Bad Pixel Status

Dark data processing algorithms include pixel status evaluation which is saved in calibration files of type pixel_status. This note explains how to find definition and statistics of the bad pixel bits.

Algorithm

Dark data processing

Dark data processing is originaly intended for base signal level a.k.a. pedestal evaluation. In addition quality of pixel response is also evaluated. This algorithm evolved for a long time. The most advanced version implemented for calibration of jungfrau panels is described here.

1. dark raw data is accumulated in array `block[nrecs1,<shape-of-data>]` for a portion of events nrecs1=50. It would be nice to accumulate it for entire volume of events (nrecs=1000), but some of detectors are too big and causes problem with memory.
2. pre-process data from block, use median and quantile for fraclo=0.05 and fcachi=0.95 fractions of spectral events to estimate mean value and gate intensity limits, respectively.
3. use **gated average algorithm** to process events in `block` and all other events (nrecs=1000) requested for dark processing.
4. at the end of the event loop evaluate mean, rms, max/min values and process a few statistical cumulative arrays which produce information about bad pixels.

Bad pixel status

For arrays of per-pixel mean intensity and rms, obtained in gated average algorithm, we use parameters of absolute limits and number of sigma for low and hight range

```
int_lo=1, int_hi=16000, intnlo=6, intnhi=6
```

```
rms_lo=0.001, rms_hi=16000, rmsnlo=6, rmsnhi=6
```

and dynamically evaluate limits for good parameters from spectra.

evaluate_limits

```
def evaluate_limits(arr, nneg=5, npos=5, lim_lo=1, lim_hi=16000, ...):
    ave, std = (arr.mean(), arr.std())
    lo = ave-nneg*std if nneg>0 else lim_lo
    hi = ave+npos*std if npos>0 else lim_hi
    lo, hi = max(lo, lim_lo), min(hi, lim_hi)
    return lo, hi
```

Bad pixel bits assignment

status 0: good pixel

status 1: pixel rms exceeds its maximal value for good pixels defined by rms_hi=16000, rmsnhi=6

status 2: pixel rms lower than its minimal value for good pixels defined by rms_lo=0.001, rmsnlo=6

status 4: pixel intensity exceeds int_hi=16000 in more than fraclm=0.1 fraction of events

status 8: pixel intensity lower than int_lo=1 in more than fraclm=0.1 fraction of events

status 16: pixel average intensity exceeds its maximal value for good pixels defined by int_hi=16000, intnhi=6

status 32: pixel average intensity lower than its minimal value for good pixels defined by int_lo=1, intnlo=6

status 64: **for jungfrau only** pixel with bad gain mode switch.

Bad pixel statistics

Dark data processing scripts dump in the log file information about bad pixel statistics, for example in case of jungfrau:

log file information about bad pixel statistics

```
raw data found/selected in 999 events
[I] L0688 begin data summary stage
[I] L0111 evaluate_limits RMS: ave=4.783 std=1.303 limits low=0.001 high=12.604
[I] L0111 evaluate_limits AVE: ave=14191.873 std=457.903 limits low=11444.455 high=16000.000

[I] L0733 bad pixel status:
status 1: 244 pixel rms > 12.604
status 2: 550 pixel rms < 0.001
status 4: 418 pixel intensity > 16000 in more than 0.1 fraction of events
status 8: 132 pixel intensity < 1 in more than 0.1 fraction of events
status 16: 0 pixel average > 16000
status 32: 918 pixel average < 11444.5
[I] L0108 status 64: 139 pixel with bad gain mode switch for jungfrau only!
```

Status extra

Constants of the type pixel_status are defined in processing of dark runs. They may not cover all possible bad pixel cases. New options for bad pixel status are in progress of development and implementation. Currently we may consider to use light data (e.g. new command det_raw_pixel_status - generates status_data), charge injection for new epix detectors (status_ci), and user defined bad pixel status (status_user), and so on. To deal with more and more emerging bad pixel status_* information we are going to collect them in repository with types status_*.

Method save_constants_in_repository - allows to user save constants in repository with recognizable file name.

Command deploy_constants

- finds in repository all available status_* files associated with detector/panel and requested run time-stamp,
- merge them by bits, preserve merged file under status_merged type,
- deploy this file in calibration database with type status_extra.

