

The Grove - Light sensor integrates a photo-resistor(light dependent resistor) to detect the intensity of light. The resistance of photo-resistor decreases when the intensity of light increases. A dual OpAmp chip LM358 on board produces voltage corresponding to intensity of light(i.e. based on resistance value). The output signal is analog value, the brighter the light is, the larger the value.

This module can be used to build a light controlled switch i.e. switch off lights during day time and switch on lights during night time.



### Warning

The light sensor value only reflects the approximated trend of the intensity of light, it DOES NOT represent the exact Lumen.

**Get One Now** 

[<https://www.seeedstudio.com/Grove-Light-Sensor-v1.2-p-2727.html>]

## Version

Product Version	Changes	Released Date
Grove - Light Sensor 1.0	Initial	Apr 28 2013
Grove - Light Sensor(P)	Move Grove connector to back side	May 15 2014
Grove - Light Sensor(P) V1.1	Replace photoresistor-5528 with LS06-S Vs.Grove - Light Sensor(P)	Dec 31 2015
Grove - Light Sensor 1.2	Replace photoresistor-5528 with LS06-S Vs.Grove - Light Sensor 1.0	Jan 20 2016

## Features

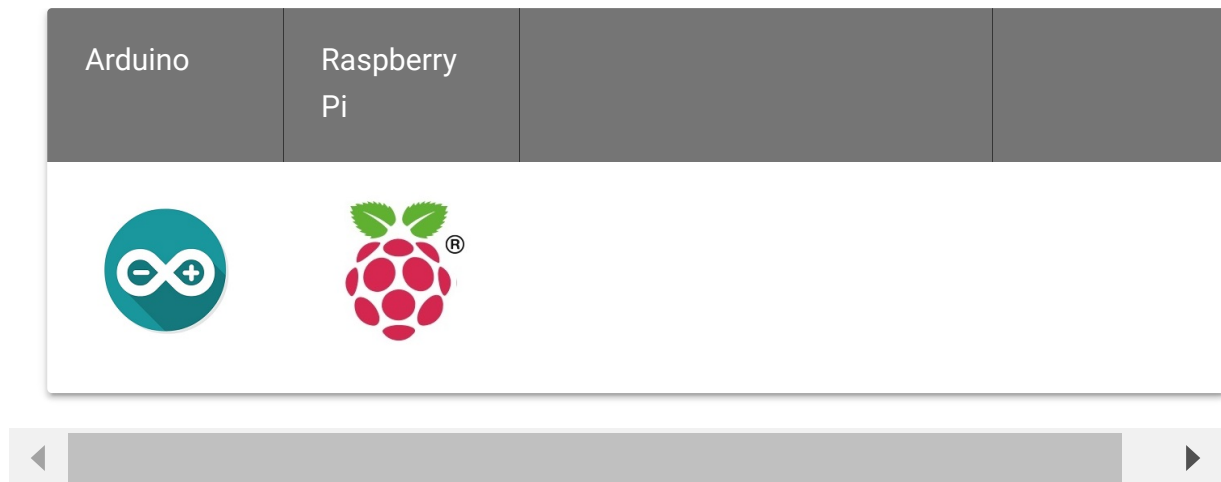
- Analog value output
- High reliability and sensibility
- Small footprint
- Recognize wider spectrum



### Tip

More details about Grove modules please refer to [Grove System](https://wiki.seeedstudio.com/Grove_System/)  
[[https://wiki.seeedstudio.com/Grove\\_System/](https://wiki.seeedstudio.com/Grove_System/)]

## Platform Support



### Caution

The platforms mentioned above as supported is/are an indication of the module's software or theoretical compatibility. We only provide software library or code examples for Arduino platform in most cases. It is not possible to provide software library / demo code for all possible MCU platforms. Hence, users have to write their own software library.

