ACTA SCIENTIFIC COMPUTER SCIENCES

Volume 2 Issue 2 February 2020

Review Article

Pre-Protection of Home Appliances from Overpower Using Mobile Phone

Abd Rahim Bin Mohamed Kifli and Nader Barsoum*

Faculty of Engineering, University Malaysia Sabah, Malaysia

***Corresponding Author:** Nader Barsoum, Faculty of Engineering, University Malaysia Sabah, Malaysia.

Received: December 31, 2019

Published: January 17, 2020

Abstract

Overpower may cause fire, explosion, and damages to the appliances that happen because of overload. thus, this paper purposes a new form of protection system at home using wifi to monitor the current usage and control the on/off of the appliances via mobile phone. Thus a prototype is developed using NodeMCU as the microcontroller, 4-channel relay board as switchs, led to represent home appliances and ACS712 as a current sensor. NodeMCU has wifi module inside it that able the wifi connection. ACS712 current sensor measures the current based on the on/off of the appliances. When the current measured reaches the current limit set in the code, the push notification will be send to the user via mobile phone and user have to turn off the appliances from his mobile phone to avoid overload. the current limit is set less than the maximum rated current of the circuit to acknowledge a pre-protection to the appliances. For the interface with the mobile phone, blynk application is used which able the control of the hardware and visualize the sensor data via mobile phone. Perhaps this development is best to be implemented on the extension board which often face overpower

Keywords: Relay; Protection; Appliances; Overpower; Wifi; Blynk; Mobile; Prototype

Introduction

In this modern world, almost all of the society used mobile phone in their daily live. It has been stated that, by 2025, mobile phone is not a device used for communication only, it is a device that act like a remote control of your life [2]. The idea of this paper is to use Internet of Things (IoT) platform to protect home appliances from overload situation. An overload may occur if we use more energy or electricity than the circuit can safely handle. All electrical circuit are designed with specific and limited amount of electricity and made from wiring, fuse, devices, appliances and etc. The power usage or electricity usage of the running devices will add to the total load of the circuit. At one point, when it exceeds the rated load for the circuit wiring, it may cause the circuit to burn or at a worst case explode [1] have created the system for controlling home and office appliances using mobile phone via Bluetooth connection In their system, when a user decides to control an appliance, the controller device downloads an abstract functional description from the appliance and uses that description to automatically generate an interface. A pre-protection of home appliances from overpower by using mobile phone is one of the initiatives toward power saving approach and modernity. This system provides various services to remotely control home appliances via mobile phone. Furthermore, this system measures and shows the reading of power usage and send the notification to the user in case of overpower. The system used the NodeMCU module that act as a brain and to form a wireless communication. In this design, Blynk application is used as the interface between microcontroller and the mobile phone. Therefore, this paper presents the process of building a protection system for home appliances from overpowers via a mobile phone monitoring.

Proposed prototype

The overload of circuit may cause damage to our appliances and most importantly it is very dangerous and can cause fire. This is often happened to the extension board due to the multiple appliances

plug on it that exceeds the maximum current rating. This is ofte happened because we are unaware and unconscious about the cur rent usage since there are no monitoring system that can measur and shows the reading of the current usage. Furthermore, there i no notification that give alert to the user regarding the current us age [3]. Notification is needed since we are not able to observe the current usage all the time.

Therefore, a new protection system for home appliances is proposed to operate by a mobile phone monitoring to prevent any overpower. A prototype is aimed to be designed, having the following scope of development.

- To model a system that measure the electricity parameter by using sensor.
- To design a system that able to control the power usage via a mobile phone.
- To develop a mobile-phone notification system from current reading assessment.
- To evaluate and enhance the developed system.

Implementation of the prototype

The main component for this project is the NodeMCU which is the microcontroller and act as a brain for the system. The other component used are 4-channel relay, ACS712 current sensor, light emitting diode (LED), battery, power bank, and resistor. Figure 1 shows these components.

Figure 1: Main Componet of The Project (NodeMCU, 4-Channel Relay, ACS712 Current Sensor).

The pin A0 on NodeMCU is used to read the analog sensor data from the ACS712 current sensor. The other pins D0, D1, D2, and D3 are used as the digital input pin. These digital input pin will be connected to the 4-channel relay board. The circuit connection is as shown in Figure 2.

