SUBJECT: **SPIRE Pipeline Description**

PREPARED BY: **Chris Pearson (Custodian)**

With input from: Tanya Lim, Matt Griffin, Darren Dowell, Pasquale Pannuzo, Ed Polehampton

DOCUMENT No: **SPIRE-RAL-DOC-002437**

ISSUE: **Issue 1.0** **Date:** **2 August 2008**

APPROVED BY: **Date:**

Distribution

SPIRE ICC

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)
Draft 0.4	November 2007	Revision and extension by Matt to provide more details of algorithmic content of modules and contents of corresponding calibration files. Only the photometer section has been revised.
Draft 0.5	December 2007	Further revisions taking into account comments from Pasquale and updates to the Photometer pipeline document (AD1)
Draft 0.6	February 2008	Updates taking into account revision of AD1; more information on pipeline algorithms and calibration files. No updates to FTS sections.
Draft 0.7	15th February 2008	Working version based on meeting at RAL on Feb. 15. No updates to FTS sections.
Draft 0.7	20th February 2008	First Draft after meeting at RAL on Feb. 15 to be distributed for comments
Draft 0.8	March 2008	Revised Draft after initial inputs
Draft 0.9	14th March 2008	Updated Flowcharts and Calibration Files Added
Draft 1.0	27th March 2008	Many sections revised following meeting at RAL 26th March
Draft 1.1	17th April 2008	Calibration Products Updated
Draft 1.2	18th June 2008	Update from June ICC Meeting
Draft 1.3	16th July 2008	Updated pointing and OPD information. Update SPEC flowchart
Draft 1.4	21st July 2008	Various edits and additions by Matt Griffin
Draft 1.5	23rd July 2008	Updated after Cardiff Meeting
Draft 1.6	1st August 2008	Spectrometer Section Added
Issue 1.0	2nd August 2008	First issue version for GSRR

TABLE OF CONTENTS

1. INTRODUCTION

1.1 Scope

This document aims to give a high level description of the Herschel Common Science System (HCSS) SPIRE photometer and spectrometer pipelines. It will be used as main prescriptive document for pipeline module implementation for pipeline developers, and will therefore be an applicable document for all pipeline module design documents. Note, that this document is not meant as a programming guide for developers and it is expected that developers follow HCSS standards (RD4). It describes the flow of data from data frames into level 0, level 1 and level 2 products.
More detail on the pipeline algorithms themselves and the rationale for adopting them can be found in AD1 and AD2. The document AD1 describes the processing in the on-board electronics for both photometer and FTS pipelines, and explains the logic and sequence of photometer pipeline modules, providing the definition of the algorithms to be implemented in each module serving as an input to this document. The document AD2 is the equivalent document for the FTS pipeline.

The SPIRE pipelines can be run in an automated mode. When run in this way, the pipeline operation is referred to as 'Standard Product Generation' (SPG). The pipelines are also designed to 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 the following major sections:

(i) Description of the pre-processing to Level 0 Products
(ii) Front-End pipeline (Level 0 to Level 0.5 processing) which covers steps up to the calculation of detector voltage and resistance, and is algorithmically common to the photometer and spectrometer instruments, differing only in the parameter values;
(iii) Photometer pipelines;
(iv) Spectrometer pipelines;
(v) Operational Day processing;
(vi) Calibration pipelines.

The photometer pipeline section covers scan map and jiggle map processing. For the spectrometer there are two sections relating to early processing then processing scanned observations. The supplementary pipelines currently include the Time Synchronization processing (Section 5.1) and the Offset processing (Section 5.2). the trend analysis processing such as PCal history will be added in a later version. The calibration processing is also yet to be defined in detail and will be added in a later version.

In this document, the pipeline steps are explained in turn giving a description of the process, the expected input and output and any associated calibration data. The expected input and output data products are defined and the definition of Product Descriptions (tabulated in RD1) should be consistent with those in this document.

1.3 Applicable and Reference 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
AD3	SPIRE Data ICD	SPIRE-RAL-PRJ -001078

1.3.2 Reference Documents

RD1	SPIRE Data Products Specification	SPIRE-RAL-DOC-002005
RD2	DCU Design Description	SAP-SPIRE-FP-0063-02
RD3	Function Guide for the Fourier Transformation Package	SPIRE-UOL-DOC-002496
RD4	Developers Manual: Herschel Data Processing	HERSCHEL-HSC-DOC-0625
RD5	Calibration products for SPIRE Data Processing	SPIRE-RAL-DOC-00261
RD6	Mask Handling in the SPIRE Pipeline	SPIRE-ICS-NOT-?????
RD7	SPIRE Data Frame Specification	SPIRE-PAD-NOT-002128
RD8	SPIRE DCU FM ACCEPTANCE document	SA-p-SPIRE-HT-0395-06
RD9	Housekeeping Conversion Tables Description Document	SPIRE-RAL-DOC-003113
RD10	SPIRE AOT Implementation Document	SPIRE-RAL-DOC-002663
RD11		

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. Each observation is made up of a series of 'building blocks' referred to by a BBID (defined in AD3) corresponding to a discrete operation of the instrument, and 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 implemented multiple times (e.g., for a nodding observations each nodding position will be processed as a building block). For later 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.

The following colour scheme is adopted:

	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

Data are propagated through the HCSS system in the form of "Products" and we maintain this as the standard terminology for the data flow. This document does not contain descriptions of actual data formats: descriptions of the data structure can be found in AD3, descriptions of the data products can be found in RD1 and a description of the calibration data in RD5.

The procedure for Standard Product Generation is divided into three successive steps:

1) Pre-processing;
2) Pipeline processing;
3) Post processing.

A processing stage is referred to as a 'module'. 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 each module (i.e. each box in the flowchart) may be self contained, when running interactively a user should be able to stop the processing at any stage between two modules. If the processing pipeline is interrupted in this way, a user should be able to access the intermediate product encompassing the data processed through to that stage. The flowcharts should therefore be viewed 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 List of Acronyms

ADU	Analogue Data Unit The (usually 16 bit) value returned from the A-D converter in the instrument electronics.
BAT	BSM Angles Timeline
BBID	Building Block ID the numbering of building blocks in an operation
BSM	Beam Steering Mechanism
CDMS	Command and Data Management System
CHK	Critical House Keeping
CUC	CCSDS Unsegmented Time Code
DAT	Detector Angles table
Data Frame Time	The time measured in sample time, in Fine Time values in microseconds since 1st January 1958. Also referred to as the sample time.
DRCU	Detector Readout Control Unit
DPU	Data Processing Unit
ERT	Earth Reception Time
Frametime	Each SPIRE frame contains a 32 bit value which contains the number of (3.2 microsecond) clock ticks since the last synchronisation time, Treset. 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.
HCSS	Herschel Common Science System
HPP	Herschel Pointing Product
HSK	House Keeping
ILS	Instrument Line Shape
NHK	Nominal House Keeping
On-board Time	The absolute time reported by spacecraft on-board computer (CDMS), to which instrument time is synchronised. This may be converted on the ground into UTC or other time formats..
OD	Operational Day
ODP	Operational Day Processing
OPD	Optical Path Difference
MPD	Mechanical Path Difference
PCAL	Photometer Calibrator
PCF	Phase Correction Function
packetTime	TM packet time
PDT	Photometer Detector Timeline
QLA	Quick Look Analysis
RNHKT	Raw Nominal Housekeeping Timeline
RSRF	Relative Spectral Response Function
SCAL	Spectrometer Calibrator
SCU	Sub-system Control Unit
SDI	Spectrometer Detector interferogram
sdfTime	SPIRE Data Frame Time
SDT	Spectrometer Detector Timeline
seqCount	Counter attached to each telemetry packet
SIAM	SPIRE Instrument Apertures Matrix
SMEC	Spectrometer Mechanism
SPG	Standard Product Generation
SPP	Spacecraft Pointing Product
TAI	International Atomic Time
TCO	Time Correlator
TM	Telemetry
Treset	Time since the last reset of the frame time in CUC format.
UT	Universal Time
UTC	Coordinated Universal Time
ZPD	Zero Path difference

1.6 List of Symbols

Symbol	Definition
a	Detector time constant amplitude factor
a_T	Constant for relation between Thermistor voltage and detector signal
A	Constant calculated from resistance and transconductance of detector channels
$B(\sigma)$	Spectrometer Spectrum as a function of wavenumber in volts
B_{DS}	Spectrum of double sided interferogram
B_{SS}	Spectrum of single sided interferogram
C_H	Capacitance of the harness between the detector and JFET input
b	BSM position
b_T	Coefficient for relation between Thermistor voltage and detector signal
c	Chop position Number
$\mathbf{C_{elec}}$	Electrical crosstalk matrix
$\mathbf{C_{opt}}$	Optical crosstalk matrix
$DATA$	16-bit ADC output value corresponding to a detector voltage value
G_{tot}	Total gain of analogue signal chain from JFET output to the ADC
$H_H(\omega_b)$	Transfer function of the harness between the detector and JFET input
$H_{JFET}(\omega_b)$	Transfer function of the JFET
K	Constant of proportionality relating source flux density to absorbed detector power
$I_{b\text{-}rms}$	RMS of bias current
$I(\sigma)$	Spectrometer Spectrum as a function of wavenumber in flux units
k	Nod cycle number
K_1	Coefficient of 1st-order correction to relationship between source flux density and V_S
K_2	Coefficient of logarithmic correction for relationship between source flux density and V_S
K_3	Constant of logarithmic correction for relationship between source flux density and V_S
j	$\sqrt{-1}$
j	Jiggle position number
n	Telescope nod position
L	Maximum OPD displacement from the position of ZPD
N_{nod}	Number of nod cycles at each jiggle position
N_{jig}	Number of jiggles
N_{chop}	Number of chops
$OFFSET$	4-bit offset used to generate offset voltage to be subtracted from LPF output voltage
R_d	Detector resistance
$R_{d\text{-}nom}$	Detector resistance when telescope views a blank sky
R_L	Total load resistance
s	Sample number within chop half cycle
S	Flux Density
t	Time
v_{SMEC}	Speed of SMEC
V_b	Bias voltage amplitude
$V_{b\text{-}RMS}$	RMS value of bias voltage
$V_{d\text{-}RMS}$	RMS value of voltage across detector
V_{JFET}	Amplitude of the voltage at the JFET output
V_o	Fixed bolometer offset voltage used in flux density conversion and linearity correction.
V_{oT}	Nominal detector array thermistor voltage.
V_r	Reference voltage for spectrometer linearity correction
V_S	Decrease in RMS detector voltage at the operating point due to the astronomical signal (Detector signal voltage)
V_T	Voltage across Thermistor
$\Delta V_{JFET\text{-}RMS}(1\text{-bit})$	Change in RMS voltage at the JFET output that corresponds to a 1-bit change in the value of $DATA$

$\Delta\phi$	Phase difference between demodulator reference and input signals
ϕ	Phase of double sided spectrum
σ	Wavenumber
τ_1	Detector nominal time constant
τ_2	Detector "slow" time constant
τ_H	Time constant defined by the JFET harness capacitance and the parallel combination of the detector and load resistances
$\tau_{H\text{-}nom}$	Detector-JFET harness time constant when looking at blank sky
ω_b	Angular frequency of detector bias voltage
$\omega_{b\text{-}ref}$	Reference frequency of detector bias voltage
ω_s	Angular frequency of detector signal modulation
x	Regularly spaced positions on interferogram
Y	Y angle along array long axis for BSM chop
Z	Z angle across array short axis for BSM chop

2. PIPELINE PRE-PROCESSING

2.1 Operational Day and Standard Product Generation

In the Standard Product Generation (SPG) framework, the pipeline processes data from individual observations. The necessary calibration products have already been created from the Operational Day Processing (ODP) stage (see Section 6). The observation context is already created and populated with the appropriate auxiliary data (e.g., spacecraft pointing) by the SPG software provided by ESA.

The SPIRE pre-processing consists of two steps:

(i) population of the observation context with appropriate calibration products;
(ii) population of the Level 0 products from the telemetry packets or data frames.

The first step attaches to the observation context the calibration products that are applicable to that observation. Some of these calibration products are independent of the observation (e.g., the channel gain table), while others have to be created from the data of the entire "operational day" or from the data of the observation itself. Calibration products that have to be created on an operational day basis (e.g. detector offset history) are produced during the Operational Day Processing before this stage.

2.2 Level 0 Products

The Level 0 Product is the starting point for the pipeline. A Level 0 Product is a set of data frames, each of which consists of unpacked telemetry data from telemetry (TM) packets (a single TM packet may contain one or more data frames as generated by the DRCU). These are then combined into timelines containing all of the data requested. Each timeline consists of a table with one row for each data frame and a column for each TM parameter. Each parameter is still in ADUs and the name of the column is the same name as used in the telemetry (e.g. "PHOTFARRAY123").

The *frametime* for each packet is the number of (3.2 μs) clock ticks since the last synchronisation time, Treset. It is converted to the *data frame time* by calculating the time in μs since the last Treset and adding this to the on-board time of the last Treset found from the Treset history. This gives each sample in the timeline an absolute timestamp. The data frame time is also referred to as sample time, and is measured in Fine Time values in μs since 1st January 1958, stored as a 64-bit integer.

In addition there is a column for the TM packet time and one for the TM packet sequence count. The frames do not necessarily arrive time-ordered, and at present it is not envisaged to re-order them at this stage of the processing.

The format of Level-0 data products is defined to be as simple as possible. Each product will contain data coming from only one Building Block of a specified Observation. Moreover, each product will contain data coming from only one TM packet type. Each product will contain only one TableDataset, with raw ADU and time columns identified with the name of the TM packet type; this table has a number of Columns, one for each quantity stored in SpireDataFrames of the specified TM packet, i.e. a column for each TM parameter contained in the specified TM packet. Level 0 Products as inputs to the pipeline are described in detail in RD1 and listed below;

- Raw Detector Timelines: Detector Signal timelines for all channels in the arrays
- Raw Nominal House Keeping Timeline: H/K data required to process the detector timelines
- Raw BSM Timeline: Beam Steering Mirror position sensor timelines
- Raw SCU Timeline: Values for Spacecraft Sub-Systems
- Raw SMEC Timeline: Spectrometer mirror position sensor timelines

- Raw Critical House Keeping Timeline: Timeline containing status of critical system values
- Raw Photometer Offset Timeline: Offsets for all channels in the Photometer arrays
- Raw Spectrometer Offset Timeline: Offsets for all channels in the Spectrometer arrays

3. COMMON PIPELINE STAGES: ENGINEERING CONVERSION FROM LEVEL 0 TO LEVEL 0.5 PRODUCTS

At this stage, the auxiliary pipeline has already been run and the auxiliary products (e.g., spacecraft pointing) are already in the observation context – i.e. the data have already been processed to the Level 0 stage (see Section 6).

The data flow up to Level 0.5 is shown in Figure 1 and described in detail below.

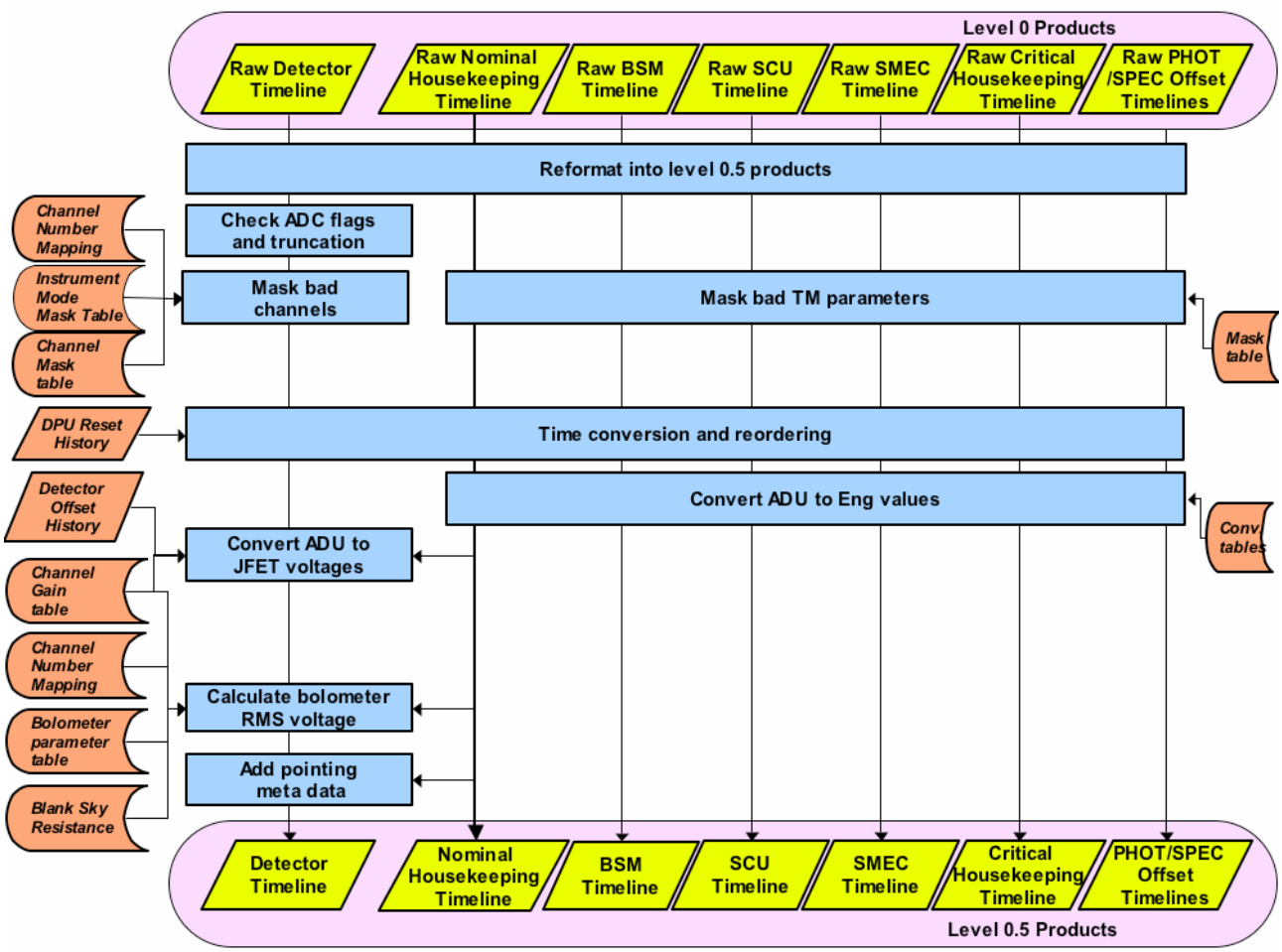

Figure 1: Flowchart for data flow from Level 0 to Level 0.5 products common to both the photometer and spectrometer pipeline processes.

3.1 Reformat Timelines into Level 0.5 Format

Description: This module is responsible for reformatting the Level 0 products into a more user-friendly and workable form. The input Level 0 products have a single table with the name of the packet type that originated them. Each column in this table has the name of the corresponding telemetry parameter (e.g. PHOTFARRAY123 for a channel). This module creates Level 0.5 products with tables for "signal" and a "mask". The first column is "sampleTime", and there are additional columns for each telemetry parameter. The columns use a more intuitive naming convention such as the real names of channels (e. g. “PSWA8” in the case of detector timelines). For other timelines (i.e. BSM, SMEC etc.)

the names of the columns are changed using telemetry dictionary lookup tables such that for example, "BSMCHOPMOTORCURR" is translated into "chopMotorCurr". Some telemetry parameters are not propagated into the reformatted products: these are normally those linked to the control of the transmission of telemetry packets and frames, e.g. the checksum, frame ID, etc.

Inputs: Level 0 product – a single table with the name of the packet type that originated it and with columns for raw detector timelines with channels names referred to by channel number.

Detector signal	Format	PDT (32 bit integer containing unsigned 16 bit integer)
	Units	ADU
Sample time	Format	64 bit integer containing the time from January 1st 1958
	Units	μs
Frame Time	Comments	For data frames only
	Format	32 bit integer containing unsigned 16 bit integer
	Units	Number of clock ticks since last DPU reset time

Calibration files: Telemetry dictionary calibration files contain information for the conversion of telemetry column names into the Level 0.5 style format. Some telemetry parameters are not propagated into the reformatted products and this is also specified by the calibration files.

