International Journal of Cloud Computing and Database Management

E-ISSN: 2707-5915
P-ISSN: 2707-5907
IJCCDM 2025; 6(1): 70-75
www.computersciencejournals.com/ijccdm
Received: 18-03-2025
Accepted: 24-04-2025

Vaibhavi Painuly
Department of Computer Applications, Tula's Institute, Dehradun, Uttarakhand, India

Corresponding Author:
Vaibhavi Painuly
Department of Computer Applications, Tula's Institute, Dehradun, Uttarakhand, India

Intrusion alert system

Vaibhavi Painuly

DOI: https://www.doi.org/10.33545/27075907.2025.v6.i1a.103

Abstract
Some existing security systems are easily tripped by small critters such as cats or squirrels and react to slightest movement. These factors trigger annoying false alarm and even worse, some alarms are unable to alert house occupants of unexpected intrusion and goes off when it is too late. These alarms also, cannot be reset unless the owner is in the house to push the reset button. The objective of this project is to design a system that senses changing pressure on different spots which are the roof and ceiling as well as to construct a system that warns and notify the owner and the house occupants. This project is developed by using Arduino Mega aided with Node MCU to enable internet implementation that can reset the system through button widgets. Inputs used are Piezoelectric Sensor and Passive Infrared (PIR) Motion Sensor while the outputs are Light Emitting Diode (LED), \ Liquid Crystal Display (LCD) and buzzer. Hence, this project will develop a system that can prevent false alarm and alert household of unexpected intruders as well as notify the owner of the house of their property while they are on a vacation or outstation for work.

Keywords: Component; Arduino, IoT, Node MCU

Introduction
Nowadays, a lot of houses are equipped with security systems that utilizes sensors, cameras and motion detectors. When an uninvited guest appears and detected by the system, the sensors will send information to trigger the alarms. Unfortunately, the existing systems can be triggered easily by small creatures or objects that passes by the sensors which will cause false alarm. Such system is very inefficient as it consumes a lot of precious time. Furthermore, the owner of the house will get complaints about the noise that is triggered by the false alarm. Our project is about having a reliable system that will alert the user as well as the neighbourhood, when something happened in our house without our knowledge, specifically the ceiling and rooftop. It concerns about the safety of our house, properties and people living inside. Lately statistics shows that burglary and robbery crimes in Malaysia increases as day passes. Such crimes threaten the lives of the innocents, especially our family, friends and neighbours. This system will alert the user even though the user is far away from home as well as the people around the house when there are intruders trying to get into the house. It will be triggered when it senses multiple heavy movement caused by the weight of the thieves. From there, this project is designed so that no false alarm can be\ triggered by small animals or objects. This system will trigger the alarm and send notification to the owner when it detects spiking change of weight as well as sudden motions around the targeted area. Even when the system triggers a false alarm, and the owner is not around the house, the system can be reset by tapping the widget in the owner 's smartphone.

Methodology
This section will explain about the input of the project, microcontroller used and the output of the system. Piezoelectric sensor and PIR motion sensor are used as an input in this project which essentially used to detect intruders. The flow chart in Figure 1 shows the process of the system. The input will send to Arduino Mega 2560 that acts as the controller of the project. When the input, piezoelectric sensor detects changes in weights or PIR motion sensor detects a movement, it will then trigger Light Emitting Diodes (LED), Piezo Buzzer, trigger Node MCU to send a notification to the user and display using Liquid Crystal Display (LCD) which act as the outputs of the project.

