LVM observing guide

the enclusure ouI would like to try to script some/all of these tasks by the time we start training observers. This list of commands is meant to be a useful guide, especially for early training when we are tired.

General observing: survey mode and Overwatcher

Beginning of the Night

- connect to ZOOM to communicate with other observers
- check weather (see below)
- check webcams to ensure no-one is in the enclosure (see below)
- identify who will drive the telescope
- log into LVM computer and close and reopen all terminal windows (need to do this step to reinitialize Gort with any updates to the software)
- check the observing plan for the night. Check LCO ephemerides.
- open and update the night log (date, weather, observers etc. Find each weeks night log here)
- contact Du Pont observer to let them know you are observing- Slack → lvm_dupont_observing channel
- If you want to request help from Jose, the most efficient way is to write him in Slack to the channel lvm-core-observations-team

- Close any terminal windows and open a new terminal to operate the telescope
- Note:
  Open a terminal window and cd to the observing_scripts directory, then open ipython. We control the telescope from ipython
- *ipython*

Check weather and enclosure

LCO weather page:

- check humidity <80%
- check wind speed <35mph
- check temperature and dew point are more than ~5 degrees different

LCO meteoblue page

webcams (from the VNC console, and now from your own computer with ssh)

- user: lvmlocal
- password: Alpha4-centauri
- This is a local version of the webcam server that does not require two-factor authentication but can only be accessed from inside the VNC window.
- check no-one is in the enclosure when opening/closing
- ensure all lamps off and telescopes parked (pointing down) when opening/closing
- To connect from your own computer, you need to update your .ssh/config file to include the line "LocalForward 18888 camaras-02.lco.cl:443" in the lvm-observer block, and then ssh into lvm-observer to be able to access the webcams. See email [lvm-observers 506] for more information

~~webcams (external) **Please do not use**~~

- ~~user: lvmuser~~
- ~~password: alpha4-centauri~~
- ~~Same service but externally accessible. Requires 2FA which cannot be widely shared.~~

Grafana page

- user: lvm
- pass: lvm00Grafana$

Tile database

- user: sdss5
- pass: SAS password

Night log spreadsheet

Observing with the Overwatcher

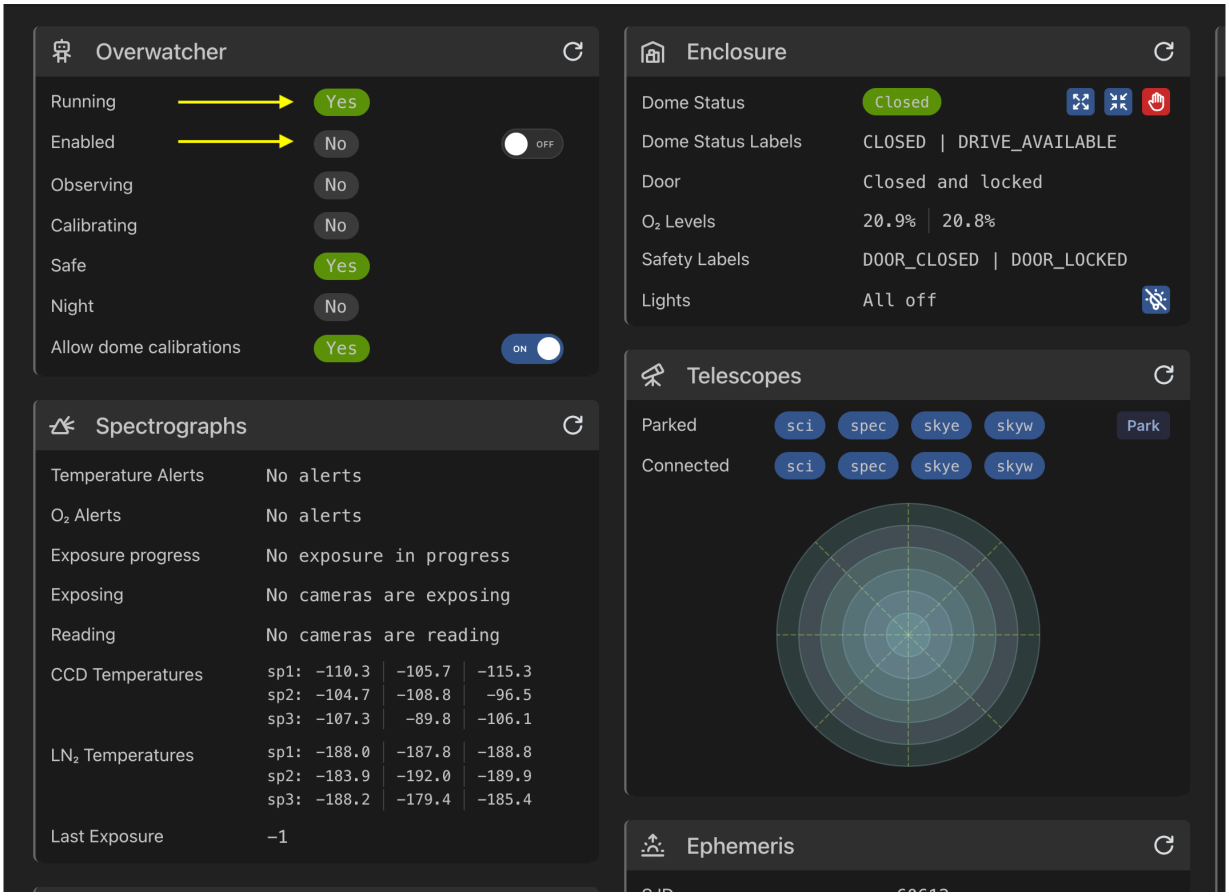
Full instructions for using the Overwatcher can be found here.

