Implementation of a Forest Monitoring and Alerting System

Akhil Gunda[1], Shreya Lokray[2], Vaishnavi Suthram[3]

[1-3]Department of ECE, Chaitanya Bharathi Institute of Technology(A), Hyderabad, India

---

Abstract - *In many parts of India where sandalwood-a priced commodity is grown, illegal chopping, and smuggling of trees has become a big menace since the past few decades. This has led to a drastic loss in the tree population as well. To tackle this, a forest monitoring and alerting system is essential. Currently, monitoring mostly takes place manually by guards in many areas. This is a very tedious and inefficient task. In the proposed paper this problem is solved by creating a network of sensors with a controller forming a node and sending the data to a more powerful central station which is the Raspberry Pi. Data from the multiple sensors is collected at the controller present at the node. This data is then transmitted to the central station where the data will be processed. The code at the central station will include the conditions when an alert has to be sent. The concerned officials are notified through an E-mail and a webpage will also be used to view the status of the sensors. This model is an efficient way of reducing human interference and error involved in the surveillance of large areas, thereby reducing the problem of illegal chopping of trees.*

Key Words: **MQTT (Message Queue Telemetry Transport), NodeMCU, Raspberry Pi, SMTP (Simple Mail Transfer Protocol), Wireless Sensor Network.**

1. INTRODUCTION

Poaching of trees is an important issue that is faced globally. In India itself, more than 1 Lakh metric tons of trees are poached every year. There is a serious need for monitoring our forests and the current monitoring systems in place are mostly manual. It is practically very difficult to manually keep a check on the large forest areas which leads to illegal trade of trees. Sandal wood and red sander trees are among those that are most prone to be poached. They are known for their medicinal properties, which is why they are in high demand. Many laws have been laid down by the Government of India to penalize those that are caught smuggling trees. This by itself is not enough and should go hand in hand with better systems to detect and alert when any such activity occurs. The existing systems require guards to patrol over the large forest area or employ monitoring via CCTVs, both of which are very straining and not very efficient. Some trees are also being marked with RFID tags but using this method an alarm is generated only when the tree moves from its initial position. In [2] integrated monitoring with satellite remote sensing technology that measures the physical parameters of an area from a long distance.

In the proposed paper a Wireless Sensor Network (WSN) model is employed that transmits the data without the hassle of wires and cables, which is important for setting up the model in forests. WSN is an application of Internet of Things (IoT) in which different sensors are used to measure their surroundings and these sensor nodes may be placed at various locations. The data collected is transmitted to the next designed node or the main node where the authorities receive it. There are many methods with which the data can be transferred from one point to another. In [3] GSM technology is used to transmit data from sensors and send alert messages to the officials. In [4] Zigbee technology is used which communicates to the far-off central station through intermediate devices. Work done in [9] adds an insight into how a WSN system is a more efficient method for large scale forest monitoring and alarming.

In the proposed model a sensor node that can be attached to a tree is developed. This sensor node comprises of four sensors namely LM393 (Sound Sensor), ADXL345 (Accelerometer), DHT11 (Humidity and Temperature Sensor), SW520D (Tilt Sensor) and a controller (NodeMCU) with a Wi-Fi chip ESP8266 (NodeMCU) embedded on it. Using these four sensors ensures that various observations while cutting a tree i.e. chopping sound, vibration of the trunk, fire and tilting of the tree, can generate an alarm. The sensors are interfaced with the NodeMCU and all the data from the sensors is collected at the NodeMCU. The NodeMCU then transmits the data to the Raspberry Pi 4 Model B controller at the central station. The Raspberry Pi is placed at the central station due to its high-speed processing and large memory among other benefits. The transmission of data between NodeMCU and Raspberry Pi is done using the MQTT (Message Queue Telemetry Transport) Protocol. This protocol is mostly used for remote area monitoring and provides guarantee of message delivery. It is also an easily scalable and cost-efficient method. MQTT also ensures security by checking and rechecking the authorization of the subscribers and publishers.