Calibration files are found in the build /develop/main/herschel/spire/ia/pipeline/common/engdata/tables/)

Column 1: Telemetry name (e.g., BSMCHOPMOTORCURR)	Format	ASCII
Column 2: Level 0.5 style format (e.g. chopMotorCurr)	Format	ASCII
Column 3: Propagated to Level 0.5 style product (Y/N)	Format	ASCII
Column 4: Description of the parameter	Format	ASCII

Telemetry dictionaries:

PM_BSMNOMINAL – conversion of telemetry names for BSM Timeline
PM_NHK– conversion of telemetry names for NHK Timeline
PM_PHOTF– conversion of telemetry names for full Photometer Detector Timeline (all arrays)
PM_PHOTLW - conversion of telemetry names for only LW Photometer Detector Timeline
PM_PHOTMW- conversion of telemetry names for MW Photometer Detector Timeline
PM_PHOTSW- conversion of telemetry names for SW Photometer Detector Timeline
PM_PHOTOFF- conversion of telemetry names for photometer Detector Offset Timeline
PM_SCUNOMINAL - conversion of telemetry names for SCU Timeline
PM_SMECSCAN - conversion of telemetry names for SMEC Timeline
PM_SMECSELECT- conversion of telemetry names for SMEC Timeline
PM_SPECF – conversion of telemetry names for full Spectrometer Detector Timeline (all arrays)
PM_SPECLW – conversion of telemetry names for reading out LW spectrometer array
PM_SPECSW– conversion of telemetry names for reading out SW spectrometer array
PM_SPECOFF – conversion of telemetry names for spectrometer Detector Offset Timeline

Outputs: Level 0.5 style product with three tables for raw signal (including ADC flags), mask and time quantities. Channels are referred to by detector name. Engineering column names are also changed.

Detector signal	Format	PDT (32 bit integer containing unsigned 16 bit integer)
	Units	ADU
Sample time	Format	64 bit integer containing the time from January 1st 1958
	Units	μs
Mask table	Comments	Values are set to zero
	Format	32 bit integer
	Units	Dimensionless

3.2 Check ADC Flags and Truncation

Description: The Analogue to Digital Converter Flags in the output of the previous step are stored in the *ADCFLGS column. These flags have a specific meaning in that they inform the DPU of the occurrence of analogue to digital converter latch-up or Spacecraft Subsystem to DPU interface fault and therefore of the risk to propagate corrupted data. When one flag or more is set, the related data have to be masked so that they will not be used in the following data processing steps. The occurrence of this error will be stored in a metadata keyword. The ADCFLGS column is thus translated into the mask and then deleted.

This step also checks if detector ADU values are truncated, i.e. if the value is at the upper or lower extreme of the ADC range (0 or 65535). In this case, the sampling is flagged as invalid by setting the appropriate bit of the mask value. The fraction of flagged sampling is recorded.

Inputs: Reformatted raw detector timelines and mask timelines from last processing step

Calibration files: None

Outputs: Reformatted raw detector timelines with the ADC mask bits now set in the mask table; format and units are the same as for the inputs.

3.3 Mask Bad Channels

Description: This module populates the sample mask table of the Level 0.5 format timeline data using the information from the Channel Mask Calibration Product and the Instrument Mode Mask. The mask table contains an integer mask for every detector for every time step. The full description of the mask handling procedure and the available masks can be found in RD6. If the mask is set in the calibration product then the relevant bit is raised in the mask table for all time samples.

Inputs: Outputs of Check ADC Flags and Truncation module.

Calibration files:

Channel Mask Table: ScalPhotChanMask, ScalSpecChanMask	Format	Boolean value for each channel for “isDead” and “isNoisy” status (1=TRUE)
	Units	Dimensionless
Instrument Mode Mask Table: ScalPhotInstMask	Format	Boolean value for each channel status (1=TRUE)
Channel Number Mapping Table: SCalPhotChanNum, ScalSpecChanNum	Comments	Provides mapping between channel numbers and channel names, both in full array mode and individual array mode; also gives the JFET output connector number, LIA board for each channel and a series of columns that can be used to distinguish between detectors viewing the sky, resistors,

		thermistors, dark detectors, non-connected channels and PTC channels
	Format	String value for each channel name. Integer for full channel number and individual channel number
	Units	Dimensionless

Outputs: Raw detector timelines with mask table populated from Channel and Observation Mask Table; format and units are the same as for the inputs.

3.4 Mask Bad TM Parameters

Description: The necessity of this module is still under discussion

Inputs:

Calibration files:

Non-Detector Telemetry Mask Table: SCalTelemMask	Format	Boolean value for each channel
	Units	Dimensionless

Outputs:

3.5 Time Conversion and Re-ordering

Description: The time in the Level 0 products is the time field of SPIRE data frames. This is in microseconds since 1st January 1958, stored as a 64-bit integer. It is computed via the following equation:

$$\text{dataFrameTime} = (\text{long})[(\text{treset})(1E6/65536) + (\text{frameTime})(3.2)] \quad (1)$$

However, if there is a building block longer than 229 minutes, the frameTime counter rolls over. It is assumed that the Treset is changed only in the initialization of building blocks, therefore for long scan lines the dataFrameTime will become invalid. Therefore this processing stage recalculates the sample time taking into account the frameTime counter roll over.

Re-ordering can now be implemented as each data sample in a timeline has an associated absolute time. Fine time in microseconds is converted to a double floating point value in seconds. Note that Nominal Housekeeping and Critical Housekeeping telemetry packets contain only a single frame and therefore do not have an associated frametime. Therefore the time that is associated with the sampling of the parameters is simply the telemetry packet time. Thus for these timelines a simple conversion of the packet time in CUC format to seconds since 1958 is performed, followed by a simple sorting of the table.

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. This correction exists as a method in the DetectorTimeline object but is **not** applied at this stage and each pixel shares a single timeline. The correction is considered insignificant for the photometer and never applied, while it is required for the spectrometer, but applied at a later stage of the spectrometer specific pipeline (Time Domain Phase Correction).

Inputs: Detector and telemetry timelines with dataFrameTime in the time field of SPIRE data frames.

Detector signal	Format	PDT (output from Mask Bad Channels module)
	Units	ADU

Sample time	Format	64 bit integer containing the time from January 1st 1958
	Units	μs

Calibration files:

DPU Counter Reset History: ScalresetHist (see 6.2)	Format	DPU counter reset time in CUC format time in 1/65536 seconds since 1st January 1958 as a long integer
	Units	1/65536 seconds

Outputs: Corrected and sorted detector timelines and sorted telemetry timelines with time converted to on-board time as a double floating point.

Detector signal	Format	PDT (output from Mask Bad Channels module)
	Units	ADU
Sample time	Format	Double floating point
	Units	seconds

3.6 Convert ADU to JFET Voltages

Description: The RMS voltage at the JFET output is computed according to the scheme describe in AD1 Section 3.11. The equation used is

$$V_{\mathrm{JFET-RMS}}(\omega_b, DATA, OFFSET) = \left[\frac{5}{G_{\mathrm{tot}}(\omega_b)}\right]\left[\frac{DATA - 2^{14} + (52428.8)(OFFSET)}{(2^{16}-1)}\right]. \quad (2)$$

where *DATA* and *OFFSET* are the values (in decimal) of the ADUs in the telemetry for the detector data and the DC offset applied to keep the detector signal within the dynamic range of the electronics. Offset values are usually automatically defined at the start of an observation and are used to create an Offset History Product which is provided as an input to this module in the pipeline. (There is no guarantee that the offsets are down-linked during an observation and an observation cannot therefore be processed until the Offset History Product becomes available). The total gain, G_{tot} is a function of the bias frequency ω_b and is stored as a calibration product which contains the value of G_{tot} at a reference frequency, stored in a metadata keyword (see RD8). The calibration file will return the value of G_{tot} at the required frequency using the equation:

$$G_{tot}(\omega_b) = G_{tot}(\omega_{b\text{-Ref}})\left[\left|\frac{f(\omega_b)}{f(\omega_{b\text{-Ref}})}\right|\right], \quad (3)$$

where

$$f(\omega_b) = \left[\frac{(4.7\times10^{-3})j\omega}{1 + (4.7\times10^{-3})j\omega + A(j\omega)^2}\right], \quad (4)$$

and A is a function of the detector channel resistance and transconductance as defined in RD2. A takes the values 5.85 x 10^{-7} for the photometer and 3.14 x 10^{-7} for the spectrometer. These are stored in the calibration product metadata.

Inputs: Time-ordered detector timelines from previous module.

Detector signal	Format	PDT (output from Time Conversion and Reordering Module)
	Units	ADU
Sample time	Format	Double Floating Point
	Units	s
Mask table	Format	32-bit integer
	Units	Dimensionless

Calibration files:

Channel Gain Table: SCalPhotChanGain, SCalSpecChanGain	Description	Total (LIA + amplifier) gain for every detector for a reference bias frequency
	Format	Double
	Units	Dimensionless
	Description	JFET gain for every detector
	Format	Rational number; four significant figures
	Units	Dimensionless
Signal Offset History File (see Section 6.2) SCalPhotOffsetHist, SCalSpecOffsetHist	Description	Timeline for each detector containing the offsets
	Format	32 bit integer (containing unsigned 16 bit integer)
	Units	ADU

Outputs: Same as inputs except detector signal now converted from ADU to JFET voltage.

Detector signal	Format	Scientific notation to 10 significant figures
	Units	V
Sample time	Format	Double Floating Point
	Units	s
Mask table	Format	32-bit integer
	Units	dimensionless

3.7 Convert non-detector data ADU to Engineering Values

Description: This module will carry out the conversion of the data from ADU to meaningful units in the raw non-detector timelines (Raw Nominal Housekeeping Timeline, Raw BSM Timeline, Raw SCU Timeline, Raw SMEC Timeline, Raw Critical House Keeping Timeline, Raw Offset Timeline).
The conversion of each non-detector timeline parameter to the appropriate engineering unit is implemented by applying an ASCII conversion table originally developed for the Quick Look Analysis (QLA) environment. These tables contain either a conversion formula or conversion table to be applied and are not delivered as calibration files. Also included in the tables are the output units of the corresponding conversion (volts, temperature, angle, etc). A full list of these tables are documented in RD9.

Inputs: Non-detector timelines in Level 0.5 Product format

Calibration files: A set of ASCII tables presently contained in spire.param.tables

Conversion Tables	Description	Conversion from Parameter x in ADU to Parameter x in appropriate units (volts, temperature, angle, etc) or Appropriate conversion table of input and output values in appropriate units.
	Format	ASCII, or rational number depending on the conversion
	Units	Dimensionless

Outputs: Non-detector timelines in Level 0.5 Product format with parameters converted from raw ADU into meaningful units.

Non-detector timelines	Format	Unchanged from input, double or string depending on table conversion type
	Units	As specified in conversion tables

3.8 Calculate RMS Bolometer Voltage and Resistance

Description: The RMS voltage at the JFET output computed as described in Section 3.6 is used to derive the RMS voltage at the bolometer, and the bolometer resistance, by an iterative procedure designed to take into account the RC roll-off due to the harness transfer function and also any changes in the phasing of the bias demodulator.

The procedure for calculation of bolometer resistance and voltage is as described below (following AD1 Section 3.9).

Step 1: Estimate detector $V_{\text{d-RMS}}$ and R_{d} from the bias voltage and current V_{b} and I_{b}. Assuming a total load resistance R_{L} and taking the transfer function of the harness between the detector and JFET input $H_{\text{H}}(\omega_{\text{b}}) = 1$:

$$V_{\text{d-RMS}} = \frac{V_{\text{JFET-RMS}}}{H_{\text{JFET}}}, \quad I_{\text{b-RMS}} = \frac{V_{\text{b-RMS}} - V_{\text{d-RMS}}}{R_{\text{L}}}, \quad \text{and} \quad R_{\text{d}} = \frac{V_{\text{b-RMS}}}{I_{\text{b-RMS}}} - R_{\text{L}}. \qquad (5)$$

Step 2: Estimate $H_{\text{H}}(\omega_{\text{b}})$ and phase difference $\Delta\phi$, using the time constant and capacitance of the harness τ_{H} and C_{H}:

$$\left|H_{\text{H}}(\omega_{\text{b}})\right| = \frac{1}{\left[1 + (\omega_{\text{b}}\tau_{\text{H}})^2\right]^{1/2}} \quad \text{with} \quad \tau_{\text{H}} = \left[\frac{R_{\text{L}} R_{\text{d}}}{R_{\text{L}} + R_{\text{d}}}\right] C_{\text{H}}, \qquad (6)$$

and $\Delta\phi$ given by equation (12) in AD1 Section 3.5.2:

$$\Delta\phi = \tan^{-1}(\omega_{\text{b}}\tau_{\text{H-nom}}) - \tan^{-1}(\omega_{\text{b}}\tau_{\text{H}}) \quad \text{with} \quad \tau_{\text{H-nom}} = \left[\frac{R_{\text{L}} R_{\text{d-nom}}}{R_{\text{L}} + R_{\text{d-nom}}}\right] C_{\text{H}}.$$

In order to compute this correction, the value of $R_{\text{d-nom}}$ (the resistance when the telescope views dark sky) must be known.

Step 3: Recalculate $V_{\text{d-RMS}}$ and R_{d}:

$$V_{\text{d-rms}} = \frac{V_{\text{JFET-RMS}}}{H_{\text{JFET}}\left|H_{\text{H}}(\omega_{\text{b}})\right|\cos(\Delta\phi)}, \quad I_{\text{b-rms}} = \frac{V_{\text{b-rms}} - V_{\text{d-rms}}}{R_{\text{L}}}, \text{ and } \quad R_{\text{d}} = \frac{V_{\text{b-RMS}}}{I_{\text{b-RMS}}} - R_{\text{L}}. \qquad (7)$$

Continue iterating (repeat steps 2 and 3) until $I_{\text{b-RMS}}$ and R_{d} converge (criterion: change on iteration < 0.1%).

The RMS detector voltage is then just $$V_{\text{d-RMS}} = I_{\text{b-RMS}} R_{\text{d}}. \qquad (8)$$

Inputs: Detector Timelines from previous Convert ADU to JFET Voltages module.
Bias Voltage values from Nominal House Keeping Timeline

Bias Voltage	Description	Bias Voltage is stored in the Nominal House Keeping Timeline as the commanded bias voltage amplitude and will be converted to rms in the

		module
	Format	Scientific notation to 10 significant figures
	Units	V

Calibration files:

Bolometer Parameter Table: SCalPhotBolPar, ScalSpecBolPar	Description	Load resistor (R_L) values – mission-fixed constants, one for each detector, from the JPL BDA EIDPs
	Format	Scientific notation to 3 significant figures
	Units	Ω
	Description	Harness capacitance (C_H) values One value for each detector; currently not likely to have different values for different detectors in an array; file to be populated with nominal values pre-flight and not likely to be updated in flight.
	Format	Scientific notation to four significant figures
	Units	F
Channel Gain Table: SCalPhotChanGain, SCalSpecChanGain	Description	JFET gains (H_{JFET} values); one value for each detector
	Format	Scientific notation to four significant figures
	Units	Dimensionless
Channel nominal resistances (SCalChanNomRes)	Description	Nominal blank sky detector resistance values ($R_{d\text{-}nom}$); derived from calibration observations in flight; pre-launch file to be populated with estimated numbers from bolometer models and estimated photon backgrounds.
	Format	Scientific notation to four significant figures
	Units	Ω
Channel Number Mapping Table: SCalPhotChanNum, SCalSpecChanNum	Description	Mapping between channel numbers and channel names, both in full array mode and individual array mode; also gives the JFET output connector number, LIA board for each channel and a series of columns that can be used to distinguish between detectors viewing the sky, resistors, thermistors, dark detectors, non-connected channels and PTC channels.
	Format	String value for each channel name; integer for full channel number and individual channel number
	Units	Dimensionless

Outputs: Detector timelines (now with voltage referred to the bolometer) and with detector resistance table appended as additional column.

Detector voltage	Description	Voltage across bolometer
	Format	Scientific notation to nine significant figures (determined by the number of bits, 20)
	Units	V
Detector resistance	Description	Resistance, R_d, of each detector
	Format	Scientific notation to nine significant figures
	Units	Ω
Phase difference	Description	Demodulator phase difference, $\Delta\phi$, for each detector
	Format	Scientific notation to three significant figures
	Units	radians
Sample time	Format	Double Precision
	Units	s

Mask table	Format	32-bit integer
	Units	dimensionless

3.9 Add Pointing Meta Data Parameters

Description: This module associates the nodding ID for photometer Point Source and Small Map observations, the scan line number for Large Map and parallel observations with the appropriate building blocks. 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 scan number ID can be added to the meta-data. Note that this is the addition of identifiers rather than actual positions. In the case of non-nodding or non-scanning the expectation is that the identifier fields in the meta-data will reflect this. The pointing information is stored the nominal housekeeping parameter STEP. The description of which bits in STEP are used to store the above information is specified in RD10.

Inputs: PDT from previous module. NHK timeline.

NHK STEP parameter nod-ID / scan line	Format	bits
	Units	none

Calibration files: None

Outputs: PDT with metadata added for Nod-ID or Scan Line Number.

This completes the description of the modules that are common to both the Photometer and Spectrometer parts of the pipeline.

4. PHOTOMETER PIPELINES

This section describes how the empirical photometer pipelines for the Scan Map and the Jiggle Map, as described in AD1, are to be implemented

4.1 Scan Map Processing (POF5: Large Map)

The data flow for the photometer Scan-Map mode, starting from the Level 0.5 products, is shown in Figure 2 and described in detail below.

