



**DesignNews**

## Raspberry Pi 5 Automation Lecture Series

**Day 4:**

### Driving Stepper Motors with the Raspberry Pi 5

Sponsored by

**DigiKey**



## Webinar Logistics

- Turn on your system sound to hear the streaming presentation.
- If you have technical problems, click “Help” or submit a question asking for assistance.
- Participate in ‘Attendee Chat’ by maximizing the chat widget in your dock.



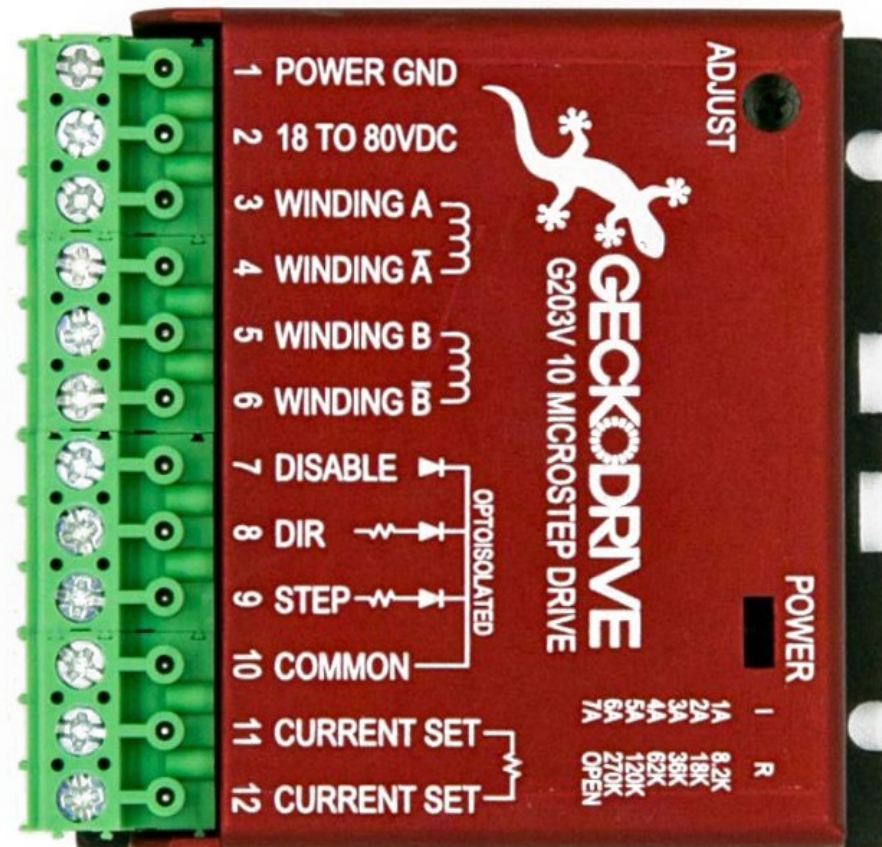
## Fred Eady

Visit 'Lecturer Profile' in your console for more details.



# AGENDA

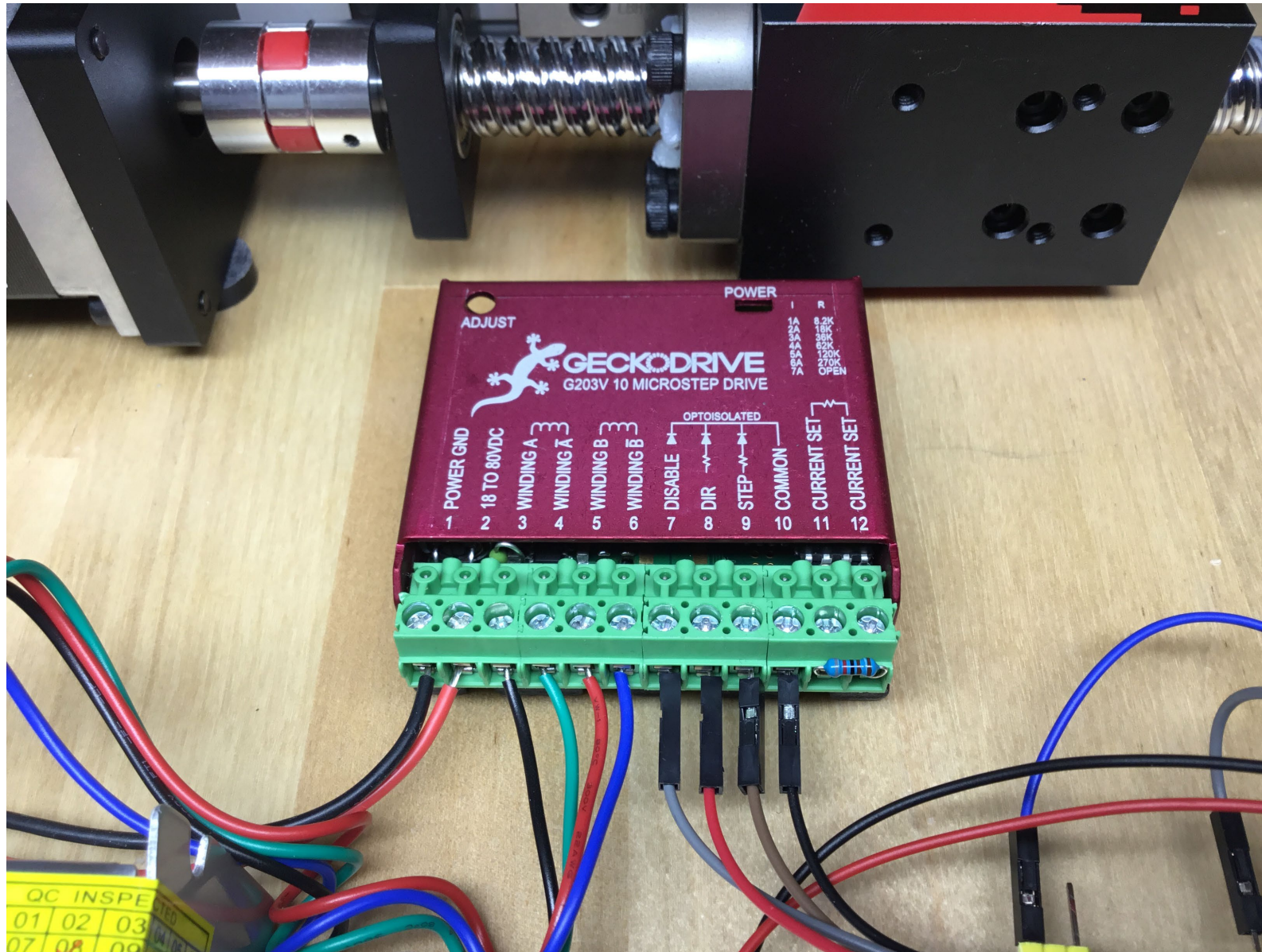
- **Raspberry Pi 5 Stepper-Motor-Driven Linear Stage Hardware**
- **Driving a Linear Stage with a Raspberry Pi 5 and Python3**