Once the data is processed if any abnormal values are identified from the sensors, then an alert needs to be sent to authorities. In [8] the final alerting message is displayed on an LCD screen. Alarm through SMS messages using a GSM system is preferred in work [7]. In [6] a webpage is used to display the information regarding the total tree count of the forest. The proposed model in this paper alerts via E-mail that is sent to the concerned mail ids. An alarm is also generated if the node or a particular sensor stops working.

In addition to this, a webpage has been created to view and analyze the values at the nodes.

2. RELATED WORK

There is a lot of research going on currently for systems that can protect our forests from poaching.

Y. G Sahin and T. Ince [1] have presented their work on Early Forest Fire Detection Using Radio-Acoustic Sounding System. The main objectives in this paper are the low energy consumption and the reliability of the system.

Yuenew Jiang's [2] work on using satellite navigation and communication system for detection of forest fires can offer profile information of appearance due to the requirements of distance and resolution.

Shridevi Soma and Swamy Sudha [3] worked on An Automatic System for Controlling Deforestation using the concepts of GSM and IoT. The requirements here are a continuous supply of power which may increase the cost factor.

Huanqi Tao and Heng Zhang [4] worked on Forest Monitoring Application Systems Based on Wireless Sensor Networks using ZigBee. Repeaters have been used in order to enhance the range of the system.

K. Muruganantham, T. Mythili, S. Thilagavathi and R. Prabhakaran [5] have suggested an arrangement of system in the forest using WSN concepts for the protection of trees. A very efficient mode power supply is used for this system.

S. Syed Husain, K. Vigneshwaran, S. Kaveya [6] have proposed a forest monitoring system that works on IoT. Here, Arduino microcontroller is connected with five sensors to determine the condition of abnormality and then an alarm is generated. The messages are displayed on an LCD screen.

Pushpalatha.R and Darshini.M. S [7] have proposed a framework using Renesas microcontroller as the core which is connected to suitable sensors. The technology used here to alert is GPRS system which in turn depends on GSM for sending the messages in the form of SMS.

Mallikarjun.N, Shilpa.N, Renuka.D, Sindhu.T, Sridevi Malipatil [8] have worked on developing a suitable positioning of system of sensors to tackle poaching of valuable trees. Alerting the authorities is done here using ZigBee or GSM.

T. Saikumar and P. Sriramya [9] have worked on a system to detect and alert forest fires using IoT. Arduino device is used here to detect forest fires with suitable sensors and along with that GSM technology is used to send alerts.

3. METHODOLOGY

The architecture for the proposed forest monitoring and alerting system is shown in figure 1.

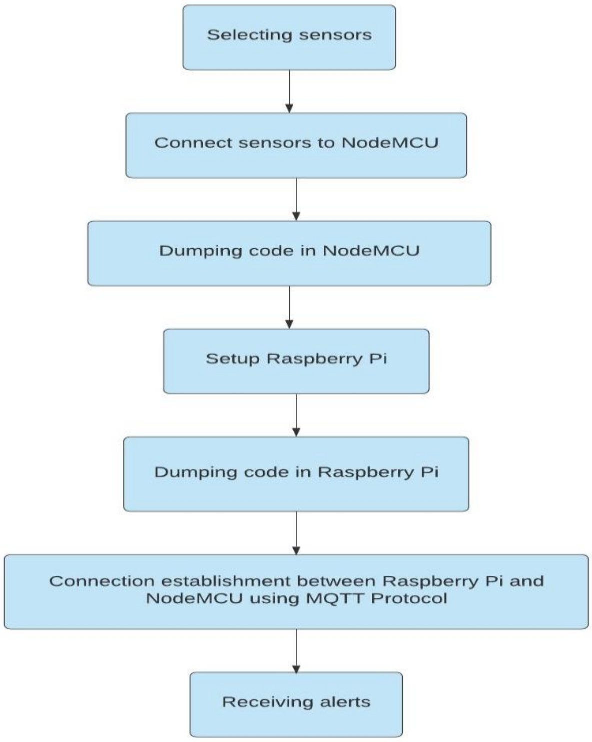

Fig -1: Architecture of the Proposed Work

It contains various steps:

3.1. Data Sensors

The main aim of the project is to collect real-time data of the trees like tilt, temperature, humidity, sound, and process it further to produce an alarm.

