Supporting Information for:

FlowCal: A user-friendly, open source software tool for automatically converting flow cytometry data from arbitrary to absolute units.

Sebastian M. Castillo-Hair, John T. Sexton, Brian P. Landry, Evan J. Olson, Oleg A. Igoshin, Jeffrey J. Tabor

Contents

Procedure S1. Installing FlowCal

To install FlowCal, follow these steps.

1. FlowCal requires Python 2.7. If you already have Python installed on your computer, you may skip to step 3.

   We recommend that Windows and OS X users download and install Anaconda, a Python distribution that already includes commonly used Python packages, many of which are used by FlowCal. OSX already includes its own version of python, but it does not include some python tools that FlowCal requires. Therefore, Anaconda is recommended. For other methods of installation, refer to the FlowCal documentation http://taborlab.github.io/FlowCal/.

   To install Anaconda, navigate to https://www.continuum.io/downloads. If you are using Windows, click on "Windows 64-bit Graphical Installer" or "Windows 32-bit Graphical Installer" under the "Python 2.7" column, depending on whether your computer is 32-bit or 64-bit. (**Figure S1**). The 64-bit version is strongly recommended. Similarly, if you are using OS X, click on "Mac OS X 64-bit Graphical Installer" under the "Python 2.7" column. This will download the installer.

2. Double click the installer (.exe in Windows, .pkg in OS X) and follow the instructions on screen.

3. Navigate to http://taborlab.github.io/FlowCal/, and click on the "Download FlowCal" link (**Figure S2**). A file called `FlowCal-master.zip` will be downloaded. Unzip this file.

4. Inside the unzipped folder, double click on "Install FlowCal (Windows).bat" or "Install FlowCal (OSX)" if you are using Windows or OS X, respectively. This will open a terminal window and install FlowCal. This may take a few minutes. When installation is finished, the terminal will show the message "Press Enter to finish…". If the installation was successful, your terminal should look like **Figure S3**. Press Enter to close the terminal window.

   Mac OS X only: If the following error message appears after double clicking "Install FlowCal (OSX)": "'Install FlowCal (OSX)' can't be opened because it is from an unidentified developer.", navigate to System Preferences -> Security and Privacy -> General, and click the "Open Anyways" button adjacent to the message stating "'Install FlowCal (OSX)' was blocked from opening because it is not from an identified developer". This will remove the security restriction from the program and allow it to run properly.

Note: After installation, the files "Run FlowCal (Windows).bat" and "Run FlowCal (OSX)", used to launch the Excel UI in Windows and OSX, can be renamed or moved to any folder. For example, they could be moved to the desktop for easier access.

Procedure S2. Running FlowCal via the Excel Interface (UI).

To use the FlowCal's Excel UI, follow these steps:

1. Make and save an Excel input file indicating the FCS files to process. Consult **Figure S4** for the spreadsheet format. An example of a properly formatted Excel file can be found in the `examples` folder inside the `FlowCal-master.zip` file, which can also be downloaded from the Supporting Online Materials.
2. Launch FlowCal's Excel UI by double clicking on "Run FlowCal (Windows).bat" or "Run FlowCal (OSX)".
3. A window will appear requesting an input Excel file (**Figure S5**). Locate the Excel file made in step 1 and click on "Open".
4. FlowCal will start processing the indicated calibration beads and cell samples. A terminal window will appear indicating the progress of the analysis. When the analysis finishes, the message "Press Enter to finish…" will appear. Press Enter to close the terminal window.

During processing of the calibration beads and cell samples, FlowCal creates two folders and an output Excel file in the same location as the input Excel file. In the details below, <ID> refers to the value specified in the ID column of the Excel file.

1. The folder "plot_beads" contains plots of the individual steps of processing of the calibration particle samples:
    a. density_hist_<ID>.png (**Figure S6**): A FSC/SSC 2D density diagram of the calibration particle sample, and a histogram for each relevant fluorescence channel.
    b. clustering_<ID>.png (**Figure S7**): A plot of the sub-populations identified during the clustering step, where the different sub-populations are shown in different colors. Depending on the number of channels used for clustering, this plot is a histogram (when using only one channel), a 2D scatter plot (when using two channels), or a 3D scatter plot with three 2D projections (when using three channels or more). If the populations have been identified incorrectly, changing the number of channels used for clustering, or the density gate fraction can improve the results. These two parameters can be changed in the Beads sheet of the input Excel file.
    c. populations_<channel>_<ID>.png (**Figure S8**): A histogram showing the identified microbead sub-populations in different colors, for each fluorescence channel in which a MEF standard curve is to be calculated. In addition, a vertical line is shown representing the median of each population, which is later used to calculate the standard curve. Sub-populations that were not used to generate the standard curve are shown in gray.
    d. std_crv_<channel>_<ID>.png (**Figure S9**): A plot of the fitted standard curve, for each channel in which MEF values were specified.
2. The folder "plot_samples" contains plots of the experimental cell samples. Each experimental sample of name "ID" as specified in the Excel input sheet results in a file named "<ID>.png". This image contains a FSC/SSC 2D density diagram with the gated region indicated, and a histogram for each user-specified fluorescence channel (**Figure S10**).
3. The file "<Name of the input Excel file>_output.xlsx", contains calculated statistics for each sample, for each relevant fluorescence channel (**Figure S11**, top). To produce this file,

FlowCal copies the Instruments, Beads, and Samples sheets from the input Excel file, unmodified, to the output file, and adds columns to the Samples sheet with statistics. Statistics per sample include: the number of events after gating and the acquisition time. Statistics per fluorescence channel include: channel gain, mean, geometric mean, median, mode, standard deviation, coefficient of variation (CV), interquartile range (IQR), and robust coefficient of variation (RCV). In addition, a “Histograms” tab is generated, with bin/counts pairs for each sample and relevant fluorescence channel, in the specified units (**Figure S11**, bottom).

Note S1. Density Gating

Density gating is an automated gating algorithm that automatically estimates a 2D probability density function from a list of events, identifies the “densest” area (having the highest estimated probability of occurrence of an event), and accepts only the events inside. If applied to the FSC/SSC channels, density gating can automatically isolate single microbeads or cells from aggregates and non-bead or non-biological debris, even when these events are a substantial fraction of the total count (**Figure S12**).

In FlowCal, density gating is implemented in the `FlowCal.gate.density2d()` function. When using this function directly in Python, the user can specify the fraction of events to keep, the two channels to use, and the amount of “smoothing” involved in estimating the probability density function. The Excel UI allows the user to specify the gating fraction only (**Figure S4**). For more details, refer to the documentation on http://taborlab.github.io/FlowCal. In short, FlowCal:

1. Determines the number of events to keep, based on the user specified gating fraction and the total number of events of the input sample.
2. Divides the 2D channel space into a rectangular grid, and counts the number of events falling within each bin of the grid. The number of counts per bin across all bins comprises a 2D histogram, which is a coarse approximation of the underlying probability density function.
3. Smoothes the histogram generated in Step 2 by applying a Gaussian Blur. Theoretically, the proper amount of smoothing results in a better estimate of the probability density function. Practically, smoothing eliminates isolated bins with high counts, most likely corresponding to noise, and smoothes the contour of the gated region.
4. Selects the bins with the greatest number of events in the smoothed histogram, starting with the highest and proceeding downward until the desired number of events to keep, calculated in step 1, is achieved.
5. Returns the gated event list.

Note S2. Processing of Calibration Particles

In a typical experiment, FlowCal processes calibration particle samples to produce a standard curve, which it then uses to convert cell data from channel units (raw fluorescence values given by the detector and stored in the FCS file) to MEF units. An intermediate transformation from channel units to arbitrary units (a.u.) (which typically involves exponentiating channel units, if using a logarithmic amplifier) is not required, and therefore not performed by FlowCal. The functions implementing these procedures are encoded in the `FlowCal.mef` module. Excel UI-based processing of calibration particles follows the following procedure:

1. Density gating (**Note S1**) is applied in the FSC/SSC channels to eliminate microbead aggregates and debris (**Figure S6**).
2. The individual microbead subpopulations are identified using automated clustering (**Figure S7**). FlowCal performs clustering in one or more fluorescent channels, as specified by the user in the “Clustering Channels” column (**Figure S4**). If the fluorophores inside the calibration beads can be detected in multiple channels, clustering in these channels can improve resolution of bead peaks that are near or at saturation in one channel. Clustering is performed using Gaussian Mixture Models, as implemented in the `sklearn` Python package (1). This algorithm attempts to fit a multivariate Gaussian distribution to each subpopulation.
3. For each subpopulation, the median channel unit fluorescence is calculated (**Figure S8**).
4. Microbead subpopulations are discarded if they are found to be close to the saturation limit of the detector. Currently, the algorithm checks whether the populations are closer than a certain number (default: 2.5) of standard deviations from values 1.5% offset from the lower and upper channel value limits (e.g. 15 and 1008 for a 10-bit detector). Only populations that are not saturating are retained.
5. Using the fluorescence values of the retained populations in channel units, and the corresponding MEF values provided by the user, the standard curve is generated (**Figure S9**). We use a microbead fluorescence model with an additional autofluorescence term, as we found that this model fits the experimental data better (**Note S3**). The parameters are fitted using the `minimize` function in `scipy`, which uses the quasi-Newton Broyden–Fletcher–Goldfarb–Shanno (BFGS) method.

Using the Python API, the user can perform other types of gating, use different clustering algorithms, or use a different standard curve model. For more details, refer to the documentation on http://taborlab.github.io/FlowCal.

Note S3. Calibration Particle Fluorescence Model and Autofluorescence

We seek to describe the relationship between the fluorescence of an event measured in gain-dependent raw channel units as reported by the flow cytometer acquisition software in the FCS file and its corresponding fluorescence in MEF units.

Both a.u. and MEF units are directly proportional to the number of photons which strike the instrument detector, n. Thus,

$$n = \alpha \cdot FL_{MEF} = \beta(gain) \cdot FL_{a.u.} \qquad \text{…(S3.1)}$$

Solving for FL_{MEF}:

$$FL_{MEF} = \gamma(gain) \cdot FL_{a.u.} \qquad \text{…(S3.2)}$$

Where $\gamma(gain) = \beta(gain)/\alpha$. For numerical accuracy (explained below), we take the logarithm on both sides of S3.2:

$$\log(FL_{MEF}) = \log(FL_{a.u.}) + \log\big(\gamma(gain)\big) \qquad \text{…(S3.3)}$$

For a linear amplifier, fluorescence in a.u. is obtained from the raw channel units as

$$FL_{a.u.} = k \cdot FL_{ch} \qquad \text{…(S3.4)}$$

Where k is an arbitrary constant. Replacing S3.4 in S3.3:

$$\log(FL_{MEF}) = \log(FL_{ch}) + \log\big(k \cdot \gamma(gain)\big) \qquad \text{…(S3.5)}$$

Similarly, for a logarithmic amplifier, fluorescence in a.u. is obtained from the raw channel units as

$$FL_{a.u.} = a \cdot \exp\left(\frac{FL_{ch}}{r}\right) \qquad \text{…(S3.6)}$$

Where r is related to the instrument range (number of decades that it covers) and the resolution of the Analog-to-Digital Converter (ADC). Replacing S3.6 in S3.3:

$$\log(FL_{MEF}) = \frac{FL_{ch}}{r} + \log\big(a \cdot \gamma(gain)\big) \qquad \text{…(S3.7)}$$

S3.5 and S3.7 can be expressed in the same general form:

$$\log(FL_{MEF}) = m \cdot f_{amp}(FL_{ch}) + b \qquad \text{…(S3.8)}$$

Where $f_{amp}(x) = x$ for a logarithmic amplifier, and $f_{amp}(x) = \log(x)$ for a linear one. Note that all dependence upon instrument gain settings is captured by the intercept term b (as seen in **Figure S14**).

