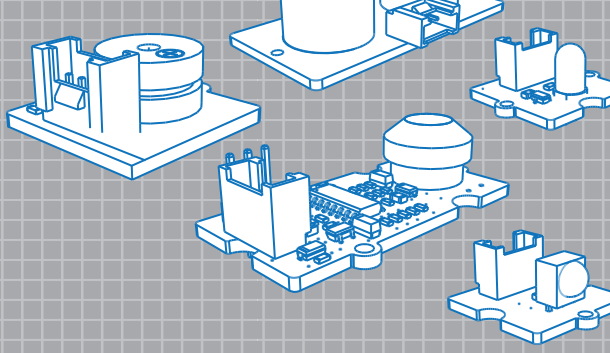
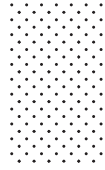


GROVE STARTER KIT for Photon

seeed studio

Grove Starter Kit for Photon



About Seeed Studio

Seeed is a hardware innovation platform for makers to grow inspirations into differentiating products. Seeed provides accessible technologies with quality, speed and supply chain knowledge. Seeed helps productize 1 to 1,000 pcs using in-house engineering, supply chain management and agile manufacture forces. Seeed also teams up with incubators, China tech ecosystem, investors and distribution channels to portal Maker startups beyond.
http://www.seeedstudio.com/depot/

About Grove

Grove is a modular, ready-to-use tool set that takes a building block approach to assembling electronics. The Grove system consists of a base shield and a large collection of modules that feature standardized connectors. The base shield allows for easy connection of any microcontroller to interface with the various Grove modules. Each Grove module addresses a unique function & the overall collection of modules covers a wide range of functionality from a simple push-button to a complex heart rate sensor. Each one comes with clear documentation and demo code to help you get started quickly.

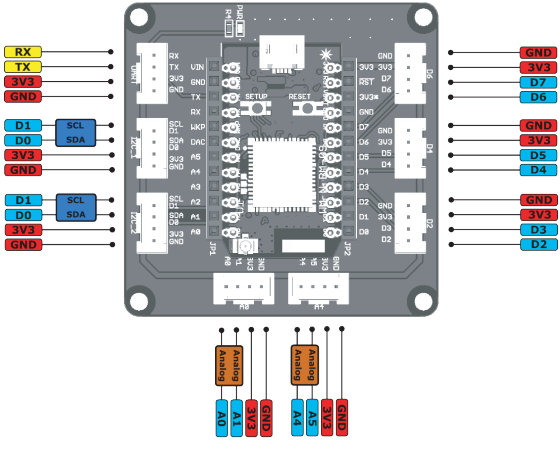
TABLE OF CONTENTS

LEARN MORE ABOUT THE PRODUCT

(1) Get to know the Photon

The Photon is a tiny Wi-Fi development board for creating connected projects and products. It's easy to use, it's powerful, and it's connected to the cloud.

Pin Map of Photon



Learn the RGB LED instructions:

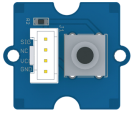
LED Instruction	Description
Blinking blue	Listening for Wi-Fi credentials
Solid blue	Getting Wi-Fi info from app
Blinking green	Connecting to the Wi-Fi network
Blinking cyan	Connecting to the Particle Cloud
Blinking magenta	Updating to the newest firmware
Breathing cyan	Connected!

(2) How to use this kit

This kit is a simple and easy tool to study how to use Groves on Photon and achieving rich functions with the sensors and actuators. There are 9 Groves, a expansion board and several cables in the kit, the silkscreen on expansion board clearly marked the pin map from Grove to Photon. This manual offers 6 basic demos and two enhanced projects to help you get started with Photon.

(3) Brief Information about the Groves

Grove - Button



What is it?

This new version of button twig contains one independent button, which is configured along with a pull-down resistor and ready for use with our microcontrollers as digital input. The pin marked SIG is for digital communication while the NC indicates 'not connected' in this case.

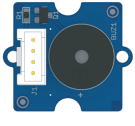
How to connect it?

Connect it to the Digital Input pin via Grove - Universal 4 Pin Buckled Wire.