The sensors are selected based on the criteria that would help recognize intruder activity. In the proposed project, a sensor node is developed that can be attached to a tree. This sensor node comprises four sensors namely Temperature and Humidity (DHT11) sensor, Accelerometer (ADXL345) sensor, Sound (LM393) sensor, and Tilt (SW520D) sensor. The data collected from various sensors is done at individual nodes. Finally, the sensors are interfaced with the NodeMCU and all the data from the sensors is collected at the NodeMCU.

The sensors used in this proposed work are shown in figure 2.

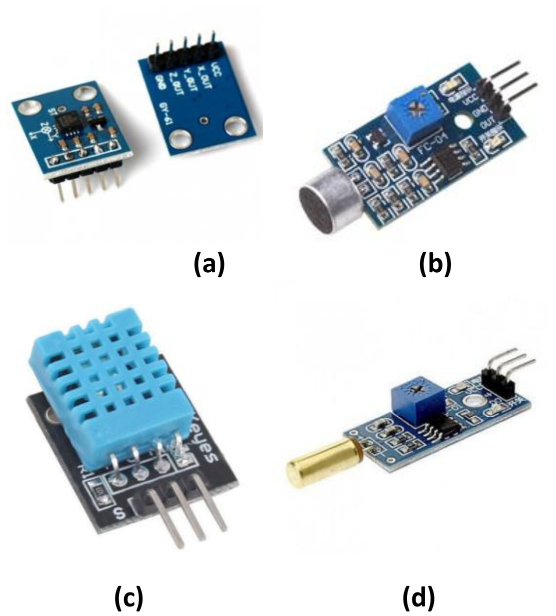

Fig -2: Various Sensors, (a) Accelerometer, (b) Sound sensor, (c) Humidity and Temperature sensor, (d) Tilt sensor

3.2. Connecting to Nodemcu

The node module in the forest monitoring project uses a NodeMCU to transmit the data collected by the sensors to the central station. This transmission is possible by exploring the functionality of the ESP8266 chip present in it. NodeMCU is an open source platform with hardware design that is open to build and thus multiple sensors can be attached to it. Arduino IDE is used for programming NodeMCU. A code dumped in it facilitates collection of sensor data from various pins, processing it and then transmitting it. This transmission is done using the MQTT protocol which is capable of delivering messages in near real time and guarantees their delivery.

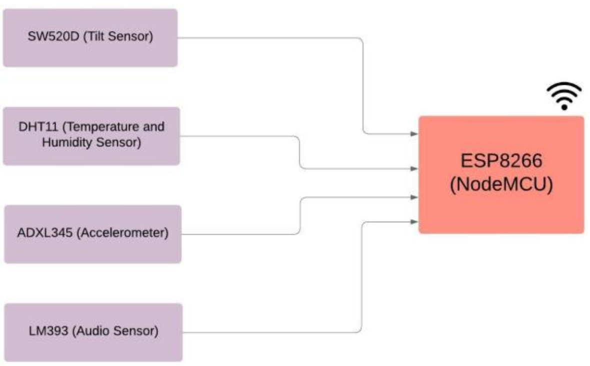

Fig -3: Single node with sensors and NodeMCU

A single node with sensors and the NodeMCU is shown in figure 3. Multiple nodes similar to the one shown send their data to the central station from different locations.

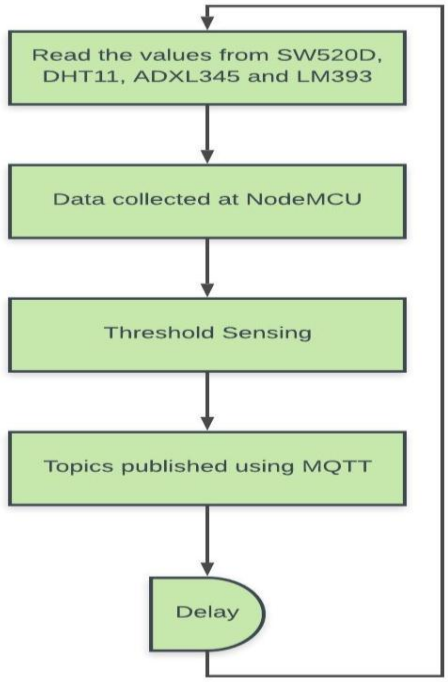

Fig -4: Flow chart for the Input node

The steps shown in figure 4 explain the analysis of the obtained sensor data and the transmission to the central station. First the values are collected from the sensors at the NodeMCU. At the threshold sensing block, the input data from each sensor is compared with their respective thresholds. The data is transmitted to the central station through MQTT.