We plan to transition to using the Overwatcher software as our default observing mode as this software will be used when we move to fully robotic observations. For now, we will use it to identify all the remaining bugs. As of October 28th 2024, we plan to use Overwatcher to observe while we monitor it, with it being supervised by the Du Pont observers for only 1-2 hours during the night between observers ONLY when conditions are good and stable. If the conditions are not good or stable, please close and leave the long term calibration scripts running.

The Overwatcher webtool can be found https://lvm-web.lco.cl/overview/.

- You can access the Overwatcher from your own browser, no need to use VNC or the ssh (though we recommend you connect via ssh so that you can access the webcams)
- Note that when observing with Overwatcher, you should not use ipython to control the system. That will result in an error message.
- The first time you visit, you need to give a username and password. These are the same as for the SAS (sdss5 and the password we are not allowed to type anywhere 😛). Note that the password is the same as for logging into the observing machine, but I think with the second P capitalised.
- To work with the Overwatcher, click on the yellow lock in the top right corner. It will ask for a password, that is **let-me-in**
- On the Overview page in the Overview section (see below), you should see that the Overwatcher is Running (i.e. on and ready to go), and you can enable it/start it running by clocking the slider to the right of "Enabled". This should change the icon to a green "on" next to Enabled. These items are marked in the image below with yellow arrows.
- Once enabled, the Overwatcher will automatically open the enclosure at sunset, take the twilight flats (script still not quite right, some flats are saturating), close, run the daily calibrations 30 minutes after sunset, then open at the end of twilight and start observing. All these processes will be reflected in the Overwatcher page.

  Overwatcher during day, when enclosure closed:



Overwatcher

Running	Yes
Enabled	No (OFF)
Observing	No
Calibrating	No
Safe	Yes
Night	No
Allow dome calibrations	Yes (ON)

Spectrographs

Temperature Alerts	No alerts
O_2 Alerts	No alerts
Exposure progress	No exposure in progress
Exposing	No cameras are exposing
Reading	No cameras are reading

CCD Temperatures

```
sp1: -110.3 | -105.7 | -115.3
sp2: -104.7 | -108.8 |  -96.5
sp3: -107.3 |  -89.8 | -106.1
```

