# User Requirements for the new Controller and Data Acquisition System for NOTCam

*Anlaug Amanda Djupvik, Graham Cox, Thomas Augusteijn, Jacob Clasen*

June 27, 2012

## 1 Introduction

### 1.1 Aim

The aim of this document is to supplement the *Detector Controller User Requirements* (Augusteijn & Cox 2007) with requirements specifically related to infrared arrays, in general, and NOTCam, in particular. The new infrared array controller offers new possibilities for data acquisition due to improved speed, and it was realized that some added functionality needed to be specified. In addition, defining particular read modes and data acquisition methods makes it possible to minimize known problems with these IR arrays, such as non-linearity and the so-called *Reset Anomaly*. Due to the many differences with respect to CCDs we realized that a rather detailed list of specifications would be useful. This document builds on the previous document *NOTCam Hawaii readout modes for the new controller* (Cox & Djupvik 2010) that described modified read modes and gave information on preferred data saving formats, but where some details have been superseded here.

### 1.2 Scope

The scope of this document is to describe the user requirements for the new controller and data acquisition system for NOTCam, i.e. the infrared specific requirements. These have been defined based on the stated performance of similar instruments (e.g. at ESO) as well as experience with NOTCam. The requirements include the whole data acquisition process, from reading the detector until the final image is stored on disk, i.e. it includes some basic data processing before the image file is stored. Any further *post-processing* or *pipe-line* that works on already stored image files will not be discussed here.

### 1.3 The terms used

In the following, the term *sequencer command* refers to a command that exists externally to the ccd3comm/controller subsystem. These commands have earlier been referred to as *external commands*. The term *low level calls* or *low level commands* are commands defined in the detector controller that can be called/used by any of the levels above.

In section 12 we list the most frequently used abbreviations.

Note that all requested commands are listed and detailed in the Appendix.

## 2 Controller Requirements

For the use of the new detector controller with the Hawaii-1 array of NOTCam we have the following requirements:

1. Shift register glow < 0.1 $e^-$/pixel/readout

2. Dark current < 0.1 $e^-$/sec at 78K

3. Full frame readout times

| **Mode** | **Read time (sec)** | **RON Goal ($e^-$)** | **RON Maximum ($e^-$)** |
|---|---|---|---|
| Default | 0.8-1 | 8-9 | 10 |
| Fast | 0.3- 0.5 | < 20 | best effort |

   A RON of 8 $e^-$ was reached with the old system. Low noise readout strategies must be sought.

4. For both the default and the fast mode there must not be variable stripes, patterns, or other systematic effects/noise in the images.

5. For all clocks and dc biases 0 and 5 volts should be used, with only Biasgate and Vreset being variable within their specified ranges.

6. For the multiple sampling readout modes the RON must be proportional to $\sqrt{(1/NSAMP)}$, where NSAMP is the number of readouts.

7. Controller must be able to read all columns.

8. The dc-gradient over each quadrant must be free of jumps.

9. The system must allow continuous reading of the array.

10. The pixel values must be stored as signed integers.

11. The reset level must not jump from one image to the next. Its drift must be < TBD $e^-$/minute.

# 3 Reset Anomaly

The reset anomaly is a non-linear change in the output level typically associated with doing a reset, though any change in clocking can produce a similar effect. The cause is due to a small change of the substrate potential of the multiplexer (Mackay 1999) resulting in the change propagating through to the measured output level of the device.

Referred to in the document on read modes (Cox & Djupvik 2010) an example of how to suppress the reset anomaly is described by continuously clocking the device with non-destructive reads (NDRs) without digitisation before doing either a reset or read of the signal. Another similar approach described in Riopel et al. (2004) is to do continuous NDRs during the integration and repetitive resets while the device is idle. The idea behind this is to 'load' (draw a constant amount of current) from the device continuously rather than have jumps in the load, e.g. as a result of a reset after being idle or start of clocking. In addition, Riopel et al. (2004) mentions "that a 3.6 ms delay must be inserted just after the read clock is raised", though no explanation is given.

Using the requested **Readout-Without-Digitisation** and the **Reset** commands it must be possible to experiment with the *Riopel* method of suppressing the effect of the reset anomaly.

To implement this successfully it is required that we can go from a **Readout-Without-Digitisation** or **Reset** command to the next command without causing a glitch, i.e., change in activity to the device, which would result in a glitch of the substrate potential and consequentially a glitch in the output level.

# 4 Exposure Commands

This section gives an overview of the different exposure commands we require. See Section 11 and the Appendix for details on the commands.

As stated in Cox & Djupvik (2010) it is required that *Fowler Sampling* is implemented but in a flexible fashion such that both the *ramp-sampling* modes (**frame, dframe, mramp, dramp**) and the *reset-read-read* modes (**mexp, mdark, burst**) can be derived from it. The actual Fowler Sampling mode takes a number of samples during an integration but concentrates them at the beginning and end of the exposure. For example the **mexp** command would be just a single sample pair, one sample at either end of the integration. The *multiple fowler-sampling* read mode is **mfowler** with its corresponding dark **dfowler**.

The other new commands are **reset, mreset, clear** and **fowler_swir** with the **reset** command having an optional period (length) parameter.

The **fowler_swir** command is the basis for all the other read modes and as such needs to be implemented as a low level controller command but with several optional parameters which define the number of sample pairs and how they are spaced. Note the maximum number of sample pairs can be limited to 32. The minimum possible exposure time increases with sample pairs. [1]

A special non-destructive read without digitisation is needed to investigate the reduction of the "Reset Anomaly". It is essential that this command, the reset and the selected read can all be executed concurrently at the controller level so no delays are introduced between their respective executions, i.e. they can not be at the sequencer level though it is necessary that a sequencer command is available to construct the low level call.

