FOREST FIRE DETECTION SYSTEM

GROUP NUMBER - 5

Proposed by:
Prianshu Shaw - 21UCS161
Priyanshu Jain - 21UCS162
Samudro Sinha - 21UCS180

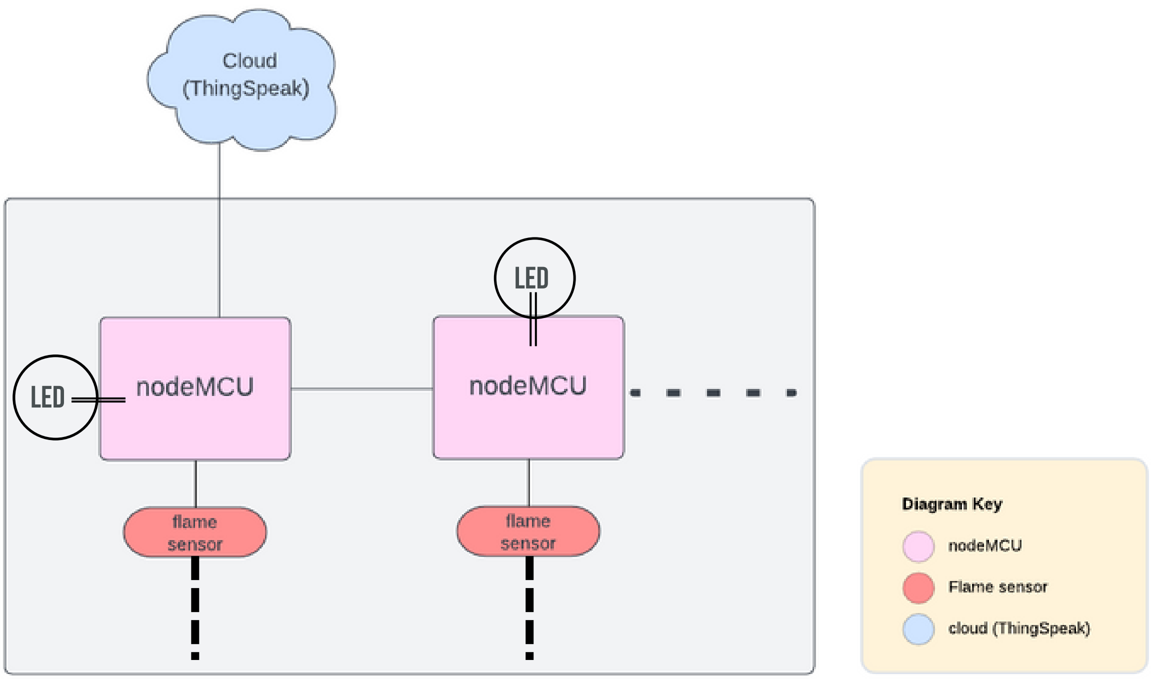
Under the Supervision of:
Dr. Sunil Kumar and Dr. Rajbir Kaur

