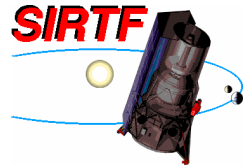


674-SO-43, Version 4.8,  
SSC-PD-4046



SIRTF Science Center

Downlink Segment

Subsystem Design Specification

AOT Products Subsystem:  
FOWLINEARIZE

24 March 2003

California Institute of Technology  
SIRTF Science Center



National Aeronautics and  
Space Administration



Jet Propulsion Laboratory  
California Institute of Technology  
Pasadena, California

**THIS IS A PRELIMINARY DOCUMENT, the module described here may  
or may not be utilized in the final pipelines as described.**

THIS PAGE IS INTENTIONALLY LEFT BLANK

**THIS IS A PRELIMINARY DOCUMENT, the module described here may or may not be utilized in the final pipelines as described.**

SIRTF Science Center

# Subsystem Design Specification

Prepared by:

Frank Masci

Concurred by:

Bill Reach      Dave Shupe

Approved by:

Mehrdad Moshir

Concurred by:

Bill Green

**THIS IS A PRELIMINARY DOCUMENT, the module described here may or may not be utilized in the final pipelines as described.**

## 1 Revision History

Version	Description	Date
1.0	Initial version	March 29, 2001
2.0	Read in an uncertainty image, perform error propagation and output a corresponding uncertainty image.	May 21, 2001
3.0	<ol style="list-style-type: none"> <li>1. Changed option to set a d-mask bit ONLY if a pixel could NOT be linearized.</li> <li>2. Implemented cubic non-linearity model (optional).</li> </ol>	September 18, 2001
4.0	<ol style="list-style-type: none"> <li>1. For the quadratic model, re-arranged the <math>DN_{lin}</math> vs. <math>DN_{obs}</math> equation to avoid possible numerical instability when the non-linearity is very small.</li> <li>2. For the quadratic model, set pixels to the maximum value that can be linearized if their observed counts lie above the non-linearity curve.</li> </ol>	February 25, 2002
4.1	Modified quadratic and cubic linearization solutions to account for a non-zero delay time between pixel resets and first readouts. The requested modifications were outlined in Myugshin Im's revision document (2/01/02).	August 5, 2002
4.8	Implemented ability to read in FITS image cubes as will apply to sub-array image data. Removed the setting NAXIS3=1 for single plane output images.	March 24, 2003

## 2 Table of Contents

<b>1 REVISION HISTORY.....</b>	<b>IV</b>
<b>2 TABLE OF CONTENTS .....</b>	<b>V</b>
<b>3 LIST OF FIGURES .....</b>	<b>VIII</b>
<b>4 LIST OF TABLES .....</b>	<b>VIII</b>
<b>1. INTRODUCTION.....</b>	<b>10</b>
1.1. Purpose and Scope.....	10
1.2. Document Organization.....	10
1.3. Relationship to Other Documents .....	10
1.4. Change Procedure .....	11
<b>2. OVERVIEW.....</b>	<b>11</b>
2.1. FOWLINEARIZE Requirements .....	11
2.2. Applicable Documents .....	12
2.3. Version History.....	12
2.3.1. Version 1.0.....	12
2.3.2. Version 2.0.....	12
2.3.3. Version 3.0.....	13
2.3.4. Version 4.0.....	13
2.3.5. Version 4.1.....	13
2.3.6. Version 4.8.....	13
2.4. Liens. 13	
<b>3. INPUT .....</b>	<b>13</b>
3.1. FOWLINEARIZE Input .....	13
3.1.1. FOWLINEARIZE NAMELIST Input .....	14
3.1.2. FOWLINEARIZE Command-Line Input .....	16

**THIS IS A PRELIMINARY DOCUMENT, the module described here may or may not be utilized in the final pipelines as described.**

3.1.3. FOWLINEARIZE FITS Input .....	16
<b>4. PROCESSING.....</b>	<b>18</b>
4.1. FOWLINEARIZE Processing .....	18
2.4. FOWLINEARIZE Processing Phases.....	18
4.1.1. FOWLINEARIZE Initialization .....	18
4.1.2. FITS-Image Data Input .....	20
4.1.3. Model Coefficients Input .....	20
4.1.4. Optional Uncertainty and Mask Image Inputs.....	21
4.1.5. Linearity Computation.....	21
4.1.6. FITS-Image Output .....	23
4.1.7. Termination.....	23
<b>5. ALGORITHM.....</b>	<b>23</b>
5.1. Algorithm Specifics .....	23
5.2. Default Bit-Mask Settings .....	24
5.3 Assumptions and Requirements .....	25
<b>6. OUTPUT.....</b>	<b>26</b>
6.1. FOWLINEARIZE Output .....	26
6.1.1 FOWLINEARIZE FITS Output.....	26
6.1.2 FOWLINEARIZE Log-File Output .....	27
<b>7. TESTING.....</b>	<b>27</b>
<b>8. USAGE EXAMPLES .....</b>	<b>28</b>
<b>9. GLOSSARY .....</b>	<b>28</b>
<b>10.APPENDIX I (QUADRATIC FOWLINEARIZE).....</b>	<b>29</b>
<b>11.APPENDIX II (QUADRATIC FOWLINEARIZE UNCERTAINTIES).....</b>	<b>31</b>
<b>12.APPENDIX III (CUBIC FOWLINEARIZE).....</b>	<b>32</b>
<b>13.APPENDIX IV (CUBIC FOWLINEARIZE UNCERTAINTIES).....</b>	<b>36</b>
<b>14. ACKNOWLEDGEMENTS .....</b>	<b>36</b>

**THIS IS A PRELIMINARY DOCUMENT, the module described here may or may not be utilized in the final pipelines as described.**

**THIS IS A PRELIMINARY DOCUMENT, the module described here may or may not be utilized in the final pipelines as described.**

## List of Figures

Figure 1. FOWLINEARIZE data and processing flow.....18

## List of Tables

Table 1. Namelist File .....	13
Table 2. Command Line Options .....	15

## 1. Introduction

### 1.1. Purpose and Scope

The Subsystem Design Specification is a document that describes the basic requirements, assumptions, definitions, software-design details and necessary interfaces for each subsystem. The document will be used to trace the incremental development of each subsystem and also to allow trace-back of levied requirements; this document should have sufficient detail to allow future modification or maintenance of the software by developers other than the original developers. This document is an evolving document as changes may occur in the course of science instrument hardware design and maturity of operational procedures. This document is not intended to repeat sections or chapters from other Project documents; when appropriate, references to proper sections of primary reference documents will be made.

### 1.2. Document Organization

This document is organized along the major themes of Requirements; Assumptions; Operational Concept; Functional Descriptions; Functional Dependencies; Input; Output; Other S/S Interfaces; Algorithm Descriptions (when applicable); and Major Liens.

