

GE1354 Introduction to Electronic Design

Lab 1: Learning the Basic Skills

Objectives:

- To practice soldering
- To get familiar with MakeCode and practice using digital write
- To practice using DMM for circuit measurement

IMPORTANT NOTE:

- You have to complete the worksheet during the Lab session. **Make sure that you bring a hardcopy of the worksheet when you attend the Lab.**
- At the end of the Lab, you should also **submit the worksheet and your soldered RGB-R module (Task 1) to the technician-in-charge.** No late submission will be accepted.
- You also have to **give a demonstration** of your work (Task 2) to the tutor during the lab session.
- **It is better for you to bring your own notebook when attending the lab.**

1. Soldering Skill

You need to assemble a module (referred as RGB-R module) by yourself which will be used in this Lab and later mini-projects. Please watch a video for the demonstration of soldering. Figure 1 shows the components and the top view of the module.

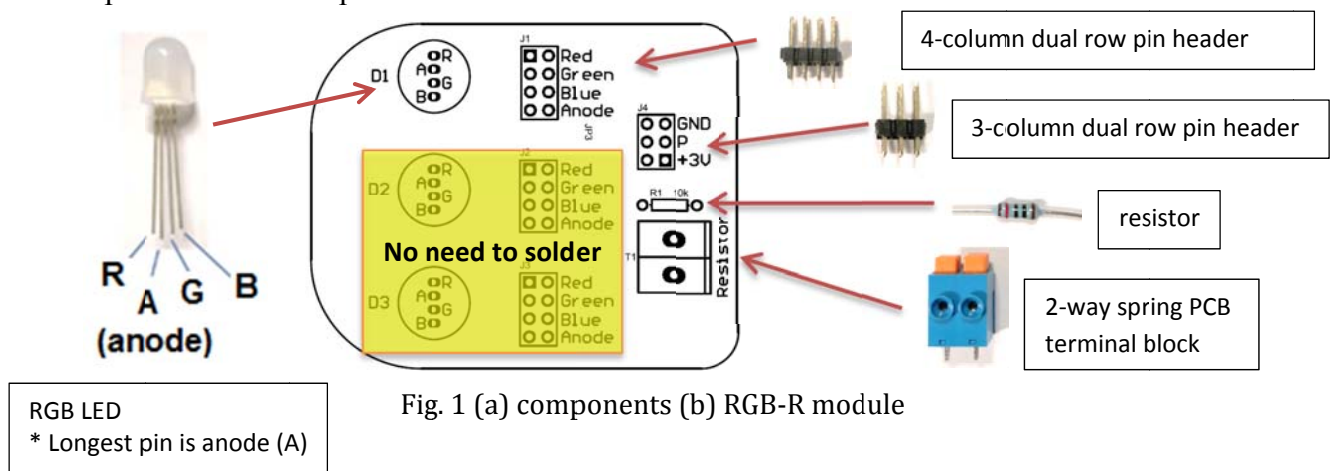


Fig. 1 (a) components (b) RGB-R module

Task 1: Complete the assembly of the RGB-R module

1.1 RGB LED

In your RGB-R module, there is a RGB LED (see Fig. 1). This component includes three light emitting elements (R: Red, G: Green, B: Blue) in a single package. They will emit Red, Green and Blue color, respectively. It is also referred as common-anode because the anodes of these 3 elements are connected together internally. **(Note: The longest pin is the common anode pin)**

You can control the color of RGB LED by providing voltages to the pins. For example, Fig. 2 shows a way to turn on the RGB-LED as red color. The current I flows from 3V to the ground, turning on the R-LED. Similar way is applied for other colors or a combination of them. It is remarked that resistor R_0 is to limit the current. If the voltage source is current limited (e.g. using I/O pins from micro:bit), R_0 may not be necessary.

