Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 12, No.2, December : 2024

IOT BASED SMART CITY WITH ELECTRIC POWER GRID AND RING MAINS SYSTEM

Gyanesh Singh School of Electrical Engineering, Shri Mata Vaishno Devi University Katra Jammu and Kashmir India (gyan9000@gmail.com)
Manish Sabraj, School of Electronics and communication Engineering, Shri Mata Vaishno Devi University Katra Jammu and Kashmir India

Abstract –
In this paper, firstly authors have developed a grid with ring mains which solves the problem of power outage. Secondly, we made smart street lights which resolves the issue if power wastage and conserves electricity. Thirdly, we have designed blind turn/hill road curve accident prevention system which solves the problem of accidents/ fatalities/vehicle collisions at blind turns of a road/ mountains. Fourthly, we have built a density-based traffic control system which helps in saving of time from long hour's traffic. Fifthly, in electric power grid and ring mains distribution system, we have shown the working of components installed at substation and also gives a demonstration of handling the fault/s at any point in transmission/ distribution lines without effecting the flow of electricity in the rest of the transmission, distribution systems and loads connected. All these features make our city not only smart but also safe and efficient in every term.

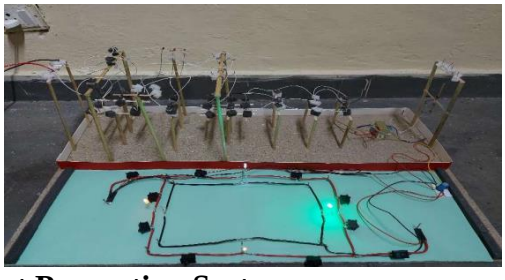
Keywords –
Smart city, aurdino, IOT, Python, Power system grid

1. INTRODUCTION

Since with the emerging developments and modern advancements in the field of science and technology, there is now a need/requirement to focus on the urban/rural sectors too so as to make them technological modern by using different types of electronic methods and sensors for collecting specific data and then analyze the data to manage resources, assets and services efficiently which as a results could be used to improve operations in the city. The uprising growth of a city/nation towards modern developments and better livelihoods of citizens in different sectors such as banking, security, shopping, parking, governance, infrastructure, education, building, health, mobility, technology, energy collectively makes a smart city. We also got inspired by the initiative of building smart cities that aims to optimize the city functions and promote economic growth while also improving the quality of life for citizens by using smart technologies and data analysis and decided to develop a smart city which is not only smart just by having some certain AI, IoT applications but also represents as a safe and efficient smart city in terms of electrical energy sector too.

1.1 Electric Power Grid with Ring Mains System:

This section comprises of electric power grid, substations/feeders and ring mains distribution system with loads. This portion shows the components installed at grids along with their working, the step-down power transmission and the ring mains distribution system with connected loads which resolves the issue of power outage even when there would be fault or maintenance work doing on in the in the system.



1.2 Smart Roads with Accident Prevention System:

This section comprises of smart street lights system, density-based traffic control system & blind turn/hill road curve accident prevention system. This portion shows the different useful features on smart city road by using Arduino UNO such as; blind curve indicators, street lights which only glows when some object gets detected and traffic light signals along with go slow indicator which changes per the density on road.