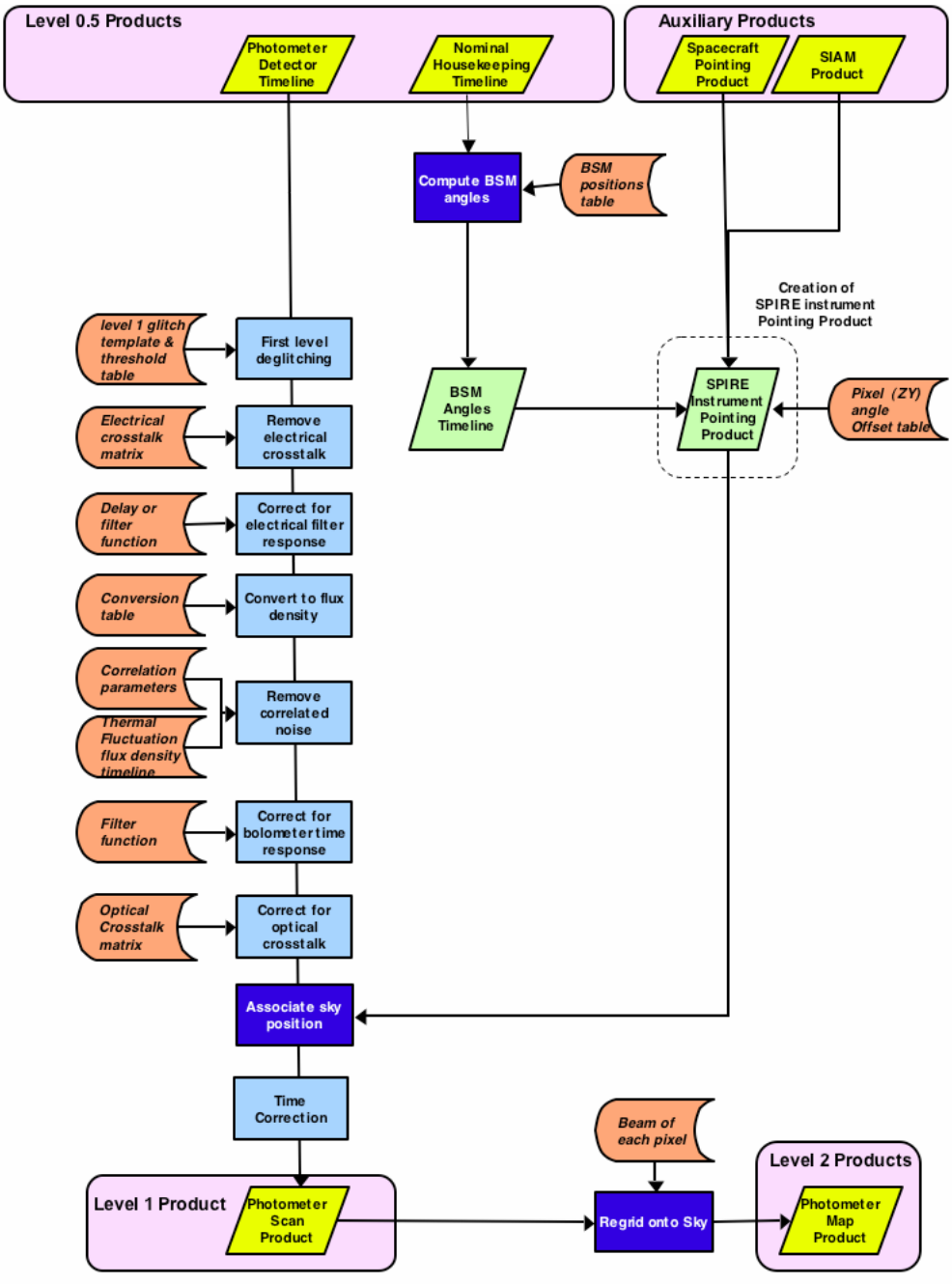

Figure 2: Flowchart for Scan-Map data flow to Level 2 products

4.1.1 Compute BSM Angles

Description: For the scan map, there is no need to create a chop-jiggle timeline as in the case of jiggle observations; however it is still necessary to create a BSM angle timeline because it will be necessary to know where the BSM is pointing, even if it should be at a fixed position. For scan map observations, the instrument does not produce BSM telemetry packets so instead the input comes from the Nominal Housekeeping Timeline (NHK) which contains the BSM sensor values. This process extracts focal plane Y,Z angles corresponding to the sample time in the NHK timeline by comparing the sensor signals in the NHK timeline and BSM Positions Table which contains the Y,Z angles for a given chop and jiggle BSM sensor value (see Figure 3). Note that the time in the NHK timeline is sampled at a lower rate of 1Hz and therefore the corresponding BSM Angles Timeline will also be sampled at the same rate.

Inputs: Nominal Housekeeping Timeline

Sample Time	Description	On board time Sampled at 1Hz
	Format	Double Floating Point
	Units	s
Chop sensor signal	Format	Integer
	Units	ADU
Jiggle sensor signal	Format	Integer
	Units	ADU

Calibration files: The BSM Position Table provides calibration between the sensor signal and angle on sky in focal plane

Column 1: Chop sensor signal	Format	Integer
	Units	ADU
Column 2: Jiggle sensor signal	Format	Integer
	Units	ADU
Column 3: Focal Plane Y angles	Format	Rational numbers specified to 8 significant figures
	Units	arcseconds
Column 4: Y angle errors	Format	Rational numbers specified to 8 significant figures
	Units	arcseconds
Column 5: Focal Plane Z angles	Format	Rational numbers specified to 8 significant figures
	Units	arcseconds
Column 6: Z angle errors	Format	Rational numbers specified to 8 significant figures
	Units	arcseconds

Outputs: BSM Angles Timeline

Column 1: Sample Time	Description	On board time Sampled at 1Hz
	Format	Double Floating Point
	Units	s
Column 2: Focal Plane Y angles	Format	Rational numbers specified to 8 significant figures
	Units	arcseconds
Column 3: Y angle errors	Format	Rational numbers specified to 8 significant figures
	Units	arcseconds
Column 4: Focal Plane Z angles	Format	Rational numbers specified to 8 significant figures
	Units	arcseconds
Column 5: Z angle errors	Format	Rational numbers specified to 8 significant figures
	Units	arcseconds

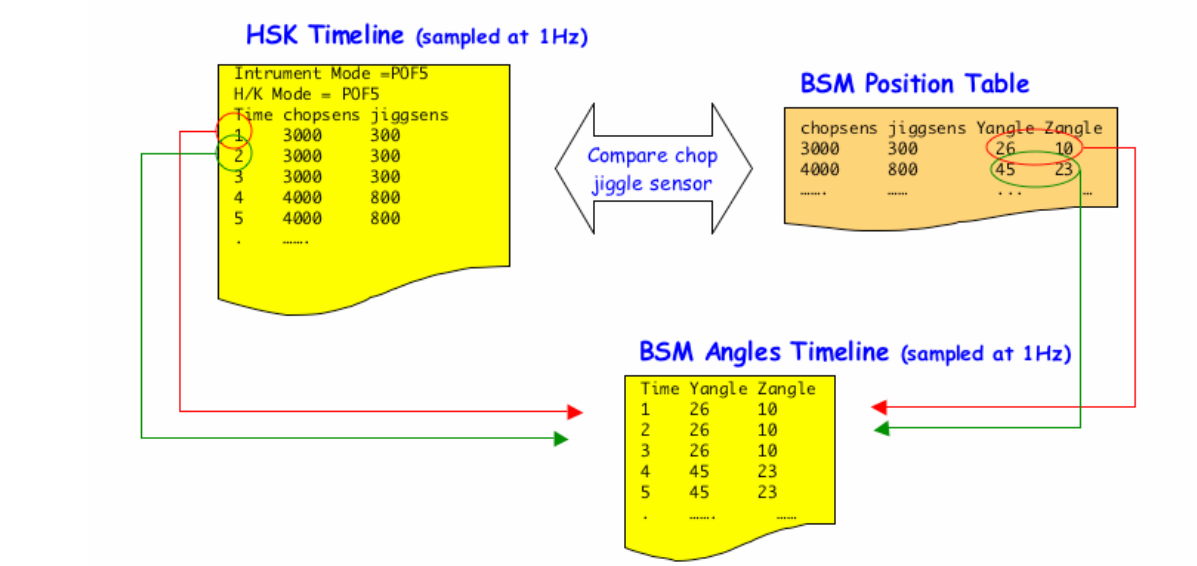

Figure 3: Creation of BSM Angles Timeline from Compute BSM Angles Module

4.1.2 Creation of the SPIRE Instrument Pointing Product

Description: The creation of the SPIRE Instrument Pointing Timeline Product is not a processing step in the strictest sense but assembles the data in the Spacecraft Pointing Product, SIAM Product, the BSM angles timeline and the detector angular offset table which contains the angular offsets of each detector from the telescope boresight (aligned with the central detector) in arcseconds on sky in spacecraft Y-Z coordinates. The calculation of the pointing algorithm is detailed in Section 4.1.10.

Inputs:

Spacecraft Pointing Product	Description	x,y,z,w Quaternions
	Format	Double Precision
	Units	none
	Description	On Board Time
	Format	Double Precision
	Units	Micro seconds
	Description	Aperture in use
	Format	String Snn_s (00<nn<76)
	Units	None
SIAM Product	Description	Instrument Aperture
	Format	String
	Units	none
BSM Angles Timeline	Description	Time
	Format	Double Integer
	Units	s
	Description	Y, Z angles
	Format	8 significant figures (determined by optical model)
	Units	arcsecs

Calibration files:

Detector Angular Offset Table SCalPhotDetAngOff	Description	*N*x4 matrix for each array, where *N* is the number of detectors in the array, giving the Y and Z offsets of each

		detector position with respect to the SPIRE primary aperture. Column 1: Y offsets Column 2: Y offset uncertainties Column 3: Z offsets Column 4: Z offset uncertainties
	Format	Rational numbers specified to 8 significant figures
	Units	Arcseconds

Outputs: SPIRE Instrument Pointing Product containing the Spacecraft Pointing Product, SIAM Product, BSM Angles Timeline and the Detector Angular Offset Table with the associated methods required to obtain the sky position of a detector at a given time.

4.1.3 First Level Deglitching

Description: Before further processing of the measured detector voltage timelines, glitches due to cosmic ray hits or other impulse-like events in the detectors will be removed. The approach is described in AD2 based on a local regularity analysis combined with a wavelet analysis. This module is applied to the thermistors as well as the bolometers. Note that the Bias Voltage flag (i.e. high or low setting, contained in the PDT metadata) is also required.
This module documentation has still to be inserted.

Inputs: Level 0.5 Photometer Detector Timeline (PDT) Product
Bias Voltage Flag from PDT meta data (glitch table values may depend on bias voltage)

Calibration files:

Glitch Table (ScalPhotGlitch)	Description	Separate files for each bias setting
	Format	
	Units	

Outputs: Deglitched PDT with glitch flags raised in the mask table

4.1.4 Remove Electrical Crosstalk

Description: The procedure for removing electrical crosstalk is to multiply the vector of bolometer voltages by an electrical crosstalk matrix, $\mathbf{C}_{\mathbf{elec}}$, as described in AD1. The vector of electrical crosstalk-corrected signals is given by

$$\mathbf{V}_{\mathbf{corrected}} = \mathbf{C}_{\mathbf{elec}}\mathbf{V}\,. \tag{9}$$

Inputs: PDT from output of First Level Deglitching module

Calibration files:

Electrical crosstalk matrix: SCalPhotElecCross	Description	*N*x*N* matrix for each array, where *N* is the number of detectors in the array; diagonal elements are unity; in the absence of crosstalk, non-diagonal elements are zero.
	Format	Rational numbers to three significant figures
	Units	Dimensionless

Outputs: Same as input; detector voltages now corrected for electrical crosstalk.

4.1.5 Correct for Electrical Filter

Description: The electronics chain imposes a delay on the data with respect to the telescope position along the scan; this effect must be taken into account to ensure that the astrometric pointing timeline is properly matched to the detector data timeline. This module is applied to the thermistors as well as the bolometers.

- Option 1 (Digital filtering)
  - Timeline is transformed to frequency domain, multiplied by correction function and transformed back
  - Calibration file = correction function $CF1_i(\omega)$; (nominally same for all detectors but provision for having it specific to each detector)
- Option 2 (Simple delay)
  - Algorithm involves imposing a simple relative shift of the detector data and pointing timelines.
  - Shift must be calculated in terms of a number of samples
  - Nominal value = 74 ms
  - Sampling intervals for the pointing and detector timelines are 4Hz and 16Hz respectively

At present the simple Delay method is favoured over the Digital Filtering Method
There are two options for implementing the Simply Delay time shift:
(i) keep the detector samples and change the timestamps by subtracting the fixed delay from each.
(ii) keep the timestamps and interpolate to derive corrected detector samples.

Since the nominal sampling rate is 16 Hz – i.e., sampling interval = 62.5 ms, the delay is not an integer number of samples. The worst-case (PSW) number of samples per FWHM-crossing time is 9.6 for 30"/s and 4.8 for 60"/s. So in all cases, the sampling of the response to a point source is better than Nyqvist. It is therefore acceptable to implement the delay by interpolation between samples if desired.

Inputs: PDT from output of Remove Electrical Crosstalk module

Calibration files:
Option 1 (Digital filtering):

Electrical Filter Correction Function (SCalPhotElecFiltCorr)	Description	Parameter values for the electrical filter transfer function
	Format	TBW
	Units	TBW

Option 2 (Simple delay):

Electrical Filter Delay Table SCalPhotElecFiltCorr	Description	Time delay of electrical filter with respect to input. Nominally the same value for all photometer detectors. One column table for each array containing the value of the delay in ms (74 ms) for each detector.
	Format	Rational number to four significant figures
	Units	s

Outputs: PDT with corrected detector timeline - corresponds to an earlier version of the timeline with respect to the pointing timeline – i.e., each detector sample is associated with a pointing ~74 ms earlier.

4.1.6 Convert to Flux Density

Description: The procedure for deriving the in-beam flux density, including flat fielding and strong source correction, is described in AD1 (Section 5.4). The flux density for detector i is calculated from the measured signal voltage, $V_{S\text{-}i}$ and a fixed voltage, $V_{o\text{-}i}$, using astronomical gain factors $K_{1\text{-}i}$, $K_{2\text{-}i}$, and $K_{3\text{-}i}$, which are contained in calibration files. They will depend on the detector bias conditions. This module is applied only to the photometric pixels (i.e., not to the dark pixels or the thermistors).

$$S_1 = K_1(V_S - V_o) + K_2 \, ln\left(\frac{V_S - K_3}{V_o - K_3}\right) \qquad (10)$$

The output of this module is a set of flux density timelines. However it should be cautioned that although the output units are in Jy, the output does not yet reflect the flux density from the sky.

Inputs: PDT (output of Correct for Electrical Filter module)
Bias Voltage Flag (dimensionless integer) from PDT meta data indicating nominal or high bias

Calibration files:

Astronomical Unit Conversion Table (SCalPhotUnitToAst) Separate calibration filee for each bias setting.	Description	*K*-parameers K_1 , K_2, K_3 for each detector. Tables of *N* values for each array where *N* is the number of detectors in the array.
	Format	Scientific notation, specified to three significant figures
	Units	K_1 is in Jy V^{-1}, K_2 is in Jy , K_3 is in V
Fixed voltage offset		One V_o value for each detector; determined by blank sky observations (equivalent to the value measured for $R_{d\text{-}nom}$ above – can be measured from the same data set);
	Format	Scientific notation, specified to three significant figures.
	Units	V

Outputs: PDT, as for inputs except with detector sample values converted to flux density

Detector voltage	Description	Detector Flux Density
	Format	Scientific notation to accuracy of input PDT
	Units	Jy
Sample time	Format	Double Precision
	Units	s
Mask table	Format	32-bit integer
	Units	dimensionless

4.1.7 Remove Correlated Noise due to Bolometer Temperature Fluctuations

Description: This module corrects scan map data timelines for 1/*f* noise noise caused by variations of the detector array bath temperature. A correction timeline is generated for each detector using thermometry data and calibration information for that detector. This is then subtracted from that detector's signal timeline. Note that the thermistor timelines are also expected to have been through the same processing steps as the PDT (except for the conversion from Volts to flux density) including the correction for the delay due to electrical filter response in order to ensure the thermistor signal and detector signal are maintained in the same phase. The correction timeline estimated by the following formulas:

Correction timelines for thermometers T_1 and T_2:

$$S_{T1}(t) \;=\; A1\left(\overline{V}_{T1} - V_{o1}\right) + 0.5B_1\left(\overline{V}_{T1} - V_{o1}\right)^2 \qquad (11)$$

V_{T1} is the smoothed (default time span = 10 s) voltage signal of T1. V_{01} is a reference signal of T1, set during the calibration. Similar definitions for T2 below.

$$S_{T2}(t) = A_2\left(\overline{V}_{T2} - V_{o2}\right) + 0.5B_2\left(\overline{V}_{T2} - V_{o2}\right)^2 \quad (12)$$

Average correction timeline:

$$S_T(t) = \frac{1}{2}\left[S_{T1}(t) + S_{T2}(t)\right] \quad (13)$$

(Note there are actually 3 options in the module: $S_T=S_{T1}$, $S_T=S_{T2}$, and $S_T=0.5*[S_{T1}+S_{T2}]$)

The corrected detector timeline is given by:

$$S_{Corr}(t) = S(t) - S_T(t) \quad (14)$$

At the high bias voltage, the thermistors are saturated, and the temperature drift will be traced by dark pixels. Therefore, for the high bias voltage, the dark pixel voltages V_{DK} are used instead of the thermistor voltages V_T.

Inputs:
PDT as output from Convert to Flux Density module
De-glitched thermistor time lines after the deglitching.
Bias Voltage Flag (dimensionless integer) from PDT meta data indicating nominal or high bias

Calibration files:

Temperature drift correction (SCalPhotTempDriftCorr)	Description	A Table data set containing parameters for T_1 and T_2, for each of the two bias settings. One for each detector array
Column 1: Channel Name	Format	String
	units	None
Column 2: V_{o1} Column 3: V_{o1} error	Format	Float (5 significant figures accurate to 1%) Float
	Units	Volts
Column 4: V_{o1} flag	Description	indicates whether T_1, T_2 or average of T_1 and T_2 is used
	Format	Boolean
	Units	None
Column 5: A_1 Column 6: A_1 error Column 7: B_1 Column 8: B_1 error	Format	Float (4 significant figures accurate to 5%) Float Float (4 significant figures accurate to 5%) Float
	Units	Jy V^{-1}
Column 9: AB_1 flag	Description	indicates whether T_1, T_2 or average of T_1 and T_2 is used
	Format	Boolean
	Units	None
Column 10: V_{o2} Column 11: V_{o2} error	Format	Float (5 significant figures accurate to 1%) Float
	Units	Volts
Column 12: V_{o2} flag	Description	indicates whether T_1, T_2 or average of T_1 and T_2 is used
	Format	Boolean
	Units	None
Column 13: A_2 Column 14: A_2 error Column 15: B_2 Column 16: B_2 error	Format	Float (4 significant figures accurate to 5%) Float Float (4 significant figures accurate to 5%) Float
	Units	Jy V^{-1}
Column 17: AB_2 flag	Description	indicates whether T_1, T_2 or average of T_1 and T_2 is used
	Format	Boolean
	Units	None

Column 21: Thermistor averaging period	Description	Period of time for averaging of thermistor voltages
	Format	Rational number to three significant figures
	Units	s

Outputs: PDT with bolometer voltages corrected detector timelines (format: same as input)

4.1.8 Correction for Bolometer Time Response

Description: The bolometer transfer function is represented as a two-component system as described in AD1. The baseline plan to correct for the slow detector time constant is to use the following procedure:

(1) Fourier transforming the signal timeline for detector i, $S_i(t)$;
(2) multiplying the FT by an appropriate correction function $CF2_i(\omega)$;
(3) transforming back to the time domain to obtain the corrected estimate of the signal, $S_{S3\text{-}i}(t)$

$CF2_i(\omega)$ will be derived from calibration file parameters stored for each detector:

(i) nominal detector time constant, $\tau_{1\text{-}i}$
(ii) slow detector time constant, $\tau_{2\text{-}i}$
(iii) time constant amplitude factors, a_i

At present the preferred method is to shift and re-grid the detector data samples, retaining the time stamps. This is more viable than to create a table data set with time for each separate detector which may complicate the association of sky position.

Inputs: PDT as output from Correct for Correlated Noise module

Calibration files:

Detector Time Constant correction function (SCalPhotDetTimeConst)	Description	3 x N table for each array giving parameters for the correction funcion Column 1 : $\tau_{1\text{-}i}$ Column 2 : $\tau_{2\text{-}}$ Column 3 : a_i
	Format	$\tau_{1\text{-}i}$: rational number to four significant figures $\tau_{2\text{-}i}$: rational number to four significant figures a_i : rational number to four significant figures
	Units	$\tau_{1\text{-}i}$: ms $\tau_{2\text{-}i}$: ms a_i : dimensionless

Outputs: PDT, with flux densities corrected for bolometer time constant (format: scientific notation to the same number of significant figures as the input detector voltage).

4.1.9 Optical Crosstalk Removal

Description: Optical crosstalk is removed by multiplying the vector of bolometer signals for an array, $\mathbf{V_S}$, by an optical crosstalk matrix, $\mathbf{C_{opt}}$:

$$\mathbf{V_{S2}} = \mathbf{C_{opt}}\,\mathbf{V_{S1}}\,. \qquad (15)$$

The elements of $\mathbf{C_{opt}}$ are determined from calibration observations involving scanning a strong point source across each of the detectors in the array. The output of this module is a set of flux density timelines corrected for optical crosstalk drifts.

Inputs: PDT (output from Correct for Bolometer Time Response module)

Calibration files:

Optical Crosstalk Matrix (SCalPhotOptCross)	Description	*N* x *N* matrix for each array, where *N* is the number of detectors in the array. In the absence of crosstalk, diagonal elements are unity and non-diagonal elements are zero.
	Format	Scientific notation, specified to three significant figures;
	Units	Dimensionless

Outputs: Same as input; flux densities corrected for optical crosstalk.

