SMART LAB USING IOT

A PROJECT REPORT

Submitted by

T.RAkhil (2013/b/11)
Apurv Rathore (2013/b/42)

in partial fulfillment for the award of the degree of

B.Tech

IN

ELECTRONICS SYSTEM ENGINEERING

At

National Institute Of Electronics And Information Technology

Aurangabad

2016-2017

Project Report Approval for B.E

This project report entitled **SMART LAB USING IOT** by **T.R Akhil, Apurv Rathore** is approved for the degree of **Bachelor in Technology**.

Examiners:

1._____.

2._____.

Project Guide(s):

_____

Mr.Lakshman K. (Scientist 'C')

Project Coordinator:

_____

Mr A.K Joshi (Principal Technical Officer)

Date:

Place

DECLARATION

We hereby declare that the project entitled **"SMART LAB USING IOT"** submitted for the **B.Tech** Degree is Our original work and the project has not formed the basis for the award of any degree, associate ship, fellowship or any other similar titles.

Signature of the Student

T.R Akhil

Apurv Rathore

ACKNOWLEDGEMENT

Before we get into thick of things I would like to add few heartfelt words for the people who are part of our team as they have been unending contribution right from the start of construction of the report.

Apart from the team I am indebted to the numbers of persons who have provided helpful and constructive guidance in the draft of material.

I acknowledge with deep sense of gratitude towards the encouragement In the form of substantial assistance provided each and every member of my team.

I would like to extend my sincere thanks to our guide

Mr.Lakshman K.(Scientist 'c')

For providing us the required technical guidance in the process of preparing this report.

ABSTRACT

This project presents a design and prototype implementation of new lab automation system that uses WiFi technology as a network infrastructure connecting its parts. The proposed system consists of two main components; the first part is the server (web server), which presents system core that manages, controls, and monitors.

Users and system administrator can locally (LAN) or Remotely (internet) manages and control system code. Second part is hardware interface module, which provides appropriate Interface to sensors and actuator of Lab automation system.

Unlike most of available home automation system in the market the proposed system is scalable that one server can manage many Hardware module.

TABLE OF CONTENTS

Contents

CHAPTER NO.01

INTRODUCTION

INTRODUCTION

The project aims at designing an advanced home automation system using normal web server and Wi-Fi technology. The devices can be switched ON/OFF and sensors can be read using a Personal Computer (PC) through Wi-Fi.

Automation is the most frequently spelled term in the field of electronics. The hunger for automation brought many revolutions in the existing technologies. These had greater importance than any other technologies due to its user-friendly nature. These can be used as a replacement of the existing switches in home which produces sparks and also results in fire accidents in few situations. Considering the advantages of Wi-Fi an advanced automation system was developed to control the appliances in the house.

Wi-Fi (Short for Wireless Fidelity) is a wireless technology that uses radio frequency to transmit data through the air. Wi-Fi has initial speeds of 1mbps to 2mbps. Wi-Fi transmits data in the frequency band of 2.4 GHz. It implements the concept of frequency division multiplexing technology. Range of Wi-Fi technology is 40-300 feet.

The controlling device for the automation in the project is a Arduino UNO. The data sent from PC over Wi-Fi will be received by Wi-Fi module connected to Arduino UNO. Arduino UNO reads the data and decides the switching action of electrical devices connected to it through Relays.

CHAPTER NO.02

OBJECTIVE OF PROJECT

- The goal of this project is to develop a lab Automation system that gives the user complete control over all remotely controllable aspects of any lab where ever installed .
- The automation system will have the ability to be controlled from a central host PC, the Internet, and also remotely accessed via a Pocket PC with a Windows Mobile based application.

CHAPTER NO.03
LITERATURE SURVEY

Literature survey:

Review of Related Literature:

When people think about home automation, most of them may imagine living in a smart home: One remote controller for every household appliance, cooking the rice automatically, starting air conditioner automatically, heating water for both automatically and shading the window automatically when night coming. To some extent home automation equals to smart home. They both bring out smart living condition and make our life more convenient and fast.

Review of Foreign Studies:

In their paper, Tan, Lee and Soh (2002) proposed the development of an Internet-based system to allow monitoring of important process variables from a distributed control system (DCS). This paper proposes hardware and software design considerations which enable the user to access the process variables on the DCS, remotely and effectively

Potamitis, Georgila, Fakotakis, and Kokkinakis, G. (2003) suggested the use of speech to interact remotely with the home appliances to perform a particular action on behalf of the user. The approach is inclined for people with disability to perform real-life operations at home by directing appliances through speech. Voice separation strategy is selected to take appropriate decision by speech recognition

In the year 2006 , S. M. Anamul Haque,S. M. Kamruzzaman and Md. Ashraful Islam proposed a system entitled "A System for Smart-Home Control of Appliances Based on Time and Speech Interaction" that controls the home appliances using the personal computer. This system is developed by using the Visual Basic 6.0 as programming language and Microsoft voice engine tools for speech recognition purpose. Appliances can be either controlled by timer or by voice command.

Ciubotaru-Petrescu, Chiciudean, Cioarga, and Stanescu (2006) present a design and implementation of SMS based control for monitoring systems. The paper has

three modules involving Sensing unit for monitoring the complex applications. A processing unit, that is microcontroller and a communication module that uses GPRS modem or cell phone via serial port RS-232. The SMS is used for status reporting such as power failure.

