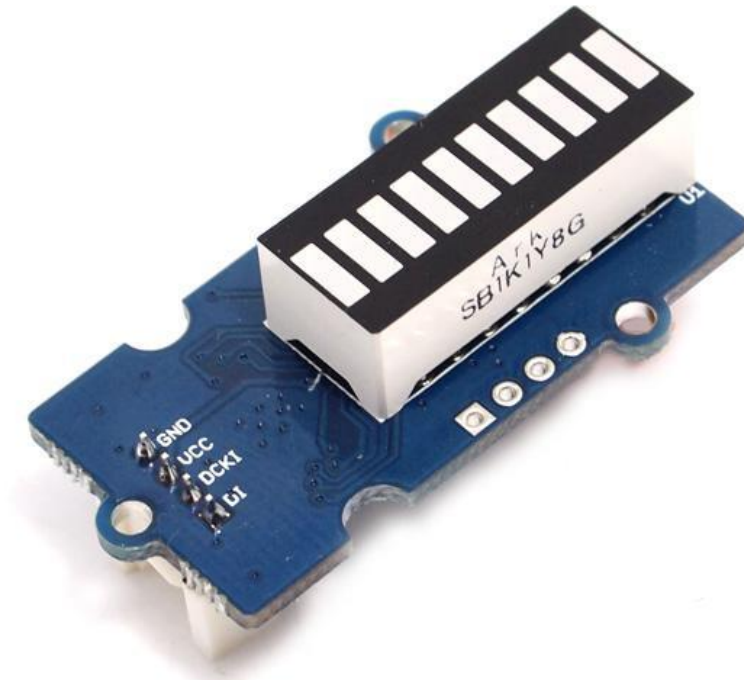


## Grove - LED Bar



Grove – LED Bar is comprised of a 10 segment LED gauge bar and an MY9221 LED controlling chip. It can be used as an indicator for remaining battery life, voltage, water level, music volume or other values that require a gradient display. There are 10 LED bars in the LED bar graph: one red, one yellow, one light green, and seven green bars. Demo code is available to get you up and running quickly. It lights up the LEDs sequentially from red to green, so the entire bar graph is lit up in the end. Want to go further? Go ahead and code your own effect.

### Version

Product Version	Changes	Released Date
Grove – LED Bar V1	Initial	June 2014
Grove – LED Bar V2	Improved the power supply	Oct 2015

## Features

- Input Voltage: 3.3V/5V
- Each LED segment can be controlled individually via code
- Intuitive display
- Flexible power option, supports 3-5.5DC
- Available demo code
- Suli-compatible Library


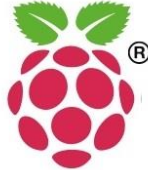



### Tip

More details about Grove modules please refer to [Grove System](#)

## Specification

Parameter	Value/Range
Operating voltage	3.3/5V
Operation Temperature	-20°C to +80°C
Peak Emission Wavelength-RED(Current 20mA)	630-637nm
Peak Emission Wavelength-Yellow Green(Current 20mA )	570-573nm
Peak Emission Wavelength-Yellow(Current 20mA )	585-592nm
Luminous Intensity Per Segment-RED(Current 20mA )	50-70mcd
Luminous Intensity Per Segment-Yellow Green(Current 20mA )	28-35mcd
Luminous Intensity Per Segment-Yellow(Current 20mA )	45-60mcd
LED segment	10
Size	40mm * 20mm

## Platforms Supported

Arduino	Raspberry Pi	BeagleBone	Wio	LinkIt ONE
				

### 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.

## Getting Started




### Note

If this is the first time you work with Arduino, we firmly recommend you to see [Getting Started with Arduino](#) before the start.

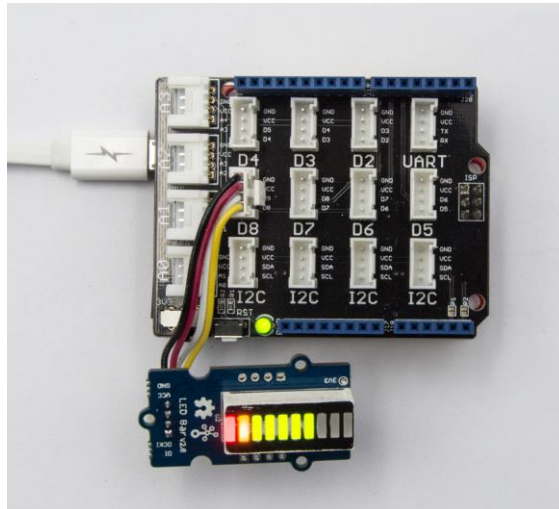
## Play With Arduino

### Hardware

- **Step 1.** Prepare the below stuffs:

Seeeduino V4.2	Base Shield	Grove-LED Bar
		

- **Step 2.** Connect Grove-LED Bar to port **D8** of Grove-Base Shield.
- **Step 3.** Plug Grove - Base Shield into Seeduino.
- **Step 4.** Connect Seeduino to PC via a USB cable.



### Note

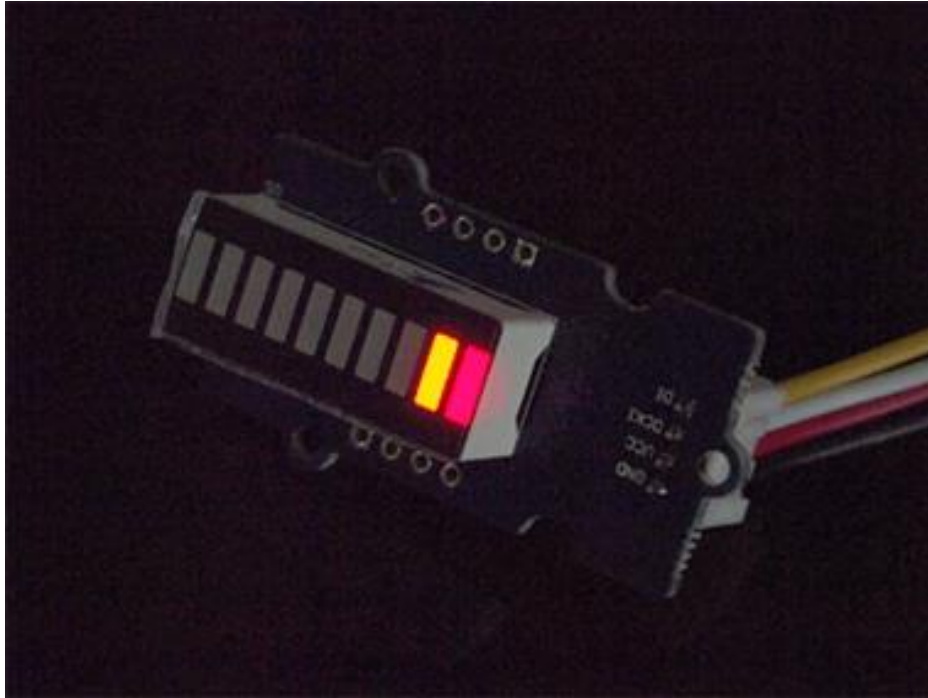
If we don't have Grove Base Shield, We also can directly connect Grove-LED Bar to Seeduino as below.

Seeduino	Grove-LED Bar
5V	Red
GND	Black
D9	White
D8	Yellow

### Software

- **Step 1.** Download the [Grove - LED Bar Library](#) from Github
- **Step 2.** Refer [How to install library](#) to install library for Arduino.
- **Step 3.** Restart the Arduino IDE. Open “Level” example via the path : **File** → **Examples** → **Grove LED Bar** → **Level**.
- **Step 4.** Upload the demo. If you do not know how to upload the code, please check [how to upload code](#).




The result should be like:



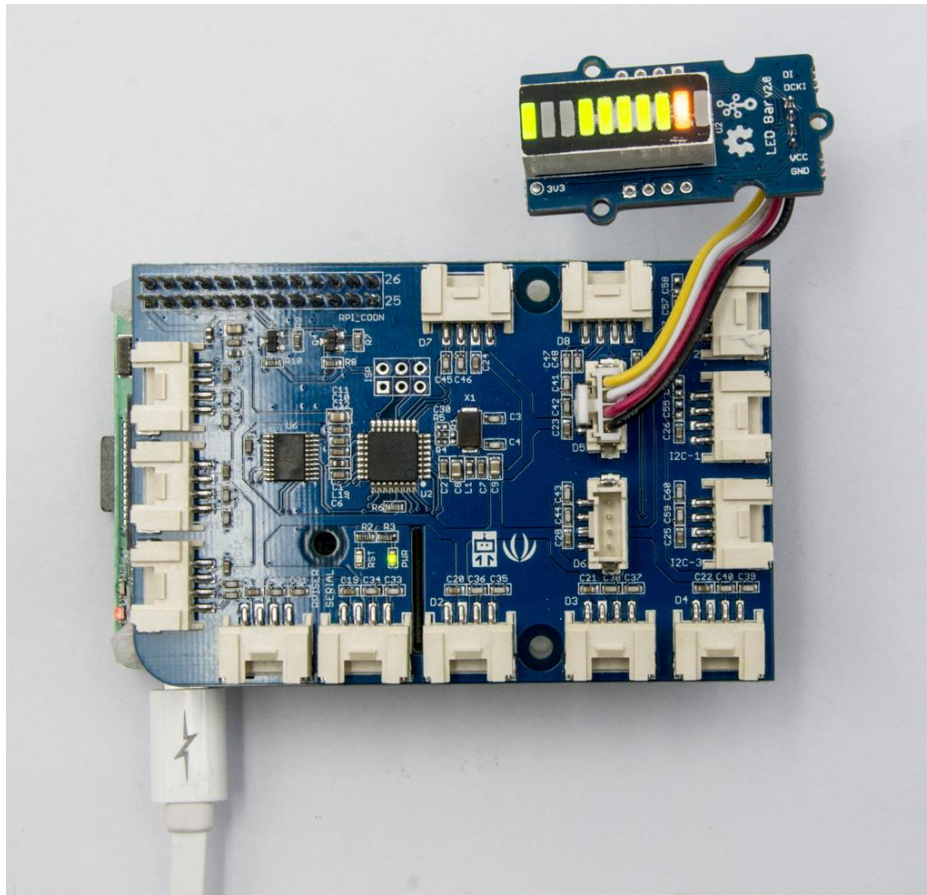
## Play With Raspberry Pi

### Hardware

- **Step 1.** Prepare the below stuffs:
- 

Raspberry pi	GrovePi_Plus	Grove-LED Bar
		

- **Step 2.** Plug the GrovePi\_Plus into Raspberry.
- **Step 3.** Connect Grove-LED Bar to **D5** port of GrovePi\_Plus.
- **Step 4.** Connect the Raspberry to PC through USB cable.



## Software

- **Step 1.** Follow [Setting Software](#) to configure the development environment.
- **Step 2.** Follow [Updating the Firmware](#) to update the newest firmware of GrovePi.

## Tip

In this wiki we use the path `~/GrovePi/` instead of `/home/pi/Desktop/GrovePi/`, you need to make sure Step 2 and Step 3 use the same path.

## Note

We firmly suggest you to update the firmware, or for some sensors you may get errors.

- **Step 3.** Git clone the Github repository.

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

- **Step 4.** Navigate to the demos' directory:

```
1cd yourpath/GrovePi/Software/Python/
```

