

Hong Kong Product Design Makeathon

2016-2017

全港中小學產品設計大賽2016-2017

第五場工作坊
Programming skills
by using littleBits
Arduino

Hong Kong Product Design Makeathon 2016-2017

全港中小學產品設計大賽2016-2017

簡介：
本大賽為了提升學生對產品設計的興趣，鼓勵創意設計，展示學生在科學、科技、工程、數學等方面學習成果。學生由概念圖開始到原型整合製作都全程負責，透過比賽學生有機會把想法帶到現實中實現，啟發學生的創意及設計的能力。

比賽目的：

- 利用STEM知識應用於解決日常生活中遇到的難題
- 提升同學對產品設計的興趣

參賽資格：

- 全港中小學校學生
- 比賽將分為小學組及中學組

比賽題目：

- 無障生活、環保及節約能源、大數據

比賽日程：

- 2016年10月至2017年4月

活動簡介會：

- 2016年10月22日下午2時至5時
- 地點：香港理工大學TU101及TU103室

評審安排及獎品：

- 詳情請參閱官方網站

官方網頁：<http://www.ise.polyu.edu.hk/html/product-design-makeathon-2016/>

報名查詢熱線：31608443

主辦機構 THE HONG KONG POLYTECHNIC UNIVERSITY
DEPARTMENT OF INDUSTRIAL AND SYSTEMS ENGINEERING
工業及系統工程系

協辦機構 DTSL

贊助機構 littleBits™

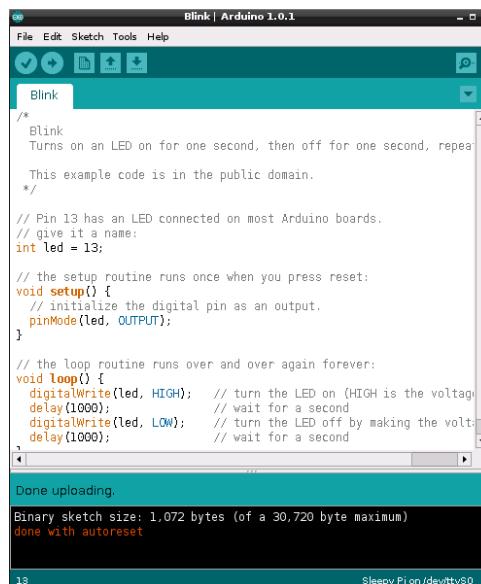
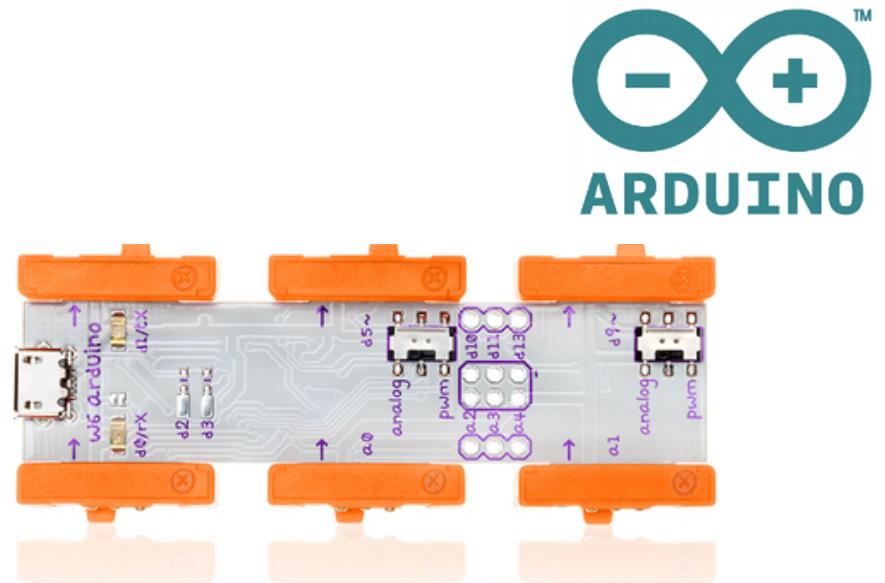
支持機構（排名不分先後）

ITLE 資訊科技教育領袖協會
HKA 香港電腦教育學會
The Hong Kong Association for Computer Education

今日校園
<http://www.educationtoday.hk>

What is Arduino?

- Open-source prototyping platform
- Building digital devices and interactive objects that sense, control physical devices
- Hardware: Arduino I/O board (microcontroller)
- Software: Arduino IDE (Integrated Development Environment)
- Able to read inputs – sensors, button etc. and turn it into outputs - activating a motor, turning on an LED
- Can run independently or communicate with computer



A screenshot of the Arduino IDE interface. The title bar says "Blink | Arduino 1.0.1". The menu bar includes File, Edit, Sketch, Tools, and Help. The toolbar has icons for Open, Save, Run, and Stop. The code editor window contains the "Blink" sketch. The code is as follows:

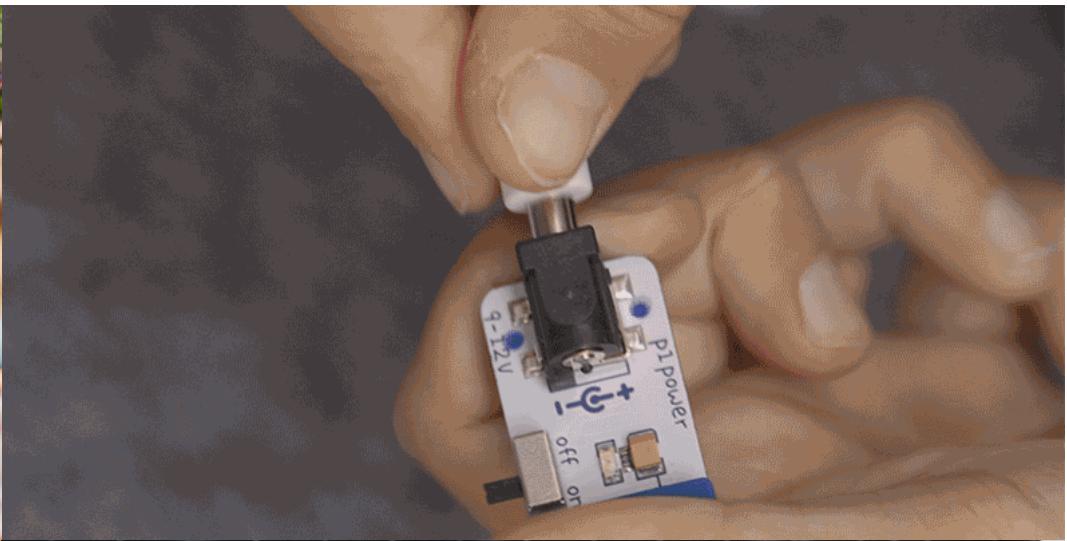
```
/*
  Blink
  Turns on an LED on for one second, then off for one second, repeating.
  This example code is in the public domain.
*/

// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

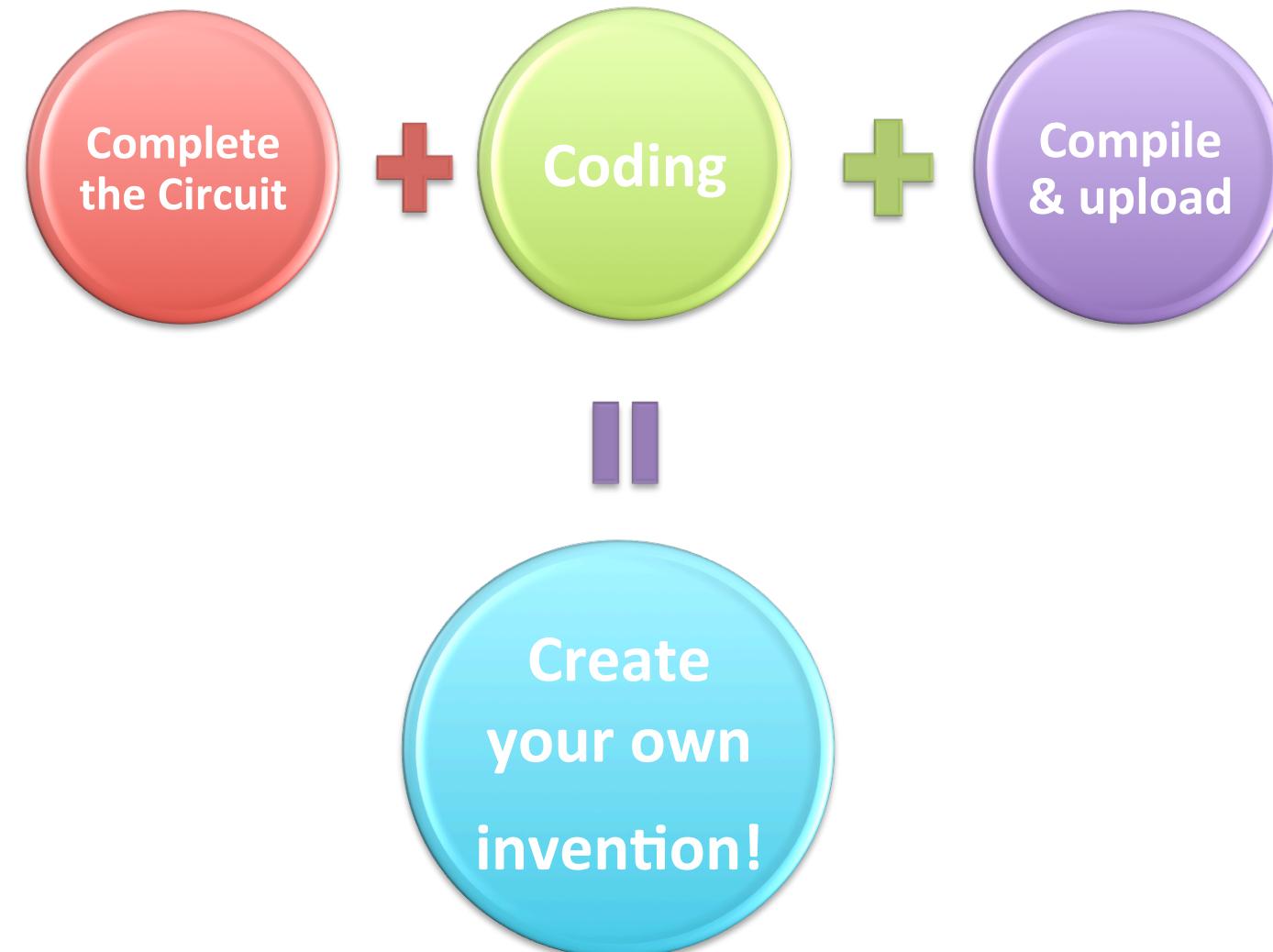
// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output:
  pinMode(led, OUTPUT);
}

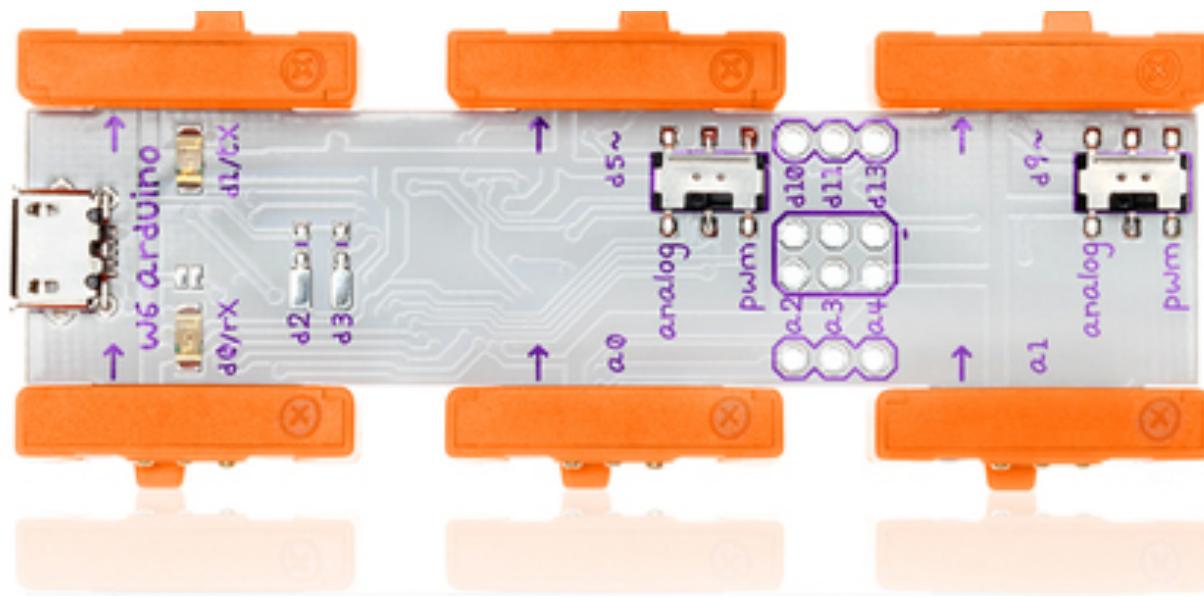
// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH);    // turn the LED on (HIGH is the voltage level)
  delay(1000);               // wait for a second
  digitalWrite(led, LOW);     // turn the LED off by making the voltage level
  delay(1000);             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

The status bar at the bottom shows "Done uploading." and "Binary sketch size: 1,072 bytes (of a 30,720 byte maximum) done with autoreset". The footer indicates "13 Sleepy Pi on /dev/ttyS0".



How it works?

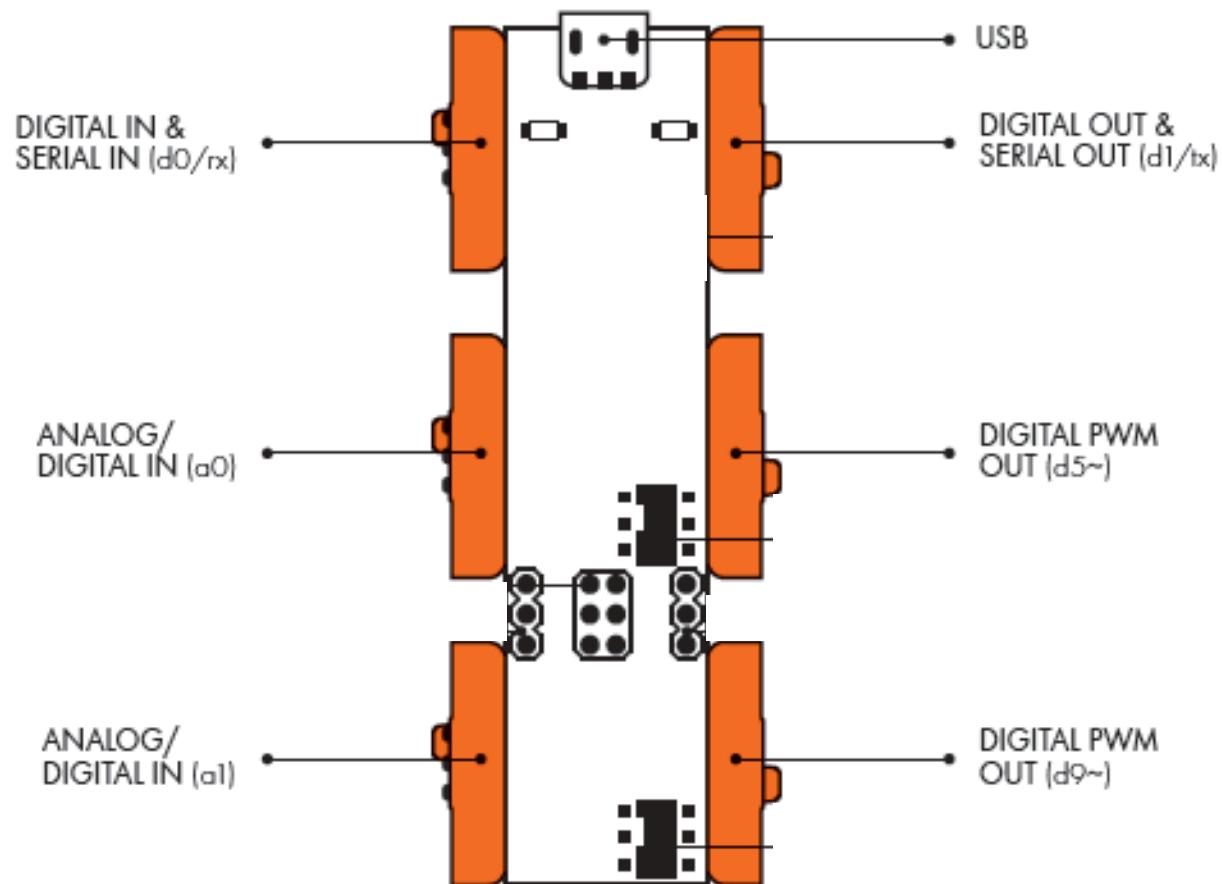




Arduino Leonardo

Structure of the board

3
Input
pins



3
Output
pins

Set up IDE

<https://www.arduino.cc/en/Main/Software>

The screenshot shows the Arduino website's main navigation bar at the top, featuring links for Home, Buy, Download, Products, Learning, Forum, Support, and Blog. Below the navigation is a search bar and user account links for Log In and Sign Up. A language selection dropdown is set to English. The main content area is titled "Download the Arduino