More details:

http://www.seeedstudio.com/wiki/Grove_-_Button

Grove - Buzzer



What is it?

This is a simple yet enjoyable twig to use. The piezo can be connected to digital outputs, and will emit a tone when the output is high. Alternatively it can be connected to an analog pulse-width modulation(PWM) output to generate various tones and sound effects.

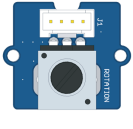
How to connect it?

Connect it to the Analog Output pin via Grove - Universal 4 Pin Buckled Wire.

More details:

http://www.seeedstudio.com/wiki/Grove_-_Buzzer

Grove - Rotary Angle Sensor



What is it?

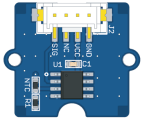
The potentiometer, can produce analog output between 0 and Vcc on its D1 connector. The D2 connector is not used. The angular range is 300 degrees with a linear change in value. The resistance value is 10k ohms which is perfect for Arduino applications. This may also be known as a rotary angle sensor.

How to connect it?

Connect it to the Analog Output pin via Grove - Universal 4 Pin Buckled Wire.

More details:
http://www.seeedstudio.com/wiki/Grove_-_Rotary_Angle_Sensor

Grove - Temperature Sensor

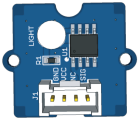


What is it?
The temperature sensor uses a thermistor which returns the ambient temperature in the form of resistance value of it . Then the voltage over the resistance is measured and converted to the corresponding temperature value. The operating range is -40 to 125°C with an accuracy of 1.5°C .

How to connect it?
Connect it to the Analog Output pin via Grove - Universal 4 Pin Buckled Wire.

More details:
http://www.seeedstudio.com/wiki/Grove_-_Temperature_Sensor

Grove - Light Sensor



What is it?
The Grove - Light Sensor v1.2 is updated version of Grove - Light Sensor 1.0 which is aimed at measuring light levels. It is an analog module and output various electrical signals which can be converted to different ranges (that depends on the Analog-to-Digital-Converter on your controller board. For example, it will output 0-255 for an 8-bit ADC). It integrates a high-sensitive and reliable photoresistor, and is interfaced with Grove port which will save you a lot of work in the wiring. This module can be used in various smart-lighting device or facilities.

How to connect it?
Connect it to the Analog Output pin via Grove - Universal 4 Pin Buckled Wire.

More details:
http://www.seeedstudio.com/wiki/Grove_-_Light_Sensor_v1.2

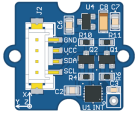
Grove - Chainable RGB LED

What is it?
Chainable RGB LED Twig is based on P9813 chip which is a full-color light source LED driver chip that can provide constant current drive and modulated output of 256 gray scale. Transmission by wire (DATA and CLK) and built-in recycling can effectively enhance the transmission distance of the module.

How to connect it?
Connect it to the CLC pin & DATA pin via Grove - Universal 4 Pin Buckled Wire.

More details:
http://www.seeedstudio.com/wiki/Grove_-_Chainable_RGB_LED

Grove - 3-Axis Digital Accelerometer(±1.5g)



What is it?

This 3-axis Accelerometer module is based on MMA7660FC with Digital Output I2C. This Module can be used for sensing data changes, product orientation, and gesture detection through an interrupt pin (INT).It contains a low power, low profile capacitive MEMS sensor.

How to connect it?

Connect it to the CLC pin & DATA pin via Grove - Universal 4 Pin Buckled Wire.

More details:

http://www.seeedstudio.com/wiki/Grove_-_3-Axis_Digital_Accelerometer(%C2%B11.5g)

Grove – 4-Digit Display

What is it?

The 4 digit display module is usually a 12 pin module. In this Grove gadget, we utilize a TM1637 to scale down the controlling pins into 2 Grove pins. It only takes 2 digital pins of Arduino or Seeeduino to control the content, even the luminance of this display. For projects that require of alpha-numeric display, this can be a nice choice.

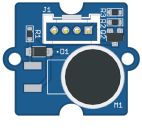
How to connect it?

Connect it to the CLC pin & DATA pin via Grove - Universal 4 Pin Buckled Wire.

