SUBJECT: SPIRE Data Products **Specification**

PREPARED BY: K.J. King

DOCUMENT No: SPIRE-RAL-DOC-002005

ISSUE: Issue 0.2 **Date:** 7th September 2004

APPROVED BY: **Date:**

Distribution

Change Record

ISSUE	DATE	Changes
Draft 1	6th May 2004	Initial draft containing detector timeline definitions
Issue 0.2	7th September 2004	Updated to include SDT, SMECT and SDS product definitions

TABLE OF CONTENTS

TABLES

Glossary

1. INTRODUCTION

This note specifies the content and structure of the SPIRE data products. These are produced as calibration products created from information provided by instrument manufacturers, calibration products produced from calibration tests and observations after processing through IA and scientific data products generated by the datat processing steps of IA and the SPIRE pipelines.

1.1 Scope

This note is intended to provide the complete definition of the format and contents of all SPIRE data products. The processing that goes into creating the content of the products is described elsewhere (in the individual data processing specification for each processing step - for use by software developers, and in the SPIRE Data Processing Description Document – for use by users of the data products).

1.2 Structure of Document

Section 2 describes the scientific data products produced at various stages of the data processing pipelines.
Section 3 describes the calibration data products used by the scientific data processing steps

1.3 Documents

1.3.1 Applicable Documents

1.3.2 Reference Documents

RD01 Engineering Data Processing Specification

2. SCIENTIFIC DATA PRODUCTS

2.1 Common Data Products

2.1.1 Attitude Timeline

TBW

2.1.2 Beam Steering Mirror Timeline

TBW

2.1.3 Spectrometer Mechanism Timeline (SMECT)

This spectrometer product is presented as a generic data product. It contains data samples from the spectrometer stage mechanism including the times at which the stage was sampled. A single timeline will contain the data from at most one observation. Normally each building block in an observation will only generate one type of SMEC science data frame but in the event that more than one type is generated multiple data products will be produced.

The data are extracted directly from a single type of science data frame from a single BB generated by the instrument MCU and subsequently stored in the HCSS. The data may have been processed. It is expected that MCU science frames with an invalid checksum, or FrameID will be rejected during the creation of the Data Frames and therefore the SMEC data product will not contain these.

This data product is produced when SMEC data is generated with the instrument in a spectrometer configuration.

2.1.3.1 SMECT Product Format

product (type=”SMECT”, description=” Spectrometer Mechanism Timeline”)			
Metadata:			
string	Creator	(description=”Creator”, quantity=””)	
date	creationDate	(description=”Creation Date”, quantity=”UTC”)	
string	instrument	(description=”Instrument”, quantity=””)	
string	modelName	(description=”Instrument Model Name”, quantity=””)	
date	StartDate	(description=”Start Date”, quantity=”UTC”)	
date	EndDate	(description=”End Date”, quantity=”UTC”)	
long	Obsid	(description=”Observation Identifier”, quantity=””)	
long	Bbtype	(description=”Building Block type”, quantity=””)	
long	Bbcount	(description=”Building Block execution counter”, quantity=””)	
string	AOT	(description=”AOT Identifier”, quantity=””)	
composite dataset	(description=”Mechanism Timeline”)		
table dataset	(description=”Mechanism Scan Timeline”)		
metadata:			
long	SCANNR	(description=”Current Scan Number”, quantity=””)	
boolean	SCANDIRECTION	**(DESCRIPTION=”SCAN DIRECTION”, QUANTITY=””)**	
integer-1d[]	FrameTime	**(DESCRIPTION=”SMEC FRAME TIME”, QUANTITY=”1/312500 SECS”)**	
double-1d[]	SMECSEL_ENCPOSN	signal (description=”SMEC Scan Optical Encoder Position timeline”, quantity=”μm”)	
double-1d[]	SMECSEL_ENCFINEPOSN	signal (description=”SMEC Scan Optical Encoder Fine Position timeline”, quantity=”nm”)	
double-1d[]	SMECLVDTPOSN	(description=”LVDT Position”, quantity=”μm”)	
integer-1d[]	Mask	(description=”Data Mask”, quantity=””)	
…..			
history			
…..			

