

# MHS control of PFI calibration lamps

Craig Loomis

2020-02-19

## MHS control of PFI calibration lamps

We propose to control the PFI lamp programs running on the Raspberry Pi from a `pfilamps` MHS actor running on one of the ICS Linux boxes. That actor will provide the obvious lamp control commands and will generate MHS keywords to populate the FITS headers. Some of the work has been done, but not all.

One PFS git repository holds the actor code, the bash scripts running on the pi, and the TCP server which makes the script logic accessible from the actor: `git@github.com:Subaru-PFS/ics_pfilampsActor.git`

### MHS commands

Since we want to expose the existing logic, the MHS commands merely rephrase the bash functions. The lamp timing must fit into SPS exposures; that is described below.

- `setupLamps [ar=S] [hgcd=S] [kr=S] [ne=S] [xe=S] [qth=S]`
- `go`
- `stop`
- `status`

There will also be some engineering commands, including the usual `raw` MHS command to run an arbitrary command within the pi shell environment.

### MHS keywords

The `pfilamps` actor will generate a `lampState` keyword indicating that the lamps are ready to be used after the configuration step, so that the shutters can be opened at the right time. Besides that, it will generate a set of lamp keywords sufficient for populating FITS headers.

We want to put lamp status into the FITS headers (and perhaps some opDb table). The DCB actor already generates cards for its lamps (e.g. `W_AITNEO`) but we will *not* use the same card names for similar lamps. The metadata ingest code for the DRP will need to understand both calibration systems.

For any given SPS exposure, several arc lamps might be turned on. In order to crudely balance the illumination from the different lamps the calibration system allows setting different on times for each lamp. We record these times in place of simple T/F logicals.

Each calibration lamp provides a measured photodiode voltage for some proxy line. Due to wavelength overlaps these are not entirely independent, but we plan to measure the crosstalk (once) and solve for the individual components. We will record all the photodiode voltages as captured at the start of the exposure's illumination, shortly after the `go` command is sent and the lamps turned on.

For a given lamp (neon, say):

- `W_CLNET`, a float, states the requested lamp on time. 0 means off.
- `W_CLNEV`, a float, shows the lamp's component of the measured photodiode voltage.

The HgCd lamp is slightly different. We *control* a single HgCd lamp, but *measure* a mercury line and a cadmium line independently. So for that lamp, there will be:

- `W_CLHGCC`, the float for the commanded HgCd lamp state.
- `W_CLHGV`, the measured Hg photodiode voltage
- `W_CLCDV`, the measured Cd photodiode voltage

## Exposure timing

For exposure time book-keeping we have to deal with lamp warmup, physical shutters, H4RG read clocking, etc. We *record* all the component timestamps in the headers, but have to select the appropriate ones for the primary timestamps (`EXPTIME`, `DARKTIME`, `DATE-OBS`).

For science exposures, the primary timestamps come from the ENU actor, which is directly controlling the (slow) shutter.

For darks, the timestamps come from the CCD wipe and read transitions, or from the H4RG ramp reset and read frame clocks.

For arcs and flats, we want to use the longest illuminated time, and obviously to record the individual lamp times. The sequence is expected to be:

1. `iicActor` configures the lamps. This would turn on the HgCd lamp.
2. `spsActor` waits until lamps are declared to be ready.
3. `spsActor` opens shutter for a bit longer than the longest lamp time.
4. immediately after shutter is fully open, and at the edge of an H4 read, `spsActor` sends 'go' to the lamps.

The details of book-keeping and processing for an exposure with the HgCd lamp illuminated still need to be decided on. The issue is that the HgCd lamp will already be on when the shutter is opened, before the `go` command is sent. The illumination time will thus be different from the other lamps. More importantly

there will be a (known) gradient in the spectral direction due to the  $\sim 1$ s shutter motion. Given the length of the arc exposures, we do have to account for the gradient.