[Date]

Taha Aflouk

[company name]

[Document title]

[Document subtitle]

Table of Contents

[LED Controller 2](#_Toc121655336)

[Second part 3](#_Toc121655337)

[Code 4](#_Toc121655338)

[Flowchart 6](#_Toc121655339)

[Conclusion 7](#_Toc121655340)

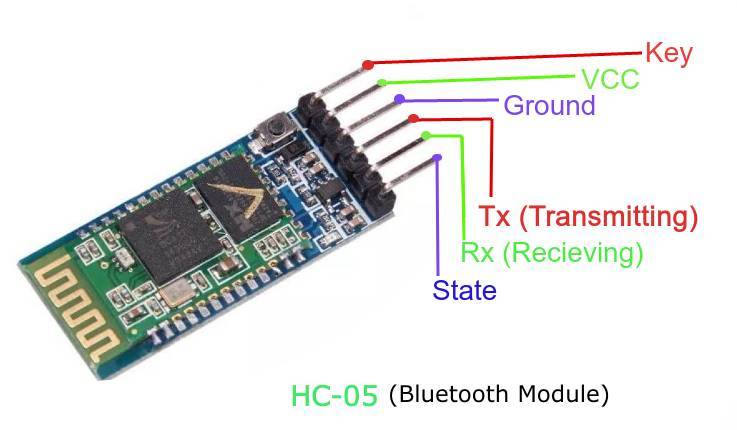
# LED Controller

The project is simply an LED controlled by an application using a phone.

The LED controlled by an Arduino device and Arduino is controlled by an application so the application connected to a Bluetooth module but the LED changes to different colours and it is depending on the user input.

The purpose of this idea is to be able to control device or a machine using another device. So now in this case we were limited in the range because we used a Bluetooth that has a limit which we cannot go over it unless we do something else which is not for this project.

The new thing in this idea is the Bluetooth component that in the picture below. This device has 5 PINs or foot and they are VCC, GND, TXD, RXD, KEY so the input voltage is from 3.6 to 6 volt and the working current is matching for 30 mA, matching the communication for 10 mA and the output power is class 2 and it has an led to let the human see if it is connected so the maximum effective distance range is around 10 meters.



# Team collaboration

My team decided to divide the project into three parts and each part contains several tasks to make the work smaller and simpler to work with so we were in contact during the work time so we created a WhatsApp group to keep all the team on the same track and see if anyone needs help whoever knows would help.

# Code

#include <SoftwareSerial.h>

#include <Wire.h>

SoftwareSerial mySerial(0,1); // RX and TX pins

// -- GLOBAL--

int PIN\_RED = 9;

int PIN\_GREEN = 10;

int PIN\_BLUE = 11;

String RGB = "";

String RGB\_Previous = "255.255.255";

String ON = "ON";

String OFF = "OFF";

boolean RGB\_Completed = false;

// -- FUNCTIONS --

void setup()

{

pinMode (PIN\_RED, OUTPUT);

pinMode (PIN\_GREEN, OUTPUT);

pinMode (PIN\_BLUE, OUTPUT);

Serial.begin(9600);

mySerial.begin(9600);

RGB.reserve(30);

}

void loop()

{

while(mySerial.available())// check if serial available

{

char ReadChar = (char)mySerial.read(); // convert it to char

if(ReadChar == ')') // check if user finishes from entering data

{

RGB\_Completed = true; // set the rgb to true

}else{

RGB += ReadChar; // add the char to the rgb var

}

}

if(RGB\_Completed) // if the rgb true

{ // print them to the screen

Serial.print("RGB:");

Serial.print(RGB);

Serial.print(" PreRGB:");

Serial.println(RGB\_Previous);

if(RGB==ON) // if they enter on

{

RGB = RGB\_Previous; // set rgb to the previous

Light\_RGB\_LED(); // call the function

}

else if(RGB==OFF) // if they enter off

{

RGB = "0.0.0"; // set rgb values to zeros

Light\_RGB\_LED();

}else{

Light\_RGB\_LED(); //. Call the function

RGB\_Previous = RGB;

}

RGB = "";

RGB\_Completed = false;

}

}

void Light\_RGB\_LED()

{

int SP1 = RGB.indexOf(' ');

int SP2 = RGB.indexOf(' ', SP1+1);

int SP3 = RGB.indexOf(' ', SP2+1);

String R = RGB.substring(0, SP1);

String G = RGB.substring(SP1+1, SP2);

String B = RGB.substring(SP2+1, SP3);

Serial.print("R=");

Serial.println( constrain(R.toInt(),0,255));

Serial.print("G=");

Serial.println(constrain(G.toInt(),0,255));

Serial.print("B=");

Serial.println( constrain(B.toInt(),0,255));

analogWrite(PIN\_RED, (R.toInt()));//comment if colors are inverted

analogWrite(PIN\_GREEN, (G.toInt()));//and uncomment part below.

analogWrite(PIN\_BLUE, (B.toInt()));

// analogWrite(PIN\_RED, (255-R.toInt()));//uncomment if colors are inverted

// analogWrite(PIN\_GREEN, (255-G.toInt()));//and comment above part.

// analogWrite(PIN\_BLUE, (255-B.toInt()));

}

# Flowchart

Diagram

Description automatically generated

# Conclusion

To be able to communicate with the Arduino board we need to use some sort of communication like ZigBee, internet, or Wi-Fi etc… used to send and receive data and do some actions based on the data to create a system or solve a specific problem.