International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362

FLOOD AND FIRE MONITORING USING IOT

T Sai Lalith Prasad , Assistant Professor, Department of Computer Science and Engineering,
Teegala Krishna Reddy Engineering College, Meerpet, Hyderabad,sailalith15@gmail.com
D.SHIVA SAI , CSE , Teegala Krishna Reddy Engineering College, Meerpet, Hyderabad .
B.SUKESH , CSE , Teegala Krishna Reddy Engineering College, Meerpet, Hyderabad .
N.SIDDARTHA , CSE , Teegala Krishna Reddy Engineering College, Meerpet, Hyderabad .

ABSTRACT:

This paper demonstrates the design process, implementation and experimental verification of a Flood Level Monitoring system. The main objective of the proposed system is to be able to read the water level at every second, display it to the supervisor and alert the affected populace and relevant authorities by means of an alarm and short message system (SMS) when the level of water surpasses a user defined threshold. Global System for Mobile Communications (GSM) network has been used for sending the mobile messages. Fire outbreak is a major concern at forest and homes, offices, industries etc. It is dangerous and requires high security and control to avoid destruction of lives and property. One of the preventive measures to avoid the danger is to install an automatic fire alarm detector at vulnerable locations, hence the Abased fire alarm detection and control system was proposed. It is capable of automatically detecting heat in a given environment, sound an alarm, switch off mains of the building and also spray water to reduce the intensity of fire. The system uses a buzzer, 5v DC (Direct Current) motor, a GSM (Global System for Mobile) to send SMS (Short Message Service) and MSP430 Microcontroller.

Keywords - MSP430 microcontroller, power supply, MQ2 sensor, flame detector, rain sensor, water level sensor, GSM, LED, Buzzer, relay.

INTRODUCTION:

Fire monitoring and alerting system - Ever since mankind first began building structures out of wood rather than stone, fire has been a part of the learning

process. In fact, so common have these infernos been throughout history that nearly every major city in the world has been largely burnt to the ground at one time or another in its history. An example of such fire outbreak was in Boston 1872. While not as large a fire as the one in Chicago the year before or the fire that was to ravage San Francisco just over three decades later. Firefighting requires skills in combating, extinguishing, and preventing fires, responding to emergency calls, operating and maintaining fire department equipment and quarters, and extensive training in performing firefighting activities.

The earliest known firefighters were in the city of Rome. In 60 A.D., emperor Nero established a Corps of Vigils (Vigils) to protect Rome after a disastrous fire. It consisted of 7,000 people equipped with buckets and axes, and they fought fires and served as police . In the 4th century B.C., an Alexandrian Greek named Ctesibius made a double force pump called a siphon. As water rose in the chamber, it compressed the air inside, which forced the water to eject in a steady stream through a pipe and nozzle.

Nowadays, some factories and buildings have proper installation and fire safety and control arrangements such as fire alarm, fire extinguishers, water supply system etc. But the problem is these conventional fire extinguishing systems are not enough to take prompt action during fire outbreak and hence, save life. The best way to reduce these losses is to respond to the emergency situation as quickly as possible.

So, there comes the necessity of a standalone fire detection systems. This project therefore seeks to design a microcontroller fire alarm and control system that will continuously monitor the presence of significant amount of heat and activate an alarm and simultaneously switch off the mains of the building, send an SMS alert and extinguish the fire as a safety measure to contain the situation.

Flood monitoring and alerting system - The global climatic change during the first half of the twentieth century has brought a tremendous impact on the high mountainous glacial environment. Due to the faster rate of ice and snow melting, possibly caused by the global warming, the sea level is increasing rapidly and resulting in a sudden discharge of large volumes of water and debris causing flooding of the low lying areas.

With this ever-increasing regularity of flood damage, a definite need has emerged for an early warning for regions deemed to be 'at high risk' from flooding. Furthermore, the high level of damage to properties and loss of lives are the underlying factor in the development an early warning system - Flood Level Monitoring System. Currently in the South Pacific the Metrological departments get the weather data from the satellite and then predict the average rainfall in the area. Research has been done on Flood monitoring system using GIS hydrological model in China. Again with the use of satellite this system reads the water level and then subtracts the surface elevation to get the height of flood.

Moreover most of Flood level systems depend on on satellite to predict the flood data. However there was a need of a system which automatically reads the data rather than predicting in the threatened area and then alerts the residence instantly. Mobile Phones have become one of the most popular communication devices amongst the people all over the world. At present with the world population of around 6.7 billion, 60% use mobile phones.

This is about 4.1 billion mobile phone users and this assistive technology is used in all facets of our livelihood. Since the GSM network started its operation in 1991, the SMS has become popular as it provides cheap, convenient and a faster method of communication. Recent advances in the automation showed that the billing system for electricity, gas or water uses GSM module based SMS metering service rather than assigning personnel to visit each house and secure the meter readings manually, where these SMS are sent only from module to users. Even remote greenhouse measure and control systems are SMS based.