With the model described in equation (S3.8), we would like to use calibration particles with manufacturer-specified known FL_{MEF} values to fit the m and b terms against the measured fluorescence of the calibration particles in raw channel units (FL_{ch}). Once determined, equation (S3.8) and the fit m and b parameters can be used to calibrate cell sample fluorescence values from raw channel units to MEF units. Note that, in most cases, the fluorescence levels of the different calibration particle subpopulations are logarithmically spaced over 3 or 4 decades. Therefore, fitting is performed using $\log(FL_{MEF})$ instead of FL_{MEF} in order to increase numerical accuracy.

After analysis of several calibration particle samples, we have observed that channel fluorescence of the microbead subpopulations versus MEF values specified by the manufacturer deviates from the perfect line described by equation (S3.8) in a concave fashion (**Figure S9**, green line). We hypothesized that this difference arises from bead autofluorescence not accounted for by the manufacturer. To compensate for this, we decompose the total fluorescence of the calibration particles as follows:

$$FL_{MEF} = FL_{MEF}^{fluorophore} + FL_{MEF}^{autofluorescence} \qquad \text{...(S3.9)}$$

where FL_{MEF} is the total fluorescence of the calibration particle, $FL_{MEF}^{fluorophore}$ is the fluorophore-dependent fluorescence specified by the manufacturer, and $FL_{MEF}^{autofluorescence}$ is the autofluorescence of the calibration particle. Thus, we augment the fluorescence model specified in equation (S3.8) with a third parameter for the unknown bead autofluorescence term:

$$\log\left(FL_{MEF}^{fluorophore} + FL_{MEF}^{autofluorescence}\right) = m \cdot f_{amp}(FL_{ch}) + b \qquad \text{...(S3.10)}$$

We can now measure calibration particles on our flow cytometer and use the resulting calibration particle fluorescence in raw channel units (FL_{ch}) in conjunction with the manufacturer-specified calibration particle fluorophore fluorescence values in MEF units ($FL_{MEF}^{fluorophore}$) to fit the three parameters of the fluorescence model specified in equation (S3.10). Once fit, the m and b parameters can be used with equation (S3.8) to calibrate new cell sample fluorescence values from raw channel units to MEF units.

An example of equation (S3.10) (after isolating for $FL_{MEF}^{fluorophore}$) is shown in green in **Figure S9**, and an example of equation (S3.8) is shown in red in **Figure S9**.

Note S4. Format of the input Excel file

FlowCal's Excel interface requires a properly formatted Excel file that depicts the samples to be analyzed and the data processing parameters. The Excel input file must have at least three sheets, named 'Instruments', 'Beads', and 'Samples'. Other sheets may be present, but FlowCal will ignore them.

Instruments sheet (**Figure S4**, top).

This sheet must be filled with basic information about the flow cytometer used to acquire the samples. Each row represents an instrument. Typically, the user would only need to specify one instrument; however FlowCal allows the simultaneous processing of samples taken with different instruments. For each row, the following columns must be filled.

1. "ID" (column A in **Figure S4**) used to reference the instrument from the other sheets. Each row must have a unique ID.
2. "Forward Scatter Channel" (C) and Side Scatter Channel (D): the names of these channels exactly as they appear in the acquisition software.
3. "Fluorescence channels" (E): The names of the relevant fluorescence channels as a comma-separated list, exactly as they appear in the acquisition software.
4. "Time Channel" (F): The name of the channel registering the time of each event. The FCS standard dictates that this should be called "Time", but some non-standard files may use a different name. This can be found in the acquisition software.

Additional columns, like "Description" (B in **Figure S4**), can be added in any place for the user's records, and will be copied unmodified to the output Excel file by FlowCal.

Beads sheet. (**Figure S4**, middle)

This sheet contains details about calibration microbeads and how to process them. Each row represents a different sample of beads. For each row, the following columns must be filled:

1. "ID" (A): used to reference the beads sample from the Samples sheet, and to name the figures produced by FlowCal. Each row must have a unique ID.
2. "Instrument ID" (B): The ID of the instrument used to take the sample.
3. "File Path" (C): the name of the corresponding FCS file.
4. "<Channel name> MEF Values" (E shown, F and G hidden): MEF values provided by the manufacturer, for each channel in which a standard curve must be calculated. If MEF values are provided for a channel, the corresponding instrument should include this channel name in the "Fluorescence Channels" field. More "<Channel name> MEF Values" columns can be added if needed, or removed if not used.
5. "Gate Fraction" (H): a gate fraction parameter used for density gating.
6. "Clustering Channels" (I): the fluorescence channels used for clustering, as a comma separated list.

Additional columns, like "Lot" (D in **Figure S4**), can be added in any place for the user's records, and will be copied unmodified to the output Excel file by FlowCal.

Samples sheet (**Figure S4**, bottom)

In this sheet, the user specifies cell samples and tells FlowCal how to process them. Each row contains the information used in the analysis of one FCS file. Several analyses can be performed on the same file (e.g. rows 10 and 12 in **Figure S4**, bottom, in which the gating fraction is varied). For each row, the following columns must be filled:

1. “ID” (A): used to name the figures produced by FlowCal. Each row must have a unique ID.
2. “Instrument ID” (B): The ID of the instrument used to take the sample.
3. “Beads ID” (C): The ID of the beads sample that will be used to perform the MEF transformation. Can be left blank if MEF units are not desired.
4. “File Path” (D): the name of the corresponding FCS file.
5. “<Channel name> Units” (E, F, G, and H): The units in which to report statistics and make plots, for each fluorescence channel. If left blank, no statistics or plots will be made for that channel. More of these columns can be added or removed if necessary. If this field is specified for a channel, the corresponding instrument should include this channel in its “Fluorescence Channels” field. The available options are:
    a. Channel: Raw “Channel Number” units, exactly as they are stored in the FCS file.
    b. a.u.: Arbitrary Units.
    c. MEF: MEF units.
6. “Gate Fraction” (I): Fraction of samples to keep in the density gating step.

Additional columns, such as “Strain name” and “IPTG (µM)” (J and K in **Figure S4**), can be added in any place for the user’s records, and will be copied unmodified to the output Excel file by FlowCal.

Note S5: Instrument Cross-Calibration and Cell Auto-Fluorescence

Here we discuss how cell auto-fluorescence can influence instrument cross-calibration. In particular, we show that measurement of cell auto-fluorescence is not necessary when performing instrument cross-calibration. In what follows, all fluorescence signals are assumed to be in MEF units.

Fluorescence of a cell sample, as measured by a flow cytometer "A", is given by

$$FL_A^{TOTAL} = FL_A^{AUTO} + FL_A^{REPORTER} \qquad \ldots\text{(S5.1)}$$

Where FL_A^{TOTAL} is the total measured fluorescent signal, and FL_A^{AUTO} and $FL_A^{REPORTER}$ are the components of FL_A^{TOTAL} produced by cell auto-fluorescence and the fluorescent reporter, respectively. We assume that $FL_A^{REPORTER}$ is proportional to the number of reporter molecules in the cell sample $n^{REPORTER}$.

$$FL_A^{REPORTER} = \alpha_A \cdot n^{REPORTER} \qquad \ldots\text{(S5.2)}$$

Where α_A is the fluorescent signal produced by a single molecule of reporter, and it is instrument-specific. Substituting S5.2 in S5.1 and solving for $n^{REPORTER}$:

$$n^{REPORTER} = \frac{FL_A^{TOTAL} - FL_A^{AUTO}}{\alpha_A} \qquad \ldots\text{(S5.3)}$$

Similarly, for an instrument "B":

$$FL_B^{REPORTER} = \alpha_B \cdot n^{REPORTER} \qquad \ldots\text{(S5.4)}$$

and

$$n^{REPORTER} = \frac{FL_B^{TOTAL} - FL_B^{AUTO}}{\alpha_B} \qquad \ldots\text{(S5.5)}$$

If we were to measure the same cell sample in both instruments "A" and "B", the relationship between $FL_A^{REPORTER}$ and $FL_B^{REPORTER}$ can be obtained from S5.2 and S5.4. For example, solving for $FL_A^{REPORTER}$:

$$FL_A^{REPORTER} = m \cdot FL_B^{REPORTER} \qquad \ldots\text{(S5.6)}$$

Where $m = \alpha_A / \alpha_B$. Cross-instrument calibration would be performed in the following way: 1) measure the fluorescence of a set of calibrating samples in both instruments, including a negative control (no reporter) and a few strains with different expression levels of reporter, $n_i^{REPORTER}$. 2) Perform auto-fluorescence subtraction using FL_A^{AUTO} and FL_B^{AUTO} measured from the negative control, to obtain $FL_{A,i}^{REPORTER}$ and $FL_{B,i}^{REPORTER}$. 3) Perform linear regression with zero-intercept to estimate m in S5.6. 4) Any new sample measured in instrument "B" can be converted to Instrument "A" units using S5.6 and the estimated m, after background subtraction.

If we don't know the auto-fluorescence of a sample, however, we can still relate the total measured fluorescence by equating S5.3 and S5.5:

$$\frac{FL_A^{TOTAL} - FL_A^{AUTO}}{\alpha_A} = \frac{FL_B^{TOTAL} - FL_B^{AUTO}}{\alpha_B}$$

$$FL_A^{TOTAL} = \frac{\alpha_A}{\alpha_B} FL_B^{TOTAL} + \left(FL_A^{AUTO} - \frac{\alpha_A}{\alpha_B} FL_B^{AUTO} \right)$$

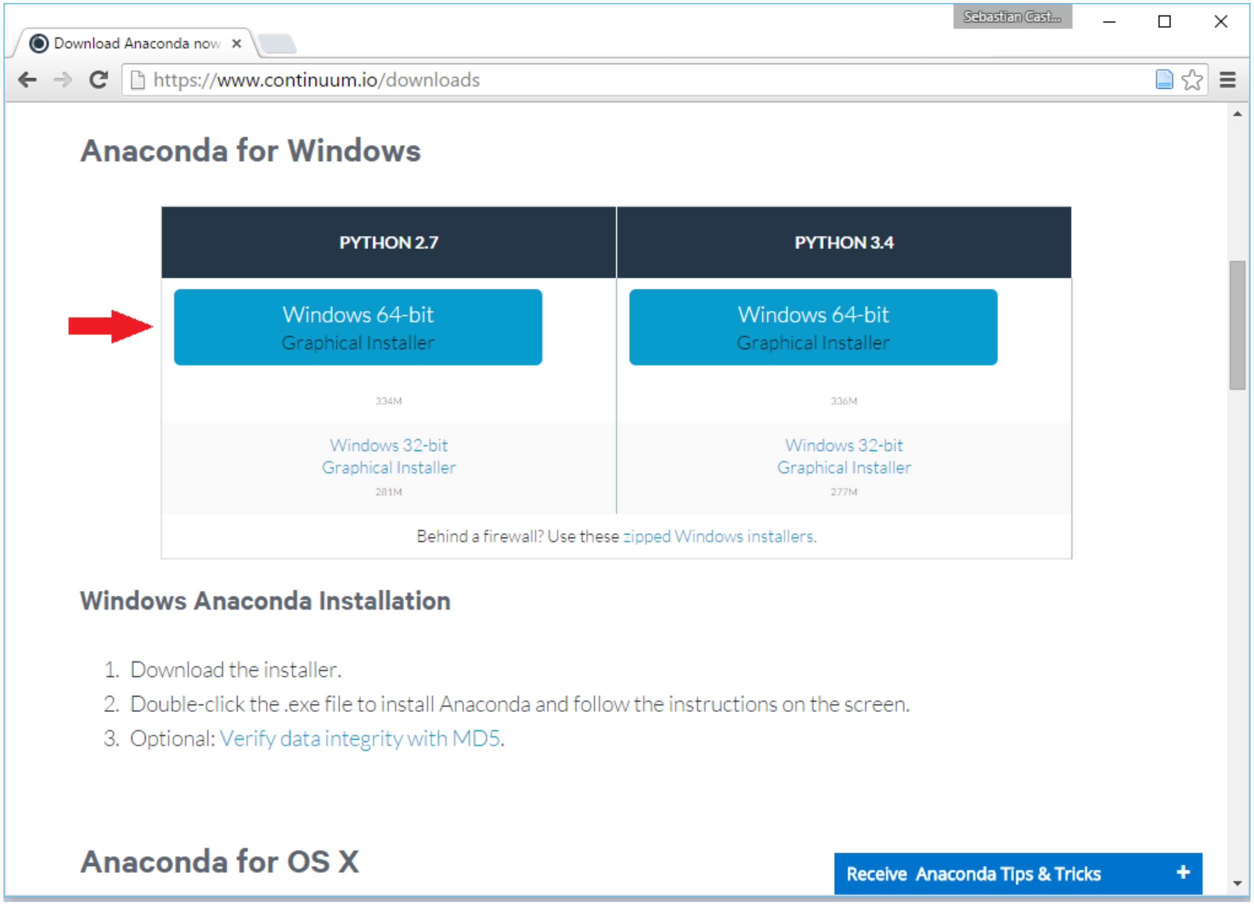
Or

$$FL_A^{TOTAL} = m \cdot FL_B^{TOTAL} + b \qquad \text{…(S5.7)}$$

Where the effects of cell auto-fluorescence are entirely captured by the constant b. Note that m here is the same as in S5.6.

