| | |
|---|---|
| **SUBJECT:** | SPIRE Pipeline Description |
| **PREPARED BY:** | Tanya Lim |
| **DOCUMENT No:** | SPIRE-RAL-DOC-002437 |

| | | | |
|---|---|---|---|
| **ISSUE:** | **Draft 0.3** | **Date:** | **19th October 2007** |

| | |
|---|---|
| **APPROVED BY:** | **Date:** |

# Distribution

**SPIRE ICC**
Bruce Swinyard
Matt Griffin

# Change Record

| ISSUE | DATE | Changes |
|---|---|---|
| Draft 0.1 | 08th September 2005 | First Draft |
| Draft 0.2 | 25th November 2005 | Added more detail to spectrometer pipeline, minor updates to flowcharts |
| Draft 0.3 | 19th October 2007 | Major update to account for the two new ADs (AD1 and AD2) |

## TABLE OF CONTENTS

<u>**FIGURES**</u>

<u>**TABLES**</u>

**Project Document**

**SPIRE Pipeline Description**

| Ref: | SPIRE-RAL-DOC-002437 |
|------|------|
| Issue: | Draft 0.1 |
| Date: | 19th October 2007 |
| Page: | 7 of 30 |

## Glossary

| | |
|------|------|
| ADU | Analogue Data Unit |
| | The (usually 16 bit) value returned from the A-D converter in the instrument electronics. |
| Frametime | Each SPIRE frame contains a 32 bit value which contains the number of (3.2 microsecond) clock ticks since the last synchronisation time, Tsync. In order to process the data from an observation during which several synchronisations may have been made, this value needs to be converted to an absolute on-board time. |
| On-board Time | The absolute time reported by the spacecraft on-board computer (CDMS), to which the instrument time is synchronised. This may be converted on the ground into UTC or any other time system. |
| BSM | Beam Steering Mechanism |
| SMEC | Spectrometer Mechanism |
| DRCU | Detector Readout Control Unit |
| PCAL | Photometer Calibrator |
| SCAL | Spectrometer Calibrator |
| UT | Universal time |
| ADU | Digital Engineering Units |

# 1. INTRODUCTION

## 1.1 Scope

This document aims to give a high level description of the SPIRE pipeline. The aim is describe the flow of data from data frames into level 0, 1 and level 2 products.

The pipeline can be run in an automated way. When run in this way, the pipeline is referred to as 'Standard Product Generation' (SPG). The pipeline can also be run in an interactive way. The flowcharts in this document are split into processing steps denoted by rectangular boxes. In interactive mode, a pipeline user should be able to inspect the data after each processing step.

## 1.2 Structure of Document

The document has four major sections, the photometer pipelines, the spectrometer pipelines, supplementary pipelines and calibration pipelines.

The photometer pipeline section has three minor sections on the common early processing (level 0 to level 0.5), scan map processing and jiggle map processing. For the spectrometer there are two sections relating to early processing then processing scanned observations. As step and look processing has not yet been specified no description is available. The supplementary pipelines currently include the Tsync processing and the Offset processing, trend analysis processing such as PCAL history will be added in a later version. Similarly the calibration processing is not yet defined in detail and hence will be added in a later version.

## 1.3 Documents

### 1.3.1 Applicable Documents

| | | |
| --- | --- | --- |
| AD1 | The SPIRE Analogue Signal Chain and Photometer Detector Data Processing Pipeline | SPIRE-UCF-DOC-002890 |
| AD2 | SPIRE Spectrometer Pipeline Description | SPIRE-BSS-DOC-002966 |

### 1.3.2 Reference Documents

| | | |
| --- | --- | --- |
| RD1 | SPIRE Data Products Specification | SPIRE-RAL-DOC-002005 |
| RD2 | DCU Design Description | SAP-SPIRE-FP-006302 |
| RD2 | Function Guide for the Fourier Transformation Package | SPIRE-UOL-DOC-002496 |

## 1.4 General Approach

Each pipeline described in this document consists of a series of processing steps and is presented in the form of a flowchart showing the links between processing steps and output data products. It is assumed that the 'unit' of data dealt with for most processing steps is a building block. However it should be noted that many of the processing steps will be done multiple times (e.g. for a nodding observation each nodding position will be processed as a building block). For latter stages more than one building block of reduced data may be used (e.g. de-nodding photometer data). Therefore each flowchart is a simple representation of the pipeline, which may be complicated in practice.

No attempt has been made in this document to describe actual data formats, a standard terminology has been applied where a piece of data is referred to as a 'file' and a processing stage is referred to as a 'process'. It should be noted that these terms are defined for this document only and the physical reality of the data could be any format. However one thing to note is that while the processes within each purple box in the flowchart may be self contained, when running interactively we should be able to stop the processing at any stage between two purple boxes. If the processing is broke in this way a user should be able to access an intermediate product encompassing the data processed through to that stage i.e. while not actually present the flowcharts could be considered as having intermediate products produced at each stage. After inspection the IA user may either continue with standard processing, do steps interactively or use their own processing.

## 1.5 Flowchart Colour Scheme

| | |
|---|---|
| | Process which changes the format of the data product |
| | Process which does not change the format of the data product |
| | Calibration input |
| | Calibration information produced by the pipeline then used later in the pipeline |
| | Level 0, 1 or 2 standard products |
| | Intermediate products |

## 2. PHOTOMETER PIPELINE

## 2.1 Stages between Level 0 and Level 0.5
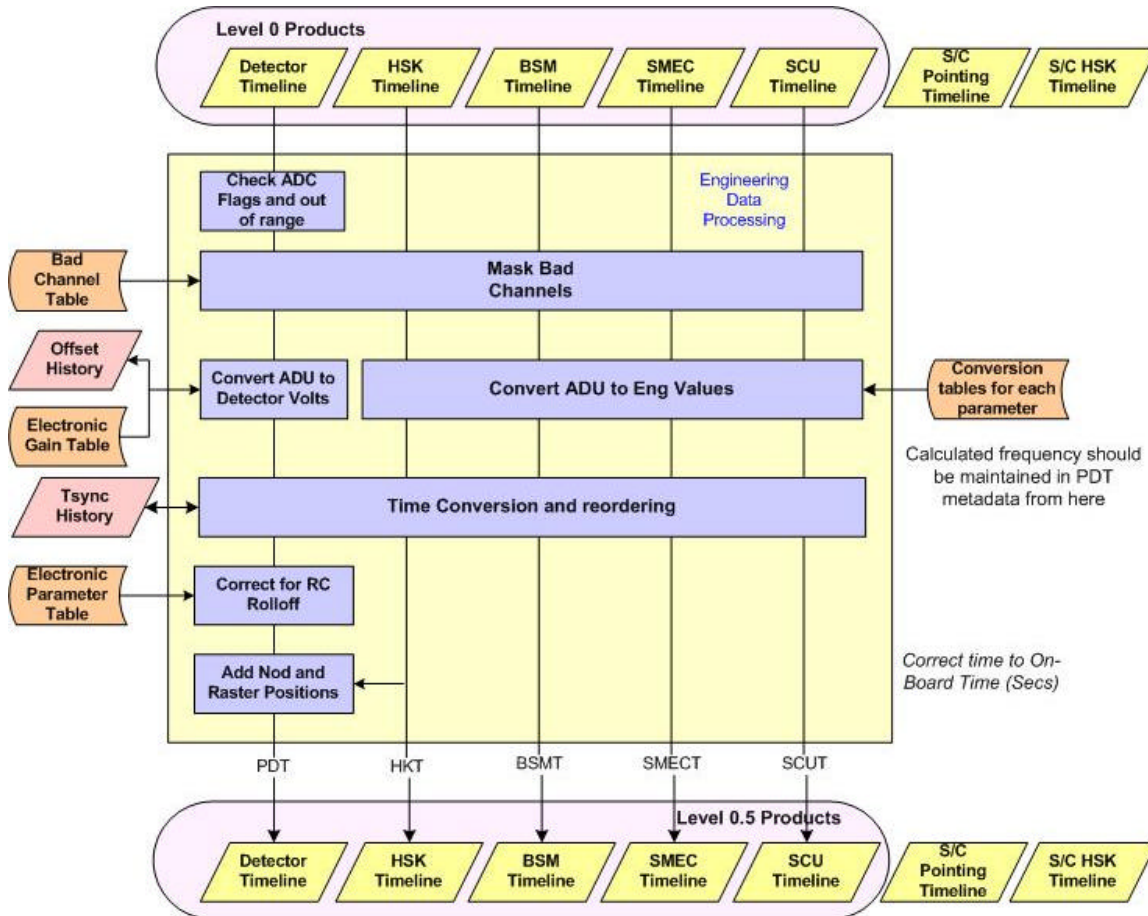


### 2.1.1 Level 0 Products

The starting point for the pipeline is data frames which, in the SPIRE case, consist essentially of unpacked telemetry packets (a single TM packet may contain one or more data frames as generated by the DRCU). These are combined into timelines containing of all the data requested. They will consist of a frametime in counts plus a single value for each parameter in the frame strung together. Each parameter is still in ADUs. The frames do not necessarily arrive time ordered and it is not envisaged to re-order them at this stage. Each detector channel will at this point be clearly labelled with a detector identifier which will be used in subsequent steps to link with calibration tables used by the processing.

At this point the timelines use frametimes as their time reference and a single frametime is used for all the data in a frame.

### 2.1.2  Check ADC Flags

The Analogue to Digital Converter Flags, stored in the *ADCFLGS parameters of both photometer and spectrometer data has a specific meaning in that it informs the DPU of occurrence of analogue to digital converter latch-up or S/S to DPU interface fault and then of the risk to pack corrupted data.

When a flag or more is set the related data of the current frame have to be discarded during on-ground data processing. This check is done by the EDP during the creation of the detector timeline product, i.e. in the Select box, just after the "reformat to timeline product".

### 2.1.3  Mask Bad Channels

Here the masking of channels currently only refers to dead channels, which can not be used in subsequent processing. This includes both detector channels and channels from other telemetry.

The calibration file design has a integer mask structure which can be extended beyond the dead pixel mask defined so far.

### 2.1.4  Spacecraft Processing

The spacecraft data passes through this flowchart unchanged. The reason for this is that the spacecraft data is supplied in physical units (RA, dec, temperatures etc) with a timeline already in UT therefore no low level processing is required.

### 2.1.5  Convert ADU to Detector Volts

Described in AD1 equation 17.

### 2.1.6  Convert ADU to Engineering Values

This is done for each housekeeping parameter by applying an ascii conversion table originally developed for QLA.

### 2.1.7  Time Conversion and Re-ordering

The starting point for this step is frametime is converted to on-board time by calculating the time in microseconds since the last Tsync and adding this to the on-board time of the last Tsync found from the Tsync history. This gives each sample in the timeline an absolute timestamp. Re-ordering can now take place as each data sample in a timeline has an associated absolute time.

For detector frames there is an additional time correction due to the fact that there is delay between the samples of successive pixels relative to the frametime. Therefore this step would take a table indicating the pixel readout order and the time delay for that pixel and then add this to time calculated for that frame. Each pixel would require its own timeline, this may be accounted for in the timeline product or it may imply a change of product at this stage. Although necessary for the spectrometer it is not clear that this step is required for the photometer as there is no obvious benefit from doing this. The calibration file to do this will be provided for users in case they wish to apply it.

### 2.1.8  Correct For RC Roll Off

The detectors are operated with an AC bias therefore have a time constant as a result of their resistance and the capacitance in the detector harness. The detector response is therefore a function of bias frequency. The RC Roll-off correction is described in equations 20-23 in AD1.

### 2.1.9 Add Nod and Raster Positions

This process associates a nod and a raster position with the building block. If we maintain the baseline of single building block processing we can assume we will have one telescope pointing per building block therefore the nod id and the raster id can be added to the metadata. Note this is the addition of identifiers rather than actual positions. In the case of non-nodding or non-rastering the expectation is that the identifier fields in the meta data will reflect this.

## 2.2 Scan Map Processing

**Project Document**

**SPIRE Pipeline Description**

Ref: SPIRE-RAL-DOC-002437
Issue: Draft 0.1
Date: 19th October 2007
Page: 14 of 30
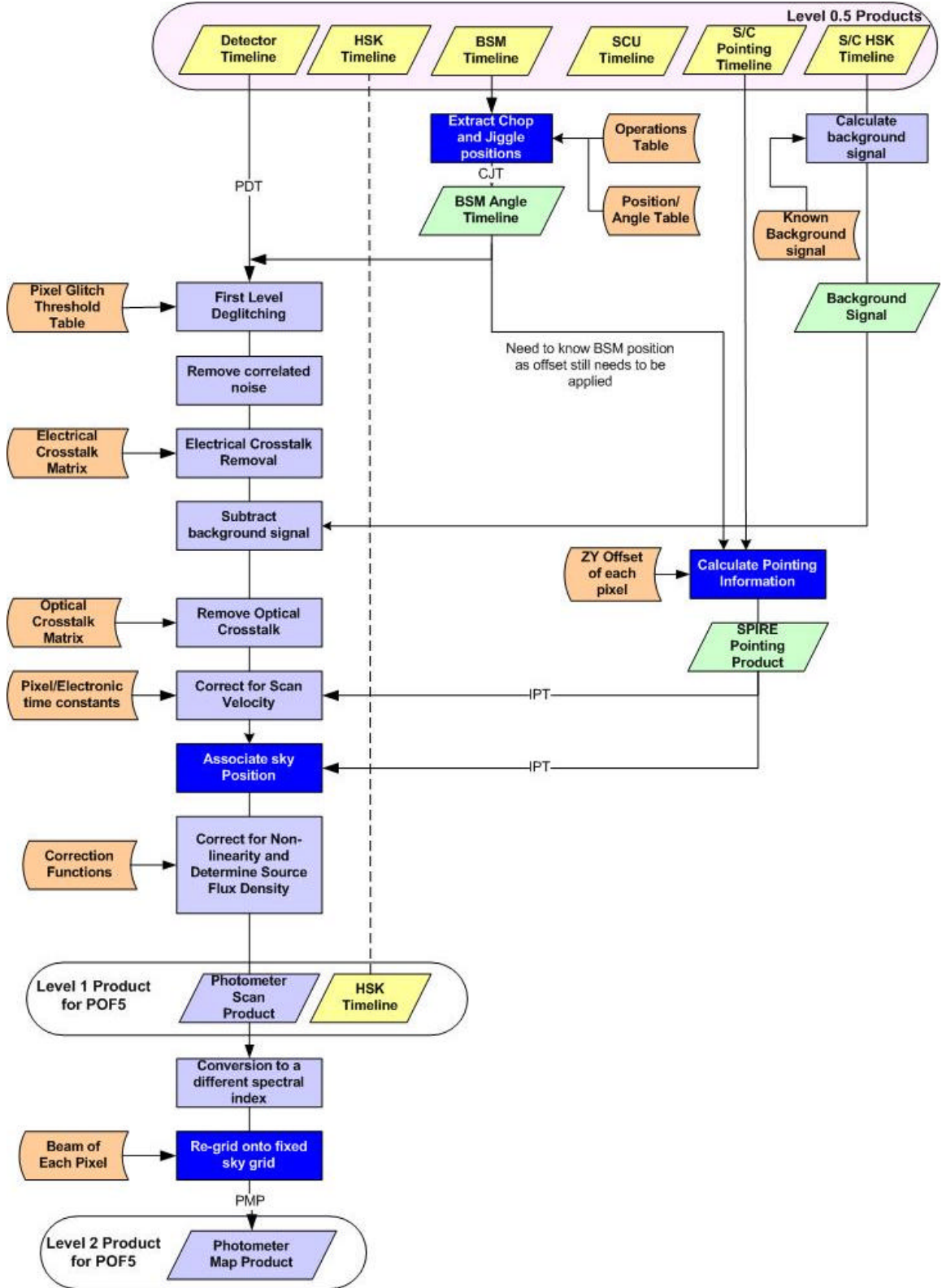
### 2.2.1  First Level Deglitching

The deglitching algorithm to use is TBD.

### 2.2.2  Remove Correlated Noise

There are several approaches to this (see AD1 section 4.2.2) but the adopted baseline is to use the correlations between the detectors and the array thermometers or blind pixels.

A voltage timeline $V_{th}(t)$ is generated for each detector by a TBD algorithm using the array thermometers and/or blind pixels. This is then subtracted from that detector's signal timeline. The thermometer signals must be averaged over a period of time to reduce the noise level, and thermal fluctuations on shorter timescales will not be tracked.

### 2.2.3  Electrical Crosstalk Removal

Electrical crosstalk is due to capacitative or inductive coupling between the detector readout channels and can be removed if the coupling between the detectors is known

We assume that
  (i)      electrical crosstalk is linear, so that the effects can be characterised by a crossstalk matrix, $C_{elec}$, with constant elements;
  (ii)     electrical crosstalk from one detector to another does not result in any diminution of the signal in theprimary detector;
  (iii)    there is no crosstalk between different arrays.

The vector of electrical crosstalk-corrected signals is given by

$$Q_2 = C_{elec}Q_1$$

The electrical crosstalk matrix will be implemented as a calibration file.

### 2.2.4  Calculate Telescope Background Signal

Standard calibration observations are carried out of "dark" areas of sky to produce a calibration file corresponding to the detector operating point voltage under the nominal conditions ("dark sky", known telescope and Level-1 temperatures).

A TBD algorithm is then implemented, which uses these calibration files and makes any small corrections for different telescope and L-1 temperatures to calculate $V_o$ for each detector for the particular conditions applying to the observation.

### 2.2.5  Subtract Background Signal

This module simply takes the calculated background signal $V_o$ then the signal power timelines ($V_s$) can then be calculated by subtracting this from the measured timeline $V$.

### 2.2.6  Optical Crosstalk Removal

Optical crosstalk is here defined as power that should be incident on one detector actually falling on another. It is important to note that in the case of SPIRE, neighbouring detectors are separated by an angle of $2l/D$ on the sky, and even if a source is on-axis for a given pixel, some fraction of the source power will be incident on the neighbouring detectors due to telescope diffraction. Non-neighbouring detectors are sufficiently far apart that they should not pick up any power from an on-axis source.

Optical crosstalk can be characterised by a crosstalk matrix, $C_{opt}$, analogous to the electrical crosstalk matrix described in Section 2.2.3.

The vector of optical crosstalk-corrected signals is given by

$$Q_{S2} = C_{opt} Q_{S1}$$

The optical crosstalk matrix will be implemented as a calibration file.

## 2.2.7  Correct For Scan Velocity

For scanning observations the detectors are experiencing a constantly changing input signal. They have a time constant which means that the signal recorded may be greater or less than the signal they should record due to the illumination history i.e. they may lag a rising slope by showing less signal and a falling slope by showing more signal. The degree of this signal change will depend on the telescope velocity. Therefore a correction needs to be made depending on the illumination history and the time constants. The time constants (in principle for each pixel) will be provided via a calibration file.

## 2.2.8  Extract Chop and Jiggle Positions

This step will produce a file which will contain the jiggle markers and chop markers in a time series. The file will also contain the converted chop and jiggle position to the relative offset in angle (in degrees on the sky) on the array of the BSM from the SPIRE boresight.  This is needed because the BSM zero position is not at the boresight (centre of the array).

It should be noted that the BSM timeline runs a lot faster than the detector timeline therefore will contain more points per time interval (typically by about a factor 5).

## 2.2.9  Calculate Pointing Information

The pointing derivation is based in instrument coordinates.

1.  We start with the pixel offset table which will be defined as angular offsets in instrument coordinates
2.  The offset of the BSM from the SPIRE boresight then needs to be added to each pixel offset.
3.  This angular offset of each pixel in instrument coordinates is now referred to the SPIRE boresight. It is assumed that the position of the SPIRE boresight on the sky is known directly.
4.  This can then be translated into sky coordinates using the spacecraft roll angle.

We should consider that the pixel offsets from the boresight assume a stationary telescope. The fact that not all pixels are read out at the same time means that the time difference between readouts should also be folded in to the pointing correction. I'm assuming for now that this correction is so small we don't need to worry about it.

Once the position is calculated, this step also needs to calculate the telescope slew velocity, the easiest way to do this is probably to calculate angular distance moved between each step then it is TBD whether this is sufficient for the next step assuming constant time samples or whether the delta time needs to be taken into account.

It is assumed that the output of this step is still referred in time to in the input pointing timeline which is more sparsely sampled than the detectors.

### 2.2.10  Associate Sky Position

This module simply interpolates the sky position timeline on to the detector timeline.

### 2.2.11  Correct For Non-Linearity and Determine Source Flux Density

Two adjustments are required to correct the detector signal.

- (i)   correction for detector non-linearity when observing strong sources;
- (ii)  correction for any drifts in detector responsivity.

For an NTD bolometer with a given applied bias voltage, the small-signal responsivity varies with the voltage across the bolometer with an approximately linear relationship over a wide range background loading conditions. This translates to a corresponding relationship for the differential sensitivity of the system to astronomical source flux density:

$$\frac{dS_S}{dV_{d-RMS}} = f(V_{d-RMS})$$

where $f(V_{d-RMS})$ may depart from linearity. The function $f(V_{d-RMS})$ is specific to a particular bias setting, and can be determined by pointing the telescope at a selection of bright sources (without chopping) and measuring the relative change in responsivity as a function of bolometer voltage using PCal. Except for one primary calibrator, it is not necessary to know how bright the sources are – they are just being used to establish a range of backgrounds on the detectors.

To calibrate observations of strong sources, the flux density can be determined by integrating the above expression between the off source (blank-sky) RMS detector voltage, $V_o$, and the RMS detector voltage when observing the source, $V_{so}$:

$$S = \int_{Vo}^{V_{SO}} f(V_{d-RMS})dV_{d-RMS}$$

### 2.2.12  Photometer Scan Product (Level 1)

The photometer scan product then consists of a set of timelines of pixel pointings in sky coordinates with an absorbed power per standard beam associated with that pointing at the readout time of the pixel.

### 2.2.13  Conversion to a Different Spectral Index

A standard set of spectral indexes will be represented in a calibration file along with the colour correction factors to apply for each one. This will not be done in standard processing but may be done by an astronomer who has knowledge of the spectral index of the source.

### 2.2.14  Re-Grid on to Fixed Sky Grid

Following a study of various algorithms the MADMAP algorithm has been adopted.

## 2.3 Photometer Jiggling Observations
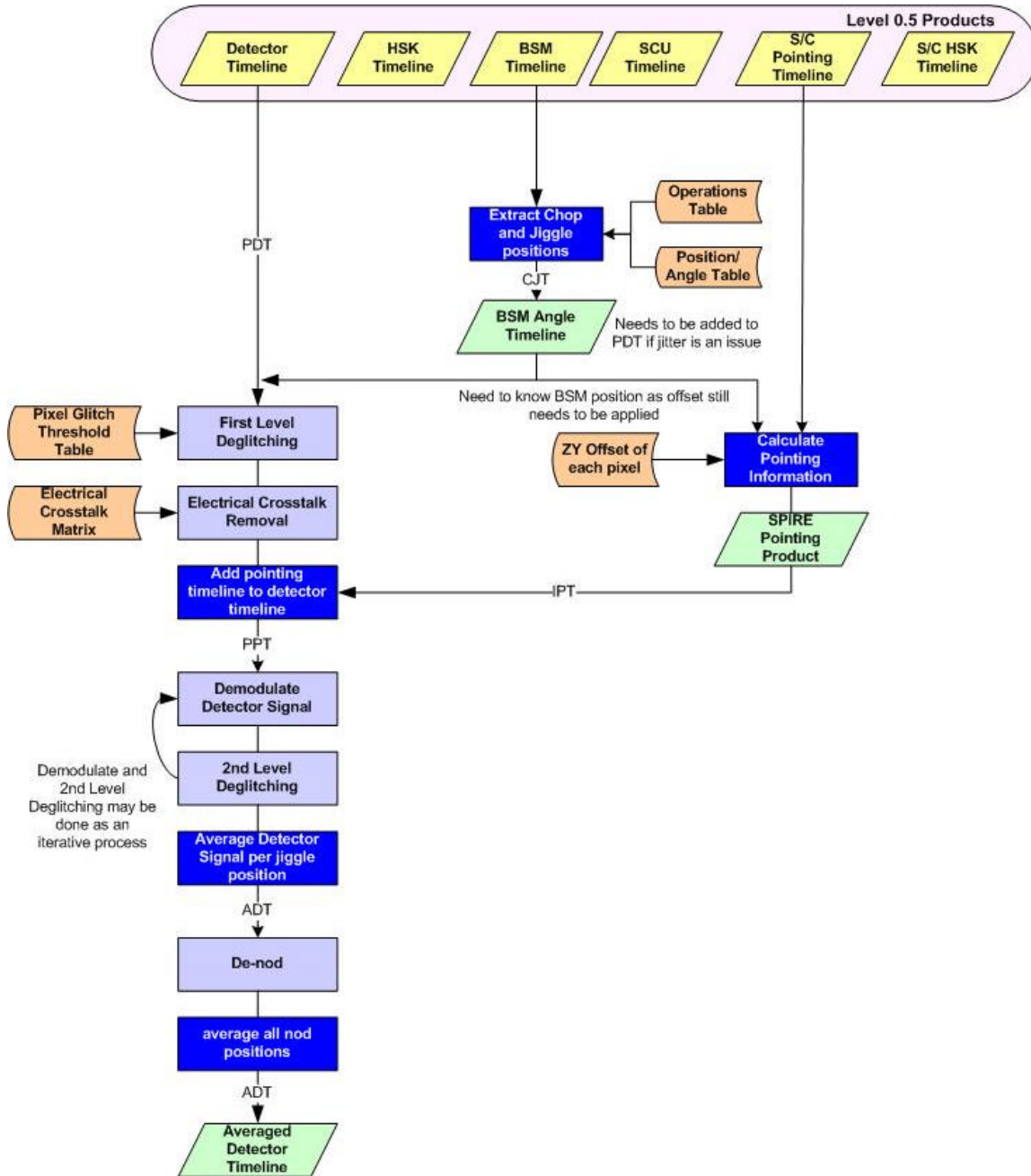
### 2.3.1 Extract Chop and Jiggle Positions

A single building block will contain a set of jiggle pointings. In order to do first level deglitching, the chop and jiggle markers are needed.

This step will produce a file which will contain the jiggle markers and chop markers in a time series. The file will also contain the converted chop and jiggle position to the relative offset in angle (in degrees on the sky) on the array of the BSM from the BSM zero position. This will be offset in spacecraft Y and Z direction which is aligned with the SPIRE FOV.

For point source observations there will be 2 detectors where a source will appear. It is assumed that these detectors are flagged at this stage but the mechanism for doing this is TBD. We could implement a calibration product which gives this information but it would be more desirable if the pipeline could obtain this from the uplink system.

It should be noted that the BSM timeline runs a lot faster than the detector timeline therefore will contain more points per time interval (typically by about a factor 5). It is assumed that allocating a chop position to a detector readout is done in the demodulation step.

### 2.3.2 First Level Deglitching

The deglitching algorithm to use is TBD.

### 2.3.3 Electrical Crosstalk Removal

This is implement with a crosstalk matrix as described in section 2.2.3.

### 2.3.4 Calculate Pointing Information

This is described in section 2.2.9.

### 2.3.5 Add Pointing Timeline to Detector Timeline

See section 2.2.10.

### 2.3.6 Demodulate Detector Signal

This is described in AD1 section 3.9.

### 2.3.7 Second Level Deglitching

For each jiggle position there should now be a timeline of points that should all contain the same value. This time line can be checked for outliers e.g. by median filtering or differentiation to find glitches not identified by first level deglitching. This step will then mask the points found. If deemed necessary second level deglitching may evolve methods for correcting for changes in signal induced by a glitch, however this is not expected in early pipeline versions. It is possible that in addition to masking the demodulated data, this process could also mask the product used as an input to the demodulation process allowing iteration between demodulation and second level deglitching. The threshold should be selectable by the user in the interactive case and for SPG a threshold could be set externally in a calibration file.

**Project Document**

**SPIRE Pipeline Description**

| | |
|---|---|
| **Ref:** | **SPIRE-RAL-DOC-002437** |
| **Issue:** | **Draft 0.1** |
| **Date:** | 19th October 2007 |
| **Page:** | 19 of 30 |

### 2.3.8  Average Data Per Nod and Jiggle Position

This outputs a single detector value per jiggle position. The two sky positions should also be attached. The reason this is a separate process is that users will want to inspect the data between the second level deglitching step and this step.

### 2.3.9  De-Nod

This step is the first pipeline step that cannot work on a single building block of detector data but instead must work on an observation. Each averaged pixel output at a nod position will contain the source minus a reference sky/telescope position for that nod position. The simplest way of de-nodding is to add the two positions i.e. if A1 B2 B3 A4 are the nodding positions then the signal is given by:

$$Signal = \frac{1}{2}\left(\left(A1 - B3\right) + \left(A4 - B2\right)\right) \tag{10}$$

(*Note For 7-point jiggling only do we only de-nod the central pixel as when we combine pixels the two either side will need to contribute just the demodulated data at one nod position?. For mapping modes I assume we de-nod all pixels.*)

### 2.3.10  Average all Nod Positions

We will also have several visits to the same nod position so these need to be averaged together. Again, as with each averaging step, outliers may be rejected at this stage.
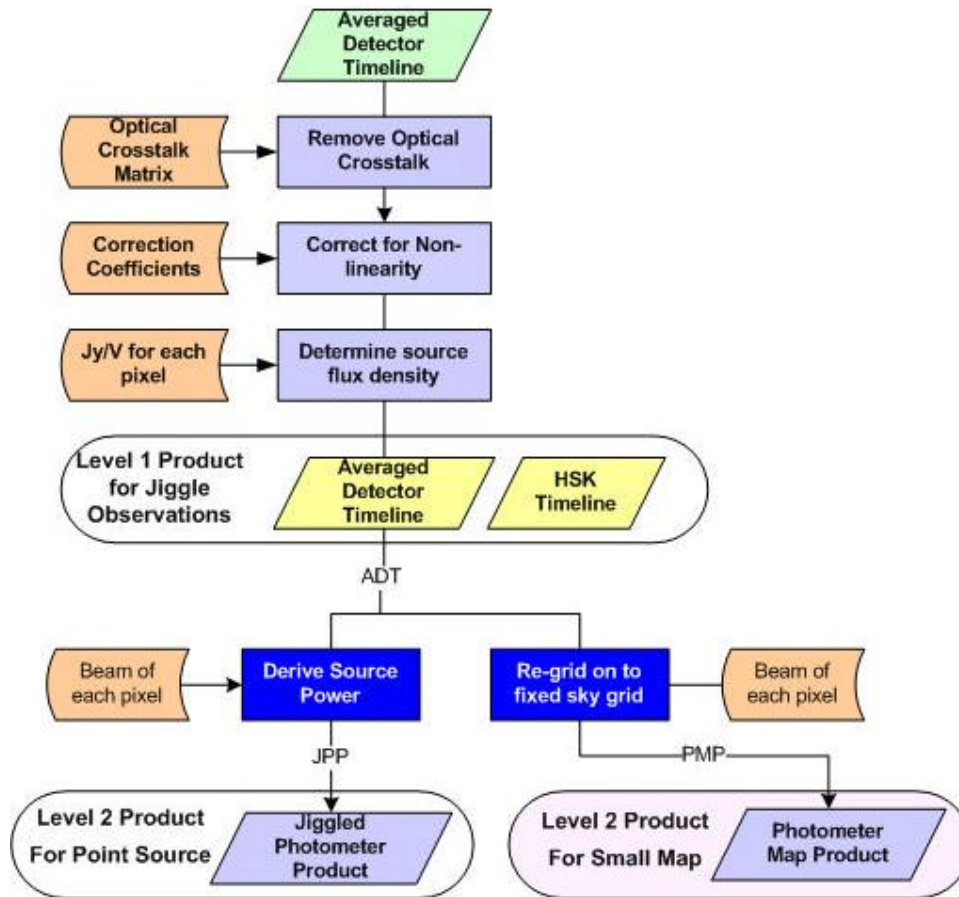
## 2.3.11  Remove Optical Crosstalk

This uses an optical crosstalk matrix to correct each pixel as described in section 2.2.6.

## 2.3.12  Correct for Non-Linearity and Determine Source Flux Density

See section 2.2.11

## 2.3.13  Level 1 Product For Jiggle Observations

The level 1 product contains the de-nodded signal per jiggle position.

## 2.3.14  Derive Source Power

For 7 point jiggle maps we now have 7 data points each referring to a jiggle position. As a 7-point jiggle map is for a point source, each of the position has been corrected for mis-pointing so the map should be fitted with a functional representative of the PSF. Some iteration may be required as the source may not be centred in the pixel.

## 2.3.15  Re-grid on to Fixed Sky Grid

As with scan mapping the MADMAP algorithm will be used.

## 2.4 Stages between Level 0 and Level 0.5
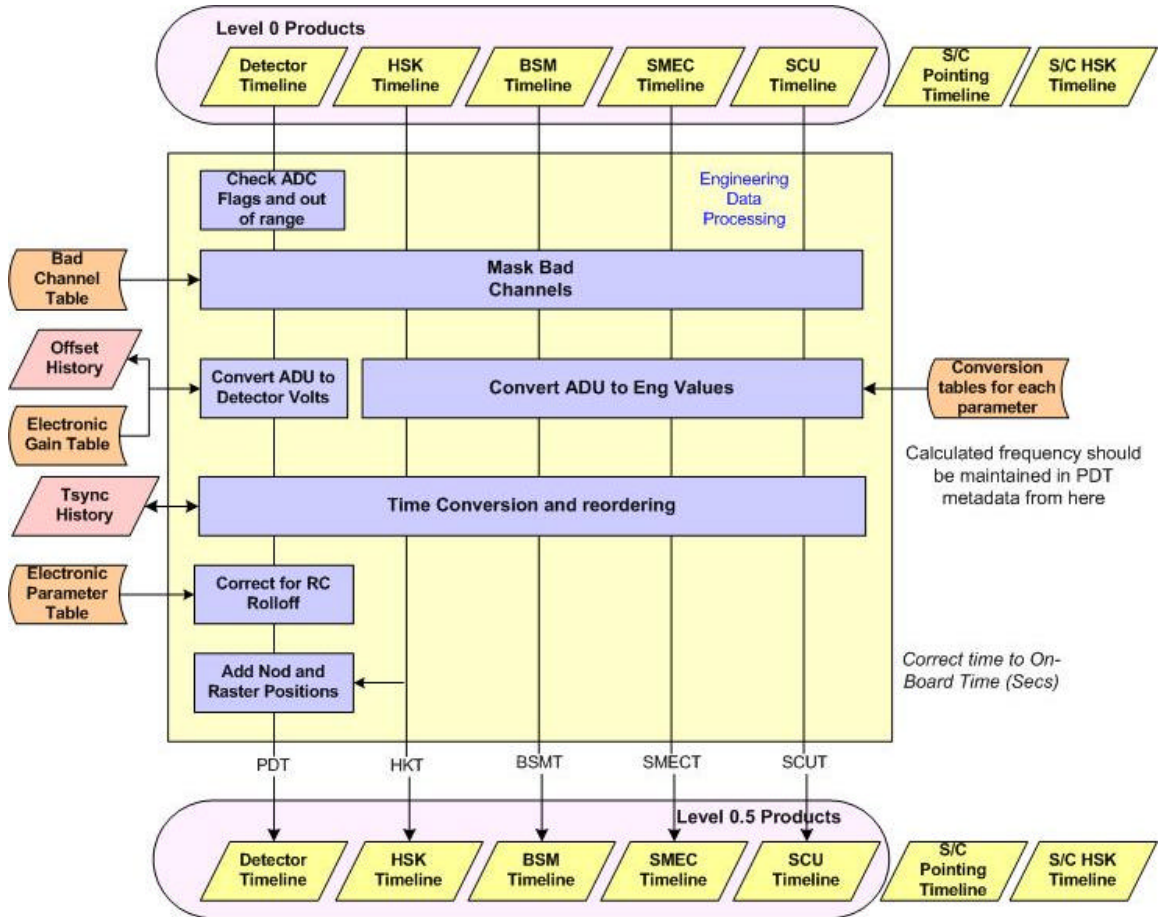


These stages are all the same as for photometer processing described in section 2.1.

## 2.5 Spectrometer Scan Processing

### 2.5.1 Clipping Correction

The purpose of this processing step is to correct for clipping of the measured signals due to the limited range of the detector ADCs. It is described in AD2 section 3.2.
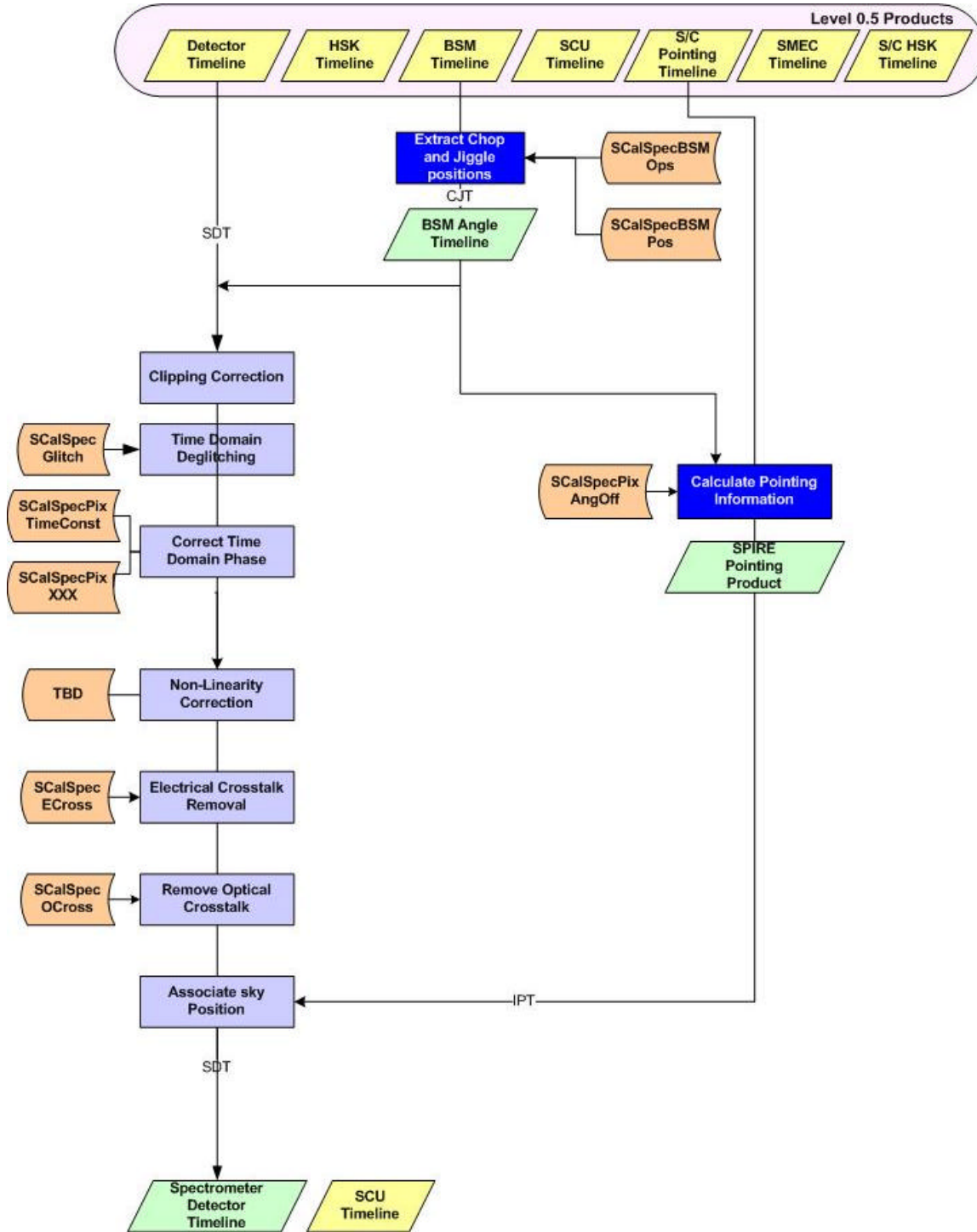
### 2.5.2 Time Domain De-Glitching

The algorithm for this is TBD.

### 2.5.3 Correct Time-Domain Phase

This is described in AD2 section 3.3.

### 2.5.4 Non-Linearity Correction

A correction will be required to account for changes in the responsivity of the detectors as a function of the intensity of the incident radiation. The details of this processing step are still TBD

### 2.5.5 Electrical Crosstalk Removal

This is described previously in section 2.2.3.

### 2.5.6 Optical Crosstalk Removal

This is described previously in section 2.2.6.

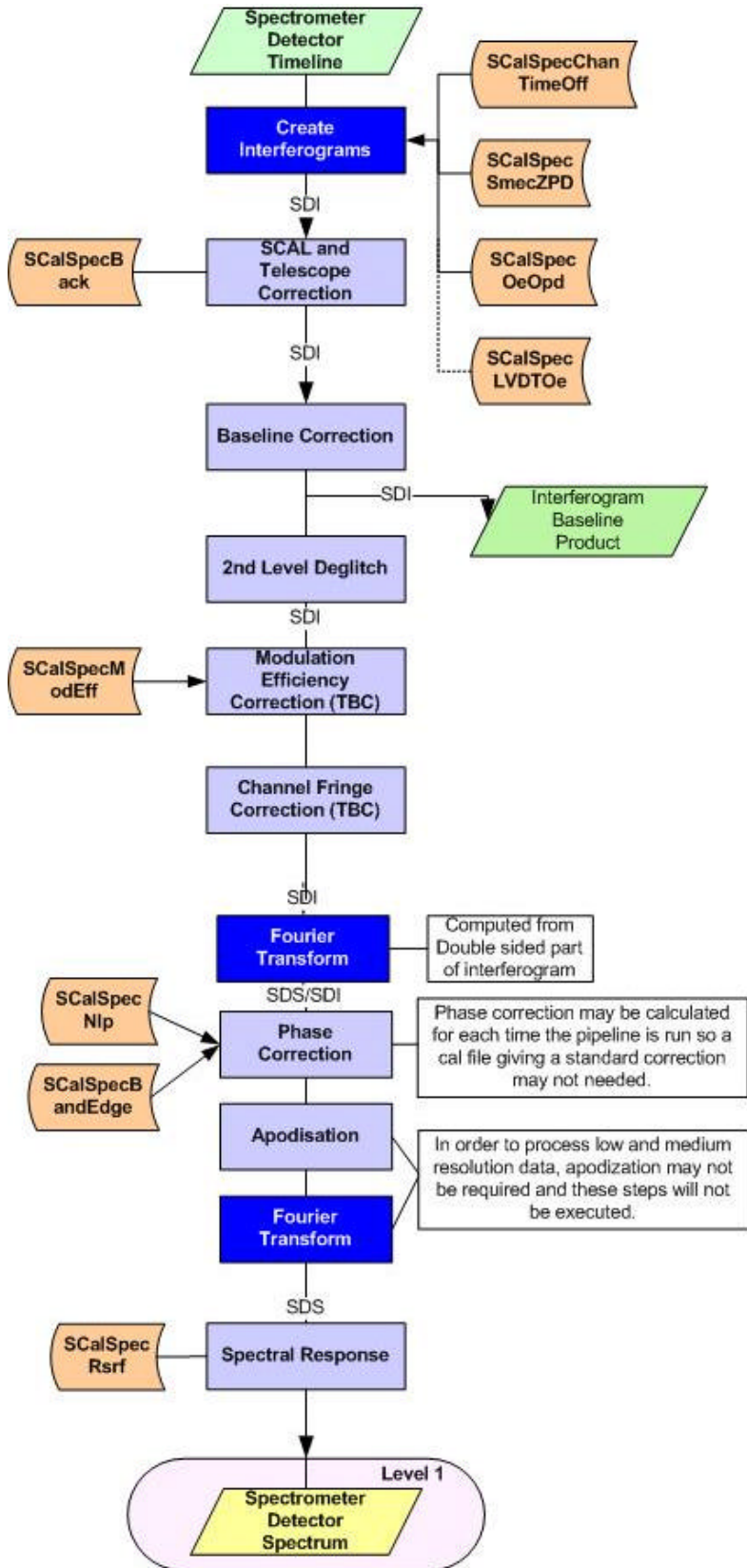### 2.5.7 Extract Chop and Jiggle Positions

This is described previously in section 2.2.8.

### 2.5.8 Calculate Pointing Information

This is described previously in section 2.2.9.

### 2.5.9 Associate Sky Position

This is described previously in section 2.2.10.

### 2.5.10  Interpolate to Regularly Sampled Interferograms

Parse scans from timelines' can be thought of as changing the baseline for the detector signals from time to SMEC mirror position.  This is accomplished with two interpolations. The following symbols are defined as:

| | |
|---|---|
| I | Detector signal |
| t, t', t'' | Time at which a particular value is sampled |
| z | Irregularly-spaced SMEC position grid |
| x | Regularly-spaced position grid |

The goal at the end of the 'Parse scans from timelines' is to produces a  set of interferograms, each of which is sampled on the same regular grid in optical path difference, i.e. I(x).

The Spectrometer Detector Timeline contains I(t) and the SMEC Timeline, which contains z(t`).  The Housekeeping timeline is also needed because it contains a parameter that delineates between the different scans in a group (SMECSTAT). The first interpolation in the 'Parse scans from timelines' step effectively changes the z-grid to an x-grid (irregular -> regular). Essentially, we interpolate to find the times when the SMEC mirror reached the regular positions in x.  Symbolically:

$$z(t`) \rightarrow x(t``)$$

The spectrometer detector timelines, I(t), are then interpolated onto the times when the SMEC was at the regular positions.  Symbolically:

$$I(t) \rightarrow \Gamma(t``)$$

The regular SMEC positions, x, and the interpolated detector samples, $\Gamma$, are combined into one interferogram since the t`` values are common.

$$\Gamma(t``), x(t``) \rightarrow \Gamma(x)$$

Jean-Paul has suggested just using the optical encoder for SMEC position and not LVDT (as shown by the calibration files used) but this left in for contingency. Pointing information will be added at this stage. Alternative ways of performing the interpolation may be possible but will provide the same functionality as the one described above.

### 2.5.11  SCAL and Telescope Correction

This is described in AD2 section 3.6.

### 2.5.12  Baseline Correction

As the SMEC moves away from ZPD the Interferogram is vignetted. The correction for this is described in AD2 section 3.7.

### 2.5.13  Level 2 Deglitching

This will use all interferograms to flag and remove outliers in standard deviation or skewness. What is then done both to flagged points from deglitching level 1 and level 2 is optional. Either the entire scan

needs to be removed or some way of filling in flagged points needs to be implemented. Presumably an average of the remaining interferograms will be used.

The deglitching is broken into two steps; the first step scans the set of interferograms in an effort to identify any glitches, the second step removes and replaces the glitches. One of the properties of the SDI products created by the Fourier Transform package is that the samples for each interferogram for a given detector pixel are all on the same position grid. As such, glitches can be detected by comparing the measured signal of each interferogram scan for a given pixel at each OPD position. Glitches are identified by flagging outliers of the scan-to-scan comparison at each sample point using metrics such as standard deviation or skewness. In general, standard deviation is more accurate for observations with a low number of scans, while the skewness technique works better for observations with a large number of scans ( = 10). After identifying the glitches present in the interferograms of the SDI product, the offending points are removed and then replaced. The values of the replacement points are taken as the average of the values from the unaffected interferograms at that OPD position.

### 2.5.14  Correct For Modulation Efficiency

As the SMEC moves away from ZPD the modulation efficiency decreases. A gain as a function of OPD per pixel must be applied to undo this effect. Respective calibration files have to be created and read in.

### 2.5.15  Channel Fringe Correction

Currently three algorithms are being evaluated. These are described in AD2 section 3.9.

### 2.5.16  Transform

The high resolution interferograms are then zero-padded (default) and Fourier transformed to the spectral domain.

### 2.5.17  Phase Correction

This is described in AD2 section 3.10.

### 2.5.18  Apodise

Apodisation, if required, removes the spectral artifacts related to the instrument line shape. This is only necessary for high resolution spectra. The natural instrument line shape for a Fourier Transform spectrometer is a cardinal sine or Sinc function. For interferograms that contain features that are at or near the resolution of the instrument (e.g. channel fringes), the natural instrument line shape can introduce artifacts in the calculated spectrum. These artifacts can be diminished by a technique referred to as apodisation whereby the interferogram is multiplied by a tapering or apodising function prior to transformation. A side effect of lowering the spectral artifacts is that the resolution of the resultant spectrum is also reduced.

In addition to some long-standing apodisation functions, the Fourier transform package makes available a number of functions that optimize the tradeoff between reduction in the ringing artifacts and reduced resolution. The apodisation function is designed to accept a spectrometer detector interferogram (SDI) product as input. On a scan-by-scan and detector pixel-by-pixel basis, the interferograms in the input SDI product are multiplied by the chosen apodisation function. The result is an SDI product that contains apodised interferograms.

### 2.5.19  Transform

TBW

**Project Document**

**SPIRE Pipeline Description**

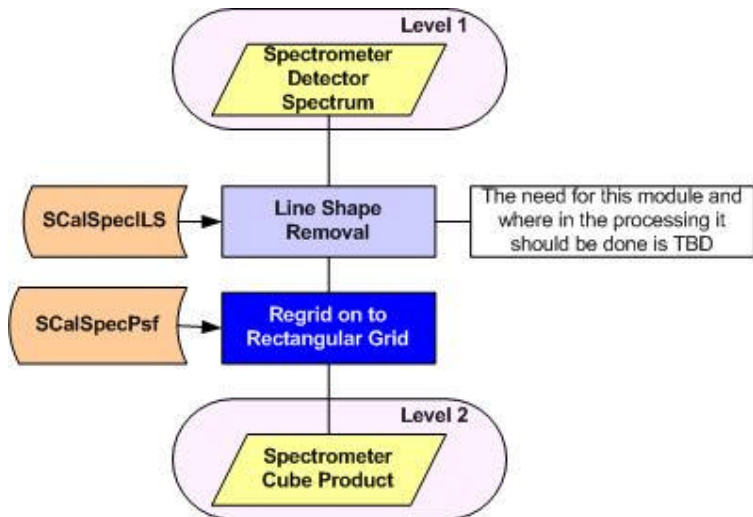| Ref: | SPIRE-RAL-DOC-002437 |
|---|---|
| Issue: | Draft 0.1 |
| Date: | 19th October 2007 |
| Page: | 27 of 30 |

### 2.5.20 Remove RSRF

The instrumental relative spectral response function then needs to be removed to give the spectral signal from the source. These will be stored in a calibration file.

### 2.5.21 Spectrometer Level 1 Product
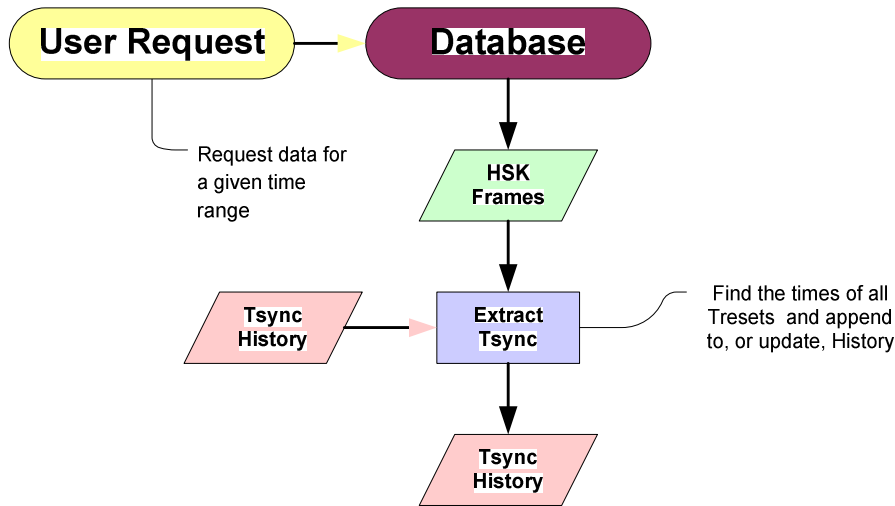
TBW



### 2.5.22 Line Shape Removal

TBW

### 2.5.23 Regrid on to Rectangular Sky Grid

TBW

## 3. SUPPLEMENTARY PIPELINES
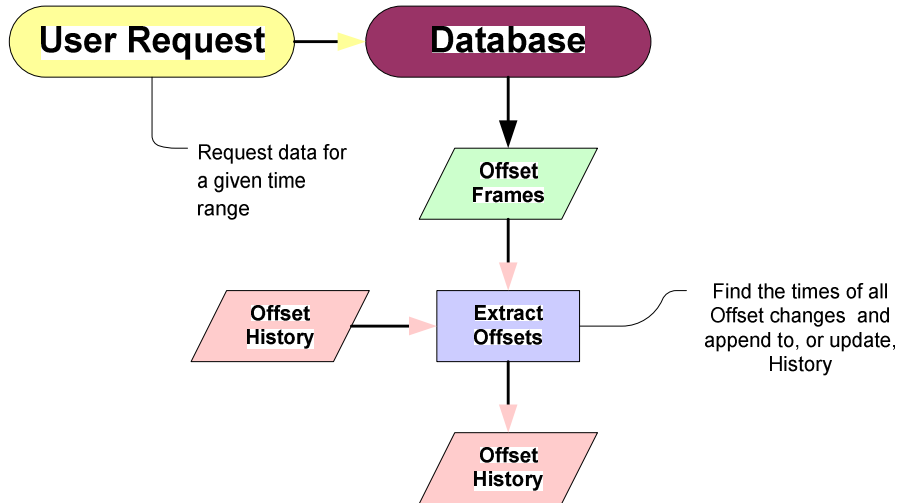
## 3.1 TSYNC History Processing Pipeline



### 3.1.1 Extract Tsync Process

Each SPIRE frame contains a 32 bit value which contains the number of (3.2 microsecond) clock ticks since the last synchronisation time, Tsync. In order to process the data from an observation during which several synchronisations may have been made, this value needs to be converted to an absolute on-board time.

Time synchronisation operations are inserted into the timeline at appropriate points in the instrument operations in order to prevent the counter rolling over (i.e at intervals of less than 229 minutes) but it may also be the case that the first synchronisation time occurs before the start of the observation.

This pipeline is used to make these Tsync times available to the data processing by creating a single Tsync History which will eventually cover the whole mission. It is expected that this pipeline will be run at appropriate intervals (e.g. between cooler recycles), to extract the Tsyncs for a given time period, and add these into the History. It should be noted that this will require the Tsync History pipeline to be run over the time period covering an observation before the normal pipeline processing can be run.

**Project Document**

**SPIRE Pipeline Description**

Ref: SPIRE-RAL-DOC-002437
Issue: **Draft 0.1**
Date: 19th October 2007
Page: 29 of 30

## 3.2 Offset History Processing Pipeline



### 3.2.1 Extract Offset Process

The detector signal has a DC offset applied to keep the detector signal within the dynamic range of the electronics. These offset values are set (usually automatically at the start of an observation), then downlinked in telemetry on request. There is no guarantee that the offsets are downlinked during an observation, although it is planned. This pipeline will be run at regular intervals to extract the offset information and create a history of offset values for the whole mission.

**Project Document**

**SPIRE Pipeline Description**

Ref: SPIRE-RAL-DOC-002437
Issue: Draft 0.1
Date: 19th October 2007
Page: 30 of 30

## 4. CALIBRATION PIPELINES

TBW