ALMA Memo 607

SAO Cycle 3 Development Study Closeout Report

Digital Correlator and Phased Array Architectures for Upgrading ALMA

Alain Baudry, Lindy Blackburn, Brent Carlson, Geoff Crew, Shep Doeleman,
Ray Escoffier, Lincoln Greenhill, Daniel Herrera, Jack Hickish, Rich Lacasse, Rurik Primiani,
Michael Rupen, Alejandro Saez, Jonathan Weintroub (PI) & André Young

December 22, 2017

Abstract

This Closeout Report documents the outcome of a SAO-led ALMA Development Study of a next generation combined correlator and VLBI phased array to take greater advantage of fundamental scientific capabilities, such as sensitivity, resolution and flexibility. ALMA already represents a huge advance in collecting area and frequency coverage making it the dominant instrument for high frequency radio astronomy. We have studied processing architectures that maximize bandwidth, and thus sensitivity, allow flexible ultra high resolution spectral processing, and supports other operational modes, such as VLBI. The ALMA Science Advisory Committee (ASAC) studies *Pathways to Developing ALMA* and *A Road Map for Developing ALMA* (both referenced as Bolatto et al., 2015) comprehensively describe the community view of ALMA upgrades and their key science impact.

The methodology of the Study was to examine a variety of technologies, algorithms, balancing costs and timelines against potential benefits. The scientific impact for the proposed study derives from several key new areas of enhanced capability. The Study is divided into eight technical work packages. This Outcomes Report gives a concise summary of each, and eight detailed appendices are provided. A top-level conceptual framing of the full installation, including specifications and rough equipment costing and schedule, is presented as Phase III of three suggested design phases. Phase I is this Study, now complete.

Figure 1 Study team group photo under the CHIME array taken at the Study closing meeting , NRC-Herzberg, Penticton, BC, Canada, 24 February 2017

1 Introduction

We report here on a next-generation ALMA correlator and phased array[1] that quadruples ALMA's current processed bandwidth and provides high spectral resolution, native VLBI capability, and a variety of other features.

Science case précis: Unsurpassed instantaneous bandwidth for spectral line surveys, will provide truly unbiased connections between chemistry and the planet formation process in proto-planetary disks. An order of magnitude increase in the number of spectrally surveyed star forming regions and extragalactic sources will provide entirely new approaches to chemical evolution across a range of exciting sources. Higher bandwidth greatly increases the cosmic volumes accessible via intensity mapping, allowing complete inventories and physics of galactic gas at high-redshifts and new views of galactic evolution across cosmic time. Simultaneous line detection will lead to rapid accumulation of high-z redshift surveys with excellent prospects for unlocking long-standing questions on the build up of stellar mass in galaxies. In the time domain, new evolutionary studies of transients in the millimeter and submillimeter will be possible, providing insight into mechanisms for variability in γ-ray bursts, whose output peaks in the ALMA frequency bands. Cometary studies rely on rapid cadence observations to disentangle coma versus jet emission from these rotating bodies: the key to solving the puzzle of their origin in the solar system. VLBI features will allow ALMA to nimbly form multiple beams to create Earth-sized virtual apertures to push the limits of angular resolution from the earth's surface. By enabling beamforming in Band 7 and beyond we will sharpen our direct views of black hole event horizons, addressing some of the most fundamental questions in astronomy, physics and mathematics.

Report structure: The study was broken down into the following eight work packages, each researched in depth by a subset of the study team.

- WP2.1 Scientific requirements & specifications
- WP2.2 Identify DSP F-engine platform
- WP2.3 Determine F-engine architecture given chosen DSP platform
- WP2.4 Identify corner-turn platform
- WP2.5 Identify DSP X-engine platform
- WP2.6 Determine optimal X-engine architecture
- WP2.7 Determine design of VLBI capability
- WP2.8 Staging of new correlator and phased array

The appendix to this report includes a detailed breakdown of each work package into sub-tasks, and shows the team leader and study team members assigned to each. A substantially abridged summary of the results of the work packages is provided in section 2. The appendix also has a full, unabridged report on each work package. The eight appended reports naturally consider alternative approaches in depth, and the trade offs associated with each, while the summary in main body of the report is written to present a clean set of conclusions. However in a system this complicated, the details are important, and the appendix reports these so that it is possible to understand the underpinnings of the recommendations. *We find that it is possible with current and projected technology to improve bandwidth and spectral resolution while reducing size and increasing reliability. Increasing ALMA's bandwidth is the least expensive approach to improving the sensitivity of the instrument. Reduction of power, magnified by concomitant reduction in cooling requirements, will reduce costs, and make the entire ALMA system more robust against power disruptions.*

2 Summaries of work package outcomes

Historically in radio interferometry, *XF* correlators—cross-correlation first, with Fourier transform in software on the integrated lags—have been favored because expensive wide multipliers are not needed, since the bit-width of the multiplied data in the X-stage remains at the width of the sampled data. The advent of wide multipliers in digital signal processing hardware such as field programmable gate arrays (FPGAs) has effectively reduced the penalty for bit growth in the butterflies of the Fast Fourier Transform (FFT). The well known economy of the Cooley-Tukey FFT

[1] Sometimes referred to in this report simply as a *correlator* for simplicity.

algorithm combines with the fact that correlation collapses to bin-wise cross-multiplication in the Fourier domain to yield computational savings, and the number of instantiated multipliers for a given array size and spectral resolution is substantially reduced—the FX architecture allows for great economy in correlators with high spectral resolution and relatively large numbers of antennas. The *FX* architecture is now widely employed for modern digital correlators in radio astronomy (Primiani, et al., 2016, Hickish et al., 2016), and it is chosen as a baseline assumption for this Study.

Also with wide multipliers, it is possible to process wider bit-widths, thereby improving the digital efficiency of the correlator from 88%, for a two bit machine, to 99% for 4-bits, say, a benefit which applies equally to spectral lines and continuum observations. The present ALMA correlator is limited to 3-bit data. While it computes using 2-, 3- and 4-bit arithmetic depending on mode, the only modes being offered to ALMA users are limited to 2-bit arithmetic and 88% efficiency.

All correlators require an interconnect system to allow communication between F-processing nodes, which operate on full-bandwidth data from a subset of antennas, and X-processing nodes, which operate on data from a relatively small bandwidth from all antennas. Since different ALMA baseband converters (BBCs) handling independent band fractions may be handled by entirely separate and independent correlators, we have considered the implementation of a system for 72 dual-polarization antennas, capable of dealing with 16 GHz of processed bandwidth—8 GHz BBC bandwidth, two polarizations, and one sideband—and 4-bit samples. This amounts to 256 Gb/s per dual-polarization antenna. Quadrupling this system covers the 64 GHz bandwidth of the complete proposed next generationALMA system.

Based on these requirements, a digital system processing a single BBC must be capable of handling a data rate of ~ 16 Tb/s. The feasibility of realizing such a system with different technologies is dependent on specific topology of the interconnect required—for example, how many nodes are required at the input and output of the system. This is heavily dependent on the technologies chosen for the various signal processing engines. For the purposes of this report, we assume that the number of input nodes is 72. In other words, there is one input node per antenna, with each delivering 256 Gb/s. The number of output nodes may vary largely depending on technology; for example, one feasible FPGA implementation may feature a small number of processors, each sinking 500+ Gb/s of data. Alternatively a GPU implementation may feature many small, low-power processors, each processing 200 Gb/s.

In summary, the work breakdown for the study was predicated on a set of baseline assumptions.

1. Correlator architecture will be FX.
2. Future available bandwidth will be 16 GHz per sideband per polarization, or 64 GHz total usable instantaneous bandwidth, a quadrupling of the current ALMA processed bandwidth.
3. Even larger bandwidths still can be handled by modular replicatiion
4. Samplers will remain at the antennas with digital data sent over fiber
5. Samplers will digitize 8 GHz bandwidth per baseband channel (BBC) at 4-bit resolution
6. The number of observation modes of the new digital system will be minimized.
7. A maximum number, 72, antennas will be supported over baselines extending to 300 km.

The next subsections give an abbreviated description of the results of each of eight work packages which comprise the study. A more detailed report on the each work package is appended.

2.1 Scientific requirements & specifications

Assumed requirements for the next generation ALMA correlator and phased array are presented in summary in this section (see Table 1). A more detailed set of specifications also with more commentary is in the WP2.1 section in the appendix. We wish, within the frame of the 'ALMA 2030' documents referenced earlier, to interact with ALMA Scientific Advisory Committee (ASAC) and the ALMA Development Working Group to reach consensus on consolidated requirements. We are aware that some of the requirements listed may need more scientific discussions, long-term technical developments and may not be easily translated into engineering specifications. We point out which requirements would benefit from further scientific discussions or technical studies. For example, in addition to funding questions, more discussions need to be conducted for an ALMA Extended Array on the number of additional antennas and maximum baseline. Assumed requirements were needed to set goals for all the other work packages in this Study. In other words requirements were a necessary starting assumption for our Study. We do not claim they represent the consensus of the ALMA community; they are however informed by extensive dialog with scientific and technical colleagues in the ALMA community, and could develop to represent such consensus.

Table 1: Abbreviated assumed specifications for the next generation correlator and phased array. An appendix with the full report of the relevant working group has more requirements and more commentary on each. For easier cross-referencing the parameter numbers from that document is kept in this table, despite that a number of requirements lines have been omitted.

	Parameter	**Requirement**	**Comments**
1	Frequency range	Process digitized IF from all receivers in range $\sim 30 - 950$ GHz	ALMA Bands 1–10: cf. SCI-90.00.00.00-10-00.
2	Number of antennas	72	72 antennas ($\sim 10\%$ increase in collecting area) would allow additional antennas for the ALMA Extended Array.
3	Maximum baseline	~ 300 km	SCI-90.00.00.00-220-00.
4	Instantaneous bandwidth	32 GHz/polarization	2SB Rx: 16 GHz per SB per pol.
5	BBC BW	8 GHz	BBC bandwidth of each chunk fed to the correlator after digitization.
6	Number of BBCs	2/SB and polarization	Required to cover the desired total BW in 8 GHz "chunks."
7	Input sample format (digitizer) & Correlation sample format	4-bit & 4-bit per sample	4-bits minimizes quantization losses
9	Best spectral resolution	0.01 km/s = 1 kHz $(\nu/30\,\mathrm{GHz})$	Resolve lines from cold starless core. From SCI-90.00.00.00-30-00
13	Integration and readout interval	1 msec (auto-correlations) 16 msec (cross-correlations)	SCI-90.00.00.00-240-00. Spectral res. limited for fast dump rates, which are needed also for on-the-fly mapping .
14	Polarization products	2- or 4-polarization products	2 pol. products reduces data rate SCI-90.00.00.00-310-00.
15	Spectral dynamic range	10,000:1 for weak spectral lines near strong ones 1,000:1 for weak lines atop strong continuum	Identical to SCI-90.00.00.00-70-00.
16	Number of subarrays	6	Must be completely independent—no frequency or control restrictions).
17	VLBI	VLBI output sum port for full phased array or 2 subarrays	One subarray could just be one antenna. Ref SCI-90.00.00.00-370-00
24	Correlator configuration time	< 1.5 sec	Complete configuration should be accomplished in less than 1.5 sec in all circumstances

2.2 Identify DSP F-engine platform

At the heart of the F-engine is a transformation of a wideband digitized signal to a channelized representation, usually computed efficiently using a Fast Fourier Transform (FFT) algorithm, for which the computational cost generally scales slightly faster than linearly with the transform size. In comparing the different platforms the real-time computational efficiency of an FFT of the required size on each was used as a primary performance measure. In the case of multi-stage solutions the sum of the sizes of the stages of the FFT was considered. The real-time execution also drives both compute rate, and the input and output rates that need to be sustained.

Four different technologies were considered: Application-Specific Integrated Circuits (ASIC), Field-Programmable Gate Arrays (FPGA), general-purpose Graphics Processing Units (GPU), and Central Processing Units (CPU). These technologies were compared based on various figures-of-merit to decide on a proposed platform. Table 2 lists F-engine specifications relevant to the comparison of platforms and the performance figure of merit each drives.

Parameter	**Requirement**	**Impact**
Maximum baseline	300 km	Coarse-delay buffer memory
BBC Bandwidth	∼8 GHz	Data throughput and FFT size(s)
Sample resolution	4-bit (in) & 4-bit (out)	Input and output data rates
Spectral resolution	∼1 kHz	FFT size(s)

Table 2 A small subset of overall specifications are the key drivers of F-engine performance and thus drive the platform selection.

ASICs were essentially ruled out based on the very high costs associated with implementing solutions in these devices, as compared to an FPGA solution. Apart from the logic design, which could reasonably be expected to be similar to that needed for FPGA, lower-level design is also needed. Furthermore, high-speed I/O and other IP crucial to the present application which comes readily available in many high-performance FPGAs might add to the design cost, either in the form of custom-design of these solutions or obtaining license from ASIC IP vendors. Finally, given the relatively low volume of units needed the mask and yield ramp-up adds a considerable per-unit cost for using this technology.

CPUs were eliminated on the basis of poor computational performance in comparison to GPUs without any cost benefit. The capability of contemporary GPUs, in terms of operations-per-second and memory bandwidth relative to cost, is vastly superior to that of the same generation of CPUs, so that achieving the same performance in CPU as in a single GPU would require multiple processors. This would increase cost substantially, not in the least due to the number of CPUs required, but also as a result of additional hardware infrastructure needed to combine multiple processors.

GPUs, which in recent years have become a popular technology in High-Performance Computing (HPC) applications, offer a competitive alternative to FPGAs in terms of both power consumption and per-unit costs. However, the FFT algorithm, especially for large sizes has a relatively low computational intensity — the number of calculations performed per each byte read from memory. In practice this translates to the calculation being memory bandwidth bound, and the compute rate capability of the platform being severely under-utilized.

Parameter	**Specification**
Coarse-delay buffer memory	128 Mb
Data throughput	∼8 M channels / ms
Input and output data rates	64 Gbps

Table 3 Key F-engine specifications which the chosen platform's performance has to meet or exceed.

Ultimately the FPGA was selected as the preferred platform. They offer a much higher degree of flexibility in terms of routing data between arithmetic units, enabling effective utilization of the overall compute capability. In addition, given that high-speed communication is a technology driving application for FPGAs, current and upcoming generations offer sufficient input and output rates so that the required throughput in the F-engine application can easily be sustained with relatively little additional hardware being required. In table 3 the parameters which the selected FPGA platform need to meet are listed, and indeed the FPGA does have the resources to meet all these requirements.

2.3 Determine F-engine architecture given chosen DSP platform

Work Package (WP) 2.2 recommended the use of a Field Programmable Gate Array (FPGA) for the so-called *F-engine* which is the Digital Signal Processing (DSP) engine that transforms the time domain sampled data from the Analog-to-Digital-Converter (ADC) into frequency domain spectra. The purpose of this work package was to evaluate various FPGA-based *architectures* for the F-engine and determine the most effective option—that meets all scientific requirements—for implementation with near-future FPGA families.

Since the ADC for next-generation ALMA is not within this Study's scope, certain assumptions had to be made about the data it will provide to the F-engine. The most critical are:
1. sample rate will be exactly 16 GSps with a sample format of 4 bits, and 2. when the operating mode requires Walshing, the step time of the 90-270 switching (used for sideband separation) will remain exactly 16 ms.

2.3.1 Motivation for Two Modes

Over the course of this study it was determined that two constraints, arising from our assumptions and the scientific requirements, when combined drive the complexity of the F-engine architecture beyond reasonable implementation. Briefly summarized these constraints are:

1. **Walsh switching**: to effectively demodulate the 90-270 Walsh pattern there must be an integer number of F-engine input sample windows within on Walsh step; ideally this would be a large number to avoid blanking losses. Additionally, this constraint drives non-2^n transform sizes.
2. **1 kHz resolution**: the scientific requirement that the correlator provide a final resolution of ~ 1 kHz drives very large transform sizes which mean large input sample windows.

Given the immutability of the sample rate and Walsh step time, a single-mode[2] F-engine implementation cannot reasonably meet both constraints as this would mean very large power-of-5 transforms with large input windows and unaccpetable blanking losses; one constraint or the other must be relaxed. Therefore this study group proposes the following two modes for the F-engine:

1. **Walsh mode**: full Walsh switching[3] is enabled but the ∼1 kHz spectral resolution is increased to ∼100 kHz, i.e. a relaxation of constraint 2. This would mean small blanking losses of $\sim 0.06\%$. *Note: this mode still requires a non power-of-two, split-radix transfrom.*
2. **LO-offset mode**: no Walsh switching[4] but LO offsets are used for spurious signal rejection, i.e. a removal of constraint 1. This mode incurs no blanking losses (since there's no Walsh switching) and the window size can be large and a power-of-two. *Note: in this mode the spectral resolution will not be exactly but instead approximately 1 kHz.*

These two modes naturally match the double-sideband (bands 9 and 10) and sideband-separating (bands 1-8) receivers, respectively, of ALMA. However, if an astronomer desires stronger sideband rejection in bands 1-8 then mode 1 can be employed but at the sacrifice of poorer spectral resolution.

2.3.2 Architectures Studied

In the context of general algorithms for the F-engine this working package explored numerous architectures for the transform itself, including:[5]

1. Single-stage channelizer
2. PFB followed by per-channel DFT
3. Two-dimensional FFT/PFB
4. Prime Factor Algorithm FFT
5. Tunable Filterbank followed by per-channel PFB

These architectures were explored in the context of the specifications with estimated resource usage determined for a modern FPGA. Additionally, various other sub-systems relevant to the F-engine were investigated such as delay tracking and complex gain multiplication.

Figure 2 shows the last of the five considered F-engine personalities, which use a "Tunable Filter Bank" or TFB stage for high spectral resolution. Though this exact architecture was ultimately

[2] In this context a *mode* is understood to mean a FPGA personality.

[3] Full Walsh switching means both 0-180 for spurious signal rejection which is taken out with sign flips at the sampler and 90-270 for sideband separation which falls to the correlator

[4] Similarly no switching means neither 0-180 nor 90-270. The LO offsets will take care of spurious signal rejection however are not used for sideband separation

[5] Not listed here is the oversampled Polyphase Filterbank architecture which was proposed but time constraints precluded its study.

rejected, the block diagram shows enough relevant detail, ad clearly shows the complexity which is subsumed into the FPGA. . For more details of the proposed transform architecture and of the other F-engine subsystems please see the full report in the study appendix.

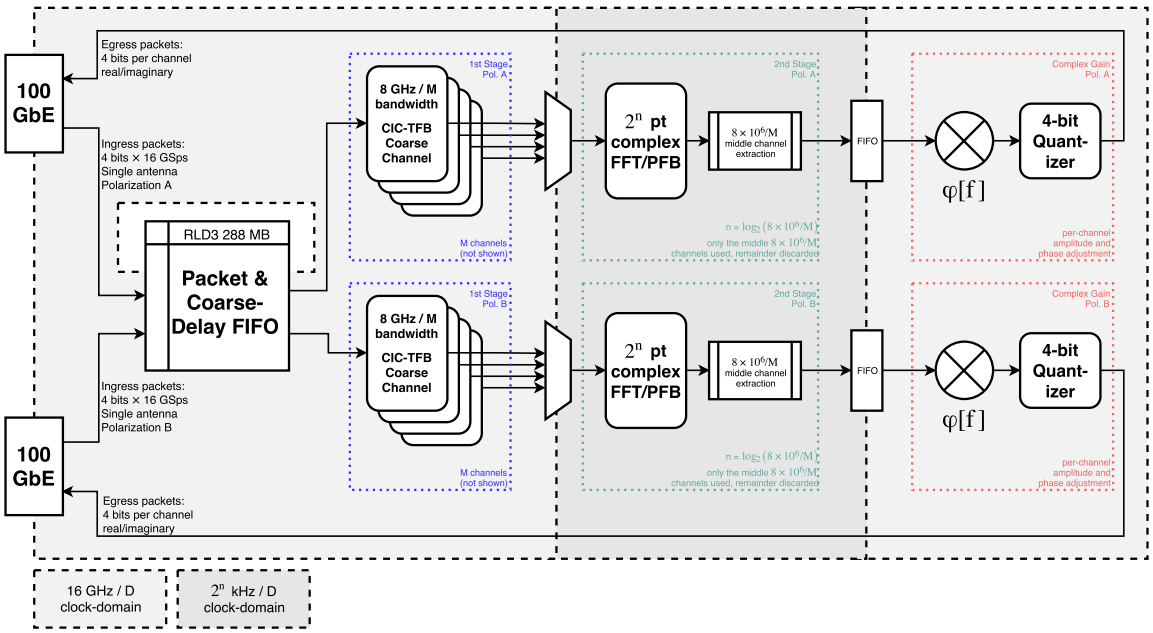


Figure 2 An example of a high spectral resolution candidate F-engine gateware personality that uses a TFB first stage channelizer followed by FFT or PFB (selectable) second stage fine channelization per coarse channel. The incoming data packets contain 4-bit samples at 16 GSps; these packets are received with the 288 MB RLDRAM3 modules available on the VCU118 which is also used as a coarse geometric compensation FIFO. The data from each polarization is then fed to the first stage channelizer which contains a series of TFB channels each of which downconverts a sub-band of the full 8 GHz passband; note that the number of TFB channels, M, will generally be a low number, for example, 64, since the TFB logic scales linearly with M. Each TFB channel, containing 8 GHz/M in usable bandwidth, is intentionally oversampled to the next power-of-two in kHz so that the proceeding second stage channelizer can be a power-of-two and the resulting fine channels are exactly 1 kHz wide. Ultimately the number of fine channels leaving the F-engine will be exactly 8 million but we've circumvented ever having to do a non-power-of-two transform. Note that only a single second stage channelizer needs to be instantiated since each TFB output is downsampled by a factor of M canceling out the fact that there are M coarse channels. Following fine channelization the complex gain subsystem provides per-channel complex gain multipliers which can used to implement sub- sample delay adjustment, 90-270 deWalshing, amplitude and phase bandpass corrections, etc. Finally the data is quantized back to 4-bits and shipped out to the BX-engines. Many subsystems are not shown, such as test vectors, monitor-&-control & (de)packetizers.

2.3.3 Conclusion

Much time and effort during the study of this working package went into architecting a single-mode F-engine before it was determined that a two-mode system fit the requirements more comfortably. Nevertheless these efforts were not in vain as much overlap exists between the **LO-offset mode** and what would have been a single-mode. After careful consideration of the above listed architectures we recommend the following for the transform architecture:

1. **Walsh mode**: 160 kilopoint (256 x 625) split-radix two-dimensional FFT
2. **LO-offset mode**: 2^{24} point (4096 x 4096) two-dimensional FFT

The appendix for this work package has a great deal of detail on the five alternative architectures considered, and the reasons for the selection of these two modes. The oversampled PFB also shows promise, and is proposed to be explored in the follow-on Project.

2.4 Identify corner-turn platform

Interconnection systems may be divided into two classes. Actively switched systems can dynamically route data from a source to any of several endpoints. These systems include Ethernet, Infiniband, and some PCI-Express based motherboards/backplanes. Passively-routed systems simply provide point-to-point connectivity from sources to endpoints. Examples of these systems are simple backplane meshes, and point-to-point connections made with optical fiber or copper cabling. A very brief overview of the applicability of these systems to an upgraded ALMA system is given below.

2.4.1 Point-to-Point Interconnect

LVDS Copper Cabling The present ALMA correlator uses 16384 LVDS twisted-pair cables operating at 250 MHz, representing "the greatest design challenge in the system" (Escoffier et al., 2007), to connect the station cards (equivalent to F-processors) to the correlator cards (equivalent to the X-processors). With the increased specifications of the next-generation ALMA correlator a corner-turn implementation using the same technology would see the total number of cables increase roughly five-fold, mainly driven by the doubling in bandwidth and sample bitwidth. Even assuming a per-lane speed increase by a factor of two or more, the complexity of such a cabling system is highly undesirable.

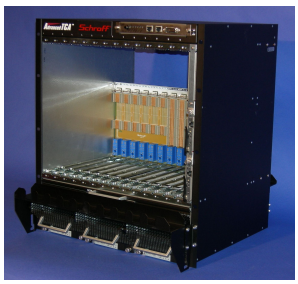
Copper Backplane The most promising copper backplane standard is the Advanced Telecommunications Computing Architecture (ATCA). The latest standard, PICMG 3.1, supports 40 and 100 Gb/s connections. ATCA enclosures can be purchased off-the-shelf, for $\sim 10\,$k$, and provide all-to-all connections between up to 16 computing cards (Figure 3(a)). While some correlator realizations may be compatible with interconnect based on one or more independent ATCA enclosures, in general it may be necessary to externally mesh together multiple such units, resulting in undesirable complexity and cost. Further, requiring computing units to be ATCA-compatible greatly increases the likelihood that they must be custom-designed, with significant associated NRE.

Fiber Circuitry For a one-off NRE fee of $\sim 10\,$k$ custom fiber-based interconnection circuits can be fabricated, providing practically any routing of inputs to outputs (Figure 3(b)). These devices can be used as part of short-, mid-, or long-range fiber runs. Provided the processing nodes at each end of such a system have adequate independent IO paths to drive the required number of fibers, fiber optic circuitry is a very cost-effective way of providing interconnect. It is already being used in astronomy applications (Hampson et al., 2013). A system involving fiber circuitry interconnect likely involves some engineering NRE to design or adapt a platform to be able to interface with the fiber circuit.

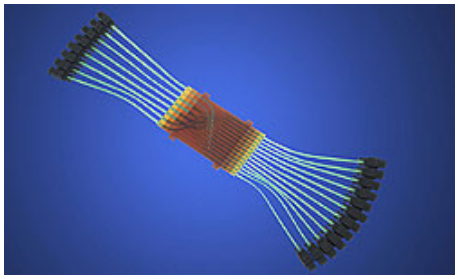
2.4.2 Active Switching

PCIe switches PCI-Express is a common standard for connecting many processing boards via a backplane type configuration supporting transfer speeds up to 125 Gb/s per endpoint (for Gen3 with 16 lanes). Additionally the standard allows data transfers between slaves by using bus mastering. All PCI-Express endpoints, however, must connect to either a root complex or a switch and given these devices with 16 or more endpoints are rare or non-existent we will not consider this technology for the proposed ALMA correlator.

Ethernet Switching Ethernet (or Infiniband) switches provide high-speed, flexible interconnect, with industry standard interfaces widely supported by commodity hardware such as FPGA, CPU/GPU platforms. The main draws of Ethernet switch systems are their extreme flexibility, tolerance to system architecture changes, and availability as COTS units, without hardware NRE. Ethernet switches are already being extensively used in the MeerKAT array, which features a digital backend with unprecedented reconfigurability (Manley, 2015). At current prices, the cost of an Ethernet-based interconnect solution for the entire, multi-BBC ALMA system is approximately 500 k$ – 1 M$, depending on choice of switches and cables. However, this cost is likely to fall significantly as the 100 Gb Ethernet standard becomes more mainstream, or 400 Gb Ethernet appears in COTS parts. The cost may also be reduced by a factor of two or more by considering the port-reduction schemes suggested by McMahon et al. (2007). Furthermore, the cost of investment in an Ethernet architecture may well be dwarfed by savings associated with being able to use COTS computing hardware for processing modules.



(a) An enclosure with copper mesh interconnect provided by an ATCA standard backplane. This backplane supports 40 Gb/s all-to-all connections for up to 14 cards.



(b) Molex FlexPlane^{TM} fiber circuitry provides user-customizable fiber-based mesh interconnect.

(c) Actively switched interconnect, provided by COTS Ethernet switches, with industry standard high-speed ports, operating at 100 Gb/s

Figure 3 Three interconnect options based on current technologies.

2.4.3 Conclusions

The final choice of interconnect technology used by a next-generation ALMA correlator will need to be made in light of a system-engineering overview of the instrument as a whole. If large engineering budgets are to be dedicated to developing custom platforms on which to implement the correlator's signal processing, it way well be the case that integrating support for a fiber circuit interconnect is the most appropriate design decision. However, if a choice is made to adopt general-purpose COTS compute platforms, such as modern FPGA or CPU/GPU platforms, an NRE-free Ethernet switch interconnect is likely to represent a cheaper total cost.

Given that an Ethernet-based interconnect solution is clearly feasible at the scale of the proposed ALMA correlator, and likely represents a relatively small part of the total hardware budget, our opinion is that this technology is the preferred choice, given the uncertainties in the other aspects of the correlator design. Choosing an Ethernet fabric interconnect maximizes the flexibility of the digital backend. Furthermore, should hardware development of a future correlator commence, such a choice would make it easy to prototype an effectively production-ready subset of the complete ALMA system and provides a clear path for staged deployment.

2.5 Identify DSP X-engine platform

The X-engine platform will perform element-wise multiplication of spectra for each pair of antennas received from the F-engine, via the network interconnect, and to accumulate the products, for each frequency channel individually. As dictated by the two modes of the F-engine, the packetized X-engine platform will accept data with 1 ms and 0.01 ms Nyquist windows. Delivery of variable time and frequency resolutions for end-use is implemented following and independently of the cross-multiplication, the priority being simplicity, modularity, and minimization of the number of modes and reallocation of resources.

The study considered two leading off-the-shelf technologies for the platform: FPGA and GPU. The computational architectures and models for these are very different (see appendix), but despite that, the primary finding is that with commercially available components as of 2018, both are capable of supporting a low-cost, power-efficient, physically compact X-engine. The caveat to this is that lab development and testing of hardware, software, and firmware are needed to confirm the finding, or identify differences in performance.

Considering the large instantaneous bandwidth of ALMA and the number of antennas, contemporary computational elements (FPGA or GPU) are sufficiently "powerful" that operation is bandwidth-bounded. Bottlenecks, if any are most likely tied to working with the high density of data passing through memory and computing elements. Per computing element, the adopted capture rate from the network is 200 Gb s^{-1}. This limit reflects a reasonable scaling of current network link capacities, though in practice, computational elements of the X-engine, scaled into the future, would not be saturated.

Either technical solution for the platform, assuming commercially available components as of

2018, would incur a cost of ∼$220K (GPU) and ∼$300K (FPGA) to process one dual polarization ALMA BBC. For current engineering assumptions, this would consume 4.4 kW or 2.5 kW depending for the choice of platform, respectively. The estimated total cost of all the core elements of the X-engine serving four BBC pairs would be within ∼20% of $1M, slightly favoring the GPU solution. Total power consumption would be between 10 and 17 kW (not including hot-spare computing and associated network components that add to cost up front but reduce operating costs in the long term).

The primary challenge for the FPGA solution is operation of the necessary firmware at clock rates > 500 MHz for current engineering tools and chips. The next uncertainty prior to detailed engineering analysis and testing, lies in whether off-the-shelf processing boards will meet ALMA requirements cost effectively, or whether custom boards will be required.

In contrast, the complexion of prospective GPU hardware, dictated by industry standards and market drivers (e.g., driverless cars, deep learning), is known. The proposed ALMA scheme relies on four developments: (i) ingest from a pair of (modern) 100 Gb s^{-1} links to a single GPU with a system-on-chip architecture (SOC), (ii) inbuilt many-core ARM processor and shared memory, (iii) PCIe4 external bus, and (iv) quad-rate (GPU) 8-bit math capability. The last three are already featured in production hardware, and the primary challenge lies in engineering one-way network capture at 200 Gb s^{-1}, without packet loss. The record known capture rate as of 2014 into a server is 80 Gb s^{-1}, which is divided between two GPUs, and achieved with low throughput 10 Gb s^{-1} links. The target Tegra GPU hardware will be introduced to the retail market in 2018. Examples of requisite individual technologies are already available in products. In particular, present-day non-SOC, GPU cards have been reported to support at least 100 Gb s^{-1} ingest.

The secondary challenge tied to a GPU platform is output of autocorrelation spectra for each antenna and BBC pair on time scales O(1-10) ms without bus saturation inside the GPU. In the event the associated memory traffic interferes critically with cross-multiplication and accumulation, the fallback is to rely on the ARM processor to compute self-products and export them (corresponding to $< 1.3\%$ of the input data rate) to the network.

2.6 Determine optimal X-engine architecture

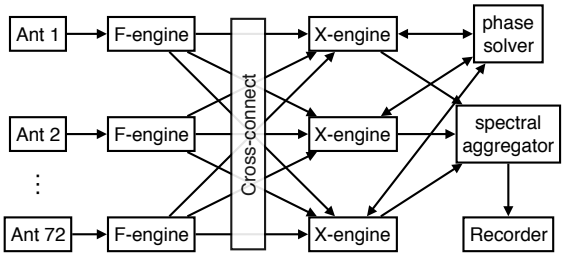


Figure 4 Overall FX correlator system architecture. Each X-engine units accepts as input a small portion of the full spectrum to be correlated from all antennas. The full-polarization cross products are calculated and accumulated before being sent back through the network for further processing. The X-engine also accepts real-time phase calibration solutions from a VLBI phase calibration engine, which are used to beamform the array data to targeted locations in the field-of-view.

The X-engine nodes process spectral data gathered from the F-engines, and send accumulated correlation products and synthesized beams back over the network. Figure 4 shows a simplified system digram with the essential elements of the correlator. Each X-engine unit accepts a fraction of the total bandwidth from all antennas, and the correlation of these spectral ranges are done in parallel. External elements of the special beamforming subsystem include the real-time phase calibrator, which aggregates visibilities across the entire bandwidth in order to solve for a unique set of antenna phases, and the spectral aggregator which accepts slices of beamformed data from all the X-engines in order to reconstruct and reformat a beamformed data stream to a given specification.

The X-engine platform chosen in the cost and power-comparison study (WP 2.5) is a cluster of Nvidia system-on-chip (SoC) devices connected to the correlator through two 100 GbE NIC over a PCI-Express version 4 backplane. Together the three devices make up an "X-engine unit", and each unit operates independently on a small slice of the total array bandwidth (∼160 MHz) prior to final downstream accumulation and processing. Each SoC device includes an ARM multi-core CPU, an Nvidia GPU, both connected to a moderately sized (many GB) block of Unified Memory.

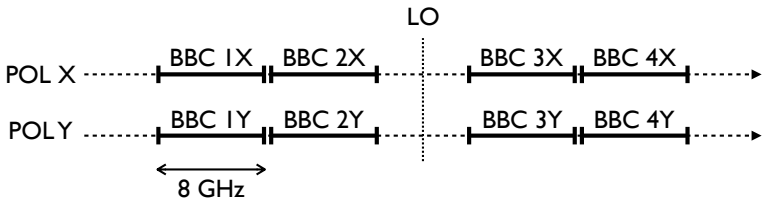


Figure 5 Frequency setup and location of 8 GHz Nyquist-sampled baseband channels (BBC) for one dual-polarization ALMA receiver. Adjacent BBC's may overlap slightly if necessary for continuous coverage of usable bandwidth, although the correlator is designed to be able to process the full 8×8 GHz of bandwidth. With 4-bit sampling, the total data rate is 8×64 Gbps per antenna.

Table 4 Resource requirements of X-engine, given optimal processing of 4+4 bit complex input data, 32+32 bit complex accumulated output, and 4+4 bit complex beamform output. Output rate is calculated at 100 kHz channelization × 1 second accumulation, from which figures can be directly scaled to other resolutions. Memory usage and bandwidth are calculated for 1.6 kHz uniform resolution (100k total channels per unit) and 0.1 second initial transpose buffer, plus 4 output beams. Finally we show the equivalent capacity of a hypothetical platform based on the next-generation Nvidia Xavier AI SoC paired with two 100 Gbps NIC's on a PCIe-v4 backplane. DLOP (deep learning operation) refers to one 8-bit multiplication with 32-bit accumulation.

Resource	**Total**	**per node (200)**	**Xavier AI unit**
DLTOPS	1419	7.1	20
Input I/O	4.6 TB/s	23 GB/s	25 GB/s
Output I/O	27 GB/s	0.13 GB/s	25 GB/s
Beamform I/O	256 GB/s	1.3 GB/s	-
Transpose memory (0.1s)	9.2 TB	4.6 GB	16–32 GB
Accumulation memory (1.6 kHz)	1.7 TB	8.4 GB	16–32 GB
Memory bandwidth	25.2 TB/s	126 GB/s	160 GB/s

Anticipated specifications for Nvidia Xavier SoC (sampling Q4 2017) are listed in table 4, along with the anticipated resource requirements distributed across 200 units. While future generation hardware is likely to be available by the time of correlator hardware acquisition, this study outlines an X-engine architecture that fits within the limitations of the anticipated Xavier platform.

Fundamentally the cross correlation operation performed by an X-engine is trivially parallelized over time and frequency as each spectral point is independent. Efficient pipelining of the staging, correlation and accumulation of antenna data in an FX correlator are available for GPU (Clark et al, 2012) and FPGA (Parsons et al, 2008) based architectures. In both architectures, the input data is first transposed so that data are sent to the X-engine in an order such that nearby data are the ones being accumulated. Beyond this transpose, the X-engine can be relatively agnostic to the specific parameters of accumulation. For GPU correlation, a tiling strategy breaks up the full array matrix outer product into blocks which enables efficient coalesced memory transfer, as well as a high degree of parallelization of smaller calculations across the $\sim O$(hundreds) of GPU cores.

A complete time-frequency transpose is not possible for 1 kHz spectral resolution resolution due to the large amount of memory needed to buffer the full integration time of 30 seconds or longer. Instead only a partial transpose (∼0.1s) occurs prior to correlation, and further accumulation is done on temporary accumulation products staged to device memory. Balancing the memory and memory bandwidth requirements of these two buffers (transpose and accumulation) is a challenge given the specifications of the ngALMA correlator. A high-utilization case of uniform 1.6 kHz resolution and 0.1 second initial transpose is shown in table 4. The requirements can be reduced by only keeping a fraction of the bandwidth at such high resolution (zoom mode).

A beamforming sub-system is present in the X-engine design which allows phase alignment and stacking of antenna data to form up to 4 synthesized beams on the sky. For real-time on-source phase calibration, the X-engines send in near-realtime cross visibilities to a single phase calibration engine, which solves for time-dependent antenna-based phase variations due to the atmosphere. The phase solutions are sent back to the X-engine units, which use the residual phase information to stack the data from all 72 antennas coherently at up to 4 phase reference locations within the

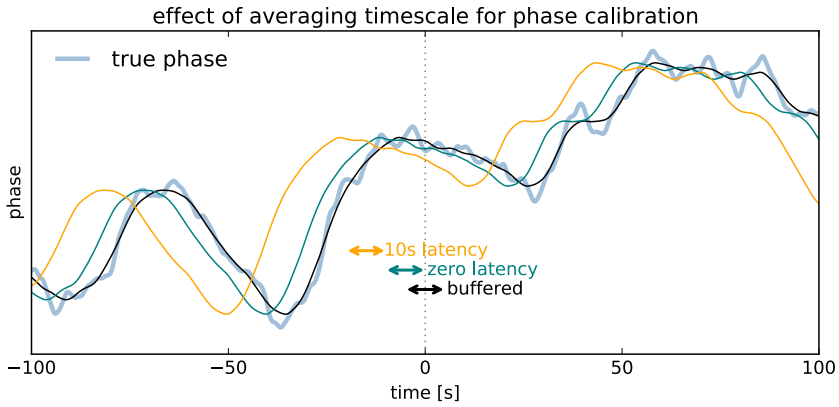


Figure 6 Effect of latency on estimate of antenna phase. The real-time phase calibration feedback latency of the new correlator is expected to be under 1 second, providing more stable band 6 phasing under moderate atmospheric conditions than the existing system. A data input buffer that is ∼25% the atmospheric coherence timescale or longer also significantly improves phasing performance.

primary beam. The effect of latency on antennas phase is illustrated in For a handful of beams, the resource use on the X-engine for this $O(N)$ operation is minimal, and it can easily be supported in parallel with the cross correlation.

2.7 VLBI capability

Beamforming the ALMA dishes creates a high sensitivity VLBI capability for ALMA that can be used to anchor centimeter, millimeter and submillimeter VLBI arrays for ultra-high angular resolution and sensitivity science applications. A full science case for ALMA beamforming is detailed in Fish et al (2013). The Next Generation ALMA Correlator will have native beamforming capability that far exceeds that of the present ALMA Phasing System (APS), enabling VLBI at high frequencies and under a variety of atmospheric conditions.

2.7.1 Beamforming requirements

Beamforming for VLBI and pulsar applications imposes several specific requirements, some of which are necessarily dependent on the atmospheric conditions, array configuration and observing Band.

- Phasing efficiency of the antenna grouping in a coherent sum will be $> 95\%$
- Phasing of the array will be done as near to real-time as possible. This requires that either:
  - the target phasing efficiency can be achieved in an integration time short compared to the atmospheric coherence time on a calibrator source, *and* that system latencies are also short compared to atmospheric coherence, or
  - a buffer is used to store data so that phasing solutions can be applied to the data used in the solution.
- A real-time measure of phasing efficiency should be computed.
- Polarization leakage in the phase sum should be no greater than the average leakage for a single antenna.
- Data output of the phasing system should be available in standard VLBI format (2, or 4-bit data with suitable headers - e.g., VDIF).
- Multiple beams may be formed within the primary beam of the ALMA antennas
- Beams may be formed on sub-arrays of antennas and on sub-bands in frequency
- Phasing efficiency shall be as stable as the atmospheric coherence timescale
- Several modes of phasing should be implemented: phasing on in-beam target, phasing on in-beam calibrator, phasing on out-of-beam calibrator.
- Should be capable of correcting for source model and time variable atmospheric screen.
- Phasing should be available for all ALMA Bands.

- For the pulsar case, the requirement is to be able to detect millisecond pulsars with a Dispersion Measure of 3000 pc cm^{-3}. This sets an upper limit on channelization of 32 MHz for ALMA Band 1. For pulsars, it is also desirable to maintain the maximum number of bits possible, but 2-bits are sufficient if any auto-leveling system has a time constant greater than 5 seconds.

2.7.2 Phasing Flux Density and Integration Time Limits

When all 72 antennas are combined and the full 64 GHz BW used to beamform, the required phasing efficiency can be obtained for short integration times that are not affected by atmospheric coherence and on sources down to a flux density of $\sim$ 10 mJy (Fig. 7). Because this figure assumes zero coherence losses due to atmospheric effects, including due to latency of the phasing solution, these flux density limits are understood to be lower limits.

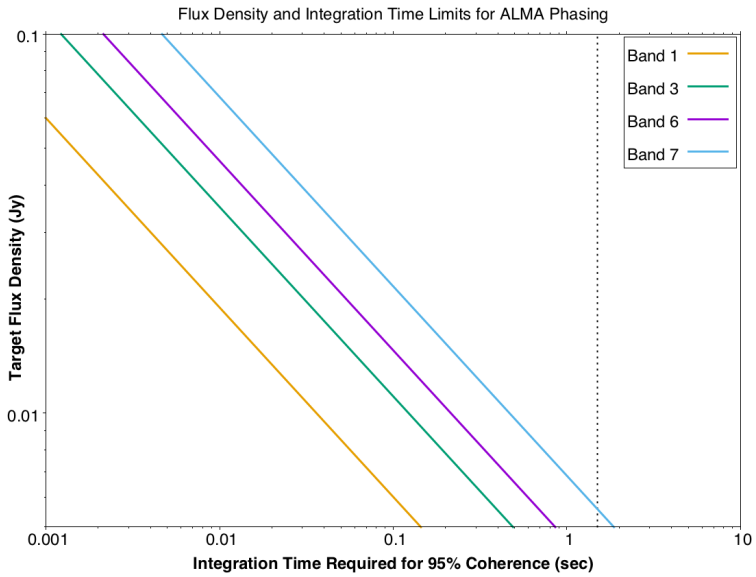


Figure 7 Relationship between flux density of the source used for phasing and the integration time required to achieve 95% coherence in the phased sum for four ALMA Bands. Limits assume 72 dishes, use of the full 64 GHz BW in the phasing solution, and SEFD (Jy) for a single ALMA dish of 1300, 2400, 3200, and 4700 for Bands 1, 3, 6, 7 respectively. The vertical dashed line marks the coherence time of the array in Bands 6 & 7 for a PWV of 3mm (see Matsushita et al 2017). This figure assumes zero coherence losses due to atmospheric phasing effects, including latency. For the Bands shown, one expects to achieve coherent phasing for source flux densities down to 10mJy in integration times that are short compared to the atmospheric coherence time.

