Grove Starter Kit for LaunchPad

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 selection of modules that feature standardized connectors. The base shield allows for easy connection of popular microcontroller devices to interface with the various Grove modules. Each Grove module addresses a unique function & the overall collection of modules expand 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.

About LaunchPad and BoosterPack

The Texas Instruments LaunchPad ecosystem is a set of low-cost & easy-to-use microcontroller development kits that enable developers to rapidly prototype with modular, open source hardware based on TI silicon. BoosterPacks are plug-in modules that can stack on top of the various LaunchPad kits to add additional functionality, such as sensors, displays, wireless modules & more.The Grove Base BoosterPack is a welcome addition to the LaunchPad/BoosterPack ecosystem, enabling any LaunchPad to interface with the growing offering of Grove modules from Seeed Studio. The Grove Base BoosterPack offers a convenient and easy way for rapid prototypers to use more than one hundred Grove modules with standardized connectors, including sensors, actuators, displays, lights, motors and more. http://www.ti.com/launchpad

About Seeed Studio

Seeed is a hardware innovation platform for makers to grow inspirations into differentiated products. Seeed provides accessible technologies with quality, speed and supply chain knowledge. Seeed helps productize 1 to 1,000 pieces 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 Texas Instruments

Texas Instruments Incorporated (TI) is a global semiconductor design and manufacturing company that develops analog ICs and embedded processors. By employing the world's brightest minds, TI creates innovations that shape the future of technology. TI is helping more than 100,000 customers innovate with 80000+ analog ICs & embedded processors, software & the industry’s largest sales/support staff. http://www.ti.com

Have Questions?
If you encounter any problem, please refer to the following websites for resources & help.

For all the Grove resources for LaunchPad
www.seeed.cc/s/boosterpack

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

For tech support for your Grove Base BoosterPack or Starter Kit
http://support.seeedstudio.com

For all the details and support of TI LaunchPad development kits
http://www.ti.com/launchpad

For all details around the Energia software
http://www.energia.nu

For community support, check out http://www.43oh.com

NOTE
This kit is fully compatible with the BoosterPack pin out standard, which is documented here:
http://www.ti.com/byob

Most of the LaunchPad development kits from Texas Instruments adhere to this pinout standard enabling support for the Grove Base BoosterPack. The following examples were developed for the MSP-EXP430F5529LP LaunchPad, which features a 40 pin BoosterPack connector. However any 40 pin LaunchPad with Energia support can leverage these tutorials as well. If you are using a 20 pin LaunchPad (i.e. MSP-EXP430G2), jumpers can be used to route the appropriate pins to the various Grove connectors on the Grove Base BoosterPack.

To ensure compatibility, we recommend that you check the BoosterPack and software pin configurations for your specific LaunchPad kit.

If you are using different LaunchPads, please refer to http://energia.nu/ for additional information. You can find the various pin mapping diagrams at: http://energia.nu/guide/pinmapping/

TABLE OF CONTENTS

LEARN MORE ABOUT THE PRODUCT			**X**
Getting to know the Grove Base BoosterPack		X	
Using the Grove Base BoosterPack	X		
Brief overview of the included Grove modules		X	
Installing the Energia Development Environment		X	
SETTING UP THE ENERGIA IDE			X
(1) Launch Energia		X	
(2) Selecting the Serial Port	X		
(3) Selecting your Board		X	
(4) Preparing Libraries		X	
THE EXAMPLES			**X**
(1) Getting Started (Blinking LED)		X	
(2) Playing Music (Buzzer)		X	
(3) Controlling Other Electronics (Relay)		X	
(4) Displaying the Numbers (4-Digit Display)		X	
(5) Reading the Potentiometer (Rotary Angle Sensor)		X	
(6) Sensing the Light (Light Sensor)	X		
(7) Turning on an LED by Sound (Sound Sensor)		X	
(8) Is Anybody Here? (PIR Motion Sensor)		X	
(9) Taking Care of Your Plants (Moisture Sensor)		X	
(10) Sensing Distance (Ultrasonic Ranger Sensor)	X		
(11) Feeling the Environment (Temperature Humidity Sensor)		X	
Advanced Projects		X	

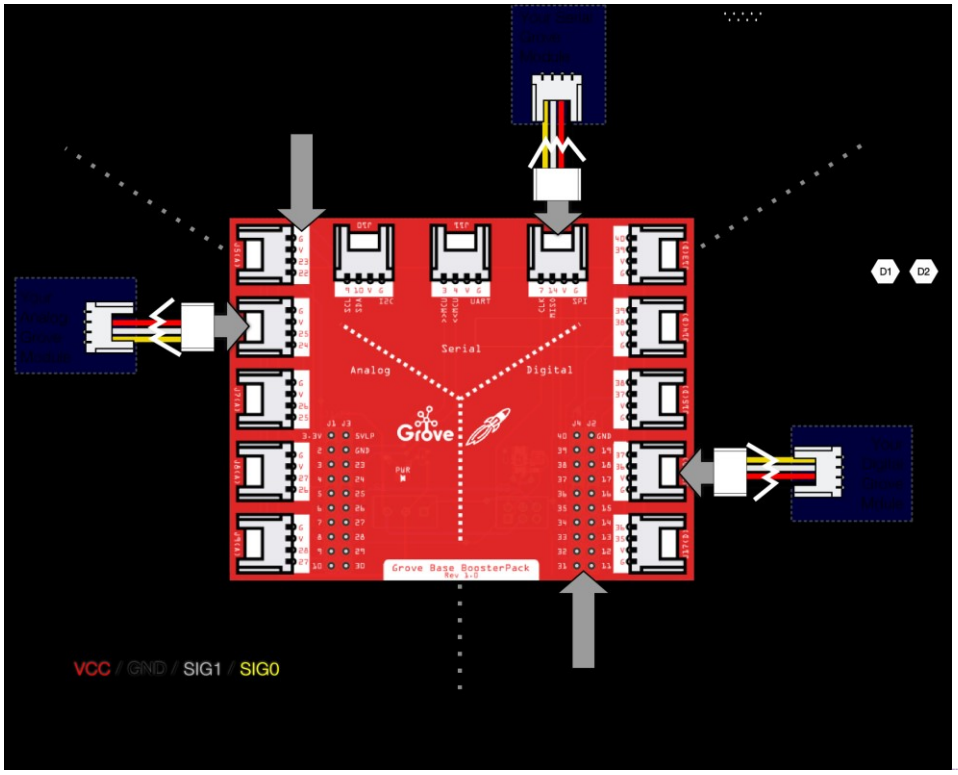
批注 [1]: Add a section on CCS Cloud?

LEARN MORE ABOUT THE PRODUCT

Getting to know the Grove Base BoosterPack

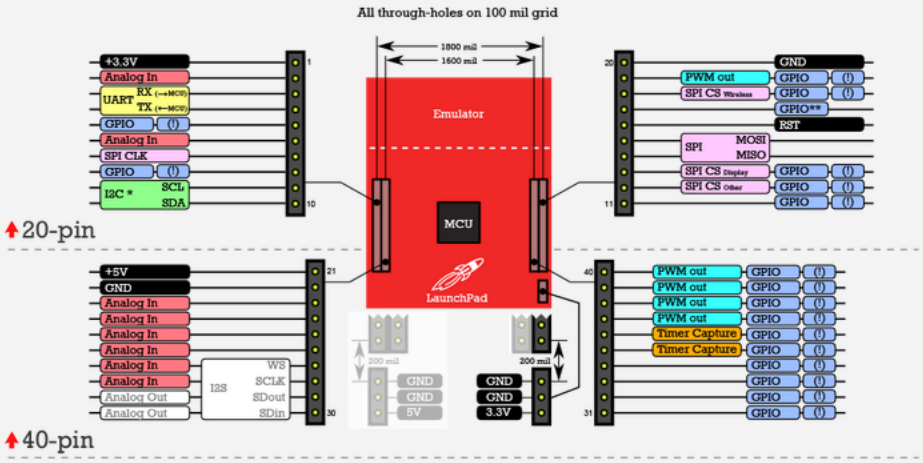
Below is an overview of the Grove Base BoosterPack. The BoosterPack is organized into 3 types of connectors - Digital, Analog & Serial. Depending on the Grove module you want to interface with, you need to ensure you plug it into the appropriate connector type. The BoosterPack has been designed to adhere to the BoosterPack pin out standard so by plugging a Grove module into one of the connectors, the signals will be routed to the appropriate pins of the BoosterPack connector.

