REQUIREMENTS AND FUNCTIONAL SPECIFICATION

LTA Controller FPGA

RFS Document: **A25091N0000**

Revision: 2.4

Brent Carlson, July 14, 2011

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

List of Abbreviations and Acronyms

DDR SDRAM – Double Data Rate Synchronous Dynamic RAM. This is dynamic ram that operates at twice the clock speed by clocking data in and out on both edges of the clock.

LTA – Long Term Accumulator.

FPGA – Field Programmable Gate Array.

MCB – Monitor and Control Bus. The on-circuit-board synchronous read/write bus for configuring and monitoring the operation of chips.

LVTTL – Low-Voltage TTL. A chip-to-chip voltage level standard.

SSTL – Stub Series Terminated Logic. A high-speed signaling standard for high-speed intra-PCB communications.

LCI – LTA Controller Interface. The interface between the LTA Controller and the Correlator Chip.

CCC – Correlator Chip Cell. This is one, 128 complex-lag correlator cell within the Correlator Chip. An independent lag frame is produced for each CCC, independent of Correlator Chip configuration.

R2 – Refers to “recirculation 2” in the Recirculation Controller and is always active. R2 supports recirculation on all 8 streams.

1 Revision History

Revision	Date	Changes/Notes	Author
DRAFT	Nov. 5, 2003	Initial DRAFT release	B. Carlson
DRAFT	Feb. 23,2004	2nd DRAFT release with changes incorporated in actual design	B. Carlson
DRAFT	March 1, 2004	3rd DRAFT release. The input clock from the Gbit Ethernet chip is now 31.25 MHz, and multiplied-up in the chip to 125 MHz. This is to improve signal integrity of this clock on the circuit board.	B. Carlson
DRAFT	October 15, 2004	4th DRAFT release. Change the Gbit interface so that it is point-to-point in a ring-style topology.	B. Carlson
1.0	January 11, 2005	First release. No support for R2 parameters different than normal recirculation. LTA outputs are SSTL2-C1. Support for the Accel regulator chip for the Correlator Chip. This is the V1.0 release of the chip Data Products (bitstream, pin file, board files etc)	B. Carlson
1.1	February 16, 2005	Added the Correlator Chip reset and PLL reset capability in the CCPCR register	B. Carlson
1.2	December 14, 2006	Add 125 MHz and 128 MHz PLL lock indicators to the CCPCR register. Add a “semaphore clear inhibit” bit to disable clearing of semaphore table bits for testing.	B. Carlson
1.3	February 8, 2007	Add current status and toggle status registers for the LTA-Correlator Chip interface, and for the communication interface to the GigE FPGA. Add DUMMY read/write register.	B. Carlson

Revision	Date	Changes/Notes	Author
1.31	February 23, 2007	Add built-in DDR SDRAM memory tests. Refer to the CCPCR register. There is no immediate software impact to these additions.	B. Carlson
1.4	August 23, 2007	In the C3FREJCR register if data integrity is max, then OVR (overrun) frames from the Corr Chip are rejected when a particular CCC is in recirculation mode (RE bit of the C3FCSR register; RBE bit indicates error). Allow for double-size RAM in two modes: 2 banks of 2000 bins, or 2 banks of 1000 bins with self-healing. See the MSR register. Remove the VB[3:0] bits from the MSR.	B. Carlson
1.41	May 30, 2008	Add the APR and APRR registers for Accel programming. Upgrade and clarify the description of the SBYCR, NBCR, and TBCR registers.	B. Carlson
1.42	January 9, 2009	Correct and clarify the wording for the NBCR (section 5.4.3.13) and the TBCR (section 5.4.3.14) registers. Add Accel REV. C programming support (sections 5.4.3.31 and 5.4.3.32)	B. Carlson
2.0	September 10, 2009	**Changes to support V2 GigE FPGA**: -Destination MAC now included in this register set. See DMAC-0…5 at addresses 0xA0…0xA5. -Add V1, V0 bits to the MSR to indicate V2-supported design. -Operation at 133 MHz, except for the Corr-Chip to LTA interface, which is still at 128 MHz. Add Corr Chip CCC frame detect status register. See CCCFDSR-0,1 at addresses 0x62, 0x63.	B. Carlson

Revision	Date	Changes/Notes	Author
2.1	February 1, 2010	Upgrade to include appendices with pin file, packaging and general user manual information, including Accel programming information.	B. Carlson
2.2	March 2, 2010	Correct error in the CCPCR register. "PLL125" is really PLL133. Change in the "LTADRR-1" register (addr=0x51). Add a "DEP10" bit to allow for 2 banks of 2000 bins, move the "BNK" bit to bit location3. When memory test fails, the test continues to allow for probing of pins on the RAM.	B. Carlson
2.3	May 25, 2010	Add the LTA Transmit Control Register (LTCR), and the LTA Transmit Holdoff count registers (LTH_COUNT-0, LTH_COUNT-1). These allow for CPU or timed delays from LTA data ready, until output to the GigE FPGA, for lag frame packet scheduling through the network.	B. Carlson
2.3a	August 17, 2010	Add 5 more bits of LTA Transmit Holdoff resolution, so that now there is 31 usec of holdoff resolution. Use the upper 5 bits of the LTCR for these bits. These changes are in effect for any FPGA binaries on or after August 17, 2010. Develop and explain algorithm for highest-performance dumping with deterministic packet transmission performance.	B. Carlson
2.3b	April 20, 2011	Fix MTC description "bug" on page 40; the memory test completes after the PU bit goes high, not after PLL128 goes high.	B. Carlson
2.4	July 14, 2011	Add OFCOUNT-0 and -1 regs at addresses 0xB3 and 0xB4	B. Carlson

2 Introduction

This document describes detailed requirements and design concepts for the LTA (Long-Term Accumulator) Controller FPGA. There is one of these FPGAs for every Correlator Chip on the Baseline Board, and it is responsible for reading out Correlator Chip data frames, accumulating the results in an external RAM, and transmitting the accumulated data to the Gbit Ethernet transmitter FPGA and on to the backend for final processing and eventual archiving. Since there is one of these FPGAs for every Correlator Chip on every Baseline Board, it is important that the implementation fit within an inexpensive FPGA to minimize cost and power requirements.

The RFS specification for communication with the Correlator Chip can be found in [1], and background information on the system design can be found in [2].

This design will probably only ever reside in an FPGA since it is likely that a small size FPGA will be cost-effective. However, if cost becomes an issue, the required quantity (~10,000) may make it feasible to eventually implement in a gate array, and thus the design should be done in a way to ensure easy migration.

3 Context

LTA Controllers reside on the Baseline Board and are used to read out and accumulate Correlator Chip data frames into accumulation "bins" and, when required, transmit frames of accumulated data to the Gbit Ethernet transmitter chip. There is one LTA Controller for every Correlator Chip, and the LTA Controller's external-world connections are to the Correlator Chip, the Gbit Ethernet transmitter, and the external DDR SDRAM. A simplified diagram of these connections is shown in Figure 3-1.

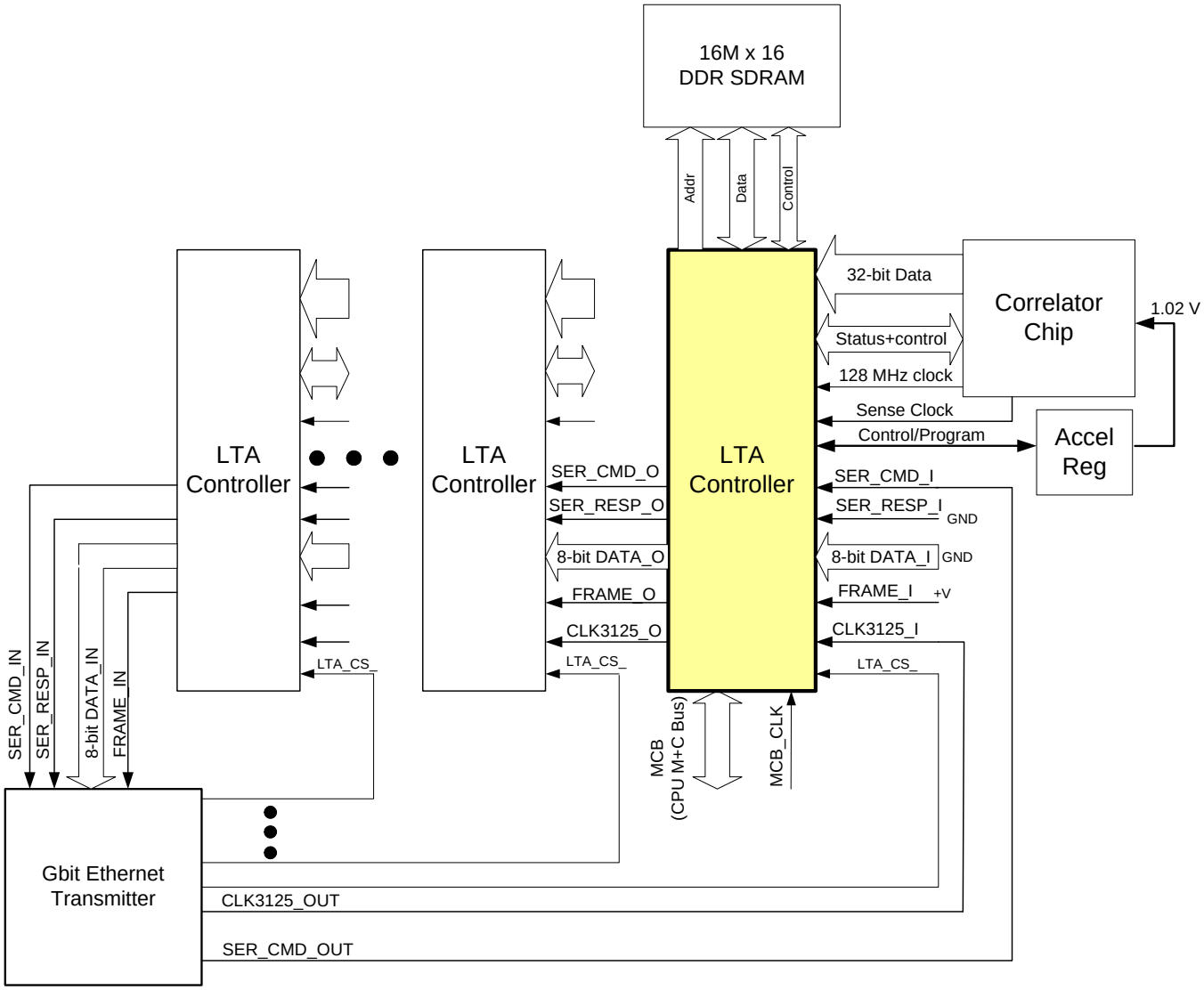


Figure 3-1 Simplified block diagram of the LTA Controller's external-world connections. Data frames are read from the Correlator Chip, accumulated in the external DDR SDRAM (Double-Data Rate Synchronous Dynamic RAM), and eventually transmitted to the Gbit Ethernet transmitter chip. The LTA also controls and programs the correlator chip's point-of-load regulator, the Accel chip.

As shown in Figure 3-1 a simple point-to-point daisy-chain scheme has been developed that pipes the data from multiple LTA Controllers to a single Gbit Ethernet transmitter FPGA. In this scheme, the Gbit Ethernet transmitter FPGA generates a 33.25 MHz clock (**noted in Figure 3-1 as "CLK3125_I/O" for pre-V2 naming consistency**) that is fed to each LTA Controller, which in turn regenerates it with the correct phase to the next LTA Controller. The Gbit Ethernet Transmitter chip does not require the return 33.25 MHz clock since it uses its own internal clock and DDR sampling to synchronize to the incoming data and control streams (it does this since there are 8 sets of incoming signals—each one from a row of LTA Controllers—refer to [4]). A 33.25 MHz clock rate is chosen for ease of transmission to the LTA Controllers, and to allow for the highest speed LTA-to-DDR SDRAM data transfer rate (translation to lower 125 MHz for GigE transmission is in the Gbit Ethernet chip). The LTA Controller multiplies the 33.25 MHz clock to 133 MHz for internal use—the SER_CMD, DATA, and FRAME signals operate w.r.t. this internal 133 MHz clock. The CLK32125_OUT and SER_CMD_OUT signals must be phase synchronous when they arrive at the first LTA Controller in the daisy chain.

A particular LTA Controller, under direction from the Gbit Ethernet chip, is enabled to generate data on its outputs when its particular LTA_CS_ is asserted. When the LTA Controller detects that its LTA_CS_ input is asserted, it routes its own internal SER_RESP, DATA, and FRAME to its outputs. When the LTA_CS_ is negated, the LTA Controller routes the INPUT signals to the output, for use by another controller somewhere down the daisy chain. Each LTA Controller always passes its SER_CMD_I input to its SER_CMD_O output, unobstructed.

There are 8 rows of LTA Controllers handled by one Gbit Ethernet Transmitter chip. Further information on how this is handled by the Gbit Ethernet chip can be found in [4].

4 Overview

A simplified block diagram of the LTA Controller FPGA is shown in Figure 4-1.

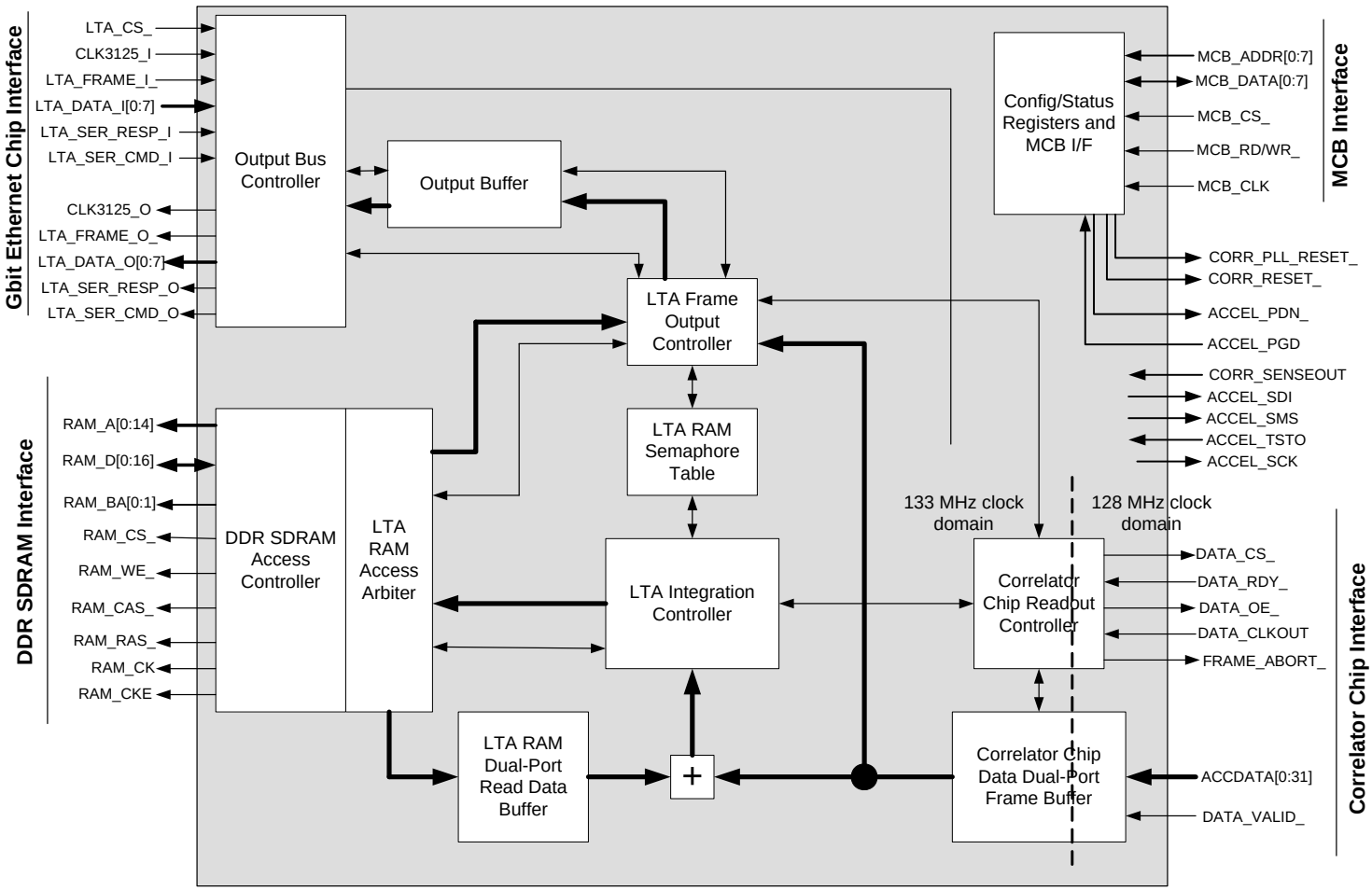


Figure 4-1 Simplified block diagram of the LTA Controller FPGA. The device reads out data from the Correlator Chip, adds it to existing data in the LTA, and saves it back into the LTA. When LTA data is ready, the LTA Integration Controller sets a semaphore in the LTA RAM Semaphore Table. The LTA Frame Output Controller looks for semaphores in the table, and when one is found, arbitrates for the LTA RAM and transfers data to an output buffer (Output Buffer A or B). The Output Bus Controller talks to the Gbit Ethernet chip and transfers data to it on the LTA_DATA[0:7] bus, when LTA_CS_ is asserted. When the data bin is empty, the semaphore is reset, indicating that the bin is again ready for accumulation. Optionally, the data bin can be zeroed after it is finished being used.

The chip consists of several separate but interconnected blocks. A brief description of the operation of each of these blocks is as follows:

- **Correlator Chip Readout Controller.** This block talks to the Correlator Chip via the *LTA Controller Interface* (LCI), and handles lag frames from the Correlator Chip, at 128 MHz. When a lag frame is ready, it is read out and, if

necessary, stored in the *Correlator Chip Dual-Port Frame Buffer*. Once lag frame header information is available, one of several things can happen depending on the situation, all in close communication with the *LTA Integration Controller* block. (These are described in more detail in the description of the *LTA Integration Controller* block.) Following direction from the *LTA Integration Controller*, the frame is stored in the *Correlator Chip Data Dual-Port Frame Buffer* OR the frame is aborted by asserting the FRAME_ABORT_ signal to the Correlator Chip. When signaled that the current frame has finished processing, the next frame from the Correlator Chip is handled. Note that for speed, this will happen in a pipelined fashion—data will start to be read into the frame buffer before the frame buffer is emptied of its previous contents.

- **LTA Integration Controller.** If the frame from the Correlator Chip is to be accumulated into an accumulation bin this controller determines from the semaphore table the status of the accumulation bin that the data is destined for. If the accumulation bin is full and waiting to be read out, then the *Correlator Chip Readout Controller* is signaled that the frame is to be aborted since it is not possible to do anything with it. If the accumulation bin can be used, then if necessary, the existing data from the LTA RAM (DDR SDRAM) is fetched and stored in the LTA RAM Dual-Port Read Data Buffer at the *same time* as data from the Correlator Chip is being stored into its frame buffer. Once both buffers are full, the controller arbitrates for the LTA RAM, removes the DATA_BIAS from the Correlator Chip frame, adds it to the RAM frame, and stores it back into the LTA. If the dump command in the lag frame indicates that it is a speed dump, the frame is stuffed directly into the *Output Buffer* if available. If the output buffer is not available, the frame is discarded.

- **LTA RAM Access Arbiter.** This block is the arbiter that determines whether the *LTA Integration Controller* or the *LTA Frame Output Controller* has sole access to the DDR SDRAM. This is necessary since the DDR SDRAM is single-port access only. A configuration bit in the chip set via the MCB will allow for exclusive access to either the integration controller or the frame output controller to allow for uninterrupted burst-mode operation.

- **DDR SDRAM Access Controller.** This block provides access to the DDR SDRAM and handles all commands, refresh and interleaving functions. Since the SDRAM is double data rate and 16 bits wide, this controller provides an interface that is single data rate (i.e. 133 MHz), and 32 bits wide. The width is convenient since it is the same width as the Correlator Chip data frame, and has the same data rate as lag frames from the Correlator Chip.

- **LTA RAM Semaphore Table.** This is an on-chip dual-port RAM that indicates, for every accumulation bin, the status of the bin. If an accumulation bin is ready for readout, this bit gets set by the *LTA Integration Controller*. Once set, the *LTA Integration Controller* will not write any data into the bin until it has detected that it is cleared (reset). The *LTA Frame Output Controller* parses this table looking for set bits. If a bit is set, and enabled to do so, the controller arbitrates for the

LTA RAM, reads out the frame, stuffs it into an output buffer (buffer A or B), and then resets the semaphore.

- **LTA Frame Output Controller.** As mentioned above, the LTA Frame Output Controller parses the semaphore table for set bits, fetches ready data from LTA RAM, and stuffs it into the output buffer. Additionally, this controller looks for indications of speed dump frames from the *Correlator Chip Readout Controller*. If a speed dump frame is detected, it stuffs it into the output buffer if available—if the output buffer is not available, the frame is discarded. This controller also rapidly parses the semaphore table to get a count of the number of semaphore bits that are set. It provides an external indication of the "fullness" of the LTA from this count, and reflects this information in an MCB status register, and on the LTA_SER_RESP line when so queried on the LTA_SER_CMD line. These bits can be read by the external Gbit Ethernet chip so that it can prioritize the unloading of data from all of the LTA Controllers on the Baseline Board.

