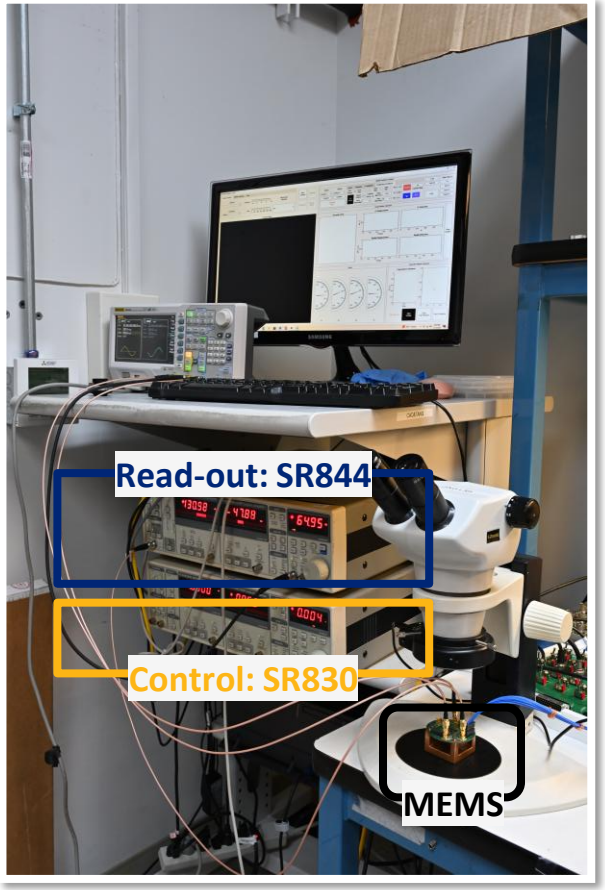
MEMStepper_DualLockIn Tutorial

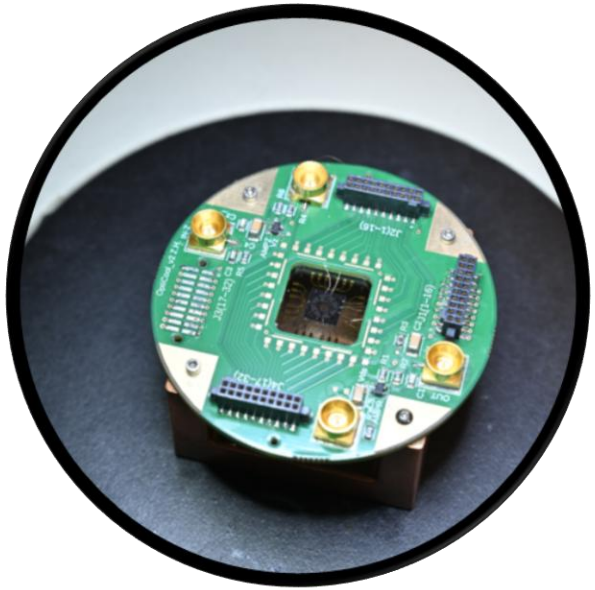
Version 1.0 Feb.19 2026
UC Berkeley-MIT Cao & Tang Lab

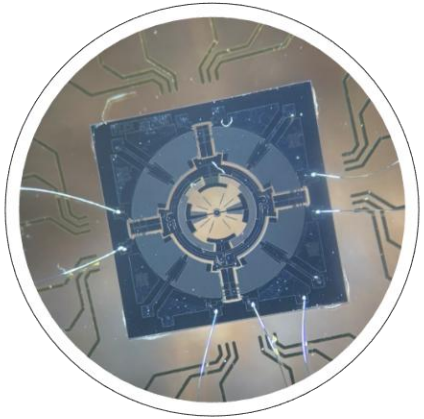
alexw25.us@berkeley.edu | alexw25.us@utexas.edu

