Computing

Lesson 3: Making Connections Part 1

Physical Computing

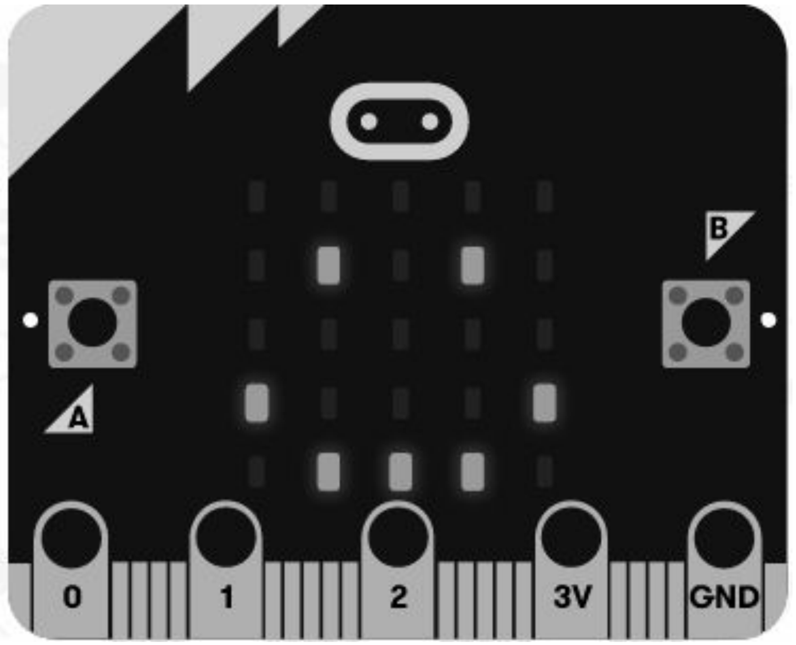
Allen Heard

Materials from the Teach Computing Curriculum created by the National Centre for Computing Education

Task 1 - Connecting through pins

Previously we saw a program that displays a ‘happy face’ when the micro:bit is lying face up, and displays nothing otherwise.

```
from microbit import *
while True:
  if accelerometer.is_gesture("face up"):
    display.show(Image.HAPPY)
  else:
    display.clear()
```



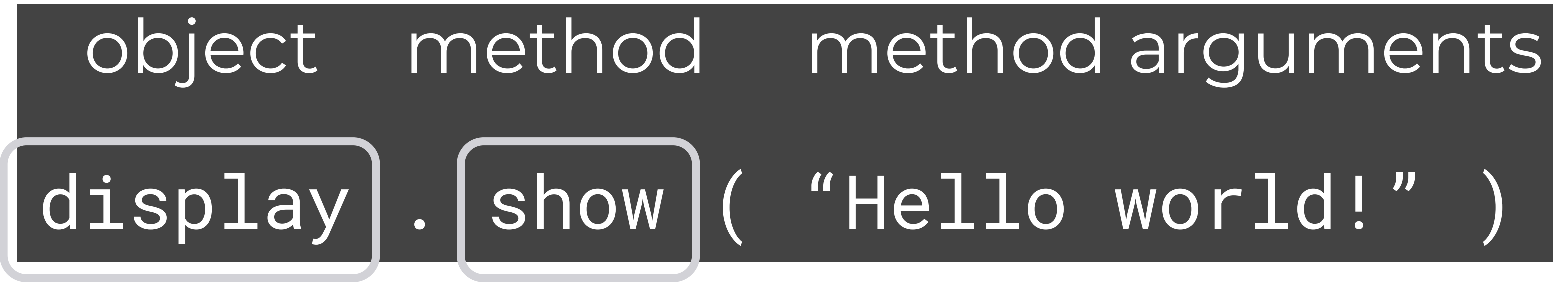
Credit: micro:bit Foundation

Task 1 - Connecting through pins

The program below computes and displays a decimal number that depends on which of the buttons are pressed.

```
from microbit import *
while True:
    decimal = 0
    if button_a.is_pressed():
        decimal = decimal + 2
    if button_b.is_pressed():
        decimal = decimal + 1
    display.show(decimal)
```

- In these examples, `display`, `accelerometer`, and `button_a` are **objects** that represent the micro:bit's components.
- **Methods** like show, `is_gesture` and `is_pressed` correspond to possible actions that can be performed on these components.