- **Output Bus Controller.** This controller is responsible for the interface to the Gbit Ethernet Transmitter chip, and transmits ready frames from the output buffer when enabled.

- **Config/Status Registers and MCB I/F.** This block contains microprocessor-accessible status and control registers. A detailed description of all control and status registers is provided in the interface requirements section 5.4.

4.1 Correlator Chip Interface Signals (LCI)

These signals are described in detail in the Correlator Chip RFS document [1], but the description of these signals from the point of view of the LTA Controller is included for completeness. **All signals on this interface are 2.5 V LVTTL**.

- DATA_CS_ - Active-low signal that, when asserted, causes all of the inputs driven by the Correlator Chip to become active, and causes the Correlator Chip to respond to output signals on this interface.

- DATA_RDY_ - This is asserted active-low by the Correlator Chip indicating that it has a lag frame ready for transmission. It stays asserted until the frame is finished transmission, or DATA_CS_ is negated. If another frame is then ready, this signal is re-asserted by the Correlator Chip.

- DATA_OE_ - This is asserted active-low by the LTA Controller when it wishes to receive frame data from the Correlator Chip. This must be held low for the duration of transmission, and may only be released once DATA_RDY_ is negated, or unless FRAME_ABORT_ is asserted.

- DATA_CLKOUT – This is a 128 MHz clock that is phase-synchronous with all input signals on this interface. All input signals on this interface change on the rising edge of this clock, and all output signals on this interface should change

after its rising edge. (Thus, the inverse of this clock should be used to clock data into the LTA Controller). Note that this clock will not be active until the Correlator Chip is powered up and the CORR_PLL_RESET_ line is brought high.

- FRAME_ABORT_ - This output is asserted active low and used to abort transmission of a frame from the Correlator Chip. It should only be asserted once DATA_RDY_ and DATA_OE_ are asserted. Shortly after this signal is asserted, the Correlator Chip responds by negating DATA_RDY_, at which point DATA_OE_ should be negated. If a frame is aborted, it is cleared from the Correlator Chip's frame buffer and lost forever.

- ACCDATA[0:31] – Frame data from the Correlator Chip.

- DATA_VALID_ - This signal is asserted by the Correlator Chip active low to indicate that data on ACCDATA[0:31] is valid. If this is high while DATA_RDY_ is low, it indicates that ACCDATA[0:31] does not contain valid data, and should not be clocked into the input buffer. However, DATA_VALID_ high while DATA_RDY_ is asserted does not indicate that the frame has finished transmission. Only the negation of DATA_RDY_ indicates the completion of frame transmission from the Correlator Chip.

- CORR_RESET_ – This signal is used to reset the Correlator Chip. It is controlled by the CR bit of the CCPCR register. This output should have a weak pulldown resistor so that when the FPGA is not programmed, it is low.

- CORR_PLL_RESET_ – This signal is used to reset the Correlator Chip's PLL. It is controlled by the CPR bit of the CCPCR register. This output should have a weak pulldown resistor so that when the FPGA is not programmed, it is low.

- CORR_SENSEOUT – This connects to the ring oscillator inside the correlator chip; the frequency of which depends on the applied core voltage and the inherent speed of the device. The LTA FPGA turns this around and feeds it to the Accel regulator via the ACCEL_SCK signal, if the Accel regulator is not being programmed.

4.2 Accel Regulator/Corr Chip Power Interface Signals

All signals on this interface are 2.5 V LVTTL.

- ACCEL_PDN_ – This signal controls the Accel regulator chip. When low, the Accel chip is powered UP. When high, it is powered DOWN. The final signal to the Accel chip is inverted thru an N-channel FET. **This output must have a weak pulldown resistor so that when the FPGA is not programmed, it is low, thereby allowing the Accel chip to function for JTAG test**. Refer to the CCPCR register PU bit, which is inverted and then drives this signal.

- ACCEL_PGD – This signal comes from the Accel chip and indicates if the Accel regulator is active and power is good. Refer to CCPCR register, PGF bit.

- ACCEL_SDI – Serial data output for programming the Accel regulator.

- ACCEL_SMS – Serial Mode Select—output control line for Accel regulator programming.

- ACCEL_TSTO—Accel serial test output. This reflects the ACCEL_SDI signal and can be used to determine if the Accel regulator has been successfully programmed.

- ACCEL_SCK—Accel serial clock (programming), or fed back CORR_SENSEOUT clock (normal operation).

- CLK3125_ACCEL – 0.947 MHz reference clock for the Accel regulator—**this is not connected**, rather the Accel regulator gets its clock from the GigE FPGA via TCK distribution for JTAG operation.

4.3 DDR SDRAM Interface Signals

These signals are specific to a DDR SDRAM (Double Data Rate, Synchronous Dynamic RAM), and their complete description is beyond the scope of this document. The LTA Controller design utilizes an Altera DDR SDRAM controller IP core and so detailed knowledge of this interface is not required. The RAM that the design is targeted to is the Micron Technology MT46V16M16FG-75E (4 MEG x 16 x 4 Banks), a 256 Mbit DDR SDRAM. However, it is possible with a different core configuration (and thus a different FPGA "personality") to support 512 Mbit, 1 Gbit, and 2 Gbit RAMs, which provide 4000, 8000, and 16000 accumulation bins respectively. **All signals on this interface are SSTL2-Class I**. Signals are as follows:

- ddr_a[12:0] – Address.

- ddr_ba[1:0] – Bank.

- ddr_cs_n – Chip select.

- ddr_cke – Clock enable.

- ddr_ras_n – RAS.

- ddr_cas_n – CAS.

- ddr_we_n – Write enable.

- ddr_dm[1:0] – Data mode.

- ddr_dq[15:0] – Bi-directional data bus.
- ddr_dqs[1:0] – Data strobe.
- clk_to_sdram, clk_to_sdram_n – 133 MHz differential clock to DDR SDRAM.
- ddr_spare1, ddr_spare2 – Spare lines for DDR RAM address expansion.

4.4 Gbit Ethernet Chip Interface

This interface facilitates the transfer of LTA data to the Gbit Ethernet transmitter chip and eventually out to backend computers. **All signals on this interface are SSTL2-Class I**. Refer to section 5.4.2 for protocol information and functional timing diagrams of signals on this interface. There are input and output signals on this interface that facilitate a daisy-chaining configuration as shown in Figure 3-1.

- LTA_CS_ - This active low signal is asserted to route internal signals to outputs on this interface and cause the chip to respond to the LTA_SER_CMD_I input. When this signal is not asserted, the chip passes all inputs to the outputs on this interface unimpeded (**except for 3 pipeline delays**), and does not respond to signaling on LTA_SER_CMD_I.
- CLK3125_I – 33.25 MHz clock (original "CLK3125_I" name kept for backwards compatibility) supplied to the chip from either the Gbit Ethernet transmitter chip or the previous LTA Controller in the daisy chain. This clock is multiplied up to 133 MHz inside the chip. All input signals are sampled on the rising edge of the 133 MHz clock, and all outputs change after the rising edge of the clock.
- LTA_SER_CMD_I – This input is a serial command line utilizing a simple protocol that allows the Gbit Ethernet chip to query the selected LTA Controller about its status and to request transmission of data on LTA_DATA_O[0:7]. This signal always passes, unimpeded, to the LTA_SER_CMD_O output. A full definition of this protocol is defined in section 5.4.2.
- LTA_SER_RESP_I – This is the serial response input from the previous LTA Controller. If this particular chip's LTA_CS_ line is asserted, then this input is ignored, otherwise it passes through to the LTA_SER_RESP_O output.
- LTA_FRAME_I_ – This signal is from the previous LTA Controller and is held low during data transmission to indicate that a valid frame is on the LTA_DATA_I lines. If LTA_CS_ is asserted, this input is ignored, otherwise this signal is passed to the LTA_FRAME_O output.
- LTA_DATA_I[0:7] – This input is from the previous LTA Controller and contains data from one LTA Controller higher-up in the daisy chain. If LTA_CS_ is asserted, this input is ignored, otherwise this signal is passed to the LTA_DATA_O[0:7] output.

- CLK3125_O – This output is phase synchronous with the CLK3125_I input, and is always active. It is 33.25 MHz.

- LTA_SER_CMD_O – This output is, at all times, a replication of the LTA_SER_CMD_I input.

- LTA_SER_RESP_O – If LTA_CS_ is asserted, this signal sources from the chip's internal logic and indicates a response to a previous command on LTA_SER_CMD_I. Refer to the protocol in section 5.4.2 for more information. If LTA_CS_ is not asserted, this output is a replication of the LTA_SER_RESP_I input.

- LTA_FRAME_O – This signal is the output frame signal that, when low, indicates there is valid frame data on LTA_DATA_O[0:7]. If LTA_CS_ is asserted (low), then the source of this signal is from this chip. If it is high, then this output is a replication of the LTA_FRAME_I_ input.

- LTA_DATA_O[0:7] -- This output is frame data when LTA_FRAME_O_ is low. If LTA_CS_ is low, then this data sources from this chip, otherwise this data is a replication of the LTA_DATA_I[0:7] input.

4.5 MCB Interface I/O Signals (Config/Status Registers Block)

This is the microprocessor monitor and control interface. **All signals on this interface are 2.5 V LVTTL**. Refer to section 5.4.3 for more detailed information on monitor and control of the chip. The maximum clock to output (MCB_CLK to MCB_DATA) delay is 11 nsec.

- MCB_ADDR[0:7] Microprocessor address for accessing chip configuration and status registers.
- MCB_DATA[0:7] Bi-directional 8-bit microprocessor data bus.
- MCB_CS_ Input chip select that enables the MCB interface drivers and internal circuitry.
- MCB_RD/WR_ Input that when low enables writing data into the chip. Otherwise, data is driven onto the MCB_DATA bus by the chip, when MCB_CS_ is asserted.
- MCB_CLK Input clock for the synchronous MCB interface. The phase and frequency of this input is independent of the main DATA_CLKOUT. This clock has a nominal frequency of 33 MHz.

5 Requirements

The following is a list of LTA Controller requirements.

5.1 Functional Requirements

1. It must be possible to maintain data integrity at all times, even under extreme performance conditions. Any data frames that are produced must contain good data, unless they are flagged as bad coming from the Correlator Chip. The device must save, retrieve, and transmit as much good data as possible.

2. It must be possible to gather statistics on the internal operation of the LTA Controller for *one lag cell (i.e. CCC) at a time*. The microprocessor must be able to set the lag cell on which statistics will be gathered. The statistics that must be gathered are:

    a. Total number of frames rejected from the Correlator Chip. This could be because of bad transmission, an overflow or synchronization error in the Correlator Chip, the frame was too long or too short, inconsistent recirculation block numbers, or because the LTA bin the data is destined for is full.

    b. Binary indication for each of:
        i. Frames too short or long detected and rejected.
        ii. Transmission error (parity error or sync word error) from Correlator Chip detected and frames were rejected.
        iii. Overflow or sync error status bit set in lag frame from Correlator Chip. Depending on a configuration bit setting, this may or may not result in the rejection of data frames.
        iv. Inconsistent recirculation block numbers and one or more frames have been rejected.
        v. LTA bin is full and one or more frames have been rejected.
        vi. Frame detected.

    c. LCI frame count—the number of frames detected/transmitted from the Correlator Chip since the last time this count was read. This is an 8-bit count, cleared on read that stops at maximum value.

    d. LTA RAM accumulation count—the number of frames accumulated into the LTA for the particular CCC since the last time this count was read. This is an 8-bit count, cleared on read that stops at maximum value.

    e. Output frame count—the number of frames transmitted to the Gbit Ethernet chip for this CCC since the last time this count was read. This is an 8-bit count, cleared on read that stops at maximum value.

f. Speed dump frame count—the number of speed dump frames detected, and the number of speed dump frames rejected or dropped since the last time this count was read. This is an 8-bit count, cleared on read that stops at maximum value.

If possible, the design should allow for capture of all CCCs' statistics in the above manner all of the time, but this could violate the requirement to keep the cost of the FPGA to below $20 in 10k quantity.

3. The depth or "fullness" measure of the LTA put out on the LTA_SER_RESP line. The depth indicates, for the selected bank, how many bins have at least one CCC ready to be read out. Thus, this depth can have a maximum value of 1000, if there are a total of 2000 phase bins available.

4. The design must support an effective LTA memory that is 8M x 32 and is normally implemented with a 16M x 16 DDR SDRAM RAM. The FPGA pin-out definition must support up to a 2 Gbit DDR SDRAM (effective 64M x 32, allowing for 16,000 bins), and provide an indication to the microprocessor via a read-only register what RAM size the chip is set for. It is recognized that support for different RAM sizes will probably require different FPGA personalities, but the microprocessor interface (register set) should remain the same.

5. It must be possible to put the chip in a "test-mode" within which it is possible to read and write from/to the LTA RAM and the Semaphore Table. Transitioning to and from this mode must not, in and of itself, corrupt either the LTA RAM or the Semaphore Table. This mode can be used for power-on diagnostic memory testing, for capturing LTA frames, and for testing the connection to the Gbit Ethernet chip independently of the Correlator Chip, but filling the RAM with frame data. When in test mode, no frames are accepted from the Correlator Chip, and no frames are transmitted to the Gbit Ethernet chip.

6. It must be possible to enable or disable "burst-mode" operation. In this mode, a control bit enables bursting of data into the LTA without interruption by the Frame Output Controller block. When this control bit is inverted, data is blocked from being saved into the LTA so that the Frame Output Controller block has free and uninterrupted access to the LTA RAM. Blocking data from being saved into the LTA means not accepting any more frames from the Correlator Chip, and this could cause overrun indications in future Correlator Chip data frames. <u>If burst-mode is disabled, the chip operates in as simple a fashion as possible, saving data into the LTA, transmitting data to the Gbit Ethernet chip, and arbitrating for LTA RAM on a first-come, first-serve basis.</u>

7. Must be able to enable or disable the ability to automatically clear the LTA RAM after a frame has been read out. Clearing the LTA RAM is not normally necessary, but adds some robustness in the system to handle the occasional dump control (DUMPTRIG) dropout. (i.e. when the RAM is cleared, it doesn't matter

if the initial dump command is received or not, or whether the frame count is the same and expected value for each bin.)

8. The design must maximize frame rate processing from the Correlator Chip under all conditions except where burst mode is selected and frames are not accepted from the Correlator Chip. What this means is that the LTA Integration Controller has highest priority when accessing the LTA RAM, however, any controller that has obtained access to the LTA RAM does not have to relinquish it until it is finished its current frame processing.

9. The LTA Frame Output Controller must be able to be told, via the LTA_SER_CMD line which bank to search for and empty out full bins. When the search bank is selected, searching and stuffing of full bins within the selected search bank are put into the output buffer for transmission to the Gbit Ethernet chip. This functionality allows the Gbit Ethernet chip to prioritize emptying of full bins in burst mode and normal operating modes.

10. Although the minimum cell size on the Correlator Chip is 128 lags, it must be possible to set, on a cell-by-cell (i.e. CCC-by-CCC basis) whether the full number of lags are transmitted to the Gbit Ethernet chip, or whether only 64 of the *center* lags are transmitted to the Gbit Ethernet chip. Support for transmitting fewer than 64 lags is not provided because Gbit Ethernet frames are padded to a certain minimum length and this negates any frame rate advantage that might be obtained by transmitting fewer than 64 lags.

11. Must be able to set the 6-byte destination MAC address that is to be transmitted to the Gbit Ethernet chip, to be used in the subsequent IEEE frame destination MAC.

12. Must be able to set the destination IP address that is to be transmitted to the Gbit Ethernet chip for eventually setting the destination IP address in the output Ethernet frame. Must be able to set whether the destination IP address is 4 bytes (32-bit), or 8 bytes (64-bit) to support v4 and v6 IP respectively. The LTA Controller will transmit the address at the beginning of frame transmission to the Gbit Ethernet chip. Only v4 IP is supported.

13. Must be able to set a 16-bit board S/N or identifier in a register on the LTA Controller. This 16-bit ID is transmitted as part of the data frame to the Gbit Ethernet transmitter chip. Refer to section 5.4.2.2 for more information on the output data frame format.

14. Must be able to set a 6-bit “ChipID” that indicates the X, Y location of the LTA Controller/Correlator Chip pair in the 8x8 matrix of chips. This 6-bit ChipID is transmitted as part of the data frame to the Gbit Ethernet transmitter chip.

15. It must be possible to set whether or not recirculation is being used for each CCC that the controller will process from the Correlator Chip.

16. It must be possible to set "Start_BlockY", "NBlocks", and "TotBlocks" for this Correlator Chip/LTA Controller combination. These numbers are only useful if recirculation is active for a particular CCC. "Start_BlockY" indicates the "start Y recirculation block number" that is being acquired by this Correlator Chip. Nblocks indicates the **number of lag blocks-*1* that are being acquired by this Correlator Chip**. "TotBlocks" indicates the ***total* number of lag blocks-*1*** this Correlator Chip is part of. Start_BlockY, NBlocks, and RECIRC_BLK-Y (from the Correlator Chip data frame) are used to calculate the LTA bin the data is destined for, for a particular phase bin according to the following equation:

    LTA_bin_# = Phase_bin x (Nblocks+1) + (RECIRC_BLK_Y- Start_BlockY)

    Start_BlockY and Nblocks are included in the output data frame transmitted to the Gbit Ethernet chip[1] so the backend can easily recalculate the Phase_bin number. TotBlocks is used by the LTA Controller to ensure that the RECIRC_BLK-Y and RECIRC_BLK-X parameters in the Correlator Chip data frame are self-consistent (i.e RECIRC_BLK-Y + RECIRC_BLK-X = TotBlocks). TotBlocks is not part of the output data frame.

    Example: if 64 lag blocks are being acquired, and this Correlator Chip is acquiring lag blocks 16 to 23, then Start_BlockY=16, NBlocks=7, and TotBlocks=63.

    If recirculation is not active for a particular CCC, then Start_BlockY is set to "0" and NBlocks is set to "0" in the output data frame.

17. The DATA_BIAS in the Correlator Chip data frame must be removed from the data before accumulation in the LTA RAM occurs. This is done to maximize LTA hardware integration time. Thus, LTA RAM output lag frames no longer contain this bias as a separate number, or as part of the lag data itself.

5.2 Performance Requirements

1. The main clock domain of the chip is 133 MHz, and is synchronized to (derived from) the 33.25 MHz clock provided to it by the Gbit Ethernet chip.

2. The interface to the Correlator chip must operate in the 128 MHz clock domain, with the clock provided to the LTA from the Correlator Chip. This Correlator Chip clock contains excessive jitter, and so it must be used in raw mode.

3. The synchronous MCB interface shall be capable of operating with a clock that is neither frequency nor phase synchronous with the 128 MHz or 133 MHz clocks. The chip will support an MCB interface clock with a maximum rate of 33 MHz.

[1] And, for this reason, the LTA RAM should be allowed to completely empty before a configuration change in upstream hardware is made that has an impact on these parameters for any CCC.

4. The power dissipation should be as low as possible (~<1 W) since there are 64 of these devices on the board. If power exceeds this limit, it will be possible to use small heatsinks.

5.3 Environmental Requirements

1. The LTA Controller FPGA will be surface-mounted on the Baseline Board motherboard, probably on the opposite side of the board as the Correlator Chip. Providing optimum decoupling capacitor placement for this device may be problematic and so every effort should be made to slew-rate control any output drivers to minimize high-frequency current demands.

2. This FPGA should be kept physically as small as possible so that it, and the DDR SDRAM, can fit on the opposite side of the circuit board as the Correlator Chip, and minimize interference with Correlator Chip decoupling capacitor placement. The current plan is to implement the design in an F256 Fine Line BGA that is 17 mm (0.7") on a side.

3. I/O levels to this device are either SSTL2 Class-II or 2.5 V LVTTL as defined in section 4.

4. All connections between this chip and the Correlator Chip and between this chip and the DDR SDRAM are point-to-point, un-terminated. The connections on the MCB interface are point-to-multipoint and final connections to the PCMC card are via a series of bus transceivers. The connections to the Gbit Ethernet chip are point-to-point, and any terminations are provided externally.

5. Since the output of the Correlator Chip (the LCI) can go into the high-impedance state, it may be desirable to include integrated pull-up resistors in the LTA Controller on these lines.

6. There are no specific voltage level requirements for this chip. If necessary, a separate power supply and power distribution plane will be provided to deliver power to these devices.

5.4 Interface Requirements

1. The DDR SDRAM interface and operating requirements may be determined by referring to the Micron Technology MT46V16M16FG-75E data sheet.

2. I/Os to/from this chip are LVTTL and SSTL2-CII. Drivers shall have appropriate drive strengths and slew rate for the particular interface.

Additional interface requirements are defined in the following sub-sections.

5.4.1 Correlator Chip (LCI) Interface Requirements

This interface and the contents of the Correlator Chip lag data frame are described completely in the Correlator Chip RFS document [1]. For completeness, the Correlator Chip data frame is shown in Figure 5-1.