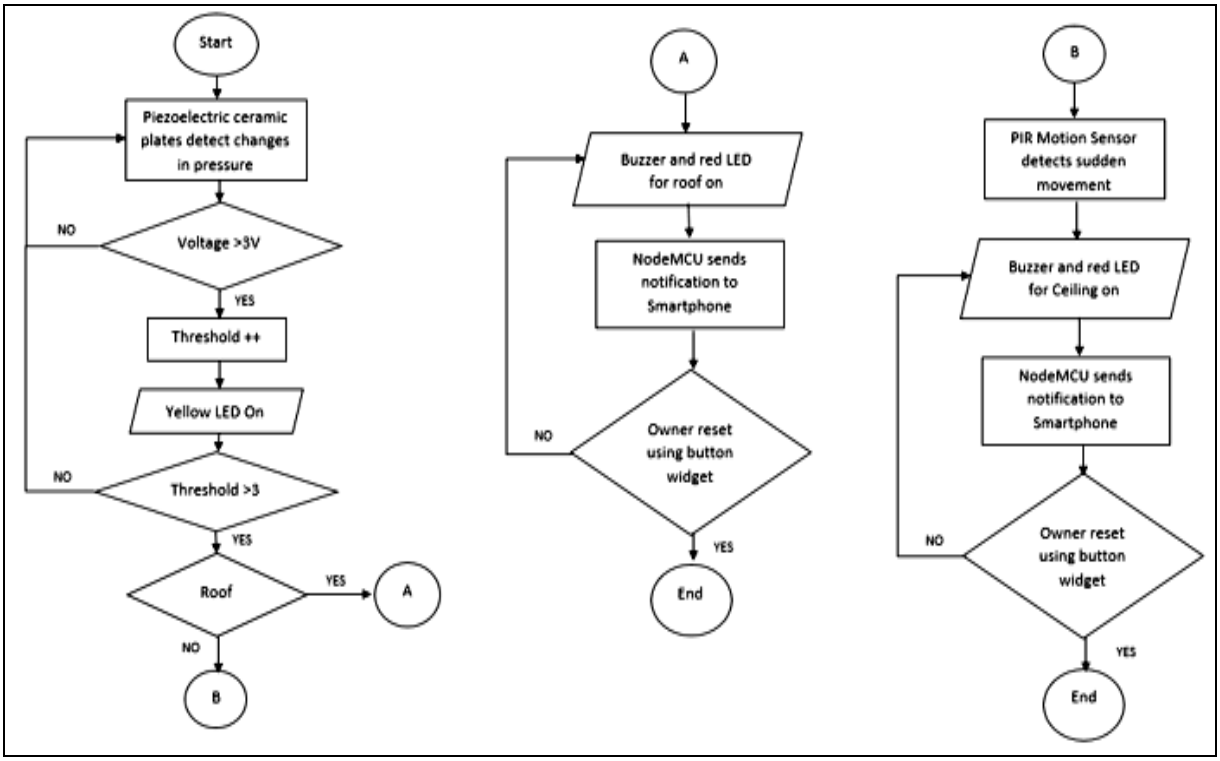

Fig 1: Flow Chart of the Intrusion Alert System

The hardware of the project showed in Figure 2. Number of Input consists of four components. There are Piezoelectric Sensor, PIR Motion Sensor, button and Node MCU meanwhile the number of outputs are consists of nine components. There are 5 LEDs, 16x2 LCD, buzzer, Node MCU and a Smart Phone. The Arduino Mega 2560 is an ATmega2560 based microcontroller board. It has 54 digital input / output pins (including 15 for PWM outputs), 16 analogy inputs, 4 hardware serial ports, a 16 MHz crystal oscillator, a USB interface, a power jack, an ICSP header, and a reset button. A piezoelectric sensor also called a piezoelectric transducer is a device that uses piezoelectric effect to observe changes in pressure, acceleration, temperature, strain or force by converting them into and electrical charge. The ability of piezoelectric material to convert mechanical stress into electrical charge is called piezoelectric effect. Generated piezoelectricity is proportional to the pressure applied to solid piezoelectric crystal materials. Liquid-Crystal display (LCD) is a flat panel display that uses the light-modulating properties of the liquefied crystals. Liquid crystals do not emit direct light, but instead produce colour or monochrome images by using backlight or reflector. Arbitrary images as in a common computer display or static pictures with low content, such as preset letters, digits, or seven segment displays can be displayed or blurred with LCD. Based on the polarizer configuration, LCDs may be either normally on positive or off negative. Passive Infrared Sensor are known as PIR sensor. It is an electronic sensor that detects infrared light produced by objects in its field of view. The term passive refers to the fact that PIR devices do not produces any energy for detection purposes. They work by detecting the infrared emitted or reflected by other objects.

