NIPES

Journal of Science and Technology Research

Journal homepage: www.nipesjournals.org.ng

Design and Implementation of an Internet of Things Based Smart Fan

***Oriaifo A. P. and Obasohan O.**
Department of Electrical/Electronic Engineering, Faculty of Engineering, University of Benin, PMB 1154, Benin City, Nigeria.
*patrick.oriaifo@uniben.edu; osamuwa.obasohan@uniben.edu

ARTICLE INFORMATION

Article history:
Received 23 April 2022
Revised 28 April 2022
Accepted 02 May 2022
Available online 10 June 2022

Keywords:
Smart devices, IoT, ATMEGA328P, ESP12F, MIT App Inventor, Smart fan

https://doi.org/10.37933/nipes.e/4.2.2022.11
https://nipesjournals.org.ng

ABSTRACT

Thermal comfort is a pervasive necessity for occupants of the equatorial region, and technically, it is achieved through devices like fans and air conditioning systems. Traditional fans are not automated which has the benefits of human convenience and management of energy consumption. Thus, this paper proposes a smart fan with automation functionality. The model consists of a microcontroller (ATMega328P) that allows dynamic and faster control and a temperature sensor that reads the room temperature. The connection to the internet is accomplished by an ESP12F Wi-Fi module. The developed IoT based smart fan was controlled and regulated in terms of speed via the mobile screen interface of the application which also displayed the temperature of the room. The user is also notified of changes in the monitored data beyond its desired range. The device has a battery backup to support the functionality in a power outage. The experiment had a timely response of an average of 2.1 seconds for 6 trials.

1.0. Introduction

The Internet of Things (IoT) is the inter-networking of physical devices, vehicles (also referred to as connected devices and smart devices) buildings and other items embedded with electronics, software, sensors, actuators and network connectivity that enable these objects to collect and exchange data [1,2]. Nigeria is located in the northern hemisphere above the equator, making it a hot region all year long. Humans provide thermal comfort through fans and air conditioning systems [3]. However, most of the living rooms in Nigeria are not fitted with an air conditioning system so a fan is one of the most adopted ways of circulating air around the environment and it is used in most houses [4]. In a traditional fan, the user has to control it manually each time by walking toward the switchboard. Also, there is wastage of electrical energy and it is controlled manually [5]. An improvement to such a design would be automation [6]. This automation would reduce the consumption of power and human effort. To overcome the drawbacks of the traditional system, an automatic control system for a fan has been proposed. The IoT has also emerged to make smart devices smarter [7]. With the evolution of the Internet of Things (IoT), most of the manually controlled electrical and electronic devices can be controlled automatically using a smartphone or other similar devices [6]. The main concept of IoT is that it can create a virtual connection between a hub or a network and an electrical appliance. This virtual connection helps to control, locate, and track down these connected objects on the basis of the device-to-device connectivity concept, the use of smart sensors together with communication technologies such as Wi-Fi, Bluetooth etc [8]. Thus, this study presents a design that will automate the traditional fan by adopting IoT technology.

Some recent related research includes a home automation system (HAS) developed to provide low-cost monitoring and controlling environment that brings safety and interconnection of devices in a home [8]. The system uses a Wi-Fi-based Wireless Sensor Network (WSN) that monitors and controls devices through a smartphone application. The application was developed using Android Studio based on the Java platform and achieved the development of a solution that is cost-effective and flexible in the control of devices and implements a wide range of sensors to capture various parameters. Also, [9] advanced the research in home automation by designing a basic home automation application on Raspberry Pi by reading the subject of E-mail. In a similar manner, this study proposes the design of a smart fan that is controlled through a mobile application. The system uses temperature sensors to determine the speed at which the fan should operate and the entire system is controlled by an ATmega328P microcontroller unit.

2.0. Materials and Method

The components of this design include the ESP 32 module, ATMEGA328P Microcontroller, transistors, Resistors, Capacitors, Fan, battery, Buck converter, Crystal Oscillators. Table 1 summarizes briefly the descriptions of these components.

