# ================ EXAMPLE PYTHON CODE - Use chunks as needed ================

# For full documentation see https://www.wifistepper.com/interfaces

import goodrobotics as gr

host = "wsx100.local" # or "192.168.4.1"

# Create motor object

MOTOR = gr.WifiStepper(host=host)

# MOTOR = gr.WifiStepper(proto=gr.ComCrypto, host=host, key="<AUTH\_KEY>") # For crypto auth

# Connect (will throw exception on failure)

MOTOR.connect()

# Write motor configuration

MOTOR.setconfig({

 'mode': "voltage", 'stepsize': 8,

 'ocd': 500, 'ocdshutdown': True,

 'maxspeed': 160, 'minspeed': 0, 'accel': 400, 'decel': 400,

 'vm\_kthold': 20, 'vm\_ktrun': 15, 'vm\_ktaccel': 20, 'vm\_ktdecel': 20,

 'fsspeed': 200, 'fsboost': False,

 'vm\_pwmfreq': 10.049999999999999, 'vm\_stall': 750, 'vm\_volt\_comp': False,

 'vm\_bemf\_slopel': 0.003, 'vm\_bemf\_speedco': 61.5,

 'vm\_bemf\_slopehacc': 0.1005, 'vm\_bemf\_slopehdec': 0.1005,

 'reverse': False, 'save': True

})

# Read configuration

config = MOTOR.getconfig()

# config is a dict() of configuration map

# Zero servo position

MOTOR.resetpos()

# Send servo command

stepsize = MOTOR.getconfig()['stepsize']

angle = gr.Angle(stepsize=stepsize)

MOTOR.goto(pos=angle(0))

# Emergency stop motor

MOTOR.estop()

# Read status

state = MOTOR.getstate()

# state is a dict() with keys: busy, stepss, hiz, vin, pos, mark, dir, movement, stepclock, switch,

# flags=dict( switch, commanderror, overcurrent, undervoltage, thermalwarning,

# thermalshutdown, stalldetect )