





## **Change Record**

**ISSUE**

**DATE**

## **Table of Contents**

<b>1.</b>	<b>INTRODUCTION .....</b>	<b>6</b>
<b>1.</b>	<b>INTRODUCTION .....</b>	<b>6</b>
1.1	SCOPE .....	6
1.2	ASSUMPTIONS.....	6
<b>2.</b>	<b>DOCUMENTS.....</b>	<b>7</b>
2.1	APPLICABLE DOCUMENTS .....	7
2.2	REFERENCE DOCUMENTS .....	7
<b>3.</b>	<b>STRUCTURE OF THE PARAMETER LIST FILE.....</b>	<b>8</b>
3.1	PARAMETER LIST FILE NAME .....	8
3.2	PARAMETER LIST HEADER.....	8
3.3	PARAMETER LIST RECORD STRUCTURE.....	8
<b>4.</b>	<b>SID TABLE FILE.....</b>	<b>11</b>
4.1	SID TABLE FILE NAME .....	11
4.2	SID TABLE HEADER.....	11
4.3	SID TABLE RECORD STRUCTURE.....	11
<b>5.</b>	<b>ANALOGUE PARAMETER TABLES.....</b>	<b>12</b>
5.1	ANALOGUE PARAMETER TABLE STRUCTURE .....	12
<b>6.</b>	<b>ENUMERATED PARAMETER TABLES .....</b>	<b>13</b>
6.1	ENUMERATED PARAMETER TABLE STRUCTURE.....	14
<b>7.</b>	<b>DERIVED PARAMETER TABLES.....</b>	<b>15</b>
<b>8.</b>	<b>OUT OF LIMITS (OOL) PARAMETER TABLES.....</b>	<b>15</b>
8.1	OOL PARAMETER TABLE STRUCTURE .....	15

## **Glossary**

AOT	Astronomical Observation Template
APID	Application Process Identifier
BSM	Beam Steering Mirror
DCU	Detector Control Unit
DPU	Digital Processing Unit
DRCU	Detector Readout and Control Unit
FCU	Focal Control Unit
FTS	Fourier Transform Spectrometer
HK	Housekeeping
ICD	Interface Control Document
IF	Instrument Function
MCU	Mechanism Control Unit
MIB	Mission Information dataBase
OBS	On-Board Software
OOL	Out Of Limits
PCAL	Internal calibrator for the Photometer
POF	Photometer Observatory Function
QLA	Quick Look Analysis
S/C	Spacecraft
SCAL	Internal calibrator for the FTS
SCU	Sub-system Control Unit
SID	Structure Identifier
SMEC	Spectrometer Mechanism
SOF	Spectrometer Observatory Function
SPIRE	Spectral and Photometric Imaging Receiver
TC	TeleCommand
TM	TeleMetry

# **1. INTRODUCTION**

## **1.1 Scope**

The SPIRE Data ICD [AD-1] defines the structure and contents of SPIRE TM and TC packets in accordance with the Packet Structure ICD [RD-2]. This document describes the structure and contents of the SPIRE TM parameter list, which includes definitions of both science and HK parameters and gives information on how to extract them from TM packets. This list is intended for use in the development of the QLA prototype. For later versions of QLA some of the information present in the list should eventually be retrieved (TBC) from the MIB. The MIB is essentially a repository of all HK parameters (as well as TC definitions) but we could also include the definitions of science parameters (TBC).

## **1.2 Assumptions**

The parameter list file is in ASCII format for transparency and ease of update and use. The file contains a number of records with a fixed format and length. Each field will be left justified and blank padded. Each record in the file ends with a new line character. Field delimiters will not be present.

## 2. DOCUMENTS

### 2.1 Applicable Documents

<b>AD-1</b>	SPIRE Data ICD	SPIRE-RAL-DOC-001078	1.0 (Draft 1)	15 <sup>th</sup> Jan 2002
-------------	----------------	----------------------	------------------	---------------------------

### 2.2 Reference Documents

<b>RD-1</b>	SPIRE Design Description	SPIRE-RAL-PRJ-000620	1.0	4 <sup>th</sup> February 2002
<b>RD-2</b>	Herschel/Planck Packet Structure Interface Control Document	SCI-PT-ICD-07527	2/0	12 <sup>th</sup> July 2001

### 3. STRUCTURE OF THE PARAMETER LIST FILE

#### 3.1 Parameter List File Name

The parameter list file name will be SPIRE\_Param\_DB.txt.