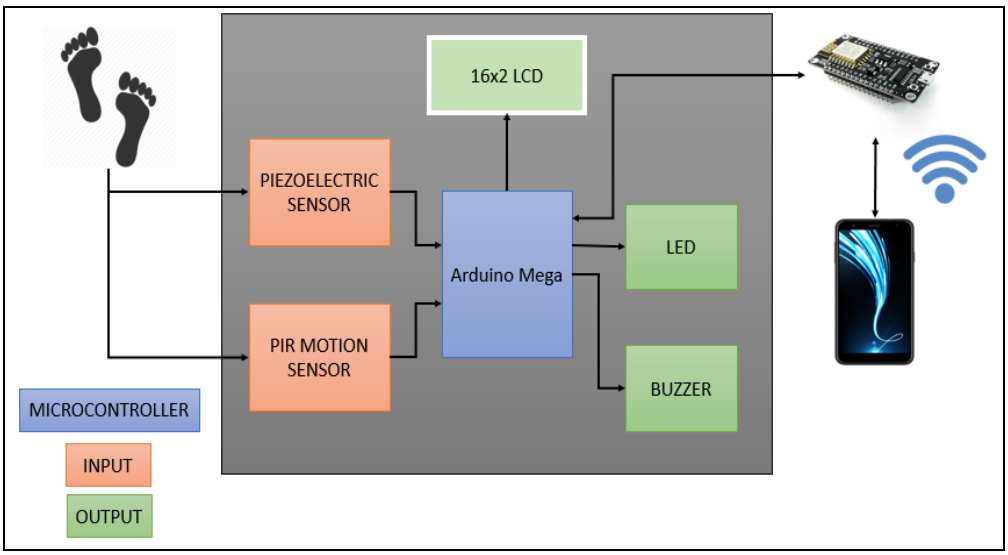

Fig 2: Block diagram of the Intrusion Alert System

If This Then That, also known as IFTTT is shown in Figure 3, is a free web-based service to create chains of simple conditional statements, called applets. Changes in other web services such as Gmail, Twitter, Telegram and Instagram causes the applet to trigger. For example, if the user tweets using a hashtag, an applet can send an e-mail message or copy a photo to a user's archive on Facebook if someone tags a user in a photo. Services are IFTTT's fundamental building blocks. They identify mostly a collection of information from a particular web service such as Twitter or Gmail. Services may also explain managed behaviour with some APIs, such as SMS. They are the output resulting from the trigger input

