REQUIREMENTS AND FUNCTIONAL SPECIFICATION

EVLA Correlator Chip

RFS Document: **A25082N0000**

Revision: 2.5

Brent Carlson, January 20, 2010

National Research Council Canada
Herzberg Institute of Astrophysics
Dominion Radio Astrophysical Observatory

P.O. Box 248, 717 White Lake Rd
Penticton, B.C., Canada
V2A 6K3

Table of Contents

List of Figures

List of Tables

1 Revision History

Revision	Date	Changes/Notes	Author
1.0	March 18, 2002	Initial Revision	B. Carlson
1.1	March 22, 2002	Remove input clock on the LTA Controller interface…only one hi-speed 128 MHz clock domain in the chip.	B. Carlson
1.2	July 26, 2002	Update interface specs to match actual design.	B. Carlson
1.3	August 28, 2003	Update interface specs to include the X and Y input buffers, and dual-clock (X-CLOCK and Y-CLOCK operation). Eliminate interface requirements for an FPGA prototype. Define I/O level requirements.	B. Carlson
1.4	November 20, 2003	Update the Performance Requirements section with maximum voltage and timing ratings, ESD requirements, overshoot and undershoot requirements, and MTBF requirements.	B. Carlson
1.5	November 25, 2003	Change the correlator chip output data frame to have an ending CHECKSUM rather than a Parity Check. A checksum is a more robust error checking mechanism.	B. Carlson
2.0	September 9, 2004	Add daisy-chained outputs. All I/O are LVTTL. Fix block diagrams. Fix functional, performance, and environmental specifications. Fix functional specifications section to more accurately reflect implementation.	B. Carlson
2.1	September 27, 2004	Take into account the tco delay on the daisy-chained outputs	B. Carlson
2.2	October 7, 2004	Add missing RESET_ line, JTAG ports, JTAG standard, and minor requirement oversights.	B. Carlson
2.3	October 29, 2004	Add autocorrelator mode, with additional internal X-6 and X-7 routing. No pin/timing/register set interface changes. Minor internal changes to the chip. Fix Table 5-3 Y-input bugs.	B. Carlson

2.4	February 15, 2005	X and Y input and output clocks are now 32 MHz. Add TESTMODE pins, PLL reset, and fix some minor description errors.	B. Carlson
2.5	January 10, 2010	Add appendices to include chip pin-outs, AC/DC characteristics, timing, physical dimensions, and known bugs+workarounds.	B. Carlson

2 Introduction

This document describes detailed requirements and design concepts for the EVLA correlator chip. The correlator chip is the heart of the correlator, and the performance and functional capabilities of the chip largely determine the capabilities, and ultimately the science output of the correlator system.

Background information on the correlator chip within the "WIDAR" correlator design concept can be found in NRC-EVLA Memo# 014 (EVLA Memo 31), "Refined EVLA WIDAR Correlator Architecture". This document has many similarities to the correlator chip described in Memo# 014, but the "VLBI delay" function has been eliminated since this function can more effectively and easily be implemented in the FIR filter chip. Also, some implementation concepts have changed, and this document presents the concepts in considerably more detail than outlined in the memo.

The development plan for the correlator chip is as follows:

1. Develop and test the full correlator chip functionality. A sophisticated test bench is required to allow testing of the chip in many dozens of configurations.

2. Implement the correlator chip in an ASIC device.

3. Test the prototype correlator chips on-the-sky using a fully-populated Baseline Board to verify that all of the performance and functional requirements have been met.

More details of the plan for testing the correlator chip can be found in [1].

3 Context

Correlator chips reside on the Baseline Board in an 8 x 8 array of 64 devices. The chips are fed with 'X' and 'Y' station (i.e. antenna-based) data from X and Y Recirculation Controller FPGAs. X and Y station data is transmitted in a daisy-chain fashion, point to point to each correlator chip. The simplified block diagram shown in Figure 3-1, illustrates how this is done.

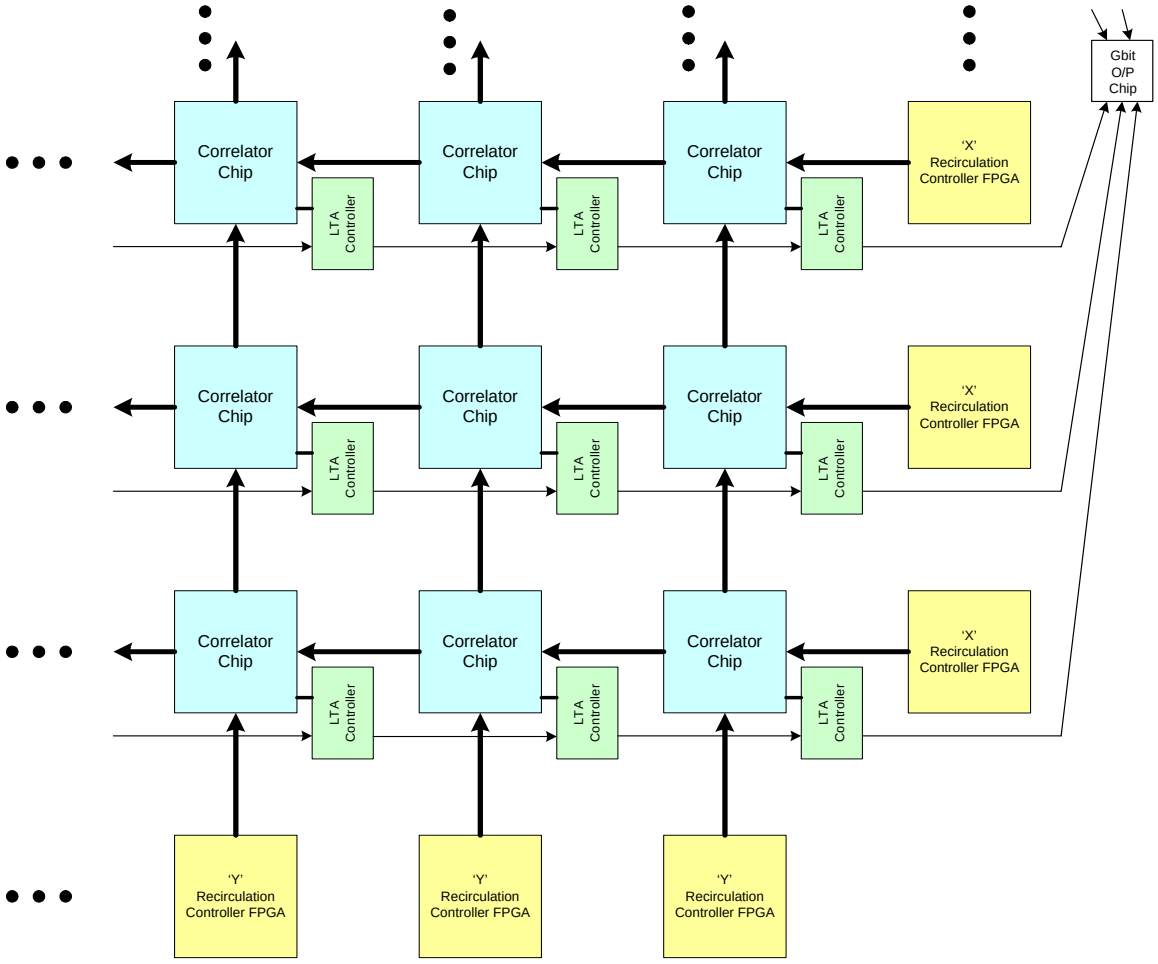


Figure 3-1 Simplified block diagram of the correlator chip environment. The correlator chip is fed data from X and Y Recirculation Controllers, and outputs the data to a dedicated LTA Controller FPGA. There is also an 8-bit synchronous microprocessor interface (not shown) for configuration and status monitoring.

The data and clock phases from each Recirculation Controller are arbitrary such that, at any given correlator chip, the X input signals and the Y input signals are "self-

synchronous"[1] but are not phase aligned w.r.t. each other. Final phase (and delay) alignment to one clock domain is performed in the correlator chip in a buffer that is sufficient in size to compensate for all X vs Y signal and clock skews that may be encountered, even in a large system. The buffer is capable of compensating for ±30 samples of delay at 256 Msamples/sec—enough to absorb 10 m of cable length mismatch and varying delays on each board to a particular correlator chip. The synchronization mechanism is automatic and requires no microprocessor interaction. This method greatly simplifies clock distribution and synchronization on the Baseline Board and in the entire system[2].

Since there will be 64 correlator chips on a board it is necessary to restrict the size and power of each chip to some reasonable values. Thermal analysis [2] indicates that a power dissipation of 6 W per chip can be dealt with, but that it is desirable to keep the correlator chip power dissipation below 4 W for simplicity of thermal management.

Additionally, it is essential to keep the correlator chip within a maximum 31 x 31 mm package using a BGA to allow for decoupling capacitors for chips on both sides of the board. A smaller package—27x27 mm—is highly desirable.

[1] i.e. the signals are synchronous with their associated clock, with required setup and hold times. No provision is made in the chip to correct for invalid setup and/or hold times of the X signals w.r.t. the X clock, or the Y signals w.r.t. the Y clock.

[2] E.g. cable lengths do not have to be matched etc.

4 Overview

A simplified block diagram of the correlator chip is shown in Figure 4-1.

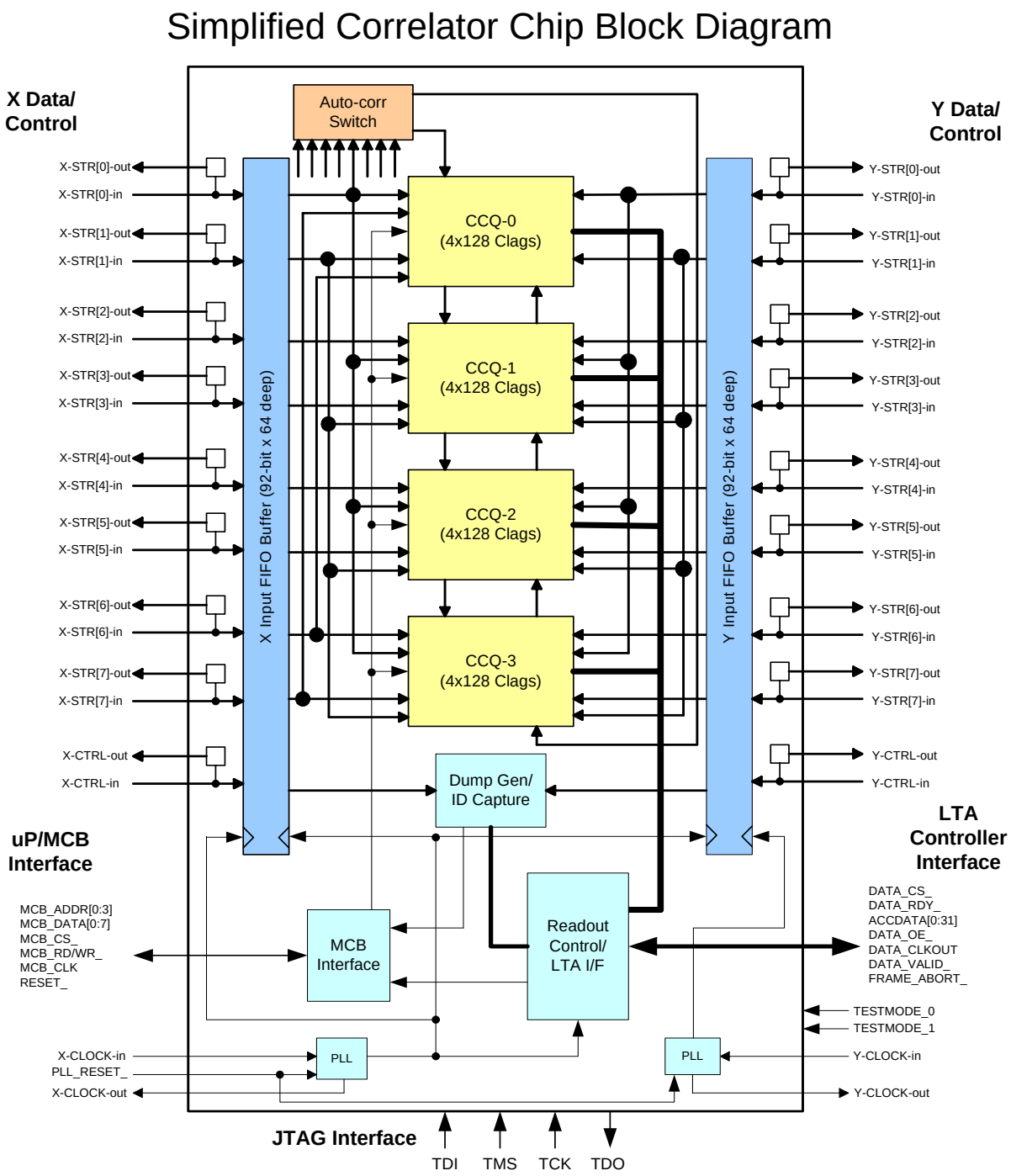


Figure 4-1 Simplified correlator chip block diagram including input synchronization buffers and daisy-chained outputs for connection to the next chip. There are **429 I/O signals**, including the 4 JTAG interface signals.

The chip consists of 4 "**CCQ**s" (Correlator Chip Quads), each of which contains 4 independent 128 Complex-lag Correlator Cells (**CCC**s). The chip is arranged this way so that each CCQ can correlate all 4 polarization products for an R/L pair of sampled data streams. In addition, adjacent CCCs within a CCQ can be concatenated and adjacent CCQs within the chip can be concatenated so that more lags can be devoted to a particular cross-correlation, at the expense of not being able to correlate as many sampled data streams.

There are 8 'X' input data streams, 8 'Y' input data streams, and corresponding outputs for daisy-chaining to the next correlator chip. Each stream consists of 4-bit sampled data, 4 bits of phase, a data valid qualifier, and a shift enable clock. The shift enable clocks control the shifting of data through the correlator delay lines to allow sampled data streams with different bandwidths to be correlated with the use of the same chip clock.

The input streams contain embedded station, baseband, and sub-band identifier information and, along with some additional real-time control signals, prompt the chip to dump data and transmit a data frame containing all necessary information to an external **LTA** (Long-Term Accumulator) controller without the need for microprocessor intervention. The LTA Controller performs any further integration.

The **MCB** (Monitor and Control Bus) Interface is used only to allow a microprocessor to configure the chip and obtain status information. It is not used for coefficient readout.

Each CCQ has access to its own input data and data from the master CCQ (CCQ-0) so that all correlations can be performed in hybrid-antenna configurations [3].

There is an **autocorrelator mode**, where *all* Y inputs are completely ignored. In this mode, all CCCs in the chip are chained together and any one of the 8 X-input data streams can be autocorrelated. To allow for "one-sided" lag correlation, and/or concatenation of longer lag chains, the autocorrelation data inserted into the X delay line (CCC-0) sources from the X-**SDATA**n input, and the data inserted into the Y delay line (CCC-15) sources from the X-**PHASE**n input. This allows a column of 8 correlator chips to independently autocorrelate all of the 8 data streams.

All chip I/O signals are 2.5 V LVTTL. All output drivers must be capable of driving one LVTTL load over a maximum of 2" of transmission line with no terminations. The only exception to this is that the MCB_DATA[0:7] drivers as noted in section 4.5.

A brief description of each I/O signal is as follows. A more detailed description is contained in the following sections.

4.1 X, Y Interface Input Signals (X-STR-in, Y-STR-in)

This section describes signals labeled generically as X-STR[*]-in and Y-STR[*]-in in Figure 4-1

- X-SDATA[0][0:3]...X-SDATA[7][0:3]. Eight X-input 4-bit sampled data streams. These inputs contain sampled Gaussian data. Normally, each bit will be undergoing transitions on average, 50% of the time.

- X-PHASE[0][0:3]...X-PHASE[7][0:3]. Eight X-input 4-bit phase streams. Phase data is used to remove any remaining frequency or phase offsets in the data. These inputs will normally be counting up, at a relatively slow rate and so their transition percentage is very low (<1%). When in autocorrelator mode, PHASE inputs must contain data, and where a sample is invalid, the data must be set to the –8 state (1000b).

- X-DVALID[0:7]. A data valid bit for each X input data stream. If low, the associated sample is invalid and is not correlated. Normally, these inputs are mostly high.

- X-SE_CLK[0:7]. A shift enable clock for each X input data stream. In the worst case, these inputs will be changing once every clock cycle.

There is an identical set of Y-station input signals to the chip which are synchronous with the Y-CLOCK, except in autocorrelator mode, these inputs are ignored. The delay between the arrival of the X inputs and the Y inputs can be +/-30 bits at 256 Mbits/sec, or a little less than +/-120 nsec. This is able to make up for about +/-10 m of cable length mismatch in the system (refer to Figure 5-1), and chip-to-chip delays from daisy-chaining. The X-CLOCK and Y-CLOCK can have an arbitrary phase w.r.t. each other.

4.2 X, Y Interface Control Input Signals (X-CTRL-in, Y-CTRL-in)

This section describes signals labeled generically as X-CTRL[*]-in and Y-CTRL[*]-in in Figure 4-1

- X-SCHID_FRAME_ - Active low signal that is a pulsed framing signal for the above streams. The X-SDATA streams contain embedded identifier information synchronized to this pulse.

- X-DUMP_SYNC, X-DUMP_EN[0:7] – X-station originating dump control signals.

- X-TIMESTAMP – X-station timestamp information passed on by the correlator chip to the LTA Controller via the output data frame.

- X-CLOCK – This is a 32 MHz clock that, on the rising and falling edges, is mid-cell with the 256 Mbit/sec inputs (refer to Figure 5-1). This clock is used to develop (with a PLL) phase-synchronous internal 128 MHz and 256 MHz clocks with coincident rising edges.

There is an identical set of Y-station input signals to the chip which are synchronous with the Y-CLOCK. The delay between the arrival of the X inputs and the Y inputs (i.e. the skew between input SCHID_FRAME_ pulses) can be +/-30 bits at 256 Mbits/sec, or a little less than +/-120 nsec. This is able to make up for about +/-10 m of cable length mismatch in the system (refer to Figure 5-1 and section 4.3), and chip-to-chip delays from daisy-chaining. The X-CLOCK and Y-CLOCK have an arbitrary phase w.r.t. each other.

4.3 Daisy-Chained Outputs

This section describes signals labeled generically as X-STR[*]-out, X-CTRL[*]-out, and the associated Y counterparts in Figure 4-1. These signals are the "-in" signals regenerated by the chip and incur a 1.5 clock-cycle delay (not including the t_{co} delay of about 2 nsec) at 256 MHz from input to output (i.e. 5.86 nsec). These output signals must have exactly the same timing relationship *relative to each other* as the input signals do according to Figure 5-1 so that the next correlator chip to receive these signals in the array will function properly. The X-CLOCK_out signal is the regenerated clock X-CLOCK_in after the PLL, and is at 32 MHz. **These clocks do not have a specific phase requirement, other than rising edges must be mid-cell with the data as shown in Figure 5-1 and Figure 5-2**. It too must have its rising and falling edges mid-cell with the data. Refer to Figure 5-2 for a functional timing diagram relating the input signals to the outputs.

The "-out" clocks must be mid-cell with the data according to Figure 5-1, and the PLL in the chip must regenerate the X and Y clocks independently and remove enough jitter such that jitter does not accumulate when 8 chips are daisy-chained (Figure 3-1). On all outputs, as on the inputs, there is an arbitrary phase relationship between the X and Y clocks and the X and Y signals.

All daisy-chained outputs have a "_O" suffix added to the signal names. For example, for input X-SDATA[0][0:3], the corresponding daisy-chained output is X-SDATA_O[0][0:3].

(It is worth noting that the total worst-case delay-skew across the circuit board due to daisy-chaining 8 chips is 7X the daisy-chaining delay—10.5 clock cycles at 256 MHz or 41 nsec—7X the chip-to-chip PCB trace delay (7x2"~=7x175 psec/in = 1.225 nsec), and 7X the t_{co} delay (estimate 2 nsec). This leaves enough in the 64-deep input FIFO buffers to compensate for ~15 clock cycles or 60 nsec. At a velocity of propagation along the cable of 4.43[3] nsec/m (1.36 nsec/ft), a cable length mismatch of 13.5 m can be tolerated, ignoring delay mismatches due to connector-to-Recirculation Controller routing on the Baseline Board. This leaves ample margin over the 10 m requirement.)

