Monitoring the Operation of Transmission Line in a Smart Grid System through IoT

Fatima Alhebshi
ECE Department
Effat University
Jeddah, KSA
Faalhebshi@effat.edu.sa

Heba Alnabilsi
ECE Department
Effat University
Jeddah, KSA
Healnabilsi@effat.edu.sa

Jauhra Alzebaidi
ECE Department
Effat University
Jeddah, KSA
Jaalzebaidi@effat.edu.sa

Ahmed Bensenouci, SIEEE
ECE Department
Effat University
Jeddah, KSA
abensenouci@effatuniversity.edu.sa

Tayeb Brahimi
ECE Department
Effat University
Jeddah, KSA
tbrahimi@effatuniversity.edu.sa

Mohamed-Amine Bensenouci
Faculty of Electronics and Computer Science
Houari Boumediene University of Science and Technology
mbensenouci@usthb.dz

Abstract — **Electrical transmission lines are generally exposed to harsh natural environment and leads to different types of failures including thundering, rain, lightning phenomena, flashover, and overheating. The Internet of Things (IoT) represents a step beyond the utilization of conventional SCADA by increasing the use of wireless technology and provides real-time data acquisition.**

The objective of this paper is to design an Arduino connected to different sensors for monitoring and controlling transmission system problems. These sensors include PIR (Pyroelectric Infra-Red) sensor, relay for control, and wireless sensors for conductor galloping due to the wind, meteorology, and temperature. The IoT is applied by simulating circuit with different sensors to monitor and control possible technical failures.

Keywords—Transmission line, IoT, Smart grid, SCADA.

I. INTRODUCTION

The power system is one of the most primitive fields that have technologically advanced within Electrical Engineering. A traditional power system deals with three systems: generation, transmission, and distribution systems of electric power. It is called “traditional”, because the generation of electricity is centralized, which means that only a few large power plants are involved. Working on the generation system, the power plants will transform other forms of energy into electric power. These sources of power contain fossil fuels such as coal and natural gas, hydro, nuclear, solar, and wind power. Furthermore, the transmission system is made up of transmission lines that are in control of passing power from the power station where the power is transferred to the location of the consumers. The distribution system is the network that supplies power to the load that can be consumed by the user's apparatus [1-2].

IoT is the network of physical substances that include entrenched technology to connect and sense or work together with their interior circumstances or the outer environment. The Internet of Things was "Born" in 2008-2009. By the year 2013, the IoT had progressed into a system by many technologies, going from the internet to wireless communication and from Micro-Electromechanical Systems (MEMS) to established systems. The IoT is supported by traditional fields, and wireless sensor networks, GPS, control systems, and others. The architecture layers of IoT are: the object layer, made of sensors and smart devices, the communication layer deals with latency, error probability, scalability, bandwidth, and security, and the application layer that is grouped based on the type of network, coverage, size, heterogeneity, business model, and real-time or non-real-time requirement.

By 2020, the internet will be connected with about 30-50 billion appliances. Thus, the IoT retained the third revolution in the digital technology after the computer and Internet [3].

IoT gets important benefits to the smart grid between other systems. Disaster prediction and prevention of power-lines outages are the most challenging problems for electricity transmission for lots of reasons. For example, analogue collection of the data being generated at remote areas is difficult, but, when using IoT for data acquisition, it becomes just a data gathering and system monitoring and controlling, which is easier. Progressive sensing and communication technologies of the Internet of Things can efficiently avoid or minimize the damage of natural disasters confronted by the power-lines, and hence develop the reliability and stability of power transmission [4]. The objective of this paper is to design an Arduino connected to different sensors for monitoring and controlling transmission system problems. After a brief introduction of IoT, we will call attention to the most important reasons of failure of transmission lines and then we will study a three-layer architecture model of the Internet of things for

978-1-5386-4817-9/18/$31.00 ©2018 IEEE

smart grid in addition to an overview of IoT based power-lines online monitoring system. The IoT will be applied by simulating circuit with specific types of sensors to monitor and control the technical failures.

II. TRANSMISSION LINE

A transmission line is a long conductor with a specific design to transmit the bulk amount of generated power of very high voltage from one station to another as per variation of the voltage level [5]. First, the types of transmission line depending on their length are: short transmission line which is about 50km and its voltage level is up to 20kV, the medium transmission line which is from 50km to 150km and its voltage level between 20kV and 100kV, and long transmission line where its length is more than 150km and its voltage is above 100kV. Second, the main practical types of transmission line depending on the conductors are a balanced two-wire line, coaxial cable, waveguide, microstrip, planer line, fiber optic [6]. Transmission line systems are facing many technical failures such as a blackout or short circuit and it is difficult to collect, count, and monitor them especially in remote areas. These technical failures can be monitored or solved by a smart grid system through IoT. In this regards, IoT will be applied by designing an Arduino UNO board connected to different sensors. These sensors will send signals to the Arduino, which in turn, will communicate this information to the database center and notify the technicians about the type of problem. With the use of IoT, the efficiency and performance of the power grid and operational efficiency can be improved significantly.