LN_2 Temperatures

```
sp1: -188.0 | -187.8 | -188.8
sp2: -183.9 | -192.0 | -189.9
sp3: -188.2 | -179.4 | -185.4
```

Last Exposure -1

Enclosure

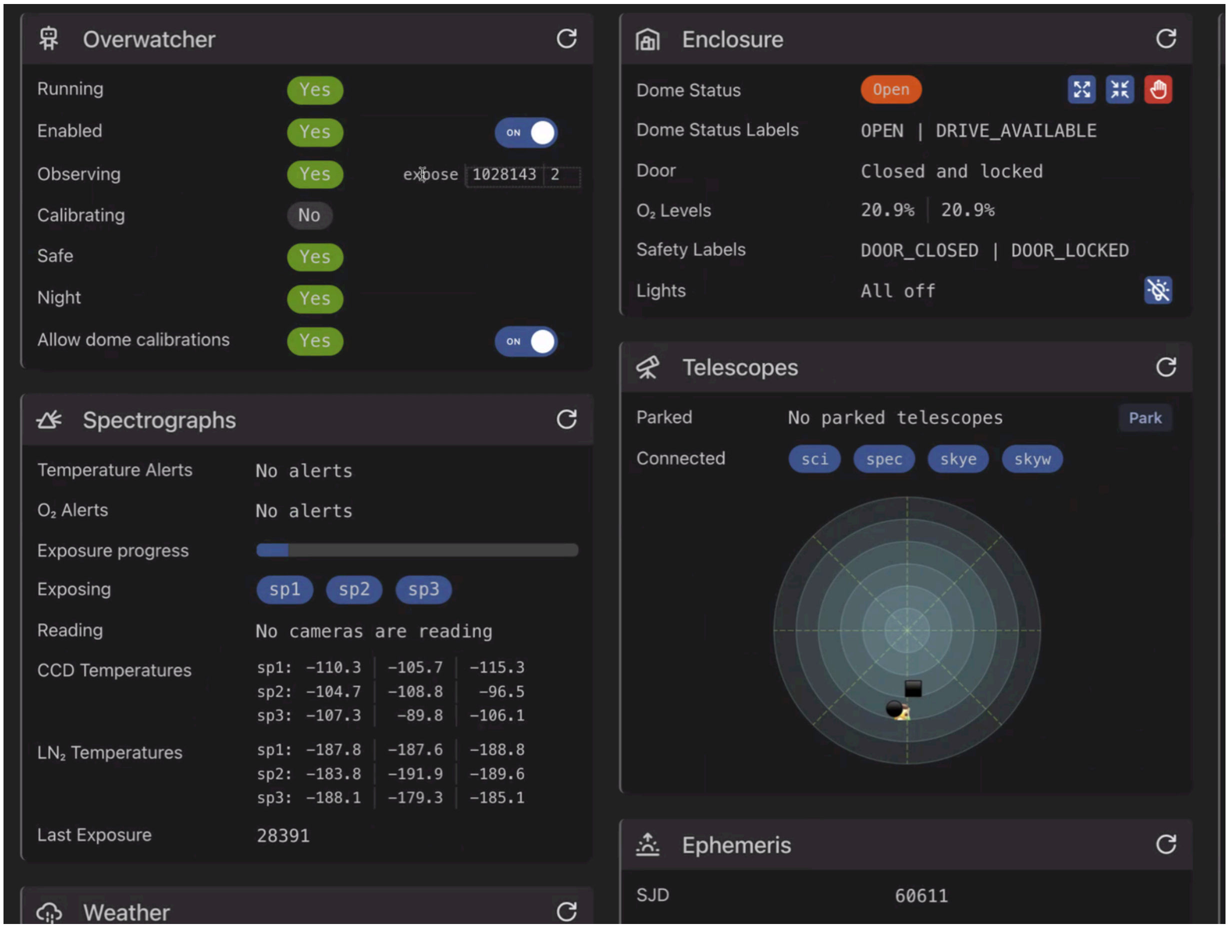
Dome Status	Closed
Dome Status Labels	CLOSED | DRIVE_AVAILABLE
Door	Closed and locked
O_2 Levels	20.9% | 20.8%
Safety Labels	DOOR_CLOSED | DOOR_LOCKED
Lights	All off

