

# Cantilevered-Capillary Force Apparatus User's Manual

Last updated: 11 April 2019

# Contents

|          |  |          |
|----------|--|----------|
| <b>1</b> | <b>Hardware</b>  | <b>3</b> |
| 1.1      | Creation of Cantilever . . . . .                           | 3        |
| 1.1.1    | Potential Problems . . . . .                               | 3        |
| 1.2      | Calibration of Cantilever . . . . .                        | 3        |
| 1.2.1    | Potential Problems . . . . .                               | 4        |
| 1.3      | Position Sensitive Photodiode Sensor Calibration . . . . . | 4        |
| 1.3.1    | Procedure . . . . .  | 4        |
| 1.3.2    | Potential Problems . . . . .                               | 5        |
| 1.4      | Microscope calibration . . . . .                           | 5        |
| 1.5      | PI E816/E625 Piezo Stage . . . . .                         | 6        |
| 1.5.1    | Troubleshooting . . . . .                                  | 6        |
| 1.6      | Thorlabs Stepper Motor . . . . .                           | 6        |
| 1.7      | Thorlabs Position Sensitive Photo Detector . . . . .       | 6        |
| 1.8      | National Instruments USB 6009 DAQ Card . . . . .           | 7        |
| 1.9      | Vision Research – Phantom High Speed Camera . . . . .      | 7        |
| 1.10     | Grabber for Side-View Camera . . . . .                     | 7        |
| <b>2</b> | <b>Software</b>  | <b>7</b> |
| 2.1      | MATLAB . . . . .   | 7        |
| <b>3</b> | <b>Measurements</b>  | <b>7</b> |
| 3.1      | Drops . . . . .  | 7        |
| 3.2      | Creating Drops . . . . .                                   | 7        |
| 3.3      | Monitor Volume . . . . .                                   | 8        |
| 3.4      | Interfacial Tension . . . . .                              | 8        |
| 3.5      | Constant Force . . . . .                                   | 8        |
| 3.6      | Buckling . . . . .   | 9        |
| 3.7      | Elastic modulus for capsules . . . . .                     | 9        |
| 3.8      | New Experiment Design . . . . .                            | 9        |

# 1 Hardware

## 1.1 Creation of Cantilever

Shield a new piece of capillary with several layers of aluminum foil leaving 2 mm of the tip exposed. The foil will protect the polymer coating. Hold the capillary and foil in a pair of needle-nose pliers or something similar so that the edge of the foil with the capillary tip is flush with the side of the pliers. Heat the tip in a torch and bend the tip to a 90° angle. Allow the capillary to cool. Glue a small piece of polished silicon wafer to the end as shown in Figure 1. The mirror can be held by applying a vacuum suction through a plastic pipette tip than can then be secured on adjustable stages and the capillary can be held using the clamping mechanism from the CCFA. Note that the mirror can extend beyond the end of the capillary. Be careful to glue the mirror so that the reflective surface is orthogonal to the bent portion of the capillary.



Figure 1: Cantilevered-Capillary with mirror attached.

### 1.1.1 Potential Problems

The thinner capillaries may have some natural curvature along their length. If this is the case try to bend the tip in the direction that will result in the weight of the mirror counteracting the curvature to make the capillary less curved in the CCFA.

## 1.2 Calibration of Cantilever

Adjust the cords to suspend and isolate the instrument from surrounding noise. To calibrate the spring constant, the resonance frequency needs to be measured as a function of the free length of the capillary. To do this, the capillary needs to be placed in the clamp which is loosened with the white screw without connecting it to the tubing or placing it in the CCFA chamber. Adjust the 3-D positioning arm so that the capillary is outside of the chamber and place the capillary in arm, but within reach of the laser (as shown in Figure 2). Reflect the laser off of the mirrors by changing their positions (height and angle) and onto the position sensitive photodiode. Open the file named "CCFA" in MATLAB and run the "Measure Noise" operating mode with the following steps:

1. Settings: Sample Rate - 500 Hz; Time - 5 s; Iterations - 1; Enable detector (make sure that the signal is strong - red, and fairly centered)
2. Measure back length using a 0.5mm precision ruler (from the metal holder to the end of the capillary  $\pm 0.125$  mm)
3. Tap the microscope instrument gently and then immediately start the measurement
4. Change back length and measure the frequency again until you have 6 to 10 data points. The position of the mirrors need to be changed accordingly.
5. Change operating mode to "Spring Constant"
6. Enter the data in the appropriate boxes separated by commas or spaces.