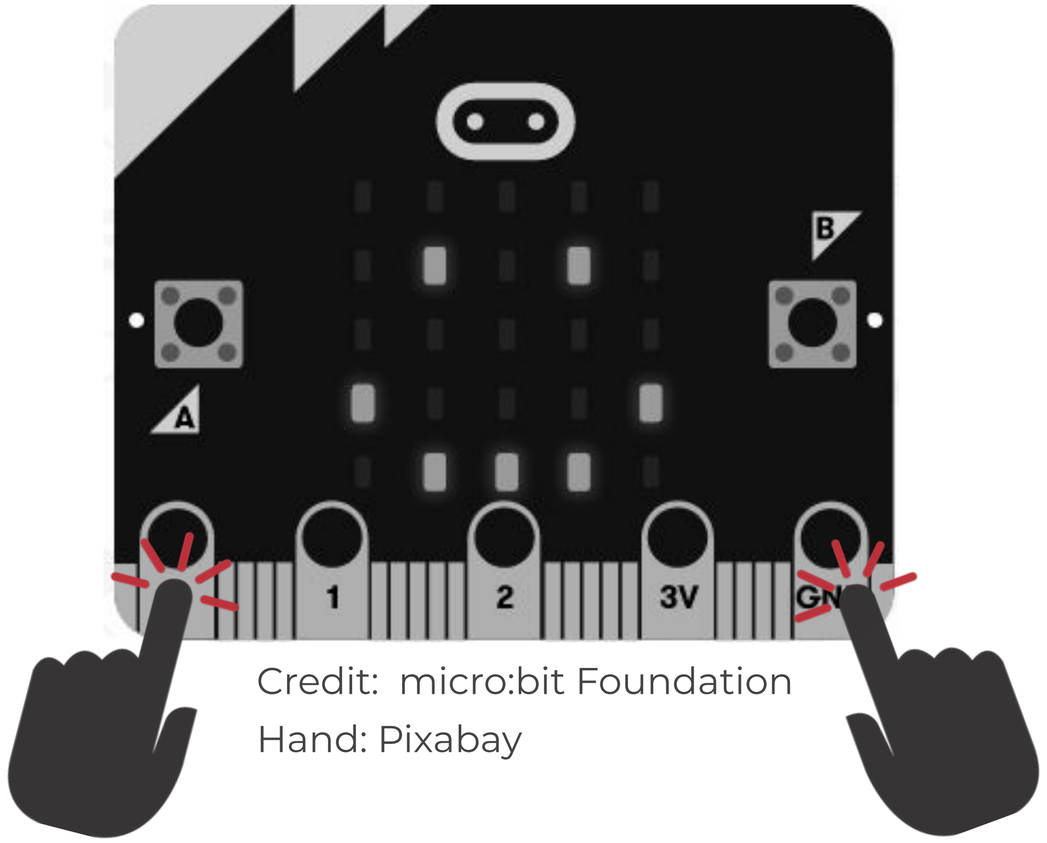
```
display . show ( “Hello world!” )
```

Task 1 - Connecting through pins - Touchy

Copy the program below in your development environment. It is similar to the example, but instead of detecting a gesture, the program now detects if `pin0` is being touched.

```
from microbit import *
while True:
  if pin0.is_touched():
    display.show(Image.HAPPY)
  else:
    display.show(Image.SAD)
```

Flash the program to your micro:bit to run it and touch `pin0` with one hand and `GND` with the other to test it.