Software". On the left, there's a large circular icon with the Arduino logo (minus and plus signs) and a brief description of the Arduino IDE: "The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software. This software can be used with any Arduino board. Refer to the Getting Started page for installation instructions." To the right, there are download links for Windows (Installer and ZIP), Mac OS X (10.7 Lion or newer), and Linux (32 bits, 64 bits, ARM experimental). Release notes, source code, and checksums links are also provided.

ARDUINO 1.6.10

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software.

This software can be used with any Arduino board. Refer to the [Getting Started](#) page for installation instructions.

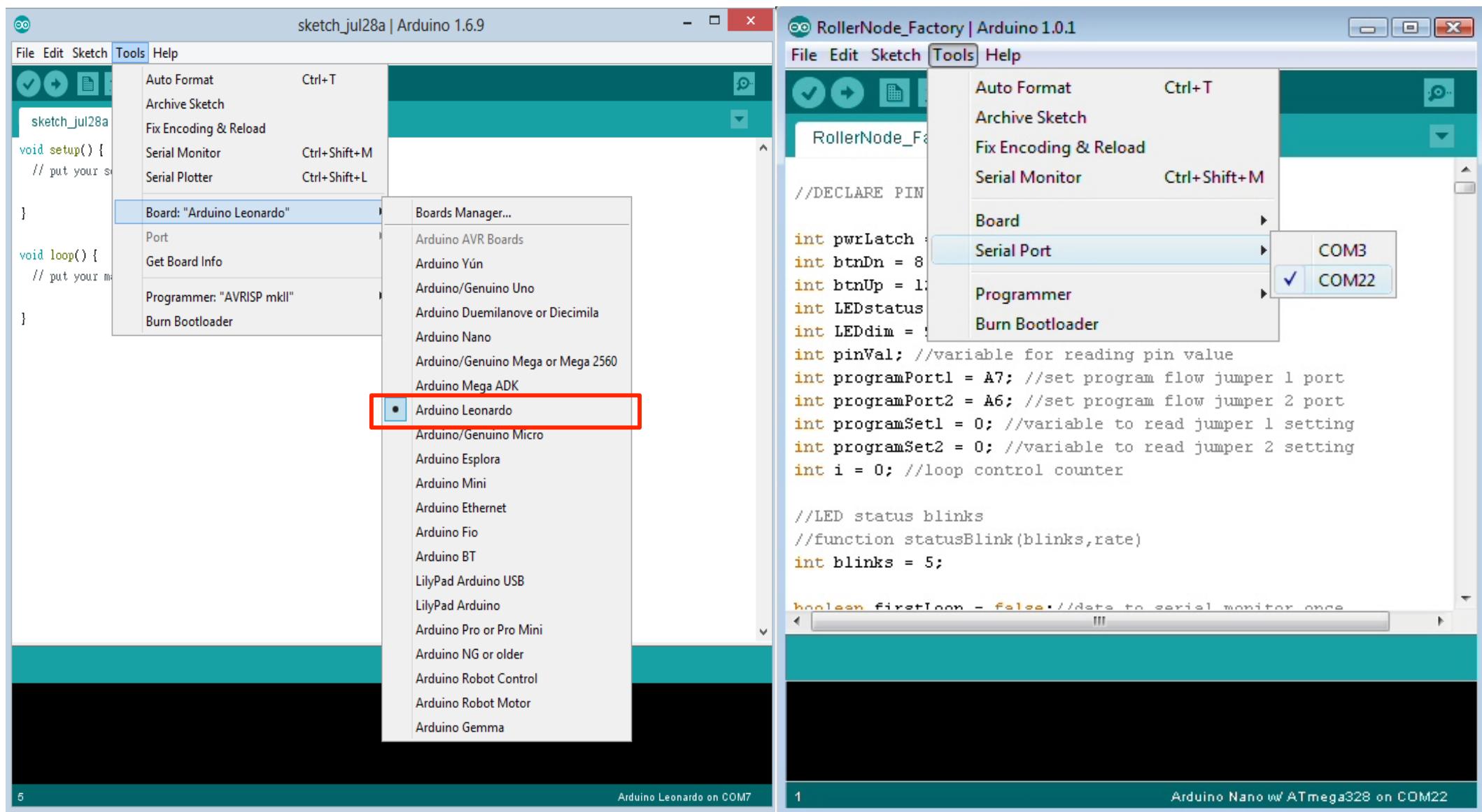
Windows Installer
Windows ZIP file for non admin install

Mac OS X 10.7 Lion or newer

Linux 32 bits
Linux 64 bits
Linux ARM (experimental)

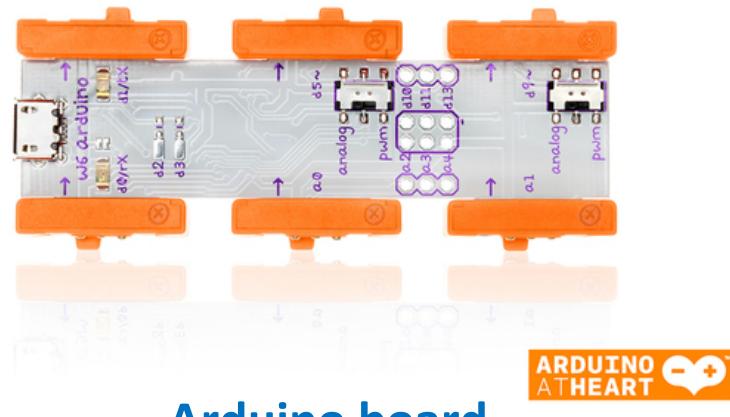
[Release Notes](#)
[Source Code](#)
[Checksums](#)

Choose the correct Arduino board setting & Serial Port



Let's get started !

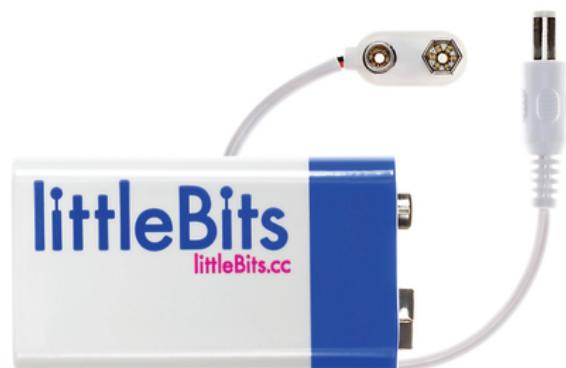
Example 1 (LED Blinking)



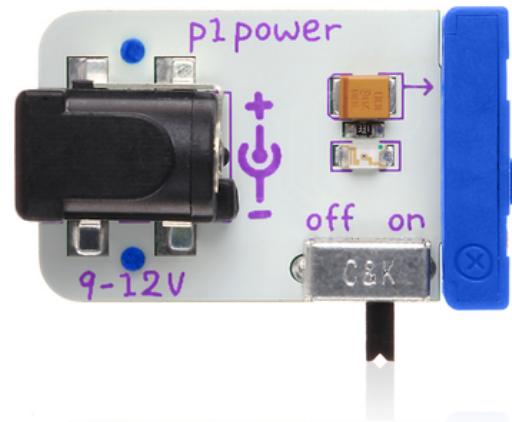
Arduino board



Bargraph



LittleBits Battery

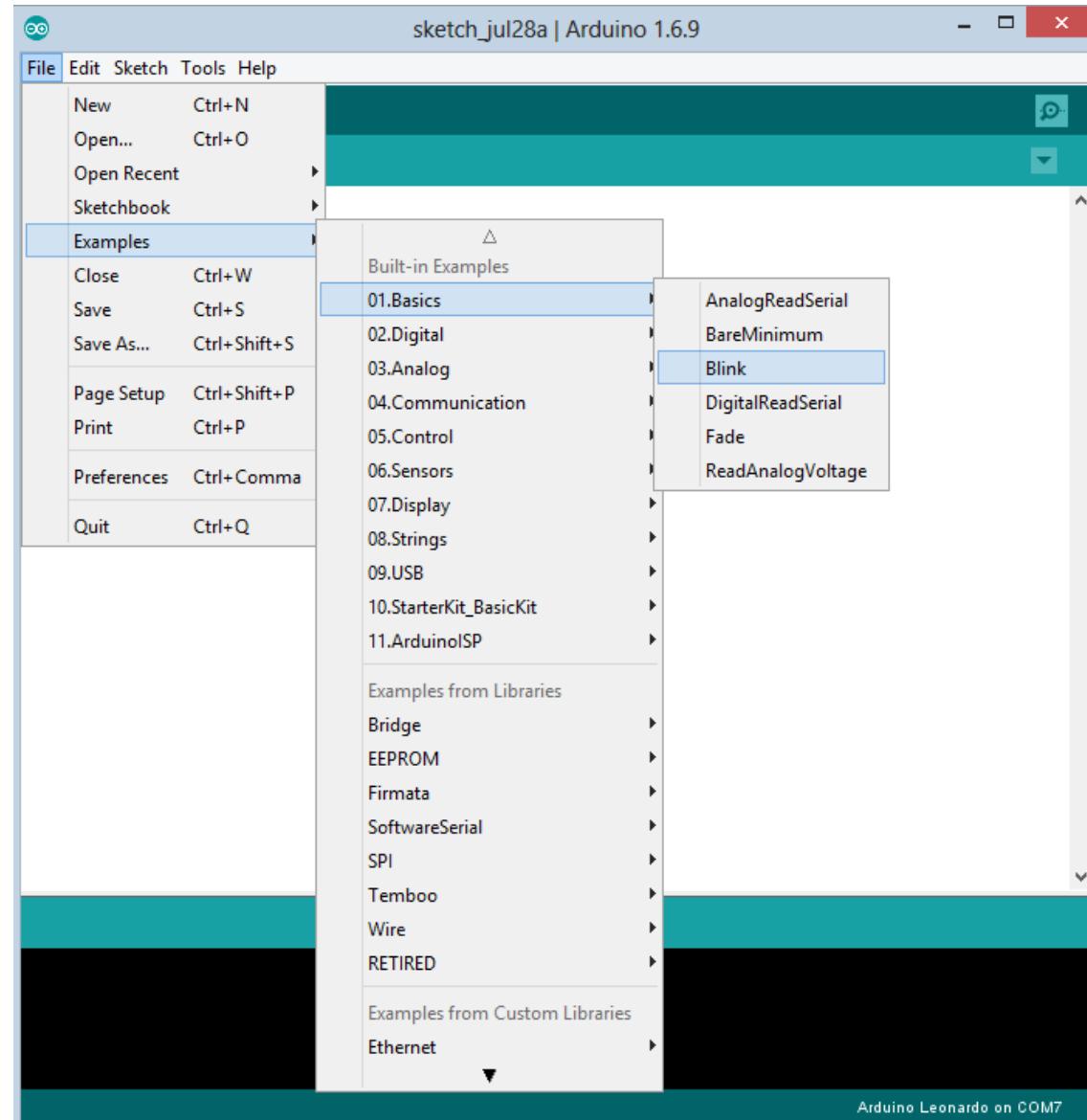


Power



Power & Battery at a0 pin
Bargraph at d5 pin

[File]
[Examples]
[01.Basics]
[Blink]

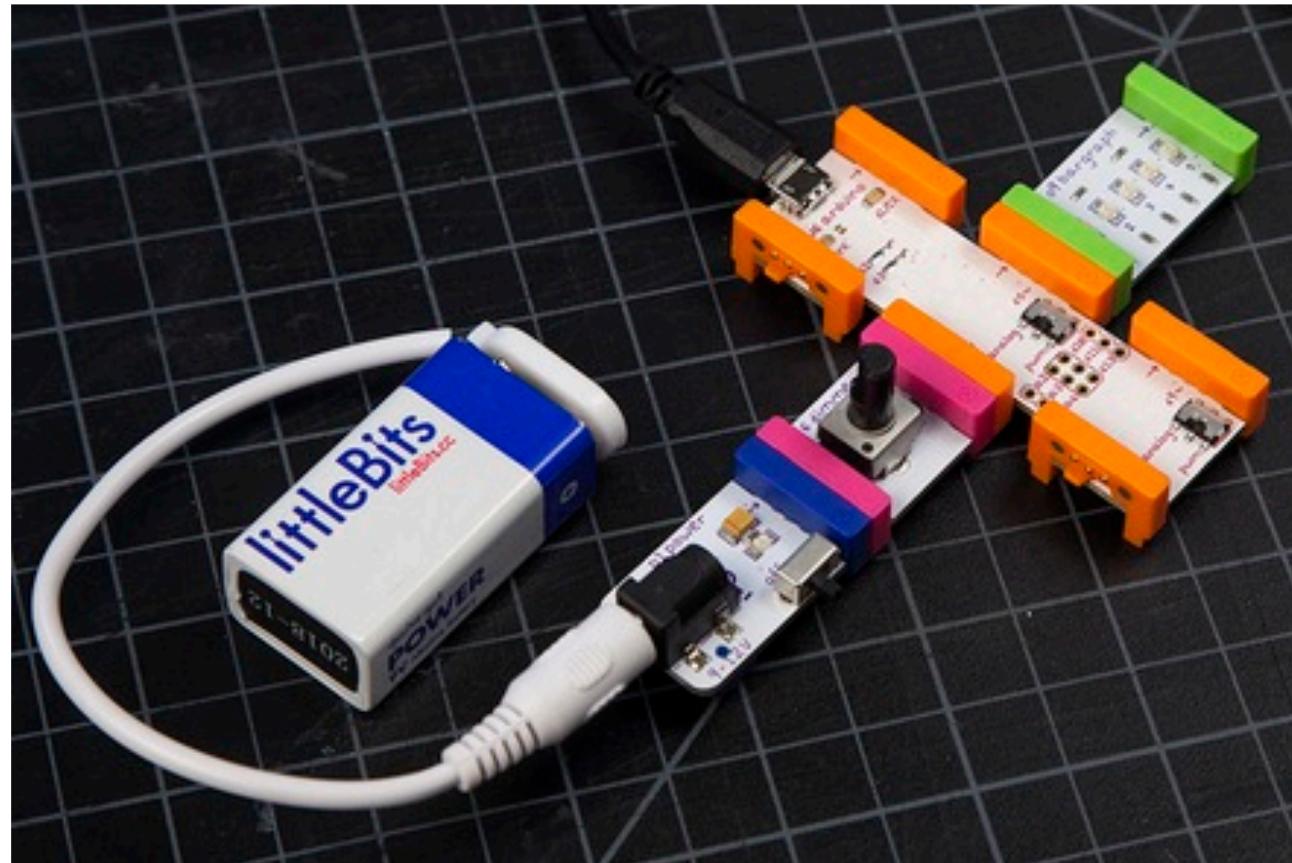


Understanding the code

