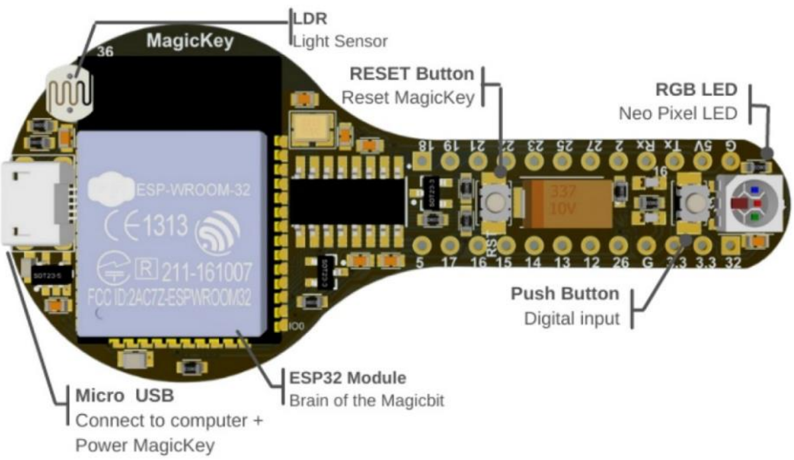
Magic Key භාවිතයෙන් 13 ශ්‍රේණිය අ.පො.ස උ/පෙළ තොරතුරු සන්නිවේදන තාක්ෂණය විෂය මාලාවෙහි සඳහන් ක්‍රියාකාරකම් සිදු කිරීම.

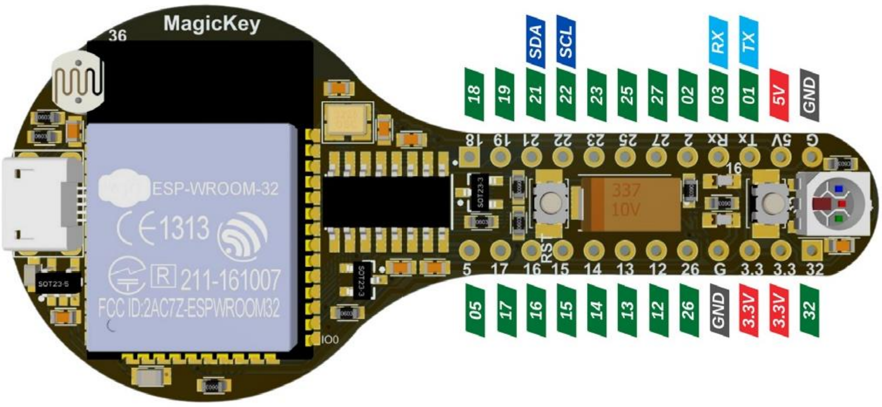
Magic Key යනු කුමක්ද?

Magic key යනු ආලෝක විමෝචක දියෝඩයක් (Light Emitting Diode – LED), Push Button එකක්, ආලෝක සංවේදී ප්‍රතිරෝධකයක් (Light Depending Resistor – LDR) සහ RGB Neo Pixel LED සහිත ESP32 මත පදනම් වූ සංවර්ධන පුවරුවකි (Development Board). මෙය breadboard එකක සවි කළ හැකි පරිදි සකසා ඇත. එමෙන්ම, MagicCode, Magicblocks, Arduino සහ Python යන සියළුම මෘදුකාංග හරහා ක්‍රමලේඛනය කළ හැකිය.

- **Magic Key පිරිසැලසුම (Layout)**



- **Magic Key තුඩු සැකැස්ම (Pinout)**



Red LED	16
LDR	36
Button	35
RGB LED	4

- All pins support PWM except pin 5
- 1,3,5,16,17,18,21,22 pins doesn't support analog in function
- When WiFi is enabled, Analog in working only for pin 32 to 39
- Max input voltage for any pin is 3.3V / battery connector is 4.2V
- 2,4,5,12,15 are strapping pins.

සැ.යු. – මෙහි සඳහන් සියළු ක්‍රියාකාරකම් සඳහා ආර්ඩිනෝ (Arduino) මෘදුකාංගය භාවිතා කර ඇත.

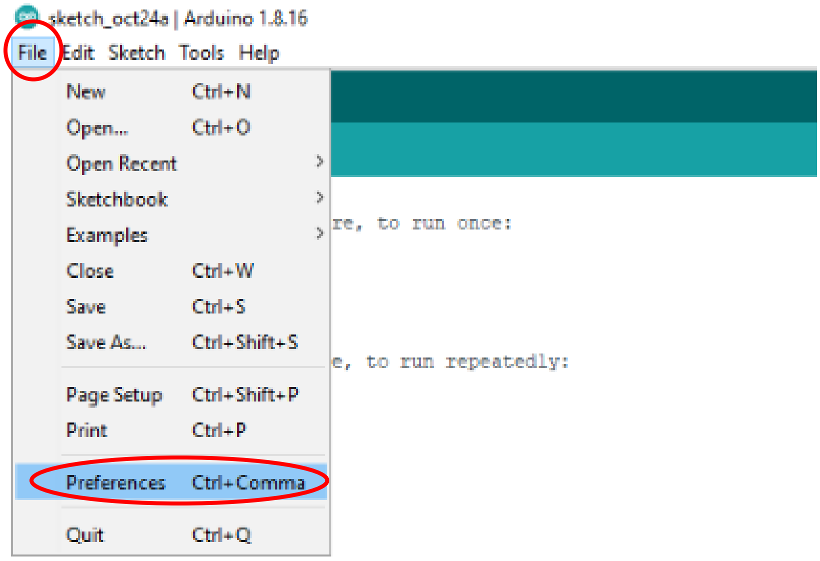
- **Magic Key** ආර්ඩිනෝ (Arduino) මෘදුකාංගයට එකතු කිරීම.