## Specification

Item	Value
Operating voltage	3~5V
Operating current	0.5~3 mA
Response time	20-30 milliseconds
Peak Wavelength	540 nm
Weight	4 g

## Getting Started

### Play With Arduino

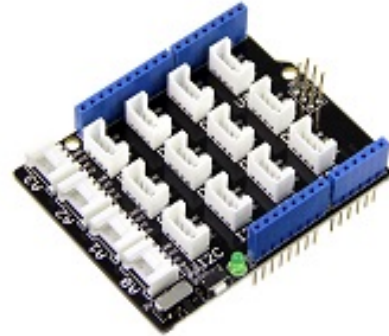
#### Hardware

- Step 1. Prepare the below stuffs:

Seeeduino V4



Base Shield



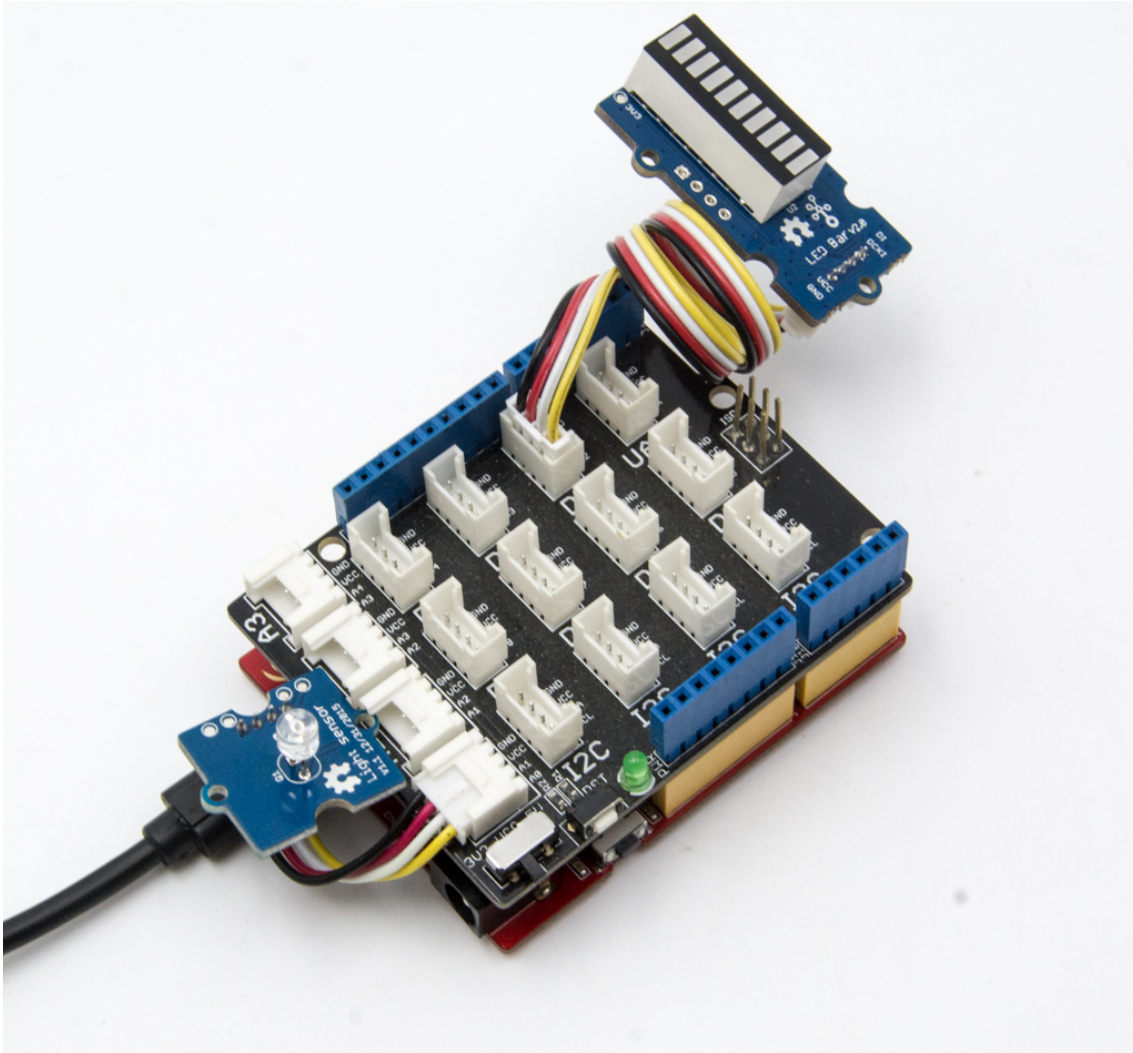
[Get One Now](#)