Correlator Chip Output Data Frame

	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 0 0 0 1 1 1 0		0 1 1 1 0 0		W265
	CHECKSUM								W266

Figure 5-1 Raw Correlator Chip data frame. The "Cmmd" is always 011b. A complete description of the contents of this data frame is contained in the Correlator Chip RFS document [1].

5.4.2 Gbit Ethernet Transmitter Interface Requirements

This interface is used to transfer LTA data frames to the Gbit Ethernet transmitter chip. Figure 5-2 is a functional timing diagram of this interface.

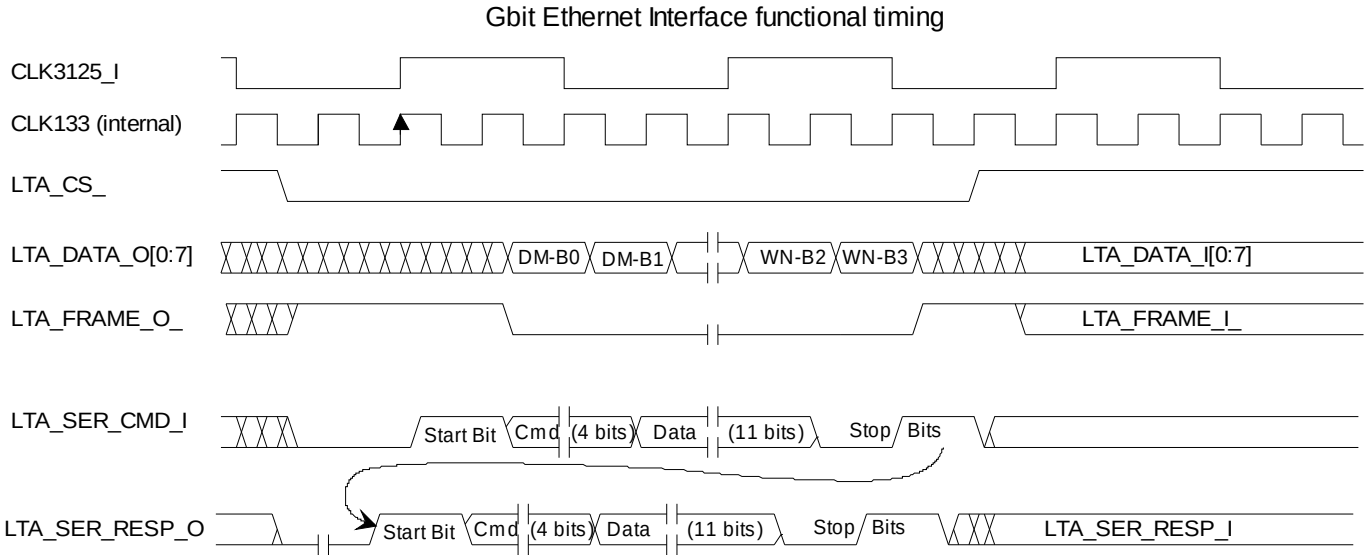


Figure 5-2 Functional timing diagram of the Gbit Ethernet transmitter interface. When LTA_CS_ is high, input signals are passed through to the outputs, and the chip does not respond to any requests. When LTA_CS_ is asserted low, the chip responds to commands and routes internal signals to outputs. The serial command (LTA_SER_CMD) and response (LTA_SER_RESP) lines implement a simple protocol to allow the Gbit Ethernet chip to query the LTA Controller and command it to do things like transmit data frames on LTA_DATA. All commands, data, and responses are transmitted LSB first. After a command is received, the response must be received within 64 bit times, or the master will assume an error and retry.

5.4.2.1 Gbit Ethernet Interface Serial Command/Response Protocol

In this protocol the LTA Controller is the server or slave, and the Gbit Ethernet chip is the client or master. The master initiates action by sending a command on the LTA_SER_CMD line, recognized by the slave only if LTA_CS_ is asserted. Within 64 bit times after finishing transmission, the master expects a response on LTA_SER_RESP to be initiated, otherwise it will assume there was an error in transmission and will report an error and try again. For every command there is exactly one response. Reception by the master or slave will only be considered complete if the serial protocol requirements are met (start bit, command/response, data, stop bits), and the command or response word is valid. Any error detected in reception results in ignoring the contents of the data packet.

Commands, responses, and data contents are defined in the following table.

CMD	CMD Bit Encoding	CMD Data	RESP	RESP Bit Encoding	Response Data
FIQ	0001	D0: bank select	FIQR	0001	D0…D9: fullness count[2] D10: bank bit
BSB	0010	D0: bank select	BSBR	0010	None
TXR	0100	None	TXRR	0100	D0: 1 if ready;
TXREQ	1000	None	TXREQACK	1000	D0: 1 if transmission will occur

Table 5-1 Simple serial command—response protocol.

A description of each command and its response in Table 5-1 is as follows:

FIQ – Fullness inquiry. This command requests a count of the number of bins in the Semaphore Table, for the selected bank that have at least one CCC ready for readout. The response (FIQR) is the fullness count (0…1000), and the bank in which the count occurred. Based on this count, the Gbit Ethernet chip can control which bank the LTA Controller outputs data from. Note that the LTA Controller automatically counts the number of full bins—this command and its response is just reporting what was found. Since the fullness count is obtained in the 128 MHz clock domain, this command should be executed twice in succession, and yield the same answer both times to ensure a valid reading.

BSB – Bank select bin search. This command allows the Gbit Ethernet chip to select which bank (0 or 1) the LTA Controller searches for bins ready for output (to the Gbit Ethernet chip). The LTA Controller exclusively searches for ready bins in the selected bank. This functionality allows the Gbit Ethernet chip to select banks and prioritize data transmission from multiple LTA Controllers to deal with high performance modes of operation (e.g. burst modes, and short integration times.) The LTA Controller responds with BSBR, a simple acknowledgement of the receipt of the command.

TXR – Transmit frame ready? This command allows the Gbit Ethernet chip to query the LTA Controller to see if there is a transmit frame ready in the output buffer for transmission to the Gbit Ethernet chip. D0 of the response data is set (1) if the buffer is ready and reset (0) if not ready.

TXREQ – Transmit Request. This command is issued to request transmission of a data packet from the LTA Controller to the Gbit Ethernet chip. The response (TXREQACK)

[2] When the MSR RM bit is set (1), this quantity must be multiplied by 2 by the GigE FPGA. This is because 10 bits (D0…D9) is not enough to represent 2000 bins. This will always give a value that is slightly larger than it really is, but still 0 when the fullness is actually 0 (the important thing).

indicates if transmission will start or not however, by the time the response is received, transmission on LTA_DATA (with LTA_FRAME_ asserted low) will already have commenced. This command can be sent even if TXR indicates that no frame is ready, resulting in a negative response, and no transmission on LTA_DATA.

5.4.2.2 LTA Output Data Frame

There can be two types of data frames transmitted by the LTA Controller to the Gbit Ethernet chip on the LTA_DATA[0:7] lines. The first is the **LTA data frame** and is shown in Figure 5-3—a description of the content of this frame follows. All of these frames have an "FType" of 100b, indicating that the frame is taken from the LTA RAM. The second data frame is a **raw Correlator Chip data frame**, and is defined in the Correlator Chip RFS document [1] with an "FType" or "Cmmd" of 011b, and shown in Figure 5-1. This is also known as a "speed dump" data frame, and is normally used for testing and high performance purposes only. Refer to the Correlator Chip RFS document for a description of the contents of this frame. Not shown in any of the figures is the destination MAC (6 bytes) and destination IP address (4 bytes) both of which are transmitted *before the beginning of the lag frame*, LSByte first. Only a 4 byte IP address is supported.

It is important to note that this frame, or the raw Correlator Chip data frame if a speed dump is present, transmitted on LTA_DATA[0:7] is transmitted in word succession, least-significant byte first.

Example: if the first few words that are to be output are:

fedcba986655 destination MAC (6 bytes)
76543210 destination IP address (4 bytes)
aaaaaaaa W0: START SYNC WORD
00000a84 W1: LTA FRAME Start_blky= 0 Nblocks= 0 nlags=000 chipIDY/X=2/5 CCC= 0 FType=100
006400c8 W2: BBIDY=0 SBIDY= 0 SIDY=100 BBIDX=0 SBIDX= 0 SIDX=200
c4020000 W3: STATUS=1100010 frame_count= 2 rblk-y= 0 rblk-x= 0
…

Then the actual transmit byte sequence will be:

55-66-98-ba-dc-fe-10-32-54-76-aa-aa-aa-aa-84-0a-00-00-c8-00-64-00-00-00-02-c4…

The transmit byte sequence in the Gbit Ethernet output UDP/IP data packet is modified from this to be compliant with IP standards (i.e. MSByte first, LSBit first).

LTA Output Data Frame

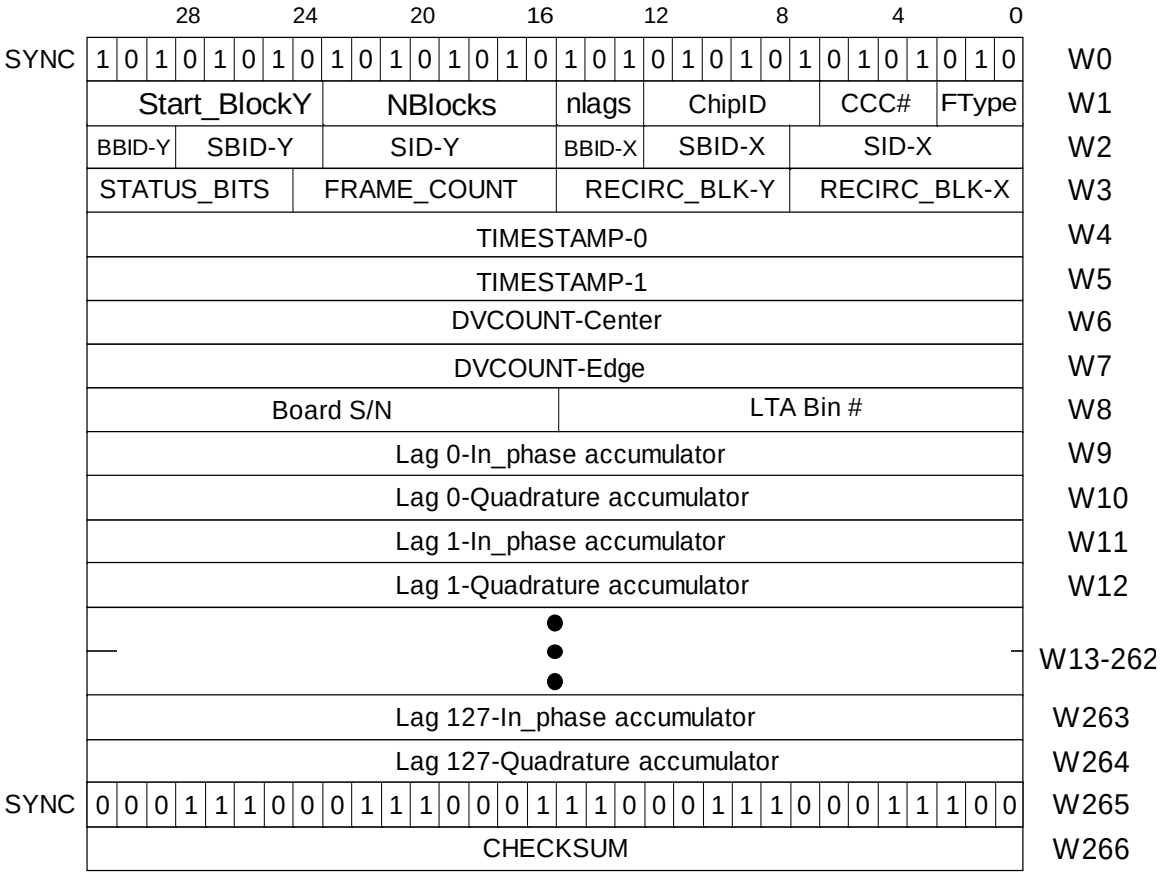


Figure 5-3 LTA output data frame. The "FType" is always 100b. If a 64-lag frame is produced, then the CHECKSUM is word 138 (W138).

Word 0 (W0): Start sync word

Word1 (W1):

- **FType** (Bits [2:0])– always 100b, indicating LTA output data frame
- **CCC#** (Bits[6:3]) – The 4-bit Correlator Chip cell number this frame originated from.
- **ChipID** (Bits[12:7]) – Bits 12, 11, 10 are the Y coordinates of the chip; Bits 9, 8, 7 are the X coordinates of the chip.
- **nlags** (Bits[15:13]) – Number of complex lag data points in this frame according to the following table:

Nlags B[15:13]	No. of complex lags
000	128
001	64
All others	error

Table 5-2 nlags bit encoding in the LTA Output Data Frame

- **NBlocks** (Bits[23:16]) – Number of recirculation blocks (-1) being acquired by the Correlator Chip this data frame came from. If recirculation is not active for this frame, then this is set to 0.

- **Start_BlockY** (Bits[31:24]) –The first Y recirculation block number acquired by the Correlator Chip this data frame came from. If recirculation is not active for this frame, then this is set to 0.

Word 2 (W2):

- **SID-X** (Bits[7:0]) – The 8-bit X station ID that identifies the X Station Board this data came from.

- **SBID-X** (Bits[12:8]) – The 5-bit X sub-band ID that identifies the X sub-band FIR filter this data came from.

- **BBID-X** (Bits[15:13]) – The 3-bit X baseband ID that identifies the baseband this data came from.

- **SID-Y** (Bits[23:16]) - The 8-bit Y station ID that identifies the Y Station Board this data came from.

- **SBID-Y** (Bits[28:24]) – The 5-bit Y sub-band ID that identifies the Y sub-band FIR filter this data came from.

- **BBID-Y** (Bits[31:29]) – The 3-bit Y baseband ID that identifies the baseband this data came from.

Word 3 (W3):

- **RECIRC_BLK-X** (Bits[7:0]) – The 8-bit X recirculation block number. If recirculation is not active for this data frame, then this is always 0.

- **RECIRC_BLK-Y** (Bits[7:0]) – The 8-bit Y recirculation block number. If recirculation is not active for this data frame, then this is always 0.

- **FRAME_COUNT** (Bits[24:16]) – A 9-bit frame counter indicating the number of raw Correlator Chip data frames that were added together in the LTA to produce this data frame. This counter is modulo 512, and is for information/error detection purposes only (i.e. it is not needed for any data normalization).

- **STATUS_BITS** (Bits[31:25]):
    - B31 – Unused, always 1.
    - B30 – Y-input switch setting for this cell: 0=new input; 1=adjacent cell.
    - B29 - X-input switch setting for this cell: 0=new input; 1=adjacent cell.
    - B28 – YSyncerr: one or more frames added to produce this frame reported a synchronization error on its Y input.
    - B27 - XSyncerr: one or more frames added to produce this frame reported a synchronization error on its X input.
    - B26 – ACC_OV: one or more frames added to produce this frame reported an accumulator overflow. Normally this means that the entire frame is bad.
    - B25 – OVR: one or more frames added to produce this frame reported an overrun condition. This does not mean that this frame is bad, only that the Correlator Chip had to drop one or more frames within the integration time of this frame.

Word 4 (W4): TIMESTAMP-0: The 32-bit count of number of seconds (1 PPS ticks) since the TIMECODE epoch.

Word 5 (W5): TIMESTAMP-1: Bits[28:0] is the count of the number of 128 MHz clock cycles since the last 1 PPS. In some cases this can be >128,000,000 but it, and TIMESTAMP-0, always produce a correct timestamp. B[31:29] is the epoch extracted from the system TIMECODE.

Word 6,7 (W6, W7): DVCOUNT-Center, DVCOUNT-Edge: Data valid counts at the center of the cell (lag N/2), and at the edge of the cell (lag 0) respectively. This is a count of the number of valid samples added in this data frame.

Word 8 (W8):

- **LTA_Bin#** (Bits[15:0]): This is the bin number this data came from in the LTA where the MSBit (bit 15) is *always* the bank number This may or may not be the actual "phase bin" of this data, depending on whether recirculation is active or not according to the following equation:

  LTA_bin# = Phase_bin x (Nblocks+1) + (RECIRC_BLK_Y- Start_BlockY)

- **Board S/N** (Bits[31:16]): This is a 16-bit board serial number that is taken from a configuration register on the chip. Nominally, this identifies the physical location of the board (i.e. rack-crate-slot), according to a system-wide numbering convention.

Word 9-264 (W9-264): 128 complex-lag data, with bias (i.e. “DATA_BIAS” from the Correlator Chip data frame) removed. If nlags=64, then this only occupies words 9-136.

Word 265 (W265): End sync word. If nlags=64, then this is word 137.

Word 266 (W266): Frame checksum. This is the 32-bit sum of all words in the frame (W0-W265) and does not include the IP address header that is not part of the frame. If nlags=64, then this is the sum of words W0-W137.

5.4.3 MCB (Microprocessor) Interface Requirements

The MCB (Monitor & Control Bus) interface allows a microprocessor to write into the LTA Controller to configure it, and to read from it to verify configuration information and obtain status information. Physical interface timing requirements are shown in Figure 5-4.

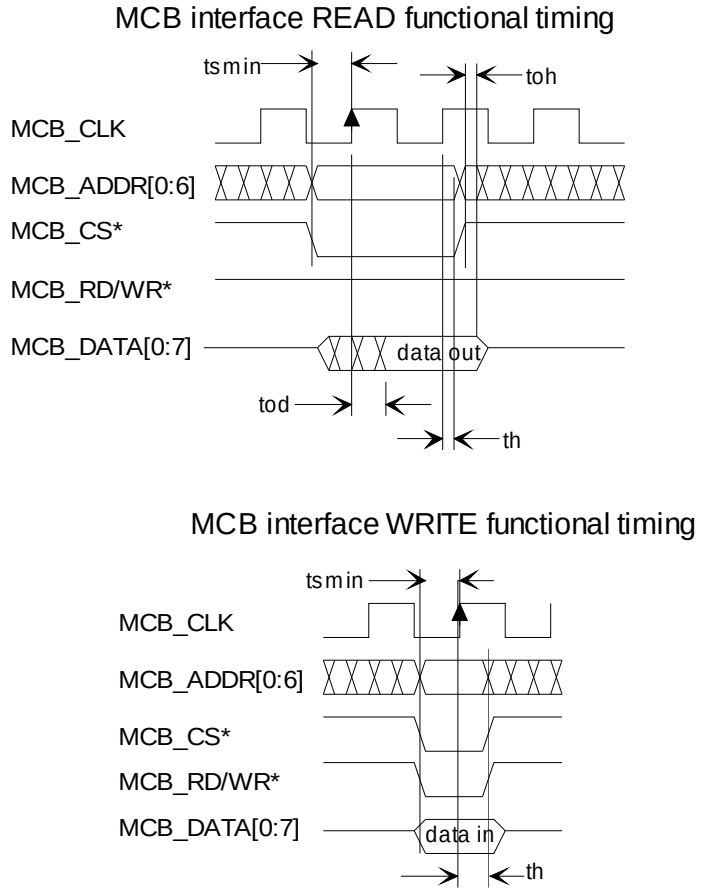


Figure 5-4 MCB interface functional timing requirements. READs require one clock cycle setup before data is ready—however, successive reads take one clock cycle each. WRITEs require one clock cycle. tod max is 11 nsec.

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

5.4.3.1 Master Control/Status Register (MCSR); R/W Address=0x00

This register set global chip configuration, and detects global status information. A complete description of the register bits is as follows:

MCSR (R/W) Addr=0x00 (Reset=0x00)

7	6	5	4	3	2	1	0
FD	ErCon	IPv	RAMsize1	RAMsize0	BS	BM	TN

TN (R/W) – Test or Normal operation. If reset (0), the LTA Controller is in **test mode** and all normal lag frame handling is suspended. In test mode, the microprocessor can read and write the LTA RAM and the Semaphore Table RAM via the LTA RAM address and data registers. This allows memory tests, setting up the LTA RAM and Semaphore Table for testing, reading the contents of the LTA etc. If set (1), the LTA Controller is operating in normal mode and the microprocessor does not have access to the LTA RAM and Semaphore Table. Test mode can be enabled or disabled without disturbing the LTA RAM—the most that will happen is that frames from the Correlator Chip will be lost (i.e. Correlator Chip overruns). A 1→0 transition of this bit will not be instantaneous—the microprocessor must read the register several times to detect when it goes low. This action is necessary to ensure that the current RAM access in progress is not interrupted, and can take several (up to 10) μsec to occur.

BM (R/W) – Burst mode operation. If reset (0), the LTA Controller is in normal mode with the LTA Integration Controller and the LTA Frame Output Controller contending for LTA RAM for accumulation and transmission purposes. In normal mode, no special action is taken by either controller (e.g. if the LTA RAM is full, there is no accumulation hold-off or anything like that). If set (1), the LTA Controller is in burst mode, and unblocked access to LTA RAM is provided to one or the other of these controllers depending on the BS bit. This bit may be changed by the microprocessor at any time.

BS (R/W) – Burst select bit. This bit is only useful if the BM bit is set. If the BS bit is reset (0) and BM=1, the LTA Integration Controller has exclusive access to LTA RAM and no frames will be transmitted to the Gbit Ethernet chip. If the BS bit is set (1) and BM=1, the LTA Frame Output Controller has exclusive access to LTA RAM, and no data frames are accepted from the Correlator Chip. This bit may be changed by the microprocessor at any time.

