Use of Arduino System for Chemistry Experiments

使用「Arduino」系統進行化學實驗

活動目標：

1. 介紹運用 Arduino 系統進行實驗活動以促進化學課程的學與教；
2. 讓參加者動手運用 Arduino 系統進行實驗；以及
3. 討論如何使用 Arduino 系統促進 STEM 相關的學習活動。。

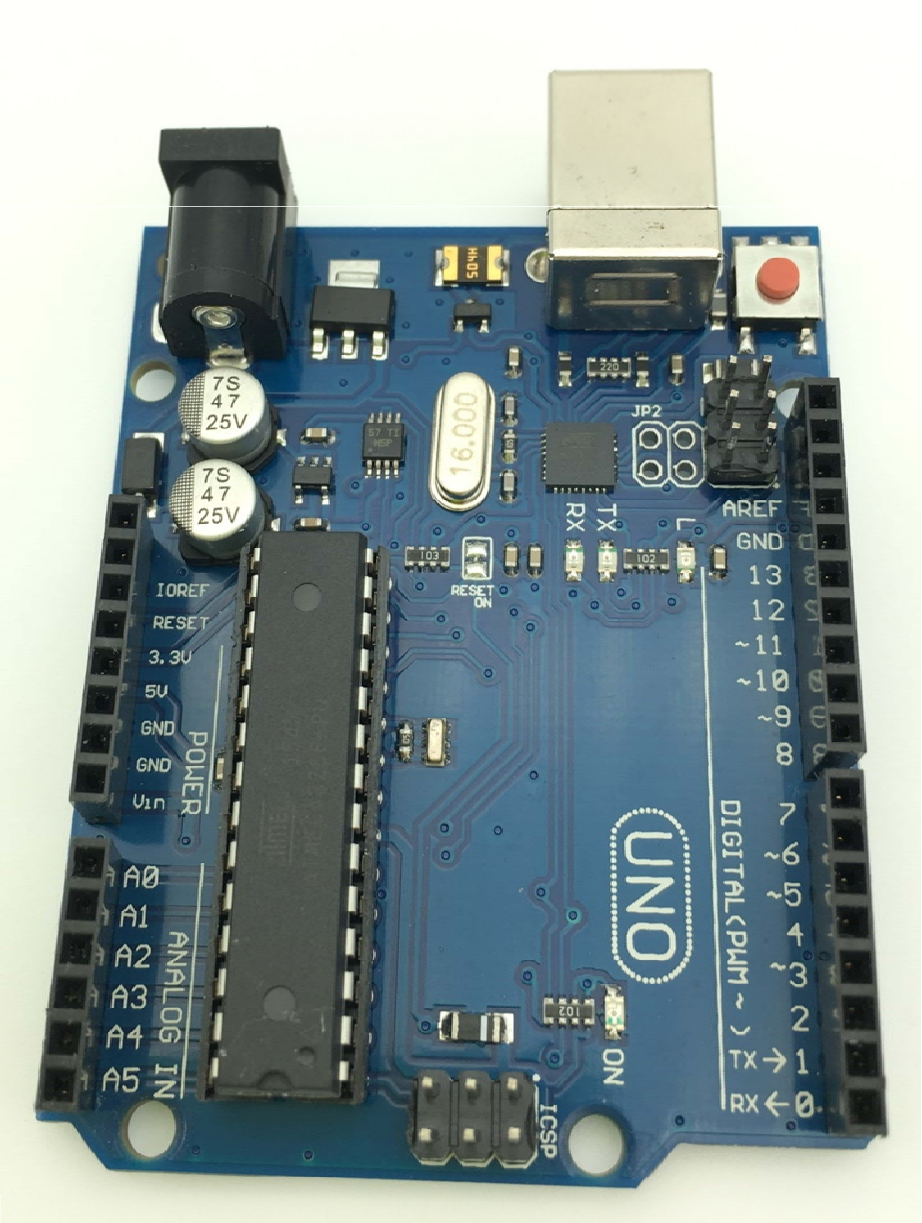
研習活動：

1. 設計並製作一個系統，閃動一顆或多顆 LED；
2. 設計並製作一個系統，量度並展示輸入的電勢；
3. 設計並製作一個酒精檢測系統，測試溶液樣本是否含有酒精；以及
4. 製作一個溫度檢測系統，並量度中和作用的焓變。

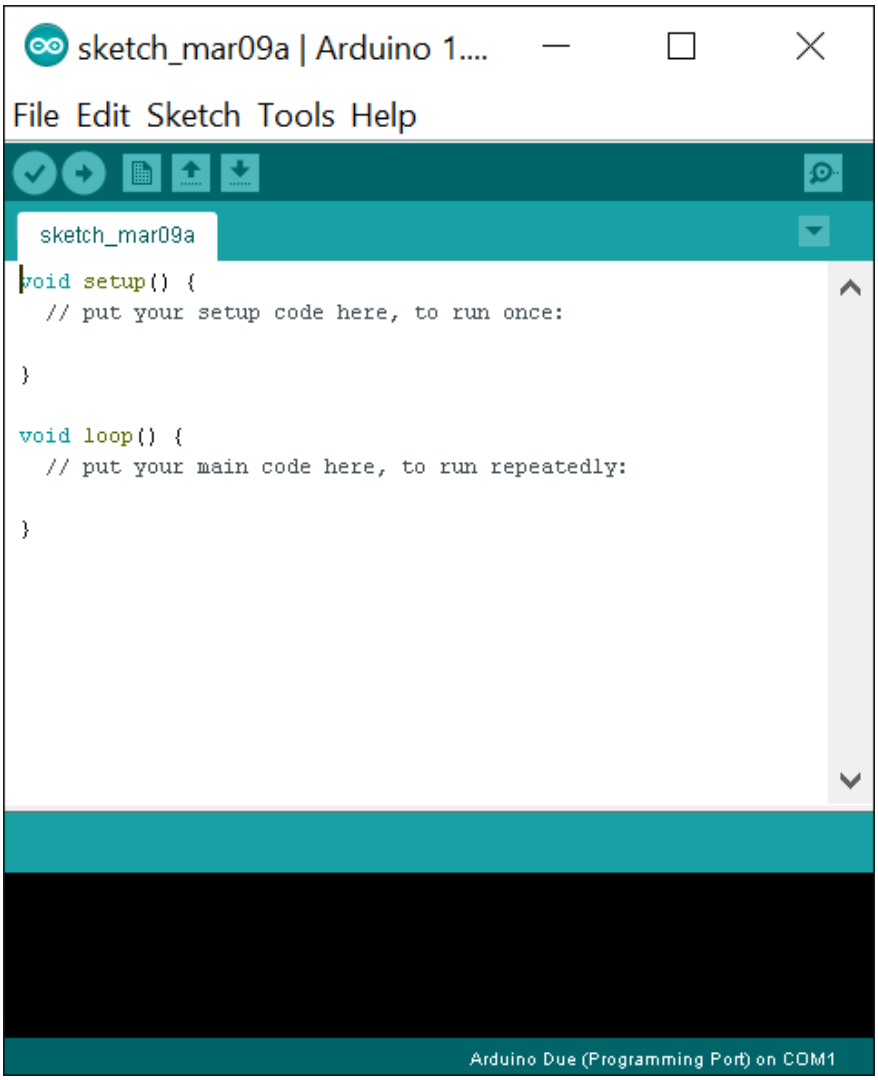
Science Education Section, Education Bureau