❖ පියවර 01

ඔබගේ පරිගණකයට Arduino මෘදුකාංගය බාගත (Download) කර ස්ථාපනය (Install) කරගන්න.

❖ පියවර 02

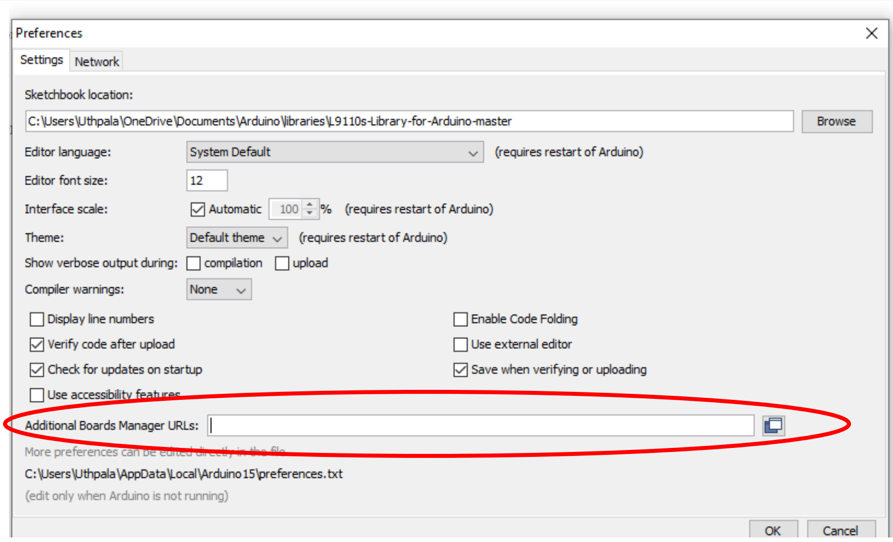
Arduino මෘදුකාංගය ආරම්භ කර, “File” මෙනුවෙහි ඇති “Preferences Window” එක විවෘත කරන්න



❖ පියවර 03

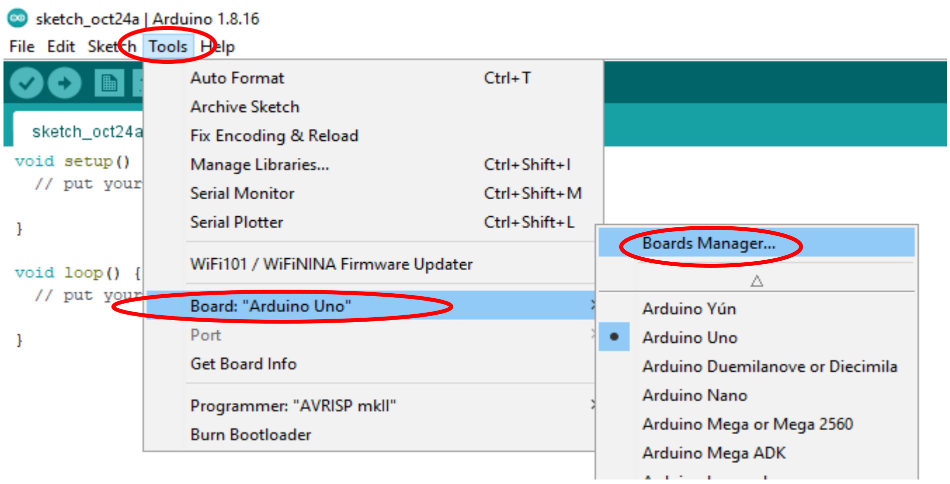
“Additional Board Manager URLs” ක්ෂේත්‍රයට පහත මුදා හැරීමේ සබැඳිය (Release link) එක ඇතුලත් කරන්න.

මුදා හැරීමේ සබැඳිය (Release link) : - http://bit.ly/3O7Ifww



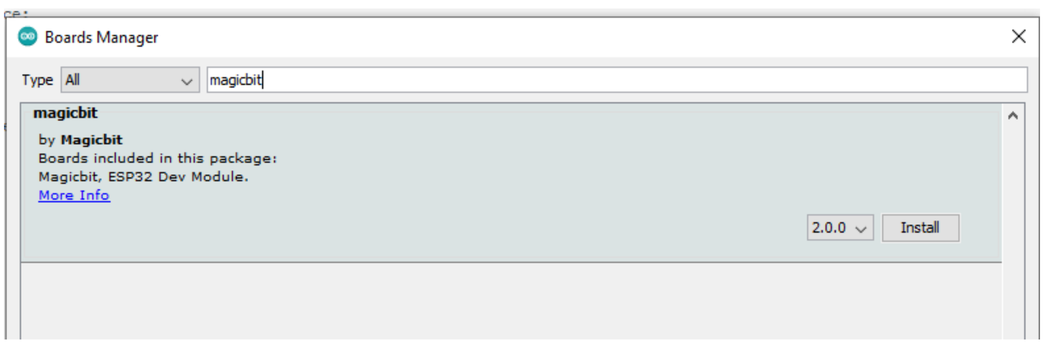
❖ පියවර 04

“Tools” මෙනුවෙහි අති “Boards Manager” වින්ඩෝව විවෘත කරන්න.



❖ පියවර 05

එහිදී “Magicbit” ලෙස search කර , එය ස්ථාපනය කරන්න. මේ සඳහා ඔබගේ පරිගණකය අන්තර්ජාලයට සම්බන්ධ කර තිබිය යුතුය.