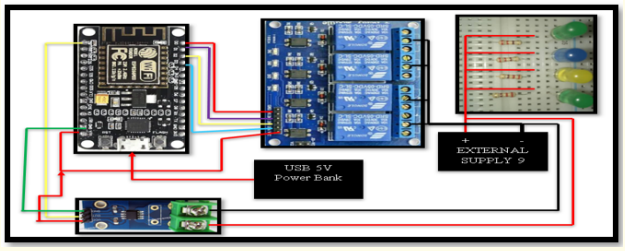


Figure 2: Circuit connection of the project.

The interface between the hardware and mobile phone is done by using Blynk application. Blynk is the application design for Internet of Things (IoT) purpose. İts process of working is illustrated in Figure 3. To program Blynk the Arduino IDE software can be used as a platform. To start using Blynk, user has to create an account which required an email address. Upon creating a new project the authentication token will be send via user email address.

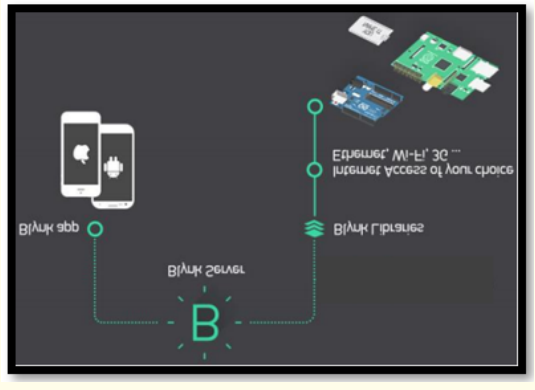


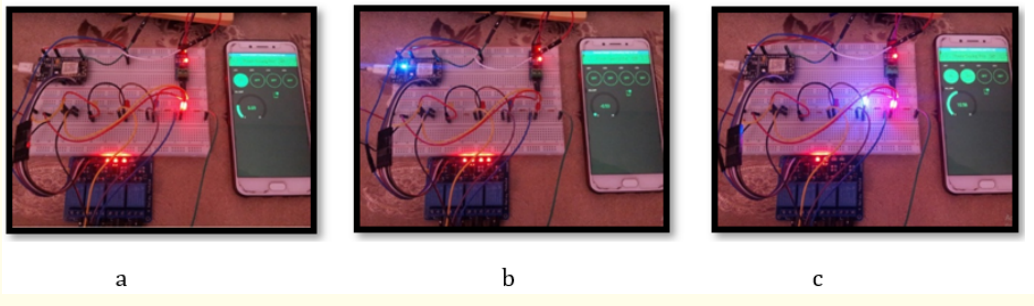
Figure 3: Main part in Blynk.

The code to able the connection between the hardware and Blynk is shown below.

```
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>     //Libraries
char auth[] = "YourAuthToken";  //Insert the auth token receive from email
char ssid[] = "YourNetworkName";
char pass[] = "YourPassword";    // WiFi Condentials
Blynk.begin(auth, ssid, pass); // activate the connection to the network, set the Auth token, and attempt to
connect to the cloud server.
Blynk.run(); // used to process the command and perform housekeeping for Blynk connection
Blynk.notify("ESP8266 Alert - High Power Usage!"); // To able push notification
```

Working principle of the system

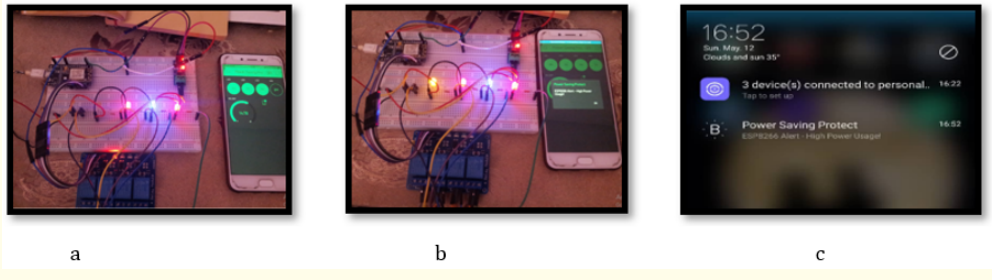
For the system to work, the NodeMCU module must be programmed by the Arduino software IDE according to appendix and make sure that the internet connection is available for the NodeMCU to be connected. Next, the NodeMCU module is powered up using the power bank that give 5V output and the relay is supplied by 9V battery to light up the LED as shown in figure 4a.



a b c

Figure 4 a: Circuit of the design. **b:** First button is pushed and 1 LED light up with 5.09 mA. **c:** Second button is pushed and 2 LED light up with 10.56 mA.

