

SKA Project Series  
LFAA Tile Beamformer structure

G. Comoretto<sup>1</sup>, C. Belli<sup>1</sup>

<sup>1</sup>INAF - Osservatorio Astrofisico di Arcetri

**Arcetri Technical Report N° 2/2015**  
**05-dec-2015**

### **Abstract**

The LFAA tile processing modules (TPM) combine the signals from 256 dual-polarization antennas, in the frequency range 50-350 MHz, into a single *station beam*. Beamforming is done hierarchically, with the TPM first combining the 16 antennas that are processed in the module, using the Tile Beamformer described here, and then combining these partial beams into the station beam, using a daisy chain structure.

The Tile Beamformer receives antenna samples already channelized, with a channel spacing of 750 KHz, and aligns each antenna in the frequency domain using a phasor computed in real time from delay and delay rate information. It is possible to select up to 16 independent portions of the input frequency range, each one pointing in a different direction.

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	TPM Firmware structure . . . . .	3
1.2	Specifications . . . . .	5
1.3	Interfaces . . . . .	5
<b>2</b>	<b>Beamformer processing and parameters</b>	<b>6</b>
2.1	Input format . . . . .	7
2.2	Calibration unit I/O format . . . . .	8
2.3	Output format . . . . .	8
2.4	Sub-band specification . . . . .	9
2.5	Control parameters to the TPM . . . . .	9
2.5.1	Example of region definition . . . . .	10
2.6	Region selector . . . . .	10
2.7	Frequency domain beamformer . . . . .	11
2.8	Antenna calibration . . . . .	12
2.9	Beam adder . . . . .	12
2.9.1	FPGA-to-FPGA interface . . . . .	12
<b>3</b>	<b>Programming interface</b>	<b>13</b>
3.1	General control registers . . . . .	13
3.2	Sub-band specification registers . . . . .	14
3.3	Delay, delay offset and tapering tables . . . . .	15

## List of Tables

1	Partition of frequency channels between the two FPGAs and the output bus streams . . . . .	9
2	Parameters for frequency domain delay correction . . . . .	13
3	Programming interface for the beamformer . . . . .	14

## List of Figures

1	General structure of the tile processing module (2 FPGAs) . . . . .	3
2	SysML Block Definition Diagram for the TPM firmware in each FPGA . . . . .	4
3	SysML Internal Block Diagram for the TPM firmware in each FPGA . . . . .	4
4	Tile Beamforming process . . . . .	6
5	Tile Beamforming block definition diagram . . . . .	7
6	Tile Beamforming internal block diagram . . . . .	8
7	Input reorder logic for the region selection memory . . . . .	11
8	Delay and delay rate calculation circuitry . . . . .	12
9	Tile beam adder . . . . .	13
10	Multiplexing of 4 samples into 3 words over the f2f bus . . . . .	14
11	Conceptual schematics for the f2f bus multiplexer and demultiplexer . . . . .	14

## List of acronyms

**ADC:** Analog to Digital Converter  
**BDD:** SysML Block Definition Diagram  
**ASIC:** Application Specific Integrated Circuit  
**COTS:** Commercial Off-The-Shelf  
**CSP:** Central Signal Processor  
**DDR3/DDR4:** Double Data Rate 3(4) memory standard  
**DSP:** Digital Signal Processing  
**EMC:** Electromagnetic Compatibility  
**EMI:** Electromagnetic Interface  
**FFT:** Fast Fourier Transformation  
**FPGA:** Field Programmable Gate Array  
**Gb:** Giga bit  
**GB:** Giga byte  
**GPU:** General Processing Unit  
**HDL:** High Level Design Language  
**HMC:** Hyper Memory Cube memory standard  
**IBD:** SysML Internal Block Diagram  
**ICD:** Interface Control Document  
**IICD:** Internal Interface Control Document  
**INAF:** National Institute for Astrophysics  
**I/O:** Input/Output  
**IP:** Intellectual Property  
**LFAA:** Low Frequency Aperture Array Element or Consortium  
**LFA:** Low Frequency Array  
**LMC:** Local Monitor and Control  
**MATLAB:** MATLAB simulation language and application  
**M&C:** Monitor and Control  
**PFB:** Polyphase Filter Bank  
**PPS:** Peak per Second  
**RFI:** Radio Frequency Interference  
**RS:** Requirement Specification  
**SKA:** Square Kilometre Array  
**SKAO:** SKA Organisation (or office)  
**SW:** Software  
**SYSML:** System Engineering Simulation Language and application  
**Tb:** Tera bit  
**TB:** Tera byte  
**TBC:** To be confirmed  
**TBD:** To be decided  
**TM:** Telescope manager  
**TPM:** Tile processing module  
**UDP:** User Datagram Protocol  
**UML:** Unified Modelling Language  
**WBS:** Work Breakdown Structure