An alternative cross-calibration procedure that does not require measuring auto-fluorescence is the following: 1) measure the fluorescence of a set of calibrating samples in both instruments. This set includes a few strains with different expression levels of reporter, $n_i^{REPORTER}$. 2) Perform standard linear regression to estimate m and b in S5.7. 3) Any new sample measured in instrument “B” can be converted to Instrument “A” units using S5.6 and the estimated m, after background subtraction. This is the approach taken in **Figure 2F**.

Figure S1. Website of the Anaconda distribution.



Sebastian Cast...
Download Anaconda now
https://www.continuum.io/downloads
Receive Anaconda Tips & Tricks +

Anaconda for Windows

PYTHON 2.7	PYTHON 3.4
Windows 64-bit Graphical Installer 334M	Windows 64-bit Graphical Installer 336M
Windows 32-bit Graphical Installer 281M	Windows 32-bit Graphical Installer 277M

Behind a firewall? Use these zipped Windows installers.

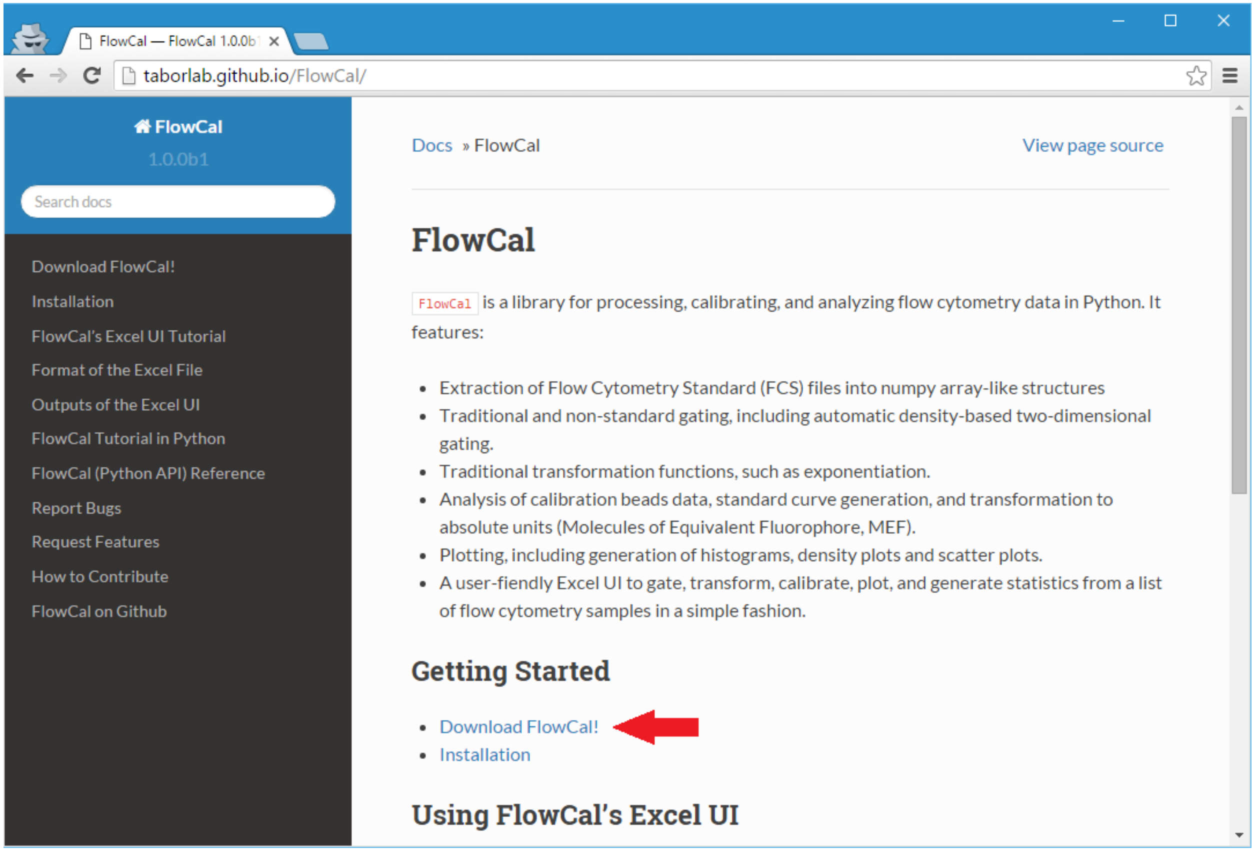
Windows Anaconda Installation

1. Download the installer.
2. Double-click the .exe file to install Anaconda and follow the instructions on the screen.
3. Optional: Verify data integrity with MD5.

Anaconda for OS X

The red arrow indicates the link to download the Python 2.7 installer for Windows 64 bit. The link for Windows 32 bit is located right below. The corresponding link for OS X is located further below.

Figure S2. The FlowCal Website.



FlowCal

FlowCal is a library for processing, calibrating, and analyzing flow cytometry data in Python. It features:

- Extraction of Flow Cytometry Standard (FCS) files into numpy array-like structures
- Traditional and non-standard gating, including automatic density-based two-dimensional gating.
- Traditional transformation functions, such as exponentiation.
- Analysis of calibration beads data, standard curve generation, and transformation to absolute units (Molecules of Equivalent Fluorophore, MEF).
- Plotting, including generation of histograms, density plots and scatter plots.
- A user-fiendly Excel UI to gate, transform, calibrate, plot, and generate statistics from a list of flow cytometry samples in a simple fashion.

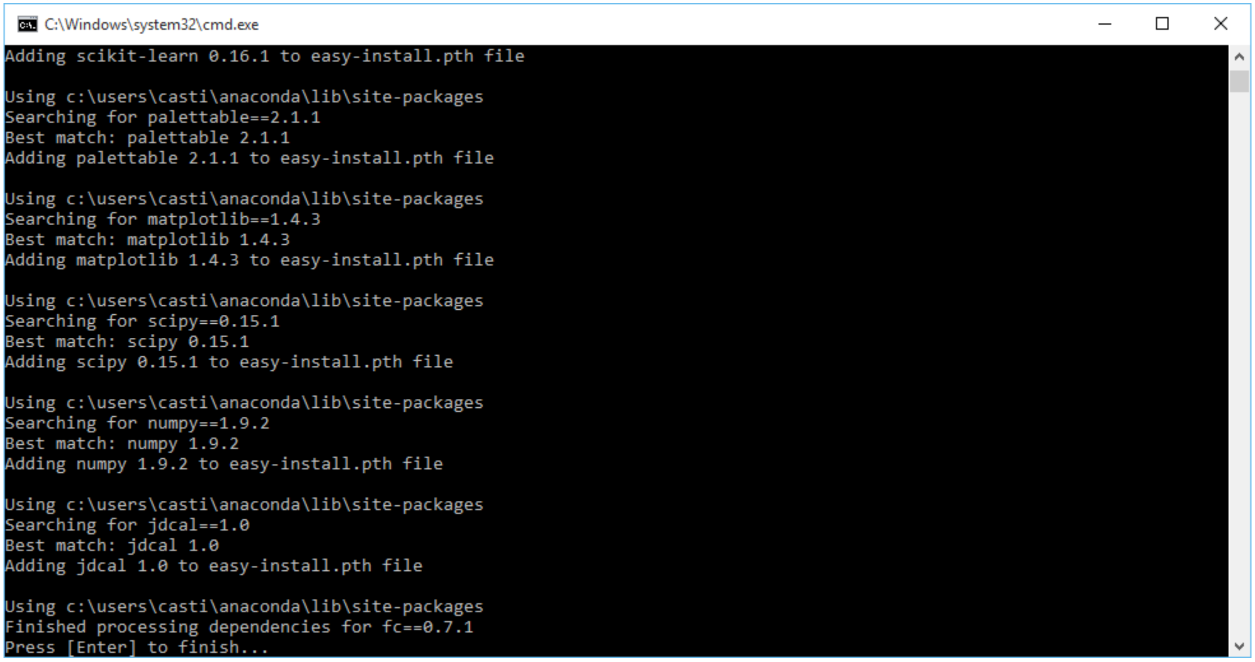
Getting Started

- Download FlowCal!
- Installation

Using FlowCal's Excel UI

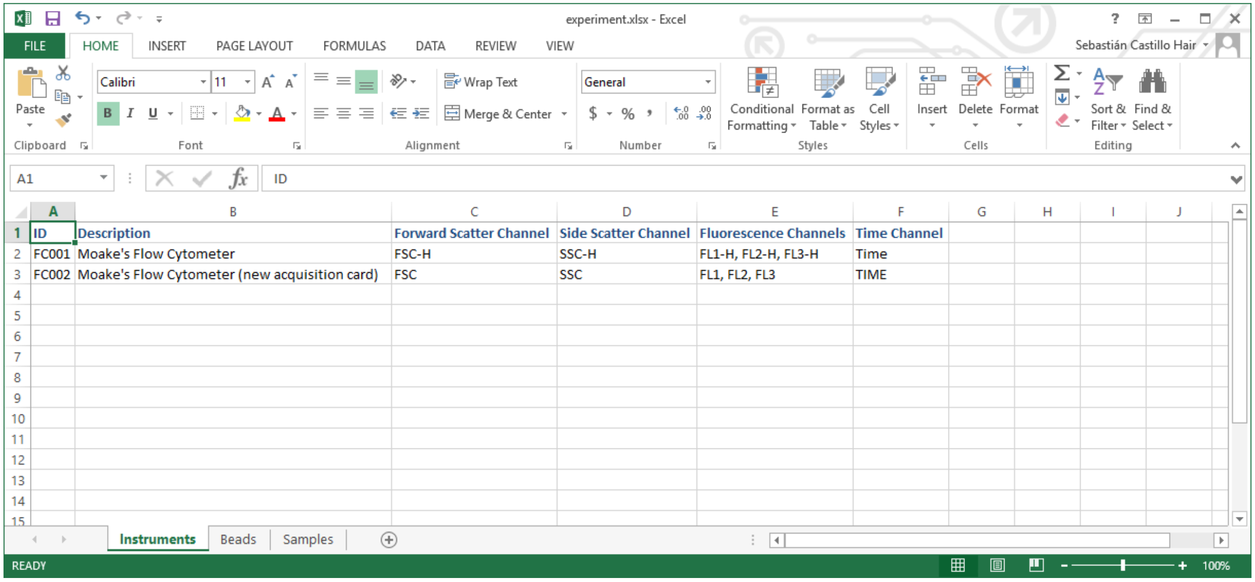
The link to download FlowCal as a zip file is indicated with a red arrow.

