A Python BBC Microbit Project: 'Meriden Robot'

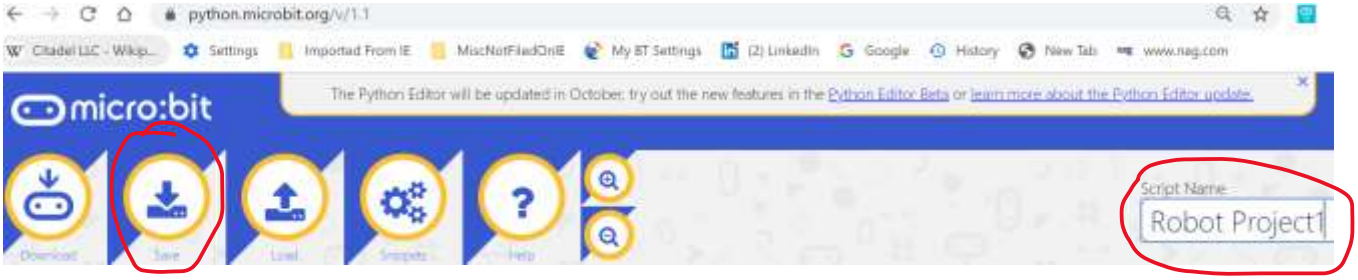
Introduction

In this project you will program your micro:bit to control a Robot using Python Code.

Getting Set-up to program the Meriden Robot

Using the Internet Explorer *(not Chrome)* search for Python Editor for micro:bit or go to www.microbit.org. Select Create Code or Let's create code. You need the Python Editor. Click Start with this editor.

At the top of the screen, in the right, change the name of your script to Robot Project.

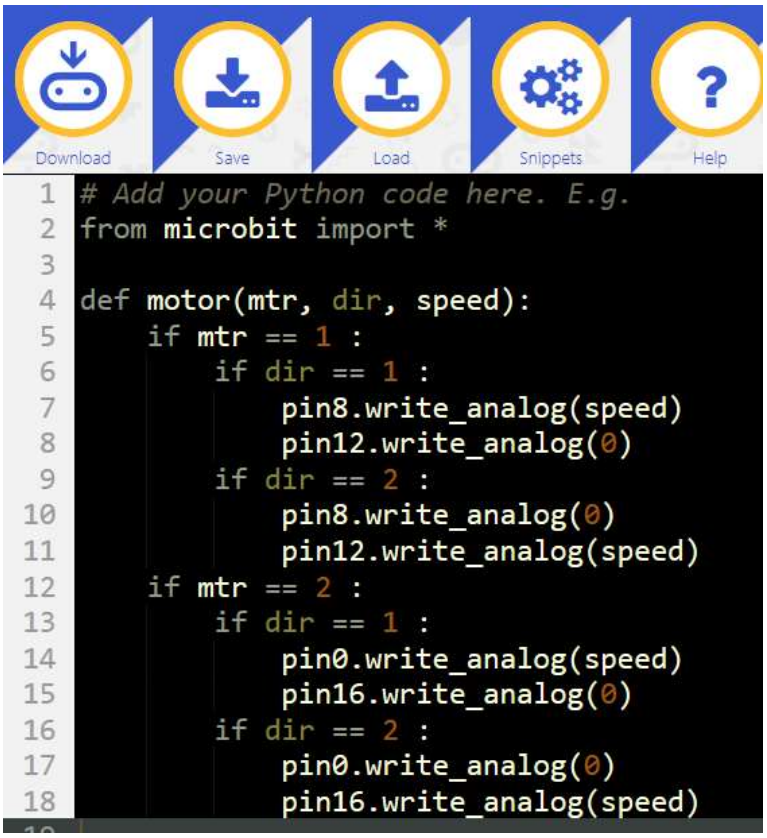


Delete lines 5, 6, 7 & 8	Download Save Load Snippets 1 # Add your Python code here. E.g. 2 from microbit import * 3 4 5 while True: 6 display.scroll('Hello, World!') 7 display.show(Image.HEART) 8 sleep(2000) 9

We will now create the code that controls the two motors, that control the two wheels of the robot.

