



NIRSpec Technical Note
NTN-2011-004 / ESA-JWST-TN-18257

Author(s): S. Birkmann
Date of Issue: April 24, 2015
Version: 2.0

Description of the NIRSpec pre-processing pipeline

Abstract:

In this document we describe how to run the NIRSpec pre-processing pipeline, the processing steps performed, and the format of the data written out by it.

1 INTRODUCTION

The NIRSpec focal plane assembly (FPA) consists of two HAWAII-2RG HgCdTe detectors with a cut-off wavelength of $\sim 5.3 \mu\text{m}$. Each detector is paired with a SIDECAR ASIC and read up-the-ramp in multi-accumulation mode (see e.g. Rauscher et al. 2007 for details), making it necessary to pre-process the acquired raw data in order to obtain count-rate images with the lowest possible noise. In addition to the "traditional" readout mode that is shared between all NIR instruments on JWST, the NIRSpec detectors can be operated using the "improved reference sampling and subtraction" (IRS²) readout mode, which needs a more sophisticated treatment of the reference pixels.

2 SCOPE

The scope of this document is to provide an overview of the NIRSpec pre-processing pipeline. The pipeline version number as of writing this document is 0.70 dated 13 April 2015. Section 3 deals with how to install and run the pre-processing pipeline. In Section 4 we describe the default pre-processing steps performed. Finally, in Section 5 we briefly discuss the files that are written out by the pipeline and their format.

3 INSTALLING AND RUNNING THE PIPELINE

The NIRSpec pre-processing pipeline is written in C and is part of the ESA SOT SVN repository. For the remainder of this document we assume that the ESA SOT repository is located

in the "standard" `~/Library/Software/` directory. It has been successfully compiled and run on Mac OS X 10.5 and later, using the GNU C-compiler version `gcc-4.2` or later. It needs to be compiled in 64-bit mode in order to ensure memory allocation for large files. This is taken care of when using the Makefile provided via the SVN repository (see below).

The pipeline utilizes OpenMP to allow the use of multiple threads on multi-core machines in order to improve performance. It also makes use of the GNU Scientific Library (GSL) for some of the FFT transforms and data interpolation tasks. Since it is reading and writing fits files, one also needs to have the `cfitsio` library installed. Details for obtaining and compiling those libraries can be found on the linked websites, but in general both libraries can be installed using the following commands (after unpacking the software into a directory and going into that directory):

```
/> ./configure --prefix=/usr/ {--enable-reentrant}
/> make
/> sudo make install
```

The `--enable-reentrant` flag is only needed for the `cfitsio` library, not GSL. The last step does require superuser privileges in order to copy the compiled libraries into the `/usr/lib/` directory. Installing the libraries in that directory will ensure that they can be picked up by the `gcc` compiler and linker.

Compilation and installation of the pipeline itself is eased by the presence of a Makefile that is part of the SVN repository. Go in the `JWST_C/src/` directory and issue the following commands:

```
/> make pipeline
/> make select_reference_files
```

This will compile the pipeline and the supporting program that is used to pick the right references, and copy the executables into the `JWST_C/bin/` directory, which should be part of your `PATH` environmental variable. If it is not, you can add the following line to your `.cshrc` file, which should reside in your home directory:

```
setenv PATH ${PATH}:~/Library/Software/JWST_C/bin
```

In order to pre-process NIRSpec data, the pipeline needs a number of reference files. The path to these reference files needs to be specified using the `REF_DATA` environmental variable. For example, on the JMAC one would set:

```
/> setenv REF_DATA /Volumes/Pegasus1/Archive/REF_DATA/
```

Once the paths are defined, the pipeline can be called from the command line to pre-process NIRSpec fits files:

```
>/ pipeline [options] <fits file(s)> [more options] [more fits file(s)] [...]
```

where [options] are optional switches to change the default behaviour and <fits file(s)> is the full path and name of the input fits file(s) to be processed. More than one fits file can be processed (sequentially) by listing its path and name [more fits file(s)] and/or by using wildcards (*,?,#). It is also possible to process a set of fits files with different options using the [more options] and [more fits file(s)] fields that can be appended multiple times. By default, the output files will be written into the directory of the input fits files. This behaviour and also the pre-processing steps performed can be changed via switches. The full list of switches is given in Appendix A.