3.3. Connecting to Raspberry Pi

The processed data at the NodeMCU is transmitted to the central station (Raspberry Pi) using the ESP8266 Wi-Fi chip present on it using MQTT protocol. In this case Raspberry Pi acts as broker as well as subscriber.

MQTT which uses the Publish/Subscribe model keeps the bandwidth at a minimum and deals with unreliable networks without complex error handling. The MQTT protocol itself provides username and password while connecting to the broker while encryption is handled by TLS. This keeps the communication secure. This protocol is most preferred for monitoring in remote areas for low memory, low power consuming small devices.

At the central station the data from various nodes is received by the raspberry pi. It has its own OS which enables writing programs for multitasking. When compared to other controllers available to us like the Arduino (with a CPU processing speed of 17MHz), the Raspberry pi offers a higher CPU processing speed of 1GHz and thus multiple inputs can be processed faster. It is also easier to program due to the presence of already present modules which can be included to the program.

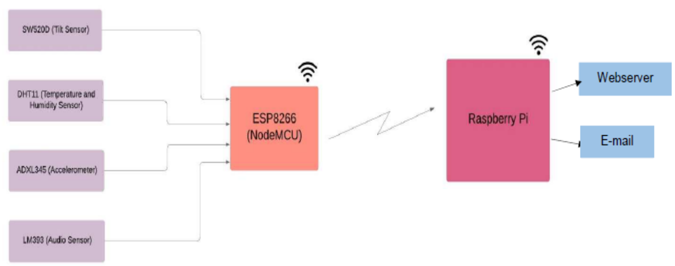

Fig -5: Block Diagram of the System

3.4. Publishing Topics from Nodemcu to Raspberry Pi

After collecting all the data from the sensors to the NodeMCU. Now NodeMCU transmits the received data publishes to the broker or the central station which is Raspberry pi. NodeMCU and Raspberry Pi are connected through a unique Broker ID which ensures a secured transmission of data using MQTT. The figure 6 shows the publishing syntax used in NodeMCU programming.

```
client.publish("/esp8266/temperature", temperatureTemp);
client.publish("/esp8266/humidity", humidityTemp);
client.publish("/esp8266/accelrometer",accelrometer);
client.publish("/esp8266/tilt",msg1);
client.publish("/esp8266/sound",msg2);
client.publish("/esp8266/status",msg3);
```

Fig -6: Publishing the values over MQTT by NodeMCU

3.5. Subscribing Topics by Broker (Raspberry Pi)

Once the values are published by the NodeMCU, the central station or the broker has to subscribe to those values published by the publisher (NodeMCU) before accessing to those values and processing it. The figure 7 shows the subscribing syntax used in broker programming.

```
client.subscribe("/esp8266/temperature")
client.subscribe("/esp8266/humidity")
client.subscribe("/esp8266/accelrometer")
client.subscribe("/esp8266/tilt")
client.subscribe("/esp8266/sound")
client.subscribe("/esp8266/status")
```

Fig -7: Subscribing the topics by Broker (Raspberry Pi)

Now the values are obtained at central station using MQTT successfully.

3.6. Updating Website and Alerting System

The last part of the project is to alert the authorities in case any discrepancies are detected. One way this is done is by sending a mail using the SMTP protocol. It is the most widely used protocol for sending e-mails. The protocol provides a means of transmission of the mail even when the user to which the mail has to be sent is not currently connected to the internet. The mail generates an alarm by notifying about the area or node at which the discrepancies were observed. Another way of alerting the authorities is by updating the real time conditions on a web page at regular intervals of time. The page is designed using HTML and provides a user-friendly page through which the sensor data and the alarm status can be checked.

