Designing Smart STEM Devices with LittleBits, Microbit and Arduino

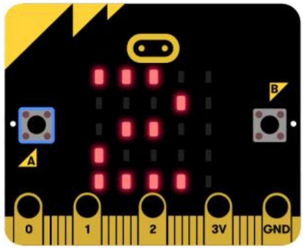
Dr. Yeung, Chi Ho Bill

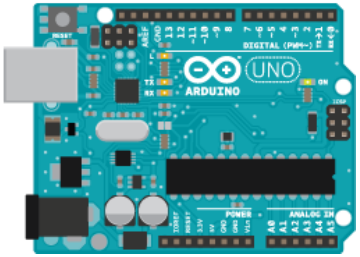
Department of Science and Environmental Studies

The Education University of Hong Kong

Objectives of the workshop

- To introduce **various electronic tools and micro-controllers for STEM education,** suitable for students at various levels
- To demonstrate the integration of **design** and **coding**





- But please note that:

1. These tools are only **platforms**, they are **not equivalent to STEM education**
2. **STEM education can be conducted without this kind of tools**
3. The introduced tools are **only a subset** of all available tools, there are other suitable tools

Smart City and Smart Home

- Smart City Consortium of Hong Kong

➢ To develop Hong Kong as a world leading Smart City, to foster the knowledge economy, **enhance quality of life** and create a vibrant eco-system by using **Information Technology** and promoting **more effective resource management**

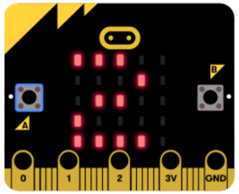
Self-driving car

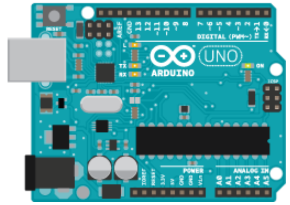
- Smart home

➢ To use technology, e.g. **internet of things (IoT),** to achieve higher home automation, or greener and smarter home (**green smart home**)

- A potential topic for STEM education - any topics in science, ICT or mathematics which can be integrated to achieve the above goals (e.g. helping elderly at home)?

Overview of the three tools





	littleBits	micro:bit	Arduino
Developer	Ayah Bdeir USA, 2011	BBC UK, 2015	Open source Italy, 2003
Suitability	Junior primary, Senior primary	Senior primary, Secondary	Secondary
Circuit connection	Magnetic electronic building blocks	Wires with crocodile clips / breadboard with jumpers	Breadboard with jumpers
Circuit knowledge required	Simple	Intermediate	Intermediate to advanced
Coding required?	No (mostly)	Yes	Yes
Coding languages	C (for Arduino bit)	Scratch-like blocks and Java Script, Python	Scratch, C
Price			

Tool 1: littleBits

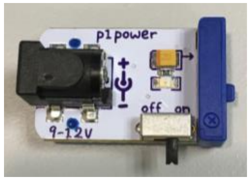
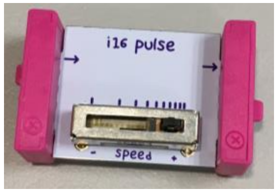
littleBits (1)

- LittleBits are **magnetic electronic building blocks**, which can be assembled as circuit to serve different purpose
- They are **color-coded** according to their functions

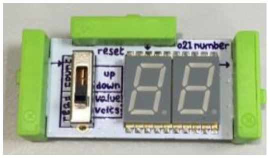
Image removed due to copyright

● Power ● Input ● Output

- Introduction of some of the bits:

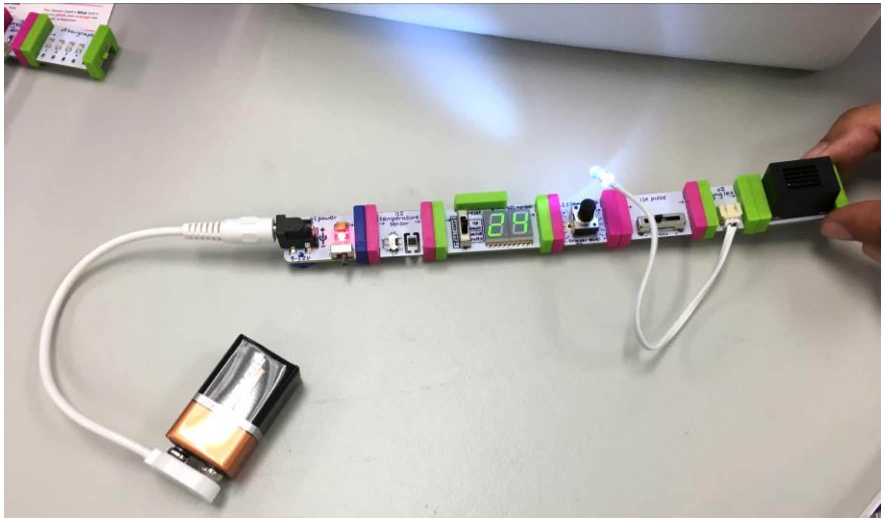
	**POWER BIT** • Power up the circuit		**PULSE BIT** • Output a series of pulse signal with adjustable interval
	THRESHOLD BIT • Convert an analog signal to a high/low signal		**TEMPERATURE SENSOR** • Operates between 0°-99°

littleBits (2)

	**NUMBER** - Display numerical numbers		**RGB LED** - LED with adjustable color
	INVERTER - Invert the binary input as an opposite output		**BUZZER** - Sound when input is high

Reference: https://shop.littlebits.cc/products/workshop-set

- Exercise: To construct a **high temperature alarm** with littleBits – an alarm which sounds when the temperature is above a threshold



IoT and Coding with littleBits

- **Internet of things** - we can construct a device to be controlled via the Internet using the “Cloudbit” and IFTTT
- **IFTTT** - a free web-based service integrated with various we applications: gmail, facebook, etc

Image removed due to copyright

- Smart-home devices can be built, e.g. pet feeders, etc
- To construct **more complicated applications**, we can use Arduino bit, combined with coding in C