CONTENTS

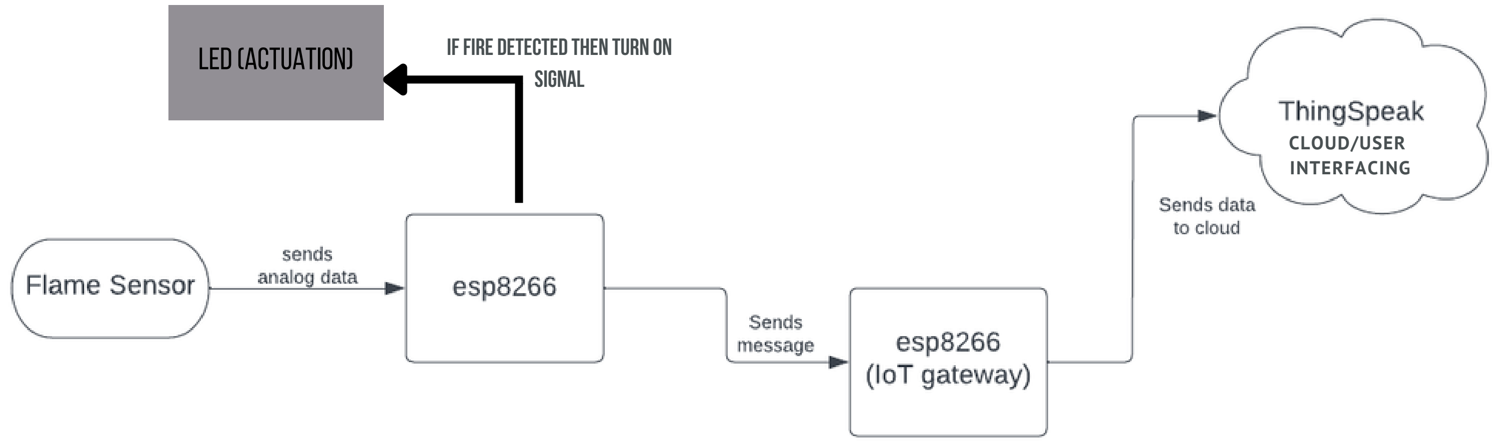
OBJECTIVE

The objective of the Forest Fire detection system project is to develop a smart IoT-based solution for detecting forest fires in their early stages. The system aims to use Flame Sensors to monitor the forest conditions and detect any potential fire hazards.
The collected data from these sensors will be analysed in real-time using algorithms to identify a forest fire. The system will then send alerts to stakeholders such as firefighters and authorities to take appropriate action and prevent the spread of the fire. By providing an early warning system for fires, while also detecting false alarms, the project aims to reduce the risk of extensive damage to forests and wildlife, prevent loss of life and property, and promote sustainable forest management practices.

BLOCK DIAGRAM



DATA FLOW DIAGRAM



COMPONENTS

COMPONENTS USED IN DEMONSTRATION:

- Microcontroller(nodeMCU): nodeMCU acts as our Microcontroller board, it runs on ESP8266 Wi-Fi SoC. It is programmable with Arduino IDE and allows easy connectivity with various sensors and actuators for building smart devices.

- Wifi Module (of nodeMCU):The NodeMCU Wi-Fi module is based on the ESP8266 SoC and provides built-in support for Wi-Fi connectivity. It can be easily programmed to connect to a Wi-Fi network, making it ideal for building IoT devices that require wireless communication.

- Flame Sensor: The flame sensor detects the presence of fire and sends a signal to the microcontroller board. It works by detecting infrared light emitted by the flame.

- Actuator(LED): LED acts as the primary actuator which denotes detection of fire.

- User Interface: The user interface provides a means for users to interact with the system and receive alerts and notifications through the publisher-subscriber model. enables users to monitor the forest conditions in real-time and take appropriate action in case of a potential fire hazard.

COMPONENTS USED FOR SCALABILITY:

- Arduino ATMEGA 2560: The Arduino ATmega acts as the main microcontroller board. It processes sensor data and controlling the various components such Zigbee Module, flame sensor and LED Light. It is used in the Master Device and the sensor nodes. The master acts as the Subscriber and the Sensor nodes acts as the Publishers.

- Flame Sensor: The flame sensor detects the presence of fire and sends a signal to the microcontroller board. It works by detecting infrared light emitted by the flame.

- Zigbee Module: The Zigbee module is a wireless communication module. It allows for the transmission of data between the sensor nodes and the master microcontroller board, enabling real-time monitoring of the forest conditions and detection of potential fire hazards. Its low power consumption and long-range capabilities make it an ideal choice for IoT-based solutions.

- Arduino IDE: It is a cross-compiler based Integrated Development Environment used to program the Microcontrollers used in the project.

- XCTU Platform: XCTU is a free multi-platform application designed to enable developers to interact with Digi RF modules through a simple-to-use graphical interface. Here we are using it to configure our Zigbee modules.

- User Interface: The user interface provides a means for users to interact with the system and receive alerts and notifications through the publisher-subscriber model. It enables users to monitor the forest conditions in real-time and take appropriate action in case of a potential fire hazard.

IMPLEMENTATION

- We developed a forest fire detection system using two **nodeMCU** boards and connected it to the cloud platform **Thingspeak**. The hardware setup consisted of flame sensors, breadboards, jumping wires, LED, and two nodeMCU boards.

- The flame sensor was connected to one nodeMCU board, which was programmed to detect flame.