2.7.3 Latency and Buffering

Latency in the context of beamforming is the offset in time between collection of the data for which phasing solutions are found and the application of those solutions to the array. For the current ALMA correlator and phasing system, there is a latency of 8-10 seconds, so that at any given time the coherent sum is being formed using phasing solutions that are 8-10 seconds out of date, and atmospheric turbulence, or changing phasing conditions of any kind, will cause coherence losses. The effect on coherence is shown in Fig. 8. In the next generation ALMA correlator it is expected that streamlining transfer of data, solving for phases, and implementing the solution will reduce latency to 10's of milliseconds. Since this is short compared to atmospheric coherence times, as is the integration time required to phase on typical calibrator sources (Fig. 7), phasing on in-beam targets should not require a data buffer.

For cases where slewing to a phase calibrator is required, a buffer that can store data for the duration of a fast-switching sequence may be useful. Such a buffer ($\sim 10 - 20$ seconds) could potentially be included in the X-engine architecture. For 72 antennas, 4-bits, 64GHz of bandwidth, each second of buffering requires 4.5 TBytes of memory.

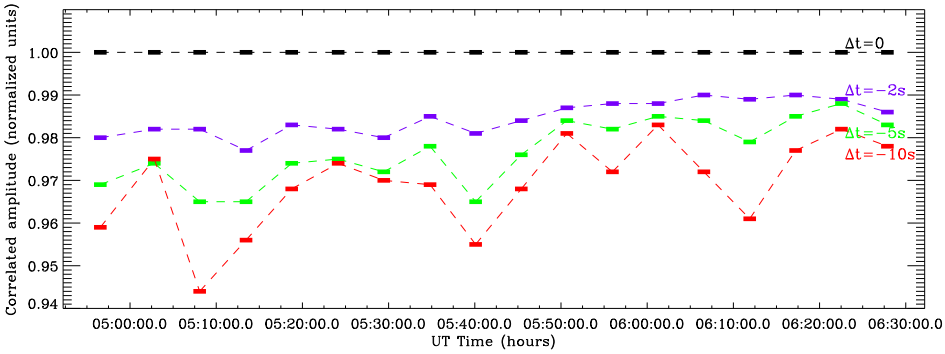


Figure 8 Coherence loss due to latency between the phasing solutions applied to the array and the data used to calculate those phasing corrections. For a 10 second latency, phasing efficiency in relatively good atmospheric conditions (PWV 0.9mm, RMS path length variations of $\sim$ 125μm on 300m baselines: mean conditions at ALMA in May), can drop by 6%. This example uses Band 6 data on quasar 1924-292 from ALMA with phasing done a posteriori in CASA with 16 antennas summed. Figure made by Lynn Matthews.

2.7.4 Data Format, Data Transfer & VLBI Recorders

Modern VLBI recorders are essentially packet capture devices, that are currently capable of 16Gb/s. Packet headers, using the VDIF format, contain all the information required for routing of data in a VLBI correlator, including time-tagging. The VDIF format may still be useful in 2022 when the ngALMA correlator is constructed. It is also possible that network appliances that are essentially just packet recorders - perhaps with Solid State Storage - could replace VLBI recorders. Capturing a single 64 GHz beam at 2-bits (256 Gb/s) could potentially be supported by extension of current VLBI recorder architectures, but to capture 4 beams, each 64 GHz bandwidth and 4-bits, the data rate would be 2048 Gb/s, for which a new generation of recorders would be required.

2.8 Staging the new correlator and phased array system

A study of the costs and trade-offs associated with four possible locations for the new correlator was one of the work packages completed. We studied the implications of siting the new correlator at the Array Operations Site (AOS) and the Operations Support Site (OSF). Also, we considered both new construction and re-purposed space in both instances. Both real costs and the costs of lost science and productivity are included in the summary. Table 5 covers construction costs, and table 6 covers operational costs. Assumptions that were made, as well as the basis for various costs, are summarized in the full work package report, included in the appendix.

Table 5 A summary of the costs of construction of a new correlator, including both the direct costs of construction—a function of whether it is sited in existing space or in new construction—and the opportunity cost of observation time lost due to new construction in space presently occupied by mission critical equipment.

Item	AOS Existing	AOS New	OSF Existing	OSF New
Rackmount	$10k	$100k	$100k	$100k
HVAC	$164k	$164k	$58k	$158k
Construction	$0	$1.3M	$320k	$ 708k
Install time	$50k	$50k	$50k	$50k
Travel	$10k	$10k	$10k	$10k
Signal Transport	$0	$20k	$2.4M	$ 2.4M
Total	$380k $1.644M	$3.088M	$3.426M	
Lost Science Time	$10M	$0	$0	$0

An interesting point is that the estimate of the cost of lost observation time, which is an opportunity cost based on the capital investment ALMA represents amortized over the expected 30

Table 6 A summary of the costs of operating a new correlator, including both the direct operational costs, which depend on where the new system is located, and the imputed cost of lost observation time due to delays in rendering repairs to equipment located at the high site.

Item	**AOS Existing**	**AOS New**	**OSF Existing**	**OSF New**
HVAC	$1190k	$1190k	$2247k	$2247k
Lost tech prod.	$400k	$400k	$0	$0
Vehicle costs	$62k	$62k	$0	$0
High alt bonus	$37k	$37k	$0	$0
Risk to personnel	?	?	$0	$0
Total	$1.689M	$1.689M	$2.247M	$2.247MM
Lost Science Time	$59M	$59M	$0	$0

year lifetime of the instrument, is the dominant cost in two of the four cases studies. In particular a roughly estimated cost of $59M was assigned to observation time lost due to the delays in making repairs to faults for equipment located at the high site. While this estimate is highly uncertain it was based on experience with the current correlator, with more details of the calculation in the WP2.8 appendix.

The OSF becomes a viable choice for siting the new equipment if this important opportunity cost is considered. Developments in fast data communications technologies with rates 100Gbps+ have made consideration of siting at OSF possible. As a counterpoint, one benefit of the proposed packetized FX architecture is that if one processor fails another can dynamically be re-assigned. Thus it is at least plausible that the new system may be more reliable, or at least fail in a softer way, which could potentially improve the extremely high opportunity cost of maintenance at AOS estimated here.

We emphasize again that the uncertainty in the numbers is high. Even so, opportunity cost is a real cost, even if not representing a cash outlay, and the potential return estimated here in our opinion means that this siting question warrants further study. We recommend that JAO carefully review the trade-offs and actively participate in the discussion of where best to site this next generation correlator and phased array.

3 A conceptual design and buildout roadmap

This section outlines a conceptual design and roadmap for full realization of a next-generation ALMA correlator and phased array with specifications set in section 2.1, and technologies and algorithms based on those selected in this Study. The overall program for development is split into three phases with a timeline that is aimed at completion and commissioning of a new digital system by 2026. Following the general principles described in this Study, the new system will minimize design time, optimize use of the latest "Commercial Off-The-Shelf" (COTS)components, and be capable of supporting all projected ALMA upgrades referenced in the community documents: *A Roadmap for Developing ALMA*, and *Pathways to Developing ALMA*. The plan for the full buildout Phase III is of course tentative in nature. It is provided to demonstrate that the Study solutions are broadly feasible on reasonable timelines and at tractable cost.

The three phases of the program are:

- *Phase I - Study*: This phase is now complete and the results are described in this document. This phase has assembled an expert team with experience in correlator design for centimeter, millimeter and submillimeter wavelength arrays, including developers of the original ALMA correlator. The study has drafted first science requirements and specifications, identified a design approach, and explored computational platforms and possible architectures.
- *Phase II - Project: Prototype*: This phase transforms the results of the Study into a detailed design and construction of a prototype system capable of processing a subset of ALMA antennas and bandwidth: 8-stations, dual-polarization, 8 GHz bandwidth. The project includes comprehensive laboratory testing using an antenna emulator engineered to allow iteration on the design and refinement of cost estimates. Phase II will develop a resourced schedule for building, installing, and commissioning the full system. This phase has been proposed as an ALMA Development Project by the Study team, with participants across ALMA regions.

- *Phase III - Implementation, buildout and commissioning*: With completion and testing of the prototype, the program continues with a formal review of the concept and a subsequent series of milestones including Preliminary Design Review (PDR), Critical Design Review (CDR), and multiple readiness reviews. Preliminary Acceptance In-house (PAI) and Preliminary Acceptance on-Site (PAS) will be completed for all subsystems, followed by Assembly, Integration and Verification (AIV). Finally Commissioning and Science Verification (CSV) will be completed under the oversight of JAO. During this phase, the final hardware platforms will be decided and procured, software and firmware refined, interfaces to all ALMA systems detailed and integration and testing plans executed.

Phase II (2017-2020) has been specified with a 3-year schedule in detail in our ALMA Development Project Proposal. A tentative and approximate timeline and equipment costing is given for Phase III (2021-2026). Substantial refinement of Phase III is expected as Phase II is executed, and one of the of the deliverables of Phase II will be to identify the work packages for Phase III. Also by the end of the the Phase II stage we will supply a more accurate estimate of the total power and total cost for the full installation. This will naturally depend on whether the choice is made to install the new system at the AOS or the OSF, as studied in WP2.8, which will naturally also impact the cooling requirements.

3.1 System engineering and project management

At its heart, the next-generation Correlator and Phased Array studied herein is the central component of a system-wide bandwidth enhancement for ALMA. As such it presumes availability of wideband receivers, analog-to-digital converters, Digital Transmission System (DTS) and a data pipeline with commensurate increases in capacity and speed. And more than that, this study emphasizes that a next-generation Correlator will necessarily exist within the context of an overall system-engineering plan for ALMA Development. This conceptual design can serve as input to that process.

Our approach to system-engineering for this study is informed by the recently completed ALMA Phasing Project (APP), which was one of the first official ALMA upgrades that went through all phases of the Development Project process. This included close connection with the JAO on:

- Development and Design of the proposed system
- Establishment of Interface Control Documentation (ICD) with other ALMA systems
- Formal Science Requirements, Technical Specifications and Verification Matrix
- Integration and Test Plans
- Design Reviews that adhered to ALMA protocols
- Commissioning Plan
- Identification of impact on JAO personnel and resources
- Acceptance with full documentation, training and support for JAO personnel

Adoption of these project management elements has now resulted in the APP being made available to the ALMA community in Cycle 4 and Cycle 5 observations for ultra-high resolution and sensitivity science in Bands 3 and 6. Just completed Band 3 and 6 VLBI observations with the phased ALMA in April 2017 for Cycle 4 appear to have gone very well, and are testament to the technical, management and resourcing strategies of the APP as well as a strong partnership with the JAO.

With this positive experience as a foundation, our view is that the proposed approach to the next-generation ALMA Correlator, while ambitious and forward looking, can be realized in a staged manner that minimizes risk and maximizes science return per dollar. As an example, we would propose to first build a sophisticated 8 antenna ALMA simulator, with fringe Doppler, delay tracking, programmable SNR, that is capable of producing visibilities for a complex sky image. This enables full testing of next-generation prototypes without the need to stage them at the ALMA site.

3.2 Phase II: Prototype

The proposed correlator and phased array is an optimized supercomputer that takes data over a packetized network interface at its input, and delivers data such as cross-correlation fringe visibilities, and coherently phased sum of antennas, at its output. It has dual polarization, and an 8 GHz "BBC bandwidth" which readily scales by replication to the 64 GHz instantaneous sky coverage our Study requirements target. An ultra-fine spectral resolution setting of 1 kHz is available, useful

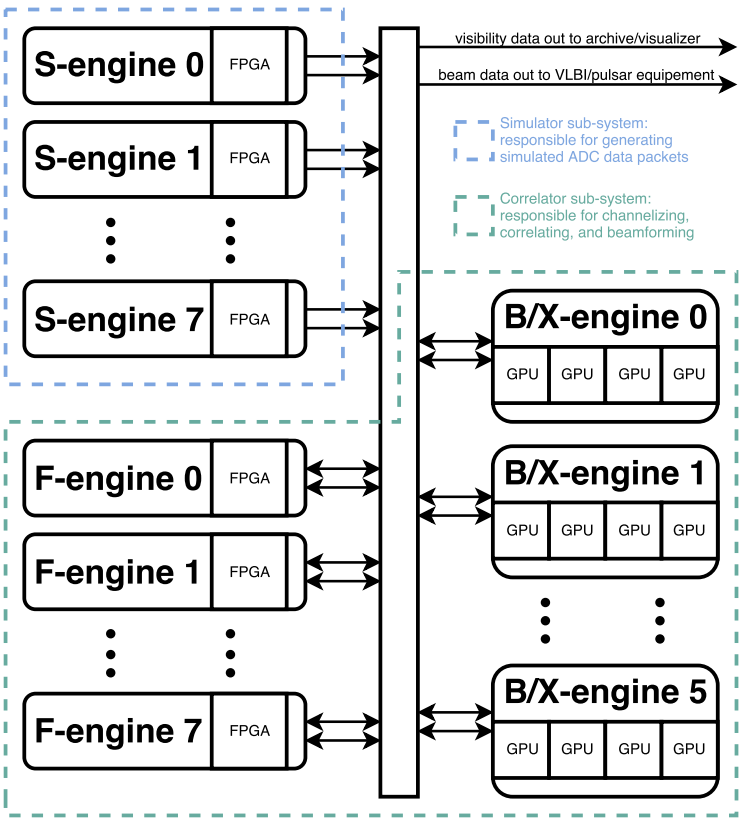


Figure 9 Simplified block diagram of the Project Phase II bench system at the top level showing the simulator and correlator sub-systems as they attach to the 100 GbE switch interconnect (the thin central rectangle). Data from the simulator, emanating from multiple "S-engines", is sent out to the switch in packets mimicking the eventual ADC- sample containing packets expected from the next-generation ALMA DTS system. The data is routed in a one-to-one pattern from S-engine to the channelizer boards, the so- called "F-engines". Both the S- and F-engine will be implemented using the Xilinx VCU118 FPGA and will emit and accept, respectively, two polarizations each. Every F-engine will then divide its outgoing channelized data and route subsets of channels to the corresponding "BX-engines", so called because they simultaneously correlate and beamform. The BX-engines then route the correlated and integrated data out of the correlator to the archive while the beamformed data is sent to either the VLBI or pulsar backends. Note that every arrow in the diagram denotes a single 100 GbE link to the switch, some are used bidirectionally (F- and BX-engines) while others are not (S-engines).

to resolve lines in cold starless cores using ALMA Band 1. This resolution aggressively drives the output data rate, so modes to throttle the data rate need are provided.

The Phase II Project proposes to build a fraction of the system developed in this Study. Parameters are 8 antennas, dual polarization, with an 8 GHz BBC bandwidth per polarization processed. All hardware for this Project will be *Commercial Off The Shelf* (COTS). Phase II is a variation of the FX approach pioneered by the Collaboration for Astronomy Signal Processing and Electronics Research (CASPER, see Hickish, 2016). FGPA processors are arranged around a fast network switch, which implements the crossbar function needed to transform antenna-based input (F-engines) to the baseline antenna pairs which are cross-correlated to produce the fringe visibilities. In the SMA SWARM system (Primiani et al., 2016), F-engines and X-engines are merged onto a single FPGA processor. For the much more demanding requirements of ALMA this Study found that the F-Engine is still optimally implemented with a Field Programmable Gate Array (FPGA) platform, but the X- engines are moved to a separate processor, per WP2.5, best computed on Graphic Processing Units (GPUs). The overall Phase II system block diagram is shown in figure 9.

3.3 Phase III: *Tentative* implementation and buildout outline

The Phase II Project costs out an 8 antenna 16 GHz (8 GHz dual polarization) system with technologies now—in 2017—on the cusp of availability. The purpose of Phase III is to show that the scale up to a full ALMA installation is *already* feasible using these technologies and algorithms selected for the Project. By December 2021—which is cautiously suggested as the start of full construction and the technology freeze date—a review of current technology would require a fresh look at selection and costing of appropriate FPGA, GPU and high speed network technologies. Assuming specifications stay as here proposed it is anticipated that costs of digital hardware should generally decline.

It should be noted that our expectation is for the Cycle 5 Project to allow us to finalize algorithms and codes, all of which are expected to be readily portable to the latest technology selected for the full build. A caveat applying to this section is that we are only able to credibly cost the technology to set a expected upper bound on the costs of the major components; *we do not attempt to cost the labor to install, commission and properly document the full system.*

Details such as properly installing and cooling the electronics are not discussed, nor is the site decision built in to this top level estimate. Such details would be considered with the assistance of JAO after the completion of the Phase II Project. The schedule is similarly a rough estimate, whose purpose is to show that the architecture and technical approach is in the ball park of feasibility.

Scaling from the project to the full implementation requires scaling by a factor of 4x in bandwidth and 9 times in schedule, or 36-fold in total. So the number of F-engines is 36 x 8 or 288. Each has two network ports, so 576 network ports are required for F-engines. An equal number of ports are required for X-engine GPUs. Further, the antenna data is assumed to be provided in place of the emulator inputs, and the number of switch ports is calculated as 72 antennas x 2 polarizations x 2 sidebands x 2 bandwidth blocs per polarization, also 576 ports needed. These means we need 1,728 100 GigE Ethernet ports for antennas, F-engines and X-engines. To make a cost estimate we assume 32 port switches are $14,997.50 each (Arista DCS-7060CX-32S-F), and we need 54 of these.

The Project Proposal block diagram shows 8 GPU servers, however an analysis of new NVIDIA Tegra GPU technology, not available when the Project was proposed in January, shows that the number of GPU nodes scale up to 48 units (see WP2.5). The four AMTF servers to house F-engines scale up to 144 units. One network cable is costed per port. Table 7 is a rough calculation of the estimated cost of major components only, It excludes necessary and important items such as equipment racks, uninterruptible power supplies (UPS), power wiring, and similar infrastructure, these will be included in the Phase II Project work.

Table 7 ALMA Correlator COTS Equipment Cost Summary

Quantity	Item description	P.U. Cost	Extended Cost
576	Xilinx VCU118 FPGA Eval. Board	$6,995	$4.03M
144	AMTF Server SYS-6028TP	$8,595	$1.23M
48	Trenton Systems Tegra GPU Server	$18,200	$0.873M
54	Arista DCS-7060CX-32S-F	$14,998	$0.809M
1728	Network Cables CAB-Q-Q-100G-5m	$450	$0.778M

Considering the major COTS components in table 7 the cost of equipment only for a 72 antenna 64 GHz total bandwidth correlator and phased array built according to the principals found in this study is just under $8M.

Table 8 is a power budget which uses these same assumptions, and estimates of per unit power consumption to estimate the total power consumption of the full correlator and phased array system scales to about 110 kW, compared to about 140 kW for the current ALMA correlator. Air conditioner ppwer is not included in either number. Given that the new system processes fourfold the bandwidth, and considering other improvements, this power consumption is considered to be in a reasonable ballpark. With true 2021 technology it is anticipated that power consumption would be reduced with the new system.

Figure 10 shows the outline of a Gantt chart documenting a schedule to build this Correlator Phased Array system and commission at ALMA starting in December 2021. A five year period of performance is suggested based on very rough estimates of workflow as presented in the Gantt chart. Following this schedule ALMA would be equipped with a fully commissioned digital back end with integrated VLBI capability by December 2026.

Table 8 Estimated power consumption summary

Quantity	**Item description**	**power p.u. (kW)**	**total power (kW)**
576	Xilinx VCU118 FPGA Eval. Board	0.12	69
144	AMTF Server SYS-6028TP	0.1	14.4
48	Trenton Systems Tegra GPU Server	0.375	18
54	Arista DCS-7060CX-32S-F	0.15	8

At the starting point of Phase III, the proposed Cycle 5 ALMA Development Project is assumed to be completed, and thus it it is expected that the technical risk of this deployment will be extremely low. Implementation then is an exercise in logistics, infrastructure, and proper ALMA compliant project management and documentation, including ICDs, notably for software.

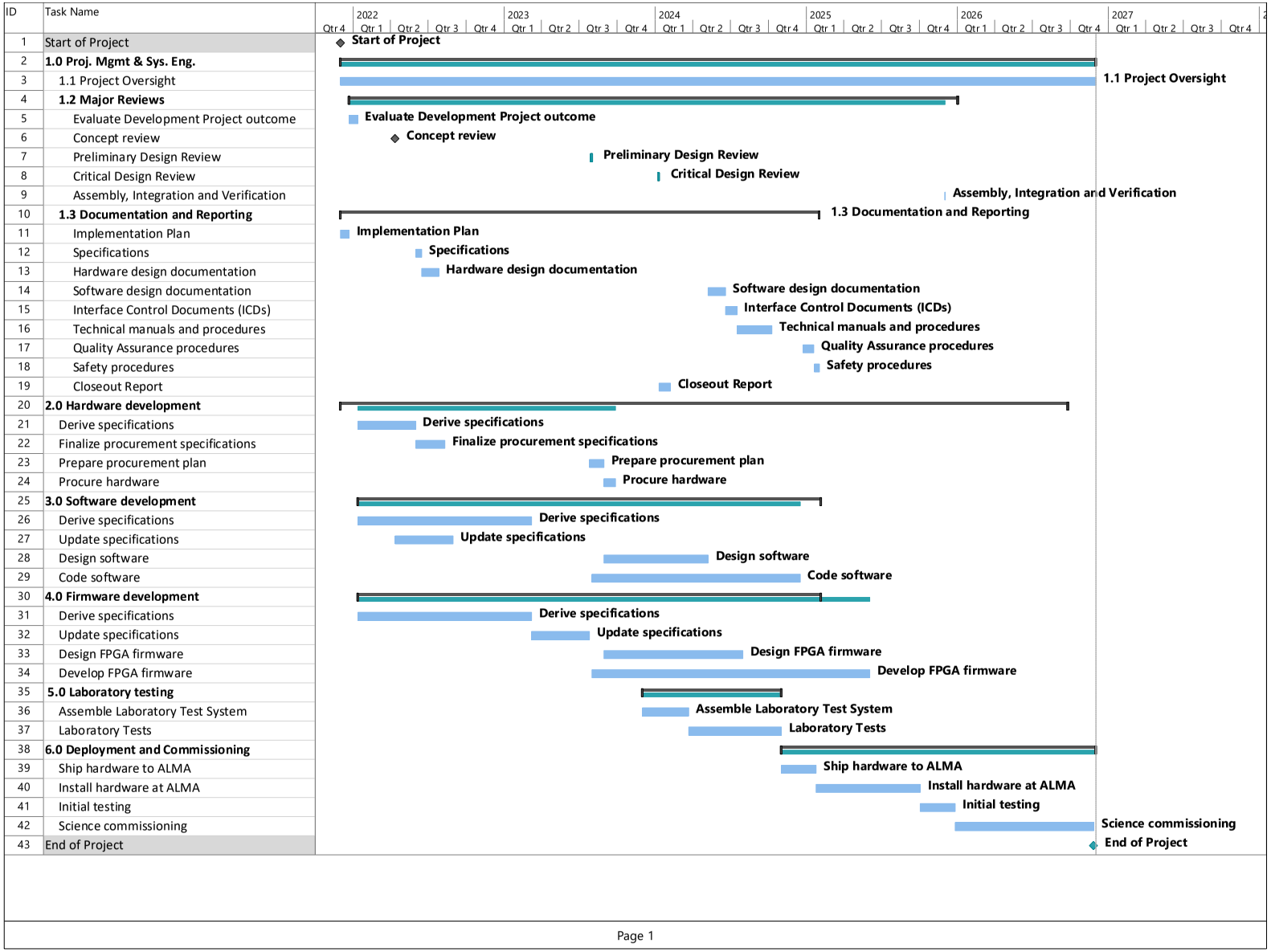


Figure 10 Simplified possible schedule for five year follow-on to ALMA Development Project starting in December 2021, and culminating in December 2026 with the a fully commissioned 64 GHz bandwidth 72 antenna ALMA correlator and phased array. This sample schedule is expected to change as informed by the outcome of Phase II.

4 Summary and closeout

This ALMA Development Study has framed the approach to a design of a next generation correlator and phased array system with which it is possible to upgrade ALMA in a cost-effective manner, quadrupling the instantaneous bandwidth and dramatically increasing spectral resolution.

The Study ran according to the planned schedule starting work on 1 April 2016 and culminating in a closing meeting held as planned in February 2017. The goals have been met and a set of assumed requirements are available in this Study. We welcome comment from the International ALMA community, especially on the assumed requirements. There is a small funding surplus, which we plan to expend on the purchase of some of the specified equipment.

We have proposed for follow on ALMA Development funding for a Project to build a correlator and phased array system supporting eight antennas and 16 GHz total processed bandwidth, split into two polarizations of 8 GHz each. The concluding section of this report shows that it is feasible and cost effective to fit a system designed according to the principals articulated in this report to ALMA and to upgrade the digital systems, in a way assumed to be aligned with other necessary systems, with project completion in late 2026.

The key advances made by this study are summarized in the following bulleted list:

- Our expert international team has studied approaches to correlator design, and framed a risk mitigated three-phase approach to a complete upgrade of the ALMA correlator and phased array, with a projected completion date of December 2026.
- The Cycle 3 Study was completed on time and within budget.
- A follow on Project has been proposed in response to the Cycle 5 call, and is under consideration.
- The proposed approach allows for high performance in respect of bandwidth, spectral resolution, integrated phased array processing, smaller size and power consumption at reasonable and tractable cost.
- It is necessary that the proposed new generation correlator and phased array dovetail with upgrades to receivers, digitization modules, digital transmission system, and software pipelines. Dialog with ALMA management, project managers, and the International ALMA community is recommended. Such dialog, ideally, would culminate in a process of formal system design for the complete next generation ALMA upgrade, considering all subsystems.
- In particular the Specifications of this Study's Work Package 2.1 should be viewed as informed assumptions made by this Study team in consultation with a small group of scientists. The assumptions were necessary to complete the Study, but we do not presume to decide on the specifications of next generation ALMA. We welcome publication and broad feedback on these specifications, and on this Development Study report as a whole.

Expanded documents follow in the appendix, documenting the work breakdown and eight work packages in this study.

References

Bolatto, A.,D, et al.*Pathways to Developing ALMA*, 2015, https://science.nrao.edu/facilities/alma/alma-dev/PathwaystoDevelopingALMA.pdf/view

Bolatto, A.,D, et al.*A Road Map for Developing ALMA*, 2015, https://science.nrao.edu/facilities/alma/science_sustainability/RoadmapforDevelopingALMA.pdf

Clark, M. A., P. C. La Plante, and L. J. Greenhill (2012). *Accelerating Radio Astronomy Cross-Correlation with Graphics Processing Units*. In: The International Journal of High Performance Computing Applications 27.2, pp. 178–192. doi: 10.1177/1094342012444794. arXiv: 1107.4264 [astro-ph.IM]

D'Addario, L.R., Emerson, D.T. 2000, ALMA Memo #331

Escoffier, R. P., Comoretto, G., Webber, J. C., Baudry, A., Broadwell, C. M., Greenberg, J. H., Treacy, R. R., Cais, P., Quertier, B., Camino, P., Bos, A., and , A. W. Gun (2007) *The ALMA correlator* A&A, 462(2):801 810.

Fish, Vincent; Alef, Walter; Anderson, James; Asada, Keiichi; Baudry, Alain; Broderick, Avery; Carilli, Chris; Colomer, Francisco; Conway, John; Dexter, Jason; Doeleman, Sheperd; Eatough, Ralph; Falcke, Heino; Frey, Sándor; Gab/'anyi, Krisztina; G/'alvan-Madrid, Roberto; Gammie, Charles; Giroletti, Marcello; Goddi, Ciriaco; G/'omez, Jose L.; Hada, Kazuhiro; Hecht, Michael; Honma, Mareki; Humphreys, Elizabeth; Impellizzeri, Violette; Johannsen, Tim; Jorstad, Svetlana; Kino, Motoki; K/"ording, Elmar; Kramer, Michael; Krichbaum, Thomas; Kudryavtseva, Nadia; Laing, Robert; Lazio, Joseph; Loeb, Abraham; Lu, Ru-Sen; Maccarone, Thomas; Marscher, Alan; Mart/'i-Vidal, Iv/'an; Martins, Carlos; Matthews, Lynn; Menten, Karl; Miller, Jon; Miller-Jones, James; Mirabel, F/elix; Muller, Sebastien; Nagai, Hiroshi; Nagar, Neil; Nakamura, Masanori; Paragi, Zsolt; Pradel, Nicolas; Psaltis, Dimitrios; Ransom, Scott; Rodr/'iguez, Luis; Rottmann, Helge; Rushton, Anthony; Shen, Zhi-Qiang; Smith, David; Stappers, Benjamin; Takahashi, Rohta; Tarchi, Andrea; Tilanus, Remo; Verbiest, Joris; Vlemmings, Wouter; Walker, R. Craig; Wardle, John; Wiik, Kaj; Zackrisson, Erik; Zensus, J. Anton (2013) *High-Angular-Resolution and High-Sensitivity Science Enabled by Beamformed ALMA* , arXiv: 1309.3519 Hampson, G.,

Brown, A., Neuhold, S., Bunton, J., Macleod, A., Tuthill, J., and Beresford, R. (2013). Askap advancements in beamformer and correlator optical backplane technology. In 2013 US National Committee of URSI National Radio Science Meeting (USNC-URSI NRSM), pages 11.

Hickish, Jack, Zuhra Abdurashidova, Zaki Ali, Kaushal D. Buch, Sandeep C. Chaudhari, Hong Chen, Matthew Dexter, Rachel Domagalski, John Ford, Griffin Foster, David George, Joe Greenberg, Lincoln Greenhill, Adam Isaacson, Homin Jiang, Glenn Jones, Francois Kapp, Henno Kriel, Rich Lacasse, Andrew Lutomirski, David MacMahon, Jason Manley, Andrew Martens, Randy McCullough, Mekhala V. Muley, Wesley New, Aaron Parsons, Daniel C. Price, Rurik Primiani, Jason Ray, Vereesé Van Tonder, Laura Vertatschitsch, Mark Wagner, Jonathan Weintroub, Dan Werthimer, on behalf of the CASPER collaboration 2016, A Decade of Developing Radio-Astronomy Instrumentation using CASPER Open-Source Technology, Journal of Astronomical Instrumentation, **5**, 4

Johnson, S. G. and Frigo, M., *A modified split-radix FFT with fewer arithmetic operations*, IEEE Transactions on Signal Processing, vol. 55, no. 1, pp. 111, 119, 2007

Lee, V.W. et al. *Debunking the 100X GPU vs. CPU myth: an evaluation of throughput computing on CPU and GPU*, ACM SIGARCH Computer Architecture News, 38.3, pp.451-460, 2010.

Manley, J. R. (2015). A scalable packetised radio astronomy imager. PhD thesis, University of Cape Town.

Matsushita, Satoki; Asaki, Yoshiharu; Fomalont, Edward B.; Morita, Koh-Ichiro; Barkats, Denis; Hills, Richard E.; Kawabe, Ryohei; Maud, Luke T.; Nikolic, Bojan; Tilanus, Remo P. J.; Vlahakis, Catherine; Whyborn, Nicholas D. (2017) *ALMA Long Baseline Campaigns: Phase Characteristics of Atmosphere at Long Baselines in the Millimeter and Submillimeter Wavelengths*, PASP, **129** 973

McMahon, P., Langman, A., Werthimer, D., Backer, D., Filiba, T., Manley, J., Parsons, A., and Siemion, A. (2007). CASPER Memo 17: Packetized FX Correlator Architectures. Technical report.

Parsons, Aaron et al. (2008). *A Scalable Correlator Architecture Based on Modular FPGA Hardware, Reuseable Gateware, and Data Packetization*. In: Publications of the Astronomical

Society of the Pacific 120.873, p. 1207

Patel, N. A.; Wilson, R. W.; Primiani, R.; Weintroub, J.; Test, J.; Young, K. 2014, Characterizing the Performance of a High-Speed ADC for the Submillimeter Array Digital Backend, Journal of Astronomical Instrumentation (JAI) 3, 1

Perley, R. 2004, EVLA Memo #64

Pospieszalski, M., Kerr, A., Mangum, J. 2016, submitted to ALMA Memo series

Primiani, Rurik A., Kenneth H.Young, André Young, Nimesh Patel, Robert W. Wilson, Laura Vertatschitsch, Billie B. Chitwood, Ranjani Srinivasan, David MacMahon and Jonathan Weintroub, 2016, *SWARM: A 32 GHz Correlator and VLBI Beamformer for the Submillimeter Array*, Journal of Astronomical Instrumentation, **5**, 4

Vertatschitsch, Laura Rurik Primiani, Andre Young, **Jonathan Weintroub**, Geoffrey B. Crew, Stephen R. McWhirter, Christopher Beaudoin, Sheperd Doeleman, and Lindy Blackburn 2015, *R2DBE: A Wideband Digital Backend for the Event Horizon Telescope*, PASP, **v127**, 958 December 2015

Weintroub, Jonathan, Raffanti, Rick and Primiani, Rurik (2015), *20 gigasample per second analog-to-digital conversion for ultra-wideband radio astronomy*, 26th International Symposium on Space Terahertz Technology, Cambridge, MA, 16-18 March, 2015

S. Williams, A. Waterman and D. Patterson. Roofline: An insightful visual performance model for multicore architectures, Communications of the ACM, vol. 52, no. 4, pp. 65?76, 2009.

Young, A., Primiani, R. A., Weintroub, J., Moran, J. M., Young, K. H., Blackburn, L., Johnson, M. D., and Wilson, R. W., Performance Assessment of an Adaptive Beamformer for the Submillimeter Array, Proceedings of the IEEE International Symposium on Phased Array Systems & Technology, 18-21 October 2016, Waltham, MA

Intel. Intel Xeon Processor E5-2699 v4, 2016. [Online]. Available: http://ark.intel.com/products/91317 . Accessed 1 September 2016.

The NVIDIA CUDA Fast Fourier Transform Library (cuFFT),2016. [On- line]. Available: https://developer.nvidia.com/cufft. Accessed 10 August 2016.

NVIDIA, “NVIDIA NVLink High-Speed Interconnect: Application Performance”, 2014.

[Xilinx, UltraScale+ FPGAs: Product Tables and Product Selection Guide, XMP103 (v1.8), 2016.

Appendices

The work breakdown document which defined the eight work packages is distributed with page breaks across the appendix as a header page for the unabridged report for each of the eight work packages.
Supplementary material in the form of Excel workbooks relevant to the final appendix section covering work package 2.8 can be downloaded at the following URL:
http://library.nrao.edu/public/memos/alma/main/appendix_607_sheets.xls

ALMA Correlator Study Work Breakdown

July 31, 2017

1 Assumptions

1. Correlator architecture will be FX (presently FXF, & XF not favored)
2. Future available bandwidth will be 16 GHz per sideband per polarization
3. Larger bandwidths still can be handled by modularly replicating the correlator
4. Samplers will remain at the antennas with digital data sent over fiber
5. Samplers will digitize 16 GHz bandwidth plus guard-band at 4-bit resolution
6. "single mode"

2 Work breakdown

2.1 Scientific requirements & specifications

Assigned to ***Rupen****, Baudry & Lacasse*

1. Bandwidth 16 GHz per sideband per polarization
2. Continuum resolution, 1 kHz at 30 GHz
3. Dump time, 1 ms
4. other details in memo by assignees

Draft version July 30, 2017
Preprint typeset using LaTeX style AASTeX6 v. 1.0

SCIENTIFIC SPECIFICATIONS AND REQUIREMENTS FOR THE NEXT GENERATION ALMA CORRELATOR

Authors from Study Team and Version 2.0

ABSTRACT

Scientific requirements and specifications for the next generation ALMA correlator are presented and briefly discussed. Interaction with the ALMA Scientific Advisory Committee will enable to reach the stage of consolidated specifications. Our main goal is to provide a coherent set of requirements to guide work by the study group for the next generation ALMA correlator. Ultimately, these requirements will be translated into realistic engineering specifications.

1. INTRODUCTION

Scientific specifications and requirements or strawman requirements for the next generation ALMA correlator are summarized in this document. They are presented in a simple tabular form together with brief comments (see Table 1). We also wish, within the frame of the 'ALMA 2030' documents ('Pathways to developing ALMA' and 'A road map for Developing ALMA') elaborated by the ALMA Scientific Advisory Committee (ASAC), to interact with the ASAC and the ALMA 'Development Working Group' to reach the stage of consolidated requirements. As expected from recent or future technology advances and ASAC recommendations, requirements for the next generation ALMA Correlator will supersede some of the detailed requirements presented in the current ALMA Scientific Specifications and Requirements documents (ALMA-90.00.00.00-001-A-SPE and its revised version including the ACA, ALMA-90.00.00.00-001-B-SPE). Technology or architecture advances impacting the design of a new generation correlator have been initially discussed by a study group prior to and during the kick-off meeting on 'Digital Correlator and Phased Array Architectures for Upgrading ALMA' organized at SAO, Cambridge, MA on 10-11 May, 2016. General technology options, correlator architectures (FX versus FXF or XF) and, to a lesser extent, software correlation were discussed in Cambridge. Our study group proposes to work on an FX correlator design.

We are aware that some of the requirements listed in Table 1 may need more scientific discussions, long-term technical developments and may not be easily translated into engineering specifications. We point out, at the end of the next Section and after Table 1, which requirements would benefit, according to us, from further scientific discussions or technical studies. For example, in addition to funding questions, more discussions need to be conducted for an ALMA Extended Array on the number of additional antennas and maximum baseline.

For reference, Table 3 in Appendix A gives the top level technical specifications of the current ALMA 64-antenna correlator (FXF) and ALMA Correlator Compact Array (FX). A list of abbreviations and acronyms is given in Appendix B.

2. DETAILED REQUIREMENTS

Detailed requirements are listed in Table 1 together with brief comments. Whenever possible we specify in the last column of Table 1 the scientific requirement number as defined in ALMA-90.00.00.00-001-B-SPE or reference a report section where we further discuss a requirement (see Sections 3 to 9). Some requirements must still be considered as strawman requirements and could be more debated in the community and/or checked for their technical feasibility (see end of this Section).

2.1. *Next Generation ALMA Correlator Requirements*

Table 1. Correlator and phased array requirements used in this study

	Parameter	**Requirement**	**Comments**
1	Frequency range	Process digitized IF from all receivers in range $\sim$ 30 – 950 GHz	ALMA Bands 1–10: cf. SCI-90.00.00.00-10-00. Supra THz band, if implemented, would be correlated as well.
2	Number of antennas	72	A minimum of 66 antennas would handle the 50 12m dishes, the 12 ACA antennas, and the 4 antennas of the Total Power array. 72 antennas ($\sim$ 10 percent increase in collecting area) would allow additional antennas for the ALMA Extended Array. More antennas improves the image dynamic range and the effective number of antennas being operated.
3	Maximum baseline	$\sim$ 300 km	The ALMA Extended Array, with maximum baseline set by a combination of surface brightness sensitivity and geographical (antenna placement) considerations. (Requirement consistent with SCI-90.00.00.00-220-00).
4	Instantaneous bandwidth	32 GHz/polarization	Goal is to match bandwidth provided by the receivers, to maximize continuum sensitivity and spectral line search/survey speed, as well as commensal observations and serendipitous discoveries (see §3). For 2SB receivers this would be 16 GHz USB and 16 GHz LSB per polarization.
5	BBC BW	8 GHz	BBC BW represents the bandwidth of each "chunk" presented to the correlator after digitization. This is a science driven parameter (see §4) consistent with future IF range (see §3) and existing, or soon available fast digitizers.
6	Number of BBCs	2/SB and polarization	Required to cover the desired total BW in 8 GHz "chunks." Two BBC's can be stitched together for each sideband if this is required.
7	Input sample format (digitizer) & Correlation sample format	4-bit & 4-bit per sample	4-bit input sample for minimal quantization losses (may require to implement more than 4-bits per digitizer to reach a 4-bit effective number). Native 4-bit correlation without loss of lag-resources.
8	Number of channels (continuum)	Over full BW, per polarization: $\sim 1.0e5(B_{max}/300\,\mathrm{km})$ $(12\mathrm{m}/D)(K/4)$	For wide-field imaging with full pol'n products, assuming 2:1 bandwidth ratios (see §5).
9	Best spectral resolution	0.01 km/s = 1 kHz (ν/30 GHz)	To resolve lines from a cold starless core (see §5). Corresponds to SCI-90.00.00.00-30-00 at 100 GHz.
10	Number of channels (spectral line) per BBC	$\sim 8e6$/BBC	Maximum set by requiring uniform channels at the best spectral resolution over the BBC bandwidth (see subsection 5.2) . More channels increase data rates and archiving problems.

	Parameter	Requirement	Comments
11	Number of configurable subbands	16 independently configurable subbands	Subband position, bandwidth or spectral resolution can be independently set up.
12	Minimum integration time (wide-field imaging)	$\sim 140\,\mathrm{msec}(K_t/3440)(D/12\,\mathrm{m})$ $(300\,\mathrm{km}/B_{max})$	See wide-field imaging in §6.
13	Integration and readout interval	1 msec (auto-correlations) 16 msec (cross-correlations)	Requirement consistent with SCI-90.00.00.00-240-00. Spectral resolution is limited for these dump rates (see §9 for limitations); full spectral resolution available for longer integration times. See §6 for on-the-fly mapping requirement.
14	Polarization products	2- or 4-polarization products	Producing fewer than 4 polarization products is only useful for practical reasons, e.g., to reduce the total data rate or allow more spectral channels in certain architectures. Scientifically one would always like at least 2 polarization products, for sensitivity (although 1 polarizationn would always be possible). SCI-90.00.00.00-310-00.
15	Spectral dynamic range	10,000:1 for weak spectral lines near strong ones 1,000:1 for weak lines atop strong continuum	Identical to SCI-90.00.00.00-70-00. (Image dynamic range, SCI-90.00.00.00-75-00, does not seem to affect the correlator.)
16	Number of subarrays	6	Must be completely independent (no frequency or antenna control restrictions). More is better for science operations and commissioning or maintenance tasks. Current ALMA system requires at least 4 subarrays (SCI-90.00.00.00-390-00) and to operate the ACA 12-m antennas independently of the 7-m array.
17	VLBI	VLBI output sum port for full phased array or 2 subarrays	One subarray could just be one antenna. Two sub-arrays allow simultaneous observations of source and calibrator. High sensitivity may require phasing the whole array. Using two subarrays for VLBI requires appropriate real time control developments. SCI-90.00.00.00-370-00 defines the phased array requirement.
18	Phased-array beams	2–4, with 2^n MHz bandwidths up to the full available bandwidth	This means 2–4 beams *total*, spread over all subarrays. More is better. Drivers are VLBI and pulsars (see §8).
19	Data rate reduction	a- range of dump times b- different spectral/temporal resolution per sub-band c- different spectral/temporal resolution per baseline	See §9 for details and suggested maximum dump rate. Dump time varies from minimum correlator integration time to maximum dump rate. Averaging in time and/or frequency could be before, during, or after correlation.
20	Spectral impulse response	10^{-4} drop by midpoint of adjacent channels; requirement for further out channels is tbd	Driver is weak line (or line wing) adjacent to strong line, e.g., a maser. SCI-90.00.00.00-70-00.