4 DESCRIPTION OF PRE-PROCESSING STEPS

When calling pipeline with one or more fits files as input, these files will be pre-processed and the results saved. The following list summarizes the pre-processing flow including default and optional steps.

1. read in fits header and determine exposure type
2. call `select_reference_file` program to find appropriate reference files
3. read in all needed reference files and initialize data quality map
4. compute weight vectors for optimal weighting
5. check saturation limits and flag saturated data (default)
6. subtract master bias (default) or first group (optional) from data
7. subtract reference pixels (default)
 - a) for IRS² data, perform subtraction of reference output (default) and interleaved reference pixels (default)
 - b) for IRS² data, remove reference output and interleaved reference pixels to arrive at traditional 2k x 2k frame (default)
8. correct the inter-pixel-capacitance effect (optional)
9. perform non-linearity correction (default)
10. subtract the master dark at cube level (optional)
11. compute the count rate using optimum weighting
12. flat field the count rate image (optional)
13. write out the resulting count rate image and cleanup

Steps above without brackets in the description will always be performed, steps with (default) will be performed when pipeline is called without switches stating otherwise, and steps with (optional) will only be performed if the corresponding switch is set. Most of the pre-processing steps will be automatically switched off if the needed reference files for that process cannot be found (e.g. due to a wrongly set REF_DATA environment variable, see Section 3).

In the following sections we will describe the individual pre-processing operations in more detail. A more detailed description of the reference files used by the pipeline can be found in Giardino (2013).

4.1 Saturation detection

This step is performed by default and goes through all the data to check for saturation, i.e. DN values that are not deemed reliable/usable anymore because they exceed the threshold given in the saturation reference files (named `nirspec_satc_nrs[1,2]_?_###.###.fits`, please see Birkmann (2013b) for a description of the reference files). The threshold can be either due to the 16-bit analog to digital converter (A/D) limit (i.e. 65535) or due to physical saturation of the pixel, so that the non-linearity correction (see below) breaks down for larger values.

The group where saturation occurs is recorded for each pixel, and the count rate is determined using the unsaturated data only.

4.2 Bias subtraction

This operation is performed by default. A "master-bias" frame (needed reference files are `nirspec_bias_nrs[1,2]_?_###.###.fits`, please see Birkmann (2013a) for a description of the reference files) is subtracted from every group in the input file, removing any systematic and static (but not time dependent) pixel-to-pixel offsets and large scale gradients. This results in normal pixels having close to zero ADU under dark conditions. As shown below, this is important for reference pixel subtraction and linearity correction, and therefore this step is performed first.

When using window or stripe mode, matching (in start position and size) master-bias frames are used, if available. If not, the corresponding section of the full frame master-bias is subtracted from the data. This latter approach is not optimal, since the time elapsed between the reset and the first read is different for sub-arrays and a different output is used. However, it is still deemed better to subtract a slightly wrong bias frame than not to correct the pixel-to-pixel variations at all.

4.3 Reference pixel subtraction

Reference pixel subtraction is performed to remove i) a time/group dependent offset and ii) some (or most when using IRS² mode) 1/f noise from each individual group. The first correction uses the 4 rows of reference pixels at the top and bottom of each detector frame.

For each of the 4 video outputs (5 for IRS²), and independently for odd/even columns, the sigma clipped mean of reference pixels is subtracted from the corresponding output (pedestal removal). Then, for the traditional readout mode, the four reference columns to the left and right of the frame are used to remove some common 1/f noise from all outputs. This is done by first averaging the eight pixel values per row, giving 2048 values along the column. Finally, an FFT based smoothing is applied to the data and the result is subtracted from every row (see Moseley et al. 2010 for more details on the principle of this approach). This second step needs the reference files `nirspec_fftr_nrs[1,2]_f_###.###.fits`.

For data obtained in IRS² readout mode, the data in the reference output and the interleaved reference pixels in the "regular" four outputs are ordered as a times series. Gaps in these time series (e.g. due to the row overheads) are interpolated using a linear model (for now). Then, that data is fourier transformed and pre-determined filters in frequency space are applied (reference files `nirspec_irs2_nirs[1,2]_i_###.###.fits`). The filtered data is then transformed back and subtracted from the real pixels in the data. Finally, the reference output and the interleaved reference pixels are removed, so that one arrives at the same 2k x 2k frame as for the traditional full frame readout more. For more information on the format of IRS² data and the pre-processing of it, please refer to Rauscher et al. (2012).