教育局　科學教育組

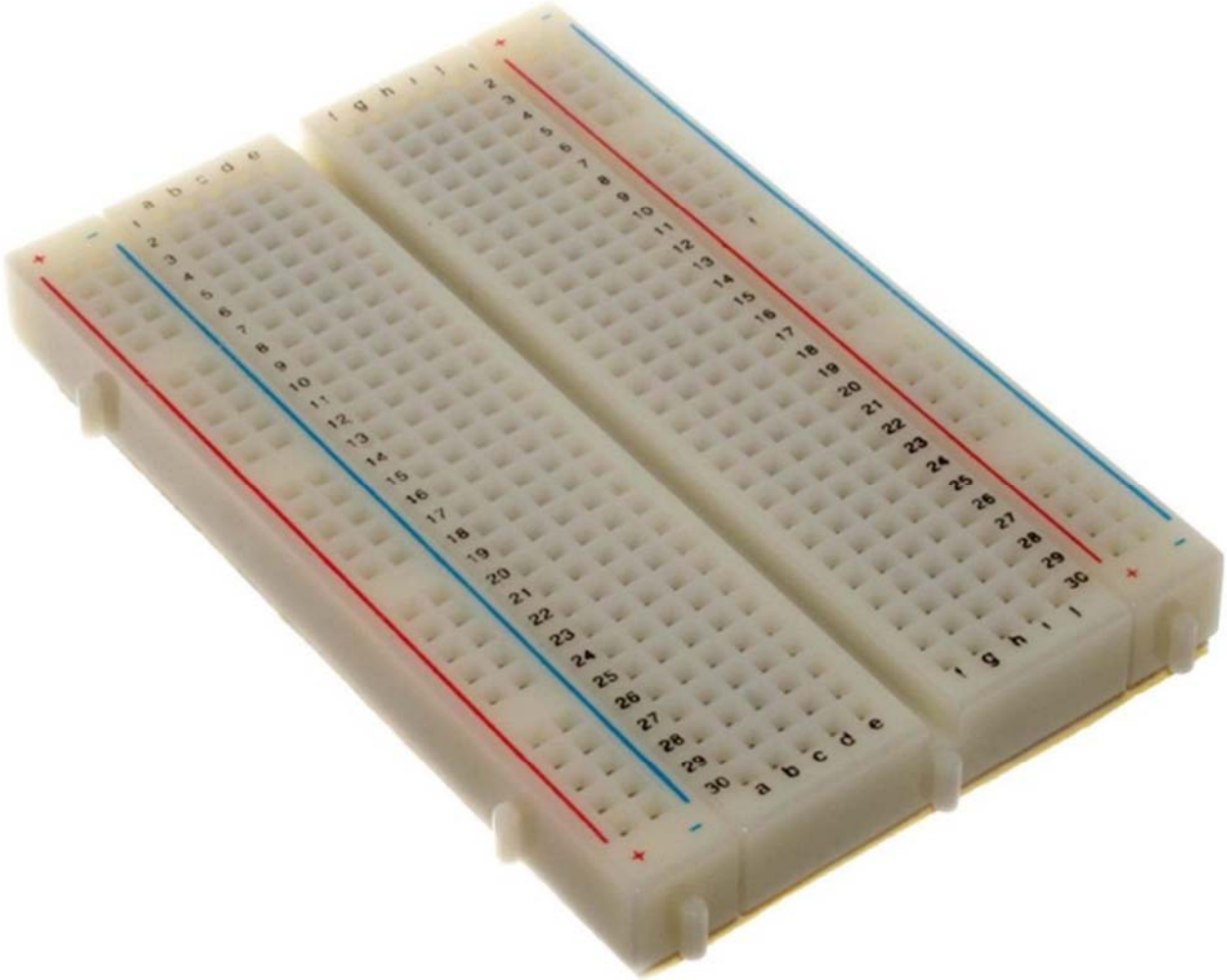
認識　Arduino Uno Rev.3



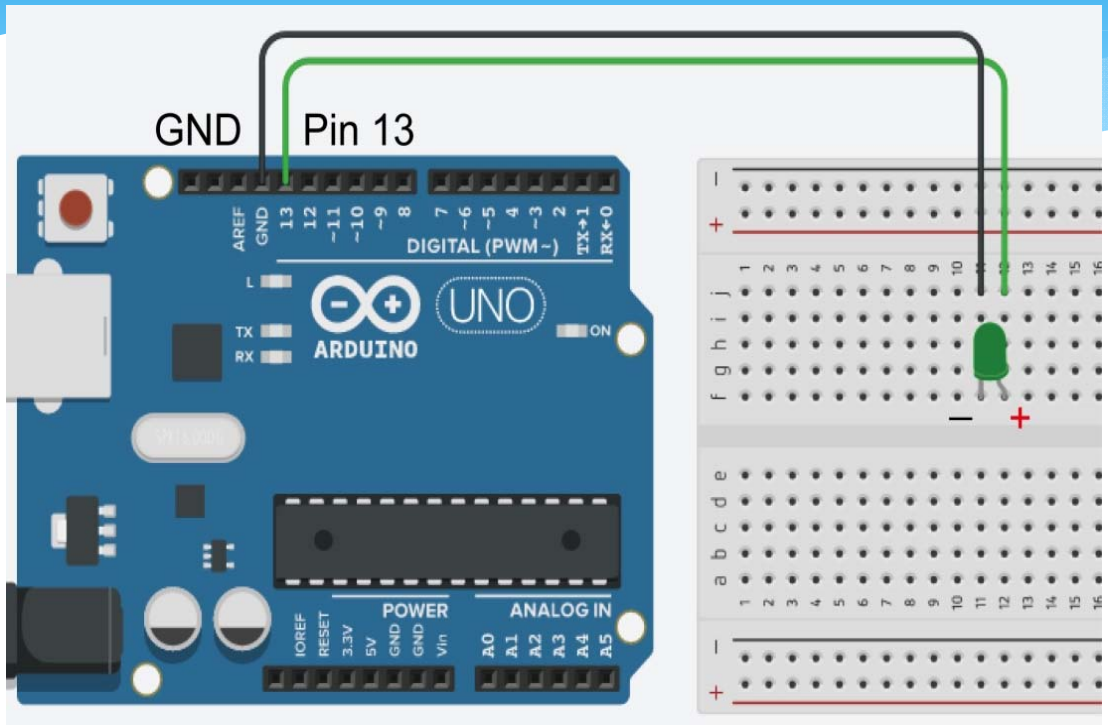
認識　Arduino　軟件



認識　試驗電路板　(Breadboard)



Example 1: Flashing LED



Example 1: Flashing LED

Human Language

Initialise an integer variables to store data like pin number

Setup the system

- Set a pin for power output

Repeat the loop again and again,

- Turn on the LED for 1000 ms
- Turn off the LED for 500 ms

Loop again

Arduino Codes

```
int pinLED = 13;

void setup() {
  pinMode(pinLED, OUTPUT);
}
void loop() {
  digitalWrite(pinLED, HIGH);
  delay(1000);
  digitalWrite(pinLED, LOW);
  delay(500);
}
```

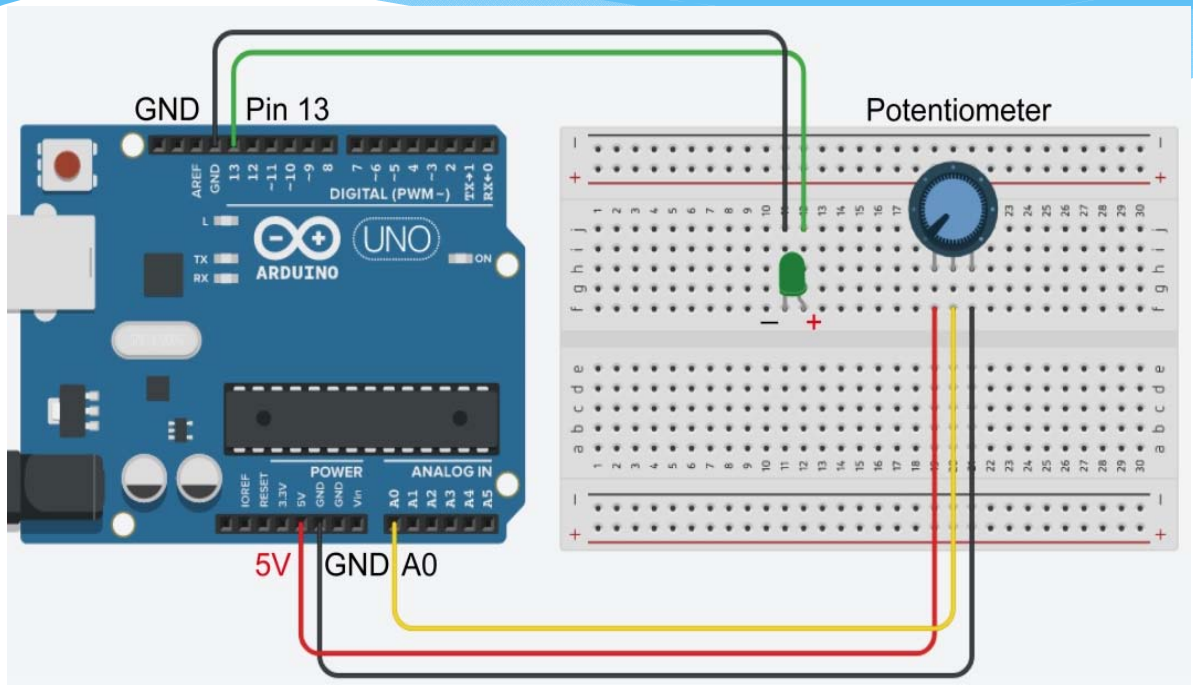
Example 1: Flashing 2 LEDs

Ardunio Codes

```
int pinLED1 = 13;
int pinLED2 = ___;
void setup() {
  pinMode(pinLED1, OUTPUT);
  pinMode(pinLED2, OUTPUT);
}
void loop() {
  digitalWrite(pinLED1, HIGH);
  ...
  delay(1000);
  digitalWrite(pinLED1, LOW);
  ...
  delay(500);
}
```

My Note

Example 2: Reading Voltage from a Potentiometer



Example 2: Reading Voltage from a Potentiometer

Human Language

Initialise some integer variables to store data like pin number and sensor reading

Setup the system

- Set a pin for power output
- Set a pin for voltage input

Repeat the loop again and again,

- Read the voltage input
- An If-else logic decision,
- If sensor value is more than or equal to 512,
  - turn the LED on
  - else turn the LED off