```
// the setup function runs once when you press reset or power the board
void setup() {
    // initialize digital pin 5 as an output. ← LED at D5 pin
    → pinMode(5, OUTPUT);
}

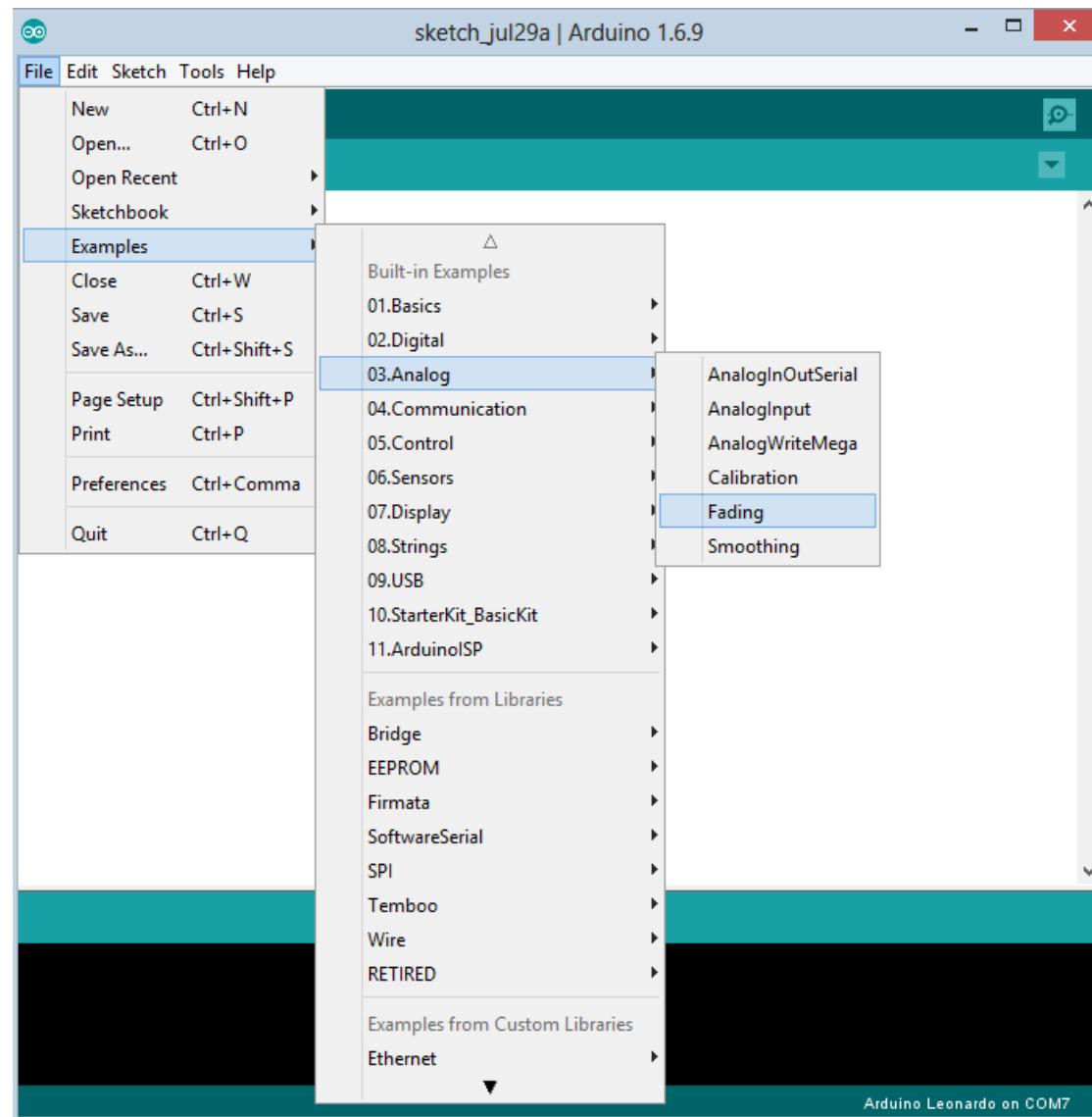
// the loop function runs over and over again forever
void loop() {
    → digitalWrite(5, HIGH);      // turn the LED on (HIGH is the voltage level)
    delay(1000);                // wait for a second
    → digitalWrite(5, LOW);       // turn the LED off by making the voltage LOW
    delay(1000);                // wait for a second
}
```

Example 2 (Fading LED)



Power & Battery & Slide Dimmer at a0 pin
Bargraph at d5 pin

[File]
[Examples]
[03.Analog]
[Fading]



Understanding the code