The material contained in this document represent the current understanding of the capabilities of the major SIRTf systems. Areas that require further analysis are noted by TBD (To Be Determined) or TBR (To Be Resolved). TBD indicates missing data that are not yet available. TBR indicates preliminary data that are not firmly established and are subject to change.

### 1.3. Relationship to Other Documents

The requirements on the operation of SIRTf flow down from the Science Requirements Document (674-SN-100) and the Facility Requirements Document (674-FE-100). The Science Operations System is governed by the SOS Requirements Document (674-SO-100). The current document is also cognizant of the requirements that appear in the Observatory Performance and Interface Control Document (674-SEIT-100) as well as the Flight Ground Interface Control Document (674-FE-101). This document is also affected by the FOS/SOS Interface Control Document (674-FE-102) that governs interfaces between the Flight Operations System and the Science Operations System. Related Software Interface Specifications (SIS) will be as indicated in Section 2.2 of this document.

## 1.4. Change Procedure

This document is a level 4 document according to the SIRTf Project Documentation Plan (674-FE-103). Changes to this document after approval require the approval of the SOS Change Board (TBD). The process for change control is described in the SOS Configuration Management Plan.

## 2. Overview

The FOWLINEARIZE program reads image data from a standard FITS file and corrects the data for detector non-linearity using either a quadratic or cubic non-linearity model. The software has the ability to read in FITS image cubes with the same algorithm applied to each plane. This software is specifically designed to linearize image data acquired with SIRTf's IRAC instrument and accounts for IRAC's use of Fowler sampling. The non-linearity (calibration) model is assumed to be derived by the LINCAL program.

The primary product of this software is a 32-bit/pixel image of "linearized" pixel values in units of data number (DN). Optionally, the software can read in an uncertainty image and propagate uncertainties to generate an output uncertainty image for the linearized data. The software also optionally reads in p-mask, d-mask and c-mask images to handle "fatal" pixels where the non-linearity correction cannot be applied. FOWLINEARIZE is written in ANSI/ISO C.

### 2.1. FOWLINEARIZE Requirements

FOWLINEARIZE is initiated by a startup script under the control of the pipeline executive and does its required functions for a given DCE image or pre-processed DCE image; this involves performing the following tasks.

- A.) Retrieve the command line parameters passed by the start up script and use them to run the program.
- B.) Read in as input a standard FITS image, a non-linearity model image cube, uncertainty and mask images.
- C.) Produce as primary output a new FITS image with "linearized" pixel values.
- D.) Provide exit codes to the pipeline executive and also provides logon and logoff messages identifying the version number and write any error messages to the standard output devices.

E.) Produce a processing summary.

## 2.2. Applicable Documents

The following documents are relevant to the FOWLINEARIZE program of the AOT PRODUCTS Subsystems.

- A.) The SOS Requirements Document
- B.) The SOS Downlink Requirements Document
- C.) The SOS Downlink Software Development Guidelines
- D.) The following Software Interface Specifications (SIS)
  - SOSDL-SIS-PD-3000 (real\*4 DCE data output)
  - SOSDL-SIS-PD-3001 (p- and d-mask inputs)
  - SOSDL-SIS-CL-3001 (calibration inputs)
  - SOSDL-SIS-CL-3003 (c-mask image input)

## 2.3. Version History

### 2.3.1. Version 1.0

Initial version created on March 29, 2001.

### 2.3.2. Version 2.0

Version 2.0 includes the option of reading in an uncertainty image to produce corresponding uncertainties in linearized pixel values using formal error propagation.

### 2.3.3. Version 3.0

Version 3.0 changes the option of updating the d-mask if a pixel was linearized to only if a pixel could not be linearized. This version also implements a cubic non-linearity model.

### 2.3.4. Version 4.0

Version 4.0 includes a refinement to the  $DN_{lin}$  vs.  $DN_{obs}$  equation for the quadratic case (Eqn. 9, Appendix I) to avoid possible cases of numerical instability when the non-linearity is very small. Also in the quadratic case, if pixels have raw counts which lie above the maximum that can be linearized according to the non-linearity model, then they are forced to the maximum value.

### 2.3.5. Version 4.1

Version 4.1 implements a modification to the quadratic and cubic linearization solutions to account for a non-zero delay time between resets and first readouts. The requested modifications were outlined in Myugshin Im's revision document (2/01/02).

### 2.3.6. Version 4.8

This version implements the ability to read in FITS image cubes as will apply to sub-array image data. Removed the setting NAXIS3=1 for single plane output images.

## 2.4. Liens

No major liens have been identified.

## 3. Input

### 3.1. FOWLINEARIZE Input

FOWLINEARIZE takes all of its input from either the command line or namelist file, which is set up by the startup script that is controlled by the pipeline executive or standalone. If the namelist is not specified, then all required inputs are expected from the command line. If both namelist and command-line inputs are specified, then the command-line inputs override the namelist values. Prior to

reading namelist and/or command-line parameters, default values for the relevant parameters are assigned.

### 3.1.1. FOWLINEARIZE NAMELIST Input

FOWLINEARIZE reads the NAMELIST file whose name is passed to it by start-up script. The name of the NAMELIST is FOWLCORRIN. The parameters that can be defined in the NAMELIST are listed in Table 1.

Namelist variable	Description	Dim.	Type	Units	Default
FITS_Image_Filename1	Required input FITS-image filename containing un-linearized data.	256	C	-	Null
FITS_Image_Filename2	Required input FITS-image filename containing non-linearity model coefficients and their uncertainties.	256	C	-	Null
FITS_Image_Filename3	Optional input FITS-image filename containing uncertainties.	256	C	-	Null
FITS_Image_PMask_Filename	Optional p-mask FITS-image	256	C	-	Null
FITS_Image_DMask_Filename	Optional d-mask FITS-image	256	C	-	Null
FITS_Image_CMask_Filename	Optional c-mask FITS-image	256	C	-	Null
FITS_Out_Filename1	Required output FITS-image filename containing linearized data.	256	C	-	Null

FITS_Out_Filename2	Optional output FITS-image filename containing uncertainties in linearized data.	256	C	-	Null
Model_Flag	Required non-linearity model flag: 1=Quadratic, 2=Cubic	1	I*1	-	-
Clock_Readout	Optional sampling readout time: either 200 ms (full array), or 10 ms (sub-array).	1	R*4	msec	200 msec
PMaskFatal	Fatal PMask data bits	1	I*2	-	8192
DMaskFatal	Fatal DMask data bits	1	I*2	-	512
CMaskFatal	Fatal CMask data bits	1	I*2	-	512
DMaskNotLin	DMask data bit flag indicating linearization <i>was not</i> performed	1	I*2	-	4096
DMaskNotLinSat	DMask data bit flag indicating data is saturated according to linearity model	1	I*2	-	8192
Log_Filename	Optional output log filename	256	C	-	Stdout
Ancillary_File_Path	Pathname where supporting source files are installed.	256	C	-	./