Telescopes

Parked	sci spec skye skyw
Connected	sci spec skye skyw

Ephemeris

How the overwatcher looks at night when it is enabled, the enclosure is open and it is observing:



Overwatcher

Running	Yes
Enabled	Yes (ON)
Observing	Yes expose 1028143 2
Calibrating	No
Safe	Yes
Night	Yes
Allow dome calibrations	Yes (ON)

Spectrographs

Temperature Alerts	No alerts
O_2 Alerts	No alerts
Exposure progress	
Exposing	sp1 sp2 sp3
Reading	No cameras are reading

CCD Temperatures

```
sp1: -110.3 | -105.7 | -115.3
sp2: -104.7 | -108.8 |  -96.5
sp3: -107.3 |  -89.8 | -106.1
```

LN_2 Temperatures

```
sp1: -187.8 | -187.6 | -188.8
sp2: -183.8 | -191.9 | -189.6
sp3: -188.1 | -179.3 | -185.1
```

Last Exposure 28391

Weather

Enclosure

Dome Status	Open
Dome Status Labels	OPEN | DRIVE_AVAILABLE
Door	Closed and locked
O_2 Levels	20.9% | 20.9%
Safety Labels	DOOR_CLOSED | DOOR_LOCKED
Lights	All off

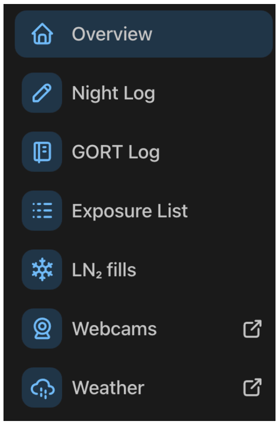
Telescopes

Parked	No parked telescopes
Connected	sci spec skye skyw

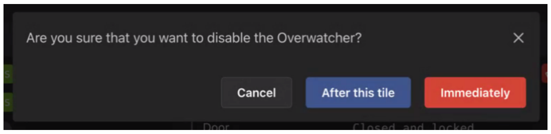
Ephemeris

SJD 60611

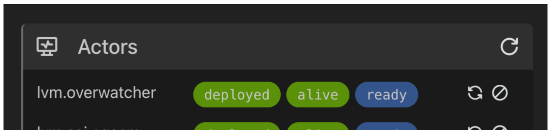
- On the left hand panel you can check the **GORT Log**, which displays the log we are used to seeing in the terminal (updated every 30s), and check the list of exposures in **Exposure List** (same as the old list_lvm output). Finally, in the **Night log** tab you can add notes to the night log, such as if anything strange happened, weird things with the weather, earthquake etc.



In case of problems, one can disable the overwatcher and go back to observing in ipython. To do so, move the slider next to "enabled" to off. This will trigger a pop up asking if you want to stop and abort the current exposure, or wait and stop after the current exposure. Once stopped, move to ipython and run a cleanup before observing again, or click on the cleanup button in the Overview section of the Overwatcher.

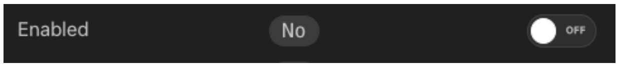

Are you sure that you want to disable the Overwatcher?

You can also force it to stop by clicking the circle with a line next to lvm.overwatcher in the Actors section. Then, again, click the cleanup button or go to ipython and run a clean up before observing again.

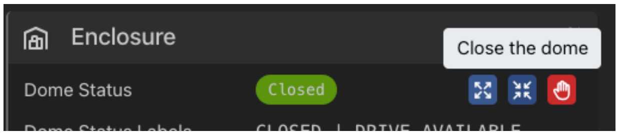

Actors
lvm.overwatcher deployed alive ready

Closing for calibrations or bad weather

If you want to close to run the long term calibrations or for bad weather etc, first disable the Overwatcher. In the pop up you can decide whether to let it finish that exposure (e.g. closing for calibrations), or to terminate it now (e.g. sudden change in weather, closing before the Overwatcher closes automatically)


Enabled No OFF

Note that once disabled, the Overwatcher will leave the system idle. You then need to close the enclosure using the middle button next to dome status. hovering over the button will show the option to close the dome.