- When the flame sensor detects the flame, the nodeMCU board sends data to the **IoT gateway**, and then to ThingSpeak. The data includes the sensor readings, as well as the location of the sensor.
- When the nodeMCU receives data from the flame sensor, it lights up the **LED (actuator)** as well as the nodeMCU gateway in it's vicinity lights up it's own LED(actuator), thus indicating **Node-to-Node Communication**.

- We used the Arduino IDE to program the nodeMCU boards. The code for each board was uploaded to the board using a USB cable. We tested the system by lighting a small fire near the sensors and confirmed that data was sent to Thingspeak.

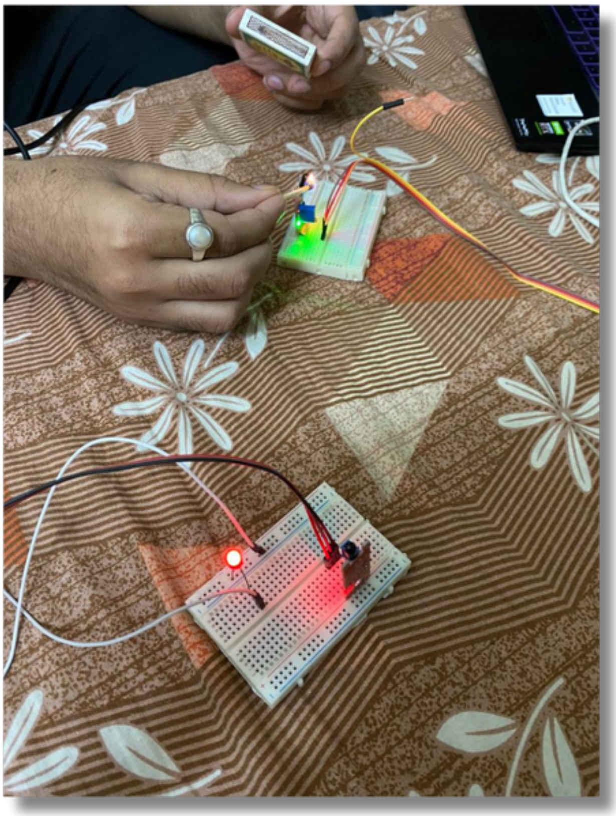
- The sensors were placed in strategic locations in areas with a high risk of forest fires.

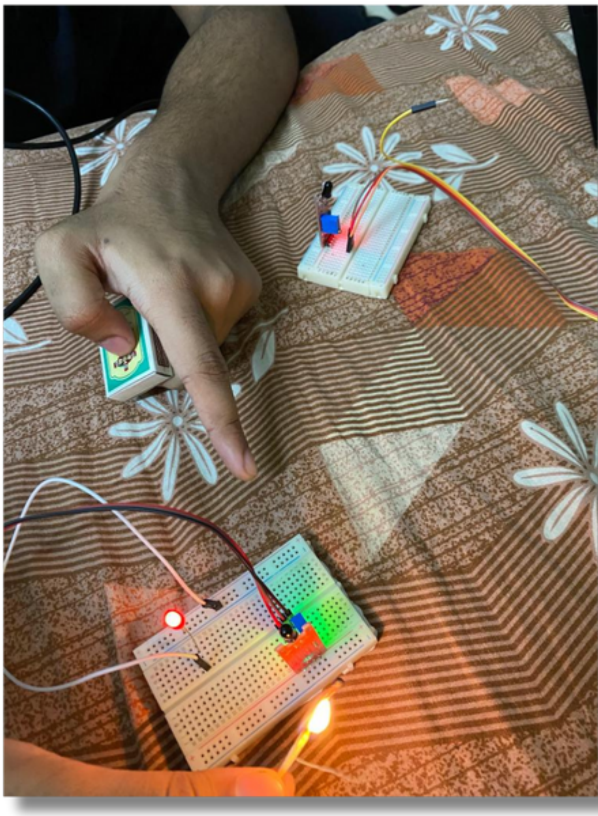
- To analyze the data, we used the Thingspeak platform to create graphs and charts that displayed the sensor readings over time. We also used **MATLAB** to analyze the data in more detail and identify any patterns or anomalies.

