

Forum:[Hack Forums](#)

Thread: [Unipolar stepper motor control](#)

SIM-1517404 - September 19, 2017, 12:22 pm

I've been working on this project for a little while now, and just a couple of days ago it worked fine, but now suddenly it doesn't work at all.

I'm trying to run it with Python on a Raspberry Pi 3 model B.

At this point, when I run the code, there is no movement in the motor at all. I can feel it tighten up a bit if I try to turn it while the code is running, but nothing else, not even vibrating.

I've tried different sequences, different GPIO pins, different RPi's, different drivers, switched out all the cables and changed power source.

Have anyone had a similar problem, or able to make out what the problem may be?

The wiring is messy, so I drew it instead

The code I'm running:

```
Code import RPi.GPIO as GPIO
import time
```

```
# Variables
```

```
delay = 0.0055
steps = 200
```

```
GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)
```

```
# Enable pins for IN1-4 to control step sequence
```

```
coil_A_1_pin = 18
coil_A_2_pin = 23
coil_B_1_pin = 24
coil_B_2_pin = 25
```

```
# Set pin states
```

```
GPIO.setup(coil_A_1_pin, GPIO.OUT)
GPIO.setup(coil_A_2_pin, GPIO.OUT)
GPIO.setup(coil_B_1_pin, GPIO.OUT)
GPIO.setup(coil_B_2_pin, GPIO.OUT)
```

```
# Function for step sequence
```

```
def set_step(w1, w2, w3, w4):  
    GPIO.output(coil_A_1_pin, w1)  
    GPIO.output(coil_A_2_pin, w2)  
    GPIO.output(coil_B_1_pin, w3)  
    GPIO.output(coil_B_2_pin, w4)
```

```
# loop through step sequence based on number of steps
```

```
# gives us a list so we can essentially iterate through each of the different step values
```

```
# meaning we can iterate through each "part" of the original stepping sequence
```

```
step_values = [{"w1":1, "w2":0, "w3":1, "w4":0},  
               {"w1":0, "w2":1, "w3":1, "w4":0},  
               {"w1":0, "w2":1, "w3":0, "w4":1},  
               {"w1":1, "w2":0, "w3":0, "w4":1}]
```

```
current_step_count = 0
```

```
def step_motor():
```

```
    global current_step_count
```

```
    needed_step_values = step_values[current_step_count % 4]
```

```
    set_step(needed_step_values["w1"], needed_step_values["w2"], needed_step_values["w3"],
```

```
    needed_step_values["w4"])
```

```
    time.sleep(delay)
```

```
    current_step_count += 1
```

```
# this should turn everything off to stop them moving -- not sure if this is necessary
```

```
# trying commenting it out if it doesn't work with it in
```

```
set_step(0, 0, 0, 0)
```

```
for i in range(0, steps):
```

```
    step_motor()
```

Goodies - September 19, 2017, 3:01 pm

I assume you have experience with RPis, but the first thing I'd ask if you're sure of the GPIO mode. I know different RPi models have different BCM values for the pins. If you use GPIO.BOARD, you can use the pin values that are printed on the board itself.

Seems to me like you've changed out everything but the motor. Verify that there is current flowing using a multimeter. If there is, it could be the motor.

Looking at the diagram, the pins do look correct. Multimeter time!

SIM-1517404 - September 19, 2017, 11:33 pm

(Sep 19, 2017 03:01 PM)Goodies Wrote: [\[CITE\]](#) I assume you have experience with RPis, but the first thing I'd ask if you're sure of the GPIO mode. I know different RPi models have different BCM values for the pins. If you use GPIO.BOARD, you can use the pin values that are printed on the board itself.

Seems to me like you've changed out everything but the motor. Verify that there is current flowing using a multimeter. If there is, it could be the motor.

Looking at the diagram, the pins do look correct. Multimeter time!

I've tried using GPIO.BOARD also, no difference. I've also put voltage directly to the motor,

energising each coil, and the motor steps when I do this. I energise it in the sequence that is written in the code, and it steps clockwise.