UC Berkeley Engineering

MEMS Test Plantform







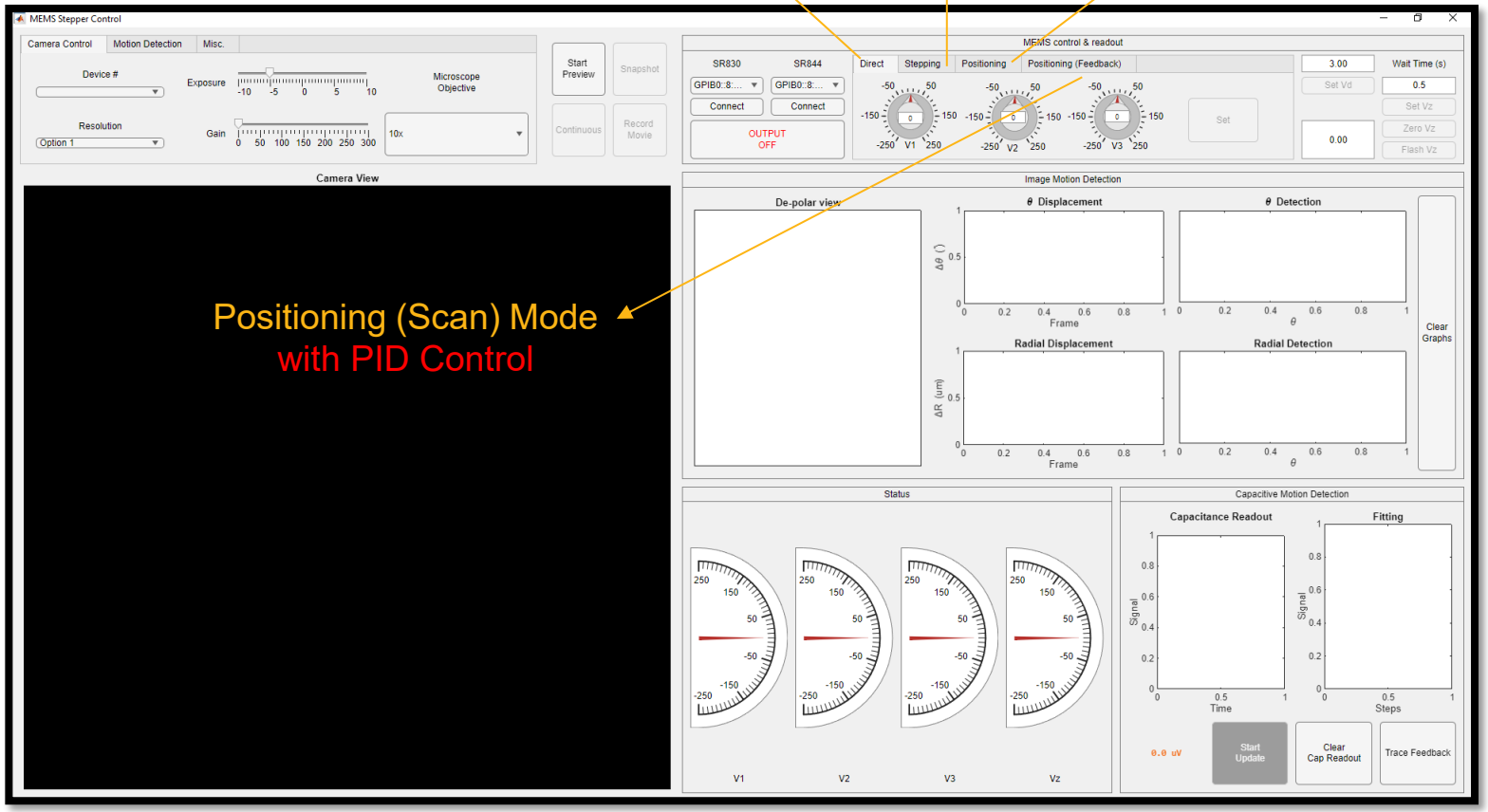
UC Berkeley
Engineering

1-MEMS Basic Control

Stepping Mode

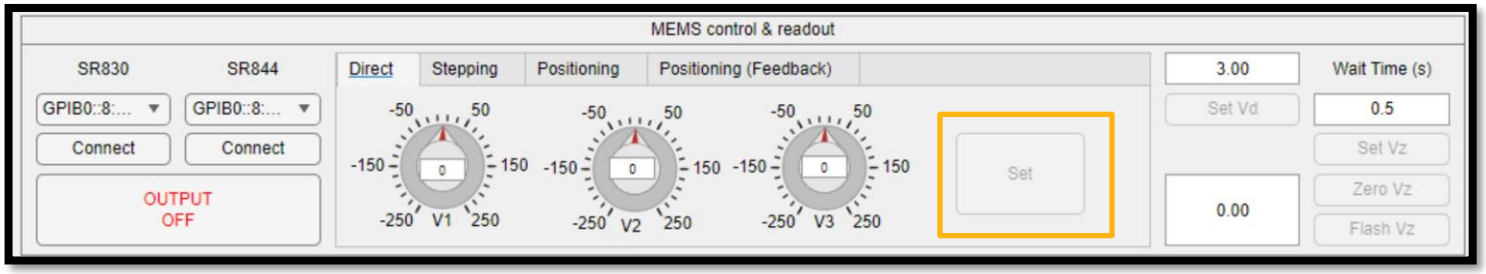
Direct Set Mode

Positioning (Scan) Mode



1-MEMS Basic Control

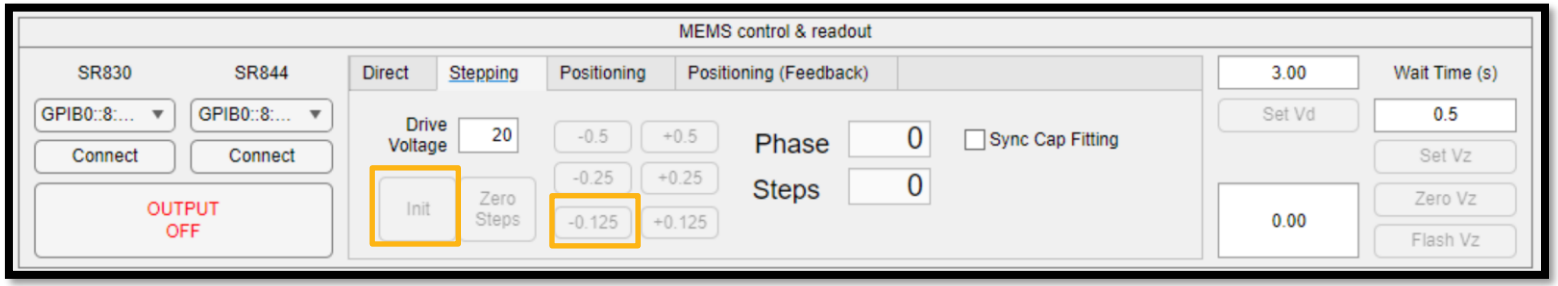
1-1 Direct Set Mode




```
% Button pushed function: SetButton
function SetButtonPushed(app, event)
    if isa(app.LockIn, 'visa') && strcmp(app.LockIn.Status,'open')
        fprintf(app.LockIn, sprintf('auxv 1,%f\n', app.V1Knob.Value/app.volt_amp_ratio));
        fprintf(app.LockIn, sprintf('auxv 2,%f\n', app.V2Knob.Value/app.volt_amp_ratio));
        fprintf(app.LockIn, sprintf('auxv 3,%f\n', app.V3Knob.Value/app.volt_amp_ratio));
        app.updateVoltages();
    end

    app.setSteppingEnable(false); % Stepping needs to be reinitialized
    app.setPositioningEnable(false); % Positioning needs to be reinitialized
end
```

