UltraSPARC™-II Data Buffer (UDB-II)

DATA SHEET Companion Device for 250/300 MHz UltraSPARC-II Systems

DESCRIPTION

The UltraSPARC-II Data Buffer (UDB-II) consists of two identical ASICs connecting the UltraSPARC-II microprocessor and its E-Cache to the system data bus (i.e., UPA bus). These two are designated UDB_H (for the most-significant data bits) and UDB_L (for the least-significant data bits). The UDB-II moves data between the E-Cache bus and the UPA data bus (see *Figure 1*), enabling the system to send data to and receive data from the faster CPU transparently. The E-Cache bus has 128 data bits and 16 parity bits, for a total width of 144 bits. The data bus is also 144 bits wide, including 128 data bits and 16 error-correction code (ECC) bits. Noncacheable loads and stores use the E-Cache bus for moving data to the data buffer located in UDB_H and UDB_L.

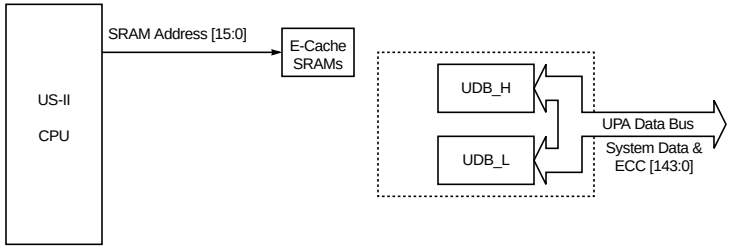
The UDB-II (model STP1081ABGA) is built with leading-edge CMOS ASIC technology featuring 0.5-micron geometries, and a three-layer metal process. It operates at 2.6 V and 3.3 V.

Features

- Isolates the processor from the system bus
- Interface to the UPA
- Operates at system frequency
- Operates at half the processor frequency
- Provides data buffering
- Supports parity on the CPU side and ECC on the system side
- Stores interrupt vectors. Interrupt registers visible to software
- IEEE standard 1149.1 (JTAG) boundary scan test
- 256 PBGA package

Benefits

- High performance: ease of design
- Fully synchronous design
- Enables secondary cache transfers and system memory transfers by isolation
- Ease of manufacturing test
- Small footprint and cost effective packaging



UDB-II has intelligent storage for two 64-byte cache lines (writeback) and eight 16-byte stores. In addition, it has storage for three 64-byte cache lines (read / load). The UDB-II delivers whatever 16-byte piece of data is requested first. It also compensates for any discrepancy between the delivery rates of the E-Cache and that of the interconnect.

UltraSPARC-II has a second-level cache of at minimum 512 Kilobytes. This second-level cache is physically indexed, physically tagged. The system address goes to the system controller and does not pass through UDB-II.

Figure 2 shows the logical operation of the UDB-II.

Figure 3 shows the data registers of the UDB-II in finer detail.

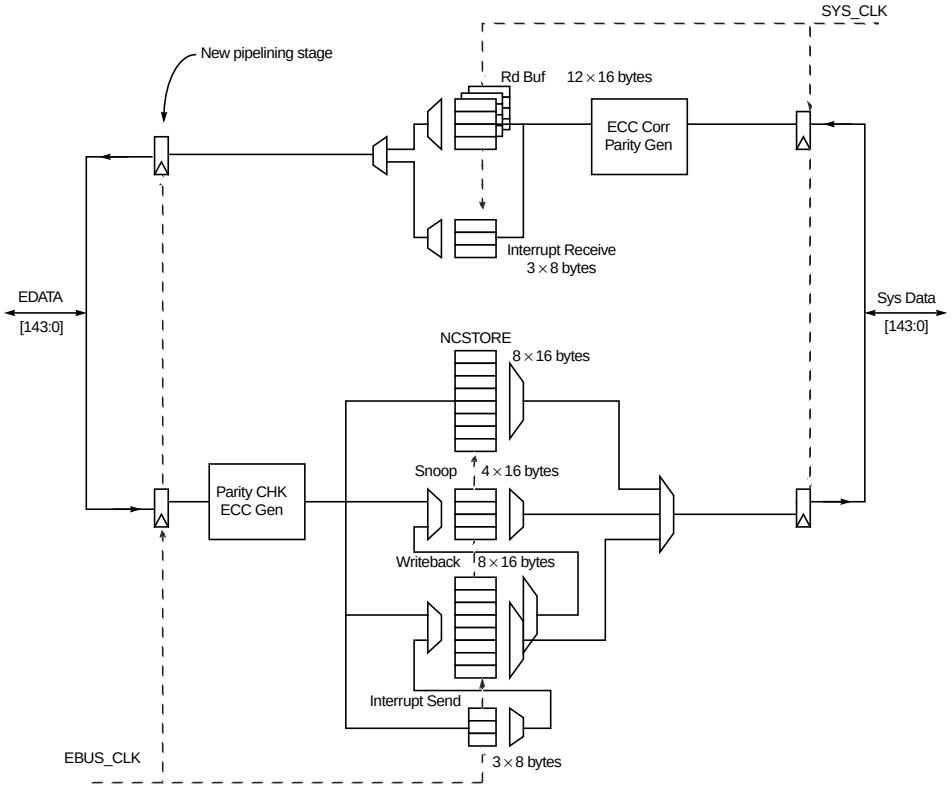


Figure 3. UDB-II Block Diagram

TECHNICAL OVERVIEW

Data Buffering

UltraSPARC-II supports multiple outstanding requests to the system. The list below gives details:

- Read requests are supported for load / store misses in the E-Cache (P_RDS / P_RDO P_REQ packets on the UPA):
  - up to three outstanding read requests on the UPA with only one outstanding instruction miss (P_RDSA)
  - two writeback requests on the UPA
- In addition, US-II supports the following requests:
  - two block loads (P_RDD / P_NCBRD)
  - only one outstanding noncacheable load

UDB-II supports data buffering for up to eight outstanding noncacheable 16-byte stores.

The UDB-II provides three 64-byte read buffers to support the three possible outstanding miss requests.

The UDB-II has two clock domains, one using the UPA clock, the other using a fixed 2:1 ratio CPU clock. (For example: 4 ns US-II CPU clock and 8 ns UDB-II EBUS_CLK.) The ECU and SRAMs communicate with the UDB-II on a fixed 2:1 bus, independent of UPA clock ratio.

64-Byte Buffers

64-byte buffers are provided for reads from the data bus, and for snoops and writebacks from the E-Cache bus. UDB-II has buffer support for two outstanding writebacks. The writeback buffer is snoopable.

16-Byte Buffers

The UDB-II can hold data for eight noncacheable stores in progress. This store data comes from the store buffer of the UltraSPARC-II.

The processor makes successive stores to the same 16-byte piece of data to be merged into one 128-bit store. The MMU E-bit inhibits store merging for pages that cannot tolerate merge due to size or order sensitivity. Consequently, noncacheable stores may have 128 bits of data, and 16-byte write enables.

8-Byte Buffers

Two mondo vectors, each consisting of three 64-bit packets, are also buffered in the UDB-II. One buffer holds data going from the UltraSPARC-II to the UPA, while the other holds data going from the UPA to the UltraSPARC.

Sub-Block Ordering

The UDB-II delivers and receives 16-byte sub-blocks in desired-word-first order. Subsequent blocks are delivered in order, wrapping to the beginning of the block, if necessary. UDB-II does no data re-ordering.

All 64-byte write-backs from UltraSPARC-II are delivered starting with sub-block zero. Snoops also return data with the desired sub-block first. The UDB-II has the ability to supply pending copyback data in the correct order for a snoop.

Error Correction Code (ECC) and Parity

The UDB-II supports error detection and correction on the system side using an ECC of 8 bits per 64 data bits. It corrects single errors and detects double errors and nibble errors. As implemented in UDB-II, the code has separate trees for ECC checking and correcting. Correctable errors are fixed in the same cycle during which they are detected. The UDB-II will log the syndrome for each corrected error (to help in diagnosing multi-bit errors).

There are some CSR read / write registers for enabling or disabling ECC. These registers are accessible through special ASI load or store activity.

There are two trap modes for UltraSPARC-II due to errors:

- Trap on any ECC error
- Trap on non-correctable ECC error

No automatic updating of memory occurs when a single-bit ECC error is corrected. Software may update memory with corrected data, maintaining multiprocessor cache coherency, by using cache flushes and compare-and-swap.

Data is protected on the UltraSPARC-II side using parity (odd parity), and on the UPA side using ECC. Parity is generated for data going from the UDB-II to the UltraSPARC-II, while syndrome bits are generated for data going from the UDB-II to the UPA. Data coming to the UDB-II from the UltraSPARC™-II is checked for parity errors, while data coming from the UPA to the UDB-II is checked for ECC.