[<https://www.seeedstudio.com/Seeeduino-V4.2-p-2517.html>]

[Get One Now](#)

[<https://www.seeedstudio.com/Base-Shield-V2-p-1378.html>]

- Step 2. Connect Grove-Light Sensor to port A0 of Grove-Base Shield.
- Step 3. Connect Grove-Led Bar to port D2 of Grove-Base Shield.
- Step 4. Plug Grove - Base Shield into Seeeduino.
- Step 5. Connect Seeeduino to PC through a USB cable.

**Note**

If we don't have Grove Base Shield, We also can directly connect Grove-Light Sensor to Seeeduino as below.

Seeeduino	Grove-Light Sensor
5V	Red
GND	Black
Not Conencted	White
A0	Yellow

Seeeduino	Grove-Led Bar
5V	Red
GND	Black
D3	White
D2	Yellow

## Software

- Step 1. Download the [Grove-LED Bar Library](https://github.com/Seeed-Studio/Grove_LED_Bar/archive/master.zip) [https://github.com/Seeed-Studio/Grove\_LED\_Bar/archive/master.zip] from Github.
- Step 2. Refer [How to install library](https://wiki.seeedstudio.com/How_to_install_Arduino_Library) [https://wiki.seeedstudio.com/How\_to\_install\_Arduino\_Library] to install library for Seeeduino.
- Step 3. Copy the code into Seeeduino IDE and upload.



```
1  #include <Grove_LED_Bar.h>
2
3  Grove_LED_Bar bar(3, 2, 0); // Clock pin, Data pin, Ori
4
5  void setup()
6  {
7      // nothing to initialize
8      bar.begin();
9      bar.setGreenToRed(true);
10 }
11
12 void loop()
13 {
14
15     int value = analogRead(A0);
16     value = map(value, 0, 800, 0, 10);
17
18     bar.setLevel(value);
19     delay(100);
20 }
```

- Step 2. The Led bar will change base on light.

## Play with Codecraft

### Hardware

**Step 1.** Connect a Grove - Light Sensor to port A0 of a Base Shield.

**Step 2.** Plug the Base Shield to your Seeduino/Arduino.

**Step 3.** Link Seeduino/Arduino to your PC via an USB cable.

### Software

**Step 1.** Open [Codecraft](https://ide.chmakered.com/) [https://ide.chmakered.com/], add Arduino support, and drag a main procedure to working area.



**Note**

If this is your first time using Codecraft, see also [Guide for Codecraft using Arduino](#)

[[https://wiki.seeedstudio.com/Guide\\_for\\_Codecraft\\_using\\_Arduino/](https://wiki.seeedstudio.com/Guide_for_Codecraft_using_Arduino/)].

**Step 2.** Drag blocks as picture below or open the cdc file which can be downloaded at the end of this page.



Upload the program to your Arduino/Seeeduino.


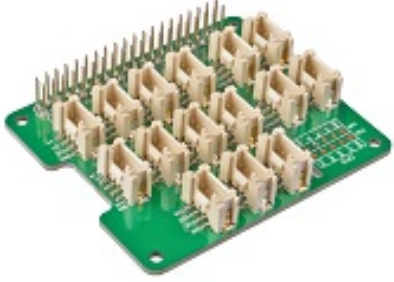
**Success**

When the code finishes uploaded, you will see the brightnedd value displayed in the Serial Monitor.