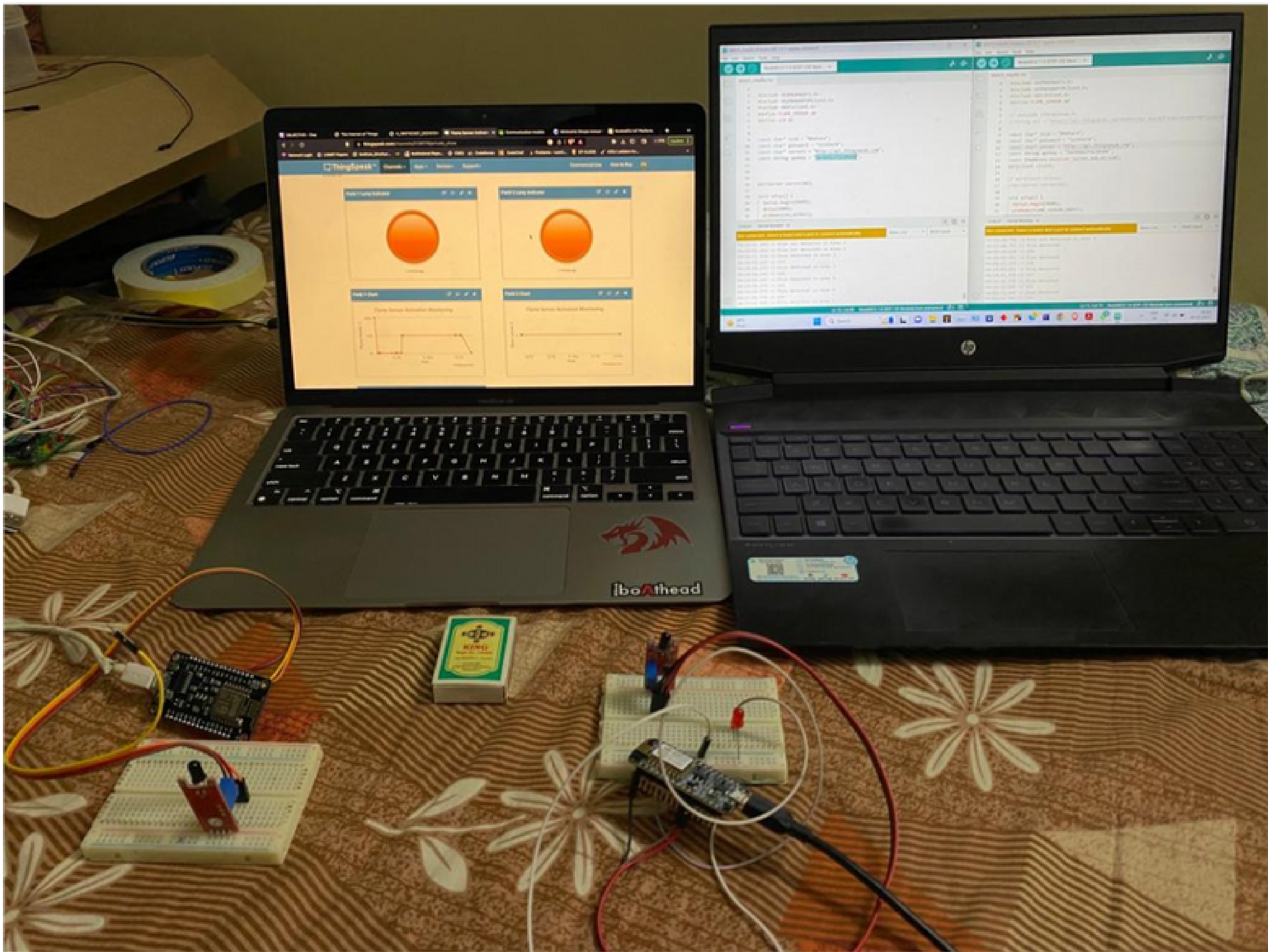
CONNECTIONS

- The flame sensor is connected with nodeMCU via breadboard.
  The GND of the flame sensor is connected to the GND pin of nodeMCU
  The VCC of the flame sensor is connected to the VCC pin of nodeMCU
  The analog output of the flame sensor is connected to the analog pin of nodeMCU

- LED is connected to receiver node via breadboad which acts as an Actuator.