Then there is an advert of SMS based intelligent homes which are designed to alert the users via SMS in case of emergencies. The owners of smart homes can inter alia switch on/off their lights and appliances via SMS from work, functions or anywhere away from home. Zigbee and Wi-Fi technology is also used for wireless remote controlling, However the distance is very small when compared to GSM control.

LITERATURE SURVEY:

1.ThinagaranPerumal, Md Nasir Suleiman, C. Y. Leong. IoT Enabled Water Monitoring System IEEE Explore, 2015 In this paper, proposed an IoT based water monitoring system that measure water level in real time. The prototype is based on idea that the level of water can be very important parameter when it comes to the flood occurrences especially in disaster prone area. A water level sensor is used to detect the desired parameter and if the water level reaches the parameter the signal will be freed in real time to social network like Twitter. A cloud server was configured as data repository. The measurement of water level are displayed in remote dashboard. The proposed solution with integrated sensory system that allows inner monitoring of water quality. Alerts and relevant data are transmitted over the internet to a cloud server and can be received by user terminal owned by consumer. The outcome of water measurement is displayed in web based remote dashboard.

2. Amjath Ali J 1 , B. Thangalakshmi 2 , A. Vincy Beaulah3 1 Lecturer, Department of Electrical and Electronics Engg., Ibra college of Technology, Disaster-LINK is a smart IoT device that acts as an alarm and monitoring system during natural disasters that operates by communicating over internet. It comes with Wi-Fi support for internet connectivity and uses an International Journal of Engineering Trends and Technology (IJETT) – Volume 67 Issue 11- Nov 2019 ISSN: 2231-5381 http://www.ijettjournal.org Page 28 IoT cloud platform which helps to control, monitor and manage the device. The device senses its local environment using onboard sensors and send early warnings to family, friends and colleagues immediately when it finds a disaster situation. It is also able to receive such warning alarms from other similar devices available on the internet and provide the user with voice, flashing light, SMS and E-mail alarm notifications. The ultimate aim of the project is to spread the disaster warning information quickly through internet and make it available to those who need it as early as possible. The fact that internet is faster than the seismic waves of an earthquake, and much faster than a flood or tsunami, helps the device to deliver the alert message much before the actual calamity reach the user’s location giving that vital extra time to take those precautionary emergency measures.

3.Syed NazmusSakib ; TanjeaAne ; NafisaMatin ; M. Shamim Kaiser This paper presents a neuro-fuzzy controller based flood monitoring system using wireless sensor network. The distributed sensor nodes use IEEE 802.15.4 protocol, also called low rate wireless personal area network, to collect the sensor information such as water level data from the river, rainfall, wind speed and air pressure data from a selected site. In order to validate the proposed flood monitoring system, Chadpur, a flood prone district of Bangladesh, has been considered as selected site. The sensors information are sent to the distributed alert center via Arduino microcontroller and the XBee Transceivers. At the distributed alert center, XBee Transceiver and a Raspberry Pi microcomputer are used to generate flood alert based on sensor information and two decade flood data and these data are stored in a database. Sensor information are analyzed by the intelligent neuro-fuzzy controller used in Raspberry Pi microcomputer to announce the flood alerts. The wireless sensor network is connected as mesh topology which can send signals over far distance. The performance evaluation reveals that the proposed system accurately detect flood alert compared to the existing flood alert system.

4. Edward Udo, EtebongIsongArticle · January 2014 Nigeria as a whole and Uyo, a southern province in Nigeria, in particular is facing a serious challenge with an increasing frequency of flood in recent years. It is therefore crucial to utilize the state-of-the-art sensing and communication technologies to monitor and detect flood occurrences The role of the designed Flood Monitoring and Detection System (FMDS) based on WSN is to continuously monitor, detect and report the environment's status to a control unit using relative humidity, temperature, water level and amount of rainfall as flood indicators, whose values are gathered by sensors in the sensor field. The flood monitoring and detection system monitors and know the development of floods and then send flood notification SMS to the inhabitant of such zones for necessary action. The developed Flood Monitoring and Detection System (FMDS) covers 15 flood prone regions in Uyo metropolis in AkwaIbom State, Nigeria. The GIS map of the flood prone zones is incorporated into the FMDS. The system is composed of three major modules which are the sensor field module, surveillance module and the phone module. The system was developed using Java Programming Language built into surveillance module of the system. The developed system is robust and gives timely alert of flood occurrences.

5. Ahmed Imteaj et.al. Studied the problems faced by factory workers in times when fire breaks out. They proposed a system using Raspberry Pi 3 which is capable of detecting fire and providing information about area of fire. The Raspberry Pi controls multiple Arduino boards which are connected with several motors and cameras to capture the fire incident.In this, they discussed about the modern technology that can be used to reduce extremely unfortunate accidents caused by fire. We designed the whole system and calculated its effectiveness.

6. Ondrej Krejcar proposed a model for location enhancement and personnel tracking using Wi-Fi networks. In this, he has represented the control system concept that is used in handling information of location and control unit operations. The location of the user present in the building, is obtained through Wi-Fi access points . We have studied this to understand the usability of the Wi-Fi networks in live tracking and then have utilized this functionality to track fire and give information about location of fire to various devices intimating people about the mishap.