Figure S3. Terminal window after installation of FlowCal.

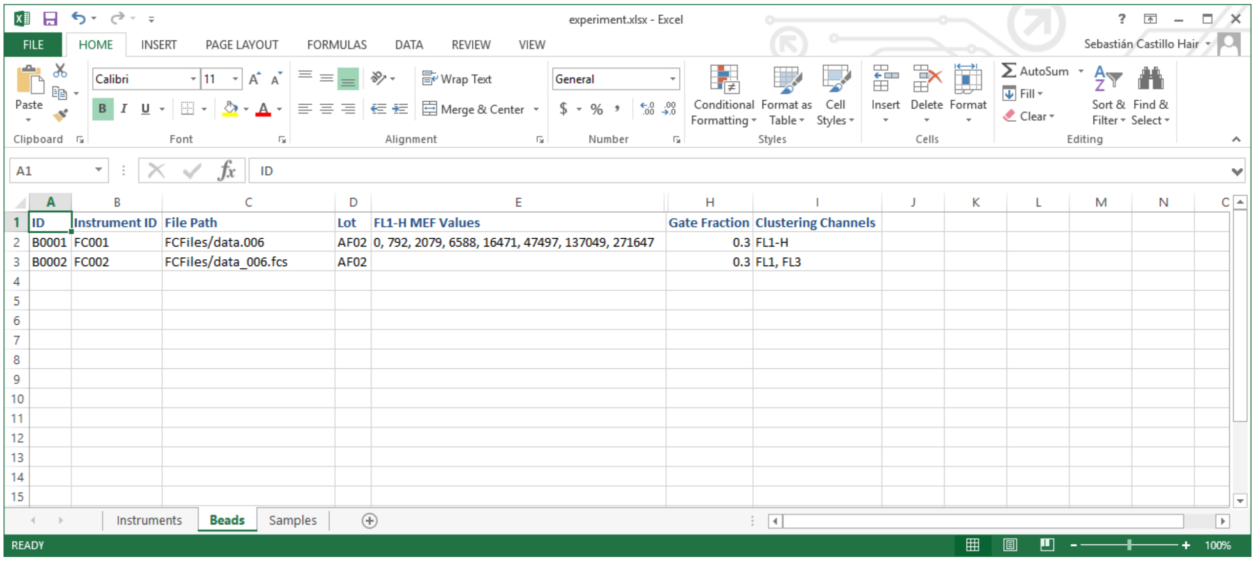


```
Adding scikit-learn 0.16.1 to easy-install.pth file

Using c:\users\casti\anaconda\lib\site-packages
Searching for palettable==2.1.1
Best match: palettable 2.1.1
Adding palettable 2.1.1 to easy-install.pth file

Using c:\users\casti\anaconda\lib\site-packages
Searching for matplotlib==1.4.3
Best match: matplotlib 1.4.3
Adding matplotlib 1.4.3 to easy-install.pth file

Using c:\users\casti\anaconda\lib\site-packages
Searching for scipy==0.15.1
Best match: scipy 0.15.1
Adding scipy 0.15.1 to easy-install.pth file

Using c:\users\casti\anaconda\lib\site-packages
Searching for numpy==1.9.2
Best match: numpy 1.9.2
Adding numpy 1.9.2 to easy-install.pth file

Using c:\users\casti\anaconda\lib\site-packages
Searching for jdcal==1.0
Best match: jdcal 1.0
Adding jdcal 1.0 to easy-install.pth file

Using c:\users\casti\anaconda\lib\site-packages
Finished processing dependencies for fc==0.7.1
Press [Enter] to finish...
```

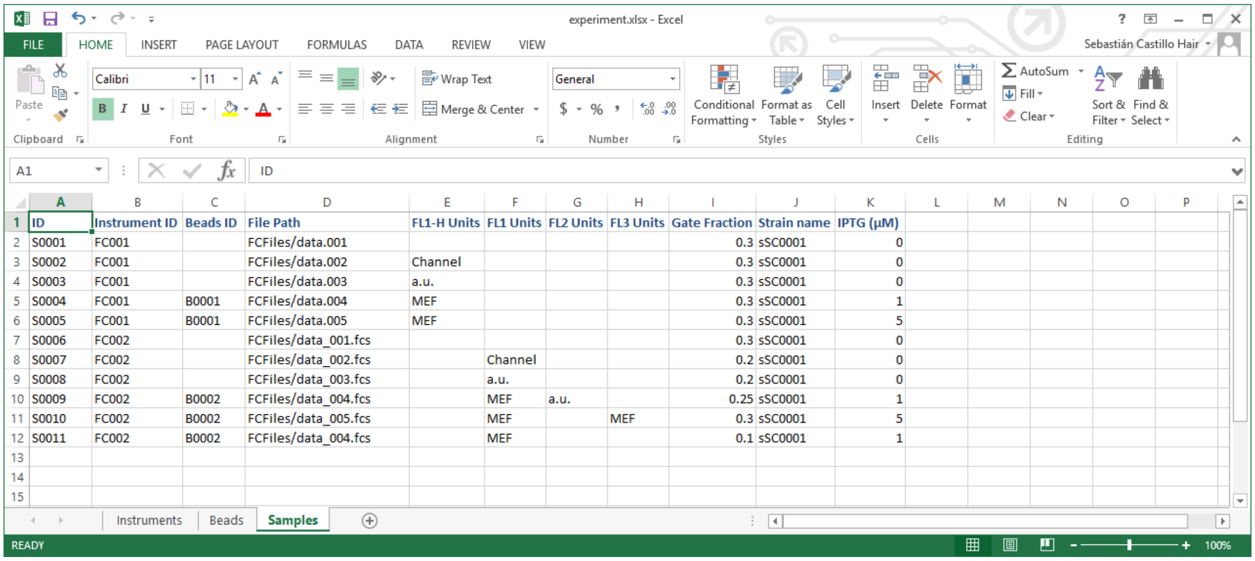
Figure S4. Format of the Input Excel Spreadsheet



ID	Description	Forward Scatter Channel	Side Scatter Channel	Fluorescence Channels	Time Channel
FC001	Moake's Flow Cytometer	FSC-H	SSC-H	FL1-H, FL2-H, FL3-H	Time
FC002	Moake's Flow Cytometer (new acquisition card)	FSC	SSC	FL1, FL2, FL3	TIME



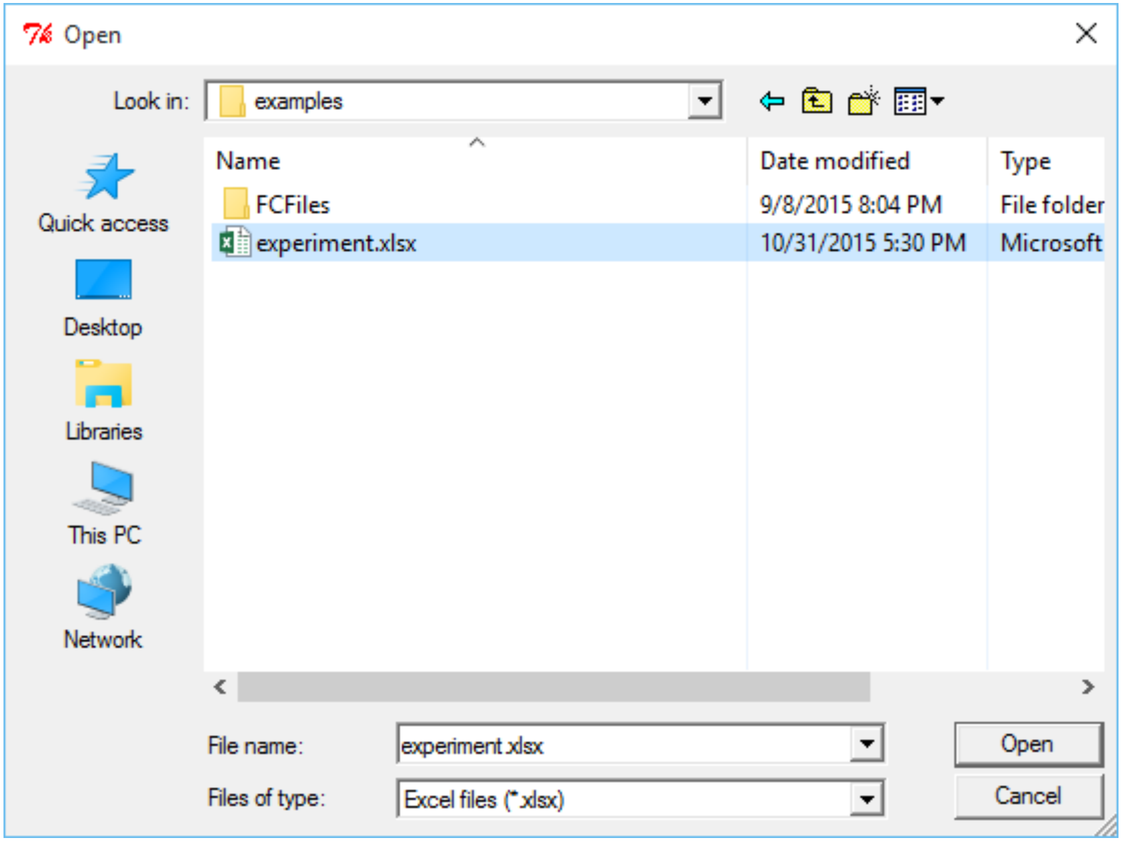
ID	Instrument ID	File Path	Lot	FL1-H MEF Values	Gate Fraction	Clustering Channels
B0001	FC001	FCFiles/data.006	AF02	0, 792, 2079, 6588, 16471, 47497, 137049, 271647	0.3	FL1-H
B0002	FC002	FCFiles/data_006.fcs	AF02		0.3	FL1, FL3



	A	B	C	D	E	F	G	H	I	J	K
1	ID	Instrument ID	Beads ID	File Path	FL1-H Units	FL1 Units	FL2 Units	FL3 Units	Gate Fraction	Strain name	IPTG (µM)
2	S0001	FC001		FCFiles/data.001					0.3	sSC0001	0
3	S0002	FC001		FCFiles/data.002	Channel				0.3	sSC0001	0
4	S0003	FC001		FCFiles/data.003	a.u.				0.3	sSC0001	0
5	S0004	FC001	B0001	FCFiles/data.004	MEF				0.3	sSC0001	1
6	S0005	FC001	B0001	FCFiles/data.005	MEF				0.3	sSC0001	5
7	S0006	FC002		FCFiles/data_001.fcs					0.3	sSC0001	0
8	S0007	FC002		FCFiles/data_002.fcs		Channel			0.2	sSC0001	0
9	S0008	FC002		FCFiles/data_003.fcs		a.u.			0.2	sSC0001	0
10	S0009	FC002	B0002	FCFiles/data_004.fcs		MEF	a.u.		0.25	sSC0001	1
11	S0010	FC002	B0002	FCFiles/data_005.fcs		MEF		MEF	0.3	sSC0001	5
12	S0011	FC002	B0002	FCFiles/data_004.fcs		MEF			0.1	sSC0001	1

Format of the Input Excel Spreadsheet. Top: Instruments Sheet. Middle: Beads Sheet. Bottom: Samples Sheet. For more information, refer to **Note S4**.

Figure S5. Window to open an input Excel file.