III. TRANSMISSION LINES PROBLEMS (OUTAGES)

The transmission lines are generally exposed in the air to the harsh environment of nature which leads to different types of failures including thundering, rain, lightning phenomena, flashover, and overheating. There are several problems relates to The Line. When a transmission line sags extremely, it causes power failure. An overheating electrical transmission line sagging into a tree sparked the greatest power failure in the Western United States in 1996 [7]. As the temperature increases, the outspread conductor length will increase and this will cause an increase in sag and decrease in conductor tension. Transmission lines that sag under heavy use and high temperatures are the devastations of transmission line operators. These lines are not only restricted in the volume of power they can transmit, but sagging lines run a higher risk of getting in touch with trees and shorting out. A sagging transmission line in Ohio is believed to be the event that triggered 2003 blackout [8]. Traditionally, workers in power line will cut the power off as soon as they discover this problem in order to avoid fire incident, and then they will send the specialist to define and solve the problem which takes a lot of time and effort.

In the experiment, this problem will be applied through IoT by using the DTH11 sensor and a heating source (lighter). By exposing the sensor to heat and monitor it to see if sag limits are breached and if breached a notification will be sent to the user's mobile by the Arduino board, and then the user will use the application to control the cooling system by using a fan in the simulating circuit to solve the overheating problem.

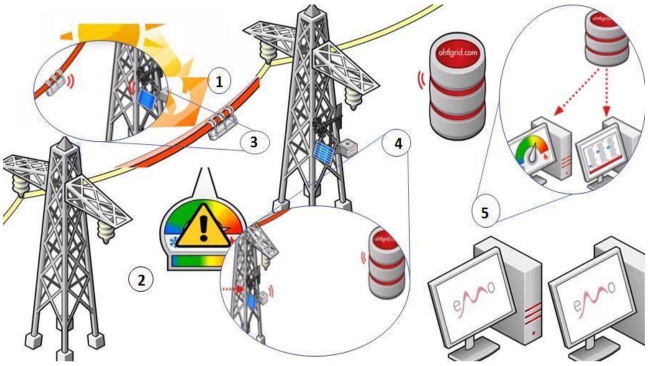


Figure 1: Processing of technical failure in a transmission line tower (1. Wire stretched due to heat, 2. Technical failure warning. 3. Temperature, sensor. 4. Receiver and charger. 5. DB center and monitoring.)

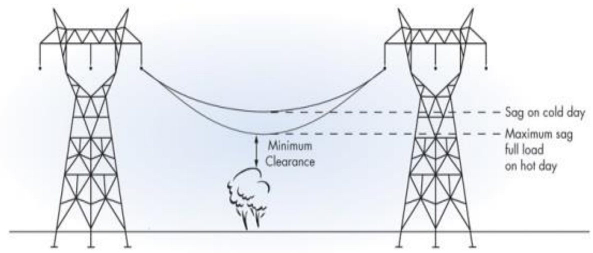


Figure 2: Sag and Safety Limit

The air nearby a power line isn't a good conductor, but very high voltages do make a considerable electrical field. For large-scale transmission lines, this field can have a radius of a foot or more. That means electricity could arc out of the wire to any metal that gets close enough, even if it never makes contact. Most power companies alert workers to stay 10 feet away from power lines and up to 25 feet away from the highest-voltage lines. Even regular people clipping trees close a power line must take care, wood isn't as conductive as metal, but a lost branch can still conduct a deadly shock down the tree trunk [9].

In the experiment, this problem will be applied through IoT by using the PIR motion sensor (3m-7m) and a 5mm LED. The distance from the power lines will be assumed to be 7 meters away. If an object exceeds the distance limit the PIR motion sensor will sense the movement and then the Arduino board will send a notification to the user's mobile to inform about the problem and then the user can control the LED (assuming that the LED is attached to the transmission line tower or to something close to it) through the application by turning the LED on to let the object that exceeded the limit (human) know that he is in the danger zone.

