

XBee Shield

Contents

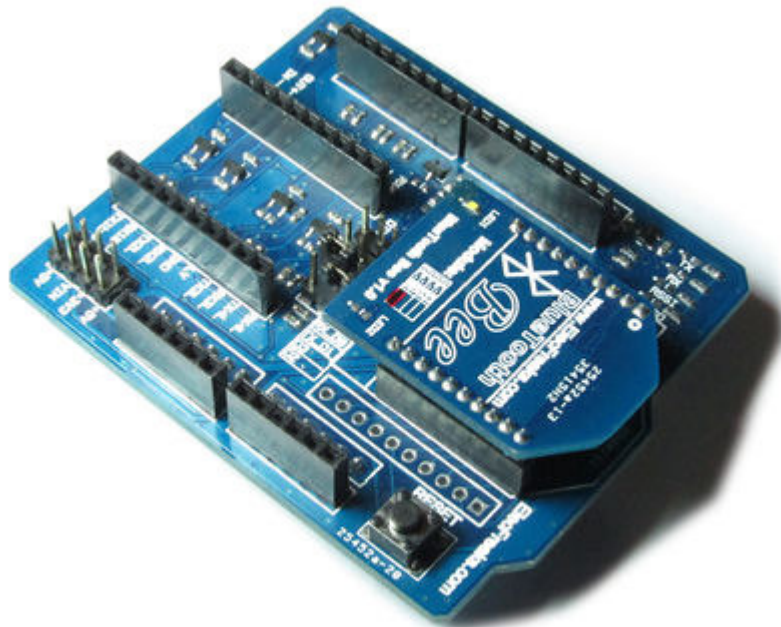
[\[hide\]](#)

- 1 Introduction
- 2 Feature
- 3 Application Ideas
- 4 Cautions
- 5 Schematic
- 6 Usage
 - 6.1 How could I sent the command to Blueooth Shield module ?
 - 6.2 How to use the Logic Level Converter ?
- 7 Bill of Materials (BOM) /parts list
- 8 FAQ
- 9 Support
- 10 Version Tracker
- 11 Bug Tracker
- 12 Resources
- 13 How to buy
- 14 See Also
- 15 Licensing
- 16 External Links

Introduction

XBee Shield with Logic Level Converter is an enhanced serial port XBee module breakout board.It can directly plug in with Arduino, and use D0-D3 of the basic board to connect with the XBee module.Its SPI and UART bus is convert the logic level 5V to 3.3V,and it's stackable, you can connect the 3.3V IO module or shield to this shield directly. The "Logic Level Converter" show in the below allowed your arduino board connect to different IO logic level module safety.

Model: [SHD08](#)

