IoT Based Industrial Automation
CS578
Ayush Kumar
180108008

November 6, 2020

Problem Statement

Since today is the generation of smart phones, people prefer smart work. Same goes with the industries The term automation has led to a great change in the world of industries. Some industries are fully automated while other are partially automated. In short automation has become an important term, whether at home or the industries.

1 Main Objective

The Objective of this Project is to design a system or find a method through which we can control machines manually through a distance.

2 Hardware Used

1. Arduino UNO
2. Node MCU (Wifi Module)

3. LEDs and Resistor
4. BreadBoard
5. Jumper Wires
6. LCD (16*2 Display)
7. Android Phone

3 Implemented Attributes

- 3 LEDs are used in place of machines to demonstrate this project
- Blynk app is used to give input or data(i.e. which led is to be turned ON or OFF)
- Node MCU is used to connect circuit to Blynk App thorugh wifi and it receives data from Blynk app and sends it to arduino
- Arduino UNO board is used to receive data from Node MCU and process it and glow the corresponding LEDs and display status of LEDs on LCD 16*2 display
- LCD 16*2 is used to show status of the LEDs

- Data Flow

there are 5 data flows in this project

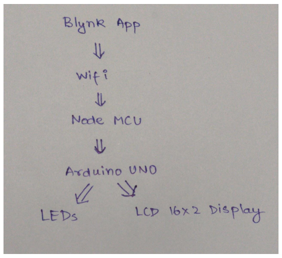

Figure 1: Data Flow

4 Hardware Description

Node MCU :

NodeMCU is a low-cost open source IoT platform. It contains wifi module (ESP8266 etc) which allows it to connect to wifi.
NODE MCU has different type pins-
-Power: Micro-USB, 3.3V, GND, Vin
-Control Pins: EN, RST
-Analog Pins Pins:A0
-GPIO Pins: GPIO1 to GPIO16
-SPI Pins: SD1, CMD, SD0,CLK
-UART Pins: TXD0, RXD0,TXD2, RXD2,TXD1
-I2C Pins: SCL, SDA

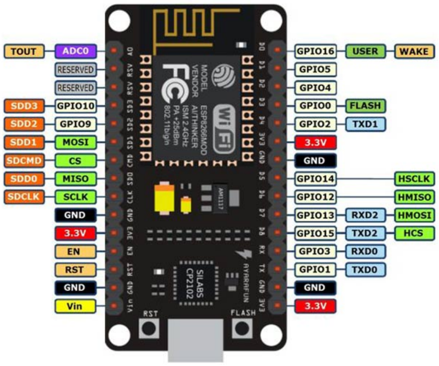

Figure 2: Node MCU pin diagram

Arduino UNO :

The Arduino Uno is an open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino.cc
Pins in arduio boards-
-14 digital input/output pins of which 6 can be used as PWM outputs)
-6 analog inputs
-6 pins related to energy/power
-a reset pin
-an analog reference pin
-a reset button
-a USB connection
-a power jack
-a 16 MHz ceramic resonator
-two ICSP header
-Atmel ATmega328 IC pins

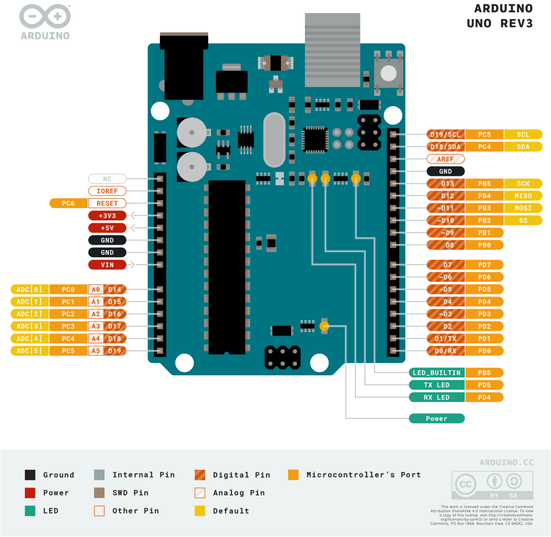

Figure 3: Arduino UNO pin diagram

LCD 16×2 display :

An LCD is an electronic display module that uses liquid crystal to produce a visible image. The 16×2 LCD display is a very basic module commonly used in DIYs and circuits. The 16×2 translates o a display 16 characters per line in 2 such lines. In this LCD each character is displayed in a 5×7 pixel matrix.