Jawarkar, Ahmed, Ladhake, and Thakare (2008) propose remote monitoring through mobile phone involving the use of spoken commands. The spoken commands are generated and sent in the form of text SMS to the control system and then the microcontroller on the basis of SMS takes a decision of a particular task.

Prof. Era Johri Dept. Of Information And Technology K.J.Somaiya College Of Engineering VIDYAVIHAR, MUMBAI "Remote Controlled Home Automation Using Android Application via WiFi Connectivity".

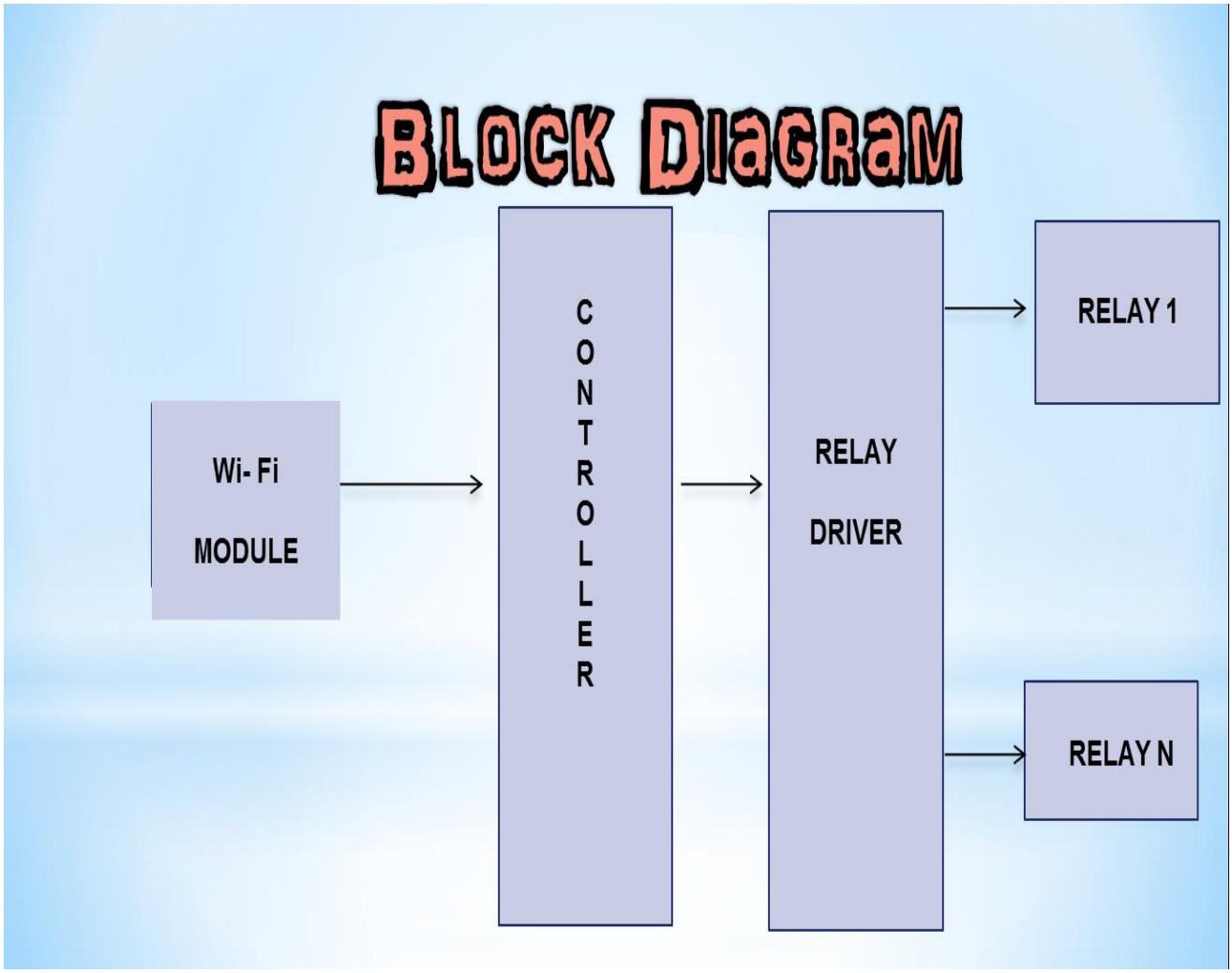
