

**FDAS HSUM Module Design Specification for Intel Agilex F Implementation**

FDAS\_HSUM\_DS Revision 2 Draft C

Classification: UNRESTRICTED

Document type: DTE

Date: 2023-08-29

Status:

© Copyright 2023 SKA Observatory.

  This work is licensed under a [Creative Commons Attribution 4.0 International License](http://creativecommons.org/licenses/by/4.0/)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *Role* | *Name* | *Designation* | *Affiliation* | *Signature* | *Date* |
| Author | Martin Droog | FPGA Design Engineer | SKAO |  | 29/08/2023 |
| Owner | Ben Stappers | Head of SKA PSS Team | SKAO |  |  |
| Approver | Ben Stappers | Head of SKA PSS Team | SKAO |  |  |
| Released by | Lina Levin Preston | SKA PSS Team Scrum Master | SKAO |  |  |

| Document History | | |
| --- | --- | --- |
| Issue | Date | Comments |
| Issue 1 Draft A | 20 Nov 2017 | Initial Draft |
| Issue 1 Draft B | 14 Dec 2017 | Updates from internal review.   * Modules and sub-modules made clear. * Typo corrections |
| Issue 1 Draft C | 19 Mar 2018 | Updates from review at UoM on 15 Mar 2018:  Now reporting:-   * The FOP Row [p] of the Threshold Crossing Point * The FOP Column [b] of the Threshold Crossing Point * The Harmonic Number of the Threshold Crossing Point * The Summed power level at the Threshold Crossing point.   Reworded Pulsar as “Periodicity Candidate” since the origin of the observed signal has not yet been established. |
| Issue 1 Draft D | 16 Apr 2018 | Corrections from UoM  Corrections to Page 39 and 63:-  There are five 32 bit words per threshold crossing result:-  Word 1 = Indication that the result information in words 2 to 5 is valid.  Word 2 = Harmonic number of the threshold crossing.  Word 3 = FOP Column of the threshold crossing point.  Word 4 = FOP Row of the threshold crossing point.  Word 5 = Summed power level at the threshold crossing point. |
| Issue 1 Draft E | 27 Apr 2018 | Corrections and additional information requested by UoM:-  Page 6: Figure 2-1:  “Note: In this implementation CLD\_PAGE, CONV\_PAGE and HSUM\_PAGE are fixed (static~~) in this implementation~~”.   🡪  Note: In this implementation CLD\_PAGE, CONV\_PAGE and HSUM\_PAGE are fixed (static).  ***Response: Correction made***  Page 8:   “change from the low frequency set the high frequency set.”   “change from the low frequency set to the high frequency set.”  ***Response: Correction made***  Page 10:  Figure 3-2: the label at the bottom:  “DATA\_SUM[1 to summer\_g][~~8~~287:0]”  DATA\_SUM[1 to summer\_g][287:0]  ***Response: Correction made***  Page 14: in the Table 3-2 the value for p[0] is repeated at adjacent locations. Is it that those locations are ignored or skipped during the HSUM processing? Also would you clarify if they are actually populated by CONV  ***Response: Clarification added***  Page 17: the same data from the DDR4 externals SDRAMS in stored in the set of 40-off ….    the same data from the DDR4 externals SDRAMS is stored in a set of 40-off ….  ***Response: Correction made***  Page 19: Typos in Table 3-6: HPSEL Number to DDRIN RAM Mapping  In row two:  “DATA\_SUM[2][95:0] = DATA\_SUM[127:31]”   DATA\_SUM[2][95:0] = DATA\_SUM[127:32]  In row eight:  DATA\_SUM[8][288:0] = DATA\_SUM[1279:992]  DATA\_SUM[8][287:0] = DATA\_SUM[1279:992]  In row four:  DATA\_SUM[4][127:0] = DATA\_SUM[383:224]  DATA\_SUM[4][159:0] = DATA\_SUM[383:224]  In row five:  DATA\_SUM[5][127:0] = DATA\_SUM[543:384]  DATA\_SUM[5][159:0] = DATA\_SUM[543:384]  ***Response: Corrections made***  Page 23: “each with 11 orbital acceleration ambiguity values” perhaps to be stated as “each with 11 orbital acceleration ambiguity slopes”  ***Response: Changed from values to slopes in all relevant places***  Page 23: “An extra value of 0x56 shall be used to select a fixed value “M[31:0]” for the information to the summing tree instead of the FOP values from the DDRIN sub-module.”  🡪”An extra value of 0x56 shall be used to select a fixed value “M[31:0]” as a diagnostic input to the summing tree instead of the FOP values from the DDRIN sub-module.”  ***Response: Statement added***  Page 24: This is the value that is supplied to the harmonic summing tree instead of the FOP information 🡪 This is a diagnostic value that is supplied to the harmonic summing tree instead of the FOP information  ***Response: Statement added***  Page 27:  It appears that contents of Table 3-7, would be repeated for each of the harmonic as in Tables 2-3 to 2-10 (pages 38-40) of FDAS\_REQUIREMENTS\_AND\_ARCHITECTURE\_125 document. Although this information is stated in the first column description, it is not very obvious. It will be good to move such common information about the table to the caption field.  ***Response: Statement added to caption field***  Page 36: In Figure 3-8, it looks like the label T[8][31:0] is missing or got cropped out  ***Response: Label was at the bottom of the diagram. It has now been rotated by 90o so that it can be placed nearer the signal***  Page 38: section number seems missing for: Threshold Reporting: TREP Sub-Module (also at the Table of contents at page 3)  ***Response: Section number added***  Page 39: Typo in the fourth para:  SEED\_COL\_READ[21:]: 🡪 SEED\_COL\_READ[21:0]:  ***Response: Correction made***  Page 40: TREP submodule is receiving the ROW\_INFO. From the SUMMER sub-module. The ROW\_INFO appears from HP\_SEL in Figure 3-4 at page 25, and at Page 7, Figure 3-1 from the SUMMER Module. Not sure, it will be more accurate to leave it as originating form HP\_SEL?  ***Response: Added a note for ROW\_INFO that it is from HP\_SEL submodule within the SUMMER sub-module***  Page 41: In addition, a word for each harmonic shall indicate if the report is valid. – Clarification would be useful to add here.  ***Response: Note added to clarify the Valid indication***  Page 44:  The following configuration shall be stored in RAMs in the SUMMER sub-module via the HSUMMCI sub-module:-  **At this level of the document, would it be more precise to say**: The following configuration shall be stored in RAMs in the HP\_SEL and T\_SEL sub-modules via the HSUMMCI sub-module:-?  ***Response: Now state that the configuration is stored in HP\_SEL and T\_SEL sub-modules within the SUMMER sub-module.***  Page 48:  In the Table 4-1: HSUM Design Requirement Tags, it would be useful to add a comment against first three items: FDAS.HARMONIC\_SUM:010/A, 020/A and 030/A, stating something like “The actual arrangements of FOP data in a physical memory is further optimized as illustrated in Tables 3-2, 3-3 and 3-4”.  ***Response: Table References added.***  Page 50:  FDAS.THRESHOLD:020/A. The comment says: “In order to provide all the required information values shall be recorded which shall require translation in the PC/Computer.”  It will be useful to show, perhaps in the appendix, with an example for how to perform this translation in the Computer.  ***Response: Added*** *Error! Reference source not found.* ***and a reference to it.***    Page 52: FDAS.THRESHOLD:050/A: Based on descriptions at page 44, it appears that the PC will have to do a separate read operation to get to know the DM number correspond to every Threshold reports page. Wonder if it can be located in a contiguous location adjacent to the Threshold reports page, so that the PC can do one single block read to get them all in one block read operation (  FDAS.THRESHOLD:030/A )  ***Response: We can look at this during the implementation and update the bit map registers accordingly***  A top level processing flow to section 3.4 will be useful for an example case with single SUMMER.  ***Response: We can add this once we have implemented the design. The exact number of pipeline retimes within the SUMMER\_TREE is not yet known. We do have a swim lane diagram in the FDAS Implementation spec which provides the overall processing flow.***  References to the related figures or sections, where possible, will improve the readability**.** For example: at Page 26:   This is just to improve the readability: “The RAM contents shall actually be the addresses for the RAMs within the DDRIN\_STORE sub-module so that the values of the desired FOP row for each harmonic can be obtained and provided to the SUMMER\_TREE sub-module via the DATA\_SUM bus.” -> “The RAM contents shall actually be the addresses for the RAMs within the DDRIN\_STORE sub-module (Figure 3-2) so that the values of the desired FOP row for each harmonic can be obtained and provided to the SUMMER\_TREE sub-module (Figure 3-3) via the DATA\_SUM bus.”  ***Response: Extra references to Figures and Tables have been made where it seems appropriate.***  **Page 33:** Sections, 3.3.1, 3.3.2 and 3.3.3 are missing explicit description of “XXX sub-module shall receive the following from YYYY module”, such as what is given at page 39 for the TREP. These descriptions may be considered for uniformity and improved readability.  ***Response: These sections are sub-modules within the SUMMER sub-module. However for completeness information from the SUMMER sub-module has been copied to these sections.***  Page 37: A description for a) how each summer tree’s output adds to the higher harmonic adders, b) if there are mechanisms to clear the adders or output registers after each processing cycle, c) if there are any pipeline delay adjustments needed, and d) clocking and MC interface, will be useful.  ***Response: We can look at adding this information once the design has been implemented.***  Page 40: Is it possible to infer the Time available to read TREP results? Do we have memory in FPGA to accumulate few DMs at each page before it can be send via PCIe? We can discuss about this if required.  ***Response: The theoretical time available to read the results for a DM is already detailed in the FDAS Implementation spec (i.e it is the HSUM DM processing time of 335.3ms for the target hardware). The measured time will be available after implementation. Accumulating results for a few DMs will need to be discussed as it will affect RAM usage and need extra signalling information to the PC/Computer.***  Page 51: FDAS.THRESHOLD:030/A and FDAS.THRESHOLD:040/A:  The comment says that “The reports for a DM shall all be readable via the MC interface and shall remain static whilst the next DM is being processed.” I wonder if the values can remain static for up to 10 DMs or so, so that the PC can read them in a bulk transfer of few kilobytes and that will be efficient without using too much of PCIe IO. We can discuss this further on the feasibility for incorporating this suggestion.   We can also discuss about this if required.  ***Response: Accumulating results for a few DMs will need to be discussed as it will affect RAM usage and need extra signalling information to the PC/Computer.***  Page 51: FDAS.THRESHOLD:035/A and, as discussed before (for Page 41: 3.3.5 T\_FLITER\_EN=1), it may be possible to consider this item under relaxations to reduce the design complexity.  ***Response: We intend to implement this feature unless it is definitely not required.***  It would be useful to explicitly state how the negative side of the FOP are proceeded.  ***Response: The HSUM module doesn’t have the concept of the negative side of the FOP. This is determined purely by the programming of HSUM to set the FOP row positions of the seed\_f0 and the HP\_SEL selection to select the appropriate FOP rows for the harmonics. These Row positions cover both positive and negative row locations within the FOP (see Table 3‑5 which relates HSUM programming to FOP Row).***  The expected HSUM processing time for an example parameter set maybe stated in the appendix, perhaps.  ***Response: We have already provided this information in the FDAS Implementation specification, which contains the expected DM processing time for the intended target card Hardware (335.3ms). The FDAS Implementation specification will also contain the measured results for the completed design.*** |
| Issue 1 Draft F | 22 Jun 2018 | Removed the TEST\_NUM signal input from TREP (signal not required). |
| Issue 1 Draft G | 19/11/2018 | Summary of functional changes:   1. Selection of configuration values for each analysis run moved to one place: TGEN 2. Fetching of FOP data from DDR and running of summation have been decoupled to improve processing time. 3. Number of pages of FOP data stored in DDRIN\_STORE increased from 2 to 4 to reduce the impact of DDR latency on processing time. 4. FOP column numbers to read for each harmonic has been changed. 5. Encoding of HPSEL modified to simplify design implementation. 6. Generic ‘fop\_row\_g’ changed to MC configuration value FOP\_ROW. 7. Added MC configuration value FOP\_COL\_OFFSET to take into account the shift in FOP column positions introduced by the filter shift in CONV. 8. Added LAST\_RESULT signal to HP\_SEL in SUMMER, for use by TREP when filtering results. 9. Addressing of RAMs in the bitmap altered to group configuration values more appropriately. 10. Added concept of register and RAM based stores for results in TREP.   Sections/Figures/Tables updated:  Figure 3‑1: Due to 1)-10).  Section 3.1: Due to 1), 2), 3), 6), 7), 10).  Figure 3‑2 : Due to 1), 2), 3), 6), 7).  Section 3.2 : Due to 1), 2), 3), 6), 7).  Section 3.2.1 : Due to 2), 3), 4), 6), 7).  Table 3‑1 : Due to 4).  Table 3‑2 : Added missing ‘-‘ for ‘P[-38]’ entry.  Table 3‑3 : Added missing ‘-‘ for ‘P[-38]’ entry. Corrected table label.  Table 3‑4 : Added missing ‘-‘ for ‘P[-38]’ entry. Corrected table label.  Section 3.2.2 : Due to 3).  Table 3‑5 : Due to 5), 10).  Figure 3‑3 : Due to 1), 5), 8), 9).  Section 3.3 : Due to 1), 5), 8), 9).  Figure 3‑4 : Due to 1), 5), 8), 9).  Section 3.3.1 : Due to 1), 5), 8), 9).  Figure 3‑6 : Due to 9).  Section 3.3.2 : Due to 9).  Figure 3‑7 : Corrected indices for DATA\_SUM feeding sum #15.  Figure 3‑11 : Due to 1), 8), 10).  Section 3.3.3.1 : Due to 1), 10). Removed repeat of Table 3‑5.  Section 3.4.1 : Due to 2), 10). Reduced number of result words to 4, by combining valid bit and harmonic number into one word, to reduce the number of reads required to get the results.  Section 0 : Due to 7). T\_EXC, P\_EXC, S\_EXC now stored in RAMs.  Section 3.5.1 : Address bus width reduced to 16 bits.  Section 3.5.2 : Address bus width reduced to 16 bits. Diagram updated to reflect timing when reading from a RAM, and that MCRWN must be high.  Table 5‑1 : MCADDR width changed to 16 bits.  Section 6 : Complete overhaul.  Section 7 : Removed ‘fop\_row\_g’. Added ‘adder\_latency\_g’. |
| Issue 1 Draft H | 26/11/2018 | Corrected references to Table 3-9 to Table 3‑5. |
| Issue 1 Draft I | 29/11/2018 | Removed section 6.3.3. (future function, not in current design). |
| Issue 1 Draft J | 18/04/2019 | Added clarification in section 1 for column numbering used within the design and frequencies represented by each column. Reverted column numbers read for each harmonic to that specified in Draft F (Table 3-1). |
| Issue 1 Draft K | 03/06/2019 | Updated for version of HSUM in FDAS release 3.x. that supports up to 16 harmonics instead of 8. |
| Issue 1 Draft L | 20/06/2020 | Added enhanced summing tree (f0 to 8\*f0) (section 3.3.3.1). |
| Issue 1 Draft M | 24/09/2020 | Added enhanced summing tree (9\*f0 to 12\*f0) (Figure 3-10).. |
| Issue 2 Draft A | 18/06/2022 | Updates from internal review:-  General update to describe how the generic “harmonic\_g” controls the DDRIN and SUMMER sub-modules.  Updated design to use the Intel Agilex Floating Point Adder  Updated the design as the Intel Agilex altsyncram cannot support different clock frequencies on the A and B ports. |
| Issue 2 Draft B | 27/03/2023 | Minor Typo Corrections and extra descriptions to align with the implemented design |
| Issue 2 Draft C | 29/08/2023 | Added hsumreset\_sync module to distribute the system reset signal to eliminate recovery timing warnings. |

**Table of Contents**

1 Introduction 10

2 Place in the System 12

3 Functional Specification 13

3.1 Timing Generator: TGEN Sub-Module 14

3.2 DDR Input: DDRIN Sub-Module 18

3.2.1 Address Generator: ADDR\_GEN Sub-Module 21

3.2.2 DDR Input Data Store: DDRIN\_STORE Sub-Module 28

3.3 Harmonic Summing: SUMMER Sub-Module 32

3.3.1 Harmonic Orbital Acceleration: HP\_SEL Sub-Module 36

3.3.2 Threshold Selection: T\_SEL Sub-Module 43

3.3.3 Original Summing Tree: SUMMER\_TREE Sub-Module 47

3.3.4 Enhanced Summing Tree 50

3.4 Threshold Reporting: TREP Sub-Module 53

3.4.1 Harmonic Results Store: HRES Sub-Module 56

3.4.2 Threshold Crossing Results Filter: FILTER Sub-module 59

3.5 HSUMMCI: HSUM Module Micro Configuration Interface 61

3.5.1 Write to HSUMMCI 62

3.5.2 Read from HSUMMCI 63

3.6 Reset Synchronisation: RESET\_SYNC Sub-Module 65

4 Design Requirement Tags 66

5 Interface Specification 71

6 MCI Memory Mapped Interface 73

6.1 FOP SELECTION CONFIGURATION 73

6.1.1 FOP ROW (FILTER "P") SELECTION 73

6.1.2 THRESHOLD CONFIGURATION 75

6.2 ANALYSIS RUN PARAMETERS 77

6.2.1 GLOBAL PARAMETERS 77

6.2.2 PER SUMMER RUN PARAMETERS 80

6.2.3 DM COUNTER 81

6.3 SUMMING RESULTS 82

6.3.1 HARMONIC THRESHOLD CROSSING RESULTS 82

6.3.2 STORAGE EXCEEDED REPORTS 83

7 Design Parameterisation 85

8 Updates for Intel Agilex Implementation 86

8.1 Floating Point DSP Adder 86

8.1.1 Instantiation of the Floating Point Adder 87

8.2 Replacement of Intel altsyncram 87

9 Abbreviations and Acronyms 88

# Introduction

This document captures the requirements for the HSUM module used in the FDAS FPGA.

This document has been updated to include the changes required for the HSUM module to be targeted for the Intel Agilex family, with the changes detailed in section 8.

The HSUM module is required to perform “Harmonic Summing” of “Periodicity Candidates” i.e. radio telescope observations with periodically varying power levels over time such as those generated by a pulsar.

The fundamental tone of each Periodicity Candidate is known as a “seed\_f0” and its location in the FOP (Filter Output Plane) identifies its frequency (FOP column) and orbital acceleration around another body.

From the FOP location of the seed\_f0 it is possible to determine the location of its harmonics which can then be summed to determine if the power level is of significant interest. The maximum number of harmonics that the design can support is controlled by a generic “harmonic\_g”. The original intention was to support a maximum of 16 harmonics (including the fundamental), however currently the enhanced summing tree architecture in the SUMMER sub-module can only support 12 harmonics.

A complication is due to the fact that each location of the FOP does not identify an exact Periodicity Candidate frequency and orbital acceleration, but rather a range. Hence, when performing the Harmonic Summing this ambiguity has to be taken into account via additional summing calculations. For a seed\_f0 in a particular column of the FOP the FOP locations that need to be analysed by the harmonic summing process are shown in Figure 1‑1 below:-

**FOP columns required to explore the possibility of “seed\_f0” existing in column “B” (orbital acceleration range p = -10 to 10)**

P = 0

P = 42

P = -42

P = 10

……………………

“B”

:

1 col

3 cols

3 cols

5 cols

5 cols

9 cols

b = 1

b = 222

f0

2xf0

3xf0

4xf0

5xf0

8xf0

5x“B”

4x“B”

3x“B”

2x“B”

P = -10

**1** = 42

**2** = 42

**3** = 42

**4** = 42

**5** = 42

**6** = 42

**7** = 42

**8** = 42

**9** = 42

**10** = 42

**1**

**1** = 42

**1**

**2** = 42

**1**

**6** = 42

**1**

**7** = 42

**3**

**6** = 42

**1**

**3** = 42

**1**

**4** = 42

**1**

**5** = 42

**3**

**9** = 42

**4**

**0** = 42

**3**

**5** = 42

**3**

**7** = 42

**3**

**8** = 42

**3**

**4** = 42

**3**

**2** = 42

**3**

**3** = 42

**2**

**1**1= 42

**2**

**0** = 42

**1**

**8** = 42

**1**

**9** = 42

6x“B”

7x“B”

8x“B”

**2**

**2**1= 42

**2**

**3**1= 42

**2**

**4**1= 42

**2**

**5**1= 42

**2**

**6**1= 42

**2**

**7**1= 42

**2**

**8**1= 42

**2**

**9**1= 42

**3**

**0**1= 42

**3**

**1**1= 42

7 cols

6xf0

7 cols

7xf0

*Seed\_f0 fundamental tone f0 = “B” in the first column analysed*

Figure ‑ : FOP columns required to explore “seed\_f0" in FOP Column “B” for 8 harmonics

In total, 40 columns of the FOP need to be analysed if eight harmonics (including the fundamental tone) are to be summed. If fewer harmonics are summed fewer FOP columns need to be analysed. For 16 harmonics, a total of 144 columns of the FOP need to be analysed.

In HSUM configuration (SEED\_COL) and results (FOPCOL), column numbering is 0-based, thus:

* SEED\_COL=0 identifies the first column of the FOP which has a centre frequency of 0 Hz,
* SEED\_COL=1 identifies the second column of the FOP which has a centre frequency of *f* Hz, where *f* is the frequency step from one column to the next.

# Place in the System

The HSUM Module’s place in the system is shown highlighted in Figure 2‑1 below:-

PCIe

**CLD**

**CTRL**

FDAS Control

CLD\_DONE

**CONV**

data[63:0]

ready

**HSUM**

**FDAS**

CONV\_DONE CONV\_FFT\_REDAY

valid

**PCIE HARD IP MACRO (INTEL IP)**

**MCI\_TOP**

CLD\_DM\_TRIGGER

CLD\_ENABLE

CLD\_PAGE[31:0]

CONV\_DM\_TRIGGER

CONV\_ENABLE

CONV\_PAGE[31:0]

IFFT\_LOOP\_NUM[5:0]

HSUM\_DM\_TRIGGER

HSUM\_ENABLE

HSUM\_PAGE[31:0]

HSUM\_DONE

**PCIF**

ADDR

DATA[511:0]

**DDRIF2 #1**

**DDRIF2 #2**

ADDR

DATA[511:0]

DATA[511:0]

ADDR

ADDR

DATA[511:0]

*Note: The PCIe Hard Macro can read and write to both External DDR memories for diagnostic purposes. Not shown in this figure for clarity.*

*Note: CLD, CONV and HSUM are designed with generically sized data width interfaces to DDRIF2 for a future implementation, but in this implementation the DDRIF2 data width is fixed. CLD, CONV and HSUM are also designed with paging of the DDR memory for a future implementation.*

**FDAS\_DDR\_CONTROLLER\_**

**CALIBRATION (INTEL IP) #1**

**FDAS\_DDR\_CONTROLLER\_**

**CALIBRATION (INTEL IP) #2**

DMA Transfer

**External 4 Gibi-Byte**

**DDR4 SDRAM Memory #1**

**External 4 Gibi-Byte**

**DDR4 SDRAM Memory #2**

MC

Interface

Via

PCIe

*Note: In this implementation CLD\_PAGE, CONV\_PAGE and HSUM\_PAGE are fixed (static) in this implementation. However with a future implementation with more DDR Interfaces to CONV/HSUM a Paging technique shall enable increased processing performance.*

MC Bus

OVERLAP\_SIZE[9:0]

FOP\_SAMPLE\_NUM[22:0]

**MSIX**

MSI-X

Figure ‑ – HSUM Location in FDAS

# Functional Specification

The HSUM Module architecture is shown in Figure 3‑1 below:-

MCADDR[15:0]

**HSUM**

DDR\_ADDR[31:0]

**ddr\_g , summer\_g, harmonic\_g, adder\_latency\_g**

**DDRIN**

DATA\_SUM

[1 to summer\_g]

[DWIDTH:0]

MCRWN

CLK\_SYS

RST\_SYS\_N

MCMS

A\_SET

DATA\_VALID

MCDATAIN[31:0]

MCDATAOUT[31:0]

CLK\_MC

RST\_MC\_N

HSUM\_TRIGGER

HSUM\_ENABLE

HSUM\_DONE

B\_START[1 to 2][21:0]

B\_STOP[1 to 2][21:0]

THRESH\_SET[21:0]

SEED\_COL\_READ[21:0]

NEW\_COL

DONE\_REQ

H[1 to 2][3:0]

HSUM\_PAGE[31:0]

DDR\_DATA

[512\*ddr\_g-1:0]

DDR\_READ

WAIT\_REQUEST

ANALYSIS\_RUN

**SUMMER**

HPSEL

[1 to summer\_g]

[1 to harmonic\_g][6:0]

HPSEL\_EN

[1 to summer\_g]

[1 to harmonic\_g]

HPSEL\_WRITE[6:0]

MC\_ADDR[15:0]

HPSEL\_READ[6:0]

P\_EN

[1 to summer\_g][1 to 2][4:0]

M[31:0]

DONE\_SUM

[1 to summer\_g]

T\_WRITE[31:0]

T\_READ[31:0]

TC

[1 to summer\_g]

[1 to NODE]

PWR

[1 to summer\_g]

[1 to NODE]][31:0]

ROW\_INFO

[1 to summer\_g]

[1 to harmonic\_g][6:0]

T\_RESULTS[31:0]

MC\_ADDR[11:0]

T\_FILTER\_EN

DM\_CNT\_RESET

DM\_CNT[31:0]

EXC[31:0]

DDRIN\_RD\_PAGE

T\_SET

**HSUMMCI**

**TREP**

NEW\_SUM

FOP\_ROW[1 to 2][6:0]

**TGEN**

*Note: MC HPSEL values for 16 harmonics for 21 seed\_f0 locations for 11 acceleration ambiguity slopes per analysis run per SUMMER stored in SUMMER Config RAM.*

*MC Threshold Values for 16 harmonics for 21 seed\_f0 locations per analysis run per SUMMER stored in SUMMER Config RAM*

*NOTE: The Enhanced SUMMER only supports up to 12 harmonics*

A[1 to summer\_g][1 to 2]][3:0]

(x summer\_g)

CLEAR\_RESULTS/SAVE\_RESULTS/TRIGGERED

A[1 to summer\_g][3:0]/P\_EN[1 to summer\_g][4:0]

SAVE\_DONE

*Note: The following is stored in RAM in TREP:*

* *400 Results (25 per harmonic) for each analysis run, each result containing four 32-bit words*
* *16 Exceeded reports (1 per harmonic) for each analysis run, each report containing three 32-bit words.*

FOP\_ROW[6:0]

H[3:0]

HPSEL\_WREN

T\_WREN

LAST\_RESULT

[1 to summer\_g]

DONE\_READ

SEED\_COL\_SUM[21:0]

TRIGGERED

FOP\_COL\_OFFSET[8:0]

|  |  |
| --- | --- |
| harmonic\_g | DWIDTH |
| 8 | 1279 |
| 9 | 1567 |
| 10 | 1919 |
| 11 | 2271 |
| 12 | 2687 |
| 13 | 3103 |
| 14 | 3583 |
| 15 | 4063 |
| 16 | 4607 |

|  |  |
| --- | --- |
| harmonic\_g | NODE |
| 8 | 40 |
| 9 | 49 |
| 10 | 60 |
| 11 | 71 |
| 12 | 84 |
| 13 | 97 |
| 14 | 112 |
| 15 | 127 |
| 16 | 144 |

**RESET**

**\_SYNC**

resets

Figure ‑ : HSUM Architecture Block Diagram

## Timing Generator: TGEN Sub-Module

The TGEN sub-module shall control the DDRIN and SUMMER sub-modules (Figure 3‑2 & Figure 3‑3) to ensure that processing occurs at the correct time.

