UniBoard^2 Design Document: A Low Spectral Resolution EVN Correlator for Continuum Processing

Deliverable 8.11
J Hargreaves, JIVE
25th June 2015

1.1 Terminology

DDR3,DDR4	Double Data Rate memory interface standards
DSP	Digital Signal Processing
ES	Engineering Sample
EVN	European VLBI Network
FPGA	Field Programmable Gate Array
FX	Correlator in which the data are first channelized and then cross-multiplied
HDL	Hardware Description Language
HEM	HMC extension module
HMC	Hybrid Memory Cube
IO	Input/output
PFB	Polyphase Filter Bank, used to channelise the data in an FX correlator
FFT	Fast Fourier Transform
MTU	Maximum Transmission Unit in a packet switched network
SFP+	Small Form factor Pluggable interface
QSFP	Quad Small Form Factor Pluggable Interface
SKA	Square Kilometer Array
MSPS	Million Samples Per Second
GSPS	Giga Samples Per Second
UDP	User Datagram Protocol (packet layer used for data transport via Ethernet)
UNB	UniBoard
VHDL	Very high-speed integrated circuit Hardware Description Language

1.2 References

1. http://www.radionet-eu.org/uniboard2 - UniBoard^2 Wiki
2. "Some Considerations Regarding The Design Of The UniBoard Correlator", Szomoru, A., JUC Memo #8 5 September 2012
3. Personal Communication, Keimpema, A., JIVE, 10 June 2015
4. "UniBoard^2 Design Document: EVN Correlator and SKA Low Frequency Channeliser" UniBoard^2 Project Deliverable 8.3, Hargreaves, J., 29 January 2014
5. "Correlator Timing and Synchronization", Verkouter, H., JUC memo #11, 11 October 2012
6. "VLBI Data Interchange Format (VDIF) Specification", Release 1.0, Ratified 26 June 2009, Madrid, Spain. Available on the UniBoard Wiki
7. "The UDP/IPv4 FPGA/UniBoard Command Protocol", Verkouter, H., 2 October 2012
8. "JIVE Uniboard Correlator Review Memo: The Delay Model", Small, D., 25 January 2013

1.3 Table of Contents

2 Introduction

2.1 Status of UniBoard^2 Hardware

UniBoard² (grant number 283393) started on July 1st 2012 with the intention to create an FPGA-based, generic, scalable, high-performance computing platform for radio-astronomical applications [1]. In May 2015 the first prototype board was received at Astron for testing and de-bugging. The first board was manufactured using engineering sample (ES) versions of Altera's 20nm Arria 10 FPGAs, part number 10AX115U4F45I3SGES.

Meanwhile design of an extension board started in early 2015, with the first prototype expected in the third quarter. The extension board (dubbed Hybrid memory Extension Module, or HEM) is intended as a proof of concept to develop applications using a new memory technology, Hybrid Memory Cube (HEM).

The chosen architectures of UniBoard^2 and the HEM are shown in Figures 2.1 and 2.2, with blue lines representing high-speed (>10Gbps) transceiver paths.

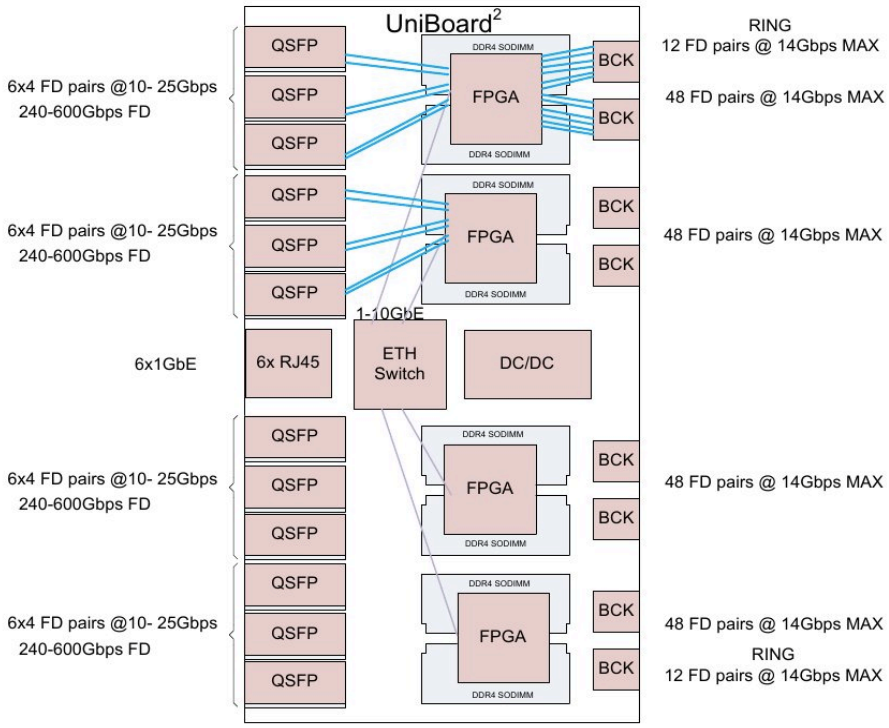

Figure 2.1: *Architecture of UniBoard^2*

UniBoard^2 comprises a single column of four Altera Arria 10 GX1150 FPGAs connected to QSFP transceiver cages on the front side of the board (to the left in the Figure) and to backplane connectors at the back. Of the 96 transceivers on each FPGA, 24 are wired to the QSFP cages and 48 to the backplane connectors. The remaining 24 transceivers connect each FPGA to its two neighbours in a ring configuration (not shown in the Figure). Two 72-bit wide DDR4 sockets are placed underneath each FPGA to provide local off-chip storage.

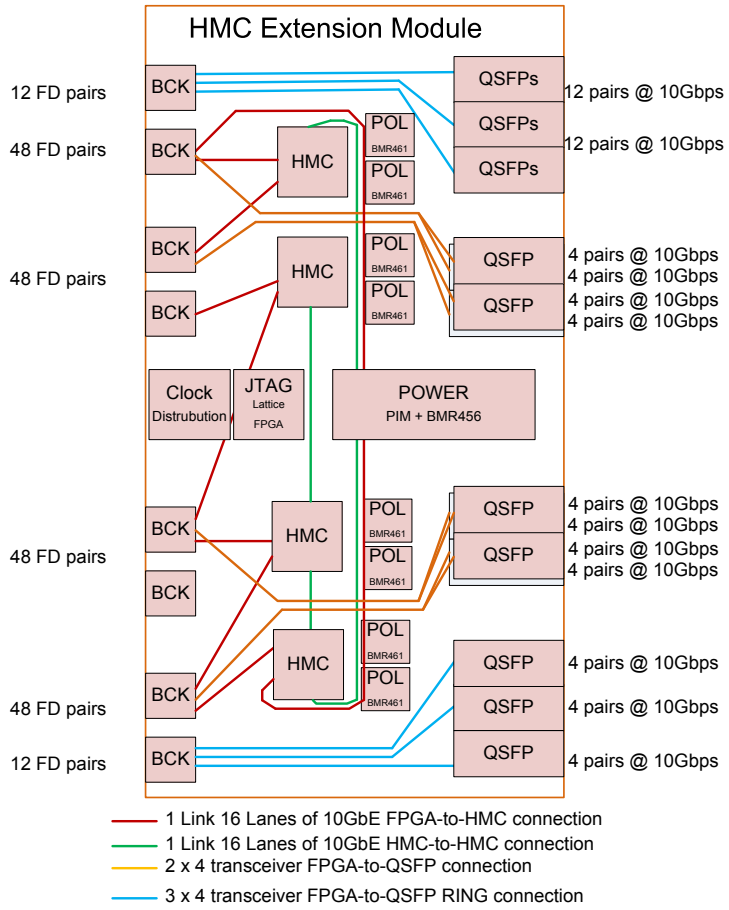

Figure 2.2: *Architecture of HMC Extension Module*

The HEM board is designed to plug directly into the UniBoard^2 via the backplane connectors labelled 'BCK' in the diagrams. Initially the HEM comprises four 4GB Hybrid Memory Cube (HMC) modules. The 'links' between HMCs, and between the FPGAs and HMCs are bundles of sixteen 10Gbps transceiver lanes. An HMC module has four of these 160Gbps links and a cross-point switch to allow the devices to be daisy chained. Each FPGA is linked directly to two HMC modules, and the HMCs themselves are connected in a duplex ring mesh (green). In this way all four links per HMC are used. Each FPGA gains direct access to two of the HMCs, and indirect access to the other two in one hop.

The blue lines in Figure 2.2 show the connections to the top and bottom of the UniBoard^2 ring mesh. Short fibre-optic cables can be connected directly between the top and bottom QSFP cages to complete the ring in a single board configuration. Alternatively the ring can be extended to daisy chain several UniBoard^{2}s.

The table below gives a comparison between the Stratix 4 GX230 FPGA used in UniBoard 1 and the FPGAs chosen for the initial Arria 10, and future Stratix 10 versions of UniBoard^2. Note that UniBoard 1 has 8 FPGAs arranged in two columns and connected by a transceiver mesh, compared to the single column of four FPGAs on UniBoard^2.

Vendor	Altera	Altera	Altera
Device	Stratix 4 GX230 (UniBoard 1)	Arria 10AX115 (UniBoard^2)	Arria 10SG280 (UniBoard^2)
Logic Cells	228,000	1,150,000	2,753,000
Flip Flops	182,400	1,710,800	3,732,480
Multipliers	1288 18x18 bit	3036 18x19 bit	11520 18x19 bit
On-chip RAM Block (Mbits)	14	66	229
On-chip RAM Distributed (Mbits)	2.8	13	15
Transceivers	24 @ 2.5Gbps	96 @ >10Gbps	96 @ >10Gbps

The logic, DSP and memory resources per FPGA increase by factors of 5, 2.4 and 3.5 respectively between Stratix 4 and Arria 10. Overall a performance increase of around 4x is expected due to better utilization of the available resources. Several factors contribute to better resource utilization in Arria 10:

- More interface logic, such as the DDR controllers, is implemented in hard IP and does not consume programmable logic resources
- There are more routing resources available per unit of logic than in Stratix IV. In Stratix IV it was found that routing rather was more often a bottleneck than resources at a system clock of 260MHz.
- Arria 10 has twice as many registers per logic block than Stratix IV. This allows more pipelining whereby a complex logic function is divided into several smaller, faster stages. Registers are used to re-clock the signals between the stages.

Migration to Stratix 10

The chosen Arria GX1150 FPGA can be upgraded to a pin compatible device from the next generation 14nm Stratix 10 family, the 10SG280. It can be seen from the table that the Stratix 10 device offers another two and a half to four times increase in DSP and logic resources over Arria 10. In addition Altera claims that the 'HyperFlex Architecture' allows a doubling of clock speeds by providing a large number of pipelining registers distributed along the routing paths.

The selected variation of the Arria GX1150 has the maximum available 96 transceivers to avoid the Stratix 10 version being IO limited. All the Arria 10 transceivers will operate at least at 10Gbps, while some of the Stratix 10 transceivers will be able to run at 28Gbps or higher. The PCB transceiver paths and backplane connectors have been specified to work at the higher rate. The power supplies have been designed to accommodate the higher loads needed for Stratix 10 devices.