	Parameter	Requirement	Comments
21	Temporal impulse response	< 1 dump time	Driver is very fast dumps – want each dump to be independent.
22	Multi-beam receivers	Adaptability to n x n beams and scalable design	Driver is survey speed. Selecting the number n needs a deep analysis of science requirements and technical possibilities. 3 x 3 pixels require a 9 times bigger correlator (a scalable design helps) and significant backend or LO upgrades. Bandwidth-for-beams (2 beams at 1/2 BW, 4 beams at 1/4 BW, etc.) and antennas-for-beams (fewer antennas allows more beams) trades would be valuable.
23	Switching frequencies	1.5 sec to change tuning within a band 10 msec to switch between frequencies within a band 1.5 sec to change bands (second band ready) 15 min to change bands (second band unready)	SCI-90.00.00.00-40-00, SCI-90.00.00.00-50-00, SCI-90.00.00.00-60-00.
24	Correlator configuration time	< 1.5 sec	Complete configuration should be accomplished in less than 1.5 sec in all circumstances (e.g. moving from continuum mode to line observations). Configuration parameters can be downloaded during an observation and later used during dead times.
25	Receiver sideband separation or suppression	Separate sidebands or suppress one band in the correlator	With DSB receivers the correlator shall have the ability to separate the Walsh-switched products for each baseline. With 2SB receivers and limited rejection in one sideband the correlator shall have the ability to suppress one sideband.
26	Local oscillator offsets removal	Remove local oscillator (LO) offsets applied at the antennas prior to correlation	The correlator shall have the ability to remove the LO offsets as a means of removing spurious signals that may leak after the LO and DC offsets due to signal quantization, or as a means of suppressing one sideband.
27	Water vapor correction	On-line water vapor correction	Optional real time visibility correction for each dump duration.

2.2. *Requirements Benefiting from Further Discussions*

Some requirements would deserve further discussions with the ALMA community and scientific prioritization would be helpful to our study team. This is especially true for *Requirements 2* and *3* (number of antennas and maximum baseline), *4* (instantaneous bandwidth), *8* and *10* (number of channels in continuum and for spectroscopy), *11* (number of subbands) or *22* (multi-beam receivers). We briefly comment these requirements below. *Requirement 2:* A new generation correlator should process the current 66 ALMA antennas but it seems reasonable to anticipate a moderate increase in the number of antennas. We have adopted 6 more antennas (around +10%) for an extended array without further scientific foundation. *Requirement 3:* We have adopted $\sim$ 300 km maximum baseline as a promising and realistic option for a good science return. *Requirement 4:* A total bandwidth of 32 GHz per polarization (4x the current ALMA system) seems desirable to match the bandwidth of future 2SB receivers providing more than 8 GHz IF even though achieving 16 GHz per sideband will not be easy and may degrade the receiver system temperature (see §3). Independently of technical considerations it would be important to know if 32 GHz/polar is a solid science goal. *Requirements 8, 10:* The number of channels across the full bandwidth (continuum) and the spectral resolution to achieve for spectroscopy over a spectral window, a BBC and/or the full bandwidth are driven by science considerations for a given category of objects. However, a new generation correlator may generate 'too many' channels and it would

be useful to know if channel averaging may become of common use. This question is also related to *Requirement 11* and we have adopted 16 configurable subbands as a reasonable goal without much science justification. *Requirement 22:* Multi-beam receivers or multi-pixel designs could heavily impact the correlator design. Clarifying the scientific priority and which ALMA bands are concerned would help to consolidate this requirement. Finally, a reasonable long-term extrapolation of the current and projected ALMA data rates based on the current and future ALMA science would help us to clarify important parameters such as the maximum number of channels to be recorded.

We believe that inputs from, and interaction with other study teams that work for developing ALMA are equally important. This is the case in particular for *Requirement 4* where we need to know if 32 GHz/polar is achievable with good noise performance, *Requirement 10* which is much related to data rate limitations and data storage, and for *Requirement 22* in order to define a reasonable number of pixels in the focal plane to be processed by the correlator.

Several requirements also have implications on the software/firmware development plan. We do not address these questions here but note that they may especially concern *Requirement 11* (multi-windows), *Requirement 16* (sub-arraying), *Requirements 17, 18* (VLBI, phased array beams) or again *Requirement 22* (multi-beams).

3. INSTANTANEOUS BANDWIDTH

Instantaneous bandwidth is the bandwidth delivered by the front-end receiver IF range. It is the bandwidth brought down from the sky that we wish to correlate. Going from the current 4 to 8 GHz per sideband (SB) and per polarization (in the 4–12 GHz IF range) to 4 times more bandwidth, i.e. 32 GHz per polarization (16 GHz USB and 16 GHz LSB for 2SB receivers), is being discussed within the ALMA community and seems achievable with present day receiver technology. The current 4–12 GHz IF range is optimum for the 8 GHz IF bandwidth selected for ALMA (see Pospieszalski et al. 2016) but this IF bandwidth could soon be expanded to 10 or 12 GHz, and 16 GHz per SB and polarization seems an achievable goal in future 2SB receivers. However, expanding the 8 GHz IF bandwidth tends to increase the noise temperature of the IF stage (Pospieszalski et al. 2016) and, hence, the receiver system temperature because of the mixer conversion loss inherent in SIS mixers. The advantage gained for continuum observations in going to wider bandwidths, as long as it is not balanced by higher system temperatures, may be lost for some categories of spectral line observations where a contiguous broad bandwidth is not a major requirement. However, spectral line search or survey speeds would be improved with wider bandwidths. (We further note that it is unclear at this stage if going to 2x32 GHz per polarization could be achieved in a distant future.)

4. BASE BAND CHANNEL (BBC)

Base Band Channel (BBC) is used here to mean the "chunk" of frequencies presented to the correlator for processing. BBC shown in Table 1 must be understood as a science-based requirement to detect wide spectral lines (up to about 2000 km/s should be analyzed for e.g. galaxy clusters) as well as narrow lines covering a continuous frequency range (to detect more lines in a wider instantaneous bandwidth for e.g. better spectroscopic identification). The broadest velocity range to be analyzed is about 2000 km/s (compact galaxy clusters) and this requires about 6 GHz bandwidth at the highest ALMA frequency. We adopt here 8 GHz so as to cover 'all' science cases. BBC could also be defined from a technical view point. Its maximum value depends then on the maximum IF range delivered by the front-end receivers and the instantaneous bandwidth that can be digitized by an analog-to-digital converter (ADC). The BBC bandwidth is meant here to be the usable bandwidth which implies a digitizer clock rate higher than 16 GHz for 8 GHz BBC. (ADC technology is continuously evolving and digitizing 8 GHz with 4 bits in one go should be attainable relatively soon.)

5. SPECTRAL CHANNELIZATION

5.1. *Wide-field Continuum Imaging*

For continuum work the required spectral resolution is set by the desire to limit chromatic aberration (bandwidth smearing), which limits the field-of-view of the interferometer.[1] RFI considerations may lead to more stringent requirements at some frequencies. In EVLA Memo 64 (Perley 2004), it is argued that wide-field imaging requires $\nu/\Delta\nu = KB_{max}/D$, with ν the observing frequency, $\Delta\nu$ the channel width, B_{max} the maximum baseline, D the antenna diameter, and K a constant reflecting the imaging requirements. Table 2 lists the values of K suggested by Perley (2004), as well as the effective K used by the current requirements for SKA1-MID. The choice of a value for K is debatable at ALMA frequencies. We adopt $K = 4$ for the requirement in Table 1 as a reasonable assumption and

[1] One also wishes to measure the spectral behavior of the continuum emission, but radio continuum emission generally changes smoothly with frequency, and the frequency resolution necessary to avoid bandwidth smearing losses more than suffices to track these slow variations.

remind that the current correlator provides, in the continuum mode (time division mode), up to 256 channels per 2 GHz baseband chunk for a maximum baseline of 16 km.

Table 2. Spectral resolution constant K for various imaging cases.

K	Condition
1	Targeted imaging (stretch by 1 synth. beam at first null)
2	Targeted imaging in the presence of RFI (allows for Hanning smoothing)
4	Distortion-free imaging of full primary beam (10% loss at first null)
8	Full primary-beam imaging in the presence of RFI
9	SKA1-MID (2% loss at first null)

If the frequency channels are equally (linearly) spaced over frequencies from ν_l to ν_u, the required number of channels is $(\nu_u - \nu_l)/\Delta\nu$. Defining the bandwidth ratio BWR as $BWR = \nu_u/\nu_l$, and inserting the previous expression for $\nu/\delta\nu$ at the more stringent (lower) frequency ν_l,

$$N_{chan} = (BWR - 1) * K * B_{max}/D$$

The *maximum* number of channels needed per BBC corresponds to the lowest observing frequency, since a BBC of fixed bandwidth has the largest bandwidth *ratio* at the lowest sky frequency.

5.2. *Spectral Line Observations*

The spectral resolution for spectral line observations is set by the required velocity resolution at the lowest observing frequency, as given in Table 1. The current science requirement is 0.01 km/s at 100 GHz (cf. SCI-90.00.00.00-30-00), corresponding to 3.3 kHz. The latter need is nearly met by the current baseline correlator which offers 3.8 kHz across 31.25 MHz bandwidth in one single polarization (this mode is called Tunable Filter Bank half-band mode). The velocity resolution of 0.01 km/s needed to resolve self-absorption line profiles from infalling protostellar envelopes also applies to lines observed around 30 GHz, the lowest ALMA frequency, and this now requires 1 kHz resolution which is about 4 times better than the best resolution attainable with the current baseline correlator. We then need about 8 million channels across one BBC (or 2^{23}-point FFT per BBC). Note that with uniform channel spacing at the best resolution over each BBC there is no need to trade resolution against bandwidth as with XF-architecture correlators. Trading sensitivity against resolution is not required either because 4-bit correlation and 4-bit digitization minimize the quantization losses (a combined 2% losses is achieved).

In practice, many science projects will not need 8 million channels per BBC (8 GHz). It is likely that the highest spectral resolution will not be useful across a velocity spread of $\sim$ 100 km/s or a few 100 km/s (for e.g. 'energetic' outflows) which would require a maximum of a few tens of thousands channels. Perhaps the most demanding science case is that of narrow maser lines to be searched for blindly across a broad velocity range, but typically we would need around 50,000 channels over 500 km/s. Channel averaging or channel 'windowing' will thus be needed in most cases.

5.3. *Configurable Subbands*

To maximize flexibility it is desirable to provide several configurable subbands. Based on practical experience with ALMA and with other mm/submm interferometers a minimum of 4 spectral windows is required or used. For a total bandwidth 4 times larger than the ALMA current system, a total of 16 subbands may be desirable. Fully independent subbands implies that subband frequency center and bandwidth, number of spectral channels per bandwidth and

integration time can be selected as desired for each subband. This requirement heavily relies on the adopted correlator architecture, implementation details and software or firmware complexity. Highly versatile tunable filter banks with many spectral channels are required to provide such a high flexibility.

The range of subband bandwidths is set by the range of science cases and could vary from a few km/s for the coldest clouds in the Galaxy to thousands of km/s for extragalactic studies. Therefore, the required spectral resolution over the total subband bandwidth heavily depends on the science case.

6. MINIMUM INTEGRATION TIME

6.1. *Wide-field Imaging*

The requirement for wide-field imaging is to avoid significant time-averaging losses. The losses are due to a reduction of the fringe visibility amplitude and vary as the sinc function of the product "time integration x fringe frequency". This is generally expressed as a limit on the integration time:

$$\Delta t \leq K_t D/B_{max}$$

wher K_t depends on the degree of loss deemed acceptable and the Earth angular rotation. Note that this is the time resolution required at the imaging stage, i.e., after the data leave the correlator. Perley (2004) suggests that the limit should correspond to 10% drop in the visibilities with the longest baseline at the first null of the antenna pattern, giving $K_t = 3440$ (in seconds of time). SKA1-MID requires at most a 2% drop, giving $K_t = 1200$, a factor 2.9 more stringent that Perley (2004).

As with the spectral channels the raw number is extremely high. Baseline-dependent averaging could help, especially given the small number of 'outlier' antennas which set the requirement.

6.2. *On-the-fly Mapping*

On-the-fly mapping requires dumping the data often enough that one knows the antenna pointing for each dump to a fraction of the size of the antenna primary beam, with that fraction set by the required accuracy of the final deconvolution. The required integration time thus depends linearly on the speed of antenna motion, and inversely on the antenna diameter and the observing frequency; there is no dependence on baseline length or spectral resolution. On the other hand, on-the-fly *fringe tracking* at a constant sky position results in a coherence loss which varies as the total scan time and is independent of the observing frequency (D'addario and Emerson, 2000). (The total number of individual antenna beams covered during mapping depends on the scanning speed and for a maximum acceptable loss is limited by antenna speed or correlator dump rate.)

7. SUBARRAYS

There are many scientific and operational drivers for subarrays defined as assembly of antennas (or a single antenna) in which the antenna control and observing frequency are completely independent. It is probably not required to go into details here. With four 12-m antenna subarrays being implemented in the current system and the requirement to operate the ACA 7-m array independently of the 12-m antenna array a minimum of 5 subarrays is required. Even more subarrays would be better, for example 6 or 7 depending on scientific and operational considerations. We have adopted 6 subarrays as a good objective.

8. PHASED–ARRAY BEAMS

The primary scientific driver for phased-array beams is VLBI, which requires in the continuum mode (non spectroscopy-type observations) the maximum possible total bandwidth (for sensitivity), as well as consistent bandwidths for each output stream. Note that bandwidth for VLBI array in the continuum mode might be limited by ALMA, as it might be easier to implement wider bandwidths on single dishes. There should be at least two phased-array beams, to allow in-beam calibration (VLBI calibrators) as well as the possibility of observing in two bands simultaneously (in different subarrays). Observations of pulsars and other highly variable sources in the ALMA low frequency receiver bands would also benefit from phased-array beams.

It is interesting to note that recording 32 GHz per polarization generates a 2-bit sampled data rate of 256 Gb/s per VLBI sum, a number which must be doubled if 2 subarrays are desired for e.g. calibration or relative astrometry. This will require VLBI backend developments at the stations which plan to observe in conjunction with ALMA. Development of the current VLBI correlators will also be needed to accommodate the high data rates.

9. DATA RATE

The number of visibilities per integration N_{vis} and baseband is:

$$N_{vis} = (N_{ant} * (N_{ant} - 1)/2 \times N_{pp} + N_{ant} \times N_{pp,ac})\, N_{ch}$$

with N_{ant} the number of antennas, N_{pp} the number of polarization products for the cross-correlations, $N_{pp,ac}$ the number of polarization products for the auto-correlations, and N_{ch} the number of spectral channels. Baseband means here the "chunk" of frequencies which is actually being digitized; this is a sub-band of BBC BW defined earlier if several digitizers are required to cover one BBC. The actual data dump rate at which visibilities are produced per baseband, N_{vis}/dt, decreases as the integration time increases and the visibility rate increases as the size of each visibility. ALMA currently offers 2 bytes per visibility (real or imaginary part) to remain within the current maximum allowed data rate while the proposed SKA1 correlator output would accommodate ~ 10 bytes per visibility.

The simplest approach from the correlator software/firmware point-of-view would be to provide the full spectral resolution for the entire available bandwidth. With 32 GHz per polarization at 1 kHz resolution, for 72 antennas, with 16 msec integrations, and assuming 8 bytes/visibility, the data rate out of the correlator would be ~ 84 TB/sec.This is highly challenging and would imply difficult operational specifications for a new generation correlator. However, this rate would go down to 172 GB/s (or 86 GB/s for 4 bytes per visibility) if, for example, we would restrict the recorded channels to 65536 channels (a still large power-of-two channel number, consistent with science cases mentioned in subsection on 'Spectral Line Observations') or would perform channel averaging after the correlation stage. We further note that these very high rates are well above 1 GB/s the peak data rate passed by the current 64-antenna correlator to the Correlator Data Processor (CDP) and well above ~ 60 MB/s, the maximum ALMA post-CDP data rate supported by ALMA. (Current ALMA data capture limitation, compared to 1GB/s, is due to limited network and connections speed.) We suggest that a peak data rate of 100 GB/s for all BBCs and all antennas is a reasonable goal (still to be debated within the ALMA community) for the next generation ALMA correlator. As for any other big correlator, given the data maximum dump rate offered to the observers, one must select for each science project the best trade-off among number of antennas, polarization products, number of spectral channels and time resolution. The number of basebands to be processed may also have to be traded against the supported output rate once it is firmly known.

ACKNOWLEDGEMENTS

Several participants in the ALMA Correlator/Phased Array Development Study group provided initial inputs to this document during the kickoff meeting of 10, 11 May 2016. We are also grateful to people outside the study group who provided useful insight into future requirements.

REFERENCES

D'Addario, L.R., Emerson, D.T. 2000, ALMA Memo #331

Perley, R. 2004, EVLA Memo #64

Pospieszalski, M., Kerr, A., Mangum, J. 2016, submitted to ALMA Memo series

APPENDIX

A. APPENDIX MATERIAL

Table A1. Top level specifications of the current ALMA correlators

	Parameter	**Baseline Correlator (FXF)**	**ACA Correlator (FX)**
1	Antennas	64	12
2	Baseband (BB) per antenna	8 x 2 GHz	8 x 2 GHz
3	Input sample format	3-bit, 8-level at 4 GSps	3-bit, 8-level at 4 GSps
4	Correlation sample format	2-bit, 4-level or 4-bit, 16-level	4-bit, 16-level
5	Maximum baseline delay range	up to 600 km	15 km
6	Spectral points per BB	up to 8192 (FDM mode, 3.8 kHz max resolution)	FX design (matches FDM max resolution)
7	Polarization products	1, 2 or 4	1, 2 or 4
8	Temporal integration	1 ms (auto-correlation) 16 ms (cross-correlation)	1 ms (auto-correlation) 16 ms (cross-correlation)

Table 2 continued on next page

B. APPENDIX MATERIAL

Table B2. Abbreviations and Acronyms

	ACA	Atacama Compact Array
	ADC	Analog to Digital Conversion
	ALMA	Atacama Large Millimeter/sub-millimeter Array
	ASAC	ALMA Scientific Advisory Committee
	BBC	Base Band Channel
	Bmax	The maximum baseline in the array
	BW	Bandwidth
	CDP	ALMA post-correlation Correlator Data Processor
	D	The diameter of an individual antenna in the array
	FFT	Fast Fourier Transform
	FX	A correlator architecture where a Fourier transform precedes the correlation
	FXF	A correlator architecture where the initial stage of processing is a digital filter bank. A correlation stage and Fourier transform stage follow.
	IF	Intermediate frequency range delivered by one antenna receiver
	K	A constant reflecting imaging requirements
	Kt	Time-averaging loss parameter counted in seconds of time
	LO	Local oscillator
	LSB & USB	Receiver lower (upper) sideband
	2SB	Two-sideband receiver
	SKA	Square Kilometer Array
	VLBI	Very Long Baseline Interferometry
	XF	A correlator architecture where a Fourier transform follows the correlation stage

2.2 Identify DSP F-engine platform

*Assigned to **A. Young**, Hickish, Escoffier, Primiani, Saez, & Herrera*

1. ASIC vs FPGA vs GPU vs CPU
2. Power, heat, and cooling
3. COTS vs custom designed
4. Ease of interfacing
5. Single-unit compute capabilities, e.g. TFlops, Slices, DSPs, etc.
6. Single-unit data bandwidth, input/ouput in Gbps
7. Single-unit memory availability, in- and off-board in GB
8. Availability and usability of design software and libraries
9. Single-unit price and life-cycle cost

Digital Correlator and Phased Array Architectures for Upgrading ALMA

WP2.2: Identify F-Engine Platform

May 9, 2017

1 Introduction

This document presents a comparison of different platforms for implementation of the F-engine as part of the wider study of developing a next-generation ALMA correlator and phased array system. A recommendation is made as to which platform is expected to yield the best performance in terms of the figures-of-merit (FoM) considered.

2 F-Engine Baseline Requirements

A set of baseline requirements for the F-engine is derived based on the detailed requirements in [1]. The relevant items from that document are listed below in Table 1 with additional comments. The derived F-engine requirements are shown in Table 2. For each platform we compute the costs associated with the real-time F-engine processing for one baseband channel, or BBC (with consideration of whether a single compute unit is able to process multiple BBCs, a single BBC, or a fraction of a BBC) and assume that overall costs will scale linearly with the number of BBCs required for the full array.

#	**Parameter**	**Requirement**	**Comments**
3	Maximum baseline	300 km	Impacts on buffer memory requirements for coarse-delay correction. Table 3 in [1] specifies 600 km delay range for current ALMA correlator, i.e. 300 km in either direction.
5	BBC BW	8 GHz	Impacts on throughput requirements.
7	Sample resolution	4-bit & 4-bit	Impacts on I/O bandwidth requirements.
9	Spectral resolution	1 kHz	Impacts on FFT size, combined with 8 GHz BBC gives close to power-of-two spectral channels.

Table 1: Parameters used to derive baseline F-engine requirements.

Parameter	**Requirement**	**Comments**
FFT size	$2^{24} = 8$ Mi	Assuming N-point FFT of real-valued sequence implemented as an $N/2$-point FFT of complex-valued sequence.
Throughput	1.049 ms / FFT	One FFT computed for every 2^{25} samples at a rate of 16 GSa/s.
I/O bandwidth	64 Gb/s	4-bit $\times$ 16 GSa/s at input and output.
Coarse-delay buffer	125000 Kib	(2$\times$300 km/c) $\times$ (4-bit$\times$16 GSa/s) / 1024.

Table 2: F-engine requirements per digitized baseband channel.

3 Overview of Various Platforms

3.1 FPGA

1This section considers various aspects of F-engine implementation on FPGA.

3.1.1 DSP Cost Model

Later we will extrapolate the resource requirements of the ngALMA channeliser. First, however, we state some simple formulae which may me used to calculate the resource requirements of different parts of the channeliser.

FFT cost, in DSP slices, for the CASPER-supplied real-input radix-2 transform with N points and M parallel inputs is given by:

$$4 \times \left[\frac{M}{4} \log_2 (N/M) + \frac{M}{2} \log_2 (M) \right] . \quad (1)$$

The leading factor of 4 represents the 4 DSP slices required for a complex multiply. The first term then represents an M individual N/M point serial FFTs (including a 4x efficiency improvement because of the underlying biplex, complex, core). The second term represents a single M-input parallel FFT, including a 2x efficiency improvement due the underlying complex core.

It should be noted that there is significant room for optimisation, for example by choosing a radix-4 FFT, or taking advantage of the ability to perform low-precision complex multiplies with fewer than 4 DSP slices. In general, it should be remembered that DSP resource use is a *very weak* function of FFT size.

PFB-FIR cost, in DSP slices, for the FIR front-end of a polyphase filterbank with t taps and M real-valued, parallel inputs is given by:

$$M \times t . \quad (2)$$

Note that this is not a function of number of channels, and in general is an insignificant cost for large filterbanks.

We may assume that other operations (delay-tracking, equalisation, etc.) have an insignificant cost relative to the channeliser. Thus, the number of DSPs required in an FPGA-based F-engine is given by Equation 1.

Since M is the ratio of ADC sampling clock to FPGA clock, we may estimate it given the 32 GHz sampling specified by the ngALMA specs. A 500 MHz FPGA clock is probably reasonable, given the timeline of the project, yielding $M = \frac{16000}{500} = 32$. The number of DSP slices required for the complete 2^{24}-point channeliser is thus, by Equation 1, 1000. This is comfortably achievable in even current-generation FPGAs.

3.1.2 RAM Cost Model

Coarse Delay RAM requirements, as given in Table 2, is approximately 128 MBytes. This is above that available in modern FPGA chips, and we immediately assert that this delay buffer will need to be implemented in off-chip RAM resources. Given that these RAM blocks are likely to be very deep, they may also be used as a short-term transient buffer.

FFT cost, in RAM usage, scales with the length, N of an FFT. Very approximately, the quantity of RAM memory required for data storage in an FFT is at least:

$$N \times b_{\mathrm{FFT}}\,, \tag{3}$$

bits, where b_{FFT} is the data word width within the FFT. This represents only part of the RAM required. If FFT coefficients ("twiddle factors") are not generated in real time, these also need to be stored. This storage is of the same order as the data storage.

PFB cost, in RAM usage, scales with both the length and number of taps in the PFB's FIR front-end filter. This cost is:

$$Ntb_{\mathrm{ADC}}\,, \tag{4}$$

bits, where b_{ADC} is the ADC data word width.

For the ngVLA specifications, with $t = 4$ and $b_{\mathrm{FFT}} = 16$, the total RAM required is approximately 134 MBytes. This is several times larger than that available with the largest chips currently available. Thus we conclude that a channelizer may only be implemented on an FPGA chip if it is implemented in two distinct stages, with an off-chip transpose separating them.

3.1.3 Extrapolating from SWARM

We will use SWARM [2] as a guide to estimate the FPGA requirements to implement the F-engine considered here. Specifically, the FPGA resources utilized to implement one F-engine (4-tap polyphase filter + 32768-point real-valued FFT) is shown in Table 3.

DSP48E	Slice LUTs	Slice Reg.	BRAM (Kib)	Slices
336	66546	74637	8352	21059

Table 3: FPGA resource utilization of single F-engine in SWARM. Data from [2, Table 2].

We note that the number of DSP slices used is approximately consistent with Equation 1 which yields 272 DSPs, given the parameters of the SWARM system ($M = 16, N = 15$). Since the non-RAM resources used for FFT calculation

dominate over that for the preceding polyphase filter we will simply apply the following scaling to the results in Table 3,

$$R_{\mathrm{ngALMA}} = \frac{D_{\mathrm{ngALMA}} \log(N_{\mathrm{ngALMA}} D_{\mathrm{ngALMA}})}{D_{\mathrm{SWARM}} \log(N_{\mathrm{SWARM}} D_{\mathrm{SWARM}})} R_{\mathrm{SWARM}}. \tag{5a}$$

Here R is the total usage of a non-RAM resource, D is the demux factor, N is the real-valued FFT size, and the subscripts are used to differentiate between SWARM and ngALMA.

For RAM use, the PFB FIR-frontend dominates, and we use the following relationship:

$$R^{\mathrm{RAM}}_{\mathrm{ngALMA}} = \frac{N_{\mathrm{ngALMA}}}{N_{\mathrm{SWARM}}} R^{\mathrm{RAM}}_{\mathrm{SWARM}}. \tag{5b}$$

For an FPGA clocked at 250 MHz, processing data sampled at 32 GSa/s requires a demux factor D_{ngALMA}=64; and for a clock rate of 500 MHz the required demux is D_{ngALMA}=32. The result of scaling the SWARM utilization in Table 3 according to (5) is shown in Table 4. In terms of logic / DSP resources the F-engine should fit fairly easily within a single unit Xilinx Ultrascale+ VU13P, however the required memory far exceeds its capability (even when considering an additional 360000 Kib of UltraRAM and 49500 Kib of distributed RAM).

Clock	**DSP48E**	**Slice LUTs**	**Slice Reg.**	**BRAM (Kib)**	**Slices**
250 MHz	2122	420291	471392	4276224	133004
500 MHz	1028	203140	227839	4276224	64285
VU13P	12288	1728000	3456000	94500	432000
250 MHz	17.27%	24.32%	13.64%	4500 %	30.79%
500 MHz	8.35%	11.76%	6.59%	4500 %	14.88%

Table 4: Projected FPGA resource utilization for ngALMA by scaling SWARM according to (5) for two clock rates (demux factors). Lower section compares requirements to resources available in Xilinx Ultrascale+ VU13P [6] .

3.1.4 Input / Output

Most of the Xilinx Ultrascale+ devices feature several GTY transceivers each rated at 32.75 Gb/s. Only four are needed per processed BBC to achieve the required 64 Gbps data rate in and out of the device; even some of the lower-performance (in terms of high-speed I/O) Ultrascale+ devices have at least eight GTYs, whereas other devices have up to 128 GTYs. I/O is not likely to be the limiting factor on this platform.

3.1.5 A Strawman FPGA F-Engine

We learned that RAM is the limiting factor in an FPGA F-Engine implementation. We therefore propose the following strawman implementation of a multi-stage channeliser, with the first stage based on an FPGA platform Figure 1.

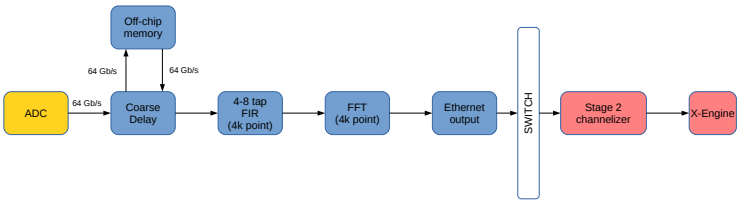


Figure 1: A *very* high-level F-Engine block diagram.

Such a channeliser, with only 4k channels generated per FPGA, would need only 1000 DSPs, and less than 1 Mib of RAM. If extra off-chip memory resources are available, the second-stage of a 8M channel filterbank could be implemented on the same chip following an off-chip transpose. In this case, the total on-chip would approximately double.

In either case, limited DSP is required. Efficiencies can thus either be gained by purchasing cheaper chips (Kintex Ultrascale chips are already available and broadly suitable, for $2k) or by processing multiple antenna inputs on each FPGA node.

3.1.6 Cost Per Unit

Single-Stage Approach Here we assume that the requirements on memory for implementation of a single-stage 8M-point channelizer could be met with some off-chip storage solution. Pricing on the Xilinx Ultrascale+ FPGAs is not readily available; instead we estimate the cost per unit using available pricing for devices in the Ultrascale family. The cost for these devices seem to lie between 20 k$ for XCVU125-1 and 55 k$ for XCVU440-2 (single-unit quantity, assuming no bulk discount). The logic resources within XCVU125 are somewhat below, whereas that in the XCVU440[1] are far above the requirements in Table 4. For comparison with other platforms we will assume an additional 10 k$[2] in cost for a hosting platform per FPGA, which places the cost per BBC in the range 30–65 k$.

Multi-Stage Approach Here we assume a multi-stage implementation as in Figure 1. Relatively cheap FPGAs (e.g. Kintex Ultrascale KU040-2 at ∼2 k$ per unit) should suffice to implement the FIR and FFT blocks in the 4k-point first stage. Assuming similar requirements for the 2k-point second stage (to achieve 8M-point in total), and adding again 10 k$ for the hosting platform, the total cost per BBC is around 14 k$. This estimate excludes the network that connects the two F-engine stages since it essentially replaces the corner-turn between the F- and X-engines that would be needed in the case of a single-stage F-engine system design.

[1] In terms of logic cells the device even outperforms the VU13P, however it does not have GTY I/O, or nearly as many DSP cores and RAM (block + ultra) as the VU13P.

[2] More or less the cost of equivalent hosting platform for Virtex-7 in the SKARAB.

For comparison to other platforms we will use the multi-stage F-engine implementation, i.e. 14 k$ per BBC.

3.1.7 Power Efficiency

Using Xilinx Power Estimator, the power consumption of the F-engine implementation in Table 4 (clocked at 250 MHz, assuming 50% toggle rate and default routing complexity) was estimated to be around 20 W.[3]

3.2 ASIC

We met with iSine to determine an estimate of the relative performance of ASICs compared to other platforms. Some general notes based on their expertise and which may be of use:

1. For 65 nm and smaller the improvement in power-vs-speed improves only by around 20% for every doubling in density (used to be around 50%).
2. The process node they would recommend is typically one point behind the current smallest point used in FPGA. Xilinx Ultrascal+ is currently at 16 nm, so for present day development on ASIC iSine would likely recommend 28/22 nm.
3. Implementation on ASIC could likely run at a clock speed around 1 GHz. This may reduce the demux factor over an FPGA clocked at say 500 MHz (at best) or 250 MHz (maybe more realistic).
4. Including high-speed serial / ethernet interfaces in the design may require purchasing intellectual property (IP) at a high cost.

3.2.1 Cost Per Unit

The cost of implementation on ASIC is expected to be dominated by NRE costs. Figure 2 shows a 2011 estimate of the design cost by process node. Ignoring software design (required on any platform) and yield ramp-up, the design and mask costs run into the tens of millions for even 65 nm. Mask costs alone seem to range between 3 M$ (65 nm) and 12 M$ (22 nm). Assuming one BBC processed per chip and a total of 80×2×2 (number of antennas, polarizations, sidebands) BBCs for the entire correlator, the mask costs per unit alone is between 9 k$ and 38 k$. Doubling this number to obtain a very conservative design cost estimate, and including a host platform cost of about 10 k$ similar to that for FPGA, implementation on ASIC seems to become prohibitively costly at 37–124 k$ per BBC depending on the process node.

[3] For comparison, the equivalent power consumption estimate for a single F-engine in SWARM is around 12 W, so the ngALMA F-engine consumes roughly 1.7 times as much power. This is more-or-less consistent with an estimate based on the increase in logic (about 6.3×), decrease in clock speed (0.87×), and an average power-vs-speed improvement of 33% for each doubling in density from 45 nm (Virtex-6 used in SWARM) to 16 nm (VU13P), $6.3 \times 0.67^3 \times 0.87 \approx 1.6$.

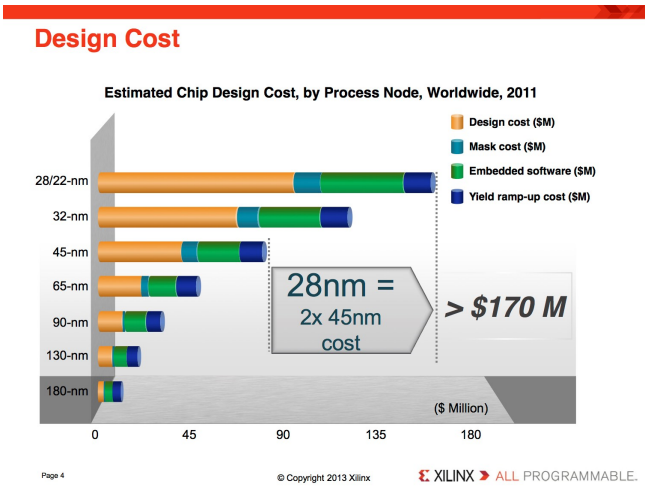


Figure 2: Estimated ASIC design cost by process node.

3.2.2 Power Efficiency

To estimate power efficiency of implementation on an ASIC platform we start off with scaling an FPGA implementation with an increased clock rate and reduced demux factor. Since the required arithmetic units for FFT calculation scales with demux as $D \log D$, a faster clock speed in ASIC could potentially scale the implementation size as,

$$\frac{D_A \log D_A}{D_F \log D_F}, \tag{6}$$

where D_A and D_F are the demux factors for ASIC and FPGA implementations, respectively. Since a sample rate of 16 GSa/s requires D_A=16 for a 1 GHz clock, the implementation size on the ASIC may be expected to be around ∼17% (for D_F=64, FPGA clocked at 250 MHz) of that on the FPGA due to the reduction in demux factor alone. Assuming a linear scaling in power consumption versus frequency the FPGA-equivalent power consumption for ASIC implementation comes out at $0.17 \times 4 \times 20$ W = 13.6 W. Accounting for ASIC implementation on one process node behind current state-of-the-art in FPGA, the ASIC power consumption increases by ∼25%, to ∼17 W.

Since ASIC implementation is typically much more power efficient than FPGA, we will use 17 W as a *very* safe upper bound estimate of power consumption per BBC on this platform.

3.3 GPU

This section discusses various aspects of F-engine implementation on GPU in some detail.

3.3.1 Compute performance

It is assumed here that the computational cost of computing the DFT of a sequence of N complex-valued elements via a simple implementation of a radix-2 FFT is $5N \log_2 N$ floating-point operations [3][4]. Since such a computation is required every 1.049 ms (see Table 2), the compute performance that needs to be achieved for the FFT implemented on GPU is,

$$C = \frac{5N \log_2 N}{2N/R} = \frac{5}{2} R \log_2 N = 920 \text{ GFLOPS}, \tag{7}$$

where $R = 16$ GSa/s is the sample rate and $N = 2^{23}$. The required compute performance is much lower even than the theoretical peak performance of NVIDIA GPUs that are already a few generations old, e.g. the NVIDIA Tesla K40 released in 2013 has a peak performance of 4291 SP-GLFOPS (single-precision giga-floating-point operations per second), whereas for the NVIDIA Tesla P100 released in 2016 the peak performance is 9519 SP-GFLOPS (or with newly added half-precision capability, 19038 HP-GLFOPS). However, due to relatively low operational intensity (number of FLOPs per byte read from memory) of the FFT algorithm the attained performance for large N is typically memory bandwidth limited[5] [4]. Given the memory bandwidth B, peak compute performance C_p and operational intensity I, the attained compute performance C_a is theoretically limited from above by,

$$C_a \leq \begin{cases} BI & \text{for } BI < C_p, \\ C_p & \text{for } BI \geq C_p. \end{cases} \tag{8}$$

This means that in order to achieve 920 GFLOPS in single-precision and assuming $I = 1.63$ the memory bandwidth should be at least 564 GB/s; for half-precision we will assume a simple factor two reduction in bandwidth requirements, or 282 GB/s. The bandwidth requirement is just within reach of contemporary GPU devices: the Tesla P100 (released 2016) delivers 720 GB/s using HBM2 memory (as opposed to GDDR5 used in previous generation devices) and would be able to keep up with real-time processing.[6]

3.3.2 Data input and output

Almost all of the GP-GPU devices currently available use PCI Express Generation 3 for data transfer between host and device. This limits I/O to 126 Gbps

[4] We neglect the $\mathcal{O}(N)$ cost required to compute a $2N$-size real-valued FFT using an N-sized complex-valued FFT.

[5] In fact, the operational intensity of the FFT generally improves with problem size up to a point where the cache size becomes the limiting factor. Generally for large N operational intensity is somewhere between 1 and 2, see for example [4, 5].

[6] Although the half-precision bandwidth requirement is more easily met even in so-called "gaming class" GPUs, the available compute performance in half-precision is not always sufficient, e.g. the NVIDIDA GTX 1080 which offers 320 GB/s memory bandwidth and up to 9216 GFLOPS single-precision, can only do half-precision computations up to 144 GFLOPS.

in each direction (over a 16-lane interface) for the time being which is sufficient to support processing of a single BBC on one device.

Apart from moving data between the host and device, the host itself should be able to keep up with the I/O rate needed to sustain the compute rate on one (or perhaps multiple) devices. Getting data onto and off of the host in the first place will likely require transport over multiple 100G (or 40G) ethernet[7] interfaces, at least until 400G ethernet becomes available.

It is not yet clear how effective technologies such as RDMA (Remote Direct Memory Access) will be in high-throughput transfers split across multiple network interfaces, or how sophisticated transmitter / receiver implementations need to be to leverage the benefits these technologies offer.

3.3.3 Cost Per Unit

Pricing for the Tesla P100 does not seem to be directly available. However, NVIDIA has released the DGX-1 HPC server featuring eight Tesla P100 GPUs, quad 100Gb InfiniBand networking, dual 10Gb ethernet and a 7 TB SSD cache, and price per unit is estimated to be around 129 k$. Assuming the host cost is about 30 k$, the cost per Tesla P100 is approximately 12 k$. Further we assume the host is capable of sustaining the I/O necessary to process two BBCs, which brings the total cost per BBC equal to $(30+2\times12)/2 = 27$ k$ per BBC.

3.3.4 Power Efficiency

The DGX-1 is rated at 3200 W, of which $2400 = 300\times8$ is attributed to the eight Tesla P100 GPUs. Assuming 300 W consumption at peak performance of 19038 SP-GFLOPS, the Tesla P100 can deliver about 63 HP-GFLOPS/W. The requirement for BBC processing is only 920 GFLOPS, so that we assume about 15 W is consumed within each GPU, per BBC.

3.4 CPU

The compute performance in this application (mainly FFT calculation) of GPUs is generally superior over that of CPUs: based on the results in for example [8] which compares a Core i7-960 (released 2010) to an NVIDIA GTX 280 (released 2008), a conservative estimate finds GPU performance at least three times better than CPU performance, in terms of GFLOPS. Considering a more recently released CPU, the Intel Xeon Processor E5-2699v4 [9] (14 nm, released Q1 in 2016) has a peak compute performance and memory bandwidth around 900 GFLOPS[8] and 76.8 GB/s, respectively; the peak memory bandwidth is almost an order of magnitude lower than needed to implement the F-engine on a single device, and overall performance is far behind contemporary GPU devices.

[7] We assume data transmitted over ethernet to and from the host, although other solutions may be available / preferable.

[8] Based on data from https://www.microway.com/knowledge-center-articles/detailed-specifications-of-the-intel-xeon-e5-2600v4-broadwell-ep-processors/ (accessed 1 September 2016).

With a per-unit price around 4 k$ (not including supporting hardware) and total dissipated power of 145 W, and factoring in several units required to process a single BBC, implementation on CPU does not seem to present a competitive solution with respect to other platforms considered here.

4 Comparison

We now compare the various platforms based on some of the FoM listed in the *ALMA Correlator Study Work Breakdown* document.

4.1 Quantitative Figures-of-Merit

The primary FoM that allow a quantitative comparison are power efficiency (as dissipation-per-BBC), single-unit processing capability (combining compute capability, I/O bandwidth, and memory in a single platform-agnostic measure), and single-unit cost scaled to reflect cost-per-BBC. A summary of the results are shown in Table 5. Of the all the platforms FGPAs are expected to provide the most cost effective (in terms of unit price) and GPUs the most power efficient solution.[9] The numbers in the CPU column can probably be scaled by a factor $\sim$10 (in the direction indicated by the greater-/less-than symbols) for comparison to other platforms. Power efficiency for ASIC implementation is estimated to be at least as efficient as that for FPGAs.

FoM	**Unit**	**FPGA**	**ASIC**	**GPU**	**CPU**
Power-per-BBC	[W]	20	$\ll$17(?)	15	$\gg$145
BBCs-per-unit	—	1	1(?)	1	$\ll$1
Cost-per-BBC	[k$]	14	37–124(?)	27	$\gg$4

Table 5: Quantitative comparison of various platforms for F-engine implementation. Results reflect some of the latest technology currently available on each platform.

4.1.1 Future Projections

Where possible we have compared the different platforms based on some of the latest available technology in each category. For GPUs (NVIDIA P100 architecture) and FPGAs (Xilinx Ultrascale+) this represents 16 nm, and for CPUs (Intel Xeon E5v4) 14 nm. Scaling according to Moore's law towards 2022 suggests that technology might be relatively well-matched across the different platforms and that the future comparison may look similar to that in Table 5.

For FPGAs the present limiting factor in per-single-unit processing capability is the on-chip memory. Using a multi-stage approach as proposed above

[9] Note that power consumption in the hosting hardware for each platform is not included in the power budget.

overcomes this constraint and allows lower-performance units to process a single BBC, or enables higher-end devices to process multiple BBCs. Future devices may have sufficient resources to process 2 or 4 BBCs in a single FPGA. High-speed I/O utilization is fairly low and it is probably safe to assume that this will not be a limiting resource.

For GPUs the present limiting factor in per-single-unit processing capability is the on-board memory bandwidth. The adoption of HBM provided a significant improvement in this area. By 2019/2020 devices may become available with HBM3 which could add another doubling of memory bandwidth[10] and allow processing of multiple BBCs in a single GPU unit. Assuming that over the same time frame NVLink transfer rate also doubles, I/O bandwidth is not expected to be limiting performance. The compute capability is sufficiently far ahead of bandwidth limits that it is not expected to be a limiting factor any time soon.

