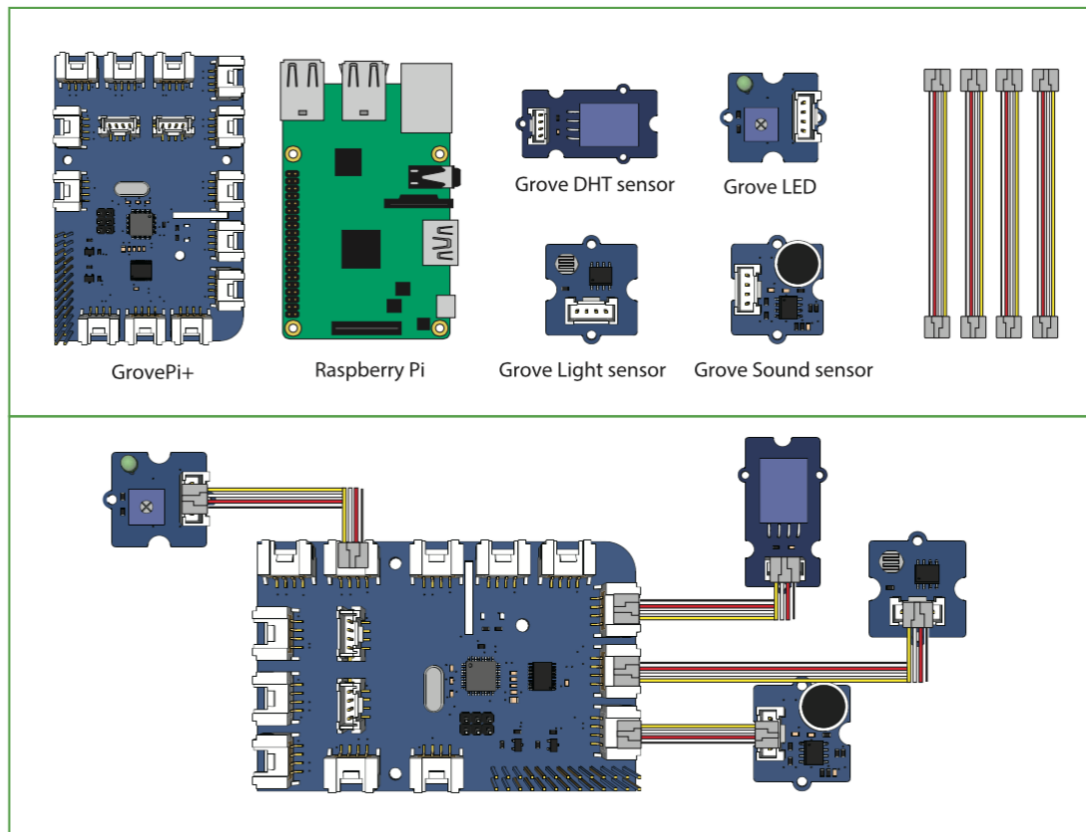


## Practical 6

### PART 1: Connecting to the cloud (internet) - thinkspeaks.com

#### Step 1: Test Light and Sound sensors

*\* Picture is for illustration ONLY, please follow the steps correctly.*



1. Connect a sound sensor to port A0, a Grove Light sensor to port A1 and a Grove DHT sensor at port A2. Connect a green LED to port D3.

\* Port A0, A1, A2 are analog ports

\* Port D3 is the Pulse Wave Modulation (PWM) port, responding to the light conditions.