#### 3.2 Parameter List Header

The header of the parameter list file will consist of a number of lines, each line beginning with a # character in column one. These header lines will include the actual name of the parameter list file, author, version, date and purpose. Subsequent lines will include any appropriate history information, as well as the headings for each column in the parameter list record. For information the column headings will also be defined between the "Start\_Columns" and "End\_Columns" header lines. An example header is shown below:

```
#
# filename:      SPIRE_Param_DB.txt
# author:       Sunil D Sidher S.D.Sidher@rl.ac.uk
# version:      1.2
# date:         14th March 2002 prototype
# purpose:      Parameter List for SPIRE QLA
#
#Start_Columns
#Parameter Name
#MIB Name
#Packet Type
#Packet Sub-Type
#APID (Hex)
#Loc (bits)
#Length (bits)
#Packet/Frame/Header Flag
#SID (Hex)
#Parameter Type (Analogue, Derived, Enumerated)
#Conversion Table
#OOL Table
#Parameter Description
#End_Columns
#
#
```

#### 3.3 Parameter List Record Structure

Following the parameter list header there are no blank lines before the start of the parameter list records. Data columns in the parameter list record are separated by white space. Tab separators are forbidden to ensure that the columns in the data record are always aligned, irrespective of the editor or display tool for the parameter list file.

The parameter description will always be in the last column of the list record. This is because it is the only field with embedded white space and consequently needs to be read differently from other parameter fields.

**Table 1: SPIRE Parameter List Data Record Structure**



Column	Content	Type	Description
1	Parameter Name	String	Name of parameter in the parameter list file
2	MIB Name	String	Name of parameter in the MIB (begins with S for SPIRE). Consists of 8 characters
3	Packet Type	Integer	Type of Packet (as specified in the PS-ICD)
4	Packet Sub-Type	Integer	Sub-Type of Packet (as specified in the PS-ICD)
5	APID	Hexadecimal	Application Process ID
6	Location	Integer	Location of parameter in the Packet or Frame (bits). The maximum TM Packet Length is 1024 octets, i.e. 8192 bits.
7	Length	Integer	Number of bits holding the parameter value in the Packet or Frame
8	Data Locator	String	<p>Specifies how the parameter value is to be located.</p> <p>The data locator can take one of three values: P, F or H</p> <p>1) If the locator is set to P (Packet) then column 6 indicates that the bit offset (viz. Location - Column 6) of the parameter is from the <i>start of the Packet</i>.</p> <p>2) If the locator is set to F (Frame) then the bit offset for the parameter is with respect to the <i>start of the Frame</i>. The frame length for each SID is given in the SID table file. The number of frames in the data field of a packet can be calculated from the Packet Length stored in the source packet header (Location 32, Length 16 bits). The Packet Length is the length of the data field in a packet, including the 80 bits long data field header and the 16 bits long Checksum.</p> <p>The number of frames in a science packet is given by:</p> $N_f = [(P_1 + 1) * 8 - D_h - C_s - N_{SID} - N_{OBSID} - N_{BBID}] / (F_1 * 8)$ <p>Where <math>P_1</math> is the Packet Length in octets  <math>D_h</math> is the length of the data field header in bits (80)  <math>C_s</math> is the length of the Checksum field in bits (16)  <math>F_1</math> is the Frame Length in octets (Given in the SID table)  <math>N_{SID}</math> is the length of the SID field in bits (16)  <math>N_{OBSID}</math> is the length of the OBSID field in bits (32)  <math>N_{BBID}</math> is the length of the BBID field in bits (32)</p> <p>If there are multiple frames in a source packet then a vector of parameter values should be returned by the parameter extraction software.</p> <p>The bit offset of a parameter in a science packet with respect to the start of a packet can be determined from:</p> $O_p = P_h + D_h + N_{SID} + N_{OBSID} + N_{BBID} + F_1 * F_1 * 8$ <p>Where <math>P_h</math> is the length of the source packet header (48 bits)  <math>F_1</math> is the frame number in the science packet (starting from 0 and going up to <math>N_f - 1</math>)</p> <p>3) If the locator is set to H (Header) then the parameter is to be extracted from the data field header. The bit offset is with respect to the <i>start of the packet</i>, as in 1) above.</p>