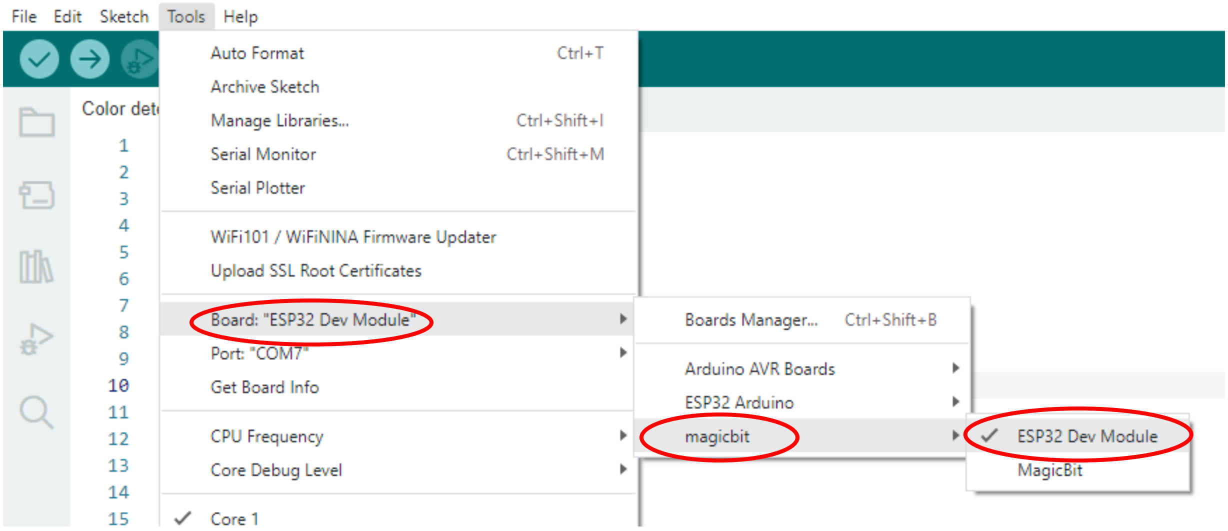


❖ පියවර 06

USB cable එක යොදාගනිමින් Magic Key එක පරිගණකයට සම්බන්ධ කරන්න

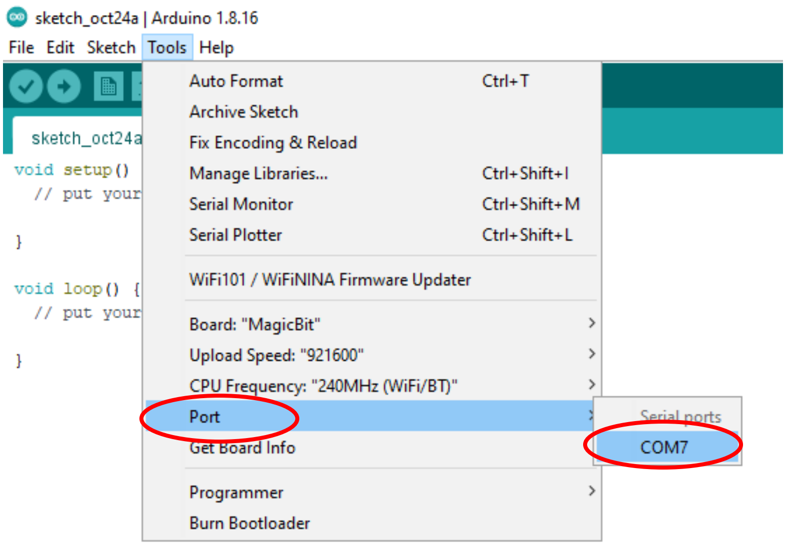
❖ පියවර 07

දැන් “Tools” මෙනුවෙහි ඇති “Boards Manager” වින්ඩෝවෙහි Magicbit හා ඉන්පසු ESP32 Dev Module තෝරන්න.



❖ පියවර 08

දැන් පහත පරිදි Magic key එක පරිගණකයට සම්බන්ධ කර ඇති පෝර්ට් (USB Port) එක තෝරන්න



පෝර්ට් එක දිස්නොවන්නේ නම්, පහත Drivers ස්ථාපනය (Installation) පියවර අනුගමනය කරන්න.

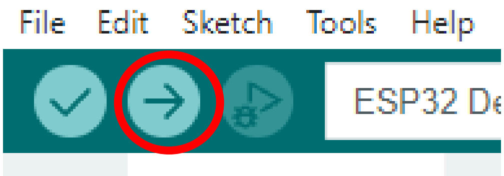
❖ Drivers ස්ථාපනය (Installation)

Magicbit සතුව USB -Serial පරිවර්තකයක් ලෙස CH340 චිප් එකක් ඇති අතර එම driver එක දැනටමත් Arduino IDE පැකේජයට අතුලත් කර ඇත. ඉහත දක්වා ඇති පරිදි Arduino හි පෝර්ට් එක දිස්නොවන්නේ නම්, පහත දක්වා ඇති සබැඳියෙන් (link) එම driver එක ස්ථාපනය කරගන්න.

https://magicbit.cc/uploader/downloads/CH341SER.EXE

❖ පියවර 09

දැන් ඔබේ ක්‍රියාකරකමට අදාළ කේතය (Code) සකසා එය Magic Key වෙත උඩුගත (Upload) කරන්න. ඒ සඳහා පහත අයිකනය (Icon) මත click කරන්න



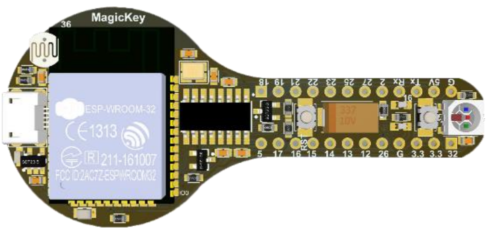
1. නිමිලීමේ පද්ධතිය (Blinker system)

- **කාර්ය**

ආලෝක විමෝචක දියෝඩයක් (Light Emitting Diode – LED) වරින් වර නිමි නිමී දැල්වීම (Blinking) මෙම පද්ධතිය මගින් සිදු කරයි.