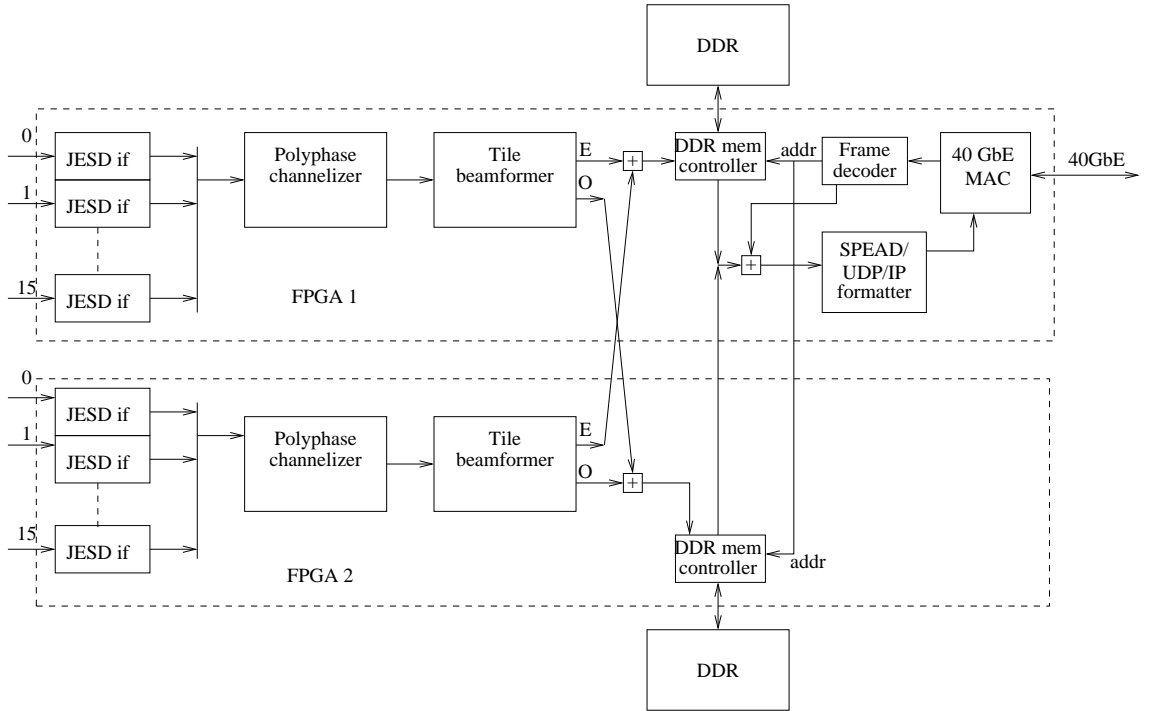


Figure 1: General structure of the tile processing module (2 FPGAs)

## List of symbols

$N_c$  Number of spectral points

$N_a$  Number of antennas

$\tau$  Geometric delay with respect to the station reference point

$f(k)$  Frequency of channel  $k$

$\phi$  Phase

## 1 Introduction

The LFAA tile beamformer is a component of the Tile Processing module (TPM). It processes channelised and calibrated data from 8 antennas, 2 polarisations, in order to produce a half-tile beam. Two identical tile beamformers are present in the two TPM FPGAs, and interchange part of the processed channels to produce a full tile (16 antennas) beam. Each FPGA then further processes half of the bandwidth to produce the full station beam. The global structure to perform this complete beamforming is shown in figure 1.

In this report the tile beamforming process is described in detail. The overall tile signal processing structure is summarised in section 1.1. The beamformer specifications are listed in chapter 1.2, together with some added assumptions.

### 1.1 TPM Firmware structure

The general structure of the TPM data processing architecture is shown in the block definition diagram in figure 2, and the relative data processing flow in the internal block diagram in figure 3.

Data from the ADCs are channelised into 512 channels over the input bandwidth. For an ADC clock frequency of 400 MHz, channel spacing is thus 781.25 kHz. The channeliser operates with a 32/27 oversampling factor, producing one output sample for each frequency channel (one channeliser frame) every 864 input ADC samples, or 1080 ns. Channelised samples are calibrated, to remove instrumental effects (including spurious cross polarisation), and to equalise the signal level, and finally requantised to 8+8 bit complex samples.