7. Set the capillary dimensions for outer diameter (OD) and inner diameter (ID) separated by a period (e.g. 360.100)
8. Enter a starting guess for the effective mass and reference length
9. Start the measurement and check the fit. If the fit doesn't seem to be reasonable, try a different starting guess.

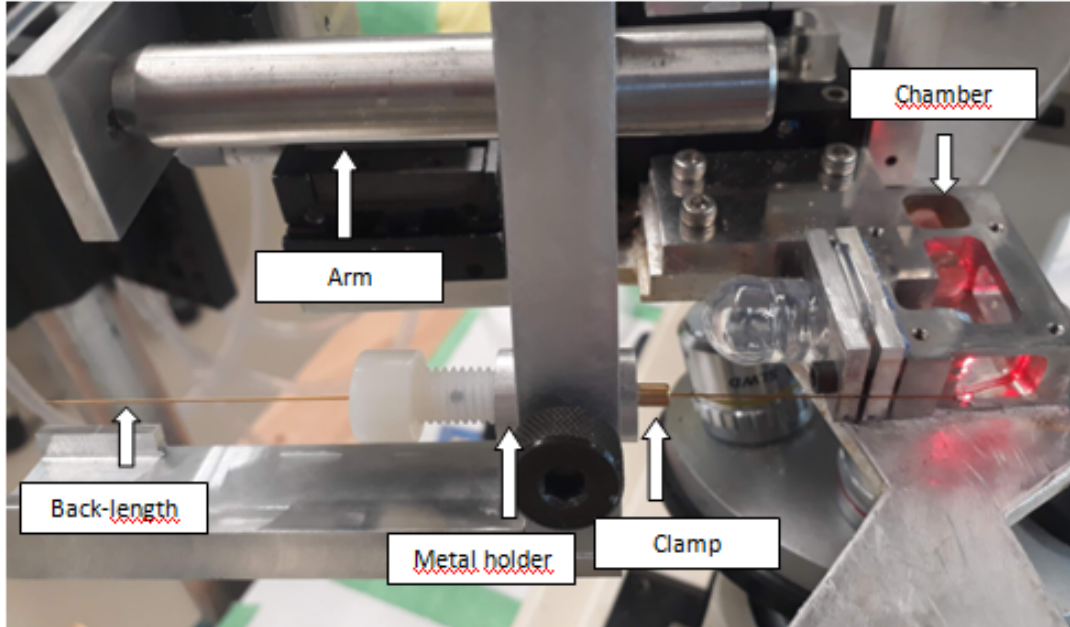


Figure 2: Cantilevered-capillary clamped outside the chamber

After these calibration steps, the cantilever should be placed into the CCFA chamber and clamped at the desired length as shown in Figure 3. To do this put the threaded metal holder with the white screw (part 1) in the 3-D positioning arm and secure it with the thumbscrew. Next insert the tip of the clamp into the silicone nipple (part 2) with the nipple in the nipple holder, but not connected to the chamber (not as pictured in Figure 3). Next untighten the clamp using the white screw and insert the cantilevered capillary through the nipple and part 1 taking care not to damage the tip, then tighten with the desired back-length. Finally, carefully rotate the 3-D positioning arm, slide the cantilever into the chamber and screw the nipple holder onto the chamber. Note, the 3-D positioning stages for the cantilevered capillary should be centered before the capillary is moved into the chamber. Finally attach the tubing adjustable (part 3) to the back of the cantilever. With the cantilever now in the chamber and set to the desired length, the spring constant can be calculated from the "Spring constant" mode of the CCFA interface, from the theoretical formula using the effective mass and reference length found during the calibration steps.

### 1.2.1 Potential Problems

If there is more than a single resonance frequency the cantilever may be contacting the inner wall of part 1.