Loop again

Arduino Codes

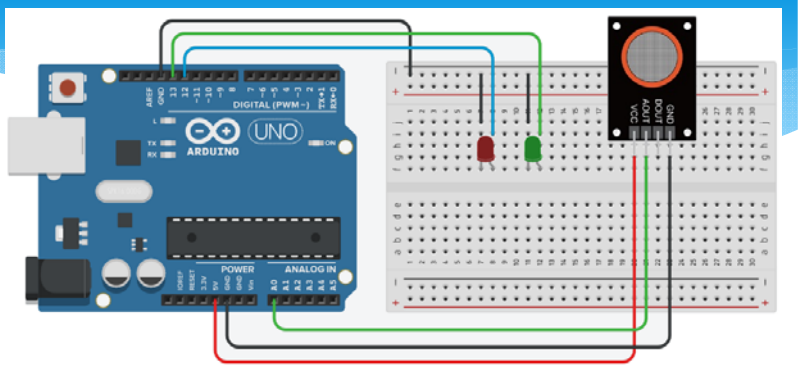
```
int pinLED = 13;
int pinPOT = 0;
int sensorValue = 0;
void setup() {
pinMode(pinLED, OUTPUT);
pinMode(pinPOT, INPUT);
}
void loop() {
 ...
 sensorValue = analogRead(...);
 if (sensorValue >= 512) {... ;}
 else {... ;}
}
```

Example 2: Showing Different Voltage Readings (e.g. 256, 768, …)

* Just 1 LED – different flashing modes, …
* 1 LED + 1 buzzer, …
* 2 LEDs, …
* any feasible way your prefer

My Note

Example 3: Detecting the presence of alcohol vapour using an alcohol sensor connected to an Arduino UNO board




```
int pinLED1 = 12;
int pinLED2 = 13;
```

or

```
int pinLED1 = 10;
int pinLED2 = 11;
```

```
if (sensorValue >= 128) {digitalWrite(pinLED1, HIGH); digitalWrite(pinLED2, LOW)}
else
{digitalWrite(pinLED1, LOW); digitalWrite(pinLED2, HIGH)}
```

Example 3: Detecting Alcohol using 2LEDAS

Human Language

Initialize some integer variables to store data like pin number and sensor value

Setup the system

- Set the communication rate between sensor and our computer
- Set pins for power output, reading input, etc.

Repeat the loop again and again,

- Read sensor value from A0 pin, print the sensor value at the Serial monitor
- An If-else logic decision,
- If sensor value is smaller than 128, turn only the LED1 on
- else turn only the LED2 on

Loop again

Arduino Codes

```
int pinLED1 = 12;
int pinLED2 = 13;
int pinPOT = A0;
int sensorValue = 0;
void setup() {
Serial.begin(9600);
pinMode(pinLED1, OUTPUT);
pinMode(pinLED2, OUTPUT);
pinMode(pinPOT, INPUT);
}
void loop() {
sensorValue=analogRead(pinPOT);
Serial.println(sensorValue);
if(sensorValue >= 128) {
    digitalWrite(pinLED1, HIGH);
    digitalWrite(pinLED2, LOW);
  }
  else {
    digitalWrite(pinLED2, HIGH);
    digitalWrite(pinLED1, LOW);
  }
}
```

Example 3: Detecting and Showing Different Level of Alcohol

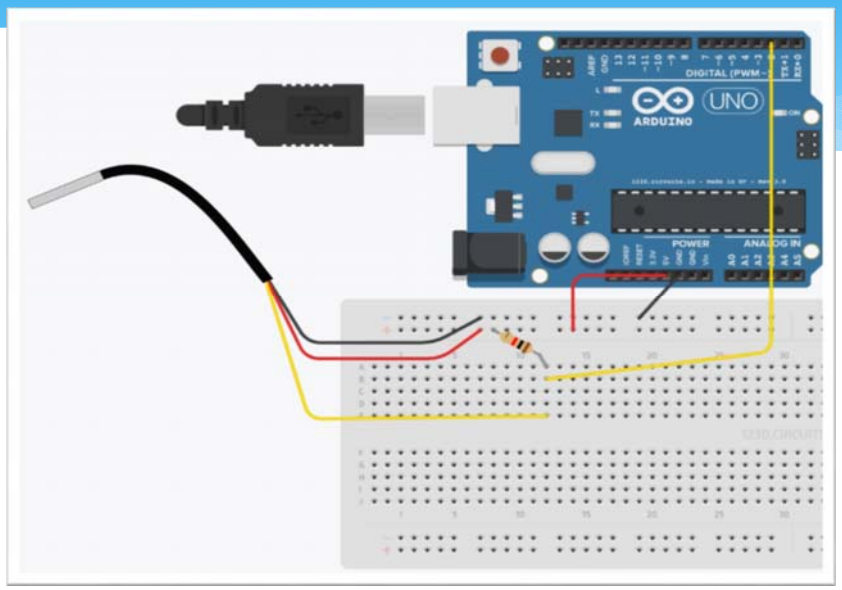
- 2 LEDs – different flashing modes, …
- 2 LEDs + 1 buzzer, …
- More than 2 LEDs, …
- any feasible way your prefer

Experiment
Breathlyser - 2 LEDs alcohol arduino system

https://www.youtube.com/watch?v=Ac6nOvpjegY

My Note

Example 4: Enthalpy of Neutralisation



Key features:

1. **Include libraries (pre-written codes that add new functions)**
2. **Use the Serial Monitor**

Example 4: Enthalpy of Neutralisation

Arduino Codes

```
#include <OneWire.h>                    //This is the required library of the thermometer
#include <DallasTemperature.h>          //This is the required library of the thermometer
#define ONE_WIRE_BUS 2                  //Set Digital pin 2 as the data pot of thermometer
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);
long newtime = 0;                       //Declare a variable to store current time
long oldtime = 0;                       //Declare a variable to store start time

void setup() {
Serial.begin(9600);                     //Switch on the "Serial Monitor" for display
Serial.println("Time,  Temp");          //Print Time, Temp in "Serial monitor"
}
void loop() {
while(newtime<=150000){                 //carry out the experiment for 150 seconds
oldtime = millis();                     //Set the start time
sensors.requestTemperatures();          //Request temperature from thermometer
newtime=millis();                       //Update the current time
Serial.print(newtime/1000);             //Print the running time in second
Serial.print("  ,  ");
Serial.println(sensors.getTempCByIndex(0));  //Print the temperature
while(newtime - oldtime < 1000){        //Set sample rate = 1s
newtime = millis();
}
}
}
```