When using window (sub-array) mode, only one video output is used. If there are reference pixels inside the window specified, then these are averaged (sigma clipped) for odd/even columns and subtracted, but

4.4 IPC correction

Due to the inter-pixel-capacitance (IPC), charge that is accumulated in one pixel is partly (to the $\sim 1\%$ level) detected in neighbouring pixels (see Sirianni (2014) for more information on IPC). In principle, this effect can be corrected by (de)convolving the 2D image (group by group) with an appropriate (3x3 or 5x5) kernel. The reference files are named `nirspec_ipcc_nrs[1,2]_?_###.###.fits`. By default the pipeline does not perform the IPC correction.

4.5 Non-linearity correction

Up-the-ramp data under stable illumination is not completely linear, because the integrating capacitance of each individual pixel is not constant, but changes (increases) with accumulated charge. Ramps are linearized by applying a fourth order polynomial to the data, with individual coefficients for each pixel (reference files `nirspec_linc_nrs[1,2]_?_###.###.fits`, please see Birkmann (2013b) for a description of the reference files). The details on how the reference file is created will be described in a separate document, but the principle follows the recipe given by Roberto (2011).

For a proper linearity correction, both bias and reference pixel subtraction must have been performed before, and the pipeline will switch off linearity correction if these steps have not been executed.

4.6 Dark current subtraction

When running pipeline from the command line, by default there will be no dark current subtraction. However, if run from within SQUAT, pipeline will subtract the dark from the science exposure at cube level using the `nirspec_drkc_nrs[1,2]_?_###.###.fits` reference files (please see Birkmann (2013a) for a description of those files). The dark cube will be read in and processed if necessary (it already is bias and reference pixel subtracted), i.e. IPC correction (if switched on) and linearity correction are performed prior to the group by group subtraction.

The group by group subtraction improves the subtraction of hot pixels that suffer from field-enhanced emission and thus non-constant dark currents. It also provides a fully processed data cube that can be written out using the `--output-cube` flag.

As the group by group reference dark depends on the frame time, it is different for traditional readout mode and IRS², and different sets of reference files are available for those two modes, as well as a number of reference files for the standard subarrays.

4.7 Slope and uncertainty estimation

In principle, once the up-the-ramp data is linearized, one could fit a straight line to the data and determine the slope. However, using a linear least squared fit with uniform weights is not optimal for high signal-to-noise data. Therefore, we adopted the algorithm described by Fixsen et al. (2000), which makes use of pre-calculated weighting vectors for 15 signal-to-noise bins for the given number of groups. In the low SN limit the weights are uniform (damping the read noise as much as possible), while in the high SN limit the weights are zero everywhere except for the two endpoints of the ramp (maximizing the effective integration time).

The uncertainty of the slope is calculated using the corresponding formulas from Ferruit (2008) and Roberto (2009, 2010). For the uniform weighted case we use formula (7) from Roberto (2010), while for the non-uniform case we use the more general formula (20) from Roberto (2009), adding the readout noise and bitshift/quantization error terms. It has to be noted that the latter formula had two typos that were corrected in the pipeline code. Both formulas are given in Appendix B.

This step also uses the gain (named `nirspec_gain_nrs[1,2]_?_###.###.fits`) and read noise reference files (named `nirspec_rdns_nrs[1,2]_f_###.###.fits`) to go from DN to e⁻ for the correct determination of the uncertainties (the formulas use e⁻). Please see Sirianni & Birkmann (2013) for a more detailed description of those reference files. The resulting count rate (and uncertainties) will however be reported in DN/s unless selected differently by using the `--electrons` switch (see Appendix A below).

4.8 P-flat correction

In principle, the calculated count rate image can be divided by a p-flat to correct for pixel-to-pixel sensitivity variations (reference file name `nirspec_flat_nrs[1,2]_?_###.###.fits`).

Table 1: Description of the multi-extension output fits file.

No	Name	BITPIX	Description
1	SCI	-32	Count rate image of the science data (in ADU s ⁻¹ by default)
2	ERR	-32	Uncertainty of the science data (in ADU s ⁻¹ by default)
3	DQ	16	Data quality flag array
4	SAMP	16	Array containing the number of groups before hitting saturation
5	CHI	-32	Root mean square of the linear fit to the up-the-ramp data

However, since NIRSpec is a spectrograph in which each pixel responds to different wavelengths, this correction is not performed by the pre-processing pipeline by default, and will rather be part of the extraction pipeline NIPS.