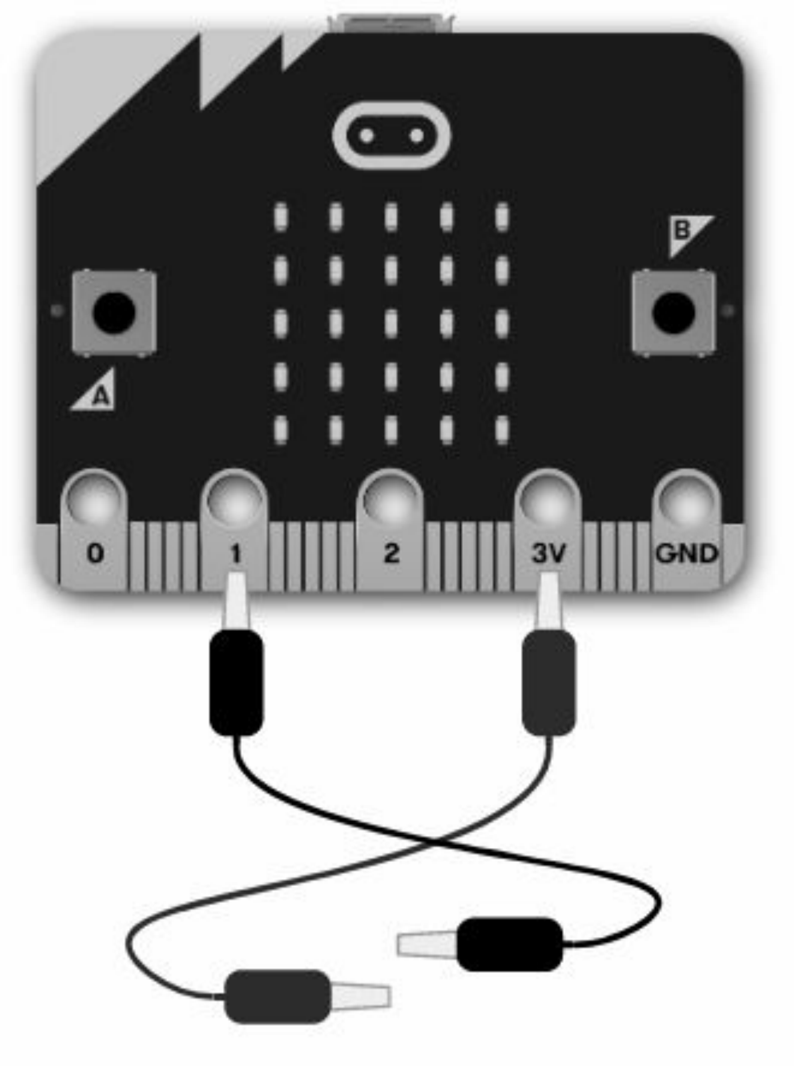


Credit: micro:bit Foundation
Hand: Pixabay

Task 2 - Makeshift switch - part 1

Note: you will need crocodile clips for this task.

- Connect a wire to the “power” pin (labelled 3V).
- Connect another wire to pin 1. This will be the **input pin**, i.e. the pin that the program will be checking to see if the switch is closed.



Task 2 - Makeshift switch - part 2

Copy the program below in your development environment:

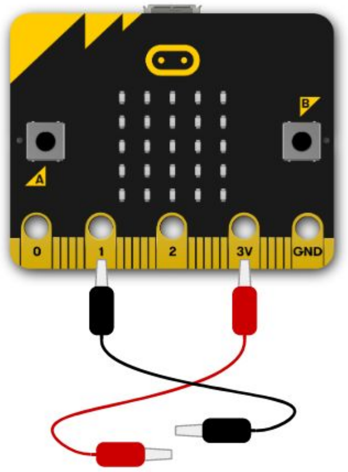
```
from microbit import *
while True:
  in_value = pin1.read_digital()
  display.show(in_value)
```

Flash the program to your micro:bit, to run it and test it.

Questions.

What is the value displayed when the wires are not connected (switch is off)? Answer:

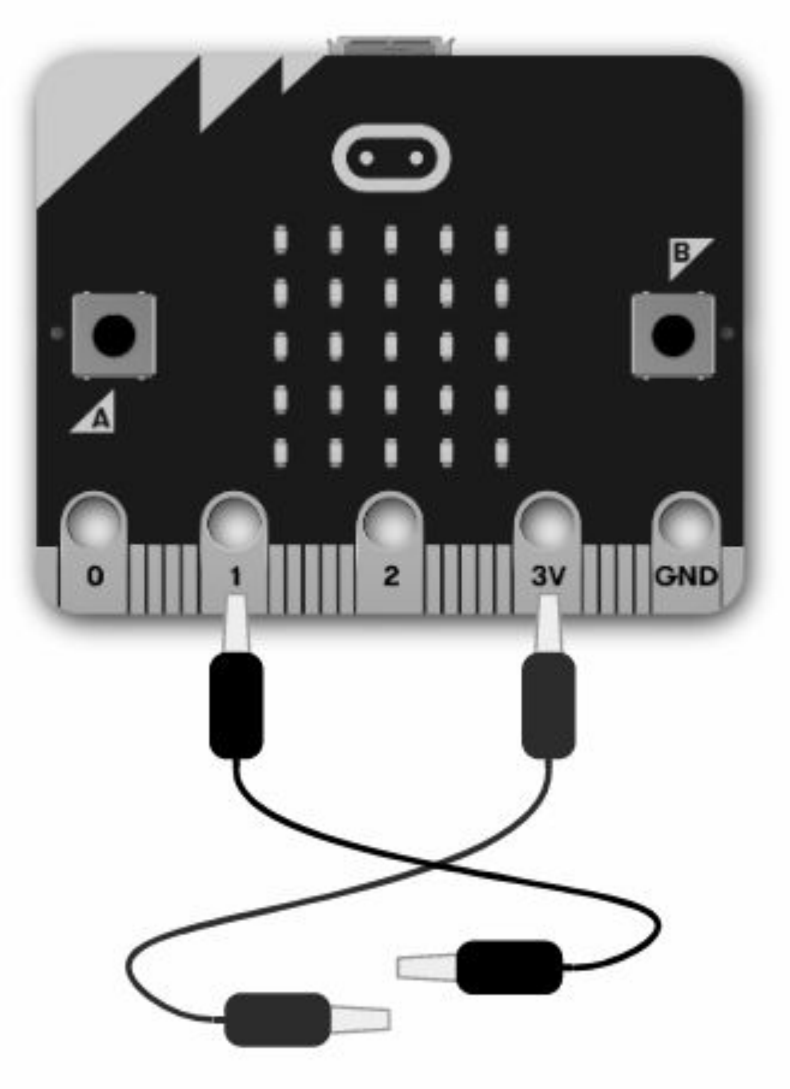
What is the value displayed when the wires are connected (switch is on)? Answer:



Task 2 - Makeshift switch - part 3

Extend your program so that a different image is displayed on the 5x5 LED matrix, depending on whether or not the switch is ‘on’ or ‘off’ (e.g. `Image.YES` and `Image.NO`).

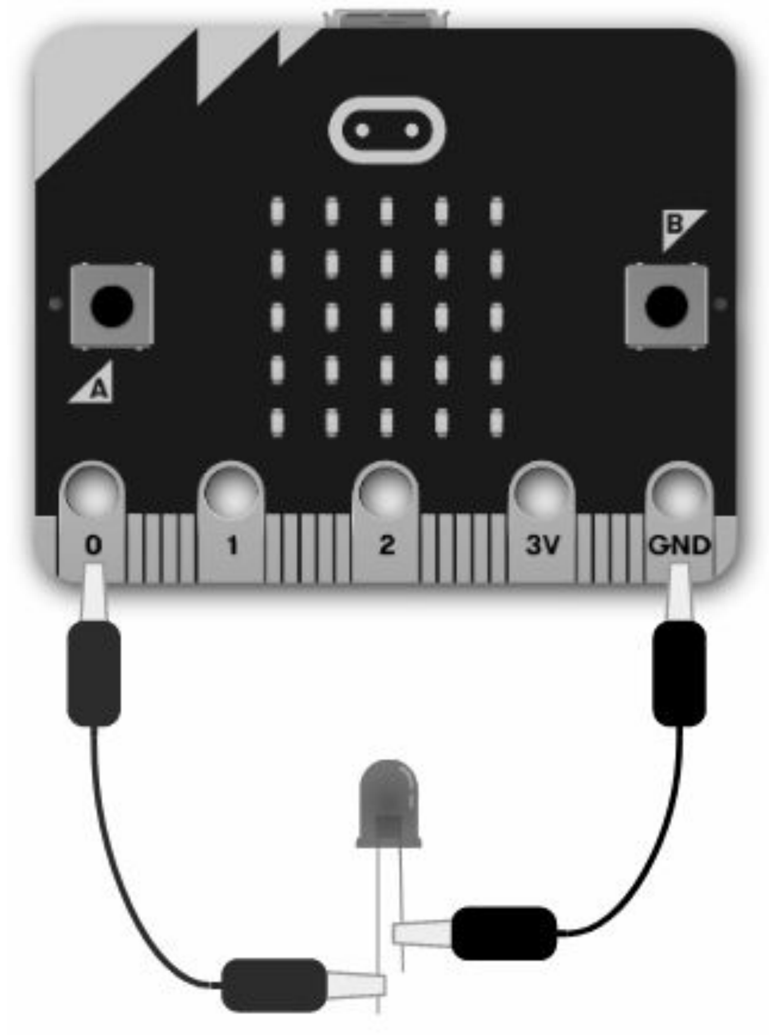
Flash the program to your micro:bit, to run it and test it.



