**The labscript suite install guide for windows**

**Install Python**

The labscript suite is written in Python. We currently support Python 2.7. We use the **Enthought Academic Python distribution (EPD)** which is free for all academic users. You can request access to the EPD download [here](https://www2.enthought.com/accounts/register/?next=/licenses/academic). Enthought comes with a bunch of useful packages including Qt, scipy and matplotlib. Please **be sure to install the 32-bit version.** Currently some of the other labscript suite dependencies only work under 32-bit mode, which requires everything to be 32-bit (except hardware device drivers, which can be 32-bit or 64-bit depending on the OS).

We recommend installing Enthought to the default location (which should be C:\Python27)

**Install Python packages**

Once python is installed, you should have access to the executable “easy\_install”. Open a command line with administrative privileges, and install the packages listed below by typing

easy\_install <package name> (or easy\_install --upgrade <package name> if it is already installed)

Package list:

* Update h5py (you may not be able to do this from easy\_install, in which case download and install the latest version from [here](https://code.google.com/p/h5py/downloads/list))
* Update pyzmq
* Update pyside
* Update pandas
* Install PyDAQmx

Other things to install:

* PyGtk – [download here](http://ftp.gnome.org/pub/GNOME/binaries/win32/pygtk/2.24/pygtk-all-in-one-2.24.2.win32-py2.7.msi)
* matplotlib with GTK support – [download here](http://www.lfd.uci.edu/~gohlke/pythonlibs/#matplotlib)

**Note for 32-bit windows 7 or 64-bit Windows Vista users:**  
PyDAQmx isn’t very clever when it works out where the NI DAQmx library is. You may need to modify the path located in the file C:\Python27\Lib\site-packages\pydaqmx-1.2.3-py2.7.egg\PyDAQmx\DAQmxConfig.py to add or remove (x86) after Program Files.

**Install the suite**

You will need to place all suite files in C:\pythonlib (so the following folders should exist: C:\pythonlib\BLACS, C:\pythonlib\labscript, etc)

You will also need to create a lab config file: C:\labconfig\<computer hostname>.ini and fill out the required fields. An example file exists in the labconfig folder already. The lab config file needs to contain the paths to labscriptlib and analysislib, which is where you will store python scripts using the labscript API and the lyse API respectively. We store these in C:\user\_scripts\labscriptlib and C:\user\_scripts\analysislib.

Finally, you need to add C:\pythonlib;C:\user\_scripts to your python path. To do this, you need to create/edit the windows system environment variable PYTHON\_PATH. This is done differently on different versions of windows, but on windows 7 you need to go to control panel->system->advanced system settings->environment variables->system variables and edit/create the PYTHON\_PATH variable in there. You will probably need to restart your system after doing this.

To test you have added the paths correctly, start a Python terminal and type:  
import labscript  
If you get no errors, you have done it correctly!

**Install HDFVIEW**

There is a “relatively nice” (it’s written in Java….) viewer for HDF5 files which can be downloaded from [here](http://www.hdfgroup.org/hdf-java-html/hdfview/). Either the 32-bit or 64-bit version is fine!

**Creating a lab (BLACS) connection table**

BLACS will not start unless it has access to a compiled (h5 file) lab connection table, located at a path specified in the C:\labconfig\hostname.ini file.

However, the only simple way to compile a connection table (without already having BLACS running) is to use runmanager to compile a very simple experiment that contains all the required connection table definitions. The resulting h5 file can then be used as the connection table file.

**Setting up runviewer**

Runviewer is currently licensed under the GPL, and will need to be downloaded separately from our website. You need to place the runviewer and pyqtgraph folders in pythonlib, and then run python setup.py build from a command line (setup.py is located in the runviewer folder) to compile the python c extension for your chosen platform.