7. Authors in have studied the safety features in home and industrial areas. They have designed new model using WSN. Not only have they incorporated temperature and humidity sensors but also included fire and International Journal of Pure and Applied Mathematics Volume 118 No. 20 2018, 4249-4253 ISSN: 1314-3395 (on-line version) url: http://www.ijpam.eu Special Issue ijpam.eu 4249 smoke sensors while developing the model. They present a preceding study of WSN is able to detect fire alarm. It is for setting up a wireless sensor network with three sensors. An application was developed for getting home information,

METHODLOGY:

SOFTWARE ARDUNIO IDE:

Introduction:

The Arduino Integrated Development Environment - or Arduino Software (IDE) – contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino and Genuino hardware to upload programs and communicate with them.

Writing sketches:

Programs written using Arduino Software (IDE) are called sketches. These sketches are written in the text editor and are saved with the file extension . The editor has features for cutting/pasting and for searching/replacing text. The message area gives feedback while saving and exporting and also displays errors. The console displays text output by the Arduino Software (IDE), including complete error messages and other information. The bottom right hand corner of the window displays the configured board and serial port. The toolbar buttons allow you to verify and upload programs, create, open, and save sketches, and open the serial monitor.

✓ Verify

Checks your code for errors compiling it. Upload Compiles your code and uploads it to the configured board.

→

See uploading below for details.

Open

Presents a menu of all the sketches in your sketchbook. Clicking one will open within the current window overwriting its content

Note: due to a bug in Java, this menu doesn't scroll; if you need to open a sketch late in the list, use the File | Sketchbook menu instead

Saves your sketch in serial monitor

Additional commands are found within the five menus: File, Edit, Sketch, Tools and Help. The menus are context sensitive, which means only those items relevant to the work currently being carried out are available.

File:

□ New: Creates a new instance of the editor, with the bare minimum structure of a sketch already in place.

□ Open: Allows loading a sketch file browsing through the computer drives and folders.

□ Open: Recent provides a short list of the most recent sketches, ready to be opened.

□ Sketchbook: Shows the current sketches within the sketchbook folder structure; clicking on any name opens the corresponding sketch in a new editor instance.

□ Examples: Any example provided bythe Arduino Software (IDE) or library shows up in this menu item. All the examples are structured in a tree that allows easy access by topic or library.

□ Close: Closes the instance of the Arduino Software from which it is clicked.

□ Save: Saves the sketch with the current name. If the file hasn't been named before, a name

□ and other operating systems char maps.

□ Serial Monitor: Opens the serial monitor window and initiates the exchange of data with any connected board on the currently selected Port. This usually resets the board, if the board supports Reset over serial port opening

□ Board: Select the board that you're using. See below for descriptions of the various boards.

□ Port: This menu contains all the serial devices (real or virtual) on your machine. It should automatically refresh every time you open the top-level tools menu.

- Programmer: For selecting a hardware programmer when programming a board or chip and not using the on board USB-serial connection. Normally you won't need this, but if you’re burning a boot loader to a new microcontroller, you will use this.
- Burn Boot loader: The items in this menu allow you to burn a boot loader onto the microcontroller on an Arduino board. This is not required for normal use of an Arduino or Genuine board but is useful if you purchase a new AT mega microcontroller (which normally comes without a boot loader). Ensure that you've selected the correct board from the Boards menu before burning the boot loader onthe target board. This command also set the right fuses

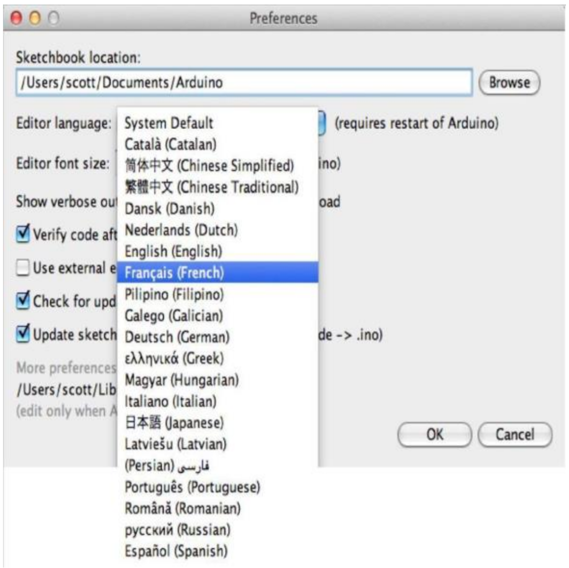
HELP

Here you find easy access to a number of documents that come with the Arduino Software (IDE). You have access to Getting Started, Reference, this guide to the IDE and other documents locally, without an internet connection

Preferences:

Some preferences can be set in the preferences dialog (found under the Arduino menu on the Mac, or File on Windows and Linux). The rest can be found in the preferences file, whose locationis shown in the preference di