4.1.10 Associate Sky Position

Description: This module attaches the sky position timeline onto the detector timeline. It does this by querying the SPIRE Pointing Product which contains the Spacecraft Pointing Product, SIAM Product, the BSM Angles Timeline and the Detector Angular Offset table. The SPIRE Pointing Product is queried for the position of the detectors at each sample time. The SPIRE Pointing Product computes this position by the following method (summarised in Figure 4).

1. Associate Sky Position Module sends Pixel and time to Spacecraft Pointing Product (SPP). Processing is carried out by looping over the timeline.
2. The SPP method returns the pixel offset (i.e., *Yoffset*, *Zoffset*) from the Detector Angles Table (DAT) and the BSM offset from the BSM Angles Timeline for that pixel and time
3. The SPP then returns the Herschel Pointing Product (HPP) for that time in Quaternians, i.e. the Attitude Common Frame (ACF).
4. The SPP then uses the available SIAM aperture (S14 in the case of the Photometer) to get the SIAM matrix in Cosine vector directions (the direction cosine matrix).
5. The SPP method applies the SIAM matrix to the HPP to produce "*Pspire*" the position of the SPIRE aperture on the sky in Quaternians.
6. The SPP method gets the absolute RA and Dec of *Pspire* by adding the combined Detector and BSM offset (*Yoffset*, *Zoffset*).
7. The result is the absolute RA, Dec of the pixel on the sky as a function of time.

Comments:
The position of the SPIRE primary aperture in RA, Dec should be included as an extra column in the resulting product?
For moving objects give the offset for pixels instead of RA, Dec

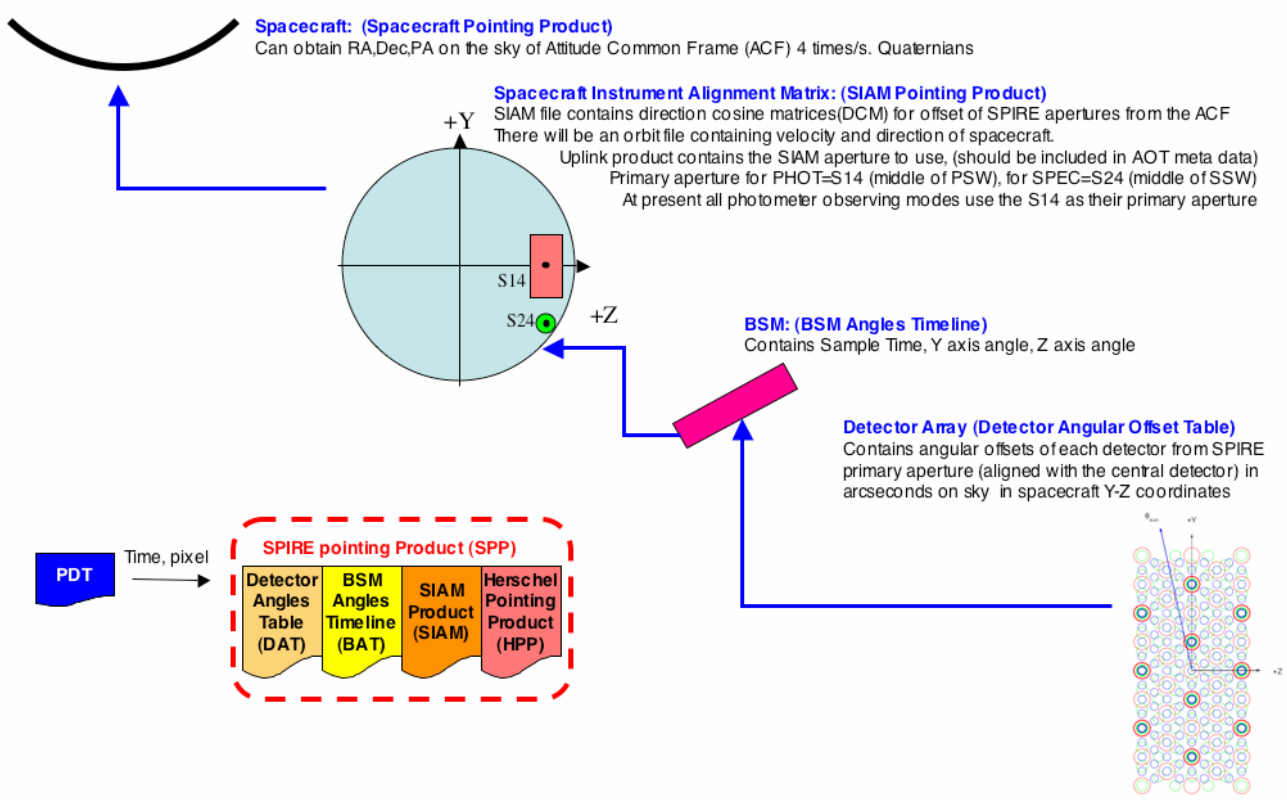

Figure 4: Steps to obtaining the absolute position on the sky from the Associate Sky module, for the timeline data for a given pixel.

Note that the pixel offsets from the primary aperture are based on the mechanical layout of the arrays and apply to a stationary telescope. The fact that not all pixels are read out at the same time means that small time differences between readouts should in principle be folded in to the pointing correction. We assume for now that this correction is negligible.

Once the position has been calculated, it is still under consideration as to whether it is necessary to calculate the telescope scan velocity. The easiest way to do this would be to calculate angular distance moved between each step. However, it is uncertain whether this is sufficient for the next step assuming constant time samples or whether the delta time needs to be taken into account

Inputs: SPIRE Instrument Pointing Product

Calibration files: None

Outputs: Detector timelines with position timelines (see Photometer Scan Product in Section 4.1.12)

4.1.11 Time Correction

Description: The On Board Time (OBT) in the detector timelines is not a "time standard" in the strictest sense as are International Atomic Time (TAI) and Coordinated Universal Time (UTC). Even if OBT is set to TAI at launch, there will be a drifting of the clock that is on board of the spacecraft. Therefore, the time following the OBT reference, are in fact uncorrected TAI values and this module is used to correct these OBT values to TAI values. Time packets are sent with TM frames to ground and contain only the OBT. Each TM frame receives a time stamp by the ground station referred to the Earth Reception Time (ERT). If the distance of the spacecraft to the ground stations is known, the ERT can be corrected by a Time Correlator (TCO)and used together with the Time Packets to calculate the time correlation between OBT and UTC.

Note the pipeline does not actually convert to UTC, and leaves all times as TAI. However, when the products are exported to FITS format, certain standard metadata values such as the start and end times are converted to UTC in the FITS headers. Times in columns remain as TAI (note that conversion from TAI to UTC only involves a simple offset of exactly 33 seconds.

Input: Timelines with time columns in OBT

Calibration Files:

Output: Timelines with time columns with corrected OBT to TAI

4.1.12 Photometer Scan Product (Level 1)

The final photometer scan pipeline Level 1 product consists of a set of timelines of detector pointings in sky coordinates with a flux density per beam associated with each pointing.
No calculation of statistical uncertainty shall be included up to this point – such errors will be evaluated in the map-making and source extraction stages.

Output:

Flux Density	Description	Flux Density Timeline
	Format	Scientific notation to 9 significant figures
	Units	Jy
Sample time	Format	Double Precision
	Units	s
Sky Coordinates	Description	RA and Dec with the equinox stored in the meta data
	Format	Double Precision
	Units	degrees
Mask table	Format	32-bit integer
	Units	dimensionless

4.1.13 Regrid onto Sky (Mapmaking)

The map making module regrids the timeline data onto the sky to produce an image map. At present two algorithms are available.

1) Naïve mapmaking
This mapmaking technique performs no additional data-processing on the timelines. The data are simply re-mapped onto the image plane, by projecting the full power seen by a detector onto the nearest sky map pixel. For each bolometer timeline at each time step, the signal measurement is added to the total signal map, the square of the signal is added to the total signal squared map, and 1 is added into the coverage map. After all bolometer signals have been mapped, the total signal map is divided by the coverage map to produce a flux density map, and the standard deviations are calculated using the total signal, total signal squared, and coverage map.

2) MADmap mapmaking
MADmap is a maximum-likelihood based method of estimating a final sky map from the input data. It uses a 'brute-force' approach to solve the system of linear equations

$$d_t = A_{tp} s_p + n_t. \tag{16}$$

where d_t is the time ordered data set (TOD) (t=1..m), s_p is the pixelized map (p=1..n). A_{tp} is the m-by-n pointing matrix

which projects the pixel domain onto the time domain. The noise vector n_t is a m-sample of random variables drawn from a multivariate gaussian distribution of mean zero, not independent (because of the 1/f noise) but of finite correlation length and piecewise stationary.
The maximum likelihood estimate of the map is

$$\hat{s} = (A^T N^{-1} A) A^T N^{-1} d . \quad (17)$$

where $N = < nn^T >$ is the time-time noise covariance matrix, A is the pointing matrix, d is the time ordered data, and s is the map estimate. Following the assumptions about the noise, N^{-1} is piecewise Toeplitz and circulant. It guarantees that the multiplication by the m-by-m matrix N^{-1} is a convolution, which is a low cost operation in Fourier space. This matrix is defined by its first row, which is obtained from the inverse discrete Fourier transform of the inverse of the noise power spectrum (PhotChanNoise calibration file). The projection from the time domain onto the map domain (multiplication by A^T) is performed by assigning the full power seen by a detector onto the nearest sky map pixel, or in other words, each row of the pointing matrix A has an entry exactly equal to one, the others being equal to zero. The inversion of the m-by-m matrix $A^T N^{-1} A$ (inverse of the pixel-pixel noise covariance matrix) is performed using the preconditioned conjugate gradient method. The pre-conditioner is a pixel domain diagonal matrix weighting the pixels by the number of time they have been observed.

Inputs: A set of Level 1 Photometer Scan Product timelines of detector pointings in sky coordinates with a flux density per beam associated with each pointing with associated mask table.

Calibration files:
Detector Noise Table (*ScalPhotChanNoise*) containing the noise power spectrum for each detector channel, to be used in the map making stage of the pipeline. There is one table dataset for each array. There will be several editions of this product for different detector bias frequency and amplitude settings (these parameters are stored as double precision values in the calibration product meta data).

Detector Noise Table (*ScalPhotChanNoise*)	Description	A Nx2 column contains the noise power spectrum for each detector channel, to be used in the map making stage of the pipeline. There is one table dataset for each array. There will be several editions of this product for different detector bias frequency and amplitude settings (these parameters are stored as double precision values in the calibration product meta data).
Frequency	Format	Double Precision
	Units	Hz
Noise Spectrum	Format	Double Precision
	Units	W/sqrt(Hz)

Outputs:
Spire Photometer Scan Product for either the naïve mapmaking or MADmap mapmaking. One Image map for each array (i.e. 3 maps). WCS information is stored in the meta data

Array Name	Format	String
	Units	None
Longitude	Format	Double Precision
	Units	degrees
Longitude error	Format	Double Precision
	Units	degrees
Latitude	Format	Double Precision
	Units	degrees
Latitude error	Format	Double Precision
	Units	degrees
Signal	Format	Double Precision
	Units	Jy
Signal Error	Format	Double Precision
	Units	Jy

4.2 Photometer Jiggle Observations (POF2: Point Source, POF3 Small Map)

Figure 5 is a flowchart showing the processing steps from the Level 0.5 to Level-2 products. The various steps are described in the following sections.

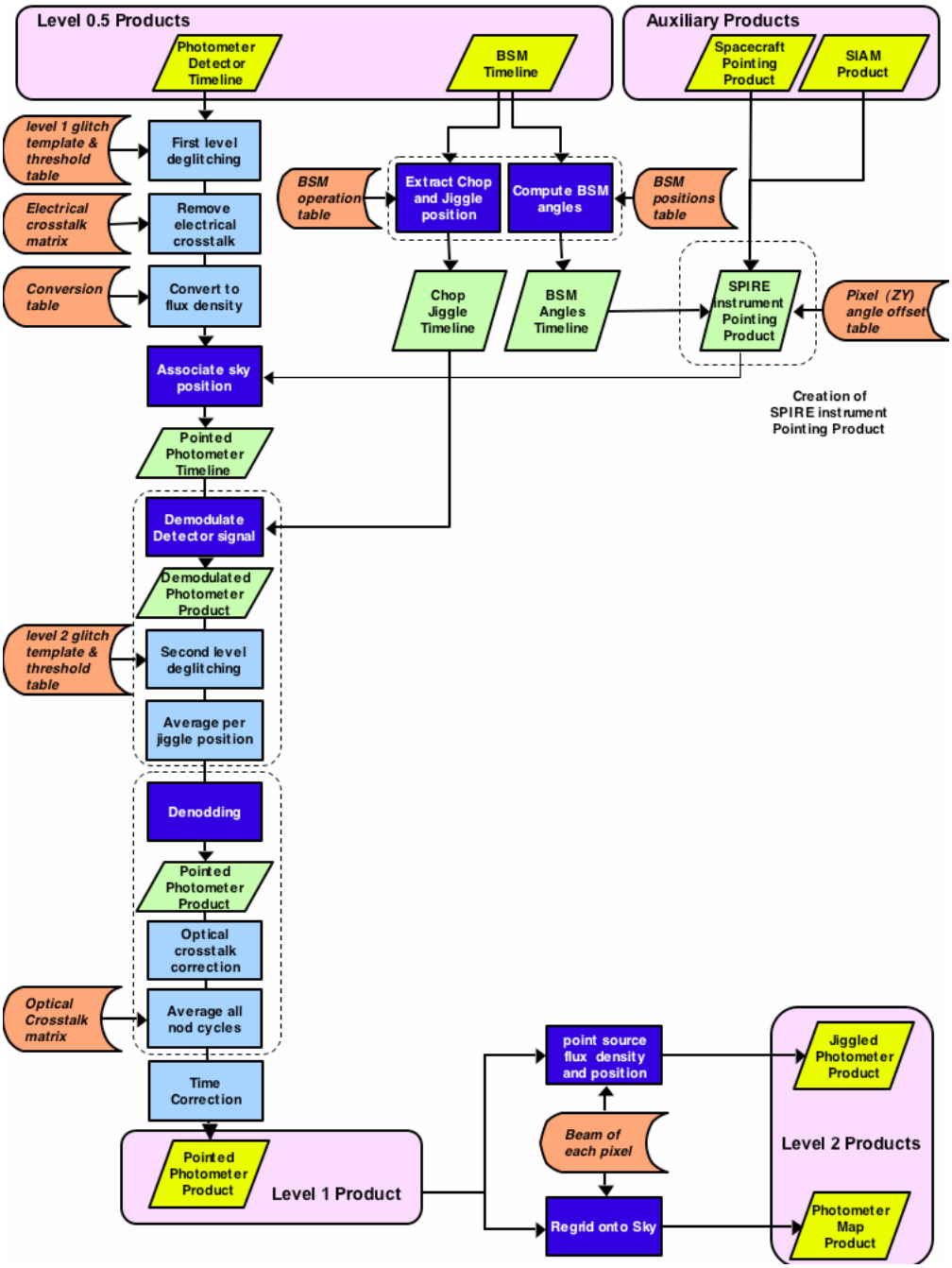

Figure 5: Flowchart for jiggle-map data flow to the creation of the Level 2 data products.

4.2.1 Extract Chop and Jiggle Positions

Description: A single building block contains a set of jiggle pointings at 8 positions (a single building Block for a point source jiggle and 8x8 building blocks for a 64-point map). In order to do first level deglitching, the chop and jiggle markers are needed. This module produces a Chop Jiggle Timeline Product containing the jiggle position markers and chop position markers in a time series. This process compares the chop and jiggle sensor signals in the BSM timeline with those in the BSM Operations table which contains the chop beam ID (+Y, -Y) and JiggleID for a given chop and jiggle BSM sensor value (See Figure 6

For point source observations, the source will appear on three detectors in each array during the observation. These are designated as the "prime", "upper" and "lower" detectors in this document. But all detectors will be processed in the same way by the pipeline up to Section 4.2.15, which only operates on the prime, upper and lower detectors where the source appears. 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.

The BSM timeline runs a lot faster (125Hz) than the detector timeline (16Hz), so it will contain about five times more points per time interval. Allocation of a chop position to a detector readout is done in the demodulation step.

Inputs: BSM Timeline

Sample Time	Description	On board time Sampled at 125Hz
	Format	Double Floating Point
	Units	s
chop sensor position jiggle sensor position	Format	Integer
	Units	ADU

Calibration files: BSM Operations Table

BSM Operations Table (SCalPhotBsmOps)	Description	Separate table for each observation mode with columns to match chop and jiggle IDs with sensor positions
Column1: chop ID Column2: jiggle ID	Format	String Integer
	Units	None
Column 3: chop sensor position Column 4: jiggle sensor position Columns 5 & 6: low and high tolerances for matching chop positions with sensor values Columns 7 & 8: low and high tolerances for matching jiggle positions with sensor values	Format	Integer
	Units	ADU

Outputs: Chop Jiggle Timeline

Start Time	Description	Start time at this jiggle position
	Format	Double Floating Point
	Units	s
End Time	Description	End time at this jiggle position
	Format	Double Floating Point
	Units	s
chop ID jiggle ID	Format	String ? Integer
	Units	None
chop sensor position	Format	Integer

jiggle sensor position		
	Units	ADU

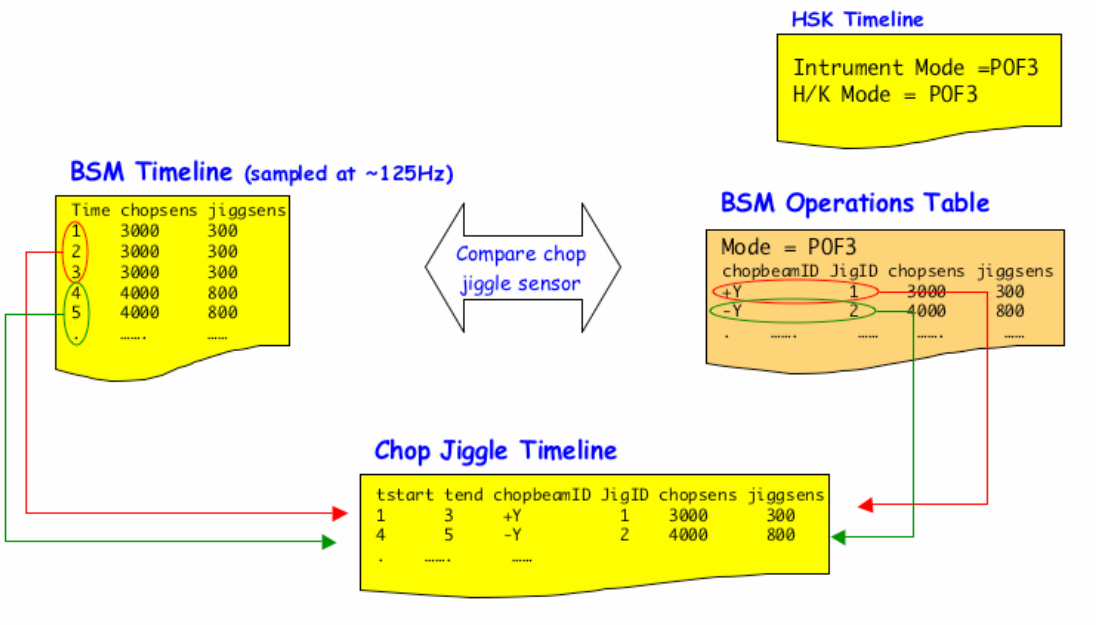

Figure 6: Creation of Chop-Jiggle Timeline from Extract Chop and Jiggle Positions module

4.2.2 Compute BSM Angles

Description: This module converts the chop and jiggle positions into the relative offsets in angle (in degrees or arcseconds 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 also be noted that the BSM timeline runs a lot faster than the detector timeline (125Hz compared to 16Hz) and therefore will contain more points per time interval (typically by about a factor of five). This process takes the BSM timeline that contains the BSM sensor values with sample time and extracts the Y,Z angles from the BSM Positions Table by comparing the sensor values found in the the BSM Positions Table as described in Figure 7.
The product meta data contains rest position of BSM in sensor units