2.1.3.2 SMECT Product Contents

Data Item	Description
Creator	Name of the module which created the product. This should indicate the version also. Is set by the creator task, e.g. to "EngineeringDataModule Version_n.m"
creationDate	UTC of the date and time of creation of the product. Is set by the creator task.
instrument	Set to "SPIRE" Can be derived from the first two bits of OBSID.
modelName	Name of the instrument model from which the data came e.g. "CQM", "PFM1", "FM", "FS" Can be derived from the first four bits of the BBID.
startDate	UTC of the date and time of the start of the observation to which this product relates. Can be derived from the contents of this product.
endDate	UTC of the date and time of the end of the observation to which this product relates. Can be derived from the contents of this product.
Obsid	The OBSID parameter from the first Normal Housekeeping TM packet with a time belonging to this building block Since OBSID serves as the overall index of all observations I would assume that it is specified by the requestor of a product.
Bbtype	The BBID's type parameter (14 bits) from the first Normal Housekeeping TM packet with a time belonging to this building block.
Bbcount	The BBID execution count parameter (16 bits) from the first Normal Housekeeping TM packet with a time belonging to this building block.
AOT	SOFn with n=1..4, depending on the operating mode used. SOF0 can be used when no template is used, e.g. in the case of a non-standard observation. (TBC) How could this be derived?
SCANNR	Number of the current scan. One scan is defined as data collected from initial to final position at stabilized speed in the same direction. Can be derived from the contents of this product if information is not available in hk packets (TBC).
SCANDIRECTION	Direction of the current scan. Can be derived as follows: 0 or false if scanned<scanstart else 1 or true.
FrameTime	Frame time in clock ticks since the last clock reset. The clock runs at a rate of 312500Hz. Pulses get counted by 32 bit counters. Integer is a signed 32 bit variable. We should check that the 32-bit signed format will convert properly into the unsigned format, i.e. negative values must turn into high positive clock counts. Taken from the science report.
SMECSEL_ENCPOSN	The coarse position given by the optical encoder at the respective time (resolution of the encoder position is 1µm). Taken from the science report.
SMECSEL_ENCFINEPOSN	The fine position given by the optical encoder at the respective time (resolution of the encoder position is 1nm with a maximum distance of 65µm). Taken from the science report.
SMECLVDTPOSN	The position given by the LVDT at the respective time (resolution of the encoder position is 0.3µm). TBC how information is calculated from SMECLVDTDCSIG.
Mask	Bit mask indicating whether the value failed to meet certain criteria (first bit = 1 indicates failure):

2.1.4 SCU Timeline

TBW

2.1.5 Spacecraft Housekeeping Timeline

TBW

2.1.6 Events Timeline

TBW

2.1.7 Orbit Timeline

TBW

2.2 Photometer Specific Products

2.2.1 Photometer Detector Timeline (PDT)

This product contains data samples from a set of detector channels (containing data from detector pixels, temperature sensors and fixed reference resistors) along with a time at which the channels were sampled. A single timeline will contain the data from at most one observation building block. Normally each building block in an observation will only generate one type of detector science data frame but in the event that more than one type is generated multiple data products will be produced.

The data is extracted directly from a single type of science data frame generated by the instrument DCU and subsequently stored in the HCSS. The data may have been processed. It is expected that DCU science frames with an invalid checksum, or FrameID will be rejected during the creation of the Data Frames and therefore the detector data product will not contain these.

This product is produced when detector data is generated with the instrument in a photometer configuration. In these configurations several different science data frames can be generated

The different photometer science data frame types are:

Frame Type	Number of Channels	DCU Frame ID (hex)	TM Packet APID (hex)	TM Packet SID (hex)	Subsystem	ArrayName
Photometer Full Array	288	00	3	0200	"PHOTOMETER"	"PSW" "PMW" "PLW"
PSW Array	144	02	3	0102	"PHOTOMETER"	"PSW"
PMW Array	96	03	3	0103	"PHOTOMETER"	"PMW"
PLW Array	48	04	3	0104	"PHOTOMETER"	"PLW"
Photometer Test Pattern	288	09	3	0309	"PHOTOMETER"	"PSW" "PMW" "PLW"
PSW Test Pattern	144	0A	3	030A	"PHOTOMETER"	"PSW"
PMW Test Pattern	96	0B	3	030B	"PHOTOMETER"	"PMW"
PLW Test Pattern	48	0C	3	030C	"PHOTOMETER"	"PLW"
Parallel Mode Data	TBD		5	0F01	"PHOTOMETER"	TBD
Transparent Data	TBD		5	FF00	TBD	TBD

