

GET STARTED TO WORK WITH
XBEE
AND
ADAFRUIT DOSTAR

STEP 1 :- COLLECT THE HARDWARE

About Xbee:


- 2+ Xbee with [XBee Shield for Arduino](#) or [Arduino Wireless XBee shield](#)
- install [Xtcu](#) - use this to configure Xbee

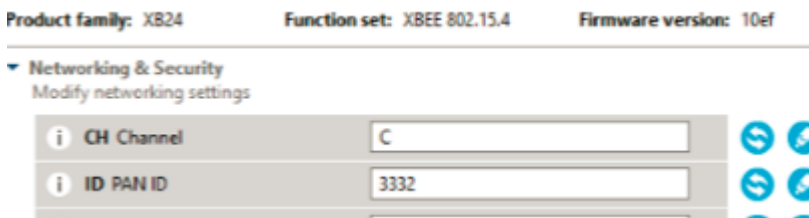
About DotStar:

- [power adapter for the DotStar](#)
- a few jumper wires , red | black | green | yellow
- breadboard

Get two Arduino, of course. Uno is fine.

STEP 2: CONFIGURE THE XBEE MODULES

1. Need to [download the Xtcu software](#).
2. Configure their unique IDs for communication.
3. Snap one Xbee module onto the XBeeshield hooked up with an Arduino
4. At the Xtcu software:
 - a. Add a radio module - the Xbee module thru this: 
 - b. Select the proper com port
 - c. Setup unique channel and pan id thru:
 - d. Working Modes - configuration working mode



5. Done with one xbee.
6. Repeat 2 to 4.d for the 2nd xbee module

STEP 3 : TEST TO ENSURE CONFIGURATION WORKS

For testing, you will need designate at least :

- One transmitter controller
- One receiver controller

TEST 1 - BASIC ECHO TEST USING THE DIGI PROGRAMMABLE BOARD

A) Bare Basic Echo Server

```
//Bare Basic Echo Server code : wait for incoming data and echo it back
int snd = 0;

void setup() {
  Serial.begin(9600);
  Serial.println("Echo-Starting....");
}

void loop() {

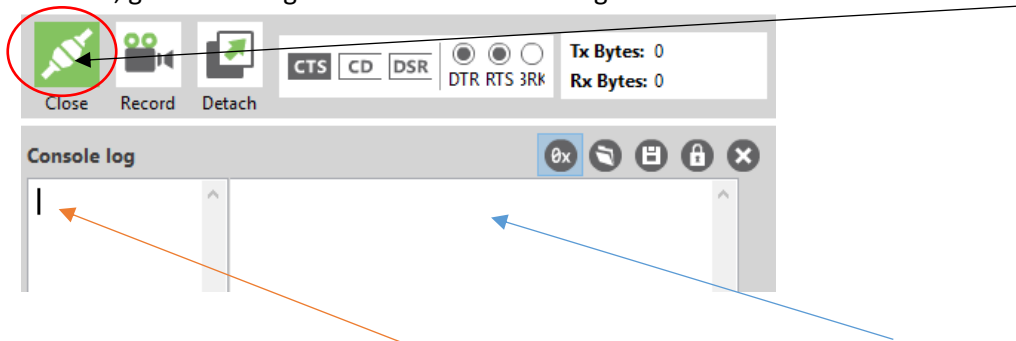
  if ( Serial.available() > 0)
    Serial.write( Serial.read( ) );

}
```

B) Echo Receiver

B.1) With the Digi Programmable board

1. I use [theDigi International XBIB-U-DEV](#) as the transmitter . Thus I don't even need to write the ode for this echo test because it will be like simple typing right at your computer console. To do this:
2. At Xtcu , go to Working Modes – console working mode. Make sure it is connected



Then, simply hitting keys at the console, you should see each being echoed back .