A rising edge of the HSUM\_TRIGGER signal from the CTRL module shall cause the HSUM module to commence processing the FOP for a DM as long as HSUM\_ENABLE is asserted.

If HSUM\_ENABLE is subsequently de-asserted the SUMMER sub-module shall stop processing when it has completed the harmonic summing for the current set of up to 144 FOP columns. To allow the results obtained so far to be examined, TGEN shall pulse the “SAVE\_RESULTS” signal for a single CLK\_SYS cycle.

If HSUM\_ENABLE is subsequently re-asserted the SUMMER sub-module shall commence processing the next set of up to 144 FOP columns from the DDRIN sub-module and DDRIN sub-module shall commence reading the next set of up to 144 FOP columns from the external DDR4 SDRAM.

The TGEN sub-module shall receive the following information from the HSUMMCI sub-module (see Figure 3‑1):-

* **A\_SET**: This is a single bit which shall indicate the number of harmonic summing analysis runs to be performed on the FOP for the DM. If A\_SET = 0 only one analysis run shall be performed. If A\_SET =1 two analysis runs shall be performed.
* **B\_START[1 to 2 analysis runs][21:0]**: For each harmonic summing analysis run B\_START shall indicate the FOP column at which the analysis run shall start to look for a potential Periodicity Candidate seed\_f0.
* **B\_STOP[1 to 2 analysis runs][21:0]**: For each harmonic summing analysis run B\_STOP shall indicate the FOP column at which the analysis run shall stop looking for a potential Periodicity Candidate seed\_f0.
* **THRESH\_SET[21:0]**: The Harmonic Summing calculation results are compared against a set of thresholds to determine if the result is of interest. Periodicity Candidates with a low frequency require a different set of threshold values to Periodicity Candidates with a high frequency. The THRESH\_SET configuration value shall indicate the FOP column number for a potential Periodicity Candidate seed\_f0 at which the thresholds shall change from the low frequency set to the high frequency set.
* **P\_EN[1 to 2 analysis runs][1 to summer\_g][4:0]**: This is the number of actual orbital acceleration values for a potential Periodicity Candidate that are analysed. This is synonymous with the number of FOP rows in which the seed\_f0 may exist. A maximum of 21-off orbital acceleration values (i.e. seed\_f0 located in 21 FOP rows) are analysed. If fewer orbital acceleration values are analysed the harmonic summing time is reduced. A FOP for a DM may be analysed twice requiring two configuration values per SUMMER, with the appropriate value for the analysis run being passed to the SUMMER and TREP.
* **A[1 to 2 analysis runs][1 to summer\_g][3:0]**: This is the number of actual orbital acceleration ambiguity slopes for the harmonics of each seed\_f0. A maximum of 11-off orbital acceleration ambiguity slopes (i.e. 11 possible row locations of each harmonic of a seed\_f0). If fewer orbital acceleration slopes are analysed the harmonic summing time is reduced. A FOP for a DM may be analysed twice requiring two configuration values per SUMMER, with the appropriate value for the analysis run being passed to the SUMMER and TREP.
* **H[1 to 2 analysis runs][3:0]**: This is the number of actual harmonics that are to be summed for each seed\_f0. A maximum of “harmonic\_g”-off harmonics (including the seed\_f0) may be summed. If fewer harmonics are analysed the harmonic summing time is reduced. A FOP for a DM may be analysed twice requiring two configuration values with the appropriate value for the analysis run being passed to DDRIN the SUMMER and TREP.
* **FOP\_ROW[1 to 2 analysis runs][6:0]**: This is the number of FOP rows that will be read from the DDR SDRAM memory. If fewer FOP Rows are read from DDR SDRAM the harmonic summing time is reduced, but the maximum orbital acceleration supported is reduced. A FOP for a DM may be analysed twice requiring two configuration values with the appropriate value for the analysis run being passed to DDRIN.

The TGEN sub-module shall receive the following from the DDRIN sub-module:-

* **DONE\_REQ**: This is a single cycle pulse to indicate that the requests to the DDR interface for the current set FOP columns is complete.
* **DONE\_READ**: This is a single cycle pulse to indicate that the requested data has been returned by the DDR interface and a complete page of FOP columns is available.

The TGEN sub-module shall receive the following from the SUMMER sub-module :-

* **DONE\_SUM[1 to summer\_g]**: This is a single cycle pulse to indicate that the summation for the current set of FOP columns is complete. When more than one SUMMER sub-module is present, they will not complete at the same time, thus TGEN will have to latch the signals and generate an internal signal to indicate that all SUMMERs have completed.

The TGEN sub-module shall receive the following from the TREP sub-module :-

* **SAVE\_DONE**: This is a single cycle pulse to indicate that the transfer of results from the register store to the RAM store is complete.

Once triggered by HSUM\_TRIGGER (assuming HSUM\_ENABLE =1) the TGEN sub-module shall :-

* Pulse the “TRIGGERED” signal to TREP and DDRIN for a single CLK\_SYS cycle. This shall increment the DM\_COUNT in TREP and reset the write page counter in DDRIN.
* Set “ANALYSIS\_RUN” to 0, “T\_SET” to 0.
* Select the configuration values of B\_START[21:0], B\_STOP[21:0], FOP\_ROW[6:0], P\_EN[4:0], A[3:0] and H[3:0] for the first analysis run.
* Initialise the read page counter “DDRIN\_RD\_PAGE” to 0.

At the start of the analysis run, TGEN shall :-

* Initialise a seed column counter “SEED\_COL\_READ” to B\_START[21:0].
* Initialise a seed column counter “SEED\_COL\_SUM” to B\_START[21:0].
* Initialise a count “REQUEST\_COUNT” to 0. This shall count the number of requests for column data that have been initiated by DDRIN.
* Initialise a flag “REQ\_IN\_PROGRESS” to 0. This shall indicate whether DDRIN is currently requesting data.
* Initialise a count “FULL\_PAGE\_COUNT” to 0. This shall count the number of completed data requests made by DDRIN.
* Pulse the “CLEAR\_RESULTS” signal to TREP for a single CLK\_SYS cycle to initialise the results store.

While the analysis run is active, TGEN shall :-

* If “REQUEST\_COUNT” < 4 and “REQ\_IN\_PROGRESS” = 0 and “SEED\_COL\_READ” <= B\_STOP[21:0] then:
  + Pulse “NEW\_COL” signal to the DDRIN sub-module for a single CLK\_SYS cycle.
  + Increment “REQUEST\_COUNT”.
  + Set “REQ\_IN\_PROGRESS” to 1.
* If “DONE\_REQ” = 1 then:
  + Set “REQ\_IN\_PROGRESS” to 0.
  + Increment “SEED\_COL\_READ”.
* If “DONE\_READ” = 1 then:
  + Increment “FULL\_PAGE\_COUNT”.
* If “FULL\_PAGE\_COUNT” > 0 and no summation is in progress and “SEED\_COL\_SUM” <= B\_STOP[21:0] then:
  + Pulse “NEW\_SUM” signal to the SUMMER sub-module for a single CLK\_SYS cycle.
* If “DONE\_SUM” = 1 then:
  + Decrement “REQUEST\_COUNT”.
  + Decrement “FULL\_PAGE\_COUNT”.
  + Increment “DDRIN\_RD\_PAGE”.
  + Increment “SEED\_COL\_SUM”.
* If “SEED\_COL\_SUM” >= “THRESH\_SET[21:0]” then:
  + Set “T\_SET” to 1.

Once the summation for seed column “B\_STOP[21:0]” has been processed then the analysis run is complete and TGEN shall :-

* Pulse “SAVE\_RESULTS” to sub-module TREP for a single CLK\_SYS cycle.
* Wait for the “SAVE\_DONE” signal from TREP to go high.
* If A\_SET = 0 or ANALYSIS\_RUN = 1, pulse the HSUM\_DONE signal to the CTRL module to indicate processing in the HSUM module for the DM is complete.
* If A\_SET = 1:-
  + Set “ANALYSIS\_RUN” to 1, T\_SET to 0.
  + Select the configuration values of B\_START[21:0], B\_STOP[21:0], FOP\_ROW[6:0], P\_EN[4:0], A[3:0] and H[3:0] for the second analysis run.
  + Commence a second analysis run using the same procedure as described above.

## DDR Input: DDRIN Sub-Module

The DDRIN sub-module architecture is shown in Figure 3‑2 below:-

DDR\_ADDR[31:0]

**ddr\_g, harmonic\_g,summer\_g**

**ADDR\_GEN**

DATA\_SUM

[1 to summer\_g]

[DWIDTH:0]

RST\_SYS\_N

DATA\_VALID

SEED\_COL [21:0]

NEW\_COL

DONE\_READ

H[3:0]

HSUM\_PAGE[31:0]

DDR\_DATA

[512\*ddr\_g-1:0]

DDR\_READ

WAIT\_REQUEST

FOP\_ROW[6:0]

HPSEL

[1 to summer\_g]

[1 to harmonic\_g]

[6:0]

DDRIN\_RD\_PAGE[1:0]

**ddr\_g**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 1 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  | 2 | 3 | 4 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  | 5 | 6 | 7 |
|  |  |  |  |  |  |  |  |  |  |  |  | 8 | 9 | 10 | 11 | 12 |
|  |  |  |  |  |  |  |  |  |  |  |  | 13 | 14 | 15 | 16 | 17 |
|  |  |  |  |  |  |  |  |  |  | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
|  |  |  |  |  |  | | |  |  | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
|  | |  | | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 |
| 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 |
| 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 |
| 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 |
| 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 | 81 | 82 | 83 | 84 |
|  | |  | | 85 | 86 | 87 | 88 | 89 | 90 | 91 | 92 | 93 | 94 | 95 | 96 | 97 |
| 98 | 99 | 100 | 101 | 102 | 103 | 104 | 105 | 106 | 106 | 108 | 109 | 110 | 111 | 112 |
| 113 | 114 | 115 | 116 | 117 | 118 | 119 | 120 | 121 | 122 | 123 | 124 | 125 | 126 | 127 |
| 128 | 129 | 130 | 131 | 132 | 133 | 134 | 135 | 136 | 137 | 138 | 139 | 140 | 141 | 142 | 143 | 144 |

|  |
| --- |
|  |

**(x summer\_g)**

DATA\_SUM[1 to summer\_g][1][31:0]

HPSEL[1 to summer\_g][2][6:0]

DATA\_SUM[1 to summer\_g][2][95:0]

HPSEL[1 to summer\_g][3][6:0]

DATA\_SUM[1 to summer\_g][3][95:0]

HPSEL[1 to summer\_g][4][6:0]

DATA\_SUM[1 to summer\_g][4][159:0]

HPSEL[1 to summer\_g][5][6:0]

DATA\_SUM[1 to summer\_g][5][159:0]

HPSEL[1 to summer\_g][6][6:0]

DATA\_SUM[1 to summer\_g][6][223:0]

HPSEL[1 to summer\_g][7][6:0]

DATA\_SUM[1 to summer\_g][7][223:0]

HPSEL[1 to summer\_g[8][6:0]

DATA\_SUM[1 to summer\_g][8][287:0]

STORE\_ADDR[2:0]

WREN[1 to No of RAMS]

CLK\_SYS

**DDRIN\_STORE Page 1-4**

*RAM Number storing its respective FOP Col for harmonic summing (up to 85-of FOP rows per FOP Col with 32-bits per entry)*

DONE\_REQ

TRIGGERED

WR\_PAGE[1:0]

FOP\_COL\_OFFSET[8:0]

HPSEL[1 to summer\_g][1][6:0]

DATA\_SUM[1 to summer\_g][9][287:0]

HPSEL[1 to summer\_g][10][6:0]

DATA\_SUM[1 to summer\_g][10][351:0]

HPSEL[1 to summer\_g][11][6:0]

DATA\_SUM[1 to summer\_g][11][351:0]

HPSEL[1 to summer\_g][12][6:0]

DATA\_SUM[1 to summer\_g][12][415:0]

HPSEL[1 to summer\_g][13][6:0]

DATA\_SUM[1 to summer\_g][13][415:0]

HPSEL[1 to summer\_g][14][6:0]

DATA\_SUM[1 to summer\_g][14][479:0]

HPSEL[1 to summer\_g][15][6:0]

DATA\_SUM[1 to summer\_g][15][479:0]

HPSEL[1 to summer\_g[16][6:0]

DATA\_SUM[1 to summer\_g][16][543:0]

HPSEL[1 to summer\_g][9][6:0]

|  |  |
| --- | --- |
| harmonic\_g | DWIDTH |
| 8 | P\_EN |
| 9 | 1567 |
| 10 | 1919 |
| 11 | 2271 |
| 12 | 2687 |
| 13 | 3103 |
| 14 | 3583 |
| 15 | 4063 |
| 16 | 4607 |

|  |  |
| --- | --- |
| harmonic\_g | No of RAMS |
| 8 | 40 |
| 9 | 49 |
| 10 | 60 |
| 11 | 71 |
| 12 | 84 |
| 13 | 97 |
| 14 | 112 |
| 15 | 127 |
| 16 | 144 |

*Note the number of RAMs shown here is for when harmonic\_g = 16*

HPSEL\_EN

[1 to summer\_g]

[1 to harmonic\_g]

HPSEL\_EN enables associated RAM output

The DDRIN sub-module (Figure 3‑2) shall read the required FOP Columns from the DDR4 external SDRAM memory via the DDRIF2 module and DDR\_CONTROLLER module in FDAS.

Figure ‑ : DDRIN Sub-module Architecture

The DDRIN sub-module shall use the “ddr\_g” generic to define the number of DDR4 external SDRAM interfaces that serve the DDRIN sub-module. Each DDR4 external SDRAM interface presents a 512-bit data bus to the DDRIN sub-module.

The DDRIN sub-module shall use the “summer\_g” generic to define the number of RAM Stores required to support the SUMMER sub-modules (Figure 3‑3). Each SUMMER sub-module requires a RAM Store in the DDRIN sub-module, since each SUMMER sub-module needs to access different locations within the FOP columns.

The DDRIN sub-module shall use the “harmonic\_g” generic to define the number of harmonics supported and thus the number of individual RAMs within the RAM Store (Each RAM stores the data for one FOP column).

The DDRIN sub-module shall receive the following from the CTRL module (see Figure 2‑1):-

* **HSUM\_PAGE[31:0]**: This is an offset that the DDRIN sub-module shall apply to the DDR4 SDRAM address that it calculates to support FDAS implementations which have FOPs stored in different regions of the DDR4 external SDRAM memory.

The DDRIN sub-module shall receive the following configuration from the HSUMMCI module (see Figure 3‑1):-

* **FOP\_COL\_OFFSET[8:0]**: Due to the filter shifting in CONV, the first valid column of FOP data does not reside in the first column in the DDR memory but instead at column “FOP\_COL\_OFFSET”.

The DDRIN sub-module shall receive the following from the TGEN sub-module (see Figure 3‑1):-

* **TRIGGERED**: This is a pulsed indication (1 cycle wide) and shall be used to reset the write page “WR\_PAGE[1:0]” to 0.
* **NEW\_COL**: This is a pulsed indication (1 cycle wide) to command the DDRIN sub-module to commence reading the required FOP columns from the DDR4 external SDRAM memory.
* **SEED\_COL[21:0]**: (SEED\_COL\_READ[21:0] from TGEN.) This is the FOP column that contains the seed\_f0 locations for a potential Periodicity Candidate. From this SEED\_COL[21:0] value, and the value of the “ddr\_g” generic it is possible to calculate DDR4 external SDRAM memory addresses to obtain the up to 144-off FOP columns for harmonic summing (as indicated by H[3:0] for the analysis run).
* **H[3:0]**: This is an indication of the number of harmonics that shall need to be summed. If fewer than “harmonic\_g” harmonics need to be analysed, fewer FOP columns need to be read from the DDR4 external SDRAM, thus saving processing time.
* **FOP\_ROW[6:0]:** This is an indication of the number of rows in the FOP. The fewer the number of rows the less time it takes to read the required FOP columns from the DDR4 external SDRAM memory.
* **DDRIN\_RD\_PAGE[1:0]**: This is an indication of which DDRIN\_STORE sub-module page shall be used to supply data to the SUMMER sub-module(s) to perform the harmonic summing.

The DDRIN sub-module shall receive the following from the SUMMER sub-module(s) (see Figure 3‑1 & Figure 3‑3):-

* **HPSEL[1 to summer\_g][1 to harmonic\_g][6:0]**

**& HPSEL\_EN[1 to summer\_g][1 to harmonic\_g]**: This is the set of information that becomes the addresses to the DDRIN\_STORE sub-module RAMs from the SUMMER sub-modules to select the FOP Rows (which are the orbital accelerations “P”) for harmonic summing. Each SUMMER sub-module instance acts independently, simultaneously requiring different FOP locations for harmonic\_g harmonics. Hence for each SUMMER sub-module instance harmonic\_g-off independent addresses are generated. Each address accesses a number of RAMs according to the number of columns processed for each harmonic (1 col for 1st harmonic also known as the fundamental, 3 cols for 2nd and 3rd harmonics, 5 cols for 4th and 5th harmonics, 7 cols for 6th and 7th harmonics and 9 cols for 8th and 9th harmonic, 11 cols for 10th and 11th harmonic, 13 cols for 12th and 13th harmonic, 15 cols for 14th and 15th harmonic, 17 cols for 16th harmonic) with the number of 32-bit values per FOP column being defined by the “FOP\_ROW” input. The HPSEL\_EN enables the RAM output.

The DDRIN sub-module shall receive the following from the DDRIF2 module (see Figure 2‑1):-

* **WAIT\_REQUEST**: This is an indication from the DDRIF2 module that it is not ready to accept any read commands. The DDRIN sub-module shall react to WAIT\_REQUEST immediately as per the Avalon MM specifications.
* **DATA\_VALID**: This is an indication that the data from the DDR4 external SDRAM memory via the DDRIF2 module is valid and that the data can be stored in the selected page of the DDRIN\_STORE sub-module and also that the address to the DDR4 external SDRAM can be incremented.
* **DDR\_DATA[512\*ddr\_g -1:0]** : This is the data from the DDR4 external SDRAM memory via the DDRIF2 module. The width of the data bus is dependent on the number of DDR4 SDRAMs operating in parallel, with each DDR4 SDRAM providing 512-bit data @ CLK\_SYS rate.

### Address Generator: ADDR\_GEN Sub-Module

The ADDR\_GEN sub-module (see Figure 3‑2) shall generate the address to read the FOP columns from DDR4 external SDRAM and the address to store the FOP columns in the selected page of the DDRIN\_STORE sub-module.

A single cycle pulse of the “NEW\_COL” signal shall start the process of reading a set of up to 144 FOP columns from the DDR4 external SDRAM memory. The first column to read shall be defined by the “SEED\_COL[21:0]” value. The FOP columns to be read from DDR4 external SDRAM shall be calculated as shown in the table below:-