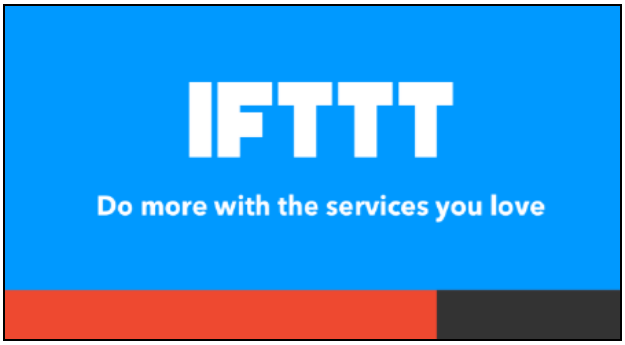

Fig 3: IFTTT software

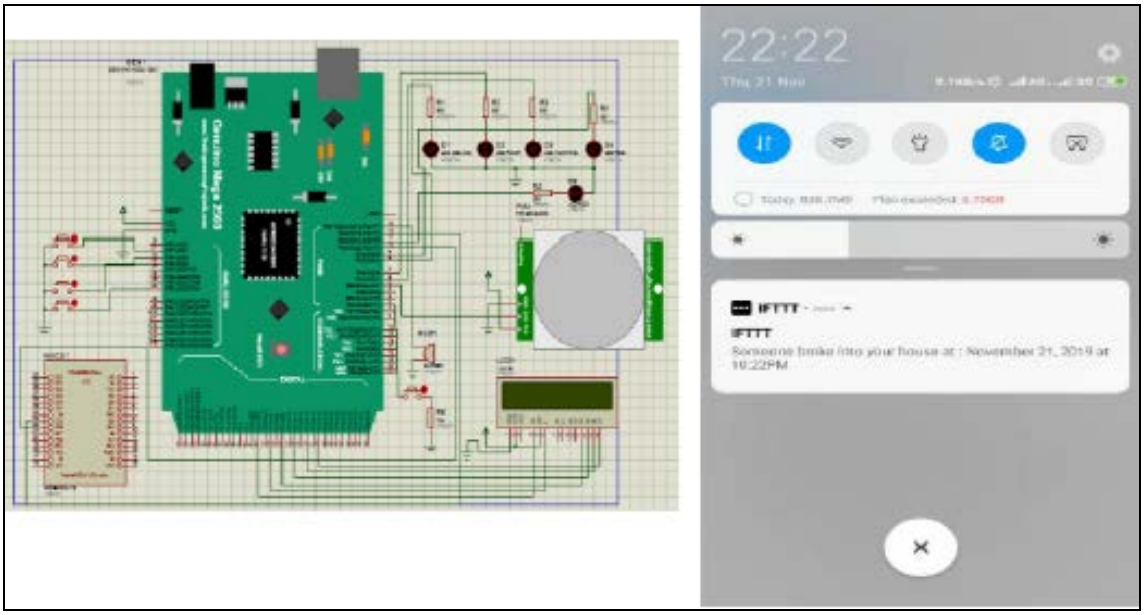

Fig 4: Circuit simulations of the Intrusion Alert System

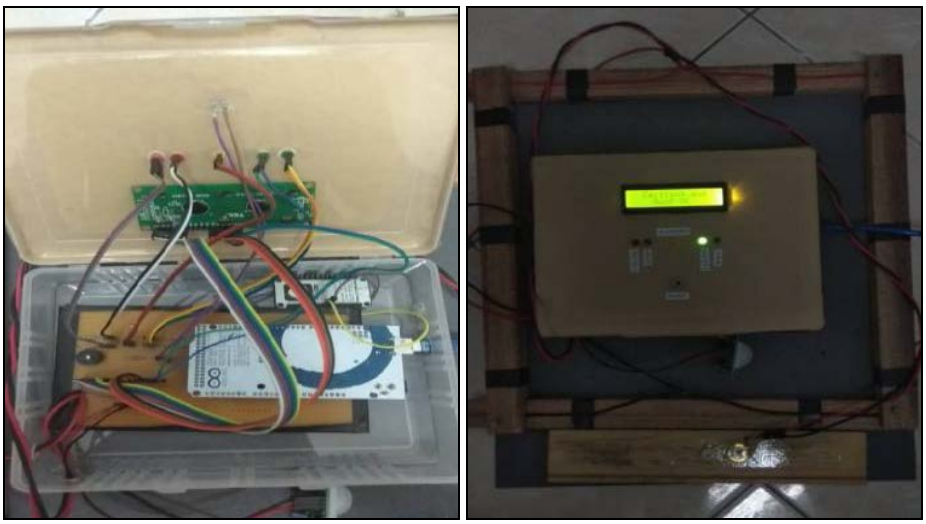

Fig 5: Hardware for Intrusion Alert System

The usage of Arduino Mega as a microcontroller on its own has a few flaws that hinders internet connection. For such reason, Node MCU is added into the system as a second microcontroller that can control internet traffic. Thus, the codes written were divided into two distinct parts, each to be uploaded into Arduino Mega and Node MCU respectively. The library used are Software Serial's and Liquid Crystal. These libraries are used to enable the Arduino to communicate properly with the outputs attached to any of its pins, in their respectively language. Software Serial library opens the opportunity for any Arduino to send and receive data through Serial Communication using two pre-declared RX and TX pins and suitable baud rate, while Liquid Crystal is a library that is used whenever a user wishes to use an LCD to display necessary data or messages. Serial communication port can be pre-declared by using the following command, having _tomcu 'as the channel name followed by pins designated to be RX and TX respectively.