2.2 Motivation for a Low Spectral Resolution EVN Correlator

The existing UniBoard1 based EVN correlator design was specified with a channelization of 1024 frequency bins per 16MHz of bandwidth (15.625kHz spectral resolution) [2]. This is effectively the native resolution of the correlator, and cannot be changed at run-time.

However not all VLBI processing requires such a high spectral resolution. For continuum (as opposed to spectral line) processing using the production software correlator at JIVE a typical setting is 32 points per 16MHz of input bandwidth [3]. This is the origin of the proposed 500kHz spectral resolution for the design outlined in section 5 of this document.

The volume of output data increases in proportion to the spectral resolution. Since each frequency bin must be transported across a network and stored, the chance of data loss and the time required for post-processing also increases. Indeed the first stage of processing is often to reduce the raw output data by averaging eight or sixteen consecutive frequency bins together.

In May 2015 this feature was implemented in the UniBoard1 firmware by allowing the accumulator process to run on for 2, 4, 8, 16 or 32 bins before exporting the products. This resulted in a proportionate reduction in the volume of output data. In this document the idea is taken to the next stage: a version of the correlator firmware with a lower native spectral resolution.

2.3 Design Consequences of Reducing Spectral Resolution

In section 2.2 it was stated that the initial UniBoard1 EVN correlator firmware has a higher native spectral resolution than necessary for much of the continuum processing done at JIVE. It will be seen that reducing the spectral resolution has design and operational consequences throughout the signal processing chain.

2.3.1 Corner turning after the filterbank

In an FX correlator the data are channelized to the required spectral resolution before cross-multiplication. The correlation products are calculated by complex multiplication per- spectral channel and per- baseline. The products are then accumulated over the integration period. A simplified view of this process is shown in Figure 2.3.

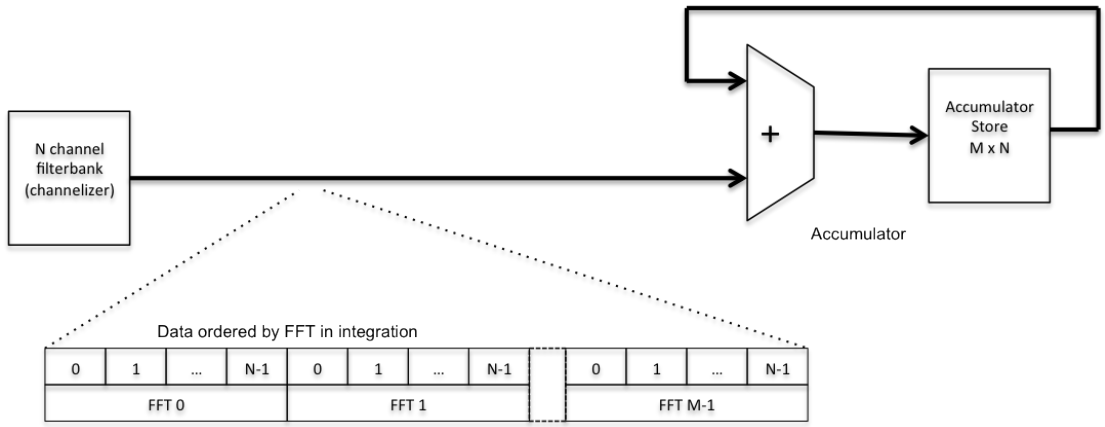

Figure 2.3: *Filterbank output fed directly to the correlator engine*

The figure does not show the cross-multipliers or multiplexer logic needed to route the incoming data to them. The key feature in this 'on-the-fly' architecture is the storage needed, alongside the accumulator, to store the intermediate accumulation results until the last FFT period of data has arrived. In an FPGA correlator, this store is best implemented in on-chip static RAM because low and predictable latency simplifies the read-add-write operation inherent to accumulation. The size of this memory requirement scales with both the number of correlation products and the number of spectral channels. The table below shows that for the UniBoard1 EVN correlator, the storage requirement was more than 10x bigger than the available RAM.

In a practical implementation there would be a bank of accumulators working in parallel, but the same amount of on-chip RAM, or its equivalent in registers, would be required.

Parameter	Value	Comments
Number of products P	2112	32 stations, 2 polarizations
Accumulator width (bits) W	72	32 bit complex
Number of spectral channels N	1024	
Storage required for accumulator	155.7 Mbits	PxWxN
Storage available on a UniBoard1 FPGA	14 Mbits	

In the UniBoard1 design the solution was to corner-turn the channelized data between the filterbank output and the correlator as shown in Figure 2.4

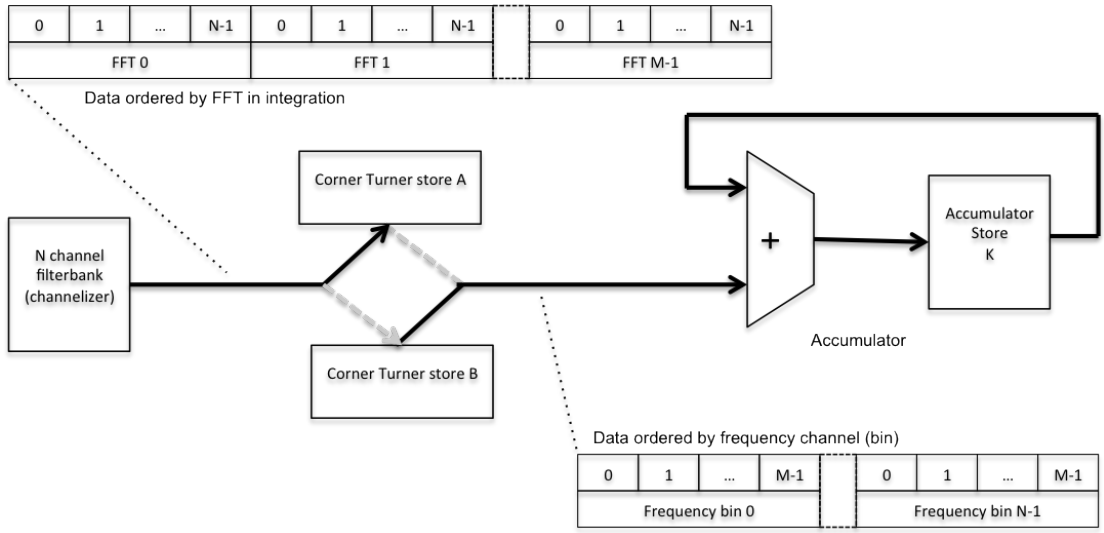

Figure 2.4: *Corner turner between the filterbank and the correlator engine*

The effect of the corner-turner stage is to move the storage problem from scarce on-chip static RAM into relatively abundant DDR3 memory. The products are accumulated one frequency channel at a time, and then exported before moving to the next frequency channel. In the real implementation there are 132 multiply-accumulate cells calculating 16 products each. Some on-chip storage is required for these 16 intermediate values, but the amount is far less than the on-the-fly configuration.

Reducing the spectral resolution, in combination with the larger on-chip static RAM available on the UniBoard^2 FPGAs, make it possible to remove the corner turner from the design, and adopt an 'on-the-fly' accumulator architecture. This offers a number of design advantages:

Integration Time

In a corner turned architecture, the volume of corner turner storage required is proportional to the maximum desired integration time. In an on-the-fly design the accumulator, associated registers and on-chip ram need only grow by one bit to double the integration time. This is about 3% bit growth on a typical 32-bit accumulator. Integration times greater than 1 second can be supported readily in a design without a corner turner.

Cost, Power and Availability of Corner Turner Storage

The table below shows that 32GB of corner turner storage is needed per second of integration for a UniBoard^2 correlator processing 128MHz of bandwidth per board.

Parameter	Value	Comments
Spectral points per 1s integration S	$128x10^6$	2x BW per correlator node
Bytes per sample (bits) W	2	8 bit complex
Number of input streams N	64	32 station, dual polarization
Storage to corner-turn one integration	16GB	SxWxN
Storage to maintain data flow	32GB	2xSxWxN

Hybrid Memory Cube is a new technology: as of Q2 2015, 2GB and 4GB modules are available. 8GB modules are planned but not yet available, and the initial prototype HEM

board will only be populated with 16GB. In addition the IP to place an HMC controller in the FPGA design will be costly, and open source alternatives are not yet available for Altera. A corner turner could be implemented using the DDR modules on UniBoard^2: two nodes populated with two 8GB DDR4 modules would support one second integrations, but DDR memory seems unlikely to scale with the processing power available in the Stratix 10 version of UniBoard^2. Each 8GB DDR4 module costs approximately EUR140 and adds on the order of 1W to the power consumption of UniBoard^2.

Data Bit-width

In the UniBoard1 EVN Correlator 9 bit complex data from the filterbank was truncated further to 8 bits to simplify packing the data into the corner turner store implemented using 64-bit wide DDR3 modules. UniBoard^2 has 72-bit wide DDR modules, and can support either 8 or 9 bit data with relatively simple addressing logic. However an on-the-fly design permits complete flexibility in the choice of filterbank output bit-width.

Validity

Since the data no longer need to be corner-turned, neither do the validity bits. This saves some on-chip storage and complexity in the correlator node.

2.3.2 Impact on the Filterbank

A high spectral resolution filterbank requires a significant amount of on chip RAM, primarily to implement the taps for the polyphase windowing function. Relaxing the spectral resolution requirement reduces the resources needed to channelize each input stream with a given window length. It becomes possible to implement longer windowing functions to improve the band shape and out-of-band noise rejection.

One advantage of a corner turned architecture is that it allows an efficient realization of the filterbank. Since the filterbank outputs are not needed until the corner turner memory bank is full, it does not matter what order the data are written into the memory. Each filterbank can be re-used, or time-division-multiplexed.

All the data from the first source for a given integration is passed through the filterbank at a multiple of real-time and stored in the corner turner memory. Then the filterbank is switched to the next source and so on until a complete set data from all the sources is stored. This scheme, which relies on the fact that FPGA logic can be clocked at a multiple of the data sample rate, is illustrated in Figure 2.5a. Data from four sources (S0-S3) are initially stored in the buffer at left, before passing sequentially through a single filterbank (channeliser) clocked at four times the data sample rate.

