReceiver(discoveryResult);
System.exit(0);
}

}

```

**SERGIO**

18 julio, 2014 en 19:25

disculpen alguien me podría ayudar para resetear HC-05 RESULTA QUE LE EMPECE A MOVER Y PARA VERIFICAR EL CAMBIO DE VELOCIDAD LO BAJE HASTA 2400 Y DESDE ENTONCES YA NO PUEDO CAMBIAR VALORES SOLO ME RESPONDE CUANDO LE ENVIO EL COMANDO AT, DEMAS SOLO ME ENVIA ERROR (0) ESPERO SU RESPUESTA.....GRACIAS.



Javier

1 julio, 2014 en 21:18

Gracias por este Bolg. Pregunta: Es posible ocn el HC-05 transmitir y recibir datos desde una misma aplicación? Digamos qu epor ej. deseo leer un sensor ultrasónico cuando se active desde el programa arduino y tomar una decisión en un programa cirriendo en la PC par agrafizar los resultados.



javiersb Autor

5 julio, 2014 en 15:19

Hola, no entiendo muy bien tu pregunta, puedes echarle un ojo a esta entrada: [Cómo leer datos de Arduino con Matlab a través del puerto USB](#) a ver si te soluciona algo.

Un saludo,
JavierSB



miguel

27 mayo, 2014 en 18:37

holá, me gustaba hacerte una pregunta, sabes como puedo hacer lo mismo que tu codigo pero arduino + bluetooth + matlab?

gracias....Miguel Sousa....



javiersb Autor

5 julio, 2014 en 14:43

Hola, puedes ojear esta entrada: [Cómo leer datos de Arduino con Matlab a través del puerto USB](#) Ahí esta explicado como hacer una conexión para leer datos de Arduino con Matlab, tendrías que editar

las líneas en las que se especifica el puerto serie que estas utilizando, en mi caso seria: /dev/tty.usbmodem621 (En Mac, en Windows los puertos serie tienen nombres del estilo COM4) por el puerto correspondiente a la conexión serie Bluetooth que sería por ejemplo en mi caso: /dev/tty.HC-07-DevB (Debes comprobar cual sería en tu caso, no tiene por qué coincidir con el mío).

Aún no he probado a enlazar Matlab con Arduino por Bluetooth y no puedo asegurarte que esto funcione.

Un saludo y gracias por tu visita.



Javier

17 abril, 2014 en 21:23

Estimado muy buen blog, te escribo para saber si sabes si existe algun modo de transmitir un pulso analogo por bluetooth, por ejemplo el modulo bluetooth RN-52 posee dos entradas analogas...mi pregunta es si puedo conectar directamente mi señal analogica al bluetooth y transmitir el umbral, espero tu respuesta, gracias.

Los comentarios están cerrados.