1-MEMS Basic Control

1-2 Stepping Mode




```
% Button pushed function: InitButton
function InitButtonPushed(app, event)
    app.PhaseEditField.Value = 0;
    app.StepsEditField.Value = 0;
    app.setVoltages(app.DriveVoltageEditField.Value, app.PhaseEditField.Value);
    app.setSteppingEnable(true);
    if app.SyncCapFittingCheckBox.Value
        app.addCapFittingPoint();
    end
end
```

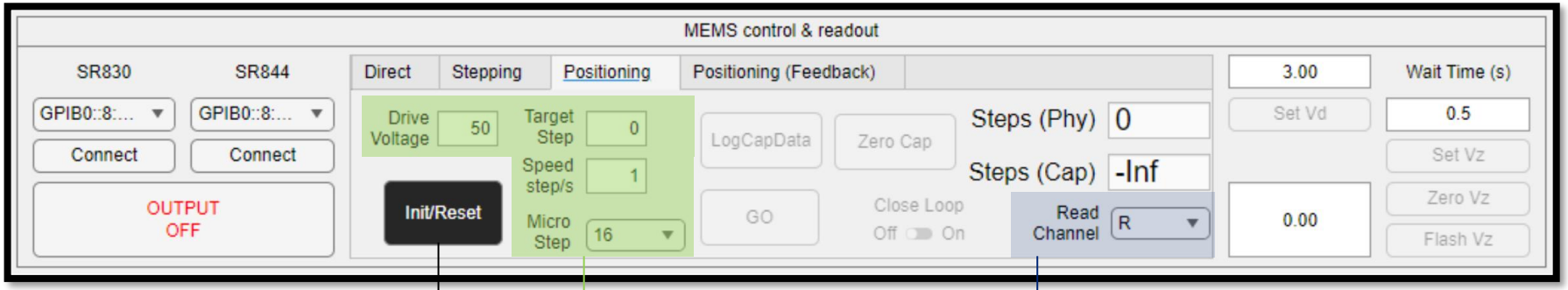
```
function setVoltages(app, volt, phase)
    if ~isa(app.LockIn, 'visa') || ~strcmp(app.LockIn.Status,'open')
        errordlg('USB connection not established');
        return;
    end
    fprintf(app.LockIn, sprintf('auxv 1,%f\n', volt/app.volt_amp_ratio*MEMStepper.vphase(phase/180*pi, 0)));
    fprintf(app.LockIn, sprintf('auxv 2,%f\n', volt/app.volt_amp_ratio*MEMStepper.vphase(phase/180*pi, -2*pi/3)));
    fprintf(app.LockIn, sprintf('auxv 3,%f\n', volt/app.volt_amp_ratio*MEMStepper.vphase(phase/180*pi, 2*pi/3)));

    app.updateVoltages();
end
```

-1/8

```
% Button pushed function: Minus18Button
function Minus18ButtonPushed(app, event)
    app.PhaseEditField.Value = mod(app.PhaseEditField.Value - 15, 360);
    app.StepsEditField.Value = app.StepsEditField.Value - 0.125;
    app.setVoltages(app.DriveVoltageEditField.Value, app.PhaseEditField.Value);
    if app.SyncCapFittingCheckBox.Value
        app.addCapFittingPoint();
    end
end
```

1-MEMS Basic Control

1-3 Positioning (Scan) Mode



(1) Initialize

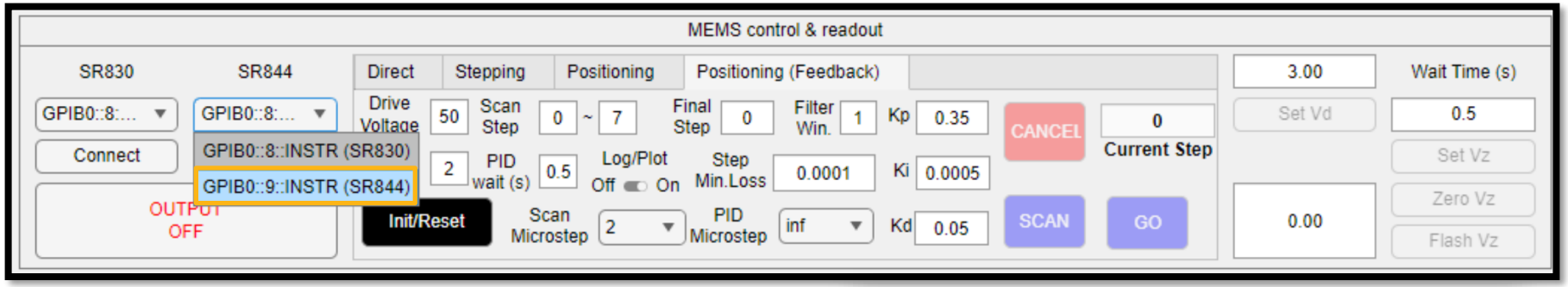
(2) Set Scan Params

(3) Cap Readout Channel

```
function goTo(app, volt, target, microstep, speed)

    curr_step = app.StepsPhyEditField.Value;
    total_steps = round(abs(target-curr_step)*(microstep));
    wait_time = 1/(speed * microstep);
    step_inc = 1/microstep*sign(target-curr_step);

    for i=1:total_steps
        curr_step = curr_step + step_inc;
        phase = curr_step * 120;

        app.setVoltages(volt, phase);
        pause(wait_time);
        if isa(app.video, 'videoinput') && isrunning(app.video)
            trigger(app.video);
        end
        app.StepsPhyEditField.Value = curr_step;
        if app.StepsCapZero > -100
            app.StepsCapEditField.Value = curr_step - app.StepsCapZero;
        end
    end
end
```