Inputs: BSM Timeline

Sample Time	Description	On board time Sampled at 125Hz
	Format	Double Floating Point
	Units	s
chop sensor position jiggle sensor position	Format	Integer
	Units	ADU

Calibration files:
The BSM Position Table provides calibration between the sensor signal and angle on sky in focal plane

Column 1: Chop sensor signal	Format	Integer
	Units	ADU
Column 2: Jiggle sensor signal	Format	Integer
	Units	ADU
Column 3: Focal Plane Y angles	Format	Rational numbers specified to 8 significant figures

	Units	arcseconds
Column 4: Y angle errors	Format	Rational numbers specified to 8 significant figures
	Units	arcseconds
Column 5: Focal Plane Z angles	Format	Rational numbers specified to 8 significant figures
	Units	arcseconds
Column 6: Z angle errors	Format	Rational numbers specified to 8 significant figures
	Units	arcseconds

Outputs: BSM Angles Timeline

Column 1: Sample Time	Description	On board time Sampled at 1Hz
	Format	Double Floating Point
	Units	s
Column 2: Focal Plane Y angles	Format	Rational numbers specified to 8 significant figures
	Units	arcseconds
Column 3: Y angle errors	Format	Rational numbers specified to 8 significant figures
	Units	arcseconds
Column 4: Focal Plane Z angles	Format	Rational numbers specified to 8 significant figures
	Units	arcseconds
Column 5: Z angle errors	Format	Rational numbers specified to 8 significant figures
	Units	arcseconds

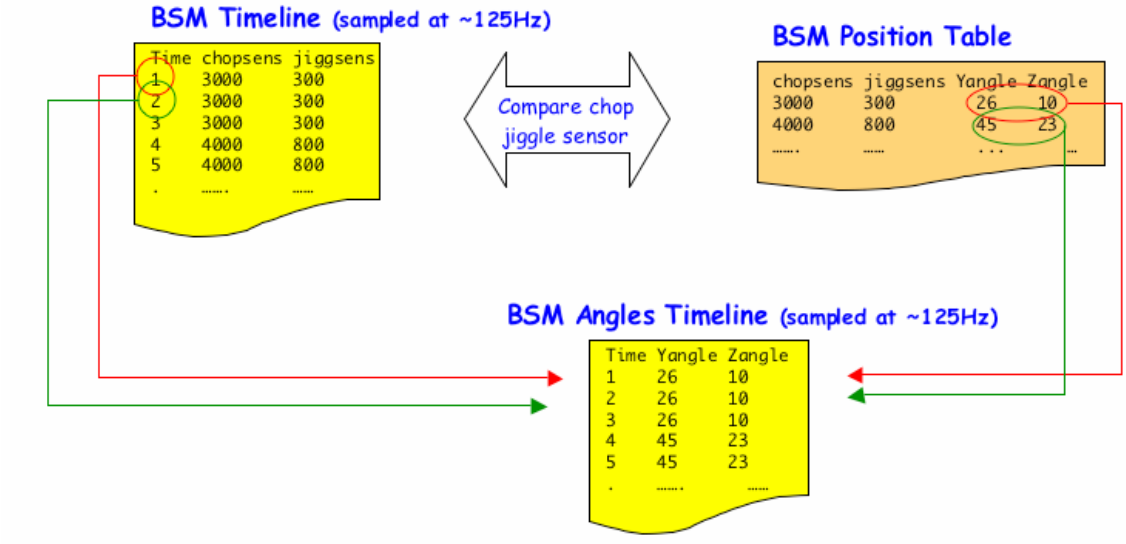

Figure 7: Creation of BSM Angles Timeline from Compute BSM Angles Module

4.2.3 First Level Deglitching

Description: The baseline plan is that this module is the same as for the scan-map pipeline. In the case of the jiggle-map pipeline, it could be turned off or omitted in the first instance as 2nd level deglitching will still pick up anomalies.

Inputs: Level 0.5 Photometer Detector Timeline (PDT) Product
Bias Voltage Flag from PDT meta data (glitch table values may depend on bias voltage)

Calibration files:

Glitch Table (ScalPhotGlitch)	Description	Separate files for each bias setting
	Format	
	Units	

Outputs: Deglitched PDT with glitch flags raised in the mask table

4.2.4 Creation of the SPIRE Pointing Product

This module is the same as described in Section 4.1.2.

4.2.5 Associate Sky Position

This module is the same as described in Section 4.1.10.

4.2.6 Electrical Crosstalk Removal

This module is the same as described in Section 4.1.4.

4.2.7 Convert to flux density

This module is the same as described in Section 4.1.6.

4.2.8 Demodulate

The demodulation of the chopped detector timelines is carried out in accordance with AD1 Sections 4.3 and 6.4. Note that

(i) detectors which are chopped out of the instrument field of view during the observation will have invalid data and will be flagged accordingly;
(ii) for each used detector there are three valid samples in each chop half cycle to be averaged – the second, third and fourth (AD1 Section 4.3);

The sky positions and their naming conventions are shown in Figure ***, which corresponds to one jiggle position for one detector.

For the purpose of demodulation, each detector sample is uniquely defined by:

Nod cycle number	k = 1 – N_{nod}	; no. of nod cycles at each jiggle position (normally = 1)
Telescope nod position:	n = 1 or 2	; A or B
Jiggle position number:	j = 1 – N_{jig}	; N_{jig} is typically 7 or 64
Chop cycle number:	c = 1 – N_{chop}	; N_{chop} is typically 4 for 64 pt jiggle; 16 for 7-pt
BSM position:	b = 1 or 2	; L or R
Sample number within the chop half cycle:	s = 1 – 4	

Note that each detector sample is associated with a unique sky position corresponding to (n, b)

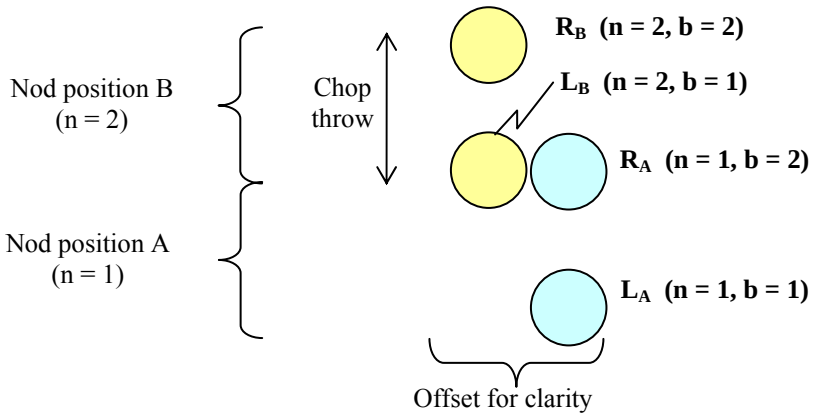

Figure 8: Sky positions observed by a detector during a chopped and nodded observations.

For each used detector, the module computes a demodulated flux density level for each chop cycle as follows:

For each nod cycle (k = 1 – N_{nod})
  For each nod position (n = 1, 2)
    For each jiggle position (j = 1 - N_{jig})
      For each chop cycle (c = 1 – N_{chop})

Averaged flux density for left beam (b = 1): $\overline{S}_{k,n,j,c,1} = \sum_{s=2}^{s=4} S_{k,n,j,c,1,s}$

Averaged flux density for right beam (b = 2): $\bar{S}_{k,n,j,c,2} = \sum_{s=2}^{s=4} S_{k,n,j,c,2,s}$

Demodulated flux density for chop cycle c: $Sd_{k,n,j,c} = \bar{S}_{k,n,j,c,1} - \bar{S}_{k,n,j,c,2}$

Inputs:

(i) flux density timeline for each detector, $S_l(t)$ (output from Convert to Flux Density module);
(ii) associated sky position timeline for each detector timeline.

The sky position timeline must be provided in (or converted to) a form which allows the following to be identified:

Nod cycle number k = 1 – N_{nod}
Telescope nod position: n = 1 or 2
Jiggle position number: j = 1 – N_{jig}
Chop cycle number: c = 1 – N_{chop}
BSM position: b = 1 or 2
Sample number within the chop half cycle: s = 1 – 4

Calibration files: List of used detectors

Output files:

- For each detector
  For each nod cycle (k = 1 – N_{nod})
  For each nod position (n =1 or 2)
  For each jiggle position (j = 1 – N_{jig})
  For each chop cycle (c = 1 – N_{chop})
  Demodulated flux density for that chop cycle: $Sd_{k,n,j,c} = \bar{S}_{k,n,j,c,2} - \bar{S}_{k,n,j,c,1}$
- Format: sci. notation to the same number of significant figures as the input flux density timeline;
- Units: Jy.

4.2.9 Second Level Deglitching and Averaging

Description: This module is based on AD1, Section 6.5. For each nod cycle, the N_{chop} estimates of the demodulated flux densities in each of the two nod positions can now be deglitched by rejecting outliers and averaging the remaining samples, to produce mean values and an associated uncertainty.

For each detector
For each nod cycle (k = 1 – N_{nod})
For each nod position (n = 1, 2)
For each jiggle position (j = 1 – N_{jig})
If $N_{chop} > 4$ (e.g., N_{chop} = 16 for 7-point jiggle) ; adopt median filtering for de-glitching
$Med_{k,n,j}$ = median of $Sd_{k,n,j,c}$; requires an odd no. of points – discard the last
$\sigma_{k,n,j}$ = std. dev. of $Sd_{k,n,j,c}$
If $\frac{\left|Sd_{k,n,j,c} - Med_{k,n,j}\right|}{\sigma_{k,n,j}} > T$ then discard $Sd_{k,n,j,c}$; T is a selectable threshold factor

Repeat median filter
r = number of rejected samples after two iterations

Deglitched mean: $\overline{Sd}_{k,n,j}$ = average of the remaining N_{chop} – r samples $Sd_{k,n,j,c}$

Standard error: $\Delta Sd_{k,n,j} = \dfrac{\sigma_{k,n}}{N_{chop} - r}$

If N_{chop} = 4 (the minimum value - e.g, for 64-point jiggle);
Adopt a TBD algorithm to be implemented to identify potential outliers
E.g:
(i) Make T very big
or
(ii)
- Compute four std deviations, excluding each point in turn
- If one such value is significantly lower than the other three (criterion TBD), the excluded point is an outlier
- Output the mean and standard error for the set of valid points

Inputs: The appropriate outputs from the Demodulate module

Calibration files:

De-glitching threshold, T	Description	One value for each array; typical value = 3; possibly different values for different N_{chop} values
	Format	Rational number to two significant figures
	Units	Dimensionless

Outputs:

- For each detector
  For each nod cycle (k = 1 – N_{nod})
  For each nod position (n = 1, 2)
  Demodulated flux density $\overline{Sd}_{k,n,j}$; not associated with any sky position at this stage
  Associated uncertainty $\Delta Sd_{k,n,j}$
- Format: same as input flux densities
- Units: same as input flux densities

4.2.10 De-Nod

Description: This module is based on AD1, Section 6.6. The de-nod process merely takes the difference between the flux densities in the two nod positions to derive the first estimate of the source flux density, S_{S1}.

For each k = 1 – N_{nod}
For j = 1 – N_{jig}

$$S_{S1,k,j} = \frac{1}{2}\left(\bar{S}d_{k,2,j} - \bar{S}d_{k,1,j}\right) \quad ; \quad \text{in-beam source flux density for jiggle position j}$$

$$\Delta S_{S1,k,j} = \frac{1}{2}\left(\Delta S_{k,1,j}{}^{2} + \Delta S_{k,2,j}{}^{2}\right)^{1/2} \quad ; \quad \text{statistical uncertainty}$$

Note that

(i) the computed flux density is associated with the jiggle position on the sky that is common to both nod cycles (b, n) = (1, 2) or (2, 1) in Figure 8 above;
(ii) it corresponds to the flux density from the sky at that position minus the average of the two chop positions on either side.

Inputs: The appropriate outputs from the Second Level Deglitching and Averaging module

Calibration files: None

Output files:

- For each detector
  For each k = 1 – N_{nod}
  For j = 1 – N_{jig}
  $S_{S1,k,j}$ and $\Delta S_{S1,k,j}$
- Format: same as input flux densities
- Units: same as input flux densities.

4.2.11 Optical Crosstalk Removal

Description: This module is the same as described in Section 4.1.9. However the calibration file (optical crosstalk matrix) is not necessarily the same. Note that it may be difficult to implement due to the different BSM positions. As a baseline, we may start with the zero-crosstalk version of the matrix.

Inputs: Output from De-Nod module

Calibration files:

Optical Crosstalk Matrix (SCalPhotOptCross)	Description	*N* x *N* matrix for each array, where *N* is the number of detectors in the array. In the absence of crosstalk, diagonal elements are unity and non-diagonal elements are zero.
	Format	Scientific notation, specified to three significant figures
	Units	Dimensionless

Outputs: Same as input; flux densities now corrected for optical crosstalk.

4.2.12 Average over Nod Cycles

Description: If $N_{\rm nod} > 1$, then a weighted mean and uncertainty is calculated from the separate estimates:

$$\bar{S}_{\rm S} = \frac{\sum_{k=0}^{N_{\rm nod}} \frac{S_{\rm S2,k}}{\Delta S_{\rm S2,k}}}{\sum_{k=0}^{N_{\rm nod}} \frac{1}{\Delta S_{\rm S2,k}}} \quad \text{and} \quad \Delta S_{\rm S} = \left[\frac{1}{\sum_{k=0}^{N_{\rm nod}} \left(\frac{1}{\Delta S_{\rm S2,k}} \right)^2} \right]^{1/2} . \qquad (18)$$

A weighted mean is only legitimate if the individual values and their uncertainties are mutually compatible. In order to avoid vulnerability to any anomalous estimates and to provide a means of identifying any such anomalies, the pipeline therefore preserves and continue to process the results of the individual nod cycles in addition to the averaged result.

Inputs: Outputs of Correct for Optical Crosstalk module.

Calibration files: None?

Format: same as input flux densities;
Units: same as input flux densities.

Outputs: For each jiggle position of each detector,
(i) for each individual nod cycle: $S_{\rm S2,k}$ and $\Delta S_{\rm S2,k}$ (i.e., just the inputs);
(ii) if $N_{\rm nod} > 1$, the weighted average flux density $\bar{S}_{\rm S}$ and its associated uncertainty $\Delta S_{\rm S}$;
(iii) weighted average quality check (TBD)

4.2.13 Time Correction

This is the same procedure described in Section 4.1.11

4.2.14 Level 1 Product For Jiggle Observations

The final Pointed Photometer Product consists of a set of values of flux density per beam for each detector (with associated uncertainty) associated with the RA and DEC of each detector for each jiggle position and for each nod cycle, and in addition the weighted mean flux density and uncertainty, based on all nod cycles. (This Level 1 product can be considered equivalent to the time line data Level 1 product produced by the scan map pipeline in Section 4.1.12)
Output:

Pointed Photometer Product	Description	Table datasets for all detectors in an array with column for jiggle ID (format byte) in each table data set
Signal	Format	Scientific notation to 9 significant figures
	Units	Jy
Signal Error	Format	Scientific notation to 9 significant figures
	Units	Jy
Latitude	Format	Double Precision
	Units	degrees
Latitude error	Format	Double Precision
	Units	degrees
Longitude	Format	Double Precision
	Units	degrees
Longitude error	Format	Double Precision

	Units	degrees
Mask table	Format	32-bit integer
	Units	dimensionless

4.2.15 Derive Point Source Flux Density and Position (Seven-Point)

Description: This is a Level-2 activity. For seven-point jiggle maps there are seven data points for each detector, one for each specific jiggle position. This small map is to be fitted to a model of the beam profile to derive the source flux density and offset with respect to the pointed position.

In the case of point source observations, for a given array three detectors see the source at some time during the observation, as shown in Figure 9. The primary detector sees it in both nod cycles, as illustrated in however, the other two only see it in one of the nod positions.

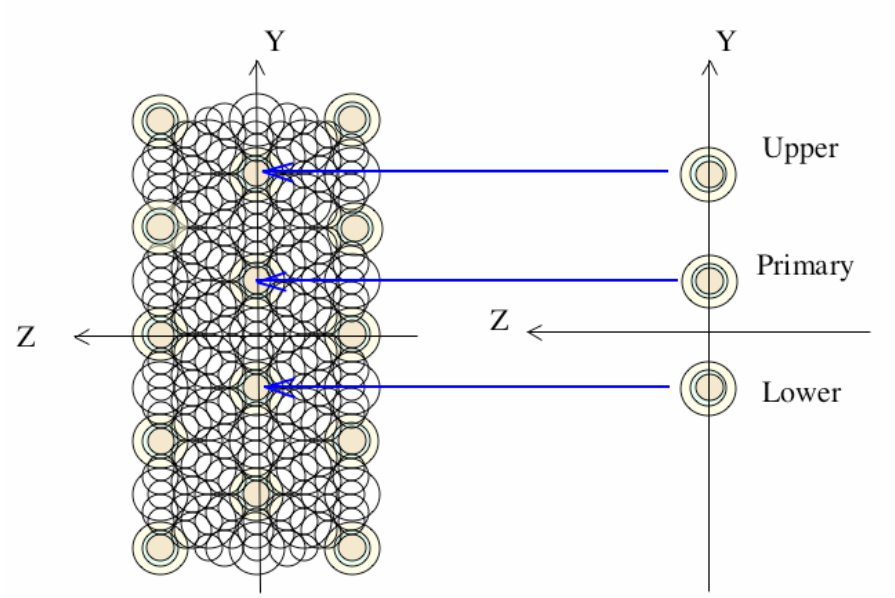

Figure 9: Nominal Detector sets used for point source photometry

In the course of the observation, five different sky positions are viewed by the three detectors (per array). The corresponding signals are indicated in Figure 10 which also shows the positions viewed by the lower detector during the chopping and nodding cycles.

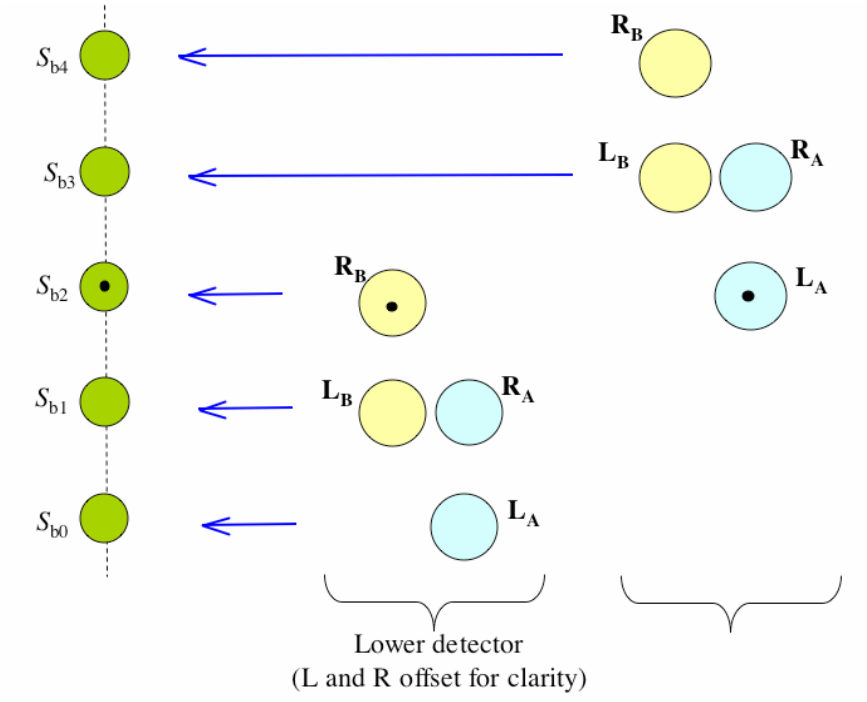

Figure 10: Five sky positions viewed by the three detectors involved in point source photometry, and positions observed by the lower and upper detector set during chopping and nodding.