ECC Code Used

ECC is performed on a 64-bit boundary, using Kaneda SEC / DED / S4ED codes. There are eight check bits per 64-bit boundary. The code provides detection of single- and double-bit errors, as well as three- and four-bit errors within a nibble. In addition, the code provides correction of any single bit error on 64-bit data.

ECC Control and Status Registers

Memory is not updated after an error is corrected. Parity errors are detected by the XOR of the eight P_syndrome bits. During ECC checking, the syndrome for the first correctable error is logged, and is only overwritten by the syndrome for an uncorrectable error. When there is an uncorrectable error, bad parity is forced on to data going out to the UltraSPARC-II. In diagnostic mode, a check bit vector of ECC bits is forced for data going from UDB-II to UPA.

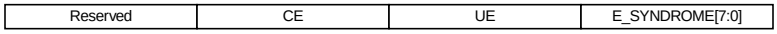


Figure 4. ECC and Parity Fault Status Register (Duplicated in UDB_H and UDB_L)

TABLE 1: ECC Status Definition [1]

Field	Description	Type
CE	Correctable Error	W1C
UE	Uncorrectable Error	W1C
E_Syndrome	ECC syndrome bits, 64 bits -> 8 bits	R

1. This register is cleared to 0 at power-on reset; W1C implies write-1-to-clear.

Figure 5. ECC Control and Diagnostic Register (Duplicated in UDB_H and UDB_L)

TABLE 2: ECC Control and Diagnostic Register Definition

Field	Description	Type
Version	Software-accessible version number, (TAP_IDreg[31:28])	R
FMODE	Force check bit vector for data to UPA	RW
FCBV	Forced check bit vector	RW

Bad parity errors are not forced by the UDB-II; UltraSPARC-II writes the SRAMs with bad parity and the UDB-II detects it. Bad ECC errors are forced in using the check bit vector FCBV[7:0] in each UDB microchip.

Operation of Error Detection and Correction

Parity Errors

If an E-Cache parity error occurs during a snoop, a known bad ECC data / checkbit pattern is forced. This causes an uncorrectable error (UE) trap at the master that requested the data. The slave processor will log error information (physical address and so on). This error log can be read by the master after trapping. The slave is not interrupted.

If the E-Cache was being read for a writeback (victimized line), bad ECC is also generated before the data goes to memory. In this case, the bad ECC guarantees no other processor will use the bad data. The error information is logged in the same register, and the processor traps. Software can clean up the bad ECC in memory in the trap handler.

ECC Errors

The UDB-II will not stall the data delivery to UltraSPARC-II on a correctable or uncorrectable error. Uncorrectable errors have bad parity forced, before the data is transferred to the E-Cache. This prevents the bad data from being used or written to memory with correct check bits later on. In either case, the error log will save the appropriate state. Separate CE and UE enables control whether the processor is trapped or not.

If the error occurs in a sub-block subsequent to one referenced by the processor, the trap will still occur.

If an ECC error is detected, bad parity is generated for the outgoing data either to UltraSPARC-II or to the SRAM, or both.

When an ECC error occurs at the same time as the UDB-II status register is being written by software, the syndrome is updated according to the UE and CE bits with the software clearing factored in. This is consistent with the conceptual model of software update being performed before the error is logged.

Assertion of ECC_VALID (same cycle as S_REPLY) indicates that the ECC bits accompanying the data being driven are valid. Error detection and correction are disabled for incoming data when ECC_VALID is deasserted.

Interrupts

The Data Path has storage for receiving mondo vectors. They reside in the UDB-II until Ultra-SPARC-II retrieves them. UDB-II has the "real" mondo-vector registers that are visible to software.

Mondo-Vector Buffers

Interrupts are delivered as write transactions from the system controller to UltraSPARC-II. The interrupt is signalled by a "mondo vector," which is a group of three 64-bit packets. I/O devices interrupt by sending mondo vector packets to a selected processor. The packet typically consists of one 64-bit PC, one 64-bit data structure pointer, and one 64-bit piece of data.

TABLE 3: Mondo Vector Format

Component	Mondo_0 (Cycle 1)	Mondo_1 (Cycle 2)	Mondo_2 (Cycle 3)	Mondo_3 (Cycle 4)
UDB_H	Data0	Data1	Data2	-
UDB_L	-	-	-	-

UltraSPARC-II receives the mondo-vector. The processor can post interrupts to itself at any level by writing to the SOFTINT register. SOFTINT register interrupts should not originate externally on the UPA, since the processor will not be able to recognize or respond correctly to them.

Parity and ECC errors on reads are reported back to the LSU, which logs the error status and traps the processor. Errors on writes are reported asynchronously by slaves using mondo vectors.

Mondo-Send Unit

UltraSPARC-II can also send mondo vectors. This begins by ASI stores loading data into a special UDB-II buffer. A final ASI store initiates the mondo send. A status register in UltraSPARC-II can be read with an ASI load to interrogate the completion status of such a mondo send.

Control and Data Transfers

The UDB-II has 64-byte buffers reserved for each request class except master writes. The noncacheable store queue has eight (16-byte) entries to handle multiple noncacheable stores to the frame buffer. Data transfers occur at some arbitrary time after address requests are made. Masters are responsible for buffering all data associated with queued requests.

Mondo vector data delivery is handled as a normal system controller write.

The E-Cache bus is controlled by the UltraSPARC-II ECU. Besides E-Cache reads and writes, there are transfers to and from the UDB-II. CNTL[4:0] provides the status of E-Cache bus to UDB-II.

TABLE 4: CNTL[4:0] Encodings

CNTL	Mnemonic	Action
00000	Noop	-
00001	rd_nodeid	Read nodeid (during reset)
00010	rd_ctrl	Read UDB control register (UDB_H and UDB_L)
00011	rd_stat	Read UDB status register (UDB_H and UDB_L)
00100	rd_mondor0	Read mondo vector[0] (PC)
00101	rd_mondor1	Read mondo vector[1] (pointer)
00110	rd_mondor2	Read mondo vector[2] (data)
00111	rd_readbuf	Read read buffer
01000	mv_wrback0	Move writeback buffer[0] to snoop buffer
01001	mv_wrback1	Move writeback buffer[1] to snoop buffer
01010	mv_wrback2	Move writeback buffer[2] to snoop buffer
01011	mv_wrback3	Move writeback buffer[3] to snoop buffer
01100	mv_wrback4	Move writeback buffer[4] to snoop buffer
01101	mv_wrback5	Move writeback buffer[5] to snoop buffer
01110	mv_wrback6	Move writeback buffer[6] to snoop buffer
01111	mv_wrback7	Move writeback buffer[7] to snoop buffer
10000	wr_ctrl_l	Write UDB_L control register (LS)
10001	wr_ctrl_h	Write UDB_H control register (MS)
10010	wr_stat_l	Write UDB_L status register (LS)
10011	wr_stat_h	Write UDB_H status register (MS)
10100	wr_mondos0	Write mondo vector [0] (by CPU)
10101	wr_mondos1	Write mondo vector [1] (by CPU)
10110	wr_mondos2	Write mondo vector [2] (by CPU)
11000	wr_snoop	Write snoop buffer
11001	wr_ncst	Write noncacheable store buffer
11010	wr_wrback	Write writeback buffer
11100	mv_mondos0	Move mondo vector[0] to writeback buffer
11101	mv_mondos1	Move mondo vector[1] to writeback buffer
11110	mv_mondos2	Move mondo vector[2] to writeback buffer
11111	mv_mondos3	Move mondo vector[3](fake) to wrback buffer

The CE pin of the UDB-II indicates the detection and correction of a single-bit ECC error. The UE pin of the UDB-II indicates the presence of uncorrectable ECC errors. If asserted during an E-Cache miss, the new line must not be marked valid in the E-Cache or D-cache. ECC on the full data bus is checked / generated for all system transactions.

Transactions on the data bus are controlled by the reply bus between the system controller and the UltraSPARC-II.

TABLE 5: S_REPLY[3:0] Encodings [1]

S_REPLY	Mnemonic	Action
0000	S_IDLE	Default
0001*	S_ERR	Bus error
0010	S_SACK	Coherence read block: UDB => snoop buffer (64B)
0011	S_WBCAN	Writeback cancel: increment wrback buffer readctr by 4
0100	S_WAS	Write ack single: UDB => non-cacheable store
0101	S_WAB	Write ack block: UDB => wrback/block_store/mondo_send
0110*	S_OAK	Ownership ack block
0111*	S_INAK	Interrupt nack
1000	S_RBU	Read block unshared: UDB <= read fill (64B)
1001	S_RBS	Read block shared: UDB <= read fill (64B)
1010	S_RAS	Read ack single: read fill (16B)
1011*	S_RTO	Read time-out
1101	S_SWIB	Slave write interrupt block: UDB <= mondo receive
1110	S_SRS	Slave read single: UDB => read_portid
1111*	S_SRB	Slave read block

1. Opcodes with * imply the UDB-II takes no action.

The DATA_STALL signal from the UPA allows the system controller to control the flow of data to or from that UPA port. It is typically used to allow memory several cycles of setup on the UPA bus, or to ask the UPA port to hold data longer, so that it can be held at memory longer.