2-MEMS PID Control

2-1 [Set Point] Positioning (Scan) Mode with PID Control



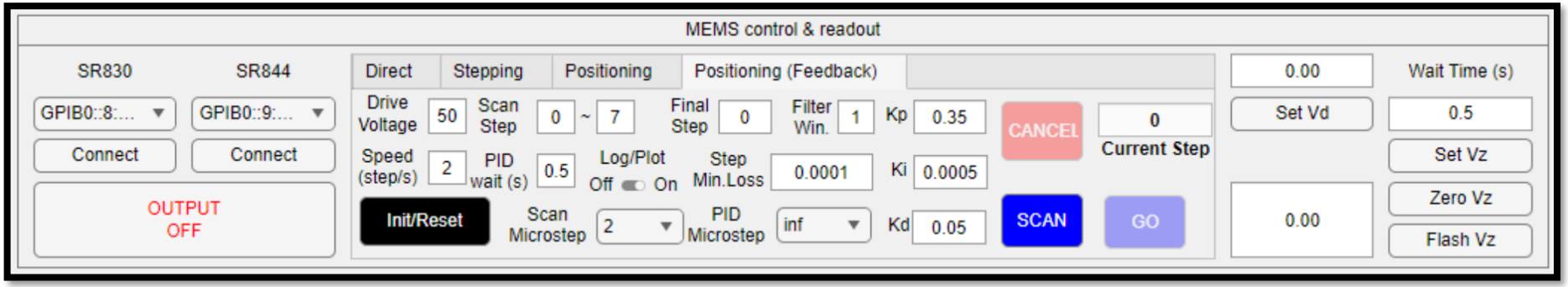
(1) Connect Lock-In First

```
LockIn       % VISA object for SR830 (typically for driving/control)
LockIn_Read  % VISA object for secondary Lock-In (SR844) for sensing
```

```
>> MEMStepper_DualLockIn
Warning: No devices were detected for the 'winvideo' adaptor. For troubleshooting device detection issues, click here.
Warning: instrfind will be removed in a future release. For objects of type
 - serialport use serialportfind
 - tcpclient use tcpclientfind
 - tcpserver use tcpserverfind
 - udpport use udpportfind
 - visadev use visadevfind
 - aardvark use aardvarkfind
 - ni845x use ni845xfind
 - icdevice with LegacyMode=false use icdevicefind
Warning: visa will be removed in a future release. Use visadev instead.

Note: To use visadev with icdevice, create an icdevice object with name-value argument LegacyMode=false.
[LOG] SR830 successfully connected.
      Resource: GPIB0::8::INSTR
      Status:   OPEN
      Instrument ID: Stanford_Research_Systems,SR830,s/n22628,ver1.01
Warning: instrfind will be removed in a future release. For objects of type
 - serialport use serialportfind
 - tcpclient use tcpclientfind
 - tcpserver use tcpserverfind
 - udpport use udpportfind
 - visadev use visadevfind
 - aardvark use aardvarkfind
 - ni845x use ni845xfind
 - icdevice with LegacyMode=false use icdevicefind
Warning: visa will be removed in a future release. Use visadev instead.

Note: To use visadev with icdevice, create an icdevice object with name-value argument LegacyMode=false.
[LOG] SR844 successfully connected.
      Resource: GPIB0::9::INSTR
      Status:   OPEN
      Purpose:  High-frequency Readout via OUTP? 1
      Instrument ID: Stanford_Research_Systems,SR844,s/n43265,ver1.006
```

2-MEMS PID Control

2-1 [Set Point] Positioning (Scan) Mode with PID Control



(2) Set Params

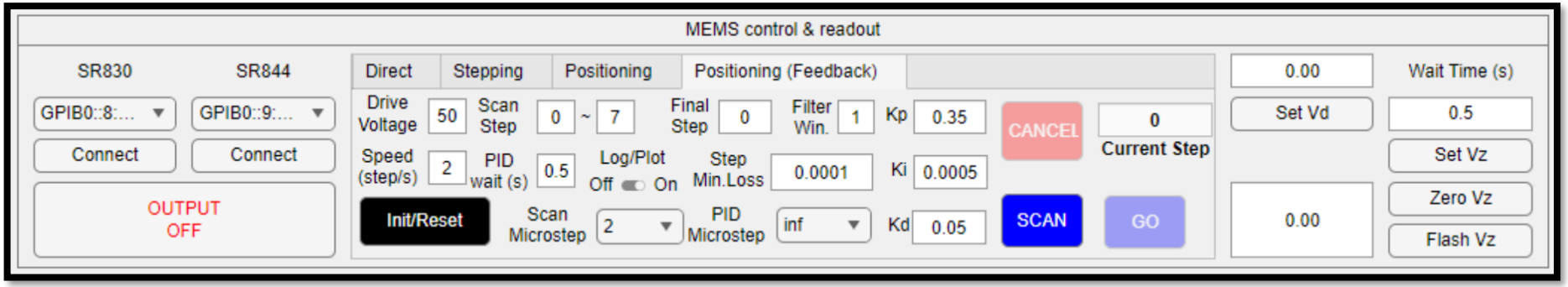
- **Scan Step 0~7** — Numeric Input start/stop step of calibration scan (physical step value)
- **Final Step** — Input final target step (position after scan calibration)
- **Filter Window** — Set the size of data filter window (number of data points). Used to smooth capacitance/step data (reduce noise) during **PID** positioning. **Recommended value: 1 (keep unchanged)**