WHOLE PROJECT SETUP

PROTOCOLS

We can connect two nodeMCU using espNOW, we have implemented a similar thing here.
Here we have implemented star topology and we have connected both the node MCU's to each other with the help of central wireless access point.
Serial communication is a common protocol for exchanging data between microcontrollers, using a two-wire interface with a baud rate that determines the speed of data transfer.
The **publisher-subscriber** (pub/sub) protocol is a messaging pattern used in distributed systems and IoT applications. It involves two types of entities: publishers, which produce and send messages, and subscribers, which receive and process messages. Instead of sending messages directly to specific subscribers, publishers send messages to a central broker, which then distributes the messages to all subscribers that have subscribed to a particular topic. This decouples the producers and consumers, allowing for a more scalable and flexible system. MQTT is an example of a pub/sub protocol commonly used in IoT applications.

RESULTS

The forest fire detection system that we developed using two nodeMCU boards and connected to the cloud platform Thingspeak was successful in detecting smoke and heat and sending data to the cloud. The system provided reliable data for monitoring forest fires, and the data was displayed in a user-friendly way using graphs and charts.

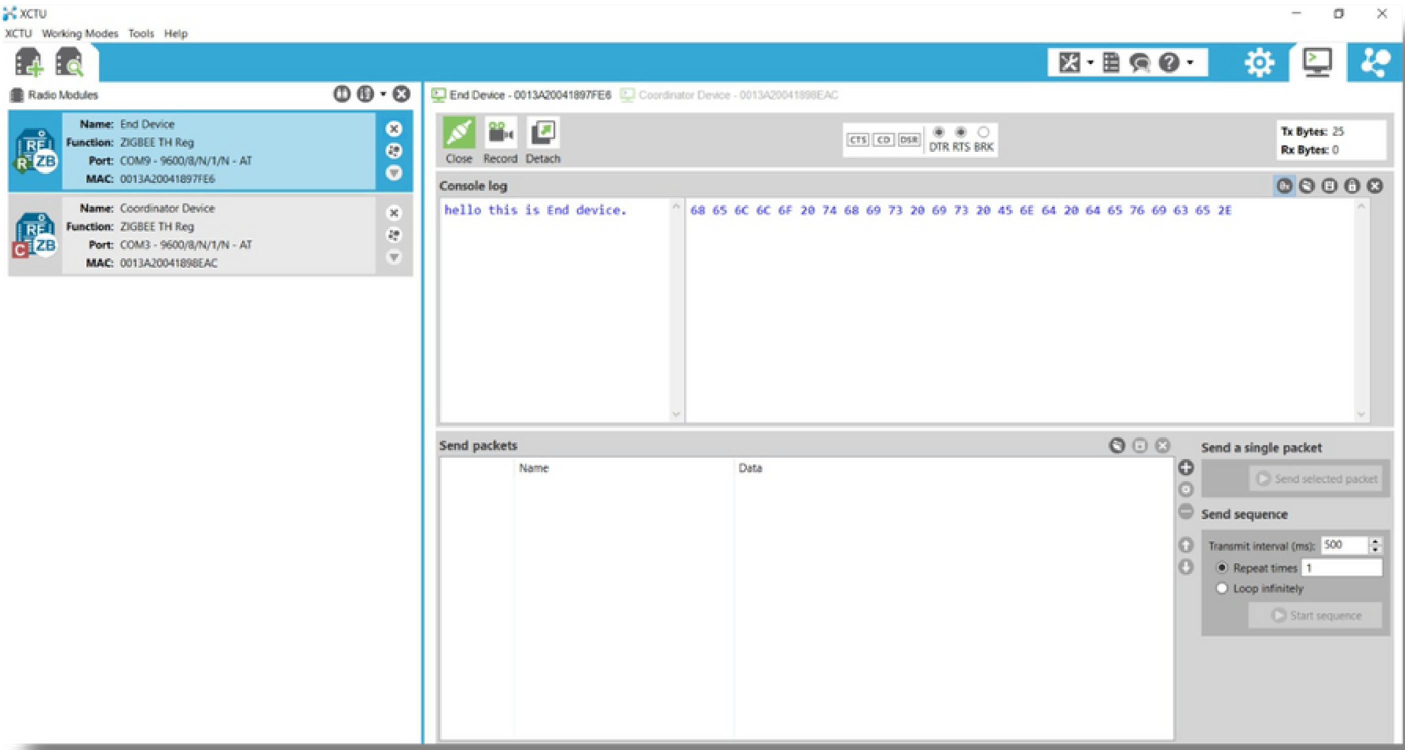
We successfully implemented the system using the HTTP protocol to send data to Thingspeak, and we were able to track the location of the sensors using GPS modules. In addition, we used MATLAB to analyze the data and identify any patterns or anomalies.

