Worksheet 2 - Programming on the Pi

This lesson assumes a basic familiarity with programming in SCRATCH
This lesson assumes the Raspberry Pi has been configured with GPIO and Pibrella add-ons.

Learning objectives

- You will learn how to connect your Pi
- You will learn how to turn outputs on and off from the raspberry pi
- You will learn how to read sensor inputs on the raspberry Pi

You will need

- Raspberry Pi
- Pibrella Interface board
- Mouse, Keyboard, Monitor
- Light sensor built in lesson 1

How to connect your Raspberry Pi

Plug the pibrella board onto the GPIO pins on the top of the Pi. When plugged in, it should look like this:

Connect the Micro-USB power cable to the TOP board.

Plug in the mouse to the first USB socket.

Plug in the keyboard to the second USB socket.

Connect the HDMI cable between the monitor and the Raspberry Pi.

Logging into your Raspberry Pi

Once your Pi has booted up, log in with the username “pi” and the password “raspberry”

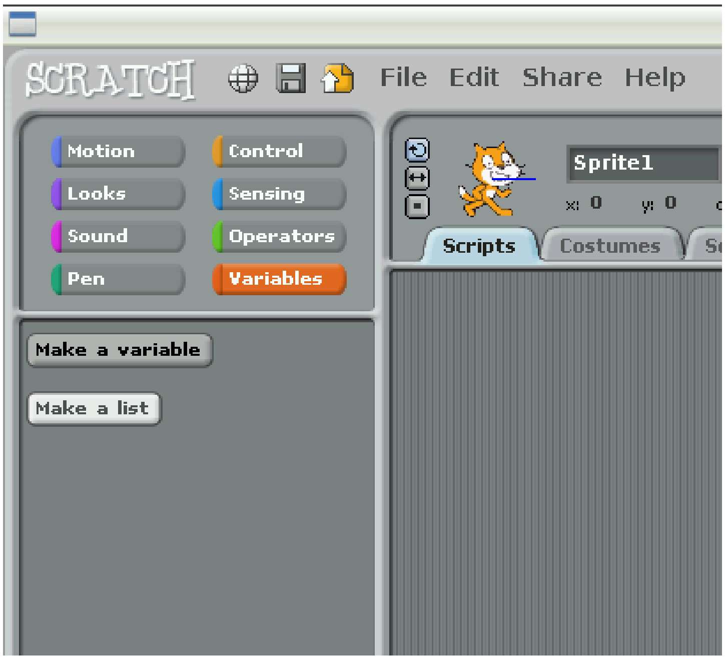
Type the command “startx” to open the desktop

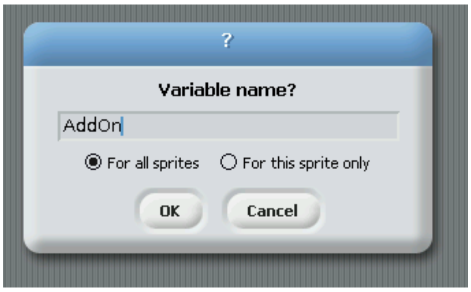
Once the desktop has opened, double click on “SCRATCH GPIO4” to start writing your computer program.

Experiment 1

In this experiment, you'll use your Raspberry Pi to turn an LED on and off.

Create a variable called "AddOn"



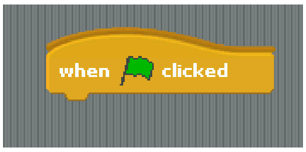

?

Variable name?

AddOn

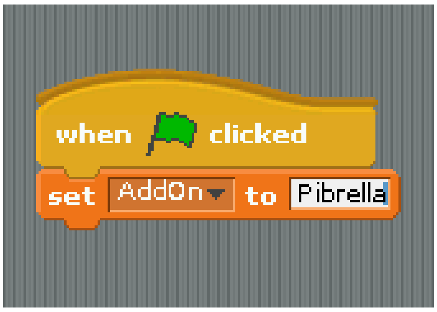
● For all sprites ○ For this sprite only

Create a "when flag clicked" Control.