Table 1: Description of the components used

Component	**Description**
ESP 32 Module [10]	Wi-Fi and Bluetooth module
ATMega382P	Microcontroller, 8-bit AVR RISC-based
Crystal Oscillator	Produces a constant frequency electrical signal
MOSFET	Transistor for amplification and switching
BMP180 Sensor	Barometric pressure sensor
12V 7.5 AH Deep cycle battery [11]	Produces electrical energy from chemical energy
Relay [12]	An electromechanical device that can be used to make or break an electrical connection
Fan	A device that circulates wind

In addition to the list of components used to make the fan smart, there is one other important part of the IoT device, which is the mobile application also referred to as an app or mobile app. It is a computer program software application, which is designed to work on a mobile device such as a laptop, phone, tablet or watch. The smart interface used in the project is named “IoT App” built with MIT App Inventor. The application was used to send feedback to the ESP32 module which is connected to the microcontroller. It is built in JavaScript. The block diagram of the developed system is shown in Figure 1 and it shows a representation of power and signal flow between the different components of the design. The block diagram describes the interconnection of the components and thus when the battery is duly charged the FAN is switched ON, the power indicator light comes ON and the circuit of the system is powered. The IoT mobile application is also switched ON in the mobile phone with an internet connection. When an input is sent from the mobile application through the webserver to the ESP12F Module, it is then read and interpreted by the microcontroller and sent to the motor of the fan.

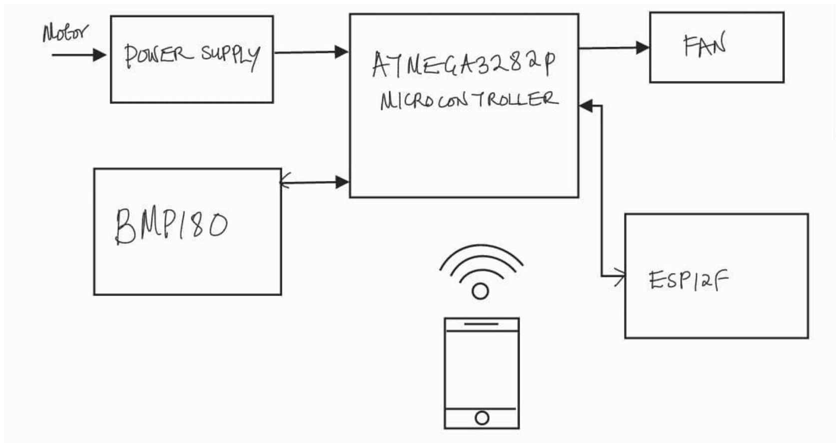

Figure 1: Block Diagram

The design is subdivided into two stages; hardware construction and implementation, and the mobile application design.

2.1. Hardware construction and implementation

The hardware construction and implementation involved the microcontroller unit, Wi-Fi and Bluetooth module and a regulator circuit between the microcontroller and the Wi-Fi and Bluetooth module. The microcontroller operates with an electrical signal at a constant frequency for its internal operations. This is generated either internally or externally. In the case of an external oscillating signal, a crystal oscillator is used and the circuit that controls the frequency produced is shown in Figure 2. Pins XTAL1 and XTAL2 are input and output, respectively, of an inverting amplifier that can be configured for use as an On-chip Oscillator. This Crystal Oscillator is a low power oscillator, with a reduced voltage swing on the XTAL2 output. It gives the lowest power consumption, but is not capable of driving other clock inputs, and may be more susceptible to noise in noisy environments.

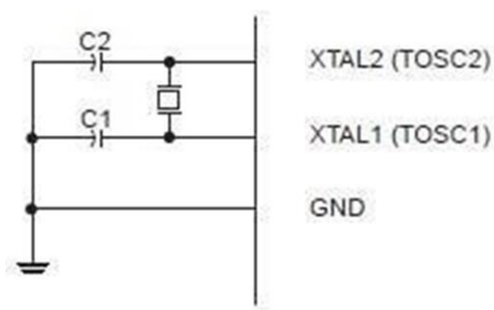

Figure 2: Crystal Oscillator connection with capacitors

The complete circuit connection of the microcontroller and the crystal oscillator is shown in Figure 3.

