ANTI-SMUGGLING OF TREES BY USING IOT

P. RAJA PRAKASH RAO [1], A. SREEKANTH [2], K. SRIKALA [2], R. MADHAVI [2], D. AJAY BABU [2], A. CHANDAN [2]

[1] *Professor, ECE department, PBR Visvodaya Institute Of Technology & Science, Kavali, Andhra Pradesh, India.*
[2] *UG students, ECE department, PBR Visvodaya Institute Of Technology & Science, Kavali, Andhra Pradesh, India.*

ABSTRACT:

Sandalwood, SAGWAN trees are smuggling by the thieves in the present days. These trees are rarely available and more cost also. There are many uses of sandalwood trees such as we can utilize in skincare products and AYURVEDIC medicine. The primary goal of this paper is to design a framework or system which will be utilized to restrict the sneaking of sandalwood trees. A system can be developed using a tilt sensor(to detect the inclination of the tree which are being cut), smoke sensor (m2q), NODEMCU device (Wi-Fi device) and GSM Module, to restrict smuggling and to track the trees much easy. These sensors data will consistently check with the BLYNK app. The obtained data is stored in BLYNK app with using of the NODEMCU module. Forestofficials get alert when any event occurs so that appropriate operation will be taken.

1.INTRODUCTION:

In recent years poaching or smuggling of environmentally and economically important species of trees in forested areas- such as Sandalwood, Teakwood, Pine and Rosewood has been tremendously increased. There have been several initiatives undertaken by different stakeholders– and in particular by the Govt. of India, to mitigate these problems. These include the recruitment, training and deployment of anti-poaching watchers and/or private/govt. security guards across forests. Strict punishments for convicted offenders, as well as giving special incentives for antipoaching activities (Twelfth Five Year Plan 2012-2017) were aimed for eradicating the menace.

The main idea presented in this paper is to design a portable wireless sensor node which is apart of wireless sensor Network. It will be mounted

on trunk of each tree, capable of detecting theft as well as

automatically initiate send alarm signals if any to remote terminal through wireless media.

An Embedded system is a special-purpose system in which the computer is completely encapsulated by or dedicated to the device or system it controls. Unlike a general-purpose computer, such as a personal computer, an embedded system performs one or a few predefined tasks, usually with very specific requirements. Since the system is dedicated to specific tasks, design engineers can optimize it, reducing the size and cost of the product.

Personal digital assistants (PDAs) or handheld computers are generally considered embeddeddevices because of the nature of their hardware design, even though they are more expandable in software terms. With the introduction of the OQO Model 2

with the Windows XP operating system and ports such as a USB port both features usually belong to "general purpose computers".

The existing system consists of hiring security personals for monitoring the entire area for suspicious activity, however due to physical limitations in human it is hard to monitor the entire area continuously, thus hiring of guards proves unreliable and inadequate. Another existing system is the installation of CCTV cameras for covering large area proves very costly and is hard to implement. Also, the latest trend for protection of trees is to tag an RF-ID to trees just like tagging an animal for knowing the whereabouts of a particular tree. However, this technology does not give the real time information while the activity is happening. Activity is detected only when the tree leaves its initial position

Fig .1. Examples of Existing System2.PROPOSED SYSTEM

The main idea presented in this paper is to design a portable wireless sensor node which will be a part of a Wireless Sensor Network. The suggested system will consist of two modules one involving sensors and controller module which will be at tree spot and another one is Android phone. We built an application which will continuously receive sensor data. This is an IOT based project where we upload sensor data continuously to cloud. Tilt sensor is used to determine whether the tree is cut down or not similarly temperature sensor is used to determine whether the forest is on fire or not. Using built application we can turn on the water pump in case of forest fire and we can turn on alarm other devices in smuggling case. This system is like from anywhere we can monitor and control by using GSM Technology. Which will convey the message from monitor station to the control station vice versa using wireless technology.

