

# UNIVERSITY OF HAWAII AT MĀNOA

Institute for Astronomy

---

Pan-STARRS Project Management System

## OTA FITS

Coop. Agreement No. : FA9451-06-2-0338  
Prepared For : Pan-STARRS  
Prepared By : John Tonry and Sidik Isani  
Document No. : PSDC-710-004-01  
Document Date : August 20, 2008  
Revision : 01

### DISTRIBUTION STATEMENT

Approved for Public Release – Distribution is Unlimited

Submitted By:

\_\_\_\_\_  
[Insert Signature Block of Authorized Developer Representative]

\_\_\_\_\_  
Date

Approved By:

\_\_\_\_\_  
[Insert Signature Block of Customer Developer Representative]

\_\_\_\_\_  
Date

The information here is available as HTML. The URL is: [https://svn.ifa.hawaii.edu/gpc/archive/gpcfits\\_doc/FITS-head/](https://svn.ifa.hawaii.edu/gpc/archive/gpcfits_doc/FITS-head/)

## Revision History

Revision Number	Release Date	Description
DR1	2004.04.20	Original text file from JT
DR2	2005.01.27	1st version accepted to PSDC from Brian to Will
DR3	2005.02.15	Reformatting so HTML conversion works again
00	2005.03.08	Fix PSDC and submit from Sidik to Will
01	2008.08.20	New info about DARKTIME, CELLMODE, video extensions, and many cleanups.

## TBD / TBR Listing

Section No.	Page No.	TBD/R No.	Description
5.6	12	1	Remove Other Random Keywords section completely after promoting any keywords of value to Recommended or Mandatory.
5.6	15	2	Sidik: I'm not sure I like this proposed accelerator format, exactly as shown here.
6	16	3	Look into combining parts of fhtool that I wrote at CFHT with fitsverify that Andrey says is available from NASA.
6	16	4	Also download, check, and reference the latest FITS Standard (3.0) which just came out.

## Abstract

Requirements for OTA FITS headers and OTA coordinate systems are defined here. Details of the other data products bundled in the MEF format with the normal image extensions, such as 3D video and tables are described as well. Coordinates (ideal, pixel based ones) are described at length. Two redundant systems are present in our headers. One is the NOAO Mosaic Iraf/DS9 compatible section notation one, based on a coordinate system with an origin at the center of the first pixel. The second system uses a coordinate system with an origin at the outer corner of the image region, and uses keywords like CNPIX (integer offset counts) and an IMNPIX coordinate system for the whole OTA with its origin at the lower left of cell00.

## Contents

<b>1</b>	<b>Essential information to come out of an OTA</b>	<b>1</b>
1.1	Data images from many cells	1
1.2	Shift information for many cells	1
1.3	Video data from guide cells	1
1.4	Time stamps	2
1.5	Exposure time	2
<b>2</b>	<b>One FITS MEF per OTA</b>	<b>2</b>
<b>3</b>	<b>Coordinate Systems</b>	<b>3</b>
3.1	Data coordinates (NAXIS1 by NAXIS2)	3
3.2	Cell coordinates (CNAXIS1 by CNAXIS2)	4
3.3	OTA coordinates (IMNAXIS1 by IMNAXIS2)	5
3.4	Coordinate Summary	7
<b>4</b>	<b>OTA MEF Data</b>	<b>7</b>
4.1	Science Image Cells	7
4.2	Video Cells	7
4.3	Tables	7
4.3.1	FITS tables for times, OT shifts, video origins, etc	7
<b>5</b>	<b>OTA MEF Headers</b>	<b>9</b>
5.1	OTA Cell Usage and Accelerators	9
5.2	Keyword Inheritance	10
5.3	Keywords We Do Not Use	10
5.4	Mandatory OTA MEF Keywords	10
5.4.1	Required Primary Header Keywords	10
5.4.2	Required Cell Extension Keywords	11
5.4.3	Required Primary Heady Keywords - Provided by Controller	11
5.5	Recommended OTA MEF Keywords	12
5.6	Other Random Keywords	13
<b>6</b>	<b>Validation Tool</b>	<b>16</b>

## List of Figures

1	OTA Cell Numbering	1
2	NAXIS (pixel data) coordinate system	3
3	CNAXIS (cell) coordinate system	4
4	IMNAXIS (image) coordinate system	6

# 1 Essential information to come out of an OTA

## 1.1 Data images from many cells

We will adopt a two digit numbering scheme following Cartesian coordinates, cell 00 is the lower left, cell 70 is the lower right, cell 01 is the left cell in the first row up, etc. This is the view of the detector as one looks at the imaging surface of the silicon. *The optical system forming an image on the detector may (as in the case of Pan-STARRS) form a mirror image in which case most displays would be flipped left-to-right to make this image correct with respect to the sky.*

In general, not all cells will have data; some will be dead and some will be used for guiding.

xy07	xy17	xy27	xy37	xy47	xy57	xy67	xy77
xy06	xy16	xy26	xy36	xy46	xy56	xy66	xy76
xy05	xy15	xy25	xy35	xy45	xy55	xy65	xy75
xy04	xy14	xy24	xy34	xy44	xy54	xy64	xy74
xy03	xy13	xy23	xy33	xy43	xy53	xy63	xy73
xy02	xy12	xy22	xy32	xy42	xy52	xy62	xy72
xy01	xy11	xy21	xy31	xy41	xy51	xy61	xy71
xy00	xy10	xy20	xy30	xy40	xy50	xy60	xy70

Figure 1: OTA Cell Numbering

NOTE that this scheme is ordered (x,y), not (y,x).

## 1.2 Shift information for many cells

For each cell there will be a sequence of two integers which represent the OT shifts which have been applied. These are all contemporaneous (or nearly so), so there is also a vector of time stamps which tells when each of these shifts took place. The shifts will be encoded as present location with respect to the original position, so the incremental shift which was actually applied between iterations is the difference between two.

## 1.3 Video data from guide cells

Some cells will have video data which is a data cube of images with origin, size, and binning. A cell that was used for video will have the usual 2D image extension in the MEF replaced by the 3D stack of video frames.

We can insist the size and binning be fixed for an exposure, but the origin will vary as the guide box may move during the course of an exposure. These video frames have the same number and share the same time stamp as the shift information,

except when shifts occur after averaging several video frames. (This stamp is again an approximation, but good enough since the lag between guide star read and application of shift should be well less than 10 msec.)