## 1.3 Position Sensitive Photodiode Sensor Calibration

### 1.3.1 Procedure

First, slide the rigid capillary through the glass cylinder and through the mechanical pencil in the piezo-stage arm. Be careful not to rub the capillary against the sharp edges of the chamber. Position the rigid capillary in the chamber so that it is visible in the microscope and then use the 3-D positioning stages to align the tip of the cantilevered-capillary with the tip of the rigid capillary vertically and horizontally. The vertical alignment should be done by visualizing the tips

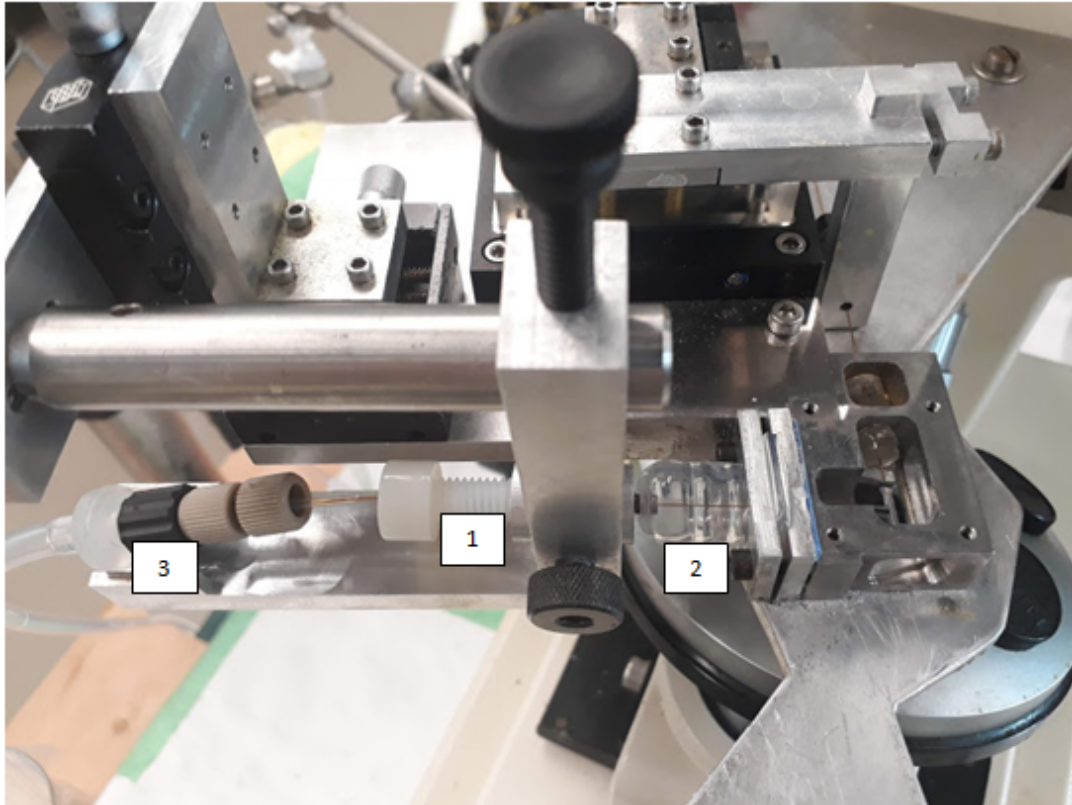


Figure 3: Cantilevered-capillary in clamping mechanism

through the side-view camera and rotating the metal clamp holder. With the tips aligned, and the laser reflected from the cantilever mirror to the the photo-diode detector, run the the CCFA in "Calibrate" operating mode. It is the easiest to start the calibration with the tips just barely in contact.

The calibration needs to be done before every experiment and whenever the fluids are changed or the path length of the laser is changed. The laser signal should be adjusted so that it is offset horizontally from the middle when the Calibration code is started so that the laser moves across the middle of the sensor during calibration. Typical settings are:

1. Sample Rate – 10,000 Hz;
2. Samples – 2000;
3. Calibration Distance - 5  $\mu\text{m}$ ;
4. Calibration Points – 20.

The calibration should be performed several times to ensure that the value of the conversion (microns per volt or MPV) is not changing with time.