CHAPTER NO.04

SCOPE OF PROJECT

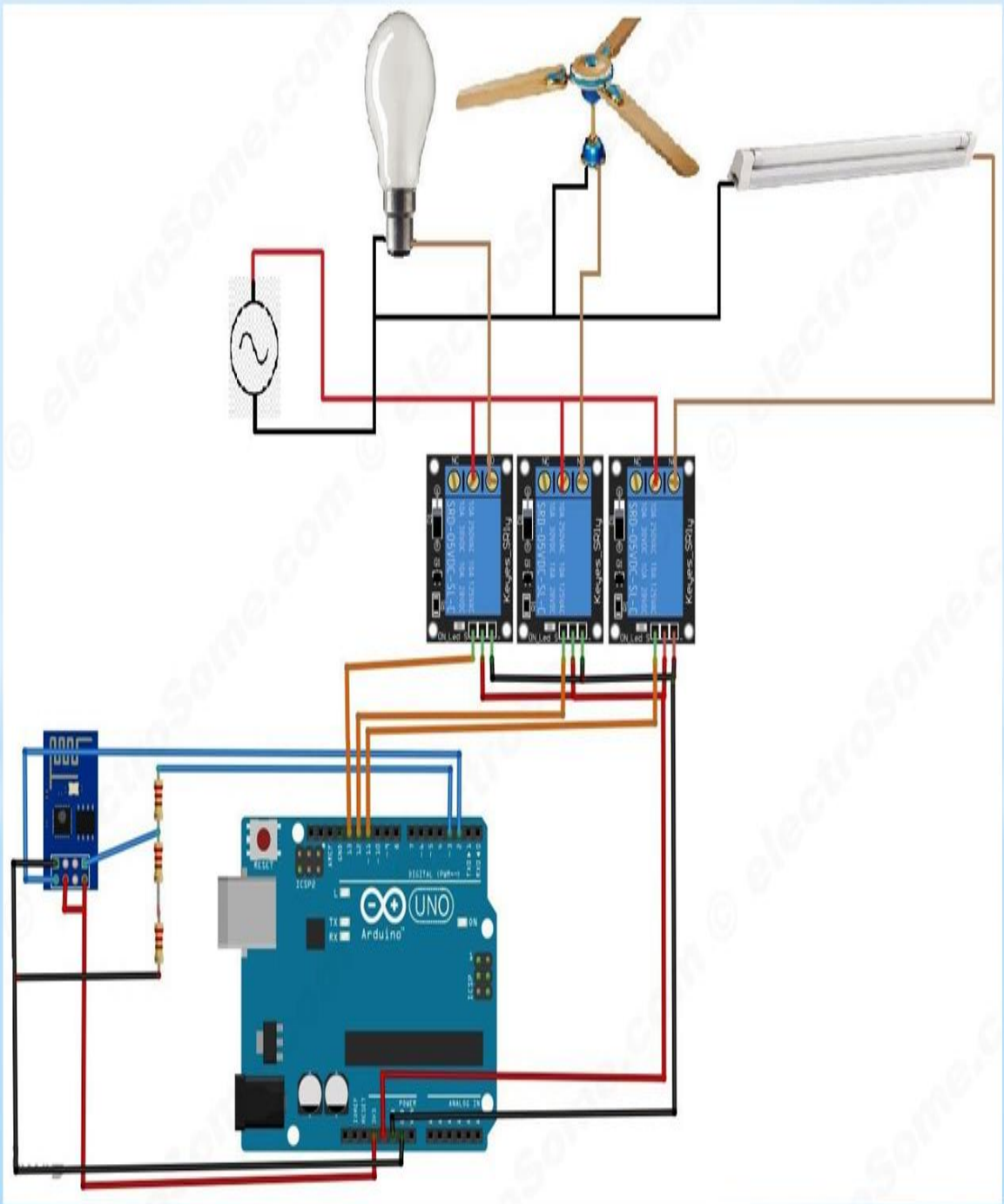
Day by day, the field of automation is blooming and these systems are having great impact on human beings. The project which is to be implemented is a home automation using Easy IOT Webserver and WIFI and has very good future development.

In the current system webserver is installed on a windows PC so the home appliances can be controlled using only by using the device on which webserver is installed.This can be further developed installing webserver on cloud .