Table 2-1

2.2.1.1 PDT Product Format

product (type=”PDT”, description=” Photometer Detector Timeline”)		
metadata:		
string	creator	(description=”Creator”, quantity=””)
date	creationDate	(description=”Creation Date”, quantity=”UTC”)
string	instrument	(description=”Instrument”, quantity=””)
string	modelName	(description=”Instrument Model Name”, quantity=””)
date	startDate	(description=”Start Date”, quantity=”UTC”)
date	endDate	(description=”End Date”, quantity=”UTC”)
long	obsid	(description=”Observation Identifier”, quantity=””)
long	bbid	(description=”Building Block Identifier”, quantity=””)
long	bbtype	(description=”Building Block Type”, quantity=””)
composite dataset	(description=”Detector Timeline”)	
metadata:		
string	subsystem	(description=”Instrument Subsystem”, quantity=””)
composite dataset	(description=”Array Timeline”)	
metadata:		
string	arrayName	(description=”Detector Array Name”, quantity=””)
string	modelName	(description=”Model Name”, quantity=””)
table dataset	(description=”Pixel Timeline”)	
metadata:		
string	pixelName	(description=” Pixel Name”, quantity=””)
string	source	(description=”Source packet”, quantity=””)
integer	pixelId	(description=”Pixel Identifier”, quantity=””)
double-1d[]	obt	(description=”On Board Time”, quantity=”Secs”)
double -1d []	signal	signal (description=”Detector Signal”, quantity=”Volts”)
double -1d []	error	(description=”Error on signal”, quantity=”Volts”)
integer-1d[]	mask	(description=”Data Mask”, quantity=””)
table dataset	(description=”Pixel Timeline”)	
…….		
table dataset	(description=”Pixel Timeline”)	
…….		
composite dataset	(description=”Array Timeline”)	
…..		
history		
…..		

2.2.1.2 PDT Product Contents

Data Item	Description
creator	Name of the module which created the product. This should indicate the version also. Normally is expected to be "EngineeringData Version_n.m"
creationDate	UTC of the date and time of creation of the product
instrument	Set to "SPIRE"
modelName	Name of the instrument model from which the data came e.g. "CQM", "PFM1", "FM", "FS"
startDate	UTC of the date and time of the start of the Building Block to which this product relates
endDate	UTC of the date and time of the end of the Building Block to which this product relates
obsid	The OBSID parameter from the first Normal Housekeeping TM packet with a time belonging to this building block
bbid	The BBID parameter from the first Normal Housekeeping TM packet with a time belonging to this building block
bbtype	The BBTYPE parameter from the first Normal Housekeeping TM packet with a time belonging to this building block
subsystem	Name of the subsystem generating the data As defined in Table 2-1
arrayName	Name of the detector array to which the data relates As defined in Table 2-1
modelName	Name of the model of the array from which the data came e.g. "CQM", "PFM1", "FM", "FS"
pixelName	Unique Name for the pixel For example pixel C5 on the PLW array = "PLWC5" This name identifies the physical pixel from which the data has come irrespective of the type of data frame in which the data was transmitted
pixelSource	Identifies the source packet from which the pixel data has been extracted. For example "PHOTLW"
pixelID	Identifies the channel in the source packet from which the data comes. For example pixel "PLWC5" may be found in packet "PHOTLW" at location 035 Note: Channel numbers start from 001
obt	The on board time of each sample of the pixel. The value is represented as a Double representing the seconds since a fixed date (nominally Jan 1st 1958)
signal	The signal from the detector. This may be represented in raw or engineering units
error	Error in the measured signal value.
mask	Bit mask indicating whether the pixel failed to meet certain criteria (bit = 1 indicates failure): **Bit 0 (lsb)** the master bit, is set whenever any of the other bits is set to allow easy detection of a bad sample **Bit 1** when set indicates invalid data sample time (see Engineering Data Specification) **Bit 2** when set indicates a possible ADC latch up error occurred for this sample (see Engineering Data Specification) Other bits are TBS

2.2.2 Demodulated Detector Timeline

TBW

2.2.3 Time Averaged Detector Timeline

TBW

2.2.4 Pointed Photometer Product

TBW

2.3 Spectrometer Specific Products

2.3.1 Spectrometer Detector Timeline (SDT)

This product contains data samples from a set of detector channels (containing data from detector pixels, temperature sensors and fixed reference resistors) along with a time at which the channels were sampled. A single timeline will contain the data from at most one observation building block. Normally each building block in an observation will only generate one type of detector science data frame but in the event that more than one type is generated multiple data products will be produced.

