ISSN: 2277-9655
(I2OR), Publication Impact Factor: 3.785

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

HUMIDITY & TEMPERATURE MONITORING BY USING CLOUD NETWORK

M. B. Randhir*, R. R. Karhe

* Electronics & Telecommunication, SGD College of Engineering, Jalgaon, MS, India.

ABSTRACT

The main objective of the paper is to develop a hardware system which can monitor humidity & temperature continuously on cloud. Monitoring various environmental parameters such as temperature & humidity along with other factors can be of significance in various fields. Cloud data can be monitored from Internet. It can be secured and easy & fast for accessing. So any humidity and temperature signals can be logged into cloud so that any authenticated person can observe the specific data from any place. Continuous varying data can be logged into cloud. Software and hardware based logging into cloud are the two different processes. Software based logging indicates logging with Personal computer. Hardware based logging indicates logging with hardware systems with specific processor. Hardware based logging system may be dedicated to particular application can be designed with RTOS with Embedded Processor. In hardware based logging process, choosing processor is the main task. Fast and error less sensor network handling and processing having internet accessing facility is the main requirement in hardware based logging system. It is also required to access real time varying data with specific intervals.

KEYWORDS: Humidity, Temperature, Cloud, Personal Computer, RTOS, Embedded Processor, Sensor Network, Real Time Varying Data.

INTRODUCTION

We designed the hardware system so that it will direct the real time temperature and humidity data to Google cloud. We can monitor the logged data to cloud from Internet. We used ARM 11 Raspberry pi board for designing dedicated Embedded system. Internet access and sensing system interface can be efficiently done with ARM 11 board so specifically used. Python coding is used for programming with Raspberry pi. Two Python modules are designed 1) humidity and temperature sensing and 2) logging data to cloud. First module senses both the parameters and can display on its monitor and second module sends the given data by logging into Google cloud. Google cloud's service is taken for logging in to their cloud. Excel sheet like data can be monitored from cloud through Internet. Humidity and temperature monitor can be logged into cloud so that any one (authenticated person) from any place can observe the specific data [4]. In case of any disasters like fire, heavy rain, heavy wind, temperature or humidity inside and or outside may be uncontrollable and different, if green house envelop or covering is disturbed due to heavy rain or heavy wind, it may be same. In these cases the immediate information can be conveyed throughout the world using cloud to the authenticated persons so that plants can be taken as early as possible with emergency help. This will be very much helpful in disaster management of green house. System consist of hardware like ARM 11 processor, humidity and temperature sensor circuitry, Internet connection through LAN or wifi. Remote terminal requires the Internet connection (for demo we can use Internet through LAN access). Raspberry Pi Arm 11 board is required to be loaded with boot loader for proper operation. Raspberry pi desktop monitoring is required to develop the Python coding. Raspberry p (Rpi) board's desktop can be accessed on Laptop/PC using LAN connection with Rpi board. Rpi can be pinged from our Laptop/PC once the Rpi is set after initial Loading. Putty and Xming software are used to access the desktop of Rpi on Laptop. One can develop the Python using desktop of Rpi easily. Tools for Python editing, executing, debugging are already available in desktop of Rpi. So, Python models can be easily developed using those tools. We can use open source Python coding for Rpi available on Internet for developing the codes for hardware.

Rpi system desktop is like a Windows desktop through which we can access the Internet, LAN, wifi devices, Bluetooth devices through ARM 11 processor. We can use the keyboard and mouse on desktop of Rpi like PC. It operates on Linux operating system. Rpi board can be act as real time dedicated system by loading our

ISSN: 2277-9655
(I2OR), Publication Impact Factor: 3.785

code on root. Features like easy to interface, easy to operate and easy to get information on Internet lead us to use this ARM 11 board for our purpose. DHT11 Sensor is used for sensing the humidity and temperature. Rather than using different sensors for humidity and temperature, we decided to use the combined sensor for our work. The signal output of DHT11 is digital as well as it is accurate, stable and lab calibrated, for this reason DTH11 sensor is preferred for this system. Big advantag is that, it is relatively inexpensive for the given performance.

In this work, we are going to build a weather measurement station that will automatically send data to the cloud using the Temboo Google service. For remote monitoring, we like to connect our sensing devices wirelessly especially using WiFi and to the web. One solution is to use WiFi breakout boards or shields, for example using the CC3000 WiFi chip, connect it to an Rpi board, and build our own code for the WiFi communication & remote interface.