# 5 Data acquisition speed

When using the minimum possible exposure time, which is defined by the read time, an image (consisting of a pre-read and a post-read) can be obtained in **readtime + readtime** seconds. This corresponds to a continuous readout of the array, and a cycle time of **readtime + readtime + overhead** between consecutive images. The requirement is that the overhead must be less than 0.1 seconds. Our goal is to have zero overhead. For the *fast* read mode where the readtime is 0.3-0.5 seconds this means a cycle time of $\sim$ 0.6 to 1 second, and a data rate of $\sim$ 8.4 Mb to 14 Mb per second. It must be possible to switch to the *fast* read speed by setting a parameter at the application level.

The data acquisition system must handle the issues related to obtaining FITS header information, subtracting the pre-read from the post-read, showing the image on a real-time display (see Sect. 10), as well as storing the image file on disk. These processes must run in parallel to - and must not affect the data acquisition rate.

The controller upgrade will allow obtaining numerous short exposures. Using the maximum available pixel readout speed, even faster frame rates will require windowed readout (see next section) and will be useful for various image restoration techniques such as simple shift-and-add, lucky imaging, or for instance speckle holography Schödel et al. (2012). The NOTCam HR camera with 0.078"/pixel samples the diffraction limited beam of the NOT at 2.2 micron (0.22") over 2.8 pixels and delivers a total FOV of 80". Its high optical quality has been demonstrated. In the K-band 0.5 second exposures obtained with the *fast* read mode would still be sky-noise limited, provided the goal of read noise is reached.

For this kind of techniques a special exposure mode, referred to as **burst** or **cube** mode at ESO[2], should be implemented using the *fast* read speed, although it

[1] At ESO Fowler Sampling is referred to as the *FowlerNsamp* readout mode and is implemented as one of their standard readout modes with 4 reads after the reset and another 4 reads at the end of the integration, leading to a minimum exposure time of 1.79 seconds (cf. CONICA user manual). This is used for SWIR spectroscopy and narrow-band imaging, while broad-band SWIR imaging with CONICA uses Double-Read-Reset-Read with a minimum exposure time of 0.3454 seconds. For 1k Hawaii arrays the standard at ESO is reset-read-read (Double Correlated Read) and Non-Destructive Read with continuous readout.

[2] The burst mode is implemented for NaCo, VISIR, SofI, ISAAC and HAWK-I, and the overheads are minimized such that typically 80% of the execution time is spent integrating. The minimum exposure time in burst mode is 0.001 - 0.1 seconds depending on windowing and fast readout.

should also be possible to use the default speed. For each command **burst t N** one large data cube is produced with all the **N** individual images and one common fits header. A maximum **N** to limit the file size should be defined as a function of window size.

A more normal use of NOTCam is the broad-band imaging mode where multiple short exposures (a few seconds) are obtained with the typical IR readout mode *reset-read-read* to avoid saturation of bright sources while building up a deep image by averaging techniques (command **mexp t N**). For low background applications such as spectroscopy and narrow-band imaging, readout modes such as the *sample-up-the-ramp* (basic command **frame dt NSAMP** and the multiple **mramp t N**), and/or *multiple fowler sampling* (command **mfowler t NSAMP N**) will be needed to minimize the read noise and suppress the reset anomaly Finger (2000). Of these two latter modes, either **mfowler** or **mramp** is optimal for a given level of serial register glow (in practice a given exptime), but this is detector dependent and must be tested in each case.

# 6 Windowing

To facilitate faster frame rates on-chip windowing of the array must be provided. The windows must always be centred on the array, using all 4 quadrants. The goal is to have flexible setting of the window sizes, i.e. the observer should be able to select any xsize/ysize. This has a lower priority, however. As a minimum requirement we must be able to set the 3 small windows listed in Table 1.

| **Read mode** | **Pixels** | **WF fov** | **HR fov** | **Readtime** (seconds) |
|---|---|---|---|---|
| Default<br>Fast | 1024 x 1024 | 4’ | 80” | $< 1$<br>$< 0.5$ |
| Default<br>Fast | 512 x 512 | 2’ | 40” | $< 0.25$<br>$< 0.13$ |
| Default<br>Fast | 256 x 256 | 1’ | 20” | $< 0.063$<br>$< 0.031$ |
| Default<br>Fast | 128 x 128 | 30” | 10” | $< 0.016$<br>$< 0.008$ |

Table 1: Default and fast readout for full frame and 3 selected windows. The readtime consist of a fixed upstart time plus a data handling time that is proportional to the number of pixels, thus, the linear relationship given here is expected to be slightly modified for very small windows.

Windowed readout combined with the fast read mode will give a very short cycle time. Again, using the minimum exposure time, an image can be obtained in **readtime + readtime + overhead** seconds, corresponding to continuously reading out the array. For a 256 $\times$ 256 pixel window and using the fast readout mode this means a minimum exposure time of less than about **0.031** seconds, and a cycle time between such consecutive images of **0.062 + overhead** seconds.

When using the **burst t N** mode the **overhead** must be zero, as in this case no processing or storing is needed until the command finishes. Thus a bulk overhead of a few seconds due to storage of the large file at the end of each command is acceptable. The internal timing between exposures, expected to be fully controlled by the detector controller, must be accurate to microseconds. No single image time stamp is needed, but it must be possible in post-processing to reliably regenerate the time of the n'th exposure from the START and END times stored in the common FITS header. For the above example of fast readout and 256 × 256 pixels, the minimum exposure time results in 50% of the time being spent integrating, while for an exposure time of 0.3 seconds, about 90% of the execution time is spent integrating.

