

Peter G. Halverson, Ph.D.

Here is a simple program to control two servo motors using the LabJack U3 to generate PWM (pulse-width modulated) signals.

```
##### pwm_for_servos_example.py #####
# Peter Halverson Nov. 1, 2014
# The LabJack U3 can output two pulse-width modulated signals.  You could use it to control
# two servo motors.
#
# PWM signal 0 comes out of the FI04 screw terminal.
# PWM signal 1 comes out of the FI05 screw terminal.  (Strange, but that's the way it is.)
#
# Documentation that allowed me to write this code is at
# http://labjack.com/support/u6/users-guide/2.9.1.1
# and in u3.py
import u3
d = u3.U3() # Opens first found U3 over USB
d.configIO(NumberOfTimersEnabled = 2)
d.configTimerClock(TimerClockBase = 6, TimerClockDivisor = 15)
# This combination selects the 48 MHz timebase.
# The PWM frequency will be 48 MHz/(15*2^16) which works out to 48.82 Hz, close to
# the 50 Hz standard for analog servos.
# Various frequencies are available.  0.06 to 732 Hz in mode 0.  15.26 to 187500 Hz in mode 1
# See the documentation.
print("This example allows the LabJack U3 control two analog servos using PWM.")
print("The control signals will come out of the FI04 and FI05 screw terminals.")
print("The PWM control number can be from 0 to 65535.")
print("Enter numbers to control the PWM width, which will cause the servo motor to move.")
print("(Exactly what the motor does depends on the type of motor.)")
print("Entering 0 will cause the PWM output to be high 100% of the time.")
print("Entering 32768 will cause the PWM output to be high 50% of the time.")
print("Entering 65535 will cause the PWM output to be high almost 0% of the time.")
print("Entering -1 will stop the program")
i=0
while (i>-1):
    i = int(raw_input('Choose a number: '))
    d.getFeedback(u3.Timer0Config(TimerMode = 0, Value = i))
    d.getFeedback(u3.Timer1Config(TimerMode = 0, Value = i))
```
