Refined EVLA WIDAR Correlator Architecture

NRC-EVLA Memo# 014

Brent Carlson, October 2, 2001

National Research Council of Canada,
Herzberg Institute of Astrophysics

ABSTRACT

This memo is a technical blueprint for further development of the WIDAR correlator for the EVLA. This document is not intended for the casual reader, and the assumption is made that the reader is familiar with all of the preceding memos in this series, and with correlator signal processing. In some sections, particularly concerning the Baseline Board, many functional and performance details are included to provide a concrete baseline plan for implementation. This helps to establish a high level of confidence that some of the more 'exotic' correlator functions can indeed be realized. In other sections, less detail is provided since functionality is reasonably straightforward.

The concepts presented in this document were developed over a period of about two years with valuable input and feedback from many NRAO engineers, scientists, and users during several meetings. Nothing in this document is 'etched in stone', but it presents at least one coherent plan so that correlator development can proceed to its next phase of implementation.

Table of Contents

List of Figures

1 Executive Summary of Correlator Capabilities

This section provides an overview of the capabilities of the correlator described in this document. Some of these capabilities exist to meet EVLA science and system requirements, and others evolved from the chosen signal processing and architecture.

- 16 GHz of bandwidth per antenna arranged as eight 2 GHz basebands (or four pairs). Bandwidth can be traded-off for number of antennas without correlator modification. Inputs support flexible allocation of basebands and baseband widths.
- Each 2 GHz baseband input can be on a different delay center on the sky for flexible multi-beaming. Different baseband inputs could process the same data (with a front-end switch) with a completely different delay center.
- 16,384 spectral channels per baseline at the widest bandwidths. Up to 256k spectral channels per *cross-correlation* on 2 basebands using “recirculation”. “Wideband recirculation” provides more spectral channels at wide bandwidths with sensitivity losses.
- High performance data output capability. The nominal 40-station configuration could produce ~3 Gvis/sec. The extreme 40-station configuration could produce ~12 Gvis/sec. However, the output data rate will largely be determined by back-end processing capability, and data volume handling limits.
- ≤4 or 8-bit sampling and/or correlation for high spectral dynamic range and spectral purity.
- 144 digital filters per station generate sub-bands for flexible deployment of spectral resources and efficient wideband correlation. Each sub-band can be on a different delay center on the sky, within a maximum offset of about 0.25^{o} from the baseband delay center (e.g. for a 2000 km baseline). “Radar-mode” sub-bands can be as narrow as 30 kHz.
- Pulsar processing: 2 banks of 1000 time bins/sub-band/baseline (up to 65k bins with back-end S/W binning); bin width as narrow as 15 μsec; independent timer for each baseband; independent gate for each sub-band.
- LO shifting for anti-aliasing and RFI robustness. Decorrelation of sampler-generated self-interference, strong narrowband harmonics, and many intermodulation products. Decorrelation of interleaved sampler artifacts.
- Real-time or tape-based VLBI capable.
- Reconfiguration/expansion capable.
- Simultaneous interferometer and phased-VLA operation. Delivered with a 1 GHz phased bandwidth, expandable to a full 16 GHz without re-design or replacement of existing hardware or cabling.
- Many interferometer and phased-array sub-arraying possibilities.
- All-digital sub-sample delay tracking to ±1/32nd of a sample.

2 System Overview

The EVLA correlator hardware components consist of three main digital printed circuit boards, a few small backplanes and interconnect modules, and high-performance cabling. Using these few components as building blocks it is possible to construct a correlator of virtually any size and configuration. A simplified correlator module connectivity diagram is shown in Figure 2-1**.** All modules in this diagram will be described in detail in following sections. All fiber-optics components and cards are not within the development scope of the correlator and will be developed by NRAO.

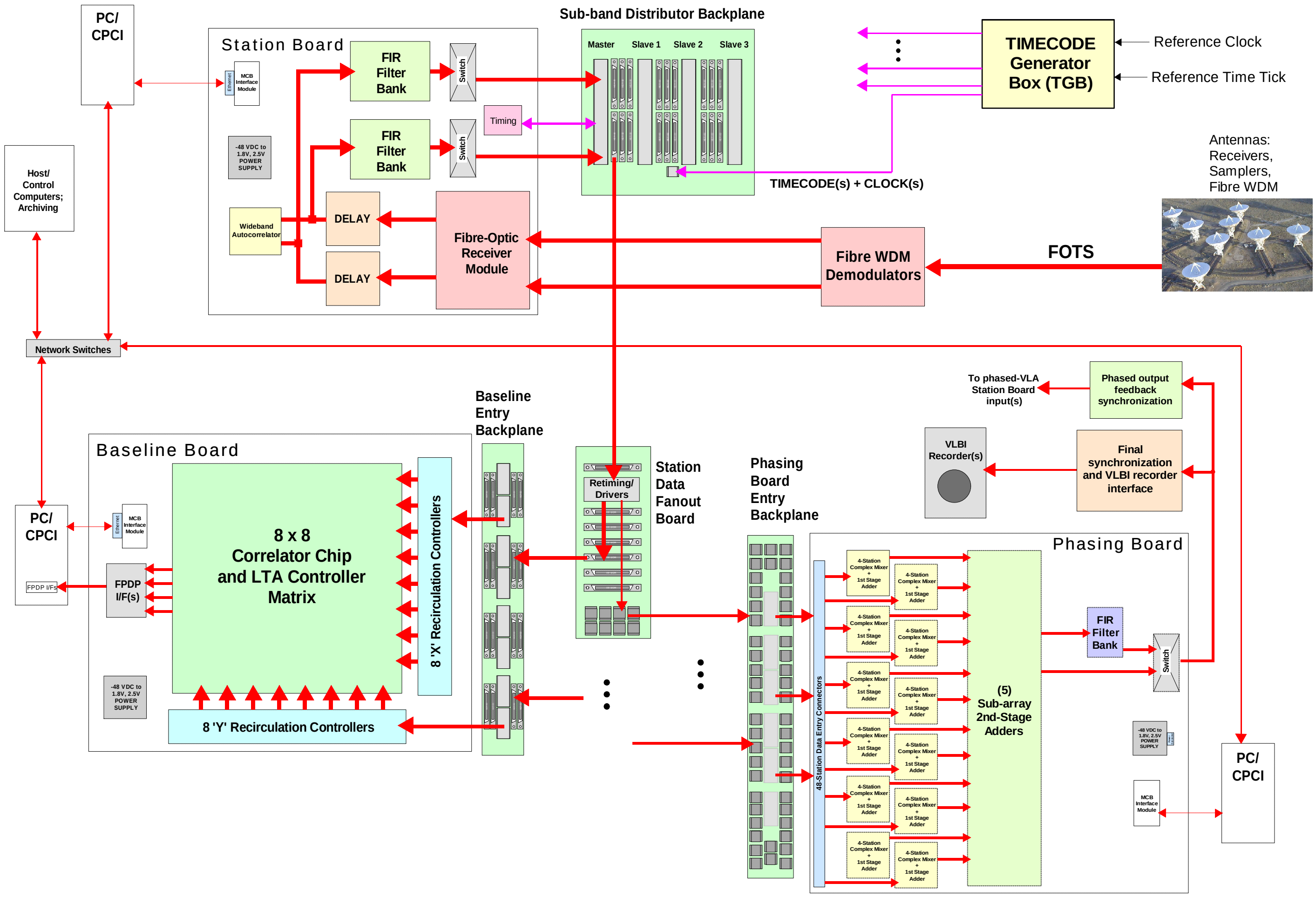

PC/
CPCI
Station Board
Ethernet
MCB Interface Module
-48 VDC to 1.8V, 2.5V POWER SUPPLY
Wideband Autocorrelator
FIR Filter Bank
FIR Filter Bank
Switch
Switch
Timing
DELAY
DELAY
Fibre-Optic Receiver Module
Sub-band Distributor Backplane
Master
Slave 1
Slave 2
Slave 3
TIMECODE Generator Box (TGB)
Reference Clock
Reference Time Tick
TIMECODE(s) + CLOCK(s)
Antennas: Receivers, Samplers, Fibre WDM
FOTS
Fibre WDM Demodulators
Host/ Control Computers; Archiving
Network Switches
Baseline Entry Backplane
Baseline Board
8 x 8 Correlator Chip and LTA Controller Matrix
8 'X' Recirculation Controllers
8 'Y' Recirculation Controllers
FPDP I/F(s)
FPDP I/Fs
Station Data Fanout Board
Retiming/ Drivers
Phasing Board Entry Backplane
To phased-VLA Station Board input(s)
Phased output feedback synchronization
VLBI Recorder(s)
Final synchronization and VLBI recorder interface
Phasing Board
48-Station Data Entry Connectors
4-Station Complex Mixer + 1st Stage Adder
(5) Sub-array 2nd-Stage Adders
FIR Filter Bank
Switch
-48 VDC to 1.8V, 2.5V POWER SUPPLY
MCB Interface Module
PC/
CPCI

Figure 2-1 Simplified correlator module connectivity diagram. The three main boards are the Station Board, the Baseline Board, and the Phasing Board. Data and signal flows are as indicated by the red arrows.

Data from the antennas arrives via fiber-optic links where it is wavelength demodulated before being presented to mezzanine cards on the Station Boards. On these cards, the fiber-optic signal is demodulated into electrical signals for use by Station Board electronics. Each "station input" in the correlator consists of four Station Boards: one "master" input and three "slave" inputs. The master Station Board is the one that generates all of the timing, model, and control signals for downstream processing, whereas the slave Station Boards are only used for data generation. Aside from these differences, each of the four Station Boards' functions is the same. Each Station Board handles two, 2 GHz sampled basebands—also referred to as a baseband pair. The Station Board "Delay" mezzanine card compensates for wavefront geometric delay as well as delay through the fiber-optic system. Data then goes to the sub-band FIR filter banks, the output of which is 16 (with provision for 18) sampled data streams no longer in de-multiplexed parallel form as it was going into the filters. This data goes through crossbar switches before going to the Sub-band Distributor Backplane, which passively re-arranges the data so that there are 16 (with provision for 18) sub-band cable outputs. Each sub-band cable output contains data, timing, model, and synchronization information for one sub-band from all 8 basebands from one station. All real-time information required for the down-stream Baseline Boards (recirculation, phase-binning, dumping, phase models, delay models) is generated on the Station Boards and flows with the data on each sub-band cable. Data gets distributed and fanned-out to all of the Baseline Boards and the Phasing Boards via Station Data Fanout Boards and data routing backplanes.

On the Baseline Board, there are 8 'X-station' and 8 'Y-station' inputs—each input being data from one sub-band cable from one station. The input data is resynchronized and formatted for transmission to a row or column of correlator chips by the 8 'X' and 8 'Y' Recirculation Controllers. The 8x8 matrix of correlator chips correlate data and respond to commands coming from the Recirculation Controllers. After integration, and on command from Recirculation Controllers, the data is read out of the correlator chip by its own dedicated LTA (Long-Term Accumulator) Controller and saved in LTA RAM. Although having one LTA Controller for each correlator chip seems extreme, it offers significant performance advantages and is cost-effective since a relatively small (and inexpensive) FPGA can be used. When enabled by an on-board FPDP (Front Panel Data Port) scheduler, LTA data is transmitted via FPDP to an external computer (PC) for further processing. The data on the Baseline Board is not handled by a microprocessor so there are virtually no bottlenecks to data flow off the board.

On the Phasing Board, data for one sub-band[1] from all antennas enters via the Phasing Board Entry Backplane. This is the same data that goes to the Baseline Boards only it is rearranged so that only one sampled data stream (one sub-band of one baseband) and associated timing/synchronization information is contained on one cable. Thus, each Phasing Board sums antennas for one sub-band of one baseband. Data is summed in two stages to keep on-board data path widths within device capabilities. In the first-stage, data from antennas are summed in groups of 4. Each antenna's data is complex multiplied before complex addition to remove the Doppler shift and the frequency shift required by the WIDAR technique. There are 5, second-stage adders—each one being the output of one sub-array. After second-stage addition, the complex data is combined using the Hilbert transform FIR, the second part of the digital single-sideband mixer. Details and test results are found in [9]. The final summed output is available in normal

[1] Although the goal is to phase two sub-bands, or a sub-band pair on every Phasing Board. This will be done if determined to be practical at the detailed design stage.

sub-band "wide" mode, or it can be filtered with on-board FIRs to generate more, smaller, sub-bands for VLBI recording.

The figure shows control PC or CompactPCI computers that control the main boards via mezzanine MCB (Monitor & Control Bus) Interface Modules. The current plan is to use the MCB mezzanine card that NRAO is developing for other array systems for this module, and that the communications to the external computer be via 100 Mbit/sec Ethernet. Data out of the Baseline Board is transmitted on a FPDP interface to external data handling computers that are shown in the figure as PCs or CompactPCI boxes. However, any appropriate back-end computer with a FPDP interface could be used. Refer to section 7.4.1 for a more complete discussion of the computing configuration.

2.1 Flexibilities

The correlator architecture is very flexible and there are a number of dynamic tradeoffs that can be made that may not be evident from the above description. Some of these "flexibilities" are as follows:

- **Tradeoff number of antennas for bandwidth**. Since each baseband input on each Station Board has its own delay compensation, it is possible to use more Station Boards per antenna to increase bandwidth, or use fewer Station Boards per antenna to increase the number of antennas processed but with decreased bandwidth. Implementing this flexibility dynamically requires a fiber-optic switch in front of the Station Boards, but it does not require any internal correlator rewiring.

- **Tradeoff bandwidth for number of beams on the sky**. Each baseband input can be used to place a beam anywhere on the sky. This can be useful for post-correlation interference cancellation where it is desired to place one beam on the radio source, and one beam on the interference source *using the same data*. Within a baseband, each FIR filter can place a sub-band beam within ~0.25^o (depending on the maximum baseline) of the baseband beam.

- **Deploy spectral channel resources as desired**. Each sub-band can be any width and placement within the baseband within its "slot" constraints [0]. The switch on the output of the Station Board can route the same data to multiple sub-band correlators and so there are numerous ways that spectral channel resources can be allocated to sub-bands. With recirculation, sub-bands narrower than 128 MHz see an (inversely proportional) increase in the number of spectral channels available to them.

- **Tradeoff baseband bandwidth for number of basebands**. Each baseband input can handle 2 GHz of total bandwidth at 4 bits per sample. This requires a "data highway" that is 64 bits wide, at 256 Mbits/sec each. This data highway can be used for multiple narrower baseband inputs in varying combinations.

- **Data routing supports expansion**. There is a linear increase in the number of Baseline Boards that a particular sub-band cable must be fed to with increasing number of stations. The design permits expansion without having to replace the existing infrastructure.

- **Data routing allows different correlator configurations**. Because sub-band data is routed to Baseline Boards with cable, it is possible to configure the data routing to support varying configurations. (Examples: big antenna multi-beams, with small antenna single beam; little-big antenna correlations, but not little-little antenna correlations.)

- **Unlimited interferometer sub-arraying**. Each sub-array can be operated with completely independent parameters within the capabilities of the hardware since each station is processed independently.

- **Phased sub-arraying can be different for each sub-band**. Each Phasing Board can be set independently of other Phasing Boards.

3 Station Board

The Station Board shown in the module connectivity diagram of Figure 2-1 is where all station-based processing happens in the correlator. Each Station Board processes two, 2 GHz sampled basebands and there are four Station Boards for each "station input" into the correlator. A detailed block diagram of the Station Board is shown in Figure 3-1.

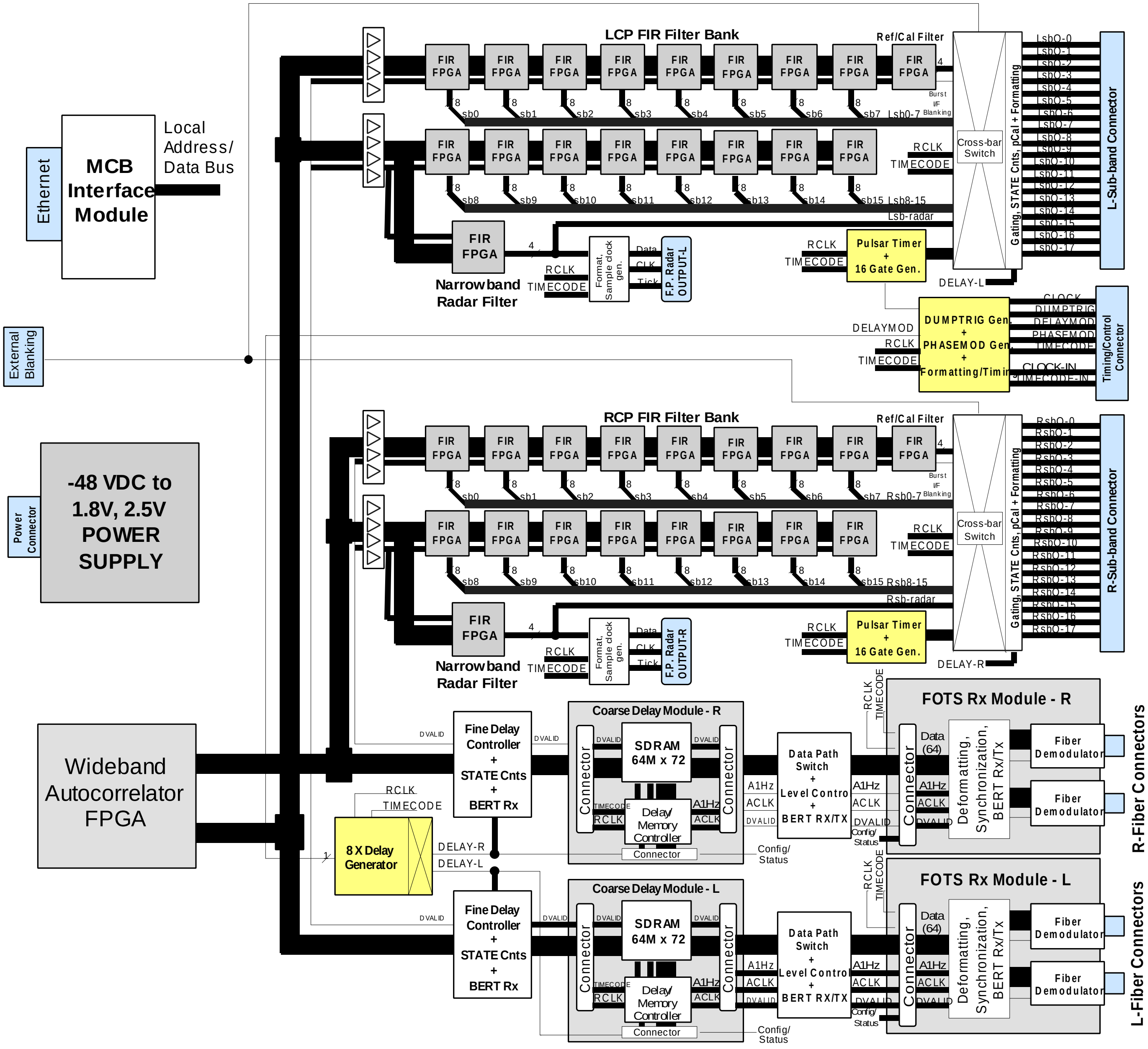
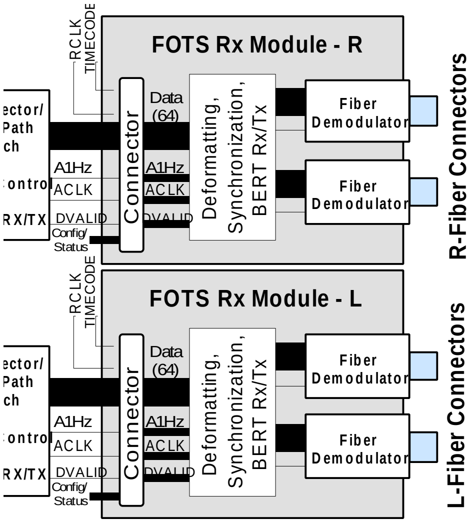


Figure 3-1 Detailed Station Board functional block diagram. Data enters via the FOTS Rx Modules, is delayed to compensate for wavefront delay, is filtered into sub-bands with digital filters, and then exits via a switch for further processing. A detailed description of each block is given in the following sub-sections.

The Station Board receives data from an antenna via a FOTS receiver module, compensates for wavefront delay, digitally filters the wideband data into sub-bands using FIR filters, and then formats the data for further downstream processing. The Station Board also generates all of the additional information such as DUMPTRIG, DELAYMOD, and PHASEMOD that travels with the data for further downstream processing. These additional signals are described in detail in section 4.1.1. Each block in Figure 3-1 will be described in more detail in the following sub-sections.

3.1 FOTS Rx Module(s)



This mezzanine module will be developed by NRAO as part of the FOTS (Fiber-Optic Transmission System) for the EVLA and will plug into the Station Board motherboard. There could be one module for each (2 GHz) baseband input (as shown in the figure) or there could be one module for both inputs. The physical input to the module is fiber via blind-mate fiber connectors. There may be additional inputs—for monitor and control purposes independent of the Station Board—and these must use blind-mate connectors as well, otherwise the Station Board can not be inserted as planned.

The module contains the fiber receivers and all of the circuitry for word alignment and error detection/monitoring that are not of concern to the Station Board but are required to ensure link integrity. The module also contains test vector/BERT (Bit Error Rate Test) generation circuitry that allows testing of the module's connection to Station Board circuitry. The method used for invoking this testing capability is TBD.

The signals at the mezzanine card interface connector (which mates with the Station Board) contain 64 bit streams at 256 Mbits/sec each, a synchronous clock, and a synchronization time tick. The clock and data are synchronous and recovered from the fiber signal. The clock and data are generally not phase synchronous[2] with Station Board clocks and signals. The data are nominally arranged as 16 time-demultiplexed sampled data streams, with 4 bits/stream for a total sample rate of 4 Gs/sec[3]. Nominal functional timing is illustrated in Figure 3-2. Performance timing parameters, the number of data valid lines present, and the exact functional and physical interface definition are TBD. Note that other arrangements permitting more, narrower basebands are possible but the clock will always operate at a nominal 128 MHz. If lower bit rate data is present, then data streams will change states in accordance with their sample rates, but synchronous to

[2] Or even phase stable because of receiver phase locking and because this could be data from a VLBI recorder.

[3] More correctly it is 4.096 Gs/s, but referred to as 4 Gs/s throughout the document.

the 128 MHz clock, and time-aligned to the PPS (1Hz) time tick. TIMECODE may be provided to the module for future interface considerations.

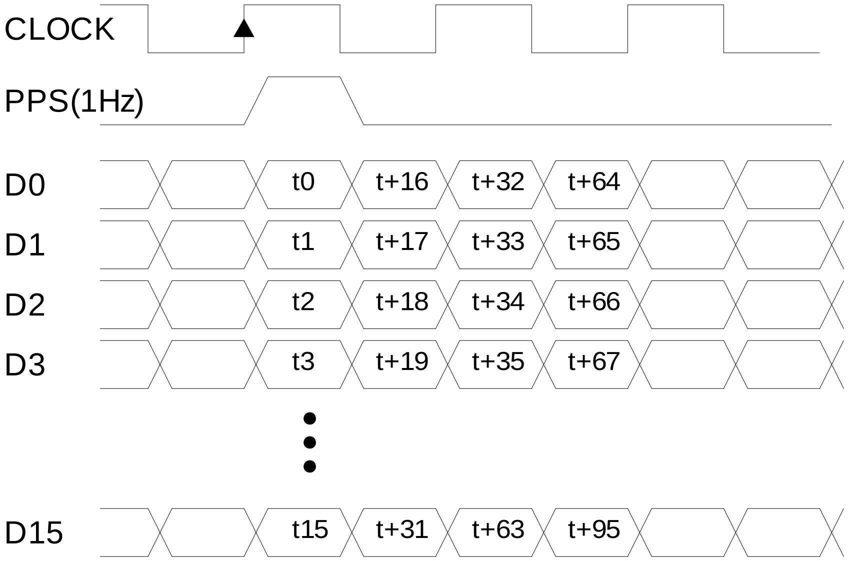
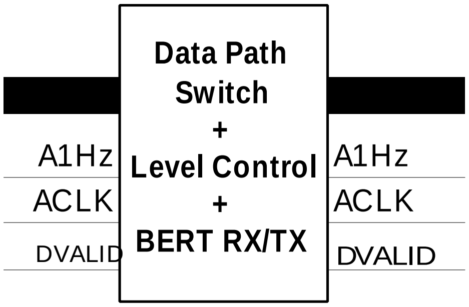


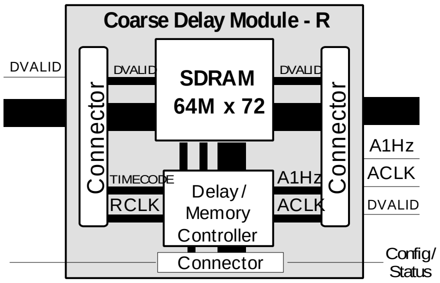
Figure 3-2 Nominal functional timing for the FOTs Rx Module to Station Board interface. The data is synchronous to a clock provided to the mezzanine card. Time slot allocations are as indicated in the figure and are bit times at the 4 Gs/s rate. Data at different rates will have different time slot assignments. The PPS (1 Hz time tick) epoch is for t_o alignment independent of time slot allocations or actual sampled data rates. This ensures that receiving circuitry and any necessary lower-rate shift clock generation are always resynchronized every PPS.

3.2 Data Path Switch



This block allows BERT testing of the connection to the FOTS receiver module, the ability to switch data paths (for data rearrangement or duplication), and the ability to change bit encoding (in case the encoding does not match the correlator’s internal encoding). As shown in Figure 3-1, there is one of these for each of the two baseband inputs. However, increased flexibility will be available if both of these are incorporated into one FPGA. This decision will be based on pin count and cost.

3.3 Coarse Delay Module

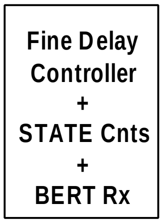


The Coarse Delay Module is a mezzanine card[4] used for wavefront delay compensation. It does this by inserting delay into the entire 16 word wide (64 bit) data path. This compensates for delay to within 16 samples at 4 Gs/s. Final delay to within +/-0.5 samples at 4 Gs/s is accomplished in the following Fine Delay Controller block. If different baseband bandwidths are used (i.e. on the 64-bit data highway), then each will be delayed the same amount in terms of absolute time and hence the correct amount in their respective bit times. Different baseband input arrangements will require modification of the fine delay logic in the following Fine Delay Controller.

The bulk of the delay memory will be in SDRAM since this is the only affordable way of introducing enough delay for the longest earth baselines. Because of the nature of the SDRAM, dual-port SRAM buffers in front and behind the SDRAM will be required. These dual-port buffers can be on the controlling FPGA(s). A 64M deep (576 Mbytes) delay is planned and is enough for 78,000 km of delay (ignoring data transmission delay) at the speed of light, *c*. If the transmission line from the antenna has a 0.5*c* speed, then the maximum delay (and maximum baseline) is 1/3rd of this or 26,000 km. If the transmission speed is slower, then the maximum baseline is decreased accordingly. 26,000 km is enough for any earth baselines and for moderate space baselines. The goal is to design the module so that additional SDRAM SIMMs can be added if necessary. A 72-bit word width is planned to accommodate the 64-bit data highway, some data valid lines, the 1 Hz time tick, and perhaps some embedded error check data. Delay control information will come from the Delay Generator block.

The Coarse Delay Module also contains logic that allows the on-board MCB microprocessor to write test vectors into it for testing downstream hardware. Since these test vectors are in memory, they can be simulated astronomical data complete with Doppler and delay as described in [2], providing a powerful testing facility. Exact operation requires further definition.

3.4 Fine Delay Controller



This block controls fine delay to within +/-0.5 samples of delay at 4 Gs/s. (N.B. Very fine delay to ±1/32 samples is accomplished with sub-band phase offsets on the Baseline Board.) It accomplishes this mainly by shifting/swapping data streams and inserting or removing one sample of delay. The block also contains wideband data statistics (state counts, and

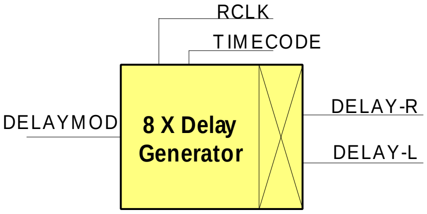
[4] A mezzanine card is planned. However if all of the delay that could ever be required is present—as is now planned to allow very long baseline connections—then the need for a mezzanine card is not that evident.

power measurement) as well as a BERT receiver for testing connectivity through the data path up to this point (most importantly through the Coarse Delay Module). Although shown as one block for each baseband in Figure 3-1, it may be advantageous to include this in one device for added flexibility. This will depend on number of I/O, cost, and number of outputs required to drive the downstream digital filter banks.

3.5 Wideband Autocorrelator

This function, probably implemented in an FPGA, is to obtain the wideband autocorrelation function for all four possible products (RR, LL, RL, LR) on the wideband data before any digital filtering. This will aid in diagnostic checking and determining where interference is on the wideband signal. It also serves as a double-check to ensure that downstream digital filters are operating properly. The number of spectral channels available, and the sensitivity loss incurred are TBD, but it will operate as a "synthetic" autocorrelator [2] since it is only used for diagnostic purposes and so some sensitivity loss is unimportant.

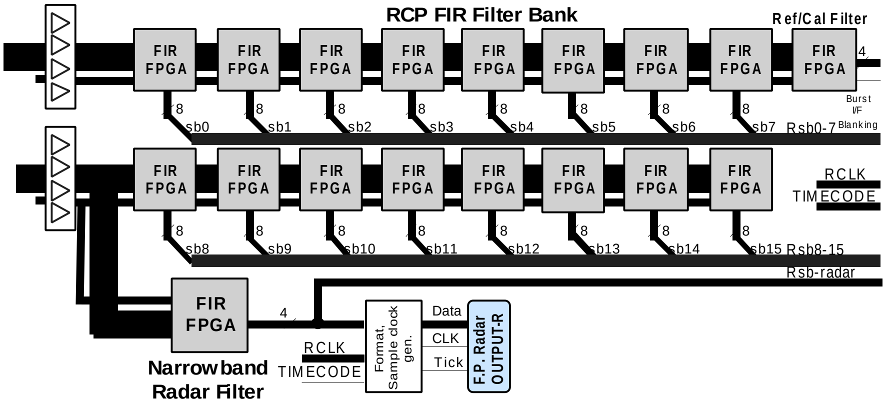
3.6 Delay Generator



The Delay Generator block contains 8, 32-bit linear frequency synthesizers that generate delay for the Station Board. Synthesizers are updated on 10 millisecond epochs embedded in the TIMECODE signal. In the case of the Station Board plugged into the master slot of the Sub-band Distributor Backplane (see Figure 2-1), delay for all 8 basebands is generated and merged to form the DELAYMOD (section 4.1.1) signal that travels with the data for very fine delay correction on the Baseline Board. Thus, the master Station Board has all 8 synthesizers active, and the slave Station Boards only have 2 synthesizers active for their own two basebands. This rather asymmetric arrangement (kludge) is required so that there are no active electronics on the Sub-band Distributor Backplane. An alternative arrangement may be possible whereby each Station Board generates its own delays that get fed to the master Station Board via the backplane for final DELAYMOD generation. The choice of which method to use will be decided during detailed design.

3.7 FIR Filter Banks

There is a bank of (poly-phase) FIR filters for each of the two basebands on the Station Board. Each FIR filter is implemented in an FPGA for flexibility and in-system programmability. Each bank consists of 18 FIRs—16 "standard" FIRs, one FIR (in a larger FPGA) for very narrowband capability, and one reference FIR filter. Changing tap coefficients—and hence filter characteristics—does not require the FPGAs to be re-booted (or "loaded with new firmware"). However, since they are in-system programmable, new designs can easily be downloaded at any time (with the appropriate software support, of course). These new designs may do other things like use two-stage filtering, change the number of lookup table (LUT) bits, or mirror changes to the input baseband configuration



The standard FIRs will provide about 512 taps with 4-bit data, and ~75% more taps with 3 bits and a cosine symmetric filter. One de-scoping/cost-saving option[5] is to use 2-bit wideband data (for the widest total bandwidth operation), put two filters in every FPGA, and only populate the board with ½ the number of FPGAs at production time. This is entirely feasible since with 2-bit data, each LUT can do two, 2-bit taps, where only one, 4-bit tap can normally be done. An 8-bit data path out of each FIR is provided to allow for this possibility (i.e. requantization for both filters must be to 4-bits to avoid excess sensitivity loss). The target FPGA for this filter is the Xilinx XC2V1500-5.

The "Ref/Cal Filter" is used as a roaming filter to find a part of the wide baseband where there is no interference. This will allow the sub-band cross-power spectra to be normalized and immune to time-variable interference (but it is *not required* to "stitch" the sub-bands together) [0]. The Ref/Cal Filter, and probably the other filters as well, contain a timer and two accumulation bins for acquisition of power measurements before requantization. These measurements are necessary to stitch the sub-band spectra together, and the two bins allow accumulation of power synchronous with antenna noise diode switching for system noise calibrations. This timer will be synchronized to TIMECODE and its epoch and period will be set by controlling software.

The "Narrowband Radar Filter" is a larger FPGA that provides 2048 taps in two stages [8]—necessary for the 30 kHz bandwidth that must be extracted from the 2 GHz input. The raw output from this filter is available on a front-panel connector for connection/capture by external equipment if desired. Since this radar mode FPGA is also

[5] Suggested at the August 28, 2001 correlator meeting in Socorro.

in-system programmable, it could be configured in other ways as desired. One of these ways might be as a poly-phase FIR/FFT filter bank with sample-rate conversion to eliminate aliasing and sub-band boundary sensitivity losses. This is attractive, but it may be problematic because of the speed requirements of the fast multipliers required in the FFT, sampler demultiplexer requirements for sample rate conversion, and loss of sub-band tuning flexibility. This option will be kept open in the design though, if the number of signals going into the downstream cross-bar switch is not excessive or costly.

3.7.1 Sub-band Multi-beaming

Since each sub-band FIR is separately configurable, each one can be any bandwidth and placement[6] in the wideband just by changing the tap coefficients and changing the output decimation factor[7]. Each filter's independence can be put to even better use by additionally allowing each sub-band to have its own delay and phase model. This enables each sub-band to form an independent beam on the sky—effectively permitting a dynamic tradeoff between bandwidth and number of beams. This may be useful on longer baselines where the input bandwidth is restricted because of data transmission costs, and where it is desirable to image as large a field of view as possible. This can also be used to (simultaneously) form multiple phased-VLA beams.

To ensure that sub-band multi-beaming comes at no extra cost, each sub-band beam should be within ~0.25° of the baseband beam (depending on baseline) so that the relatively small internal dual-port memory of the FPGA is used *before* filtering. Performing the delay before filtering in an identical fashion to the way baseband delay is implemented, allows sub-band digital sub-sample delay tracking so that the final delay precision—including baseband delay error—is ±1/16 of a sample. Restricting the sub-band beam offset from the baseband beam means that the sub-band delay buffer need only compensate for the *difference* between the sub-band beam and the baseband beam, a quantity much smaller and changing much more slowly than the baseband beam delay across the entire array.

A block diagram of the circuitry required in the sub-band FIR FPGA is shown in Figure 3-3. The essential elements and restrictions of sub-band multi-beaming are summarized below:

1. Use on-chip dual-port memory to eliminate cost increase. With an XC2V1500 Xilinx FPGA, a 32 μsec delay buffer is available (i.e. 8k x 64). With this buffer, the maximum multi-beam baseline is restricted to about 2200 km with a 0.25° sub-band beam offset (from the zenith, the worst case). With a 10,000 km baseline, the maximum beam offset is ~0.055°. These calculations are based on the following equation:

$$t_b - t_s = \frac{B}{c} \cdot \left(\cos(q) - \cos(q + \Delta)\right)$$

[6] Within sub-band slot and decimation restrictions (i.e. 1/16, 1/32, 1/64 etc).

[7] In itself, just an interpretation of the output sample stream by downstream hardware.

Where: τ_b - τ_s is the difference between the baseband delay and the sub-band beam delay, B is the baseline, *c* is the speed of light, θ is the baseband beam angle to the radio source, and Δ is the sub-band beam offset.

2. The chip must include a 32-bit point-slope frequency synthesizer that tracks the sub-band beam to baseband beam delay difference. This is a small amount of logic, but incurs additional real-time software overhead.

3. The changing phase (due to changing delay) at the middle of the sub-band must be tracked by including it in the phase model (PHASEMOD) for the sub-band since it is unwieldy to track this phase as is done with the baseband delay-to-phase lookup table (Figure 4-10). If discrete delay jumps are restricted to occurring on PHASEMOD update boundaries (i.e. every 10 milliseconds), then phase can be precisely tracked with no significant coherence losses. With a 2200 km baseline and a 0.25° beam offset, $d(\tau_b\text{-}\tau_s)/dt$ is a maximum of 9 samples/sec (i.e. at the 4 Gs/sec sample rate) for a sidereal source. This very small sample rate will ensure that there is negligible coherence loss when the delay jump is restricted to 10 millisecond boundaries. That is, the station discrete delay error will be a maximum of 0.5 + 9/100 = 0.590 samples yielding a sub-sample error of 0.590/32 samples. The net *baseline* delay error, including baseband delay error will be: ±1/32 ±0.59/16 samples, or ±12.3°. This amounts to a maximum coherence loss at the edge of the sub-band of 0.8%.

If sub-band multi-beaming is used for in-beam calibration on very long baselines (10,000 km) and a larger beam offset is required than the memory in the XC2V1500 can provide, then the radar-mode's larger FPGA and its subsequent larger delay memory could be employed.

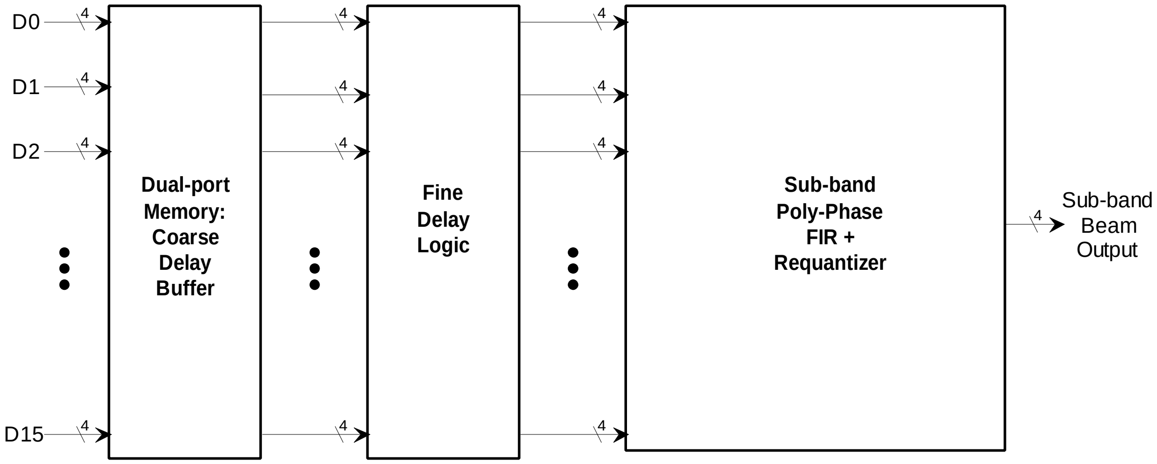

D0
D1
D2
D15
4
Dual-port Memory: Coarse Delay Buffer
Fine Delay Logic
Sub-band Poly-Phase FIR + Requantizer
Sub-band Beam Output

Figure 3-3 FIR filter with additional elements required for sub-band multi-beaming. The on-chip Dual-port Memory buffer performs coarse delay to within 16 samples at 4 Gs/sec and the Fine Delay Logic performs delay to within +/-0.5 samples at 4 Gs/sec in a similar fashion to the baseband delay. Restricting the sub-band beam offset to within ~0.25° of the baseband beam ensures that the on-chip delay buffer is small, and that the sub-band delay relative to the baseband delay is changing slowly.

An alternative to the above sub-sample delay tracking scheme is to perform digital delay tracking *after* FIR filtering. In this case, sub-sample delay tracking is performed by updating the FIR tap coefficients as delay changes. This can be unwieldy and will require asynchronous interrupts to the controlling CPU so that tap coefficients are updated on every 1/16th of a sample change in delay. A tap coefficient buffer is also required in the FIR filter that could consume additional chip resources (depending on how quickly new coefficients are switched in). Nevertheless, this may be the only acceptable solution in the case where the input data is not filtered into many sub-bands and therefore cannot utilize the previously described method.

3.7.2 7-bit Requantization and Correlation

The requirements for 7-bit requantization and correlation have been studied in [1]. However, since the data path for each sub-band from the Station Board to the Baseline Board is restricted in width to 4 bits, it is necessary to restrict the bandwidth available for correlation. There are a few ways of doing this, but probably the best method is to restrict the sub-band bandwidth to ½ its normal maximum when requantizing to 7 bits. The essential elements when 7-bit requantization and correlation are chosen is as follows:

1. The maximum sub-band bandwidth is 64 MHz.

2. The 8-bit output of each FIR filter is used to transport the data to the output cross-bar switch. Only 7 bits of this is used.

3. The output cross-bar switch multiplexes the 7-bit data @ a 128 MHz clock rate to 4 bits (LSN[8]) and 3 bits (MSN) at 256 MHz for transmission to the correlator. The MSB of the MSN is used for data valid flagging, overcoming a previously stated limitation when requantizing to 7 bits.

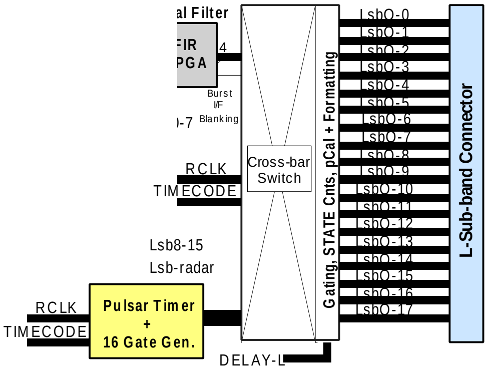
4. The Recirculation Controller (see section 4.1) demultiplexes the data and sets up the data highways to perform all of the necessary distributed arithmetic. Recirculation is used so that only a factor of 2 reduction in the number of spectral channels is realized.

Note that the maximum baseband bandwidth that can be correlated with 7 bit requantization is 1 GHz. Since 7-bit requantization will normally be used when 8-bit initial quantization is used, the baseband bandwidth is restricted to 1 GHz anyway, and so this not an additional performance degradation.

The above 7-bit requantization data transport method will also support 7-bit phasing on the Phasing Board with no additional data path requirements. It will just require reconfiguration of the Phasing Board FPGAs by either downloading a new logic configuration (firmware), or by setting control register bits.

[8] Least significant nibble (4 bits).

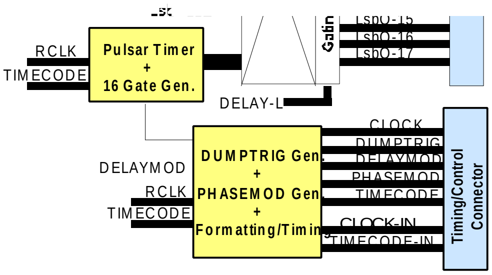
3.8 Output Cross-Bar Switch and Pulsar Timing



This block performs the basic function of allowing any of the 18 Station Board outputs (for each baseband) to be connected to any of the 18 FIR outputs. Additionally, this block performs pulsar gating, acquires sub-band requantizer statistics (state counts and power), and extracts sub-band phase-cal. There will be a minimum of one phase-cal extractor (for each baseband) that can be time-multiplexed across sub-bands. Finally, this block embeds identification information into the data streams for extraction by downstream hardware.