```
Int ledPin = 5; // Declare that LED is at Pin D5
void setup(){
}
    // nothing happens in setup
void loop(){
    // This will act as the inhale for the PWM value starting from zero to a max value of 255
    → for(int fadeValue = 0; fadeValue <= 255; fadeValue ++){
        analogWrite(ledPin, fadeValue);
    → delay( (1034 - analogRead(A0)) / 10 ); // wait for a moment before moving forward
    }
    // This will act as the exhale for the PWM value starting from 255 to a min value of 0
    → for(int fadeValue = 255; fadeValue >= 0; fadeValue --){
        analogWrite(ledPin, fadeValue);
    → delay( (1034 - analogRead(A0)) / 10 ); // wait for a moment before moving forward
    }
}
```

Using Scratch Coding for littleBits Arduino

Download Scratch 2 Offline version

- Go to <https://scratch.mit.edu/scratch2download/> to download Scratch 2.0 offline version

The screenshot shows the Scratch website's download page for the Offline Editor. At the top, there's a blue header bar with the Scratch logo, navigation links (Create, Explore, Discuss, About, Help), a search bar, and user options (Join Scratch, Sign in). Below the header, the main content area has a title "Scratch 2 Offline Editor". It explains that you can install the Scratch 2.0 editor to work on projects without an internet connection, supporting Mac, Windows, and some Linux versions (32 bit). A note for Mac users states that the latest version requires Adobe Air 20, with a link to upgrade manually. The page is divided into three numbered steps: 1. Adobe AIR (with a download link for Mac OS X and Windows), 2. Scratch Offline Editor (with download links for Mac OS X, Mac OS 10.5 & Older, Windows, and Linux), and 3. Support Materials (with links to Starter Projects, Getting Started Guide, and Scratch Cards).

SCRATCH Create Explore Discuss About Help Search Join Scratch Sign in

Scratch 2 Offline Editor

You can install the Scratch 2.0 editor to work on projects without an internet connection. This version will work on Mac, Windows, and some versions of Linux (32 bit).

Note for Mac Users: the latest version of Scratch 2.0 Offline requires Adobe Air 20. To upgrade to Adobe Air 20 manually, go [here](#).

Adobe AIR	Scratch Offline Editor	Support Materials
1 If you don't already have it, download and install the latest Adobe AIR Mac OS X - Download Mac OS 10.5 & Older - Download Windows - Download Linux - Download	2 Next download and install the Scratch 2.0 Offline Editor Mac OS X - Download Mac OS 10.5 & Older - Download Windows - Download Linux - Download	3 Need some help getting started? Here are some helpful resources. Starter Projects - Download Getting Started Guide - Download Scratch Cards - Download

The screenshot shows a Windows file explorer window with a context menu open over a ZIP archive named 'scratch2LittleBits-master'. The 'Extract Here' option is highlighted. A large blue curved arrow points from the bottom right towards the 'littleBits_Scratch2' folder in the main file list.

Context Menu Options:

- Open
- Open with WinRAR
- Extract files...
- Extract Here**
- Extract to scratch2LittleBits-master\
- Edit with Notepad++
- Open with
- Share with

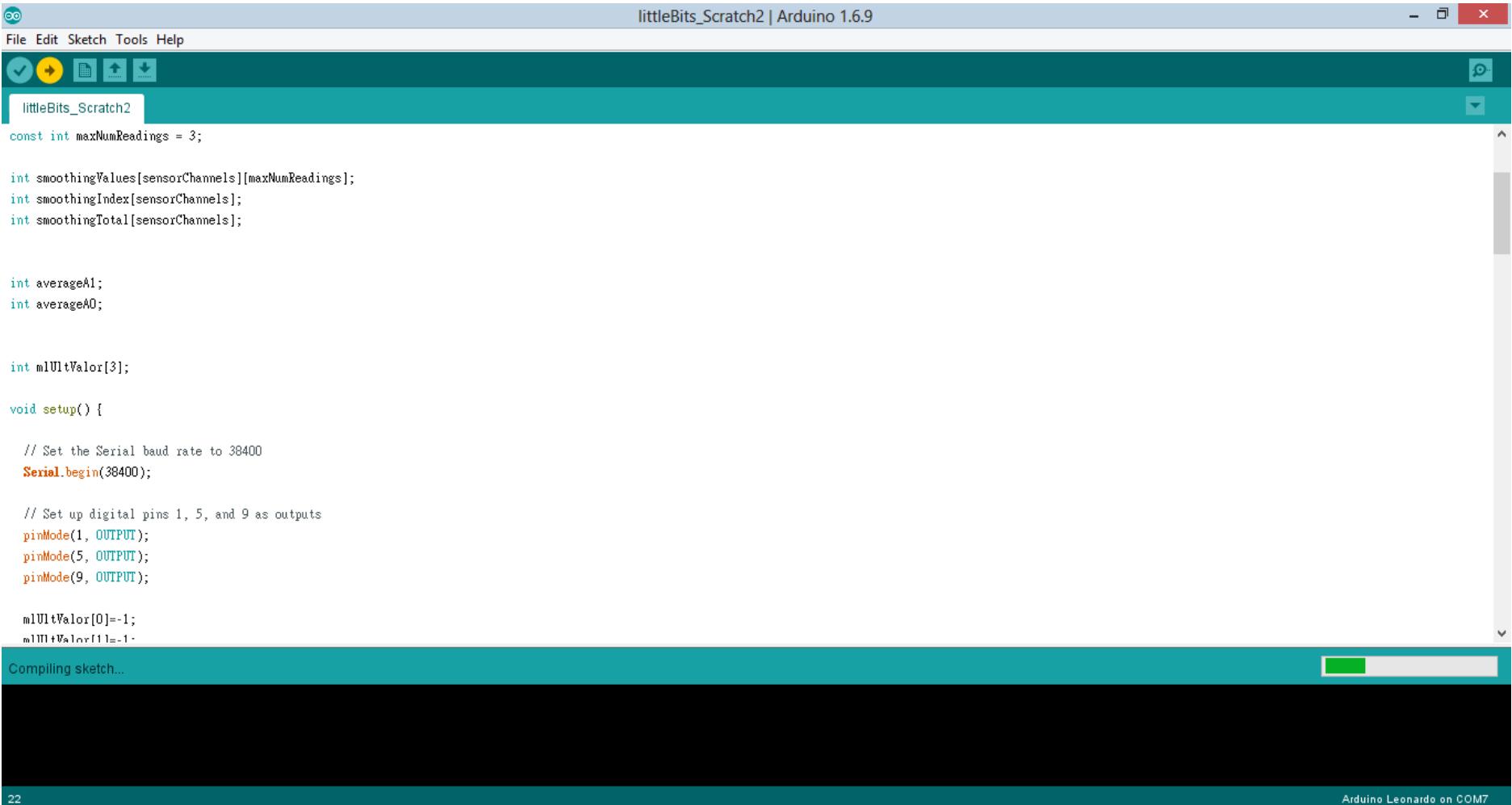
File Explorer Main View:

- Downloads
- scratch2LittleBits-master
- littleBits_Scratch2
- manualconfiguracionarduino

Name	Date modified	Type	Size
scratch2LittleBits-master	30/7/2016 5:32 PM	WinRAR ZIP archive	5,238
GoogleAppEngine-1.9.40		installer ...	56,150
python-2.7.12		installer ...	18,464
ubuntu-16.04.1-desktop-a		File	1,477,840
VirtualBox-5.1.2-108956-W		VirtualBox D...	118,410
Packt_Developing-RESTful		obat D...	3,983
heroku-toolbelt		n	48,343
AI2_Arduino		archive	4,001
Screen Shot 2016-07-19 at		e	80
4.50 PM.png		212,281	

Upload the Arduino program to the Arduino board

→ Check the port & board on “Tools”



The screenshot shows the Arduino IDE interface with the following details:

- Title Bar:** littleBits_Scratch2 | Arduino 1.6.9
- File Menu:** File Edit Sketch Tools Help
- Sketch Name:** littleBits_Scratch2
- Code Area:** The code is written in C++ and defines variables for sensor readings, smoothing values, and average values, along with setup and loop functions for serial communication and pin modes.