[3] Obtained from Meritec test results web site: http://www.meritec.com/Pages/pdf/testrpthpm5.pdf. This is the cable we intend to use in the correlator system.

4.4 LTA Controller Interface I/O Signals

This section describes LTA Controller Interface signals indicated in Figure 4-1.

- DATA_CS_ - Active low input chip select that enables the correlator chip LTA Controller interface drivers[4]. All LTA interface output signals except DATA_CLKOUT are high-impedance when this signal is not asserted (i.e. high).

- DATA_RDY_ - Output driven low by the correlator chip to indicate that data in an output buffer is ready for transmission. This signal stays low for the duration of one complete lag frame, and can thus be used as a framing signal by external logic.

- ACCDATA[0:31] – Correlator chip output accumulator data bus. Bit 0 is the least-significant bit. Data is only valid on these lines when DATA_VALID_ is low.

- DATA_OE_ - If driven low when DATA_RDY_ is asserted, this input enables the correlator chip to start driving data on the ACCDATA bus. This signal is sampled on the rising edge of DATA_CLKOUT by the correlator chip.

- DATA_CLKOUT – 128 MHz output clock for this interface. All signals on the LTA Controller interface are synchronous with this clock, and the output signals of the correlator chip change a t_{cod} (clock-to-output delay) time after the rising edge of this clock.

- DATA_VALID_ - Output that is driven low to indicate that ACCDATA contains valid data. The LTA Controller should only clock in data when this is low.

- FRAME_ABORT_ - Input, that if asserted during transmission of any ACCDATA word, will cause the correlator chip to terminate transmission of the current frame, and clear the current frame buffer.

4.5 MCB Interface I/O Signals

This section describes the signals use for microprocessor monitor and control of the correlator chip (the MCB—Monitor and Control Bus).

- MCB_ADDR[0:3] – Microprocessor address for accessing internal correlator chip configuration and status registers.

[4] This chip select is provided because it may be necessary for cost reasons to have an LTA Controller connect to more than one correlator chip.

- MCB_DATA[0:7] – Bi-directional 8-bit microprocessor data bus. This is high-impedance if MCB_CS_ is high or if MCB_CS_ and MCB_RD/WR_ are both low. The MCB_DATA output drivers should have a drive strength of at least 12 mA and have a slew-rate controlled rise time of ~5 nsec (if possible). Any required series or parallel impedances are provided external to the chip.

- MCB_CS_ - Input chip select that enables the MCB interface logic and possibly MCB_DATA output drivers. The chip will not respond to any signals on the MCB unless this input is low.

- MCB_RD/WR_ - Input that when low (along with MCB_CS_ low) enables writing data into the correlator chip. Otherwise, data is driven onto the MCB_DATA bus by the chip, when MCB_CS_ is low.

- MCB_CLK – Input clock for the synchronous MCB interface. The phase and frequency of this input is independent of X-CLOCK and Y-CLOCK.

- RESET_ – This is an active-low signal that, when asserted, performs a chip-wide asynchronous reset. This reset does not affect the PLLs or the daisy-chained signals, and so asserting it should not affect any chips further down the daisy-chain. This must be held low for at least 0.5 milliseconds after PLL_RESET_ is released.

4.6 JTAG Interface

This interface consists of 4 signals, conforming to IEEE-1149.1. For more detailed information on these signals, refer to the IEEE-1149.1 standard.

- TDI – Data input.

- TDO – Data output.

- TCK – Clock input.

- TMS – Mode select input.

4.7 Miscl Signals

- PLL_RESET_ — This active-low signal resets the PLLs on the chip. This signal should be released at least 0.5 milliseconds before RESET_ is released to ensure the PLL is locked and stable before the chip starts up.

- TESTMODE_0, TESTMODE_1 – These inputs are used for putting the chip in internal test modes. For normal operation, both of these inputs must be held low. Other test modes are as follows:

- TESTMODE_1=0, TESTMODE_0=1 – scan-insertion test for mfg test.
- TESTMODE_1=1, TESTMODE_0=0 – split accumulator test for gate-level sim.
- TESTMODE_1=1, TESTMODE_0=1 – Bypass PLL. The internal clocks are equal to the X-CLOCK_in and Y-CLOCK_in inputs.

5 Requirements

The following is a list of correlator chip requirements.

5.1 Functional Requirements

1. Total of 2048 complex-lags arranged as 16 independent 128 complex-lag cross-correlator cells (**CCC**s). Each CCC will simultaneously correlate "lead" and "lag" cross-correlations, with lag numbering from 0...N-1, and the center lag at lag N/2. Refer to Figure 6-1 for CCC lag layout. Adjacent CCCs within the chip can be concatenated to form longer lag correlations, at the expense of less total correlated bandwidth.
2. Four CCCs are logically arranged into **CCQ**s ("Correlator Chip Quads"). Each CCQ can correlate all 4 polarization products for a sampled data stream pair.
3. Each CCQ has access to a sampled data stream pair from the 'X' input and the 'Y' input. Additionally, each CCQ has secondary access to the "master" CCQ input data, referred to as CCQ-0. This access allows efficient mixed-bandwidth correlation between different antennas [3].
4. It shall be possible to perform mixed bandwidth cross-correlations with the use of "shift enable clocks", thus keeping the correlator clock at a constant rate for all correlated bandwidths. The shift enable clocks are clock enable inputs to the lag delay-line flip-flops. CCCs that are concatenated must use the same shift enable clocks, and shift enable clocks must be properly pipelined with the data until reaching the CCC.
5. Except for concatenation, each CCC will operate independently.
6. A sampled data stream consists of: a 4-bit, 2's complement data word taking on values (–8, -7, -6, ..., -1, 0, +1, +2, ... +7); a 4-bit phase word taking on values (0, 1, 2,...,14, 15), representing quantized phase from 0 to 360^{o}; and a data valid bit that when low inhibits correlation by effectively forcing the data word to zero.
7. At each complex lag, 4-bit data is multiplied and then complex phase-rotated with 3-level sine and cosine functions. All products in the 4-bit multiplication result must be retained so that it is possible to perform >4-bit correlations using distributed arithmetic. Sine and cosine values are computed on the difference between the X and Y phase, modulo 16 (i.e. $\phi = (\phi_x - \phi_y)\%16$). The **sine** function shall have values (0,1,1,1,1,1,1,0,0,-1,-1,-1,-1,-1,-1,0) and the **cosine** function shall have values (1,1,1,0,0,-1,-1,-1,-1,-1,-1,0,0,1,1,1) for each of the 16 phase states (0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15) respectively. These are well-known phase rotation functions in radio astronomy [4][5].
8. Each CCC will contain two data-valid counters. These counters shall increment every time the data valid bit from X and Y are both asserted (high). The counters shall be located at lag N/2, and lag 0 (see Figure 6-1). These counters will be transmitted as part of the CCC's output data frame.

9. The accumulators will each be of sufficient length to enable correlation for a minimum of 500 microseconds at a 256 MHz clock rate without overflow (this is an accumulator with a minimum total length of 23 bits). There shall be no truncation of any accumulator bits, data valid counter bits, or bias accumulator bits during accumulation or readout. Each accumulator within a CCC will accumulate data for *precisely* the same time window.
10. Normally, a bias is added to the data before accumulation so that the bulk of the accumulator can be a ripple counter or incrementer. Thus, each CCC will contain a bias accumulator that calculates the *precise* bias that is contained in the lag data for that CCC. This bias value shall be transmitted with the CCC data frame for removal by off-chip hardware.
11. The maximum correlation dead time during dumping of accumulators from active registers to internal storage registers shall be 1 microsecond. Less dead time, if possible, is desirable.
12. All integration, timestamping, and recirculation control will happen with the use of externally provided hardware control signals and shall not depend on any microprocessor interaction. A microprocessor provides configuration and non-real-time status information functions only.
13. A “correlation holdoff” function shall be provided to prevent correlation between non-time-contiguous chunks of data when using all or part of the chip for recirculation. This is a hardware signal provided to the correlator chip, via the X/Y input paths.
14. The X and Y input data contains “station ID”, “baseband ID”, and “sub-band ID” identifiers. These identifiers will be captured and transmitted with each CCC output data frame.
15. The correlator chip will pass on timestamp, ID, and recirculation control information via the CCC’s transmitted data frame. Except for passing on information, the correlator chip has no knowledge of recirculation.
16. Each CCC will detect accumulator overflow and data overrun (meaning a dump request before data has been cleared from the on-board storage registers) conditions and provide this status information with the CCC output data frame. Overrun conditions must be handled so that no un-flagged, corrupted data is produced.
17. A test vector receiver for all X and Y input data and control signals shall be invoked via the microprocessor interface. This receiver will be able to compare its internally generated test vectors with received test vectors to detect errors. The test vector sequence will restart coincident with the X-SCHID_FRAME_ and Y-SCHID_FRAME_ input frame pulses (and restart with X/Y DUMP_SYNC for DUMP_EN and TIMESTAMP). The error detector will indicate a pass or fail condition on each input, and this condition is read and reset via the microprocessor interface. The test vector sequence will be defined later in this specification.

18. While on-line, the correlator chip shall check for synchronization using the structure of the input signals. When an inconsistency is detected, it must be possible for the error detector to set error bits inside the chip. These error bits can be read and cleared via the microprocessor interface.
19. The LTA Controller interface will be tested by virtue of signals embedded in the output data frame.
20. The MCB/microprocessor interface shall be tested by virtue of the microprocessor's ability to correctly write and read configuration data to/from the chip.
21. All configuration registers will be read/write, and all error status registers shall be read-only, with error bits being cleared on read.
22. The chip must contain on-board X and Y input buffers that accept X and Y input clocks and data that have an arbitrary phase and delay w.r.t. each other. The buffer length must be a minimum 64 words deep, allowing for up to +/-30 bits (at 256 MHz) of skew in the arrival time of data and timing synchronization pulses. Longer depths are permitted. The algorithm for operation of these buffers is as follows:

    X-memory operation:
    1. Write data into the X memory synchronous with the X-CLOCK but at 256 MHz.
    2. When the synchronization pulse (X-SCHID_FRAME_) is asserted, ALWAYS reset the write address counter. This means that this pulse is stored at memory address 0.
    3. The X read address counter is the write counter + (or -) 1/2 the buffer depth. When the read address counter is zero the X-SCHID_FRAME_ pulse out of the buffer should be asserted because of condition 2.

    Y-memory operation:
    1. Write data into the Y memory synchronous with the Y-CLOCK but at 256 MHz.
    2. When Y-SCHID_FRAME_ is asserted, ALWAYS reset the write address counter. This means that this pulse is stored at memory address 0.
    3. The Y read address counter is equal to the X read address counter and is driven synchronous with X-CLOCK at 256 MHz.
23. The chip must contain a Boundary Scan, JTAG interface.
24. The chip must re-generate the X and Y input signals and clocks for daisy-chaining to the next chip in the array. The delay from input to output is 1.5 clock cycles at 256 MHz clock rate, and the relative phase relationships on these outputs are the same as the inputs. The re-generated clocks must attenuate jitter so that jitter does not accumulate when going from chip-to-chip across the board.
25. The chip must implement an autocorrelator mode where the Y inputs are completely ignored and the X DATA and X PHASE are correlated in a concatenated lag chain.

5.2 Performance Requirements

1. The chip will develop its own internal 128 MHz and 256 MHz clocks (using PLLs) from the X and Y 128 MHz input clocks. The input 128 MHz clocks will have a maximum cycle-to-cycle jitter of +/- 150 psec. The input 128 MHz clock is rising/falling-edge mid-cell with the 256 Mbit/sec data (refer to Figure 5-1), and phased so that there is a 5.8 nsec setup time and 1.9 nsec hold time for the 128 Mbit/sec inputs (i.e. "3/4ths cell").

2. All X and Y inputs have a minimum setup time of 1 nsec and a maximum hold time requirement of 1 nsec, relative to the respective 128 MHz clock edge.

3. The 128 MHz and 256 MHz internal clocks are rising-edge phase synchronous with each other and phase-locked to the X-CLOCK input. These clocks must be developed internally in the chip using a PLL. Thus, the X-CLOCK input is the master clock.

4. The output X and Y 128 MHz clocks are phase synchronous with their respective inputs according to Figure 5-2. Output cycle-to-cycle jitter must be less than +/- 150 psec. A larger value is allowed if the correlator chip PLLs can tolerate more input jitter, and jitter does not accumulate beyond tolerable range in the daisy-chain configuration of Figure 3-1.
5. It must be possible to operate the chip within the range of 128 MHz +/-1%.
6. Internally, data will be read out of the active accumulators into the storage buffers at a 256 MHz clock rate that is frequency and phase synchronous with the X-CLOCK input.
7. Data transmitted to the LTA Controller will be phase synchronous with the DATA_CLKOUT output. There is a t_{co} (clock-to-output delay) time from the rising edge of DATA_CLKOUT until the output data changes. t_{co} must be in the range of $0 \text{ nsec} < t_{co} < 5.8 \text{ nsec}$. Minimum setup and hold time requirements are 1 nsec for inputs on this interface.
8. The synchronous MCB interface shall be capable of operating with a clock that is neither frequency nor phase synchronous with the 128 MHz clocks. The chip will support an MCB interface clock with a maximum rate of 33 MHz. The minimum t_{su} is 5 nsec, and the minimum required t_{hold} is 2 nsec on inputs, relative to the rising edge of MCB_CLK. For MCB_DATA outputs, t_{co} must be in the range of: $0 \text{ nsec} < t_{co} < 10 \text{ nsec}$.
9. The **absolute maximum** t_{co} (clock to output delay) for the X/Y inputs to the daisy-chained outputs is 4.2 nsec, once functional pipeline delay has been factored out. The desired t_{co}, and one that it is believed can be achieved is 2 nsec. Thus, the **absolute maximum** pin-to-pin X/Y input to daisy-chained output delay, including functional pipeline delay, is 10 nsec.

10. The absolute maximum power dissipation of the chip, running at a 256 MHz clock rate with 4-bit Gaussian-sampled data (containing 50% transitions), phase that is changing at a rate not exceeding 1 Mcycles/sec, and with all data being dumped and transmitted via the LTA Controller interface every 200 microseconds is 6.0 W. However, the goal for the maximum power dissipation under these conditions is 3.5 W [2].
11. All I/O are 2.5 V LVTTL. Drivers, except the MCB_DATA drivers must be capable of driving one LVTTL load over 2" of PCB trace with a nominal impedance of 50 ohm. The MCB_DATA output drivers must be 12 mA drivers, and have a transition time in the range of 1 to 5 nsec.
12. The chip can have multiple voltage supplies—one for the I/O standard and one for the core voltage. Nominally, the core voltage supply is 1.2 V, but there is no restriction on what this must be, as long as power dissipation requirements are met. An adequate number of power and ground pins must be provided for the expected number of voltages, inputs, SSOs, and power dissipation.
13. The absolute maximum DC input voltage relative to ground on any pin is –0.5V to Vcco+0.5V. Where Vcco is the "rail voltage" of the particular I/O standard.
14. The device must tolerate transient voltages (overshoot and undershoot) during transitions for durations of ≤ 10 nsec and current ≤ 100 mA of a minimum of –2V and a maximum of Vcco+1.0V, relative to ground.
15. The device must meet JEDEC standard JESD22-A114 (Mil Std 3015.7) for HBM ESD with full qualification at 1000 V minimum, or equivalent [6].
16. The required MTBF for the chip is $>10^7$ hours. The desired MTBF under normal operating conditions is $>10^8$ hours. MTBF estimations are largely dependent on calculation methods and testing methods, are statistical in nature, and subject to unknown estimation errors. An MTBF $<10^7$ may be acceptable, provided the calculation methods are well understood, quantified, and conservative in nature.
17. The chip technology must be resistant to "latch-up" and tested as per JEDEC standard No. 78 entitled "IC Latchup Test" [6].

5.3 Environmental Requirements

1. The correlator chip will be surface-mounted on the Baseline Board carrier board. A chip-array-wide heat spreader will be mechanically attached to the array of chips to reduce chip operating temperatures and temperature gradients. Thus, a chip package must be chosen to facilitate heat transfer to the heat sink.

2. The maximum outside dimensions of the chip package is 31 x 31 mm. Smaller dimensions (e.g. 27 x 27 mm in a 672-pin FBGA package) are desirable to ease routing congestion, reduce the array footprint, and reduce problems associated with double-sided surface mount BGA (i.e. more easily permit placement of decoupling capacitors for the correlator chip and any chips mounted on the reverse side of the board). Absolute maximum package height, including balls, is 5 mm.

3. The correlator chip I/O pin assignments are TBD. If possible, the pins should be arranged to facilitate easy PCB X/Y daisy chain signal routing in a matrix of correlator chips as shown in Figure 3-1. Further preliminary investigation may be required to determine I/O pin assignments.

4. For thermal performance, a metal, cavity-down flip-chip package is desired. In any case, the maximum junction-to-case thermal resistance requirement is 2 ^{o}C/W. The desired junction-to-case thermal resistance is <0.2^{o}C/W. A wire-bond package is also likely acceptable since there are no extreme pin-to-pad performance requirements. Pin-to-pad transmission line impedance is a free parameter, although it should nominally be 50 ohm.

5.4 Interface Requirements

The following sub-sections define the interface requirements of the correlator chip. Precise timing requirements are not defined, and except for input specifications, are largely a function of the correlator chip design.

5.4.1 X, Y Input Interface Requirements

The X, Y data interface requirements are shown in Figure 5-1. Many inputs contain embedded identifier and control bits and are defined as follows:

SID[0:7] – 8 bits of Station ID embedded in the 4-bit SDATA word. This ID is set on the Station Board that the sampled data stream originates from.

SBI[0:4] – 5-bit sub-band identifier embedded in the 4-bit SDATA word. This is normally the FIR filter number the data originated from on the Station Board.

BBI[0:2] – 3-bit baseband identifier embedded in the 4-bit SDATA word. This is the baseband number that the data originated from.

DC[0:2] – 3-bit dump control embedded in each of the 8 DUMP_EN lines that controls what the correlator chip, and what the LTA Controller does with the dumped data. These bits are defined in Table 5-1. .

Table 5-1 Dump control bits and associated action.

DC2	DC1	DC0	Action
0	0	0	First dump of data into LTA. Just save data in LTA bin.
0	0	1	Add to existing LTA data and save in LTA.
0	1	0	Last dump: add data to LTA, and flag data as ready for readout.
0	1	1	Speed dump: bypass LTA directly to output. The data bias is removed.
1	0	0	Correlator chip dumps, discards data, clears registers
1	0	1	Dump, save in LTA, flag data as ready.
1	1	X	Reserved.