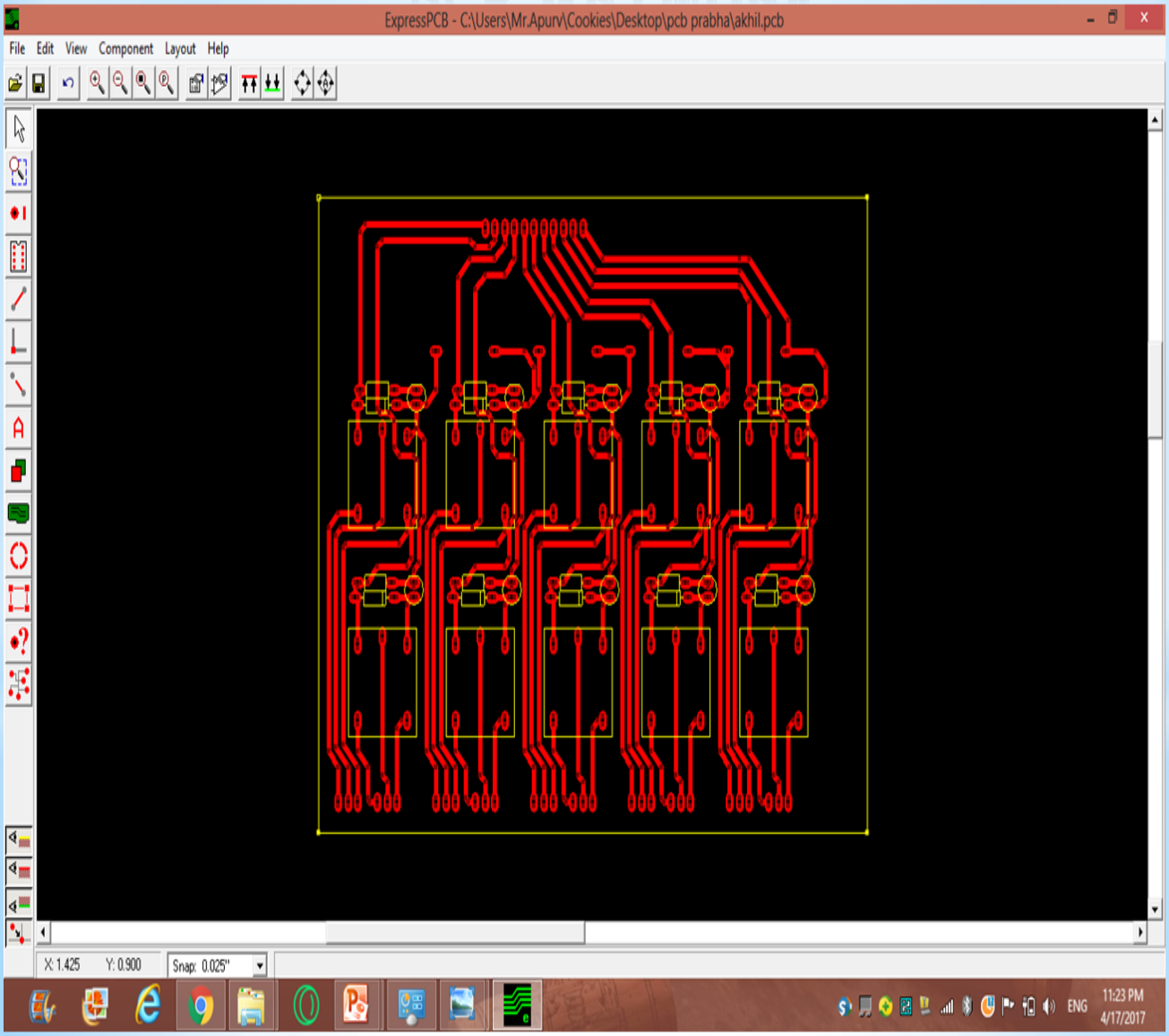
Advantage of installing webserver on the cloud is that home can be controlled by using any device which has WIFI 802.1 and a web browser. By visiting the IP address of the cloud the control actions can be taken.