## 1.4 Time stamps

We want a vector of time stamps of when video frames are read out and shifts are applied. These should have high relative accuracy (less than 0.1 msec) so that we can do accurate photometry of video frames as short as 10 msec. There are three time intervals in this vector: time the video exposure started, duration of video exposure, and processing overhead. We expect that the duration plus overhead plus start time should equal the start time of the next frame. (With OPTIC it turns out to be useful to make a distinction between overhead created by the SDSU controller and overhead from the host computer. Also it may be that we want to make a distinction between video exposure duration and OT shift dwell time.)

Metadata (such as time stamps) for each frame of video will be stored in a separate 2D FITS table extension. All FITS tables will be stored after the image extensions for an OTA's cells. A keyword in the 3D video extension will point to the name of the FITS table extension and vice versa.

*Improvements to the video mode for Giga Pixel Camera 1 are now in development to support multiple guide boxes with cross-triggering (synchronization) and monotonic timestamps (perhaps rendering them obsolete, so we can record just a start time and the exact frame.)*

## 1.5 Exposure time

In general we must expect a shutter which gives us a start time and exposure duration which depend on position. Each cell will have parameters of an equation which tells us these times as a function of position. On a cell by cell basis, we expect that a linear equation is fine.

*The shutter for Giga Pixel Camera 1 has not yet been profiled. A set of SHUT\* keywords are recorded with the absolute time of day that the shutter was commanded to open, to better than a millisecond.*

## 2 One FITS MEF per OTA

All data from a single OTA will form a single FITS MEF file. This is probably the simplest and most portable format for copying, and lends itself easily to unbundling into different formats (e.g. 5K x 5K contiguous image). We take advantage of the garbage can nature of MEF to bundle together the 2D cells, the 3D video, the tabular time stamp and shift data, and possibly the tabular guide star data.

There will be mandatory FITS keywords for OTA MEF files, for example, the nomenclature for cell extensions (the EXTNAME keyword) can not be arbitrary. In a later section, we list typical keywords and indicate which ones are mandatory and which ones have reserved keyword names.

Because OTA cells are independent CCDs, features such as overlocks making bias strips should be saved as part of the pixel data making up that cell's extension (as opposed to creating a new extension). However, an important design feature of OTAs is the fact that the cells have integer pixel offsets from one another, so the metrology of the entire OTA should be as good as the metrology of a single cell. Mosaics of CCDs are often stored in FITS with a detector section (DET-SEC) for Iraf which is meant to be approximate, and WCS or other coordinate systems provide a means of quantifying misalignments between CCDs. For OTA, our detector sections are exact.

### 3 Coordinate Systems

The NAXIS coordinate system is inherent in each 2D FITS extension. In addition, we define two of our own unbinned coordinate systems; one for the cell, and one for the entire OTA.

- 3.1 The NAXIS coordinate system is standard to all FITS image data. We treat it and all other coordinate systems as having an origin at one corner of the region read out. The center of the lower left pixel has a coordinate of (0.5,0.5). This is in accord with some extant analysis software and contrary to other; be aware of this distinction and definition if your application is sensitive to sub-pixel coordinates.
- 3.2 A cell's coordinate system, like NAXIS, is oriented from the point of view of the amplifier, and all data is stored respecting the sequence in which pixels were read from the readout electronics. A full cell would have the lower left corner of its lower left pixel (the one closest to the amplifier) at the origin. Units are always in *unbinned* pixels, and the origin is always in the same place even if the full cell is not read.
- 3.3 OTA coordinates are Cartesian, and oriented to represent the image seen by the detector. OTA  $y=0$  coincides with  $y=0$  of the "bottom" row of cells,  $xy00$  through  $xy70$ . OTA  $x=0$  coincides with the left edge of the column of cells  $xy00$  through  $xy07$ . Note that OTA  $x=0$  never coincides with any cell's  $x=0$  because the amplifiers are on the lower right of the cells, as seen from OTA coordinates. The following sections illustrate this concept. Like a cell's coordinates, units are *unbinned* pixels.

#### 3.1 Data coordinates (NAXIS1 by NAXIS2)

The units of data coordinates are by necessity binned pixels. NAXIS1 NAXIS2 give the total number of pixel columns and rows that are in each image extension, including overscan. Our keyword for the number of overscan readings in a row is OVRSCAN1. (For completeness, OVRSCAN2, PRESCAN1, and PRESCAN2 exist, but are always 0 for OTA.) OTA data will always keep per-amplifier extensions in amplifier order. Given this, and the fact that there is no prescan, the number of image columns read is always NAXIS1 minus OVRSCAN1, with the image pixels first then the overscan, repeated for each row, moving "up" the cell. The number of binned rows read is NAXIS2.

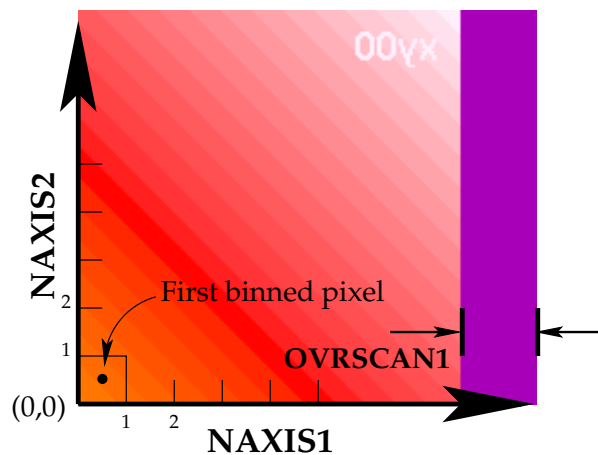


Figure 2: NAXIS (pixel data) coordinate system

*Iraf*: section notation keywords *DATASEC* and *BIASSEC* contain the same information, but beware that these use a coordinate system with an origin  $-0.5, -0.5$  from the corner of the imaging area. The origin of our coordinate systems always coincides with the corner of the imaging area or a pixel. See the appendix for more on the implications of this.

### 3.2 Cell coordinates (CNAXIS1 by CNAXIS2)

The values of keywords CNAXIS1 CNAXIS2 give the unbinned imaging size of a cell (e.g. 480 x 496). We do not allow for any rotation or flipping between CNAXIS and NAXIS. They only differ by translation and scaling. Cell coordinates are always unbinned, and the origin always coincides with the lower left corner of the imaging region of the cell, the corner near the amplifier.