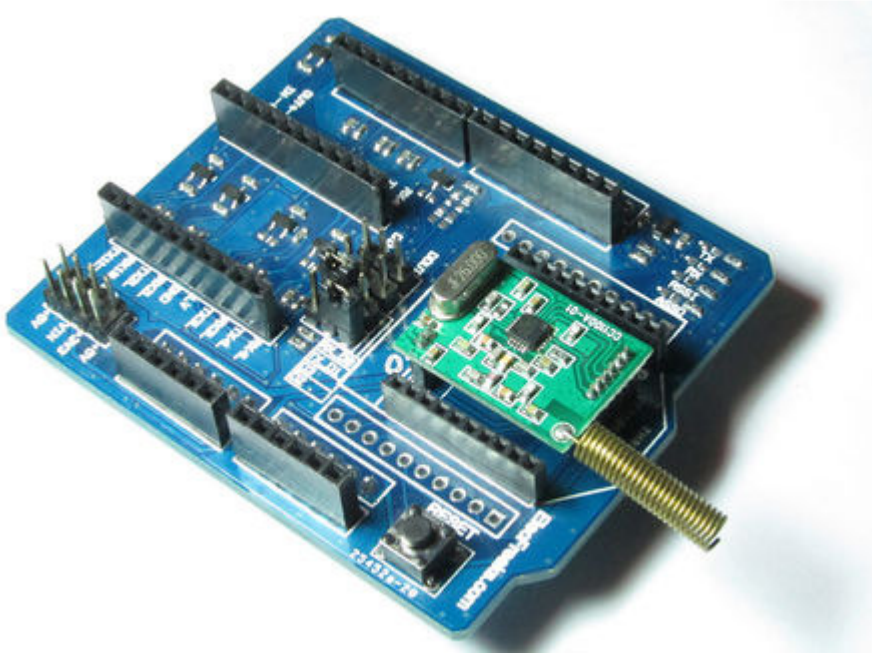


Feature

- With Logic Level Converter (3.3V to 5V or 1.8V to 5V or 1.8V to 3.3V)
- With CC1101 module connector
- Provide maximal 800mA under 3.3V
- 4 indicator LED (RSSI, ASS, Din, Dout) for XBee
- Switchable of communication with FTDI-USB /Base board
- Full 2.54mm break out for XBee
- With wireless program hardware interfaces
- Reset button for base board
- Stackable interfaces

Application Ideas

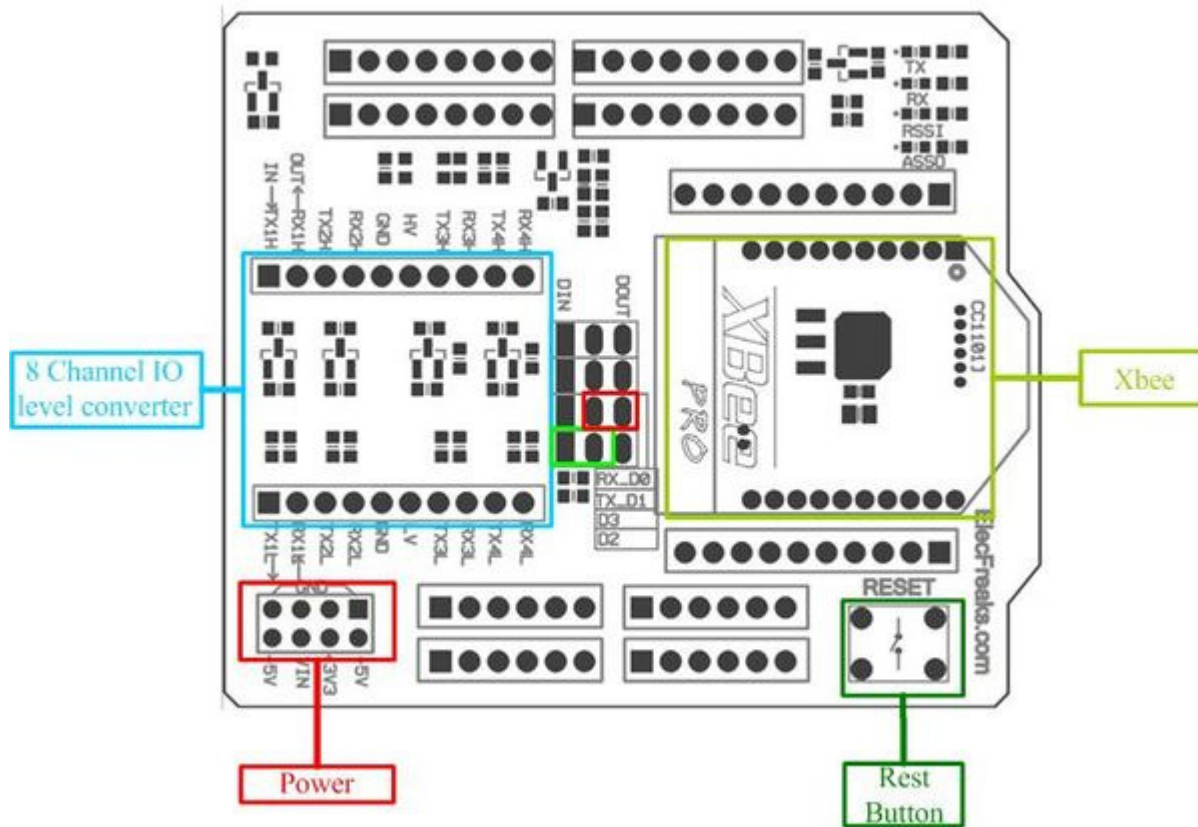
This Shield compatible **Logic Level Converter**, **XBee**, **Bluetooth Bee** and **CC1101** interface.