BLOCK DIAGRAM
Wi- Fi
MODULE
CONTROLLER
RELAY
DRIVER
RELAY 1
RELAY N

Circuit Diagram

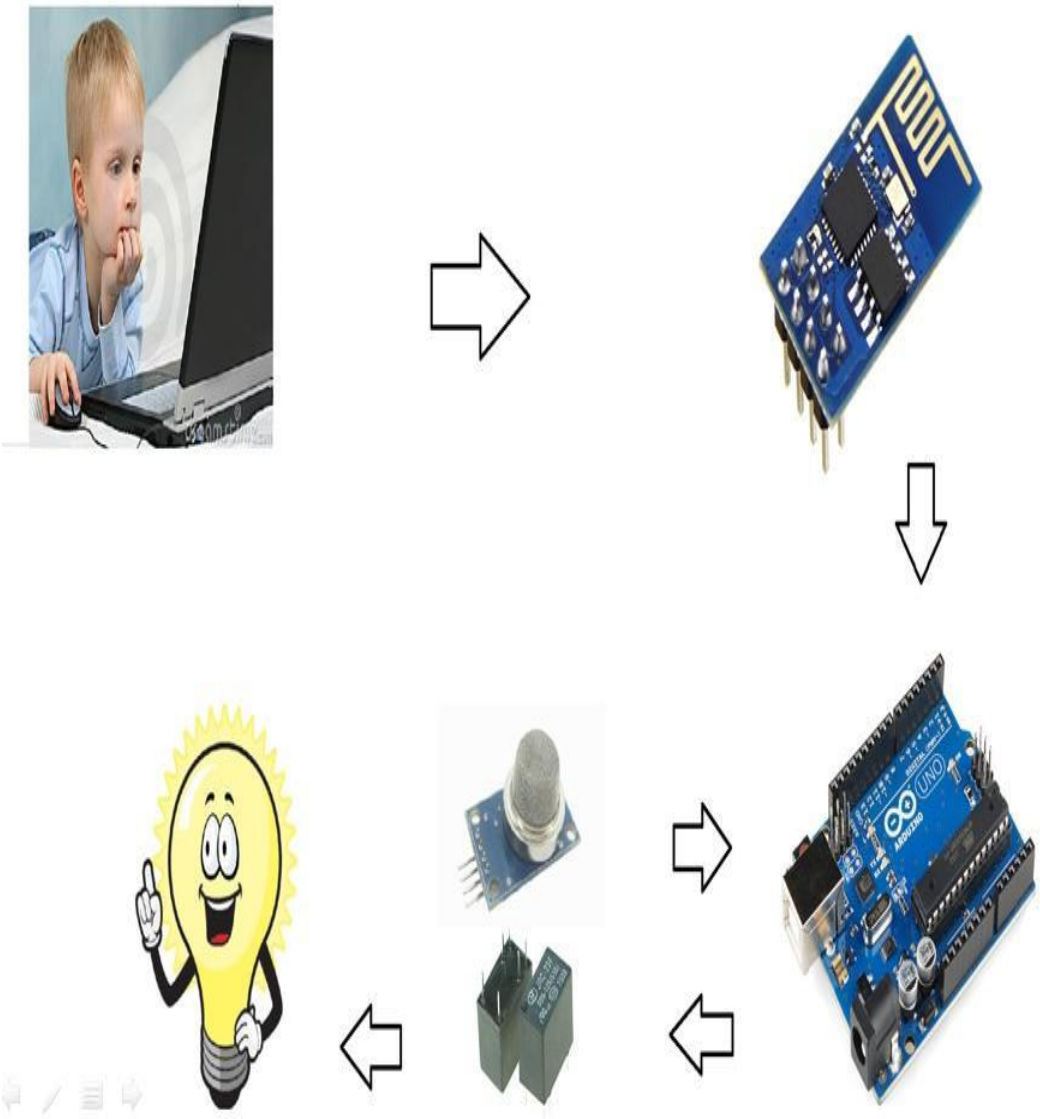


*PCB ARTWORK


ExpressPCB - C:\Users\Mr.Apurv\Cookies\Desktop\pcb prabha\akhil.pcb
File Edit View Component Layout Help
X: 1.425 Y: 0.900 Snap: 0.025"
ENG 11:23 PM 4/17/2017

CHAPTER NO.05

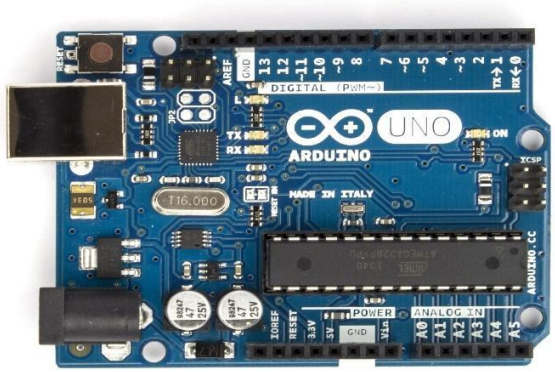
METHODOLOGY



CHAPTER NO.06

HARDWARE

Arduino:-



The Arduino Uno is a microcontroller board based on the ATmega328 (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.The Uno differs from all preceding 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.

The Uno is a microcontroller board based on the ATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started..

"Uno" means one in Italian and was chosen to mark the release of Arduino Software (IDE) 1.0. The Uno board and version 1.0 of Arduino Software (IDE) were the reference versions of Arduino, now evolved to newer releases. The Uno board is the first in a series of USB Arduino boards.

Technical specifications:-

MicrocontrollerATmega328P

Operating Voltage 5V

Input Voltage (recommended) 7-12V

Input Voltage (limit) 6-20V

Digital I/O Pins 14 (of which 6 provide PWM output)

PWM Digital I/O Pins 6

Analog Input Pins 6

DC Current per I/O Pin 20 mA

DC Current for 3.3V Pin 50 mA

Flash Memory 32 KB (ATmega328P)

of which 0.5 KB used by bootloader

SRAM 2 KB (ATmega328P)

EEPROM 1 KB (ATmega328P)

Clock Speed 16 MHz

Length 68.6 mm

Width 53.4 mm

Weight 25 g.