My Note

Reference

1. http://www.arduino.cc
2. http://arduino.tw/
3. https://sites.google.com/site/cdichem/
4. http://www1.cpshs.hcc.edu.tw/cparduino/
5. https://www.youtube.com/watch?v=Ac6nOvpjegY

Common Arduino Codes and Commands

Structure

```
void setup() {
//code runs once at the beginning of the code execution.
}
void loop() {
//code runs repeatedly as the board is powered.
}
```

Setup pin Mode

Syntax:

```
pinMode(pin, MODE);
//Sets the mode of the digital I/O pin.
//Mode can be set as an INPUT, OUTPUT, or INPUT_PULLUP
```

Example:

```
pinMode(10, OUTPUT);
//Set digital pin 10 as power OUTPUT
```

Logic

Syntax:

```
if(condition) { // if condition is TRUE, do something
}
else { // otherwise, do this
}
```

Example:

```
//Assume we declared an interger a = 1
if(a == 1){
Serial.println("True");}
else{
Serial.println("False");}
//The serial monitor will print out True as a is equal to 1
```

Analog I/O

Syntax:

```
int var = analogRead(pin);
//reads the value from a specified analog pin.
```

Example:

```
int reading = analogRead(A0);
//store the reading from A0 pin to reading
```

Delay

Syntax:

```
delay(millisecond);
//pauses the program for the amount of time specified
```

Example:

```
delay(1000);
//pause the program for 1 second
```

Declare a variable

Syntax:

```
datatype variable-name;
//A variable is a way of naming and storing a value for
later use by the program
```

Example:

```
int Variable1 = 0;
float Variable2 = 0;
//you can assign the variable with a meaningful name
```

Using a buzzer

Syntax:

```
tone(pin, frequency);   //Range from(31 to 65535Hz)
noTone(pin);            //Stop generating the tone
```

Example:

```
tone(11, 8000);
//the buzzer connected to pin 11 give out 8000Hz
noTone(11);             //turn it off
```

DigitalWrite

Syntax:

```
digitalWrite(pin, value);
//write a HIGH(5V) or a LOW(0V) value to a
digital pin.
```

Example:

```
digitalWrite(13, HIGH);
//sets the pin 13 to a certain voltage
```

Mathematical Operators

=	// assignment
+	// addition
-	// subtraction
*	// multiplication
/	// division
%	// modulus

Logical Operators

==	// boolean equal to
!=	// not equal to
<	// less than
>	// greater than
<=	// less than or equal to
>=	// greater than or equal to
&&	// Boolean AND
||	// Boolean OR
!	// Boolean NOT

Data Types

int	//store integer, 16bits
float	//store value with decimal, 32 bits
boolean	//store 0 or 1 only
char	//store ASCII character, 8 bits

Alcohol Content in Samples using 2-LED Alcohol Sensing System

Objective

The purpose of this experiment is to construct an alcohol sensing system and to detect the presence of alcohol.

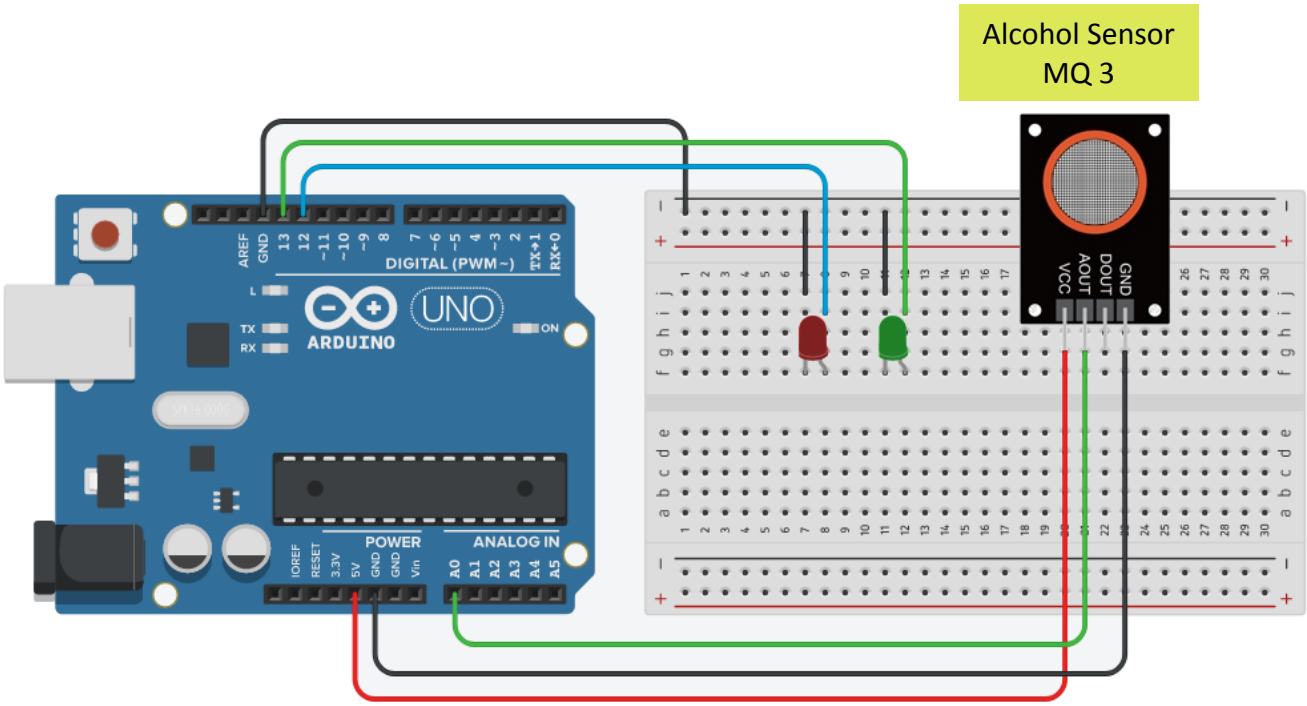
Curriculum link

Topic VIII Redox Reactions, Chemical Cells and Electrolysis

Topic XI Chemistry of Carbon Compounds

Construction of a 2-LED Alcohol-sensing System (2LEDAS)

Part A Constructing the circuit:



Part B Uploading the programme codes to the Arduino board:

1. Connect your 2LEDAS with the given USB cable to a computer with Arduino IDE, and upload your codes.
2. Allow the 2LEDAS to warm up for a while before conducting the experiment. This can be done by allowing the 2LEDAS to operate for about 2 or 3 minutes in alcohol free environment
3. Check if the green LED is on and the red LED is off. If yes, the 2 LEDAS is ready for use.

