TELLIE automation plan

We believe the best way to feasibly run TELLIE automatically in the near future (without the need for human intervention) is by monitoring the PIN values, which are already fed (live) to ORCA. Analysing all TELLIE runs with a fixed IPW (no "modern" tuning runs included, where the run rollover is suppressed), it was found that the linear correlation between NHit and PIN is reasonably stable over time, and likely to change only due to changes in the detector (trigger thresholds etc.) or the TELLIE hardware (recalibrating PIN boards, power supply issues). A clear example of this is shown below, although the statement seems to hold for most fibres (barring the noted exceptions).

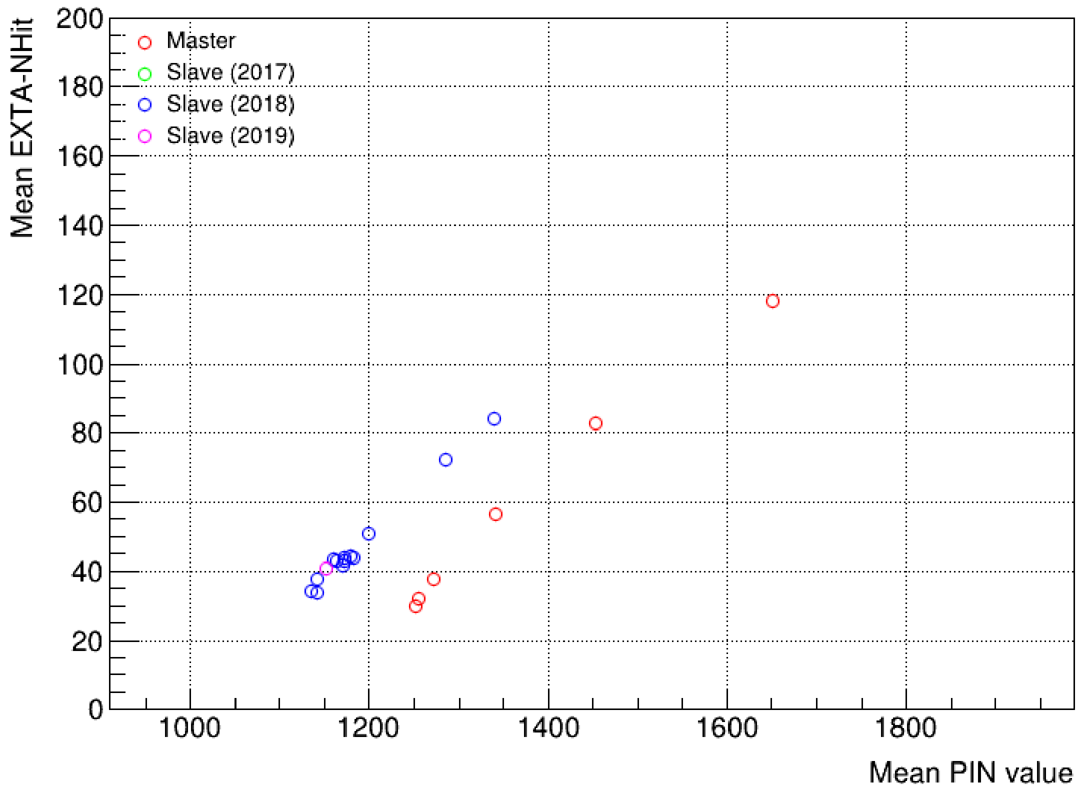

Figure 1: TELLIE pulse intensity for channel 95 monitored externally (EXTA-NHit) and internally (PIN reading) at different intensity setpoints at different points in time, showing a linear correlation for both Master and Slave mode.

There are certain points that need looking at (PIN outliers probably due to us recalibrating the boards last week), but in general there is a consistent trend: if a LED driver board + LED cone + PIN driver board works consistently and we don't mess with the box / cables / fibers, then the hypothesis "NHit vs PIN is a linear dependency and can be used to calibrate the light output of TELLIE" appears to be correct.