RAMsize[1:0] (R) – These bits indicate the size of the LTA RAM according to the following table:

RAMsize[1:0]	Total No. of bins	Memory used
00	2000	256 Mbit
01	4000	512 Mbit
10	8000	1 Gbit
11	16000	2 Gbit

IPv (R/W) – This bit sets IP version 4 or IP version 6. If reset (0), IP v4 is used and the destination IP address transmitted to the Gbit Ethernet chip is 32 bits. If set (1), IP V6 is used and the destination IP address transmitted to the Gbit Ethernet chip is 64 bits. **Currently only IP v4 is supported, and this bit will therefore always read 0, no matter what its write value.**

ErCon (R/W) – Error level control for frame rejection. If this bit is reset (0), then frames are rejected from the Correlator Chip only if unrecoverable errors are detected. If set (1) then frames are rejected if errors in the frame's STATUS BITs are detected. Refer to section 5.4.3.5 for a list of frame reject conditions. Normally, this bit should be set (1) to ensure no suspect data is saved into the LTA (if it is, though, the STATUS bits will reflect what occurred).

FD (R) – Frame detect bit. If set (1), then one or more frames have been transferred from the Correlator Chip to the LTA Controller. If reset (0), then no frames have been transferred since the last time this register was read.

5.4.3.2 CCC Statistics Control Register (C3SCR); R/W Address=0x01

This register is used to set the CCC that statistics are being captured for, and capture critical statistics information from the frame header from the Correlator Chip.

C3SCR (R/W) Addr=0x01 (Reset=0x00)

7	6	5	4	3	2	1	0
YSynerr	XSynerr	OV	OVR	Cs3	Cs2	Cs1	Cs0

Cs[3:0] (R/W) – These bits select the Correlator Chip cell (CCC) that lag frame statistics will be performed on for the registers in following sections.

OVR (R) – Set (1) if the OVR (overrun) bit is set in a received data frame from the Correlator Chip, for the CCC defined by Cs[3:0]. Cleared on read.

OV (R) – Set (1) if the OV (overflow) bit is set in a received data frame from the Correlator Chip for the CCC defined by Cs[3:0]. Cleared on read.

XSyncerr (R) – Set (1) if the XSynerr (input sync error) bit is set in a received data frame from the Correlator Chip for the CCC defined by Cs[3:0]. Cleared on read.

YSyncerr (R) – Set (1) if the YSynerr (input sync error) bit is set in a received data frame from the Correlator Chip for the CCC defined by Cs[3:0]. Cleared on read.

5.4.3.3 Miscl. Status Register (MSR); R/W Address=0x02

This register is used to obtain miscellaneous status information, defines the build version of the FPGA, and controls LTA RAM clearing after readout.

MSR (R/W) Addr=0x02 (Reset=0x50)

7	6	5	4	3	2	1	0
V1	V0	PS	RM	CRE	RIOK	CERR	CD

CD (R) – Command Detect. If set (1), this bit indicates that a command was successfully detected on the LTA_SER_CMD input line, meaning that the Gbit Ethernet chip has been talking to this chip. Cleared on read.

CERR (R) – Command Error Detect. If set (1), this bit indicates that there has been some activity on the LTA_SER_CMD input line, but that one or more commands were discarded because the receiver detected an incorrect protocol (e.g. unrecognized command, missing or incorrect stop bits etc.). Cleared on read.

RIOK (R) – RAM Init OK. If set (1) it indicates that the semaphore table has been correctly initialized, and that the external DDR SDRAM (LTA RAM) has been correctly initialized, and passed memory tests (MTC=1, MTF=0). If reset (0) it indicates that there was an initialization error and that no accumulator operations will be performed (the chip won't accept data frames from the Correlator Chip, nor will it generate data frames to the Gbit Ethernet chip). **This bit will normally be reset for about 32 seconds after the PU bit of the CCPCR goes high**.

CRE (R/W) – Clear RAM control bit. If set (1), the LTA Controller will write zeros into the LTA RAM after a finished data frame has been extracted from RAM. If reset (0), the LTA Controller will not write zeros to RAM. Normally this bit is set (1) so that even if the "initial dump and save" frame from the Correlator Chip is missed for some reason, it will result in good data being saved into the LTA, albeit with a frame_count that is less than expected. If this bit is reset (0), it could result in frame_counts that are more than expected (or worse, wrap-around past zero), and with data from more than one integration time. This bit should be reset (0) only if the Correlator Chip and LTA are operated in the most demanding dump applications such as burst dumping with single-CCC integration times < 10 μsec, or in pulsar binning applications with similar single-dump accumulation performance.

RM (R/W) – LTA RAM Mode. If reset (0), then the LTA RAM is 2 banks of 1000 bins each, and the PS bit selects which RAM partition is used. This allows for self-healing capability if one partition of the RAM happens to have a RAM test failure. If a RAM test failure occurs, then the alternate partition should be selected in an effort to workaround

the bad RAM partition. If set (1), then the LTA RAM is 2 banks of 2000 bins, and the PS bit is ignored since there is only 1 partition. Default=1.

PS (R/W) – Partition Select. This bit is only used if the RM bit is reset (0). If reset (0), then the lower RAM partition is selected for use. If set (1), then the upper RAM partition is selected for use.

V1, V0 (R) – FPGA major version number. If this is V1=0 and V0=0, then it is the original LTA version built for the original/old GigE FPGA design. If this is V1=0 and V0=1, this it is the LTA version built to support the V2 GigE FPGA design.

5.4.3.4 Correlator Chip Power Control Register (CCPCR); (R/W) Addr=0x03

This is the register which controls the Accel regulator chip for the Correlator Chip, and the resets to the Correlator Chip.

CCPCR (R/W) Addr=0x03 (Reset=0x00)

7	6	5	4	3	2	1	0
MTF	MTC	PLL 128	PLL 133	CPR	CR	PGF	PU

PU (R/W) – Power Up. This bit, when set, tells the Accel regulator chip to power up. When this bit is set (1) the Accel chip is active. When it is reset (0), the Accel chip is not active and the Correlator Chip is effectively powered down. **An additional side effect is that setting this bit (0-to-1 transition) starts the self-test of the external DDR SDRAM; resetting this bit resets the DDR SDRAM interface.**

PGF (R) – Power Good Flag. This bit, when set (1), indicates that the Accel regulator is powered up and active. This means that the Correlator Chip is operational.

CR (R/W) – Correlator Chip reset. This bit, when reset (0) holds the Correlator Chip reset line low. When in reset, the Correlator Chip PLLs are still active and data is still being passed onto the next chip in the daisy chain, however all internal logic in the chip is inactive. Thus, a Correlator Chip can be reset, without affecting other chips on the board. When set (1), the Correlator Chip reset line is brought high, and the chip is active.

CPR (R/W) – Correlator Chip PLL reset. This bit, when reset (0) holds the Correlator Chip PLLs in the reset/power down state. When set (1), the PLLs become active. Typically, the CR bit should be held low for at least 0.5 milliseconds after this line (CPR) is brought high.

This register is an important control register for activating the Correlator Chips on the board. To prevent power transients and ensure correct operation, this register should be used to power up, and then activate Correlator Chips in the following fashion:

- One-by-one, assert the PU bit in every LTA Controller on the board. To prevent power spikes, there should be a few milliseconds between each chip activation.
- Once all of the PGF bits in all of the LTA Controllers are set—indicating Correlator Chips are powered up—the CPR bits get set starting with row0, column0 and up columns and across rows [5]: before a Correlator Chip CPR bit is set, it must be guaranteed that the X and Y clocks into the chip are active.

- Once all PLLs are active (CPR bits in all LTA Controllers set for at least 0.5 milliseconds), then the CR bits are brought high one-by-one, spaced apart in time by a few milliseconds each.

PLL133 (R) – This read bit indicates the status of the 133 MHz PLL. If set (1), the PLL is locked; if reset (0) the PLL is not locked. The 133 MHz clock reference (via a divide by 4=33.25 MHz) is provided to the LTA by the Gigabit Ethernet FPGA.

PLL128 (R) -- This read bit indicates the status of the 128 MHz PLL. If set (1), the PLL is locked; if reset (0) the PLL is not locked. The 128 MHz clock is provided to the LTA by the associated Correlator Chip.

MTC (R) – Memory Test Complete. This bit goes high (1) when the LTA FPGA has completed the built-in memory test of its DDR SDRAM. This test ensures all bit locations are ok, and clears the memory (sets all locations to 0) before this bit goes high. The success of the memory test is determined by the state of the MTF bit, and the RIOK bit of the MSR. The **memory test takes about 32 seconds** after the PU bit goes high to complete, and this bit allows the CPU to determine when normal processing can start if the memory test has passed.

MTF (R) – Memory Test Fail. This bit remains high (1) until the built-in memory test is complete (MTC=1), and successful. Thus, once the MTC bit goes high, this bit must be low for normal processing to complete**. If MTC=1 and MTF=1, then the memory test has failed.**

If the memory test fails, the LTARADR0-3 registers contain the address where the failure occurred, and the LTARDATA0-3 registers contain the failed read data value. Valid data patterns used in the test are: 0xaaaaaaaa, 0x55555555, 0xf0f0f0f0, 0x0f0f0f0f, 0x591f3da6, and 0x00000000. Any data patterns other than these that are deposited in the LTARDATA0-3 register indicate a stuck memory location, a stuck/shorted data line, or a stuck/shorted address line. In the event of failure, further CPU-generated/controlled memory tests should be performed via the LTARADR0-3, LTARDATA0-3, and LTARRWR registers to determine the exact cause of failure.

NOTE: Until MTC=1 and MTF=0, the LTA will not be able to be proceed with normal processing, as the FPGA prevents any attempt to pull TN bit of the MCSR high. This restriction ensures that the LTA won't process any frames if the memory is not good, thereby preventing possible and undetectable LTA memory and output frame corruption.

NOTE: If the memory test fails, internally the chip continues to run the test to allow probing of signals on the RAM chip to assist in determining the fault. Thus, no software RAM test after a memory test fail is possible.

5.4.3.5 CCC Frame Reject Count Register (C3FREJCR); (R) Addr=0x08

This is an 8-bit register that counts the number of frames rejected from the Correlator Chip for the CCC selected in the C3SCR register. This register will saturate (i.e. stop counting) at 0xFF. Cleared on read. Frames from the Correlator Chip will *always* be rejected (i.e. these are non-maskable errors) if at least one of the following conditions occurs:

- Frame too long or frame too short detected.
- Inconsistent recirculation block numbers in the lag frame.
- LTA bin the frame is destined for is full (i.e. waiting to be read out).
- Checksum error in the frame.
- Lag frame start/end sync word error.
- Phase bin, or the resulting data bin number calculated for recirculation is out of range.
- STP bit is set in the C3FCSR register.

Frames *may* be rejected, depending on the ErCon bit setting in the MCSR register. If the ErCon bit is set (1), frames will additionally be rejected because of:

- XSyncerr, or YSyncerr detected in the STATUS BITs of the frame.
- OV (overflow) detected in the STATUS BITs.
- OVR (overrun condition) if recirculation is active for a particular CCC (determined by the RE bit of the C3FCSR register). In this case, the frame is discarded because it contains correlated data from the last time burst due to the fact that the Correlator Chip ignores the "holdoff" signal under overrun conditions.

A frame is usually never rejected if only the lag frame OVR STATUS BIT is set, except as indicated above. Additionally, speed-dump frames are only ever reject if the frame is too long or too short, not for any other reason.

IMPORTANT NOTE: The LTA *silently* aborts/rejects all correlator chip data frames with a SID-X or SID-Y of 0. Zero SIDs are set by the Recirc FPGA to indicate the data channel is not synchronized, and the LTA discards frames with zero SIDs to prevent garbage frames from making their way to the output. The C3FREJCR register does not count these aborted frames.

C3FREJCR (R) Addr=0x08 (Reset=0x00)

7	6	5	4	3	2	1	0
c7	c6	c5	c4	c3	c2	c1	c0

c[7:0] (R) – Count of number of frames rejected for the selected CCC since this register was last read. Cleared on read.

5.4.3.6 CCC Correlator Chip Frame Count Register (C3CCFCR); (R) Addr=0x09

This is an 8-bit register that counts the number of lag frames detected from the Correlator Chip for the CCC selected in the C3SCR register since the last time it was read. This is a count of all frames detected, even errored or dropped frames. This register will saturate (i.e. stop counting) at 0xFF.

C3CCFCR (R) Addr=0x09 (Reset=0x00)

7	6	5	4	3	2	1	0
c7	c6	c5	c4	c3	c2	c1	c0

c[7:0] (R) – Count of the number of lag frames detected since last read. Cleared on read.

5.4.3.7 CCC LTA RAM Frame Count Register (C3LTAFCR); (R) Addr=0x0A

This is an 8-bit register that counts the number of lag frames actually accumulated into the LTA for the CCC selected in the C3SCR register since the last time it was read. This register will saturate (i.e. stop counting) at 0xFF.

C3LTAFCR (R) Addr=0x0A (Reset=0x00)

7	6	5	4	3	2	1	0
c7	c6	c5	c4	c3	c2	c1	c0

c[7:0] (R) – Count of the number of lag frames accumulated into the LTA since last read. Cleared on read.

5.4.3.8 CCC Output Frame Count Register (C3OFCR); (R) Addr=0x0B

This is an 8-bit register that counts the number of LTA lag frames transmitted to the Gbit Ethernet chip for the CCC selected in the C3SCR register since the last time it was read. This register will saturate (i.e. stop counting) at 0xFF.

C3OFCR (R) Addr=0x0B (Reset=0x00)

7	6	5	4	3	2	1	0
c7	c6	c5	c4	c3	c2	c1	c0

c[7:0] (R) – Count of the number of LTA lag frames transmitted to the Gbit Ethernet chip since last read. Cleared on read.

5.4.3.9 CCC Speed Frame Count Register (C3SFCR); (R) Addr=0x0C

This is an 8-bit register that counts the number of "speed-dump" lag frames transmitted to the Gbit Ethernet chip for the CCC selected in the C3SCR register since the last time it was read. This register will saturate (i.e. stop counting) at 0xFF.

C3SFCR (R) Addr=0x0C (Reset=0x00)

7	6	5	4	3	2	1	0
c7	c6	c5	c4	c3	c2	c1	c0

c[7:0] (R) – Count of the number of speed-dump lag frames transmitted to the Gbit Ethernet chip since last read. Cleared on read.

5.4.3.10 CCC Speed Frame Dropped Count Register (C3SFDCR); (R) Addr=0x0D

This is an 8-bit register that counts the number of speed-dump lag frames dropped for the CCC selected in the C3SCR register since the last time it was read. Speed-dump frames will be dropped if an output buffer is full, and so there is no place to put the frame. This does not count the number of speed-dump frames discarded because of errors. This register will saturate (i.e. stop counting) at 0xFF.

C3SFDCR (R) Addr=0x0D (Reset=0x00)

7	6	5	4	3	2	1	0
c7	c6	c5	c4	c3	c2	c1	c0

c[7:0] (R) – Count of the number of speed-dump lag frames dropped since last read. Cleared on read.

5.4.3.11 CCC Frame Control/Status Registers (C3FCSR-0, C3FCSR-1, … C3FCSR-15); R/W Address 0x10…0x1F

There are 16 of these registers, and each register provides critical control and status information for lag frames coming from a particular Correlator Chip CCC. C3FCSR-0 is the control/status register for CCC 0 lag frames, C3FCSR-1 is the control/status register for CCC 1 lag frames etc.

C3FCSR-(0…15) (R/W) Addr=0x10…0x1F (Reset=0x00)

7	6	5	4	3	2	1	0
FSE	RBE	CE	SLE	LTAF	Nlags	STP	RE

RE (R/W) – Recirculation Enable. If reset (0), then the associated CCC *is not* performing recirculation. If set (1), then the associated CCC *is* performing recirculation. The setting of this bit affects whether consistency of recirculation block numbers is performed, and affects the header content of the output data frame.

STP (R/W) – If set (1), lag frames for this CCC are blocked (i.e. dropped) from the Correlator Chip. If reset (0), lag frames for this CCC are accepted. This bit does not affect the output of LTA lag frames to the Gbit Ethernet chip.

Nlags (R/W) – Number of lags output from the LTA. If reset (0), 128 lags are output. If set (1), 64 lags are output. This bit does not affect the output of speed frames.

LTAF (R) – LTA full status bit. If set (1), one or more lag frames from the Correlator Chip had to be dropped because the LTA bin they were destined for is full, or the phase bin it is destined for is out of range. If reset (0), no lag frames were rejected for this reason. Cleared on read.

SLE (R) – Frames too short or too long detected. If reset (0), no short/long frames have been detected from the Correlator Chip. If set (1), one or more frames that are too long or too short have been detected.

CE (R) – Checksum error bit. If reset (0), there have been no checksum errors detected in any associated CCC lag frames since the last time this register was read. If set (1), there have been checksum errors detected in lag frames since last read. Cleared on read.

RBE (R) – Recirculation Block Error. If reset (0), all detected recirculation block numbers have been consistent in any associated CCC lag frames since the last time this was read. If set (1), CCC lag frames with inconsistent X and Y recirculation block numbers have been detected –OR—a frame with an OVR (overrun) has been detected indicating that it contains correlated data from the previous time burst. In both cases, the offending frames have been discarded. Cleared on read.

FSE (R) – Frame sync error. If reset (0), no lag frame sync word errors have been detected. If set (1), lag frame sync errors have been detected. Cleared on read.

5.4.3.12 Start_BlockY Configuration Register (SBYCR); (R/W) Addr=0x20

This 8-bit register sets the "Start_BlockY" for any CCCs that are performing recirculation. This is the lowest RECIRC_BLKY number that can be expected from any lag frame from the Correlator Chip this LTA is connected to. For example, if the total number of recirculation blocks acquired for a particular set of lags (i.e. lags that go into a Fourier Transform) is 16, but this particular LTA/Correlator Chip duo only acquires 4 of them, starting at block 4, then this register is set to 4.

SBYCR (R/W) Addr=0x20 (Reset=0x00)

7	6	5	4	3	2	1	0
b7	b6	b5	b4	b3	b2	b1	b0

b[7:0] (R/W) – The start block Y number.

5.4.3.13 NBlocks Configuration Register (NBCR); (R/W) Addr=0x21

This 8-bit register is used to set the NBlocks parameter—the number of recirculation blocks **minus 1** acquired by this LTA/Correlator Chip duo—for any CCCs that are performing recirculation (refer to section 6.2 for use). This number is required by the LTA to determine how to use lag frame storage bins in the LTA RAM. For example, if the total number of recirculation blocks acquired for a particular set of lags (i.e. lags that go into a Fourier Transform) is 64 (TBCR=63) , but this particular LTA/Correlator Chip duo only acquires 16 of them, then this number is set to 15.

NBCR (R/W) Addr=0x21 (Reset=0x00)

7	6	5	4	3	2	1	0
b7	b6	b5	b4	b3	b2	b1	b0

b[7:0] (R/W) – The NBlocks number.

5.4.3.14 TotBlocks Configuration Register (TBCR); (R/W) Addr=0x22

This 8-bit register is used to set the TotBlocks **minus 1** parameter for any CCCs that are performing recirculation (refer to section 6.2 for use). This is the TOTAL number of recirculation blocks acquired by any LTA/Correlator Chip duo that produces lag frames

that are part of the same Fourier Transform. This number is purely used to allow the LTA to consistency check the RECIRC_BLK-Y and RECIRC_BLK-X values coming from the Correlator Chip. If RECIRC_BLK-Y + RECIRC_BLK-X ≠ TotBlocks, and the particular CCC is set for recirculation (RE bit of the C3FCSR-*ccc#* register), then the LTA dumps/discards the Correlator Chip data frame, and flags a recirculation block error (RBE bit of the C3FCSR-*ccc#* register goes high). This ensures that bad frames, which might occur because Recirc FPGA recirculation counters have lost sync, are not integrated into the LTA. For example, if this particular LTA is acquiring 64 blocks (NBCR=63), but there are 256 blocks in total being acquired, then this number is set to 255.

TBCR (R/W) Addr=0x22 (Reset=0x00)

7	6	5	4	3	2	1	0
b7	b6	b5	b4	b3	b2	b1	b0

b[7:0] (R/W) – The TotBlocks number.

5.4.3.15 ChipID Configuration Register (CHIDCR); (R/W) Addr=0x23

This 6-bit register is used to set the "ChipID" that is transmitted as part of the LTA lag data frame to the Gbit Ethernet chip. This ChipID sets the X and Y location of the associated Correlator Chip on the Baseline Board.

CHIDCR (R/W) Addr=0x23 (Reset=0x00)

7	6	5	4	3	2	1	0
N/C	N/C	Yc2	Yc1	Yc0	Xc2	Xc1	Xc0

Xc[2:0] (R/W) – The X-coordinate of the associated Correlator Chip.

Yc[2:0] (R/W) – The Y-coordinate of the associated Correlator Chip.

N/C – Unused bits.

5.4.3.16 Board ID Configuration Registers (BIDCR-0, BIDCR-1); (R/W) Addr=0x24, -0x25

These two 8-bit registers are used to set the 16-bit Board ID that is transmitted as part of the LTA lag data frame (Figure 5-3). This is used to uniquely identify the source of this data in the system.