Clocking

The UDB-II has two frequency domains, which can be denoted “EBUS_CLK” (for external cache bus clock) and “SYSCLK” (for system clock). (*Figure 3* and *Figure 6* show more detail.) EBUS_CLK is for data transfers between US-II and UDB-II. SYS_CLK is for data transfers between UPA and UDB-II. There is no overlap between the two domains, meaning that no data is clocked by both clocks at the same time.

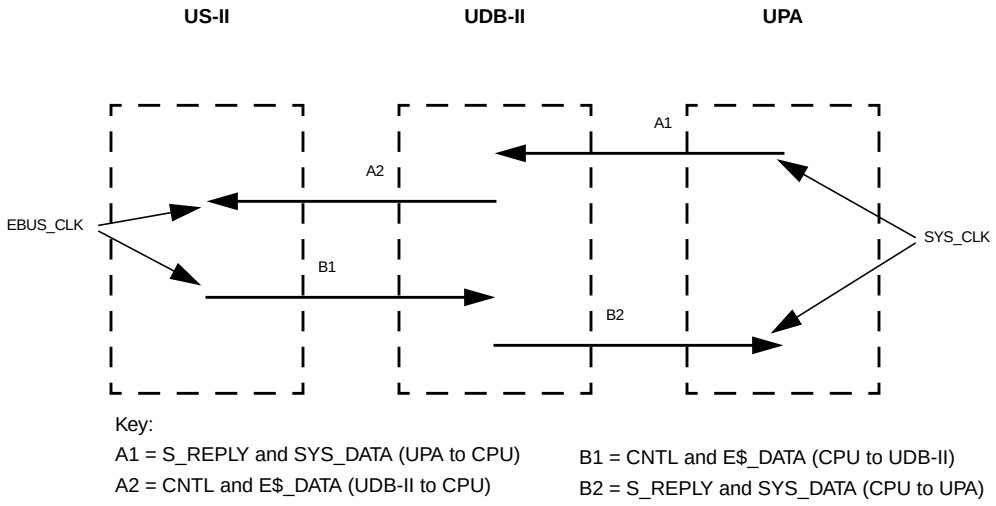


Figure 6. Relationship of SYS_CLK to EBUS_CLK

The table below shows the interconnect timing between the clocks.

TABLE 6. Clock Interconnect

Clock Mode	EBUS_CLK	SYS_CLK	Unit
1:1	10	10	ns
2:3	7	10.5	ns
1:2	7	14	ns

Testability

The UDB-II provides five signals for supporting the IEEE 1149.1 standard JTAG serial scan interface. This interface is used for fault coverage testing, boundary, and interconnect testing.

SIGNAL DESCRIPTIONS

Symbol	Type	Name and Function
EDATA[63:0]	I/O	The UDB-II transfers data over this bus with both the E-Cache SRAMs and the UltraSPARC-II. On E-Cache misses, this bus is an output, and supplies data to the E-Cache RAMs from one of its buffers. On E-Cache write backs, data is transferred from the E-Cache RAMs on this bus into one of the UDB-II microchip buffers. Each of the two UDB-II microchips handle half the width of the 128-bit data bus from the UltraSPARC-II and E-Cache. Noncacheable loads and stores transfer data directly between the Ultra-SPARC-II and the UDB-II data buffers on this bus. For enabling and disabling ECC, the UDB-II also supports ASI loads and stores from the Ultra-SPARC-II over this bus. This allows writing or reading of certain control and status registers on the UDB-II for ECC purposes.
EDPAR[7:0]	I/O	This signal provides eight parity bits for E-Cache data.
SYSDATA[63:0]	I/O	The UDB-II transfers data to the system interconnect over this bus. Data is transferred at the system interconnect clock rate.
SYSECC[7:0]	I/O	For each write by UltraSPARC-II to the system interconnect, UDB-II generates eight ECC check bits for each 64-bit piece of data being transferred. The SYSECC bus is then configured as an output. On input, good ECC check bits generated by slaves are transferred to the UDB-II over this bus, and checked inside the UDB-II.
CE	O	This pin is driven by the UDB-II to tell the UltraSPARC-II that it has detected a correctable ECC error on the data that it received from the interconnect (that is, a single-bit error). The delivery of data to the UltraSPARC-II is stalled until that error is corrected.
UE	O	This pin is driven by the UDB-II to tell the UltraSPARC-II that it has detected an uncorrectable ECC error on the data that it received from the interconnect.
S_REPLY[3:0]	I	These pins are connected to a unidirectional four-bit bus that receives encoded acknowledgments from the system controller in response to an address transaction sent out by UltraSPARC-II.
SYSID[4:0]	I	This bus is used to convey a five-bit system node ID to the UDB-II from the system interconnect.
SYSCLK[1:0]	I	This is an input, and has the same frequency as the UPA clock. It is a differential, PECL clock.
EBUS_CLK[1:0]	I	This is an input from the clock controller, and has the same frequency as the clock for the processor E-Cache bus. It is a differential, PECL clock pair.
EXT_EVENT	I	This is an input signal used to indicate the clock should be stopped. It is a debug signal which is set inactive on production systems. This signal powers down the chip when asserted.
PLL_BYPASS	I	When asserted, this pin causes the phase-lock loop to be bypassed. In that case, the clock from the differential receiver is passed directly to the clock trunk.
RESET_L	I	This signal is asserted when an external reset request occurs.
CNTL[4:0]	I	UltraSPARC-II uses these signals to request the UDB-II to either load its buffers from the external cache data bus or to drive the content of its buffers on to this bus.
UDB_H	I	This pin is hardwired to V_{DD} for UDB_H (the UDB-II microchip for the most significant 72 bits) and to GND for UDB_L (the UDB-II microchip for the least significant 72 bits).
EPD	I	Indicates that the UltraSPARC–II is in power-down mode.
TDI	I	IEEE standard 1149 test data input. This pin is internally pulled to logic 1 when not driven.

SIGNAL DESCRIPTIONS (CONTINUED)

Symbol	Type	Name and Function
TDO	O	JTAG Interface. IEEE standard 1149 test data output. A three-state signal driven only when that TAP controller is in the shift-DR state.
TMS	I	IEEE standard 1149 test mode select input. This pin is internally pulled to logic 1 when not driven.
TCK	I	IEEE standard 1149 test clock input. If not hooked to a clock source, this pin must always be driven to a logic 1 or a logic 0.
TRST_L	I	IEEE standard 1149 test reset input (active low). This pin is internally pulled to logic 1 when not driven.
DATA_STALL	I	Input from the UPA bus to indicate the data is delayed.
ECC_VALID	I	Input from the UPA to indicate when to ignore ECC of SYSDATA coming in from the UPA bus. When this value is one, UDB-II generates ECC from SYSDATA and compares it with incoming ECC. If there is a mismatch, UDB-II attempts to correct the data, it then updates CE and UE (see the explanations earlier in this table). When this signal is zero, the data from the UPA bus is not checked for errors.
LF1A	I	SYSCLK PLL loop filter input of charge pump.
LF2A	I	SYSCLK PLL loop filter input pin to VCO.
LF3A	I	SYSCLK Loop filter ground.
LF1B	I	EBUS_CLK PLL loop filter input of charge pump.
LF2B	I	EBUS_CLK PLL loop filter input pin to VCO.
LF3B	I	EBUS_CLK Loop filter ground.
SYS_CLKOUT	O	Output of the internal (SYSCLK) clock for characterization of clock in PLL or BYPASS mode.
EBUS_CLKOUT	O	Output of the internal (EBUS_CLK) clock for characterization of clock in PLL or BYPASS mode.

TIMING CONSIDERATIONS

General

The UDB-II will not stall data delivery to UltraSPARC-II on a correctable error or an uncorrectable error.

The number of transaction cycles on the data bus are as follows:

- One for noncacheable stores (128 bits)
- Four for mondo vectors (192 bits)
- Four for cache lines (512 bits)

Timing Diagrams

Four Timing Modes

The clock relationships between the various combinations and timing possibilities are shown in *Figure 7*.