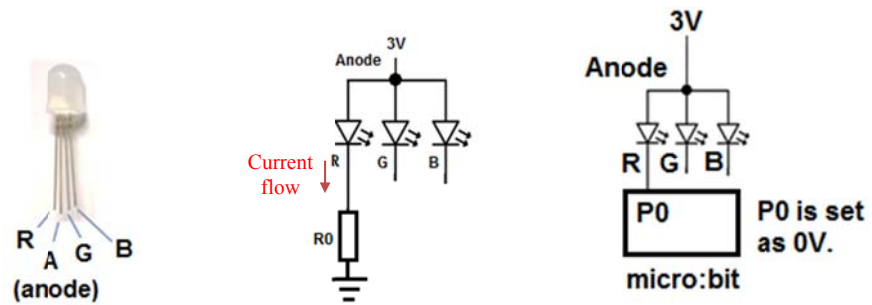


Fig. 2. (a) Common anode RGB LED (b) and (c) Connection to make RGB LED emitting red light.

The RGB LED can emit many colors. The observed color is the result from the emitted light. For example, if the RGB LED emitted Red and Blue at the same time, we observe the Magenta color (see Fig. 3a). Indeed, if we can control brightness of each R-, G-, B-LED, many colors can be observed (see Fig. 3b).

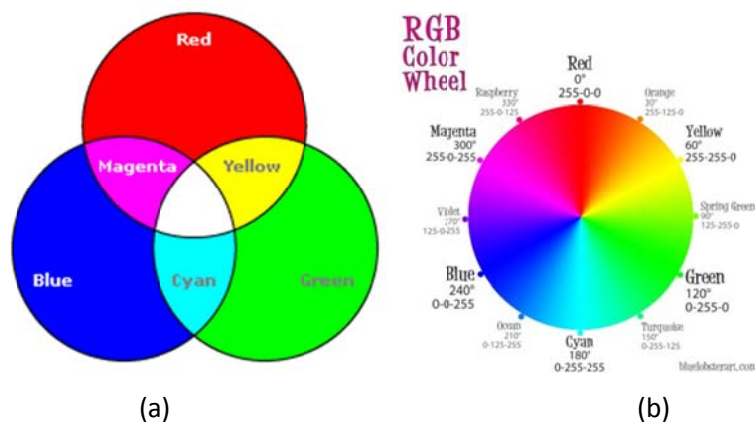


Fig. 3. (a) RGB Color and their combinations (b) Color Wheel

1.2 Resistor

Another part of your RGB-R module has a resistor. It will be used in later mini-projects, and you will measure its value with the DMM in this Lab.

To read the nominal value of a resistor, the following color code can be referred to. However, it should be emphasized that the actual value of your resistor may not be the same as the nominal value.

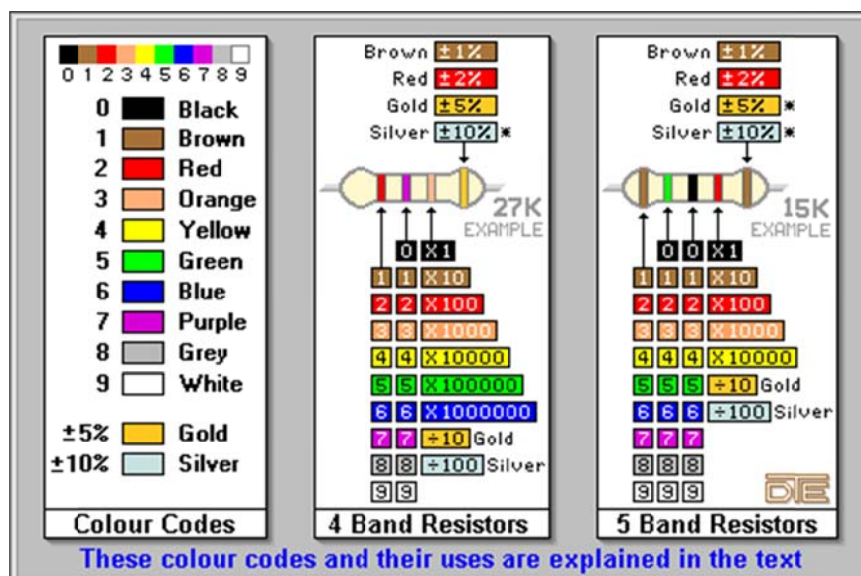


Fig. 4. Color code for resistor

2. Get familiar with the web programmer **MakeCode**

2.1 Background

Micro:bit is a tiny programmable electronic system (see Fig. 5). It consists of sensors, 5x5 LED screen, two buttons, and some connection pins (for external connection), etc. By programming the micro:bit, you can read the sensors, control the LED screens and interface with other modules connected to the pins. (More details can be referred to Lecture Note Unit 1)