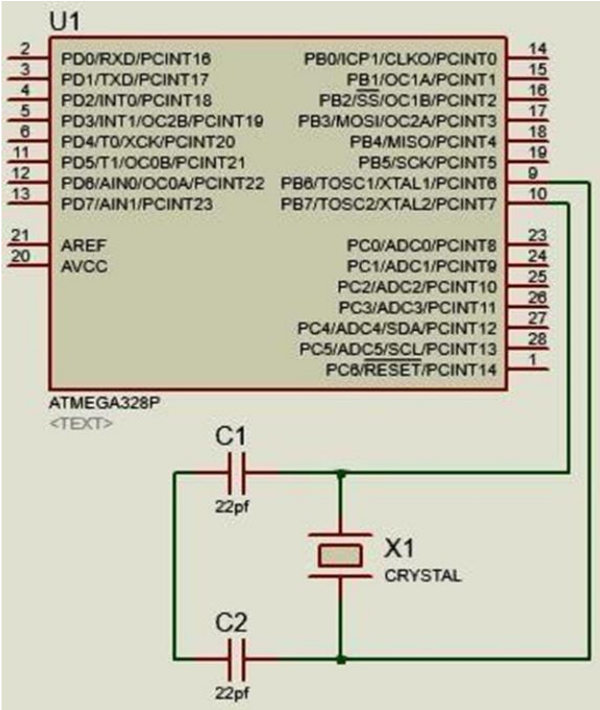

Figure 3: Crystal Oscillator circuit connection to the microcontroller

Table 2 presents the characteristics of the microcontroller used.

Table 2: ATmega328 characteristics

Parameter	**Value**
Maximum CPU speed	20 MHz
Performance	20 MIPS at 20MHz
Flash memory	32 KB
SRAM	2 KB
EEPROM	1 KB
Package pin count	28 or 32
Capacitive touch sensing channels	16
Maximum I/O pins	23
External interrupts	2
USB interface	No

2.2. Regulator circuit

This is a two-stage amplifier and a MOSFET. It acts as a switch, as a constant current source, an increase in the current flowing will cause an increase in the speed of the fan which is also represented on the mobile app as a displayed motor current.

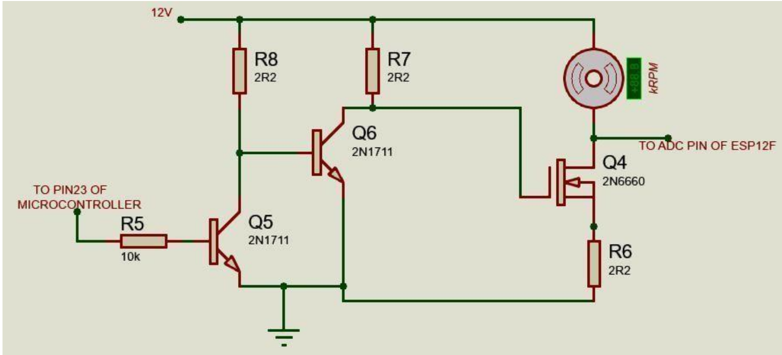

Figure 4 Regulator circuit connection

2.3. Buck converter

It is connected to the power supply and converts a higher DC to a lower DC. It is needed to step the 12V DC supply to a compatible voltage for the ESP Module (5V).

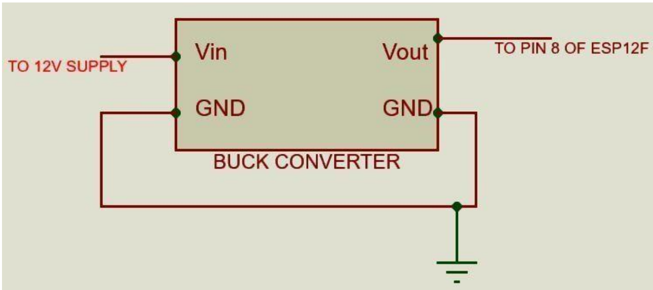

Figure 5 Buck converter unit

2.4. BMP 180 unit

BMP180 is one of the sensors of the BMP XXX series. It can measure temperature and altitude; it measures the pressure that falls in the range of 300 to 1100hPa. It has a relatively high accuracy of ±0.12hPa. Its operating voltage is between 1.3V to 3.6V and the input voltage is between 3.3V to 5.5V. Figure 6 shows the circuit diagram of the BMP 180 unit while Figure 7 shows the pinout schematics of the barometric sensor and the description of those pins is given in Table 3.