Look in: examples

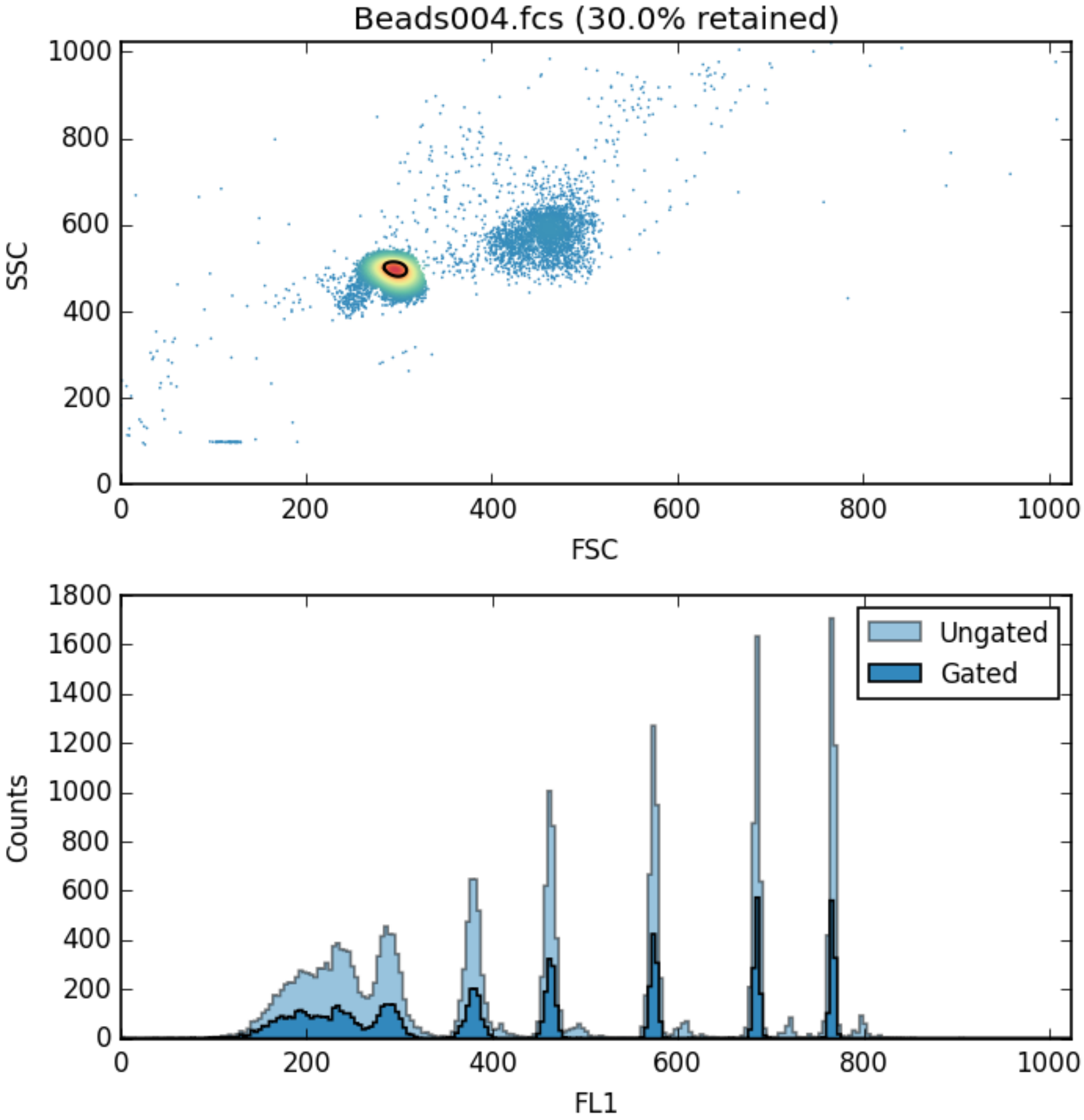
Name	Date modified	Type
FCFiles	9/8/2015 8:04 PM	File folder
experiment.xlsx	10/31/2015 5:30 PM	Microsoft

File name: experiment.xlsx

Files of type: Excel files (*.xlsx)

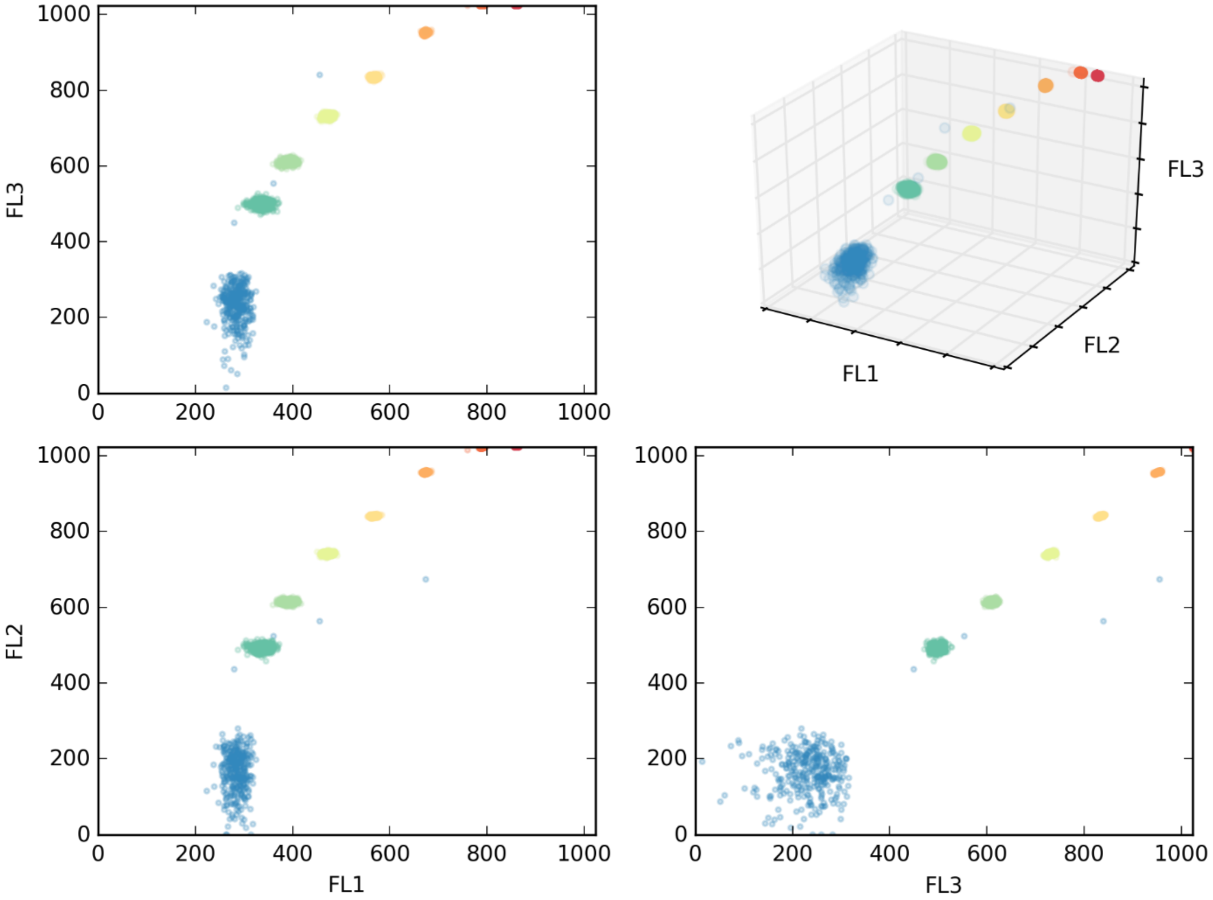
A file named experiment.xlsx is being selected.

Figure S6. Density Gating on a Calibration Particle Sample



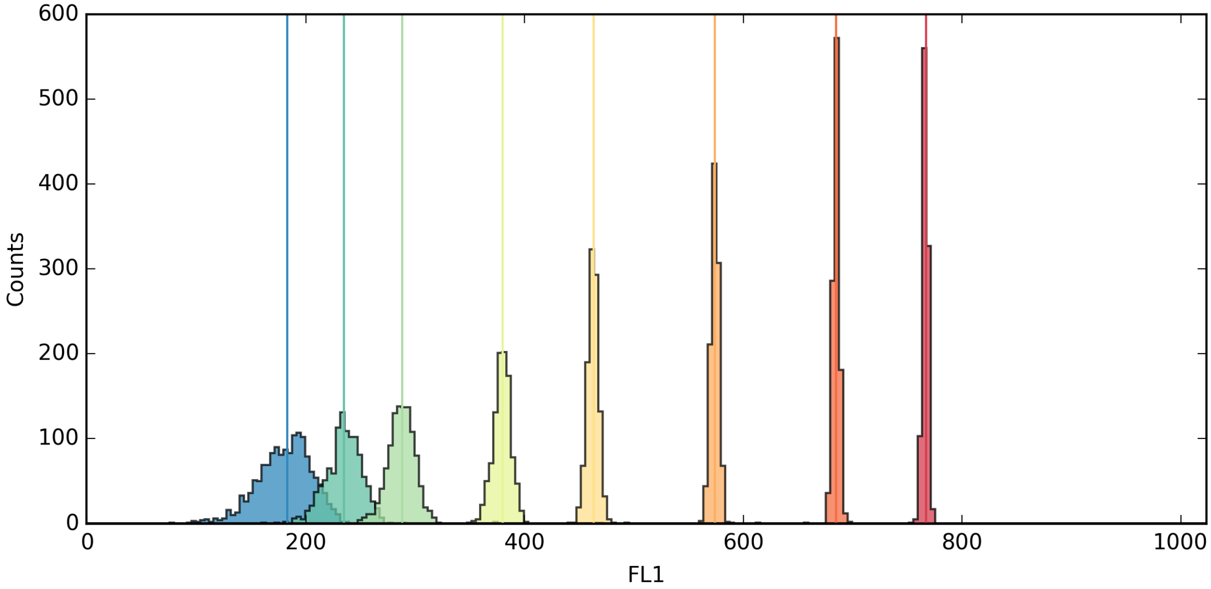
A typical plot showing density gating of a calibration particles sample, as generated by FlowCal as part of the Excel UI workflow. (Top) FSC/SSC scatter diagram. The black contour was obtained by applying density gating to retain 30% of the samples in the densest region. Note the secondary "cluster" of events at the right of the main group. These are likely to be rare bead aggregations, and are therefore excluded by density gating. (Bottom) Histogram of fluorescence channel FL1 in raw channel units. The light and dark histograms represent the data before and after applying density gating, respectively. Note the small secondary peaks to the right of the large main peaks, corresponding to bead agglomerations, in the ungated data.