```
const int maxNumReadings = 3;

int smoothingValues[sensorChannels][maxNumReadings];
int smoothingIndex[sensorChannels];
int smoothingTotal[sensorChannels];

int averageA1;
int averageA0;

int m1UltValor[3];

void setup() {
    // Set the Serial baud rate to 38400
    Serial.begin(38400);

    // Set up digital pins 1, 5, and 9 as outputs
    pinMode(1, OUTPUT);
    pinMode(5, OUTPUT);
    pinMode(9, OUTPUT);

    m1UltValor[0]=-1;
    m1UltValor[1]=-1;
}

void loop() {
    // Read sensor values
    for (int i = 0; i < maxNumReadings; i++) {
        for (int j = 0; j < sensorChannels; j++) {
            smoothingValues[j][i] = sensorValues[j];
        }
    }

    // Calculate averages
    for (int i = 0; i < sensorChannels; i++) {
        averageA1 += smoothingValues[i][smoothingIndex[i]];
        averageA0 += smoothingValues[i][maxNumReadings - 1];
    }

    // Smooth total
    for (int i = 0; i < sensorChannels; i++) {
        smoothingTotal[i] = (averageA1 * smoothingWeights[i]) + (averageA0 * (1 - smoothingWeights[i]));
    }

    // Set output levels
    for (int i = 0; i < sensorChannels; i++) {
        if (smoothingTotal[i] > 0) {
            digitalWrite(i, HIGH);
        } else {
            digitalWrite(i, LOW);
        }
    }