PROPOSED SYSTEM

Our proposed system has extension in its concepts in an successful way, this is an attractive thing of our systm. To promote the sensor network for cloud based data storage service, we decided to work on the specific application and example of sending data of real time humidity and temperature over Google cloud and monitoring it over Internet. The existing real time sytems monitor the parameters with less accuracy due to the improper time interval between the updation of sensed value of temperature & humidity. So we are mainly focused on this limitation of the existing systems & by selecting proper time inerval between the updation of sensed value of parameters our proposed system is more accurate than the existing systems.

REQUIRED TOOLS

- Sensor circuit.
- Embedded system.
- wireless / Internet Interface.
- Google Cloud system.

SYSTEM ARCHITECTURE

The system model consists of ARM 11 Rpi Processor, Cloud system, sensing circuit, wireless/Internet Interface. The main intention of our proposed system is to design a Embedded based circuit which can monitor humidity & temperature continuously on cloud. The connectivity between the various above components of system is shown in fig 1.

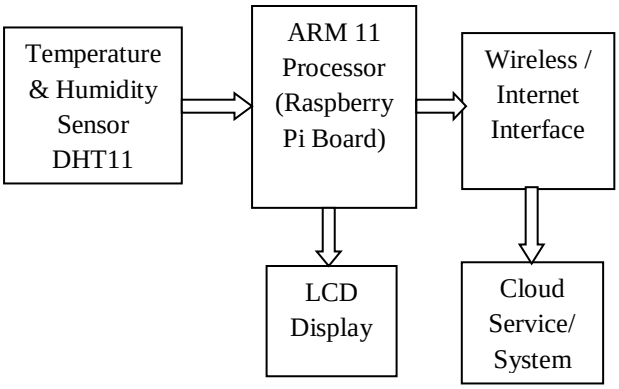


Fig 1. Block Diagram of The System

Block diagram includes ARM 11 processor, Humidity and temperature circuitry and Internet connecting Port for ARM board. Continuous signals can be given to the processor through the analog terminal which can convert into the digital signal and processed to send through internet in to packet format. The processed data will be sent to cloud server using web access. It requires python programming for web design. Python programming a also suitable for ARM 11 processing using Linux operating system. It can intimate irrelevant data or uncontrolled data through cloud or through mail or through sms to particular person in case of disaster management.

HARDWARE DESCRIPTION

A. Sensing Circuit

Sensing circuit consist of DHT11 sensor having temperature and humidity sensing capability. It is constructed by analog conditioning and analog to digital serial converter. Serially data can be connected to processor MCU as shown in fig 2. DHT11 is the combined sensing device which can send the humidity as well as temperature data serially [14][16].

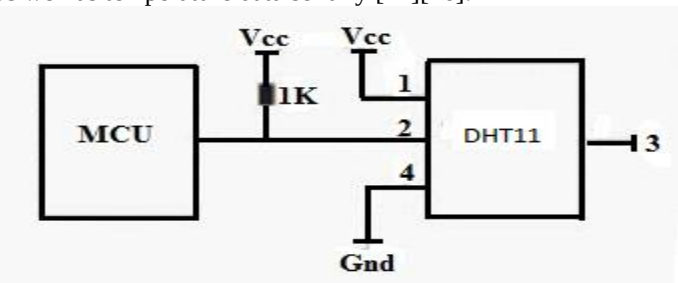


Fig 2. DHT11 Sensor Interface with Processor

The practical view of DHT11 sensor is as shown in fig 3.

ISSN: 2277-9655
(I2OR), Publication Impact Factor: 3.785

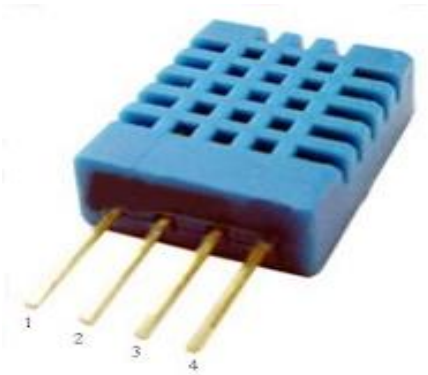


Fig 3. Practical View Of DHT11 Sensor[16]

Table 1 shows the pinout of the DHT11 sensor[16].