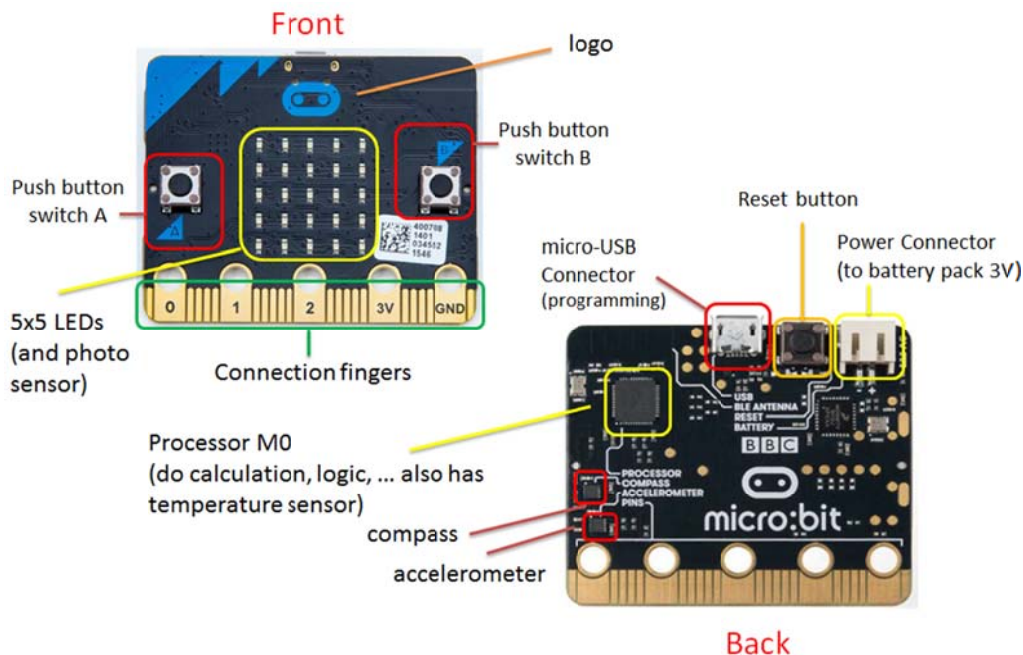
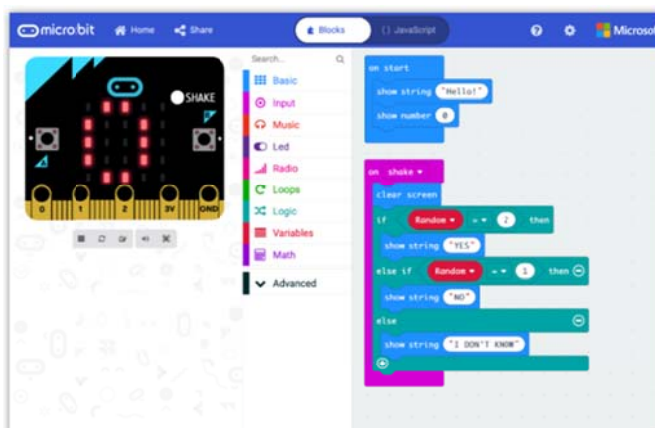


Fig. 5. Elements of a micro:bit

Micro:bit can be programmed by several programming languages, including JavaScript Blocks, Arduino and micropython. In this course, we only use JavaScript Block.

Self-study: Visit <https://microbit.org/code/> and read the lessons



MakeCode Editor

The MakeCode editor provided by Microsoft makes it easy to program your micro:bit with blocks and JavaScript.

We have recently updated the editor, and the previous version is still available for anyone that needs it. If you have any issues accessing the editor, check that it isn't blocked in your school.

Let's Code

Reference

Lessons

2.2 Using the web programmer: **MakeCode**

Firstly, let's follow the below steps and get familiar with the procedures.