LANGUAGE SUPPORT



Preferences

Sketchbook location: /Users/scott/Documents/Arduino

Editor language: (requires restart of Arduino)

Editor font size:

Show verbose ou

☑ Verify code aft

☐ Use external e

☑ Check for upd

☑ Update sketch

More preferences
/Users/scott/Lib
(edit only when A

Fig Language support

o Since version 1.0.1, the Arduino Software (IDE) has been translated into 30+ different languages. By default, the IDE loads in the language selected by your operating system. (Note: on Windows and possibly Linux, this is determined by the

locale setting which controls currency and date formats, not by the language the operating system is displayed in.)

o If you would like to change the language manually, start the Arduino Software (IDE)and open the Preferences window. Next to the Editor Language there is a dropdown menu of currently supported languages. Select your preferred language from the

o menu, and restart the software to use the selected language. If your operating system language is not supported, the

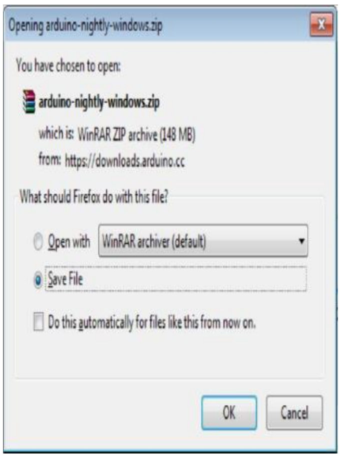
Arduino Software (IDE) will default to English.

- You can return the software to its default setting of selecting its language based
- On your operating system by selecting System Default from the Editor Language dropdown. This setting will take effect when you restart the Arduino Software (IDE).
- Similarly, after changing your operating system's settings, you must restart the Arduino Software (IDE) to update it to the new default language.

ARDUINO UNO INSTALLATION:

- In these we will get know of the process of installation of Arduino IDE and connecting Arduino Uno to Arduino IDE.
- **Step 1 -** First we must have our Arduino board (we can choose our favorite board) and a USB cable. In case we use Adriana UNO, Arduino Delanie, Nano, Arduino Mega 2660, or Decimals, we will need a standard USB cable (A plug to B plug), t In case we use Arduino Nano, we will need an A to Mini-B cable...
- **Step 2** - Download Arduino IDE Software. We can get different versions of Arduino IDE from the Download page on the Arduino Official website. We must select were software, which is compatible with operating system (Windows, IOS, or Linux).

After wear file download is complete, unzip the file. o **Step 3** − Power up our board



Opening arduino-nightly-windows.zip

You have chosen to open:

arduino-nightly-windows.zip

which is: WinRAR ZIP archive (148 MB)

from: https://downloads.arduino.cc

What should Firefox do with this file?

○ Open with WinRAR archiver (default)

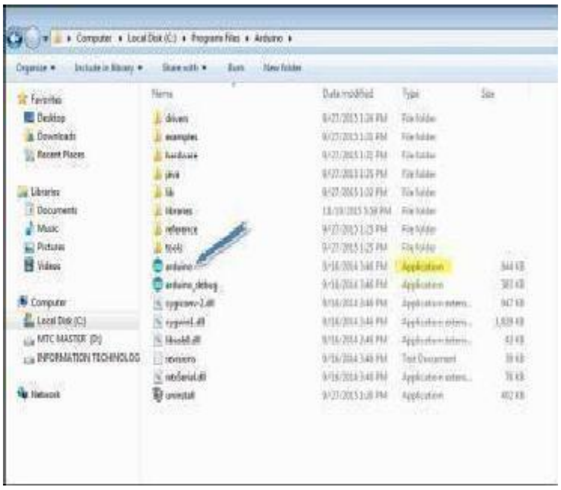
● Save File

☐ Do this automatically for files like this from now on.

The Arduino Uno, Mega, Delanie and Arduino Nano automatically draw power from either, the USB connection to the computer or an external power supply. If we are using an Arduino Decimal, we have to make sure that the board is configured to draw power from the USB connection. The power source is selected with a jumper, a small piece of plastic that fits onto two of the three pins between the USB and power jacks.

Check that it is on the two pins closest to the USB port.Connect the Arduino board to were computer using the USB cable. The green power LED(labelled PWR) should glow.

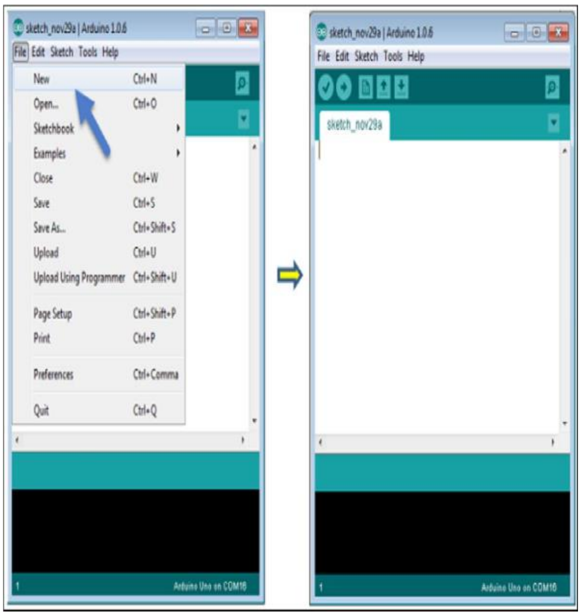
- **Step 4** − Launch Arduino IDE.



