

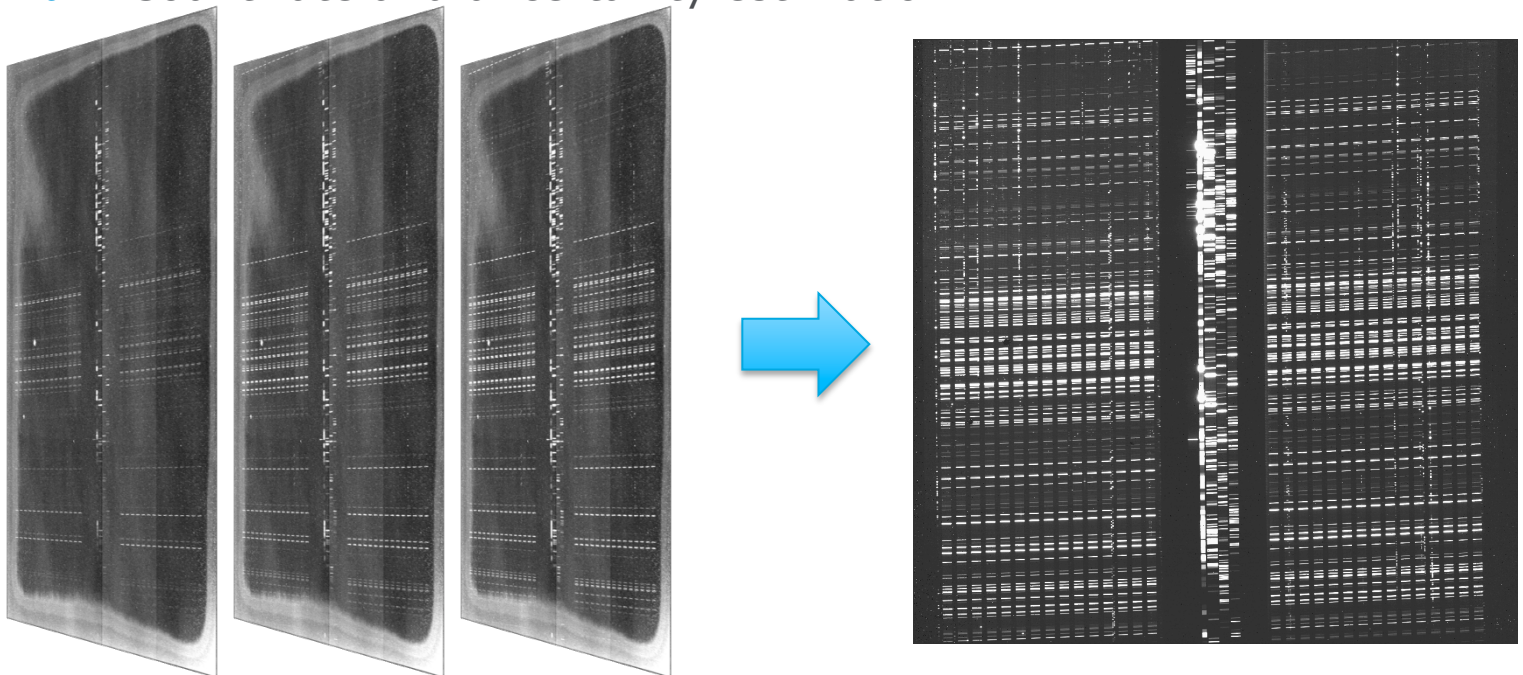
NIRSpec preprocessing pipeline

From MULTIACCUM sampled data to count rate images

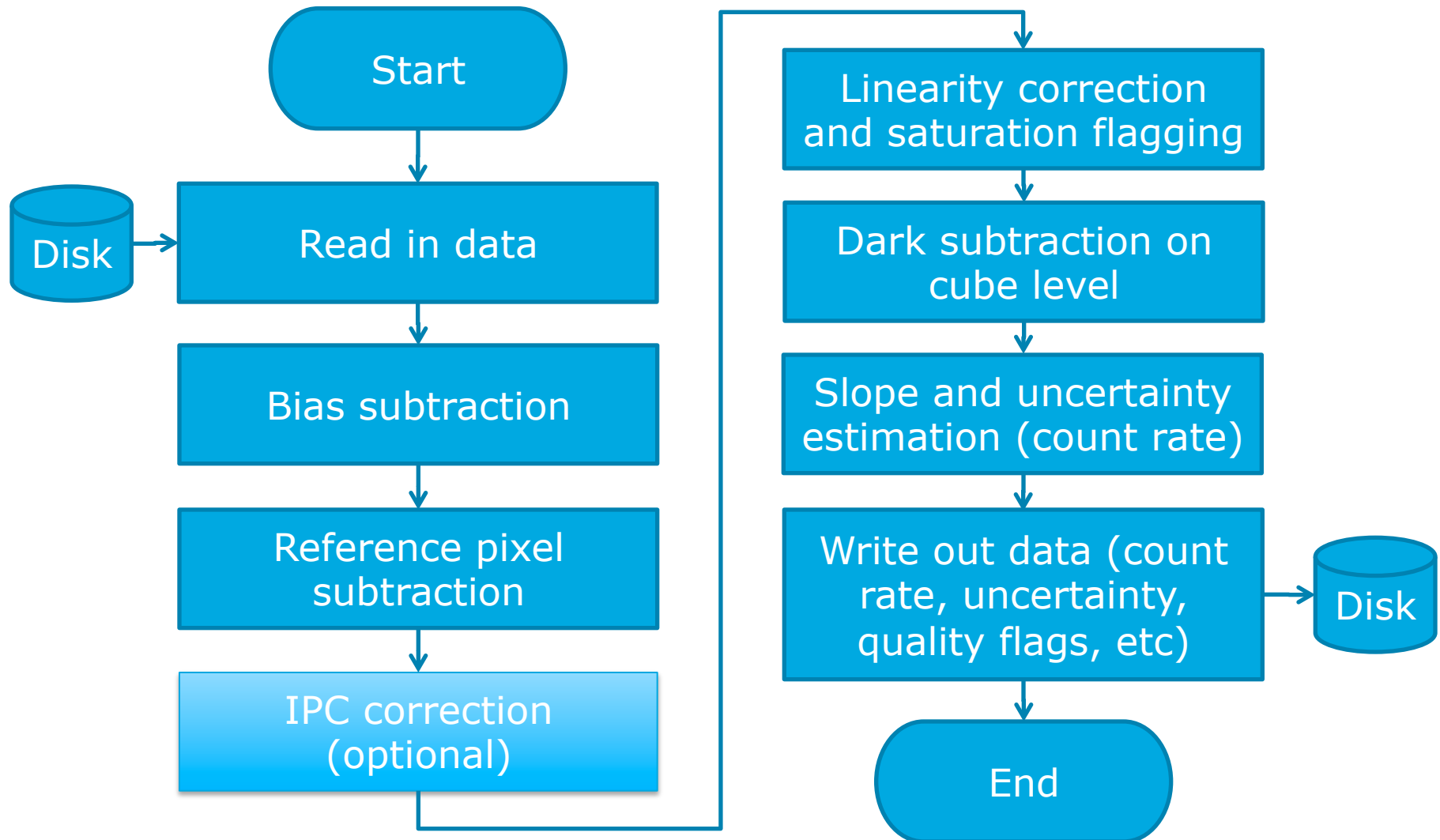
ELIXIR School III – ESA/ESTEC – 26./27. September 2012