- **අවශ්‍ය සංරචක**

  - 1 x Magic Key ක්ෂුද්‍ර පාලක මත පදනම් වූ සංවර්ධන පුවරුව (නිමිලීමේ පද්ධතිය පාලනය කිරීමට)
  - 1 x LED (වරින්වර ආලෝකය විමෝචනයට)
  - 1 x 220 Ω ප්‍රතිරෝධකයක් (LED ය තුළින් ගලන ධාරාව අවශ්‍ය පමණට සීමා කිරීමට)

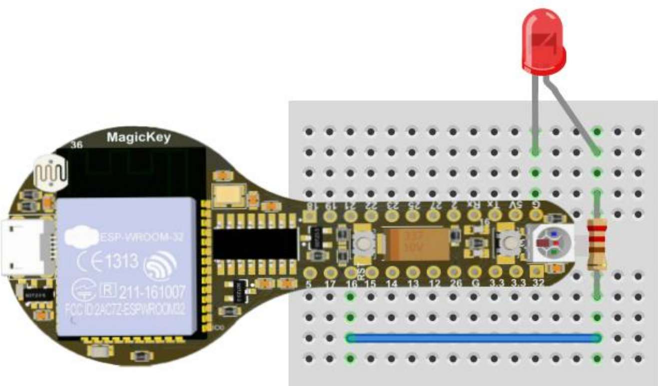


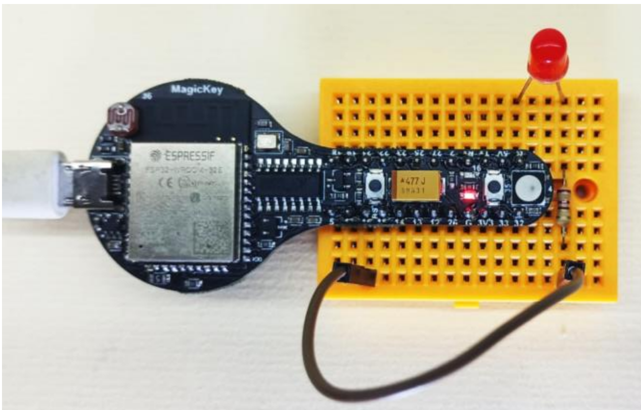
Magic Key

LED

220 Ω ප්‍රතිරෝධකය

- **පරිපථ සටහන (Circuit Diagram)**





- **ආර්ඩිනෝ කේතය (Arduino Code)**

```
const int ledpin = 16;

void setup() {
pinMode(ledpin,OUTPUT);
}

void loop() {
digitalWrite(ledpin,HIGH);
delay(1000);
digitalWrite(ledpin,LOW);
delay(1000);
}
```

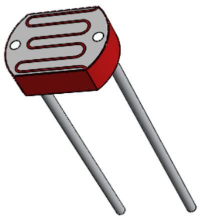
2. ස්වයංක්‍රීය ආලෝක පද්ධතිය (Automatic Light System)

- **කාර්ය**

ආලෝක සංවේදී ප්‍රතිරෝධකයක් (Light Depending Resistor – LDR) භාවිතා කරමින් ස්ථානික ආලෝක තීව්‍රතාව (Ambient Light Intensity) මත පදනම්ව ආලෝක විමෝචක දියෝඩයක් දැල්වීම (On) හෝ නිවීම (Off) මෙම පද්ධතිය මගින් සිදු කරයි.

- **අවශ්‍ය සංරචක**

  - 1 x Magic Key ක්ෂුද්‍ර පාලක මත පදනම් වූ සංවර්ධන පුවරුව (ස්වයංක්‍රීය ආලෝක පද්ධතිය පාලනය කිරීමට)
  - 1 x LED (ආලෝකය විමෝචනයට)
  - 1 x 220 Ω ප්‍රතිරෝධකයක් (LED ය තුළින් ගලන ධාරාව අවශ්‍ය පමණට සීමා කිරීමට)
  - 1 x 10 kΩ ප්‍රතිරෝධකයක් (භූගත මාර්ගයකට ගලා යන ධාරාව සීමා කිරීමට)
  - 1 x ආලෝක සංවේදී ප්‍රතිරෝධකයක් (LDR)

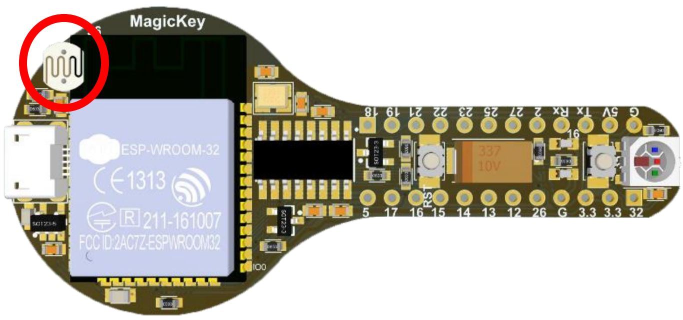


LDR

10 kΩ ප්‍රතිරෝධකය

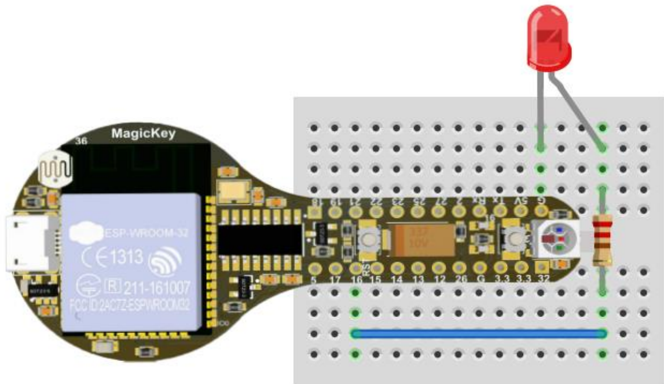
සැ..යු. – Magic Key පුවරුවෙහි LDR එකක් සවි කර ඇති බැවින් (pin - 36) වෙනත් LDR එකක් බාහිරව සහ 10 kΩ ප්‍රතිරෝධකය භාවිතා කිරීම අවශ්‍ය නොවේ.