Name	Date modified	Type	Size
drivers	9/27/2015 1:26 PM	File folder	
examples	9/27/2015 1:31 PM	File folder	
hardware	9/27/2015 1:32 PM	File folder	
java	9/27/2015 1:25 PM	File folder	
lib	9/27/2015 1:32 PM	File folder	
libraries	10/19/2015 5:58 PM	File folder	
reference	9/27/2015 1:25 PM	File folder	
tools	9/27/2015 1:25 PM	File folder	
arduino	9/16/2014 3:46 PM	Application	844 KB
arduino_debug	9/16/2014 3:46 PM	Application	383 KB
cygiconv-2.dll	9/16/2014 3:46 PM	Application extens...	947 KB
cygwin1.dll	9/16/2014 3:46 PM	Application extens...	1,829 KB
libusb0.dll	9/16/2014 3:46 PM	Application extens...	43 KB
revisions	9/16/2014 3:46 PM	Text Document	39 KB
rxtxSerial.dll	9/16/2014 3:46 PM	Application extens...	76 KB
uninstall	9/27/2015 1:28 PM	Application	402 KB

After our Arduino IDE software is downloaded, we need to unzip the folder. Inside the folder, we can find the application icon with an infinity label (application.exe).Double click the icon to start the IDE.

o **Step 5** − Open our first project.



Once the software starts, we have two options

* Create a new project

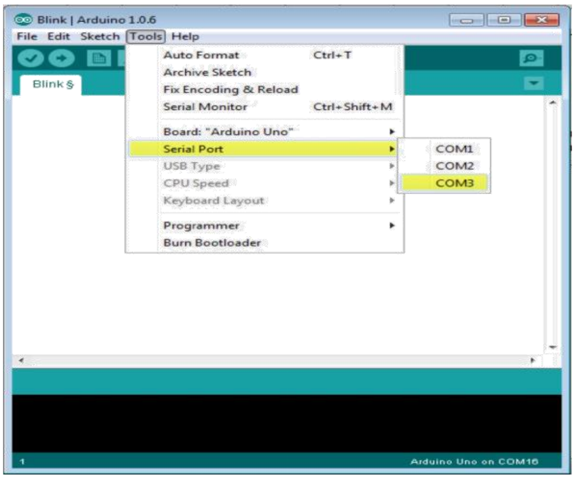
* Open an existing project example.

To create a new project, select File → new.

To open an existing project example, select File → Example → Basics → Blink.

Here, we are selecting just one of the examples with the name Blink. It turns the LED onand off with some time delay. We can select any other example from the list.

o **Step 6** − Select our Arduino board.



To avoid any error while uploading wear program to the board, we must select the correctArduino board name, which matches with the board connected to were computer.

Go to Tools → Board and select wear board.

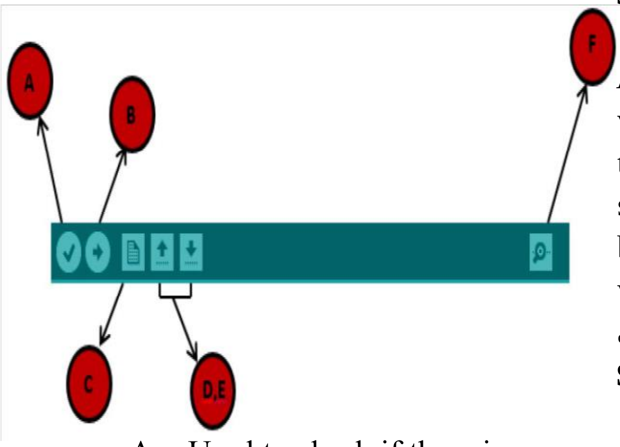
Here, we have selected Arduino Uno board according to our tutorial, but we must selectthe name matching the board that we are using.

- **Step 7** − Select were serial port.

Select the serial device of the Arduino board. Go to Tools → Serial Port menu. Thisis likely to be COM3 or higher (COM1 and COM2 are usually reserved for hardware serial ports). To find out, we can disconnect were Arduino board and re-open the menu, the entry that disappears should be of the Arduino board. Reconnect the board and select that serial port.

- **Step 8** − Upload the program to were board.

Before explaining how we can upload our program to the board, we must demonstratethe function of each symbol appearing in the Arduino IDE toolbar.



A − Used to check if there is any compilation error.

B − Used to upload a program to the Arduino board.

C − Shortcut used to create a new sketch.

D − Used to directly open one of the example sketch.

E − Used to save were sketch.

seconds; we will see the RX and TX LEDs on the board, flashing. If the upload is successful, the message

"Done uploading" will appear in the status bar.