9	SID	Hexadecimal	Structure Identifier for Packet
10	Parameter Type	String	Type of parameter: A - Analogue D - Derived E - Enumerated (For example Low, Medium, High - includes Status parameters)
11	Conversion Table	String	<p>Name of table for converting the raw parameter value. There will be three types of conversion tables: Analogue, Enumerated and Derived. The name of the table can be deduced in the following ways:</p> <p>1) If this field contains the single character "Y" (double quote delimiters are not part of the field) then it denotes that an ascii conversion table exists for the parameter. The table name will have the syntax :</p> <p>&lt;Parameter Name&gt;.ATAB for analogue parameter &lt;Parameter Name&gt;.ETAB for enumerated parameter &lt;Parameter Name&gt;.DTAB for derived parameter</p> <p>where &lt;Parameter Name&gt; is the name of the parameter as given in Column 1 of the Parameter List. e.g. PSWJFET5V.ATAB     BIASTEMP.ATAB     SCANMODE.ETAB     CHOPLOOPMODE.ETAB</p> <p>2) If this field contains the single character "N" (double quote delimiters are not part of the field) then it denotes that no conversion table exists for the parameter.</p> <p>3) If this field contains any other string then it denotes the full name of the ascii table for converting the parameter value. The type of parameter will be deduced from the table extension name. e.g. PMWJFETV.ATAB     SMECLATSTAT.ETAB</p>
12	OOL Table	String	<p>Name of table for finding the soft and hard limits for the parameter.</p> <p>1) If this field contains the single character "Y" (double quote delimiters are not part of the field) then the OOL table will have the name &lt;Parameter Name&gt;.OTAB</p> <p>e.g. PMWPHSE.OTAB     SSWBIAS.OTAB</p> <p>2) If this field contains the single character "N" (double quote delimiters are not part of the field) then there will be no OOL table for the parameter.</p> <p>3) If this field contains any other string then it denotes the full name of the ascii OOL table. e.g. PMWJFETV.OTAB     SMECLATSTAT.OTAB</p>
13	Parameter Description	String	Description of the parameter

## 4. SID TABLE FILE

In science TM packets the Structure Identifier (SID) can be used to determine the length of a science frame. Quite often there will be more than one frame in a science packet. The SID table will list the frame lengths for all SIDs for all types of science packets. At present the SID table is only being used for science packets but it should be possible (TBC) to extend its scope to other types of packets (e.g. HK, Diagnostic, etc).

### 4.1 SID Table File Name

The SID table file name will be SPIRE\_SID\_Table.txt.

### 4.2 SID Table Header

The header of the SID table file will consist of a number of lines, with each line beginning with a # character in column one. These header lines will include the name of the parameter list file itself, author, version, date and purpose. Subsequent lines will include any appropriate history information, as well as the headings for each column in the parameter list record. For information the column headings will also be defined between the "Start\_Columns" and "End\_Columns" header lines. An example header is shown below:

```
# filename:      SPIRE_SID_Table.txt
# author:       Sunil D Sidher S.D.Sidher@rl.ac.uk
# version:      1.4
# date:         March 14th 2002
# purpose:      SID Table for SPIRE QLA
#
#Start_Columns
#SID (Hex)
#Frame length (Octets)
#End_Columns
#
```

### 4.3 SID Table Record Structure

Following the SID table header there are no blank lines before the start of the SID table records. Data columns in the SID table record are separated by white space. Tab separators are forbidden to ensure that the columns in the data record are always aligned, irrespective of the editor or display tool for the SID table. The contents of the SID table data record are defined in Table 2.