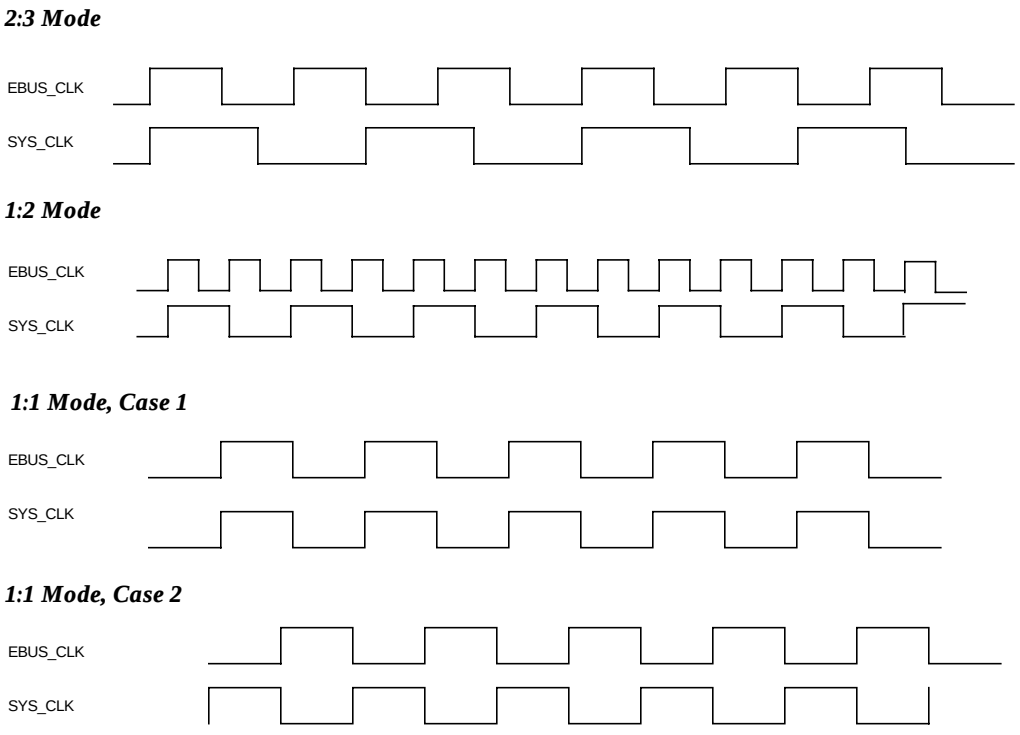


Figure 7. Timing Comparisons between EBUS_CLK and SYS_CLK Cases

Data Transfer Timing Diagrams

The timing diagrams which follow show data transfers from the UPA to the UltraSPARC-II and from the UltraSPARC-II to the UPA. Arrows indicate when data is registered at the UDB-II except when otherwise noted. These diagrams portray signal timing for minimum latency.

Two-to-Three Clock Ratio for Data Transfer

Figure 8 shows the most common data transfer case. For UPA-to-CPU data transfer, the operation begins with S_REPLY, which is clocked at the rising edge of SYS_CLK, corresponding to time t_0. At time t_1, the next rising edge of SYS_CLK, the data is latched into UDB-II. In other words (given that there is no data stall), UDB-II will take what is on the UPA bus on the next SYS_CLK rising edge following S_REPLY.

The processor requests the data in UDB-II by asserting CNTL. Note that one whole EBUS_CLK cycle must complete after SYS_CLK rise where the data was latched, before CNTL can be asserted. Once this cycle completes, CNTL is given at the rising edge of EBUS_CLK.

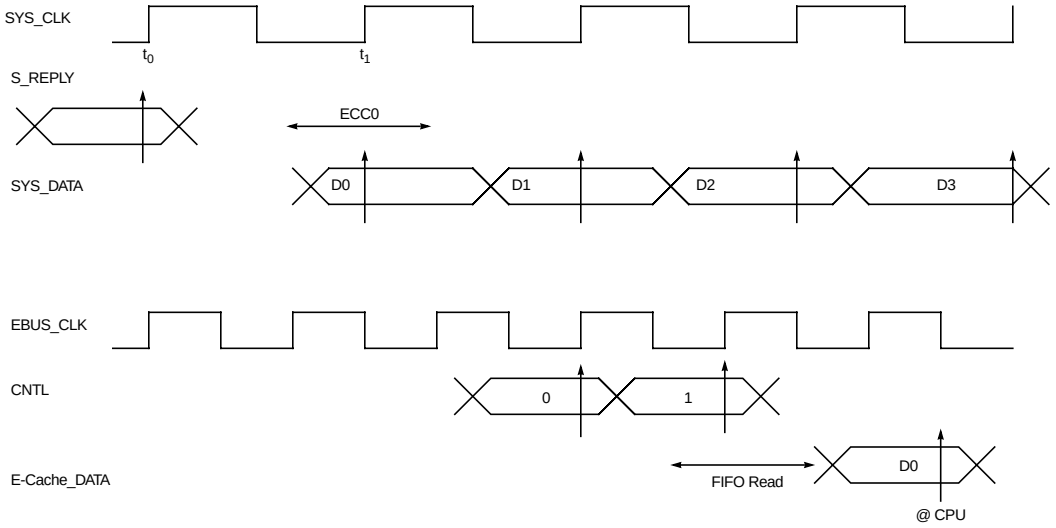
Two cycles later, the requested data packet appears at the output of UDB-II (the same as the microprocessor input.)

For CPU-to-UPA data transfer, the processor sends CNTL and E$_DATA at the same time. When UPA requests the data, S_REPLY may be asserted as little as two cycles after data is latched, and that data appears at the output of UDB-II two cycles later.

Note the following differences:

The S_REPLY is only given once for every transaction, one cycle before the data is latched into UDB-II. However, CNTL is given for every E$_DATA and during the same cycle as E$_DATA.

UPA-to-CPU Data Transfer



CPU-to-UPA Data Transfer

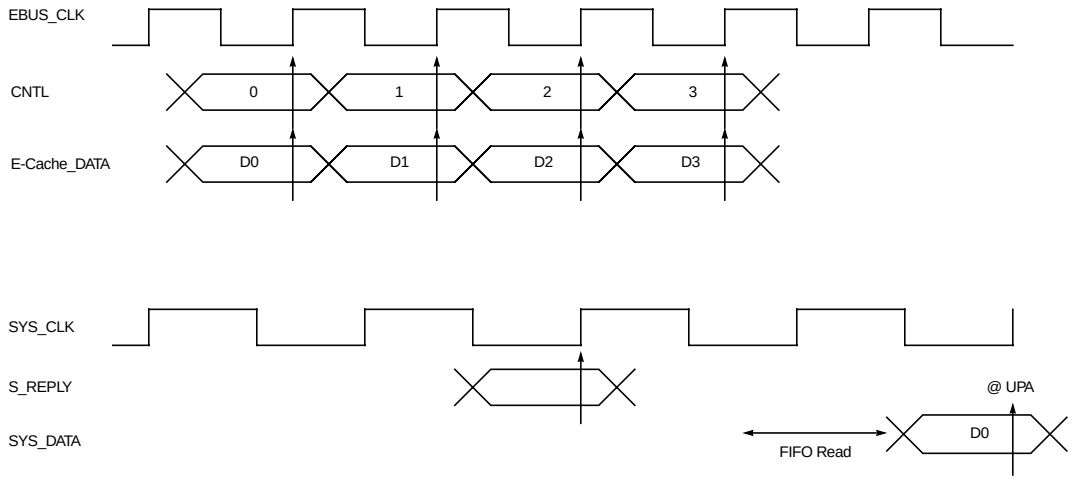


Figure 8. UPA-to-CPU and CPU-to-UPA Data Transfer Timing Diagram 2:3

One-to-Two Clock Ratio for Data Transfer

Figure 9 shows the most common data transfer case. For UPA-to-CPU data transfer, the operation begins with S_REPLY, which is clocked at the rising edge of SYS_CLK, corresponding to time t_0. At time t_1, the next rising edge of SYS_CLK, the data is latched into UDB-II. In other words, (given that there is no data stall) UDB-II will take what is on the UPA bus on the next SYS_CLK rising edge following S_REPLY.

The processor requests the data in UDB-II by asserting CNTL. Note that one whole EBUS_CLK cycle must complete after SYS_CLK rise where the data was latched, before CNTL can be asserted. Once this cycle completes, CNTL is given at the rising edge of EBUS_CLK.

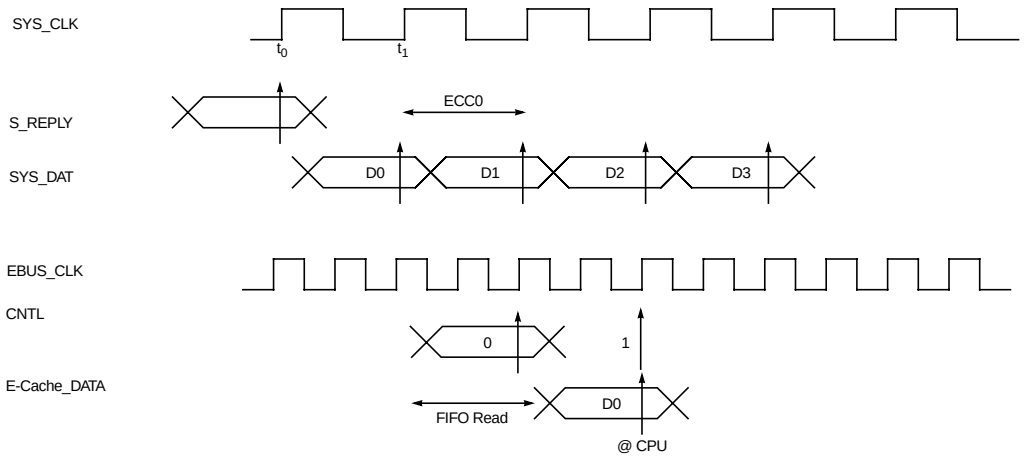
Two cycles later, the requested data packet appears at the output of UDB-II (the same as the microprocessor input.)