- **Speed (step/s)** — Scan movement speed. unit: step/s, `wait_time = max(app.default_wait_time, 1/speed)`
- **PID wait (s)** — PID sampling wait time input box. **Recommended value: 1/Speed**
- **Log/Plot Switch** — Control real-time log plotting for PID feedback (capacitance-step curve). On: Plot curve + data points; Off: Only log data (no plot)
- **Step Min. Loss** — This (step_tolerance) is the minimum step error threshold for PID control. It prevents integral saturation by stopping the accumulation of the **Integral (I)** term when the actual step error is smaller than this value. **Recommended value: 0.0001 to keep Integral (I) always "On"**

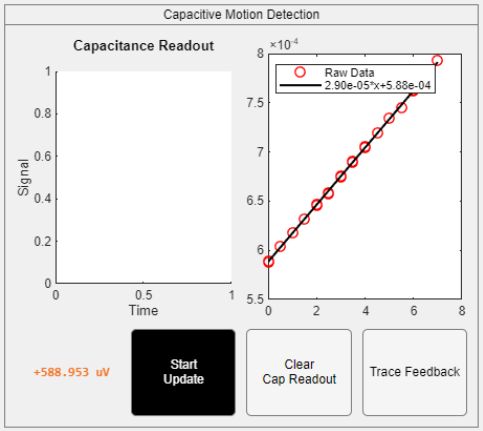
- **Scan Microstep** — Select microstep resolution for scan (2/4/8...256; higher = more precise positioning). **Recommended value: 2 to keep linear**
- **PID Microstep** — PID microstep resolution for feedback (2/4/8...256/**inf**; inf = continually moving). **Recommended value: inf**

2-MEMS PID Control

2-1 [Set Point] Positioning (Scan) Mode with PID Control



(3) Scan to Build Linear Model

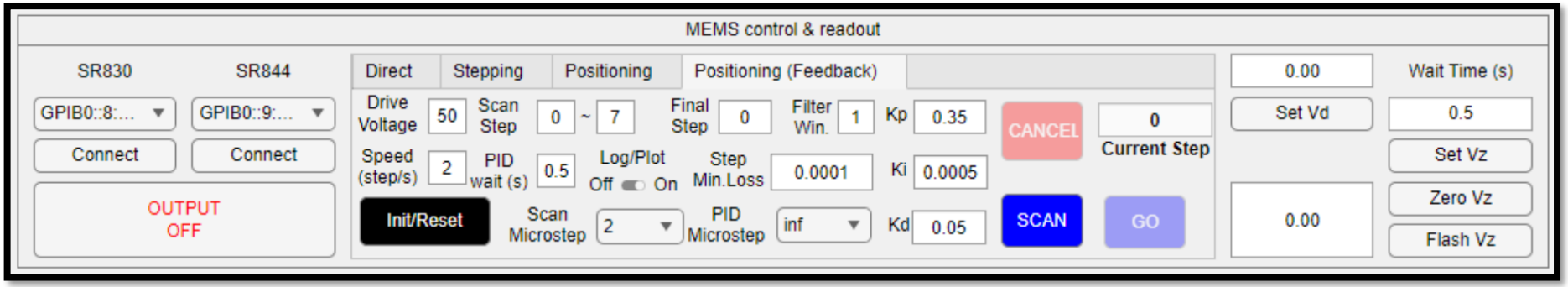



```
[LOG] goTo_fit (scan version):  wait_time=0.5000 s
[LOG] goTo_fit (scan version):  wait_time=0.5000 s
[LOG] SCAN_PART Channel                : R
[LOG] SCAN_PART_1 (Forward 15 points) : k1 = 2.903e-05, b1 = 5.881e-04
[LOG] SCAN_PART_2 (Backward 15 points): k2 = 2.895e-05, b2 = 5.885e-04
[LOG] SCAN_TOTAL  (Symmetric)         : k_total = 2.899e-05,    b_total = 5.883e-04
[LOG] HYSTERESIS_AVG: Mean Delta Cap  (dy) = 4.766e-07
[LOG] HYSTERESIS_AVG: Mean Delta_Step (dx) = 0.0164 steps
    (0.000000, 5.878450e-04)
    (0.500000, 6.037570e-04)
    (1.000000, 6.179570e-04)
    (1.500000, 6.319560e-04)
    (2.000000, 6.458540e-04)
    (2.500000, 6.581410e-04)
    (3.000000, 6.751610e-04)
    (3.500000, 6.898650e-04)
    (4.000000, 7.048700e-04)
    (4.500000, 7.188690e-04)
    (5.000000, 7.347810e-04)
    (5.500000, 7.448520e-04)
    (6.000000, 7.630810e-04)
    (6.500000, 7.758710e-04)
    (7.000000, 7.929920e-04)
    (7.000000, 7.930920e-04)
    (6.500000, 7.764750e-04)
    (6.000000, 7.624770e-04)
    (5.500000, 7.446510e-04)
    (5.000000, 7.337740e-04)
    (4.500000, 7.193730e-04)
    (4.000000, 7.052730e-04)
    (3.500000, 6.902670e-04)
    (3.000000, 6.749590e-04)
    (2.500000, 6.578390e-04)
    (2.000000, 6.465590e-04)
    (1.500000, 6.317550e-04)
    (1.000000, 6.176550e-04)
    (0.500000, 6.042610e-04)
    (0.000000, 5.889530e-04)
```

