674-SO-43, Version 3.2,
SSC-PD-4059

SIRTF Science Center

Downlink Segment

Subsystem Design Specification

AOT Products Subsystem: DESATSLOPE

26 August 2002

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 Latter 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	October 31, 2001
2.0	Changed algorithm to allow two "Ignore_Frame" parameters.	February 20, 2002
3.0	Modified the algorithm to account for the DCENUM keyword starting at zero.	July 29, 2002
3.1	Re-scale the input "Diff_Sat_Threshold" parameter internally according to the EXPTIME keyword value in the input FITS header. On input, it is assumed that "Diff_Sat_Threshold" pertains to a "30 sec" exposure in input image units.	August 19, 2002
3.2	Saturation detection is now performed by checking if the saturation d-mask bit was set in the input D-Mask from upstream processing (hardcoded as bit 13). This is only possible if the input D-Mask is specified. If unspecified, the software uses the Diff_Sat_Threshold input parameter – but only if it has a non-zero value.	August 26, 2002

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

2 Table of Contents

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

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

List of Tables

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

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 DESATSLOPE program reads (SUR-mode) “slope” image data from a FITS file and corrects those slopes which have been fit to saturated ramp-data using a quadratic non-linearity model. The purpose of slope de-saturation is to facilitate a more robust droop-correction in downstream processing.

The algorithm is equivalent to a “reverse” linearization whereby pixel values in the (first) difference plane are assumed to be close to linear and from these, observed slopes are predicted using the non-linearity model. The primary product is a 32-bit/pixel output image of “de-saturated” slopes at pixel locations where saturation has been detected. Output data units are equivalent to input data units.

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. This program is specifically designed to de-saturate image data acquired with SIRTF’s 24μm array in the “SUR-mode”. DESATSLOPE is written in standard C.

2.1. DESATSLOPE Requirements

DESATSLOPE 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 SUR-mode FITS file, a non-linearity model image, and mask images.

C.) Produce as primary output a new SUR-mode image with a “de-saturated” slope plane.

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 DESATSLOPE 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 October 31, 2001.

2.3.2. Version 2.0

Version 2.0 includes a modification to the way input parameters are handled in the de-saturation algorithm and it also accommodates two initial Ignore_Frame parameters.

2.3.3. Version 3.1

This version re-scales the input "Diff_Sat_Threshold" parameter internally according to the EXPTIME keyword value in the input FITS header. On input, it is assumed that "Diff_Sat_Threshold" pertains to a "30 sec" exposure in input image units.

2.3.4. Version 3.2

Saturation detection is now performed by checking if the saturation d-mask bit was set in the input D-Mask from upstream processing (hardcoded as bit 13). This is only possible if the input D-Mask is specified. If unspecified, the software uses the Diff_Sat_Threshold input parameter – but only if it has a non-zero value.

2.4. Liens

No major liens have been identified.

3. Input

3.1. DESATSLOPE Input

DESATSLOPE 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. DESATSLOPE NAMELIST Input

DESATSLOPE reads the NAMELIST file whose name is passed to it by start-up script. The name of the NAMELIST is DESATSLOPEIN. 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 slopes to de-saturate.	161	C	-	Null
FITS_Image_Filename2	Required input FITS-image filename containing quadratic-model coefficients and their uncertainties.	161	C	-	Null
FITS_Image_PMask_Filename	Optional p-mask FITS-image	161	C	-	Null
FITS_Image_DMask_Filename	Optional d-mask FITS-image	161	C	-	Null
FITS_Image_CMask_Filename	Optional c-mask FITS-image	161	C	-	Null
FITS_Out_Filename1	Required output FITS-image filename containing linearized slopes (and first difference).	161	C	-	Null
Ignore_Frames1	Number of initial samples to ignore if image cube is the first in a sequence.	1	I*2	-	0
Ignore_Frames2	Number of initial samples to ignore if image cube has order > 1 in image sequence.	1	I*2	-	0
Diff_Sat_Threshold	Optional saturation threshold in difference plane pertaining to a 30 sec exposure.	1	R*4	Input image units	0
CmdFrm_Keyword	FITS keyword name designating total frame count	8	C	-	DCE_FRMS

PMaskFatal	Fatal PMask data bits	1	I*2	-	8192
DMaskFatal	Fatal DMask data bits	1	I*2	-	8192
CMaskFatal	Fatal CMask data bits	1	I*2	-	512
DMaskDESAT	DMask data bit flag indicating de-saturation *was performed*	1	I*2	-	16
Log_Filename	Optional output log filename	161	C	-	Stdout
Ancillary_File_Path	Pathname where supporting source files are installed.	161	C	-	./ (current directory)

Table 1. Namelist file

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

```
  &DESATSLOPEIN
  Comment = 'Generic namelist file for desatslope, default values.',
  Ancillary_File_Path = '../desatslope_v1',
  FITS_Image_Filename1 = './testing/sur_input_img.fits',
  FITS_Image_Filename2 = './testing/lincal.fits',
  FITS_Image_PMask_Filename = './testing/pmask.fits',
  FITS_Image_DMask_Filename = './testing/dmask.fits',
  FITS_Image_CMask_Filename = './testing/cmask.fits',
  FITS_Out_Filename1 = './testing/desatslope.fits',
  Log_Filename = 'stdout',
  Comment = 'FITS header keyword for total frame count',
  CmdFrm_Keyword = 'DCE_FRMS',
  Comment = 'Number of frames to ignore for de-saturation (only used if not
in header)',
  Ignore_Frames1 = 1,
  Ignore_Frames2 = 1,
  Comment = 'Saturation threshold in difference plane: optional, d-mask bit
setting is searched for first',
  Diff_Sat_Threshold = 900,
```

```
  PMaskFatal = 8192,
  DMaskFatal = 16384,
  CMaskFatal = 256,
  Comment = 'Dmask bit indicating slope pixel was de-saturated',
  DMaskDESAT = 16,
 &END
```

3.1.2. DESATSLOPE 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. DESATSLOPE FITS Input

DESATSLOPE 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.

Command-line option	**Variable**
-n	Namelist_Filename
-i1	FITS_Image_Filename1
-i2	FITS_Image_Filename2
-ip	FITS_Image_PMask_Filename
-id	FITS_Image_DMask_Filename
-ic	FITS_Image_CMask_Filename
-o1	FITS_Out_Filename1

-l	Log_Filename
-a	Ancillary_File_Path
-s	Diff_Sat_Threshold
-g1	Ignore_Frames1
-g2	Ignore_Frames2
-k	CmdFrm_Keyword
-fp	PMaskFatal
-fd	DMaskFatal
-fc	CMaskFatal
-fn	DMaskDESAT
-v (verbose switch)	-
-vv (super-verbose switch)	-
-d (debug switch)	-

Table 2. Command-line options

4. Processing

4.1. DESATSLOPE Processing

DESATSLOPE begins processing by writing its name and version number to standard output (verbose mode only), and then it initializes relevant variables with defaults 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 a log file.

4.2 DESATSLOPE Processing Phases

DESATSLOPE operates in eight phases: initialization, slope-image data input, model-coefficient data input, p-,d-,c-mask inputs, saturation detection, de-saturation computation, results output, and termination. This processing level is depicted in Figure 1.

4.1.1. DESATSLOPE Initialization

DESATSLOPE 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.

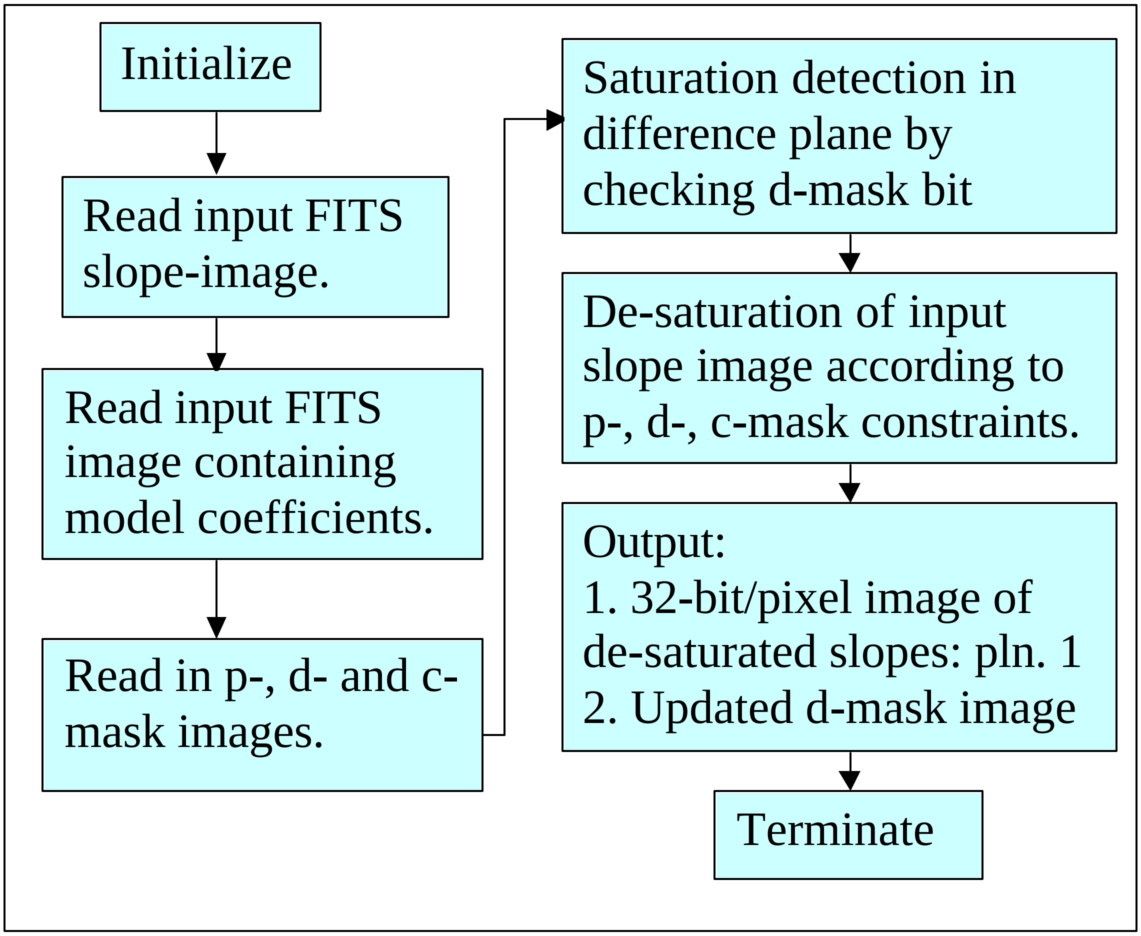

Figure 1. DESATSLOPE data and processing flow

4.1.2. Slope-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 is a data cube (as currently assumed in SIRTF's 24µm SUR-mode where plane 1 is the slope image and plane 2 the difference image), DESATSLOPE assumes the first plane contains the slope values to de-saturate.

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

4.1.3. Model Coefficients Input

Coefficients of the quadratic model used for the non-linearity correction are read from a multi-plane FITS image as generated by the LINCAL program (see Eqn. 1). A unique set of coefficients is stored in memory for each pixel. The linear coefficient m and quadratic coefficient A (see equation 1, Section 4.1.4) are read in as the combination A/m^2 from the first plane of LINCAL's output cube (input namelist parameter: FITS_Image_Filename2).

4.1.4. Mask Image Inputs

DESATSLOPE 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. Saturation Detection

Only those pixels which are suspected to saturate in a ramp (of non-destructive cumulative reads) are used in the de-saturation computation below. There are two options for the saturation detection step:

I. If the D-Mask filename is specified on input (FITS_Image_DMask_Filename), the saturation d-mask bit (#13; value $2^{13} = 8192$) is searched for. In a pipeline, this is set upstream by the SATMASK software on unprocessed input data.

II. If no D-Mask is specified, saturated pixels are detected by thresholding pixel values in the difference plane (plane 2) of the input image via the user-specified namelist/command-line parameter Diff_Sat_Threshold. This input parameter pertains to a 30 sec exposure so that prior to thresholding, it is re-scaled according to the actual exposure time of the input image (header keyword "EXPTIME"):

$$\text{Diff\_Sat\_Threshold} = \text{Diff\_Sat\_Threshold} * \frac{31.46}{\text{EXPTIME}},$$

where the EXPTIME value is in "real" seconds. Note that thresholding via the "Diff_Sat_Threshold" parameter is only performed if it's value is non-zero on input. Pixels whose first-difference values in the ramp exceed this parameter are expected to eventually saturate in the ramp. This procedure is essentially equivalent to the algorithm used in the SATMASK software to initially flag saturated pixels in the SUR-mode.

4.1.6. De-saturation Computation

The input (SUR-mode) slopes are initially derived from on-board fitting to samples of non-destructive reads (ramps) using linear regression software. Here we correct for the possible effect of saturation in the ramp. Ramps which saturate (expected if the difference between first and second reads exceeds some threshold – see section 4.1.5), will lead to a biased estimate of a slope fit as shown in Figure 2. Consequently, this will lead to an inaccurate droop correction in downstream processing.

The *main assumption* is that the difference between first and second reads in the ramp is a good representation of the linearized slope. From this assumption, the slope that would be observed which includes the effects of detector non-linearity can be computed using the non-linearity model. In essence, this is a "reverse" linearization of the slope. This section gives a brief review of the equations used by DESATSLOPE and a complete derivation is given in Appendix I.

The method assumes that a sample of initial non-destructive reads (observed cumulative counts DN_{obs}) as a function of time t for a particular pixel follows a simple quadratic:

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

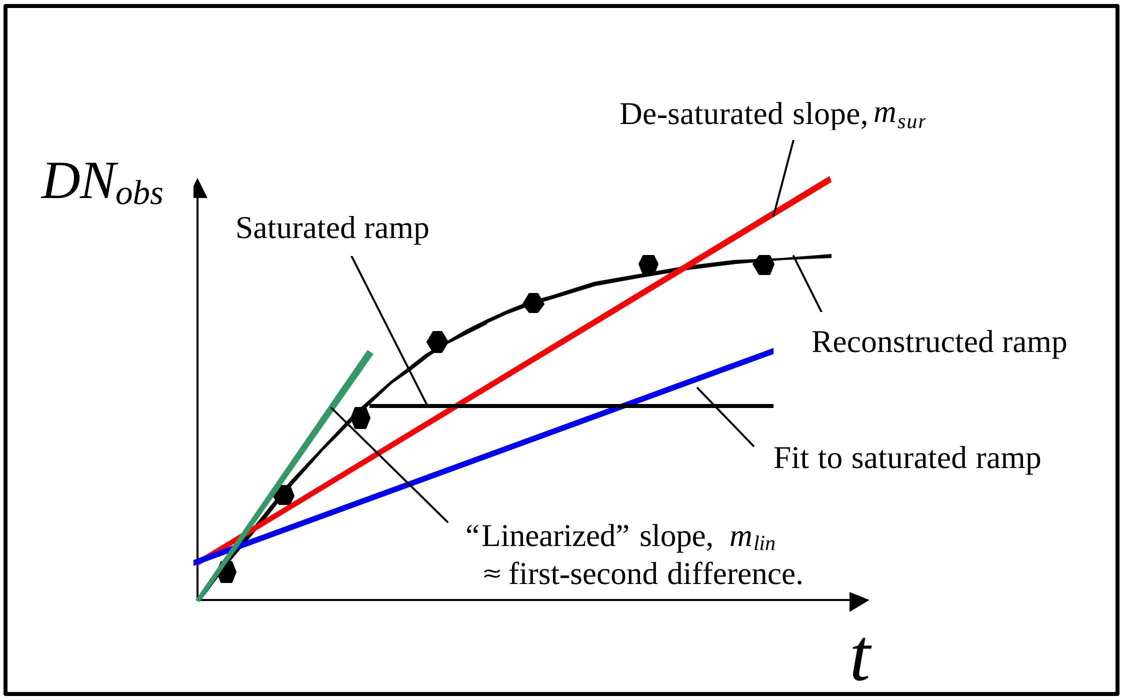

Figure 2. Schematic showing effect of ramp saturation

where m_{lab} and A_{lab} are coefficients derived by the LINCAL program using laboratory data. Denoting the (input) linearized slope by m_{lin} (computed from the first and second difference reads), its corresponding (de-saturated) SUR-mode slope fit to the ramp m_{sur} can be written analytically as follows:

$$m_{sur} = m_{lin} - L m_{lin}^2 \quad , \qquad (2)$$

where

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

$$L = \left(\frac{A_{lab}}{m_{lab}^2}\right)\sum_{i=N_{start}}^{N_{end}} f_1 t_i^2 - f_2 t_i^3, \quad \text{and} \qquad (3)$$

$$f_1 = \frac{\sum_i t_i}{\left(\sum_i t_i\right)^2 - N_s \sum_i t_i^2}, \qquad f_2 = \frac{N_s}{\left(\sum_i t_i\right)^2 - N_s \sum_i t_i^2}$$

$$N_{start} = \text{First sample number (plane) to process in input cube}$$
$$N_{end} = \text{Maximum sample (plane) number in input cube}$$
$$N_s = \text{Number of samples to which slope-fit applies} = N_{end} - N_{start} + 1$$

The last three parameters are discussed in section 5.3. See the Appendix for a full derivation.

The formalism which leads to Eqns (2) and (3) is a new method and has the advantage of being independent of the intensity (absolute DN) in a particular pixel. For a given pre-determined set of coefficients (m_{lab} , A_{lab}) for each pixel and a knowledge of the effective number of samples used in determining the on-board (SUR-mode) slope, Eqns (2) and (3) can be used to infer the "observed" slope (count-rate) corrected for effects of saturation in the ramp.

4.1.7. FITS-Image Output

The primary product of this software is a 32-bit FITS image of slopes in units equivalent to the input data. The slope image is contained in the *first plane* of the output FITS cube and the second plane contains the difference image copied from the input image cube. Pixels where no saturation is detected are not modified. These are also copied directly from the input image.

Additional processing information concerning pixels which were desaturated is contained in an "updated" d-mask image (with the same filename as the input d-mask image).

4.1.8. 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

To avoid unrealistic or discrepant desaturated slope values due to anomalies in either the linearity model or suspect input slope values, we perform a sanity check by comparing the desaturated slope m_{sur} with the input linear value m_{lin}. In all cases, we expect $m_{sur} < m_{lin}$ (see Figure 2). If this is not satisfied, we copy the input SUR-mode slope pixels directly to the output image in their original form.

The simple 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. These pixels are not used in processing. See below for current bit-mask definitions.

Problems and anomalies in the linearity model generated by LINCAL are obtained from its accompanying c-mask image output. DESATSLOPE will not desaturate pixels for which a model could not be determined. These are copied to the output image in their original form.

The input image cube containing the model coefficients (LINCAL output) must have the same (NAXIS1 and NAXIS2) dimension as the input SUR-mode slope-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 DESATSLOPE are defined as input command-line/namelist parameters (see Table 1). Their variable names and default settings are as follows:

PMaskFatal = 8192 (bit 13)

DMaskFatal = 8192 (bit 13)

CMaskFatal = 512 (bit 9)

Pixels in the slope plane which have been de-saturated are indicated by setting a bit in the d-mask:

DMaskDESAT = 16 (bit 4: pixel *was de-saturated*)

5.3. Algorithm-Implementation Details

Version 2.0 of DESATSLOPE uses a specific algorithm to compute the first (N_{start}), last (N_{end}) and total number of read samples (N_s) for determining the de-saturated slope via Equations (2) and (3). This is specific to image data acquired with SIRTF's 24μm array. The algorithm is as follows:

As a consequence of the data acquisition method, the FITS header keyword DCENUM is used to determine which read samples are used in the slope-fit. This parameter represents a DCE sequence number and depending on whether we're dealing with the first DCE in a sequence of exposures, the first non-zero (valid) sample will be different according to the following logic:

If DCENUM = 0:

$$N_{start} = 3 + Ignore\_Frames1$$

Otherwise, if DCENUM > 0:

$$N_{start} = 1 + Ignore\_Frames2,$$

where Ignore_Frames1 and Ignore_Frames2 = 0. These represent any initial unwanted read samples and are defined by FITS header keywords IGN_FRM1 and IGN_FRM2 respectively. If these do not exist in the input image header, they are read from corresponding namelist/command-line parameters. If they are not specified in the namelist or command-line, default values of 0 are assigned.

The "N_{end}" parameter in equation (3) is computed from the FITS header keywords DCE_FRMS and FRMFLYBK. DCE_FRMS represents the number of commanded samples (associated Ge frames) in an acquisition cycle. FRMFLYBK represents the effective number of samples during a "fly-back" of the scan-mirror. This parameter is given by:

$$N_{end} = 0.25 * (DCE\_FRMS - FRMFLYBK)$$

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

5.4. Assumptions and Requirements

A. DESATSLOPE assumes that the slope-image is the *first plane* in the input FITS-image cube.

B. DESATSLOPE assumes that the input FITS-image cube containing the linearity model coefficients (namelist parameter: FITS_Image_Filename2) is exactly that produced by the LINCAL program. The *first plane* of this cube is assumed to contain the parameter combination A/m^2 (see equation 1, Section 4.1.6).

C. DESATSLOPE requires that the following keywords be present in the FITS-header of the input slope-image: T_INT – the sampling time interval between non-destructive reads, DCENUM (see above), DCE_FRMS or its equivalent if specified via the namelist parameter CmdFrm_Keyword (see also above), FRMFLYBK – the number of samples in a scan-mirror “fly-back” duration and EXPTIME – the total exposure time. If any of these are not specified, an error message is written to standard output and the program aborts.

6. Output

6.1. DESATSLOPE Output

DESATSLOPE 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 “slopes” corrected for saturation in the ramp with units equivalent to the input data. This is contained in the *first plane* of the output FITS cube. Only pixels which are expected to saturate in the ramp are modified. The *second plane* contains the first difference values copied directly from the input image.

C.) An “updated” d-mask image containing information on which pixels were desaturated.

D.) A log file containing processing statistics, status messages and ancillary information.

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

6.1.1 DESATSLOPE FITS Output

DESATSLOPE 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 DESATSLOPE 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

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

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

1. Tested the de-saturation algorithm used by DESATSLOPE directly on RAW-mode image data consisting of different intensity levels. Results show that the ramp can be accurately reconstructed given prior knowledge of the linearity curve. A slope fit to this RAW-cube recovers the “observed” de-saturated slope.

2. Executed DESATSLOPE 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.

3. Executed DESATSLOPE with input images consisting of simulated MIPS 24μm images containing rad-hits and pixels loaded with NaNs.

4. Executed DESATSLOPE with different p-mask, d-mask and c-mask inputs to test that they integrate and function properly.

5. Executed DESATSLOPE for all combinations of input parameters, in order to test that they function properly.

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

8. Usage Examples

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

```
DESATSLOPE –n desatslope.nl -v | & tee out.log
```

Without using a namelist file:

```
DESATSLOPE –i1 input_slopes.fits –i2 lincal_model.fits –a
../ancpath –ip pmask.fits -id dmask.fits –ic cmask.fits –o1
output_slopes.fits –s 830 –g1 1 –g2 1 –fp 8192 –fd 8192
-fc 512 –fn 16 -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

For comparison with equation (3) of section 4.1.6, we assume here $N_{start} = 1$, $N_{end} = N_s = N$. From a set of N non-destructive read samples (RAW-mode acquisition), software on board SIRTF will convert this data to SUR-mode format which consists of slopes m_{sur} for every pixel in the array. This is achieved using a linear regression algorithm and fitting for the equation:

$$y_i = m_{sur} t_i + c, \qquad (1)$$

where $y_i = DN_{obs}$ is the observed (cumulative) DN after sample time t_i. m_{sur} is computed by minimizing c^2 where

$$c^2 = \sum_{i=1}^{N} (y_i - m_{sur} t_i - c)^2, \quad \text{and evaluating} \quad \frac{\partial c^2}{\partial m_{sur}} = \frac{\partial c^2}{\partial c} = 0 .$$

Carrying out the differentiations and solving these equations for m_{sur} leads to the analytic expression:

$$m_{sur} = \sum_{i=1}^{N} [f_1 - f_2 t_i] y_i, \qquad (2)$$

where $$f_1 = \frac{\sum_i t_i}{\left(\sum_i t_i\right)^2 - N \sum_i t_i^2}, \qquad f_2 = \frac{N}{\left(\sum_i t_i\right)^2 - N \sum_i t_i^2} \qquad (3)$$

We now outline the method used to relate m_{sur} to the linear slope m_{lin} (see Fig.2). For a quadratic non-linearity model, we will assume that a sample of (RAW-mode) “non-destructive reads” from laboratory data can be fit by the following equation:

$$y_i = m_{lab} t_{lab} - A_{lab} t_{lab}^2 , \quad (4)$$

where m_{lab} and A_{lab} are coefficients computed using a least-squares regression algorithm in the LINCAL program.

For a "real" source with arbitrary uniform illumination level (i.e. constant photon output rate) m_{lin} (effectively the slope in the linear regime of a detector), its slope is related to a pre-determined laboratory value at any time t_i by equating the total accumulated counts:

$$m_{lab} t_{lab} = m_{lin} t_i \quad (5)$$

Using Eqn. (5), Eqn. (4) can be transformed to describe a new non-linearity curve for a source with arbitrary count rate m_{lin}:

$$y_i = m_{lin} t_i - m_{lin}^2 \left(\frac{A_{lab}}{m_{lab}^2} \right) t_i^2 . \quad (6)$$

Substituting Eqn. (6) into the general SUR-mode slope definition (Eqn. 2) leads to the following simple expression for m_{sur} :

$$m_{sur} = m_{lin} - L\, m_{lin}^2 , \quad (7)$$

where L is a constant depending on pre-determined values of the model coefficients and summations over the sampling time interval t_i :

$$L = \left(\frac{A_{lab}}{m_{lab}^2} \right) \sum_{i=1}^{N} f_1 t_i^2 - f_2 t_i^3 , \quad (8)$$

and f_1 , f_2 are defined by Eqn. (3).

For a given pre-determined set of (m_{lab} , A_{lab}), which is given by LINCAL as the convenient combination A_{lab}/m_{lab}^2 for each pixel in the array, and a knowledge of the number of samples acquired in the RAW-mode, eqns (7) and (8) completely determine the "observed" slope that would result in the absence of saturation.