IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 10, October 2016

Internet of Things (IoT) based Sensors to Cloud system using ESP8266 and Arduino Due

Mr. Nerella Ome[1], Mr. G. Someswara Rao[2]

Master of Engineering, Assistant Professor, GRIET, Hyderabad, Telangana, India[1]

M.Tech, Assistant Professor, VJIT, Hyderabad, Telangana, India[2]

Abstract: The system proposed in this paper is an advanced solution for monitoring the weather conditions at a particular place and make the information visible anywhere in the world. The technology behind this is Internet of Things (IoT), which is an advanced and efficient solution for connecting the things to the internet and to connect the entire world of things in a network. Here things might be whatever like electronic gadgets, sensors and automotive electronic equipment. The system deals with monitoring and controlling the environmental conditions like temperature, relative humidity, light intensity and sound level with sensors and send this information to the cloud and then plot the sensor data as graphical statistics. The data updated from the implemented system can be accessible in the internet from anywhere in the world.

Keywords: Internet of Things (IoT), Sensors to cloud System; Arduino Due, Arduino IDE, ESP8266, Thing speak.

I. INTRODUCTION

Internet of Things where 'things'- sensors and devices transmit data directly to the Internet has become an enabling technology eco-system with several application areas are Smart Home, Smart Farming, Smart Grid, Industrial Internet, Connected Health, Smart Supply Chain etc.
The application list is impressive, however, since the technologies involved are many- sensors, microcontrollers, wireless networking, cloud based services, mobile apps, web pages -practical implementation of an IoT application is complex.

Present innovations in technology mainly focus on controlling and monitoring of different activities. These are increasingly emerging to reach the human needs. Most of this technology is focused on efficient monitoring and controlling different activities.
An efficient environmental monitoring system is required to monitor and assess the conditions in case of exceeding the prescribed level of parameters (e.g., noise, CO and radiation levels).

When the objects like environment equipped with sensor devices, microcontroller and various software applications becomes a self-protecting and self-monitoring environment and it is also called as smart environment.
In such environment when some event occurs the alarm or LED alerts automatically.

The effects due to the environmental changes on animals, plants and human beings can be monitored and controlled by smart environmental monitoring system. By using embedded intelligence into the environment makes the environment interactive with other objectives, this is one of the application that smart environment targets.

Thing Speak is a IoT cloud service provider. Embedded IoT devices like Arduino, Raspberry Pi can be connected to internet. These boards then can fetch data or upload data to Thing Speak storage using APIs.

The data stored by a device can be accessed by other client entities like Mobile, Tablet, laptop connected to internet using Thing Speak APIs. So in short Thing Speak is an IoT service provider that provides APIS to upload, retrieve and visualize data from IoT devices over cloud.

II. EXISTING SYSTEM MODEL

In today's world many pollution monitoring systems are designed by considering different environmental parameters.

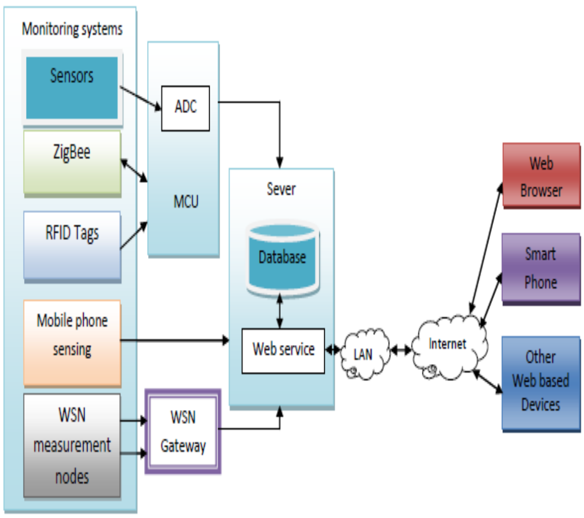


Fig. 1: Existing System Model

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 10, October 2016

Existing system model is presented in figure 1 uses Zigbee [3] based wireless sensor networks to monitor physical and environmental conditions with thousands of application in different fields. The sensor nodes directly communicated with the moving nodes deployed on the object of interest which avoided the use of complex routing algorithm but local computations are very minimal.

RFID [4] is a means of storing and retrieving data through electromagnetic transmission to an RF compatible integrated circuit. It is usually used to label and track items in supermarkets and manufactories.