Correlator Chip X, Y Inputs Functional Timing

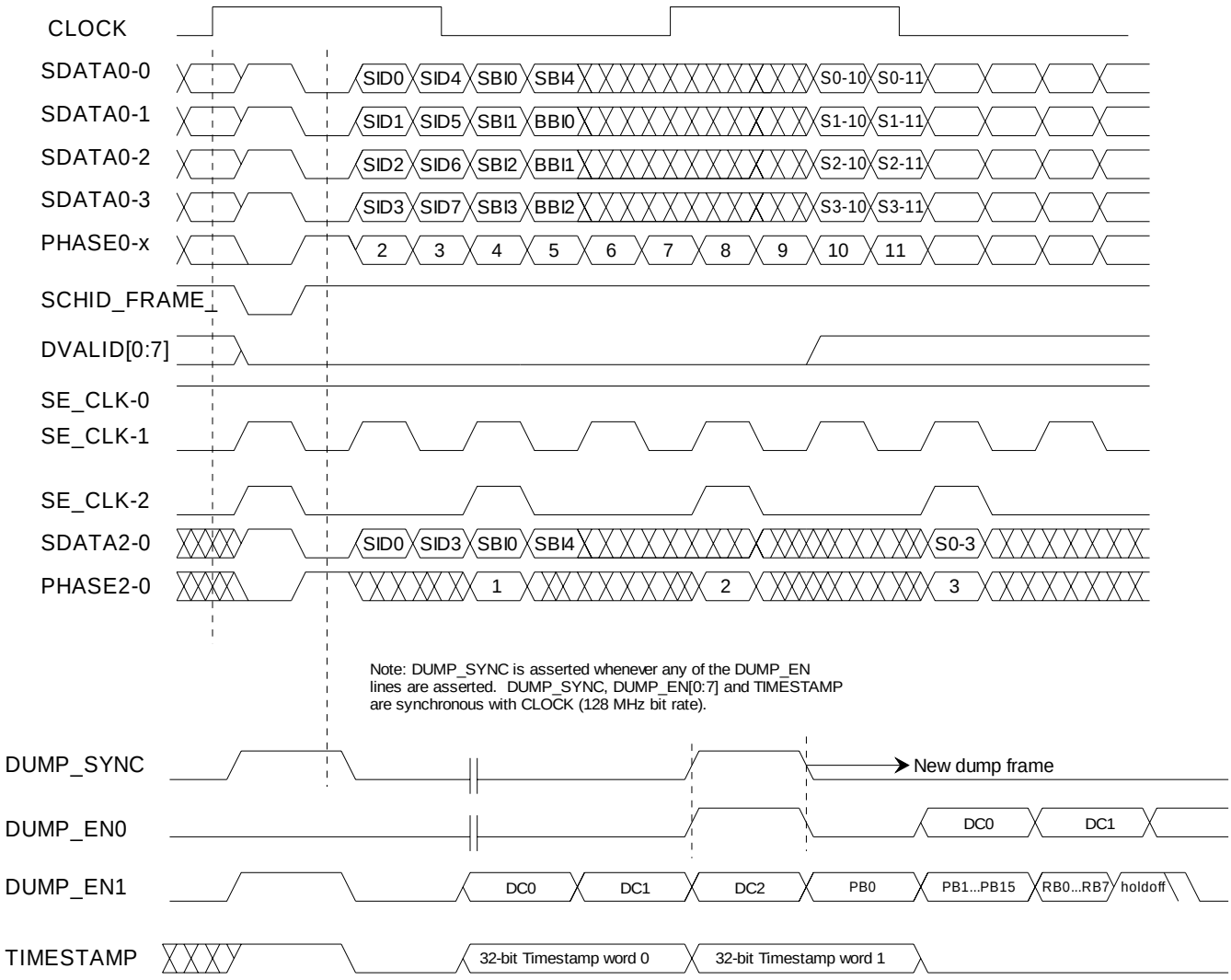


Figure 5-1 Correlator chip X, Y interface functional timing requirements. Embedded identifier and control bit information is as shown, and only representative signals are in the diagram. The long DATA_VALID negation after SCHID_FRAME_ is simply to show that SDATA and PHASE can be anything when DATA_VALID is low. Normally, DATA_VALID could be high immediately after the last embedded identifier in SDATA. Note that because of the length of TIMESTAMP, DUMP_SYNCs must be separated by at least 136 clock cycles (~1.0625 usec).

PB[0:15] – Phase bin number embedded in the DUMP_EN lines that the data gets dumped/accumulated into. PB15 is the bank number (0 or 1) (currently the LTA Controller only supports 1000 bins). With Speed dumping, all 16 PB bits can be used for a total of 65,536 bins. The correlator chip simply passes this information onto the external downstream LTA Controller.

RB[0:7] – These bits are embedded in the DUMP_EN lines and indicate the "Recirculation Block" that the data is part of. This information is simply passed on

by the correlator chip to the LTA Controller via the output data frame. Embedding these bits in DUMP_EN allows simultaneous operation and dumping of data in recirculation and non-recirculation modes.

Holdoff – After the DUMP_EN control bits are transmitted, Holdoff is held high to disable correlation until the lag shift registers fill up with data. There is no limit on the duration of the holdoff (controlled by the Recirculation Controller), and once negated correlation resumes. This is to ensure that when recirculation is active, non-contiguous chunks of data are not correlated.

TIMESTAMP – This input contains two embedded 32-bit timestamps that are passed on by the correlator chip to the LTA Controller via the output data frame. These bits are framed by DUMP_SYNC as shown in the figure.

As shown in Figure 5-1, DVALID is held low while the data lines contain embedded identifiers to prevent incorrect correlation. It is not necessary for the correlator chip to take any special action during this state, other than to not correlate.

Since the correlator chip can be dumped synchronous with (for example) a pulsar, DUMP_SYNC is generally asserted independent of SCHID_FRAME_.

The data, phase, data valid, and shift enable clocks take on defined states during and after SCHID_FRAME_. These states can be used to check for data stream errors and alignment during on-line operation. Similar well-defined states for DUMP_EN[0:7] and TIMESTAMP occur synchronized with DUMP_SYNC.

5.4.2 X/Y Daisy-chained Output Interface Requirements

These outputs are carbon-copies of the associated inputs, but are registered and regenerated by the chip for use by the next correlator chip in the daisy-chain. There is a 1.5 clock cycle delay at 256 MHz from the inputs to the daisy-chained outputs plus the clock-to-output delay (t_{co}). Figure 5-2 below indicates the timing relationship between the inputs and the daisy-chained outputs, not including the t_{co} delay. Note that CLOCK_O has a -90^{o} phase relative to the CLOCK input, not including the t_{co} delay.

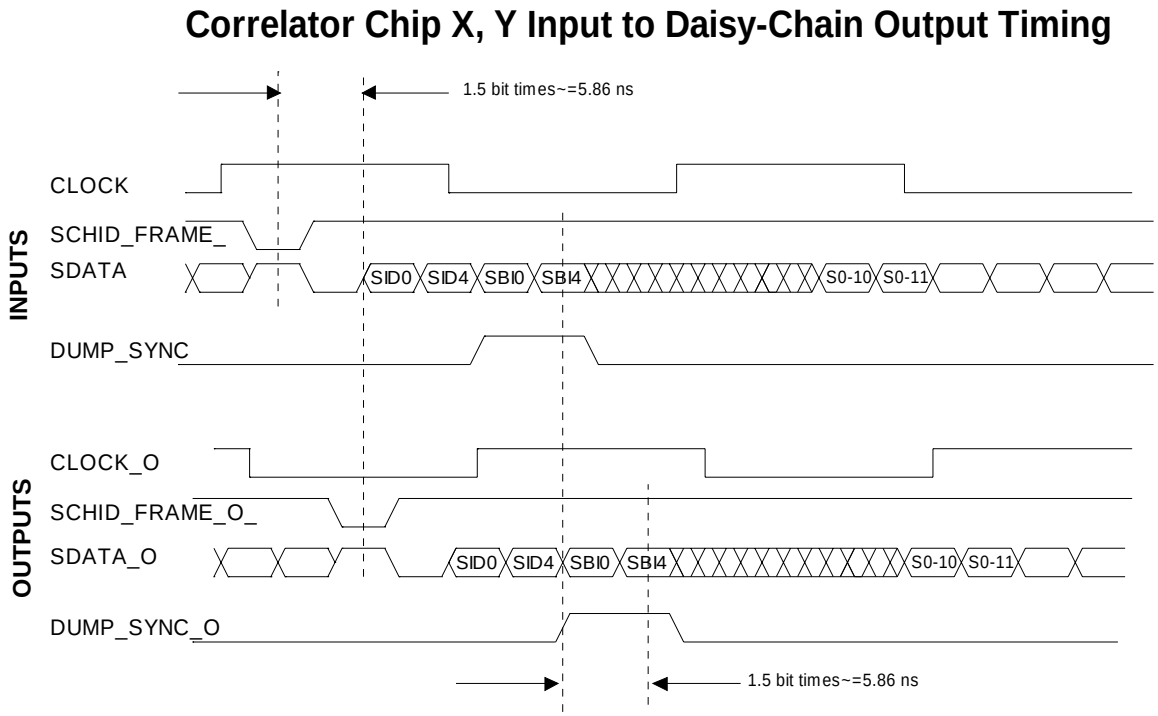


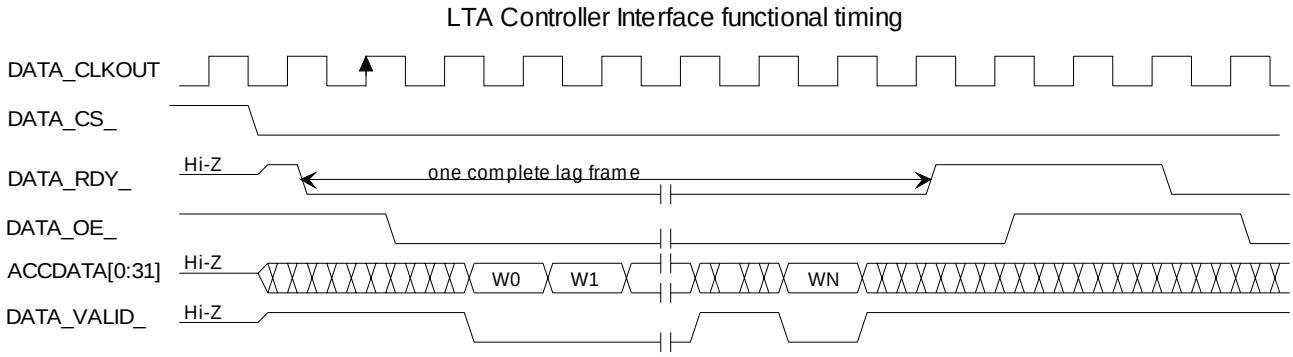
Figure 5-2 Functional timing diagram showing the phase relationships between the input X or Y signals and their associated daisy-chained output signals. All daisy-chained signals are delayed by the same amount—5.86 nsec (1.5 clock cycles at 256 MHz). This value does not include the absolute maximum t_{co} delay of 4.2 nsec, thus the maximum total input to output delay is 10 nsec. However, a maximum t_{co} of 2 nsec is desired.

5.4.3 LTA Controller Interface Requirements

The LTA Controller Interface (LCI) is the interface whereby integrated correlator chip data is transmitted to the external LTA Controller. Data is transmitted in frames, where each frame is the control, status, and data for one Correlator Chip Cell (CCC). Frames are transmitted in the same fashion for each CCC independent of whether or not CCCs are concatenated to form longer lag lengths.

Figure 5-3 is a functional timing requirements diagram for the LCI. It is a fully synchronous interface. DATA_CLKOUT is phase synchronous with the input X-CLOCK to the chip. *All* I/O signals on the LCI are synchronous with DATA_CLKOUT.

Not shown in the figure is FRAME_ABORT_. When FRAME_ABORT_ is asserted (low), the correlator chip negates DATA_RDY_ shortly after sampling FRAME_ABORT_. The current frame is then aborted and, if other frames are ready for transmission, DATA_RDY_ is asserted again. Assertions of FRAME_ABORT_ are only recognized by the correlator chip when DATA_RDY_ is asserted.



Note: DATA_CS_ 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 5-3 LTA Controller Interface (LCI) functional timing requirements. DATA_CLKOUT is phase synchronous with the X-CLOCK input to the correlator chip. All I/O timing is synchronous with DATA_CLKOUT.

The data frame transmitted by the correlator chip on ACCDATA[0:31] is as defined in Figure 5-4.

	28	24	20	16	12	8	4	0	
SYNC	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 0								W0
	STATUS BITS	Reserved		NUM_CLAGS		CCC#		Cmmd	W1
	BBID-Y	SBID-Y	SID-Y		BBID-X	SBID-X	SID-X		W2
	LTA (Phase) BIN				RECIRC_BLK-Y		RECIRC_BLK-X		W3
	TIMESTAMP-0								W4
	TIMESTAMP-1								W5
	DVCOUNT-Center								W6
	DVCOUNT-Edge								W7
	DATA_BIAS								W8
	Lag 0-In_phase accumulator								W9
	Lag 0-Quadrature accumulator								W10
	Lag 1-In_phase accumulator								W11
	Lag 1-Quadrature accumulator								W12
	● ● ●								W13-262
	Lag 127-In_phase accumulator								W263
	Lag 127-Quadrature accumulator								W264
SYNC	0 0 0 1 1 1 0 0 0 1 1 1 0 0 0 1 1 1 0 0 0 1 1 1 0 0 0 1 1 1 0 0								W265
	CHECKSUM								W266

Figure 5-4 Correlator chip data frame requirements. This frame is transmitted on the LCI to the LTA Controller. An independent frame is transmitted for each CCC, independent of any configured CCC concatenation.

The contents of the transmitted data frame of Figure 5-4 are defined as follows. All bit numbering is MSB…LSB, left to right in the figure.

Word 0 (W0)

Start sync word as defined in the figure.

Word 1 (W1)

Cmmd – These 3 bits contain the command word that tells the correlator chip what to do with the data frame. Bit definitions are identical to that shown in Table 5-1, except that the (1,x,x) state is reserved.

CCC# - The Correlator Chip Cell (0…15) that this data frame came from. This is a 6-bit word to support up to 64 CCCs.

NUM_CLAGS – The number of complex lags in the data frame. Normally this is 128, but could be more *if* the final correlator chip has more lags/CCC.

Reserved – Unused bits, set to 0. These bits may contain additional useful information if warranted in the design.

STATUS BITS:

OVR (b25) – Set (1) if a dump overrun has occurred. This frame is ok, but one or more *previous* frames had to be discarded due to a buffer overrun (tried to dump data, but the output buffer was in use).

ACC_OV (b26) – Set (1) if one or more accumulators in this frame overflowed.

XSyncerr (b27) – Set (1) if an error was detected on one or more X input data, phase, or control streams that affects this CCC. This status is reset every frame, and applies to the integration time the frame is applicable to. Similar status information is provided through the MCB interface, but will be cleared using independent mechanisms.

YSyncerr (b28) – Same as XSyncerr, only it is for the Y input streams.

X_CCC_input (b29) – Set (1) if the X input to the CCC is connected to the adjacent CCC. Reset (0) if the X input is not connected to the adjacent CCC (i.e. it is connected to chip input data).

Y_CCC_input (b30) – Same as X_CCC_input, only this applies to the CCC's Y input.

FPGA/ASIC (b31) **–** Reset (0) if this is an FPGA data frame where only the first48 lags are unique, the rest repeated. Set (1) if NUM_CLAGS is unique because it is the full ASIC implementation. Normally, this is always set (1).

NOTE: X_CCC_input and Y_CCC_input are configuration information and, strictly speaking, not necessary. However, these bits, and recirculation block identifiers in W3 allow CCC frames to be unambiguously concatenated together by downstream computers.

Word 2 (W2)

SID-X, **SID-Y** – 8-bit station IDs that identify the X and Y Station Boards this data came from.

SBID-X, SBID-Y – 5-bit sub-band IDs that identify the X and Y sub-bands this data came from. This identifies the FIR filter the data came from. These identifiers can take on the range (0…17).

BBID-X, BBID-Y – 3-bit baseband IDs that identify the baseband input (0…7) that this data came from. There are 2 baseband inputs to each Station Board, so these uniquely identify the 8 baseband inputs to a group of 4 Station Boards plugged into the same Sub-band Distributor Backplane.

Word 3 (W3)

RECIRC_BLK-X, **RECIRC_BLK-Y** – 8-bit recirculation block identifiers. This is information supplied to the correlator chip on the DUMP_ENx inputs.

LTA (PHASE) BIN – The LTA bin number that indicates where this data is to be stored/accumulated. This data is directly derived from the DUMP_ENx input. Note that only one number is required here because each CCC selects exactly one DUMP_EN input from dump control.

Word 4, 5 (W4, W5))

TIMESTAMP-0, **TIMESTAMP-1** – Two, 32-bit timestamps derived from the X or Y input TIMESTAMP that is the same as the X or Y DUMP_EN selected for this CCC (i.e. these come from the X or Y inputs, depending on whether dumping is selected from the X or Y DUMP_EN). TIMESTAMP-0 is the 32-bit "COUNTPPS" extracted from the system TIMECODE. TIMESTAMP-1 B0-B28 is the count of the number of 128 MHz clock cycles since the last 1 PPS, and could be greater than 128,000,000 but is always self-consistent with TIMESTAMP-0. TIMESTAMP-1 B29-B31 is the "epoch" extracted from the system TIMECODE.

Word (W6)

DVCOUNT-Center – The data valid counter for the center of this CCC (refer to Figure 6-1 for location).

Word (W7)

DVCOUNT-Edge – The data valid counter for the edge of this CCC (refer to Figure 6-1 for location).

Word 8 (W8)

DATA_BIAS – The bias that is in each lag accumulator value (but not in the DVCOUNTs).

Words 9-264 (W9-W264) 128 words of complex lag data for this CCC. Lag numbering is as defined in Figure 6-1.

Word 265 (W265) End sync word as defined in the figure. This sync word and the following checksum help to identify interface communication errors while on-line.

Word 266 (W266) Checksum. This is the least significant 32 bits of the sum of all words, W0-W265.

5.4.4 MCB (Microprocessor) Interface Requirements

The MCB (Monitor & Control Bus) interface allows a microprocessor to write into the correlator chip to configure it, and to read from it to verify configuration information and obtain status information. Physical interface timing requirements are shown in Figure 5-5.

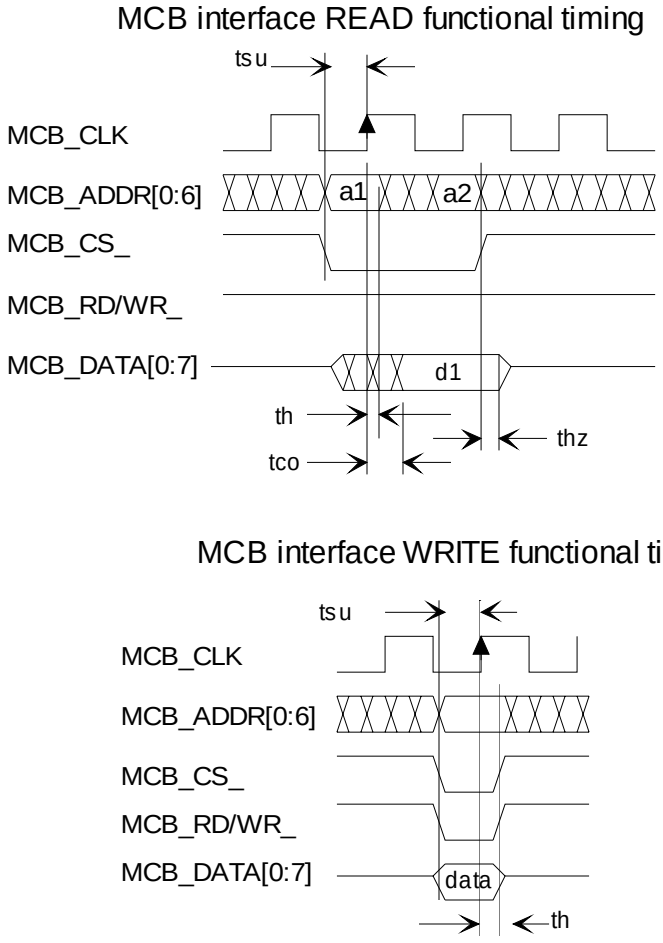


Figure 5-5 MCB interface functional timing requirements. READs require one clock cycle setup before data is ready—however, successive reads take one clock cycle each. WRITEs require one clock cycle.

The correlator chip contains a number of configuration and status registers that can be accessed via the MCB. These registers are defined in the following sub-sections.

5.4.4.1 Master Control/Status Register (MCSR); R/W Address=0x0

This register sets global chip configuration, and detects global status information such as input synchronization errors, and overflow and overrun errors.

MCSR (R/W) Addr=0x0

7	6	5	4	3	2	1	0
X/Y_DS	SyncER	TvER	AOV	OVR	PhEN	TVEN	CEN

CEN (R/W) – Correlation ENable. If set (1), then the correlator chip is operational and responds to dump commands. If reset (0), then the correlator chip does not respond to dump commands and does not generate output data frames, although internally it is still handling data in the normal fashion (i.e. power dissipation of the chip does not change based on the setting of this bit). **Other R/W bits in this register can *only* be changed when this bit is reset (0)**. This restriction prevents indeterminate operation in the chip.

TVEN (R/W) – Test Vector ENable. If set (1), then the X, Y input test vector receivers are invoked and the correlator chip does not respond to any X, Y inputs (section 4.1) in the normal way. If reset (0), then the X and Y interfaces are active and operate as normal. If this bit is set at the same time as the CEN bit is set, then the CEN bit is reset (0) internally.