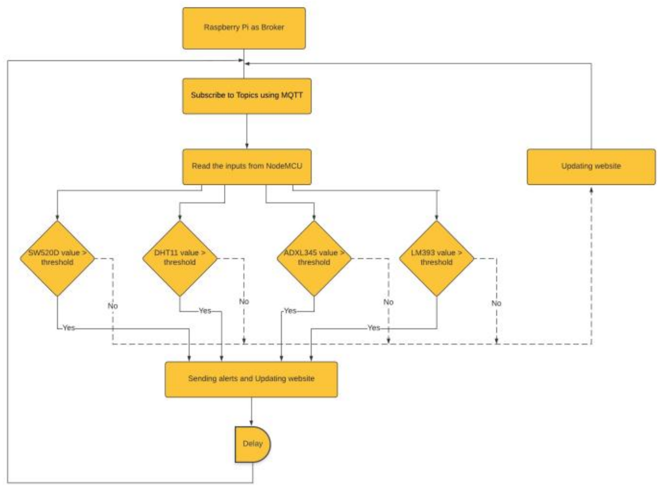

Fig -8: Process flow at the Central Station.

The steps shown in figure 8 explain how the data is received and then processed to generate an alarm i.e. by updating the webpage and sending mail alerts. First Raspberry pi as broker subscribes all the available topics using MQTT and read the input data. When an abnormal activity is indicated both web page is updated and an email is sent to the officials/authorities. With some added delay this loop continues. And the webpage is continuous updated and provides a quite readable window or environment to client at the central station.

4. RESULTS AND DISCUSSION

The results are to illustrate that all the moules are operating correctly without any data loss and each sub-module in all modules are performing their function. The four sensors i.e accelerometer, tilt sensor, sound sensor and temperature-humidity sensor should be able to extract accurate readings from the surroundings. These values should be processed at the node module (NodeMCU) and sent to the Raspberry Pi using the ESP8266 chip present on the nodeMCU. Based on the thresholds set for each sensor, an alarm has to be generated if the recorded value exceeds the threshold. The alarm should be sent in the form of an email to the assigned email address. Simultaneously, the webserver should be updates every 10 seconds to display the current values and alarm status.

Fig -9: Node module

The node module has to be placed on a tree to record various sensor values particular to that tree. To mimic these conditions the node module is placed on a long cylindrical bottle, so that working of all the various alarm conditions may be verified as shown in figure 9.

To ensure that the that all the devices are properly connected and ready to transmit (for the NodeMCU) and receive (for the Raspberry Pi) data the following initial conditions should be verified.

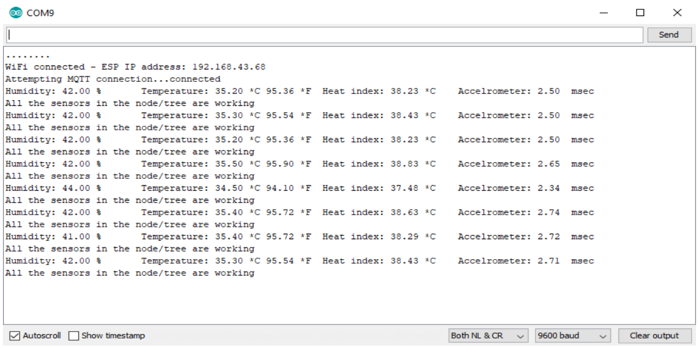

Fig -10: NodeMCU Serial monitor display

For the NodeMCU the first requirement would be for it to recognise available wifi connections and connect to the specified network. The next step is to connect to the MQTT broker i.e. the Raspberry Pi by checking the broker's IP address. The following message is shown if bothe the steps are successful. In case of a failure to connect, the node attempts for the connection again after 5 seconds. Only after both the connections are established, the node publishes the data to the Raspberry Pi. And the values are shown on the serial monitor of the arduino IDE as shown in figure 10.

For the Raspberry Pi once the connections are secured, the data published for different sensors is received with QoS 0 and the message is shown on the monitor as shown in figure 11.