The pulsar timer and gate generators each contain one 32-bit point-slope frequency synthesizer that can be updated every 10 milliseconds. The gate generator is capable of generating multiple independent gates with different epochs and durations but synchronized to the timer. Thus, gates can track the pulse as it moves across frequency space (sub-bands). An output from the timer also goes to the DUMPTRIG generator block to support pulsar time/phase binning. The operation of DUMPTRIG and how it controls time/phase binning is defined in section 4.1.5.

3.9 Formatting and Timing



This block generates DUMPTRIG, DELAYMOD, PHASEMOD, CLOCK, (and regenerates TIMECODE). The format and functionality of these signals is defined in detail in section 4.1.1. DUMPTRIG controls downstream dumping of correlator data, PHASEMOD contains phase models that are used to perform downstream phase rotation, and DELAYMOD contains real-time baseband delay models for very fine delay tracking using a delay-to-phase conversion. All of these signals travel with the data to downstream hardware for correlation and phasing processing. It is not yet defined how these signals are generated, but they are synchronized to TIMECODE. Only these signals from the Station Board plugged into the master slot of the Sub-band Distributor Backplane make their way to downstream processing.

When pulsar phase/time binning is used, DUMPTRIG is synchronized to the pulsar timer epoch which is itself synchronized to TIMECODE (i.e. timer coefficients are loaded on TIMECODE 10 millisecond epochs). In this case, careful synchronization of DUMPTRIGs on every Station Board is required.

3.10 Miscellaneous Functions

The Station Board also contains the following miscellaneous functions:

- External blanking. A front panel connector with an internal pullup resistor is provided that allows an external signal to blank data (i.e. turn data valid off). This is an asynchronous signal that may be useful for real-time interference blanking.

- Downstream Station Data Fanout Board programming. The Station Data Fanout Board (section 6.2) will contain an FPGA that must be loaded with a configuration bitstream and must have its programming state and simple good/no-good state monitored. These functions can be provided by using the 3 spare differential pairs on the MDR-80 connectors that carry all of the signals, timing, and data out of the Sub-band Distributor Backplane (section 6.1). The use of these spare lines is TBD.

- Temperature and voltage monitoring. Temperature and voltage monitors will be on the Station Board and can be read by the CPU on the MCB interface module.

- Dead-man thermal protection and remote power control. The DC-DC power supplies on the module contain built-in thermal overload protection. When these power supplies shutdown, or when there is a problem with them, an output signal is asserted. This signal will be routed to a rear-entry connector and eventually to a control computer where it can be monitored. This same line can be used to remotely power-cycle the power supplies on the board.

4 Baseline Board

This section presents a very detailed definition of the Baseline Board design. This level of detail is provided to ensure that a solid baseline design that meets the high performance requirements of the correlator is available. A block diagram of the Baseline Board is shown in Figure 4-1 and for clarity a 2x2 correlator chip ‘slice’ of the board is shown in Figure 4-2. A physical layout of the Baseline Board is shown in a following section.

The Baseline Board contains an 8x8 array of 64, 2048 complex-lag correlator chips. A square array of chips is used because each array will correlate one “parallelogram” of baselines in a large baseline matrix [0]. Each row of correlator chips is fed with data from one ‘X’ station using a Recirculation Controller FPGA. Columns are fed with ‘Y’ station data in a similar fashion. In this design, each correlator chip has its own dedicated LTA Controller FPGA and 256 Mbit SDRAM. This arrangement is capable of meeting the high-speed readout requirements demanded of narrow pulsar phase binning—with 1000 phase bins in 2 memory banks—and recirculation using available and affordable high-speed DPSRAM memory.

The MCB Interface Module is a small mezzanine board that allows an external computer to configure and monitor all of the devices on the Baseline Board. The MCB module contains a CPU that accesses the devices on the board via a dedicated 8-bit (or 16-bit) memory mapped bus. The required address space and bus width is TBD. Communication to the outside world is via 100 Mbit Ethernet. The baseline plan is to use the MCB mezzanine card that NRAO is developing for other array systems since the design should meet the low performance monitor and control requirements and since this module will then be standard across all array systems.

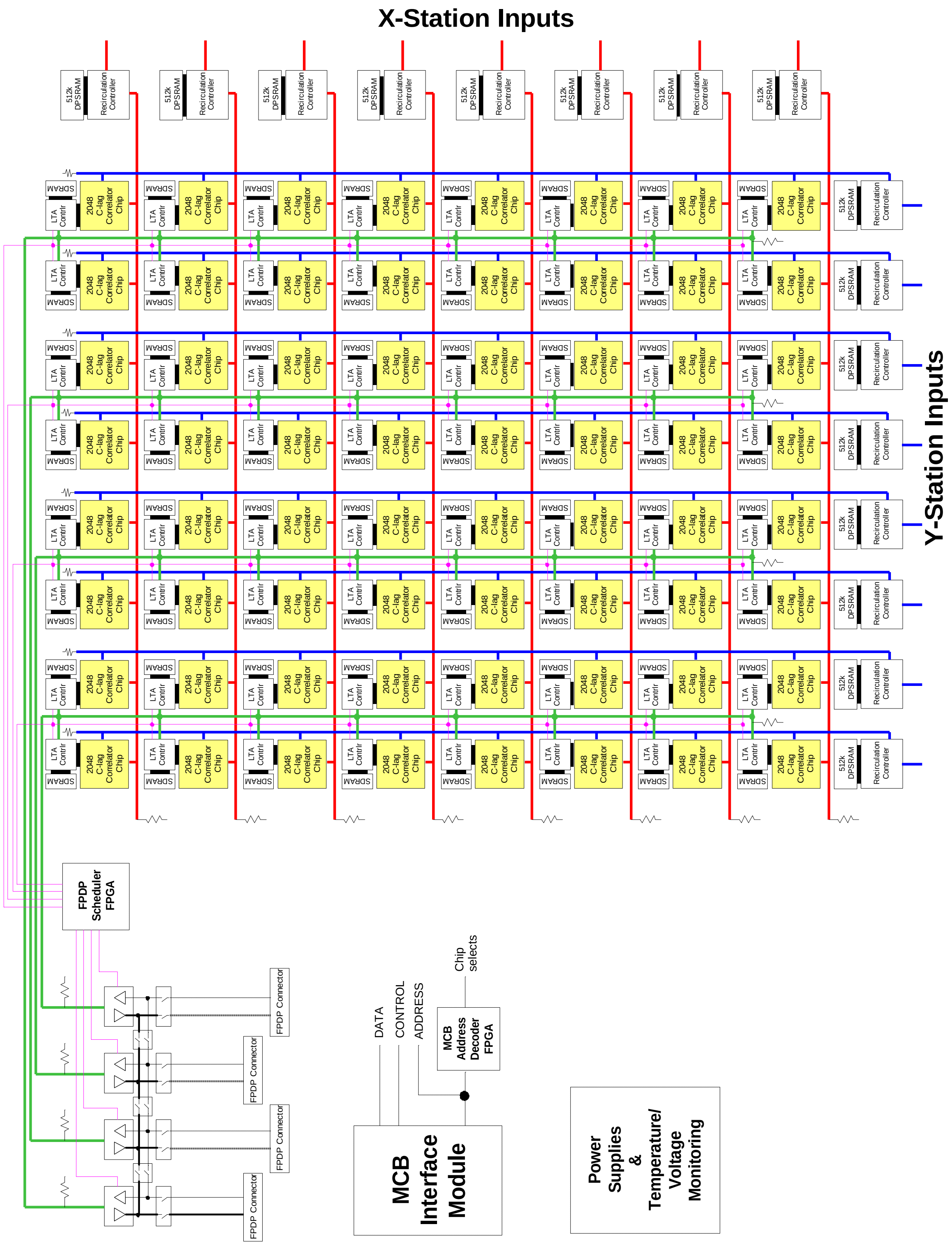


Figure 4-1 Baseline Board block diagram. The board consists of an 8x8 array of 64 custom 2048 lag correlator chips, fed by data from Recirculation Controller FPGAs. Each correlator chip is equipped with its own LTA Controller that reads out data, saves it in dedicated LTA SDRAM and then, when enabled by the FPDP Scheduler, becomes an FPDP transmit master to send the data out on up to 4 FPDP interfaces.

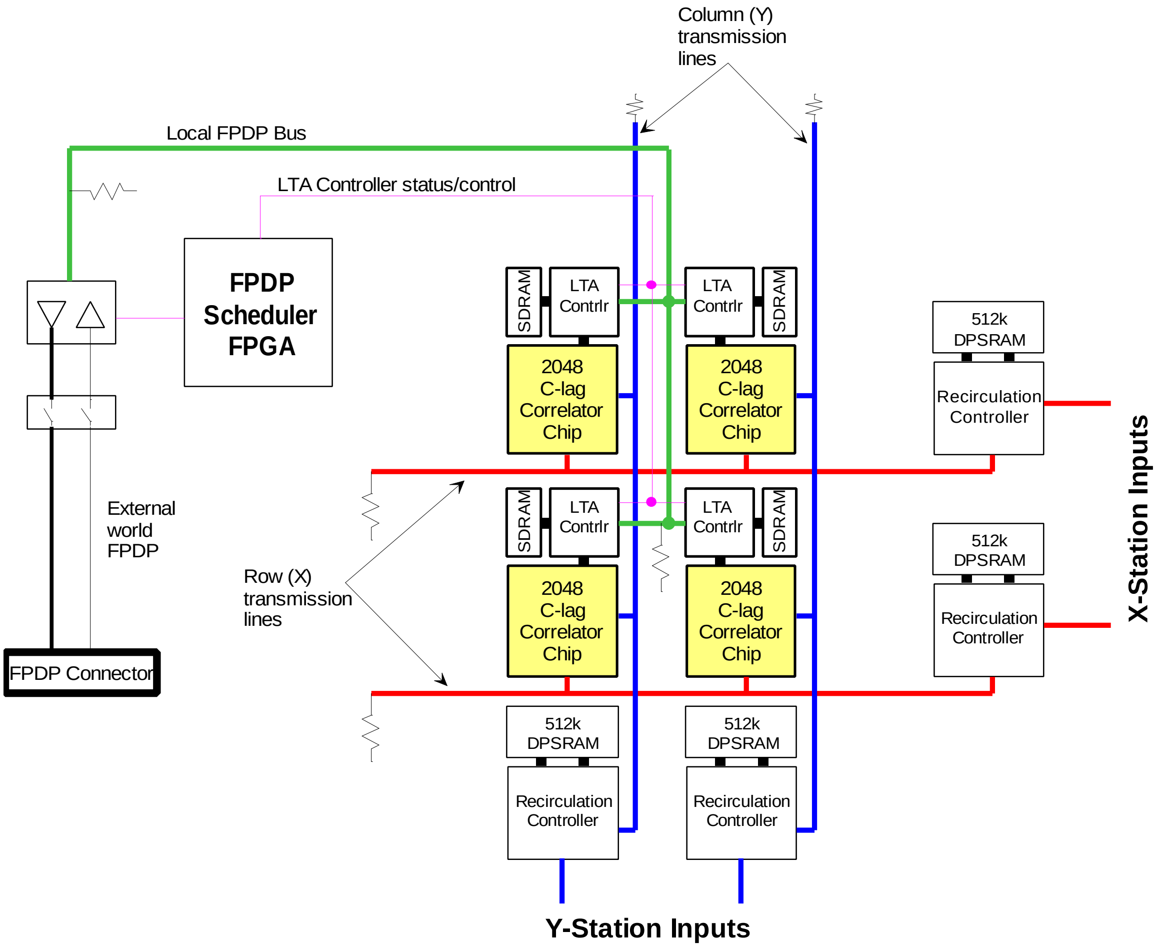


Figure 4-2 2x2 correlator chip 'slice' of the Baseline Board block diagram shown in **Figure 4-1**. Data from a particular set of Station Boards arrives at a Recirculation Controller where it is modified and formatted for transmission to a row or column of correlator chips. On command from the Recirculation Controller, the correlator chip dumps data and transmits it to a dedicated LTA Controller that saves it in local SDRAM. Once LTA data is ready it is transmitted, when enabled by the FPDP Scheduler FPGA, by the LTA Controller onto the local FPDP bus and finally onto the external world FPDP via FPDP bus drivers.

System organization is such that final synchronization of X and Y station data streams before correlation is performed on the correlator chip. Some final synchronization is needed because it is impossible to guarantee precision phase matching across the system at the clock rates being contemplated[9]. To achieve synchronization and allow for hot-swap capability, the signals entering the Recirculation Controller from the Station Boards have been designed so that they contain embedded information that allows for continuous synchronization and error checking while on-line. To enable final correlator chip synchronization, the Recirculation Controller generates test vectors (off-line) that allow the correlator chip to resolve all of its inputs to one time domain (synchronous clock) and allow the correlator chip to line up time epochs coming from X and Y Recirculation Controllers. While on-line, data formats are such that the correlator chip can monitor the

[9] However, X and Y station path lengths should try to be matched as well as possible since any skew will ultimately have to be absorbed in the correlator chip.

health of the synchronous connection without having to do a whole bunch of more sophisticated error checking. Thus, when a board is hot-swapped in the system the following steps occur:

1. Power-on the new board and boot-up on-board logic.
2. Tell the Recirculation Controllers to synchronize and align input signals[10].
3. Tell the Recirculation Controller to generate test vectors.
4. Tell each correlator chip to receive test vectors and synchronize signals.
5. Once synchronization is achieved, put the Recirculation Controller and correlator chip in on-line mode.
6. The Recirculation Controllers and correlator chips monitor inputs for transmission and synchronization errors.

This strategy ensures that only the board that is being replaced is affected—data going to other Baseline Boards does not need to be interrupted to get Recirculation Controllers and correlator chips synchronized.

The Baseline Board also contains temperature and voltage monitoring, dead-man thermal protection, and remote power-cycle capability as described in section 3.10 on the Station Board.

The following sections will describe each chip on the Baseline Board and protocols between chips in much more detail. It will be demonstrated that the architecture shown in Figure 4-1 is capable of meeting all of the performance requirements of the EVLA.

4.1 Recirculation Controller

The Recirculation Controller[11] is responsible for taking data, timing, and control information from a station (consisting of one sub-band from all 8 basebands) and formatting it for transmission to, and use by, a row or column of correlator chips. A black-box diagram of the Recirculation Controller is shown in Figure 4-3. Data and control information[12] enters into the right of the box and exits to the correlator chip row or column on the left. Two banks (devices) of DPSRAM (Dual-Port synchronous Static RAM) recirculation memory are shown, each one 256k x 18 bits. Two banks are used in ping-pong mode to be able to operate up to the 256 Ms/s rate (i.e. each RAM is capable of ~133 MHz). The total memory depth—which sets the data burst length and the correlator chip readout time—is 512k. At the maximum synthesized lag length of 512k

[10] It is thought that it is better to have the Recirculation Controller synchronize and align signals on command rather than have it do it continuously and autonomously.

[11] Calling it a Recirculation Controller is a bit of a misnomer. It does perform this function, but it performs many others as well.

[12] Each signal uses two differential transmission lines.

lags (256k spectral points) each burst uses ½ of the memory—requiring a correlator dump/readout period of 1 millisecond. The 18-bit width of the memory allows 4-bit data, phase, delay, and synchronization information for two sampled data streams to be fed through the memory. Additionally, it is possible to operate in modes where recirculation is performed on one or two basebands, but not on the other basebands—increasing the flexibility of the system.

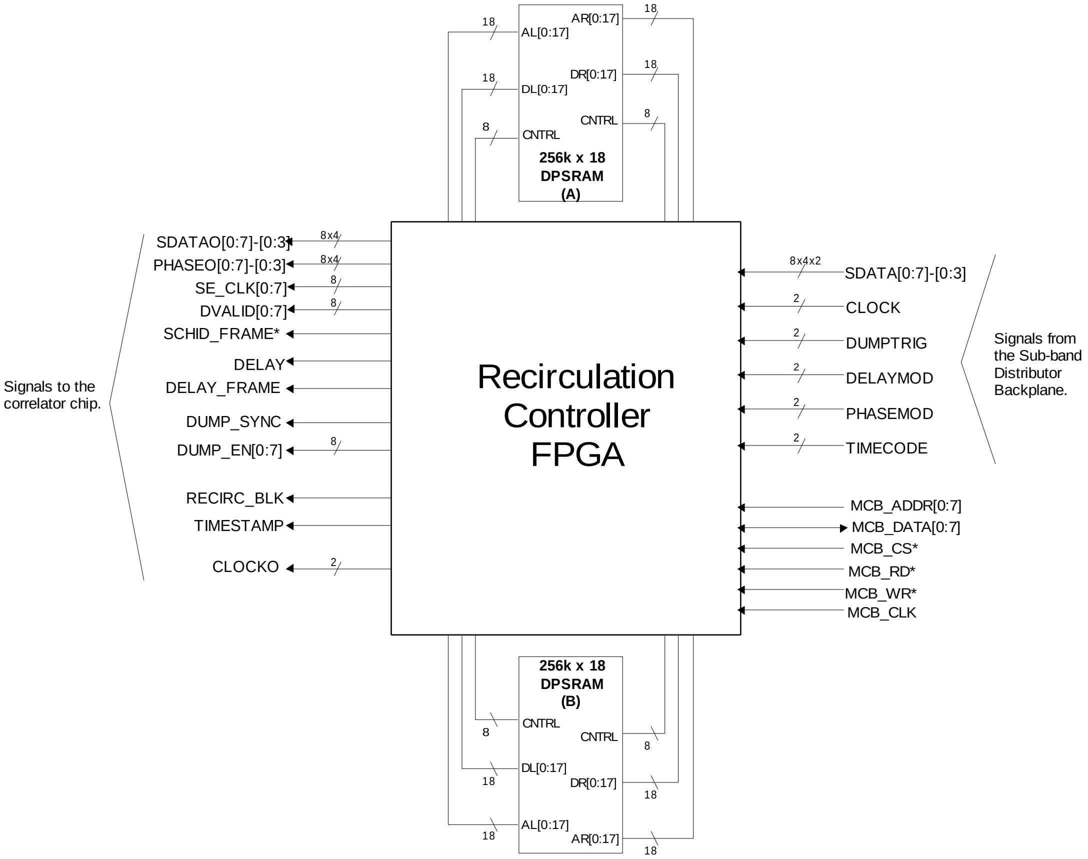


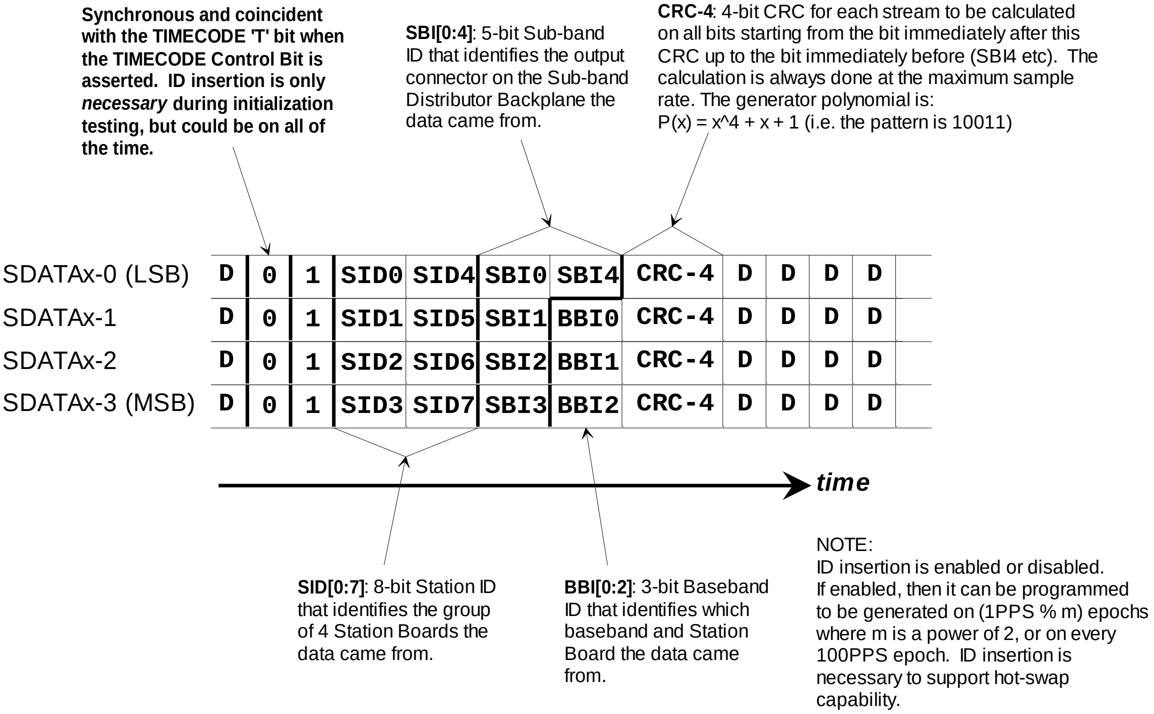
Figure 4-3 Black-box diagram of the Recirculation Controller FPGA. Data, timing, and synchronization information enter from Station Boards on the right. Data and information formatted for use by a row or column of correlator chips exits on the left. Two, 256k x 18 DPSRAM memories are used in ping-pong mode to provide an effective 512k x 18 recirculation memory that operates at a 256 MHz clock rate. An MCB bus interface is shown, but the actual address space and word width requires further definition. Input signals from the Station Boards (and CLOCKO to the correlator chips) are LVDS and all other signals are LVTTL.

4.1.1 Signal Descriptions

This section contains a description of each of the important labeled signals shown in Figure 4-3.

4.1.1.1 Inputs from Station Boards

SDATA[0:7][0:3] – 8 sampled data streams at 4 bits per sample. Each sampled data stream comes from the same sub-band output of each of the 8 station basebands. The format for SDATA is shown in Figure 4-4. CRC-4 error checking is proposed because the generation and detection circuitry is simple, it is ideally suited to serialized data, and is capable of detecting about 95% of all bit errors on a serial data stream. The generator polynomial is the same as the one chosen in some digital telephony carrier systems.



Synchronous and coincident with the TIMECODE 'T' bit when the TIMECODE Control Bit is asserted. ID insertion is only *necessary* during initialization testing, but could be on all of the time.

SBI[0:4]: 5-bit Sub-band ID that identifies the output connector on the Sub-band Distributor Backplane the data came from.

CRC-4: 4-bit CRC for each stream to be calculated on all bits starting from the bit immediately after this CRC up to the bit immediately before (SBI4 etc). The calculation is always done at the maximum sample rate. The generator polynomial is:
P(x) = x^4 + x + 1 (i.e. the pattern is 10011)

SDATAx-0 (LSB)	D	0	1	SID0	SID4	SBI0	SBI4	CRC-4	D	D	D	D
SDATAx-1	D	0	1	SID1	SID5	SBI1	BBI0	CRC-4	D	D	D	D
SDATAx-2	D	0	1	SID2	SID6	SBI2	BBI1	CRC-4	D	D	D	D
SDATAx-3 (MSB)	D	0	1	SID3	SID7	SBI3	BBI2	CRC-4	D	D	D	D

SID[0:7]: 8-bit Station ID that identifies the group of 4 Station Boards the data came from.

BBI[0:2]: 3-bit Baseband ID that identifies which baseband and Station Board the data came from.

NOTE:
ID insertion is enabled or disabled. If enabled, then it can be programmed to be generated on (1PPS % m) epochs where m is a power of 2, or on every 100PPS epoch. ID insertion is necessary to support hot-swap capability.

4-bit SDATA Format

Figure 4-4 Sampled data stream, SDATA, format. Each data stream contains embedded station ID[13] (SID[0:7]), sub-band ID (SBI[0:4]), and baseband ID (BBI[0:3]). Each *bit* stream also contains an embedded CRC-4 code that allows for continuous error checking—required during time-skew removal and to monitor the integrity of the data link from the source. When present, this embedded data will be recognized and flagged as data invalid by the Recirculation Controller when it is passed on to the correlator chip—effectively blanking it from being correlated. However, because the number of data valid counters in the correlator chip is restricted, this blanking can introduce small, unwanted systematic effects (incorrect data valid counts at some lags). To allow for no blanking correlation[14] (i.e. once synchronization and IDs are established), the 'C' bit (Control Bit) in TIMECODE determines whether the embedded data is present or not. If the 'C' bit is 1, then embedded data is present, if 0, then it is not.

CLOCK – Constant clock rate at 128 MHz (or 128 MHz). The phase of this clock relative to all input signals is arbitrary and this clock is ½ the actual clocking rate in the system.

[13] Doing this permits each Station Board to have its own station ID—a nice feature for low-bandwidth applications [5]. The baseband numbers would still have to be the normal 0...7 number across all 4 Station Boards to avoid ambiguities, though.

[14] Ignoring the short blanking after a correlator chip dump needed to properly save all accumulator data in buffer registers. This does not have the same effect as data valid blanking since all lags undergo the same blanking at the same time.

DUMPTRIG – This signal contains all of the information needed to direct data dumping from the correlator chip. Ultimately, each correlator chip can choose to dump its data based on DUMPTRIG signaling from the X or Y input. The format for DUMPTRIG is shown in Figure 4-5.

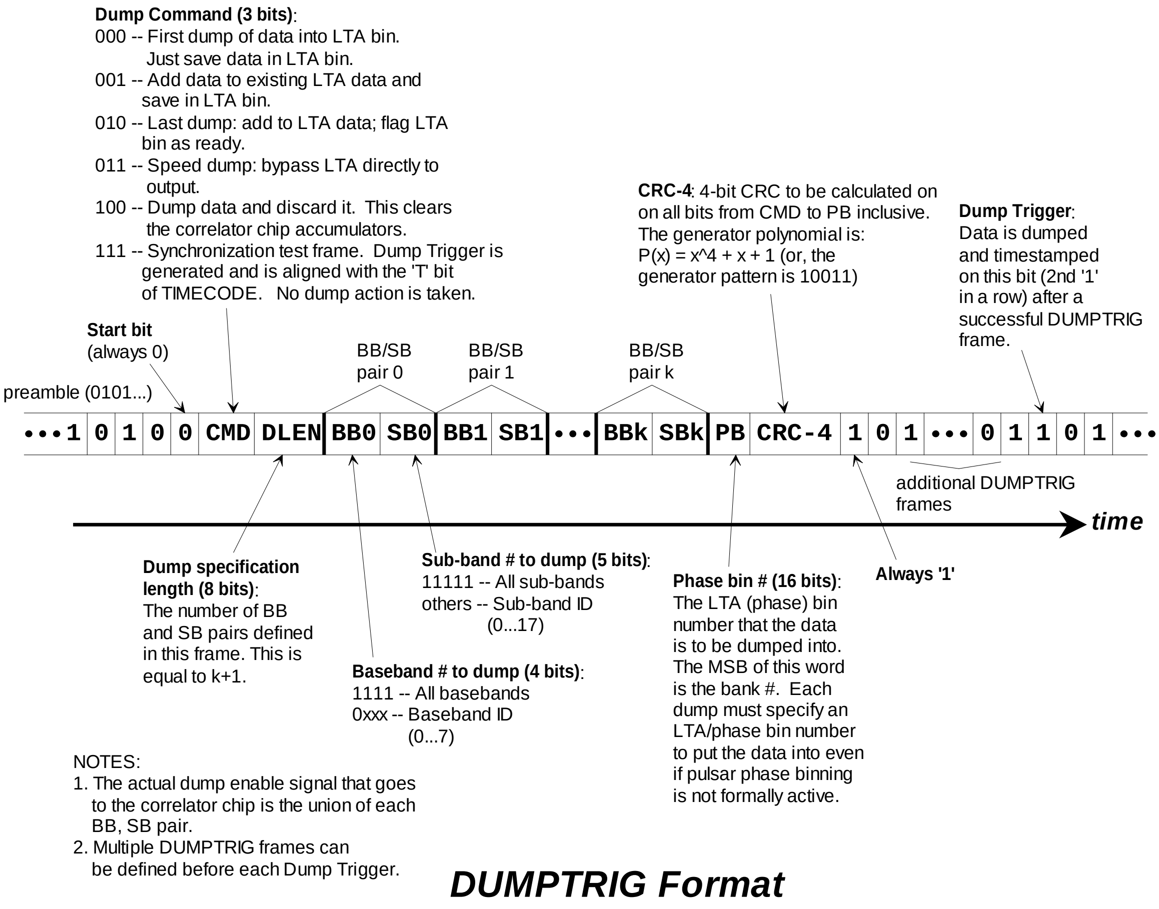
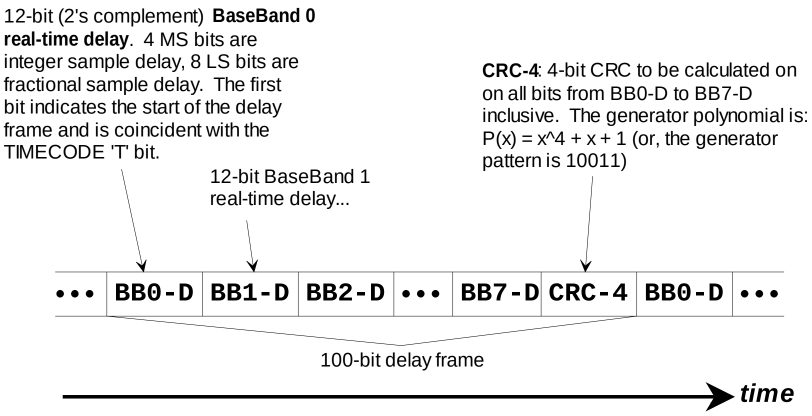

Dump Command (3 bits):
000 -- First dump of data into LTA bin. Just save data in LTA bin.
001 -- Add data to existing LTA data and save in LTA bin.
010 -- Last dump: add to LTA data; flag LTA bin as ready.
011 -- Speed dump: bypass LTA directly to output.
100 -- Dump data and discard it. This clears the correlator chip accumulators.
111 -- Synchronization test frame. Dump Trigger is generated and is aligned with the 'T' bit of TIMECODE. No dump action is taken.
CRC-4: 4-bit CRC to be calculated on on all bits from CMD to PB inclusive. The generator polynomial is: P(x) = x^4 + x + 1 (or, the generator pattern is 10011)
Dump Trigger: Data is dumped and timestamped on this bit (2nd '1' in a row) after a successful DUMPTRIG frame.
Start bit (always 0)
preamble (0101...)
BB/SB pair 0
BB/SB pair 1
BB/SB pair k
•••1 0 1 0 0 CMD DLEN BB0 SB0 BB1 SB1 ••• BBk SBk PB CRC-4 1 0 1 ••• 0 1 1 0 1 •••
additional DUMPTRIG frames
time
Dump specification length (8 bits): The number of BB and SB pairs defined in this frame. This is equal to k+1.
Sub-band # to dump (5 bits): 11111 -- All sub-bands others -- Sub-band ID (0...17)
Baseband # to dump (4 bits): 1111 -- All basebands 0xxx -- Baseband ID (0...7)
Phase bin # (16 bits): The LTA (phase) bin number that the data is to be dumped into. The MSB of this word is the bank #. Each dump must specify an LTA/phase bin number to put the data into even if pulsar phase binning is not formally active.
Always '1'
NOTES:
1. The actual dump enable signal that goes to the correlator chip is the union of each BB, SB pair.
2. Multiple DUMPTRIG frames can be defined before each Dump Trigger.
DUMPTRIG Format

Figure 4-5 DUMPTRIG format. DUMPTRIG consists of one or more frames that define one or more dumps that are to occur, followed by the Dump Trigger that causes the actual dump to occur. This format provides information that the Recirculation Controller needs to trigger dumps to the correlator chip—and ultimately command the LTA controller as well. The CMD '111' (Synchronization test frame) allows synchronization checks of DUMPTRIG to TIMECODE to occur even when dumping is synchronized to a pulsar rather than system timing by inserting dummy frames. This is important to guarantee recirculation synchronization between 'X' and 'Y' Recirculation Controllers.

DELAYMOD – Serialized real-time delay models for each baseband. This delay information is used for real-time digital sub-sample delay compensation (in WIDAR delay and VLBI delay modes). The DELAYMOD format is shown in Figure 4-6.

PHASEMOD – Serialized linear phase models for the entire station—only some of which are applicable to this particular Recirculation Controller. The Recirculation Controller generates the real-time phase data that is used by the correlator chips. The PHASEMOD format is shown in Figure 4-7.



12-bit (2's complement) **BaseBand 0 real-time delay**. 4 MS bits are integer sample delay, 8 LS bits are fractional sample delay. The first bit indicates the start of the delay frame and is coincident with the TIMECODE 'T' bit.

12-bit BaseBand 1 real-time delay...

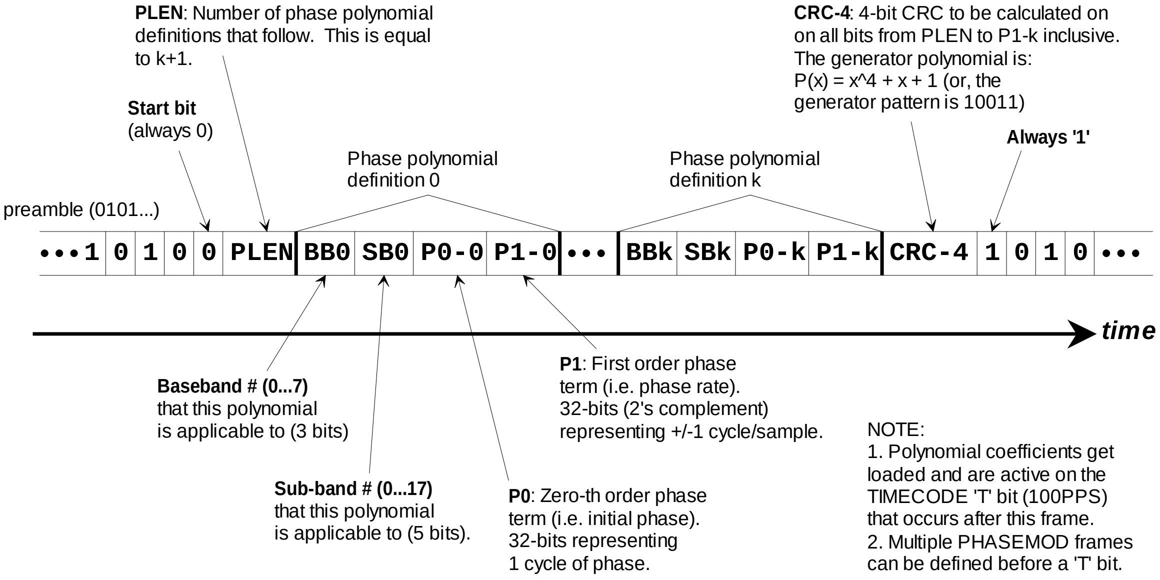
CRC-4: 4-bit CRC to be calculated on on all bits from BB0-D to BB7-D inclusive. The generator polynomial is: P(x) = x^4 + x + 1 (or, the generator pattern is 10011)

••• | BB0-D | BB1-D | BB2-D | ••• | BB7-D | CRC-4 | BB0-D | •••

100-bit delay frame

DELAYMOD Format

Figure 4-6 DELAYMOD format. A 100-bit delay frame consists of 12 bits of delay information for each baseband followed by a CRC-4 code calculated on all of the bits in the frame. Of the 12 bits of delay information, 8 bits contains the fractional sample delay (2’s complement in the range of ±0.5 samples of delay), and 4 bits contains the integer sample delay (2’s complement in the range of ±8 samples of delay). The fractional sample delay is used in sub-sample WIDAR delay compensation [3], and the fractional as well as the integer delay is used in VLBI delay calculations on the correlator chip. The 12 bits of delay information is transmitted LSB first.



PLEN: Number of phase polynomial definitions that follow. This is equal to k+1.

Start bit (always 0)

preamble (0101...)

Phase polynomial definition 0

Phase polynomial definition k

CRC-4: 4-bit CRC to be calculated on on all bits from PLEN to P1-k inclusive. The generator polynomial is: P(x) = x^4 + x + 1 (or, the generator pattern is 10011)

Always '1'

•••1 | 0 | 1 | 0 | 0 | PLEN | BB0 | SB0 | P0-0 | P1-0 | ••• | BBk | SBk | P0-k | P1-k | CRC-4 | 1 | 0 | 1 | 0 | •••

Baseband # (0...7) that this polynomial is applicable to (3 bits)

Sub-band # (0...17) that this polynomial is applicable to (5 bits).

P0: Zero-th order phase term (i.e. initial phase). 32-bits representing 1 cycle of phase.

P1: First order phase term (i.e. phase rate). 32-bits (2's complement) representing +/-1 cycle/sample.

NOTE:
1. Polynomial coefficients get loaded and are active on the TIMECODE 'T' bit (100PPS) that occurs after this frame.
2. Multiple PHASEMOD frames can be defined before a 'T' bit.

PHASEMOD Format

Figure 4-7 PHASEMOD format. A PHASEMOD frame can contain one or more linear phase models that apply to one or more SDATA streams, defined by a baseband number and sub-band number pair (BB/SB). The models get loaded on the next occurrence of the TIMECODE ‘T’ bit (1PPS or 100PPS). Using this method, models can get updated as frequently as every 10 milliseconds, or as infrequently as desired. Note that since one PHASEMOD stream is generated for the entire station, only some of the models will be applicable to a particular Recirculation Controller’s SDATA inputs.

TIMECODE – Signal that contains precision real-time time information. All events in the correlator occur logically[15] synchronized to this signal. The TIMECODE format is shown in Figure 4-8.

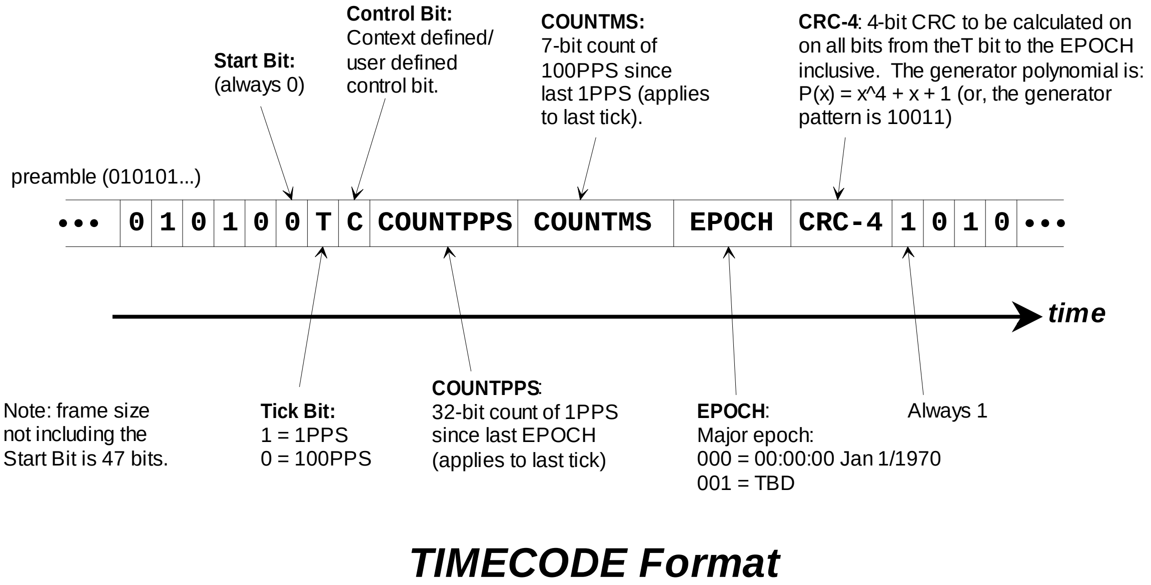


Figure 4-8 TIMECODE format. The Control Bit tells receivers of SDATA whether or not to expect embedded ID and CRC-4 data on this tick. If the Control Bit is 1, then the embedded data is present, otherwise it is not. COUNTMS is not used by the Recirculation Controller to form the TIMESTAMP that is transmitted to the correlator chip, but it is used on the Station Board.

4.1.1.2 Outputs to Correlator Chips

SDATAO[0:7][0:3] – Sampled data output streams. In some cases, these are simply a reflection of input SDATA. If recirculation is active, applicable streams contain data bursts. A switch in the Recirculation Controller allows any of these outputs to be connected to any SDATA input.

PHASEO[0:7][0:3] – For each SDATAO sampled data stream, there is a 4-bit phase stream that provides the correlator chip with phase for each sample. These real-time phase streams are produced by the Recirculation Controller from the PHASEMOD models.

SE_CLK[0:7] – Shift Enable Clock for each sampled data stream. When asserted high, each line indicates that the associated SDATAO sample can be shifted through the lag-chain shift registers. Each of these signals could be at a different rate, allowing each SDATAO stream to be at a different sample rate. These are always asserted high coincident with the TIMECODE 'T' bit (i.e. 100PPS and 1PPS epochs). If an SDATAO stream is operating at the maximum sample rate, then the associated SE_CLK line is always asserted.

[15] Meaning that events are synchronized to the same epoch in TIMECODE even though at a particular instant in time there may be skew.

DVALID[0:7] – There is one of these DVALID lines for each SDATAO sample stream. These are asserted active high to indicate that the associated SDATAO sample is valid. Note that the SDATA input to the Recirculation Controller normally uses one state (-8: 1000b) to indicate that data is invalid. If enabled, the Recirculation Controller intercepts this state and negates the associated DVALID line. If not enabled, the –8 state passes through without affecting the DVALID line—needed to support 7-bit correlation [1].

SCHID_FRAME* - This signal is asserted for one CLOCKO cycle to indicate the beginning of information embedded in the SDATAO data streams. This signal is only asserted when the TIMECODE Control Bit is asserted.

DELAY – This signal contains real-time delay information for each SDATAO stream. This signal is normally used by the correlator chip in VLBI modes where the normal sub-sample delay tracking mechanism is not used because the original real-time sample rate of the sampled data stream is only twice the bandwidth. X and Y DELAY information is used on the correlator chip to generate the baseline-based fine/vernier delay and phase modifier [2].

