KIET IJCE KIET International Journal of Communications & Electronics
Volume. No. 2, Issue No. 1, Jan – April 2014, ISSN: 2320 - 8996

# Loopback Delay Analysis by transmitting and receiving Data Packet of Four Characters for Zigbee Devices

*Preeti Sohni*
*Asst. Prof., Electronics and Communication Department, Sushila Devi Bansal College of Engineering, Indore, Madhya Pradesh, India*
*preetisohni1980@gmail.com*

***Abstact*-. This Paper discuss about variation in roundtrip journey of signal when distance is varied. An experimental analysis has been done by transmitting a localized standard communication packets as transmit signals. The objective of this analysis is to show when we increase distance, we should decrease packet transfer rate. Because delay(time) is inversely proportional to speed. An experimental analysis has been done by keeping one Xbee modem stationary(transmitter module), who will transmit a string of four character of "test" and this string is received by another Xbee modem which is placed at uniform varying distance.And also a mill sec timer is intialized by embedded software who will record loopback time, when string is transmitted back to transmiter module from recevier.This setup is used to record loopback time for distance in feet (10, 20, 30, 50, 60, 80, 100) with some obstacles or with line of sight position of these two modules.Arduino MCU is used with transmitter module in order to intializing calculation of loopback time.**

***Keywords* — Zigbee, Xbee modem, IEEE 802.15.4, Arduino, X-CTU, loopback time**

## I. INTRODUCTION

IEEE 802.15.4, commonly known as ZigBee, is a Media Access Control (MAC) and physical layer standard specifically designed for short range wireless communication where low rate, low power, and low bandwidth are required.

ZigBee[2] is a low-power communication system using digital radios. It's intended to be easier to work with than Bluetooth. Using two Xbee radios and USB Xplorer break out board, we can wirelessly monitor sensor.

This paper proposes how such monitoring system can be setup and emphasizing on the aspects of low cost, easy ad hoc installation and easy handling and maintenance. The paper shows xbee modem configuration details and power consumption for a complete end device. Xbee modem is chosen due to its features that fulfill customization.

This makes ZigBee an ideal choice when it comes to sensor networks for monitoring data collection and/or triggering process responses.

This paper shows how one model having two xbee radios, one MCU(Arduino), and USB Xplorer breakout board establish wireless sensor monitoring system, Fig 1 is block diagram representation for system and Fig 3 is Hardware view.

Several preliminary results of measurement to evaluate the reliability and effectiveness of the system are the requirement for a low cost, easy to use, minimal also presented.

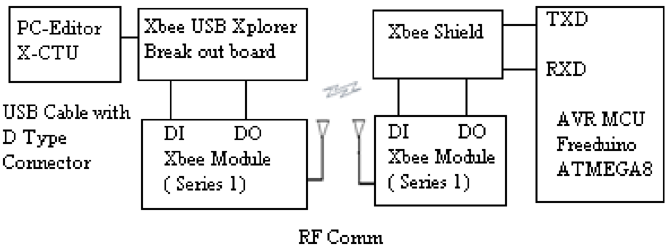
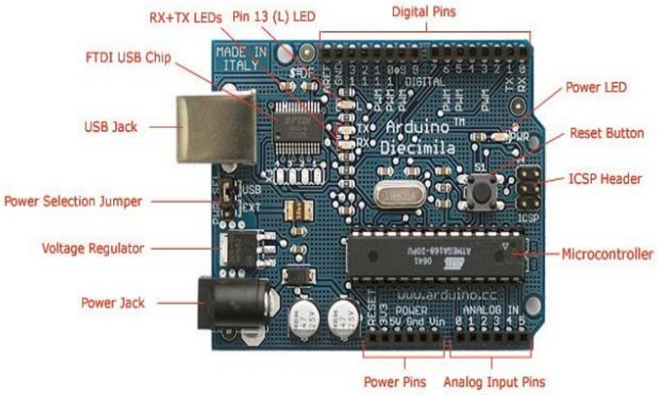


Fig 1: Block Diagram of Wireless comm between PC and MCU using Xbee Modem

## II. Arduino

### A. *What is Arduino [10]*

Arduino is an open-source electronics prototyping platform based on flexible, easy-to-use hardware and software. It's intended for artists, designers, hobbyists, and anyone interested in creating interactive objects or environments.