# 7 Shutter

The shutter in NOTCam has a limited speed and accuracy, but due to the long readout time of the old controller, its use has up to now been mandatory. The absolute minimum exposure time it can handle is 0.6 second, but ∼ 1-2% flux inaccuracies are measured for exposure times of the order of 1-2 seconds. With the new fast controller the shutter is no longer needed to determine the length of an exposure. Since NOTCam does not have a dark slide, however, we must use the shutter for this.

Most infrared cameras do not have a shutter and exposure times are defined by the time between readouts. The controller must be able to operate an IR camera without a shutter. Specifically for NOTCam we require the following:

1. The exposure time must be accurately defined by the time between readouts, not by using the shutter.

2. The shutter must be open during exposures.

3. The shutter must be closed during dark exposures.

4. The shutter must be closed in between exposure commands, when NOTCam is not in use.

5. For all multiple exposure commands (**mexp, burst, mfowler, mramp**) the shutter must remain open during the entire duration of the command, i.e. we must avoid internal overheads and unnecessary opening/closing of the shutter before/after every single exposure.

6. In addition, a command is needed to explicitly set the shutter to open or closed.

# 8 Data Saving

With a much faster system, storing everything must be possible for test purposes, regular detector quality control, and special applications, but it is not needed for most applications. Some basic image processing steps such as reset subtraction (post-read minus pre-read) and linear regression analysis are natural parts of the data acquisition itself (see Section 9).

For added user flexibility, data capacity concerns, and easier data administration, we select to follow the ESO convention for default saving options, i.e. not to store everything. Storing everything must be an option, however. This is contrary to what we suggested in Cox & Djupvik (2010). Below we describe the saving strategy for the different modes.

## 8.1 Mexp/mdark mode

The standard exposure mode for broad-band imaging is **mexp t N**, where **t** is the individual exposure time (typically a few seconds) and **N** is the number of multiple exposures. For performance verification and test purposes (detector quality control) it must be possible to store all images, i.e. the pre-read, the post-read, as well as the actual image which is the difference between the two. This means **N** files with 3 image extensions in each file, where the *difference image* is always stored as the first extension. This storage option is invoked by the command **mexp t N -s**, where the **-s** flag means save all.

The default user mode, however, must be to store only the *average image* of all (but see Section 9) the **N** *difference images*, producing one file with one image extension, where the fits header keyword EXPTIME is **t** and NAVERAGE is **N**. [3]

The minimum exposure time **t** is determined by the readtime, which is again determined by the hard-coded windowing. The number of multiple exposures **N** can be from 1 to a maximum of 1000. We explicitly do not require the single **exp/dark t** command.

## 8.2 Frame/dframe mode

The *sample-up-the-ramp* or *ramp-sampling* readout mode is used at the NOT with the exposure command **frame dt NSAMP**.[4] After a reset read (pre-read) the array is read out every **dt** seconds a total of **NSAMP** times to produce the exposure time **dt x NSAMP**. The minimum **dt** being defined by the readout time. Its corresponding dark mode is **dframe dt NSAMP** (shutter closed).

[3]This corresponds to the default ESO convention where the parameter DIT is the individual integration time and NDIT is the number of images to average to get one exposure. The additional NEXPO and NOFFSET at ESO is accommodated at the NOT in dither scripts by the number of times to repeat mexp per sky position and the number of sky positions.

[4]Note that the definition for the old system was **frame t N**. With the upgrade we modify the input parameter naming in order that **t** always means exposure time and **N** always means number of independent exposures. Thus, for **frame/dframe** the input parameters are **dt** for the time between reads and **NSAMP** for the number of reads.

The default saving option is to store all the individual readouts, as well as the *per pixel linear regression analysis (LRA)* resulting image, in one MEF fits file where there is an upper limit on **NSAMP** of 64. The LRA resulting image must always be stored in the first image extension. The individual reads shall be stored as they are read (i.e. not reset subtracted as in the old system). Storing everything is more interesting for this mode as it gives a high dynamical range.

The special case of **NSAMP = 1** is effectively the same as **mexp t 1** and the LRA result would be based only on the reset image (i.e. the pre-read) and the single post-read.

The new command **mramp t N** is a multiple *ramp-sampling* command that for **N = 1** is identical to **frame dt NSAMP**, with the difference that the user supplies the total exposure time **t** = **dt** × **NSAMP** from which the acquisition program computes the number of samples **NSAMP** to obtain the maximum possible (up to 64) for the given readout time. (See the Appendix for details.) This command has its corresponding dark command **dramp t N** to be take with the shutter closed.

## 8.3 Multiple Fowler Sampling mode

During performance verification of the basic **fowler_swir t NSAMP** mode, the optimal number of readout pairs (**NSAMP**) will be defined for typical observing modes. The resulting image of one **fowler_swir t NSAMP** command has the exposure time **t**, and the image is defined as the *difference image* from the average pre-read and the average post-read. An observer will use the higher level command **mfowler t NSAMP N**, which will take **N** multiple **fowler_swir t NSAMP** exposures with the exposure time **t**. The default data storage is that every **mfowler t NSAMP N** command produces one MEF file with an average image produced by all the **N** *difference images*. The corresponding dark is **dfowler t NSAMP N** done with the shutter closed.

In testing mode it must be possible to store all individual images with the command **mfowler/dfowler t NSAMP N -s**. This means producing **N** files with 3 extensions each, the first extension being the *difference image*, the second being the average of the **NSAMP** pre-reads, and the third the average of the **NSAMP** post-reads.

## 8.4 Burst/dburst mode

For the special **burst t N** mode with **N** multiple short exposures of exposure time **t**, all **N** images are stored in one large file with one common fits header. The maximum **N** will depend on the image size for windowed readout. It is expected that a practical limit to the file size must be set somewhere. If it is set at e.g. 256 MB, then full frame readout can have a maximum **N** of 63 while for 256 × 256 pixel windows the maximum number is around 1000. For each image only the *difference image* between the pre-read and the post-read should be stored. An average of these **N** individual difference images should be stored in the first image extension. Since the main objective with this mode is speed, however, saving all reads with no subtraction is acceptable if it improves speed. Thus, the exact maximum **N** is still to be defined. This command has its corresponding dark command **dburst t N** to be taken with the shutter closed.