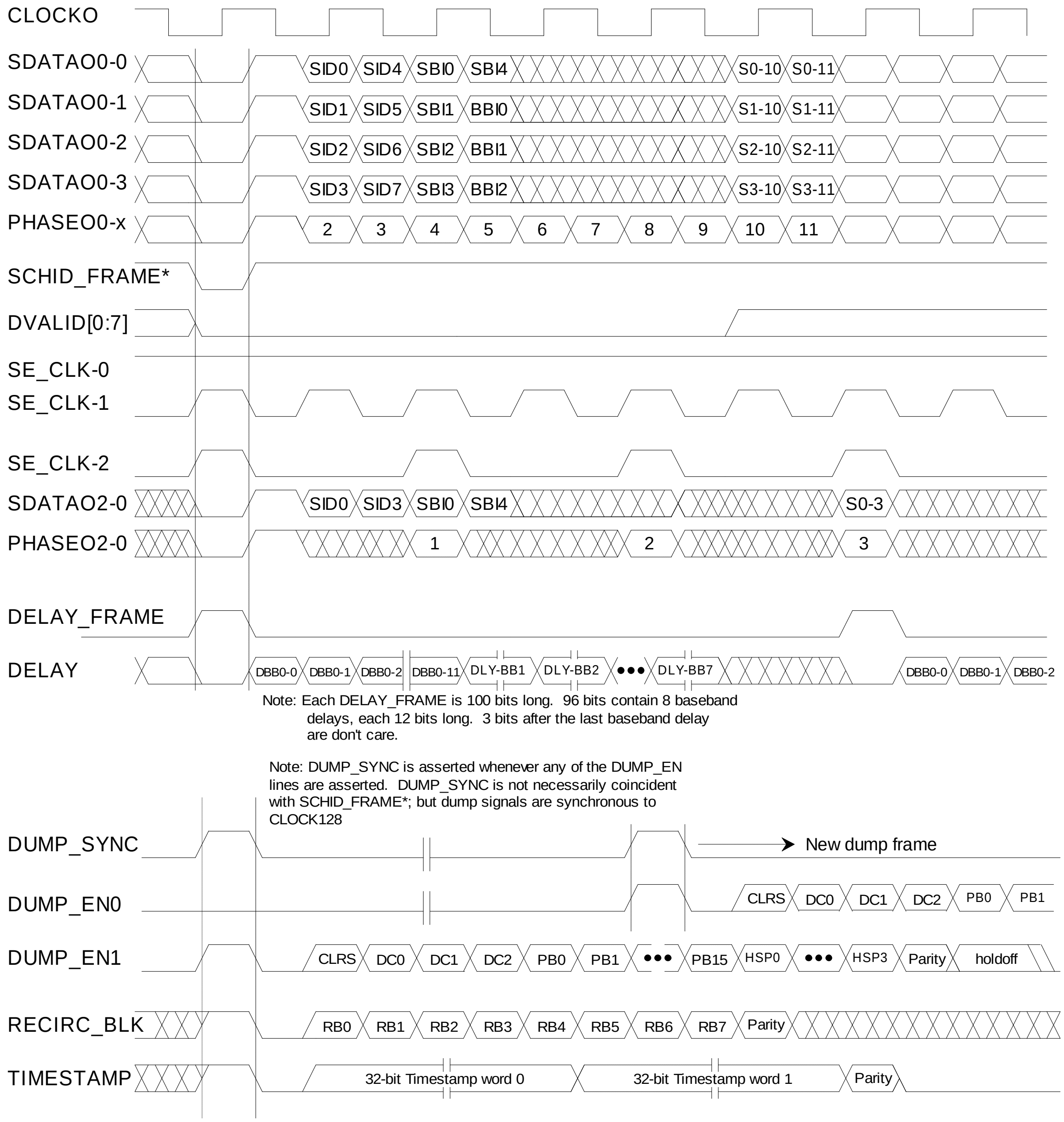
DELAY_FRAME – A framing/synchronization signal for DELAY. Each DELAY frame consists of 100 bits.

DUMP_SYNC – This signal is asserted when at least one of the DUMP_EN lines is asserted and indicates the beginning of information on the RECIRC_BLK and TIMESTAMP lines. Correlator chip dumping can occur logically synchronized to astronomical events and so this does not, in general, occur when SCHID_FRAME* occurs.

DUMP_EN[0:7] – Each of these signals indicates that a set of correlator chip lags connected to the associated SDATAO data stream should be dumped. Whether dumping actually occurs in the correlator chip depends on whether the correlator chip has been programmed to dump on an X or Y station command. These signals also contain additional information such as the dump command, the LTA/phase bin the dump is to go to, the harmonic suppression phase that needs to be removed prior to accumulation in the LTA, and the correlation “holdoff” indication.

RECIRC_BLK – This line contains the recirculation block counter for the asserted DUMP_EN lines. The correlator chip simply captures this counter and includes it in the associated output data frame for downstream processing. Details of how recirculation is implemented are contained in following sections.

TIMESTAMP – This line contains 64-bit timestamp information for the dump. This is a derived copy of TIMECODE when the most recent DUMP_SYNC occurred. TIMESTAMP word 0 (refer to Figure 4-9) is a copy of the TIMECODE COUNTPPS (i.e. number of seconds since the epoch). Word 1, bits 0...28, is a count of the number of clocks (at the maximum rate of 256 MHz) since the last 1PPS. Word 1, bits 29...31, is the epoch from TIMECODE. This format ensures that the dump is accurately and precisely timestamped when dumping is synchronized to astronomical events (pulsars).



Note: Each DELAY_FRAME is 100 bits long. 96 bits contain 8 baseband delays, each 12 bits long. 3 bits after the last baseband delay are don't care.

Note: DUMP_SYNC is asserted whenever any of the DUMP_EN lines are asserted. DUMP_SYNC is not necessarily coincident with SCHID_FRAME*; but dump signals are synchronous to CLOCK128

Figure 4-9 Functional timing diagram of signals transmitted by the Recirculation Controller to a row or column of correlator chips. Each signal contains a distinct 'eye' that allows simple real-time monitoring of synchronization by the correlator chip even after a formal synchronization step has been completed. If there is an error in the eye, the correlator chip can detect it and report the error via an associated data frame to the LTA Controller. SCHID_FRAME* and associated embedded information is only present when the Control Bit (i.e. the 'C' bit) of TIMECODE is present.

CLOCKO – 128 MHz clock synchronous to all correlator chip destined data. Data changes on both the rising and falling edges of this clock. The Recirculation Controller guarantees that all signals are aligned and synchronized to this clock.

Figure 4-9 defines representative functional timing of the data transmitted from the Recirculation Controller to a row or column of correlator chips. Unless otherwise indicated, all serial embedded data is transmitted LSB first.

The DUMP_ENx control bits are defined as follows:

- **CLRS** – If set (1), then any lag shift registers that use data from the associated sampled data stream are cleared. This must be done after every recirculation burst to avoid incorrect correlation of non-contiguous time chunks of data (although, strictly speaking, the correlation holdoff function eliminates the need for this bit).
- **DC[0:3]** – This is identical to the dump command 'CMD' defined in the DUMPTRIG format of Figure 4-5 except that the synchronization test frame (CMD='111') is never generated. This command is passed on by the correlator chip to the downstream LTA Controller.
- **PB[0:15]** – This is the 16-bit phase bin number encoded in the DUMPTRIG command. Each DUMP_ENx line could have its own unique phase bin number—possibly required when mixing recirculation and non-recirculation modes and different bandwidth sub-bands. This is passed on by the correlator chip to the LTA Controller. Although the LTA controller only provides for two banks of 1000 phase bins, if the dump command is a speed dump (011), then all of these phase bins could be utilized by a downstream computer.
- **HSP[0:3]** – Harmonic suppression phase. This is a phase offset for this dump that the LTA Controller (or downstream computing engine) must remove before integrating with other data. This is to help attenuate narrowband signal harmonics generated by the quantizer [4]. This phase is simply passed on by the correlator chip to the LTA Controller.

In addition, the DUMP_ENx signal functions as a "correlation holdoff" by staying high until correlation should start. This function is necessary to eliminate edge effects when recirculation is active by providing a mechanism to allow the lag shift register to fill up with data before correlation begins.

4.1.2 Simplified Block Diagram

A simplified block diagram of the Recirculation Controller FPGA is shown in Figure 4-10.

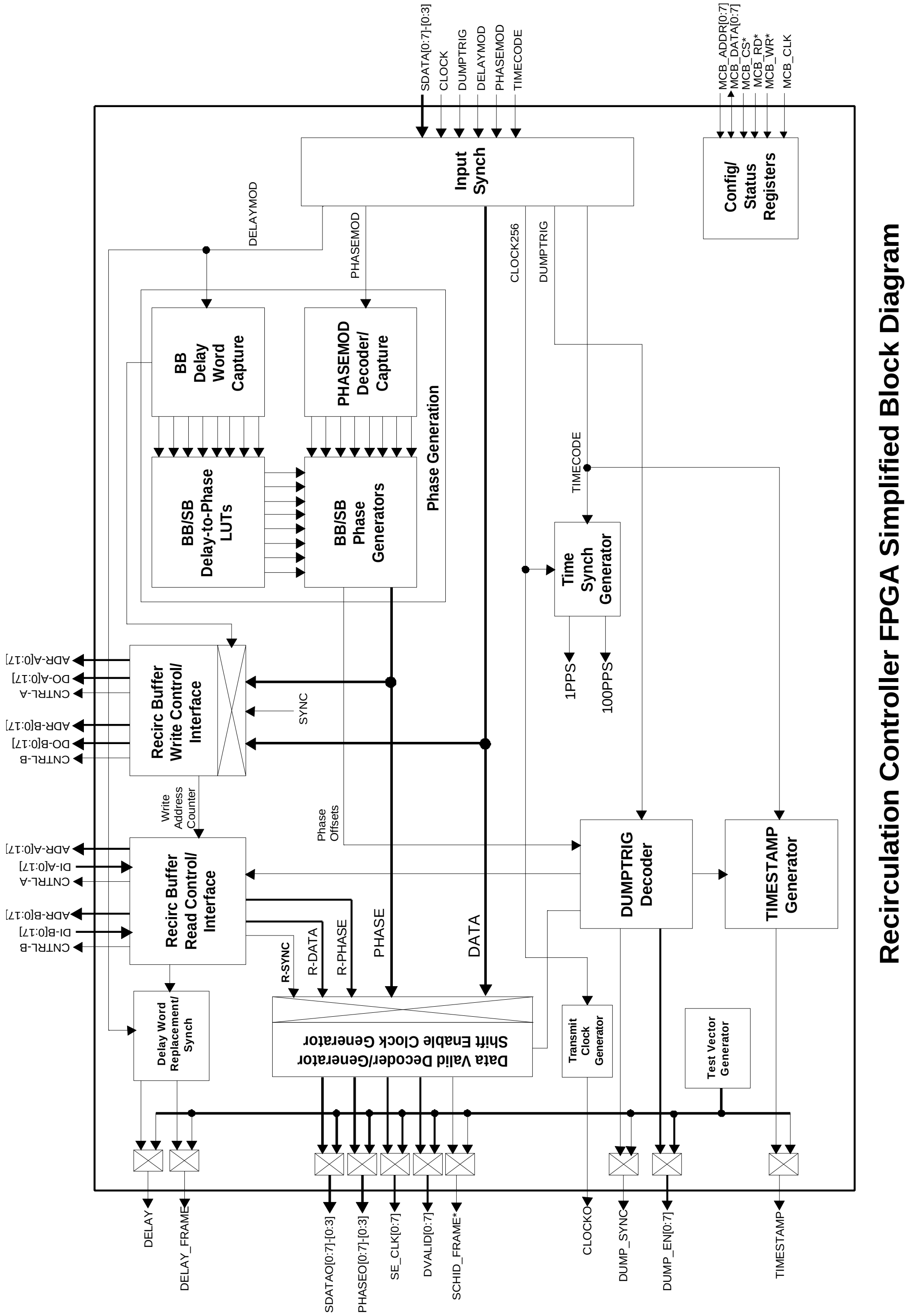


Figure 4-10 Simplified block diagram of the Recirculation Controller. The device generates real-time 4-bit phase that is transmitted to the correlator chip with the data. Phase is generated and carried with the data through the recirculation memory so there is no need for ‘rewind control’ of the phase generators. A switch allows any output sampled data stream to be connected to any input or recirculation data stream. A Test Vector Generator is used to facilitate correlator chip timing and synchronization in an off-line procedure.

The Recirculation Controller's main functions are phase generation, recirculation control, reformatting data for the correlator chips, and facilitating final signal synchronization before correlation using the Test Vector Generator. The Test Vector Generator can be switched on and replaces all data going to the row or column of correlator chips. This allows each correlator chip to synchronize its X and Y input data to the same time domain (clock), line up timing epochs (100PPS and 1PPS), and test the integrity of the input signals. Once alignment and signal integrity is checked, the Recirculation Controller generates normal data as outlined in Figure 4-9. The actual test vector sequence requires further definition but it will probably be a LFSR (linear feedback shift register) pseudo-random code synchronized to an embedded 100PPS signal transmitted on SCHID_FRAME*[16].

4.1.3 Input Timing and Synchronization

The data, timing, and control signals arriving at the Recirculation Controller will have traveled over about 13 m of cable, several inches of backplane, and through several connectors on its way from the Station Boards. There is one repeater (on the Station Data Fanout Board) in this path that resynchronizes and re-times the data, but a conservative estimate is that the data and clock will arrive—at the Recirculation Controller and the Station Data Fanout Board—skewed in time. With resynchronization logic and the codes and error checking embedded in the signals, it is possible to remove this skew and present the rest of the chip's internal logic with deskewed data synchronized to the internal 256 MHz clock. A schematic of the logic required for each input is shown in Figure 4-11 (ref: Xilinx application note XAPP225 v1.0, Sept. 2000).

The circuitry in Figure 4-11 requires a 256 MHz clock and a 256 MHz clock with a 90^o phase-shift. Unfortunately, the Virtex-E[17] DLL (Delay Lock Loop) cannot generate a 256 MHz 90^o phase-shift clock from an input clock of 128 MHz because of DLL frequency limitations. A workaround to this problem is shown in Figure 4-12. An external delay line of 1 nsec creates a 45^o phase-shift at 128 MHz that is then fed into an additional clock input to the device. Two separate DLLs then double the clock—yielding the desired clock phases and a 128 MHz clock to be used for the ping-pong recirculation DPSRAM.

[16] SCHID_FRAME* cannot simply be a tick asserted every 100PPS because the correlator chip will not know for sure if this tick is being properly timed and received on its own.

[17] The newly available Virtex-II devices can handle 256 MHz clock generation without an external delay so this may be a mute point.

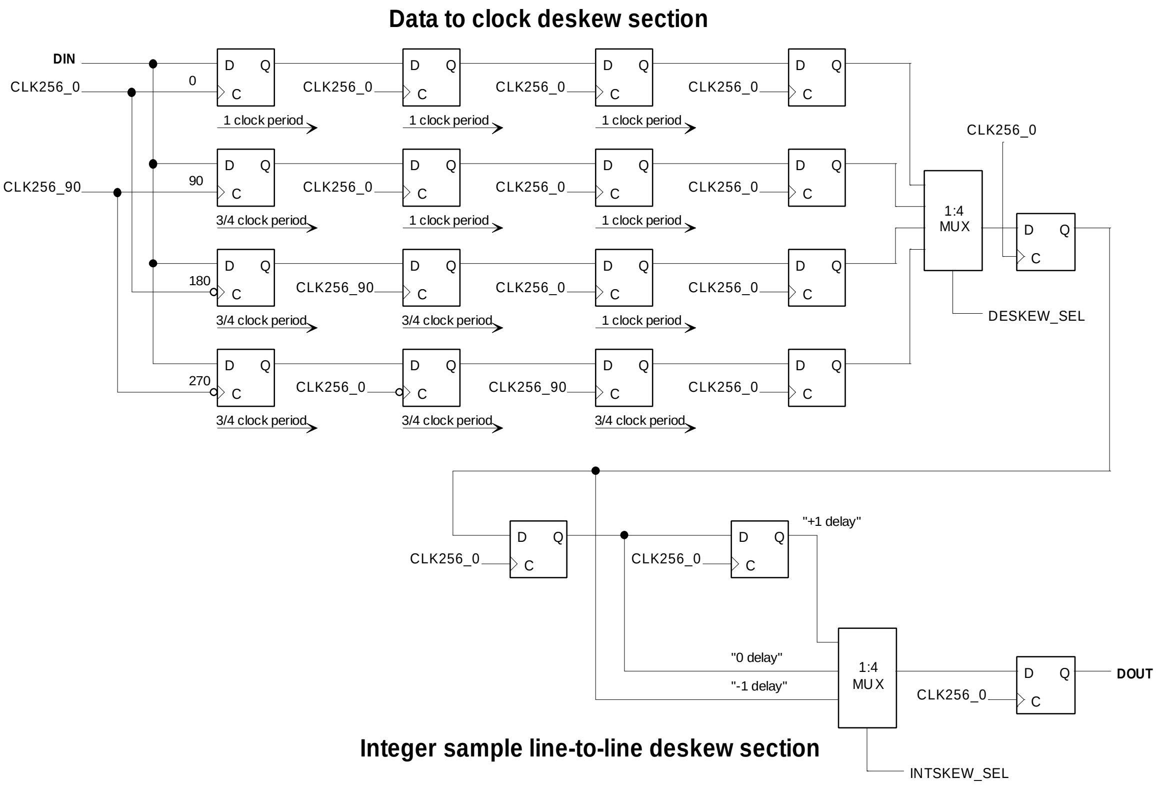


Figure 4-11 Schematic diagram of circuitry needed to ensure that input data (DIN) is properly sampled and deskewed by the time it reaches the output (DOUT). The input data is sampled on 4 phases of a 256 MHz clock and after several stages, it is all sampled on the same clock edge. The path with the smallest error rate is selected with ‘DESKEW_SEL’. Integer sample signal-to-signal skew is then removed by selecting the appropriately delayed signal with ‘INTSKEW_SEL’. Additional logic is required for embedded signal detection and control of DESKEW_SEL and INTSKEW_SEL.

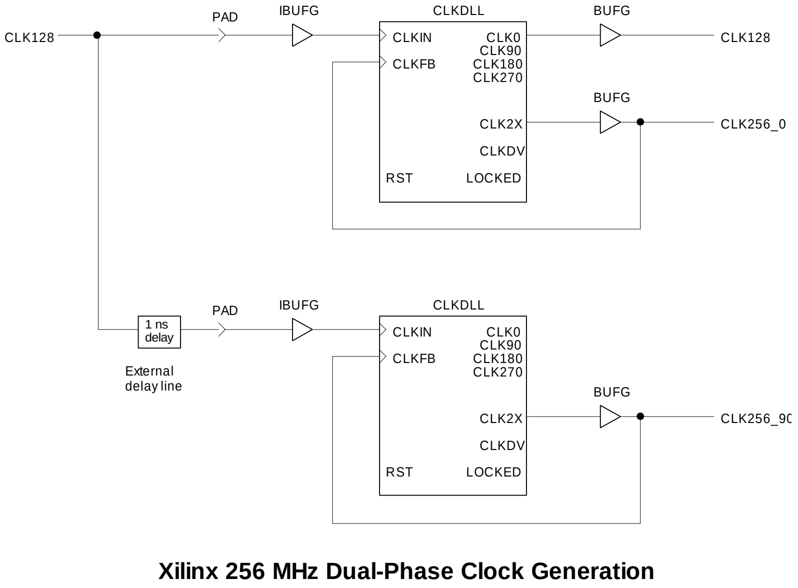


Figure 4-12 Xilinx Virtex-E DLL arrangement to produce desired internal 256 MHz clock phases. Two DLLs are used along with a 1 nsec external delay line. A 128 MHz clock is also generated that can be used for the recirculation memory and for transmission to the row or column of correlator chips.

4.1.4 Recirculation

Recirculation is used to obtain a larger number of spectral channels than would otherwise be available when the sample rate is less than the maximum sample rate the correlator chip can run at. This is achieved by bursting time-contiguous chunks of data—that was originally sampled at a lower rate—through the correlator chip at the high sample rate. This bursting requires a circular memory buffer that is sufficiently large enough to attenuate edge effects at the start of each burst, to ensure that the correlator chip readout period is not extreme[18], and to provide enough memory for the desired size of the synthesized lag chain. Recirculation processing requires several conceptually simple steps:

1. Write data[19] into a memory buffer at the rate it is actually sampled at. (To support VLBI-mode operation, the serialized delay stream and a synchronization stream is also written. This synchronization stream contains delay frame information as well as information required to identify when the data contains embedded IDs.)

2. Choose 'zero-delay' read data pointers for X and Y[20] at the same logical instant in time so that the samples they point to have zero relative delay.

3. Offset the X and Y read pointers so they account for the delay in the lag chain that occurs before the chunk of the lag chain that this particular burst is synthesizing.

4. Burst the data through the correlator chip. Before the burst starts, the lag chain shift registers must be cleared of data so that correlation with previous burst data does not occur.

5. Readout the correlator chip data and save in an LTA bin identified for that chunk of the lag chain.

To perform the above steps within the context of the correlator design (where X and Y Recirculation Controllers are operating essentially independently), it is useful to develop an algorithm from a very simple example. Figure 4-13 shows one such example with an N=32-lag correlator split into 4-lag, 'lag blocks'. Each lag block is to be synthesized using recirculation. The 'zero relative delay point' is where the X and Y data enters the lag chain.

[18] i.e. because the correlator chip must be read out after each burst.

[19] Here 'data' refers to sampled data and phase.

[20] Note that X and Y Recirculation Controllers are operating independently and so some logical synchronization method must be established.

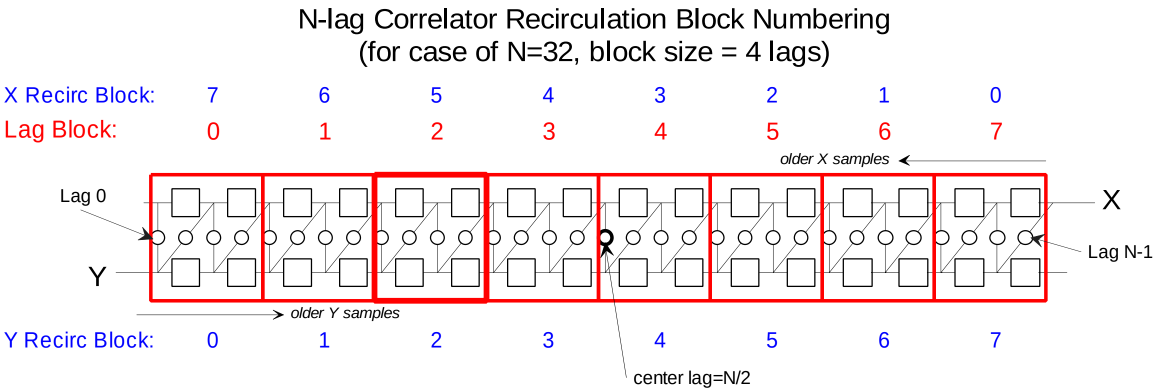

N-lag Correlator Recirculation Block Numbering
(for case of N=32, block size = 4 lags)
X Recirc Block: 7 6 5 4 3 2 1 0
Lag Block: 0 1 2 3 4 5 6 7
older X samples
Lag 0
X
Y
Lag N-1
older Y samples
Y Recirc Block: 0 1 2 3 4 5 6 7
center lag=N/2

Figure 4-13 Simple 32-lag example with a lag block size of 4 lags. Here, we have a 4-lag correlator chip and we want to use recirculation to synthesize 32 lags. The 'Lag Block' indicates the chunk of the lag chain that we want to synthesize on a given burst. The 'Y Recirc Block' indicates the number of blocks of delay (equal to ½ the block lag size) that must be inserted in the Y-station data path for a particular Lag Block burst. The 'X Recirc Block' indicates the number of blocks of delay that must be inserted in the X-station data path for the same Lag Block burst. For example, for Lag Block=2, the Y Recirc Block is 2 (delay=4), and the X Recirc Block is 5 (delay=10). Data further down a shift register is older in time indicating that to 'insert delay' means to choose older samples in the recirculation memory relative to the chosen zero relative delay point.

A circular memory buffer diagram showing the position of the read and write pointers at the beginning and end of the burst for Lag Block number 2 of Figure 4-13 is shown in Figure 4-14. Here, the X and Y 'zero-delay read pointers' are chosen to be ½ of a buffer away from the X and Y current (start) write pointers. Note that the actual X and Y write pointers are arbitrary relative to each other. In the actual implementation this means that data is simply just written into the X and Y buffers without the need to synchronize the write pointers. The choice of zero-delay read pointers is somewhat arbitrary, but allows for a synthesized delay up to ½ the buffer size and a burst of data of at least ½ a buffer size. It could be adjusted to allow for more synthesized lags (requiring a shorter correlator chip readout period), or fewer synthesized lags (requiring a longer correlator chip readout period). The start read pointers are then offset in the direction of older samples based on the particular lag block being synthesized. In this case, we're doing lag block 2, requiring a Y delay of 4 samples and an X delay of 10 samples (see Figure 4-13). Finally, the burst occurs and the next X and Y zero-delay read pointers have advanced by the same amount as the write pointers have advanced during the burst. In this way, provided the synthesized delay is not too large, the read and write pointers never crash into each other.

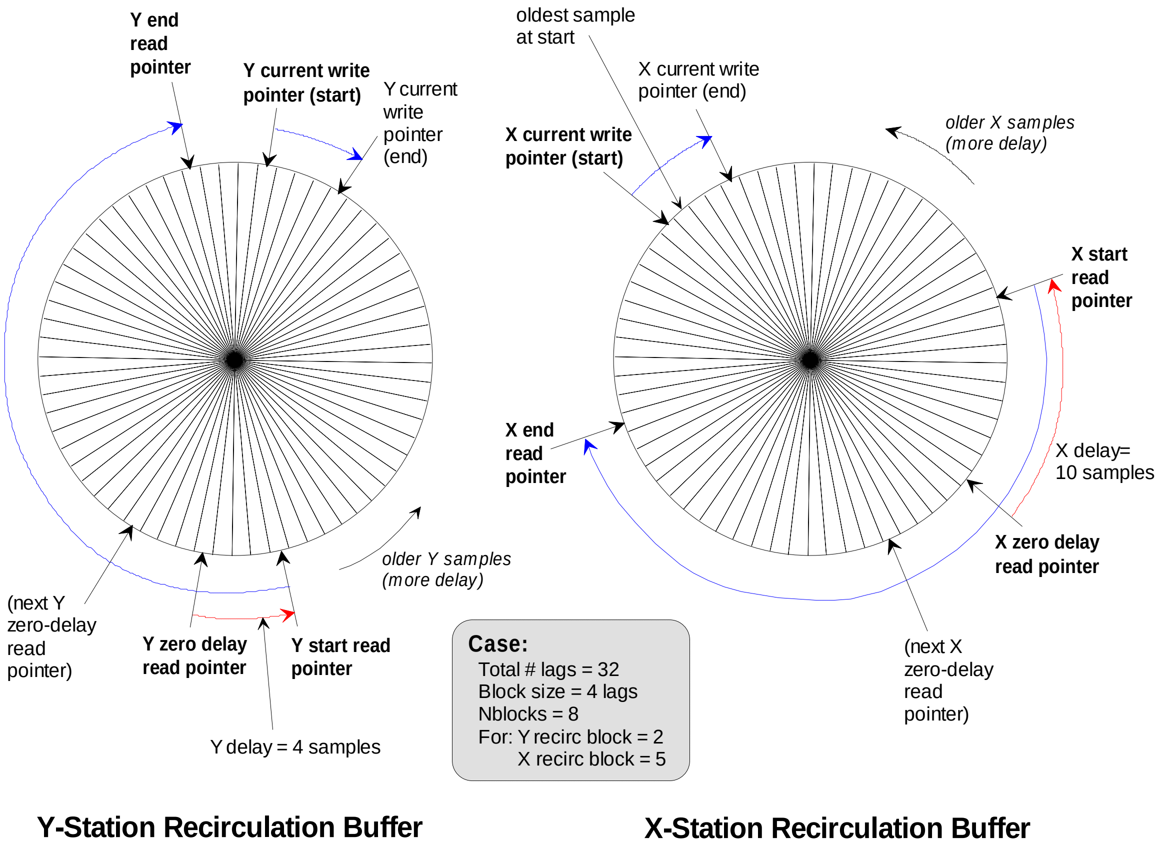

Y end read pointer
Y current write pointer (start)
Y current write pointer (end)
older Y samples (more delay)
(next Y zero-delay read pointer)
Y zero delay read pointer
Y start read pointer
Y delay = 4 samples
oldest sample at start
X current write pointer (end)
X current write pointer (start)
older X samples (more delay)
X start read pointer
X end read pointer
X delay= 10 samples
X zero delay read pointer
(next X zero-delay read pointer)
Case:
Total # lags = 32
Block size = 4 lags
Nblocks = 8
For: Y recirc block = 2
X recirc block = 5
Y-Station Recirculation Buffer
X-Station Recirculation Buffer

Figure 4-14 Example X and Y circular buffers for the simple recirculation example of Figure 4-13. The absolute X and Y write pointers are not important as long as the zero-delay read pointers (which are some function of the write pointers) point to samples that have a zero relative delay at some logical instant in time. The X and Y zero-delay read pointers are then offset in the direction of older samples (more delay) to get the final start read pointers. Once the burst is complete, the X and Y write pointers have advanced to new locations.

It is important that the memory buffer actually contain valid data before the read bursting occurs. This means that read bursting must wait awhile for the buffer to fill with data. Practically speaking, in a continuously operating system this is not problematic since it will take very little real time for the buffer to fill. This makes it very simple: the buffer is continuously written to with no need for X and Y write pointer synchronization, and read bursts occur when we want the correlator to actually produce data.

4.1.4.1 Timestamps

In Figure 4-14, an algorithm is illustrated in which, on average, the read pointer advances at the same rate as the write pointer. This is a robust, simple algorithm that ensures that the read and write pointers don’t crash into each other. However, it is also evident that each burst of data going through the correlator chip occurs on a different time-contiguous chunk of data. This will have the effect of introducing a time skew in the lag data—resulting in a smearing of the cross-power spectrum over a duration of time that is larger than the actual integration time. A circular buffer diagram illustrating this effect for the Y-station is shown in Figure 4-15. The X-station effect is similar except for the different synthesized delay offsets. Also shown in the figure is the timestamp of the last burst dump record that gets sent to the LTA controller. This timestamp must be eventually modified to yield the correct timestamp of the data since it is not automatically modified by the Recirculation Controller (Figure 4-10).

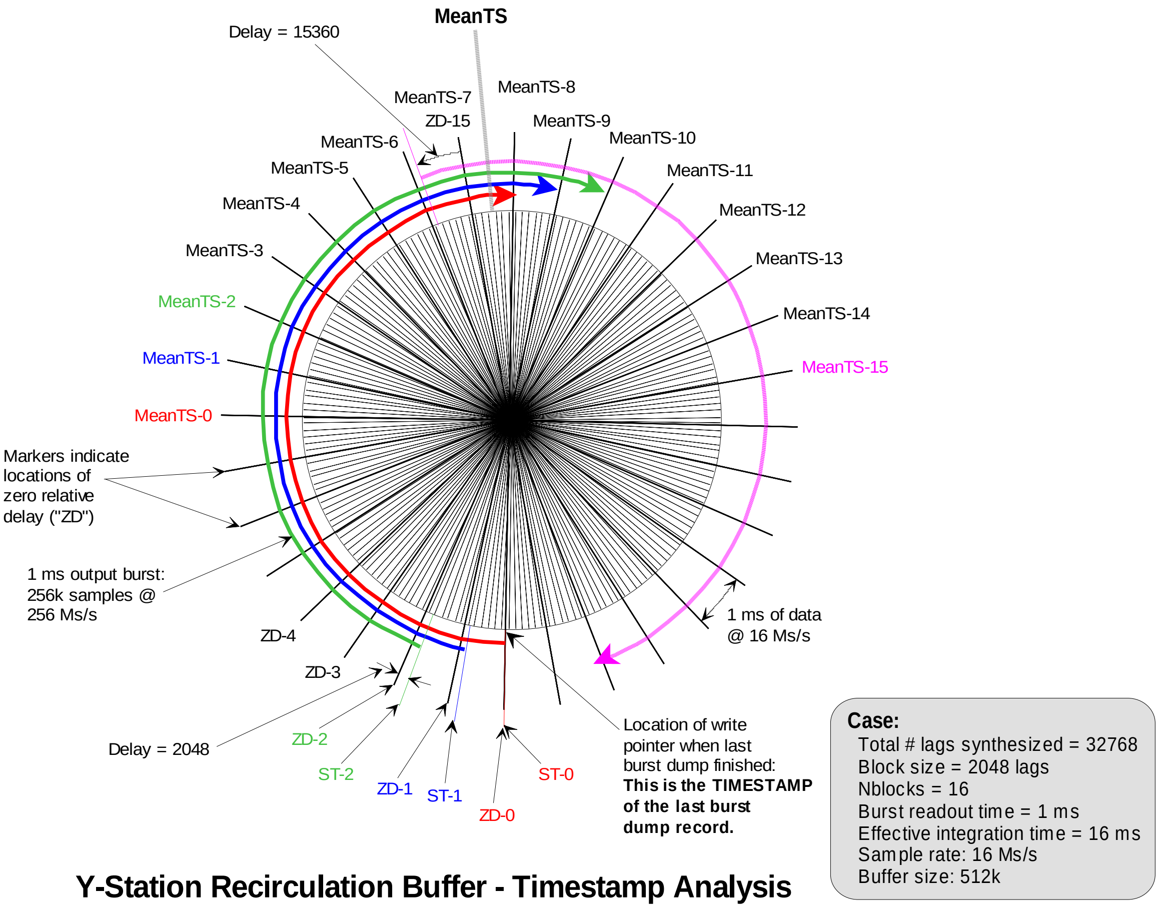
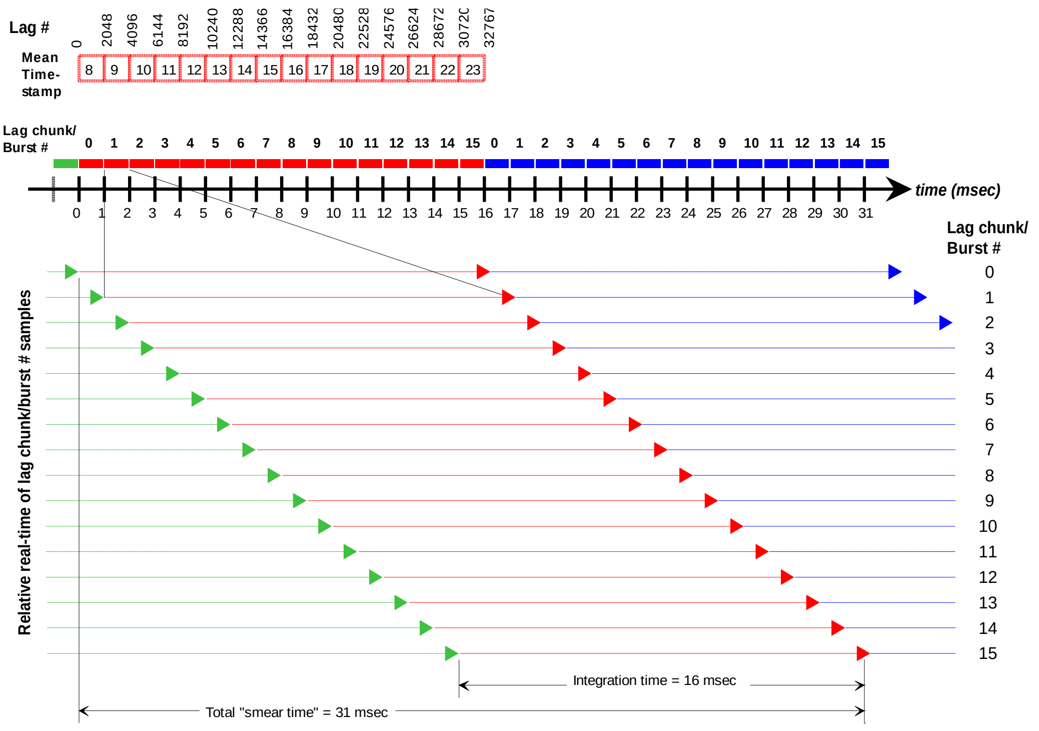

MeanTS
Delay = 15360
MeanTS-7
ZD-15
MeanTS-8
MeanTS-9
MeanTS-10
MeanTS-6
MeanTS-5
MeanTS-11
MeanTS-12
MeanTS-4
MeanTS-3
MeanTS-13
MeanTS-2
MeanTS-14
MeanTS-1
MeanTS-15
MeanTS-0
Markers indicate locations of zero relative delay ("ZD")
1 ms output burst: 256k samples @ 256 Ms/s
1 ms of data @ 16 Ms/s
ZD-4
ZD-3
Delay = 2048
ZD-2
ST-2
ZD-1
ST-1
ZD-0
ST-0
Location of write pointer when last burst dump finished: This is the TIMESTAMP of the last burst dump record.
Case:
Total # lags synthesized = 32768
Block size = 2048 lags
Nblocks = 16
Burst readout time = 1 ms
Effective integration time = 16 ms
Sample rate: 16 Ms/s
Buffer size: 512k
Y-Station Recirculation Buffer - Timestamp Analysis

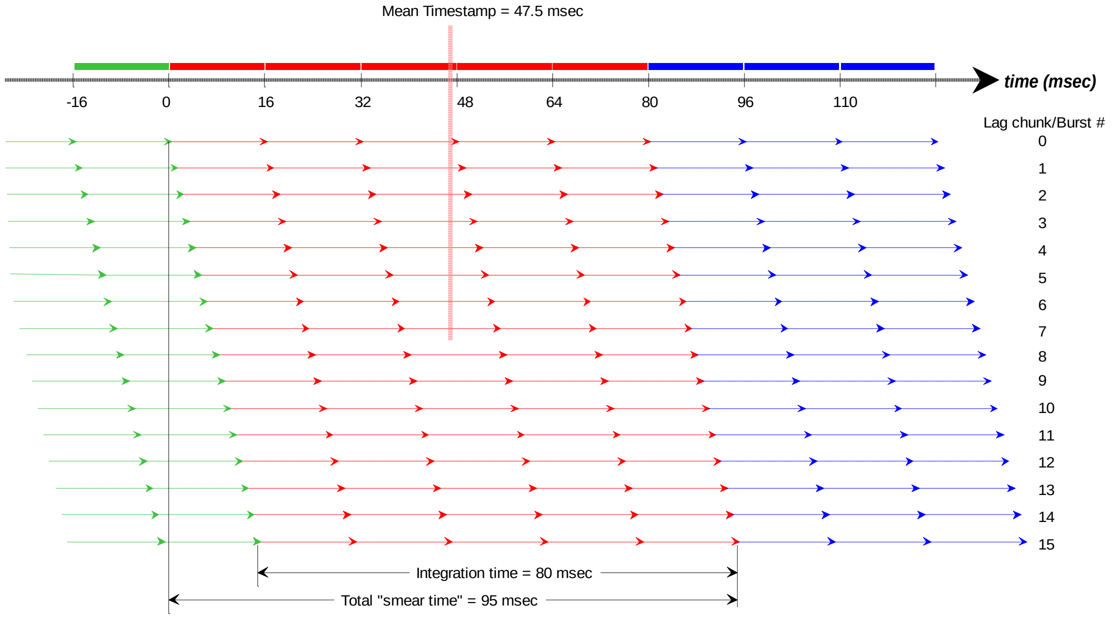
Figure 4-15 Circular buffer diagram of a real recirculation configuration to illustrate the time skew that each recirculation burst experiences. The "ZD-n" pointer is the zero-delay read pointer for the nth lag block. The "ST-n" pointer is the actual start read pointer for the nth lag block. Bold arrows indicate burst read pointer ranges for several bursts. The mean time stamp (MeanTS-n) is shown for each nth burst and the overall mean timestamp for the cross-power spectrum is "MeanTS". In this example, we are recirculating by a factor of 16 with a 512k recirculation buffer and a correlator chip integration time of 1 millisecond. The total integration time is 16 milliseconds, but the actual "smear" time is 31 milliseconds.

Perhaps a simpler diagram that illustrates this effect is shown in Figure 4-16. In this diagram the bursts shown along the time axis are expanded to show the relative real time of the data samples for each burst. This diagram clearly shows that a time shift of 1 millisecond is present for each consecutive burst and is for the case where one complete 32768-lag result is produced with a single burst of data for each chunk of the lag chain. In this case the total smear time is nearly double the actual integration time. However, if the integration time is increased so that many bursts are integrated for one result, the smear time starts to approach the integration time. This effect is illustrated in Figure 4-17. In this example, five bursts are integrated for an integration time of 80 milliseconds, and a total smear time of 95 milliseconds.



Case: 16 x recirculation with original real-time sample rate of 16 Ms/s, 512 k buffer with 256k sample burst size.

Figure 4-16 Diagram illustrating the relative real time of each burst of data with 16X recirculation and 1 millisecond burst times. In this case, even though the integration time is only 16 milliseconds, the actual “smear time” of the data is 31 milliseconds.



Case: 16 x recirculation with original real-time sample rate of 16 Ms/s, 512 k buffer with 256k sample burst size. 16 msec frame integration; total integration time 80 msec; "smear time" 95 msec.

Figure 4-17 Example illustrating the reduction in total smear time compared to the actual integration time when many (five in this case) bursts are integrated to yield one result.

As will be shown later it will be possible to reduce the burst length, and thus the correlator chip dump period and smear time, under configuration control. This may be useful in those cases where the number of lags being synthesized is not too large and there is a desire to minimize the smear time.

4.1.4.2 Wide-band Recirculation

Recirculation is normally used when the original sampled sub-band data is at a lower rate than the correlator chip is capable of running at. In that case, the write pointer advances more slowly than the read pointer during the burst. However, there is nothing in the design that prevents recirculation from being active even when the data is being written to the memory at the highest sample rate. In this case, the Recirculation Controller works to time-multiplex the synthesis of more lags than the correlator chip is capable of producing. There is a penalty however, in that the reduction in sensitivity is the square root of the recirculating factor because each lag chunk does not integrate all of the data. (Also, this capability would only be available on two basebands because of the limited recirculation memory width.) A "time burst" diagram illustrating this capability is shown in Figure 4-18. Despite the sensitivity loss, under certain circumstances it may be useful to obtain more spectral channels than would otherwise be available. The sensitivity loss may be partially made up for by resolving spectral lines that would otherwise be unresolved.

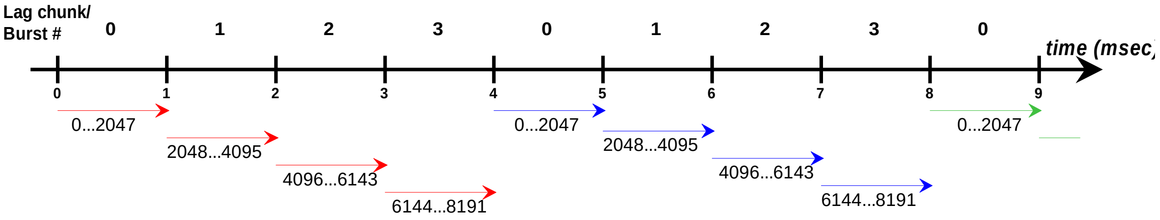


Figure 4-18 Time burst diagram of wide-band recirculation with a recirculation factor of 4. Here, 8192 lags are synthesized with a 2048-lag correlator chip resulting in a factor of 2 penalty in sensitivity, but with a factor of 4 more spectral channels.

There is another special case of wide-band recirculation that is really not recirculation at all since the memory buffer is simply used to synthesize delay. In this case, each sub-band correlator is correlating one chunk of the total lags *all of the time*. Here, the X or Y recirculation block number for a particular Recirculation Controller is constant and simply indicates how much delay to synthesize. Data is continuously written to the memory buffer, and read from the memory buffer at the same rate. The 'CLRS' signal (Figure 4-9) is not asserted when there is a dump since data going to the correlator chip is always contiguous in time. There is no sensitivity penalty in this mode since a particular correlator chip is always correlating one chunk of the lag chain. This mode will be useful if it is desired to obtain more spectral points across a full bandwidth (i.e. 128 MHz) sub-band than one correlator chip is capable of without sacrificing sensitivity (but at the expense of being able to correlate fewer total sub-bands).

4.1.5 Control and Synchronization Issues

4.1.5.1 Recirculation Real-Time Control

The recirculation algorithm developed in the previous section is simple and robust. The data gets written to the circular buffer in a free-wheeling fashion at the original sample rate. Read pointers are a function of the individual write pointers and the desired synthesized delay. However, in order for the read pointers in the X and Y Recirculation Controllers to be at the correct relative delay, it is necessary to logically synchronize them to a common reference. Fundamentally, this common reference is TIMECODE since this signal contains epochs that all events are synchronized to. Practically speaking however, it is necessary to not only synchronize the read pointers to some common epoch, but also provide for a way of telling each of the X and Y recirculation controllers what the current lag block (Figure 4-13) is so that the correct synthesized delay can be set. A proposed method of doing this is to use DUMPTRIG for synchronization. DUMPTRIG will synchronize the generation of the X and Y 'recirculation blocks' (Figure 4-9) within each recirculation controller as well as tell the correlator chip and LTA controller via the dump command (Figure 4-5) where to put the data. Double checking of recirculation synchronization can be done on the correlator chip, because it should detect DUMP_ENn-X and DUMP_ENn-Y pulses at exactly the same instant in time. The steps for recirculation synchronization are as follows:

1. Ensure that all DUMPTRIGs are synchronized to a TIMECODE epoch (the 'T' bit) within the Recirculation Controllers. Use the dummy DUMPTRIG command '111' (synchronization test frame) to do this during initialization and on a periodic basis[21].