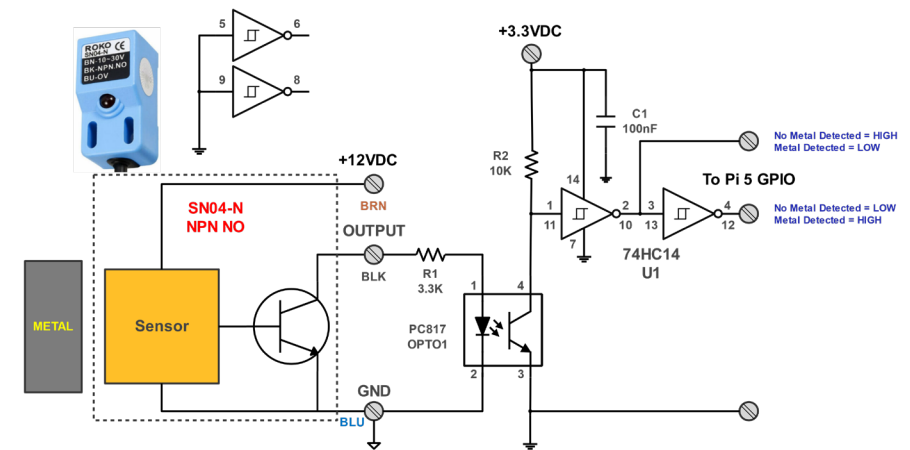
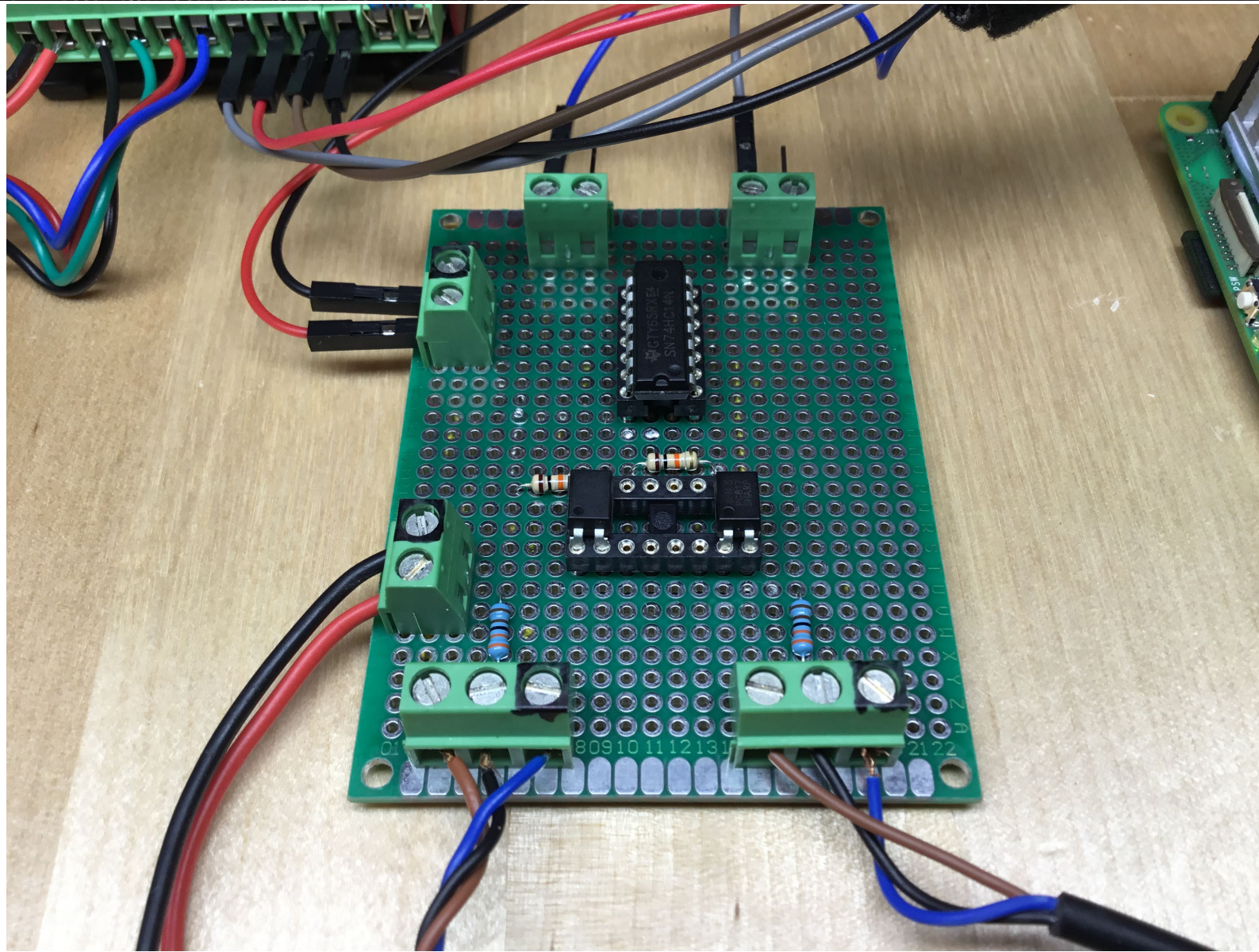
**Stepper Motor and Limit Switch Power Supplies**



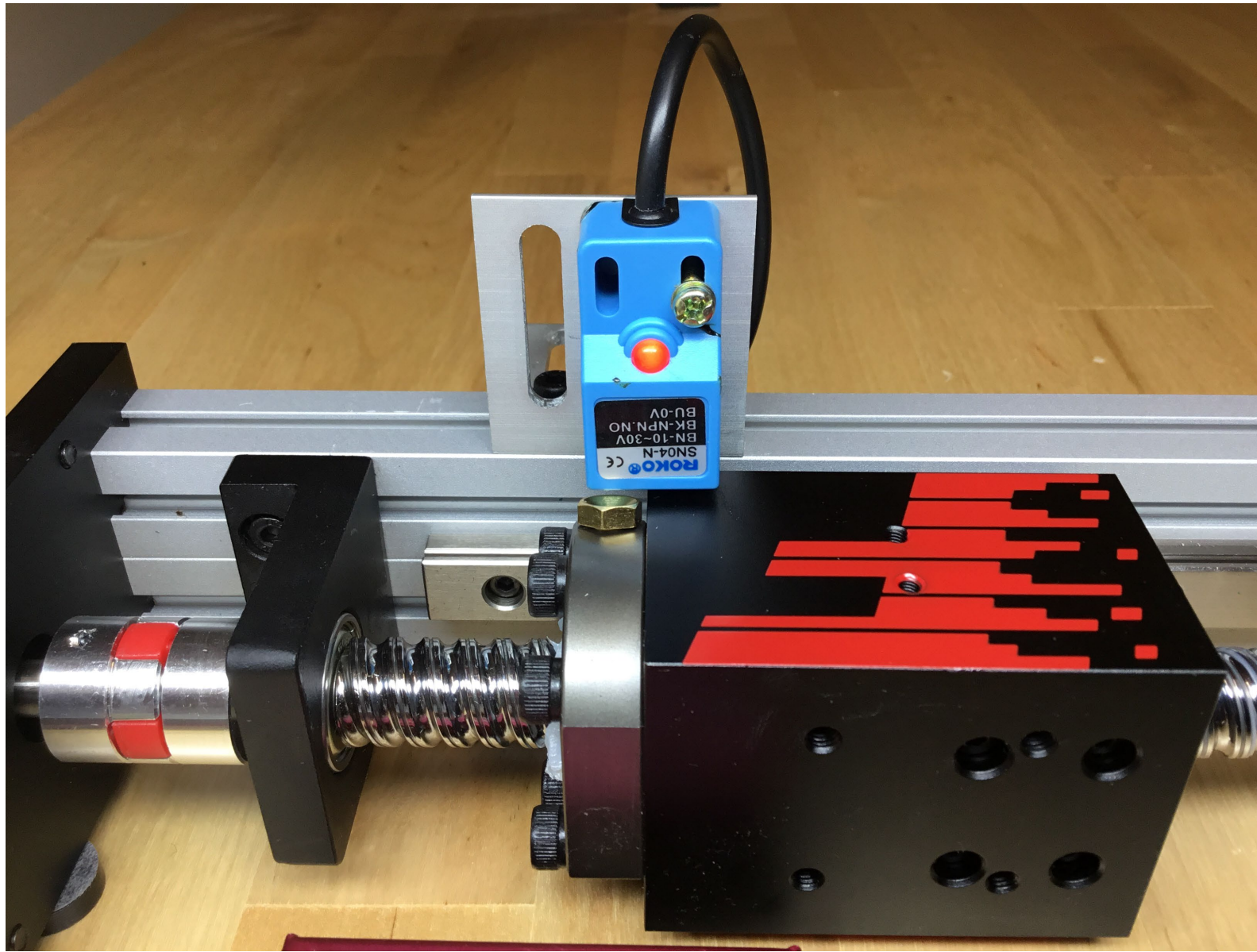
**GECKODRIVE Vampire 10 Microstep Stepper Motor Drive**