**Table 1. Namelist file**

The following is an example of the contents of a FOWLCORRIN NAMELIST file that might be used, where the values specified are not necessarily realistic.

```
&FOWLCORRIN
```

**THIS IS A PRELIMINARY DOCUMENT, the module described here may or may not be utilized in the final pipelines as described.**

```
Comment = 'Generic namelist file for fowlinearize, default values.',  
Ancillary_File_Path = '../fowlinearize_v2',  
FITS_Image_Filename1 = './testing/chan4_tx4_flat_1.fits',  
FITS_Image_Filename2 = './testing/lincal_with_error.fits',  
FITS_Image_Filename3 = './testing/errormodel.fits',  
FITS_Image_PMask_Filename = './testing/pmask_ch1.fits',  
FITS_Image_DMask_Filename = './testing/dmask1.fits',  
FITS_Image_CMask_Filename = './testing/cmask.fits',  
FITS_Out_Filename1 = './testing/linearized.fits',  
FITS_Out_Filename2 = './testing/error_linearized.fits',  
Comment = 'Non-linearity model: 1 = Quadratic, 2 = Cubic',  
Model_Flag = 1,  
Comment = 'Sampling read-out time in msec, Default=200 msec (full-array)',  
Clock_Readout = 200,  
Log_Filename = 'stdout',  
PMaskFatal = 8192,  
DMaskFatal = 512,  
CMaskFatal = 256,  
Comment = 'Dmask bit indicating pixel was not linearized',  
DMaskNotLin = 4096,  
Comment = 'Dmask bit indicating pixel is saturated in linearity model and  
beyond correctable non-linearity',  
DMaskNotLinSat = 8192,  
&END
```

### 3.1.2. FOWLINEARIZE Command-Line Input

Alternatively, all inputs can be specified via command line, in which case, a namelist file is not needed. Or, inputs can be provided with a hybrid of both namelist and command-line mechanisms, with the latter overriding the former. Table 2 lists the available command-line options associated with their namelist-variable counterparts, as well as other options for specifying the namelist-file name and making the standard output more verbose.

### 3.1.3. FOWLINEARIZE FITS Input

FOWLINEARIZE uses the FITSIO library routines to read in the FITS-formatted input data file. The routines used are: `fits_open_file`, `fits_read_keys_lng`, `fits_read_keys_dbl`, `fits_read_img`, and `fits_close_file`.

<b>Command-line option</b>	<b>Variable</b>
-n	Namelist_Filename
-i1	FITS_Image_Filename1
-i2	FITS_Image_Filename2
-i3	FITS_Image_Filename3
-ip	FITS_Image_PMask_Filename
-id	FITS_Image_DMask_Filename
-ic	FITS_Image_CMask_Filename
-o1	FITS_Out_Filename1
-o2	FITS_Out_Filename2
-m	Model_Flag
-c	Clock_Readout
-l	Log_Filename
-a	Ancillary_File_Path
-fp	PMaskFatal
-fd	DMaskFatal
-fc	CMaskFatal
-fn	DMaskNotLin
-fs	DMaskNotLinSat

**THIS IS A PRELIMINARY DOCUMENT, the module described here may or may not be utilized in the final pipelines as described.**

-v (verbose switch)	-
-vv (super-verbose switch)	-
-d (debug switch)	-

**Table 2. Command-line options**

## **4. Processing**

### **4.1. FOWLINEARIZE Processing**

FOWLINEARIZE begins processing by writing its name and version number to standard output (verbose mode only), and then it initializes relevant variables with default values, and checks that the required namelist parameters and/or command-line parameters were passed to it. If this condition is not true, then it writes a message stating which parameters are missing, recommends a look at this document, and terminates by issuing an appropriate exit code to the pipeline executive; otherwise it proceeds as follows.

If an error occurs during processing, then an error message is written to standard output, a termination-status code is written to the log file, and an exit code to the pipeline executive issued.

After processing, the program name and version number, namelist filename (if used), input, and output filenames, values of other input parameters, date and time, processing time, and a termination-status code are written to a log file.

### **2.4. FOWLINEARIZE Processing Phases**

FOWLINEARIZE operates in seven phases: initialization, “unlinearized” data input, model-coefficient data input, optional uncertainty and (p-,d-,c-) mask inputs, linearity computation, results output, and termination. This processing level is depicted in Figure 1.

#### **4.1.1. FOWLINEARIZE Initialization**

FOWLINEARIZE initializes itself by performing the following tasks.

- A.) A message is printed to `STDOUT` (verbose mode only), which includes the program name and version number.
- B.) If specified on the command line, the `NAMELIST` file is opened and read. If any errors are encountered, a message is printed, and execution aborts.
- C.) The remaining command-line inputs are read and checked for correct data range, consistency, etc. If any errors are encountered, a message is printed, and execution aborts.