2. On each DUMPTRIG dump command, take a snapshot of the write pointer. Add (or subtract) ½ the buffer length to the pointer to form the zero-delay read pointer. This pointer, in the X and Y Recirculation Controllers will point to data with a zero relative delay.

3. On each DUMPTRIG dump command, update the recirculation block counter to form the X and Y recirculation block numbers. This number times the size of the block (in samples of delay) is subtracted from the associated zero-delay read pointer. Always reset the recirculation block counter when the dump command '100' is encountered, and on the first occurrence of the '010' command after a sequence of '000's or '001's. This will ensure that the counters get reset and resynchronized from time to time.

4. Start the data burst at some time after the dump command—this time must be the same in X and Y Recirculation Controllers. The data burst will be free-wheeling until the next DUMPTRIG dump command (that is not a synchronization test) has come along. Thus, the period between dump commands determines the duration

[21] To allow for hot-swapping a Baseline Board without having to interrupt the operation of other Baseline Boards.

of the data burst. It can be shortened to reduce integration time skew or it can be shortened (within the correlator chip readout performance limits) if recirculation is used with pulsar phase binning.

This process is illustrated in Figure 4-19. The figure shows an initial command to dump and discard data followed by commands to dump and add data to the LTA. Finally, to produce an LTA result ready for transmission on the FPDP, a series of ‘010’ dump commands are issued. Note the phase bin numbering (i.e. ‘PB0’, ‘PB1’...) and the ‘Recirc Block’ numbering.

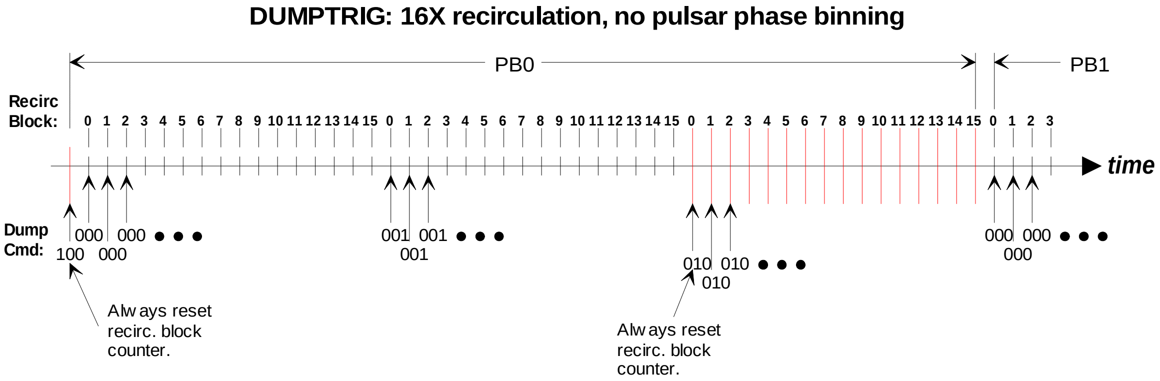


Figure 4-19 DUMPTRIG timing diagram with 16X recirculation active. This diagram illustrates how DUMPTRIG dump commands control the recirculation block counter, the phase bin number (*every* dump requires a phase bin number), and tells the LTA controller what to do with the data.

Using this synchronization method, the only configuration information the Recirculation Controller needs to know[22] is what the start recirculation block number is[23], how big each recirculation block is in samples of delay, and whether it is an X or Y controller. All other associated real-time actions are triggered by DUMPTRIG. Note that in the instances where different stations have different integration times (i.e. long integration times), it is necessary to ensure that X and Y recirculation controllers are synchronized. This can be achieved by using the same short integration time (e.g. 1 millisecond) in both stations, but changing the long integration time (i.e. when ‘010’ commands are issued) in one station. Also, the initial ‘100’ command (dump and discard) generation must be synchronized across all stations.

Finally, if recirculation is used, the effective number of LTA phase bins that are available also drops. This is because each lag chunk that is synthesized must have its own buffer space[24], for a specified DUMPTRIG phase bin, in the finite sized LTA. For example, if 4X recirculation is used, then only 2 banks of 250 phase bins each are available. If 16X recirculation is used, then only 2 banks of 62 phase bins can be used (we can’t use 0.5 of

[22] Aside from whether or not recirculation is active etc.

[23] i.e. what it is when it gets reset—since within a sub-band correlator it could be non-zero if a particular sub-band correlator is synthesizing only one chunk of much longer synthesized lag chain

[24] The LTA controller automatically calculates the actual buffer that is used in the LTA based on the number of recirculation blocks being synthesized, the current recirculation block, and the DUMPTRIG phase bin number.

a phase bin). It is important that the phase bin specified in the DUMPTRIG frame is within acceptable limits for the current configuration, otherwise undetected data overwrite errors will occur.

4.1.5.2 Normal Dumping Real-Time Control

The previous section defined a method for real-time recirculation and control. In that case a number of short integrations ('000' or '001' commands) were followed by a burst of 'Last dump' commands ('010'). If recirculation is not active (or, some sampled data streams are being correlated without recirculation, while others are being correlated with recirculation active), then the dump command sequence is different. An example sequence is shown in Figure 4-20. The DUMPTRIG protocol allows this sequence or the recirculation sequence to co-exist and use the same or different dump epoch. This is important because it allows mixed recirculation and non-recirculation operation.

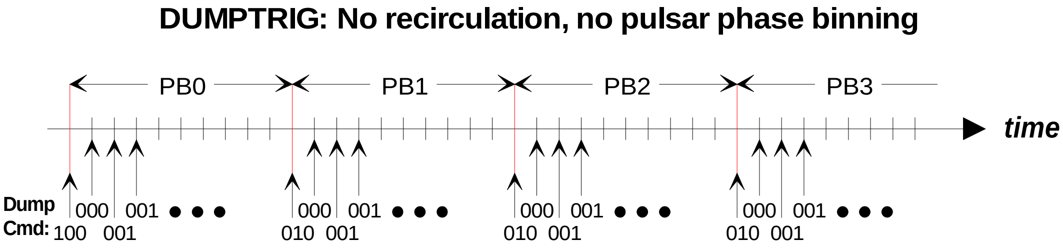


Figure 4-20 DUMPTRIG commands and phase bins ('PBn') when recirculation is not active. The sequence of commands is different than with recirculation. Note that the DUMPTRIG protocol allows this sequence or the recirculation sequence to co-exist and use the same or different dump epoch (as long as the epoch frequencies are harmonically related, that is).

It is important to note that every DUMPTRIG frame must contain a phase bin number that tells the LTA controller where in its buffer memory to put the data. If pulsar phase binning is not active, then the number of phase bins (out of the two banks of 1000 each that are available) that need to be used is a free parameter. The number that are used will depend on how much output buffer space is required—more if the output dump rate is high, and less if the output dump rate is low.

4.1.5.3 Pulsar Phase Binning Real-Time Control

DUMPTRIG and LTA operation when pulsar phase binning is active is similar to when recirculation is active except that there is no recirculation block. There are a number of '000' and '001' dumps, followed by a burst of '010' dumps that tell the LTA controller to flag the data as ready for transmission out on the FPDP interface. The LTA memory is such that there are 2 banks of 1000 phase bins[25] each. When one bank is full, the other bank is used while the first bank of data gets transmitted on the FPDP. An example timeline of this process is shown in Figure 4-21. In this figure there are 10 phase bins (0...9) all spaced equally in time. It is completely within the DUMPTRIG protocol to

[25] Or, 1 bank of 2000 phase bins if some blanking time while the bins are being dumped out the FPDP is acceptable.

bunch up (dump more rapidly) some phase bins and spread out (dump less rapidly) others as long as the correlator chip integration period is not violated and the correlator chip is not dumped too rapidly (see the section on the LTA controller).

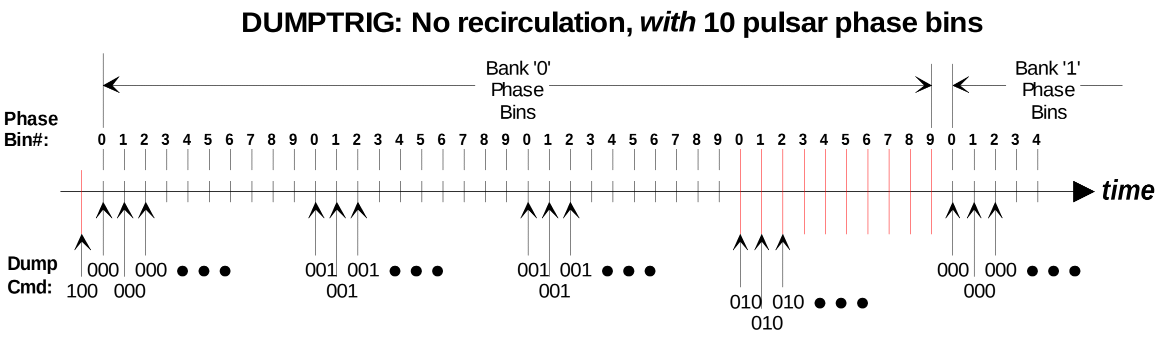


Figure 4-21 Example DUMPTRIG dump sequence with 10 pulsar phase bins. The dump command sequence is similar to that of recirculation.

4.1.5.4 Pulsar Phase Binning *and* Recirculation Real-Time Control

It may be desirable in some circumstances to have recirculation active at the same time that pulsar phase binning is active. There may be some cases when this is not feasible because of the small pulsar period and the complication of smearing the cross-power spectrum over a longer period of time than the actual integration time. However, the system configuration could be modified to still make it feasible by not recirculating too much, and by using just a small part of the recirculation buffer available (dumping fast). An example timeline showing 4X recirculation with 6 pulsar phase bins is shown in Figure 4-22. Note that the number of phase bins specified in DUMPTRIG is subject to the restrictions explained in the above section.

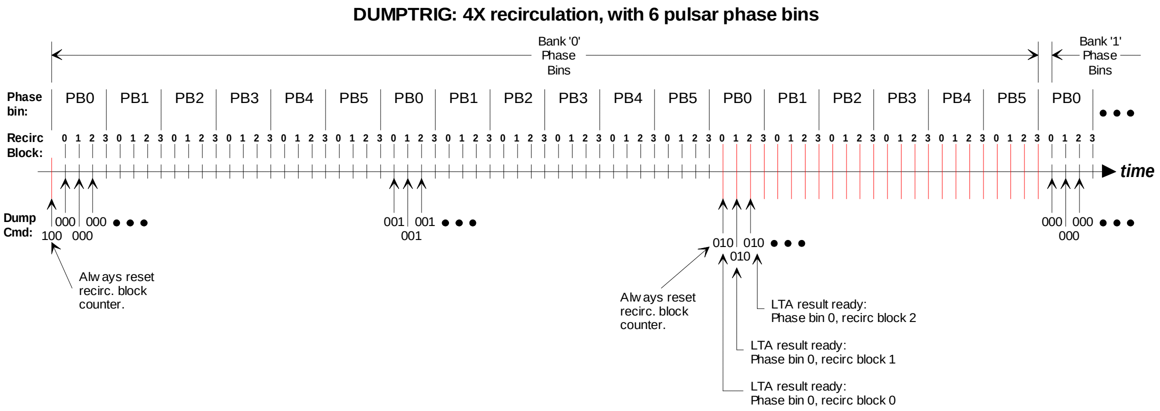

DUMPTRIG: 4X recirculation, with 6 pulsar phase bins
Bank '0' Phase Bins
Bank '1' Phase Bins
Phase bin:
PB0 PB1 PB2 PB3 PB4 PB5 PB0 PB1 PB2 PB3 PB4 PB5 PB0 PB1 PB2 PB3 PB4 PB5 PB0
Recirc Block:
0 1 2 3
time
Dump Cmd:
100 000 000
001 001 001
010 010 010
000 000 000
Always reset recirc. block counter.
Always reset recirc. block counter.
LTA result ready: Phase bin 0, recirc block 2
LTA result ready: Phase bin 0, recirc block 1
LTA result ready: Phase bin 0, recirc block 0

Figure 4-22 Example showing recirculation active when pulsar phase binning is active. In this example there is 4X recirculation and 6 pulsar phase bins.

4.2 Correlator Chip

4.2.1 Simplified Functional Description

The correlator chip is where final synchronization and cross-correlation of the X and Y data streams coming from Recirculation Controllers occurs. A 2048 complex-lag cross-correlator chip is proposed whereby both X and Y data streams are delayed in time and cross-correlated to produce both positively and negatively delayed lags. A simplified block diagram of an example 8-lag[26] section is shown in Figure 4-23 below. Lag numbering is a chosen convention that is consistent with Figure 4-13. In this architecture, phase is quantized to 4 bits, carried along with the data, and final baseline phase generation and phase rotation is performed at each lag. Although handling phase this way *may* require more silicon than the alternative complex-lag architecture [7], it results in power savings since phase is changing very slowly and contributes virtually nothing to chip power dissipation as it moves along its shift register path. It also requires only one data multiplier at each lag as shown in Figure 4-24—resulting in significant savings in power dissipation and silicon, particularly since the multiplier is a 4-bit multiplier. This architecture and its properties are well understood [2][0][6].

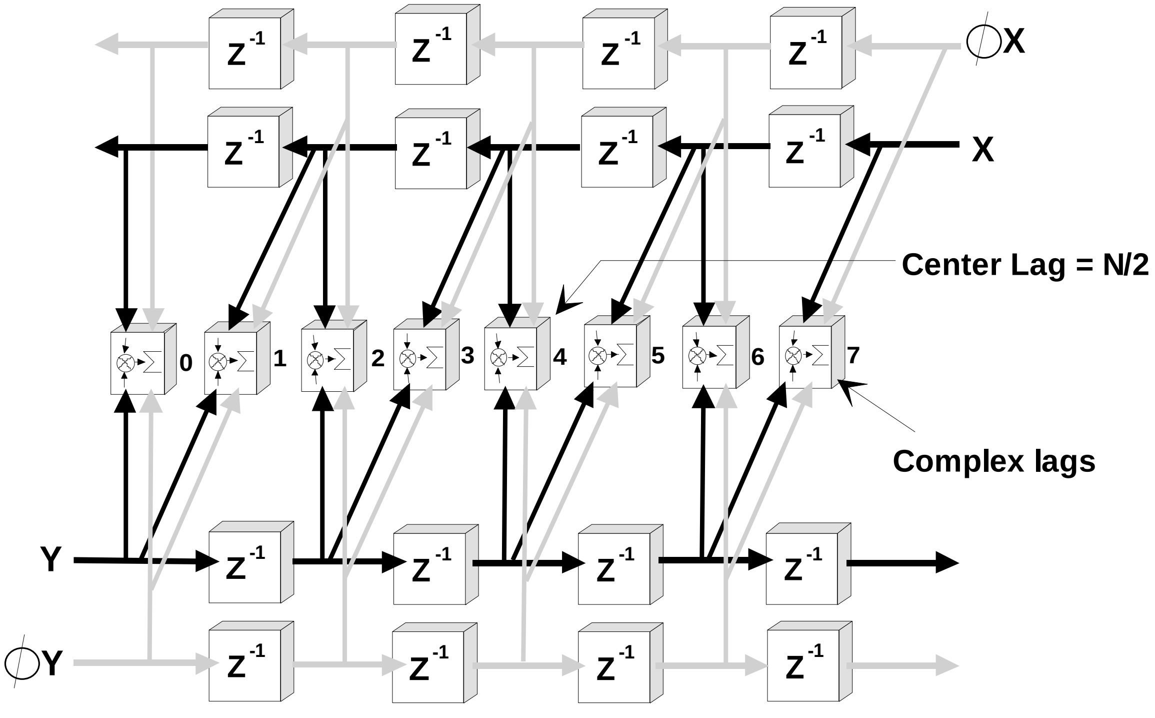


Figure 4-23 Simplified block diagram of an example N=8-lag section. Lag numbering is a chosen convention.

Within each complex lag the X and Y data multiplication is performed followed by complex phase rotation. This functionality is illustrated in Figure 4-24. Phase rotation

[26] In the actual correlator chip, each of the 16 lag sections will have 128 complex lags.

after the 4-bit multiplication is relatively simple and, in the 5-level case, can be performed with bit shifts and sign flips.

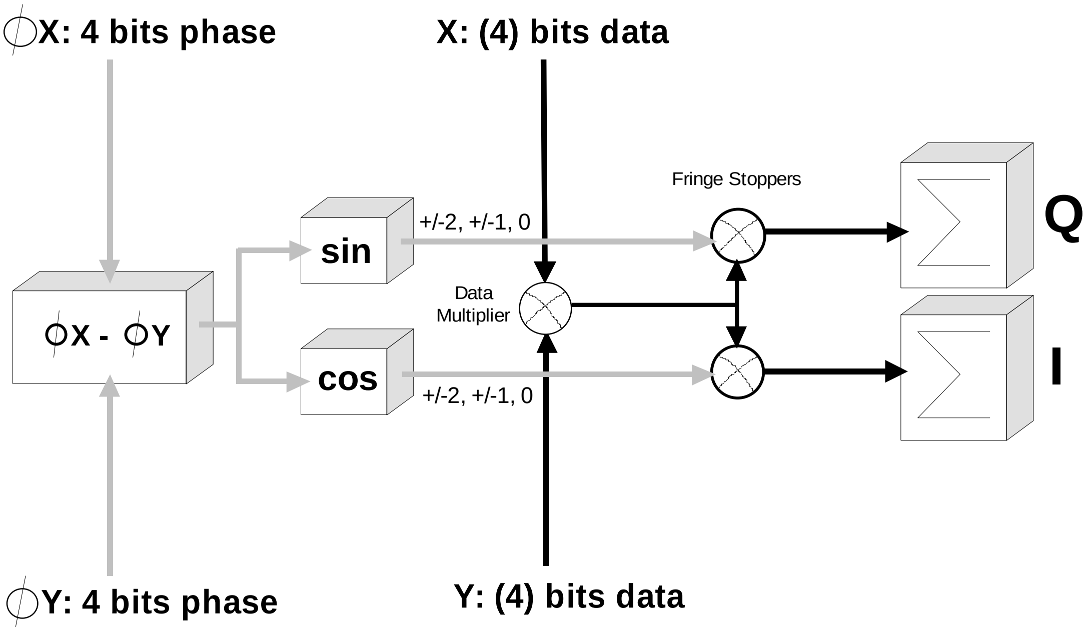


Figure 4-24 Functional block diagram of one complex-lag in the correlator chip. In this design, only one X*Y data multiplier is required—important to reduce silicon and power dissipation with a 4-bit multiplier. Five-level fringe stopping (complex phase rotation) is performed after data multiplication and can probably be done with relatively simple sign flips and bit shifts.

A simplified block diagram of the entire correlator chip lag arrangement is shown in Figure 4-25. In this arrangement, the correlator chip consists of four 'CCQs' ('Correlator Chip Quads'). Each CCQ is capable of producing all of the polarization products required for one baseband pair. The additional connections from CCQ#1 inputs to other CCQs are required to meet *minimal* connectivity requirements if narrower-band VLBA antennas eventually get correlated with VLA antennas in real time [5]. More general data routing could allow more flexibility in the correlator chip configuration if such data routing is not too costly in silicon and power dissipation.

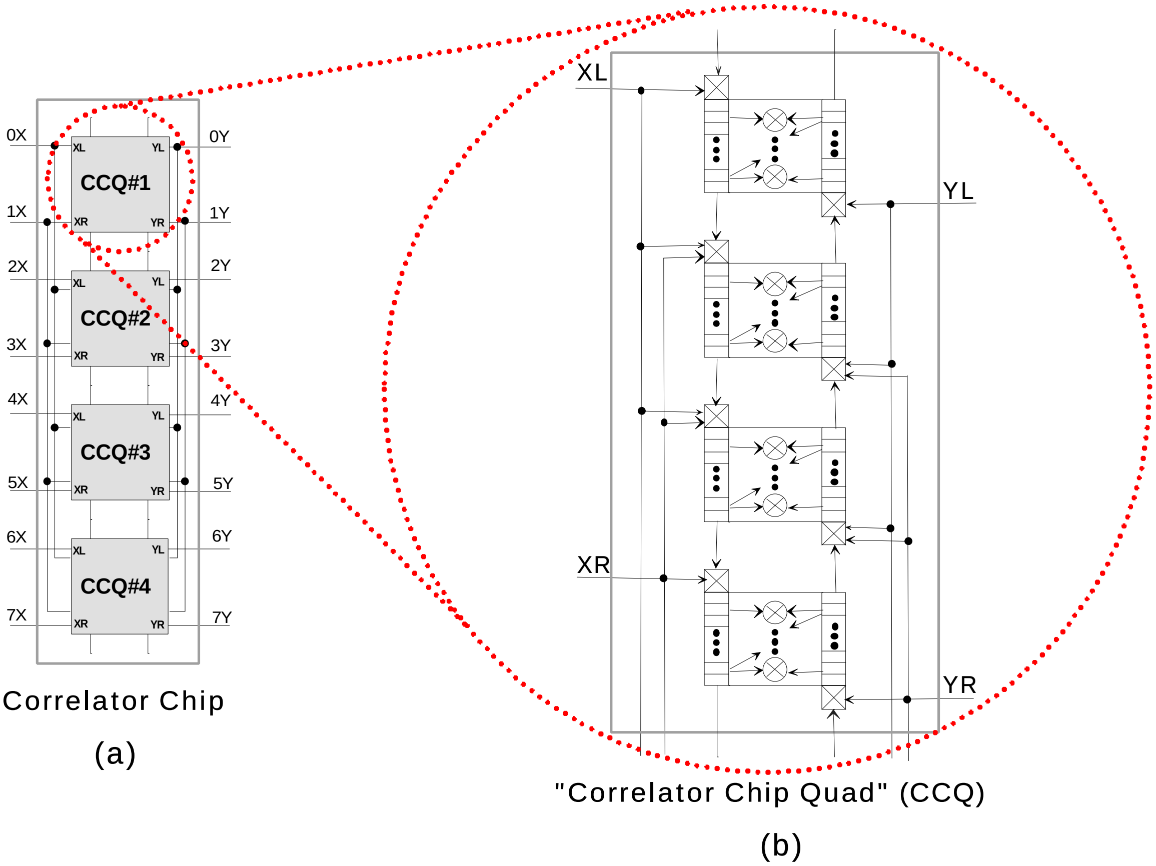


Figure 4-25 Simplified correlator chip Correlator Chip Quad (CCQ) and lag arrangement. There are four CCQs (a) and each CCQ contains four, 128-lag cross-correlators (b). CCQ#1 is a master in that every CCQ has access to its input data. Additionally, 128-lag correlator sections and CCQs can be concatenated to yield combinations of lag correlators up to a single 2048-lag configuration.

4.2.2 Detailed Functional Description

There are several points of consideration in the design of the correlator chip other than the basic lag architecture illustrated in the previous section. These points are as follows:

- Any "delay-path" differences between the X and Y station data due to varying conductor lengths in the correlator system must be absorbed in the correlator chip since it is in the correlator chip where the data from both stations is first brought together. The input sections of the chip must thus contain circuitry similar to that of Figure 4-11. However, in the correlator chip case one clock—either the X or Y clock—is chosen to be the clock that everything is eventually synchronized to. Sufficient integer-delay deskew circuitry must be built in to compensate for X and Y differential path delays.

- In order to know whether X and Y data is finally synchronized, a test vector generator, error detector, and deskew controller is needed to receive test vectors from the Recirculation Controllers. This test vector generator/detector is only invoked in test modes when the Recirculation Controllers are sending test vectors.

In normal ("on-the-sky") operation, the input circuitry must detect errors by looking for protocol violations and signal "eyes" as shown in the timing diagram of Figure 4-9.

- The correlator chip does not know much about recirculation. The only cues it gets are the 'CLRS' bit and holdoff in a particular DUMP_EN line and the RECIRC_BLK line associated with each dump. The CLRS bit simply tells the correlator chip that it is to clear the output of associated lag shift registers, holdoff prevents the chip from correlating until it is negated, and the RECIRC_BLK is captured and passed on to the LTA controller in the correlator chip output data frame.

- Based on internal switch settings, and X-station/Y-station selection, the correlator chip must route each DUMP_EN line to a particular set of lags in a particular CCQ. When recirculation is active, associated DUMP_EN lines from the X and Y Recirculation Controllers must be asserted coincidentally (once synchronized on the correlator chip). This is because dumping controls recirculation counters and pointers in the Recirculation Controllers. The correlator chip should be designed to check for this synchronization.

- Each set of 128 lags operates independently as far as dumping goes. Each set of lags has its own independent output data frame that is treated independently by the LTA controller.

- Dump overruns, where data has not been read out before another dump signal comes along, must be properly handled in the correlator chip. It is suggested that when a dump overrun occurs, existing data waiting to be read out is not disturbed—rather new data is discarded, and a status bit is set to indicate that one or more data dumps were discarded.

4.2.2.1 Black-Box Correlator Chip Diagram

A black-box diagram of the correlator chip is shown in Figure 4-26. In the figure, there are identical X and Y inputs, an LTA controller interface for data output, and a Monitor and Control Bus (MCB) interface for configuration and status monitoring. In this design there is no provision for allowing multiple correlator chips to be concatenated since this requires more I/O, is probably somewhat more costly, and is not a requirement for the EVLA[27]. However, when the chip is actually developed, this possibility should be considered since it results in a more general device (although it may be impossible to do because of synchronization, time-skew concerns, and increased die size and cost).

[27] i.e. to be able to tradeoff number of baselines processed for spectral resolution.

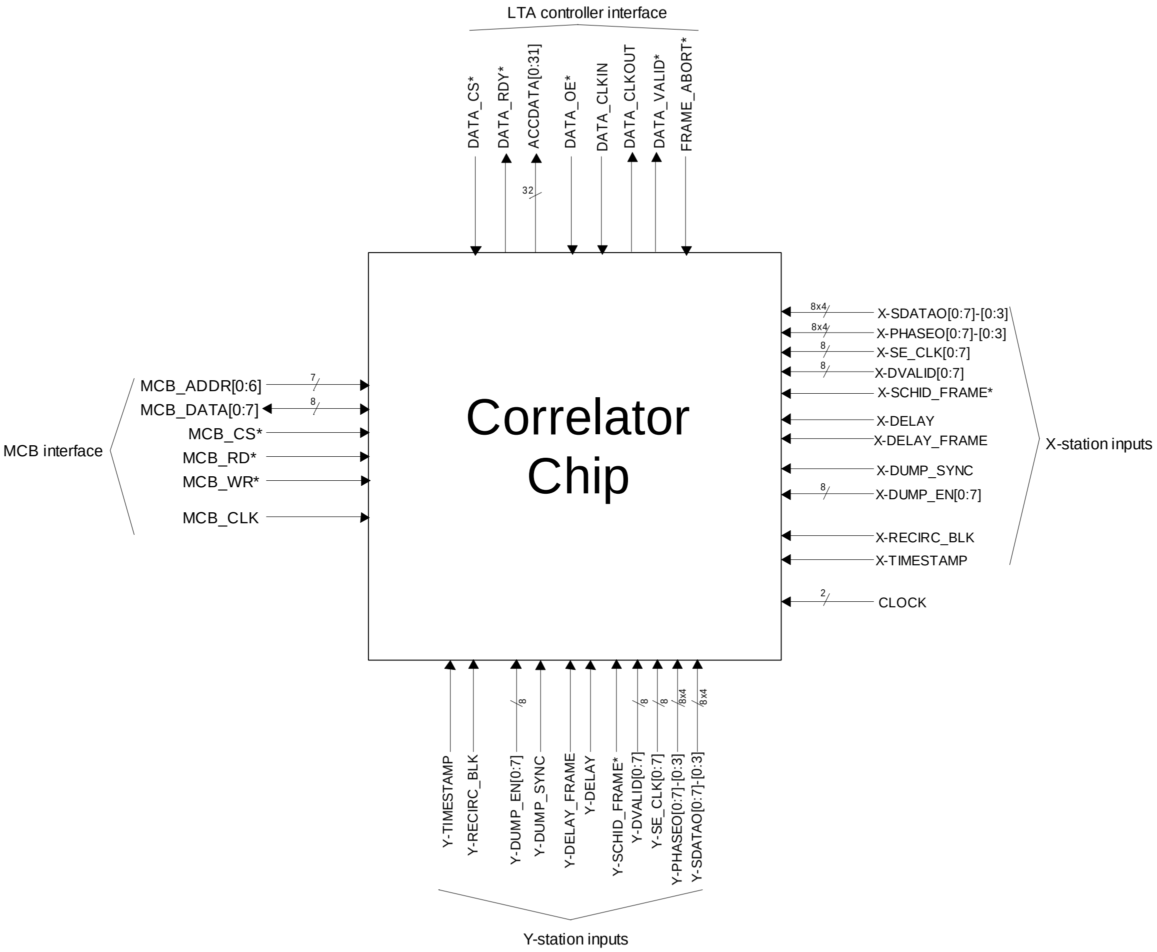


Figure 4-26 Correlator chip black-box diagram. There are X and Y inputs, an LTA controller interface and an MCB interface. The ‘CLOCK’ input is the 128 MHz clock that can come from either the X or Y Recirculation Controller since the safe assumption is made that X and Y input signals are skewed in time relative to the clock and each other.

X and Y station input functional timing is as defined in Figure 4-9. In Figure 4-27 below, functional timing for the LTA controller interface is defined. The LTA Controller activates the correlator chip by asserting the DATA_CS* line. If the correlator chip has data ready to transmit, it asserts DATA_RDY*. The LTA Controller then asserts DATA_OE* and the transfer begins. Each clock edge that DATA_VALID* is low contains valid data. Once the transfer is complete, the correlator chip negates DATA_RDY* at which time DATA_OE* is negated by the LTA Controller. If the correlator chip has more data ready then, if DATA_CS* is still asserted, it will assert DATA_RDY* once it detects DATA_OE* negation and the transfer process will start again. Note that the chip is sequential readout so that once data transfer starts it can only be *aborted* using the FRAME_ABORT* signal. Aborting data transfer causes the lag buffer registers to be cleared, ready for new lag data on the next dump. This interface is fully chip-to-chip synchronous at 128 MHz. That is, DATA_CLKIN from the LTA Controller is 128 MHz.

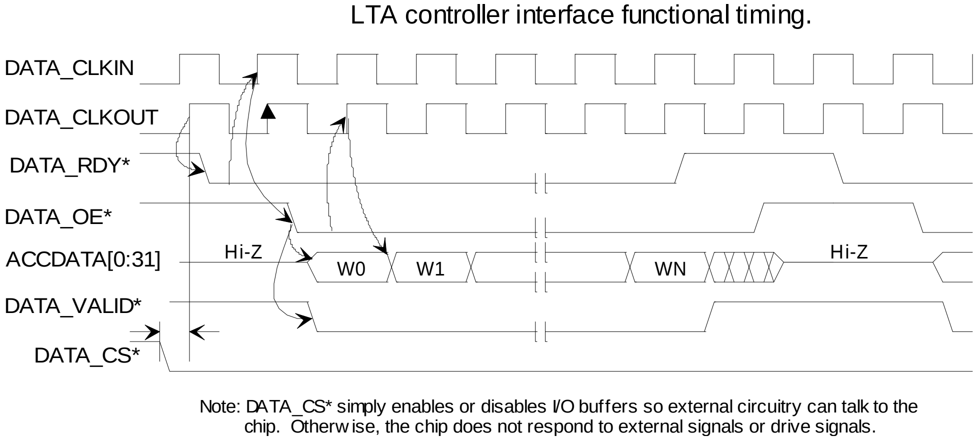

LTA controller interface functional timing.
DATA_CLKIN
DATA_CLKOUT
DATA_RDY*
DATA_OE*
ACCDATA[0:31]
DATA_VALID*
DATA_CS*
Hi-Z
W0
W1
WN
Hi-Z
Note: DATA_CS* simply enables or disables I/O buffers so external circuitry can talk to the chip. Otherwise, the chip does not respond to external signals or drive signals.

Figure 4-27 Correlator chip LTA controller interface functional timing. Once enabled by the LTA Controller and if data is ready, the correlator chip transmits data to the LTA Controller. Once transmission is complete, the particular 128-lag section buffer registers are cleared to capture more data. Not shown in this timing diagram is FRAME_ABORT* that aborts transmission of the current frame and clears the lag section buffer registers.

Functional timing diagrams for MCB READ and WRITE cycles are shown in Figure 4-28. This is a simple 8-bit synchronous interface. Seven address lines are shown—the actual number of address lines needed by the correlator chip requires further investigation.

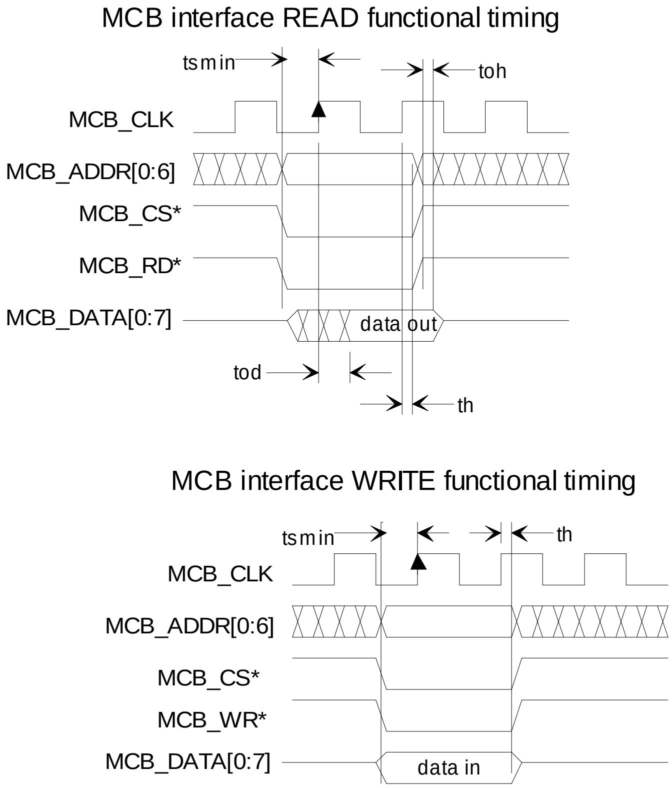


Figure 4-28 Correlator chip MCB interface READ and WRITE cycle functional timing diagrams. Seven address bits are shown, but the actual number required requires further definition. MCB_CLK has a nominal maximum frequency of 128 MHz—although it would normally be much lower than that.

4.2.2.2 Correlator Chip Block Diagram

A top-level block diagram of the correlator chip is shown in Figure 4-29. It contains straw-man concepts for several blocks and inter-block connections, each of which is described in the following paragraphs.

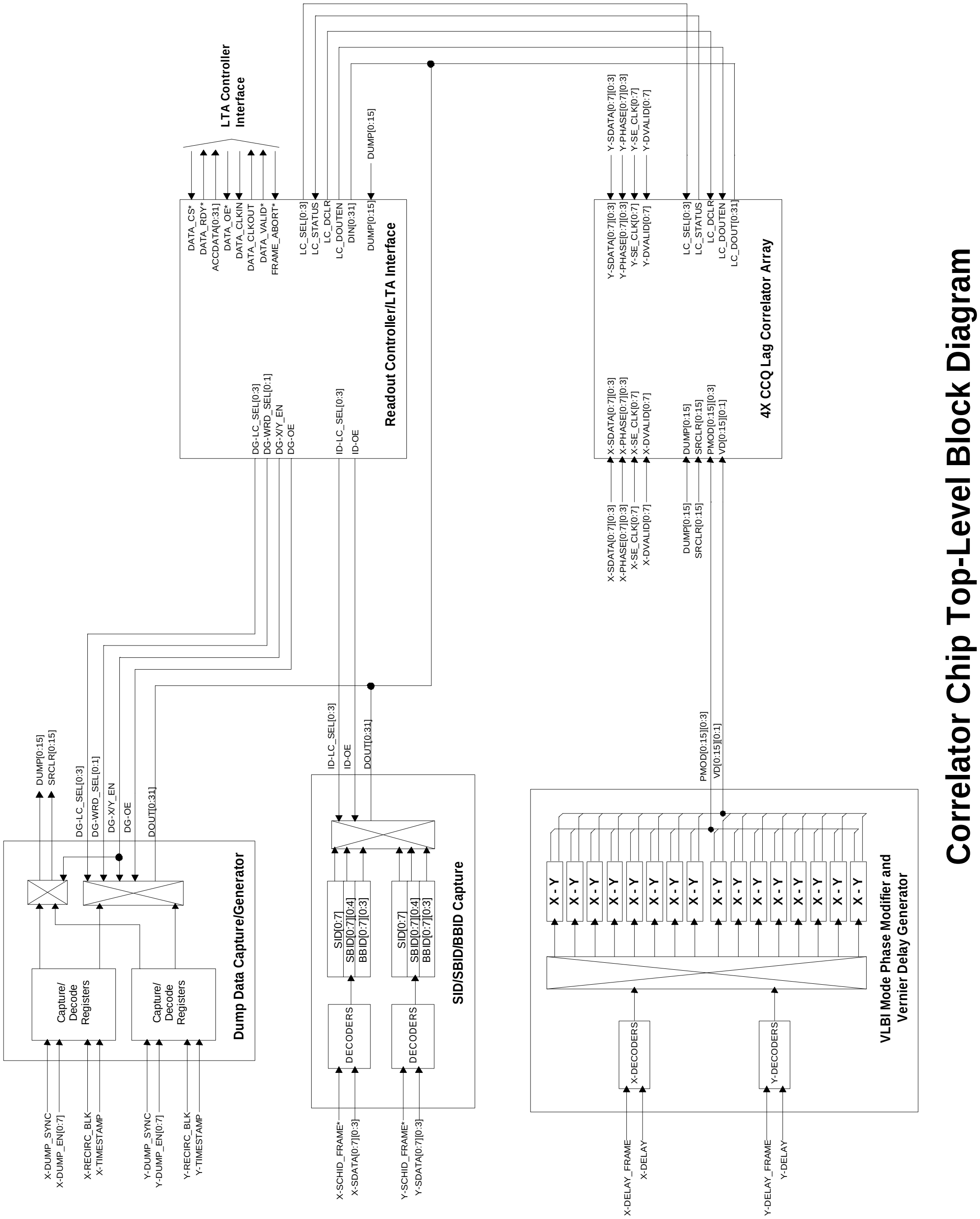


Figure 4-29 Correlator chip top-level block diagram. This diagram contains straw-man concepts for all functional blocks and interconnect signals required for correlator chip functions. Not shown is the logic required for synchronization of X and Y input data (test vector receivers, de-skew controllers etc.).

4.2.2.2.1 Dump Data Capture/Generator

This block receives X and Y dump signals (DUMP_SYNC, DUMP_EN, RECIRC_BLK, TIMESTAMP) and generates internal dump signals and shift register clear signals (DUMP[0:15], SRCLR[0:15]) that go to the 16, 128-lag correlator '**lag cells**'. It also captures the phase bin, X and Y recirculation block numbers, and timestamps for eventual readout by the internal Readout Controller/LTA Interface. Control signals in the diagram include:

- DG-LC_SEL[0:3] – Address to select a particular lag cell's data.
- DG-WRD_SEL[0:1] – Address to select a header word for transmission to the LTA Controller. The correlator chip output data frame is defined in a following section.
- DG-X/Y_EN – Determines whether dump signals are generated by X or Y dump signaling. This is common to all lag cells (i.e. for the entire chip).
- DG-OE – Enables DOUT[0:31] drivers for a particular lag cell's data.
- DOUT[0:31] – This data bus will include selected header words of the correlator chip output data frame.

Not shown are configuration registers that determine, for each lag cell, the mapping of DUMP_EN inputs to DUMP and SRCLR outputs.

4.2.2.2.2 SID/SBID/BBID Capture

This block captures embedded identifiers SID, SBID, and BBID from the input X and Y SDATA streams and makes them available to the Readout Controller/LTA Interface in the output header format (see following section on correlator chip output data frame). An output register, selected by ID-LC_SEL[0:3] is available for each of the 16 lag cells. Not shown is configuration logic required to map inputs to output registers.

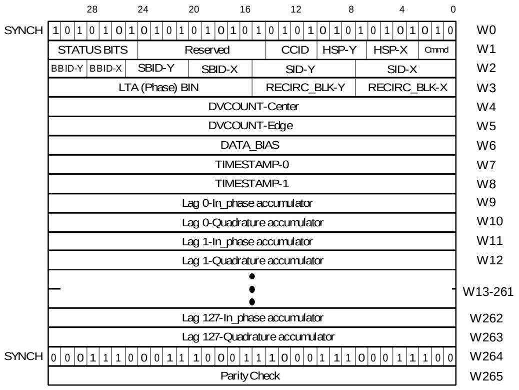
4.2.2.2.3 VLBI Mode Phase Modifier and Vernier Delay Generator

This block contains logic that, when enabled for each of the 16 lag cells, captures X and Y delay information and generates the 4-bit 'phase modifier' (PMOD) and 'vernier delay' (VD) control. These signals are required when 'WIDAR-style' sub-sample delay tracking is not being performed in the Recirculation Controllers. Normally, this only occurs when the original sample data stream into the correlator is *not* split into sub-bands with FIR filters. In this case, the point of zero-phase error is (nominally) set to the center of the band—requiring baseline-based phase offsets and vernier (fine) delay—and the coherence loss at the edge of the band is 10%. Although this is referred to as a VLBI mode correction, it is not strictly limited to VLBI. Details for these calculations are in [2].

4.2.2.2.4 Readout Controller/LTA Interface

This block is responsible for transmitting dumped correlator data out on the LTA Controller Interface. The correlator chip output data frame contains header data, lag data, and error detection information. Header data is obtained from the Dump Data Capture/Generator and SID/SBID/BBID Capture blocks and lag data (and some status) is obtained from the 4X CCQ Lag Correlator Array. The correlator chip output data frame is shown in Figure 4-30.

Correlator chip to LTA controller data frame.



W2: Identifier bits:

SID-X (b7-b0), **SID-Y** (b15-b8): 8-bit X and Y station ids. A station consists of input to 4 Station Boards.

SBID-X (b20-b16), **SBID-Y** (b25-b21): 5-bit X and Y sub-band ID (0...17). The data comes from this particular sub-band output of a Station Board.

BBID-X (b28-b26), **BBID-Y** (b31-b29); 3-bit X and Y baseband ID (0...7).

W3:

RECIRC_BLK-X(b7-b0): X-input recirculation block: the number of delay blocks inserted into the X data path.

RECIRC_BLK-Y(b15-b8): Y-input recirculation block: the number of delay blocks inserted into the Y data path.

LTA (Phase) BIN(b31-b16): The LTA phase bin to put this data into. b31 indicates buffer bank 0 or bank 1.

W4, W5:

DVCOUNT-Center: count of number of valid samples taken at the center of this lag chain.

DVCOUNT-Edge: count of number of valid samples taken at the edge of this lag chain.

W6:

DATA_BIAS: exact DC bias that is in accumulator data and that must be removed before normalizing with data valid count.

W7, W8:

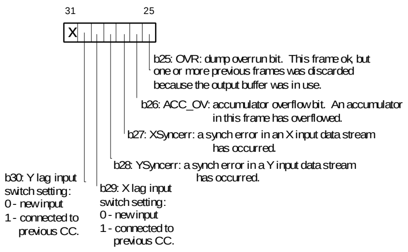
Timestamp-0: number of 1 second ticks since last MAJOR EPOCH.
Timestamp-1:
  MAJOR EPOCH (b31-b29):
    000 = 00:00:00 Jan 1/1970
    all others: reserved for future definition.
  Sample Count (b28-b0): number of samples at CLOCK rate since last 1 second tick.