## Limit Switch Interface

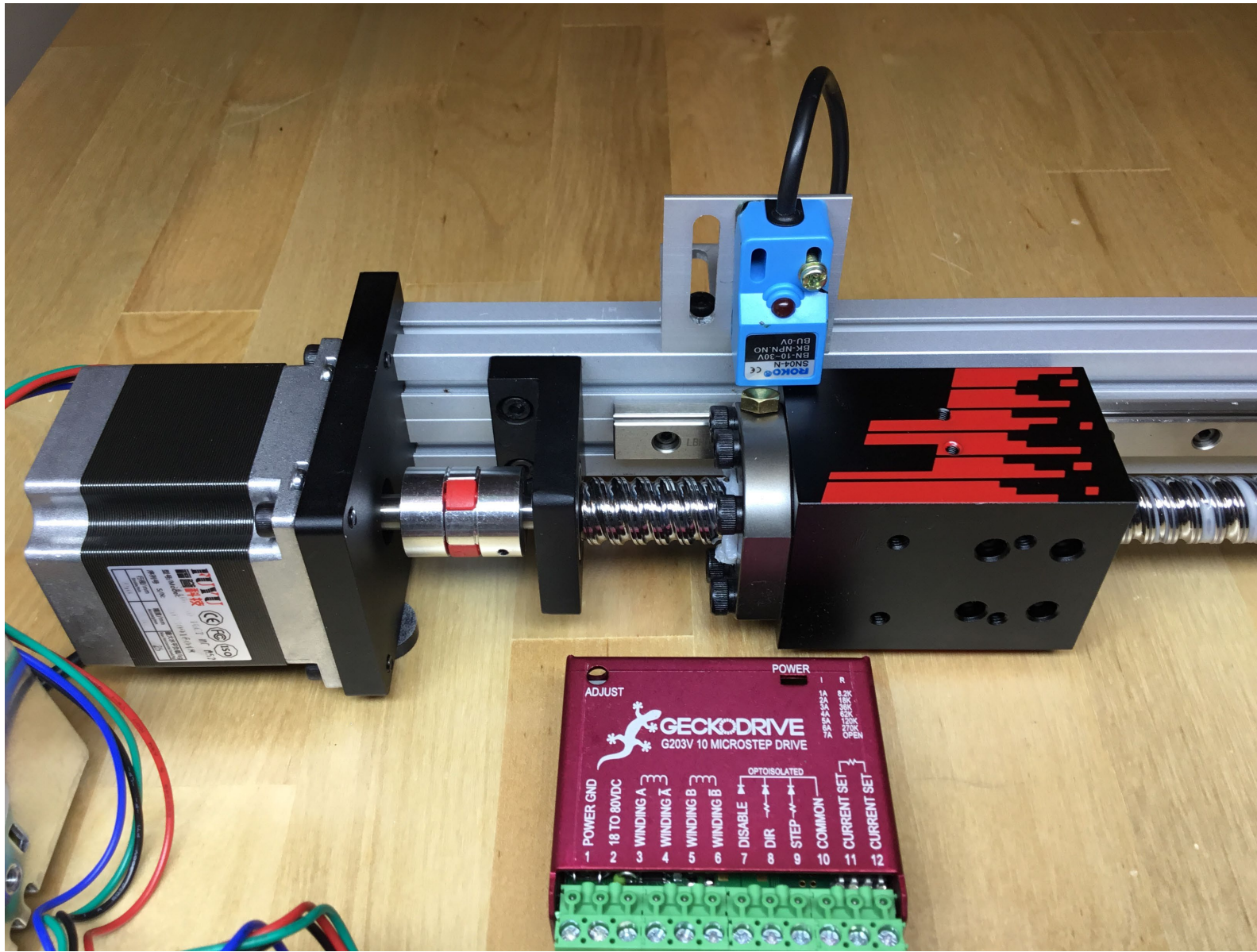




**Limit Switch**

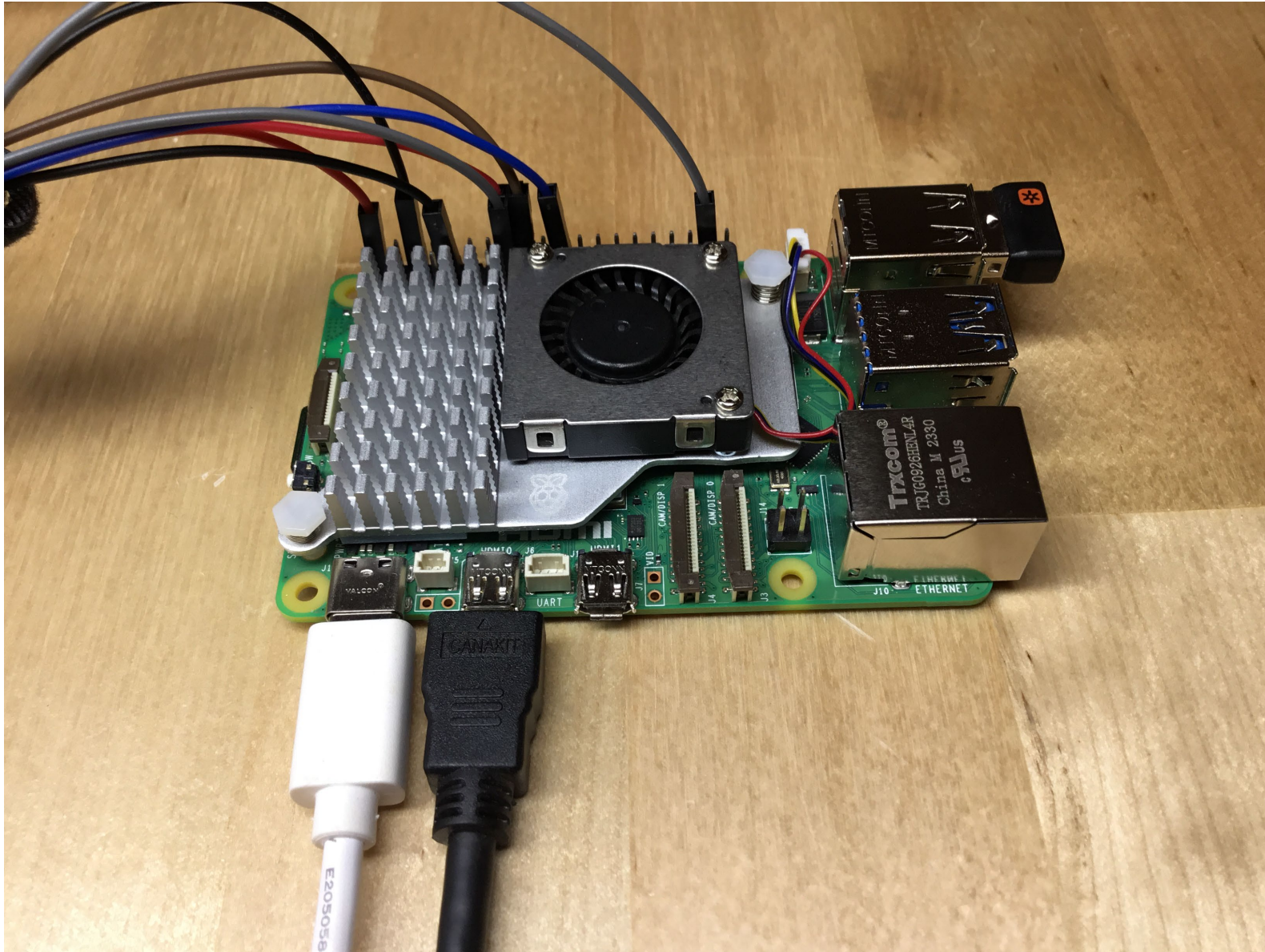


## Linear Stage



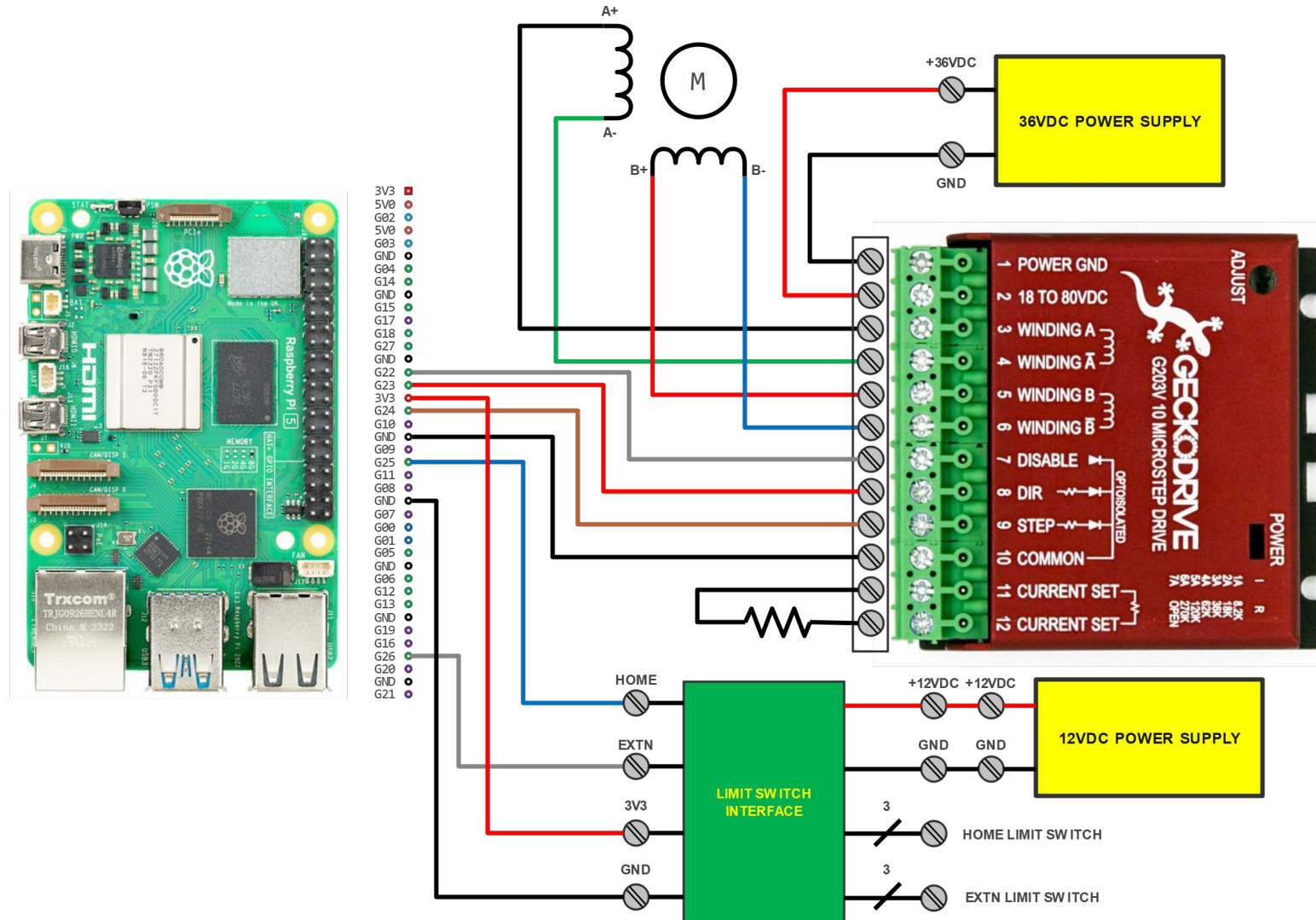


## Raspberry Pi 5 Controller

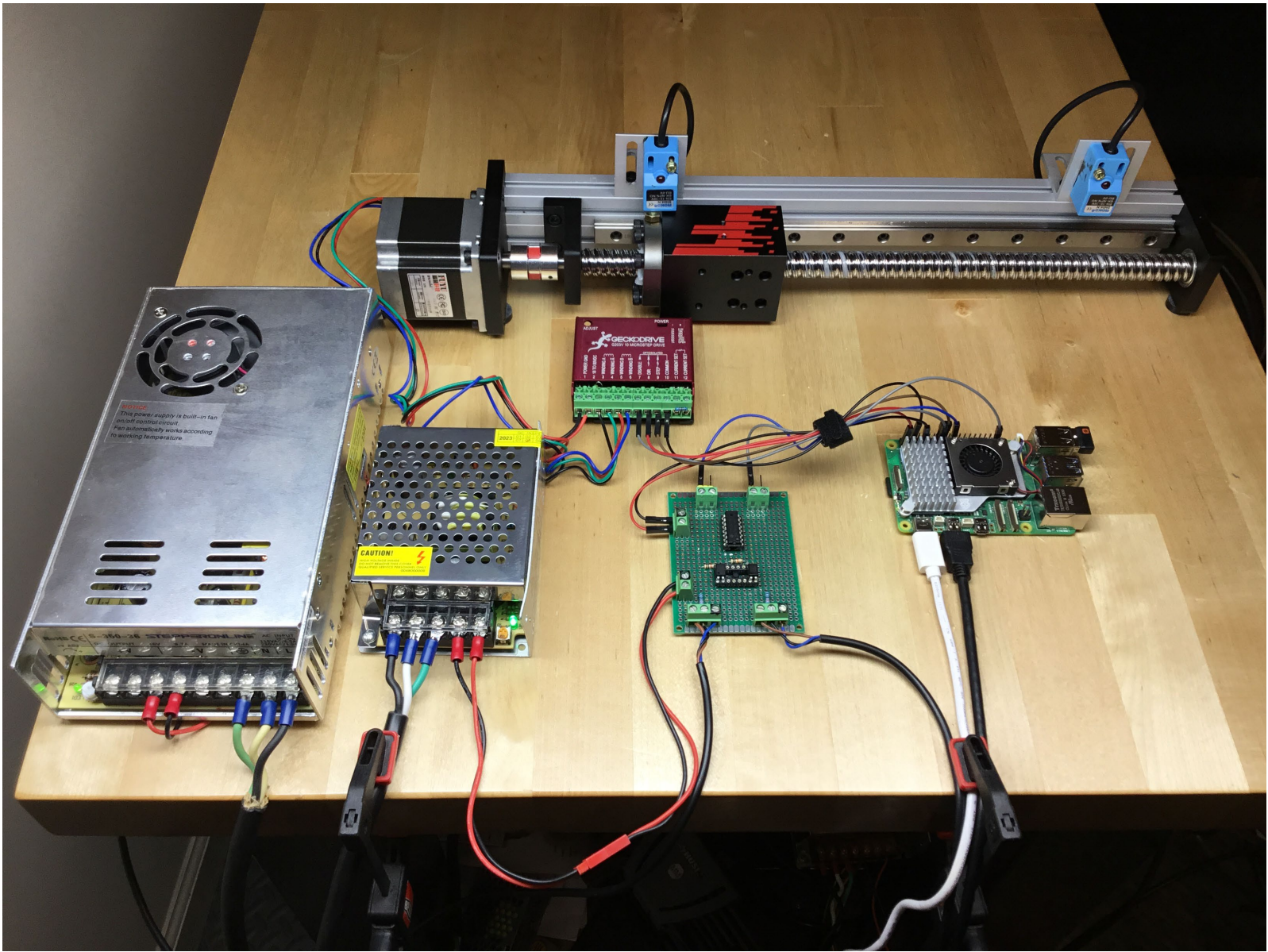




## Linear Stage Drive Component Diagram



**Linear Stage Drive Composite View**





## Initialize the Raspberry Pi 5 GPIO

```

MOTION_ENABLED = 0