### 8.5 Storage capacity

With the new NOTCam controller and data acquisition system, the maximum data rate from the controller is dramatically increased from 0.38 MB/second (mexp 1 N) to 8.4 MB/second. The typical use will depend strongly on the selected observing mode and saving/averaging option. For the extreme limit of storing everything, we also have to take into account the addition of the extra image extensions by the *difference image* for the **mexp/mdark** mode and the *LRA resulting image* for the **frame/dframe mramp/dramp** mode. The maximum data rate arriving to the hard disk is then 12.6 MB/second in the case of using **fast** full-frame readout (45 GB per hour), and the amount of data produced in a single NOTCam night could reach $\sim$ 500 GB.

## 9 Data processing

By data processing we refer here to on-line processing of the data before the final image file is stored on disk. This processing must not affect the speed of the data acquisition. Any further processing of data after the image file is stored is beyond the scope of this document.

As defined in the flexible saving options, practically all exposure modes, except for the *sample-up-the-ramp* mode, require the subtraction of the pre-read from the post-read to produce a useful image of the exposure. This subtraction is always performed and pixel values will be stored as *signed integers*.[5] The resulting image is always stored in the first image extension.

When numerous short exposures are obtained, in most cases only an average of the **N** images is of interest. This implies that for every **mexp t N** command **N** *difference images* are obtained by subtracting each pre-read from each post-read. All these *difference images* are averaged and the *average image* is stored in the final file. If there is no speed or overhead penalty involved, the average should be calculated as a *cumulative moving average*. Due to the *reset anomaly* we need to experiment with this controller/detector combination in-situ before deciding whether the first 1 or 2 images must be skipped in this averaging. The number of images to skip must be defined as a system parameter that can be set.

Similarly, for the **mfowler t NSAMP N** mode where for each exposure (**N**) a *difference image* is found from the average of all **NSAMP** pre-reads and post-reads, the default is to store the average image from all the **N** *difference images* only. For the **burst** mode, on the other hand, all individual images are always stored and the average image is calculated and added as one extra image in the MEF file.

[5] In the old system pixel values were stored as unsigned integers and negative valued pixels in difference images wrapped around to high values.

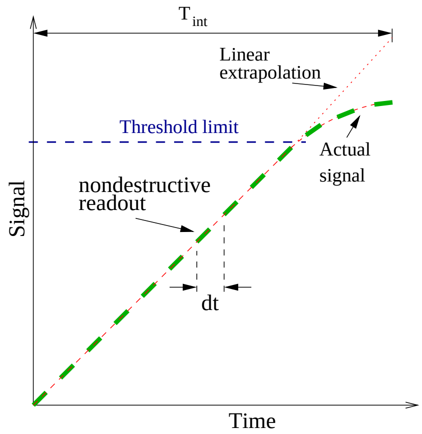


Figure 1: threshold-limited integration

For the *ramp-sampling* exposure modes (**frame/dframe** and **mramp/dramp** commands) the final image is produced by a pixel-by-pixel linear regression analysis (LRA) on all readouts, including the first readout which is the reset readout (or pre-read). The slope of the regressional fit and the exptime (**dt x NSAMP**) then gives the final image where the read noise is reduced by approximately $\sqrt{(1/NSAMP)}$.

The LRA should be implemented using the principle of *threshold-limited integration*(see Fig. 1).[6] For every pixel, all the reads with ADU levels below a given threshold (determined by the linear range), are used to calculate the slope of the regressional fit. The signal level written to the final image is the value calculated from the fit evaluated at the total exposure time by exrapolation. Note that memory effects produced by saturation or near to saturation exposure, will not be avoided by this approach.

The threshold value is the absolute ADU level, i.e. the level including the reset level. The threshold should be a system variable that can be set.

# 10 Display

As part of the data-acquisition system we require a real-time image display that displays images as they are being read out and processed. It is expected that an additional display will be available for stored images in order to display for instance sky-subtracted data (e.g. subtract the n'th previous image), but that is beyond the scope of this document. We specify below the requirements for the real-time display:

[6] Used for instance with HAWK-I at ESO.

1. It should be possible to switch the real-time displaying of an image ON or OFF with a sequencer command (for instance, **autoshow_off/autoshow_on**).
2. The display must not erase the previous image before the new image is being displayed.
3. For the **mexp/mdark t N** and the **mfowler/dfowler t NSAMP N** commands, the display should show the *average image* as it builds up through the calculation of a *cumulative moving average*, irrespectively of the storing options selected. If not, then the individual *difference images* must be displayed successively.
4. For the **burst/dburst t N** commands the real-time display must show only the final average of the **N** individual *difference images* to avoid adding any overheads to calculations and display.
5. For the **frame/dframe dt NSAMP** and the **mramp/dramp t N** commands the real-time display must show every successive non-destructive readout, as well as the final LRA result.

# 11 Summary of Commands

The commands are separated into two levels, the actual command executed by the controller and the higher calling command (see Sect 1.3). An example is the **mfowler** command where at the low level, referred to as a *Direct* command and given the extension '_swir' to identify it, it acquires all the specified images for a given exposure, but at the higher level (an *Application* command) can be considered as a Sequencer (e.g. **mexp**) command. Some commands will be identical at both the *Direct* and *Application* level, e.g. **reset** has the same functionality to reset the whole array, at both levels.