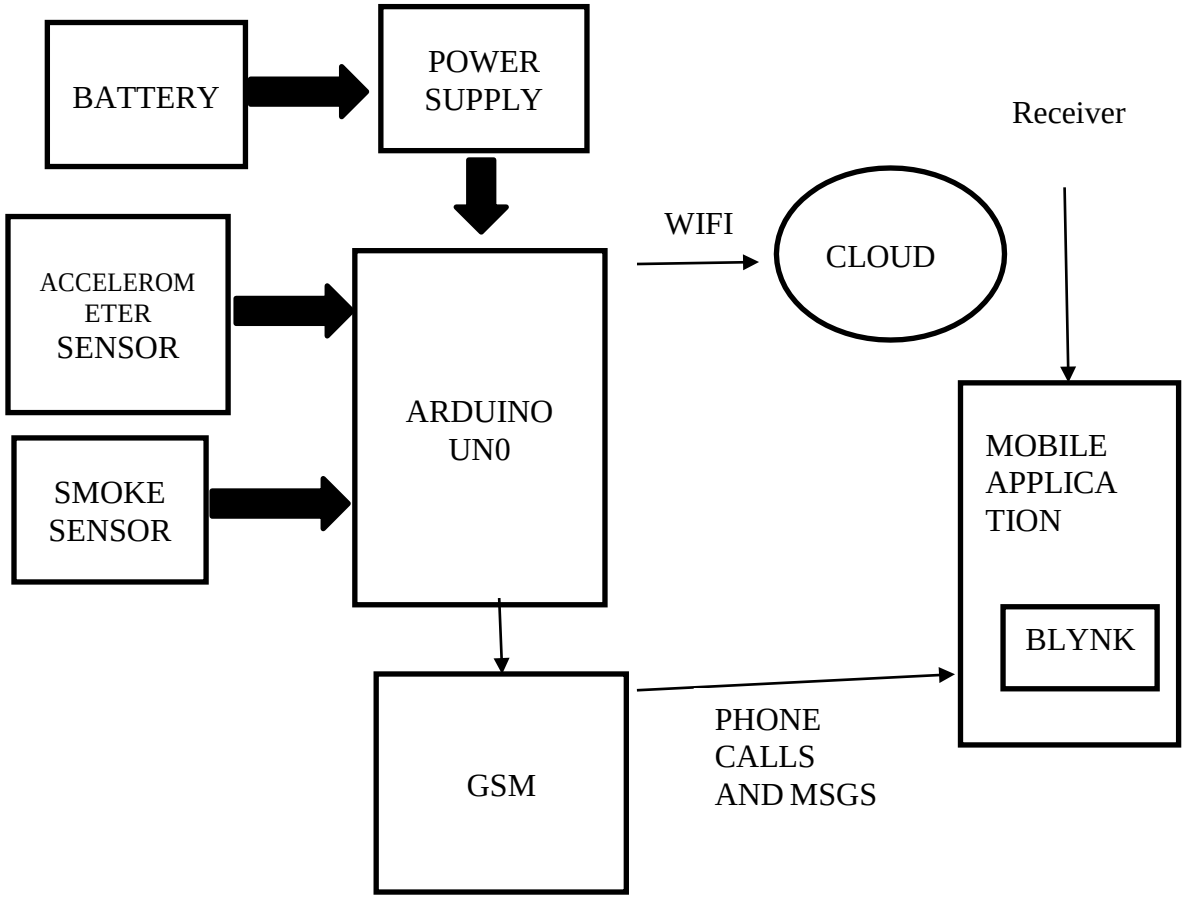


Fig.2. Block diagram of Proposed System

3.COMPONENTS AND DESCRIPTION:

3.1.ARDUNIO UNO

The Arduino Uno is a microcontroller board based on the ATmega328 (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to-serial converter. "Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions, see the index of Arduino boards.

The **ATmega328** is one kind of single-chip microcontroller formed with Atmel within the **megaAVR family**. The architecture of this Arduino Uno is a customized Harvard architecture with 8 bit **RISC processor** core. **Other boards of Arduino Uno** include Arduino Pro Mini, Arduino Nano, Arduino Due, Arduino Mega, and Arduino Leonardo.

Fig.3. Arduino Uno

FEATURES OF ARDUINO UNO BOARD:

The features of Arduino Uno ATmega328 includes the following:

- The operating voltage is 5V

- The recommended input voltage will range from 7v to 12V
- The input voltage ranges from 6v to 20V
- Digital input/output pins are 14
- Analog i/p pins are 6
- DC Current for each input/output pin is 40 mA
- DC Current for 3.3V Pin is 50 mA
- Flash Memory is 32 KB
- SRAM is 2 KB
- EEPROM is 1 KB
- CLK Speed is 16 MHz

The 14 digital input/output pins can be used as input or output pins by using pin Mode(), digital Read() and digital Write() functions in Arduino programming. Each pin operate at 5V and can provide or receive a maximum of 40mA current, and has an internal pull-up resistor of 20-50 Kilo Ohms which are disconnected by default. Out of these 14 pins, some pins have specific functions as listed below:

- **Serial Pins 0 (Rx) and 1 (TX):** RX and TX pins are used to receive and transmit TTL serial data. They are connected with the corresponding ATmega328P USB to TTL serial chip.
- **External Interrupt Pins 2 and 3:** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value.
- **PWM Pins 3, 5, 6, 9 and 11:** These pins provide an 8-bit PWM output by using analog Write() function.
- **SPI Pins 10 (SS), 11 (MOSI), 12 (MISO) and 13 (SCK):** These pins are used for SPI communication.
- **In-built LED Pin 13:** This pin is connected with an built-in LED, when pin 13 is HIGH – LED is on and when pin 13 is LOW, its off.
- Along with 14 Digital pins, there are 6 analog input pins, each of which provide 10 bits of resolution, i.e. 1024 different values. They measure from 0 to 5 volts but this limit can be increased by using AREF pin with analog Reference() function.
- Analog pin 4 (SDA) and pin 5 (SCA) also used for TWI communication using Wire library.
- Arduino Uno has a couple of other pins as explained below:
- **AREF:** Used to provide reference voltage for analog inputs with analog Reference() function.