Keywords CNPIX1 CNPIX2 give the location (in CNAXIS-space) of the data coordinate system origin. In other words, the number of unbinned columns and the number of unbinned rows that the readout electronics skipped for a windowed readout. If the full cell was read, CNPIX1 CNPIX2 are both 0.

Keywords CCDBIN1 and CCDBIN2 give the length, in unbinned pixels of an NAXIS1 and NAXIS2 unit. These are just the binning factors used by the readout electronics. (E.g., if CCDBIN1 is 2, then each pixel on NAXIS1 represents 2 units along CNAXIS1.)

Note that CNPIX1 CNPIX2 CCDBIN1 CCDBIN2 are the only four keywords needed to go from data to cell coordinates, with this transformation:

$$\begin{aligned}x_{cell} &= (CCDBIN1)x_{data} + CNPIX1 \\y_{cell} &= (CCDBIN2)y_{data} + CNPIX2\end{aligned}$$

For a full unbinned readout, cell and data coordinate systems are identical. But suppose only the far quadrant, diagonally opposite the amplifier was read with binning factors 2 (serial) and 4 (parallel). The transformation would look like this:

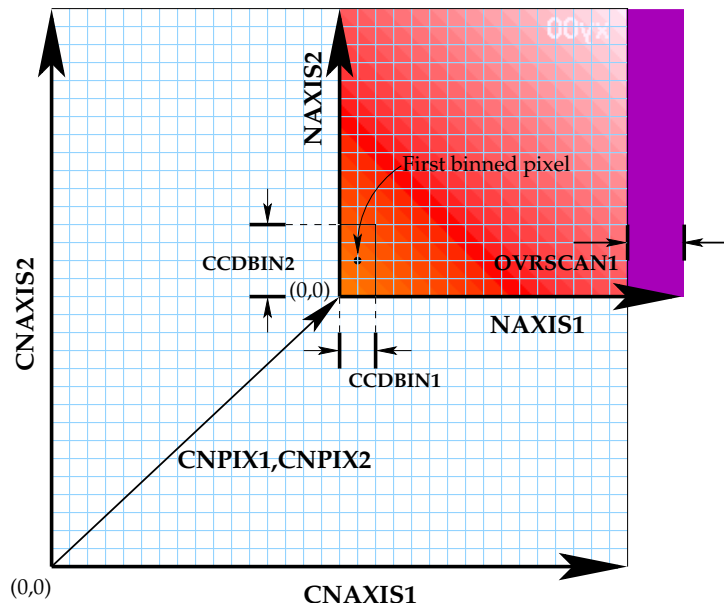


Figure 3: CNAXIS (cell) coordinate system

*Iraf: CCDSUM is a duplication of our 'CCDBIN1 CCDBIN2' expressed as a string. NOAO also recommends using both an old and a new way of expressing this coordinate transform. The old way is in a section notation keyword called CCDSEC, but we do not use it for MEF. The new way is a set of floating point keywords LTV1 LTV2 LTM1\_1 LTM1\_2 LTM2\_1 LTM2\_2. According to NOAO prescription, they do reflect flipping (while our cell coordinate system does not). The LT values also reflect half pixel offset differences from our coordinate systems. Again, see the appendix for more on this.*

### 3.3 OTA coordinates (IMNAXIS1 by IMNAXIS2)

Primary header keywords `IMNAXIS1` `IMNAXIS2` give the full size of the OTA coordinate system (approx 5K x 5K). The units are the same unbinned pixels of `CNAXIS`. Values for `IMNAXIS1` and `IMNAXIS2` can be used as the size of an array to allocate for a display or analysis tool. The transformation from `CNAXIS` to `IMNAXIS` is similar to the one from `NAXIS` to `CNAXIS`. `IMNAXIS`-space is intended to represent the image seen by the detector. Since units are the same, this ends up being the coordinate transform that takes care of any flipping, rotation, and offset translation needed to tile an amplifier region into the image typically viewed by the user.

Extension keywords `IMNPIX1` `IMNPIX2` give the location in `IMNAXIS` space of the origin of the cell coordinate space. An easy way to visualize this is as a vector which points to the outer corner of each cell where the amplifier is located (indicated by triangles in the figure below). This point is `(IMNPIX1,IMNPIX2)` in `IMNAXIS` on the OTA. It is also the origin of a given cell's `CNAXIS`-space.

Any scaling to unbinned pixel units has already been done in the `NAXIS` to `CNAXIS` transformation. This leaves a coordinate transform matrix of unit vectors, of which we only use one specific case for back-side illuminated OTAs:

Keywords `IMTM1.1` `IMTM1.2` must be a unit vector in the OTA coordinate system that points from an amplifier origin along its "fast" direction. For OTA, since the devices are back-side illuminated, this is always -1 0.

Keywords `IMTM2.1` `IMTM2.2` is a unit vector in the OTA coordinate system that points from an amplifier origin along its "slow" direction. This is usually vertical, and not flipped so it is 0 1 for OTA.

Together `IMTM1.1` `IMTM1.2` `IMTM2.1` `IMTM2.2` make up a coordinate transformation matrix between our cell and OTA image coordinate systems. The offset is given by `IMNPIX1` `IMNPIX2`:

$$\begin{aligned}x_{ota} &= (IMTM1.1)x_{cell} + (IMTM1.2)y_{cell} + IMNPIX1 \\y_{ota} &= (IMTM2.1)x_{cell} + (IMTM2.2)y_{cell} + IMNPIX2\end{aligned}$$

Again, for OTA, the `IMTM` values are always  $\begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$  so the equations simplify to:

$$\begin{aligned}x_{ota} &= -x_{cell} + IMNPIX1 \\y_{ota} &= y_{cell} + IMNPIX2\end{aligned}$$

Data analysis software need not implement arbitrary handling of `IMTM` to be used on OTA data. Instead, it is sufficient to verify that the `IMTM` matrix is  $\begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$  and use the simplified equation for efficiency. To handle other common mosaic configurations with amplifiers in the other three corners, also implement cases for  $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ ,  $\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$ , and  $\begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}$ . It is acceptable to leave all other cases unimplemented.