Lower is better

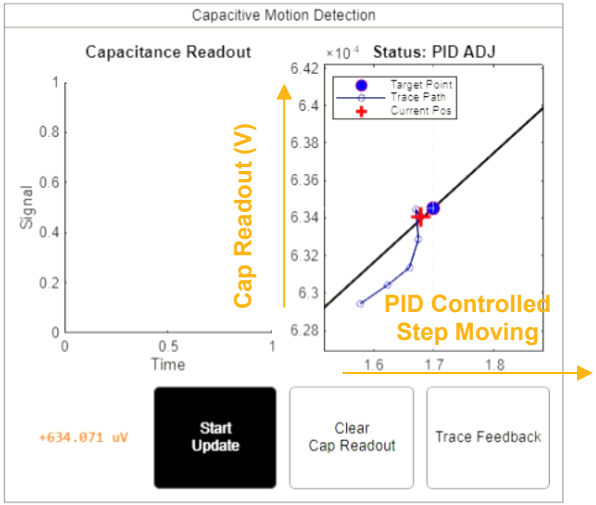
2-MEMS PID Control

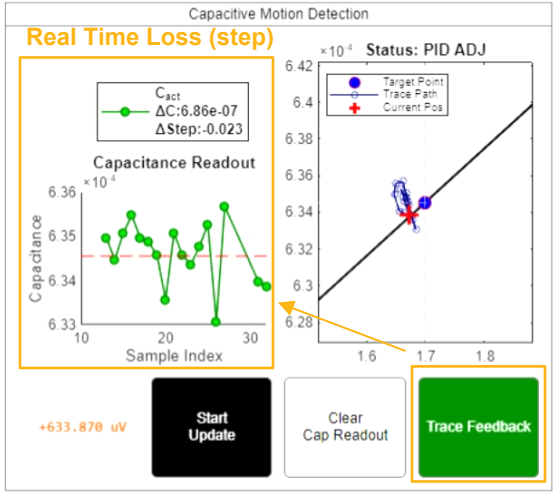
2-1 [Set Point] Positioning (Scan) Mode with PID Control



(4) PID Positioning

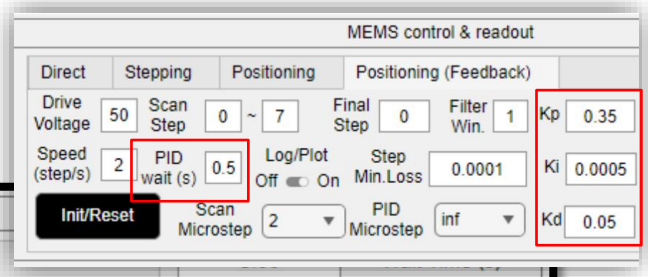
```
Cap(theor-act): [  5.919e-04 -   5.880e-04 =   3.891e-06]
Cap(theor-act): [  5.919e-04 -   5.902e-04 =   1.776e-06]
Cap(theor-act): [  5.919e-04 -   5.918e-04 =   1.639e-07]
Cap(theor-act): [  5.919e-04 -   5.920e-04 =  -3.709e-08]
Cap(theor-act): [  5.919e-04 -   5.913e-04 =   6.679e-07]
Cap(theor-act): [  5.919e-04 -   5.918e-04 =   1.639e-07]
Cap(theor-act): [  5.919e-04 -   5.920e-04 =  -3.709e-08]
Cap(theor-act): [  5.919e-04 -   5.922e-04 =  -2.381e-07]
Cap(theor-act): [  5.919e-04 -   5.913e-04 =   6.679e-07]
Cap(theor-act): [  5.919e-04 -   5.918e-04 =   1.639e-07]
Cap(theor-act): [  5.919e-04 -   5.916e-04 =   3.659e-07]
Cap(theor-act): [  5.919e-04 -   5.924e-04 =  -4.401e-07]
Cap(theor-act): [  5.919e-04 -   5.920e-04 =  -3.709e-08]
Cap(theor-act): [  5.919e-04 -   5.923e-04 =  -3.391e-07]
Cap(theor-act): [  5.919e-04 -   5.919e-04 =   6.391e-08]
Cap(theor-act): [  5.919e-04 -   5.911e-04 =   8.689e-07]
Cap(theor-act): [  5.919e-04 -   5.922e-04 =  -2.381e-07]
Cap(theor-act): [  5.919e-04 -   5.919e-04 =   6.391e-08]
Cap(theor-act): [  5.919e-04 -   5.913e-04 =   6.679e-07]
Cap(theor-act): [  5.919e-04 -   5.923e-04 =  -3.391e-07]
Cap(theor-act): [  5.919e-04 -   5.920e-04 =  -3.709e-08]
Cap(theor-act): [  5.919e-04 -   5.916e-04 =   3.659e-07]
>>
```

