Using the Automated Workstation

Getting Started:
Open the AutomatedLabWorkstation folder in visual studio code. This is currently under 'Summer2022/AutomatedLabWorkstation' within the github repository.

Open and read the example script files located in the example scripts folder, 'Summer2022/AutomatedLabWorkstation/example_scripts'. Now open an IPython terminal in the AutomatedLabWorkstation folder. Use the following command in the standard command prompt in order to open an ipython terminal.

```
ipython
```

Run an example script of your choice using this command in IPython terminal.

```
run example_scripts/basic_experiment.py
```

After the script has finished running, restart the IPython terminal. This is necessary in order to create a new instance of the workstation object. **TODO:** add a better shutdown procedure that doesn't require terminal restart.
Also be sure to close any open instances of ccu-log or ccu-monitor. These will build up if you don't clear them out. They may cause problems if there are too many open at once.

Run LabWorkstation.py by sending the following command. This will import all necessary modules into a new IPython terminal.

```
run LabWorstation.py
```

Now you may begin to use the workstation by sending commands line by line. First create the workstation object. This will begin to perform the start up procedure outlined below.

```
w = LabWorstation("configuration.json")
```

See the sections below for more information on specific aspects of the workstation's behavior.

Technical Python Details

TODO: add a requirements file containing all the packages that are used by the automated workstation. This is helpful for setting up the workstation on another computer. Although this is not an explicit use case.

Additionally, the version of python used is not the most up to date because newer versions do not run on the operating system of the computer in our lab.

Start Up Procedure

When an instance of the automated workstation is created it will perform the following actions:

1. Read the config file.
2. Start the Log.
3. Start the CCU Log+Monitor.
    - If the CCU Log+Monitor were open from a previous instance then it will open new instances of each. This should not interfere with any functionality but may be confusing to visually track.
4. Initialize and document all of the motor hubs / motors.
    - Homes the Elliptec motors
    - Moves creation motors to make the calibrated $|\Phi^+\rangle$ state.
5. Takes a purity measurement.
    - If the purity comes out below a threshold set in the config file then the workstation will throw an error that it has become uncalibrated.

Workstation can also be started in debug mode in which case it will still home the elliptec motors but it will not move any motors to create the $|\Phi^+\rangle$ state or measure purity.

Logs

The workstation will produce a variety of logs. It will place these logs in a directory similar to "temp/07-07-2022/logs".

The main log begins when the workstation is instantiated and persists indefinitely. This log will track every action that is requested by the user or the script. It will also document internal details in the execution of those requests. For example, if the user asks the workstation to reposition a component to a particular orientation then it would log that request as well as logging the internal commands that it sends to the specific motor to be moved.

The main logs are filled to the brim with information so it is often preferable to open these files in excel and filter the "Record Type" column for the specific type of record that you'd like to investigate: Error, Warning, Call, Detail, Message, Return, ...

Upon initialization and shutdown the workstation will also output a file containing a "snapshot" of where it believes every adjustable component is at shutdown. **TODO**: The snapshots are currently not particularly useful because they round to the nearest degree.

The CCU Log will independently temporarily record all detector data.

Data

The workstation will generate a lot of data. It will place these files in a directory similar to "temp/07-07-2022/data".

The exact format of storage depends on what data is being stored. Typically there will be a sub-folder for a particular experiment. Within each of these subfolders there will be files corresponding to individual measurements along with a summary file.

These folders can get full quite quickly and it might be useful to zip/archive old folders now and then.