Detecting the presence of alcohol using the 2LEDAS

1. You are given 5 solutions, A to E, which may or may not contain alcohol.

2. By using your 2LEDAS, detect for the presence of alcohol in the 5 solutions:

   a. Place your MQ-3 sensor on the top of sample A as shown, and wait for at least 15 seconds.
   b. Wait until the red LED turns off, repeat the detection for the remaining solutions.

   (Precaution: do not dip the sensor into the solution)

3. Result:

Solution	Red LED on or off	Presence of Alcohol in the sample
A		
B		
C		
D		
E		

Enthalpy Change of Neutralisation with an Arduino system

Objective

The purpose of this experiment is to determine the enthalpy change for the following reaction: $HCl(aq) + NaOH(aq) \rightarrow NaCl(aq) + H_2O(l)$

Background

Arduino boards are microcontrollers with versatile functions. People can build digital devices and many interactive objects with sensors and electronic components compatible with Arduino board. A sensor, including the one to be used in this experiment, will generate a voltage based on a specific stimulus from the environment. The received voltage reading will be processed by Arduino board according to the instructions (or codes uploaded to the board). With linkage of different electronic components to an Arduino board, the device built can be used for logging data for investigations or other purposes (e.g. response to environmental stimulus with some appropriate actions).

In this experiment, by adding an excess amount of aqueous dilute NaOH(aq) to a known amount of dilute HCl(aq), and then measuring the temperature change by a temperature sensor over a period of time, the enthalpy change of the reaction can be calculated.

Curriculum link

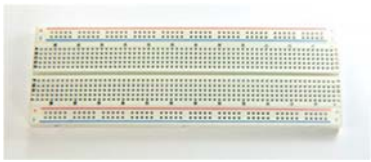
Topic IV	Acids and Bases
Topic VIII	Chemical Reactions and Energy

Chemical and apparatus (per group)

- About 1M NaOH(aq) solution — 50 cm^3
- 1.0 M HCl(aq) solution — 25.0 cm^3
- Measuring cylinder, 100 cm^3 — x 1
- Beaker, 250 cm^3 — x 2
- Pipette, 25.0 cm^3 — x 1
- Polystyrene cup, around 150 cm^3 — x 1
- Wash bottle — x 1
- Goggles — x 1 per participant

Computer and Arduino

- Computer (with pre-installed Arduino IDE and a spreadsheet software) — x 1
- Arduino UNO and USB cable — x 1
- Solderless breadboard (large) — x 1



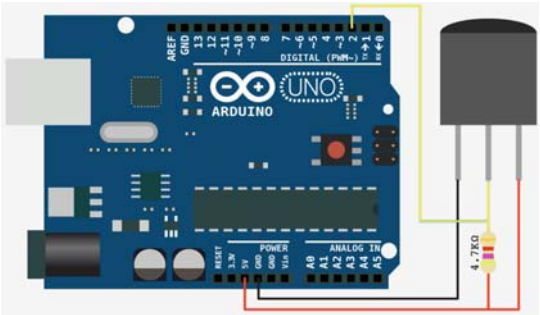
- Jumper wire (Male to Male) x 5
- Resistor (4.7kΩ)  x 1

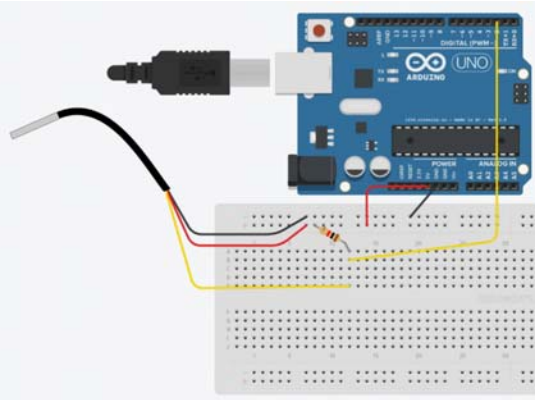
Procedure

Part A: Installation of Software and Arduino board

Download and install the Arduino software (or Arduino integrated development environment (IDE)) from https://www.arduino.cc/en/main/software

Part B: Constructing the circuit





1. Use a wire to connect 5V (5 Volt) port of an Arduino board to + (Positive) hole on the breadboard.
2. Use another wire to connect GND (Ground) port of the Arduino board to – (Negative) hole on the breadboard.
3. Use another wire to connect Digital 2 port of the Arduino board to ***b12*** on the breadboard.
4. Connect one end of a resistor to the + (Positive) hole and the other end to ***a12*** on the breadboard.
5. Connect the RED, BLACK and YELLOW wires of the temperature sensor to +, – and ***e12*** holes on the breadboard respectively.