Also, notice that the Grove connector has support for 2 signal pins. However, many of the modules only require 1 signal pin (typically SIG0). In that case, SIG1 is a no connect (NC). Since most Grove connectors typically only need 1 pin, the Grove connectors on the Grove Base BoosterPack duplicate some pins of the BoosterPack connector. If you are using a Grove module that needs both signal pins (i.e. a dual-axis joystick -- X axis = SIG1, Y axis = SIG0), you should avoid using the adjacent connector since the repeated pin is taken up by the 2-axis joystick Grove module. This is called a pin conflict, which will cause problems.



批注 [2]: Add a zoomed in shot to show how the pins at the Grove module are the same as those on the BP connector!

BoosterPack Pinout Standard



Legend

Pin function

White boxes indicate functions that are not yet availalable on any existing LaunchPads. However, these functions are defined in the standard to 'future-proof' the pinout.

GPIO**

Some LaunchPads do not comply with this GPIO pin (i.e. MSP-EXP430G2 LaunchPad uses this pin for programming/debugging). De-prioritize this when making a BoosterPack.

(!)

The exclamation point indicates the GPIO is interruptible.

I2C*

Most LaunchPads have true hardware enabled I2C at these pins, but some LaunchPads may not. If no hardware I2C is available a software-emulated I2C is needed to adhere to standard.

Check the documentation of specific LaunchPad boards to confirm compatiability with the BoosterPack standard. While most LaunchPads comply with the standard, there may be some deviations.

Also note that LaunchPads usually offer more than one function at a specific pin. Be aware of pin muxing for additional features and capabilities at each pin.

Using the Grove Base BoosterPack

Using a 40-pin LaunchPad (i.e. MSP-EXP430F5529LP, MSP-EXP432P401R, EK-TM4C123GXL, etc) ***Recommended***

The Grove Base BoosterPack was designed in a way to leverage pin numbers 21 through 40. Visually, these are the inner 20 pins of the BoosterPack connector. The pins are connected as shown in the table below:

Using the table below, developers should be able to read an analog value from a Grove module (i.e. potentiometer/turn knob) that is connected to Grove connector 'J6' by using the analogRead(24) API call with Energia.

Connector Type	Grove connector Label	GND	VCC	SIG1 (connection to the BoosterPack pin)	SIG0 (connection to the BoosterPack pin) *
Analog	J5	GND	3.3V	23 (analog capable pin)	22 (analog capable pin)
Analog	J6	GND	3.3V	25 (analog capable pin)	24 (analog capable pin)
Analog	J7	GND	3.3V	26 (analog capable pin)	25 (analog capable pin)
Analog	J8	GND	3.3V	27 (analog capable pin)	26 (analog capable pin)
Analog	J9	GND	3.3V	28 (analog capable pin)	27 (analog capable pin)
I2C	J10	GND	3.3V	10 (I2C SDA)	9 (I2C SCL)
UART	J11	GND	3.3V	4 (UART to MCU)	3 (UART from MCU)
SPI	J12	GND	3.3V	14 (SPI MISO)	7 (SPI CLK)
Digital	J13	GND	3.3V	39 (Digital/PWM pin)	40 (Digital/PWM pin)
Digital	J14	GND	3.3V	38 (Digital/PWM pin)	39 (Digital/PWM pin)
Digital	J15	GND	3.3V	37 (Digital/PWM pin)	38 (Digital/PWM pin)
Digital	J16	GND	3.3V	36 (Digital/PWM pin)	37 (Digital/PWM pin)
Digital	J17	GND	3.3V	35 (Digital/PWM pin)	36 (Digital/PWM pin)

Using a 20-pin LaunchPad (i.e., MSP-EXP430G2, MSP-EXP430FR5969, MSP-EXP430FR4133, etc)

If you are using a 20-pin LaunchPad, you can use jumpers or jumper wire to make the appropriate connections between a Grove connector & the BoosterPack connector.

Using your specific LaunchPad's pin out diagram, you can physically/electrically connect the Grove module to the appropriate pin. Pinout diagrams for each LaunchPad are available here: http://energia.nu/pin-maps/

With the help of these pin diagrams, you know which pin has the function you need. If you want to use Grove connector J8 for an analog Grove module (i.e. potentiometer knob), you can use the Energia pin maps to identify an analog-capable pin on the BoosterPack connector. Using a jumper or jumper wire, you can connect pin number 26 with the analog-capable pin that is available. For example, if you are using an MSP-EXP430G2 LaunchPad, you can use a jumper or jumper wire to connect pin 26 with pin 6.

[Add a picture here to show jumper/wire making the above connection]

批注 [3]: PICTURE!

overview of the included Grove modules

DIGITAL MODULES

批注 [4]: For each module, should we have a small diagram highlighting the appropriate grove connector?

The Grove-Buzzer
What is it?
The Grove Buzzer module features a piezo buzzer that can be connected to digital outputs. It will emit a tone when the output is high. Alternatively it can be connected to a pulse-width modulation (PWM) capable output pin to generate various tones and sounds.

How to connect it?
This is a Digital module that should connect to a digital or PWM-capable pin (i.e. Grove connector J13 - J17)

How to program it?
In Energia, you can use either the digitalWrite(pin, state) API to buzz the piezo at a fixed tone. Or, you can use the analogWrite(pin, dutyCycle) or tone(pin, frequency, duration) function if you want to modulate a specific frequency or tone.

More details:
http://www.seeedstudio.com/depot/Grove-Buzzer-p-768.html

批注 [5]: Vanity/alias URL

批注 [6]: Applies to all module overviews

The Grove-Relay
What is it?
The Relay is a digital normally- open switch that controls a relay capable of switching much higher voltages and currents than your normal LaunchPad boards. When set to HIGH, the on-board LED on the Grove-Relay module will light up and the relay will close to allow current to flow. The peak voltage capability is 250V at 10 amps.

How to connect it?
This is a Digital module that should connect to a digital pin (i.e. Grove connector J13 - J17)

How to program it?
In Energia, you can use the digitalWrite(pin, state) API to close the relay.

More details:
http://www.seeedstudio.com/depot/Grove-Relay-p-769.html

The Grove-PIR Motion Sensor
What is it?

When any motion occurs within the detection range (3m by default), the sensor outputs a HIGH signal on its SIG0 pin.

How to connect it?
This is a digital module that should connect to a digital pin (i.e. Grove connector J13 - J17). This requires just a single signal pin, SIG0.

How to program it?
In Energia, you should be able to use the digitalRead(pin) API. This function will return back the state of the incoming signal from the motion sensor. If motion was detected, this function will return a 1, else 0.

More details:
http://www.seeedstudio.com/depot/Grove-PIR-Motion-Sensor-p-802.html?cPath=25_31

The Grove-4-Digit Display
What is it?
The 4-Digit 7-segment display module usually takes 12 pins to control, but in this Grove module, a TM1637 was used to allow us to control the display with only two pins. In addition to displaying data using these two pins we can even change the brightness of the display. For projects that require a numeric display this board would be an excellent choice!

How to connect it?
This is a digital module that requires 2 digital pins (i.e. Grove connector J13 - J17). This requires both SIG1 and SIG0 (one clock pin & one data pin)

How to program it?
In Energia, you will need the Grove 4-Digit Display (DigitalTube) library. This library is available in the Grove wiki, which is linked below. Along with the library are a few example sketches that showcase how to use the library to display characters on the display.

If the library is installed in Energia, you can find it at:
File > Examples > Libraries > DigitalTube > [Examples]

More details:
http://www.seeedstudio.com/depot/Grove-4Digit-Display-p-1198.html?cPath=34_787

The Grove-Ultrasonic Ranger Sensor

What is it?
This Grove-ultrasonic sensor is a non-contact distance measurement module. This sensor offers industrial performance & provides a detecting range of 3cm to 4 meters (~1cm of resolution). It features a transmitter & receiver. The module transmits ultrasonic waves that reflect off the nearest object & the receiver detects the “echo”. The time it takes for the receiver to see the reflected signal is how we can detect distance.

How to connect it?
This is a digital module that should connect to a digital pin (i.e. Grove connector J13 - J17). This requires just a single signal pin, SIG0.

How to program it?
In Energia, you will need the Grove-Ultrasonic Ranger Sensor library. This library is available in the Grove wiki, which is linked below. Along with the library are a few example sketches that showcase how to use the library to display the distance on the display (in either cm or inches).

If the library is installed in Energia, you can find it at:
File > Examples > Libraries > UltrasonicRanger > [Examples]

批注 [7]: File path should be: file>sketchbook>launchpad kit>grove modules>ultrasonic ranger

More details:
http://www.seeedstudio.com/depot/BackorderGroveUltrasonic-Ranger-p-960.html?cPath=25_31

ANALOG MODULES

The Grove-Light Sensor
What is it?
The Light sensor module detects the light intensity of the environment. This is an analog Grove module that will return a voltage between 0V and 3.3V. The brighter the environment, the higher the output voltage.