The IMNAXIS coordinate system spans 8x8 cells. The x-axes of the other two coordinate systems point along the negative x-direction of IMNAXIS1. In addition to finding IMNAXIS, IMNPIX, and IMTM keywords in OTA FITS, we provide CELLGAP1 and CELLGAP2 – size in unbinned pixels of gaps between columns and rows of cells. The following figure shows all three coordinate systems:

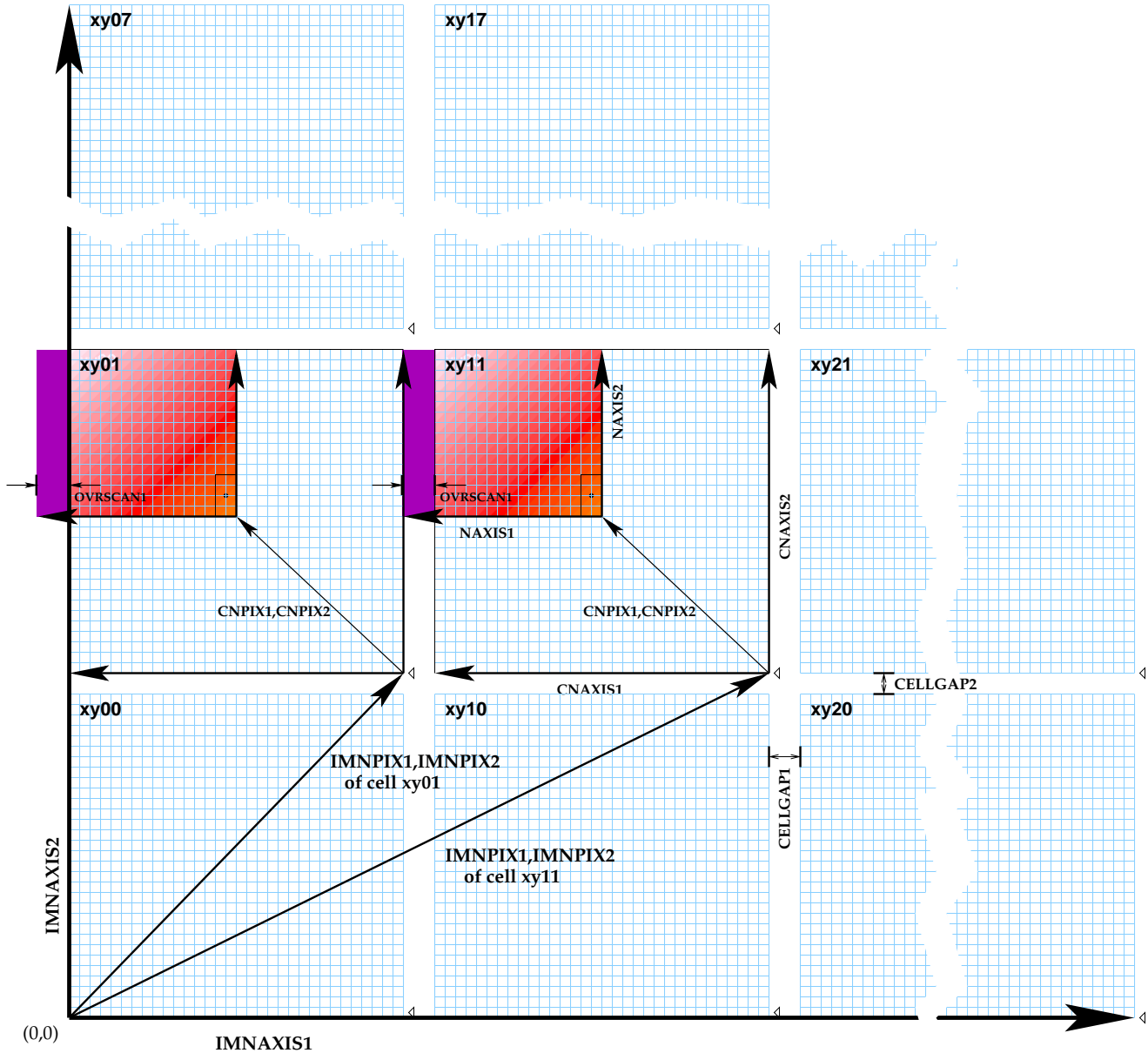


Figure 4: IMNAXIS (image) coordinate system

*Iraf*: Section keywords *DETSIZE DETSEC* and transform coefficients *ATV1 ATV2 ATM1\_1 ATM2\_2* provide the same information. See the appendix for important differences in the definition of the origin.

### 3.4 Coordinate Summary

Coordinate System	Data	Cell	OTA
Origin (0,0)	Corner(*) of 1st pixel read	Corner(*) of pixel by amp.	Corner(*) of cell 00
Origin of prev. system	-	(CNPIX1,CNPIX2)	(IMNPIX1,IMNPIX2)
Flip/rotate	-	-	IMTM1_1, IMTM2_2
Dimensions	NAXIS1 x NAXIS2	CNAXIS1 x CNAXIS2	IMNAXIS1 x IMNAXIS2
Dimensions: meaning	Readout size	Cell size	OTA image size
Units	CCDBIN1 x CCDBIN2 pix	1 x 1 (unbinned pixels)	1 x 1 (unbinned pixels)

(\*) "Corner" means the outer (lower left) corner of the first pixel.

## 4 OTA MEF Data

### 4.1 Science Image Cells

Science cells produce raw images saved in 16-bit signed data with a BZERO value of 32768.0 to map to the range of 0..65535 in BUNITs of 'ADU' as provided by our A/D's. Special diagnostic data may appear in some engineering files that include separate pedestal and video values for each pixel. In this case, NAXIS will equal 3 instead of 2, and NAXIS3 will be equal to 2. The image contains planarized versions of the raw (not pedestal subtracted image) and then the pedestal values.

### 4.2 Video Cells

Video cells can be identified from the CELLMODE keyword in the primary header unit. Video data will have NAXIS set to 3, and NAXIS3  $\geq 2$  (i.e., equal to the number of video frames encoded in the data.) The GRASP video tool **grasp\_vidgrab** produces an alternate format for video which consists of padded, 2D FITS files concatenated together (and typically, shoved over a pipe or socket in that form.) Tools **fitspipe-2d** and **fitspipe-3d** exist to convert back and forth between these formats. Per-frame metadata that is stored in the interleaved FITS headers must be converted to a FITS table once the video stream is converted to the standard 3D form found in the OTA FITS files.