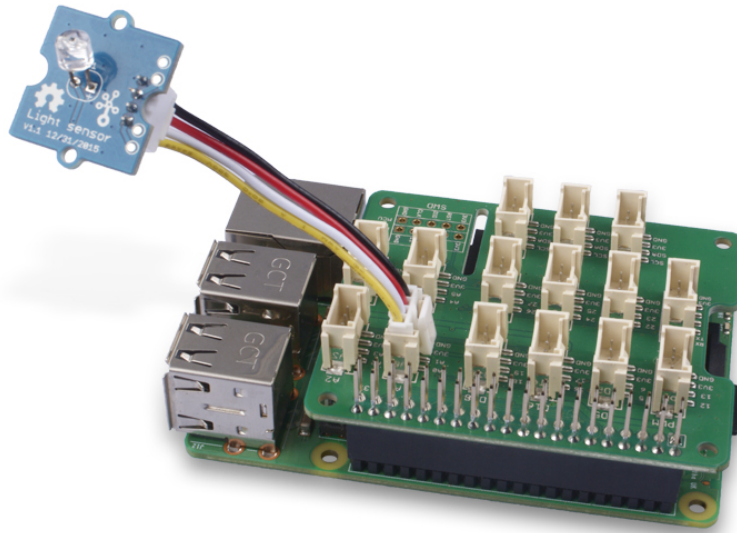
## Play With Raspberry Pi (With Grove Base Hat for Raspberry Pi)

### Hardware

- **Step 1.** Things used in this project:

Raspberry pi	Grove Base Hat for RasPi
	
<p><a href="https://www.seeedstudio.com/Raspberry-Pi-3-Model-B-p-2625.html">Get ONE Now</a> [https://www.seeedstudio.com/Raspberry-Pi-3-Model-B-p-2625.html]</p>	<p><a href="https://www.seeedstudio.com/Grove-Base-Hat-for-Raspberry-Pi-p-3186.html">Get ONE Now</a> [https://www.seeedstudio.com/Grove-Base-Hat-for-Raspberry-Pi-p-3186.html]</p>

- **Step 2.** Plug the Grove Base Hat into Raspberry.
- **Step 3.** Connect the light sensor to port A0 of the Base Hat.
- **Step 4.** Connect the Raspberry Pi to PC through USB cable.



### Note

For step 3 you are able to connect the light sensor to **any Analog Port** but make sure you change the command with the corresponding port number.

## Software



### Attention

If you are using **Raspberry Pi with Raspberrypi OS >= Bullseye**, you have to use this command line **only with Python3**.

- **Step 1.** Follow [Setting Software](https://wiki.seeedstudio.com/Grove_Base_Hat_for_Raspberry_Pi/#installation) [https://wiki.seeedstudio.com/Grove\_Base\_Hat\_for\_Raspberry\_Pi/#installation] to configure the development environment.
- **Step 2.** Download the source file by cloning the grove.py library.



```
1 cd ~
2 git clone https://github.com/Seeed-Studio/grove.py
```

- **Step 3.** Excute below commands to run the code.

```
1 cd grove.py/grove
2 python3 grove_light_sensor_v1_2.py 0
```

Following is the grove\_light\_sensor\_v1\_2.py code.

```
1 import math
2 import sys
3 import time
4 from grove.adc import ADC
5
6
7 class GroveLightSensor:
8
9     def __init__(self, channel):
10         self.channel = channel
11         self.adc = ADC()
12
13     @property
14     def light(self):
15         value = self.adc.read(self.channel)
16         return value
17
18 Grove = GroveLightSensor
19
20
21 def main():
22     if len(sys.argv) < 2:
23         print('Usage: {} adc_channel'.format(sys.argv[0]))
24         sys.exit(1)
25
26     sensor = GroveLightSensor(int(sys.argv[1]))
27
```

```
28     print('Detecting light...')
29     while True:
30         print('Light value: {}'.format(sensor.light))
31         time.sleep(1)
32
33 if __name__ == '__main__':
34     main()
```



### Success

If everything goes well, you will be able to see the following result corresponding to the surrounding light

```
1 pi@raspberrypi:~/grove.py/grove $ python3 grove_light_sensor_v1_2.py
2 Detecting light...
3 Light value: 600
4 Light value: 448
5 Light value: 267
6 Light value: 311
7 Light value: 102
8 Light value: 82
9 Light value: 63
10 Light value: 54
11 Light value: 49
12 Light value: 45
13 Light value: 545
14 ^CTraceback (most recent call last):
15   File "grove_light_sensor_v1_2.py", line 67, in <module>
16     main()
17   File "grove_light_sensor_v1_2.py", line 64, in main
18     time.sleep(1)
19 KeyboardInterrupt
```

You can quit this program by simply press `Ctrl+C`.