|  |  |  |  |
| --- | --- | --- | --- |
| Harmonic | Number of FOP Columns to read from DDR4 SDRAM | FOP Column values to read from DDR4 SDRAM | Calculation method |
| 1st | 1 | Col 1 = SEED\_COL[21:0] | Col 1 = SEED\_COL[21:0] |
| 2nd | 3 | Col 2 = (2 x SEED\_COL[21:0]) -1  Col 3 = (2 x SEED\_COL[21:0])  Col 4 = (2 x SEED\_COL[21:0]) +1 | Col 2 = Last Col + SEED\_COL[21:0] - 1  Col 3 = Last Col + 1  Col 4 = Last Col + 1 |
| 3rd | 3 | Col 5 = (3 x SEED\_COL[21:0]) -1  Col 6 = (3 x SEED\_COL[21:0])  Col 7 = (3 x SEED\_COL[21:0]) +1 | Col 5 = Last Col + SEED\_COL[21:0] - 2  Col 6 = Last Col + 1  Col 7 = Last Col + 1 |
| 4th | 5 | Col 8 = (4 x SEED\_COL[21:0]) -2  Col 9 = (4 x SEED\_COL[21:0]) -1  Col 10 = (4 x SEED\_COL[21:0])  Col 11 = (4 x SEED\_COL[21:0]) +1  Col 12 = (4 x SEED\_COL[21:0]) +2 | Col 8 = Last Col + SEED\_COL[21:0] - 3  Col 9 = Last Col + 1  Col 10 = Last Col + 1  Col 11 = Last Col + 1  Col 12 = Last Col + 1 |
| 5th | 5 | Col 13 = (5 x SEED\_COL[21:0]) -2  Col 14 = (5 x SEED\_COL[21:0]) -1  Col 15 = (5 x SEED\_COL[21:0])  Col 16 = (5 x SEED\_COL[21:0]) +1  Col 17 = (5 x SEED\_COL[21:0]) +2 | Col 13 = Last Col + SEED\_COL[21:0] - 4  Col 14 = Last Col + 1  Col 15 = Last Col + 1  Col 16 = Last Col + 1  Col 17 = Last Col + 1 |
| 6th | 7 | Col 18 = (6 x SEED\_COL[21:0]) -3  Col 19 = (6 x SEED\_COL[21:0]) -2  Col 20 = (6 x SEED\_COL[21:0]) -1  Col 21 = (6 x SEED\_COL[21:0])  Col 22 = (6 x SEED\_COL[21:0]) +1  Col 23 = (6 x SEED\_COL[21:0]) +2  Col 24 = (6 x SEED\_COL[21:0]) +3 | Col 18 = Last Col + SEED\_COL[21:0] - 5  Col 19 = Last Col + 1  Col 20 = Last Col + 1  Col 21 = Last Col + 1  Col 22 = Last Col + 1  Col 23 = Last Col + 1  Col 24 = Last Col + 1 |
| 7th | 7 | Col 25 = (7 x SEED\_COL[21:0]) -3  Col 26 = (7 x SEED\_COL[21:0]) -2  Col 27 = (7 x SEED\_COL[21:0]) -1  Col 28 = (7 x SEED\_COL[21:0])  Col 29 = (7 x SEED\_COL[21:0]) +1  Col 30 = (7 x SEED\_COL[21:0]) +2  Col 31 = (7 x SEED\_COL[21:0]) +3 | Col 25 = Last Col + SEED\_COL[21:0] - 6  Col 26 = Last Col + 1  Col 27 = Last Col + 1  Col 28 = Last Col + 1  Col 29 = Last Col + 1  Col 30 = Last Col + 1  Col 31 = Last Col + 1 |
| 8th | 9 | Col 32 = (8 x SEED\_COL[21:0]) -4  Col 33 = (8 x SEED\_COL[21:0]) -3  Col 34 = (8 x SEED\_COL[21:0]) -2  Col 35 = (8 x SEED\_COL[21:0]) -1  Col 36 = (8 x SEED\_COL[21:0])  Col 37 = (8 x SEED\_COL[21:0]) +1  Col 38 = (8 x SEED\_COL[21:0]) +2  Col 39 = (8 x SEED\_COL[21:0]) +3  Col 40 = (8 x SEED\_COL[21:0]) +4 | Col 32 = Last Col + SEED\_COL[21:0] - 7  Col 33 = Last Col + 1  Col 34 = Last Col + 1  Col 35 = Last Col + 1  Col 36 = Last Col + 1  Col 37 = Last Col + 1  Col 38 = Last Col + 1  Col 39 = Last Col + 1  Col 40 = Last Col + 1 |
| 9th | 9 | Col 41 = (9 x SEED\_COL[21:0]) -4  Col 42 = (9 x SEED\_COL[21:0]) -3  Col 43 = (9 x SEED\_COL[21:0]) -2  Col 44 = (9 x SEED\_COL[21:0]) -1  Col 45 = (9 x SEED\_COL[21:0])  Col 46 = (9 x SEED\_COL[21:0]) +1  Col 47 = (9 x SEED\_COL[21:0]) +2  Col 48 = (9 x SEED\_COL[21:0]) +3  Col 49 = (9 x SEED\_COL[21:0]) +4 | Col 41 = Last Col + SEED\_COL[21:0] – 8  Col 42 = Last Col + 1  Col 43 = Last Col + 1  Col 44 = Last Col + 1  Col 45 = Last Col + 1  Col 46 = Last Col + 1  Col 47 = Last Col + 1  Col 48 = Last Col + 1  Col 49 = Last Col + 1 |
| 10th | 11 | Col 50 = (10 x SEED\_COL[21:0]) -5  Col 51 = (10 x SEED\_COL[21:0]) -4  Col 52 = (10 x SEED\_COL[21:0]) -3  Col 53 = (10 x SEED\_COL[21:0]) -2  Col 54 = (10 x SEED\_COL[21:0]) -1  Col 55 = (10 x SEED\_COL[21:0])  Col 56 = (10 x SEED\_COL[21:0]) +1  Col 57 = (10 x SEED\_COL[21:0]) +2  Col 58 = (10 x SEED\_COL[21:0]) +3  Col 59 = (10 x SEED\_COL[21:0]) +4  Col 60 = (10 x SEED\_COL[21:0]) +5 | Col 50 = Last Col + SEED\_COL[21:0] – 9  Col 51 = Last Col + 1  Col 52 = Last Col + 1  Col 53 = Last Col + 1  Col 54 = Last Col + 1  Col 55 = Last Col + 1  Col 56 = Last Col + 1  Col 57 = Last Col + 1  Col 58 = Last Col + 1  Col 59 = Last Col + 1  Col 60 = Last Col + 1 |
| 11th | 11 | Col 61 = (11 x SEED\_COL[21:0]) -5  Col 62 = (11 x SEED\_COL[21:0]) -4  Col 63 = (11 x SEED\_COL[21:0]) -3  Col 64 = (11 x SEED\_COL[21:0]) -2  Col 65 = (11 x SEED\_COL[21:0]) -1  Col 66 = (11 x SEED\_COL[21:0])  Col 67 = (11 x SEED\_COL[21:0]) +1  Col 68 = (11 x SEED\_COL[21:0]) +2  Col 69 = (11 x SEED\_COL[21:0]) +3  Col 70 = (11 x SEED\_COL[21:0]) +4  Col 71 = (11 x SEED\_COL[21:0]) +5 | Col 61 = Last Col + SEED\_COL[21:0] – 10  Col 62 = Last Col + 1  Col 63 = Last Col + 1  Col 64 = Last Col + 1  Col 65 = Last Col + 1  Col 66 = Last Col + 1  Col 67 = Last Col + 1  Col 68 = Last Col + 1  Col 69 = Last Col + 1  Col 70 = Last Col + 1  Col 71 = Last Col + 1 |
| 12th | 13 | Col 72 = (12 x SEED\_COL[21:0]) -6  Col 73 = (12 x SEED\_COL[21:0]) -5  Col 74 = (12 x SEED\_COL[21:0]) -4  Col 75 = (12 x SEED\_COL[21:0]) -3  Col 76 = (12 x SEED\_COL[21:0]) -2  Col 77 = (12 x SEED\_COL[21:0]) -1  Col 78 = (12 x SEED\_COL[21:0])  Col 79 = (12 x SEED\_COL[21:0]) +1  Col 80 = (12 x SEED\_COL[21:0]) +2  Col 81 = (12 x SEED\_COL[21:0]) +3  Col 82 = (12 x SEED\_COL[21:0]) +4  Col 83 = (12 x SEED\_COL[21:0]) +5  Col 84 = (12 x SEED\_COL[21:0]) +6 | Col 72 = Last Col + SEED\_COL[21:0] – 11  Col 73 = Last Col + 1  Col 74 = Last Col + 1  Col 75 = Last Col + 1  Col 76 = Last Col + 1  Col 77 = Last Col + 1  Col 78 = Last Col + 1  Col 79 = Last Col + 1  Col 80 = Last Col + 1  Col 81 = Last Col + 1  Col 82 = Last Col + 1  Col 83 = Last Col + 1  Col 84 = Last Col + 1 |
| 13th | 13 | Col 85 = (13 x SEED\_COL[21:0]) -6  Col 86 = (13 x SEED\_COL[21:0]) -5  Col 87 = (13 x SEED\_COL[21:0]) -4  Col 88 = (13 x SEED\_COL[21:0]) -3  Col 89 = (13 x SEED\_COL[21:0]) -2  Col 90 = (13 x SEED\_COL[21:0]) -1  Col 91 = (13 x SEED\_COL[21:0])  Col 92 = (13 x SEED\_COL[21:0]) +1  Col 93 = (13 x SEED\_COL[21:0]) +2  Col 94 = (13 x SEED\_COL[21:0]) +3  Col 95 = (13 x SEED\_COL[21:0]) +4  Col 96 = (13 x SEED\_COL[21:0]) +5  Col 97 = (13 x SEED\_COL[21:0]) +6 | Col 85 = Last Col + SEED\_COL[21:0] – 12  Col 86 = Last Col + 1  Col 87 = Last Col + 1  Col 88 = Last Col + 1  Col 89 = Last Col + 1  Col 90 = Last Col + 1  Col 91 = Last Col + 1  Col 92 = Last Col + 1  Col 93 = Last Col + 1  Col 94 = Last Col + 1  Col 95 = Last Col + 1  Col 96 = Last Col + 1  Col 97 = Last Col + 1 |
| 14th | 15 | Col 98 = (14 x SEED\_COL[21:0]) -7  Col 99 = (14 x SEED\_COL[21:0]) -6  Col 100 = (14 x SEED\_COL[21:0]) -5  Col 101 = (14 x SEED\_COL[21:0]) -4  Col 102 = (14 x SEED\_COL[21:0]) -3  Col 103 = (14 x SEED\_COL[21:0]) -2  Col 104 = (14 x SEED\_COL[21:0]) -1  Col 105 = (14 x SEED\_COL[21:0])  Col 106 = (14 x SEED\_COL[21:0]) +1  Col 107 = (14 x SEED\_COL[21:0]) +2  Col 108 = (14 x SEED\_COL[21:0]) +3  Col 109 = (14 x SEED\_COL[21:0]) +4  Col 110 = (14 x SEED\_COL[21:0]) +5  Col 111 = (14 x SEED\_COL[21:0]) +6  Col 112 = (14 x SEED\_COL[21:0]) +7 | Col 98 = Last Col + SEED\_COL[21:0] – 13  Col 99 = Last Col + 1  Col 100 = Last Col + 1  Col 101 = Last Col + 1  Col 102 = Last Col + 1  Col 103 = Last Col + 1  Col 104 = Last Col + 1  Col 105 = Last Col + 1  Col 106 = Last Col + 1  Col 107 = Last Col + 1  Col 108 = Last Col + 1  Col 109 = Last Col + 1  Col 110 = Last Col + 1  Col 111 = Last Col + 1  Col 112 = Last Col + 1 |
| 15th | 15 | Col 113 = (15 x SEED\_COL[21:0]) -7  Col 114 = (15 x SEED\_COL[21:0]) -6  Col 115 = (15 x SEED\_COL[21:0]) -5  Col 116 = (15 x SEED\_COL[21:0]) -4  Col 117 = (15 x SEED\_COL[21:0]) -3  Col 118 = (15 x SEED\_COL[21:0]) -2  Col 119 = (15 x SEED\_COL[21:0]) -1  Col 120 = (15 x SEED\_COL[21:0])  Col 121 = (15 x SEED\_COL[21:0]) +1  Col 122 = (15 x SEED\_COL[21:0]) +2  Col 123 = (15 x SEED\_COL[21:0]) +3  Col 124 = (15 x SEED\_COL[21:0]) +4  Col 125 = (15 x SEED\_COL[21:0]) +5  Col 126 = (15 x SEED\_COL[21:0]) +6  Col 127 = (15 x SEED\_COL[21:0]) +7 | Col 113 = Last Col + SEED\_COL[21:0] – 14  Col 114 = Last Col + 1  Col 115 = Last Col + 1  Col 116 = Last Col + 1  Col 117 = Last Col + 1  Col 118 = Last Col + 1  Col 119 = Last Col + 1  Col 120 = Last Col + 1  Col 121 = Last Col + 1  Col 122 = Last Col + 1  Col 123 = Last Col + 1  Col 124 = Last Col + 1  Col 125 = Last Col + 1  Col 126 = Last Col + 1  Col 127 = Last Col + 1 |
| 16th | 17 | Col 128 = (16 x SEED\_COL[21:0]) -8  Col 129 = (16 x SEED\_COL[21:0]) -7  Col 130 = (16 x SEED\_COL[21:0]) -6  Col 131 = (16 x SEED\_COL[21:0]) -5  Col 132 = (16 x SEED\_COL[21:0]) -4  Col 133 = (16 x SEED\_COL[21:0]) -3  Col 134 = (16 x SEED\_COL[21:0]) -2  Col 135 = (16 x SEED\_COL[21:0]) -1  Col 136 = (16 x SEED\_COL[21:0])  Col 137 = (16 x SEED\_COL[21:0]) +1  Col 138 = (16 x SEED\_COL[21:0]) +2  Col 139 = (16 x SEED\_COL[21:0]) +3  Col 140 = (16 x SEED\_COL[21:0]) +4  Col 141 = (16 x SEED\_COL[21:0]) +5  Col 142 = (16 x SEED\_COL[21:0]) +6  Col 143 = (16 x SEED\_COL[21:0]) +7  Col 144 = (16 x SEED\_COL[21:0]) +8 | Col 128 = Last Col + SEED\_COL[21:0] – 15  Col 129 = Last Col + 1  Col 130 = Last Col + 1  Col 131 = Last Col + 1  Col 132 = Last Col + 1  Col 133 = Last Col + 1  Col 134 = Last Col + 1  Col 135 = Last Col + 1  Col 136 = Last Col + 1  Col 137 = Last Col + 1  Col 138 = Last Col + 1  Col 139 = Last Col + 1  Col 140 = Last Col + 1  Col 141 = Last Col + 1  Col 142 = Last Col + 1  Col 143 = Last Col + 1  Col 144 = Last Col + 1 |

Table 3‑1 : FOP Columns to be Read for Harmonic Summing

The actual number of FOP columns that need to be read from DDR4 external SDRAM shall be indicated by the H[3:0] configuration for the analysis run. If H[3:0] = “1111” (and harmonic\_g = 16 to support the storage of all the data) all 16 harmonics are to be summed thus requiring all 144 columns listed in the table above to be read from DDR4 external SDRAM memory. If, for example, H[3:0] = “0011” only the first four harmonics are to be summed thus requiring the first 12 columns listed in the table above to be read from DDR4 external SDRAM memory. Once all the required addresses have been requested from the DDR4 interface, ADDR\_GEN shall pulse the output “DONE\_REQ” for a single cycle.

The above column numbers shall be translated to DDR4 external SDRAM memory addresses.

The relationship between FOP columns and the DDR4 external SDRAM memory is shown below for different values of the “ddr\_g” generic. Note that DDR\_ADDR[5:0] is always zero, since the data from the DDR4 SDRAM memory is 512-bit (64-byte) internally in the FPGA.

Note:-

* The FOP column number to be read shall have FOP\_COL\_OFFSET[8:0] added to it prior to conversion to the DDR\_ADDR.
* The HSUM\_PAGE[31:0] must be added to the DDR\_ADDR in the tables below to obtain the actual DDR\_ADDR to the DDR4 external SDRAM memory.
* The number of DDR4 SDRAM locations to read are dependent on H[3:0] since it is not necessary to read FOP data that will not be used.
* The number of rows of the FOP that exist is set by FOP\_ROW[6:0] which affects the number of DDR4 SDRAM locations to read (quantised on the 512-bit wide data)
* The value for P[0] is repeated twice, just to allow for a consistent arrangement for the positive and negative filters (i.e. p[1] and p[-1] etc.) in the remainder of the data read from the DDR4 SDRAM. They are populated by the CONV module in this way, so either location for p[0] may be used.
* ddr\_g = 1

DDR4 SDRAM #1

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DDR\_ADDR  [31:6]  (add HSUM\_PAGE  To obtain the actual address) | FOP  Col | FOP: PWR for filter p[\*] Mapped to the DDR \_DATA[512:0] Interface  Each p[\*] value pertains to a row of the FOP  Bit 511 Bit 0 | | | | | | | | | | | | | | | |
| 0x0000000 | 1 | p[-7] | p[7] | p[-6] | p[6] | p[-5] | p[5] | p[-4] | p[4] | p[-3] | p[3] | p[-2] | p[2] | p[-1] | p[1] | p[0] | p[0] |
| 0x0000001 | p[-14] | p[14] | p[-13] | p[13] | p[-12] | p[12] | p[-11] | p[11] | p[-10] | p[10] | p[-9] | p[9] | p[-8] | p[8] |  |  |
| 0x0000002 | p[-21] | p[21] | p[-20] | p[20] | p[-19] | p[19] | p[-18] | p[18] | p[-17] | p[17] | p[-16] | p[16] | p[-15] | p[15] |  |  |
| 0x0000003 | p[-28] | p[28] | p[-27] | p[27] | p[-26] | p[26] | p[-25] | p[25] | p[-24] | p[24] | p[-23] | p[23] | p[-22] | p[22] |  |  |
| 0x0000004 | p[-35] | p[35] | p[-34] | p[34] | p[-33] | p[33] | p[-32] | p[32] | p[-31] | p[31] | p[-30] | p[30] | p[-29] | p[29] |  |  |
| 0x0000005 | p[-42] | p[42] | p[-41] | p[41] | p[-40] | p[40] | p[-39] | p[39] | p[-38] | p[38] | p[-37] | p[37] | p[-36] | p[36] |  |  |
| 0x0000008 | 2 | p[-7] | p[7] | p[-6] | p[6] | p[-5] | p[5] | p[-4] | p[4] | p[-3] | p[3] | p[-2] | p[2] | p[-1] | p[1] | p[0] | p[0] |
| 0x0000009 | p[-14] | p[14] | p[-13] | p[13] | p[-12] | p[12] | p[-11] | p[11] | p[-10] | p[10] | p[-9] | p[9] | p[-8] | p[8] |  |  |
| 0x000000A | p[-21] | p[21] | p[-20] | p[20] | p[-19] | p[19] | p[-18] | p[18] | p[-17] | p[17] | p[-16] | p[16] | p[-15] | p[15] |  |  |
| 0x000000B | p[-28] | p[28] | p[-27] | p[27] | p[-26] | p[26] | p[-25] | p[25] | p[-24] | p[24] | p[-23] | p[23] | p[-22] | p[22] |  |  |
| 0x000000C | p[-35] | p[35] | p[-34] | p[34] | p[-33] | p[33] | p[-32] | p[32] | p[-31] | p[31] | p[-30] | p[30] | p[-29] | p[29] |  |  |
| 0x000000D | p[-42] | p[42] | p[-41] | p[41] | p[-40] | p[40] | p[-39] | p[39] | p[-38] | p[38] | p[-37] | p[37] | p[-36] | p[36] |  |  |
| : |  | : | : | : | : | : | : | : | : | : | : | : | : | : | : | : | : |
| 0x1FFFFF8 | 222 | p[-7] | p[7] | p[-6] | p[6] | p[-5] | p[5] | p[-4] | p[4] | p[-3] | p[3] | p[-2] | p[2] | p[-1] | p[1] | p[0] | p[0] |
| 0x1FFFFF9 | p[-14] | p[14] | p[-13] | p[13] | p[-12] | p[12] | p[-11] | p[11] | p[-10] | p[10] | p[-9] | p[9] | p[-8] | p[8] |  |  |
| 0x1FFFFFA | p[-21] | p[21] | p[-20] | p[20] | p[-19] | p[19] | p[-18] | p[18] | p[-17] | p[17] | p[-16] | p[16] | p[-15] | p[15] |  |  |
| 0x1FFFFFB | p[-28] | p[28] | p[-27] | p[27] | p[-26] | p[26] | p[-25] | p[25] | p[-24] | p[24] | p[-23] | p[23] | p[-22] | p[22] |  |  |
| 0x1FFFFFC | p[-35] | p[35] | p[-34] | p[34] | p[-33] | p[33] | p[-32] | p[32] | p[-31] | p[31] | p[-30] | p[30] | p[-29] | p[29] |  |  |
| 0x1FFFFFD | p[-42] | p[42] | p[-41] | p[41] | p[-40] | p[40] | p[-39] | p[39] | p[-38] | p[38] | p[-37] | p[37] | p[-36] | p[36] |  |  |

Table 3‑2: FOP Arrangement from DDR4 SDRAM when ddr\_g =1

* ddr\_g = 2

DDR4 SDRAM #1

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DDR\_ADDR  [31:6]  (add HSUM\_PAGE  To obtain the actual address) | FOP  Col | FOP: PWR for filter p[\*] Mapped to the DDR \_DATA[512:0] Interface  Each p[\*] value pertains to a row of the FOP  Bit 511 Bit 0 | | | | | | | | | | | | | | | |
| 0x0000000 | 1 | p[-7] | p[7] | p[-6] | p[6] | p[-5] | p[5] | p[-4] | p[4] | p[-3] | p[3] | p[-2] | p[2] | p[-1] | p[1] | p[0] | p[0] |
| 0x0000001 | p[-21] | p[21] | p[-20] | p[20] | p[-19] | p[19] | p[-18] | p[18] | p[-17] | p[17] | p[-16] | p[16] | p[-15] | p[15] |  |  |
| 0x0000002 | p[-35] | p[35] | p[-34] | p[34] | p[-33] | p[33] | p[-32] | p[32] | p[-31] | p[31] | p[-30] | p[30] | p[-29] | p[29] |  |  |
| 0x0000004 | 2 | p[-7] | p[7] | p[-6] | p[6] | p[-5] | p[5] | p[-4] | p[4] | p[-3] | p[3] | p[-2] | p[2] | p[-1] | p[1] | p[0] | p[0] |
| 0x0000005 | p[-21] | p[21] | p[-20] | p[20] | p[-19] | p[19] | p[-18] | p[18] | p[-17] | p[17] | p[-16] | p[16] | p[-15] | p[15] |  |  |
| 0x0000006 | p[-35] | p[35] | p[-34] | p[34] | p[-33] | p[33] | p[-32] | p[32] | p[-31] | p[31] | p[-30] | p[30] | p[-29] | p[29] |  |  |
| : |  | : | : | : | : | : | : | : | : | : | : | : | : | : | : | : | : |
| 0x0FFFFFC | 222 | p[-7] | p[7] | p[-6] | p[6] | p[-5] | p[5] | p[-4] | p[4] | p[-3] | p[3] | p[-2] | p[2] | p[-1] | p[1] | p[0] | p[0] |
| 0x0FFFFFD | p[-21] | p[21] | p[-20] | p[20] | p[-19] | p[19] | p[-18] | p[18] | p[-17] | p[17] | p[-16] | p[16] | p[-15] | p[15] |  |  |
| 0x0FFFFFE | p[-35] | p[35] | p[-34] | p[34] | p[-33] | p[33] | p[-32] | p[32] | p[-31] | p[31] | p[-30] | p[30] | p[-29] | p[29] |  |  |

DDR4 SDRAM #2

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DDR\_ADDR  [31:6]  (add HSUM\_PAGE  To obtain the actual address) | FOP  Col | FOP: PWR for filter p[\*] Mapped to the DDR \_DATA[512:0] Interface  Each p[\*] value pertains to a row of the FOP  Bit 511 Bit 0 | | | | | | | | | | | | | | | |
| 0x0000000 | 1 | p[-14] | p[14] | p[-13] | p[13] | p[-12] | p[12] | p[-11] | p[11] | p[-10] | p[10] | p[-9] | p[9] | p[-8] | p[8] |  |  |
| 0x0000001 | p[-28] | p[28] | p[-27] | p[27] | p[-26] | p[26] | p[-25] | p[25] | p[-24] | p[24] | p[-23] | p[23] | p[-22] | p[22] |  |  |
| 0x0000002 | p[-42] | p[42] | p[-41] | p[41] | p[-40] | p[40] | p[-39] | p[39] | p[-38] | p[38] | p[-37] | p[37] | p[-36] | p[36] |  |  |
| 0x0000004 | 2 | p[-14] | p[14] | p[-13] | p[13] | p[-12] | p[12] | p[-11] | p[11] | p[-10] | p[10] | p[-9] | p[9] | p[-8] | p[8] |  |  |
| 0x0000005 | p[-28] | p[28] | p[-27] | p[27] | p[-26] | p[26] | p[-25] | p[25] | p[-24] | p[24] | p[-23] | p[23] | p[-22] | p[22] |  |  |
| 0x0000006 | p[-42] | p[42] | p[-41] | p[41] | p[-40] | p[40] | p[-39] | p[39] | p[-38] | p[38] | p[-37] | p[37] | p[-36] | p[36] |  |  |
| : |  | : | : | : | : | : | : | : | : | : | : | : | : | : | : | : | : |
| 0x0FFFFFC | 222 | p[-14] | p[14] | p[-13] | p[13] | p[-12] | p[12] | p[-11] | p[11] | p[-10] | p[10] | p[-9] | p[9] | p[-8] | p[8] |  |  |
| 0x0FFFFFD | p[-28] | p[28] | p[-27] | p[27] | p[-26] | p[26] | p[-25] | p[25] | p[-24] | p[24] | p[-23] | p[23] | p[-22] | p[22] |  |  |
| 0x0FFFFFE | p[-42] | p[42] | p[-41] | p[41] | p[-40] | p[40] | p[-39] | p[39] | p[-38] | p[38] | p[-37] | p[37] | p[-36] | p[36] |  |  |

Table 3‑3: FOP Arrangement from DDR4 SDRAM when ddr\_g =2

* ddr\_g = 3

DDR4 SDRAM #1

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DDR\_ADDR  [31:6]  (add HSUM\_PAGE  To obtain the actual address) | FOP  Col | FOP: PWR for filter p[\*] Mapped to the DDR \_DATA[512:0] Interface  Each p[\*] value pertains to a row of the FOP  Bit 511 Bit 0 | | | | | | | | | | | | | | | |
| 0x0000000 | 1 | p[-7] | p[7] | p[-6] | p[6] | p[-5] | p[5] | p[-4] | p[4] | p[-3] | p[3] | p[-2] | p[2] | p[-1] | p[1] | p[0] | p[0] |
| 0x0000001 | p[-28] | p[28] | p[-27] | p[27] | p[-26] | p[26] | p[-25] | p[25] | p[-24] | p[24] | p[-23] | p[23] | p[-22] | p[22] |  |  |
| 0x0000002 | 2 | p[-7] | p[7] | p[-6] | p[6] | p[-5] | p[5] | p[-4] | p[4] | p[-3] | p[3] | p[-2] | p[2] | p[-1] | p[1] | p[0] | p[0] |
| 0x0000003 | p[-28] | p[28] | p[-27] | p[27] | p[-26] | p[26] | p[-25] | p[25] | p[-24] | p[24] | p[-23] | p[23] | p[-22] | p[22] |  |  |
| : |  | : | : | : | : | : | : | : | : | : | : | : | : | : | : | : | : |
| 0x07FFFFE | 222 | p[-7] | p[7] | p[-6] | p[6] | p[-5] | p[5] | p[-4] | p[4] | p[-3] | p[3] | p[-2] | p[2] | p[-1] | p[1] | p[0] | p[0] |
| 0x07FFFFF | p[-28] | p[28] | p[-27] | p[27] | p[-26] | p[26] | p[-25] | p[25] | p[-24] | p[24] | p[-23] | p[23] | p[-22] | p[22] |  |  |