Pin	Name	Description
1	VDD	Power supply 3 - 5.5 V DC
2	DATA	Serial Data Output
3	NC	Not Connected
4	GND	Ground

Table 1. Pin Description of DHT11 sensor

B. ARM 11 Rpi Processor

Over the 90% of the embedded market is based on ARM architecture. Over $100 billion USD ARM Ltd. makes annually in licensing & royalties fees for this technology. More than two billion units are shipped per year. ARM stand for Advanced RISC Machine. ARM11 is belongs to an ARM architecture of 32 bit RISC microprocessor cores. The ARM11 is mainly function of the ARMv6 instruction set architecture, which include new cache architecture, multiprocessor support & SIMD media instructions. The system implimentation includes a significantly improved instruction processing pipeline & can be used in smart phones from Nokia, Apple and others as compared to previous ARM10 or ARM9 families. It supports Bi-endian, which can operate in either big-endian or little endian format. Today, no of the devices uses little-endian format. Little-endian format uses the 32 bit ARM, 16 bit Thumb & two instruction set. Sometimes many embedded devices small amounts of memory, i. e. 16 bit instruction set can be used. It makes use of implied operands and reduced functionality which helps to reduce the code size. At execution time, Thumb instructions are decoded into ARM instruction on fly by consuming an additional cycle. CPU is either "Thumb state" or "ARM state", which consist of 17 general purpose "mode specific" register, 7 status registers & 16 general purpose resisters. There are one status register for each operating mode. The main hardware parts required for ARM processor are given below [3][21].

1. Raspberry Pi Model B+

The Raspberry Pi is low cost ARM based palm-size computer. ARM processor operates at 700 MHz & it has 512 MB RAM. It consumes 5V electricity at 1A current due to which power consumption of raspberry pi is less. It has many peripherals such as USB port, 10/100 ethernet, GPIO, HDMI & composite video outputs and SD card slot. SD card slot is used to connect the SD card which consist of raspberry linux operating system. Fig 4 shows the block diagram of Raspberry Pi.

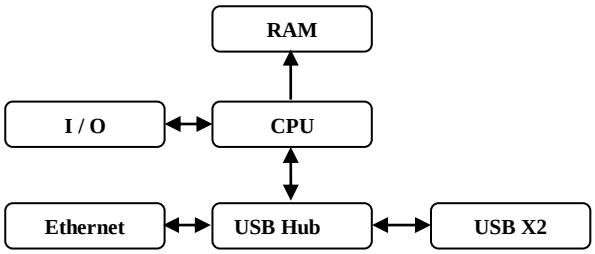


Fig 4. Block Diagram of Raspberry Pi [1]

The Raspberry Pi is a small, powerful and lightweight ARM based computer which can do many of the things a desktop PC can do. The Raspberry Pi is built using ARM11 processor. The Raspberry Pi is based on a Broadcom BCM2835 chip. The Raspberry Pi is a credit card sized computer that runs the freely available Linux Operating System. It is powered by a typical mobile phone charger using a micro USB connector, which can supply at least 700 milli-amps. The Raspberry Pi can be connected to a TV using an HDMI cable although an analogue connection is also available. The powerful graphics capabilities and HDMI video output make it ideal for multimedia applications such as media centers and narrowcasting solutions. It does not feature a built-in solid-state drive or hard disk, instead relying on an SD card for long-term storage & booting [2].

The practical view of Raspberry Pi B+ model as shown in fig 5.

ISSN: 2277-9655
(I2OR), Publication Impact Factor: 3.785

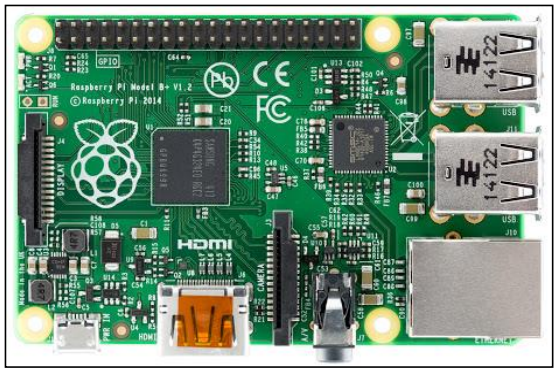


Fig 5. Raspberry Pi B+ Model

Raspbian is optimized for the Raspberry Pi hardware which is based a Debian-based free operating system [3].

2. Output Display for Rpi ARM 11

The output of the proposed system should be seen on LCD which has HDMI connection facility. The HDMI connector is used to connect monitor of personal computer. Nowadays, HDMI connection facility is also available on LCD and LED Televisions. Fig 6 shows the HDMI connector & LCD display.

Fig 6. HDMI Connector & LCD Display

HDMI (High-Definition Multimedia Interface) is a compact audio/video interface for transferring uncompressed video data and compressed or uncompressed digital audio data from a HDMI-compliant source device, such as a display controller, to video projector, a compatible computer monitor, digital audio device or digital television. For existing analogue video standards, HDMI is a digital replacement. HDMI implements the standards, which define waveforms & video formats, transport of uncompressed, compressed and auxiliary data, LPCM audio and implementations of the VESA EDID. That HDMI standards are EIA/CEA-861 standards [20].