1. Connect your micro:bit to your PC/notebook with the USB cable. (Note: The PC/notebook may take some times to get the drivers. Moreover, you may be asked to do some actions to calibrate the micro:bit if it has never been used before.)
2. Go to <http://makecode.microbit.org>, then you will see a page as shown in Fig. 6.

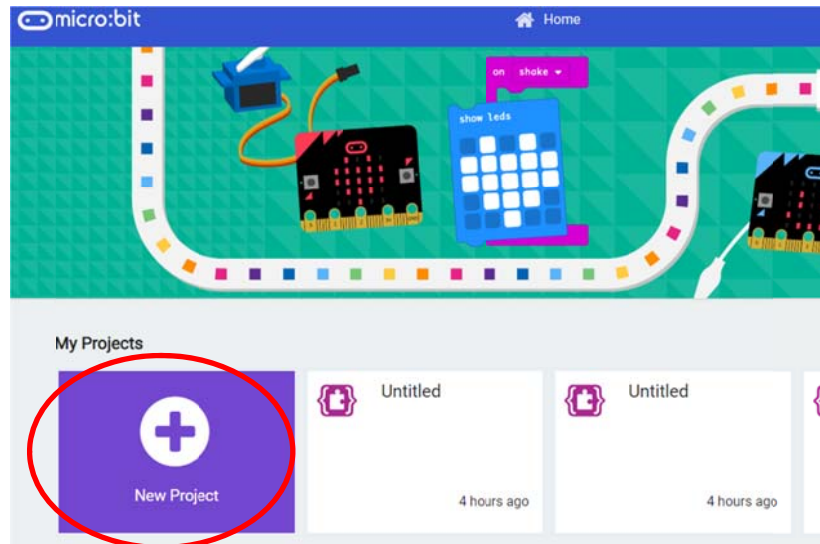


Fig. 6. <http://makecode.microbit.org>

3. Select “New Project”, you will see the following user screen (Fig. 7), and can then start programming.

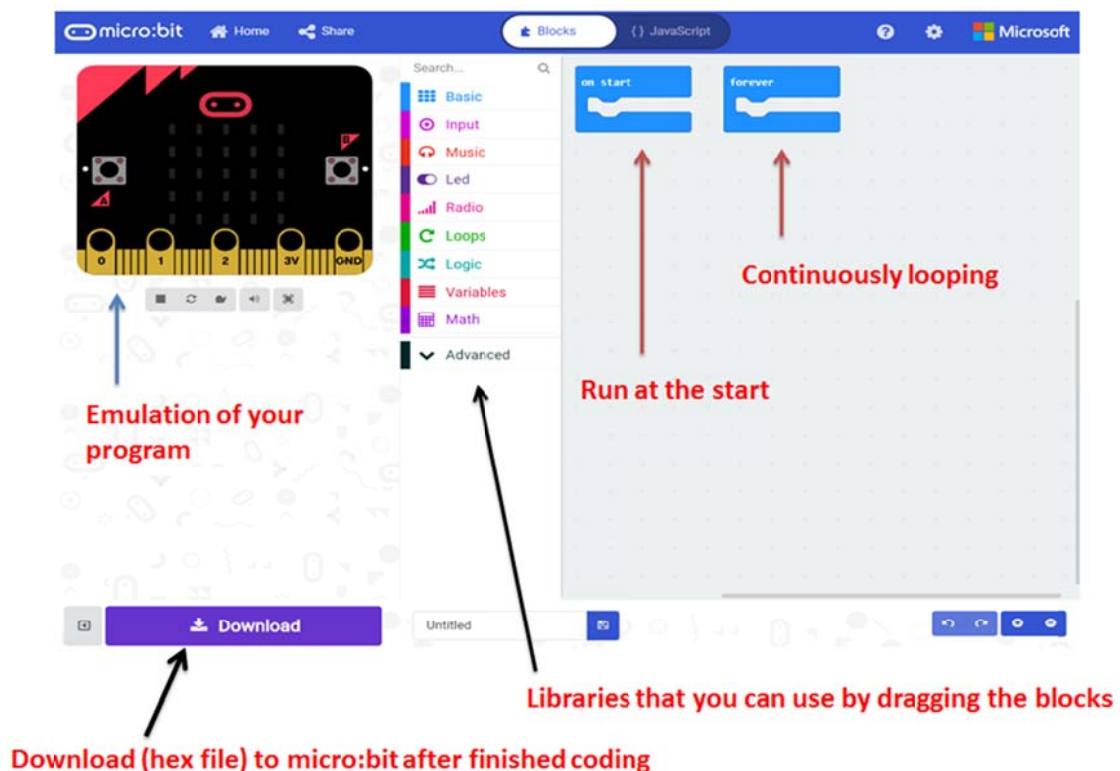


Fig. 7. User Screen in **MakeCode**

4. Let's try first to turn on the LED positioned at (0,0). The Block command is



This command is in Led library  **Led** . You simply open the library and then select the block. An example is shown in Fig. 8.

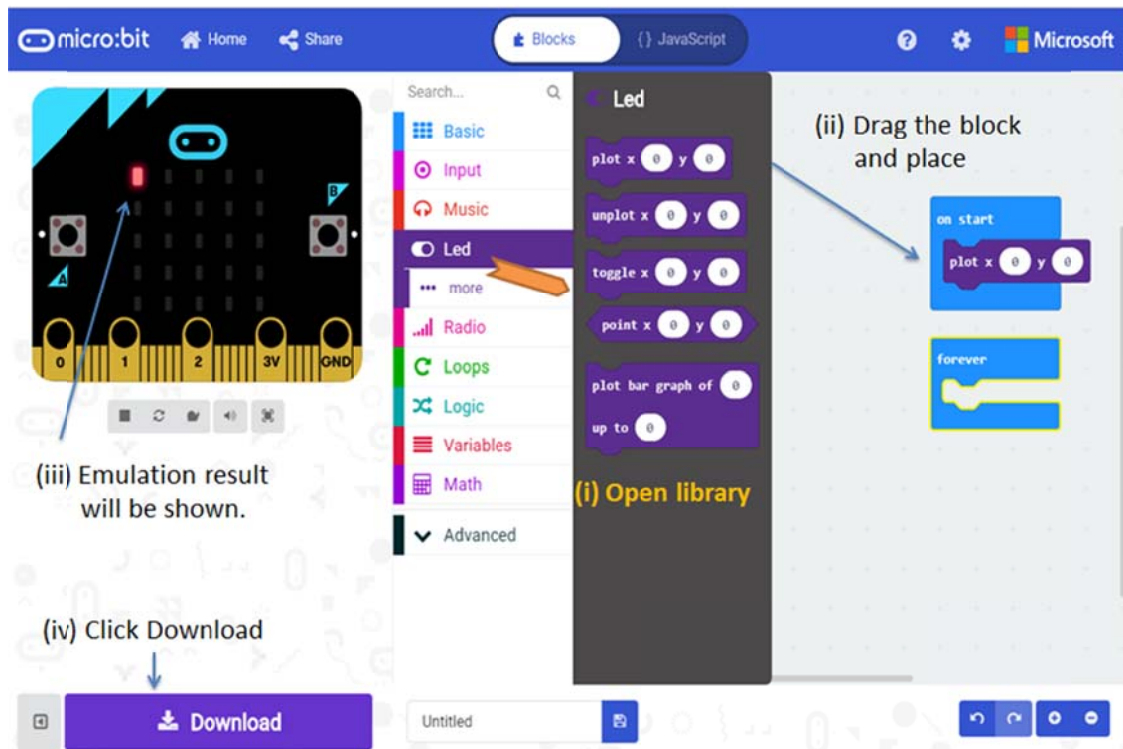


Fig. 8. Programming micro:bit using **MakeCode**

5. After complete your program (Steps (i) and (ii) in Fig. 8), you will see the emulation of your program (Step (iii)). If it is ok, then you can click the download.
6. Your program will be converted (the process is called compilation), and a file named **microbit-Untitled.hex** will be created and downloaded to the PC. [Note: if the file **microbit-Untitled.hex** already exists, the file name will become **microbit-Untitled(1).hex** , and etc.]
7. Identify the location of the hex file, and then drag the file to the MICROBIT drive (Note: Remember to connect micro:bit to PC). An example is shown in Fig. 9 (Note: The drive ID is F: in this example, but it may not be the same in your case.)
8. After complete the download, your designed program has been flashed into your micro:bit, and you should see the LED (0,0) being turned on.