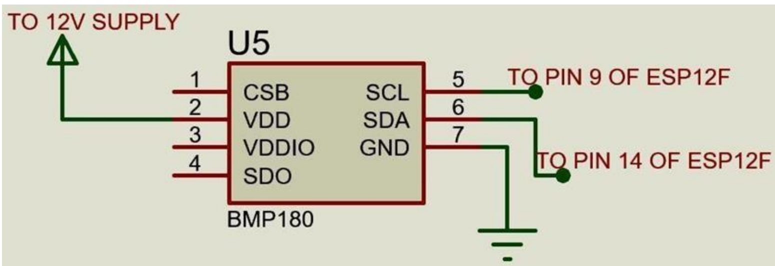

Figure 6 BMP 180 unit

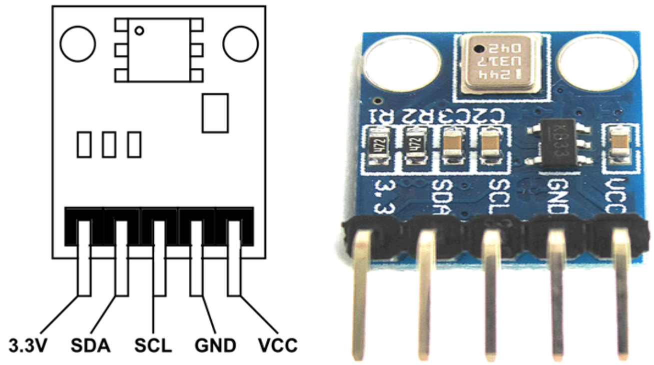

Figure 7 BMP180 pinout schematic diagram

Table 3: BMP180 pin description

Pin Name	Description
VCC	Connected to +5V
GND	Connected to ground.
SDA	Serial Data pin (I2C interface)
SCL	Serial Clock pin (I2C interface)
3.3V	If +5V is not present. Can power module by connecting +3.3V to this pin.

2.5. Battery sizing

The battery is suggested as a backup to store energy during charging and to compensate for days of lesser charging. The battery system voltage is 12V, this is because the voltage of the table fan is also 12V, this will also enable a reduction of quantity and cost of components and installation.

Maximum depth of discharge (DoD) = 50%

The number of storage days (no charge allowance) = 2

$$Battery\ capacity\ = \frac{total\ energy}{system\ voltage}.\ [11] \qquad (1)$$

From equation 1, the required battery capacity is $\frac{150Wh}{12V} = 12.5\text{Ah}$

There is now a need to compensate for when the charge in the battery would be less than the design i.e., considering 2 days of storage (to compensate for no charge situation) and the battery is to be discharged by at most 50%.

The battery capacity that will meet this condition as

$$\frac{12.5Ah \times 2}{0.5} = 50Ah \qquad (2)$$

Different sizes of deep cycle batteries available in the market include 50Ah, 75Ah, 100Ah etc.

Therefore, the number of 12V, 50Ah batteries that will meet this need = 1

2.6.Voice control unit

The ESP12F module is configured with the help of google assistant and then synchronized with the hardware through relays. The configuration was done with the following components;

i. ESP12F Module
ii. 5VDC 10A relays(4pcs)
iii. 330 Resistors (4pcs)
iv. D882 Transistor (4 pcs)
iv. Lead for soldering
v. Diodes (4 pcs)
vi. Vero board
vii. Arduino configuration software
viii. Android phone

In the form of voltages from the ESP12F output pins, signals are not enough to activate a relay; hence we use a transistor to amplify the voltage just enough for the relays to be activated. The circuit is shown in Figure 8. In actualising this, a resistor is connected from the output pins of the ESP12F to the base of the D882 transistor. Then the collector is connected to the negative coil terminals. All emitters are grounded because it is a common-emitter configuration. Diodes were used across the coil terminals to avoid reverse current in any form.

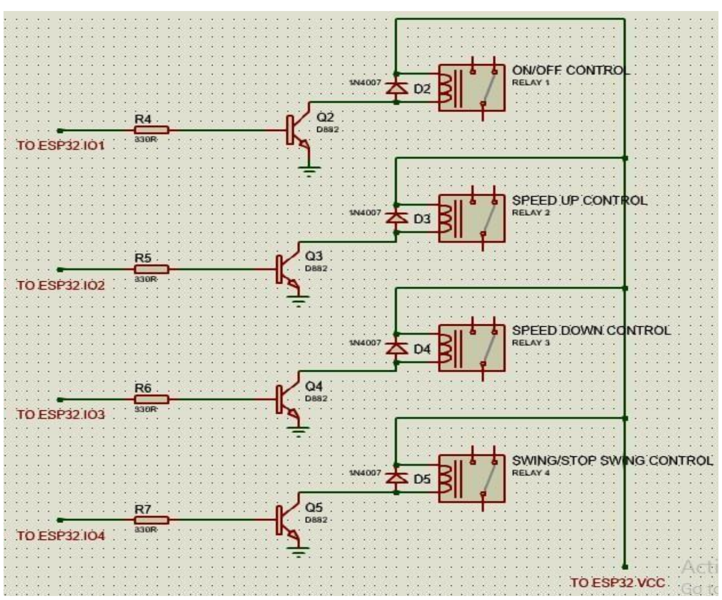

Figure 8 Schematics of the Relay Connection

The ESP12F was programmed using the Arduino simulation software. The commands programmed into the ESP12F are:

i. "ON"
ii. "OFF"
iii. "SPEED UP"
iv. "SPEED DOWN"
v. "SWING"
vi. "STOP SWING"

