ALMA Memo # 392

A Cross Multiply Accumulate Unit for FX Correlators

John Bunton
CSIRO, Telecommunications and Industrial Physics, Australia
25 August, 2001

1. Introduction

FX correlators enjoy an advantage over XF correlators because they need fewer computations. However, the amount of storage needed is the same for both. In XF correlators recirculation is used to achieve larger lag ranges over reduced bandwidths. The extra storage needed for the larger lag range is implemented in external low cost RAM. In FX correlators the first stage storage is implemented within the ASIC or FPGA performing the cross multiplication. This RAM is much more expensive than external RAM. This reduces the cost advantage of FX correlators when compared to an XF implementation. In this memo, a frequency channel reordering scheme is described which allows most storage in FX correlators to be implemented in external RAM. This frequency channel reordering is the FX-correlator analogue of recirculation in XF correlators and allows a considerable reduction in the cost of FX correlators.

2. Recirculation in XF correlators

First, let us consider recirculation in an XF correlator. At the highest bandwidth the data rate into the cross-multiply accumulator (XMAC) of an XF correlator is equal to the maximum clock speed. If the bandwidth is reduced so is the clock speed leading to under-use of the XMAC. This capacity is recovered by clocking the low speed data into a temporary store and passing this data through the XMAC multiple times. For each pass of the data through the XMAC one of the inputs is delayed. Thus, each pass of the data through the XMAC generates a set of correlations at different values of delay. If the bandwidth is reduced by a factor N then the number of correlations generated per baseline increases by N. This results in an increase in frequency resolution by a factor N^2.

A second benefit of recirculation is that the extra storage for the increased number of lags can be implemented in cheap external RAM. This is achieved by buffering the lag data within the XMAC at the end of each accumulation. The accumulators in the XMAC are cleared and the cycle starts again. While the new lag data is being accumulated, the buffered data can be accumulated into the external memory. The XMAC can operate at its full potential and on-chip memory is minimised.

3. Channel Reordering in FX correlators

In FX correlators, it is a simple matter to maintain full usage of the XMAC units as frequency resolution is increased. Increase the number of frequency channels without altering the total bandwidth. This is the approach taken by Okumura et al [Okumura 2001]. The total data rate remains unchanged, because the data rate in each frequency channel decreases as the number of frequency channels increases. However, the cost of increasing the number of frequency channels by N is an N-times increase in the storage needed in the correlator. In traditional FX correlators, the data arrives in frequency-channel order and the XMAC units must accumulate into different memory locations on each clock cycle. This forces the memory to be on-chip, which increases costs relative to a recirculating XF correlator.

What is needed is an FX correlator design which has the same memory costs as a recirculating XF correlator. To do this, the XMAC on-chip circuitry must be reduced to a double-buffered multiply-accumulate unit with all full accumulations going to off-chip memory. Figure 1 is diagram showing how this might be implemented.