### B. *Details of Board [11]*

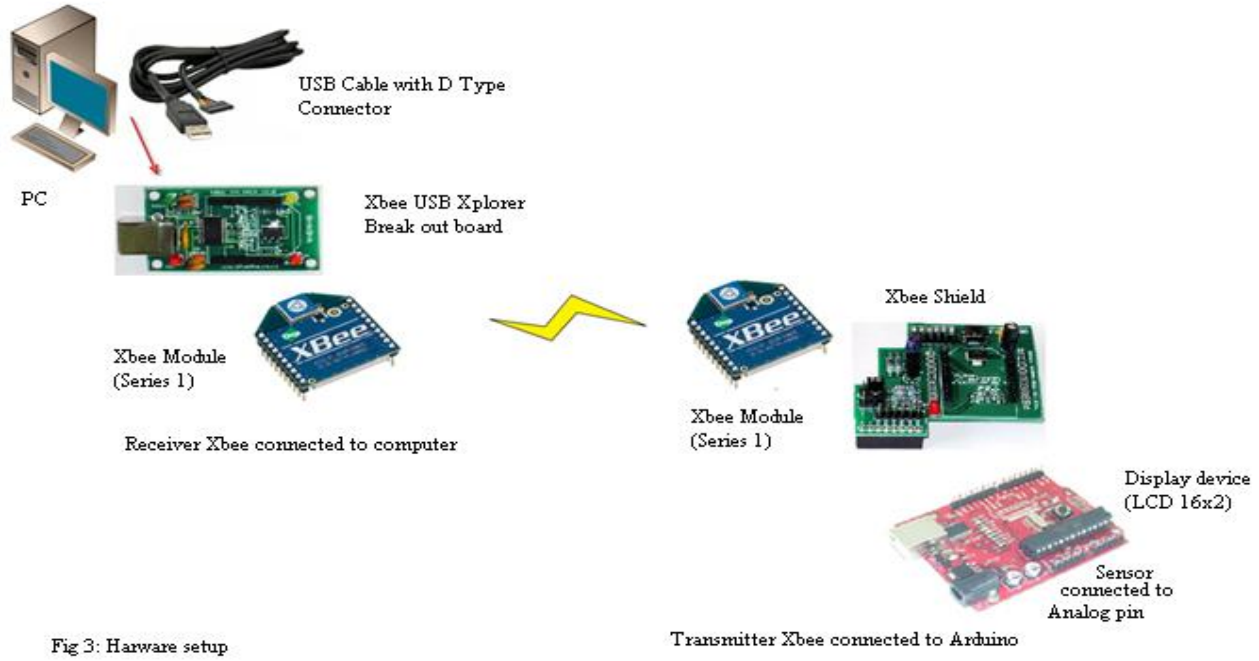

USB Cable with D Type Connector
PC
Xbee USB Xplorer Break out board
Xbee Module (Series 1)
Receiver Xbee connected to computer
Xbee Shield
Xbee Module (Series 1)
Display device (LCD 16x2)
Sensor connected to Analog pin
Transmitter Xbee connected to Arduino

Fig 3: Harware setup

The Arduino Duemilanove ("2009") is a microcontroller board based on the ATmega168 ( or ATmega328) . It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

The Arduino Duemilanove can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The ATmega168 has 16 KB of flash memory for storing code (of which 2 KB is used for the bootloader). Each of the 14 digital pins on the Duemilanove can be used as an input or output, using pinMode(), digitalWrite(), and digitalRead() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. 0 (RX) and 1 (TX) pin used to receive (RX) and transmit (TX) TTL serial data respectively. These pins are connected to the corresponding pins of the FTDI USB-to-TTL Serial chip.

Pin 3, 5, 6, 9, 10, and 11can provide 8-bit PWM output with the analogWrite() function.

## III. XBEE MODEM

### A. *What is Xbee Modem [6]*

The XBee modules by Digi are small radios that can communicate across hundreds of feet. They include serial interfaces as well as inputs and outputs (9 ports, 5 of which can be analog inputs) for sensors and effectors. They are low cost, low power, and in many cases can be used without any microcontroller. Once set up, they are very easy to use and very reliable.

### B. *Operating Modes*

There are two basic modes that the XBee's can operate in: Transparent (AT) and API (Packet) Mode.

*1) Transparent (AT) Mode* : If you want to create a simple point-to-point configuration, where the Xbee acts as a wireless serial modem between a computer or microcontroller and a remote device (e.g. Arduino) using simple serial communications, you want to set up the XBee in "Transparent Mode".