When lightning hits a power line, the flow of electricity can cause a flashover, and the suitable corrective action depends on how flashover occurred. A back flashover occurs when a lightning strike hits a shield wire or a tower structure. When this happens, current flows in both directions and down the tower into the ground, evolving a voltage on the cross arm enough to flashover the insulator string. A back flash is usually caused by big stroke currents, high tower flow impedances (tall towers) and/or high footing resistance. On high-voltage transmission lines, low-current lightning strokes, in general, causes shielding failures while the high-current strokes result in back flashovers of the tower to the phase conductors. On lower-voltage lines, this distinction is not clear [10]. Flashover occurs when the voltage across the insulators increase beyond the limits. Back flashover occurs when the lightning which has struck the tower is unable to get discharged to the earth. This can occur due to high drift resistance of the ground. It decreases power transmission lines reliability [11]. Usually to solve this problem a lightning conductor transmission line lightning over-voltage protection is selected, it is the most frequently used anti- Ray device. Also, reducing the tower grounding resistance will accomplish the aim of lightning protection [12].

In the experiment, this problem will be simulated through IoT by using the flame sensor by exposing the sensor to a flame source (fire) by natural disasters, then, when the sensor captures the waves that pop up from the spark of fire, the Arduino board will send a notification to the user's mobile to inform about the problem. By applying IoT the threat of flashover can be predicted and act what must be done before the risk exceeds a specified level.

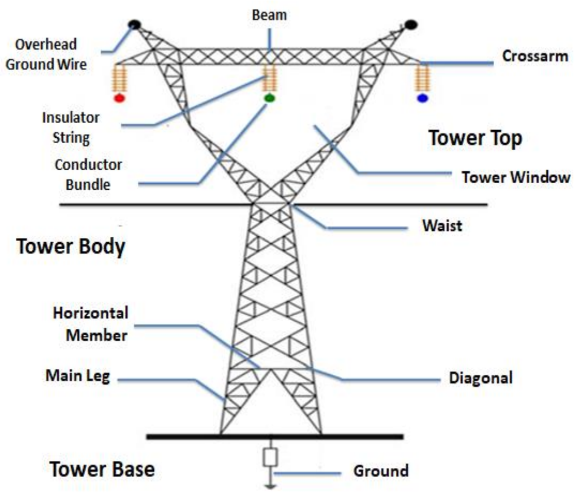


Figure 3: Transmission Line Tower

IV.ARCHITECTURE OF IOT

The architecture of IoT is expressed by three layers i) perception layer, ii) network layer, and iii) application layer, as shown in Figure 3. The perception layer contains two-dimensions, and is usually split into two sub-layers: perception communication extension and control sub-layers. The functions of the perception layer depend on sub-layers. First, the perception control is realizing smart perception of physical world together with recognition, data acquisition, processing, and automatic control with the second sub-layer which is communication extension sub-layer that is connected to the physical entities with the network layer and the application layer by the communication module. The network layer consists of all types of communication and the core network. When the information transmission, routing, and control are often implemented in the fundamental network, the communication network is looked at as the access network. The application layer provides many smart applications for certain industry. Hence, IoT technologies can be combined with all forms of the industrial undertaking. The application layer contains application infrastructure/middleware and terminal units. Through the application layer, the development of economy and society would be influenced greatly when the deep integration of IoT technology with industry is achieved [13].

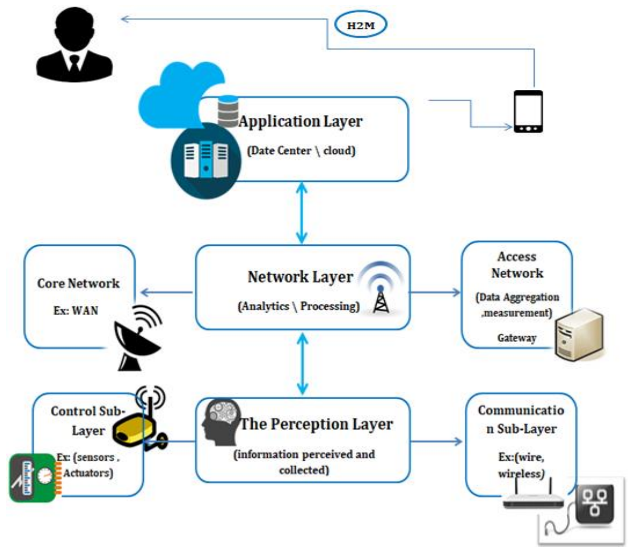


Figure 4: Architecture of IoT

V.ARCHITECTURE OF ARDUINO UNO BOARD

Arduino Uno is the latest revision of the basic Arduino USB board. It is a microcontroller board based on the ATmega328P. It has 14 digital input/output pins: 6 can be used as Pulse Width Modulation- outputs, 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP (in-circuit serial programming) header and a reset button as shown in Figure 5. It covers everything required to support the microcontroller; simply connect it to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started. It can be extended with a variety of shields: custom daughter-boards with specific features. It is similar to the Duemilanove (Another type of Arduino), but has a diverse USB-to-serial wafer the ATMega8U2, and anew designed labeling to make inputs and outputs easier to identify. The Uno varies from all other Arduino boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter. In the project, we chose the Uno because the drivers (operating systems) are easily available, and the installation takes very little time. The Arduino is shifted to using ATmega16U2 from FTDI chips for 'code burning' [14].