Stephan Birkmann – Instrument Calibration Scientist

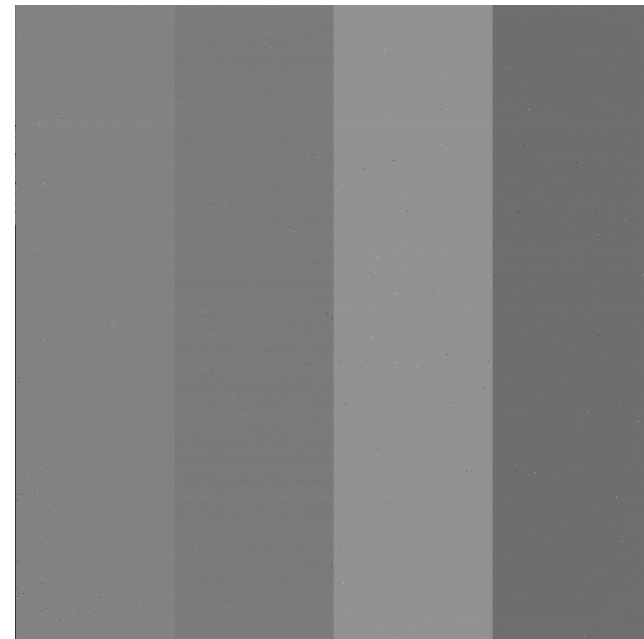
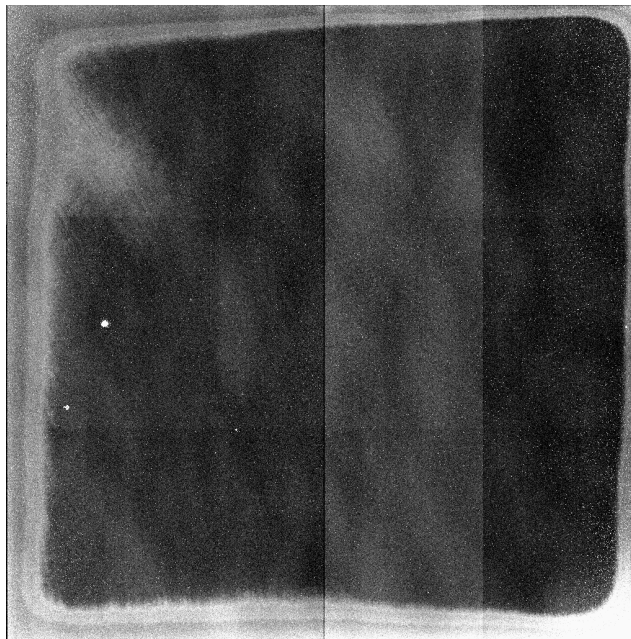
1. Overview of pipeline pre-processing steps
2. Details on main algorithms
 - a. Bias subtraction
 - b. Reference pixel subtraction
 - c. Linearity correction and saturation detection/flagging
 - d. Count rate and uncertainty estimation



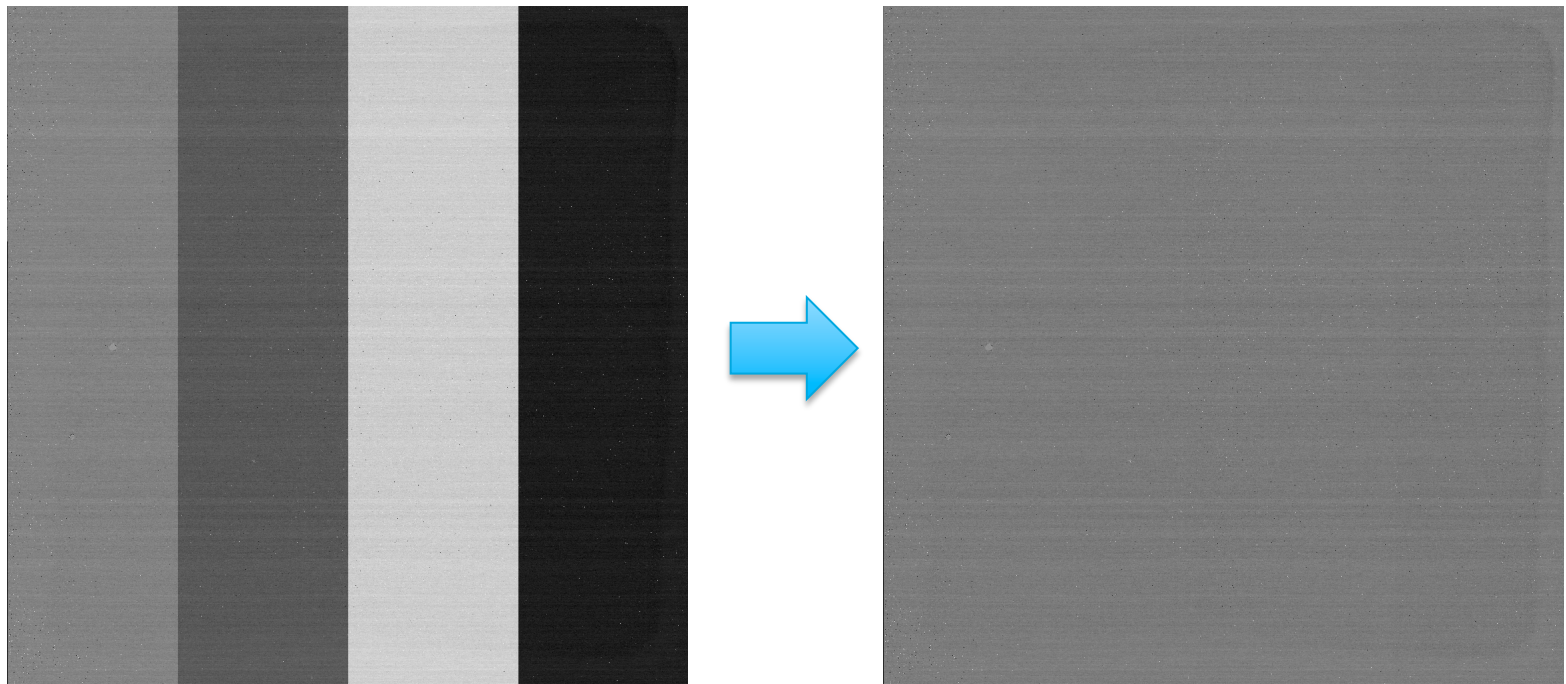
Overview of pre-processing steps



1. Uses “Masterbias” created from first frame of dark exposure
2. Subtracted from all frames in science exposure
3. Goal: remove pixel-to-pixel variation in offset
 - a. Proper outlier removal during reference pixel subtraction
 - b. Close to zero offset for all pixels at beginning of integration
 - c. Important for linearity correction