For CPU-to-UPA data transfer, the processor sends CNTL and E$_DATA at the same time. When UPA requests the data, S_REPLY may be asserted as little as two cycles after data is latched, and that data appears at the output of UDB-II two cycles later.

Note the following differences:

The S_REPLY is only given once for every transaction, one cycle before the data is latched into UDB-II. However, CNTL is given for every E$_DATA and during the same cycle as E$_DATA.

UPA-to-CPU Data Transfer



CPU-to-UPA Data Transfer

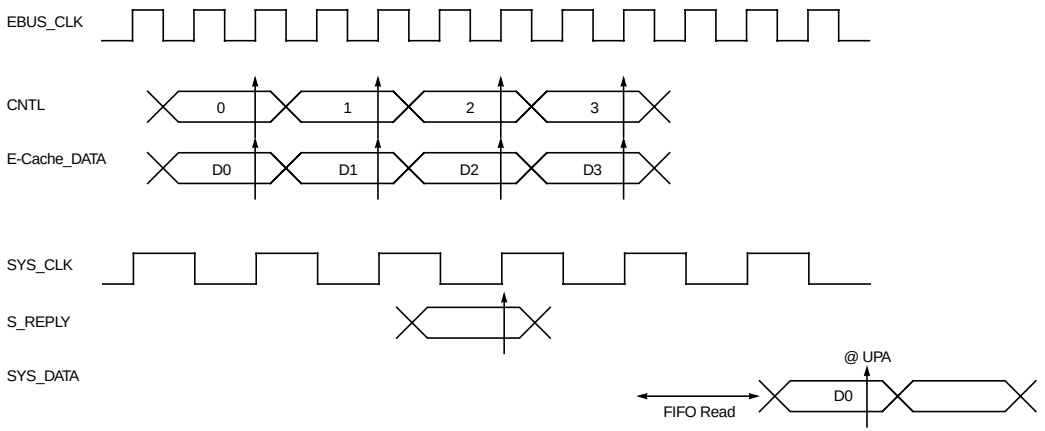


Figure 9. UPA-to-CPU and CPU-to-UPA Data Transfer Timing Diagram 1:2

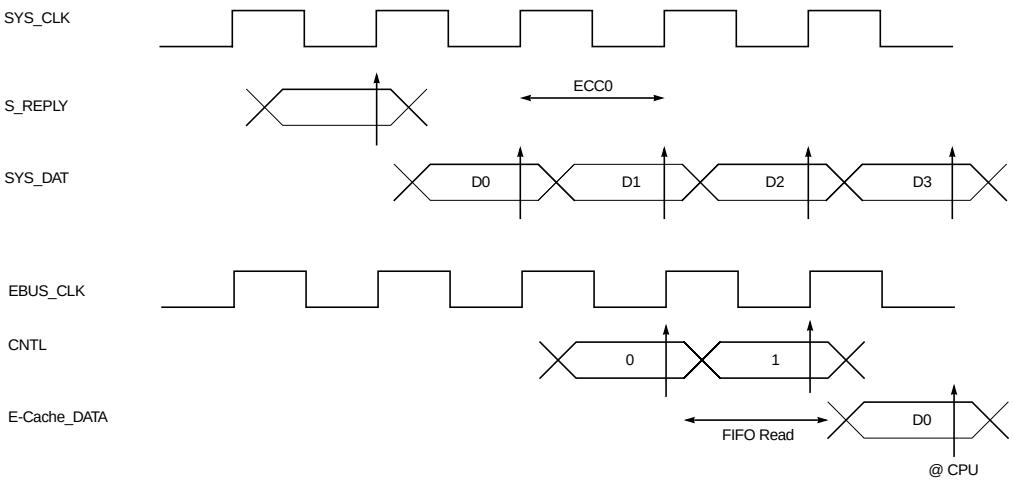
One-to-One Clock Ratio for Data Transfer, Case 1

When S_REPLY is asserted, data will transfer from UPA to UDB-II for up to four cycles, as specified in the S_REPLY coding.

Once the UPA data is latched into UDB-II, the UltraSPARC-II asserts CNTL and two cycles later, data appears at the output of UDB-II.

In cases of data stall, the system receives S_REPLY but SC_DATA_STALL is held high until data is complete. Then, SC_DATA_STALL goes low and one cycle later, data from the UPA bus is transferred as described above.

UPA-to-CPU Data Transfer



CPU-to-UPA Data Transfer

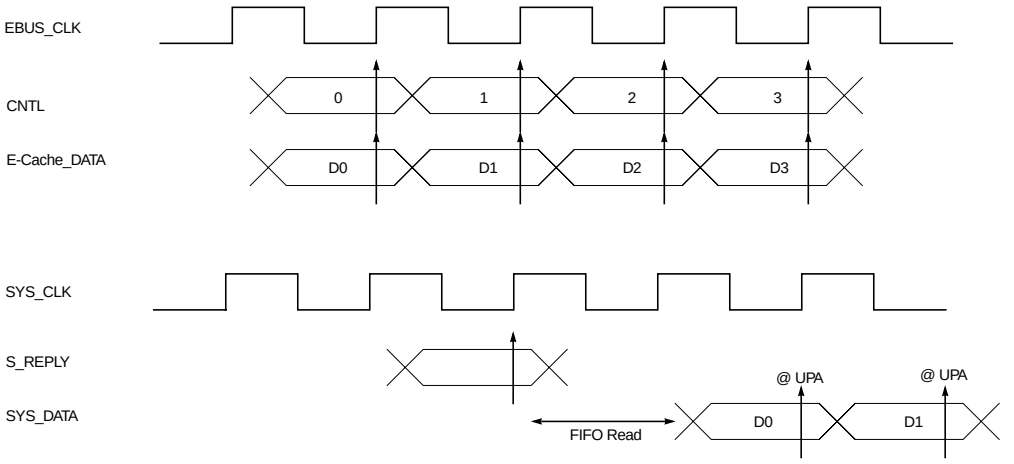
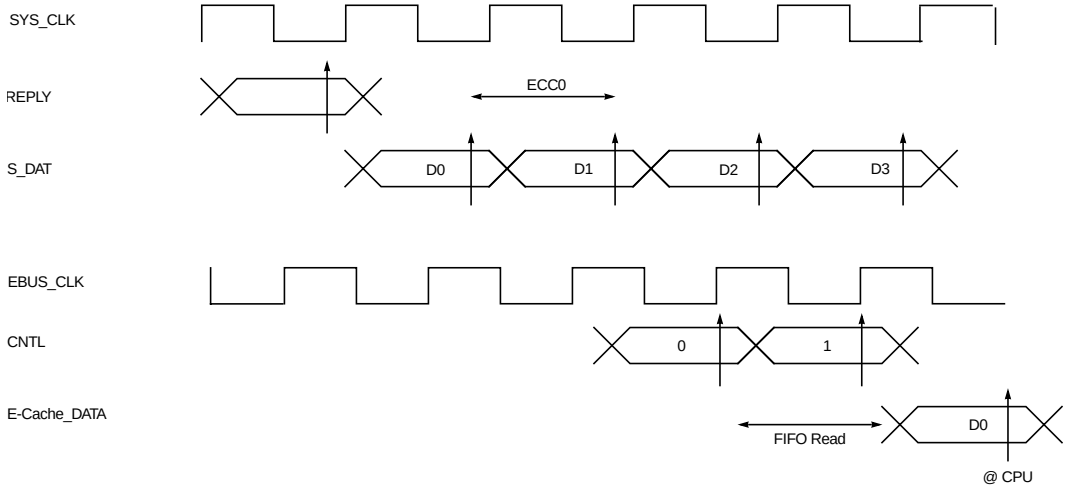