- **Reset Pin:** Making this pin LOW, resets the microcontroller.

3.2. ARDUINO UNO TO ATMEGA328 MAPPING:

When ATmega328 chip is used in place of Arduino Uno, or vice versa, the image below shows the pin mapping between the two.

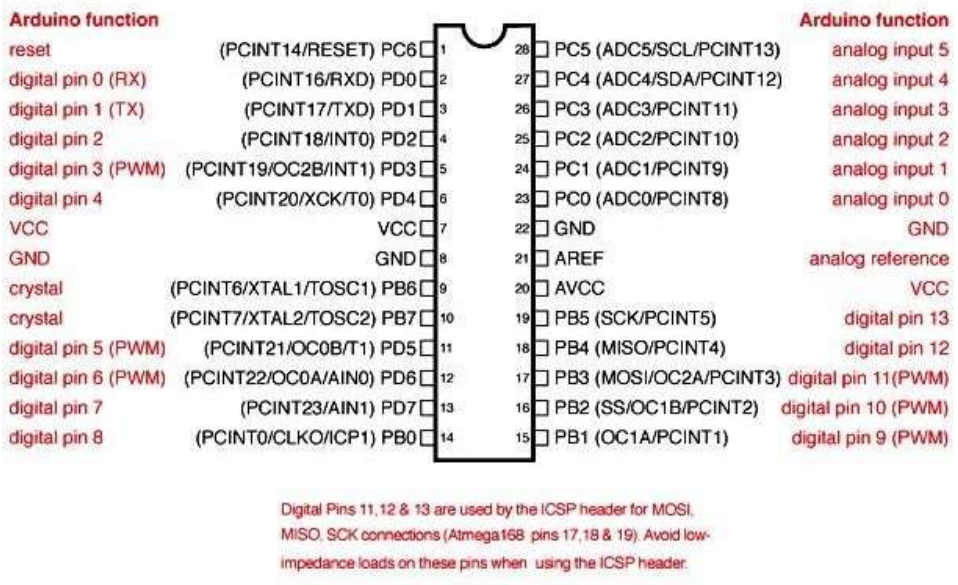


Fig. 4. AT mega 328 pin diagram

APPLICATIONS:

- Prototyping of Electronics Products and Systems
- Multiple DIY Projects.
- Easy to use for beginner level Dyers and makers
- Projects requiring Multiple I/O interfaces and communications

3.3. NODEMCU:

The ESP8266 is a System on a Chip (SOC), manufactured by the Chinese company Espressif. It consists of a Tensilica L106 32-bit micro controllerunit (MCU) and a Wi-Fi transceiver. It has 11 GPIO pins* (General Purpose Input/output pins), and an analog input as well. This means that you can program it like any normal Arduino or other microcontroller. And on top of that, you get Wi-Fi communication, so you can use it to connect to your Wi-Fi network, connect to the Internet, host a web server with real web pages, let your smartphone connect to it,

etc. The possibilities are endless! It's no wonder that this chip has become the most popular IOT device available.

There are many different modules available, standalone modules like the ESP## series by AI Thinker, or complete development boards like the NODEMCU DEVKIT or the WEMOS D1. Different boards may have different pins broken out, have different Wi-Fi antennas, or a different amount of flash memory on board.

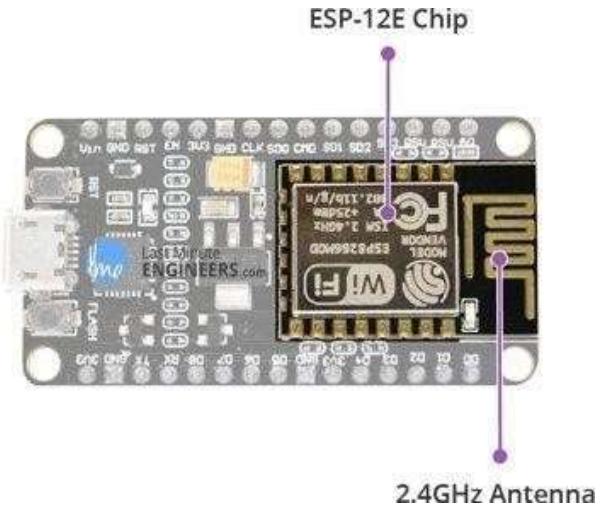


Fig .5. NODEMCU

3.4. ESP-12E MODULE:

The development board equips the ESP-12E module containing ESP8266 chip having **Tensilica Tensas® 32-bit LX106 RISC microprocessor** which operates at **80 to 160 MHz** adjustable clock frequency and supports **RTOS.**

There's also **128 KB RAM and 4MB of Flash memory** (for program and data storage) just enough to cope with the large strings that make up web pages, JSON/XML data, and everything we throw at IOT devices nowadays.

