

Internet of Things (IOT) based Reconfigurable and Scalable Wireless Sensor Network (WSN)

Ashwin Pai¹, Lakshmi Nair², Abhay Kshirsagar³

U.G. Student, Department of Electronics Engineering, VESIT, Mumbai, Maharashtra, India^{1,2}

Associate Professor, Department of Electronics Engineering, VESIT, Mumbai, Maharashtra, India³

ABSTRACT: IoT based Reconfigurable and Scalable WSN is a wireless sensor network which aims to monitor the data recorded by the sensors over the internet. It provides real time feedback to the observer and also alerts them when the sensor values are out of range. It supports plug and play feature making it reconfigurable and based on the values of the sensor it will change the transmission rate making it scalable. In addition, the transmission of data will be security encrypted. IoT based WSN will definitely be used in all the domains in the future right from Manufacturing to Remote Health Monitoring. In this paper, we are going to discuss creation of the system, the advantages of such a system in addition to the ease of implementing it.

KEYWORDS: IoT, WSN, Reconfigurable, Scalable, Security Encrypted.

I. INTRODUCTION

As the technology is advancing, we can see a tremendous surge in the use of Internet. In order to update ourselves and our surroundings with this change, it is necessary to create a basic IoT based sensor network which can be used for multiple applications without much modification. This will help us connect our surroundings over the internet and get useful information online. Wikipedia defines Wireless Sensor Network (WSN) as spatially distributed autonomous sensors which are used to monitor physical or environmental conditions like temperature, sound, pressure, etc. and to cooperatively pass their data through the network to a main location. IoT refers to the networked interconnection of everyday objects, which are often equipped with ubiquitous intelligence. IoT will increase the ubiquity of the Internet by integrating every object for interaction via embedded systems, which leads to a highly distributed network of devices communicating with human beings as well as other devices [1]. For demonstration purpose, in our project we used two sensor nodes and one coordinator node. We have used IEEE 802.15.4 standard i.e. ZigBee. From the coordinator node the data is sent over the internet using a Wi-Fi Module. We have used a temperature sensor and a heart-beat sensor. So we will be demonstrating a basic health monitoring system. Currently the health monitoring is done manually i.e. the nurse has to present with the patient in order to monitor the various parameters. Our system will be uploading the information online so that there is no need for the patient to be attended all the time. The patient will be monitored remotely over the internet. There will be a control room where the parameters of all the patients can be monitored easily.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 4, April 2017

II. BLOCK DIAGRAM

As shown in Figure 1, block diagram of IoT based WSN consists of the following components:

1. Sensor

In our project we have used two sensors.

A. Temperature Sensor

We have used a LM35 for measuring the temperature. It is a precise temperature sensor with its output directly proportional to the temperature in degree centigrade. The temperature can be measured more accurately as the sensor circuitry is sealed and it possesses low self-heating.

B. Heart Beat Sensor

The heart beat sensor is used to detect the heart rate of the person. We have used KY-039 module. This module uses bright infrared (IR) LED and a phototransistor to detect the pulse of the finger. The light side of the finger is kept on the LED while the other side of the finger is kept on the phototransistor. The phototransistor is used to obtain the flux emitted. When the finger is placed, the resistance of the photo transistor will change. The most important part is to keep the shield stray light away from the phototransistor. For home lighting that is particularly important because the lights at home mostly based 50HZ or 60HZ fluctuate, so faint heartbeat will add considerable noise. We can count the number of pulses and detect the heart rate.

2. Arduino Uno

Arduino Uno is used as a controller in this project. The main function of the arduino is to receive the analog sensor values, process it according to the sensor specification and then transmit it serially to the xBee S2C. We can use any other controller in place of Arduino. We have used Arduino because of its simplicity and ease of operation.