PhEN (R/W) – Phase rotation ENable. If set (1), phase rotation in all CCCs is enabled. If reset (0), phase rotation in all CCCs is disabled (i.e. all phases are set to zero, but the fringe rotators still operate on "zero phase"). This bit is normally set, but could be reset in cases where phase rotation is unnecessary (such as auto-correlation). If the TVEN bit is set (1), then internally, this bit is reset to prevent fast-changing phase bits from propagating through the chip and consuming extra power.

OVR (R)– Overrun status bit. If set (1), then at least one CCC has had a buffer overrun where a dump request occurred, but the data had to be discarded because the output storage buffers were still full. Cleared on read. Clearing this bit does not affect any current CCC overrun status bit that is transmitted in an output data frame.

AOV (R) – Accumulator overflow status bit. If set (1), at least one accumulator in one CCC has had an accumulator overflow, a result of integrating for too long. Cleared on read. Clearing this bit does not affect any current CCC overrun status bit that is transmitted in an output data frame.

TvER (R) – Test vector receiver error. If set (1), then one or more X or Y inputs is reporting a test vector receiver error. This bit can only be set if TVEN is set. Cleared when the XSTATUS or YSTATUS register causing the error condition is cleared.

SyncER (R) – Input synchronization error. If set (1), one or more X or Y inputs *that affects an output data frame* is reporting an input synchronization error: data, phase, data

valid, and shift enable clocks or other inputs are not yielding values as expected on and after SCHID_FRAME_. Refer to Figure 5-1. Cleared on read (but does not affect XSTATUS or YSTATUS registers).

X/Y_DS (R/W) – X or Y dump select bit. If set (1), data is dumped based on X-DUMP_EN signals. If reset (0), data is dumped based on Y-DUMP_EN signals. Dumping for each CCC is determined by this setting and by the X and Y data streams that are being correlated in a particular CCC. For example, if a CCC is correlating input streams 0 (X) and 4 (Y), and X/Y_DS is set (1), X-DUMP_EN0 controls dumping for the CCC. If X/Y_DS is reset (0), Y-DUMP_EN4 controls dumping.

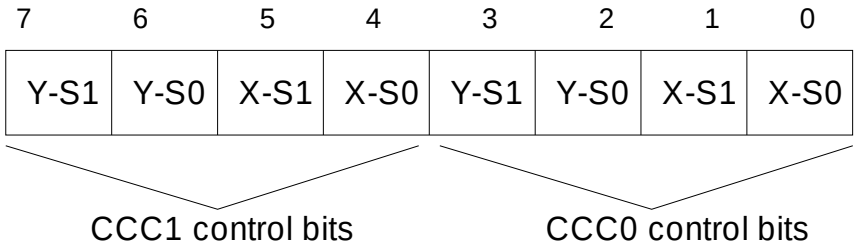
5.4.4.2 CCC Switch Configuration Registers; (R/W) Addr=0x1…0x8

These 8 registers set the X and Y input switches to all of the 16 CCCs (refer to Figure 6-4). Each register selects X and Y inputs for two CCCs. **These bits are only allowed to change when the CEN bit in the MCSR register is reset (0)**. This prevents indeterminate operation in the chip. The addresses for each register are according to the following table:

Register Name	Address	Description
CCCSCR_0_1	0x1	CCC0 and CCC1 switch configuration register.
CCCSCR_2_3	0x2	CCC2 and CCC3 switch configuration register.
CCCSCR_4_5	0x3	…
CCCSCR_6_7	0x4	…
CCCSCR_8_9	0x5	…
CCCSCR_10_11	0x6	…
CCCSCR_12_13	0x7	…
CCCSCR_14_15	0x8	CCC14 and CCC15 switch configuration register.

The bit definitions for CCCSCR_0_1 are defined below. Other CCCSCR registers have analogous bit assignments.

CCCSCR_0_1 (R/W) Addr=0x1



The X-inputs that are selected are according to the following table:

X-S1	X-S0	Selection
0	0	Adjacent CCC
0	1	Primary CCC input (CCQ-Sw0)
1	0	Secondary CCC input (CCQ-Sw1)
1	1	Undefined: no lag frames produced.

The Y-inputs that are selected are according to the following table:

Y-S1	Y-S0	Selection
0	0	Adjacent CCC
0	1	Primary CCC input (CCQ-Sw1)
1	0	Secondary CCC input (CCQ-Sw0)
1	1	Undefined: no lag frames produced.

Note that if the X-input switches for CCC0 are set to *Adjacent*, then it is autocorrelator mode and the Y-input switches for CCC15 must be set to *Adjacent*. In this case all of the other CCC switches for X and Y must be set to *Adjacent* as well for correct operation. If it is desired to disable lag-frame generation for any particular CCC (even though it may have valid inputs), the CCC's X or Y switches must be set to *Undefined*.

5.4.4.3 Autocorrelator Mode—CCCSCR_0_1 register

Autocorrelator mode is invoked when B0 and B1 of the CCCSCR_0_1 register are both reset (0). This forces the CCC0 X-input data switch to use adjacent data, and in this case is data from the "x-output" of the autocorrelator switch (Figure 6-6). When in this mode, B2, B3, and B4 of the CCCSCR_0_1 register select which X-input stream is to be autocorrelated according to the following table:

X-stream autocorrelated	B4	B3	B2
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

Table 5-2 Autocorrelator mode X-stream input selection for the CCCSCR_0_1 register

When in autocorrelator mode, the following additional points must be considered:

1. All other CCCSCR_a_b register bits must be reset to 0, or the CCCs will not be properly concatenated, and no useful data will be produced.
2. The X-PHASE*n* (where *n* is the X input stream that is to be autocorrelated) input must contain data that is to be fed into the adjacent of CCQ3 (CCC15) (i.e. coming from the "y output" of the autocorrelator switch). When an invalid sample is present, X-PHASE*n* input must be set to the –8 state (1000b). As such, on-line input PHASE synchronization errors are ignored by the chip in this mode.
3. Phase in the correlator lags (cmams) is forced to 0, to prevent overheating when phase may contain data.
4. RECIRC_BLK-Y=RECIRC_BLK-X in W3 of the output data frame.

5.4.4.4 CCQ Configuration Registers (CCQR_1/2, CCQR_3/4); (R/W) Addr=0x09, 0x0A

The bits in these registers control the data routing selection switches for each CCQ. These are the first stage switches that determine data routing from the correlator chip inputs to the CCC selection switches. **These bits are only allowed to change when the CEN bit in the MCSR register is reset (0)**. This prevents indeterminate operation in the chip. There are 2 registers, with 4, 1 of 2 switches per CCQ. Refer to Figure 6-5. When in autocorrelator mode, these switch settings are ignored.

CCQR_0/1 (R/W) Addr=0x9

7	6	5	4	3	2	1	0
Q1-YSw1	Q1-YSw0	Q1-XSw1	Q1-XSw0	Q0-YSw1	Q0-YSw0	Q0-XSw1	Q0-XSw0

Q0-XSw0 – CCQ-0, X switch ‘0’ selection. If reset (0), it selects the *primary* switch input (X-0 data), if set (1), it selects the *secondary* switch input (X-6 data).

Q0-XSw1 – CCQ-0, X switch ‘1’ selection. If reset (0), it selects the *primary* switch input (X-1 data), if set (1), it selects the *secondary* switch input (X-7 data).

Q0-YSw0 – CCQ-0, Y switch ‘0’ selection. If reset (0), it selects the *primary* switch input (Y-0 data), if set (1), it selects the *secondary* switch input (no data).

Q0-YSw1 – CCQ-0, Y switch ‘1’ selection. If reset (0), it selects the *primary* switch input (Y-1 data), if set (1), it selects the *secondary* switch input (no data).

Q1-XSw0 – CCQ-1, X switch ‘0’ selection. If reset (0), it selects the *primary* switch input (X-2 data), if set (1), it selects the *secondary* switch input (X-0 data).

Q1-XSw1 – CCQ-1, X switch ‘1’ selection. If reset (0), it selects the *primary* switch input (X-3 data), if set (1), it selects the *secondary* switch input (X-1 data).

Q1-YSw0 – CCQ-1, Y switch ‘0’ selection. If reset (0), it selects the *primary* switch input (Y-2 data), if set (1), it selects the *secondary* switch input (Y-0 data).

Q1-YSw1 – CCQ-1, Y switch ‘1’ selection. If reset (0), it selects the *primary* switch input (Y-3 data), if set (1), it selects the *secondary* switch input (Y-1 data).

CCQR_2/3 (R/W) Addr=0xA

7	6	5	4	3	2	1	0
Q3-YSw1	Q3-YSw0	Q3-XSw1	Q3-XSw0	Q2-YSw1	Q2-YSw0	Q2-XSw1	Q2-XSw0

Q2-XSw0 – CCQ-2, X switch ‘0’ selection. If reset (0), it selects the *primary* switch input (X-4 data), if set (1), it selects the *secondary* switch input (X-0 data).

Q2-XSw1 – CCQ-2, X switch ‘1’ selection. If reset (0), it selects the *primary* switch input (X-5 data), if set (1), it selects the *secondary* switch input (X-1 data).

Q2-YSw0 – CCQ-2, Y switch ‘0’ selection. If reset (0), it selects the *primary* switch input (Y-4 data), if set (1), it selects the *secondary* switch input (Y-0 data).

Q2-YSw1 – CCQ-2, Y switch ‘1’ selection. If reset (0), it selects the *primary* switch input (Y-5 data), if set (1), it selects the *secondary* switch input (Y-1 data).

Q3-XSw0 – CCQ-3, X switch ‘0’ selection. If reset (0), it selects the *primary* switch input (X-6 data), if set (1), it selects the *secondary* switch input (X-0 data).

Q3-XSw1 – CCQ-3, X switch ‘1’ selection. If reset (0), it selects the *primary* switch input (X-7 data), if set (1), it selects the *secondary* switch input (X-1 data).

Q3-YSw0 – CCQ-3, Y switch ‘0’ selection. If reset (0), it selects the *primary* switch input (Y-6 data), if set (1), it selects the *secondary* switch input (Y-0 data).

Q3-YSw1 – CCQ-3, Y switch ‘1’ selection. If reset (0), it selects the *primary* switch input (Y-7 data), if set (1), it selects the *secondary* switch input (Y-1 data).

5.4.4.5 Data Switch Selection Summary Table

The previous two sections define two types of registers for selecting data routing into each CCC. This method provides an efficient way for routing data to meet requirements, but the definition does not give a clear indication of what the actual data routing to each CCC is. This section contains a table that defines absolute data routing for each CCC, as a function of the CCCSCR and CCQR selection registers. This table (Table 5-3) can be used to map desired connectivity to selection register bits. X-0, Y-0, X-1, Y-1 etc. refer to X and Y data, phase, and data valid stream numbers—see section 4.1. In the table, '*n*' refers to the CCC number.

In the table, X-S0, X-S1, Y-S0, and Y-S1 refer to CCC switch select bits in the associated CCCSCR registers, and Q0-XSw0…Q3-YSw1 bits refer to specific CCQ switch selection register CCQR_0/1 or CCQR_2/3 bits.

It is important to note that SE_CLK routing is automatically controlled by the X and Y data path routing switches. For example if several CCCs are concatenated together and are correlating inputs 4 (X) and 5 (Y) then SE_CLK inputs 4 and 5 are used to enable shifting of data through all the CCCs' lag delay lines. **Both SE_CLKs must be operating at the same frequency for the correlation to make sense and produce useful data**.

Table 5-3 CCC X/Y input data routing summary table.

CCQ-0 : CCCn – X input [n=0…3]				
X-S1	**X-S0**	**Q0-XSw1**	**Q0-XSw0**	**Data/phase/data valid input**
0	0	X	X	[CCC(n-1)-X_OUT, n≠0]; [autocorrelator mode—autocorr switch x output, n=0]
0	1	X	0	X-0
1	0	0	X	X-1
0	1	X	1	X-6
1	0	1	X	X-7
CCQ-0 : CCCn – Y input [n=0…3]				
Y-S1	**Y-S0**	**Q0-Ysw1**	**Q0-YSw0**	**Data/phase/data valid input**
0	0	X	X	CCC(n+1)-Y_OUT
0	1	0	X	Y-1
1	0	X	0	Y-0

CCQ-1 : CCCn – X input [n=4…7]				
X-S1	**X-S0**	**Q1-XSw1**	**Q1-XSw0**	**Data/phase/data valid input**
0	0	X	X	CCC(n-1)-X_OUT
0	1	X	0	X-2
0	1	X	1	X-0
1	0	0	X	X-3
1	0	1	X	X-1
CCQ-1 : CCCn – Y input [n=4…7]				
Y-S1	**Y-S0**	**Q1-YSw1**	**Q1-YSw0**	**Data/phase/data valid input**
0	0	X	X	CCC(n+1)-Y_OUT
0	1	0	X	Y-3
0	1	1	X	Y-1
1	0	X	0	Y-2
1	0	X	1	Y-0

CCQ-2 : CCCn – X input [n=8…11]				
X-S1	**X-S0**	**Q2-XSw1**	**Q2-XSw0**	**Data/phase/data valid input**
0	0	X	X	CCC(n-1)-X_OUT
0	1	X	0	X-4
0	1	X	1	X-0
1	0	0	X	X-5
1	0	1	X	X-1
CCQ-2 : CCCn – Y input [n=8…11]				
Y-S1	**Y-S0**	**Q2-YSw1**	**Q2-YSw0**	**Data/phase/data valid input**
0	0	X	X	CCC(n+1)-Y_OUT
0	1	0	X	Y-5
0	1	1	X	Y-1
1	0	X	0	Y-4
1	0	X	1	Y-0

CCQ-3 : CCCn – X input [n=12…15]				
X-S1	**X-S0**	**Q3-XSw1**	**Q3-XSw0**	**Data/phase/data valid input**
0	0	X	X	CCC(n-1)-X_OUT
0	1	X	0	X-6
0	1	X	1	X-0
1	0	0	X	X-7
1	0	1	X	X-1
CCQ-3 : CCCn – Y input [n=12…15]				
Y-S1	**Y-S0**	**Q3-YSw1**	**Q3-YSw0**	**Data/phase/data valid input**
0	0	X	X	[CCC(n+1)-Y_OUT, n≠15] [y autocorrelator switch output, n=15]
0	1	0	X	Y-7
0	1	1	X	Y-1
1	0	X	0	Y-6
1	0	X	1	Y-0

5.4.4.6 X/Y Status Selection Register (XYSSR) (R/W) Addr=0xB; and XSTATUS (R) Addr=0xC, and YSTATUS (R) Addr=0xD, registers

The XYSSR register selects which X and Y input status bits are to be accessed when the XSTATUS and YSTATUS registers are read. This indirection is to allow efficient access to all X and Y input error detection bits. These registers can be read and contain valid status during test vector or normal operation.

XYSSR (R/W) Addr=0xB

7	6	5	4	3	2	1	0
YAd3	YAd2	YAd1	YAd0	XAd3	XAd2	XAd1	XAd0

XAd[0:3] – Selects the XSTATUS register bits according to Table 5-4. All XSTATUS register contents are LSB right-aligned in the bit ordering shown. X-TIMESTAMP is in bit location 0, SCHID_SYNC is in bit location 1, and DUMP_SYNC is in bit location 2 of the XSTATUS register. The actual XSTATUS register is at Addr=0xC. All XSTATUS register bits are cleared when a read from XAd[3:0]=1011 (0xB) is performed.

Table 5-4 XYSSR address and XSTATUS register contents.

XAd3	XAd2	XAd1	Xad0	XSTATUS[7:0] Contents
0	0	0	0	X-SDATA[1][3:0], X-SDATA[0][3:0]
0	0	0	1	X-SDATA[3][3:0], X-SDATA[2][3:0]
0	0	1	0	X-SDATA[5][3:0], X-SDATA[4][3:0]
0	0	1	1	X-SDATA[7][3:0], X-SDATA[6][3:0]
0	1	0	0	X-PHASE[1][3:0], X-PHASE[0][3:0]
0	1	0	1	X-PHASE[3][3:0], X-PHASE[2][3:0]
0	1	1	0	X-PHASE[5][3:0], X-PHASE[4][3:0]
0	1	1	1	X-PHASE[7][3:0], X-PHASE[6][3:0]
1	0	0	0	X-DVALID[7:0]
1	0	0	1	X-SE_CLK[7:0]
1	0	1	0	X-DUMP_EN[7:0]
1	0	1	1	X-TIMESTAMP(0), SCHID_SYNC[5](1), DUMP_SYNC[6](2)
1	1	x	X	Undefined

[5] Set if X-SCHID_FRAME_ and Y-SCHID_FRAME_ are not asserted coincidentally—this will only occur if the input FIFO buffer is smaller than the relative delay skew.

[6] Set if X-DUMP_SYNC and Y-DUMP_SYNC are not asserted coincidentally. This may or may not be a problem depending on the desired synchronization and operating mode (recirculation).

YAd[0:3] – Selects the YSTATUS register bits in an identical fashion as the XSTATUS register bits according to Table 5-4. The actual YSTATUS register is Addr=0xD. All YSTATUS register bits are cleared when a read from YAd[3:0]=1011 (0xB) is performed.

5.4.4.7 Dump Enable Sync Status Registers (DESSR_0/7, DESSR_8/15) (R) (Addr=0xE, 0xF)

These registers contain status information that is used to determine if the X and Y DUMP_EN[0:7] inputs to each CCC are properly synchronized (i.e. if the X-DUMP_ENn and Y-DUMP_ENn inputs are asserted simultaneously). If the DUMP_EN inputs to a particular CCC are not synchronized, then the corresponding bit in this register is set (1).

DESSR_0/7 contains status for CCCs 0 through 7 (bit 0==CCC0, bit1==CCC1 etc.) and DESSR_8/15 contains status for CCCs 8 through 15.

A set bit in this register may or may not be important depending on the mode of operation. If a CCC is being used for recirculation, then it is crucial that the corresponding DESSR status bit is not set (1). If it is, it indicates that dump signals—and corresponding recirculation synchronization—are not operating properly. If a CCC is not being used for recirculation, then a bit set in the DESSR may or may not indicate a problem depending on the desired dump configuration.

These registers are cleared on read.

5.4.5 Input Test Vector Requirements

This section defines the precise X and Y input test vector requirements to allow off-line testing of data connectivity to the correlator chip. In Figure 5-6, the basic LFSR (Linear-Feedback Shift Register) circuit is shown that generates an N-bit wide ($B_0 \ldots B_{N-1}$) test vector sequence synchronous with a clock, and reset/restarted with a synchronous preset [7].

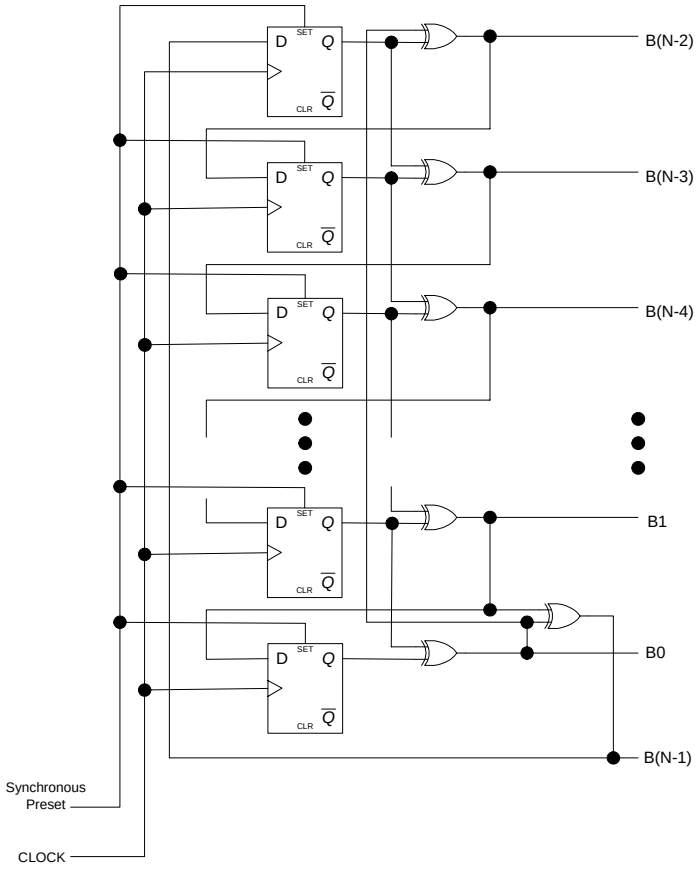


Figure 5-6 Synchronous test vector generator with synchronous preset. The synchronous preset is SCHID_FRAME_, so that the sequence restarts on every occurrence of SCHID_FRAME_. Bit numbering is as shown.