W1:

Command: (b2-b0)
000 - first dump of data into LTA. Just save in LTA bin.
001 - add to existing LTA data and save in LTA bin.
010 - last dump: add to LTA data; flag LTA bin as ready.
011 - speed dump: bypass LTA directly to output. The only operation is that DATA_BIAS is removed.
1xx - Reserved.

HSP-X (b6-b3): Harmonic suppression phase-X. X-phase has been offset by this quantity.
HSB-Y (b10-b7): Harmonic suppression phase-Y. Y-phase has been offset by this quantity.
CCID (b14-b11): Chip cross-correlator ID (0...15). Indicates which physical correlator in the chip this data comes from.

STATUS BITS (b31-b25):


31
25
X
b25: OVR: dump overrun bit. This frame ok, but one or more previous frames was discarded because the output buffer was in use.
b26: ACC_OV: accumulator overflow bit. An accumulator in this frame has overflowed.
b27: XSyncerr: a synch error in an X input data stream has occurred.
b28: YSyncerr: a synch error in a Y input data stream has occurred.
b30: Y lag input switch setting: 0 - new input 1 - connected to previous CC.
b29: X lag input switch setting: 0 - new input 1 - connected to previous CC.

W9-W263:

Lag 0-127 In-phase and Quadrature accumulator data. This data shou have DATA_BIAS removed before being saved into the LTA to maximize LTA integration time.

W0, W264:

Start and end SYNCH words.

W265:

Parity Check. Each bit represents a parity check of all like bit positions from W0-W264.

Figure 4-30 Correlator chip output data frame. Each frame contains header information and lag data from one of the 16, 128-lag lag cells. Data valid counts are provided that are at the center lag (lag N/2) and at an edge lag (lag 0). By providing counts at these locations, a center lag and an edge lag data valid count are always available even when multiple lag cells are concatenated to form a longer lag chain. Two data valid counts can help to mitigate the systematic effects of data valid blanking that occurs at the same time in both X and Y stations. Note that even if multiple 128-lag cells are concatenated, each output data frame only ever contains data from one lag cell.

Two things are of note in the figure. First, two data valid counts are provided, one at the center lag and one at lag 0 (using our chosen lag numbering convention—see Figure

4-13). Second, the data bias that is present in the lag data is in the frame (DATA_BIAS) so that it can be removed before the LTA Controller integrates the data. This minimizes the memory width in the LTA Controller RAM since useless bias information is not using up valuable memory width.

An additional capability that is not supported by the definition of the data frame in Figure 4-30 is the ability to output less that the total 128 lags for a correlator lag cell. This would require an additional field that indicates exactly how much lag data is in the frame. With proper configuration, the LTA Controller could be configured to only keep the central few lags to support this mode. This capability may be attractive to minimize data output rates where very high-speed real-time dumping of data is required for continuum observations (i.e. probably solar observations).

4.2.2.2.5 4X CCQ Lag Correlator Array

This block contains the four CCQs ('Correlator Chip Quads') that were shown in simplified form in Figure 4-25. The block contains only the information required for correlation, dumping, fine delay and phase control (for VLBI mode), and readout control/data signals. Not shown are inputs for individual lag-cell data selector configurations. Some selected control signals for this block are as follows:

- DUMP[0:15] – Dump pulse for each lag cell. This pulse is asserted for a single (128 MHz) clock cycle, causing the particular lag cell to stop correlating, and dump its data.
- SRCLR[0:15] – Shift register clear for each lag cell. This is asserted after DUMP and sometime during the dead time when the particular lag cell has stopped correlating. This is normally only asserted when this lag cell is being used for recirculation to prevent correlation of time-discontinuous data.
- LC_SEL[0:3] – Address select for a particular lag cell for data readout/clear.
- LC_STATUS – Output that is asserted if the selected lag cell has data waiting to be read out.
- LC_DCLR – Assert this to clear the selected lag cell's output data buffer. Useful if the FRAME_ABORT* signal is asserted.
- LC_DOUTEN – Enable output data drivers and output data clocking. Each cycle that this is asserted causes new data in the sequence of output data to appear on LC_DOUT[0:31]. The sequence of output data is closely related to the correlator chip output data frame.

A more detailed block diagram of the 4X CCQ Lag Correlator Array is shown in Figure 4-31. This figure contains all logical functionality and connectivity for the array. Figure 4-32 is a simplified block diagram of a CCQ. Each CCQ contains four, 128 complex-lag 'cells' and associated switching circuitry to allow selection of data from one of three sources. Not shown is data selector configuration information for each lag cell.

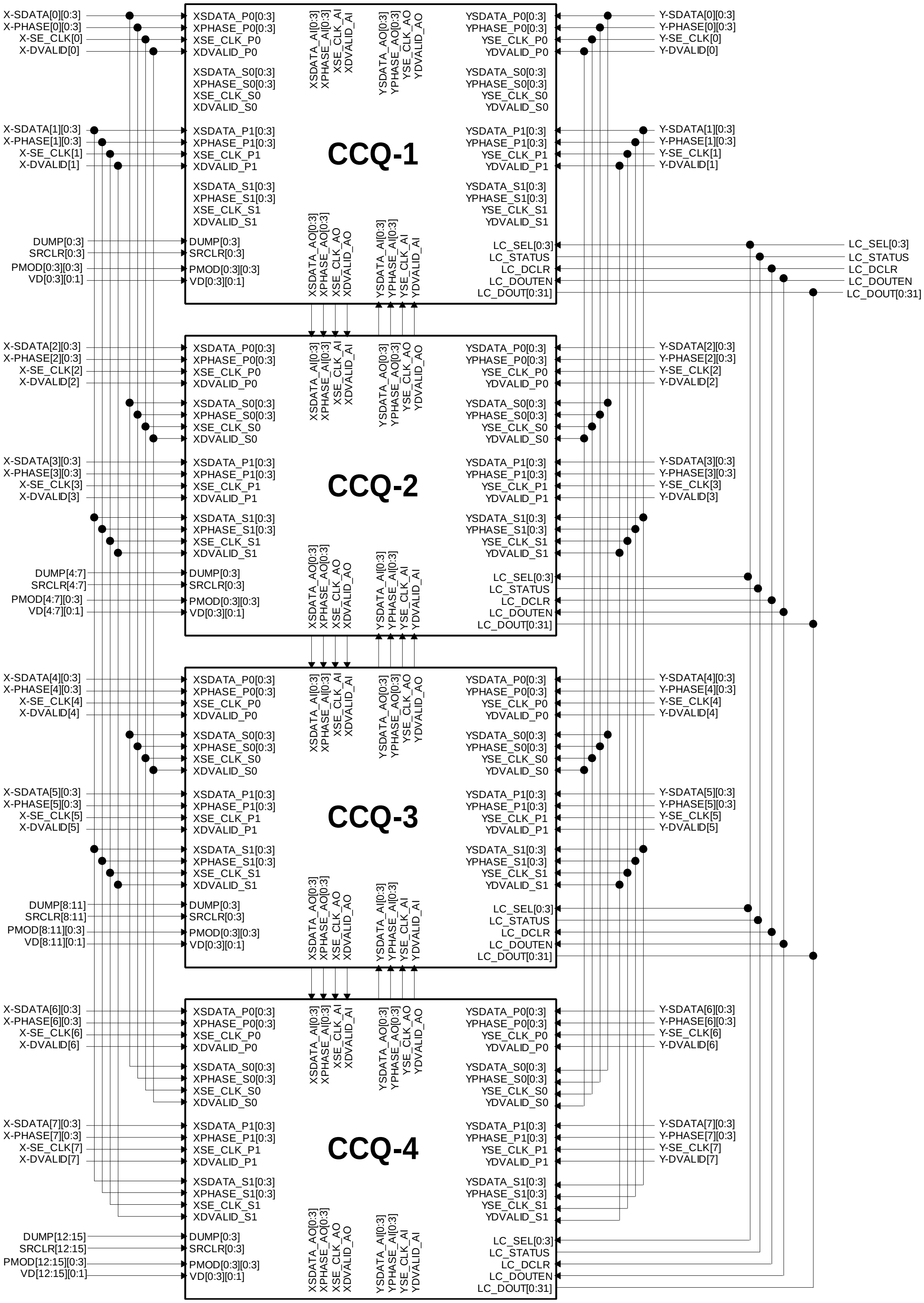


Figure 4-31 Detailed block diagram of the array of four CCQs. CCQ-1 is the master and CCQs 2-4 are the slaves in that they have access to the master's input data. Data flows between adjacent CCQs so that chaining of CCQs can occur.

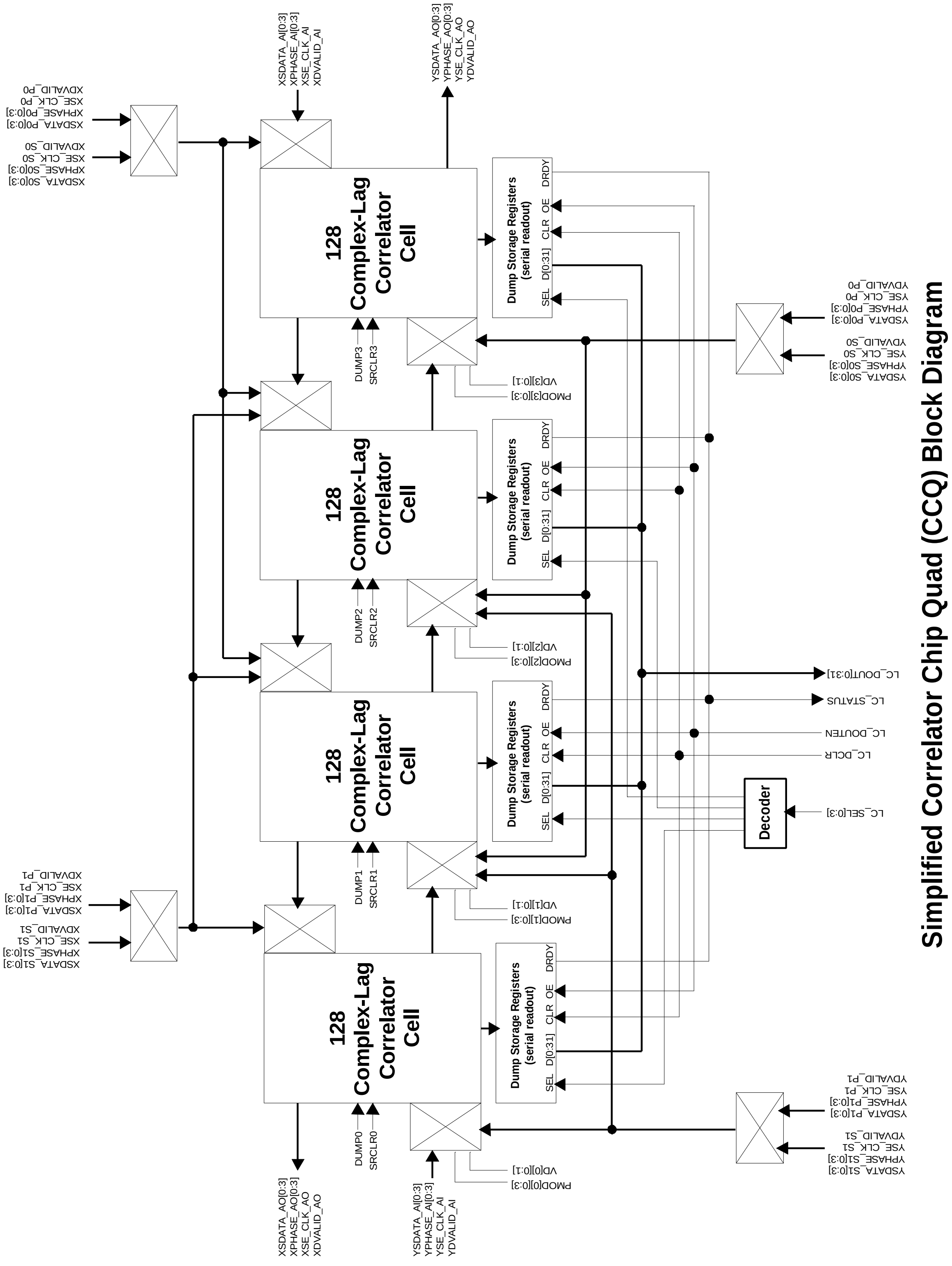


Figure 4-32 Simplified block diagram of one correlator chip CCQ. There are four, 128 complex-lag 'cells'—each cell has the lag architecture as shown in **Figure 4-23**. Switches in front of the X and Y inputs of each cell allow the cell to select new data, master data, or data from an adjacent cell.

The bulk of the silicon, although perhaps not the bulk of the power dissipation, is from the accumulators and the accumulator readout buffer registers. The correlator multipliers are (probably) 2's complement multipliers operating in the range of –8, -7, ... 0, +1, ..., +6, +7. This full range is required so that 7-bit correlation is possible [1]. The output of the multiplier will thus take values over the range –56...+64. After 5-level fringe rotation, the range is –112...+128. In order for the bulk of the accumulators to be ripple counters, a bias of +112 is added to this result before accumulation. Thus, for each sample, the range of outputs that must be accumulated are 0...+240. In the worst case, if +240 is added on every clock cycle, then in 1 millisecond of integration at 256 MHz, the output of the accumulator is 61.44 x 10^6—requiring a 26 bit accumulator. This accumulator is divided into an 8-bit "pre-accumulator", and an 18-bit ripple counter, both of which must be stored in on-chip buffers and read out of the correlator chip (i.e. the goal is to read out accumulators without any truncation).

In the nominal case where we are just integrating noise (assuming lag 0 autocorrelation), the RMS output of a 4-bit sampler is approximately 2.8, and the RMS output of the 5-level fringe stopper is approximately 1.38 (this is 1.955*0.707 [6]). Thus, the RMS output of a fringe stopper is about $2.8^2 * 1.38 \approx 30.3$. Adding a bias of +112, results in an output of +142.3 on every clock cycle. With a 1 millisecond integration time, at a 256 MHz clock rate, the output is 36.4288 x 10^6—requiring a total accumulator size of 25.11 or 26 bits. Clearly, the worst-case accumulator size is required to handle the nominal case.

In the best case (zero average correlation), the bias (+112) must be added to the accumulator on every clock cycle. At 256 MHz and a 1 millisecond integration time, the accumulator output is 28.672 x 10^6—requiring 24.8 or 25 bits. Obviously, using 26 bits to handle the worst case is not that much of an overhead. If the integration time is extended to 10 milliseconds, the accumulator must be 30 bits—a significant increase in silicon when there are 4096 accumulators (16384 accumulator bits, or 15% more silicon). It would seem that it is advantageous to have a maximum on-chip integration time of 1 millisecond—the same dump time required for recirculation with an affordable and available dual-port recirculation memory.

If 26-bit accumulators are not achievable, then the minimum case is as follows. The data takes on values in the range from –7...+7. The –8 state is not used and so 7-bit correlation would not be possible. The multiplier output range is –49...+49, and if a 3-level fringe rotator is used, the bias is +49. In the worst case, +98 must be added to the accumulator on every clock cycle. At 256 MHz and with a 1 millisecond integration time, the accumulator output is 25.088 x 10^6—requiring 24.5 or 25 bits. The best case here (with zero average correlation) requires a 24-bit accumulator. Clearly, this minimum case is not a significant reduction in accumulator size over the 26-bit worst-case requirement.

The correlator chip will need to be a large fully custom VLSI device. A preliminary design [0] at the University of Alberta VLSI design lab indicated that, *if low-power techniques are used,* the device will dissipate <2W at a clock rate of 256 MHz if fabricated in 0.18 μm CMOS running at ~1.6 V. The preliminary design indicated that the device would be about a 4 or 5 million-transistor device. This preliminary design did not include any additional synchronization, readout, or control logic.

4.3 LTA Controller

4.3.1 Black-Box Description and the Front Panel Data Port Interface

The design of the Baseline Board shown in Figure 4-1 includes one LTA Controller and associated LTA SDRAM (Synchronous Dynamic RAM) for each correlator chip. Although this seems like design overkill, the high-speed correlator chip readout requirements driven by recirculation, fast pulsar phase binning, and the need to minimize the size of the correlator chip accumulators makes this a virtual necessity. Because of the extreme requirements and the cost-sensitive nature of using one LTA Controller FPGA for each correlator chip, a detailed paper design of the LTA Controller was done. In this design, a clock rate of 128 MHz was used so that the cheapest speed grade FPGA and the cheaper[28] single data rate SDRAM could be used. The results of this design indicate that the LTA Controller should fit into an XCV100E-6FG256C with ~67% CLB utilization[29] and ~80% I/O utilization. The projected cost of this device for 2002 is $20.92. The 256 Mbit SDRAM—needed for two banks of 1000 phase bins each—is currently priced at $37. Thus, the total cost of the LTA Controller and SDRAM for one Baseline Board is about $3648—within the cost envelope for these functions estimated in the December, 2000 correlator budget.

Alternatively, it is possible to integrate the LTA Controller functions in the correlator chip. This would save the cost of the FPGA, but would reduce flexibility somewhat since the correlator chip would always have to interface to the same SDRAM. It is also quite risky since many of the functions the LTA Controller performs are fairly complex. Using an FPGA minimizes the design risk compared a hardwired device. Nevertheless, it may be worth considering integrating this function with the correlator chip since it could potentially save $300k in FPGA costs in a 40-station correlator system (almost enough to pay for the correlator chip NRE).

A black-box diagram of the LTA Controller and SDRAM is shown in Figure 4-33. There is a correlator chip interface, an MCB interface, the dedicated SDRAM interface, and finally an interface to an ‘FPDP Scheduler’ (Front Panel Data Port) and the ‘local’ FPDP bus itself (also refer to Figure 4-2 [BB slice diagram]).

[28] i.e. compared to DDR SDRAM.

[29] Post-correlation phase rotation (a.k.a. ‘harmonic phase suppression’) was not included in this design because of the cost and complexity of the phase rotator and the questionable performance benefit.

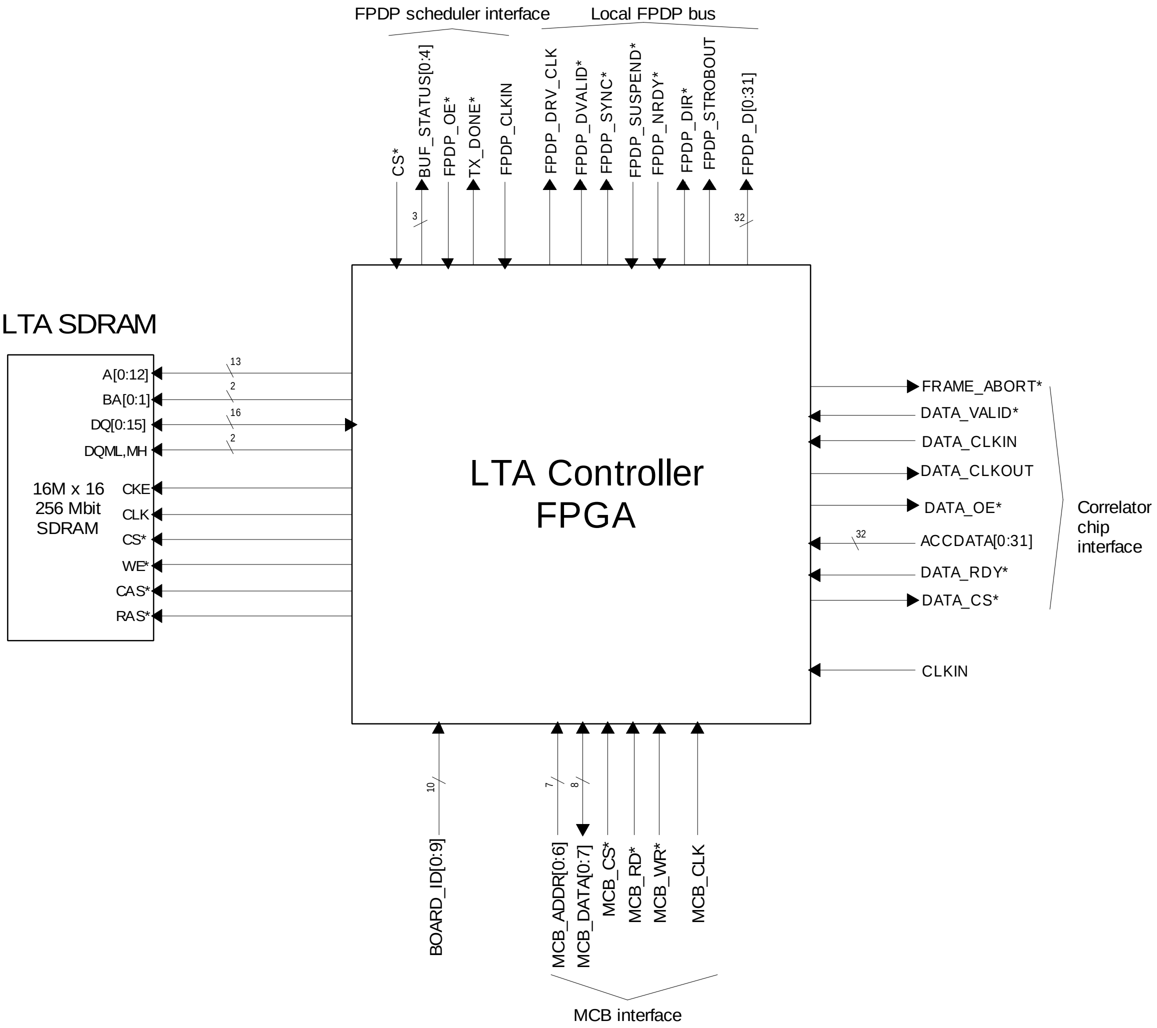


Figure 4-33 Black box diagram of the LTA Controller FPGA. Data enters the device via the correlator chip interface, is saved in the SDRAM, and when ready, exits the chip via the local FPDP bus. Transmission by a particular LTA Controller on the local FPDP bus is determined by signaling on the FPDP scheduler interface.

A high-speed data output pipeline is provided from each LTA Controller via a local FPDP interface that, with external drivers, drives an external FPDP. The FPDP is an ANSI/VITA standard (ANSI/VITA 17-1998). Many high-speed interfaces and schemes were considered, but the FPDP provided the highest-performance interface, with a simple protocol that does not require any processor interaction or special chip sets. Also, COTS FPDP PCI interface cards can be purchased so the back-end computers need not have any specially designed hardware. (The only disadvantage of the FPDP is that the cable length is limited to somewhere between 2 and 5 m depending on configuration.) Thus, the FPDP interface can be easily included in the LTA Controller. To provide scaleable high-performance output capability, it is possible to have multiple FPDP busses on every Baseline Board. This is shown in Figure 4-1 (main BB diagram) but is repeated in a simplified form in Figure 4-34 below.

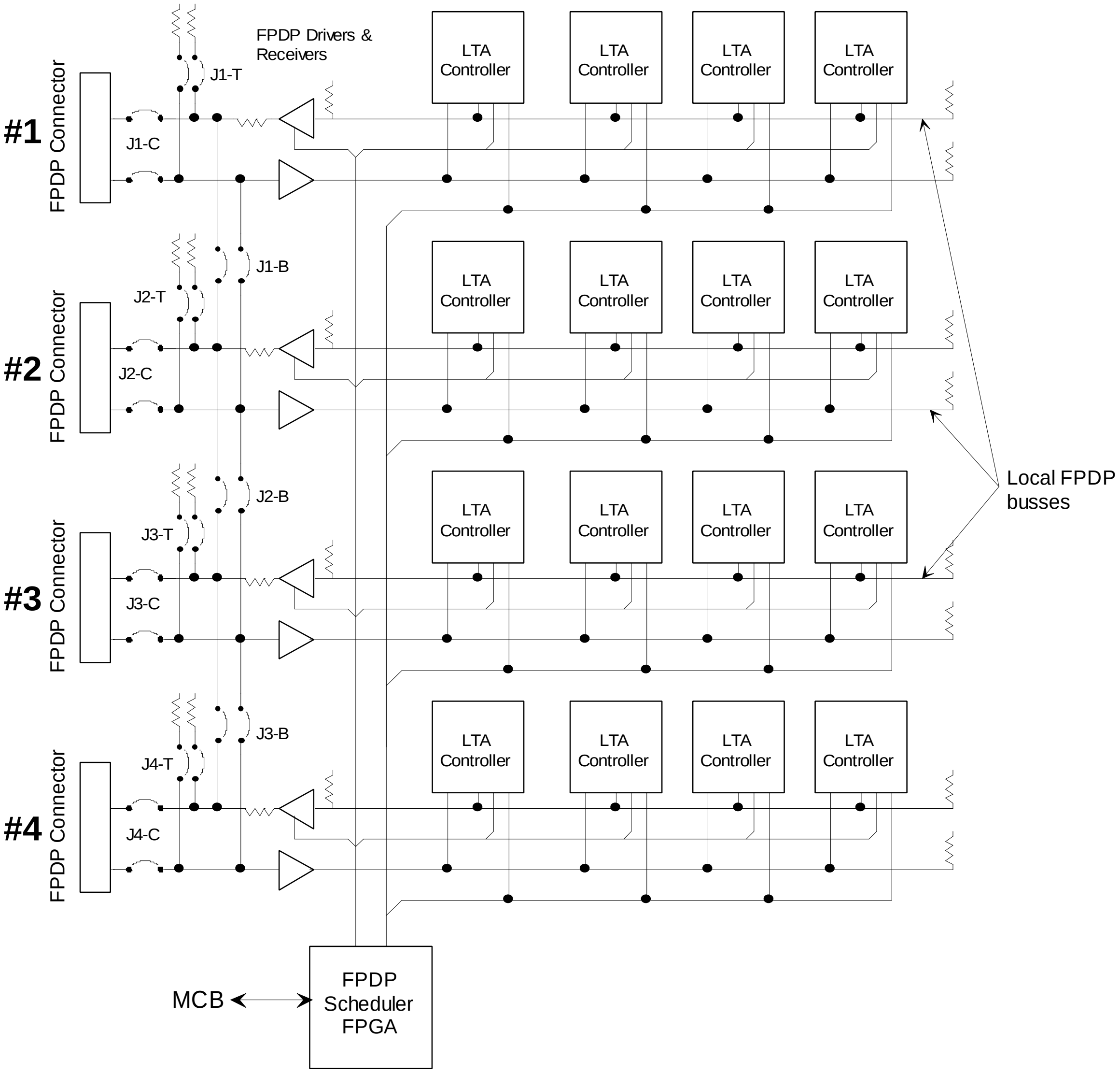


Figure 4-34 Simplified LTA Controller/FPDP interface diagram. Only 16 LTA Controllers are shown but in reality there will be 64. There are 4 local FPDP busses that can terminate on anywhere from 1 to 4 external FPDP interfaces depending on jumper settings. Each local bus has its own FPDP drivers.

Up to 4 external FPDP interfaces[30] are available on each Baseline Board—how many of these interfaces are actually used is entirely user-configurable. The FPDP specification allows for multiple TMs (Transmit Masters) connecting to a single FPDP via bussed ribbon cable as long as only one TM is active at a time and the stub lengths are within specification. In the configuration shown in Figure 4-34, the same (or better) electrical performance is obtained as long as the drivers are close together and stub lengths to unused FPDP connectors are short (hence the jumpers[31] to the connectors). This configuration also ensures that the local FPDP bus is short and that all stubs to the bus are

[30] There could be 8 or 16 but there is not enough front-panel space available (without some more cost and complexity).

[31] Because of the quantity, these jumpers will probably be PCB land pads. Zero-ohm resistors are installed to put in a jumper.

short. The relatively slow speed of the bus (25 nsec clock cycle) is also a good initial indication that the planned configuration should operate satisfactorily. The FPDP Scheduler FPGA controls which LTA Controller is transmitting on the bus(ses) at a given point in time (and controls which set of FPDP interface drivers are enabled for the given configuration and enabled LTA Controller). It does this by simply querying each LTA Controller to see if it has data to transmit and what its ‘urgency’ is, and then enabling the most urgent LTA Controller for transmission.

Referring to Figure 4-33, the FPDP Scheduler interface signal functionality is as follows.

- CS* – Asserting this line enables the arbiter interface I/O drivers. Otherwise output drivers are disabled, and the interface does not respond to arbiter signaling.

- BUF_STATUS[0:4] – These lines tell the FPDP Scheduler whether the LTA Controller has data to transmit and what the state of the LTA buffer memory is (i.e. how full it is). Bits 0-2 indicate relatively how full the LTA memory is. If this is approaching the all 1’s state, then the LTA is getting ready to overflow. Bit 3, if set, indicates that data is ready to be transmitted on the FPDP interface. Bit 4, if set, indicates that ‘speed dump’ data is ready for transmission. This is data that comes directly from the correlator chip, bypassing the LTA buffer (refer to Figure 4-5—DUMPTRIG format). BUF_STATUS bits can change at any time and so they should be double-sampled by the arbiter.

- FPDP_OE* – Asserting this input causes the LTA Controller to enable its FPDP output drivers and start data transmission on the FPDP. The FPDP Scheduler asserts this signal when it wants the CS* enabled LTA Controller to transmit data.

- TX_DONE* – The LTA Controller asserts this line for one clock cycle when it has finished an FPDP transmission.

- FPDP_CLKIN – This input clock is used to generate FPDP output clocks by the LTA Controller. This clock can be completely independent of the correlator chip clock.

A functional timing diagram for the LTA Controller’s FPDP scheduler interface and FPDP TM interface is shown in Figure 4-35. The FPDP TM timing is representative—all FPDP timing will be met once FPDP_OE* is asserted including FPDP_SUSPEND*, FPDP_NRDY*, and FPDP_DIR*. TX_DONE* is asserted for one clock edge only. If FPDP_OE* is not removed within the next 4 clock edges, the next frame of data will be transmitted. If there is no data to be transmitted, TX_DONE* will be asserted again until FPDP_OE* is negated or until the transmit buffer has data in it again. This way, multiple frames can be transmitted consecutively from one LTA Controller without having to query its status and suffer waiting for 16 FPDP clock cycles[32] every time. FPDP_DRV_CLK is used to drive the FPDP output drivers. FPDP_STROBOUT is used

[32] The specification requires that the FPDP strobe be stable for 16 cycles before data is transmitted.

for the actual FPDP STROB of PSTROB. This way, FPDP_DRV_CLK can be doubled, and FPDP_STRBOUT can be phase controlled for future FPDP-II performance.

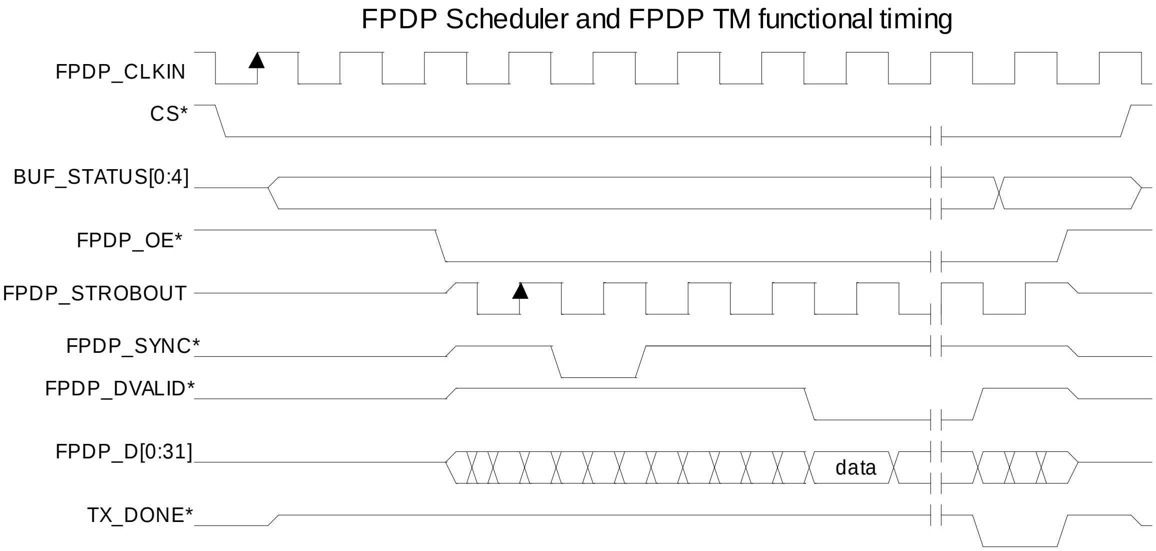


Figure 4-35 FPDP Scheduler and TM functional timing. The FPDP Scheduler enables data transmission from a particular LTA Controller by asserting the FPDP_OE* line. The LTA Controller indicates it has finished transmitting a frame by asserting the TX_DONE* line for one clock cycle.

4.3.1.1 Output Data Frame Formats

Two output data frame formats that are transmitted by the LTA Controller are defined in this design. There is an 'LTA dump' format, and a 'speed dump' format. The LTA dump format contains data that has been read from the LTA RAM. The speed dump format contains data that comes directly from the correlator chip—bypassing the LTA completely. The speed dump frame is provided for high performance configurations that are only limited by the number of correlator chips that are active on a board, the aggregate FPDP output bandwidth, and back-end computing performance[33]. LTA Controller output frame formats are shown in Figure 4-36 and Figure 4-37.

With data bias (DATA_BIAS) removed, in the lag 0 autocorrelation case (Gaussian noise, RMS=2.8 levels), a 32-bit 2's complement accumulator is able to integrate for ~1.024 seconds before overflow occurs. For maximum cross-correlation coefficients of, say 1%, the LTA can integrate for up to ~100 seconds before overflow. That is, the maximum LTA integration time in seconds[34] is approximately:

$$T \cong \frac{e^{21.483}}{r \cdot 2.8^2 \cdot 256 \cdot 10^6}$$

[33] One can imagine a special rack—say for phased-VLA autocorrelations—which has a few Baseline Boards, each with 4 FPDP interfaces connected to dedicated high-performance back-end computers. With speed dumping, 65536 phase bins are available.
[34] Ignoring a possible data valid count overflow.

Where r is the expected correlation coefficient.

LTA dump data frame (FPDP output)

SYNCH	0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1						W0
	DATA_BIN#	Reserv	ChipID	CCID	FType		W1
	BBID-Y	BBID-X	SBID-Y	SBID-X	SID-Y	SID-X	W2
	STATUS BITS	FRAME_COUNT	RECIRC_BLK-Y	RECIRC_BLK-X			W3
	DVCOUNT-Center						W4
	DVCOUNT-Edge						W5
	Reserved						W6
	TIMESTAMP-0						W7
	TIMESTAMP-1						W8
	Lag 0-In_phase accumulator						W9
	Lag 0-Quadrature accumulator						W10
	Lag 1-In_phase accumulator						W11
	Lag 1-Quadrature accumulator						W12
	⋮						W13-261
	Lag 127-In_phase accumulator						W262
	Lag 127-Quadrature accumulator						W263
SYNCH	0 1 1 1 0 0 0 1 1 1 0 0 0 1 1 1 0 0 0 1 1 1	Board ID					W264
	CHECKSUM						W265

W2: Identifier bits:

SID-X (b7-b0), **SID-Y** (b15-b8): 8-bit X and Y station ids. A station consists of input to 4 Station Boards.

SBID-X (b20-b16), **SBID-Y** (b25-b21): 5-bit X and Y sub-band ID (0...17). The data comes from this particular sub-band output of a Station Board.

BBID-X (b28-b26), **BBID-Y** (b31-b29); 3-bit X and Y baseband ID (0...7).

W4, W5:

DVCOUNT-Center: count of number of valid samples taken at the center of this lag chain.

DVCOUNT-Edge: count of number of valid samples taken at the edge of this lag chain.

W6:

Reserved for future use.

W7, W8:

Timestamp-0: number of 1 second ticks since last MAJOR EPOCH.

Timestamp-1:

MAJOR EPOCH (b31-b29):
000 = 00:00:00 Jan 1/1970
all others: reserved for future definition.

Sample Count (b28-b0): number of samples at CLOCK rate since last 1 second tick.

NOTE: this is the timestamp of the last frame added to the LTA before being dumped out.

W1:

FType: (b2-b0)
Frame type identifier. Always 100 (LTA dump).

CCID (b6-b3): Chip cross-correlator ID (0...15). Indicates which physical correlator in the chip this data comes from.

ChipID (b12-b7): Correlator chip ID (0...63). This identifies the physical correlator chip that the data came from.

DATA_BIN# (b31-b16): The LTA bin number that this data came from. b31 is the bank number. This can be different than the phase bin number, depending on how many recirculation blocks are in each bin (i.e. each recirculation block has its own bin).

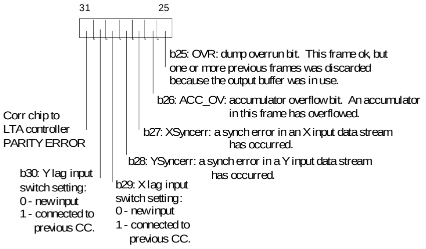
W3:

RECIRC_BLK-X (b7-b0): X-input recirculation block: the number of delay blocks inserted into the X data path.

RECIRC_BLK-Y (b15-b8): Y-input recirculation block: the number of dela blocks inserted into the Y data path.

FRAME_COUNT (b16-b24): A count of the number correlator chip data dumps have been integrated in the LTA. This count is modulo 512.

STATUS BITS (b31-b25):


31
25
b25: OVR: dump overrun bit. This frame ok, but one or more previous frames was discarded because the output buffer was in use.
b26: ACC_OV: accumulator overflow bit. An accumulator in this frame has overflowed.
Corr chip to LTA controller PARITY ERROR
b27: XSyncerr: a synch error in an X input data stream has occurred.
b28: YSyncerr: a synch error in a Y input data stream has occurred.
b30: Y lag input switch setting: 0 - new input 1 - connected to previous CC.
b29: X lag input switch setting: 0 - new input 1 - connected to previous CC.

W9-W263:

Lag 0-127 In-phase and Quadrature accumulator data. This data has already had DATA_BIAS removed.

W0, W264:

Start and end SYNCH words. W264 b0-b9 is a 10-bit board ID--it is use to identify the physical Baseline Board the data came from.

W265:

CHECKSUM. Checksum of W0 thru W264 (inclusive) modulo 2^32.

Figure 4-36 LTA dump data frame transmitted from the LTA Controller. This is data read from the LTA RAM after one or more dumps from the correlator chip have been integrated. The DATA_BIN# is the actual physical bin (of two banks of 1000 each) that the data comes from. This can be different than the 'phase bin' specified by DUMPTRIG if recirculation is active. The phase bin number is a simple post-correlation calculation involving DATA_BIN# and the number of recirculation blocks active. Note that the integrated DATA_BIAS is not present in the frame because this would quickly overflow the 32-bit limit.

SPEED dump data frame (FPDP output)

28 24 20 16 12 8 4 0

SYNCH	0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1							W0
	STATUS BITS	Reserv.	ChipID	CCID	HSP-Y	HSP-X	FType	W1
	BBID-Y	BBID-X	SBID-Y	SBID-X	SID-Y	SID-X		W2
	Phase BIN#	RECIRC_BLK-Y	RECIRC_BLK-X					W3
	DVCOUNT-Center							W4
	DVCOUNT-Edge							W5
	DATA_BIAS							W6
	TIMESTAMP-0							W7
	TIMESTAMP-1							W8
	Lag 0-In_phase accumulator							W9
	Lag 0-Quadrature accumulator							W10
	Lag 1-In_phase accumulator							W11
	Lag 1-Quadrature accumulator							W12
	⋮							W13-261
	Lag 127-In_phase accumulator							W262
	Lag 127-Quadrature accumulator							W263
SYNCH	0 1 1 1 0 0 0 1 1 1 0 0 0 1 1 1 0 0 0 1 1 1	Board ID						W264
	CHECKSUM							W265

W2: Identifier bits:

SID-X (b7-b0), **SID-Y** (b15-b8): 8-bit X and Y station ids. A station consists of input to 4 Station Boards.
SBID-X (b20-b16), **SBID-Y** (b25-b21): 5-bit X and Y sub-band ID (0...17). The data comes from this particular sub-band output of a Station Board.
BBID-X (b28-b26), **BBID-Y** (b31-b29); 3-bit X and Y baseband ID (0...7).

W3:

RECIRC_BLK-X (b7-b0): X-input recirculation block: the number of delay blocks inserted into the X data path.
RECIRC_BLK-Y (b15-b8): Y-input recirculation block: the number of delay blocks inserted into the Y data path.
Phase BIN (b31-b16): The phase bin that this data is labelled with. b31 indicates bank 0 or bank 1.

W4, W5:

DVCOUNT-Center: count of number of valid samples taken at the center of this lag chain.
DVCOUNT-Edge: count of number of valid samples taken at the edge of this lag chain.

W6:

DATA_BIAS: exact DC bias that was in accumulator data and that has already been removed from the data.

W7, W8:

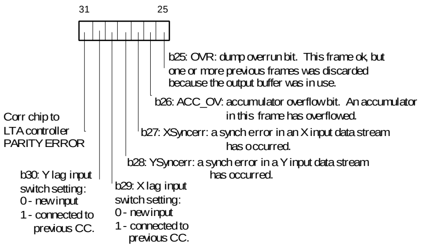
Timestamp-0: number of 1 second ticks since last MAJOR EPOCH.
Timestamp-1:
MAJOR EPOCH (b31-b29):
000 = 00:00:00 Jan 1/1970
all others: reserved for future definition.
Sample Count (b28-b0): number of samples at CLOCK rate since last 1 second tick.

W1:

FType: (b2-b0)
Frame type identifier. Always 011 (SPEED dump).

HSP-X (b6-b3): Harmonic suppression phase-X. X-phase has been offset by this quantity.
HSB-Y (b10-b7): Harmonic suppression phase-Y. Y-phase has been offset by this quantity.
CCID (b14-b11): Chip cross-correlator ID (0...15). Indicates which physical correlator in the chip this data comes from.
ChipID (b20-b15): Correlator chip ID (0...63). This identifies the physical correlator chip that the data came from.

STATUS BITS (b31-b25):


31
25
b25: OVR: dump overrun bit. This frame ok, but one or more previous frames was discarded because the output buffer was in use.
b26: ACC_OV: accumulator overflow bit. An accumulator in this frame has overflowed.
Corr chip to LTA controller PARITY ERROR
b27: XSyncerr: a synch error in an X input data stream has occurred.
b28: YSyncerr: a synch error in a Y input data stream has occurred.
b30: Y lag input switch setting: 0 - new input 1 - connected to previous CC.
b29: X lag input switch setting: 0 - new input 1 - connected to previous CC.

W9-W263:

Lag 0-127 In-phase and Quadrature accumulator data. This data has already had DATA_BIAS removed.

W0, W264:

Start and end SYNCH words. W264 b0-b9 is a 10-bit board ID--it is use to identify the physical Baseline Board the data came from.

W265:

CHECKSUM. Checksum of W0 thru W264 (inclusive) modulo 2^32.

Figure 4-37 Speed dump data frame transmitted from the LTA Controller. This is data that has by-passed the LTA RAM and essentially comes straight from the correlator chip. This frame is very similar to the correlator chip output data frame of **Figure 4-30**. Note that the LTA Controller includes the DATA_BIAS in this frame even though it has already been removed from the data.