In this case, the channel name is tomcu, while its RX and TX pins are 10 and 12 respectively. In order to control the alarm, a switch is required. Other than the reset button, the Arduino can be reset programmatically using a\ specific function These variables does not, however, reset the Arduino, although it will be soon, instead they hold a certain value to control the switch. These variables are a great necessity as an\ indication for the Arduino from the Node MCU to control the switch ‘s state – ON or OFF. By default, trigger variable is TRUE to switch the NodeMCU from normal state to _panic ‘mode. Panic mode sends notifications and awaits further instruction from user online. Enable variable controls the Arduino ‘s alarm system and by default, it holds a TRUE to allow alarm to turn on whenever it needs to, but once a STOP instruction is sent by NodeMCU, this value turns FALSE and the Arduino resets automatically.

```
//DIGITAL OUTPUT
int ledgreen1 = 6;            //ROOF LED
int ledgreen2 = 7;            //CEIL LED
int ledyellow = 8;
int ledred1 = 9;              //ROOF LED
int ledred2 = 11;             //CEIL LED
int buzzer = 13;              //BUZZER

//PIR
int sensor = 5;               //SENSOR PIR
int state = LOW;              //DEFAULT : NO MOTION
int pir = LOW;                //SENSOR STATUS
```

Fig 6: Coding Output for PIR sensor

These variables hold the digital output values is shows in Figure 6. Since the design implemented two detecting zones; the roof and the ceiling, there will be a need for two green LEDs to flash during normal condition, a common yellow LED to alert intrusions from any of the two zones, and two red LEDS to turn on during an intrusion in followed by a buzzer. These values are stored as ledgreen1, ledgreen2, led yellow, ledred1, ledred2 and buzzer respectively. To use a PIR sensor, a control variable, holding digital value of HIGH or LOW, must be declared in order to detect changes of state, aside of having a pre-declared integer as a pin indicator. PIR is a variable, which define the initial state of the PIR, considering LOW states having nothing in front of the sensor directly, whereas state variable\ is used to detect movement of passing objects, giving out HIGH state whenever an object is stationary in front of it.

```
//ANALOG VARIABLE
float analog1 = 0.00;         //ROOF-ROOF
float voltage1 = 0.00;        //ROOF-ROOF
int threshold1;               //ROOF-ROOF
float analog2 = 0.00;         //CEIL-CEIL
float voltage2 = 0.00;        //CEIL-CEIL
int threshold2;               //CEIL-CEIL
```

```
//LCD JER
const int rs = 40, en = 38, d4 = 36, d5 = 34, d6 = 32, d7 = 30;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
```

```
//FUNCTIONS
void alarm1();
void alarm2();
void blinky1();
void blinky2();
void blinkywarn();
void check1();
void check2();
void writing();
void reading();
void (*resetFunc)(void) = 0;
```

```
//FUNCTIONS
void alarm1();
void alarm2();
void blinky1();
void blinky2();
void blinkywarn();
void check1();
void check2();
void writing();
void reading();
void (*resetFunc)(void) = 0;
```