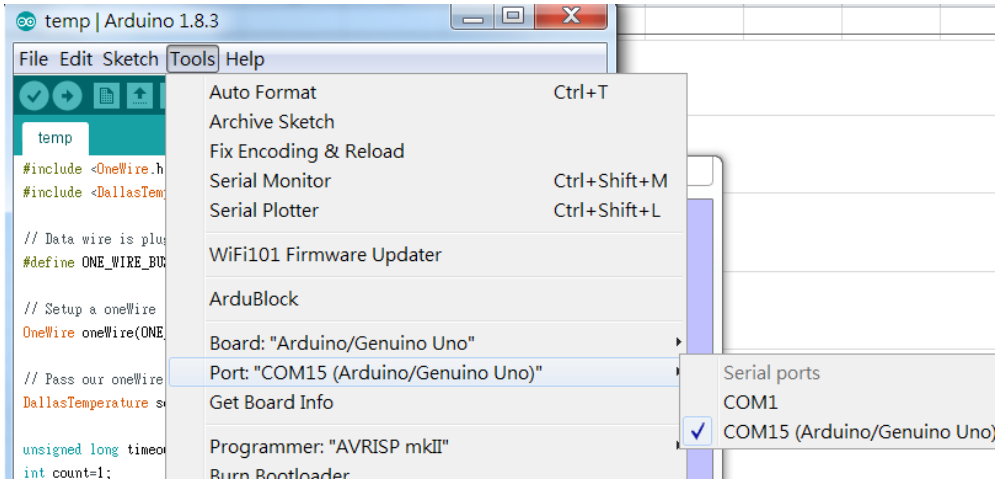
Part C: Arduino board setup

1. Arduino Board connection

a. Connect your Arduino board to computer via the provided USB cable.

b. Start the Arduino IDE and wait for the computer to install the Arduino driver.

c. Select the correct device according to your Arduino board (i.e. Uno) from the menu: Tools > Board & Tools > Port.

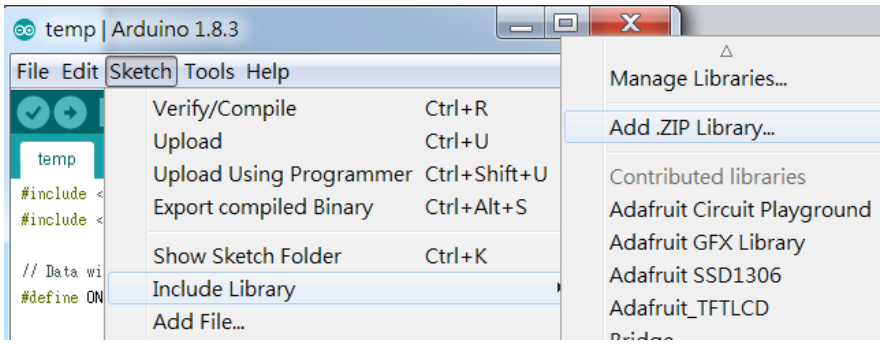


2. <u>Arduino Library installation for the temperature sensor</u>

a. Locate the two library files, i.e. OneWire.zip and DallasTemperature.zip, from the folder "Arduino_EDB".

b. Go to menu: Sketch > Include Library > Add .ZIP Library…

c. Add the OneWire and DallasTemperature libraries respectively



3. <u>Codes for temperature measurement</u>

a. Copy the required codes from the file "temp.txt" in the folder "Arduino_EDB".

b. In the Arduino IDE window, clear all the existing codes and paste the codes.

c. Save the file and check the codes by pressing  the verify button.

d. Upload the codes to your Arduino board by pressing  the upload

Part D: Measurement

1. Using a pipette, put 25.0 cm^3 of the given aqueous HCl solution (1.0 M) into the polystyrene cup.
2. Using a measuring cylinder, measure 50 cm^3 of the given aqueous NaOH solution (about 1 M).
3. Put the thermometer sensor into the polystyrene cup.
4. When the solutions attained the room temperature (remains more or less constant over a period of time), start recording the temperature by clicking the "Serial Monitor" button.

5. At around 30 seconds, quickly add all the aqueous NaOH solution to the polystyrene cup, keep stirring the reaction mixture with the temperature sensor.
6. Record the temperature for another 120 seconds.

Part E: Experimental Result and Data Analysis

1. Time of addition of NaOH(aq) solution = _____ second
2. Concentration and volume of HCl(aq) solution used = _____ mol dm^{-3} and ___ cm^3
3. Temperature change = _____ – _____ = _____ ^oC
4. Calculate the enthalpy change of the neutralisation (per mole of water formed) based on the experimental data.

_____

_____

_____

_____

_____

_____

_____

Part F: Question

1. List some possible reasons for the difference between the experimental enthalpy change and the theoretical one (other than the apparatus errors mentioned above).

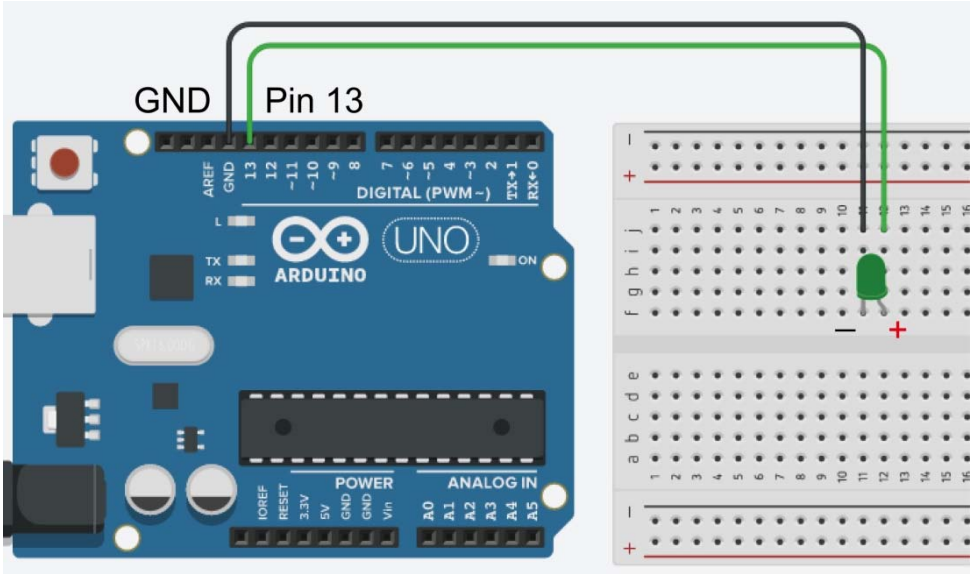
2. Explain why the temperature of the reaction mixture varies?