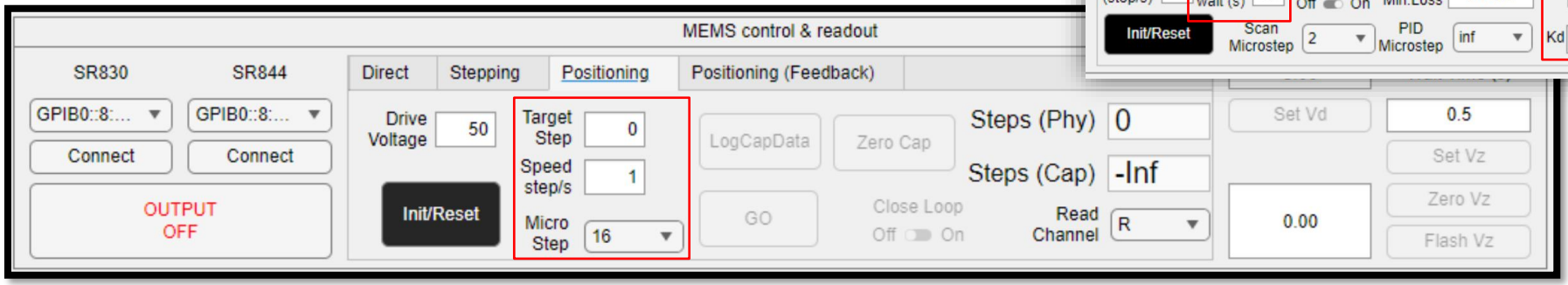




2-MEMS PID Control

2-3 [Scan Point] Positioning (Scan) Mode with PID Control



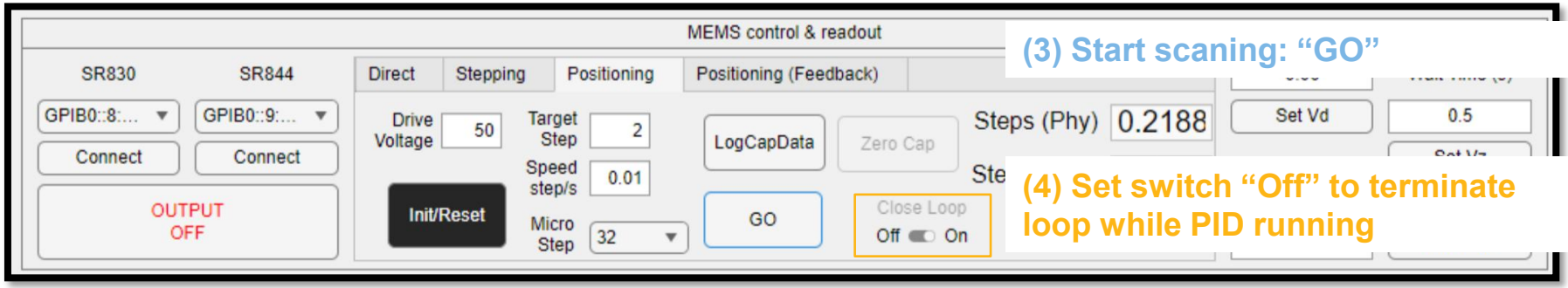


(1) Set Params

- **Speed (step/s)** defines the movement time of each step (eg. from "2" to "3")
- **PID wait time (s)** the delay for PID sampling
- **Micro Step** Microstep count within each Step

(2) Set switch "On" to enable PID

- **Total time per step = 1/speed**
- **Total time per micro step = 1/(speed*Micro Step)**
- **Total PID sample count for each micro step = Total time per micro step / PID wait time**



(3) Start scaning: "GO"

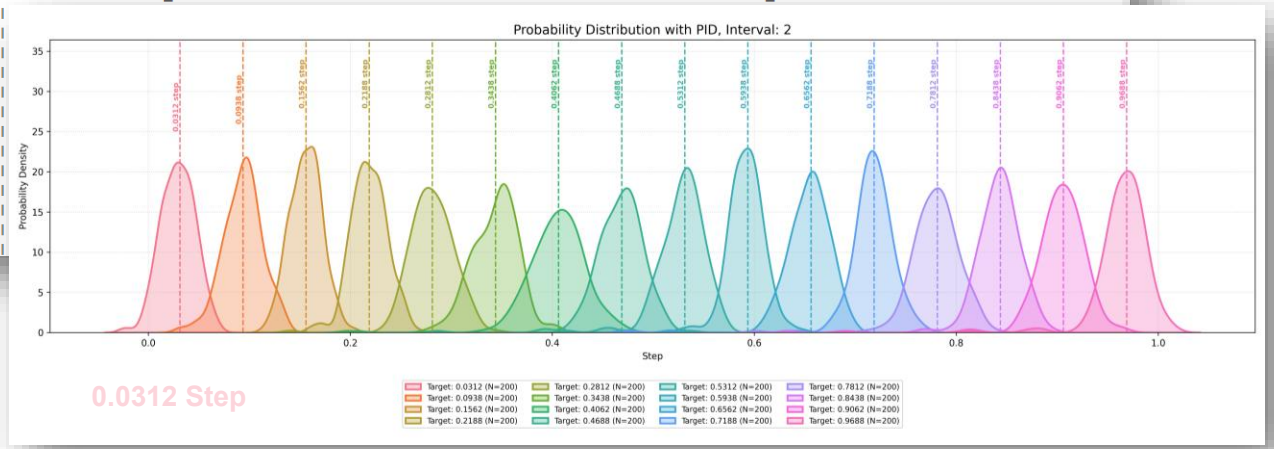
(4) Set switch "Off" to terminate loop while PID running