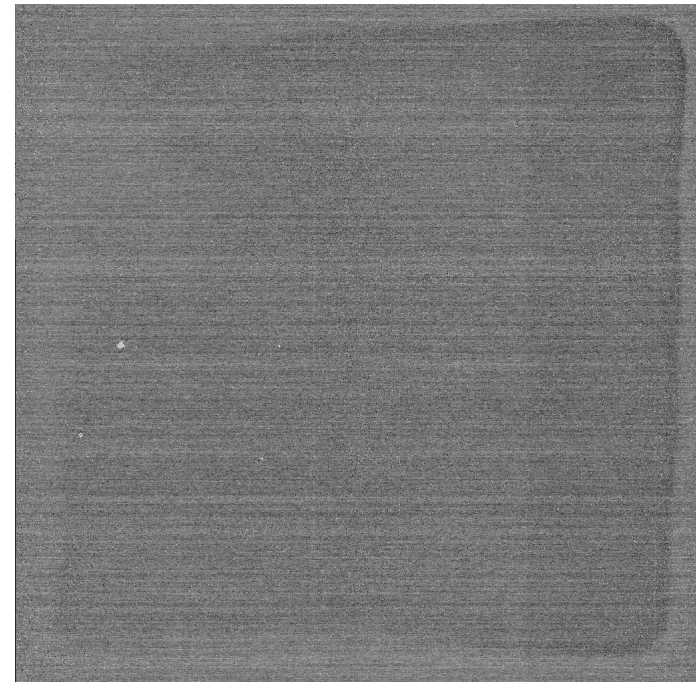
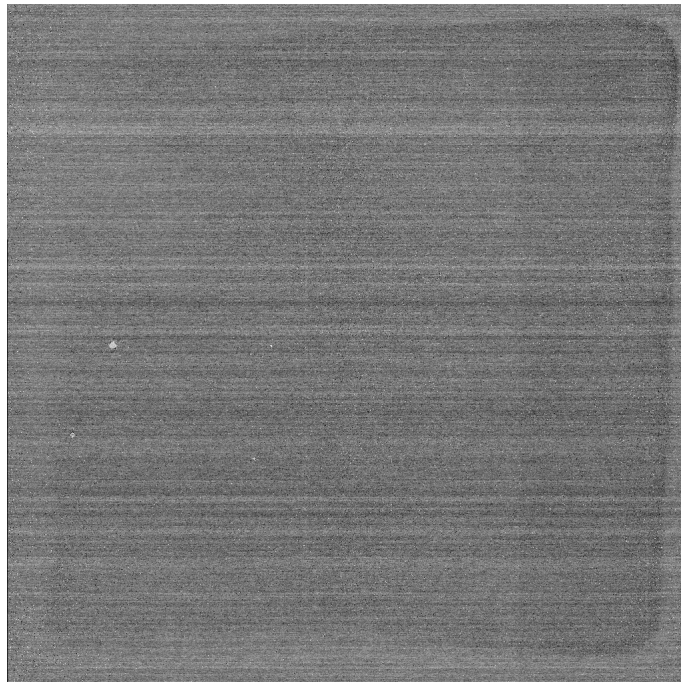


1. Subtract average values (sigma clipped) of 4 top/bottom rows, effectively removing pedestal, separately for
 - a. Each frame/group
 - b. Each output (4 in total)
 - c. Odd and even columns

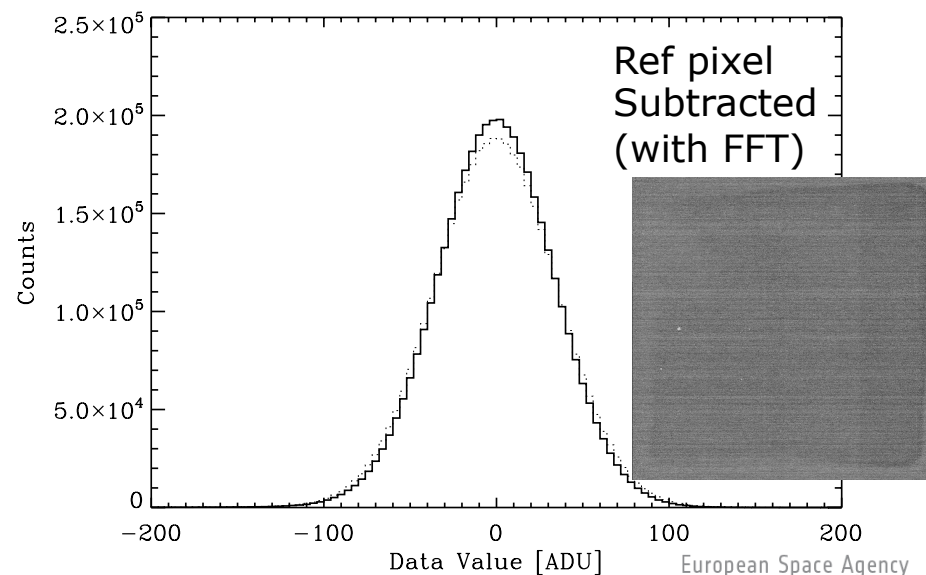
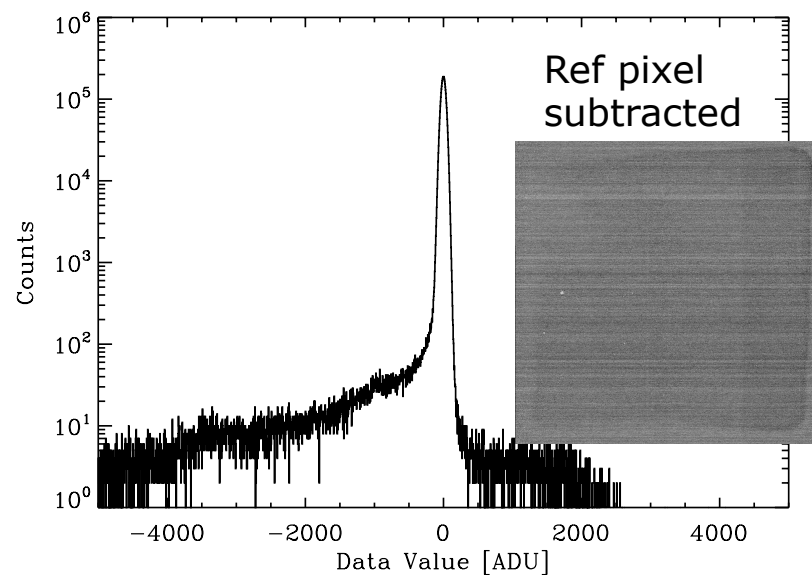
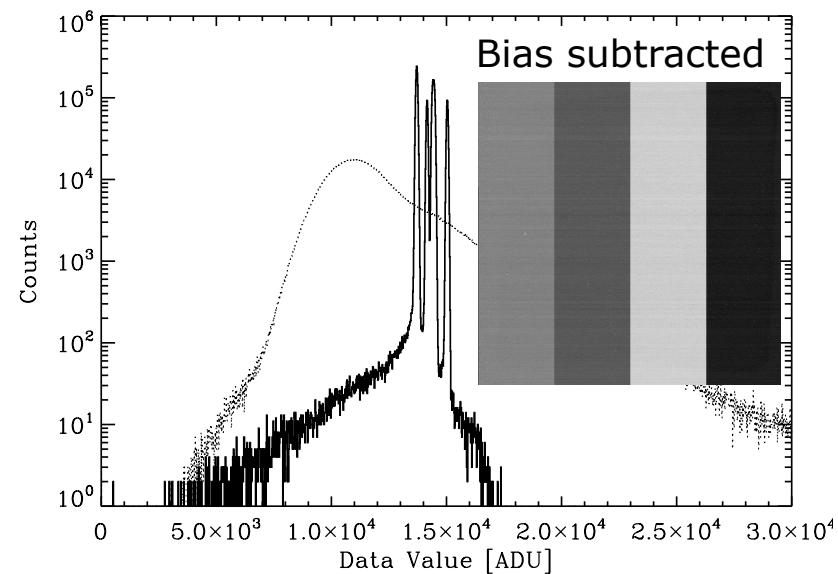
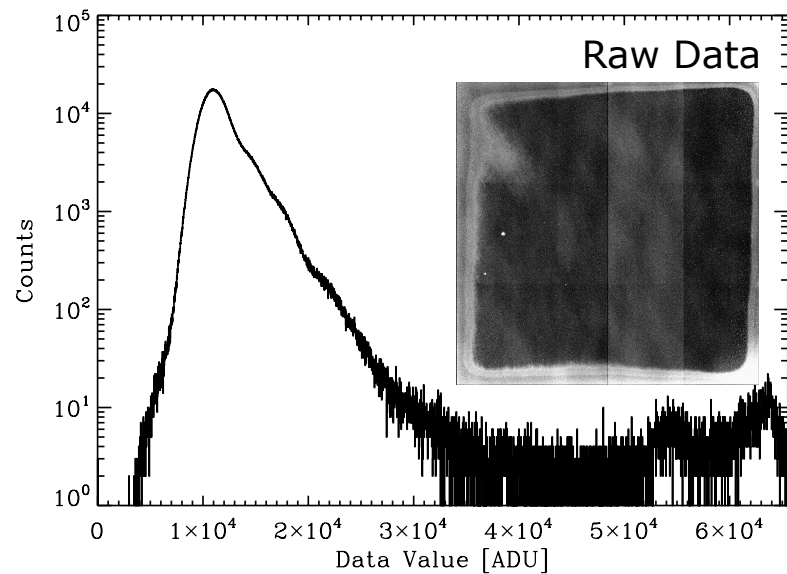