The ESP8266 Integrates **802.11b/g/n HT40 Wi-Fi transceiver**, so it can not only connect to a Wi-Fi network and interact with the Internet, but it can also set up a network of its own, allowing other devices to connect directly to it. This makes the ESP8266 NODEMCU even more versatile.

POWER REQUIREMENT:

As the operating voltage range of ESP8266 is **3V to 3.6V**, the board comes with a LDO voltage regulator to keep the voltage steady at 3.3V. It can reliably supply up to 600mA, which should be more than enough when ESP8266 pulls as much as **80mA during RF transmissions**. The

output of the regulator is also broken out to one of the sides of the board and labeled as 3V3. This pin can be used to supply power to external components.

- Operating Voltage: 2.5V to 3.6V
- On-board 3.3V 600mA regulator
- 80mA Operating Current
- 20 µA during Sleep Mode

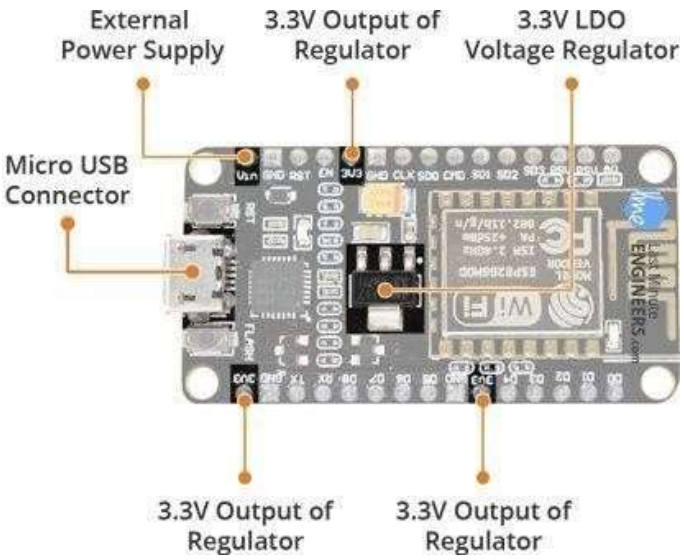


Fig .6. Power Requirements of NODEMCU

Power to the ESP8266 NODEMCU is supplied via the on-board MicroUSB connector. Alternatively, if you have a regulated 5V voltage source, the Vin pin can be used to directly supply the ESP8266 and its peripherals.

GSM MODULE

Whether you want to listen to what happens in your house that's miles away from you or activate sprinkler system in your garden just with a silent call; Then SIM800L GSM/GPRS module serves as a solid launching point for you to get you started with IOT!

SIM800L GSM/GPRS module is a miniature GSM modem, which can be integrated into a great number of IOT projects. You can use this module to accomplish almost anything a normal cell phone can; SMS text messages, Make or receive phone calls, connecting to internet through GPRS, TCP/IP, and more! To top it off, the module supports quad-band GSM/GPRS network, meaning it works pretty much anywhere in the world.

HARDWARE OVERVIEW OF SIM800L:

At the heart of the module is a SIM800L GSM cellular chip from SimCom. The operating voltage of the chip is from 3.4V to 4.4V, which makes it an ideal candidate for direct LiPo battery supply. This makes it a good choice for embedding into projects without a lot of space.

Fig.6. Hardware overview of SIM800L

All the necessary data pins of SIM800L GSM chip are broken out to a 0.1″ pitch headers. This includes pins required for communication with a microcontroller over UART. The module supports baud rate from 1200bps to 115200bps with AutoBaud detection.

The module needs an external antenna to connect to a network. The module usually comes with a Helical Antenna and solders directly to NET pin on PCB. The board also has a U.FL connector facility in case you want to keep the antenna away from the board.