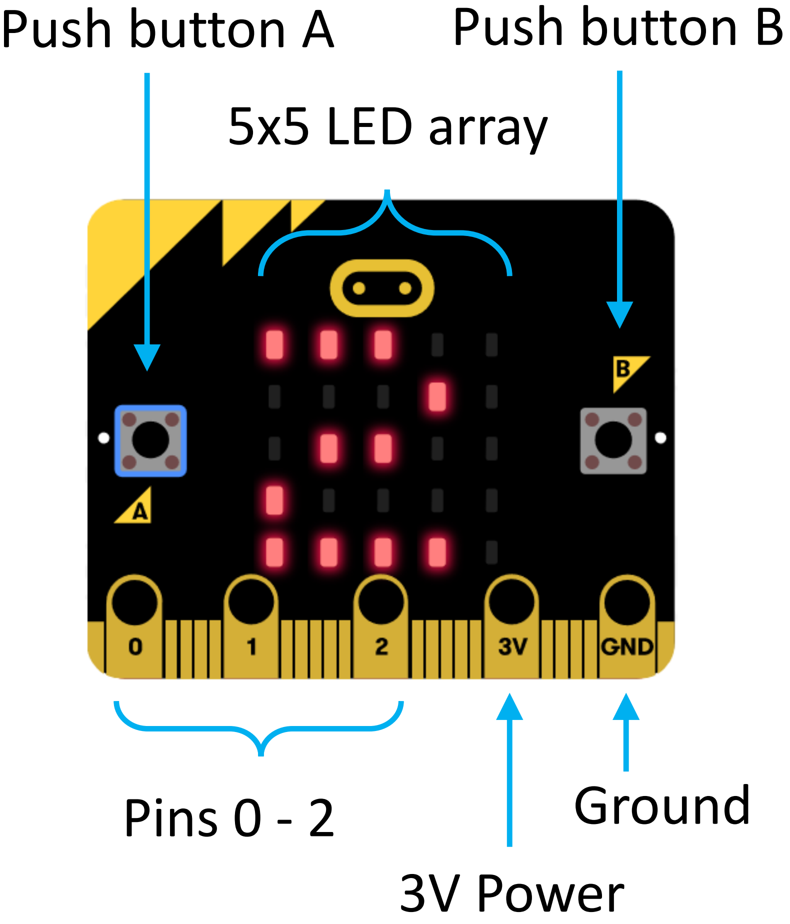
Image removed due to copyright

Tool 2: micro:bit

micro:bit

- micro:bit is a micro-controller designed by **BBC** for **computer and coding education**
- It has

1. two control buttons
2. one reset button
3. one 5x5 LED display array
4. one 3-axis accelerometer
5. one 3-axis magnetometer
6. a 2.4GHz radio module (for Bluetooth communication)



Advantages of using micro:bit?

- **Research findings** on the use of micro:bits:

1. 90% of students said the micro:bit showed them that anyone can code.
2. 86% of students said the micro:bit made Computer Science more interesting.
3. 70% more girls said they would choose Computing as a school subject after using the micro:bit.
4. 85% of teachers agree it has made ICT/Computer Science more enjoyable for their students.
5. Half of teachers who've used the micro:bit say they now feel more confident as a teacher, particularly those who say they're not very confident in teaching Computing.

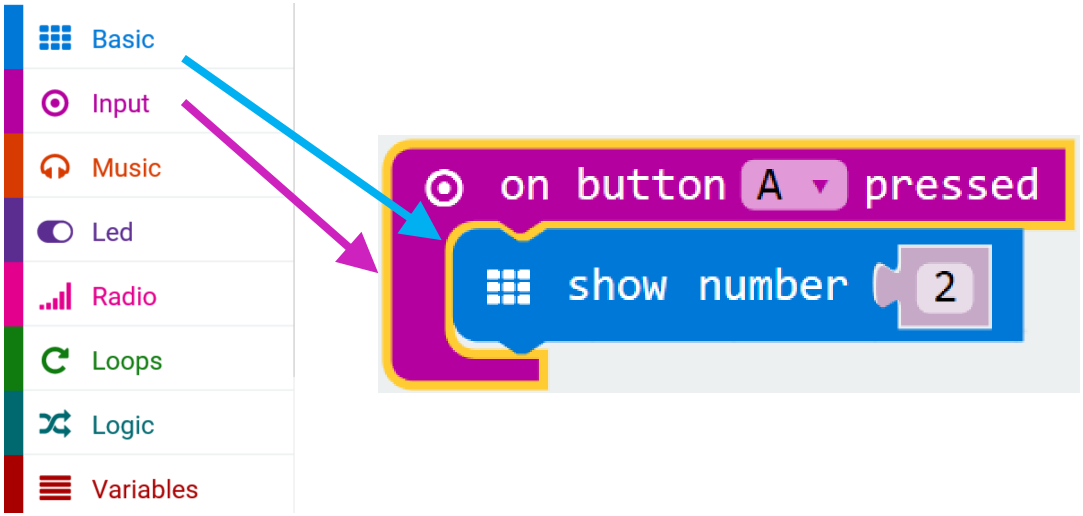
Reference:

[1] "Creating cool stuff" – Pupils' experience of the BBC micro:bit, Proceedings of the 48th ACM Technical Symposium on Computer Science Education: SIGCSE 2017, Sentance, S., Waite, J., Hodges, S., MacLeod, E., & Yeomans, L. E. (2017)

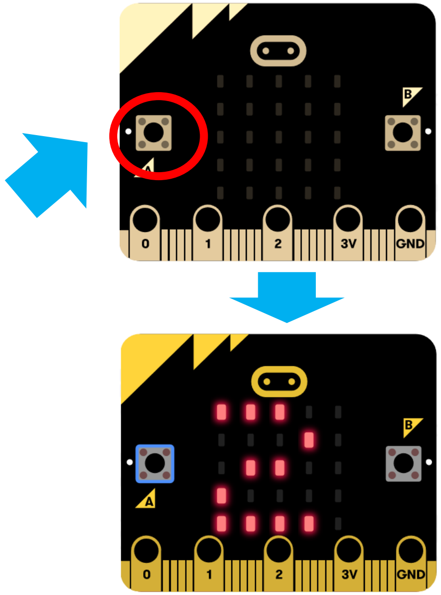
[2] Microbit's website: http://microbit.org/teach/, retrieved on 14th Sept, 2017

micro:bit – Basic operation

- Open a **browser**, go to http://microbit.org/code/
- Click "**Let's code**" under "JavaScript Blocks Editor"
- Coding blocks are grouped into various categories
- Input the following codes:



```
on button A pressed
    show number 2
```