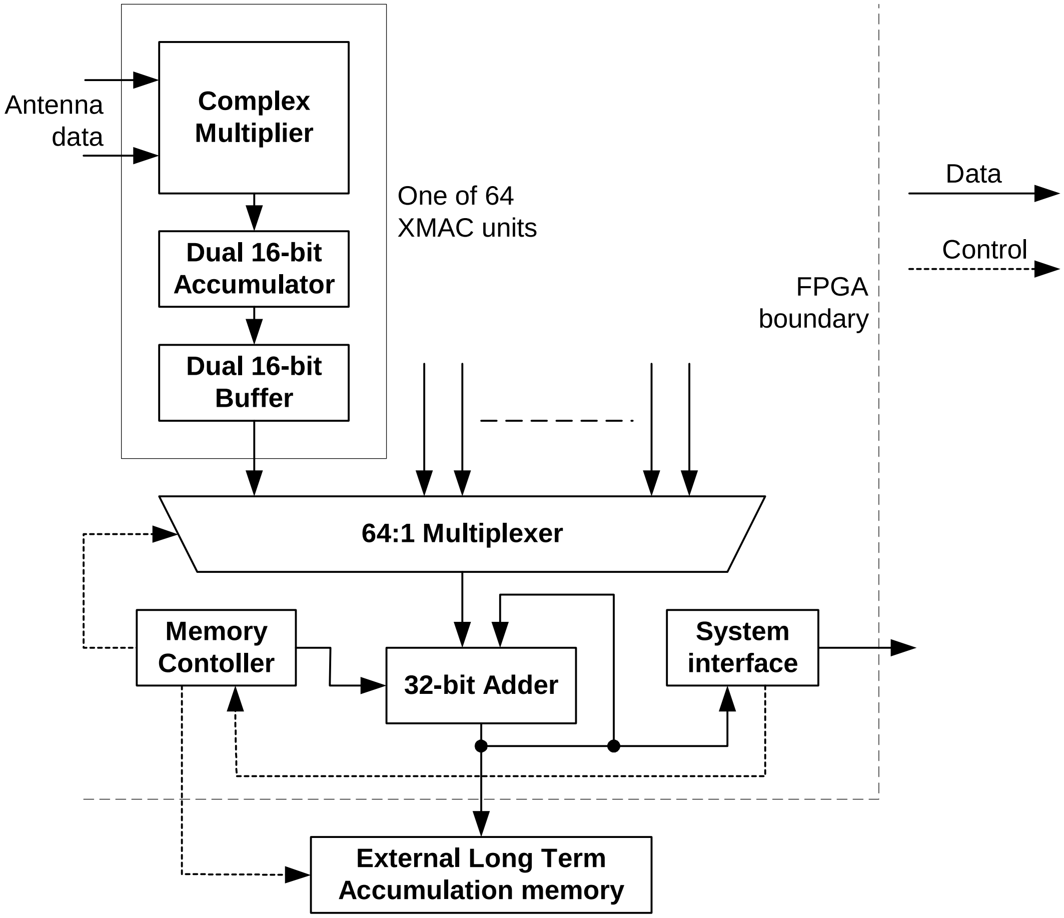

Figure 1 FPGA implementation of 64 double buffered XMAC units with interface to external RAM for long term accumulation.

To allow sufficient time for the buffered data to be accumulated in external RAM the ordering of the data to the XMAC must be changed so that multiple sequential samples for a *single* frequency channel are accumulated at one time. The reordering required can be implemented with a corner turner. This is a versatile device, which has previously been used to implement FFTs [Chikada 1984] and concentrating data from many antennas into a single data stream [Urry 2000]. The corner turner requires storage to enable the reordering of the data. However, a corner turner only allows each data sample to be used once in the XMAC array. When recirculation is used in XF correlators, the data can pass through the XMAC array a number of times. The flexibility to control the number of data passes confers a significant advantage on XF correlators that use recirculation. In FX correlators this flexibility is maintained by using a RAM for storage and a memory controller to perform the reordering. As will be seen this confers the advantage of being able to configure the correlator in many ways, such as full polarisation, single polarisation, recirculation, subarray and variable precision modes.

Unlike recirculation in XF correlators there are no penalties associated with channel reordering, as it is simply a change to the order in which the calculations are done. Thus there are no missed product terms due to samples for a particular lag product being in adjacent recirculation blocks, as occurs in XF correlators.

A simple example illustrates the channel-reordering technique. Consider a 4 channel system that generates at time 1 the frequency sample $f_{1,1}$ $f_{1,2}$ $f_{1,3}$ $f_{1,4}$ and at time 2 $f_{2,1}$ $f_{2,2}$ $f_{2,3}$ $f_{2,4}$ and so on for times 3,4,5,6,7 and 8. The resulting data sequence is
$f_{1,1}$ $f_{1,2}$ $f_{1,3}$ $f_{1,4}$ $f_{2,1}$ $f_{2,2}$ $f_{2,3}$ $f_{2,4}$ $f_{3,1}$ $f_{3,2}$ $f_{3,3}$ $f_{3,4}$ $f_{4,1}$ $f_{4,2}$ $f_{4,3}$ $f_{4,4}$
$f_{5,1}$ $f_{5,2}$ $f_{5,3}$ $f_{5,4}$ $f_{6,1}$ $f_{6,2}$ $f_{6,3}$ $f_{6,4}$ $f_{7,1}$ $f_{7,2}$ $f_{7,3}$ $f_{7,4}$ $f_{8,1}$ $f_{8,2}$ $f_{8,3}$ $f_{8,4}$

These 8 blocks of 4 frequency samples can be clocked through an XMAC unit in 32 clock cycles. This data forms 4 correlations each with 8 accumulations. Instead of clocking the data directly through the XMAC first store it in a 32-word memory and clock it out in the order
$f_{1,1}$ $f_{2,1}$ $f_{3,1}$ $f_{4,1}$ $f_{5,1}$ $f_{6,1}$ $f_{7,1}$ $f_{8,1}$ $f_{1,2}$ $f_{2,2}$ $f_{3,2}$ $f_{4,2}$ $f_{5,2}$ $f_{6,2}$ $f_{7,2}$ $f_{8,2}$
$f_{1,3}$ $f_{2,3}$ $f_{3,3}$ $f_{4,3}$ $f_{5,3}$ $f_{6,3}$ $f_{7,3}$ $f_{8,3}$ $f_{1,4}$ $f_{2,4}$ $f_{3,4}$ $f_{4,4}$ $f_{5,4}$ $f_{6,4}$ $f_{7,4}$ $f_{8,4}$

In the first 8 clock cycles data is accumulated for frequency channel 1. At the end of this time the accumulated datum is transferred to an on-chip buffer memory and the accumulator cleared to start processing frequency channel 2. The buffered datum for frequency channel 1 is now added to the correlation for channel 1 in the external RAM. In this example, eight XMAC units can be serviced by a single external RAM if the clock speed of the RAM is the same as that of the XMAC.