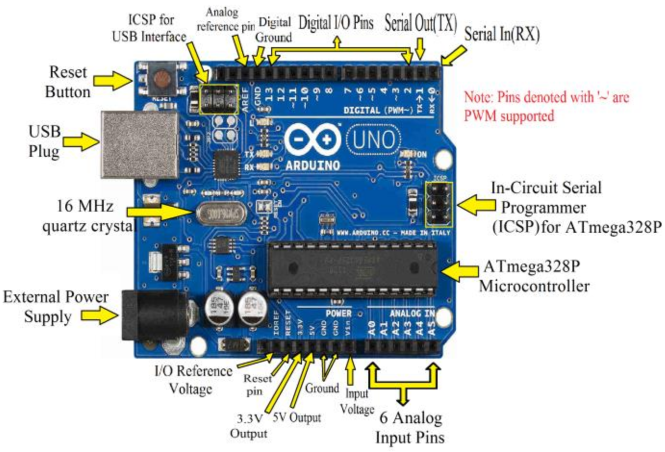


Figure 5: Architecture of Arduino UNO Board

VI.CIRCUIT COMPONENTS AND DESIGN

In simulating circuit part, the following data are the data sheets of element for our design:

A. Motion sensor (PIR)

The PIR detects the motion with the neighborhood. The distance is detected in the range of 3 to 7 meter by detecting the emitted infrared energy given off by humans and animals in the form of heat. In addition, it senses the presence of movement by an angle 110 degrees 5V-12V.

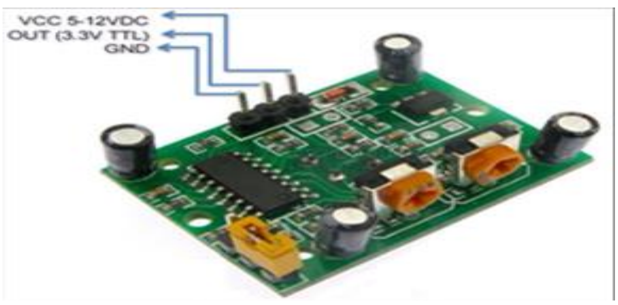


Figure 6: Motion Sensor

B. Flame Sensor

Can capture the waves which are a pop-up from a small spark of fire and control its accuracy also it covers an area of 4 meters.

Figure 7: Flame Sensor

Temperature and Humidity Sensor (DHT11):

It is a low-cost digital temperature and humidity sensor. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air, and provide a digital signal on the data pin. Properties of DHT11 sensor: humidity range (20-90% RH), humidity accuracy (±5% RH), temperature range (0-50°C), temperature accuracy (±2% °C), and the operating voltage (3V to 5.5V) [15].

Figure 8: Temperature and Humidity Sensor

C. ESP8266 processor

Is an 80 MHz microcontroller with a full Wi-Fi front-end (both as a client and access point) and TCP/IP stack with DNS support as well [14]. For general properties Of ESP8266-01 review [16].

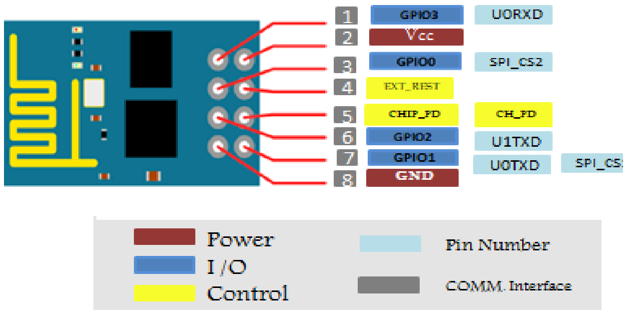


Figure 9: ESP8266 Processor

D. Female/Male Jumper Wires:

Used to connect the ESP8266 processor with the Arduino UNO board [15].

Figure 10: Female/Male Jumper Wires

E. Flexible Stranded Core Wires With Stiff Ends

These are the main development over the "box of bent wires" that are sometimes sold with breadboards, and quicker than stripping your own solid core wires. Makes breadboarding super-fast [15].

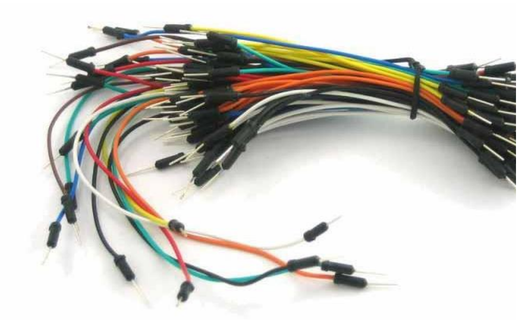


Figure 11: Flexible stranded core wires with stiff ends

F. Full-sized Breadboard

A breadboard is a solderless tool for a short-term prototype with electronics and checks circuit designs. Most electronic components in electronic circuits can be linked by placing in their leads or terminals into the holes and then creating connections through wires where appropriate. The breadboard has strips of metal underneath the board and links the holes on the top of the board. The top and bottom rows of holes are associated horizontally and divided in the middle while the remaining holes are linked vertically [17].

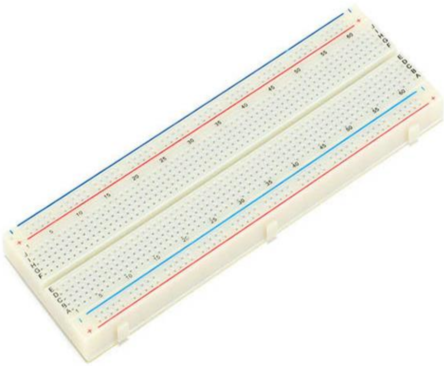


Figure 12: Full sized breadboard

An Arduino UNO board that was explained before.

Blue LED 5mm

Figure 13: 5mm LED

G. Fan

The Fan works in 12V DC. The fan's speed will be controlled by temperature and Arduino.

Figure 14: Fan

VII. Steps of Experiment

The main steps involved in the experiment will be given in this portion.

A. Server Creation

In this part, we are going to install the ESP- modules as a server and verify that there is a communication between the module and the Arduino UNO board. The components needed to program the ESP-01 module: Esp8266 Wi-Fi module first version, Breadboard, Arduino UNO, Male/Female jumper wires, and the programs that you will need are Arduino IDE and ESP8266 User Program Ver1.

Step1:

Connect the WIFI-module to Arduino board by jumper wires as shown in Figure 15. Second, to ensure that no previous programs are running, download the BareMinimum code (empty code) in Arduino IDE. Then, open the serial monitor and typing AT commands. You should get a response which means that the module is working.

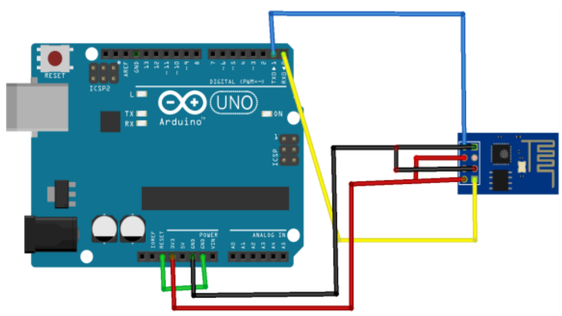


Figure 15: ESP8266 Connection While Programming

Step2:

There are several commands and operation modes to test the communication between the module and another device, and to implement that set up the module to operate by using this AT+CWMODE= (mode number). The first mode is the access point (AP=1) that make ESP-01 works as a WI-FI network, and the second Station (STA=2) means that the module can connect the access point like the WI-FI network from a building. The last mode (Both=3) is used to permit the module to act as both an AP and an STA. And all these allow the devices and the module to be connected to each other through the network to communicate.

Step 3:

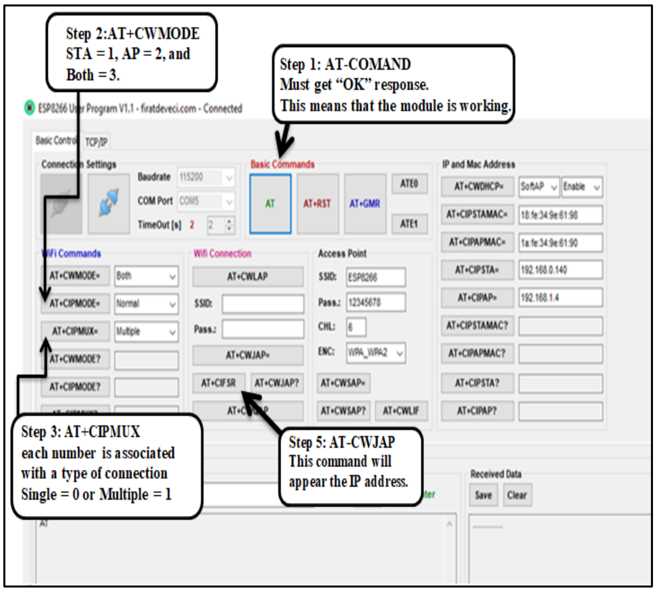
To check if the module is already connected to a Network, send the command: AT+CIFSR, this will show the station IP address of the ESP-01 module. Then, to configure the ESP8266 ESP-01 module as a server, we need to enable multiple connections before. The next step is to start the server at port 80: (AT+CIPSERVER=1, 80). The first number is used to open server mode (1). We chose port 80 because this is the default port for HTTP protocol.

Step 4:

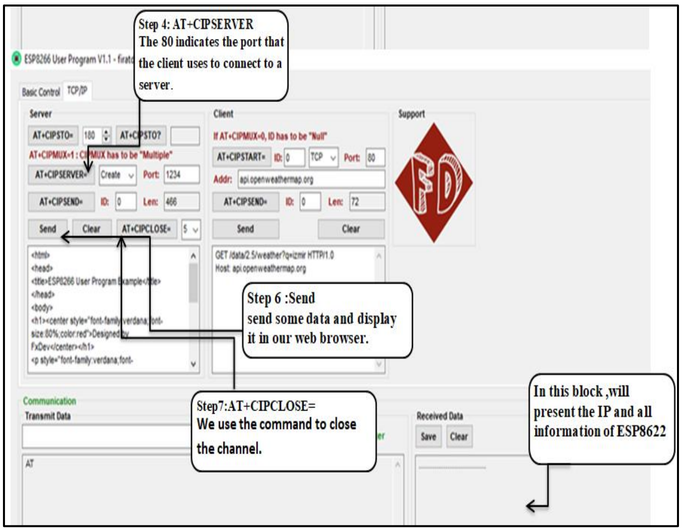
When opening a web browser, and typing the IP address of ESP module and the port of it in the browser address bar, the response shown in the image above will appear.

Step 5:

After a few seconds, a message "SEND OK" appears in serial monitor that means the data has been transmitted successfully to the client.



(a)



(b)

Figure 16: Server Creation

B. *ESP8266 Programing & Connection*

To start programming, open the Arduino IDE. Then, connect the module to the computer via Arduino board. Then, add the library of ESP8266 to the Arduino IDE program. After installing the library, select the port of the board from the tools bar change board type by selecting ESP8266. After that, select the Serial monitor and set the rate of data transfer speed to 115200 and select Both NL & CR and start writing the program. In this step, we are programming the module with Arduino to work automatically. Then, write the code in Arduino IDE, after that upload the code. Lastly, open Serial monitor to take the IP to enter to the server without connecting to the module by connecting the module to the router. The method is writing the name of network and password in the code of Arduino to connect the module to the router automatically.

For the code, from file go to examples and then ESP8266WebServer and select HelloServer and change the SSID to your Wi-Fi SSID, and also change the password.

C. *Connect the Circuit*

Keep the ESP8266 as it's connected to Figure 17, and then connect the input devices DTH11, PIR, and flame sensors. After that, connect the output devices which are the LED and fan. Refer to Figure 17 for the connection.

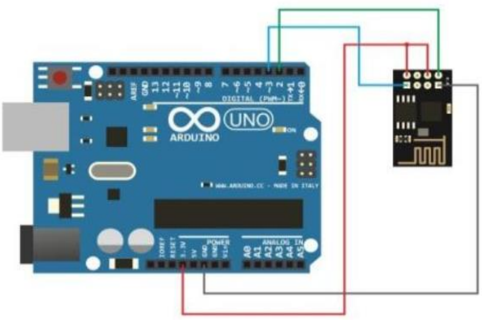


Figure 17: After Programming Connection

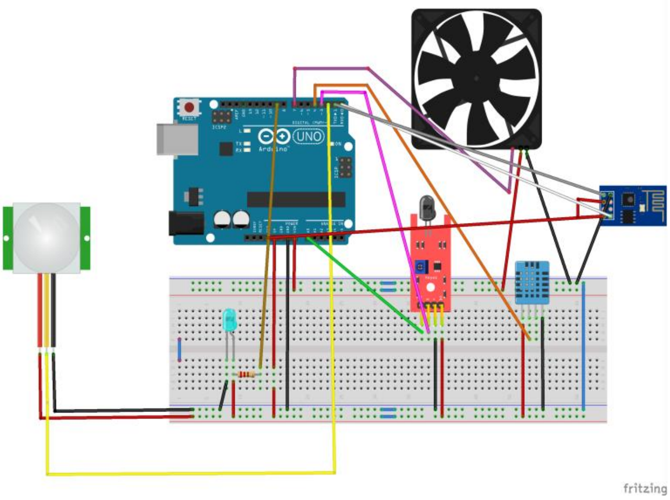


Figure 18: Simulating Circuit

D. Arduino UNO Programming

This part cares about the codes that will make the Arduino UNO board interact with the inputs (sensors), outputs (devices). It is under formation.

E. Application Schemes

We created the application to control the devices by using the app inventor to create the interface of our app like in Figure 19. To connect the application with the Arduino board just write the IP address that you have made in above steps for the ESP8266 in the application's textbox.

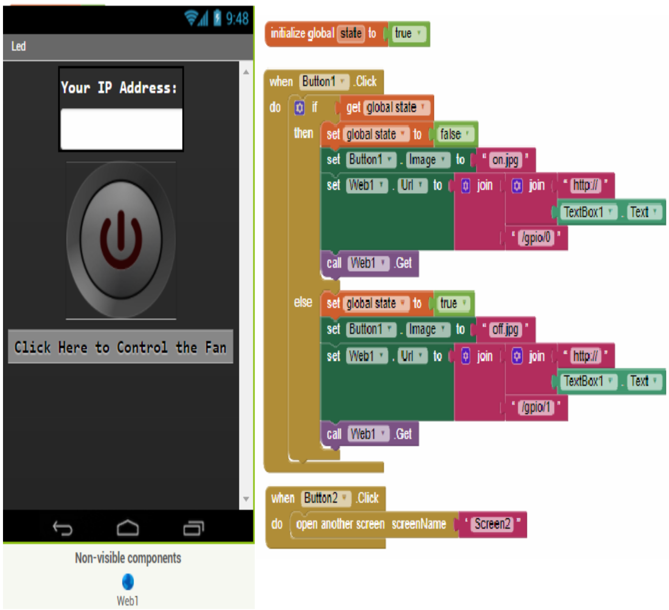


Figure 19: LED Control Code & Screen

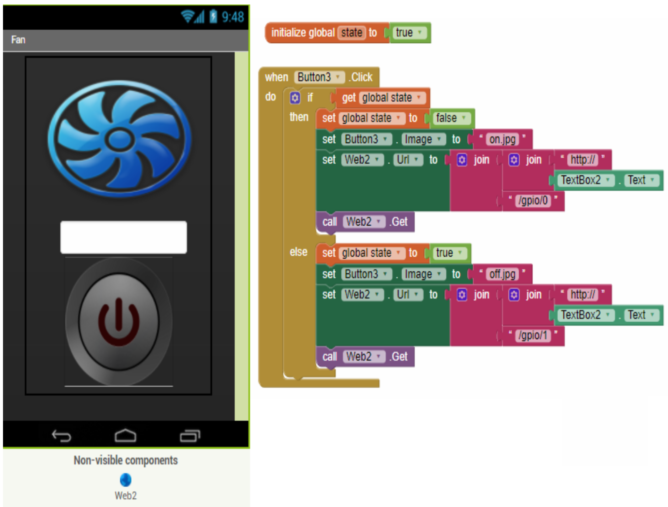


Figure 20: Fan Control Code & Screen

In this experiment, we get the results depending on the steps which we have presented and implemented. These steps led to the creation of the server and IP address like in Figure 21-22.

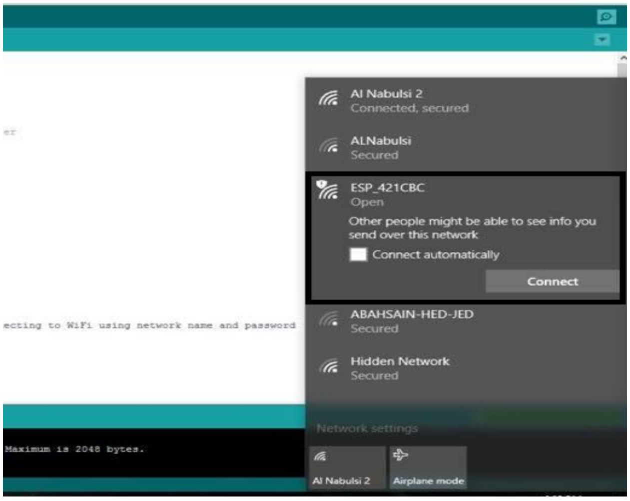


Figure 21: The access point AP

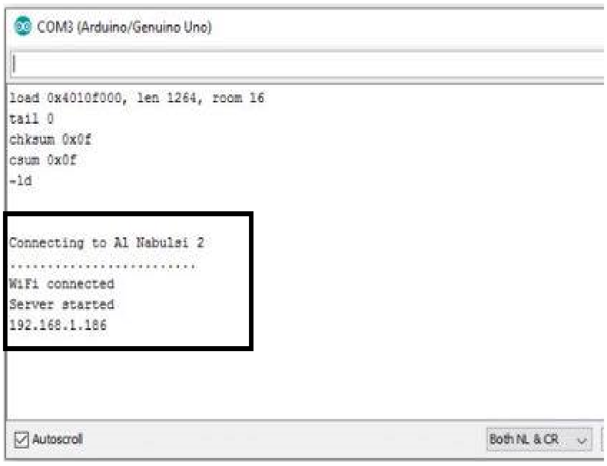


Figure 22: The SIP address

The IP address is used to connect the Arduino board with the application to continue our work. Because of the time constraint we couldn't get all the results. We expected to get a response from the Arduino board, and this response is an email that will specify the type of failure which will help the user to choose what action he should take to solve the problem including mobile application.

VII. Conclusion

In this paper, we explored the application of IoT for monitoring transmission line system by introducing which can lead to large-scale improvement in efficiency and performance.

Because we are not allowed to access any transmission line, a small circuit is created to monitor transmission line outages. The circuit includes a microcontroller (Arduino), different sensors (PIR, relay for control, sensors for conductor galloping due to wind, meteorology, and temperature), and a mobile app connected using IoT. The software and hardware parts of the design are working properly.

This work can be extended to cover many applications in power system and beyond.

References

[1] The Basics of Power & Energy Engineering. Retrieved November13, 2017, from https://www.eescholarship.org/careers-in-power energy/engineering-basics.

[2] Csanyi, E., “The structure of electric power systems,” Generation, Distribution and Transmission of Energy,” EEP. Retrieved November 20, 2017 from http://electrical-engineering-portal.com/electric-power-systems.

[3] Gartner, “Internet of Things defined - tech definitions,” February 15, 2017. Retrieved November 13, 2017, from https://www.gartner.com/it-glossary/internet-of-things.

[4] Chen, X., Sun, L., Zhu, H., Zhen, Y., & Chen, H. “Application of internet of things in power-line monitoring,” December 24, 2012. Retrieved Nov. 13, 2017 from http://ieeexplore.ieee.org/document/6385004/.

[5] “Voltage in Power Electric Lines”. Retrieved Nov. 22, 2017 from https://www.electrical4u.com/voltage-in-power-lines/.

[6] “Transmission line”. Retrieved Nov. 22, 2017 from http://www.daenotes.com/electronics/communication-system/transmission-line.

[7] Harrison, J., “Blackout of 1996,” October 31, 2008. Available: https://www.nwcouncil.org/history/blackout. [Accessed December 09, 2017].

[8] Oluwajobi F. I., Ale O. S. and Ariyanninuola A. “Effect of Sag on Transmission Line,” Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.299.8246&rep=rep1&type=pdf. [Accessed Novemeber 24, 2017].

[9] Engber, D. (2005, July 28).” Why are power lines so dangerous?”. Available: http://www.slate.com/articles/news and politics/explainer/2005/07/why_are_power_lines_so_deadly.html. [Accessed November 24, 2017].

[10] King, K. (2003, January 8). “Primer on effects of lightning on electrical T&D”. Available: http://www.elp.com/articles/print/volume-81/issue-8/power-pointers/primer-on-effects-of-lightning-on-electrical-td.html. [Accessed November 24, 2017].

[11] Mobarakei, S. , Sami, T., & Porkar, B. (2012, June 21). “Back Flashover phenomenon analysis in power transmission substation for insulation coordination”. Available: http://ieeexplore.ieee.org/document/6221567/. [Accessed November 24, 2017].

[12] Panth, D., “Reasons for failure of transmission tines and their prevention strategies,” 2014, pp. 1-4.

[13] Yun, M., & Yuxin, B. (2010, August 26). “Research on the architecture and key technology of Internet of Things (IoT) applied on smart grid”.Available: http://ieeexplore.ieee.org/document/5557611/. [Accessed November 20, 2017].

[14] “Arduino Uno Rev3,” Available: https://store.arduino.cc/arduino-uno-rev3. [Accessed November 20, 2017].

[15] “Adafruit Learning System”. Available: https://learn.adafruit.com/. [Accessed November 24, 2017].

[16] AlKarkuki, A. .ESP8266 esp-01 .

[17] Barrag, H., “Breadboard \ Wiring,” Available: http://wiring.org.co/learning/tutorials/breadboard/. [Accessed November 24, 2017].