### Notice

You may have noticed that for the analog port, the silkscreen pin number is something like **A1**, **A0**, however in the command we use parameter **0** and **1**, just the same as digital port. So please make sure you plug the module into the correct port, otherwise there may be pin conflicts.

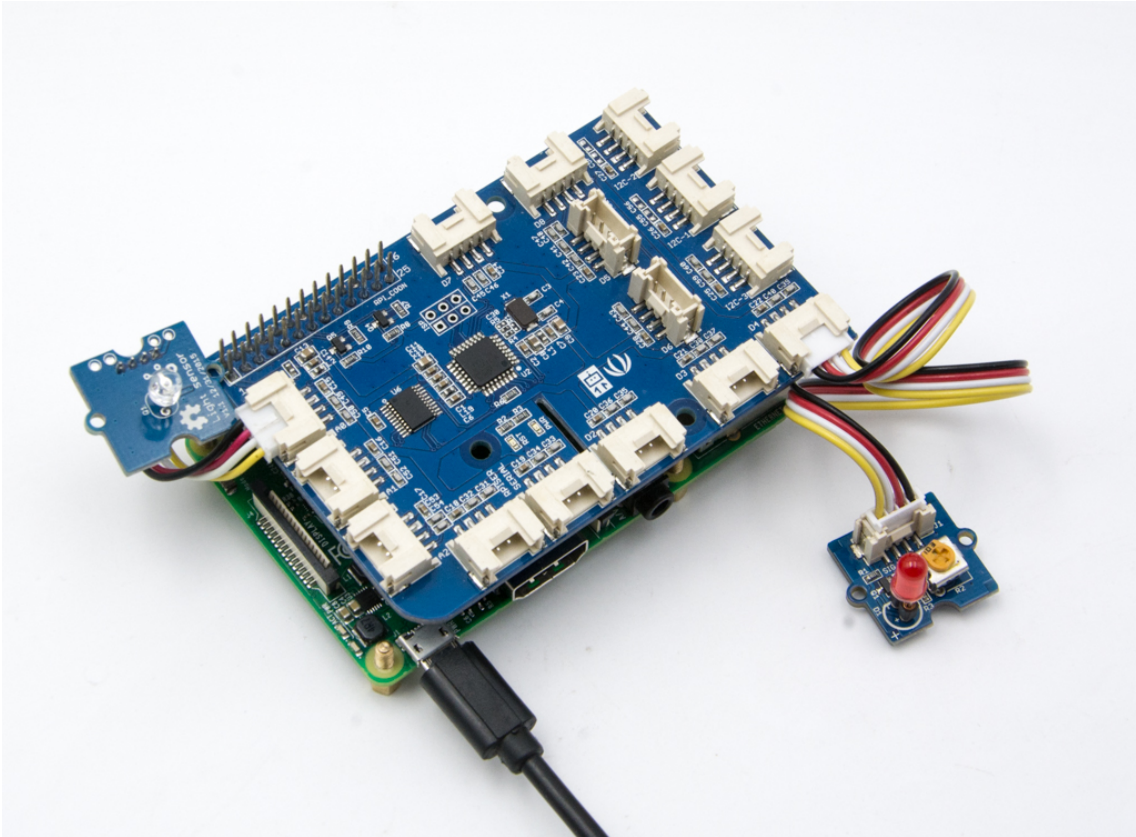
## Play With Raspberry Pi (with GrovePi\_Plus)

### Hardware

- Step 1. Prepare the below stuffs:

Raspberry pi	GrovePi_Plus
	
<p><a href="https://www.seeedstudio.com/Raspberry-Pi-3-Model-B-p-2625.html">Get One Now</a> [https://www.seeedstudio.com/Raspberry-Pi-3-Model-B-p-2625.html]</p>	<p><a href="https://www.seeedstudio.com/GrovePi-Plus-p-2241.html">Get One Now</a> [https://www.seeedstudio.com/GrovePi-Plus-p-2241.html]</p>

- Step 2. Plug the GrovePi\_Plus into Raspberry.
- Step 3. Connect Grove-light sensor to A0 port of GrovePi\_Plus.
- Step 4. Connect Grove-Red Led to D4 port of GrovePi\_Plus.
- Step 5. Connect the Raspberry to PC through USB cable.