### 1.3.2 Potential Problems

Air trapped in the opening of the metal cylinder that enters the silicone nipple may cause problems with the calibration. Be sure to remove all air from the metal cylinder if there is a problem getting a consistent value of the MPV.

## 1.4 Microscope calibration

When images from the top camera are used for calculations, the pixels of the images are converted to micrometers. For this, a conversion has been defined in the code but may need updating in case of new hardware. To do so, images of a micrometer slide are taken at all different magnifications. These images can then be successively loaded in the "MicroscopeCalibration.mat"

file, which will trace the boundaries of two lines of the micrometer and give the number of pixels between those lines, from which a meters per pixel ratio can be calculated. It will be necessary to modify the starting column in the "getEdgePointsLine.mat" function for the code to find the micrometer lines according to its position in the image.

## 1.5 PI E816/E625 Piezo Stage

To be able to control the peizo stage through Matlab a class definition m-file is needed which interfaces with the DLL files for the control. A class file was obtained from PI and modified to contain a few extra control features like constant velocity motion (see PIStage.m file). A matlab object is used to control the stage through object oriented programming style (ex: PS.qPOS() gets the current position of the stage). Another m-file was written to make an instance of the PI stage object (see 'PIStageObject.m' in Mat).

### 1.5.1 Troubleshooting

**CAUTION:** This device cost over \$10,000 and you should be very careful not to damage it. If you are not sure what you are doing then you should call PI for assistance.

#### **MATLAB Error: Piezo Stage not detected. No USB Connection Detected**

Possible Solutions:

1. USB cable may be unplugged
2. Occasionally, if Matlab crashes or communication to the stage is interrupted abnormally it may be necessary to power cycle the stage. This should be done by unplugging the black power cord from the back of the grey PI Stage controller box and then plugging it back in and restart Matlab. This should only need to be done once. If a single power cycle doesn't solve the problem then something else is wrong.

There is some kind of bug related to Windows 7 that causes USB communications errors with this device if plugged into a regular USB port. A workaround is to plug it into the USB 3.0 card and it seems to work.

## 1.6 Thorlabs Stepper Motor

**CAUTION:** Never apply power to the controller by plugging the power cord directly into the controller. Always plug the cord into the controller first and then into the wall. Doing this in the wrong order will damage the controller.

The active-x control that communicates with the stepper motor controller is very unstable and prone to crashing when used via the Matlab CCFA GUI. A consequence of this is a crash of MatLab every time the detector is enabled. Therefore it may be necessary to power cycle the controller to reset and reestablish communication. **THIS SHOULD NEVER BE DONE BY UNPLUGGING THE POWER CORD FROM THE CONTROLLER!!!** That will damage the device Power cycling should be done by unplugging the power cord from the wall and **THE CORD SHOULD ALWAYS BE PLUGGED INTO THE CONTROLLER BEFORE POWER UP THE CONTROLLER.** In extreme cases it may also become necessary to reset the firmware on the controller using the firmware update utility that comes with the installation of the APT software. After the power cycling the detector will be in 'open loop' modus. It is necessary to change it to 'monitr' modus in order to do experiments.

## 1.7 Thorlabs Position Sensitive Photo Detector

??

## 1.8 National Instruments USB 6009 DAQ Card

To use this hardware I installed the NI-DAQmx 9.5.1 drivers/software from the National Instruments website.

See the following help page for information on communicating with the DAQ card: [MathWorks Help Page](#) Use the analoginput function in Matlab to establish control of the DAQ card via a COM object.

There is some kind of bug related to Windows 7 that causes USB communications errors with this device if plugged into a regular USB port. A workaround is to plug it into the USB 3.0 card and it seems to work.

## 1.9 Vision Research – Phantom High Speed Camera

There are two ways of using the camera. One is using the PCC software that is written by the company. The other way is through Matlab. By registering on the Vision Research website you can get access to the “Developer Zone” to see discussion forums on how to use this and post questions if you have them.

Use the functions related to DLL libraries to communicate with the camera (loadlibrary, lib-functionsview, calllib, libpointer).