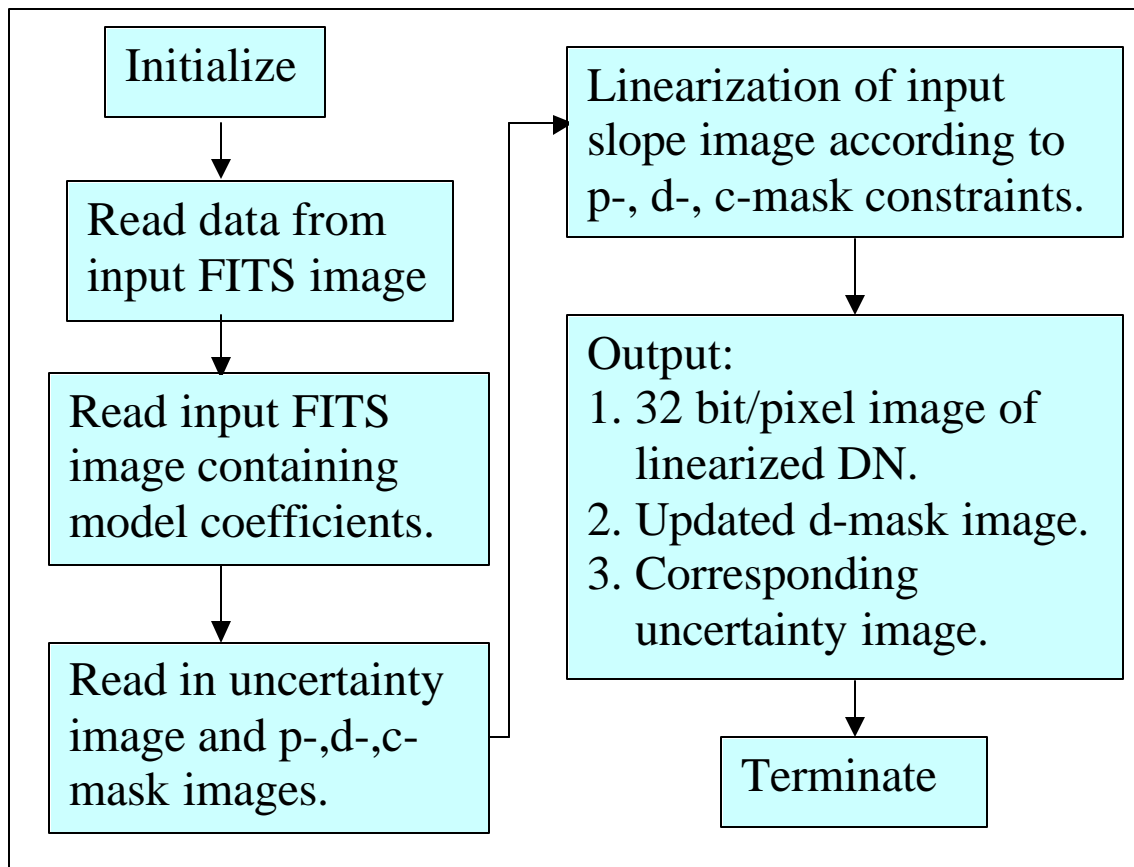


Figure 1. FOWLINEARIZE data and processing flow

**THIS IS A PRELIMINARY DOCUMENT, the module described here may or may not be utilized in the final pipelines as described.**

#### 4.1.2. FITS-Image Data Input

The input image (namelist parameter: FITS\_Image\_Filename1) is read as a standard FITS file and stored in memory. If the input image consists of more than one data plane, FOWLINEARIZE will linearize pixel values in all data planes.

#### 4.1.3. Model Coefficients Input

Coefficients of the non-linearity model are read from a multi-plane FITS image generated by the LINCAL program. A unique set of coefficients is stored in memory for each pixel. For the *quadratic model* (specified by namelist parameter Model\_Flag = 1), the model is specified by three parameters. The linear coefficient  $m$  and quadratic coefficient  $A$  (see Equation 1 below) are read in as the combination  $\alpha = A/m^2$  from the *first plane* of LINCAL's output cube (input namelist parameter: FITS\_Image\_Filename2). Uncertainties for the quadratic model are read from the *third plane* of LINCAL's output cube. These are read as one-sigma values:  $\sigma(A/m^2)$ . The second plane for LINCAL's quadratic case is a saturation plane and is redundant for the purpose of this software.

The *cubic model* (specified by Model\_Flag = 2) is specified by ten parameters in LINCAL's output cube (three coefficients – see Equation 5 below, a saturation plane, one-sigma uncertainty for each coefficient, covariance for every mutual coefficient). These are read from ten separate planes as follows:

Plane 1:  $A'$  coefficient (quadratic term)

Plane 2:  $C'$  coefficient (cubic term)

Plane 3:  $B'$  coefficient (linear term)

Plane 4: Saturation limit.

Plane 5: one-sigma uncertainty in  $A'$

Plane 6: one-sigma uncertainty in  $C'$

Plane 7: one-sigma uncertainty in  $B'$

Plane 8: one-sigma covariance in  $A'$  and  $C'$

Plane 9: one-sigma covariance in A' and B'

Plane 10: one-sigma covariance in C' and B'.

#### 4.1.4. Optional Uncertainty and Mask Image Inputs

An optional uncertainty image corresponding to the input “unlinearized” image can also be read. FOWLINEARIZE also optionally reads in p-mask (pixel), d-mask (DCE) and c-mask (calibration) images. These are also stored in memory and used in the processing stage (see below).

#### 4.1.5. Linearity Computation

The input (non-linear) data are initially derived from summing a set of non-destructive reads using an on board Fowler sampling method. Here we correct for the effect of non-linearity between cumulative “observed” and “expected” photon counts using a novel method. This section gives a brief review of the equations used by FOWLINEARIZE given a quadratic non-linearity model and a complete derivation is given in Appendix I. Corresponding uncertainties and their propagation are described in Appendix II. Appendices III and IV describe the cubic non-linearity model.

The method assumes that a sample of non-destructive reads (or a set of observed cumulative counts  $DN_{obs}$  from an increasing illumination level) as a function of time  $t$  for a particular pixel follows a simple quadratic:

$$DN_{obs} = m_{lab} t - A_{lab} t^2, \quad (1)$$

where  $m_{lab}$  and  $A_{lab}$  are coefficients derived from the LINCAL program using laboratory data. Denoting the input non-linear DN by  $DN_{obs}$ , its corresponding linearized value can be written analytically as follows:

$$DN_{lin} = \frac{1 - \sqrt{1 - 4L DN_{obs}}}{2L}, \quad (2)$$

where

$$L = \frac{\mathbf{a}}{n(w+n)^2} \left[ \left( \sum_{i=w+n+1}^{w+2n} i^2 - \sum_{i=1}^n i^2 \right) - 2 \left( 1 - \frac{t_d}{t_c} \right) n(n+w) \right] \quad (3)$$

$$\mathbf{a} = \left( \frac{A_{lab}}{m_{lab}^2} \right) \quad (4)$$

$n$  = Fowler Number

$w$  = Number of Wait Periods

$t_d$  = Pixel reset - first readout delay time

$t_c$  = Clock period (sampling time)

The parameter  $t_d$  is the delay time from a pixel reset until the first readout. This is pixel dependent and ranges from about 4 msec for pixel (1,1) and about 160 msec for pixel (256,256). It's value as a function of pixel position can be computed from the following general formulae, depending on whether the image was taken in "full-array" or "sub-array" mode:

**Full Array:**  $[i, j] = [1-256, 1-256]$

$$t_d(i, j) = 16.8(256 - i) + 1180 + 10 \text{int} \left[ \frac{j-1}{4} \right] + 648(i-1) \quad \text{msec.} \quad (5)$$

**Sub Array:**  $[i, j] = [1-32, 1-32]$

$$t_d(i, j) = 16.8(248 - i) + 1160 + 10 \text{int} \left[ \frac{j+7}{4} \right] + 108(i+7) \quad \text{msec.} \quad (6)$$

where the "int" function converts the argument to an integer without rounding.

The parameter  $t_c$  is the dock period or sampling-time and can be specified on input via the namelist/command-line parameter "Clock\_Readout". Currently,  $t_c = 200$  msec and 10 msec for full-array and sub-array mode respectively. The software will use this parameter to select the appropriate time-delay ( $t_d$ ) formula to use from above (equations 5 and 6).

For a given pre-determined set of coefficients ( $m_{lab}$ ,  $A_{lab}$ ) for each pixel and a knowledge of the number of Fowler samples and wait periods, equations (2) and (3) completely determine the “true” DN corrected for detector non-linearity.