Features of the Transparent mode:

- Simple
- Compatible with any device that speaks serial
- Limited to point to point communication between two XBees

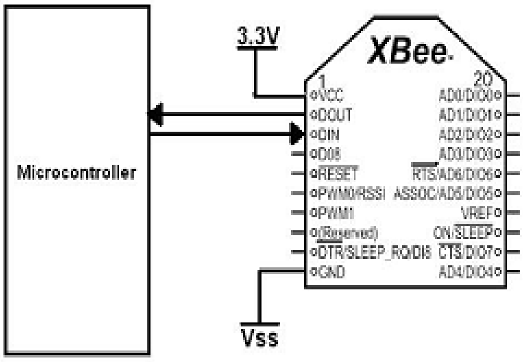
*2) API (Packet) (Application Programming Interface) Mode* : The XBee in API mode, offers more capabilities. In particular, the API mode enables you to set up a network of XBees and communicate to each one individually.

Features of the API mode:

- I/O Line passing (receive data from a stand-alone remote XBee)
- Allows for Broadcast communication and communication with more than one XBee
- Receive acknowledgement that a packet was successfully delivered
- Obtain RSSI (signal strength)
- Remote Configuration

### C. XB24-AWI-001[6]

The XBee RF Modules were engineered to meet IEEE 802.15.4 standards and support the unique needs of low-cost, low-power wireless sensor networks. The modules require minimal power and provide reliable delivery of data between devices. The modules operate within the ISM 2.4 GHz frequency band.



**Typical Microcontroller Interfacing to the XBee**

### D. Key Features : [6]

1) *Long Range Data Integrity :*
- Indoor/Urban: up to 100’ (30 m)
- Outdoor line-of-sight: up to 300’ (90 m)
- Transmit Power: 1 mW (0 dBm)
- Receiver Sensitivity: -92 dBm

2) *RF Data Rate: 250,000 bps*

3) *Advanced Networking & Security*

- *Retries and Acknowledgements*
- *DSSS (Direct Sequence Spread Spectrum)*
- *Each direct sequence channels has over 65,000 unique network addresses available*
- *Source/Destination Addressing*
- *Unicast & Broadcast Communications*
- *Point-to-point, point-to-multipoint and peer-to-peer topologies supported*

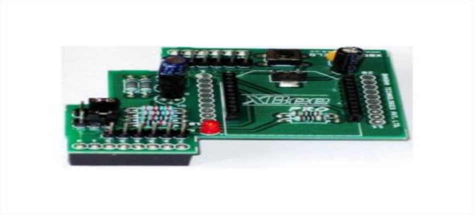
4) *Low Power*
- TX Peak Current: 45 mA (@3.3 V)
- RX Current: 50 mA (@3.3 V)
- Power-down Current: < 10 µA

5) *ADC and I/O line support :* Analog-to-digital conversion, Digital I/O, I/O Line Passing

## IV. XBEE SHIELD

### A. What is Xbee Shield [9 ]

The XBee shield is intended for the use of Maxstream XBee modules to wireless transmit data using the serial port, or for transmitting data from / to the Xbee IO-pins.The Arduino Xbee shield allows Arduino board to communicate wirelessly using Xbee module.



The shields breaks out each of the Xbee's pins to a through-hole solder pad. It also provides female pin headers for use of digital pins 2 to 7 and the analog inputs, which are covered by the shield (digital pins 8 to 13 are not obstructed by the shield, so you can use the headers on the board itself).

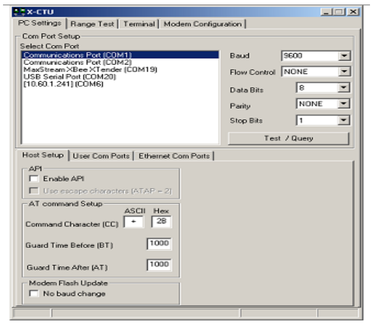
### B. Overview

The Xbee shield allows an Arduino board to communicate wirelessly using Zigbee. It is based on the Xbee module from MaxStream. The module can communicate up to 100 feet indoors or 300 feet outdoors (with line-of-sight). It can be used as a serial/usb replacement or you can put it into a command mode and configure it for a variety of broadcast and mesh networking options.

## V. X-CTU SOFTWARE (Configuration and test utility software)

### A. What is X-CTU [7]

X-CTU is a Windows-based application provided by Digi. This program was designed to interact with the firmware files found on Digi provide a simple-to-use graphical user interface to them.