If the super-linear trend in ASIC development cost with process node continues as shown in Figure 2, the price of implementation on this platform may even be less competitive in the future.

4.2 Qualitative Figures-of-Merit

4.2.1 COTS

GPU and CPU platforms fall very clearly into the commercial-off-the-shelf category. However, it should be noted that GPU platforms geared towards HPC applications seem to rely on InfiniBand as the interconnect of choice, which for example has standard support for RDMA. There may be some small engineering cost in achieving optimal performance with using ethernet instead.

FPGA implementation will likely require some intermediate point between COTS and custom design. A suitable hardware framework may perhaps become available from within the CASPER community; alternatively the availability of evaluation kits / reference designs may be leveraged to aid rapid development of a custom solution.

ASIC implementation will necessarily involve mostly custom design.

4.2.2 Ease of Interfacing

GPU and CPU platforms will likely receive incoming data and transmit output data over high-speed network interfaces. For GPUs there is an additional layer of complexity in getting data on and off the device. These are standard interfaces or at least expected to be well-supported (in the case of NVLink).

Interfacing with an FPGA in this application will require utilizing the very fast GTY transceivers (or possibly even faster future solutions), which may add some additional engineering cost.

[10] http://arstechnica.com/gadgets/2016/08/hbm3-details-price-bandwidth/

Interfacing with an ASIC will likely require similar challenges as that for FPGA implementation; in addition, this may also cause a rise in design cost to acquire externally developed IP for inclusion into the ASIC design.

4.2.3 Design Software and Libraries

Libraries for efficient FFT computation on either GPU or CPU are widely available. The challenge on these platforms will likely be the development of software to efficiently transfer data to / from the FFT computation itself.

A suitable toolset may become available from within the CASPER community to aid development for FPGA implementation. Some high-level synthesis solutions either in development[11] (e.g. Python-wrapped HDL, ALCHA) or already available (e.g. Xilinx HLS, Matlab HDL Coder).

Apart from perhaps some form of functional implementation of the F-engine processing, ASIC design is likely to be exclusively done by a third party developer.

5 Conclusion

Overall GPU and FPGA platforms are expected to outperform ASIC and CPU platforms by far, so our choice comes down to a decision between GPU and FPGA.

Assuming a multi-stage F-engine implementation could be used, FPGAs seem to offer the most cost-effective and second most power-efficient solution. (GPUs slightly outperform in the latter area, although that may change when factoring in power consumption of hosting hardware and peripherals.) In general developing on FPGA may prove somewhat more difficult than GPU, although toolflow development over the next few years could close this gap to some extent. Finally, FPGA implementations benefit from more fine-grained control of dataflow and execution through the F-engine pipeline.

References

[1] M. P. Rupen, A. Baudry and R. Lacasse. "Scientific Specifications and Requirements for the Next Generation ALMA Correlator," Version 0, June 30, 2016.

[2] R. Primiani, et al. "SWARM: A 32 GHz Correlator and VLBI Beamformer for the Submillimeter Array," *JAI*, 2016.

[3] S. G. Johnson and M. Frigo. "A modified split-radix FFT with fewer arithmetic operations," *IEEE Transactions on Signal Processing*, vol. 55, no. 1, pp. 111–119, 2007.

[11] See CASPER Workshop 2016 for examples.

[4] S. Williams, A. Waterman and D. Patterson. "Roofline: An insightful visual performance model for multicore architectures," *Communications of the ACM*, vol. 52, no. 4, pp. 65–76, 2009.

[5] "The NVIDIA CUDA Fast Fourier Transform Library (cuFFT)," 2016. [Online]. Available: `https://developer.nvidia.com/cufft`. Accessed 10 August 2016.

[6] Xilinx, "UltraScale+ FPGAs: Product Tables and Product Selection Guide," XMP103 (v1.8), 2016.

[7] NVIDIA, "NVIDIA NVLink High-Speed Interconnect: Application Performance," 2014.

[8] V. W. Lee, et al. "Debunking the 100X GPU vs. CPU myth: an evaluation of throughput computing on CPU and GPU," *ACM SIGARCH Computer Architecture News*, 38.3, pp.451-460, 2010.

[9] Intel. "Intel Xeon Processor E5-2699 v4," 2016. [Online]. Available: `http://ark.intel.com/products/91317` . Accessed 1 September 2016.

2.3 Determine F-engine architecture given chosen DSP platform

*Assigned to **Primiani**, Saez, Herrera, A. Young, Carlson, Lacasse, Baudry & Weintroub*

1. Polyphase Filter Bank vs FFT-only (consider possible spur isolation)
2. Number of F-engines per single DSP platform (may be more or less than 1)
3. Minimum channel width given DSP platform compute capability and demux
4. Coarse and fine delay tracking location and memory considerations
5. Complex gain requirements: bandpass correction, de-Walsh, fringe stopping, etc.
6. Transpose necessary for X-engine? Consider allocating memory if needed
7. Quantization before the corner-turn
8. Output interface to the corner-turn

Digital Correlator and Phased Array Architectures for Upgrading ALMA

WP2.3: Determine F-engine architecture given chosen DSP platform

July 29, 2017

Contents

1 Introduction

The main outcome of WP2.2 was the decision that FPGA is expected to be the most suitable platform for implementation of the F-engine [**wp2d2**]. Furthermore, since on-chip memory is an important limitation for FPGA platforms it was also proposed in WP2.2 that the F-engine be implemented using a multi-stage architecture.

Herein we compare various architectures, including a single-stage F-engine, to decide on the most suitable solution. In addition to the Fourier transform core functionality we also include other subsystems associated with this component, e.g. input/output, monitor and control, complex gain, etc. A detailed top-level description as well as an estimate of platform resources for the chosen architecture are provided in the conclusion.

Throughout this document the baseline specifications for the F-engine listed in Table 2 will be used.[1] These specifications are derived from relevant require-

[1] These specifications may deviate to some extent from those in Table 2 in [**wp2d2**] due to recent changes in the scientific requirements.

ments found in [**mainspec**] and repeated here in Table 1; that document itself is an abbreviated set of the requirements listed in [**wp2d1**].

Parameter	**Requirement**	**Comments**
Baseline delay range	300 km	Impacts on buffer memory requirements for coarse-delay correction.
BBC bandwidth	8 GHz	Impacts on throughput requirements.
Sample format	4-bit (in) & 4-bit (out), 16 GS/s	Impacts on I/O bandwidth requirements.
Spectral resolution	1 kHz	Impacts on FFT size.
Spectral channels per BBC	8×10^6	Impacts on FFT size, maximum number assumed here.

Table 1: Parameters used to derive baseline F-engine requirements.

Parameter	**Requirement**	**Comments**
FFT size	$2^{23} = 8$ Mi	Assuming N-point FFT of real-valued sequence implemented as an $N/2$-point FFT of complex-valued sequence. Achieved spectral resolution is 0.953... kHz.
Throughput	1.049 ms / FFT	One FFT computed for every 2^{24} samples at a rate of 16 GSa/s.
I/O bandwidth	64 Gb/s (in) & 64 Gb/s (out)	4-bit $\times$ 16 GSa/s per BBC.
Coarse-delay buffer	125000 Kib	(2$\times$300 km/c) $\times$ (4-bit$\times$16 GSa/s) / 1024 per BBC.

Table 2: F-engine requirements per digitized baseband channel.

2 Delay Tracking

This section will list and state the hardware (logic and memory) requirements for implementing the instrumental delay given the correlator capabilities (baseline length and sampling rate). A possible implementation will be presented. The subsystem presented here will be able to generate instrumental delays with granularities of 64 and 1 sample (@ 16GHz) of accuracy. Granularity of 1/16 is expected inside antenna electronics but are not under the scope of this study. Finer resolution (less than a 1/16 of a sample) will be conducted after the F-engine modifying the phase values along the bandwidth

The delay naming convention for this study is defined as follows:

- Bulk delay: A delay applied inside correlator electronics (before F-engine) with a resolution of 64 samples per step (Section 2.2.1).
- Coarse delay: A delay applied inside correlator electronics (before F-engine) with a resolution of a sample per step (Section 4).
- Fine delay: A delay applied inside antenna with a *expected* resolution of 1/16 of a sample per step (Section 2.2.4).
- Residual delay: A delay applied inside correlator electronics (after F-engine) with a smaller resolution than fine delay (Section 2.2.5).

2.1 Memory requirements

According to the system specifications given for the next generation correlator, (relevant specifications for the instrumental delay design in Table 3),

Spec	**Value**	**Impacts**
Maximum baseline	300 [Km]	on buffer memory size for instrumental delay implementation
BBC bandwidth	8 [GHz]	on logic speed
Sample format	4-bits in/out	on memory organization and I/O ports
Sample rate	16[GS/s]	on logic speed
Demux factor	64	on logic speed, memory organization, I/O ports and logic resources

Table 3: Delay specifications

The memory requirements to perform a delay with such extend are shown below. First, maximum baseline length is converted into time delay:

$$maximum\,delay = \frac{maximum\,baseline}{speed\,of\,light} = \frac{300[km]}{299792.46[km/s]} = 1.000692285[ms]$$

considering the sample rate is 16GS/s, the amount of samples for such maximum baseline will be:

$$samples\,to\,be\,stored = 16{\cdot}10^9[samples/s]{\times}max.\,baseline \approx 16011077[samples]$$

for allowing flexibility and changes in both direction, the required size of the block ram will be doubled, this means the needed capacity will be:

$$samples\,to\,be\,stored = 32,022,154[samples]$$

Samples are meant to be stored in 4 bits. Therefore, the total memory needed for storing delays per BBC is 128,088,616 bits.

Since the demux factor is 64, 256-bits (4 bits x 64) will be presented simultaneously, this means the selected memory must be able to store words 256-bits wide. Another requirement is the one regarding the memory speed, in this case considering the demux factor the clock rate must be 250[MHz].

The memory requirements for implementing the instrumental delay for one antenna and one BBC are:

- word width = 256 bits
- Frequency of operation = 250MHz
- Memory Size = 32,022,154 [samples] x 4 [bits/samples] = 128,088,616 [bits]

The amount of memory to serve the purpose of sampling at 16[GS/s] for baselines of 300[km] can be obtained for certain group of Virtex UltraScale+ on-chip memory as can be seen in Table 4.

Device Name	**VU3P**	**VU7P**	**VU11P**	**VU13P**
System Logic Cells (K)	862	1,724	2,835	3,780
Total Block RAM (Mb)	25.3	50.6	70.9	94.5
UltraRAM (Mb)	90.0	180.0	270.0	360.0
DSP Slices	2,280	4,560	9,216	12,288
GTY 32.75Gb/s Transceivers	40	80	96	128

Table 4: Virtex UltraScale+ FPGA families specifications

Therefore, memory requirements can be satisfied with current technology by means of UltraRAM memory blocks from FPGA families having over than ∼128Mb of capacity.

2.2 Implementation approaches

In order to process the incoming data stream and given the FPGA speed limitation, a de-multiplexing scheme will be used, in this case the the demux factor will be 64, which means to process 64 simultaneous samples at a rate of 250MHz. In addition, delays to be applied in correlator will be multiple of a sample, therefore we plan to implement a two-fold delay processing scheme namely *bulk delay* (Section 2.2.1) and *coarse delay* (Section 2.2.3), its graphical representation is depicted in Figure 1. In addition, we also present an alternative solution for storing bulk delay samples based on *off-chip memory* (Section 2.2.2)

2.2.1 Bulk Delay

It will be named "Bulk Delay" to the mechanism for implementing the instrumental delay with an accuracy of 64 samples, the designation of 64 samples comes directly from the demux factor, since it will define how much samples are devoted to be stored simultaneously in a single clock operation.

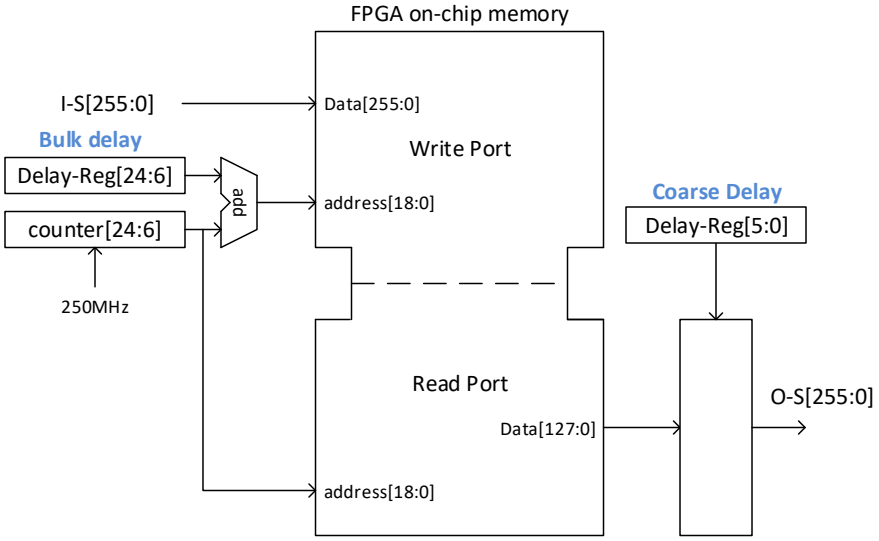


Figure 1: High level representation of delay approach.

Given the size limitations to store the bulk delay, we foresee the usage of the latest addition of on-chip memory such as UltraRAM blocks, these units were devoted to match UltraScale+ FPGAs memory and speed requirements. A summary of its key features are:

- 288K bits of storage in a single block.
- Dual port, 4K x 72, single clock synchronous memory.
- UltraRAM cascade for building larger blocks.
- Error correction coding (ECC) on both ports.
- Optional pipeline flip-flops on the inputs, outputs, and cascade paths.

Having in mind that these blocks manage only 72 bits-wide data, and considered we need to store 4 x 64 bits data in a single clock rise. We propose to access memory under the scenario described in Figure 2.

Several memory blocks will be cascaded for each bit in order to access the total requirement of 32,022,154 bits. The way how they will be cascaded will be defined according to the FPGA generation to be considered in the implementation stage.

2.2.2 Off Chip Memory Based Bulk Delay

This section describes an alternative to the integrated on-chip memory approach for the bulk delay, the aim of this allocation is to free up on-chip memory for the

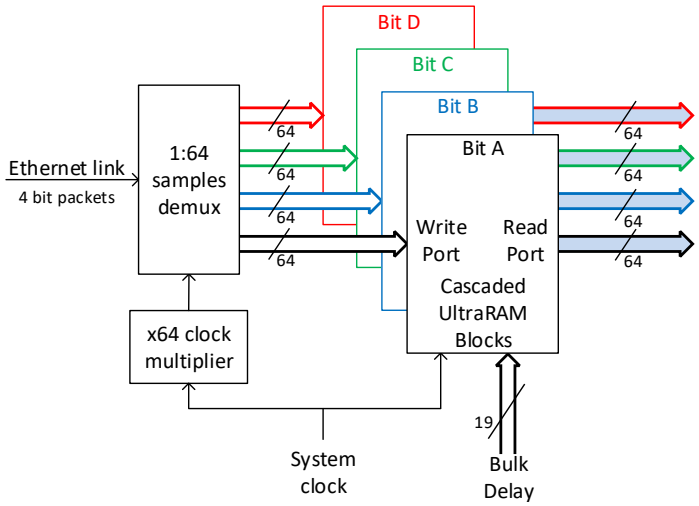


Figure 2: Memory structure for storing and reading samples.

F-engine processing, this alternative structure maintains the sample allocation requirements as handled by the on-chip approach.

The high end signal processing requirements in this project will make usage of the most up-to-date technologies, one of the latest memory architectures to interface with UltraScale+ devices are the RLDRAM3 (Reduced Latency DRAM) memories [**Micron16**]. Some of the main features are:

- 1066 MHz DDR operation (2133 Mb/s/ball data rate).
- 16 Meg x 36 common I/O (CIO).
- Single Port RAM.
- SDR addressing.
- Programmable read/write latency and burst length.
- Extended operating range (200MHz to 1066MHz).

In order to comply with the on-chip bulk delay access, the memory organization to seamlessly move from on-chip to off-chip approach is shown in Figure 3.

Since RLDRAM3 memories are brought up to 36 bits wide, we propose to combine two chips in order to process 72 bits bus, where 64 of them will be used to store samples (dividing data into MSBs and LSBs buses). Unlike the independent memory structure presented for UltraRAM blocks to process samples bits (4 entities, one per sample bit), the architecture for off-chip memory

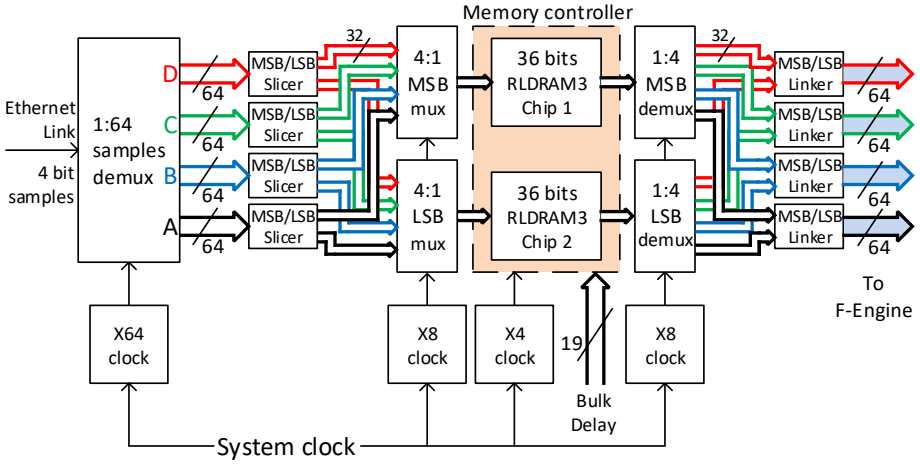


Figure 3: Off-chip memory structure for storing and reading samples.

will take advantage of the large capacity these chips have (576Mb each chip, 1152Mb total) allowing to store all bits samples (a single entity to store/read ∼128Mb). The changes for storing and reading from this scheme comprises the split and link of the bits' samples, the transactions with RLDRAM3 will be handled by a memory controller, this interface which consist in converting single port memories into dual port memories by means of the usage of DDR capabilities aided with read/write burst access under the restrictions of SDR addressing.

This memory alternative will then require additional logic for processing bulk delay offsets in compliance with the original on-chip approach. The clocking rate required for this memory is four times the system clock (1000MHz) which is an acceptable rate (maximum 1066MHz), the mux/demux stages will require to clock eight times (2000MHz) the system clock (250MHz).

The RLDRAM3 will require a bus of 32 bits (MSBs or LSBs) for storing/reading half of the total 64 samples in their 4 bits, the addressing is subject to the size of the bulk delay which are 19 bits long, the memory controller will translate delay coarse commands (19 bits) into proper addressing commands for each RLDRAM3 chips (20 bits addressing and 4 bits memory banks).

Although one single RLDRAM3 memory can store all delay samples for each antenna and even combining polarizations, the chip would have to increase its clocking rate which is currently not feasible.

2.2.3 Coarse Delay

It will be named "Coarse Delay" to the mechanism for implementing the instrumental delay with an accuracy of 1 sample.

As seen with the bulk delay design in Figure 2, 64-sample steps can be achieved using demux and cascaded memory blocks approach. In order to apply delays with a smaller granularity, we introduce the coarse delay architecture depicted in Figure 4 and the detailed barrel shifter approach in Figure 5.

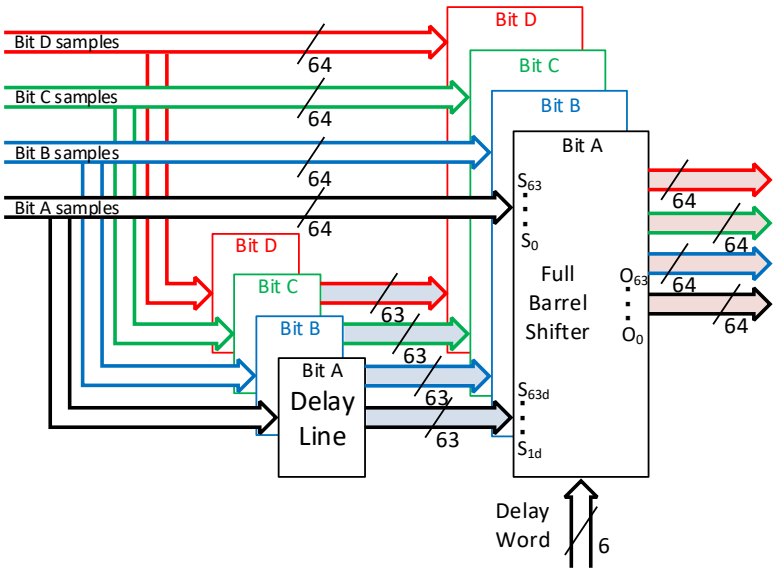


Figure 4: General view of fine delay.

The core of this architecture is a fully connected barrel shifter, the coarse delay block is feed by samples directly from bulk delay stage and a delayed bus of the latest 63 samples. Using such architecture, delays can be configured from 0 to 63 samples period with steps of one sample. The added chip resources for this feature are presented in Table 5.

Resource	Amount
CLB	304
LUTs as logic	768
LUT Flip Flop pairs	768

Table 5: Resources for coarse delay

Concluding from the report of required logic, the coarse delay feature needs only a tiny fraction of FPGA resources.

2.2.4 Fine Delay

This section identifies there is a delay of a fraction of a sample defined as "Fine Delay" that is applied in the digitalization stage inside ALMA antennas with a resolution of 1/16 of a sample period, for the next generation ALMA it is

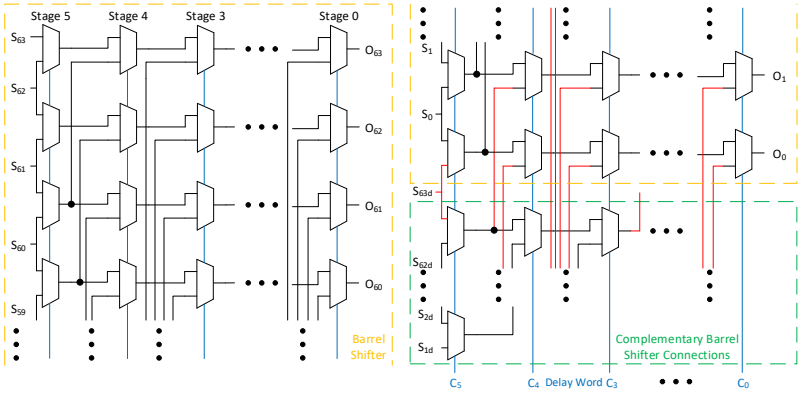


Figure 5: Fully connected barrel shifter.

expected this delay will be implemented changing the digitizer clock phase, further details of its internal function are beyond the scope of this report.

2.2.5 Residual Delay

This section points the existence of a delay whose resolution is less than the fine delay (less than 1/16 according to our future expectations of fine delay) and it is defined as "Residual Delay". The remainder delay after the signal is adjusted in the processing of bulk delay, coarse delay and fine delay is corrected in this stage. The location of this delay unit is therefore at the output of the F-engine

Some of the main considerations for this unit are:

- It will be corrected changing the phase values along the spectral channel.
- It will be implemented in the F-engine output stage as a set of values multiplying each spectral channel coarse delay = 1 sample delay resolution.
- It will be implemented using a barrel shifter logic bulk delay = 64 samples resolution.
- It will be implemented using a big dual port memory.

3 Single Stage Channelizer

The resources required to implement a PFB on an FPGA are approximately [**smamemo1**]:

$$R_{mul} = D\log_2(ND) + TD - 2D \tag{1}$$

$$R_{mem} = bN\left(\frac{1}{2}\log_2 D + T + 2\right) - 5bD \tag{2}$$

where

$$\begin{aligned} N &= \text{number of spectral points} \\ D &= \text{number of parallel inputs} \\ T &= \text{number of taps in PFB FIR} \\ b &= \text{number of bits used for data buffering / coefficient storage} \\ R_{mul} &= \text{number of multipliers needed} \\ R_{mem} &= \text{amount of memory needed in bits.} \end{aligned}$$

For an ≈8 million spectral point PFB, $T = 4$, $b = 18$, and $D = 64$ (which assumes an FPGA clock of about 250 MHz and sampling rate of 16 GSa/a), these expressions evaluate to

$$R_{mul} = 1984 \quad \text{and} \quad R_{mem} = 1.266 \text{ Gib.} \tag{3}$$

The number of multipliers (and similarly the number of adders, which is of the same order) is not much of a challenge for even contemporary FPGAs available from Xilinx. However, the memory requirements far exceed what is available in the latest generation of FPGAs. Since the memory usage is distributed across the stages within the FIR and FFT using off-chip memory to meet the requirement is also not practical. For this reason a single-stage channelizer architecture is ruled out.

4 Multi Stage Channelizer

This section should descrive various multi-stage channelizer architectures in enough detail to compare them against each other (and the single stage approach). Estimated resource utilization for the target FPGA is the most critical figure here.

4.1 PFB followed by per-channel DFT, critically sampled

This section will present a possible implementation of the F-engine based on a two stages processing: first a Polyphase Filter-bank (coarse channelization) followed by a FFT per each PFB output channel (fine spectral analysis).

A description of this approach is presented and a possible implementation will be described. FPGA resources utilization will be analyzed in order to help to select a suitable FPGA model (in which the proposed design fits). In addition this analysis will be used for comparing the resources utilization respect to other possible approaches.

Performance of the proposed implementation will be studied, paying attention to the artifacts introduced by the discrete and finite nature of the signal processing (leakage, scalloping and quantization noise).

4.1.1 Description

The proposed architecture in this section consists in two stages:

- The first stage is a coarse channelization step, where the wideband signal is decomposed in several narrowband signals, in this specific case in 1024 narrowband signals (which will be called sub-bands).
- The second stage is the spectral analysis of each individual sub-band, in this case a 15625-points FFT ($15625 = 5^6$) will be the option.

The output of this subsystem is the spectra of the entire wideband signal.

4.1.2 Relevant specifications

The relevant specifications which will drive the design of this implementation are listed in table 6.

It is important to note that the total amount of spectral channels is not a power of two, this is because the proposed design must produce data at a multiple rate of the ALMA Walsh period, 16[ms] for side band separation, in this case it was selected 1[ms].

In addition, all the real time operations of the ALMA instrument are based on a time reference called TE (Timing Event) whose period is 48[ms], therefore is it strongly advisable to select an integration time multiple of this signal period.

Parameter	**Requirement**	**Comments**
FFT size	$8 \cdot 10^6$ [complex values]	Assuming N-point FFT of real-valued sequence implemented as $N/2$-point FFT of complex-valued sequence. The achieved spectral resolution is 1[kHz].
Throughput	FFT/1[mS]	One FFT computed for every 16000000 samples at a rate of 16 GSa/s.
Sample format	4-bits (in) & 4-bits (out)	4-bit $\times$ 16 GSa/s per BBC.
Demux factor	64	16 GSa/s /64 $\approx$ 64 consecutive samples every 4ns per BBC.

Table 6: F-engine requirements per digitized baseband channel.

4.1.3 Architecture

The figure 6 shows a conceptual view of this implementation. The x[n] is a time series produced after the quantization (4-bits, 16 levels) and sampling at 16[GS/s] an IF signal. The Sx[n] with x ranging from 0 to 1023 are the outputs of the Polyphase Filter-bank, each output is a time series of 7.8125[MHz] bandwidth. Then the spectral analysis of each Sx[n] is performed using a 15625-points FFT, in this case the idea is take advantage of $O(NlogN)$ optimization the Radix-5 DFT algorithm can afford.

Finally, the output is the result of stitching 1024 narrowband spectrums (each spectra is called XN[K], with N ranging from 0 to 1023).

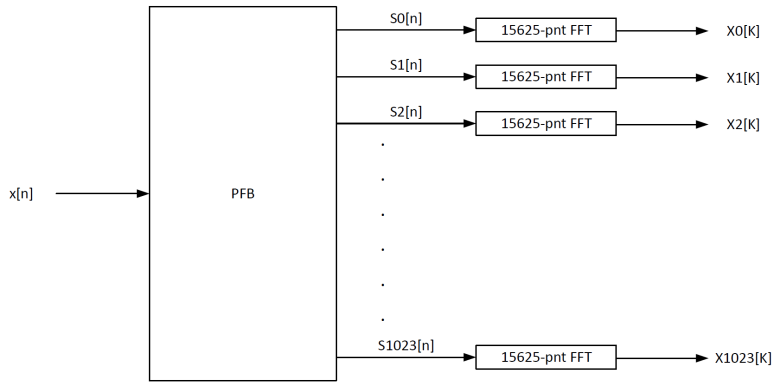


Figure 6: Block diagram representing the Polyphase Filter-bank followed by a set of 1024 15625-points FFT. The demux factor for the PFB is 64 (in order to fit the design into the FPGA time constraints). The amount of T taps will be evaluated later. The re-quantization is not presented here, but it must be considered for latter stages

4.1.4 Resources utilization

In this section the amount of operations needed for performing the proposed architecture will be calculated.

In addition an evaluation of the needed memory capacity will be presented.

For evaluating the needed amount of multiplications, additions and memory the expressions derived in [**PFBSMA**] and two dimensional FFT/PFB will be used.

The coefficient bit width will be assumed to be 18+18 bits (given by the DSP slice input width).

Analysis of the PFB and FFT implementation based on serial data input will not be considered since the available results in "two dimensional FFT/PFB" shows a highly demanding memory resources for those approaches.

Given the values stated in table 6, we have:

- $D_{PFB} = 64$ (for the PFB)
- $T_{PFB} = 4$ (for the PFB)
- $N_{PFB} = 1024$ (for the PFB)
- $N_{FFT} = 15625$ (for the FFT)
- $b^{\Sigma} = 72$ (for the PFB)

Polyphase Filter-bank

$R_{\times} = D \cdot log_2(DN) + TD - 2D$
$R_{+} = \frac{3}{2}D \cdot log_2 ND + D(T-1) + D$

$R_M = b^{\Sigma} N(\frac{1}{2} log_2 D + T + 2) - 5b^{\Sigma} D$
$R_\tau = 2(N - D)$

Where the $R_\times$ is the number of real-multipliers, R_+ is the number of real-adders, R_M is the number of synchronous register for storing the FIR coefficients and twiddle factors, and R_τ is the number of data points that require buffering.

Resource	**Quantity**
Multipliers, $R_\times$	1152
Adders, R_+	1792
Memory, R_M	475Kib
Memory, R_τ	1920 samples, 68Kib

Table 7: Summary of the resources requiered for the PFB

15625-points FFT

$R_x = 16D \cdot log_5(N)$
$R_+ = 16D \cdot log_5(N)$
$R_M = 16D \cdot log_5(N)$
$R_\tau = \frac{3}{2}(N - D)$

Resource	**Quantity**
Multipliers, $R_\times$	12000
Adders, R_+	12000
Memory, R_M	844Kib
Memory, R_τ	2350 samples, 83Kib

Table 8: Summary of the resources requiered for the FFT

2-banks Buffer

An intermediate step is needed between the output of the PFB and the input of the 15625-points FFT.

The aim of this step is to re-sort the outputs of the PFB in a suitable format for feeding the 15625-points FFT. In order to achieve this goal, a change of the Demux factor between the PFB and the 15625-points FFT is essential (where $D_{PFB} < D_{15625-FFT}$), considering the clock rate limitations and the FFT sizes, we have $D_{PFB} = 64$ and $D_{15625-FFT} = 125$.

For accomplishing this step, a 2-banks buffer will be needed. It must be able to support concurrent process of reading and writing. While one bank is being written by the output of the PFB, the other bank will be read by the 15625-points FFT.

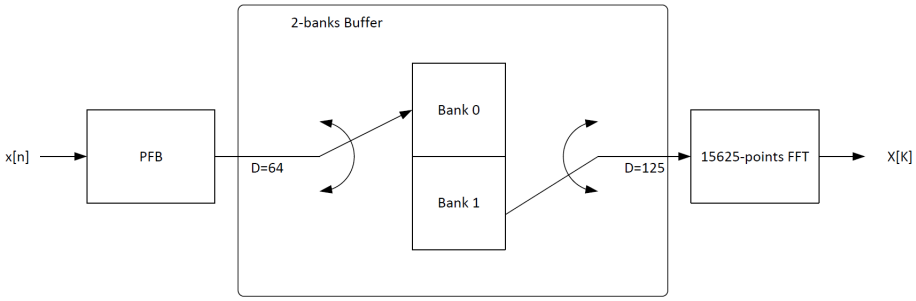


Figure 7: Conceptual representation of the 2-banks buffer, the switches changes their position every 1msec

4.1.5 FPGA resources

For making an estimation of the needed FPGA resources the listed assumptions will be used:

- 18 bits real values out of the FIR.
- 18+18 bits complex values for the data after the first stage of the PFB's DFT.
- 18+18 bits complex values for the coefficients of the PFB's DFT.
- The output of the PFB is re-quantized to 4+4 bits complex.
- The twiddle factors are stored as 18+18 bits complex.

The realization of one real multiplication can be done using a DSP48E1 primitive (48-bit Multi-functional arithmetic block).

Resource	**PFB**	**15625-points FFT**	**2-banks buffer**	**Required**
Multipliers	1152	12000	0	13152
Adders	1792	12000	0	13792
On-chip memory	543Kib	927Kib	0	1470Kib
Off-chip memory	0	0	$16 \cdot 10^6$x4b(write) + $16 \cdot 10^6$x4b(read)	$128 \cdot 10^6$b

Table 9: Summary of FPGA resources requiered for the proposed architecture.

4.1.6 Performance

Due to the finite nature of the amount of samples and quantization levels, it is expected to introduce undesired artifacts like:

- Scalloping
- Leakage
- Quantization noise

In addition the finite nature of the number representation will also add noise and therefore a degradation of the signal to noise ratio $\frac{S_{rms}}{N_{rms}}$, this noise will be named *quantization noise*

The above listed artifacts will be further described, and characterized:

Leakage

In this section two types of leakage effects will be analyzed:

- Leakage inherent to the DFT
- Leakage between sub-bands

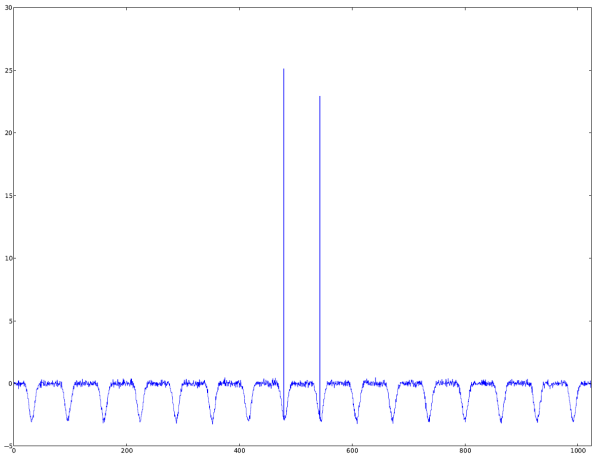


Figure 8: Leakage of a tone into an adjacent channel when using a critically-sampled PFB as first stage followed by a second stage transform. Also of note is the scalloping due to the PFB channel edge roll offs.

Scalloping

The scalloping loss inherent to the straightforward DFT creates an uncertainty in sine-waves peak amplitude estimations.

The effect of the scalloping will be evaluated in the second stage of the spectral analysis.

Since this design adds a filter bank before the DFT, the scalloping loss is reduced using an adequate windowing function.

The scalloping loss of a window $a(k)$ of length N is:

$$SL = \frac{\left|\sum_{k=0}^{N-1} a(k)e^{-\frac{j\pi k}{N}}\right|}{\sum_{k=0}^{N-1} a(k)} = \frac{\sqrt{(\sum_{k=0}^{N-1} a(k)cos(\frac{\pi k}{N}))^2 + (\sum_{k=0}^{N-1} a(k)sin(\frac{\pi k}{N}))^2}}{\sum_{k=0}^{N-1} a(k)}$$

Using the Hanning window, the scalloping loss can be reduced down to $-1.75[db]$.

Quantization noise

For N-bits, the signal to noise ratio is given by:

$$fracSN = 6.02N + 1.76dB$$

for 18 bits we get,

$$fracSN = 6.02N + 1.76dB = 110[db]$$

Consequently, the signal to noise ratio will be dominated by the 4-bits sampler (24db).

4.1.7 Conclusions

As a conclusion of this section:

- Using a PFB for coarse channelization will introduce power drops between sub-bands, one way to get rid of those artifacts is using an oversampling scheme, in this case the spectral channels located in the edges of the filter can be discarded and a final spectra can be made by stitching the passband of consecutive filters.
- Leakage between spectral channels, the spectral analysis conducted by the Radix-5 FFT will introduce the usual leakage between channels, this is because no windowing or PFB is applied before the FFT.

4.2 Two-dimensional FFT/PFB

Using a mixed-radix implementation of the Cooley-Tukey FFT algorithm, a transform of size $N = N_R \times N_C$, where N_R and N_C are integers, can be computed as a series of smaller transforms as in a two-dimensional DFT (cite Darren Leigh's thesis). Specifically, the computation will consist of,

1. N_R number of transforms of size N_C,
2. multiplication by N twiddle factors,
3. and N_C number of transforms of size N_R.

That is, the DFT X_k of x_n can be written as,

$$Y(\kappa, \rho) = \sum_{r=0}^{N_R-1} G'(r, \kappa) W_{N_R}^{\rho r} \tag{4a}$$

$$G'(r, \kappa) = G(r, \kappa) W_N^{\kappa r} \tag{4b}$$

$$G(r, \kappa) = \sum_{c=0}^{N_C-1} y(r, c) W_{N_C}^{\kappa c}, \tag{4c}$$

where $Y(\kappa, \rho) = X_{\kappa+\rho N_C}$ and $y(r, c) = x_{cN_R+r}$. The compute complexity remains $\mathcal{O}(N \log N)$ as for the usual implementation as a single N-point FFT. However, the decomposition has certain implications for an implementation on FPGA:

1. The calculation consists of smaller FFTs which potentially use less distributed memory.
2. Since many smaller FFTs need to be calculated along each dimension, a possible trade-off exists between performing a single FFT with large demux factor, and performing several serial input FFTs in parallel.
3. The twiddle factors needed to compute G' from G increases the total memory required.

Here it is assumed that the data is presented at the input in canonical order. Since the first transform in (4c) is computed on the data with a stride of N_R, 1) N_R number of parallel transforms need to be implemented, or 2) the data first needs to be transposed. Since $N = 16'000'000$ and for useful values of N_R option 1) is unlikely to be feasible. Similarly, a transpose operation is also needed between the first and second transforms.

The prime factor decomposition of N is $16'000'000 = 1024 \times 15'625 = 2^{10} \times 5^6$. Since the power-of-two and power-of-five sizes are not vastly different, we will start with a first iteration of this architecture that uses a 5^6-sized transform along one dimension, and a 2^{10}-sized transform along the other. For now it is also assumed that the impact of the order of the transforms is negligible (i.e. it does not matter whether $N_R = 2^{10}, N_C = 5^6$ or $N_R = 5^6, N_C = 2^{10}$).

The general architecture is shown in Figure 9.

4.2.1 Parallel input vs parallel transforms

An important consideration for implementation as in Figure 9 is whether to implement multiple FFTs in parallel that process serial data, or whether to

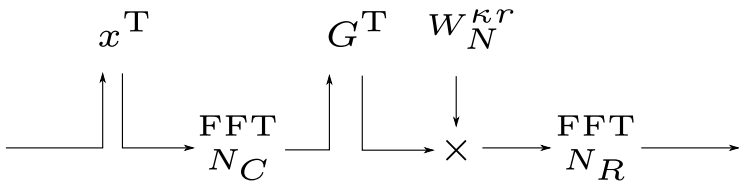


Figure 9: General architecture of F-engine implemented as two-dimensional FFT.

implement one or more FFTs that processes parallel-input data. The data will be presented to the F-engine as 64 real-valued samples per clock cycle (250 MHz) and whichever solution is chosen should process at least at this rate; faster processing may be an option, although it may complicate control logic to compensate for dead-time while the processor waits for a new frame of data, and may possibly require a rate-transition.

Power-of-Two FFT, Parallel Data Input: Consider first the implementation of a power-of-two FFT. For data presented as D parallel samples per clock, an N-sized FFT will require [**smamemo1**],

$$R_\times = D\log_2(DN) - 2D \tag{5a}$$

$$R_+ = \frac{3}{2}D\log_2(DN) \tag{5b}$$

$$R_r = 2(N-D), \tag{5c}$$

where $R_\times$ is the number of real-multipliers, R_+ is the number of real-adders, and R_r is the number of data points that require buffering.

Power-of-Two FFT, Serial Data Input: Various efficient implementations for serial data input can be found in [**Wold1984**] and [**He1996**]. Minimal memory usage is obtained with the single-path delay feedback (SDF) implementations, and for a radix-2 (R2SDF) such design the requirements are,

$$R_\times = P(4\log_2 N - 8) \tag{6a}$$

$$R_+ = P(8\log_2 N) \tag{6b}$$

$$R_r = P(N-1), \tag{6c}$$

where P is the number of parallel FFT implementations need to allow the same processing rate as for parallel data input. A block diagram of the R2SDF architecture is shown in Figure 10.

Since we require $P = D$ for a similar processing rate in the parallel data and serial data implementations, it is clear that the parallel data implementation will likely be much more efficient. A summary of the requirements for either implementation is given in Table 10.

Resource	**Parallel input**	**Serial input**
$R_\times$ (# of real mul)	896	2'048
R_+ (# of real add)	1'536	5'120
R_r (# of samples)	1'920	65'472

Table 10: Comparison of resources required for parallel and serial data input implementation of radix-2 FFT. Parameters used are $D = 64$, $N = 1024$, and $P = 64$.

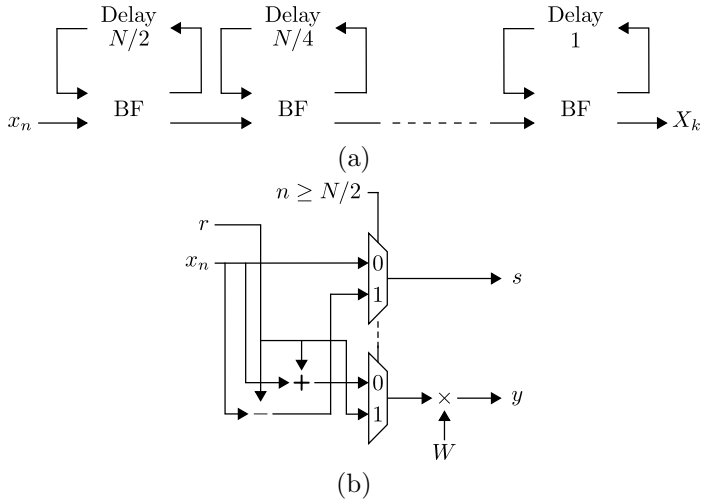


Figure 10: Radix-2 single-path delay feedback implementation of an N-point FFT. (a) Top level description. (b) First stage butterfly structure. After the $N/2$-th input the multiplexers switch from output 0 to output 1. The output s feeds into the delay buffer so that after $N/2$ clock cycles the same data appears at r. The output y is presented as input to the next stage butterfly.