Note − If we have an Arduino Mini, NG, or other board, we need to press the reset button

physically on the board, immediately before clicking the upload button on the Arduino Software.

STEPS TO WRITE A PROGRAM:

STEP1

Arduino microcontrollers come in a variety of types. The most common is the Arduino UNO, but there are specialized variations. Before you begin building, do a little researchto figure out which version will be the most appropriate for your project.

STEP2

To begin, you'll need to install the Arduino Programmer, aka the integrated developmentenvironment (IDE).

STEP3

Connect your Arduino to the USB port of your computer. This may require a specific USBcable.

Every Arduino has a different virtual serial-port address, so you'll need to reconfigure theport if you're using different Arduinos.

STEP4

Set the board type and the serial port in the Arduino Programmer.

STEP5

Test the microcontroller by using one of the preloaded programs, called sketches, in theArduino Programmer. Open one of the example sketches, and

press the upload button toload it. The Arduino should begin responding to the

program: If you've set it to blink anLED light, for example, the light should

start blinking.

BLYNK APP

Blynk was designed for the Internet of Things. It can control hardware remotely, it can display sensor data, it can store data, vizualize it and do many other cool things.

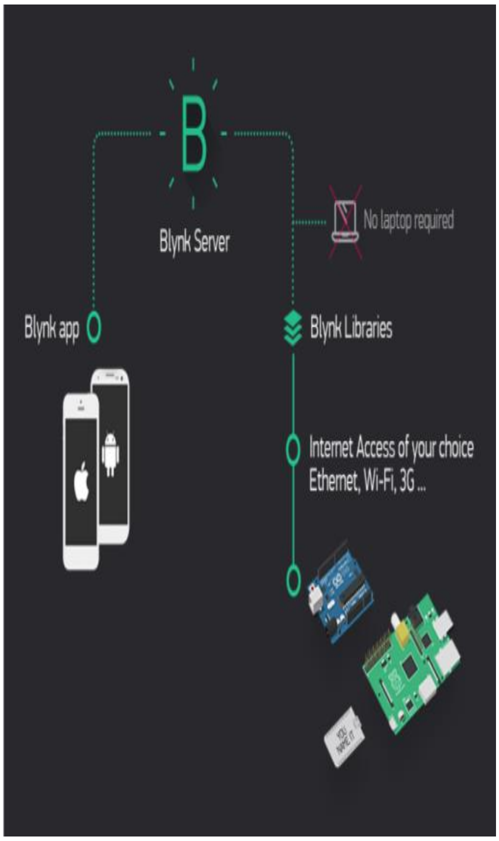
There are three major components in the platform:

- **Blynk App** - allows to you create amazing interfaces for your projects using various widgets we provide.

Blynk Server - responsible for all the communications between the smartphone and hardware. You can use our Blynk Cloud or run your private Blynk server locally. It’s open-source, could easily handle thousands of devices and can even be launched on a Raspberry Pi.

- **Blynk Libraries** - for all the popular hardware platforms - enable communication with the server and process all the incoming and outcoming commands.

Now imagine: every time you press a Button in the Blynk app, the message travels to the Blynk Cloud, where it magically finds its way to your hardware. It works the same in the opposite direction and everything happens in a blynk of an eye.



- Similar API & UI for all supported hardware & devices
- Connection to the cloud using:
  - Wi-Fi
  - Bluetooth and BLE
  - Ethernet
  - USB (Serial)

GSM

- Set of easy-to-use Widgets
- Direct pin manipulation with no code writing
- Easy to integrate and add new functionality using virtual pins

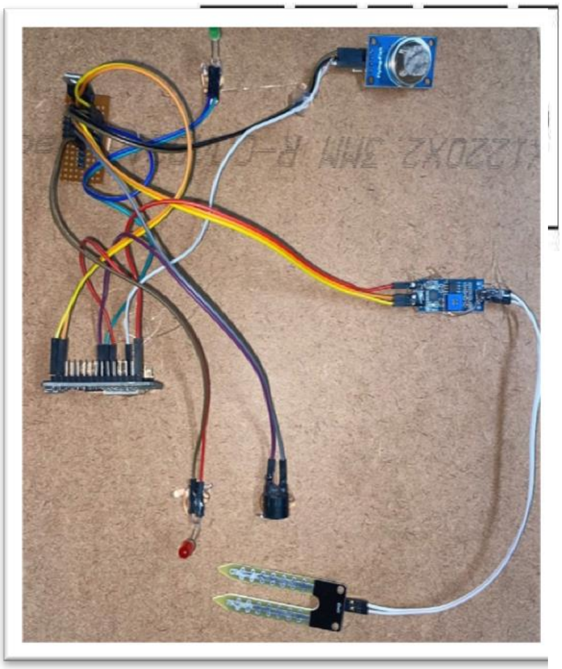
- History data monitoring via Super Chart widget
- Device-to-Device communication using Bridge Widget
- Sending emails, tweets, push notifications, etc.
- New features are constantly added!

You can find example sketches covering basic Blynk Features. They are included in the library. All the sketches are designed to be easily combined with each other.