BIDCR-0 (R/W) Addr=0x24 (Reset=0x00)

7	6	5	4	3	2	1	0
bid7	bid6	bid5	bid4	bid3	bid2	bid1	bid0

bid[7:0] (R/W) – Bits 0…7 of the 16-bit Board ID.

BIDCR-1 (R/W) Addr=0x25 (Reset=0x00)

7	6	5	4	3	2	1	0
bid15	bid14	bid13	bid12	bid11	bid10	bid9	bid8

bid[15:8] (R/W) – Bits 8…15 of the 16-bit Board ID.

5.4.3.17 Destination IP Address Registers (DIPADR-0, DIPADR-1, … DIPADR-7); (R/W) Addr=0x30…0x37

These eight, 8-bit registers set the destination IP address that is sent to the Gbit Ethernet chip at the beginning of a frame transmission. Depending on the setting of the **IPv** bit in the **MCSR**, this can be a 32-bit or a 64-bit address. NOTE: ONLY A 4-BYTE IP ADDRESS (FIRST 4 REGISTERS) ARE CURRENTLY SUPPORTED. The DIPADR-0 register contains bits 0…7 of the address, the DIPADR-1 register contains bits 8…15 of the address etc. If a 32-bit address is required, only the first four registers are used.
Reset=0x00

5.4.3.18 LTA RAM Address Registers (LTARADR-0, LTARADR-1, LTARADR-2, LTARADR-3); (R/W) Addr=0x38…0x3B

These 8-bit registers select the (up to) 26-bit address that is used for the following LTA RAM reads or writes. LTARADR-0 is the LSByte of the LTA RAM address, and LTARADR-3 is the MSByte of the LTA RAM address. The number of address bits that must be used depends on the LTA RAM size that is being used (see the RAMsize bits in the MCSR register), according to the following table.

RAMsize[1:0]	Total Number of bins	Memory used	No. of RAM addr bits used
00	2000	256 Mbit (8Mx32)	23
01	4000	512 Mbit (16Mx32)	24
10	8000	1 Gbit (32Mx32)	25
11	16000	2 Gbit (64Mx32)	26

Table 5-3 LTA RAM Address Register bits used as a function of the size of the LTA RAM.

It is important to note that read/write access to the LTA RAM is word (32-bit) aligned, and so the address bits in the LTA RAM address registers do not include the lowest 2 address bits that would always be zero anyway.

The actual content of the LTA RAM memory (i.e. the memory map) is described in section 6.2. **All registers are 0 after reset.**

In the event of a built-in DDR SDRAM memory test failure, these registers contain the first location where the failure occurred (refer to the MTC and MTF bits of the CCPCR register). Writing anything to these registers overwrites this value.

The full extent of the memory that is accessed depends on the RM bit of the MSR register. If the memory is set for 2 banks of 2000 bins, then all memory locations are accessible. If the memory is set for 2 banks of 1000 bins, then only each memory partition is accessible, depending on the state of the PS bit of the MSR.

5.4.3.19 LTA RAM Data Registers (LTARDATA-0, LTARDATA-1, LTARDATA-2, LTARDATA-3); (R/W) Addr=0x3C…0x3F

These registers contain the data that is to be written to the LTA RAM or read from the LTA RAM, depending on whether the LTA RAM Read/Write register (0x38) is written to or read from. LTARDATA-0 is the LSByte of this operation and LTARDATA-3 is the MSByte of this operation. All registers are 0 after reset.

In the event of a built-in DDR SDRAM memory test failure, these registers contain the read data value where the failure occurred (refer to the MTC and MTF bits of the CCPCR register). Writing anything to these registers, or reading from the LTARRWR register, overwrites this value.

5.4.3.20 LTA RAM Read/Write Register (LTARRWR); (R/W) Addr=0x40

If the LTA Controller is in test mode (the TN bit in the MCSR is reset (0)), then writing (a dummy value) to this register will cause the contents of the LTARDATA-0…3 registers to be written to the LTA RAM address location defined by the LTARADR-0…3 registers. Conversely, under the same conditions, reading from this register will cause a read of LTA RAM to be deposited into the LTARDATA-0…3 registers. If the TN bit in the MCSR is set, this functionality is disabled, and no LTA RAM reads or writes are actually executed.

5.4.3.21 LTA RAM Semaphore Table Address Registers (LTARSTADR-0, LTARSTADR-1); (R/W) Addr=0x48, 0x49

These two 8-bit registers are used to set the address to which a write, or from which a read will be performed on the LTA RAM Semaphore Table, only if the TN bit in the MCSR is reset (0). The LTARSTADR-0 is the LSByte, and LTARSTADR-1 is the MSByte of the address. The address is the bin number of the selected CCC (refer to the LTARSTCR) that will be accessed. The number of address bits actually used depends on the number of bins in the LTA RAM. For the baseline design of 2000 bins, only addresses from 0 to 999 (bank 0) and from 32768 to 33767 (bank 1) are useful (i.e. the MSBit of this register is the bank bit). All registers are 0 after reset.

The full extent of the semaphore table that is accessed depends on the RM bit of the MSR register. If the memory is set for 2 banks of 2000 bins, then all semaphore locations are accessible (and the ranges are 0-1999 and 32768-34767). If the memory is set for 2 banks of 1000 bins, then only each semaphore table partition is accessible, depending on the state of the PS bit of the MSR.

5.4.3.22 LTA RAM Semaphore Table Control Register (LTARSTCR); (R/W) Addr=0x4A

This register is used to set the CCC that a semaphore table read or write will be performed on, and to set the read/write data bit. This register is only useful if the TN bit in the MCSR is reset (0).

LTARSTCR (R/W) Addr=0x4A (Reset=0x00)

7	6	5	4	3	2	1	0
SEMI	N/C	Rbit	Wbit	Cs3	Cs2	Cs1	Cs0

Cs[3:0] (R/W) – The CCC that is selected for the semaphore table access. Conceptually, each CCC has its own semaphore table.

Wbit (R/W) – The bit that is to be written into the semaphore table for the selected CCC at the address of the LTARSTADR-0,1 registers.

Rbit (R) – The bit read from the semaphore table for the selected CCC at the address of the LTARSTADR-0,1 registers.

N/C – Unused bit.

SEMI (R/W) – This is the "semaphore clear inhibit" bit. If reset (0), then normal semaphore table clearing occurs when data frames are transferred from the RAM to the output. If set (1), then it inhibits semaphore table bit clearing **in all operating modes**. This bit is normally only set (1) for testing, to allow the LTA to continuously generate data frames to the Gigabit Ethernet FPGA.

5.4.3.23 LTA RAM Semaphore Table Read/Write Register (LTARSTRW); (R/W) Addr=0x4B

Writing a dummy value to this location causes a write to the semaphore table (i.e. causes the Wbit of the LTARSTCR to be written). Reading from this location causes a read of the semaphore table to be performed, with the read value deposited in the Rbit of the LTARSTCR register. This action is only taken if the TN bit of the MCSR is reset (0).

5.4.3.24 LTA Depth Read Registers (LTADRR-0 and LTADRR-1); (R/W) Addr=0x50, 0x51

These two 8-bit registers are used to determine the depth or "fullness" of the LTA. This value is the same depth that is output on the LTA_ SER_RESP line in response to an FIQ (fullness inquiry) command. Note that the depth of the LTA is always being calculated internally, these registers simply allow access to that information.

LTADRR-0 (R) Addr=0x50 (Reset=0x00)

7	6	5	4	3	2	1	0
DEP7	DEP6	DEP5	DEP4	DEP3	DEP2	DEP1	DEP0

LTADRR-1 (R/W) Addr=0x51 (Reset=0x00)

7	6	5	4	3	2	1	0
N/C	N/C	N/C	N/C	BNK	DEP10	DEP9	DEP8

DEP[10:0] (R) – The depth of the selected bank (BNK).

BNK (R/W) – The bank selected for the depth measure.

5.4.3.25 LTA Interface Handshake Current Status Register (LIHCSR); (R) Addr=0x60

This read-only register provides current level information for the Correlator Chip – LTA FPGA interface. This information is useful for testing to determine the current state of handshaking between the LTA FPGA and the Correlator Chip.

Bit 7 – FRAME_ABORT_ – The current status/level of the FRAME_ABORT_ line.

Bit 6 – DATA_VALID_ – The current status/level of the DATA_VALID_ line.

Bit 5 – DATA_OE_ – The current status/level of the DATA_OE_ line.

Bit 4 – DATA_RDY_ – The current status/level of the DATA_RDY_ line.

Bit[3:0] – The current state of the Correlator Chip "frame getter" controller.

5.4.3.26 LTA Interface HandshakeToggle Status Register (LIHTSR); (R) Addr=0x61

This read-only register provides current toggle status information for the Correlator Chip – LTA FPGA interface. This information is useful for testing to determine whether the handshaking lines are toggling or not. Cleared on read.

Bit 7 – FRAME_ABORT_ – Toggle status of the FRAME_ABORT_ line. If set (1), the line has toggled since the last time it was read. If reset (0), it has not toggled.

Bit 6 – DATA_VALID_ – Toggle status of the DATA_VALID_ line. If set (1), the line has toggled since the last time it was read. If reset (0), it has not toggled.

Bit 5 – DATA_OE_ – Toggle status of the DATA_OE_ line. If set (1), the line has toggled since the last time it was read. If reset (0), it has not toggled.

Bit 4 – DATA_RDY_ – Toggle status of the DATA_RDY_ line. If set (1), the line has toggled since the last time it was read. If reset (0), it has not toggled.

Bit[3:0] – Toggle status of the Correlator Chip "frame getter" controller.

5.4.3.27 Correlator Chip CCC Frame Detect Status Registers (CCCFDSR-0 and CCCFDSR-1); (R) Addr=0x62, 0x63

Each bit of each of these 8-bit registers is set if a frame from the corresponding CCC (Correlator Chip Cell) of the attached Correlator Chip has been detected since the last time it was read. If a bit is set (1), then one or more frames have been detected; if a bit is reset (0) then no frames have been detected for the particular CCC since last time the register was read. These registers are (independently) cleared on read. If the V1, V0 bits of the MSR are both 0, these registers do not exist.

Bit assignments are as follows:

CCCFDSR-0 [7…0] = CCC7…CCC0.

CCCFDSR-1 [7…0] = CCC15 … CCC8.

5.4.3.28 LTA Interface DATA Toggle Status Registers (LIDTSR-0, LIDTSR-1, LIDTSR-2, LIDTSR-3) (R); Addr=0x70, 0x71, 0x72, 0x73

These 4 registers indicate the toggle status of the 32 "ACCDATA" lines coming from the Correlator Chip. If the LTA FPGA is reporting checksum errors, these status bits can narrow-down which lines are failing. If read values are set (1), then the line(s) were toggling since the last time they were read. If reset (0), then the line(s) were not toggling since the last time they were read.

LIDTSR-0 [7:0] – ACCDATA lines [7:0].

LIDTSR-1 [7:0] – ACCDATA lines [15:8].

LIDTSR-2 [7:0] – ACCDATA lines [23:16].

LIDTSR-3 [7:0] – ACCDATA lines [31:24].

5.4.3.29 Gbit Ethernet Chip Interface Handshake Current/Toggle Status Register (GIHCTSR) (R), Addr=0x80

This status register provides toggle and current status information for the input handshaking signals from the previous LTA FPGA, originally sourcing from the Gbit Ethernet FPGA. These status bits are useful for determining which lines have failed when communications failures occur.

Bit 7 – LTA_CS_ – Toggle status of the active-low LTA_CS_ (chip select line). Each LTA FPGA has its own LTA_CS_ line, sourcing from the Gbit Ethernet FPGA. If set (1) the line toggle since this register was last read. If reset (0), it has not toggled. Cleared on read.

Bit 6 – LTA_SER_CMD – Toggle status of the LTA_SER_CMD input. If set (1) the line toggle since this register was last read. If reset (0), it has not toggled. Cleared on read.

Bit 5 – LTA_SER_RESP – Toggle status of the LTA_SER_RESP input. If set (1) the line toggle since this register was last read. If reset (0), it has not toggled. Cleared on read.

Bit 4 – LTA_FRAME_ – Toggle status of the LTA_FRAME input. If set (1) the line toggle since this register was last read. If reset (0), it has not toggled. Cleared on read.

Bit [3:0] – Same assignments as above, only these bits contain current level information for the respective lines.

5.4.3.30 Gbit Ethernet Chip Interface DATA Toggle Status Register (GIDTSR) (R), Addr=0x81

This 8-bit register provides toggle status information for the LTA_DATA [7:0] inputs from the previous LTA FPGA. If the LTA is row 7 (top of the board), then this register always reads 0x00, since there is no data input. If set (1), then the line has toggled since the last time this register was read. If reset (0), then the line has not toggled. Cleared on read. These status bits are useful for determining which line(s) are failing if the GigE FPGA is reporting checksum errors on inputs from the LTA FPGA columns.

5.4.3.31 Accel Programming Register (APR); (R/W) Addr=0x90

Writing **0x3A** to this register when the PU bit of the CCPCR register is low (i.e. the Corr Chip/Accel is powered off) causes the Accel regulator's (the point-of-load regulator that provides the Correlator Chip with its core supply voltage) NVRAM to be programmed with required parameters for normal operation, and sets correct parameters for chip operation. (N.B. if a mode which writes parameters to the Accel chip is needed without programming the NVRAM [i.e. to mitigate the 100,000 cycle NVRAM write limit], it will be done by writing 0xCA to this register; but this is not yet implemented).

The BIDCR-0 and BIDCR-1 registers **must** be set to the board's correct 16-bit serial number (range 2000-20FF hex) before this write occurs, because there are now 2 different versions of Accel regulators that must be supported, and each version has a different program bit sequence. Any particular board has only one version of Accel regulator on it:

S/Ns 2000-2008 inclusive—Original Accel version (chip marked "**VR2001-XXX**").

S/Ns 2009-20FF inclusive—Rev. C Accel version (chip marked "**VR2001C**").

If the BIDCR registers do not contain a valid serial number in the range 2000-20FF, programming will fail (bit 6 of this register set (1)). If an incorrect (but valid range) serial number is written to the BIDCR registers, it could result in incorrect Accel regulator operation, and if the correlator chip happens to short, it could result in burning up the Accel chip, the correlator chip, or both as short circuit current protection could be disabled with the wrong program.

Once the write occurs, it takes about 10 msec for it to take effect. If bit 7 of this register is set (1), then programming was successful. If bit 6 of this register is set (1), then programming was not successful; also, success is determined by examining the contents of the Accel Programming Readback Registers (APRR-0…7), **and** programming is not successful if these registers do not match the correct serial number of board. Both Bit 6 and Bit 7 are cleared on read.

Once programming is successful, the Accel/Correlator Chip may be powered up as usual (by pulling the PU bit of the CCPCR register high).

Note that the Accel chip's NVRAM can tolerate only about 100,000 program cycles; thus, this register should never be written unless it is determined that the Accel chip needs to be programmed.

5.4.3.32 Accel Programming Readback Registers (APRR-0…7); (R) Addr=0x91, 0x92, …, 0x98

After 0x3A is written to the APR, and bit 7 readback is successful, these registers contain the readback contents of the "Program Register" on the Accel chip. Program Register contents are according to the following table:

Register	MCB Address	Accel Program Register bits	Successful Value (S/Ns 2000-2008)	Successful Value (S/Ns 2009-20FF)
APRR-0	0x91	PR[7:0]	0x00	0x00
APRR-1	0x92	PR[15:8]	0x00	0x00
APRR-2	0x93	PR[23:16]	0x83	0x83
APRR-3	0x94	PR[31:24]	0xFF	0xEF
APRR-4	0x95	PR[39:32]	0xAF	0xAB
APRR-5	0x96	PR[47:40]	0x74	0x74
APRR-6	0x97	PR[56:48]	0x30	0x30
APRR-7	0x98	PR[63:57]	0x57	0x5F

5.4.3.33 Destination MAC Address Registers (DMAC-0...DMAC-5) (R/W) Addr=0xA0...0xA5

These R/W registers are used to set the destination MAC address of the GigE FPGA-generated IEEE 802.3 frame generated by the GigE FPGA. This address is sent by the LTA to the GigE FPGA with each LTA (or CC) data frame. DMAC-0 is at address 0xA0, and contains the LSByte of the DMAC address sent … DMAC-5 is at address 0xA5. If the V1, V0 bits of the MSR are both 0, this register does not exist.

5.4.3.34 LTA Transmit Control Register (LTCR); (R/W) Addr=0xB0

This register is used to allow the CPU to manually, or semi-manually via timed operation, control transmission of LTA data frames to the GigE FPGA.

LTCR (R/W) Addr=0xB0 Reset=0x05

7	6	5	4	3	2	1	0
hc-4	hc-3	hc-2	hc-1	hc-0	TxES	THEn	TxEn

TxEN (R/W) – Transmit ENable. This is the manual CPU-controlled transmit enable bit. For any LTA data to find its way to the output (i.e. GigE FPGA), this bit must be set (1). If this bit is reset (0), no data will be transmitted under any conditions.

THEn (R/W) – Transmit Holdoff Enable. If reset (0), then no transmit holdoff timing occurs and output data transmission is inhibited only if the TxEn bit is low. If set (1), the *first* occurrence of a LDS (Last Dump Save '010') "Cmmd" (refer to Figure 5-1, W1) OR a SDS (Single Dump Save '101') Cmmd in a correlator chip data frame results in an internal timer being started; while the timer is active, no data will be transmitted to the GigE FPGA, when the timer times out, data will be transmitted until such time as the timer is re-activated by detection of the next LDS or SDS Cmmd. The timer value, in milliseconds, is set with the hc-4:hc-0 bits in this register, and the hc-20:hc-5 bits in the LTH_COUNT-0,1 registers.

TxES (R) – Transmit Enable Status. This read-only bit indicates the current status of transmit enable as affected by the TxEN and/or THEn bits. If set (1), it indicates that transmission is enabled, if reset (0), it indicates that transmission is disabled.

hc-4 : hc-0 – This forms the least significant 5 bits of the 21-bit holdoff count value. The holdoff count is in units (steps) of 31.0303 μseconds. Reset value of [hc-4 : hc-0] is 0.

5.4.3.35 LTA Holdoff Count Register (LTH_COUNT-0, LTH_COUNT-1); (R/W) Addr=0xB1, 0xB2

These 2 registers, in addition to the upper 5 bits of the LTCR register, are set by the CPU to form a 21-bit transmit holdoff count, in units of 31.0303 μseconds, used when the THEn bit of the LTCR register is set.

Reset value of LTH_COUNT-0 is 0x01 and LTH_COUNT-1 is 0x00.

LTH_COUNT-0 contains bits [**hc-12 : hc-5**]. LTH_COUNT-1 contains bits [**hc-20 : hc-13**].

For predictable operation, these registers should be stable (unchanged), when the THEn bit is set.

Note that the 21-bits contained in these registers provide 31.0303 μsecond timing resolution over a range of 65 seconds. However, if this kind of timing resolution is used, care must be taken to ensure deterministic operation noting that data from all chips on the board must ultimately go out on one Gbit Ethernet link. With the Gig Ethernet FPGA readout scheduler accessing chips in a round-robin time-multiplexed fashion, depending on board configuration this can lead to indeterminate operation, packet collisions in the switch etc.

The figure below shows hc (holdoff count) assignments for the simplified case of maximum output performance (10 msec dumps) for all chips on 16 boards (say, one Baseline rack) going to 16 CPUs (CBE nodes), dumping all 16 CCCs on every chip. To obtain maximum board output frame rate of ~108 kframes/sec, requires that 4 chips in 4 separate columns are actively generating output frames at the same time. This is because the maximum frame rate cannot be obtained from just one or even two chips due to LTA RAM to output latency when the correlator chip is actively dumping to the LTA. (N.B. performance tests indicate with 1 chip dumping, the maximum frame rate is 63.9 kframes/sec, with 2 chips dumping 91.2 kframes/sec, with 3 or more dumping 108 kframes/sec.)

Board 1 (B1) is highlighted, showing that the GigE FPGA scheduler only reads from one set of 4 chips at a time within one timeslice, thereby achieving deterministic operation. Also, it is clear that it is never the case that data from more than one board goes to the same CPU at the same time. An hc count of 20 allows for 20 x 31.0303 usec = 620.6 usec to transmit 64 frames from 4 chips, or ~9.7 usec per frame. For this highest performance case, the GigE FPGA inter-frame delay must be set to 0.

If the dump time is increased to 100 msec, the diagram extends out to the right to 128 boards (a factor of 8 boards, with a factor of 10 time to do it in), and only 16 CPUs are required to handle the entire bandwidth, with bandwidth to spare (technically only 12.8 CPUs, but not a nice number to deal with!).