More details:

http://www.seeedstudio.com/wiki/Grove_-_4-Digit_Display

Grove - Vibration Motor



What is it?

This module consists of a coin-sized Permanent Magnet DC motor. It vibrates whenever the input is logic HIGH. It is very easy to drive and Grove compatible. It can be used in DIY toys or cell phone modules.

How to connect it?

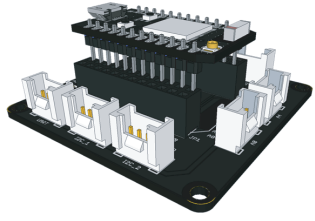
Connect it to the Digital Output pin via Grove - Universal 4 Pin Buckled Wire.

More details:

http://www.seeedstudio.com/wiki/Grove_-_Vibration_Motor

SETTLE THE ENVIRONMENT AND GET READY

(1) Put them together



(2) Connect your Photon to the Internet

Power the Photon

Connect the Photon with your computer via the USB cable. Now you can see the blinking blue LED indicator on the Photon.

Enter Listening Mode

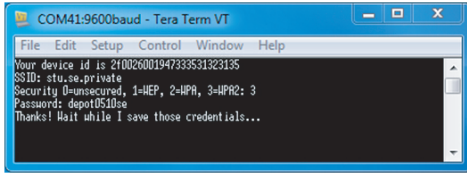
Unused Photon would boot up into the Listening Mode by default. or you can hold the SETUP button for more than 10 seconds to enter the Listening Mode. The RGB LED will be flashing blue in this mode.

Configure the Photon using terminal tool

Take PC as an example, open Tera Team and open the correct serial port of Photon, set the UART configuration as following, Baudrate: 9600, Data Bits: 8, Parity: none, Stop Bits: 1. You have two commands at your disposal by hitting either 'w' or 'i' on the keyboard. Here's what they do:

w: Set up your Wi-Fi SSID and password

i: ("i" as in identify) Read out the Photon ID

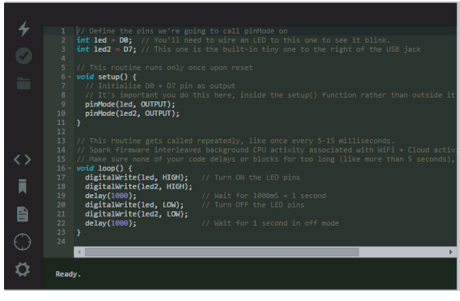


The main LED will flash Breathing cyan once the Photon has successfully connected to the Internet.

(3) IDE preparing

The Web IDE

Particle Build is a Web IDE, a complete development environment available through any modern web browser. Write code in your browser, save it online, and flash code over-the-air to Particle boards anywhere in the world. https://build.particle.io/.



Signing up and logging in

You should have an account before starting any coding in Particle's web IDE. And you can do this on https://build.particle.io/signup.

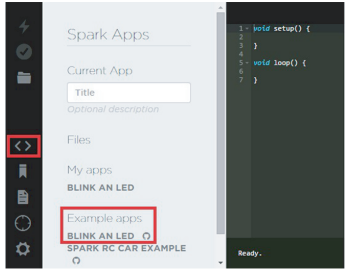
Claim your Photon

Enter the Web IDE and login in. Set your Photon into listening mode, open a serial connection and then type 'i', it should show you a series number, that is your Photon ID, copy it. Then click the 'DEVICE' icon. Click the button that says 'ADD NEW DEVICE', and enter your ID in the text box(before this please make sure your Photon has connected the Internet already). Then you could name the Photon. And then the Photon will appears on the list.

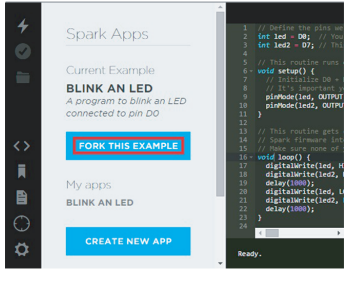
(4) Hello world

Open the example

Click "BLINK AN LED" under the Example apps title. This code turns D7 on and off once per second, a LED on board will flash accordingly.