How to connect it?
This is an analog module that should connect to an analog pin (i.e. Grove connector J5 - J9). This requires just a single signal pin, SIG0.

How to program it?
In Energia, you will be able to use the analogRead(pin) API. This function will return back the raw analog readings coming from the sensor. Depending on the resolution of the LaunchPad ADC channels, the API will return a number between 0-255 (8-bit ADC), 0 - 1024 (10-bit ADC), or 0 - 4095 (12-bit ADC).

More details:
http://www.seeedstudio.com/depot/Grove-Light-Sensor-p-746.html?cPath=25_27

The Grove-Sound Sensor
What is it?
The Sound sensor module is a simple microphone. It can be used to determine the acoustic noise level of the environment. The resistance of the sensor decreases when the sound intensity of the environment increases. The value of output can be adjusted by the potentiometer knob on the Grove-Sound Sensor module.

How to connect it?
This is an analog module that should connect to an analog pin (i.e. Grove connector J5 - J9). This requires just a single signal pin, SIG0.

How to program it?
In Energia, you should be able to use the analogRead(pin) API. This function will return back the raw analog readings coming from the sensor. Depending on the resolution of the LaunchPad ADC channels, the API will return a number between 0-255 (8-bit ADC), 0 - 1024 (10-bit ADC), or 0 - 4095 (12-bit ADC).

More details:
http://www.seeedstudio.com/depot/Grove-Sound-Sensor-p-752.html

The Grove-Moisture Sensor
What is it?
This Moisture Sensor can be used to detect the moisture of soil or judge if there is water around the sensor. The Moisture Sensor is very user-friendly: just insert it into the soil and then read the data. Thanks to this sensor, you can easily see when your house plant is thirsty! Perhaps you can have LaunchPad print out a message if moisture levels are too low: “Hey, I am thirsty now! Please give me some water.”

How to connect it?
This is an analog module that should connect to an analog pin (i.e. Grove connector J5 - J9). This requires just a single signal pin, SIG0.

How to program it?
In Energia, you should be able to use the analogRead(pin) API. This function will return back the raw analog readings coming from the sensor. Depending on the resolution of the LaunchPad ADC channels, the API will return a number between 0-255 (8-bit ADC), 0 - 1024 (10-bit ADC), or 0 - 4095 (12-bit ADC).

More details:
http://www.seeedstudio.com/depot/Grove-PIR-Motion-Sensor-p-802.html?cPath=25_31

The Grove-Temperature Humidity Sensor
What is it?
This module is an accurate temperature and humidity sensor. It consists of a capacitive sensor element used for measuring relative humidity and a negative temperature coefficient (NTC) thermistor used for measuring temperature.

How to connect it?
This is an analog module that should connect to an analog pin (i.e. Grove connector J5 - J9). This requires just a single signal pin, SIG0.

How to program it?
In Energia, you will need the Grove-Temperature and Humidity Sensor library. This library is available in the Grove wiki, which is linked below. Along with the library are a few example sketches that showcase how to use the library to display the temperature & humidity readings. Note that the sensor on this module is the *DHT22*.

NOTE: Since several DHT devices share this library, be sure to comment/uncomment the appropriate sensor within the library.

If the library is installed in Energia, you can find it at:
File > Examples > Libraries > Humidity_Temperature_Sensor > [Examples]

More details:
http://www.seeedstudio.com/depot/Grove-TemperaureHumidity-Sensor-Pro-p-838.html?cPath=25_125

The Grove-Rotary Angle Sensor
What is it?
The potentiometer module produces an analog output between 0 and Vcc on SIG0. The angular range is 300 degrees with a linear change in the output voltage. This may also be known as a rotary angle sensor.

How to connect it?
This is an analog module that should connect to an analog pin (i.e. Grove connector J5 - J9). This requires just a single signal pin, SIG0.

How to program it?
In Energia, you should be able to use the analogRead(pin) API. This function will return back the raw analog readings coming from the sensor. Depending on the resolution of the LaunchPad ADC channels, the API will return a number between 0-255 (8-bit ADC), 0 - 1024 (10-bit ADC), or 0 - 4095 (12-bit ADC).

More details:

http://www.seeedstudio.com/depot/Grove-Rotary-Angle-Sensor-p-770.html?cPath=85_52

Installing the Energia IDE

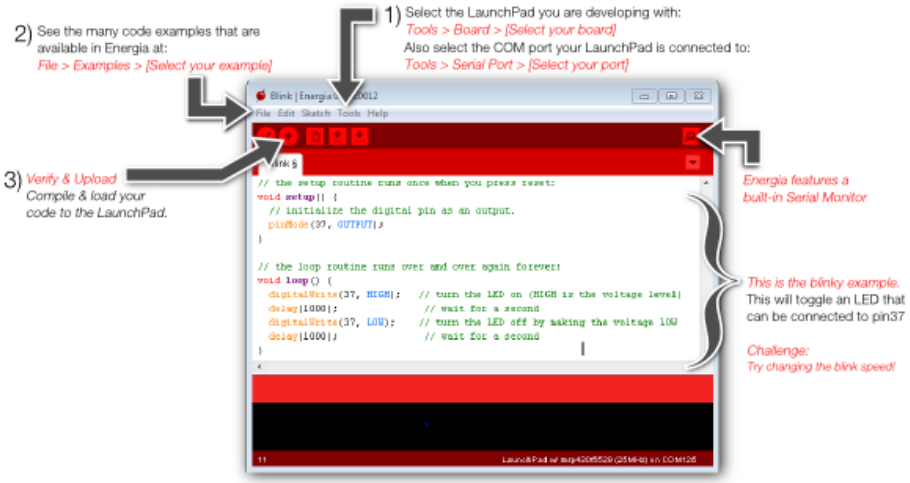
Installing Energia

Energia is an open-source and community-driven development environment for LaunchPad kits. It is based on the popular Wiring framework, offering a large collection of easy-to-use functions and APIs. Energia support is available for many LaunchPad development kits across various TI microcontroller devices: MSP430/MSP432, TM4C, C2000, and CC3200.

Download the latest version of Energia @
www.energia.nu/guide

Note that Energia is available for Windows, Linux & Mac. The link above explains how to install the required drivers & Energia on these various OS options. Once downloaded & installed, you will need to configure Energia for the specific LaunchPad you are using.

Quick Overview of the Energia IDE



Energia is available for Windows, Linux & Mac OS X.

SETTING UP THE ENERGIA IDE

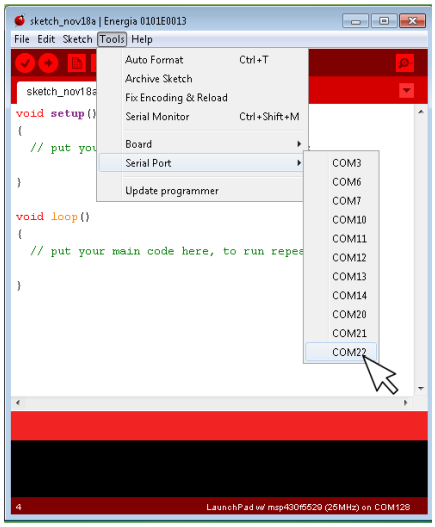
(1) Launch Energia

Once installed, go ahead and plug in your LaunchPad into the USB port of your computer. Then, double-click Energia.exe (Windows users) / Energia.app (Mac OS X users). Energia will start and an empty Sketch window will appear.

(2) Selecting the Serial Port

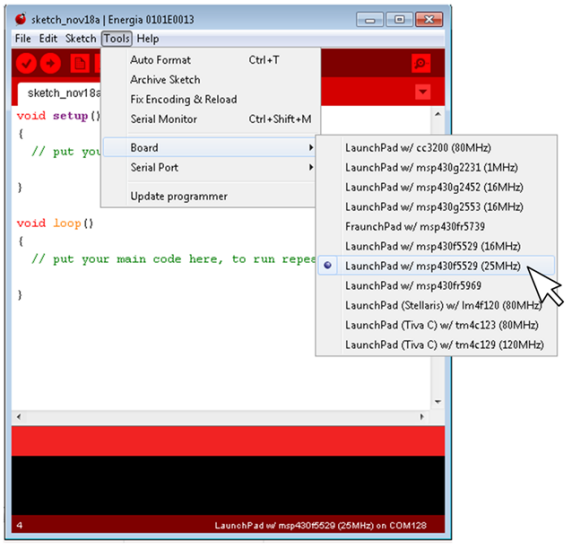
Select Serial Port from the Tools menu to view the available serial ports. On windows, they will be listed as COMXXX port and usually a higher number is the LaunchPad com port. On Mac OS X they will be listed as /dev/cu.uart-XXXX. On Mac OS X make sure to select the /dev/cu.uart-XXXX and NOT the /dev/tty.uart-XXX. If needed, check your computer's device manager to verify which port to use.

