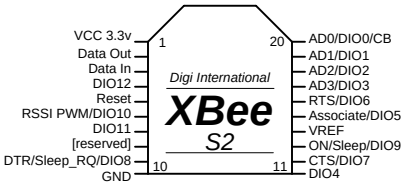
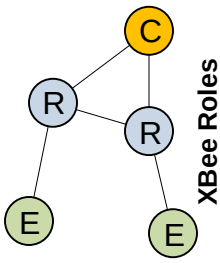
XBee S2 Quick Reference Guide



IEEE 802.15.4 = Zigbee Protocol. XBee is a microcontroller made by digi which uses the Zigbee protocol.

The XBee uses 3.3V and has a smaller pin spacing than most breadboards/proto boards. Because of this, it is often useful to purchase a kit to interface the XBee with a breadboard.

Sept/2012 http://tunnelsup.com



XBee Roles

- Coordinator – 1 required in every network
  In charge of setting up the network
  Can never sleep
- Router – multiple may exist
  Can relay signals from other routers/EPs
  Can never sleep
- End Point – multiple may exist
  Cannot relay signals
  Can sleep to save power

Specs

Operating Voltage: 2.1 – 3.6V	Digital I/O pins: 11	RF Data Rate: 250kbps
Operating Current: 40mA@3.3V	Analog input pins: 4	Throughput speed: 35kbps
Indoor range: 40 Meters	Mesh routable	Frequency: ISM 2.4GHz
Line of sight range: 120 Meters	Self Healing network	OK Temp: -40 to 85C
Max Analog Pin Reading: 1.2V	Firmware: ZB ZigBee	

XBee Modes

Transparent – Communication through the XBee. If data is not generated from the XBee itself then both XBees should be set to AT.
Command – Communication to the XBee. If one XBee is sensing data, that XBee should be in AT mode while the receiving one should be in API mode.

XBee Setup

Connect the XBee to a TTL Serial FTDI adapter – OR – Arduino hack: Connect RX to RX, TX to TX, RESET to ground to bypass the Arduino entirely and get serial to XBee.
Use the free X-CTU software to configure the XBee.
Baud: 9600 – FC: Hardware – Data Bits: 8 – Parity: None – Stop Bits: 1

Basic Settings

PAN ID – The network to communicate over. If 0, the XBee will join any.
DH/DL – Destination Serial number. Used to send to a specific XBee's Serial. Set to 0 to send to just the Coorindator. Set to 0x0000000000FFFF to broadcast.
JV – Router/EP should be set to 1 so it rejoins the network on startup

Pin Settings

For pin settings to work, receiver XBee must be in API mode
D0 – Set pin 0 to start sensing
IR – Collect data on sensing pins every XX millisecs

Arduino Connectivity:
Arduino TX connects to XBee RX (Data in)
Arduino RX connects to XBee TX (Data out)

Arduino Integration:
Data sent to Serial.print() will go out TX port of Arduino which is then connected to the RX port of XBee. If XBee is in AT mode it will transmit it wirelessly. Data received from XBee will be sent to the Serial.

Arduino Example: Read an analog value using API

```
// Remote XBee: AT, Base XBee: API
if (Serial.available() >= 21) {   // Make sure the frame is all there
 if (Serial.read() == 0x7E) {     // 7E is the start byte
  for (int i = 1; i<19; i++) {    // Skip ahead to the analog data
    byte discardByte = Serial.read();
  }
  int analogMSB = Serial.read(); // Read the first analog byte data
  int analogLSB = Serial.read();  // Read the second byte
  int analogReading = analogLSB + (analogMSB * 256);
 }
}
```