4.3.2 Detailed LTA Controller Functional Description

Figure 4-38 is a detailed functional block diagram of the LTA Controller FPGA. The main controller block is the LTA Integration Controller. It takes and integrates fresh correlator chip data sitting in the Correlator Chip Dual-Port Frame Buffer with existing data in the SDRAM. The result is then written back to the SDRAM. Once an SDRAM bin is ready for output, it sets a bit in the LTA Semaphore Table. The LTA Frame Output Controller parses the LTA Semaphore Table looking for ready LTA bins. When a ready LTA bin is found, it arbitrates for the SDRAM, reads out the data, stuffs it into one of Output Buffers A or B, and sets an appropriate output buffer semaphore. The FPDP Output Controller waits for output buffer semaphores to be set and then, when enabled by the Baseline Board's FPDP Scheduler, transmits the data on the local FPDP bus (that also drives an external FPDP interface). For speed dumping, the LTA Frame Output

Controller gets notified directly by the Correlator Chip Readout Controller that data is ready. This prompts the LTA Frame Output Controller to stuff the data directly into one of the output buffers—bypassing the LTA SDRAM. In the figure, bold lines are data flows and thin lines are control lines.

The LTA Integration Controller is designed so that if all phase bins (that are active) for a particular CCID are full it discards data from the correlator chip for that CCID until all of those phase bins are empty. This way it is possible to burst dump 2000 phase bins, not lose any data, and then suffer some dead time while the buffers are being emptied. Also, correlator chip data is discarded if the phase bin it is destined for contains data ready for readout.

Previously, it was mentioned that a detailed paper design of the LTA Controller was done so that an accurate logic device estimate could be made. The top-level schematic diagram from this paper design is shown in Figure 4-39. This design, if a 128 MHz correlator chip readout clock is used with a 100 Mbytes/sec FPDP, should meet the following performance specifications:

- If all correlator chip data is dumped with reasonably long phase-bin integration times so the SDRAM is not accessed very often to output the data, the phase bins can be as narrow as ~180 µsec.
- If only one 'lag cell' (known by the LTA Controller as a 'CCID') is dumped, the phase bins can be as narrow as ~12 µsec (with a reasonably long phase-bin integration time).
- If the LTA just integrates one frame and the entire correlator chip is dumped, the integration time is ~320 µsec (assuming no FPDP bottleneck). That is, the real-time sustained dump rate is 320 µsec.
- The entire Baseline Board can be dumped every ~10.5 msec (each dump is, of course, several correlator chip dumps and LTA integrations).
- If speed dumping is used with a dedicated FPDP (i.e. only one correlator chip active per FPDP and with a ~200 Mbytes/sec FPDP), one CCID frame can be output every 5 µsec. With this mode, up to 65,536 pulsar phase bins can be defined. <u>This is the extreme performance case.</u>

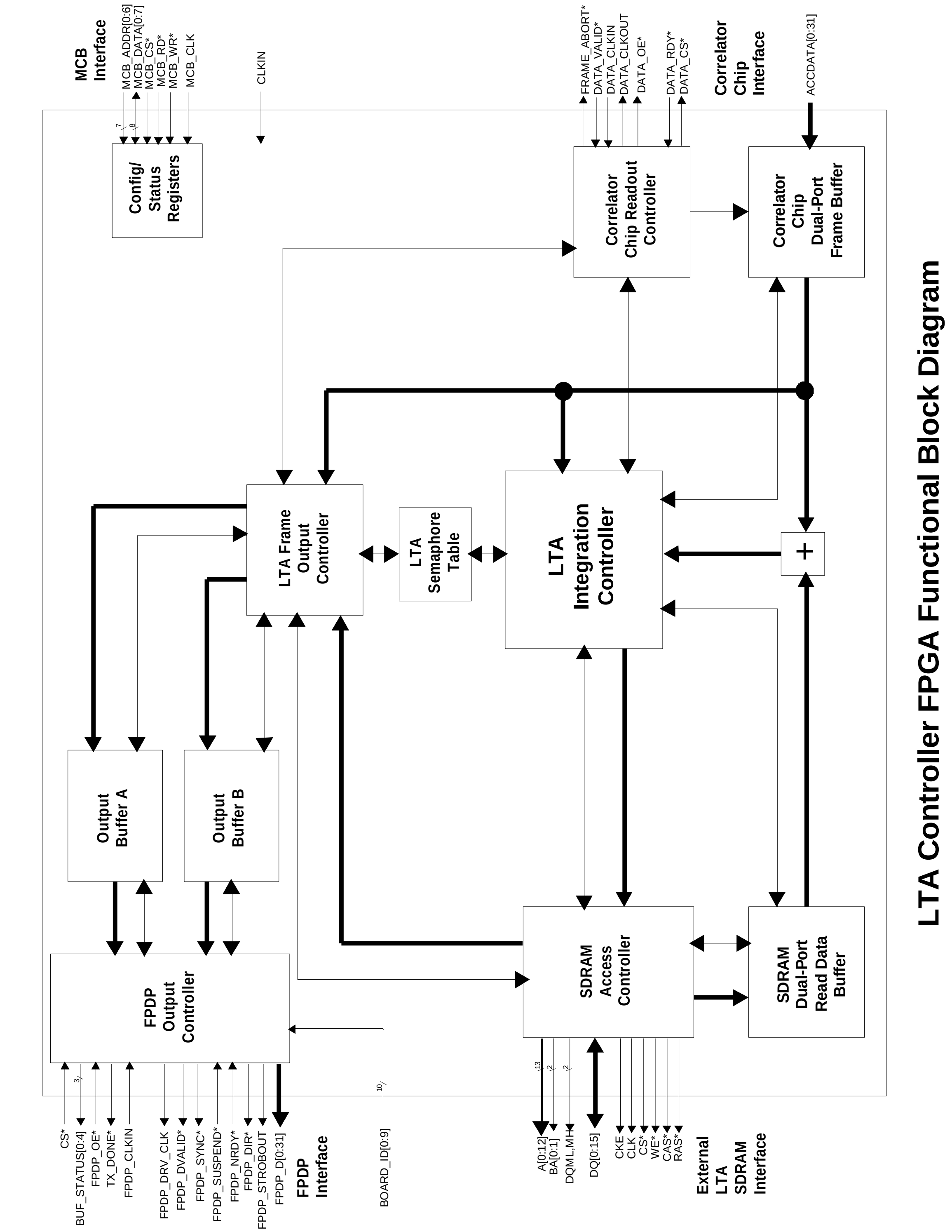


Figure 4-38 LTA Controller functional block diagram. Data enters into a frame buffer from the correlator chip. If required, at the same time, data is read from SDRAM into the SDRAM read buffer. These data are integrated and written back to SDRAM. When an SDRAM data bin is ready for output, an associated semaphore is set in the LTA Semaphore Table. The frame output controller looks at the semaphore table for ready data and then transfers ready data from the SDRAM into output buffer A or B, where it is eventually transmitted onto the FPDP interface.

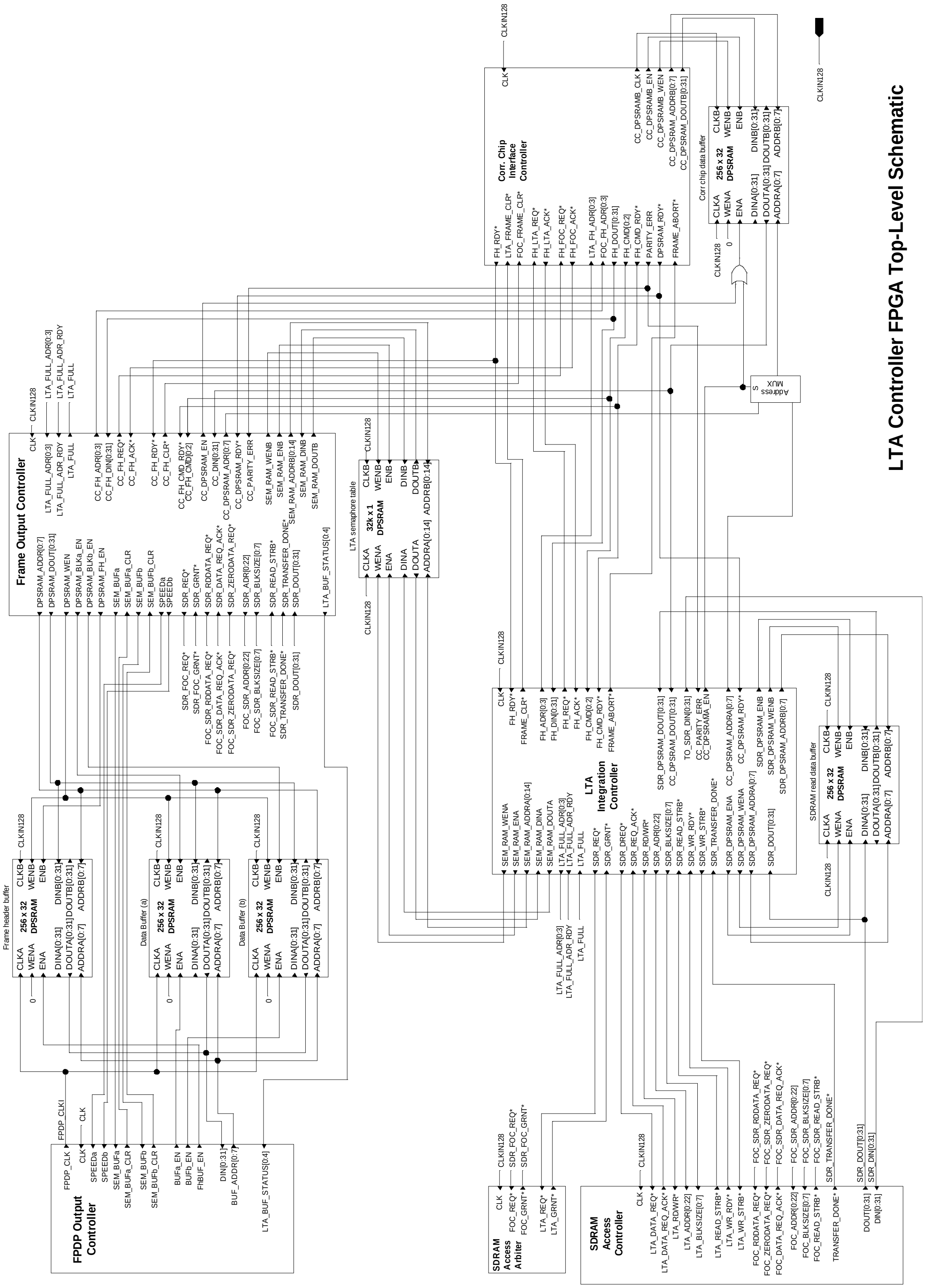
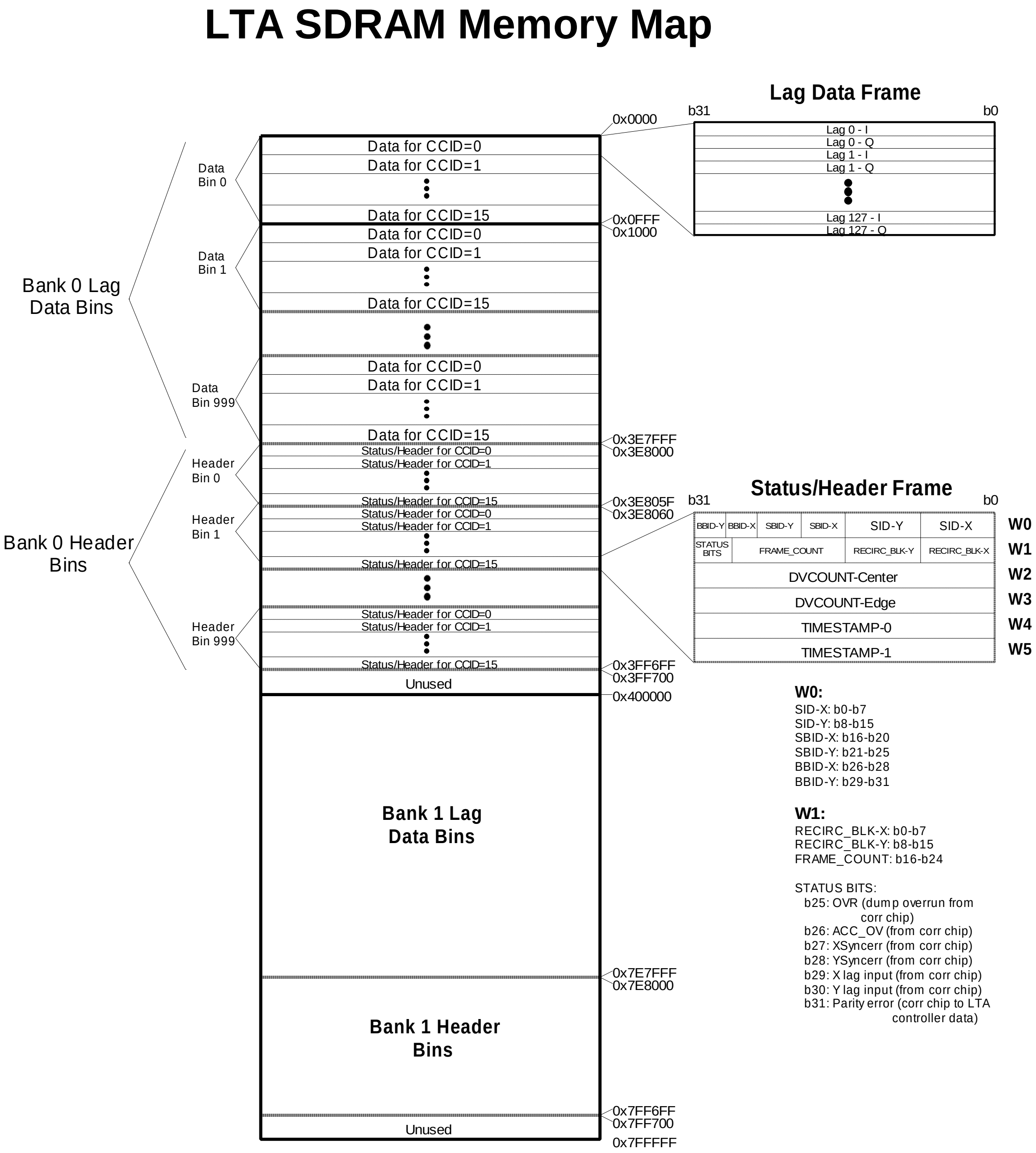


Figure 4-39 LTA Controller paper design top-level schematic. There are 31 schematic sheets in the design and the design entry time was 6 person-weeks. This design should comfortably fit in a $20 FPGA (XCV100E-6FG256C).

As part of the detailed paper design, an LTA RAM memory map was developed to ensure that the performance specifications could be met with a not-too-expensive SDRAM memory device. This memory map is shown in Figure 4-40 for a 256 Mbit RAM—arranged in the figure as 8 M x 32 bits when it is actually a 16 M x 16-bit device . *It is important to note that each CCID has its own memory locations—it is not possible for one CCID to use memory that is not currently being used by another CCID.*



W0:
SID-X: b0-b7
SID-Y: b8-b15
SBID-X: b16-b20
SBID-Y: b21-b25
BBID-X: b26-b28
BBID-Y: b29-b31

W1:
RECIRC_BLK-X: b0-b7
RECIRC_BLK-Y: b8-b15
FRAME_COUNT: b16-b24

STATUS BITS:
b25: OVR (dump overrun from corr chip)
b26: ACC_OV (from corr chip)
b27: XSyncerr (from corr chip)
b28: YSyncerr (from corr chip)
b29: X lag input (from corr chip)
b30: Y lag input (from corr chip)
b31: Parity error (corr chip to LTA controller data)

Figure 4-40 LTA SDRAM memory map. There are two banks of (exactly) 1000 phase bins (or more correctly, data bins) each. A lag data frame and a status/header data frame and their contents are defined. A CCID is the same as a correlator chip 'lag cell'—a single 128 complex-lag correlator block. Note that the CCID is not stored in the LTA memory since the CCID is a function of the LTA memory address and therefore does not have to be stored.

The following Figure 4-41 contains specific LTA memory addressing information that matches the memory map of the previous figure.

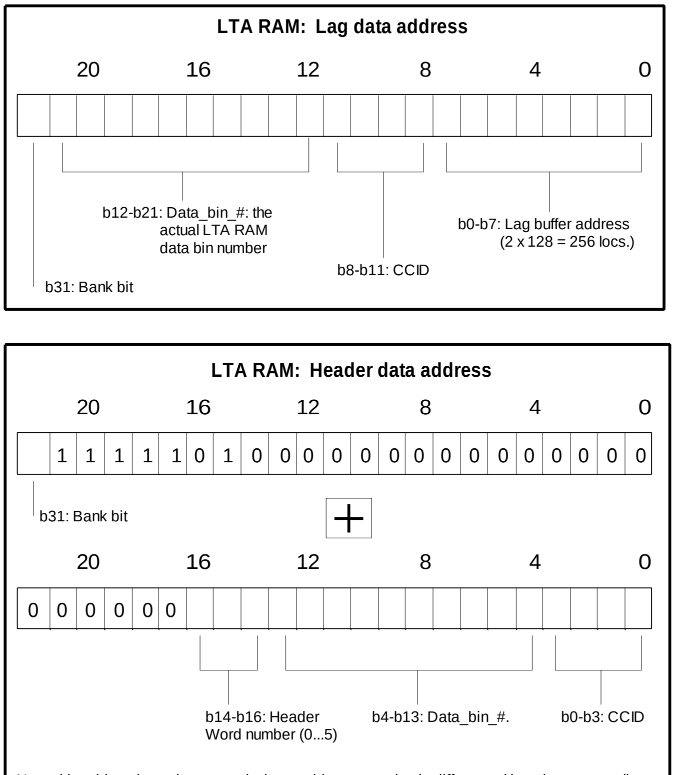
LTA Memory Access Equations & Addressing

LTA RAM data bin number calculation

Data_bin_# = Phase_bin x Nblocks + (RECIRC_BLK_Y - Start_Blk_Y)

where:

- -- **Data_bin_#** is the actual LTA RAM data bin number.
- -- **Phase_bin** is the phase bin indicator coming from the correlator chip (ignoring the 'Bank buffer' bit).
- -- **Nblocks** is the number of lag blocks being synthesized. This is set in a register in the Recirculation Controller and the LTA Controller FPGAs. Without any recirculation, this is 1. Data from each CCID must be enabled to use Nblocks or 1 (depending on whether recirculation is active for the CCID or not).
- -- **RECIRC_BLK_Y**is the 8-bit recirculation block number coming from the correlator chip.
- -- **Start_Blk_Y** is the Y start lag block the recirculation controller is handling. This is set in a register in the Recirculation Controller and the LTA Controller FPGAs.


LTA RAM: Lag data address
20
16
12
8
4
0
b12-b21: Data_bin_#: the actual LTA RAM data bin number
b0-b7: Lag buffer address (2 x 128 = 256 locs.)
b8-b11: CCID
b31: Bank bit
LTA RAM: Header data address
20
16
12
8
4
0
1 1 1 1 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
b31: Bank bit
+
20
16
12
8
4
0
0 0 0 0 0 0
b14-b16: Header Word number (0...5)
b4-b13: Data_bin_#.
b0-b3: CCID

Note: this addressing scheme results in an address map that is different, although conceptually the same, as that shown in the LTA RAM Memory Map. (But it does require accessing data one word at a time rather than in blocks.)

The final address is formed by the addition of the base address and the address formed with the CCID, the Data_bin_#, and the Header word number (W0-W5).

Figure 4-41 Specific LTA memory addressing information. The top box is the equation used to calculate the actual data bin number that a particular phase bin maps into when recirculation is active. The middle box is the lag data address breakdown. The bottom box is the header data address breakdown. Note that, for logic simplicity, the header data is not contiguous in memory as conceptually shown in the LTA SDRAM Memory Map. However, the memory can be thought of as being contiguous as long as burst accesses are not performed.

4.4 FPDP Scheduler FPGA

The FPDP Scheduler FPGA is responsible for scheduling the output of data from LTA Controllers. It does this by querying each LTA Controller's BUF_STATUS bits to determine which controller should be enabled to transmit data on the FPDP interface(s). The scheduler prioritizes LTA Controllers for data transmission based on how full LTA memory is, whether a 'speed dump' data frame is ready, and whether an output buffer has data ready for transmission. The scheduler also decides how many frames an LTA Controller can transmit before it switches to another device. A black-box diagram of the FPDP Scheduler FPGA is shown in Figure 4-42.

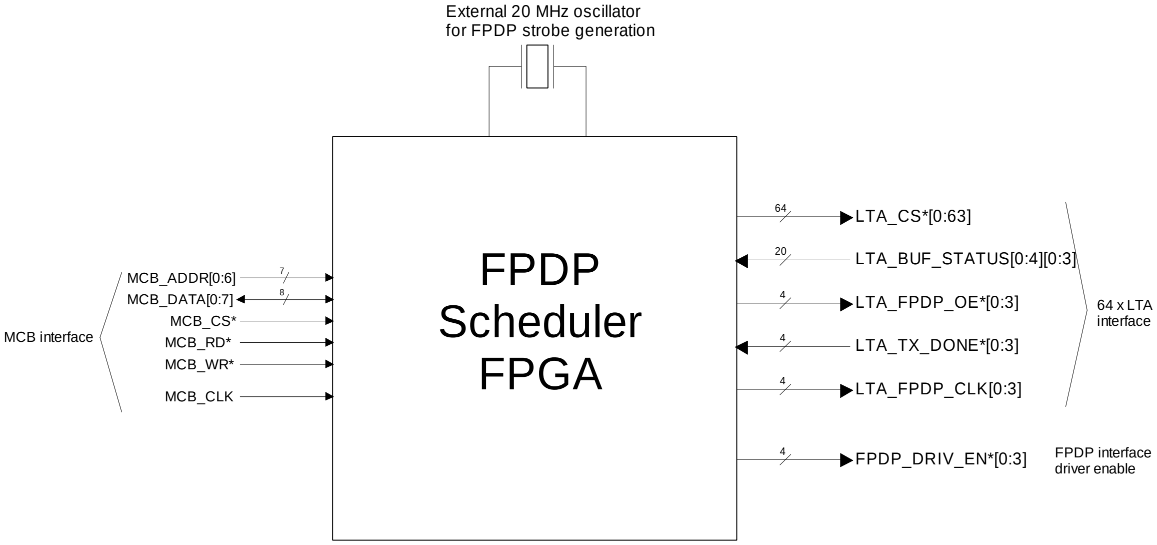


Figure 4-42 FPDP Scheduler FPGA black-box diagram. This device queries each LTA Controller to determine its priority for data transmission on the FPDP. When the highest-priority LTA Controller has been identified, it and its associated FPDP interface drivers are enabled.

4.5 Baseline Board Physical Layout

A baseline plan for the physical layout of the Baseline Board has been developed. This board is a 12U x 400 mm board (~18" x 15.7"). The front-side layout of the board is shown in Figure 4-43. Each correlator chip is allocated a 1" x 1" footprint. The rear-side layout is shown in Figure 4-44. Not shown are ejector handles and a stiffener plate that will be required to handle the board's large insertion/extraction force resulting from the large number of connector pins required to get all of the data into the board. In the figure, the 8x8 correlator chip array and associated Recirculation Controllers and LTA Controllers have been rotated by 45^{o} so that there is better delay-path matching for the X and Y data inputs than would otherwise be obtained without the rotation[35]. This also spreads the heat load over a wider cross-section of airflow (airflow is vertical in the orientation shown), although heating of correlator chips at the top center may be worse than without the rotation because of the longer vertical column of correlator chip area.

[35] Recall that any X/Y delay-path mismatch must be absorbed in the correlator chip.

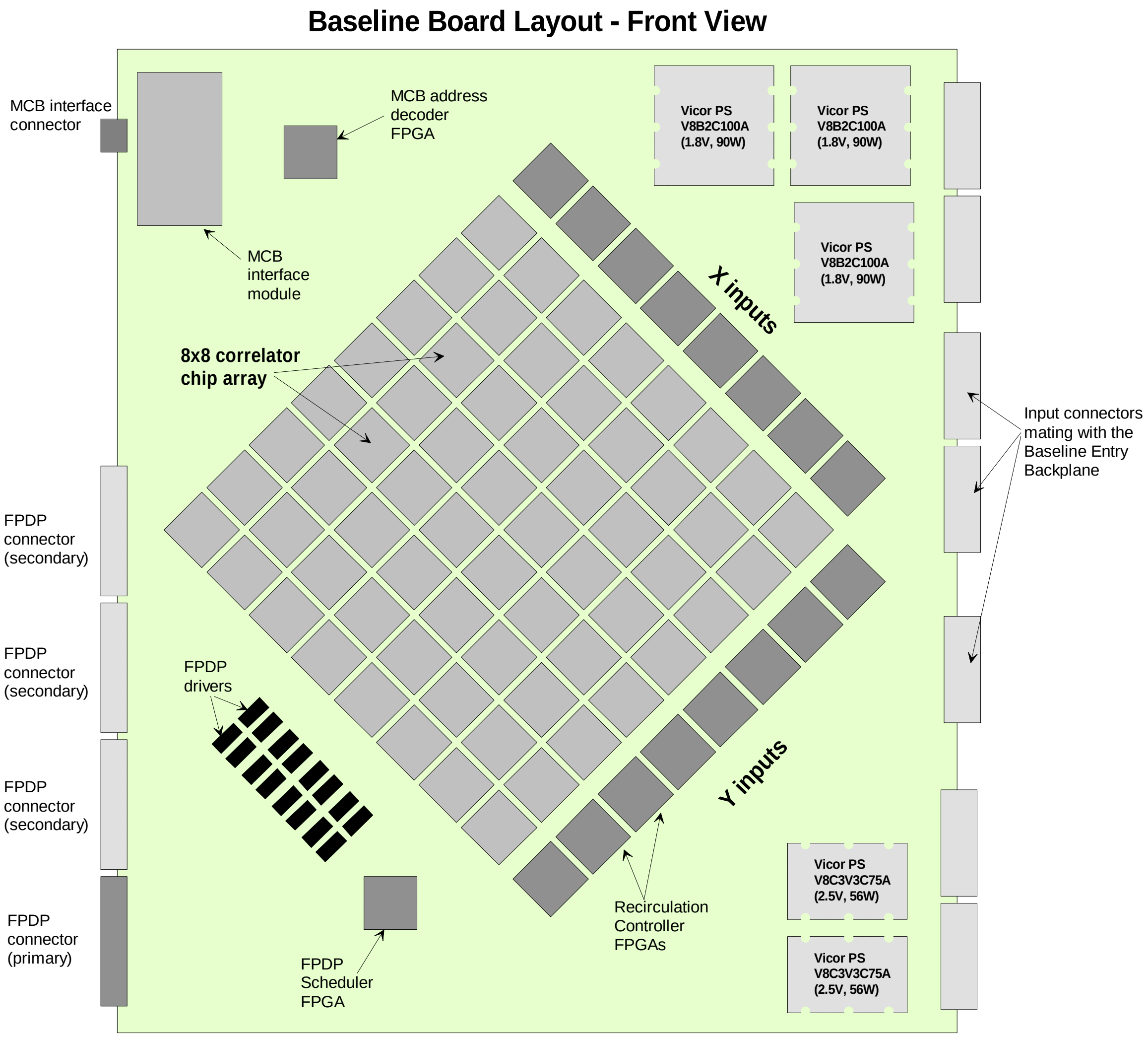

Baseline Board Layout - Front View
MCB interface connector
MCB interface module
MCB address decoder FPGA
Vicor PS V8B2C100A (1.8V, 90W)
Vicor PS V8B2C100A (1.8V, 90W)
Vicor PS V8B2C100A (1.8V, 90W)
X inputs
8x8 correlator chip array
Input connectors mating with the Baseline Entry Backplane
FPDP connector (secondary)
FPDP connector (secondary)
FPDP connector (secondary)
FPDP drivers
Y inputs
Recirculation Controller FPGAs
Vicor PS V8C3V3C75A (2.5V, 56W)
Vicor PS V8C3V3C75A (2.5V, 56W)
FPDP connector (primary)
FPDP Scheduler FPGA

Figure 4-43 Baseline plan for the physical layout of the Baseline Board. The 8x8 correlator chip array and associated circuitry is rotated by 45^{o} for better X and Y signal delay matching than would be obtained without the rotation.

The seven input connectors (on the right hand side of Figure 4-43) are proposed to be 200-pin 'Type E hardmetric 2.0 mm' connectors. The manufacturer (ept) supplies these connectors with the capability of having staggered pin heights for hot swapping. It is proposed that staggered pin heights be used to reduce the overall insertion/extraction force. The insertion force per pin is 0.75 N and the extraction force is only 0.15 N. The insertion force is the force required to spread the socket fingers, whereas the extraction force is the force necessary to overcome the coefficient of static friction after the fingers have been spread. If pin heights are staggered, then it should be possible to reduce the total insertion force from 7x200x0.75N=1.05 kN (241 lbs) to 435 N (100 lbs). A simple numerical analysis indicated that this could be done if 81 long pins, 66 medium pins, and 53 short pins per connector are used. Although this scheme is not formally in the manufacturer's data, the company's engineering representative said that this seems like a logical possibility.

Baseline Board Layout - Rear View

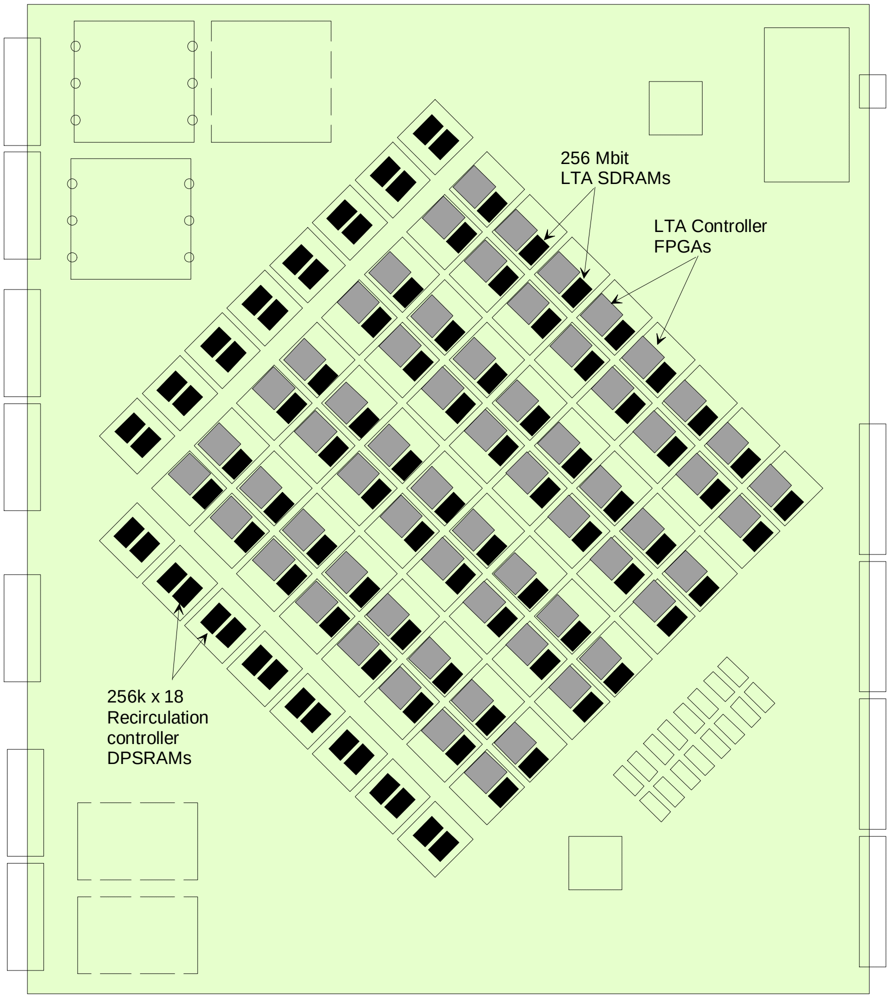


Figure 4-44 Baseline Board rear-side physical layout. Behind each correlator chip is its associated LTA Controller and SDRAM. Behind each Recirculation Controller are its associated DPSRAMs.

Baseline Board Layout - Front View - High Speed Data Routing

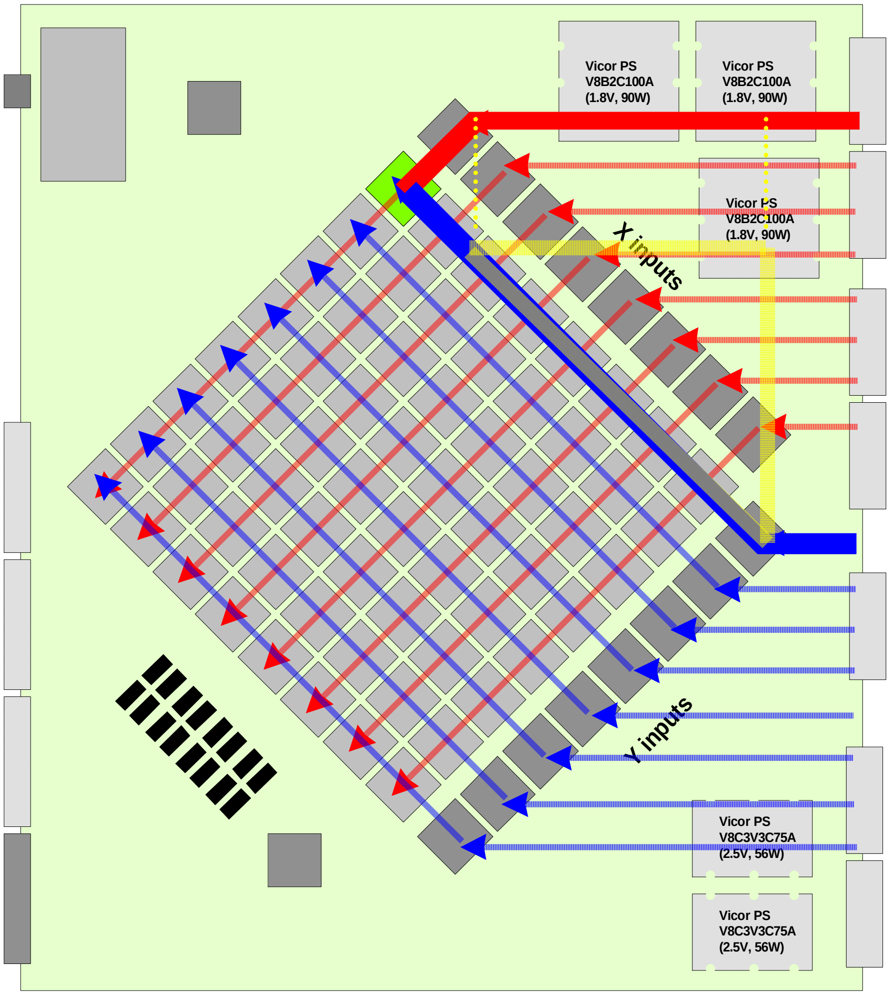


Figure 4-45 Baseline Board high speed data routing paths. The paths in red are X-station data and the paths in blue are Y-station data. This orientation has better delay-path matching than a vertical orientation of the correlator chip array, however there is still path mismatch. The worst case mismatch is shown to a correlator chip (in green). The mismatch is the difference between the hypotenuse and the side of the 45-45-90 triangle (yellow) shown.

The X/Y delay-path mismatch shown in Figure 4-45 can be reduced by modifying data routing to insert more delay in the longer paths from the connectors to the Recirculation Controllers. This routing is shown in Figure 4-46.

Baseline Board Layout - Front View - High Speed Data Routing - Better

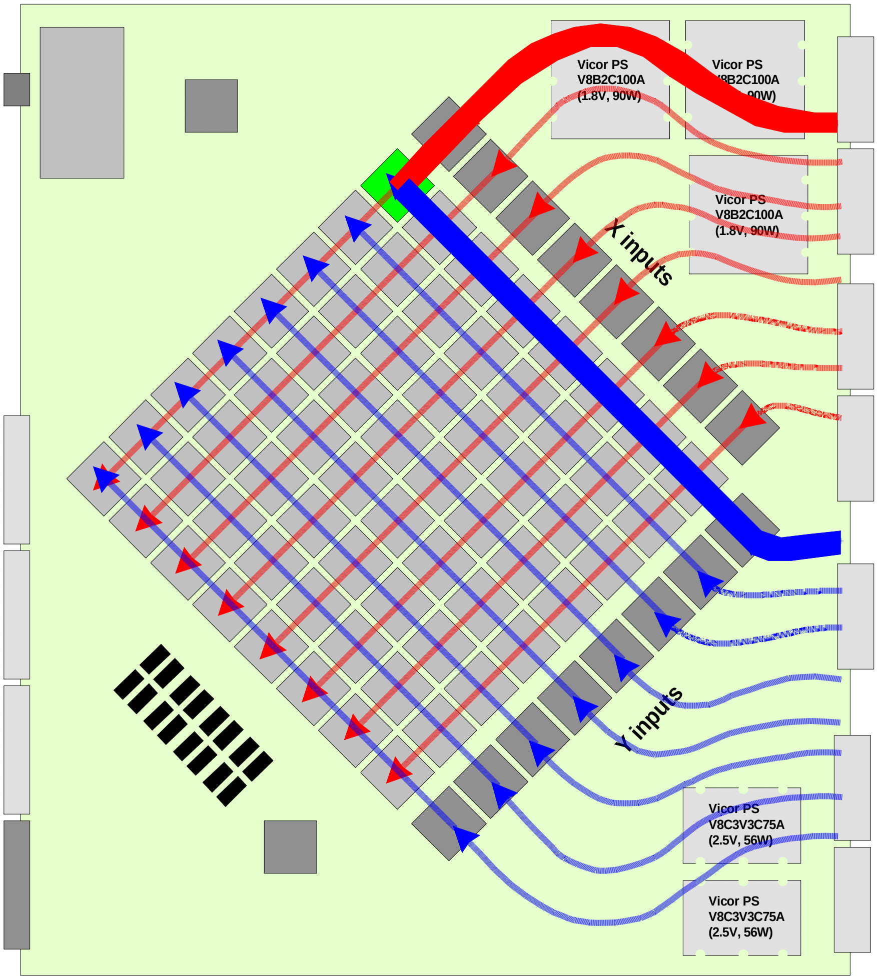


Figure 4-46 Baseline Board high-speed data routing for better X/Y delay-path matching. The longer the data path is from the connectors to the Recirculation Controllers, the more additional routing delay is added. This routing ensures that X and Y data arrives at the correlator chips at about the same time—reducing the delay-path mismatch circuitry requirements in the correlator chip.

Local FPDP bus routing with a *single* external FPDP interface output is shown in Figure 4-47. There are 4 independent local FPDP busses, each one terminating in its own set of FPDP drivers. Since only one FPDP output is used, the FPDP Scheduler must only enable one correlator chip and one set of FPDP drivers at a time. The FPDP drivers are bunched together and the unused FPDP connectors are disconnected (with jumpers as indicated in Figure 4-34) close to the drivers so that stub lengths are kept as short as possible.

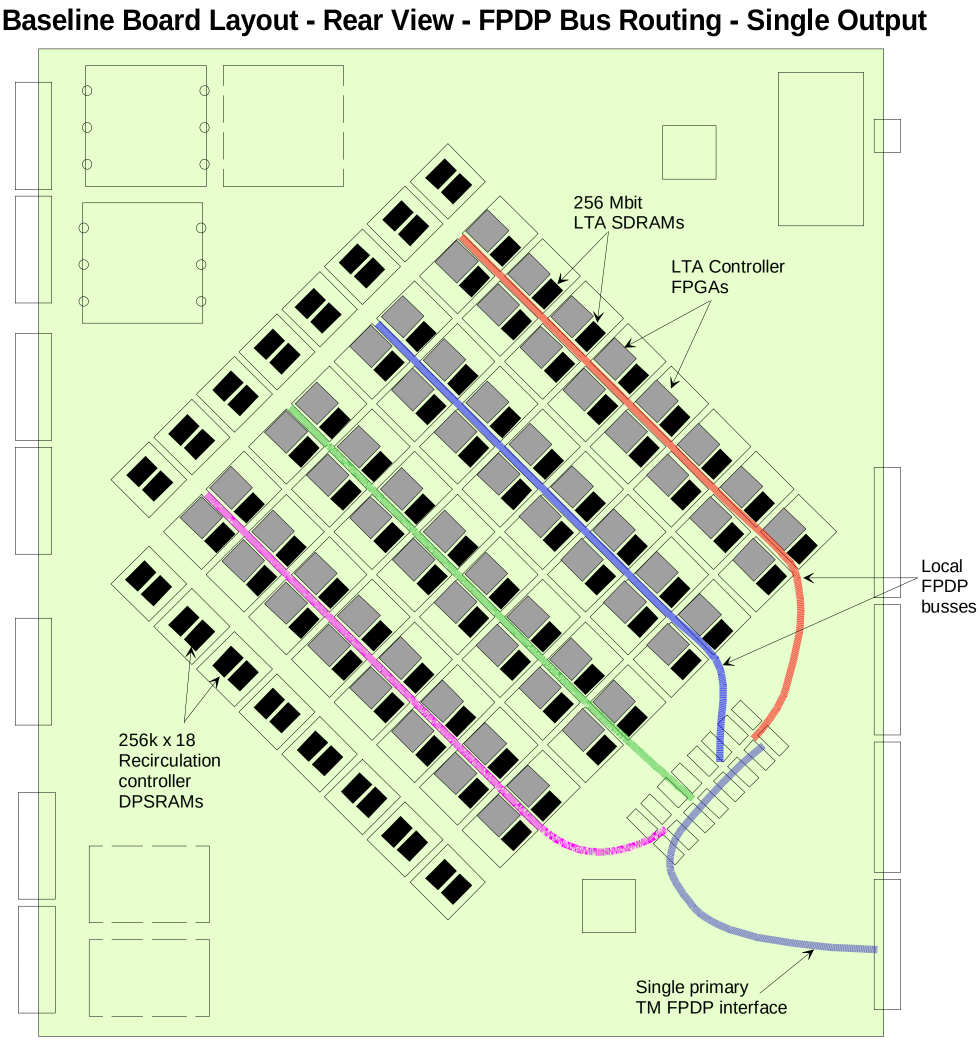


Figure 4-47 Local FPDP bus routing with one active external FPDP transmit interface. The FPDP Scheduler enables only one correlator chip and one set of FPDP drivers at a time. With this arrangement and a 100 Mbytes/sec FPDP interface, all lags from all correlator chips can be dumped about every 11 milliseconds. This should be more than adequate for most applications.

Routing with multiple outputs is shown in Figure 4-48. The only difference between this figure and Figure 4-47 above is that jumpers (zero-ohm resistors) have been installed to enable four FPDP outputs. In this case, with four 100 Mbyte/sec FPDP interfaces, all correlator chips can be dumped every 2.75 milliseconds. If these are upgraded to 400 Mbyte/sec FPDP-II interfaces (which, as previously mentioned could just entail reprogramming the LTA Controller FPGAs), all correlator chips could be dumped every 700 µsec. Although this is probably extreme, it does indicate that the proposed design is capable of meeting all foreseeable performance requirements (given enough back-end computing power).