MOTION_DISABLED = 1
FORWARD = 1
REVERSE = 0

class moveConstants:
    forward = 1
    reverse = 0

global fdirection

chip = gpiochip.Chip('gpiochip4')

disable_line = chip.get_line(DISABLE_PIN)
dir_line = chip.get_line(DIR_PIN)
step_line = chip.get_line(STEP_PIN)
home_line = chip.get_line(LIMITHOME_PIN)
extn_line = chip.get_line(LIMITEXTENT_PIN)

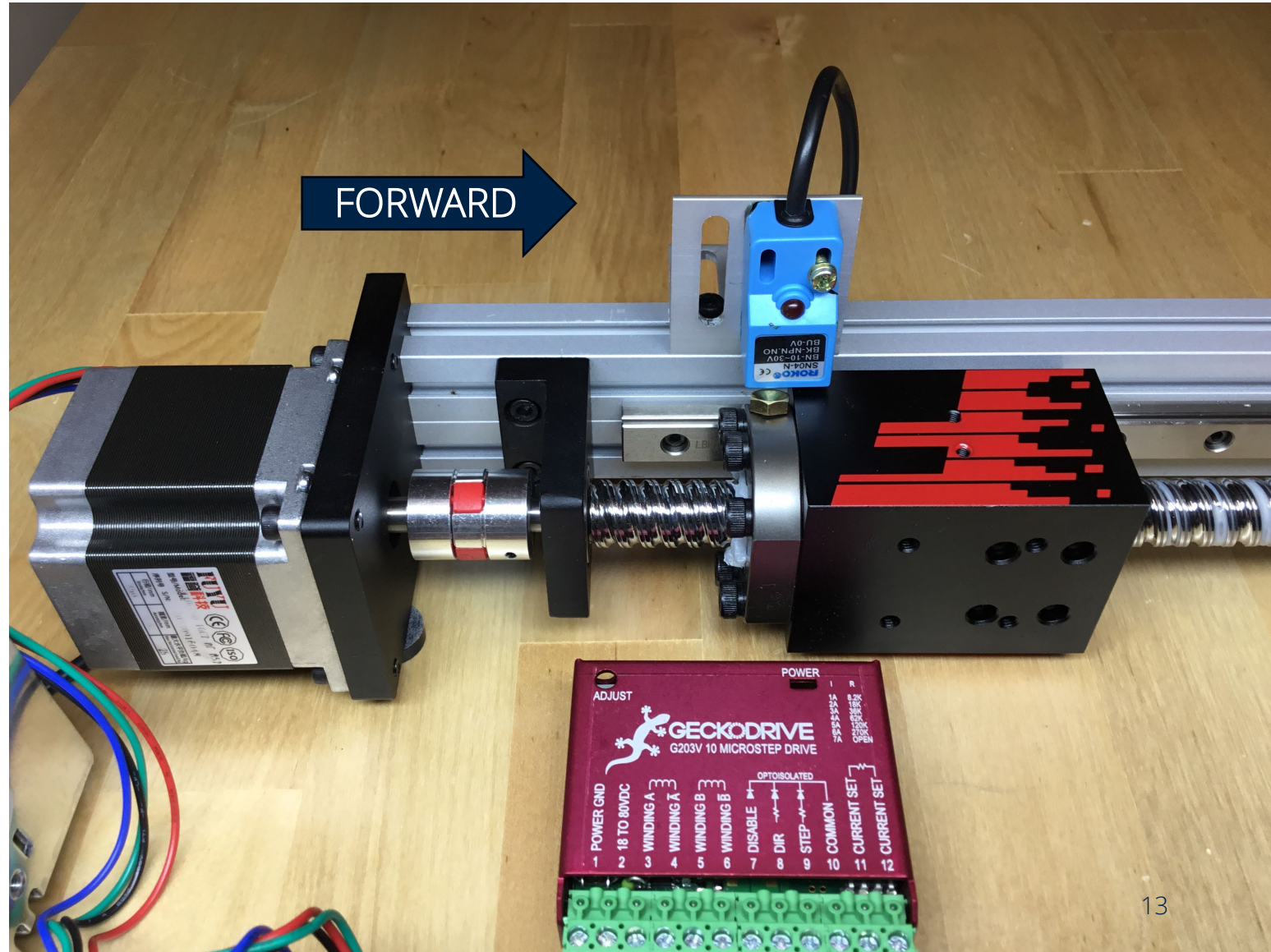
disable_line.request(consumer="disable",
                    type=gpiod.LINE_REQ_DIR_OUT)
disable_line.set_value(MOTION_DISABLED)

dir_line.request(consumer="dir", type=gpiod.LINE_REQ_DIR_OUT)
dir_line.set_value(REVERSE)
fdirection = REVERSE

step_line.request(consumer="step",
                 type=gpiod.LINE_REQ_DIR_OUT)
step_line.set_value(0)

home_line.request(consumer="home", type=gpiod.LINE_REQ_DIR_IN)
extn_line.request(consumer="extn", type=gpiod.LINE_REQ_DIR_IN)