B.2) Instead of the programmable board, using Another Arduino with an already configured XBee module

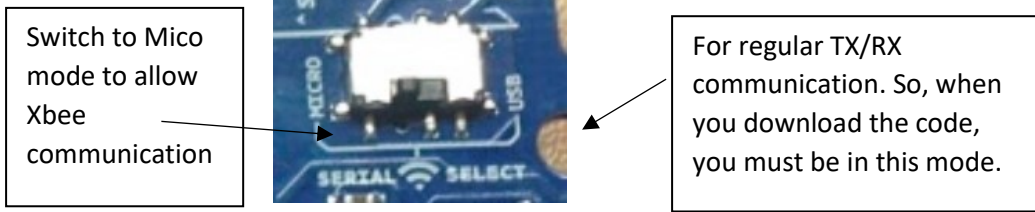
```
int LED = 13;
int snd = 0;

void setup() {
  pinMode(LED, OUTPUT);
  Serial.begin(9600);
  Serial.println("RX-Starting....");
}

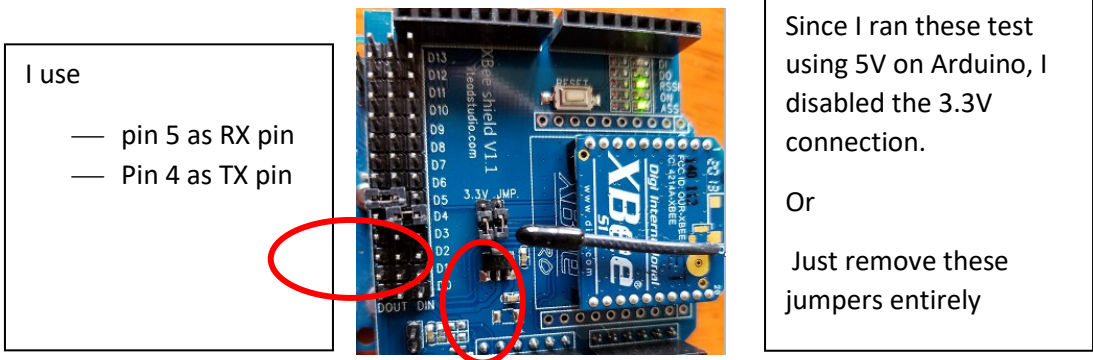
void loop() {
  snd = (++snd)%5;
  Serial.write ( snd ); delay(10);
  if ( Serial.available() > 0)
  {
    int ch = Serial.read();
    for (int i=0; i<ch; i++)
    {
      digitalWrite(LED, HIGH); // turn the LED on
      delay(500);
    }
    Delay(2000);
  }
}
```

IMPORTANT HARDWARE SETUP:

- A) If you use the [XBee Shield from Itead Studio](#), you need to make sure your jumpers are set properly – see [the specification document](#).
If you use the Arduino Wireless SD Shield module, Make sure you have toggle from usb to micro connection on the xbee shield.



- B) If you use the Arduino Xbee Shield:



TEST 2: WRITE YOUR OWN TRANSMITTER AND RECEIVER USING SOFTWARE SERIAL

- XBee Sender
- Send 1 byte via software serial
- Get the byte back from echo server
- Display the data to the hardware serial monitor.

- XBee Echo Server
- Receive 1 byte via Hardware Serial
- Send it back via Hardware Serial

```
// Sender
#include <SoftwareSerial.h>

int irx = 5, itx = 4;
SoftwareSerial Ss( irx, itx);

void setup() {
  Ss.begin(9600);
  Serial.begin(9600);
  Serial.println("TX-Starting....");
}

void loop()
{
  for (int i=0; i<20; i++)
    Ss.write(i);
  if ( Ss.available() > 0 )
  {
    Serial.print( Ss.read());
    Serial.print(" ");
  }
  delay(1000);
}
```

```
//Echo Server Bare Basic Sample

void setup() {
  Serial.begin(9600);
  Serial.println("Echo-Starting....");
}

void loop() {
  if ( Serial.available() > 0)
    Serial.write( Serial.read()+1);
}
```