DDR4 SDRAM #2

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DDR\_ADDR  [31:6]  (add HSUM\_PAGE  To obtain the actual address) | FOP  Col | FOP: PWR for filter p[\*] Mapped to the DDR \_DATA[512:0] Interface  Each p[\*] value pertains to a row of the FOP  Bit 511 Bit 0 | | | | | | | | | | | | | | | |
| 0x0000000 | 1 | p[-14] | p[14] | p[-13] | p[13] | p[-12] | p[12] | p[-11] | p[11] | p[-10] | p[10] | p[-9] | p[9] | p[-8] | p[8] |  |  |
| 0x0000001 | p[-35] | p[35] | p[-34] | p[34] | p[-33] | p[33] | p[-32] | p[32] | p[-31] | p[31] | p[-30] | p[30] | p[-29] | p[29] |  |  |
| 0x0000002 | 2 | p[-14] | p[14] | p[-13] | p[13] | p[-12] | p[12] | p[-11] | p[11] | p[-10] | p[10] | p[-9] | p[9] | p[-8] | p[8] |  |  |
| 0x0000003 | p[-35] | p[35] | p[-34] | p[34] | p[-33] | p[33] | p[-32] | p[32] | p[-31] | p[31] | p[-30] | p[30] | p[-29] | p[29] |  |  |
| : |  | : | : | : | : | : | : | : | : | : | : | : | : | : | : | : | : |
| 0x07FFFFE | 222 | p[-14] | p[14] | p[-13] | p[13] | p[-12] | p[12] | p[-11] | p[11] | p[-10] | p[10] | p[-9] | p[9] | p[-8] | p[8] |  |  |
| 0x07FFFFF | p[-35] | p[35] | p[-34] | p[34] | p[-33] | p[33] | p[-32] | p[32] | p[-31] | p[31] | p[-30] | p[30] | p[-29] | p[29] |  |  |

DDR4 SDRAM #3

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DDR\_ADDR  [31:6]  (add HSUM\_PAGE  To obtain the actual address) | FOP  Col | FOP: PWR for filter p[\*] Mapped to the DDR \_DATA[512:0] Interface  Each p[\*] value pertains to a row of the FOP  Bit 511 Bit 0 | | | | | | | | | | | | | | | |
| 0x0000000 | 1 | p[-21] | p[21] | p[-20] | p[20] | p[-19] | p[19] | p[-18] | p[18] | p[-17] | p[17] | p[-16] | p[16] | p[-15] | p[15] |  |  |
| 0x0000001 | p[-42] | p[42] | p[-41] | p[41] | p[-40] | p[40] | p[-39] | p[39] | p[38] | p[38] | p[-37] | p[37] | p[-36] | p[36] |  |  |
| 0x0000002 | 2 | p[-21] | p[21] | p[-20] | p[20] | p[-19] | p[19] | p[-18] | p[18] | p[-17] | p[17] | p[-16] | p[16] | p[-15] | p[15] |  |  |
| 0x0000003 | p[-42] | p[42] | p[-41] | p[41] | p[-40] | p[40] | p[-39] | p[39] | p[-38] | p[38] | p[-37] | p[37] | p[-36] | p[36] |  |  |
| : |  | : | : | : | : | : | : | : | : | : | : | : | : | : | : | : | : |
| 0x07FFFFE | 222 | p[-21] | p[21] | p[-20] | p[20] | p[-19] | p[19] | p[-18] | p[18] | p[-17] | p[17] | p[-16] | p[16] | p[-15] | p[15] |  |  |
| 0x07FFFFF | p[-42] | p[42] | p[-41] | p[41] | p[-40] | p[40] | p[-39] | p[39] | p[-38] | p[38] | p[-37] | p[37] | p[-36] | p[36] |  |  |

Table 3‑4 : FOP Arrangement from DDR4 SDRAM when ddr\_g =3

As the data arrives from the external DDR4 SDRAM memory it shall be stored in the DDRIN\_STORE sub-module page selected by the “WR\_PAGE[1:0]” signal, i.e. there shall be four pages. The arrival of a 512-bit data word from the DDR4 external SDRAM is indicated by assertion of the “DATA\_VALID” signal. This shall be used by the ADDR\_GEN sub-module to increment the “STORE\_ADDR[2:0]” signal and used by the DDRIN\_STORE sub-module as a write strobe. The STORE\_ADDR[2:0] sequence is dependent on the generic “ddr\_g”:-

* ddr\_g =1: STORE\_ADDR[2:0] sequence 0,1,2,3,4,5,0…
* ddr\_g =2: STORE\_ADDR[2:0] sequence 0,1,2,0, …
* ddr\_g =3: STORE\_ADDR[2:0] sequence 0,1,0, …

The DDRIN\_STORE sub-module consists of 144-off independent RAMs (duplicated by the number of SUMMER sub-module instances), such that each FOP column is stored in a separate RAM. The “WREN[1 to 144]” signal generated by the ADDR\_GEN sub-module is the write enable to each RAM within the DDRIN\_STORE sub-module.

When all the required FOP Columns (i.e. up to 144 as indicated by H[3:0]) have been read from the DDR4 external SDRAM memory and stored in the DDRIN\_STORE sub-module the ADDR\_GEN sub-module shall pulse the “DONE\_READ” signal to the TGEN sub-module for a single cycle and shall increment “WR\_PAGE[1:0]”.

### DDR Input Data Store: DDRIN\_STORE Sub-Module

The DDRIN\_STORE sub-module (see Figure 3‑2) shall store the up to 144 FOP columns for harmonic summing, - dependent on the generic “harmonic\_g”

For each SUMMER sub-module instance the DDRIN\_STORE sub-module shall provide up to 144-off RAMs (“harmonic\_g” dependent), each capable of storing the data for a column of the FOP. The “WREN[1 to No of RAMs]” signal shall identify which of the up to 144-off RAMs (“harmonic\_g” dependent) shall be used to store the data, and the WR\_PAGE[1:0] and STORE\_ADDR[2:0] signals provide the address to the RAM. In the case of more than one SUMMER sub-module instance (i.e. summer\_g > 1), the same data from the DDR4 external SDRAMs is stored in each set of up to 144-off RAMs (“harmonic\_g” dependent). This is needed since each SUMMER sub-module shall independently address its set of RAMs within the DDRIN\_STORE sub-module.

Each RAM within the DDRIN\_STORE sub-module shall be divided into four pages. Data from the DDR4 external SDRAM memory shall be loaded into one page whilst the SUMMER sub-module(s) are accessing another page. Having more than two pages gives an improvement in the DDR efficiency as it reduces the impact of the initial latency of a DDR read access. The DDRIN\_RD\_PAGE[1:0] signal from the TGEN sub-module determines the page to read to supply data to the SUMMER sub-module. At the start of processing a new FOP, both WR\_PAGE[1:0] and DDRIN\_RD\_PAGE[1:0] shall be reset to 0, but during an analysis run they shall simply increment as a page is filled or consumed respectively.

The read side of each RAM within the DDRIN\_STORE sub-module shall be addressed by a 7-bit address and shall provide 32-bit data (i.e. one of the rows within the FOP column).

The arrangement of the 32-bit values for each FOP row for a FOP column in a RAM within the DDRIN\_STORE sub-module is shown below (take note of the gaps in the HPSEL selection values):-

| FOP Row selection value HPSEL[6:0] from the SUMMER sub-module which becomes the Address to the associated RAM in the DDRIN\_STORE sub-module | FOP Row  (Rows 1 to 85) | FOP Orbital Acceleration Filter | DDR\_DATA  ddr\_g = 1  512-bit word read from SDRAM and data bits of the word | DDR\_DATA  ddr\_g = 2  1024-bit word read from SDRAMs and data bits of the word | DDR\_DATA  ddr\_g = 3  1536-bit word read from SDRAMs and data bits of the word |
| --- | --- | --- | --- | --- | --- |
| 0x00 | 43 (central) | p[0] | word 1 [31:0] | word 1 [31:0] | word 1 [31:0] |
| 0x01 | 43 (central) | p[0] | word 1 [63:32] | word 1 [63:32] | word 1 [63:32] |
| 0x02 | 44 | p[1] | word 1 [95:64] | word 1 [95:64] | word 1 [95:64] |
| 0x03 | 42 | p[-1] | word 1 [127:96] | word 1 [127:96] | word 1 [127:96] |
| 0x04 | 45 | p[2] | word 1 [159:128] | word 1 [159:128] | word 1 [159:128] |
| 0x05 | 41 | p[-2] | word 1 [191:160] | word 1 [191:160] | word 1 [191:160] |
| 0x06 | 46 | p[3] | word 1 [223:192] | word 1 [223:192] | word 1 [223:192] |
| 0x07 | 40 | p[-3] | word 1 [255:224] | word 1 [255:224] | word 1 [255:224] |
| 0x08 | 47 | p[4] | word 1 [287:256] | word 1 [287:256] | word 1 [287:256] |
| 0x09 | 39 | p[-4] | word 1 [319:288] | word 1 [319:288] | word 1 [319:288] |
| 0x0A | 48 | p[5] | word 1 [351:320] | word 1 [351:320] | word 1 [351:320] |
| 0x0B | 38 | p[-5] | word 1 [383:352] | word 1 [383:352] | word 1 [383:352] |
| 0x0C | 49 | p[6] | word 1 [415:384] | word 1 [415:384] | word 1 [415:384] |
| 0x0D | 37 | p[-6] | word 1 [447:416] | word 1 [447:416] | word 1 [447:416] |
| 0x0E | 50 | p[7] | word 1 [479:448] | word 1 [479:448] | word 1 [479:448] |
| 0x0F | 36 | p[-7] | word 1 [511:480] | word 1 [511:480] | word 1 [511:480] |
| 0x12 | 51 | p[8] | word 2 [95:64] | word 1 [607:576] | word 1 [607:576] |
| 0x13 | 35 | p[-8] | word 2 [127:96] | word 1 [639:608] | word 1 [639:608] |
| 0x14 | 52 | p[9] | word 2 [159:128] | word 1 [671:640] | word 1 [671:640] |
| 0x15 | 34 | p[-9] | word 2 [191:160] | word 1 [703:672] | word 1 [703:672] |
| 0x16 | 53 | p[10] | word 2 [223:192] | word 1 [735:704] | word 1 [735:704] |
| 0x17 | 33 | p[-10] | word 2 [255:224] | word 1 [767:736] | word 1 [767:736] |
| 0x18 | 54 | p[11] | word 2 [287:256] | word 1 [799:768] | word 1 [799:768] |
| 0x19 | 32 | p[-11] | word 2 [319:288] | word 1 [831:800] | word 1 [831:800] |
| 0x1A | 55 | p[12] | word 2 [351:320] | word 1 [863:832] | word 1 [863:832] |
| 0x1B | 31 | p[-12] | word 2 [383:352] | word 1 [895:864] | word 1 [895:864] |
| 0x1C | 56 | p[13] | word 2 [415:384] | word 1 [927:896] | word 1 [927:896] |
| 0x1D | 30 | p[-13] | word 2 [447:416] | word 1 [959:928] | word 1 [959:928] |
| 0x1E | 57 | p[14] | word 2 [479:448] | word 1 [991:960] | word 1 [991:960] |
| 0x1F | 29 | p[-14] | word 2 [511**:**480] | word 1 [1023:992] | word 1 [1023:992] |
| 0x22 | 58 | p[15] | word 3 [95:64] | word 2 [95:64] | word 1 [1119:1088] |
| 0x23 | 28 | p[-15] | word 3 [127:96] | word 2 [127:96] | word 1 [1151:1120] |
| 0x24 | 59 | p[16] | word 3 [159:128] | word 2 [159:128] | word 1 [1183:1152] |
| 0x25 | 27 | p[-16] | word 3 [191:160] | word 2 [191:160] | word 1 [1215:1184] |
| 0x26 | 60 | p[17] | word 3 [223:192] | word 2 [223:192] | word 1 [1247:1216] |
| 0x27 | 26 | p[-17] | word 3 [255:224] | word 2 [255:224] | word 1 [1279:1248] |
| 0x28 | 61 | p[18] | word 3 [287:256] | word 2 [287:256] | word 1 [1311:1280] |
| 0x29 | 25 | p[-18] | word 3 [319:288] | word 2 [319:288] | word 1 [1343:1312] |
| 0x2A | 62 | p[19] | word 3 [351:320] | word 2 [351:320] | word 1 [1375:1344] |
| 0x2B | 24 | p[-19] | word 3 [383:352] | word 2 [383:352] | word 1 [1407:1376] |
| 0x2C | 63 | p[20] | word 3 [415:384] | word 2 [415:384] | word 1 [1439:1408] |
| 0x2D | 23 | p[-20] | word 3 [447:416] | word 2 [447:416] | word 1 [1471:1440] |
| 0x2E | 64 | p[21] | word 3 [479:448] | word 2 [479:448] | word 1 [1503:1472] |
| 0x2F | 22 | p[-21] | word 3 [511**:**480] | word 2 [512:480] | word 1 [1535:1504] |
| 0x32 | 65 | p[22] | word 4 [95:64] | word 2 [607:576] | word 2 [95:64] |
| 0x33 | 21 | p[-22] | word 4 [127:96] | word 2 [639:608] | word 2 [127:96] |
| 0x34 | 66 | p[23] | word 4 [159:128] | word 2 [671:640] | word 2 [159:128] |
| 0x35 | 20 | p[-23] | word 4 [191:160] | word 2 [703:672] | word 2 [191:160] |
| 0x36 | 67 | p[24] | word 4 [223:192] | word 2 [735:704] | word 2 [223:192] |
| 0x37 | 19 | p[-24] | word 4 [255:224] | word 2 [767:736] | word 2 [255:224] |
| 0x38 | 68 | p[25] | word 4 [287:256] | word 2 [799:768] | word 2 [287:256] |
| 0x39 | 18 | p[-25] | word 4 [319:288] | word 2 [831:800] | word 2 [319:288] |
| 0x3A | 69 | p[26] | word 4 [351:320] | word 2 [863:832] | word 2 [351:320] |
| 0x3B | 17 | p[-26] | word 4 [383:352] | word 2 [895:864] | word 2 [383:352] |
| 0x3C | 70 | p[27] | word 4 [415:384] | word 2 [927:896] | word 2 [415:384] |
| 0x3D | 16 | p[-27] | word 4 [447:416] | word 2 [959:928] | word 2 [447:416] |
| 0x3E | 71 | p[28] | word 4 [479:448] | word 2 [991:960] | word 2 [479:448] |
| 0x3F | 15 | p[-28] | word 4 [511**:**480] | word 2 [1023:992] | word 2 [511:480] |
| 0x42 | 72 | p[29] | word 5 [95:64] | word 3 [95:64] | word 2 [607:576] |
| 0x43 | 14 | p[-29] | word 5 [127:96] | word 3 [127:96] | word 2 [639:608] |
| 0x44 | 73 | p[30] | word 5 [159:128] | word 3 [159:128] | word 2 [671:640] |
| 0x45 | 13 | p[-30] | word 5 [191:160] | word 3 [191:160] | word 2 [703:672] |
| 0x46 | 74 | p[31] | word 5 [223:192] | word 3 [223:192] | word 2 [735:704] |
| 0x47 | 12 | p[-31] | word 5 [255:224] | word 3 [255:224] | word 2 [767:736] |
| 0x48 | 75 | p[32] | word 5 [287:256] | word 3 [287:256] | word 2 [799:768] |
| 0x49 | 11 | p[-32] | word 5 [319:288] | word 3 [319:288] | word 2 [831:800] |
| 0x4A | 76 | p[33] | word 5 [351:320] | word 3 [351:320] | word 2 [863:832] |
| 0x4B | 10 | p[-33] | word 5 [383:352] | word 3 [383:352] | word 2 [895:864] |
| 0x4C | 77 | p[34] | word 5 [415:384] | word 3 [415:384] | word 2 [927:896] |
| 0x4D | 9 | p[-34] | word 5 [447:416] | word 3 [447:416] | word 2 [959:928] |
| 0x4E | 78 | p[35] | word 5 [479:448] | word 3 [479:448] | word 2 [991:960] |
| 0x4F | 8 | p[-35] | word 5 [511**:**480] | word 3 [512:480] | word 2 [1023:992] |
| 0x52 | 79 | p[36] | word 6 [95:64] | word 3 [607:576] | word 2 [1119:1088] |
| 0x53 | 7 | p[-36] | word 6 [127:96] | word 3 [639:608] | word 2 [1151:1120] |
| 0x54 | 80 | p[37] | word 6 [159:128] | word 3 [671:640] | word 2 [1183:1152] |
| 0x55 | 6 | p[-37] | word 6 [191:160] | word 3 [703:672] | word 2 [1215:1184] |
| 0x56 | 81 | p[38] | word 6 [223:192] | word 3 [735:704] | word 2 [1247:1216] |
| 0x57 | 5 | p[-38] | word 6 [255:224] | word 3 [767:736] | word 2 [1279:1248] |
| 0x58 | 82 | p[39] | word 6 [287:256] | word 3 [799:768] | word 2 [1311:1280] |
| 0x59 | 4 | p[-39] | word 6 [319:288] | word 3 [831:800] | word 2 [1343:1312] |
| 0x5A | 83 | p[40] | word 6 [351:320] | word 3 [863:832] | word 2 [1375:1344] |
| 0x5B | 3 | p[-40] | word 6 [383:352] | word 3 [895:864] | word 2 [1407:1376] |
| 0x5C | 84 | p[41] | word 6 [415:384] | word 3 [927:896] | word 2 [1439:1408] |
| 0x5D | 2 | p[-41] | word 6 [447:416] | word 3 [959:928] | word 2 [1471:1440] |
| 0x5E | 85 | p[42] | word 6 [479:448] | word 3 [991:960] | word 2 [1503:1472] |
| 0x5F | 1 | p[-42] | word 6 [511**:**480] | word 3 [1023:992] | word 2 [1535:1504] |

Table 3‑5 : HPSEL to FOP Row Translation

Each SUMMER sub-module (Figure 3‑3) shall provide a total of “harmonic\_g”-off read addresses “HPSEL[6:0]” to its set of RAMs within the DDRIN\_STORE sub-module. Each address shall be related to a harmonic and shall simultaneously address the RAMs within the DDRIN\_STORE sub-module containing the FOP column numbers relevant to that harmonic as detailed in the table below :-

|  |  |  |  |
| --- | --- | --- | --- |
| HPSEL[n][6:0] vector per SUMMER sub-module instance  (see Figure 3‑2) | Associated Harmonic | RAMs numbers (FOP Cols) within DDRIN\_STORE Simultaneously Addressed  (see Figure 3‑2) | DATA\_SUM bits  Expressed as bits per harmonic and also bits within the aggregate data bus.  (see Figure 3‑2) |
| [1] | 1st (fundamental) | 1 | DATA\_SUM[1][31:0] = DATA\_SUM[31:0] |
| [2] | 2nd | 2, 3, 4 | DATA\_SUM[2][95:0] = DATA\_SUM[127:32] |
| [3] | 3rd | 5, 6, 7 | DATA\_SUM[3][95:0] = DATA\_SUM[223:128] |
| [4] | 4th | 8, 9, 10, 11, 12 | DATA\_SUM[4][159:0] = DATA\_SUM[383:224] |
| [5] | 5th | 13, 14, 15, 16, 17 | DATA\_SUM[5][159:0] = DATA\_SUM[543:384] |
| [6] | 6th | 18, 19, 20, 21, 22, 23, 24 | DATA\_SUM[6][223:0] = DATA\_SUM[767:544] |
| [7] | 7th | 25, 26, 27, 28, 29, 30, 31 | DATA\_SUM[7][223:0] = DATA\_SUM[991:768] |
| [8] | 8th | 32, 33, 34, 35, 36, 37, 38, 39, 40 | DATA\_SUM[8][287:0] = DATA\_SUM[1279:992] |
| [9] | 9th | 41, 42, 43, 44, 45, 46, 47, 48, 49 | DATA\_SUM[9][287:0] = DATA\_SUM[1567:1280] |
| [10] | 10th | 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60 | DATA\_SUM[10][351:0] = DATA\_SUM[1919:1568] |
| [11] | 11th | 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71 | DATA\_SUM[11][351:0] = DATA\_SUM[2271:1920] |
| [12] | 12th | 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84 | DATA\_SUM[12][415:0] = DATA\_SUM[2687:2272] |
| [13] | 13th | 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97 | DATA\_SUM[13][415:0] = DATA\_SUM[3103:2688] |
| [14] | 14th | 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112 | DATA\_SUM[14][479:0] = DATA\_SUM[3583:3104] |
| [15] | 15th | 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127 | DATA\_SUM[15][479:0] = DATA\_SUM[4063:3584] |
| [16] | 16th | 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144 | DATA\_SUM[16][543:0] = DATA\_SUM[4607:4064] |

Table 3‑6 : HPSEL Number to DDRIN RAM Mapping

An independent DATA\_SUM bus is generated from independent addresses to independent sets of up to 144-off RAMs (dependent on “harmonic\_g”) within the DDRIN\_STORE sub-module for each SUMMER sub-module instance, since each SUMMER sub-module instance will be processing different “seed\_f0” potential Periodicity Candidate locations within the FOP.

## Harmonic Summing: SUMMER Sub-Module

The SUMMER sub-module architecture is shown in Figure 3‑3 below:-

HPSEL[1][6:0]

**HP\_SEL**

RST\_SYS\_N

M[31:0]

DONE\_SUM

A[3:0]

HPSEL\_WRITE[6:0]

CLK\_SYS

HPSEL\_READ[6:0]

MC\_ADDR[15:0]

HPSEL\_WREN

P\_EN[4:0]

HPSEL[2][6:0]

HPSEL[3][6:0]

HPSEL[4][6:0]

HPSEL[5][6:0]

HPSEL[6][6:0]

NEW\_SUM

DATA\_SUM[31:0]

**SUMMER\_TREE**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

DATA\_SUM[127:32]

DATA\_SUM[223:128]

DATA\_SUM[383:224]

DATA\_SUM[543:384]

DATA\_SUM[767:544]

DATA\_SUM[991:768]

DATA\_SUM[SLICE]

**T\_SEL**

SEED\_NUM

[1 to harmonic\_g][4:0]

T\_READ[31:0]

T\_WRITE[31:0]

T\_SET

CLK\_MC

T[harmonic\_g]

[31:0]

T[7][31:0]

T[6][31:0]

T[5][31:0]

T[4][31:0]

T[3][31:0]

T[2][31:0]

T[1][31:0]

TC[1 to Nodes]

PWR[1 to Nodes][31:0]

H[3:0]

ANALYSIS\_RUN

MSEL[1 to harmonic\_g]

ROW\_INFO

[1 to harmonic\_g] [6:0]

T\_WREN

[12:0]

LAST\_RESULT

**harmonic\_g**

**:**

|  |  |
| --- | --- |
| harmonic\_g | Nodes |
| 8 | 40 |
| 9 | 49 |
| 10 | 60 |
| 11 | 71 |
| 12 | 84 |
| 13 | 97 |
| 14 | 112 |
| 15 | 127 |
| 16 | 144 |

**:**

**:**

**+**

|  |  |
| --- | --- |
| harmonic\_g | SLICE |
| 8 | 1279:992 |
| 9 | 1567:1280 |
| 10 | 1919:1568 |
| 11 | 2271:1920 |
| 12 | 2687:2272 |
| 13 | 3103:2688 |
| 14 | 3583:3104 |
| 15 | 4063:3584 |
| 16 | 4607:4064 |

T\_EN[ 1 to harmonic\_g]

COMP\_EN[ 1 to harmonic\_g]

HPSEL\_EN

[1 to harmonic\_g]

Figure ‑ : SUMMER Sub-module Architecture

HPSEL

[harmonic\_g][6:0]

The SUMMER sub-module (Figure 3‑3) shall request the required FOP information for up to 16 harmonics (upper configurable H[3:0] value limited by “harmonic\_g”) (including the fundamental tone) of a potential Periodicity Candidate “seed\_f0”, sum the harmonics and check if the summed power at the various summing stages exceeds a configurable threshold, and report any occasions where the power thresholds are crossed.

For a set of up to FOP columns, as shown in Figure 1‑1, the seed\_f0 is located in the first column.

Each location of the FOP actually identifies a Periodicity Candidate frequency range (FOP x-axis) and Periodicity Candidate orbital acceleration range (FOP y-axis) rather than a spot frequency and acceleration. Due to this quantisation there is ambiguity in the FOP column and row that a harmonic corrected by the convolution exists.

The search for the frequency of a potential Periodicity Candidate shall be accommodated by the TGEN sub-module stepping through (as time goes by) the FOP to action the reading of successive sets of up to 144-off FOP columns (upper configurable value set by H[3:0] and limited by “harmonic\_g”). The ambiguity in the potential Periodicity Candidate frequency of the seed\_f0 in the first column is the reason why more columns of the FOP are analysed as the harmonic increases (as shown in Figure 1‑1). The harmonic summing tree structure inherently accommodates the analysis of this ambiguity in the Periodicity Candidate frequency.