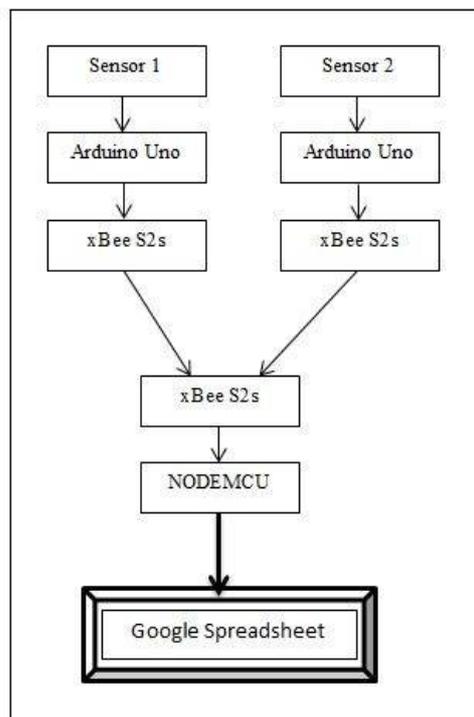


Figure 1: Block Diagram of IoT based WSN

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 4, April 2017

3. xBee S2C

The xBee S2C is used to transmit the data wirelessly. It has an advantage of low power consumption and it provides reliable, self-configurable and self-healing networks. xBee S2C also provides security encryption making the transmission of data safe.

4. NodeMCU Development Kit 1.0

NodeMCU is an open source IoT platform. It includes firmware which runs on the ESP8266 Wi-Fi SoC, and hardware which is based on the ESP-12 module. To sum it up we can say that a NodeMCU Dev Board is a combination of Arduino and ESP8266. NodeMCU can be easily programmed using the Arduino IDE. The main task of NodeMCU in our project is to connect to the Wi-Fi and transmit the data to a Google Spreadsheet.

III. CREATING THE SYSTEM

The sensor node is very simple to build. It consists of an Arduino Uno, Xbee Shield, xBee S2C module and the sensor. The Xbee Shield is mounted on top of the arduino Uno and it has a provision to attach the xBee module [2]. The sensor pins are connected to the arduino via the Xbee Shield. The sensor output pin is connected to the analog input of arduino. The arduino will process the sensor input and convert it to digital output. The system can be created by following the given steps:

1. Creating the xBee network
To program the xBee, we use XCTU Software. Using this we create a xBee network by assigning a common PAN ID to all the xBee modules. Then define a unique address for each of the xBee's and we encrypt the data using 128 bit encryption[3].
2. Programming the Arduino
We can program the Arduino according to the sensor we used. We require Arduino IDE software in order to program the Arduino. The arduino has 8 analog pins. We can connect the sensor to any of these pins and choose that pin as input pin in the program.
3. Creating a Google Spreadsheet
We first create a Google Spreadsheet and then write a script for it. The script is mainly written in order to enter the data in the proper format along with the time stamp in the spreadsheet [4].
4. Coding the NodeMCU
The NodeMCU is coded using Arduino IDE. The NodeMCU is programmed to connect to the Wi-Fi and in order to assign the delay according to the pre decided range conditions.

Thus we can see that in four easy steps the IoT based reconfigurable and Scalable WSN is created.

IV. WORKING

1. The temperature sensor senses the body temperature of the patient and sends the data to the Arduino Uno. The data is processed here and the temperature is calculated. This value is then sent to the xBee module via the serial pins. The xBee module transmits the data after encryption.
2. When the fingertip is placed in between the heart beat sensor, the pulse gets detected and it is recorded by the Arduino Uno. It counts the pulses for a short period of time and calculates the pulse rate. After it is calculated, it transmits the value to the xBee module via the serial pins. The xBee module on receiving the data transmits it wirelessly after encryption.
3. The xBee module present at the coordinator node receives the data from the sensor nodes and decrypts it. It then sends it to the NodeMCU via the serial pins. The NodeMCU checks the data and classifies it as temperature and pulse rate. It then checks whether the values are within the range, if the value is within the range then it uploads the data on the Google Spreadsheet and after a pre-determined interval it receives the next sensor data, but if the value exceeds or is less than the prescribed range then it uploads the value and immediately receives the next reading from the sensor node.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 4, April 2017

- The data uploaded on the Google Spreadsheet can be accessed from any part of the world, but to access the data the owner has to grant permission. Thus making the readings secure from public access.