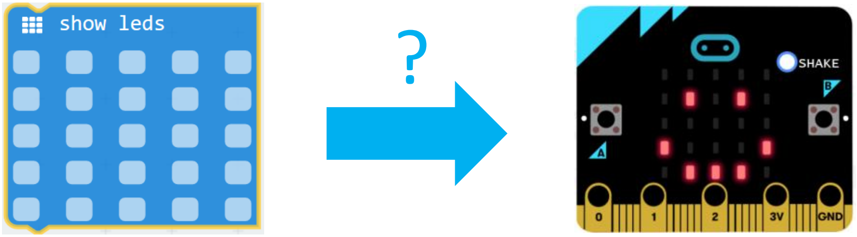


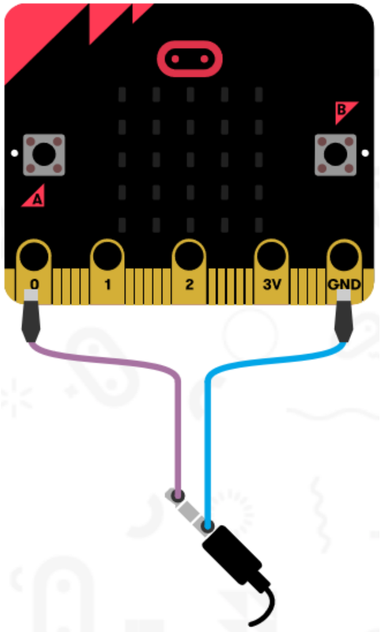
- **Connect** the micro:bit to the computer using a **USB cable**
- Click "**Download**" and save the file in the "MICROBIT (D:)" (micro:bit may be on a different drive in your computer)
- Press the "A" button on the micro:bit

micro:bit – Basic operation

- **STEAM education** – micro:bit is also a good tool for STEAM (STEM+ART) education since we can display artworks on the LED array, or code to produce music

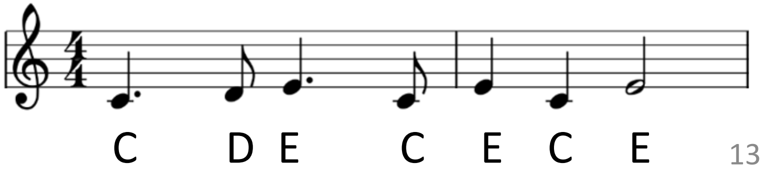
- Exercise: use the “show leds” block to code the micro:bit to display a smile for 1 second when it is shaked





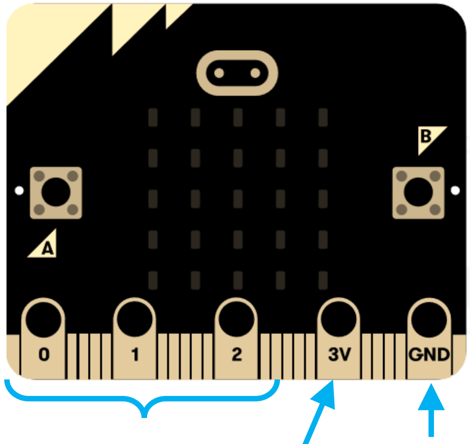
- To produce sound from micro:bit, one can **connect the micro:bit with speakers or earphones** as shown in the right figure

- Exercise: use the blocks in the “Music” category to code the melody “Do-Re-Me” in “Sound of Music”



micro:bit + input/output devices

- We can **connect** the micro:bit with **other components** using crocodile-clip wires, and the different pins on the micro:bit

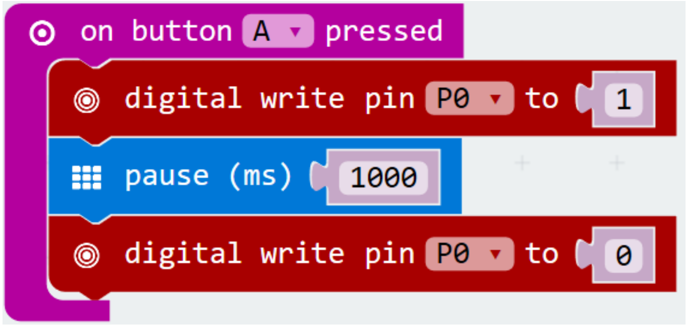


Pins 0 - 2

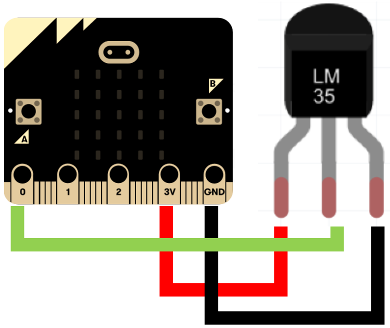
Ground

3V Power

- Example: connect an LED across pin 0 and GND, and use this code to turn it on:

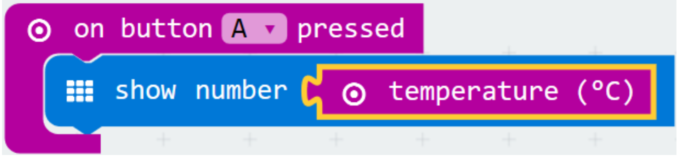


```
on button A pressed
    digital write pin P0 to 1
    pause (ms) 1000
    digital write pin P0 to 0
```



- To read data from internal/external sensors:

Reading from internal temperature sensor:



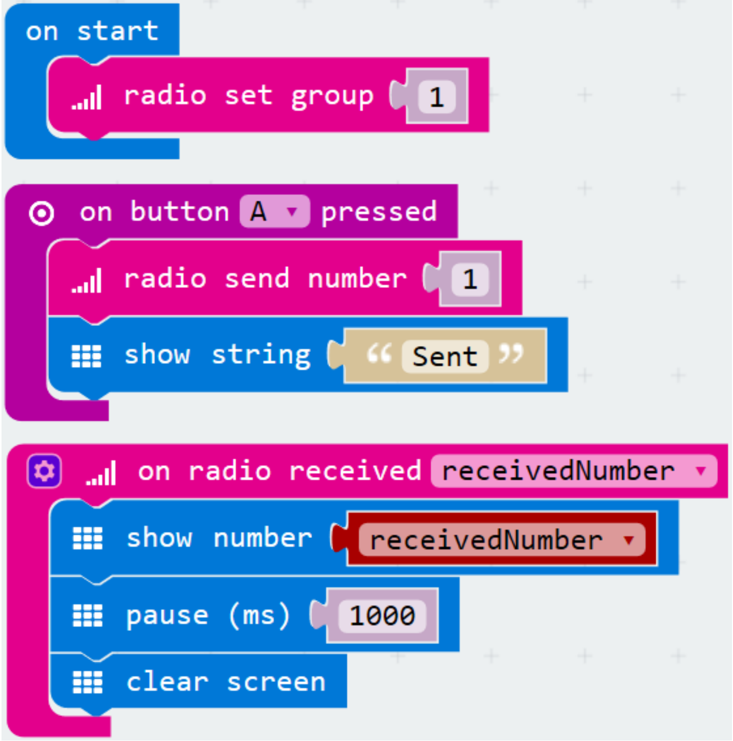
```
on button A pressed
    show number temperature (°C)
```

Reading from external temperature sensor LM35: Temperature = reading x 300/1024

Code = ?

micro:bit - bluetooth communication

- A **2.4GHz radio module** is built-in in the micro:bit for wireless communication with other devices or micro:bit
- The range of the transmission is less than 100m
- Now, pair with another group to use the following code on both micro:bit at a distance to each other

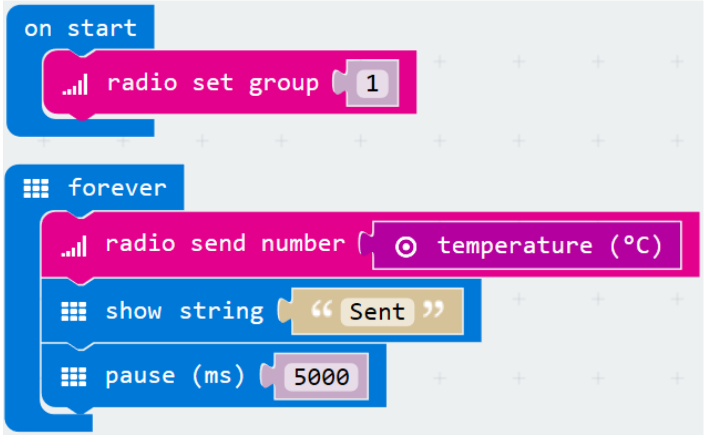


```
on start
    radio set group 1

on button A pressed
    radio send number 1
    show string "Sent"

on radio received receivedNumber
    show number receivedNumber
    pause (ms) 1000
    clear screen
```

- Exercise 1: construct a **high temperature alarm** with a micro:bit, a buzzer, and a LM35
- Exercise 2: construct **a remote temperature sensor** which send temperature reading to another micro:bit at a distance apart

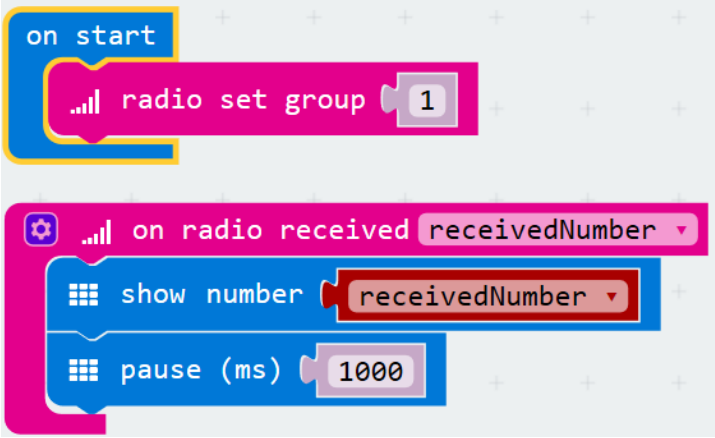
Remote sensor experiment

- Codes on the **sender** micro:bit and the **receiver** micro:bit



```
on start
    radio set group 1
forever
    radio send number temperature (°C)
    show string "Sent"
    pause (ms) 5000
```

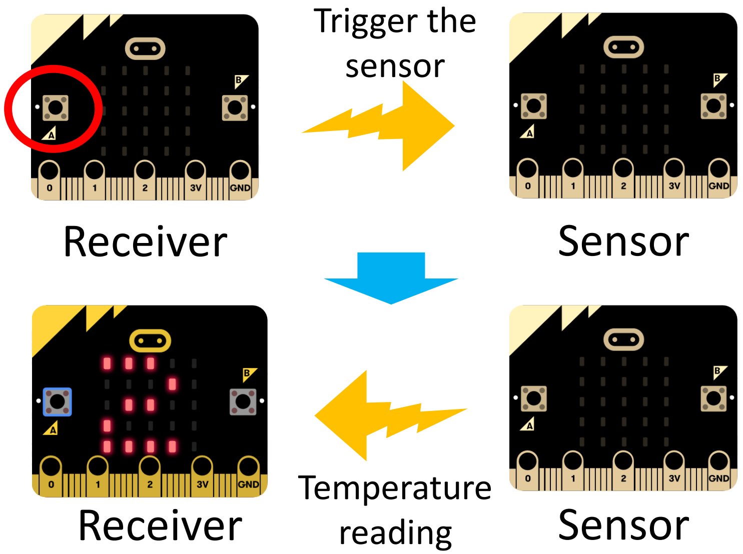
Sender (sensor) micro:bit



```
on start
    radio set group 1
on radio received receivedNumber
    show number receivedNumber
    pause (ms) 1000
```

Receiver (sensor) micro:bit

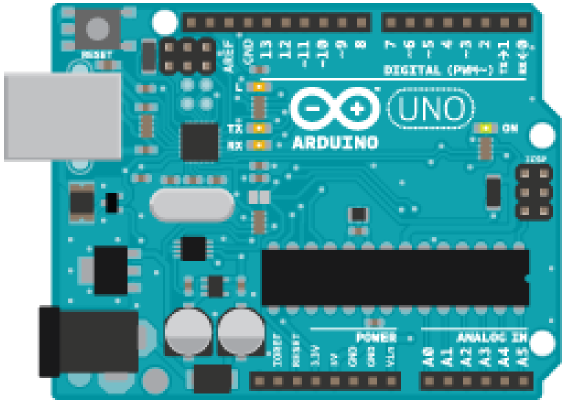
- Exercise 3: Develop **a remote sensor system** which returns the temperature of a remote sensor module only when the control button of the receiver module is triggered.



