SOFTWARE INSTALLATION INSTRUCTIONS

The VBAcmd software is known to run on:
- Windows 7 Professional, WinPython 2.7 distribution
- Windows 10, Anaconda 2.7 distribution

Anaconda is strongly recommended over WinPython

1) Install DAQmx

Download and install NI DAQmx at this link:
http://www.ni.com/en-us/support/downloads/drivers/download.ni-daqmx.html

Make sure to download the latest available version; note that even relatively recent versions (e.g. v16) are not compatible with the NI USB-6001 DAQ we employ, so please upgrade to the most recent version in case you have an older one installed.

2) Install Anaconda

Download and install Anaconda at this link: https://www.anaconda.com/download/?lang=en-us
If installing Anaconda for the first time, select python 2.7 64bit.
Select the option to include directories in PATH (ignore warnings).

3) Create vba virtual environment

Launch **cmd**
Update conda and display current version by entering:

```
conda update conda
```

(select **y** if there are new packages available)

```
conda -V
```

Not all conda releases come with conda-build so just in case enter:

```
conda install conda-build
conda update conda-build
```

Create a virtual environment called “vba” by entering:

```
conda create -n vba python=2.7 anaconda
```

Then chose **y** to proceed
Next initialize the vba environment by entering:

```
activate vba
```

Note that your command prompt now has "(vba)" at the beginning.
You will need to activate the environment every time you re-open cmd.
Upgrade sip:

```
conda upgrade sip
```

4) Clone Github repositories

If you don’t already have it, download and install git https://git-scm.com/download/win
Choose or create a new directory to store code and clone these repos by entering:

```
git clone https://github.com/goatsofnaxos/lib.git
git clone https://github.com/pytransitions/transitions.git
```

```
git clone https://github.com/goatsofnaxos/VBAcmd.git
```
Instructions below will refer to the directory that contains these cloned repositories as **<PATH>**.

5) Install PyQt4
Making sure you're in the "vba" environment (see step 3), enter:
```
pip install <PATH>/VBAcmd/PyQt4-4.11.4-cp27-cp27m-win_amd64.whl
--no-deps -qq
```
(Note: This uses a wheel copied to the VBAcmd repository, originally downloaded from here. Alt: **conda install -n vba pyqt=4**)

6) Install PyQwt5
Still making sure you're in the "vba" environment, enter:
```
pip install <PATH>/VBAcmd/PyQwt-5.2.1-cp27-cp27m-win_amd64.whl
--no-deps -qq
```
(Note: This uses a wheel copied to the VBAcmd repository, originally downloaded from here. Alt: build from source here.)

7) Install PyDAQmx
Still making sure you're in the "vba" environment, enter:
```
pip install <PATH>/VBAcmd/PyDAQmx-1.4.3.tar.gz --no-deps -qq
```
(Note: This uses source copied to the VBAcmd repository, originally downloaded from here. Alt: **pip install PyDAQmx**)

8) Add modules to venv path
Navigate to the **<PATH>** directory and enter:
```
conda develop lib transitions VBAcmd
```

9) Configure VBAconfig.py
In the **VBAcmd** folder rename **VBAconfig.py.template** to **VBAconfig.py**, and make edits as needed):

- Uncomment a pair of `rangeForce` and `rangeForceVout` parameters according to whether using the FUTEK IA100 or LEO BODNER load cell amplifier (versions after Jan 2019 all use the LEO BODNER; start with that if unknown),
- Uncomment one if the `rangeLaserVout` parameters according to whether you are using a 500 Ohm or a 250 Ohm scaling resistor (versions after Dec 2018 use 250 Ohm resistor).
- **Connect the VBA's electronics box to your computer via the USB cable.** If this is the first time you're plugging in a NIDAQ device into this computer, it will be named `dev1` in which case no ruther edits are required. If you have already plugged one or more NIDAQs on this computer, you need to look up the new one's ID in NI Measurement & Automation Explorer (MAX) which should have installed with NI DAQmx and replace all instances of `dev1` in `VBAconfig.py` with `devN` where N is the VBA DAQ's device number. Note that you can only look up the device number after you have plugged the VBA's USB into the computer and it has been recognized by the system.

10) Run

Make sure that the VBA's electronics box is connected to your PC via the USB cable, then:

open **`cmd`**
enter: **`activate vba`**
navigate to the **`<PATH>/VBAcmd`**
enter: **`python VBAcmd3.pyw`**