**Step 2:** In **Thonny Python (ID)**, click “New” to create a new python file and Save As “test07.py”. Type the following codes:

```

test07.py
1 from time import *
2 from grovepi import *
3 from grove_rgb_lcd import *
4 from random import *
5 from urllib.request import *
6
7 led = 3
8 soundsensor = 14
9 lightsensor = 15
10 dhtsensor = 16
11 pinMode(led, "OUTPUT")
12 pinMode(soundsensor, "INPUT")
13 pinMode(lightsensor, "INPUT")
14 pinMode(dhtsensor, "INPUT")
15 apikey = [YOUR_API_KEY]
16
17 while True:
18     try:
19         # adjust the sleep time if you have succesfully push data to your thingspeak
20         sleep(5) // change to 15 seconds
21         [temp, hum] = dht(dhtsensor, 0)
22         light = analogRead(lightsensor)
23         sound = analogRead(soundsensor)
24         print("Temp = %.2f, hum = %d, light = %d, sound = %d" %(temp, hum, light, sound))
25         t = str(temp)
26         h = str(hum)
27         setText("Temp = " + t + '\337' + "C   Hum = " + h + " %")
28         r = randint(0, 255)
29         g = randint(0, 255)
30         b = randint(0, 255)
31         setRGB(r, g, b)
32         content = urlopen("https://api.thingspeak.com/update?api_key=" + apikey + "&field1=" + t).read()
33         if (content):
34             print("Updated Thingspeak")
35         # remove the break if you have succesfully push data to your thingspeak
36         break
37     except KeyboardInterrupt:
38         setText("Program Exited")
39         break
40     except TypeError:
41         print("Type Error occurs")
42     except IOError:
43         print("IO Error occurs")
44
45

```

**Step 3:** Register an account at Thingspeak.com, refer to *Setup - Thingspeak.pdf*

**Step 4:** Replace the api\_key with your own one. Run the code.

**\* Explore the other way to display data (line graph, gauge chart, etc) by selecting different visualization features in Thingspeak.com**

**\* Attach with another display device (\*Tips: setRGB is for \_\_\_\_\_, refer to Practical 4)**

**Task 1:** Update more sensors value to the Thingspeak.com

**Task 2:** Modify the code to make LED brightness responses to the changes of input sensors (e.g., sound sensor / light sensor)

## PART 2: Connecting to the cloud (internet) - Beebotte.com

**Step 1:** We use Beebotte.com, that is a simple IOT dashboard display, pre-installed in our Raspberry Pi module

\* If you are using your own Raspberry Pi, do install Beebotte by typing:

*sudo pip3 install beebotte*

\* Reference: [https://beebotte.com/tutorials/quick\\_start](https://beebotte.com/tutorials/quick_start)

**Step 2:** In **Thonny Python (ID)**, click “New” to create a new python file and Save As “test08.py”. Type the following codes:

```
test08.py *  
1  from time import *  
2  from grovepi import *  
3  from grove_rgb_lcd import *  
4  from random import *  
5  from urllib.request import *  
6  from beebotte import *  
7  
8  led = 3  
9  soundsensor = 14  
10 lightsensor = 15  
11 dhtsensor = 16  
12 pinMode(led, "OUTPUT")  
13 pinMode(soundsensor, "INPUT")  
14 pinMode(lightsensor, "INPUT")  
15 pinMode(dhtsensor, "INPUT")  
16 apikey = [YOUR_API_KEY]  
17 secretkey = [YOUR_SECRET_KEY]  
18 bclient = BBT(apikey, secretkey)  
19  
20 while True:  
21     try:  
22         # adjust the sleep time if you have succesfully push data to your beebotte  
23         sleep(5)  
24         [temp, hum] = dht(dhtsensor, 0)  
25         light = analogRead(lightsensor)  
26         sound = analogRead(soundsensor)  
27         print("Temp = %.2f, hum = %d, light = %d, sound = %d" %(temp, hum, light, sound))  
28         setText("Temp = " + t + '\337' + "C  Hum = " + h + " %")  
29         bclient.write('test', 'temperature', temp)  
30         bclient.write('test', 'humidity', hum)  
31         # remove the break if you have succesfully push data to your beebotte  
32         break  
33     except KeyboardInterrupt:  
34         setText("Program Exited")  
35         break  
36     except TypeError:  
37         print("Type Error occurs")  
38     except IOError:  
39         print("IO Error occurs")  
40  
41
```

2. Get the Beebotte credential from the Beebotte site > Account Setting > Access Management (refer to Setup - Beebotte.pdf file), replace the content with your own API Key and Secret Key.

3. Run the code.

**Task 1: Choose various graphical presentations to show the updated values.**

**Task 2: Extend the module to online reporting using one of this visualization websites.**

**<https://www.thethingsnetwork.org/forum/t/visualize-and-push-your-iot-data/1788>**

**Task 3: Compare the advantages and disadvantages of each visualization website.**