### 4.3 Tables

#### 4.3.1 FITS tables for times, OT shifts, video origins, etc

Metadata about frames of video will be stored in a 2D ASCII FITS table. The actual columns of data stored will vary depending on the actual implementation, but properly configured FITS tables are self-documenting, with entries providing fields widths, a name, a description and units for each column.

The body of an ASCII FITS table is a 2D set of values. Each row is NAXIS1 bytes in length, and there are NAXIS2 rows in the table.

Within each row, the mandatory TFIELDS keyword specifies the number of columns. A series of mandatory keywords then follows describing where each column (field) begins and ends, namely:

- TFORMn - specifies the format of the data for field *n*, the width of the field and (for floating-point values) the number of significant digits after the decimal point.
- TBCOLn - specifies the first character in the row for field *n*.

The numbering convention for FITS tables is that all numbering starts from one, not zero. The first field is number 1, the first character in each row is number 1, etc.

For the TFORM keyword, the FITS standard specifies identifiers for the following data formats:

Field Value	Format
Aw	Character (ASCII text)
Iw	Decimal integer
Fw.d	Single precision real
Ew.d	Single precision real, exponential notation
Dw.d	Double precision real, exponential notation

There are other keywords associated with ASCII FITS tables (one set per column of data) that are optional, but will always be provided in the FITS files from the camera. These provide further documentation of each column as follows:

- TTYPE $n$  - text indicating the type of data in column  $n$ . This is analogous to a column header that provides the name of the column.
- TUNIT $n$  - text indicating the units for the data in column  $n$ .
- TBCOM $n$  - longer plain-text description on the contents of column  $n$ .

Additionally, the VIDEXT keyword will name the 3D image extension containing the video data that this table relates to.

The following is an abbreviated example of the header for a 5-column, 10-row table of values:

```
XTENSION= 'TABLE'
BITPIX = 8 / 8-bit ASCII
NAXIS = 2 / 2D table of values
NAXIS1 = 70 / Characters in each table row
NAXIS2 = 10 / Number of rows in table
PCOUNT = 0 / Random parameters before each array in a group
GCOUNT = 1 / Number of random groups
TFIELDS = 5 / Number of columns in table
TFORM1 = 'D20.6' / FITS table field format
TBCOL1 = 1 / First character of column
TBCOL2 = 21 / First character of column
TFORM3 = 'I2' / FITS table field format
...
TFORM5 = 'D20.6' / FITS table field format
TBCOL5 = 50 / First character of column
EXTNAME = 'video_table_xy00' / Extension name
VIDEXT = 'xy00' / Table relates to frames of video from this exte
TTYPE1 = 'frame_start_time' / Column name
TUNIT1 = 'secs since 1970 (UTC)' / Column data units
TBCOM1 = 'Time at which frame was requested from controller' / Column descripti
TTYPE2 = 'frame_complete_time' / Column name
TUNIT2 = 'secs since 1970 (UTC)' / Column data units
...
TTYPE5 = 'centroid_y' / Column name
TUNIT5 = 'pixels' / Column data units
TBCOM5 = 'Star centroid, y axis' / Column description
END
```



## 5.2 Keyword Inheritance

We will use inheritance for the extension's headers, so the complete header for an extension consists of the primary header augmented by the extension's header. The FITS standard does not allow keywords that are used to determine final pixel values (BZERO and BSCALE) to appear in a header unit with NAXIS=0. These keywords must then appear in each extension with data, and cannot be inherited from a primary header. In creating an OTA MEF it is up to the discretion of the library whether other recommended keywords should appear (once) in the primary header (PHU) or repeatedly in the extensions. For example, 'CCDBIN1' might be common for all cells and can appear in the PHU, or different cells might be binned differently and require their own 'CCDBIN1'. In general it is preferred to put common keywords in the PHU, and repeated keywords in an extensions will override the value found in the PHU.

## 5.3 Keywords We Do Not Use

We will avoid using the following keywords in OTA data, for various reasons described below:

- **EXTVER** The FITS standard has a much better explanation of what the keyword could be for than NOAO's suggestion for use in mosaic data. The FITS standard also says that **EXTNAME** need not be unique within a MEF. In our case, we only create MEFs with unique **EXTNAME** and avoid **EXTVER** completely.
- **DARKTIME** in the primary header unit. This keyword occurs only in extension headers for OTA data, because the dark time can be different for each cell row. The primary header contains **TSP\_\*** keywords, in milliseconds, reported by the controller. If an OTA file has been converted to non-MEF, these must be used to calculate dark time for each row.
- **BYTEORDER** The FITS standard only supports one byte order. Placing a keyword in the data even stating that the data is the "standard" byte order is confusing, because it implies that we could write the other byte order (which we won't). Certainly, software supporting non-standard byte order must assume the correct thing if this keyword is not there.
- **UNSIGN** When recording 16-bit data, the FITS standard only supports signed numbers. Even setting **UNSIGN=F** is confusing since it implies we could at some point use **UNSIGN=T**, which we won't. Certainly, software supporting this non-standard keyword must assume the correct thing if the keyword is not there.
- **CHECKSUM** and **DATASUM**. At the moment, we form a simple unix md5sum of the entire file before it is compressed and written to disk. Insertion of the **CHECKSUM** and **DATASUM** keywords (as sometimes done by **funpack**, even with the **-C** option) breaks this. We modified **fpack/funpack** to force the **-C** option to completely disable checksums (as was probably intended by the authors.) Now we can use any kind of compression, and the external md5sum provides an easy to understand end-to-end check.