For a cubic non-linearity model, the cumulative observed  $DN$  as a function of time is described by:

$$DN_{obs} = C't^3 + A't^2 + B't. \quad (7)$$

The solution in this case is more complicated. This is solved using an iterative method described in Appendix III.

#### 4.1.6. FITS-Image Output

The primary product of this software is 32-bit FITS image in units of linearized DN with the same dimensions as the input image (e.g. an image cube). Additional processing information concerning pixels which could not be linearized is contained in an “updated” d-mask image with the same dimensions as the linearized image.

An uncertainty image is produced if its *output filename* is specified (namelist parameter FITS\_Out\_Filename2, command-line option `-o2`). If no *input uncertainty image* filename (parameter FITS\_Image\_Filename3) is specified, the output uncertainty image will be filled with zeros.

#### 4.1.7. Termination

Summary output is appended to the log file (the log file is created if previously non-existent), which includes diagnostic reports for the Q/A Subsystem and the appropriate exit code issued to be picked up by the pipeline executive. A detailed list of log file contents is given in Section 6.1.2.

## 5. Algorithm

### 5.1. Algorithm Specifics

The algorithm employed in this software has been adequately described in the previous section. As a detail, “fatal” pixels defined in the p- and d-mask are handled by replacing them by NaNs in the

output image. Accordingly, all pixels that are NaNs in the output image are reported by updating the relevant d-mask bit for that pixel (defined with bit number 12, integer value 4096). See below for current bit-mask definitions.

Furthermore, there may exist pixels whose “raw” counts lie above the maximum value that can be linearized according to the *quadratic non-linearity model*. For this model, such pixels lead to negative values for the argument of the square root in Eqn. (2) above implying an unphysical solution. These pixels are set to the maximum value that can be linearized as defined by the non-linearity model:  $DN_{lin}(max) = 2 DN_{obs}$  (see Appendix I). If this occurs, a warning is printed to stdout only if the program is executed in “super-verbose” mode (with the `-vv` command-line specification).

Problems and anomalies in the output model generated by LINCAL are obtained from its accompanying c-mask image output. FOWLINEARIZE will not linearize pixels for which a model could not be determined. These are copied to the output image in their original form. Accordingly, the relevant d-mask bit (bit number: 12, integer value 4096) is also set to indicate this.

If a pixel is found to have an observed DN value greater than the saturated value predicted by LINCAL (i.e. the maximum below which a non-linearity model could be determined), bit number 13 (value 8192) is set to indicate this. Before performing this thresholding, it is ensured that the linearized DN is greater than that observed.

The input image containing the model coefficients (LINCAL output) must have the same (NAXIS1 and NAXIS2) dimension as the input image. If not, an error is sent to standard output and the program is aborted.

## 5.2. Default Bit-Mask Settings

All fatal bit settings in the (input) p-mask, (input/output) d-mask and (input) c-mask used by FOWLINEARIZE are defined as input command-line/namelist parameters (see Table 1). Their variable names and default settings are defined below.

PMaskFatal = 8192 (bit 13)

DMaskFatal = 512 (bit 9)

CMaskFatal = 512 (bit 9)

DMaskNotLin = 4096 (bit 12 - fatal; pixel *was not* linearized)

DMaskNotLinSat = 8192 (bit 13 - pixel is saturated in lincal model  
and is beyond correctable non-linearity.  
This is only a warning!)

### 5.3 Assumptions and Requirements

- A. FOWLINEARIZE is generic enough to linearize data in multi-plane FITS image cubes. The same algorithm is applied to each plane.
- B. FOWLINEARIZE assumes that the input FITS-image cube containing the non-linearity model coefficients (namelist parameter: FITS\_Image\_Filename2) is exactly that produced by the LINCAL program. If the namelist parameter Model\_Flag=1, a *quadratic* model LINCAL cube with *three planes* is expected. If Model\_Flag=2, a *cubic* model cube with *ten* planes is expected. These planes were defined in section 4.1.3. Sanity checking is performed to ensure the correct input model cube is specified according to the Model\_Flag parameter. If this input is inconsistent, the program will abort with a message sent to standard output.
- C. The namelist parameter “Clock\_Readout” (command-line option `-c`), is used to determine whether the image was taken in full-array or sub-array mode. This is used to determine which time-delay formula to use (equation 5 or 6, section 4.1.5). Currently, full-array is assumed if Clock\_Readout = 200 msec and sub-array is assumed if Clock\_Readout = 10 msec. Only these two values are recognized by the software.
- D. The input uncertainty image must have the same dimensions as the input image containing the data to be linearized. The NAXIS1 and NAXIS2 header keywords must be equivalent.
- E. FOWLINEARIZE requires that the following keywords be present in the FITS-header of the input slope-image: AFOWLNUM – representing the “Fowler number” and AWAITPER – the number of wait periods. If any of these are not specified, an error message is sent to standard output and the program aborts.
- F. An uncertainty image is only produced if its *output filename* is specified in the namelist (parameter FITS\_Out\_Filename2) or command-line (with option `-o2`). If no *input*

*uncertainty* image (parameter FITS\_Image\_Filename3) was specified, the output uncertainty image will be filled with zeros.

- G. Verbose output (directed to standard output) on which pixels could not be linearized due to either masking or numerical imprecision can be obtained by running the software in “SuperVerbose” mode via the command line parameter `-vv`.

## 6. Output

### 6.1. FOWLINEARIZE Output

FOWLINEARIZE is capable of generating the following output:

- A.) Standard-output processing and status messages.
- B.) A 32-bit FITS image representation of pixels in terms of linearized DN with same number of planes as the input unlinearized FITS image.
- C.) An “updated” d-mask image containing information on pixels which could not be linearized with the same dimensions as the input image.
- D.) A 32-bit uncertainty image in units of DN corresponding to the output linearized image.
- E.) A log file containing processing statistics, status messages and ancillary information.

All FOWLINEARIZE disk output is written to the pathnames that are specified with the output filenames in the command-line or namelist inputs.

#### 6.1.1 FOWLINEARIZE FITS Output

FOWLINEARIZE uses the FITSIO library routines to create FITS-formatted output data files. The routines used are: `fits_read_key_lng`, `fits_insert_key_lng`, `fits_create_file`, `fits_open_file`, `fits_copy_hdu`, `fits_flush_file`, `fits_write_key`, `fits_update_key`, `fits_write_date`, `fits_write_key_str`,

fits\_write\_key\_fixflt, fits\_write\_img, fits\_get\_hdrspace, fits\_read\_record, fits\_write\_record, and fits\_close\_file.

### 6.1.2 FOWLINEARIZE Log-File Output

The information stored in the log file at the output of this program includes: program name and version number, values of all namelist and/or command-line inputs, a message indicating the type of calculation performed, status code, processing time, date and time, and a message indicating program termination.