A. Wireless and Internet Interface

Wifi module can be interface to Rpi for Internet connectivity. The CC3000 WiFi chip from Texas Instrument is a quite versatile chip that can be used to connect the system to the web. However, connecting Rpi project to a web server can be tricky for which we need to know how to install & configure a web server, and know a bit about HTML & PHP. Here how to connect a temperature & humidity sensor to an online platform for connected objects, Google cloud. The sensor will be connected to an Rpi board, which will also communicate with the Adafruit CC3000 breakout board for the WiFi connectivity. But the CC3000 chip will communicate directly with the Google cloud server and send the data, instead of communicating with a local server. Finally, just by logging into the Google cloud website, wherever we are in the world we will be able to monitor the data sent by the server directly from our browser [18].

B. Cloud System

Cloud computing has become the rapidly growing area in industry today with the advancements in the field of science and technology. Cloud computing by using the resources, information, software and shared equipment provides a client's service within a specific time. Cloud computing is a way to increase the capacity or add capabilities dynamically without investing in new infrastructure, training new personnel, or licensing new software. It extends Information Technology's (IT) existing capabilities. There are cloud service providers who provide large scale computing infrastructures as services in which the users can use it according to their requirements. In present day scenario of the network cascaded with task consummation, and the aspect of heterogeneity along with platform divergence, dynamic load balancing plays a vital role in optimizing the performance of the server in the cloud computing environment [12].

Google DOC Cloud is used for data storing purpose. Before we can upload data to Google Docs, we need to have a Temboo account. We have to enter our email over to the Temboo website to start the account creation process as shown in fig 7.

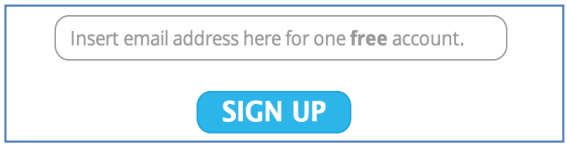


Fig 7. Temboo Account Creation

ISSN: 2277-9655
(I2OR), Publication Impact Factor: 3.785

Then, create a new spreadsheet in Google Docs, name it and set the title of each column in the first row. Finally, we are ready to test the project. Upload the code to the RPI ARM 11 board, open the Google Docs spreadsheet again in your browser, and wait a moment. Then after sometime, the first measurement should appear as shown in fig 8 [14].

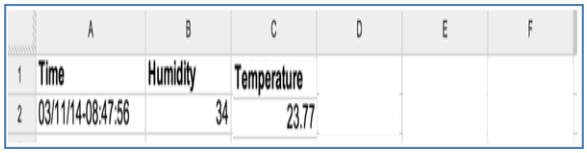


Fig 8. Spreadsheet in Google Docs

SOFTWARE DEVELOPMENT

Some specific Software Libraries are required for the development the Python codes for this application. Initially we will discuss about the Libraries installation process and then we will see the code development process.

A. Software Library Install

Python codes for both the modules are developed as per the requirements. For running this code the different libraries are required to install. To install the Python library on Raspberry Pi, we first need a few dependencies. To install these dependencies, execute the following command (assuming you're using Raspbian on the Pi). Raspbian is the operating system for Rpi which is based on Linux. Library installation steps are as follows [15].

Copy the code as follows and sun in command prompt on Rpi.sudo apt-get upgradesudo apt-get install build-essential python-dev

sudo apt-get upgrade

sudo apt-get install build-essential python-dev

If there are error occur then that a package is already installed or at the latest version, you can ignore it and move on.

2) Next, to install the library execute, copy Code as follows and sun in command prompt on Rpi.sudo apt-get upgradesudo apt-get install build-essential python-devsudo python setup.py install

sudo python setup.py install

This should compile the code for the library and install it on Rpi device so any Python program can access the Adafruit_DHT python module. To test the Python library you can run some of the example programs in the examples folder. The AdafruitDHT.py example is a simple program which takes from the command line parameters the type of sensor DHT 11 and GPIO pin connected to the sensor, and displays a single reading from the sensor. We are going to modify this command for fast accessing.

First navigate to the examples folder by executing:

cd examples

Now to run the example on a Raspberry Pi with an AM2302 sensor connected to GPIO #4, execute:

sudo ./AdafruitDHT.py 2302 4

Make sure to run the command as root with the sudo command as shown above or else the program will fail to run (root access is required for reading and writing the GPIO pins). After the program executes, you should see both the temperature and humidity displayed once. Double check you have the right GPIO pin connected to the data line of the DHT sensor and specified in the last parameter if you see an error that the sensor could not be read. Note that sometimes you might see an error that the sensor can't be read and to try again, even if you have your connections setup correctly. This is a limitation of reading DHT sensors from Linux--there's no guarantee the program will be given enough priority and time by the Linux kernel to reliably read the sensor. If this occurs, run the program again (or call the read function again in your code) to try to get a new reading. In testing on Raspberry Pi, about 75% of the read requests should generally succeed and return a result (assuming the board is not under heavy load). Examine the source code for AdafruitDHT.py and simpletest.py to see simple examples of reading the DHT sensors from Python code [15].