Reference Pixel Subtraction (2)

1. Use the 8 reference columns (4 left/right each) to remove $1/f$ noise
 - a. Code adapted from B. Rauscher
 - b. Averages the 8 values per row (clipped)
 - c. The resulting 2048 column values are smoothed using FFTs and phase information

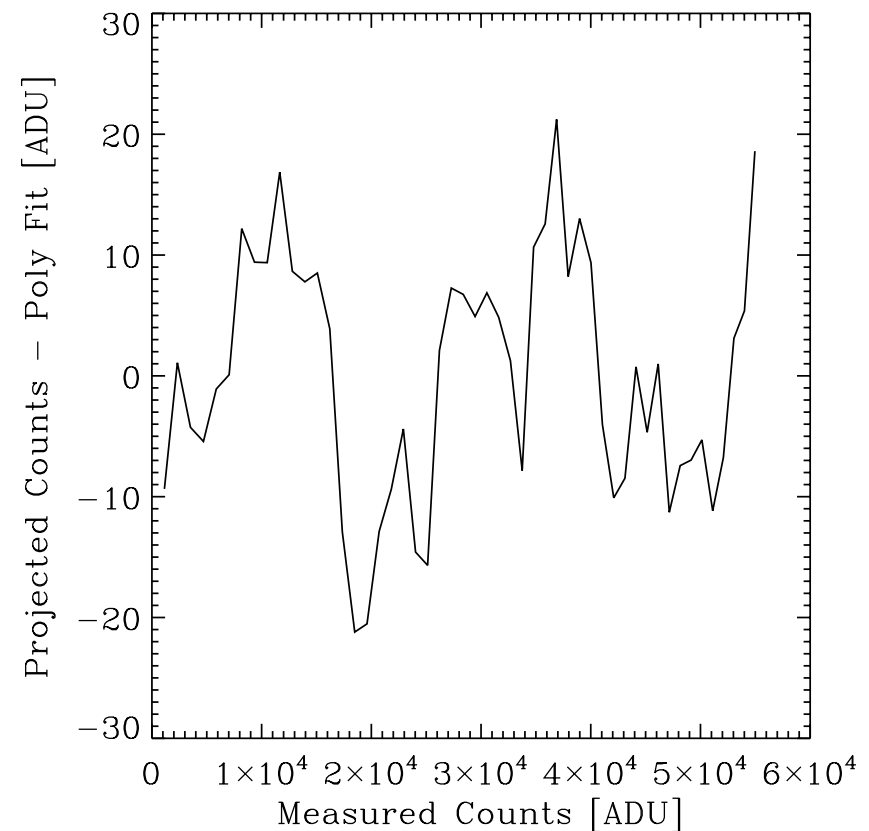
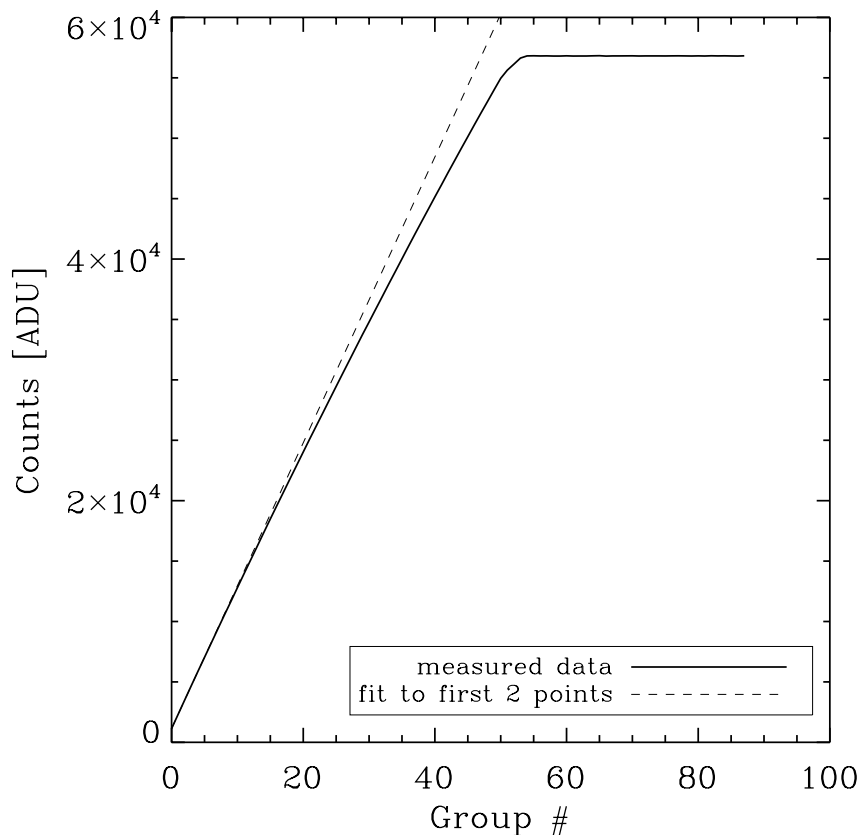