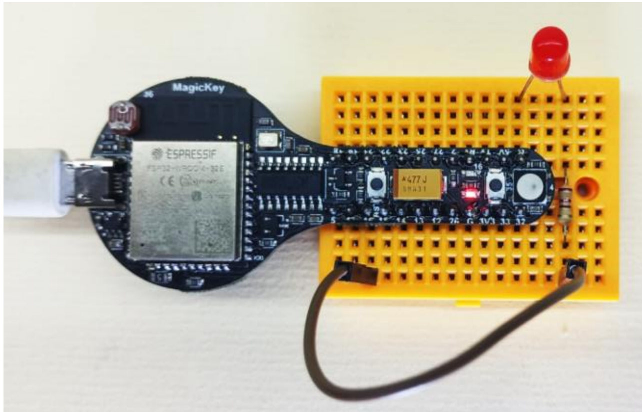
බාහිරව LDR එකක් භාවිතා කරන්නේ නම් (10 kΩ ප්‍රතිරෝධකය ද සමග) අදාළ pin අංකය ආර්ඩිනෝ කේතයේ සඳහන් කරන්න. මීට අදාළ පරිපථ සටහන ද පහත දක්වා ඇත.



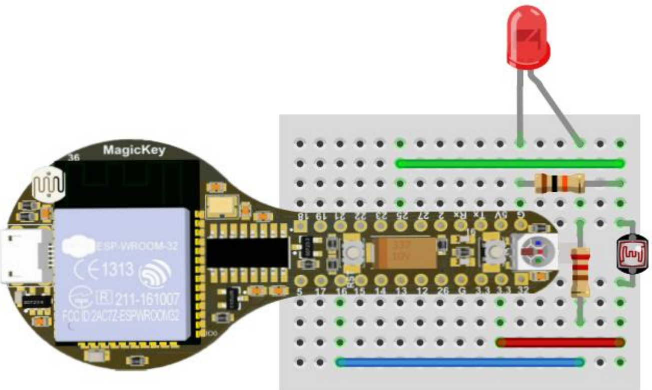
- **පරිපථ සටහන (Circuit Diagram)**

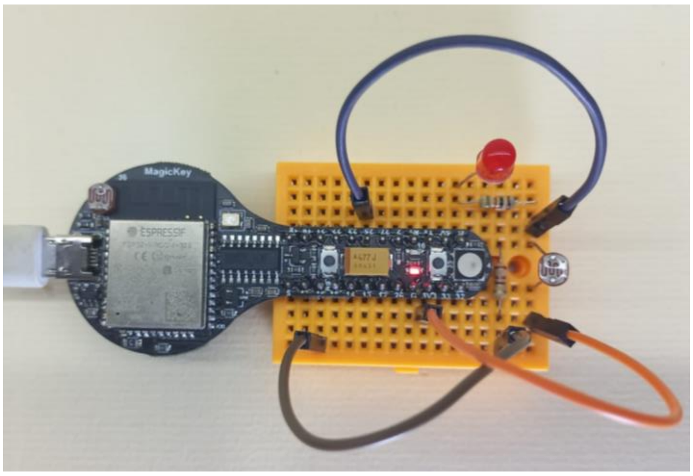
❖ Magic Key හි සවි කර ඇති LDR එක භාවිතයෙන්





❖ වෙනත් LDR එකක් බාහිරව සම්බන්ධ කිරීමෙන්





- **සංකල්පය අවබෝධ කරගැනීම සඳහා පරීක්ෂණ කේතය (Test Code)**

LDR එකෙන් ලබා ගන්නා ආලෝක තීව්‍රතා අගයන් අනුක්‍රමික මොනිටරයක් (Serial Monitor) හරහා ලබා ගෙන, ඒ අනුව ගැළපෙන ආලෝක තීව්‍රතා අගයක් එළිපත්ත අගය (Threshold Value) ලෙස තෝරා ගැනීම සඳහා මෙම පරීක්ෂණ කේතය භාවිතා කළ හැකිය.