Task 3 - Light it up

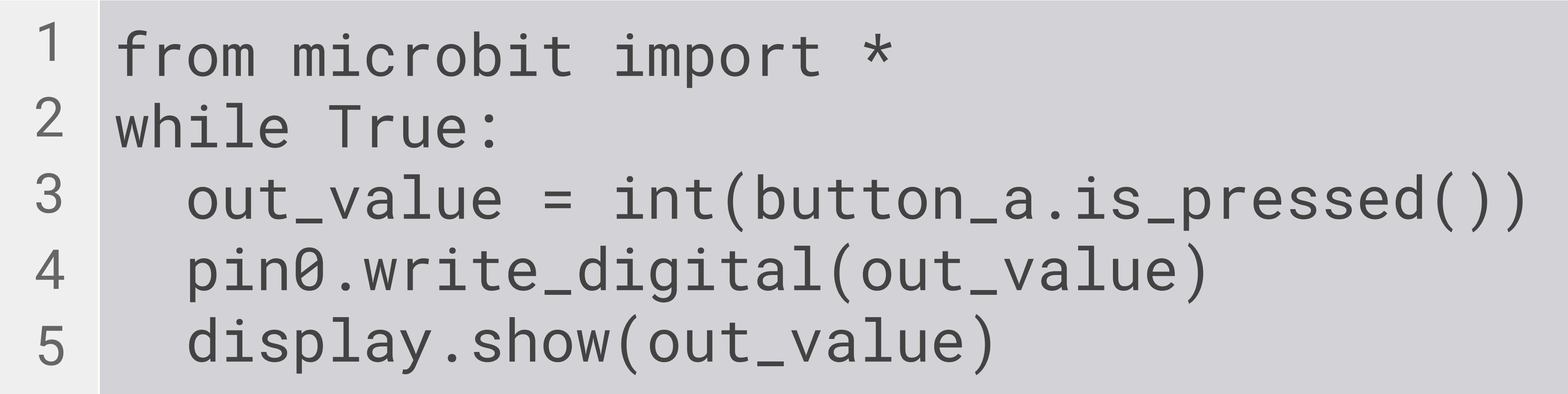
Note: you will need crocodile clips and an LED for this task.

- Connect `pin 0` to the long leg of the LED (the anode). This will be the **output pin**, i.e. the pin that the program will use to control the LED.
- Connect the “ground” pin (`GND`) to the short leg of the LED (the cathode).



Task 3 - Light it up

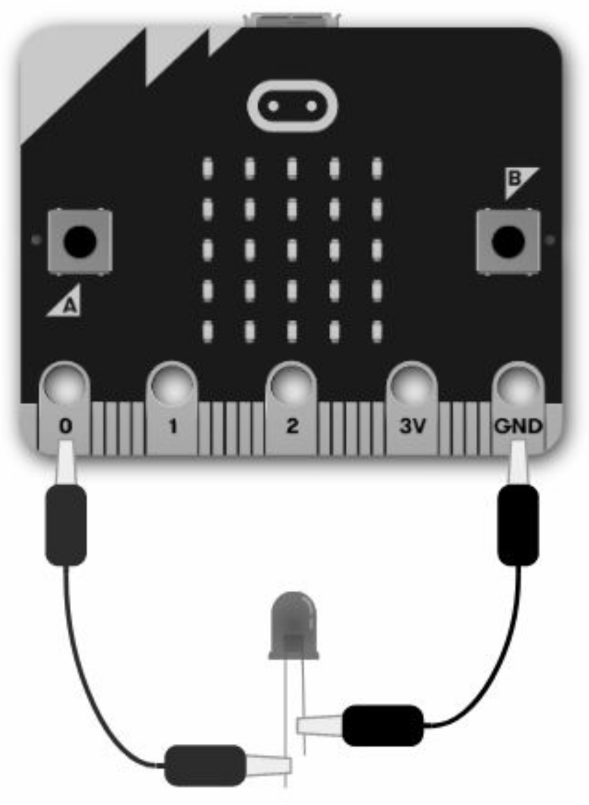
Copy the program below into your development environment.



```
from microbit import *
while True:
  out_value = int(button_a.is_pressed())
  pin0.write_digital(out_value)
  display.show(out_value)
```

Flash the program to your micro:bit, to run it and test it.

- **Try to figure out how the program works.**
- **What do you need to do in order for the LED to light up?**


B
A
0
1
2
3V
GND