**Table 2: SPIRE SID Table Data Record Structure**

Column	Content	Description
1	SID	Structure Identifier of the Science Packet (Hex)
2	Frame Length	Length of Science Frame (Octets)

An example SID table for SPIRE is shown in Table 3.

**Table 3: SPIRE SID Table**

Structure Identifier (SID) / Hex	Frame Length / octets
0103	200
0200	584
...	...
...	...

## 5. ANALOGUE PARAMETER TABLES

For parameters possessing unique lookup tables between raw and converted analogue values, there will be *separate files* with names defined by the following syntax:

<Parameter Name>.ATAB

where <Parameter Name> is the name of the parameter as given in the Parameter List (Table 1, Column 1). The character "A" in column 10 of the Parameter List will denote that the named table is for converting between raw and analogue values for the given parameter. The character "Y" in column 11 of the Parameter List denotes that the table does indeed exist for the given parameter.

Parameters that share a common lookup table will have the name of their actual conversion table given in column 11 of the Parameter List (see Table 1).

The character "N" in column 11 of the Parameter List denotes that an analogue parameter table does not exist for the given parameter.

### 5.1 Analogue parameter table structure

The header of the analogue parameter table file will consist of a number of lines, each line beginning with a # character in column one. These header lines will include the name of the analogue table itself, author, version, date and purpose. Subsequent lines will include any appropriate history information, as well as the headings for each column in the analogue table record. For information the column headings will also be defined between the "Start\_Columns" and "End\_Columns" header lines. An example header is shown below:

```
# filename:  
# author:  
# version:  
# date:  
# purpose:  
#  
#Start_Columns  
#Column 1  
#Column 2  
#End_Columns  
#  
#
```

.  
.  
#

The lookup table section, which follows immediately after the header, will have the structure defined in Table 4.

**Table 4: AnalogueTable Data Record Structure**

Column	Content	Description
1	Raw	Raw value of parameter (unsigned 32 bit integer)
2	Analogue	Analogue converted value of parameter (double)

The lines **START\_ATAB\_<Parameter Name>** and **END\_ATAB\_<Parameter Name>** will precede the respective start and end of the analogue table data records, signifying the start and end of the table. A number of comment lines beginning with # are permitted before the **START\_ATAB\_<Parameter Name>** identifier. No blank lines are permitted. The two-column table contains the raw and converted values.

Example analogue table:

```
# filename:      SMECPOSN.ATAB
# author:       Sunil D Sidher S.D.Sidher@rl.ac.uk
# version:      2.1
# date:         21 March 2002
# purpose:      Analogue Parameter Table for SPIRE QLA
# Start_Columns
# Raw SMEC position
# Converted SMEC position
# End_Columns
# Table updated after latest CQM tests
#
START_ATAB_SMECPOSN
  1230  0.20
  1240  0.25
  ...
  62145 4.90
END_ATAB_SMECPOSN
```

## 6. ENUMERATED PARAMETER TABLES

An enumerated table will be used for all boolean parameters (True/False, On/Off, etc) as well as for parameters with more than two states. For parameters possessing unique lookup tables between raw and enumerated values, there will be *separate files* with names defined by the following syntax:

<Parameter Name>.ETAB

where <Parameter Name> will be the name of the parameter as given in the Parameter List (Table 1, Column 1). The character "E" in column 10 of the Parameter List will denote that the named table is for converting between raw and enumerated values for the given parameter. The character "Y" in column 11 of the Parameter List denotes that the named table does indeed exist for the given parameter.

Parameters that share a common lookup table will have the name of their enumerated conversion table given in column 11 (see Table 1).

The character "N" in column 11 of the Parameter List denotes that an enumerated parameter table does not exist for the given parameter.