Power-of-Five FFT, Parallel Data Input: For a power-of-five FFT the parallel data input implementation requires[2],

$$R_{\times} = (4 \times 4) \times 5 \times \frac{D}{5} \log_5 (N) = 16D \log_5 (N) \tag{7a}$$

$$R_{+} = (4 \times 2 + 4 \times 2) \times 5 \times \frac{D}{5} \log_5 (N) = 16D \log_5 (N) \tag{7b}$$

$$R_r = \frac{15}{2} (N - D) . \tag{7c}$$

A schematic of the building block for a radix-5 parallel data input implementation is shown in Figure 11. Each stage consists of $D/5$ of these blocks. The delay parameter in stage s is,

$$\alpha = \frac{N}{D5^s}, \tag{8}$$

so that the total number of input data samples needed to be buffered for that stage is equal to

$$R_r^s = \frac{D}{5} \times (4 + 5 + 6 + 7 + 8) \alpha = 6D\alpha = \frac{6N}{5^s}. \tag{9}$$

Since there are $\log_5(N/D)$ pipelined stages, the total number of samples needed for buffering is equal to

$$R_r = \sum_{s=1}^{\log_5(N/D)} \frac{6N}{5^s} = \frac{6N}{5} \frac{1 - 5^{-\log_5(N/D)}}{1 - 5^{-1}} = \frac{6N}{5} \frac{1 - D/N}{4/5} = \frac{3}{2}(N - D). \tag{10}$$

In the direct stages, all the data samples that need to be combined are available simultaneously so that no buffering is needed and these stages do not contribute to the memory requirements (except for coefficient storage). The number of multipliers and adders needed per stage remains the same as for pipelined stages.

Power-of-Five FFT, Serial Data Input: A serial data input implementation is also presented in [**Wold1984**] and uses the fact that a 5-point DFT can be computed efficiently using a convolution of two 4-element sequences. The radix-5 module is shown in Figure 12. This design requires,

$$R_{\times} = P \left((4 \times 4) \log_5 N \right) = P \left(16 \log_5 N \right) \tag{11a}$$

$$R_{+} = P \left(2 \times (4 \times 2 \times 2 + 4 + 2) \log_5 N \right) = P \left(44 \log_5 N \right) \tag{11b}$$

$$R_r = P \left(\frac{5}{2} (N - 1) \right) . \tag{11c}$$

In comparing parallel input data with serial input data it may be required to have $D = 5^n$ for a parallel input implementation where $D \geq 64$, i.e. $D = 125$

[2]This does not yet include any optimizations.

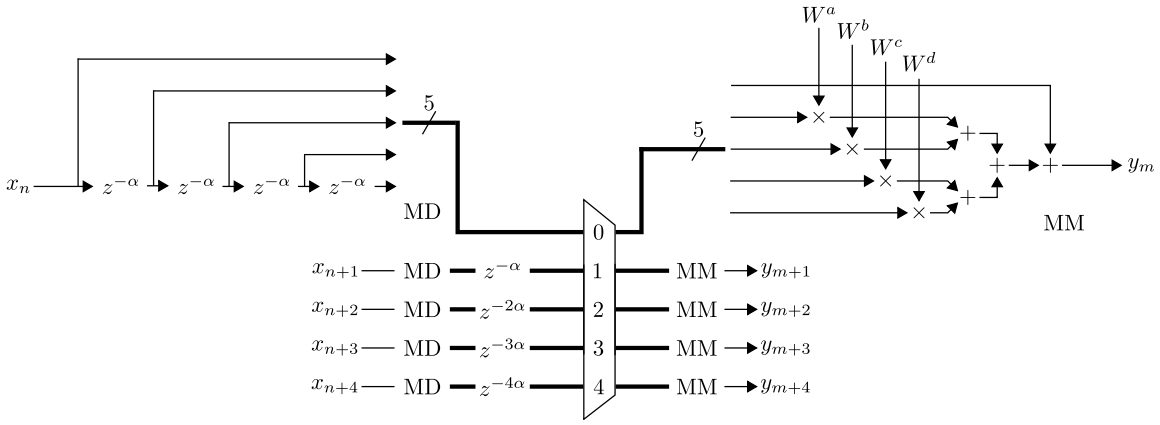


Figure 11: Component of a radix-5 pipelined stage for parallel input data. The entire stage consists of $D/5$ number of these components.

Resource	**Parallel input**	**Serial input**
$R_\times$ (# of real mul)	12'000	6'144
R_+ (# of real add)	12'000	16'896
R_r (# of samples)	23'250	2'499'840

Table 11: Comparison of resources required for parallel and serial data input implementation of radix-5 FFT. Parameters used are $D = 125$, $N = 15'625$, and $P = 64$.

to ensure that the data is processed at the required rate.[3] For a serial input implementation, however, we can use $P = 64$ which meets the data throughput requirement. A summary of the requirements for either implementation is given in Table 11.

[3]Note that a combined solution is also possible where three parallel input streams, each with $D = 25$ are implemented for a total throughput of 75 samples-per-clock. For multipliers / adders this would mean a reduction in resources by a factor 3/5, but for memory it would mean an increase in resources by approximately a factor 3.

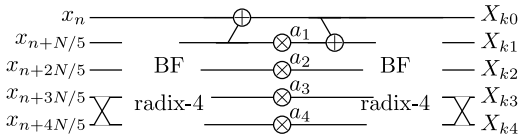


Figure 12: Implementation of a radix-5 stage using convolution of 4-point DFTs. Adapted from [**Wold1984**].

4.2.2 Off-chip memory requirements

Capacity: Each of the off-chip memory units in Figure 9 used for data storage needs to be able to store $2 \times N = 32'000'000$ data points; the factor 2 results from the need to double buffer the data so that as data from one frame is being read out, data from the next frame can be stored without overwriting unread data.

Since the first data storage chip will transpose the raw 4-bit data the total capacity needed is only 128 Mb. The second data storage chip will need to store data after possible bitgrowth within the first stage FFT; for now we assume that the data is requantized to 4-bit directly after the first stage, so that the capacity of the second data storage chip is also 128 Mb.

The third memory unit stores $N = 16'000'000$ twiddle factors. Since these factors are of the form $e^{-i2\pi x/N}$ with N very large, potentially high bit-resolution may be required for accurate representation. The capacity needed for 22-bit coefficients is around 352 Mb.

Data access: The data storage chips need to provide two independent access ports, one for writing and one for reading, which each supports a speed of at least 64-samples per FPGA clock. The twiddle storage chip only requires a single access port (initial write can be done once at start-up and thereafter only reads are needed) and needs to support 64-samples read per FPGA clock.

4.2.3 Proposed architecture

Given the much larger distributed memory requirements for serial-input FFT implementations as compared to parallel-input implementations, the latter is selected as the better option. An important implication of this choice is the need for a rate transfer, since the demux factor for the parallel-input 5^6-point FFT needs to be a power-of-five greater than or equal to the demux factor of the F-engine input / output. For a demux factor $D = 125$ the power-of-five FFT needs to be clocked at $f_{FPGA} \times 64/125 = 128$ MHz. Additionally, a FIFO buffer is needed on each boundary between different clock regions, which would be on both the input and output sides of the power-of-five FFT; the power-of-two FFT will have a demux $D = 64$ which is equal to that of the F-engine input / output. Note that by using dual-clock off-chip memory for the transpose operations in Figure 9, and requiring that the power-of-five FFT is done first, the transpose memory buffers act as the FIFOs between the clock domains.

4.2.4 Resource utilization summary

A summary of the estimated resources required for implementing the proposed architecture is listed in Table 12. The following assumptions have been made in deriving these estimates:

1. In the first transform buffered data is 4-bit real within the first stage, and then 18+18-bit complex in the remaining stages.

2. Coefficients in the first transform are 18+18-bit complex, and each stage s needs to store exactly one copy of all 5^s-th roots of unity.
3. The output of the power-of-five FFT is requantized to 4+4-bit complex before the second transpose operation.
4. The twiddle factors are stored as 22+22-bit complex values and the output after multiplication with these coefficients is 25+25-bit complex.
5. The second transform uses 25+25-bit complex data throughout.
6. Coefficients in the second transform are 18+18-bit complex, and each stage s needs to store exactly one copy of all 2^s-th roots of unity.

Resource	**Subsystem**	**Required**
Multipliers	power-of-five FFT	12'000
	twiddle multiplications	256
	power-of-two FFT	896
	Total	**13'152**
Adders	power-of-five FFT	12'000
	twiddle multiplications	128
	power-of-two FFT	1'536
	Total	**13'664**
On-chip memory	power-of-five FFT (data)	231 Kib
	power-of-five FFT (coeff)	687 Kib
	power-of-two FFT (data)	94 Kib
	power-of-two FFT (coeff)	72 Kib
	Total	**1'084 Kib**
Off-chip memory I/O	first data transpose (write)	64x4b @ 250 MHz
	first data transpose (read)	125x4b @ 128 MHz
	second data transpose (write)	125x4b @ 128 MHz
	second data transpose (read)	64x4b @ 250 MHz
	twiddle factors (read)	64x22b @ 250 MHz
	Total	**1920b @ 250 MHz** **1000b @ 128 MHz**

Table 12: Summary of FPGA resources required for the proposed architecture.

4.3 Prime Factor Algorithm FFT

Discuss the prime factor algorithm and how it improves on the two-dimensional FFT resource usage.

4.4 Tunable Filterbank followed by per-channel PFB

Describe, in detail, an architecture that closely follows what the present ALMA correlator uses to channelize its data and make a table or graph of estimated

Slice, LUT, DSP Slice, BRAM, etc. usage. In particular this would be a tunable filter-bank where each coarse channel is tunable to anywhere in the band (allowing overlap) then followed by a fine channelizer, likely just a PFB/FFT per TFB channel.

5 Complex-Gain Multiplication

In this section we derive the resources required to apply a complex-gain to the output of the channelizer, and then develop a suitable architecture for the implementation of such a subsystem. It is assumed that an independent magnitude and phase (or equivalently real and imaginary component) should be available for each spectral channel. The subsystem will thus provide a mechanism for performing various operations such as bandpass correction, fine delay control, beam phasing, etc[4].

5.1 Resource Requirements

The basic functionality of this subsystem requires (a) multipliers that apply the complex-valued gains to the spectral samples, and (b) memory for storing the complex-valued gains. In terms of FPGA resources,

$$R_{mul,cgain} = 4D \tag{12}$$

$$R_{mem,cgain} = 2bN \tag{13}$$

where

N	=	number of spectral points
D	=	number of parallel inputs
b	=	number of bits used for complex-gain representation
$R_{mul,cgain}$	=	number of (real-real) multipliers needed
$R_{mem,cgain}$	=	amount of memory needed in bits.

In terms of multipliers resource utilization is very low: for a sample rate of 16 GSa/s and FPGA clock of 250 MHz, $D = 64$. Memory requirements are much higher, although not prohibitively so, for $b = 18$ and 2^{23} spectral channels $R_{mem,cgain} = 288$ Mib. Since the gain coefficients will be accessed in a predetermined order and since usage is localized the use of off-chip memory is easily accommodated; in this case external access to storage may be needed so that coefficients can be set independently of the FPGA, and depending on constraints regarding the timing of updates coefficients may also need double buffering.

[4]Walsh demodulation, however, is excluded since that can be more efficiently implemented in the X-engine / beamformer.

6 Synchronization and Timing

Special consideration is foreseen for the handling of the timing signals coming from the central reference distribution stage. The following sections describes the scheme we propose for accomplish proper synchronization across the several signal processing engines

6.1 General picture

The aim of this section is to present a functional structure which processes incoming reference signals and produce stable feeds for the whole electronics, this will ensure signals are properly processed inside a specific timeframe. Figure 13 presents an overview of how this should be managed.

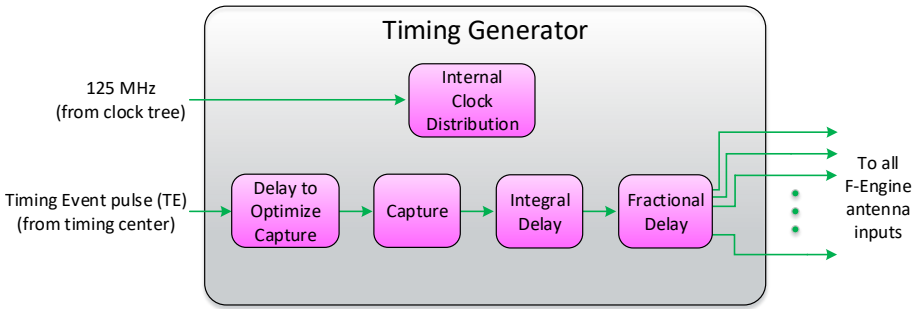


Figure 13: Timing management - general diagram.

The 125MHz signal generated from a very high stable reference is connected to an internal clock distribution scheme that in turn use this reference as a base to generate 250MHz system clock and its multiples required for the different stages in both F and X engines.

7 Corner-turn Packet Output

Describe the corner-turn output packetizer format. Also, estimate necessary resources needed for buffering and packetization.

8 Monitor and Control

This section will describe and draft a list of the envisioned monitor and control points needed for operating and maintaining the ALMA correlator. In addition an estimation of the needed bandwidth for the Monitor and control channel will be provided and a communication standard will be suggested. A safe margin of bandwidth will be suggested in order to allow further improvements and address features like node to node communication.

8.1 Points of interest

This is the summary of readable and writable points required to conduct the configuration, maintenance and monitoring of the F-engine and X-engine and all their supporting subsystems.

Read-Write	Name	Size (bytes)	Update interval	Real Time	Total Instances
R/W	Metaframe delay	3	1 [min]	No	288
R	Parity error counter	4	1 [min]	No	288
R	PLL status	1	1 [min]	No	1728
R	Statistics	60	1 [min]	No	288
R/W	Configuration	16	1 [min]	Yes	1152
R/W	Coarse delay	3	1 [min]	Yes	288
R/W	Fine Delay	1	1 [min]	Yes	288
R/W	Update Delay	8	1 [min]	Yes	288
R	Optical link status	4	1 [min]	No	1440
R	Temperature	4	1 [min]	No	1728
R/W	Walsh sequence	16	1 [min]	Yes	1152
R	SQL detector	4	1 [min]	Yes	288

Table 13: Readable and writable points

The instances of each point are calculated based on a single antenna and they are counted as 1 DRX, 1 F-engine and 4 X-engines across the 4 baseband pairs as a preliminary estimation.

Each read/write point will be stored in an offline database, which will be consulted for validation and troubleshooting purposes, the description of each one is defined below.

8.1.1 Metaframe delay

A proper calculation of the antenna delay in the transmission path should be accessible through a readable and writeable point, 3 bytes can hold up to 16,777,216 metaframe counts, if each count can cover 4[ns] of delay then the maximum delay we can cover is 0.067108864[s], which is a total distance of 20,118[km] at the speed of light. This point will be accessed through DRXs.

8.1.2 Parity error counter

A data integrity metric such as parity error counter will aid checking if a performance issue in the FFT conversion is due to problems in the incoming data. 4,294,967,296 counts can be registered with 4 bytes. This point will be accessed through DRXs.

8.1.3 PLL status

Timing status tracking of PLL and TE will be checked using this readable point, this point is considered to be accessed in both F and X-engine. With 1 byte we can store up to 8 timing status. This point will be accessed through DRXs, F-engines and X-engines.

8.1.4 Statistics

The statistical properties of the incoming signal will be read here, we will use this point to retrieve the histogram from a set of collected samples since this approach will allow us to reconstruct any statistical measure.

For accomplish this premise and considering we should be able to integrate results up to 1[ms], we foresee an amount of $1ms/(1/16GHz) = 1.6e7$ samples will be captured in 1[ms], in the worst case scenario all samples will fall under a single level from the 4 bit samples, a minimum of $log2(1.6e7) = 29.32$ bits will be required to store all counts in a single level, therefore it is advisable an usage of 30 bits to count occurrences for each level of a sample. In 4 bit samples there are 16 levels, then the size of this register will be $16x30 = 480$ bits or 60 bytes.

This metric can serve us to adjust digitizers parameters towards a low error from a theoretical Gaussian distribution and achieve the maximum efficiency of 99% for 4 bit samples. This point will be accessed through F-engines.

8.1.5 Configuration

For acceding and setting configuration of the interferometry mode, VLBI mode and a set of extra parameters to be defined in the future, 128 bits (16 bytes) will hold all the required setup to configure the entire correlator. This point will be accessed through X-engines.

8.1.6 Coarse delay

The coarse delay will able to be monitored and controlled using this point. at least 3 bytes are needed to set or collect this 64-samples granularity delay. It ranges from 0 to 524,287*64 samples delay, this point will be accessed through F-engines.

8.1.7 Fine delay

The sample-wide granularity sample will be managed using 1 byte word. It ranges from 0 to 63 samples delay. This point will be accessed through F-engines.

8.1.8 Update delay

According to the minimum bandwidth calculation in Section 8.2, the maximum delay rate is 1170 delays, which means a maximum of 57 delays should be handled in a single TE event. In order to properly manage this delay rate

under tightly timed conditions, an update register will contain at least 57 time slots that will increase or decrease a pre-configured delay initialization (coarse and fine delay). A 8 byte register will serve to allocate these 57 bits and an additional bit will control the increment/decrement direction of the fine delays. This point will be accessed through F-engines.

8.1.9 Optical link status

The optical power from DTS link will me measured and reported in this monitor point. Each count from the 4 bytes register will represent steps of 1 [nW]. This point will be accessed through DRXs and X-engines.

8.1.10 Temperature

Temperatures in several locations of the new correlator is meant to be one of the most critical variables for the health of the electronics. This point will be accessed through DRXs, F-engines and X-engines.

8.1.11 Walsh sequence

The access to the current Walsh sequence and the ability to provide a new one will be addressed with a R/W point of 128 bits wide (16 bytes). This point will be accessed through X-engines.

8.1.12 Square-Law Detector

A digital measure of the incoming IF signal will be obtained by reading this location. This point will be accessed through F-engines.

8.2 Communication channel

According to the project specifications for next generation correlator, it is foreseen that higher bandwidth for the communication channel will be required, the current approach for CAN bus is not sufficient to comply with the new bandwidth requirements, therefore a new bus communication is proposed to be used.

One approach for defining the maximum bandwidth is based on estimation of how often the delay events will be issued to the correlator (where the instrumental delay is deployed for time adjustment defined in steps of 1 sample). The delay to be compensated [**1999ASPC..180...11T**] is given by:

$$Tg(\theta) = \frac{Bcos(\theta)}{c}$$

where B is the magnitude of the baseline vector, θ is the angle made by the source position and the baseline and c is the speed of light. Since we want to know how often the instrumental delay must be updated, we must know the delay change ratio (or the derivative respect to the time).

$$\frac{dTg(\theta)}{dt} = \frac{dTg(\theta)}{d\theta} \times \frac{d\theta}{dt} = \frac{Bsin(\theta)}{c} \times \frac{d\theta}{dt}$$

Therefore, considering the worst case scenario (longest baselines, source close to the zenith, under a east-west projection), the maximum delay change ratio can be expressed as:

$$Max(\frac{dTg(\theta)}{dt}) = \frac{B}{c} \times \frac{d\theta}{dt} = \frac{Bsin(\theta)}{c} \times \frac{d\theta}{dt}$$

$$= \frac{300[Km]}{3 \times 10^5[Km/s]} \times 7.29 \times 10^{-5} = 7.3 \times 10^{-8}[s/s]$$

the above means it will be needed to update up to 1170 delays in a second based on a sample period of $62.5[ps]$ as the sample rate is assumed to be 16×10^9 samples per second).

Estimating bandwidth for the worst case scenario, the system should be able to execute delay change rates at a similar maximum delay change rate for the whole array (72 antennas) and the total packet size to hold delay values (24 bits) is 94 bits [**Corrigan08**]. The expected minimum required bandwidth is then:

$$MinBW = \frac{Max(\frac{dTg(\theta)}{dt})}{SamplePeriod} \times antennas \times packetSize$$

$$= \frac{7.3 \times 10^{-8}[s/s]}{62.5 \times 10^{-12}[s]} \times 72 \times 94[bits] = 7,918,560[bits/s]$$

The specified minimum bandwidth ($8M[bit/s]$ approximately) is higher than the maximum transmission rate as defined for CAN bus of $1M[bits/s]$. Moreover, a minimum bandwidth requirement for the new correlator would add a 30% extra for allocation of the rest of communications and to accommodate space for growing its complexity of the telescope over time, therefore the minimum new bandwidth is estimated in $10.4M[bit/s]$.

According to this, a well suited candidate to succeed CAN bus topology is Ethernet technology due to its physical attributes can nowadays hold transactions at a speed of $100M[bits/s]$ with low-cost commercial equipment, in addition, Ethernet possesses the ability to broadcast messages across the entire network, it is a standard in communication, allowing to analyze data by means of network sniffer tools for diagnostics purposes, it is natively integrated in FPGAs which enables the customization of communication protocols including automatic error checking.

2.4 Identify corner-turn platform

*Assigned to **Hickish**, & Primiani*

1. Back-plane vs network switch vs custom switch
2. F-engine platform interfacing capabilities
3. X-engine platform interfacing capabilities
4. Power density limitations

ALMA Correlator Study
WP2.4: Corner-turn platform

Jack Hickish & Rurik Primiani

July 2016

1 Introduction

The ALMA Correlator Study assumes an upgraded ALMA correlator will be of an FX architecture, widely employed for digital correlators in radio astronomy. Hence, the correlator will require a *corner-turn*, or data transpose, between the F and X stages. Such a transpose allows data processing to be parallelized on a per-antenna basis in the F-stage and a per-frequency basis in the X-stage.

In this document we consider the physical hardware used to implement the corner-turn. Broadly these can be placed in one of two categories – *actively switched* systems, such as Ethernet and Infiniband can dynamically route data from input sources to addressable endpoints. *Passive point-to-point* interconnect systems require each input and output to be connected via a passive link, such as a PCB trace, or copper or fiber cable.

We begin by detailing the top-level specifications which determine the requirements of the corner-turning system. Specifications are taken from work package 2.1 of the ALMA correlator upgrade report (Rupen et al., 2017).

2 Interconnect Specifications

The specifications relevant to the corner-turning system discussed in this document are:

Rupen spec.	Description	Value	Symbol
2	Number of antennas	72	N
5	BBC Bandwidth	8 GHz	B
4	Number of BBCs per polarization	4	n
7	Cross-correlation input bitwidth	8 (4+4) bit	w
14	Number of polarizations	2	p

We make the following further assumptions that different BBCs are effectively independent. That is, the complete correlator may be constructed from n distinct correlators, with no interconnect required between them.

Considering a correlator for a single BBC, we may consider the inputs of the corner turner to be N F-Engines. That is, each F-engine processes signals from a single dual-polarization antenna. Each F-Engine generates data at a rate $p \times B \times w$ bits/s. For the assumed ALMA correlator, this is 128 Gb/s.

In the interests of simplicity and platform agnosticism we consider the N corner-turner inputs to be physically separate. That is, we do not allow an input to the corner turner which requires multiple F-Engines to be co-located on common hardware, or otherwise share data. It should be immediately recognized that this requirement has significant implications for the corner-turner, as it sets the required number of inputs. In many real astronomical systems, multiple F-engines are implemented on common hardware, and thus may implement a subset of the total corner-turning operation internally. This is the case for systems built with CSIRO's RedBack board (Hampson et al., 2014) and ASTRON's Uniboard (Hargreaves, 2012), which are multi-FPGA boards which implement all-to-all connections between chips. In these systems, while there may be a large number of FPGAs, the number of boards requiring interconnection is significantly lower, substantially simplifying and reducing the cost of the central interconnect hardware. Nevertheless, we assume that all N correlator inputs are physically separate so as to minimally constrain the choice of F-Engine platform.

Having assumed that the corner-turner inputs comprise N separate 128 Gb/s streams, we further assume that input streams may be split over parallel interfaces in order to be practically feasible. For example, a 128Gb/s stream may be implemented using four parallel 40 Gb/s Ethernet interfaces, two 100 Gb/s interfaces, or some large number of low-bandwidth links.

The output data-rate of the corner-turner is the same as the input, i.e., $N \times 128$ Gb/s.

3 F-Engine interface

In this document we assume an F-Engine processes data from a single BBC for a single polarization of a single antenna. The total ALMA correlator has $N_f = N$ F-Engines. We assume nothing about the F-Engine interface except that it is capable of outputting a total of 128 Gb/s over f independent links. Where $f > 1$, we assume that the multiple outputs contain sub-bands of the total processed bandwidth. If necessary this allows the downstream corner-turner and correlator to be constructed from f clones of a smaller system processing a bandwidth $\frac{B}{f}$.

4 X-Engine interface

We assume that a complete correlator system comprises N_x X-Engines, each processing a subset of the total correlator bandwidth. The number of X-Engines is determined by the computational performance of a single unit, and need not

be related to the number of F-Engines, N. There is no assumed requirement on the X-engine interface, other than a single X-Engine is capable of sinking its fraction of the total system bandwidth: $\frac{pNBw}{N_x}$. This may be achieved via a single wide-band link, or via x multiple parallel links.

Where the connection between the N_f F-Engines and N_x X-Engines is not direct (because, for example, it is mediated by an Ethernet switch) there is no requirement that the protocol of the F- and X-Engine interfaces should be the same.

5 Interconnection Systems

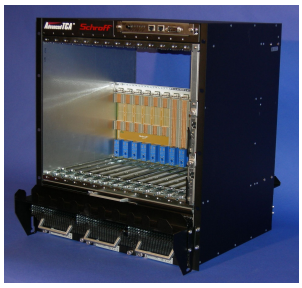
Interconnection systems may be divided into two classes. Actively switched systems can dynamically route data from a source to any of several endpoints. These systems include Ethernet, Infiniband, and some PCI-Express based motherboards/backplanes. Passively-routed systems simply provide point-to-point connectivity from sources to endpoints. Examples of these systems are simple backplane meshes, and point-to-point connections made with optical fiber or copper cabling. A very brief overview of the applicability of these systems to an upgraded ALMA system is given below.

5.1 Passive Point-to-Point Interconnect

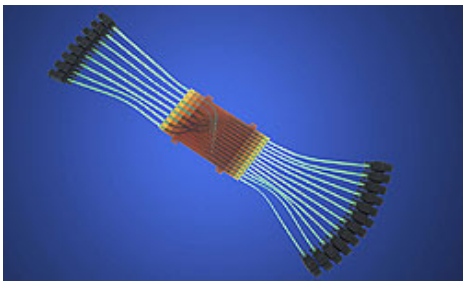
LVDS Copper Cabling The present ALMA correlator interconnect uses 16384 LVDS twisted-pair cables operating at 250 MHz to connect the *station cards* (effectively equivalent to F-engines) to the *correlator cards* (similarly equivalent to the X-engines). This represented "the greatest design challenge in the system" Escoffier, R. P. et al. (2007). With the increased specifications of the next-generation ALMA correlator a corner-turn implementation using the same technology would see the total number of cables increase more than four-fold, mainly driven by the doubling in bandwidth and sample bitwidth of the upgraded system. Increasing the per-lane speed by a factor of two or even four potentially reduces the total number of cables but the complexity of such a system is highly undesirable in light of other available options.

Copper Backplane Off-the-shelf standards for providing all-to-all connections between microprocessors exist in the form of industry-standard backplanes. The most promising copper backplane standard is the Advanced Telecommunications Computing Architecture (ATCA). The latest standard, PICMG 3.1, supports 40 and 100 Gb/s connections. ATCA enclosures can be purchased off-the-shelf, for $\sim$ 10 k$, and provide all-to-all connections between up to 16 computing cards (Figure 1(a)).

With a 16-node full-mesh interconnect operating at 100 Gb/s, an ATCA backplane is sufficient to corner-turn $16^2 x 100$ Gb/s – around 25 Tb/s. This is sufficient for at least one ALMA BBC, which has a bandwidth of $N \times 128$ Gb/s – about 9 Tb/s. However, this assumes the processing required by the correlator



(a) An enclosure with copper mesh interconnect provided by an ATCA standard backplane. This backplane supports 40 Gb/s all-to-all connections for up to 14 cards.



(b) Molex FlexPlane^{TM} fiber circuitry provides user-customizable fiber-based mesh interconnect.

(c) Actively switched interconnect, provided by COTS Ethernet switches, with industry standard high-speed ports, operating at 100 Gb/s

Figure 1: Three interconnect options based on current technologies.

can be accommodated in the available 16 ATCA cards. This requirement While some correlator realizations may satisfy this specification, it is not met in the assumed case of N independent F-Engines. In general it may be necessary to externally mesh together multiple such units, resulting in undesirable complexity and cost. Further, requiring computing units to be ATCA-compatible rules out many off-the-shelf processing platforms, greatly increasing the likelihood that they must be custom-designed, with significant associated NRE.

Fiber Circuitry Fiber circuitry (for example, Molex FlexPlane$^{\mathrm{TM}}$) represents a promising point-to-point interconnect solution and is already being used in astronomy applications (Hampson et al., 2013). For a one-off NRE fee of $\sim 10\,\mathrm{k\$}$ custom fiber-based interconnection circuits can be fabricated, providing practically any routing of inputs to outputs (Figure 1(b)). Fiber circuits can be manufactured in a variety of packages, either protected between layers of FR4, or on flexible substrates. These can be connectorized with standard multi-fiber push-on connectors such as MTP to provide short-, mid-, or long-range fiber runs. Provided the processing nodes at each end of such a system have adequate independent IO paths to drive the required number of fibers, fiber optic circuitry is a very cost-effective way of providing interconnect. However, it should be noted that this requirement potentially limits the applicability of fiber-circuits to systems involving custom processing platforms.

In the most general case, a fiber-circuit interconnect solution is an $N_f \times N_x$ all-to-all connection mesh between F-Engines and X-Engines. If it is not possible to manufacture the interconnect in a single circuit, the interconnect can be provided by multiple smaller circuit assemblies, with an associated increase in cabling complexity.

It is possible that an ALMA upgrade may employ F-Engines based on processors with many independent IO paths (eg. FPGAs) and X-Engines based on processors with a small number of high bandwidth paths requiring a high-level transmission protocol such as Ethernet (eg. CPU/GPUs). In this scenario, a potentially attractive design is to loop the data back onto the F-Engines, where it can be reformatted as Ethernet (or other) data streams and transmitted to X-Engines (Figure 2). Such a design would make it relatively straightforward to interface with generic processing platforms. Such a configuration may also be adopted with the ATCA backplane, subject to the limited number of cards supported by ATCA enclosures.

5.2 Active Switching

PCI-Express PCI-Express is a common standard for connecting many processing boards via a backplane type configuration supporting transfer speeds up to 125 Gbps per endpoint (for Gen3 with 16 lanes). Additionally the standard allows data transfers between slaves using bus mastering. All PCI-Express endpoints, however, must connect to either a root complex or a switch and given these devices with 16 or more endpoints are rare or non-existent we will not consider this technology for an ALMA upgrade.

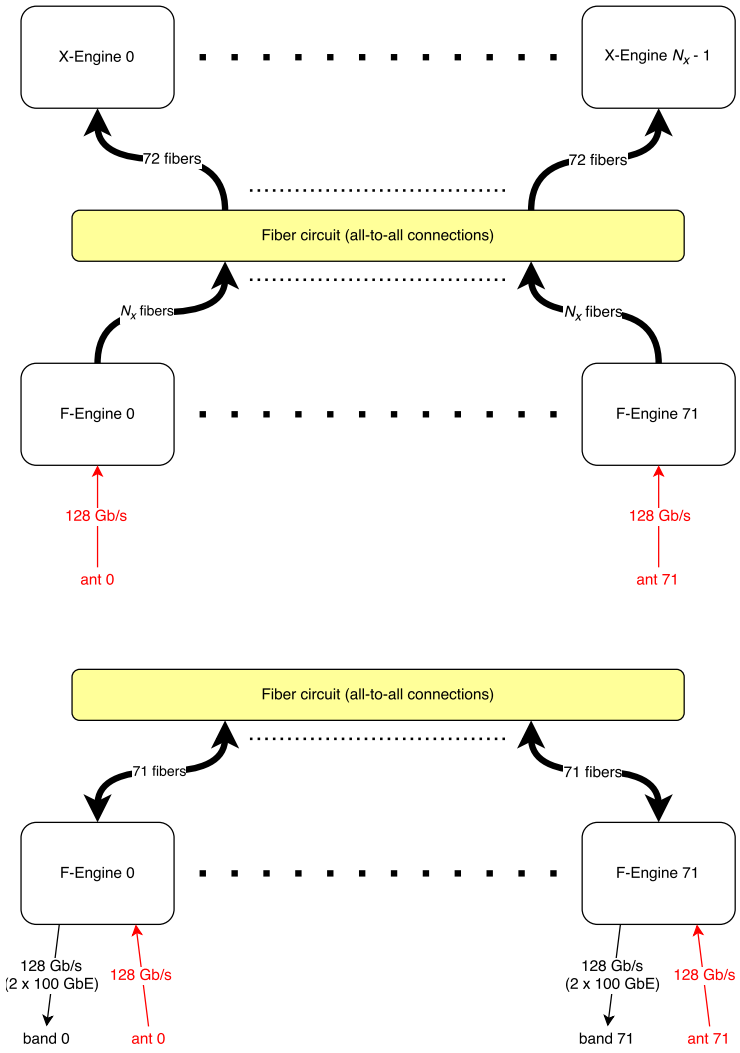


Figure 2: A single fiber-circuit may be used to connect F- and X-Engines directly, or data may be shuffled between F-Engines such that the ultimate output is a data stream using a higher-level protocol such as Ethernet.

Rapid IO Another popular data transmission standard is RapidIO. This is commonly used in high-performance computing and data centers. RapidIO supports both backplane- and switch-based interconnect systems with sub-microsecond latency transfers at speeds of up to 160 Gbps per port. However the availability of switches with greater than 16 ports is limited are generally only available in the form of chips that would need to be designed into a PCB backplane.

Ethernet The Ethernet protocol is ubiquitous in consumer and industrial computing systems. The use of Ethernet switches to provide ALMA interconnect has a variety of attractive attributes:

- No hardware NRE.
- Industry-standard interface, widely supported by commodity hardware (eg. off-the-shelf FPGA boards, CPU/GPU platforms).
- Extremely tolerant to changes in F- and X-Engine implementations or changes to number of antennas.
- Trivially supports hardware testing via CPU-driven test-vector injection.

Though the specification and cost of available Ethernet technology at the time of deployment of a new correlator is uncertain, one can demonstrate the feasibility of an Ethernet corner-turn solution based on 2016 technology and assume that future solutions will be cheaper and denser.

We hypothesize a system where the F-Engine output uses 100 GbE links, with $f = 2$ independent interfaces, each carrying $\frac{128}{2} = 64$ Gb/s. For the purposes of this example, we let the number of X-Engines, $N_x = 48$.

Such a system requires, for each of the four BBCs, two duplicates of a corner-turner with 72 F-Engine inputs, and 48 X-Engine outputs. Thus, the corner-turner is a switch with at least $72 + 48 = 120$ 100 GbE ports. Such switches are available off-the-shelf today. For example, the Arista 7508R with up to 288 100 GbE ports.

Alternatively, one can construct a larger switch from smaller modules, which can have more predictable performance for the all-to-all corner-turn systems required by correlators. In this case, the same system can be achieved with 6 individual 32-port 100 GbE switches (Figure 3). Since the total ALMA correlator requires $fn = 8$ duplicates of this system, 48 switches are required. The per-switch cost at current pricing is \$7–15k, depending on the switch brand and operating system, giving a total cost of \$336–720k. Transceivers also contribute a significant cost. Low cost short-range "direct-attach" copper cables are available and likely are sufficient for switch-switch interconnect. However, links between the F- and X-Engines and the switches are potentially best served by fiber connections. At current pricing, third-party short-range (100GBASE-SR4) links are approximately \$220 each[1], or approximately \$500 for a terminated active optical cable. The number of these cables required for each of the eight

[1] Pricing obtained from Fiber Store (fs.com)

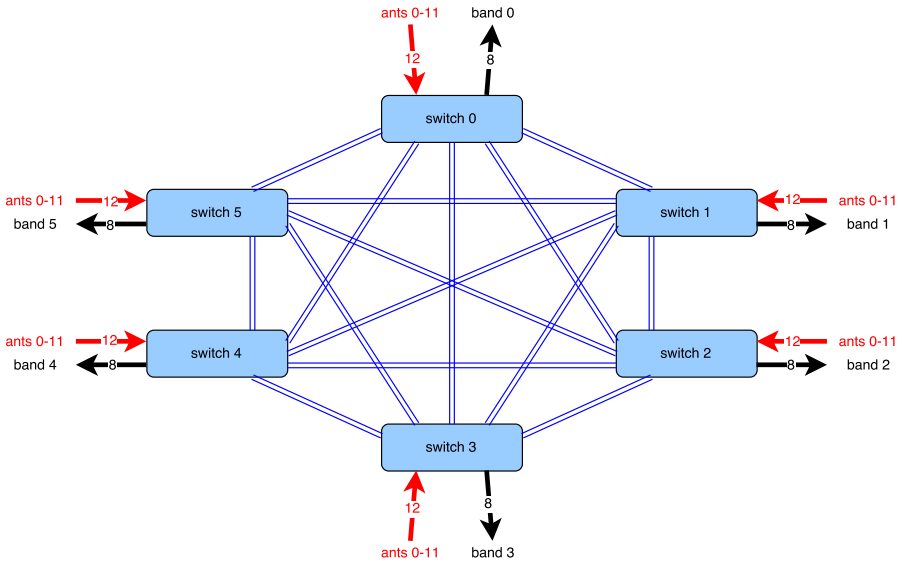


Figure 3: A potential interconnect implementation based on six 32-port 100 GbE switches. The six switches are interconnected with all-to-all connections providing 200 Gb/s bi-directional IO between switches. Similar architectures are possible with larger number of switches, should more ports for X-Engines be required.

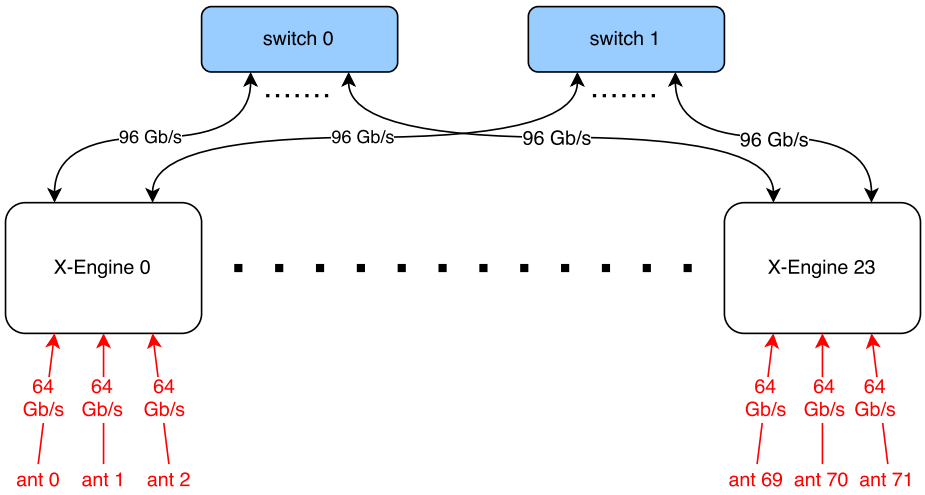


Figure 4:

corner turning systems is $N_f + N_x = 120$, resulting in a current cost of approximately \$60k per corner turner, or \$480k in total. While this report has avoided relying on projecting future hardware prices, here we note that an assumption that at the time of deployment 100 Gb transceivers will have a price similar to that of current 40 Gb transceivers results in a total cost closer to \$100k.

It should be noted that cost is very strongly affected by overall system architecture. For example, significant savings can be realized by assuming a different number of X-Engine nodes, and by utilizing some of the optimizations discussed in (McMahon et al., 2007), which reduce the number of switch ports required by a system by utilizing them in a bidirectional fashion. An example of such an approach is given in Figure 4, which assumes $N_x = 24$, and provides an interconnect solution for one half of a BBC using just two switches. This would result in a total switch cost of \$112 – 240k. However, such an architecture makes various assumptions about the X-Engine design: that is provides at least five 100 GbE ports, and that F-Engine data can be routed through an X-Engine processor without impacting the X-Engine's performance. These assumptions likely are not true for X-Engines which are IO limited, such as those based on GPUs.

Though not strictly part of the corner-turner requirements of the system, one may wish to rout ADC data through the same switches used for the F-X interconnect. This provides the ability to deploy redundant F-processors on the network to be used in the case of hardware failure, and allows dynamic routing between antenna signals and F-Engines which may be useful when first testing and deploying the system. In such an architecture, simulators on the network could be used to take the place of antennas and provide a mechanism for testing of the whole processing chain. An example of such a configuration is a system

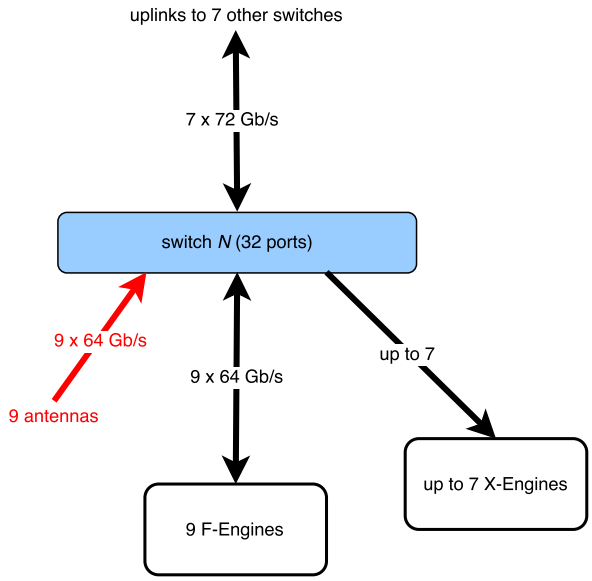


Figure 5:

similar to Figure 3, but utilizing eight 32-port switches to provide extra ports to accommodate data streams from the antennas (Figure 5). The extra switches increase the cost of this system to $448–960k, with cabling costs assumed to be approximately the same (we consider the antenna-switch links to be outside the scope of corner-turn costing, since links from antennas will be required in some form regardless of corner-turn implementation.

6 Reliability

In this report we make no attempt to quantify the reliability of the proposed systems, though we note that modern switches quote mean time between failure ratings of $O(100,000)$ hours. Passive interconnect systems are likely to have longer lifetimes, though this does not take into consideration the active drivers (eg. fiber optic transceivers) of such systems. Failure modes and their effects should be studied further before committing to any interconnection architecture. In particular, the support of the interconnect for supporting "hot-spare" processors may be deemed a critical requirement of the system.

7 Conclusions

The final choice of interconnect technology used by a next-generation ALMA correlator will need to be made in light of a system-engineering overview of the

instrument as a whole. Certainly the choice of F- and X-Engine platforms will constrain the available interconnect choices.

If large engineering budgets are to be dedicated to developing custom platforms on which to implement the correlator's signal processing, it way well be the case that integrating support for a fiber circuit interconnect or ATCA enclosures is the most appropriate design decision. Custom platforms may also allow some of the corner-turning requirements to be offloaded to this hardware. However, if a choice is made to adopt general-purpose COTS compute platforms, prioritizing the value of low-cost processing nodes such as single-chip FPGA or CPU/GPU platforms, an NRE-free Ethernet switch interconnect is likely to represent a cheaper total cost. Given that an Ethernet-based interconnect solution is already feasible at the scale of the proposed ALMA correlator, and likely represents a relatively small part of the total hardware budget, we believe that this technology is the preferred choice, given the uncertainties in the other aspects of the correlator design. Choosing an Ethernet fabric interconnect maximizes the flexibility of the digital backend. Furthermore, should hardware development of a future correlator commence, such a choice would make it easy to prototype an effectively production-ready subset of the complete ALMA system and provides a clear path for staged deployment.