## 5.4 Mandatory OTA MEF Keywords

Without these this is not an OTA MEF:

### 5.4.1 Required Primary Header Keywords

```
SIMPLE = T / Standard FITS
BITPIX = 16 / (not applicable to PHU)
NAXIS = 0 / Number of axes
EXTEND = T / File contains extensions
NEXTEND = 65 / Number of extensions
NAMPS = 64 / Number of amplifiers in the detector
FILENAME= 'o4690g0086o34' / Base filename at acquisition
DATE = '2008-08-12T09:30:39' / UTC Date of file creation
PIXSIZE1= 12.000 / Pixel size for axis 1 (microns)
```



```

TSP_SHOP=          0 / Milliseconds from last clean to shutter open
TSP_SHCL=          0 / Milliseconds from last clean to shutter close
TSP_RD00=        36077 / Milliseconds darktime for buffer (cell row) 0
TSP_RD01=        36927 / Milliseconds darktime for buffer (cell row) 1
TSP_RD02=        37778 / Milliseconds darktime for buffer (cell row) 2
TSP_RD03=        38629 / Milliseconds darktime for buffer (cell row) 3
TSP_RD04=        39479 / Milliseconds darktime for buffer (cell row) 4
TSP_RD05=        40330 / Milliseconds darktime for buffer (cell row) 5
TSP_RD06=        41180 / Milliseconds darktime for buffer (cell row) 6
TSP_RD07=        42031 / Milliseconds darktime for buffer (cell row) 7
TSP_NOW =         37646 / Milliseconds from last clean until save
CLV_PPG4= 'ecbb:cbb2:bb2e:65d8:5d97:38ba:6622:3154' / Coded parallel timings
CLV_PG3 = '340e:40e0:1c03:c070:06c1:0417:649b:0136' / Coded serial timings
CLV_PG4 = '1038:8010:0104:00b0:07c2:0000:3732:08a2' / Coded SW/RST/VCLAMP/SAMP
CLV_ADC = '1500' / Coded ADC parameters
CLV_TRIG=          2 / Cross-trigger phase delay (x 10nsec)
CLV_PRES=          3 / Serial prescan, (skipped in addition to CNPIX1
CLV_PIPE=          1 / Clocking pattern pipeline (in addition to ADC)
CLV_PXTP=          1 / OT Pixel Type 1 or 2 (or 0 for normal CCD)
ADC_CONF= '0c8:0c8:0c8:0c8:0c8:0c8:0c8:0c8' / ADC configuration
ADC_MUX = '0c0:0c0:0c0:0c0:0c0:0c0:0c0:0c0' / ADC MUX configuration
ADC_PGAR= '019:019:019:019:019:019:019:019' / Red ADC pga (adcgain 1.5)
ADC_PGAG= '000:000:000:000:000:000:000:000' / Grn ADC pga (adcgain val)
ADC_PGAB= '000:000:000:000:000:000:000:000' / Blu ADC pga (adcgain val)
ADC_OFSR= '000:000:000:000:000:000:000:000' / Red ADC offset
ADC_OFSG= '000:000:000:000:000:000:000:000' / Grn ADC offset
ADC_OFSB= '000:000:000:000:000:000:000:000' / Blu ADC offset
DAC_INIT=         1600 / DAC internal vref offset voltage (millivolts)
DAC_S_L =        -1000 / Serial and summing well LO (millivolts)
DAC_S_H =         7000 / Serial and summing well HI (millivolts)
DAC_P1_L=         500 / Parallel 1 LO (millivolts)
DAC_P1_H=        8500 / Parallel 1 HI (millivolts)
DAC_P2_L=        1850 / Parallel 1 LO (millivolts)
DAC_P2_H=        9850 / Parallel 1 HI (millivolts)
DAC_P3_L=         500 / Parallel 1 LO (millivolts)
DAC_P3_H=        8500 / Parallel 1 HI (millivolts)
DAC_P4_L=         500 / Parallel 1 LO (millivolts)
DAC_P4_H=        8500 / Parallel 1 HI (millivolts)
DAC_PSL =         500 / Parallel Standby LO (millivolts)
DAC_PSH =        8500 / Parallel Standby HI (millivolts)
DAC_RG_L=       -2000 / Reset Gate LO (millivolts)
DAC_RG_H=       10000 / Reset Gate HI (millivolts)
DAC_SO =         4100 / 1st Stage Source (millivolts)
DAC_DR =         6165 / Drain Bias (millivolts)
DAC_VSS =        -500 / Logic Low Supply (millivolts)
DAC_VDDL=        6000 / Logic Drain Supply LO (millivolts)
DAC_VDDH=        8000 / Logic Drain Supply HI (millivolts)
DAC_OG1 =         0 / Output Gate Bias (millivolts)
DAC_RD =         6100 / Reset Drain Bias (millivolts)
DAC_SUB =       -2335 / Substrate (millivolts)
DAC_SCP =       10000 / Scupper Bias (millivolts)
DAC_LREF=       -1000 / Logic Reference Ground (millivolts)
DAC_CAL0= '-399 -394 -381 -341 -376 -361 -366 -328' / Pedestal cal (mV)
DAC_CAL1= '-401 -381 -376 -325 -376 -360 -357 -318' / Pedestal cal (mV)
DAC_CAL2= '-398 -380 -376 -323 -362 -338 -345 -315' / Pedestal cal (mV)
DAC_CAL3= '-400 -384 -367 -311 -364 -341 -347 -320' / Pedestal cal (mV)
DAC_CAL4= '-401 -383 -369 -321 -355 -342 -346 -312' / Pedestal cal (mV)
DAC_CAL5= '-386 -376 -362 -301 -338 -322 -335 -301' / Pedestal cal (mV)
DAC_CAL6= '-373 -362 -351 -289 -328 -315 -325 -293' / Pedestal cal (mV)
DAC_CAL7= '-357 -349 -344 -274 -316 -298 -306 -272' / Pedestal cal (mV)

```

## 5.5 Recommended OTA MEF Keywords

These should be written, even if with nominal values.

```

==== Primary Header ====
MJD-OBS =          54690.39586 / Modified Julian date at start of observation
IMAGESWV= 'grasp_save_ota v1.13 SVN 2764 (Jul 29 2008)' / Image creation softwa
ORIGIN = 'PS1' / IFA Pan-STARRS 1
INSTRUME= 'gpcl' / Instrument Name
DETECTOR= 'CCID58-2-04a1' / Science Detector
DETTEM =          -76.5 / Detector temperature (Celsius)
FPPPOS = 'xy34' / Position of detector in focal plane mosaic
OBSTYPE = 'OBJECT' / Observation/Exposure type
EXPTIME =          30.000 / Actual exposure time as measured by shutter dr
EXPREQ =          30.000 / Exposure time requested (seconds)

```

## 5.6 Other Random Keywords

**Remove Other Random Keywords section completely after promoting any keywords of value to Recommended or Mandatory. (TBD)**

The following hasn't been looked at in a while. These just serve as reminders of what various other headers look like, or what we had thought of doing for accelerators (but this is not implemented.)

What follows is a smorgasbord of keywords and values gleaned from various detectors and control programs. Please use these keywords if possible, and please notify me if you disagree with their use, you want to change their use, or you want to create new keywords.