1. Analysis and Coding

The software platform used: Arduino UNO

2.1 Density based Traffic Control System:

```
int R1 = 12;
int O1 = 11;
int G1 = 10;
int R2 = 9;
int O2 = 8;
int G2 = 7;
int Blue1=4;
int Blue2 =5;
int sensor1 = 3;
int sensor2 = 2;
int L =13;
void setup() {
pinMode(R1, OUTPUT);
pinMode(O1, OUTPUT);
pinMode(G1, OUTPUT);
pinMode(R2, OUTPUT);
pinMode(O2, OUTPUT);
pinMode(G2, OUTPUT);
pinMode(Blue1, OUTPUT);
pinMode(Blue2, OUTPUT);
pinMode(L, OUTPUT);
pinMode(sensor2, INPUT);
pinMode(sensor1, INPUT);
Serial.begin(9600);
}
void loop() {
int sensorval1 = digitalRead(sensor1);
int sensorval2 = digitalRead(sensor2);
Serial.println(sensorval1);
Serial.println(sensorval2);
if (sensorval1 == HIGH && sensorval2 == HIGH) {
green_light1();
red_light2();
```

```
digitalWrite(Blue2, HIGH);
digitalWrite(Blue1, LOW);
delay(10000);
yellow_light1();
yellow_light2();
digitalWrite(Blue2, LOW);
digitalWrite(Blue1, LOW);
delay (4000);
green_light2();
red_light1();
digitalWrite(Blue1, HIGH);
digitalWrite(Blue2, LOW);
delay(10000);
digitalWrite(L, HIGH);
delay(1000);
digitalWrite(L, LOW);
}
if (sensorval1 == LOW && sensorval2 == HIGH) {
green_light2();
red_light1();
digitalWrite(Blue1, HIGH);
digitalWrite(Blue2, LOW);
delay(10000);
digitalWrite(Blue2, LOW);
digitalWrite(Blue1, LOW);
yellow_light1();
yellow_light2();
delay (4000);
green_light1();
red_light2();
digitalWrite(Blue2, HIGH);
digitalWrite(Blue1, LOW);
delay(10000);
digitalWrite(L, HIGH);
delay(1000);
digitalWrite(L, LOW);
}
if (sensorval1== HIGH && sensorval2 == LOW) {
green_light1();
red_light2();
digitalWrite(Blue2, HIGH);
digitalWrite(Blue1, LOW);
delay(10000);
yellow_light1();
digitalWrite(Blue1, LOW);
digitalWrite(Blue2, LOW);
yellow_light2();
delay (4000);
```

```
green_light2();
red_light1();
digitalWrite(Blue1, HIGH);
digitalWrite(Blue2, LOW);
delay(10000);
digitalWrite(L, HIGH);
delay(1000);
digitalWrite(L, LOW);
}
else {
green_light1();
red_light2();
digitalWrite(Blue2, HIGH);
digitalWrite(Blue1, LOW);
delay(10000);
yellow_light1();
yellow_light2();
digitalWrite(Blue2, LOW);
digitalWrite(Blue1, LOW);
delay (4000);
green_light2();
red_light1();
digitalWrite(Blue1, HIGH);
digitalWrite(Blue2, LOW);
delay(10000);
digitalWrite(L, HIGH);
delay(1000);
digitalWrite(L, LOW);
}
}
void green_light1()
{
digitalWrite(G1, HIGH);
digitalWrite(O1, LOW);
digitalWrite(R1, LOW);
}
void yellow_light1()
{
digitalWrite(G1, LOW);
digitalWrite(O1, HIGH);
digitalWrite(R1, LOW);
}
void red_light1()
{
digitalWrite(G1, LOW);
digitalWrite(O1, LOW);
digitalWrite(R1, HIGH);
}
```

```
void green_light2()
{
digitalWrite(G2, HIGH);
digitalWrite(O2, LOW);
digitalWrite(R2, LOW);
}
void yellow_light2()
{
digitalWrite(G2, LOW);
digitalWrite(O2, HIGH);
digitalWrite(R2, LOW);
}
void red_light2()
{
digitalWrite(G2, LOW);
digitalWrite(O2, LOW);
digitalWrite(R2, HIGH);
}
```

2.2 Smart Automatic Street Light:

```
int read1 = 2;
int read2 = 3;
int led = 12;
int led1 = 11;
int led2 = 10;
int led3 = 9;
void setup() {
Serial.begin(115200);
pinMode(read1,INPUT);
pinMode(read2,INPUT);
pinMode(led, OUTPUT);
pinMode(led1, OUTPUT);
pinMode(led2, OUTPUT);
pinMode(led3, OUTPUT);
}
void loop()
{
if(digitalRead(read1)== HIGH && digitalRead(read2) == HIGH)
{
digitalWrite(led,HIGH);
digitalWrite(led1,HIGH);
digitalWrite(led2,HIGH);
digitalWrite(led3,HIGH);
delay(1000);
}
if (digitalRead(read1)== HIGH)
{
digitalWrite(led,HIGH);
digitalWrite(led1,HIGH);
```

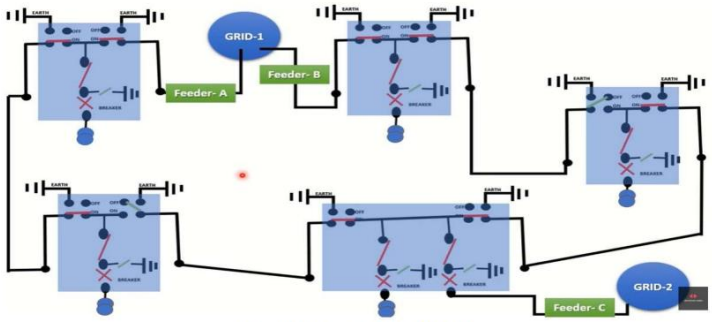
```
}
else if(digitalRead(read2)==HIGH)
{
digitalWrite(led2,HIGH);
digitalWrite(led3,HIGH);
}
else {
digitalWrite(led,LOW);
digitalWrite(led1,LOW);
digitalWrite(led2,LOW);
digitalWrite(led3,LOW);
}
}
```