5 PIPELINE OUTPUT FILES

By default, the pipeline will output one fits file per input fits file into the directory of the input file. A different output directory can be chosen using the `--output-dir <dir>` switch (see also Appendix A). The name of the output file will have the same root name, but ending in `.cts.fits` instead of `.fits`. For example, for input file name

```
NRS DRY-RUN-ARGON-01_1_491_SE_2011-01-25T19h19m33.fits
```

The output file name will be

```
NRS DRY-RUN-ARGON-01_1_491_SE_2011-01-25T19h19m33.cts.fits
```

The output file contains an empty (apart from the header) primary array and five image extensions holding the output data: the count rate image, the uncertainty of the count rate, the data quality flags, the number of good samples (groups before hitting saturation), and the deviation from a straight line fit (root mean square). The names and contents of all extensions are summarized in Table 1. The flags for the data quality array can be found in the header of the corresponding extension of the output fits file, and also as a table in Giardino et al. (2011).

The primary header contains all the keywords from the original raw data file (where some, like date and origin, might have been updated), plus some additional keywords related to the pre-processing pipeline. All latter keywords start with PIPE and contain information on the pre-processing steps performed and the path to and name of the reference files used.

6 REFERENCES

Birkmann, S. 2013a, Description of the NIRSpec bias and dark reference files, NIRSpec Technical Note NTN-2013-003, ESA/ESTEC

- Birkmann, S. 2013b, Description of the NIRSpec linearity correction and saturation check reference files, NIRSpec Technical Note NTN-2013-004, ESA/ESTEC
- Ferruit, P. 2008, Computation of noise levels for different readout patterns, Technical Note NIRS-CRAL-TN-0024, CRAL
- Fixsen, D. J., Offenbergl, J. D., Hanisch, R. J., et al. 2000, PASP, 112, 1350
- Giardino, G. 2013, NIRSpec reference data product specification, NIRSpec Technical Note NTN-2013-007, ESA/ESTEC
- Giardino, G., Sirianni, M., & Birkmann, S. 2011, NIRSpec reference files and quality flags, NIRSpec Technical Note NTN-2011-005, ESA/ESTEC
- Moseley, S. H., Arendt, R. G., Fixsen, D. J., et al. 2010, in Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, Vol. 7742
- Rauscher, B. J., Alexander, D., Brambora, C. K., et al. 2007, in Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, Vol. 6690
- Rauscher, B. J., Arendt, R. G., Fixsen, D. J., et al. 2012, in Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, Vol. 8453, Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, 1
- Roberto, M. 2009, Derivation of the correct noise equation for general MULTIACCUM readout, Tech. Rep. JWST-STScI-001853, SM-12, STScI
- Roberto, M. 2010, Two Fundamental Equations for IR Ramp Fitting, Tech. Rep. JWST-STScI-002161, SM-12, STScI
- Roberto, M. 2011, Implementation of a new algorithm for the correction of non-linearity in JWST detectors, Tech. Rep. JWST-STScI-002344, SM-12, STScI
- Sirianni, M. 2014, Interpixel capacitance in NIRSpec FPA104 from hot pixels, NIRSpec Technical Note NTN-2014-009, ESA/ESTEC
- Sirianni, M. & Birkmann, S. 2013, Description of the NIRSpec amplifier and gain reference files, NIRSpec Technical Note NTN-2013-005, ESA/ESTEC

A LIST OF PIPELINE SWITCHES

The following options are available:

- h OR --help
Print this help. Program will terminate without further processing.
- version
Print version information.
- t <number of threads> OR --threads <number of threads>
Specify the maximum number of threads the program should ever use.
Default is $\max(2, \text{numberofcores} / 2)$.
- b OR --bias
Subtract master bias from each frame (default).
- nb OR --nobias
Do not subtract master bias from each frame.
- ffb OR --firstframebias
Subtract first frame from all data before reference pixel subtraction
and add back afterwards (STScI mode)
- abb OR --addbackbias
Add back bias frame after reference pixel subtraction. For testing only.
Default is off
- zf OR --zeroframe
Assume reset data is good and use zeroth read when fitting slopes. Default
is off, for testing only.
- r OR --refpix
Perform reference pixel subtraction (default).
- nr OR --norefpix
Do not perform reference pixel subtraction.
- refcol
Use fft smoothed reference columns during reference pixel subtraction
(default).
- norefcol
Do not use reference columns during reference pixel subtraction.
- forceref [nrows]
Force reference pixel correction on, even if no real reference pixels
are present. Optional nrows parameter gives the number of reference rows
used (top and bottom), if not given 4 will be used.