Navigate to *Tools* > *Serial Port [select your serial port]*



(3) Select your board

Also, ensure that Energia is configured for the specific LaunchPad device you are using. You can do so by navigating to *Tools* > *Board* > *[select your LaunchPad]*



(4) Preparing Libraries

Go to **https://github.com/Seeed-Studio/LaunchPad_Kit** http://www.energia.nu/grovekitcode and download the latest zip file. This zip file includes example sketches & required libraries for each of the Grove modules in the Starter Kit. Unzip the zip file, and then copy the contents into the generic Energia folder on your computer.

批注 [8]: Add vanity URL/alias & host a zip file on Energia site or Seeed site.

- In Windows, you can unzip contents into this folder: [USERDIR] >My Documents > Energia
- In Mac, you can unzip contents into this folder: /Users/USERDIR]/Documents/Energia.
- In Linux, you can unzip contents into this folder: /home/[USERDIR]/sketchbook

Here is how it should look in your directory:



Once you have the folder structure as shown above, you may have to restart Energia in order for the IDE to discover these new examples/libraries.

Note that these libraries and example projects are open source and available on github: **https://github.com/Seeed-Studio/LaunchPad_Kit**

THE EXAMPLES

Example - 01 : Getting Started (Blinking LED)

WHAT ARE WE DOING HERE?

In this example, we will not be using any Grove modules. First, we'll learn how to blink one of the on-board LEDs that are featured on the LaunchPad development kit itself. All LaunchPad kits feature a few user LEDs, typically a red LED. By running the example code, you will see the LED switched on and off once per second.

THE CIRCUIT

We used the on-board red LED that is available on most LaunchPad kits. You don't need to add any other components.

CODE

This example is available here: *File* > *Example* > *Basics* > *Blink*

```
/*
  Blink
  The basic Energia example.
  Turns on an LED on for one second, then off for one second, repeatedly.
  Change the LED define to blink other LEDs.

  Hardware Required:
  * LaunchPad with an LED

  This example code is in the public domain.
*/

// most launchpads have a red LED
#define LED RED_LED

//see pins_energia.h for more LED definitions
//#define LED GREEN_LED

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(LED, OUTPUT);
}

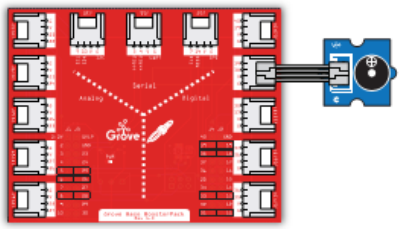
// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(LED, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);               // wait for a second
  digitalWrite(LED, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);               // wait for a second
}
```

RESULT
The LED on board will periodically switch on and off, with each state lasting for 1 second.

TIPS
The 2 main APIs we are using are digitalWrite(pin, state) & delay(milliseconds). Note that we need to also use the pinMode(pin, direction) function to set the appropriate pin as an OUTPUT port before we can use the digitalWrite function for toggling it ON & OFF. Try changing the blink speed or controlling another LED!

Example - 02 : Playing Music (Buzzer)
WHAT ARE WE DOING HERE?
This example shows how to use the Grove buzzer module to play melodies. It sends a square wave of the appropriate frequency to the buzzer, generating the corresponding tone.

THE CIRCUIT



Parts:

- Grove-Buzzer x1
- Grove cable x1
- Grove-Base BoosterPack x1
- LaunchPad Development Kit

Connections:
Plug the Buzzer module into Grove connector J14 on the BoosterPack. This will link the SIG pin of the buzzer to pin 39 of the LaunchPad.

CODE
This example is available here: *File > Sketchbook > LaunchPad_Kit > Grove_Modules > buzzer*

```
/*
  Buzzer
 The example use a buzzer to play melodies. It sends a square wave of the
 appropriate frequency to the buzzer, generating the corresponding tone.

 The circuit:
 * Buzzer attached to pin39 (J14 plug on Grove Base BoosterPack)
 * one side pin (either one) to ground
 * the other side pin to VCC
 * LED anode (long leg) attached to RED_LED
 * LED cathode (short leg) attached to ground

 * Note:


 This example code is in the public domain.

 http://www.seeedstudio.com/wiki/index.php?title=GROVE_-_Starter_Kit_v1.1b#Grove_-
_Buzzer

*/

/* Macro Define */
#define BUZZER_PIN              39          /* sig pin of the buzzer */

int length = 15;        /* the number of notes */
char notes[] = "ccggaagffeeddc ";
int beats[] = { 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 2, 4 };
int tempo = 300;

void setup()
{
```

批注 [9]: Update with latest code

```
    /* set buzzer pin as output */
    pinMode(BUZZER_PIN, OUTPUT);
}

void loop()
{
    for(int i = 0; i < length; i++) {
        if(notes[i] == ' ') {
            delay(beats[i] * tempo);
        } else {
            playNote(notes[i], beats[i] * tempo);
        }
        delay(tempo / 2);    /* delay between notes */
    }

}

/* play tone */
void playTone(int tone, int duration) {
  for (long i = 0; i < duration * 1000L; i += tone * 2) {
    digitalWrite(BUZZER_PIN, HIGH);
    delayMicroseconds(tone);
    digitalWrite(BUZZER_PIN, LOW);
    delayMicroseconds(tone);
  }
}

void playNote(char note, int duration) {
  char names[] = { 'c', 'd', 'e', 'f', 'g', 'a', 'b', 'C' };
  int tones[] = { 1915, 1700, 1519, 1432, 1275, 1136, 1014, 956 };

  // play the tone corresponding to the note name
  for (int i = 0; i < 8; i++) {
    if (names[i] == note) {
      playTone(tones[i], duration);
    }
  }
}
```

RESULT

Your LaunchPad will play music! “Twinkle Twinkle Little Star” or “The Alphabet Song”

TIPS

The calculation of the tones is made using the mathematical operation:

timeHigh = period / 2 = 1 / (2 * toneFrequency)

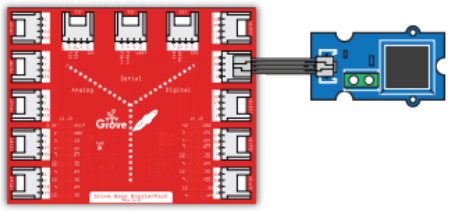
Where the different tones are described as in the table below:

Note	frequency	period	timeHigh
c	261 Hz	3830	1915
d	294 Hz	3400	1700
e	329 Hz	3038	1519
f	349 Hz	2864	1432
g	392 Hz	2550	1275
a	440 Hz	2272	1136
b	493 Hz	2028	1014
c	523 Hz	1912	956

Example - 03 : Controlling other electronics (Relay)
WHAT ARE WE DOING HERE?
This example showcases how to use the Grove-relay module to control a larger load, i.e. a desk lamp light. A 3V voltage signal can cause the relay to switch on, allowing current to flow through the connected appliance.

WARNING special care should be taken when using the Relay.

THE CIRCUIT



Parts:
- Grove-Relay x1
- Grove cable x1
- Grove-Base BoosterPack x1
- LaunchPad Development Kit

Connections:
Plug the relay module into Grove connector J14 on the BoosterPack. This will link the SIG pin of the relay to pin 39 of the LaunchPad.

CODE
This is essentially the same as the blink LED code from Example #1. In this case, we are simply opening & closing the relay switch once per second.
This example is available here: *File* > *Sketchbook* > *LaunchPad_Kit* > *Grove_Modules* > *relay*

批注 [10]: update to proper file location

批注 [11]: update to latest code

```
/*
Relay
The basic Energia example.
This example code is in the public domain.
*/

#define RELAY_PIN 39

// the setup routine runs once when you press reset:
void setup() {
        pinMode(RELAY_PIN, OUTPUT); // initialize the digital pin as an output.
}

// the loop routine runs over and over again forever:
void loop() {
        digitalWrite(RELAY_PIN, HIGH); // turn the relay on (HIGH is the voltage level)
        delay(1000);   // wait for a second
        digitalWrite(RELAY_PIN, LOW);   // turn the relay o by making the voltage LOW
        delay(1000);   // wait for a second
```

```
}
```

RESULT

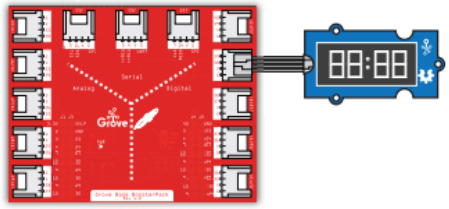
Once a second, the relay switch will open & close. You should hear a clicking noise whenever the relay is switched open or close.

TIPS

Please pay attention to the maximum value of the current and voltage marked on the Relay.