With 2-level complex input data, the accumulator for each XMAC can be reduced to a pair of 4-bit registers. On-chip memory for the 8 accumulators and buffers is 128 bits. Assuming 32-bit accumulators are used for the 4x8 = 32 frequency channels then 1024 bits of storage are needed in external RAM. The on-chip memory required has been reduced by an order of magnitude.

4. Example for a 64 antenna system

4.1. The basic XMAC module

Assume the module is implemented in an FPGA or ASIC with the equivalent of about 5,000 logic cells[1]. The input data is complex and the accumulators are double buffered 16-bit accumulators for a total of 64 logic cells for storage. About half the logic in these logic cells is used for the accumulator logic and the other half is available for the complex multiplier. With 5,000 logic cells 64 complex XMACs can be implemented with sufficient resources left over for control.

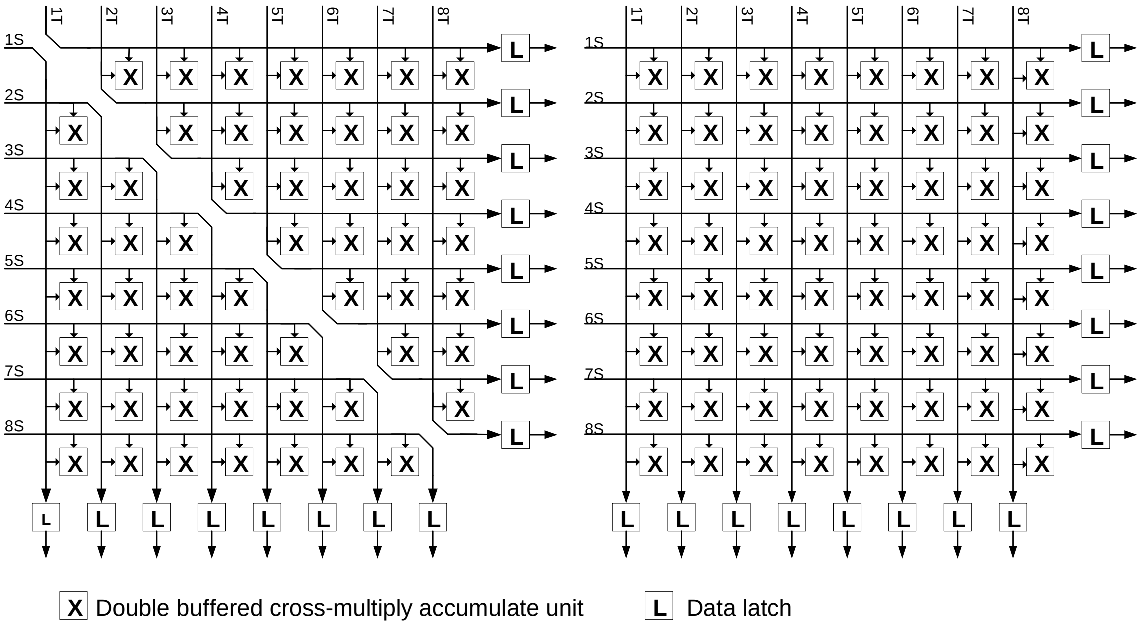

Figure 2 Example of a XMAC array with two sets of 8 inputs, data pass through and output latching. On the left it is shown configured to form the two sets of 28 correlations within the 8 side and 8 top inputs (dual 8 mode). On the right it is configured to form the 64 correlations between the 8 top input antennas and the 8 side input antennas (8 by 8 mode)

Data connection between the 64 XMAC units is shown in Figure 2. In the connection on the right (8 by 8 mode), all correlations are formed between the 8 antennas 1T to 8T on the top and the 8 antennas 1S to 8S on the side. If correlation between antennas within a single set of 8 is wanted identical data could be used for the top and side data but this is wasteful because each correlation is generated twice, excluding auto correlation. Only 56 useful correlations are formed. If auto correlations are generated elsewhere, say in the filterbank, the configuration on the left is possible. This configuration (dual 8 mode) generates the 28 cross correlations for two sets of 8 antennas increasing the usage of the

[1] Here a logic cell consist of a single flip flop and a LUT capable of performing any logic function on 4-bit data

array to 56 XMACs of the 64 available, giving an 88% usage of the available XMACs. For larger arrays the percentage of useful XMACs is even greater.