Histograms

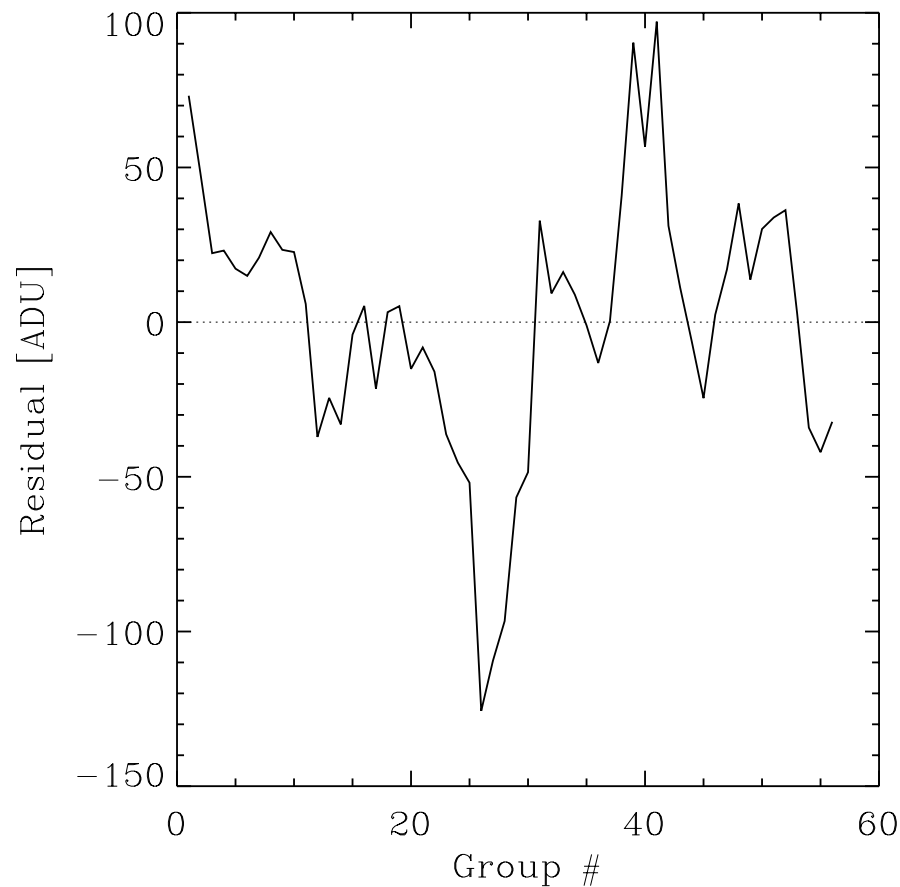
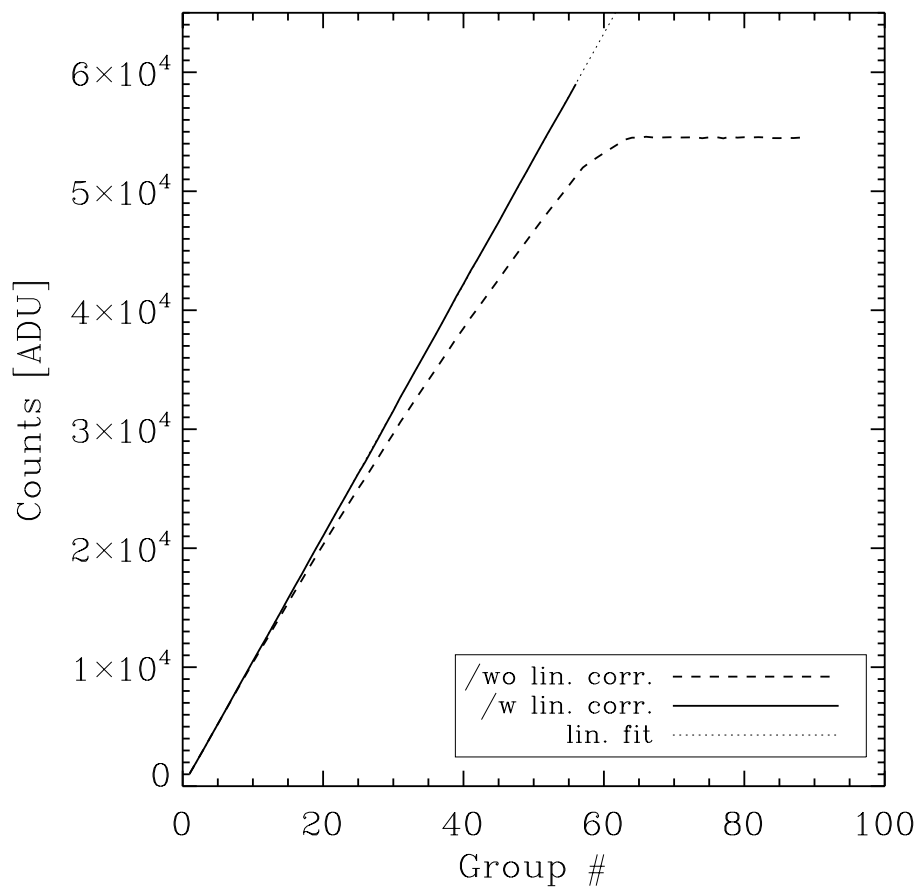


Linearity Correction

1. Ramps are not linear due to charge dependent integration capacitance
 - a. Corrected by 4th order polynomial (for each pixel)
 - b. Coefficients found by fitting “ideal” vs. measured counts