Example - 04 : Displaying the Numbers (4-Digit Display)
WHAT ARE WE DOING HERE?
This example demonstrates how to display some digital numbers using a Grove-4-Digit Display.

THE CIRCUIT



Parts:
- Grove-4-Digit Display x1
- Grove cable x1
- Grove-Base BoosterPack x1
- LaunchPad Development Kit

Connections:
Plug the Grove-4-Digit Display module into Grove connector J14 on the BoosterPack. This will link the 2 SIG pins of the 4 digit module to pins 38 (DIO) & 39 (CLK) of the LaunchPad.

CODE

This example is available here: *File* > *Examples* > *DigitalTube* > *NumberFlow*

批注 [12]: put header file inside of example folders

```
/*
 * TM1637.cpp
 * A library for the 4 digit display
 *
 * Copyright (c) 2012 seeed technology inc.
 * Website    : www.seeed.cc
 * Author     : Frankie.Chu
 * Create Time: 9 April,2012
 * Change Log :
 *
 * The MIT License (MIT)
 *
 * Permission is hereby granted, free of charge, to any person obtaining a copy
 * of this software and associated documentation files (the "Software"), to deal
 * in the Software without restriction, including without limitation the rights
 * to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
 * copies of the Software, and to permit persons to whom the Software is
 * furnished to do so, subject to the following conditions:
 *
 * The above copyright notice and this permission notice shall be included in
 * all copies or substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR
```

```
 * IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY,
 * FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT
SHALL THE
 * AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR
OTHER
 * LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
ARISING FROM,
 * OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
DEALINGS IN
 * THE SOFTWARE.
 */
#include "TM1637.h"
#define CLK 39 //pins definitions for TM1637 and can be changed to other ports
#define DIO 38
TM1637 tm1637(CLK,DIO);
void setup()
{
  tm1637.init();
  tm1637.set(BRIGHT_TYPICAL);//BRIGHT_TYPICAL = 2,BRIGHT_DARKEST =
0,BRIGHTEST = 7;
}
void loop()
{
  int8_t NumTab[] = {0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15};//0~9,A,b,C,d,E,F
  int8_t ListDisp[4];
  unsigned char i = 0;
  unsigned char count = 0;
  delay(150);
  while(1)
  {
    i = count;
    count ++;
    if(count == sizeof(NumTab)) count = 0;
    for(unsigned char BitSelect = 0;BitSelect < 4;BitSelect ++)
    {
      ListDisp[BitSelect] = NumTab[i];
      i ++;
      if(i == sizeof(NumTab)) i = 0;
    }
    tm1637.display(0,ListDisp[0]);
    tm1637.display(1,ListDisp[1]);
    tm1637.display(2,ListDisp[2]);
    tm1637.display(3,ListDisp[3]);
    delay(300);
  }
}
```

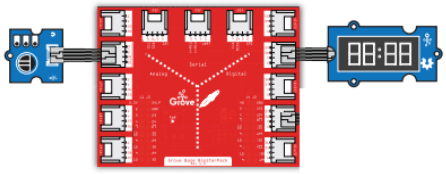
批注 [13]: not correct pins in code

RESULT
Numbers will cycle through the display one by one. Pretty, isn’t it?

TIPS
The Energia library offers control of both the content and the brightness of the display using just 2 digital pins of the LaunchPad.

Example - 05 : Reading the Potentiometer (Rotary Angle Sensor)
This example shows how to read the analog output coming from the Grove potentiometer module. We will be combining a few Grove modules in this example! By turning the potentiometer knob, we will display the analog reading value on the Grove 4-Digit display.

THE CIRCUIT



Parts:
- Grove-Rotary Angle Sensor x1
- Grove-4-Digit Display x1
- Grove cable x2
- Grove-Base BoosterPack x1
- LaunchPad Development Kit

Connections:
Plug the Grove-Rotary Angle Sensor module into Analog Grove connector J6 on the BoosterPack. This will link the SIG pins of the potentiometer to pin 24 of the LaunchPad.

Plug the Grove-4-Digit Display module into Grove connector J14 on the BoosterPack. This will link the 2 SIG pins of the 4 digit module to pins 38 (DIO) & 39 (CLK) of the LaunchPad.

CODE
This example is available here:
File > *Sketchbook* > *LaunchPad_Kit* > *Grove_Modules* > *rotary_angle_sensor*

批注 [14]: Update file path and add header files to sketch folder

```
/*
  Rotary Angle Sensor
 Demonstrates analog input by reading an analog sensor on J16 of the Grove Base
BoosterPack. The speed of the red LED on the LaunchPad will change depending on the
position of the potentiometer knob. This example will also display the analog reading value on
the Grove 4-digital display.

 The circuit:
* Potentiometer attached to pin 24 (J6 on Grove Base BoosterPack)
* center pin of the potentiometer to the analog pin
* one side pin (either one) to ground
* the other side pin to VCC (3.3V)

* Note: Because of unstable of the voltage, the value of the rotary angle sensor
        varies slightly from run to run even you don't touch it.
```

```
 Created by Oliver Wang

 This example code is in the public domain.

 http://www.seeedstudio.com/wiki/GROVE_-_Starter_Kit_v1.1b#Grove_-
_Rotary_Angle_Sensor
 */

#include "TM1637.h"

/* Macro Define */
#define CLK              39              /* 4-digital display clock pin */
#define DIO              38              /* 4-digital display data pin */
#define ROTARY_ANGLE_P   24              /* pin of rotary angle sensor */

/* Global Variables */
TM1637 tm1637(CLK, DIO);                 /* 4-digital display object */
int analog_value = 0;                    /* variable to store the value coming from rotary angle
sensor */

int8_t bits[4] = {0};                    /* array to store the single bits of the value */

/* the setup() method runs once, when the sketch starts */
void setup() {

   /* Initialize 4-digital display */
   tm1637.init();
   tm1637.set(BRIGHT_TYPICAL);

}

/* the loop() method runs over and over again */
void loop() {

   analog_value = analogRead(ROTARY_ANGLE_P);      /* read the value from the sensor */
   memset(bits, 0, 4);                             /* reset array when we use it */
   for(int i = 3; i >= 0; i--) {
      /* get single bits of the analog value */
      bits[i] = analog_value % 10;
      analog_value = analog_value / 10;
      tm1637.display(i, bits[i]);                  /* display by 4-digital display */
   }
   delay(100);
}
```

RESULT

A digital number corresponding to the analog reading coming from the potentiometer knob will display on the 4-Digit Display module. Note that depending on the resolution of your LaunchPad

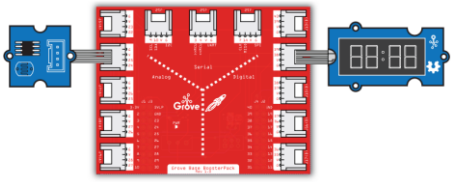
kit's ADC (analog-to-digital converter), your range may vary. For an 8-bit ADC, a number from 0-255 will show. For a 10-bit ADC, a number from 0-1023 will show. For a 12-bit ADC, a number from 0-4095 will be displayed on 4-Digit Display. Turn the knob to see the values change on the 4-Digit Display!

TIPS

The rotary angle sensor is a very useful device. When it is connected with VCC (3.3V) across its outer pins, the middle pin will present the value of the signal between 0 and VCC (3.3V). This is a useful component for receiving inputs from the outside world, i.e. human input.

Example - 06 : Sensing the Light (Light Sensor)
WHAT ARE WE DOING HERE?
This example reads the amount of light incident on the Grove-Light Sensor and displays the reading on the Grove 4-Digit Display. This is similar to the previous example except we are now using the ambient light sensor instead of the potentiometer rotary angle knob.

THE CIRCUIT



Parts:
- Grove-Ambient Light Sensor x1
- Grove-4-Digit Display x1
- Grove cable x2
- Grove-Base BoosterPack x1
- LaunchPad Development Kit

Connections:
Plug the Grove-Ambient Light Sensor module into Analog Grove connector J6 on the BoosterPack. This will link the SIG pins of the potentiometer to pin 24 of the LaunchPad.

Plug the Grove-4-Digit Display module into Grove connector J14 on the BoosterPack. This will link the 2 SIG pins of the 4 digit module to pins 38 (DIO) & 39 (CLK) of the LaunchPad.

CODE
This example is available here: *File* > *sketchbook* > *LaunchPad_Kit* > *Grove_Modules* > *light_sensor*
Note that this code is pretty much identical to the rotary angle example above.