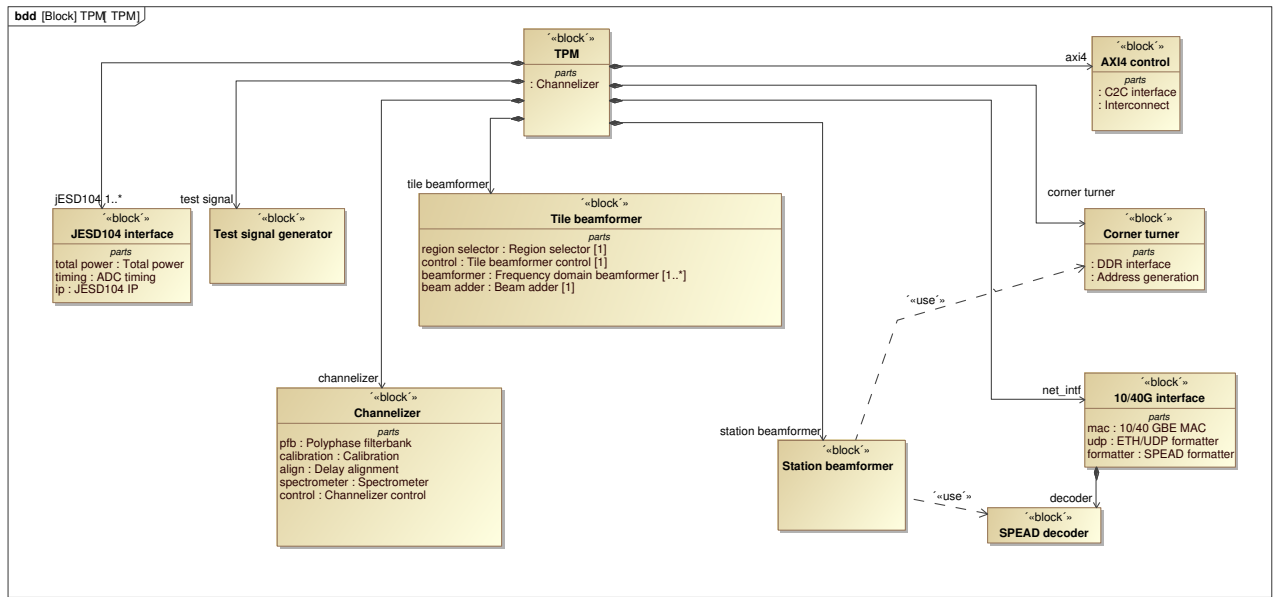


Figure 2: SysML Block Definition Diagram for the TPM firmware in each FPGA

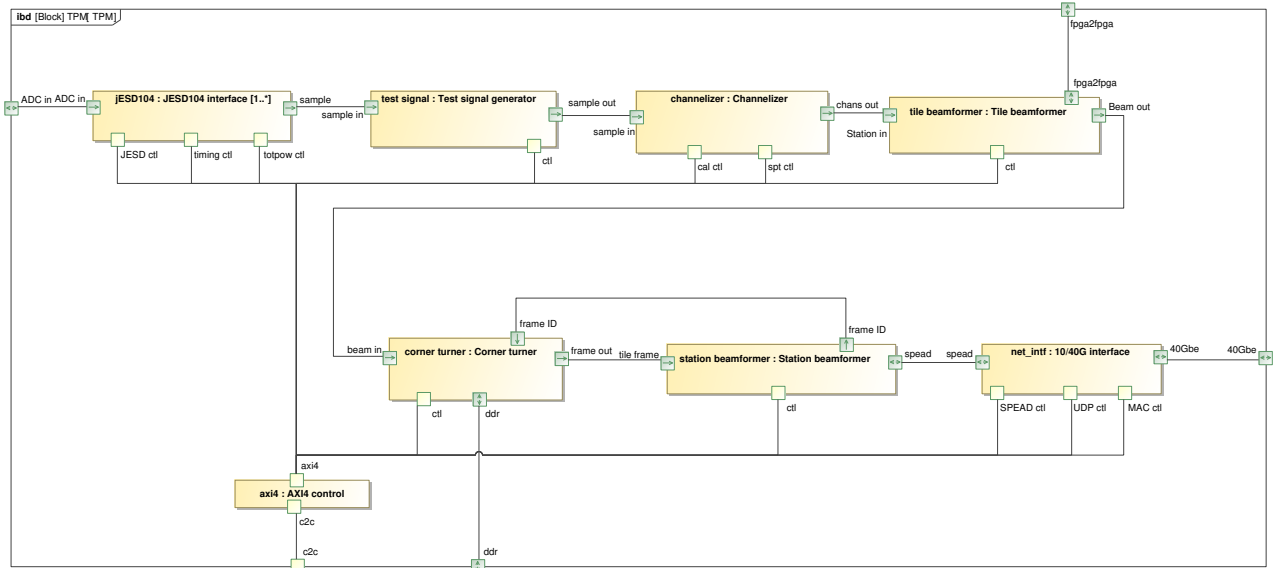


Figure 3: SysML Internal Block Diagram for the TPM firmware in each FPGA

The beamformer extracts a number of *beamformer regions* or sub-bands (the two terms are used interchangeably in this report) from each channeliser frame, applies a delay (a phase slope in the frequency domain) and tapering to each antenna signal, and sums 8 antenna signals together.

The beamformed frame is composed of two parallel data streams, respectively for odd and even channels. These two outputs are combined with the respective signals from the other 8 antennas, using the FPGA-to-FPGA internal bus. In this way the two FPGAs will contain the tile beamformed data for odd and even beamformed channels respectively, and no other interconnections between the two FPGAs are necessary.

The tile output is sent to the DDR memory, and retrieved in the correct sequence for the CSP input frames. These frames are sent to the other TPMs, and added together to form the station beam.

When a packet is received by the TPM, the corresponding packet is looked for in DDR memory, co-added with the received packet, and the result sent to the next TPM in the chain. In this way the amount of buffering required to compensate for daisy chain delay is kept to a minimum.

## 1.2 Specifications

SKA-low specifications require the beamformer to produce:

- up to 16 independently tunable frequency regions (sub-bands), placed anywhere in the digitised band
- up to 8 independent beams, placed anywhere in the sky
- Each frequency region is associated to one beam
- the total bandwidth, over all frequency regions and beams is limited to 300 MHz

This means that the beamformer can select up to 16 sub-bands. Sub-bands may overlap, repeat, or be totally disjointed. Each sub-band can be beamformed to a different beam centre. Therefore the beamformer must be able to select groups of frequency channels in a very flexible way.

To simplify the control structure, we added the following limitations:

- The basic resolution element of the beamformer is one coarse frequency channel. This is exactly 1/512 of the ADC Nyquist bandwidth, e.g. 781.25 kHz for an ADC clocked at 800 MHz (Nyquist band = 0–400 MHz).
- The position of each sub-band can be tuned with a resolution of two frequency channels (1.5625 MHz).
- To simplify DDR memory organisation, each sub-band is composed of a multiple of 8 frequency channels (6.25 MHz). This constraint can be reduced to a multiple of 2 frequency channels with relatively little effort.

No other limitations are given by the hardware. It is possible to have overlapping sub-bands, disjointed sub-bands, repeated identical sub-bands in different beams, etc.

The beamforming is done by applying a phase factor proportional to the channelised sample frequency and to the geometric delay of the antenna. An amplitude tapering can also be applied. The amplitude tapering is assumed to be constant for one integration.

Timing structure operates at three time scales. The observation is divided into integrations, typically corresponding to one scheduling block. Observed channels, bandwidth, assignment of frequency regions to beams are done before the start of the observation, and are not synchronised. Integration start and stop times are synchronous among TPMs.

Calibration is performed on a timescale of nominally 10 minutes (600 s). Calibration coefficients, amplitude tapering, polarisation correction matrix is updated on this timescale, using a dual bank scheme. Coefficients are loaded asynchronously in the inactive bank, and switching between old and new coefficient banks occurs synchronously among TPMs. The delay may vary during one integration. For a maximum distance of 18 m from the station centre, the delay rate is  $\approx 4.4 \cdot 10^{-12}$ , corresponding to a phase error of 4.3 milliradians at the edges of a 0.9 s integration time at a sky frequency of 350 MHz. The resulting decorrelation is  $3 \cdot 10^{-6}$ , that is negligible, but updating all coefficients at a rate of about 1 second requires a significant bandwidth in the LMC control network. To simplify control, however, the beamformer has the capability to track a constant delay rate (linearly varying delay), allowing an update period of about 15 seconds. The update of delay and delay rate will occur synchronously, at predetermined times.

## 1.3 Interfaces

The beamformer interfaces with:

- The general data processing clock, with a frequency of 237 MHz or greater. A frequency of 240 MHz can be assumed for timing closure calculation. All data interfaces use this clock.

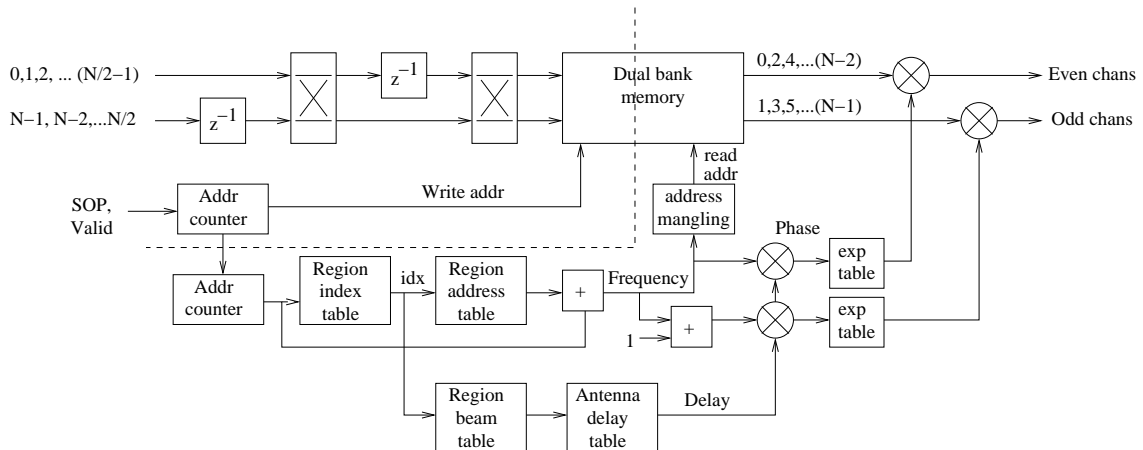


Figure 4: Tile Beamforming process

- The channeliser, using a AXI4 streaming interface. Each frame contains one time sample, for all antennas, polarisations and frequencies. The only control signals employed are *tvalid* and *tlast*. Frame size is fixed to 256 valid clock cycles per frame. No back-pressure is used
- The FPGA-to-FPGA (f2f) bus. This is represented by two AXI4 streaming interfaces, one per direction, for one summed beam two polarisations. The beamformer needs to exchange samples quantized at 12+12 bits, 2 polarizations, for a total of 48 bits, with 178 MHz of total bandwidth (half of 300 MHz, oversampled). This is implemented using a parallel bus of 48 bits, plus start of frame and data valid, running at 200 Mhz clock. Frames are composed of 192 valid samples per frame. The hardware interface uses 18 physical lines at 1.25 Gbps in each direction, demultiplexed into 144 lines at 156.25 MHz. A small mux/demux maps these lines into 105 lines at 200 MHz, of which 50 are used by the beamformer. The whole interface is equivalent to a set of wires, with a small and constant latency.
- The corner turner and station beamformer, using a AXI4 streaming interface. Each frame contains one time sample, for all beams/sub-bands, with a maximum total band of 150 MHz (192 channels). Frame size is variable, with a maximum of 192 valid clock cycles per frame, and is constant across one integration period. No back-pressure is used.
- The AXI4lite control bus. The control system can address in read and/or write access all the registers specifying the beamformer parameters, using 32 bit accesses. No larger or smaller (e.g. 8 bit) accesses are used. A standard AXI4lite interface developed as part of the LFAA support work is used, so parameters are assumed to be available directly as outputs from the interface, or a memory block with a simple memory mapped interface can be instantiated. The programing interface is outlined in section 3

## 2 Beamformer processing and parameters

The general structure of the tile beamformer is shown in figure 4. The two clock domains are shown separated by a dashed line, with the input section (top left) running at oversampled (237 MHz), and the beamformer running at 200 MHz.

The SysML BDD and IBD diagrams for the tile beamformer are shown in figure 5 and 6 respectively.

The tile beamformer is composed of 4 main units.

- The input frames are reformatted by the *region (sub-band) selector* that produces frames composed of the required portions of the spectrum. It also generates for each sample the sample frequency and the sample region ID, that will be used to calculate the beamforming coefficients.
- The *frequency domain beamformer* calculates the beamforming coefficients, using the input sample attributes (frequency, region ID), and delay/tapering coefficients stored in region tables. It then multiplies each sample for the calculated beamforming coefficient.
- The *beam adder* adds together the phased and tapered samples. It also provides cross connection among the two FPGAs to add together all the 16 antennas in the tile.



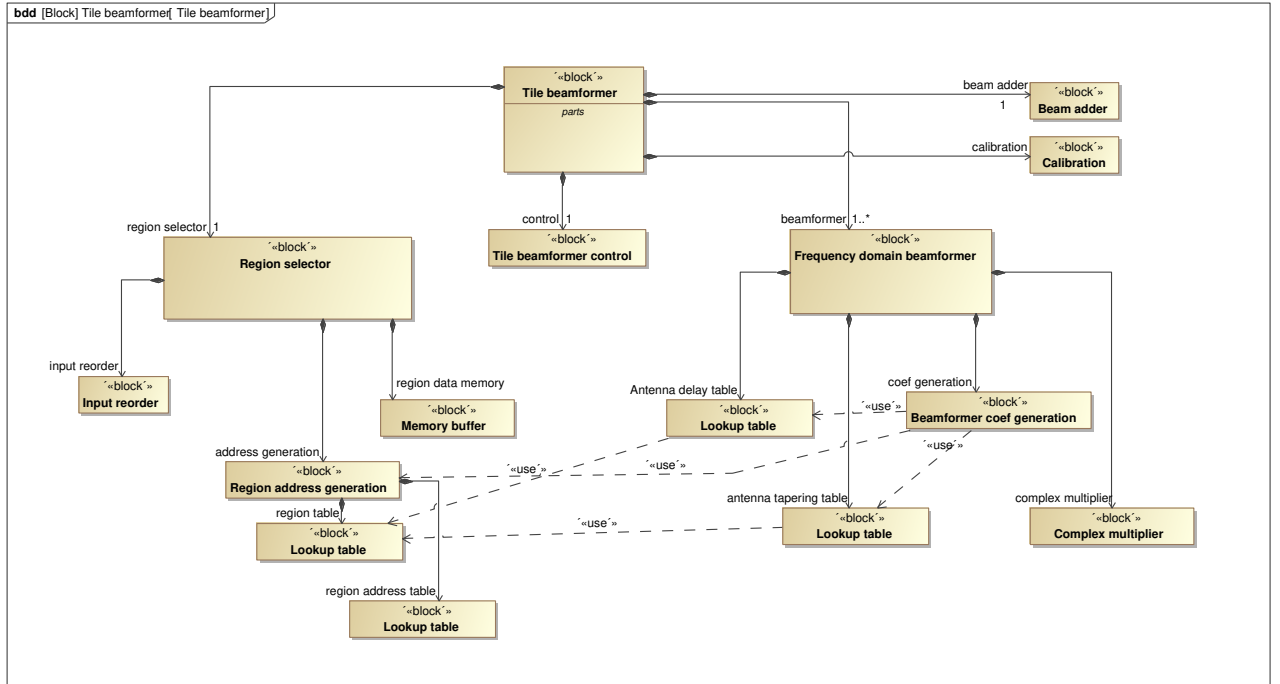


Figure 5: Tile Beamforming block definition diagram

- Everything is controlled by the *beamformer controller*.

The beamformer accepts streams of channelised data from  $N_a$  channelisers, selects portions of interest of the band, multiplies each antenna signal for an appropriate phase  $\phi(s, f) = \tau_s f$ , and sums together the results.

## 2.1 Input format

The input interface is a standard AXI4 streaming interface, using the `ecord` structure defined in the `axi4s_pkg` package. The associated clock is the same used by the hannelizer, nominally 237 MHz. The input data samples are complex, dual polarisation, 12+12 bits/sample. Frequency channels are multiplexed, with frequency index  $k$  and  $(N_c - k)$  presented simultaneously on 2 separate streams for each polarisation. Each antenna stream is thus composed of a 96 bits vector with the following bit order:

Bits	Real/imag	Frequency	Polaris.
11–0	Real	$k = 0 \dots N_c/2 - 1$	H
23–12	Imag	$k = 0 \dots N_c/2 - 1$	H
35–24	Real	$k = N_c/2 \dots N_c - 1$	H
47–36	Imag	$k = N_c/2 \dots N_c - 1$	H
55–48	Real	$k = 0 \dots N_c/2 - 1$	V
...			
95–84	Imag	$k = N_c/2 \dots N_c - 1$	V

A total of 8 antenna streams are present, for a total of 784 bits per clock cycle. A frame is composed of 256 valid clock cycles. The total number of channels is 512, over a digitised frequency range of 400 MHz (LFAA band low, digitised band 0–400 MHz, used band 50–350 MHz). The LFAA band high, with a digitised band of 375–750 MHz, and a used band 400–725 MHz) could also be used, but is not considered in this report as it is not part of the baseline design. The frequency order is direct for the lower  $N_c/2$  channels, and reversed (from  $N_c - 1$  down to  $N_c/2$ ) for the higher  $N_c/2$  channels, as shown in figure 4.

Framing and timing use the *tvalid* and *tlast* AXI4 stream signals. Frames represent consecutive time samples, with all data in a frame pertaining to the same time. Time is set when the system is reset, and proceeds by counting the incoming frames. The frame rate is fixed at one frame every 1080 ns (band low) or

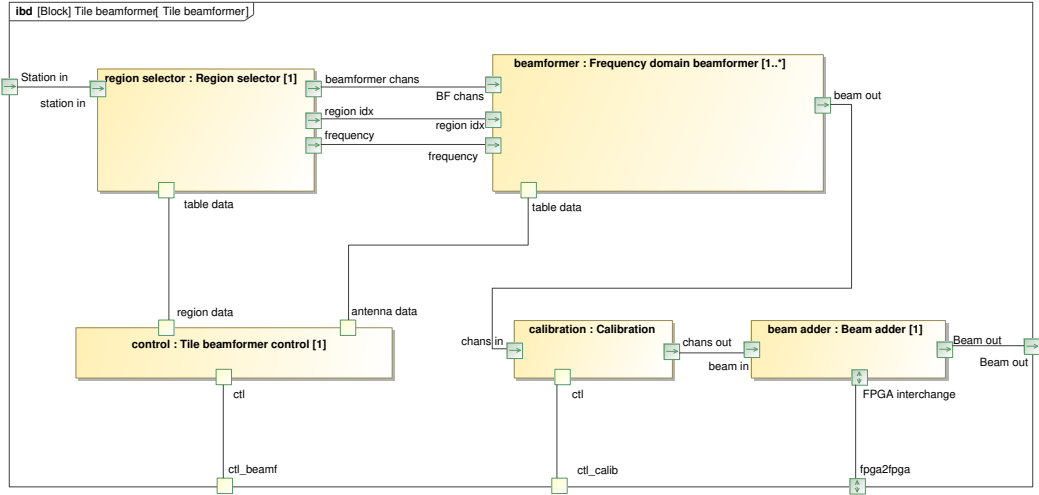


Figure 6: Tile Beamforming internal block diagram

1152 ns (band high). The frame time is given by the reset time plus the frame rate multiplied by the frame counter value, plus a fixed delay determined by the channelizer structure.

## 2.2 Calibration unit I/O format

After the region selection unit, data is still represented as frames of 12+12 bit complex samples, in a format similar to the input format, but with some differences

- Frame length is variable, and depends on the total number of selected channels, i.e. the total bandwidth of the selected regions. For normal operations, the bandwidth is 300 MHz, or  $21 \times 192$  channels.
- The two parallel streams from each antenna/polarization represent two consecutive even/odd channels.
- Channel order is in frequency ascending order, consecutively within each beam/sub-band.
- The clock is reduced to 200 MHz, to reduce the power consumption of the successive logic.

The calibrated samples are contained in frames with exactly the same structure.

## 2.3 Output format

The output format is also composed of frames representing a single channelized sample time. The format is the same for the FPGA-to-FPGA and the output streams, with the difference that odd and even frequency channels are sent separately to the two streams. Data sent through the f2f interface are further multiplexed in order to fit in the different bus width of the f2f connection. Samples are 12 + 12 bits, signed complex. Only one *antenna* (that is, the tile) is present, with both polarisations. Stream word size is thus 48 bytes with the format:

Bits	Real/imag	Polaris.
11–0	Real	H
23–12	Imag	H
35–24	Real	V
47–36	Imag	V

Sample order is in frequency ascending order, consecutively within each beam/sub-band. Only odd or even channels are present, as from table 1. Sub-bands are presented consecutively, in the order they are defined. Frame length is variable depending on the total number of frequency channels in all sub-bands, and is limited to at most 192 samples per frame. The maximum output data rate is thus 177.8 MS/s, for a total instantaneous bandwidth of 150 MHz per stream.

	<b>Upper FPGA</b>	<b>Lower FPGA</b>
F2F stream	Odd chans	Even chans
output stream	Even chans	Odd chans

Table 1: Partition of frequency channels between the two FPGAs and the output bus streams

## 2.4 Sub-band specification

Up to 16 sub-bands can be provided. Each sub-band is described to the LMC by:

- Number of frequency channels
- First frequency channel
- Tapering, delay and delay rate for each antenna.

Each sub-band may represent a frequency region, to be processed e.g. in zoom mode, and/or a different station pointing (beam).

Sub-bands are identified with a numeric code (range 0-15). Total number of frequency channels, summed over sub-bands, must not exceed 384 (300 MHz total).

For simplicity, sub-bands are expressed in multiple of 8 frequency channels. Total number of channels/beams can range from 8 to 384 in multiple of 8.

The channeliser produces as output a set of up to 384 *beamformer channels*, grouped by sub-band. Within each group, frequency channels are contiguous, in ascending order.

## 2.5 Control parameters to the TPM

The TM transmits to the LMC a list of regions to be processed. The LMC translates these to a list of parameters transmitted to the TPMs.

Parameters that need to be specified are:

- total number of output frequency channels/beams. As it is a multiple of 8, the actual number/8 is provided (the 3 LSB of the specified number are ignored)
- For each region (up to 16, but only those actually used must be specified):
  - First spectral channel to be used
  - First output (beamformer) channel used. Only the difference between these is used, so only that is specified to the TPM.
  - for each antenna
    - \* Delay
    - \* Delay rate
    - \* Tapering factor (amplitude)

These parameters are organised as:

- A register holding the total number of beamformed channels
- A *region index table*, with 64 entries, specifying the region corresponding to each group of 8 beamformer channels
- A *region address table*, with one entry for each region (total of 16 entries), specifying the difference between spectral and beamformer channels for that region; the offset is 8 bit signed, with an implicit lower bit of 0, and computed module 512 channels.
- A *beam table*, with one entry for each region (total of 16 entries), specifying the beam assigned to each region
- A *Beam definition table*, with 8 entries (one per beam), and each entry containing
- One *Antenna delay/delay rate table* with 8 entries per FPGA (one for each antenna). The table is dual bank: one bank is used while the other is updated, and they are switched at a predetermined time
- An *Update time* register, containing the frame number at which the beam definition table is updated synchronously.

### 2.5.1 Example of region definition

For example let us assume we have 32 frequency channels, divided into 3 beams:

- first beam of 16 channels starting at frequency 32 (frequency channels 32-47)
- other two beams of 8 channels each, starting at frequencies 40 and 96 respectively (frequency channels 40-47, and 96-103)

Then the region index table is:

Beamformer channel	region tab. index	Region index
0-7	0	0
8-15	1	0
16-23	2	1
24-31	3	2

The region address table is then (only last column stored in table):

Region index	Start freq channel	Start bamform channel	Difference
0	32	0	+32
1	40	16	+24
2	96	24	+72

## 2.6 Region selector

Channelised data frames are reorganised in the region selector, to provide the required spectral regions to the beamformer.

The region selector uses a double buffered memory to perform this selection. A frame is written on one bank of the memory, while the previous frame is being retrieved in the required order from the second half. Region selection is done in parallel for all antennas, so conceptually the memory is a single wide memory with a single word (768 bits) for each channel pair, all antennas and polarisations.

To reduce power consumption, the memory uses different clocks for write and read. The write clock is the same used for the channelizer, and for the input streaming interface. The output clock is 200 Mhz, sufficient to output frames of 196 words in 1080 ns.

The ordering of the channeliser frame does not allow for a completely general region selection, as at any time only one sample from each half of the spectrum can be selected. The first portion of the data stream then rearranges the input samples in a way that makes the selection more generic. The sequence of delay and cross switches shown in figure 7 produces a sequence of channels always composed of pairs of consecutive channels, with channel  $2k$  and  $2k + 1$  respectively in the first and second data stream. Successive pairs of samples refer to channels in the first and second half of the spectrum. Those channels are stored consecutively in one bank of 256 memory words. When the frame has been completely written, the bank is switched and the event is propagated to the read clock domain

The read address generation logic produces a beamformer channel number that is monotonically increasing. This quantity is used to address the region index table, and retrieves the corresponding region index. The index is then used to retrieve the channel offset that, summed with the beamformer channel number results in the required channeliser index (frequency). The frequency is also used in the coefficient generation module to calculate the proper beamforming phase.

Reworking the memory read address produce a read of channels  $2k$  and  $2k + 1$  by directly addressing location  $k$  of the dual bank memory. The algorithm for the address generation is quite simple. If  $k < N/4$  (the address is in the first half of the frame) the address is  $2k$ , otherwise it is  $N - 2k + 1$ . This can be implemented by looking at the most significant bit of  $k$ , that is always used as the least significant bit of the address. If it is 0, then the remaining bits of  $k$  are used directly, with a left shift of one position, otherwise they are complemented, again with a left shift. The following VHDL code implements this function:

```

IF k(k'left) = '0' THEN
  address <=      k(k'left-1 DOWNT0 0) & k(k'left);
ELSE
  address <= (NOT k(k'left-1 DOWNT0 0)) & k(k'left);
END IF;

```

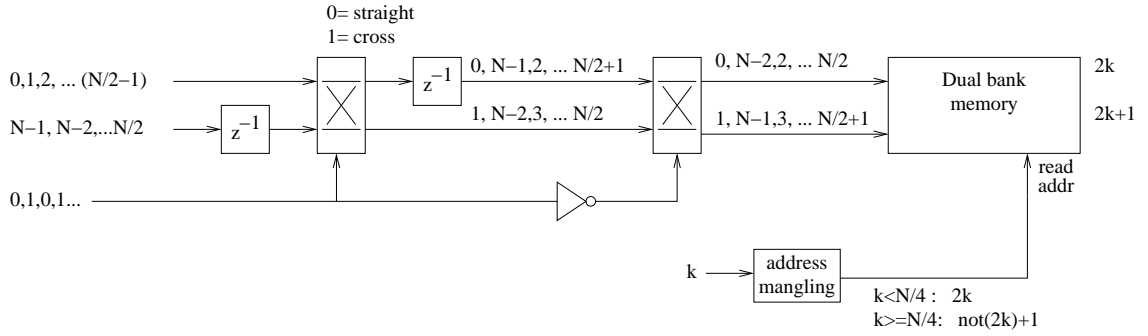


Figure 7: Input reorder logic for the region selection memory

## 2.7 Frequency domain beamformer

The frequency domain beamformer is composed of a delay generation module, a phase generation module and a complex multiplier. A separate module is required for each beam and antenna, but is common for the two polarisations. Thus a total of 64 such modules are instantiated in the beamformer. For each antenna the correct delay is selected from the beam index. Delay is converted to a phase in a phase generation module (one per antenna). For each antenna the correct delay is selected from the beam index.

The delay and delay rate are used to generate the antenna phase, as in figure 8. Initial delay is loaded at a predefined time to the delay register, and is incremented by the delay rate value every 1024 coarse samples ( $t_2 = 1.10592$  ms). The upper portion of the delay register is then multiplied by the frequency channel index, resulting in a phase. The portion of the phase corresponding to fractional turns is selected (integer turns discarded) and used to derive a phasor, addressing a look-up table.

The channel frequency is expressed in terms of the nominal channel central frequency, i.e. the channel number times the inverse of the FFT channel spacing,  $t_1 = 1/\Delta f = 1280$  ns. The phase is calculated using a dual port RAM as a look-up table for the real and imaginary parts of a complex exponential. An optimal size for the LUT, considering the dimension of the Xilinx RAM blocks, is  $2^{12}$  steps per turn, i.e. the phase is expressed as a 12 bit integer. To reduce the table size, the phase is reduced to the 1st quadrant, and the table stores a tabulated cosine only for the 1<sup>st</sup> quadrant.

The frequency is expressed as a signed value, to support negative (i.e. Nyquist reversed) frequencies, and uses 10 bits + sign to allow for extensions of the LFAA to higher frequencies. The frequency value is derived from the frequency index computed by the region selector. Half channel is added to compute the correct phase at the channel centre. The delay is specified with 20 bit precision (signed) and, considering all the rescaling of the signals (fig. 8), the value  $D$  stored in the register is:

$$D = \tau_0 2^{23}/t_1 = \tau_0/\delta\tau \quad (1)$$

with the measure unit for the delay,  $\delta\tau = 152.588$  fs. The delay range is  $\pm 80$  ns, i.e.  $\pm 64$  ADC samples. The delay is corrected before the channeliser, in the time domain, for a source at the zenith to  $\pm 1/2$  ADC sample, so the beamformer delay must compensate only the geometric delay due to station pointing. For a pointing range of up to 90 degrees from the zenith, the maximum station radius is 24 m.

The delay is multiplied by  $2^{14}$ , and the delay rate sign extended to the delay register size of 34 bits. Therefore the delay rate unit is  $2^{-14} \delta\tau/t_2 = 8.4212/10^{-15}$  s/s. The maximum delay rate expected for a LFAA station (radius of 20 m) is  $\approx \pm 5 \cdot 10^{-12}$  s/s, so 11 bits should be sufficient to specify it. 12 bits have been used, to accommodate for larger station diameters.

A linear approximation for the delay produces a phase error of  $10^{-3}$  radians after about 60 s, at the frequency of 350 MHz. The rounding error for the delay rate, using 12 bit accuracy, produces a similar error in 120 s. Therefore this accuracy is considered sufficient.

The delay and delay rate values must be specified at least once every 120 seconds, assuming that their values refer to the centre of the 120 seconds interval and a maximum station radius of 20 m.

A more frequent update, of course, increases the accuracy, reducing both the nonlinear delay variation term and the rounding errors in the delay accumulator.

The delay/delay rate is specified as a 32 bit quantity, with the delay using the 20 MS bits and the delay rate the 12 LS bits. The corrected samples are rounded to the original accuracy of 12+12 bits.



Frequency step	781.25 Hz	
Frequency width	11	bits
Frequency range	$\pm 800$	MHz
Delay width	22	bits
Delay step	152.587890625	fs
Delay range	$\pm 80.000$	ns
	$\pm 23.98$	m
	$\pm 64$	ADC samples
Sideral delay rate (20 m radius)	$\pm 4.85$	ps/s
Rate width	12	bits
Rate step	8.42124723863822	fs/s
Rate range	$\pm 17.246$	ps/s
Max. rate	$3.5\times$	sideral

Table 2: Parameters for frequency domain delay correction

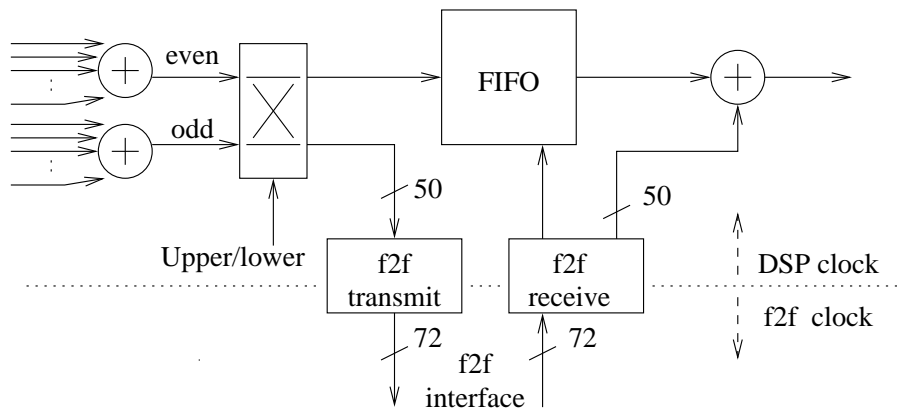


Figure 9: Tile beam adder

The tile adder uses 50 of the available 105 lines (48 sample bits plus data valid and end of frame), with an average symbol rate of 178 MS/s, while the remaining lines are used in the station beamformer.

### 3 Programming interface

The programming interface is composed of a series of tables. Each table is seen as a memory mapped region of the AXI4 address space. When tables refer to individual antennas, the antenna to be programmed is specified using a page register. A set of registers specify general parameters.

A list of the programming registers is shown in table 3. Date code register is read only. Control register is read/write, i.e. it can be read-modified-written. Other registers are write only, and do not return back the values written into.

The description given here is tentative, small modifications may occur in order to optimise the implementation, or to simplify the programming procedure.

#### 3.1 General control registers

Each peripheral in the TPM uses the first address to specify a date code. This is a read-only register, that provides a value hardcoded in the module.

The control register sets a few functions. Bit 0 resets the machine. As the timing refers to the first frame in the sequence, a reset is mandatory for correct timing. Bit 1 forces a delay update at the next frame received. Bit 2 forces a delay update at the frame specified in the `load_time` register. Both these bits must be asserted after initial programming of all the 48 `delay` registers. they must be reset before reasserting, i.e. the function is performed when these bits change from 0 to 1.

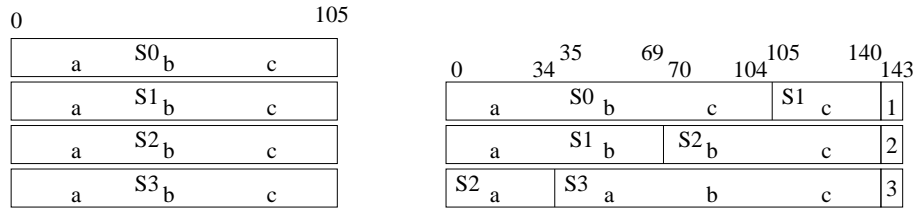


Figure 10: Multiplexing of 4 samples into 3 words over the f2f bus

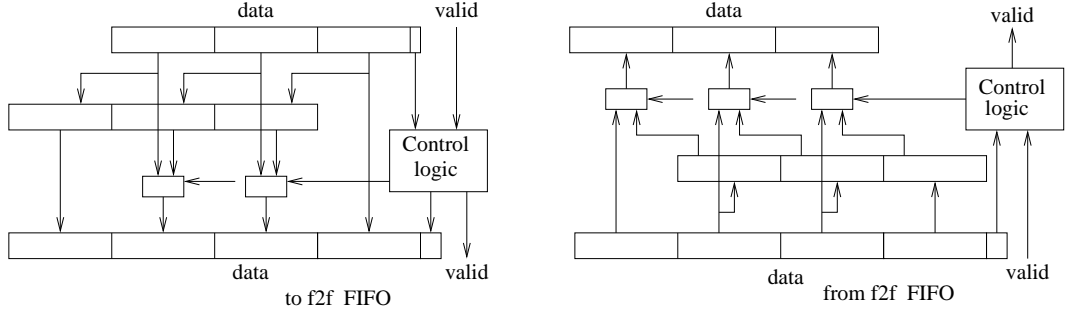


Figure 11: Conceptual schematics for the f2f bus multiplexer and demultiplexer

The number of output frequency channels, in all beams, is specified in the third register. As channels are processed in groups of 8, the 3 LSB are always ignored. They are however allowed in the register, as this limitation could be waived in future releases of the beamformer.

Delay may change during an observation. A double bank mechanism is used to allow precise synchronization of the delay value. The values for the next time period have been programmed in one bank, while the other bank is being used. Then the time (in 1024 frames units) for the exact update time is specified, in register `load_time`, and bit 2 in the control register is pulsed, arming the update state machine. When the time is detected, the two banks are exchanged, and the new delay and delay rate values are used. This has the side effect of resetting the delay, with the delay rate being applied starting at the given time.

### 3.2 Sub-band specification registers

The mechanism for specify the sub-bands is described in section 2.4. The region selection table contains a 4-bit entry for each group of 8 consecutive output channels, indicating the index for the sub-band. The index is 4 bit wide, so 8 groups (64 frequency channels) are specified in each 32 bit AXI4 word register.

The region address table specifies a 9 bit cyclic offset of the frequency channel to be used for each given output channel and sub-band. For example, if offset for sub-band 1 is 46, and output channels 8-15 belong

Address	Size	Register	
0x000	32	<code>date_code</code>	Date code
0x004	32	<code>control</code>	General control
0x008	32	<code>load_time</code>	Frame for next delay update
0x00c	9	<code>nof_chans</code>	Number of output frequency channels
0x010	32	<code>current_frame</code>	Current frame number
0x014	8	<code>tp_sel</code>	Hardware test point selection
0x80-0xbc	$16 \times 9$	<code>region_off</code>	Region offset table
0xc0-0xfc	$16 \times 3$	<code>beam_index</code>	Beam table
0x100-0x1fc	$64 \times 4$	<code>region_sel</code>	Region selection table
0x200-0x2ff	$8 \times 8 \times 32$	<code>delay</code>	Delay/delay rate table

Table 3: Programming interface for the beamformer



to sub-band 1, the frequency channels used for these output channels will be 54-61. The LS bit of the offset is ignored.

The beam index table specifies the beam associated with each sub-band. It is specified by a 3-bit index in the range 0-5. Specifying an invalid beam index is possible, but the result is unspecified.

### 3.3 Delay, delay offset and tapering tables

These tables must be specified for each antenna.

The delay and delay rate is specified as a 32 bit quantity, with the uppermost 20 bits specifying the delay, and the lowermost 12 the delay rate. As shown in section 2.7 and in table 2, the delay is expressed in units of 1280 ns (the inverse of the frequency spacing) divided by  $2^{23}$ , or 152.588 fs. This corresponds to 46  $\mu\text{m}$  of electric length, or 0.02 degrees at 350 MHz. The total delay range is  $\pm 80$  ns, or  $\pm 23.98$  m.

The delay rate is expressed in units that are  $2^{-14}$  times those used for the delay. The delay is updated every 1024 frames, i.e. every 896 times the inverse of the channel spacing. The delay rate units are thus  $2^{-37}/896 = 8.4212 \cdot 10^{-15}$  s/s, with a maximum delay rate of  $\pm 17$  ps/s, or  $\pm 5.17$  mm/s, corresponding to 3.55 times the Earth rotation rate at a radius of 20 m. Faster rates are possible, but require a partial redesign of the interface.