## 7. Testing

FOWLINEARIZE has been successfully unit-tested as a stand-alone program for a variety of different input cases. The tests were designed to check FOWLINEARIZE robustness and capability of generating corrected results.

Here is a summary of the unit tests that were conducted:

1. Executed FOWLINEARIZE with inputs read from and output written to directories different from where the program was run. Both namelist and command-line input mechanisms were exercised.
2. Executed FOWLINEARIZE with input images consisting of simulated IRAC images containing cosmic radiation hits, saturated pixels and pixels loaded with NaNs. Both quadratic and cubic non-linearity models were exercised.
3. Executed FOWLINEARIZE with different p-mask, d-mask and c-mask bit word settings to test that they integrate and function properly.
4. Executed FOWLINEARIZE with an uncertainty image derived from a noise model and tested that the error propagation is performed correctly.
5. Executed FOWLINEARIZE for all combinations of input parameters, in order to test that they function properly.

6. Executed FOWLINEARIZE on a non-square, large (COSMIC) image.

## 8. Usage Examples

Using a namelist file with verbose (-v) output re-directed to a file "out.log":

```
FOWLINEARIZE -n fowlinearize.nl -v | & tee out.log
```

Without using a namelist file:

```
FOWLINEARIZE -i1 input.fits -i2 lincal.fits -i3  
input_uncert.fits -a ../ancpath -ip pmask.fits -id dmask.fits  
-ic cmask.fits -o1 linearized.fits -o2 output_uncert.fits -m 1  
-fp 8192 -fd 512 -fc 512 -fn 4096 -fs 8192 -v
```

## 9. Glossary

DCE	Data Collection Event
DN	Data Number
IOC	In-Orbit Checkout
SDS	Subsystem Design Specification
SIS	Software Interface Specification
TBD	To Be Determined
TBR	To Be Resolved

## 10. Appendix I (Quadratic FOWLINEARIZE)

We assume that the IRAC non-linearity model relating non-linear observed DN,  $DN_{obs}$ , to sampling read-out time  $t$  is a quadratic function:

$$DN_{obs} = mt - At^2, \quad (1)$$

where  $m$  and  $A$  are pre-determined coefficients computed using a least-squares regression algorithm in the LINCAL program. We can re-write Eqn.1 in the more convenient form:

$$DN_{obs} = mt - \mathbf{a} m^2 t^2, \quad (2)$$

where

$$\mathbf{a} = \left( \frac{A}{m^2} \right) \text{ and generally } \mathbf{a} \geq 0. \quad (3)$$

(As an aside, it is important to note that LINCAL's algorithm adopts the convention  $DN_{obs} = mt + At^2$  so that the  $\alpha$  values are actually negative).

For simplicity we assume that the "linear" rate  $m$  is in units of DN per Fowler time sample length, so that  $t$  can be expressed as a multiple of IRAC sampling read time ( $t_c$ ):

$$t = (i-1)t_c + t_d \rightarrow t = (i-1) + \frac{t_d}{t_c},$$

where  $t_c = 0.2$  and  $0.01$  seconds for full array and sub-array modes respectively and  $i$  is an integer.  $t_d$  is the delay time from the reset of a pixel until the first readout and is pixel dependent (see section 4.1.5). The exact definition of the Fowler sampled result is:

$$DN_{obs} = \frac{1}{n} \sum_{i=w+n+1}^{w+2n} \left( m \left[ (i-1) + \frac{t_d}{t_c} \right] - \mathbf{a} m^2 \left[ (i-1) + \frac{t_d}{t_c} \right]^2 \right) - \frac{1}{n} \sum_{i=1}^n \left( m \left[ (i-1) + \frac{t_d}{t_c} \right] - \mathbf{a} m^2 \left[ (i-1) + \frac{t_d}{t_c} \right]^2 \right), \quad (4)$$

where  $n$  is the Fowler number and  $w$  the number of wait periods. The left summation in Eqn. 4 is the set of “signal reads” and the right summation is the set of “pedestal reads”. Included here is the normalization by Fowler number, which will normally be done on-board as a result of the “barrel-shifting. Rearranging Eqn. 4,

$$DN_{obs} = m \left[ \frac{1}{n} \left( \sum_{i=w+n+1}^{w+2n} [(i-1) + \frac{t_d}{t_c}] - \sum_{i=1}^n [(i-1) + \frac{t_d}{t_c}] \right) \right] - a m^2 \left[ \frac{1}{n} \left( \sum_{i=w+n+1}^{w+2n} [(i-1) + \frac{t_d}{t_c}]^2 - \sum_{i=1}^n [(i-1) + \frac{t_d}{t_c}]^2 \right) \right]. \quad (5)$$

The terms in the outer most square brackets of Eqn.5 are constants for any given values of  $w$ ,  $n$ ,  $t_c$  and  $t_d$ . We will define the first term in brackets to be  $B$  and the second term to be  $C$ . Thus

$$DN_{obs} = Bm - Ca m^2. \quad (6)$$

The only physically acceptable solution for the “linear” rate  $m$  is then

$$m = \frac{B - \sqrt{B^2 - 4Ca DN_{obs}}}{2Ca} \quad (7)$$

The linearized DN,  $DN_{lin}$ , is given by the linear rate  $m$  times the exposure time  $(n + w)$  (multiples of  $t_c$ ). Thus

$$DN_{lin} = [n + w] \frac{B - \sqrt{B^2 - 4Ca DN_{obs}}}{2Ca}. \quad (8)$$

Furthermore, evaluating  $B$  in Eqn.5 we find that  $B \equiv (n + w)$ . Given this and rearranging Eqn.8, we have the final result:

$$DN_{lin} = \frac{1 - \sqrt{1 - 4L DN_{obs}}}{2L}, \quad (9)$$

where

$$L = \mathbf{a} \frac{C}{B^2} = \frac{\mathbf{a}}{n(w+n)^2} \left[ \left( \sum_{i=w+n+1}^{w+2n} i^2 - \sum_{i=1}^n i^2 \right) - 2 \left( 1 - \frac{t_d}{t_c} \right) n(n+w) \right]. \quad (10)$$

The constant  $\alpha$  containing coefficients of the quadratic non-linearity model is defined by Eqn.3.