Tool 3: Arduino

Arduino microcontroller (微控制器)

- An **open-source** (開放原始碼) electronics platform based on easy-to-use hardware and software



- A (i) **programmable**, and (ii) **single-board microcontroller** which

1. **reads** inputs, e.g. sensor readings, button states,
2. **processes** the inputs based on instructions, e.g. check with codes and conditions
3. **produces** outputs, e.g. to turn on an LED, activate a motor

- Connection to different hardware - by simple **wiring** (電路)
- Instruction implementation - by the **Arduino Software IDE (Arduino IDE 軟件)** and simple programming language

Arduino microcontroller

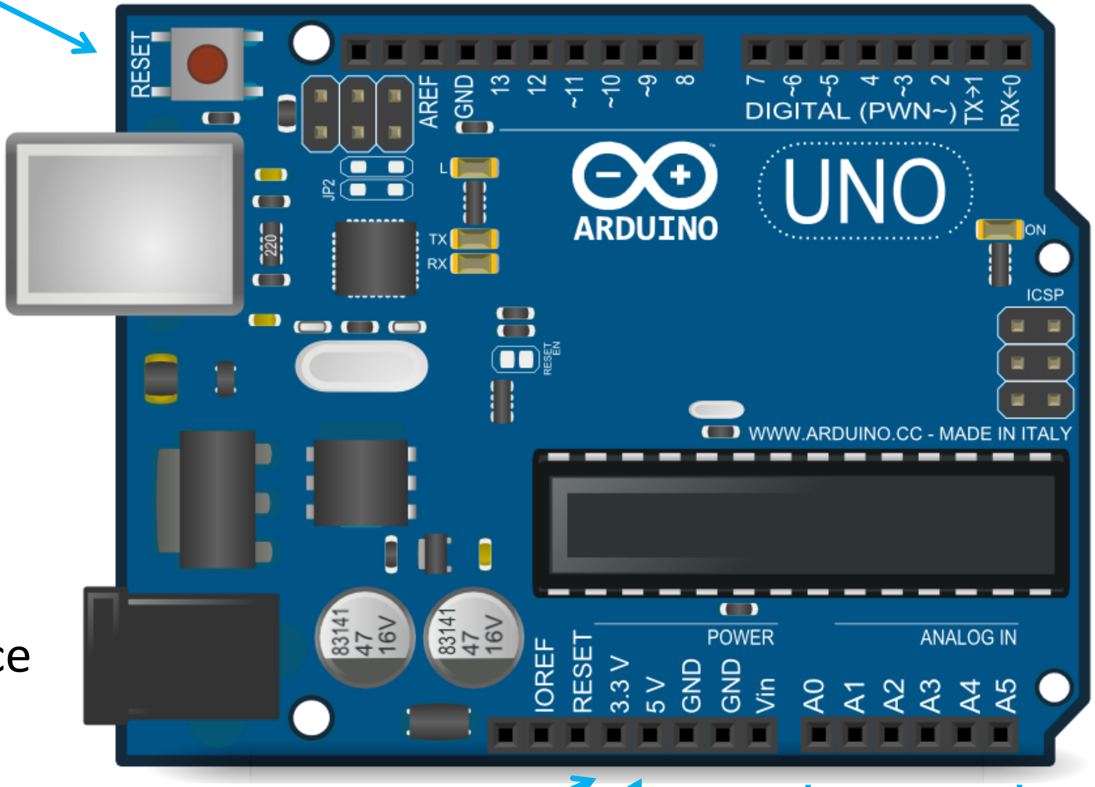
- The structure on the “**Arduino Uno**” board:

14 digital input and output pins, i.e. high or low

Reset button

USB receptacle

Power source



3V power

5V power

Ground

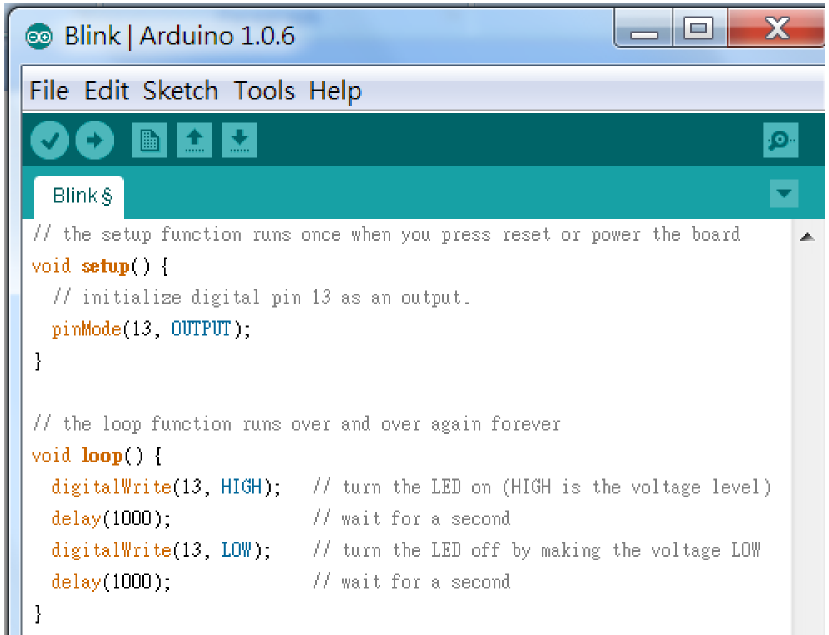
6 analog inputs
e.g. reading voltage

Photo credit:

Switch on an LED (開啓LED)

1. Connect the Arduino board to a computer by a USB cable
2. Open “**Device manager**” and **check the port** of the Arduino
3. Open the **Arduino software (IDE)** in “Tools” then “Serial port”, and select the correct port
4. In “Tools” and then “Board”, select the correct board (we are now using “Arduino Uno”)
5. Go to “File” → “Examples” → “Basics” → “Blink” to open examplar file “blink.ino”