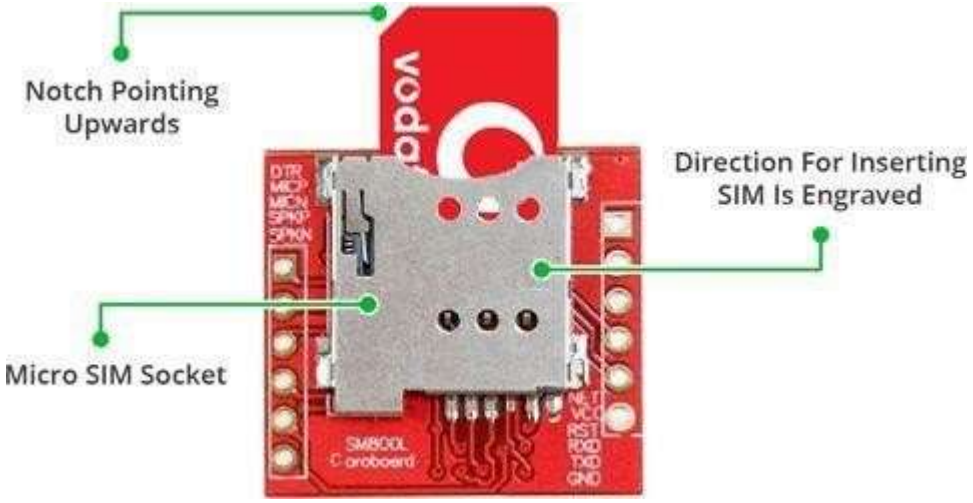


Fig.7. SIM socket on the back

There's a SIM socket on the back! Any activated, 2G micro SIM card would work perfectly. Correct direction for inserting SIM card is normally engraved on the surface of the

SIM socket.

This module measures only 1 inch but packs a surprising amount of features into its little frame. Some of them are listed below:

- ✓ Supports Quad-band: GSM850, EGSM900, DCS1800 and PCS1900
- ✓ Connect onto any global GSM network with any 2G SIM
- ✓ Make and receive voice calls using an external 8Ω speaker & electret microphone
- ✓ Send and receive SMS messages
- ✓ Send and receive GPRS data (TCP/IP, HTTP, etc.)
- ✓ Scan and receive FM radio broadcasts
- ✓ Transmit Power:
- ✓ Class 4 (2W) for GSM850
- ✓ Class 1 (1W) for DCS1800
- ✓ Serial-based AT Command Set
- ✓ FL connectors for cell antennae
- ✓ Accepts Micro SIM Card

3.5. ARDUNIO SOFTWARE DESCRIPTION:

Arduino is an open source hardware and software company project and user community that designs and manufactures single board microcontroller and kits for building digital devices and interactive objects that can sense and control both physically and digitally. Its products are licensed under the GNU Lesser General Public License (LGPL) or the GNU General Public License (GPL), emitting the manufacture of Arduino boards and software distribution by anyone. Arduino boards are available commercially in preassembled form or as do-it-yourself (DIY) kits.

Arduino board designs use a variety of microprocessors and controllers. The boards are equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards or breadboards (*shields*) and other circuits. The boards feature serial communications interfaces, including Universal Serial Bus (USB) on some models, which are also used for loading programs from personal computers. The microcontrollers are typically programmed using a dialect of features from the programming languages C and C++. In addition to using traditional compiler toolchains, the Arduino project provides an integrated development environment (IDE) based on the Processing language project.

A program for Arduino hardware may be written in any programming language with compilers that produce binary machine code for the target processor. Atmel provides a development environment for their 8-bit AVR and 32-bit ARM Cortex-M based

microcontrollers: AVR Studio (older) and Atmel Studio (newer).

The Arduino integrated development environment (IDE) is a crossplatform application (for Windows, macOS, Linux) that is written in the programming language Java. It originated from the IDE for the languages *Processing* and *Wiring*. It includes a code editor with features such as text cutting and pasting, searching and replacing text, automatic indenting, brace matching, and syntax highlighting, and provides simple *one-click* mechanisms to compile and upload programs to an Arduino board. It also contains a message area, a text console, a toolbar with buttons for common functions and a hierarchy of operation menus. The source code for the IDE is released under the GNU General Public License, version.

The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub *main()* into an executable cyclic executive program with the GNU tool chain, also included with the IDE distribution. The Arduino IDE employs the program *argued* to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware.

3.6. CONFIGURING SIMULATOR:

Simulator for Arduino is the most full featured Arduino Simulator available at the present time (watch the latest video below).

The benefits and features of the Arduino Simulator are:

- ✓ The ability to teach and demonstrate the inner workings of an Arduino sketch
- ✓ Test out a sketch without the hardware, or prior to purchasing hardware
- ✓ Debug a sketch
- ✓ Demonstrate a project to a potential customer
- ✓ Develop a complicated sketch faster than using the hardware

Download the free version below with a short delay timer on loading a sketch, and when ready upgrade to the Pro Version. *Simulator for Arduino* Pro Version is currently used in many countries over six continents. The download consists of a zip file containing a setup.exe file which installs an exe file, help files, images and examples. It is designed for the Arduino Uno, Mega and most other common Arduino boards and does the following:

- ✓ Steps through the program line by line. If a new line is selected, the program will continue from that point.