Cautions

- Only between master device and slave device can be paired with each other. Between Master and Master devices or Slave and Slave devices can't be paired each other. The Slave mode could be paired with PDA, PC etc.
- If use Bluetooth Bee, Please make sure your main board IO is 3.3V

Schematic



Usage

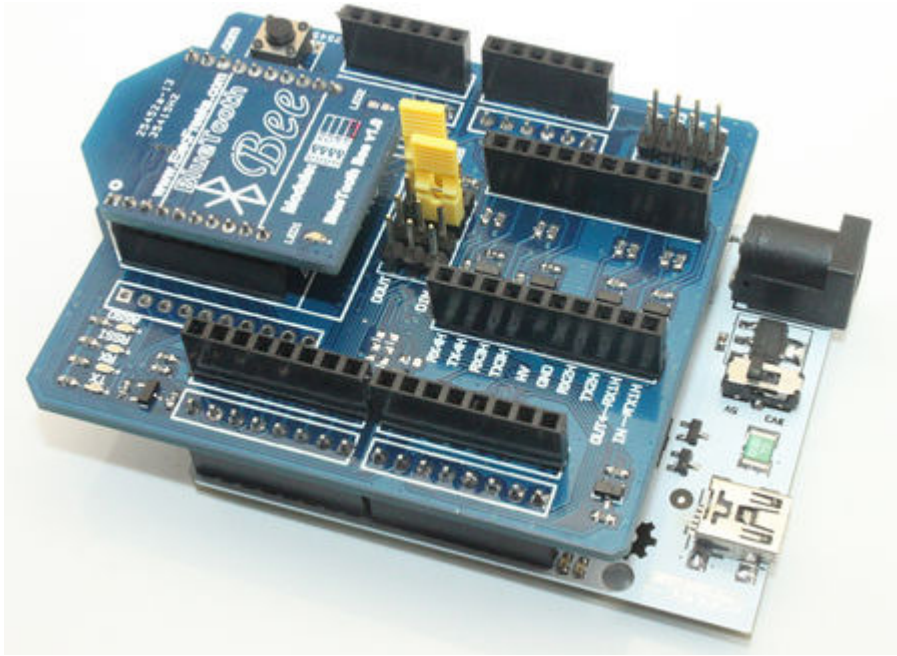
How could I sent the command to Bluetooth Shield module ?

First, you need enter AT mode in accordance with the above method. Then, there are three ways use Serial port send command to Bluetooth Shield. Detail option please reference my post [Communication between Android and Arduino with Bluetooth\(2\)](#).

- **Case 1.** Direct connect to Arduino main board Serial port, **DIN to RX_H(D0)** , **DOUT to TX_H(D1)** but it need you remove the MCU chip.
- **Case 2.** Setting the Bluetooth shield's jumper connect: **DIN to TX_H(D1)** , **DOUT to RX_H(D0)**. So this way just for you has perfect code and don't need monitor from serial port, because you can't see anything from serial port tool.
- **Case 3.** use Software Serial library to define D2 and D3 as a virtual serial port TX and RX. By this way, you can use Arduino's Serial Monitor to Monitor operation of the code.

Connect as **DOUT to D3** , **DIN to D2**

Here we used **case 3** for the demo code with [Freaduino v1.0 with ATmega328 \(100% Arduino compatible\)](#).



Usually we used software Serialport library to control and communication with Bluetooth Module. The Preparation steps :

Step One: Hardware connect

Use the Case 3 connection. And we jumper cap to SW side and set switch to H side. Then use Software Serial library to define D2 and D3(Of course others) as a virtual serial port TX and RX. By this way, you can use Arduino's Serial Monitor to Monitor operation of the code.

Step two: Software control

First of all, you need a software library. There we used NewSoftSerial library. Get it from [here](#)

Put the library to : ...\\your_ArduinoIDE_Path\\arduino-0022\\libraries

Copy below demo code and run. This just a sample code for your reference and you can modified and change it whatever you want.

Step Three: AT command

Now, You can send AT command from Serial port to control and setting the Bluetooth module. And you can set the module to Master or Slave.....