Example of files in repository and DB

```
Repository, presumably common for most of detectors. Even can be extended to LCLS-II...:
/reg/g/psdm/detector/calib/constants/
generic path to files:
/reg/g/psdm/detector/calib/constants/<dettype>/<panel-or-detector-id>/<status_type>/<dettype>_<panel-
alias>_<YYYYMMDDHHmmSS-of_run>_<expname>_r<runnum>_<status_type>.txt
example:
/reg/g/psdm/detector/calib/constants/epix100a/3925999616-0996579585-0553648138-1232098304-1221641739-2650251521-
3976200215/status_user/epix100a_0001_20160318191036_xpptut15_r0260_status_user.txt
/reg/g/psdm/detector/calib/constants/epix100a/3925999616-0996579585-0553648138-1232098304-1221641739-2650251521-
3976200215/status_extra/epix100a_0001_20160318191036_xpptut15_r0260_status_extra.txt
merged to:
/reg/g/psdm/detector/calib/constants/epix100a/3925999616-0996579585-0553648138-1232098304-1221641739-2650251521-
3976200215/status_merged/epix100a_0001_20160318191036_xpptut15_r0260_status_merged.txt
deployed to:
/cds/data/psdm/XPP/xpptut15/calib/Epix100a::CalibV1/XcsEndstation.0:Epix100a.1/status_extra/0-end.data
```

Finding Bad Pixel status information

Due to algorithms evolution the bad pixel bit assignment may be different from description above. The best source of true information about bad pixel content and statistics is the log file. Another option is to look at code. Tables of this note contain references to both sources of information for different type of commands/detectors.

LCLS

Detector	Processing command	Log files directory	Log file name	Code
epix10ka	epix10ka_pedestals_calibration	/reg/g/psdm/detector/gains/epix10k/panels/logs/<year>/	<timestamp>_log_epix10ka_pedestals_calibration_<uid>.txt	UtilsEpix10kaCalib.py#L522
jungfrau	jungfrau_dark_proc	/reg/g/psdm/detector/gains/jungfrau/panels/logs/<year>/	<timestamp>_log_jungfrau_dark_proc_<uid>.txt	UtilsCalib.py#L727
other detectors	calibrun det_ndarr_dark_proc	/reg/g/psdm/logs/calibman/<year>/<month>/	<time-stamp>-log-<login>-<job-id>.txt	det_ndarr_raw_proc#L337
other detectors	NEW: det_raw_pixel_status deploy_constants	/reg/g/psdm/detector/calib/constants/<dettype>/logs/<year>/ /reg/g/psdm/detector/calib/constants/<dettype>/logs/<year>/	<time-stamp>_log_det_raw_pixel_status_<uid>.txt <time-stamp>_log_deploy_constants_<uid>.txt	det_raw_pixel_status deploy_constants

Examples

/reg/g/psdm/detector/gains/epix10k/panels/logs/2022/2022-07-05T142149_log_epix10ka_pedestals_calibration_jortiz.txt

```
process panel:15 id:0000000002-0173621761-3221225494-1014046789-0019435010-0000000000-0000000000
[I] L0111 evaluate_limits RMS: ave=3.418 std=0.344  limits low=1.353 high=5.483
[I] L0111 evaluate_limits AVE: ave=3155.633 std=147.439  limits low=2271.000 high=4040.267
[I] L0528 Bad pixel status:
  status  1:       35 pixel rms       > 5.483
  status  2:        0 pixel rms       < 1.353
  status  4:        0 pixel intensity > 16000 in more than 0.1 fraction of events
  status  8:        0 pixel intensity < 1 in more than 0.1 fraction of events
  status 16:       13 pixel average   > 4040.27
  status 32:      354 pixel average   < 2271
```

/reg/g/psdm/detector/gains/epix10k/panels/logs/2022/2022-07-05T142149_log_epix10ka_pedestals_calibration_jortiz.txt

```
raw data found/selected in 999 events
[I] L0688 begin data summary stage
[I] L0111 evaluate_limits RMS: ave=4.783 std=1.303  limits low=0.001 high=12.604
[I] L0111 evaluate_limits AVE: ave=14191.873 std=457.903  limits low=11444.455 high=16000.000
[I] L0733 bad pixel status:
  status  1:      244 pixel rms       > 12.604
  status  2:      550 pixel rms       < 0.001
  status  4:      418 pixel intensity > 16000 in more than 0.1 fraction of events
  status  8:      132 pixel intensity < 1 in more than 0.1 fraction of events
  status 16:        0 pixel average   > 16000
  status 32:      918 pixel average   < 11444.5
[I] L0108
  status 64:      139 pixel with bad gain mode switch
```

/reg/g/psdm/logs/calibman/2022/07/2022-07-05-14:24:06-log-jortiz-131554.txt

```
Raw data for XcsEndstation.0:Epix100a.1 found/selected in 1001 events , begin data summary stage
  evaluate_limits: RMS ave, std = 3.411, 0.276  low, high limits = 1.500, 7.000
  evaluate_limits: AVE ave, std = 4515.065, 203.890  low, high limits = 10.000, 10000.000
  Bad pixel status:
  status  1:      134 pixel rms       > 7.000
  status  8:        5 pixel rms       < 1.500
  status  2:        6 pixel intensity > 10000 in more than 0.1 fraction of events
  status  4:        0 pixel intensity < 10 in more than 0.1 fraction of events
  status 16:        0 pixel average   > 10000
  status 32:        5 pixel average   < 10
```

Command det_raw_pixel_status

Script det_raw_pixel_status is specifically designed to search for bad pixels in dark and light data using different algorithms, named as "features 1-6" after publication [Journal of Applied Crystallography - 2022 - Sadri]. Algorithms of det_raw_pixel_status in for features 1-6 closely follow to this publication. Feature 11 is a new development to search for bad pixels with too low or too high gain factors.