Table 5-5 defines input bit assignments to test vector bits (B_0, B_1, B_{N-1}). X and Y data input assignments are identical, but the X and Y test vector generators are independent and each one is synchronized to its own respective SCHID_FRAME_ pulse. Test vectors for X and Y DUMP_EN[7:0] and TIMESTAMP are synchronized to their respective X and Y DUMP_SYNCs. Also, for DUMP_EN[7:0] and TIMESTAMP, the lower 9 bits of a 10-bit generator are used[7].

[7] An even-number-of-bits generator is required for correct operation.

Table 5-5 Input assignments to test vector bits.

Test Vector Bits	Inputs
B[7:0]	SDATA[1][3:0], SDATA[0][3:0]
B[15:8]	SDATA[3][3:0], SDATA[2][3:0]
B[23:16]	SDATA[5][3:0], SDATA[4][3:0]
B[31:24]	SDATA[7][3:0], SDATA[6][3:0]
B[63:32]	PHASE[7][3:0]…PHASE[0][3:0]
B[71:64]	DVALID[7:0]
B[79:72]	SE_CLK[7:0]
B[7:0]	DUMP_EN[7:0]
B[8]	TIMESTAMP

The following table defines the last few words of the test vector generator sequence. N is 80 (SCHID_FRAME_ sync) or 10 (DUMP_SYNC sync).

Table 5-6 Last few words of the test vector generator sequence.

wd	Bit Number (B)							
	N-8	N-7	N-6	N-5	N-4	N-3	N-2	N-1
1	0	0	0	0	0	0	1	0
2	0	0	0	0	1	1	0	0
3	0	0	1	0	1	0	0	0
4	1	1	1	1	0	0	0	0
5	0	0	1	0	0	0	0	0
6	1	1	0	0	0	0	0	0
7	1	0	0	0	0	0	0	0
8	0	0	0	0	0	0	1	0
9	0	0	0	0	1	1	0	0
10	0	0	1	0	1	0	0	0
11	1	1	1	1	0	0	0	0
12	0	0	1	0	0	0	1	0
13	1	1	0	0	1	1	0	0
14	1	0	1	0	1	0	1	0
15	1	1	1	1	1	1	0	1
16	0	0	0	0	1	1	1	0
17	0	0	1	0	0	1	0	0
18	1	1	0	1	1	0	0	0
19	1	1	0	1	0	0	0	0
20	1	1	1	0	0	0	0	0

Since the circuit uses a synchronous preset, the first output word (wd 1) is produced in the clock cycle immediately after SCHID_FRAME_ (and also DUMP_SYNC).

5.4.6 Additional Interface Requirements

5.4.6.1 JTAG Interface and Function

The JTAG Interface and functionality shall conform to IEEE-1149.1.

6 Functional Specifications

This section describes in more detail the basic functionality of the correlator chip. The correlator chip is comprised of 16, 128-lag sections[8], each one operating independently and with the ability to connect each lag section's X and Y inputs to an adjacent lag section or some input data to the chip (see Table 5-3).

A simplified block diagram for an example 8-lag section is shown in Figure 6-1. The basic architecture showing the location and numbering of CMAMs (Complex Multiplier-Accumulator Modules—a.k.a. "lags"), data valid counters (DVCs), and shift register elements △ is defined by this diagram. Numbering and locations for an arbitrary number of even lags can be derived from this diagram.

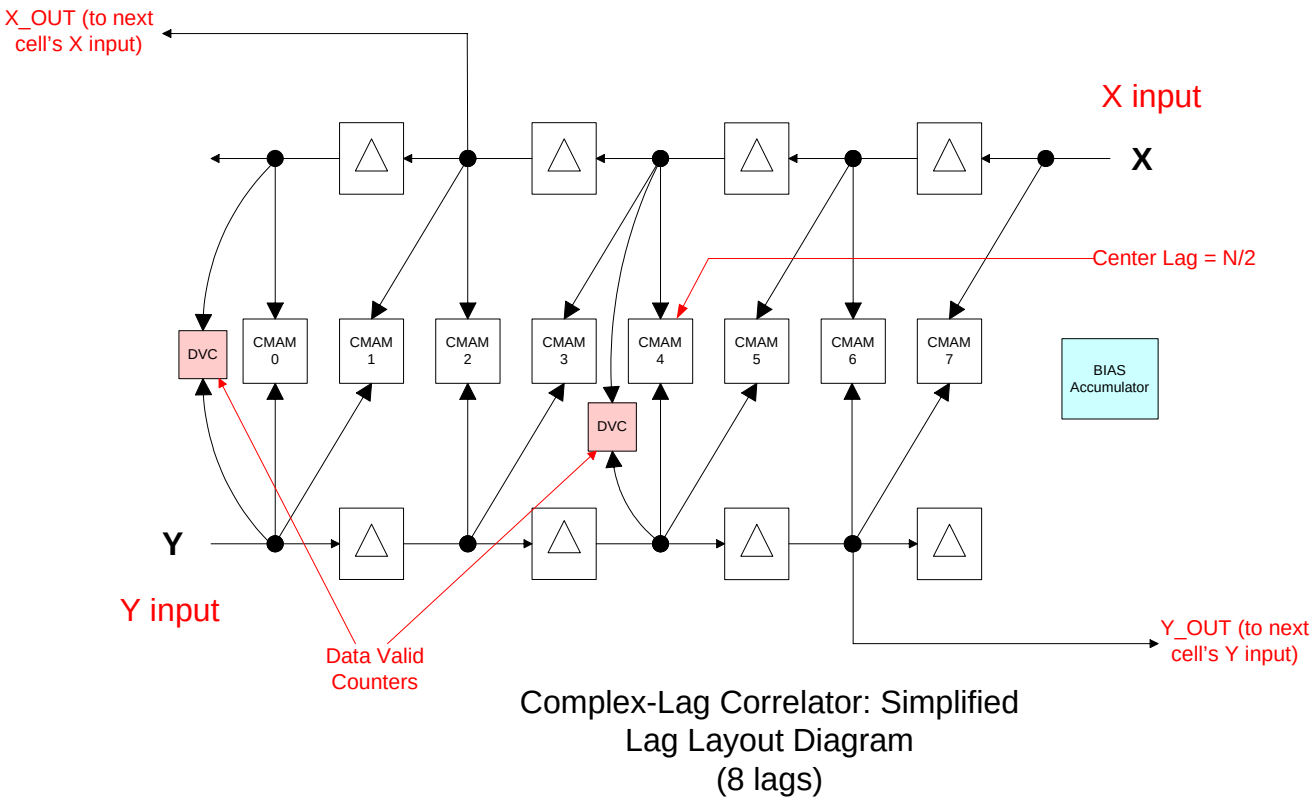


Figure 6-1 Simplified lag-section architecture for an 8-lag cross-correlator. There are N lags, with numbering from 0…N-1, and the center lag is at N/2. Each CMAM is a Complex Multiplier Accumulator Module and includes a data multiplier, complex mixer, and complex accumulator. The location of the data valid counters (DVC) is shown. This diagram is useful to establish lag numbering for an arbitrarily large correlator section (N even).

[8] A lag section is also known as a CCC – Correlator Chip Cell, as described later.

In order to compensate for pipeline delays when concatenating lag sections (see Figure 6-5), data is tapped off the X and Y shift registers 1 or more delay elements before the end of the shift register. Thus, multiple lag sections can be concatenated while still maintaining correct relative delays for each CMAM.

Each lag section contains a bias accumulator that is the exact bias that is included in the accumulator results. The bias is a result of efficient implementation of the 23-bit accumulator. This bias accumulator is not associated with any particular lag/CMAM because it is the same for each one in a lag section. The bias value is included in the chip output data frame for removal by off-chip hardware.

Each X and Y shift register element △ shifts 4 bits of data, 4 bits of phase, and one data valid bit by one clock cycle but enabled by a "shift enable clock" (SE_CLK) to permit correlation of data sampled at rates less than the actual clock cycle. Thus, the correlator CMAMs and shift registers can always operate at the highest clock rate, but the data propagates along the shift registers at a rate determined by the supplied SE_CLKs. A logic diagram of a single shift register element (a.k.a. "lag delay element") is shown in Figure 6-2. This element is a 9-bit wide flip-flop with a clock and clock enable.

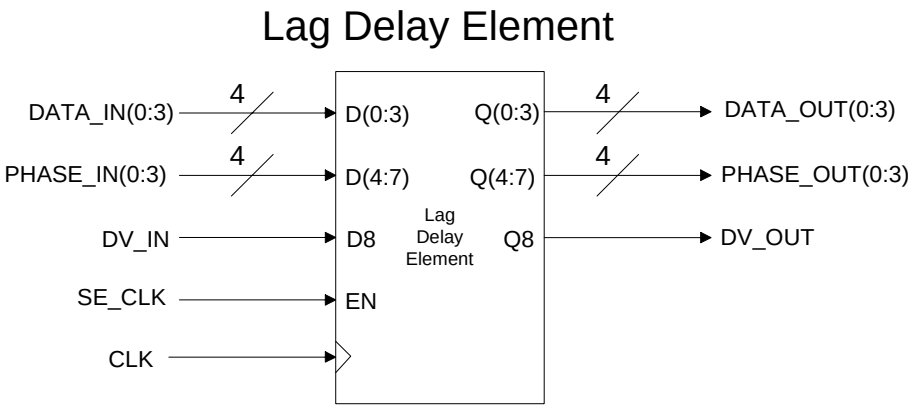


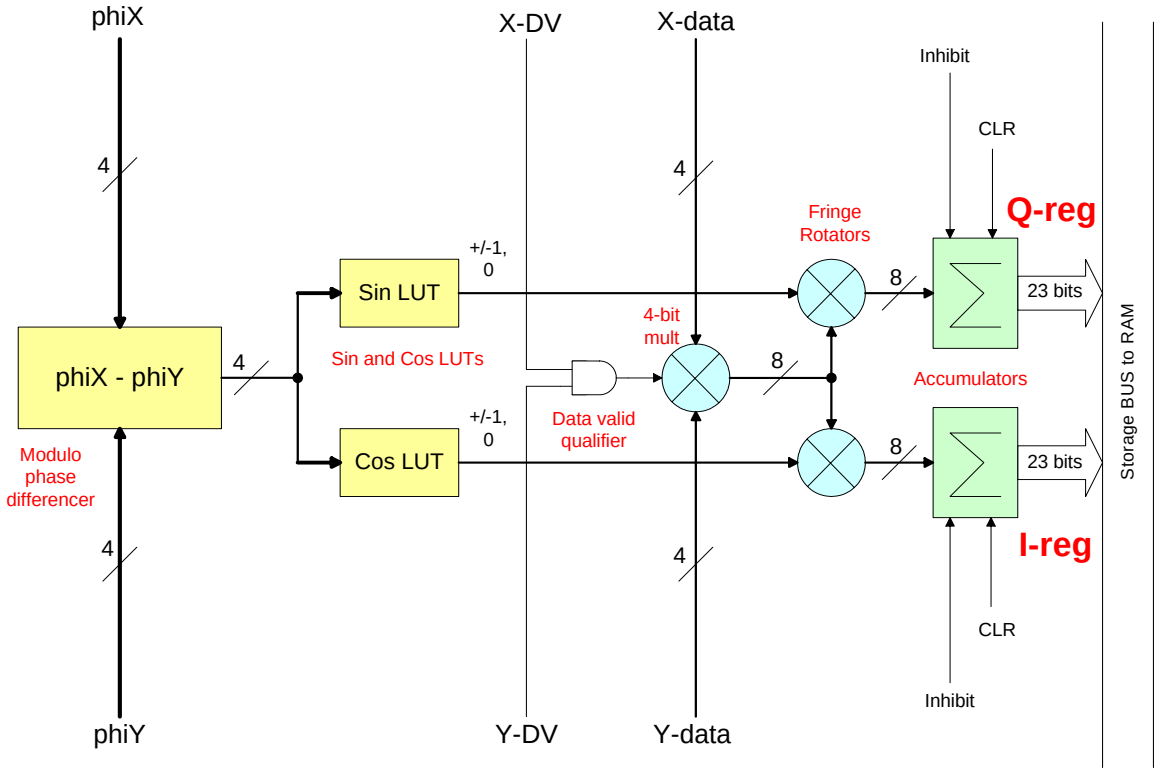
Figure 6-2 A shift register delay element (lag delay element).

In order to mitigate across-chip distribution of SE_CLKs (say, when all lag sections are concatenated), a series of selection switches allow each lag section to use its own SE_CLK—effectively implementing a selectable clock tree. Adjacent lag sections that are concatenated *must* select the same SE_CLK for use, otherwise indeterminate operation will occur. Also, the SE_CLKs must undergo the same number of clock delays (pipeline delays) on their way to the shift register delay elements as the data, and each X or Y shift register delay element within a particular lag section (CCC) must use the same SE_CLK.

The rest of this section outlines a plan to implement the above series of lag sections, including actual CMAM architecture, dump and readout control, input data selection, and output frame building and transmission.

6.1 CMAM Architecture

The simplified architecture of a CMAM (Complex Multiplier-Accumulator Module) is shown in Figure 6-3 below. In each CMAM, an *[X-Y] modulo 16* phase difference is calculated, and the result goes into sine and cosine lookup tables to produce 3-level sine and cosine functions. After data is multiplied with a 2's complement multiplier and qualified with data valid (i.e. if either of X or Y data valid is low, the multiplier result is zero), it is multiplied by the sine and cosine functions to produce quadrature and in-phase data respectively. This data is accumulated into separate in-phase (I-reg) and quadrature (Q-reg) accumulators. When the integration is complete, correlation is inhibited with the Inhibit signal, and stored into internal storage RAM. Once storage is complete, the active I-reg and Q-reg registers are cleared, inhibit is removed, and correlation continues. Meanwhile, the data stored in the storage RAM is read out on the readout bus and transmitted to off-chip hardware.



CMAM: Complex Multiplier-Accumulator Module

Figure 6-3 Simplified functional architecture of a Complex Multiplier-Accumulator Module (CMAM). Data is multiplied and then complex phase-rotated before being accumulated into separate in-phase (I-reg) and quadrature (Q-reg) registers. When a dump occurs, correlation is inhibited before it is saved into RAM via the "Storage BUS to RAM". Once storage is complete, the active accumulators (I-reg, Q-reg) are cleared, inhibit is removed, and correlation continues.

In the chip implementation, the "Storage BUS to RAM" is parameterized to be one of a tri-state bus with a maximum of 16 drivers on it, or a MUX implemented using wide OR gates.

There are some important factors to consider in the implementation of the CMAM:

- The CMAM must be made as efficient as possible both in terms of logic used and power dissipated since CMAM logic dominates the chip.
- Since phase is changing much more slowly than the data, the contribution to power dissipation in the phase difference calculation and the sine/cosine LUTs (Look Up Tables) is negligible.
- Pipelining can be used to the maximum extent necessary to achieve the desired speed since—within reason—it doesn't matter how long after data actually enters the CMAM that it is accumulated.
- When a dump occurs, it is acceptable if some new input data is ignored as long as it doesn't exceed the specification of 1 μsec outlined in section 5.1, and as long as it doesn't result in any systematic biases in the data. (i.e. the bias accumulator and data valid counters must experience identical dead-time at the identical time).
- Readout of CMAM accumulators need only be sequential, one-time access. Random access is not a requirement.

6.2 Correlator Chip Cell (CCC)

A Correlator Chip Cell (CCC) is a lag section and consists of 128 CMAMs, input selection logic, dump control logic, and readout control logic. A simplified block diagram of a CCC is shown in Figure 6-4.

The bits of the CCCSCR_m_n and the CCQR_a/b registers control the Data Source Switches and the Shift Enable Select switches. The Cell Readout Controller is responsible for handling a DUMP_REQ (originating from a selected input DUMP_EN signal) and dumping the active CMAM accumulators into the Storage DPSRAM memory. It does this by inhibiting the accumulators with the Inhibit_corr signal, saving the accumulators into memory, clearing the accumulators, and then removing Inhibit_corr. The externally supplied CORR_HOLDOFF will continue to inhibit correlation as long as it is asserted. This functionality allows correlation to be inhibited by the Recirculation Controller (see section 5.4.1, signal "Holdoff").

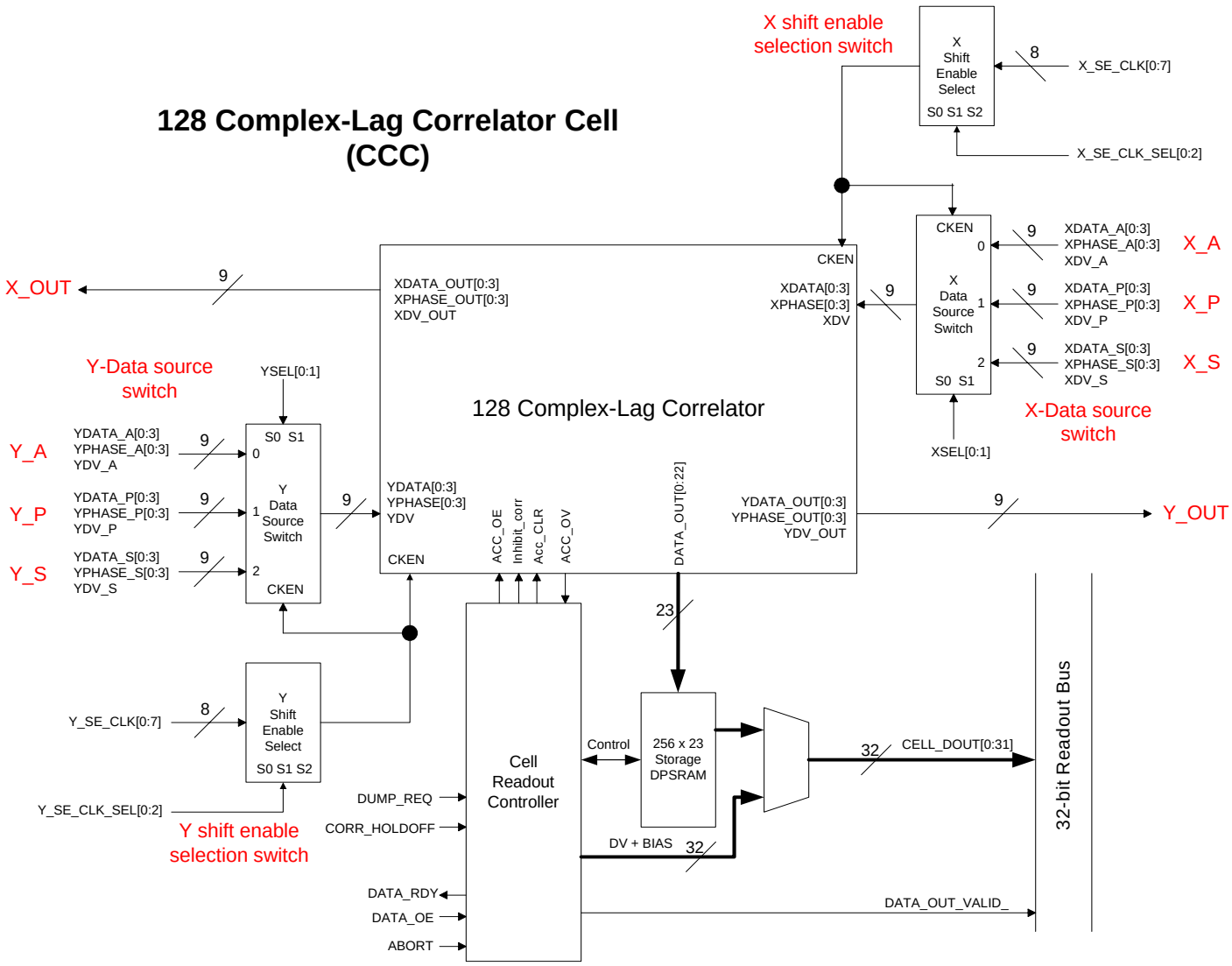


Figure 6-4 128 complex-lag (CMAM) Correlator Chip Cell (CCC). This cell contains 128 CMAMs, data valid counters, a bias counter arranged as shown in Figure 6-1, dump and readout logic, and X and Y data/shift enable selection switches.