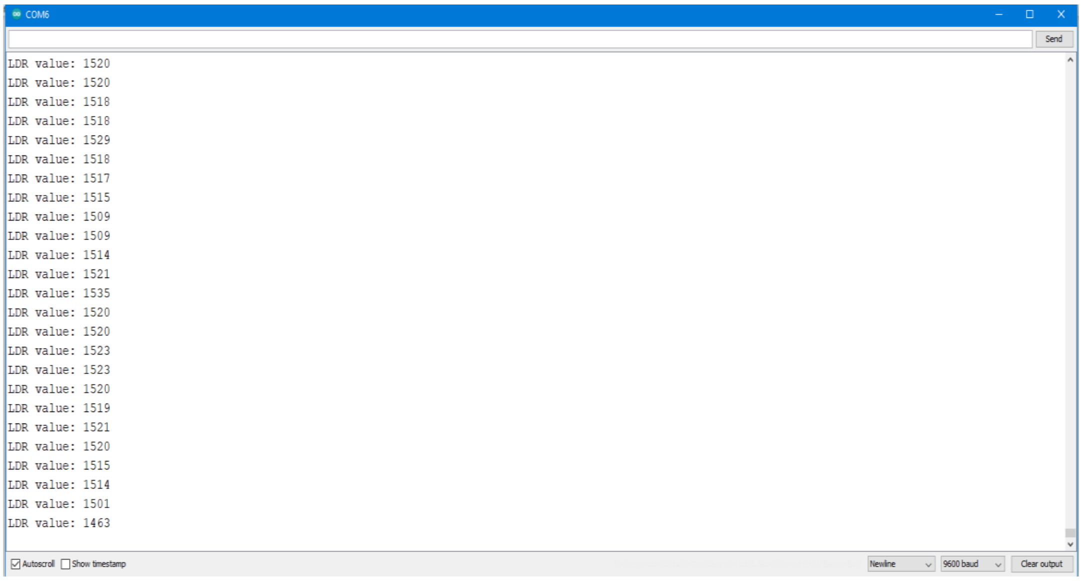
සැ..යු. – පහත කේතයේ "//" සලකුණු වලින් දක්වා ඇත්තේ කේතය තේරුම් ගැනීමේ පහසුව සඳහා අදහස් (comments) වේ. මේවායින් කේතයේ ක්‍රියාකාරීත්වයට කිසිදු බලපෑමක් නොමැත.

```
const int ldrPin = 36 ; // pin number where LDR is connected

void setup() {
Serial.begin (9600) ; // Initialization  of the serial monitor (baud rate)
}
void loop() {
int sensorValue = analogRead(ldrPin);  // getting the reading from LDR
Serial.print("LDRValue  :") ; // Print the label "LDR Value"
Serial.println(sensorValue);  // print the value from the senor in the serial monitor
delay (1000); // wait for 1 second ( 1000 milliseconds)
}
```

Serial Monitor හි baud rate එක කේතයේ සඳහන් baud rate එක ම දැයි සලකා බලන්න.

මෙහි දිස්වෙන ආලෝක තීව්‍රතා අගයන් මගින්, ස්වයංක්‍රීය ආලෝක පද්ධතිය සඳහා ගැළපෙන threshold value එක තීරණය කර ගත හැකිය.



ඔබ මේ මොහොතේ සිටින පරිසරයට අදාළ ආලෝක තීව්‍රතා අගයන් serial monitor හි දිස්වෙන අතර එයින් ඔබගේ ස්වයංක්‍රීය ආලෝක පද්ධතියට ගැළපෙන threshold value එක තෝරා ගන්න.

- **ආර්ඩිනෝ කේතය (Arduino Code)**

```
const int ldrPin = 36;  // pin number where LDR is connected
const int ledPin = 16;  // pin number where LED is connected


void setup()  {
pinMode(ledPin,OUTPUT)  ;
}
void loop() {
int sensorValue  = analogRead(ldrPin);  // getting the reading from LDR
if (sensorValue  < 1000 ) { // compare the value  from the LDR – Setting the Condition
digitalWrite(ledPin, HIGH); // Turn ON the LED if the  condition is satisfied
}
else{
digitalWrite(ledPin, LOW);
}
delay(1000);  // wait for 1 second ( 1000 milliseconds)
}
```

3. ස්වයංක්‍රීය පංකා පද්ධතිය (Automatic Fan System)

- **කාර්ය**

  යෝජකයක (Actuator) ක්‍රියාවේ යෙදවීම සහ පරිසරයෙන් ආදානයක් ලබගෙන, එම ආදානය අනුව යෝජකයක් ක්‍රියාත්මක කිරීම ඉහත නිහිත පද්ධති මගින් සිදු කරන ලදි.
  දැන්, මෙම පද්ධතියෙන් කාමර උෂ්ණත්වය අනුව, ස්වයංක්‍රීයව විදුලි පංකාවක මෝටරයක් ක්‍රියාත්මක වීම හෝ ක්‍රියාවිරහිත වීම සඳහා ස්වයංක්‍රීය පංකා පද්ධතියක් නිර්මාණය කිරීම සිදු කරයි.

- **අවශ්‍ය සංරචක**

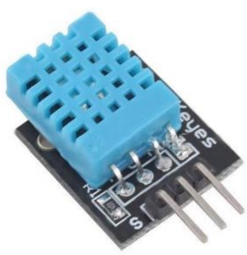
  - 1 x Magic Key ක්ෂුද්‍ර පාලක මත පදනම් වූ සංවර්ධන පුවරුව (ස්වයංක්‍රීය ආලෝක පද්ධතිය පාලනය කිරීමට)
  - 1 x 9V සරල ධාරා මෝටරයක් (DC motor) (පංකාව ක්‍රියාවේ යෙදවීමට)
  - 1 x DHT11 (උෂ්ණත්වය (Temperature) සහ වාතයේ ආද්‍රතාවය (Humidity) මැණිය හැකි සංවේදකයක්)
  - 1 x BC547 ට්‍රාන්සිස්ටරයක් (මෝටරය ක්‍රියාත්මක කිරීම සඳහා ප්‍රමාණවත් ධාරාවක් ලබාගැනීමට)

සැ..යු. – දැනට වෙළඳපොළේ ඇති LM35 උෂ්ණත්ව සංවේදක වල ක්‍රියාකාරීත්වය නිවැරදි නොවන බැවින් එයට ආදේශකයක් ලෙස DHT11 උෂ්ණත්ව සහ ආද්‍රතා සංවේදකය භාවිතා කර ඇත.