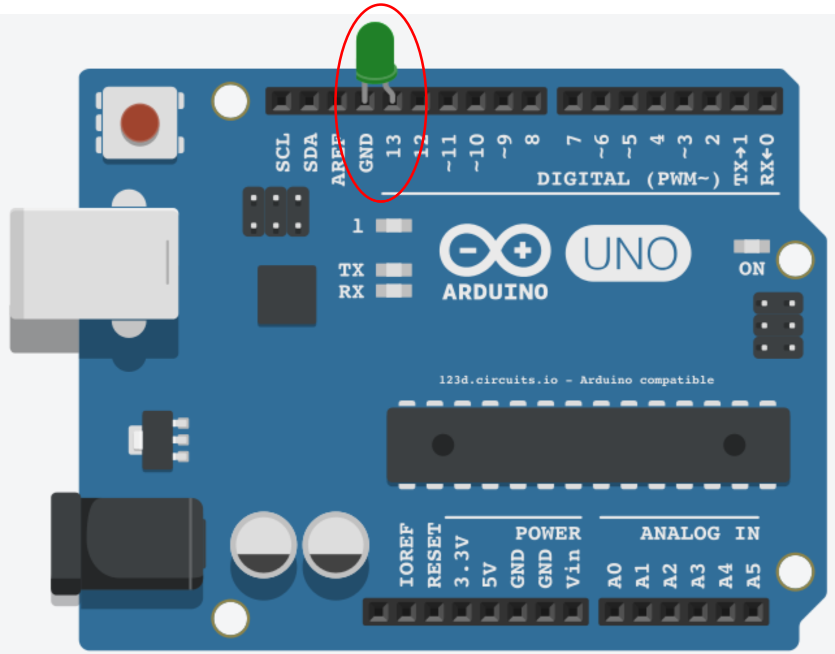
```
void setup() {
pinMode(13, OUTPUT);
}
void loop() {
  digitalWrite(13, HIGH);
  delay(1000);
  digitalWrite(13, LOW);
  delay(1000);
}
```



```
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin 13 as an output.
  pinMode(13, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(13, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);              // wait for a second
  digitalWrite(13, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);              // wait for a second
}
```

Switch on an LED (2)

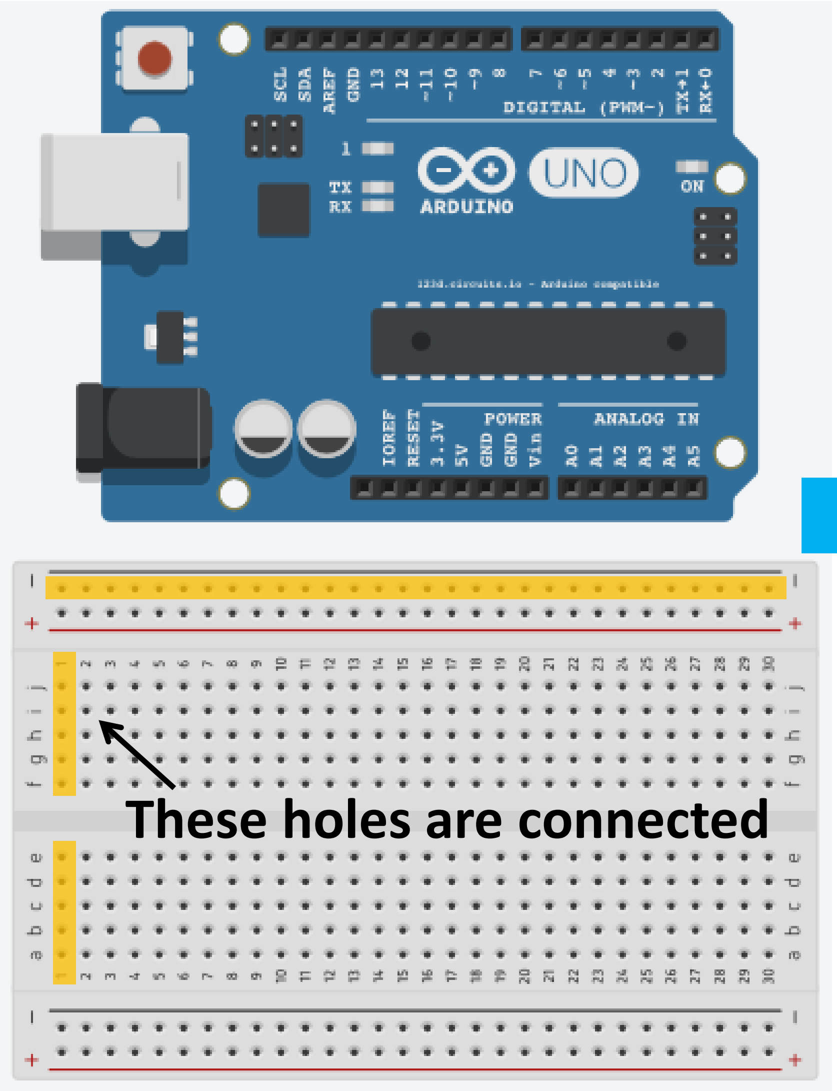
6. Click the **tick** ✓ , the program is checked (compiled)
7. If there is no problem, click the **arrow** ➔ , the program is uploaded to the broad
8. What do you observe for the on-board orange LED?
9. Connect the positive and the negative terminal of a **Green LED** to pin 13 and the ground pin respectively, what do you observe?