1. **reset t** - single reset with an optional timing parameter
2. **mreset N t** - multiple resets
3. **clear** - This is basically a mreset or long reset
4. **fowler_swir t NSAMP** Fowler sampling readout, where **t** is the exposure time, or the separation between readout pairs, and **NSAMP** is the number of sample pairs.
5. **mfowler/dfowler t NSAMP N** Multiple fowler sampling mode, where **t** is the exposure time, or the separation between readout pairs, **NSAMP** is the number of readout pairs and **N** is the number of exposures. Darks are obtained with **dfowler**.
6. **set-shutter status** A command to set the shutter to a given status, either "open" or "closed".
7. **abort** Abort an on-going exposure.

8. **burst/dburst t N** Burst mode for very fast frame rates, **N** consecutive images of exptime **t**. All images are saved in one MEF file. Darks are obtained with **dburst**.

9. **mexp/mdark t N** Multiple exposure mode, **N** images with individual exposure time **t**.

10. **frame_swir dt NSAMP** Sample-up-the-ramp integration with **NSAMP** non-destructive readouts every **dt** seconds.

11. **frame/dframe dt NSAMP** Sample-up-the-ramp integration with **NSAMP** non-destructive readouts every **dt** seconds, producing an image with exposure time **dt x NSAMP**. Darks are obtained with **dframe**.

12. **mramp/dramp t N** These are **N** multiple **frame/dframe** commands where **t** is the total exptime (i.e. **dt** × **NSAMP** in **frame/dframe**) and the number of samples to be read is internally computed from **t**. Produces **N** MEF files.

The following commands are required to be able to experiment with a method of reducing the *Reset Anomaly* mentioned in Cox & Djupvik (2010). It would be desirable that we had the ability (documentation) to create these commands ourselves so any possible changes can also be implemented in house.

1. **read-without-digitisation** Read, but do not digitise.

2. **mread-without-digitisation** Multiple reads of the array, but do not digitise.

3. **integrate t** Starting an independent integration (without reset).

4. **read** Read the array.

A complete description of commands is found in the Appendix.

# 12 Abbreviations used

- ADU = Analog-Digital Unit
- DS9 = Imaging and data visualization application from SAOimage
- ESO = European Southern Observatory
- FOV = Field Of View
- HR = High Resolution
- LRA = Linear Regression Analysis
- MEF = Multi-Extension Fits format
- NDR = Non-Destructive Read
- RON = Read Out Noise
- SWIR = Short Wavelength InfraRed

# References


T. Augusteijn & G. Cox, 2007, *Detector Controller User Requirements*

G. Finger, 2000, *Performance of large format HgCdTe and InSb arrays for low background applications. Optical and IR telescope Instrumentation and Detectors*, SPIE Vol 4008

G. Cox & A.A. Djupvik, 2010, *NOTCam Hawaii readout modes for the new controller*

A.A. Djupvik, 2008, *NOTCam array electronics upgrade Dec 2007*, http://www.not.iac.es/instruments/notcam/staff/newpcb/newpcb.html NOT Internal Report

R. Schödel, S. Yelda, A. Ghez et al., 2012, *Holographic imaging of crowded fields: an effective poor man's adaptive optics system*, submitted to MNRAS, arXiv:1110.02261v3

Craig D. Mackay, 1999 *Reset Anomaly*, www.ast.cam.ac.uk/∼optics/cirsi/documents/resetanomaly.html

M. Riopel, R. Doyon, D. Nadeau, C. Marois, 2004 *An Optimized Data Acquisition System Without reset Anomaly for the Hawaii and Hawaii-2 Arrays*, Kluwer Academic Publishers, Astrophysics and Space Science Library, Vol. 300, p453-457.

## RESET [t]

**Description :** Reset the whole detector with an optional reset pulse length argument

**Command type :** Direct/Application

**Arguments :** `t` : Optional length of the individual reset pulses ('ResetB') in microseconds with a resolution of 100ns. Range from TBD to 20.0$\mu$s>). Default = TBD

**Timing :** Must return immediately on completion i.e. negligible latency

**Returns :** no data

**Example :**

```
reset 1.2
```

or

```
reset
```

## MRESET N [t]

**Description :** Multiple resets of the whole detector with optional pulse length argument

**Command type :** Direct/Application

**Arguments :**

`N` : Number of resets. Range <1 - 20>.
`t` : Optional length of the individual reset pulses ('ResetB') in microseconds with a resolution of 100ns. Range TBD - 20.0$\mu$s>. Default = TBD

**Timing :** Return immediately on completion i.e. negligible latency

**Returns :** no data

**Example :**

```
mreset 3 1.2
```
or
```
mreset 3
```

## CLEAR

**Description :** Clear the whole detector. This will call the *Direct* command `reset` with the default pulse length

**Command type :** Application

**Arguments :** None

**Timing :** No specific requirements

**Returns :** no data

**Example :** `clear`

## **FOWLER_SWIR** t [NSAMP]

**Description :** Fowler sampling readout, where a number **NSAMP** of overlapping pairs of readouts are done separated by the desired exposure time **t**. After the initial reset the first **NSAMP** readouts are concentrated at the start of the integration and the second **NSAMP** readouts are at the end of the integration. The exposure time **t** is defined as the period between the Fowler pairs. The multiplexer glow-induced photon shot noise limits the number of Fowler pairs to $\leq 32$.

**Command type :** Direct

**Arguments :**

`t` : Exposure time. Separation between the sample pairs
`NSAMP` : Number of sample pairs. Range <1 - 32>.