You can fork it by click 'FORK THIS EXAMPLE'. Then it could be modified.

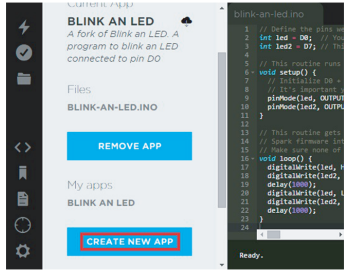


Compile and flash

Click the 'Verify' button to compile the project. Any warnings or errors would be reported on the bottom of the Spark IDE. Click the "Flash" button appears as a lightning icon and it will upload or "flash" this code onto your Photon. You'll see a series of status colors on the main LED, and then the little blue LED blinking.

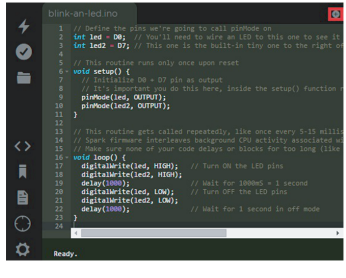
Create new Apps

If you are ready to write new programs, click 'CREATE NEW APP' in the Code Segment to simply add a new App. You may be required to give a name for your App.



Add libraries in the project

A complex project may include library files. You can add libraries with example codes by clicking 'Library' button and selecting the one you want. you can create a new one by clicking the 'Add' button on the top right. After type a name, a pair of new files will appear in the source code list.



(5) How to debug

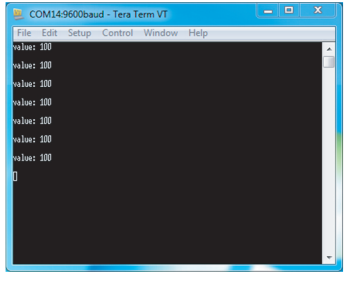
Printing functions

Photon provides compatible functions with Arduino to print debug info through virtual serial port, here're parts of useful API:

```
Serial.begin(9600);             // init serial port on USB interface
int value = 100;
Serial.print("value: ");        // print string
Serial.println(value);          // print variable with carriage return
Serial.println("");             // print a blank line
```

Using serial port

You could open applications such as Tera Team, set up a COM connection with the Photon, and read the debug information from your application prints when it is running.

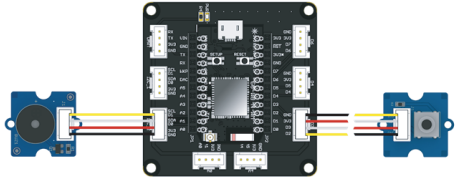


Example - 01: Singing A Song

WHAT ARE WE DOING HERE?

The example uses a buzzer to play melodies. A square wave at a appropriate frequency is applied to the buzzer to generate the 'song of square wave'.

THE CIRCUIT



CODE

```
// ------------------------------------
// Example - 01: Singing A Song
// ------------------------------------
#include "application.h"

// name the pins
#define BUTTONPIN D2
#define BUZZERPIN D1

int melody[] = {1908,2551,2551,2273,2551,0,2024,1908};  // notes in the melody
int noteDurations[] = {4,8,8,4,4,4,4,4 };               // note durations

// This routine runs only once upon reset
void setup()
{
  pinMode(BUTTONPIN, INPUT);                            // set user key pin as
input
}

// This routine loops forever
void loop()
```

```
{
  if(digitalRead(BUTTONPIN) == 1) {                    // if the button was
pressed
    for (int thisNote = 0; thisNote < 8; thisNote++) {  // ergodic all notes
      int noteDuration = 1000/noteDurations[thisNote];  // calculate the note
duration
      tone(BUZZERPIN, melody[thisNote], noteDuration);  // let speaker sonds
      int pauseBetweenNotes = noteDuration * 1.30;      // set a minimum time
between notes
      delay(pauseBetweenNotes);                         // delay for the while
      noTone(BUZZERPIN);                                // stop the tone
playing:
    }
  }
}
```

RESULT

When press the key, the speaker will sing a short song.