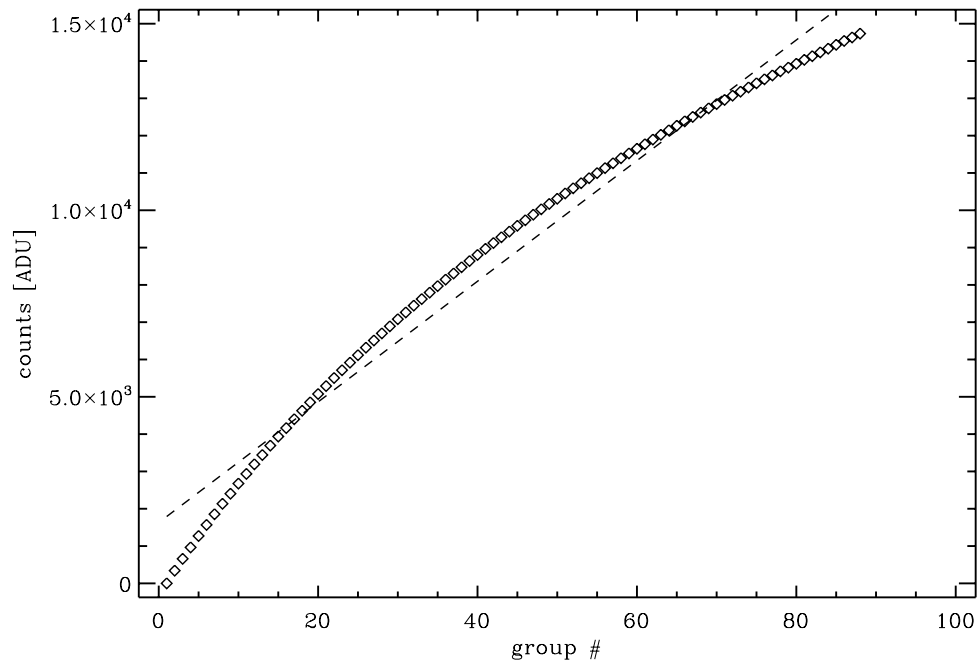


Linearity Correction - Applied



Linearity correction also takes care of saturated data

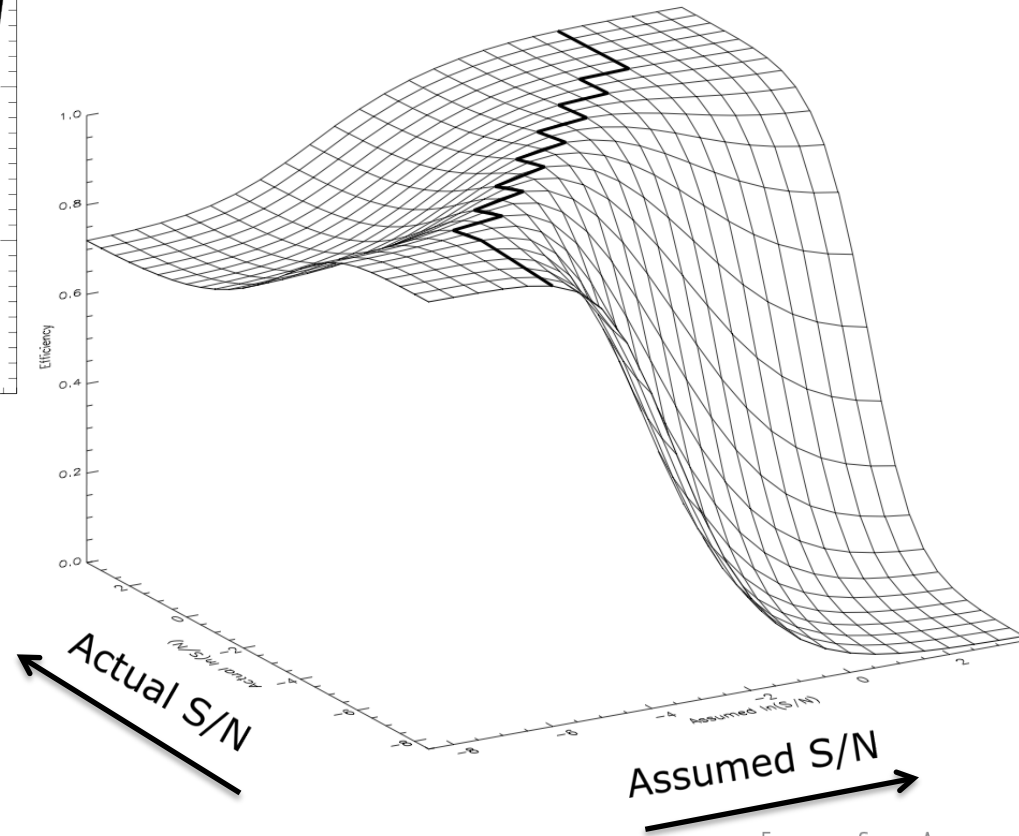
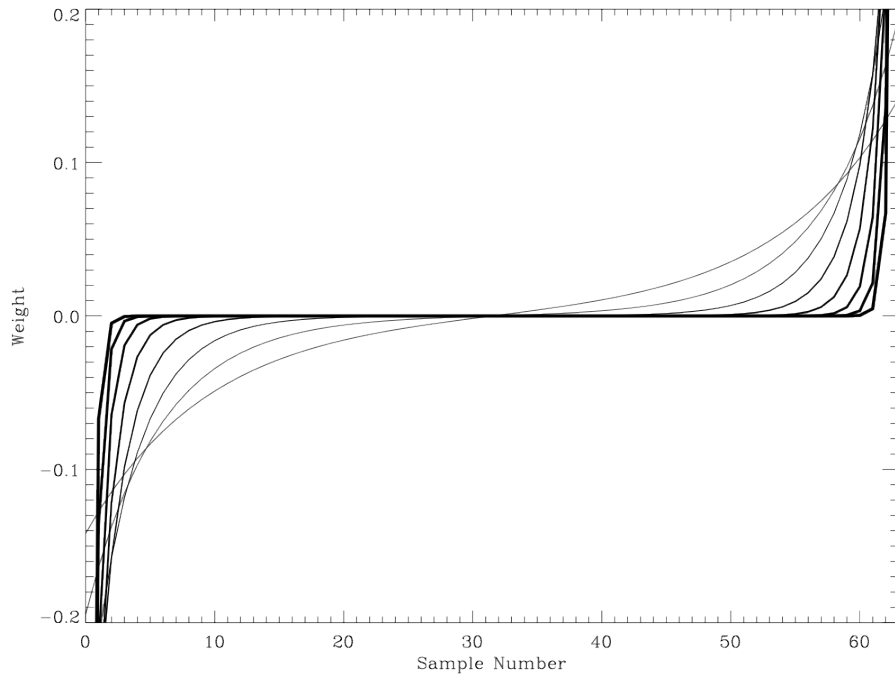
1. Dark subtraction is done on preprocessed cube level using dark reference files
 - a. Even after linearity correction many hot pixels show non-linear behavior due to field enhanced emission effect
 - b. Result is non constant slope and thus count rate
 - c. Subtracting on cube level helps mitigating this issue