    // Print results to Serial
    Serial.print("A1: ");
    Serial.print(averageA1);
    Serial.print(" A0: ");
    Serial.print(averageA0);
    Serial.print(" Total: ");
    Serial.println(smoothingTotal[0]);
}
```

- Status Bar:** Compiling sketch... (progress bar), Arduino Leonardo on COM7, page 22

Download Java runtime

https://java.com/zh_TW/download/



所有 Java 下載
如果您想為其他電腦或作業系統下載 Java，請按一下以下連結。
[所有 Java 下載](#)

報告問題
為何我在瀏覽含 Java 應用程式的網頁時，總是會被重導至此頁面？
[深入瞭解](#)

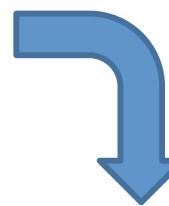
免費 Java 下載
立即下載桌上型電腦專用的 Java！

Version 8 Update 101

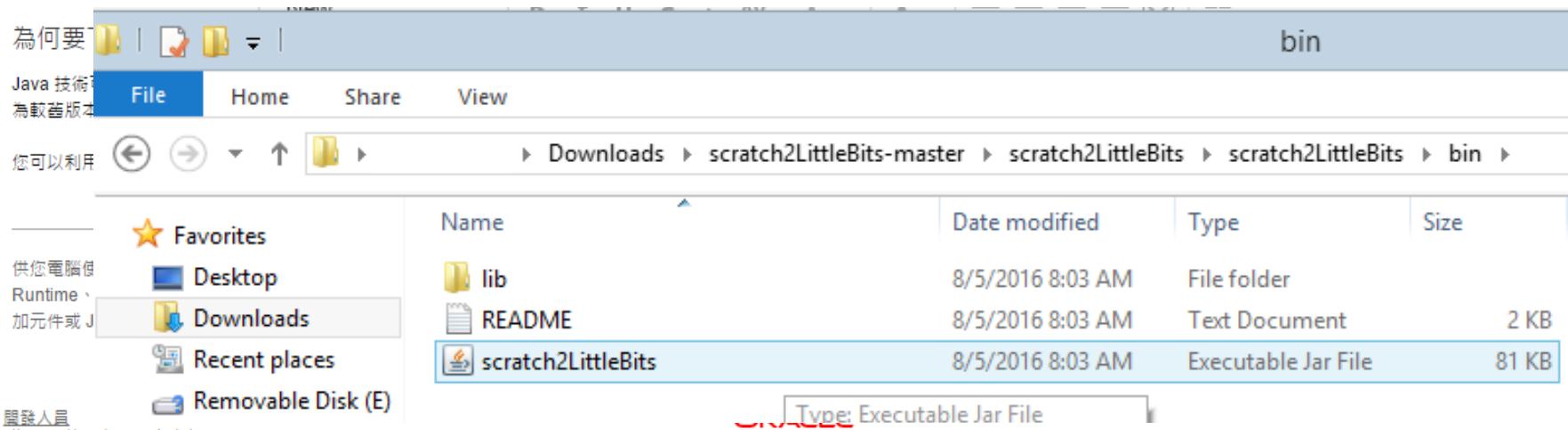
發行日期：2016 年 7 月 19 日

免費 Java 下載

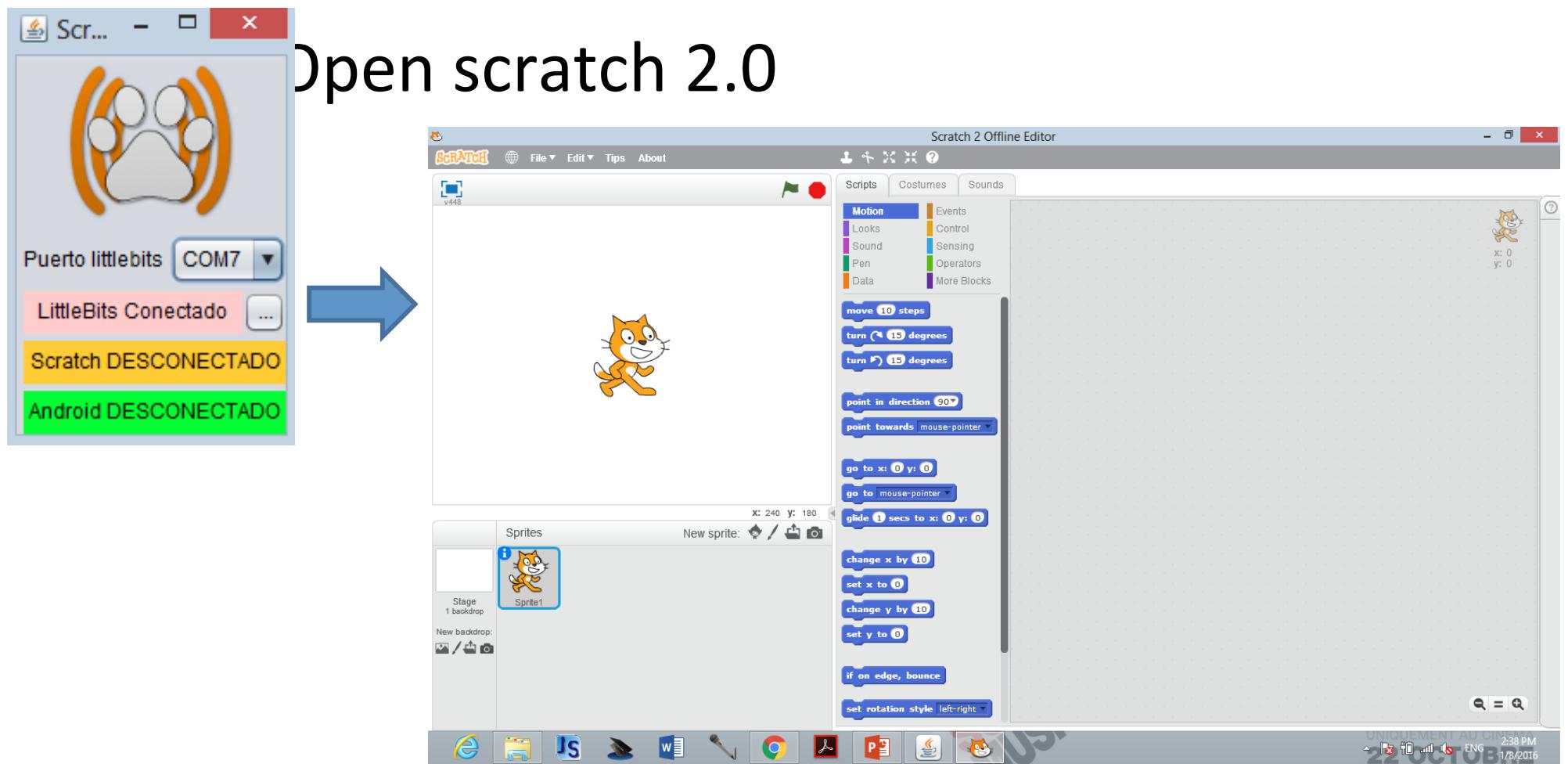
» 什麼是 Java？ » 我有 Java 嗎？ » 需要說明嗎？



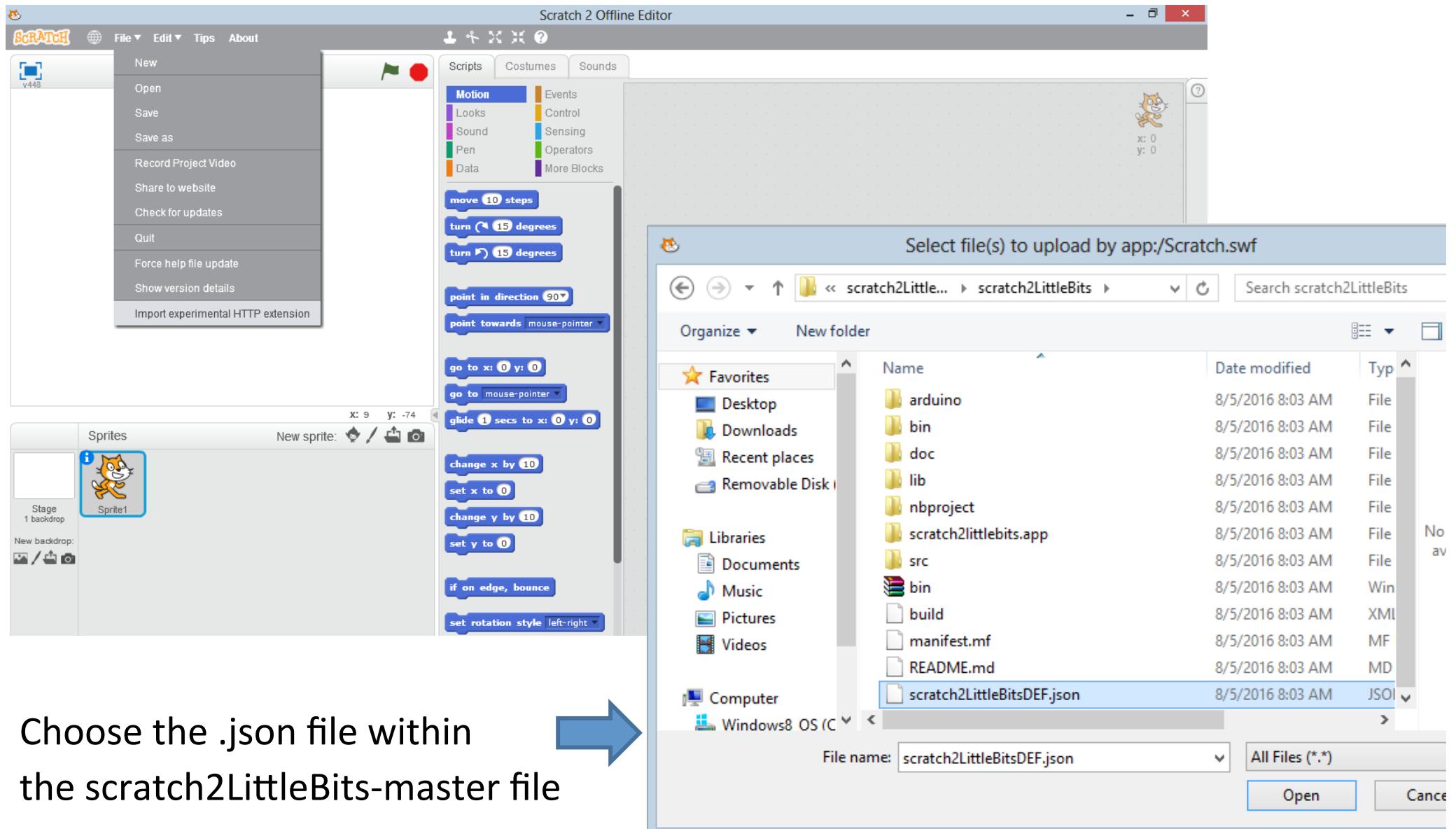
Open
scratch2LittleBits
file



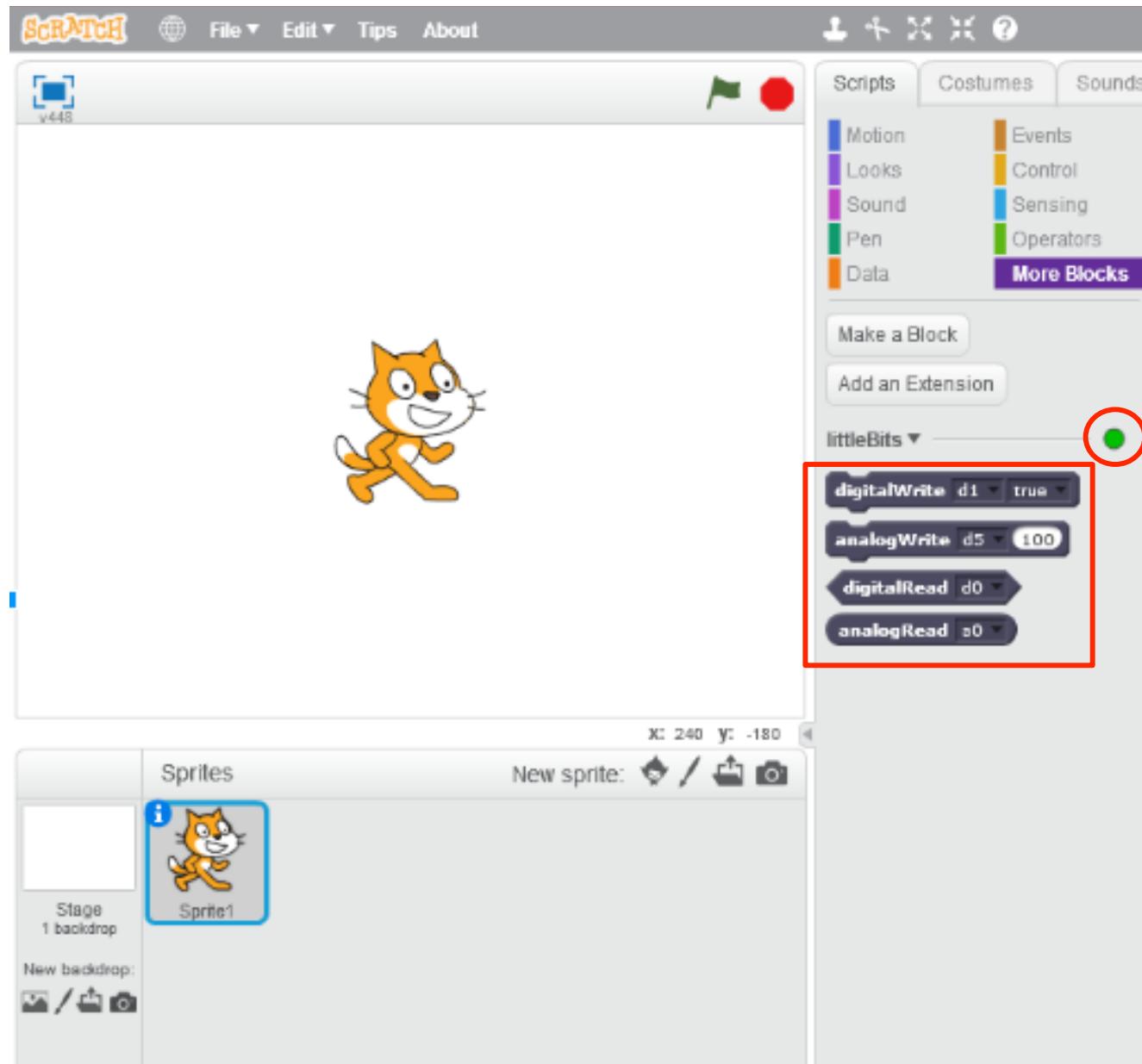
Select the right port to connect the Arduino with scratch