After that, the system is ready. User can control the on/off of the LED wirelessly via smartphone by using wifi connection through Blynk application. When the push button in the smartphone is clicked, the LED will light up and at the same time the meter records the current reading that flows in the system as shown in Figures 4 and 5.



a b c

Figure 5 a: Third button is pushed and 3 LED light up with 14.78 mA. **b:** Fourth button is pushed and all LED light up with 20.11 mA. **c:** Notification from the Blynk Application.

When all the LED are lighted up, the Blynk application will send a notification to user on his mobile phone stating that the power usage is high as shown in figure 5c. The notification is sent when the current detected by the ACS712 is more than 20 mA. The value can be set to any value by manipulating the code depending on the design.Thus, the user has to switch off one or more of the LED to reduce the power usage.

In this prototype, each LED is supplied with 9V battery and a 2kΩ resistor. So, the current flows in each LED will be 9V/2000 Ohm = 4.5mA. The exprimental value obtain is slightly differ that is LED1 = 4.56 mA, LED2 = 5.47 mA, LED3 = 4.22 mA, and LED4 = 5.33 mA. The differences is due to the ACS712 sensor itself has a hall effect transducer device. This means that it cannot be used near a significant magnetic fields. Besides, the lack of accuracy affects the value due to the noise from the wire connection.

Conclusion

The prototype hardware system used a wifi connection to control the on/off relays of the home appliances and monitor the current usage via smartphone and able to send a push notification to user mobile phone in case of overpower. The brain for this system is the NodeMCU module that acts as a microcontroller. The 4-channel relay is used as the switch to control the on/off of the LED, where they represent the home appliances. The current is measured by ACS712 current sensor. For the recommendation, the design should be able to use in all building with the system install inside the distribution box. Next recommendation is that the system should be able to be used anywhere around the world not only in the wifi range.

Appendix

```
#define BLYNK_PRINT Serial

#include <ESP8266WiFi.h>

#include <BlynkSimpleEsp8266.h>

#include <SimpleTimer.h>

// You should get Auth Token in the Blynk App.

// Go to the Project Settings (nut icon).

char auth[] = "c05ec1701da9478a9e73cae935761fcd";

// Your WiFi credentials.

// Set password to "" for open networks.

char ssid[] = "CPH1715";

char pass[] = "123456789";

const int analogIn = A0;

int mVperAmp = 185; // use 100 for 20A Module and 66 for 30A Module

int RawValue= 0;

int ACSoffset = 2500;

double Voltage = 0;

double Amps = 0;

SimpleTimer timer;

void ACS712()

{

RawValue = analogRead(analogIn);

Voltage = (RawValue / 1024.0) * 5000; // Gets you mV

Amps=((Voltage - ACSoffset) / mVperAmp) ;

40

Serial.print("Raw Value = " ); // shows pre-scaled value

Serial.print(RawValue);

Serial.print("\t mV = "); // shows the voltage measured

Serial.print(Voltage,3); // the '3' after voltage allows you to display 3 digits after

decimal point

Serial.print("\t Amps = "); // shows the voltage measured

Serial.println(Amps,3); // the '3' after voltage allows you to display 3 digits after

decimal point

Blynk.virtualWrite(V6, Amps);

pinMode(analogIn, INPUT);

if(Amps >20)

{

Blynk.email("rahimkifli@gmail.com", "ESP8266 Alert", "High Power Usage!");

Blynk.notify("ESP8266 Alert - High Power Usage!");

}

}

void setup()

{
```

```
// Debug console
Serial.begin(9600);
Blynk.begin(auth, ssid, pass);
// You can also specify server:
//Blynk.begin(auth, ssid, pass, "blynk-cloud.com", 80);
//Blynk.begin(auth, ssid, pass, IPAddress(192,168,1,100), 8080);
timer.setInterval(1000, ACS712);
41
}
void loop()
{
Blynk.run();
timer.run();
}
```

Bibliography

1. Nicols B. "Controlling Home and Office Appliances with Smart Phones". *The Smart Phone* (2006): 60-67.
2. Webb W. "The future of mobile phones: A remote control for your life" (2007).
3. Nader Barsoum and Karen Lee. "Simulation of Overcurrent Relay in 33kV Power Grid Protection". *Journal of Electrical and Electronic Engineering* 5 (2018): 54-63.