```

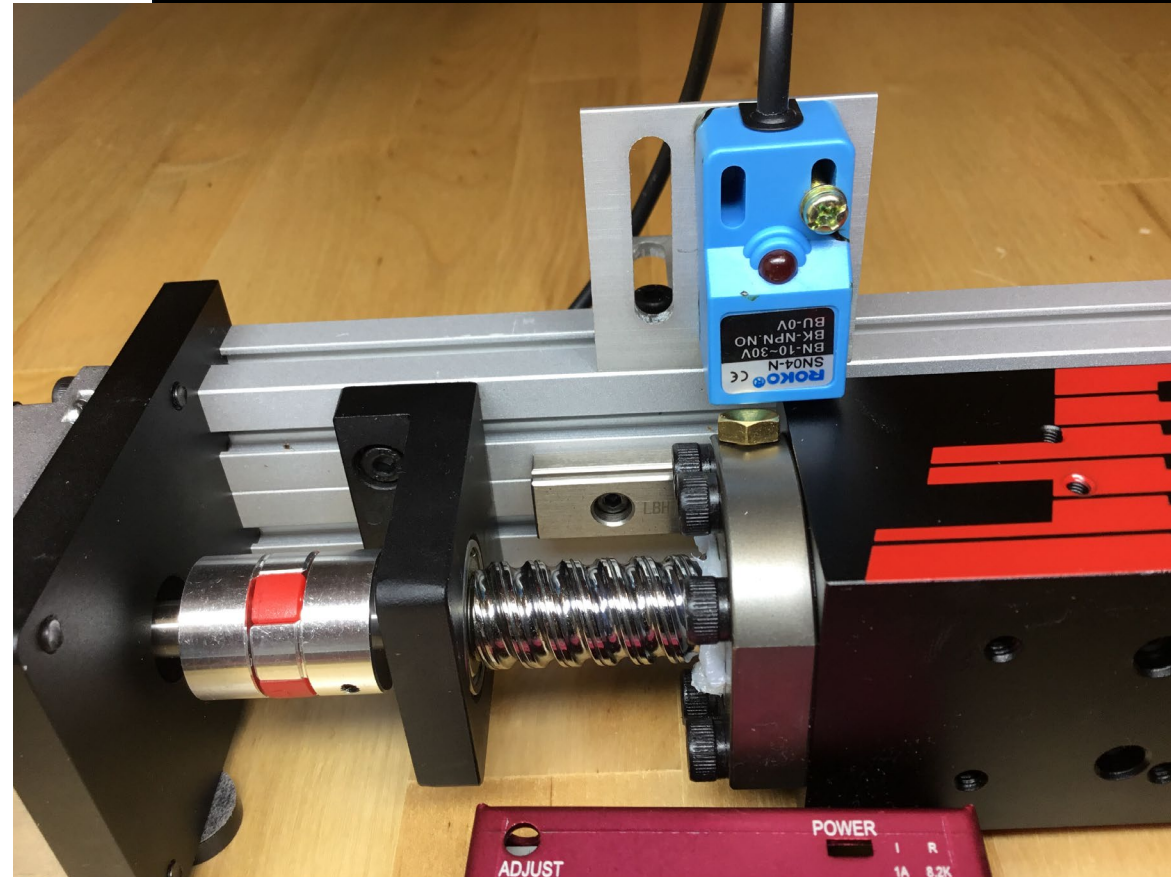




Linear Stage *home* Function

```
def home():
print("Performing HOME operation")
# puck is between home and extn limit switches
# move puck to home limit switch
if(home_line.get_value() == 0 and extn_line.get_value() == 0):
    disable_line.set_value(MOTION_ENABLED)
    dir_line.set_value(REVERSE)
    while(home_line.get_value() == 0):
        step_line.set_value(1)
        time.sleep(0.00005)
        step_line.set_value(0)
        time.sleep(0.00005)
# puck is at extn limit switch
# move puck to home limit switch
if(extn_line.get_value() == 1):
    disable_line.set_value(MOTION_ENABLED)
    dir_line.set_value(REVERSE)
    while(home_line.get_value() == 0):
        step_line.set_value(1)
        time.sleep(0.00005)
        step_line.set_value(0)
        time.sleep(0.00005)
# puck is at home limit switch
# move puck to home position
if(home_line.get_value() == 1):
    disable_line.set_value(MOTION_ENABLED)
    dir_line.set_value(FORWARD)
    while(home_line.get_value() == 1):
        step_line.set_value(1)
        time.sleep(0.00005)
        step_line.set_value(0)
        time.sleep(0.00005)
print("HOME operation completed")
```

```
pi5user@pi5a: ~/pi5_apps_python
File Edit Tabs Help
pi5user@pi5a:~/pi5_apps_python $ python3 pi5_linear_stage.py
Performing HOME operation
HOME operation completed
Stopped!
pi5user@pi5a:~/pi5_apps_python $
```

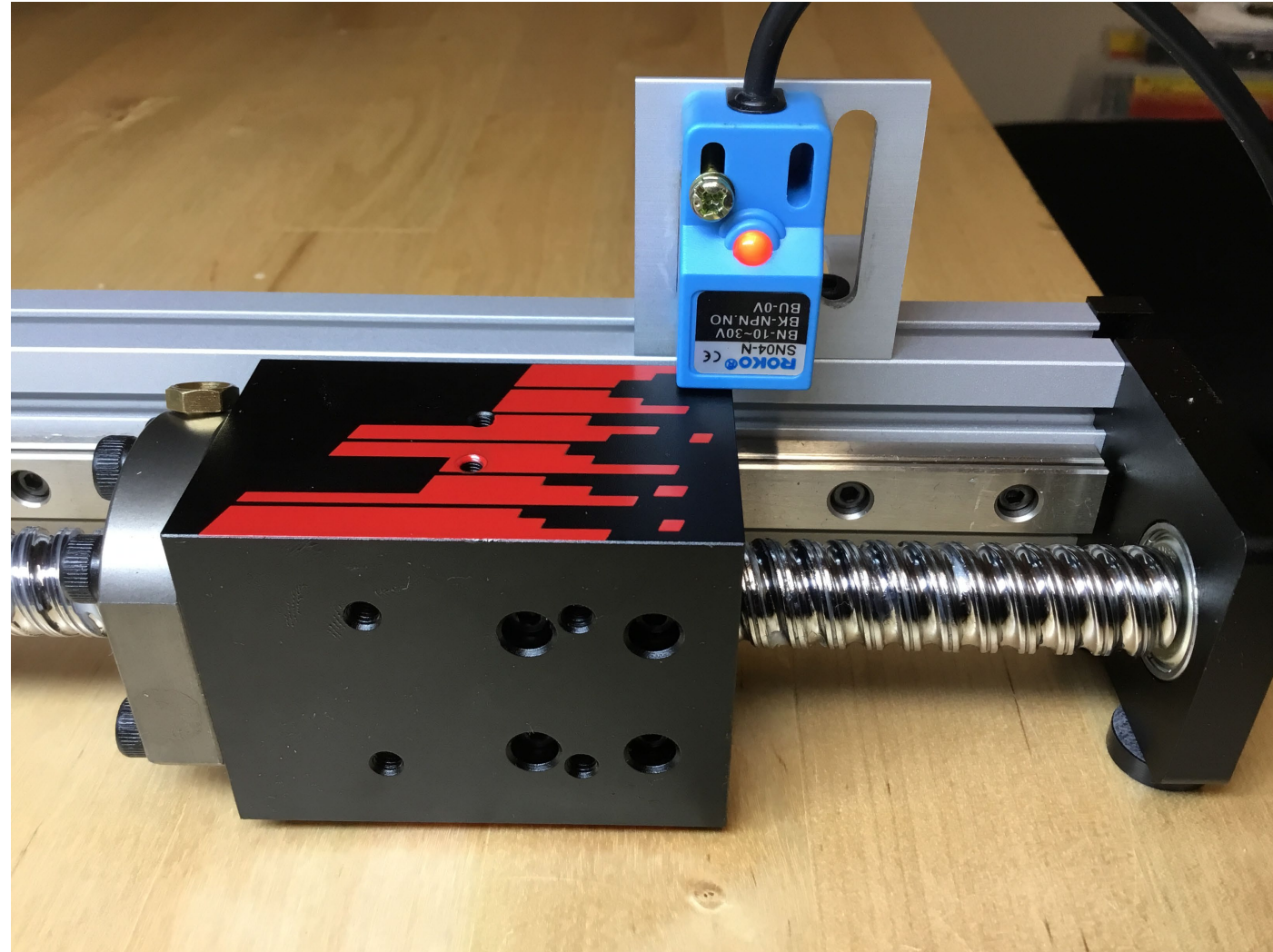




Linear Stage *moveToExtent* Function

```
def moveToExtent():  
    print("Moving To Extent")  
    dir_line.set_value(FORWARD)  
    while(extn_line.get_value() == 0):  
        step_line.set_value(1)  
        time.sleep(0.00005)  
        step_line.set_value(0)  
        time.sleep(0.00005)
```