The search for the orbital acceleration of a potential Periodicity Candidate shall be accommodated by the SUMMER sub-module stepping through (as time goes by) rows of the first column of the up to 144-off FOP columns analysed (upper configurable value set by H[3:0] and limited by “harmonic\_g”). Figure 1‑1 shows the example where the orbital acceleration of the potential Periodicity Candidate seed\_f0 is within the central 21 rows of the first of the FOP columns analysed. However the SUMMER sub-module shall be more flexible than that shown in Figure 1‑1, since any 21 rows (even non-contiguous) of the first of the FOP columns analysed may contain the seed\_f0. The ambiguity in the potential Periodicity Candidate orbital acceleration of the seed\_f0 shall be accommodated by the SUMMER sub-module by further stepping through (as time goes by) rows of the up to 144-off FOP columns analysed (upper configurable value set by H[3:0] and limited by “harmonic\_g”) (apart from the first which contains the selected seed\_f0).

The selection of FOP rows to support the analysis of the potential Periodicity Candidate seed\_f0 orbital acceleration and the ambiguity of the orbital acceleration shall be provided by MC configuration via the HPSEL\_WRITE/READ[6:0] signals and associated WREN/ADDR signals to allow storage in configuration RAM within the SUMMER sub-module. Up to 21-off different orbital acceleration values (i.e. seed\_f0 in up to 21 FOP rows of the first FOP column analysed) shall be supported and for each seed\_f0 row up to 11 orbital acceleration ambiguity slopes shall be supported. In addition there are two sets of configuration information to support two independent analysis runs of the DM. Two analysis runs shall be supported for the FOP of a DM, with each analysis run having independent configuration for the seed\_f0 locations and orbital acceleration ambiguity settings.

MC configurable power thresholds via T\_WRITE/READ[6:0] shall be used at each stage of summation in the summing tree. Each of the up to 21 seed\_f0 row locations requires a unique threshold setting, and the harmonics of that seed\_f0 also require unique threshold settings. In addition at a certain point in the FOP x-axis (Periodicity Candidate frequency) the set of thresholds shall need to change to a different set, with T\_SET indicating when the changeover should occur. The result of any threshold crossing shall be passed to the TREP sub-module.

The SUMMER sub-module shall receive the following from the HSUMMCI sub-module (see Figure 3‑1):-

* **HPSEL\_WRITE[6:0]**: This is the MC configuration to allow the required FOP row locations (for “harmonic\_g”-off harmonics inclusive of the seed\_f0 tone) to be read from the DDRIN sub-module to allow summing of different orbital acceleration and orbital acceleration ambiguity slopes of a potential Periodicity Candidate to be summed. 21-off different Periodicity Candidate orbital acceleration values are supported (equating to 21 different FOP row locations for the seed\_f0), each with 11 orbital acceleration ambiguity slopes. In addition a FOP may be analysed twice, requiring two separate sets of configuration values. The value of each configuration word is 7-bits to allow selection of one of 85 FOP rows, with legal values in the range 0x00 to 0x5F. An extra value of 0x60 shall be used to select a fixed value “M[31:0]” as a diagnostic input for the information to the summing tree instead of the FOP values from the DDRIN sub-module.
* **T\_WRITE[31:0]**: This is the MC configuration for the power thresholds for each harmonic stage of the harmonic summing. Each of the 21-off orbital acceleration values (pertaining to a FOP rows containing a seed\_f0) require unique threshold values for each “harmonic\_g”--off harmonics (inclusive of the fundamental seed\_f0). In addition two sets of thresholds are required for the low and high Periodicity Candidate frequency regions of the FOP, with the T\_SET signal indicating when to changeover from the low frequency set to the high frequency set of thresholds. Each configuration word is a 32-bit IEEE 754 value indicating the value of the summed power, above which a threshold crossing is reported by the summing tree.
* **MC\_ADDR[15:0]**: This is the address to the configuration RAMs within the HP\_SEL and T\_SEL sub-modules. The lower address bits shall be used to address the separate RAMs for each harmonic. The upper address bits shall be used to address a particular SUMMER instance.
* **HPSEL\_WREN**: This is the write enable to the configuration RAMs within the HP\_SEL sub-module.
* **T\_WREN**: This is the write enable to the configuration RAMs within the T\_SEL sub-module.
* **M[31:0]**: This is a diagnostic value that is supplied to the harmonic summing tree instead of the FOP information (via the DATA\_SUM bus) for the relevant harmonic. If a HPSEL configuration value is set to 0x60 for a harmonic the fixed “M[31:0]” value is supplied to the summing tree for that harmonic.

The SUMMER sub-module shall receive the following from the TGEN sub-module (see Figure 3‑1):-

* **NEW\_SUM**: This is a pulsed indication (1 cycle wide) to command the SUMMER sub-module to commence the harmonic summing analysis of the next set of up to 144-off FOP columns from the DDRIN sub-module.
* **ANALYSIS\_RUN**: This is an indication of the analysis run that is being performed.
* **T\_SET**: This is an indication of which set of threshold values to use in the harmonic summing analysis. There are two sets of threshold values, one for low frequency Periodicity Candidates and one for high frequency Periodicity Candidates. When T\_SET = 0 the low frequency thresholds shall be used and when T\_SET = 1 the high frequency thresholds shall be used.
* **P\_EN[4:0]**: This is the number of actual orbital acceleration values for a potential Periodicity Candidate that are analysed. This is synonymous with the number of FOP rows in which the seed\_f0 may exist. A maximum of 21-off orbital acceleration values (i.e. seed\_f0 located in 21 FOP rows) are analysed. If fewer orbital acceleration values are analysed the harmonic summing time is reduced.
* **A[3:0]**: This is the number of actual orbital acceleration ambiguity slopes for the harmonics of each seed\_f0. A maximum of 11-off orbital acceleration ambiguity slopes (i.e. 11 possible row locations of each harmonic of a seed\_f0). If fewer orbital acceleration slopes are analysed the harmonic summing time is reduced.
* **H[3:0]**: This is the number of actual harmonics that are to be summed for each seed\_f0. A maximum of “harmonic\_g”-off harmonics (including the seed\_f0) may be summed. If fewer harmonics are analysed the harmonic summing time is reduced.

The SUMMER sub-module shall receive the following from the DDRIN sub-module (see Figure 3‑1 & Figure 3‑2):-

* **DATA\_SUM[DWIDTH:0]**: This is the FOP information required for the harmonic summing analysis:

|  |  |
| --- | --- |
| harmonic\_g | DWIDTH |
| 8 | 1279 |
| 9 | 1567 |
| 10 | 1919 |
| 11 | 2271 |
| 12 | 2687 |
| 13 | 3103 |
| 14 | 3583 |
| 15 | 4063 |
| 16 | 4607 |

### Harmonic Orbital Acceleration: HP\_SEL Sub-Module

The HP\_SEL sub-module architecture is shown in Figure 3‑4 below:-

HPSEL[1][6:0]

MC\_WREN

HPSEL[4][6:0]

HPSEL[5][6:0]

HPSEL[6][6:0]

HPSEL[7][6:0]

HPSEL

[harmonic\_g]

[6:0]

SEED\_NUM

[1 to harmonic\_g[4:0]

**ADDRESS\_**

**DECODE**

RST\_SYS\_N

DONE\_SUM

A [3:0]

HPSEL\_WRITE[6:0]

CLK\_SYS

HPSEL\_READ[6:0]

MC\_ADDR[13:0]

P\_EN [4:0]

HPSEL[2][6:0]

HPSEL[3][6:0]

NEW\_SUM

CLK\_MC

H[3:0]

**RAM #1**

**HPSEL Config for Harmonic #1**

**(i.e. 1x f0)**

**RAM #2**

**HPSEL Config for Harmonic #2**

**(i.e 2 x f0)**

**RAM #3**

**HPSEL Config for Harmonic #3**

**(i.e 3 x f0)**

**RAM #4**

**HPSEL Config for Harmonic #4**

**(i.e 4 x f0)**

**RAM #5**

**HPSEL Config for Harmonic #5**

**(i.e 5 x f0)**

**RAM #6**

**HPSEL Config for Harmonic #6**

**(i.e 6 x f0)**

**RAM #7**

**HPSEL Config for Harmonic #7**

**(i.e 6 x f0)**

**SELECTION\_ENGINE**

Write Enables

SEL

ANALYSIS\_RUN

ADDR[1][9:0]

[13:4]

ADDR[2][9:0]

ADDR[3][9:0]

ADDR[4][9:0]

ADDR[5][9:0]

ADDR[6][9:0]

ADDR[7][9:0]

ADDR

[harmonic\_g][9:0]

**0x60 CHK**

MSEL[1]

**0x60 CHK**

**0x60 CHK**

**0x60 CHK**

**0x60 CHK**

**0x60 CHK**

**0x60 CHK**

**0x60 CHK**

MSEL[2]

MSEL[3]

MSEL[4]

MSEL[5]

MSEL[6]

MSEL[7]

MSEL

[harmonic\_g]

ROW\_INFO

[1to harmonic\_g]

[6:0]

**PIPELINE\_DELAY\_1**

LAST\_RESULT

**harmonic\_g**

**:**

**RAM #harmonic\_g**

**HPSEL Config for #harmonic\_g**

**(i.e harmonic\_g x f0)**

HPSEL\_EN

[1 to harmonic\_g]

COMP\_EN

[1 to harmonic\_g]

T\_EN

[1 to harmonic\_g]

Figure ‑ : HP\_SEL Sub-module Architecture

The HP\_SEL sub-module shall receive the following from the HSUMMCI sub-module (see Figure 3‑1, Figure 3‑3 & Figure 3‑4):-

* **HPSEL\_WRITE[6:0]**: This is the MC configuration to allow the required FOP row locations (for harmonic\_g-off harmonics inclusive of the seed\_f0 tone) to be read from the DDRIN sub-module to allow summing of different orbital acceleration and orbital acceleration ambiguity slopes of a potential Periodicity Candidate to be summed. 21-off different Periodicity Candidate orbital acceleration values are supported (equating to 21 different FOP row locations for the seed\_f0), each with 11 orbital acceleration ambiguity slopes. In addition a FOP may be analysed twice, requiring two separate sets of configuration values. The value of each configuration word is 7-bits to allow selection of one of 85 FOP rows, with legal values in the range 0x00 to 0x5F. An extra value of 0x60 shall be used to select a fixed value “M[31:0]” as a diagnostic input for the information to the summing tree instead of the FOP values from the DDRIN sub-module.
* **MC\_ADDR[13:0]**: This is the address to the configuration RAMs within the HP\_SEL sub-module. The lower address bits shall be used to address the separate RAMs for each harmonic, so that in the MC bitmap the FOP row selections for each harmonic for one seed/ambiguity slope setting are in adjacent addresses.
* **HPSEL\_WREN**: This is the write enable to the configuration RAMs within the HP\_SEL sub-module.

The HP\_SEL sub-module shall receive the following from the TGEN sub-module (see Figure 3‑1 & Figure 3‑3) :-

* **NEW\_SUM**: This is a pulsed indication (1 cycle wide) to command the SUMMER sub-module to commence the harmonic summing analysis of the next set of up to 144-off FOP columns from the DDRIN sub-module.
* **ANALYSIS\_RUN**: This is an indication of the analysis run that is being performed and shall be used to address the correct set of HP\_SEL configuration values.
* **P\_EN[4:0]**: This is the number of actual orbital acceleration values for a potential Periodicity Candidate that are analysed. This is synonymous with the number of FOP rows in which the seed\_f0 may exist. A maximum of 21-off orbital acceleration values (i.e. seed\_f0 located in 21 FOP rows) are analysed. If fewer orbital acceleration values are analysed the harmonic summing time is reduced.
* **A[3:0]**: This is the number of actual orbital acceleration ambiguity slopes for the harmonics of each seed\_f0. A maximum of 11-off orbital acceleration ambiguity slopes (i.e. 11 possible row locations of each harmonic of a seed\_f0). If fewer orbital acceleration slopes are analysed the harmonic summing time is reduced.
* **H[3:0]**: This is the number of actual harmonics that are to be summed for each seed\_f0. A maximum of 16-off harmonics (including the seed\_f0) may be summed. If fewer harmonics are analysed the harmonic summing time is reduced.

The HP\_SEL sub-module shall consist of RAMs to store the MC configuration to support selection of the FOP rows for the different harmonics to support the orbital acceleration search and orbital acceleration ambiguity, with a “SELECTION ENGINE” to select the required RAM location for a particular orbital acceleration and orbital acceleration ambiguity. The RAM contents shall actually be the addresses for the RAMs within the DDRIN\_STORE sub-module so that the values of the desired FOP row for each harmonic can be obtained and provided to the SUMMER\_TREE sub-module via the DATA\_SUM bus.

Each harmonic shall have a separate RAM in the HP\_SEL sub-module so that it can be independently addressed at the appropriate time to output the address required by the DDRIN\_STORE sub-module for the FOP columns pertaining to the harmonic. The arrangement of information within one of HP\_SEL sub-module MC configuration RAMs is shown below in Table 3‑7.

|  |  |  |  |
| --- | --- | --- | --- |
| HP\_SEL sub-module  RAM  Address.  (Configuration RAM per harmonic) | Harmonic Summing test that  HPSEL[n][6:0]  supports | | |
| Analysis Run | Seed\_f0  Orbital Acceleration Setting | Acceleration Ambiguity Setting |
| 0x000 | 1st | 1st Seed\_f0 Row setting | 1st Ambiguity setting |
| 0x001 | 2nd Ambiguity setting |
| 0x002 | 3rd Ambiguity setting |
| 0x003 | 4th Ambiguity setting |
| 0x004 | 5th Ambiguity setting |
| 0x005 | 6th Ambiguity setting |
| 0x006 | 7th Ambiguity setting |
| 0x007 | 8th Ambiguity setting |
| 0x008 | 9th Ambiguity setting |
| 0x009 | 10th Ambiguity setting |
| 0x00A | 11th Ambiguity setting |
| 0x010 | 2nd Seed\_f0 Row setting | 1st Ambiguity setting |
| 0x011 | 2nd Ambiguity setting |
| 0x012 | 3rd Ambiguity setting |
| 0x013 | 4th Ambiguity setting |
| 0x014 | 5th Ambiguity setting |
| 0x015 | 6th Ambiguity setting |
| 0x016 | 7th Ambiguity setting |
| 0x017 | 8th Ambiguity setting |
| 0x018 | 9th Ambiguity setting |
| 0x019 | 10th Ambiguity setting |
| 0x01A | 11th Ambiguity setting |
| : | : | : |
| 0x140 | 21st Seed\_f0 Row setting | 1st Ambiguity setting |
| 0x141 | 2nd Ambiguity setting |
| 0x142 | 3rd Ambiguity setting |
| 0x143 | 4th Ambiguity setting |
| 0x144 | 5th Ambiguity setting |
| 0x145 | 6th Ambiguity setting |
| 0x146 | 7th Ambiguity setting |
| 0x147 | 8th Ambiguity setting |
| 0x148 | 9th Ambiguity setting |
| 0x149 | 10th Ambiguity setting |
| 0x14A | 11th Ambiguity setting |
| : | : | : | : |
| 0x200 | 2nd | 1st Seed\_f0 Row setting | 1st Ambiguity setting |
| 0x201 | 2nd Ambiguity setting |
| 0x202 | 3rd Ambiguity setting |
| 0x203 | 4th Ambiguity setting |
| 0x204 | 5th Ambiguity setting |
| 0x205 | 6th Ambiguity setting |
| 0x206 | 7th Ambiguity setting |
| 0x207 | 8th Ambiguity setting |
| 0x208 | 9th Ambiguity setting |
| 0x209 | 10th Ambiguity setting |
| 0x20A | 11th Ambiguity setting |
| : | : | : |
| 0x340 | 21st Seed\_f0 Row setting | 1st Ambiguity setting |
| 0x341 | 2nd Ambiguity setting |
| 0x342 | 3rd Ambiguity setting |
| 0x343 | 4th Ambiguity setting |
| 0x344 | 5th Ambiguity setting |
| 0x345 | 6th Ambiguity setting |
| 0x346 | 7th Ambiguity setting |
| 0x347 | 8th Ambiguity setting |
| 0x348 | 9th Ambiguity setting |
| 0x349 | 10th Ambiguity setting |
| 0x34A | 11th Ambiguity setting |

Table 3‑7 : HP\_SEL RAM Address to Seed\_f0 Setting/ Acceleration Ambiguity Setting Relationship (There is a separate configuration RAM / configuration table as shown above for each harmonic)

The contents of each location of the RAMs within the HP\_SEL sub-module shall normally be in the range 0x00 to 0x5F relating to a FOP row as detailed in sec 3.2, however if the selected RAM location contains a value of 0x60 the associated “MSEL” signal shall be asserted to allow the global configuration value “M[31:0]” to be passed to the summing tree instead of FOP information.

The SELECTION\_ENGINE sub-module shall be triggered by a single cycle pulse of the “NEW\_SUM” signal which shall indicate that the next set of up to 144-off FOP columns (dependent on harmonic\_g) is available in the DDRIN sub-module.

The SELECTION\_ENGINE sub-module shall use the following information to control the addresses to the RAMs containing the HPSEL information (see Figure 3‑4):-

* **P\_EN[4:0]** to determine how many orbital acceleration values are to be analysed (i.e. the number of seed\_f0 locations in the first of the up to 144 FOP columns (dependent on harmonic\_g). A maximum of 21 seed\_f0 locations are supported.
* **A[3:0]** to determine how many orbital acceleration ambiguity settings are to be analysed for each orbital acceleration (i.e. seed\_f0 location). A maximum of 11 orbital acceleration ambiguity settings are supported.
* **H[3:0]** to determine how many harmonics are to be analysed. A maximum of 16 harmonics (inclusive of the fundamental tone) are supported.
* **ANALYSIS\_RUN** to determine which of up to two analysis runs are being performed

The SELECTION\_ENGINE sub-module shall address each of the RAMs within the HP\_SEL sub-module in a staggered fashion, since the summing tree acts in a pipeline fashion, with the summing calculation for a harmonic using the results of the summation of the previous harmonic. Hence the windows when the HPSEL\_EN signals are asserted to the DDRIN sub-module are staggered as the harmonic increases. This staggering is shown in Figure 3‑5 below for the case where “harmonic\_g” = 8. The terms in brackets in Figure 3‑5 below are (Acceleration Ambiguity Number, Orbital Acceleration Seed\_f0 Row Number), with 11 Acceleration Ambiguity settings and 21 Orbital Acceleration Seed\_f0 Row in this example (i.e. A[3:0] = “1010” and P\_EN[4:0] = “10100”).

Figure ‑ : HPSEL Staggered Progression

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  | |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |  | |  | | |  | | |  | |  | | |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |
|  |  |  |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | |  | | | | |  | | | | |  | | | |  | | | |  | | | | |  | | |  | | | | |  | | |  |

Clock

DATA\_SUM[31:0]

HPsel[1][6:0]

**(2,1)**

**(4,1)**

**(3,1)**

**(1,1)**

**(11,21)**

**……**

**(11,1)**

**(1,2)**

**(2,2)**

**(3,2)**

**(4,2)**

**(1,3)**

**(11,2)**

**......**

**……**

**(10,1)**

**(10,21)**

**(9,21)**

**(2,1)**

**(4,1)**

**(3,1)**

**(1,1)**

**(11,21)**

**……**

**(11,1)**

**(1,2)**

**(2,2)**

**(3,2)**

**(4,2)**

**(1,3)**

**(11,2)**

**......**

**……**

**(10,1)**

**(10,2 1)**

**(11,21)**

**(9,21)**

**(10,21)**

**(2,1)**

**(4,1)**

**(3,1)**

**(1,1)**

**(11,21)**

**……**

**(11,1)**

**(1,2)**

**(2,2)**

**(3,2)**

**(4,2)**

**(1,3)**

**(11,2)**

**......**

**……**

**(10,1)**

**(9,21)**

**(10,21)**

**(9,21)**

**(2,1)**

**(4,1)**

**(3,1)**

**(1,1)**

**(11,21)**

**……**

**(11,1)**

**(1,2)**

**(2,2)**

**(3,2)**

**(4,2)**

**(1,3)**

**(11,2)**

**......**

**……**

**(10,1)**

**(10,21)**

**(9,21)**

**(8,21)**

**(2,1)**

**(4,1)**

**(3,1)**

**(1,1)**

**(11,21)**

**……**

**(11,1)**

**(1,2)**

**(2,2)**

**(3,2)**

**(4,2)**

**(1,3)**

**(11,2)**

**……**

**(10,1)**

**(10,21)**

**(9,21)**

**(8,21)**

**(7,21)**

**(2,1)**

**(4,1)**

**(3,1)**

**(1,1)**

**(11,21)**

**……**

**(11,1)**

**(1,2)**

**(2,2)**

**(3,2)**

**(4,2)**

**(11,2)**

**……**

**(10,1)**

**(10,21)**

**(9,21)**

**(8,21)**

**(7,21)**

**(6,21)**

**(2,1)**

**(4,1)**

**(3,1)**

**(1,1)**

**(11,21)**

**……**

**(11,1)**

**(1,2)**

**(2,2)**

**(3,2)**

**(4,2)**

**……**

**(10,1)**

**(10,21)**

**(9,21)**

**(8,21)**

**(7,21)**

**(6,21)**

**(5,21)**

**(2,1)**

**(4,1)**

**(3,1)**

**(1,1)**

**(11,21)**

**……**

**(11,1)**

**(1,2)**

**(2,2)**

**(3,2)**

**(4,2)**

**(10,1)**

**(10,21)**

**(9,21)**

**(8,21)**

**(7,21)**

**(6,21)**

**(5,21)**

**(4,21)**

**(2,1)**

**(4,1)**

**(3,1)**

**(1,1)**

**(11,21)**

**……**

**(11,1)**

**(1,2)**

**(2,2)**

**(3,2)**

**(4,2)**

**(1,3)**

**(11,2)**

**......**

**……**

**(10,1)**

**(10,21)**

**(11,21)**

**(9,21)**

**(1,1)**

**(2,1)**

**(4,1)**

**(3,1)**

**(1,1)**

**(11,21)**

**……**

**(11,1)**

**(1,2)**

**(2,2)**

**(3,2)**

**(4,2)**

**(1,3)**

**(11,2)**

**......**

**……**

**(10,1)**

**(10,2 1)**

**(11,21)**

**(9,21)**

**(10,21)**

**(2,1)**

**(4,1)**

**(3,1)**

**(1,1)**

**(11,21)**

**……**

**(11,1)**

**(1,2)**

**(2,2)**

**(3,2)**

**(4,2)**

**(1,3)**

**(11,2)**

**......**

**……**

**(10,1)**

**(10,21)**

**(9,21)**

**(10,21)**

**(9,21)**

**(2,1)**

**(4,1)**

**(3,1)**

**(1,1)**

**(11,21)**

**……**

**(11,1)**

**(1,2)**

**(2,2)**

**(3,2)**

**(4,2)**

**(1,3)**

**(11,2)**

**......**

**……**

**(10,1)**

**(9,21)**

**(10,21)**

**(9,21)**

**(8,21)**

**(2,1)**

**(4,1)**

**(3,1)**

**(1,1)**

**(11,21)**

**……**

**(11,1)**

**(1,2)**

**(2,2)**

**(3,2)**

**(4,2)**

**(1,3)**

**(11,2)**

**......**

**……**

**(10,1)**

**(10,21)**

**(9,21)**

**(8,21)**

**(7,21)**

**(2,1)**

**(4,1)**

**(3,1)**

**(1,1)**

**(11,21)**

**……**

**(11,1)**

**(1,2)**

**(2,2)**

**(3,2)**

**(4,2)**

**(1,3)**

**(11,2)**

**……**

**(10,1)**

**(10,21)**

**(9,21)**

**(8,21)**

**(7,21)**

**(6,21)**

**(2,1)**

**(4,1)**

**(3,1)**

**(1,1)**

**(11,21)**

**……**

**(11,1)**

**(1,2)**

**(2,2)**

**(3,2)**

**(4,2)**

**(11,2)**

**……**

**(10,1)**

**(10,21)**

**(9,21)**

**(8,21)**

**(7,21)**

**(6,21)**

**(5,21)**

**(2,1)**

**(4,1)**

**(1,1)**

**(11,21)**

**……**

**(11,1)**

**(1,2)**

**(2,2)**

**(3,2)**

**(4,2)**

**……**

**(10,1)**

**(10,21)**

**(9,21)**

**(8,21)**

**(7,21)**

**(6,21)**

**(5,21)**

**(4,21)**

HPsel[2][6:0]

HPsel[8[6:0]

HPsel[7][6:0]

HPsel[6][6:0]

HPsel[5][6:0]

HPsel[4[6:0]

HPsel[3][6:0]

**(9,21)**

**(10,21)**

**(8,21)**

**(6,21)**

**(5,21)**

**(4,21)**

**(3,21)**

**(7,21)**

DATA\_SUM[127:32]

DATA\_SUM[223:128]

DATA\_SUM[383:224]

DATA\_SUM[543:384]

DATA\_SUM[767:544]

DATA\_SUM[991:768]

DATA\_SUM[1279:992]

**Key : (Acceleration Ambiguity Number, Orbital Accelerationseed\_f0 Number)**

The actual delays in the SUMMER\_TREE increase as the harmonic number increases. Hence the HPSEL\_EN signal per harmonic to the DDRIN sub-module has to be delayed further to ensure the comparisons in the SUMMER\_TREE with the threshold values occur at the correct point in time, with different delays for each harmonic. The COMP\_EN signal per harmonic is a delayed version of the HPSEL\_EN signal per harmonic, with a potentially different delay per harmonic, ensuring the window when the comparisons occur is at the correct point in time.