2.3 Blind Curve Indication and Accident Prevention System

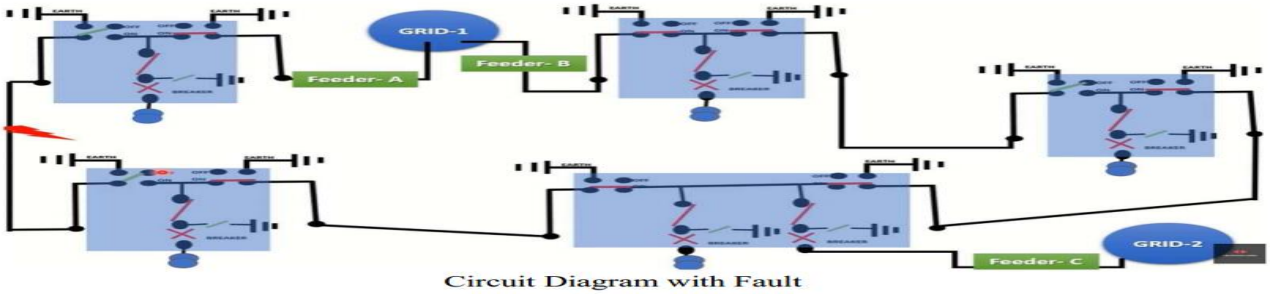
```
int read1 = 2;
int read2 = 3;
int led = 12;
int led1 = 11;
int led2 = 10;
int led3 = 9;
void setup() {
Serial.begin(115200);
pinMode(read1,INPUT);
pinMode(read2,INPUT);
pinMode(led, OUTPUT);
pinMode(led1, OUTPUT);
pinMode(led2, OUTPUT);
pinMode(led3, OUTPUT);
}
void loop()
{
if (digitalRead(read1)== LOW && digitalRead(read2) == LOW)
{
digitalWrite(led,HIGH);
digitalWrite(led1,HIGH);
digitalWrite(led2,HIGH);
digitalWrite(led3,HIGH);
delay(3000);
digitalWrite(led,LOW);
digitalWrite(led1,LOW);
digitalWrite(led2,LOW);
digitalWrite(led3,LOW);
}
else if (digitalRead(read1)== LOW)
{
digitalWrite(led,HIGH);
delay(1000);
digitalWrite(led1,HIGH);
```

```
digitalWrite(led,HIGH);
delay(1000);
digitalWrite(led,LOW);
digitalWrite(led1,HIGH);
delay(1000);
digitalWrite(led,LOW);
digitalWrite(led1,LOW);
}
else if(digitalRead(read2)==LOW)
{
digitalWrite(led2,HIGH);
delay(1000);
digitalWrite(led2,HIGH);
digitalWrite(led3,HIGH);
delay(1000);
digitalWrite(led2,LOW);
digitalWrite(led3,HIGH);
delay(1000);
digitalWrite(led2,LOW);
digitalWrite(led3,LOW);
}
else {
digitalWrite(led,LOW);
digitalWrite(led1,LOW);
digitalWrite(led2,LOW);
digitalWrite(led3,LOW);
}
}
```

2. Results



Circuit Diagram without Fault



Circuit Diagram with Fault

REFERENCE

[1] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of Things for smart cities," IEEE Internet of Things Journal, vol. 1, no. 1, pp. 22–32, 2014.
[2] H. Zhang, M. Babar, M. U. Tariq, M. A. Jan, V. G. Menon, and X. Li, "SafeCity: toward safe and secured data management design for IoT-enabled smart city planning," IEEE Access, vol. 8, pp. 145256-1–14145267, 2020.
[3] R. Mageshvaran, I. J. Raglend, V. Yuvaraj, P.G Rizwankhan, T. Vijayakumar, Sudheera ―Implementation of Non-Traditional Optimization Techniques for the Optimal Load Flow Solution‖ IEEE Region 10 Conference, TENCON 2008.
[4] O. L. Elgerd, ―Electric Energy Systems Theory: An Introduction‖, 2nd Edition, Mc-Graw-Hill 2012.
[5] I. J. Nagrath, D. P. Kothari, ―Modern Power System Analysis ‖, 3rd Edition, New York, 2007.
[6] H.W. Hale, R.W. Goodrich, ―Digital Computation or Power Flow—Some New Aspects. Power Apparatus and Systems‖, Transactions of the American Institute of Electrical Engineers, 78, 919-923, 1959.