Exercise

1. Extend the on-state of the Green LED light to 3s
2. Construct a simple system which simulate the traffic light for pedestrian

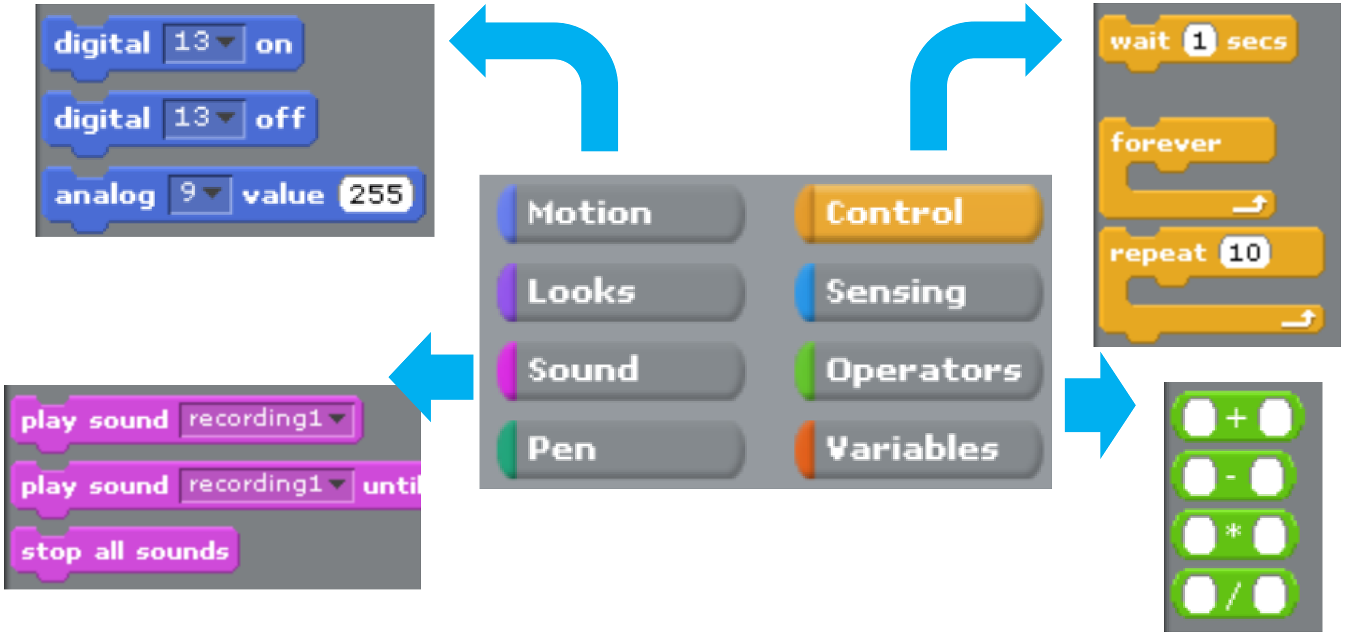
Breadboard (麵包板)



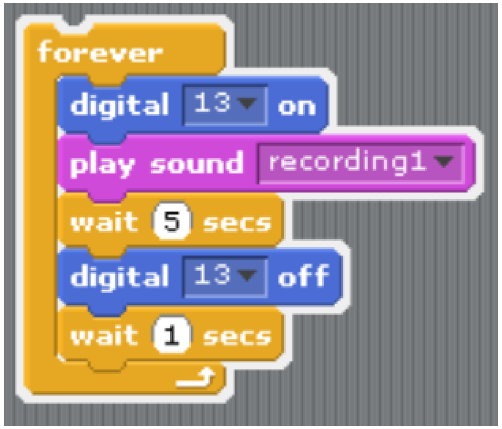
These holes are connected

S4A and ArduBlock

- **S4A** is a platform modified from **Scratch** - a graphical programming language developed by MIT which is suitable for youngsters to learn programming
- Another other common block-programming software for Arduino is called **ArduBlock**
- Although script coding is most fundamental, many codes can also be written in S4A and ArduBlock



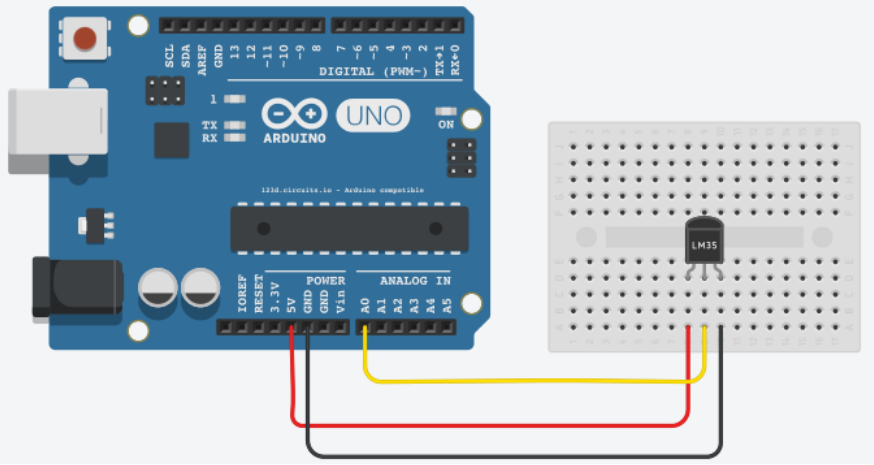
Blink.ino in S4A with sound



```
forever
    digital 13 on
    play sound recording1
    wait 5 secs
    digital 13 off
    wait 1 secs
```

Reading data (讀入數據)

- We will use the sensor **LM35** to sense **temperature**
- Connect the circuit on the left:



- This code displays the sensed temperature in the serial monitor (“Tools”→“Serial Monitor”)

```
int tempin = 0;// LM35 output connected to analog pin 0
void setup(){
        pinMode(tempin, INPUT);
        Serial.begin(9600);
}
void loop(){      // read the value of the tempin
        float reading = analogRead(tempin);
        float temperature =  (reading*5.0*100.0)/1024;
        Serial.print(temperature);
        Serial.println();
        delay(5000);    // take data every 5s
}
```

- Exercise 1: build a **high temperature alarm** which switches on an LED when temperature is too high
- Hint:

  if(temperature > 24)

  Your action;

Control over other devices (1)

- Arduino can be used to control other electrical appliances by using a **relay**, which is especially useful **in controlling a high voltage by a low voltage** (Arduino)
- **Arduino circuit** - Connect the GND and VCC pin on the relay to GND and 5V pin of the Arduino board, and connect “IN1” to one of the digital pin, e.g. pin 10
- **Relay (繼電器) circuit** -Connect the light bulb, the power supply, and the relay as shown in the circuit here:

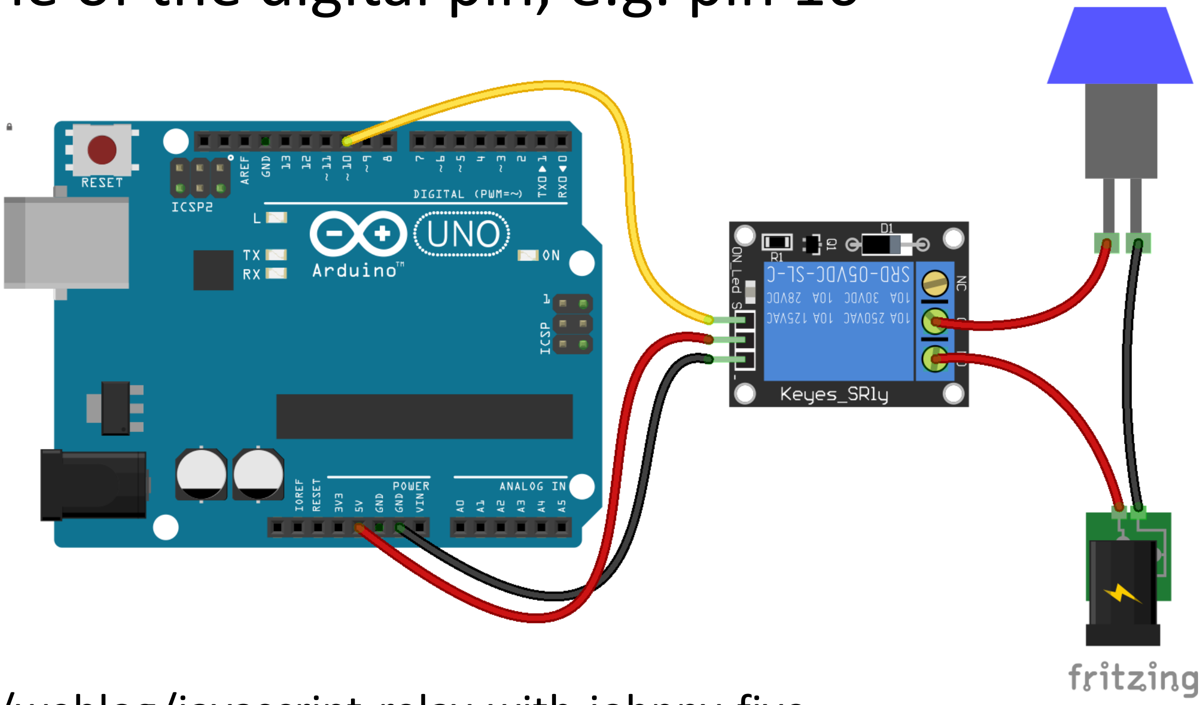


Photo credit: https://bocoup.com/weblog/javascript-relay-with-johnny-five

Control over other devices (2)

- The following code control the relay by pin 10:

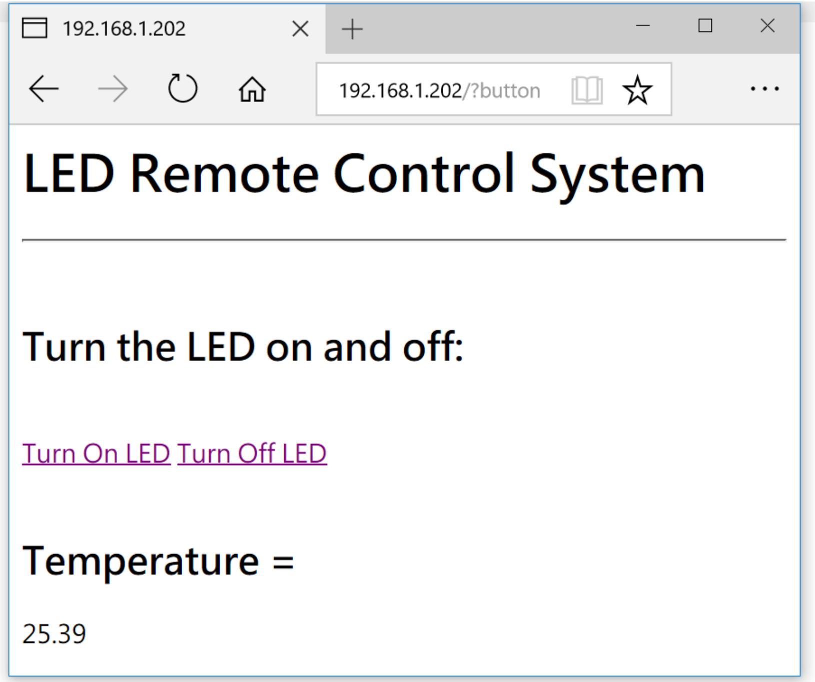
```
int relayPin = 10;

void setup() {
  pinMode(relayPin, OUTPUT);
}

void loop() {
  digitalWrite(relayPin, HIGH);   // turn the relay on
  delay(2000);              // wait for 2s
  digitalWrite(relayPin, LOW);    // turn the relay off
  delay(2000);             // wait for 2s
}
```

Exercise

1. Use the above circuit and a relay to control the on/off state of a toy car motor
2. Use a distance sensor HC-SR04 and a relay to develop a toy car which turns on/off the motor when the distance is more/less than 5cm from the wall respectively

Internet of things with Arduino

- With the **Ethernet shield**, Arduino can be used as a web server to (1) **control other devices**, or to (2) **read and report sensor readings via the internet**
- For instance, the following is an example which turn on or off an LED via a web browser connecting to **the IP of the Arduino Ethernet**, and to measure the temperature at a remote location
- For details and the exemplar codes, please refer to the attached notes



LED Remote Control System

Turn the LED on and off:

Turn On LED Turn Off LED

Temperature =

25.39

Summary

- We have introduced various electronic tools and micro-controllers including **LittleBits, micro:bit and Arduino**, suitable for students at various levels
- We have introduced the **programming languages Scratch and C** for coding these micro-controllers
- We have learnt how to **control other devices, obtain sensor readings** using micro-controllers with coding
- **Smart devices** contributing to Smart Cities, Smart Homes or IoT are good topics for **integrating science with the other disciplines in STEM**
- The introduced tools are only potential platforms for STEM education, and **STEM education can be conducted without these tools**