Similarly the HPSEL[6:0] signal per harmonic that is sent to the DDRIN sub-module also needs to be delayed by an appropriate amount when it is sent to the TREP sub-module. This is because the TREP sub-module needs to associate any threshold crossings from the SUMMER\_TREE with the selected FOP Rows, i.e. the HPSEL[6:0] information per harmonic. The PIPELINE\_DELAY\_1 sub-module depicts this delaying of the output (i.e. HPSEL[6:0]) of the RAMs within the HP\_SEL sub-module, effectively providing the FOP Row values (translation required in the PC/Computer using a fixed mapping detailed in Table 3‑5). The delayed RAM outputs are renamed to ROW\_INFO[6:0] for each harmonic and shall be passed to the TREP sub-module so that they can be reported to the PC/Computer if a threshold is crossed. Hence the ROW\_INFO[6:0] signal per harmonic that is sent to the TREP sub-module has delays per harmonic in sympathy with the COMP\_EN delays.

The SELECTION\_ENGINE sub-module shall provide an indication per harmonic of which of the up to 21-off seed\_f0s that it is processing to control the phase of the threshold selection values in the T\_SEL sub-module. The SEED\_NUM[4:0] signal per harmonic provides this information and the T\_EN signal pulses at the appropriate time to ensure the threshold values for the new SEED\_NUM are presented to the SUMMER\_TREE at the appropriate time.

Once the SELECTION ENGINE sub-module has completed processing the current up to 144-off (dependent on harmonic\_g) FOP columns it shall pulse the “DONE\_SUM” signal for a single cycle.

The SELECTION\_ENGINE shall provide a LAST\_RESULT signal to TREP that shall indicate the end of processing the orbital acceleration ambiguity tests for a seed\_f0.

### Threshold Selection: T\_SEL Sub-Module

The T\_SEL sub-module architecture is shown in Figure 3‑6 below:-

Figure ‑ : T\_SEL Sub-module Architecture

T[1][31:0]

T\_WREN

T[4][31:0]

T[5][31:0]

T[6][31:0]

T[7][31:0]

T[harmonic\_g]

31:0]

**ADDRESS\_**

**DECODE**

RST\_SYS\_N

T\_WRITE[31:0]

CLK\_SYS

T\_READ[31:0]

MC\_ADDR[10:0]

T[2][31:0]

T[3][31:0]

CLK\_MC

T\_SET

**RAM #1**

**T Configuration for Harmonic #1**

**(i.e. fundamental 1x f0)**

**RAM #2**

**T Configuration for Harmonic #2**

**(i.e 2 x f0)**

**RAM #3**

**T Configuration for Harmonic #3**

**(i.e 3 x f0)**

**RAM #4**

**T Configuration for Harmonic #4**

**(i.e 4 x f0)**

**RAM #5**

**T Configuration for Harmonic #5**

**(i.e 5 x f0)**

**RAM #6**

**T Configuration for Harmonic #6**

**(i.e 6 x f0)**

**RAM #7**

**T Configuration for Harmonic #7**

**(i.e 6 x f0)**

**RAM #harmonic\_g**

**T Configuration for Harmonic #harmonic\_g**

**(i.e harmonic\_g x f0)**

**SELECTION\_ENGINE**

Write Enables

SEL

SEED\_NUM

[1 to harmonc\_g] [4:0]

ADDR[1][5:0]

[10:4]

ADDR[2][5:0]

ADDR[3][5:0]

ADDR[4][5:0]

ADDR[5][5:0]

ADDR[6][5:0]

ADDR[7][5:0]

ADDR

[harmonic\_g]

[5:0]

**:**

**harmonic\_g**

T\_EN

[1 to harmonic\_g]

The T\_SEL sub-module shall receive the following from the HSUMMCI sub-module (see Figure 3‑1, Figure 3‑3 & Figure 3‑6):-

* **T\_WRITE[31:0]**: This is the MC configuration for the power thresholds for each harmonic stage of the harmonic summing. Each of the 21-off orbital acceleration values (pertaining to a FOP rows containing a seed\_f0) require unique threshold values for each 16-off harmonics (inclusive of the fundamental seed\_f0). In addition two sets of thresholds are required for the low and high Periodicity Candidate frequency regions of the FOP, with the T\_SET signal indicating when to changeover from the low frequency set to the high frequency set of thresholds. Each configuration word is a 32-bit IEEE 754 value indicating the value of the summed power, above which a threshold crossing is reported by the summing tree.
* **MC\_ADDR[10:0]**: This is the address to the configuration RAMs within the T\_SEL sub-modules. The lower address bits shall be used to address the separate RAMs for each harmonic.
* **T\_WREN**: This is the write enable to the configuration RAMs within the T\_SEL sub-modules.

The T\_SEL sub-module shall receive the following from the TGEN sub-module (see Figure 3‑1, Figure 3‑3 & Figure 3‑6):-

* **T\_SET**: This is an indication of which set of threshold values to use in the harmonic summing analysis. There are two sets of threshold values, one for low frequency Periodicity Candidates and one for high frequency Periodicity Candidates. When T\_SET = 0 the low frequency thresholds shall be used and when T\_SET = 1 the high frequency thresholds shall be used.

The T\_SEL sub-module shall receive the following from the HP\_SEL sub-module (see Figure 3‑3 & Figure 3‑4):-

* **SEED\_NUM[1 to harmonic\_g][4:0] & T\_EN[1 to harmonic\_g]**: This is a number in the range 0 to 20 to represent which of up to 21 different FOP rows that a seed\_f0 may exist in a FOP column, with enables to ensure the each harmonic has its threshold staggered in presentation to the summing tree. The T\_EN for the harmonics pulses when there is a new SEED\_NUM value for the harmonic.

The T\_SEL sub-module shall consist of RAMs to store the MC configuration to support selection of the power thresholds for the harmonics of different potential Periodicity Candidate orbital acceleration values (i.e. different seed\_f0 row locations in the FOP).

Each harmonic shall have a separate RAM in the T\_SEL sub-module so that it can be independently addressed at the appropriate time to output the threshold required by the associated harmonic being processed by the summing tree. The arrangement of information within one of T\_SEL sub-module MC configuration RAMs is shown below in Table 3‑8.

|  |  |  |
| --- | --- | --- |
| T\_SEL sub-module  RAM  Address.  (Configuration RAM per harmonic) | Threshold that  T[n][31:0]  supports | |
| T\_SET | Seed\_f0  Threshold Setting |
| 0x00 | Low Freq  Set | 1st Seed\_f0 Threshold Setting |
| 0x01 | 2nd Seed\_f0 Threshold Setting |
| 0x02 | 3rd Seed\_f0 Threshold Setting |
| 0x03 | 4th Seed\_f0 Threshold Setting |
| 0x04 | 5th Seed\_f0 Threshold Setting |
| 0x05 | 6th Seed\_f0 Threshold Setting |
| 0x06 | 7th Seed\_f0 Threshold Setting |
| 0x07 | 8th Seed\_f0 Threshold Setting |
| 0x08 | 9th Seed\_f0 Threshold Setting |
| 0x09 | 10th Seed\_f0 Threshold Setting |
| 0x0A | 11th Seed\_f0 Threshold Setting |
| 0x0B | 12th Seed\_f0 Threshold Setting |
| 0x0C | 13th Seed\_f0 Threshold Setting |
| 0x0D | 14th Seed\_f0 Threshold Setting |
| 0x0E | 15th Seed\_f0 Threshold Setting |
| 0x0F | 16th Seed\_f0 Threshold Setting |
| 0x10 | 17th Seed\_f0 Threshold Setting |
| 0x11 | 18th Seed\_f0 Threshold Setting |
| 0x12 | 19th Seed\_f0 Threshold Setting |
| 0x13 | 20th Seed\_f0 Threshold Setting |
| 0x14 | 21st Seed\_f0 Threshold Setting |
| 0x20 | High Freq Set | 1st Seed\_f0 Threshold Setting |
| 0x21 | 2nd Seed\_f0 Threshold Setting |
| 0x22 | 3rd Seed\_f0 Threshold Setting |
| 0x23 | 4th Seed\_f0 Threshold Setting |
| 0x24 | 5th Seed\_f0 Threshold Setting |
| 0x25 | 6th Seed\_f0 Threshold Setting |
| 0x26 | 7th Seed\_f0 Threshold Setting |
| 0x27 | 8th Seed\_f0 Threshold Setting |
| 0x28 | 9th Seed\_f0 Threshold Setting |
| 0x29 | 10th Seed\_f0 Threshold Setting |
| 0x2A | 11th Seed\_f0 Threshold Setting |
| 0x2B | 12th Seed\_f0 Threshold Setting |
| 0x2C | 13th Seed\_f0 Threshold Setting |
| 0x2D | 14th Seed\_f0 Threshold Setting |
| 0x2E | 15th Seed\_f0 Threshold Setting |
| 0x2F | 16th Seed\_f0 Threshold Setting |
| 0x30 | 17th Seed\_f0 Threshold Setting |
| 0x31 | 18th Seed\_f0 Threshold Setting |
| 0x32 | 19th Seed\_f0 Threshold Setting |
| 0x33 | 20th Seed\_f0 Threshold Setting |
| 0x34 | 21st Seed\_f0 Threshold Setting |

Table 3‑8 : T\_SEL RAM Address to Seed\_f0 Threshold Setting Relationship

The contents of each location of the RAMs within the T\_SEL sub-module shall be a 32-bit IEEE 754 single precision floating point value representing a power threshold.

The SELECTION\_ENGINE sub-module shall use the SEED\_NUM[4:0] per harmonic value (with new value indicated by a pulse of the associated T\_EN signal) and the T\_SET value to set the appropriate address to each RAM within the T\_SEL sub-module.

The SEED\_NUM[4:0] and T\_EN signals are staggered by a cycle from the HP\_SEL module, however further staggering is required as the delays in the SUMMER\_TREE increase with increasing harmonic. The SELECTION\_ENGINE sub-module shall address each of the RAMs in the T\_SEL sub-module in an increasingly staggered fashion to take account of these increasing delays, such that the correct threshold value is presented to each harmonic in the SUMMER\_TREE at the appropriate point in time. Hence the staggered delay of setting addresses to each RAM within the T\_SEL sub-module shall be based upon the delays between harmonics in the SUMMER\_TREE.

### Original Summing Tree: SUMMER\_TREE Sub-Module

The Original SUMMER\_TREE sub-module summation architecture within the SUMMER sub-module for the case when harmonic\_g = 8 is shown in Figure 3‑7 below:-

Figure ‑ : Original SUMMER Harmonic Summing Tree Summation Architecture for harmonic\_g = 8

DATA

\_SUM

[31:0]

f0 Fundamental

2xf0

3xf0

4xf0

5xf0

6xf0

7xf0

8xf0

**+**

**+**

**+**

**+**

**+**

**+**

*FOP Harmonic Frequency Ambiguity (i.e. FOP Cols)*

*Results.*

*11 results over time for the different acceleration ambiguity slopes for each seed\_f0 (i.e FOP Rows)*

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

FOP

**+**

**+**

**+**

**+**

**+**

**+**

**+**

FOP

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

DATA

\_SUM

[63:32]

DATA

\_SUM

[95:64]

DATA

\_SUM

[127:96]

DATA

\_SUM

[159:128]

DATA

\_SUM

[191:160]

DATA

\_SUM

[223:192]

DATA

\_SUM

[383:352]

DATA

\_SUM

[351:320]

DATA

\_SUM

[319:288]

DATA

\_SUM

[287:256]

DATA

\_SUM

[255:224]

DATA

\_SUM

[543:512]

DATA

\_SUM

[511:480]

DATA

\_SUM

[479:448]

DATA

\_SUM

[447:416]

DATA

\_SUM

[415:384]

DATA

\_SUM

[991:960]

DATA

\_SUM

[703:672]

DATA

\_SUM

[671:640]

DATA

\_SUM

[639:608]

DATA

\_SUM

[607:576]

DATA

\_SUM

[575:544]

DATA

\_SUM

[767:736]

DATA

\_SUM

[735:704]

DATA

\_SUM

[959:928]

DATA

\_SUM

[927:896]

DATA

\_SUM

[895:864]

DATA

\_SUM

[863:832]

DATA

\_SUM

[831:800]

DATA

\_SUM

[799:768]

DATA

\_SUM

[1023:992]

DATA

\_SUM

[1055:1024]

DATA

\_SUM

[1087:1056]

DATA

\_SUM

[1119:1088]

DATA

\_SUM

[1151:1120]

DATA

\_SUM

[1183:1152]

DATA

\_SUM

[1215:1184]

DATA

\_SUM

[1247:1216]

DATA

\_SUM

[1279:1248]

**1**

**2**

**3**

**4**

**5**

**6**

**7**

**8**

**9**

**11**

**12**

**13**

**14**

**15**

**16**

**10**

**17**

**18**

**19**

**20**

**21**

**22**

**23**

**24**

**25**

**26**

**27**

**28**

**29**

**30**

**31**

**32**

**33**

**34**

**35**

**36**

**38**

**39**

**40**

**37**

*Number associated with the summation function relates to the FOP column number (1 to 40) identified in Figure 1‑1*

This original SUMMER\_TREE has been replaced by the “Enhanced” SUMMER\_TREE, however a description is provided here for completeness.

The SUMMER\_TREE sub-module shall receive the following from the HP\_SEL sub-module:-

* **COMP\_EN [1 to harmonic\_g]**: This is an enable per harmonic to ensure that comparisons of the summed powers with the threshold values occur at the correct point in time to take account of the staggered operation of the summing tree.

The SUMMER\_TREE sub-module shall receive the following from the DDRIN sub-module (see Figure 3‑1, Figure 3‑2, Figure 3‑3 & Figure 3‑7):-

* **DATA\_SUM[DWIDTH:0]**: This is the FOP information required for the harmonic summing analysis:

|  |  |  |
| --- | --- | --- |
| harmonic\_g | DWIDTH | FOP Columns analysed |
| 8 | 1279 | 40 |
| 9 | 1567 | 49 |
| 10 | 1919 | 60 |
| 11 | 2271 | 71 |
| 12 | 2687 | 84 |
| 13 | 3103 | 97 |
| 14 | 3583 | 112 |
| 15 | 4063 | 127 |
| 16 | 4607 | 144 |

* For 144-off FOP columns 144-off 32-bits values shall be required (4608 bits). On a per harmonic basis the data from the DDRIN module to each summation function within the SUMMER\_TREE sub-module may be replaced with the M[31:0] configuration value as a diagnostic aid.

The SUMMER\_TREE sub-module shall receive the following from the T\_SEL sub-module (see Figure 3‑3, Figure 3‑6 & Figure 3‑7):-

* **T[1 to harmonic\_g][31:0]**: This is the Threshold power value in IEEE 754 single precision format. Each harmonic receives an independent 32 bit value.

Each of the summation functions within the Harmonic Summing Tree of the SUMMER sub-module is an IEEE 754 single precision format (32-bit) adder.

The data supplied to each of the summation functions is the DATA\_SUM bus from the DDRIN sub-module. Each summation function requires 32-bits of the DATA\_SUM[DWIDTH:0] bus, consistent with the FOP column that is required for the harmonic and frequency ambiguity of the harmonic.

The original SUMMER\_TREE sub-module threshold detection architecture within the SUMMER sub-module is shown in Figure 3‑8 below for the case when harmonic\_g = 8.:-

f0 Fundamental

2xf0

3xf0

4xf0

5xf0

6xf0

7xf0

8xf0

**+**

**+**

**+**

**+**

**+**

**+**

*FOP Harmonic Frequency Ambiguity (i.e. FOP Cols)*

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

FOP

**+**

**+**

**+**

**+**

**+**

**+**

**+**

FOP

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**+**

**1**

**2**

**3**

**4**

**5**

**6**

**7**

**8**

**9**

**11**

**12**

**13**

**14**

**15**

**16**

**10**

**17**

**18**

**19**

**20**

**21**

**22**

**23**

**24**

**25**

**26**

**27**

**28**

**29**

**30**

**31**

**32**

**33**

**34**

**35**

**36**

**38**

**39**

**40**

**37**

*Number associated with the summation function relates to the FOP column number (1 to 40) identified in Figure 1‑1*

T[8][31:0]

T[7][31:0]

T[6][31:0]

T[5][31:0]

T[4][31:0]

T[3][31:0]

T[2][31:0]

T[1][31:0]

PWR[1]

[31:0]

PWR[4]

[31:0]

PWR[3]

[31:0]

PWR[2]

[31:0]

PWR[5]

[31:0]

PWR[6]

[31:0]

PWR[7]

[31:0]

PWR[8]

[31:0]

PWR[9]

[31:0]

PWR[10]

[31:0]

PWR[11]

[31:0]

PWR[13]

[31:0]

PWR[12]

[31:0]

PWR[14]

[31:0]

PWR[15]

[31:0]

PWR[16]

[31:0]

PWR[17]

[31:0]

PWR[18]

[31:0]

PWR[19]

[31:0]

PWR[20]

[31:0]

PWR[21]

[31:0]

PWR[22]

[31:0]

PWR[23]

[31:0]

PWR[24]

[31:0]

PWR[27]

[31:0]

PWR[28]

[31:0]

PWR[29]

[31:0]

PWR[30]

[31:0]

PWR[31]

[31:0]

PWR[36]

[31:0]

PWR[25]

[31:0]

PWR[26]

[31:0]

PWR[38]

[31:0]

PWR[33]

[31:0]

PWR[39]

[31:0]

PWR[32]

[31:0]

PWR[40]

[31:0]

PWR[35]

[31:0]

PWR[37]

[31:0]

PWR[34]

[31:0]

TC[1]

TC[4]

TC[3]

TC[2]

TC[18]

TC[7]

TC[6]

TC[5]

TC[8]

TC[9]

TC[10]

TC[11]

TC[12]

TC[14]

TC[15]

TC[16]

TC[17]

TC[20]

TC[40]

TC[22]

TC[19]

TC[38]

TC[29]

TC[23]

TC[21]

TC[30]

TC[24]

TC[34]

TC[25]

TC[26]

TC[36]

TC[37]

TC[27]

TC[28]

TC[31]

TC[39]

TC[35]

TC[33]

TC[32]

TC[13]

*The comparisons of power with the thresholds occur when the COMP\_EN for the harmonic is asserted.*

Figure ‑: Original SUMMER Harmonic Summing Tree Threshold Detection Architecture for harmonic\_g = 8

The results of each of the summation functions within the summing tree is compared against the appropriate threshold value “T[31:0]” from the T\_SEL sub-module. Each time new data is supplied to the summation functions (even if it has the same values) it shall result in the threshold detector carrying out a comparison. Each time a comparison occurs if the result of the summation equals or exceeds the T[31:0] value the associated “TC” signal shall be pulsed for a single cycle, and the actual summation result is provided as PWR[31:0]. The summation result and the threshold level are both represented using IEEE 754 32-bit single precision format.

### Enhanced Summing Tree

This is the SUMMER\_TREE architecture used in this design. Currently the design architecture only supports 12 harmonics (including the fundamental frequency) and this effectively limits the “harmonic\_g” generic to a maximum of 12.

The enhanced summing tree utilises multiple summations for some of the FOP columns (see Figure 3‑9 - Enhanced Summing Tree (f0 to 8xf0)). The numbers in the adder circles refer to the original adder/comparator numbers in Figure 3‑8. The highest summation result from adders with the same number is used to compare against the threshold value.

Figure 3‑10 - Enhanced Summing Tree (9xf0 to 12xf0) shows how the summing tree is continued for harmonics above 8. Due to the large number of adders only the left half of the tree is shown. The tree is symmetrical.



Figure ‑ - Enhanced Summing Tree (f0 to 8xf0)



Figure ‑ - Enhanced Summing Tree (9xf0 to 12xf0)

## Threshold Reporting: TREP Sub-Module

The TREP sub-module architecture is shown in Figure 3‑11 below:-

RST\_SYS\_N

A[1 to summer\_g]

[3:0]

ROW\_INFO

[1 to summer\_g]

[1 to harmonic\_g][6:0]

CLK\_SYS

MC\_ADDR[11:0]

CLK\_MC

H[3:0]

ANALYSIS\_RUN

PWR[1 to summer\_g]

[1 to Node] [31:0]

TC[1 to summer\_g]

[1 to Node]

SEED\_COL\_SUM

[21:0]

T\_FILTER\_EN

DM\_CNT\_RESET

DM\_CNT

[31:0]

T\_RESULTS

[31:0]

EXC[31:0]

**DM\_COUNTER**

HSUM\_DONE

**summer\_g, adder\_latency\_g, harmonic\_g**

TRIGGERED

SAVE\_DONE

LAST\_RESULT

CLEAR\_RESULTS]

SAVE\_RESULTS

WORKING\_PAGE

**FILTER**

**FILTER**

**FILTER**

**(per SUMMER)**

TC\_FILT

[1 to summer\_g]

[1 to Node]

PWR\_FILT

[1 to summer\_g]

[1 to Node] [31:0]

ROW\_INFO

[1 to summer\_g]

[1 to harmonic\_g]

[6:0]

SEED\_COL\_SUM

[1 to summer\_g]

[21:0]

**Bus Restructuring**

TC\_FILT

[1 to summer\_g][1]

PWR\_FILT

[1 to summer\_g][1] [31:0]

ROW\_INFO

[1 to summer\_g][1][6:0]

TC\_FILT

[1 to summer\_g][2 to 4]

PWR\_FILT

[1 to summer\_g][2 to 4] [31:0]

ROW\_INFO

[1 to summer\_g][2][6:0]

TC\_FILT

[1 to summer\_g]

[Node\_Range]

ROW\_INFO

[1 to summer\_g][harmonic\_g][6:0]

**HRES**

**Harmonic 1**

**HRES**

**Harmonic 2**

**HRES**

**Harmonic**

**Harmonic\_g**

PWR\_FILT

[1 to summer\_g][Node\_Range] [31:0]

*To all RESULT STORE blocks.*

Note: The number of HRES blocks is determined by harmonic\_g.

Select SEED\_COL\_SUM from FILTER with smallest A.

|  |  |
| --- | --- |
| harmonic\_g | Node |
| 8 | 40 |
| 9 | 49 |
| 10 | 60 |
| 11 | 71 |
| 12 | 84 |
| 13 | 97 |
| 14 | 112 |
| 15 | 127 |
| 16 | 144 |

|  |  |
| --- | --- |
| harmonic | Node\_Range |
| 1 | 1 |
| 2 | 2 to 4 |
| 3 | 5 to 7 |
| 4 | 8 to 12 |
| 5 | 13 to 17 |
| 6 | 19 to 24 |
| 7 | 25 to 31 |
| 8 | 32 to 40 |

|  |  |
| --- | --- |
| harmonic | Node\_Range |
| 9 | 41 to 49 |
| 10 | 50 to 60 |
| 11 | 61 to 71 |
| 12 | 72 to 84 |
| 13 | 85 to 97 |
| 14 | 98 to 112 |
| 15 | 113 to 127 |
| 16 | 128 to 144 |

Figure ‑ : TREP Sub-Module Architecture

The TREP sub-module shall have a capacity of 25 threshold crossing reports for each harmonic of an analysis run. Since two analysis runs per DM are supported a total of 50 threshold crossing reports per harmonic per DM are supported. To allow capture of threshold crossing reports for a new DM, whilst the threshold crossing reports are being read via the MC interface the TREP sub-module shall support two pages, with a page swap over when DM processing is complete, triggered via HSUM\_DONE.

**The TREP sub-module** **may receive simultaneous threshold crossing indications** from the SUMMER sub-module (see Figure 3‑3 & Figure 3‑8), - up to a limit of 144-off indications (determined by “harmonic\_g”), and these must all be processed without any loss of information.

The number of SUMMER sub-module instances that the TREP sub-module supports shall be set by the “summer\_g” generic. If more than one SUMMER sub-module instance is supported the TREP sub-module shall potentially receive simultaneous threshold crossing indications from all SUMMER sub-modules (i.e. 288-off simultaneous indications for two SUMMER sub-module instances and 432-off simultaneous indications for three SUMMER sub-module instances for the case when “harmonic\_g” = 16).

Since TREP has to handle many threshold detections arriving simultaneously from the SUMMER sub-modules, it is envisaged that a register based store will be necessary to store the results for one analysis run that can then be transferred to a RAM based store at the end of the analysis run.

The TREP sub-module shall receive the following from the HSUMMCI sub-module (see Figure 3‑1 & Figure 3‑11):-

* **T\_FILTER\_EN**: This shall determine if all threshold crossings for a seed\_f0 shall be reported, or only the highest harmonic crossing. (1 = report only the highest harmonic crossing).
* **MC\_ADDR[11:0]**: This shall provide the address to read the threshold crossing results “T\_RESULTS[31:0]” information from RAM within the TREP sub-module. TREP shall support 25 crossings per harmonic for two analysis runs, with each report consisting of four 32-bit words (1600 words per page).
* **DM\_CNT\_RESET**: A rising edge on this signal shall reset the free running DM counter within the TREP sub-module.