4.2. Pipelined Arrays of XMAC Arrays - a 64 antenna single polarisation system

The XMAC array, above, is shown with input data passed through the unit and latched at the output. This technique turns the array into a systolic array. It is used in correlators to avoid data skew in large arrays of XMAC units [Urry 1998] and eliminates the need to provide data delays within each integrated circuit [Escoffier 1997]

To avoid confusion this XMAC array will be called an XMAC chip indicating that the circuitry resides in a single FPGA or ASIC. The output latches allow the basic units to be connected together into larger arrays. An example of a 4 by 4 array of XMAC chips is shown below. The array can form all the cross correlations between 64 antennas with two passes of the antenna data through the array. During the first pass, the array is configured in a dual 32 mode shown on the left of the Figure 3. At time 0 data enters the top left hand unit, which is configured in a dual 8-antenna mode. This XMAC chip transfers the top input data to the side output at time 1. It is then correlated against a second set of 8 antennas in the adjoining XMAC during the next clock cycle before the data is passed to the next XMAC unit in the line. After 4 clock cycles the data for the 8 antennas supplied to the top of the first XMAC chip has been correlated against themselves and all other antennas in the top group of 32. After 7 clock cycles all correlations within the top group of 32 antennas has been formed as well as within the group of 32 side antennas. During the second pass of the data through the array the XMAC chips are configured as shown on the right. In this mode (32 by 32 mode) all correlations between the top and side antennas are formed.

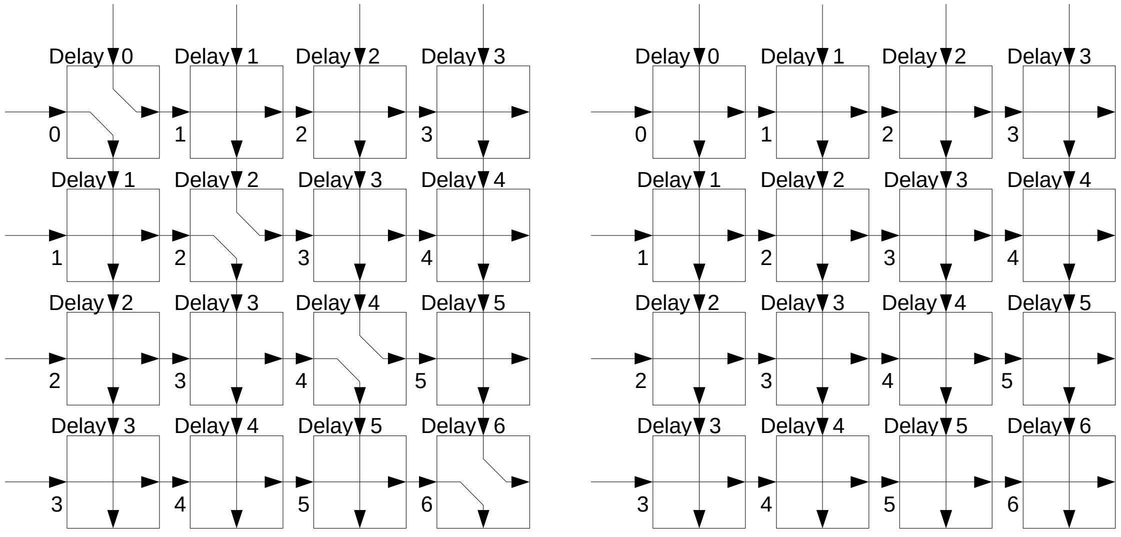

Figure 3 4 by 4 array of XMAC chips showing a dual 32 antenna connection left and a 32 by 32 antenna connection left. Relative delays to individual XMACs are shown.

The total number of correlations formed with the array in dual 32 mode is 12x64 + 4x56 = 992 and in 32 by 32 mode 16x64=1024 for a total of 2016 correlations. Both modes are needed to form all the cross correlations for 64 antennas. Thus each datum from the antennas is used twice. It takes two clock cycles to process each data sample from the filterbanks, once to form cross correlations within a group of 32 and once to form correlation with the second group of 32. With a global system clock of 128MHz this allows 64MHz of data to be processed for each antenna. Cabling into a correlator module is minimised by time multiplexing 64MHz data for two antennas onto the one cable. This reduces the number of input cables to 32 if there is one data stream per cable. Each input must have its data processed in a channel-reordering buffer, which could consist of a memory controller and a 24 bit wide memory. Over 6 clock periods there would be 4 reads supplying 12 bytes of data to the XMAC chip array and 2 writes for the 6 bytes of data coming into the controller from the filterbanks.

