Load Balancing Using Load Sharing Technique In Distribution System

[1]G.JAYA LAXMI, [2]T.MOHAN SAI, [3]S.AKSHAY KUMAR, [4]D.KALYAN BABU, [5]V.DURGA RAO

[1]Assistant Professor, [2,3,4,5]U.G. Students
Department of Electrical & Electronics Engineering
Usha Rama College of Engineering & Technology
Krishna District, Andhra Pradesh, India.

Abstract- **The purpose of this project is to monitor the current levels in two different electrical grids using an Arduino-based system that interfaces with a temperature sensor and two current sensors. These sensor values are uploaded to a web server via the NodeMCU module in order to provide data logging and monitoring. The primary function of the system is to collect data from various sensors using a microcontroller, like the Arduino. Temperature sensors are used to track the outside air temperature, which is important information to know when it comes to how temperature affects electrical performance. Two current sensors are also used to keep an eye on the current levels in two separate power lines or electrical grids.**

Keywords: **Arduino Uno (At Mega 328), LCD 16*2 Display, PZEM module, NodeMCU, Current Sensor.**

INTRODUCTION:

In this novel project, we present a powerful Arduino-based system intended for the all-encompassing control and monitoring of two different electricity grids. The system uses a temperature sensor, two current sensors, and a microcontroller to deliver real-time information on the electrical networks' current levels and ambient temperature. A key component is a relay module-enabled dynamic load-sharing technology that guarantees effective distribution of electrical loads between the two grids. In order to facilitate the smooth transfer of gathered data to a web server for remote monitoring and data logging, the NodeMCU module serves as the gateway. This integrated solution meets the requirement for load optimization and advances our knowledge of how temperature affects electrical performance, ultimately promoting the safe and effective operation of electrical systems.

Proposed System:

The proposed system introduces a modern and automated approach to address the limitations of conventional methods. It involves the integration of Arduino microcontrollers, current sensors, a temperature sensor, and a NodeMCU module for web connectivity. The procedure begins with the Arduino continuously monitoring the current levels in two separate electrical grids using current sensors. Simultaneously, it also collects temperature data from the environment using a temperature sensor.

When the system detects a significant increase in current beyond a predefined threshold in one of the grids, it triggers a relay module to connect the load to the other grid. This load-sharing mechanism ensures that electrical loads are evenly distributed between the two grids, preventing overloads and minimizing the risk of electrical faults. The Arduino then sends this data, including current readings and temperature, to a web server via the NodeMCU module. Users can access the web-based dashboard or application to monitor the real-time data, enabling them to stay informed about the electrical performance of the grids and environmental conditions. They can also remotely control the load-sharing mechanism based on the data received, ensuring efficient and safe operation without the need for manual intervention. Overall, the proposed system offers a more efficient and automated solution compared to conventional methods, addressing drawbacks such as manual monitoring, delayed response to grid imbalances, and the lack of load-sharing capabilities. It enhances the reliability and safety of electrical systems by providing real-time data and control options, making it a valuable upgrade for modern electrical grid management.

Block Diagram:

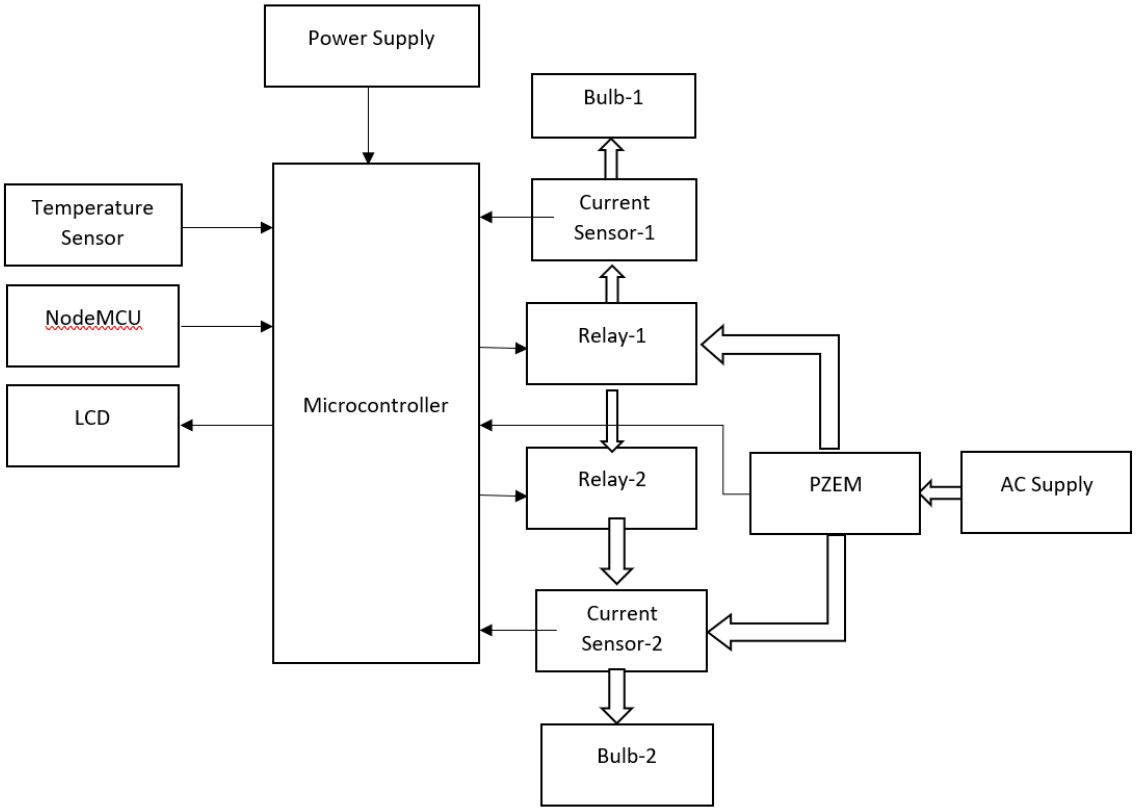

Figure 1: Block Diagram of Load Balancing Using Load Sharing Technique In Distribution System

Components:

(i) ARDUINO UNO

Arduino is an open-source hardware and software platform that simplifies the process of creating interactive electronic projects. It provides a range of microcontroller boards and a development environment for writing, compiling, and uploading code to these boards**.** Its simple programming language, extensive community support, and vast ecosystem of add-on modules (shields) make it accessible for beginners while also offering enough flexibility for advanced projects.

(ii) 16*2 LCD DISPLAY

A 16x2 LCD display is a liquid crystal display with a grid of 16 columns and 2 rows of characters. Each character slot can display alphanumeric characters, symbols, or custom characters. It's commonly used to output text-based information in electronic devices such as microcontroller-based projects.

(iii) CURRENT SENSOR

Current flowing through a conductor causes a voltage drop. The relation between current and voltage is given by Ohm's law. In electronic devices, an increase in the amount of current above its requirement leads to overload and can damage the device. Measurement of current is necessary for the proper working of devices. Measurement of voltage is Passive task and it can be done without affecting the system. Whereas measurement of current is an Intrusive task which cannot be detected directly as voltage.

(iv) RELAY

Current flowing through a conductor causes a voltage drop. The relation between current and voltage is given by Ohm's law. In electronic devices, an increase in the amount of current above its requirement leads to overload and can damage the device. Measurement of current is necessary for the proper working of devices. Measurement of voltage is Passive task and it can be done without affecting the system. Whereas measurement of current is an Intrusive task which cannot be detected directly as voltage.

(v) PZEM-004T MODULE

PZEM-004T is an electronic module that functions to measure: Voltage, Current, Power, Frequency, Energy and Power Factors. With the completeness of these functions / features, the PZEM-004T module is ideal for use as a project or experiment for measuring power on an electrical network such as a house or building. The PZEM-004T module is produced by a company called Peacefair, there are 10 Ampere and 100 Ampere models. Please be careful because the

wiring between the 10 Ampere models with 100 Amperes is different, if a short circuit or a short circuit can occur in the electrical network.

(vi) DALLAS TEMPERATURE

The digital temperature sensor like DS18B20 follows single wire protocol and it can be used to measure temperature in the range of -67oF to +257oF or -55oC to +125oC with +-5% accuracy. The range of received data from the 1-wire can range from 9-bit to 12-bit. Because, this sensor follows the single wire protocol, and the controlling of this can be done through an only pin of Microcontroller. This is an advanced level protocol, where each sensor can be set with a 64-bit serial code which aids to control numerous sensors using a single pin of the microcontroller. This article discusses an overview of a DS18B20 temperature sensor

(vii) TRANSFORMER

A transformer is a static electrical gadget that exchanges control between at least two circuits. A fluctuating current creates a changing attractive motion in one transformer curl, which thus actuates a differing electromotive power over a second loop twisted around a similar center. Without a metallic association between the two circuits, electrical vitality can be exchanged between the two loops. The enlistment law of Faraday found in 1831 portrayed the impact of prompted voltage in any curl because of the changing attractive flux surrounded by the coil.