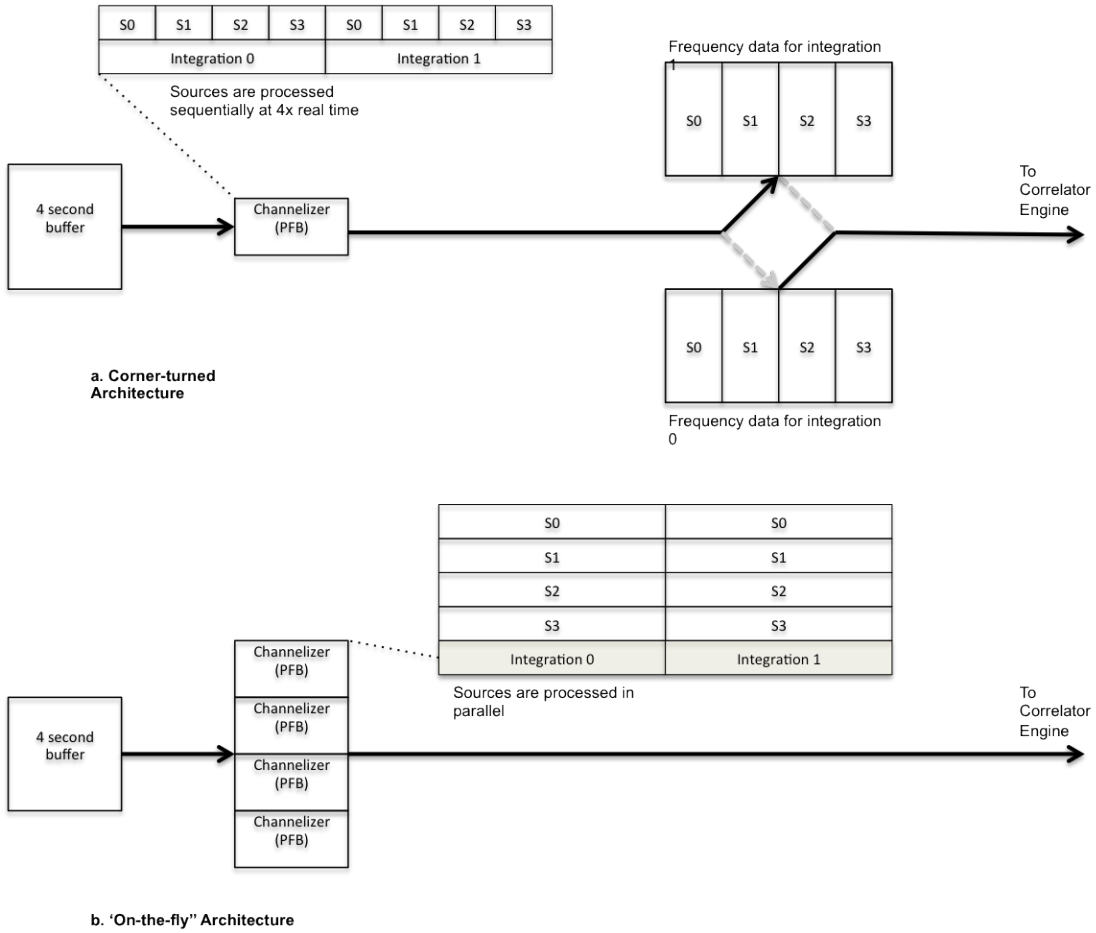

Figure 2.5: *Structure of channeliser for (a) Corner-turned, (b) On-the fly architectures*

On-the-fly accumulation requires data from all sources to be available simultaneously: the channeliser must be implemented in parallel as illustrated in Figure 2.5b. This implies that the delay corrections must also be calculated and applied to all sources in parallel, as must the validity bits.

The implications of removing the corner turner must be considered for the 'F' part as well as the 'X' part of the design.

3 EVN Correlator Design for Low Spectral Resolution Continuum Processing

3.1 UniBoard^2 Arria10 Version

3.1.1 Overview

Proposed specifications for the UniBoard^2 EVN correlator are the following[1]. Note that compared to the specification in [4] the spectral resolution has been changed from 15.625kHz to 500kHz and the maximum integration time has been increased to 8s.

Parameter	Value
Stations	32
Polarizations	2
Bandwidth (real time)[2]	128 MHz per UniBoard^2
Sub-bands (real time)	1, 2, 4, 8, 16, **32**, 64, 128MHz
Input resolution (max)	1, 2, 4, 8 bits
Bitwidth after filter-bank	9 bits complex
Integration time	0.022s – 8s
Correlation products	2112
Frequency resolution	500kHz
Data Input Format	VDIF (arbitrary length frames)

Data Input

The data path from antenna to correlator for real time data is shown in Figure 3.1. A data sender (such as a UniBoard configured as a digital receiver) at each station divides the sampled continuum signal into sub-bands. The sub-bands are packetized and transmitted across the network to the correlator.

The data senders must allocate destination IP addresses such that all the data for a given chunk of bandwidth arrives at a single correlator UniBoard. Each UniBoard^2 can process sub-bands totalling 128MHz, with 1 to 8 bit resolution. Initially the firmware will be configured for four bands of 32MHz, with 64MHz and 128MHz supported later.

Up to 32 stations can be processed simultaneously. If fewer stations are needed it is possible to trade off stations for bandwidth, for example 16 stations and 256MHz per UniBoard.

Alternatively pre-recorded data can be played back locally from disk packs, synchronised by the control system. Again the data sub-bands are distributed to the UniBoards over the 10GbE network.

[1] Note on units

Network transfer rates are base 10: 10Gbps = 10^{10} bits per second.

VLBI sample rates are binary in megasamples per second 1GSps = 1024 x 10^6 samples per second.

Memory sizes are binary: 1GByte = 2^{30} bytes

[2] Bandwidth and sub-band width are arbitrary for prerecorded data

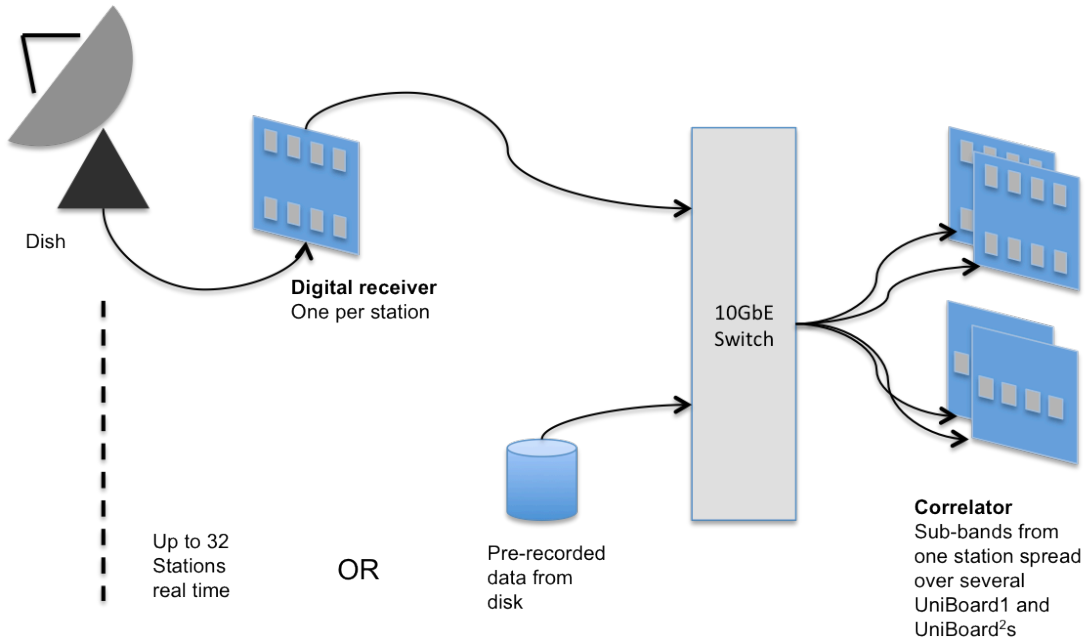

Figure 3.1: *UniBoards at the Stations Send Data to the Correlator Over a 10Gb Network*

Two of the four FPGAs on UniBoard^2 are designated 'Station Nodes' (F) and two as 'Correlation Nodes' (X). The station data are distributed between the station nodes as shown in Figure 3.2. The table below shows the data rate into each station node for different bit resolutions

BW (MHz)	Nyquist	Stations	Pols	Resolution	Data Rate (Gbps)
128	2	16	2	1	8.192
128	2	16	2	2	16.384
128	2	16	2	4	32.768
128	2	16	2	8	65.536

The 240Gbps per node input bandwidth of UniBoard^2 is sufficient to support all resolutions.

The station nodes perform all station-based processing, including compensating for network and geodetic delays, and conversion to the spectral domain using a polyphase filter bank and FFT. After the FFT the data are distributed to the correlator nodes, with each FPGA receiving half the frequency points.

Each correlator node contains the equivalent of four of the correlator engines used in UniBoard1 in order to process 64MHz of bandwidth from all 32 stations x 2 polarizations. The correlation products are exported via the QSFP cages on the front of the board.

The F and X nodes run at a system clock rate of 261MHz, or 2% higher than the real time aggregate sample rate of 256MHz. The extra 2% allows some dead time for housekeeping between integrations. The actual throughput rate is determined by the rate at which the control system feeds data into the 'F' nodes [5].

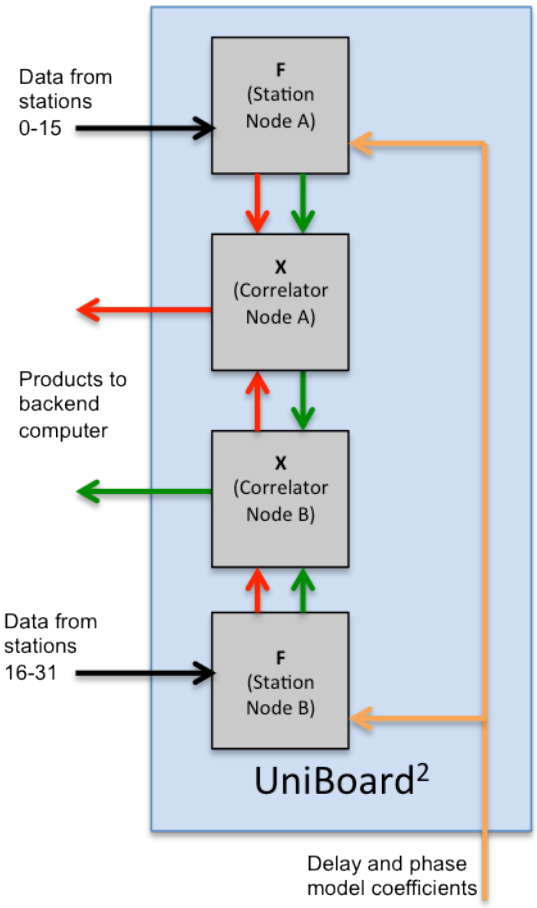

Figure 3.2: *Overview of data paths through UniBoard*^{2}

The black arrows on the left of Figure 3.2 represent data arriving from the stations. These external interfaces are implemented using the optical QSFP ports. Each node has a 1Gbps Ethernet port for control and monitoring. The control computer sends the delay and phase models to the 'F' nodes over this port as shown by the orange line in Figure 3.2.

The red and green lines in Figure 3.2 show the paths of each half of the frequency points from the station nodes to the correlator nodes, and finally as correlation products exported to the backend computer for post processing.

The internal ring mesh is used to transport the data from the 'F' to the 'X' nodes. The links between two adjacent nodes comprise 12 transceivers operating at a minimum of 10Gbps to provide 120Gbps in each direction. The correlator nodes include logic to forward data destined for the other correlator engine. For example half the data from Station node A is correlated in Correlator node A: the 'red' half of the spectrum. The other, 'green' half, must be forwarded to correlator node B to be combined with the same part of the spectrum from Station node B.

The worst-case transceiver bandwidth required to transport the output from one of the F nodes is shown in the table below.

Parameter	Value	Comments
Spectral points per 1s integration S	$128x10^6$	2x BW per correlator node
Bitwidth (bits) W	18	9 bit complex
Number of streams from F node 'A' N	32	16 station, dual polarization
Storage to corner-turn one integration	73.7Gbps	SxWxN
Available transceiver BW	120Gbps	12 x 10Gbps links

Overview of Signal Flow

Figures 3.3 and 3.4 show the signal flow through the station and correlator nodes respectively. Sections 3.1.2 and 3.1.3 discuss each block in more detail.