1. 4th order polynomial is not the physical correct description (but it works)
2. Needs special data set to be established
 - a. Slowly going into saturation/ADC limit with all pixels
 - b. Very stable illumination
3. Will depend on detector tuning/temperature/gain
4. Needs data to start from right offset
 - a. Bias subtraction needed
 - b. Reference pixel subtraction needed
 - c. Both are issues in window mode

1. Use optimum weighting according to Fixsen et al. 2000 (PASP 112:1350-1359)
 - a. Principle:
 - low signal → uniform weights (read-noise limit)
 - Higher signal → endpoints more important (shot-noise limit)
 - b. Involves inverting the sum of the difference and read-noise covariance matrices
 - c. Creates a weighting vector to be directly applied to the data
 - d. Depends on $\text{signal}/\sigma_{\text{read}}^2$ and number of groups
 - e. Set of weight vectors pre-calculated during run-time for 15 S/N bins and all possible number of groups (\leq groups in exp)
 - f. Signal is first estimated with uniform weights
 - Result is used to select the correct S/N bin
 - g. Signal is recomputed with optimum weights

Slope Estimation - Weights



Fixsen et al. 2000 (PASP 112:1350-1359)

1. Using formula derived by Pierre Ferruit
 - a. Needs exposure parameters (m , n , t_g , t_f , see below)
 - b. Read noise σ_{read} (e^-)
 - c. Gain (e^-/ADU)
 - d. Signal f (e^-/s , from countrate image and gain)

$$\sigma_{\text{total}}^2 = \frac{12 (n - 1)}{m n (n + 1)} \sigma_{\text{read}}^2 + \frac{6 (n^2 + 1)}{5 n (n + 1)} (n - 1) t_g f - \frac{2 (m^2 - 1) (n - 1)}{m n (n + 1)} t_f f.$$