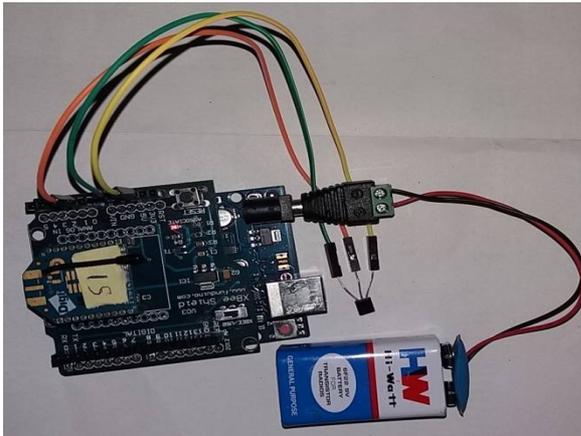


Figure 2: Sensor node 1

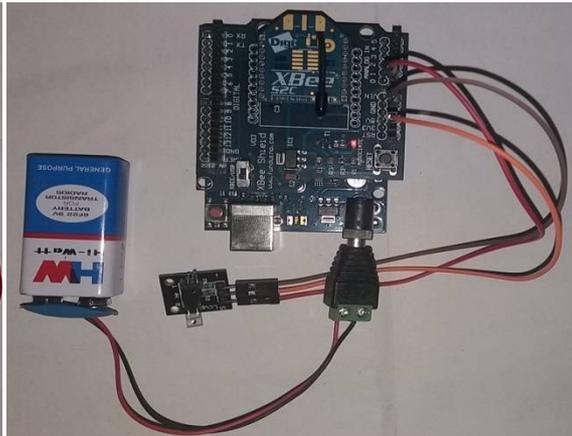


Figure 3: Sensor Node 2

Figure 2 shows the first sensor node. We have attached a temperature sensor LM35 to analog pin 1 of the Arduino. Figure 3 shows the second sensor node. Over here we have attached a heart beat sensor KY039 to the analog pin 2 of the Arduino.



Figure 4: Coordinator Node

Figure 4 shows the coordinator node. This consists of the NodeMCU and the xBee module. The NodeMCU is responsible for connecting to the Wi-Fi and transmitting data to the Google Spreadsheet.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 4, April 2017

V. RESULTS

A	B	C	D
Sr. No.	Date Time	Temperature	Heartbeat
1	3/30/2017 9:25:08	32	65
2	3/30/2017 9:30:08	34	65
3	3/30/2017 9:31:09	28	92
4	3/30/2017 9:32:09	33	97
5	3/30/2017 9:37:09	30	102
6	3/30/2017 9:38:10	30	68
7	3/30/2017 9:43:10	31	72
8	3/30/2017 9:48:11	31	74
9	3/30/2017 9:53:11	32	105
10	3/30/2017 9:54:11	32	112
11	3/30/2017 9:55:12	33	73
12	3/30/2017 10:00:12	33	59
13	3/30/2017 10:01:13	34	119
14	3/30/2017 10:02:13	34	72
15	3/30/2017 10:03:13	35	80
16	3/30/2017 10:04:14	28	80
17	3/30/2017 10:09:14	30	79
18	3/30/2017 10:14:15	30	86
19	3/30/2017 10:19:15	31	85
20	3/30/2017 10:24:15	31	77

Figure 5: Output on Google Spreadsheet

Figure 5 shows us the data that is received on the Google Spreadsheet. We can clearly see that out of range values are highlighted in red which makes it easy for us to identify them. The figure also helps us understand the time delay i.e. if the data is within the range the next reading is recorded after an interval of 5 minutes, but if the values are out of range, the next reading is taken after a delay of 1 minute.

