Engineering Subarrays

The CDR design

Sometime after CDR it became apparent that the design presented was not good at implementing engineering subarrays. At the time it was conceived it was a good design. The technology to be used was fresh and it limits not well known. This meant a fairly conservative clock rate was used. The signals from the SKA office had an emphasis on cost constraint and different classes of computation had different resource usage. This pointed to every FPGA implementing all classes of computation: filterbanks, correlators and beamformers.

The resulting design was a crystalline structure, an 8x6x6 array of FPGAs. Each FPGA could accept data from LFAA, each would implement filterbanks, correlators and beamformers. The FPGAs also implement all data routing, avoiding the cost of network switches. Every FPGA was involved in every stage of data routing. This brings us to the question of how to implement an engineering subarray?

The crystalline structure of the design is predicated on a fixed routing structure. It did not allow the inclusion of hot spares and engineering subarray would need to use one or more of the 288 FPGAs in the crystal and effectively introduce a defect into that crystal. To see the effect of this defect we can look at the lost input data products that will not now be processed. For the correlator these lost products are:

- Input data from two LFAA stations (300MHz, 384 coarse channels)
- On the first level cross connect 16 stations for (37.5MHz lost)
- On the second level cross connect 96 stations (6.25MHz lost)
- On the final cross connect 512 stations (1.04MHz lost)
    - This bandwidth is spread over 48 coarse channels, 22kHz is lost from each coarse channel

If all these losses are added up it is seen the correlator is still operationally capable. But engineering subarrays must also be independent of astronomy subarrays. This is not possible because, for example, 3% of the bandwidth is lost for one eighth of the coarse channels. At least one if not all astronomy subarrays will have coarse channels where data is lost. An engineering subarray consisting of a single FPGA is not independent of other subarrays. Loss at other levels of the cross connect will have a greater effect but on fewer subarrays. Also a single FPGA cannot, by itself, emulate the full suite of FPGA functionality. In particular, data routing is not implementable at all.

By changing the operation of the cross connect it is possible to form an independent engineering subarray but the minimum set of FPGAs is a 6x6 plane (there are 8 such planes). For astronomy there are 7 planes and the first cross connect is only between these planes. No cross connect is implemented to the 8th plane. This is achieved by simply making any packet crossing to or from the 8th plane invalid. Operation of the two sets of planes can now proceed independently. We now have an independent engineering subarray that implements the full suite of FPGA functionality, excluding the first level of cross connect. But to achieve this independent engineering subarray we have lost 12.5% of capability. We have an engineering subarray that is close to emulating system functionality but the correlator and beamformer are no longer operationally capable.

The Separate Filterbank Design (ECP-200029)

In an effort to find a better solution to providing an engineering subarray CBF realised that a previous design approach (pre-CDR) would be better. The previous design followed a more traditional approach where the filterbanks were implemented separately to the correlator and beamformer. This led to the design proposed in ECP-200029. This design has 32 FPGAs implementing the filterbanks on LFAA data. Any single filterbank FPGA processes data from 3% of the LFAA stations and can be used for an engineering subarray testing filterbank functionality. The correlator and beamformer remain operationally capable.

The correlator and beamformer are separate entities so subarrays for one function does not affect the other. Each of the correlator and beamform functions is subdivided into 12 independent processing systems. Any of these can be allocated to an engineering subarray and the loss of bandwidth is 8% which corresponds to 4% loss of sensitivity and operational capability. It is possible to test correlator and beamformer functionality as engineering subarrays and still be operationally capable.

If finer granularity is desired each of the 12 subsystems is cabled to accept extra FPGA boards. These extra boards would act as hot spares. In normal operation data can be duplicated to them and a fully function correlator implemented for engineering subarray purposes. This duplicated FPGA could also output data to be compared to that generated by equivalent astronomy FPGA. Only the testing of inter FPGA routing needs to have astronomy FPGAs dedicated to an engineering subarray.

It is seen that the Separate Filterbank approach can implement engineering subarrays and maintain the telescope as operationally capable. Some further benefits accrued from the change. It was now possible to include hot spares or equivalently add extra compute capability incrementally (until each subrack is full). These can have duplicate station data sent to them to implement a parallel engineering subarray for the correlator and beamformer functions.

Atomic COTS (ECP-200039)

Atomic COTS goes one step further. Data routing is separated from signal processing and the system includes fully functional hot spares. Correlators can become beamformers - there are no fixed communication or processing elements. A second correlator can be created, or even a second beamformer. Comparisons between different versions of firmware and software is straight forward. LFAA data can be duplicated to the hot spare(s) which acts as the engineering subarray. The engineering subarray can include as much system hardware as desired. Each Alveo can execute a different FPGA firmware kernel with independent Tango software. Most modes can completely implement correlator or beamformer functionality in a single hot spare. But some fine resolution zoom modes and larger substation subarrays require more. It doesn't matter what array assembly is being deployed, it is always possible to create an engineering subarray.