Some of these keywords violate the "keyword family" notion above, for example the environmental information. These will probably be changed in the future to agree with the "keyword family" organization as specific data values and needs become known.

Please note that these may change, so keep checking against the web site (or ask!) to try to stay consistent. Because we are creating new devices, we have a unique opportunity to create and maintain a great deal of consistency and we should not squander it.

```
DATASEC = '[1:512,1:489]' / Image area of each cell
BIASSEC = '[513:532,1:489]' / Overscan area of each cell
CELL36 = 'SCIENCE, offset 17455680' / status and byte offset
CELL37 = 'DEAD, offset -1' / status and byte offset
CELL53 = 'VIDEO, offset 25018560' / status and byte offset
CELL77 = 'UNTESTED, offset -1' / status and byte offset
GTABLE = 'TABLE, offset 41895360' / status and byte offset

==== For each image cell ====
CCDBIN1 = 1 / On-chip column binning (fast=col dir)
CCDBIN2 = 1 / On-chip row binning (slow=row dir)
OVRSCAN1= 32 / Number of bias over-scan pixels in each row
PRESCAN1= 0 / Number of pre-scan pixels in each row
OVRSCAN2= 0 / Number of bias over-scan pixels in each col
PRESCAN2= 0 / Number of pre-scan pixels in each col
GAIN = 1.450 / Nominal gain (e-/ADU)
BIAS = 1000.0 / Nominal bias level (ADU)
MAXLIN = 63500 / Maximum linear range (ADU)
BIASSEC = '[513:520,1:489]' / Overscan area of each cell
CCDSEC = '[2665:3176,1002:1491]' / Image area relative to full chip

*****

SIMPLE = T / Standard FITS
BITPIX = 16 / Bits per pixel (not applicable)
NAXIS = 0 / No image data with primary header
EXTEND = T / File contains extensions
NEXTEND = 65 / Number of extensions

COMMENT ----- General Information -----
FILENAME= '20050311T101417.135' / Base filename at acquisition
PATHNAME= '/d0/20050311' / Original directory name at acquisition
DATE = '2005-03-11T10:14:17' / UTC Date of file creation
IMAGESWV= 'PS GPCCom v3.50 (Aug 08 2004)' / Image creation software version
ORIGIN = 'Institute for Astronomy' / University of Hawaii
TELESCOP= 'HK-PS1' / Pan-STARRS telescope
TELELAT = 19.8261 / Latitude (degrees N)
TELELONG= -155.4708 / Longitude (degrees E)
TELEALT = 3123.2 / Telescope altitude (m)
FRATIO = 4.4 / Focal ratio of telescope
INSTRUME= 'PS_GPC1' / Instrument name
DETECTOR= 'OTA35' / OTA mosaic name (indicates position)
OTAINDEX= 35 / OTA mosaic index
ROTANGLE= 180.0 / Cass Rotation angle (deg)
TELFOCUS= 5210 / Telescope focus
TELFOCTM= 'focus135.dat' / Focus vs time save file (not sure about this)
PREFLASH= 0 / Preflash, seconds
OBSERVER= 'Kaiser-Tyson' / Observers
PROPID = 'CTIO 221' / Telescope proposal ID

COMMENT ----- Time Information -----
```

```

TIMESYS = 'UTC' / Time System for DATExxxx
TSYSOBS = 'UTC' / Time System for TIMExxxx
DATE-OBS= '2005-03-11' / Date at start of exposure
TIME-OBS= '10:14:17.667' / UTC time at start of exposure
EXPTIME = 1.0312 / Integration time (sec)
DARKTIME= 1.0511 / Duration between erase and readout (sec)

```

```

COMMENT ----- Pointing Information -----
RA = 103.125292 / Right Ascension (deg)
DEC = 0.460892 / Declination (deg)
HA = -21.290625 / Hour angle at start (deg)
ST = 5.459264 / Sidereal time at start (hr)
ZD = 28.424078 / Zenith distance (deg)
EQUINOX = 2000.000 / Equinox of RA and DEC
EPOCH = 2000.000 / Equinox of RA and DEC
AIRMASS = 1.137 / Airmass at start
RASTRNG = '06:52:30.07' / Actual Right Ascension (incl offset)
DECSTRNG= '00:27:39.21' / Actual Declination (incl offset)
HASTRNG = '-01:25:09.75' / Hour angle at start
STSTRNG = '05:27:33.35' / Sidereal time at start
ZDSTRNG = '28:25:26.68' / Zenith distance (deg:mm:ss)

```

```

COMMENT ----- Environmental Information -----
ENVUTC = '2004-02-20 05:51:01' / UTC timestamp for environmental data
TTDOME = 4.06 / Temp: Top of dome
TTTUBE = 4.40 / Temp: Top of telescope tube
TPMIRR = 3.96 / Temp: Primary mirror
TPIER1 = 8.79 / Temp: Telescope pier (1st floor)
TVGALL = 5.44 / Temp: Visitor's gallery
TTBASE = 4.06 / Temp: Telescope base
TDMOIL = 15.88 / Temp: Dome drive oil
TPBOIL = 8.64 / Temp: Polar bearing oil
TDMWAL = 1.52 / Temp: Observing floor wall
TGLYCL = -5.13 / Temp: Glycol temperature
TOUTSD = 4.52 / Outside temperature
HOUTSD = 8.6 / Outside humidity
TINSID = 4.30 / Inside temperature
HINSID = 10.2 / Inside humidity
TINTUBE = 3.69 / Inside tube temperature
HINTUBE = 12.1 / Inside tube humidity
WINDSPD = 15.0 / Wind speed (MPH)
WINDDIR = 40.0 / Wind direction (deg)
RAIN = 0 / Rain?
WINDHEAT= 0 / Wind heater on?
MASTHEAT= 0 / Mast heater on?