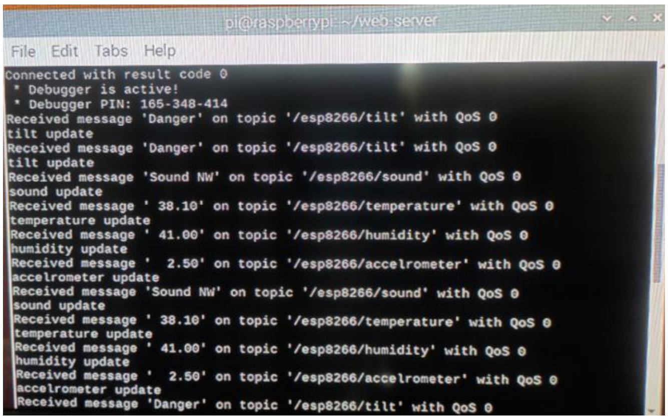

Fig -11: Sensor updates at Broker side (Raspberry Pi)

An SMTP mail is sent to the security personnel as shown whenever an alarm has to be generated. The message is sent if either one of the sensors or all sensors exceed their thresholds and detect an abnormal activity as shown in figure 12.

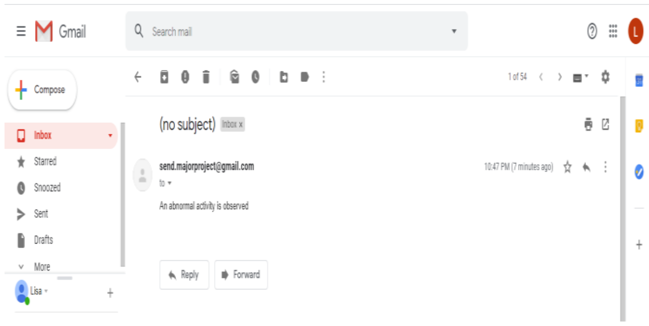

Fig -12: SMTP email for alarm

The following are the results that are observed and updated on the HTML webpage under various situations.

1. The tilt and sound values are under normal ranges(NW-normal working) and so there is no alarm signal. The temperature, humidity and accelerometer values are displayed as shown in figure 13.

2. The tilt sensor value has exceeded the threshold and so a danger message is displayed. The other sensors are in their normal working ranges and their values are displayed as shown in figure 14.

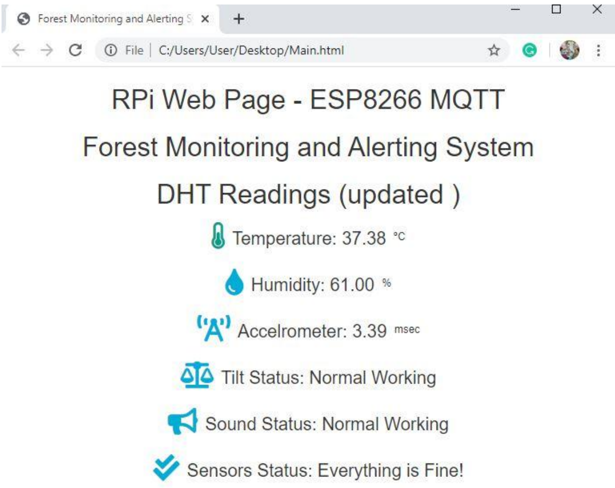

Fig -13: Normal status of sensors

3. The sound sensor has exceeded its threshold and gives a danger message. The other sensors are in their normal working ranges and their values are displayed as shown in figure 15.