API format for Remote AT Command Request

Byte	Example	Description
0	0x7e	Start byte – Indicates beginning of data frame
1	0x00	Length – Number of bytes (ChecksumByte# – 1 – 2)
2	0x10	
3	0x17	Frame type - 0x17 means this is a AT command Request
4	0x52	Frame ID – Command sequence number
5	0x00	64-bit Destination Address (Serial Number)
6	0x13	MSB is byte 5, LSB is byte 12
7	0xA2	
8	0x00	0x0000000000000000 = Coordinator
9	0x40	0x000000000000FFFF = Broadcast
10	0x77	
11	0x9C	
12	0x49	
13	0xFF	Destination Network Address
14	0xFE	(Set to 0xFFFE to send a broadcast)
15	0x02	Remote command options (set to 0x02 to apply changes)
16	0x44 (D)	AT Command Name (Two ASCII characters)
17	0x02 (2)	
18	0x04	Command Parameter (queries if not present)
19	0XF5	Checksum

Arduino Example: Change the pin setting on a remote Xbee

```
// Remote XBee: AT, Base XBee: API
Serial.write(0x7E);  // Sync up the start byte
Serial.write((byte)0x0);    // Length MSB (always 0)
Serial.write(0x10);  // Length LSB
Serial.write(0x17);  // 0x17 is the frame ID for sending an AT command
Serial.write((byte)0x0);    // Frame ID (no reply needed)
Serial.write((byte)00);     // Send the 64 bit destination address
Serial.write((byte)00);     // (Sending 0x000000000000FFFF (broadcast))
Serial.write((byte)00);
Serial.write((byte)00);
Serial.write((byte)00);
Serial.write((byte)00);
Serial.write(0xFF);
Serial.write(0xFF);
Serial.write(0xFF); // Destination Network
Serial.write(0xFE); // (Set to 0xFFFE if unknown)
Serial.write(0x02); // Set to 0x02 to apply these changes
Serial.write('D');    // AT Command: D1
Serial.write('1');
Serial.write(0x05); // Set D1 to be 5 (Digital Out HIGH)
long chexsum = 0x17 + 0xFF + 0xFF +0xFF + 0xFE + 0x02 + 'D' + '1' + 0x05;
Serial.write( 0xFF – (chexsum & 0xFF)); // Checksum
```

API format for I/O Data Sample RX Indicaton

Byte	Example	Description
0	0x7e	Start byte – Indicates beginning of data frame
1	0x00	Length – Number of bytes (ChecksumByte# – 1 – 2)
2	0x14	
3	0x92	Frame type - 0x92 indicates this will be a data sample
4	0x00	64-bit Source Address (Serial Number)
5	0x13	MSB is byte 4, LSB is byte 11
6	0xA2	
7	0x00	
8	0x40	
9	0x77	
10	0x9C	
11	0x49	
12	0x36	Source Network Address – 16 Bit
13	0x6A	
14	0x01	Receive Opts. 01=Packet Acknowledged. 02=Broadcast packet
15	0x01	Number of sample sets. Always set to 1 due to XBEE limitations
16	0x00	Digital Channel Mask – Indicates which pins are set to DIO
17	0x20	
18	0x01	Analog Channel Mask – Indicates which pins are set to ADC
19	0x00	Digital Sample Data (if any) – Reads the same as Digital Mask
20	0x14	
21	0x04	Analog Sample data (if any)
22	0x25	There will be two bytes here for every pin set for ADC
23	0xF5	Checksum(0xFF - the 8 bit sum of the bytes from byte 3 to this byte)

Sleep Mode

Endpoints can sleep to save power. An endpoint that only wakes up every 5 minutes to send data may only be awake for 6 seconds a day.
SM – 4 = Cyclic sleep
SP – Sleep time (up to 28 secs)
SN – Number of sleep cycles
ST – Time awake

Pin I/O Options

0 – Disabled
1 – N/A
2 – ADC
3 – Digital IN
4 – Digital OUT, LOW
5 – Digital OUT, HIGH

Digital Ch Mask

First Byte
n/a n/a n/a D12 D11 D10 n/a n/a
Second Byte
D7 D6 D5 D4 D3 D2 D1 D0
Example:
0x00 0x13 = 0000 0000 0000 1101
Pins D3, D2 and D0

Analog Ch Mask

(volt) n/a n/a n/a A3 A2 A1 A0
Example:
0x05 = 0000 0101 = Pin A2 and A0