References

Escoffier, R. P., Comoretto, G., Webber, J. C., Baudry, A., Broadwell, C. M., Greenberg, J. H., Treacy, R. R., Cais, P., Quertier, B., Camino, P., Bos, A., and t, A. W. Gun (2007). The ALMA correlator. *A&A*, 462(2):801–810.

Hampson, G., Brown, A., Neuhold, S., Bunton, J., Macleod, A., Tuthill, J., and Beresford, R. (2013). Askap advancements in beamformer and correlator optical backplane technology. In *2013 US National Committee of URSI National Radio Science Meeting (USNC-URSI NRSM)*, pages 1–1.

Hampson, G. A., Brown, A., Bunton, J. D., Neuhold, S., Chekkala, R., Bateman, T., and Tuthill, J. (2014). Askap redback-3; an agile digital signal processing platform. In *2014 XXXIth URSI General Assembly and Scientific Symposium (URSI GASS)*, pages 1–4.

Hargreaves, J. E. (2012). Uniboard: generic hardware for radio astronomy signal processing. *Proc. SPIE*, 8452.

McMahon, P., Langman, A., Werthimer, D., Backer, D., Filiba, T., Manley, J., Parsons, A., and Siemion, A. (2007). CASPER Memo 17: Packetized FX Correlator Architectures. Technical report.

Rupen, M. et al. (2017). ALMA Correlator Upgrade Study: 2.1 Scientific Requirements & Specifications. Technical report.

2.5 Identify DSP X-engine platform (similar to step 2)

Assigned to Blackburn, Hickish, ***Greenhill*** *& Primiani*

1. all the steps in section 2.2
2. For GPUs: future fixed-width (e.g. 8-bit) computation vs floating point
3. For GPUs (and CPUs?): benchmarking of common tasks (e.g. XGPU, DifX, etc.)

Digital Correlator and Phased Array Architectures for Upgrading ALMA

WP2.5: DSP X-engine platform

Greenhill, Blackburn, Hickish, Primiani, Young

May 12, 2017

1 Introduction

1.1 X-engine Scope

The primary task of the X-engine platform will be to perform the element-wise multiplication of the received spectra for each pair of antennas, to accumulate this result for specified times and frequency channelization, and to send the output on to the next stage in the digital pipeline. It is assumed that the X-engine receives the output of a single-stage channelization of the digitized band. Related operations on the X-engine platform are (i) execution of a transpose operation on data within Ethernet packets (cf. the "coarse" transpose operation achieved by routing of packets by the network layer), and (ii) implementation of beamforming using calibration information provided externally (Work Package 2.7; =0mu plus 3mu**wp2d7**. The latter is motivated by the availability at the X-engine stage of Nyquist-sampled data that have been "corner turned" and similarity in the high-speed computing hardware needed for multiplication and summation, spectral channel by channel.

1.2 Baseline Requirements

The most relevant items listed in the main specifications table are repeated here in Table 1. In the subsequent sections these specifications are used to calculate the compute and I/O rates for the most intensive operation, pairwise multiplication and accumulation. For this, there are two observing modes specified, distinguished by accumulation intervals in time and frequency. Prioritizing simplicity of correlator pipeline configuration, the study treats a single case where the input from the F-engine is the same for both modes, and X-engine processing is distinguished only in operation of the accumulator stage.

Table 1: Project Specifications Most Relevant to the X-engine Platform

Item	**Requirement**	**Impact**
Antennas	72	Number of baselines
Number of BBC pairs per antenna	4	Number of multiplies/baseline (two total)
Sample format	4 bit	Input data and processing rate; 16 GS/s
Maximum spectral points per BBC	$\approx$ 8M	Number of multiplies/baseline
Number of configurable sub-bands	16	Bookkeeping / data routing
Polarization products	2 or 4	Number of multiplies/baseline; adopt 4
Time integration (s)	0.001 (a/c), 0.016 (x/c) 1.6 (a/c), 29 (x/c)	Data output rate
Peak data rate after accumulation (GB/s)	100 total	Frequency resolution vs time resolution trade-off; bookkeeping / additional data manipulation
Number of subarrays	6	Bookkeeping / data routing implications
Sideband separation / suppression	Embedded in LO system and FX design	Bookkeeping / additional data manipulation
VLBI subarrays	2	Bookkeeping / data routing implications
Phased-array beams	4	Number of beamformer sums

1.2.1 Compute rate

The number of complex-complex multiplications that need to be performed per FFT window is[1]

$$N_\times = (N_{bl} + N_{ant}) \times N_{bbc} \times N_{ch} \times N_{pol} \tag{1}$$

where

N_{bl} = Number of baselines
N_{ant} = Number of antennas
N_{bbc} = Number of BBC's per antenna per polarization
N_{ch} = Number of spectral channels per BBC
N_{pol} = Number of polarization products.

Assuming the FFT window is $T_{fft} = 2 \times N_{ch}/f_s$, the number of complex-complex multiplications needed per second is

$$R_\times = N_\times/T_{fft} = \frac{1}{2}(N_{bl} + N_{ant}) \times N_{bbc} \times N_{pol} \times f_s \tag{2}$$

where f_s is the sampling rate.

The output of each multiplication is accumulated in each FFT window so that the number of complex-complex additions[2] equal the number of multiplications, $N_\times = N_+$ and $R_\times = R_+$.

The compute rate is independent of the trade-off between spectral resolution and time resolution in the correlator output. The rate of complex-complex multiplications is

$$R_\times = 3.36384 \times 10^{14} \text{ s}^{-1} \tag{3}$$

per second.

1.2.2 Input / Output Rate

The input data rate in bits-per-second is

$$R_{in} = b_{in} \times N_{ant} \times 2N_{bbc} \times f_s \tag{4}$$

where

b_{in} = Input data bit resolution

and the factor of 2 is due to two polarizations produced by each antenna[3]. The input data rate is equal to

$$R_{in} = 36864 \text{ Gbps.} \tag{5}$$

[1] This includes the auto-correlations which are assumed to produce 4 polarization products.

[2] The output of auto-correlation products are of course real-valued, but the cost is dominated by the cross-correlation products so that this can be neglected.

[3] An additional factor equal to a half would account for the possibility that both sidebands are embedded in the same data.

The output data rate in bits-per-second is

$$R_{out} = R_{out,a} + R_{out,c} \tag{6}$$

where

$$R_{out,a} = 1.5 b_{out,a} \times N_{ant} \times N_{bbc} \times N_{ch} \times N_{pol} / T_{int,a} \tag{7}$$

$$R_{out,c} = 2 b_{out,c} \times N_{bl} \times N_{bbc} \times N_{ch} \times N_{pol} / T_{int,c} \tag{8}$$

and

$$\begin{array}{rcl} b_{out,a},\, b_{out,c} & = & \text{Output bit depth for auto- and cross-correlations} \\ T_{int,a},\, T_{int,c} & = & \text{Integration period for auto- and cross-correlations.} \end{array}$$

The factor 2 in the expression for $R_{out,c}$ accounts for the fact that the cross-correlation products are complex-valued, similarly the factor 1.5 in the expression for $R_{out,a}$ accounts for the fact that co-polarized auto-correlations products are real-valued whereas the cross-polarized products are generally complex-valued.

Since the total data rate for the X-engine output is limited to 100 GB/s, the number of spectral channels and the integration period satisfy the relation,

$$R_{out} \leq 8 \times 100 \times 10^9. \tag{9}$$

Substituting the values from Table 1 into this relation yields, for the shortest auto- and cross-correlation integration times, and assuming[4] $b_{out,a} = b_{out,c} = 32$,

$$N_{ch} \leq 3654. \tag{10}$$

At the other extreme, for the maximum number of frequency channels, and assuming $T_{int,c} = 16 T_{int,a}$, the shortest allowable auto- and cross-correlation integration times are,

$$T_{int,a} \geq 1.649 \text{ s} \quad \text{and} \quad T_{int,c} \geq 26.376 \text{ s}. \tag{11}$$

These figures are consistent with the specifications in Table 1.

1.2.3 Memory

The memory required to accumulate the correlator products is equal to

$$M_{acc} = M_{acc,a} + M_{acc,c} \tag{12}$$

where

$$M_{acc,a} = 1.5 b_{out,a} \times N_{ant} \times N_{ch} \times N_{bbc} \times N_{pol} \tag{13}$$

$$M_{acc,c} = 2 b_{out,c} \times N_{bl} \times N_{ch} \times N_{bbc} \times N_{pol}. \tag{14}$$

For the maximum number of spectral channels the total memory required is equal to

$$M_{acc} = 20880 \text{ Gb}. \tag{15}$$

[4]It is yet to be determined whether 32 bits is an appropriate representation for the correlator output.

1.3 Beamforming

Beamforming entails a weighted sum of Nyquist-sampled data from all antennas, channel by channel at the Nyquist rate, which unlike correlation is computationally a relatively low intensity operation, O(N_{ant}). An inverse Fourier transform of the data provides a fast-sampled voltage time series. Where the bandpass has been divided among computing engines, the frequency domain data must be joined before the transform in order to achieve a sample rate on the order of $1/f_s$.

1.3.1 Compute rate

To form one beam requires one multiplication per data point received in the X-engine, and a sum over all antennas in the beamformed array. The total number of multiplications for N_{beam} independent beams is,

$$N_{\times}^{beam} = N_{beam} \times N_{ant} \times N_{ch} \times N_{bbc} \times N_{pol}/2, \tag{16}$$

whereas the total number of additions is,

$$N_{+}^{beam} = N_{beam} \times (N_{ant} - 1) \times N_{ch} \times N_{bbc} \times N_{pol}/2. \tag{17}$$

The factor of a half accounts for the fact that the beams are only formed for two polarizations, not four. At the highest spectral resolution the compute rate in terms of complex-multiplications is,

$$R_{\times}^{beam} = 7.3728 \times 10^{13} \text{ s}^{-1} \tag{18}$$

per second, roughly 21% of that required for cross-correlation only.

1.3.2 Output rate

Assuming that the beamformed output is requantized to b_{beam} bits, that is $2b_{beam}$ bits for each complex-valued output, the total phased array output rate is,

$$R_{out}^{beam} = 2b_{beam} \times N_{beam} \times N_{bbc} \times N_{pol}/2 \times N_{ch}/T_{fft}, \tag{19}$$

or

$$R_{out}^{beam} = 4096 \text{ Gbps}, \tag{20}$$

which is roughly five times greater than the output for cross-correlation only.

1.3.3 Memory

Since the beamformed signals are continuously streamed the additional memory needed for execution of beamforming is that required to store beamformer weights, one per beam per ant-pol-BBC,[5] which is negligible relative to the memory available to support as a matter of course accumulation of correlation

[5] Bandpass shaping is assumed to be performed further upstream within the F-engine.

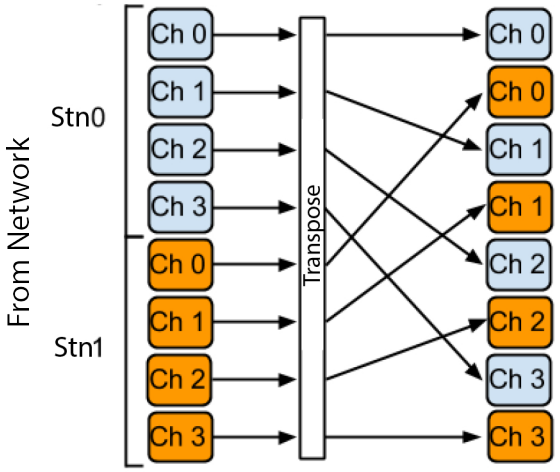


Figure 1: Corner turn completion by the X-engine (Kocz et al. 2014) via reordering of data for different frequency channels received from each F-engine (station) to be adjacent. The blue and red blocks indicate frequency channels from different F-engines.

products. However, if weights are to be applied in real-time to the data from which they were derived, then the time required to compute these weights outside of the X-engine determines the memory buffer size inside the X-engine, which may be considerable. For thus study, we assume that weights will be applied in a trailing sense, i.e., computed but applied to data close in time to that from which weights were estimated. Impact on algorithms and introduction of systematics are discussed in following sections.

2 Transpose operations

For packetized data flow, the corner turn is executed in two operations. The network layer initiates a corner turn or transpose operation by sorting at the packet level in the process of conveying data from the F to the X stage. Each packet contains data for one station and multiple frequency channels. Where there are multiple computing nodes, packets for any particular range of frequency channels are brought to an assigned node. In turn, payload data must be reordered after X-engine capture, so that for each frequency channel, data for all antennas are adjoining in memory and thereby ready for pairwise multiplication (correlation) or addition (beamforming). See Figure 1. In principle, the "small transpose" can instead be completed in the F-engine. For the study, we assign the calculation to the X-engine so as to simplify the anticipated firmware.

The transpose operation is memory and memory bandwidth intensive. For off the shelf hardware, it has been demonstrated with CPUs and FPGAs. While the problem is readily parallelizable by output memory location, which favors

GPU execution and promises high speeds, it will require two developments (i) implementation of the algorithm the effects byte-wise manipulation of memory locations to suit low-precision data (i.e., 4+4 or 8+8 complex values) and (ii) management of memory page constraints among thread blocks and streaming multiprocessors.[6] We assume that, going forward, low-precision transpose operations are part of standard linear algebra libraries, driven by the algorithmic needs of other (larger) communities, such as Deep Learning. In this case (ii) is not separate from (i). However, that aside, the option of CPU execution remains viable because Nvidia system-on-chip (SOC) products (see later discussion in this section) include integrated multi-core ARM processors that share GPU memory, obviating the need for an explicit memory copy to the GPU.

3 Platforms

3.1 FPGA Resource Analysis

The goal in this resource analysis is estimation of the number of FPGAs needed to process a single ALMA BBC-pair (i.e. 8 GHz, full-stokes). The well-proven CASPER windowed X-engine architecture =0mu plus 3mu[**parsons08**], which has been optimized for 4-bit input data =0mu plus 3mu[**hickish14**], provides a template. The analysis focuses on counting DSP resources and assumes that the architecture will not be limited by BRAM or logic resources. This is reasonable for sane implementation parameters, such as having modestly sized accumulation windows, though detailed analysis and/or trial implementations are required prior to settling on a specific realization of the design. This document assumes the use of Xilinx FPGAs, since these are most readily supported by the CASPER X-engine design. However other vendors (Altera likely being the only alternative) may also be used, likely at similar cost.

3.1.1 Correlation resources

Having chosen the CASPER windowed X-engine as an ansatz, counting resources can be calculated in a straightforward manner. Firstly, the X-engine bandwidth, B_x, processed by an X-engine module on an FPGA running with clock rate F_c is given by:

$$B_x = \frac{F_c}{N_{ant}} \,. \tag{21}$$

Such an X-engine is formed from a number of separate taps. Each tap computes 4 complex multiplies (one for each correlation product) for a pair of antennas. The number of taps needed in a complete module, N_t is given by:

$$N_t = \frac{N_{ant}}{2} + 1 \,. \tag{22}$$

[6] "Thread block" and "streaming multiprocessor" are borrowed from the terminology of Nvidia GPU architecture, but the concepts are not unique to Nvidia systems.

DSP Slices The number of DSPs used by a single X-engine module, D_x is given by $4N_t$, since it takes a single DSP slice to perform a 4-bit complex multiply.

Block RAM The number of block RAMs – discrete, dedicated memory elements provided by FPGAs – used by a complete X-engine design, R_x, is implementation specific. However, it is usually possible to achieve:

$$R_x \approx \frac{D_x}{4}\,, \tag{23}$$

provided that at least 4 X-engines may be implemented on a single FPGA =0mu plus 3mu[**hickish14**]. For currently available FPGAs, and the ALMA specifications, this is an easy requirement to meet. For the DSP-optimized FPGAs models likely to be most appropriate for the ngALMA project the number of 18 kb block RAMs provided is by Xilinx is approximately half the number of DSP slices. Thus, block RAM is not likely to be the limiting factor in an FPGA-based correlator.

Input Data Rate The data-rate input required to feed such an X-engine module, I_x, is given by:

$$I_x = 2B_x N_{ant} b_{in} = 2F_c b_{in}\,, \tag{24}$$

where b_in is the bitwidth of a (complex) input sample, and the factor of 2 accounts for the two polarizations input from each antenna.

Output Data Rate The output data rate is artificially capped at 100 GB/s in this document, and is limited by the capabilities of the post-correlation data recording and archiving systems. Here we simply note that any correlator system will reduce total data rate, and an FPGA-based X-engine can be chosen such that it has symmetric input and output data-rate capacities. Thus, the data output rate is automatically satisfied by any sane correlator implementation, even factoring in the output requirements of a hypothetical beamforming system.

Input Buffering A buffer is required to read data from the corner-turner and collate it ready to be read into the X-engine(s). The per-X-engine input and output data rates of this buffer are the same as the per-X-engine input rate, I_x. The minimum size of this buffer, D, is implementation specific, but for resource estimation purposes here it is assumed that the buffer is sufficient to store 4 packets from each antenna. That is, for packets of size P bytes, it is:

$$D = 4PN_{ant}\,. \tag{25}$$

It is assumed that this buffer may be shared over multiple X-engines. In practice this is likely the case, though one-buffer is probably required per independent input-stream. I.e., if a 160 Gb/s input stream is split over two 100 GbE

links, two buffers will be needed. If split over four 40 GbE links, four buffers are required.

There are two ways to implement this buffer. The first is with on-chip block RAM resources. These distributed, parallel RAM elements can easily meet the I/O requirements of the buffers, but are limited to kB - Mb sizes. The second is with off-chip (or off-die) memory modules, such as DDR5, High Bandwidth Memory (HBM), Hybrid Memory Cube (HMC), or other technologies. These can acheive buffer depths of ∼GB, but are lower bandwidth.

For the purposes of the X-engine, with a packet size, P, of 8 kB (almost the maximum allowed by Ethernet Jumbo packets), $D = 2.25$ Mb. This is feasibly handled by the on-chip *UltraRAM* resources provided by the latest generation of Ultrascale+ Xilinx FPGAs.

If additional buffering is required, for example to facilitate buffering of O(seconds) of beamforming data, then off-chip memory will be required. Assuming a calibration buffer of t seconds is required, the buffer depth per X-engine will be $I_x t$ bytes.

Output Buffering The output buffer is required to store the cross-correlation matrix for the array. Data are averaged into this buffer for one integration period, before being read out to a downstream processor. The total size of this buffer, distributed across multiple X-engine nodes, is given by the product of the correlator output word size (assumed to be no greater than $32 + 32$ bits), the total number of baselines ($\sim 4\frac{N_{ant}^2}{2}$) and the number of frequency channels in the system – approximately 8 Million, in the most demanding case. Thus, for ngALMA, the total output buffer in the most demanding case is given by:

$$D_o = 64\,\text{bits} \times 4\frac{72^2}{2} \times 8 \times 10^6 \approx 664\,\text{GB}\,. \tag{26}$$

For even modest numbers of X-engine processors, this is likely to amount to only tens of GB of memory per node, even if a factor of 2 is allowed to facilitate double buffering (i.e., accumulating into one buffer while reading from another). It is also the case that appropriate choice of buffering mechanisms upstream of the X-engine can lower the output requirement – essentially by permitting the X-engine to output each frequency channel after correlation, rather than attempting to buffer complete correlation spectra. On the timescale of ngALMA, and even today, this is not likely to be a limiting factor in platform choice. It is especially noted that this buffer need be of only modest speed, since the goal of the X-engine DSP architecture is to reduce the data rate which must be absorbed by the final accumulation buffers. RAM requirements for output buffering purposes are not considered further here, since they are very sesitive to specific system-level implementation of the correlator.

3.1.2 Total Resources

In building a full-bandwidth cross-correlation system, one instantiates multiple X-engine modules over multiple FPGAs. Given the parameters of ngALMA as

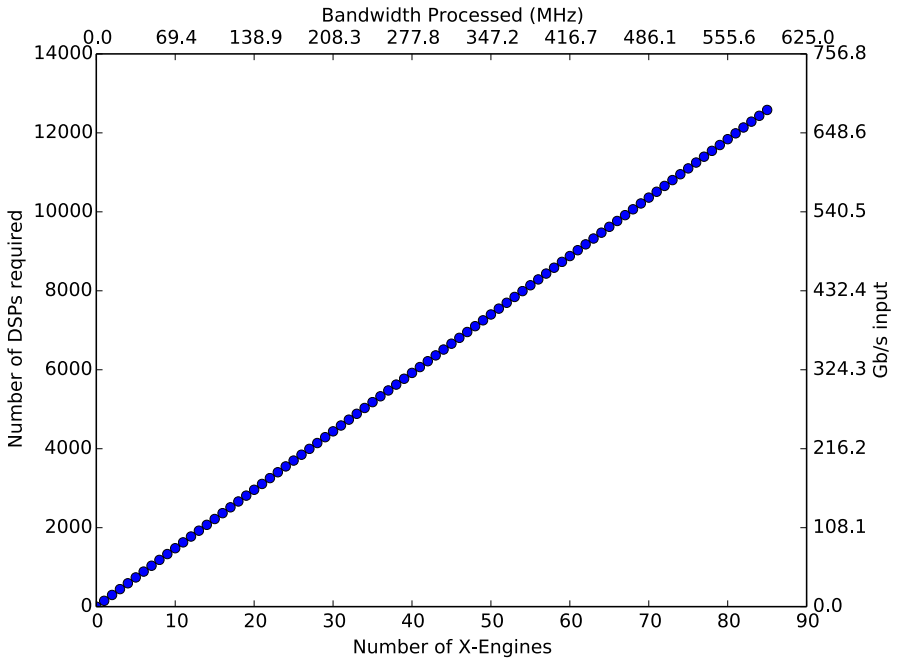


Figure 2: Number of DSP slices and input data required by an FPGA X-node as a function of bandwidth processed. FPGA clock-rate, F_c is assumed to be 500 MHz.

assumed in this document (i.e., $N_{ant} = 72$, $b_{in} = 8$), and assuming an FPGA clock rate, F_c, one can compute the number of computation resources and input data-rate of a correlator as a function of the number of X-engine modules – or, equivalently, the total bandwidth processed. Figure 2 shows this relationship for a correlator processing up to 600 MHz of bandwidth with $F_c = 500$ MHz. 500 MHz is chosen as a conservative estimate, with current generations of FPGAs being used by the SKA CSP consortium already demonstrating capabilities of > 700 MHz =0mu plus 3mu[**carlson17**].

The total bandwidth needing to be processed for ngALMA is specified as 8 GHz per BBC. We thus require N_x X-engine modules, where N_x is given by:

$$N_x = \frac{8 \text{ GHz}}{B_x} = \frac{N_{ant} \times 8 \text{ GHz}}{F_c} = 1152\,, \tag{27}$$

where again we take F_c to be 500 MHz.

The correlator can be built by splitting these 1152 X-engine modules among multiple FPGAs. The number of modules instantiated on each chip is principally limited only by the total input bandwidth and number of DSP slices available on a given target platform. One of these two resources will dictate an upper-bound on the number of X-engines which a platform can support, implying a

total number of processing nodes required for a full (1152 X-Engine) system.

3.1.3 Potential Implementation platforms

We may examine currently available FPGA platforms to get a lower-bound on the power and cost of the ngALMA correlator. We expect that over time the cost of such a system will decrease in line with Moore's law. Two such platforms are:

SNAP2 Designed for astronomy applications by the Institute of Automation, Chinese Academy of Sciences, the SNAP2 board features a Xilinx Kintex Ultrascale XCKU115 chip with 5520 DSP slices and $\sim$ 400 Gb/s of Ethernet-based IO. A reasonable expectation is that 32 X-engines could fit on such a platform, using 4736 DSP slices and 256 Gb/s of input bandwidth. In it's current form, which was not optimised for correlator applications, the SNAP2 does not have adequate memory bandwidth to perform input buffering on off-chip resources. Input buffers would have to implemented on-chip, potentially limiting the size of input packets depending on exact implementation of the total system. It would not be possible to perform any kind of significant buffering for calibration purposes on SNAP2, though a similar board could be designed which factored this requirement in.

An X-engine for a complete dual-pol BBC-pair would require 36 SNAP2 boards. The Xilinx power estimation tool, assuming reasonable implementation parameters and a 20% overhead for peripheral components, suggests each FPGA would dissipate approximately 75 W of heat. SNAP2 is expected to retail at approximately $15k.

HTG-910 This commercially developed platform from Hitech Global[7] features a latest-generation Virtex Ultrascale+ XCVU13P FPGA, with 12288 DSP slices and 600 Gb/s of IO via multiple 100 GbE capable QSFP+ expansion ports. this board is capable of hosting approximately 64 X-engine modules, with an imput data-rate of 512 Gb/s. Unlike SNAP2, this latest generation board has 32 GB of data in off-chip memory. However, this only amounts to $\sim$ 0.5 s of input data.

An X-engine for a complete dual-pol BBC-pair would require 18 HTG-10 boards. The Xilinx power estimation tool, assuming reasonable implementation parameters and a 20% overhead for peripheral components, suggests each FPGA would dissipate approximately 150 W of heat. The HTG-910 is available for purchase at a cost of $15k without QSFP+ connectors or external memory. Cost including these parts is likely around $16k..

Custom Platforms One may also consider platforms which are custom-built for ALMA around a given FPGA. This allows optimization of the ratio of IO

[7] `http://www.hitechglobal.com/Boards/Virtex_UltraScale+_Vita57.4.htm`

bandwidth and arithmetic resources, as well as providing for custom peripherals, such as high speed memories. Potentially these are capable of hosting $>$ 128 GB of DDR4 memory, allowing several seconds of buffering of input streams. Custom platforms also provide the opportunity to improve compute-density, by developing boards with many FPGAs. However, the overhead of designing such a board is significant – likely around $500k, based on experience with the CASPER ROACH2 platform =0mu plus 3mu[**werthimer17**].

Future Platforms Trends in IO and compute capacity of FPGAs from 2009 forward (Table 3.1.3), extrapolated to 2022, suggest that FPGAs based on 5 or 10 nm technology may be available and would probably achieve twice the IO and computational performance of current products for comparable power consumption and perhaps cost. If so, we might imagine that a future system could be constructed with just 8 such FPGAs servicing each dual-pol BBC-pair.

Table 2: FPGA Devices

Device	**Release**	**Process (nm)**	**IO[1] [Gb/s]**	**Compute[2] [TOPS]**
Virtex 6 (SX475T)	2009	40	11.25	8
Virtex 7 (VX690T)	2011	28	100	14
Kintex Ultrascale (KU115)	2014	20	80	22
Virtex Ultrascale+ (VU13P)	2016	16	400	49
Projected (approx 2x 16 nm technology)	2022	5-10	800	98

(1) For high-speed multi-gigabit transcievers configured to 10/40/100Gb/s Ethernet (bi-directional).
(2) Based on 1 DSP slice per CMAC, operating at 500 MHz. 1 CMAC = 8 OPS.

3.2 Reliability and SEU

This overview has not addressed issues of hardware reliability and Single Event Upsets (SEU). However, we note the following:

- The X-engine design is readily applied to an actively-switched (eg. Ethernet) interconnect, which would allow spare processing units to be dynamically switched into the system should any individual processing node fail.
- Though FPGAs are susceptible to SEUs altering their configuration memory, Xilinx provides IP for SEU mitigation to detect and correct for these events.

3.2.1 Conclusions

Taking the HTG-910 as an example of a current latest-generation Xilinx FPGA platform, available off-the-shelf from commercial vendors, **an ngALMA X-engine could feasibly be built with currently available FPGA technology. The cost, power, and footprint for the X-engine would be**

∼$300k, ∼2.5 kW, and ∼ $\frac{1}{2}$ racks per dual-pol ngALMA BBC-pair. In future, the cost, power consumption, and physical size of this system may decline by at least a factor of 2. Given that the figures for a full 32 GHz bandwidth (4 dual-polarization BBC-pairs) may be as low as $600k and ∼6 kW, investment in custom platforms may not make economic sense.

3.3 GPU Resource Analysis

3.3.1 Comparison to FPGAs

The logic of resource utilization for GPU stream computing is fundamentally different than for FPGAs in a few key area. Firstly, competitive solutions may involve waste. Traditionally, a prime example has been application of FP32 arithmetic where, in a fixed precision system, 4-bit arithmetic would be used, chiefly because the GPU multiply and accumulate hardware units ("cores") have been limited to FP32. With that said, the latest Nvidia microarchitecture (Pascal) enables cores to *accept four 4-bit or 8-bit integers packed into 32 bits* and execute four parallel multiplies with one 32-bit accumulation as a single operation The net boost in performance over earlier generations of hardware is 4x for complex cross multiplication of noise-like signals, thereby reducing "waste." Data may be stored in memory in 4 or 8-bit representations. Packing into 32-bits can be accomplished either on a host node or via execution of a secondary kernel in addition to the one executing cross-multiplication. This involves a round-trip for data between device memory (see point 2 below) and cores that will reduce performance. However, if the calculation is not limited by transfer speed to/from device memory and computational resources, then degradation will be mitigated (see point 3 below).

Secondly, the metrics for optimization are Arithmetic Intensity (AI) and Computational Resource Utilization (CRU), the actual operations count per second as a fraction of a theoretical maximum operations count per second. AI refers to the number of operations executed per byte of information moved. The concept of data re-use is critical because GPU memory is hierarchical. CRU refers to how continuously cores are kept occupied as data is streamed into low-level memory. The denominator refers to an idealized limit set by clock rate and number of cores. The numerator is regulated by real-world bottlenecks, e.g., in data transport. At the highest level, e.g., several GB of DDR5 ("device") memory off-die, transfer rates are $O(10^2)$ GB/s. At the lowest level, register transfers occur at $O(10^2)$ TB/s. L2 cache speed, just below device memory, is typically $> 2\times$ that of device memory and shared memory speed is in between. Beyond this, the bus transfer speed from host to high-level GPU memory is more constraining, and beyond that, the external network transfer speed. In the end other factors pertaining to the algorithm and implementation may be more critical (e.g., successful coalescence of memory access within pages).

Thirdly, achieving high CRU depends in many cases on concealing latencies during parallel execution by overlapping operations. Two examples that pertain

to stream processing are (i) performing bus transfers into device memory while data from the previous transfer are processed, and (ii) parallel execution of secondary kernels on blocks of cores (a.k.a. steaming multiprocessors in Nvidia systems) that would are otherwise underutilized because a computation is bandwidth bound. The latter could refer a kernel that packs multiple low-precision numbers into 32-bits (an order-N operation) while the primary kernel executes outer products (an order N^2 operation).

Lastly, kernel execution is asynchronous, determined by optimizations applied by the C/CUDA compiler and the GPU resource scheduler at run-time. While it is up to the programmer to avoid race conditions and bottlenecks in parallelized algorithms, which is somewhat analogous to timing issues in FPGA firmware implementations, the dynamic allocation by the resource scheduler, e.g., of thousands of cores, is transparent to the user. (Though dynamic, for a regimented streaming application such as cross correlation, there is little variation in allocation.

3.3.2 Computation and I/O

For FP32 calculations on a single GPU, the maximum, theoretical, computational rate is

$$N_{\mathrm{FLOP}}^{max} = 2F_c N_{core}, \tag{28}$$

where F_c is the core clock rate, and N_{core} is the number of cores. For 8-bit arithmetic and current generation hardware,

$$N_{\mathrm{DLOP}}^{max} = 8F_c N_{core}. \tag{29}$$

The real-world execution of operations for a given time tick is

$$N_{OP}^{actual} = CRU \times N_{OP}^{max}. \tag{30}$$

The required execution count is

$$N_{OP}^{actual} = 8N_{ant}(2N_{ant} + 1), \tag{31}$$

which reflects calculation of cross and self-products in full polarization.

In the limits, algorithmic implementations are either compute bound or bandwidth bound, where data transfer rates throttle execution and cores are not used continuously. For correlation, this refers to the balance of O(N^2) operations, pairwise multiplication and manipulation of products for short accumulation intervals, and O(N_{ant}) manipulations of input data such as bus transfers, unpacking of payloads, and corner turning (Fig. 3; Clark et al. 2012). In the case of ALMA, N_{ant} is small enough that application of small low-power GPUs is most efficient, and network throughput per GPU is the predominant design parameter (B_{GPU}).

For a network-bandwidth bound calculation,

$$N_{OP}^{actual} = AI \times B_{network}, \tag{32}$$

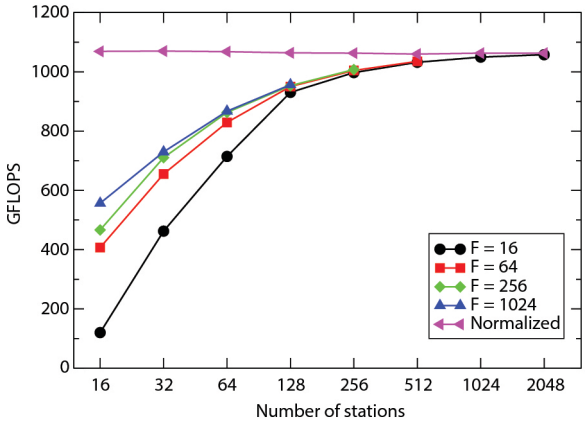


Figure 3: Demonstration of bandwidth boundedness. For fixed numbers of frequency channels (F), and different numbers of stations, GPU resource utilization grows as the O(N^2) portion of the problem grows, asymptoting when 100% of the cores are active. For small numbers of stations, a large GPU is less efficiently used because data transfer time dominates run time. The example is drawn from Clark et al. (2012) and a prior generation GPU.

where AI is the number of operations performed on a sample of F-engine output, independent of spectral bandwidth:

$$AI = 8N_{ant}(2N_{ant} + 1)/(2N_{ant}b_{in}), \tag{33}$$

from which CRU may be calculated for any combination of GPU and network throughput:

$$CRU = 4(2N_{ant} + 1)/b_{in} \times B_{network}/N_{OP}^{max}. \tag{34}$$

3.3.3 Input Buffering

Assuming NICs and GPUs are paired 1:1 to suit the bandwidth-boundedness of the problem, there will be one buffer for each stream into which packetized data will be unpacked[8] Selection of packet size depends on NIC protocols and offload tools intended to reduce CPU interrupt traffic. The LEDA correlator uses 8k packets. Use of packets as small as 1k saturated CPU cores and degraded throughput.

In contrast to the FPGA case, high-speed device memory for buffering is abundant, typically O(10) GB per GPU with current technology for O(10^3) cores, though the volume cannot be extended beyond the original complement.

[8]When implemented, the buffer is in fact three parallel buffers that step filled blocks through memory toward the GPU. The technique is used to hide from the GPU delays collecting a full set of the packets and to eliminate memory access contention (Clark et al. 2012).

The minimum buffer size is set by the minimum accumulation time after cross multiplication. The minimum accumulation time is that at which GPU memory traffic saturates some level in the hierarchical memory. Ideally, execution of the cross-multiplication takes at least as long as the time required to stage data in RAM and transfer it to the CPU cores.

Use of 8-bit arithmetic in hardware also designed to execute FP32 instructions requires a parallel set of buffers to enable re-packing of data prior to cross multiplication. The unpacked 4+4 bit complex numbers must be repacked (two 4+4 numbers per 32 bits) to enable the multiple and accumulate units to run at effectively quadruple rate. The repacking (a.k.a. bit swizzling) entails a round trip from device memory to the GPU prior to and separate from the cross multiplication and presumably managed by a second kernel. Thie secondary operation motivates the additional buffering.

3.3.4 Total Resources per BBC

As is the case for an FPGA platform, in building a full-bandwidth system, the architecture is parallelized over frequency, instantiating individual multi-channel X-engines on multiple GPUs. Each GPU is allocated to a specific channel range within a BBC bandwidth. For a bandwidth-bound system, the total number of GPUs required to process one BBC will depend on the network interface provided to each GPU,

$$N_{GPU} = B_{BBC} \times b_{in}/B_{network}, \tag{35}$$

ignoring the fraction of line rate that can actually be achieved and encoding losses.

We treat the case of two correlator modes differentiated by F-engine output: (i) high time resolution and low frequency resolution, and (ii) the reverse balance of these resolutions. The resource count per BBC will not change if the product of frequency and time resolution is constant, because GPU execution is parallelized over frequency and data are processed in time blocks to hide transfer times. The minimum integer number of frequency channels per GPU is one, corresponding to a frequency resolution, $\Delta\nu = B_{BBC}/N_{GPU} = B_{network}/b_{in}$. In either case, (i) or (ii), explicit bench testing is required to confirm that for an array of only 144 inputs, the time required for cross multiplication is sufficiently long to hide data transfers into and out of device memory.

3.4 Platform Architecture

Trends in network and GPU interface speeds largely determine the platform architecture, because the calculation will be bandwidth bound. GPUs are served by PCIe buses.[9] Packet capture and processing at high rates is a demanding application-specific research problem. The record is held by the system used

[9] PCIe ver.3 has a theoretical capacity of 16 GB/s in each direction for 16 lanes, which is standard for discrete GPUs.

for CHIME, 8×10 Gb/s on a dual CPU, dual GPU system (Vanderlinde et al. 2016).

We consider two architectures for ALMA, built around (i) discrete GPU devices suited for HPC, and new (ii) System-on-Chip (SoC) devices for which a light-weight host system with minimal if any CPU capacity is adequate. First generation SoC devices are available now as commercialized prototypes. Starting in 2018, Nvidia will ramp up supply of first generation units to serve large-scale deep learning and small-scale mobile applications (e.g., automotive). Because the potential markets for deep learning and mobile emphasize large numbers of units over high-density computing within each node, so we anticipate economies of scale that bring unit cost to approximately that of a mid-range consumer card today, $\sim$ \$500.

The minimalist limit for a "deep-learning system" (referring to the hardware) is an enclosure with many individual pairs of NICs and SoC GPUs, linked by a 16-lane PCIe v.4 bus, on the same PCIe controller. Though cost considerations drive this model, ultimately, a low-power CPU resources *may* be required for "housekeeping" pertaining to networking, with subsets of cores locked to processes associated with individual GPUs. Owing to a lack of information regarding the development path for Nvidia's competing NVLINK bus (on IBM Power processor-based systems), we do not consider it here. However, for reference, we note that at present NVLINK offers a 100 Gb/s bidirectional rate and intra-node unified memory environment for up to O(10) GPUs, and announcement on acceleration of inter-node networking is anticipated in the current quarter.

We adopt conservative scaling of GPU specifications into the future and root these in the current capabilities of low-power consumer models (the GTX750TI and 1050TI) and initial specifications already released by Nvidia for the Tegra Xavier product. We assume that PCIe v.4 will become standard in equipment in 2019, and that computational speed, expressed in INT8 operations per second (DLop/s), jumps by 2$\times$ with each new architecture (3 years), and by 20% when an architecture is refreshed midway between (3.5$\times$ over six years), with no increase in power utilization (Table 3).

3.4.1 Correlation resources

Mapping the X-engine specifications to GPU resources requires the total required compute and input bit rates to be divided equally among N_x of identical X-engine nodes. Because the computation is bandwidth limited, in the calculations that follow, the input rate is close to 100% of maximum, throttling the computing rate (Table 4). As the network line rate increases, the number of GPUs required to achieve the rate specified in equation (2) decreases. The content of Table 4 may also be presented graphically (see Figure 4). We note that of the several types of data transfer in the X-engine hardware (network$\rightarrow$host, host$\rightarrow$device memory (RAM), RAM$\rightarrow$shared memory), for well designed algorithms and implementations, the last is unlikely to be a limiting factor, in contrast to the first.

Table 3: Current and Forecast GPU models

GPU[(1)]	**Year**[(2)]	**PCIe bus (Gb/s)**	**Process (nm)**	**TDP**[(3)] **(W)**	**Theoretical**[(4)] **(DLTOP/s)**
Discrete – Tesla					
P4 (GP104)	2017	*128*	*16*	*50*	*17*
GV1xx	c.2018	128	16	50	33
GV1xx/refr	c.2019.5	256	16	50	40
GW1xx	c.2021	256	10	50	66
GW1xx/refr	c.2022.5	256	10	50	80
SOC – Tegra					
Drive PX2	2017	128	16	40	12
GV1xx (Xavier)	c.2018	192	16	20	20
GV1xx/re	c.2019.5	256	16	20	24
GW1xx	c.2021	256	10	20	40
GW1xx/re	c.2022.5	256	10	20	48

(1) Discrete GPUs to be HPC-grade, scaled with respect to the extant P4 and Xavier (italics). We focus on HPC units because they provide the least power consumption and greatest reliability. Continuous clock management is also enabled for HPC units.
(2) Forecast product cycle is 3 years between microarchitectures with a refresh midway between.
(3) Forecast availability of low-power discrete GPU models with a 50W TDP is based on experience with the last two microarchitectures. Forecast TDP for Tegra units assumes that Xavier performance will be at least matched.
(4) Maximum computing speed refers to synchronous 4×8-bit multiplications and 32-bit accumulation. Forecasting assumes $2\times$ gain with each generation of microarchitecture and $1.2\times$ gain with each refresh.

For a Tegra SoC GPU implementation c. 2018, assuming ingest at line-rate, an X-engine comprising 376 GPUs (excluding hot spares) and using 2x100 GbE network protocol can support cross-correlation of 8-BBCs with 27% computing headroom. The physical footprint would be 4 racks of 48 RU each.

3.4.2 Power Utilization

Power dissipations for systems based on GPUs shown in Table 3 and network protocols in Table 4 are shown in Table 5. These figures include estimated savings that result from operation with reduced clock rates and higher computational loads than indicated in Table 4 (80%). The power consumption for NICs are described in the table notes. Operating on the aforementioned assumption that the host for NICs and GPUs will require a minimal CPU capacity but a large number of cores, the power consumptions are included using the following budgets. Discrete-GPU system host total of180 W per 4 GPU-NIC pairs, derived from – 2x e5-2650Lv4: 130 W, 8x 8GB RAM: 11 W, C612 chipset: 7 W, PCIe v.4 equivalent to 4x PEX8747 PCIe switches and backplane: 32 W. SoC-GPU system host total of 118 W per 8 GPU-NIC pairs, derived from – e3-1268L v3: 45 W, 4x 8GB RAM PC3-12800L: 6.6 W, C226 chipset: 4.1 W, W, PCIe v.4 equivalent to 4x PLX8780 PCIe switches and backplane: 62.8 W.