## Software



### Attention

If you are using **Raspberry Pi with Raspberrypi OS >= Bullseye**, you have to use this command line **only with Python3**.

- Step 1. Follow [Setting Software](https://www.dexterindustries.com/GrovePi/get-started-with-the-grovepi/setting-software/) [https://www.dexterindustries.com/GrovePi/get-started-with-the-grovepi/setting-software/] to configure the development environment.
- Step 2. Git clone the Github repository.

```
1 cd ~  
2 git clone https://github.com/DexterInd/GrovePi.git
```

- Step 3. Excute below commands.

```
1 cd ~/GrovePi/Software/Python
2 python3 grove_light_sensor.py
```

Here is the grove\_light\_sensor.py code.

```
1 import time
2 import grovepi
3
4 # Connect the Grove Light Sensor to analog port A0
5 # SIG,NC,VCC,GND
6 light_sensor = 0
7
8 # Connect the LED to digital port D4
9 # SIG,NC,VCC,GND
10 led = 4
11
12 # Turn on LED once sensor exceeds threshold resistance
13 threshold = 10
14
15 grovepi.pinMode(light_sensor,"INPUT")
16 grovepi.pinMode(led,"OUTPUT")
17
18 while True:
19     try:
20         # Get sensor value
21         sensor_value = grovepi.analogRead(light_sensor)
22
23         # Calculate resistance of sensor in K
24         resistance = (float)(1023 - sensor_value) * 10 /
25
26         if resistance > threshold:
27             # Send HIGH to switch on LED
28             grovepi.digitalWrite(led,1)
29         else:
30             # Send LOW to switch off LED
31             grovepi.digitalWrite(led,0)
32
33         print("sensor_value = %d resistance = %.2f" %(sen
```



```
34     time.sleep(.5)
35
36     except IOError:
37         print ("Error")
```

- Step 4. The led will turn on when the light sensor gets covered.

```
1  pi@raspberrypi:~/GrovePi/Software/Python $ python3 grove_
2  sensor_value = 754 resistance = 3.57
3  sensor_value = 754 resistance = 3.57
4  sensor_value = 752 resistance = 3.60
5  sensor_value = 752 resistance = 3.60
6  sensor_value = 752 resistance = 3.60
7  sensor_value = 313 resistance = 22.68
8  sensor_value = 155 resistance = 56.00
9  sensor_value = 753 resistance = 3.59
```

## Eagle File for Grove - Light Sensor V1.0



Eagle File for Grove - Light  
Sensor(P) V1.0



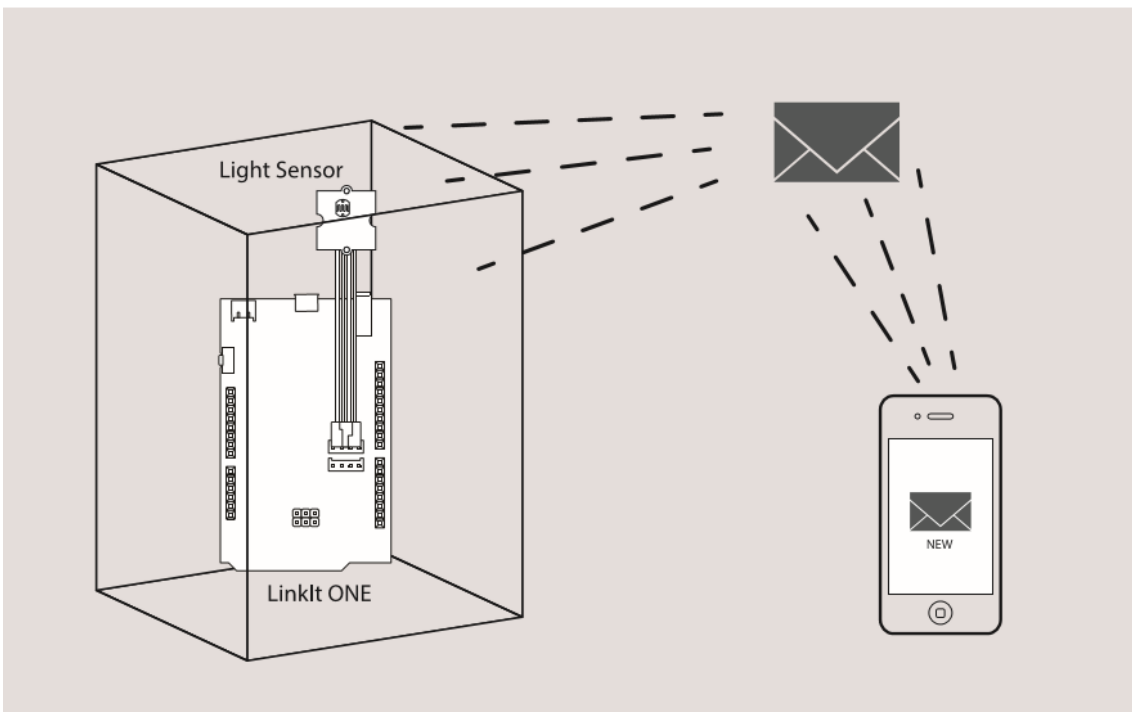
Eagle File for Grove - Light  
Sensor(P) V1.1