TIPS

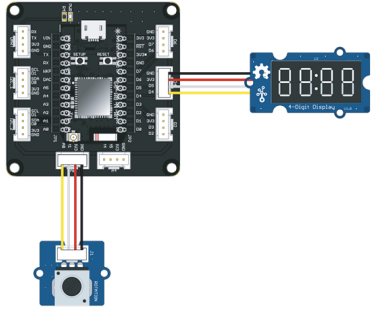
The tone takes three arguments, pin: the pin on which to generate the tone, frequency: the frequency of the tone in hertz and duration: the duration of the tone in milliseconds (a zero value = continuous tone).

Example - 02: Display the Analog Value

WHAT ARE WE DOING HERE?

The example demonstrates analog input by reading an analog pin. And display the voltage calculated on Grove 4-Digit Display.

THE CIRCUIT



```
// ------------------------------------
// Example - 02: Display the Analog Value
// ------------------------------------
#include "application.h"
#include "TM1637.h"

// name the pins
#define ROTARYPIN A0
#define CLK D4
#define DIO D5

void dispNum(unsigned int num);

TM1637 tm1637(CLK,DIO);

// This routine runs only once upon reset
void setup()
{
  tm1637.set();                                   // cofig TM1637
  tm1637.init();                                  // clear the display
}
```

```
// This routine loops forevera
void loop()
{
  int analogValue = analogRead(ROTARYPIN);      // read rotary pin
  int voltage = 3300 *  analogValue / 4096;     // calculate the voltage
  dispNum(voltage);                             // display the voltage
  delay(50);
}

//display a integer value less then 10000
void dispNum(unsigned int num)
{
  int8_t TimeDisp[] = {0x01,0x02,0x03,0x04};    // limit the maximum number

  if(num > 9999) num = 9999;
  TimeDisp[0] = num / 1000;
  TimeDisp[1] = num % 1000 / 100;
  TimeDisp[2] = num % 100 / 10;
  TimeDisp[3] = num % 10;
  tm1637.display(TimeDisp);
}
```

RESULT

Rotate the knob you will get the voltage changes.

TIPS

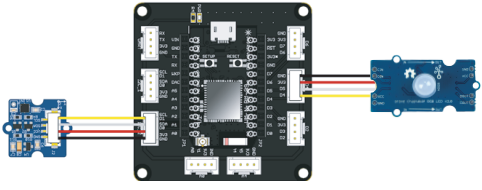
We predicate the reference voltage of the ADC as 3300mV, but the actual value we obtained might be a bit different from this due to the reference voltage error.

Example - 03: Controlling the RGB LED with Acceleration Sensor

WHAT ARE WE DOING HERE?

This example will demonstrate how to visualize the acceleration information with a RGB LED

THE CIRCUIT



CODE

```
// ------------------------------------
// Example - 03: Controlling the RGB LED with Acceleration Sensor
// ------------------------------------
#include "application.h"
#include "MMA7660.h"
#include "ChainableLED.h"

MMA7660 accelemeter;
#define NUM_LEDS  1
ChainableLED leds(D4, D5, NUM_LEDS);

// This routine runs only once upon reset
void setup()
{
  accelemeter.init();                          // initialize the g-sensor
  leds.setColorRGB(0, 0, 0, 0);                // turn down the LED
}

// This routine loops forever
void loop()
{
  int8_t x, y, z;

  accelemeter.getXYZ(&x,&y,&z);                // read the g-sensor
  // convert acc data for LED display
  x = x - 32;
  x = abs(x) * 8;
```

```
    y = y - 32;
    y = abs(y) * 8;
    z = z - 32;
    z = abs(z) * 8;
    leds.setColorRGB(0, x, y, z);                    // write LED data
    delay(50);
}
```

RESULT

We can see the LED changes color while rotating the acceleration sensor.

TIPS

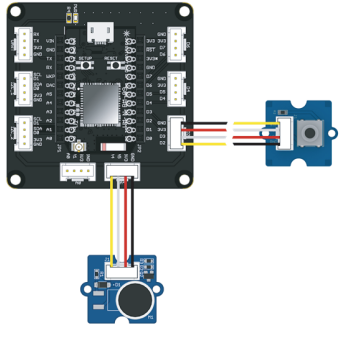
In this case, we may find that the LED light flickers, we can solve this by repeatedly reading the g-sensor value in the code (eg. 5 times), and then average the results. In this way the LED will be stable.