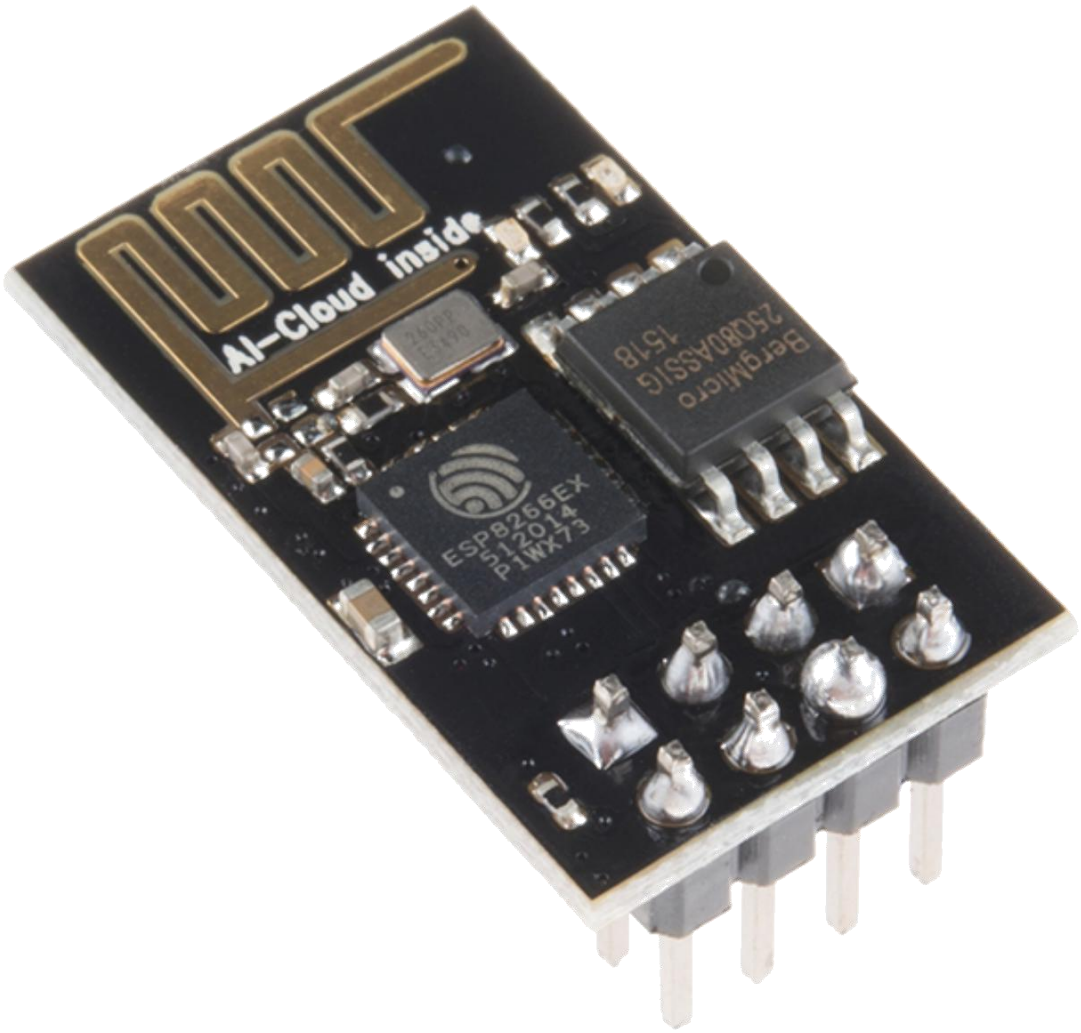
Arduino Code:-

```
#include <SoftwareSerial.h>

#define DEBUG true

Software Serial esp 8266(2,3);
void setup()
{
  Serial.begin(9600);
  esp8266.begin(115200); // your esp's baud rate might be
different

  pinMode(11,OUTPUT);
  digitalWrite(11,LOW);

  pinMode(12,OUTPUT);
  digitalWrite(12,LOW);

  pinMode(13,OUTPUT);
  digitalWrite(13,LOW);

    pinMode(9,OUTPUT);
  digitalWrite(9,LOW);

  pinMode(8,OUTPUT);
  digitalWrite(8,LOW);

  pinMode(7,OUTPUT);
  digitalWrite(7,LOW);
```

```
    pinMode(6,OUTPUT);
  digitalWrite(6,LOW);

  pinMode(5,OUTPUT);
  digitalWrite(5,LOW);

  pinMode(4,OUTPUT);
  digitalWrite(4,LOW);

    pinMode(10,OUTPUT);
  digitalWrite(10,LOW);


  sendData("AT+RST\r\n",2000,DEBUG); // reset module
  sendData("AT+CWMODE=2\r\n",1000,DEBUG); // configure as
access point
  sendData("AT+CIFSR\r\n",1000,DEBUG); // get ip address
  sendData("AT+CIPMUX=1\r\n",1000,DEBUG); // configure for
multiple connections
  sendData("AT+CIPSERVER=1,80\r\n",1000,DEBUG); // turn on
server on port 80
}

void loop()
{
  if(esp8266.available()) // check if the esp is sending a
message
  {


    if(esp8266.find("+IPD,"))
    {
     delay(1000); // wait for the serial buffer to fill up
(read all the serial data)
     // get the connection id so that we can then disconnect
     int connectionId = esp8266.read()-48; // subtract 48
because the read() function returns
                                           // the ASCII
decimal value and 0 (the first decimal number) starts at 48

     esp8266.find("pin="); // advance cursor to "pin="

     int pinNumber = (esp8266.read()-48)*10; // get first
number i.e. if the pin 13 then the 1st number is 1, then
multiply to get 10
     pinNumber += (esp8266.read()-48); // get second number,
i.e. if the pin number is 13 then the 2nd number is 3, then
add to the first number

     digital Write(pinNumber, !digitalRead(pinNumber)); //
toggle pin

     // make close command
     String close Command = "AT+CIPCLOSE=";
     Close Command+=connection Id; // append connection id
     close Command+="\r\n";

     send Data(closeCommand,1000,DEBUG); // close connection
    }
  }
}

/*
* Name: sendData
* Description: Function used to send data to ESP8266.
* Params: command - the data/command to send; timeout - the
time to wait for a response; debug - print to Serial
window?(true = yes, false = no)
* Returns: The response from the esp8266 (if there is a
reponse)
*/
```

```
String sendData(String command, const int timeout, boolean
debug)
{
    String response = "";

    esp8266.print(command); // send the read character to
the esp8266

    long int time = millis();

    while( (time+timeout) > millis())
    {
      while(esp8266.available())
      {

        // The esp has data so display its output to the
serial window
        char c = esp8266.read(); // read the next character.
        response+=c;
      }
    }

    if(debug)
    {
      Serial.print(response);
    }

    return response;
}
```

ESP 8266:-



Description: The ESP8266 WiFi Module is a self-contained SOC with integrated TCP/IP protocol stack that can give any microcontroller access to your WiFi network. The ESP8266 is capable of either hosting an application or offloading all Wi-Fi networking functions from another application processor. Each ESP8266 module comes pre-programmed with an AT command set firmware, meaning, you can simply hook this up to your Arduino device and get about as much WiFi-ability as a WiFi Shield offers (and that's just out of the box)! The ESP8266 module is an extremely cost effective board with a huge, and ever growing, community.