B. Steps for creating the Google Docs Cloud are as follows

To create and prepare spreadsheet, first up you will need to sign up for Google Docs and create a spreadsheet. We're going to call ours DHT Humidity Logs. Once you've created it, delete all but one line since we don't want more empty rows [15].

ISSN: 2277-9655
(I2OR), Publication Impact Factor: 3.785

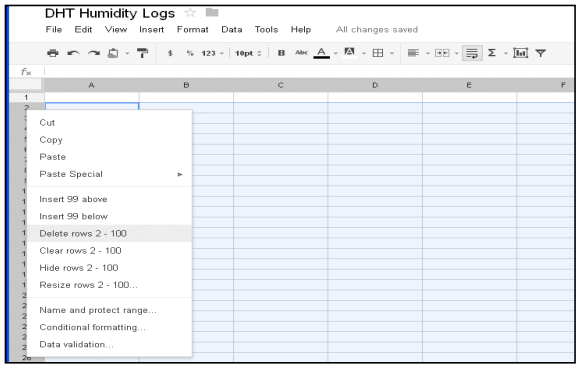


Fig 9. Creation of Spreadsheet in Google Docs

Then make the one remaining line a header with row names as follows shown in fig 10.

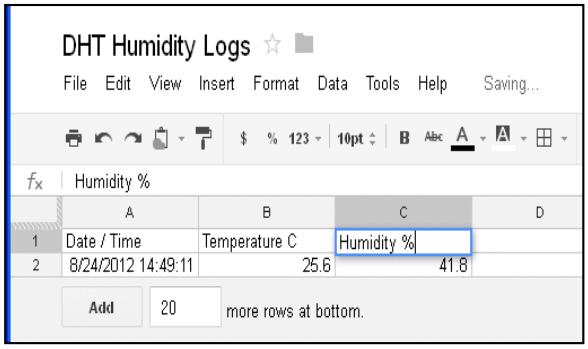


Fig 10. Created Spreadsheet in Google Docs

C. Python Code Design and development

Cloud code for humidity and temperature real time data is designed by combining the code for humidity sensing, code for temperature sensing and code for data transfer to Google cloud through wifi Internet connection. Code is divided in to two parts 1) Code for sensing system and 2) Code for cloud logging. We then combined the code as per our requirement to get more efficiency.

1. Run Python Code for Sensing and data logging combine

Steps are:

1) First up we will have to install the gspread python library, which will do the heavy lifting of connecting to google docs and updating the spreadsheet. With our Rpi board connected and online, run the following code.sudo apt-get install python-pipsudo pip install gspread

sudo apt-get install python-pip

sudo pip install gspread

2) Assign Username and password.
Next, in the examples directory again, edit code for cloud and adjust the configuration values towards the top of the file as,

Type of sensor, can be Adafruit_DHT.DHT11,Adafruit_DHT.DH 22, or Adafruit_DHT.AM2302.

DHT_TYPE = Adafruit_DHT.DHT11

Example of sensor connected to Raspberry Pi pin 23

DHT_PIN = 23

Google Docs account email, password, and spreadsheet name.

GDOCS_EMAIL = 'your google docs account email address'

GDOCS_PASSWORD = 'your google docs account password'

GDOCS_SPREADSHEET_NAME = 'your google docs spreadsheet name'

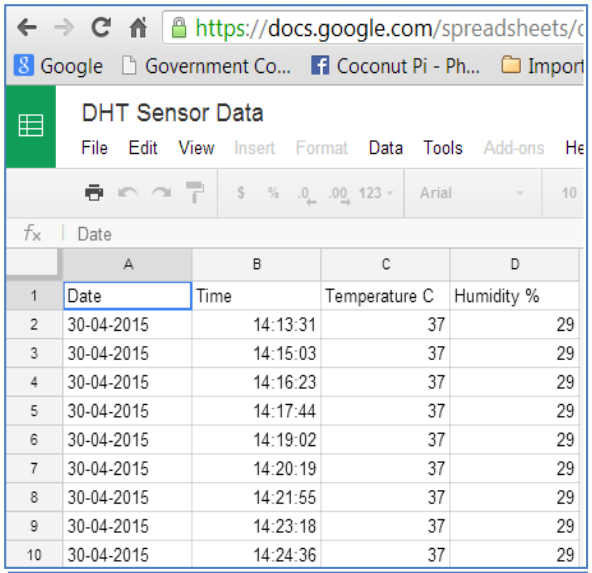
RESULTS AND DISCUSSION

In this section we provide the detailed results of the experiments carried out using the proposed System. The following monitoring results are obtained using temperature & humidity sensor. These real time monitoring results are recorded on server. The monitoring of temperature & humidity are plotted for the different days & months of 2015. Result shows the real time snap from google cloud spreadsheet.