## 1.10 iGrabber for Side-View Camera

NOTE: This device does not seem to be working on Windows 7. It works just fine on Windows XP so it is not the hardware that is the problem. Updating the driver does not help. There appears to be something wrong with the USB communication. When the iGrabber is plugged in the blue light and the red light turn on. When it is working properly, the red light turns off. I am using the device with the old computer right now. I recommend that someone find a better way to do this with the new computer.

The side-view camera is accessed through the iGrabber USB Analog to Digital video converted. The video can be previewed using Image Acquisition Toolbox in Matlab. From the Matlab command line type “imaqtool” and a window will open that allows you to preview the video for a close-up view of the capillaries from the side. Normally you should choose the 640 x 480 resolution option from the list and then click “Start Preview.”

# 2 Software

## 2.1 MATLAB

Run MATLAB 2009, open file ccfa.m. User interface: cameras shown through preview side/top view. Enable detector control turn on. Select difference. This allows the user to detect where the laser is on the sensor. Use the up-down button to control the position of the piezo stage and the position of the cantilevered capillary. Monitoring Calibration

# 3 Measurements

## 3.1 Drops

## 3.2 Creating Drops

The easiest way to create drops is using gravity. Before doing experiments the point of equilibrium should be found. This point should be when the syringes are at the same height as the chamber. For growing the drops the plungers can be used. For shrinking the drops the syringes can be lowered under the level of the chamber. If the drops are not reacting immediately to a change in the height of the syringes, there must be air in the capillary or in one of the pieces that are holding the capillary. After every change of the capillary liquid should be flowing through the capillary for 30 minutes or an hour. If this is not helping every piece should be unscrewed and filled with liquid.

### 3.3 Monitor Volume

During an experiment with drops there shouldn't be any change in the volume of the drops and the drops on both capillaries should be around the same size. To test this you can check the volume in the "Monitor Volume"-mode. It allows you to monitor the volume of either one or both drops. Be careful to enter the right magnification. You can also choose between a monitoring time or a continuous measurement. When you start the measurement it will display the change of volume with the time for the choosen drops (left one in the upper box) and also their radii. In order to prevent crashes of MatLab the waiting time should not be set to 0. It is better to enter a small value like 0.1.

### 3.4 Interfacial Tension

In order to measure the interfacial tension there should be one drop between the two capillaries (figure 3) with a round shape. The code will automatically create this round shape after the initial cropping. You need to choose the magnification, the sample rate and the amount of samples. You can change the distance that the rigid capillary is moving during the experiment and also the step size. As the temperature has a major effect on the interfacial tension, you need to enter the actual temperature before the measurement in order to save it. Once the measurement is started, you will be asked to crop the image around the drop, and two blue circles will show up at the boundaries of the drop. These should overlap and their centers should be visible as yellow markers. The drop will then be made into a sphere shape and you will be asked to click at the four points where the drop comes into contact with both capillaries. This should be done first for the left capillary then for the right one. There should be a slight change in the shape of the drop during the experiment. Once the measurement is underway, there should be displayed a linear decrease in the force with the change of position. You can read out the value for the interfacial tension. At the same time the spring constant for this drop will be saved with the data file.