One may expect to run into precision problems in Eqn.9 when  $L$  becomes very small (as a consequence of small  $\alpha$  values, or when the non-linearity becomes negligible). Taking the limit as  $L \rightarrow 0$  in Eqn. 9 formally yields  $DN_{lin} = DN_{obs}$  which is expected when the non-linearity becomes very small. However as  $L \rightarrow 0$ , the numerator and denominator in Eqn. 9 both approach zero which could lead to numerical instability given the available 4 byte floating point precision used. To avoid this, we can convert Eqn. 9 to a more stable form by multiplying the numerator and denominator by the factor  $1+(1-4LDN_{obs})^{1/2}$ . A little algebra leads to the equivalent result:

$$DN_{lin} = \frac{2DN_{obs}}{1 + \sqrt{1 - 4LDN_{obs}}} \quad (11)$$

Equation (11) implies that no physical solution exists if  $DN_{obs} > (4L)^{-1}$ . This is equivalent to saying that the maximum linearized value that can result from the above model occurs when  $DN_{obs} = (4L)^{-1}$  and hence,

$$DN_{lin}(\max) = 2DN_{obs} \quad (12)$$

For a given pre-determined set of model coefficients ( $m, A$ ), which is given by LINCAL as the convenient combination  $\alpha=A/m^2$  for each pixel in the array, and a knowledge of the number of Fowler samples ( $n$ ) and wait periods ( $w$ ), Eqn. 11 completely determines the “true” DN (or effective linear count-rate) that would result in the absence of detector non-linearity.

The validity of Eqn.9 has been verified via an IRAC simulator. A simulator was written which computed the observed DN by duplicating the Fowler sampling scheme using an input source flux and a quadratic non-linearity model. The simulated, observed non-linear DN were then converted to linearized DN using the above formalism. This correctly reproduced the “true” input flux.

## 11. Appendix II (Quadratic FOWLINEARIZE Uncertainties)

Given a one-sigma uncertainty in measured pixel DN,  $\sigma(DN_{obs})$ , we derive the corresponding uncertainty in the linearized value  $\sigma(DN_{lin})$  using formal error propagation in Equations (9) and (10) (Appendix I). Since the non-linearity model coefficients (contained in the  $L$  parameter) and observed DN are independent, the variance can be written:

$$\mathbf{s}_{DN_{lin}}^2 = \left( \frac{\partial DN_{lin}}{\partial L} \right)^2 \mathbf{s}_L^2 + \left( \frac{\partial DN_{lin}}{\partial DN_{obs}} \right)^2 \mathbf{s}_{DN_{obs}}^2.$$

Using Equations (9) and (10), the terms in this error formula are given by the following expressions:

$$\left( \frac{\partial DN_{lin}}{\partial L} \right)^2 = \left[ \frac{DN_{obs}}{L(1-4LDN_{obs})^{1/2}} - \frac{1-(1-4LDN_{obs})^{1/2}}{2L^2} \right]^2$$

$$\left( \frac{\partial DN_{lin}}{\partial DN_{obs}} \right)^2 = (1-4LDN_{obs})^{-1}$$

$$\mathbf{s}_L^2 = \mathbf{s}^2 \left( \frac{A}{m^2} \right) \left[ \frac{1}{n(w+n)^2} \right]^2 \left[ \sum_{i=w+n+1}^{w+2n} i^2 - \sum_{i=1}^n i^2 - 2 \left( 1 - \frac{t_d}{t_c} \right) n(n+w) \right]^2$$

$$\mathbf{s}_{DN_{obs}}^2 = \text{From input uncertainty image (computed via a noise model).}$$

The quantity  $\sigma^2(A/m^2)$  is extracted from the *third plane* of LINCAL's output image cube. The one-sigma uncertainties in the linearized slopes are stored in a single plane output uncertainty image.

## 12. Appendix III (Cubic FOWLINEARIZE)

We assume that the non-linearity model relating non-linear observed DN,  $DN_{obs}$ , to sampling read-out time  $t$  is a quadratic function:

$$DN_{obs} = C't^3 + A't^2 + B't. \quad (1)$$

where  $C$ ,  $A$  and  $B$  are pre-determined coefficients computed using a least-squares regression algorithm in the LINCAL program. Using the substitution  $DN_{lin} = B't$ , we can re-write Eqn.1 in the more convenient form:

$$DN_{obs} = \mathbf{b}DN_{lin}^3 + \mathbf{a}DN_{lin}^2 + DN_{lin}, \quad (2)$$

where  $DN_{lin}$  is the linearized DN,

$$\mathbf{a} = \left( \frac{A'}{B'^2} \right), \quad \mathbf{b} = \left( \frac{C'}{B'^3} \right). \quad (3)$$

$DN_{lin}$  can be described as a “rate” ( $R$ ) multiplied by time ( $t$ ):

$$DN_{lin} = Rt. \quad (4)$$

We can therefore re-write Eqn.(2):

$$DN_{obs} = \mathbf{b}R^3t^3 + \mathbf{a}R^2t^2 + Rt, \quad (5)$$

For simplicity, we assume that the “linear” rate  $R$  is in units of DN per Fowler time length, so that  $t$  can be expressed in terms of IRAC sampling read time ( $t_c$ ):

$$t = (i-1)t_c + t_d \rightarrow t = (i-1) + \frac{t_d}{t_c},$$

where  $t_c = 0.2$  and  $0.01$  seconds for full array and sub-array modes respectively and  $i$  is an integer.  $t_d$  is the delay time from the reset of a pixel until the first readout and is pixel dependent (see section 4.1.5). The exact definition of the Fowler sampled result is:

$$DN_{obs} = \frac{1}{n} \sum_{i=w+n+1}^{w+2n} \left( \mathbf{b}R^3 \left[ (i-1) + \frac{t_d}{t_c} \right]^3 + \mathbf{a}R^2 \left[ (i-1) + \frac{t_d}{t_c} \right]^2 + R \left[ (i-1) + \frac{t_d}{t_c} \right] \right) - \frac{1}{n} \sum_{i=1}^n \left( \mathbf{b}R^3 \left[ (i-1) + \frac{t_d}{t_c} \right]^3 + \mathbf{a}R^2 \left[ (i-1) + \frac{t_d}{t_c} \right]^2 + R \left[ (i-1) + \frac{t_d}{t_c} \right] \right), \quad (6)$$

where  $n$  is the Fowler number and  $w$  the number of wait periods. The left summation in Eqn. 6 is the set of “signal reads” and the right summation is the set of “pedestal reads”. Included here is the normalization by Fowler number, which will normally be done on-board as a result of the “barrel-shifting. Re-arranging Eqn. (6),

$$\begin{aligned}
 DN_{obs} = & \mathbf{b}R^3 \left( \frac{1}{n} \sum_{i=w+n+1}^{w+2n} \left[ (i-1) + \frac{t_d}{t_c} \right]^3 - \frac{1}{n} \sum_{i=1}^n \left[ (i-1) + \frac{t_d}{t_c} \right]^3 \right) \\
 & + \mathbf{a}R^2 \left( \frac{1}{n} \sum_{i=w+n+1}^{w+2n} \left[ (i-1) + \frac{t_d}{t_c} \right]^2 - \frac{1}{n} \sum_{i=1}^n \left[ (i-1) + \frac{t_d}{t_c} \right]^2 \right) \\
 & + R \left( \frac{1}{n} \sum_{i=w+n+1}^{w+2n} \left[ (i-1) + \frac{t_d}{t_c} \right] - \frac{1}{n} \sum_{i=1}^n \left[ (i-1) + \frac{t_d}{t_c} \right] \right). \tag{7}
 \end{aligned}$$

Further simplifying, we have

$$\begin{aligned}
 DN_{obs} = & \mathbf{b}R^3 \left[ \frac{1}{n} \left( \sum_{i=w+n+1}^{w+2n} i^3 - \sum_{i=1}^n i^3 \right) - \frac{3}{n} \left( 1 - \frac{t_d}{t_c} \right) \left( \sum_{i=w+n+1}^{w+2n} i^2 - \sum_{i=1}^n i^2 \right) + 3(n+w) \left( 1 - \frac{t_d}{t_c} \right)^2 \right] \\
 & + \mathbf{a}R^2 \left[ \frac{1}{n} \left( \sum_{i=w+n+1}^{w+2n} i^2 - \sum_{i=1}^n i^2 \right) - 2 \left( 1 - \frac{t_d}{t_c} \right) (n+w) \right] \\
 & + R[w+n]. \tag{8}
 \end{aligned}$$

The terms in the square brackets of Eqn. (8) are constants for any given values of  $w$ ,  $n$ ,  $t_c$  and  $t_d$ . We will define these terms from left-to-right in Eqn. (8) to be  $C$ ,  $A$ , and  $B$ . Thus,

$$DN_{obs} = \mathbf{b}CR^3 + \mathbf{a}AR^2 + BR \tag{9}$$

The next step is to solve Eqn. (9) for  $R$  as a function of the observed DNs ( $DN_{obs}$ ). The linearized DN can then be computed by multiplying the rate with the effective exposure time ( $w + n$ ) or equivalently  $B$ :

$$DN_{lin} = RB \tag{10}$$

A potential snag is which root of the cubic equation (9) is the one we really want for  $R$ . From behavior of the non-linearity, we expect the physical solution to be the “smallest”, positive “real” (non-imaginary) root. The larger of two or three positive roots would correspond to a given DN level ( $DN_{obs}$ ) that lies to the right of a local maximum (the saturation limit) on the non-linear curve.

To avoid any pathological solutions derived from standard analytic formulae for the cubic, we can make use of the fact that the physical solution for  $R$  ( $DN_{lin}/B$ ) will be within 5-20% of the measured

(observed) value,  $DN_{obs}/B$ , and use a Newton-Raphson iterative scheme to solve for  $R$ . The problem reduces to solving for the roots of the function:

$$f(R) = \mathbf{b}CR^3 + \mathbf{a}AR^2 + BR - DN_{obs} = 0 \quad (11)$$

The Newton-Raphson recurrence relation is given by:

$$R_{i+1} = R_i - \frac{f(R_i)}{f'(R_i)}, \quad (12)$$

where  $f'(R)$ , is the first derivative of the function defined by Eqn. (11). With our trial solution  $R_i = DN_{obs}/B$ , equation (11) is solved recursively for  $R$  until the result converges to what we “hope” will be our desired solution to the cubic (equation 9). More formally, the recursion is terminated when the value for  $R$  between two successive iterations satisfies

$$\frac{|R_{i+1} - R_i|}{R_{i+1}} \leq \text{FRACTIONAL\_NEWT\_RAPH\_TOL}, \quad (13)$$

where FRACTIONAL\_NEWT\_RAPH\_TOL is a fractional convergence parameter defined in the include file *fowlinearize.h*. This parameter is currently set at  $10^{-10}$ . This value is adequate enough (confirmed from testing) to handle all non-linearity measurements encountered on IRAC so far. For typical pixel values, we find that no more than *five* iterations are needed to achieve this level of convergence.

This method is expected to be robust due to our presumption that the linearized DN will be in the vicinity of that observed. However, there may be cases where the solution fails to converge due to either bad pixels, or, a “bad” non-linearity model fit. If the solution fails to converge within the fractional tolerance parameter defined in equation (13), the pixel values are copied in their original form to the output linearized image. Furthermore, there is also the possibility that we may converge on a value which does not make physical sense. We account for this by flagging those solutions for  $DN_{lin}$  which are highly discrepant from their corresponding  $DN_{obs}$  values. These pixels are not modified. Their original values are copied to the output linearized image if either  $DN_{lin} < 0$ ,  $DN_{lin} < 0.5DN_{obs}$  or  $DN_{lin} > 2DN_{obs}$ .

### 13. Appendix IV (Cubic FOWLINEARIZE Uncertainties)

The uncertainty in a linearized DN value is computed from two components, first the uncertainty in the non-linearity model coefficients derived from LINCAL's fitting, and second, the actual uncertainty in the measure value  $DN_{obs}$ . The first component can be computed by propagating errors in the model coefficients as defined by equation (1) in section 12:

$$\mathbf{s}_{DN_{obs}}^2 (coeff) = \mathbf{s}_C^2 t^6 + 2\mathbf{s}_{C'A}^2 t^5 + \mathbf{s}_{A'}^2 t^4 + 2\mathbf{s}_{C'B'}^2 t^4 + 2\mathbf{s}_{A'B'}^2 t^3 + \mathbf{s}_{B'}^2 t^2, \quad (14)$$

where the variances and covariances are computed by LINCAL and the time variable can be written

$$t = \frac{DN_{lin}}{B'} \quad (15)$$

The corresponding error in  $DN_{lin}$  due to uncertainties (only) in the non-linearity model coefficients can be approximated (to first order) by

$$\mathbf{s}_{DN_{lin}}^2 (coeff) \cong \mathbf{s}_{DN_{obs}}^2 (coeff) \left/ \left( \left. \frac{dDN_{obs}}{dt} \right|_{t=\frac{DN_{lin}}{B'}} \right)^2 \right. \quad (16)$$

$$\text{where } \left. \frac{dDN_{obs}}{dt} \right|_{t=\frac{DN_{lin}}{B'}} = \frac{3C'DN_{lin}^2}{B'^2} + \frac{2A'DN_{lin}}{B'} + B' \quad (17)$$

The total uncertainty on the linearized DN is given by the quadrature sum of uncertainties of the derived model coefficients (equations 14, 16 and 17) and those in the measured DN ( $\sigma_{DN_{obs}}$ ) read from an input image (pre-computed via a noise model):

$$\mathbf{s}_{DN_{lin}} = \left[ \mathbf{s}_{DN_{lin}}^2 (coeff) + \mathbf{s}_{DN_{obs}}^2 \right]^{1/2}. \quad (18)$$

### 14. Acknowledgements

Frank Masci is grateful to Jason Surace, Rick Arendt and Myungshin Im for providing the linearization algorithm and to John Fowler for helping out with the uncertainty estimation.

**THIS IS A PRELIMINARY DOCUMENT, the module described here may or may not be utilized in the final pipelines as described.**