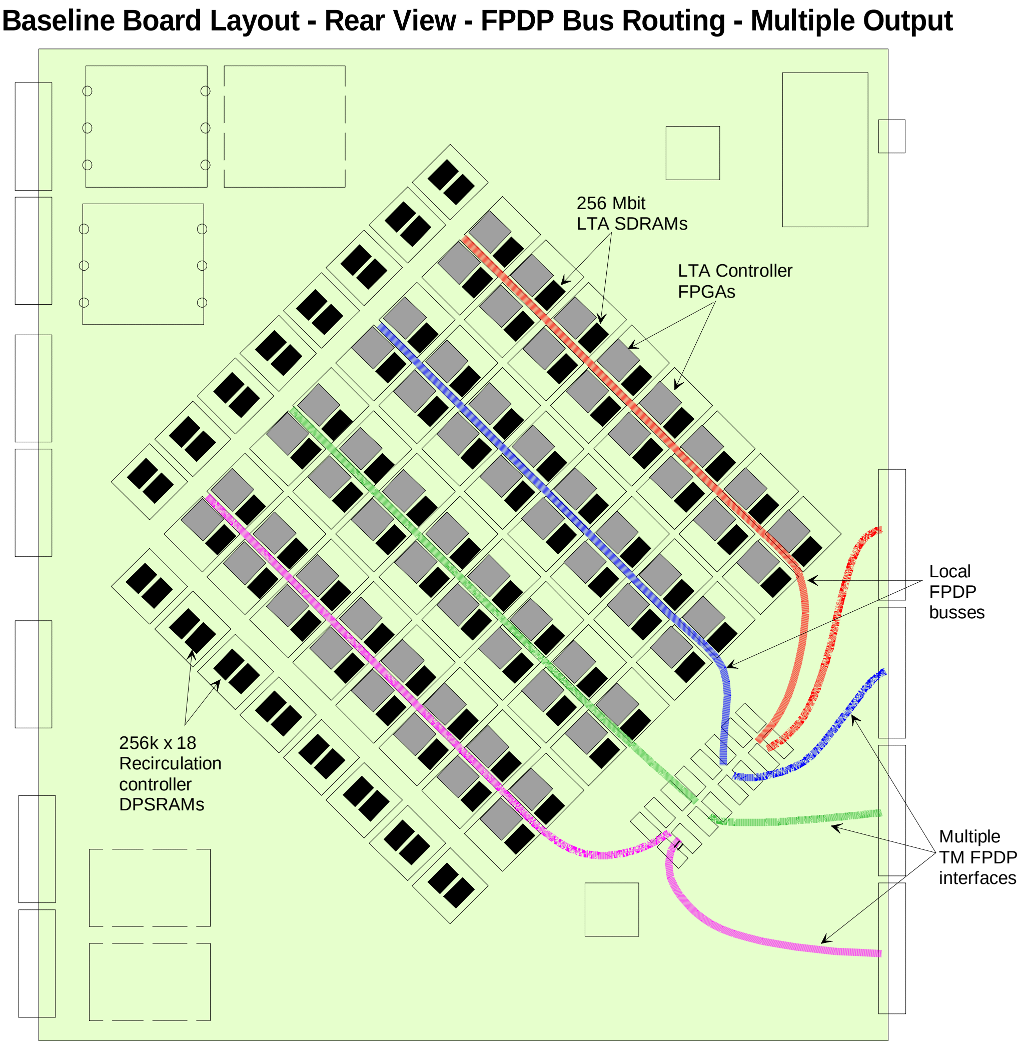


Figure 4-48 Local and external FPDP data routing with all four FPDP interfaces installed and enabled. With 400 Mbyte/sec FPDP-II capability, all lags from all correlator chips could be dumped every 700 µsec.

5 Phasing Board

Phasing Boards are where the phased outputs of the array are generated. Each Phasing Board phases data from one sub-band from up to 48 stations, although provision may be made for phasing data from a sub-band pair. A detailed functional block diagram is shown in Figure 5-1. The design contains a great deal of flexibility for sub-arraying, output requantization, and output formatting.

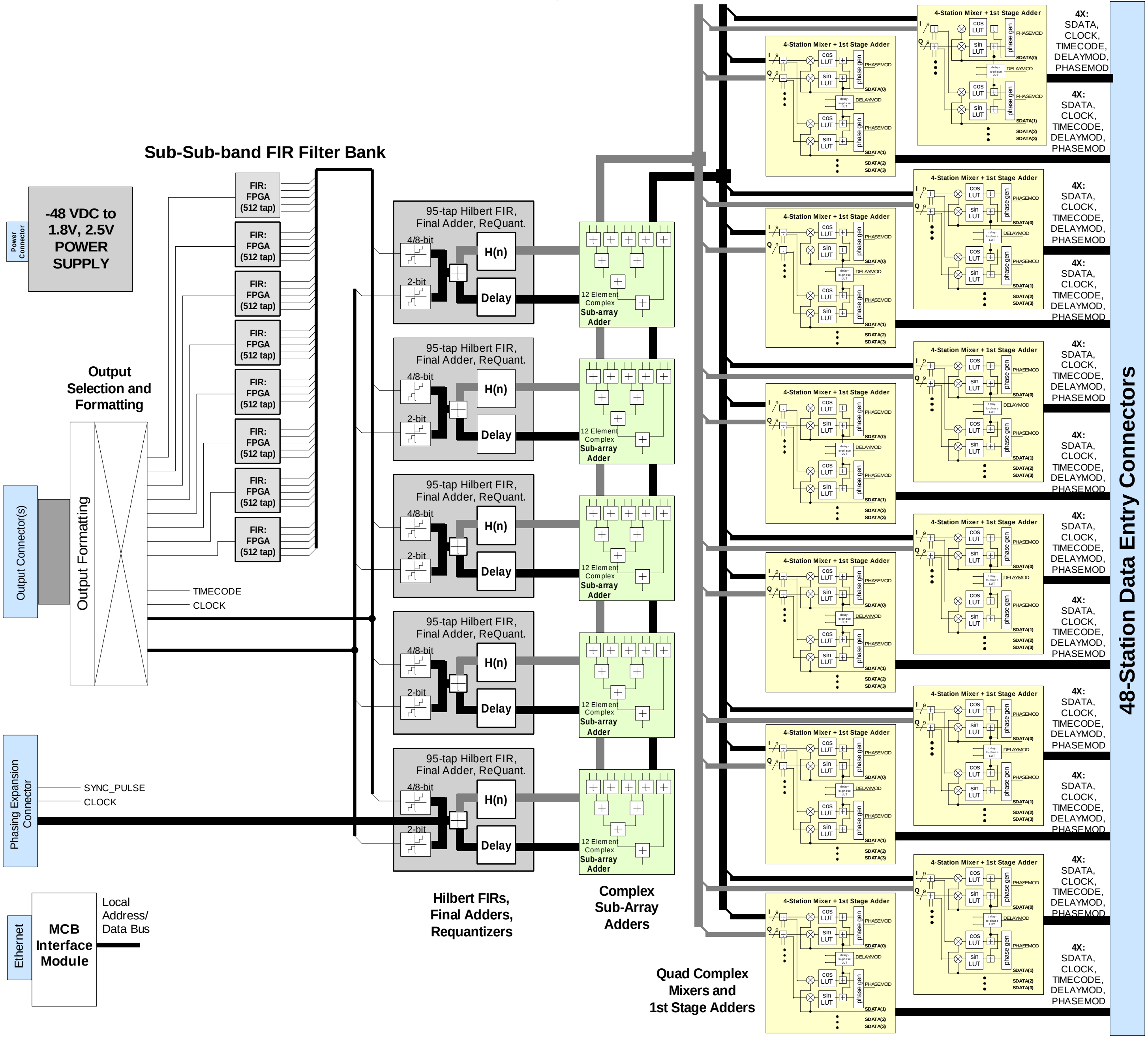

Sub-Sub-band FIR Filter Bank
Power Connector
-48 VDC to 1.8V, 2.5V POWER SUPPLY
FIR: FPGA (512 tap)
Output Selection and Formatting
Output Connector(s)
Output Formatting
TIMECODE
CLOCK
Phasing Expansion Connector
SYNC_PULSE
CLOCK
Ethernet
MCB Interface Module
Local Address/ Data Bus
95-tap Hilbert FIR, Final Adder, ReQuant.
4/8-bit
2-bit
H(n)
Delay
12 Element Complex Sub-array Adder
Hilbert FIRs, Final Adders, Requantizers
Complex Sub-Array Adders
Quad Complex Mixers and 1st Stage Adders
4-Station Mixer + 1st Stage Adder
I
Q
cos LUT
sin LUT
phase gen
PHASEMOD
DELAYMOD
SDATA(0)
SDATA(1)
SDATA(2)
SDATA(3)
4X: SDATA, CLOCK, TIMECODE, DELAYMOD, PHASEMOD
48-Station Data Entry Connectors

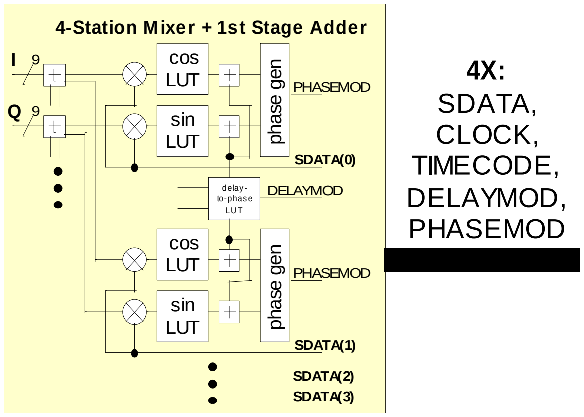
Figure 5-1 Phasing Board block diagram. Up to 5 sub-arrays from up to 48 stations can be handled. The output section allows phased sub-bands to be split into "sub-sub-bands" using digital FIR filters in FPGAs, and an output switch selects the desired data. Simultaneous 2 and 4/8-bit re-quantization is possible, and a non-requantized output connector is provided for expansion beyond 48 stations.

Some notable features and restrictions of the Phasing Board design shown in Figure 5-1 are as follows:

- To alleviate extreme interconnect routing, data is phased-up in two stages. In the first stage, complex phase-rotation and summation is performed in groups of 4 stations. In the second stage, 5 independent sub-array adders can choose any of the outputs from the first stage adders.
- The current concept is for each Phasing Board to operate on only a single sub-band for a total bandwidth of 128 MHz. However, it may be possible to operate on a sub-band pair for a total 256 MHz bandwidth. The decision between the two options will be made during detailed design (but entry via the Phasing Board Entry Backplane seems feasible—see section 6.4).
- Each phased output can be requantized to 2, 4, or 8 bits.
- Eight “sub-sub-band” FIR filters are provided to allow selection of a small part of a phased sub-band.
- An output selection switch allows dynamic selection of various phased outputs to the output connector. The output connector and number of data streams available is TBD.
- One sub-array output before requantization is provided so that with some small amount of external hardware, expansion beyond 48 stations is possible.

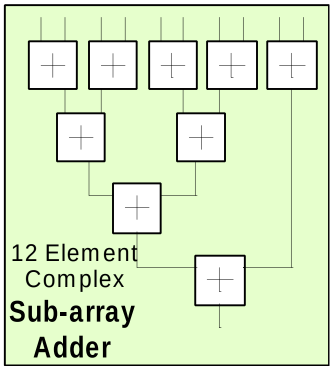
A detailed description of the functional blocks shown in Figure 5-1 is provided in the following sub-sections. Test results of the DSP functions on the Phasing Board are in [9].

5.1 4-Station Mixer and 1st Stage Adder



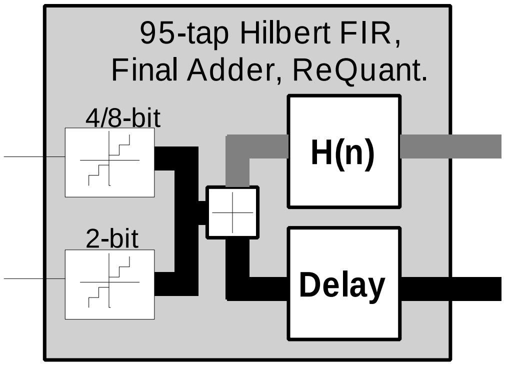
This block performs a many-bit complex mix on the data followed by a complex addition. Phase used in the complex mixer is generated from PHASEMOD and from a delay-to-phase lookup table with input from DELAYMOD for very fine delay tracking. The adder can select which stations it is to add together so that it is possible to omit one or more stations from the result. Because this block has 4 stations going to it, (but not the same stations to any other similar block) if the station is not used in this block, it cannot be used in any other block. This 4-station granularity is necessary to avoid excessive routing and device pin requirements that would be necessary for full phasing flexibility. It is expected that this block will be implemented in a single FPGA. A study of the number of bits in the complex mixer and expected dynamic range is given in [9].

5.2 2nd Stage Sub-Array Adder



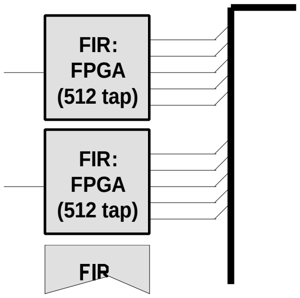
There are five 2nd stage sub-array adders, one for each sub-array that can be formed. Each of the sub-array adders has access to all data from the 1st stage adders so that at this point there is complete flexibility in how sub-arrays can be formed. Each of these is a complex adder tree that can select any of inputs to be added together. The output is a many-bit complex result that goes to its own dedicated downstream Hilbert FIR for final simple data stream generation. This block will be in an FPGA and, because it is quite simple, it may be possible and advantageous to fully integrate it with the following Hilbert FIR FPGA.

5.3 Hilbert FIR



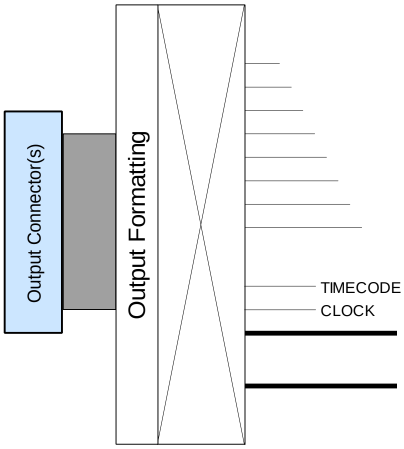
This block uses a Hilbert digital FIR filter to shift the quadrature data by 90° so that it can be properly added to the in-phase data to form a real signal output. An investigation of the performance of this process in [9] indicates that a 95-tap FIR filter should yield more than adequate performance. After the real signal is generated, it is fed into two quantizers that can generate a 4 or 8-bit result at the same time as generating a 2-bit result. The two bit data is normally used just for VLBI data recording. The 4 or 8-bit result is used for real-time feedback into the correlator, or it is used to go to the sub-sub-band FIR filters for narrowband data generation. Four or 8-bit quantization has a negligible sensitivity loss and so filtering and requantizing yet again after this incurs virtually no performance penalty. This block is in an FPGA, although it may be integrated with the 2nd stage adder block since the 2nd stage adder block contains relatively little logic.

5.4 Sub-sub-band FIR Filter Bank



This block is a bank of 8 FIR filters that facilitate the generation of “sub-sub-bands” from phased sub-bands from any or all of the 5 sub-array adder outputs. This function is an NRAO requirement for VLBI that allows recording of only those frequency regions of interest from multiple phased sub-bands.

5.5 Output Switch And Formatting



This block enables dynamic selection of any sub-array data products for routing to the output connector. It also performs any required output data formatting and bit-stream alignment. The connector, the actual output data products available, and the output formatting are TBD. It is important to note that external hardware will be required to align multiple Phasing Board outputs for VLBI data recording or real-time feedback into the correlator (as shown in Figure 2-1 at the beginning of this document).

5.6 Error Detection

Not described in any previous sub-sections are the intended methods for error detection. This is an important consideration since it must be possible to detect and then remove from the phased output any bit streams that are bad due to faulty connections or faulty hardware. Straw-man error detection mechanisms are as follows:

- Data into the Phasing Board from the Station Data Fanout Board contains embedded identifiers and error detection capability. This should be sufficient for error detection of data and data paths into the 1st stage adder blocks.

- A digital power meter and average value meter will be placed at each of the following locations: (Each power meter requires a many-bit multipler and accumulator, although it should be possible to time-multiplex the use of the multiplier across multiple locations on the same chip if it is necessary to save logic.)

    1. Complex output of the 1st stage adders.

2. Inputs and output of the 2nd stage adder.

3. Inputs and outputs of the Hilbert FIR block. There will be a power meter and average value meter before and after requantization.

4. Inputs and output of each sub-sub-band FIR.

5. Inputs and outputs of the output switch and formatter. Due to the large number of I/O here, it may be prudent to use one power meter that can be connected by a switch to any desired point within the device.

- The data out of the Phasing Board may contain an embedded identifier and error detection code such as CRC-4 in a similar fashion to that embedded in the data coming into the Phasing Board. However, this may have a negative impact on data quality, and so a selectable output test vector sequence may be more appropriate.

6 Miscellaneous Modules

There are several miscellaneous modules in the system that facilitate connection of the main digital boards as shown in Figure 2-1. These modules are described in the following sub-sections.

6.1 Sub-band Distributor Backplane

This is the backplane that four Station Boards plug into and that defines a 16 GHz "station input". This backplane is used to passively rearrange the data from the Station Boards (each one producing all sub-bands for 2 basebands) into sub-band outputs on MDR-80 connectors. Each of these sub-band outputs (MDR-80 connectors) contains data for one sub-band from each of the Station Boards' basebands. This arrangement allows the flexibility of trading off spectral resolution and number of stations for bandwidth since each sub-band correlator can process one, some, or all of the sampled data streams (sub-bands of basebands) going to it. A preliminary physical layout of the backplane is shown in Figure 6-1.

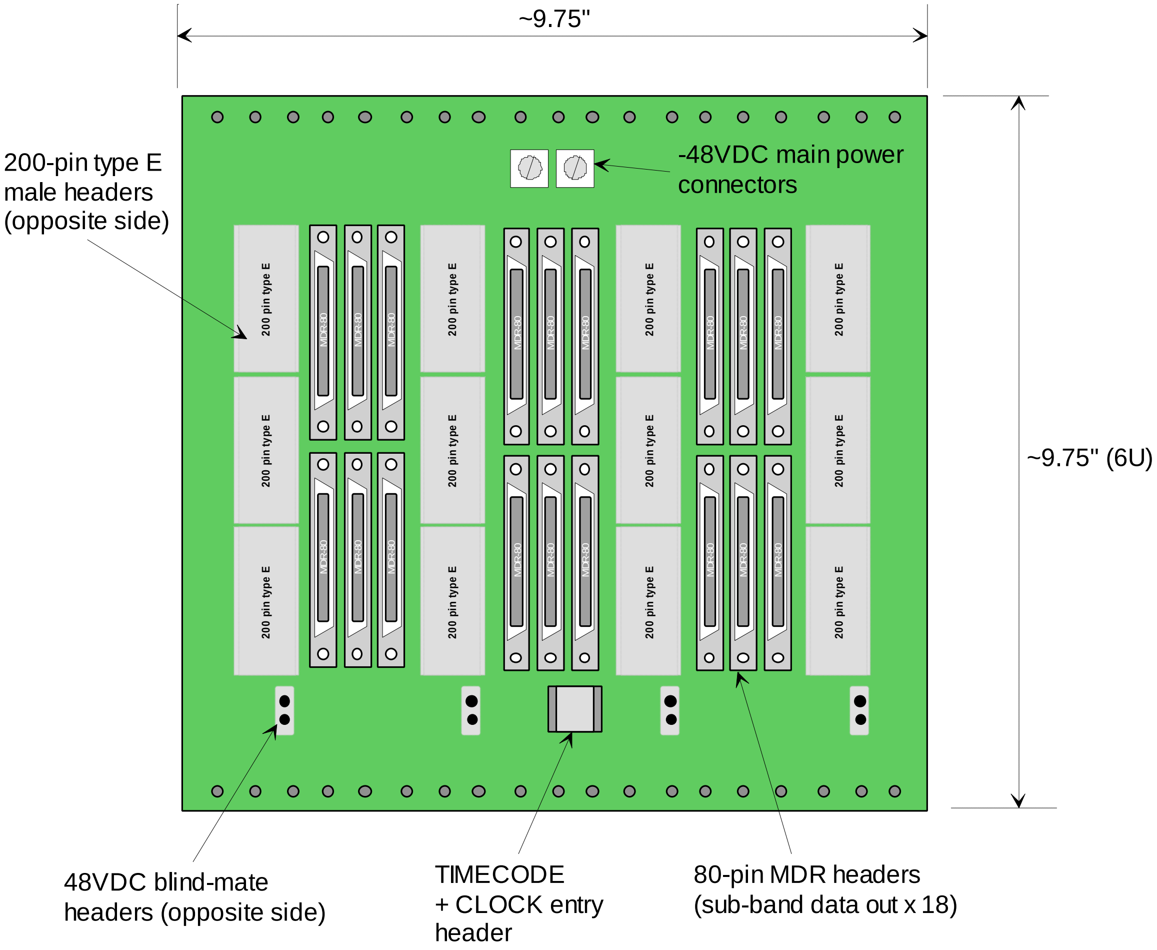

~9.75"
200-pin type E male headers (opposite side)
-48VDC main power connectors
200 pin type E
MDR-80
~9.75" (6U)
48VDC blind-mate headers (opposite side)
TIMECODE + CLOCK entry header
80-pin MDR headers (sub-band data out x 18)

Figure 6-1 Preliminary layout of the Sub-band Distributor Backplane. Four Station Boards plug into it, and each MDR-80 connector contains data from one sub-band of basebands from all Station Boards. Station Boards are spaced apart so that each one has 3 full (VME) slots to itself.

A simplified diagram showing data paths on the backplane for one MDR-80 connector is shown in Figure 6-2 (a). A pseudo-schematic representation is shown in Figure 6-2 (b). All signals on this backplane that eventually travel over cables connected to the MDR-80 connectors use differential signaling (LVDS). Note that DUMPTRIG and the other control signals should be driven from the master slot (#1), so that only the master slot need have a Station Board in it to function. Any skew between clocks, data, and control signals will be compensated for in downstream hardware.

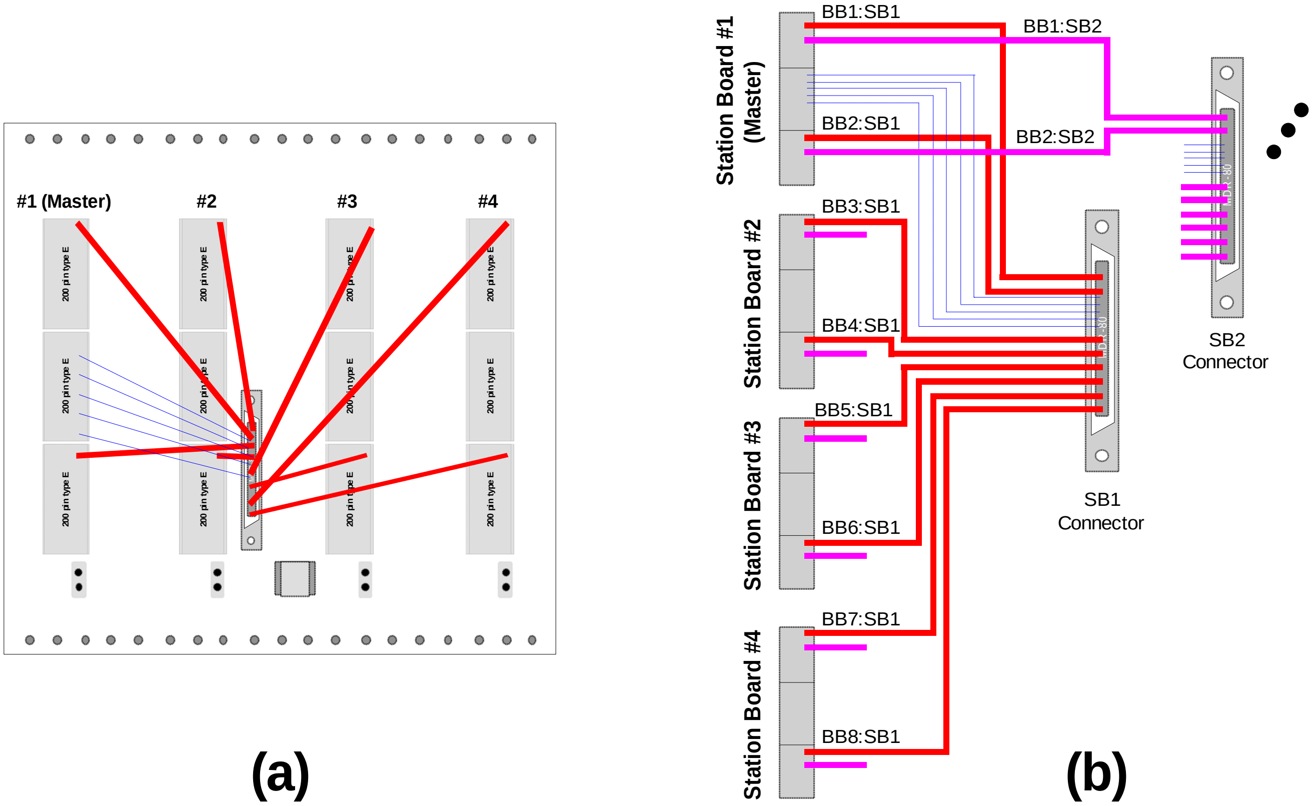


Figure 6-2 Sub-band Distributor Backplane data routing. In (a), data routing for the physical board is shown for one connector. The thick red lines are sub-band data, and the thin blue lines are control signals (CLOCK, TIMECODE, DUMPTRIG, DELAYMOD, PHASEMOD). (b) is a pseudo-schematic representation of the connections on the board.

Note that the sub-band designators of Figure 6-2 (b) for a particular connector are physical designators only. That is, the "SB1 Connector" contains data coming from switch output #1 on every Station Board (section 3.8). The *use* of each of the switch outputs—the sub-band FIR it is connected to, the sub-band's placement in its baseband, and its width—is user configurable.

6.1.1 MDR-80 Connector Pin Assignments

The MDR-80 connector headers on the Sub-band Distributor Backplane have pin locations according to Figure 6-3. Signal assignments are shown in Table 6-1.

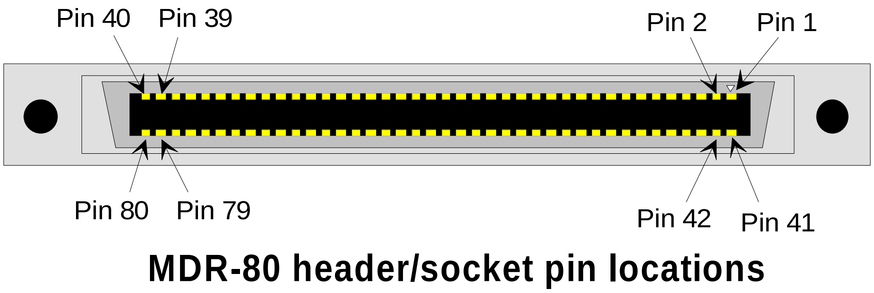


Figure 6-3 MDR-80 header/socket pin locations. Source: 3M data sheet, P/N 10280-6212VC.

Signal	Pin (+)	Pin (-)	Description
DATA1-0	1	2	**BB1** data line 0 (LSB) (Slot1-R)
DATA1-1	3	4	BB1 data line 1
DATA1-2	5	6	BB1 data line 2
DATA1-3	7	8	BB1 data line 3 (MSB)
DATA2-0	9	10	**BB2** data line 0 (LSB) (Slot1-L)
DATA2-1	11	12	.
DATA2-2	13	14	.
DATA2-3	15	16	.
SPARE1	17	18	spare (1) for SDFB prog/mon
CLOCK	19	20	128 MHz clock
SPARE2	21	22	spare (2) for SDFB prog/mon
TIMECODE	23	24	timing
DUMPTRIG	25	26	dump control
SPARE3	27	28	spare for SDFB prog/mon
DATA3-0	29	30	**BB3** data line 0 (LSB) (Slot2-R)
DATA3-1	31	32	.
DATA3-2	33	34	.
DATA3-3	35	36	.
DATA4-0	37	38	**BB4** data line 0 (LSB) (Slot2-L)
DATA4-1	39	40	.
DATA4-2	41	42	.
DATA4-3	43	44	.
DATA5-0	45	46	**BB5** data line 0 (LSB) (Slot3-R)
DATA5-1	47	48	.
DATA5-2	49	50	.
DATA5-3	51	52	.
DATA6-0	53	54	**BB6** data line 0 (LSB) (Slot3-L)
DATA6-1	55	56	.
DATA6-2	57	58	.
DATA6-3	59	60	.
DELAYMOD	61	62	real-time delay model
PHASEMOD	63	64	phase models
DATA7-0	65	66	**BB7** data line 0 (LSB) (Slot4-R)
DATA7-1	67	68	.
DATA7-2	69	70	.
DATA7-3	71	72	.
DATA8-0	73	74	**BB8** data line 0 (LSB) (Slot4-L)
DATA8-1	75	76	.
DATA8-2	77	78	.
DATA8-3	79	80	.

Table 6-1 MDR-80 pin assignments. Slot numbers refer to Sub-band Distributor Backplane slot numbers (**Figure 6-2**) and whether it is the ‘R’ or ‘L’ output. This maps into ‘BB’ numbers as indicated.

6.2 Station Data Fanout Board

The Station Data Fanout Board acts as a repeater and fanout board for sub-band data coming from Sub-band Distributor Backplanes (see Figure 2-1). This board will reside at the back of the baseline racks and fanout cables will be routed within the rack space (refer to section 7). A layout diagram of this board is shown in Figure 6-4.

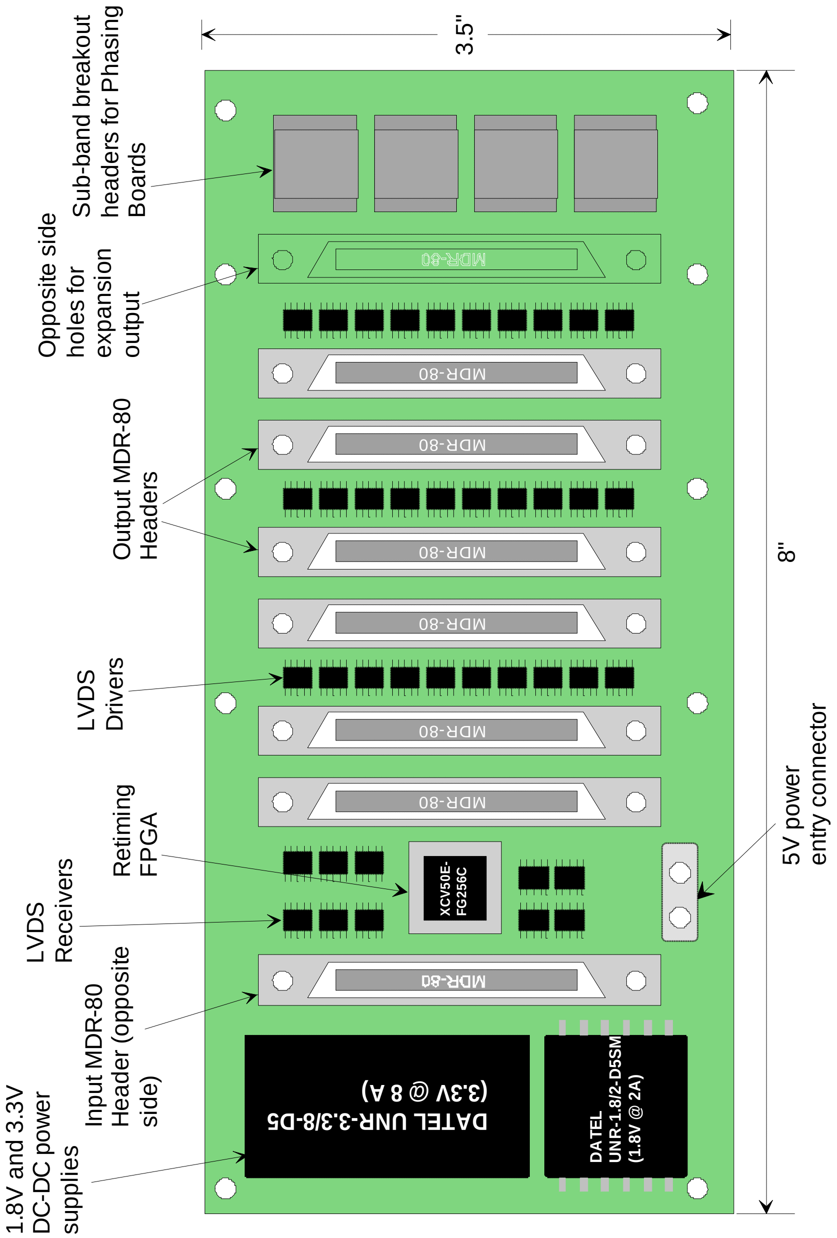

3.5"
8"
Sub-band breakout headers for Phasing Boards
Opposite side holes for expansion output
Output MDR-80 Headers
LVDS Drivers
Retiming FPGA
LVDS Receivers
Input MDR-80 Header (opposite side)
1.8V and 3.3V DC-DC power supplies
5V power entry connector
MDR-80
XCV50E-FG256C
DATEL UNR-3.3/8-D5 (3.3V @ 8 A)
DATEL UNR-1.8/2-D5SM (1.8V @ 2A)

Figure 6-4 Station Data Fanout Board physical layout. The sub-band cable input is fanned-out by a factor of 6 with an additional output for routing to other racks. Also, breakout for cables going to Phasing Boards is provided (shown here with a sub-band pair per breakout). MDR-80 pinouts are according to **Figure 6-3** and **Table 6-1**.

The “Retiming FPGA” resynchronizes the data before re-transmission. This FPGA must be programmed and must have its status checked using signals coming from Station Boards as outlined in section 3.10. It may also be desirable/possible to power cycle this board by controlling power supply inhibit signals to the DC-DC power supplies. (Status LEDs, although not shown in the figure, will be provided to allow secondary visual checks by on-site personnel.) It must also be possible for daisy-chained boards to be programmed and monitored in a similar fashion. Finally, since this board contains active components it must be possible to hot-swap it. A straw-man concept of mechanics that could facilitate hot-swapping is shown in Figure 6-5.

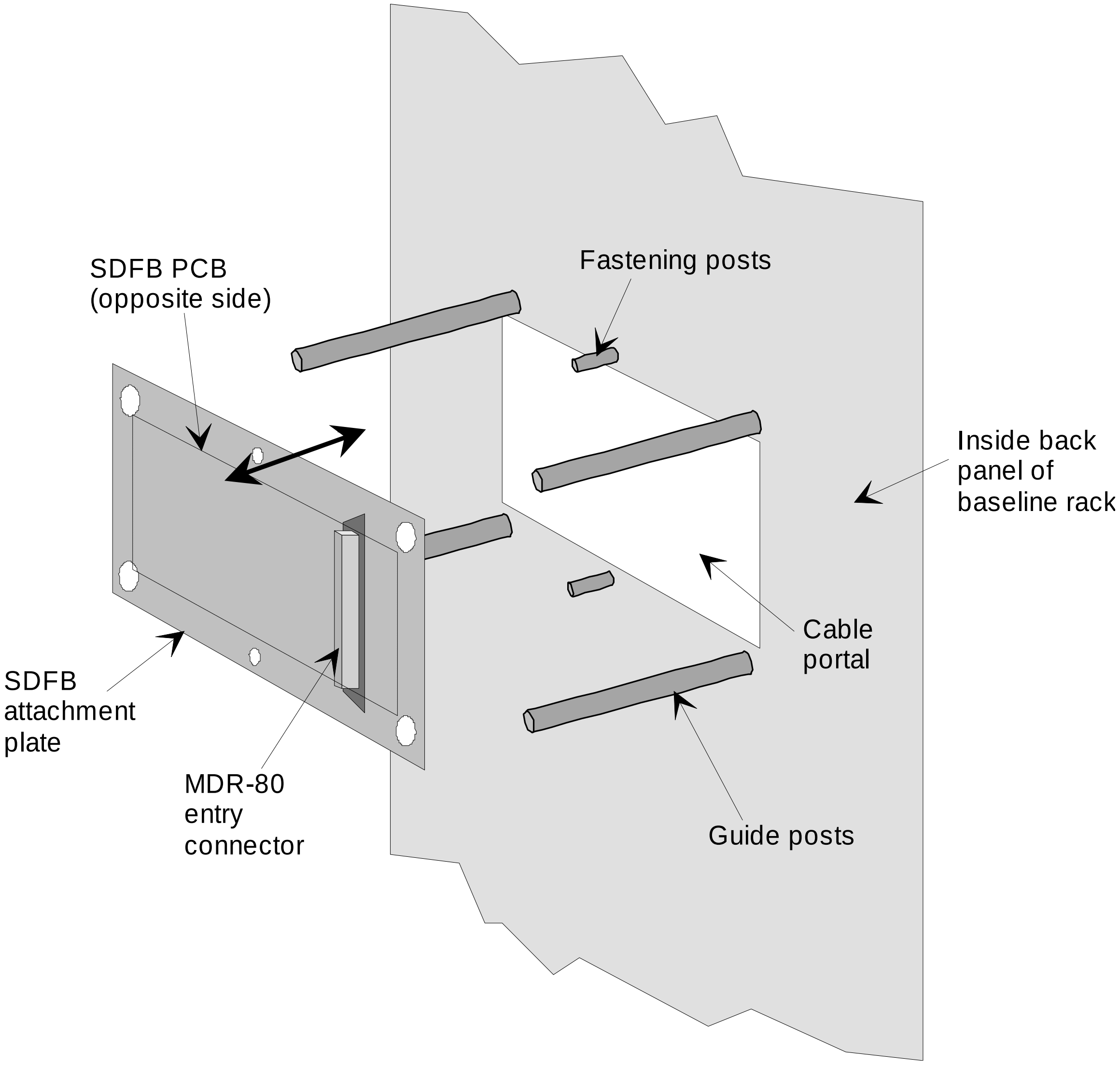


Figure 6-5 Straw-man concept for fastening the Station Data Fanout Board (SDFB) to the inside back panel of the baseline rack. It must be possible to hot-swap the SDFB without shorting any signals or power. The guide posts allow the SDFB assembly (includes the PCB and an attachment plate) to be safely extracted from the panel so that the existing cabling that feeds through the “cable portal” and connects to the SDFB PCB can be removed. The fastening posts protrude through the attachment plate and are used to fasten the SDFB assembly with wing-nuts. Not shown is a power switch protruding through the SDFB attachment plate to remove +5V power from the PCB.

6.3 Baseline Entry Backplane

This backplane is used to passively feed data from 16 stations/MDR-80 connectors to the Baseline Board. The backplane blind-mates with the front-entry Baseline Board. A physical layout of the backplane is shown in Figure 6-6. In each sub-rack (i.e. “crate” or “card cage”) up to eight of these will be mounted to allow insertion of up to eight Baseline Boards. It is possible that these could be formed into one monolithic backplane, but small individual modules provide more flexibility in mixing different boards in the same sub-rack.

An important consideration is the total number of pins (and more importantly the insertion force that a large number of pins demands) in the connectors that mate with the Baseline Board. A full discussion of insertion force considerations is provided in section 4.5.

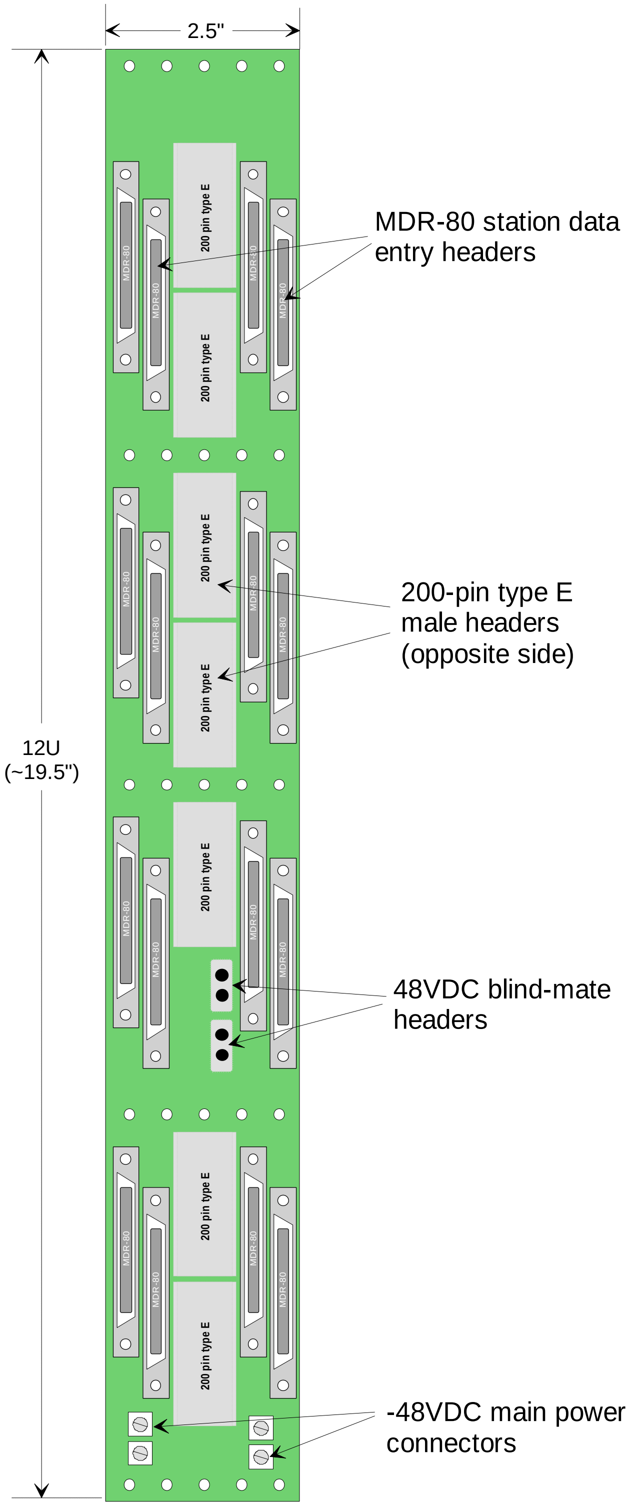

2.5"
12U
(~19.5")
MDR-80
200 pin type E
MDR-80 station data entry headers
200-pin type E male headers (opposite side)
48VDC blind-mate headers
-48VDC main power connectors

Figure 6-6 Baseline Entry Backplane physical layout. This backplane routes signals on the 16 (8 ‘X’; 8 ‘Y’) input MDR-80 connectors to connectors that blind-mate with the front-entry Baseline Board. The MDR-80 connectors are staggered to prevent “pile-up” of cable that is thicker than the connector.

6.4 Phasing Board Entry Backplane

This backplane facilitates entry of sub-band data from 48 stations into the Phasing Board. The layout shown in Figure 6-7 contains 48, 36-pin 2.0 mm connectors that mate with GORE “Eye-opener” cable connectors—each one carries 2 sub-bands, CLOCK, TIMECODE, DELAYMOD, and PHASEMOD coming from a Station Data Fanout Board. This preliminary layout indicates that 48 connectors, each containing a sub-band pair, fit on the backplane. Thus, it should be possible to phase 2 sub-bands on each Phasing Board as indicated in section 5. Note that it may be possible to mate the Eye-opener cable *directly* with the “200-pin type E” connectors if the 8 row configuration is used.