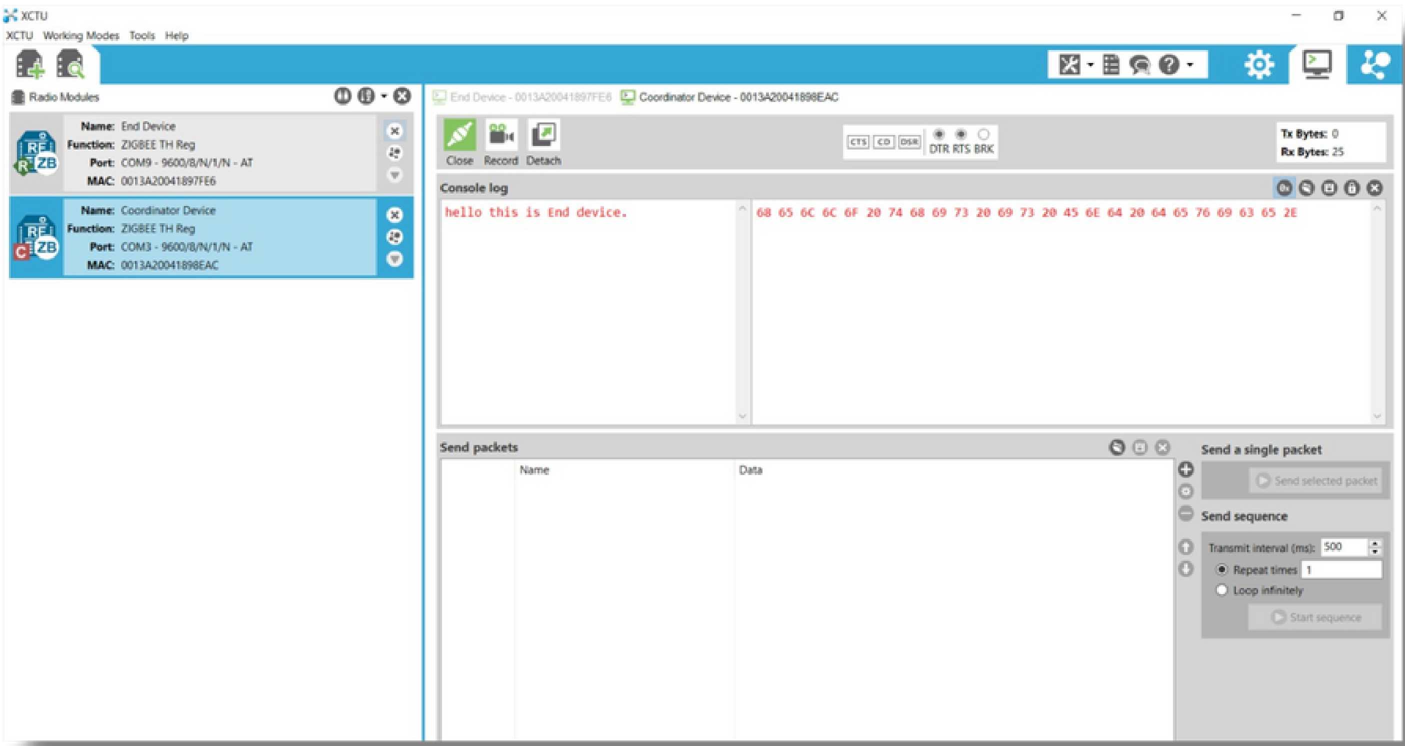
While the system was able to detect smoke and heat, we recognize that it is not a complete solution for forest fire prevention and management. In the future, we plan to add more sensors to the system and explore using machine learning algorithms to analyze the data in real-time and provide more accurate predictions of fire risk.

Overall, our project demonstrates the potential of using IoT technologies to monitor and manage environmental risks, such as forest fires. With further development and refinement, we believe that the system has the potential to make a significant contribution to efforts to prevent and manage forest fires.