Enclosure
Dome Status Closed

You can then go to ipython in the VNC window and run the calibration scripts in the usual way.

End of the night

We've had a few issues with Overwatcher reopening the dome at the end of the night. Therefore, please follow the following procedure for now until we know we have found and fixed the bug:

- Let the overwatcher close the dome. Do **not** disable the overwatcher yet.
- Wait 5-10 minutes after twilight and confirm the dome does not open. If that's the case then disable the overwatcher and call it a night.
- If the overwatcher tries to open again after closing, the easiest is to probably disable the overwatcher, let the dome fully open, and close the dome from lvm-web button. Then confirm the dome is closed, the overwatcher disabled, wait for a few minutes to confirm that nothing weird is happening, and then call it a night.
- If the overwatcher misbehaves very much, just kill it (see instructions above to stop it with the actor) and close the dome.

General observing: survey mode with manual observing

startup the facility and open enclosure

Close any terminal windows and open a new terminal to operate the telescope

```
await g.startup()
```

startup will do the following:

- startup (homing devices, AGC darks)
- calibration sequence
- ask if you want to open enclosure
- if open, will focus guide cameras

Note: when opening at twilight, it's normal for the focus to fail, since there aren't any stars visible to focus on

run twilight flats

Run twilight flats in the evening at sunset (optional) and sunrise.

```
run twilight_sky_cycle_fibers_v3.py
```

Script should be run starting a few minutes BEFORE sunset, or starting ~20minutes before sunrise (and will finish a few minutes AFTER sunset). see ephemeris here: https://www.lco.cl/ephemeris-for-lco/

This script automatically adapts the exposure times, and will attempt to take a series of 12 exposures, cycling through the 12 spectrophotometric fiber mask positions.

run daily quick calibrations

Close the enclosure and run the daily quick calibrations to monitor the health of the instrument daily

```
await g.shutdown()
run quick_cals.py
run bias_7_sequence.py
```

Open enclosure and observe survey tiles

```
await g.startup()
await g.observe()
```

To change the exposure time of a tile, for example if it is saturated after 900s, use:

```
tile = Tile.from_scheduler(tile_id=TILE_ID)
await g.observe_tile(tile, exposure_time=300)
```

The command above will run on a loop until it's stopped. To stop observing the survey tiles, do the following:

```
ctrl + c
await g.cleanup()
```

Close enclosure and end observing for the night

At the end of the night, or when the weather takes a turn for the worse, close the enclosure

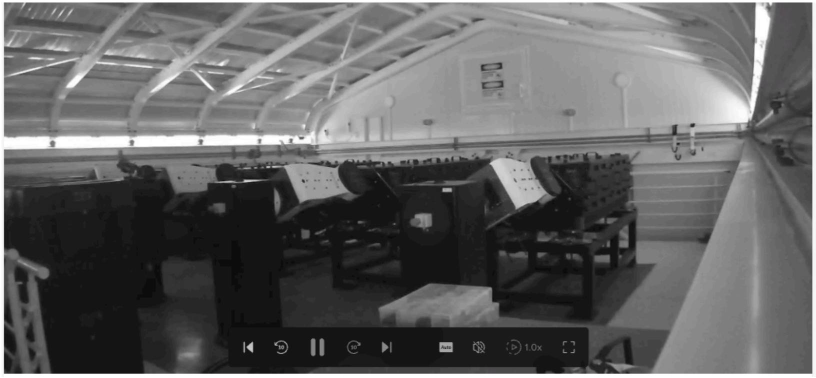
await g.shutdown()