Figure 10. UPA-to-CPU and CPU-to-UPA Data Transfer Timing Diagrams 1:1 Case 1

UPA-to-CPU Data Transfer



CPU-to-UPA Data Transfer

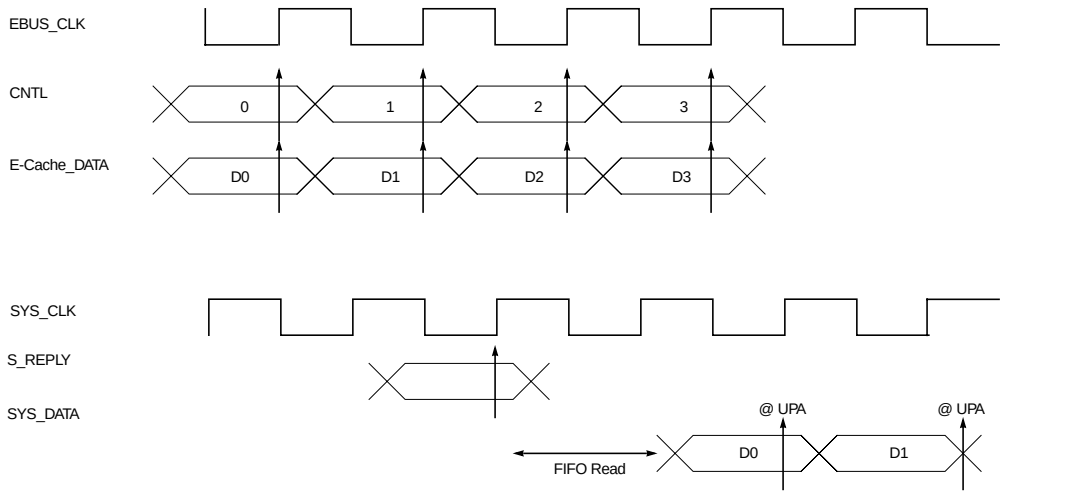


Figure 11. UPA-to-CPU and CPU-to-UPA Data Transfer Timing Diagram 1:1 Case 2

External Power Down (EPD)

The power-down mode is intended to support Energy Star compliance. The EPD pin initiates an internal reset, which stops the clock and PLL, and the UDB-II enters power-down mode. Exiting power-down mode is similar to normal power-up sequence.

ELECTRICAL SPECIFICATIONS

Recommended Operating Conditions

Symbol	Parameter	Min	Nom	Max	Unit
V_{CC1}	Core Supply Voltage	3.135	3.3	3.465	V
V_{CC2}	UPA I/O Supply Voltage	3.135	3.3	3.465	V
V_{CC3}	E-Cache I/O Supply Voltage	2.47	2.6	2.73	V
V_{CC_PECL}	PECL Supply Voltage	3.135	3.3	3.465	V
V_{CC_APLL}	APLL Supply Voltage	3.135	3.3	3.465	V
V_{IH-CLK}	SYS_CLK & EBUS_CLK high-level input voltage	2.275	-	2.420	V
V_{IL-CLK}	SYS_CLK & EBUS_CLK low-level input voltage	1.490	-	1.680	V
V_{IH}	UPA side high-level input voltage	2.0	-	-	V
V_{IL}	UPA side low-level input voltage	-	-	0.8	V
V_{IH}	E-Cache side high-level Input Voltage	1.65	-	-	V
V_{IL}	E-Cache side low-level input voltage	-	-	1.15	V
I_{OH}	UPA & E-Cache side high-level output current	-	-	12	mA
I_{OL}	UPA & E-Cache side low-level output current	-	-	12	mA
T_{OJ}	Operating junction temperature	0	-	105	°C

Absolute Maximum Rating

Symbol	Parameter	Rating	Unit
V_{CC1}	Core Supply Voltage	-0.5 to 4.0	V
V_{CC2}	UPA I/O Supply Voltage	-0.5 to 4.0	V
V_{CC3}	E-Cache I/O Supply Voltage	-0.5 to 3.0	V
V_{CC_PECL}	PECL Supply Voltage	-0.5 to 4.0	V
V_{CC_APLL}	APLL Supply Voltage	-0.5 to 4.0	V
V_{I-CLK}	SYS_CLK & EBUS_CLK input voltage	-0.5 to V_{CC2}+0.5	V
V_I	UPA side input voltage	-0.5 to V_{CC2}+0.5	V
V_O	UPA side output voltage	-0.5 to V_{CC2}+0.5	V
V_I	E-Cache side input voltage	-0.5 to V_{CC3}+0.5	V
V_O	E-Cache side output voltage	-0.5 to V_{CC3}+0.5	V
V_{I_PECL}	E-Cache side input voltage	VCC_3-2.0 to V_{CC3}+0.5	V
I_O	UPA & E-Cache side output current (Vo = 0V)	20	mA
I_{IK}	Input clamp current (VI<0 OR VI>VCC)	+/-20	mA
I_{OK}	Output clamp current (VO<0 OR VO>VCC)	+/-20	mA
T_{STG}	Storage temperature	-40 to 150	°C

DC Characteristics

Symbol	Parameter	Min	Max	Unit
V_{IH-CLK}	SYS_CLK & EBUS_CLK high-level input voltage	2.275	2.420	V
V_{IL-CLK}	SYS_CLK & EBUS_CLK low-level input voltage	1.490	1.680	V
V_{IH}	UPA side high-level input voltage	2.0	-	V
V_{IL}	UPA side low-level input voltage	-	0.8	V
V_{IH}	E-Cache side high-level Input Voltage	1.65	-	V
V_{IL}	E-Cache side low-level input voltage	-	1.15	V
V_{OH}	UPA side high-level output voltage (I_{OH} = max)	Vcc2 - 0.6	-	V
V_{OL}	UPA side low-level output voltage (I_{OL} = max)	-	0.4	V
V_{OH}	E-Cache side high-level output voltage (I_{OH} = max)	Vcc3 - 0.6	-	V
V_{OL}	E-Cache side low-level output voltage (I_{OL} = max)	-	0.4	V
Ioz	UPA side high impedance output current (V_O = 0 or 3.3V)	-	+/- 10	uA
Ioz	E-Cache side high impedance output current (V_O = 0 or 2.6V)	-	+/- 10	uA
I_{IH}	UPA side high-level input current (V_I = 3.3V)	-	1.0	uA
I_{IH}	E-Cache side high-level input current (V_I = 2.6V)	-	1.0	uA
I_{IL}	UPA & E-Cache side low-level input current (V_I = 0V)	-	-1.0	uA
C_{IN}	UPA & E-Cache side input capacitance (input only pins)[1]	-	2.62	pF
C_{IN}/C_{OUT}	UPA & E-Cache side input capacitance (IO pins)[1]	-	5.27	pF
C_{OUT}	UPA & E-Cache side output capacitance (output only pins)[1]	-	4.38	pF
P_D	Power Dissipation	-	4	W

1. Device and pad capacitance only. Package capacitance is not included.

PLL Specifications

Clock	Operating Frequency Range	Center Frequency	Clock Skew Max	Clock Skew Min
EBUS_CLK	62.5 MHz (16ns) to 125Mhz (8ns)	100 MHz (10ns)	400ps	-400ps
SYSCLK	83.3 MHz (12ns) to 166.6 Mhz (6ns)	143 MHz (7ns)	400ps	-400ps

Clock skew budget for PLL includes phase offset, jitter and clock trunk delay. External clock skew between EBUS_CLK and SYS_CLK is + / - 300ps.

All timing parameters are measured using the following test conditions:

Timing Specifications

Parameter	Condition
Temperature (junction)	0 - 105°C
V_{CC1}	3.3V +/- 5%
V_{CC2}	3.3V +/- 5%
V_{CC3}	2.6V +/- 5%
Input slew rate for setup time	1.25V/ns
Input slew rate for hold time	2V/ns
Process models used	Register to Register timing: slow slow process model Output propagation delay: slow slow process model Input setup time: slow slow process model Output hold time: fast fast process model Input hold time: fast fast process model
Clock skew budget	EBUS_CLK: 400ps SYS_CLK: 400ps
Tester Guardband included in the IO timing	500ps
PLL mode	Enabled
Load for output delay/hold measurements	Preliminary V-series tester load provided by TI. See Appendix 1 for the model SPICE deck. This model will be updated by TI at a later date and the IO specifications will be regenerated by SUN. TI has agreed to accept the updated IO specifications.

Register to Register Timing Specifications

Source register clock domain	End register clock domain	Timing
EBUS_CLK	EBUS_CLK	7 ns
SYS_CLK	EBUS_CLK	7 ns
SYS_CLK	SYS_CLK	10 ns
EBUS_CLK	SYS_CLK	10 ns

External clock skew between the EBUS_CLK and the SYS_CLK is +/- 300ps.

Clock skew budget for each clock domain is 400ps.