批注 **[15]:** Update file path and add header file to sketch folder

```
/*
  Light Sensor
 A simple program that display the value of light incident on the grove-light-sensor
 by grove-4-digital-display, this example is similar to grove-rotary-angle-sensor

 The circuit:
 * grove-4-digital-display attached to dio9&dio10 (IIC plug on Grove Base BoosterPack)
 * sig pin of the light sensor to the analog pin
 * one side pin (either one) to ground
 * the other side pin to VCC

 * Note:
```

```
 Created by Oliver Wang

 This example code is in the public domain.

 http://www.seeedstudio.com/depot/Grove-Light-Sensor-p-746.html?cPath=25_27
 */

#include "TM1637.h"

/* Macro Define */
#define CLK             39              /* 4-digital display clock pin */
#define DIO             38              /* 4-digiral display data pin */
#define LIGHT_SENSOR    24              /* pin of grove light sensor */

/* Global Varibles */
TM1637 tm1637(CLK, DIO);                /* 4-digital display object */
int analog_value = 0;                   /* varible to store the value coming from rotary angle sensor
*/
int8_t bits[4] = {0};                   /* array to store the single bits of the value */

/* the setup() method runs once, when the sketch starts */
void setup() {

   /* Initialize 4-digital display */
   tm1637.init();
   tm1637.set(BRIGHT_TYPICAL);

}

/* the loop() method runs over and over again */
void loop() {

   analog_value = analogRead(LIGHT_SENSOR);     /* read the value from the sensor */
   memset(bits, 0, 4);                          /* reset array when we use it */
   for(int i = 3; i >= 0; i--) {
      /* get single bits of the analog value */
      bits[i] = analog_value % 10;
      analog_value = analog_value / 10;
      tm1637.display(i, bits[i]);               /* display by 4-digital display */
   }
}
```

RESULT

The value will be displayed on 4-Digit Display when you shine a light on the light sensor. Note that depending on the resolution of your LaunchPad kit's ADC (analog-to-digital converter), your range may vary. For an 8-bit ADC, a number from 0-255 will show. For a 10-bit ADC, a number

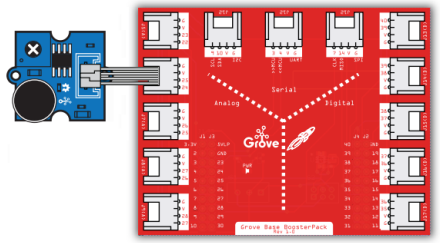
from 0-1023 will show. For a 12-bit ADC, a number from 0-4095 will be displayed on 4-digital-display. Watch the values change as you present light to the sensor.

TIPS
Now that we can tell how bright it is in the environment, try to turn an LED on when the room is too dark.

Example - 07 : Turning on an LED by Sound (Sound Sensor)

WHAT ARE WE DOING HERE?
This example program shows how to use the sound sensor (microphone). When the incoming sound amplitude exceeds a threshold, we will turn an LED on for 1 second.

THE CIRCUIT



Parts:
- Grove-Sound Sensor x1
- Grove cable x1
- Grove-Base BoosterPack x1
- LaunchPad Development Kit

Connections:
Plug the sound sensor module into Analog Grove connector J6 on the BoosterPack. This will link the SIG pin of the microphone to pin 24 of the LaunchPad.

CODE

This example is available here:
File > *Sketchbook* > *LlaunchPad_Kit* > *Grove_Modules* > *sound_sensor*
Note that depending on your LaunchPad's ADC resolution, you may have to adjust the THRESHOLD_VALUE for the desired sensitivity. For 8-bit ADC 200 is acceptable, for 10-bit ADC 500, and for 12-bit 3000 is good.

批注 [16]: update file path

```
/*
 Sound Sensor
 A simple program demonstrate sound sensor senses a sound that is up to the threshold you set
 in the code, the LED is on for 1s.

 The circuit:
 * sig pin of the sound sensor to the analog pin 24 (J6 plug on Grove Base BoosterPack)
 * one side pin (either one) to ground
 * the other side pin to +VCC
 * LED anode (long leg) attached to RED_LED
 * LED cathode (short leg) attached to ground

 * Note:


 This example code is in the public domain.
```

```
 http://www.seeedstudio.com/depot/Grove-Sound-Sensor-p-752.html?cPath=25_128
 */

#define SOUND_SENSOR      24          /* sound sensor pin */
#define LED               RED_LED     /* LED pin */
#define THRESHOLD_VALUE   3000        /* Depending on your LaunchPad's ADC
resolution you may have to adjust the threshold */

#define ON                HIGH                /* led on */
#define OFF               LOW                 /* led off */
#define _handle_led(x)    digitalWrite(LED, x)    /* handle led */

/* Global Varibles */
int sound_value = 0;

void setup() {

   /* Initialize led pin */
   pinMode(LED, OUTPUT);
   digitalWrite(LED, LOW);
}

void loop() {

   /* read the sound value */
   sound_value = analogRead(SOUND_SENSOR);
   /* if the value is larger than threshold, turn on led */
   if(sound_value > THRESHOLD_VALUE) {
      _handle_led(ON);
      delay(1000);
   }
   _handle_led(OFF);
}
```

批注 [17]: explain how to change this better in above paragraph

RESULT

The red LED will be turned on for about 1 second when you clap your hands or snap your fingers near the microphone.

TIPS

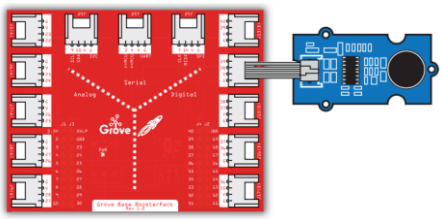
Analog Sound Sensor is typically used for detecting the loudness in your surroundings. Noise is a continuous signal. For a given sample rate (fs), the bandlimit (B) for perfect reconstruction is B ≤ fs/2 according to the Nyquist Shannon sampling theorem.

Example - 08 : Is Anybody Here? (PIR Motion Sensor)

WHAT ARE WE DOING HERE?

The following sketch demonstrates a simple application of sensing motion. When someone moves in its detecting range, the output of the application will be high through its SIG0 pin and the LED will turn on. Otherwise, the output will be LOW. Note that the PIR motion sensor is very sensitive to motion!

THE CIRCUIT



Parts:

- Grove-PIR Motion Sensor x1
- Grove cable x1
- Grove-Base BoosterPack x1
- LaunchPad Development Kit

Connections:

Plug the motion sensor module into Digital Grove connector J14 on the BoosterPack. This will link the SIG pin of the motion sensor to pin 39 of the LaunchPad.

CODE

This example is available here:
File > Sketchbook > LaunchPad_Grove_Kit> pir_motion_sensor

```
/*
  Grove-PIR-Motion-Sensor
 The following sketch demonstrates a simple application of sensing montion.
 When someone moves in its detecting range, it will output High through its
 SIG pin and the LED will light. Otherwise, it will output LOW. Then you can
 use it to detect the motion of people.

 The circuit:
 * sig pin of the Grove-PIR-Motion-Sensor to pin39 (J14 plug on Grove Base BoosterPack)
 * one side pin (either one) to ground
 * the other side pin to +VCC
 * LED anode (long leg) attached to RED_LED
 * LED cathode (short leg) attached to ground

 * Note: Position the trig jumper on N_Retrig to ensure

 This example code is in the public domain.

http://www.seeedstudio.com/wiki/index.php?title=Twig_-_PIR_Motion_Sensor
```

```
*/

/* Macro Define */
#define PIR_MOTION_SENSOR       39          /* sig pin of the PIR sensor */
#define LED                     RED_LED     /* led */

#define ON                      HIGH                /* led on */
#define OFF                     LOW                 /* led off */
#define _handle_led(x)          digitalWrite(LED, x)    /* handle led */

/* the setup() method runs once, when the sketch starts */
void setup() {

    pinMode(PIR_MOTION_SENSOR, INPUT);   /* declare the sig pin as an INPUT */
    pinMode(RED_LED, OUTPUT);           /* declare the red_led pin as an OUTPUT */
    _handle_led(OFF);
}

/* the loop() method runs over and over again */
void loop() {

    if(isPeopleDetected()) {
        _handle_led(ON);          /* if we detect a people, turn on the led */
    } else {
        _handle_led(OFF);         /* found nobody, turn off the light */
    }

}

/* judge if there is a people around */
boolean isPeopleDetected() {
    int sensor_val = digitalRead(PIR_MOTION_SENSOR);        /* read sig pin */
    if(HIGH == sensor_val) {
        return true;                            /* people detected */
    } else {
        return false;                           /* people un-detected */
    }
}
```

RESULT

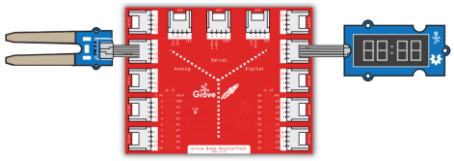
When a person comes closer to the PIR sensor, the red LED will be turned on.