The data is extracted directly from a single type of science data frame generated by the instrument DCU and subsequently stored in the HCSS. The data may have been processed. It is expected that DCU science frames with an invalid checksum, or FrameID will be rejected during the creation of the Data Frames and therefore the detector data product will not contain these.

This product is produced when detector data is generated with the instrument in a spectrometer configuration. In these configurations several different science data frames can be generated

The different spectrometer science data frame types are:

Frame Type	Number of Channels	DCU Frame ID (hex)	TM Packet APID (hex)	TM Packet SID (hex)	Subsystem	ArrayName
Spectrometer Full Array	72	01	4	0201	"SPECTROMETER"	"SSW" "SLW"
SSW Array	48	05	4	0105	"SPECTROMETER"	"SSW"
SLW Array	24	06	4	0106	"SPECTROMETER"	"SLW"
Spectrometer Test Pattern	72	0D	4	030D	"SPECTROMETER"	"SSW" "SLW"
SSW Test Pattern	48	0E	4	030E	"SPECTROMETER"	"SSW"
SLW Test Pattern	24	0F	4	030F	"SPECTROMETER"	"SLW"
Transparent Data	TBD		5	FF00	TBD	TBD

Table 2-2

2.3.1.1 SDT Product Format

product (type=”SDT”, description=” Spectrometer Detector Timeline”)		
metadata:		
string	Creator	(description=”Creator”, quantity=””)
date	creationDate	(description=”Creation Date”, quantity=”UTC”)
string	instrument	(description=”Instrument”, quantity=””)
string	modelName	(description=”Instrument Model Name”, quantity=””)
date	StartDate	(description=”Start Date”, quantity=”UTC”)
date	EndDate	(description=”End Date”, quantity=”UTC”)
long	Obsid	(description=”Observation Identifier”, quantity=””)
long	Bbtype	**(DESCRIPTION=”BUILDING BLOCK TYPE”, QUANTITY=””)**
long	Bbcount	(description=”Building Block execution counter”, quantity=””)
string	AOT	(description=”AOT Identifier”, quantity=””)
string	Subsystem	(description=”Instrument Subsystem”, quantity=””)
composite dataset	(description=”Array Timeline”)	
Metadata:		
string	arrayName	(description=”Detector Array Name”, quantity=””)
table dataset	(description=”Pixel Timeline”)	
metadata:		
string	pixelName	(description=”Pixel Name”, quantity=””)
string	pixelSource	(description=”Source packet”, quantity=””)
integer	pixelId	(description=”Pixel Identifier”, quantity=””)
integer-1d[]	FrameTime	(description=”Detector frame time”, quantity=”1/312500 s”)
double–1d []	signal	signal (description=”Detector Signal”, quantity=”Volts”)
double–1d []	error	(description=”Error on signal”, quantity=”Volts”)
integer-1d[]	mask	(description=”Data Mask”, quantity=””)
table dataset	(description=”Pixel Timeline”)	
…….		
table dataset	(description=”Pixel Timeline”)	
…….		
history		
…..		