For a Tegra SoC GPU implementation c. 2018, assuming ingest at line-rate, an X-engine comprising 376 GPU-NIC pairs and using 2x100 GbE network protocol will consume ∼17.9 kW in 48 nodes. Power consumption is inelastic with advances in GPU compute den-

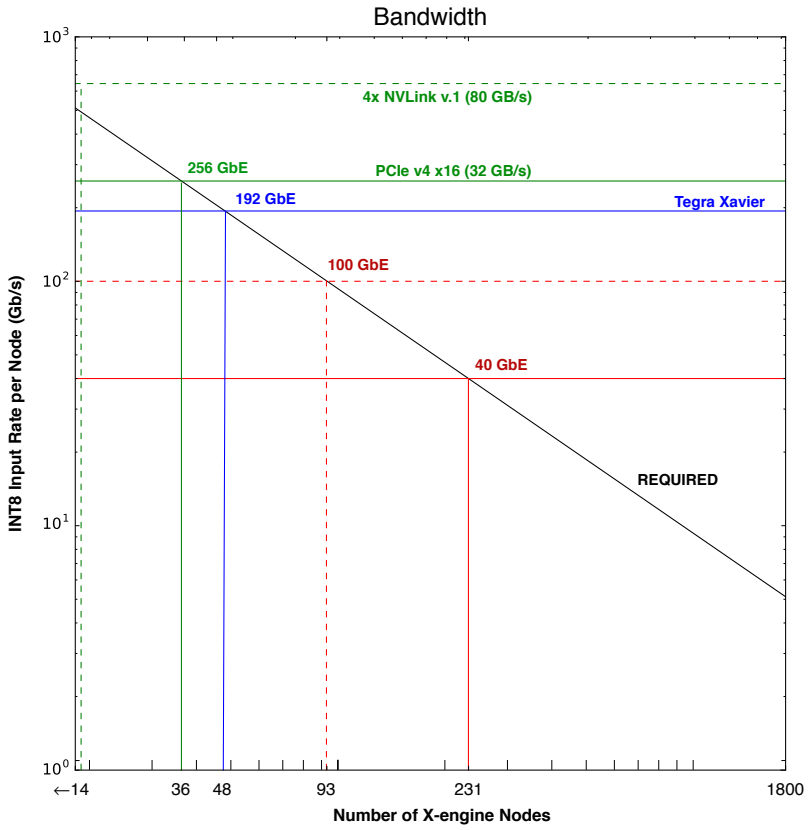


Figure 4: Required input rate per X-engine node as a function of the network protocol. In place of $2 \times 100\,\mathrm{Gb/s}$, we indicate explicitly the peak bandwidth anticipated for a Tegra Xavier SoC unit, which can in principle be supported by two 100 GbE links.

Table 4: GPU Bandwidth Bounding and Computational Resource Utilization

GPU	Ingest$\|_{CRU=1}$ (Gb/s)	CRU vs. Ingest (2x100 GbE)	(100 GbE)	(2x40 GbE)	(56 GbE)	(40 GbE)
Discrete						
P4 (GP104)	183.04	87.41%	43.71%	34.97%	24.48%	17.48%
GV1xx	366.08	43.71%	21.85%	17.48%	12.24%	8.74%
GV1xx/refr	439.30	36.42%	18.21%	14.57%	10.20%	7.28%
GW1xx	732.16	21.85%	10.93%	8.74%	6.12%	4.37%
GW1xx/refr	878.59	18.21%	9.11%	7.28%	5.10%	3.64%
SOC – Tegra						
Drive PX2	132.41	120.83%	60.42%	48.33%	33.83%	24.17%
GV1xx (Xavier)	220.69	72.50%	36.25%	29.00%	20.30%	14.50%
GV1xx/refr	264.83	60.42%	30.21%	24.17%	16.92%	12.08%
GW1xx	441.38	36.25%	18.13%	14.50%	10.15%	7.25%
GW1xx/refr	529.66	30.21%	15.10%	12.08%	8.46%	6.04%
No. GPUs		**47**	**93**	**116**	**165**	**231**

Note: PCIe v.4 x16 peak transfer rate (half-duplex) is 256 Gb/s. We adopt unofficial specifications for NVLINK v.1: 160 GB/s, assuming 8 lanes. Referring to equation 5, the aggregate X-engine ingest is 9216 Gb/s for one BBC, one polarization, one sideband, assuming 4+4 representation of complex numbers.

sity because operation is bandwidth bound. It scales downward approximately linearly with ingest rate per GPU beyond 200 Gb/s.

3.4.3 Cost

The assumptions underlying the cost equation are outlined in notes to Table 6. In addition, we adopt, based on quotes for hosts that use **current-day** analogs to the high-density, PCIe v.4, 2x 100 GbE systems required, the following costs. Discrete-GPU system host – Supermicro 4028GR-TRT supporting 4x GPU-NIC pairs: $2310 each. SoC-GPU system host – TrentonSys BPG8032 backplane and THD8141 single board computer: $704 each.

For a Tegra SoC GPU implementation c. 2018, assuming ingest at line-rate, an X-engine comprising 376 GPU-NIC pairs and using 2x100 GbE network protocol will cost ∼$874 K to cross-correlate four dual-polarization BBC pairs with 48 nodes. The cost is inelastic with advances in GPU compute density because operation is bandwidth bound. It scales downward approximately linearly with ingest rate per GPU beyond 200 Gb/s.

3.4.4 Beamformer resources

Beamforming in on each GPU will increase computation rates by $\sim$ 20% and entail introduction of an additional kernel. Calibration data can be moved into device memory at a low bit rate. Coherently added data represents a perturbation on I/O rates because the volume is reduced by $1/N_{ant}$ with respect to the input rate. A related system is operating at the Long Wavlength Array station in the Sevilleta national wildlife refuge.

Table 5: GPU-NIC Power Dissipation[1]

GPU	Power (kW)				
	(2x100 GbE)	(100 GbE)	(2x40 GbE)	(56 GbE)	(40 GbE)
Discrete					
P4 (GP104)	5.06	9.29	11.33	15.57	21.14
GV1xx	4.69	8.69	10.62	14.65	19.96
GV1xx/refr	4.61	8.54	10.45	14.43	19.68
GW1xx	4.39	8.18	10.02	13.87	18.96
GW1xx/refr	4.32	8.06	9.88	13.69	18.72
SOC – Tegra					
Drive PX2	3.34	5.95	7.20	9.77	13.11
GV1xx	2.23	4.11	5.03	6.95	9.48
GV1xx/re	2.19	4.05	4.96	6.85	9.36
GW1xx	2.09	3.88	4.76	6.60	9.03
GW1xx/re	2.06	3.83	4.70	6.52	8.92

[1] GPU power assumptions: core clocks reduced to achieve 80% utilization. Actual scaling requires study. Fourth root assumed based on experience with P4. NIC power assumptions: MCX516A-CDAT 2x100 GbE, 14.2W; MCX415A-CCAT, 1x100 GbE, 13.9W; MCX314A-BCBT, 2x40 GbE, 7.7W; MCX313A-BCBT, 1x40/56 GbE, 6.5W. Host power consumption is described in §3.4.2)

Table 6: GPU System Cost per BBC

GPU[1]	Cost (k$)				
	(2x100 GbE)	(100 GbE)	(2x40 GbE)	(56 GbE)	(40 GbE)
Discrete	$75k	$149k	$186k	$264k	$370k
"Discrete host"	$109k	$215k	$268k	$381k	$534k
SOC–Tegra	$19k	$37k	$46k	$66k	$92k
"SOC host"	$33k	$66k	$82k	$116k	$163k
NIC[2]	$57k	$76k	$62k	$71k	$99k
Total (Discrete)	$241k	$439k	$516k	$716k	$1003k
Total (SOC–Tegra)	$109.3k	$178.3k	$190.2k	$253.2k	$355.5k

[1] GPU cost assumptions: HPC-grade GPU $1600; SOC–Tegra GPU $400 (ignoring Drive PX2). Host: SOC @ 8-GPU $0.64K/stream$; $Discrete@4 - GPU4$K/GPU

[2] NIC cost assumptions: MCX415A-CCAT, 1x100 GbE, $812.83 (bulk quote 100 units); MCX516A-CDAT, 2x100 GbE on one NIC, $1,220.56 (scaled from previous); MCX313A-BCBT, 1x40/56 GbE, 430.00; $MCX314A - BCBT, 2x40GbE$,535.00.

3.5 Reliability and SEU

The study does not address issues of FPGA and GPU hardware reliability and Single Event Upsets (SEU). However:

- the X-engine design is readily applied to an actively-switched interconnect, which by its nature allows spare processing units to be dynamically switched into the system should any individual processing node fail;
- the odds of catastrophic failure for industry standard FPGA or GPU components are extremely small, and those being considered are low-power devices
- the impact of failure of a node on other nodes, or the network, based on experience in the High Performance Computing community, are smaller still;

- software-control over mains power for each node individually, using standard power distribution units, would reflect best practice;
- though FPGA nodes are susceptible to SEUs altering chip configuration memory, Xilinx provides IP for SEU mitigation, to detect and correct for these events; and
- though GPU nodes are susceptible to SEUs, these would lead to a node going offline or internal throughput between processing stages collapsing, either of which would be readily detected by even crude a monitor and control systems, triggering an automated hardware swap, rerouting of network traffic, and power down (of the afflicted node).

For GPU platforms, the risk of a "mundane event upset" (MEU) is greater than an SEU, but experience thus far shows that it is small for a system with thoughtfully engineered software. Because an X-engine GPU platform is asynchronous and is governed by a general purpose computing operating system, a often posited example of a MEU is the launch of a linux housekeeping process that leads to a bottleneck and dropped F-engine packets. To enable preliminary assessment of a GPU-driven platform, we review basic specifications and statistics for operation of the ~ 16 TOP/s LEDA correlator which has relatively tight engineering tolerances.

The LEDA X-engine comprises 11 servers that each capture 21.4 Gb/s (235 Gb/s aggregate) on a single 40 GbE Ethernet link. Each node contains 2 GPUs. The K20X clocks are set to 1.08 GHz, 47% above the manufacturer specification. Each Supermicro 1027GR-TQF server is equipped with a non-redundant PSU, two 115W Xeon CPUs, 128GB of Samsung DDR3-1600 registered ECC RAM, and a Mellanox single-channel 40/56GbE NIC.

Since system start in September 2013, no hardware has failed. The ensemble of **HPC-grade** GPUs operates at $90 \pm 5\%$ of the maximum power, and none is above 97%. Time variability in die temperature variation is the primary contributor to component failure (Nvidia p.c., 2012), so a 66 ± 2 F, over-pressured plenum feeds the front panel air intakes. The instantaneous-average die temperature is ~ 49 C, well below the ~ 85 C operating maximum. One interval of continuous operation began in December 2016. The system ran without interruption or fault (e.g., dropped packets) for 39 days. The run was interrupted by a mains failure.

4 Conclusion

For hardware available c. 2018, FPGA and GPU solutions are comparable in cost and power consumption for technologies available c. 2018. Assumptions underlying the conceptual designs presented here require testing. An FPGA-based X-engine would consume $\sim$10 kW, $\sim$\$1.20M, and $\sim$2 racks. An equivalent GPU-based system would consume $\sim$17 kW, $\sim$\$870K, and $\sim$4 racks.

Primary uncertainties in the above calculations of this appendix are (i) whether operation at the FPGA clock rate assumed here can be achieved; (ii) whether cost and availability of off-the-shelf FPGA processing boards will motivate design and production a custom FPGA board, (iii) whether it will be possible to ingest a packetized stream of 200 Gb/s per GPU using an SoC system as expected, and (iv) whether the desired configuration of high-density PCIe v.4 backplane hardware needed for GPU operation will be readily available. If market pressures are such that no stock PCIe v.4 backplane configuration is optimized for continuous, high-throughput between pairs of peripheral devices (i.e., GPUs and NICs), then a minor board redesign, relying on stock chipsets, should be practical and relatively straightforward.

2.6 Determine optimal X-engine architecture given DSP platform

Assigned to Hickish, Greenhill, Escoffier, ***Blackburn*** *& Primiani*

1. Bandwidth per unit X-engine
2. Number of baselines per DSP platform (influenced by A)
3. X-engine minimum integration time and dump time
4. Final accumulator implementation, requires lots of memory
5. Visibility read-out implementation (i.e. 1 GbE, 10 GbE, etc.)
6. Total amount of data, implications (may require separate study)
7. Interaction with the archive

Digital Correlator and Phased Array Architectures for Upgrading ALMA

WP2.6: X-engine architecture

L. Blackburn, L. Greenhill, J. Hickish, R. Primiani, A. Young

May 5, 2017

Contents

Table 1: Project Specifications Most Relevant to the X-engine Architecture

Item	**Requirement**	**Impact**
Antennas	72	Number of baselines
BBC bandwidth	8 GHz	BBC compute and data rate
BBC's per antenna	4	Total compute and data rate
Polarizations per BBC	2	Number of pol. products
Sample format	4+4 bit complex	Input data rate into X-engine
Channels per BBC	$\sim$8M, or 2^{23}	Data transpose requirement
Time integration	$\sim$1ms (ac), 16ms (xc)	Data output rate, rate comp.
Frequency resolution	$\sim$1 kHz, 4 MHz	Line resolution, delay comp.
Peak output rate	$<$100 GB/s	Accumulation requirements
Phased-array beams	$\sim$4	Output data rate

1 Introduction

The role of the X-engine in an FX correlator is to perform the pairwise cross products of complex spectral data for each baseline, and accumulate the results by vector averaging. Because each X-engine unit will necessary gather data from all antennas, it also provides a suitable place for beamforming — phase-aligning and stacking signals from multiple antennas to provide a single synthesized aperture. **WP2.5** identified both FPGA and GPU platforms as appropriate for cross correlation of a hypothetical ngALMA array. A GPU platform based on GPU system-on-chip (SoC) commodity hardware was projected to be more cost effective in $\sim$2020, and is the focus of this X-engine architecture study.

1.1 Baseline requirements

System specifications and requirements which are relevant to X-engine design (also see **Table 1** in **WP2.5**) are summarized here in table 1. The total input rate is driven by the number of antennas, total bandwidth, and sample bit-depth and must be supported by the total network and system bus input capacity distributed across a number of X-engine computing units. Computational cost for pairwise correlation also scales linearly with bandwidth, but quadratically with the number of antennas for an arbitrarily spaced array. For arrays with very many antennas, the quadratic scaling will drive X-engine design, but for 72 antennas the computational cost is easily met by modern GPU devices. Finally because correlation is trivially parallelizable, memory capacity and bandwidth constraints are typically met by careful ordering of the calculation. Additional features such as beamforming and sub-arrays do not significantly increase (and may reduce) resource requirements, but they increase the complexity of the X-engine architectural design.

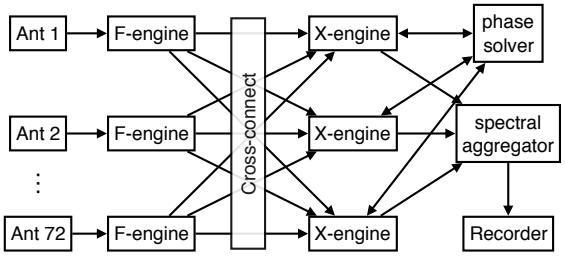


Figure 1: Overall FX correlator system architecture. Each X-engine units accepts as input a small portion of the full spectrum to be correlated from all antennas. The full-polarization cross products are calculated and accumulated before being sent back through the network for further processing. The X-engine also accepts real-time phase calibration solutions from a VLBI phase calibration engine, which are used to beamform the array data to targeted locations in the field-of-view.

1.2 Overall system architecture

The X-engine nodes process data from the F-engines, which first convert antenna voltage time-series into frequency domain representations, perform the baseline correlation, and then send output for downstream processing such as visibility archival, real-time phase calibration, and synthesis of beamformed data. Figure 1 shows a simplified system digram with the essential elements of the correlator. Each X-engine unit accepts a fraction of the total bandwidth from all antennas (and polarizations). The correlation of these spectral ranges are done in parallel across all the X-engine units. External elements of the special beamforming subsystem include the real-time phase calibrator, which aggregates visibilities across the entire bandwidth in order to solve for a unique set of antenna phases, and the spectral aggregator which accepts slices of beamformed data from all the X-engines in order to reconstruct and reformat a beamformed data stream to a given specification.

2 Resource Requirements

In this section we reproduce the basic calculations which determine I/O, computational, and memory requirements for the X-engine. We define the following variables,

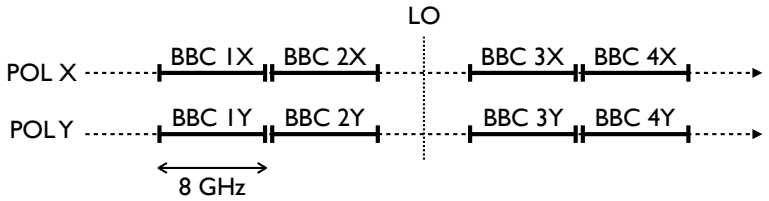


Figure 2: Frequency setup and location of 8 GHz Nyquist-sampled baseband channels (BBC) for one dual-polarization ALMA receiver. Adjacent BBC's may overlap slightly if necessary for continuous coverage of usable bandwidth, although the system is designed to process the full 8×8 GHz of bandwidth. With 4-bit sampling, the total data rate is 8×64 Gbps per antenna.

$N_{\rm ant}$	number of antennas (72)
$N_{\rm bl}$	number of baselines (2556)
$N_{\rm bf}$	number of phased-array beams (4)
BW	total bandwidth per antenna per polarization (32 GHz)
$b_{\rm in}$	bit-depth of each real-valued sample (4)
Δt	dump time resolution
$\Delta\nu$	dump frequency resolution
$b_{\rm out}$	bit-depth of each complex component of visibility output (32)

2.1 Input/Output rate

The input and output rates for the X-engine are,

$$R_{\rm in} = 2\, b_{\rm in} \times {\rm BW} \times 2\, N_{\rm ant} = 36864 \text{ Gbps, dual polarization}$$

$$R_{\rm out,ac} = 2\, b_{\rm out} \times {\rm BW}/(\Delta\nu\Delta t) \times 4\, N_{\rm ant} = 589824/(\Delta\nu\Delta t) \text{ Gbps, full Stokes}$$
$$R_{\rm out,xc} = 2\, b_{\rm out} \times {\rm BW}/(\Delta\nu\Delta t) \times 4\, N_{\rm bl} = 20938752/(\Delta\nu\Delta t) \text{ Gbps, full Stokes}$$
$$R_{\rm out,bf} = 2\, b_{\rm in} \times {\rm BW} \times 2\, N_{\rm bf} = 2048 \text{ Gbps, dual polarization}$$

We have separated output rate due to auto-correlation (ac) products and cross-correlation (xc) products in case different time-frequency accumulation is needed for each type. In **WP2.5** it is shown that the ouput data rate places significant constrains on the time-frequency resolution of the output data. For nominal wide-field mapping parameters $\Delta t = 0.015$s, $\Delta\nu = 4$ MHz, the output rate for visibility cross products is 349 Gbps, and a strategy other than full data archival would likely be necessary. Another scenario is narrow lines $\Delta\nu \sim 1$ kHz at high-frequency, where phase coherence due to the atmosphere limits $\Delta t \sim 1$s. This sets a data rate for cross products of 21 Tbps, so some targeted zoom-mode and possibly online phase calibration to allow for increased time accumulation is required.

2.2 Compute rate

The total rate of complex multiplications for correlation is,

$$R_{\mathrm{cm,corr}} = \mathrm{BW} \times 4\,(N_{\mathrm{bl}} + N_{\mathrm{ant}}) = 336384 \text{ Gcmps, full Stokes}$$

Each complex multiply represents 4 real-valued multiplications and 2 real-valued additions (collectively referred to as one CMAC). During accumulation, every cross product is added to a total, so that the number of complex additions during accumulation is approximately the same as the number of complex multiplications during correlation.

Beamforming requires each antenna data stream to be multiplied by a complex phase factor and stacked. This requires an additional,

$$R_{\mathrm{cm,bf}} = \mathrm{BW} \times 2\,N_{\mathrm{ant}} \times N_{\mathrm{bf}} = 18432 \text{ Gcmps, dual polarization}$$

and a similar number of accumulations.

2.3 Memory

Because cross correlation is trivially parallelizable across time and frequency, the fundamental memory caching requirements can be quite small — equal only to the memory required to stage antenna data and baseline visibility products for a single spectral sample times the number of threads that are to be executed in parallel due to platform considerations. However, streamlining the pipeline in this way requires the data to be in proper order for sequential accumulation. This creates a memory staging requirement for time-frequency transpose operations on the input data, where the transpose operation itself can occur at the end of the F-engine, the beginning of the X-engine, or be split across both. We assume the dump time will be larger than the F-engine's segmentation time for channelization. Then, the amount of memory required for the time-frequency transpose operation is,

$$M_{\mathrm{trans}} = R_{\mathrm{in}} \times \Delta t = 36864 \times (\Delta t/1\mathrm{s}) \text{ Gb}$$

The amount of memory required to support continuous operation is likely twice this to enable at least two buffers for alternating read/write.

For dump times of more than few seconds, this buffer size is quite large and instead it's likely that input data will not be ordered perfectly for sequential accumulation and dumps, and that accumulated visibilities will need to be staged and accumulated in a hierarchical fashion. Generally the cost of staging temporary visibilties is small because initial accumulation very rapidly reduces the data size. The limiting case will be at the highest frequency resolution, where the amount of memory to stage temporary visibilities for the full bandwidth during accumulation over time will be,

$$M_{\mathrm{accum}} = 2\,b_{\mathrm{out}} \times \mathrm{BW}/\Delta\nu \times 4\,(N_{\mathrm{bl}} + N_{\mathrm{ant}}) = 21528.576 \times (1\mathrm{kHz}/\Delta\nu) \text{ Gb}$$

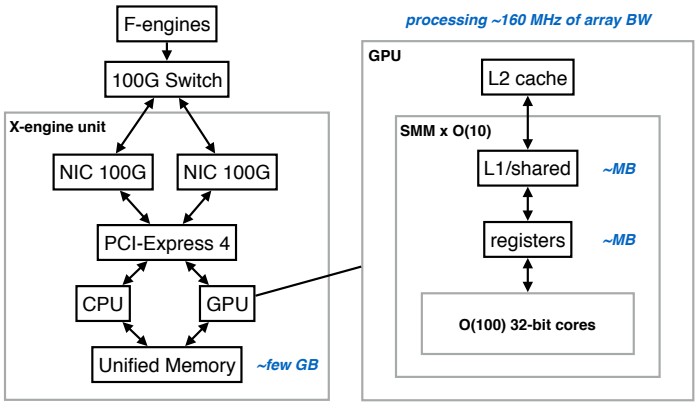


Figure 3: Diagram of GPU system-on-chip (SoC) platform, showing a single X-engine unit connected to two 100 Gbps network interface cards through a PCIe-v4 bridge.

If a high frequency resolution is to be used across the entire bandwidth, a large offline buffer to support staging of temporary visibilities during time accumulation is needed.

In most cases the memory bandwidth will be set by the input data rate, assuming no bit-inflation prior to data transfer,

$$R_{\mathrm{mem}} = R_{\mathrm{in}}$$

The staging of temporary visibilities can become comparable if accumulating at high channel resolution without sufficient transpose capacity. In this case,

$$R_{\mathrm{mem,accum}} = M_{\mathrm{accum}}/\Delta t = R_{\mathrm{out}}$$

where Δt is the dump time of the temporary visibilities. This is equal to R_{out} calculated for the temporary visibility dump time, rather than the final dump time. Since it can include bidirectional transfer, the loading of temporary visibilities may compete with the transfer of input data if done on the same device.

3 X-engine platform

3.1 Nvidia SoC configuration

The X-engine platform chosen in the cost and power-comparison study **WP2.5** is a cluster of Nvidia system-on-chip (SoC) devices connected to the correlator through two 100 GbE NIC over a PCI-Express version 4 backplane. Together the three devices will be referred to as "X-engine unit", and each unit operates independently on a small slice of the total array bandwidth (∼160 MHz) prior to

Table 2: Anticipated specifications for Xavier AI SoC

Item	**Xavier AI SoC**
Process	16nm FinFET+
SoC	1x Tegra "Xavier"
CPU cores	4x Custom ARM64
CPU cache	∼2 MB
GPU cores	512x Volta
GPU compute	20 DLTOPs
System memory	∼16 GB (∼160 GB/s)
Graphics memory	n/a (unified)
TDP	20 W

final downstream accumulation and processing. A block layout of the expected X-engine unit is shown in figure 3.

Each SoC device includes an ARM multi-core CPU, an Nvidia GPU, both connected to a moderately sized (many GB) block of Unified Memory. Anticipated specifications for Nvidia Xavier SoC (sampling Q4 2017) are listed in table 2. While future generation hardware is likely to be available by the time of correlator hardware acquisition, in this study we explore an X-engine that fits within the limitations of the anticipated Xavier platform.

3.2 SoC platform resource analysis

Here we compare the available resources of the Xavier AI platform (table 2) against the anticipated resource demands of the X-engine specification (section 2). The total input rate 36864 Gbps distributed across 200 X-engine units requires 184.32 Gbps input bandwidth per node, implying 92.2% utilization of input on both 100 Gbps network interfaces. A fully switched 16-channel PCI-Express v4 bus supports 31.5 GB/s bidirectional point-to-point transfer implying 73.1% utilization of the PCI-e v4 bandwidth to transfer input data to the Xavier SoC.

16 GB of unified memory corresponds to ∼0.7s of input data at 184.32 Gbps. For buffering of data while beamforming, this is only relevant for situations where the atmospheric coherence is quite short (∼3 seconds or less) and the source is bright enough that phase calibration of the array is achievable in a comparable integration time (section 5.2). A more generally useful input data buffer for beamforming will be several times as large. Since the SoC platform is unlikely to be easily customizable, we will not consider on-board memory for buffering of input data specifically for beamforming. The 16 GB does support the double-buffering of up to ∼0.35 s of data for time-frequency transpose operations, which can aid in pipelining efficient GPU accumulation but will not fully accommodate all integration times which will extend to at least 30 seconds (**WP2.5**). Thus the primary use of SoC unified memory will be for the staging of temporary visibilities to manage output data rate in the case of a large

number of channels, while short input data buffers will be used to minimize the rate and total number of these transfers.

At 8 bytes per complex accumulated visibility, ∼84 kB is required to capture a single full-stokes spectral visibility channel over the entire array. 8.4 GB of unified memory would stage 100k visibilties, corresponding to 1.6 kHz wide channels over the 160 MHz of total bandwidth allocated to each of the 200 X-engine units. The memory bandwidth requirement of these temporary visibilties will be 8.4 GB/Δt, where Δt is the transpose duration (and the maximum size for accumulation using only GPU shared memory). If we allocate 84 GB/s toward these transfers ($\Delta t = 0.1$s), an additional 4.6 GB (0.2s) is required for the transpose operation.

Under this configuration of 100k channels, the data transpose, loading, and the staging of visibilities for accumulation uses 13 GB of memory (81%) and 103 GB/s device memory bandwidth (64%). Thus even modest memory and memory bandwidth capacity projections for the Xavier AI SoC can support very fine channelizations of the ALMA bandwidth. Beamforming will add add an additional bandwidth requirement of 23 GB/s for loading of input data asynchronously from the correlation operation, for a total of 126 GB/s (79%). By loading input data asynchronously, the beamformer can take advantage of the data transpose buffer to mitigate the solution latency associated with accumulating first over the transpose duration.

Finally we compare the total rate of complex multiplications (354816 Gcmps) with the available GPU compute rate of 20 DLTOP per Xavier AI SoC, assuming 4 DLTOP (8-bit real) are required for a single complex multiply/accumulate. Assuming 200 units, the computational demand is ∼7.1 DLTOP per node, or a utilization of 35%.

4 Design

Here we outline specific design details for implementing the X-engine correlation architecture onto the proposed Nvidia GPU SoC platform. We discuss the design and applicability of the currently available open-source **xGPU** (Clark et al. 2012) CUDA correlator, and also describe the open-source **CASPER X-engine** (Parsons et al. 2008), a correlator for FPGA platforms.

4.1 Input from F-engine

The F-engine has two modes of operation (**WP2.3**),

- LO-Offset mode: A dual-stage power-of-two channelization is applied to each BBC/polarization data stream, sampled at 16 Gsps. The effective FFT window size is 2^{24} samples, corresponding to $\Delta t = 1.048576$ ms, and results in 2^{23} (∼8.3M) complex spectral points across each 8 GHz BBC bandwidth per polarization. This results in ∼336k spectra sent to each of 200 X-engine units every window duration from every antenna over the

complete 32 GHz dual-polarization bandwidth. If LO offsets are multiples of $1/\Delta t$, the data can be aligned post-channelization.

- 90/270 Walsh mode: Single sideband receivers will implement 90/270 Walsh phase switching for sideband separation. In this case, the FFT window size must align to the Walsh period of exactly 16 ms. Due to the high computational cost of non power-of-two channelization and benefit of short FFT window duration in terms of reduced blanking intervals during Walsh transitions, a window size of 10 μs is used giving 80k spectra across each BBC/polarization, or 3.2k spectra every 10 μs to each of 200 X-engine units from every antenna.

For efficient routing, spectral data from the F-engine will be packetized into jumbo packets of $\sim$9 kB, where each complex spectral point uses 1 byte. With no time-frequency transform applied to the data at the end of the F-engine, successive packets will contain spectrally sequential data for each FFT window duration. Since the F-engines operate in parallel, it is likely that common intervals of the spectrum arrive at the X-engine units from all antennas at the same time.

F-engine output is assumed to be fully delay-corrected, so that no further delay corrections are necessary to steer data to the phase center. This includes any slowly-varying instrumental delays which may be solved during calibration, and includes any residual sub-sample delay corrections that are applied to frequency-domain data prior to correlation/accumulation.

4.2 Auxiliary input

The packetized data model allows bookkeeping information to be transported along with the data, and we assume that packet headers supply information about data origin (for example Kocz et al. 2014): antenna, spectral range, polarization, and provide a time-tag. Along with requested accumulation parameters, this is sufficient to define and control X-engine correlation and accumulation behavior.

Some additional auxiliary input is required for the beamformer engine: locations of beamformer phase centers and pre-computed delay and phase offsets, as well as external real-time phase calibration information from the phase solver. These parameters will be further detailed in section 5. Real-time phase calibration input may also be useful for coherent accumulation over timescales similar to the atmospheric coherence timescale, in the case where the output data rate would otherwise be too large to do the accumulation downstream (e.g. in the case of very high spectral resolution).

4.3 Data staging

Data staging — accepting F-engine packets over the 2×100G network interface, placing the data into memory, and feeding it to the GPU cores to enable efficient correlation and accumulation — is of critical importance for enabling high

utilization of the GPU's computational capacity. Around 10^6 packets (∼8 GB) may be managed in memory at any time. It is assumed some form of direct transfer (bypassing CPU copy) will be available to offload data from the network to unified memory at the necessary throughput.

Time-frequency transpose occurs between writing and reading of data from memory, where data most rapidly varying in frequency, then antenna, then time, is arranged so that it is arranged to vary most rapidly in time-frequency blocks corresponding to the accumulation parameters. In the case of the highest frequency resolution (only time-accumulation), some degree of upstream time-frequency transpose at the F-engine may be preferable to facilitate coalesced (32–128 byte) access from SoC unified memory.

4.4 Cross multiplication and accumulation

A small amount of input data is read from unified memory and loaded onto GPU shared memory, and possibly expanded to 8+8 bit for low-bit integer cross multiplication (the alternative being on-the-fly inflation in register memory each time the data is used). Each complex multiplication product will ultimately be stored as 32+32 bit (8 bytes), which provides sufficient dynamic range for all subsequent accumulations. 1 MB of shared memory will be sufficient to stage 125k visibility products, or about 12 spectra over all baselines and polarizations. Thus if a few MB of shared memory is available in total, it should be possible to load in 32 sequential spectral channels at a time from all antennas while supporting arbitrary accumulation up to the available input buffer duration.

Accumulations in time that are longer than the buffer duration will require the writing and reading back of temporary visibilities while the bandwidth allocated to the X-engine unit (∼160 MHz) is processed serially. Two parallel accumulation pathways may be used if a rapid-cadence, large-bandwidth accumulation is desired (e.g. for online phase calibration), as well as a high spectral-resolution product acccumulated for ∼seconds or longer.

The beamform engine will also use the same input data (possibly re-loaded to shared memory after some delay), and will multiply the antenna data by predetermined phase factors before stacking and requantizing to 4+4 bits to form up to 4 separate beams. The additional demand on shared memory is small because it only scales with the number of antennas.

4.5 Visibility output

The X-engine will output visibilities at sufficient accumulation as to not overwhelm downstream processing, which includes array self-calibration and additional accumulation prior to archival. Typically the aggregate visibility output should be less than 100 GB/s, and likely closer to 1 GB/s. For 1 GB/s this implies an accumulation factor of ∼2.7 million. At the extreme, this requires the dynamic range of an additional $\log_2(2.7 \times 10^6) \sim 21$ bits beyond a single correlation product (9 bits per component), so that the 32+32 bit value should be sufficient. Further accumulation should be done hierarchically before

decimating or converting to floating-point. The accumulated visibilities are accompanied by header meta-data which specifies time-frequency and baseline, as well as any calibration that has been applied.

4.6 xGPU implementation

The open-source xGPU (Clark et al. 2012) correlator implements an X-engine in CUDA for Nvidia GPU hardware. It has been successfully deployed for real-time correlation as part of the Large Aperture Experiment to Detect the Dark Ages (LEDA) at the Long Wavelength Array station at Owens Valley Radio Observatory (LWA-OV) (Kocz et al. 2015). The LEDA correlator currently correlates 256 dual-polarization analog inputs of 98 MHz bandwidth, and thus has a 50-fold higher computational demand from pairwise correlation relative to input rate than the anticipated ngALMA array. However it otherwise follows a similar architecture to what is proposed here: channelization of digitized antenna data on FPGA hardware, distributed by switched ethernet to GPU correlation nodes which each process a fraction of the total bandwidth.

In xGPU, input data is assumed to be ordered such that it varies most rapidly in polarization, then station, then frequency, then time. GPU threads are parellelized over frequency and baseline blocks (selected to maximize data re-use), and iteratively accumulate correlation products over time. This organization effectively leverages the GPU's memory hierarchy and large number of cores. Clark et al. 2012 reach a sustained 79% of the peak single-precision floating-point throughput of the Nvidia GeForce GTX 480 (Fermi) with their approach.

Differences between the xGPU correlator described in Clark et al. 2012 and the X-engine GPU design for ngALMA are,

1. ngALMA X-engine will use 8-bit integer multiplication with 32-bit accumulation, instead of single-precision floating point. Morever data will be transfered from SoC unified memory as 4-bit, minimizing the GPU memory bandwidth required. Low-bit integer operations are supported on modern CUDA platforms. They are also used in the CHIME Pathfinder FX correlator (Denman et al. 2015; Klages et al. 2015), which uses a different open-source GPU kernel written for OpenCL.

2. The potentially large number of output channels per X-engine node ($\sim$100k) means that input data will not be ordered ideally for iterative accumulation. As described in section 3.2, this is because the full correlation matrix with the finest channelization will not fit all at the same time on the SoC (unlike Clark et al. 2012), and the memory requirements for first rearranging the data to vary more rapidly over time than frequency over a large accumulation interval are too great. Thus the bi-directional transfer of temporary visibilities for staging during accumulation will compete with other input and output for GPU memory bandwidth.

3. The tiling strategy of Clark et al. 2012 results in the redundant loading of

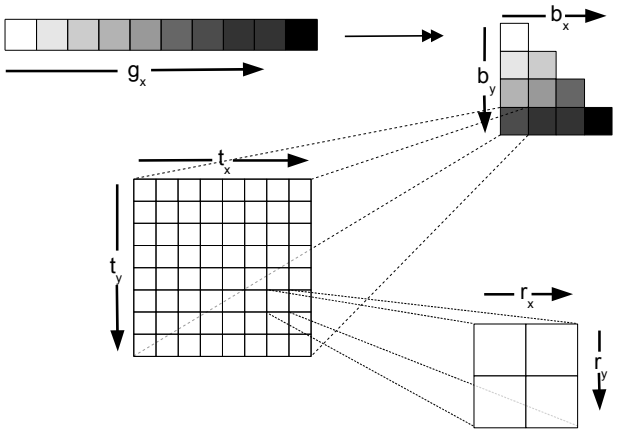


Figure 4: xGPU tiling strategy from Clark et al. 2012, showing the mapping between correlation sub-matrices and GPU threads. Correlation vector g_x is indexed by baseline blocks and is representable as the full triangular correlation matrix $b_x \times b_y$, with x and y indexing groups of antennas. The corresponding input data for each (b_x, b_y) is loaded into GPU shared memory and assigned a thread block $t_x \times t_y$ for which each thread (t_x, t_y) is responsible for calculating correlation $r_x \times r_y$. Larger initial tile size (b_x, b_y) minimizes redundant input data transfer from GPU general to shared memory, and is ultimately constrained by practical limits such as maximum registers per thread. Clark et al. 2012 found optimal performance on Nvidia GeForce GTX 480 (Fermi) using 8×8 sized thread blocks of 2×2 each.

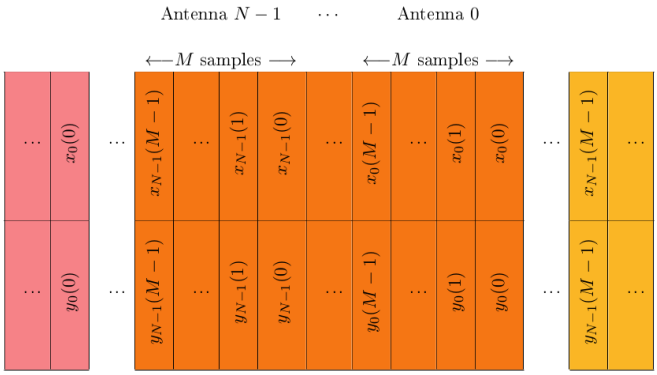


Figure 5: Data are input into the CASPER FPGA X-engine in a series of windows. An example of the organization of data in such a window is shown above, where $x_m(s)$, $y_m(s)$ represent the s^{th} time sample from the x and y polarizations of the m^{th} antenna, respectively. Each window comprises M time samples from a single frequency channel for a series of N dual-polarization antennas.

input data, that scales with the number of divisions of the initial correlation matrix (figure 4). With the optimal tiling parameters of Clark et al. 2012 (16×16 correlation submatrix per thread block), this would result in 5x the input bandwidth compared to a single loading of the antenna data, or from 23 GB/s to 115 GB/s. A different arrangement of initial thread blocks may be needed to avoid competition for memory bandwidth with temporary visibility transfers.

4.7 CASPER FPGA implementation

As discussed in **WP2.5**, FPGA platforms are a viable alternative to GPU platforms for the X-engine in a 72-element array. Here we describe the open-source CASPER X-engine[1] (Parsons et al. 2008), which is a parameterized module for cross correlation on FPGA platforms.

The CASPER X-engine processes data input windows which comprise of blocks of M time samples of a single frequency channel, serially from N dual-polarization antennas, $A_{0,\ldots,N-1}$. Consecutive windows of data usually cycle through a collection of different frequency channels, though this is not an absolute requirement of the design (Figure 5).

This input reordering requires a transpose of data for each antenna prior to the X-engine, in order to change from ordering time $\times$ frequency, to frequency $\times$ time. More specifically, a reordering of M spectra is required, requiring a memory buffer with a size scaling linearly with M and the number of channels

[1]Designed by Lynn Urry and implemented and maintained by members of the CASPER collaboration – see `https://casper.berkeley.edu/wiki/Win_x_engine`.

in a spectra.

The output of the X-engine module is a full-stokes correlation matrix for all $N(N+1)/2$ baselines for the input frequency channel, summed over all M time samples. In this way, with appropriate choice of $M > N$, the output date rate of the module can be kept approximately equal to the input rate. This has the benefit of reducing the I/O rate into the long-term vector accumulator which usually follows the X-engine.

In most systems, since consecutive X-engine input windows contain data from different frequency channels, this final stage vector accumulator has enough depth to store a correlation matrix for all the different frequency channels being processed by an X-engine. An alternative architecture is to process multiple consecutive windows from a single frequency channel at a time, in which case the vector accumulator need only store one frequency channel prior to outputting. The downside of this approach is that it implies larger reorders in the data transpose prior to the X-engine, with a corresponding increase in memory requirements.

The challenges for adapting the CASPER X-engine for the ngALMA correlator are similar to that for xGPU — due to the high degree of channelization and large input bandwidth processed by each X-engine unit, it is not possible to transpose the data under all configurations to support full iterative accumulation during correlation, and as a result the full correlation matrix needs to be temporarily saved for final-stage vector accumulation. The requirements of ∼GB of memory and ∼tens of GB of bandwidth, as discussed in section 3.2, are achievable on FPGA platforms.

5 Beamforming

5.1 Beamform requirements

The X-engine design supports the phase alignment and stacking of data from all antennas in order to form up to four synthesized beams at full data rate (128 Gbps). The X-engine resources used for this are modest, with the limit on number of beams primarily due to the practical limitations of recording data for offline analysis. The X-engine itself will be required to phase center data at up to 4 locations within the telescope primary beam, apply an externally-driven time-dependent phase calibration to each antenna, stack and requantize the data to 4+4 bits, and send the beamformed data back over the network. The beamform data will be at the same channelization as the antenna input data, and it is assumed that a downstream data-synthesis engine will combine and reformat the spectral data to whatever is needed prior to recording.

5.2 Input buffer

An input buffer that stages ∼few seconds of data in the X-engine prior to beamforming allows non-causal on-source phase calibration solutions to be used

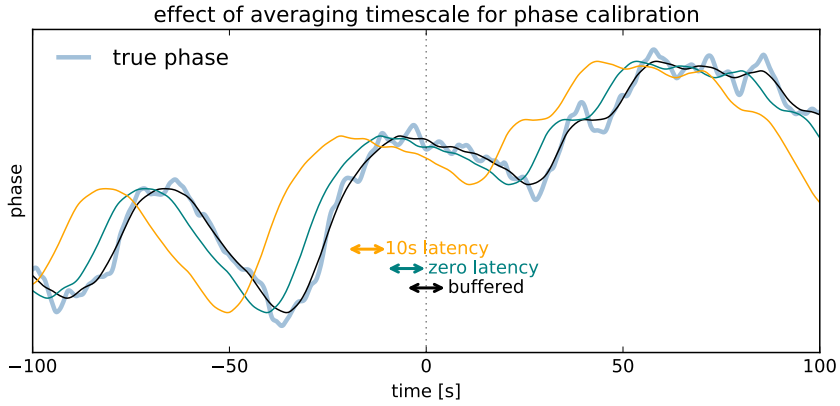


Figure 6: Effect of averaging timescale on estimate of antenna phase. Phase variation is modeled as due to 3D Kolmogorov turbulence with power-law index of 5/3. The magnitude scale of the phase variations (y-axis) sets the coherence timescale τ, which is independent of other parameters in this simulation. The integration length is 10 seconds, where the integration period for calculating a simple average phase is centered at -15s (10s latency), -5s (zero latency), and 0s (fully buffered). In this demonstration, no statistical thermal noise is shown. Differences between the average and true phase results from phase drift after a given lag from the center of the averaging window to the application time, as well as a difference between the average and center value due to random phase variation within the averaging period. In general the integration length is constrained by the available signal-to-noise, as well as the time after which the atmosphere becomes decoherent. The generic benefit of buffering is to allow for ∼twice as long integration periods for the same degree of atmospheric phase drift.

while phase-aligning and stacking data from multiple antennas. Since phase variations due to the atmosphere are a noisy stochastic random process, this improves the estimate. It also mitigates the effects of latency introduced through the real-time phase calibration loop.

For a simple model of the utility of buffering, we characterize phase wander by a coherence timescale, τ, representing the amount of time it takes for phase to drift over a single antenna by 1 radian on average. The expected phasing efficiency (large N) if integrating over an expected Gaussian phase drift distribution characterized by standard deviation σ is,

$$\mathcal{E} = e^{-\sigma^2/2}. \tag{1}$$

A lag corresponding to a full coherence timescale τ gives 61% efficiency. An efficiency factor of $\mathcal{E} = 95\%$ requires $\sigma = 0.32$. For Kolmogorov turbulence where

$$\sigma^2_{\rm atm} = (t/\tau)^\beta, \tag{2}$$

this limits lag $t < (-2\log(95\%))^{1/\beta} \approx \tau/4$ for $\beta = 5/3$. Filtering by moving average results in an additional systematic offset due to residual high-frequency power in the difference signal (true phase minus filtered phase). The residual power is related to the averaging time in units of τ,

$$\sigma^2_{\rm avg} = \frac{(\Delta t/\tau)^\beta}{2 + 3\beta + \beta^2} \quad 0 < \beta < 2 \tag{3}$$

Averaging over a full coherence timescale will result in a systematic offset between average phase and true phase which corresponds to a phasing efficiency of ∼95%, a much smaller effect than an actual lag of the same size but relevant when considering perfectly centered averaging windows with zero lag as achievable with buffered data.