Here is the grove\_ledbar.py code.

```
1 import time
2 import grovepi
3 import random
4
5 # Connect the Grove LED Bar to digital port D5
6 # DI,DCKI,VCC,GND
7 ledbar = 5
8
9 grovepi.pinMode(ledbar,"OUTPUT")
10 time.sleep(1)
11 i = 0
12
13 # LED Bar methods
14 # grovepi.ledBar_init(pin,orientation)
15 # grovepi.ledBar_orientation(pin,orientation)
16 # grovepi.ledBar_setLevel(pin,level)
17 # grovepi.ledBar_setLed(pin,led,state)
18 # grovepi.ledBar_toggleLed(pin,led)
19 # grovepi.ledBar_setBits(pin,state)
20 # grovepi.ledBar_getBits(pin)
21
22 while True:
23     try:
24         print "Test 1) Initialise - red to green"
25         # ledbar_init(pin,orientation)
26         # orientation: (0 = red to green, 1 = green to red)
27         grovepi.ledBar_init(ledbar, 0)
28         time.sleep(.5)
29
30
31         print "Test 2) Set level"
32         # ledbar_setLevel(pin,level)
33         # level: (0-10)
34         for i in range(0,11):
35             grovepi.ledBar_setLevel(ledbar, i)
36             time.sleep(.2)
37         time.sleep(.3)
38
39         grovepi.ledBar_setLevel(ledbar, 8)
40         time.sleep(.5)
41
42         grovepi.ledBar_setLevel(ledbar, 2)
43         time.sleep(.5)
44
45         grovepi.ledBar_setLevel(ledbar, 5)
46         time.sleep(.5)
47
48
49         print "Test 3) Switch on/off a single LED"
50         # ledbar_setLed(pin,led,state)
51         # led: which led (1-10)
52         # state: off or on (0,1)
53         grovepi.ledBar_setLed(ledbar, 10, 1)
54         time.sleep(.5)
```

```

55
56     grovepi.ledBar_setLed(ledbar, 9, 1)
57     time.sleep(.5)
58
59     grovepi.ledBar_setLed(ledbar, 8, 1)
60     time.sleep(.5)
61
62     grovepi.ledBar_setLed(ledbar, 1, 0)
63     time.sleep(.5)
64
65     grovepi.ledBar_setLed(ledbar, 2, 0)
66     time.sleep(.5)
67
68     grovepi.ledBar_setLed(ledbar, 3, 0)
69     time.sleep(.5)
70
71
72     print "Test 4) Toggle a single LED"
73     # flip a single led - if it is currently on, it will become off
74 and vice versa
75     # ledbar_toggleLed(ledbar, led)
76     grovepi.ledBar_toggleLed(ledbar, 1)
77     time.sleep(.5)
78
79     grovepi.ledBar_toggleLed(ledbar, 2)
80     time.sleep(.5)
81
82     grovepi.ledBar_toggleLed(ledbar, 9)
83     time.sleep(.5)
84
85     grovepi.ledBar_toggleLed(ledbar, 10)
86     time.sleep(.5)
87
88
89     print "Test 5) Set state - control all leds with 10 bits"
90     # ledbar_setBits(ledbar, state)
91     # state: (0-1023) or (0x00-0x3FF) or (0b0000000000-0b1111111111)
92 or (int('0000000000',2)-int('1111111111',2))
93     for i in range(0,32):
94         grovepi.ledBar_setBits(ledbar, i)
95         time.sleep(.2)
96     time.sleep(.3)
97
98
99     print "Test 6) Get current state"
100    # state = ledbar_getBits(ledbar)
101    # state: (0-1023) a bit for each of the 10 LEDs
102    state = grovepi.ledBar_getBits(ledbar)
103    print "with first 5 leds lit, the state should be 31 or 0x1F"
104    print state
105
106    # bitwise shift five bits to the left
107    state = state << 5
108    # the state should now be 992 or 0x3E0
109    # when saved the last 5 LEDs will be lit instead of the first 5
110 LEDs
111    time.sleep(.5)

```



```

112
113
114     print "Test 7) Set state - save the state we just modified"
115     # ledbar_setBits(ledbar, state)
116     # state: (0-1023) a bit for each of the 10 LEDs
117     grovepi.ledBar_setBits(ledbar, state)
118     time.sleep(.5)
119
120
121     print "Test 8) Swap orientation - green to red - current state is
122 preserved"
123     # ledbar_orientation(pin,orientation)
124     # orientation: (0 = red to green, 1 = green to red)
125     # when you reverse the led bar orientation, all methods know how
126 to handle the new LED index
127     # green to red
128     grovepi.ledBar_orientation(ledbar, 1)
129     time.sleep(.5)
130
131     # red to green
132     grovepi.ledBar_orientation(ledbar, 0)
133     time.sleep(.5)
134
135     # green to red
136     grovepi.ledBar_orientation(ledbar, 1)
137     time.sleep(.5)
138
139
140     print "Test 9) Set level, again"
141     # ledbar_setLevel(pin,level)
142     # level: (0-10)
143     # note the red LED is now at index 10 instead of 1
144     for i in range(0,11):
145         grovepi.ledBar_setLevel(ledbar, i)
146         time.sleep(.2)
147     time.sleep(.3)
148
149
150     print "Test 10) Set a single LED, again"
151     # ledbar_setLed(pin,led,state)
152     # led: which led (1-10)
153     # state: off or on (0,1)
154     grovepi.ledBar_setLed(ledbar, 1, 0)
155     time.sleep(.5)
156
157     grovepi.ledBar_setLed(ledbar, 3, 0)
158     time.sleep(.5)
159
160     grovepi.ledBar_setLed(ledbar, 5, 0)
161     time.sleep(.5)
162
163
164     print "Test 11) Toggle a single LED, again"
165     # ledbar_toggleLed(ledbar, led)
166     grovepi.ledBar_toggleLed(ledbar, 2)
167     time.sleep(.5)
168