The minimum integration time for any one correlation is limited by the time needed to store the buffered accumulation data to external memory. The system must not only accumulate the results but also allow for read out for final storage on tape. If half the memory bandwidth is used for accumulations then at least 128 memory cycles are needed to accumulate the data in the 64 buffers. If the memory can cycle at 128MHz and the channel-reordering buffer can hold 1ms of data then each cross-multiply accumulate unit can process 1024 correlations at the highest bandwidth. With two data passes to form all baselines, this allows 512 frequency channels to be processed. With 64 MHz of bandwidth from each antenna the frequency resolution is 125KHz. Reducing the width of the data bus to the accumulation memory increases the minimum frequency resolution. A reduction by a factor of 4 provides a good match to the frequency resolution of the ATNF 2GHz filterbank [Ferris 2001] when operating at its full bandwidth. This reduction in the width of the data bus to 12 –16 bits reduces the number of frequency channels to 128. For the Japanese FFT processor [Okumura 2001] the individual channels are 2kHz wide which requires a full bandwidth accumulation memory, 48 to 64 bits wide, and a 64ms channel-reordering buffer.

As the bandwidth of the filterbank is decreased, the output clock rate of the data from the filterbank is reduced. This increases the time duration of data stored in the channel-reordering buffer, which in turn increases the number of frequency channels that can be processed. If the bandwidth is reduced by N then the number of frequency channels can be increased by N for an increase in frequency resolution of N^2. For the ATNF filterbank operating with an output bandwidth of 256 MHz each correlator unit processes 8 MHz of bandwidth. The channel-reordering buffer can store 8ms of data allowing 1024 frequency channels to be processed and the resulting maximum frequency resolution is 8kHz. At this resolution, each external long-term accumulator holds 1024 frequencies x 128 correlations per frequency = 128k complex correlations. With 32-bit accumulators, 8 bytes per complex correlation, this requires 2Mbytes of storage if the accumulations are double buffered. The total number of frequency channels possible per 2GHz filterbank are 1024 x (256MHz/8MHz) = 32,768 when the filterbank is operated at a bandwidth of 256MHz. Greater numbers of frequency channels are possible at lower bandwidths or with faster or wider buswidth accumulation memories.

5. Recirculation

The channel reordering structure can also be used to implement recirculation. In the extreme case data for only one frequency channel is stored in the channel-reordering buffers. This is then read out with a variable time delay between the top and side signal of the XMAC array together with a method of routing the data from the side buffers to the top and from the top to the side. This could be achieved by routing the data from the bottom of the array to the side buffers and the left side outputs to the top buffers. A full bandwidth connection is not needed if a small amount of inefficiency is allowed. Instead, pass the data through the XMAC array a number of times with part of the data being transferred each time. Each buffer now holds two sets of data: the original top data A and the original side data B.

After generating the zero lag correlation a change to the operating sequence is needed. A 4-phase sequence is used with the individual XMAC chips always in the 8 by 8 mode. The phases are (top A and side A delayed) (top A and side B delayed) (top B and side A delayed) (top B and side B delayed). These 4 phases generate all the correlations for both senses of delay for each antenna pair. As an example, consider the first phase of the sequence and correlations formed between antennas 2 and 3 of a group. Antennas 2 and 3 with no delay drive the top inputs and antennas 2 and 3 with a delay drive the side inputs. Internally correlations between (antenna 2 with 3 delayed) and (antenna 3 with 2 delayed) are both formed.

On average, it still takes 2 clock cycles to form all the correlations for one lag. Thus, a single 64kHz channel can be recirculated 1000 times providing a frequency resolution of 64Hz. As this approach is identical to recirculation in XF correlators, the resolution improves as the square of the reduction in input bandwidth (Reducing the input bandwidth to 2kHz gives a further 32^2 improvement in resolution to 0.062Hz). Recirculation can also be applied to multiple frequency channels at the same time. For example eight 64kHz channels could be processed each generating 128 lags for a frequency resolution of 500Hz over a 0.5MHz band.