The TREP sub-module shall receive the following from the TGEN sub-module (see Figure 3‑1 and Figure 3‑11):-

* **TRIGGERED**: This shall indicate the start of a DM and shall be used to increment the DM counter.
* **ANALYSIS\_RUN**: This shall indicate which of two analysis runs on a DM that the threshold crossing information from the SUMMER sub-module is pertinent to.
* **HSUM\_DONE**: This shall indicate when a DM has been processed. This shall swap the reporting pages within the TREP sub-module that shall be used to capture the results for the new DM.
* **H[3:0]**: This shall indicate the number of harmonics that are being processed by the HSUM module. Threshold crossing results shall only be reported for the processed harmonics.
* **A[3:0]**: For each of two independent analysis runs this shall indicate the number of orbital acceleration ambiguity tests to be performed on each seed\_f0 location. This shall be required since if the configuration signal “T\_FILTER\_EN” is asserted it is necessary to know the number of orbital acceleration ambiguity tests for a seed\_f0.
* **CLEAR\_RESULTS**: This shall be pulsed high at the start of an analysis run to clear the register based result store.
* **SAVE\_RESULTS**: This shall be pulsed high at the end of an analysis run to cause the internal register based result store to be transferred to the RAM based result store.
* **SEED\_COL\_SUM[21:0]**: This shall indicate the seed FOP Column number that the threshold crossing results from the SUMMER sub-module are pertinent to.

The TREP sub-module shall receive the following from each SUMMER sub-module (see Figure 3‑1, Figure 3‑3, Figure 3‑8 & Figure 3‑11):-

For the descriptions below “Node” is related to “harmonic\_g as:-

|  |  |
| --- | --- |
| harmonic\_g | Node |
| 8 | 40 |
| 9 | 49 |
| 10 | 60 |
| 11 | 71 |
| 12 | 84 |
| 13 | 97 |
| 14 | 112 |
| 15 | 127 |
| 16 | 144 |

* **TC[1 to Node]**: This shall be a single cycle pulsed signal to indicate that the result of the associated summation function within the summing tree has met or exceeded the configured power threshold. The particular signal number indicates the FOP column number and harmonic number that the threshold crossing is pertinent to.
* **PWR[1 to Node][31:0]**: This shall indicate the actual power level result from the associated summation function within the summing tree. The particular signal number indicates the FOP column number and harmonic number that the power level is pertinent to. The power level is in the form of an IEEE 754 32-bit single precision value.
* **LAST\_RESULT**: This shall indicate the end of processing the orbital acceleration ambiguity tests for a seed\_f0. This shall facilitate the filtering of results when T\_FILTER\_EN = 1.
* **ROW\_INFO[1 to harmonc\_g][6:0]**: This shall effectively provide the FOP Row number (per harmonic) from the HP\_SEL sub-module within the SUMMER sub-module for which the summations have been performed for each harmonic. There is a fixed relationship between the ROW\_INFO value and the actual FOP Row number (translation detailed in Table 3‑5). The translation between ROW\_INFO and the FOP Row can be performed in the PC/Computer.

The TREP sub-module shall generate the following signals :-

* **SAVE\_DONE**: This shall be a single cycle pulsed signal to indicate that the transfer of the register based result store to the RAM base result store is complete.
* **DM\_COUNT**: Free running count of DMs processed that shall be readable by the PC/Computer via the PCIe interface.

### Harmonic Results Store: HRES Sub-Module

This sub-module stores the results for one harmonic. The number of columns processed for each harmonic is different, and thus the sizes of the input ports are different. A generic ‘harmonic\_num\_g’ (with range 0 to 15) shall be provided to configure the sub-module appropriately.

#### Threshold Crossing Results

When a threshold is crossed the following information shall be stored in four 32-bit words so that it can be accessed via T\_RESULTS[31:0]:-

* **Harmonic Number of Threshold Crossing Location**: The number of the TC signal with a single cycle pulse from the SUMMER sub-module shall be used to determine the harmonic number for the threshold crossing FOP location, - via reference to the summation function number of the summing tree.
* **FOP Column of Threshold Crossing Location [b]**: The SEED\_COL\_SUM[21:0] from the TGEN sub-module shall indicate the Seed\_f0Column.By using the TC signal in conjunction with the SEED\_COL\_SUM[21:0] it is possible to calculate the actual frequency bin [b] of the threshold crossing within the FOP.
* **FOP Row of Threshold Crossing Location [p]:** The ROW\_INFO[6:0] signal for the threshold crossing harmonic indicated by TC shall effectively provide the FOP Row for the threshold crossing [p]. The actual ROW\_INFO value has to be translated in the PC/Computer to get the actual FOP Row at which the threshold was crossed (see **Table 3‑5** for translation).
* **PWR[31:0]** **Providing the summed power of Threshold Crossing Location:** The PWR[31:0] signal power level associated with the pulsed TC shall be provided. This is an IEEE 754 32-bit single precision value.

In addition the word containing the harmonic number shall contain a bit to indicate if the report is valid (i.e. a single bit within the word shall indicate if the associated four words containing the Harmonic Number, FOP Column, FOP Row and Power level for a threshold crossing are to be treated as valid).

In total 4-off 32-bit words x 25 reports per harmonic x 2 analysis runs = 200 words are required per reporting page, with two pages to support an active page being updated by the current DM and a static page for the previous DM being accessed by the MC interface.

Hence after all necessary translation in the PC/Computer the following information for a threshold crossing shall exist:-

* The Harmonic number of the threshold crossing.
* The FOP Row [p] and Column Location [b] of the threshold crossing.
* The summed power level of the threshold crossing.

#### Exceeded Threshold Crossing Reports

Once the number of threshold crossing results exceeds 25, HRES shall continue to count the number of results and provide the following information to the PC/Computer via EXC[31:0]:

* **T\_EXC[31:0]** : The number of threshold crossing results that could not be stored.
* **P\_EXC[6:0]** : The ROW\_INFO[6:0] value for the first result that could not be stored. The PC/Computer shall be able to translate the ROW\_INFO value to a FOP Row for the threshold crossing location via a fixed mapping detailed in Table 3‑5.
* **S\_EXC[31:0]** : The SEED\_COL\_SUM[31:0] value for the first result that could not be stored. The SEED\_COL\_SUM directly provides the seed\_f0 FOP Column.

In total 3-off 32-bit words x 1 report per harmonic x 2 analysis runs = 6 words are required per reporting page, with two pages to support an active page being updated by the current DM and a static page for the previous DM being accessed by the MC interface.

#### Results Processing

The HRES sub-module has to be able to store multiple results arriving in one clock cycle and also be able to store further results arriving in consecutive cycles. The number of results that occur in one cycle is dependent on the harmonic that the results are for and the number of SUMMER modules instantiated.

The following steps describe how the results shall be processed (words in italics refer to labels in Figure 3‑12):

1. Store the input numbers for the active TC signals in an array (*active*) and count the number of active TC signals (*new\_count*).
2. From the number of active TC signals and the number of currently held results (*held\_count*), determine which of the 25 result stores need to be loaded (*load\_en*). Set the input result selector (*selector*) for these stores using the TC input number recorded previously.
3. Load the selected results into the stores:
   1. *PWR* is selected by the TC input number.
   2. *FOP ROW* [p] is ROW\_INFO from the appropriate SUMMER instance determined by the TC input number.
   3. *FOP COL* [b] shall be calculated as follows:  
      SEED\_COL\_SUM \* (harmonic\_num\_g + 1) + (offset dependent on TC input number)
4. Update the number of held results (*held\_count*). If the number of held results would exceed 25, then it should be held at 25 and T\_EXC incremented for any results that cannot be stored.

The Figure 3‑12 illustrates the above steps for a specific case:

* harmonic\_num\_g = 3
* summer\_g = 2
* two new results to be stored (one from each SUMMER)
* three results already stored

Figure ‑ : HRES Sub-Module Architecture

TC[1 to 2][2:0]

PWR[1 to 2][95:0]

ROW\_INFO[1 to 2][6:0]

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 5 | 4 | 3 | 2 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 5 | 4 | 3 | 2 | 1 | 0 |  |
| 0 | 0 | 0 | 0 | 1 | 1 | valid |
| 0 | 0 | 0 | 0 | 5 | 1 | active |

∑

new\_count

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |  |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | load\_en |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 5 | 1 | 0 | 0 | 0 | selector |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | PWR |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | FOP ROW |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | FOP COL |
| 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |  |

SEED\_COL[21:0]

harmonic\_num\_g+1

held\_count

2

3

### Threshold Crossing Results Filter: FILTER Sub-module

When T\_FILTER\_EN = 0, no filtering shall be performed, i.e. TC, PWR and ROW\_INFO shall be passed unmodified to the HRES sub-modules so that any threshold crossing shall be reported via T\_RESULTS[31:0] to the PC/Computer up to the capacity limit of the TREP sub-module (Figure 3‑11), even multiple crossings at different harmonics for the same seed\_f0 location.

When T\_FILTER\_EN = 1, only the highest harmonic of a seed\_f0 that results in a threshold crossing shall be reported via T\_RESULTS[31:0] to the PC/Computer up to the capacity limit of the TREP sub-module. Essentially this shall prevent the situation whereby multiple threshold crossings at different harmonics of a seed\_f0 are all recorded, which would effectively fill the limited reporting capacity of the TREP sub-module with duplicated information (in the sense that we only need to really identify the seed\_f0 and whether one or multiple harmonics have a threshold crossing is somewhat immaterial).

To clarify, it is the results from one seed\_f0 postulate (seed row) that are filtered, i.e. the results from up to 11 ambiguity slopes.

Since the summation results from the summing tree are staggered due to the pipelining between harmonics, it shall be necessary align the results with the last harmonic (from 1 to 16 dependent on H[3:0] for the analysis run) to determine the highest harmonic for the current ambiguity slope, and then consider up to 10 further results (dependent on A[3:0]) to determine the highest harmonic for the complete seed row.

T

T

T

Harmonic 1

Results

Harmonic 2

Results

Harmonic 3

Results

Harmonic *n*

Results

*n* = harmonic\_g

Delay equal to adder delay in SUMMER\_TREE

Select higher harmonic with threshold crossing

H[3:0]

Input Buffer

Max Harmonic

Output Buffer

Max Harmonic

A[3:0]

A[3:0] determines size of buffer.

Records highest harmonic as results are written to buffer.

LAST\_RESULT indicates when results should be transferred between buffers.

=

Harmonic 1

Results Out

Harmonic 2

Results Out

Harmonic 3

Results Out

Harmonic *n*

Results Out

LAST\_RESULT

**adder\_latency\_g, harmonic\_g**

SEED\_COL\_

SUM[21:0]

Pipeline Delay

SEED\_COL\_

SUM\_OUT

[21:0]

Figure ‑ : FILTER Sub-Module Processing when T\_FILTER\_EN =1

## HSUMMCI: HSUM Module Micro Configuration Interface

The HSUMMCI sub-module shall operate at the CLK\_MC domain.

The HSUMMCI sub-module shall connect to the MCI\_TOP module to support the writing of configuration information into the HSUMMCI sub-module by the PC/Computer via the PCIe interface (via MCDATAIN[31:0]), and read-back of configuration and status information from HSUMMCI to the PC/Computer via the PCIe interface (via MCDATAOUT[31:0]).

The HSUMMCI sub-module shall be parameterised by the “summer\_g” generic, to allow unique configuration of the required number of SUMMER sub-module instances.

The following configuration information shall be stored in D-type registers in the HSUMMCI sub-module with the configuration values being made available to the TGEN, DDRIN, SUMMER and TREP sub-modules:-

* A\_SET
* B\_START[1 to 2 analysis runs][21:0]
* B\_STOP[1 to 2 analysis runs][21:0]
* THRESH\_SET[1 to 2 analysis runs][21:0]
* H[1 to 2 analysis runs][3:0]
* P\_EN[1 to summer\_g SUMMER Instances][1 to 2 analysis runs][4:0]
* A[1 to summer\_g SUMMER Instances][1 to 2 analysis runs][3:0]
* M[31:0]
* T\_FILTER\_EN
* DM\_CNT\_RESET
* FOP\_COL\_OFFSET[8:0]

The following configuration shall be stored in RAMs in the HP\_SEL and T\_SEL sub-modules within the SUMMER sub-module via the HSUMMCI sub-module:-

* HPSEL[1 to summer\_g SUMMER Instances][1 to 2 analysis runs]

[1 to 16 harmonics][1 to 21 Orbital Acceleration seed\_f0 settings]

[1 to 11 Acceleration Ambiguity Slopes][6:0]

* TSEL[1 to summer\_g SUMMER Instances]

[1 to 2 threshold settings for low/high freq Periodicity Candidates]

[1 to 16 harmonics][1 to 21 Orbital Acceleration seed\_f0 settings][31:0]

The following status information in D-types in the TREP sub-module shall be sampled by the HSUMMCI sub-module when it is read-back to the PC/Computer via the PCIe interface:-

* DM\_CNT[31:0]

The following status information in RAMs in the TREP sub-module shall be sampled by the HSUMMCI sub-module when it is read-back to the PC/Computer via the PCIe interface:-

* T\_RESULT[1 to 2 analysis runs][1 to 16 harmonics]

[1 to 25 reports per harmonic][1 to 4 words][31:0]

* T\_EXC[1 to 2 analysis runs][1 to 16 harmonics][31:0]
* P\_EXC[1 to 2 analysis runs][1 to 16 harmonics][6:0]
* S\_EXC[1 to 2 analysis runs][1 to 16 harmonics][31:0]

To minimise the address decoding within the HSUMMCI sub-module the MCI\_TOP module shall supply an MCMS signal (active high) to indicate that the HSUMMCI sub-module has been selected.

Back-to-back accesses without an intermediate “idle” cycle shall be supported.

### Write to HSUMMCI

When MCMS = 1 and MCRWN = 0, if MCADDR[15:0] is a valid address of the memory map the value of MCDATAIN[31:0] shall be written. This operation shall be synchronous to CLK\_MC, with the storage address identified by MCADDR[15:0] being loaded with MCDATAIN[31:0] each clock cycle at which MCMS =1 and MCRWN = 0.

All internal stored values in D-type memory shall be set to 0 if the asynchronous reset “RST\_MC\_N” is asserted (active low).

The figure below shows a write access with the minimum access time that is required to store the data.

Figure ‑ : HSUMMCI Write Access Timing Diagram showing the minimum access time

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  | A |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  | D |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  | D |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

MCDATAIN[31:0]

MCADDR[15:0]

MCMS

Internal Storage at Location “A”

CLK\_MC

**DATAIN “D” stored at MCADDR “A” here, hence the access can complete on this cycle**

MCRWN

### Read from HSUMMCI

When MCMS = 1 and MCRWN = 1, if MCADDR[15:0] is a valid address of the memory map the value of MCDATAOUT[31:0] shall be set to the value of the targeted location. This operation shall be synchronous to CLK\_MC, with the targeted location identified by MCADDR[15:0] driving MCDATAOUT[31:0] each clock cycle at which MCMS =1 and MCRWN = 1.

If MCMS = 0 or MCADDR[15:0] is not a valid address of the memory map then MCDATAOUT[31:0] shall be set to 0x0000\_0000

The figure below shows a Read access with the minimum access time that is required to read the data.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  | A |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | D |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

MCDATAOUT[31:0]

MCADDR[15:0]

MCMS

CLK\_MC

**MCDATAOUT “D” from MCADDR “A” available here to be sampled.**

MCRWN

Figure ‑ : HSUMMCI Read Access Timing Diagram showing the minimum access time

## Reset Synchronisation: RESET\_SYNC Sub-Module

The RESET\_SYNC sub-module retimes and distributes the System Reset signal to reduce the signal loading and eliminate recovery timing warnings in the FDAS build.

The RESET\_SYNC sub-module retimes the system reset signal in six separate shift registers to support six reset signals to dstribute in the HSUM module.

# Design Requirement Tags

The relevant design requirement tags from the FDAS Design Specification Document for the HSUM module are listed below.

| **Design Requirement Tag** | **Description** | **Comment** |
| --- | --- | --- |
| FDAS.HARMONIC\_SUM:010/A | Each of the 84 post-convolution 222 freq-bin frequency domain sequences containing power information form part of the Filter Output Plane (FOP). | Definition of the FOP. The actual arrangement of the FOP data in a physical memory is further optimised as illustrated in Table 3‑2, Table 3‑3 and Table 3‑4 |
| FDAS.HARMONIC\_SUM:020/A | The original FDAS input 222 freq-bin frequency domain sequence also forms part of the FOP, with the contents of each freq-bin having been converted from complex amplitude to a power value. |
| FDAS.HARMONIC\_SUM:030/A | The sequences are arranged in relation to the convolution filter number “p” (-42, -41.. 0…41, 42) and freq-bin “b” (1 …. 4,194,304) with the original FDAS input sequence occupying the central row (i.e. p =0) of the FOP. |
| FDAS.HARMONIC\_SUM:035/A | It shall be possible to analyse the same FOP twice (mainly to facilitate reprocessing of the central FOP row p =0) with different user configuration for the summing parameters.  An MC configured value “A\_SET” shall determine if one analysis-run or two analysis-runs are to be performed. The relevant MC registers have two sets of configuration values, as indicated by the {a\_set} field to support the two different analysis-runs (where “a\_set” takes the values 1 and 2).  The time gap between the end of one analysis-run and the start of the second analysis-run shall be as short as possible (ideally no time gap). | Two Analysis runs per DM over different regions of the FOP are supported, with the number of seed\_f0 rows, harmonics and acceleration ambiguity slopes unique to each of the two analysis runs |
| FDAS.HARMONIC\_SUM:040/A | Harmonic summing over a globally user configurable freq-bin (b) range “B\_START {a\_set}[21:0]” to “B\_STOP {a\_set}[21:0]” and a globally configurable number of Filter Convolutions (p) “P\_EN{a\_set}[4:0]” up to a maximum of 21 Filters shall be carried out. Two harmonic summing analysis-runs on the same FOP with different configuration (via {a\_set}) shall be supported. . |
| FDAS.HARMONIC\_SUM:050/A | A globally user configurable number of harmonics “H {a\_set}[2:0]” up to a maximum of eight harmonics (f0, 2xf0, 3xf0 ….. 8xf0) shall be analysed during the harmonic summing process. Two harmonic summing analysis-runs on the same FOP with different configuration (via {a\_set}) shall be supported. |  |
| FDAS.HARMONIC\_SUM:055/A | A globally user configurable number of orbital acceleration ambiguity slopes “A {a\_set}[3:0]” up to a maximum of 11 slopes shall be used during the harmonic summing process. Two harmonic summing analysis-runs on the same FOP with different configuration (via {a\_set}) shall be supported. |
| FDAS.HARMONIC\_SUM:060/A | The FOP (SP0), Stretched Planes SP2 …. SP32 and Harmonic Planes H1… H32 may be seen as a way to visualise which terms need summing. | The stretched planes concept has been realised as a “summing tree” |
| FDAS.HARMONIC\_SUM:070/A | Harmonic summing function shall be required to select freq-bins from any of the 85 rows of the FOP, inferring that 85:1 muxing of 32-bit vectors shall be required. To support a default selection if the desired FOP row is greater than the FOP row range (+/- 42 rows) the mux shall actually be required to be 86:1. The value of the 86th 32-bit selected value shall be an MC configurable value “M [31:0]”. | Instead of Muxes, the address to RAMs shall be used to select the required FOP data for the harmonic summing. |
| FDAS.HARMONIC\_SUM:080/A | For correct operation the Harmonic Summing must start at the left side of the FOP (i.e. the low frequency end) and proceed for the different “seed\_f0” columns to the right side of the FOP (i.e. the high frequency end). | The HSUM module shall progress through the FOP from B\_START to B\_STOP. |
| FDAS.THRESHOLD:010/A | Threshold values will be supplied by software via the MC interface and written to internal FPGA memory for the FDAS function to access.  The number of different thresholds is based on the following:-   * Each “seed\_f0” row in its FOP column representing a different orbital acceleration (i.e. FOP row = p) shall require a different threshold level for itself and its harmonics (total of 21 FOP rows). * Each fundamental and its harmonics shall require a different threshold level (total of 8 thresholds). * Periodicity Candidates of a lower frequency shall require different thresholds than those of a higher frequency and it is deemed that two sets of thresholds shall be required to support this.   Hence in total 21 x 8 x 2 = 336 threshold values shall be required.  A “seed\_f0” freq-bin value “THRESH\_SET [21:0]” shall indicate the Periodicity Candidate frequency at which the thresholds shall change from one set to the other.  Each threshold value shall conform to the IEEE 754 32-bit single precision format. | These are the T[31:0] values which are stored in a separate RAM per harmonic in the SUMMER sub-module. |
| FDAS.THRESHOLD:020/A | FDAS shall indicate in internal FPGA memory a list of locations which exceed the threshold in terms:-   * Harmonic Number (f0, 2xf0, 3xf0 ….. 8xf0) * Filter Convolution Number “p” (-42, -41 … 0…. 41, 42) * Freq-bin Number “b” (1 …… 4,194,304) * Power Level of the location   Each the four values shall be in IEEE 754 32-bit single precision format. | In order to provide all the required information values shall be recorded which shall require translation in the PC/Computer. |
| FDAS.THRESHOLD:030/A | The list of locations that have crossed the threshold shall be of a finite size.  There shall be a defined limit to the number of threshold crossing locations that can be reported. The limit shall be 25 reports per harmonic. Since the same FOP can be analysed twice, results shall be provided for both analysis-runs.  i.e. total number of threshold reports is given by:-  No. of threshold reports = up to 2 analysis-runs  x (25 reports/harmonic)  x (8 harmonics)  = 400 reports  (each containing four IEEE 754 32-bit values)  If the reported list reaches its limit a counter for each harmonic (“T\_EXC{harmonic}[31:0]”) shall indicate the number of overflow reports. The “seed\_f0” location (“S\_EXC{harmonic}[28:0]”) and the filter number (p) (“P\_EXC{harmonic}[6:0]”) of the first overflowed threshold crossing shall be recorded against each harmonic. | The reports for a DM shall all be readable via the MC interface and shall remain static whilst the next DM is being processed. |
| FDAS.THRESHOLD:035/A | Ideally there shall be an MC configurable option (“T\_FILTER\_EN”) to store only the most significant threshold crossing for a particular “seed\_f0” postulate, since typically the threshold shall be crossed for a number of harmonics of “seed\_f0”, and it is sufficient to only store the threshold crossing for the highest harmonic. This shall require the 11-off acceleration ambiguity slopes for a “seed\_f0” to be completed before selection of the appropriate threshold crossings to be stored. | This shall ensure information that is effectively a duplicate is not using up the valuable storage capacity of the HSUM module. |
| FDAS.THRESHOLD:040/A | The output memory containing the threshold results shall be paged to allow one page to be statically available to software via the MC interface over the PCIe, whilst to other page is being written to by FDAS. | The reports for a DM shall all be readable via the MC interface and shall remain static whilst the next DM is being processed. |
| FDAS.THRESHOLD:050/A | A free-running internal 32-bit unsigned count (“DM\_CNT[31:0]”) which increments for each DM shall be provided as a header to identify each set of threshold results to software via the MC interface over the PCIe. The software shall have the ability to reset this count to zero via “DM\_CNT\_RESET”. | The DM\_CNT shall increment each time the HSUM module is triggered to process a DM. |

Table 4‑1 : HSUM Design Requirement Tags

# Interface Specification

| Signal | Direction | Clock Domain | Description |
| --- | --- | --- | --- |
| **Interface to CTRL Module** |  |  |  |
| HSUM\_TRIGGER | IN | CLK\_SYS | A 0 > 1 transition triggers the HSUM module to commence processing a DM |
| HSUM\_ENABLE | IN | CLK\_SYS | Enable the HSUM processing. The HSUM module only processes when HSUM\_ENABLE = 1. If HSUM\_ENABLE is set to 0 the processing in the HSUM module halts and only re-commences from its current position when HSUM\_ENABLE is set to 1. |
| HSUM\_PAGE [31:0] | IN | CLK\_SYS | Indication of the base address for the external DDR SDRAM to read the FOP from. |
| HSUM\_DONE | OUT | CLK\_SYS | Indication that the HSUM module has completed processing the DM. HSUM\_DONE shall pulse for a single cycle when the processing has completed. |
|  |  |  |  |
| **Interface to DDRIF2 Module** |  |  |  |
| DDR\_ADDR[31:0] | OUT | CLK\_SYS | Address to the DDRIF2 module (Byte granularity)  [31] = msb |
| DDR\_READ | OUT | CLK\_SYS | Read Enable to the DDRIF2 module to request data.  ‘1’ = Read |
| DDR\_DATA[512\*ddr\_g -1:0] | IN | CLK\_SYS | Freq-bin sample data from the external DDR SDRAM via the DDRIF2 module.  . |
| DATA\_VALID | IN | CLK\_SYS | Indication that the data from the DDRIF2 module is valid.  ‘1’ = valid |
| WAIT\_REQUEST | IN | CLK\_SYS | Indication from the DDRIF2 module to pause the Address generation to the DDRIF2 module.  ‘1’ = Wait. |
|  |  |  |  |
| **Interface to MCI\_TOP** |  |  |  |
| MCMS | IN | CLK\_MC | Module Select. When MCMS =1 the HSUM module is selected by the MCI\_TOP module. |
| MCADDR[17:0] | IN | CLK\_MC | Micro Configuration Address.  [17] = msb |
| MCRWN | IN | CLK\_MC | Micro Configuration Read Not Write.  ‘0’ = Write to HSUM.  ‘1’ = Read from HSUM. |
| MCDATAIN[31:0] | IN | CLK\_MC | Micro Configuration Data In from the MCI\_TOP module. This data will be written to address location MCADDR if MCCSN =0 and MCRWN = 0 |
| MCDATAOUT[31:0] | OUT | CLK\_MC | Micro Configuration Data Out to the MCI\_TOP module. This data will be valid from a valid MCADDR if MCMS =1  If MCMS = 0 or MCADDR is invalid then MCDATAOUT will be 0x00000000 |
|  |  |  |  |
| **Global Clock/Resets** |  |  |  |
| CLK\_SYS | IN | - | 200MHz Core System Clock |
| RST\_SYS\_N | IN | - | Asynchronous Logic reset for the CLK\_SYS domain  ‘0’=Reset |
| CLK\_MC | IN | - | MC Clock |
| RST\_MC\_N | IN | - | Asynchronous Logic reset for the CLK\_MC domain.  ‘0’=Reset |

Table 5‑1 : HSUM Pinlist

# MCI Memory Mapped Interface

The addresses in the following bitmap are byte addresses (relative to the module’s base address).