(viii) NODE MCU

Introduction to NodeMCU Node MCU is an open-source firmware and development kit that plays a vital role in designing your own IoT product using a few Lua script lines. Multiple GPIO pins on the board allow you to connect the board with other peripherals and are capable of generating PWM, I2C, SPI, and UART serial communications. The interface of the module is mainly divided into two parts including both Firmware and Hardware where former runs on the ESP8266 Wi-Fi SoC and later is based on the ESP-12 module. The firmware is based on Lua – A scripting language that is easy to learn, giving a simple programming environment layered with a fast scripting language that connects you with a well-known developer community. And open source firmware gives you the flexibility to edit, modify and rebuilt the existing module and keep changing the entire interface until you succeed in optimizing the module as per your requirements.

PROJECT MODEL

Figure 2: Hardware Kit of Load Balancing Using Load Sharing Technique In Distribution System

CODE

```
#include <ESP8266WiFi.h>
String apiKey = "3XFUUF44Y6XZCE3S";  // Enter your ThingSpeak API key
const char* ssid = "project";   // Replace with your Wi-Fi SSID
const char* pass = "123456789"; // Replace with your Wi-Fi password
const char* server = "api.thingspeak.com";
WiFiClient client;
void setup() {
  Serial.begin(9600);
  delay(10);
  Serial.println("Connecting to Wi-Fi...");
  WiFi.begin(ssid, pass);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println();
  Serial.println("WiFi connected");
}
void loop() {
  if (Serial.available() > 0) {
    String data = Serial.readStringUntil('\n');
    Serial.println("Received data: " + data);
    int indexA = data.indexOf("a") + 1;
    int indexB = data.indexOf("b") + 1;
    int indexC = data.indexOf("c") + 1;
    int indexD = data.indexOf("d") + 1;
    int indexE = data.indexOf("e") + 1;
    int indexF = data.indexOf("f") + 1;
    int indexG = data.indexOf("g") + 1;
    String valueA = data.substring(indexA, indexB - 1);
    String valueB = data.substring(indexB, indexC - 1);
    String valueC = data.substring(indexC, indexD - 1);
    String valueD = data.substring(indexD, indexE - 1);
    String valueE = data.substring(indexE, indexF - 1);
    String valueF = data.substring(indexF, indexG - 1);
    String valueG = data.substring(indexG);
    Serial.println("Value a: " + valueA);
    Serial.println("Value b: " + valueB);
    Serial.println("Value c: " + valueC);
    Serial.println("Value d: " + valueD);
    Serial.println("Value e: " + valueE);
    Serial.println("Value f: " + valueF);
    if (client.connect(server, 80)) {
      String postStr = "api_key=" + apiKey;
      postStr += "&field1=" + valueA;
      postStr += "&field2=" + valueB;
      postStr += "&field3=" + valueC;
      postStr += "&field4=" + valueD;
      postStr += "&field5=" + valueE;
      postStr += "&field6=" + valueF;
      client.println("POST /update HTTP/1.1");
      client.println("Host: api.thingspeak.com");
      client.println("Connection: close");
      client.println("Content-Type: application/x-www-form-urlencoded");
      client.print("Content-Length: ");
```

```
    client.println(postStr.length());
    client.println();
    client.println(postStr);
    delay(1000); // Allow time for the server to process the request
    if (client.connected()) {
      Serial.println("Data sent to ThingSpeak successfully");
      client.stop();
    } else {
      Serial.println("Failed to send data to ThingSpeak");
    }
  } else {
    Serial.println("Failed to connect to ThingSpeak server");
  }
  delay(15000); // Minimum delay between updates for ThingSpeak
 }
}
```

RESULTS

Figure 3

Figure 4: Voltage display

Figure 5: Load Balance Display

CONCLUSION

In conclusion, the proposed system presents a innovative and automated solution to overcome the limitations associated with traditional methods of electrical grid management. By leveraging Arduino microcontrollers, current sensors, a temperature sensor, and a NodeMCU module for web connectivity, the system establishes a robust framework for continuous monitoring and efficient load-sharing between two electrical grids.The integration of current sensors allows the system to detect abnormal increases in current levels, triggering a load-sharing mechanism through a relay module. This proactive approach helps prevent overloads and mitigates the risk of electrical faults, thereby enhancing the overall reliability and safety of the electrical grids.

REFERENCES:

1] Sam Weckx; Johan Driesen .Load balancing with EV chargers and PV inverters in unbalanced distribution grids,2015, 1949-3029

[2]Leszek S. Czarnecki; Paul M Haley; Unbalanced power in four wire system and its reactive compensation, 2014, 0885-8977.

[3] Raghvendra Singh; Dharmendra Singh ,Three phase load balancing and harmonic reduction using distribution static compensator, 2015, 2250-3153.

[4] Willy Siti; Adisa Jimoh; Dan Nicolae; Phase load balancing in the secondary distribution network using a fuzzy logic and a combinatorial optimization based on Newton Raphson, 2009, 92-100.