The readout, save, and clear operation for a 128-lag section takes about 1 μsec at a 256 MHz readout rate. If data is ready (DATA_RDY), the cell's master frame generator (external to this block) can enable transmission of data by selecting the cell and asserting DATA_OE for sequential transmission of lag data, data valid counts, and the bias accumulator onto the 32-bit readout bus. (This readout bus is parameterized to be either a tri-state bus or a MUX implemented with wide OR gates.) If a new dump request occurs before the existing data has been transmitted, the active accumulators are cleared and the existing data is not overwritten. If ABORT is asserted, the current output frame transmission is terminated and the buffer is ready for new data.

In Figure 6-4, for simplicity, groups of X and Y signals are referred to as “X_OUT”, “X_A”, “X_P” etc. These designations will be used in the next section where a CCQ block diagram is presented.

6.3 Correlator Chip Quad (CCQ)

The Correlator Chip Quad (CCQ) consists of 4 CCCs connected to input data selection switches (X-Sw0, X-Sw1, Y-Sw0, Y-Sw1) in a way to allow the correlation of 1, 2, or 4 polarization products using 1 or more CCCs as required. A CCQ is thus able to perform all necessary correlations for a polarized pair of sampled data signals. Each CCC within the CCQ drives the output bus (OUT_BUS[0:31]) and drives the DATA_VALID_ line indicating that there is valid data on the bus (Note: this DATA_VALID_ line is *not* the same as the X/Y input data valids).

There is one bus per CCQ (either a tri-state or a MUX) and one final bus tying together data from all CCQs (not shown).

X_S0, X_P0, X_S1, X_P1, Y_S0 etc. are primary and secondary X and Y data inputs. Refer to Figure 6-6 for a diagram of how these CCQ inputs are connected to correlator chip inputs.

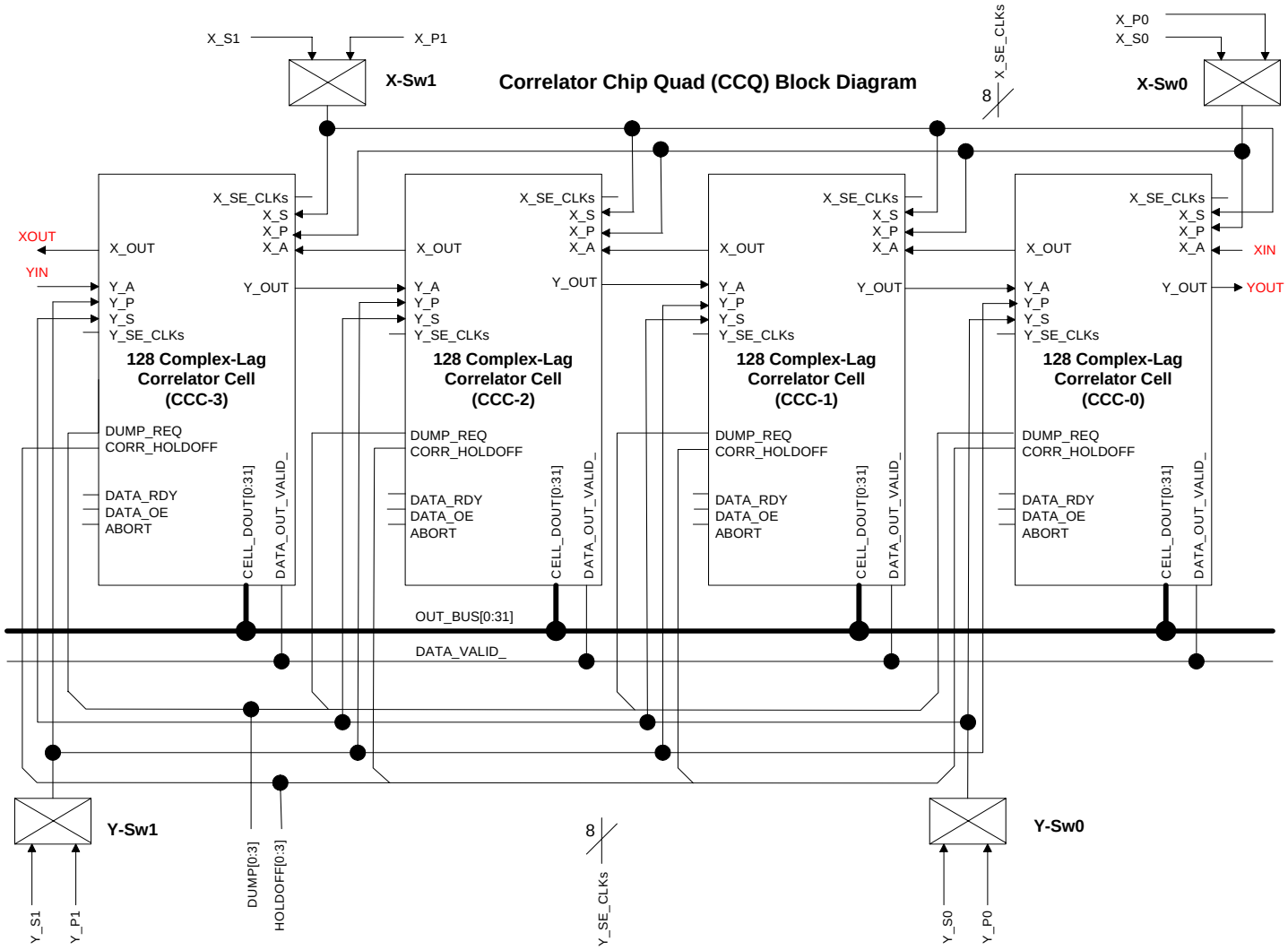


Figure 6-5 Correlator Chip Quad block diagram. This diagram shows the location of X and Y input selection switches, and how the outputs get mapped into CCC A—Alternate, P—Primary, and S—Secondary inputs. XIN, YOUT, XOUT, and YIN signals get routed to adjacent CCQs. The lowest numbered CCC within the CCQ is to the right.

The CCQ is a collection of CCCs wired to allow them to operate as a separate entity in the design hierarchy. The top level of the hierarchy, encompasses all functions, and is described in the next section.

6.4 Top-Level Chip Block Diagram

Figure 6-6 is the top-level chip block diagram. This diagram is an elaboration of that shown in Figure 3-1, except that the input FIFO buffers, the daisy-chained outputs, and the PLLs are not shown. This simplified diagram illustrates how all CCQs and chip functions are tied together, with the primary purpose of showing how data streams into the correlator chip are routed to CCQs.

EVLA Correlator Chip Top-Level Block Diagram

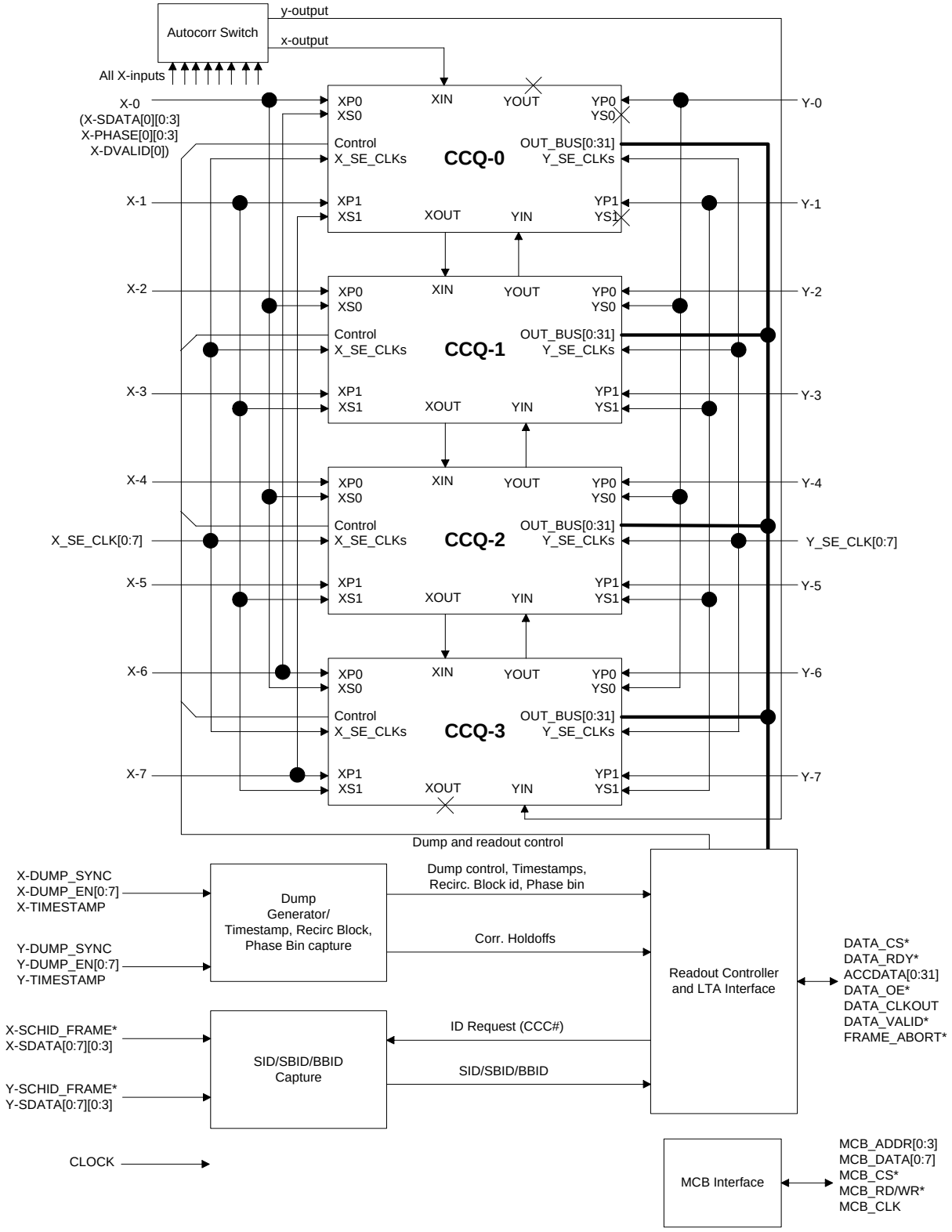


Figure 6-6 Correlator chip top-level block diagram.

The “X-0”, “X-1”, …, “Y-0”, “Y-1”… are short-form input designators, each of which contains 4 bits of SDATA, 4 bits of PHASE, and one DATA_VALID for the indicated data stream number.

The functionality of each of the blocks shown in Figure 6-6 is described below.

- CCQ-0 (1, 2, 3) – These are the Correlator Chip Quads that each contain 4, 128-lag CCCs described in previous sections. X and Y inputs[9] to these blocks and their interconnections are defined in the block diagram.

- Dump Generator, Timestamp, Recirc Block, Phase Bin Capture – This block selects (according to the CCCSCR_m_n and CCQR_a/b configuration registers) which input DUMP_EN signal goes to each CCC. It also captures the Timestamp, Recirc Block and Phase Bin number and saves it for eventual transmission as part of the correlator chip output data frame for the particular CCC.

- SID/SBID/BBID Capture – This block captures embedded identifier information from the input SDATA data streams and saves it in a table. Data routing configuration for each CCC will then form pointers into this table, so that the correct identifiers are embedded in a particular CCC’s output data frame.

- Readout Controller and LTA Interface – This block is responsible for control of output data transmission from each CCC. It sequentially queries each CCC to see if it has data ready to transmit, and then facilitates transmission of the data frame on the LTA Interface.

- MCB Interface – This is the microprocessor interface and contains all configuration and status register information.

- Refer to section 5.4.4.3 for a description of autocorrelator mode.

Not shown in Figure 6-6 are the input test vector receivers and error detectors.

[9] X_0, Y_0 etc. inputs refer to DATA, PHASE, DVALID inputs for X and Y sampled data streams 0 through 7.

7 References

[1] Carlson, B.R., TEST AND VERIFICATION PLAN: EVLA Correlator Chip, TVP Document A25082N0001, Rev. DRAFT, August 26, 2004.

[2] Carlson, Brent, Thermal Analysis of Cooling Strategies for the EVLA Correlator Baseline Board, NRC-EVLA Memo# 017, September 3, 2004.

[3] Carlson, Brent, 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 (EVLA Memo 16), September 28, 2000

[4] Thompson, A.R., Moran, J.M., and Swenson, G.W., Interferometry and Synthesis in Radio Astronomy, Wiley, New York, 1986

[5] 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.

[6] Altera Reliability Report, Q1, 2003 http://www.altera.com/literature/rr/rr38.pdf.

[7] S2-PT User's Manual, CRESTECH, Version 2.5, January 1995.

8 Appendix I – Chip Signal Pinouts

Signal	Pin	Description	Signal	Pin	Description
X_SDATA0_I_pad[0]	N2	X input data stream 0	X_PHASE0_I_pad[0]	AA5	X input phase stream 0
X_SDATA0_I_pad[1]	AA4		X_PHASE0_I_pad[1]	W1	
X_SDATA0_I_pad[2]	T6		X_PHASE0_I_pad[2]	M1	
X_SDATA0_I_pad[3]	J2		X_PHASE0_I_pad[3]	AA1	
X_SDATA1_I_pad[0]	U1	X input data stream 1	X_PHASE1_I_pad[0]	Y7	X input phase stream 1
X_SDATA1_I_pad[1]	L2		X_PHASE1_I_pad[1]	L3	
X_SDATA1_I_pad[2]	Y2		X_PHASE1_I_pad[2]	Y3	
X_SDATA1_I_pad[3]	AA2		X_PHASE1_I_pad[3]	P1	
X_SDATA2_I_pad[0]	R6	X input data stream 2	X_PHASE2_I_pad[0]	W6	X input phase stream 2
X_SDATA2_I_pad[1]	P4		X_PHASE2_I_pad[1]	P5	
X_SDATA2_I_pad[2]	T7		X_PHASE2_I_pad[2]	V1	
X_SDATA2_I_pad[3]	U6		X_PHASE2_I_pad[3]	H7	
X_SDATA3_I_pad[0]	U3	X input data stream 3	X_PHASE3_I_pad[0]	Y8	X input phase stream 3
X_SDATA3_I_pad[1]	K2		X_PHASE3_I_pad[1]	Y6	
X_SDATA3_I_pad[2]	H6		X_PHASE3_I_pad[2]	W8	
X_SDATA3_I_pad[3]	K4		X_PHASE3_I_pad[3]	V3	
X_SDATA4_I_pad[0]	AB4	X input data stream 4	X_PHASE4_I_pad[0]	AB3	X input phase stream 4
X_SDATA4_I_pad[1]	AA6		X_PHASE4_I_pad[1]	K3	
X_SDATA4_I_pad[2]	P3		X_PHASE4_I_pad[2]	U4	
X_SDATA4_I_pad[3]	K1		X_PHASE4_I_pad[3]	M4	
X_SDATA5_I_pad[0]	AB2	X input data stream 5	X_PHASE5_I_pad[0]	AA3	X input phase stream 5
X_SDATA5_I_pad[1]	V7		X_PHASE5_I_pad[1]	Y5	
X_SDATA5_I_pad[2]	N1		X_PHASE5_I_pad[2]	W7	
X_SDATA5_I_pad[3]	T3		X_PHASE5_I_pad[3]	W3	
X_SDATA6_I_pad[0]	J5	X input data stream 6	X_PHASE6_I_pad[0]	P2	X input phase stream 6
X_SDATA6_I_pad[1]	T4		X_PHASE6_I_pad[1]	U5	
X_SDATA6_I_pad[2]	W5		X_PHASE6_I_pad[2]	V5	
X_SDATA6_I_pad[3]	M5		X_PHASE6_I_pad[3]	V2	
X_SDATA7_I_pad[0]	N3	X input data stream 7	X_PHASE7_I_pad[0]	T1	X input phase stream 7
X_SDATA7_I_pad[1]	P7		X_PHASE7_I_pad[1]	Y1	
X_SDATA7_I_pad[2]	Y4		X_PHASE7_I_pad[2]	R5	
X_SDATA7_I_pad[3]	N5		X_PHASE7_I_pad[3]	R7	
X_DVALID_I_pad[0]	U2	X input data valid. Each input is applicable to the associated/numbered X input stream	X_DUMP_EN_I_pad[0]	H4	X input dump enable. Each input is applicable to the associated/numbered X input stream
X_DVALID_I_pad[1]	U7		X_DUMP_EN_I_pad[1]	V6	
X_DVALID_I_pad[2]	L1		X_DUMP_EN_I_pad[2]	R4	
X_DVALID_I_pad[3]	M3		X_DUMP_EN_I_pad[3]	M2	
X_DVALID_I_pad[4]	N4		X_DUMP_EN_I_pad[4]	R2	
X_DVALID_I_pad[5]	W2		X_DUMP_EN_I_pad[5]	T5	
X_DVALID_I_pad[6]	W4		X_DUMP_EN_I_pad[6]	N7	
X_DVALID_I_pad[7]	R1		X_DUMP_EN_I_pad[7]	H5	
X_SE_CLK_I_pad[0]	P6	X input shift-enable clocks (clock enables). Each input is applicable to the associated/numbered X input stream	X_DUMP_SYNC_I_pad	V4	X input dump sync
X_SE_CLK_I_pad[1]	K5		X_TIMESTAMP_I_pad	L4	X input timestamp
X_SE_CLK_I_pad[2]	J4		X_SCHID_FRAME_I_pad_	J1	X input schid_frame pulse
X_SE_CLK_I_pad[3]	T2		X_CLOCK_I_pad	G14	X input 32 MHz clock
X_SE_CLK_I_pad[4]	N6				
X_SE_CLK_I_pad[5]	J3				
X_SE_CLK_I_pad[6]	L5				
X_SE_CLK_I_pad[7]	R3				