Figure S7. Multivariate Clustering of Subpopulations of Calibration Particles



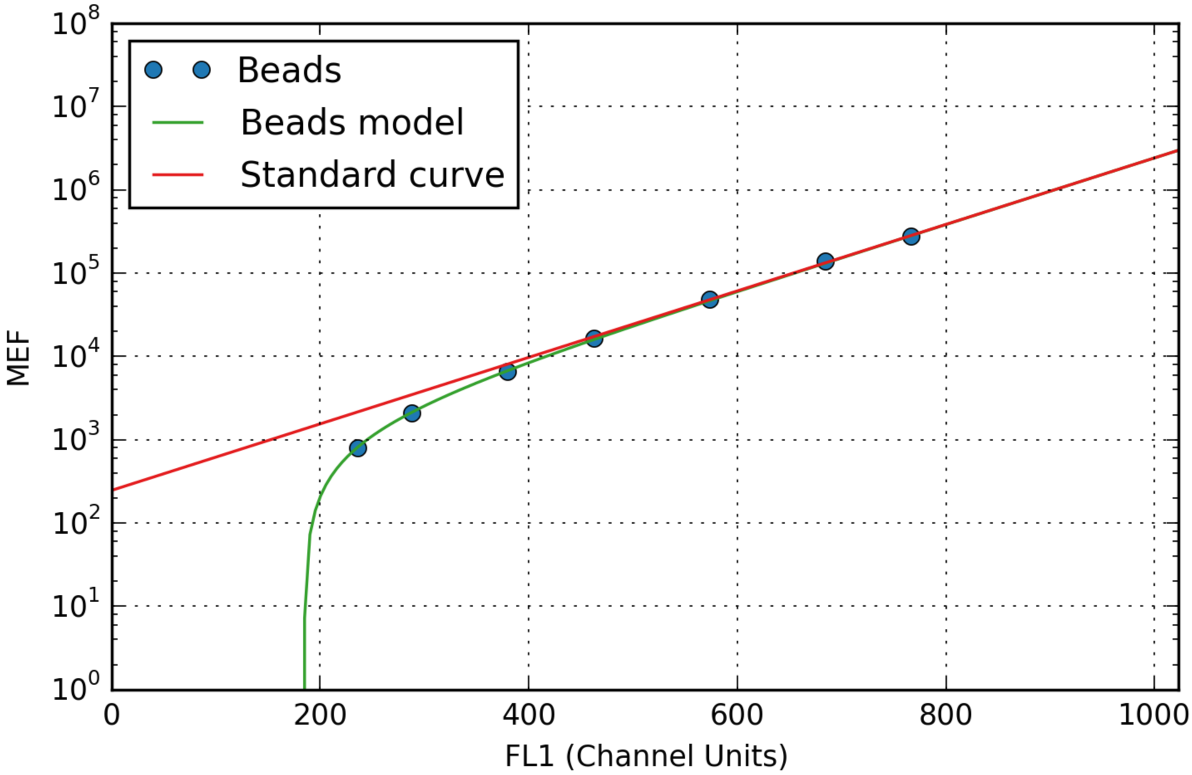
A typical plot showing the result of clustering of different populations of calibration particles when using three fluorescent channels, as generated by FlowCal as part of the Excel UI workflow. Top right subplot shows a 3D scatter plot with the three fluorescent channels, and the other three subplots show different 2D projections. Clustering was performed using Gaussian Mixture Models. Different subpopulations are shown with different colors.

Figure S8. Clustering of Subpopulations of Calibration Particles



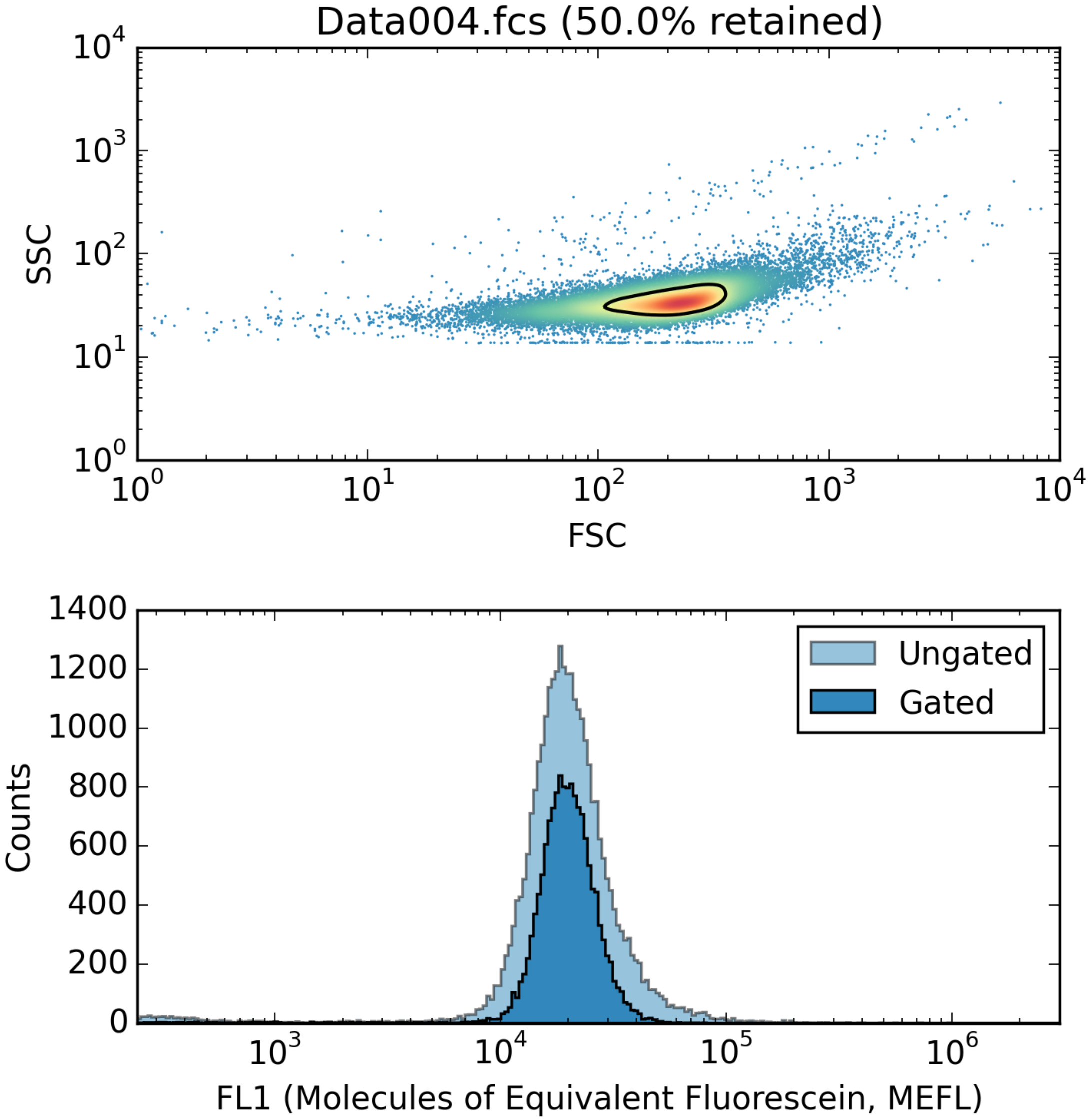
A typical plot showing the different subpopulations of calibration particles and the median fluorescence of each population, as generated by FlowCal as part of the Excel UI workflow. Different subpopulations are shown with different colors. Vertical lines show the median fluorescence value of each subpopulation.

Figure S9. Standard Curve



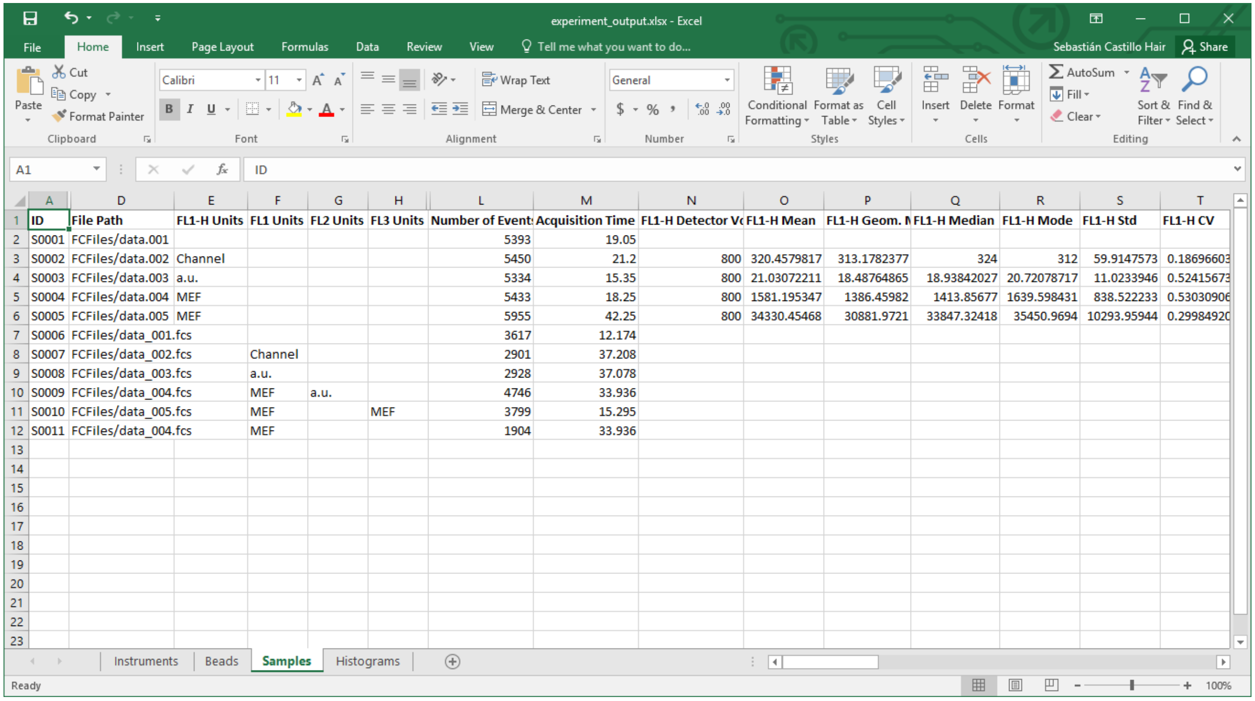
A typical plot showing the calculated standard curve, as generated by FlowCal as part of the Excel UI workflow. Each blue dot represents the fluorescence in channel units (x axis) and in MEF units (y axis) of each subpopulation of the calibration particles. The green line shows the fitted beads fluorescence model with bead autofluorescence. The red line shows the resulting standard curve, used later to convert cell samples to MEF.

Figure S10. Generated Cell Sample Plot

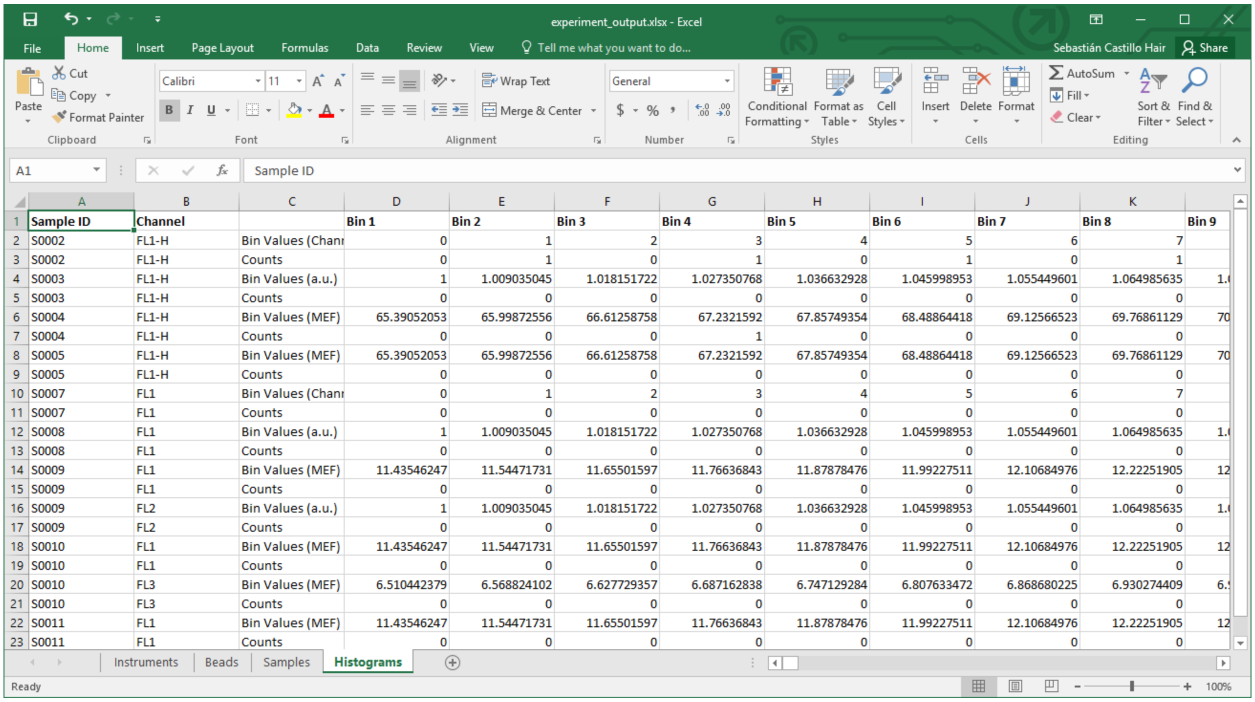


A typical plot showing density gating of an experimental cell sample, as generated by FlowCal as part of the Excel UI workflow. (Top) FSC/SSC scatter diagram. The black contour was obtained by applying density gating to retain 50% of the samples in the densest region. (Bottom) Histogram of fluorescence channel FL1 in MEF units. The clear and dark histograms represent the data before and after applying the density gate, respectively.

