Detailed Design of the Arts FPGA Beamformer

|  |  |  |
| --- | --- | --- |
|  | **Organisatie / Organization** | **Datum / Date** |
| **Auteur(s) / Author(s):** |  |  |
| Eric Kooistra | ASTRON | 2015 |
| **Controle / Checked:** |  |  |
| Andre Gunst | ASTRON |  |
| **Goedkeuring / Approval:** |  |  |
| Andre Gunst | ASTRON |  |
| **Autorisatie / Authorisation:** |  |  |
| **Handtekening / Signature**  Andre Gunst | ASTRON |  |
|  |  |  |
| © ASTRON 2015 All rights are reserved. Reproduction in whole or in part is prohibited without written consent of the copyright owner. | | |

**Distribution list:**

|  |  |
| --- | --- |
| **Group:** | **Others:** |
| Joeri van Leeuwen (JvL)  Andre Gunst (AG)  Daniel van der Schuur (DS)  Hajee Pepping (HJP)  Alwin Zanting (AZ)  Leon Hiemstra (LH) | Gert Kruithof (GK)  Wim van Capellen (WvC) |

**Document history:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Revision** | **Date** | **Author** | **Modification / Change** |
| 0.1 | 2015-05-13 | E. Kooistra | First draft that describes the CB input. |
| 0.2 | 2015-06-10 | E. Kooistra | Added sections on:   * Bsub=1 MHz change request * TAB, IAB data processing relations * CB12, TAB-1/12/444, IAB-37 output * DSP resource usage (multipliers, RAM) |
| 0.3 | 2015-07-06 | E. Kooistra | Defined Muni = Nband = 16. |
|  |  |  |  |
|  |  |  |  |

**Table of contents:**

[1 Introduction 8](#_Toc423617590)

[1.1 Scope 8](#_Toc423617591)

[1.2 Specification 9](#_Toc423617592)

[1.3 Change request Bsub = 1 MHz 9](#_Toc423617593)

[2 System overview 10](#_Toc423617594)

[2.1 Apertif BF subsystem 10](#_Toc423617595)

[2.2 Apertif X subsystem and Arts subsystem 10](#_Toc423617596)

[3 Hardware architecture 12](#_Toc423617597)

[3.1 Apertif BF using Uniboard 12](#_Toc423617598)

[3.2 Signal path transpose to Arts 12](#_Toc423617599)

[3.3 Arts using UniBoard 13](#_Toc423617600)

[4 Data path processing 15](#_Toc423617601)

[4.1 Channel filterbank 15](#_Toc423617602)

[4.2 Stokes power vector (IQUV) 15](#_Toc423617603)

[4.3 Tied array beamformer (TAB) 15](#_Toc423617604)

[4.3.1 ‘Voltage’ beams 15](#_Toc423617605)

[4.3.2 ‘Power’ beams 15](#_Toc423617606)

[4.4 Incoherent array beamformer (IAB) – ‘power’ beams 15](#_Toc423617607)

[4.5 Requantization 16](#_Toc423617608)

[5 Streaming data interfaces 17](#_Toc423617609)

[5.1 System load overview 17](#_Toc423617610)

[5.1.1 CB input load from Apertif BF 17](#_Toc423617611)

[5.1.2 Arts BF output load to Arts PL 17](#_Toc423617612)

[5.2 CB signal path input 18](#_Toc423617613)

[5.2.1 Entire system 18](#_Toc423617614)

[5.2.2 Per UniBoard 19](#_Toc423617615)

[5.2.3 Distribution on the UniBoard mesh 19](#_Toc423617616)

[5.2.4 Per PN 20](#_Toc423617617)

[5.3 Beamlet mapping in the Apertif BF 21](#_Toc423617618)

[5.4 Processing input and output 21](#_Toc423617619)

[5.4.1 Channel filterbank 21](#_Toc423617620)

[5.4.2 Tied array beamformer (TAB) 21](#_Toc423617621)

[5.4.3 Incoherent array beamformer (IAB) 22](#_Toc423617622)

[5.5 SC1 TAB-1 output 23](#_Toc423617623)

[5.6 SC2 TAB-12 output 23](#_Toc423617624)

[5.7 SC2 CB-12 output 24](#_Toc423617625)

[5.8 SC3 IAB-37 output 25](#_Toc423617626)

[5.9 SC4 TAB-444 output 26](#_Toc423617627)

[6 Transient data capture 27](#_Toc423617628)

[6.1 External memory on UniBoard 27](#_Toc423617629)

[6.2 Capture integrated power data 27](#_Toc423617630)

[6.3 Capture CB voltage data 27](#_Toc423617631)

[7 Memory-mapped control interface 28](#_Toc423617632)

[8 Processing 28](#_Toc423617633)

[9 Storage 28](#_Toc423617634)

[10 Verification 28](#_Toc423617635)

[11 Appendix : VHDL component resource usage and IO formats 29](#_Toc423617636)

[11.1 Available FPGA resources on UniBoard 29](#_Toc423617637)

[11.1.1 Multipliers 29](#_Toc423617638)

[11.1.2 Memory 29](#_Toc423617639)

[11.2 PFB component 29](#_Toc423617640)

[11.2.1 Multiplier usage 29](#_Toc423617641)

[11.2.2 Memory usage 30](#_Toc423617642)

[11.2.3 Input - output 31](#_Toc423617643)

[11.3 Reorder component 31](#_Toc423617644)

[11.3.1 Memory usage 32](#_Toc423617645)

[11.3.2 Input - output 32](#_Toc423617646)

[11.4 BF unit component 32](#_Toc423617647)

[11.4.1 Multiplier usage 32](#_Toc423617648)

[11.4.2 Memory usage 33](#_Toc423617649)

[11.4.3 Input - output 33](#_Toc423617650)

[11.5 IBF unit component 33](#_Toc423617651)

**References:**

1. “Arts Requirements Specification”, ASTRON-RS-020, J. van Leeuwen
2. “Apertif System Requirement Specification”, ASTRON-RS-006, Rev C, W.A. van Capellen
3. “Analysis of tied-array beamforming for Arts”, ASTRON-MEM-191, S.J. Wijnholds
4. “Impact analysis of change request to use 1 MHz beamlets instead of 0.78125 MHz beamlets in Apertif and Arts”, ASTRON-CR-032, E. Kooistra
5. “Detailed Design of the Digital Beamformer System for Apertif”, ASTRON-RP-413, G. Schoonderbeek, A. Gunst, E. Kooistra
6. “Apertif Beamformer Firmware Description”, ASTRON-RP-1377, D. vd Schuur
7. “Detailed Design of the Correlator System for Apertif”, ASTRON-RP-1400, G. Schoonderbeek, A. Gunst, E. Kooistra, HJ Pepping
8. “Representing Streaming Data”, ASTRON-RP-1486, E. Kooistra
9. “Apertif Beamformer Output Interface Specification”, ASTRON-SP-061, E. Kooistra
10. “Understanding Digital Signal Processing”, 3rd edition, R. Lyons
11. “Apertif Filterbank Firmware Specification Part 2”, ASTRON-SP-054, E. Kooistra
12. "Selection and Reorder modules", ASTRON-RP-1399, H.J. Pepping
13. “Uthernet interface specification”, ASTRON-SP-041, E. Kooistra
14. “Data path packet interface specification”, ASTRON-SP-042, E. Kooistra

**Terminology:**

ADC Analogue to Digital Conversion

ADU Analogue to Digital Unit (board with 8 ADC)  
AGC Automatic Gain Control   
Apertif APERture Tile In Focus

Arts Apertif Radio Transient System

beam Group of beamlets that point in the same direction

beamlet Beam formed subband, a small beam spanning one subband

BF BeamFormer

BN Back Node FPGA on UniBoard

bps Bits per second

BSN Block Sequence Number (timestamp)  
BW BandWidth

CB Compound Beam, formed at dish level over the FPA  
channel Unit frequency band within a beamlet  
cint Complex integer

CPU Central Processing Unit

CR Change Request  
CW Carrier Wave (single frequency signal)

DP Data Path (streaming interface)

DSP Digital Signal Processing

DT Delay Tracking in Aperif BF  
eop End of Packet (or frame, or block)  
FFT Fast Fourier Transform

FN Front Node FPGA on UniBoard

FoV Field of View  
FPA Focal Plane Array (= PAF)  
FPGA Field Programmable Gate Array

FR Functional Requirement  
FRB Fast Radio Burst

Fsub Subband filterbank in Apertif BF  
Fchan\_x Channel filterbank in Apertif X  
Fchan Channel filterbank in Arts BF for SC3 and SC4

FS Fringe Stopping (is DT + PT)  
GbE Gigabit Ethernet

GPU Graphics Processing Unit  
HDL Hardware Description Language

IAB Incoherent array beam, formed by incoherently combining dishes

IBF Incoherent BeamFormer

Iab-let An incoherently beamformed beamlet, spanning one channel  
Im Imaginary

int Signed integer

IO Input Output

MAC Multiply and Accumulate, Medium Access, Monitoring and Control

MM Memory Mapped (control interface)

node Processing node (PN), typically one FPGA chip  
Nof Number of

OEB Optical-Electrical Board (provides UniBoard BN with same optical IO as the FN)

PFB Poly phase Filter Bank  
power beam Full Stokes power values: I, Q, U, V

PAF Phased Array Feed (= FPA, better use term FPA)  
PL Pipeline processing  
PN Processing Node (FN or BN), PN0:3 = FN0:3, PN4:7=BN0:3

PPS Pulse Per Second  
PT Phase Tracking in Aperif BF  
Re Real

RF Radio Frequency

Rsub Reorder and select subbands for CBBW=300 MHz in aperitif BF

Rbeam Reorder and select beamlets for NCB=37 compound beam directions aperitif BF

SC Science Case  
SNR Signal to Noise Ratio

sop Start of Packet (or frame, or block)  
SP Signal Path, 1 CB consists of Npol = 2 SP, 1 SP per Apertif BF subrack

SR Science Requirement

ST Streaming, statistics  
sps Samples per second

subband Frequency band, unit output of the filterbank

TAB Tied array beam, formed by coherently combining dishes  
tab-let A coherently beamformed beamlet, a grating or pencil beam within the CB   
Tant Transpose to group data from all S = 64 (≥ Nant) antenna elements in the FPA  
Tdish Transpose to group data from all Ndish = 12 dishes  
Tpol Transpose to group data from both Npol = 2 polarizations  
Tsp Transpose to group data from all Nsp = Npol \* Ndish signal paths, so combines Tdish and Tpol   
Tband Transpose to group data from all Nband = 16 bands

Tintegration Transpose to group data from an integration interval of Nint\_x values in time  
TFoV Transpose to group data from all NCB = 37 beams for the full FoV

ToA Time of Arrival  
TT Terrestrial Time  
uint Unsigned integer

VLBI Very Large Baseline Interferometry  
voltage beam Dual polarization sample values with phase information: Xre, Xim, Yre, Yim

WSRT Westerbork Synthesis Radio Telescope  
X Correlator

**Definitions:**

Ncomplex 2 Two part of a complex number, the real and imaginary part   
Npol 2 Number of polarizations, X and Y  
NStokes 4 Number of power values in the Stokes vector [I, Q, U, V]  
Ndish 12 Number of WSRT dishes in Apertif  
Nsp 24 Number of signal paths = Ndish \* Npol at the output of the Apertif BF  
Nant 61 Number of antennas in the frontend FPA of the Apertif BF

S 64 Number of ADC signal paths in the frontend FPA of the Apertif BF (≥ Nant)

SBN 4 Number of ADC signal paths per BN in the frontend FPA of the Apertif BF

fs 800 MHz Digitizer sample frequency of the ADC at the Apertif BF frontend  
Ts 1.25 ns = 1/ fs, digitizer sample period  
f0 Lower edge frequency of a subband, beamlet or channel [1]  
RFBW 400 MHz = fs/2, sampled RF bandwidth   
CBBW 300 MHz Full bandwidth of the CB and also of the TAB and IAB (SR-0.2)  
Bsub 781250 Hz Subband bandwidth in Apertif BF, = beamlet bandwidth  
fclk 200M Data processing clock rate in the FPGA, = fs / P

P 4 Wideband rate factor of sample clock rate divided by digital processing clock rate

Pcmult 4 Number of real multiplications per complex multiplication

N 1024 FFT size of the FFT in the Apertif BF subband polyphase filter

Nclk 256 = N/P, number of DP clock cycles per subband period  
Nsub 512 = N/2, number of subbands that covers RFBW=400MHz  
Nsel 384 Number of selected subbands to cover CBBW=300 MHz  
Nband 16 = nof\_fn\_bf, Number of bands in the Apertif BF to process the full CBBWNFN 24 = Nsel/Nband , number of subbands per band or per FN in the Apertif BF

NFNpol 48 = NFN\*Npol, number of subbands per dual polarization band

NCB 37 Required number of compound beams  
KCB 40 Implemented average number of beamlets per subband (≥ NCB)

PBF 4 Number of parallel BF units per FN in the Apertif BF

Nblk 240 ≤ Nclk, number of valid DP clock cycles per subband period in the Apertif BF

Nbeamlet 960 Number of compound beamlet slots per FN output, maximum PBF \* Nclk = 1024,

actual PBF \* Nblk = 960, required NCB \* NFN = 888

Ninterleave  2 = nof\_un/PBF, additional beamlet output interleave factor

Pinterleave  2 Number of interleaved streams that are unfolded in parallel at rate 1/Pinterleave

Mblk =Nblk/Ninterleave, default block size per subband period in the Apertif X and Arts

Ngr 12 Required number of TAB grating lobe patterns to cover the full CB (SR-0.41)  
NVLBI 12 Required number of TABs in the central CB for VLBI, choose = Ngr (SR-0.23)  
KTAB 12 Implemented number of TABs per beamlet (≥ Ngr)

NTAB 444 = NCB\*KTAB, number of TABs   
NIAB 37 = NCB, number of IABs   
Nlink 384 = NPN, number of physical 10G output links of the Apertif BF, so 1 link per PN  
NPN 384 = Nsp \* Nband, total number of parallel processing nodes in the Apertif BF  
MPN 128 = Nband \* nof\_un, total number of parallel processing nodes in the Arts

Muni 16 = Nband = nof\_fn\_bf, total number UniBoards in the Arts BF and in Apertif X

MPFB\_units 12 Number of parallel PFB units per PN for Nsp = 24 SP

MBF\_units 12 Number of parallel BF units per PN for KTAB = 12 TABs

MIBF\_units 1 Number of parallel dual polarization IBF units per PN for the one IAB per CB

Nchan\_x 64 Number of channels per beamlet in the Apertif X

Nchan 4 Number of channels per beamlet, for SC3 and SC4  
Bchan = Bsub/Nchan, channel bandwidth within a beamlet, for SC3 and SC4

Nint\_x 800000 Number of channel power values that are integrated in the Apertif X  
Nint ≈ 10 Number of Stokes channel power values that are integrated in Arts  
TStokes ≈ 50 μs Minimum required sample period for the Stokes power values  
fStokes ≈ 20 kHz = 1/TStokes, minimum required sample frequency for the Stokes power values  
nof\_uni 4 Number of UniBoards per polarization and dish in the Apertif BF

nof\_bn 4 Number of back node FPGAs (BN) per UniBoard

nof\_fn 4 Number of front node FPGAs (FN) per UniBoard

nof\_un 8 = nof\_fn + nof\_bn, number of processing node FPGAs per UniBoard

nof\_10g 3 Number of 10G links per FPGA node on UniBoard

nof\_pn Number of processing nodes (BN or FN)

nof\_bn\_fb 16 = nof\_uni\*nof\_bn, number of subband filterbank BN per SP in the Apertif BF

nof\_fn\_bf 16 = nof\_uni\*nof\_fn, number of beamformer FN per SP in the Apertif BF

byte\_w 8 Number of bits in a byte or an octet

word\_sz 4 Number of bytes per 32 bit long word  
longword\_sz 8 Number of bytes per 64 bit long word  
Wbeamlet 6 Word width in number of bits of a beamlet voltage sample  
Wchan 6 Word width in number of bits of a channel voltage sample  
Wtab 4 Word width in number of bits of a TAB voltage sample  
Wpower 4 Word width in number of bits of a IAB or TAB power sample

# Introduction

## Scope

Arts [1] implements the tied array and VLBI functionality of Apertif [2]. Figure 1 shows the place of Arts within Apertif. Both the Apertif correlator (X) [6] and Arts use the beam data from the Apertif beamformer (BF) [4].



Figure 1: Top level overview of Apertif with Arts included

Within Arts the processing is consists of a FPGA beamformer and a GPU pipeline, as shown in Figure 2. Arts has four science cases (SC) and for all four SC the FPGA beamformer will be implemented on Uniboards.



Figure 2: Arts FPGA beamformer and GPU pipeline

This document specifies the detailed design for the Arts FPGA beamformer (BF) on UniBoard FPGAs. At the input interface the Arts BF receives NCB=37 compound beams from Ndish=12 dishes from the Apertif BF. At the output interface the Arts BF outputs compound beams (CB) for SC2, tied array beams (TAB) for SC1, 2 and 4 or incoherent array beams (IAB) for SC3 to a GPU cluster for further processing.

Note the difference between the Apertif BF and the Arts BF. The Apertif BF forms the compound beams over the focal plane array of each dish. These compound beams are input to Aperitif X and to Arts. The Arts BF uses the compound beams to form IAB or TAB over the array of dishes.

## Specification

This detailed design document is the L3 specification of the FPGA beamformer because it specifies how the FPGA beamformer should be implemented on Uniboards to fulfill all L0 science, L1 system and L2 subsystem requirements that are specified in [1]. This document only specifies the FPGA firmware design and the required FPGA interconnect and IO architecture. The UniBoard hardware and the subrack hardware are assumed to be available. The output to the pipeline processing (PL) is specified. The pipeline processing is not described in this document, because the PL will be implemented on a GPU cluster. Similar the monitoring and control (MAC) interface is specified, but the MAC design itself is not described.

The array notation that is used in this document to describe the data format for the streaming data interfaces is explained in [7].

## Change request Bsub = 1 MHz

There is a pending change request from Arts SC1 and SC2 for the Apertif BF to change Bsub from 781250 Hz to 1 MHz [3]. This will ripple through in all related parameters. In this document the parameter values for Bsub = 781250 Hz are still used as default. Table 1 shows how the parameter values will change for Bsub = 1 MHz.

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Bsub = 781250 Hz** | **Bsub = 1 MHz** | **Remark** |
| N | 1024 | 800 | = fs / Bsub, keep fs = 800 MHz |
| Nsub | 512 | 400 | = N / Ncomplex |
| Nsel | 384 | 304 | Nsel \* Bsub ≥ CBBW = 300 MHz |
| NFN | 24 | 19 | = Nsel/Nband |
| Nbeamlet | 960 | 760 | = KCB \* NFN, keep KCB = 40 |
| Nclk | 256 | 200 | = N / P, keep P = 4 and fclk = fs / P |
| Nblk | 240 | 190 | = Nbeamlet / PBF |
| Bchan | 195.3125 kHz | 250 kHz | = Bsub / Nchan |
| Bchan\_x | 12.207 kHz | 15.625 kHz | = Bsub / Nchan\_x |
|  |  |  |  |

Table 1: Parameter values that depend on Bsub

# System overview

## Apertif BF subsystem

The Apertif BF separates the digitized data from the dish FPA into subbands by means of a filterbank and then it forms beamlets for these subbands. The beamforming (BF) for one single polarization of the FPA cannot be done on a single FPGA node for the full bandwidth, so therefore the subband load has to be distributed across Nband = nof\_fn\_bf = 16 processing nodes. The beamlet for one subband requires the input from all FPA elements, so therefore there needs to be a transpose Tant that groups the subbands from all S = 64 antennes. A compound beam (CB) is formed by a group of Nsel= 384 beamlets all with the same direction that span CBBW = 300 MHz. Figure 3 shows the filterbank Fsub, the transpose Tant and the beamformer (BF) that is distributed over Nband nodes. The Tintegration transpose is used for the Apertif correlator (X), for Arts it needs to be bypassed. The MAC takes care of the proper operation, the subband selection and the BF weights.



Figure 3: The Apertif BF subsystem

## Apertif X subsystem and Arts subsystem

In the appendix of [1] various options for the Arts subsystem were investigated. Figure 4 shows the selected option for the Arts subsystem and how it relates to the Apertif X subsystem. The same Tdish and Tpol transpose that are needed for Apertif X to group the data from all Nsp = Npol \* Ndish = 24 signal paths can also be used for Arts.



Figure 4: The Apertif X subsystem and the Arts subsystem

SC1 and SC2 use the central CB so they only need to process CB-12, where Ndish = 12. For SC1 only one TAB-1 needs to be made within the CB. For SC2 TAB-12 needs to be made, where the number of TABs for VLBI is NVLBI = 12. For SC2 it must also be possible to pass on the CB-12.

Apertif X, SC3 and SC4 use all CB, so they process CB-444, where Ndish \* NCB = 12 \* 37 = 444. For SC3 the IAB-37 needs to be made which is one incoherent ‘power’ beam for each CB. For SC4 the TAB-444 needs to be made which is KTAB = 12 ‘voltage beams per CB, so KTAB \* NCB = 12 \* 37 = 444. Both SC3 and SC4 output integrated (over Nint samples) Stokes (IQUV) ‘power’ beam data to the pipelining (PL).

# Hardware architecture

## Apertif BF using Uniboard

The Apertif BF outputs NCB=37 compound beams with CBBW=300 MHz. The Apertif BF beam forms the FPA input per polarization and per dish. The single dish, single polarization output of the Apertif BF is called a signal path (SP) and to beam form 1 signal path requires a subrack with nof\_uni=4 UniBoards. To be able to distribute the processing over nof\_fn\_bf=16 front nodes (FN) on nof\_uni=4 UniBoards the Apertif BF is separated into Nband= nof\_fn\_bf =16 frequency bands. Figure 5 shows the Apertif BF subrack with 4 UniBoards. The 4 UniBoards connect to the ADU ADC cards and to each other via a backplane. Each FN in the subrack uses one 10GbE port to output its frequency part of the signal path.



Figure 5: One Apertif BF subrack per signal path with nof\_uni=4 UniBoards and Nband=16 FN

## Signal path transpose to Arts

In total the Apertif BF has Nsp = Npol \* Ndish = 24 signal paths so also 24 subracks, 96 UniBoards and NPN=Nsp\*Nband = 384 processing (front) nodes. Hence the total Apertif BF output is carried via Nlink=NPN=384 10GbE links as shown in Figure 6. For both the Apertif X and for Arts the Nsp=24 signal paths from the Apertif BF need to be transposed to gather them together. This transpose Tsp can be implemented by interconnecting the Apertif BF to Nband=Muni = 16 Uniboards as shown in Figure 6. Each of the Nband= Muni =16 UniBoards in Figure 6 processes 1/Nband part of the CBBW band for all Nsp=24 signal paths.



Figure 6: Apertif BF transpose interconnect to Apertif X and Arts

## Arts using UniBoard

Figure 7 shows the UniBoard with the Optical-Electrical Board (OEB). The OEB is needed to be able to use fiber optics IO for the BN. For the Arts application (and also for the Apertif X application) the distinction between FN and BN is not needed, because all nof\_un = nof\_fn + nof\_bn = 8 FPGA have the same function. Therefore the FPGAs on UniBoard are also referred to as processing nodes (PN). Each PN has nof\_10G = 3 10G links so in total the UniBoard has nof\_un \* nof\_10G = 24 10G links. This is just enough IO to accept the input from Nsp=24 links. Hence the number of input links determines that all nof\_un = 8 FPGAs on UniBoard have to be used for Arts.



Figure 7: One UniBoard to process 1/Nband part of the CBBW for Nsp=24 signal paths

The mesh interconnect on UniBoard consists of gigabit transceivers that connect each BN to each FN and each FN to each BN. On UniBoard the Nsp=24 inputs need to be distributed further via the on board mesh interconnect to gather them together at each PN. Therefore the input needs to be divided into 1/nof\_un parts to evenly distribute the processing load over the PN.

In total the Arts FPGA beamformer requires Nband = Muni = 16 UniBoards as shown in Figure 8. The Muni=16 UniBoards are independent, because each processes 1/Nband part of the CBBW. Similar the Apertif X also uses Nband = Muni = 16 independent UniBoards.



Figure 8: Nband = Muni = 16 independent UniBoards for the Arts BF

# Data path processing

## Channel filterbank

The channel filter is indicated by Fchan in Figure 4. For SC3 and SC4 the beamlet bandwidth of Bsub is too wide so therefore an additional channel filter is needed that separates the beamlets into Nchan = 4 channels. Hence Bchan = Bsub/Nchan = 195.3125 kHz or 250 kHz, dependent on whether Bsub = 781250 Hz or 1 MHz respectively.

The channel filterbank in Arts and in the Apertif X reuse the same WPFB component that is also used for the subband filterbank in the Apertif BF. The channel filter is implemented by a poly-phase filterbank (PFB). The PFB has a FIR pre-filter section and an FFT section. The number of taps in the FIR filters determines the steepness of the pass-stop band transition. The stop band attenuation is independent of the number of taps. The subband PFB in the Apertif BF uses Ntaps=16. However for the channel filters in Arts using Ntaps=16 may take too many multiplier resources, therefore set Ntaps=8.

## Stokes power vector (IQUV)

The Stokes power vector [I, Q, U, V] consists of powers that are based on both polarizations X and Y:

Equation 1:

The advantage of the (Stokes) power values is that they can be integrated to achieve data reduction.

## Tied array beamformer (TAB)

### ‘Voltage’ beams

The tied array beamformer is indicated by TAB in Figure 4 and operates on ‘voltage’ samples. For SC1 TAB-1 and SC2 TAB-12 the beamformer operates directly on the beamlets samples. For SC4 the TAB-444 operates on the channel samples. The TAB in Arts reuses the same BF component that is also used for the subband beamformer in the Apertif BF that makes beamlets of the compound beams (CB).

### ‘Power’ beams

For SC4 the Stokes powers of the TAB-444 voltage beam data are calculated conform Equation 1 and then integrated over Nint ≈ 10 samples as shown by TAB-444 🡪 IQUV🡪 ∑ in Figure 4. The integration achieves a data rate reduction of 1/Nint \* (Wpower/WTAB).

## Incoherent array beamformer (IAB) – ‘power’ beams

The incoherent array beamformer is indicated by the IAB in Figure 4 and operates on ‘power’ samples. The incoherent beamforming makes that the IAB has the same FoV as the CB and that the IAB does not use weights. Therefore there is one IAB per CB. The input ‘power’ data is calculated conform Equation 1. The integration over Nint ≈ 10 samples can be done before the IAB as shown in Figure 4 by IQUV🡪 ∑🡪IAB-37 or after the IAB. The incoherent beamformer (IBF) component is new but merely consists of a adder tree to sum the ‘power’ data from the Ndish = 12 inputs.

## Requantization

Wbeamlet = 6 bit, Wchan = 6 bit, WTAB = 6 bit, Wpower = 8 semi floating point with 1 bit exponent.

# Streaming data interfaces

## System load overview

### CB input load from Apertif BF

Table 2 copied from [1] list loads that can be defined regarding the Apertif BF output interface assuming that the Apertif BF outputs beamlets with Wbeamlet = 6 bits.

|  |  |  |  |
| --- | --- | --- | --- |
| **Load** | **Equation** | **Value** | **Description** |
| LBF\_SP1 | = CBBW \* Ncomplex \* Wbeamlet | 3.6 Gbps | Load for 1 SP |
| LBF\_SP1\_band | = LBF\_SP1 / Nband | 225 Mbps | Load for 1 SP per band (= per BF node) |
| LBF\_SP37\_band | = NCB \* LBF\_SP1\_band | 8.325 Gbps | Load for NCB = 37 SP per band (= per BF node) |
|  |  |  |  |
| LBF\_CB1 | = Npol \* LBF\_SP1 | 7.2 Gbps | Load for 1 CB (= 2 SP, Npol = 2) |
| LBF\_CB12 | = Ndish \* LBF\_CB1  = NPN \* LBF\_SP1\_band | 86.4 Gbps | Total load from Ndish = 12 dishes, for 1 CB |
|  |  |  |  |
| LBF\_CB444 | = NCB \* LBF\_CB12  = NPN \* LBF\_SP37\_band | 3.2 Tbps | Total load from Ndish = 12 dishes, for NCB= 37 CB |
|  |  |  |  |
| LBF\_link1 | = LBF\_SP1\_band | 225 Mbps | Link load for 1 SP per band (= per BF node) |
| LBF\_link37 | = LBF\_SP37\_band | 8.325 Gbps | Link load for NCB= 37 SP per band (= per BF node) |
|  |  |  |  |

Table 2: Load definitions for Apertif BF output interface with Wbeamlet = 6 bit

### Arts BF output load to Arts PL

Table 3 copied from [1] list output loads for SC1, SC2, SC3 and SC4 from the Arts BF to the Arts Pipeline (PL).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Load** | **SC** | **Equation** | **Value** | **Description** |
| LTAB1 | 1 | = CBBW \* Npol \*Ncomplex \* Wtab | 4.8 Gbps | Load for voltage TAB-1 |
| LTAB1\_band | 1 | = LTAB1 / Nband | 300 Mbps | Load for voltage TAB-1 per PN0 |
|  |  |  |  |  |
| LTAB12 | 2 | = NVLBI \* LTAB1 | 57.6 Gbps | Load for voltage TAB-12 |
| LTAB12\_band | 2 | = LTAB12 / Nband | 3.6 Gbps | Load for voltage TAB-12 per PN0 |
| LCB12 | 2 | = LBF\_CB12 | 86.4 Gbps | Load for voltage CB-12 |
| LCB12\_band | 2 | = LCB12 / Nband | 5.4 Gbps | Load for voltage CB-12 per PN0 |
|  |  |  |  |  |
| LIAB1\_stokes | 3 | = CBBW \* NStokes \* Wpower | 4.8 Gbps | Load for IAB-1 without integration |
| LIAB1\_stokes\_int | 3 | = LIAB1\_stokes / Nint | 0.48 Gbps | Load for IAB-1 with Tstokes ≈ 50 μs |
| LIAB37\_stokes | 3 | = NCB \* LIAB1\_stokes | 177.6 Gbps | Load for IAB-37 without integration |
| LIAB37\_stokes\_int | 3 | = NCB \* LIAB1\_stokes\_int | 17.76 Gbps | Load for IAB-37 with Tstokes ≈ 50 μs |
| LIAB37\_stokes\_I\_int | 3 | = LIAB37\_stokes\_int / NStokes | 4.44 Gbps | Load for IAB-37-I with Tstokes ≈ 50 μs |
|  |  |  |  |  |
| LTAB1\_stokes\_int | 4 | = LIAB1\_stokes\_int | 0.48 Gbps | Load for TAB-1 with Tstokes ≈ 50 μs |
| LTAB444 | 4 | = NCB \* Ngr \* LTAB1 | 2.1312 Tbps | Load for voltage TAB-444 |
| LTAB444\_stokes | 4 | = LTAB444\* (Wtab/Wpower) | 2.1312 Tbps | Load for power TAB-444 |
| LTAB444\_stokes\_int | 4 | = NCB \* Ngr \* LTAB1\_stokes\_int | 213.12 Gbps | Load for TAB-444 with Tstokes ≈ 50 μs |
| LTAB444\_stokes\_I\_int | 4 | = LTAB444\_stokes\_int / NStokes | 53.28 Gbps | Load for TAB-444-I with Tstokes ≈ 50 μs |
|  |  |  |  |  |

Table 3: Load definitions for Arts BF output interface (with Wbeamlet = 6 bit, Wtab = 4 bit, Wpower = 4 bit)

## CB signal path input

### Entire system

The compound beam data output interface of the Apertif BF is defined by [8]:

Equation 2:

The wiring of the Nlink=384 10GbE links in Figure 6 implements the first part of the Tdish and Tpol transpose to group the SP data from all Nsp=24 signal paths per band at a single UniBoard. The Tdish and Tpol transposed beam data input for Arts BF (and also for Apertif X) is defined by (note the swapped band and dish indices):

Equation 3:

The subscript indices indicate parallel links and the array index contains serial data on the link. The subscript *band* has range 0:Nband-1, subscript *pol* has range 0:Npol-1, subscript *dish* has range 0:Ndish-1. In total there are Nband \* Npol \* Ndish = 16 \* 2 \* 12 = Nlink = 384 parallel links. The array index *t* increments at the rate of Bsub. The array index *b* has range 0:Nbeamlet-1 where Nbeamlet  is the number of compound beamlet slots per FN output of the Apertif BF. Required Nbeamlet ≥ NCB\*NFN = 37 \* 24 = 888, where NFN = Nsel / Nband. The actual number of beamlet slots that will be implemented is Nbeamlet = KCB \* NFN = 40 \* 24 = 960. The order of beamlet directions and beamlet frequencies can be mapped to the beamlet slots in almost any order by the reorder function in the Apertif BF [8].

The order of the subscript indices indicates that band 0 maps on UniBoard 0 and band 15 maps on UniBoard 15. The pol index before the dish index implies that the X-pol inputs are connected via stream [0:11] to FN0:3 and the Y-pol inputs are connected via stream [12:23] to BN0:3 as shown in Figure 7. Based on this the *pol* and *dish* indices can be mapped to *port* and *pn* indices and to index *sp* = 0:Nsp-1 = 0:23 according to:

Equation 4:

Equation 5:

Equation 6:

With Equation 4 the Equation 3 can be rewritten as:

Equation 7:

### Per UniBoard

One UniBoard processes one band of all Nsp=24 SP. Call this signal *cb\_uniboard*, so *cb\_uniboard* = *CBband* where a *band* is mapped to this UniBoard. For one UniBoard equation Equation 7 then reduces to:

Equation 8:

Internally the Apertif FN beamformer contains PBF=4 parallel BF units that each process Nclk=256 beamlets to achieve Nbeamlet= PBF \* Nclk=1024 in total. From these Nclk beamlets only Nblk=240 are actually output [8]. With subscript *u* = 0:PBF-1 = 0:3 to indicate the parallel BF units and index *bu* = 0:Nblk-1 = 0:239 to count the number of valid beamlet slots per BF unit, then the relation with the absolute beamlet index *b* is given by:

Equation 9:

With Equation 9 the Equation 8 can be rewritten as:

Equation 10:

### Distribution on the UniBoard mesh

Next part to complete the Tdish and Tpol transpose is to bring all SP to one processing node (PN) on UniBoard. Therefore the beamlets that are received at each PN need to be split into nof\_un=8 parts, whereby one part is kept at this PN and the other nof\_un-1 = 7 parts are passed on via the UniBoard mesh to the other 7 PN on the UniBoard. Choose to distribute the beamlets over the nof\_un=8 in the order in which they arrive at the PN. The index *u* already provides range PBF=4 beamlets, therefore define an additional interleave subblock size of Ninterleave = nof\_un/PBF = 2 beamlets to be able to distribute the beamlets in order over the nof\_un=8 PN. The default block size in the Apertif BF is Nblk=240. The corresponding block size on the UniBoard for the Apertif X and for Arts BF is therefore call Mblk=Nblk/Ninterleave=120. Equation 10 can then be rewritten as:

Equation 11:

Whereby the relation between index *bu* and index *bi* = 0:Ninterleave-1 =0:1 and *bu\_i* = 0:Mblk-1 = 0:119 is given by:

Equation 12:

Define index *dest* = 0:nof\_un-1 = 0:7 for the destination PN as:

Equation 13:

With Equation 13 the Equation 11 can be rewritten as:

Equation 14:

Whereby the Ninterleave = 2 beamlets in series are unfolded as Pinterleave = Ninterleave = 2 parallel streams that run at 1/Pinterleave rate of fclk.

Together the indices *port* and *dest* cover the entire range of SP, because Nsp = nof\_10g \* nof\_un = 3\*8 = 24. The order of the indices *port* and *dest* can be swapped at no cost, because all streams are available in parallel within the PN FPGA. It is convenient to swap *port* and *dest*, because then the Nsp=24 streams are again in incrementing order. Therefore rewrite Equation 14 as:

Equation 15:

The beamlets stream for which index *pn* = *dest* remains on this PN and the beamlet streams for which *pn* != *dest* are received from the corresponding other PN via the UniBoard mesh. This transport operation across the UniBoard mesh implements the last part of the Tdish and Tpol transpose. Starting with Equation 15 this swaps the *dest* and *pn* indices and results in:

Equation 16:

With Equation 4 the Equation 16 can be expressed in terms of the SP index *sp* = 0:Nsp-1:

Equation 17:

### Per PN

One processing node (PN) on UniBoard processes 1/nof\_un = 1/8 part of the beamlets of one band of all Nsp=24 SP. Call this signal *cb\_node*, so *cb\_node* = *cb\_uniboarddest* where *dest* is this PN. For one PN Equation 17 then reduces to:

Equation 18:

The index *bu\_i* = 0:Mblk-1 = 0:119, so there are Mblk=120 beamlets per SP stream. The 1/Pinterleave = ½ rate implies that 2 SP can be folded (multiplexed) in per stream. Choose to fold the Npol=2 SP that form a dual polarization CB into one stream. Equation 18 then becomes:

Equation 19:

## Beamlet mapping in the Apertif BF

The beam direction and beam band of the Mblk=120 beamlets in Equation 19 depends on the two programmable reorder stages in the Apertif BF [8] as shown by Rsub and Rbeam in Figure 3. The reorder stage Rsub in the BN filterbank of the Apertif BF maps the subband frequencies to the band of NFN=24 frequencies that are output to one UniBoard in Aperitif X and Arts. The reorder stage Rbeam in the FN beamformer of the Apertif BF replicates these frequency slots to create KCB beamlets per frequency. The order of the Nbeamlets = KCB \* NFN = 960 beamlets in this reorder stage can eg. be set for maximum band per PN or for maximum FoV per PN:

1. For maximum band per PN the Mblk beamlets contain KPN = Mblk /NFN = KCB/nof\_un= 5 beamlet directions (so 5 out of the KCB = 40 CB) each covering a full band of NFN=24 beamlet frequencies.
2. For maximum FoV per PN the Mblk beamlets contain Mblk /KCB = NFN/nof\_un = 3 frequencies each covering a full FoV of KCB = 40 beamlet directions (so all CB).

For Arts the reorder stage in the FN beamformer of the Apertif BF needs to be set for full band per PN. Each PN can process KPN = 5 full band beamlet directions. Assume that the central CB is mapped to the first full band beamlet direction in PN0.

## Processing input and output

### Channel filterbank

For SC3 and SC4 the beamlets have to be separated into Nchan = 4 channels. This can be done with a poly-phase filterbank (PFB) that is also used for the subband filterbank in the BN of the Apertif BF. For Arts call the beamlet filterbank output *ch\_nodesp*, then with the PFB function described in appendix 11.2 applied to the beamlet inputs of Equation 19 yields:

Equation 20:

The range of index *ch* = 0:Nchan-1 = 0:3, so every 4 beamlet inputs in time yield 4 channel outputs in frequency. The channel rate is fchan = fsub/Nchan, so the channel time index *t\_ch* increments at a rate 1/Nchan compared to the beamlet time index *t*. Within Arts the channels are kept together, so therefore from a data transport point of view there is no difference in routing groups of Nchan beamlet time samples as in Equation 18 or routing Nchan channel frequency samples as in Equation 20, except for that the data width may differ due to Wchan ≥ Wbeamlet.

The number of parallel data streams that is needed to implement the entire channel filterbank for all Nsp=24 SP is MPFB\_units = 12 as given by:

Equation 21:

### Tied array beamformer (TAB)

The BF component that is used in the FN of the Apertif BF is a ‘voltage’ beamformer and can therefore also be reused to beamform the TABs in Arts. The BF weights and adds the Ndish = 12 inputs. For Arts call the TAB output *tab\_node*, then with the BF function described in appendix 11.4 applied to the beamlet inputs of Equation 19 yields:

Equation 22:

The complex output samples of TAB are called tab-lets. A tab-let is a tied array beamformed beamlet. For SC1 TAB-1 and SC2 TAB-12 the tab-lets are creating using *cb\_nodedish* as in Equation 19. For SC4 TAB-444 the tab-lets have to be created per channel, so then using *ch\_nodedish* from Equation 20 as input yields.

Equation 23:

In total Arts needs to make up to KTAB = 12 TAB per beamlet (i.e. per CB). From Equation 22 it follows that within one stream the beamlets cannot be replicated because the interleave factor Pinterleave is already used to fold the Npol = 2 polarizations. Therefore to have KTAB = 12 TABs the stream can simply be to be replicated KTAB times in parallel. Hence per PN each TAB needs one BF unit. Each BF unit takes the same *cb\_nodedish* input streams as described by Equation 22 and Equation 23. In total per PN there are MBF\_units = 12 BF units in parallel:

Equation 24:

Call the TAB Stokes power data *tab\_stokes\_node*. Applying Equation 1 to Equation 23 then yields:

Equation 25:

The range of index *st* = 0:NStokes/2-1 = 0:1 and represents the 4 integer powers of the Stokes vector that are packed with IQ in parallel and UV in parallel. The two parallel values are represented by the *(int\_2)* type cast. The TAB Stokes power data is integrated over every Nint ≈ 10 samples in time. Call the TAB Stokes power data *tab\_stokes\_int\_node*. Summing every Nint samples in Equation 26 then yields:

Equation 26:

where t\_int = (t\_ch – t\_ch\_offset) / Nint and t\_ch\_offset is typically 0.

### Incoherent array beamformer (IAB)

The IAB is defined by the Stokes vector [I, Q, U, V] which consists of real powers that are based on both polarizations X and Y. Call the CB stokes data *cb\_stokes\_node*. Applying Equation 1 to the beamlet channel inputs of Equation 20 yields:

Equation 27:

The IAB adds these powers for all Ndish=12 dishes. Call the IAB output *iab\_stokes\_node*. The summation by the IBF over the Ndish inputs then yields

Equation 28:

The range of index *st* = 0:NStokes/2-1 = 0:1 and represents the 4 integer powers of the Stokes vector that are packed with IQ in parallel and UV in parallel. The two parallel values are represented by the *(int\_2)* type cast. There is only 1 IAB per beamlet (i.e. per CB), so Equation 28 describes the entire IAB per PN. The full Stokes power output samples of IAB are called iab-lets. An iab-let is an incoherent array beamformed beamlet. The iab-let contains Nstokes = 4 real power values that are formed from the dual polarization complex beamlet values, so also Npol \* Ncomplex = 4 real values.

One IAB unit processes both Npol = 2 polarizations and there is only 1 IAB per beamlet. Therefore the number of parallel dual polarization data streams that is needed to implement the IAB per CB is:

Equation 29:

## SC1 TAB-1 output

For SC1 only one TAB for only the central CB needs to be output. The TAB-1 for SC1 uses the beamlets of the central CB as input. One PN can process KPN = 5 full band beamlet directions. The central CB can be mapped PN0 and mapped to the first of these beamlets directions of PN0 (section 5.3). For SC1 only one BF unit is needed. In this way the one TAB for the central CB is available via the first NFNpol = NFN \* Npol = 24\*2 = 48 tab-lets of the stream. Call the range *bo* = 0:NFN-1. Take range *bo* from range *bu\_i* in Equation 22. The SC1 TAB-1 tab-lets output per PN0 is then given by:

Equation 30:

In total there are Nband = Muni = 16 PN0 output nodes for SC1, so the total SC1 TAB-1 with range band = 0:Nband-1 becomes:

Equation 31:

The data rate per link conform Equation 30 is LTAB1\_band = NFN \*Npol \*Ncomplex \*Wtab \*Bsub = 24 \*2 \*2 \*4 \*781250 = 300 Mbps. This LTAB1\_band = 300 Mbps can be offloaded via the 1 GbE control interface of the UniBoards as shown in Figure 9. The total output load for SC1 conform Equation 31 is LTAB1 =Nband \* LTAB1\_band = 4.8 Gbps, which agrees with Table 3. Figure 9 shows in green that only PN0=FN0 is used for the TAB-1 processing of SC1, the other PN are only needed to pass on the data from the other SP to PN0 as indicated in yellow.



Figure 9: SC1 TAB1 output

## SC2 TAB-12 output

The TAB-12 for SC2 uses the beamlets of the central CB as input. Like for SC1 (section 5.5) the central CB can be mapped PN0 and mapped to the first NFNpol = NFN \* Npol = 24\*2 = 48 beamlets of PN0 (section 5.3). To have KTAB = 12 TABs the stream can simply be to be replicated KTAB times in parallel, similar as needed for SC4, rather than trying to make use of the Nblk – NFNpol = 240 – 48 = 192 beamlet slots that are also available. Define index range *tab* = 0:KTAB-1. Take range *bo* from range *bu\_i* in Equation 22. The SC2 TAB-12 tab-lets output per PN0 is then given by:

Equation 32:

In total there are Nband = Muni = 16 PN0 output nodes for SC2, so the total SC2 TAB-12 output with range band = 0:Nband-1 becomes:

Equation 33:

The data rate per link conform Equation 32 is LTAB12\_band = KTAB \*NFN \*Npol \*Ncomplex \*Wtab \*Bsub = 12 \*24 \*2 \*2 \*4 \*781250 = 3.6 Gbps. This LTAB12\_band = 3.6 Gbps can be offloaded via one 10 GbE output port of the PN0 on the UniBoards as shown in Figure 10. The total output load for SC2 conform Equation 33 is LTAB12 =Nband \* LTAB12\_band = 57.6 Gbps, which agrees with Table 3. Figure 10 shows in green that only PN0=FN0 is used for the TAB-12 processing of SC2, the other PN are only needed to pass on the data from the other SP to PN0 as indicated in yellow.



Figure 10: SC2 TAB12 output

## SC2 CB-12 output

For SC2 it is also required to be able to pass on and output the original central CB from the individual dishes. Like for SC1 (section 5.5) the central CB can be mapped PN0 and mapped to the first NFNpol = NFN \* Npol = 24\*2 = 48 beamlets of PN0 (section 5.3). Take range *bo* = 0:NFN-1 from range *bu\_i* in Equation 19. The SC2 CB-12 beamlets output per PN0 is then given by:

Equation 34:

In total there are Nband = Muni = 16 PN0 output nodes for SC2, so the total SC2 CB-12 with range band = 0:Nband-1 becomes:

Equation 35:

The data rate per link conform Equation 34 is LCB12\_band = Ndish \*NFN \*Npo l\*Ncomplex \*Wbeamlet \*Bsub = 12 \*24 \*2 \*2 \*6 \* 781250 = 5.4 Gbps. This LCB12\_band = 5.4 Gbps can be offloaded via one 10 GbE output port of the PN0 on the UniBoards as shown in Figure 11. The total output load for SC2 conform Equation 35 is LCB12 =Nband \* LCB12\_band = 86.4 Gbps, which agrees with Table 3. Figure 11 shows in that only PN0=FN0 is used for the CB-12 output of SC2. PN0 is drawn in yellow, because the CB-12 output does not require processing. The other PN are only needed to pass on the data from the other SP to PN0 as indicated in yellow.



Figure 11: SC2 CB12 output

## SC3 IAB-37 output

CB-444🡪UIQV 🡪Nint🡪 IAB-37 🡪 output ????

Figure 12 shows in green that all PN are used for the IAB-37 processing of SC3.



Figure 12: SC3 IAB37 output

## SC4 TAB-444 output

The TAB-444 for SC4 uses the channels of all CB as input. There are NCB = 37 CB and for each CB there are KTAB = 12 TABs so NTAB = KTAB \* KCB ≥ KTAB \* NCB = 444.

TAB-444🡪UIQV 🡪 Nint 🡪 output ????

Figure 13 shows in green that all PN are used for the TAB-444 processing of SC4.



Figure 13: SC4 TAB444 output

# Transient data capture

## External memory on UniBoard

Each PN on UniBoard has two DDR3 memories of maximum 16 GiByte each. In total the Arts BF has MPN = 128 PN so 4 TiByte.

## Capture integrated power data

For SC3 and SC4 the integrated Stokes power values need to be captured for 15 s (SR-0.35). The integrated power samples are output to the PL so therefore these transient capture buffers are implemented in the PL. for SC4 the load is given by LIAB\_stokes\_int = 17.75 Gbps and for SC4 the load is given by LTAB444\_stokes\_int = 213.12 Gbps.

## Capture CB voltage data

For SC3 the input ‘voltage’ samples of the CB-444 need to be captured for Tcapture = 10 s (SR 0.40). The ‘voltage’ samples need to be stored on the UniBoards, because they are not output to the PL. From Table 2 it follows that for Wbeamlet = 6 bit the total CB input load is LBF\_CB444 = 3.2 Tbps = 400 GByte/s. Hence if the Muni = 16 UniBoards of the Arts BF are equipped with 16 GiByte DDR3 memories then it has just sufficient storage to capture 10 s of CB-444 input ‘voltage’ beamlet data.

When the PL detects a transient then only the CB in which the transient occurred needs to be read. This amounts to reading Tcapture \* LBF\_CB444 / NCB / MPN = 10 \* 3.2 T / 37 / 128 = 6.7 Gbit per PN.

# Memory-mapped control interface

Apertif BF control (DT, Rsub, Rbeam, BF weights).

Apertif X control (none, except for adding status meta data to Apertif X output).

Arts BF:

* Default board control (as with unb1\_minimal reference design)
* Default IO test control (as with unb1\_test reference design)
* TAB control (BF weights)
* IAB control (none)
* CB-444 transient data buffer trigger and read out

Control rate:

* Only at start of measurement (e.g. reorder/selections)
* Occasionally (e.g. for each noise source calibration, read out transient capture buffers)
* Regularly during measurement (e.g. DT and BF coefficients)

# Processing

* Block diagram (IO terminals, BSN aligners + DSP for SC1, 2, 3, 4)
* New Stokes IQUV powers + intergation component
* Reuse PFB for channel filterbank
* Reuse BF for TAB
* New IBF component for IAB (= common\_adder\_tree)
* One image for all SC or one image revision per SC ????

# Storage

* 4, 8 or 16 G DDR3 modules
* Store on Arts Uniboards (not on Apertif BF UniBoards)
* Store voltage values or intergrated power values (Wpower = ????)
* Buffer size (seconds) ????
* Trigger control and read out control ????

# Verification

* Unb1\_test for IO
* Where to put BG and DB for Arts test ????

# Appendix : VHDL component resource usage and IO formats

## Available FPGA resources on UniBoard

### Multipliers

The Stratix IV FPGA on UniBoard has 1288 (18x18)-bit real multipliers. These multipliers are grouped per four to fit one complex multiplication. The factor Pcmult = 4 is used to define that 1 complex MAC requires 4 real MACs. For a complex multiplication all 4 real multipliers can be used, because a complex 18x18 bit multiply yields a complex 36x36 bit product that fits in the 72 bit output. Four independent real multiplications cannot fit, because they would need 144 bit output, whereas only 72 bit output is available per group of four (18x18)-bit real multipliers. Therefore a group of four (18x18) bit real multipliers can fit:

* 1 complex (18x18) bit multiply
* 2 independent real (18x18) bit multiplies
* 4 independent real (9x9) bit multiplies
* 2 complex (9x9) bit multiplies ???

### Memory

The Stratix IV FPGA on UniBoard has 1235 M9K RAM blocks. One M9K RAM block is 1024 x 9 bit large, so size is about 1 Kbyte or exactly SM9K = 9 kbit. The M9K RAM blocks can be used with various data widths (WM9K) and address depths (DM9K), whereby SM9K = DM9K \* WM9K. The maximum data width to access a M9K block is WM9K=36 bit and the depth is then at least DM9K=256 words, so 36x256. Other combinations are 18x512 and 9x1024. Translating number of bits into number of M9K is difficult to predict in advance because it is implementation dependent:

* The limited number of possible block RAM dimensions can cause that not all bits or words can be used.
* Parallel instances (streams) that have the same control can share the same RAM, but this may depend on whether the synthesis tool recognizes this optimization

Therefore in practice it is necessary to do a trial synthesis to get a more reliable estimate of the RAM.

## PFB component

### Multiplier usage

The PFB consists of two FIR filters with real coefficients and an NFFT points (complex) FFT. The number of (18x18)-bit multiplier elements per PFB instance is given by the sum of Equation 36 and Equation 37:

Equation 36:

Equation 37:

These Equation 36 and Equation 37 are partly empirical. The data rate per signal path is fdata and the processing clock rate is fclk. When fclk < fdata then the wideband factor fdata / fclk = P and the PFB becomes a wideband PFB (WPFB), as is the case for the subband filter in the Apertif BF. When fclk ≥ fdata and then fclk / fdata = Ninterleave is the number of multiplexed signal paths that can be processed by a single PFB component, as is the case for the channel filter in Arts. The number of PFB instances that is needed to implement the entire PFB for all SP is given by:

Equation 38:

The number of multipliers in the FIR part is equal to the number of taps. The FIR coefficients are real and the input per each FIR filter is also real, however this does not reduce the number of multipliers, because the fitter cannot make use of this to fit them in the FPGA when the operands are wider than 9 bit. Therefore in Equation 36 Ffit =2 when the multiplier operands are ≤ 9 bit and Ffit=1 when the operands are ≤ 18 bit.

In theory the FFT would require 0.5\*log2(NFFT) multipliers, but the radix-2 FFT implementation achieves only 50% utilization. The last stage of an FFT reduces to multiplication by ±1 which would yield log2(NFFT)-1, however the FFT implementation does not yet make use of that.

Table 4 shows the multiplier usage of the channel filterbank with Nchan=4 in the Arts BF. For comparison the resource usage of the similar channel filterbank with Nchan\_x=64 for the Aperif X are shown. The numbers for the Apertif X channel filterbank agree with fitter reports from FPGA synthesis. The number of multipliers can be reduced by using them with 9x9 bit operands (data and FIR coefficients) instead of 18x18bit operand as shown by respectively Ffit = 2 and Ffit = 1.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Application** | **Ntaps** | **Ffit** | **NFFT** | **FIRnof\_mult** | **FFTnof\_mult** | **PFBnof\_mult** |
| Arts BF | 8 | 1 | 4 | 24/2 \*8 \*4/1 = 384 | 24/2 \*log2(4)\*4 = 96 | 480 |
| Arts BF | 8 | 2 | 4 | 24/2 \*8 \*4/2 = 192 | 24/2 \*log2(4)\*4 = 96 | 288 |
| Apertif X | 8 | 1 | 64 | 24/2 \*8 \*4/1 = 384 | 24/2 \*log2(64)\*4 = 288 | 672 |
| Apertif X | 8 | 2 | 64 | 24/2 \*8 \*4/2 = 192 | 24/2 \*log2(64)\*4 = 288 | 480 |
|  |  |  |  |  |  |  |

Table 4: Multiplier usage for the channel filterbank (PFB) with MPFB\_units=24/2, Pcmult=4

### Memory usage

The FIR part of the PFB has complex input but real coefficients. The number of memory bits that the implementation of the FIR part for one instance of the PFB needs is given by:

Equation 39:

The FFT part of the PFB needs memory to store one block of data for the FFT operation and a dual-page memory in case the output of the FFT needs to be reordered. Therefore the number of memory bits that the implementation of the FFT part for one instance of the PFB needs is given by:

Equation 40:

The memory consists of RAM blocks that best fit either 9 bit data or 18 bit data. Therefore assume in Equation 39 en Equation 40 that Wcoeff and Wdata are 9 bit or 18 bit to be able to express then in number of M9K blocks. The data at different taps are processed in parallel and can share the same RAM blocks, but the maximum data width per RAM block is 36 bit, therefore define width WM9K= 36, 18 or 9 bit and corresponding size SM9K = 256, 512 or 1024 words. The coefficients and the data cannot share the same RAM blocks. With these assumptions Equation 39 and Equation 40 can be expressed in M9K RAM block units:

????

Equation 41:

Equation 42:

Equation 43:

Equation 44:

Equation 45:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Application** | **MPFB\_units,**  **Ninterleave,**  **Npage** | **Ntaps** | **NFFT** | **Wcoeff,**  **Wdata** | **FIR** | **FFT** | **PFB** |
| Apertif BF | 2, 1, 2 | 16 | 1024 | 18, 18 | 2\*(32+64)=192 |  |  |
| Apertif X | 12, 2, 0 | 8 | 4 | 18, 18 |  |  |  |
| Arts BF |  | 8 | 64 | 18, 18 |  |  |  |
|  |  |  |  |  |  |  |  |

Table 5: Memory usage for the channel filterbank (PFB) with Ncomplex=2

????

### Input - output

In general the PFB with input *s* and output *b* can be described by:

Equation 46:

The range of the frequency index *f*=0:N-1 and N is the number of frequencies. The actual order of index *f* may be scrambled due to the FFT in the PFB. For every N inputs in time there are N outputs in frequency, so output time index *t\_f* increments when input time index *t* has incremented by N.

The PFB component can also operate on multiplexed input signals in series. This can be represented by index *ser*=0:Nser-1 in:

Equation 47:

The PFB can also operate on input signals that have a sample rate fs that is a wideband factor P larger than the processing clock rate fclk. Typically Nser=1 when P>1. Therefore the WPFB can be represented by index *p*=0:P-1 in:

Equation 48:

The PFB has a processing gain that requires that the output samples have log2(N)/2 more bits than the input samples [5].

## Reorder component

The reorder component can:

* transpose multiplexed streams into parallel streams
* change the order of the input data
* can omit some input data
* replicate inputs

Replication of inputs can be used to eg. facilitate forming multiple beams per input.

### Memory usage

### Input - output

If the input *s* with Npar parallel streams (index *par*) and Nser multiplexed serial streams (index *ser*) and with Nf frequency samples (index *f*) per time block (index t) is:

Equation 49:

A serial to parallel transpose with index Nsp = *par* \* Nser + *ser* can be used to yield:

Equation 50:

A replication of KCB beams (index *k*) per frequency can be used to yield:

Equation 51:

The total block size must fit within a block processing interval, so Nf \* KCB ≤ Nclk clock cycles.

## BF unit component

The BF unit component consists of a reorder function (section 11.3) and a MAC function. The MAC weights each input SP and then adds them.

### Multiplier usage

The (18x18) multiplier usage of the MAC in the BF unit with S parallel inputs is:

Equation 52:

For TAB-12 and TAB-444 each PN needs to calculate KTAB = 12 TAB. The factor fclk/fdata = Ninterleave = 2 in Equation 52, so MBF\_units = KTAB \* Npol / Ninterleave = 12 BF units are needed in parallel.

Table 6 shows the multiplier usage of the ‘voltage’ beamformer in the Arts BF. For comparison the resource usage of the similar BF for the Aperif BF are shown. The numbers for the Apertif BF agree with fitter reports from FPGA synthesis. The number of multipliers can be reduced by using them with 9x9 bit operands (data and FIR coefficients) instead of 18x18bit operand as shown by respectively Ffit = 2 and Ffit = 1.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Application** | **MBF\_units** | **S** | **Npol** | **Ndish** | **Ninterleave** | **Ffit** | **BFnof\_mult** |
| Apertif BF | 4 | 64 | 1 | - | 1 | 1 | 4\*64\*4/1/1=1024 |
| Arts BF | 12 | - | 2 | 12 | 2 | 1 | 12\*24\*4/2/1 = 576 |
| Arts BF | 12 | - | 2 | 12 | 2 | 2 | 12\*24\*4/2/2 = 288 |
|  |  |  |  |  |  |  |  |

Table 6: Multiplier usage for the ‘voltage’ beamformer (BF) with Pcmult=4

### Memory usage

The BF Mac does not use RAM. If the BF power statistics are needed then these use a few M9K RAM blocks per BF unit.

### Input - output

With Equation 51 the BF component with input *s* and output *b* can be described by:

Equation 53:

Per time block the BF needs Nsp \* N \* KCB weights. Typically the same weights can be used for multiple time blocks. The weights are updated via the MM interface.

## IBF unit component