**Timing :** Return immediately on completion i.e. negligible latency

**Returns :** all data

**Example :**

`fowler_swir 60 10`
or
`fowler_swir 60 1` (this is equivalent to Reset-Read-Read)

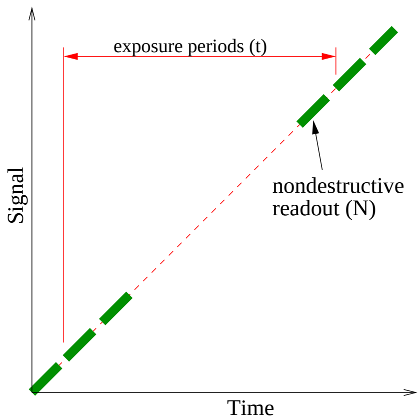


Figure 2: Fowler sampling

# **MFOWLER** t NSAMP N -s

**Description :** Multiple executions of the Fowler exposure mode. This command calls the low level `fowler_swir` command but includes `N` multiple exposures and the extra '`-s`' optional argument for requesting all data to be saved to disk. Fowler sampling readout, where a number **NSAMP** of overlapping pairs of readouts are done, separated by the desired exposure time **t**. The first **NSAMP** readouts are concentrated at the start of the integration and the second **NSAMP** readouts are at the end of the integration. The exposure time **t** is defined as the period between the sample pairs. An averaged frame is computed from all the Fowler pairs of all calls. The shutter is to remain open during the entire multiple exposure mode.

**Command type :** Application

**Arguments :**

| | |
|---|---|
| `t` : | Exposure time. Separation between the sample pairs. |
| `NSAMP` : | Number of sample pairs. Range <1 - 32>. |
| `N` : | Number of Fowler mode exposures |
| `-s` : | All N images are saved |

**Timing :** Return the prompt immediately on completion of the integration

**Returns :** Per default one MEF file, containing a single image being the average of the N difference images. Each difference image is the average of the NSAMP post-reads minus the average of the NSAMP pre-reads. If the `-s` option is used, N MEF files are produced, each with 3 image extensions: in the first the difference image, in the 2nd and 3rd extension the average of the NSAMP pre-reads and post-reads, respectively.

**Example :**

| | |
|---|---|
| `mfowler 60 10 5` | Return the average of the 5 difference images, EXPTIME = 60, NAVERAGE = 5 |
| or | |
| `mfowler 60 1 5` | Equivalent to mexp 60 5 |
| or | |
| `mfowler 15 4 5 -s` | Save 5 MEF files, each with 3 extensions; in the 1st the difference image, in the 2nd the average of the 4 pre-reads, and in the 3rd the average of the 4 post-reads |

## DFOWLER t NSAMP N -s

| | |
|---|---|
| **Description :** | Multiple dark (shutter closed) executions of the Fowler read. This command calls the low level `fowler_swir` command but includes `N` multiple integrations and the extra '`-s`' optional argument for requesting all data to be saved to disc. Fowler sampling readout, where a number `NSAMP` of overlapping pairs of readouts are done, separated by the desired exposure time `t`. The first `NSAMP` readouts are concentrated at the start of the integration and the second `NSAMP` readouts are at the end of the integration. The exposure time `t` is defined as the period between the sample pairs. An average image is computed from all the Fowler pairs of all calls. The shutter is to remain closed during the entire multiple integration mode. |
| **Command type :** | Application |

**Arguments :**

| | |
|---|---|
| `t` : | Integration time. Separation between the sample pairs. |
| `NSAMP` : | Number of sample pairs. Range <1 - 32>. |
| `N` : | Number of Fowler mode integrations |
| `-s` : | All images are saved |

| | |
|---|---|
| **Timing :** | Return the prompt immediately on completion of the integration |
| **Returns :** | Per default one MEF file, containing a single image being the average of the N difference images. Each difference image is the average of the NSAMP post-reads minus the average of the NSAMP pre-reads. If the `-s` option is used, N MEF files are produced, each with 3 image extensions: in the 1st the difference image, in the 2nd and 3rd extension the average of the NSAMP pre-reads and post-reads, respectively. |

**Example :**

| | |
|---|---|
| `dfowler 60 10 5` | return the average of the 5 difference images, EXPTIME = 60, NAVERAGE = 5 |
| or | |
| `dfowler 15 4 5 -s` | Saves all 5 integrations in 5 MEF files, each with 3 extensions; in the 1st the difference image, in the 2nd the average of the 4 pre-reads, and in the 3rd the average of the 4 post-reads |

## SET-SHUTTER status

**Description :** A sticky command to set the shutter to a given status, either "open", or "closed". This the main shutter control command.

**Command type :** Direct/Application

**Syntax :** `set-shutter status`

**Arguments :** `status` : `open` or `closed`

**Timing :** No specific requirements

**Returns :** no data

**Example :** `set-shutter open`

## **FRAME_SWIR** dt NSAMP

**Description :** After an initial reset image, the actual integration, divided up in `NSAMP` equally separated (by the amount `dt`) non-destructive reads is started. This command will be called by a higher level command including data saving requirements.

