Project 3 – Motorised Percussion

Can a robot play the drums? Well it can with a little help.

For this project you will need

A Raspberry Pi B+, 2 or 3

The latest version of Raspbian

A Cam Jam Edukit #3 Robotics

AA Batteries

Building bricks or other construction materials

## Building the hardware.

Attaching the Cam Jam Edukit motor controller to the Raspberry Pi requires little more than slotting it on top of the Raspberry Pi, ensuring that it connects with the first 26 GPIO pins. With pin 1 being the pin closest to the SD card.

You will also need to screw your motors and battery pack to the correct terminals of the controller.

When connected attach your peripherals and power up your Raspberry Pi to the desktop.

How the robot is built is entirely up to you. We used Meccano to build a simple frame and then cable tied the motors to the frame. Cable ties were also used to hold the drumsticks to the wheels of our robot. Our cymbals were two platters taken from a broken hard drive but you can use any items that make a noise. Try bells, wood and plastic for different tones.

## Coding the project.

We start by going to the main menu, located in the top left of the screen, and click on

Programming » Python 3.

A new screen will open, in this screen click on File » New Window to open a new editor window. In the new editor window click on File » Save and call your work "percussion.py"

We start our code by importing a series of modules. We will use GPIO Zero to make working with the Raspberry Pi GPIO easier. We also import time to control the pace of our project.

*from gpiozero import Motor*

*import time*

Next we create two variables, motor1 and motor2. These will be used to store the location of the GPIO pins used to control the motors.

*motor1 = Motor(9,10)*

*motor2 = Motor(7,8)*

Now we use an infinite loop to continually run our code.

*while True:*

Using a for loop, we use a range of two, which means to run the code twice. The code that will be run controls the motors, motor1 travelling backward at only a third of its top speed. Motor2 travels forward at 90% of its full speed. Values between 0.1 and 1.0 can be used for the speed, but the motors will stall if you start at 0.1. We use a second for loop to change the pace, giving our percussion a noticeable tempo change.

*for i in range(2):*

*motor1.backward(speed=0.3)*

*motor2.forward(speed=0.9)*

*time.sleep(1)*

*for i in range(2):*

*motor1.backward(speed=1)*

*motor2.forward(speed=0.2)*

*time.sleep(1)*

We now break out of the for loops and stop both the motors for one second, before the loop repeats once more.

*motor1.stop()*

*motor2.stop()*

*time.sleep(1)*

## *Complete code listing.*

*from gpiozero import Motor*

*import time*

*motor1 = Motor(9,10)*

*motor2 = Motor(7,8)*

*while True:*

*for i in range(2):*

*motor1.backward(speed=0.3)*

*motor2.forward(speed=0.9)*

*time.sleep(1)*

*for i in range(2):*

*motor1.backward(speed=1)*

*motor2.forward(speed=0.2)*

*time.sleep(1)*

*motor1.stop()*

*motor2.stop()*

*time.sleep(1)*

## *Starting the project*

In Python, click on Run >> Run Module to start the code. Make sure that the battery box is turned on, and that your hands are out of the way.