## FOP SELECTION CONFIGURATION

### FOP ROW (FILTER "P") SELECTION

Look up table for FOP rows to use in summing.

Array, size summer\_gx2x21x11x16 (max 3x2x21x11x16), indexed by 'SUMMER\_INSTANCE' (0 to summer\_g-1), 'ANALYSIS\_RUN' (0 to 1), 'SEED\_NUM' (0 to 20), 'ACC\_AMBIGUITY\_NUM' (0 to 10) and 'HARMONIC' (0 to 15). Address range 0x00000 to 0x00000+summer\_g\*65536-1 (max 0x00000 to 0x2FFFF).

Register: HPSEL (default: 0x00000000)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Address | Mode | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x00000\* | RW |  | HPSEL | | | | | | |

\*Address = 0x00000 + SUMMER\_INSTANCE \* 65536 + ANALYSIS\_RUN \* 32768 + SEED\_NUM \* 1024 + ACC\_AMBIGUITY\_NUM \* 64 + HARMONIC \* 4.

HPSEL[6:0]: The value programmed into the HPSEL register is the FOP Row Number using the encoding shown below.  
The special value 0x60 selects the user configurable value M[31:0] instead.  
(Note: FOP Row 1 is at the bottom and FOP Row 85 is at the top.)  
Default: 0.

|  |  |  |
| --- | --- | --- |
| Value | Label | Description |
| 0x00 | P[0] | FOP row 43(central) |
| 0x01 | P[-0] | FOP row 43(central) |
| 0x02 | P[1] | FOP row 44 |
| 0x03 | P[-1] | FOP row 42 |
| 0x04 | P[2] | FOP row 45 |
| 0x05 | P[-2] | FOP row 41 |
| 0x06 | P[3] | FOP row 46 |
| 0x07 | P[-3] | FOP row 40 |
| 0x08 | P[4] | FOP row 47 |
| 0x09 | P[-4] | FOP row 39 |
| 0x0A | P[5] | FOP row 48 |
| 0x0B | P[-5] | FOP row 38 |
| 0x0C | P[6] | FOP row 49 |
| 0x0D | P[-6] | FOP row 37 |
| 0x0E | P[7] | FOP row 50 |
| 0x0F | P[-7] | FOP row 36 |
| 0x12 | P[8] | FOP row 51 |
| 0x13 | P[-8] | FOP row 35 |
| 0x14 | P[9] | FOP row 52 |
| 0x15 | P[-9] | FOP row 34 |
| 0x16 | P[10] | FOP row 53 |
| 0x17 | P[-10] | FOP row 33 |
| 0x18 | P[11] | FOP row 54 |
| 0x19 | P[-11] | FOP row 32 |
| 0x1A | P[12] | FOP row 55 |
| 0x1B | P[-12] | FOP row 31 |
| 0x1C | P[13] | FOP row 56 |
| 0x1D | P[-13] | FOP row 30 |
| 0x1E | P[14] | FOP row 57 |
| 0x1F | P[-14] | FOP row 29 |
| 0x22 | P[15] | FOP row 58 |
| 0x23 | P[-15] | FOP row 28 |
| 0x24 | P[16] | FOP row 59 |
| 0x25 | P[-16] | FOP row 27 |
| 0x26 | P[17] | FOP row 60 |
| 0x27 | P[-17] | FOP row 26 |
| 0x28 | P[18] | FOP row 61 |
| 0x29 | P[-18] | FOP row 25 |
| 0x2A | P[19] | FOP row 62 |
| 0x2B | P[-19] | FOP row 24 |
| 0x2C | P[20] | FOP row 63 |
| 0x2D | P[-20] | FOP row 23 |
| 0x2E | P[21] | FOP row 64 |
| 0x2F | P[-21] | FOP row 22 |
| 0x32 | P[22] | FOP row 65 |
| 0x33 | P[-22] | FOP row 21 |
| 0x34 | P[23] | FOP row 66 |
| 0x35 | P[-23] | FOP row 20 |
| 0x36 | P[24] | FOP row 67 |
| 0x37 | P[-24] | FOP row 19 |
| 0x38 | P[25] | FOP row 68 |
| 0x39 | P[-25] | FOP row 18 |
| 0x3A | P[26] | FOP row 69 |
| 0x3B | P[-26] | FOP row 17 |
| 0x3C | P[27] | FOP row 70 |
| 0x3D | P[-27] | FOP row 16 |
| 0x3E | P[28] | FOP row 71 |
| 0x3F | P[-28] | FOP row 15 |
| 0x42 | P[29] | FOP row 72 |
| 0x43 | P[-29] | FOP row 14 |
| 0x44 | P[30] | FOP row 73 |
| 0x45 | P[-30] | FOP row 13 |
| 0x46 | P[31] | FOP row 74 |
| 0x47 | P[-31] | FOP row 12 |
| 0x48 | P[32] | FOP row 75 |
| 0x49 | P[-32] | FOP row 11 |
| 0x4A | P[33] | FOP row 76 |
| 0x4B | P[-33] | FOP row 10 |
| 0x4C | P[34] | FOP row 77 |
| 0x4D | P[-34] | FOP row 9 |
| 0x4E | P[35] | FOP row 78 |
| 0x4F | P[-35] | FOP row 8 |
| 0x52 | P[36] | FOP row 79 |
| 0x53 | P[-36] | FOP row 7 |
| 0x54 | P[37] | FOP row 80 |
| 0x55 | P[-37] | FOP row 6 |
| 0x56 | P[38] | FOP row 81 |
| 0x57 | P[-38] | FOP row 5 |
| 0x58 | P[39] | FOP row 82 |
| 0x59 | P[-39] | FOP row 4 |
| 0x5A | P[40] | FOP row 83 |
| 0x5B | P[-40] | FOP row 3 |
| 0x5C | P[41] | FOP row 84 |
| 0x5D | P[-41] | FOP row 2 |
| 0x5E | P[42] | FOP row 85 |
| 0x5F | P[-42] | FOP row 1 |
| 0x60 | M | User configurable value M[31:0] instead of a FOP Row |

### THRESHOLD CONFIGURATION

Power threshold values for each summer instance for each harmonic and for each of 21 different seed\_f0 row locations within a column of the FOP.  
There are two sets of Thresholds, one for low frequency Periodicity Candidates (T\_SET = 0) and one for high frequency Periodicity Candidates (T\_SET = 1).

Array, size summer\_gx2x21x16 (max 3x2x21x16), indexed by 'SUMMER\_INSTANCE' (0 to summer\_g-1), 'T\_SET' (0 to 1), 'SEED\_NUM' (0 to 20) and 'HARMONIC' (0 to 15). Address range 0x30000 to 0x30000+summer\_g\*4096-1 (max 0x30000 to 0x32FFF).

Register: THRESHOLD (default: 0x00000000)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Address | Mode | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
| 0x30000\* | RW | THRESHOLD[31:24] | | | | | | | |
| 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| THRESHOLD[23:16] | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| THRESHOLD[15:8] | | | | | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| THRESHOLD[7:0] | | | | | | | |

\*Address = 0x30000 + SUMMER\_INSTANCE \* 4096 + T\_SET \* 2048 + SEED\_NUM \* 64 + HARMONIC \* 4.

THRESHOLD[31:0]: Power Level Threshold at or above which the summed power at the associated location of the SUMMER "summing tree" indicates a potential existence of a Periodicity Candidate. The value is in IEEE 754 32-bit single precision format.  
Default: 0.

## ANALYSIS RUN PARAMETERS

### GLOBAL PARAMETERS

Parameters common to all SUMMERs.

Register: B\_START\_1 (default: 0x00000000)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Address | Mode | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| 0x34000 | RW |  |  | B\_START\_1[21:16] | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| B\_START\_1[15:8] | | | | | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| B\_START\_1[7:0] | | | | | | | |

Register: B\_STOP\_1 (default: 0x00000000)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Address | Mode | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| 0x34004 | RW |  |  | B\_STOP\_1[21:16] | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| B\_STOP\_1[15:8] | | | | | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| B\_STOP\_1[7:0] | | | | | | | |

Register: B\_START\_2 (default: 0x00000000)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Address | Mode | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| 0x34008 | RW |  |  | B\_START\_2[21:16] | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| B\_START\_2[15:8] | | | | | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| B\_START\_2[7:0] | | | | | | | |

Register: B\_STOP\_2 (default: 0x00000000)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Address | Mode | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| 0x3400C | RW |  |  | B\_STOP\_2[21:16] | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| B\_STOP\_2[15:8] | | | | | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| B\_STOP\_2[7:0] | | | | | | | |

Register: H\_1 (default: 0x00000000)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Address | Mode | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| 0x34010 | RW |  |  |  |  | H\_2 | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|  |  |  |  | H\_1 | | | |

Register: FOP\_ROW\_1 (default: 0x00000055)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Address | Mode | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| 0x34014 | RW |  | FOP\_ROW\_2 | | | | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|  | FOP\_ROW\_1 | | | | | | |

Register: FOP\_COL\_OFFSET (default: 0x000000D2)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Address | Mode | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| 0x34018 | RW |  |  |  |  |  |  |  | FOP\_COL\_OFFSET[8] |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| FOP\_COL\_OFFSET[7:0] | | | | | | | |

Register: A\_SET (default: 0x00000000)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Address | Mode | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x34020 | RW |  |  |  |  |  |  |  | A\_SET |

Register: THRESH\_SET (default: 0x00000000)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Address | Mode | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| 0x34024 | RW |  |  | THRESH\_SET[21:16] | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| THRESH\_SET[15:8] | | | | | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| THRESH\_SET[7:0] | | | | | | | |

Register: M (default: 0x00000000)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Address | Mode | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
| 0x34028 | RW | M[31:24] | | | | | | | |
| 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| M[23:16] | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| M[15:8] | | | | | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| M[7:0] | | | | | | | |

Register: T\_FILTER\_EN (default: 0x00000000)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Address | Mode | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x34030 | RW |  |  |  |  |  |  |  | T\_FILTER\_EN |

B\_START\_1[21:0]: The first FOP column containing a seed\_f0 analysed in the first analysis run of the DM. Default: 0.

B\_STOP\_1[21:0]: The last FOP column containing a seed\_f0 analysed in the first analysis run of the DM. Default: 0.

B\_START\_2[21:0]: The first FOP column containing a seed\_f0 analysed in the second analysis run of the DM. Default: 0.

B\_STOP\_2[21:0]: The last FOP column containing a seed\_f0 analysed in the second analysis run of the DM. Default: 0.

H\_1[3:0]: The number of harmonics analysed in the first analysis run of the DM. ("0000" = 1 harmonic analysed, "1111" = 16 harmonics analysed.) Default: 0.

H\_2[3:0]: The number of harmonics analysed in the second analysis run of the DM. Default: 0.

FOP\_ROW\_1[6:0]: The number of FOP rows to read from DDR for each column for the first analysis run. Valid values are 1 to 85. Default: 85.

FOP\_ROW\_2[6:0]: The number of FOP rows to read from DDR for each column for the second analysis run. Valid values are 1 to 85. Default: 0.

FOP\_COL\_OFFSET[8:0]: First valid column of FOP data. Default: 210.

A\_SET: The number of analysis runs performed on a DM. '0' = 1 analysis run performed, '1' = 2 analysis runs performed. Default: 0.

THRESH\_SET[21:0]: The FOP column number at which the power thresholds used in the harmonic summing change from the "Low Periodicity Candidate Frequency" set to the "High Periodicity Candidate Frequency Set". Default: 0.

M[31:0]: If the HPSEL value for a harmonic is set to 0x60 then instead of FOP location values being passed to the SUMMER sub-module, this value M[31:0] is passed instead. Default: 0.

T\_FILTER\_EN: If set to 1 then only the results from the highest harmonic are stored for a particular seed\_f0 run. Default: 0.

FOP Column Numbering:  
0x000000 = FOP Column 1  
0x000001 = FOP Column 2  
:  
0x3FFFFF = FOP Column 4,194,304

### PER SUMMER RUN PARAMETERS

Array, size summer\_g (max 3), indexed by 'SUMMER\_INSTANCE' (0 to summer\_g-1). Address range 0x34040 to 0x34040+summer\_g\*8-1 (max 0x34040 to 0x34057).

Register: P\_EN\_1 (default: 0x00000000)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Address | Mode | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| 0x34040\* | RW |  |  |  | P\_EN\_2 | | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|  |  |  | P\_EN\_1 | | | | |

\*Address = 0x34040 + SUMMER\_INSTANCE \* 8.

Register: A\_1 (default: 0x00000000)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Address | Mode | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| 0x34044\* | RW |  |  |  |  | A\_2 | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|  |  |  |  | A\_1 | | | |

\*Address = 0x34044 + SUMMER\_INSTANCE \* 8.

P\_EN\_1[4:0]:   
The number of orbital accelerations (i.e. seed\_f0s in a FOP column) analysed in the first analysis run of the DM.  
Default: 0.

P\_EN\_2[4:0]:   
The number of orbital accelerations (i.e. seed\_f0s in a FOP column) analysed in the second analysis run of the DM.  
"00000" = 1 orbital acceleration analysed  
"00001" = 2 orbital accelerations analysed  
:  
"10100" = 21 orbital accelerations analysed  
Default: 0.

A\_1[3:0]:   
The number of orbital acceleration ambiguity slopes analysed in the first analysis run of the DM.  
Default: 0.

A\_2[3:0]:   
The number of orbital acceleration ambiguity slopes analysed in the second analysis run of the DM.  
"0000" = 1 orbital acceleration ambiguity slope analysed  
"0001" = 2 orbital acceleration ambiguity slopes analysed  
:  
"1010" = 11 orbital acceleration ambiguity slopes analysed Default: 0.

### DM COUNTER

Register: DM\_CNT

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Address | Mode | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
| 0x34080 | RO | DM\_CNT[31:24] | | | | | | | |
| 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| DM\_CNT[23:16] | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| DM\_CNT[15:8] | | | | | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| DM\_CNT[7:0] | | | | | | | |

Register: DM\_CNT\_RESET (default: 0x00000000)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Address | Mode | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x34084 | RW |  |  |  |  |  |  |  | DM\_CNT\_RESET |

DM\_CNT[31:0]: Free Running count of the DMs that have been processed by HSUM.   
DM\_CNT can be reset to zero by writing to DM\_CNT\_RESET ('0' to '1' transition).

DM\_CNT\_RESET: A 0 to 1 transition resets the free running DM Counter which can be used to identify which DM the Threshold Crossing results belong to. Default: 0.

## SUMMING RESULTS

### HARMONIC THRESHOLD CROSSING RESULTS

Two pages of 25-off Results per harmonic per Analysis Run, each consisting of 4-off 32-bit words.  
The first page contains the static results from the previous run. The second page contains the results for the current run which will change as the run progresses, but may be accessed for debug purposes if the processing is paused.

Array, size 2x2x16x25, indexed by 'PAGE' (0 to 1), 'ANALYSIS\_RUN' (0 to 1), 'HARMONIC' (0 to 15) and 'HARMONIC\_REPORT\_NUM' (0 to 24). Address range 0x40000 to 0x47FFF.

Register: HARMONIC

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Address | Mode | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| 0x40000\* | RO |  |  |  |  |  |  |  | VALID |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|  |  |  |  | HARMONIC | | | |

\*Address = 0x40000 + PAGE \* 16384 + ANALYSIS\_RUN \* 8192 + HARMONIC \* 512 + HARMONIC\_REPORT\_NUM \* 16.

Register: FOPCOL

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Address | Mode | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| 0x40004\* | RO |  |  | FOPCOL[21:16] | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| FOPCOL[15:8] | | | | | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| FOPCOL[7:0] | | | | | | | |

\*Address = 0x40004 + PAGE \* 16384 + ANALYSIS\_RUN \* 8192 + HARMONIC \* 512 + HARMONIC\_REPORT\_NUM \* 16.

Register: FOPROW

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Address | Mode | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x40008\* | RO |  | FOPROW | | | | | | |

\*Address = 0x40008 + PAGE \* 16384 + ANALYSIS\_RUN \* 8192 + HARMONIC \* 512 + HARMONIC\_REPORT\_NUM \* 16.

Register: PWR

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Address | Mode | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
| 0x4000C\* | RO | PWR[31:24] | | | | | | | |
| 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| PWR[23:16] | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| PWR[15:8] | | | | | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| PWR[7:0] | | | | | | | |

\*Address = 0x4000C + PAGE \* 16384 + ANALYSIS\_RUN \* 8192 + HARMONIC \* 512 + HARMONIC\_REPORT\_NUM \* 16.

VALID: Indicates this result record is valid.  
Valid records will be first followed by invalid records. They will not be mixed.

|  |  |  |
| --- | --- | --- |
| Value | Label | Description |
| 0 | INVALID |  |
| 1 | VALID |  |

HARMONIC[3:0]: Harmonic Number of the Threshold Crossing Location. (Note that the report address also inherently provides the harmonic).

FOPCOL[21:0]: FOP Column [b] of the Threshold Crossing Location.

FOPROW[6:0]: FOP Row [p] of the Threshold Crossing Point encoded with the same mapping as HPSEL.

PWR[31:0]: The summed power at the Threshold Crossing point in IEEE 754 Format.

### STORAGE EXCEEDED REPORTS

Information regarding threshold crossing reports that could not be stored. This information is presented on a per harmonic per analysis basis.  
There are two pages. The first page contains the static results from the previous run. The second page contains the results for the current run which will change as the run progresses, but may be accessed for debug purposes if the processing is paused.

Array, size 2x2x16, indexed by 'PAGE' (0 to 1), 'ANALYSIS\_RUN' (0 to 1) and 'HARMONIC' (0 to 15). Address range 0x80000 to 0x803FF.

Register: T\_EXC

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Address | Mode | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
| 0x80000\* | RO | T\_EXC[31:24] | | | | | | | |
| 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| T\_EXC[23:16] | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| T\_EXC[15:8] | | | | | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| T\_EXC[7:0] | | | | | | | |

\*Address = 0x80000 + PAGE \* 512 + ANALYSIS\_RUN \* 256 + HARMONIC \* 16.

Register: S\_EXC

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Address | Mode | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| 0x80004\* | RO |  |  | S\_EXC[21:16] | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| S\_EXC[15:8] | | | | | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| S\_EXC[7:0] | | | | | | | |

\*Address = 0x80004 + PAGE \* 512 + ANALYSIS\_RUN \* 256 + HARMONIC \* 16.

Register: P\_EXC

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Address | Mode | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x80008\* | RO |  | P\_EXC | | | | | | |

\*Address = 0x80008 + PAGE \* 512 + ANALYSIS\_RUN \* 256 + HARMONIC \* 16.

T\_EXC[31:0]: The number of threshold crossing results that could not be recorded.

S\_EXC[21:0]: FOP Column number of the seed\_f0 for the first threshold crossing that could not be recorded.

P\_EXC[6:0]: FOP Row [p] for the first threshold crossing that could not be recorded, encoded using the same mapping as HPSEL.

# Design Parameterisation

The HSUM module has the following generic parameters:

HSUM Module Top Level Generics

| **Generic** | **Description** |
| --- | --- |
| ddr\_g | The number of external DDR SDRAM Interfaces.  Valid values:  1 = 1 DDR SDRAM Interface  2 = 2 DDR SDRAM Interface  3 = 3 DDR SDRAM Interface |
| summer\_g | The number of SUMMER sub-module instances  1 = 1 SUMMER sub-module  2 = 2 SUMMER sub-modules  3 = 3 SUMMER sub-modules |
| adder\_latency\_g | The number of clock cycles required to perform IEEE-754 floating point addition.  Valid range is 1 to 7. |
| harmonic\_g | The maximum number of harmonics supported.  Valid range is 8 to 16.  If it is not possible to fit or time-out the design in the FPGA, this generic provides a simple mechanism for reducing the design size and complexity.  **Note that currently the Enhanced SUMMER\_TREE is only capable of supporting 12 harmonics (i..e harmonic\_g = 12 maximum). Further design work shall be required to increase the Enhanced SUMMER\_TREE to support 16 harmonics.** |

Table 7‑1: HSUM Generic Parameters

The generic values are type natural (i.e. +ve integers).

Settings for one DDR Interface and one SUMMER instance and ALTERA DSP floating point adder and up to 12 harmonics:-

| **Generic** | **Value** |
| --- | --- |
| ddr\_g | 1 |
| summer\_g | 1 |
| adder\_latency\_g | 2 |
| harmonic\_g | 12. **This is the maximum the Enhanced SUMMER\_TREE is currently designed to support.** |

Table ‑ : Example of settings

# Updates for Intel Agilex Implementation

## Floating Point DSP Adder

The Floating Point DSP Adder IP has been updated for Intel Agilex in Quartus Prime 21.3. The settings to create the Floating Point DSP Adder are:-

The DSP Adder is generated using the Quartus Prime IP Catalog.

The “IP Catalog” shall be used with the following selected:-

DSP > Primitive DSP > Native Floating Point DSP Intel Agilex FPGA IP

This launches IP Platform Designer” which then asks for a name for the IP Variation which shall be entered as “fp\_add”

The settings are as follows:-

**General Tab**

**Choose the operation mode:** fp32\_add

**Enable fp32\_chainin:** No

**Enable fp\_chainout:** No

**Perform subtraction in fp32\_adder:** No

**Select the mode for fp16:** FLUSHED

**Select the width size for fp16 (only for bfloat16 mode):** 16

**Perform subtraction in fp16\_adder:** No

**Enable exception flag:** No

**Registers Tab**

**Type of clear signal:** aclr

**Enable clr0 signal for all input registers:** Yes

**Enable clr1 for output and pipeline registers:** Yes

**Enable for input “accumulate”:** no\_reg

**Enable for “fp32\_adder\_a”:** ena0

**Enable for “fp32\_adder\_b”:** ena0

**Enable for “fp32\_mult\_a”:** no\_reg

**Enable for “fp32\_mult\_b”:** no\_reg

**Enable for “16\_mult\_input”:** no\_reg

**Enable output register:** ena0

**Enable “accum\_adder” register:** no\_reg

**Enable “adder\_input” register:** no\_reg

**Enable “adder\_pl” register:** no\_reg

**Enable “fp32\_adder\_a\_chainin\_pl” register:** no\_reg

**Enable “accum\_pipeline” register:** no\_reg

**Enable “mult\_pipeline” register:** no\_reg

**Enable “fp32\_adder\_a\_chainin\_2nd\_pl” register:** no\_reg

**Enable “accum\_2nd\_pipeline” register:** no\_reg

**Enable “mult\_2nd\_pipeline” register:** no\_reg

This populates all the necessary Floating Point Adder settings. This can then be saved and the HDL can be generated (click “Generate HDL” button) ensuring that the selected language is VHDL.

### Instantiation of the Floating Point Adder

This adder design can then be instantiated in the hsumadder\_synth.vhd file which acts as a wrapper for the floating point adder.

## Replacement of Intel altsyncram

The Intel altsyncram in the Agilex family is not able to support different clock frequencies on the A and B ports. Hence the hsumhpselram\_synth.vhd and hsumtselram\_synth.vhd files have been re-coded to use an inferred RAM.

# Abbreviations and Acronyms

|  |  |
| --- | --- |
| DM | Dispersion Measure |
| DMA | Direct Memory Access |
| DDR | Double Data Rate |
| FDAS | Fourier Domain Acceleration Search |
| FFT | Fast Fourier Transform |
| FOP | Filter Output Plane |
| FPGA | Field Programmable Gate Array |
| IEEE | Institute of Electrical Engineers |
| IP | Intellectual Property |
| MC | Micro Controller Interface |
| MSI-X | Extended Message Signalled Interrupt |
| PCIe | Peripheral Component Interconnect Express |
| SDRAM | Synchronous Dynamic RAM |