2-MEMS PID Control

2-3 [Scan Point] Positioning (Scan) Mode with PID Control

Save “mems.log” for data plot/analysis

```
Cap(theor-act): [  5.920e-04 -   5.920e-04 =   2.347e-09] | Step Loss (=Cap/k_fit): [  0.000] | Gains(P,I,D): [3.80e-01, 0.00e+00, 1.00e-01] | PID_Out(P+I+D): [ 0.0000 + -0.0000 +  0.0007
Cap(theor-act): [  5.920e-04 -   5.921e-04 =  -9.865e-08] | Step Loss (=Cap/k_fit): [ -0.003] | Gains(P,I,D): [3.80e-01, 0.00e+00, 1.00e-01] | PID_Out(P+I+D): [-0.0013 + -0.0000 + -0.0007
Cap(theor-act): [  5.920e-04 -   5.915e-04 =   5.063e-07] | Step Loss (=Cap/k_fit): [  0.017] | Gains(P,I,D): [3.80e-01, 0.00e+00, 1.00e-01] | PID_Out(P+I+D): [ 0.0066 + -0.0000 +  0.0041
Cap(theor-act): [  5.920e-04 -   5.923e-04 =  -2.997e-07] | Step Loss (=Cap/k_fit): [ -0.010] | Gains(P,I,D): [3.80e-01, 0.00e+00, 1.00e-01] | PID_Out(P+I+D): [-0.0039 + -0.0000 + -0.0055
Cap(theor-act): [  5.920e-04 -   5.921e-04 =  -9.865e-08] | Step Loss (=Cap/k_fit): [ -0.003] | Gains(P,I,D): [3.80e-01, 0.00e+00, 1.00e-01] | PID_Out(P+I+D): [-0.0013 + -0.0000 +  0.0014
Cap(theor-act): [  5.920e-04 -   5.920e-04 =   2.347e-09] | Step Loss (=Cap/k_fit): [  0.000] | Gains(P,I,D): [3.80e-01, 0.00e+00, 1.00e-01] | PID_Out(P+I+D): [ 0.0000 + -0.0000 +  0.0007
Cap(theor-act): [  5.920e-04 -   5.925e-04 =  -5.017e-07] | Step Loss (=Cap/k_fit): [ -0.017] | Gains(P,I,D): [3.80e-01, 0.00e+00, 1.00e-01] | PID_Out(P+I+D): [-0.0065 + -0.0000 + -0.0034
Cap(theor-act): [  5.920e-04 -   5.913e-04 =   7.073e-07] | Step Loss (=Cap/k_fit): [  0.024] | Gains(P,I,D): [3.80e-01, 0.00e+00, 1.00e-01] | PID_Out(P+I+D): [ 0.0092 + -0.0000 +  0.0083
[LOG] Appended 200 points to capdata.log @ 2026-02-18 19:46:29.443288
[LOG] Loged successfully @ 2026-02-18 19:46:29.443288
[LOG] Feedback loop completed for step 7/64. Moving to next step.

[LOG] Starting feedback loop with total_cnt=200 for step 8/64 (scan_loop_step=2.500e-01).
Cap(theor-act): [  5.929e-04 -   5.924e-04 =   5.150e-07] | Step Loss (=Cap/k_fit): [  0.018] | Gains(P,I,D): [3.80e-01, 0.00e+00, 1.00e-01] | PID_Out(P+I+D): [ 0.0067 + -0.0000 + -0.0013
Cap(theor-act): [  5.929e-04 -   5.947e-04 =  -1.801e-06] |
Cap(theor-act): [  5.929e-04 -   5.926e-04 =   3.140e-07] |
Cap(theor-act): [  5.929e-04 -   5.939e-04 =  -9.960e-07] |
Cap(theor-act): [  5.929e-04 -   5.938e-04 =  -8.950e-07] |
Cap(theor-act): [  5.929e-04 -   5.926e-04 =   3.140e-07] |
Cap(theor-act): [  5.929e-04 -   5.928e-04 =   1.120e-07] |
Cap(theor-act): [  5.929e-04 -   5.925e-04 =   4.140e-07] |
Cap(theor-act): [  5.929e-04 -   5.929e-04 =   1.202e-08] |
Cap(theor-act): [  5.929e-04 -   5.933e-04 =  -3.910e-07] |
Cap(theor-act): [  5.929e-04 -   5.932e-04 =  -2.910e-07] |
Cap(theor-act): [  5.929e-04 -   5.918e-04 =   1.119e-06] |
Cap(theor-act): [  5.929e-04 -   5.933e-04 =  -3.910e-07] |
Cap(theor-act): [  5.929e-04 -   5.932e-04 =  -2.910e-07] |
```



mems.log

3-More

3-1	**Script**	**(Matlab R2024b)**	MEMStepper_DualLockIn.m
	App	**(Matlab R2025b)**	MEMStepper_DualLockIn.mlapp

```
% Create ImageMotionDetectionPanel
% app.ImageMotionDetectionPanel = uipanel(app.GridLayout);
% app.ImageMotionDetectionPanel.TitlePosition = 'centertop';
% app.ImageMotionDetectionPanel.Title = 'Image Motion Detection';
% app.ImageMotionDetectionPanel.Layout.Row = 2;
% app.ImageMotionDetectionPanel.Layout.Column = [3 4];
```

```
% [FIX] The modified code (the second section with [FIX] annotations) follows the constraint of MATLAB R2024 and ensures compatibility with both versions:
% 1. First, create the uipanel and directly bind it to the parent app.GridLayout at the time of, while setting basic properties simultaneously.
app.ImageMotionDetectionPanel = uipanel(app.GridLayout, 'Title', 'Image Motion Detection', 'BorderType', 'line', 'TitlePosition', 'centertop');
% 2. After the uipanel is successfully created and associated with its parent GridLayout
%    set the Layout.Row and Layout.Column properties to define the component's position in the GridLayout.
app.ImageMotionDetectionPanel.Layout.Row = 2;
app.ImageMotionDetectionPanel.Layout.Column = [3 4];
```

3-2	**Github**	https://github.com/CHENfd515/MEMStepper_DualLockIn.git
3-2	**Reference Article**	An adaptive moiré sensor for spectro-polarimetric hyperimaging


THE UNIVERSITY OF CALIFORNIA BERKELEY
LET THERE BE LIGHT
1868