- ✓ Performs digitalWrite, digitalRead and PinMode for pins 0-53
- ✓ AnalogRead for pins 0-16 and analogWrite for digital pins 0-53
- ✓ Emulates Serial, LCD output, Ethernet, Servo, SD card, EEPROM, SoftSerial,
- ✓ SPI, Wire
- ✓ If, while, for, switch, dowhile loop functionality
- ✓ Subroutines (multi-level) with arguments
- ✓ View variables in real-time
- ✓ Step Into, Step Over, Step Out of or Run mode
- ✓ Ability to edit sketch or open in Arduino IDE
- ✓ Tabs for separate files in the sketch
- ✓ Context-sensitive help
- ✓ 2 and 4 line LCD support only with improvised CGRAM
- ✓ 2 dimensional arrays (without initialization)
- ✓ Breakpoint now with a conditional option
- ✓ load custom libraries automatically after setting the Library Directory

- ✓ Change the font, size and style of the Simulator
- ✓ Advanced watch for easy variable viewing
- ✓ Minimize mode for demo/training
- ✓ Limited support for custom libraries
- ✓ Limited support for pointer and structures

4. WORKING PROCEDURE:

- We are going to place the ADXL335 sensor in the top of a valuable trees and note the sensor value and change the code to the sensor value.
- If anyone cuts the tree or tree fall down ,the sensor value get changes ,so Arduino calls GSM ,then GSM make a call to registered number and sends messages also.
- The ADXL335 value is send to Arduino board, Arduino send it to the Nodemcu, Nodemcu to the cloud, and cloud to Blynk .we can monitor the tree condition through Blynk app.
- M2Q sensor is sensing carbon dioxide, so we can the carbon dioxide at forest
- .if the carbon dioxide value .excess then the limit .so it indicate the fire .then Arduino calls GSM , then GSM make a call to registered phone number and sends messages also.
- The M2Q value is send to Arduino board, Arduino send it to the Nodemcu, Nodemcu to the cloud, and cloud to Blynk .we can monitor the tree condition through Blynk app.

4.2.BLOCK DIAGRAM:

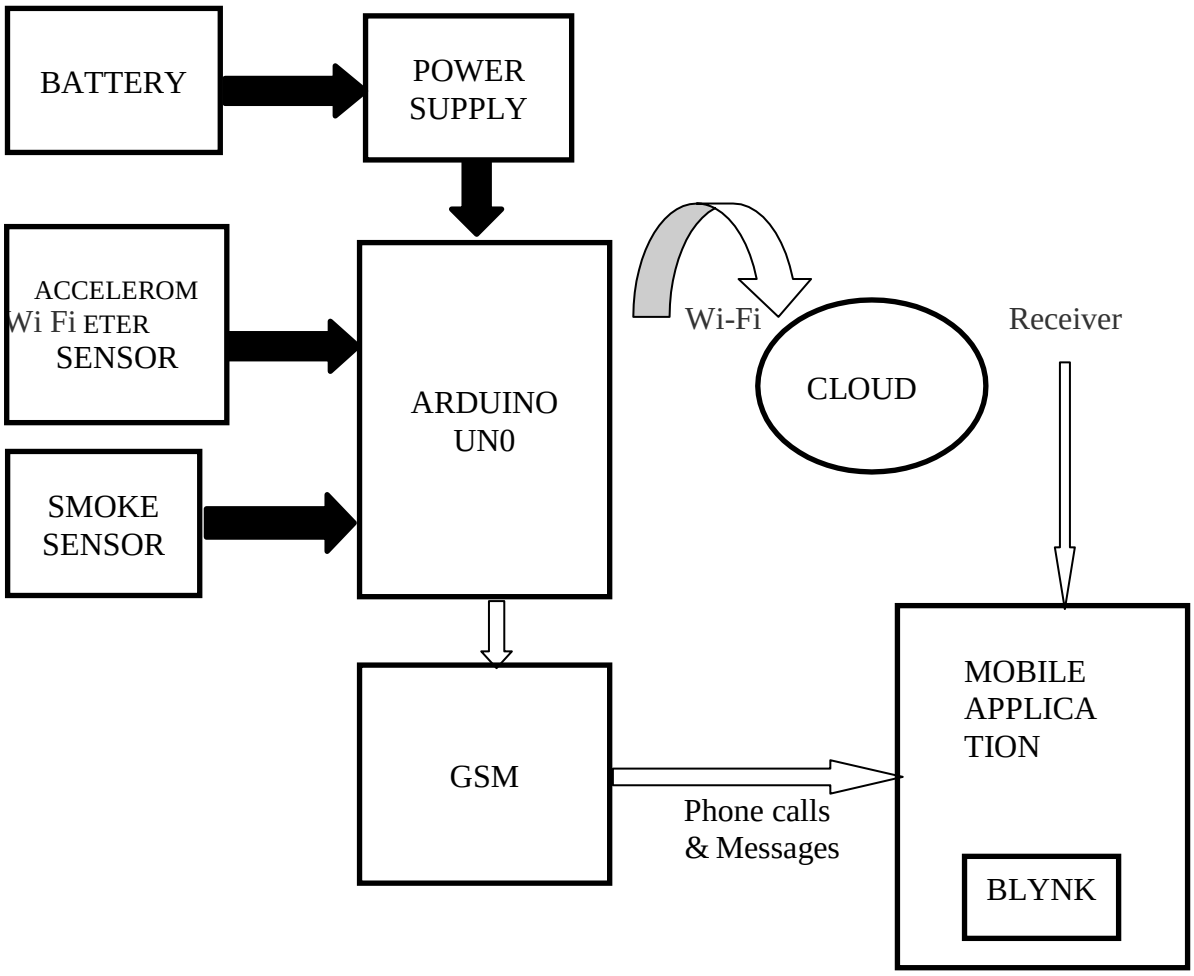


Fig.8. Block Diagram of Anti-Smuggling of Trees By using IOT

4.3.FLOW CHART:

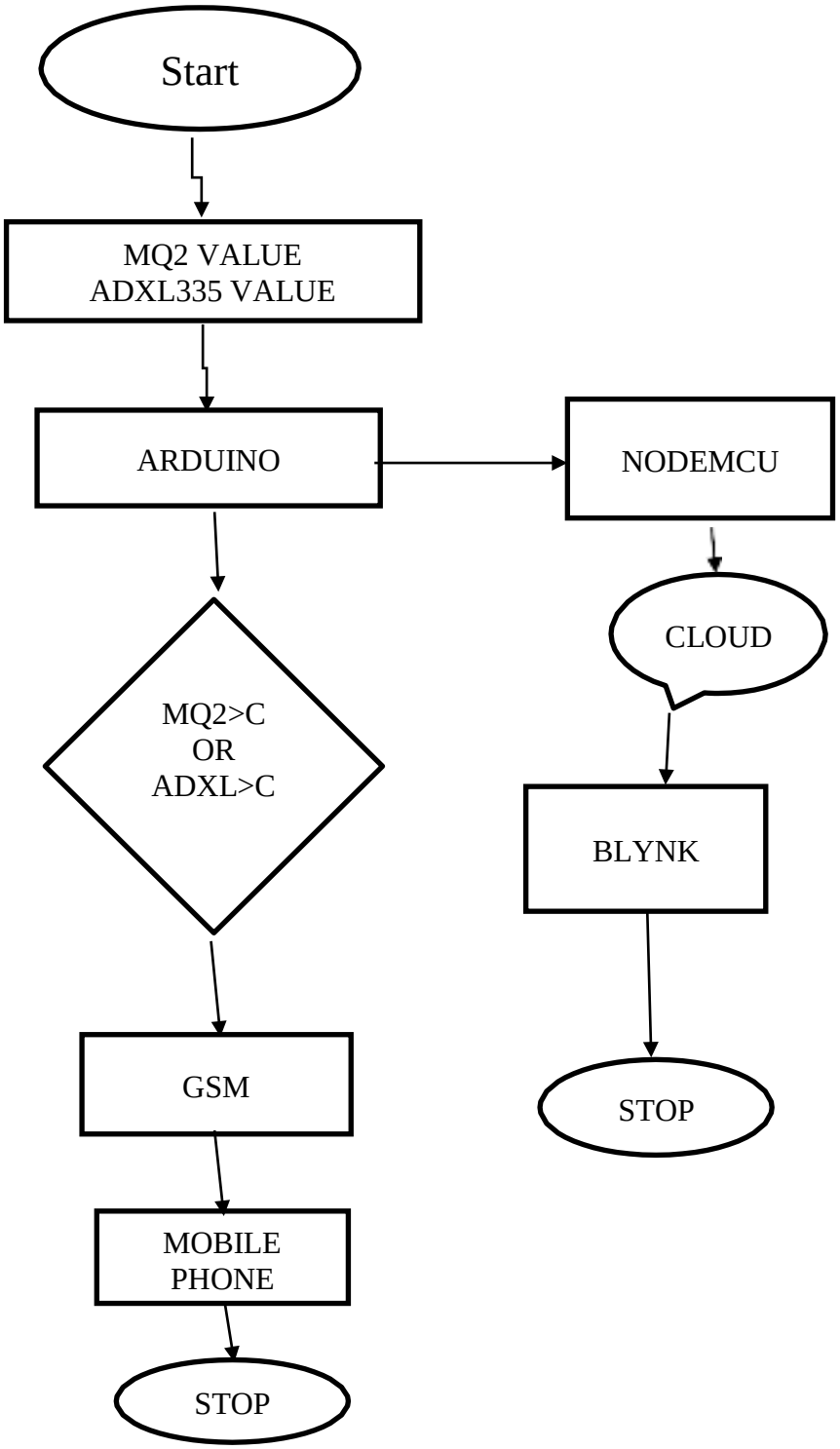
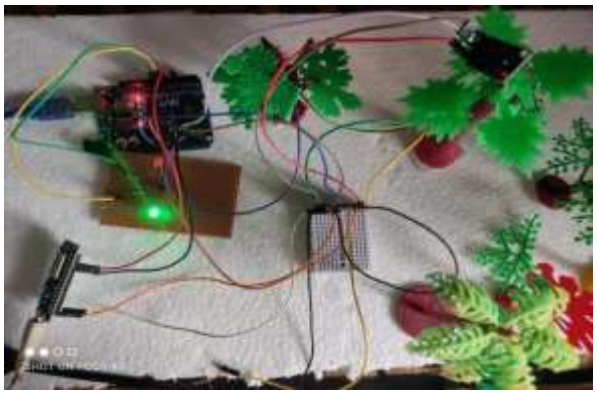