```

```

COMMENT ----- Observation Information -----
FILTER = 'r' / Filter description
FILTERID= 'r ID 007' / Specific filter ID number
OBSTYPE = 'DARK' / Observation/exposure type
PHOTOZP = 25.5 / Photometric zeropoint mag giving 1e/sec

```

```

COMMENT ----- OTA Information -----
OTADEV = 'CCID45-a' / Identification of device type
OTAID = 161 / OTA device ID (unique identifier)
PIXSCAL1= 0.301 / Arcseconds per pixel in fast=col dir
PIXSCAL2= 0.301 / Arcseconds per pixel in slow=row dir
CCDBIN1 = 1 / On-chip column binning (fast=col dir)
CCDBIN2 = 1 / On-chip row binning (slow=row dir)
NBIASPIX= 64 / Number of bias over-scan pixels in each cell
DATASEC = '[1:512,1:489]' / Image area of each cell
BIASSEC = '[513:532,1:489]' / Overscan area of each cell
SATURATE= 65535 / ADU value indicating saturated pixels
PIXSIZE1= 12.000 / microns per pixel in fast=col dir
PIXSIZE2= 12.000 / microns per pixel in slow=row dir

```

```

COMMENT ----- OTA Housekeeping Information -----
HIVOLT = 35.1 / OTA electronics +36 V voltage
MIDVPOS = 16.4 / OTA electronics +15 V voltage
MIDVNEG = -16.1 / OTA electronics -15 V voltage
LOWV = 5.03 / OTA electronics +5 V voltage
DETTEM = -78.22 / Detector temperature (C)
SETTEM = -77.93 / OTA set temperature
ELECTEM = 27.02 / OTA electronics temperature
HEATPERC= 6.1 / OTA heater voltage percentage
CRYOTEM = -175.2 / Cryostat head temperature (C)
CRYOVAC = 1.4E-3 / Cryostat pressure (mtorr)
CONTROLR= 'IOTA 14_1' / Controller ID_Channel
CONTRVER= '20040522T141620' / Microcode build date-time
CONTRSWV= '20040612T101214' / Acquisition software build date-time

```

```
XTALKFIL= 'IOTA14_OTA35.xtk' / Cross-talk correction factor filename
(and a whole lot more...)
```

```
COMMENT ----- Status Info and Accelerators -----
CELL00 = 'SCIENCE, offset 51840' / status and byte offset
...
CELL36 = 'SCIENCE, offset 17455680' / status and byte offset
CELL37 = 'DEAD, offset -1' / status and byte offset
...
CELL53 = 'VIDEO, offset 25018560' / status and byte offset
...
CELL77 = 'UNTESTED, offset -1' / status and byte offset
GTABLE = 'TABLE, offset 41895360' / status and byte offset
```

**Sidik: I'm not sure I like this proposed accelerator format, exactly as shown here. (TBD)**

```
==== For each image cell ====
```

```
XTENSION= 'IMAGE' / Image extension
BITPIX = 16 / 16bit 2s complement 0..65535 (BZERO=32768)
NAXIS = 2 / Number of axes
NAXIS1 = 512 / Number of pixel columns
NAXIS2 = 489 / Number of pixels rows
PCOUNT = 0 / No 'random' parameters
GCOUNT = 1 / Only one group
INHERIT = T / Must inherit global keywords
EXTNAME = 'CELL52' / Extension name
```

```
COMMENT ----- Cell Information -----
CELLINDX= 52 / Which cell in the OTA is this?
BSCALE = 1.0000000E+00 / PHYSICAL = BSCALE * DATA + BZERO
BZERO = 32768.
GAIN = 1.450 / Nominal gain (e-/ADU)
BIAS = 1000.0 / Nominal bias level (ADU)
MAXLIN = 63500 / Maximum linear range (ADU)
CNPIX1 = 2665 / x offset of data start on OTA
CNPIX2 = 1002 / y offset of data start on OTA
CCDSEC = '[2665:3176,1002:1491]' / Image area relative to full chip
COMMENT
COMMENT the actual start time at location x,y (cell coords) is
COMMENT MJD-OBS + STARTT00 + STARTT01*(x/NAXIS1) + STARTT10*(y/NAXIS2)
COMMENT
STARTT00= 0.131 / MJD-OBS equation zero point (sec)
STARTT01= 0.012 / MJD-OBS equation x slope (sec)
STARTT10= -0.001 / MJD-OBS equation y slope (sec)
COMMENT
COMMENT the actual exposure time at location x,y (cell coords) is
COMMENT EXPTIME + EXPT00 + EXPT01*(x/NAXIS1) + EXPT10*(y/NAXIS2)
COMMENT
EXPT00 = 0.089 / EXPTIME equation zero point (sec)
EXPT01 = -0.037 / EXPTIME equation x slope (sec)
EXPT10 = 0.014 / EXPTIME equation y slope (sec)
```

```
==== For each video cell ====
```

```
XTENSION= 'IMAGE' / Image extension
BITPIX = 16 / 16bit 2s complement 0..65535 (BZERO=32768)
NAXIS = 3 / Number of axes
NAXIS1 = 24 / Number of pixel columns
NAXIS2 = 24 / Number of pixels rows
NAXIS3 = 172 / Number of frames
PCOUNT = 0 / No 'random' parameters
GCOUNT = 1 / Only one group
INHERIT = T / Must inherit global keywords
EXTNAME = 'CELL36' / Extension name
```

```
COMMENT ----- Cell Information -----
CELLINDX= 36 / Which cell in the OTA is this?
BSCALE = 1.0000000E+00 / PHYSICAL = BSCALE * DATA + BZERO
BZERO = 32768.
GAIN = 1.450 / Nominal gain (e-/ADU)
BIAS = 1000.0 / Nominal bias level (ADU)
MAXLIN = 63500 / Maximum linear range (ADU)
PIXSCAL1= 0.603 / Arcseconds per pixel in fast dir
```

```
PIXSCAL2=          0.603 / Arcseconds per pixel in slow dir
CCDBIN1 =           2 / On-chip column binning (fast dir)
CCDBIN2 =           2 / On-chip row binning (slow dir)
NBIASPIX=          64 / Number of bias over-scan pixels in each cell
DATASEC = '[1:512,1:489]' / Image area of each cell
BIASSEC = '[513:532,1:489]' / Overscan area of each cell
CCDSEC = '[1:4096,1:1032]' / Image area relative to full chip
CNPIX1 =           3198 / x offset of data start on OTA
CNPIX2 =           1503 / y offset of data start on OTA
```

## 6 Validation Tool

A validation tool should be provided which checks basic FITS structure and data, and also keyword compliance.

**Look into combining parts of fhtool that I wrote at CFHT with fitsverify that Andrey says is available from NASA. (TBD)**

**Also download, check, and reference the latest FITS Standard (3.0) which just came out. (TBD) [http://fits.gsfc.nasa.gov/fits\\_stan](http://fits.gsfc.nasa.gov/fits_stan)**