**Command type :** Direct

**Arguments :**
`dt` : Time between sample reads
`NSAMP` : Number of nondestructive reads, range <1- 64>.

**Timing :** Prompt returned immediately on completion of the last read

**Returns :** all read data

**Example :** `frame_swir 10 32`

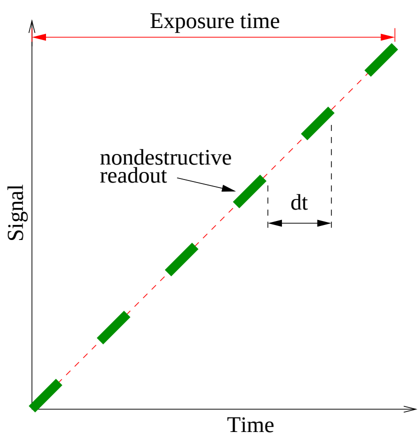


Figure 3: Ramp sampling

# **FRAME** dt NSAMP

| | |
|---|---|
| **Description :** | This command calls the low level `frame_swir` command. After an initial reset image, the array is read out non-destructively every `dt` time interval a total of `NSAMP` times. A linear regression analysis (LRA) is done on the resulting data for each pixel up to the specified threshold limit and the linear extrapolated result saved. The total exposure time is `dt` × `NSAMP`. For each individual readout, i = 1,2,..NSAMP, the EXPTIME keyword stored in the FITS header of the i'th image is `dt` × `i`. |
| **Command type :** | Application |
| **Arguments :** | `dt` : Time between non-destructive reads, the minimum `dt` being defined by the readtime<br>`NSAMP` : Number of equally separated non-destructive reads, range <1- 64>. |
| **Timing :** | Prompt returned immediately on completion of the last read |
| **Returns :** | One MEF file with NSAMP + 2 image extensions. Includes the reset frame, all individual reads, and the linear regression result image, the latter being stored in the first image extension. |
| **Example :** | `frame 10 32` — Returns a MEF file with 34 extensions. One reset image, 32 reads and one LRA result image. Total exptime is 320 seconds. |

## **DFRAME** dt NSAMP

**Description :** This command calls the low level `frame_swir`. The difference with `frame` is that the shutter remains closed. After an initial reset image, the array is read out non-destructively every `dt` time interval a total of `NSAMP` times. A linear regression analysis (LRA) is done on the resulting data for each pixel up to the specified threshold limit and the linear extrapolated result saved. The total exposure time is `dt` $\times$ `NSAMP`. For each individual readout, i = 1,2,..NSAMP, the EXPTIME keyword stored in the FITS header of the i'th image is `dt` $\times$ `i`.

**Command type :** Application

**Arguments :**

- `dt` : Time between non-destructive reads, the minimum `dt` being defined by the readtime
- `NSAMP` : Number of equally separated non-destructive reads, range <1- 64>.

**Timing :** Prompt returned immediately on completion of the last read

**Returns :** MEF file of NSAMP + 2 image extensions. Includes reset frame, all individual reads and the linear regression image, the latter stored in the first extension.

**Example :**

| | |
|---|---|
| `dframe 10 32` | A dark integration of 320 seconds, read out every 10 seconds. All readouts, reset image and LRA resulting image stored in one MEF file with 12 image extensions. |

## MRAMP t N

**Description :** This command is a multiple `frame` command that gives `N` independent ramp-sampling exposures (i.e. `N` MEF files). Here the total exposure time `t` is given as input and the number of reads `NSAMP` and the time between reads `dt` is calculated. This command calls the low level `frame_swir dt NSAMP` command. After an initial reset image, the actual integration time `t` is divided in equally spaced non-destructive reads (NDR). A linear regression analysis is done on the resulting data up to the specified threshold limit, and the linear extrapolated result saved. The number of samples NSAMP is calculated based on the length of the integration `t`, the readout time, and the maximum allowed number of samples (64).

**Command type :** Application

**Arguments :**
- `t` : Exposure time, i.e. total length of exposure
- `N` : Number of exposures

**Timing :** Prompt returned immediately on completion of the last read

**Returns :** N number of MEF files, each including the reset frame, the individual reads and the linear regression image, the latter in the first image extension.

Since **mramp t N** $\equiv$ **N** $\times$ "**frame dt NSAMP**", in order to calculate **dt** (time between reads) and **NSAMP** (number of samples, excluding the reset frame) use the following relations:

| | |
|---|---|
| Minimum t (exposure time) = readtime: | NSAMP = 1<br>dt = readtime |
| For t > readtime & t <= 64 × readtime: | NSAMP = int(t/readtime)<br>dt = t/NSAMP |
| For t > 64 × readtime: | NSAMP = 64<br>dt = t/64 |

**Example :** `mramp 300 3` Gives 3 independent MEF files (exposures), each with 66 image extensions: a reset image, then 300/64 sample period NDRs (i.e. reading out every 4.6875 seconds), and the LRA result image, the latter stored in the first extension. The exptime of each exposure is 300 seconds.

## **DRAMP** t N

**Description :** This command is a multiple `dframe` command that gives `N` independent ramp-sampling integrations (i.e. `N` MEF files). Here the total exposure time `t` is given as input and the number of reads `NSAMP` and the time between reads `dt` is calculated. This command calls the low level `dframe_swir` command. After an initial reset image, the actual integration time `t` is divided in equally spaced non-destructive reads (NDR). A linear regression analysis is done on the resulting data up to the specified threshold limit, and the linear extrapolated result saved. The number of samples NSAMP is calculated based on the length of the integration `t`, the readout time, and the maximum allowed number of samples (64). See details under `mramp`.

**Command type :** Application

**Arguments :**

- `t` : Total length of integration
- `N` : Number of integrations

**Timing :** Prompt returned immediately on completion of the last read

**Returns :** N number of MEF files, each including the reset frame, the individual reads and the linear regression image, the latter stored in the first image extension.

**Example :**

| | |
|---|---|
| `dramp 300 3` | Gives 3 independent MEF files (dark integrations), each with 66 image extensions: a reset image, 300/64 sample period NDRs (reading out every 4.6875 seconds), and the LRA result image, the latter stored in the first extension. |

## **MEXP** t N -s

**Description :** Makes multiple `N` exposures of `t` seconds. The shutter should be open during the entire duration of the multiple exposure command.

**Command type :** Application/Direct

**Arguments :**

| | |
|---|---|
| `t` : | Exposure time |
| `N` : | Number of multiple exposures |
| `-s`: | Optionally store everything |

**Timing :** no specific requirements

**Returns :** By default returns one MEF file with one image extension which contains the average of all the N or N-1 or N-2 *difference images*. It must be possible to exclude the first 1 or 2 images in this averaging. Most probably the default will be to exclude the 1st but this needs to be determined during commissioning. If the storing option `-s` is selected, then it returns N MEF files, one for each exposure, each file having 3 image extensions. In the first extension the *difference image* (i.e. the post-read minus the pre-read), in the 2nd extension the pre-read, and in the 3rd extension the post-read.