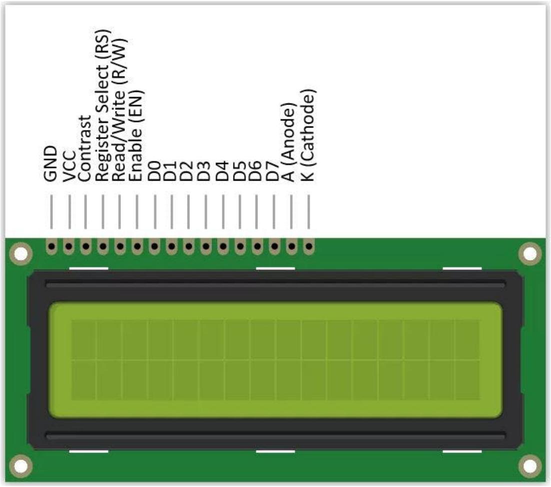

Figure 4: LCD 16×2 pin diagram

5 Connection between Hardware

Connection between Arduino UNO and Node MCU:

Arduino UNO and Node MCU are connected in a way such that serial communication between between them is done
in Arduino UNO we are using pins (8,9) as (RX,TX) respectively
in NODE MCU we are using pins (D6,D5) as (RX,TX) respectively
pin 8 of Arduino UNO is connected to D5 of NODE MCU
pin 9 of Arduino UNO is connected to D6 of NODE MCU

Connection between Arduino UNO and LCD 16×2:

LCD pins - Arduino uno pins
Vss - Gnd

Vdd - 5v
Vo - Digital pin 6
Rs - Digital pin 12
Rw - Gnd
E - Digital pin11
D4 - Digital pin 5
D5 - Digital pin 4
D6 - Digital pin 3
D7 - Digital pin 2
A - 5v
k - Gnd

Connection between Arduino UNO and LEDs:

LEDs are connected on the pin number 7,10,13 of Arduino board.

6 Code

Code uploaded in Arduino UNO

```
#include <SoftwareSerial.h>
#include <LiquidCrystal.h>

SoftwareSerial s(8,9);   //(Rx,Tx)

  // initialize the library by associating any needed LCD interface pin
  //with the arduino pin number it is connected to

 LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
 int data;


void setup() {
  // put your setup code here, to run once:
  s.begin(9600);
  analogWrite(6,90);          //to control the contrast of LCD screen
  lcd.begin(16, 2);           //to begin the display on LCD
  // Print a message to the LCD.
  lcd.print("IOT PROJECT");   //to print the Heading on LCD Screen


  Serial.begin(9600);

  pinMode(7,OUTPUT);          //LED-1 at digital pin 7 of Arduino uno board
  digitalWrite(7,LOW);
  pinMode(10,OUTPUT);         //LED-2 at digital pin 10 of Arduino uno board
  digitalWrite(10,LOW);
  pinMode(13,OUTPUT);         //LED-3 at digital pin 13 of Arduino uno board
  digitalWrite(13,LOW);
}

void loop() {
  // put your main code here, to run repeatedly:
  s.write(100);


  if(s.available()>0){
    data=s.read();            //to read the data sent from node mcu

    int data2=data%10;        //to extract the states of three LED from
    data=data/10;             //the recived data
    int data1=data%10;
    data=data/10;
    int data0=data;

    Serial.print("LED-1");
    Serial.print('\t');
    Serial.print("LED-2");
    Serial.print('\t');
    Serial.println("LED-3");


    Serial.print(data0);      //to print the states of all three LED
    Serial.print('\t');       //in serial monitor
    Serial.print(data1);
    Serial.print('\t');
    Serial.println(data2);
```

```
//in the following code State of the LED
//is assigned according to the recieved data
//and their State is displayed on the LCD screen accordingly

if(data0>=1&&data1>=1&&data2>=1){
  digitalWrite(7,HIGH);
  digitalWrite(10,HIGH);
  digitalWrite(13,HIGH);

  lcd.setCursor(0, 1);
  lcd.print("ALL ON");
}
else{
  if(data0>=1){
  digitalWrite(7,HIGH);
  lcd.setCursor(0, 1);
  lcd.print("LED1-ON");
}
if(data0==0){
  digitalWrite(7,LOW);
}
if(data1>=1){
  digitalWrite(10,HIGH);
  lcd.setCursor(0, 1);
  lcd.print("LED2-ON");
}
if(data1==0){
  digitalWrite(10,LOW);

}
if(data2>=1){
  digitalWrite(13,HIGH);
  lcd.setCursor(0, 1);
  lcd.print("LED3-ON");
}
if(data2==0){
  digitalWrite(13,LOW);
}
```

```
    if(data0==0&&data1==0&&data2==0){
      lcd.setCursor(0, 1);
      lcd.print("ALL OFF");
    }
    }
  }

}
```

Code uploaded in NODE MCU

```
#define BLYNK_PRINT Serial


#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <SoftwareSerial.h>

SoftwareSerial s(D6,D5);  //(Rx,Tx)


// You should get Auth Token in the Blynk App.
// Go to the Project Settings (nut icon).
char auth[] = "4ztfftDtp3UqqMNpZ7UbfLhyLqt2JhSu";

// Your WiFi credentials.
// Set password to "" for open networks.
char ssid[] = "Ayush";
char pass[] = "ayush123";

void setup()
{
  // Debug console
  s.begin(9600);


  Serial.begin(9600);


  Blynk.begin(auth, ssid, pass);
  // You can also specify server:
  //Blynk.begin(auth, ssid, pass, "blynk-cloud.com", 80);
  //Blynk.begin(auth, ssid, pass, IPAddress(192,168,1,100), 8080);
}

void loop()
{
  Blynk.run();

  int data0=digitalRead(D0);  //reading the input for LED-1
  int data1=digitalRead(D1);  //reading the input for LED-2
  int data2=digitalRead(D2);  //reading the input for LED-3

   //converting all three states
   //in one integer to send it to arduino board

  int data=data0*100+data1*10+data2;

  if(s.available()>0){
    s.write(data);   //sending the data to arduino board
  }

}
```