Let the sky background flux density in the five positions be $S_{b0} - S_{b4}$, and let the source position coincide with S_{b2}. Then, for the lower detector we have:

Demodulated (R – L) signal for nod position A: $S_A = (S_{oR} + S_{b1}) - (S_{oL} + S_{b0})$
Demodulated (R – L) signal for nod position B: $S_B = (S_{oR} + S_{b2} + S_S) - (S_{oL} + S_{b1})$
Difference (de-nodded signal: $S_A - S_B = 2S_{b1} - (S_{bo} + S_{b2}) - S_S$

Similarly, for the upper detector we have:

Demodulated signal for nod position A: $S_A = (S_{oR} + S_{b3}) - (S_{oL} + S_{b2} + S_S)$
Demodulated signal for nod position B: $S_B = (S_{oR} + S_{b4}) - (S_{oL} + S_{b3})$
Difference (de-nodded signel: $S_A - S_B = 2S_{b3} - (S_{b4} + S_{b2}) - S_S$

If the sky background is uniform or linear, then $2S_{b1} = (S_{bo} + S_{b2})$ and $2S_{b3} - (S_{b4} + S_{b2})$, so we have

$$S_A = -S_S \quad \text{and} \quad S_B = -S_S. \tag{19}$$

We therefore get two additional estimates of the source signal, of half the magnitude as for the primary detector (but with the same noise level).

So the point source observation produces three separate estimates of the flux density for a given jiggle position: $2S$ for the primary detector and S for each of the upper and lower detectors. Let the S/N for the central detector be σ, and assume that the three detectors have equal sensitivity. In that case the S/N for the upper and lower detectors is $\sigma/2$. If the three measurements are combined, the overall S/N is thus:

$$\sigma_{\rm tot} = \left[\sigma^2 + 2\left(\frac{\sigma}{2}\right)^2\right]^{1/2} = \left[\sqrt{\frac{3}{2}}\right]\sigma = (1.22)\sigma \qquad (20)$$

The pipeline calculates and quotes the three estimates of the source flux density separately, and provides an option to combine them if the user so desires. Differences in the three measured values may be used to identify non-linear sky gradients.

The seven-point involves observations of seven BSM offset positions, with the central (0,0) position observed twice. For each of the eight positions we have a flux density estimate and its statistical uncertainty, $S_i \pm \Delta S_i$, and an angular offset on the sky with respect to the nominal (0,0) pointed position, ($\Delta\theta_{yi}$, $\Delta\theta_{zi}$).

The pipeline assumes that the source is point-like and carries out a weighted fit of the eight points to a 2-D Gaussian representation of the beam profile. The free parameters for the fit are the peak flux density and the Y and Z positional offsets with respect to the central position (0,0). The results are fitted flux density and offsets, and their associated uncertainties. The estimation of flux density and postion are carried out independently

- for each of the $N_{\rm nod}$ nod cycles
- for each of the three detectors (primary, upper and lower) on each array.

All of these measurements are also combined together by taking a weighted mean.

Note that:

(i) The quality of the positional offset fit will be highly sensitive to S/N. As a rule of thumb, the S/N on the position fit is roughly equivelaent to the S/N on the peak position (so for instance, a S/N of about 20 for PSW should result in an uncertainty of ~ 1" in position since the beam FWHM is close to 20"). For low S/N observations, the position fit will not be reliable. The adopted routine must therefore produce an indication of the reliability of the fit.

(ii) In the case of low-S/N data, the fitted flux density should be essentially equivalent to the weighted sum of the eight measured points (i.e., weighted with respect to the relative beam profile response in the different positions, under the assumption of accurate pointing).

Inputs: TBW

Calibration files:

Beam profiles (SCalPhotBeamProf)	Description	2-D beam profiles for each detector, measured in flight *N* x *N* grid size = 362, 502, 722 for (PSW, PMW, PLW)
	Format	Sampling grid 2" Side: 4 x FWHM = 4 x (18, 25, 36)" = (72,100,144)"
	Units	Dimensionless

Outputs:
- Estimated flux density and stat. error.;
- Estimated position offsets (RA, DEC) (and stat. errors) from the pointed position (provided the S/N is high enough – need a quality flag of some sort to indicate if the position fit is credible

4.2.16 Re-Grid onto Sky (Mapmaking)

Description: This method is identical to that described in Section 4.1.13 except that only the naïve mapmaking process is available (the MADmap mapmaker is not used)

Input: Pointed Photometer Product from Section 4.2.14

Calibration Files: Detector Noise Table (*ScalPhotChanNoise*) as Section 4.1.13

Output: Averaged Pointed Photmeter Product
. One Image map for each array (i.e. 3 maps). WCS information is stored in the meta data

Array Name	Format	String
	Units	None
longitude	Format	Double Precision
	Units	degrees
Longitude error	Format	Double Precision
	Units	degrees
latitude	Format	Double Precision
	Units	degrees
Latitude error	Format	Double Precision
	Units	degrees
Signal	Format	Double Precision
	Units	Jy
Signal Error	Format	Double Precision
	Units	Jy

4.3 Photometer Level 3 processing

4.3.1 Conversion to a Different Source Spectral Index (Colour Correction)

Description:
To be Written

- Level 3 activity.
- Not part of the automatic pipeline as it requires prior information on the source SED.
- Pipeline will calculate flux densities under the assumption of a standard spectral index of -1 (see AD1 Section TBD).
- Correction tables and/or an interactive routine will be provided to allow astronomers to convert to a different spectral index if required.

Calibration files: TBW
Spectral index conversion (SCalPhotSpecIndex)

4.3.2 Source extraction

Description:TBW
- Applies to either scan map or jiggle-map
- Nominally point source extraction only.

5. SPECTROMETER PIPELINE

This section describes how the spectrometer pipeline described in AD2 is implemented.

5.1 Spectrometer Scan Processing (SOF1: Point Source, SOF2: Small Map)

The purpose of the SPIRE spectrometer data processing pipeline is to transform the spectrometer detector samples acquired during a single building block of a single SPIRE spectrometer observation into a set of spectra. In scanning mode, a building block consists of a set of scans of the spectrometer mechanism of the same resolution, with each scan defined as a single movement of the mechanism from one extreme position to the other.

The spectrometer scan processing pipeline is described by Figure 11

The SPIRE spectrometer pipeline takes as input the Level 0.5 Products from the Engineering Conversion processing stage (Section 3). The spectrometer pipeline consists of five fundamentals operations, listed below and outlined as dotted boxes in Figure 11;

1. **Modify Timelines**
The processing modules in this group perform time domain operations on the spectrometer detector samples.

2. **Create Interferograms**
The processing modules in this group merge the timelines of the spectrometer detectors and spectrometer mechanism into interferograms. The spectrometer detector samples are split into different sets, with each set defined by a single scan of the spectrometer mechanism.

3. **Modify Interferograms**
The processing modules in this group perform operations on the spectrometer detector interferograms. These operations differ from those in the "Modify Timelines" group in that they are designed to act on spatial domain data rather than time domain data.

4. **Transform Interferograms**
The processing modules in this group transform the interferograms into a set of spectra via a Fourier Transform process

5. **Modify Spectra**
The processing modules in this group perform operations on the spectrometer detector spectra.

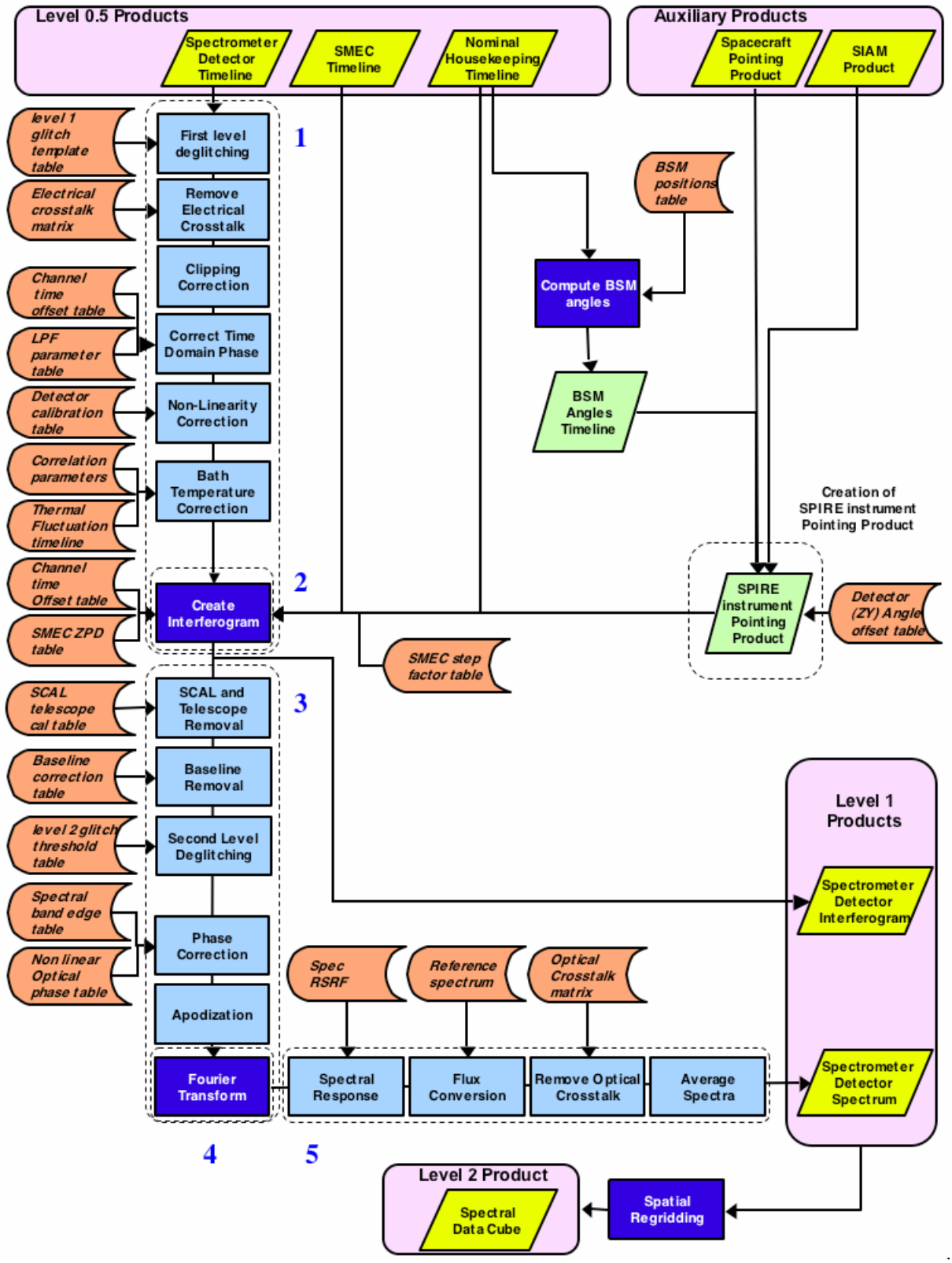

Figure 11: Spectrometer Scan Processing Pipeline. The numbers correspond to the processing steps explained above in Section 5.1

5.1.1 Compute BSM Angles

Description: For the Spectrometer, although the BSM is used for the creation of jiggle maps the instrument does not produce BSM telemetry packets so instead the input comes from the Nominal Housekeeping Timeline (NHK) which contains the BSM sensor values in the "step" parameter. This process extracts focal plane Y,Z angles corresponding to the sample time in the NHK timeline by comparing the sensor signals in the NHK timeline and BSM Positions Table which contains the Y,Z angles for a given chop and jiggle BSM sensor value (see Figure 3). Note that the time in the NHK timeline is sampled at a lower rate of 1Hz and therefore the corresponding BSM Angles Timeline will also be sampled at the same rate.

Inputs: Nominal Housekeeping Timeline

Sample Time	Description	On board time Sampled at 1Hz
	Format	Double Floating Point
	Units	s
Chop sensor signal	Format	Integer
	Units	ADU
Jiggle sensor signal	Format	Integer
	Units	ADU

Calibration files: The BSM Position Table provides calibration between the sensor signal and angle on sky in focal plane

Column 1: Chop sensor signal	Format	Integer
	Units	ADU
Column 2: Jiggle sensor signal	Format	Integer
	Units	ADU
Column 3: Focal Plane Y angles	Format	Rational numbers specified to 8 significant figures
	Units	arcseconds
Column 4: Y angle errors	Format	Rational numbers specified to 8 significant figures
	Units	arcseconds
Column 5: Focal Plane Z angles	Format	Rational numbers specified to 8 significant figures
	Units	arcseconds
Column 6: Z angle errors	Format	Rational numbers specified to 8 significant figures
	Units	arcseconds

Outputs: BSM Angles Timeline

Column 1: Sample Time	Description	On board time Sampled at 1Hz
	Format	Double Floating Point
	Units	s
Column 2: Focal Plane Y angles	Format	Rational numbers specified to 8 significant figures
	Units	arcseconds
Column 3: Y angle errors	Format	Rational numbers specified to 8 significant figures
	Units	arcseconds
Column 4: Focal Plane Z angles	Format	Rational numbers specified to 8 significant figures
	Units	arcseconds
Column 5: Z angle errors	Format	Rational numbers specified to 8 significant figures
	Units	arcseconds

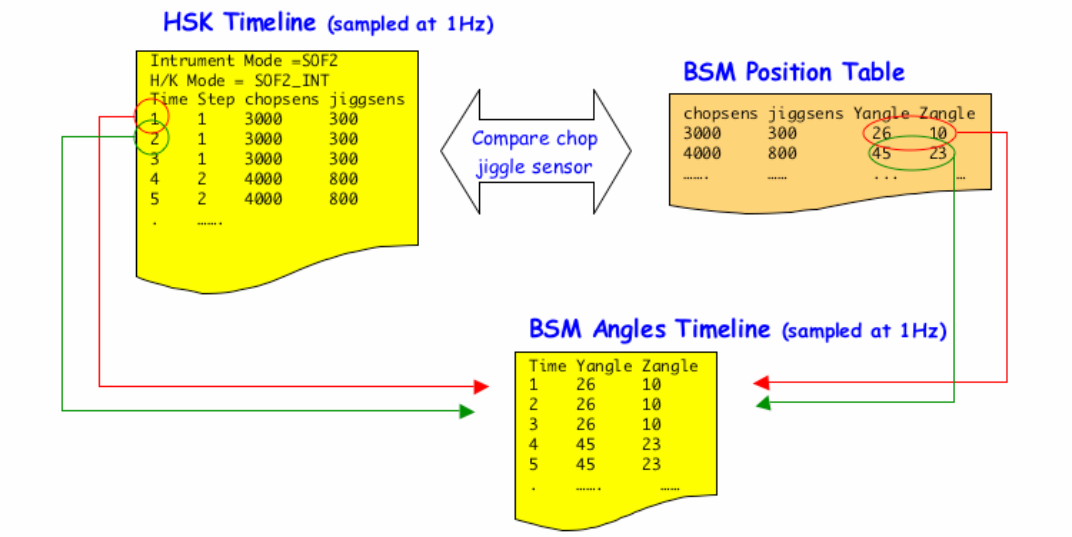

Figure 12: Creation of BSM Timeline from the Compute BSM Angles Module

5.1.2 First Level Deglitching

Description: Glitches due to cosmic ray hits or other impulse-like events in the detectors will be removed using an algorithm based on a wavelet-based local regularity analysis (see Section A.1). This process is composed of two steps: the first step detects glitch signatures over the measured signal; the second step locally reconstructs a signal free of such glitch signatures.
Glitches are detected and flagged in the input Spectrometer Detector Timeline (SDT) product by wavelet analysis assuming that the glitch signature is similar to that of a delta function Each glitch has associated localized wavelet coefficients specific to the glitch. These coefficients are removed and a local, inverse wavelet transform is performed to create an SDT product that is free of glitches. The deglitching algorithm itself is explained in AD2

Inputs: Level 0.5 Spectrometer Detector Timeline (PDT) Product from the Engineering conversion process.
Bias Voltage Flag from SDT meta data (glitch table values may depend on bias voltage)

Calibration files:

Glitch Table (SCalSpecGlitchWavelet)	Description	Separate files for each bias setting
	Format	
	Units	

Outputs: Deglitched SDT with glitch flags raised in the mask table

5.1.3 Removal of Electrical Crosstalk

Description: Electrical crosstalk between detectors is removed under the assumption of;
1. The crosstalk is linear, so that the effects can be characterised by a crosstalk matrix with constant elements
2. the crosstalk from one detector to another involves a negligible diminution of the signal in the primary detector
3. there is no crosstalk accross different arrays.
The procedure for removing electrical crosstalk is to multiply the vector of bolometer voltages by an electrical crosstalk matrix, $\mathbf{C}_{\mathbf{elec}}$, as described in AD2. The vector of electrical crosstalk-corrected signals is given by

$$\mathbf{V}_{\mathbf{corrected}} = \mathbf{C}_{\mathbf{elec}}\mathbf{V} \tag{21}$$

Inputs: SDT from output of previous module

Calibration files:

Electrical crosstalk matrix: *(ScalSpecElecCross)*	Description	*N*x*N* matrix for each array, where *N* is the number of detectors in the array; diagonal elements are unity; in the absence of crosstalk, non-diagonal elements are zero.
	Format	Rational numbers to three significant figures
	Units	Dimensionless

Outputs: Same as input; detector voltages now corrected for electrical crosstalk.

5.1.4 Clipping Correction

Description: The purpose of this processing step is to correct for clipping of the measured signals due to the limited range of the detector ADCs. Clipped signals in the SDT are problematic as they represent missed samples in the timeline. If left uncorrected, missed or erroneous samples can lead to further complications in particular when the timelines are converted into interferograms (Section 3.2.1). Missed or erroneous samples can be corrected in a given SDT timeline as long as the number of consecutive missed samples SDT does not result in a violation of the Nyquist criteria. For sampled signals the Nyquist frequency defines the maximum independent frequency that is sampled. Using the nominal SPIRE spectrometer scanning parameters, the Nyquist frequency exceeds the maximum SLW frequency by a factor of 6 and exceeds the maximum SSW frequency by a factor of 4. As such, the sampled signals can be completely reconstructed even in the absence of five consecutive samples for SLW and three consecutive samples for the SSW detectors.

The clipping Correction follows the following algorithm;

(1) Identify the clipped samples in the SDT timelines. Erroneous samples will have been flagged during the Engineering Conversion process described in Section 3.2 and can be identified by querying the Mask Table in the SDT.

(2) Interpolate the modified SDT timeline. A polynomial of degree eight is applied to the five points before and after those identified as being clipped

(3) Replace the SDT timeline. Replace only those samples, identified as clipped in the original detector timeline, are simply propagated to the resultant timeline.

Inputs: SDT from the previous processing step

Calibration files: None

Outputs: SDT with clipped samples in the timeline corrected.

5.1.5 Time-Domain Phase Correction

Description: The Spire spectrometer detector chain contains a 6-pole Bessel low pass filter (LPF) as well as an additional RC LPF. In addition, to the electronic LPF, the thermal behaviour of the bolometers can also be modeled as a simple RC LPF with a detector-specific time constant dependent upon the detector electronics parameters contained in the Electrical Filter Correction Function calibration file (See AD1, RD2 for details of the low pass filters). These two effects may be combined into a single detector frequency response $H_{TOTAL}(\omega_s)$ where ω_s is the angular frequency of detector signal modulation.

Any LPF in the detector chain will affect the magnitude of the signal recorded by the detectors. In addition, the LPFs will induce a phase shift to the recorded signal. The overall phase (ϕ) imparted by the combined effects of the LPF and the thermal response is then given by the arc-tangent of the ratio of the imaginary and real part of the overall frequency response $H_{TOTAL}(\omega_s)$:

$$\phi_{Total}(\omega_s) = \tan^{-1}\left[\frac{\Im(H_{TOTAL}(\omega_s))}{\Re(H_{TOTAL}(\omega_s))}\right] \tag{22}$$

The phase shift from the combination of the read-out electronics and the thermal response of the detectors manifests itself, to first order, as a delay in time of the recorded signal. This effect is particularly problematic for the spectrometer in scanning mode where any the delay induced by the electronic and thermal phase can lead to errors in the interpolation of the detector signals. The measured detector timelines are corrected by first characterizing the phase shift (as above) then by deriving a time domain phase correction function (TDPCF) which is The shift in the time domain, TDPCF, is quantified as the inverse Fourier Transform of the frequency domain phase shift;

$$TDPCF_t(t) = FT^{-1}\left[e^{-\phi_{Total}(\omega_s)}\right] \tag{23}$$

where FT denotes a Fourier transform. A convolution of the measured detector timelines with the derived time domain phase correction function results in a corrected timeline.

$$V(t)_{shifted} = V(t) \otimes TDPCF(t) \tag{24}$$

This convolution operation shifts the detector signals backward in time (of the order of 30ms) to account for the delay imparted by the combined electronic and thermal response.