The "ON" command turns on supply to the main fan motor at speed 0. The "SPEED UP" command increases the voltage going to the main fan motor; hence, increasing its speed. The "SPEED DOWN" command reduces the voltage going to the main fan motor; hence, decreasing its speed. The "SWING" command turns on the supply to the gear motor, and it starts rotating the entire fan head. The "STOP SWING" command turns off the supply to the gear motor and the rotation stops. Lastly, the "OFF" command turns off the supply to the fan.

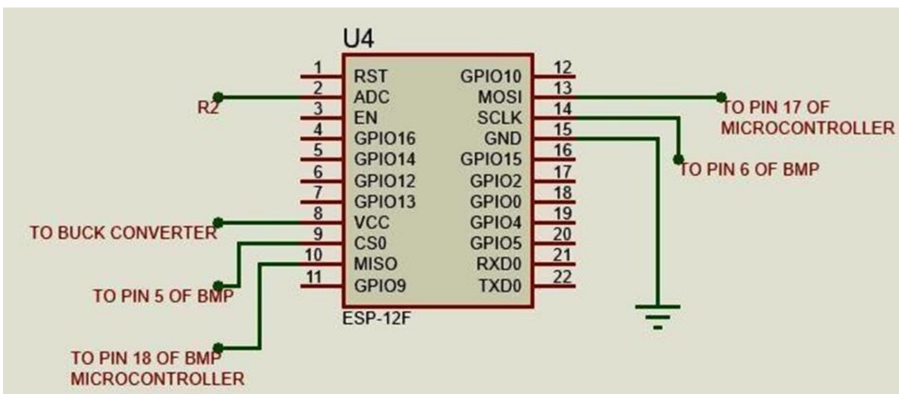

Figure 9: ESP12F circuit connection

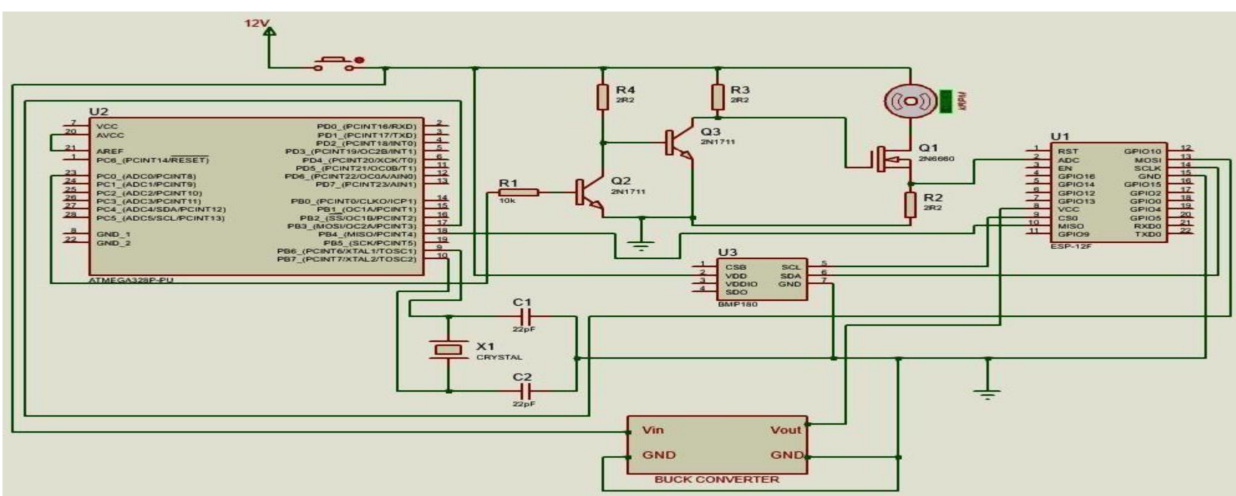

Figure 10 Complete schematics showing all components connected together

2.7. Mobile application and programming

This stage deals with the mobile application interface between the ESP12f and the IoT mobile application. The application was built with the MIT APP inventor [13]. It was built in JavaScript language and integrated with C programming language into the Microcontroller.