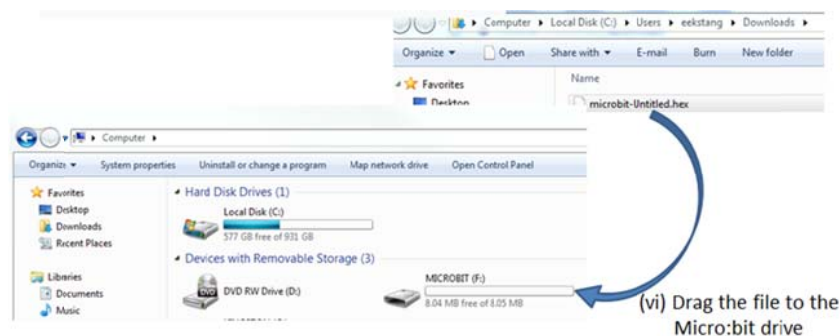


Fig. 9. Flashing the hex file to micro:bit

2.3 Programming Tasks

At this stage, you should know the procedure to program your micro:bit. If not, please ask tutors for help.

Your design task is to turn on the LED of your RGB+R module. Firstly, use the jumping wires to connect the following circuit.

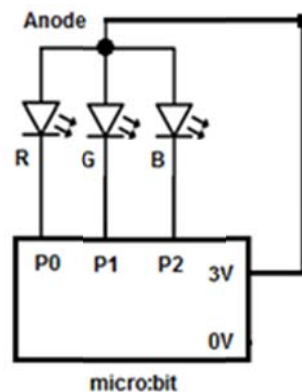


Figure 10. Experimental circuit to connect RGB-LED to micro:bit

Remark: When doing programming, you can connect the micro:bit to the PC/notebook using the USB cable. When doing demo, connect a battery to micro:bit via the power connector. (See Fig. 11).

By referring to the following block commands,

	Set the specified pin as either 0 or 1 (output 0V or Vdd, respectively)
	Pause for a given time (in millisecond)

Finish the following tasks:

Task 2.1: Turn on the RGB LED to show red color.

Task 2.2: Turn on the RGB LED to show three colors in sequence

(eg. RED->GREEN->BLUE, then off; each color turns on 1 second)

Task 2.3: Turn on the RGB LED to show four colors in sequence repetitively

(eg. RED->GREEN->YELLOW->BLUE->RED->GREEN->...)

Power up your micro:bit with battery, and show a demonstration of your results to tutors.

3. Using Digital multimeter (DMM)

Digital multimeter (DMM) is a handheld equipment to measure some basic electrical values, such as voltage (unit in volt or V), current (unit in ampere or A) and resistance (unit in ohm or Ω).

3.1. Measure voltage using DMM

When we use DMM to measure voltage, it actually measures the voltage difference between the two probes (red and black).

READING = voltage at the RED probe – voltage at the BLACK probe

When we say the voltage of a pin (or of a point in a system/circuit), it means its voltage as referred to the ground (GND) of the entire system.

For example, to measure the “3V” pin, we firstly select the “V” menu, then touch the “0V” pin (it is GND) and the “3V” pin (see Fig. 11(c)) with the black and red probes.