6. Alternative Implementations for Dual Polarisation

The above design does not consider the possibility of dual polarisation observation with full Stokes parameters. The calculation of full Stokes parameters results in a doubling of the total number of correlations. As well as the correlations for the two polarisations, the two cross-polarisation correlations must be formed. It is seen that all correlations are formed if each signal is correlated with all other signals, in effect a 128-station correlator is built, where each polarisation from each antenna is considered to be a separate station. This can be done by using 64 basic 8x8 XMAC chips. When compared to two separate single-polarisation systems there is no increase in the size and number of channel-reordering buffers. Only the XMAC array increases. This increases the cost by 50% in a system where the channel-buffers are half the cost. When cross polarisation data is not needed, the system can be operated in a dual 64-station mode. In this mode all cross correlations for a given set of data is formed in one clock cycle, doubling the bandwidth and allowing the correlator to handle a bandwidth of 128MHz. The problem to be solved with this design is handling all the interconnections for the high data rate needed. There would be 128 inputs into the correlator. This would pose considerable problems in a system based on standard copper cables.

A more compact and manageable unit is produced if the size of the XMAC array is maintained at 16 XMAC chips. To generate full polarisation data the correlator must be provided with data for both polarisations for all antennas doubling the number of signals. Without cross-polarised correlations, this doubles the number of clock cycles to process the data. Adding the cross-polarised correlation doubles the number of clock cycles again. Thus, a 128MHz correlator can process two sets of 16MHz data from each antenna. This data rate can be handled by 16 byte-wide cables operating at 128MHz. Results for the 8x8 and 4x4 arrays of XMAC chips are summarised below. The XMAC phases are the number of clock cycles needed to form a single accumulation for a full set of correlations for a single data set for one frequency channel.

Table 1 Data rates and input bus width for a correlator unit using an 8 by 8 array of basic XMAC units

	Bandwidth MHz	Byte wide input signals	XMAC phases
Full Polarisation	64	64	2
Two Polarisations	128	128	1

Table 2 Data rates and input bus width for a correlator unit using a 4 by 4 array of basic XMAC units

	Bandwidth MHz	Byte wide input signals	XMAC phases
Full Polarisation	16	16	8
Two Polarisations	32	32	4
Single Polarisations	64	32	2

Physically the correlator must accommodate the maximum number of input signals. With byte-wide 128MHz cables, this means that the 8x8 array has 128 input cables and the 4x4 array 32. The use of the 4x4 XMAC chip array makes the data input to the correlator unit manageable with copper based interconnects. Improvements in the capacity of high speed copper and optical interconnects could see a move to 8x8 XMAC chip arrays.

The disadvantage of the 4x4 array is an inability to form all correlations when top and side channel-reordering buffers hold data for half the antennas. The top and side inputs to the XMAC array can take 32 signals at a time. Thus, when two polarisations are processed, the channel-reordering buffers for a single side of the XMAC array must store the data as two sets of 32 antennas, say all 64 antenna for a single polarisation. The cross correlations between 64 antennas of a single polarisation can be formed by using the XMAC chip array in repeated dual 32 mode on multiple sets of 32 antennas. However, this results in redundant accumulations and a loss of efficiency.

To prevent any inefficiency the two sets of 32 antennas must be processed once using the XMAC chip array in its 32 by 32 mode. While one group of 32 antennas drives the side inputs the other 32 must drive the top inputs. This requires a coupling between channel-reordering buffers. The routing required to do this is shown in Figure 4.