**Example :**

| | |
|---|---|
| `mexp 2 30` | Gives one MEF file with one image: the average of all the 30 *difference images*, each being 2 second exposures. This image has NCOMBINE=30 and EXPTIME=2. |
| `mexp 10 1` | One MEF file with one image: the *difference image* obtained by subtracting the pre-read from the post-read. EXPTIME=10 and NCOMBINE=1. |
| `mexp 5 12 -s` | Gives 12 MEF files, each with 3 image extensions, storing the difference image, the pre-read and the post read. The difference images have EXPTIME = 5 and NCOMBINE = 1. |

## MDARK t N -s

**Description :** Makes multiple `N` integrations of `t` seconds. The shutter should be closed during the entire duration of the multiple exposure command.

**Command type :** Application

**Arguments :**

| | |
|---|---|
| `t` : | Integration time |
| `N` : | Number of multiple dark integrations |
| `-s`: | Optionally store everything |

**Timing :** no specific requirements

**Returns :** By default returns one MEF file with one image extension which contains the average of all the N or N-1 or N-2 *difference images*. It must be possible to exclude the first 1 or 2 images in this averaging. Most probably the default will be to exclude the 1st but this needs to be determined during commissioning. If the storing option `-s` is selected, then it returns N MEF files, one for each exposure, each file having 3 image extensions. In the first extension the *difference image* (i.e. the post-read minus the pre-read), in the 2nd extension the pre-read, and in the 3rd extension the post-read.

**Example :**

| | |
|---|---|
| `mdark 2 30` | Gives one MEF file with one image: the average of all the 30 *difference images*, each being 2 second exposures. This image has NCOMBINE=30 and EXPTIME=2. |
| `mdark 10 1` | One MEF file with one image: the *difference image* obtained by subtracting the pre-read from the post-read. EXPTIME=10 and NCOMBINE=1. |
| `mdark 5 12 -s` | Gives 12 MEF files, each with 3 image extensions, storing the difference image, the pre-read and the post read. The difference images have EXPTIME = 5 and NCOMBINE = 1. |

## ABORT

**Description :** Terminate the currently executing command and return the prompt!

**Command type :** Application

**Arguments :** None

**Timing :** no specific requirements

**Returns :** nada

**Example :** `abort`

## **BURST** t N

**Description :** Continuous reset-read-read `N` times with an integration time of `t` seconds. For this mode the data is saved as N reset-subtracted *difference images* in one single MEF file. In addition, in the first MEF extension an average image of all N difference images shall be stored. The maximum size of `N` will be determined by the amount of data that can be temporarily stored (memory buffer?) before creating the final fits file on disk. This mode requires the highest possible speed, for which having one common FITS header per command is regarded as sufficient. Also, the storage requirements can be slightly modified if that improves speed.

**Command type :** Application

**Arguments :**

- `t` : Length of individual exposures
- `N` : Number of pairs of reads

**Timing :** Prompt returned immediately on completion of the last read

**Returns :** Always one MEF file with N+1 extensions, an average image in the first extension and all the reset subtracted images in the following extensions.

**Example :**

| | |
|---|---|
| `burst 0.5 100` | One MEF file with 101 image extensions where the 100 difference images are stored together with the average image, the latter in the first image extension. |

## DBURST t N

**Description :** Continuous reset-read-read `N` times with an integration time of `t` seconds, with the shutter closed. For this mode the data is saved as N reset-subtracted *difference images* in one single MEF file. In addition, in the first MEF extension an average image of all N difference images shall be stored. The maximum size of `N` will be determined by the amount of data that can be temporarily stored (memory buffer?) before creating the final fits file on disk. This mode requires the highest possible speed, for which having one common FITS header per command is regarded as sufficient. Also, the storage requirements can be slightly modified if that improves speed.

**Command type :** Application

**Arguments :**

`t` : Length of individual integrations
`N` : Number of pairs of reads

**Timing :** Prompt returned immediately on completion of the last read

**Returns :** Always one MEF file with N+1 extensions, an average image in the first extension and all the reset subtracted images in the following extensions.

**Example :** `dburst 0.5 100` One MEF file with 101 image extensions where the 100 difference images are stored together with the average image, the latter in the first image extension.

## READOUT-WITHOUT-DIGITISATION

**Description :** Readout whole array but without digitising the signal. i.e. address all the pixels with the **READ** clock enabled

**Command type :** Direct/Application

**Syntax :** `readout-without-digitisation`

**Arguments :** None

**Timing :** Return immediately on completion i.e. negligible latency

**Returns :** no data

**Example :** `readout-without-digitisation`

## MREADOUT-WITHOUT-DIGITISATION N

**Description :** Multiple **N** Readouts whole array but without digitising the signal.i.e. address all the pixels with the **READ** clock enabled

**Command type :** Direct/Application

**Arguments :** `N` : Number of resets

**Timing :** Return immediately on completion i.e. negligible latency

**Returns :** no data

**Example :** `readout-without-digitisation 5`

## INTEGRATE t

| | |
|---|---|
| **Description :** | Make an exposure without resetting the array. |
| **Command type :** | Direct/Application |
| **Arguments :** | `t` : Length of exposure |
| **Timing :** | Return immediately on completion i.e. negligible latency |
| **Returns :** | nothing |
| **Example :** | `integrate 10` |

## READ

**Description :** Read the array.

**Command type :** Direct/Application

**Arguments :** none

**Timing :** Return immediately on completion i.e. negligible latency

**Returns :** FITs file with a single frame of data. Header keyword values TBD.

**Example :** `read`