A. Result1

The result1 shows the real time values of temperature & humidity recorded on 30th April 2015.

ISSN: 2277-9655
(I2OR), Publication Impact Factor: 3.785



DHT Sensor Data

	A	B	C	D
1	Date	Time	Temperature C	Humidity %
2	30-04-2015	14:13:31	37	29
3	30-04-2015	14:15:03	37	29
4	30-04-2015	14:16:23	37	29
5	30-04-2015	14:17:44	37	29
6	30-04-2015	14:19:02	37	29
7	30-04-2015	14:20:19	37	29
8	30-04-2015	14:21:55	37	29
9	30-04-2015	14:23:18	37	29
10	30-04-2015	14:24:36	37	29
13	30-04-2015	14:28:39	37	29
14	30-04-2015	14:29:58	37	29
15	30-04-2015	14:31:28	37	29
16	30-04-2015	14:32:45	37	29
17	30-04-2015	14:34:05	38	29
18	30-04-2015	14:35:20	38	29
19	30-04-2015	14:36:43	38	29
20	30-04-2015	14:38:03	38	29
21	30-04-2015	14:39:19	38	29
22	30-04-2015	14:41:03	38	29
23	30-04-2015	14:42:22	38	29
24	30-04-2015	14:43:42	38	29
25	30-04-2015	14:45:17	38	29
26	30-04-2015	14:46:53	38	28
27	30-04-2015	14:48:17	39	27
28	30-04-2015	14:49:36	39	27
29	30-04-2015	14:50:52	39	27
30	30-04-2015	14:52:28	38	28
31	30-04-2015	14:54:14	38	28
32	30-04-2015	14:55:38	38	28
33	30-04-2015	14:57:06	38	28
34	30-04-2015	14:58:25	38	28
35	30-04-2015	14:59:49	38	28

Fig 11. Google DOC Cloud Logged Error Free Data Captured On 30/04/2015

The graph can be plotted for the recorded real time data given in table no.2. The respective graph is as given in fig 12 & fig 13.

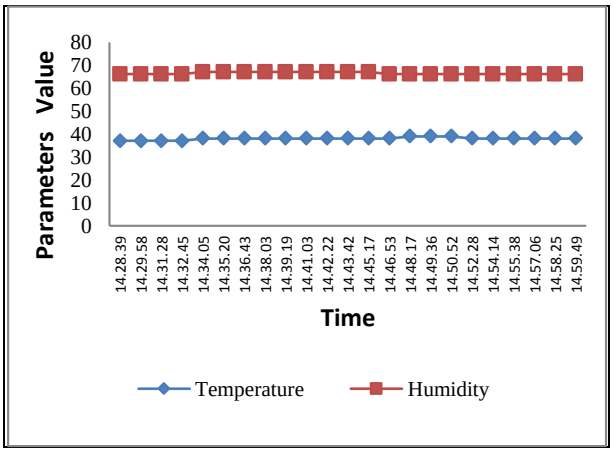


Fig 12. Graphical Representation1 of Data Captured On 30/04/2015

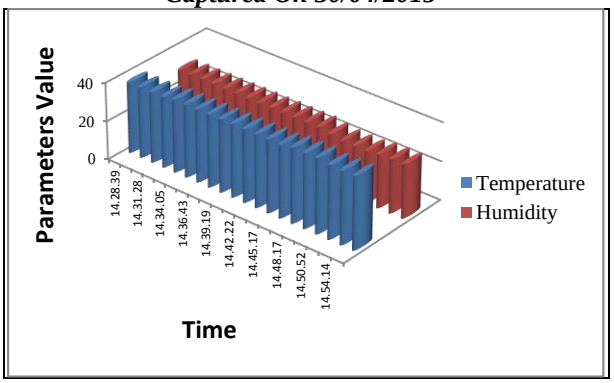
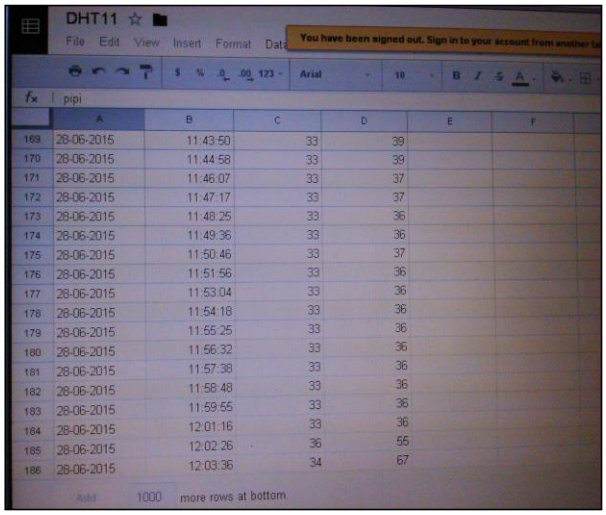