TIPS

Now that you know how to use the PIR sensor to detect motion, try to pair this with other Grove modules, i.e. buzz the buzzer or trigger the relay when motion is detected.

Example - 09 : Taking Care of Your Plants (Moisture Sensor)
WHAT ARE WE DOING HERE?
The following sketch demonstrates a simple application of sensing the moisture in soil. With this, you can know whether your plant needs water or not by observing the results from the output of the sensor.

THE CIRCUIT



Parts:

- Grove-Moisture Sensor x1
- Grove-4-Digit Display x1
- Grove cable x2
- Grove-Base BoosterPack x1
- LaunchPad Development Kit

Connections:
Plug the Grove-Moisture Sensor module into Analog Grove connector J6 on the BoosterPack. This will link the SIG pin of the moisture sensor to pin 24 of the LaunchPad.

Plug the Grove-4-Digit Display into Digital Grove connector J14. This will link the 2 SIG pins (CLK & DIO) of the 4 digit module to pins 38 & 39 of the LaunchPad.

CODE
This example is available here:
File > *Sketchbook* > *LaunchPad_Grove_Kit> moisture_sensor*
Note that you may need to adjust the THRESHOLD_VALUE for your desired sensitivity.

批注 [18]: Update file path and add header files and tell them to reset the LaunchPad if display is off.

```
/*
  Moisture-Sensor
 The following sketch demonstrates a simple application of sensing
 the moisture of the soil. You can know whether a plant needs water
 or not by observing the results that the sensor outputs.

 The circuit:
 * Moisture-Sensor attached to pin 24 (J6 plug on Grove Base BoosterPack)
 * one side pin (either one) to ground
 * the other side pin to +VCC
 * LED anode (long leg) attached to RED_LED
 * LED cathode (short leg) attached to ground

 * Note:
```

```
 This example code is in the public domain.

 http://seeedstudio.com/wiki/Grove_-_Moisture_Sensor

*/
#include "TM1637.h"

/* Macro Define */
#define CLK              39                /* 4-digital display clock pin */
#define DIO              38                /* 4-digiral display data pin */
#define BLINK_LED        RED_LED           /* blink led */
#define MOISTURE_PIN     24                /* pin of moisture sensor */
#define THRESHOLD_VALUE  300               /* threshold for watering the flowers */

#define ON               HIGH              /* led on */
#define OFF              LOW               /* led off */
#define _handle_led(x)   digitalWrite(BLINK_LED, x)    /* handle led */

/* Global Varibles */
TM1637 tm1637(CLK, DIO);                   /* 4-digital display object */
int analog_value = 0;                      /* varible to store the value coming from rotary angle sensor
*/
int8_t bits[4] = {0};                      /* array to store the single bits of the value */

/* the setup() method runs once, when the sketch starts */
void setup() {

    /* Initialize 4-digital display */
    tm1637.init();
    tm1637.set(BRIGHT_TYPICAL);

    /* declare the red_led pin as an OUTPUT */
    pinMode(BLINK_LED, OUTPUT);

}

/* the loop() method runs over and over again */
void loop() {

    analog_value = analogRead(MOISTURE_PIN);      /* read the value from the sensor */

    /* if the value is smaller than threshold, turn on led */
    if(analog_value < THRESHOLD_VALUE) {
        _handle_led(ON);
    } else {
        _handle_led(OFF);
    }

    memset(bits, 0, 4);                           /* reset array when we use it */
```

```
    for(int i = 3; i >= 0; i--) {
        /* get single bits of the analog value */
        bits[i] = analog_value % 10;
        analog_value = analog_value / 10;
        tm1637.display(i, bits[i]);                /* display by 4-digital display */
    }
    delay(200);
}
```

RESULT

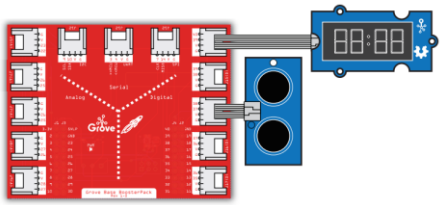
If a plant needs water, the red LED of the LaunchPad will be turned on. Note that depending on the resolution of your LaunchPad kit's ADC (analog-to-digital converter), your range may vary. For an 8-bit ADC, a number from 0-255 will show. For a 10-bit ADC, a number from 0-1023 will show. For a 12-bit ADC, a number from 0-4095 will be displayed on 4-Digit Display. Carefully dip the moisture sensor into water & watch the values change on the 4-Digit Display. Also watch the red LED turn off and on.

TIPS

Moisture-Sensor is an analog sensor, by which you can measure the water amount that your plant needs. Try to buzz a buzzer when your plant needs watering.

Example - 10 : Sensing Distance (Ultrasonic Ranger Sensor)
WHAT ARE WE DOING HERE?
This example shows how to measure the distance to obstacles using the ultrasonic sensor and display the value on a 4-Digit Display (centimeters).

THE CIRCUIT



Parts:

- Grove-Ultrasonic Ranger Sensor x1
- Grove-4-Digit Display x1
- Grove cable x2
- Grove-Base BoosterPack x1
- LaunchPad Development Kit

Connections:
Plug the Grove-Ultrasonic Sensor module into Digital Grove connector J15 on the BoosterPack. This will link the SIG pin of the ultrasonic sensor to pin 38 of the LaunchPad.

Plug the Grove-4-Digit Display into Digital Grove connector J13. This will link the 2 SIG pins (CLK & DIO) of the 4 digit module to pins 39 & 40 of the LaunchPad.

CODE
This example is available here: *File* > *Examples* > *Grove_Ultrasonic_Ranger* > *Ultrasonic4DigitDisplay*

批注 [19]: Update file path and add header files for both display and range sensor.

```
/*
  Ultrasonic-Ranger to 4-digit-display
 Measure the distance to obstacles in front and display the value on
 4-digital-display

 The circuit:
 * Ultrasonic Ranger attached to SPI plug on Grove Base BoosterPack
 * one side pin (either one) to ground
 * the other side pin to +VCC
 * LED anode (long leg) attached to RED_LED
 * LED cathode (short leg) attached to ground

 * Note:


 This example code is in the public domain.
```

```
 http://www.seeedstudio.com/wiki/Grove_-_Ultrasonic_Ranger
 */

#include "TM1637.h"
#include "Ultrasonic.h"
/* Macro Define */
#define CLK             40              /* 4-digital display clock pin */
#define DIO             39              /* 4-digital display data pin */
#define BLINK_LED       RED_LED         /* blink led */
#define ULTRASONIC_PIN  38              /* pin of the Ultrasonic Ranger */

/* Global Varibles */
TM1637 tm1637(CLK, DIO);                /* 4-digital display object */
Ultrasonic ultrasonic(ULTRASONIC_PIN);  /* Ultrasonic Ranger object */
int distance = 0;                       /* varible to store the distance to obstacles in front */
int blink_interval = 0;                 /* led delay time */
int8_t bits[4] = {0};                   /* array to store the single bits of the value */

/* the setup() method runs once, when the sketch starts */
void setup() {

    /* Initialize 4-digital display */
    tm1637.init();
    tm1637.set(BRIGHT_TYPICAL);

    /* declare the red_led pin as an OUTPUT */
    pinMode(RED_LED, OUTPUT);

}

/* the loop() method runs over and over again */
void loop() {

    distance = ultrasonic.MeasureInCentimeters();   /* read the value from the sensor */

    memset(bits, 0, 4);                         /* reset array when we use it */
    for(int i = 3; i >= 0; i--) {
        /* get single bits of the analog value */
        bits[i] = distance % 10;
        distance = distance / 10;
        tm1637.display(i, bits[i]);             /* display by 4-digital display */
    }
    delay(100);
}
```

RESULT

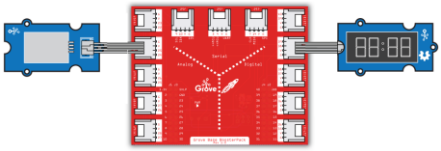
批注 [20]: update pin number in code to 40 and 39
update ultrasonic pin to 38

The 4-Digit Display will show you the distance between the ultrasonic ranger and the object in front of it (in centimeters).

TIPS
Try to create a musical instrument by pairing this with the Grove-buzzer module & change the tone based on distance.

Example - 11 : Feeling the Environment (Temperature and Humidity)
WHAT ARE WE DOING HERE?
This example allows you to read the value from the Grove-Temperature-Humidity sensor. The temperature & humidity values are displayed on the Grove 4-Digit Display.

THE CIRCUIT



Parts:

- Grove-Temp & Humidity Sensor x1
- Grove-4-Digit Display x1
- Grove cable x2
- Grove-Base BoosterPack x1
- LaunchPad Development Kit

Connections:
Plug the Grove-Temp & Humidity Sensor module into Analog Grove connector J6 on the BoosterPack. This will link the SIG pin of the ultrasonic sensor to pin 24 of the LaunchPad.