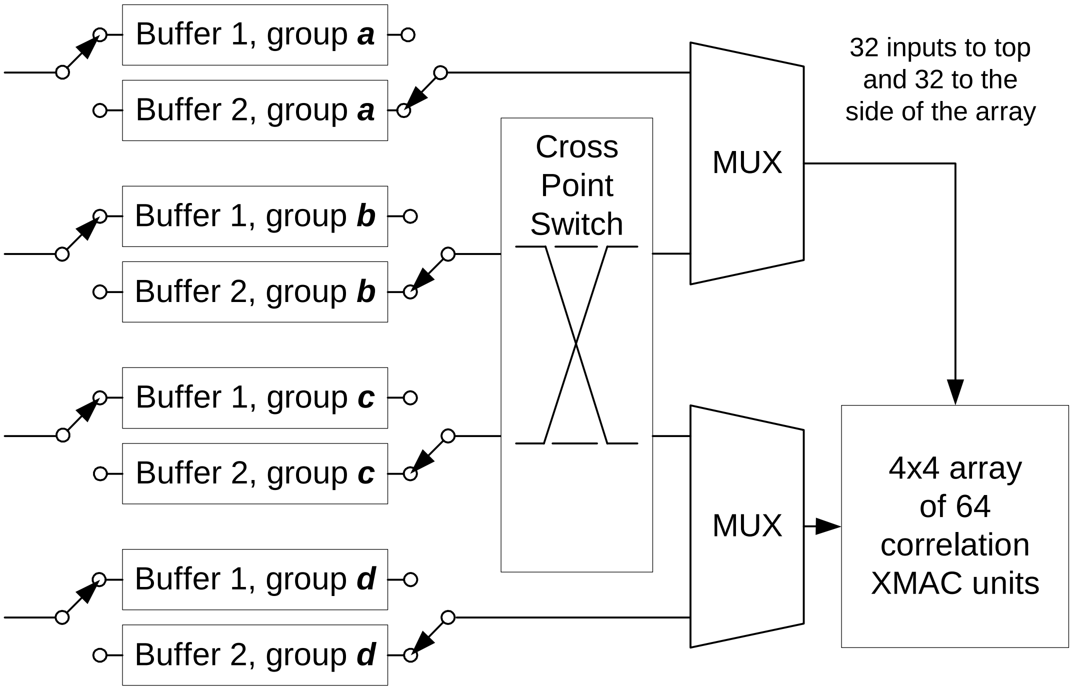

Figure 4 Buffering and routing for a channel-reordering buffer for a correlator based on a small array of basic XMACs chips.

As an example group **a** in the figure above could buffer half the left hand polarisations and group **b** the other half. Likewise, groups **c** and **d** would each buffer half the right polarisation. For full-polarisation correlations 8 accumulations must be formed in the XMAC unit to give all 8064 (32.63*4) correlations (the array can form at most 1024 correlations at any one time). Two accumulations with the XMAC chip array in dual 32 mode are needed to form all the correlations between antennas within the four groups, (**a,a**) and (**c,c**) then (**b,b**) and (**d,d**). A further four accumulations form the correlations between left and right hand polarisation, groups (**a,c**), (**a,d**), (**b,c**) and (**b,d**). The final two correlation use the crossbar switch and form correlation between groups (**a**,**b**) and groups (**c**,**d**). The design is easily adapted to systems with greater numbers of antennas. With 256 antennas it takes 128 separate accumulations to form full polarisation data and the system can process 1MHz of bandwidth. The increased number of accumulations means that it now takes at least 128x128 = 16384 clock cycles to form the accumulations and store them to external memory. With a 1ms channel-reordering buffer (128k samples) the highest frequency resolution is 125KHz when the system is processing a full 1MHz of bandwidth and the XMAC takes two clock cycles to store accumulations to external memory. The frequency resolution can be improved by reducing the time to store accumulations to external memory or by increasing the length of the channel-reordering buffer.

A practical implementation of the channel-reordering buffer shown in Figure 4 would see the data divided into bit-planes. Each bit plane provides 64 bits of data to the 4 by 4 array of basic XMAC chips. When operating in single-polarisation mode each input to the correlator module provides data for 2 antennas so the input data is 32 bits wide (64 antenna/2). Thus the buffer memory is at least 96 bits wide and on average in 3 clock cycles it supplies 192 bits of data to the 4 by 4 array of basic XMAC units and stores 96 bits of input data. To hold 1ms worth of data, double buffered, the memory must have 2buffers x 32inputs x 128Kbits = 1Mbyte. By use of suitable buffering a 128 bit wide memory running at 96MHz can be used to provide the necessary data rates. This memory can be formed from eight 64k x 16bit SRAM memories. This memory and an FPGA control to it is estimated to cost of about US$40 for each bit-plane. With 8 bit-planes the silicon cost of the channel-reordering buffer is US$320 per 4 by 4 XMAC chip array.

7. Sub-arraying, mixed resolution and high accuracy modes

The ALMA specifications specify the possibility of operating the telescope as a number of subarrays. The 4 by 4 array of basic XMAC chips naturally lend themselves this. For two subarrays of 32 antennas the buffers can be made to hold the left polarisation on one side of the array and the right polarisations on the the other side. In dual 32 mode, the array can form the correlations for both 32-antenna subarrays and both polarisation in two accumulations. The formation of cross polarisation terms would take two more accumulations with the array of XMAC chips in 32 by 32 mode. This allows each correlator unit to process twice as much bandwidth. For a full system operating as two subarrays of 32, this allows a doubling of the bandwidth over which full polarisation data can be formed. When formed into 4 subarrays of 16 antennas it takes only 2 accumulations, with the XMAC chip array in dual 32 mode, to form full polarisation data. At higher levels of sub-arraying it is easiest to form all correlations within a group of 16 dual polarisation antennas and discard correlations between separate subarrays.

The full correlator system also handles mixed frequency observation because each correlator module processes a different frequency band. There are 64 modules per GHz of bandwidth (full polarisation). Each of these modules can be processing data at a different frequency resolution. In addition, the data going to each module need not be contiguous. This allows a module to process a number of separate bands.