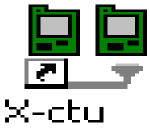
Com Port Setup
Select Com Port
Communications Port (COM1)
Communications Port (COM2)
MaxStream XBee XTender (COM19)
USB Serial Port (COM20)
[10.60.1.241] (COM6)

Baud 9600
Flow Control NONE
Data Bits 8
Parity NONE
Stop Bits 1

API
□ Enable API
□ Use escape characters (ATAP = 2)

AT command Setup
Command Character (CC) ASCII + Hex 2B
Guard Time Before (BT) 1000
Guard Time After (AT) 1000

Modem Flash Update
□ No baud change



X-CTU Window X-CTU Icon

### B. Why is X-CTU needed?

With the help of this software, we can perform following things
- To know COM Port
- To know Modem Type
- To know Modem Firmware version
- Reading a radio firmware
- Making changes to a radios firmware
- Writing firmware to the Radio

- Downloading Updated Firmware Files
- Modem can configure into AT or API mode
- It Provides Terminal Emulator
- To set Networking Parameter (PAN Id) and channel
- To set Addressing Parameters( Source and destination address)
- To set Serial interfacing parameter (BaudRate)

## VI. XBEE SETUP AND RESOURCES

### *A. What we need to setup Xbee radio before use them*

- Computer running Windows XP
- FTDI Driver for XBee Explorer USB
- X-CTU software by Digi
- XBee Explorer USB break out board (allows you to attach the XBee to your computer via USB)

### *B. Configuration for the Coordinator (local) XBee*

The Coordinator XBee is the module that's connected to your computer and it is the master unit. It can receive data from any remote XBee, as well as tell individual remote XBees how to act (in API mode).

*1) Networking & Security Section*: ID=3332 (the default) This is called the PAN ID, and establishes a common network among all the XBees - Every XBee in your network must have the same number for the PAN ID (you can set it differently from 3332 if you want). DH=0, DL=0, MY=1234,
CE=1 (Coordinator)

*2) Serial Interfacing Section* : BD=3 for 9600 baud, AP=0 API Disable

### *C. Configuration for End-Device (remote) XBee*

*1) Networking & Security Section:* ID=3332, DH=0, DL=1234, MY=1 - This is the identifier for the remote XBee, CE=0 (End-Device)

*2) Serial Interfacing Section :* BD=3 for 9600 baud rate, AP=0API Disable

## VII. EXPERIMENTAL RESULTS

### *A. Analysis 1 of XBEE Modem for loopback time versus distance*

In order to observe loopback time, things have been done, are follows:

1. Written a code in embedded C which transmit a string like "test"(contain four characters) from transmitter module , and just before that I am capturing transmitting time from milli sec timer and stored into one variable.
2. Written a code into Microsoft Visual Basic , which open port and make all required setting like baud rate etc. and receive a string "test". And as soon as it receive , it transmit back the received sting.
3. Transmitter module received this string and compare with transmitting sting and capture received time again from milli sec timer ,Which runs continues from the time of transmission.
4. And calculate loopback time by subtracting transmitting time from received time.

Table 1 shows results ( Modules (both transmitter and receiver are kept in corridor means indoor )
10 readings have been taken for each distance point and then calculated average of those reading.

Table 1

| S. No. | Distance in feet | Loopback time in ms | Remark |
|---|---|---|---|
| 1. | 10 | 400 to 600 | Readings come in values like 437, 875, 567,534………….. |
| 2. | 20 | 400 to 600 | Not remarkable changes |
| 3. | 30 | 500 to 700 | Out of ten reading most of values are like 676,716,449,758………….. |
| 4. | 50 | 700 to 800 | Values are like 799,1006,799,848,826…. |
| 5. | 60 | | Not Remarkable changes |
| 6. | 80 | | Readings comes like 800, 987,1054, 456,967… |
| 7. | 100 | Upto 1000 | Values are like 1008,1600,1004,989,1210,1101… |

### B. Analysis 2 of XBEE Modem for loopback time versus distance with different settings

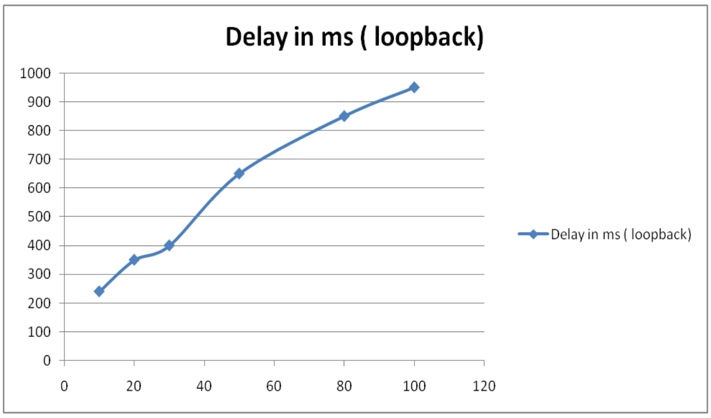
In this analysis the role of PC based module and Embedded module is changed. Now packet transfer is initiated from PC based module and Embedded module is retransmit back the packet as soon as it received. 10 or 12 readings are taken for each distance point and saved as text file showing readings for 10 feet distance, 30 feet distance, 50 feet distance, 80 feet and 100 feet distance.