This module has a powerful enough on-board processing and storage capability that allows it to be integrated with the sensors and other application specific devices through its GPIOs with minimal development up-front and minimal loading during runtime. Its high degree of on-chip integration allows for minimal external circuitry, including the front-end module, is designed to occupy minimal PCB area. The ESP8266 supports APSD for VoIP applications and Bluetooth co-existance interfaces, it contains a self-calibrated RF allowing it to work under all operating conditions, and requires no external RF parts.

There is an almost limitless fountain of information available for the ESP8266, all of which has been provided by amazing community support. In the *Documents* section below you will find many resources to aid you in using the ESP8266, even instructions on how to transforming this module into an IoT (Internet of Things) solution!

Features:

- 802.11 b/g/n
- Wi-Fi Direct (P2P), soft-AP
- Integrated TCP/IP protocol stack
- Integrated TR switch, balun, LNA, power amplifier and matching network
- Integrated PLLs, regulators, DCXO and power management units
- +19.5dBm output power in 802.11b mode
- Power down leakage current of <10uA
- 1MB Flash Memory

Integrated low power 32-bit CPU could be used as application processor

SDIO 1.1 / 2.0, SPI, UART

STBC, 1×1 MIMO, 2×1 MIMO

A-MPDU & A-MSDU aggregation & 0.4ms guard interval

Wake up and transmit packets in < 2ms

Standby power consumption of < 1.0mW (DTIM3)

Specification of ESP 8266:

Wi-Fi Direct (P2P), soft-AP

Integrated TCP/IP protocol stack

Integrated TR switch, balun, LNA, power amplifier and matching network

Integrated PLLs, regulators, DCXO and power management units

19.5dBm output power in 802.11b mode

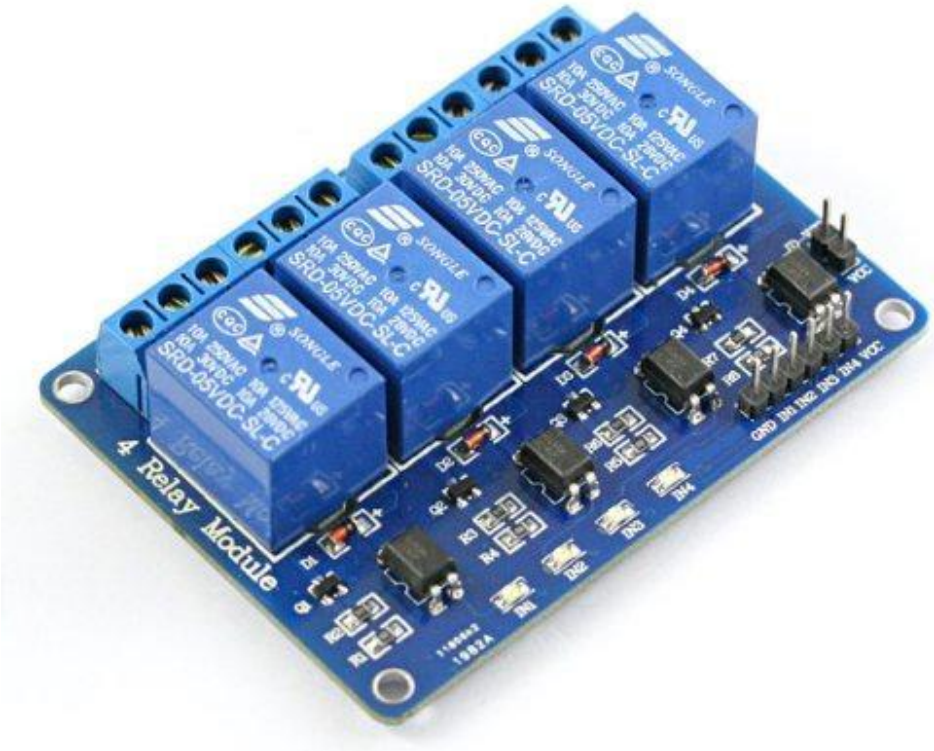
Power down leakage current of <10uA

1MB Flash Memory

Integrated low power 32-bit CPU could be used as application processor

Standby power consumption of < 1.0mW (DTIM3)

Relay board:-



A relay is an electrical device which is generally used to control high voltages using very low voltage as an

Input. This consists of a coil wrapped around a pole and a two small metal flaps(nodes) that are used to close the circuit. One of the node is fixed and other is movable. Whenever an electricity is passed through the coil, it creates a magnetic field and attracts the moving node towards the static node and the circuit gets completed. So, just by applying small voltage to power up the coil we can actually complete the circuit for the high voltage to travel. Also, as the static node is not physically connected to the coil there is very less chance that the Microcontroller powering the coil gets damaged if something goes wrong.