Fig.9. Flowchart of Anti-Smuggling of Trees By using IOT

5.RESULTS:

- ***Case-i:***
  - ✓ If tree is safe and no fire in the forest an near the tree, then the results in the Blynk app is showing like this:



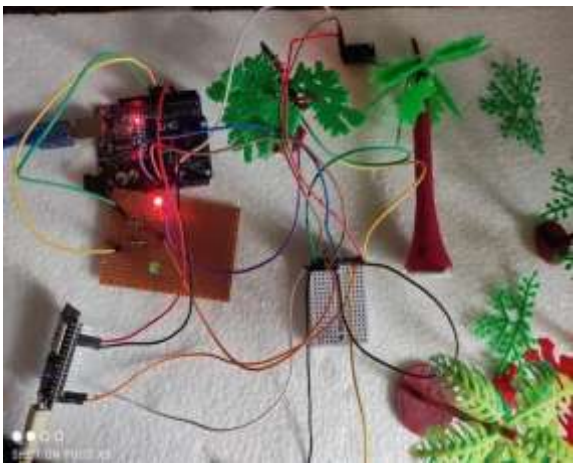
- ***Case-ii:***

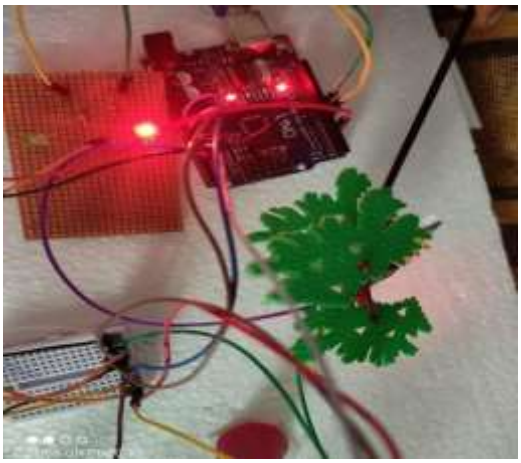
✓ If anyone cuts the tree or tree fell down or any detection of fire immediately we get calls from GSM module. We get phone call as shown below to the registered mobile number.

✓ In Blynk app it is shown in below:

1. Tree fall down

2. Detects fire





➢ ***Case-iii:***

✓ when both conditions happen to gather then the Blynk app is going show like this:

6.CONCLUSION

It is identified from the beginning that producing a complete result would be impossible within the given time frame. Viewing the project as a journey where we learnt many lessons and gained insights to the subject which we tried to share in this report and summarized in this chapter. Tried to look at the problem from many points of view which generated some new ideas that could be explored in future. It is suggested formal approaches for modelling and analyzing the system which are by no means complete but could become the initiation for further research. We also created a working system and algorithms which we claim to be useful and extensible. However, as we have seen in these chapter, all these achievements are only partially successful. Personally, it would be consider that this project is a success if the ideas described in the report can be a useful reference for future work on the subject.

REFERENCES

1. Anil Kulkarni, Ajay Khandare, Mandar Malve, ”Wireless Sensor Network (WSN) for protection high cost trees in remote jungles from fire and poaching”, International Seminar on Sandalwood: Current Trends and Future Prospects , Feb 2014,pp.68-73.
2. Digital Output MEMS Accelerometer-ADXL345, Analog Devices, 2009, datasheet available at www.analog.com
3. 3. Sridevi Veerasingam, Saurabh Karodi, Sapna Shukla, Design of Wireless Sensor Network node on Zigbee for Temperature
   Monitoring”,2009 International Conference on Advances in Computing, Control and Telecommunication Technologies, IEEE Journals 978-0-7695-3915-7/09,2009.
4. Manish Y. Upadhye, P. B. Borole, Ashok K. Sharma, Real-Time Wireless Vibration Monitoring System Using LabVIEW, 2015 International Conference on Industrial Instrumentation and Control Pune, India. May 28-30, 2015, pp. 925- 928.
5. Pedro Cheong, Student Member, IEEE, Ka-Fai Chang, Member, IEEE, YingHoi Lai, Sut-KamHo,Iam-Keong Sou, and Kam-Weng Tam, Senior Member, IEEE,”A ZigBee-Based Wireless Sensor Network Node for Ultraviolet Detection of Flame”,IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, VOL. 58, NO. 11, NOVEMBER 2011.
6. Banzi, Jamal. (2014). A Sensor Based Anti-Poaching System in Tanzania National Parks. International Journal of Scientific & Technology Research. Volume 4.