Mobile phones [5] or smart phones that are enabled with sensors are used for impact on social including how mobile technology has to be used for environmental protecting, sensing and to influence just-in-time information to make movements and actions environmental friendly. Mobile phone sensors were deployed and used on urban areas for monitoring.

A Wireless Sensor Network [6] consists of many inexpensive wireless sensors, which are capable of collecting, storing, processing environmental information, and communicating with neighboring nodes. In the past, sensors are connected by wire lines.

The access method of WSN gateway node is convenient because data can be received from a WSN via the gateway at any time and any place.

A server is an instance of a computer program that accepts and responds to requests made by another program; known as a client. Less formally, any device that runs server software could be considered a server as well. Servers are used to manage network resources.

The services or information in the servers are provided through the Internet that are connected through LAN and made available for users via smart phones, web browser or other web browser devices to make the system more intelligent, adaptable and efficient.

III. PROPOSED MODEL SYSTEM ARCHITECTURE

The proposed model consists of a Microcontroller (Arduino Due-ARM SAM3X8E) as a main processing unit for the entire system and all the sensors and devices can be connected with the microcontroller.

The sensors can be operated by the microcontroller to retrieve the data from them and it processes the analysis with the sensor data and updates into the cloud through Wi-Fi module connected to it. Here we are using Arduino Due because it is compatible with 3.3v ESP8266 Wi-Fi module and it also contain more than one on chip UART's so we can connect more number of Serial devices.

BLOCK DIAGRAM :

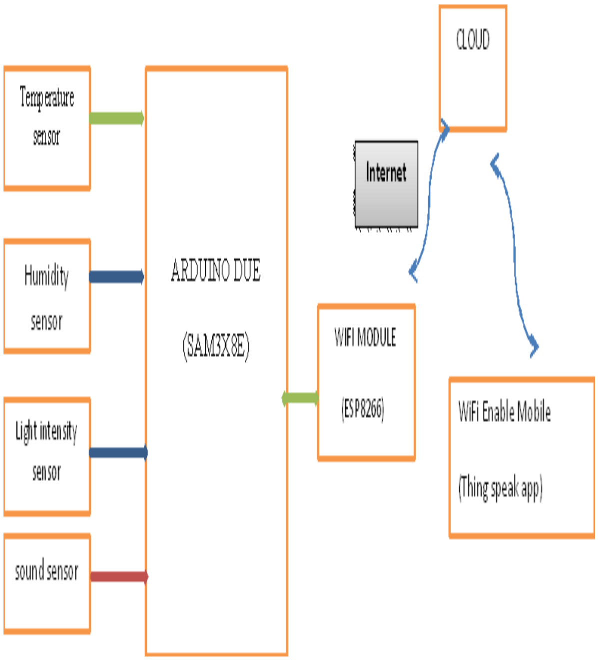
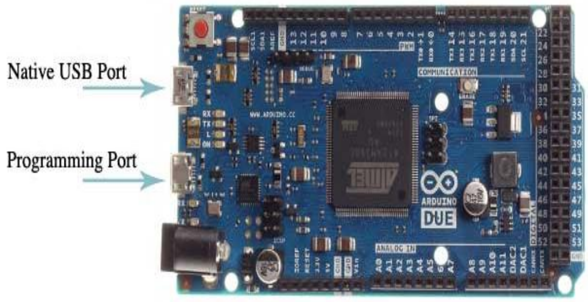


Fig.2. Block diagram of the Implementation model

Arduino Due:



The Arduino Due is a microcontroller board based on the Atmel SAM3X8E ARM Cortex-M3 CPU. It is the first Arduino board based on a 32-bit ARM core microcontroller. The basic features of the Arduino Due are:

- 54 digital input/output pins (of which 12 can be used as PWM outputs)
- 12 analog inputs
- 4 UARTs(hardware serial ports)
- 84 MHz clock
- USB OTG(On the Go) capable connection
- 2 DACs(digital to analog)
- 2 TWI(Two wire interface)
- Power jack
- SPI(Serial Peripheral Interface) header

The Due is compatible with all Arduino shields that work at 3.3V; the Arduino Due board runs at 3.3V. The maximum voltage that the I/O pins can tolerate is 3.3V.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 10, October 2016

Arduino IDE:
Arduino is an open source tool for making computers that can sense and control more of the physical world than your desktop computer. It's an open-source physical computing platform based on a simple micro-controller board, and a development environment for writing software for the board. Arduino can be used to develop interactive objects, taking inputs from a variety of switches or sensors, and controlling a variety of lights, motors, and other physical outputs. Arduino projects can be stand-alone, or they can communicate with software running on your computer (e.g. Flash, Processing, MaxMSP).