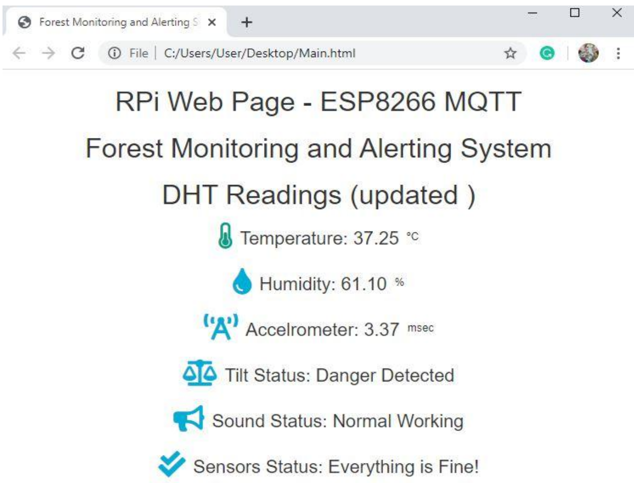

Fig -14: Deviation observed in the Tilt sensor

4. Both the sound sensor and the tilt sensor have exceeded their values and both display the danger message as shown in figure 16.

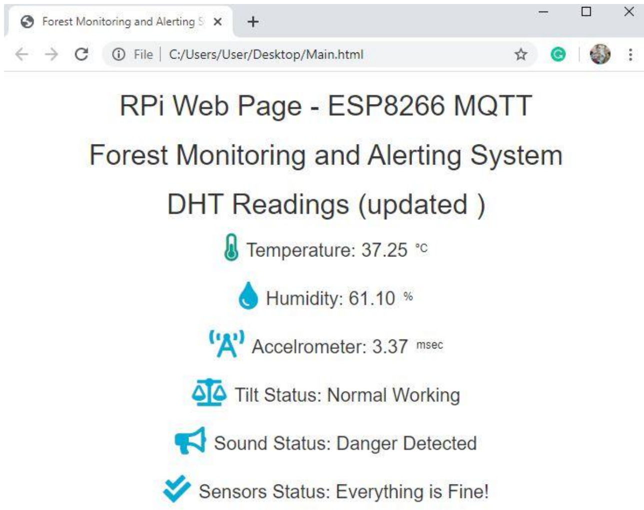

Fig -15: Deviation observed in the sound sensor

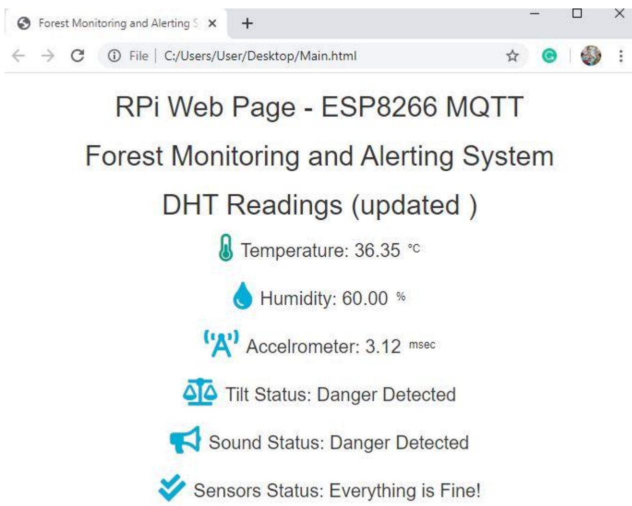

Fig -16: Deviation observed in the Tilt and Sound sensors

5. CONCLUSIONS AND FUTURE SCOPE

The need of a forest monitoring and alerting system arises as the current systems in place are mostly manual. Hence, an automatic forest monitoring and alerting system has been created. Four sensors namely DHT11, SW520D, LM393 and ADXL345 have been used along with ESP8266 and a Raspberry Pi. The sensors have been tested individually and data from them has been used to set necessary thresholds. A circuit comprising of the sensors and the ESP8266 has been made as a single node. Alerts are sent depending upon the status of the sensors. A connection between ESP8266 and Raspberry Pi has been established using MQTT protocol. This protocol has been chosen because it ensures safety of the source and destination addresses, secure transmission of data and a stable connection. Whenever an abnormal condition is detected, an alert in the form of E-mail reaches the concerned officials. A webpage has also been created to see the status of each sensor and analyze the situation.