Example - 04: Start a Vibrator

WHAT ARE WE DOING HERE?

Vibration motor is often used for human-computer interactions. It's very easy to drive. We use a button to control it. it will vibrates whenever the key was pressed.

THE CIRCUIT



CODE

```
// ----------------------------------
// Example - 04: Start a Vibrator
// ----------------------------------
#include "application.h"
// name the pins
#define BUTTONPIN D2
#define MOTORPIN A4

// This routine runs only once upon reset
void setup()
{
  pinMode(BUTTONPIN, INPUT);
  pinMode(MOTORPIN, OUTPUT);
}

// This routine loops forever
void loop()
{
  int val = digitalRead(BUTTONPIN);               // read the hall sensor pin

  if(val == 0)                                   // if magnet detected
    digitalWrite(MOTORPIN, LOW);                // let the motor vibrate
  else
    digitalWrite(MOTORPIN, HIGH);                // stop it
  delay(50);
}
```

RESULT

As we expected, the motor does vibrate when we push the button.

TIPS

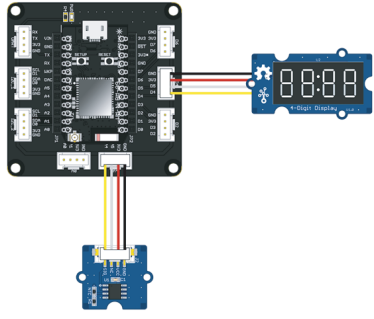
Vibration motor is often used in mobile phones to provide a third way to interact with users rather than visual and auditory interactions.

Example - 05: Measuring Temperature

WHAT ARE WE DOING HERE?

This example allows you to read the environmental temperature and show it on the 4-Digit Display.

THE CIRCUIT



CODE

```
// ----------------------------------
// Example - 05: Measuring Temperature
// ----------------------------------
#include "application.h"
#include <math.h>
#include "TM1637.h"

void dispNum(unsigned int num);

// name the pins
#define CLK D4
#define DIO D5
TM1637 tm1637(CLK,DIO);
#define TEMPPIN A4

// This routine runs only once upon reset
void setup()
{
  tm1637.set();                                   // cofig TM1637
  tm1637.init();                                  // clear the display
```

```
  Serial.begin(9600);                              // init serial port on USB
interface
}

// This routine loops forever
void loop()
{
  int B = 3975;                                    // B value of the thermistor

  int analogValue = analogRead(TEMPPIN);           // read rotary pin
  float resistance=(float)(4095-analogValue)*10000/analogValue;   //get the
resistance of the sensor
  float temperature=1/(log(resistance/10000)/B+1/298.15)-273.15;  //convert to
temperature via datasheet
  Serial.print("analogValue: ");
  Serial.println(analogValue);
  Serial.print("resistance: ");
  Serial.println(resistance);
  Serial.print("temperature: ");
  Serial.println(temperature);
  Serial.println("");
  dispNum((unsigned int) (temperature + 0.5));  // display the voltage
  delay(200);
}

// display a integer value less then 10000
void dispNum(unsigned int num)
{
  int8_t TimeDisp[] = {0x01,0x02,0x03,0x04};    // limit the maximum number

  if(num > 9999) num = 9999;
  TimeDisp[0] = num / 1000;
  TimeDisp[1] = num % 1000 / 100;
  TimeDisp[2] = num % 100 / 10;
  TimeDisp[3] = num % 10;
  tm1637.display(TimeDisp);
}
```

RESULT

The Grove 4-Digit Display shows the temperature, and if you touch the NTC on Temperature Sensor the number will change accordingly.

TIPS

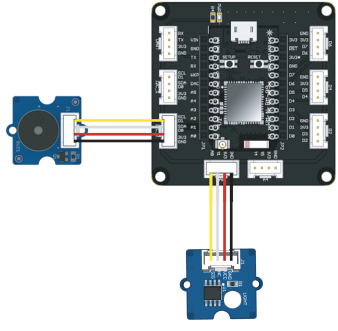
The NTC could only detect temperature approximately.