Based on the Processing multimedia programming environment.. Sensors are connected to Arduino Due board for monitoring, ADC will convert the corresponding sensor reading to its digital value and from that value the corresponding environmental parameter will be evaluated.

Thing Speak:
Thing Speak is an Internet of Things (IoT) platform that lets you collect and store sensor data in the cloud and develop IoT applications. The Thing Speak IoT platform provides apps that let you analyze and visualize your data in MATLAB, and then act on the data. Sensor data can be sent to Thing Speak from Arduino, Raspberry Pi, Beagle Bone Black, and other hardware. Thing Speak has integrated support from the numerical computing software MATLAB from MathWorks Allowing Thing Speak users to analyze and visualize uploaded data using Matlab without requiring the purchase of a Matlab license from MathWorks.

Wi-Fi Module

Here we used ESP8266 Wi-Fi module which is having TCP/IP protocol stack integrated on chip. So that it can provide any microcontroller to get connected with Wi-Fi network. ESP8266 is a preprogrammed SOC and any microcontroller has to communicate with it through UART interface.

It works with a supply voltage of 3.3v. The module is configured with AT commands and the microcontroller should be programmed to send the AT commands in a required sequence to configure the module in client mode. The module can be used in both client and server modes.

Sensors:
The Implemented system consists of Temperature and Humidity sensor(DHT11 sensor) , LDR and sound sensor. These 4 sensors will measure the primary environmental factors light intensity, temperature, Humidity levels and sound intensity respectively. All these sensors will give the analog voltage representing one particular weather factor. The microcontroller will converts this analog voltages into digital Data.ESP8266 Wifi module send these sensor values to Thing speak platform(Cloud).

DHT11 (Temperature and Humidity) sensor:
The DHT11 humidity and temperature sensor measures relative humidity (RH) and temperature. This sensor includes a resistive element and a sense of wet NTC temperature measuring devices. It has excellent quality, fast response, anti-interference ability and high cost performance advantages. Relative humidity is the ratio of water vapor in air vs. the saturation point of water vapor in air.

Relative Humidity = (density of water vapor / density of water vapor at saturation) x 100%

The DHT11 calculates relative humidity by measuring the electrical resistance between two electrodes. The humidity sensing component of the DHT11 is a moisture holding substrate (usually a salt or conductive plastic polymer) with the electrodes applied to the surface. When water vapor is absorbed by the substrate, ions are released by the substrate which increases the conductivity between the electrodes.

The change in resistance between the two electrodes is proportional to the relative humidity. Higher relative humidity decreases the resistance between the electrodes while lower relative humidity increases the resistance between the electrodes. Inside the DHT11 you can see electrodes applied to a substrate on the front of the chip:

The temperature readings from the DHT11 come from a surface mounted **NTC temperature sensor** (thermistor) built into the unit

Sound Sensor:

The Sound Sensor detects the decibel level: the softness or loudness of a sound. The Sound Sensor detects both dB and dBA. dBA: the sounds human ears are able to hear. dB: all actual sound, including sounds too high or low for the human ear to hear. The Sound Sensor can measure sound pressure levels up to 90 dB – about the level of a lawnmower. For comparison, 4-5% is like a silent living room and 5-10% is about the level of someone talking

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 10, October 2016

some distance away. From 10-30% is normal conversation close to the sensor or music played at a normal level and 30-100% represents a range from people shouting to music playing at high volumes. These ranges are assuming a distance of about 1 meter between the sound source and the Sound Sensor.

LDR - Light-Dependent Resistor:

An LDR is a component that has a (variable) resistance that changes with the light intensity that falls upon it. This allows them to be used in light sensing circuits. A light-dependent resistor (LDR) is a light-controlled variable resistor. The resistance of this decreases with increasing incident light intensity; in other words, it exhibits photo-conductivity. An LDR can be applied in light-sensitive detector circuits, and light- and dark-activated switching circuits.

An LDR is made of a high resistance semiconductor. In the dark, an LDR can have a resistance as high as a few mega ohms (MΩ), while in the light, an LDR can have a resistance as low as a few hundred ohms. If incident light on an LDR exceeds a certain frequency, photons absorbed by the semiconductor give bound electrons enough energy to jump into the conduction band.