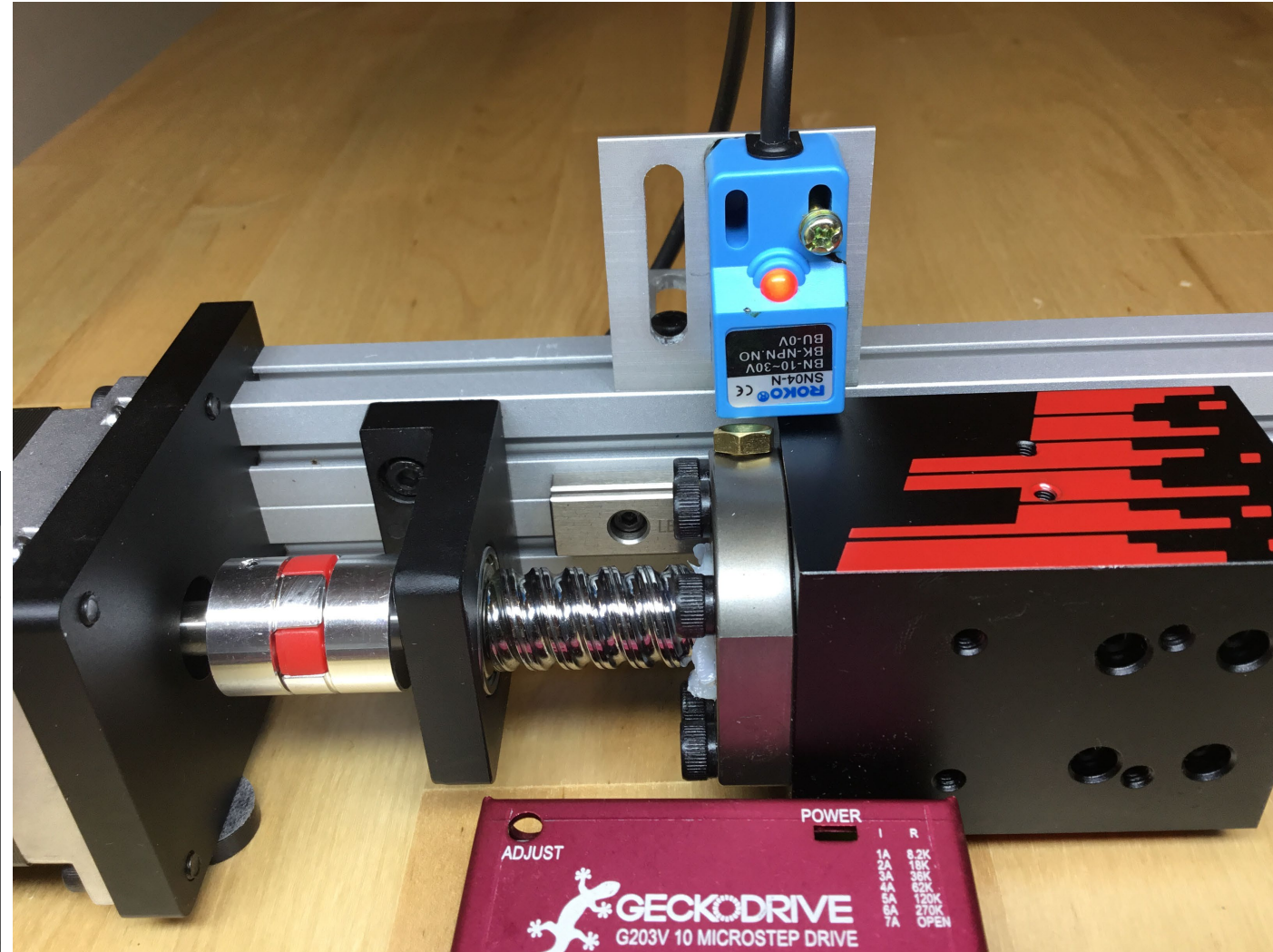
```
pi5user@pi5a: ~/pi5_apps_python  
File Edit Tabs Help  
pi5user@pi5a:~/pi5_apps_python $ python3 pi5_linear_stage.py  
Performing HOME operation  
HOME operation completed  
Moving To Extent  
Stopped!  
pi5user@pi5a:~/pi5_apps_python $
```





Linear Stage *moveToHome* Function

```
def moveToHome():  
    print("Moving To Home")  
    dir_line.set_value(REVERSE)  
    while(home_line.get_value() == 0):  
        step_line.set_value(1)  
        time.sleep(0.00005)  
        step_line.set_value(0)  
        time.sleep(0.00005)
```

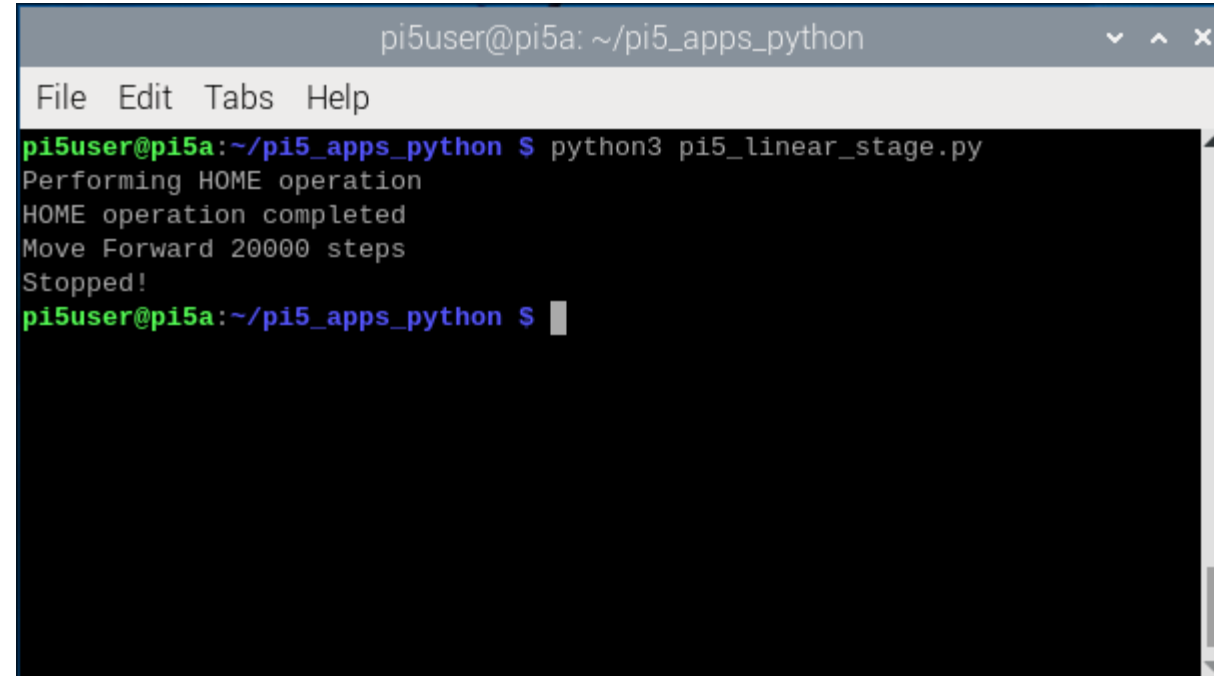


```
pi5user@pi5a: ~/pi5_apps_python  
File Edit Tabs Help  
pi5user@pi5a:~/pi5_apps_python $ python3 pi5_linear_stage.py  
Performing HOME operation  
HOME operation completed  
Moving To Home  
Stopped!  
pi5user@pi5a:~/pi5_apps_python $
```



Linear Stage *moveForward* Function

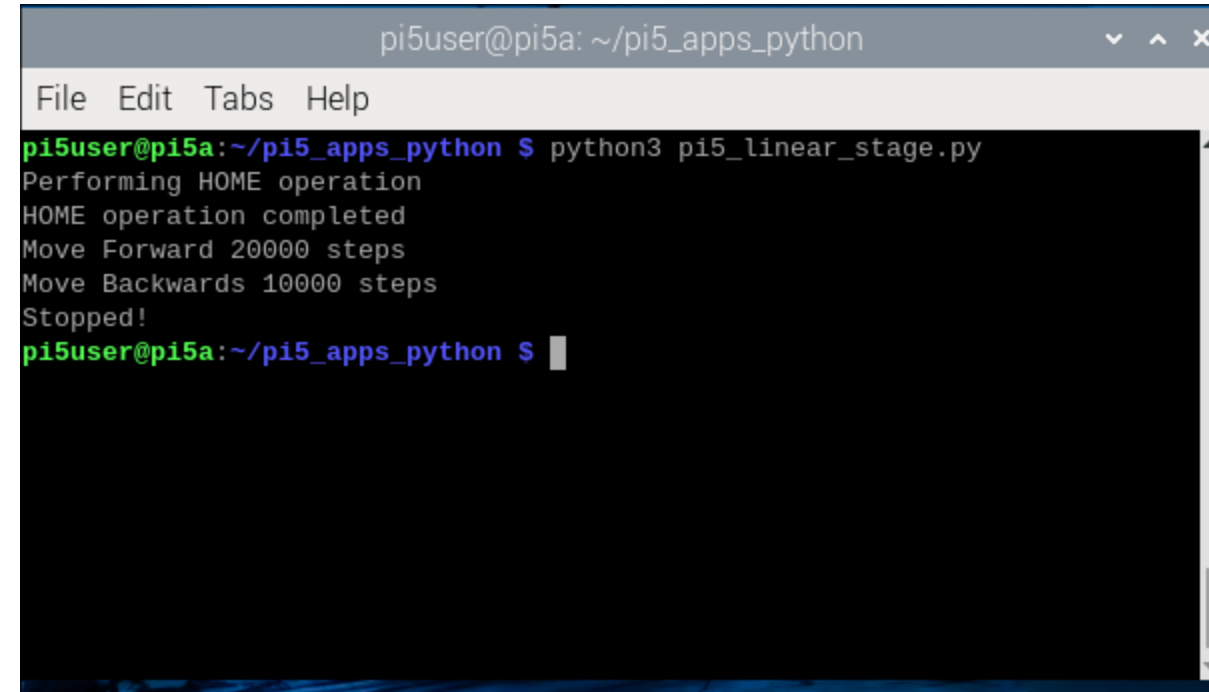
```
def moveForward(numofsteps):  
    print("Move Forward", numofsteps, "steps", sep=" ")  
    disable_line.set_value(MOTION_ENABLED)  
    dir_line.set_value(FORWARD)  
    for x in range(0, numofsteps):  
        step_line.set_value(1)  
        time.sleep(0.00005)  
        step_line.set_value(0)  
        time.sleep(0.00005)  
    if(extn_line.get_value() == 1):  
        disable_line.set_value(MOTION_DISABLED)  
        step_line.set_value(0)
```