Figure S11. The Output Excel Spreadsheet



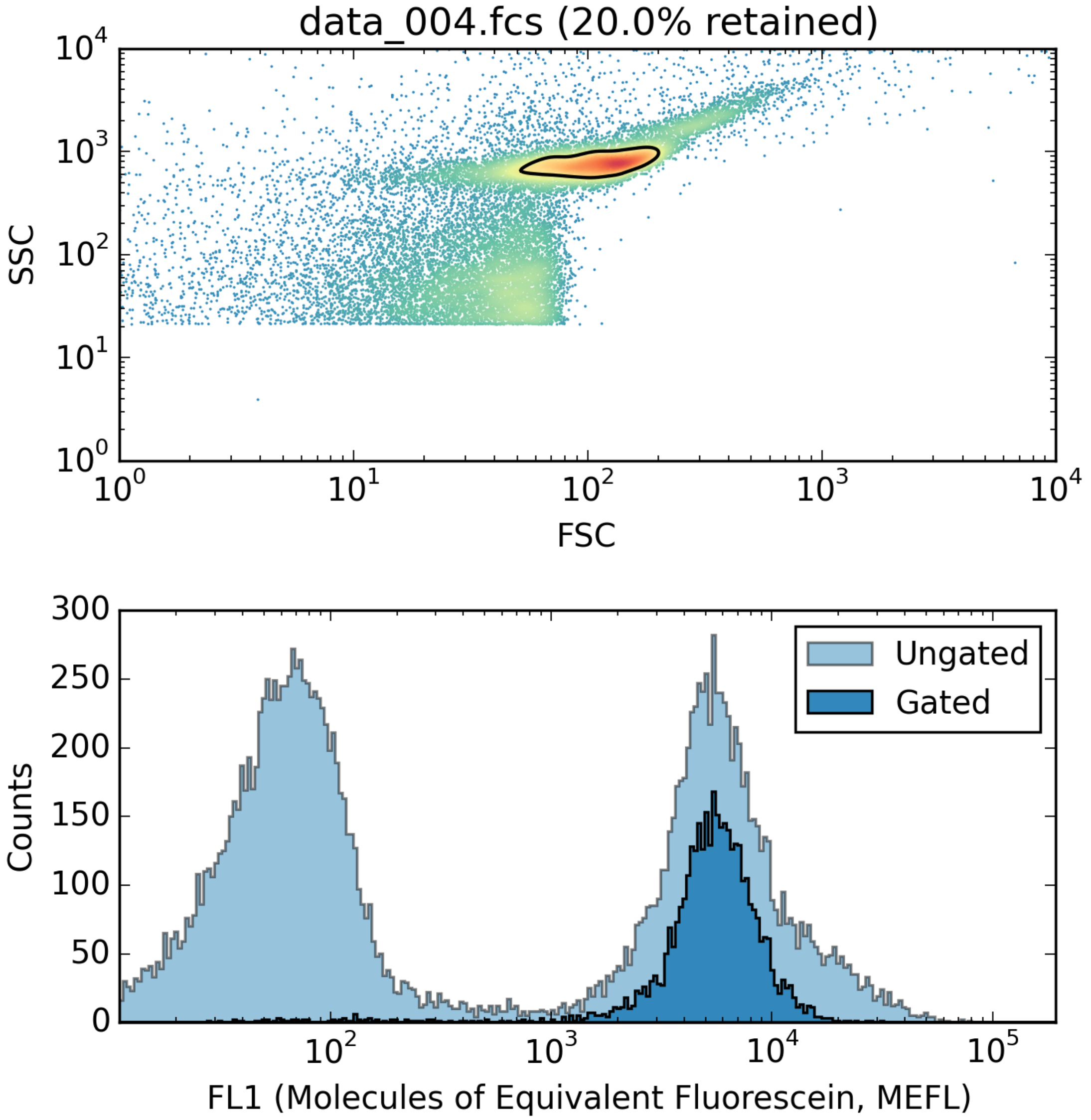
	A	D	E	F	G	H	L	M	N	O	P	Q	R	S	T
1	ID	File Path	FL1-H Units	FL1 Units	FL2 Units	FL3 Units	Number of Event	Acquisition Time	FL1-H Detector Vo	FL1-H Mean	FL1-H Geom. M	FL1-H Median	FL1-H Mode	FL1-H Std	FL1-H CV
2	S0001	FCFiles/data.001					5393	19.05							
3	S0002	FCFiles/data.002	Channel				5450	21.2	800	320.4579817	313.1782377	324	312	59.9147573	0.18696603
4	S0003	FCFiles/data.003	a.u.				5334	15.35	800	21.03072211	18.48764865	18.93842027	20.72078717	11.0233946	0.52415673
5	S0004	FCFiles/data.004	MEF				5433	18.25	800	1581.195347	1386.45982	1413.85677	1639.598431	838.522233	0.53030906
6	S0005	FCFiles/data.005	MEF				5955	42.25	800	34330.45468	30881.9721	33847.32418	35450.9694	10293.95944	0.29984920
7	S0006	FCFiles/data_001.fcs					3617	12.174							
8	S0007	FCFiles/data_002.fcs		Channel			2901	37.208							
9	S0008	FCFiles/data_003.fcs		a.u.			2928	37.078							
10	S0009	FCFiles/data_004.fcs		MEF	a.u.		4746	33.936							
11	S0010	FCFiles/data_005.fcs		MEF		MEF	3799	15.295							
12	S0011	FCFiles/data_004.fcs		MEF			1904	33.936							



	A	B	C	D	E	F	G	H	I	J	K	
1	Sample ID	Channel		Bin 1	Bin 2	Bin 3	Bin 4	Bin 5	Bin 6	Bin 7	Bin 8	Bin 9
2	S0002	FL1-H	Bin Values (Chan	0	1	2	3	4	5	6	7	
3	S0002	FL1-H	Counts	0	1	0	1	0	1	0	1	
4	S0003	FL1-H	Bin Values (a.u.)	1	1.009035045	1.018151722	1.027350768	1.036632928	1.045998953	1.055449601	1.064985635	1.
5	S0003	FL1-H	Counts	0	0	0	0	0	0	0	0	
6	S0004	FL1-H	Bin Values (MEF)	65.39052053	65.99872556	66.61258758	67.2321592	67.85749354	68.48864418	69.12566523	69.76861129	70
7	S0004	FL1-H	Counts	0	0	0	1	0	0	0	0	
8	S0005	FL1-H	Bin Values (MEF)	65.39052053	65.99872556	66.61258758	67.2321592	67.85749354	68.48864418	69.12566523	69.76861129	70
9	S0005	FL1-H	Counts	0	0	0	0	0	0	0	0	
10	S0007	FL1	Bin Values (Chan	0	1	2	3	4	5	6	7	
11	S0007	FL1	Counts	0	0	0	0	0	0	0	0	
12	S0008	FL1	Bin Values (a.u.)	1	1.009035045	1.018151722	1.027350768	1.036632928	1.045998953	1.055449601	1.064985635	1.
13	S0008	FL1	Counts	0	0	0	0	0	0	0	0	
14	S0009	FL1	Bin Values (MEF)	11.43546247	11.54471731	11.65501597	11.76636843	11.87878476	11.99227511	12.10684976	12.22251905	12
15	S0009	FL1	Counts	0	0	0	0	0	0	0	0	
16	S0009	FL2	Bin Values (a.u.)	1	1.009035045	1.018151722	1.027350768	1.036632928	1.045998953	1.055449601	1.064985635	1.
17	S0009	FL2	Counts	0	0	0	0	0	0	0	0	
18	S0010	FL1	Bin Values (MEF)	11.43546247	11.54471731	11.65501597	11.76636843	11.87878476	11.99227511	12.10684976	12.22251905	12
19	S0010	FL1	Counts	0	0	0	0	0	0	0	0	
20	S0010	FL3	Bin Values (MEF)	6.510442379	6.568824102	6.627729357	6.687162838	6.747129284	6.807633472	6.868680225	6.930274409	6.
21	S0010	FL3	Counts	0	0	0	0	0	0	0	0	
22	S0011	FL1	Bin Values (MEF)	11.43546247	11.54471731	11.65501597	11.76636843	11.87878476	11.99227511	12.10684976	12.22251905	12
23	S0011	FL1	Counts	0	0	0	0	0	0	0	0	

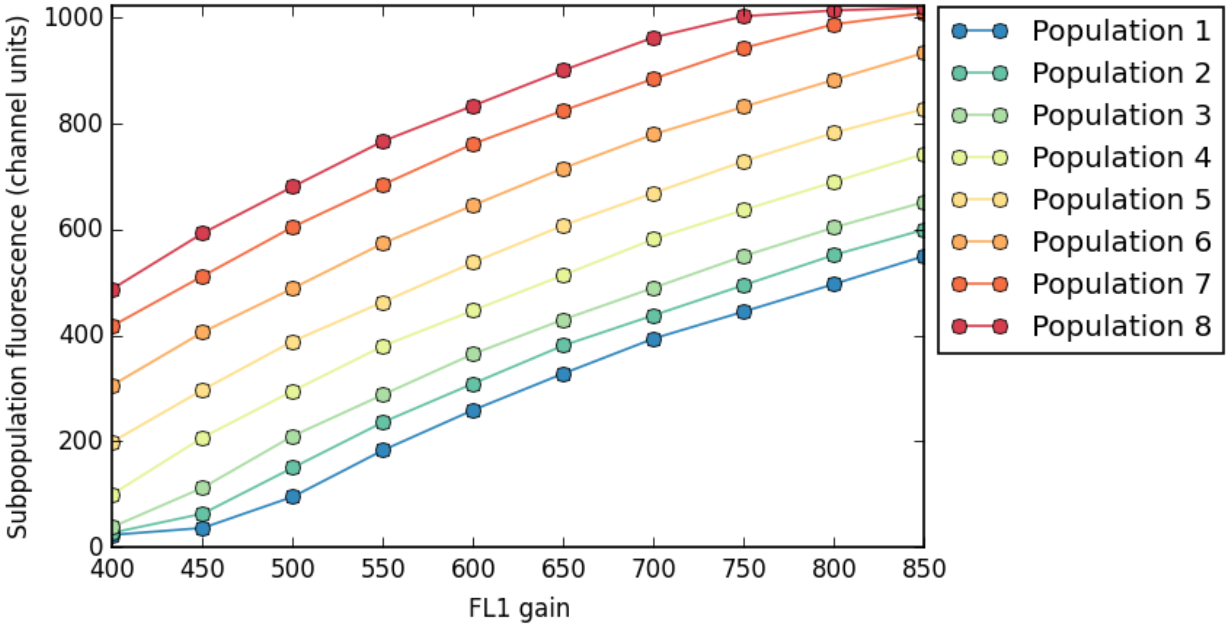
Top: An example of the Samples sheet of an output file, as produced by FlowCal's Excel UI. Columns in the Samples sheet of this file are the same as in the input file (**Figure S4**, some were hidden for this figure), plus several columns for statistics. Bottom: Histograms sheet, generated from the same input file. For more information, refer to Procedure S2.