7 Sample Output

When all LEDs are OFF

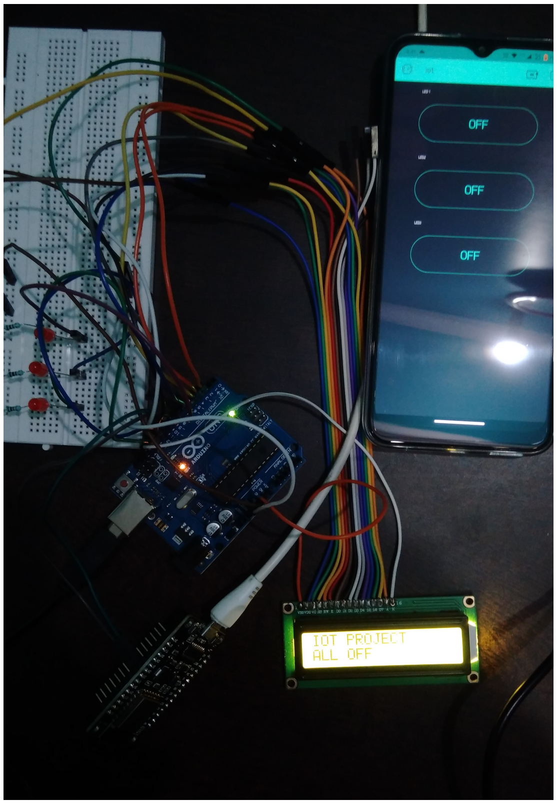

Figure 5: All LEDs OFF

When all LED 3 is ON and when All are ON

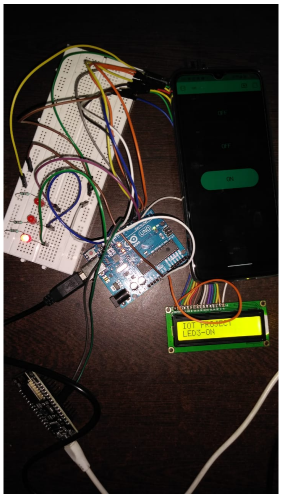

Figure 6: LED 3 is ON

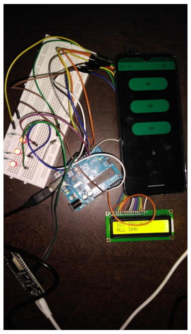

Figure 7: All LEDs ON

8 Blynk App

Blynk is a Platform with IOS and Android apps to control Arduino, Raspberry Pi and the likes over the Internet. It's a digital dashboard where you can build a graphic interface for your project by simply

dragging and dropping widgets.
In this project Blynk app is used to send input(that is which LED is truned ON and OFF) to Node MCU.

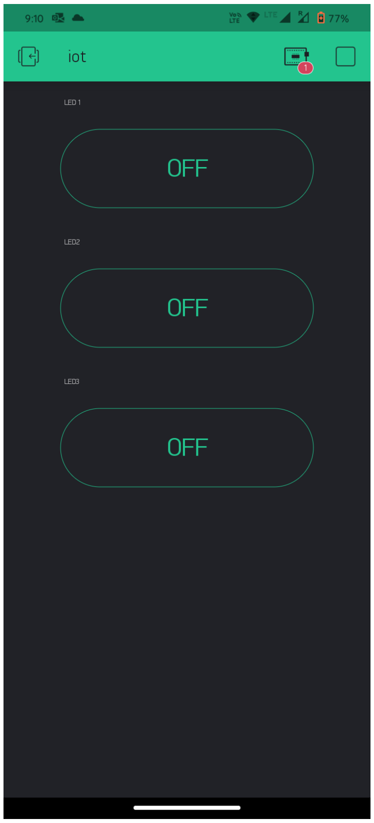

Figure 8: Project Setup

9 User Manual

in order to use this project, user should use the above codes.
upload these codes in arduino uno and Node mcu as mentioned in the code name
in the node mcu code update ssid and password of your wifi and update auth code according to your blynk app
and connect all the hardware as mentioned above in the project

10 Working

Node MCU connects the Arduino UNO to Wifi and to Blynk App. Through Blynk app we send input(i.e. which LED is to be turned ON or OFF) to NODE MCU which in turn sends it to arduino uno thorugh serial communication and then Arduino UNO board reads it and glow works accordingly and displays the status on 16*2 LCD.

11 Conclusion

Through Iot technologies we can achieve or make an system through which we can control machines and other appliances through an android phone.