Fig 13. Graphical Representation1 of Data Captured On 30/04/2015

A. Result2

The result2 shows the real time values of temperature & humidity recorded on 28th June 2015.



DHT11

	A	B	C	D
169	28-06-2015	11:43:50	33	39
170	28-06-2015	11:44:58	33	39
171	28-06-2015	11:46:07	33	37
172	28-06-2015	11:47:17	33	37
173	28-06-2015	11:48:25	33	36
174	28-06-2015	11:49:36	33	36
175	28-06-2015	11:50:46	33	37
176	28-06-2015	11:51:56	33	36
177	28-06-2015	11:53:04	33	36
178	28-06-2015	11:54:18	33	36
179	28-06-2015	11:55:25	33	36
180	28-06-2015	11:56:32	33	36
181	28-06-2015	11:57:38	33	36
182	28-06-2015	11:58:48	33	36
183	28-06-2015	11:59:55	33	36
184	28-06-2015	12:01:16	33	36
185	28-06-2015	12:02:26	36	55
186	28-06-2015	12:03:36	34	67

Fig 14. Google DOC Cloud Logged Error Free Data Captured On 28/06/2015

ISSN: 2277-9655
(I2OR), Publication Impact Factor: 3.785

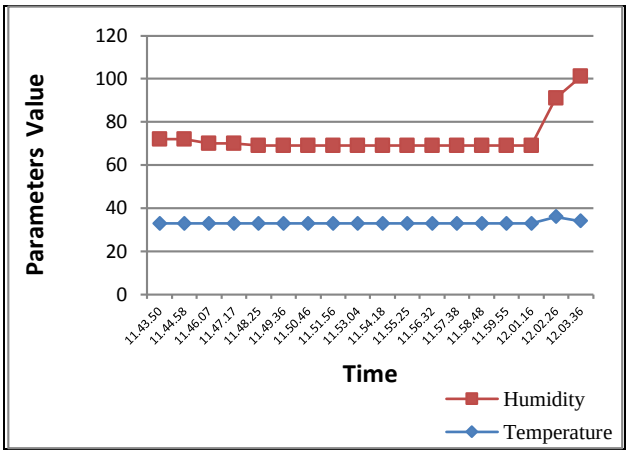


Fig 15. Graphical Representation2 of Data Captured On 28/06/2015

The graph can be plotted for the recorded real time data given in table no.2. The respective graph is as given in fig 15 & fig 16.

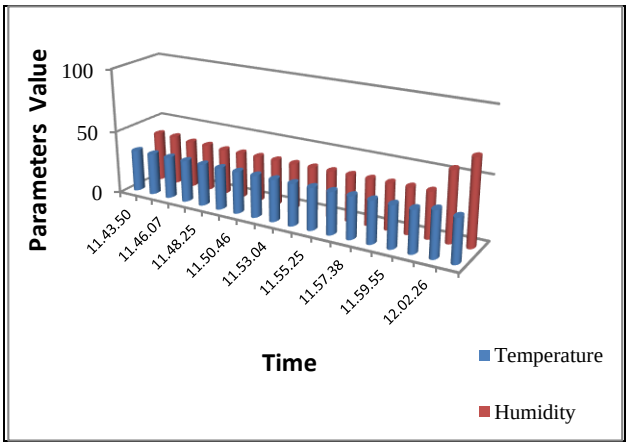


Fig 16. Graphical Representation2 of Data Captured On 28/06/2015

During this experiments, it was observed that the system successfully updated the relative humidity and temperature of the boxes in the Google spreadsheet.

CONCLUSION

It is concluded that the system is developed for remote humidity and temperature sensing using DHT11 sensor, Rpi ARM 11 board and Google cloud service. We developed the system so that we can reduce the error for data sensing using modifying the existing codes for humidity and temperature sensing. About 80% (normally having 75%) of the read requests are found succeed by observing the cloud data. The proposed system will be very useful for the use with existing system. But the usefulness of the system will be for disaster management. The system will be cost effective, speedy for real time operation and helpful to the society. Newest technique will be used for remote monitoring using cloud logging.