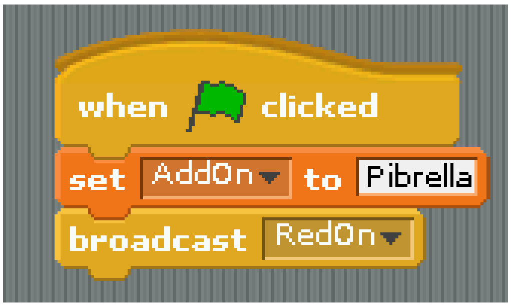
```
when 🟩 clicked
```

Add the AddOn variable and set its value to "Pibrella"



```
when 🟩 clicked
set AddOn ▾ to Pibrella
```

From the Control menu, broadcast the word “RedOn”



```
when ⚑ clicked
set AddOn to Pibrella
broadcast RedOn
```

Press the green flag. Write down what happens

From the Control menu, broadcast the word “RedOff”

Press the green flag. What happens?

Using the “wait 1 secs” block in the control menu, flash your Red LED on and off 10 times.

Write your program out here

Use the commands “RedOn”, “RedOff”, “AmberOn”, “AmberOff”, “GreenOn”, “GreenOff” to turn your LEDs on and off in sequence.

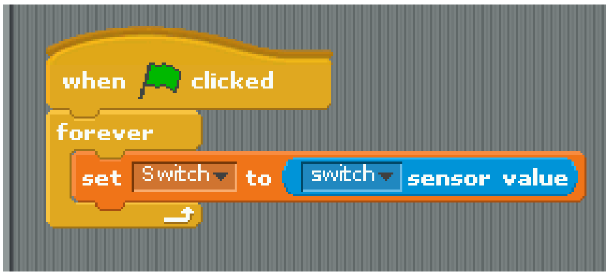
Experiment 2

In this experiment, you'll use your Raspberry Pi to turn an LED on when the button is pressed.

Create a "when flagged clicked" Control

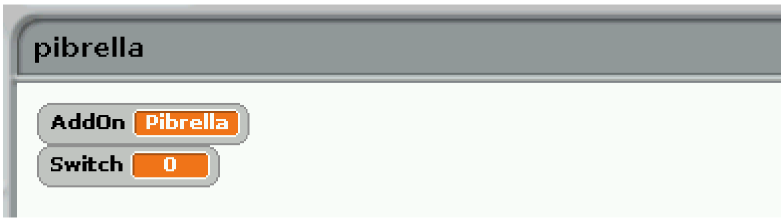
Make a variable called "Switch" in the same way you did in Experiment 1. Set its value to the "switch" sensor value using the "sensor value" block from the "sensing" menu.

Put this code inside a "forever" loop



```
when clicked
forever
    set Switch to switch sensor value
```

Press the green flag. You should see a box in the top right corner of your screen telling you what the current value of your Switch variable is.



pibrella

AddOn Pibrella

Switch 0

Write down what the value of this variable is when the switch is pressed.

Write down what the value of this variable is when the switch is not pressed.

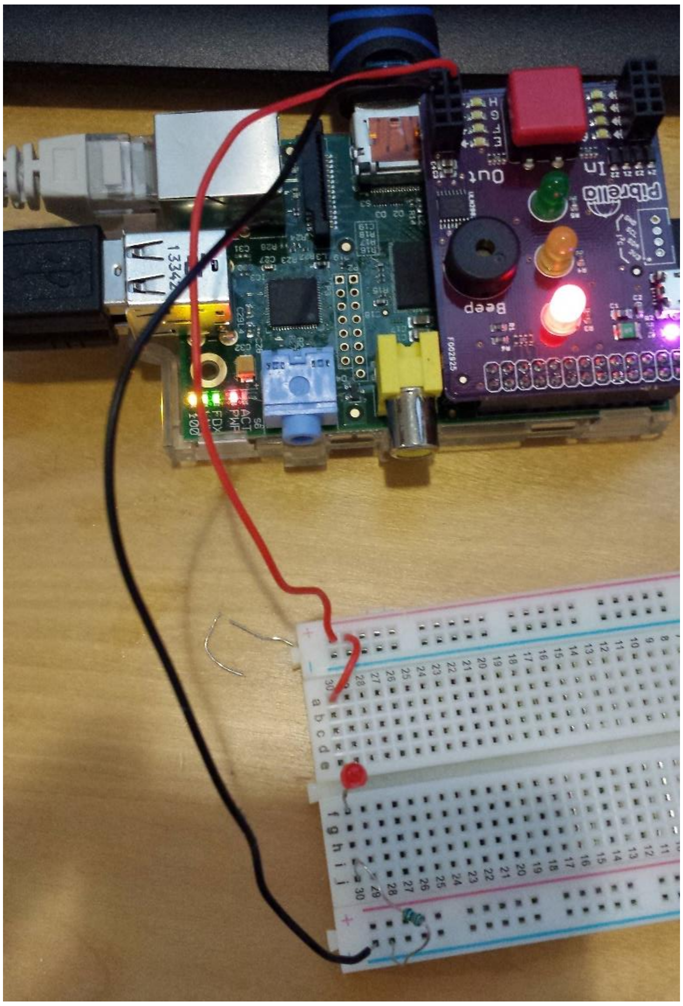
Using the "if" block found in the "control" menu, turn on the Amber LED when the switch is pressed.