Inputs: SDT from the previous processing step

Calibration files:

Electrical Filter Correction Function (*SCalSpecLpfPar*)	Description	contains the resistances and capacitances in the electronic low pass filters that are needed to calculate the frequency response of the detectors
Filter Resistors	Format	Double Precision
		Ohms
Filter Capacitors	Format	Double Precision
	Units	Farads

Channel Time Offsets (*SCalSpecChanTimeOff*)	Description	time offsets due to the time difference between reading out the data from each pixel for individual pixel data relative to the overall 'frame time' that is recorded with the data. The offset depends on whether each array was read out individually or all together. One table for each array.
Column 1: channel name	Format	String
	Units	none
Column 2: single array offset	Format	Double Precision
	Units	s
Column 3: full array offset	Format	Double Precision
	Units	s

Outputs: SDT with timelines corrected for the time delay caused by the phase shift from the combination of the read-out electronics and the thermal response of the detectors.

5.1.6 Non-Linearity Correction

Description: This correction is required to account for changes in the responsivity of the detectors as a function of the intensity of the incident radiation. The form of this correction will be a function that is dependent on the amplitude of the signal itself as in;

$$V(t)_{corrected} = \int_{Vo}^{V} \frac{f(V)}{f(V_r)} dV \tag{25}$$

where $f(V)$, the real detector responsivity, V_r is a reference voltage, and V_0 is a fixed bolometer voltage. The normalized value of $f(V)$ is derived as

$$\frac{f(V)}{f(V_r)} = K_1 + \frac{K_2}{V - K_3} \tag{26}$$

Where the K values are coefficients. A calibration table will contain the values for *Vo*, *K1*, *K2*, and *K3* for each detector. Distinct calibration tables will be used for each detector bias configuration and for each value of *Vo*. and *Vr*. Initially, the quantities in these calibration tables will be based on model predictions but should be updated in orbit.

Input: SDT from the previous processing step.
Bias Voltage Flag (dimensionless integer) from SDT meta data indicating nominal or high bias

Calibration files:

Non-linearity correction coefficients (*alSpecNonLinCorr*)	Description	Contains the coefficients for the non-linearity correction for the SpectrometerOne table for each array.
Column 1: channel name	Format	String
	Units	none
Column 2: zero point voltage	Format	Float
	Units	Volts
Column 3: voltage error	Format	Double Precision
	Units	s
Column 4: zero point voltage flag	Format	Boolean (good=TRUE)
	Units	None
Column 5: K1 coefficient	Format	Float
	Units	none
Column 6: K1 coefficient error	Format	Float
	Units	none
Column 7: K2 coefficient	Format	Float
	Units	Volts
Column 8: K2 coefficient error	Format	Float
	Units	Volts
Column 9: K3 coefficient	Format	Float
	Units	Volts
Column 10: K3 coefficient error	Format	Float
	Units	Volts
Column 11: K value flags	Format	Boolean (good=TRUE)
	Units	Volts
Column 12: Min voltage limit	Description	Calibrated voltage limit (min) per detector channel used to set flag warning of possible flux inaccuracy
	Format	Float
	Units	Volts

Column 10: Max voltage limit	Format	Float
	Units	Volts

- **Output:** SDT timelines with signals with for linearity correction applied

5.1.7 Removal of Correlated `Noise due to Bath Temperature Fluctuations

Description: This process is identical to the algorithm described in Section 4.1.7.

Input:
SDT as output from the previous module
De-glitched thermistor time lines after the deglitching.
Bias Voltage Flag (dimensionless integer) from SDT meta data indicating nominal or high bias

Calibration files:

Temperature drift correction (SCalSpecTempDriftCorr)	Description	See the Photometer Calibration file in Section 4.1.7

- **Output:** SDT with bolometer voltages corrected detector timelines (format: same as input)

5.1.8 Interferogram Creation

Description: This module marks the 2nd basic step (See Figure 11) in the processing of the Spectrometer data. A single building block of a SPIRE spectrometer observation in scanning mode consists of a series of scans of the spectrometer mechanism while the instrument is pointed at a given target. The sampling of the spectrometer detectors and the spectrometer mechanism is decoupled; the two subsystems are sampled at different rates and at different times. In order to derive the source spectrum from the measured data, the spectrometer detector samples must be linked with the position of the SMEC in the form of interferograms. Additionally, the SMEC positions onto which the spectrometer detector signal samples are to be interpolated should be regularly-spaced in terms of optical path difference (OPD). The purpose of this processing step is to ensure proper transformation of the interferogram with the Discrete Fourier Transform.

The creation of the interferograms follows a 2-tier process
(1) Interpolation of the SMEC timeline
(2) Merging of the spectrometer detector and the mapped SMEC timelines.

- **Interpolation of the SMEC timeline**: This process converts the spectrometer mechanism timeline from one that is non-uniform in position to one that is uniform in position
  (i) Establish a common OPD position vector: This step creates a common vector of OPD positions that will be the basis of the interferograms for all of the spectrometer detectors and for all of the scans in the observation. This common position vector will contain samples that are uniformly-spaced in terms of OPD position as well as a sample at the position of zero-path-difference (ZPD). The step size of the common OPD vector is chosen is such a way as to match the sampling rate of the spectrometer detector signal samples. For an SDT sampling rate s [Hz] and a SMEC scanning speed v_{SMEC} [cm/s for the Mechanical Path Difference (MPD)], the position step size, δMPD in units of cm; is given by

$$\delta MPD = v_{SMEC} / s \tag{27}$$

This step is then converted such that it is in terms of OPD by the following relation;

$$\delta OPD = INT[4\delta MPD] \tag{28}$$

where INT[] denotes that the step size is rounded to the nearest integer in units of microns OPD and the factor of four is the nominal conversion between MPD and OPD for a Mach-Zehnder FTS. Using the nominal SPIRE spectrometer settings of s=80Hz, v_{SMEC}=0.05cm/s, results in an OPD step size of 25 microns.

(ii) Map the common OPD position vector to a SMEC position vector for each spectrometer detector: This step maps the common OPD positions for each spectrometer detector established in the preceding positions in units of mechanical path difference. This step involves: a scaling factor, f, that takes into account the step size for a Mach-Zehnder FTS; and a shifting factor, ZPD, which establishes the position of zero optical path difference. Since these quantities are unique to each spectrometer detector, i, this mapping is performed on a detector-by-detector basis as;

$$MPD_i = \frac{OPD}{f_i} + ZPD_i \tag{29}$$

(iii) Parse the measured SMEC timeline into discrete scans: This step splits the full SMEC timeline $z(t_{SMEC})$, from the input SMEC timeline product into a series of discrete timelines, $z_n(t_{SMEC})$ each of which represents one spectrometer scan. The delineation of the SMEC timeline is accomplished by comparing consecutive SMEC position samples and finding those samples where the motion of mirror mechanism changed direction.

(iv) Interpolate the measured SMEC timelines onto the mapped SMEC timelines. The final step in the interpolation of the SMEC timeline is to determine, on a detector-by-detector and scan-by-scan basis, the times when the spectrometer mechanism reached the mapped SMEC positions. Since, for each detector, there is a one-to-one relationship between the mapped SMEC positions and the regularly spaced OPD positions, this step effectively determines the times when the SMEC reached the regularly spaced OPD positions for each detector. The interpolation mapping is carried out as;

$$z_n(t_{SMEC})MPD_i \rightarrow MPD_{n,i}(t_{MPD,i}) \tag{30}$$

- **Merge the spectrometer detector and the mapped SMEC timelines.** This process combines the signal samples from the signal timeline of a given spectrometer detector with the mapped SMEC timelines.

(i) Interpolation of the spectrometer detector timelines: The spectrometer detector signal samples are mapped onto the times corresponding to the regular MPD ($t_{MPD,i}$) positions by way of interpolation. Since there is a one-to-one relationship between these time samples, $t_{MPD,i}$, and the regular MPD positions, MPD_i this interpolation effectively maps, for each detector, the signal samples to the regularly-spaced MPD positions. Moreover, since there is a one-to-one relationship between the regular MPD positions for each detector and the common OPD positions, this step accomplishes the mapping of the signal samples for each detector to the common OPD positions, which is the resultant interferogram that is desired.

$$V_i \rightarrow V_{mapped,i}(t_{MPD,i}) \rightarrow V_{mapped,i}(t_{OPD}) \rightarrow V_{mapped,i}(OPD) \equiv V_{mapped,i}(x) \tag{31}$$

where, x, denotes regularly spaced positions The above 2-tier process is repeated for all spectrometer detectors for each scan of the observation building block. The resultant interferograms are then combined into a single Spectrometer Detector Interferogram (SDI) product. In addition, the mean value of the pointing, $P(t)$, as derived from the input SPIRE Pointing product (SPP, see Section 4.1.2) for the observation building block is assigned to the output SDI product. The assigmanet of the pointing information follows the same methodology outlined in 4.1.10 for the Photometer, **with the exception that in the case of the spectrometer, a range of times rather than a specific time are sent to the SPP** and an average is taken.

Input:
SDT as output from the previous module
SPIRE Pointing Product (see Section 4.1.2)

SMEC Timeline

Sample Time	Format	Double Precision
	Units	s
encoderCoarse	Description	Optical encoder values for coarse position of SMEC
	Format	Double Precision
	Units	cm^{-1}
encoderFine	Description	Optical encoder values for finer position accuracy
	Format	Double Precision
	Units	cm^{-1}
lvdtDCSignal	Description	Used as a back-up for the above. Linear Variable Differential Transformer. This is an absolute measure of the position of the mirror mechanism as opposed to the relative values from the optical encoder. The LVDT has a precision of 0.1microns within +/-3.27mm of ZPD as opposed to 0.01microns for the optical encoder over the entire scan length
	Format	Integer
	Units	none

Nominal House Keeping Timeline,

NHK Timeline	Description	The optical encoder values as a function of sample time. lvdtDCSignal may be used as a back-up
Sample Time	Format	Double Precision
	Units	s
Scan number (SCANS)	Description	Current scan number
	Format	Integer
	Units	None
Scan start position	Description	SMEC position at start of scan
	Format	Double Precision
	Units	cm^{-1}
Scan end position	Description	SMEC position at end of scan
	Format	Double Precision
	Units	cm^{-1}

Calibration files:

Channel Time Offsets Table (*SCalSpecChanTimeOff*)	Description	time offsets due to the time difference between reading out the data from each pixel for individual pixel data relative to the overall 'frame time' that is recorded with the data. The offset depends on whether each array was read out individually or all together. One table for each array.
Column 1: channel name	Format	String
	Units	none
Column 2: single array offset	Format	Double Precision
	Units	s
Column 3: full array offset	Format	Double Precision
	Units	s
SMEC Step Factor Table (*SCalSpecSmecStepFactor*)	Description	Provides the conversion factor between mechanical path difference (MPD) and optical path difference (OPD) as a function of channel
Column 1: channel name	Format	String
	Units	none
Column 2: SMEC step factor	Format	Double Precision
	Units	

Output: Spectrometer Detector Interferogram (SDI) Product

	Description	Separate table for each detector containing the OPD and signal with the detector channel stored in the meta data
OPD	Format	Double Precision
	Units	cm^{-1}
OPD error	Format	Double Precision
	Units	cm^{-1}
Signal	Format	Double Precision
	Units	Volts
Signal error	Format	Double Precision
	Units	Volts
Mask	Format	Integer
	Units	None

5.1.9 SCAL and Telescope Correction

Description: This module marks the beginning of the 3rd basic step (See Figure 11) in the processing of the Spectrometer data. Note that the data is now in the format of the Spectrometer Detector Interferogram (SDI) Product. This module removes the emission (which are modulated signals) from the measured interferogram. Due to the Herschel Telescope and from each of the components of the spectrometer calibrator (SCAL).
The spectrometer calibrator is composed of two controlled emitters, at 2% emissivity (SCAL2), and 4% emissivity (SCAL4). The remaining emitting portion (the other 94%) is referred to as SCAL. Since it is possible for these three components to be at different temperatures (indeed, their temperatures are measured independently) the total SCAL emission is split into a set of three independent interferograms: SCAL, SCAL2, and SCAL4.
Interferograms are corrected by subtracting a blank sky observation from a targeted observation with the SCAL2, SCAL4 calibrators set to the same temperatures as was the case for the on-source observation.

$$V_{observation} = V_{source} + V_{telescope} + V_{SCAL} + V_{SCAL\,2} + V_{SCAL\,4} \tag{32}$$

$$V_{reference} = V_{telescope} + V_{SCAL} + V_{SCAL\,2} + V_{SCAL\,4} \tag{33}$$

$$V_{source} = V_{observation} - V_{reference} \tag{34}$$

The result of the Telescope and SCAL correction step will be a set of interferograms for each spectrometer detector (One per SMEC scan per detector), stored in an SDI product. The resultant interferograms contain the radiation from the astronomical source (note that the resultant interferograms still contain effects due to the transmission through the instrument, and spectral response effects, etc which will be removed in later processing steps).

I**nput:** SDI product from previous processing step

Calibration files:

SCAL Reference Interferogram (*SCalSpecInterRef*)	Description	Blank sky observation interferogram. one calibration interferogram per detector, per spectral resolution, per bias voltage (for strong sources). There may also be one interferogram per scan direction as it may be advantageous to ensure that systematics are not introduced due to SMEC scan direction. There maybe a family of such interferograms for a range of Telescope/SCAL temperatures
OPD	Format	Double Precision
	Units	cm^{-1}
OPD error	Format	Double Precision
	Units	cm^{-1}

Signal	Format	Double Precision
	Units	Volts
Signal error	Format	Double Precision
	Units	Volts
Mask	Format	Integer
	Units	None

Output: SDI product with background subtracted interferograms

5.1.10 Interferogram Baseline Correction

Description: The overall intensity incident on the SPIRE spectromter detectors can be separated into two components: a component that is constant as a function of OPD and a component that is modulated as a function of OPD. As the offset term does not contain any spectral information, it may be removed without affecting the source spectrum. On a detector-by-detector and scan-by-scan basis, the baseline correction algorithm evaluates and removes the offset portion of the derived interferogram The preferred manner to evaluate the offset is to fit a fourth-order polynomial to the measured interferogram to obtain a fit to the baseline which is then simply subtracted from the input interferogram.

$$V_{corrected} = V_{input} + V_{baseline} \tag{35}$$

Input: SDI product from previous processing step

Calibration files: None

Output: SDI with baseline removed

5.1.11 Level 2 Deglitching

Description: Localized artifacts in the interferograms, glitches, pose a serious problem for Fourier Transform Spectrometer observations. As such, a glitch that affects as few as one interferogram sample can adversely affect each and every spectral component. Glitches in an interferogram must therefore be identified and removed prior to transformation in order to avoid unwanted spectral artifacts. Glitches are identified for each spectrometer detector by comparing, on a OPD-position-by-OPD-position basis, the samples from one scan to those from all other scans in the same observation. The samples that deviate more than a prescribed amount from the median are flagged as glitches.
For each given detector taking all N of its scans in the input product, at each x_k in the common OPD vector for these scans, the voltage for all N scans is compared. If an outlier is found in the jth scan ($V_j(x_k)$), it is replaced with the average voltage of the other $N-1$ scans (i.e. all scans n, except where where $n=j$) at that x_k.

$$V_{\mathrm{deglitched}\ j}(x_k) = \frac{1}{N_{scans}-1}\sum_{n=1,n\neq j}^{N_{scans}} V_n(x_k) \tag{36}$$

where, j is the current deglitched scan, n is the scan number, k is a particular OPD value and x denotes regularly spaced positions.
Note: interferogram deglitching module rely on a statistical analysis of the measured interferograms. As such, a minimum number of six interferograms per building block will be required so that these statistics will be meaningful.

Input: SDI product from previous processing step

Calibration files:

Glitch Threshold Table ( )	Description	
	Format	
	Units	

Output: SDI with interferograms corrected for glitches, with the value of the replacement sample determined by the average of the non-glitch samples from the other observed interferograms at that position and glitch flag raised in mask table.

5.1.12 Phase Correction

Description:
The symmetry of a Fourier Transform spectrometer theoretically implies that the should exhibit even symmetry. Since the spectrum of an evenly symmetric interferogram contains only real components, it is therefore expected that the phase should be zero for all spectral components. However, the presence of dispersive elements and the possibility that the position of zero path difference not being sampled can result in an interferogram whose *signal* samples are not symmetric about the ZPD. The spectrum calculated from this sort of asymmetric interferogram will contain both real and imaginary components and therefore a non-zero phase. The phase correction module is separated into two components: the first step identifies whether any phase is present in the measured interferogram and the second step removes this phase.

(1) Phase Identification. **In order to make this determination, the spectrum of the double-sided portion B_{DS} is computed for each interferogram below (see appendix in AD2). Note that this involves a Fourier Transform of the interferogram using the processing step described in Section 5.1.14;**

$$B_{DS}(\sigma) = FT\left[V(x)\Big|_{-L}^{L}\right] = \int_{-L}^{L} V(x) e^{-i2\pi\sigma x} dx \tag{37}$$

where σ is the wavenumber (in cm^{-1}), x is regularly spaced positions along the OPD (in cm) and L is the maximum OPD displacement from the position of ZPD. The phase of the computed spectrum may be evaluated for each spectral component or wavenumber by;

$$\phi_{DS}(\sigma) = \tan^{-1}\left[\frac{\Im(B_{DS}(\sigma))}{\Re(B_{DS}(\sigma))}\right] \tag{38}$$

(1) Phase Removal. A fourth-order polynomial fit to the measured in-band phase rather than the measured phase itself is used in this process. The basis for using the fitted phase rather than the calculated phase is that, by doing so, the noise associated with the imaginary portion of the computed spectrum remains in the imaginary domain. If the phase is stable and the noise is primarily random, the usage of a fitting function can lead to an increase in the resultant signal-to-noise ratio by a factor of square-root of 2. A phase correction function (PCF) is then derived from the fitted phase, ϕ_{fit}, for each interferogram;

$$PCF(\phi) = e^{-i\phi_{fit}(\sigma)} \tag{39}$$

The PCFs are then applied multiplicatively to the spectra computed from each of the interferograms in the input SDI product. phase correction process. If the observing mode is low- or medium-resolution, this represents the final step in the phase correction process. On the other hand, high-resolution observations require an extra step. The phase correction for the high-resolution interferograms, proceeds in the interferogram (spatial) domain where a convolution of the measured interferogram and the inverse FT of the PCF is performed;

$$
\begin{aligned}
V(x)_{corrected} &= V(x) \otimes FT^{-1}[PCF(\sigma)] \\
&= V(x) \otimes FT^{-1}[e^{-i\phi_{fit}(\sigma)}] \\
&= V(x) \sum_{x=-l}^{l} \left(FT^{-1}[e^{-i\phi_{fit}(\sigma)}] \right)(x)
\end{aligned}
\tag{40}
$$

Input: SDI product from previous processing step

Calibration files:

Spectral Band Edges Table (*SCalSpecBandEdge*)	Description	Contains frequencies at which overall spectral response of detectors falls to 50% of the average in-band spectral response for all channels (one table for each array)
Channel Name	Format	String
	Units	none
Low frequency edge	Format	Double Precision
	Units	cm^{-1}
High feequency edge	Format	Double Precision
	Units	cm^{-1}

Non-linear (Optical) Phase Correction Table (*SCalSpecNlp*)	Description	Contains known non-linear optical phase as function of wavenumber.. The wavenumber grid ranges from 0 to 200 cm-1 and must be regularly spaced. The phase-error will be different depending on input and output port of the FTS, making it necessary to store four different phases. (one table for each array)
Wavenumber	Format	Double Precision
	Units	cm^{-1}
Telescope Phase	Format	Double Precision
	Units	None
Telescope Phase Error	Format	Double Precision
	Units	None
SCAL Phase	Format	Double Precision
	Units	None
SCAL Phase Error	Format	Double Precision
	Units	None

Output: SDI Product with interferograms corrected for any phase present in the input interferogram.