// Example 1 Flashing LED

```
int pinLED = 13;                    //Store the pin number of the LED

void setup() {
  pinMode(pinLED, OUTPUT);          //Set pin 13 as power output
}

void loop() {
  digitalWrite(pinLED, HIGH);       //Give power to pin 13, turn on the LED
  delay(1000);                      //Wait for 1 second (1000 ms)
  digitalWrite(pinLED, LOW);        //Remove power from pin 13, turn off the LED
  delay(500);                       //Wait for 0.5 second (500 ms)
}
```

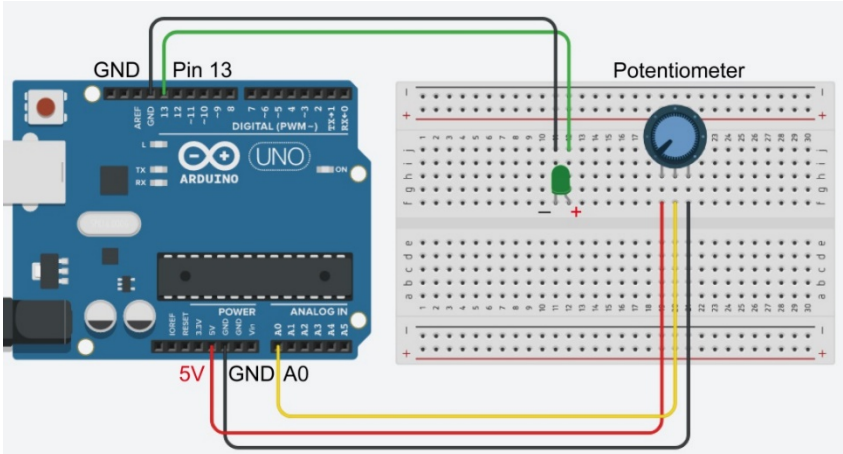


// Example 2 Reading Voltage from a Potentiometer

```
int pinLED = 13;                    //Store the pin number of the LED
int pinPOT = A0;                    //Store the pin number of the potentiometer (POT)
int sensorValue = 0;                //Declare a variable to store POT's reading

void setup() {
  Serial.begin(9600);               //Switch on the "Serial Monitor" for display
  pinMode(pinPOT, INPUT);           //Set pin A0 as input, get signal from potentiometer
  pinMode(pinLED, OUTPUT);          //Set pin 13 as power output
}

void loop() {
  sensorValue = analogRead(pinPOT); //Get reading from A0 pin
  Serial.println(sensorValue);      //Print out the reading on Serial Monitor
  if(sensorValue >= 512) {          //If the reading is at or above 512, turn on the LED
    digitalWrite(pinLED, HIGH);
  }
  else{                             //The reading is below 512, turn off the LED
    digitalWrite(pinLED, LOW);
  }
}
```

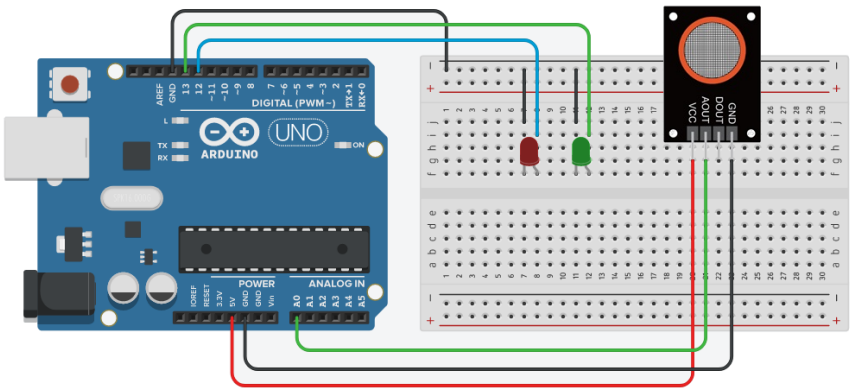


// Example 3 Detecting Alcohol

```
int pinLED1 = 12;                    //Store the pin number of the Red LED
int pinLED2 = 13;                    //Store the pin number of the Green LED
int pinPOT = A0;                     //Store the pin number of the sensor
int sensorValue = 0;                 //Declare a variable to store Sensor's reading

void setup() {
  Serial.begin(9600);                //Switch on the "Serial Monitor" for display
  pinMode(pinPOT, INPUT);            //Set pin A0 as input
  pinMode(pinLED1, OUTPUT);          //Set pin 12 as power output
  pinMode(pinLED2, OUTPUT);          //Set pin 13 as power output
}

void loop() {
  sensorValue = analogRead(pinPOT);  //Get reading from A0 pin
  Serial.println(sensorValue);       //Print out the reading on Serial Monitor
  if(sensorValue >= 128) {           //If the reading is at or above 128, turn on the Red LED
    digitalWrite(pinLED1, HIGH);
    digitalWrite(pinLED2, LOW);
  }
  else {                             //If the reading is below 128, turn on the Green LED
    digitalWrite(pinLED2, HIGH);
    digitalWrite(pinLED1, LOW);
  }
}
```



// Example 4 Enthalpy of Neutralisation

```
#include <OneWire.h>                    //This is the required library of the thermometer
#include <DallasTemperature.h>          //This is the required library of the thermometer
#define ONE_WIRE_BUS 2                  //Set Digital pin 2 as the data pot of thermometer
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);
long newtime = 0;                       //Declare a variable to store current time
long oldtime = 0;                       //Declare a variable to store start time

void setup() {
   Serial.begin(9600);                  //Switch on the “Serial Monitor” for display
   Serial.println("Time,  Temp");       //Print Time, Temp in “Serial monitor”
}
void loop() {
   while(newtime<=150000){              //carry out the experiment for 150 seconds
      oldtime = millis();               //Set the start time
      sensors.requestTemperatures();    //Request temperature from thermometer
      newtime=millis();                 //Update the current time
      Serial.print(newtime/1000);       //Print the running time in second
      Serial.print("   ,  ");
      Serial.println(sensors.getTempCByIndex(0));   //Print the temperature
      while(newtime - oldtime < 1000){              //Set sample rate = 1s
         newtime = millis();
      }
   }
}
```