BC547 ට්‍රාන්සිස්ටරය මගින් ලබා ගත හැකි උපරිම ධාරාව 100mA වන බැවින්, DC මෝටරය ක්‍රියාත්මක කිරීමට ඊට වැඩි ධාරාවක් අවශ්‍ය වන බැවින්, මෙම ක්‍රියාකාර්කම සඳහා D400 ට්‍රාන්සිස්ටරය භාවිතා කර ඇත.

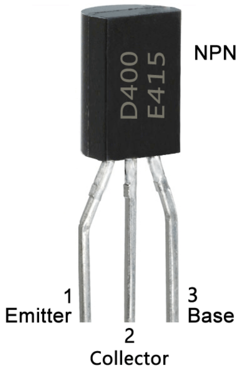
  - 1 x 220Ω ප්‍රතිරෝධකයක් (ට්‍රාන්සිස්ටරයට ගලායන ධාරාව සීමා කිරීමට)
  - 1 x 1N4001 සෘජුකාරක ඩයෝඩයක් (Rectifier Diode) (මෝටරය තුළ සිදුවන ස්වයං ප්‍රේරණය මගින් ට්‍රාන්සිස්ටරය ආරක්ෂා කර ගැනීමට)

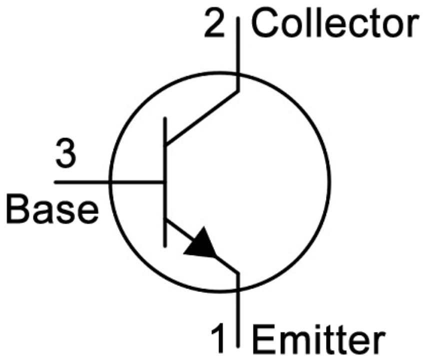
DC Motor



DHT11

220Ω ප්‍රතිරෝධකය

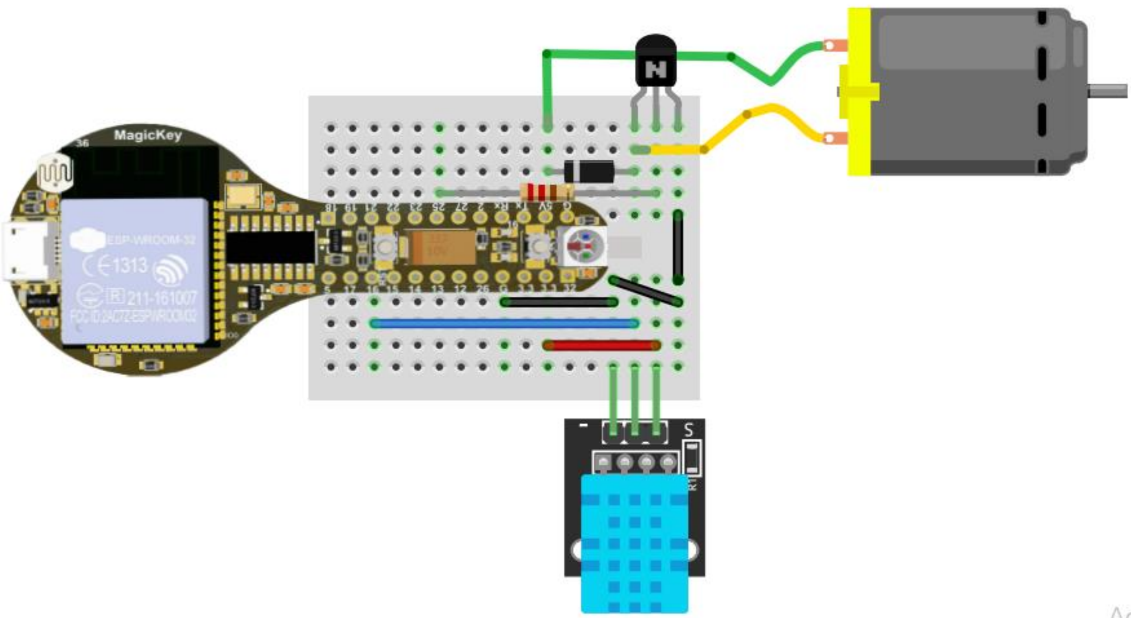


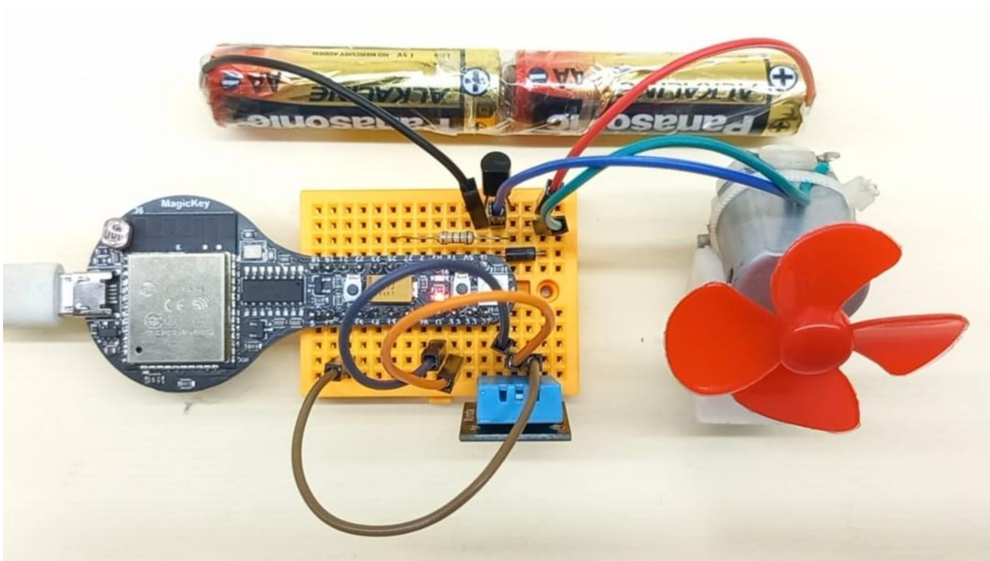


1N4001 සෘජුකාරක ඩයෝඩය

D400 ට්‍රාන්සිස්ටරය

- **පරිපථ සටහන (Circuit Diagram)**





- **ආර්ඩිනෝ කේතය (Arduino Code)**

```
#include <ESP32Servo.h>
#include "DHT.h"

#define DHTTYPE  DHT11
#define DHTPIN  16

DHT dht( DHTPIN  ,DHTTYPE);
const int motorPin = 25;

void setup() {
pinMode  (motorPin,OUTPUT);
dht.begin();
}

void loop() {
float hum = dht.readHumidity();
float temp = dht.readTemperature();
Serial.print  ("temp.");
Serial.println  (temp);
if (temp > 30 ) {
digitalWrite  (motorPin,HIGH);
} else{
digitalWrite  (motorPin,LOW);