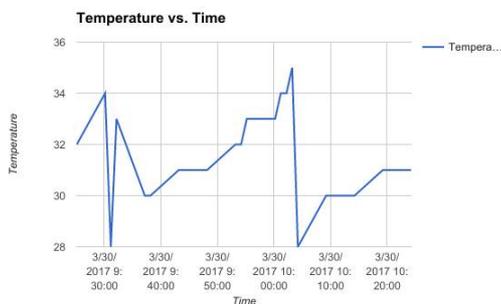


Figure 5: Temperature vs. Time Graph

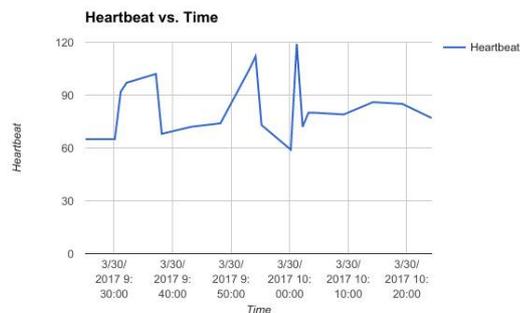


Figure 6: Heartbeat vs. Time Graph

Figure 5 and 6 shows us the graph of the readings recorded on the Google Spreadsheet. This graph can be viewed on the Google Spreadsheet itself with the help of Graph Function. The graph gets updated on a real time basis and it becomes easier to understand the readings.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 4, April 2017

VI. ADVANTAGES

- The Sensor Network will be easy and convenient to use since there will be no wired connections. A wired system consists of a lot of wiring, a small cut in any of the wires means the system is down. It is difficult to add new sensor nodes in a wired network as it will involve reconfiguration of the system. The wired network cannot go through partitions like walls. This is the main advantage of a Wireless Sensor Network.
- The system would be very useful for remote monitoring. This would allow the concerned people to view the data from any location. This is possible since the data is being sent and stored in the Google Spreadsheet.
- The data transmission is secured with the help of encryption, thus making the system private and protected. It is also made sure that the data can only be accessed by authorized parties which in this case would be the hospital, the doctors and the patient. Nobody else would be able to access this data.
- The control unit would be checking the parameter to see whether the data is in the normal range. If it is not in the normal range then it will send the data with minimum delay and this will be alerted on the Spreadsheet.

VII. APPLICATIONS

The system has been designed to be an application oriented system. We can build many applications by just changing the sensor used. Some of the applications that are possible are:-

- Health Monitoring System
- Weather Monitoring System
- Agricultural Monitoring System
- Pollution Monitoring System
- Forest Fire Detection
- River Water Quality Detection

VIII. CONCLUSION AND FUTURE SCOPE

The sensor network transmits the values from the sensor node after encrypting it to the coordinator node using IEEE 802.15.4 standard. The coordinator node on checking the sensor value for the range condition varies the time interval accordingly and sends the data over the internet to a Google spreadsheet. The out of range values are highlighted on the spreadsheet. The Wireless Sensor Network created can be used for a lot of applications. If we change the heart rate sensor with a humidity sensor, then the system becomes a Weather Monitoring System.

The future scope of the project lies in creating an online database for the system using cloud. The data can then be displayed on mobile applications or web-pages with password protection. The sensor nodes can be reduced to the size of a small pen drive by using Wireless MCU's like CC2531. This will make it very convenient for the user. We can also use sleep function in arduino and xBee to make it a low power system.

REFERENCES

- [1] Feng Xia, Laurence T. Yang, Lizhe Wang and Alexey Vinel, "Internet of Things", International Journal of Communication Systems, Vol.25, pp.1101-1102, 2012
- [2] Scott C, "Arduino UNO - XBee Setup", 2012, Retrieved from <http://arduinobasics.blogspot.in/2012/10/arduino-basics-arduino-uno-xbee-setup.html>
- [3] Robert Faludi, "Building Wireless Sensor Networks", December 2010
- [4] ANir, "HOW TO POST DATA TO GOOGLE SHEETS USING ESP8266", 2016, Retrieved from <http://embedded-lab.com/blog/post-data-google-sheets-using-esp8266>