ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 4, Issue 3, March 2017

# Analysis of Routing Protocol for Different Packet Size Data Transfer over WSN

[1] Poonam Thakur, [2] Rahul Pethe,r, [3] Dr. G. M. Asutkar, [4] Dr. S. A. Chaturvedi
[1] [2][3]1Priyadarshini Institute of Engineering and Technology, Nsgpur, Maharashtra, India

*Abstract* - **This paper describe to create distributed client node wireless sensor network using nRF24 wireless networking protocol to send their data on cloud through Ethernet based sink node. WSN widely used in health monitoring, environmental, traffic control, automation system. In each area we required sink and server node. We are presenting the results during evaluating WSN by using ATMEGA 32u4 Microcontroller, transceiver nRF24L01 with Ethernet Shield for Real Time Data Acquisition Process. For real time data acquisition system microcontroller Atmega32u4 is easily interfaced with nRF24. Client sensors continuously generates amount of data in the form of packets and frame (date, time, address, sample, data, CRC, ERC) which can easily reach to the destination and observed the result on different monitor format by using Ethernet Shield and open source API i.e. Thingspeak. Nowadays, WSN processes apply on different size packet formats for example text, images, or video data. In this work we proposed DSR, AODV, DSDV, AOMDV routing protocol using coding and compression technique for different parameters like energy, throughput, jitter, packet delivery ratio.**

*Keywords:* **Routing Protocol, NS2, ThingSpeak API.**

## I. INTRODUCTION

A wireless sensor network consisting of distributed autonomous sensor equipped with less power transceivers used to gather numerous data form environments. The working of nodes is used to measure and need to transmit all the collected information to base station over a wireless channel. The data are then processed in the base station to draw some conclusions about the current activity in the area. The idea of wireless sensor network is to interact with the environment. Wireless sensor network are usually embedded in an environment. Number of sensor nodes combines form a network. These wireless sensor nodes are equipped with sensors which are capable to measure environmental parameters i.e. temperature, pressure, humidity and more others. The measured data is processed into certain packet format and makes it available to different applications. The Internet of things is the network of physical objects with embedded technology to sense, communicate and interact with their internal states or the external environment [1]. In the WSN context, data may transmit by using different hardware nodes which can be transmit data to the cluster head of a network then cluster can transmit it to the base station. In this technique data can transmit over a small distance. Consider a situation where many host can send their data and it can be download by sink node but some packet data may loss that can be retrieved by using some of the algorithm used for large data recovery. Data can be transmitted over a large distance but it requires large power otherwise data may loss. Data should be sending in different packet size for different application. In our research work we concentrate on large packet size with less number of packet losses. In this research work, we design hardware for specific packet format, but for different packet size we can go through NS2 simulators.

A wireless sensor network (WSN) consists of Microcontroller, Sensor, Transceiver, display unit, Open Source API. In wireless sensor network, energy model is one of the optional attributes of a node. The energy model denotes the level of energy in a mobile node. The design of routing protocols in WSNs is influenced by many challenging factors. These factors must be overcome before efficient communication can be achieved in WSNs.

- Energy Efficiency
- Delay
- Through put
- Jitter
- Packet Delivery

By using NS2 simulator, we proposed the large data transfer packet format, Ns is a discrete event simulator targeted at networking research. Ns provides substantial support for simulation of TCP, routing, and multicast protocols over wired and wireless (local and satellite) networks. Ns-2 is an open source discrete event simulator used by the research community for research in networking [2]. It has support for both wired and wireless networks and can simulate several network protocols such as TCP, UDP, multicast routing, etc.

ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 4, Issue 4, March 2017

## II. PROPOSED METHODOLOGY

This research work attempts to give generalized solution of transmitting the data over wireless sensor network using hardware and also through NS2 simulators in different packet format for wireless sensor network. There are different ways to design such type of hardware but in our research we used ATMEGA32u4, nRF24L01, DHT11 sensor, for storing result we used SD card, Serial monitor, LCD display and open source API through real time clock. This same process can implement by using NS2 simulator, by using simulators we used DSR, AODV, AOMDV DSDV routing protocol for different packet size. Performing simulation for large data we used Runlen Encoding technique for compression of data and transmitting end and decompression of data at receiving end.

### 3. Node Specification

Fig. 1 Sink Node with Ethernet Node

#### *3.1 Packet Format Algorithm*

In this project we used Arduino software available as an open source platform which makes it easier to write program and upload it to the circuit boards. As shown in figure 1, basically there are two nodes used in any network i.e. Server node and Client node. Server node is always connected to the serial port of the computer. During initialization, server node initializes RTC (Real time Clock) for real time data acquisition, nrf24L01 (transceiver) for packet data transceiver and SD card for data storage and other parameter which are connected. On the other hand the client node also initialized by software with to observe packet data transmitted or not then it will be powered by an adapter and kept it away from server node in range of transceiver. Client node will send the measured data in the form of packets to the server node.

| Address (Hide) | Date (YY:MM:DD) | Time (HH:MM:SS) | Node Name | Data | ERC |
|---|---|---|---|---|---|

***Fig. 2 Packet Format***

Fig. 2 shows the packet format of data transmits and received by client node and server node.1st block indicated by source and destination address, 2nd and 3rd block for date in YY: MM: DD and time in HH:MM:SS format, 4th block indicate node name which is mentioned in program, 5th block indicate data collected by the sensors connected with client node and send it to server node, 6th block for error check whether data is properly send by client node and received by server node.

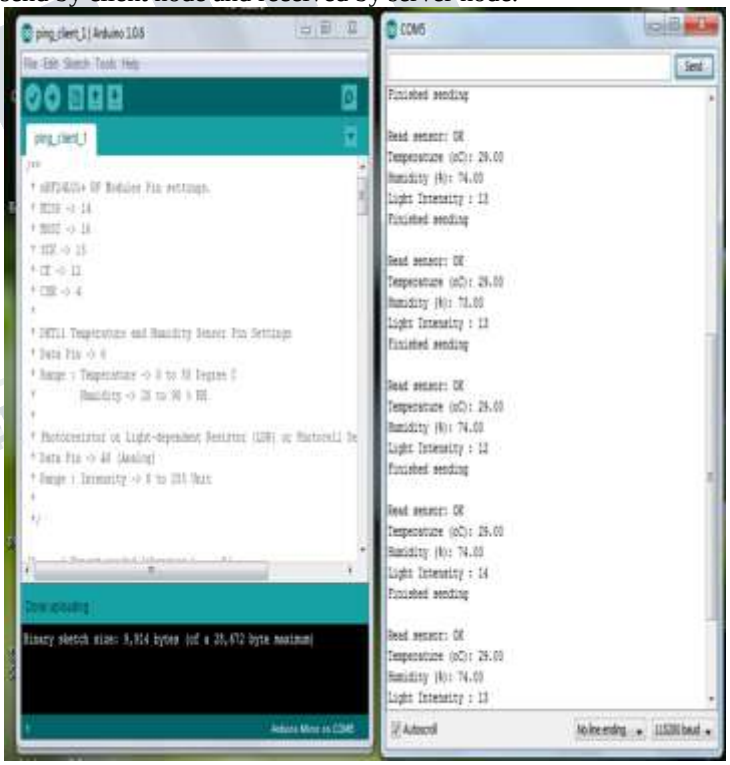

***Fig. 3 Packet format send by Client node.***

Fig. 3 shows the packet format send by client node and also monitor whether it has been properly send by client node or not. Whenever the first packet finish sending then and then only second packet send by client node otherwise stop sending.

ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 4, Issue 4, March 2017

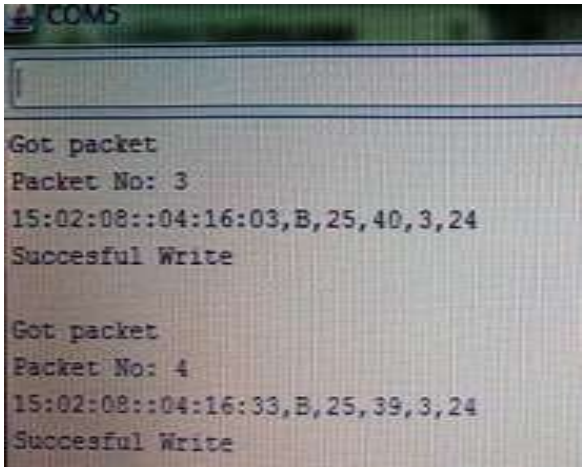


***Fig. 4 Packet format received by server node.***

In fig. 4 we can see the format of packet, i.e. Date, Time, Node name, data (temperature, Pressure, Humidity and light intensity). To reduce the size of packet format we can also skip some or the other parameters.

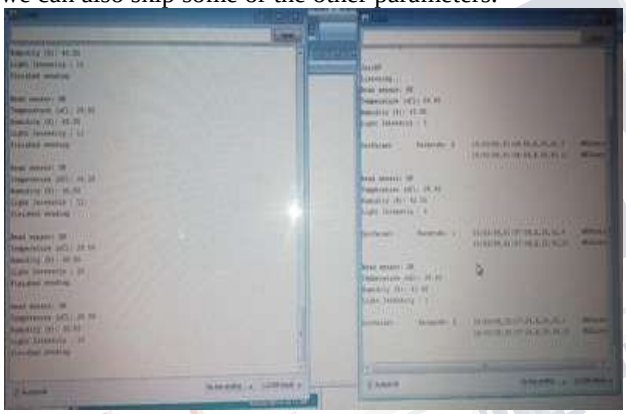

***Fig. 5 Server and Client node Data Monitor***

This same data can also monitor on cloud by an open source AP for example "ThingSpeak". Here we create an account on Thingspeak API for monitoring real time data acquired by Ethernet module which is interfaced/connected with server node. So the data which is send by client node and received by server node. Since server node interface with Ethernet node and Ethernet node connected with MODEM so the data from client node easily available on Thingspeak API in separate graphical format. [3]

ThingSpeak is an open application platform designed to enable meaningful connections between things and people. ThingSpeak has an open source API to store and retrieve data from things using HTTP over the Internet or via a Local Area Network. With ThingSpeak, we can create sensor logging applications, location tracking applications, and a social network of things with status updates. [4]Following Result observed on Thingspeak API as shown below