In an ideal world, we would get a "live NHit" reading sent back to Orca to be certain (i.e. check both NHit and PIN continuously while running), but failing to do so we should start the automation development under the aforementioned assumption that this linear correlation is stable, and continue as described on the following pages.

– Martti, 02/05/2019

Take Calibration curves

1. Start "TELLIE_calibration" run, initialise a channel (Slave mode)
2. Select a high IPW setting (e.g. last PCA setpoint + 1000)
3. Fire 1000 pulses at 1 kHz (good enough stats), do not roll over run
4. Decrease the IPW in steps of 100 and repeat - PIN response should be flat
5. When the LED starts emitting light, observe an increase in PIN (and NHit)
6. When the PIN has increased significantly (e.g. by 300), stop decreasing IPW.
7. Increase the IPW in steps of 10 and repeat until PIN reaches noise levels
8. Roll over run and repeat for next channel/fibre from step 1.

This must be done with the detector at HV, and will take about 5 minutes per channel (maybe less), with each run corresponding to a single TELLIE channel. Within a run, each subrun should have an individual IPW setpoint and PIN reading associated to it.

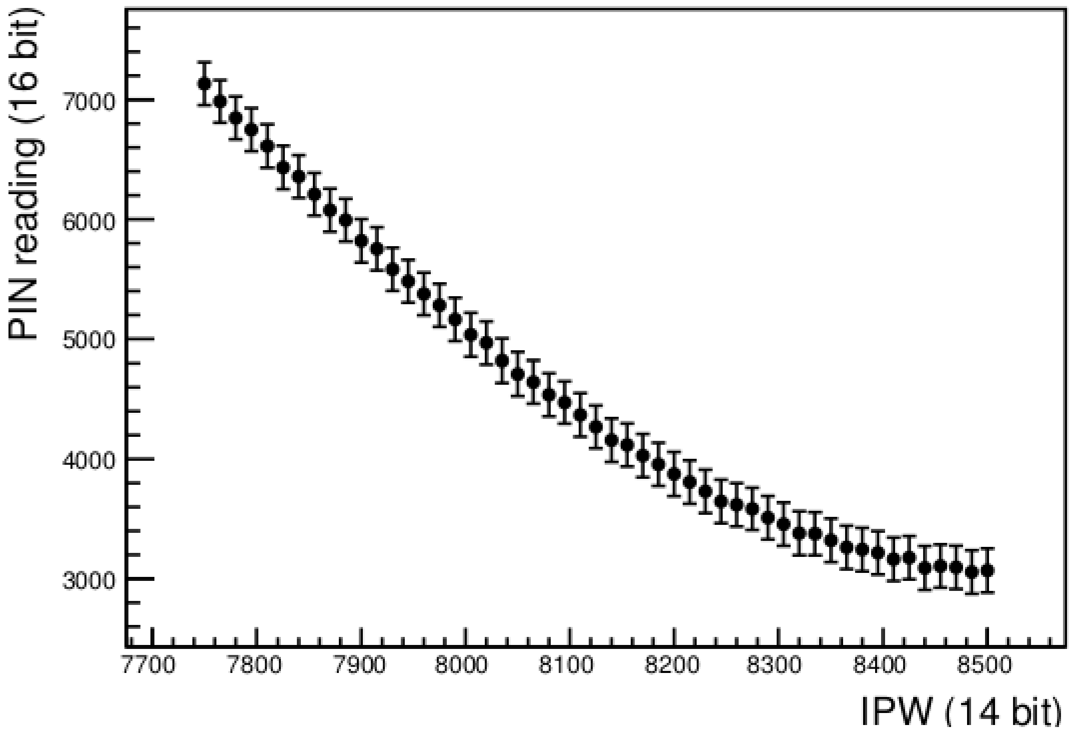

Figure 2: Calibration curve for a typical TELLIE channel, showing only the low-intensity range (plot taken from the TELLIE hardware manual). The PIN reading (which measures the LED light output) is inversely proportional to the IPW setpoint (which determines the LED pulse intensity) up until the LED emission threshold, where the PIN response becomes flat (noise level ~3000). It should be noted that the LED intensity tends to drift significantly over time, with the same IPW values producing vastly different output. For this reason, the PIN readings should be trusted rather than the IPW setpoints, as these appear to give a much more stable estimate of the light output (see Fig. 1).