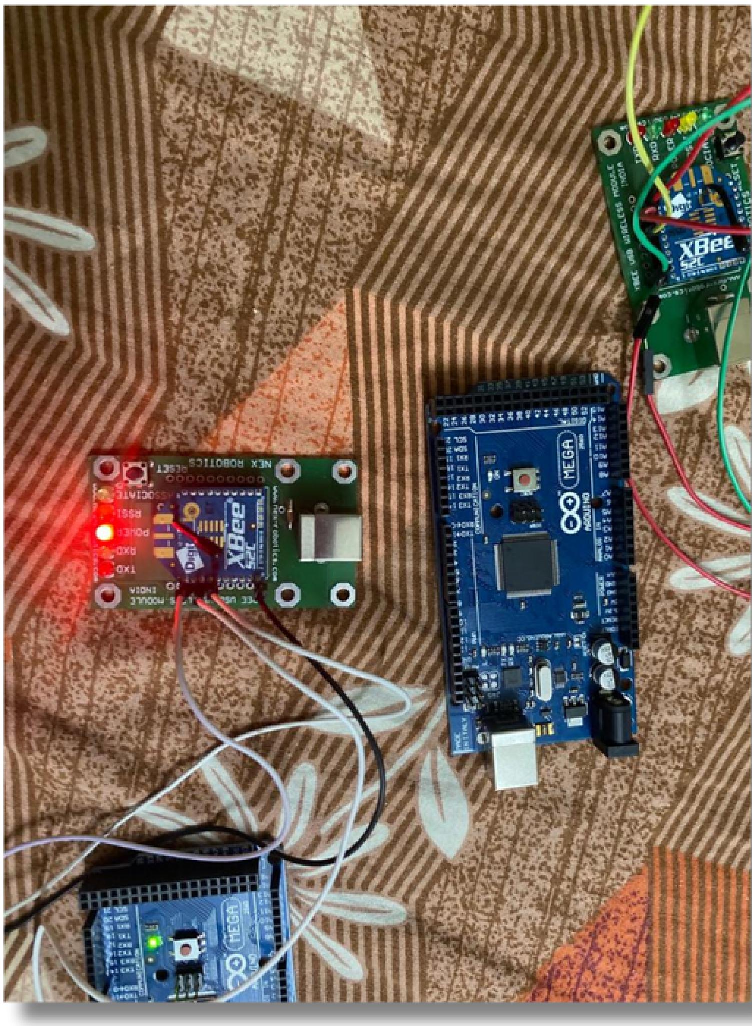
SCALABILITY

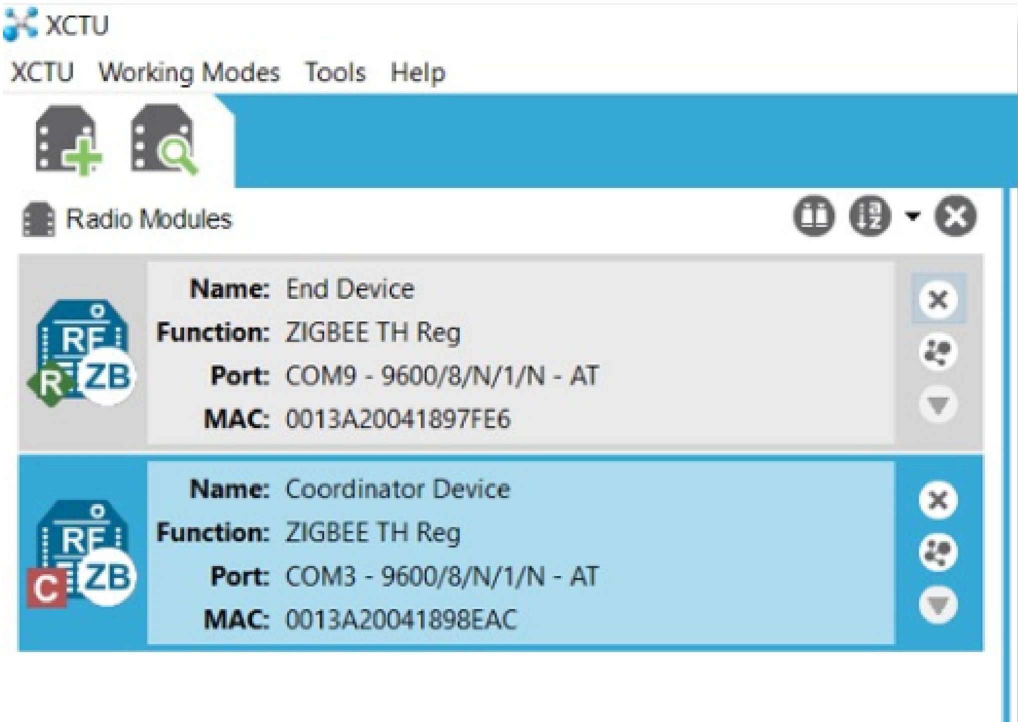
1. Add more sensors: One way to scale up the project is to add more sensors to the system. For example, you could add humidity sensors to monitor the moisture content of the air, or wind sensors to detect changes in wind speed and direction. By adding more sensors, you can get a more complete picture of the environmental conditions that contribute to forest fires.
2. Use machine learning algorithms: Another way to scale up the project is to use machine learning algorithms to analyze the data in real-time and provide more accurate predictions of fire risk. For example, you could use supervised learning algorithms to train the system to recognize patterns in the data that are associated with increased fire risk.
3. Deploy the system in more locations: To scale up the project, you could deploy the system in more locations. For example, you could work with local authorities to deploy the system in areas with a high risk of forest fires, or in areas where fires are more likely to occur due to human activity.
4. Use alternative power sources: To make the system more portable and easier to deploy in remote locations, you could explore using alternative power sources, such as solar panels or rechargeable batteries.
5. Integrate the system with other technologies: To make the system more effective, you could integrate it with other technologies, such as drones or satellites. For example, you could use drones to collect data from the sensors in real-time and transmit the data to a central location for analysis.