Further additions could be made to the project in the future like creating a database to store the real time sensor values. This database may then be used to know about general conditions of the area like temperature, humidity, etc. and also to predict those values. This may be useful in predicting rainfall or draught in a certain area. Furthermore, sensors can be added to the system which check sources of infections and attacks by pests. Sensors that measure the CO2 content could also be used to know about the health of the forest. Another advanced scope for the project would be applying audio recognition through advanced machine learning algorithms. In this audio sensors are used to send clips of real time recording to the central station where they are analyzed for sounds similar to predefined sounds like gunshots, chainsaw sounds, etc. These algorithms will help recognize sounds that relate to cutting off of trees and leave out the other noises.

REFERENCES

[1] Y.G Sahin, T. Ince, "Early Forest Fire Detection Using Radio-Acoustic Sounding System", Sensors, vol. 9.3, pp. 1485-1498, March 2009

[2] Yuenew Jiang. "Application prospect of compass satellite navigation and communication system for forest fire", Satellite and Network, 2008

[3] Shridevi. Soma, Swamy Sudha, "An Automatic System for Controlling Deforestation Using IoT and GSM", International Conference on Recent Trends Computational Engineering and Technologies (ICTRCET), pp. 780-785, January 2019

[4] Huanqi Tao, Heng Zhang, "Forest Monitoring Application Systems Based on Wireless Sensor Networks", 2009 Third International Symposium on Intelligent Information Technology Application Workshops, pp. 227-230, 2009

[5] K. Muruganantham, T. Mythili, S. Thilagavathi, R. Prabhakaran, "WSN Based Forest Tree Protection System", South Asian Journal of Engineering and Technology, vol.3, no.4, pp. 44–50, March 2017

[6] Mr. S. Syed Husain, Mr. K. Vigneshwaran, Mrs. S. Kaveya, "Design and Implementation of Forest Monitoring System using IOT", International Journal of Advanced Research in Computer and Communication Engineering, vol. 8, issue 5, pp. 136-140, May 2019

[7] Pushpalatha.R, Darshini M.S, "Real Time Forest Anti-Smuggling Monitoring System based on IOT using GSM", International Journal of Advanced Research in Computer and Communication Engineering, vol. 8, issue 2, pp. 42-48, February 2019

[8] Mallikarjun N, Shilpa N, Renuka D, Sindhu T, Sridevi Malipatil, "Design and Development of Wireless Sensor Node for Anti-Poaching of Valuable Trees", International Journal of Engineering Research in Computer Science and Engineering (IJERCSE), vol 5, issue 4, pp. 547-550, April 2018

[9] T. Saikumar, P. Sriramya, "IoT Enabled Forest Fire Detection and Alerting the Authorities", International Journal of Recent Technology and Engineering (IJRTE), vol. 7, issue. 6S4, April 2019

[10] Pareek. S, Shrivastava. S, Jhala. S, Siddiqui. J.A, Patidar. S, "Iot and Image processing based Forest Monitoring and Counteracting system", 2020 4th International Conference on Trends in Electronics and Informatics (ICOEI), 2020

[11] Othman K. A, Isa M. A. H. M, Baharuddin M. A, Ghazali M. A, Khan Z.I, Zakaria N.A, "Forest Monitoring System Implementation using Visual Basic and Android Application", 2018 18th International Symposium on Communications and Information Technologies (ISCIT), 2018

[12] R. Sundararaj, J. John Wilson, D. Vimala, "Stem borers of Indian sandalwood (Santalum album Linn.) in Karnataka, India", Journal of the Indian Academy of Wood Science, vol. 16, pp. 31-35, 2019

BIOGRAPHIES

Akhil Gunda - Pursuing B.E. final year in the ECE branch at Chaitanya Bharathi Institute of Technology. He has a passion for image processing and machine learning. Worked on projects related to communication and IoT. He got placed at Micron Technologies, Hyderabad, India.

Shreya Lokray - Pursuing B.E. final year in the ECE branch at Chaitanya Bharathi Institute of Technology. She has interned with DRDL, Hyderabad under the Missile Ranging and Navigation team and with Air India Eng. Services, Hyderabad. Her interests are pertaining to the field of machine learning.

Vaishnavi Suthram - Pursuing B.E final year at Chaitanya Bharathi Institute of Technology in the ECE department. She has a keen interest in the field of pattern recognition. Worked on projects related to communication, IoT and Databases. She got placed at Deloitte, Hyderabad, India.