The second effect of averaging timescale is on the amount of accumulated signal-to-noise available for calculating a phase solution. The visibility signal-to-noise on a single baseline is,

$$S/N = \frac{S_\nu}{\rm SEFD}\eta\sqrt{2\Delta t\Delta\nu} \tag{4}$$

For total integration time Δt and correlated bandwidth $\Delta\nu$ (including multiple polarizations), and amplitude efficiency factor $\eta \leq 1$. Decoherence in the baseline visibility over averaging timescale will result in an efficiency loss which we approximate using the residual power in the moving average error filter,

$$\eta \approx e^{-\sigma^2_{\rm avg}} \tag{5}$$

keeping in mind that the baseline visibility phases will have twice the variance due to atmosphere as widely-spaced (uncorrelated over Δt) antenna phase. There are $\sim N^2/2$ baseline measurements to solve N phases, so we can approximate the signal-to-noise ρ of each antenna phase solution as $\sqrt{N/2}$ times the S/N of each baseline. This adds an additional thermal noise to each antenna phase of $\sigma_{\rm thermal} = 1/\rho$ radians,

$$\sigma_{\rm thermal} = \frac{\rm SEFD}{S_\nu\eta\sqrt{N\Delta t\Delta\nu}} \tag{6}$$

For the parameters of the array: $N = 72$, $\Delta\nu = 2 \times 4 \times 8$ GHz,

$$\sigma_{\rm thermal} \approx 4.7 \times 10^{-7}\frac{\rm SEFD}{S_\nu\eta\sqrt{\Delta t}} \tag{7}$$

with source flux S_ν and single-dish SEFD (∼3500 Jy).

For our simplified model of phase error due to the three effects,

$$\sigma^2 = \sigma^2_{\rm lag} + \sigma^2_{\rm avg} + \sigma^2_{\rm thermal} \tag{8}$$

where $\sigma_{\rm lag}$ is due to systematic phase drift from the center of the integration window to application time, $\sigma_{\rm avg}$ is due to difference in mean phase from actual

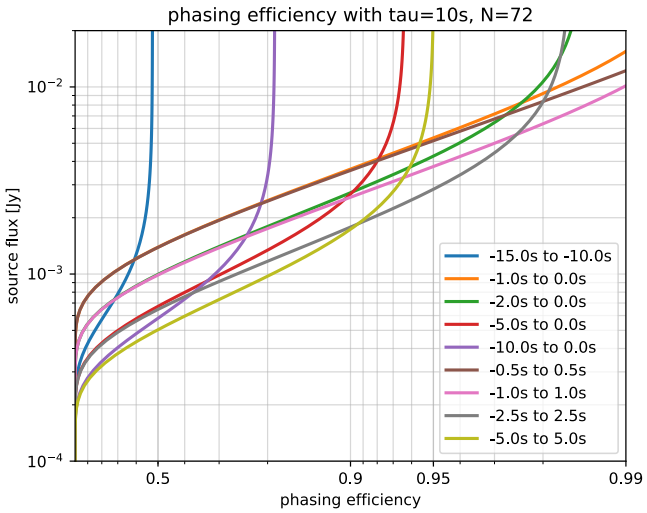


Figure 7: Phasing efficiency as a function of averaging strategy for an atmospheric coherence timescale of $\tau = 10s$. The phasing efficiency is calculated using a simple model which assumes independent phase noise in all antennas, and a least-squares solution based on a straightforward moving-average of baseline visibilities available at a given latency. Averaging intervals which extend into positive time imply a data buffer of sufficient size. For coherence timescales other than 10s, the curves are valid if integration segments are scaled proportionally, and source flux is adjusted to maintain the same total integrated signal-to-noise.

phase at the center of the window, and σ_{thermal} is from statistical error in the phase solution. Figure 7 shows on-source phasing efficiencies calculated at nominal system parameters over a range of source flux. When more than $\sim\tau/5$ seconds of integration time are needed to accumulate enough signal-to-noise for an adequate independent phasing solution, a similarly sized buffer greatly improves phasing success. Otherwise where the source is sufficiently strong, it is sufficient to have negligible latency for the phasing solution. Other strategies which exploit the spatial and temporal correlation of phase across antennas (e.g. using a moving frozen-screen atmospheric model) improve phase estimation under equal conditions without impacting X-engine design.

6 Conclusion

We have outlined a possible X-engine architectural design for a 72-element ngALMA array supporting 32 GHz of dual-polarization bandwidth. The basic strategy follows closely that of other GPU-based correlatation engines that are implemented in modern interferometers designed to scale to a large number

of elements (Kocz et al. 2015; Denman et al. 2015). It is anticipated that the X-engine platform selected in **WP2.5**, based on a cluster of Nvidia GPU/SoC units accepting data over switched 100G ethernet, will be able to satisfy the scientific requirements outlined for ngALMA correlation. Specifically we fit parameters of the design to the projected specifications of the next generation Nvidia Xavier AI SoC (sampling Q4 2017), assuming correlation is distributed across 200 units.

The ngALMA array, given the moderate number of antennas, large total bandwidth with wide baseband channels, and fine channelization requirements presents unique challenges for the design,

1. With only 72 antennas, ngALMA has a high ratio of inptut data rate versus computation required for pairwise correlation relative to other GPU correlators. The 200 Gbps aggregate input per X-engine unit is required for reasonable utilization of the GPU computing throughput, but has not been demonstrated. It will likely require a high-throughput low-overhead (zero-copy) direct network stack.

2. At the finest frequency channelization, the X-engine cannot support full time-frequency transpose of the data or full staging of temporary visiblities at the native F-engine dump time, and will require carefully tuned transpose and accumulation buffers to operate within memory and memory bandwidth limits.

References

Clark, M. A., P. C. La Plante, and L. J. Greenhill (2012). "Accelerating Radio Astronomy Cross-Correlation with Graphics Processing Units". In: *The International Journal of High Performance Computing Applications* 27.2, pp. 178–192. DOI: 10.1177/1094342012444794. arXiv: 1107.4264 [astro-ph.IM].

Denman, N. et al. (2015). "A GPU-based Correlator X-engine Implemented on the CHIME Pathfinder". In: *ArXiv e-prints*. arXiv: 1503.06202 [astro-ph.IM].

Klages, P. et al. (2015). "GPU Kernels for High-Speed 4-Bit Astrophysical Data Processing". In: *ArXiv e-prints*. arXiv: 1503.06203 [astro-ph.IM].

Kocz, J. et al. (2014). "a Scalable Hybrid Fpga/gpu FX Correlator". In: *Journal of Astronomical Instrumentation* 3, 1450002-330, pp. 1450002–330. DOI: 10.1142/S2251171714500020. arXiv: 1401.8288 [astro-ph.IM].

Kocz, J. et al. (2015). "Digital Signal Processing Using Stream High Performance Computing: A 512-Input Broadband Correlator for Radio Astronomy". In: *Journal of Astronomical Instrumentation* 4, 1550003, p. 1550003. DOI: 10.1142/S2251171715500038. arXiv: 1411.3751 [astro-ph.IM].

Parsons, Aaron et al. (2008). "A Scalable Correlator Architecture Based on Modular FPGA Hardware, Reuseable Gateware, and Data Packetization". In: *Publications of the Astronomical Society of the Pacific* 120.873, p. 1207. URL: http://stacks.iop.org/1538-3873/120/i=873/a=1207.

2.7 Determine design of VLBI capability

Assigned to A. Young, ***Doeleman****, Crew, & Lacasse*

1. Beamformer functionality, number of beams
2. Phasing loop software considerations
3. High-bandwidth data recording systems
4. Transient buffer capabilities
5. VLBI post-processing, sample rate conversion

3.8 WP 2.7 Summary: VLBI Capability

3.8.1 Background

Beamforming the ALMA dishes creates a high sensitivity VLBI capability for ALMA that can be used to anchor mm and submm VLBI arrays for ultra-high angular resolution and sensitivity science applications. The ALMA Phasing System (APS), designed to work with the current ALMA Correlator, is capable of forming a single beam over the full 16GHz of bandwidth. In combination with Mark 6 VLBI recorders, the APS enabled the Global mmVLBI Array (GMVA) and the Event Horizon Telescope (EHT) to offer a 3mm (Band 3) and 1.3mm (Band 6) VLBI capability to the ALMA community in proposal Cycles 4 and 5. In addition, the phasing capability can be used to study high frequency pulsars and magnetars using the same VLBI data capture systems. A full science case for ALMA beamforming is detailed in Fish et al (2013). The Next Generation ALMA Correlator will have native beamforming capability that far exceeds that of the APS, enabling VLBI at high frequencies and under a variety of atmospheric conditions.

3.8.2 VLBI/Beamforming Requirements

Beamforming for VLBI and pulsar applications imposes several specific requirements, some of which are necessarily dependent on the atmospheric conditions, array configuration and observing Band. Listed first are general requirements of the phasing system:

- Phasing of the array is done as near to real-time as possible. This is so that the phasing efficiency, defined as:

$$\eta_{\text{eff}} = \frac{|\vec{S} \star A_{\text{ref}}|}{\sum_i |A_i \star A_{\text{ref}}|} \qquad \text{where} \qquad \vec{S} = \sum_i A_i \qquad \text{and} \qquad A_{\text{ref}} = \text{non-summed reference antenna.}$$

  is minimally dependent on latency in the phasing system and primarily dependent on the thermal receiver noise in the antennas. One measure of real-time phasing is that the coherent sum of all antennas is computed over a time short enough to ensure the phase drift on the longest baseline is < 1 radian.

- A real-time measure of phasing efficiency should be computed. This can be done through routing the phased sum of antenna signals either back through the correlator, or through a second single-baseline correlator that compares the sum to a reference antenna.
- Polarization leakage in the phase sum should be no greater than the average leakage for a single antenna.
- Phasing should be supported for all ALMA Bands.
- Phasing should have the capability to remove known source structure.
- Phasing algorithms should have the capability of solving for an evolving atmospheric model.
- Several modes of phasing should be implemented: phasing on in-beam target, phasing on in-beam calibrator, phasing on out-of-beam calibrator.
- Data output of the phasing system should be available in standard VLBI format (2, or 4-bit data with suitable headers – e.g., VDIF).
- For the pulsar case, the requirement is to be able to detect millisecond pulsars with a Dispersion Measure of 3000 cm^{-3} pc. This sets an upper limit on channelization of 32MHz for ALMA Band 1. For pulsars, it is also desirable to maintain the maximum number of bits possible, but 2-bits are sufficient if any auto-leveling system has a time constant greater than ~5 seconds.
- Normal ALMA data should not be affected by the phasing system.

A requirement that captures the variations in conditions can be stated as:

- Phasing efficiency > 95% under 'nominal' conditions: A compact 1 Jy source, in Band 6, 35 antennas, baselines < 2km, less than 1mm PWV, and rms path fluctuations < 0.125mm); or
- RMS phase error of < 1 radian on baselines < x km in weather acceptable for a given band.

3.8.3 Flux Density Limits

To identify lower flux density limits for phasing calibrators, we assume a System Equivalent Flux Density (SEFD) for an ALMA antenna of 3500 Jy in Band 6 at nominal elevation, (aperture efficiency of 0.7, System Temperature of 100 K). If we further assume 2-bit data, the ability to use the full 64GHz of bandwidth to determine phasing solutions, and summing 35 dishes, we can plot the signal-to-noise of the phasing solution and the expected coherence loss as a function of integration time and source flux density (Figure below). Because this figure assumes zero coherence losses due to atmospheric effects, including due to latency of the phasing solution, these flux density limits are understood to be lower limits. It should be noted that if atmospheric turbulence introduced noise into the phasing solution because of latency issues (ie the phase solver requires integration times of order – or longer than – the coherence time of the atmosphere), then one could solve for the correct phasing solution a posteriori and adjust the computed signal accordingly. This will not recover the signal-to-noise ratio, but it will recover the expected signal value of the phased sum.

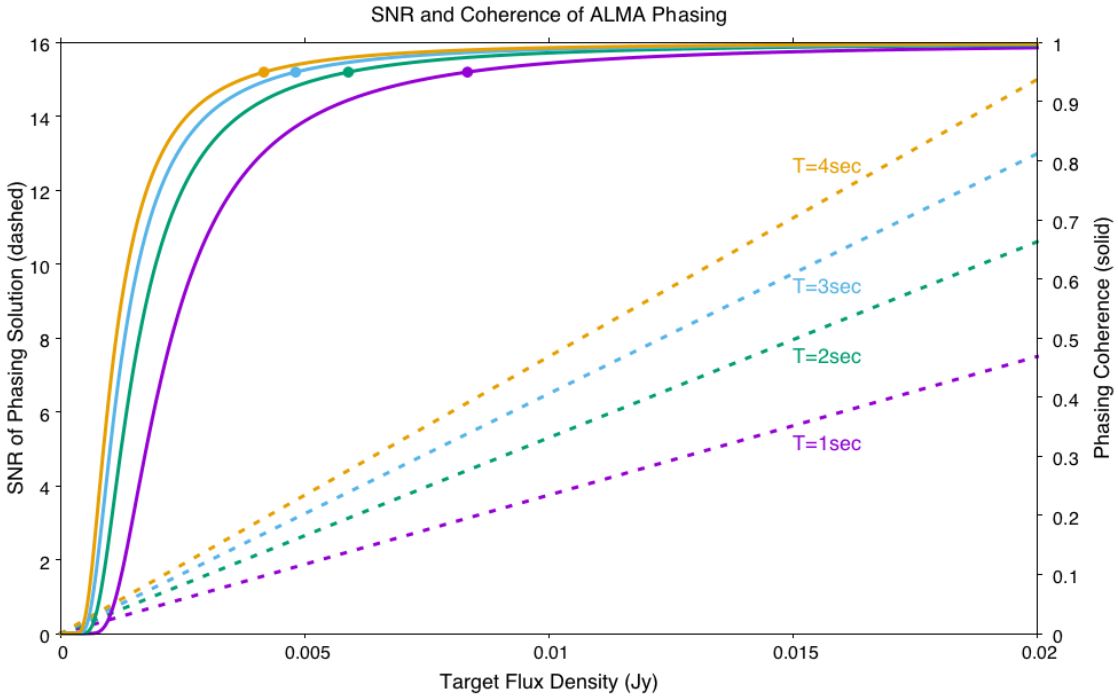


Figure 1: Signal to noise of the phasing solution for 35 ALMA antennas over 64GHz BW (dashed lines) as a function of phasing calibrator brightness for integration times of 1 to 4 seconds. Also shown is the phasing coherence as a function of target flux density. Solid circles mark points where coherence loss is 5%. This figure assumes zero coherence losses due to atmospheric phasing effects, including latency. The required flux density of the phasing target is <10mJy.

3.8.4 Data Buffer for Zero Latency

The current APS requires between 8-10 seconds to calculate and apply the phasing solutions needed to form the coherent sum of antenna signals. Thus the coherent sum at any given time is being formed using phasing solutions that are 8-10 seconds out of date, and atmospheric turbulence, or changing phasing

conditions of any kind, will cause coherence losses. At present, this latency is due primarily to communication delays of data and solutions imposed by the correlator architecture, the data archiving system, and the path that phasing solutions take to the registers where they are applied. As an example, the Figure below shows the effects of latency on phasing coherence: the longer the latency, the more coherence loss is incurred.

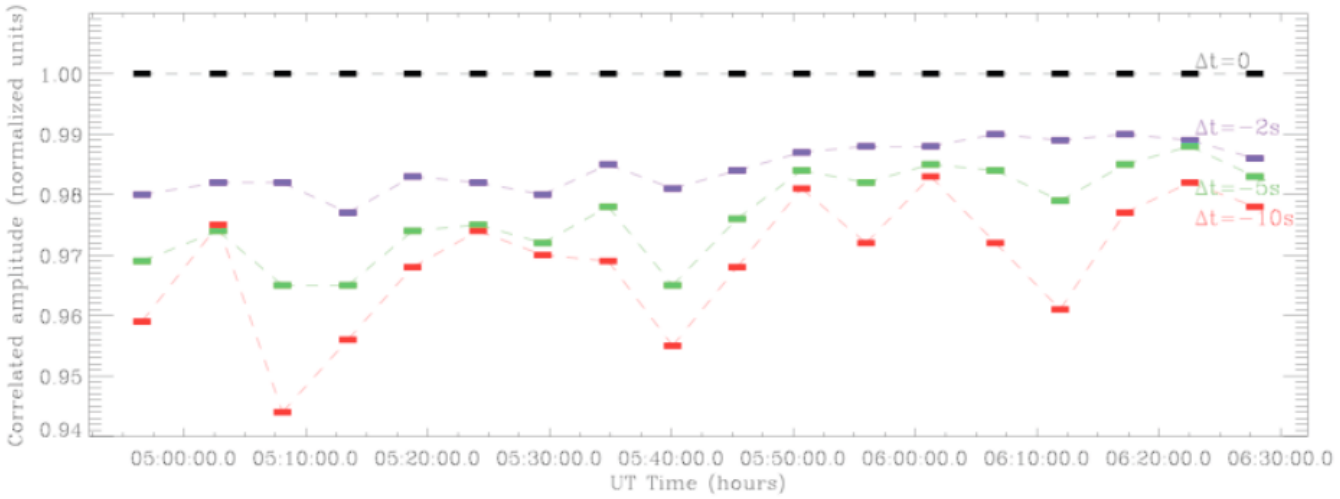


Figure 2: Coherence loss due to latency between the phasing solutions applied to the array and the data used to calculate those phasing corrections. For a 10 second latency, phasing efficiency in relatively good atmospheric conditions (PWV~0.9mm, RMS path length variations of ~125μm on 300m baselines: mean conditions at ALMA in May), can drop by 6%. This example uses Band 6 data from ALMA with phasing done a posteriori in CASA. Figure made by Lynn Matthews.

The Next Generation ALMA Correlator will need to address the latency issue in two ways. First, it will streamline communication and transfer of data needed for computing phasing solutions as well as routing of solutions to the X-engines where phasing is implemented. It is expected that this can straightforwardly reduce latency by up to an order of magnitude from the current APS. Second, a data buffer can be implemented that stores antenna signals for a flexible length of time. The size of the required buffer will depend on atmospheric coherence time, flux density of phasing calibrator, and baseline lengths in the array. At its simplest, the data buffer would have the following basic architecture:

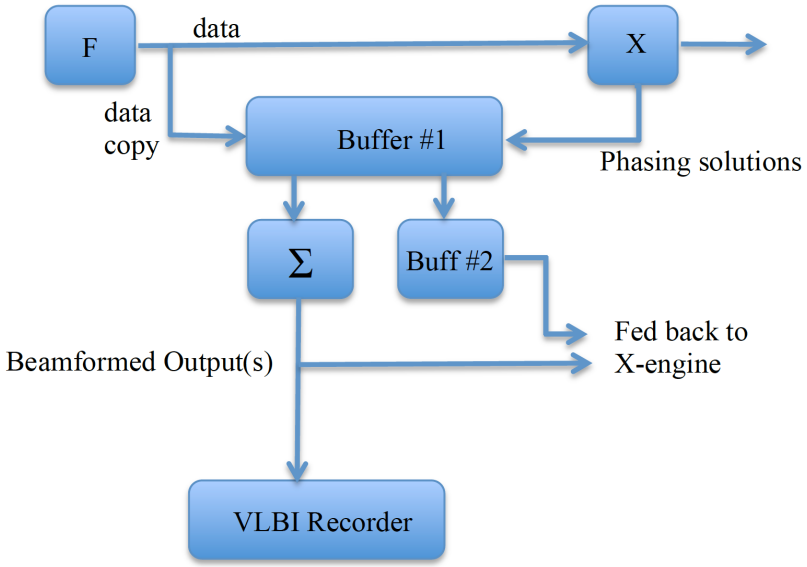


Figure 3: Possible architecture for a buffer that would reduce phasing solution latency to zero. The buffer could be implemented in the X-engine ("X") fabric. The summed signal and signals from several antennas not included in the sum could be routed back through the X-engine for calculation of phasing efficiency.

The size of the buffer can be expressed as a function of bit-depth, antenna number, bandwidth, and required integration time to achieve the necessary signal-to-noise:

Buffer Size = N_{bits} * N_{ant} * T * BW * 2
= 4 * 72 * 1sec * 64GHz * 2
= 35.7 Tbits per second of latency.
= 4.5 TBytes per second of latency.

Given the flux density limits shown in Figure 1, the buffer could reasonably be reduced to cover an integration time of ~100msec, or a total storage amount of ~0.5 TBytes.

3.8.5 Data Format, Data Transfer and VLBI Recorders

The Mk6 VLBI recorders currently installed at the ALMA OSF (4 units) can capture 16Gb/s, but could be upgraded to 32Gb/s, possibly higher. If only 32Gb/s, then adding four new recorders would handle a 4x increase in bandwidth at ALMA, but only for a single beam. The ngALMA correlator is specified to produce ~2-4 beams, which requires an aggregate data rate of 512 – 1024 Gb/s.

The VDIF format may still be useful in 2022 when the ngALMA correlator is constructed. It is also possible that network appliances that are essentially just packet recorders – perhaps with Solid State Storage – could replace VLBI recorders.

Multiple beams require high BW for the phased sum to be sent from the AOS to the OSF. The APP currently uses a single fiber with 64Gb/s sent using DWDM (8 channels, each transferring 8 Gb/s). To transfer 4 beams at an aggregate data rate of 1024 Gb/s, one could use 8 x 32 Gb/s channels for each beam, or a total of 32 channels, each 32Gb/s. Wider band DWDM channels up to 100Gb/s are now available, and it is a matter of cost and fiber availability that will determine the optimal data transfer scheme should the Next Generation Correlator be sited at the AOS.

2.8 Staging of new correlator ("ICD with Site")

*Assigned to **Lacasse**, Doeleman, Saez, Herrera & Baudry*

1. Study of location of correlator at OSF
2. Logistics of assembly on-site
3. Running in parallel with present correlator
4. (Science commissioning plan)—viewed as likely out-of-scope for this study.

Correlator Location Trade-offs

Alain Baudry, Shep Doeleman, Rich Lacasse, Alejandro Saez

Abstract

Advantages, disadvantages, costs and other impacts related to the location chosen for a new correlator are presented. The initial drafts of this document are based on the experience and expertise of personnel in this study and of colleagues who have provided input. The goal of this note is to provide a starting point for discussion with cognizant ALMA personnel and to ultimately produce a final product which lays out the trade-offs of various possible locations both qualitatively and quantitatively.

Introduction

There are four clear possibilities for locating a new correlator: either at the AOS or at the OSF and either in a new space or an existing space. There are advantages and disadvantages for each possibility. This document attempts to present these both qualitatively and quantitatively.

Two types of costs are considered in this analysis. The first is “real” costs where real money must be appropriated in a budget. An example of this is the cost of purchasing a new HVAC system or building a new space. The second is “productivity” costs. These are not real dollars; they are a dollar measure of the impact of a given option. The primary example of this is the amount of science time lost; lost engineering productivity can also be counted this way.

This report includes a section on assumptions used to evaluate the options. It is then followed by a table which provides an executive summary of the trade-offs. Next, additional information and discussion is provided in sections dealing with each option. A list of questions and issues to be discussed with ALMA personnel is included. A spreadsheet, CorrelatorCostTradeoffCalculations.xlsx is a companion to this document providing the calculations used to arrive at the cost estimates shown below.

Assumptions and Implications

The following assumptions are made:

1. It is necessary to operate both the new and old correlators during a transition period. The reasons for this are twofold. First it minimizes downtime during the transition period, allowing ALMA to keep doing science during the installation and commissioning periods. Second, it provides a ready means of comparison between the old system and new system; this simplifies the commissioning.
2. It is necessary to split the signals from all antennas to provide signals to both the new and old correlators. Swapping the cables back and forth is impractical and results in significant lost time. Note that this has been done successfully before. We further assume that the signal will not need to be amplified before the split (a splitter introduces a loss equal or greater than 3db this can be a problem if an antenna is located far away; the DRX needs at least -18dbm for error-free transmission.)

3. The new correlator can handle both the existing antenna data format and anticipated ones as well. This is required to allow operation of both the old and new correlators during the same time period It also provides some flexibility in the timetable for installing a new sampler and Data Transmission system.
4. A correlator satisfying the requirements produced by this group will dissipate approximately 225 KW (not counting HVAC dissipation) and occupy approximately 8 racks. These numbers are provided Brent Carlson, based on scaling the SKA design. Baudry has produced a separate estimate of 200 KW based on scaling the current correlator. The current correlator dissipates approximately 150 KW.

Tabular Summary of Trade-offs

Table 1. Summary of advantages, disadvantages and costs of locating a new correlator in four possible locations. [1]

Location:	Case 1: AOS, Existing space	Case 2: AOS, New space	Case 3: OSF, Existing space	Case 4: OSF, New space
Main Advantages	Cheapest option	Cheapest option that does not violate above assumptions	Better science and engineering productivity	Better science and engineering productivity
Main disadvantages	Lost time during transition Worst science productivity Limited science during transition	Science productivity slightly better than Case 1.	Signal transport costs	Signal transport costs
Capital costs[2]	Rack mounting to seismic mount modification: 10K New HVAC[7]: 164K Installation time: 50K Travel: 10K Equip storage: 10K Total: 380K	Rack mounting to new seismic mount: 100K New HVAC[7]: 164K Space for HVAC: 480K Room Construction[6]: 1.3M Installation time: 50K Travel: 10K Total: 2.1M	Rack mounting to new seismic mount: 100K New HVAC[7]: 158K Space for HVAC: 320K Room Renovation[6]: 50K Installation time: 50K Travel: 10K Signal transport: 2.4M Total: 3.1M	Rack mounting to new seismic mount: 100K New HVAC[7]: 158K Space for HVAC: 578K Room Construction[6]: 130K Installation time: 50K Travel: 10K Signal transport: 2.4M Total: 3.4M
Operational costs[3]	HVAC: 1190K Lost technical productivity: 400K Vehicle costs: 62K High altitude bonus: 37K Risk to personnel: **Total**: 1689K	HVAC: 1190K Lost technical productivity: 400K Vehicle costs: 62K High-altitude bonus: 37K Risk to personnel: **Total**: 1689K	HVAC: 2247K **Total**: 2247K	HVAC: 2247K **Total**: 2247K
Incremental Science costs[4]	$69M	$59M Time lost to longer MTTR.	$0.0M Time lost to shorter MTTR	$0.0M Time lost to shorter MTTR

	Half capacity during installation Time lost to longer MTTR Time lost to greater number of SEUs[5] is ignored.	Time lost to greater number of SEUs is ignored.	Fewer failures due to SEUs ignored	Fewer failures due to SEUs ignored

Notes:

1. Some numbers in this table are educated guesses and need to be refined. See the spreadsheet CorrleatorLocationCosts.xlsx for details.
2. For infrastructure, not including the cost of the correlator itself.
3. Cost over 20 years.
4. Lost time, at $11.4K per hour, over 20 years. Putting the correlator in the existing correlator room results in added time lost during the non-productive transition time. Assume that time lost due to commissioning a new correlator is more than compensated for by the added productivity provided by the new correlator.
5. MTTR = Mean time to repair. SEU = Single Event Upset (logic errors due to cosmic rays)
6. For the AOS, construction costs are based on actual construction costs inflated by 1.5%/year to 2016 dollars. OSF costs are based on actual construction costs and an estimate of the square footage. Possibly the space for the HVAC can be obtained for less. AOS construction will have to address the possible impact of mechanical vibrations on the Hydrogen Maser frequency standard.
7. HVAC costs are 2003 estimates [1] inflated to 2016 dollars. A potential alternative way to increase the HVAC capacity could be to merge one of the existing HVAC units which is being under used (the BLC is cooled by AHU 1 and 2). The AHU4 serves the computing room, but that room is not fully populated maybe we can use that unit. We could benefit from some expert advice here.

Case 1: AOS Using Existing Space

In this scenario, the new correlator would be housed in the space now occupied by the current correlator. The logistics required in this scenario are more complicated than in any of the others. They violate assumptions 1 and 2 above, but are included because this is the lowest cost option. The HVAC system must be swapped out for a higher capacity one. We estimate that this should take about 60 days, but this estimate should be verified. (There is also the possibility of using the unused capacity of AHU4 to supplement AHU1 and AHU2 which are already supplying the correlator room.) In parallel with this activity, half the existing correlator would be removed and the new correlator would be installed in its place. Once the HVAC system is refurbished, the remaining two quadrants of the existing could be powered up and tested both at the hardware and software levels. Expended time for testing the correlator would be about two days. At that point one or the other correlator could be used, but not both at the same time due to cooling and power limitations. Some means of directing airflow to the operational correlator could be required.

Clearly the biggest advantage of this scenario is the minimization of capital costs. No new space is required. Optical power splitters for the antenna data are not required either. Possibly the HVAC system could be upgraded rather than replaced saving further dollars. The loss in science productivity is about 62 days, dominated by the rough estimate of the time required to replace the HVAC.

There are several disadvantages of this approach. The largest is the lost science time due to longer MTTR (Mean Time to Repair). The higher MTTR results not only from the travel time, but also from the policy of prohibiting travel to the AOS at night and in inclement weather. We estimate loss of science time to be 260 hours per year. At a cost of about $11K per hour this is equivalent to about $59M. Another productivity cost associated with this approach is the lost engineering time associated with the time it takes to travel to the AOS and back, and the possibility of injury to the staff during the trips. A service call to the AOS requires two people and approximately two hours per person are wasted on each trip due to loading/unloading equipment and transit time. Based on recent experience, this amounts to 208 hours per year. Factoring in labor costs, vehicle costs and high-altitude bonus cost, this amounts to 500K dollars over 20 years. The largest real cost associated with this option is the cost differential between cooling the correlator at the AOS as compared to the OSF. It is less expensive to cool the correlator at the AOS because of the free cooling available from the cold environment [1]. This amounts to a real operational cost difference of approximately $2.1M over 20 years. The same system would be installed at either location [1] but a slightly higher installation cost would be incurred for installation at the AOS (~5K).

Case 2: AOS Using Newly Constructed Space

In this scenario, new space would be constructed at the AOS. The space would include room for the correlator and a new HVAC system and is estimated to cost $1.32M, based on the inflated actual construction costs. This approach would require a small amount of down time as splitters are installed in the antenna data paths. This could be accomplished as “rolling down time” where small sets of antennas would be unavailable for a short time. Science down time would be negligible. Both correlators would be continuously available allowing science and commissioning to proceed in parallel.

This approach has the same advantages and disadvantages as Case 1 with two exceptions. First, this approach results in no down-time during installation. Second, both correlators would be available during the time that the new correlator is being commissioned.

Case 3: OSF Using Existing Space

In this scenario, existing space at the OSF technical facility would be converted for use by the correlator. Approximately 33 square meters of laboratory-grade space is required for the correlator racks and 150 square meters of lower-grade space for the HVAC system.

The main advantage of this approach is the lower MTTR due to the proximity of the correlator to the correlator personnel, day and night. The value of the science time saved is approximately $60M. The incalculable value of risk to personnel making weekly trips to the AOS goes to near zero. The capital cost of the HVAC system is decreased slightly. However, the operational cost of the HVAC system is increased significantly, costing an additional $2.1M over 20 years. This is due to the fact that there is much more “free” cooling at the AOS due to the cold environment. Both the high altitude bonus and vehicle costs are zero, resulting in a savings of 35K. Lost engineering time costs are zero resulting in a savings of 400K over 20 years. As with Cases 2 and 4, there is no down time associated with this approach. Another very significant cost of this approach is the need to transport all the data from all the antennas to the OSF. For a bandwidth of 4 times the existing correlator and 4-bit samples, we estimate this capital cost to be $2.0M. Having the antenna signals available at the OSF may provide additional benefits (for example, inputs to a test correlator) which cannot be calculated.

Case 4: OSF Using New Space

In this scenario, new space would be constructed at the OSF technical facility. Approximately 33 square meters of laboratory-grade space is required for the correlator racks and 150 square meters of lower-grade space for the HVAC system.

The advantages and disadvantages of this approach are almost identical to those of Case 3. It has the additional disadvantage of the cost of the new space (estimated at $500K). We assume this space can be built for $200/sq-ft because the requirements are very modest.

To Do List

This section summarizes the things that need to be done to complete this report

- Quantify SEU error rates at AOS versus OSF and their impact on data quality. Add these to the science costs if possible.
- Get the square footage of the OSF to complete the cost/sq-ft estimate

Questions for ALMA Systems People (Nick Whyborn?)

It would be useful to get input from experience people at ALMA on facilities-related items. On the other hand, they have better things to do than continuously be interrupted by us. The purpose of this section is to gather questions to submit to them once we are at a fairly stable point.

- We are looking at four options for the location of a future correlator. These include the four combinations of AOS/OSF and new/re-purposed space. Should any of these *not* be considered?
- Our costs per square foot for new space for the correlator and HVAC system are based on the actual construction costs, inflated to 2016 dollars. Is this a reasonable approach in your view?
- Is there existing space at the OSF that could be re-purposed to hold a correlator housed in 11 racks. It dissipates a lot of power (~225 KW). Space requirements include 33 sq-m for the correlator and 148 sq-m for the HVAC.

References

[1] Correlator Room HVAC Study, STE-20.01.02.00-002-A-REP.pdf, on EDM at http://edm.alma.cl/forums/alma/dispatch.cgi/ipt20docs/docProfile/100073/

Costs associated with travel to the AOS to service the correlator

Number of FTEs required to go to AOS to service the correlator	2	
time wasted/FTE on a trip to the AOS[1]	2	
Time wasted/trip:	4	hrs
Number of trips/yr [2]	52	
Number of years	20	
Hours wasted	4160	
Number of working hours per year per FTE	2080	
Fraction of FTE time wasted	0.05	
Cost of FTE-year	$200,000	
Dollar cost of hours wasted per FTE per year	$10,000	
Wasted labor over 20 years	**$400,000**	
Number of km driven per trip	60	km
Cost of vehicle operation per km	$1	should try to get a better estimate
Vehicle cost over 20 years	**$62,400**	
Cost of high-altitude bonus per person per trip [2]	$18	
Number of people required per trip	2	
Cost of high-altitude bonus over 20 years	**$37,440**	
Total incremental personnel cost over 20 years	$499,840	

Notes:
1 - Includes driving time and time loading and unloading the vehicle
2 - Source is Alejandro Saez

Costs of space for correlator and HVAC for Cases 2 and 4

Need space for 11 racks
Need space to work around racks
Need space for HVAC

			metric equivalent	
			0.093	square meters per square foot
Square feet/rack	7.0		0.6555208	
Square feet/11 racks[1]	90.0	sq-ft	8.37	
Working space	270.0	sq-ft	25.11	
space for HVAC	1600.0		148.8	source: existing HVAC is
Total	1967.0		182.9	

cost of construction				
AOS (correlator space)	673.0	$/sq-ft	62.589	source: Jason Jennings,
AOS(HVAC space)	673.0	$/sq-ft		Perhaps this space can
OSF (correlator space)	361.0	$/sq-ft	33.573	From Jason Jennings: $3
OSF (HVAC space)	361.0	$/sq-ft		Perhaps this space can
college laboratory, US avg	200.0	$/sq-ft	18.6	source: BuildingJourna
AOS total	$1,319,080			both seem high. Space
OSF total	$707,560			

Notes:
1 - 8 racks for the correlator proper (source: Carlson), 2 rack for computers and networking, 1 rack for fiber
2 - rack mount ("earthquake mount")is called out separately
3 - HVAC is costed separately

Jason Jennings info:
Subject:
RE: ALMA costs
From:
Jason Jennings <jjenning@nrao.edu>
Date:
7/22/16 16:58
To:
Rich Lacasse <rlacasse@nrao.edu>
CC:
William Randolph <wrandolp@nrao.edu>

Hi Rich,

Bill and I have been looking into this, and have reached out to some folks at JAO and the AUI Office and we

AOS was approx. $11.1M from back around 2009, so at 1.5% inflation that bring you up to around $12.3M

Total SqFt of AOS is ~18.3k, so you are looking at $673 per sqft at AOS

OSF was 25.0M EUR, so at 1.4 $/EUR (back then) that comes up to $35M USD. With 1.5% escalation we ar

OSF room costs breakdown

Correlator	129960
HVAC	577600

s about the size of the existing correlator room

private communication, see below; average cost per square foot of AOS
be less expensive?
38.8M/10,000 m^2 => $361/ft^2
be less expensive?
l.com
for HVAC drives the cost. Cheaper space for HVAC?

r demultiplexing

e this is what have found in the records:

in 2016.

e up to $38.8M in 2016.

"Productivity" costs of lost science with the correlator housed at the AOS

Capital cost of ALMA	$1,500,000,000	
Operating cost per year	$50,000,000	
Lifetime of ALMA	30	years
30-year operating costs	$1,500,000,000	
Total ALMA cost for 30 yr	$3,000,000,000	
Cost per year[1]	$100,000,000	
Hours per year	8766	
Cost per hour	$11,407.71	
Number of maintenance trips to the AOS per year	52	
Hours of science time lost per trip, day	2	
Hours of science time lost per trip, night	8	
Mean hours lost per trip	5	
Science hours lost per year	260	
Number of years	20	
Science time lost over number of years	5200	
Value of lost science for Case 2	$59,320,100	
Time lost during transition period for AOS existing room		
#days	60	estimate ba
mean fraction of days normally used for science	0.6	guess
total hours lost during transition	864	
cost of science time lost for Case 1	$69,176,363.22	

Cost of transporting current data to OSF in 2016 dollars

Desc.	Qty	Unit cost	total cost	
2:1 splitters	55	95	5,225	source OEQuest.com
4:1 splitters	18	140	2,520	source OEQuest.com
DWDM	18	8700	156,600	source: Doeleman
transceivers (in packaged modules)	880	1150	1,012,000	source: http://www.fs.com/dwdm-sfp-plus-transceivers-sid-191.html
receivers (on daughter cards in correlator)	880	150	132,000	source: various
cables	150	150	22,500	source: guestimate
racks and misc	1	50000	50,000	source: guestimate
Installation/test (4 weeks, 2 FTE)	1	12000	12,000	source: guestimate
Design, Assemble, Document, PAI	1	250000	250,000	0.5 FTE year
Travel	2	5000	10,000	
Contingency (10%)	1		165,285	
Total			**$1,818,129.50**	

Big assumption: Space for this is available at both AOS and OSF
Big question: what to do about transceiver spares?
Note: In the 2022 time frame these costs will go down. Chris Jacques estimates decrease of about 30%

Cost of transport 4x BW and 4 bits in year 2022

Assumption: In 2022, 40 Gb/s tranceivers should be available for the present cost of 10 Gb/s transceivers
The requirement to transmit 4 bits instead of 3 will increase the cost by ~4/3
For the final estimate, simply multiply the above estimate by 4/3
Estimated Cost: **$2,424,172.67**

Cooling Cost Comparison

	AOS	OSF		
Number of Years	20	20		
Capital cost	163637	158637	Note 1	
Maintenance Cost	44056	44056		**operating plus maintenance:**
Operating Costs	1145456	2202801		$1,189,512 2246857
Total 20-year cost	1353149	2405493		
Cost savings over 20 years:	**$1,052,344**	**-$1,052,344**		

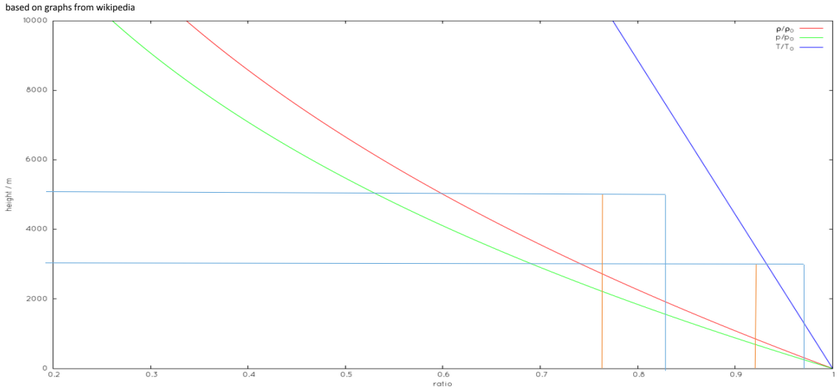
Conclusion: Cooling at AOS saves significant dollars due to operational savings due to use of ambient cooling

Background information:

Based on current weather

location	hPA	in mercury	ratios relative to sea level	ratios relative to AOS	
AOS	556	16.41	0.549407115	1	1.8201439
OSF	722	21.32	0.713438735	1.298561151	1.401662
VA Beach	1012	29.87	1	1.820143885	1

based on graphs from wikipedia



density ratio at 5000 meters = 0.6 + (2/44)0.1 = .605
density ratio at 3000 meters = 0.7 + (18/44)0.1 = .741
ratio of densities @ 3000 m and 5000 m = .741/0.605 = 1.225

pressure ratio at 5000 meters = 0.5 + (15/48)0.1 = .53
pressure ratio at 3000 meters = 0.6 + (43/48)0.1 = .690
ratio of pressures @ 3000 m and 5000 m = .690/0.530 = 1.30 (very close to above from pressure readings on 6/15/2016)

From [1]

Internal load	170	KW
Required Safety Factor	1.20	
Altitude	5075	m
Barometric Pressure	58.9	% of sea level
Desired Room Air Temperature	20	degrees C

$Q = mfr\ C_p\ D_T$
$= 1.085\ cfm\ D_T$
where
Q = heat loss/gain (kw)
mfr = mass flow rate, kg/hr
D = supply air temperature minus return air temperature
cfm = cubic feet per minute
Q = 685,000 Btuh = 1.085 cfm (68 - 52)°F
At 5075m
cfm = 39,500/0.0589 = 67,000 cfm (31,620 l/s)
The intent is to use two air handlers, each capable of supplying 33,500 cfm (15,810 l/s)

From [2]

Use of an "economizer" saves money at AOS, not possible at OSF

At AOS:	"Option 2"	"Option 5"	"Option 6"
Capital cost	83000	115000	91000
Annual Maintenance cost	2000	2500	1500
Annual Operating cost	88000	56000	39000
At OSF			
Capital cost	83000	not doable	91000
Annual Maintenance cost	2000		1500
Annual Operating cost	88000		75000

According to chiller manufacturers, any equipment derating for the chiller condenser fans are largely offset by the gain received from the lower ambient temperatures
In actuality, the lower site will require slightly less airflow and therefore the costs will be slightly less, possibly $1,000 to $2,000 per air handler.
Freight, taxes and import duties not included in above

cost per kw-hr	kw	hr	yearly cost		
	0.1	170	8760	148920	
	0.05	170	8760	74460	M3 probably assumed $0.05 per kw-hr
	0.3	170	8760	446760	real cost at 30 cents per kw-hr
				6	operatingCostCorrectionFactor

inflation factor

per cent per year	3
number of years	13
inflation factor	1.468533713

freight, taxes, import, installation

AOS	30000	estimate based on correlator shipping and installation
OSF	25000	

[1] Correlator Room AC Calculation.pdf, M3 Engineering Technology Corp., Oct 24, 2003 available on EDM at http://edm.alma.cl/forums/alma/dispatch.cgi/ipt20docs/docProfile/100301/
[2] Correlator Room HVAC Study, STE-20.01.02.00-002-A-REP.pdf, on EDM at http://edm.alma.cl/forums/alma/dispatch.cgi/ipt20docs/docProfile/100073/