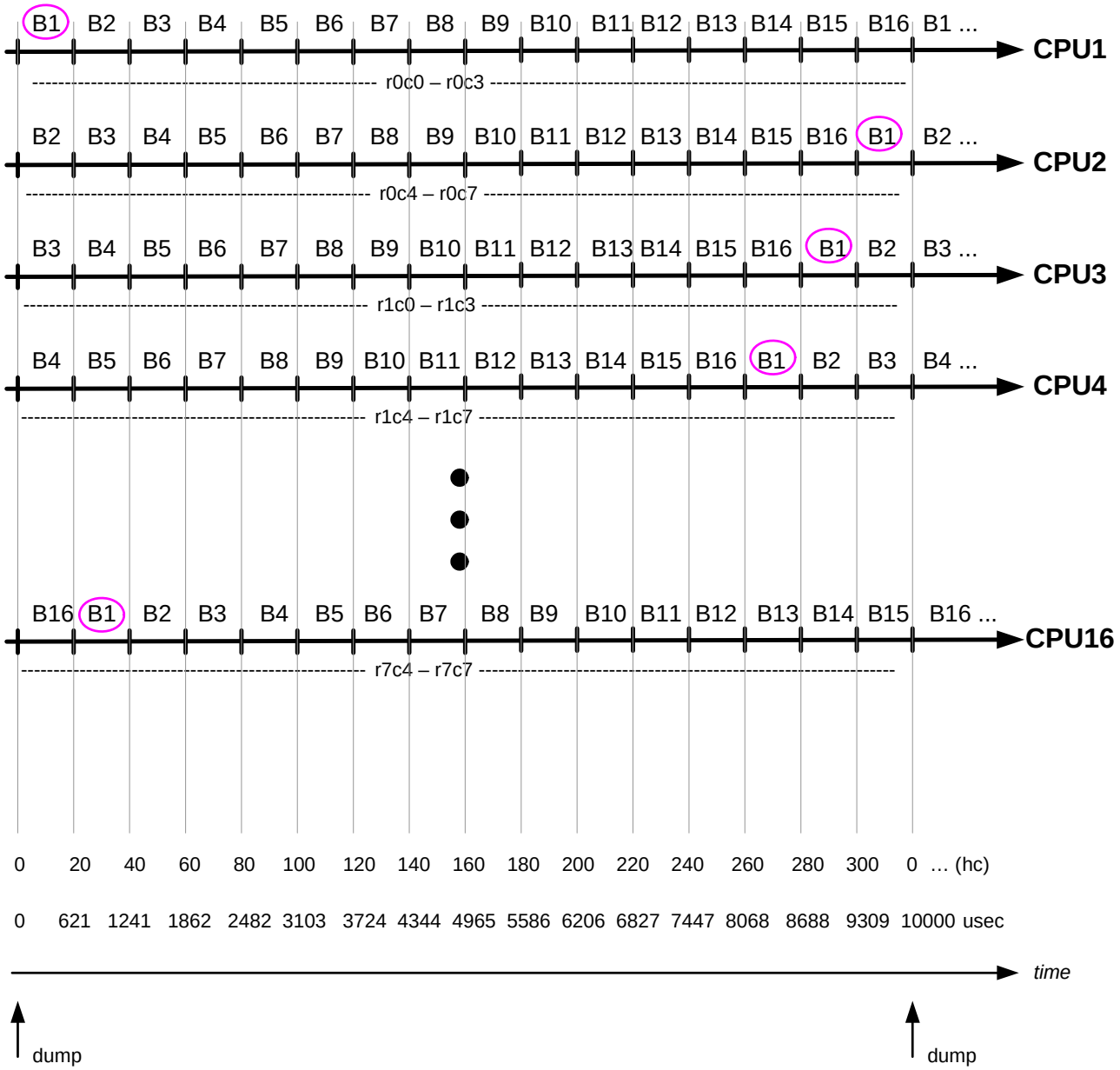


Figure 5-5 hc (holdoff count) assignments for 16 boards (B1-16) all chips (in groups of 4) to meet highest performance (10 msec) dump time requirements. Note that 16 CPUs (CBE nodes) are required for each Baseline rack, and so 128 CBE nodes are required in total, just to handle the total output bandwidth. If the dump time increases to 100 msec the diagram continues out to the right to 128 boards with hc incremented by 20 for each board step, requiring a total of only 16 CBE nodes to handle the entire bandwidth, with bandwidth to spare.

Note for Configuration Mapper: With 10 msec dumping, as it is only possible to shuffle a maximum of 16 board's worth of data (from only 4 chips on the board) to one CPU, to achieve 16,384 channels on one sub-band—with all data required for the FFT having to get to one CPU—requires a configuration where each chip is doing one baseline, all 2048 lags. 16 boards then produces 32,768 lags, or 16,384 channels.

5.4.3.36 Output Frame Count Registers (OFCOUNT-0, OFCOUNT-1) (R); Addr=0xB3, 0xB4

These registers form a 16-bit counter of the number of frames transmitted to the GigE FPGA *by this LTA* since last read. OFCOUNT-0 (Addr=0xB3) is the LSByte, and OFCOUNT-1 (Addr=0xB4) is the MSByte.

When OFCOUNT-0 is read, a snapshot of the internal frame counter is taken, the internal frame counter is cleared, and the LSByte of the snapshot is the read value. A subsequent read of the OFCOUNT-1 register is the MSByte of the snapshot value. Multiple reads of OFCOUNT-1 have no effect on the internal counter, and do not in themselves, take snapshots of the internal counter.

5.4.3.37 DUMMY R/W Register (R/W) Addr=0xFF

This 8-bit R/W register is to allow software to see if the chip is alive or not. This register does not control any functions in the chip.

5.4.3.38 Summary of Config/Status Registers and Addresses

Table 5-4 is a summary of all LTA Controller configuration and status registers and their associated addresses.

Table 5-4 Summary of LTA Controller Registers

Register (page)	R/W?	Address	Description
MCSR (35)	R/W	0x00	Master Control/Status Register
C3SCR (37)	R/W	0x01	CCC Statistics Control Register
MSR (38)	R/W	0x02	Miscl Status Register
CCPCR (40)	R/W	0x03	Correlator Chip Power Control Register
C3FREJCR (42)	R	0x08	CCC Frame Reject Count Register
C3CCFCR (44)	R	0x09	CCC Corr Chip Frame Count Register
C3LTAFCR (45)	R	0x0A	CCC LTA RAM Frame Count Register
C3OFCR (46)	R	0x0B	CCC Output Frame Count Register
C3SFCR (47)	R	0x0C	CCC Speed Frame Count Register
C3SFDCR (48)	R	0x0D	CCC Speed Frame Dropped Count Register
C3FCSR-0… C3FCSR-15 (49)	R/W	0x10… 0x1F	CCC Frame Control/Status Registers
SBYCR (51)	R/W	0x20	Start_BlockY Configuration Register
NBCR (51)	R/W	0x21	NBlocks Configuration Register
TBCR (51)	R/W	0x22	TotBlocks Configuration Register
CHIDCR (52)	R/W	0x23	ChipID Configuration Register
BIDCR-0 (53)	R/W	0x24	Board ID Configuration Register 0
BIDCR-1 (53)	R/W	0x25	Board ID Configuration Register 1
DIPADR-0… DIPADR-7 (54)	R/W	0x30… 0x37	Destination IP Address Registers
LTARADR-0… LTARADR-3 (55)	R/W	0x38… 0x3B	LTA RAM Address Registers
LTARDATA-0… LTARDATA-3 (56)	R/W	0x3C… 0x3F	LTA RAM Data Registers
LTARRWR (56)	R/W	0x40	LTA RAM Read/Write Register
LTARSTADR-0, LTARSTADR-1 (57)	R/W	0x48, 0x49	LTA RAM Semaphore Table Address Registers
LTARSTCR (57)	R/W	0x4A	LTA RAM Semaphore Table Control Reg.
LTARSTRW (58)	R/W	0x4B	LTA RAM Semaphore Table Read/Write Reg
LTADRR-0 (59)	R	0x50	LTA Depth Read Register-0
LTADRR-1 (59)	R/W	0x51	LTA Depth Read Register-1
LIHCSR (60)	R	0x60	LTA Interface Handshake Current Status Register
LIHTSR (60)	R	0x61	LTA Interface Handshake Toggle Status Register
CCCFDSR-0,1 (61)	R	0x62,	Correlator Chip CCC Frame Detect Status

		0x63	Registers 0, 1
LIDTSR-0… LIDTSR-3 (62)	R	0x70… 0x73	LTA Interface DATA Toggle Status Register
GIHCTSR (63)	R	0x80	Gbit Interface Handshake Current/Toggle Status Regiser
GIDTSR (63)	R	0x81	Gbit Interface DATA Toggle Status Register
APR (64)	R/W	0x90	Accel programming register
APRR-0…7 (65)	R	0x91…0x98	Accel programming readback register
DMAC-0…5 (66)	R/W	0xA0…0xA5	Destination MAC address registers.
LTCR (67)	R/W	0xB0	LTA Transmit Control Register
LTH_COUNT-0,1 (68)	R/W	0xB1, 0xB2	LTA Transmit Holdoff Count registers
OFCOUNT-0,1 (70)	R	0xB3, 0xB4	Output Frame Count read registers.
DUMMY (66)	R/W	0xFF	DUMMY read/write register.

5.5 Miscellaneous Requirements

1. Owing to the number of LTA Controllers on the Baseline Board (one for every Correlator Chip), it is necessary to keep the cost of this device to a minimum and below $20 USD. If necessary, functionality, except for the basic required functionality, must be sacrificed to keep within this cost limit. The design should thus be done in a way so that it is possible to cut any non-essential functionality as easily as possible (e.g. things like depth indication, some statistics etc.).

2. The chip must have JTAG test port capability.

3. The chip must be ISP (In-System Programmable), via a serial programming interface.

6 Functional Specifications

This section presents a plan for implementing the functionality of the LTA Controller chip. Since an adequate description of the requirements and functionality of the chip is provided for in preceding sections, this section will focus on some key implementation details. This description is for an LTA RAM with 2000 total bin capacity.

6.1 DDR SDRAM Interface

A Double Data Rate Synchronous Dynamic RAM (DDR SDRAM) chip will be used for the LTA RAM. This type of RAM is more difficult to use than Synchronous Static RAM (SRAM), however the LTA RAM memory requirements, and the need to minimize cost necessitates the use of dynamic RAM. Double data rate ram is chosen to maximize LTA accumulation performance, since that it is possible to read and write lag frames from/to the RAM at the same rate as lag frame data is transmitted from the Correlator Chip.

Interfacing to dynamic RAM is no easy task. There are initial setup requirements, CAS latency, burst requirements, refresh requirements, and double-edge clocking requirements. Thus, a target FPGA architecture has been chosen and an "IP core" has been purchased to minimize design effort. The target architecture is the Altera Cyclone FPGA (EP1C6F256C6) with 4000 Logic Elements, and the IP core is an Altera DDR SDRAM interface V2.2.0. A system-level block diagram[3] of this IP core and the interface to the DDR SDRAM is shown in Figure 6-1.

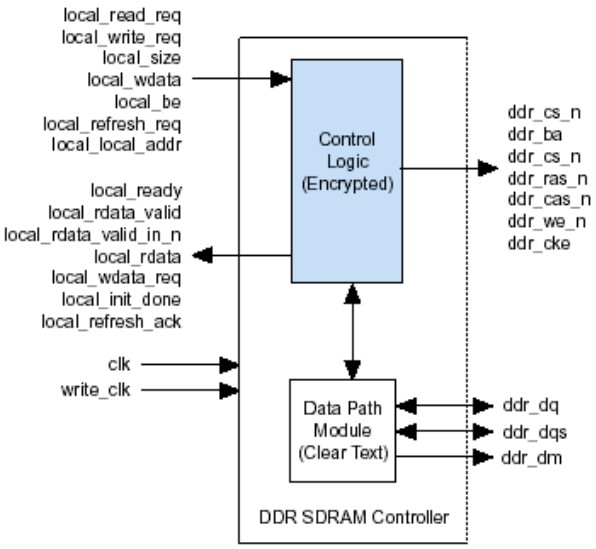


Figure 6-1 DDR SDRAM system-level block diagram. A simple handshake interface for reading and writing data is provided to the rest of the FPGA.

[3] Taken from [3] the Altera DDR SDRAM core IP User Guide, V2.1.0. The V2.2.0 interface has not changed.

The core provides a simple handshake interface for reading and writing data from/to the RAM, with selectable burst lengths. Since it is a double-data-rate RAM, the data width provided to the user is twice the width of the actual RAM data bus. In the LTA Controller case, the RAM is a 16M x 16 device, and thus the interface to the user is an 8M x 32 interface. This is an exact match to the LTA Controller requirements, since it operates on 32-bit wide data words at 128 MHz. The Altera "MegaWizard" allows the IP core to be configured to the user's and the DDR SDRAM requirements. The core takes about 800 logic elements, or 1/5th of the planned Cyclone part's resources.

A write-access timing diagram, also taken from the Altera DDR SDRAM core User Guide V2.1.0, is shown in Figure 6-2:

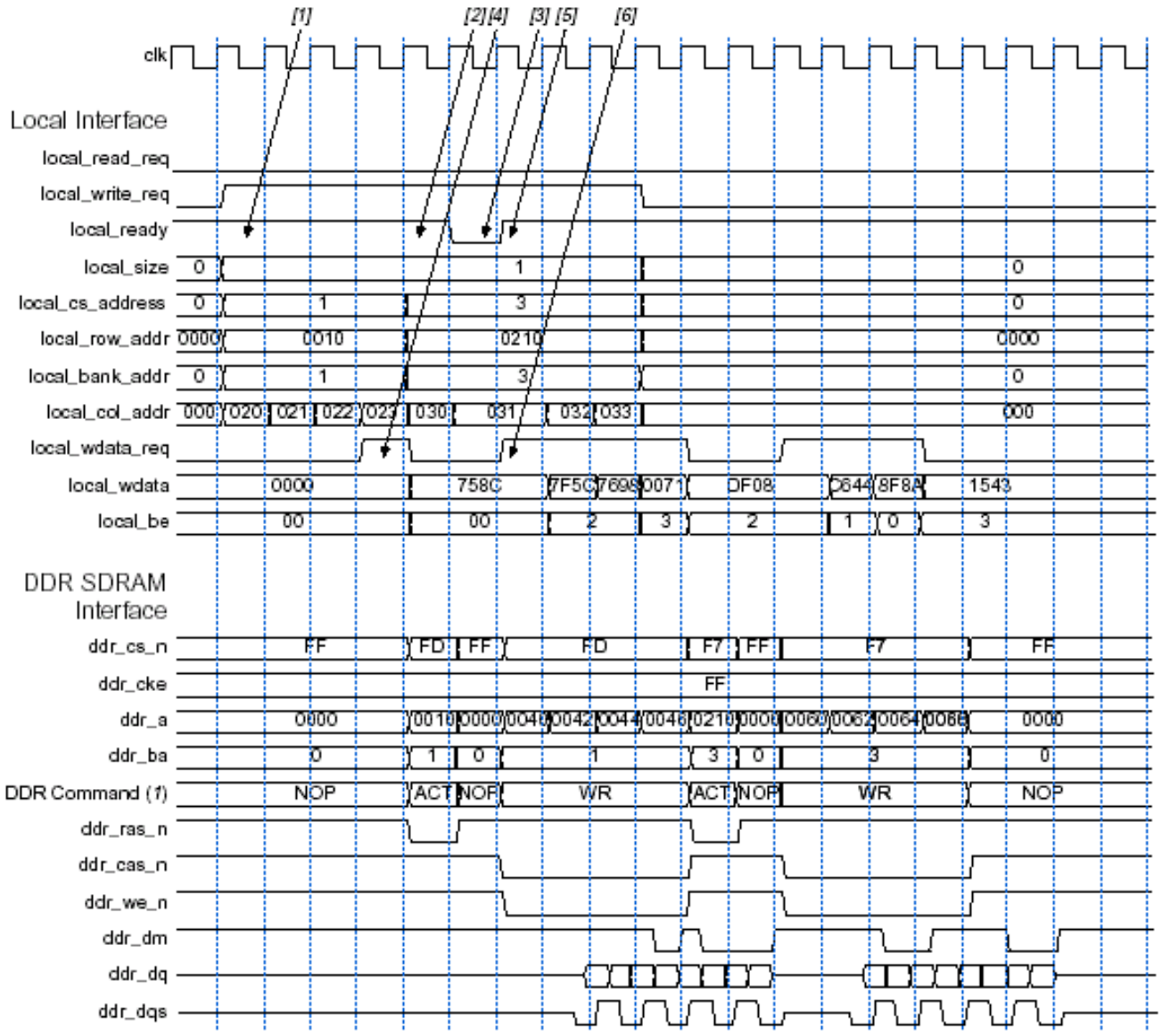


Figure 6-2 Write-access timing diagram using the Altera DDR SDRAM IP core. Writing consists of setting the burst size (local_size) and the write address and then asserting the write request (local_write_req) line. Data to write is provided when local_wdata_req is asserted.

A read-access timing diagram is shown in Figure 6-3.

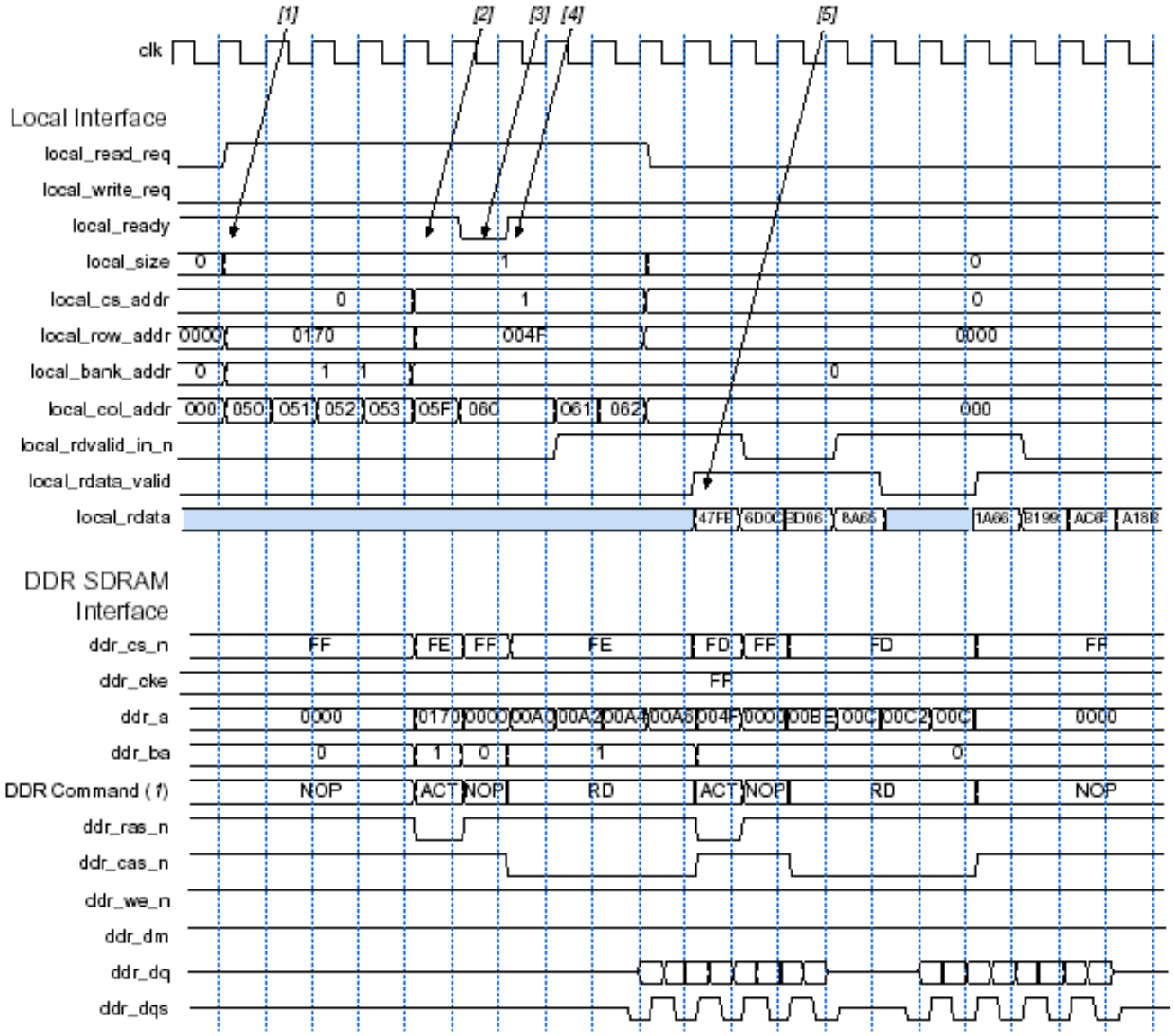


Figure 6-3 Read-access timing diagram for the DDR SDRAM IP core, also showing operation on the DDR SDRAM interface.

6.2 LTA RAM Memory Map

The LTA RAM memory map for an 8M x 32 (effective) memory, and 2000 total bins is described in detail in [2], and is repeated and clarified here for completeness. For 16M x 32, the memory map is double in size, but with the same basic addressing. Refer to the RM and PS bits of the MSR.

