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	

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: • B_{sub}=1 MHz change request • TAB, IAB data processing relations • CB12, TAB-1/12/444, IAB-37 output • DSP resource usage (multipliers, RAM)

Table of contents:

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] “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
[4] “Detailed Design of the Digital Beamformer System for Apertif”, ASTRON-RP-413, G. Schoonderbeek, A. Gunst, E. Kooistra
[5] “Apertif Beamformer Firmware Description”, ASTRON-RP-1377, D. vd Schuur
[6] “Detailed Design of the Correlator System for Apertif”, ASTRON-RP-1400, G. Schoonderbeek, A. Gunst, E. Kooistra, HJ Pepping
[7] “Representing Streaming Data”, ASTRON-RP-1486, E. Kooistra
[8] “Apertif Beamformer Output Interface Specification”, ASTRON-SP-061, E. Kooistra
[9] “Understanding Digital Signal Processing”, 3^{rd} edition, R. Lyons
[10] “Apertif Filterbank Firmware Specification Part 2”, ASTRON-SP-054, E. Kooistra
[11] "Selection and Reorder modules", ASTRON-RP-1399, H.J. Pepping
[12] “Uthernet interface specification”, ASTRON-SP-041, E. Kooistra
[13] “Data path packet interface specification”, ASTRON-SP-042, E. Kooistra

Terminology:

ADC	Analogue to Digital Conversion
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
F_{sub}	Subband filterbank in Apertif BF
F_{chan_x}	Channel filterbank in Apertif X
F_{chan}	Channel filterbank in Arts BF for SC3 and SC4
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 (BN or FN)
PPS	Pulse Per Second
Re	Real
RF	Radio Frequency
R_{sub}	Reorder and select subbands for CB_{BW}=300 MHz in aperitif BF

R_{beam}	Reorder and select beamlets for N_{CB}=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 N_{pol} = 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
T_{ant}	Transpose to group data from all S = 64 (≥ N_{ant}) antenna elements in the FPA
T_{dish}	Transpose to group data from all N_{dish} = 12 dishes
T_{pol}	Transpose to group data from both N_{pol} = 2 polarizations
T_{sp}	Transpose to group data from all N_{sp} = N_{pol} * N_{dish} signal paths, so combines T_{dish} and T_{pol}
T_{band}	Transpose to group data from all N_{band} = 16 bands
$T_{integration}$	Transpose to group data from an integration interval of N_{int_x} values in time
T_{FoV}	Transpose to group data from all N_{CB} = 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: X_{re}, X_{im}, Y_{re}, Y_{im}
WSRT	Westerbork Synthesis Radio Telescope
X	Correlator

Definitions:

$N_{complex}$	2	Two part of a complex number, the real and imaginary part
N_{pol}	2	Number of polarizations, X and Y
N_{Stokes}	4	Number of power values in the Stokes vector [I, Q, U, V]
N_{dish}	12	Number of WSRT dishes in Apertif
N_{sp}	24	Number of signal paths = N_{dish} * N_{pol} at the output of the Apertif BF
N_{ant}	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 (≥ N_{ant})
S_{BN}	4	Number of ADC signal paths per BN in the frontend FPA of the Apertif BF
f_s	800 MHz	Digitizer sample frequency of the ADC at the Apertif BF frontend
T_s	1.25 ns	= 1/ f_s, digitizer sample period
f_0		Lower edge frequency of a subband, beamlet or channel [1]
RF_{BW}	400 MHz	= f_s/2, sampled RF bandwidth
CB_{BW}	300 MHz	Full bandwidth of the CB and also of the TAB and IAB (SR-0.2)
B_{sub}	781250 Hz	Subband bandwidth in Apertif BF, = beamlet bandwidth
f_{clk}	200M	Data processing clock rate in the FPGA, = f_s / P
P	4	Wideband rate factor of sample clock rate divided by digital processing clock rate
P_{cmult}	4	Number of real multiplications per complex multiplication
N	1024	FFT size of the FFT in the Apertif BF subband polyphase filter
N_{clk}	256	= N/P, number of DP clock cycles per subband period
N_{sub}	512	= N/2, number of subbands that covers RF_{BW}=400MHz
N_{sel}	384	Number of selected subbands to cover CB_{BW}=300 MHz
N_{band}	16	= nof_fn_bf, Number of bands in the Apertif BF to process the full CB_{BW}
N_{FN}	24	= N_{sel}/N_{band} , number of subbands per band or per FN in the Apertif BF
N_{FNpol}	48	= N_{FN}*N_{pol}, number of subbands per dual polarization band
N_{CB}	37	Required number of compound beams
K_{CB}	40	Implemented average number of beamlets per subband (≥ N_{CB})
P_{BF}	4	Number of parallel BF units per FN in the Apertif BF
N_{blk}	240	≤ N_{clk}, number of valid DP clock cycles per subband period in the Apertif BF

$N_{beamlet}$	960	Number of compound beamlet slots per FN output, maximum P_{BF} * N_{clk} = 1024, actual P_{BF} * N_{blk} = 960, required N_{CB} * N_{FN} = 888
$N_{interleave}$	2	= nof_un/P_{BF}, additional beamlet output interleave factor
$P_{interleave}$	2	Number of interleaved streams that are unfolded in parallel at rate 1/$P_{interleave}$
M_{blk}		=N_{blk}/$N_{interleave}$, default block size per subband period in the Apertif X and Arts
N_{gr}	12	Required number of TAB grating lobe patterns to cover the full CB (SR-0.41)
N_{VLBI}	12	Required number of TABs in the central CB for VLBI, choose = N_{gr} (SR-0.23)
K_{TAB}	12	Implemented number of TABs per beamlet (≥ N_{gr})
N_{TAB}	444	= N_{CB}*K_{TAB}, number of TABs
N_{IAB}	37	= N_{CB}, number of IABs
N_{link}	384	= N_{PN}, number of physical 10G output links of the Apertif BF, so 1 link per PN
N_{PN}	384	= N_{sp} * N_{band}, total number of parallel processing nodes in the Apertif BF
M_{PN}	128	= N_{band} * nof_un, total number of parallel processing nodes in the Arts
M_{PFB_units}	12	Number of parallel PFB units per PN for N_{sp} = 24 SP
M_{BF_units}	12	Number of parallel BF units per PN for K_{TAB} = 12 TABs
M_{IBF_units}	1	Number of parallel dual polarization IBF units per PN for the one IAB per CB
N_{chan_x}	64	Number of channels per beamlet in the Apertif X
N_{chan}	4	Number of channels per beamlet, for SC3 and SC4
B_{chan}		= B_{sub}/N_{chan}, channel bandwidth within a beamlet, for SC3 and SC4
N_{int_x}	800000	Number of channel power values that are integrated in the Apertif X
N_{int}	≈ 10	Number of Stokes channel power values that are integrated in Arts
T_{Stokes}	≈ 50 μs	Minimum required sample period for the Stokes power values
f_{Stokes}	≈ 20 kHz	= 1/T_{Stokes}, 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
$W_{beamlet}$	6	Word width in number of bits of a beamlet voltage sample
W_{chan}	7	Word width in number of bits of a channel voltage sample
W_{tab}	4	Word width in number of bits of a TAB voltage sample
W_{power}	4	Word width in number of bits of a IAB or TAB power sample

1 Introduction

1.1 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].

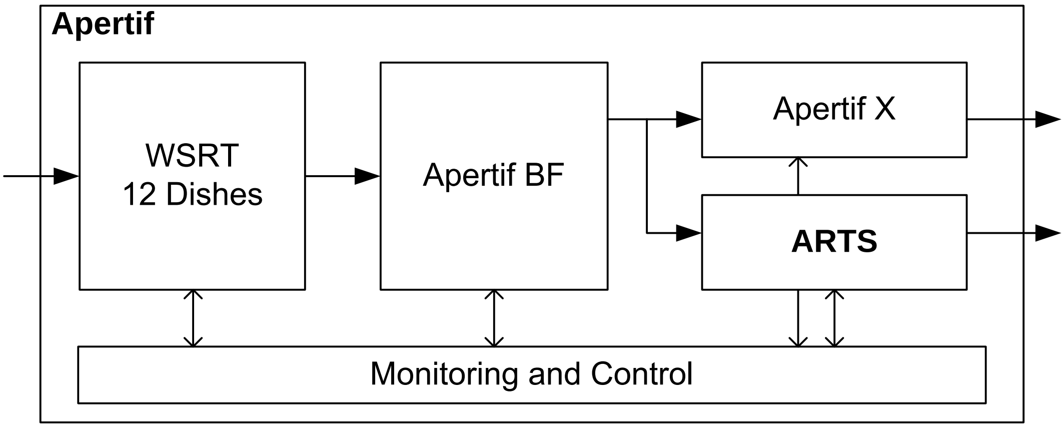

Figure 1: Top level overview of Apertif with Arts included

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.

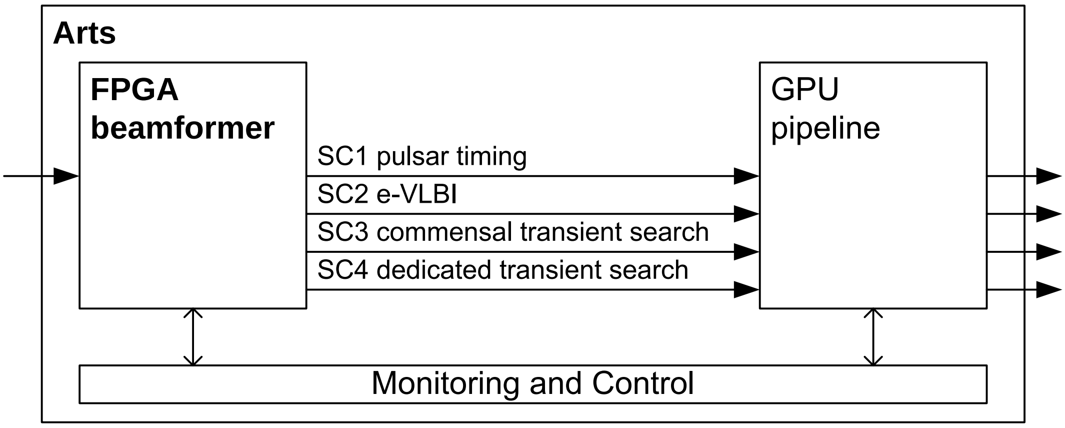

Figure 2: Arts FPGA beamformer and GPU pipeline

This document specifies the detailed design for the Arts FPGA beamformer (BF) on UniBoard FPGAs. At the input interface the Arts BF receives N_{CB}=37 compound beams from N_{dish}=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.

1.2 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].

1.3 Change request B_{sub} = 1 MHz

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

Parameter	**B_{sub} = 781250 Hz**	**B_{sub} = 1 MHz**	**Remark**
N	1024	800	= f_s / B_{sub}, keep f_s = 800 MHz
N_{sub}	512	400	= N / $N_{complex}$
N_{sel}	384	304	N_{sel} * B_{sub} ≥ CB_{BW} = 300 MHz
N_{FN}	24	19	= N_{sel}/N_{band}
$N_{beamlet}$	960	760	= K_{CB} * N_{FN}, keep K_{CB} = 40
N_{clk}	256	200	= N / P, keep P = 4 and f_{clk} = f_s / P
N_{blk}	240	190	= $N_{beamlet}$ / P_{BF}
B_{chan}	195.3125 kHz	250 kHz	= B_{sub} / N_{chan}
B_{chan_x}	12.207 kHz	15.625 kHz	= B_{sub} / N_{chan_x}

Table 1: Parameter values that depend on B_{sub}

2 System overview

2.1 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 N_{band} = 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 T_{ant} that groups the subbands from all S = 64 antennas. A compound beam (CB) is formed by a group of N_{sel}= 384 beamlets all with the same direction that span CB_{BW} = 300 MHz. Figure 3 shows the filterbank F_{sub}, the transpose T_{ant} and the beamformer (BF) that is distributed over N_{band} nodes. The $T_{integration}$ 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.

= Processing (on FPGA)
= IO, reordering/selection
= Control (on PC)

DT	= Delay tracking
F_{sub}	= Filterbank
BF	= Beamformer
R_{sub}	= Reorder and select subbands (for 300 MHz bandwidth)
R_{beam}	= Reorder and select beamlets (for 37 beams)
T_{ant}	= Transpose to group all S=64 antennas in the FPA
$T_{integration}$	= Transpose to group all beamlets per integration interval

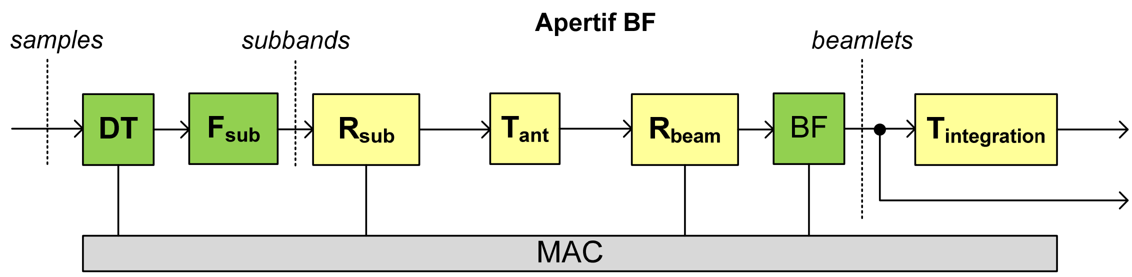

Figure 3: The Apertif BF subsystem

2.2 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 T_{dish} and T_{pol} transpose that are needed for Apertif X to group the data from all $N_{sp} = N_{pol} * N_{dish}$ = 24 signal paths can also be used for Arts.

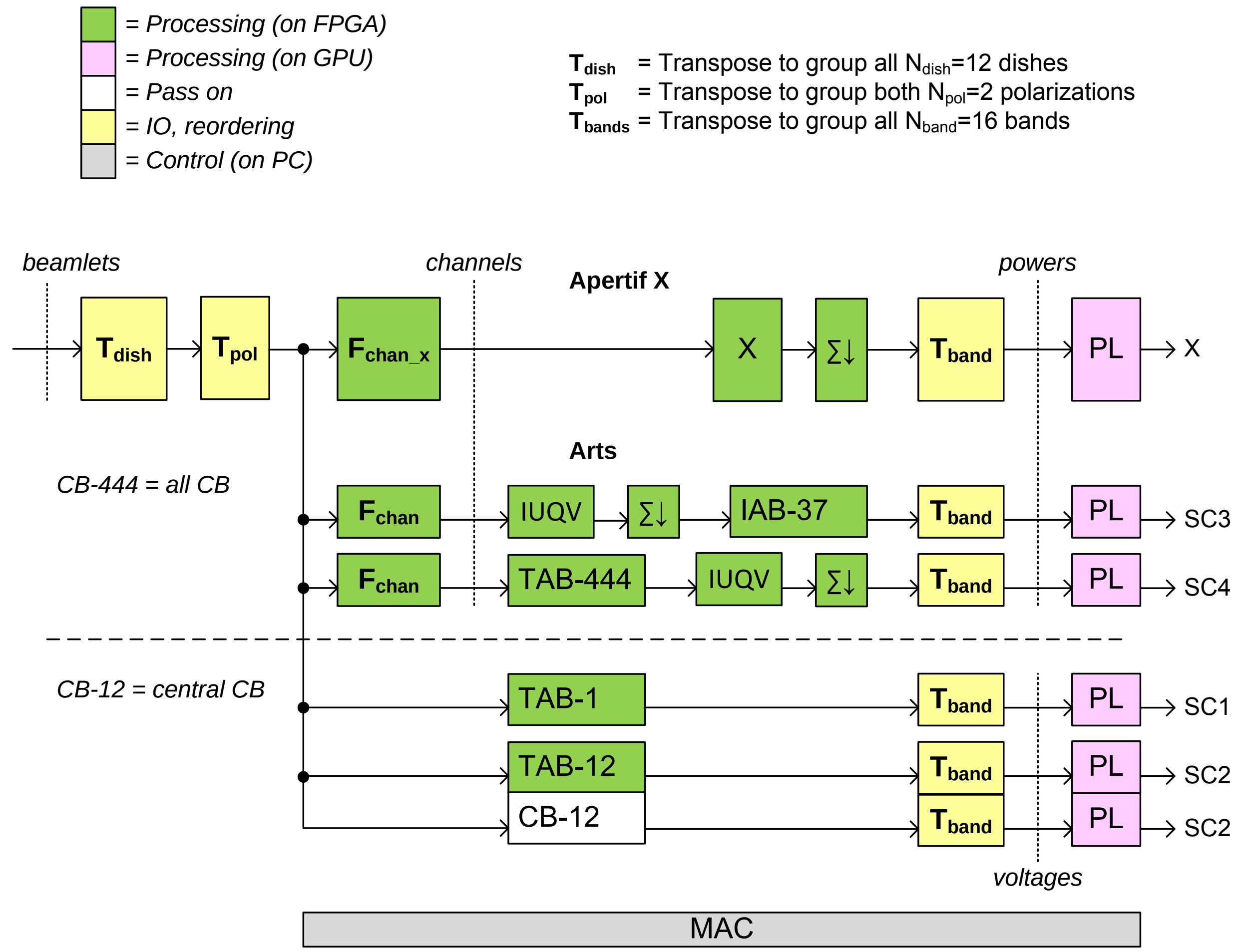

Figure 4: The Apertif X subsystem and the Arts subsystem

SC1 and SC2 use the central CB so they only need to process CB-12, where N_{dish} = 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 N_{VLBI} = 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 N_{dish} * N_{CB} = 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 K_{TAB} = 12 ‘voltage beams per CB, so K_{TAB} * N_{CB} = 12 * 37 = 444. Both SC3 and SC4 output Stokes (IQUV) ‘power’ beam data to the pipelining (PL).

3 Hardware architecture

3.1 Apertif BF using Uniboard

The Apertif BF outputs N_{CB}=37 compound beams with CB_{BW}=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 N_{band}= nof_fn_bf =16 frequency bands. Figure 5 shows the Apertif BF subrack with 4 UniBoards. Each FN in the subrack uses one 10GbE port to output its frequency part of the signal path.

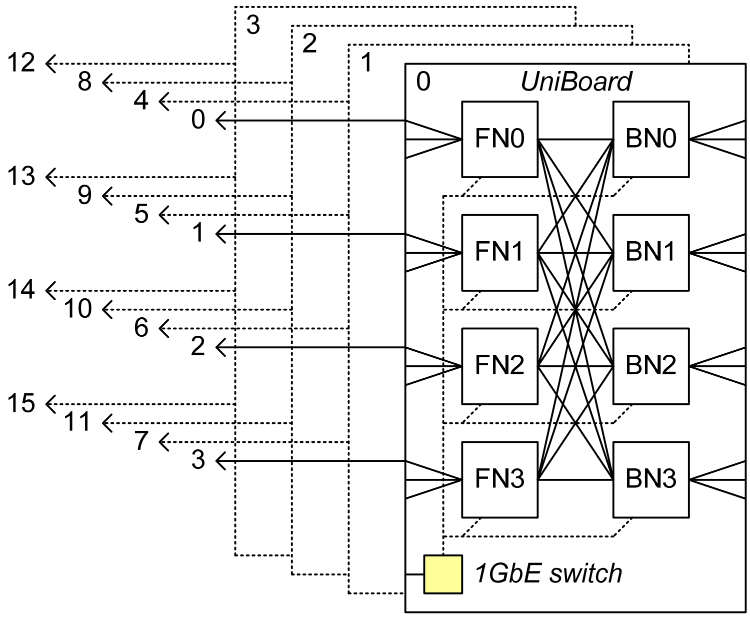

Figure 5: One Apertif BF subrack per signal path with nof_uni=4 UniBoards and N_{band}=16 FN

3.2 Signal path transpose to Arts

In total the Apertif BF has N_{sp} = N_{pol} * N_{dish} = 24 signal paths so also 24 subracks, 96 UniBoards and N_{PN}=N_{sp}*N_{band} = 384 processing (front) nodes. Hence the total Apertif BF output is carried via N_{link}=N_{PN}=384 10GbE links as shown in Figure 6. For both the Apertif X and for Arts the N_{sp}=24 signal paths from the Apertif BF need to be transposed to gather them together. This transpose T_{sp} can be implemented by interconnecting the Apertif BF to N_{band}=16 Uniboards as shown in Figure 6. Each of the N_{band}=16 UniBoards in Figure 6 processes 1/N_{band} part of the CB_{BW} band for all N_{sp}=24 signal paths.

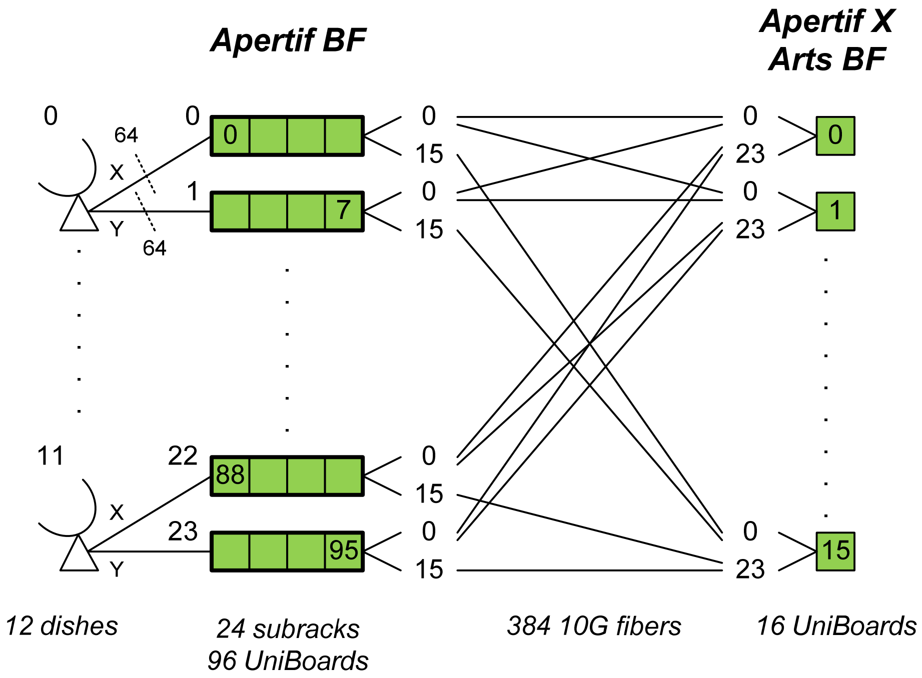

Figure 6: Apertif BF transpose interconnect to Apertif X and Arts

3.3 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 N_{sp}=24 links. Hence the number of input links determines that all nof_un = 8 FPGAs on UniBoard have to be used for Arts.

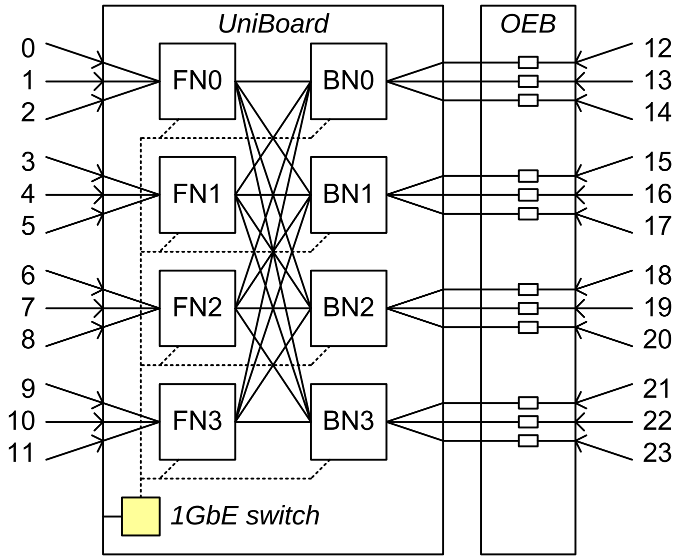

Figure 7: One UniBoard to process 1/N_{band} part of the CB_{BW} for N_{sp}=24 signal paths

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 N_{sp}=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.

4 Data path processing

4.1 Channel filterbank

For SC3 and SC4 the beamlet bandwidth of B_{sub} is too wide so therefore an additional channel filter is needed that separates the beamlets into N_{chan} = 4 channels. The channel filter is indicated by F_{chan} in Figure 4.

The channel filter is implemented by a poly-phase filterbank (PFB). 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 N_{taps}=16. However for the channel filters in Arts using N_{taps}=16 may take too many multiplier resources, therefore set N_{taps}=8.

4.2 Tied array beamformer (TAB) – ‘voltage’ beams

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

4.4 Requantization

5 Streaming data interfaces

5.1 System load overview

5.1.1 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 $W_{beamlet}$ = 6 bits.

Load	Equation	Value	Description
L_{BF_SP1}	$= CB_{BW} * N_{complex} * W_{beamlet}$	3.6 Gbps	Load for 1 SP
$L_{BF\_SP1\_band}$	$= L_{BF\_SP1} / N_{band}$	225 Mbps	Load for 1 SP per band (= per BF node)
$L_{BF\_SP37\_band}$	$= N_{CB} * L_{BF\_SP1\_band}$	8.325 Gbps	Load for N_{CB} = 37 SP per band (= per BF node)
L_{BF_CB1}	$= N_{pol} * L_{BF\_SP1}$	7.2 Gbps	Load for 1 CB (= 2 SP, N_{pol} = 2)
L_{BF_CB12}	$= N_{dish} * L_{BF\_CB1}$ $= N_{PN} * L_{BF\_SP1\_band}$	86.4 Gbps	Total load from N_{dish} = 12 dishes, for 1 CB
L_{BF_CB444}	$= N_{CB} * L_{BF\_CB12}$ $= N_{PN} * L_{BF\_SP37\_band}$	3.2 Tbps	Total load from N_{dish} = 12 dishes, for N_{CB}= 37 CB
L_{BF_link1}	$= L_{BF\_SP1\_band}$	225 Mbps	Link load for 1 SP per band (= per BF node)
L_{BF_link37}	$= L_{BF\_SP37\_band}$	8.325 Gbps	Link load for N_{CB}= 37 SP per band (= per BF node)

Table 2: Load definitions for Apertif BF output interface with $W_{beamlet}$ = 6 bit

5.1.2 Arts BF output load to Arts PL

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

Load	SC	Equation	Value	Description
L_{TAB1}	1	$= CB_{BW} * N_{pol} * N_{complex} * W_{tab}$	4.8 Gbps	Load for voltage TAB-1
L_{TAB1_band}	1	$= L_{TAB1} / N_{band}$	300 Mbps	Load for voltage TAB-1 per PN0
L_{TAB12}	2	$= N_{VLBI} * L_{TAB1}$	57.6 Gbps	Load for voltage TAB-12
L_{TAB12_band}	2	$= L_{TAB12} / N_{band}$	3.6 Gbps	Load for voltage TAB-12 per PN0
L_{CB12}	2	$= L_{BF\_CB12}$	86.4 Gbps	Load for voltage CB-12
L_{CB12_band}	2	$= L_{CB12} / N_{band}$	5.4 Gbps	Load for voltage CB-12 per PN0
L_{CB444}	3,4	$= N_{CB} * L_{CB12}$	3.1968 Tbps	Load for voltage CB-444
L_{IAB1_stokes}	3	$= CB_{BW} * N_{Stokes} * W_{power}$	4.8 Gbps	Load for IAB-1 without integration
$L_{IAB1\_stokes\_int}$	3	$= L_{IAB1\_stokes} / N_{int}$	0.48 Gbps	Load for IAB-1 with $T_{stokes} \approx 50$ μs
L_{IAB37_stokes}	3	$= N_{CB} * L_{IAB1\_stokes}$	177.6 Gbps	Load for IAB-37 without integration
$L_{IAB37\_stokes\_int}$	3	$= N_{CB} * L_{IAB1\_stokes\_int}$	17.76 Gbps	Load for IAB-37 with $T_{stokes} \approx 50$ μs
$L_{IAB37\_stokes\_I\_int}$	3	$= L_{IAB37\_stokes\_int} / N_{Stokes}$	4.44 Gbps	Load for IAB-37-I with $T_{stokes} \approx 50$ μs
$L_{TAB1\_stokes\_int}$	4	$= L_{IAB1\_stokes\_int}$	0.48 Gbps	Load for TAB-1 with $T_{stokes} \approx 50$ μs
L_{TAB444}	4	$= N_{CB} * N_{gr} * L_{TAB1}$	2.1312 Tbps	Load for voltage TAB-444
L_{TAB444_stokes}	4	$= L_{TAB444} * (W_{tab}/W_{power})$	2.1312 Tbps	Load for power TAB-444
$L_{TAB444\_stokes\_int}$	4	$= N_{CB} * N_{gr} * L_{TAB1\_stokes\_int}$	213.12 Gbps	Load for TAB-444 with $T_{stokes} \approx 50$ μs
$L_{TAB444\_stokes\_I\_int}$	4	$= L_{TAB444\_stokes\_int} / N_{Stokes}$	53.28 Gbps	Load for TAB-444-I with $T_{stokes} \approx 50$ μs

Table 3: Load definitions for Arts BF output interface (with $W_{beamlet}$ = 6 bit, W_{tab} = 4 bit, W_{power} = 4 bit)

5.2 CB signal path input

5.2.1 Entire system

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

Equation 1: $(cint6)CB_{dish,\ pol,\ band}[t][b]$

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

Equation 2: $(cint6)CB_{band,\ pol,\ dish}[t][b]$

The subscript indices indicate parallel links and the array index contains serial data on the link. The subscript *band* has range 0:N_{band}-1, subscript *pol* has range 0:N_{pol}-1, subscript *dish* has range 0:N_{dish}-1. In total there are $N_{band} * N_{pol} * N_{dish} = 16 * 2 * 12 = N_{link} = 384$ parallel links. The array index *t* increments at the rate of B_{sub}. The array index *b* has range 0:$N_{beamlet}$-1 where $N_{beamlet}$ is the number of compound beamlet slots per FN output of the Apertif BF. Required $N_{beamlet} \geq N_{CB}*N_{FN} = 37 * 24 = 888$, where $N_{FN} = N_{sel} / N_{band}$. The actual number of beamlet slots that will be implemented is $N_{beamlet} = K_{CB} * N_{FN} = 40 * 24 = 960$. 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:N_{sp}-1 = 0:23 according to:

Equation 3: $$sp = dish + pol * N_{dish} = port + dish * nof\_10G$$

Equation 4: $$pol = sp\ MOD\ N_{pol}$$ $$dish = sp\ DIV\ N_{pol}$$

Equation 5: $$port = sp\ MOD\ nof\_10G$$ $$pn = sp\ DIV\ nof\_10G$$

With Equation 3 the Equation 2 can be rewritten as:

Equation 6: $$(cint6)CB_{band,\ pn,\ port}[t][b]$$

5.2.2 Per UniBoard

One UniBoard processes one band of all N_{sp}=24 SP. Call this signal cb_uniboard, so cb_uniboard = CB_{band} where a *band* is mapped to this UniBoard. For one UniBoard equation Equation 6 then reduces to:

Equation 7: $$(cint6)cb\_uniboard_{pn,\ port}[t][b]$$

Internally the Apertif FN beamformer contains P_{BF}=4 parallel BF units that each process N_{clk}=256 beamlets to achieve $N_{beamlet}$= P_{BF} * N_{clk}=1024 in total. From these N_{clk} beamlets only N_{blk}=240 are actually output [8]. With subscript u = 0:P_{BF}-1 = 0:3 to indicate the parallel BF units and index bu = 0:N_{blk}-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 8: $$b = u * N_{clk} + bu$$

With Equation 8 the Equation 7 can be rewritten as:

Equation 9: $$(cint6)cb\_uniboard_{pn,\ port,\ u}[t][bu]$$

5.2.3 Distribution on the UniBoard mesh

Next part to complete the T_{dish} and T_{pol} 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 P_{BF}=4 beamlets, therefore define an additional interleave subblock size of $N_{interleave}$ = nof_un/P_{BF} = 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 N_{blk}=240. The corresponding block size on the UniBoard for the Apertif X and for Arts BF is therefore call M_{blk}=N_{blk}/$N_{interleave}$=120. Equation 9 can then be rewritten as:

Equation 10: $$(cint6)cb\_uniboard_{pn,\ port,\ u}[t][bu\_i][bi]$$

Whereby the relation between index *bu* and index *bi* = 0:$N_{interleave}$-1 =0:1 and *bu_i* = 0:M_{blk}-1 = 0:119 is given by:

Equation 11: $\boldsymbol{bu = bu\_i + bi * N_{interleave}}$

Define index *dest* = 0:nof_un-1 = 0:7 for the destination PN as:

Equation 12: $\boldsymbol{dest = u + bi * N_{interleave}}$

With Equation 12 the Equation 10 can be rewritten as:

Equation 13: $\boldsymbol{(cint6)cb\_uniboard_{pn,\ port,\ dest}[t][bu\_i] \quad @\ 1/P_{interleave}}$

Whereby the $N_{interleave}$ = 2 beamlets in series are unfolded as $P_{interleave}$ = $N_{interleave}$ = 2 parallel streams that run at 1/$P_{interleave}$ rate of f_{clk}.

Together the indices *port* and *dest* cover the entire range of SP, because N_{sp} = 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 N_{sp}=24 streams are again in incrementing order. Therefore rewrite Equation 13 as:

Equation 14: $\boldsymbol{(cint6)cb\_uniboard_{pn,\ dest,\ port}[t][bu\_i] \quad @\ 1/P_{interleave}}$

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 T_{dish} and T_{pol} transpose. Starting with Equation 14 this swaps the *dest* and *pn* indices and results in:

Equation 15: $\boldsymbol{(cint6)cb\_uniboard_{dest,\ pn,\ port}[t][bu\_i] \quad @\ 1/P_{interleave}}$

With Equation 3 the Equation 15 can be expressed in terms of the SP index *sp* = 0:N_{sp}-1:

Equation 16: $\boldsymbol{(cint6)cb\_uniboard_{dest,\ sp}[t][bu\_i] \quad @\ 1/P_{interleave}}$

5.2.4 Per PN

One processing node (PN) on UniBoard processes 1/nof_un = 1/8 part of the beamlets of one band of all N_{sp}=24 SP. Call this signal cb_node, so cb_node = $\text{cb\_uniboard}_{dest}$ where *dest* is this PN. For one PN Equation 16 then reduces to:

Equation 17: $\boldsymbol{(cint6)cb\_node_{sp}[t][bu\_i] \quad @\ 1/P_{interleave}}$

The index *bu_i* = 0:M_{blk}-1 = 0:119, so there are M_{blk}=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 17 then becomes:

Equation 18: $\boldsymbol{(cint6)cb\_node_{dish}[t][bu\_i][pol]}$

5.3 Beamlet mapping in the Apertif BF

The beam direction and beam band of the M_{blk}=120 beamlets in Equation 18 depends on the two programmable reorder stages in the Apertif BF [8] as shown by R_{sub} and R_{beam} in Figure 3. The reorder stage R_{sub} in the BN filterbank of the Apertif BF maps the subband frequencies to the band of N_{FN}=24 frequencies that are output to one UniBoard in Aperitif X and Arts. The reorder stage R_{beam} in the FN beamformer of the Apertif BF replicates these frequency slots to create K_{CB} beamlets per frequency. The order of the $N_{beamlets}$ = K_{CB} * N_{FN} = 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 M_{blk} beamlets contain K_{PN} = M_{blk} /N_{FN} = K_{CB}/nof_un= 5 beamlet directions (so 5 out of the K_{CB} = 40 CB) each covering a full band of N_{FN}=24 beamlet frequencies.
2. For maximum FoV per PN the M_{blk} beamlets contain M_{blk} /K_{CB} = N_{FN}/nof_un = 3 frequencies each covering a full FoV of K_{CB} = 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 K_{PN} = 5 full band beamlet directions. Assume that the central CB is mapped to the first full band beamlet direction in PN0.

5.4 Processing input and output

5.4.1 Channel filterbank

For SC3 and SC4 the beamlets have to be separated into N_{chan} = 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_node$_{sp}$, then with the PFB function described in appendix 10.2 applied to the beamlet inputs of Equation 18 yields:

Equation 19: $\boldsymbol{(cint)ch\_node_{dish}[t\_ch][ch][bu\_i][pol]}$

The range of index *ch* = 0:N_{chan}-1 = 0:3, so every 4 beamlet inputs in time yield 4 channel outputs in frequency. The channel rate is f_{chan} = f_{sub}/N_{chan}, so the channel time index *t_ch* increments at a rate 1/N_{chan} 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 N_{chan} beamlet time samples as in Equation 17 or routing N_{chan} channel frequency samples as in Equation 19, except for that the data width may differ due to $W_{chan} \geq W_{beamlet}$.

The number of parallel data streams that is needed to implement the entire channel filterbank for all N_{sp}=24 SP is M_{PFB_units} = 12 as given by:

Equation 20: $M_{PFB\ units} = N_{dish} = N_{sp}/N_{pol} = N_{sp}/N_{interleave}$

5.4.2 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 N_{dish} = 12 inputs. For Arts call the TAB output tab_node, then with the BF function described in appendix 10.4 applied to the beamlet inputs of Equation 18 yields:

Equation 21: $\boldsymbol{(cint)tab\_node[t][bu\_i][pol] = BF\{cb\_node_{dish}[t][bu\_i][pol]\}}$

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_node_{dish} as in Equation 18. For SC4 TAB-444 the tab-lets have to be created per channel, so then using ch_node_{dish} from Equation 19 as input yields.

Equation 22: $(\boldsymbol{cint})\boldsymbol{tab\_node}[\boldsymbol{t\_ch}][\boldsymbol{ch}][\boldsymbol{bu\_i}][\boldsymbol{pol}] = \boldsymbol{BF}\{\boldsymbol{ch\_node_{dish}}[\boldsymbol{t\_ch}][\boldsymbol{ch}][\boldsymbol{bu\_i}][\boldsymbol{pol}]\}$

In total Arts needs to make up to K_{TAB} = 12 TAB per beamlet (i.e. per CB). From Equation 21 it follows that within one stream the beamlets cannot be replicated because the interleave factor $P_{interleave}$ is already used to fold the N_{pol} = 2 polarizations. Therefore to have K_{TAB} = 12 TABs the stream can simply be to be replicated K_{TAB} times in parallel. Hence per PN each TAB needs one BF unit. Each BF unit takes the same cb_node_{dish} input streams as described by Equation 21 and Equation 22. In total per PN there are M_{BF_units} = 12 BF units in parallel:

Equation 23: $M_{BF\ units} = K_{tab} = K_{tab} * N_{pol}/N_{interleave}$

5.4.3 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. The IAB adds these powers for all N_{dish}=12 dishes. Call the IAB output iab_node. The IAB applied to the beamlet channel inputs of Equation 19 yields:

Equation 24: $(int2)iab\_node[\boldsymbol{t\_ch}][\boldsymbol{ch}][\boldsymbol{bu\_i}][\boldsymbol{iq\_uv}] = \boldsymbol{IAB}\ \{\boldsymbol{ch\_band_{dish}}[\boldsymbol{t\_ch}][\boldsymbol{ch}][\boldsymbol{bu\_i}][\boldsymbol{pol}]\}$

The range of index *iu_qv* = 0:N_{Stokes}/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 *(int2)* type cast. There is only 1 IAB per beamlet (i.e. per CB), so Equation 24 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 N_{stokes} = 4 real power values that are formed from the dual polarization complex beamlet values, so also N_{pol} * $N_{complex}$ = 4 real values.

One IAB unit processes both N_{pol} = 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 25: $M_{IBF\ units} = 1$

5.5 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 K_{PN} = 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 N_{FNpol} = N_{FN} * N_{pol} = 24*2 = 48 tab-lets of the stream. Call the range *bo* = 0:N_{FN}-1. Take range *bo* from range *bu_i* in Equation 21. The SC1 TAB-1 tab-lets output per PN0 is then given by:

Equation 26: $(\boldsymbol{cint})\boldsymbol{tab1\_node\_output}[\boldsymbol{t}][\boldsymbol{bo}][\boldsymbol{pol}]$

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

Equation 27: $(\boldsymbol{cint})\boldsymbol{tab1\_output_{band}}[\boldsymbol{t}][\boldsymbol{bo}][\boldsymbol{pol}]$

The data rate per link conform Equation 26 is $L_{TAB1\_band} = N_{FN} * N_{pol} * N_{complex} * W_{tab} * B_{sub}$ = 24 *2 *2 *4 *781250 = 300 Mbps. This L_{TAB1_band} = 300 Mbps can be offloaded via the 1 GbE control interface of the UniBoards. The total output load for SC1 conform Equation 27 is $L_{TAB1} = N_{band} * L_{TAB1\_band}$ = 4.8 Gbps, which agrees with Table 3.

5.6 SC2 TAB-12 output

The TAB-12 for SC2 uses the beamlets of the central CB as input. Like for SC1 (section 5.7) the central CB can be mapped PN0 and mapped to the first $N_{FNpol} = N_{FN} * N_{pol}$ = 24*2 = 48 beamlets of PN0 (section 5.3). To have K_{TAB} = 12 TABs the stream can simply be to be replicated K_{TAB} times in parallel, similar as needed for SC4, rather than trying to make use of the $N_{blk} – N_{FNpol}$ = 240 – 48 = 192 beamlet slots that are also available. Define index range *tab* = 0:K_{TAB}-1. Take range *bo* from range *bu_i* in Equation 21. The SC2 TAB-12 tab-lets output per PN0 is then given by:

Equation 28: $(\boldsymbol{cint})\boldsymbol{tab12\_node\_output}_{\boldsymbol{tab}}[\boldsymbol{t}][\boldsymbol{bo}][\boldsymbol{pol}]$

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

Equation 29: $(\boldsymbol{cint})\boldsymbol{tab12\_output}_{\boldsymbol{band,tab}}[\boldsymbol{t}][\boldsymbol{bo}][\boldsymbol{pol}]$

The data rate per link conform Equation 28 is $L_{TAB12\_band} = K_{TAB} * N_{FN} * N_{pol} * N_{complex} * W_{tab} * B_{sub}$ = 12 *24 *2 *2 *4 *781250 = 3.6 Gbps. This L_{TAB12_band} = 3.6 Gbps can be offloaded via one 10 GbE output port of the PN0 on the UniBoards. The total output load for SC2 conform Equation 29 is $L_{TAB12} = N_{band} * L_{TAB12\_band}$ = 57.6 Gbps, which agrees with Table 3.

5.7 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 5.7) the central CB can be mapped PN0 and mapped to the first $N_{FNpol} = N_{FN} * N_{pol}$ = 24*2 = 48 beamlets of PN0 (section 5.3). Take range *bo* = 0:N_{FN}-1 from range *bu_i* in Equation 18. The SC2 CB-12 beamlets output per PN0 is then given by:

Equation 30: $(\boldsymbol{cint6})\boldsymbol{cb12\_node\_output}_{\boldsymbol{dish}}[\boldsymbol{t}][\boldsymbol{bo}][\boldsymbol{pol}]$

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

Equation 31: $(\boldsymbol{cint})\boldsymbol{cb12\_output}_{\boldsymbol{band,dish}}[\boldsymbol{t}][\boldsymbol{bo}][\boldsymbol{pol}]$

The data rate per link conform Equation 30 is $L_{CB12\_band} = N_{dish} * N_{FN} * N_{pol} * N_{complex} * W_{beamlet} * B_{sub}$ = 12 *24 *2 *2 *6 * 781250 = 5.4 Gbps. This L_{CB12_band} = 5.4 Gbps can be offloaded via one 10 GbE output port of the PN0 on the UniBoards. The total output load for SC2 conform Equation 31 is $L_{CB12} = N_{band} * L_{CB12\_band}$ = 86.4 Gbps, which agrees with Table 3.

5.8 SC4 TAB-444 output

The TAB-444 for SC4 uses the channels of all CB as input. There are N_{CB} = 37 CB and for each CB there are K_{TAB} = 12 TABs so $N_{TAB} = K_{TAB} * K_{CB} \geq K_{TAB} * N_{CB}$ = 444.

TAB-444→UIQV → Nint → output ????

5.9 SC3 IAB-37 output

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

6 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)

Control rate:
- Only at start of measurement (e.g. reorder/selections)
- Occasionally (e.g. for ease noise source calibration)
- Regularly during measurement (e.g. DT and BF coefficients)

7 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 ????

8 Storage

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

9 Verification

- Unb1_test for IO
- Where to put BG and DB for Arts test ????

10 Appendix : VHDL component resource usage and IO formats

10.1 Available FPGA resources on UniBoard

10.1.1 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 P_{cmult} = 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 ???

10.1.2 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 S_{M9K} = 9 kbit. The M9K RAM blocks can be used with various data widths (W_{M9K}) and address depths (D_{M9K}), whereby $S_{M9K} = D_{M9K} * W_{M9K}$. The maximum data width to access a M9K block is W_{M9K}=36 bit and the depth is then at least D_{M9K}=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.

10.2 PFB component

10.2.1 Multiplier usage

The PFB consists of two FIR filters with real coefficients and an N_{FFT} points (complex) FFT. The number of (18x18)-bit multiplier elements per PFB instance is given by the sum of Equation 32 and Equation 33:

Equation 32: $$FIR_{nof\ mult} = N_{taps} * P_{cmult} * \frac{1}{F_{fit}}$$

Equation 33: $$FFT_{nof\ mult} = \log_2(N_{FFT}) * P_{cmult}$$

These Equation 32 and Equation 33 are partly empirical. The data rate per signal path is f_{data} and the processing clock rate is f_{clk}. When $f_{clk} < f_{data}$ then the wideband factor f_{data} / f_{clk} = P and the PFB becomes a wideband PFB (WPFB), as is the case for the subband filter in the Apertif BF. When $f_{clk} \geq f_{data}$ and then $f_{clk} / f_{data} = N_{interleave}$ 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 34: $M_{PFB\ units} = N_{sp} * \frac{f_{data}}{f_{clk}} = N_{sp}/N_{interleave}$

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 32 F_{fit} =2 when the multiplier operands are ≤ 9 bit and F_{fit}=1 when the operands are ≤ 18 bit.

In theory the FFT would require 0.5*$\log_2(N_{FFT})$ 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 $\log_2(N_{FFT})$-1, however the FFT implementation does not yet make use of that.

Table 4 shows the multiplier usage of the channel filterbank with N_{chan}=4 in the Arts BF. For comparison the resource usage of the similar channel filterbank with N_{chan_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 F_{fit} = 2 and F_{fit} = 1.

Application	N_{taps}	F_{fit}	N_{FFT}	FIR_{nof_mult}	FFT_{nof_mult}	PFB_{nof_mult}
Arts BF	8	1	4	24/2 *8 *4/1 = 384	24/2 *$\log_2(4)$*4 = 96	480
Arts BF	8	2	4	24/2 *8 *4/2 = 192	24/2 *$\log_2(4)$*4 = 96	288
Apertif X	8	1	64	24/2 *8 *4/1 = 384	24/2 *$\log_2(64)$*4 = 288	672
Apertif X	8	2	64	24/2 *8 *4/2 = 192	24/2 *$\log_2(64)$*4 = 288	480

Table 4: Multiplier usage for the channel filterbank (PFB) with M_{PFB_units}=24/2, P_{cmult}=4

10.2.2 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 35: $FIR_{nof\ memory\ bits} = N_{taps} * N_{FFT} * \left(W_{coeff} + N_{interleave} * W_{data} * N_{complex}\right)$

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 36: $FFT_{nof\ memory\ bits} = N_{FFT} * \left(W_{twiddle} + N_{interleave} * \left(1 + N_{page}\right) * W_{data}\right) * N_{complex}$

The memory consists of RAM blocks that best fit either 9 bit data or 18 bit data. Therefore assume in Equation 35 en Equation 36 that W_{coeff} and W_{data} 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 W_{M9K}= 36, 18 or 9 bit and corresponding size S_{M9K} = 256, 512 or 1024 words. The coefficients and the data cannot share the same RAM blocks. With these assumptions Equation 35 and Equation 36 can be expressed in M9K RAM block units:

????

Equation 37: $FIR_{nof\ M9K\ coeff} = \left\lceil\frac{N_{FFT}}{S_{M9K}}\right\rceil * \left\lceil\frac{N_{taps}*W_{coeff}}{W_{M9K}}\right\rceil$

Equation 38: $FIR_{nof\ M9K\ data} = \left\lceil\frac{N_{interleave}*N_{FFT}}{S_{M9K}}\right\rceil * \left\lceil\frac{N_{taps}*N_{complex}*W_{data}}{W_{M9K}}\right\rceil$

Equation 39: $FFT_{nof\ M9K\ twiddle} = N_{FFT} * W_{twiddle} * N_{complex}$

Equation 40: $FFT_{nof\ M9K\ data} = N_{FFT} * N_{interleave} * W_{data} * N_{complex}$

Equation 41: $FFT_{nof\ M9K\ reorder} = N_{page} * \left\lceil \frac{N_{FFT} * N_{interleave}}{S_{M9K}} \right\rceil * \left\lceil \frac{W_{data} * N_{complex}}{W_{M9K}} \right\rceil$

Application	M_{PFB_units}, $N_{interleave}$, N_{page}	N_{taps}	N_{FFT}	W_{coeff}, W_{data}	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 $N_{complex}$=2

????

10.2.3 Input - output

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

Equation 42: $\boldsymbol{b[t\_f][f] = PFB\{s[t]\}}$

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:N_{ser}-1 in:

Equation 43: $\boldsymbol{b[t\_f][f][ser] = PFB\{s[t][ser]\}}$

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

Equation 44: $\boldsymbol{b_p[t\_f][f] = WPFB\{s_p[t]\}}$

The PFB has a processing gain that requires that the output samples have $\log_2(N)/2$ more bits than the input samples [5].

10.3 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.

10.3.1 Memory usage

10.3.2 Input - output

If the input *s* with N_{par} parallel streams (index *par*) and N_{ser} multiplexed serial streams (index *ser*) and with N_f frequency samples (index *f*) per time block (index t) is:

Equation 45: $$s_{par}[t][f][ser]$$

A serial to parallel transpose with index N_{sp} = *par* * N_{ser} + *ser* can be used to yield:

Equation 46: $$s_{sp}[t][f] \quad @\, 1/N_{ser}$$

A replication of K_{CB} beams (index *k*) per frequency can be used to yield:

Equation 47: $$s_{sp}[t][f][k]$$

The total block size must fit within a block processing interval, so N_f * K_{CB} ≤ N_{clk} clock cycles.

10.4 BF unit component

The BF unit component consists of a reorder function (section 10.3) and a MAC function. The MAC weights each input SP and then adds them.

10.4.1 Multiplier usage

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

Equation 48: $$BF_{nof\ mult} = S * P_{cmult} * \left(\frac{f_{data}}{f_{clk}}\right) * \frac{1}{F_{fit}}$$

For TAB-12 and TAB-444 each PN needs to calculate K_{TAB} = 12 TAB. The factor f_{clk}/f_{data} = $N_{interleave}$ = 2 in Equation 48, so M_{BF_units} = K_{TAB} * N_{pol} / $N_{interleave}$ = 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 F_{fit} = 2 and F_{fit} = 1.

Application	M_{BF_units}	S	N_{pol}	N_{dish}	$N_{interleave}$	F_{fit}	BF_{nof_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 P_{cmult}=4

10.4.2 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.

10.4.3 Input - output

With Equation 47 the BF component with input *s* and output *b* can be described by:

Equation 49: $\boldsymbol{b}[\boldsymbol{t}][\boldsymbol{f}][\boldsymbol{k}] = \boldsymbol{BF}\{\boldsymbol{s}_{\boldsymbol{sp}}[\boldsymbol{t}][\boldsymbol{f}][\boldsymbol{k}]\}$

Per time block the BF needs N_{sp} * N * K_{CB} weights. Typically the same weights can be used for multiple time blocks. The weights are updated via the MM interface.

10.5 IBF unit component