Run nearline processor

The average EXTA-NHit can be extracted from data only if the detector was at HV during data taking. In this case, a nearline processor will generate the corresponding NHit-vs-PIN plot (see Fig. 1), fit a straight line through the data points and write the corresponding parameters to a database. These parameters should include:

- LED emission threshold (lowest IPW value producing ~0 EXTA-NHit)
- PCA intensity setpoint (IPW value corresponding to ~40 EXTA-NHit)
- Internal noise level (PIN value seen below LED emission threshold)
- Fit parameters to match external NHit to internal PIN reading (y = ax + b)

This processor does not yet exist, but would be fairly easy to implement (Martti).

Take PCA dataset (dedicated TELLIE run)

1. Start "TELLIE" run, initialise a channel
2. Select IPW setting corresponding to PCA intensity setpoint (from database)
3. Fire 1000 pulses at 1 kHz (tuning run), do <u>not</u> roll over run
4. Use fit parameters from database to extrapolate NHit from PIN (y = ax + b)
   - If EXTA-NHit is too high (>45), <u>increase</u> IPW by 10. Repeat from step 3.
   - If EXTA-NHit is too low (<40), <u>decrease</u> IPW by 10. Repeat from step 3.
5. When the extrapolated NHit is in the desired range (40-45), save IPW setpoint.
6. Restart into "TELLIE_PCA" run, initialise same channel.
7. Fire 200’000 pulses at 1 kHz (PCA run) with the obtained IPW setpoint.
8. Restart into “TELLIE” run and repeat for next channel/fibre from step 1.

This is what is currently done manually by the operator while taking a TELLIE PCA dataset. High levels of concentration are required over a long course of time, as an entire dataset (95 channels, at least 5 min/channel) takes about 10 hours to complete. Automating this would greatly facilitate things, and following thorough testing open up the possibility of continuous PCA data taking (during physics runs).

Take PCA dataset (continuously during physics run)

1. Follow similar procedure to above, but fire at a reduced rate of 50 Hz while ensuring zero overlap with PulseGT (alternate between PGT+EXTA triggers, i.e. the combined rate would be 100 Hz).
2. Continuously update IPW setpoint (e.g. every 1000 pulses / 20s) to ensure the desired PCA intensity setpoint (NHit ~40, PIN ~1150 in Fig. 1) are maintained despite potential intensity shifts.
3. Only switch to the next fibre if the previous run rolled over by itself.

A normal physics run (60 min) would then contain 180’000 TELLIE pulses, with none of the EXTA triggers being stolen by PulseGT triggers (it will still be possible for other triggers to “steal” the EXTA trigger, but these can be flagged due to the precise timing of the triggers sent by TUBii). This would provide comparably high statistics to the dedicated PCA data taking (1 run/fibre will suffice), while simultaneously taking physics data.

A full PCA dataset would then require 95 completed physics runs (~4 days of straight data taking), which is usually achieved every week (leaving 73 hours for detector maintenance, calibration using deployed sources and unforseen issues).

The processor for extracting the PCA constants using TELLIE data has been developed, though automated running on the nearline still needs to be implemented (Michal).

Moving towards this option would be incredibly helpful, as it would increase the detector livetime while significantly reducing the need for TELLIE operators to be present. Intervention by TELLIE experts would only be required when things go wrong, which should become increasingly rare once the automatisation is tried and tested.