The Blynk App is a well designed interface builder. It works on both iOS and Android, so no holy wars here,

RESULTS :

OUTPUT SCREENS:



IoT Early Flood Detection & Avoidance System" is an intelligent system which keeps close watch over various natural factors to predict a flood, so we can embrace ourselves for caution, to minimize the damage caused by the flood. Natural disasters like a flood can be devastating leading to property damage and loss of lives. To eliminate or lessen the impacts of the flood, the system uses various natural factors to detect flood. The system has a wi-fi connectivity, thus it's collected data can be accessed from anywhere quite easily using IoT.

To detect a flood the system observes various natural factors, which includes humidity, temperature, water level and flow level. To collect data of mentioned natural factors the system consist of different sensors which collects data for individual parameters. For detecting changes in humidity and temperature the system has a DHT11 Digital Temperature Humidity Sensor. It is a advanced sensor module with consists of resistive humidity and temperature detection components. The water level is always under observation by a float sensor, which work by opening and closing circuits (dry contacts) as water levels rise and fall. It normally rest in the closed position, meaning the circuit is incomplete and no electricity is passing through the wires yet. Once the water level drops below a predetermined point, the circuit completes itself and sends electricity through the completed circuit to trigger an alarm. The flow sensor on the system keeps eye on the flow of water. The water flow sensor consists of a plastic valve body, a water rotor, and a hall-effect sensor. When water flows through the rotor, rotor rolls. Its speed changes with different rate of flow.

The system also consist of a HC-SR04 Ultrasonic Range Finder Distance

Sensor. The Ultrasonic sensor works on the principle of SONAR and is designed to measure the distance using ultrasonic wave to determine the distance of an object from the sensor. All the sensors are connected to Arduino UNO, which processes and saves data. The system has wi-fi feature, which is useful to access the system and its data over IoT.

Fire Detectors play a very important role in Industries, Shops, Malls, Residential complexes, and parking areas. They help in detecting fire or smoke at an early stage and can help in saving lives. Commercial Fire detecting systems usually have an alarm signaling, with the help of a buzzer or Siren. We have designed an IOT based Fire Alerting System using a Temperature sensor, flame sensor, and smoke sensor. This project would not only signal the presence of fire in a particular premise but will also send related information through IOT.

IOT Based Fire Alerting System uses three Sensors, namely, Temperature, Flame, and Smoke sensors. Adruino has an inbuilt ADC converter, which converts the analog signals received at the sensor end to digital.

The Arduino is programmed to turn on the buzzer when the temperature or flame or smoke reaches a threshold value. At the same time, Arduino sends the data to the GSM module. GSM modem is used for connecting microcontrollers to the internet. GSM modem will then send the following data to the IOT website, where, authorized people can take appropriate measures in order to curb the fire.

1. Temperature (in Degrees Celsius)
2. Smoke Value (in Percentage)
3. Flame status (ON or OFF)
4. Device ID
5. Date and Time Stamp

The device ID is the unique ID given to a device, which would help the person get information related to the location, where the fire is detected.

The Prerequisite for this IoT-based fire alarming system is that the GSM module should be connected to a network. This project is also implemented without the IOT module. In place of the IOT module, we have sent SMS alerts and in this project, an SMS is triggered when the buzzer is turned ON.

Technical Specifications of IOT based Fire Alerting System

The components that are used in the IOT based Fire Alerting System using the Temperature and Smoke Sensor project are:

1. Smoke sensor
2. Temperature sensor
3. Flame sensor
4. Microcontroller – Arduino.
5. GSM Technology
6. 6.Buzzer

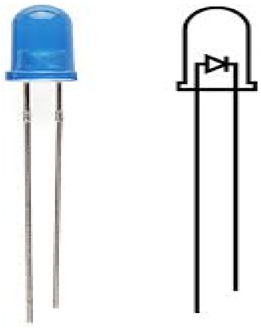
NodeMcu

TheNodeMCU(***N***ode ***M***icro***C***ontroller ***U*** nit) is an open-source software and hardware development environment built around an inexpensive System-on-a-Chip (SoC) called the ESP8266. The ESP8266, designed and manufactured by Espressif Systems, contains the crucial elements of a computer: CPU, RAM, networking (WiFi), and even a modern operating system and SDK. That makes it an excellent choice for Internet of Things (IoT) projects of all kinds.

However, as a chip, the ESP8266 is also hard to access and use. You must solder wires, with the appropriate analog voltage, to its pins for the simplest tasks such as powering it on or sending a keystroke to the "computer" on the chip. You also have to program it in low-level machine instructions that can be interpreted by the chip hardware. This level of integration is not a problem using the ESP8266 as an embedded controller chip in mass-produced electronics. It is a huge burden for hobbyists, hackers, or students who want to experiment with it in their own IoT projects.