Figure S12. Density Gating



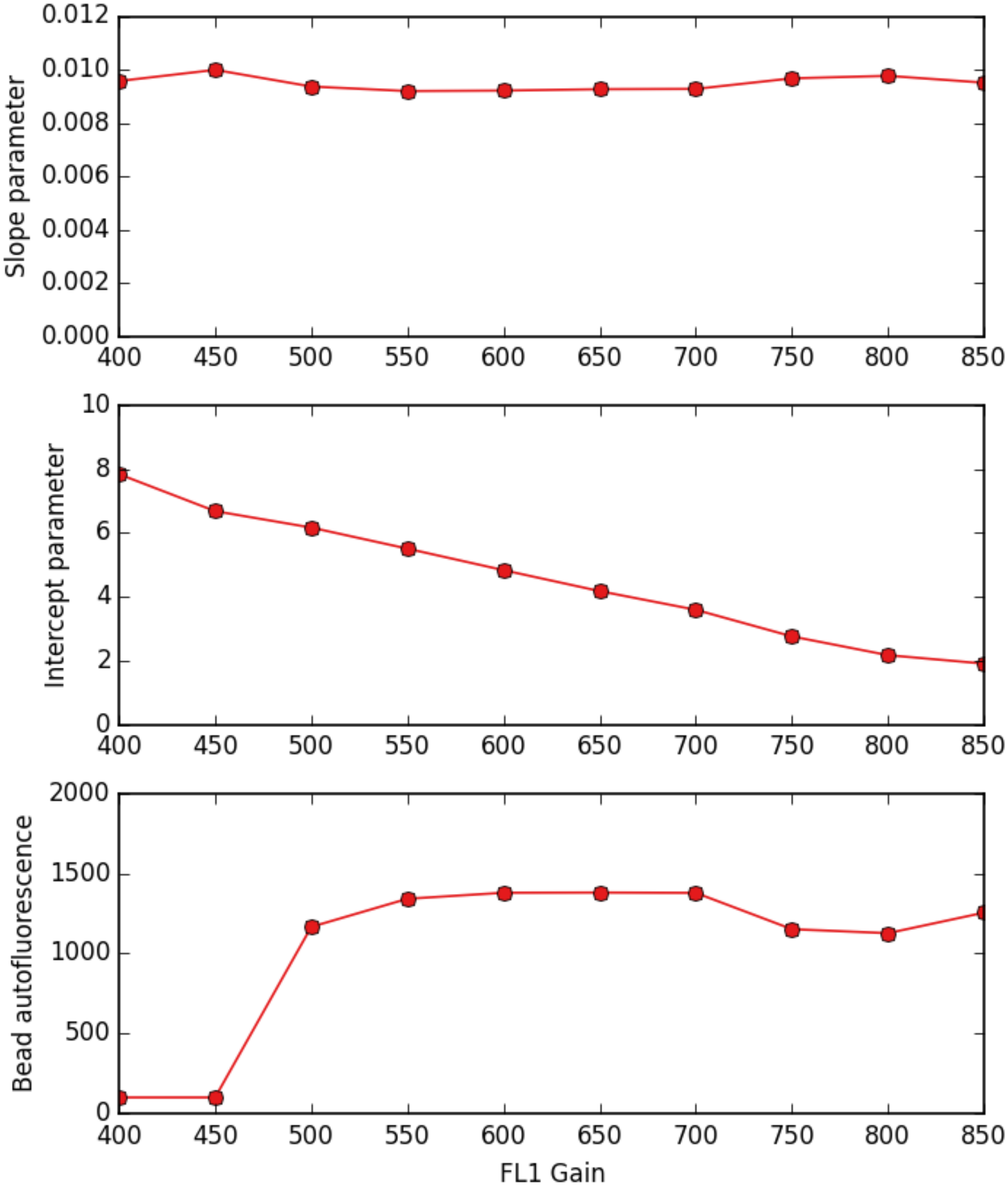
An example of density gating of bacteria is shown in an FSC/SSC density plot (top). In this sample, the SSC acquisition threshold of the flow cytometer was intentionally lowered so that half of the total events corresponded to non-biological debris. Density gating was applied to retain 20% of the events in the densest region. Because cells have a more uniform size than the observed debris, density gating retains mostly cells, which is reflected in the fact that MEFL fluorescence is bimodal before gating, but not after (bottom).

Figure S13. Fluorescence of Calibration Particles vs gain



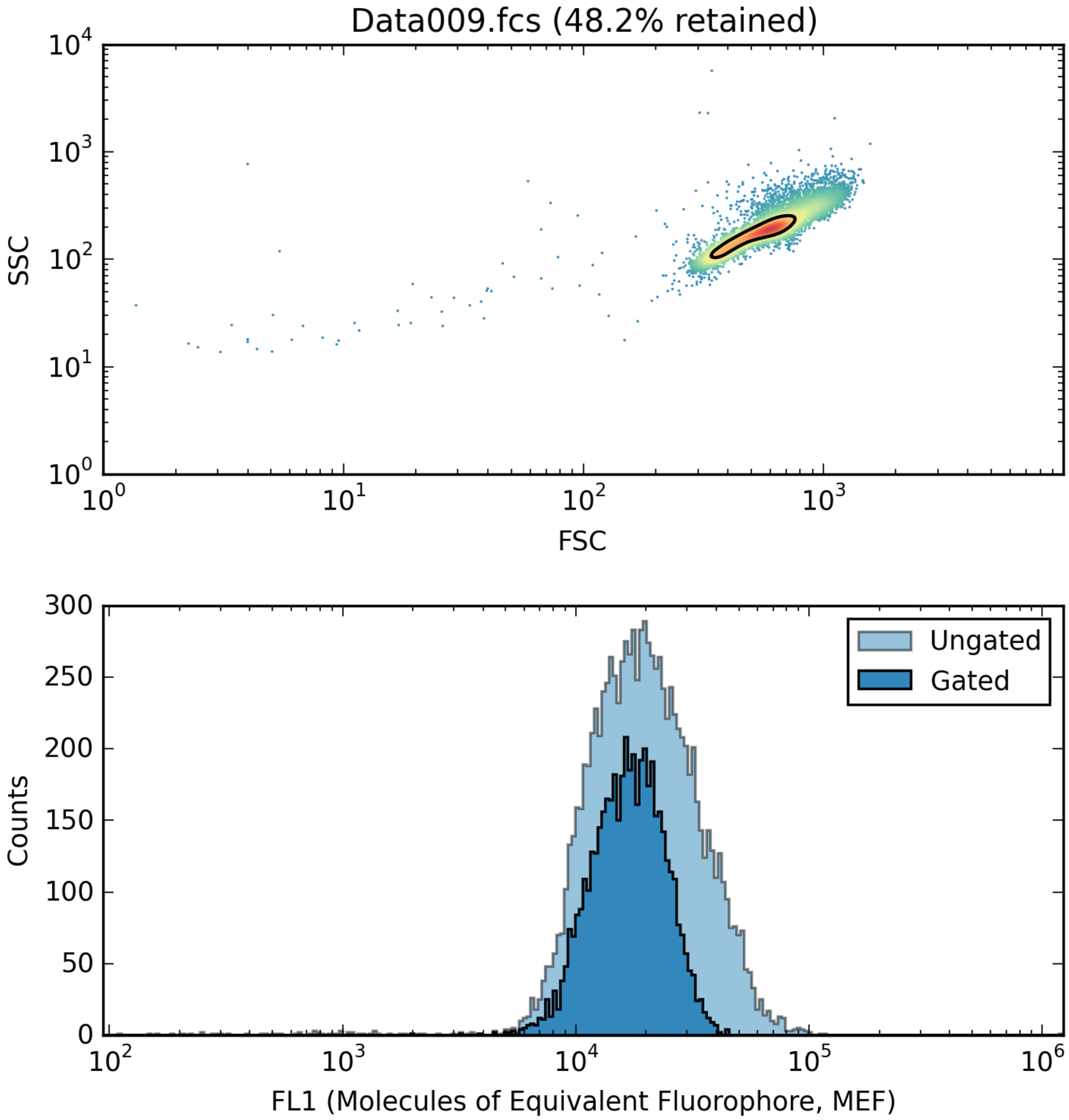
Fluorescence of Calibration Particles vs gain. Samples with Spherotech Rainbow RCP-30-5A calibration particles were taken at 10 different gains, as in **Figure 2A**. The resulting data was gated at 30% and the individual peaks were identified using FlowCal. The median of each subpopulation is shown in this plot for all gains.

Figure S14. Parameters of the Calibration Particles Fluorescence Model vs Gain



Parameters of the Standard Curve vs Fluorescence Channel FL1 Gain. Following peak identification from **Figure S13**, a Calibration Particle Fluorescence Model with Autofluorescence was fit for each different gain using FlowCal. The resulting parameters, as defined in **Note S3**, are shown here as a function of the channel Gain. Note that the slope parameter stays relatively constant, while the intercept parameter varies as a function of the gain, as indicated by equations 5-7 in **Note S3**.

Figure S15. *S. cerevisiae* data processed with FlowCal.



S. cerevisiae expressing the light-responsive PhyB-GBD and PIF3-GAD fusion proteins, and containing the yellow fluorescent protein Venus under the Gal1 promoter (2), was incubated under different intensities of 647nm light for 18 hours. Next, fluorescence was measured via flow cytometry, and the resulting FCS files were analyzed using FlowCal. Data from one of these samples is shown here. Top: Forward Scatter/Side Scatter diagram, along with the gate contour. Bottom: Histogram of Venus Fluorescence, transformed to MEFL.

Table S1: Flow Cytometry Settings used for Data Acquisition with the BD FACScan Flow Cytometer

	FSC	SSC	FL1
Ptac system – Fig. 2A and 2B (beads)	10 (linear)	460 (linear)	Variable (log)
Ptac system – Fig. 2A and 2B (cells)	10 (log)	400 (log)	Variable (log)
CcaS/CcaR v2.0 (beads)	10 (linear)	451 (linear)	600 (log)
CcaS/CcaR v2.0 (cells)	100 (log)	600 (log)	600 (log)
PhyB/PIF3 (beads)	10 (linear)	451 (linear)	611 (log)
PhyB/PIF3 (cells)	0.1 (log)	301 (log)	611 (log)
Ptac system – Fig. 2F (beads)	10 (linear)	460 (linear)	800 (log)
Ptac system – Fig. 2F (cells)	10 (log)	400 (log)	800 (log)

Table S2: Flow Cytometry Settings used for Data Acquisition with the BD FACSCanto II Flow Cytometer

	FSC-A	SSC-A	FITC-A
Ptac system – Fig. 2F (beads)	540 (linear)	419 (linear)	650 (log)
Ptac system – Fig. 2F (cells)	540 (log)	419 (log)	650 (log)

Supporting References

(1) Pedregosa, *et al.* (2011) Scikit-learn: Machine Learning in Python, *JMLR* 12, 2825-2830.

(2) Milias-Argeitis A., Summers S., Stewart-Ornstein J., Zuleta I., Pincus D., El-Samad H., Khammash M., and Lygeros M. (2011). In Silico Feedback for in Vivo Regulation of a Gene Expression Circuit. *Nat. Biotechnol* 29, no. 12 , 1114–16. doi:10.1038/nbt.2018.