IMPLEMENTATION OF ZIGBEE





IMPLEMENTATION OF ZIGBEE

FUTURE SCOPE

1. Real-time alerts and notifications: One possible future scope for the project is to incorporate a real-time alert system that sends notifications to relevant authorities or emergency services when a fire is detected. This could include SMS alerts or push notifications to mobile devices, or even automated phone calls to designated personnel.
2. Smart firefighting: Another possible future scope is to integrate the system with smart firefighting technologies, such as drones or autonomous vehicles. This could allow for more targeted and effective firefighting efforts, by providing real-time data on fire spread and direction.
3. Public awareness campaigns: To increase the impact of the system, you could also launch public awareness campaigns to educate people about the risks of forest fires and how the system works. This could involve creating social media campaigns, distributing informational flyers or organizing workshops or seminars.
4. Collaborative data sharing: To maximize the impact of the project, you could also explore opportunities for collaborative data sharing with other organizations or governments. This could involve sharing data with research institutions or government agencies to support their efforts to prevent and manage forest fires.
5. Commercial applications: The forest fire detection system could also have commercial applications, such as in the forestry industry or for insurance companies. For example, the system could be used to monitor the risk of forest fires in areas where timber is harvested, or to provide data to insurance companies to help them assess the risk of insuring properties in high-risk fire areas.

CONCLUSION

Certainly, statistics demonstrate the critical need for effective forest fire detection systems. According to the National Interagency Fire Center, in 2020, there were over 58,000 wildfires in the United States that burned approximately 10.1 million acres of land. These fires caused significant property damage, ecological destruction, and loss of life. Similarly, in India, the Forest Survey of India reported over 54,000 forest fires in 2019 alone, causing an estimated loss of Rs 1,090 crore (approximately USD 146 million) in the last five years.

However, our forest fire detection system offers a cost-effective and life-saving solution to this critical issue. By using flame sensors, the system can detect fires in their early stages, allowing for swift and effective action to be taken. The system's affordability and ease of deployment make it accessible to rural communities that may be at higher risk of forest fires but lack resources for advanced detection systems.

With the ability to detect fires in their early stages, our system can help prevent damage to property, reduce ecological impact, and most importantly, save lives. By providing an affordable, effective, and accessible solution to wildfire detection, our forest fire detection system has the potential to significantly reduce the economic and ecological losses caused by forest fires globally.

In conclusion, our forest fire detection system represents an innovative and life-saving development in the field of wildfire prevention and mitigation. The statistics on forest fires demonstrate the urgent need for effective detection systems, and our system offers an affordable, accessible, and effective solution to this pressing issue.