Input Setup and Hold Time Specifications[1]

Symbol	Parameter	Waveform	UDB-II for 150 MHz CPU		UDB-II for 300 MHz CPU		Unit
			Min	Max	Min	Max	
tsu	EDATA[63:0] setup time	1	-	1.59	-	1.35	ns
tsu	EDPAR[7:0] setup time	1	-	1.69	-	1.43	ns
tsu	CNTL[4:0] setup time	1	-	4.53	-	3.73	ns
tsu	SYSDATA[63:0] setup time	1	-	2.00	-	1.98	ns
tsu	SYSECC[7:0] setup time	1	-	2.00	-	1.98	ns
tsu	DATA_STALL setup time	1	-	1.41	-	1.58	ns
tsu	ECC_VALID setup time	1	-	1.48	-	1.38	ns
tsu	S_REPLY[3:0] setup time	1	-	1.48	-	1.38	ns
tsu	SYS_ID [4:0] setup time	1	-	4.78	-	4.28	ns
tsu	RESET_L setup time	1	-	-[2]	-	-[2]	ns
tsu	EPD setup time	1	-	-1.[2]	-	-1.[2]	ns
tsu	EXT_EVENT setup time	1	-[2]	-	-[2]	-	ns
th	EDATA[63:0] hold time	1	1.27	-	1.30	-	ns
th	EDPAR[7:0] hold time	1	1.27	-	1.30	-	ns
th	CNTL[4:0]) hold time	1	0.18	-	0.18	-	ns
th	SYSDATA[63:0] hold time	1	1.04	-	0.95	-	ns
th	SYSECC[7:0] hold time	1	0.94	-	0.85	-	ns
th	DATA_STALL hold time	1	0.99	-	0.67	-	ns
th	ECC_VALID hold time	1	1.11	-	1.05	-	ns
th	S_REPLY[3:0] hold time	1	0.91	-	0.75	-	ns
th	SYS_ID[4:0] hold time	1	-0.20	-	-0.55	-	ns
th	RESET_L hold time	1	-[2]	-	-[2]	-	ns
th	EPD hold time	1	-[2]	-	-[2]	-	ns
th	EXT_EVENT hold time	1	-[2]	-	-[2]	-	ns

1. These timing numbers are determined using tester loads. The actual spice netlists should be used to determine actual timing in the system being designed.
2. Included for completeness only. These are initialization signals.

Output Propagation Delay and Hold Time Specifications

Symbol	Parameter	Waveform [1]	UDB-II for 150 MHz CPU		UDB-II for 300 MHz CPU		Unit
			Min	Max	Min	Max	
t_p	EDATA[63:0] clock to output delay	2	-	4.46	-	4.07	ns
t_p	EDPAR[0:7] clock to output delay	2	-	4.43	-	4.05	ns
t_p	UE & CE clock to output delay	2	-	4.46	-	4.06	ns
t_p	SYSDATA[63:0] clock to output delay	2	-	4.81	-	4.34	ns
t_p	SYSECC[7:0] clock to output delay	2	-	4.44	-	4.05	ns
t_p	EBUS_CLKOUT clock to output delay	2	-	- [2]	-	- [2]	ns
t_p	SYS_CLKOUT clock to output delay	2	-	- [2]	-	- [2]	ns
toh	EDATA[63:0] output hold time	2	1.48	-	1.28	-	ns
toh	EDPAR[7:0] output hold time	2	1.53	-	1.32	-	ns
toh	UE & CE output hold time	2	1.47	-	1.27	-	ns
toh	SYSDATA[63:0] output hold time	2	1.88	-	1.6	-	ns
toh	SYSECC[7:0] output hold time	2	1.88	-	1.26	-	ns
t_{oh}	EBUS_CLKOUT output hold time	2	- [2]	-	- [2]	-	ns
t_{oh}	SYS_CLKOUT output hold time	2	- [2]	-	- [2]	-	ns

1. See the Waveform Diagrams which follow (*Figure 12*, *Figure 13*, and *Figure 14*).
2. Included for completeness. These are initialization signals.

Load Board Power Supplies[1]

Power Supply	Bypass Capacitor and EMI Filter Information
Core Power Supply	8 uF
Output_2.6V Power Supply	10.8 uF
Output_3.3V	10.8 uF
DPECLs Power Supply	Jump Wire Connect to Power Supply of Core
First APLL	EMI Filter
Second APLL	EMI Filter

1. There is only one ground plane. See *Figure 15* for loop filter information.

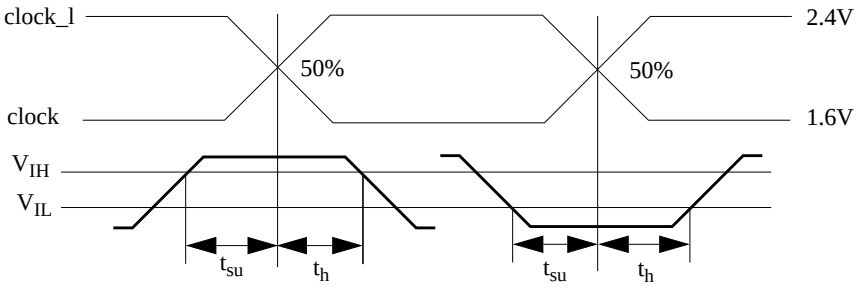


Figure 12. Waveform 1: Input, setup and hold time measurement points

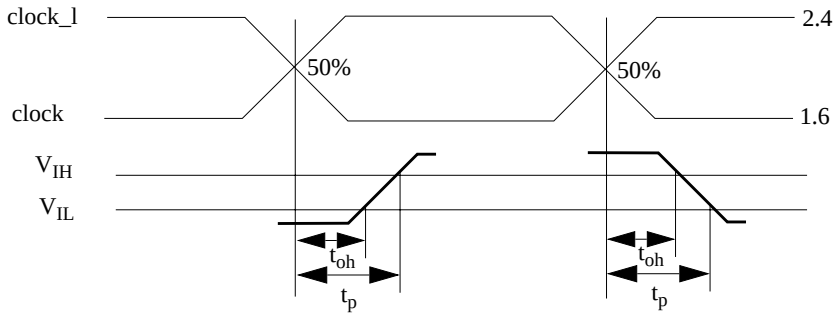


Figure 13. Waveform 2: Output propagation and hold time measurement points

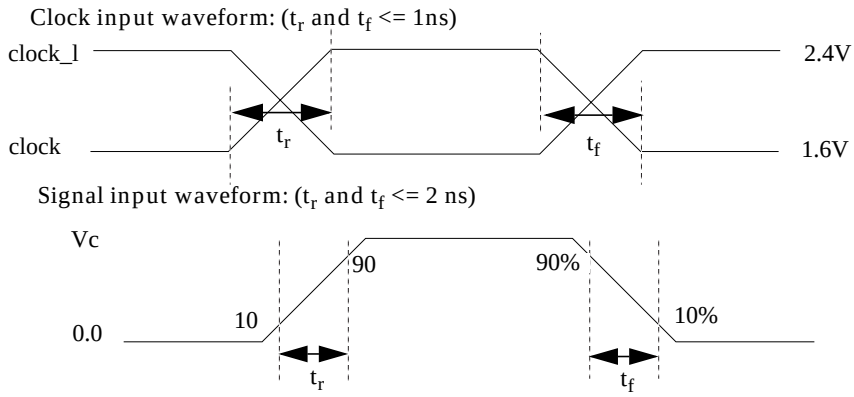


Figure 14. Waveform 3: Clock and Signal Input Rise Time and Fall Time Requirements

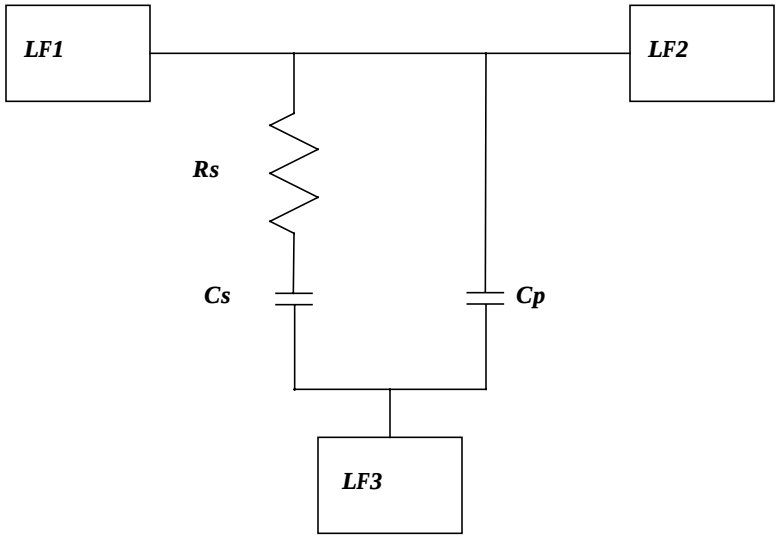


Figure 15. Loop Filter Information

E-Cache Bus Loop Filter (125 MHz)

Rs = 15 Ohms; Cs = 56 nF; Cp = 1.0 nF

System Loop Filter (83 MHz)