Table 4 Bill of Engineering Measurements and Evaluation

S/N	ITEMS	QUANTITY	UNIT COST (₦)	TOTAL COST (₦)
1	ATMEGA328P	1	2000	2000
2	RIBBON CABLE	5	200	1000
3	ESP32	1	5300	5300
4	TERMINAL SCREWS	4	50	200
5	BUCK CONVERTER	1	3500	3500
6	TRANSISTORS	4	40	200
7	CRYSTAL OSCILLATORS	2	100	200
8	CAPACITOR	1	30	30
9	RESISTORS	10	20	200
10	BMP180	1	2000	2000
11	JUMPER	1	400	400
12	VERO BOARD	2	400	800
13	BUCK CONVERTER	1	1200	1200
14	FAN	1	29000	29000
	TOTAL COST			₦46,030

3.0. Results and Discussion

Figure 11 is the soldering of the relay connection and the corresponding complete circuit. Figures 12 and 13 are the complete circuitry of the model and its placement in the fan. Figure 14 shows the interface of the IoT App designed with MIT App inventor. It shows three stages of the system test; before connection to the fan, after connection which shows the temperature of the room, the motor current and the battery level; lastly the speech recognition configuration of the model.

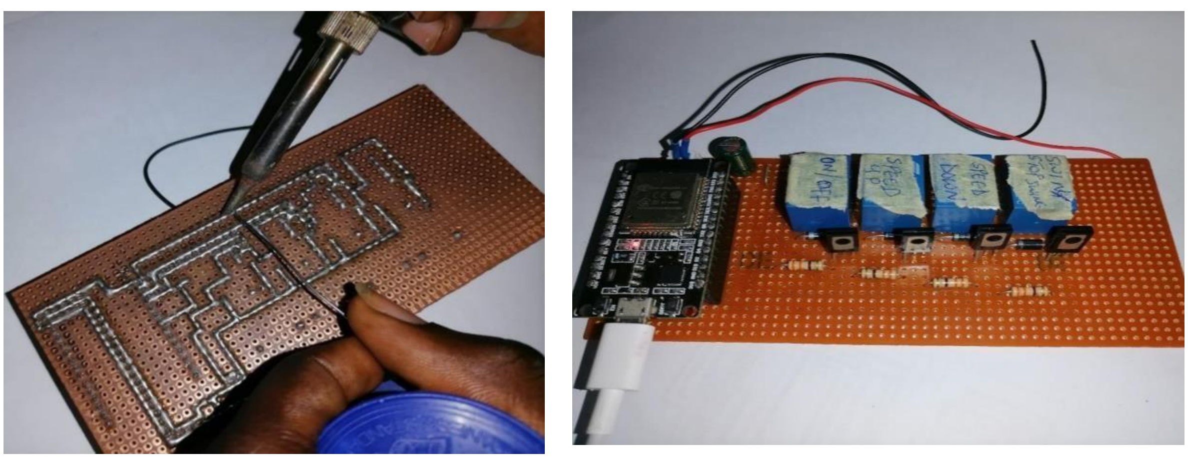

Figure 11 Veroboard during soldering (b) Complete circuit of the relay connection