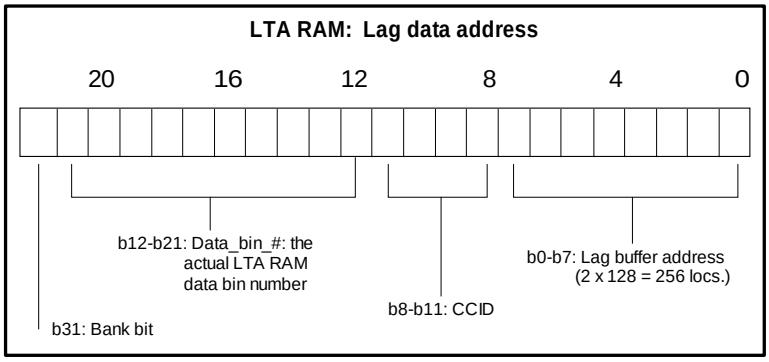
The lag data addressing and header data addressing bit assignments are shown in Figure 6-4, and the conceptual memory map for the LTA RAM is shown in Figure 6-5.

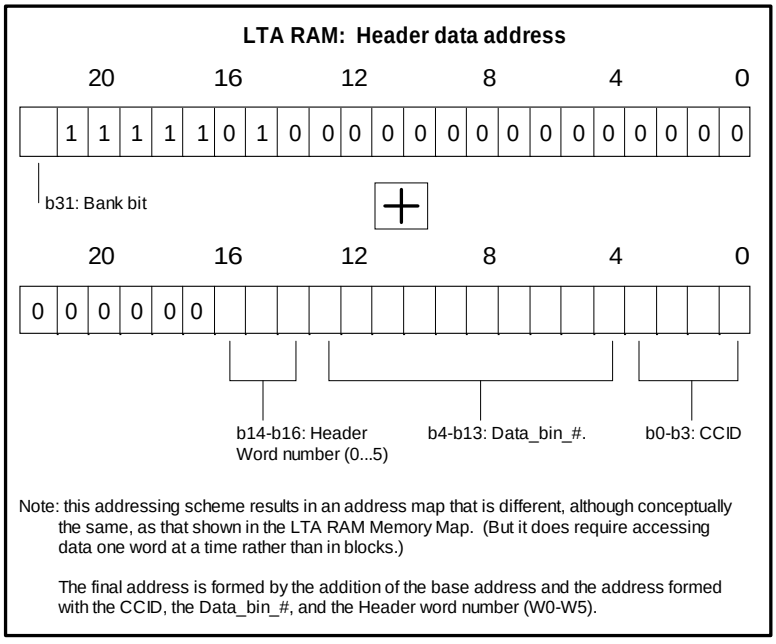
LTA Memory Access Equations & Addressing

LTA RAM data bin number calculation

Data_bin_# = Phase_bin x (Nblocks+1) + (RECIRC_BLK_Y - Start_Blk_Y)

where:
- -- **Data_bin_#** is the actual LTA RAM data bin number.
- -- **Phase_bin** is the phase bin number from the correlator chip data frame, not including the Bank bit.
- -- **Nblocks** is the number of lag blocks-1 being synthesized. This is set in the the LTA Controller NBCR register. Without any recirculation, this is 0. Data from each CCID must be enabled to use Nblocks or 0 (depending on whether recirculation is active for the CCID or not).
- -- **RECIRC_BLK_Y** is the 8-bit recirculation block number coming from the correlator chip.
- -- **Start_Blk_Y** is the Y start lag block the recirculation controller is handling. This is set in the SBYCR register.





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

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

Figure 6-4 LTA 8Mx32 RAM addressing scheme. This results in header data words being placed in non-contiguous locations, contrary to Figure 6-5.

LTA SDRAM Memory Map

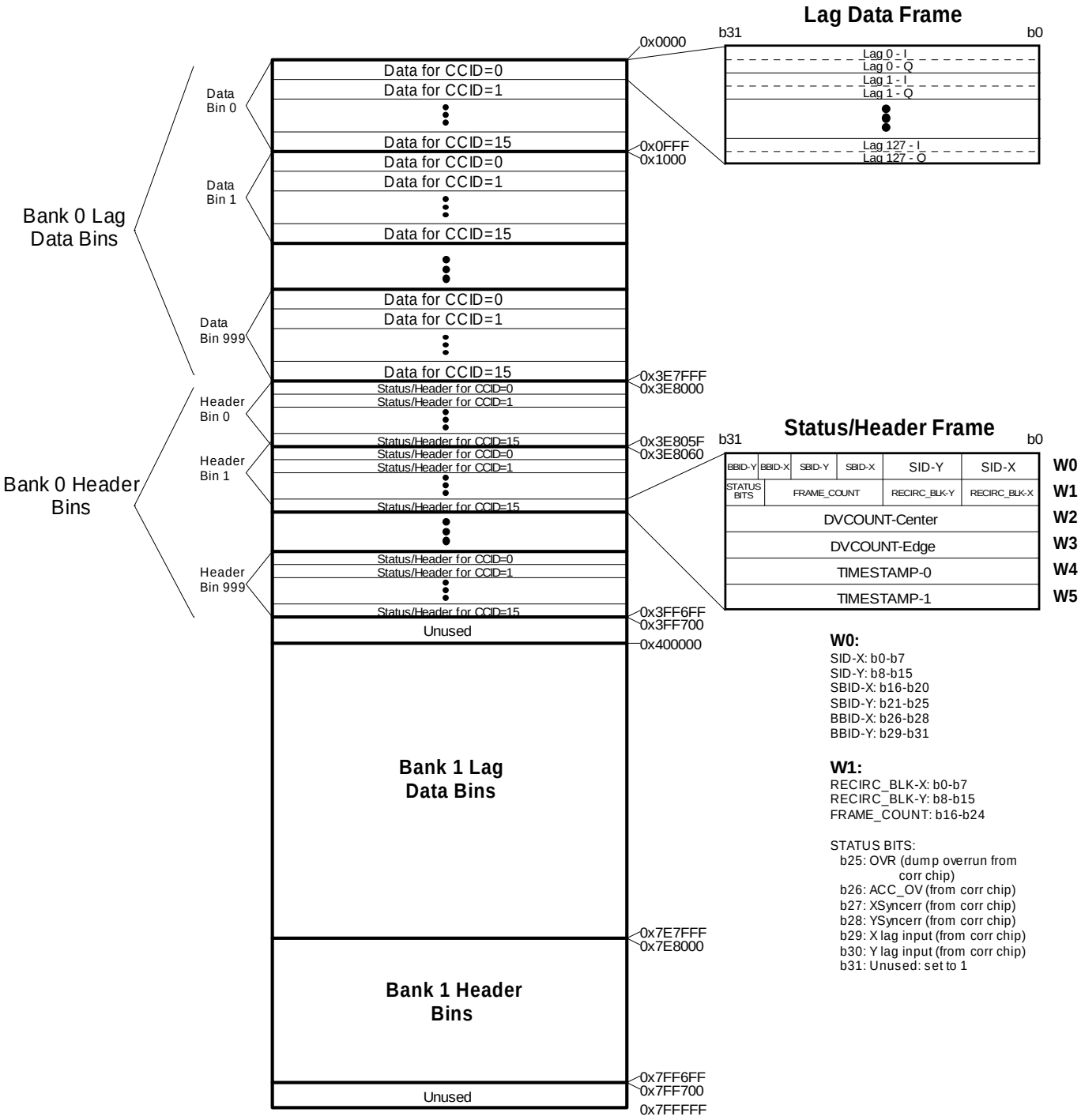


Figure 6-5 8Mx32 LTA RAM memory map. **Note that header frame words do not actually reside in contiguous memory as shown**. Refer to Figure 6-4 for correct header word addressing.

There are two important points to note in the addressing scheme and memory map diagrams:

1. The "Bank bit" is **always** taken from the MSBit of the correlator lag frame Phase Bin, even if the LTA is being conceptually operated as a single bank of up to 2000 phase bins.

2. There are **exactly** 2000 bins[4] (numbered **bank 0**, 0…999, and **bank 1**, 0…999) available, not 2048 bins. The extra locations in memory are needed for holding header information. Thus, even when the LTA is conceptually being operated in single bank, 2000 bin, mode, the first 10 bits of the Correlator Chip lag frame Phase Bin number must only go as high as 999. If the Phase Bin number goes higher it will result in the frame being rejected, the LTAF bit of the corresponding C3FCSR register being set, and will not overwrite existing memory.

[4] Or, 4000 bins, depending on the RM and PS bits of the MSR.

7 References

[1] Carlson, Brent, REQUIREMENTS AND FUNCTIONAL SPECIFICATION: EVLA Correlator Chip, RFS Document number A25082N0000, Rev. 2.2, October 7, 2004.

[2] Carlson, Brent, Refined EVLA WIDAR Correlator Architecture, NRC-EVLA Memo# 014, October 2, 2001.

[3] Altera Megacore Function User Guide, DDR SDRAM Controller, V2.1.0.

[4] Carlson, Brent, REQUIREMENTS AND FUNCTIONAL SPECIFICATION: Gbit Ethernet Chip, RFS Document number A25092N0000, Rev. DRAFT, April 5, 2004.

[5] Carlson, Brent, INTERFACE CONTROL DOCUMENT: EVLA Correlator System Numbering Plan, ICD Document number A25010N0002, Rev. 1.0, November 9, 2004.

8 Appendix I – LTA Pinouts, Pin, Package, and Programming Notes

This section contains an exhaustive chip pinout listing generated by the FPGA compiler, the Quartus-II Pin Planner output graphic, package, and programming notes. The LTA is implemented in an Altera Cyclone EP1C12F256C6 FPGA, and compiled using Quartus-II V6.1.

Pin Name/Usage	Location	Dir.	I/O Standard	Voltage	I/O Bank
GND	A1	gnd			
GND*	A2				2
VCCIO2	A3	power		2.5V	2
ddr_dq[6]	A4	bidir	SSTL-2 Class II		2
GND	A5	gnd			
ddr_cs_n	A6	output	SSTL-2 Class II		2
VCCINT	A7	power		1.5V	
ddr_a[1]	A8	output	SSTL-2 Class II		2
ddr_ras_n	A9	output	SSTL-2 Class II		2
VCCINT	A10	power		1.5V	
ddr_a[5]	A11	output	SSTL-2 Class II		2
GND	A12	gnd			
ddr_dq[1]	A13	bidir	SSTL-2 Class II		2
VCCIO2	A14	power		2.5V	2
GND*	A15				2
GND	A16	gnd			
GND*	B1				1
ACCEL_TSTO_pad	B2	input	2.5 V		2
GND*	B3				2
ddr_dq[7]	B4	bidir	SSTL-2 Class II		2
ddr_dq[5]	B5	bidir	SSTL-2 Class II		2
GND*	B6				2
ddr_a[2]	B7	output	SSTL-2 Class II		2
ddr_a[8]	B8	output	SSTL-2 Class II		2
ddr_a[4]	B9	output	SSTL-2 Class II		2
ddr_a[6]	B10	output	SSTL-2 Class II		2
ddr_a[0]	B11	output	SSTL-2 Class II		2
ddr_dq[2]	B12	bidir	SSTL-2 Class II		2
ddr_dq[0]	B13	bidir	SSTL-2 Class II		2
GND*	B14				2
SPARE_pad[0]	B15	output	2.5 V		2
GND*	B16				3
VCCIO1	C1	power		2.5V	1
GND*	C2				1
LTA_SER_CMD_I_pad	C3	input	SSTL-2 Class I		1
ddr_cke	C4	output	SSTL-2 Class II		2
ddr_dq[4]	C5	bidir	SSTL-2 Class II		2
GND*	C6				2
ddr_a[12]	C7	output	SSTL-2 Class II		2
ddr_a[9]	C8	output	SSTL-2 Class II		2
ddr_ba[1]	C9	output	SSTL-2 Class II		2
ddr_cas_n	C10	output	SSTL-2 Class II		2
GND*	C11				2
ddr_dq[3]	C12	bidir	SSTL-2 Class II		2
ddr_spare2	C13	output	2.5 V		2
ACCEL_PDN_pad_	C14	output	2.5 V		3

GND*	C15					3
VCCIO3	C16	power			2.5V	3
GND*	D1					1
LTA_SER_RESP_I_pad	D2	input	SSTL-2 Class I			1
GND*	D3					1
GND*	D4					1
LTA_DATA_O_pad[5]	D5	output	SSTL-2 Class I			2
GND*	D6					2
ddr_ba[0]	D7	output	SSTL-2 Class II			2
ddr_we_n	D8	output	SSTL-2 Class II			2
ddr_dm[0]	D9	output	SSTL-2 Class II			2
ddr_a[7]	D10	output	SSTL-2 Class II			2
ddr_a[3]	D11	output	SSTL-2 Class II			2
SPARE_pad[2]	D12	output	2.5 V			2
ACCEL_SCK_pad	D13	output	2.5 V			3
ddr_dq[9]	D14	bidir	SSTL-2 Class II			3
GND*	D15					3
GND*	D16					3
GND*	E1					1
LTA_DATA_I_pad[5]	E2	input	SSTL-2 Class I			1
LTA_SER_RESP_O_pad	E3	output	SSTL-2 Class I			1
LTA_DATA_I_pad[4]	E4	input	SSTL-2 Class I			1
LTA_DATA_O_pad[4]	E5	output	SSTL-2 Class I			2
VREF2B2	E6				1.25V	2
GND*	E7					2
ddr_a[11]	E8	output	SSTL-2 Class II			2
GND*	E9					2
ddr_a[10]	E10	output	SSTL-2 Class II			2
VREF0B2	E11				1.25V	2
ddr_dqs[0]	E12	bidir	SSTL-2 Class II			2
ddr_dq[10]	E13	bidir	SSTL-2 Class II			3
ddr_dq[11]	E14	bidir	SSTL-2 Class II			3
GND*	E15					3
GND*	E16					3
GND*	F1					1
GND*	F2					1
LTA_SER_CMD_O_pad	F3	output	SSTL-2 Class I			1
CORR_SENSEOUT_pad	F4	input	2.5 V			1
ACCEL_SDI_pad	F5	output	2.5 V			1
GND	F6	gnd				
VCCIO2	F7	power			2.5V	2
GND	F8	gnd				
GND	F9	gnd				
VCCIO2	F10	power			2.5V	2
GND	F11	gnd				
ddr_dqs[1]	F12	bidir	SSTL-2 Class II			3
GND*	F13					3
LTA_DATA_I_pad[0]	F14	input	SSTL-2 Class I			3
ACCDATA_pad[22]	F15	input	2.5 V			3
ACCDATA_pad[31]	F16	input	2.5 V			3
DATA_CLKOUT_pad	G1	input	2.5 V			1
LTA_CS_pad_	G2	input	SSTL-2 Class I			1
GND*	G3					1
~nCSO~ / RES_IN_WEAK_PULLUP	G4	input	2.5 V			1
VREF0B1	G5				1.25V	1
VCCIO1	G6	power			2.5V	1
GND	G7	gnd				
VCCINT	G8	power			1.5V	
GND	G9	gnd				
VCCINT	G10	power			1.5V	
GND	G11	gnd				
VREF0B3	G12				1.25V	3
LTA_DATA_O_pad[0]	G13	output	SSTL-2 Class I			3

ddr_dm[1]	G14	output	SSTL-2 Class II		3	
ACCDATA_pad[29]	G15	input	2.5 V		3	
ACCDATA_pad[20]	G16	input	2.5 V		3	
MCB_CLK_pad	H1	input	2.5 V		1	
DATA0	H2	input			1	
nCONFIG	H3				1	
nCEO	H4				1	
ACCEL_PGD_pad	H5	input	2.5 V		1	
VCCA_PLL1	H6	power		1.5V		
VCCINT	H7	power		1.5V		
GND	H8	gnd				
VCCINT	H9	power		1.5V		
GND	H10	gnd				
VCCA_PLL2	H11	power		1.5V		
VREF1B3	H12			1.25V	3	
ddr_dq[8]	H13	bidir	SSTL-2 Class II		3	
TDI	H14	input			3	
TDO	H15	output			3	
CLK3125_I_pad	H16	input	SSTL-2 Class I		3	
MCB_DATA_pad[5]	J1	bidir	2.5 V		1	
MSEL1	J2				1	
MSEL0	J3				1	
nCE	J4				1	
GNDG_PLL1	J5	gnd				
GNDA_PLL1	J6	gnd				
GND	J7	gnd				
VCCINT	J8	power		1.5V		
GND	J9	gnd				
VCCINT	J10	power		1.5V		
GNDA_PLL2	J11	gnd				
GNDG_PLL2	J12	gnd				
nSTATUS	J13				3	
TCK	J14	input			3	
TMS	J15	input			3	
CLK3125_O_pad	J16	output	SSTL-2 Class I		3	
MCB_DATA_pad[2]	K1	bidir	2.5 V		1	
MCB_DATA_pad[3]	K2	bidir	2.5 V		1	
~ASDO~ / RES_IN_WEAK_PULLUP	K3	input	2.5 V		1	
DCLK	K4	bidir			1	
MCB_DATA_pad[7]	K5	bidir	2.5 V		1	
GND	K6	gnd				
VCCINT	K7	power		1.5V		
GND	K8	gnd				
VCCINT	K9	power		1.5V		
GND	K10	gnd				
VCCIO3	K11	power		2.5V	3	
VREF2B3	K12			1.25V	3	
CONF_DONE	K13				3	
ddr_dq[14]	K14	bidir	SSTL-2 Class II		3	
ACCDATA_pad[19]	K15	input	2.5 V		3	
LTA_DATA_I_pad[7]	K16	input	SSTL-2 Class I		3	
MCB_DATA_pad[4]	L1	bidir	2.5 V		1	
MCB_ADDR_pad[7]	L2	input	2.5 V		1	
MCB_DATA_pad[0]	L3	bidir	2.5 V		1	
LTA_DATA_O_pad[3]	L4	output	SSTL-2 Class I		1	
RESET_pad_	L5	input	2.5 V		1	
GND	L6	gnd				
VCCIO4	L7	power		2.5V	4	
GND	L8	gnd				
GND	L9	gnd				
VCCIO4	L10	power		2.5V	4	
GND	L11	gnd				
ACCDATA_pad[5]	L12	input	2.5 V		3	