UPDATE 13th April 2024

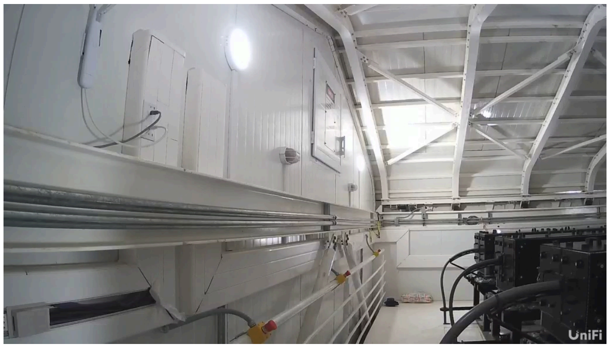
It was reported that sometimes the enclosure is not fully closed despite being marked as closed on the LVMweb and Grafana pages. To avoid any damage to the telescopes and instruments, please follow the procedure below to ensure the enclosure is closed at the end of the night:

- Run the shut down script to close the enclosure
- When finished, check the webcams
    - await g.enclosure.lights.telescope_bright.on()
      await g.enclosure.lights.telescope_bright.off()
      Or
    - await g.enclosure.lights.telescope_red.on()
      await g.enclosure.lights.telescope_red.off()
    - Check the telescopes are in the park position (pointing towards the floor)
    - Check that the enclosure looks closed
    - Turn on the lights and check the enclosure on the webcams from the outside (LVM-EXT01) for any obvious leaks of light where the enclosure closes. Below are the commands to turn the bright and red lights on and off
- If the enclosure is still open, try running : await g.enclosure.close(force=True)
- If it still won't close, please inform the Du Pont observer and ask them if they can manually close the enclosure.
- If there is no response from Du Pont, please contact Felipe Besser and Abner Zapata (both on the SDSS5 slack) to ask them to check the enclosure. Felipe and Abner work alternating weeks, so please message both of them if you don't know who is on the mountain.
- I would appreciate it if you could also let us know in this channel and in the night log (notes section at the top) when the enclosure doesn't fully close so we get an idea of how often this is happening.

Below I will add some reference images of what the enclosure should look like from the inside and outside if properly closed



new webcam view, enclosure is not quite closed after await g.shutdown() :



After running await g.enclosure.close(force=True) , enclosure is now fully closed

Check header information

```
list_lvm    (or ~/list_lvm)
```

(note- run this one in the shell, outside of ipython)

Show latest exposure information

```
exp
```

(note from Kathryn - I think this doesn't work anymore?)

Turn enclosure lights on and off

```
await g.enclosure.lights.telescope_bright.on()
await g.enclosure.lights.telescope_bright.off()
```

If you are not able to execute the above commands, please check the webcams and issue the command

```
await g.enclosure.lights.telescope_bright.toggle()
```

To turn on and off the red lights, you can use the following command:

```
await g.enclosure.lights.telescope_red.on()
await g.enclosure.lights.telescope_red.off()
```

Turn off lights in the utility room (LVM-ITCAM) or spectrograph room (LVM-TEL02)

Occasionally the lights are left on by accident in the utility room that shows up on the LVM-ITCAM webcam. you can turn off the lights using the following commands:

```
await g.enclosure.lights.utilities_room.off()
await g.enclosure.lights.spectrograph_room.off()
```

Focus telescope

To run the focus outside of the startup script:

```
await g.guiders.focus()
```

```
await g.guiders.focus(guess=36.0) -->select starting estimate
await g.guiders.focus(step_size=0.25, steps=5) -->define step size
await g.guiders.sci.focus(guess=36.0) -->only science telescope
```

If you want to check that the focus has worked, take a quick exposure:

```
await g.ags.expose(5)
```

Other useful commands

Old restart commands

```
await g.telescopes.restart()
await g.specs.restart()
await g.ags.restart()
```

old/original calibrations sequence

To run the daily calibrations outside of the startup script:

```
await g.specs.calibrate(sequence='testcal')
```

Every night we should aim to carry out the hartmann shutter tests (with the dome closed):

```
run hartmanns_with_HgNe_Ne_Ar.py:
```

second generation calibration sequence

```
await g.startup()
```

Open enclosure

To open the enclosure outside of the startup sequence (i.e. only open the enclosure, no movements of the telescopes etc):

```
run open_sequence.py
```

Run Science Scripts

check "observing_scripts" directory and look for the Science Verification scripts "SV_*.py".