## Resources

- **[Codecraft]** [CDC File](https://files.seeedstudio.com/wiki/Grove_Light_Sensor/res/Grove_Light_Sensor_CDC_File.zip)  
[https://files.seeedstudio.com/wiki/Grove\_Light\_Sensor/res/Grove\_Light\_Sensor\_CDC\_File.zip]
- **[Eagle&PDF]** [Eagle File for Grove - Light Sensor V1.0](https://files.seeedstudio.com/wiki/Grove_Light_Sensor/resources/Grove%20-%20Light%20Sensor.zip)  
[https://files.seeedstudio.com/wiki/Grove\_Light\_Sensor/resources/Grove%20-%20Light%20Sensor.zip]

- **[Eagle&PDF]** [Eagle File for Grove - Light Sensor\(P\) V1.0](https://files.seeedstudio.com/wiki/Grove_Light_Sensor/resources/Grove%20-%20Light%20Sensor%28P%29.zip)  
[https://files.seeedstudio.com/wiki/Grove\_Light\_Sensor/resources/Grove%20-%20Light%20Sensor%28P%29.zip]
- **[Eagle&PDF]** [Eagle File for Grove - Light Sensor\(P\) V1.1](https://files.seeedstudio.com/wiki/Grove_Light_Sensor/resources/Grove%20-%20Light%20Sensor%28P%29%20v1.1.zip)  
[https://files.seeedstudio.com/wiki/Grove\_Light\_Sensor/resources/Grove%20-%20Light%20Sensor%28P%29%20v1.1.zip]
- **[Datasheet]** [LS06-MΦ5 Reference Information](https://files.seeedstudio.com/wiki/Grove_Light_Sensor/res/LS06-M%CE%A65_datasheet.pdf)  
[https://files.seeedstudio.com/wiki/Grove\_Light\_Sensor/res/LS06-M%CE%A65\_datasheet.pdf]
- **[Datasheet]** [LM358.PDF](https://files.seeedstudio.com/wiki/Grove_Light_Sensor/res/LM358.pdf)  
[https://files.seeedstudio.com/wiki/Grove\_Light\_Sensor/res/LM358.pdf]
- **[More Reading]** Secret Box



Here we will show you a project made with Grove - Light Sensor - Secret Box. First you need a box, a paper box, wooden box, any box is ok. Put something in the box, because we named it secret box,

that means we don't want anybody to open it, otherwise there will be an alarm to inform you.

Here we use LinkIt ONE as the controller, which is an Arduino compatible board and consist of rich function. And you need things below:

- [LinkIt ONE](https://www.seeedstudio.com/LinkIt-ONE-p-2017.html) [https://www.seeedstudio.com/LinkIt-ONE-p-2017.html]
- Grove - Light Sensor
- Grove - Base Shield
- A Sim Card

Let's connect Grove - Light Sensor to A0 or Base Shield, and open Arduino IDE, copy below code and upload the example to LinkIt ONE. Then someone open the box, the light will detect it, and send you a SMS.