2.3.1.2 SDT Product Contents

Data Item	Description
Creator	Name of the module which created the product. This should indicate the version also. Is set by the creator task, e.g. to "EngineeringDataModule Version_n.m"
CreationDate	UTC of the date and time of creation of the product Is set by the creator task.
Instrument	Set to "SPIRE" Can be derived from the first two bits of OBSID.
ModelName	Name of the instrument model from which the data came e.g. "CQM", "PFM1", "FM", "FS" Can be derived from the first four bits of the BBID.
StartDate	UTC of the date and time of the start of the Building Block to which this product relates. Can be derived from the contents of this product.
EndDate	UTC of the date and time of the end of the Building Block to which this product relates. Can be derived from the contents of this product.
Obsid	The OBSID parameter from the first Normal Housekeeping TM packet with a time belonging to this building block Since OBSID serves as the overall index of all observations I would assume that it is specified by the requestor of a product.
Bbtype	The BBID's type parameter (14 bits) from the first Normal Housekeeping TM packet with a time belonging to this building block.
Bbcount	The BBID execution count parameter (16 bits) from the first Normal Housekeeping TM packet with a time belonging to this building block.
AOT	SOFn with n=1..4, depending on the operating mode used. SOF0 can be used when no template is used, e.g. in the case of a non-standard observation. (TBC) How could this be derived?
Subsystem	Name of the subsystem generating the data As defined in Table 2-2: "Spectrometer" Can be derived from the APID in the science telemetry.
ArrayName	Name of the detector array to which the data relates As defined in Table 2-2, either "SLW", "SSW", or "Full" Can be derived from the APID in the science reports.
pixelName	Unique Name for the pixel For example pixel C5 on the PLW array = "PLWC5" This name identifies the physical pixel from which the data has come irrespective of the type of data frame in which the data was transmitted Can be taken from the science reports.
pixelSource	Identifies the source packet from which the pixel data has been extracted. "SPECSW", "SPECLW", or "SPECF" Can be derived from the APID in the science reports.
pixelId	Identifies the channel in the source packet from which the data comes. For example pixel "SLWC5" may be found in packet "SPECLW" at location 021. Note: Channel numbers start from 001 Can be taken from the science reports.
FrameTime	The number of clock ticks since the last clock reset. The clock runs at a rate of 312500Hz. Pulses get counted by 32 bit counters. Integer is a signed 32 bit variable. We should check that the 32-bit signed format will convert properly into the unsigned format, i.e. negative values must turn into high positive clock counts. Can be taken from the science reports.
Signal	The signal from the detector which has been converted into Volts, taking into account biases and gains from the instrument electronics. Can be taken from the science reports and hk data on the electronics.
Error	Error in the measured signal value.

Data Item	Description
Mask	Bit mask indicating whether the pixel failed to meet certain criteria (bit = 1 indicates failure): **Bit 0 (lsb)** the master bit, is set whenever any of the other bits is set to allow easy detection of a bad sample **Bit 1** when set indicates invalid data sample time (see Engineering Data Specification) **Bit 2** when set indicates a possible ADC latch up error occurred for this sample (see Engineering Data Specification) Other bits are TBS

2.3.2 Spectrometer Detector Spectrum Product (SDS)

This product contains spectral data for a set of detector channels that contain data from detector pixels along with a wavenumber array. A single product will contain all spectra taken within one observation/building block, i.e. it will contain spectra from one or more scans or from an observation in step & integrate mode. Normally each building block in an observation will only generate one type of detector science data frame but in the event that more than one type is generated multiple data products will be produced.

The data has been processed from spectrometer science data frames (SDT and SMECT) or it can be extracted from the HCSS database.

This product is the output of the data processing step Fourier Transformation that operates on spectrometer data SDT and SMECT.

This product may be used by secondary software tools for scientific analysis. The different spectrometer science data frame types are:

2.3.2.1.1 Frame Type	Number of Channels	DCU Frame ID (hex)	TM Packet APID (hex)	TM Packet SID (hex)	Subsystem	Array Name
Spectrometer Full Array	72	01	4	0201	"SPECTROMETER"	"SSW" "SLW"
SSW Array	48	05	4	0105	"SPECTROMETER"	"SSW"
SLW Array	24	06	4	0106	"SPECTROMETER"	"SLW"
Spectrometer Test Pattern	72	0D	4	030D	"SPECTROMETER"	"SSW" "SLW"
SSW Test Pattern	48	0E	4	030E	"SPECTROMETER"	"SSW"
SLW Test Pattern	24	0F	4	030F	"SPECTROMETER"	"SLW"
Transparent Data	TBD		5	FF00	TBD	TBD

