Laser_Tests_02-Copy1

September 2, 2017

1 CNC Laser Power Settings Test 3

2 Code:

```
In [1]: import GCode
        import GRBL

In [2]: cnc = GRBL.GRBL(port="/dev/cnc_3018")

        print("Laser Mode: {}".format(cnc.laser_mode))
```

```
Laser Mode: 0.0
```

```
In [3]: cnc.laser_mode = 1
```

```
ok
```

```
In [4]: from enum import IntEnum
        class Tool(IntEnum):
            SPINDLE = 0
            LASER = 1

In [5]: from enum import IntEnum
        class LaserPower(IntEnum):
            CONSTANT = 0
            DYNAMIC = 1

In [6]: LaserPower.CONSTANT
```

```
Out[6]: <LaserPower.CONSTANT: 0>
```

```
In [12]: def init(power = LaserPower(0), feed = 200, laser = 25):
             program = GCode.GCode()
             program.G21() # Metric Units
             program.G91() # Absolute positioning.
             program.G1(F=feed) #
```

```
            if power==LaserPower.CONSTANT:
                program.M3(S=laser) # Laser settings
            else:
                program.M4(S=laser) # Laser settings
            return program
```

```
In [13]: def end():
            program = GCode.GCode()
            program.M5() # Laser settings.
            return program
```

```
In [14]: def square(size=20):
            program = GCode.GCode()
            program.G1(X=size)
            program.G1(Y=size)
            program.G1(X=-size)
            program.G1(Y=-size)
            return program
```

2.1 Test Setup

Position the paper & other things.

```
In [15]: cnc.cmd("M5") # Laser off
```

```
Out[15]: ['ok', 'ok']
```

```
In [16]: # Set minimal power setting to focus and position laser
         cnc.cmd("M3 S1")
         cnc.cmd("G1 X0") # Laser On
```

```
Out[16]: ['ok', 'ok']
```

```
In [30]: def pulse(pulse_duration=100):
            prog = GCode.GCode()
            prog.M5()
            prog.G1(X=0)
            prog.M3(S=255)
            prog.M4(P=pulse_duration)
            prog.G1(X=0)
            prog.M5()
            return prog
```

```
In [31]: pulse(100)
```

```
Out[31]: <GCode>[cmds=6]
```

```
In [32]: cnc.run(pulse(100))
```

```
^C
```

```
Out[32]: 21.264846563339233

In [38]: test_run=GCode.GCode()
         test_run.G21()
         test_run.G91()
         test_run.G0(F=500)
         test_run.G1(F=500)
         for test_num in range(16):
             pulse_duration = (test_num+1)*25
             test_run += pulse(pulse_duration)
             test_run.G0(X=5)
         test_run+=end()

In [39]: test_run

Out[39]: <GCode>[cmds=117]

In [40]: cnc.run(test_run)

Out[40]: 30.084847450256348

In [37]: cnc.status

Out[37]: '<Idle|MPos:-123.276,0.000,2.800|Bf:15,127|FS:0,0|Ov:100,100,100>'
```