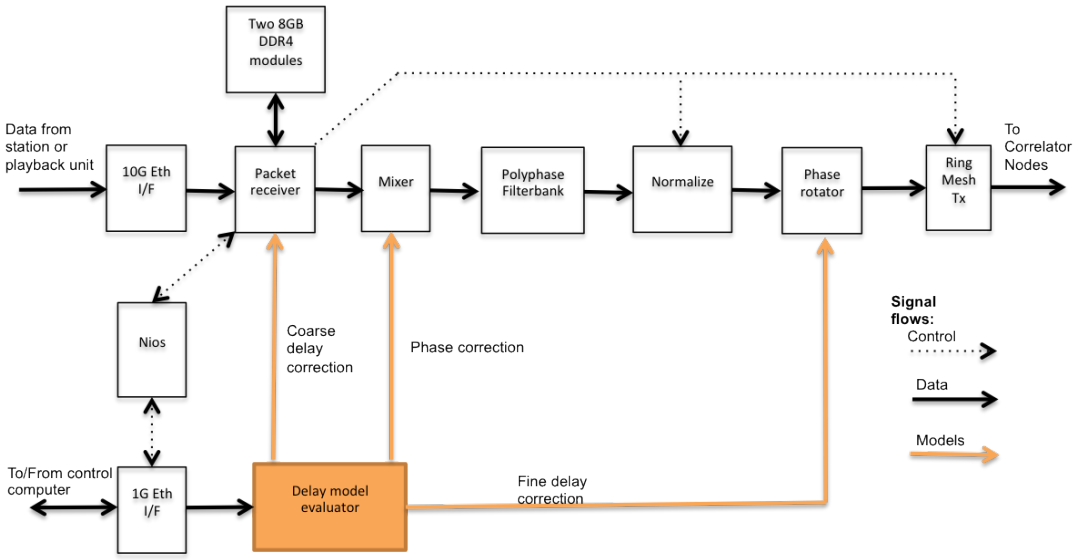

Figure 3.3: *Signal flow through a Station Node*

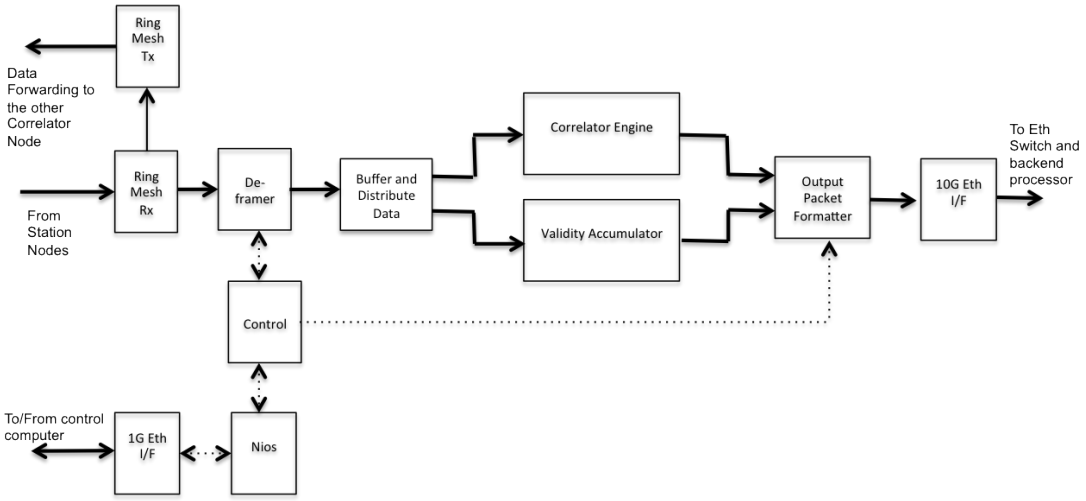

Figure 3.4: *Signal flow through a Correlator Node*

3.1.2 F nodes: Station Based Processing

Packet Reception

Data are transmitted from the stations in VDIF [6] formatted jumbo UDP packets. Any valid frame length up to 8192 bytes will be possible as long as the frame length remains constant during an experiment. VDIF frames must contain an integer number of 64 bit words. There must also be an integer number of frames per second, and the first sample in a second must also be the first sample in a VDIF frame.

When two polarizations are used they must be transmitted in separate packets.

One packet contains one VDIF frame. The VDIF station ID and thread ID fields are ignored; instead a UDP port number is hard coded into the firmware for each station and sub-band. Because the same UDP port numbers are reused in each station node, the combination of IP address and UDP port number is needed to fully identify each data stream. The UDP port to stream mapping is defined in [7].

Data inflow must start on a second boundary. The time field in the VDIF header is compared to a pre-set start time. After the start time data are stored in a 4 second deep circular buffer in a slot determined by the second, epoch and frame-within-the-second fields in the VDIF header. When data from different VDIF epochs are combined, the control system provides the required offset to convert each station to a common epoch prior to correlation.

At higher resolutions the storage requirements for the four-second buffer become substantial. Two 8GB DDR4 modules will accommodate the 1, 2 and 4-bit cases. 16GB modules or HMC will be needed to support the 8-bit case.

Synchronization of Data and Delay Model

The control system coordinates sending data and delay model information to the UniBoard, and is thus aware of the fullness of the circular buffer. Data are read from the circular buffer and correlated when the control system instructs the UniBoard to process a batch of N integration periods. The integration period is set by the control system to an integral number of FFT periods (1 FFT period = 128 samples). Real time and pre-recorded data are treated identically from the point of view of the UniBoard firmware. Further details of the synchronization mechanism are given in [5].

Upper and Lower Sidebands

The control computer marks each input data stream as an upper or lower sideband by clearing or setting a bit in a control register. Lower sidebands are converted to upper by multiplying odd samples by -1, effectively mixing the signal with the Nyquist frequency. This is done before the data are stored in the circular buffer and subsequent processing treats upper and lower sidebands the same.

Delay and Phase Correction - Introduction

The control system sends a set of delay and phase coefficients per integration period. The delay models are per station, and the phase models are per sub-band per station. The coefficients are held in a FIFO until the correlator is instructed to process the corresponding data segment. The polynomial order and coefficient resolution required for the models to remain valid over a given integration time are discussed by Small [8]. To summarize:

Delay model

Second order polynomial (delay, delay rate, and delay acceleration) with 48 bit coefficients.

$\tau = d_0 + d_1 t + d_2 t^2$

Coefficients are updated from the FIFO at the start of an integration period. The polynomial is evaluated once per FFT period by a simple accumulator. The delay is scaled such that the top 48 bits of the calculated delay represent the number of whole samples to adjust the circular buffer read pointer. The next 8 bits are fed to a look up table to translate the fractional time delay to a per-frequency bin phase rotation. This phase correction is applied to the data after the FFT in the “Phase Rotator” module shown in Figure 3.3, while the integer part is applied in the “Packet Receiver” module.

Phase model

Second order polynomial (phase, phase rate, and phase acceleration) with 64 bit coefficients.

$\phi = p_0 + p_1 t + p_2 t^2$

Coefficients are updated from the FIFO at the start of an integration period. The phase polynomial is evaluated every sample using a two stage accumulator. The top 9 bits of the phase accumulator, representing a 2pi rotation full scale, are applied to the complex mixer shown in Figure 3.3.

The following section provides more details on how the models are evaluated and applied in the UniBoard.

Delay and Phase Models - Implementation

The control computer calculates a geodetic delay model for each station and transmits it to the FN FPGAs as coefficients of a polynomial of the form

$\tau = d_0 + d_1 t + d_2 t^2$

in which t is time during the period during which the coefficients d_0, d_1 and d_2 are valid, and τ is the delay correction for that station. In this case t is simply a count of the FFT number within each integration period, since the UniBoard evaluates the delay model once per FFT. A phase correction is required because the delay correction is done at sub-band frequency, not sky frequency. This is calculated as

$\phi = p_0 + p_1 t + p_2 t^2$

Using coefficients p_0, p_1 and p_2 calculated by the control computer. In this case t is a count of the sample number within the integration period.

Coefficient Resolution, Storage and Bandwidth

Both the delay and phase model coefficients are updated per integration. Given a realistic minimum integration time of 8192 FFT periods, and a maximum of eight seconds, the coefficients need to be updated between 1 and 61 times per second. The UniBoard provides a 128 word deep FIFO so that the control computer can send coefficients a second in advance without overflowing.

The coefficient storage requirement for the delay models is

48 bits x 3 coeffs x 16 stations x 128 fifo depth = 288k bits

The coefficient storage requirement for the phase models is

64 bits x 3 coeffs x 16 stations x 4 sub-bands x 128 fifo depth = 1536kbits

The bandwidth needed to transmit half this data each second is approximately 912 kbps per station node.

Evaluation of the Models

Figure 3.5 shows the path of the delay coefficients through the FIFOs and into the evaluator. At the start of an integration period, the next pair of coefficients is read from the FIFO into the delay and delay rate registers as shown by the green arrows. Note that the delay constant is inserted at bit 20, effectively multiplying it by 2^{20} relative to the delay rate. This allows the relatively large constant delay, and much smaller delay rate to be represented by 48 bit coefficients.

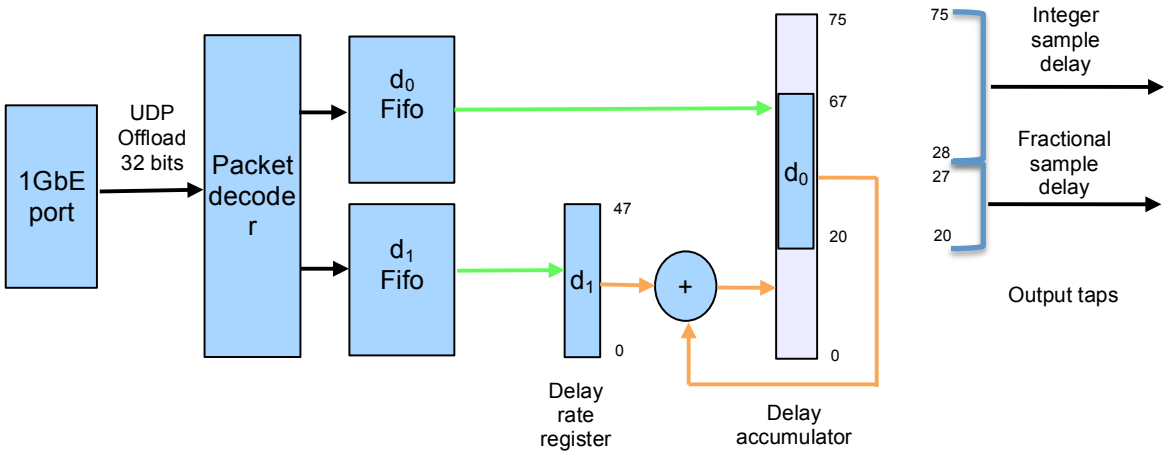

Figure 3.5: *Evaluation of the Delay Model*

The second derivative d_2 is also used, though not shown in Figure 3.5. After every FFT period the delay model is evaluated using the same algorithm used for the phase calculation shown in Figure 3.6. The new delay value is tapped off at the bit positions shown to the right of the Figure. The 48-bit integer part is sent to the logic streaming data from the circular buffer to the filter bank. When its value changes by +/-1, one sample is skipped or repeated.

The next eight bits represent the fractional delay to the nearest 1/256th of a sample. They are delayed by sixteen FFT periods to match the latency of the filter bank, and then applied as a per-frequency bin phase correction to the FFT outputs. Figure 3.7 later in this section shows where the delay and phase models are applied to the data.

