CamJam EduKit Robotics – Running the Motors

Project   Running the Motors

Description   You will learn how to make the wheels on your robot turn, and ensure they turn in the same direction.

Equipment Required

For this worksheet you will require:

- The basic robot built in Worksheet One

Making the Wheels Turn

The EduKit Motor Controller Board takes the output of some of the Raspberry Pi's GPIO (General Purpose Input/Output) pins and turns the motors forwards or backwards depending on whether those pins are 'on' or 'off'. The controller board is a type of controller called an 'H Bridge'. Wikipedia has a good article on the H Bridge if you want to learn more about it.

The EduKit Motor Controller Board uses two pins to control the left motor and two pins to control the right motor. GPIO pins 9 and 10 control Motor A (the right-hand motor), and 7 and 8 control Motor B (the left-hand motor).

Models A & B   **Models A+, B+ & Pi2**
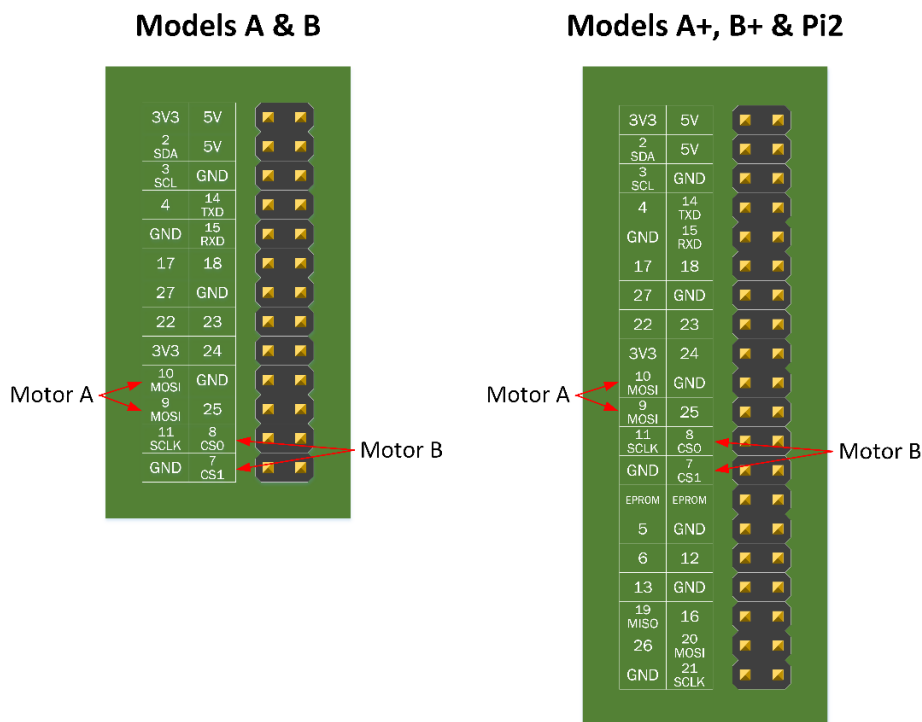
The Raspberry Pi GPIO Pins used by the Controller Board

One pin is used to set the direction of the motor forward by turning that pin 'on'. The other GPIO pin will make the motor turn backwards, again when that pin is turned 'on'. When both pins are off, or both pins are on, the motor will not turn.

For the robot you built in the previous worksheet, you are going to use GPIO pin 10 to make the right hand motor turn forwards, and GPIO pin 9 to make it turn backwards.

The left hand motor uses GPIO pin 8 to turn it forwards, and GPIO pin 7 to turn it backwards.

Code

First, if you have not already done so, create a directory for all the code for this EduKit to go in. On your Raspberry Pi, open a terminal window and type the following:

```
cd ~
mkdir EduKitRobotics
cd EduKitRobotics
```

This will move you to the 'pi' user's home directory, create a new directory called 'EduKitRobotics', and then move into that directory.

Create a new Python script and edit it using the 'nano' editor in the terminal window, by typing:

```
nano 2-motors.py
```

Type in the following code:

Note: Any text after the '#' is a comment, and can be left out if you want. However, it helps to tell you what the code does.

```python
# CamJam EduKit 3 - Robotics
# Worksheet 2 - Motor Test Code

import RPi.GPIO as GPIO # Import the GPIO Library
import time # Import the Time library

# Set the GPIO modes
GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)

# Set the GPIO Pin mode
GPIO.setup(7, GPIO.OUT)
GPIO.setup(8, GPIO.OUT)
GPIO.setup(9, GPIO.OUT)
GPIO.setup(10, GPIO.OUT)

# Turn all motors off
GPIO.output(7, 0)
GPIO.output(8, 0)
GPIO.output(9, 0)
GPIO.output(10, 0)

# Turn the right motor forwards
GPIO.output(9, 0)
GPIO.output(10, 1)

# Turn the left motor forwards
GPIO.output(7, 0)
GPIO.output(8, 1)

# Wait for 1 second
time.sleep(1)

# Reset the GPIO pins (turns off motors too)
GPIO.cleanup()
```

Once complete use "Ctrl + x" then "y" then "enter" to save the file.

Running the Code

> Note: Before running the code, turn the battery pack on.
>
> Note: Put your robot onto a stand so that the motors are off the ground – at the moment, unless you are using a battery for your Pi, it is tethered to your power supply. Also, you don't want your robot running off your desk!

To run the code, type the following into the terminal window:

```
python3 2-motors.py
```

You will notice that Python3 is being used; this is the most up-to-date version of Python and considered by the Raspberry Pi Foundation as the standard that everyone should use.

If you find that the code does not run correctly, there may be an error in the code you have typed. You can re-edit the code by using the nano editor, typing `nano 2-motors.py`.

You should now see both motors turning forward at the same time. If only one motor turns, check your wiring. If either of the motors turns backwards, swap the red and black wires for that motor over in the terminal block. Keep running the code until you have the wires in the right place.

Summary

You now have a basic robot whose wheels can be turned in the correct direction.

In the next worksheet, you will learn how to drive the robot backwards, left and right.