5.1.13 Apodisation

Description: The natural instrument line shape (ILS) for a Fourier Transform spectrometer is a cardinal sine, or Sinc function. If the source signal contains features at or near the resolution of the spectrometer, the ILS can introduce secondary maxima in the spectra. This module may be used to reduce these secondary maxima at the cost of reducing the resolution of the resultant spectrum.

Apodization is performed by multiplying the input interferograms on a detector by detector and on a scan-by-scan basis with a tapering or apodizing function.

$$V(x)_{apodized} = V(x) \times f_{APODIZE}(x) \tag{41}$$

The result is an SDI product that contains apodised interferograms.
The application of an apodization function is also the current preferred method for fringe correction. The effect of channel fringes on spectrometer data is similar to that of glitches and if left uncorrected the channel fringes will contaminate the measured spectrum. While apodization is effective at removing the spectral artifacts due to the channel fringes, its application results in a reduction of the observed spectral resolution. Given that the fringe features appear at the extreme high-resolution OPD end for the SLW array and at the extreme medium-resolution end for the SSW array, the reduced resolution is not expected to be significant for those observing modes.

Input: SDI product from previous processing step

Calibration files: None

Output: SDI product containing apodised interferograms

5.1.14 Fourier Transform of Interferograms

Description: This module marks the beginning of the 4th basic step (See Figure 11) in the processing of the Spectrometer data. To this point the processing best performed in the the interferogram domain have been implemented. Further processing should now be implemented in the spectral domain and the purpose of the Fourier Transform module is to transform the set of interferograms from a spectrometer observation into a set of spectra. This process is capable of transforming both double-sided and single-sided interferograms (see AD2 Appendix A for the definition of double-sided and single-sided interferograms).

(1) **Double-Sided Transform (DS).** For the double-sided transform, each interferogram in the SDI is examined and only the double-sided portion of the interferogram is used to compute the resultant spectrum. The resultant spectra will contain both real and imaginary components;

$$B(\sigma)_{DS} = FT\left[V(x)\Big|_{-L}^{L}\right] = \int_{-L}^{L} V(x) e^{-i2\pi\sigma x} dx \tag{42}$$

where σ is the wavenumber and L is the maximum OPD displacement from the position of ZPD. In this case the discrete Fourier transform that is used to compute the spectral components takes the form;

$$B(\sigma_k)_{DS} = FT\left[V(x)\Big|_{-L}^{L}\right] = \sum_{x_k=0}^{N-1} V(x_k) e^{-i\frac{2\pi\sigma_k x_k}{N}}$$

where the σ_k and x_k denote a summation rather than integral for the Fourier Transform.

(2) **Single-Sided Transform (SS).** In the case of the single-sided transform, only those interferogram samples to one side of the position of ZPD are considered. The spectra that result from the single-sided transform therefore contain only real components.

$$B(\sigma)_{SS} = FT\left[V(x)\Big|_{0}^{L}\right] = \int_{0}^{L} V(x) e^{-i2\pi\sigma x} dx \tag{44}$$

In this case the discrete Fourier transform that is used to compute the spectral components takes the form;

$$B(\sigma_k)_{SS} = FT\left[V(x)\Big|_{0}^{L}\right] = \sum_{x_k=0}^{N-1} V(x_k) Cos\left(\frac{2\pi\sigma_k x_k}{N}\right)$$

For both the single-sided and double-sided transforms the wavenumber grid onto which the spectrum is registered is calculated based on the interferogram sampling rate (ΔOPD) and on the maximum OPD displacement from the position of ZPD, L. The spacing between independent spectral samples is given by $\Delta\sigma=1/2$. The spacing between spectral samples can be modified by padding the interferogram with zeroes. This procedure does not add any information to the spectrum but allows for an easier comparison between observations. In this case, a zero-padded (ZP) interferogram is given by;

$$V(x)_{ZP} = V(x)\Big|_0^L, 0\Big|^{L \le x \le L_{ZP}} \tag{46}$$

The corresponding spectral sampling interval is given by $\Delta\sigma = 1/2$, and the resultant spectrum of the zero-padded interferogram is given by;

$$B(\sigma_k)_{ZP} = \sum_{x_k=0}^{N_{ZP}-1} V_{ZP}(x_k)\, e^{-i\frac{2\pi\sigma_k x_k}{N_{ZP}}}$$

The output of this process will create a Level-1 Spectrometer Detector Spectrum (SDS) product.

Input: SDI product from previous processing step

Calibration files: None

Output: Spectrometer Detector Spectrum (SDS) Level 1 format Product

	Description	Separate table for each detector spectrum and each SMEC scan. In addition, the interferogram number (stored as a long integer), Scan Direction (string), Resolution (Double Precision) and Scan Number (long integer) are stored in the Product meta data.
Pixel Name	Format	String (stored in table dataset metadata)
	Units	None
Wavenumber	Format	Double Precision
	Units	cm^{-1}
Signal	Format	Double Precision
	Units	Volts
Signal error	Format	Double Precision
	Units	Volts
Mask	Format	Integer
	Units	None

5.1.15 Spectral Response Correction

Description: This module marks the beginning of the 5th and final basic steps (See Figure 11) in the processing of the Spectrometer data. This and the following modules modify the spectra created from the previous Fourier Transform step. The end result of these spectral modifying processing steps will be a Level 1 SDS product that contains a single, flux-calibrated, averaged spectrum for each spectrometer detector.
The Spectral Response Correction module will remove from each measured spectrum for each detector in the input SDS product the relative spectral response function (RSRF) for that particular detector by dividing the spectrum by the RSRF;

$$B(\sigma)_{RSRFcorrected} = \frac{B(\sigma)}{RSRF(\sigma)} \tag{48}$$

where σis the wavenumber in cm^{-1}.The RSRF curves for each detector (RSRF(σ)) represent the relative transmission of the SPIRE instrument from the Telescope port to any given detector.

Input: SDS Product from previous processing step

Calibration files:

Spectrometer RSRF (SCalSpecRsrf)	Description	For each detector there will be a RSRF scaling factor per wavenumber bin (for each of the low, medium and high commanded spectral resolutions)
Wavenumber	Format	Double Precision
	Units	cm^{-1}
RSRF Scaling Factor	Format	Double Precision
	Units	None

Output: SDS Product with signal spectrum divided by the RSRF

5.1.16 Flux Conversion

Description: The flux conversion module translates each of the measured spectra in the input SDS product from voltage quantities (B(σ)) with units of Volts/cm^{-1} to optical power quantities (I(σ)) with units of [either Watts/m^2/cm^{-1} or Janskys, TBD] on a wavenumber-by-wavenumber basis multiplicatively;

$$I(\sigma)_{flux} = B(\sigma)_{volts} \times f(\sigma) \tag{49}$$

Note: The exact manner by which the conversion curves, *f(σ),* will be derived is still TBD but the current baseline is to perform a calibration observation of a source of known flux. The spectrum derived for each spectrometer detector will then be used as the conversion curve for this module.

Input: SDS Product from previous processing step

Calibration files:

Spectrometer Reference Spectrum	Description	A reference spectrum containing the conversion from volts to flux units.
Wavenumber	Format	Double Precision
	Units	cm^{-1}
Conversion factor	Format	Double Precision
	Units	$W/m^2/cm^{-1}$ /V or Jy/V

Output: SDS Product with signal spectrum converted from volts to Flux units

5.1.17 Optical Crosstalk Removal

Description: Optical crosstalk will be characterised by a crosstalk matrix, C_{opt}, analogous to the electrical crosstalk matrix described in Section 5.1.3.

$$I_{\text{corrected}} = \mathbf{C}_{opt} I \tag{50}$$

Note that however, unlike the case of electrical crosstalk, the diagonal elements are not necessarily equal to unity since optical crosstalk involves loss of power from the primary detector.

Input: SDS Product from previous processing step

Calibration files:

Optical Crosstalk Matrix	Description	*N* x *N* matrix for each array, where *N* is the number of detectors

(SCalSpecOptCross)		in the array. In the absence of crosstalk, diagonal elements are unity and non-diagonal elements are zero.
	Format	Scientific notation, specified to three significant figures;
	Units	Dimensionless

Output: SDS Product with spectrum signal corrected for optical crosstalk.

5.1.18 Spectral Averaging

Description: This module also computes, on a wavenumber-by-wavenumber basis for each spectrometer detector, the average of the spectral intensities across all scans for the k^{th} wavenumber;

$$\langle I(\sigma_k)\rangle = \frac{1}{N_{scans}} \sum_{n=1}^{N_{scans}} I_n(\sigma_k) \tag{51}$$

The module also computes, on a wavenumber-by-wavenumber basis for each spectrometer detector, the uncertainty in the spectral average calculated as the standard deviation of the spectral components;

$$\delta I(\sigma_k) = \sqrt{\frac{1}{N_{scans}-1} \sum_{n=1}^{N_{scans}} I_n(\sigma_k) - \langle I_n(\sigma_k)\rangle^2} \tag{52}$$

Input: SDS Product from previous processing step

Calibration files: None

Output: Level 1 SDS Product with spectra averaged from all scans for this building block (i.e. from input SDS product).

Pixel Name	Format	String (stored in table dataset metadata)
	Units	None
Wavenumber	Format	Double Precision
	Units	cm^{-1}
Signal	Format	Complex
	Units	$Watts/m^2/cm^{-1}$
Signal error	Format	Double Precision
	Units	$Watts/m^2/cm^{-1}$
Mask	Format	Integer
	Units	None

5.2 Spectrometer Level 2 Processing

The pipeline modules that follow in this section describe the operations on the Level-1 SDS products created in the preceding section. The result of the Level 2 processing steps will be a Level-2 Spectral Cube product.

5.2.1 Spatial Regridding

Description: The spectrometer pipeline produces one spectrum per detector. The spatial distribution of the spectra in the astronomical region of interest from a single pointing follows approximately a honeycomb pattern as per the design of the detector arrays (See Figure 13). For an observation performed at intermediate or full spatial sampling, the set of

Level 1 SDS products is interpolated onto a hyperspectral data cube that is equidistantly sampled in the two spatial dimensions while leaving the equidistant grid along the spectral dimension unchanged. This operation will not be applied in the sparse spatial sampling mode as the spatial sampling in that mode will not meet the Nyquist criteria.

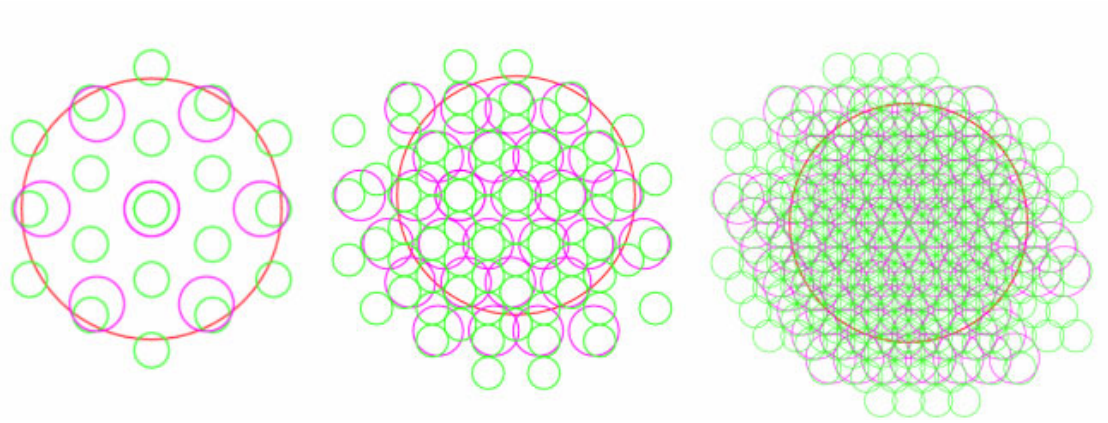

Figure 13: Astronomical footprint of the SPIRE detector arrays in from left to right, sparse, intermediate & full spatial sampling mode

An algorithm for the interpolation of spectral data collected at non-uniformly sampled locations has been identified and a normalized convolution algorithm has been implemented [See AD02 for details]. The algorithm iterates along the spectral dimension and evaluates a two-dimensional convolution of the measurements at given sky positions with a separable two-dimensional kernel describing the field of view of the detectors. Ground-based measurements have shown that the of the beam of the SPIRE spectrometer detectors vary in a non-linear fashion with frequency between 15.5 and 17.5 arcsec for the SSW and between 31 and 41 arcsec for the SLW band. The interpolation algorithm will take such a frequency-dependent beam size into account. It does, however, assume that the beam sizes of all detectors are identical. The suitability of this procedure remains to be verified.

Input: Level 1 SDS Product

Calibration files: None

Output: Level 2 Hyperspectral Cube.

	Description	
	Format	
	Units	

6. OPERATIONAL DAY PROCESSING

In the SPG framework, the pipeline works on the data from a single observation. This assumption is needed to be able during operations to run the pipeline on many observations at same time on a parallel machine (a grid). The primary goal of the Operational Day Processing (ODP) stage is to generate those calibration products needed by the pipeline that can be produced only on a operational day basis and not from the single observation. An example for SPIRE is the detector signal offset history. The Operational Day Processing is intended to be performed before processing any observation of the specific operational day. The second goal of the ODP is to produce basic trend analysis products that are needed for monitoring the instrument health and performance. This also includes the processing of data (in normal operation consisting only of housekeeping) taken by the instrument outside observations or when the instrument is not prime. In fact, data outside observations are not processed by the SPG so the ODP is the only stage in which these data can be processed in a systematic way. It is expected that no auxiliary data should be assumed to be available for running the ODP stage. The Operational Day Processing is clearly a potential bottle neck for the SPG and the systematic reprocessing (because no observation can be until the ODP is completed) so it should be kept as efficient as possible.
Two basic calibration products have to be created during the ODP,

- the DPU reset history
- the detector offset history

The OPD is implemented as outlined below;
(1) All Nominal Housekeeping dataframes from the entire operational day (OD) are extracted to create a single level 0 Raw Nominal Housekeeping Timeline (RNHKT).
(2) The Engineering Conversion pipeline is used to convert the RNHKT into a NHK timeline which will be stored as a Trend Analysis product.
(3) Steps 1 and 2 are repeated for the Critical Housekeeping (CHK). Note that the DPU reset history is not required for steps (2) & (3) since the NHK and CHK telemetry packets contain only a single frame. There is no frametime for these packets, so the time that is associated with the sampling of the parameters is simply the telemetry packet time
(4) The Treset History Processing Pipeline (Section) extracts the TRESET column from the from the NHKT taking only the unique values creating the DPU reset history for the entire OD.
(5) All photometer and spectrometer offset frames from the OD are extracted to create raw photometer/spectrometer offset timelines In this step, the reset history calculated from step (4) is required.
(6) The raw offset timelines are processed by the Engineering to effect time reordering and conversion. In this step, the reset history calculated from step (4) is required. Offset values in the output are in ADUs.
(7) The Extract Offset Process (Section 6.2) extracts the offsets from the offset timelines to create the Offset History product for each instrument. In this step, the reset history calculated from step (4) is required.
(8) The data frames for all building blocks during the OD where PCAL flashes where performed are extracted to create the Level 0 raw photometer/spectrometer detector timelines (PDT/SDT) and raw SCU timelines for each building block.
(9) The raw photometer/spectrometer detector and SCU timelines are processed by the Engineering Conversion to create the Level 0.5 S/PDT and SCUT products. In this step, the reset history calculated from step (4) is required.
(10) The Pcal processing (see Section 7) in made on each one of these block containing the detector data to produce a Pcal calibration product. These calibration products are then saved and will be attached to the calibration context of each observation.

6.1 Treset History Processing Pipeline

Science telemetry packets (e.g. detector packets, BSM, SMEC, SCU and offset packets) contain data stored in several frames (see RD7 for details). The time at which each SPIRE frame was sampled is contained in a 32 bit value which corresponds the number of (3.2 microsecond) clock ticks since the last DPU reset time, Treset. 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. Figure 14 shows the Treset History Processing Pipeline

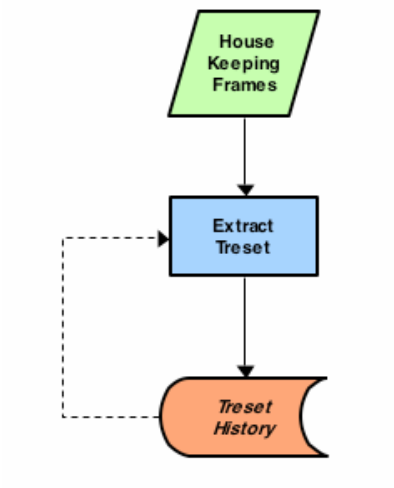

Figure 14: Treset History Processing Pipeline

6.1.1 Extract Treset Process

Description: A step in the ODP is responsible to extract these Treset times creating a single DPU reset history for each Operational Day which will be made available to the standard pipeline. The Extract Treset Process is used to make these Treset times available to the data processing by creating a single Treset History which will eventually cover the whole mission. The process will extract the Treset times and append to (or update) the Treset History. It is expected that this pipeline will be run at appropriate intervals (e.g. between cooler recycles), to extract the Treset for a given time period, and add these into the History. It should be noted that this will require the Treset History pipeline to be run over the time period covering an observation before the normal pipeline processing can be run.

Input: Housekeeping data frames

DPU counter reset time	Format	CUC format time in units of 1/65536 seconds. since 1st January 1958 as Long integer
	Units	1/65536 seconds

Calibration files:
None but DPU Counter Reset History Table may be updated

Output:

DPU Counter Reset History (SCalresetHist)	Format	CUC format time in units of 1/65536 seconds. since 1st January 1958 as Long integer
	Units	1/65536 seconds

6.2 Offset History Processing Pipeline

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 the telemetry on request. There is no guarantee that the offsets are downlinked during an observation, although it is planned. Figure 15 shows the Offset History Processing Pipeline.

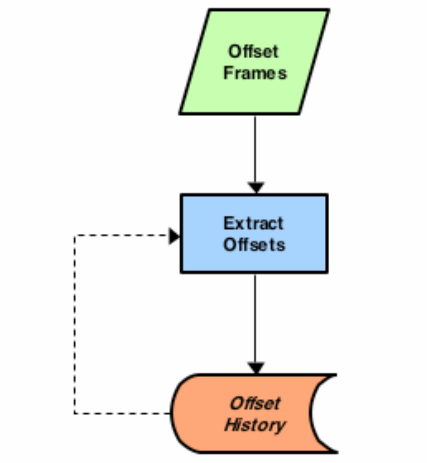

Figure 15: Offset History Processing Pipeline

6.2.1 Extract Offset Process

Description:
This process is run as a step in the ODP and is responsible to extract the offset raw values creating for each Operational Day a single detector offset history for both the photometer and for the spectrometer which will be made available to the standard pipeline. This pipeline will ultimately be used to create a history of offset values for the whole mission, Every time a new signal offset is set, a new row is added with the time.

Input:

Offset frames	Format	Channel offsets as longer integer one for each channel
	Units	ADU

Calibration files:
DPU reset history
Channel Offset History Table may be updated

DPU Counter Reset History (SCalresetHist)	Format	CUC format time in units of 1/65536 seconds. since 1st January 1958 as Long integer
	Units	1/65536 seconds

Output:

Channel Offset History Table (ScalPhotOffsetHist, SCalSpecOffsetHist)	Description	Sample Time
	Format	64 bit integer fine time since Jan 1st 1958
	Units	microseconds
	Description	Signal Offsets
	Format	Channel offsets as longer integer one for each channel
	Units	ADU

7. CALIBRATION PIPELINES

TBW