## 6.1 Enumerated Parameter Table Structure

The header of the enumerated parameter table file will consist of a number of lines, each line beginning with a # character in column one. These header lines will include the name of the enumerated table itself, author, version, date and purpose. Subsequent lines will include any appropriate history information, as well as the headings for each column in the enumerated table record. For information the column headings will also be defined between the "Start\_Columns" and "End\_Columns" header lines. An example header is shown below:

```
# filename:
# author:
# version:
# date:
# purpose:
#
#Start_Columns
#Column 1
#Column 2
#End_Columns
#
#
.
.
#
```

The lookup table section, which follows immediately after the header, will have the data record structure defined in Table 5.

**Table 5: Enumerated Table Data Record Structure**

Column	Content	Description
1	Raw	Raw value of parameter (unsigned 32 bit integer)
2	Enumerated	Enumerated value of parameter (string)

The lines **START\_ETAB\_<Parameter Name>** and **END\_ETAB\_<Parameter Name>** will precede the respective start and end of the enumerated table data records, signifying the start and end of the table. A number of comment lines beginning with # are permitted before the

**START\_ETAB\_<Parameter Name>** identifier. No blank lines are permitted. The two-column table contains the raw and enumerated values.

Example Enumerated Table:

```
# filename:      CHOP_MODE.ETAB
# author:       Sunil D Sidher S.D.Sidher@rl.ac.uk
# version:      5.1
# date:         5 September 2007
# purpose:      Enumerated Parameter Table for SPIRE QLA
# Start_Columns
# Raw Value
# Enumerated Value for CHOP_MODE
# End_Columns
# Table updated after latest CQM tests
#
START_ETAB_CHOP_MODE
0 STOP
1 STEP
2 TOGGLE
END_ETAB_CHOP_MODE
```

## 7. DERIVED PARAMETER TABLES

TBW

## 8. OUT OF LIMITS (OOL) PARAMETER TABLES

Parameters possessing low or high OOL values will have their OOL definitions stored in separate files with names defined by the following syntax:

<Parameter Name>.OTAB

where <Parameter Name> will be the name of the parameter as given in the Parameter List (Table 1, Column 1). The character "Y" in column 12 of the Parameter List will denote the fact that such an OOL table exists for the parameter concerned.

Parameters that share a common OOL table will have the name of the OOL table given in column 12 of the Parameter List (see Table 1).

The character "N" in column 12 of the Parameter List denotes that an OOL table does not exist for the given parameter.

### 8.1 OOL Parameter Table Structure

The table for each parameter will consist of a header section:

```
# filename:
# author:
```

```
# version:
# date:
# purpose:
#
#Start_Columns
#Column 1
#Column 2
#End_Columns
#
#
#
#
```

The OOL table will follow immediately afterwards and will have the following structure:

**Table 6: OOL Table Record Structure**

Column	Content	Description
1	OOL Type	String signifying the type of OOL event type: SOFT_LO: Low soft limit SOFT_HI: High soft limit HARD_LO: Low hard limit HARD_HI: High hard limit
2	Raw Value	Raw value of parameter for OOL event type in column 1
3	Converted Value	Converted value of parameter for OOL event type in column 1

The lines **START\_OTAB\_<Parameter Name>** and **END\_OTAB\_<Parameter Name>** will precede the respective start and end of the OOL table data records, signifying the start and end of the table. A number of comment lines beginning with # are permitted before the **START\_OTAB\_<Parameter Name>** identifier. No blank lines are permitted.

Example OOL Table:

```
# filename:    LIA01TEMP.ATAB
# author:     Sunil D Sidher S.D.Sidher@rl.ac.uk
# version:    4.01
# date:       6 September 2008
# purpose:    OOL Parameter Table for SPIRE QLA
# Start_Columns
# Limit type for parameter
# Value of parameter for limit type
# End_Columns
# Table updated after Tiger Team Congress
#
START_OTAB_LIA01TEMP
SOFT_LO 1548 0.22
SOFT_HI 3990 2.10
HARD_LO 1310 0.16
HARD_HI 4050 3.10
END_OTAB_LIA01TEMP
```



**SPIRE**

**Project Document**

SPIRE TM Parameter List Interface Control  
Document

**Ref:** SPIRE-RAL-DOC-001318

**Issue:** 0.6 (DRAFT)

**Date:** 27<sup>th</sup> June 2002

**Page:** 17 of 17

---

---