REFERENCES

[1] Girish Birajdar, "Implementation of Embedded Web Server Based on ARM11 and Linux using Raspberry PI" International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-3 Issue-3, July 2014. PP: 64-66.

[2] UNNATI P PATEL, PROF. SUGNESH D. HIRPARA "SURVEY ON RASPBERRY PI FOR INDUSTRIAL SENSOR" INTERNATIONAL JOURNAL OF PURE AND APPLIED RESEARCH IN ENGINEERING AND TECHNOLOGY. (IJPRET) ISSN: 2319-507X 2015; Volume 3 (7) PP: 98-104.

[3] BHARGAV GORADIYA, H.N.PANDYA "Real time Monitoring & Data logging Systemusing ARM architecture of Raspberry pi & Ardiuno UNO" International Journal of VLSI and Embedded Systems-IJVES. ISSN: 2249 – 6556. Vol 04, Article 06118; July 2013. PP: 513-517.

[4] Mayur Randhir, R. R. Karhe, "Monitoring Of Environmental Parameters by Using Cloud Computing" International Journal of Computer Science Trends and Technology (IJCST) – Volume 3 Issue 3, May-June 2015. PP: 151-155.

[5] Nelson Gonzalez, Charles Miers, Fernando Red´ıgolo, Marcos Simpl´ıcio, Tereza Carvalho, Mats N¨aslund and Makan Pourzandi, "A quantitative analysis of current security concerns and solutions for cloud computing" Journal of Cloud Computing: Advances, Systems and Applications 2012, 1:11.

[6] Mahesh D. S, Savitha S, Dinesh K. Anvekar "A Cloud Computing Architecture with Wireless Sensor Networks for Agricultural Applications" International Journal of Computer Networks and Communications Security VOL.2, NO.1, JANUARY 2014, 34–38 Available online at: www.ijcncs.org ISSN 2308-9830.

[7] Prof C. H. Chavan, Mr.P. V.Karande, "Wireless Monitoring of Soil Moisture, Temperature & Humidity Using Zigbee in Agriculture" International Journal of

Engineering Trends and Technology (IJETT) – Volume 11 Number 10 - May 2014.

[8] D.D.Chaudhary, S.P.Nayse, L.M.Waghmare, "APPLICATION OF WIRELESS SENSOR NETWORKS FOR GREENHOUSE PARAMETER CONTROL IN PRECISION AGRICULTURE" International Journal of Wireless & Mobile Networks (IJWMN) Vol. 3, No. 1, February 2011.PP: 140-149.

[9] S.Thenmozhi, M.M.Dhivya, R.Sudharsan, K.Nirmalakumari, "Greenhouse Management Using Embedded System and Zigbee Technology" International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering". ISSN (Print): 2320 – 3765, Vol. 3, Issue 2, February 2014. PP-7382-7389.

[10] Y.R.Dhumal, J.S.Chitode, "Green House Automation using Zigbee and Smart Phone" International Journal of Advanced Research in Computer Science and Software Engineering, ISSN: 2277 128X, Volume 3, Issue 5, May 2013. PP: 495-501.

[11] Ibrahim Al-Adwan, Munaf S. N. Al-D, "The Use of ZigBee Wireless Network for Monitoring and Controlling Greenhouse Climate" International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-2, Issue-1, October 2012.PP: 35-39.

[12] Anindita Kundu, "AN EFFICIENT FUZZY LOAD BALANCING ALGORITHM FOR PUBLIC CLOUDS" INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY. ISSN: 2277-9655, June, 2015. PP-30-34.

[13] Prof. Pranit P. Kathale*, Prof. P. W. Pawade, Dhanashri Gawande, Pradnya Ingle, "PRACTICAL RESULTS ON HARDWARE IMPLEMENTATION OF MONITORING AND CONTROLLING SYSTEM FOR THE OPERATION OF GREENHOUSE ENVIRONMENT" INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY. ISSN: 2277-9655, May, 2015. PP-502-509.

[14] Marco Schwartz, "Internet of Things with the Arduino Yún" Packt Publishing 2014.

[15] DHT Humidity Sensing on Raspberry Pi or Beaglebone Black with GDocs Logging, Adafruit Learning System.

[16] DHT11 Humidity & Temperature Sensor, D-Robotics UK. (www.droboticsonline.com).

[17] "Raspbian."[Online].Available:http://www.raspbian.org/down loads. Accessed on 2nd July 2015.

[18] https://www.youtube.com/watch?v=Opx8wzEuvhU. Accessed on 3rd July 2015

[19] https://en.wikipedia.org/wiki/HDMI. Accessed on 3rd July 2015.

[20] https://www.raspberrypi.org/magpi-issues/MagPi31.pdf. Accessed on 4th July 2015.