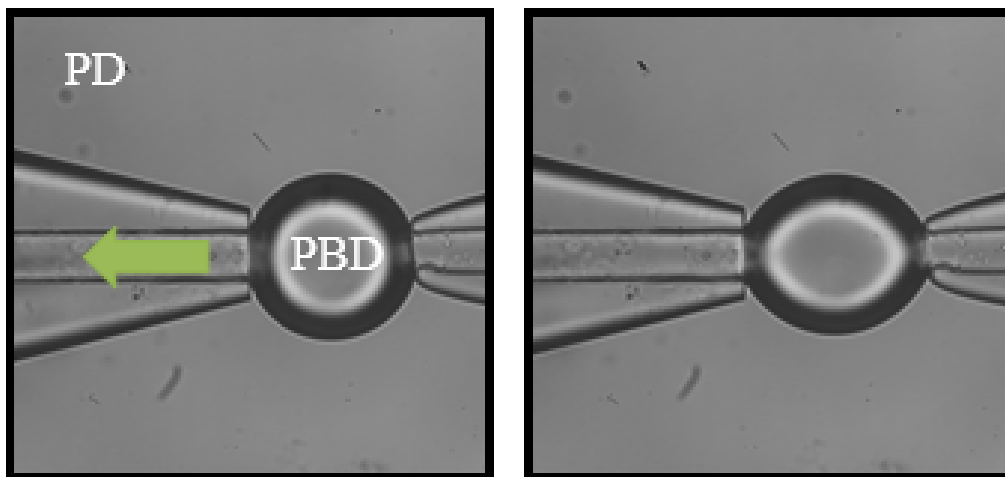


Figure 4: Measurement of the interfacial tension of a drop.

### 3.5 Constant Force

These experiments are performed with two drops. They should have the same volume and should not be growing or shrinking during the measurements. The rigid capillary is driven forward with a defined velocity and stop as the drops come to contact. This is the point from which a constant force is applied on the drop. In order to control that force the alignment of the capillaries is most important. Especially the vertical alignment should be checked before every measurement. When the intended force is not matching the actually applied force, the value for the interfacial tension should be adjusted. If the applied force too high, gamma needs to be raised and vice versa.



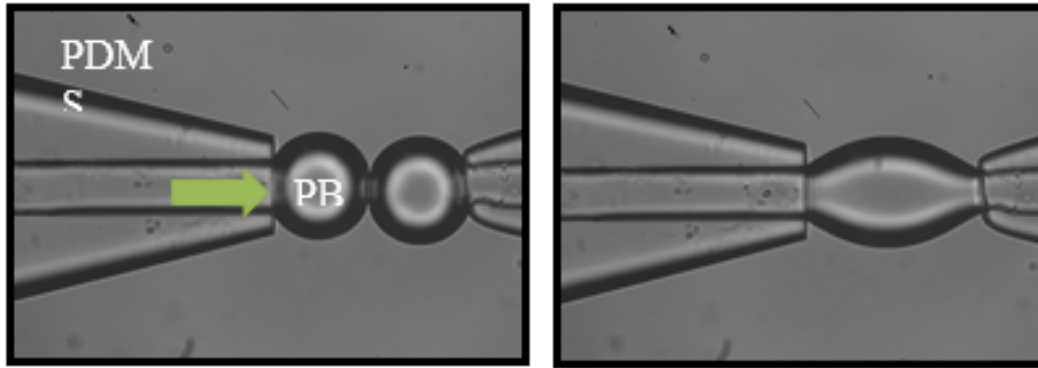


Figure 5: Measurement of the drainage time with two drops.

### 3.6 Buckling

With this modus two drops are moved together with a defined velocity and the force is constantly increasing. This modus can be used to determine the point of instability in which the drops make a sideways or vertical movement because of the force. For this measurement perfect alignment is also very important.

### 3.7 Elastic modulus for capsules

This experiment can be run with either one or two capsules. In the case of a one particle measurement the capsule must be held on the right capillary. There are two parts to it. In the first part, the code finds the contact position between both capsules or between the capsule and the flat surface of the rigid capillary. It estimates first the original gap through geometrical calculations. Note that similarly the blue boundaries that shows up after the cropping must outline the capsule as tightly as possible. Then the rigid capillary is moved forward until the force increases past a threshold that is define as a function of the noise. Then the capillary will move in small steps, the blue crosses being plotted should form a line. The threshold factor may be changed in the "finding contact position" code itself if necessary. After this step the capillary will move back to the desired starting gap. In the second part, the capillary is moved forward by the distance chosen by the user as a percent of the radius of the capsule.

### 3.8 New Experiment Design

In order to design a new experimental procedure a new MATLAB script must be written. This will require a few additional steps:

10. Add the name of the experiment to the list of operating modes in the CCFA.fig GUI using GUIDE TOOL.
11. Update CCFA.m > Mode\_Callback function with new experiment values.
12. Update CreateVals.m function to provide default values for input and output.
13. Update CCFA.m > StartMeasurement\_Callback with new commands.
14. Write new script for running experiment. Be sure to add proper error handling and cleanup of resource allocation for hardware communications.

It is always easiest to start with an existing experiment as a template and modify it according to your needs.