Rs = 15 Ohms; Cs = 68 nF; Cp = 1.5 nF

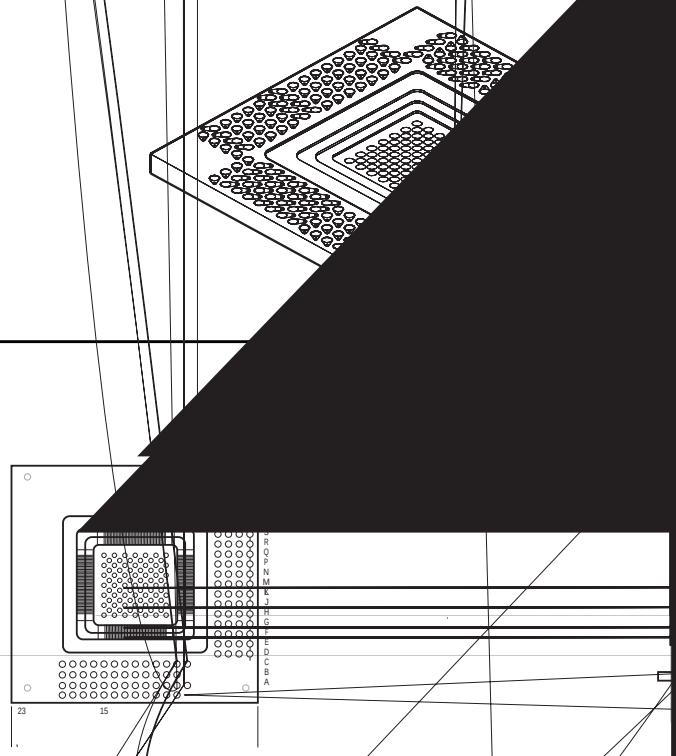
PIN ASSIGNMENTS

Pin	Pin Name	Pin	Pin Name	Pin	Pin Name	Pin	Pin Name
A4	EDATA[50]	D20	EDATA[45]	M20	ECC_VALID	V5	SYSDATA[24]
A5	EDATA[56]	D21	EDATA[44]	M21	GND	V6	SYSDATA[28]
A6	EDPAR[6]	D22	GND	M22	SYSECC[5]	V7	SYSDATA[30]
A7	EDATA[52]	D23	EDATA[43]	M23	SYSECC[7]	V8	DATA_STALL
A8	TDI	E1	EDATA[58]	N1	SYSDATA[12]	V9	VCC1
A9	EDATA[30]	E2	EDATA[62]	N2	SYSDATA[11]	V10	SYSID[0]
A10	TRST_L	E3	EDATA[61]	N3	GND	V11	SYS_CLK[0]
A11	VCC_APLL	E4	EDATA[60]	N4	VCC1	V12	GND_PECL
A12	EBUS_CLK[1]	E20	EDATA[40]	N20	SYSECC[6]	V13	VCC1
A13	EDATA[28]	E21	EDPAR[1]	N21	VCC2	V14	S_REPLY[3]
A14	EDPAR[2]	E22	VCC3	N22	SYSECC[3]	V15	SYSDATA[51]
A15	EDATA[20]	E23	EDATA[13]	N23	SYSECC[4]	V16	VCC1
A16	TCK	F1	EDATA[33]	P1	SYSDATA[9]	V17	GND
A17	EDATA[17]	F2	EDATA[63]	P2	VCC2	V18	SYS_CLKOUT
A18	EDATA[23]	F3	VCC3	P3	SYSDATA[13]	V19	SYSDATA[62]
A19	EDATA[47]	F4	EDPAR[7]	P4	SYSDATA[15]	V20	SYSDATA[33]
B4	EDATA[54]	F20	EDATA[12]	P20	SYSECC[2]	W5	VCC2
B5	EDPAR[3]	F21	EDATA[9]	P21	SYSDATA[7]	W6	GND
B6	EDATA[31]	F22	EDATA[8]	P22	SYSECC[1]	W7	RESET_L
B7	GND	F23	EDATA[42]	P23	SYSECC[0]	W8	EXT_EVENT
B8	VCC3	G1	EDATA[32]	Q1	SYSDATA[10]	W9	VCC2
B9	EDATA[49]	G2	EDATA[34]	Q2	GND	W10	GND
B10	EDATA[29]	G3	GND	Q3	SYSDATA[41]	W11	SYS_CLK[1]
B11	GND_APLL	G4	VCC1	Q4	SYSDATA[42]	W12	GND_APLL
B12	GND_PECL	G20	VCC1	Q20	GND	W13	SYSID[3]
B13	VCC_PECL	G21	GND	Q21	SYSDATA[5]	W14	VCC2
B14	EDATA[24]	G22	EDATA[41]	Q22	GND	W15	GND
B15	EDATA[21]	G23	EDATA[15]	Q23	SYSDATA[6]	W16	SYSDATA[55]
B16	GND	H1	EDATA[35]	R1	SYSDATA[40]	W17	SYSDATA[58]
B17	VCC3	H2	EDATA[36]	R2	SYSDATA[14]	W18	VCC2
B18	EDATA[16]	H3	EDATA[37]	R3	VCC2	W19	GND
B19	EDPAR[5]	H4	EDATA[38]	R4	SYSDATA[43]	W20	SYSDATA[63]
C4	EDATA[55]	H20	EDATA[14]	R20	SYSDATA[0]	X5	SYSDATA[25]
C5	GND	H21	VCC3	R21	SYSDATA[4]	X6	SYSDATA[27]
C6	VCC3	H22	EDATA[11]	R22	VCC2	X7	VCC2
C7	EBUS_CLKOUT	H23	EPD	R23	SYSDATA[3]	X8	GND
C8	TDO	J1	EDPAR[4]	S1	SYSDATA[44]	X9	SYSDATA[31]
C9	GND	J2	VCC3	S2	SYSDATA[47]	X10	S_REPLY[0]
C10	VCC3	J3	EDATA[39]	S3	GND	X11	VCC_PECL
C11	LF2B	J4	EDATA[0]	S4	VCC1	X12	LF3A
C12	LF3B	J20	EDATA[10]	S20	VCC1	X13	SYSID[4]
C13	EBUS_CLK[0]	J21	CNTL[1]	S21	SYSDATA[39]	X14	SYSDATA[48]

Pin	Pin Name	Pin	Pin Name	Pin	Pin Name	Pin	Pin Name
C14	GND	J22	UE	S22	SYSDATA[1]	X15	SYSDATA[49]
C15	VCC3	J23	CNTL[4]	S23	SYSDATA[2]	X16	VCC2
C16	EDATA[26]	K1	EDATA[3]	T1	SYSDATA[16]	X17	GND
C17	EDATA[19]	K2	GND	T2	SYSDATA[45]	X18	SYSDATA[53]
C18	GND	K3	EDATA[2]	T3	SYSDATA[46]	X19	SYSDATA[56]
C19	VCC3	K4	GND	T4	SYSDATA[18]	X20	SYSDATA[60]
D4	EDATA[57]	K20	UDB_H	T20	SYSDATA[38]	Y5	SYSDATA[26]
D5	EDATA[59]	K21	CNTL[3]	T21	GND	Y6	SYSDATA[29]
D6	EDATA[51]	K22	GND	T22	SYSDATA[36]	Y7	S_REPLY[1]
D7	GND	K23	CE	T23	SYSDATA[37]	Y8	SYSID[1]
D8	VCC1	L1	EDATA[1]	U1	SYSDATA[17]	Y9	PLL_BYPASS
D9	EDATA[53]	L2	EDATA[4]	U2	VCC2	Y10	SYSID[2]
D10	EDATA[48]	L3	EDATA[6]	U3	SYSDATA[21]	Y11	VCC_APLL
D11	VCC1	L4	EDATA[7]	U4	SYSDATA[20]	Y12	LF1A
D12	LF1B	L20	VCC1	U20	SYSDATA[61]	Y13	LF2A
D13	TMS	L21	CNTL[2]	U21	VCC2	Y14	S_REPLY[2]
D14	EDATA[27]	L22	VCC3	U22	SYSDATA[34]	Y15	SYSDATA[50]
D15	VCC1	L23	CNTL[0]	U23	SYSDATA[35]	Y16	SYSDATA[54]
D16	EDATA[25]	M1	EDATA[5]	V1	SYSDATA[19]	Y17	SYSDATA[52]
D17	EDATA[22]	M2	SYSDATA[8]	V2	GND	Y18	SYSDATA[59]
D18	EDATA[18]	M3	VCC3	V3	SYSDATA[22]	Y19	SYSDATA[57]
D19	EDATA[46]	M4	EDPAR[0]	V4	SYSDATA[23]	Y20	SYSDATA[32]

PACKAGE DIMENSIONS

256-Pin PBGA Package



ORDERING INFORMATION

Part Number	Description
STP1081ABGA-125	UltraSPARC-II Data Buffer for 250 MHz UltraSPARC-II Systems
STP1081ABGA-150	UltraSPARC-II Data Buffer for 300 MHz UltraSPARC-II Systems

Document Part Number: 805-0567-01