The phase models are evaluated using the accumulators and registers shown in Figure 3.6. Again the green registers are loaded with fresh p_0, p_1 and p_2 coefficients from the FIFO (not shown) at the start of an integration period. The phase models are evaluated every sample while data is flowing. All the registers and accumulators are 64 bits wide and permitted to overflow since phase can wrap round. The top nine bits of the output register are tapped off and fed to the sine/cosine lookup tables in the mixer module as shown in Figure 3.7.

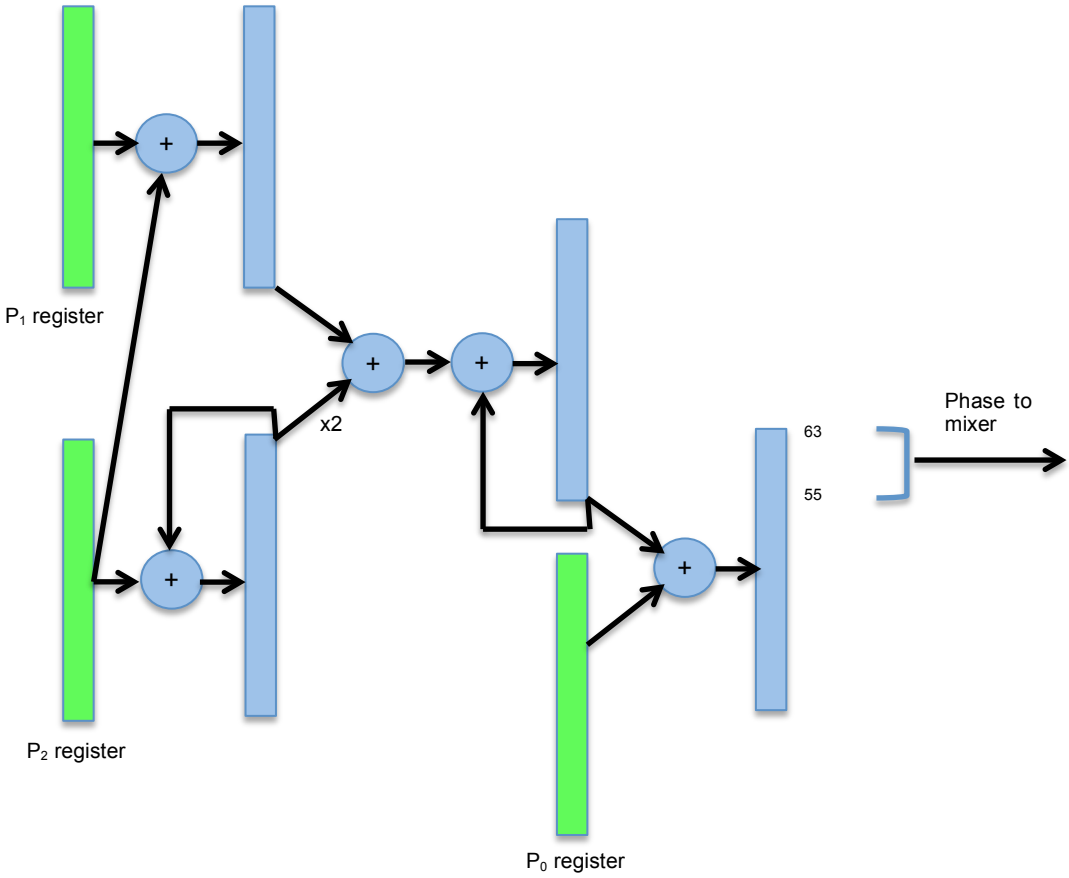

Figure 3.6: *Evaluation of the phase model*

The system of adders and accumulators evaluates the following:

$\phi = p_0 + \Sigma(p_1 + p_2 + 2\Sigma p_2)$

Iteration	Value
0	p_0
1	$p_0 + p_1 + p_2$
2	$p_0 + 2p_1 + 4p_2$
3	$p_0 + 3p_1 + 9p_2$
4	$p_0 + 4p_1 + 16p_2$
5	$p_0 + 5p_1 + 25p_2$

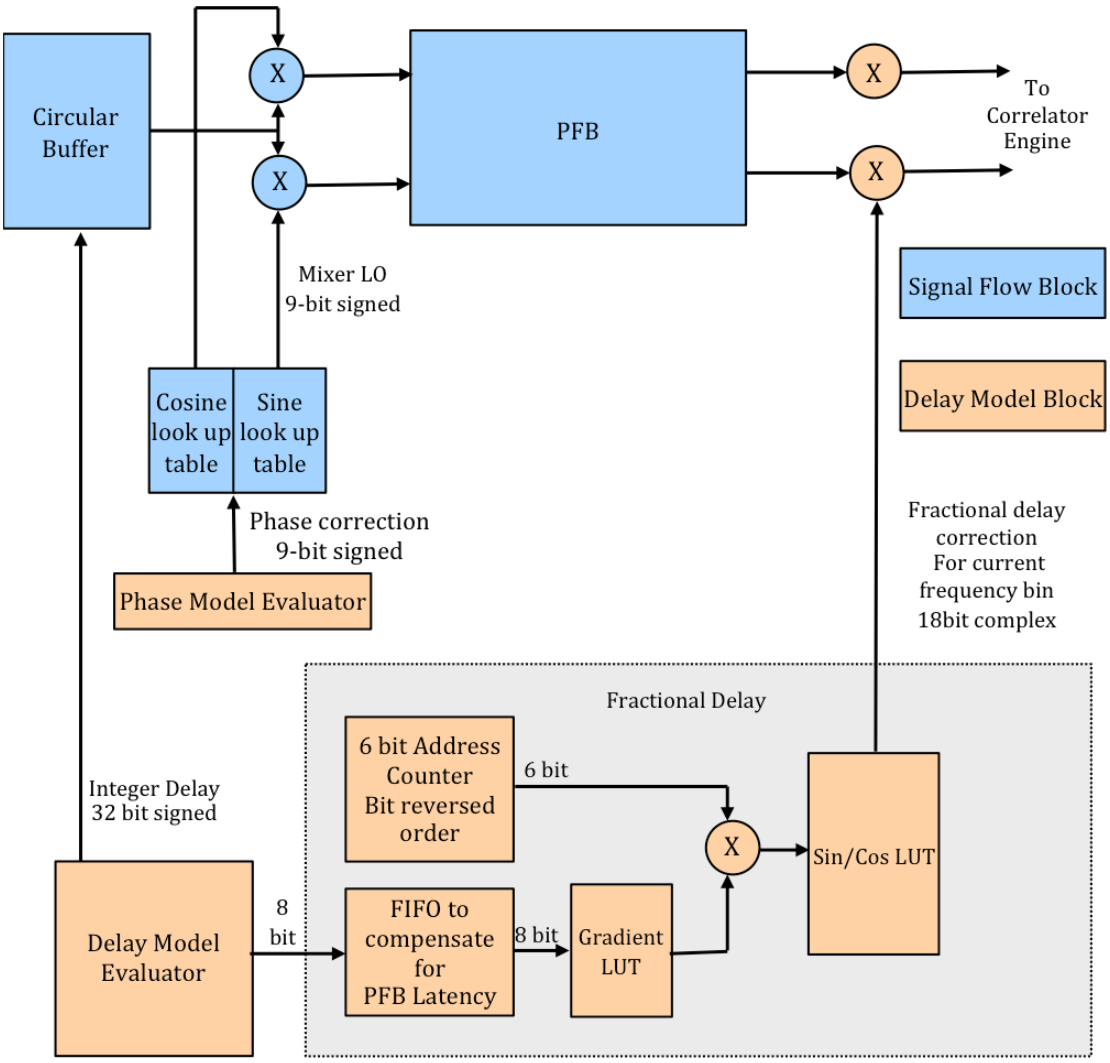

Figure 3.7: *Applying the Delay and Phase Corrections to the Data*

The fractional component of the delay is passed through a FIFO to compensate for the pipeline delay through the PFB and FFT. It is used to look up the phase slope across the band, or 'gradient' that must be applied for that fractional delay. The gradient is flat (no correction) when the fractional delay is zero, and has maximum slope when the fractional delay is 255.

The gradient is multiplied by the frequency bin number (0, 32, 16, 48 … 63) to calculate the phase correction, which in turn is translated to sine and cosine values using another look-up table. The frequency bins are counted in bit-reversed order to match the channels emerging from the FFT.

The fractional correction is applied at the centre of the band, meaning all the gradient lines pass through zero in the centre of the band. Frequency bins 0-31 have a negative correction and 32-63 a positive correction. When the integral delay changes by one sample a 90 degree correction is made to the phase model to compensate for the phase jump caused by the fractional delay rolling over.

The values in the look-up tables were generated using Matlab.

Phase and Delay Coefficient Transmission

Coefficients are transmitted to the FPGAs via the UDP offload port of the 1GbE module. The packet format is shown in the table below.

Bit 31 Bit 0	Description
Packet header (Ethernet, IP and UDP headers)	The first word of a new packet is marked by SOP='1'
0x00000002 or 0x0000000a	WRITE. Discard the rest of the packet if not 2 or 10
0x00000060	Number of data points (96 decimal)
0x4000000M	MSB=0 indicates delay. Bit 30 = 1 indicates 48 bit coefficients. M is the station number
0x0000 d0 0(47:32)	Block of delay data for station M in time order. 48 bits per coefficient, padded to 64 bits. 192 thirty-two bit words in total. For earth base VLBI with an integration time less than 1s the d2 coefficients are negligible
d0 0(31:0)	
0x0000 d1 0(47:32)	
d1 0(31:0)	
0x0000 d2 0(47:32)	
d2 0(31:0)	
…	
0x0000 d0 31(47:32)	
d31 0(31:0)	
0x0000 d1 31(47:32)	
d31 0(31:0)	
0x0000 d2 31(47:32)	
d31 0(31:0)	
0x00000002 or 0x0000000a	WRITE. Discard the rest of the packet if not 2 or 10
0x00000060	Number of data points (96 decimal)
0xc00000LM	MSB=1 indicates phase. Bit 30 = 1 indicates 64 bit coefficients. M is station & L is sub-band
p0 0(63:32)	Block of phase data for station M, subband L in time order. 64 bits per coefficient 192 thirty-two bit words in total More delay or phase data blocks can follow in the same packet
p0 0(31:0)	
p1 0(63:32)	
p1 0(31:0)	
p2 0(63:32)	
p2 0(31:0)	
…	
p0 31(63:32)	
p0 31(31:0)	
p1 31(63:32)	
p1 31(31:0)	
p2 31(63:32)	
p2 31(31:0)	
0x00000000	The last word in the packet is 0x00 and is marked by EOP='1'

Polyphase Filterbank

The filter bank is a complex 128-point (per 32MHz sub-band) polyphase design with a window length of 16 taps per point. It is implemented as a double-rate design with throughput twice the input sample rate. At the output the duplicate frequency bins are dropped leaving a 64-

point single sided spectrum whose data rate matches the input. The frequency bin size is 500kHz for a nominal 32MHz input sub-band.

Data from the circular buffer first enter a quadrature mixer where they are converted to complex, Doppler shift corrected, samples. They then pass to the polyphase filter bank, composed of a pre-filter structure and FFT. The pre filter structure applies a Blackman Harris window function, chosen to give good out-of-band rejection at the expense of a somewhat rounded frequency bin shape.