- irs2-alpha-only
Use alpha filter only for IRS2 correction, i.e. only reference output data will be used.
- irs2-beta-only
Use beta filter only for IRS2 correction, i.e. only the interleaved reference pixels in the outputs will be used.
- i OR --ipc
Perform pixel intra-capacitance correction.
- ni OR --noipc
Do not perform pixel intra-capacitance correction (default).

- s OR --saturation
Perform saturation check (default).
- ns OR --nosaturation
Do not perform saturation check.

- l OR --linearity
Perform linearity correction (default).
- nl OR --nonlinearity
Do not perform linearity correction.

- d OR --dark
Perform dark subtraction on cube level (default is off). The dark cube will be processed exactly the same way as the input data before subtraction. Subtraction occurs before computing the count rate.
- D OR --darkcube
Same as options -d and --dark above.

- c [threshold] OR --cosmic [threshold]
Perform cosmic ray detection (not rejection yet) on the up the ramp data (default is off), using the optional threshold value (4.5 times the expected noise by default).

- u OR --uniform
Use uniform (and not optimal) weighting when estimating count rate. Default is off.

- e OR --electrons
Do output in electron (rates) instead of ADU (rates). Uses the GAIN map reference file if available. Default is off.

- f OR --flat
Perform p-flat fielding on countrate image (default is off).

- p <x_index,y_index> OR --pixel <x_index,y_index>
Write out the ramp for the given pixel (indexes start at 0). By default no ramp is written out.
- np OR --nopixel
Do not write out ramp (default).
- pulsed
Switch on pulsed mode. Here, the output will be simply last minus first group. Uncertainties are calculated accordingly. Default is off. If switched on, the --cosmic and --uniform switches will have no effect.
- reffiles <file>
If set, the pipeline will look for the file given by <file> to read in the reference file names and will not create the list of files itself (which is the default). Useful for testing purposes.
- no-split
For multi-integration exposures the individual integrations are split and written out by default. This flag will switch that feature off, each integration will still be processed and a count rate computed.
- output-cube
Write out the full data cube after reference pixel subtraction and linearization. Not written out by default.
- compress
Turn on compression for output fits files (experimental feature). All 16-bit integer data (e.g. splitted integrations of multi-integration exposures and masks) will be compressed lossless. Float images will be written with a lossy RICE compression with a quantization factor that is typically much smaller than the noise in the data.
- max-group <groups>
Only process the first <groups> in the exposure (per integration). The default is to process up to 400 groups.
- output-dir <output directory>
Puts the processed data into the specified output directory. Default is to put it into the same directory as the input data.

B USED FORMULAS FOR SLOPE ERROR ESTIMATION

For up-the-ramp data with uniform weights the variance off the slope b is:

$$Var[b] = \frac{6}{5} \frac{(n^2 + 1)}{n(n^2 - 1)} \frac{b}{t_g} \left[1 - \frac{5}{3} \frac{m^2 - 1}{m(n^2 + 1)} \frac{t_f}{t_g} \right] + 12 \frac{1}{n(n^2 - 1)} \frac{1}{t_g^2} \left[\frac{\sigma_{ron}^2}{m} + \frac{g^2 m^2}{12} \right], \quad (1)$$

where n is the number of groups, m is the number of (averaged) frames per group, t_g is the time between groups, t_f is the time between frames, σ_{ron} is the read noise, and g is the gain. For up-the-ramp data with optimal (non-uniform) weights the variance of the slope b is:

$$Var[b] = \sum_{i=1}^n a_i^2 \left[b(i-1)t_g + \frac{m+1}{2m} b t_f + 2 \frac{1}{m^2} \sum_{k=2}^m \sum_{l=1}^{k-1} l b t_f + \frac{\sigma_{ron}^2}{m} + \frac{g^2 m^2}{12} \right] + 2 \sum_{i=2}^n \sum_{j=1}^{i-1} a_i a_j \left[b(j-1)t_g + \frac{m+1}{2} b t_f \right], \quad (2)$$

where a_i and a_j are the weights of the i th and j th sample, respectively.