Table 2

| S.No. | Distance in feet (in metre) | Average of 10 readings Loopback time (in ms) | Readings |
|---|---|---|---|
| 1. | 10 feet (3.048) | 240 | time_loopback:309<br>time_loopback:489<br>time_loopback:57<br>time_loopback:338<br>time_loopback:107<br>time_loopback:107<br>time_loopback:268<br>time_loopback:261<br>time_loopback:341 |

| | | | |
|---|---|---|---|
| | | | time_loopback:106<br>time_loopback:248 |
| 2. | 20 feet (6.096 ) | 350 | time_loopback:447<br>time_loopback:333<br>time_loopback:226<br>time_loopback:210<br>time_loopback:468<br>time_loopback:302<br>time_loopback:493<br>time_loopback:441<br>time_loopback:362<br>time_loopback:335<br>time_loopback:293 |
| 3. | 30 feet ( 9.144 ) | 350 to 400 | Readings are same |
| 4. | 50 feet ( 15.24 ) | 650 | time_loopback:447<br>time_loopback:333<br>time_loopback:626<br>time_loopback:710<br>time_loopback:668<br>time_loopback:602<br>time_loopback:793<br>time_loopback:741<br>time_loopback:662<br>time_loopback:635<br>time_loopback:693 |
| 5. | 80 feet ( 24.358) | 850 | time_loopback:309<br>time_loopback:886<br>time_loopback:957<br>time_loopback:838<br>time_loopback:907<br>time_loopback:907<br>time_loopback:788<br>time_loopback:1005<br>time_loopback:998<br>time_loopback:1006<br>time_loopback:1048 |
| 6. | 100 feet (30.48 ) | 950 | time_loopback:1079<br>time_loopback:886<br>time_loopback:957<br>time_loopback:838<br>time_loopback:907<br>time_loopback:907<br>time_loopback:998<br>time_loopback:1005<br>time_loopback:998<br>time_loopback:1046<br>time_loopback:1048 |

### C. Graph between Distance and Time



As showing in graph the time on Y-Scale is proportionally increasing with distance on X-Scale.

## VIII. Conclusion

By examine this graph, this is concluded that packet tranfer rate is affected by change in distance. Because delay(time) is inversely proportional to speed. So for successful communication of packet transfer between two Zigbee devices, packet transfer rate should be decrease when distance is being increased.

## REFERENCES

[1] Design and Implementation of a Testbed for IEEE 802.15.4 (Zigbee) Performance Measurements, Patrick R. Casey, Kemal E. Tepe, and Narayan Kar ,Electrical and Computer Engineering Department, University of Windsor, Windsor, Ontario, N9B 3P4, Canada.

[2] ZigBee Alliance, Zigbee Specification Version 1.0 ZigBee Document 053474r06, December 14th, 2004.

[3] M. Lin, Y. Wu, and I. Wassell, "Wireless sensor network: water distribution monitoring system," in Proceedings of the IEEE Radio and Wireless Symposium (RWS '08), pp. 775–778, Orlando, Fla, USA, January 2008.

[4] International Journal of Engineering & Technology IJET Vol: 9 No: 10, Water Quality Monitoring System Using Zigbee Based Wireless Sensor Network , Mohd Rizal Abdullah Zulhani Rasin, Faculty of Electrical Engineering University Teknikal Malaysia Melaka (UTeM)

[5] http://www.arduino.co.cc

[6] XBee™/XBee-PRO™ OEM RF Modules, IEEE ® 802.15.4 OEM RF Modules by MaxStream, Inc.

[7] X-CTU Configuration & Test Utility Software, User's Guide, ftp1.digi.com/support/documentation/90001003_a.pdf

[8] Arduino Tutorials. Available : www.arduino.cc/en/Tutorial /HomePage

[9] XbeeShield,[Online]. Available : http://arduino.cc/en/Main/Arduino XbeeShield

[10] Arduino Website [Online]. Available : http://www.arduino.cc/

[11] Arduino Duemilanove, [Online]. Available : http://www.arduino.cc/ en/Main/ArduinoBoardDuemilanove

[12] Preeti Sohni, "Wireless sensor monitoring system based on the ZigBee$^{TM}$ technology," in Proceedings of the Third National Conference on Communication, Information and Telematics (CITEL2011) 3rd & 4th March 2011, Department of ECE, Kumaraguru College of Technology, Coimbatore – 641 049, Tamilnadu, India.