Hello,

I started to use the Adafruit motor hat with this library.

I discovered several issues:

1. When a new motorhat is initialised, a new steppermotor instance is also initialized in line 230:  
   *self.steppers = [ Adafruit\_StepperMotor(self, 1), Adafruit\_StepperMotor(self, 2) ]*  
     
   In this way. The steps per revolution is not changeable during initialization and remains 200.
2. The timing is of by using *time.sleep(s\_per\_s)* in line 159. This can be greatly Improved by:  
   *time.sleep(time.time() - time\_before\_start - s\_per\_s)*
3. The speed of execution can be improved by only sending the commands needed with this snippet starting from line 130:  
    *if (dir == Adafruit\_MotorHAT.FORWARD):*

*prev\_coil = step2coils[(self.currentstep/(self.MICROSTEPS/2) - 1) % len(step2coils)]*

*else:*

*prev\_coil = step2coils[(self.currentstep/(self.MICROSTEPS/2) + 1) % len(step2coils)]*

*coils = step2coils[self.currentstep/(self.MICROSTEPS/2)]*

*#print "coils state = " + str(coils)*

*pin\_to\_set = [self.AIN2, self.BIN1, self.AIN1, self.BIN2]*

*for pin in range (0, 4):*

*if prev\_coil[pin] != coils[pin]:*

*self.MC.setPin(pin\_to\_set[pin], coils[pin])*

1. Is it possible that at line 239, the logic bits are reversed?

|  |
| --- |
| *if (value == 0):* |
|  |

|  |
| --- |
|  |
| *self.\_pwm.setPWM(pin, 0, 4096)* |
|  |
|  |
| *if (value == 1):* |
|  |

|  |
| --- |
| *self.\_pwm.setPWM(pin, 4096, 0)* |

Shouldn’t it be like this:

|  |
| --- |
| *if (value == 1):* |
|  |

|  |
| --- |
|  |
| *self.\_pwm.setPWM(pin, 0, 4096)* |
|  |
|  |
| *if (value == 0):* |
|  |

|  |
| --- |
| *self.\_pwm.setPWM(pin, 0, 0)* |

5. The loop running the commands is locked and cannot be aborted. This is often needed in all sorts of situations. Is it possible to have an abort bit programmed in the loop in line 157

5.