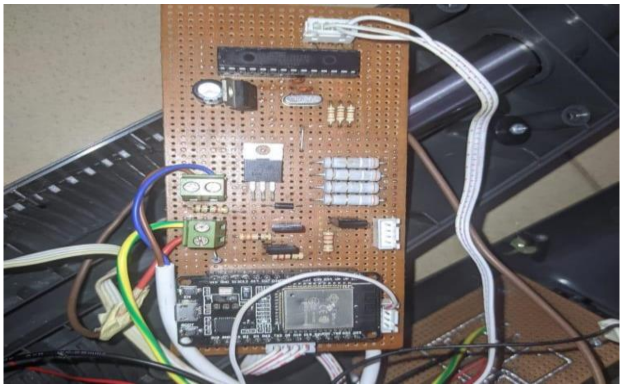

Figure 12 Schematics of the Microcontroller and ESP Module

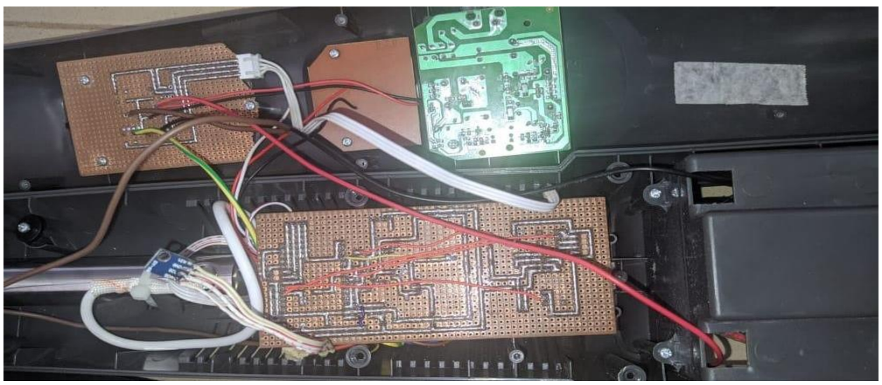

Figure 13 Overview of the entire circuitry

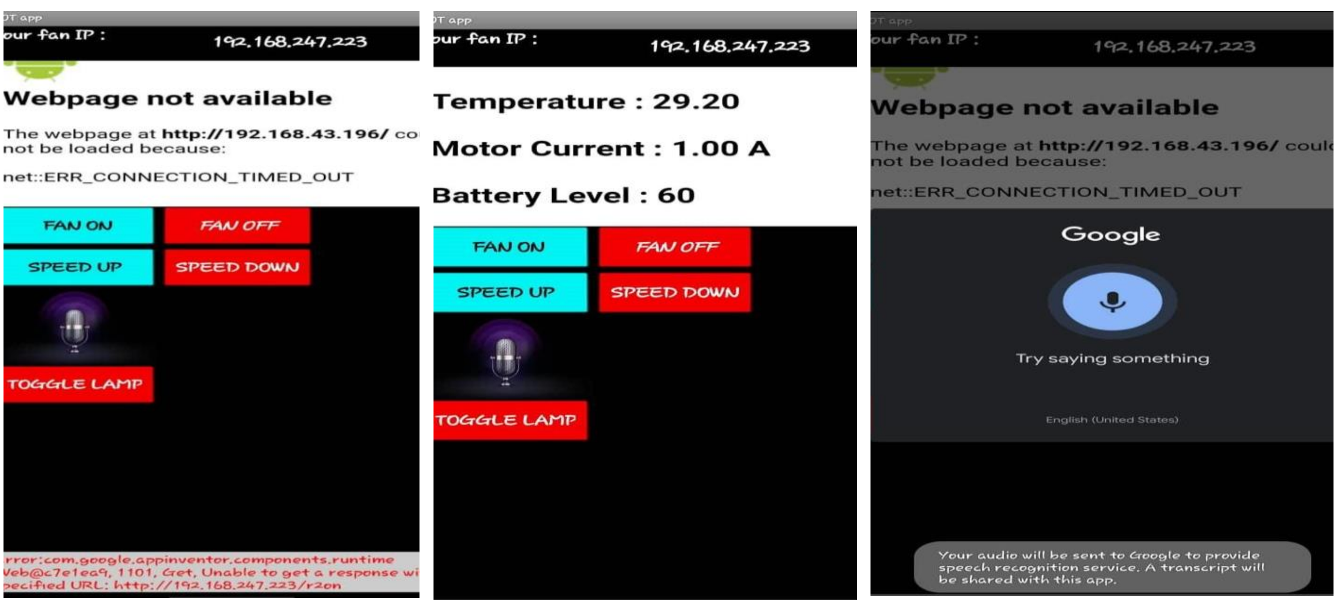

Figure 14 (a) Display screen of the Mobile Application without internet connection displaying an Error message (b) Display screen of the Mobile Application with the fan fully functional (c) Display screen showing the Google Speech to text function.

During the test that was carried out on the implementation, the average time taken to respond to the command from the App was calculated. Using a stopwatch, the time taken to execute each command was recorded and Table 5 shows the records for 6 command attempts. Figure 15 is the corresponding plot of Table 5 and Figure 16 is the complete view of the smart fan that was implemented.