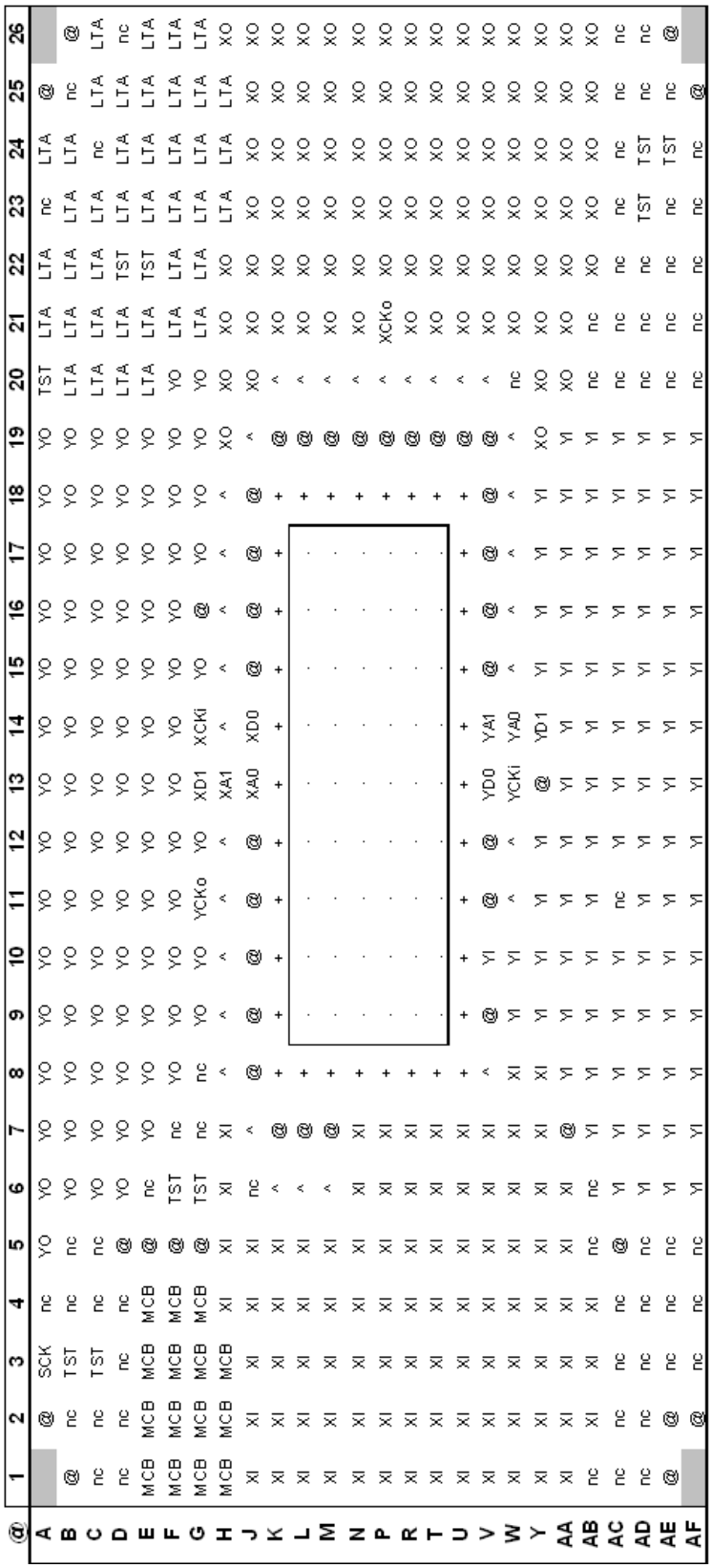
Signal	Pin	Description	Signal	Pin	Description
X_SDATA0_O_pad[0]	P21	X output data stream 0	X_PHASE0_O_pad[0]	AA25	X output phase stream 0
X_SDATA0_O_pad[1]	AA24		X_PHASE0_O_pad[1]	AA20	
X_SDATA0_O_pad[2]	M26		X_PHASE0_O_pad[2]	M21	
X_SDATA0_O_pad[3]	H21		X_PHASE0_O_pad[3]	AB22	
X_SDATA1_O_pad[0]	T21	X output data stream 1	X_PHASE1_O_pad[0]	Y26	X output phase stream 1
X_SDATA1_O_pad[1]	L21		X_PHASE1_O_pad[1]	K21	
X_SDATA1_O_pad[2]	Y22		X_PHASE1_O_pad[2]	Y23	
X_SDATA1_O_pad[3]	AA22		X_PHASE1_O_pad[3]	N21	
X_SDATA2_O_pad[0]	P26	X output data stream 2	X_PHASE2_O_pad[0]	W24	X output phase stream 2
X_SDATA2_O_pad[1]	N25		X_PHASE2_O_pad[1]	N26	
X_SDATA2_O_pad[2]	T26		X_PHASE2_O_pad[2]	Y19	
X_SDATA2_O_pad[3]	T24		X_PHASE2_O_pad[3]	J26	
X_SDATA3_O_pad[0]	U23	X output data stream 3	X_PHASE3_O_pad[0]	W26	X output phase stream 3
X_SDATA3_O_pad[1]	H20		X_PHASE3_O_pad[1]	Y25	
X_SDATA3_O_pad[2]	J25		X_PHASE3_O_pad[2]	V26	
X_SDATA3_O_pad[3]	K24		X_PHASE3_O_pad[3]	V21	
X_SDATA4_O_pad[0]	AB26	X output data stream 4	X_PHASE4_O_pad[0]	AB25	X output phase stream 4
X_SDATA4_O_pad[1]	AA26		X_PHASE4_O_pad[1]	J21	
X_SDATA4_O_pad[2]	P24		X_PHASE4_O_pad[2]	U24	
X_SDATA4_O_pad[3]	L22		X_PHASE4_O_pad[3]	J20	
X_SDATA5_O_pad[0]	AB24	X output data stream 5	X_PHASE5_O_pad[0]	AA23	X output phase stream 5
X_SDATA5_O_pad[1]	V25		X_PHASE5_O_pad[1]	W23	
X_SDATA5_O_pad[2]	AB23		X_PHASE5_O_pad[2]	W25	
X_SDATA5_O_pad[3]	R21		X_PHASE5_O_pad[3]	W21	
X_SDATA6_O_pad[0]	J23	X output data stream 6	X_PHASE6_O_pad[0]	P23	X output phase stream 6
X_SDATA6_O_pad[1]	R22		X_PHASE6_O_pad[1]	T25	
X_SDATA6_O_pad[2]	V23		X_PHASE6_O_pad[2]	U25	
X_SDATA6_O_pad[3]	L23		X_PHASE6_O_pad[3]	Y21	
X_SDATA7_O_pad[0]	N24	X output data stream 7	X_PHASE7_O_pad[0]	U21	X output phase stream 7
X_SDATA7_O_pad[1]	L26		X_PHASE7_O_pad[1]	Y20	
X_SDATA7_O_pad[2]	Y24		X_PHASE7_O_pad[2]	P25	
X_SDATA7_O_pad[3]	M24		X_PHASE7_O_pad[3]	L25	
X_DVALID_O_pad[0]	T22	X output data valid. Each output is applicable to the associated/numbered X output stream	X_DUMP_EN_O_pad[0]	H26	X output dump enable. Each output is applicable to the associated/numbered X output stream
X_DVALID_O_pad[1]	U26		X_DUMP_EN_O_pad[1]	V24	
X_DVALID_O_pad[2]	P22		X_DUMP_EN_O_pad[2]	R26	
X_DVALID_O_pad[3]	P21		X_DUMP_EN_O_pad[3]	M22	
X_DVALID_O_pad[4]	M23		X_DUMP_EN_O_pad[4]	R24	
X_DVALID_O_pad[5]	AA21		X_DUMP_EN_O_pad[5]	T23	
X_DVALID_O_pad[6]	W22		X_DUMP_EN_O_pad[6]	K26	
X_DVALID_O_pad[7]	R23		X_DUMP_EN_O_pad[7]	J24	
X_SE_CLK_O_pad[0]	M25	X output shift-enable clocks (clock enables). Each output is applicable to the associated/numbered X output stream	X_DUMP_SYNC_O_pad	V22	X output dump sync
X_SE_CLK_O_pad[1]	K25		X_TIMESTAMP_O_pad	K22	X output timestamp
X_SE_CLK_O_pad[2]	J22		X_SCHID_FRAME_O_pad_	H19	X output schid_frame pulse
X_SE_CLK_O_pad[3]	U22		X_CLOCK_O_pad	P21	X output 32 MHz clock
X_SE_CLK_O_pad[4]	L24				
X_SE_CLK_O_pad[5]	H22				
X_SE_CLK_O_pad[6]	K23				
X_SE_CLK_O_pad[7]	R25				

Signal	Pin	Description	Signal	Pin	Description
Y_SDATA0_I_pad[0]	AE11	Y input data stream 0	Y_PHASE0_I_pad[0]	AF19	Y input phase stream 0
Y_SDATA0_I_pad[1]	Y18		Y_PHASE0_I_pad[1]	AF17	
Y_SDATA0_I_pad[2]	AF11		Y_PHASE0_I_pad[2]	AC10	
Y_SDATA0_I_pad[3]	AD6		Y_PHASE0_I_pad[3]	AA16	
Y_SDATA1_I_pad[0]	AF14	Y input data stream 1	Y_PHASE1_I_pad[0]	AB17	Y input phase stream 1
Y_SDATA1_I_pad[1]	AC9		Y_PHASE1_I_pad[1]	AB9	
Y_SDATA1_I_pad[2]	AF18		Y_PHASE1_I_pad[2]	AE18	
Y_SDATA1_I_pad[3]	AB18		Y_PHASE1_I_pad[3]	AD10	
Y_SDATA2_I_pad[0]	Y11	Y input data stream 2	Y_PHASE2_I_pad[0]	AC16	Y input phase stream 2
Y_SDATA2_I_pad[1]	AB11		Y_PHASE2_I_pad[1]	AA11	
Y_SDATA2_I_pad[2]	AD14		Y_PHASE2_I_pad[2]	AA13	
Y_SDATA2_I_pad[3]	AC13		Y_PHASE2_I_pad[3]	AB7	
Y_SDATA3_I_pad[0]	AE15	Y input data stream 3	Y_PHASE3_I_pad[0]	AA17	Y input phase stream 3
Y_SDATA3_I_pad[1]	AE8		Y_PHASE3_I_pad[1]	AC18	
Y_SDATA3_I_pad[2]	AC7		Y_PHASE3_I_pad[2]	AA14	
Y_SDATA3_I_pad[3]	AA8		Y_PHASE3_I_pad[3]	Y16	
Y_SDATA4_I_pad[0]	AA19	Y input data stream 4	Y_PHASE4_I_pad[0]	AB19	Y input phase stream 4
Y_SDATA4_I_pad[1]	AE19		Y_PHASE4_I_pad[1]	AD8	
Y_SDATA4_I_pad[2]	AE12		Y_PHASE4_I_pad[2]	AD15	
Y_SDATA4_I_pad[3]	AA9		Y_PHASE4_I_pad[3]	AF8	
Y_SDATA5_I_pad[0]	AC19	Y input data stream 5	Y_PHASE5_I_pad[0]	AA18	Y input phase stream 5
Y_SDATA5_I_pad[1]	AB15		Y_PHASE5_I_pad[1]	AB16	
Y_SDATA5_I_pad[2]	AD19		Y_PHASE5_I_pad[2]	AA15	
Y_SDATA5_I_pad[3]	AF13		Y_PHASE5_I_pad[3]	AD17	
Y_SDATA6_I_pad[0]	AE7	Y input data stream 6	Y_PHASE6_I_pad[0]	AF12	Y input phase stream 6
Y_SDATA6_I_pad[1]	AE13		Y_PHASE6_I_pad[1]	AE14	
Y_SDATA6_I_pad[2]	AE16		Y_PHASE6_I_pad[2]	AC15	
Y_SDATA6_I_pad[3]	Y9		Y_PHASE6_I_pad[3]	Y15	
Y_SDATA7_I_pad[0]	AD11	Y input data stream 7	Y_PHASE7_I_pad[0]	AC14	Y input phase stream 7
Y_SDATA7_I_pad[1]	AE10		Y_PHASE7_I_pad[1]	Y17	
Y_SDATA7_I_pad[2]	AD18		Y_PHASE7_I_pad[2]	AD12	
Y_SDATA7_I_pad[3]	W10		Y_PHASE7_I_pad[3]	AF10	
Y_DVALID_I_pad[0]	AF15	Y input data valid. Each input is applicable to the associated/numbered Y input stream	Y_DUMP_EN_I_pad[0]	AF6	Y input dump enable. Each input is applicable to the associated/numbered Y input stream
Y_DVALID_I_pad[1]	AB14		Y_DUMP_EN_I_pad[1]	AD16	
Y_DVALID_I_pad[2]	AD9		Y_DUMP_EN_I_pad[2]	Y12	
Y_DVALID_I_pad[3]	AA10		Y_DUMP_EN_I_pad[3]	AB10	
Y_DVALID_I_pad[4]	Y10		Y_DUMP_EN_I_pad[4]	AB12	
Y_DVALID_I_pad[5]	AE17		Y_DUMP_EN_I_pad[5]	AD13	
Y_DVALID_I_pad[6]	AC17		Y_DUMP_EN_I_pad[6]	AE9	
Y_DVALID_I_pad[7]	AC12		Y_DUMP_EN_I_pad[7]	AD7	
Y_SE_CLK_I_pad[0]	V10	Y input shift-enable clocks (clock enables). Each input is applicable to the associated/numbered Y input stream	Y_DUMP_SYNC_I_pad	AF16	Y input dump sync
Y_SE_CLK_I_pad[1]	AF9		Y_TIMESTAMP_I_pad	AC8	Y input timestamp
Y_SE_CLK_I_pad[2]	AF7		Y_SCHID_FRAME_I_pad_	AE6	Y input schid_frame pulse
Y_SE_CLK_I_pad[3]	AB13		Y_CLOCK_I_pad	W13	Y input 32 MHz clock
Y_SE_CLK_I_pad[4]	W9				
Y_SE_CLK_I_pad[5]	AC6				
Y_SE_CLK_I_pad[6]	AB8				
Y_SE_CLK_I_pad[7]	AA12				

Signal	Pin	Description	Signal	Pin	Description
Y_SDATA0_O_pad[0]	E11	Y output data stream 0	Y_PHASE0_O_pad[0]	F19	Y output phase stream 0
Y_SDATA0_O_pad[1]	A18		Y_PHASE0_O_pad[1]	G20	
Y_SDATA0_O_pad[2]	G12		Y_PHASE0_O_pad[2]	E10	
Y_SDATA0_O_pad[3]	B6		Y_PHASE0_O_pad[3]	D18	
Y_SDATA1_O_pad[0]	B14	Y output data stream 1	Y_PHASE1_O_pad[0]	E18	Y output phase stream 1
Y_SDATA1_O_pad[1]	D9		Y_PHASE1_O_pad[1]	C9	
Y_SDATA1_O_pad[2]	G19		Y_PHASE1_O_pad[2]	F20	
Y_SDATA1_O_pad[3]	C18		Y_PHASE1_O_pad[3]	F10	
Y_SDATA2_O_pad[0]	C11	Y output data stream 2	Y_PHASE2_O_pad[0]	D17	Y output phase stream 2
Y_SDATA2_O_pad[1]	E12		Y_PHASE2_O_pad[1]	D11	
Y_SDATA2_O_pad[2]	F14		Y_PHASE2_O_pad[2]	B15	
Y_SDATA2_O_pad[3]	B13		Y_PHASE2_O_pad[3]	A6	
Y_SDATA3_O_pad[0]	F15	Y output data stream 3	Y_PHASE3_O_pad[0]	B17	Y output phase stream 3
Y_SDATA3_O_pad[1]	E8		Y_PHASE3_O_pad[1]	F18	
Y_SDATA3_O_pad[2]	B7		Y_PHASE3_O_pad[2]	B16	
Y_SDATA3_O_pad[3]	A7		Y_PHASE3_O_pad[3]	A14	
Y_SDATA4_O_pad[0]	A19	Y output data stream 4	Y_PHASE4_O_pad[0]	B19	Y output phase stream 4
Y_SDATA4_O_pad[1]	E19		Y_PHASE4_O_pad[1]	D8	
Y_SDATA4_O_pad[2]	A12		Y_PHASE4_O_pad[2]	E15	
Y_SDATA4_O_pad[3]	B9		Y_PHASE4_O_pad[3]	F8	
Y_SDATA5_O_pad[0]	C19	Y output data stream 5	Y_PHASE5_O_pad[0]	B18	Y output phase stream 5
Y_SDATA5_O_pad[1]	A16		Y_PHASE5_O_pad[1]	E17	
Y_SDATA5_O_pad[2]	D19		Y_PHASE5_O_pad[2]	C17	
Y_SDATA5_O_pad[3]	E14		Y_PHASE5_O_pad[3]	F17	
Y_SDATA6_O_pad[0]	D7	Y output data stream 6	Y_PHASE6_O_pad[0]	B11	Y output phase stream 6
Y_SDATA6_O_pad[1]	D13		Y_PHASE6_O_pad[1]	A13	
Y_SDATA6_O_pad[2]	D16		Y_PHASE6_O_pad[2]	D15	
Y_SDATA6_O_pad[3]	A9		Y_PHASE6_O_pad[3]	A15	
Y_SDATA7_O_pad[0]	F12	Y output data stream 7	Y_PHASE7_O_pad[0]	D14	Y output phase stream 7
Y_SDATA7_O_pad[1]	G10		Y_PHASE7_O_pad[1]	A17	
Y_SDATA7_O_pad[2]	G18		Y_PHASE7_O_pad[2]	F13	
Y_SDATA7_O_pad[3]	A10		Y_PHASE7_O_pad[3]	F11	
Y_DVALID_O_pad[0]	G15	Y output data valid. Each output is applicable to the associated/ numbered Y output stream	Y_DUMP_EN_O_pad[0]	D6	Y output dump enable. Each output is applicable to the associated/numbered Y output stream
Y_DVALID_O_pad[1]	C15		Y_DUMP_EN_O_pad[1]	C16	
Y_DVALID_O_pad[2]	E9		Y_DUMP_EN_O_pad[2]	B12	
Y_DVALID_O_pad[3]	C10		Y_DUMP_EN_O_pad[3]	D10	
Y_DVALID_O_pad[4]	B10		Y_DUMP_EN_O_pad[4]	D12	
Y_DVALID_O_pad[5]	G17		Y_DUMP_EN_O_pad[5]	C13	
Y_DVALID_O_pad[6]	F16		Y_DUMP_EN_O_pad[6]	F9	
Y_DVALID_O_pad[7]	E13		Y_DUMP_EN_O_pad[7]	C7	
Y_SE_CLK_O_pad[0]	A11	Y output shift-enable clocks (clock enables). Each output is applicable to the associated/ numbered Y output stream	Y_DUMP_SYNC_O_pad	E16	Y output dump sync
Y_SE_CLK_O_pad[1]	G9		Y_TIMESTAMP_O_pad	C8	Y output timestamp
Y_SE_CLK_O_pad[2]	E7		Y_SCHID_FRAME_O_pad_	C6	Y output schid_frame pulse
Y_SE_CLK_O_pad[3]	C14		Y_CLOCK_O_pad	G11	Y output 32 MHz clock
Y_SE_CLK_O_pad[4]	A8				
Y_SE_CLK_O_pad[5]	A5				
Y_SE_CLK_O_pad[6]	B8				
Y_SE_CLK_O_pad[7]	C12				

Signal	Pin	Description	Signal	Pin	Description
LCI_DATA_CS_pad_	D20	LCI output enable, active low	MCB_CLK_pad	H3	33 MHz MCB clk
LCI_DATA_RDY_pad_	E20	Data ready output	MCB_CS_pad_	H2	MCB chip select in-Note 1
LCI_DATA_OE_pad_	E21	Data output enable (input)	MCB_RD_WR_pad_	H1	MCB RD WR_ in Active-low write
LCI_DATA_CLKOUT_pad	C26	128 MHz out clock	MCB_ADDR_pad[0]	E1	MCB addr – bit 0
LCI_FRAME_ABORT_pad_	C25	Frame abort in	MCB_ADDR_pad[1]	E2	
LCI_ACCDATA_VALID_pad_	D25	LTA data valid active low	MCB_ADDR_pad[2]	E3	
LCI_ACCDATA_pad[0]	A21	Output data - 0	MCB_ADDR_pad[3]	E4	MCB addr – bit 3
LCI_ACCDATA_pad[1]	B21	Output data - 1	MCB_DATA_pad[0]	F1	MCB data – bit 0 bidir
LCI_ACCDATA_pad[2]	C21		MCB_DATA_pad[1]	F2	
LCI_ACCDATA_pad[3]	D21		MCB_DATA_pad[2]	F3	
LCI_ACCDATA_pad[4]	A22		MCB_DATA_pad[3]	F4	
LCI_ACCDATA_pad[5]	B22		MCB_DATA_pad[4]	G1	
LCI_ACCDATA_pad[6]	C22		MCB_DATA_pad[5]	G2	
LCI_ACCDATA_pad[7]	B23		MCB_DATA_pad[6]	G3	
LCI_ACCDATA_pad[8]	C23		MCB_DATA_pad[7]	G4	
LCI_ACCDATA_pad[9]	D23		TMS_pad	AD24	JTAG TMS-Note 2
LCI_ACCDATA_pad[10]	A24		TDO_pad	F6	JTAG TDO
LCI_ACCDATA_pad[11]	B24		TCK_pad	AD23	JTAG TCK-Note 2
LCI_ACCDATA_pad[12]	D24		TDI_pad	AE24	JTAG TDI
LCI_ACCDATA_pad[13]	E23		TESTMODE_0_pad	B3	Connect to GND
LCI_ACCDATA_pad[14]	E24		TESTMODE_1_pad	C3	Connect to GND
LCI_ACCDATA_pad[15]	E25		SENSEOUT_pad	A3	Output 3.80 MHz sense clk-Note 3
LCI_ACCDATA_pad[16]	E26		PLL_RESET_pad_	B20	PLL reset low-Note 4
LCI_ACCDATA_pad[17]	F21		RESET_pad_	C20	Chip reset low-Note 4
LCI_ACCDATA_pad[18]	F22		SCAN_EN_pad	D22	Connect to GND
LCI_ACCDATA_pad[19]	F23		TESTMODE_pad	A20	Connect to GND
LCI_ACCDATA_pad[20]	F24		SCAN_CLK_pad	E22	Connect to GND
LCI_ACCDATA_pad[21]	F25				
LCI_ACCDATA_pad[22]	F26				
LCI_ACCDATA_pad[23]	G21				
LCI_ACCDATA_pad[24]	G22		**Special Signal notes:** 1. If MCB_CS_ low, MCB_DATA drivers turned on. 2. JTAG TMS must be held high, and JTAG TCK must be oscillating at ~1 MHz to keep the chip from going into test mode. 3. SENSEOUT_pad connects to an internal ring oscillator that is set for 3.80 MHz output at Vcore=1.02 V nominal. This output increases in freq with increasing Vcore. 4. PLL_RESET_pad_ must be asserted low for minimum 200 nsec, and RESET_pad_ must be held low for minimum 500 microseconds after PLL_RESET_pad_ goes high. 5. All I/Os are 2.5 V TTL levels. 6. Power and GND pins are noted in the next section.		
LCI_ACCDATA_pad[25]	G23				
LCI_ACCDATA_pad[26]	G24				
LCI_ACCDATA_pad[27]	G25				
LCI_ACCDATA_pad[28]	G26				
LCI_ACCDATA_pad[29]	H23				
LCI_ACCDATA_pad[30]	H24				
LCI_ACCDATA_pad[31]	H25	Output data - 31			

9 Appendix II – Ball Map and Power Pins



TOP VIEW

V13, W14, J13, J14 = AGND
"." = DGND
"@" = DGND
"+" = Vcore = 1.02 V
"^" = V I/O = 2.5 V
V14, H13 = V PLL I/O = 2.5 V
nc = no connect
= no ball

10 Appendix III – DC and AC Characteristics

10.1 DC Characteristics

Absolute maximum ratings

Parameter	Value	Units
V I/O supply max operating	2.75	V
Vcore supply max operating	1.32	V
Tj max	125	^{o}C
V I/O signal max	3.6	V
V I/O signal min	-0.5	V
Maximum core power dissipation at Vcore=1.2V, worst-case vectors	3.44	W
Maximum I/O power dissipation	1	W

Normal operating conditions

Parameter	Min	Nominal	Max	Units
V I/O supply[10]	2.375	2.5	2.625	V
Vcore supply	0.9	1.02	1.2	V
Core power dissipation at Vcore=1.02 V, f=256MHz (X/Y clock in=32 MHz), Gaussian noise vectors		1.88		W
I/O power dissipation, f=256 MHz, Gaussian noise vectors		0.7		W

DC I/O characteristics[10]

Parameter	Min	Nominal	Max	Units
VIH – Input high	1.7	2.5	3.0	V
VIL – Input low			0.7	V
VOH – Output high	2.0	2.5		V
VOL – Output low			0.4	V
Load capacitances to meet timing models[11]:				
LCI_DATA_CLKOUT_pad			16.3	pF
X/Y_CLOCK_O_pad			20	pF
MCB_DATA_pad[7:0]			143	pF
LCI*_pad			15.6	pF
X*_pad, Y*_pad			20	pF

[10] The mfg did not provide these numbers; these are taken from Altera Stratix-II data sheet for 2.5 V I/O specifications.