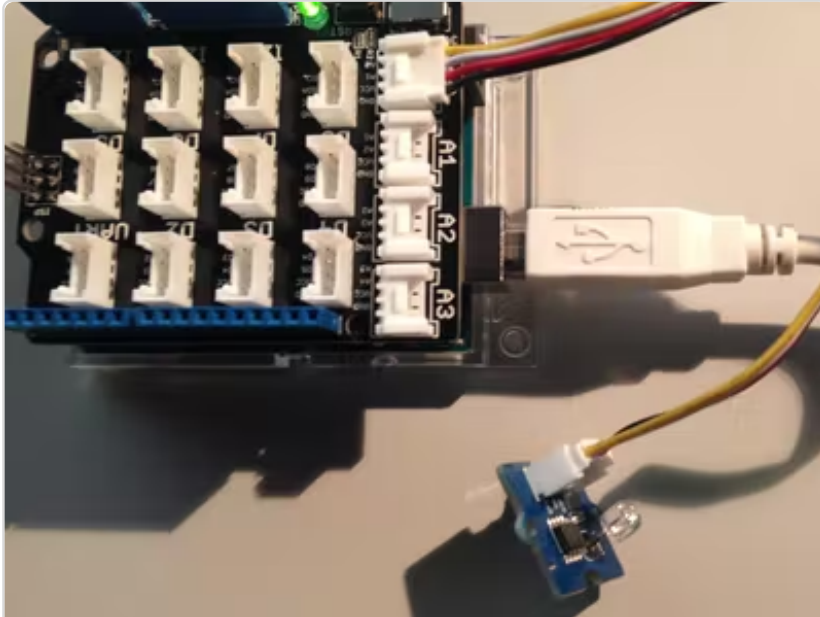
```
1 // demo of Grove Starter kit for LinkIt ONE
2 // Secret box
3
4 #include <LGSM.h>
5
6 char num[20] = "13425171053"; // your number w
7 char text[100] = "Warning: Your box had been opened!!";
8
9
10 const int pinLight = A0; // light sensor
11
12 bool isLightInBox()
13 {
14     return (analogRead(pinLight)<50); // when get data
15 }
16
17 void setup()
```

```
18 {
19     Serial.begin(115200);
20
21     while(!isLightInBox());           // until put in l
22     delay(2000);
23 }
24
25
26 void loop()
27 {
28     if(!isLightInBox())               // box is open
29     {
30         Serial.println("box had been opened");
31
32         while(!LSMS.ready())
33         {
34             delay(1000);
35         }
36
37         Serial.println("SIM ready for work!");
38         LSMS.beginSMS(num);
39         LSMS.print(text);
40
41         if(LSMS.endSMS())
42         {
43             Serial.println("SMS is sent");
44         }
45         else
46         {
47             Serial.println("SMS send fail");
48         }
49
50         while(!isLightInBox());       // until put
51         delay(2000);
52     }
53
54     delay(10);
55 }
```

Have fun.

# Projects

## Grove - Introduction in a Light Sensor:



(<https://www.hackster.io/ingo-lohs/grove-introduction-in-a-light-sensor-a55efd>)

**The Environment Cube! Know the Land Beneath You using Sigfox:**  
A cube with all the necessary sensors, suitable for a wide range of applications like agriculture, monitoring, ,etc.





(<https://www.hackster.io/dhairya-parikh/the-environment-cube-know-the-land-beneath-you-using-sigfox-952f29>)

**Light sensor Grove module:**

## Lesson 12 : Light sensor Grove module.



## Lección 12 : Módulo sensor...



## Tech Support

Please submit any technical issue into our [forum](https://forum.seeedstudio.com/)  
[<https://forum.seeedstudio.com/>].



[[https://www.seeedstudio.com/act-4.html?utm\\_source=wiki&utm\\_medium=wikibanner&utm\\_campaign=newproducts](https://www.seeedstudio.com/act-4.html?utm_source=wiki&utm_medium=wikibanner&utm_campaign=newproducts)]