Write your program below

Experiment 3

Using the knowledge learned in worksheet 1. Connect your Raspberry Pi to an LED on your breadboard to build the following circuit

Connect your LED to the output H connector on your board. Make sure to get the + and – the correct way round.



Broadcast the message OutputHOn to turn the LED on.

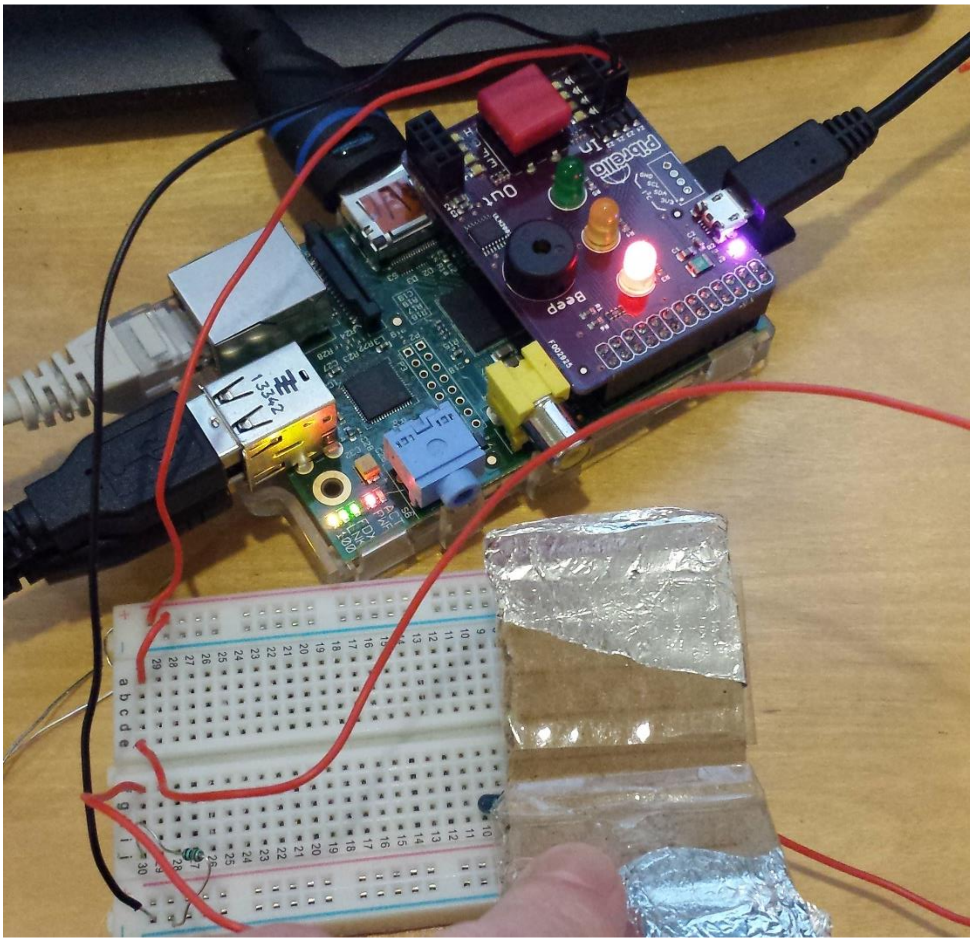
What happens?