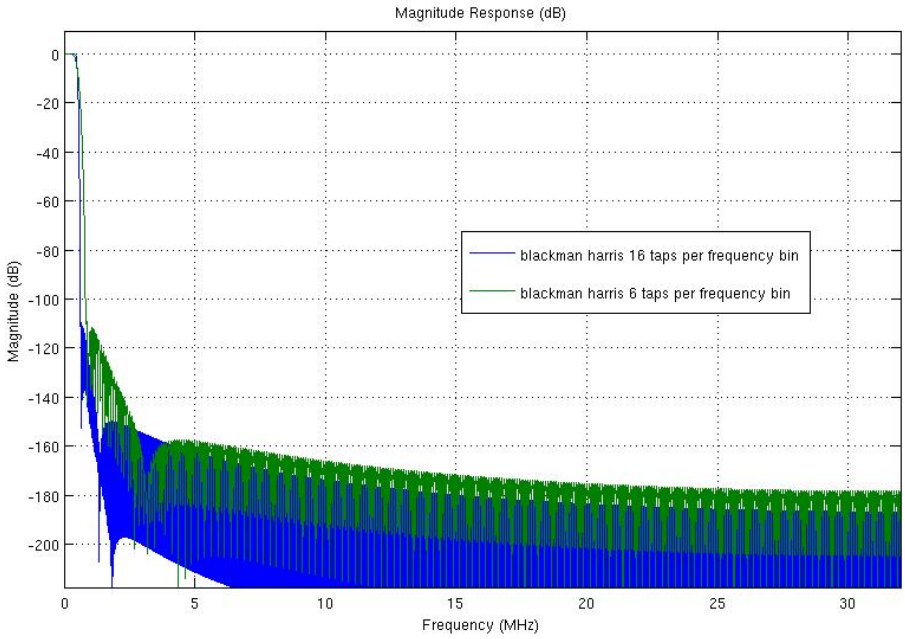

Figure 3.8: *FIR filter prototypes for 16- and 6- taps per FFT point*

The window coefficients are derived using the Matlab Filter Design and Analysis Tool, and are based on the FIR prototype shown in Figure 3.8. The figure compares six- and sixteen-taps per FFT bin (768 vs. 2048 taps). The longer window has a squarer band with steeper edge, and the about 6dB better noise rejection. A six-tap window was used for the UniBoard1 correlator due to resource limitations. Given the increased on-chip RAM available in UniBoard^2, and the reduced spectral resolution, a sixteen-tap window was implemented in this design.

The coefficients are stored in RAM on the FPGAs, and the user can load alternative windowing functions at run time by reloading the filter coefficients through the control system.

Figure 3.9 shows a Matlab simulation of the filterbank response. The signal strength in bin 40 was recorded as a sinusoidal input signal was swept from bins 30 to 49.9 in frequency steps of a tenth of a bin. The simulation used the same 16 bit fixed point coefficients as the hardware, but a floating point FFT.

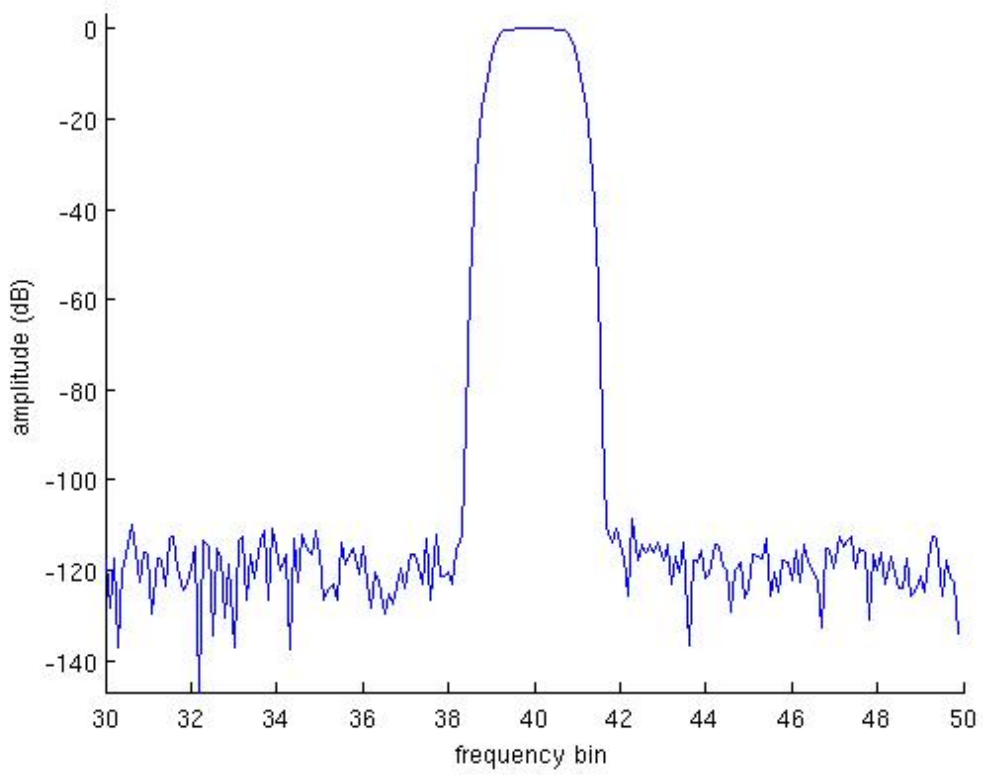

Figure 3.9: *Simulation of bin 40 response to a swept sinusoid*

Architecture of Polyphase Filterbank

The architecture of the sixteen-tap pre filter structure and FFT is shown in Figure 3.10. Data enter a shift register with taps every 128 samples. The newest sample enters from the left while the next tap selects the sample 128 clocks earlier and so on. The taps are fed to multipliers where the data are weighted with the selected values from the coefficient memory. On each clock the data shift one tap to the right, while the coefficient selector moves to the next row. After 128 clocks the coefficient selector returns to row 0 at the same time as the first sample emerges at the second tap.

The outputs from the sixteen multipliers are added together and fed into the FFT module. Note that the data entering the pre filter structure are complex, so the weights are applied to both the real and imaginary parts.

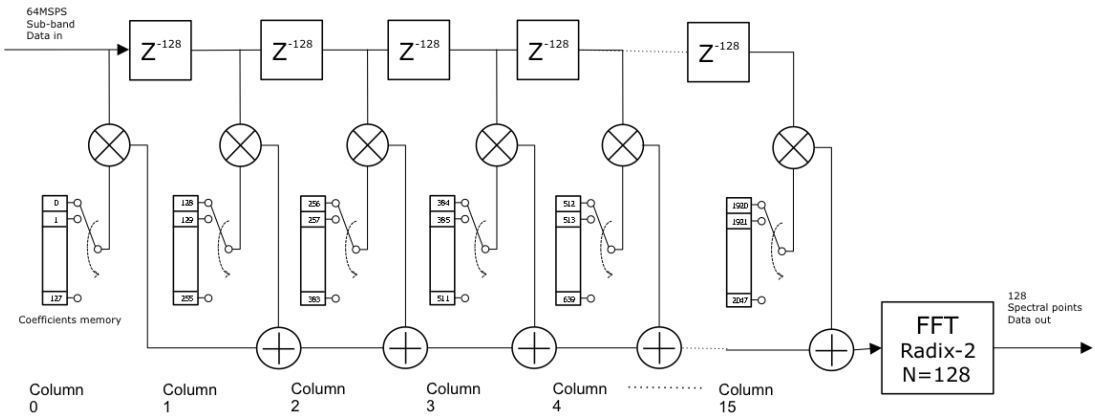

Figure 3.10: *Architecture of the pre filter structure*

As previously mentioned, the filterbank processes all sixteen dual-polarization stations simultaneously. The signal flow for a single station is shown in Figure 3.11. In the current configuration the aggregate 128MHz bandwidth is comprised of four sub-bands A, B, C & D. The sub-band samples are time multiplexed as shown in the Figure. This improves efficiency when the same operation must be performed on all four sub-bands: for example application of the fractional delay after the FFT, and of the windowing function in the polyphase filter.

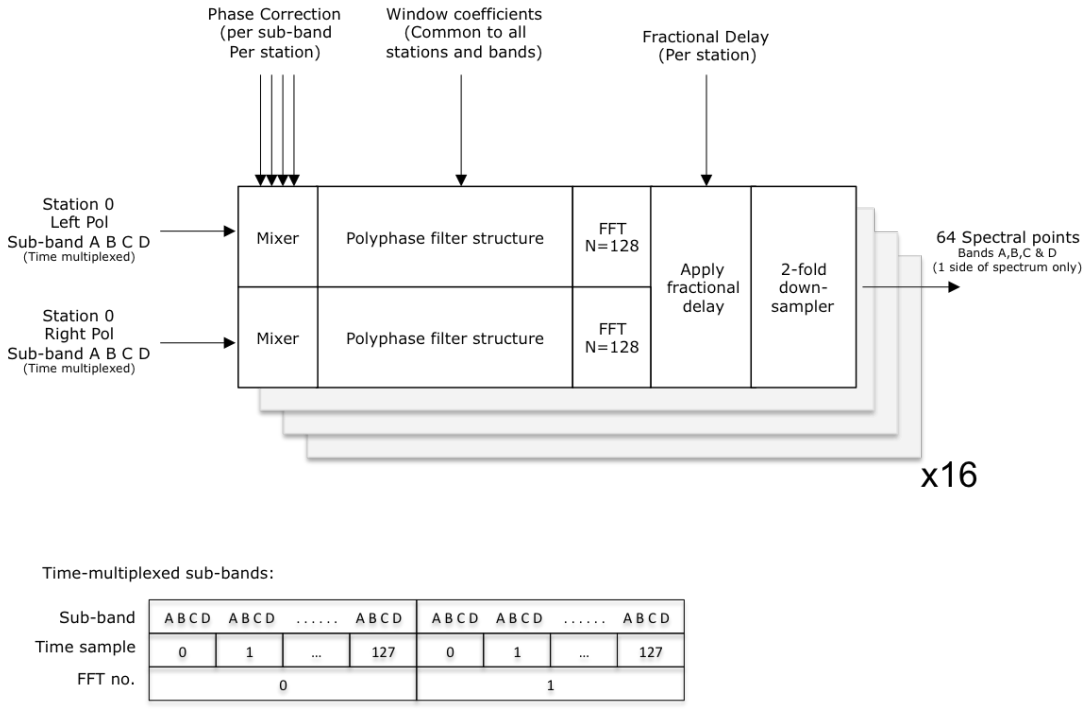

Figure 3.11: *Filter banks for 4 sub-bands, 4 stations, 2 polarizations*

After the FFT the duplicate spectral points are discarded and the sub-sample part of the delay correction is applied. The outputs are truncated to 9 bits complex. The tap point for the truncation can be adjusted via the control system.

F Node Output Framer

The output module aggregates the A & B filter-bank outputs from all the stations and generates one frame per FFT period for dispatch to the first X node. Similarly the other half of the spectrum, the C & D streams, is framed and dispatched to the second X node. The frame format is shown in the table below.