Fig 7: Coding for Analog variable

These are the functions declared when a certain condition is met. The main objectives of having plenty of functions available in Figure 7 is to have a better control in actions intended in the system. The function named alarm is used whenever the threshold value reached the maximum value, causing the buzzer to go off and notification to be sent into mobile phones through IFTTT application. Next, blinky functions are used to indicate a normal operating condition, assuming no potential harm to the household through a constant blink of green LED that is assorted into two distinct sections of the house – the roof and the ceiling. A single function blinky warn is used to alert the user of potential intrusion through a long beep of buzzer and bright yellow light from an LED. This function also increases threshold each time it is called. Function check is called to constantly calculate the potential difference of the piezoelectric sensor, converting inconvenient analogy value into a proper voltage value. reading and writing functions are used to control the activity of the NodeMCU especially through serial communications of two designated pins. Lastly, reset Func is called if the Arduino needs to be reset

```
void writing() {

  ledstate = digitalRead(led);
  Serial.print("LED : ");
  Serial.print(ledstate);
  Serial.print("\n");

  if (i < 1) {
    light();
    delay(300);
  }

  if (i < 1) {
    alarm();
    delay(200);
    thing.handle();
  }

  delay(2000); //Give time for server to light LED

  while (ledstate == LOW) {
    Serial.println("Waiting for off LED");
    delay(2000); //Give time for server to light LED
    thing.handle();
    ledstate = digitalRead(led);
    Serial.println("LED: ");
    Serial.print(ledstate);
    Serial.print(ledstate);

    i++;
  }

  while (ledstate == HIGH) {
    thing.handle();
    ledstate = digitalRead(led);
    disable = false;
  }

  if (tomega.available() > 0) {
    tomega.write(disable);

    Serial.print("Disable: ");
    Serial.print(disable);
    Serial.print("\n");
  }

  if (disable == false) { //this is to disable
    trigger = false;

    threshold = 0;
    i = 0;
  }

    Serial.print("Delay time to allow trigger change");
    Serial.print(trigger);
```

Fig 8: Coding for Writing Function

This is alarm and light function. Whenever they are called, the function will send a web request to IFTTT server and a notification is sent. Finally, is the writing function showed in Figure 8. This function controls the state of the LED. The LED is first controlled from the server and later, the NodeMCU will attempt to read its state. The state will then deter the next action the NodeMCU must execute. If the LED state is HIGH, meaning the LED is turned off (the system is active low), alarm will be turned on. However, when it is otherwise, the system loops again until there are changes in\ threshold value.

Conclusion

To conclude, the project can provide an extra layer of security to house occupants, designed specifically to alert the whole neighbourhood, using only a small sensor of 15mm diameter plate. The system is not only time-saving, as it can be reset remotely and capable of reaching out household of wide spreaded neighbourhood, it is also cost efficient as replacing materials cost user cheaply. Other than that, the system is also readily available to be integrated with another closed system. Users can connect their devices, for instance washing machine and house lights to the system, through the internet. Since the system uses a common server to manipulate its actions, it is also possible to control more devices from their mobile phones. The cheap assembling cost is affordable by almost everyone. Therefore, this project had been successfully completed, with of course more improvements that can be added into the system to increase its reliability, functionality and efficiency in order to achieve a common goal which is family safety as well as home security.

Acknowledgments

The author wishes to express her gratitude to Universiti Teknologi MARA (UiTM) Johor, Pasir Gudang Campus for providing the laboratory facilities and Research management Institute (RMI) UiTM for the financial support.

References

1. Arnau A. Piezoelectric transducers and applications. Berlin: Springer; 2004. p. 1-320.
2. Monk S. Arduino: getting started with sketches. 2nd ed. Toronto: McGraw-Hill; 2012. p. 1-210.

3. Trunk A, Inman DJ. Piezoelectric energy harvesting. Chichester: Wiley; 2011. p. 1-264.
4. Shamieh C, McComb G. Electronics for dummies. 2nd ed. Hoboken: Wiley; 2009. p. 1-350.
5. Moubarak P, *et al*. A self-calibrating mathematical model for the direct piezoelectric effect of a new MEMS tilt sensor. IEEE Sens J. 2011;11(12):3201-3208.
6. Praveena KS, Bhargavi K. A real-time approach for home security and alert system using CAN protocol. Int Res J Eng Technol. 2016;3(5):2395-2402.
7. Norrosnan MH. Home security system. Melaka: Universiti Teknikal Malaysia, Faculty of Electronic and Computer Engineering; 2010. p. 1-85.