}
}
```

4. දොර සංවේදී සීනු පද්ධතිය (Door Alarm System)

- **කාර්ය**

  මෙහිදී දොරක් විවෘත කරන විට අනතුරු ඇඟවීමක් සිදු කරන දොර සංවේදී සීනු පද්ධතියක් ගොඩනැගීම සිදු කරයි. මෙහිදී ආදානය ලබා ගැනීම සඳහා සංවේදකයක් ලෙස චුම්බක යතුරක් භාවිත කරයි.

- **අවශ්‍ය සංරචක**

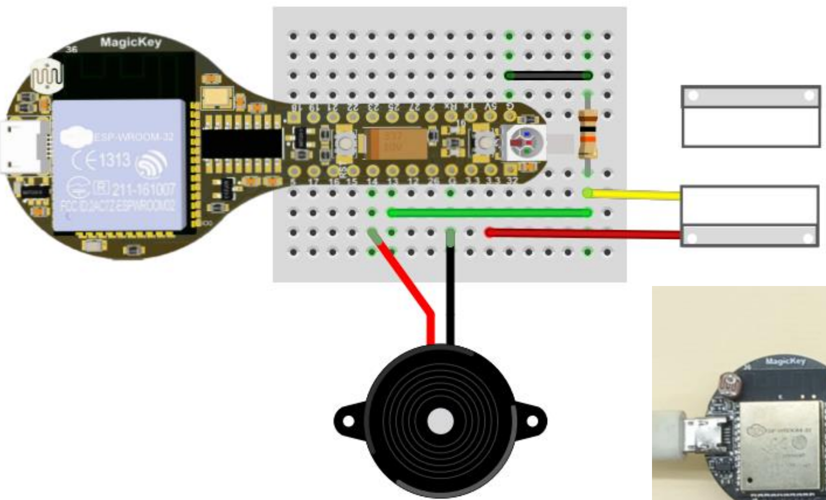
  - 1 x Magic Key ක්ෂුද්‍ර පාලක මත පදනම් වූ සංවර්ධන පුවරුව (ස්වයංක්‍රීය ආලෝක පද්ධතිය පාලනය කිරීමට)
  - 1 x පීඩ ගොමුව (Piezo Buzzer) (ප්‍රතිදානය ලෙස ශබ්දයක් නිකුත් කිරීමට)
  - 1 x චුම්බක යතුර (Reed Switch) (ආදානය ලබා ගැනීමට)
  - 1 x 10 kΩ ප්‍රතිරෝධකයක් (වෝල්ටීයතාවය පහළ හෙලා, භූගත මාර්ගයකට ගලා යන ධාරාව සීමා කිරීමට)

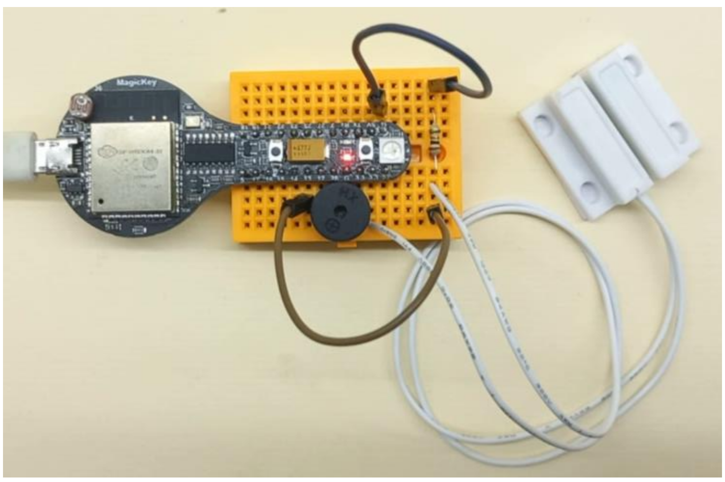
පීඩ ගොමුව (Piezo Buzzer)

චුම්බක යතුර (Reed Switch)

10 KΩ ප්‍රතිරෝධකය

- **පරිපථ සටහන (Circuit Diagram)**





- **ආර්ඩිනෝ කේතය (Arduino Code)**

```
const int switchPin = 13;

const int buzzerPin = 26;

#include <ESP32Servo.h>

void setup (){

pinMode (switchPin , INPUT);

pinMode (buzzerPin , OUTPUT);

}


void loop(){

int switchstate = digitalRead(switchPin);

if (switchstate == LOW){

tone(buzzerPin,3500);

}else{

noTone(buzzerPin);

}

}
```