Field	Width (bits)	Comment
Real Time	64	In practise use the integration count since start of scan
FFT in integration	32	
Validity	8	bit 0: A band validity, bit 1: B band validity
Source	8	
Unused	16	
Freq bin 0 S0, P0, A, real	9	End of frequency bin 0, station 0
Freq bin 0 S0, P0, A, imag	9	
Freq bin 0 S0, P0, B, real	9	
Freq bin 0 S0, P0, B, imag	9	
Freq bin 0 S0, P1, A, real	9	
Freq bin 0 S0, P1, A, imag	9	
Freq bin 0 S0, P1, B, real	9	
Freq bin 0 S0, P1, B, imag	9	End of frequency bin 0, station 0
Freq bin 0 S1, P0, A, real	9	Start of frequency bin 0, station 1
Freq bin 0 S1, P0, A, imag	9	
…		
Freq bin 0 S15, P1, B, imag	9	End of frequency bin 0, station 15
Freq bin 32 S0, P0, A, real	9	Start of frequency bin 32, station 0
…		
Freq bin 63 S15, P1, B, imag	9	End of frame

The frequency channels are packed into the frame in the bit-reversed order they emerge from the FFT: 0 32 16 48 … 15 47 31 63. For each frequency channel there is a block containing the data for two bands, two polarizations and 16 stations. The frame length is 73728 bits of data plus a 128-bit header.

3.1.3 X nodes: Correlator Engine

De-framing and Buffering

The de-framer extracts the header information from each frame from the 'F' nodes. The time stamp and integration number are passed through to the output stage to fill the corresponding header fields in the exported packets. The FFT in integration is used to determine if it's a new integration (when zero) or the last FFT of the old one.

The channelized data are split four ways into separate buffers as shown in Figure 3.12. The buffers hold two frames in a ping-pong configuration to allow continuous data flow. Each buffer feeds one of the four 132-cell correlator engines described in the next section.

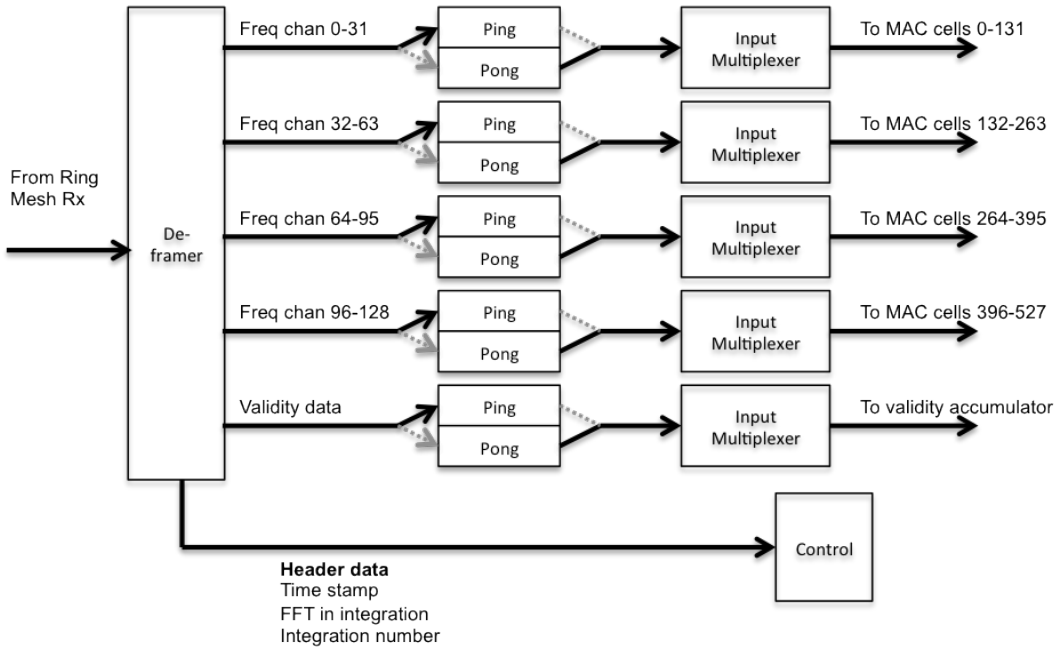

Figure 3.12: *X node de-framer and input buffer*

Each frame header contains one validity bit per source to cover the entire FFT. The validity bits are buffered and then distributed to a validity accumulator in parallel with the data.

Correlation Engines

The 2112 correlation products are computed by 528 complex multiply-accumulate cells which each calculate sixteen products sequentially. The throughput matches the input rate at a nominal 256MHz clock, but as in the station nodes the clock is run 2% faster to allow for dead time between integrations.

The 528 mac cells are arranged in four identical groups of 132, each of which processes 32 of the 128 frequency bins allocated to that node.

In Figure 3.13 below the black dots represent the cross correlation MAC cells and the red dots the (identical) auto correlation cells. The numbers below and to the right represent the 32 stations presented to the MAC cells during the first four passes. During these four passes all pol 0 x pol 0 products are calculated. A further 12 passes then compute pol 0 x pol 1, pol 1 x pol 0 and pol 1 x pol 1 for all 32 stations.

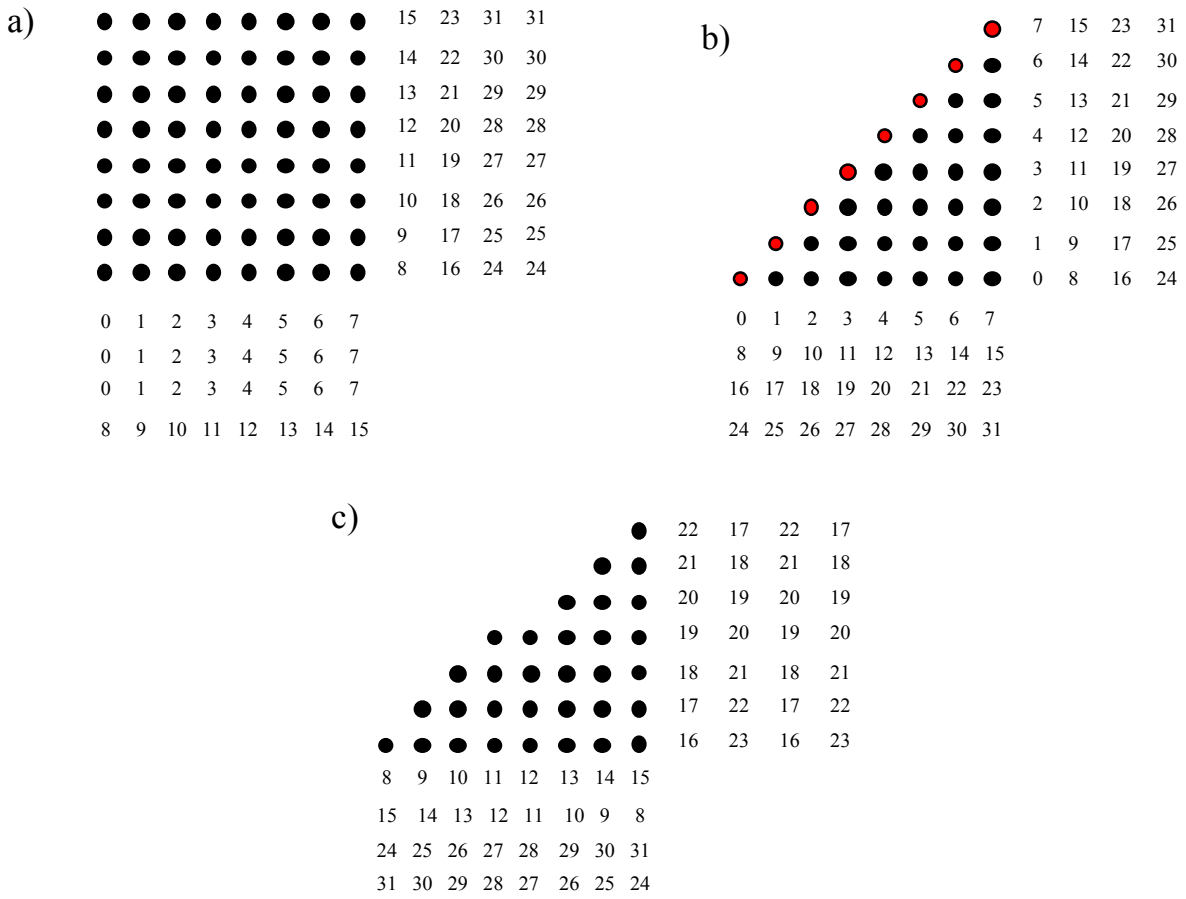

Figure 3.13: *Correlation Points Processed by 132 MAC Cells*

The cycle of 16 passes is repeated 32 times during each FFT period, until all the frequency bins have been processed. The complete sequence is illustrated in Figure 3.14.

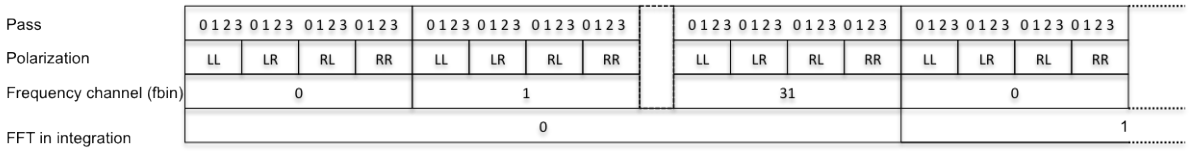

Figure 3.14: *Correlator engine processing sequence*

Figure 3.15 illustrates the architecture of a single multiply-accumulate cell. The two complex input signals a+jb and c+jd are fed in from the left. For autocorrelations a=c and b=d. The accumulator memory is an 82-bit wide, 512 word deep dual port RAM. On every clock one of the 512 intermediate results is read out and fed to the accumulator adders with the correct pipeline delay to be combined with the next input value for that pair of stations.

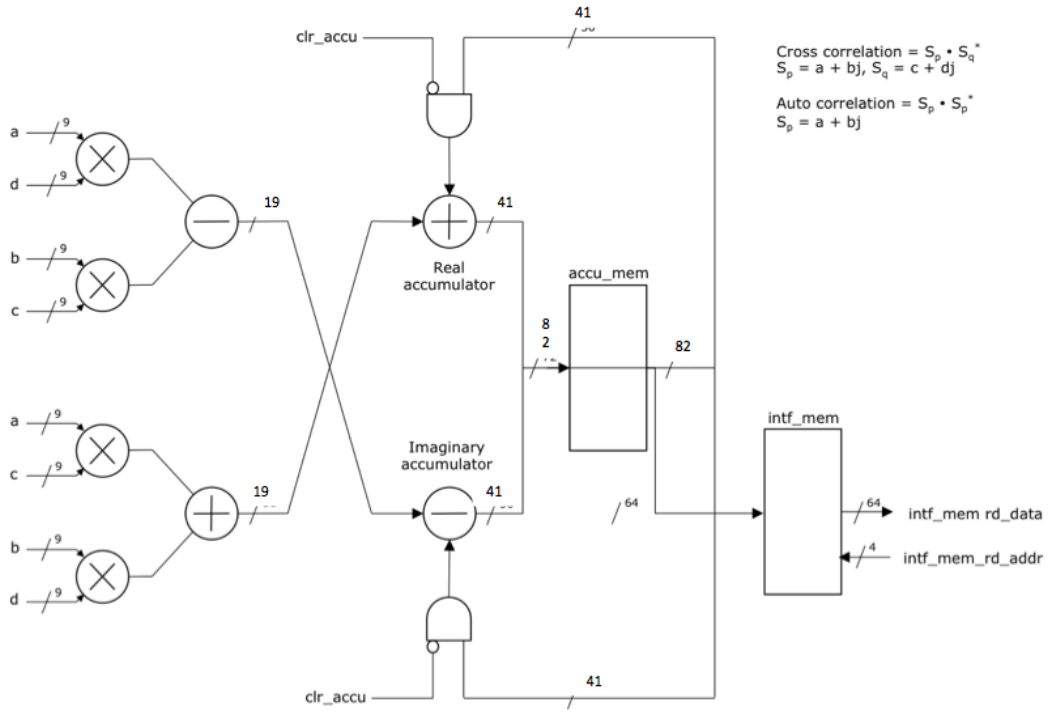

Figure 3.15: *Correlator Multiply-Accumulate Cell*

Two Arria 10 DSP blocks (using four of the 3036 available multipliers) can be configured as a single 18x19 bit complex multiplier, thus there are 759 complex multipliers available on the chip. In principle the number of MAC cells could be doubled by configuring each 18x19 bit multiplier as two 9x9 bit multipliers, but in practice the bottleneck is likely to be the multiplexer logic needed to distribute data into and out of the MAC cells.