Now highlight the text below and then copy (CTRL C) the following lines of code from this document and paste (CTRL V) to your Python Project.

```
def motor(mtr, dir, speed):
    if mtr == 1 :
        if dir == 1 :
            pin8.write_analog(speed)
            pin12.write_analog(0)
        if dir == 2 :
            pin8.write_analog(0)
            pin12.write_analog(speed)
    if mtr == 2 :
        if dir == 1 :
            pin0.write_analog(speed)
            pin16.write_analog(0)
        if dir == 2 :
            pin0.write_analog(0)
            pin16.write_analog(speed)
```



- There are two motors: show above as 'mtr' – 1 or 2
- Wheel directions forward as 1, backwards as 2
- Speed can be in the range of 0 to 1000

(1) Meriden Robot – First Movements

OK. Now we can begin properly. Add the following code…

You do not have to added the commented code (anything prefixed with a #)

```
# these first three lines make sure when we 'plug in' the motors are stopped for 3 seconds
motor(1,1,0)
motor(2,1,0)
sleep (3000)  #3000 milliseconds = 3 seconds
# now we tell both motors to move for 5 seconds
motor(1,1,250)
motor(2,1.250)
sleep (5000)
# now we stop the motors
motor(1,1,0)
motor(2,1,0)
```

- Save your work giving it a useful file name e.g. Robot Project1

  Load/Save,

  Download Python Script

  Save As (Save As can be found from the Save button – click on the upside down triangle)

  Save your file to your preferred location, ideally to your network drive where you can recall this when working from a different PC/laptop]

  Download Project Hex*

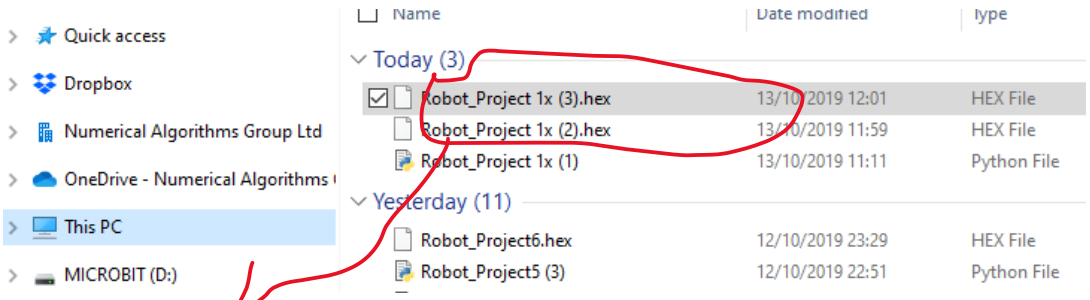
  Save As

  Save the Hex directly onto the Microbit (D:)

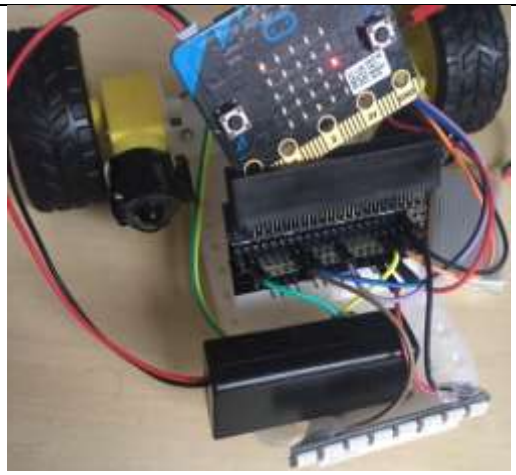
*ALTERNATIVELY

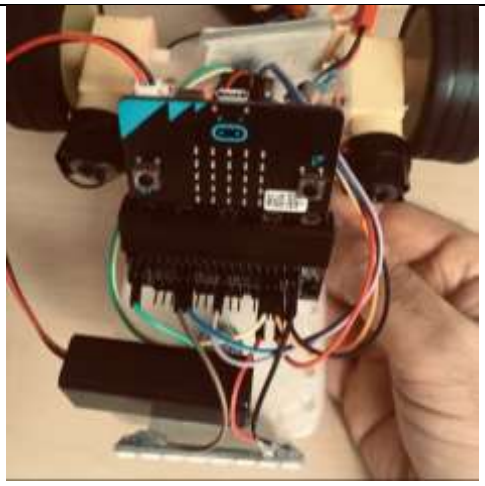
- Download the Hex file to your preferred location