The resulting free electrons (and their whole partners) conduct electricity, thereby lowering resistance. The resistance range and sensitivity of an LDR can substantially differ among dissimilar devices

IV. COMPUTATIONAL ANALYSIS ON SENSOR PARAMETERS

Here we include some basic analytic methods to calculate the Sensors parameters, like Temperature, Humidity, Light intensity and Sound levels in the surrounding environment.

Temperature & Humidity Calculation:
LM35 Temperature sensor gives output voltage 10 mv for 1°C.this sensor output is connected to any analog pin of Arduino Due. Due converts analog voltage into digital using on chip ADC.

ADC reading=analogRead(A1);

Voltage= ADC reading*5/(1023);

Temperature=Voltage*100;

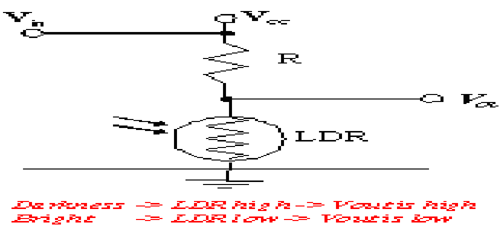
Relative Humidity = (density of water vapor / density of water vapor at saturation) x 100%.

The DHT11 sensor is easy to connect to the Arduino. Before we can use the DHT11 on the Arduino, we need to install the DHT11 library, which contains all of the functions we will need to get the humidity and temperature readings from the sensor.

- DHT.read11(Pinno);
- DHT. Temprature;
- DHT. Humidity ;

By calling this function we will obtain the temperature and humidity

LIGHT INTENSITY CALULATION:



Output across LDR is given to any analog pin of Arduino Due.

When light is falling on LDR its resistance decreases .then voltage across its decreases, ADC reading decreases.
When no light is falling on LDR its resistance increases .then voltage across its increases, ADC reading also increases.

ADC reading=analogRead(A2);

Voltage= ADC reading*5/(1023);

ADC Reading =400 to 650 when brightness
ADC Reading =700 to 1023 when darkness

SOUND INTENSITY CALULATION:
The common unit of measurement for sound is decibel, dB and its intensity is measured in Sound Pressure Level (SPL).
The noise levels are measured in the A–weighted (low-level sensitivity) SPL, abbreviated as dB (A). Sound of frequencies from 800 to 3000Hz is covered by the A–weighted scale.

ADC reading=analogRead(A3)

Sound in Db=20*log(ADC reading)

If the SPL, L1 in dB is measured at r1 meters, then SPL, L2 in dB at r2 meters is given by

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 10, October 2016

$$L2 = L1 - 20 \log_{10} (r2/r1)$$

Day – Night equivalent noise levels (Ld_n) of a community can be expressed as

$$Ld_n, dB(A) = 10 * \log_{10}[\ 15/24(10^{L_d/10}) + 9/24(10^{(L_n+10)/10})\ ]$$

Where,
L_d= day- equivalent noise levels (from 7AM – 10PM), dB(A)

L_n = night-equivalent noise levels (from 10PM – 7AM), dB(A)

Based on intensity, the sound intensity may be expressed in decibels above the standard threshold of hearing I_0. The expression is

$$I(dB) = 10 \log_{10}[I/I_0] \text{ intensity in decibels}$$

TableI: Standard Sound levels

Night (10pm-7am) Unit in decibels	Day (7am-10pm) Unit in decibels	Type of region
45	55	Residential
40	60	Residential – commercial
55	65	Commercial
60	70	Residential-industry
65	75	Industry

Using the Table 1 and values obtained from above calculations, the threshold value can be set to the requirements dynamic nature of the environment and to monitor the parameters data through sensors.

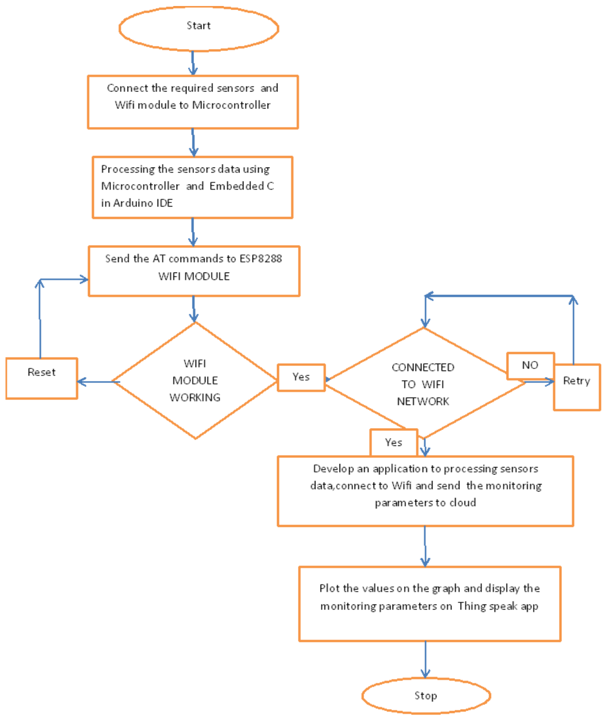
V. IMPLEMENTATION

In this implementation model we used Arduino Due board, Sensors and ESP8266 Wi-Fi module as an embedded device for sensing and storing the data in to cloud. Arduino Due board consist of 12 analog input pins (A0-A11), 54digital output pins (D0-D53), inbuilt ADC. Wi-Fi module connects the embedded device to internet.

Sensors are connected to Arduino Due board. Its read the sensors and on chip ADC will convert the corresponding sensor reading to its digital value and from that values the corresponding environmental parameter will be evaluated.

Here we are connected ESP8266module to 19 (Rx1) and 18 (Tx1) pins of Arduino Due ,DHT11 sensor to one of the digital pin of (PIN 5)Arduino Due ,LDR circuit arrangement is connected to Analog pin (A4) and sound sensor is connected to Analog pin (A2).

Flowchart:



The Wi-Fi connection has to be established to transfer sensors data to end user and also send it to the cloud storage for future usage.

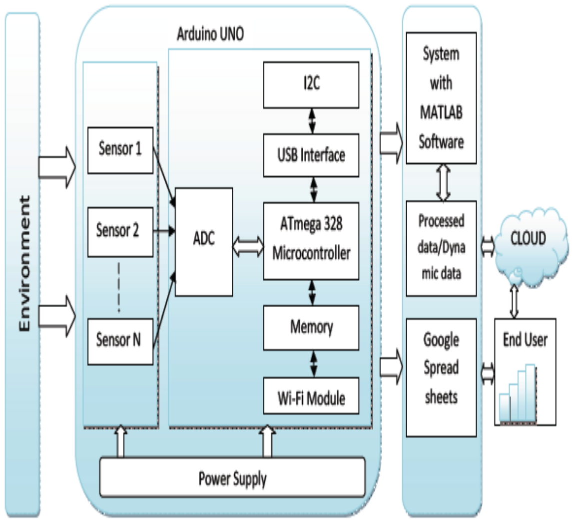


Fig. 4: Schematic diagram of implementation model

An embedded system designed for environmental monitoring and its components are shown in figure 5.The embedded device is placed in particular area for testing purpose. The sound sensor detects sound intensity levels in that area and DHT11sensor and LDR will record the Temperature ,Humidity and Light intensity in that region, if the threshold limit is crossed the corresponding controlling action will be taken (like issuing message alarm or buzzer or LED blink).All the sensor devices are connected to internet through Wi-Fi module.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 10, October 2016

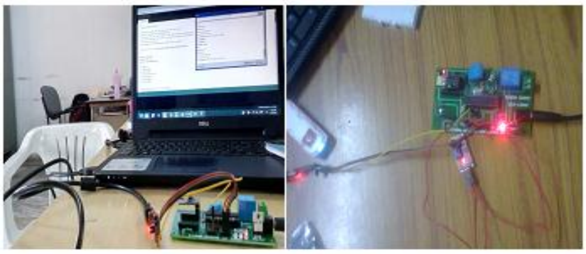


Fig. 5:IoT Based Sensors to cloud system

Figure 5 shows the embedded system with its components for reading and to store the Sensors reading to Thing speak platform(cloud). After successful completion of sensing, the data will be processed and stored in database for future reference. After completing the analysis on data the threshold values will be set for controlling purpose.

VI. SIMULATION RESULTS

1. Create an account on Thing speak platform.
2. Connect Arduino Due board to system through USB cable.
3. After connecting select board and COM port in Arduino IDE.

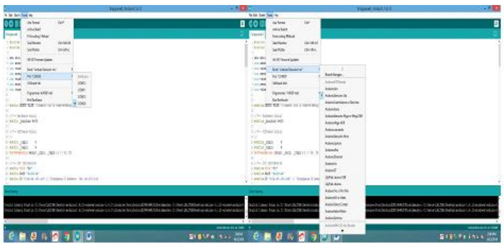


Fig. 6: Arduino IDE Window for Board and COM port selection

4. Develop an Arduino Code for sensors to cloud system in Arduino IDE ,compile and upload the code in Arduino Due board

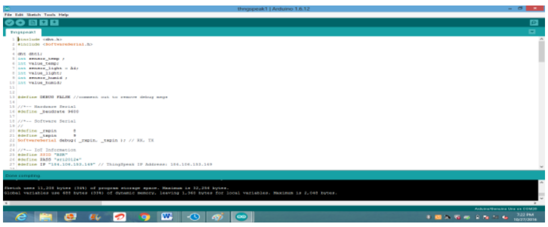


Fig. 7: Arduino IDE Window for Compilation successful.

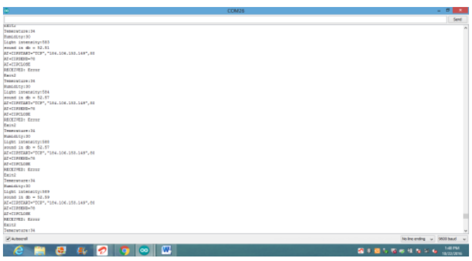


Fig. 8: Sensors Result and Wi-Fi status on Serial monitor.

After sensing the data from different sensor devices, which are placed in particular area of interest. The sensed data will be automatically sent to the Thing speak platform (Cloud), when a proper connection is established with cloud. Then Thing Speak platform analyze and visualize uploaded data using Mat lab

Result:

Temperature =32°C, Humidity = 30
Light intensity= 550 when light is present
= 600-1000 when light is not present
Sound level =45 db in normal
=60 db when present of sound

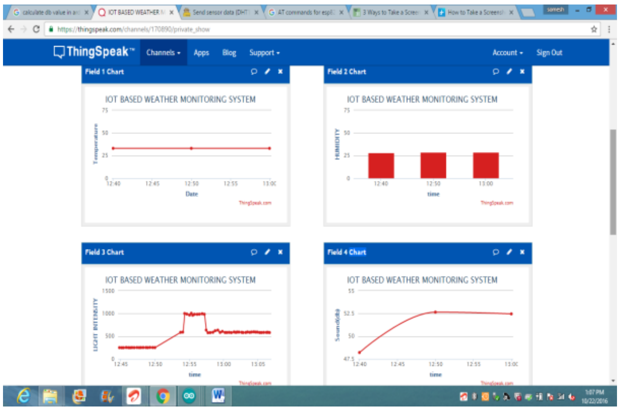


Fig.9.Graphical view of monitoring parameters on Thing speak platform

VII CONCLUSION

By keeping the embedded devices in the environment for monitoring enables self protection (i.e., smart environment) to the environment. To implement this need to deploy the sensor devices in the environment for collecting the data and analysis. By deploying sensor devices in the environment, we can bring the environment into real life i.e. it can interact with other objects through the network.

Then the collected data and analysis results will be available to the end user through the Wi-Fi. The smart way to monitor environment and an efficient, low cost embedded system is presented with different models in this paper. In the proposed architecture functions of different modules were discussed.

The sensors to cloud system with Internet of Things (IoT) concept experimentally tested for monitoring four parameters. It also sent the sensor parameters to the cloud (Thing speak). This data will be helpful for future analysis and it can be easily shared to other end users.

This model can be further expanded to monitor the developing cities and industrial zones for pollution monitoring. To protect the public health from pollution, this model provides an efficient and low cost solution for continuous monitoring of environment

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 10, October 2016

REFERENCES

[1] Nashwa El-Bendary, Mohamed Mostafa M. Fouad, Rabie A. Ramadan, Soumya Banerjee and Aboul Ella Hassanien, "Smart Environmental Monitoring Using Wireless Sensor Networks",K15146_C025.indd, 2013
[2] Grzegorz Lehmann, Andreas Rieger, Marco Blumendorf, SahinAlbayrakDAI, "A 3-Layer Architecture for Smart Environment Models"/A model-based approach/Labor Technische University Berlin, Germany 978-1-4244-5328-3/10 © IEEE,2010.
[3] Moghavvemi M. and Tan. S. "A reliable and economically feasible remote sensing system for temperature and relative humidity measurement". Sensors and Actuators. 2005. 181-185.

BIOGRAPHIES

N. Ome, M.E, Assistant Professor in Electronics and Communication Engineering in Goka Raju Rangaraju Institute of Engineering and Technology, Hyderabad, India.

G. Someswara Rao, M.Tech, Assistant Professor in Electronics and Communication Engineering in Vidyajyothi Institute Of Technology, Hyderabad, India