Meaning and application of features in det_raw_pixel_status can be briefly listed as

Feature #	Data type	Description	# of events	status type in repository	Comments
1	dark	mean intensity of frames should be in good range	~1000	status_dark	
2	dark	pixel mean intensity should be in good range	~1000	status_dark	
3	dark	pixel rms intensity should be in good range	~1000	status_dark	
4		gain stage indicators between stages			for gain switching detectors only, TBD
5		gain stage indicators for each gain stage			for gain switching detectors only, TBD
6	light	average SNR of pixels over time	~1000	status_light	
11	light	pedestal subtracted pixel intensity max value for entire run should be in good range	entire run sample >10k	status_max	[Bad pixel mask for epix100a xpplw3319]

where "good range" is equivalent to the "robust statistics" in publication to the bulk of data.

det_raw_pixel_status command examples

```
det_raw_pixel_status
det_raw_pixel_status -h
...
det_raw_pixel_status <dark-data> ... -F 1,2,3  # creates status file of type status_dark
det_raw_pixel_status <light-data> ... -F 1,6  # creates status file of type status_light
det_raw_pixel_status <light-data> ... -F 11 -n 1000000 # creates status file of type status_max

deploy_constants ... # merges all status_* constants from repository to calib directory type status_extra
```

LCLS-II

Detector	Dark processing command	Log file directory	Log file name	Code
epix10ka	epix10ka_pedestals_calibration	/cds/group/psdm/detector/gains2/epix10ka/panels /logs/<year>/	<time-stamp>_log_epix10ka_pedestals_calibration_<uid>txt	UtilsEpix10kaCalib.py#L355
epix100a /opal/etc	det_dark_proc	/cds/group/psdm/detector/calib2/epix100/logs /<year>/	<time-stamp>_log_det_dark_proc_<uid>.txt	UtilsCalib.py#L281

Examples

/cds/group/psdm/detector/calib2/epix100/logs/2022/2022-03-01T071247_log_det_dark_proc_akamalov.txt

```
[I] L0230 ________________________________________________________________________________
raw data found/selected in 1000 events
[I] L0233 begin data summary stage
[I] L0279 bad pixel status:
  status  1:      300 pixel rms       > 6.533
  status  2:        1 pixel rms       < 0.001
  status  4:     1434 pixel intensity > 16000 in more than 0.1 fraction of events
  status  8:    10170 pixel intensity < 1 in more than 0.1 fraction of events
  status 16:       31 pixel average   > 5280.34
  status 32:        2 pixel average   < 2692.9
```

/cds/group/psdm/detector/gains2/epix10ka/panels/logs/2022/2022-04-06T163108_log_epix10ka_pedestals_calibration_xshen.txt

```
[I] L0275 Pre-processing time 16.339 sec
Results for median over pixels intensities:
    0.500 fraction of the event spectrum is below 2901.542 ADU - pedestal estimator
    0.050 fraction of the event spectrum is below 2896.826 ADU - gate low limit
    0.950 fraction of the event spectrum is below 2906.187 ADU - gate upper limit
    event spectrum spread    median(abs(raw-med)): 1.869 ADU - spectral peak width estimator
[I] L0361 Bad pixel status:
  status  1:        6 pixel rms       > 3.898
  status  2:        0 pixel rms       < 0.944
  status  4:        0 pixel intensity > 16000 in more than 0.1 fraction of events
  status  8:        0 pixel intensity < 1 in more than 0.1 fraction of events
  status 16:        2 pixel average   > 3821.78
  status 32:        0 pixel average   < 1986.12
```

References

- Calibration Scripts Repository and Logging
- calibrun - LCLS CLI for dark processing
- Dark processing for LCLS2 area detectors
- Jungfrau and Epix10ka Calibration
- save_constants_in_repository - user's helper to save constants in repository
- det_raw_pixel_status - generates status_data - new algorithms in development
- deploy_constants - deploys constants from repository to DB
- Bad pixel mask - new development
- Paper in JAC: Automatic bad-pixel mask maker for X-ray pixel detectors with application to serial crystallography,
- Journal of Applied Crystallography - 2022 - Sadri - Automatic badpixel mask maker for Xray pixel detectors with.pdf
- Bad pixel mask for epix100a xpplw3319