A terminal window titled 'pi5user@pi5a: ~/pi5\_apps\_python' with a menu bar (File, Edit, Tabs, Help). The terminal shows the execution of 'python3 pi5\_linear\_stage.py'. The output is: 'Performing HOME operation', 'HOME operation completed', 'Move Forward 20000 steps', and 'Stopped!'. The prompt returns to 'pi5user@pi5a: ~/pi5\_apps\_python \$'.

```
pi5user@pi5a: ~/pi5_apps_python  
File Edit Tabs Help  
pi5user@pi5a:~/pi5_apps_python $ python3 pi5_linear_stage.py  
Performing HOME operation  
HOME operation completed  
Move Forward 20000 steps  
Stopped!  
pi5user@pi5a:~/pi5_apps_python $
```

## Linear Stage *moveBackward* Function

```
def moveBackward(numofsteps):  
    print("Move Backward", numofsteps, "steps", sep=" ")  
    disable_line.set_value(MOTION_ENABLED)  
    dir_line.set_value(REVERSE)  
    for x in range(0, numofsteps):  
        step_line.set_value(1)  
        time.sleep(0.00005)  
        step_line.set_value(0)  
        time.sleep(0.00005)  
    if(home_line.get_value() == 1):  
        disable_line.set_value(MOTION_DISABLED)  
        step_line.set_value(0)
```



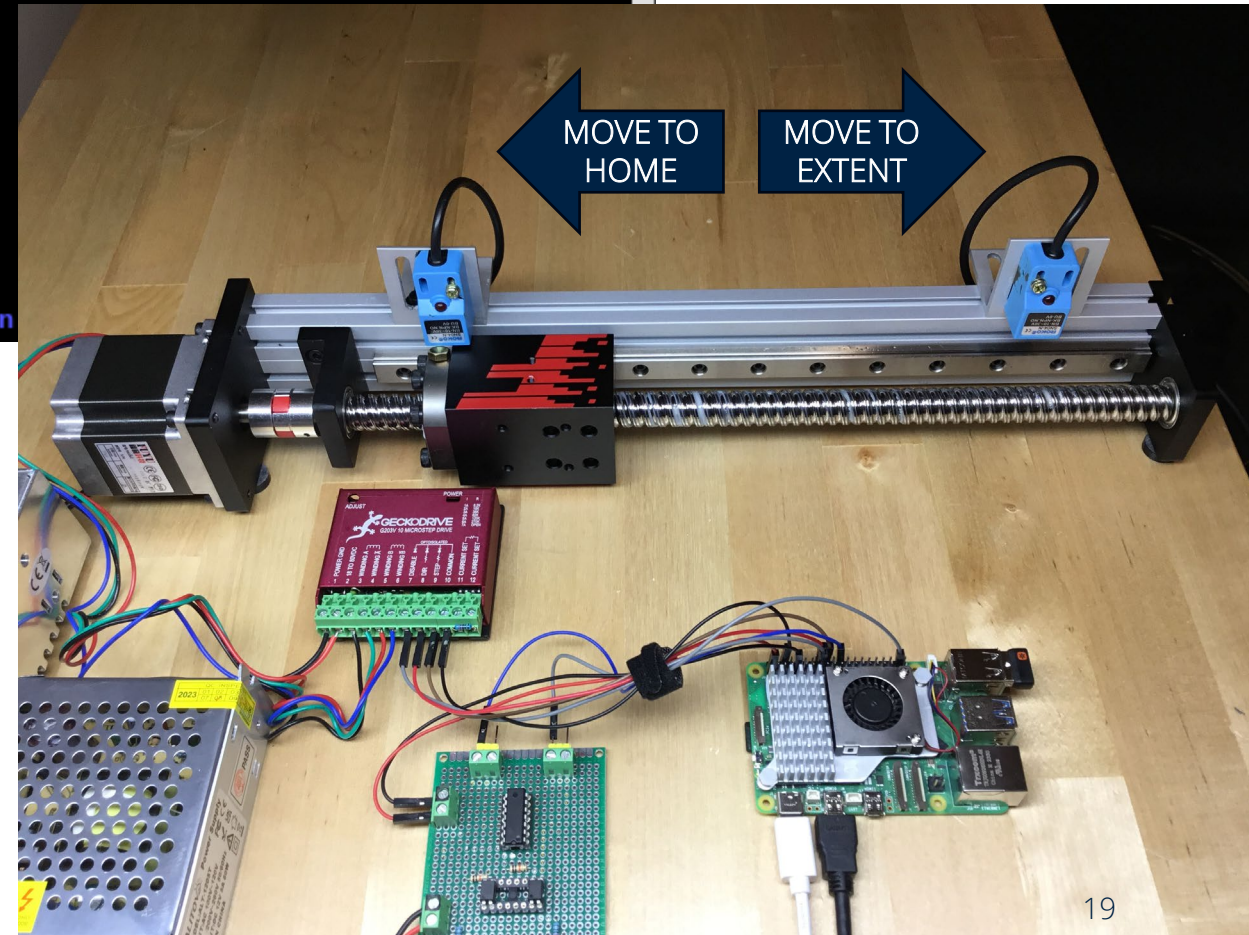
```
pi5user@pi5a: ~/pi5_apps_python  
File Edit Tabs Help  
pi5user@pi5a:~/pi5_apps_python $ python3 pi5_linear_stage.py  
Performing HOME operation  
HOME operation completed  
Move Forward 20000 steps  
Move Backwards 10000 steps  
Stopped!  
pi5user@pi5a:~/pi5_apps_python $
```



Linear Stage *pingpong* Function

```
def pingpong():
    fdirection = FORWARD
    try:
        while True:
            match fdirection:
                case moveConstants.forward:
                    moveToExtent()
                    fdirection = REVERSE
                case moveConstants.reverse:
                    moveToHome()
                    fdirection = FORWARD
    except KeyboardInterrupt:
        home()
        disable_line.set_value(MOTION_DISABLED)
        step_line.set_value(0)
```

```
pi5user@pi5a: ~/pi5_apps_python
File Edit Tabs Help
pi5user@pi5a:~/pi5_apps_python $ python3 pi5_linear_stage.py
Performing HOME operation
HOME operation completed
Moving To Extent
Moving To Home
Moving To Extent
Moving To Home
Moving To Extent
Moving To Home
^CPerforming HOME operation
HOME operation completed
Stopped!
pi5user@pi5a:~/pi5_apps_python
```



Next Time...

MORE TO COME..

# Thank you for attending!!!

Please consider the resources below:

- [Today's Download Package](#)
- [raspberrypi.org](http://raspberrypi.org) – Raspberry Pi 5



Linear Stage Videos – [www.edtp.com/pi5Automation/automation\\_videos\\_mp4.zip](http://www.edtp.com/pi5Automation/automation_videos_mp4.zip)





**DesignNews**

Thank You

Sponsored by

**DigiKey**