Table 2-3

2.3.2.2 SDS Product Format

product (type="SDS", description=" Spectrometer Detector Spectrum")		
Metadata:		
String	Creator	(description="Creator", quantity="")
Date	creationDate	(description="Creation Date", quantity="UTC")
String	instrument	(description="Instrument", quantity="")
String	modelName	(description="Instrument Model Name", quantity="")
date	StartDate	(description="Start Date", quantity="UTC")
date	EndDate	(description="End Date", quantity="UTC")
long	Obsid	(description="Observation Identifier", quantity="")
long	Bbtype	(description="Building Block type", quantity="")
long	Bbcount	(description="Building Block execution counter", quantity="")
integer	numSpectra	(description="Number of spectra", quantity="")
string	Subsystem	(description="Instrument Subsystem", quantity="")
composite dataset	(description="Detector Spectrum")	
metadata:		
String	ArrayName	(description="Detector Array Name", quantity="")
table dataset	(description="Pixel Spectrum")	
metadata:		
string	PixelName	(description=" Pixel Name", quantity="")
string	pixelSource	(description="Source packet", quantity="")
integer	PixelId	(description="Pixel Identifier", quantity="")
double	LowEdge	(description="Lower edge of band", quantity="cm-1")
double	HighEdge	(description="Higher edge of band", quantity="cm-1")
double	Resolution	(description="Resolution element", quantity="cm-1")
complex –1d []	flux	(description="Intensity", quantity="mJy")
double –1d []	frequency	(description="Wavenumber Array", quantity="cm-1")
complex –1d []	weight	(description="Error on intensity", quantity="mJy")
long-1d []	specId	(description="Spectrum Identifier", quantity="")
integer-1d[]	flag	(description="Data Mask", quantity="")
table dataset	(description="Pixel Spectrum")	
…….		
table dataset	(description="Pixel Spectrum")	
…….		
history		
…..	Must identify all measurements/processing steps that went into the calculation of the SDS and include the complete histories of these data sets.	

2.3.2.3 SDS Product Contents

Data Item	Description
Creator	Name of the module which created the product. This should indicate the version also. Is set by the creator task, e.g. to "EngineeringDataModule Version_n.m"
CreationDate	UTC of the date and time of creation of the product Is set by the creator task.
Instrument	Set to "SPIRE" Can be derived from the first two bits of OBSID.
ModelName	Name of the instrument model from which the data came e.g. "CQM", "PFM1", "FM", "FS" Can be derived from the first four bits of the BBID.
StartDate	UTC of the date and time of the start of the Building Block to which this product relates. Is set by the constructor of this product.
EndDate	UTC of the date and time of the end of the Building Block to which this product relates. Is set by the constructor of this product.
Obsid	The OBSID parameter from the first Normal Housekeeping TM packet with a time belonging to this building block Since OBSID serves as the overall index of all observations I would assume that it is specified by the requestor of a product.
Bbtype	The BBID's type parameter (14 bits) from the first Normal Housekeeping TM packet with a time belonging to this building block.
Bbcount	The BBID execution count parameter (16 bits) from the first Normal Housekeeping TM packet with a time belonging to this building block.
numSpectra	Number of spectra in this product. Is equal to the number of scans.
Subsystem	Name of the subsystem generating the data As defined in Table 2-2: "Spectrometer" Can be derived from the APID in the science reports.
ArrayName	Name of the detector array to which the data relates As defined in Table 2-2: Either "SSW", "SLW", or "Full"
PixelName	Unique Name for the pixel For example pixel C5 on the PLW array = "PLWC5" This name identifies the physical pixel from which the data has come irrespective of the type of data frame in which the data was transmitted Can be taken from the science reports.
pixelSource	Identifies the source packet from which the pixel data has been extracted. "SPECSW", "SPECLW", or "SPECF" Can be derived from the APID in the science reports.
PixelID	Identifies the channel in the source packet from which the data comes. For example pixel "SLWC5" may be found in packet "SPECLW" at location 021. Note: Channel numbers start from 001 Can be taken from the science reports.
lowEdge	The smallest element in the wavenumber array. Can be taken from the product contents.
highEdge	The largest element in the wavenumber array. Can be taken from the product contents.
resolution	The resolution element $\Delta\sigma$ of the wavenumber array. Can be taken from the product contents.
flux	The complex spectrum as a function of wavenumber
frequency	The uniformly sampled wavenumber array. It is still TBD how the three instruments on Herschel will deal with different x-axes such as wavelength, frequency, and wavenumber.
weight	Error in the spectrum as a weight which is proportional to 1 / (error ^2)
specId	Counter for the spectra in the dataset.

Data Item	Description
flag	Bit mask indicating whether the pixel failed to meet certain criteria (bit = 1 indicates failure): **Bit 0 (lsb)** the master bit, is set whenever any of the other bits is set to allow easy detection of a bad sample **Bit 1** when set indicates invalid data sample time (see Engineering Data Specification) **Bit 2** when set indicates a possible ADC latch up error occurred for this sample (see Engineering Data Specification) Other bits are TBS I assume this is the quality control bitmask and no other quality control feature will have to be handled.

3. CALIBRATION DATA PRODUCTS

TBW