Plug the Grove-4-Digit Display into Digital Grove connector J14. This will link the 2 SIG pins (CLK & DIO) of the 4 digit module to pins 38 & 39 of the LaunchPad.

CODE

This example is available here: *File* > *Example* > *LaunchPad_Kit> temp_humi_sensor*

批注 [21]: update file path and add header files to sketch folder

```
/*
  Grove-Temperature-Humidity Sensor
 Read value from grove-temperature-humidity sensor then display on grove-4-digital-display

 The circuit:
 * 4-digital-display attached to pin38 & pin39 (J14 plug on Grove Base BoosterPack)
 * sig pin of the Grove-Temperature-Humidity Sensor to the analog pin 24 (Grove connector
J6)
 * one side pin (either one) to ground
 * the other side pin to +VCC
 * LED anode (long leg) attached to RED_LED
 * LED cathode (short leg) attached to ground

 * Note: Put your hands on Grove-Temperature-Humidity Sensor, both of the value
       will rise.

       4-digital-display:
       |--------------------------|
       | temperature :  humidity  |
       |--------------------------|

 This example code is in the public domain.

 http://www.seeedstudio.com/wiki/Grove_-_Temperature_and_Humidity_Sensor_Pro
```

```
 */

#include "TM1637.h"
#include "DHT.h"

/* Macro Define */
#define CLK              39                /* 4-digital display clock pin */
#define DIO              38                /* 4-digital display data pin */
#define BLINK_LED        RED_LED           /* blink led */
#define TEMP_HUMI_PIN    24                /* pin of temperature&humidity sensor */

/* Global Varibles */
TM1637 tm1637(CLK, DIO);                   /* 4-digital display object */
DHT dht(TEMP_HUMI_PIN, DHT22);             /* temperature&humidity sensor object */

int8_t t_bits[2] = {0};                    /* array to store the single bits of the temperature */
int8_t h_bits[2] = {0};                    /* array to store the single bits of the humidity */

/* the setup() method runs once, when the sketch starts */
void setup() {

   tm1637.init();                          /* initialize 4-digital display */
   tm1637.set(BRIGHT_TYPICAL);             /* set the brightness */
   tm1637.point(POINT_ON);                 /* light the clock point ":" */

   dht.begin();                            /* initialize temperature humidity sensor */

   pinMode(RED_LED, OUTPUT);               /* declare the red_led pin as an OUTPUT */

}

/* the loop() method runs over and over again */
void loop() {

   int _temperature = dht.readTemperature();          /* read the temperature value from the
sensor */
   int _humidity = dht.readHumidity();                /* read the humidity value from the sensor */

   memset(t_bits, 0, 2);                              /* reset array when we use it */
   memset(h_bits, 0, 2);

   /* 4-digital-display [0,1] is used to display temperature */
   t_bits[0] = _temperature % 10;
   _temperature /= 10;
   t_bits[1] = _temperature % 10;

   /* 4-digital-display [2,3] is used to display humidity */
   h_bits[0] = _humidity % 10;
   _humidity /= 10;
```

```
    h_bits[1] = _humidity % 10;

    /* show it */
    tm1637.display(1, t_bits[0]);
    tm1637.display(0, t_bits[1]);

    tm1637.display(3, h_bits[0]);
    tm1637.display(2, h_bits[1]);

}
```

RESULT

The first two digits display the temperature (celsius) and the next two digits display humidity (%).

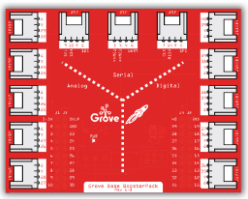
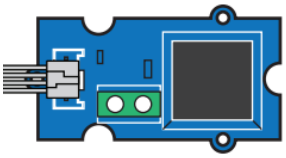
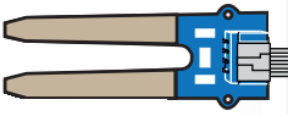
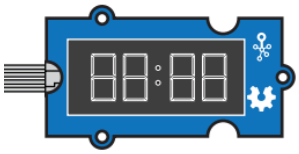
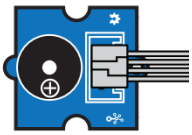
TIPS

When paired with the Grove-Moisture module, you can create a greenhouse monitoring system.

LET'S USE OUR NEW KNOWLEDGE TO BUILD A FEW PROJECTS
Do you think you can build the projects below on your own?

PROJECT ONE: Intelligent Tulip Water Reminder Device
WHAT ARE WE DOING HERE?
Decorate your room with tulips, and make your home a pleasant space. But what if we are too busy to take care of the lovely tulips? Perhaps we can build a device that will remind us when they are thirsty! We can enable our plants to reach out for help using the Moisture Sensor, which can detect the electric resistance of the soil, and thus show us the soil moisture using the 4-Digit Display module.

THE RECOMMENDED HARDWARE:

 **The Grove Base BoosterPack**	 **The Grove-Relay Module**	 **The Grove-Moisture Sensor Module**
 **The Grove-4-digit-display Module**	 **The Grove-Buzzer Module**	

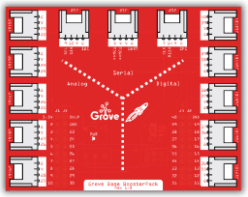
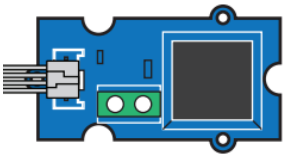
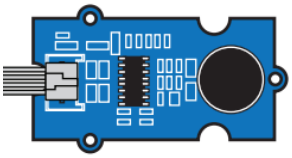
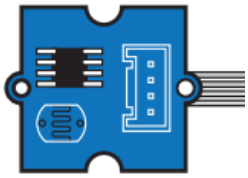
DESIRED RESULTS:
When the moisture is below the value you have set, the buzzer will respond with a beeping noise, reminding you that your plants are in need of water. At this moment, we can also use the Relay to open a valve to release water onto your tulips.

PROJECT TWO: Intelligent Lamp

WHAT ARE WE DOING HERE?

How about we create a project for home automation? For example, when night falls or if motion is detected, perhaps we can turn on a lamp in your room automatically. When no one is around or the day dawns, it can also automatically turn off. Now, let's try to work our magic with the help of a few Grove modules!

THE RECOMMENDED HARDWARE:

 **The Grove Base BoosterPack**	 **The Grove-Relay Module**	 **The Grove-PIR Motion Sensor Module**
 **The Grove-Ambient Light Sensor**		

DESIRED RESULTS:

When someone gets closer, the motion sensor will detect presence & we can turn on the lamp by closing the signal on the relay.

CREATE CAPTURE USING CC3100 WI-FI BOOSTERPACK

INTRODUCE TEMBOO?

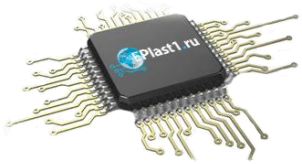
Компания «ЭлектроПласт» предлагает заключение долгосрочных отношений при поставках импортных электронных компонентов на взаимовыгодных условиях!

Наши преимущества:

- Оперативные поставки широкого спектра электронных компонентов отечественного и импортного производства напрямую от производителей и с крупнейших мировых складов;
- Поставка более 17-ти миллионов наименований электронных компонентов;
- Поставка сложных, дефицитных, либо снятых с производства позиций;
- Оперативные сроки поставки под заказ (от 5 рабочих дней);
- Экспресс доставка в любую точку России;
- Техническая поддержка проекта, помощь в подборе аналогов, поставка прототипов;
- Система менеджмента качества сертифицирована по Международному стандарту ISO 9001;
- Лицензия ФСБ на осуществление работ с использованием сведений, составляющих государственную тайну;
- Поставка специализированных компонентов (Xilinx, Altera, Analog Devices, Intersil, Interpoint, Microsemi, Aeroflex, Peregrine, Syfer, Eurofarad, Texas Instrument, Miteq, Cobham, E2V, MA-COM, Hittite, Mini-Circuits,General Dynamics и др.);

Помимо этого, одним из направлений компании «ЭлектроПласт» является направление «Источники питания». Мы предлагаем Вам помощь Конструкторского отдела:

- Подбор оптимального решения, техническое обоснование при выборе компонента;
- Подбор аналогов;
- Консультации по применению компонента;
- Поставка образцов и прототипов;
- Техническая поддержка проекта;
- Защита от снятия компонента с производства.



Как с нами связаться

Телефон: 8 (812) 309 58 32 (многоканальный)
Факс: 8 (812) 320-02-42
Электронная почта: org@eplast1.ru
Адрес: 198099, г. Санкт-Петербург, ул. Калинина, дом 2, корпус 4, литера А.