You can play with Master mode to send paired require to others Bluetooth devices or Slave mode to wait others Bluetooth devices discover.

The AT command please reference:

- [Bluetooth Bee AT Command\(HC-05\)](#)

- [Bluetooth Bee AT Command\(HC-06\).](#)

Includes important code snippet. Demo code like :

```
/******  
**  Description:                                     **  
**  This file is a sample code for your reference.   **  
**                                                    **  
**  Copyright (C) 2011 ElecFreaks Corp.             **  
**  Created by ElecFreaks Robi.W /29 Sep 2011       **  
**                                                    **  
**  http://www.electfreaks.com                     **  
*****/  
#include <NewSoftSerial.h>  
#include <TimerOne.h>  
  
#define rxPin 2  
#define txPin 3  
//#define KEY 11 for HC-05 Bluetooth Module  
  
NewSoftSerial mySerial(rxPin, txPin);  
  
void Callback()  
{  
    Serial.println("-----> Callback Send AT");  
    mySerial.print("AT\r\n");  
}  
  
void setup()  
{  
    //pinMode(KEY, OUTPUT);  
    //digitalWrite(KEY, HIGH); // if used HC-05 need set the Bluetooth Bee's pin20(HC-05's  
pin34) to HIGH level to enter AT mode.  
  
    // define pin modes for tx, rx, led pins:  
    pinMode(rxPin, INPUT);  
    pinMode(txPin, OUTPUT);  
    mySerial.begin(9600);  
    Serial.begin(9600);  
  
    //Timer1.initialize(2000000); // setting callback is 2s  
    //Timer1.attachInterrupt(Callback);  
    Serial.println("-----> Start ");  
}  
  
void loop()  
{  
    int i = 0;
```

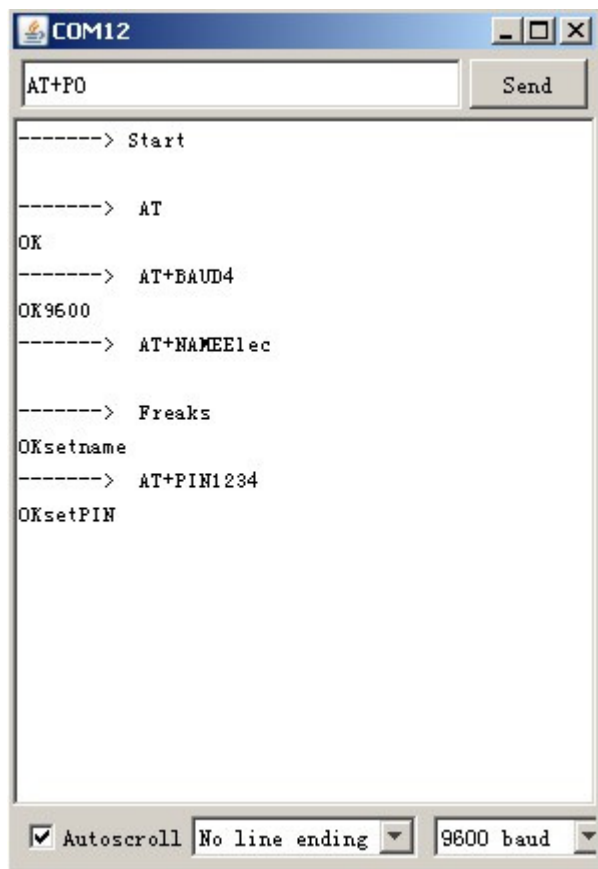
```

char someChar[32] = {0};
// when characters arrive over the serial port...
if(Serial.available()) {
  Serial.println();
  do{
    someChar[i++] = Serial.read();          // Note if use HC-06 there is not '/r/n' in the
end.
    //As data trickles in from your serial port you are grabbing as much as you can,
    //but then when it runs out (as it will after a few bytes because the processor
    //is much faster than a 9600 baud device) you exit loop, which then restarts,
    //and resets i to zero, and someChar to an empty array. So please be sure to keep this
delay
    delay(3);
  }while (Serial.available() > 0);

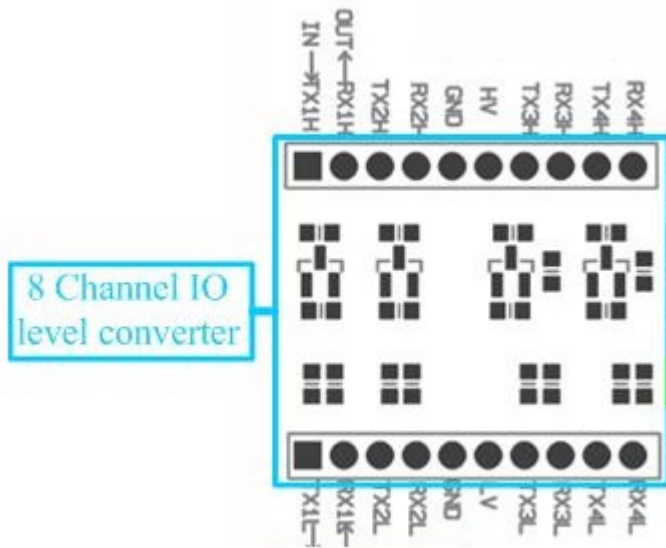
  mySerial.println(someChar);
  Serial.println(someChar);
}

while(mySerial.available())
  Serial.print((char)mySerial.read());
}