- Open folder
- Drag and Drop the hex file you have just saved and downloaded to your PC onto the Microbit (D:)



Now fit the Microbit to the Robot as shown below





Put robot on the floor, next fit motor battery

…… if you've done everything correct your Robot will move and you have done your first project 'First Movements'

CONGRATULATIONS

(2) Meriden Robot – Take Control

In your first project putting the batteries in could be a little bit hazardous with the robot moving as soon as both batteries were in. Let's add some control and checks.

We would like you to use buttons to

- Test the hex file has downloaded successfully and has no Syntax errors
- Start the motor

In the real world when programming you are not given worksheets, so we are going to encourage you to work in a similar manner and search the internet or read the online manual.

Search for BBC micro:bit MicroPython documentation (or click on the hyperlink)

Look for the Buttons tutorial, read the Example

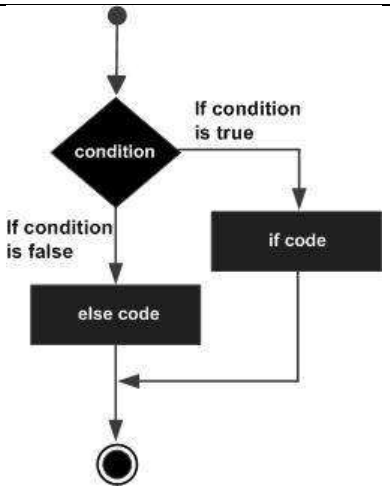
Need a hint…. The code you will is found in the Handling an Event section of the Buttons tutorial

YOUR TASKS:

- Make your code move if one of the button a is pressed
- Make your code display a letter or image if button b is pressed
- Else….. display something else

You will use the two lines of code from 'First Movements' within a "While true:….if…….elif….. else:"

```
if something is True:
    # do one thing
elif some other thing is True:
    # do another thing
else:
    # do yet another thing.
```



You will need to read the BBC micro:bit MicroPython documentation again to find a list of images and the command. Pick an image you have not used before!

(3) Meriden Robot – 'More movements'

In your second project 'Adding Some Control' you made it easier to get the Robot started and check your hex files were downloading OK. You also learnt to refer back to BBC micro:bit MicroPython documentation

3.1 More Robot Movements

You are now going to experiment with the two robot motors and their commands

One motor controls the left wheel and the other the right. You now know making the speed 0 stops the wheel(s)!

motor(1,1,500) - this means Motor 1, Motor Direction 1, Motor Speed 500

motor(2,2,500) - this means Motor 2, Motor Direction 2, Motor Speed 500

Motor Direction can be 1 or 2

Questions

Discuss with your colleague what you think will happen before testing it

What happens if you

- change motor direction on one motor? (i.e. the two motor directions are different)
- make motor speed zero on one motor?
- change motor speed on one motor? (i.e. so the motor speeds are different)

Challenges?

- Can you make the robot make a bigger circle / smaller circle
    - Experiment with the sleep function so it makes exactly one circle and not more or less

- Can you make the robot turn at 90 degrees (right angle)?
    - If so can you make it go forward, turn 90 degress and go forward again for the same distance (and stop)
    - Take it a bit further… can you make it complete a square?
    - A triangle?
    -

Note this may be a little bit fiddly so we don't expect it to be exact triangle or square, but you should be able to get close.

Super Challenge

Using what you have learnt above

- Can you use a For Loop to
    - Make your Robot complete 4 circles
- Can you use a For Loop to
    - Make your Robot complete 4 circles
    - and then complete 3 triangles in the opposite direction (i.e. clockwise -> anticlockwise or vice versa)

Written by

John Holden – @JohnCHolden , Jon Rogers, Worksheet Vn 0.2