Table 5: Response Time for Commands

COMMAND ATTEMPTS	RESPONSE TIME(s)
1	2.1
2	2.8
3	2.0
4	2.0
5	1.8
6	1.9

From Table 5, the response time $= \frac{2.1+2.8+2.0+2.0+1.8+1.9}{6} = 2.1\ seconds.$

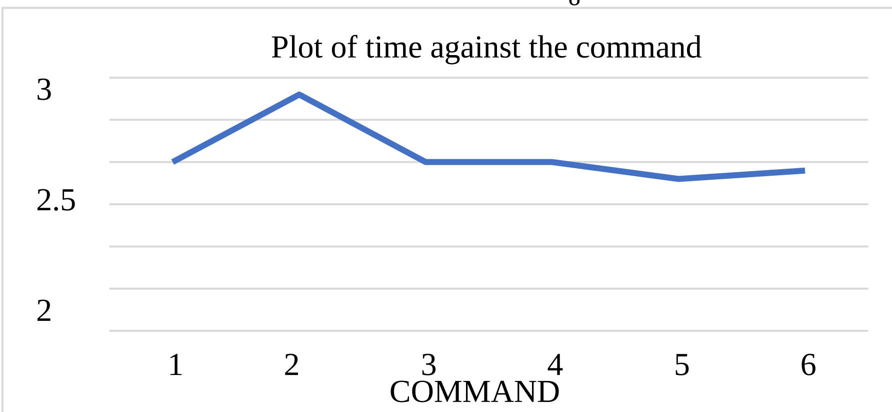

Figure 15 Plot of time against the command

Figure 16 Fully operational Fan

4.0. Conclusion

In this paper, we successfully implemented the process of operating or controlling a fan using various control systems automatically or remotely. This method of operating or controlling such applications is referred to as automation, thus this paper proposed a new technique of speed control of the standing fan with a mobile device and voice control via google assistant. The proposed fan is smart with IoT technology such that it makes use of a mobile app to control the entire system which executes a set of commands. The model has its roots in an IoT platform that allows devices to synchronize with the IoT platform so that it can be controlled remotely. This method of speed control is economical, saves energy and uses less human effort.

Acknowledgment

The authors wish to acknowledge the assistance and contributions of the project students of the Department of Electrical/Electronic Engineering, Faculty of Engineering, University of Benin, Benin City toward the success of this work.

Conflict of Interest

There is no conflict of interest associated with this work.

References

[1.] Wigmore, I., 2014. Internet of things (iot). *TechTarget.*
[2.] Mattern, F. & Flörkemeier, C., 2010. Vom Internet der Computer zum Internet der Dinge. Informatik-Spektrum, February, Volume 33, p. 107–121
[3.] Ashrae, A. N. S. I., 2013. Standard 55-2013: Thermal environmental conditions for human occupancy. American Society of Heating, Refrigerating, and Air-Conditioning Engineers, Inc. Atlanta.
[4.] Patowary K. (2019). Punkah: The Hand Operated Ceiling Fans of Colonial India. [online] Available at: https://www.amusingplanet.com/2019/09/punkah-hand-operated-ceiling-fans-of.html [Accessed 16 Mar. 2022].
[5.] Jake (2018). 10 Most Popular Type of Fans for Home Use that Every Dweller Must Know. [online] Top Cooling Fan. Available at: https://www.topcoolingfan.com/type-of-fans-for-home/ [Accessed 16 Mar. 2022].
[6.] Dey, S., Roy, A. & Das, S., 2016. Home automation using Internet of Thing. s.l., IEEE, p. 1–6.
[7.] Ishrat, T., Rahaman, M. A. & Ahammad, A., 2014. Smart fan for human tracking. s.l., IEEE.
[8.] Vikram, N. et al., 2017. A low cost home automation system using Wi-Fi based wireless sensor network incorporating Internet of Things (IoT). s.l., IEEE, p. 174–178.
[9.] Jain, S., Vaibhav, A. & Goyal, L., 2014. Raspberry Pi based interactive home automation system through E-mail. s.l., IEEE, p. 277–280.
[10.] Biswas, S. B. & Iqbal, M. T., 2018. Solar water pumping system control using a low cost ESP32 microcontroller. s.l., IEEE, p. 1–5.
[11.] Skundin, A. M., 2000. Batteries. Kirk-Othmer Encyclopedia of Chemical Technology, p. 1–26.
[12.] circuitdigest.com. (2022). What is a Relay? How Relay Works & Different Types of Relay. [online] Available at: https://circuitdigest.com/article/relay-working-types-operation-applications [Accessed 16 Mar. 2022].
[13.] appinventor.mit.edu. (2012). Getting Started with MIT App Inventor. [online] Available at: https://appinventor.mit.edu/explore/get-started.