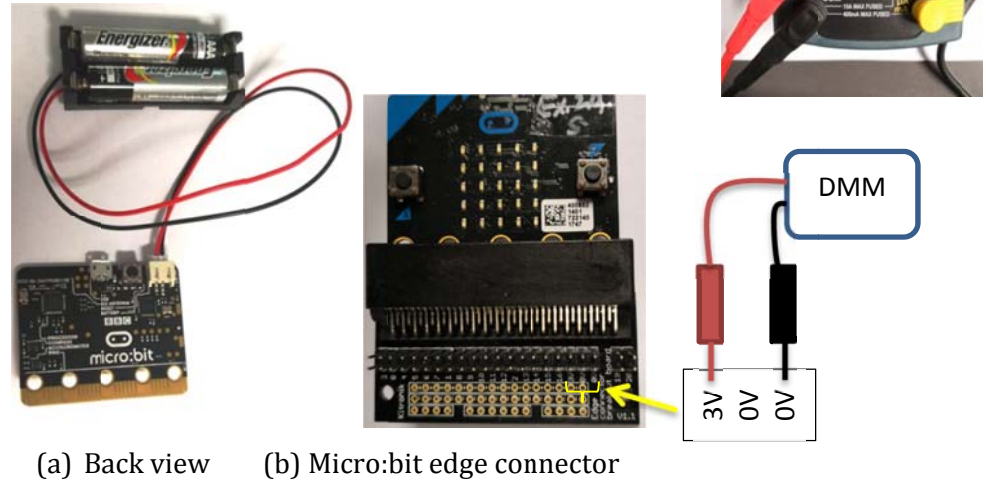


Fig. 11. Connecting a battery pack to micro:bit (a) back view (b) Measuring 3V pin of micro:bit

3.2. Measure resistance of a component using DMM

Resistor is a kind of electronic components and a sample is shown in Fig. 12(a). You have also soldered one in your module.

When we say, the resistance of a component, we mean the resistance between the two ends of that component. For example, if we want to know the value of the resistor in Fig. 12(a), we can measure it as shown in Fig. 12(b). The detailed procedures are given as follows:

1. Select the “ Ω ” menu (The unit of resistance is called Ohm.)
2. Touch the two ends of the resistor with the black and red probes
3. Get the reading as shown in the display



Fig. 12 (a) A resistor (b) Measuring a resistor with DMM

Important Note: Don't touch the ends or probes with your fingers. It will affect the reading.

3.3 Measuring Tasks:

Task 3.1: Measure the voltage of P0, P1, P2 and the “3V” pins when RGB LED is in RED.

Task 3.2: Measure the value of the resistor that you soldered in your RGB-R module.

(Write down the answers in the Worksheet 1)

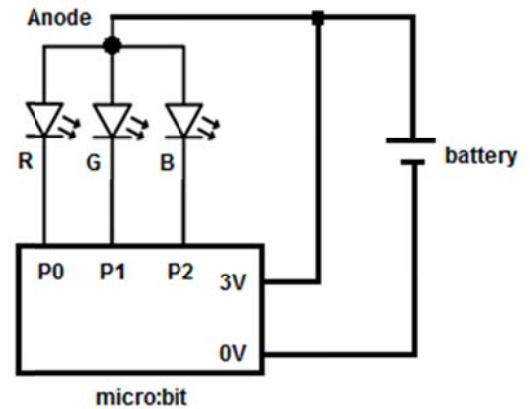
This worksheet must be completed in the Lab. Submit **the worksheet** together with your **soldered module** (stick the label with your SID at the back) to the technician-in-charge. Late submission is not accepted.

Worksheet 1: Name: _____ (Student ID: _____)

- a) Referring to Fig. 10 (also given on the right), measure the voltage of P0, P1, P2 and the “3V” pin when RGB LED is turned on as red
(Give the unit) of your answers.

3V pin = _____
P0 = _____
P1 = _____
P2 = _____

[Note: Your answers must be based on measurements]



Power up the RGB-R board with battery

- b) Draw arrow(s) in the circuit diagram to show the direction of the current flowing through the 3 LEDs of the RGB-R board when yellow color is observed. (If no current, draw a cross “X”)
c) By using digital write to turn on RGB LED, how many different colors you can get? _____
d) What are the nominal and measured values of resistor in your RGB-R module?
Nominal value of R = _____ Ω ; Measured value of R = _____ Ω

- e) Identify the library that each of the following blocks belongs to.



1) **Basic** (this is an example) _____



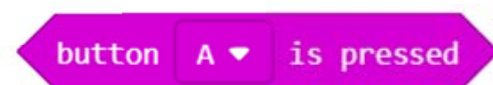
2) _____



3) _____



4) _____



5) _____



6) _____