 Press “shift” & click “File” to choose the last option



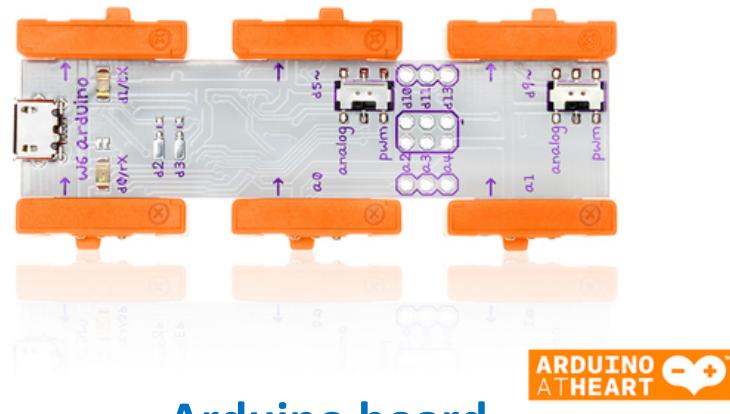
Choose the .json file within
the scratch2LittleBits-master file



Green light
indicate the
connection being
successful

We can now use
the littleBits
blocks for our
functions

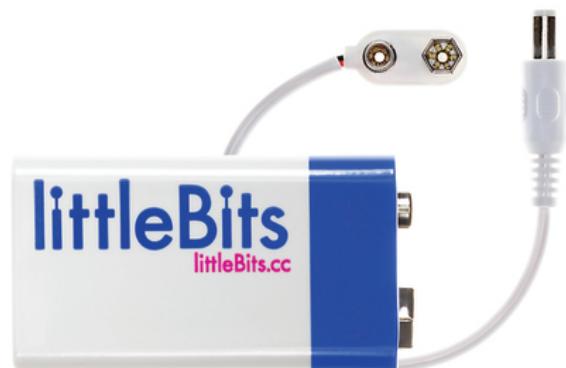
Example 3 (LED Blinking)



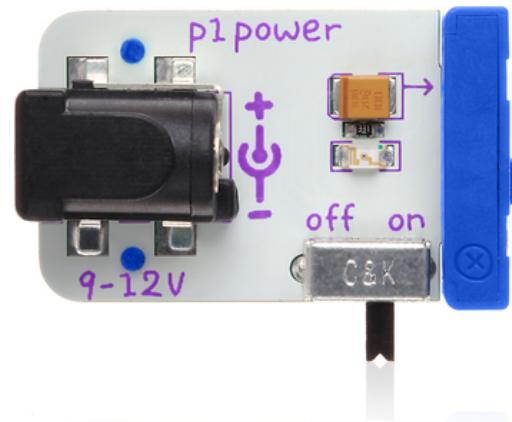
Arduino board



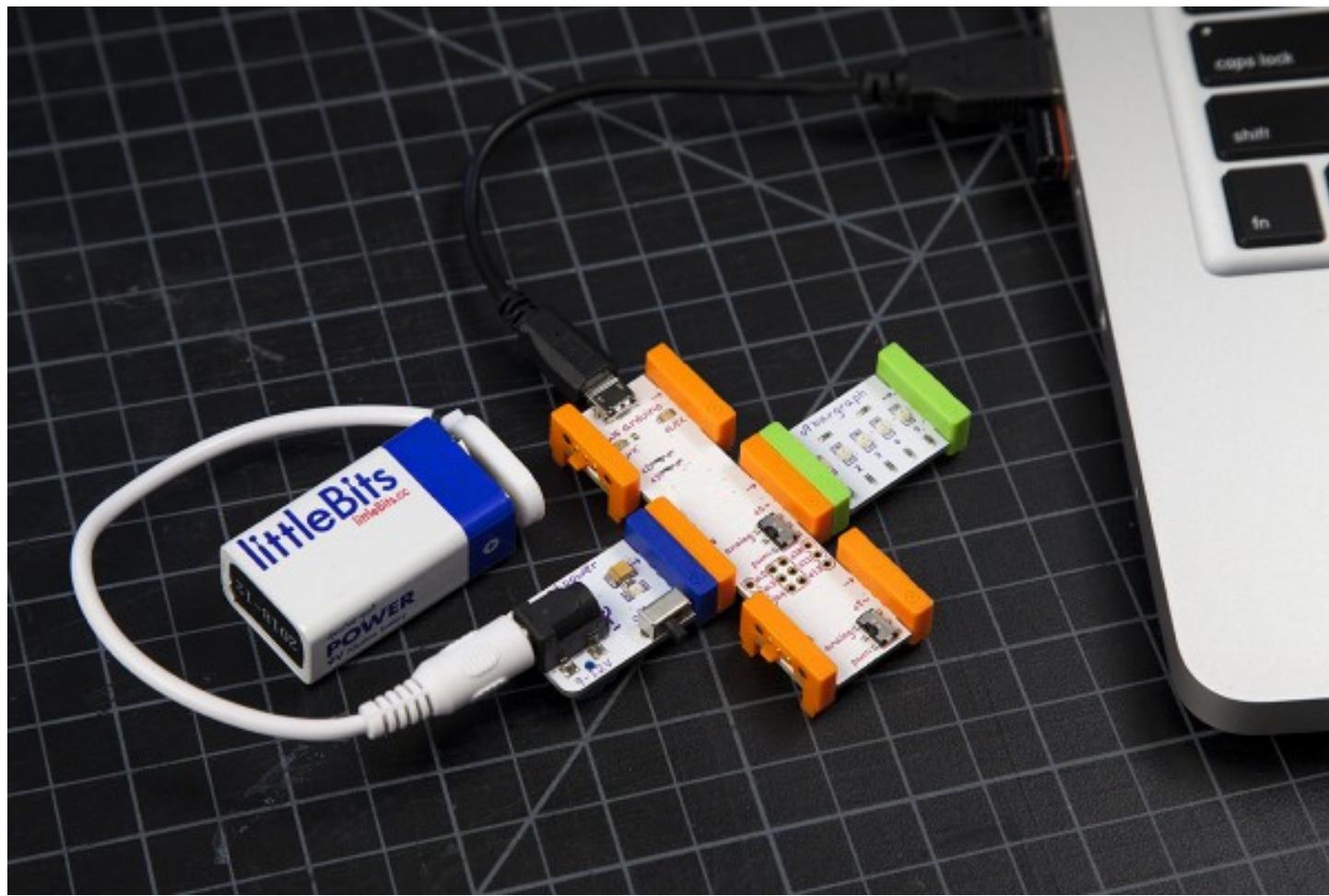
Bargraph



LittleBits Battery

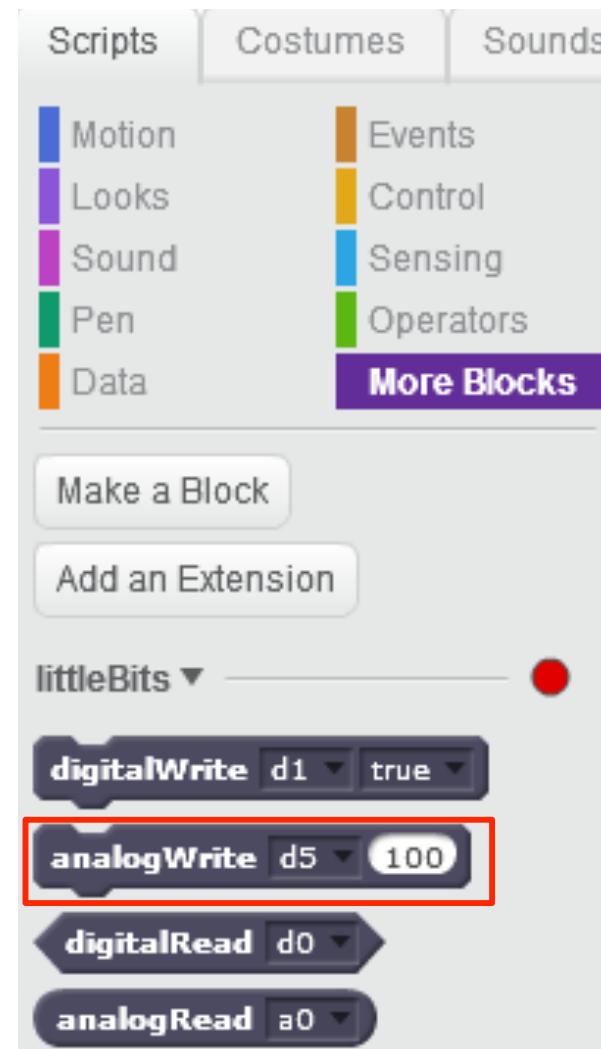
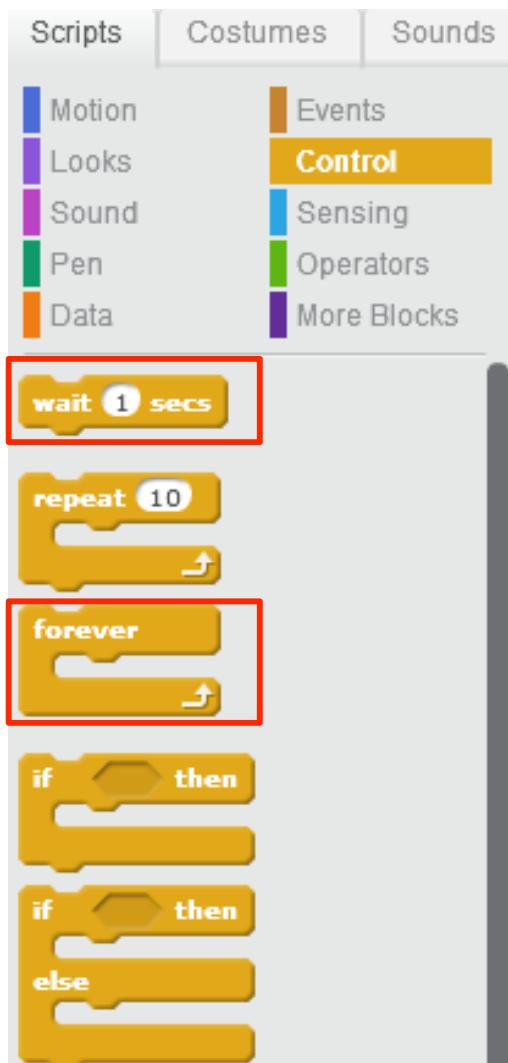


Power



Power & Battery at a0 pin
Bargraph at d5 pin

Set the program





Forever:

the function will keep repeating

analogWrite d5 100:

Set pin d5(LED) to the brightness of 100

Wait 1 secs:

hold the previous state for 1 second

analogWrite d5 0:

Set LED to 0 brightness(turn off)