```

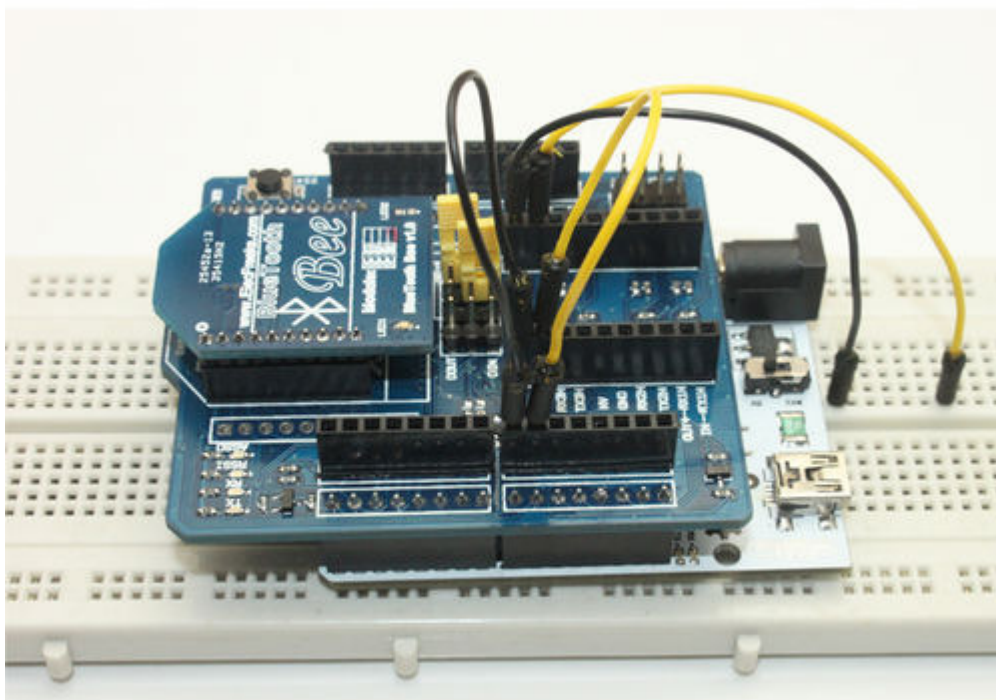


How to use the Logic Level Converter ?



The Logic Level Converter there are two side, and need supply power to HV and LV. That means the IO level can get to HV voltage level and LV voltage level. Usually the **HV connect to 5V** and **LV connect to 3.3V** or **HV connect to 3.3V** and **LV connect to 1.8V**

You need use jumper wires to HV or LV side TX or RX pin (Note this not Serial port TX and RX, just send data and receive data IO), and get the other Logic voltage level.



Version Tracker

Revision	Descriptions	Release
v1.0	Initial public release	date

Resources

- [NewSoftSerial Library](#)
- [Schematic](#)
- [AT Command](#)
- [Android chat Arduino by Bluetooth Bee](#)