Alternatively, check tasks in the nightly Commissioning Plan pages.

Take an exposure

```
ra = XXXX
dec = XXXX
target_name= 'XXXX'
```

[Use one of the following options]

```
tile = Tile.from_coordinates(ra=ra, dec=dec, sky_coords={'skye': (ra,dec), 'skyw': (ra,dec)}, spec_coords=[(ra,dec)], object=target_name) #point all 4 telescopes to the same target, 1 fibre of sectrograph used
tile = Tile.from_coordinates(ra=ra, dec=dec, sky_coords={'skye': (ra,dec), 'skyw': (ra,dec)}, spec_coords=[(ra,dec),(ra,dec),(ra,dec),(ra,dec)], object=target_name) #point all 4 telescopes to the same target, 4 sdt fibres used
tile = Tile.from_coordinates(ra=ra, dec=dec, sky_coords={'skye': (ra,dec), 'skyw': (ra,dec)}, spec_coords=[], object=target_name) #std telescope idle
tile = Tile.from_coordinates(ra=ra, dec=dec, sky_coords={'skye': (ra,dec), 'skyw': (ra,dec)}, object=target_name) #std stars selected from database for tile closest to the target frame
```

To take multiple exposures, use a loop:

```
for i in range(0,20):
    await g.specs.expose(0,flavour='bias')
```

Close enclosure/End of night

~~await g.nps.calib.on('Quartz') #ONLY if we are confident other telescopes not observing. otherwise, skip this step~~
~~await g.telescopes.park()~~
~~await g.nps.calib.all_off()~~
~~await g.enclosure.close()~~

```
await g.shutdown()
```

Stop an exposure

```
ctrl+c
```

```
await g.cleanup()
```

~~*await g.guiders.stop()*~~
~~*await g.specs.abort()*~~

Check header information

```
list_lvm    (or ~/list_lvm)
```

(note- run this one in the shell, outside of ipython)

Show latest exposure information

exp

Turn enclosure lights on and off

await g.enclosure.lights.telescope_bright.on()

await g.enclosure.lights.telescope_bright.off()

Where is the Spec fiber selector?

This is the mapping between Spec fiber names and fiber selector encoder values (readable from the "Motor Controllers"):

```
mask_positions:
P1-1: 1650
P1-2: 215
P1-3: 16610
P1-4: 15180
P1-5: 13640
P1-6: 12205
P1-7: 10645
P1-8: 9205
P1-9: 7665
P1-10: 6205
P1-11: 4645
P1-12: 3210
P2-1: 15885
P2-2: 14335
P2-3: 12885
P2-4: 11375
P2-5: 9915
P2-6: 8395
P2-7: 6940
P2-8: 5405
P2-9: 3945
P2-10: 2395
P2-11: 935
P2-12: 17340
```

Twilight flats

run twilight_sky_cycle_fibers_v2.py

You can edit the main call to have it do either the primary or secondary set of standard fibers.

This should be run starting almost directly after sunset or ~20m before sunrise (https://www.lco.cl/ephemeris-for-lco/). if you start early or late then the integration times will not be well matched to the sky brightness.

General strategy for twilights:

At sunset (pointed west, az=270):

```
await g.telescopes.home()
await g.telescopes.goto_coordinates_all(alt=40, az=270, altaz_tracking=False) # points west
await g.specs.expose(300, flavour='flat') # 20m after sunset
```

At sunrise(pointed east, az=90):

```
await g.telescopes.home()
await g.telescopes.goto_coordinates_all(alt=40, az=90, altaz_tracking=False) # points east
await g.specs.expose(300, flavour='flat') # 20m before sunset
```

rough exposures times:
- 30s when 10m away from sunset/sunrise
- 300s when 20m away from sunset/sunrise