***Fig 6a Temperature sensor reading on ThingSpeak API***

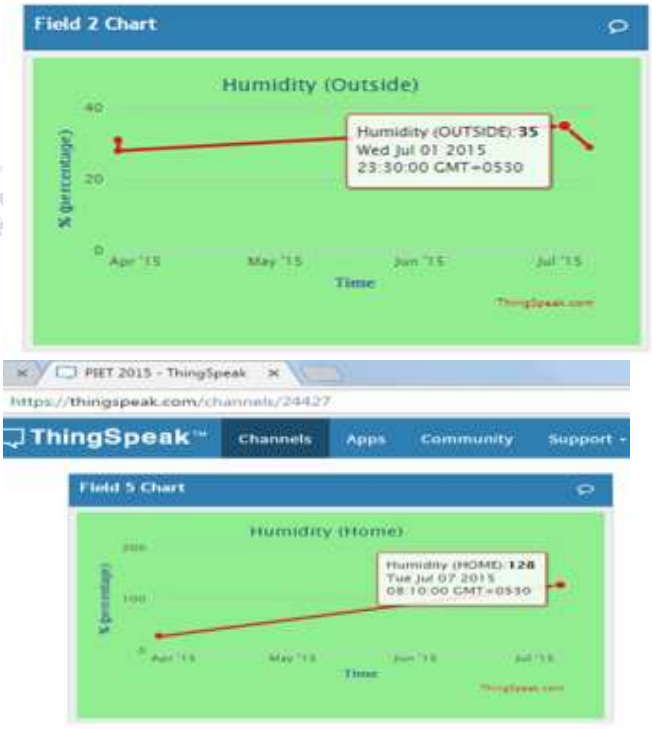


***Fig 6b Humidity sensor reading on ThingSpeak API***

ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 4, Issue 4, March 2017

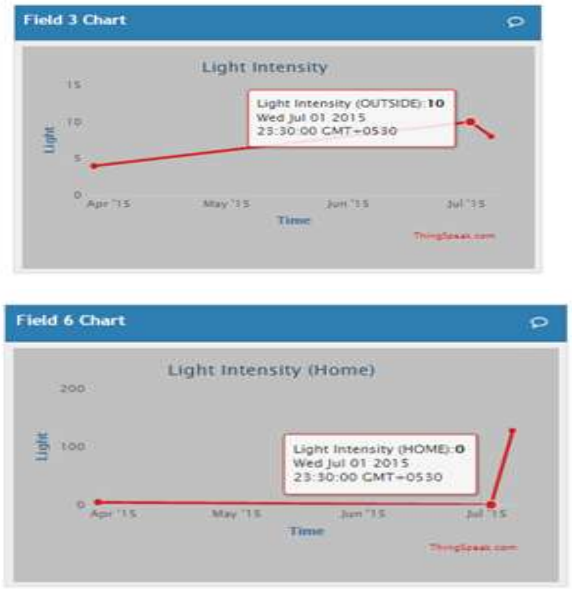


*Fig. 6c Light intensity result on ThingSpeak API*

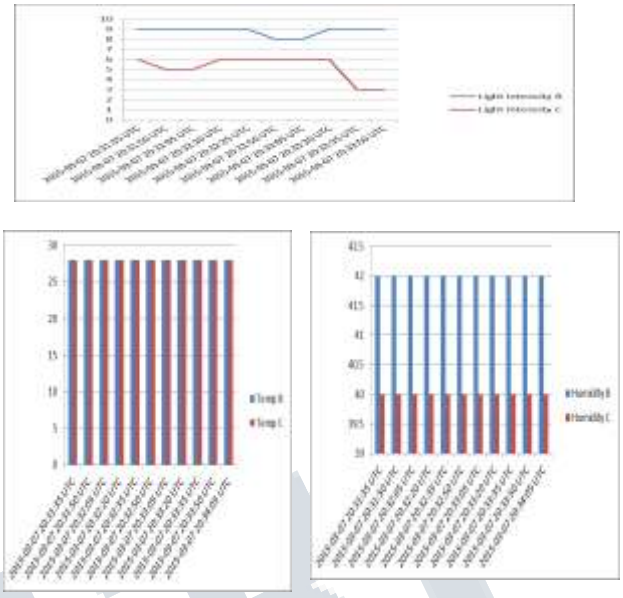

*Fig. 8 Graphical presentation of result through ThingSpeak API*

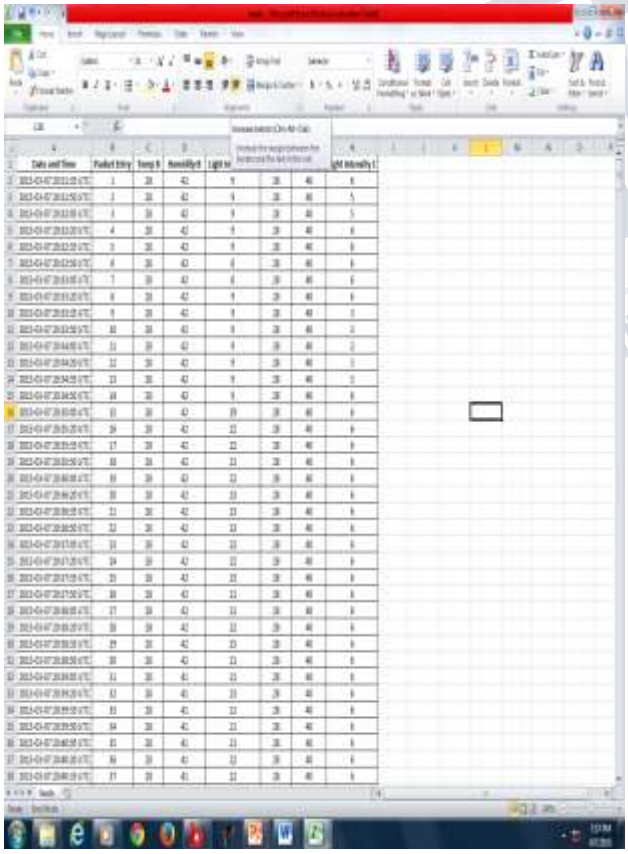

*Fig 7 Result stored in excel format on ThingSpeak API*

Above fig shows the result on ThingSpeak API, the packet format send by client node and also monitor whether it has been properly send by client node or not. we can see the format of packet, i.e. Date, Time, Node name, data (temperature, Pressure, Humidity and light intensity). Graphical presentation of result are available in mentioned packet format in fib 6a,6b and 6c same result will convert in excel format (shown in fig 7) and it can easily convert in graphical representation (shown in fig 8).

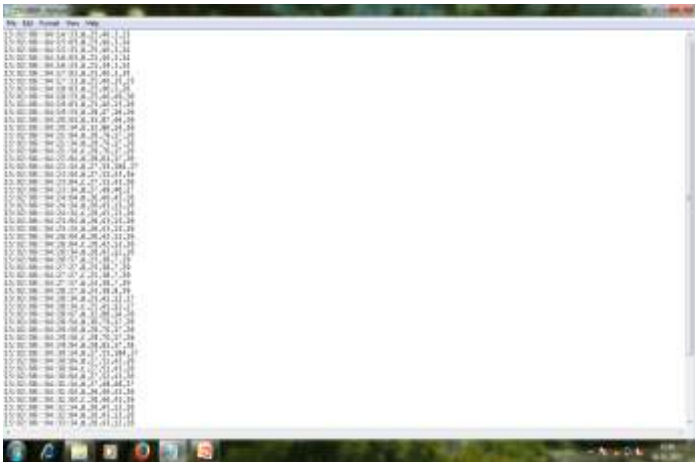

*Fig 9 Result Stored in SD Card*

## III. CONCLUSION

It has been conclude that we can design the WSN's for different application where the data transmitted by client node, received by server node and the same is made available on an open source API i.e. ThingSpeak. The same data can also monitor by serial monitor by using COM port and also store in SD card for future use.

We can design the WSN's for industrial applications which fulfills the application requirements, providing practically nearly 100% reliability with the acceptable latency. This shows that WSNs are fully capable of robust and reliable communication in the harsh environment found on industrial platforms.

Although the use of WSN provides many benefits for industry and there are many various potential application areas for Industrial WSN, the problem of implementing reliable wireless communication in real-life industrial environment is still very complicated and requires further research.

## REFERENCES

[1] E. Altman, T. Basar, T. Jimenez, and N. Shimkin, "Competitive routing in networks with polynomial costs," IEEE Trans.

[2] ns-2 Home page : http://www.isi.edu/nsnam/ns

[3] R. Pethe, Dr. S. M. Asutkar, "Wireless Sensor Network for Industrial Application" International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169, Volume: 3 Issue: 5, page no 068-071

[4] Poonam Thakur, R. M. Pethe, "Real Time Data Acquisition over Cloud using Hardware Based WSN, International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169, Volume: 3 Issue: 5 060 – 064