High accuracy operation can be achieved at lower bandwidths. If the basic data precision is say 4 bits then an 8-bit multiply can be formed from four 4-bit multiplies. All necessary multiplications are formed if the 64 antenna 8-bit data is thought of as 128 antenna 4-bit data. For antennas **a** and **b** the partitioning results in two low 4 bit components **al** and **bl** and the two high 4-bit components **ah** and **bh**. Forming all correlations between these components generates the product **al.bl, al.bh, ah.bl** and **ah.bh.** Forming the sum **al.bl+16(al.bh + ah.bl)** + 256 **ah.bh** gives the required 8-bit product (assuming each product is of the correct form unsigned-unsigned, signed-unsigned etc). This move from 4 to 8-bit data reduces the bandwidth each correlator module can handle by a factor of four but allows high accuracy operation.

8. Autocorrelations

Autocorrelations are not computed by the XMAC chips. Instead, autocorrelations will be calculated in the filterbanks. This increases the complexity of the filterbank by a small amount without compromising the basic simplicity of the XMAC chips. To accommodate autocorrelations, the XMAC chips would need pairs of cross-multiply elements on the diagonals of the arrays so that in dual 32 mode the autocorrelations for the signals from the top and sides could be calculated simultaneously. The array would have 272 elements and the extra hardware would be wasted in 32 by 32 mode. Having the autocorrelations in the filter bank also allows pre-and post-quantisation autocorrelations to be calculated (it is assumed the data is requantised at the output of the filterbank) The pre-quantisation autocorrelations allow a precise determination of the input power spectrum and the post-quantisation autocorrelations provide a correction factor needed to calibrate the cross correlations.

9. Costing

For a 64 antenna system using the ATNF filterbank a 16MHz full polarisation correlator system would have memory and controllers for 8 bit-planes each costing about US$40 or a total of US$320. The 16 XMAC units with external RAM for the storage of results are estimated to cost US$40 each or a total of US$640. Total silicon cost is estimated to be US$960 for a full polarisation correlator that process 16MHz of bandwidth. Per GHz of bandwidth the silicon cost is US$60,000 and for the 8 GHz of bandwidth required by ALMA the cost is US$480,000. If the cross polarisations are not needed the correlator can process 16GHz of bandwidth for both polarisation from each antenna. For the Japanese FFT processor, the channel-reordering buffers are as much as 64 times larger which would increases the cost by an order of magnitude. For both filterbank implementations the cost can be reduced by the use of DRAMs, and ASICS or Hardwired FPGAs.

This cost of the new design should be compared to the cost of an FX XMAC implemented in an FPGA and using FPGA memory for storage. In the Virtex-II XV2C6000 there are 144 RAM blocks each of which can be configured as 36-bit by 512 word memory. For complex data, this allows 72 baselines to be implemented. Even at a clock rate of 256MHz the number of FPGAs needed is (8GHz/256MHz) *(2016 baselines/72) * 4 polarisations = 3500. At about US$3000 per FPGA this gives a silicon cost of US$10m. Even at this cost, the correlator is limited to at most 4k frequency channels per 2GHz filterbank. In the channel-reordering approach, the number of channels is only limited by external memory and large numbers of channels can be implemented cheaply.

10. Conclusion

The use of channel reordering greatly reduces the cost of cross multiply units in FX correlators while still maintaining great flexibility. This, coupled with the flexible 2GHz filterbank being developed by the ATNF, provides a high performance solution to implementing the ALMA correlator.

11. Acknowledgments

The author would like to thank Dick Ferris, Colin Jacka and Don McLean for their critical appraisal of this memo.

12. References

Chikada Y., et al., 'Digital FFT Spectro-Correlator for Radio astronomy', in Indirect Imaging (ed J.A. Roberts), Cambridge University Press, pp387-404, 1984

Escoffier, R. 'The MMA Correlator', ALMA Memo 166, July 1997, http://www.alma.nrao.edu/memos/html-memos/abstracts/abs166.html

Ferris, R.D., Bunton, J.D., Stuber, B., "A 2GHz Digital Filterbank Correlator", The SKA: Defining the Future, Berkeley California, July 9-12, 2001. http://www.skatelescope.org/meeting/

Okamura, S.K., Chikada, Y., Momose, M., Iguchi, S., "Feasibility Study of the Enhanced Correlator for 3-way ALMA I.", ALMA memo 350, February 2001, http://www.nrao.alma.edu/memos/

Urry, L.W., 'A New Correlator Architecture Eases Wiring Problems', BIMA Technical Memo #71, November 1998. http://bima.astro.umd.edu/memo/memo.html

Urry, L.W., 'A Corner Turner Architecture', ATA Memo 14, November 2000, http://astron.berkeley.edu/ral/ATAMemo14.pdf