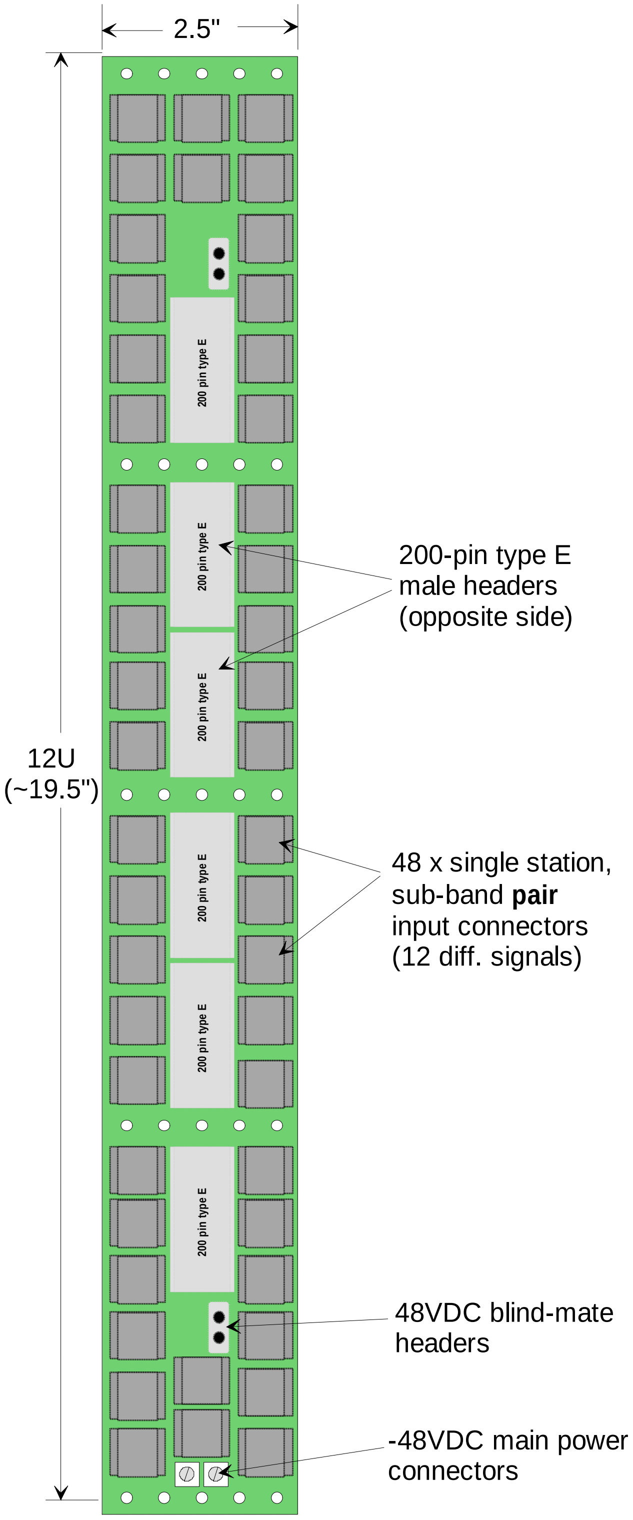


Figure 6-7 Phasing Board Entry Backplane preliminary layout. This layout supports 48 stations and 2 sub-band pairs.

6.5 TIMECODE Generator Box

The TIMECODE Generator Box (TGB) provides the correlator with timing information that all operations within the correlator are synchronized to. This box is identified in Figure 2-1.

The TGB has the following inputs:

- Reference clock at some (sub-?) multiple of 128 MHz. **A project-wide decision (J. Jackson) has been made to use VLBI standard frequencies, and use of this frequency reflects that decision.**
- Reference time tick. This should be compatible with TIMECODE (e.g. 1 Hz tick).

The TGB has 48[36] outputs, each of which is identical and contains the following signals:

- 128 MHz digital clock.
- Four programmable TIMECODE signals. One of these TIMECODEs is the real “wall-clock time”, and three of them are programmable. The parameter space within which the programmable TIMECODEs can be configured is TBD. Nominally, they are to provide non-real-time VLBI capability (possibly with a speed-up factor as well). Each set of four Station Boards can select one of the TIMECODEs for use.

All outputs are LVDS (probably) using the GORE “Eye-opener” 2.0 mm connectors and 50-ohm cable. Routing within the TGB and cables that route signals to Sub-band Distributor Backplanes should be reasonably well length-matched since any mismatch ultimately must be absorbed in the correlator chip. The TGB should be located in the center of the correlator station racks.

The TGB will contain an MCB interface module as described in other sections. This module will allow the setting of the TIMECODE epoch synchronous to the input reference time tick. Communication to the TGB should be such that there is no confusion in the epoch setting.

6.6 Other Modules

Several other modules are required to realize the full functionality that is possible with the correlator. These modules are indicated in Figure 2-1 and are as follows:

[36] It could have more, but this seems to be the number that satisfies the entire EVLA project requirements.

- VLBI recorder interface. This interface synchronizes data from one or more Phasing Boards in preparation for transmission to a VLBI data recorder. This interface is an NRAO responsibility.

- Phased output feedback interface. This interface synchronizes data from one or more Phasing Boards for feedback, in real time, into the correlator. This interface may be combined with the VLBI recorder interface. This interface is an NRAO responsibility.

- VLBI recorder Station Board interface. This interface is required to feed the data from a VLBI recorder (i.e. playback unit) to the Station Board. The straw-man concept is to *always* use the fiber-optic receiver module (section 3.1) to import data into the Station Board. Thus, the VLBI interface would be a unit that connects to a VLBI data output unit and produces a fiber-optic signal compatible with the fiber-optic receiver module.

- General phased output interface. This interface may be required to allow the output from multiple Phasing Boards to be routed to an external data analysis system. This interface could be integrated with the VLBI recorder interface and output feedback interface.

7 System Design

Previous sections have defined in some detail all of the modules in the correlator system that can be used as building blocks to construct a correlator system of virtually any size. This section looks at system design issues including rack design, floor layout, power supplies, and external computers.

7.1 Sub-Rack and Rack Design

Twenty-four inch wide racks will be used so that each board (Station Board, Baseline Board, Phasing Board) can have a full 3 VME slots (~2.5”) of its own. This width is primarily necessary for data routing to the boards (sections 6.1, 6.3, and 6.4), but also provides ample room between boards for airflow. This width allows 8 boards to be plugged into each sub-rack. Since 4 Station Boards are required for each full-bandwidth antenna, one sub-rack can thus hold enough boards for 2 antennas (stations). The current concept for backplanes for the Baseline Boards and Phasing Boards allow these boards to be mixed within a sub-rack as desired. Indeed, this capability will allow phasing of “some” sub-bands without having to have dedicated phasing sub-racks.

The profile of a straw-man design for a 12U sub-rack is shown in Figure 7-1. Airflow is as indicated with blue and orange arrows.

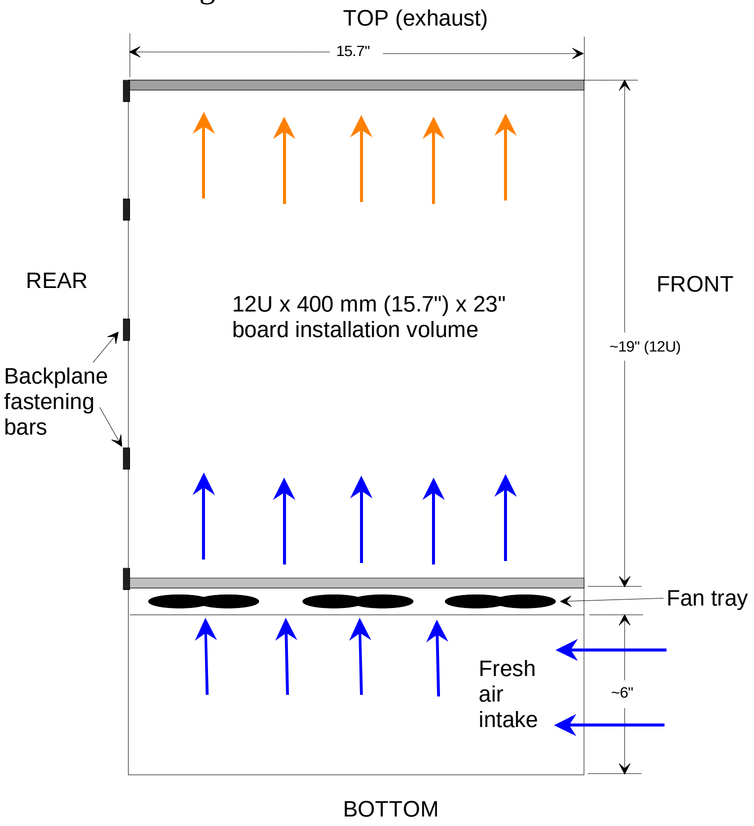


Figure 7-1 Profile of a straw-man design for the sub-rack. Each sub-rack has its own fresh (cool) air supply. The fans in the fan tray can fail and thus must be able to be hot-swapped.

The sub-rack design allows each sub-rack to have its own fresh air supply provided by an air channel. About 8" of exhaust space is required above each sub-rack, thus the total linear space occupied by a sub-rack is about 34". Using this configuration, it should be possible to install two sub-racks in a ~7' rack, the profile is as shown in Figure 7-2.

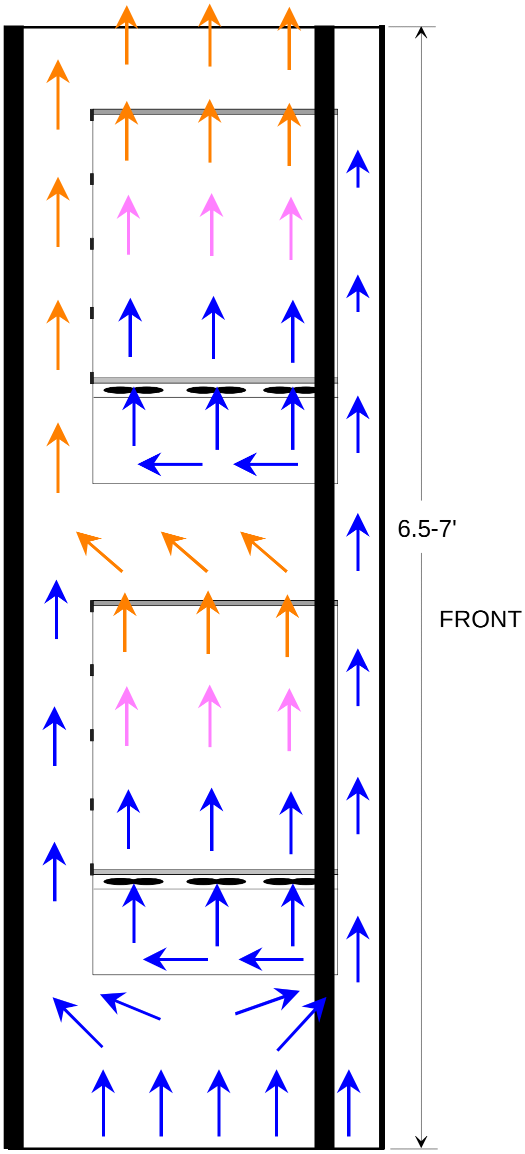


Figure 7-2 Profile of 7' rack with two 12U sub-racks. Approximate airflow vectors are shown with arrows.

The total estimated power dissipation for each fully populated sub-rack is about 1800 W (200 W per board and 90% DC-DC power supply efficiency), thus each rack will dissipate about 3600W. A denser configuration is possible if floor space is limited. This would see the rack height reach about 9.5' with 3 sub-racks in an arrangement similar to Figure 7-2. This configuration could have a total estimated power dissipation of 3x1800 W = 5.4 kW—probably the maximum power that could be handled in one rack. A denser configuration where air flows upward across boards in all sub-racks is possible, but more extreme cooling measures may be required. However, this configuration is advantageous in that there are no fans that can fail and need to be hot-swapped in the racks (i.e. the failure point is the air conditioners only).

The rack profile shown in Figure 7-2 is essentially the station rack profile. This rack has a total depth of about 3', enough for the cabling to the Sub-band Distributor Backplane and the FOTs receiver module mezzanine cards that plug into the Station Boards. However, the cabling inside the baseline rack is much more extensive and therefore extra rack depth is required to accommodate it. A straw-man profile of the baseline rack that should accommodate all of the necessary cabling is shown in Figure 7-3.

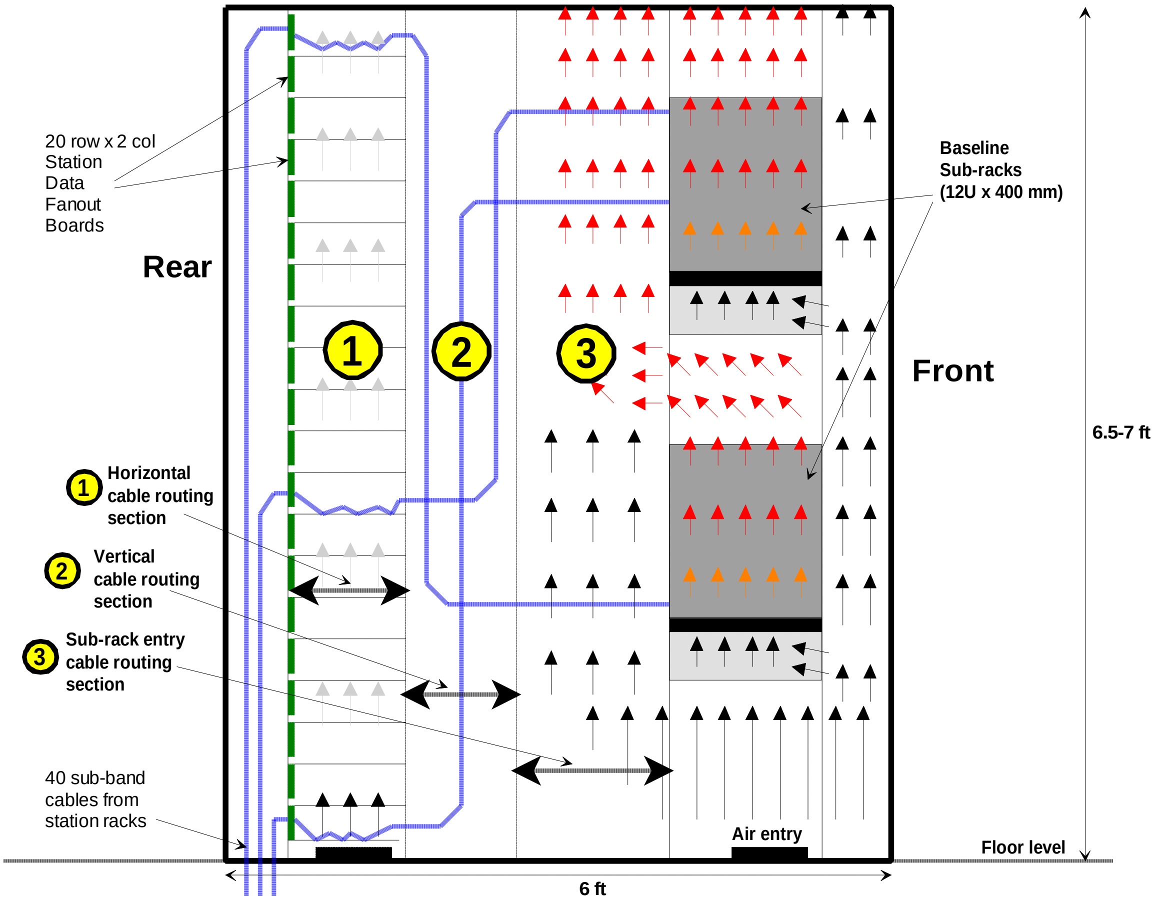

20 row x 2 col
Station
Data
Fanout
Boards
Rear
1
2
3
Baseline
Sub-racks
(12U x 400 mm)
Front
6.5-7 ft
1 Horizontal cable routing section
2 Vertical cable routing section
3 Sub-rack entry cable routing section
40 sub-band cables from station racks
Air entry
Floor level
6 ft

Figure 7-3 Baseline rack profile showing 2 baseline sub-racks, approximate airflow vectors, Station Data Fanout Boards, and a straw-man cable routing plan. Cables from station racks (Sub-band Distributor Backplanes) enter through the floor at the back and plug into Station Data Fanout Boards. Cable from the Station Data Fanout Boards (up to 256 cables, each one 3 m long) is routed in three sections as shown. In the first section, cable is routed horizontal-only (into the plane of the figure) until it reaches the correct location. In the second section, cable is routed vertical-only until it reaches the right vertical location. In the third section, cable is routed horizontal-only directly to its plug-in point on the Baseline Entry Backplane. Sections are separated by "grids" that constrain the cable paths.

The cabling arrangement shown in the figure (and described in the figure caption) should be able to arrange the cables in an order that facilitates easy installation and maintenance. The rack is an additional 3 feet (for a total of 6 feet) deep and both sides of the rack must be accessible (with removable panels) in any correlator configuration. Although this plan looks reasonable on paper, development will include building a full mock-up of the baseline rack complete with mock cabling to ensure that the plan is indeed feasible.

7.2 Remote Power Control and Monitoring

Each circuit board's power supply[37] in the correlator system will be monitored[38] for health by using a monitor and control signal provided by the board's DC-DC converter. This signal is pulled low if the power supply is experiencing some problem (temperature, input voltage, internal parameters etc.). This signal can source ~1.5 mA of current at 5.7 VDC, and shorting it to ground with a relay or transistor will shut the DC supply off. Thus, it is possible to remotely control power to each individual circuit board in the system. All of the monitor and control lines in each rack will be routed, via associated backplanes, to one (DB-25 pin) connector connected to a terminal block as shown in Figure 7-4.

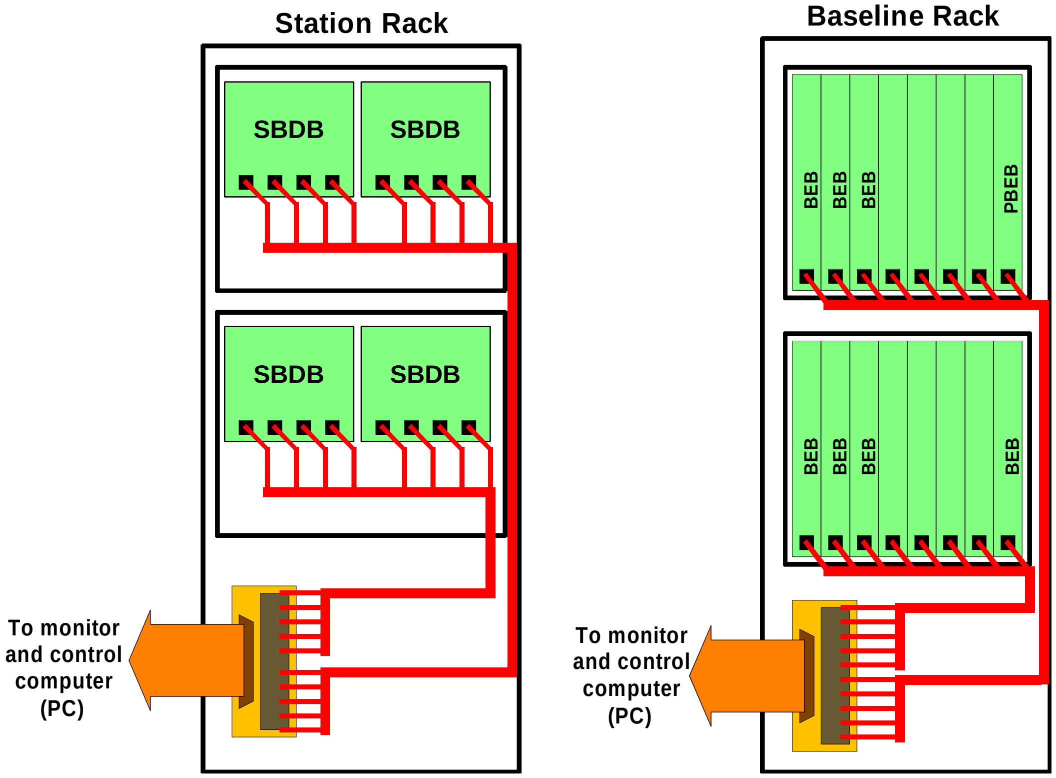

Station Rack
SBDB
SBDB
SBDB
SBDB
To monitor and control computer (PC)
Baseline Rack
BEB
BEB
BEB
PBEB
BEB
BEB
BEB
BEB
To monitor and control computer (PC)

Figure 7-4 Routing of DC-DC converter monitor and control lines for each rack. Each rack has a single cable used for power supply monitor and control.

Each rack thus has a cable coming from it that will be plugged into an external monitor and control card plugged into a computer that is normally not part of correlator system processing and thus less likely to experience crashes from bugs. Also this computer can

[37] This may or may not include Station Data Fanout Board DC-DC supplies. However, it will include the 48 VDC to 5 VDC power supply that supplies 5 VDC to all Station Data Fanout Boards in a rack.
[38] In addition to normal board temperature and voltage level monitoring.

be a more expensive and robust model since there will probably only be one of them. PCI cards that perform the sorts of functions required are inexpensive and readily available.

7.3 Correlator Floor Plan

Floor plans for a 40-station and a 48-station correlator have been developed. These plans are shown in Figure 7-5 and Figure 7-6 respectively.

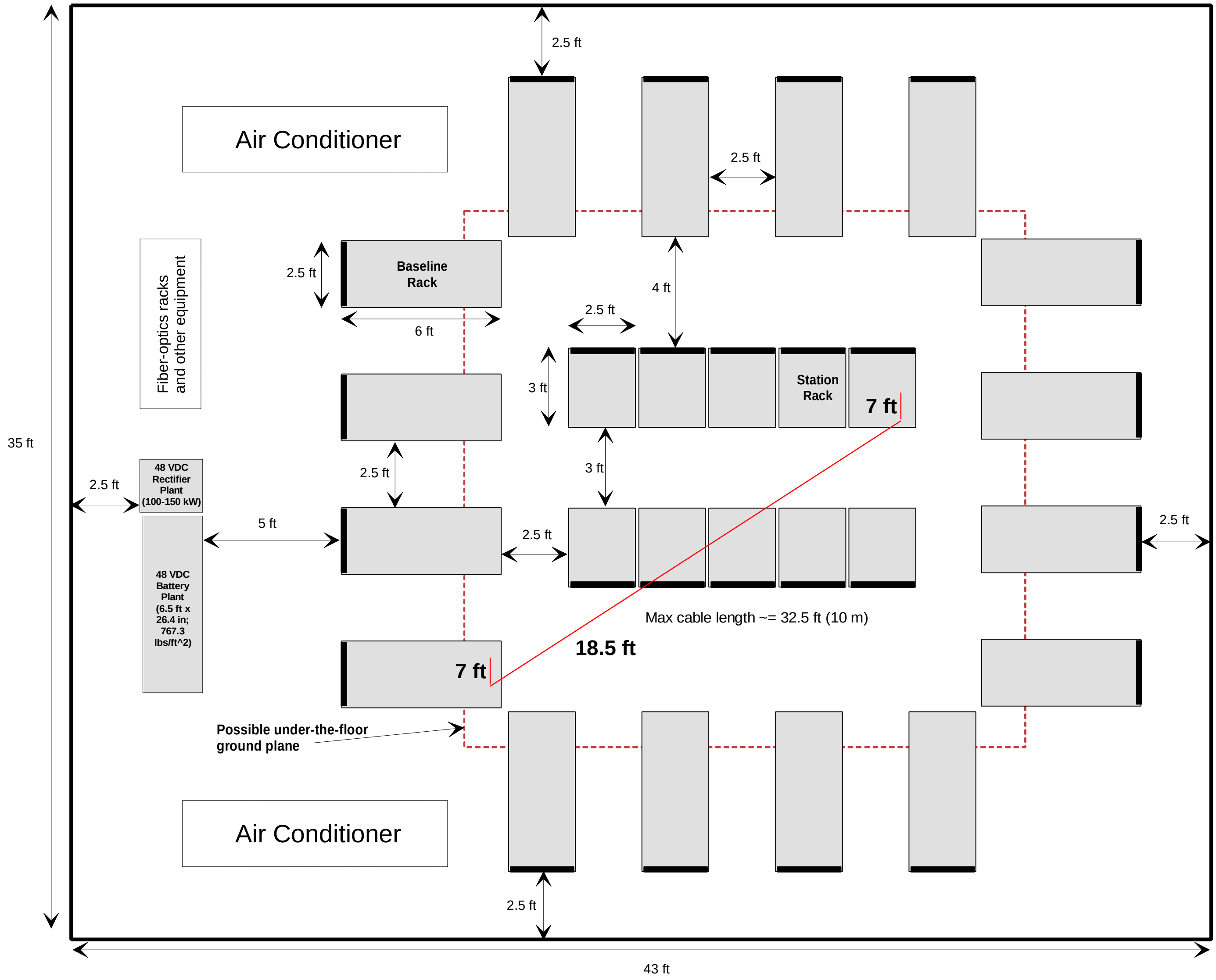

2.5 ft
Air Conditioner
2.5 ft
Fiber-optics racks and other equipment
2.5 ft
Baseline Rack
4 ft
6 ft
2.5 ft
3 ft
Station Rack
7 ft
35 ft
3 ft
2.5 ft
48 VDC Rectifier Plant (100-150 kW)
2.5 ft
5 ft
2.5 ft
2.5 ft
48 VDC Battery Plant (6.5 ft x 26.4 in; 767.3 lbs/ft^2)
Max cable length ~= 32.5 ft (10 m)
18.5 ft
7 ft
Possible under-the-floor ground plane
Air Conditioner
2.5 ft
43 ft

Figure 7-5 Possible **40-station** correlator floor plan. The station racks are in the center of all of the racks since cable from each station rack goes to every baseline rack, each cable must be the same length, and it is desirable to minimize the cable lengths. The floor plan has dimensions of 35 x 43 ft, and there is enough room for additional racks of equipment. Data output processing computers could be mounted in racks or shelves beside each baseline rack so as not to interfere with access to baseline rack cabling. The location of the TIMECODE Generator Box is not yet defined, but it can probably be located in one of the central station racks.

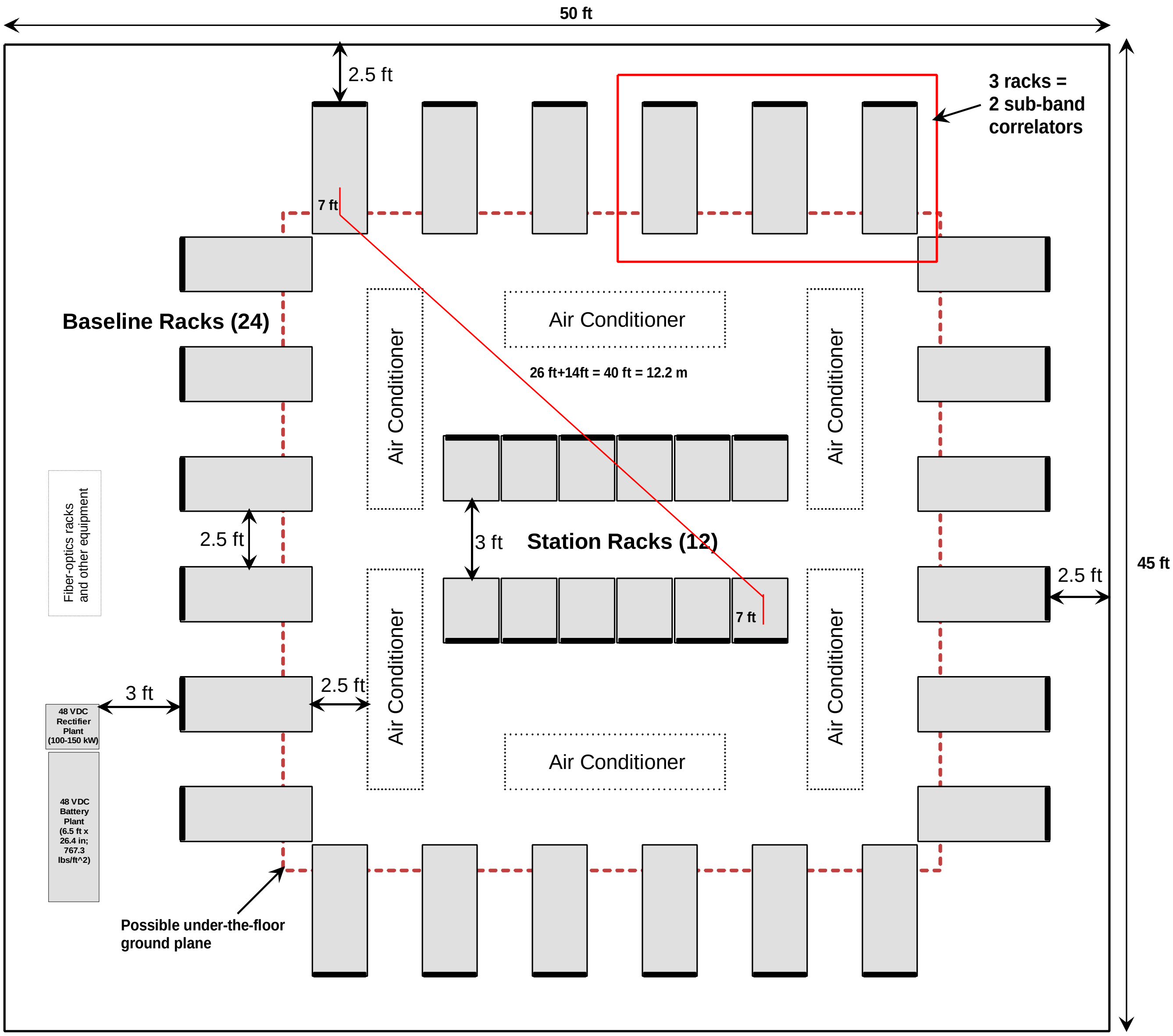


Figure 7-6 Possible **48-station** correlator floor plan. In this arrangement, 3 baseline racks contain all of the boards for 2 sub-band correlators and so there are a total of 1.5 x 16 = 24 baseline racks. The station racks are in the center of all of the racks since cable from each station rack goes to every baseline rack, each cable must be the same length, and it is desirable to minimize the cable lengths. The floor plan has dimensions of 45 x 50 ft, and there is enough room for additional racks of equipment. Data output processing computers could be mounted in racks or shelves beside each baseline rack so as not to interfere with access to baseline rack cabling. The location of the TIMECODE Generator Box is not yet defined, but it can probably be rack-mounted in one of the central station racks.

The racks will be installed on a raised floor and all cabling will route under the floor. The air conditioners will pressurize the floor and each rack will draw cold air from the floor. Because of cabling and the uneven distribution of air conditioners and racks, there will be pressure gradients across the installation. NRAO has requested that each rack have an adjustable vent at the bottom so that airflow can be properly equalized. It is not yet known whether or not additional fans will be installed at the bottom of each rack.

An artist's rendering of a 48-station correlator installation (not including the mains 48 V supply or other racks) is shown in Figure 7-7.

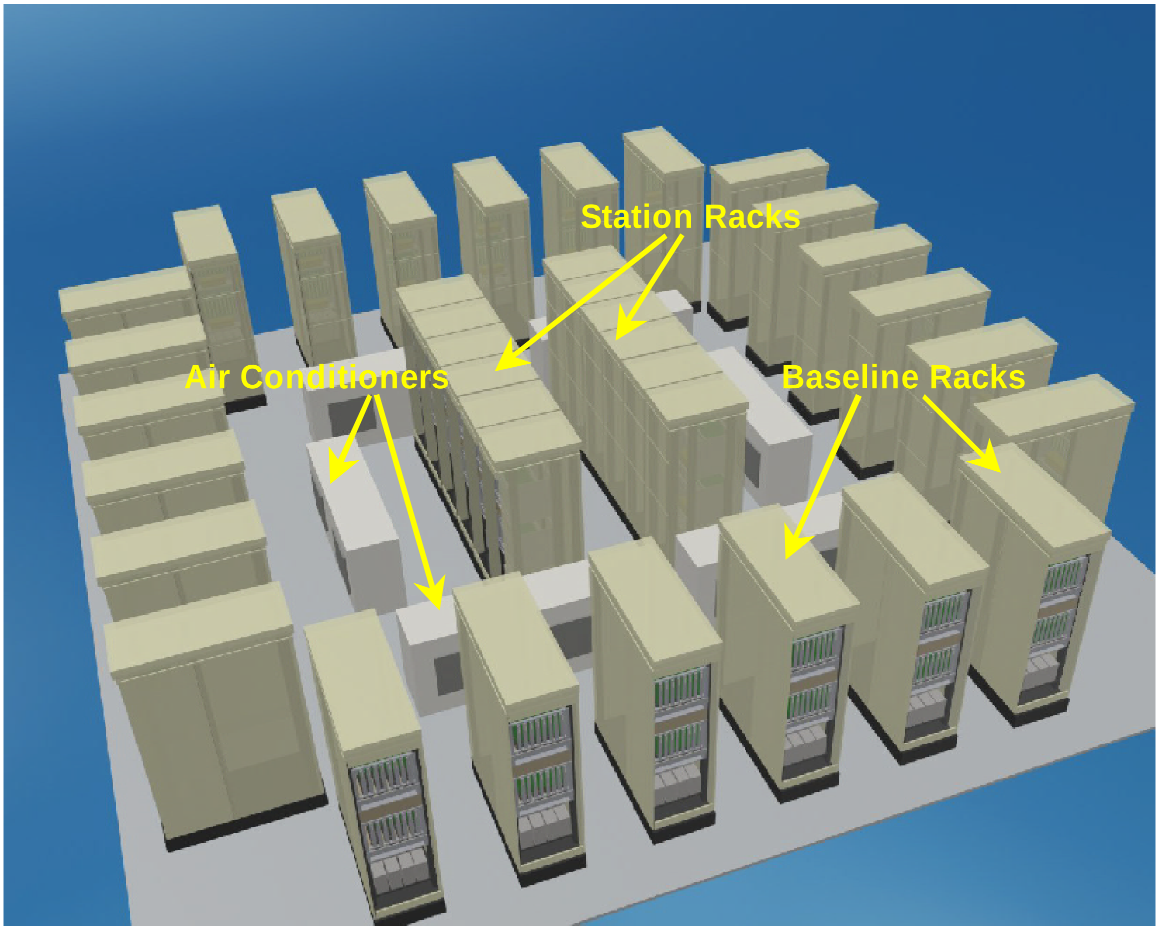


Figure 7-7 Artist's rendering of the 48-station correlator installation.

The "under-the-floor" ground plane in Figure 7-5 and Figure 7-6 is a low-impedance path for shunting common-mode currents that may flow from rack to rack. Backplane grounds (that connect to circuit board grounds) are bonded to sub-racks, sub-racks are bonded to racks, and the racks are bonded to this ground plane. The ground plane is used in conjunction with common-mode filters[39] on the sub-band cables that travel from station racks to baseline racks. That is, the proposed noise attenuation method is to both filter and shunt common-mode noise so that it does not travel on the differential signal lines that connect boards in different racks. Since the correlator will be installed in a screened room, the ground plane can probably be the inside floor of the screened room.

The correlator will take its power from a 48 VDC power plant. By doing this, single-point power supply failures are isolated to a single circuit board that can easily be controlled and hot-swapped as described in previous sections. The 48 VDC plant is a telephony central office-grade supply that contains redundancy so that any failure can be

[39] These filters will probably be installed on entry to the Station Data Fanout Board.

detected and replaced on-line. The hold-time requirement for this plant in the event of an AC power failure is 15 minutes. The power plant manufacturer will be consulted regarding the best method for routing DC power to each rack.

The planned hardware delivered by NRC is everything for a 32-station correlator, with racks for 40 stations. There is a strong indication, however, that the full EVLA expansion correlator will require 48 stations and thus the system design must allow for this eventuality.

7.4 Correlator Computing

7.4.1 Hardware Configuration

The system module diagram of Figure 2-1 shows Station Boards, Phasing Boards, and Baseline Boards connecting to external control and data processing computers. Each of the boards has an MCB/CPU mezzanine card mounted on it with a bus interface for communication with on-board devices, and an Ethernet interface for communication with the external computers. In addition, each Baseline Board has a Front Panel Data Port (FPDP) interface that pipes correlated data to a back-end computer.

The straw-man concept for the control and back-end computers is to use COTS PC boxes running Linux since they offer a high performance-to-price ratio in a generic package. These boxes can easily be upgraded to the latest performance machine with minimal impact on software as long as Ethernet and FPDP interface cards continue to be available, and as long as the software is written so that it is loosely coupled to the hardware platform. This should be possible since all of the “hard” real-time control software will reside on the MCB mezzanine cards. The computers with the FPDP interfaces will see data frames coming from Baseline Boards (section 4.3), that get processed and/or buffered as they arrive without any special real-time processing requirements. These computers write FITS file fragments to network disks—these fragments get “vacuumed up” by downstream image/archive processing computers.

The current estimate (ref: Aug. 27-28/01 meeting in Socorro) is that 1 computer will be required for Station Board control and 1 to 4 computers will be required for Baseline Board and Phasing Board control. The number of computers required to handle FPDP data is scaleable, but the straw-man plan is to use 1 computer for every 4 Baseline Boards, and to arrange these computers in multiple Beowulf clusters. A simplified diagram of the correlator computing environment is shown in Figure 7-8.

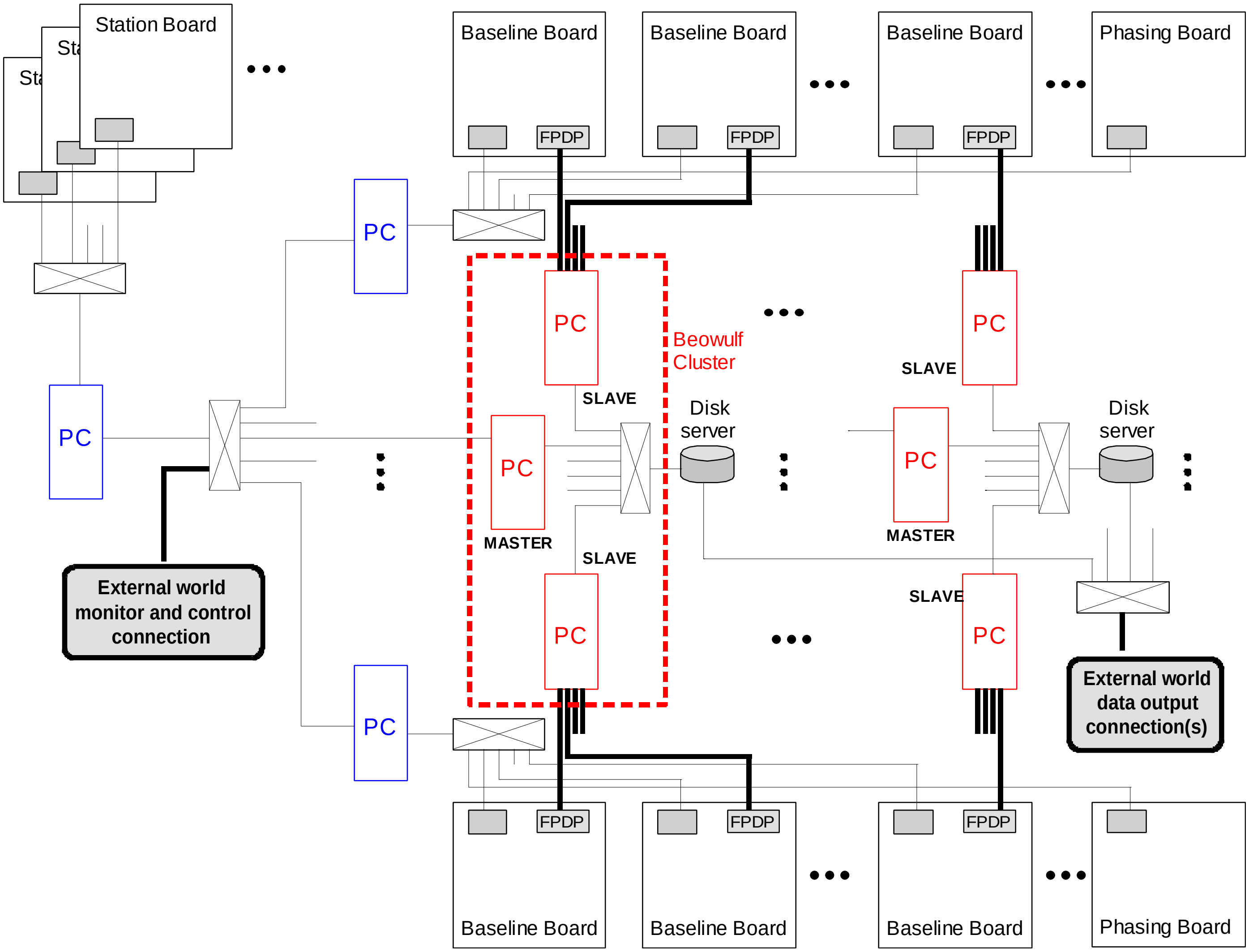


Figure 7-8 Straw-man correlator computing environment. COTS PC boxes are control computers and data processing computers. The data processing computers are arranged in Beowulf clusters so that each cluster gets data from Baseline Boards that process the same baselines. With this configuration there is no need for inter-cluster communication that could produce unacceptable bottlenecks in some correlator configurations. Performance is increased by increasing the *number of clusters* (i.e. each PC crunches data from fewer Baseline Boards), not the cluster size. The master PC in each cluster is used to obtain configuration information, not available on the FPDP interface, that is necessary for creating FITS file fragments.

NRC has budgeted to pay for COTS PC boxes as described, but NRAO has the final decision regarding the choice of the control and back-end computing hardware.

The correlator floor plans of Figure 7-5 and Figure 7-6, do not explicitly indicate where the COTS PC boxes of Figure 7-8 will be physically located. The few control computers could be located anywhere in the room including in a separate rack if rack-mount computers are used, or on a free standing shelf if desktop units are used. Because of cable length restrictions, the data processing computers that connect to Baseline Boards via the FPDP interface must be physically close to the Baseline Racks. It is possible that these could be in the racks themselves or in a rack or shelf beside the baseline racks as long as the location does not cause high-speed intra-cabinet cable access problems. Final details are TBD.

7.4.2 Software Configuration

The "correlator software" is roughly defined as any software that resides inside the boundaries shown in Figure 7-8. The external world has access points[40] for monitor and control of the correlator, and for data output as identified in the figure. These access points form an interface that has become known as the "Virtual Correlator Interface" (VCI). All of the details of how to configure the correlator and all of the real-time processing necessary to run the correlator lie below the VCI. The level of functionality, communications protocols, and internal details of the correlator software is TBD.

[40] There is another access point for remote power cycling independent of any correlator control computer (sections 3.10, 7.2).

8 References

[0] Carlson, B., A Proposed WIDAR Correlator for the Expansion Very Large Array Project: Discussion of Capabilities, Implementation, and Signal Processing, NRC-EVLA Memo# 001, May 18, 2000.

[1] Carlson, B., Requirements for 8-bit Processing in the Proposed WIDAR Correlator for the EVLA, NRC-EVLA Memo# 010, January 29, 2001.

[2] Carlson, B.R., Dewdney, P.E., Burgess, T.A., Casorso, R.V., Petrachenko, W.T., Cannon, W.H., The S2 VLBI Correlator: A Correlator for Space VLBI and Geodetic Signal Processing, Publications of the Astronomical Society of the Pacific, 1999, 111, 1025-1047.

[3] Carlson, B., Simulation Tests of Sub-Sample Delay Tracking in the Proposed WIDAR Correlator for the Expanded Very Large Array, NRC-EVLA Memo# 007, October 3, 2000.

[4] Carlson, B., Dewdney, P., Simulation Tests to Quantify the Spectral Dynamic Range and Narrowband Interference Robustness of the WIDAR Correlator for the EVLA, NRC-EVLA Memo# 009, November 1, 2000.

[5] Carlson, B., Two Correlators for the Price of One: How a VLBA Correlator Could Fit Within the Proposed 40-Station WIDAR EVLA Correlator, NRC-EVLA Memo# 006, September 28, 2000.

[6] Carlson, B., An Analysis of the Effects of Phase Dithering in a Lag-based Fringe-Stopping XF Correlator, NRC-EVLA Memo# 002, May 26, 2000.

[7] Whitney, A.R., 1982, The Mark III Correlator: Design, Operation, and Productivity in Proc. IAG Symp. 5, Geodetic Applications of Radio Interferometry (NOAA Tech. Rep. NOS 95) (Rockville: NOAA), 152.

[8] Carlson, B., A Closer Look at 2-Stage Digital Filtering in the Proposed WIDAR Correlator for the EVLA, NRC-EVLA Memo# 003, June 29, 2000.

[9] Carlson, B., Simulation Tests of Phasing Subsystem Signal Processing in the WIDAR Correlator for the EVLA, NRC-EVLA Memo# 008, November 7, 2000.

Index