[11] Output drive strengths are not characterized in terms of mA, but rather load capacitance to meet timing models in the remainder of this section.

10.2 Output Timing

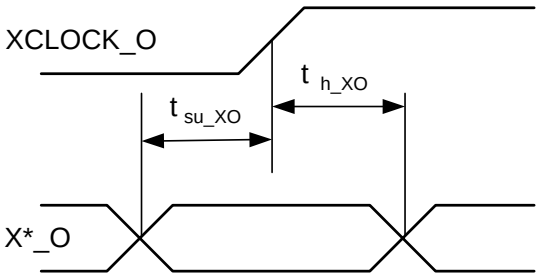
These parameters (except jitter) are derived from the gate-level simulations.

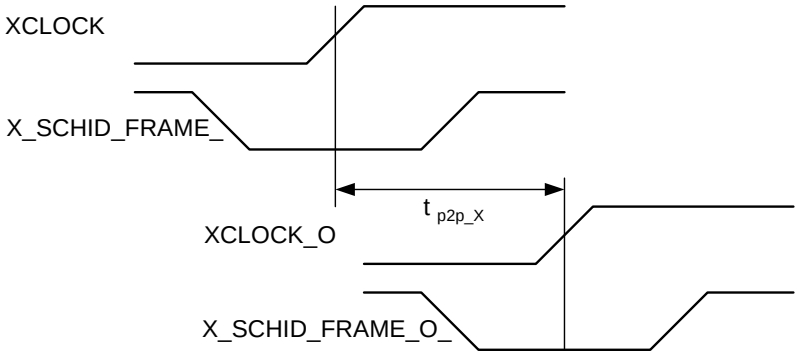
Measurement Description	Timing parameter	Max (nsec)	Min (nsec)	Load cap (pF)
Daisy-chain setup time: X*_O (X outputs) stable relative to ↑ XCLOCK_O	$\mathbf{t_{su\_XO}}$	2.19	2.05	20
Daisy-chain hold time: X*_O (X outputs) stable relative to ↑ XCLOCK_O	$\mathbf{t_{h\_XO}}$	0.69	1.34	20
Daisy-chain setup time: Y*_O (Y outputs) stable relative to ↑ YCLOCK_O	$\mathbf{t_{su\_YO}}$	1.77	1.88	20
Daisy-chain hold time: Y*_O (Y outputs) stable relative to ↑ YCLOCK_O	$\mathbf{t_{h\_YO}}$	0.81	1.39	20
Daisy-chain pin-to-pin input-to-output delay X (measured from X_SCHID_FRAME input mid-cell to X_SCHID_FRAME_O output at XCLOCK_O edge)	$\mathbf{t_{p2p\_X}}$	12.2	8.55	20
Daisy-chain pin-to-pin input-to-output delay Y (measured from Y_SCHID_FRAME input mid-cell to Y_SCHID_FRAME_O output at YCLOCK_O edge)	$\mathbf{t_{p2p\_Y}}$	11.4	7.86	20
LCI data and control signals **max** clock-to-output delay relative to ↑ LCI_DATA_CLKOUT	$\mathbf{t_{co\_max\_LCI}}$	1.5	0.6	15.6
LCI data and control signals **min** clock-to-output delay relative to ↑ LCI_DATA_CLKOUT	$\mathbf{t_{co\_min\_LCI}}$	0.31	-0.17	15.6
MCB DATA read **max** clock-to-output delay relative to ↑ of MCB_CLK	$\mathbf{t_{co\_max\_MCB}}$	13.5	5.8	143
MCB DATA read **min** clock-to-output delay relative to ↑ of MCB_CLK	$\mathbf{t_{co\_min\_MCB}}$	12.2	5.17	143
XCLOCK_O, YCLOCK_O, LCI_DATA_CLKOUT jitter, pk-pk		1[12]	0.2	
X*O, Y*O, LCI*O cycle-cycle jitter, pk-pk		1	0.2	

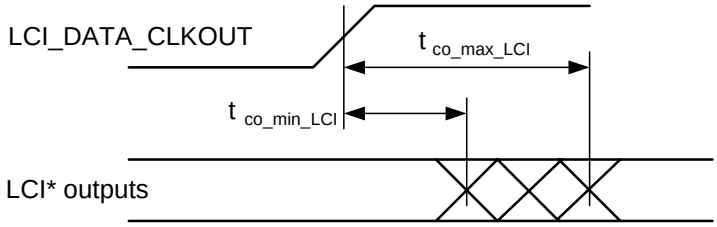
Duty Cycle	Min (%)	Max (%)
XCLOCK_O duty cycle	47.3	48.7
YCLOCK_O duty cycle	47.6	48.8
LCI_DATA_CLKOUT duty cycle	40.1	44.7

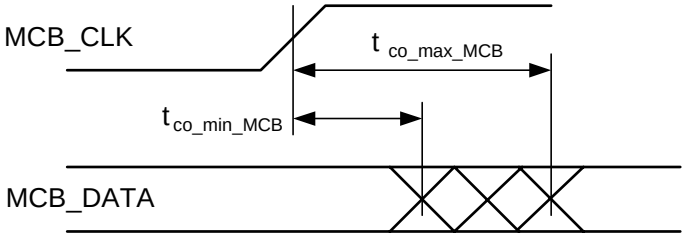
[12] This output jitter occurs when any SE_CLK input is oscillating (i.e. not DC). This effectively renders XCLOCK_O and YCLOCK_O unusable. When all SE_CLK inputs are held high, max jitter is 200 psec pk-pk.

Reference timing diagrams.









10.3 Input Timing Parameters

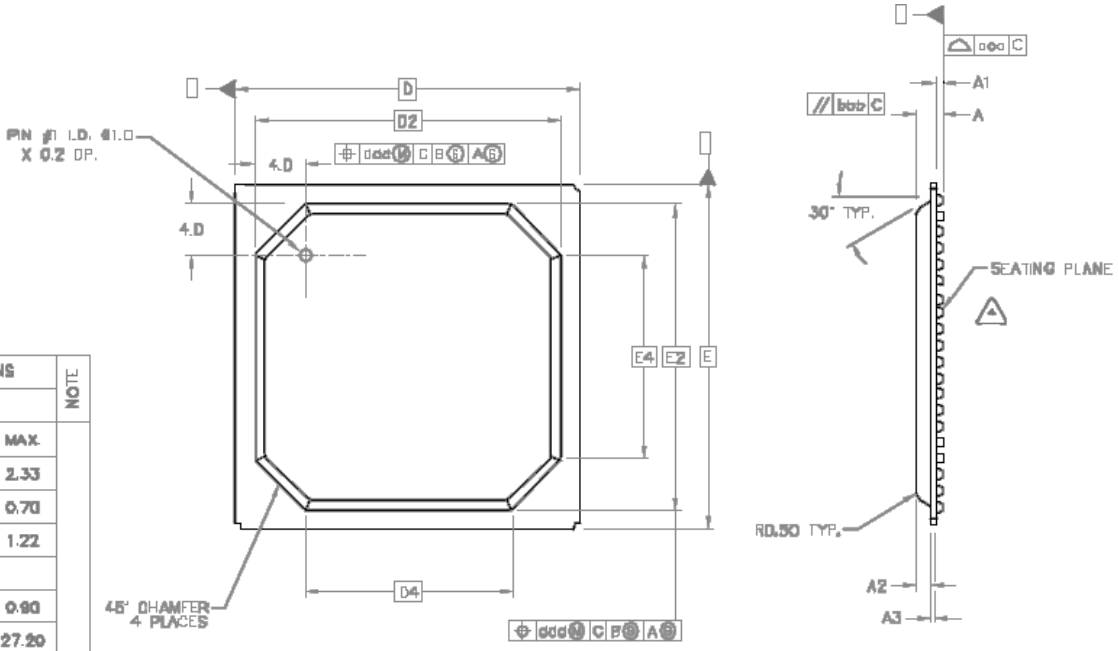
These input timing parameters provided by the mfg.

Description	Timing parameter	Max (nsec)	Min (nsec)	Input cap (pF)
Input setup time, X inputs relative to ↑ X_CLOCK_I	$\mathbf{t_{su\_XI}}$	0.66	0.15	4.7
Input hold time, X inputs relative to ↑ X_CLOCK_I	$\mathbf{t_{h\_XI}}$	0.30	0.25	4.7
Input setup time, Y inputs relative to ↑ Y_CLOCK_I	$\mathbf{t_{su\_YI}}$	0.48	0.26	4.7
Input hold time, Y inputs relative to ↑ Y_CLOCK_I	$\mathbf{t_{h\_YI}}$	0.49	0.28	4.7
Input setup time, LCI inputs (CS, OE, FRAME_ABORT) relative to ↑ LCI_DATA_CLKOUT	$\mathbf{t_{su\_LCI}}$	5.1	2.48	4.7
Input hold time, LCI inputs (CS, OE, FRAME_ABORT) relative to ↑ LCI_DATA_CLKOUT	$\mathbf{t_{h\_LCI}}$	-4.5	-2.23	4.7
Input setup time, MCB inputs (CS, RD_WR, DATA, ADDR) relative to ↑ MCB_CLK	$\mathbf{t_{su\_MCB}}$	3.15	1.27	4.7
Input hold time, MCB inputs (CS, RD_WR, DATA, ADDR) relative to ↑ MCB_CLK	$\mathbf{t_{h\_MCB}}$	-0.49	-0.02	4.7
Minimum RESET_ time	$\mathbf{t_{RESET}}$	12	12	4.7
Minimum PLL_RESET_ time	$\mathbf{t_{PLL\_RESET}}$	200	200	4.7
Minimum time from PLL_RESET_ release to RESET_ release	$\mathbf{t_{P\_R}}$	500 us	500 us	
Input setup time, JTAG inputs (TDI, TMS) relative to TCK	$\mathbf{t_{su\_JTAG}}$	2.56	0.96	4.7
Input hold time, JTAG inputs (TDI, TMS) relative to TCK	$\mathbf{t_{h\_JTAG}}$	-0.14	0.13	4.7
X/Y_CLOCK_I input cycle-cycle jitter tolerance pk-pk		0.5[13]		
Minimum frequency, X_CLOCK_I, Y_CLOCK_I	$\mathbf{freq_{min}}$	25 (MHz)	25 (MHz)	
Maximum frequency, X_CLOCK_I, Y_CLOCK_I	$\mathbf{freq_{max}}$	32 (MHz)	32 (MHz)	
Maximum frequency, MCB_CLK	$\mathbf{MCB_{max}}$	33 (MHz)	33 (MHz)	

[13] Based on empirical tests.

11 Appendix IV – Physical Characteristics

The correlator chip is package in a 27 mm, F672 BGA with 1 mm lead pitch. The following drawings are derived from manufacturer's drawings, and Altera F672 (bottom view) drawings.



SYMBOL	TOTAL # OF PINS			NOTE
	272			
	MIN.	NOM.	MAX.	
A	1.93	2.13	2.33	
A1	0.50	0.60	0.70	
A2	1.12	1.17	1.22	
A3	0.38 REF.			
b	0.60	0.76	0.90	
D	26.80	27.00	27.20	
D1	24.13 REF.			
D2	23.80	24.00	24.20	
D3	3.81 REF.			
D4	16.10 REF.			
E	26.80	27.00	27.20	
E1	24.13 REF.			
E2	23.80	24.00	24.20	
E3	3.81 REF.			
E4	16.10 REF.			
e	1.27 BSC			
aaa	---	---	0.20	
bbb	---	---	0.25	
ddd	---	---	0.20	

NOTES:

1. ALL DIMENSIONS ARE IN MM.
2. DIMENSION TOLERANCES ARE ± 0.1 (ANGULAR TOLERANCE ±3°) UNLESS OTHERWISE SPECIFIED.
3. ALL DIMENSIONS AND TOLERANCES CONFORM TO ANSI Y14.5M–1982.
4. PRIMARY DATUM –C– AND SEATING PLANE ARE DEFINED BY THE SPHERICAL CROWNS OF THE SOLDER BALLS.
5. SOLDER BALL COMPOSITION: Sn 63%, Pb 37%, Ø0.030".
6. PACKAGE EDGE RADIUS 0.60 MM (MAX).
7. PIN ONE DEFINED BY IRREGULAR CORNER AND/OR Au PLATED GATE MARK AND/OR MARK ON THE TOP SIDE OF PACKAGE.
8. Reference: JEDEC MS034 A.

Figure 11-1 Correlator chip top and side view dimension drawings.

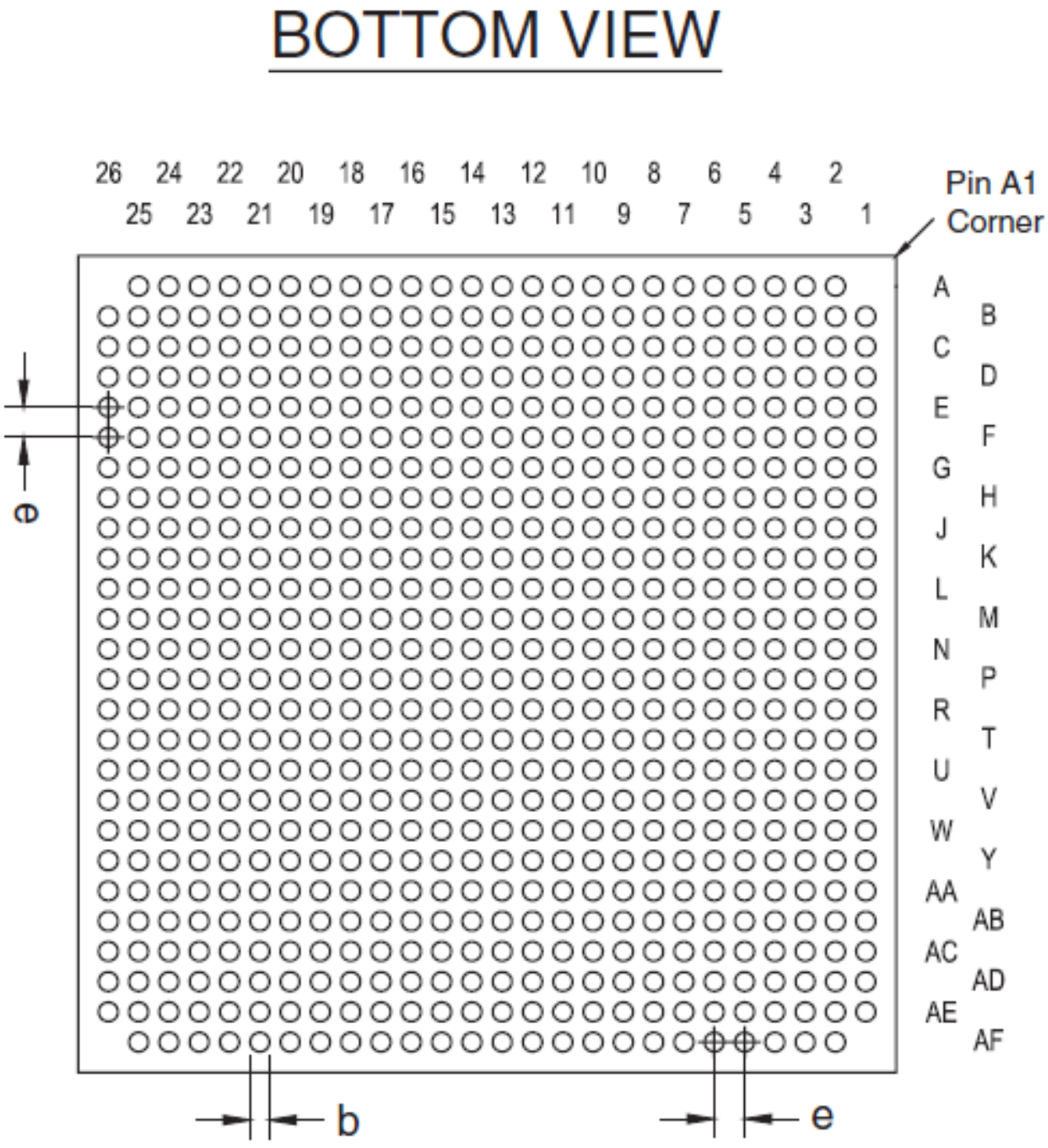


Figure 11-2 Correlator chip F672 bottom view drawing. Dimension “e” is 1.0 mm. Dimension “b” is 0.50 to 0.70 mm, nominally 0.60 mm.

12 Appendix V – Known bugs and workarounds

The only known minor bug/anomaly is the following:

Background:

When a dump occurs (DUMP_SYNC and DUMP_EN for a particular input/CCC asserted), and data has not be read out of the chip's internal buffer from the last dump, the existing accumulators' results are discarded, cleared, and correlation starts again. The next frame that is successfully dumped will have its OVR bit (overrun STATUS bit, bit 25 of W1 of the correlator chip data frame, Figure 5-4) set to indicate that one or more *previous* frames were discarded due to overruns.

Normally the chip should not be operated in this dump overrun condition, as it results in lost data.

Bug/Anomaly Description:

When dump/discard/overrun occurs, "holdoff" signaling, encoded in the DUMP_EN bit stream (bottom right-hand corner of Figure 5-1) is ignored by the chip. This is only a potential problem if holdoff signaling is used, and this is normally only the case if the chip is used for dynamic[14] recirculation to prevent correlation of old time-burst data within the current integration.

Workaround:

When dynamic recirculation is active, any output frames with the OVR STATUS bit set should be discarded to prevent data contamination. When dynamic recirculation is not active, no such action needs to be taken.

[14] Dynamic recirculation is where different recirculation blocks are acquired with each dump.

Index

S

T

V

W

X

Y