Example - 06: It's Time to Get Up

WHAT ARE WE DOING HERE?

In this example, the Photon measures the light strength using a light sensor, if it's bright enough, the buzzer would chirp to tell you it's time to get up.

THE CIRCUIT



CODE

```
// ------------------------------------
// Example - 06: It's Time to Get Up
// ------------------------------------
#include "application.h"

// name the pins
#define LIGHTPIN A0
#define BUZZERPIN D1

// This routine runs only once upon reset
void setup()
{
  pinMode(BUZZERPIN, OUTPUT);                    // set user key pin as input
  Serial.begin(9600);
}

// This routine loops forever
void loop()
{
```

```
  int analogValue = analogRead(LIGHTPIN);       // read light sensor pin
  Serial.print("light strength: ");
  Serial.println(analogValue);
  if(analogValue > 1000)                        // if it is bright enough
    digitalWrite(BUZZERPIN, HIGH);              // let the buzzer chirp
  else
    digitalWrite(BUZZERPIN, LOW);               // stop it
  delay(500);
}
```

RESULT

The buzzer chirps only when the light strength is higher than the set threshold.

TIPS

You can modify the threshold to change the trigger condition.

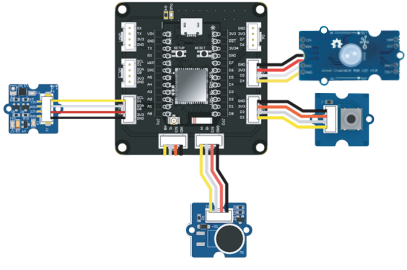
ADVANCED PROJECTS

PROJECT ONE: Direction Recorder

WHAT ARE WE DOING HERE?

This is a special recorder which can remember the direction user set. The vibration motor is activated only when the acceleration sensor is facing the specific direction set by user, and the RGB LED works as an indicator showing different colors at different directions.

THE CIRCUIT



RESULT

When the program starts running, put the acceleration sensor at a fixed spot, then press the button to record the current location. The vibrator will begin to vibrate. Rotate acceleration sensor to another location, vibrator stops vibrating. When the acceleration sensor is put back to the pre-set location, and the brightness around the light sensor is high enough, the vibrator will be activated again to remind user. Also the RGB LED will change its color according to the sensor locations.

TIPS

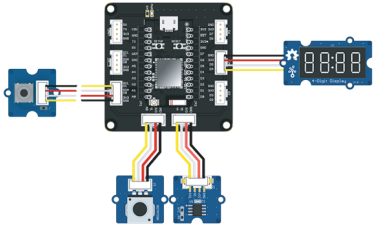
We can modify the code to set the threshold.

PROJECT TWO: Temperature Alarm

WHAT ARE WE DOING HERE?

Here we will use several modules to build a temperature alarm, which can monitor the surrounding temperature and raise the alarm when the temperature exceeds the pre-set threshold. The threshold value can be set through the Grove - Rotary Angle Sensor.

THE CIRCUIT



RESULT

The digital tube will keep showing the real-time temperature until user try to change the temperature threshold through the knob at a range from -20 to 100 degrees Celsius. Then after 2s without any operation, the digital tube will go back to the normal state showing the environmental temperature. The buzzer will chirp to remind the user once the temperature reaches a value higher than the set threshold, and stop the alarm when the temperature drops below the threshold.

TIPS

You can exhale to the temperature sensor to raise the temperature around the sensor.

Technical Support

If you encounter any problems, please refer to the following websites for resources & help.

For all the Grove resources for Photon
https://github.com/Seeed-Studio/Grove_Starter_Kit_for_Photon_Demos

For the details of Grove system
http://www.seeedstudio.com/wiki/GROVE_System

For the tech support for your Starter Kit
http://support.seeedstudio.com

For all the details and support of Photon
https://www.particle.io/

For community support, check out
https://community.particle.io/

ATTENTION

Best to keep away from fire.

Innovate with China

For more information, please visit
seeed.cc/photon