The Arduino project created an open-source hardware design and software SDK for their versatile IoT controller. Similar to NodeMCU, the Arduino hardware is a microcontroller board with a USB connector, LED lights, and standard data pins. It also defines standard interfaces to interact with sensors or other boards. But unlike NodeMCU, the Arduino board can have different types of CPU chips (typically an ARM or Intel x86 chip) with memory chips, and a variety of programming environments. There is an Arduino reference design for the ESP8266 chip as well. However, the flexibility of Arduino also means significant variations across different vendors. For example, most Arduino boards do not have WiFi capabilities, and some even have a serial data port instead of a USB port.

Light-Emitting Diodes:



A **light-emitting diode** (**LED**) is a semiconductor device that emits light when current flows through it. Electrons in the semiconductor recombine with electron holes, releasing energy in the form of photons. The color of the light (corresponding to the energy of the photons) is determined by the energy required for electrons to cross the band gap of the semiconductor.[5] White light is obtained by using multiple semiconductors or a layer of light-emitting phosphor on the semiconductor device.

Appearing as practical electronic components in 1962, the earliest LEDs emitted low-intensity infrared (IR) light. Infrared LEDs are used in remote-control circuits, such as those used with a wide variety of consumer electronics. The first visible-light LEDs were of low intensity and limited to red. Early LEDs were often used as indicator lamps, replacing small incandescent bulbs, and in seven-segment displays. Later developments produced LEDs available in visible, ultraviolet (UV), and infrared wavelengths, with high, low, or intermediate light output, for instance white LEDs suitable for room and outdoor area lighting. LEDs have also given rise to new types of displays and sensors, while their high switching rates are useful in advanced communications technology with applications as diverse as aviation lighting, fairy lights, automotive headlamps, advertising, general lighting, traffic signals, camera flashes, lighted wallpaper, horticultural grow lights, and medical devices

LEDs have many advantages over incandescent light sources, including lower power consumption, longer lifetime, improved physical robustness, smaller size, and faster switching. In exchange for these generally favorable attributes, disadvantages of LEDs include electrical limitations to low voltage and generally to DC (not AC) power, inability to provide steady illumination from a pulsing DC or an AC electrical supply source, and lesser maximum operating temperature and storage temperature. In contrast to LEDs, incandescent lamps can be made to intrinsically run at virtually any supply voltage, can utilize either AC or DC current interchangeably, and will provide steady illumination when powered by AC or pulsing DC even at a frequency as low as 50 Hz. LEDs usually need electronic support components to function, while an incandescent bulb can and usually does operate directly from an unregulated DC or AC power source.

As a transducer of electricity into light, LEDs operate in reverse of photodiodes.

Buzzer

A **buzzer** or **beeper** is an audio signaling device which may be mechanical, electromechanical, or piezoelectric (*piezo* for short). Typical uses of buzzers and beepers include alarm devices, timers, train and confirmation of user input such as a mouse click or keystroke.

CONCLUSION :

This project demonstrates the design process, implementation and experimental verification of a Flood Level Monitoring system. The main objective of the proposed system is to be able to read the water level at every second, display it to the supervisor and alert the affected populace and relevant authorities by means of an alarm and IOT notification when the level of water surpasses a user defined threshold. Fire outbreak is a major concern at homes, offices, industries etc. It is dangerous and requires high security and control to avoid destruction of lives and property. One of the preventive measures to avoid the danger is to install an automatic fire alarm detector at vulnerable locations, hence the Arduino based fire alarm detection and control system was proposed. It is capable of automatically detecting heat in a given environment, sound an alarm, switch off mains of the building and also spray water to reduce the intensity of fire in the future.

FUTURE ENHANCEMENT:

This prototype can be used in different applications as following to see in future use are ;

1.Water level bodies like dams, drinking water supply

2.Main switch boards

3.In the household applications

4.Industrial purposes also

REFERENCES:

1.Heath, Steve (2003). Embedded system design. EDN series for design engineers (2nded.), Newness, pp-11-12. ISBN 9780760655460

2.David Harris & Sarah Harris (2012) Digital Design and Computer Architecture, Second Edition, p.515 Morgan Kaufmann, ISBN-0123944244.Robert Katona;

3.Dénes Fodor,Texas instruments MSP430 microcontroller based portable multi-purpose instrument for android platforms Education and Research Conference (EDERC), 2014 6th European Embedded Design in Year: 2014,Pages: 1 -5, DOI: 10.1109/EDERC.2014.6924347

4.Shize Guo; Xinjie Zhao; Fan Zhang; Tao Wang; Zhijie Jerry Shi; Francois-Xavier Standaert; Chujiao Ma,Exploiting the Incomplete Diffusion Feature: A Specialized Analytical Side-Channel Attack Against the AES and Its Application to Microcontroller Implementations

5.Year: 2014, Volume: 9, Issue: 6,Pages: 999 - 1014, DOI: 10.1109/TIFS.2014.2315534.Beneteau ;

6.G. Di Caterina; L. Petropoulakis; J. J. Soraghan,Low-cost wireless surfaceEMG sensor using the MSP430 microcontroller Education and Research Conference (EDERC), 2014 6th European Embedded Design in

Year: 2014,Pages: 264 - 268, DOI: 10.1109/EDERC.2014.6924401.