FlowControl configuration

This document details how to configure the flume operating software FlowControl.

General instructions

In the following steps, you will modify variables in the source code. Note that text variables are entered between quotations marks and numerals are entered either between an equal sign and a semi-colon, or as a comma separated list bracketed by curly braces. Some numerals are entered as floating point and some as integers. You must maintain the proper numerical representation when entering numerals.

Initial set-up

There are several steps that must be completed before FlowControl is set-up for running an experiment. The recommended procedure is as follows.

1. Open FlowControl. Scroll down to the first double dotted line, the section labeled *INITIAL SET-UP* (around line 34).
2. Go to the first sub-section, *1. Diameter of the working section*. Enter the diameter, in inches, of the working section of the flume.
3. Go to the next sub-section, *2. PWM stall value*. At low pump outputs, the pumps may stall. The value entered here sets the upper limit of the PWM values output by the program (because the PWM value and flow rate are inversely related, a greater PWM value represents a slower flow). By setting this limit, you prevent the pumps from stalling during low-flow data collection. The PWM stall value can be determined empirically or you can except the existing value by default. Note that this value is an integer.
4. Set-up your flume, upload the program to the Arduino, and calibrate your flow meters (see S3 Appendix: Calibration and operation). After calibration, power down the flume and Arduino.
5. Within the *INITIAL SET-UP* section of the program, go to the sub-section labeled, *3. Flow meter calibrations*, and enter the regression coefficients obtained from your flow meter calibration.
    (a) At the line `char cal_info_1[ ]`, enter text describing the particular meter, the date, etc., for flow meter 1
    (b) At the line `float cal_slope_1`, enter the slope of your calibration, in ml/pulse, for flow meter 1.
    (c) At the line `float cal_yint_1`, enter the y-intercept of your calibration for flow meter 1.
    (d) Repeat this procedure over the next three lines for flow meter 2.

6. Go to the next sub-section, *4. Set-up information for mode 3 (PWM x flow calibration)*. This section allows you set-up the PWM values the program will automatically use in mode 3, the PWM x flow calibration. Between the curly brackets after `int cal_array[]`, enter a series of PWM values, separated by commas, that represent the values you wish to use in your calibration. Note that the PWM values are entered as integers. At `int cal_interval`, enter the time duration at each calibration stage, in seconds. Note that this value is an integer. You may enter your own PWM and time values or simply accept the default values.

This completes the initial set-up of FlowControl.

Pre-experiment set-up

After the initial set-up, you are ready to run an experiment. The following steps must be addressed before each data collection session. If the flume is reconfigured during a session, then this section should be repeated, as the original PWM by flow calibration will no longer be valid.

1. Open FlowControl and scroll down to the second set of double dotted lines, the section labeled *EXPERIMENT SET-UP* (around line 67).
2. In the sub-section labeled *1. Information about this session*, enter information about the investigator, the date, and the project.
3. Set-up the flume, upload FlowControl to your Arduino, and run mode 3 (PWM x flow rate calibration). Perform your PWM by flow rate calibration (see S3 Appendix: Calibration and operation).
4. Power down the flume and your Arduino. Open FlowControl, go to the *EXPERIMENT SET-UP* section, and in the sub-section labeled *2. Coefficients for the PWM x flow calibration*, enter the first, second, and third regression coefficients from your PWM by flow rate calibration.
5. The last step before data collection is to configure your protocol. This is detailed in the next section.

Creating a custom protocol

Following are instructions for creating and storing a custom protocol in FlowControl. Open the program and scroll down to the second set of double dotted lines, the section labeled *EXPERIMENT SET-UP* (around line 67). Find the sub-section labeled *3. Custom protocol*. You will need to modify two lines in this subsection.

The first line you need to modify starts with `float flow_array[]` followed by a series of *floating point values* (separated by commas) enclosed in curly brackets. Each value within the curly brackets specifies a target flow rate in cm s^{-1}. These flow rate values are arranged in the same sequence that they will be administered during the protocol.

The next line in the source code starts with `int time_array[]` followed by a series of *integers*

(separated by commas) enclosed in curly brackets. These values represent the duration of each corresponding stage in seconds.

As an example, if the source code contains the following

```
float flow_array[]{ 5.0, 7.5, 10.0, 20.0}

int time_array[]{ 10, 30, 30, 60}
```

the protocol will consist of a first stage at a flow rate of 5 cm s^{-1} for 10 s, a second stage at a flow rate of 7.5 cm s^{-1} for 30 s, a third stage at 10 cm s^{-1} for 30 s, and a fourth and final stage at 20 cm s^{-1} for 60 s.

It is possible to store a number of protocols in the program. Copy and paste the two lines describing your protocol. Modify the flow and timing of the second set as necessary. Then simply comment out the set(s) that will not be used for the current experiment by typing double backslashes at the start of each line. In the following example, the second protocol will be run when the investigator selects mode 1.

```
//    float flow_array[]{ 15.0, 20.0, 25.0, 30.0}

//    int time_array[]{ 60, 60, 60, 60}

      float flow_array[]{ 2.0, 4.0, 6.0, 8.0, 10.0, 12.0}

      int time_array[]{ 30, 30, 30, 30, 30, 30}

//    float flow_array[]{ 5.0, 10.0, 15.0, 20.0}

//    int time_array[]{ 30, 30, 30, 30}
```

One further note. The pumps may not be able to overcome inertia when a protocol starts at a slow flow rate. To prevent this, simply code in a faster flow rate as the initial, very brief (i.e. 5 s), stage of the protocol. After this initial brief stage, immediately drop the flow rate to the desired value on the next stage.

Running an experiment

FlowControl should now be ready for data collection. Save the program, set-up your flume, upload FlowControl to your Arduino and select mode 1 to call up your custom protocol or mode 2 to manually control the flume.