This is Four Channel relay board controlled by computer USB port. The usb relay board is with 4 SPDT relays rated up to 10A each. You may control devices 220V / 120V (up to 4) directly with one such relay unit. It is fully powered by the computer USB port. Suitable for home automation applications, hobby projects, industrial automation. The free software allows to control relays manually, create timers (weekly and calendar) and multivibrators, use date and time for alarms or control from command line. We provide software examples in Labview, .NET, Java, Borland C++, Python

Features:-

Datasheet - here

Power led: Yes

Relay leds: YesHigh quality

4 SPDT Relay channels - selectable by user:

- JQC-3FC/T73 DC5V (7A / 250VAC, 10A / 125VAC, 12A / 120VAC, 10A / 28VDC)
- RAS-05-15 (10A / 250VAC, 15A / 120VAC, 15A / 24VDC)

PCB parameters: FR4 / 1.5mm / two layers / metalized holes / HAL / white stamp / solder mask / extra PCB openings for better voltage isolation / doubled high voltage tracks

Power supply: from USB port Current consumption: 400 mA Chip: FT245RL

Size: 77mm x 56mm x 17mm

Supported by DRM software (Windows and Linux): Yes

Supported by Denkovi Command line tool (Windows, Linux): Yes

Android software available (low cost but very useful): Yes - **New** Software examples - here

Documentation: here

Advantages:-

High quality

Low cost

No extra power supply

Software with many functions

Control electrical devices according weekday/date/time

Create timers or pulses with our software

Applications:-

Home automation

Robotics

Alarms

Timers

Open doors and windows via

PC Aquariums applications

Additional information:-

This is relay board with 4 SPDT Relays controlled from USB port of your computer. The main purpouse of this USB relay module is to help you building projects regarding robotics and home automation (domotic). You may control differenet electrical devices like home lights, DC motors, pneumatic cylinders, lasers and so on. Each such board requires one USB port. The more USB ports you have the more such relay units you may connect and control. . The relay module outputs are controlled by FT245RL. It has 8 bit data output register (this device use only 4 of them). The usb relay card can not be controlled directly via COM port - you need to download our DRM Software to control the device. The usb relay unit can not work without PC. Only one such device can be supplyed from single USB port. If you want to supply many such devices you need USB HUB with extra power supply.

CHAPTER NO.07

SOFTWARE

Webserver:


NIELIT LAB APPLIANCES
apurv
file:///C:/Users/Mr.Apurv/Cookies/Desktop/123.html
NIELIT LAB APPLIANCES CONTROL SYSTEM
SYSTEM 1
SYSTEM 2
SYSTEM 3
SYSTEM 4
SYSTEM 5
SYSTEM 6
SYSTEM 7
SYSTEM 8
SYSTEM 9
SYSTEM 10
ENG
5:31 PM
6/1/2017

HTML CODE:

```
<html>
     <head>
          <title>NILET LAB APPLIENCES CONTROL SYSTEM</title>
     </head>
<body style="background-color:powderblue;">
          <h1><h1 style="text-align:center;"><p style="font-
size:300%;">Nielit Lab Appliances Control
System</p></p></h1></h1>
     <body>

     <!-- in the <button> tags below the ID attribute is the
value sent to the arduino -->

     <button id="13" class="led"><p style="font-
size:300%;">SYSTEM 1</p></button> <!-- button for pin 11 -->
     <button id="12" class="led"><p style="font-
size:300%;">SYSTEM 2</p></button> <!-- button for pin 12 -->
     <button id="11" class="led"><p style="font-
size:300%;">SYSTEM 3</p></button> <!-- button for pin 13 -->
     <button id="10" class="led"><p style="font-
size:300%;">SYSTEM 4</p></button> <!-- button for pin 10 -->
     <button id="09" class="led"><p style="font-
size:300%;">SYSTEM 5</p></button> <!-- button for pin 09 -->
     <button id="08" class="led"><p style="font-
size:300%;">SYSTEM 6</p></button> <!-- button for pin 08 -->
     <button id="07" class="led"><p style="font-
size:300%;">SYSTEM 7</p></button> <!-- button for pin 07 -->
     <button id="06" class="led"><p style="font-
size:300%;">SYSTEM 8</p></button> <!-- button for pin 06 -->
     <button id="05" class="led"><p style="font-
size:300%;">SYSTEM 9</p></button> <!-- button for pin 05 -->
     <button id="04" class="led"><p style="font-
size:300%;">SYSTEM 10</p></button> <!-- button for pin 04 -->

     <script src="jquery.min.js"></script>
     <script type="text/javascript">
          $(document).ready(function(){
               $(".led").click(function(){
                    var p = $(this).attr('id'); // get id value
(i.e. pin13, pin12, or pin11)
                    // send HTTP GET request to the IP address
with the parameter "pin" and value "p", then execute the function
                    $.get("http://192.168.4.1:80/", {pin:p}); //
execute get request
               });
          });
     </script>
     </body>
</html>
```

CHAPTER NO.08

REFERENCE AND BIBLIOGRAPHY

References & Bibliography:-

Websites:

- http://www.iot-playground.com
- http://www.instructables.com
- http://en.wikipedia.org
- http://www.journals.elsevier.com/easyiot

Journals & other books:

- 1 .Kusuma S M, Assistant Professor, Department of telecommunication, MSRIT, Bangalore, India. "Home Automation Using Internet of Things."
- 2.Niharika Shrotriya, Anjali Kulkarni, Priti Gadhave, International Journal of Science, Engineering and Technology Research (IJSETR), "SMART HOME USING WI-FI"
- Anushri Aware, SonaliVaidya,PriyankaAshture, VarshaGaiwal PES's Modern College of Engineering, Pune-04, International Journal of Engineering Research and General Science Volume 3, "Home Automation using Cloud Network"