m : number of frames per group

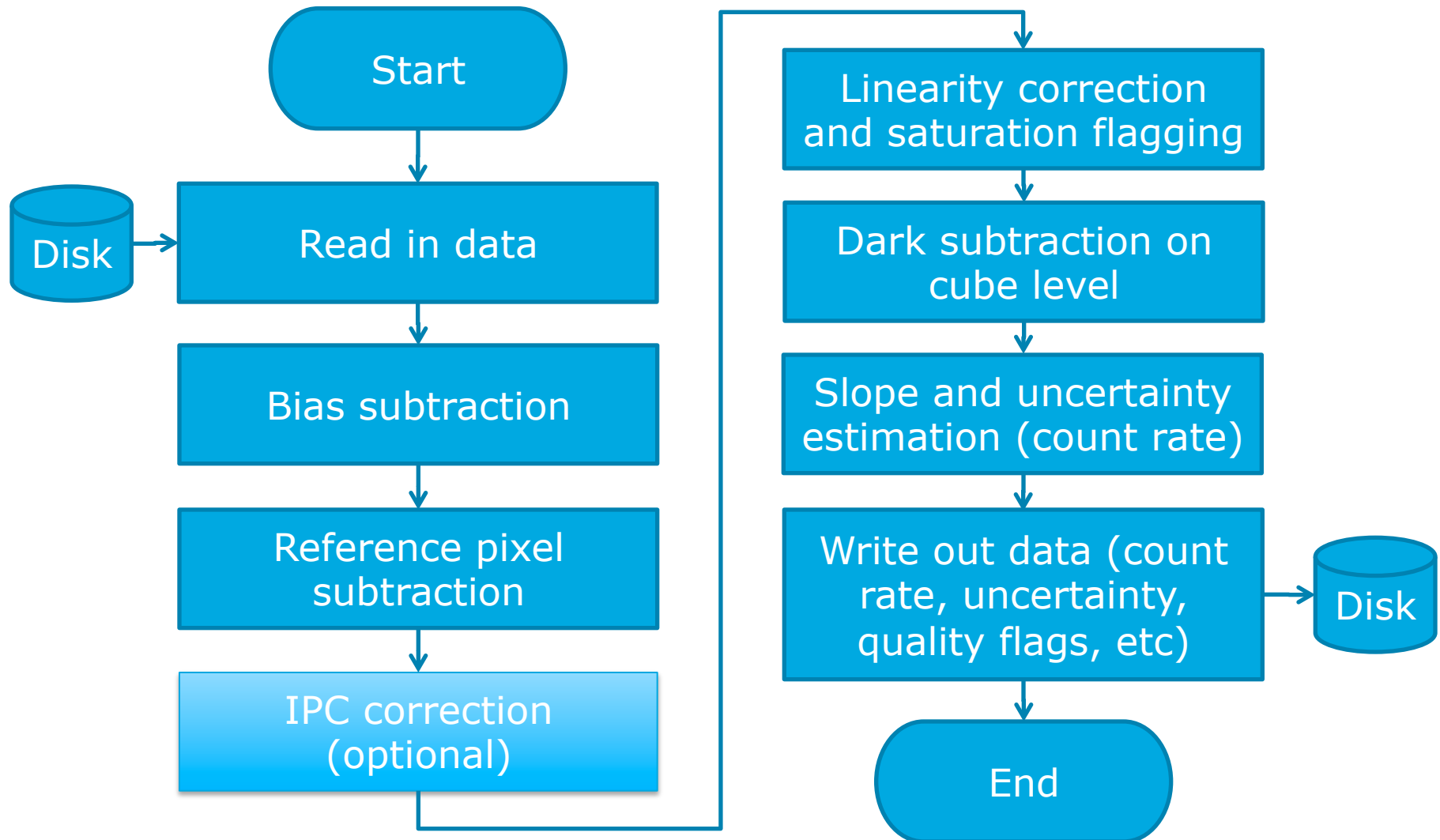
t_g : group time

n : number of groups per integration

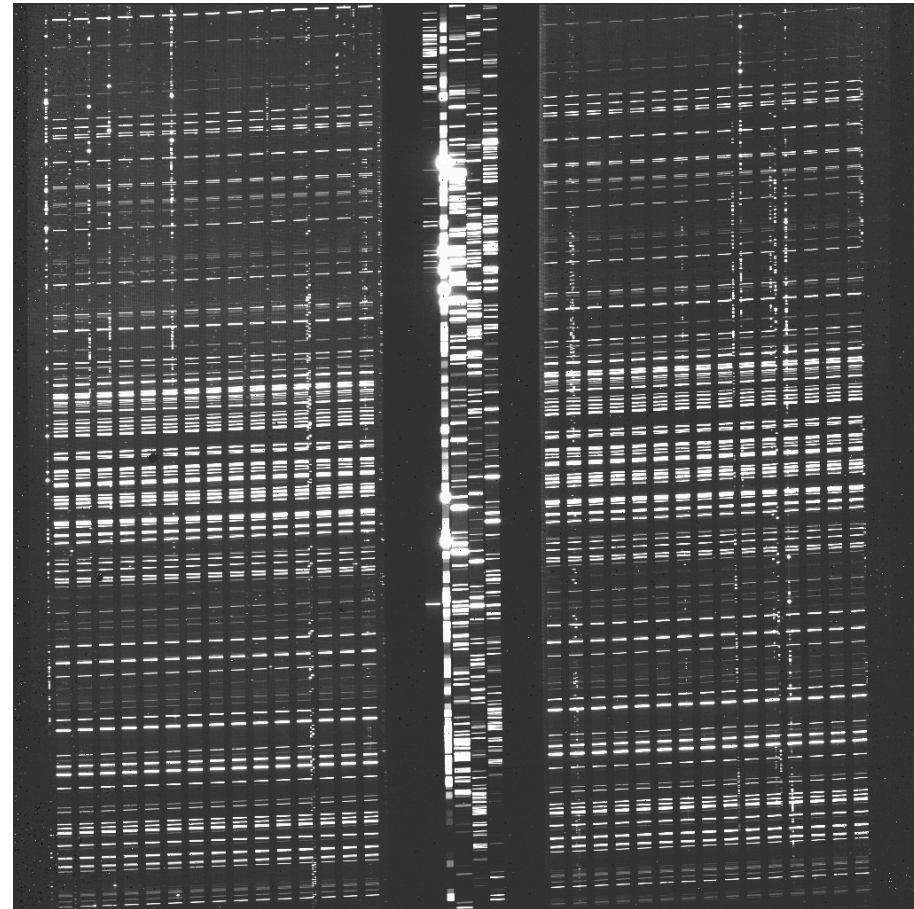
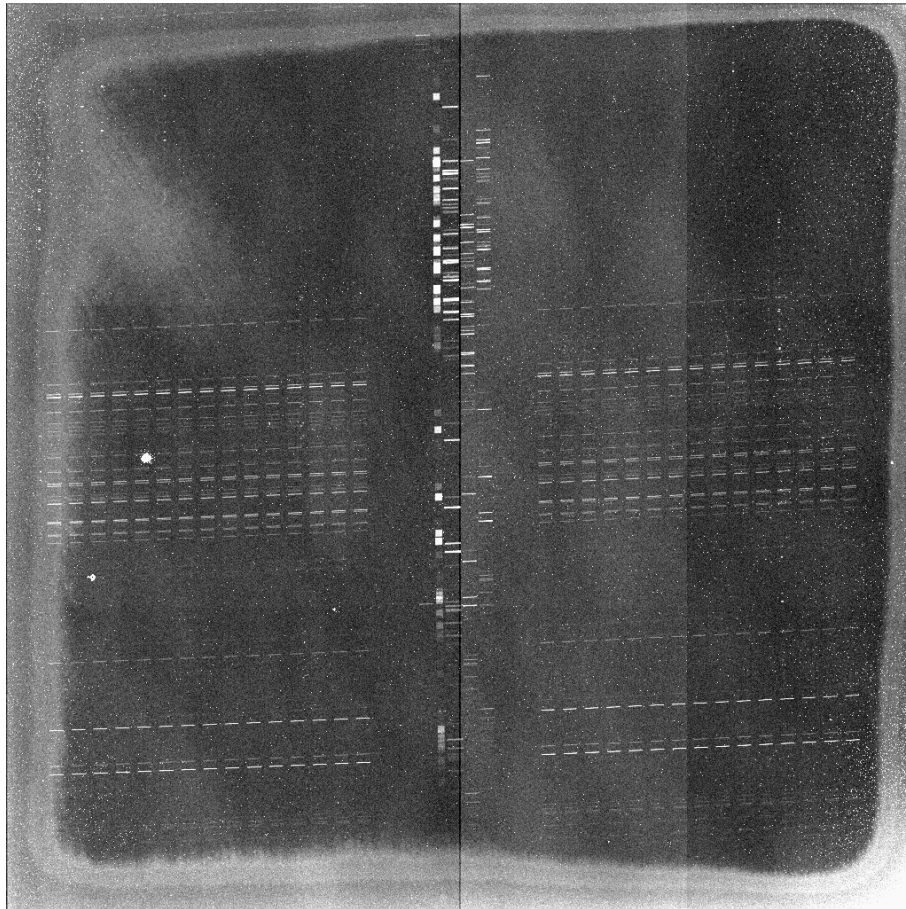
t_f : frame time

2. Only correct for uniform weighted data (low signal, detector limited case), overestimates uncertainty for high signals (by up to 23%)
 - a. Estimate using the actual weights (per pixel) is available in the pipeline (more complex formula with sums over weights)

Overview of pre-processing steps



The End / Result



1. When called within SQUAT, the pre-processing pipeline will do
 - a. Bias subtraction
 - b. Reference pixel subtraction (with FFT columns)
 - c. Linearity correction and saturation detection
 - d. Slope (count rate) estimation using optimal weighting
 - e. Write out data (pipeline):
 - Count rate image (ADU/s)
 - Uncertainty image (ADU/s)
 - Goodness of “fit” image (X^2 in ADU)
 - Output mask with data quality bits
 - Up-the-ramp data for one pixel per SCA in plain text
 - f. Write out summary and some statistics (SQUAT):
 - Exposure summary sheet (PDF)
 - File holding IDL structure on statistics (ask Marco)

2. Pipeline can also be called from the command line and is configurable by switches
 - a. Typical call is “pipeline [filename]”
 - b. Invoke “pipeline -h” to see full set of switches

```
[smac013027:Software/JWST_C/src] sbirkman% pipeline -h
NAME
    pipeline -- pre-process NIRSpec detector data

SYNOPSIS
    pipeline [options] <fitsfile(s) ...> [[options] [fitsfile(s)] ...

DESCRIPTION
    Pipeline pre-processes NIRSpec detector data. The basic steps are i) read
    in the data cube, ii) subtract the reference pixels (optional), iii) sub-
    tract a bias frame, flag saturated data, and perform linearity correction
    (optional), iv) compute the slope of the ramp in counts per second, and v)
    write the output data. Output data is written in the same directory as the
    input files and can include the count rate image, an uncertainty image, a
    chi squared image, and the up the ramp data for one selected pixel.

    Pipeline accepts multiple files as input and the use of wildcards. It also
    takes a number of options (see below) that can be different for each input
    file, given that the options precede the file (input is parsed in the
    order given).

    Pipeline makes use of multiple processors/cores if available, leaving at
    least one core free by default on multi-core systems.

    The following options are available:

    -h OR --help
        Print this help. Program will terminate without further processing.
```

1. Cosmic ray rejection on up-the-ramp data
2. Better support for window mode exposures
 - a. Needs fix in microcode and fits writer
 - First two columns per output show lower bias
 - Fits header is wrong about reference pixels
 - b. What to do when no reference pixels are present?

1. We have the possibility to create $n/4 \times 4$ multiaccum data from the recorded $n \times 1$ data in software as long as $n/4$ is an integer
 - a. Simulate in orbit data
 - b. Will need own set of Masterbias and linearity correction files

Backup - IPC Correction

1. Charge accumulated in one pixel is partly detected in adjacent pixels
 - a. Effects good pixels next to “hot” pixels the most
 - b. Can be corrected by convolution with the “right” 3x3 kernel