ACCDATA_pad[16]	L13	input	2.5 V		3	
ACCDATA_pad[28]	L14	input	2.5 V		3	
ACCDATA_pad[11]	L15	input	2.5 V		3	
LTA_DATA_O_pad[7]	L16	output	SSTL-2 Class I		3	
MCB_ADDR_pad[6]	M1	input	2.5 V		1	
MCB_RDWR_pad_	M2	input	2.5 V		1	
MCB_DATA_pad[1]	M3	bidir	2.5 V		1	
ddr_spare1	M4	output	2.5 V		1	
MCB_ADDR_pad[3]	M5	input	2.5 V		4	
VREF2B4	M6			1.25V	4	
CORR_RESET_pad_	M7	output	2.5 V		4	
GND*	M8				4	
ACCDATA_pad[23]	M9	input	2.5 V		4	
ACCDATA_pad[4]	M10	input	2.5 V		4	
clk_to_sdram	M11	output	SSTL-2 Class II		4	
ACCDATA_pad[21]	M12	input	2.5 V		4	
ddr_dq[12]	M13	bidir	SSTL-2 Class II		3	
ACCDATA_pad[6]	M14	input	2.5 V		3	
ACCDATA_pad[25]	M15	input	2.5 V		3	
ACCDATA_pad[14]	M16	input	2.5 V		3	
MCB_CS_pad_	N1	input	2.5 V		1	
DATA_CS_pad_	N2	output	2.5 V		1	
MCB_DATA_pad[6]	N3	bidir	2.5 V		1	
LTA_DATA_O_pad[1]	N4	output	SSTL-2 Class I		1	
MCB_ADDR_pad[5]	N5	input	2.5 V		4	
MCB_ADDR_pad[4]	N6	input	2.5 V		4	
PLL_LOCKED_pad	N7	output	2.5 V		4	
GND*	N8				4	
ACCDATA_pad[3]	N9	input	2.5 V		4	
ACCDATA_pad[12]	N10	input	2.5 V		4	
ACCDATA_pad[15]	N11	input	2.5 V		4	
ACCDATA_pad[17]	N12	input	2.5 V		4	
ACCDATA_pad[7]	N13	input	2.5 V		3	
ddr_dq[13]	N14	bidir	SSTL-2 Class II		3	
ddr_dq[15]	N15	bidir	SSTL-2 Class II		3	
ACCDATA_pad[0]	N16	input	2.5 V		3	
VCCIO1	P1	power		2.5V	1	
LTA_DATA_O_pad[2]	P2	output	SSTL-2 Class I		1	
SPARE_pad[1]	P3	output	2.5 V		1	
LTA_DATA_I_pad[3]	P4	input	SSTL-2 Class I		4	
MCB_ADDR_pad[0]	P5	input	2.5 V		4	
CORR_PLL_RESET_pad_	P6	output	2.5 V		4	
ACCEL_SMS_pad	P7	output	2.5 V		4	
clk_to_sdram_n	P8	output	SSTL-2 Class II		4	
ACCDATA_pad[2]	P9	input	2.5 V		4	
ACCDATA_pad[9]	P10	input	2.5 V		4	
DATA_VALID_pad_	P11	input	2.5 V		4	
ACCDATA_pad[10]	P12	input	2.5 V		4	
GND*	P13				4	
CLK3125_ACCEL_pad	P14	output	2.5 V		3	
ACCDATA_pad[18]	P15	input	2.5 V		3	
VCCIO3	P16	power		2.5V	3	
LTA_FRAME_O_pad_	R1	output	2.5 V		1	
LTA_DATA_O_pad[6]	R2	output	SSTL-2 Class I		4	
LTA_DATA_I_pad[6]	R3	input	SSTL-2 Class I		4	
LTA_DATA_I_pad[2]	R4	input	SSTL-2 Class I		4	
MCB_ADDR_pad[1]	R5	input	2.5 V		4	
LTA_DATA_I_pad[1]	R6	input	SSTL-2 Class I		4	
GND*	R7				4	
GND*	R8				4	
SPARE_pad[3]	R9	output	2.5 V		4	
ACCDATA_pad[1]	R10	input	2.5 V		4	
ACCDATA_pad[24]	R11	input	2.5 V		4	

```
ACCDATA_pad[30]              : R12       : input  : 2.5 V             :         : 4         :
ACCDATA_pad[26]              : R13       : input  : 2.5 V             :         : 4         :
ACCDATA_pad[8]               : R14       : input  : 2.5 V             :         : 4         :
DATA_RDY_pad_                : R15       : input  : 2.5 V             :         : 4         :
GND*                         : R16       :        :                   :         : 3         :
GND                          : T1        : gnd    :                   :         :           :
LTA_FRAME_I_pad_             : T2        : input  : SSTL-2 Class I    :         : 4         :
VCCIO4                       : T3        : power  :                   : 2.5V    : 4         :
MCB_ADDR_pad[2]              : T4        : input  : 2.5 V             :         : 4         :
GND                          : T5        : gnd    :                   :         :           :
FRAME_ABORT_pad_             : T6        : output : 2.5 V             :         : 4         :
VCCINT                       : T7        : power  :                   : 1.5V    :           :
GND*                         : T8        :        :                   :         : 4         :
GND*                         : T9        :        :                   :         : 4         :
VCCINT                       : T10       : power  :                   : 1.5V    :           :
DATA_OE_pad_                 : T11       : output : 2.5 V             :         : 4         :
GND                          : T12       : gnd    :                   :         :           :
ACCDATA_pad[27]              : T13       : input  : 2.5 V             :         : 4         :
VCCIO4                       : T14       : power  :                   : 2.5V    : 4         :
ACCDATA_pad[13]              : T15       : input  : 2.5 V             :         : 4         :
GND                          : T16       : gnd    :                   :         :           :
-------------------------------------------------------------------------------------------
 -- NC            : No Connect. This pin has no internal connection to the device.
 -- VCCINT        : Dedicated power pin, which MUST be connected to VCC  (1.5V).
 -- VCCIO         : Dedicated power pin, which MUST be connected to VCC
 --                  of its bank.
 --                                  Bank 1:        2.5V
 --                                  Bank 2:        2.5V
 --                                  Bank 3:        2.5V
 --                                  Bank 4:        2.5V
 -- GND           : Dedicated ground pin. Dedicated GND pins MUST be connected to GND.
 --
 -- GND+          : Unused input pin. It can also be used to report unused dual-purpose pins.
 --                  This pin should be connected to GND. It may also be connected  to a
 --                  valid signal  on the board  (low, high, or toggling)  if that signal
 --                  is required for a different revision of the design.
 -- GND*          : Unused  I/O  pin.   This pin can either be left unconnected or
 --                  connected to GND.  Connecting this pin to GND will improve the
 --                  device's immunity to noise.
 -- RESERVED      : Unused I/O pin, which MUST be left unconnected.
 -- RESERVED_INPUT    : Pin is tri-stated and should be connected to the board.
 -- RESERVED_INPUT_WITH_WEAK_PULLUP    : Pin is tri-stated with internal weak pull-up resistor.
 -- RESERVED_INPUT_WITH_BUS_HOLD       : Pin is tri-stated with bus-hold circuitry.
 -------------------------------------------------------------------------------------------
```

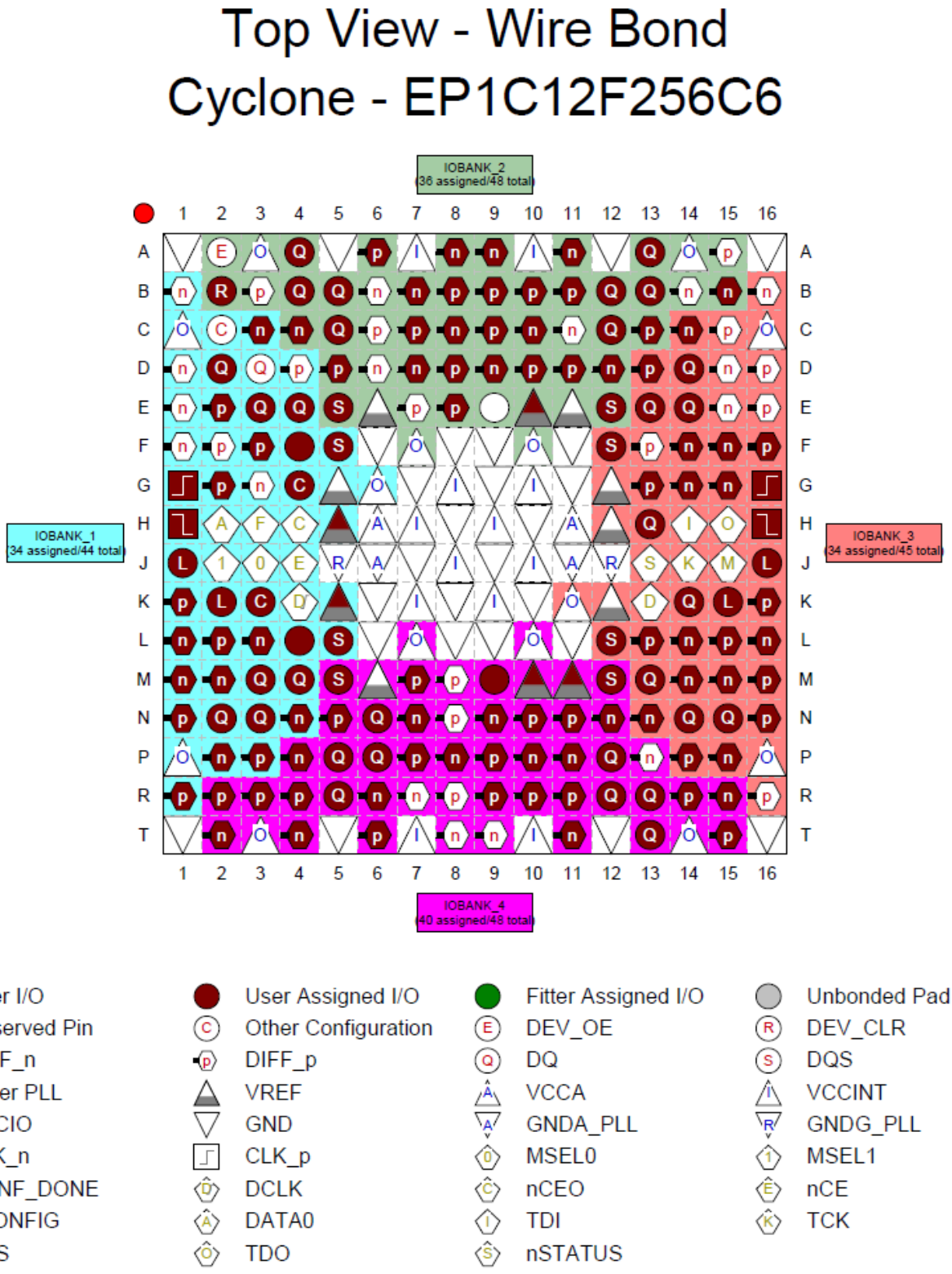
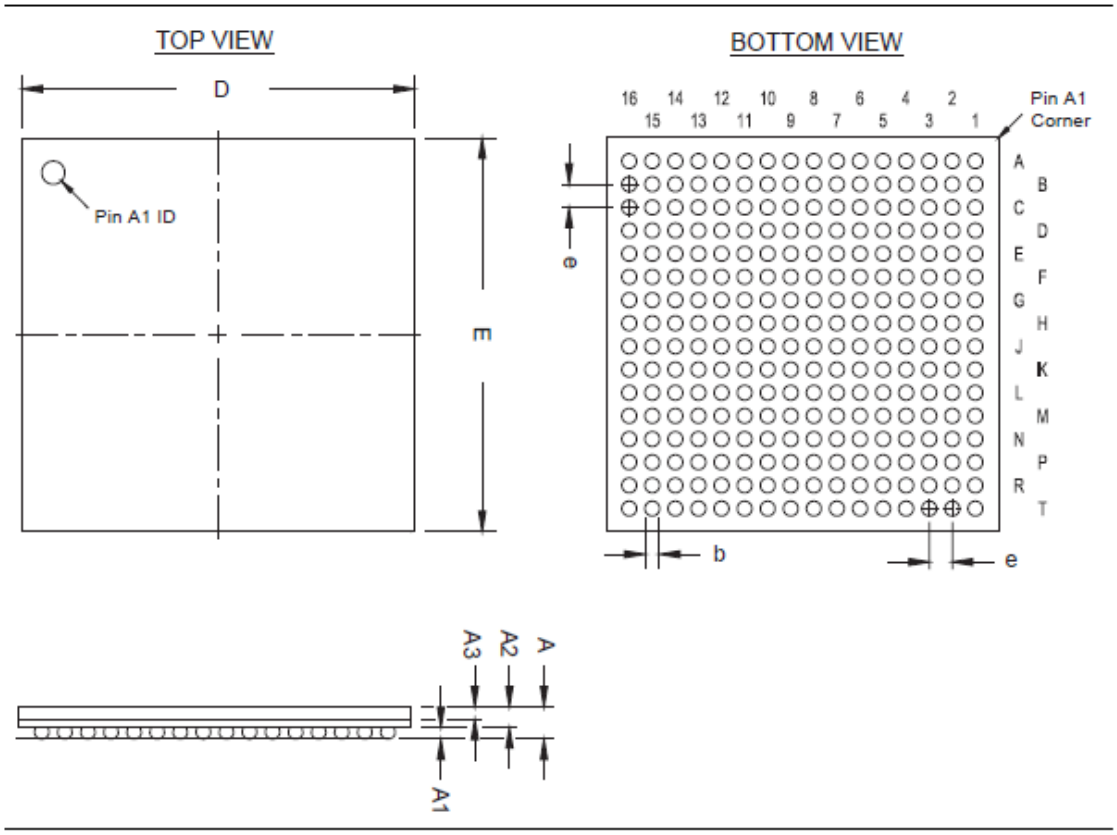


Figure 8-1 Quartus-II V6.1 pin planner output of the LTA FPGA.

8.1 Altera EP1C12F256C6 FBGA Package Drawing



Package Information	
Description	**Specification**
Ordering Code Reference	F
Package Acronym	FBGA
Substrate Material	BT
Solder Ball Composition	Regular: 63Sn:37Pb (Typ.) Pb-free: Sn:3Ag:0.5Cu (Typ.)
JEDEC Outline Reference	MS-034 Variation: AAF-1
Lead Coplanarity	0.008 inches (0.20 mm)
Weight	1.5 g (Typ.)
Moisture Sensitivity Level	Printed on moisture barrier bag

Package Outline Dimension Table			
Symbol	**Millimeters**		
	Min.	**Nom.**	**Max.**
A	–	–	2.20
A1	0.30	–	–
A2	–	–	1.80
A3		0.70 REF	
D		17.00 BSC	
E		17.00 BSC	
b	0.50	0.60	0.70
e		1.00 BSC	

Figure 8-2 LTA FPGA package information.

8.2 Altera FPGA Programming Notes

That Baseline Board on which the LTA FPGAs reside is set up for 1-bit Passive Serial programming via the PCMC FPGA. The LTA FPGAs are in series in each column, and each column is programmed separately; this allows each FPGA on the board to have a different personality, although currently only one personality is used throughout. The following figure is a screen shot of the "Convert Programming Files" dialog in the Altera Quartus-II V6.1 software, showing the setup required to produce the .rbf (raw binary file) needed by the software to program the FPGA.

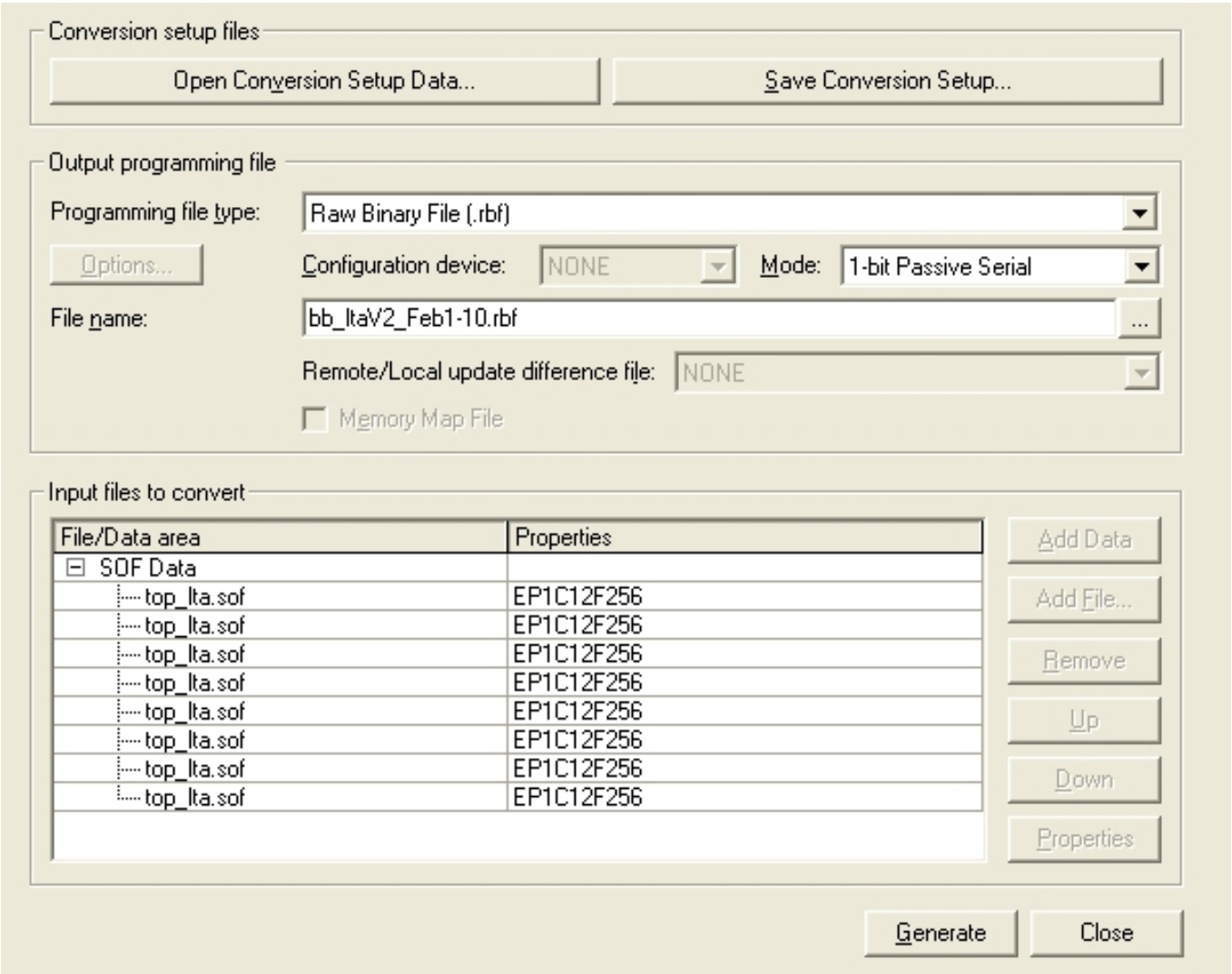


Figure 8-3 Altera Quartus-II "Convert Programming Files" screenshot showing settings necessary for generating the .rbf file for the LTA FPGAs.

9 Appendix II – Accel Programming Notes

The LTA FPGA contains logic to program the Accel Regulator's internal NVRAM (non-volatile RAM). The NVRAM contains information specific to frequencies and dividers required for the correlator chip ASIC. The Accel is provided a 0.947 MHz reference clock (from the GigE FPGA via the TCK distribution network) and works in a feedback loop with a ring oscillator output of the correlator chip to maintain the correlator chip's core voltage, nominally set for 1.02 V, high enough to maintain required speed. This minimizes power dissipation while allowing compensation of variations in chip-to-chip process speed, and process speed degradation over time. If the feedback loop gets broken, the regulator drifts to its highest programmed output of 1.2 V. **On rare occasion if the NVRAM gets wiped, the output can go up to 1.8 V. The correlator chip should never have its PLL or core RESET lines released, or operated for any length of time under these conditions**.

Two versions of the Accel are supported:

- Parts marked "**VR2001-XXX**" are original parts contained on Baseline Boards with serial numbers up to 2008 inclusive. These parts also have output current limits that are just barely capable of working with the correlator chip. These parts also see the NVRAM occasionally being erased for no reason; the original reason for implementing programming in the LTA FPGA.

- Parts marked "**VR2001C**" are new parts with a different NVRAM program; the current limit has been relaxed somewhat. The parts don't seem to see NVRAM erasure. These parts are used on all Baseline Boards with serial numbers 2009 and higher.

The Accel NVRAM essentially consists of a 64-bit Program Register "PR".

For **VR2001-XXX** parts, the contents of this register are set as follows:

```
// PROGRAMMING for 0.947 MHz operation, clock supplied by GigE FPGA; VR2001-XXX
// Program Register, Bank 0
    PR[7:0]  = 8'h00;  // Set PWMCK freq divide/multiply: fRCK=0.947 MHz, divide by 1, fPWMCK=0.947MHz
    PR[15:8] = 8'h00;  // Set PLLRCK freq divide/multiply: fRCK=0.947 MHz, divide by 1, fPLLRCK=0.947MHz
    PR[21:16]= 6'h03;  // Set PLLSCK freq divide: fSCK=3.80 MHz, divide by 4, fPLLSCK=0.95MHz
    PR[22]   = 1'b0;   // En/Dis external compensation: Disable
    PR[24:23]= 2'h3;   // Set VX output drive strength: set to max
    PR[25]   = 1'b1;   // En/Dis sensor feedback
    PR[26]   = 1'b1;   // En/Dis thermal protection: enable
    PR[28:27]=2'h3;    // Set compensation element R7
    PR[31:29]=3'h7;    // Set compensation element C3

// Program Register, Bank 1
    PR[34:32]=3'h7;    // Set bias current: Set to max
    PR[41:35]=7'h15;   // Set output voltage nominal value: Vout=1.02V
    PR[48:42]=7'h1d;   // Set output voltage min limit: 0.95 V (-6.86 %)
    PR[55:49]=7'h18;   // Set output voltage max limit: 1.2 V (+17.65 %)
    PR[58:56]=3'h7;    // Set short cct current limit: max
    PR[60:59]=2'h2;    // Set compensation element R4
    PR[63:61]=3'h2;    // Set compensation element C2
```

For **VR2001C** parts, the contents of this register are set as follows:

```
// PROGRAMMING for 0.947 MHz operation, clock supplied by GigE FPGA; VR2001C
// Program Register, Bank 0
    PR[7:0]  = 8'h00; // Set PWMCK freq divide/multiply: fRCK=0.947 MHz, divide by 1, fPWMCK=0.947MHz
    PR[15:8] = 8'h00;  // Set PLLRCK freq divide/multiply: fRCK=0.947 MHz, divide by 1, fPLLRCK=0.947MHz
    PR[21:16]= 6'h03;  // Set PLLSCK freq divide: fSCK=3.80 MHz, divide by 4, fPLLSCK=0.95MHz
    PR[22]   = 1'b0;   // En/Dis external compensation: Disable
    PR[24:23]= 2'h3;   // Set VX output drive strength: set to max
    PR[25]   = 1'b1;   // En/Dis sensor feedback
    PR[26]   = 1'b1;   // En/Dis thermal protection: enable
    PR[28:27]=2'h1;    // Set compensation element R7 ****REV C**** (was 3)
    PR[31:29]=3'h7;    // Set compensation element C3

// Program Register, Bank 1
    PR[34:32]=3'h3;    // Set bias current: Set to max ****REV C**** (was 7)
    PR[41:35]=7'h15;   // Set output voltage nominal value: Vout=1.02V
    PR[48:42]=7'h1d;   // Set output voltage min limit: 0.95 V (-6.86 %)
    PR[55:49]=7'h18;   // Set output voltage max limit: 1.2 V (+17.65 %)
    PR[58:56]=3'h7;    // Set short cct current limit: max
    PR[60:59]=2'h3;    // Set compensation element R4  ****REV C**** (was 2)
    PR[63:61]=3'h2;    // Set compensation element C2
```

The contents of these registers can be changed (but should normally never be changed) in the LTA Verilog files “accel_prog_rom_gen.v”, and “accelrevc_prog_rom_gen.v”. These files, when run with Modelsim, generate ROM .mif and .rif files, which the LTA FPGA includes in the “altera_accel_rom256x2.v” and “altera_accelrevc_rom256x2.v” ROM files. These ROM files are then included in the LTA FPGA design, and logic in the “lta_mcbif.v” file actually does the programming under CPU control.

Refer to the “ACCEL Regulator VR2001 Program/Test Guide, Rev. 0.5, 4/30/2007” for further detailed information on Accel regulator programming.

10 Index

D

E

F

G

I

J

L

M

N

O

W

X

Y