The bit growth in the accumulator is estimated in the following table.

Parameter	Value	Comments
Bit-width from filterbank W_{in}	9	
Bit-width after complex multiply W_{out}	19	2x W_{in} + 1
Number of FFTs in 8 seconds N	4000000	
Bit growth during accumulation G	22	$\ln_2(N)$
Accumulator width required	41	W_{out} + G

The total accumulator storage required for 528 MAC cells is calculated as

528 cells x 82 bits x 16 words x 32 frequency bins = 22Mbits

The cycle of sixteen passes times 32 frequency channels is repeated with data from each FFT period in the integration time. On the last FFT the accumulated products are transferred to a dual port memory, the interface memory, where they can be read out and dispatched over the ten gigabit Ethernet port. At the start of the next integration period clr_accu is held high for the first cycle to clear the accumulator RAM. The interface memory is slightly smaller than the accumulator RAM, 17.3Mbits, because only 64 bits of the accumulator data are exported.

It is not necessary to send all 2112 products over the port. The control system selects which products must be sent, each product is assigned an address range with a fixed mapping to the input channels. The output formatter logic reads the frequency bins for the selected

products via the interface shown on the right of Figure 3.15 while the next integration is being processed.

Correlator Product Output

Accumulated products and validity bits are held in the interface memory while the next integration is in progress. During this time the packetizer module reads out the selected products, packs them into packets and sends them to the 10GbE ports.

The interface memory is carried over from the UniBoard1 correlator design, but is not the only solution. It would also be possible to use the 'dead time' between integrations to export the data. The amount of data to be exported per X node per integration (not including packet headers), when all products are selected, is

2112 products x 96 bits x 128 bins = 26Mbits

That is less than 3ms over a single 10GbE port, or 0.3% of a 1s integration. The drawback to this approach is that the data are sent in bursts, which can overload network switches, and the percentage dead time becomes more significant at shorter integrations.

Output Format

The correlation products are sent to the backend computer in UDP packets. The packet format is the same as used for the UniBoard1 correlator: separate packets are still sent for each frequency bin. The packet has a four 32-bit word header to identify the data by its frequency bin, correlator engine, FPGA and UniBoard number. Two 32-bit word fields contain the 'time stamp'. In fact this is simply a count of the number of integrations in the scan, from which the back end processor can calculate the time of the first sample in the integration period.

A packet may contain any number of correlation products plus their validity counts, provided the MTU of 9000 bytes is not exceeded.

Each product occupies three 32-bit words for the real and imaginary data and validity values. For future expansion, a flag in the header denotes whether the data fields are 32-bit or 64-bit. If there are an odd number of products in a packet, one additional padding product with zero real, imaginary and validity fields, is appended at the end.

The following table shows the first 7 words for the 32-bit case. For 64-bit data the real part of the first product would occupy words 4 and 5, the imaginary part words 6 and 7, and the validity count word 8.

Word 0	VVV_3	F_1	Chip ID_8	CE_2	$reserved_6$	Frequency bin $\#_{12}$
Word 1	$reserved_8$		Payload $size_{12}$			First product $\#_{12}$
Word 2	Integer no of seconds since epoch to start of integration $period_{32}$					
Word 3	No of samples since last second to start of integration $period_{32}$					
Word 4	Data for first product $(read)_{32}$					
Word 5	Data for first product $(imaginary)_{32}$					
Word 6	Data for first product $(validity)_{32}$					

First 7 words of an output packet for 32 bit correlation products. Msb at left.

Word 0		
Bits 0-11	12	Frequency bin number (0-1023)
Bits 12-17	6	Reserved
Bits 18-19	2	Correlator engine number within an FPGA
Bits 20-27	8	Chip (node) ID. Bits 20-22 are the FPGA; bits 23-27 are the board number.
Bit 28	1	Flag to indicate 32/64 bit data representation (0/1)
Bits 29-31	3	Header version code (0-7) default 0
Word 1		
Bits 0-11	12	Number of the first product in this packet (0-2111)
Bits 12-23	12	Payload size. The number of products in this packet (not including padding when payload size is odd).
Bits 24-31	8	Reserved
Word 2		
Bits 0-31	32	Integer number of seconds at start of integration period
Word 3		
Bits 0-31	32	Sample number within second at start of integration period

Function of the header fields in the output packet

3.1.4 Validity in the F and X Nodes

Validity bits are carried through the correlator in parallel with the data. The validity of the arriving data is determined and stored per VDIF frame: the frame either arrives or it doesn't; if it arrives the VDIF valid bit indicates the validity of the entire frame. All the bits in the validity store are initially '0'. When a valid packet arrives the bit corresponding to its frame address is set '1'. When, later, that frame has been completely read out from the buffer its validity bit is reset to '0'. If a valid packet does not arrive to fill that row by the time it is read again, the validity bit remains '0'.

As data are read from the buffer and fed into the filter bank, the corresponding validity bits are sampled. The validity of the FFT output is calculated such that the whole of the FFT is marked invalid if any of the contributing data are invalid. Since a 16-tap polyphase window is used, the contributing data include the previous sixteen FFT periods, that is 128 x 16 samples. The first six FFT periods during an integration period are always invalid while the polyphase structure is filling with new data.

The output (ring mesh interface) stage in the Station Node checks the validity status of each FFT output and substitutes zeros in all frequency bins of an invalid FFT. The invalid, zeroed, frame is then sent to the Correlator Nodes and correlated in the same way as valid data, but does not contribute to the products.

The validity bits are buffered along with the data and accumulated in a parallel 'validity accumulator' simultaneously with the data correlation, except that the validity calculation only needs to be done once per FFT since one validity bit applies to the whole FFT frame. The maximum validity result is

500000 FFTs per second x 8 second max. integration = 4×10^6

This fits easily in a 32-bit accumulation register. Thus for every correlation product a corresponding validity count is generated which can be used to normalize the data in post-processing.

3.1.5 Bit Truncation in the F and X Nodes

Input

The data read out from the circular buffer are padded to 8 bits regardless of their original sampled resolution. The mixer LO signal is a 9-bit signal from a cosine lookup table whose phase input is also 9 bits. The mixer is implemented with 9-bit multipliers whose output is truncated to 14 bits before entering the polyphase filter bank.

Filter bank

In the pre-filter structure, the 14-bit data are multiplied by 18-bit coefficients and the outputs of sixteen multipliers summed to 36 bits, of which the top 18 bits are passed on to the FFT.

In the UniBoard 1 design the FFT truncates to 18 bits at every stage, and drops a bit every stage to prevent bit growth. Simulation showed that this gave identical output amplitude to the radix-4 Lofar FFT. Arria 10 DSP blocks support 27x27 bit and floating-point modes as well as 18x19 bit multiply. The higher precision modes will be used within the filterbank and FFT if simulation shows that this improves the signal to noise ratio or reduces dc bias in the output.

The fine delay phase rotations are applied to the FFT outputs using an 18 bit complex multiplier to give a 37-bit result. This is then truncated to 9 bits at the point set by the control system.

Correlator Engine

The correlator engine is built from 18x19 bit complex multipliers, but because the input is truncated to 9 bits, the result is only 19 bits wide. The products are summed in a 41 bit complex accumulator.

3.1.6 FPGA Resources

The tables below give resource estimates for the Station an Correlator nodes respectively. The estimates are based on scaling up the actual resources used by equivalent modules in the UniBoard 1 design. Logic (ALUT), memory and register totals include 20% overhead to cover other minor modules, signal taps and so on.

Station Node

Stage	**Multipliers (18x19 equivalent)**	**ALUTs**	**SRAM(kbits)**	**Registers**
Packet Receiver	0	48800	9600	80000
Nios Controller	4	2400	340	1500
1GbE Control port	0	3800	100	5500
Delay Module	16	60700	5700	65400
Complex Mixers	64	1800	36	2800
Pre Filter	1024	11520	16850	9600
FFTs	1024	57600	570	86000
Ring Mesh Interface	0	25000	480	25000
Total/Available	**2132/3036**	**211620/854400**	**33676/63000**	**275800/1710800**

Correlator Node

Stage	**Multipliers (18x19 equivalent)**	**ALUTs**	**SRAM(kbits)**	**Registers**
Ring transceiver I/F	0	25000	480	25000
Nios Controller	4	2400	340	1500
1GbE Control port	0	3800	100	5500
Deframer/buffer	0	12000	150	11200
Validity Accumulator	0	28000	160	36000
Correlator Engine	2112	38600	22168	120000
Interface RAM	0	11600	17302	500
10GbE Product out ports	0	40000	500	24000
Total/Available	**2116/3036**	**161400/854400**	**41200/63000**	**191300/1710800**

3.2 UniBoard^2 Stratix10 Version

The design outlined above is targeted at the Arria 10 version of the UniBoard^2. The Stratix 10 version is expected to offer at least a four times increase in processing power, and possibly eight times if system clock rates above 500MHz can be achieved.

For the Arria 10 based design outlined in section 3.1, each F node processed 16 stations and each X node processed 128MHz bandwidth. For Stratix 10 it is proposed that the X and F functions are combined such that each FPGA processes 32 stations and 128MHz. This would save some resources transporting the data between nodes. The whole board would process 528MHz bandwidth from 32 stations.

The table below gives an estimate of the resources required for this configuration

Stage	**Multipliers (18x19 equivalent)**	**ALUTs**	**SRAM(kbits)**	**Registers**
Packet Receiver	0	48800	9600	80000
Nios Controller	4	2400	340	1500
1GbE Control port	0	3800	100	5500
Delay Module	32	120000	12000	130000
Complex Mixers	128	3600	100	5600
Pre Filter	2048	23000	35000	20000
FFTs	2048	120000	1200	180000
Buffer before X	0	24000	300	22400
Validity Accumulator	0	56000	320	7200
Correlator Engine	4224	77200	44400	240000
Interface Ram	0	24000	35000	1000
10GbE out ports	0	80000	1000	48000
Total/Available	**8484/11520**	**582800/1866240**	**139360/229000**	**709600/3732480**

These estimates assume a system clock speed of 256MHz plus 2%. If clock rates above 528MHz become possible in Stratix 10, a further doubling of processing bandwidth to 1024MHz per board will be possible.

A potential bottleneck is likely to be the two DDR4 modules: these are unlikely to have a matching increase in either speed or capacity. The geodetic delay buffering may have to be done in Hybrid Memory Cube instead.