```

```

169 grovepi.ledBar_toggleLed(ledbar, 4)
170 time.sleep(.5)
171
172
173 print "Test 12) Get state"
174 # state = ledbar_getBits(ledbar)
175 # state: (0-1023) a bit for each of the 10 LEDs
176 state = grovepi.ledBar_getBits(ledbar)
177
178 # the last 5 LEDs are lit, so the state should be 992 or 0x3E0
179
180 # bitwise shift five bits to the right
181 state = state >> 5
182 # the state should now be 31 or 0x1F
183
184
185 print "Test 13) Set state, again"
186 # ledbar_setBits(ledbar, state)
187 # state: (0-1023) a bit for each of the 10 LEDs
188 grovepi.ledBar_setBits(ledbar, state)
189 time.sleep(.5)
190
191
192 print "Test 14) Step"
193 # step through all 10 LEDs
194 for i in range(0,11):
195     grovepi.ledBar_setLevel(ledbar, i)
196     time.sleep(.2)
197 time.sleep(.3)
198
199
200 print "Test 15) Bounce"
201 # switch on the first two LEDs
202 grovepi.ledBar_setLevel(ledbar, 2)
203
204 # get the current state (which is 0x3)
205 state = grovepi.ledBar_getBits(ledbar)
206
207 # bounce to the right
208 for i in range(0,9):
209     # bit shift left and update
210     state <<= 1;
211     grovepi.ledBar_setBits(ledbar, state)
212     time.sleep(.2)
213
214 # bounce to the left
215 for i in range(0,9):
216     # bit shift right and update
217     state >>= 1;
218     grovepi.ledBar_setBits(ledbar, state)
219     time.sleep(.2)
220 time.sleep(.3)
221
222
223 print "Test 16) Random"
224 for i in range(0,21):
225     state = random.randint(0,1023)

```

```

226         grovepi.ledBar_setBits(ledbar, state)
227         time.sleep(.2)
228     time.sleep(.3)
229
230
231     print "Test 17) Invert"
232     # set every 2nd LED on - 341 or 0x155
233     state = 341
234     for i in range(0,5):
235         grovepi.ledBar_setBits(ledbar, state)
236         time.sleep(.2)
237
238         # bitwise XOR all 10 LEDs on with the current state
239         state = 0x3FF ^ state
240
241         grovepi.ledBar_setBits(ledbar, state)
242         time.sleep(.2)
243     time.sleep(.3)
244
245
246     print "Test 18) Walk through all possible combinations"
247     for i in range(0,1024):
248         grovepi.ledBar_setBits(ledbar, i)
249         time.sleep(.1)
250     time.sleep(.4)
251
252     except KeyboardInterrupt:
253         grovepi.ledBar_setBits(ledbar, 0)
254         break
255     except IOError:
256         print "Error"

```

- **Step 5.** Run the demo.

```
1sudo python grove_ledbar.py
```

## Resources

- [\[Eagle&PDF\]Grove - LED Bar Eagle File](#)
- [\[Library\]Grove - LED Bar Library](#)
- [\[Library\]Suli-compatible Library](#)
- [\[Datasheet\]MY9221 Datasheet](#)
- [\[More Reading\]Wooden Laser Gun](#)

## Projects

**Grove LED Bar v2.0:** Calliope Mini is equipped with two Grove connectors. In this project, I want to explore, how to talk to these Seeed Grove parts.

**Grove LED Bar Controller with the Bean+:** Learn the basics of using popular Grove components with the new LightBlue Bean+ to get started with building your own projects!

## Tech Support

Please submit any technical issue into our [forum](#).

## **X-ON Electronics**

Largest Supplier of Electrical and Electronic Components

*Click to view similar products for [Seeed Studio Accessories](#) category:*

*Click to view products by [Seeed Studio](#) manufacturer:*

Other Similar products are found below :

[101990565](#) [114992282](#) [114992268](#) [101990647](#) [103060000](#) [101020004](#) [101020006](#)

[101020012](#) [101020018](#) [101020028](#) [101020038](#) [101020045](#) [101020049](#) [101020052](#)

[101020055](#) [101020056](#) [101020058](#) [101020472](#) [101020580](#) [101020603](#) [101990007](#)

[101990028](#) [101990053](#) [101990058](#) [101990059](#)