What do you think the command to turn the LED off is?

Write a simple program to flash the LED on and off 5 times when the button is pressed.

Experiment 4

Remove the LED from your breadboard. Plug the black and the red wires into InputA and connect the contact sensor you made in worksheet to where the LED was plugged in.



What happens when you press your contact sensor?

In experiment 2, we read the value of the button on the circuit board. Using similar code, read the value from the contact sensor plugged into InputA to turn on and off the green LED.

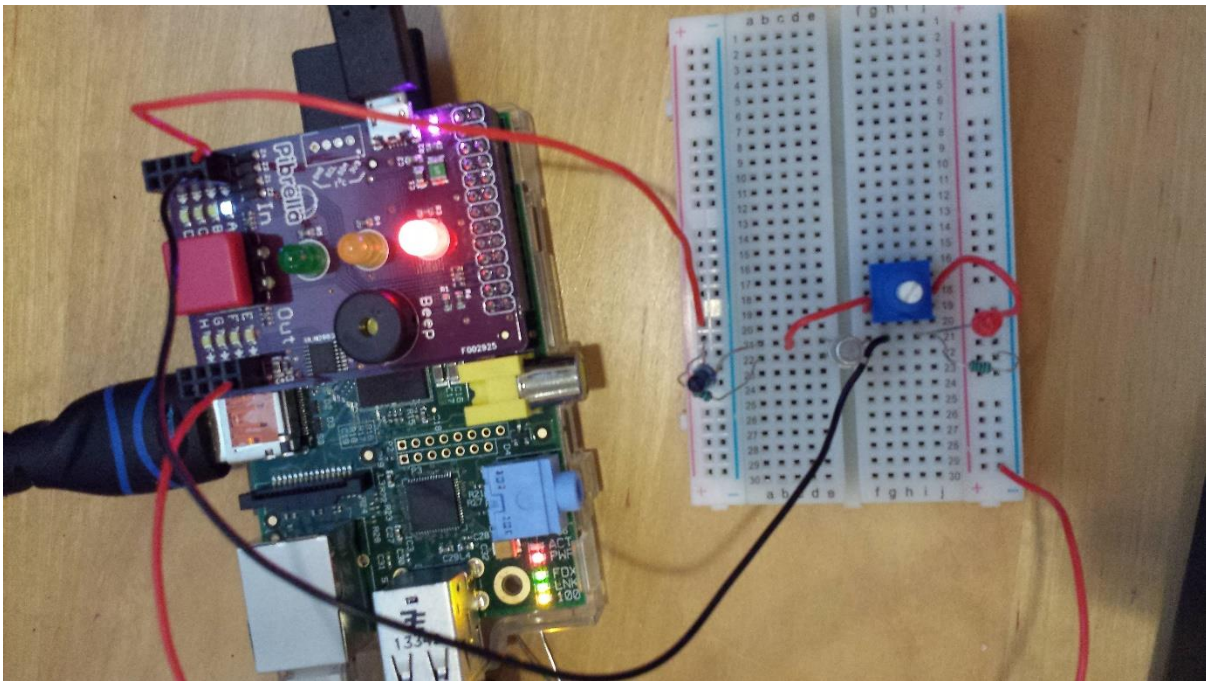
The “sensor value” block can be found in the “sensing” section. The input for InputA is called “InA”.

Experiment 5

Unplug your contact sensor from InputA.

Plug the light sensor you made in lesson 1 into your Rasperry Pi. The light sensor requires 3 wires to be connect.

- Input A +ve – connect to top rail of breadboard
- Input A –ve – connect to emitter of transistor
- Output E –ve – connect to bottom rail of breadboard



Calibrate your light sensor by adjusting the potentiometer so that when the Phototransitor is covered. The value for InputA should switch from 0 to 1 and back when covered.

Run the program from experiment 4. You should now be able to use your Raspberry Pi to turn the LED off and on.

Extension Activity

Write a program on your raspberry pi to turn the count how many times your light sensor turns on and off.

Write your program down below: