SPARC Technology Business

Preliminary

STP1020A

June 1995

SuperSPARC™

DATA SHEET

Highly Integrated 32-Bit RISC Microprocessor

DESCRIPTION

The STP1020A is a new member of the SuperSPARC family of microprocessor products. Like its predecessors (STP1020N and STP1020) this new part is fully SPARC version 8 compliant and is completely upward compatible with the earlier SPARC version 7 implementations running over 8500 SPARC applications and development tools.

The STP1020A is a highly integrated, high performance superscalar microprocessor designed using a state-of-the-art BiCMOS process. Through this high integration the entire processor subsystem which consists of integer and floating point execution units, memory management unit (MMU), large level-1 instruction and data caches (total of 36 KBytes of cache memory), and the bus interface unit for supporting two different buses (MBus and VBus) are implemented on a single chip. This high level of integration reduces the cost of the processor subsystem and increases the overall system reliability by reducing the total number of devices required in the system.

Features	*Benefits*
• High performance superscalar engine with 50/60 MHz operating frequency	• Delivers 73/87 SpecInt92, 84/100 SpecFp92 and 136/163 Dhrystone MIPS
• Large register window (8 windows / 136 registers)	• Fewer loads/stores, fast procedure calls/context switches
• On-chip SPARC reference MMU	• Support for virtual memory and protection
• High performance IEEE754 floating-point unit	• Increased performance for floating-point intensive applications
• Large on-chip instruction/data caches (20 KByte I-cache and 16 KByte D-cache)	• Increased performance for variety of applications by decoupling processor from slower main memory
• Large store buffer	• Reduces processor wait cycles on store operations
• Hardware integer multiply and divide	• Increased performance for many applications
• Cache coherency support for multi-processing	• Allows a wide range of scalable systems to be built
• Support of interface is VBus or MBus	• MBus mode allows direct interface to MBus. VBus mode allows interface to an external cache controller and different buses
• Built-In Self Test (BIST) logic	• Provides quick check of device integrity
• Full JTAG interface (IEEE1149.1)	• Provides better testability at the board/system level

TYPICAL STP1020A APPLICATIONS

The STP1020A is intended for use in a broad range of applications from uniprocessor desktop machines to large multiprocessor servers. Uniprocessor and multiprocessor systems can be built with STP1020A processor either in direct MBus mode or in VBus mode with the use of an external cache controller ((MXCC) STP1090). The STP1090 external cache controller supports multiprocessor configurations using either MBus or XBus interfaces with up to 2 MBytes of secondary cache.

A block diagram of a system with STP1020A interfacing directly to MBus is shown in *Figure 1*.

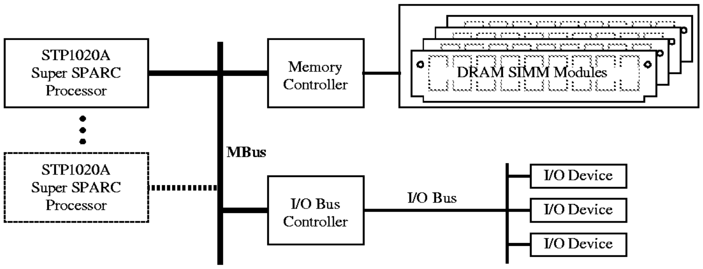


Figure 1. Typical STP1020A Uniprocessor / Multiprocessor MBus System

Figure 2 shows an STP1020A based system using external cache controller, STP1090.

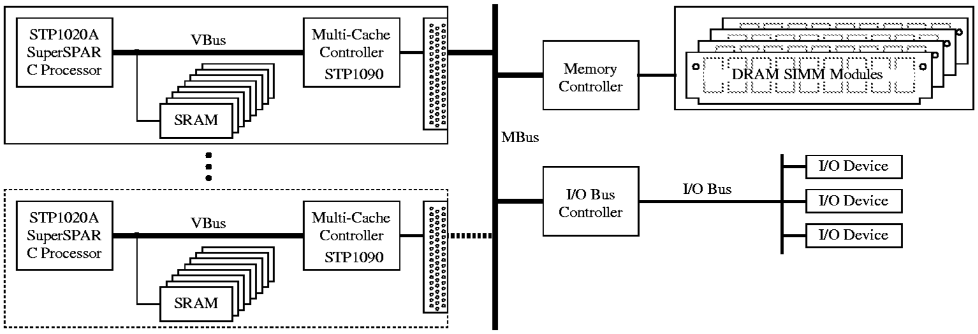


Figure 2. Typical STP1020A Uniprocessor / Multiprocessor System with External Cache

PROCESSOR MICROARCHITECTURE

Figure 3 shows an overview of the STP1020A processor microarchitecture. The SuperScalar Integer Execution Unit performs instruction grouping and then decodes/ executes arithmetic, shift, branch, and load/ store instructions. There are 3 ALUs in the Integer Execution Unit which are dynamically configured as two independent or cascadable ALUs depending on the instruction stream.

The Floating-Point Unit consists of the floating-point register file, double precision adder/ multiplier arrays, and the control logic. This unit also performs integer multiply/ divide operation. The Memory Management Unit performs virtual to physical address translations. It consists of a 64-entry fully associative TLB with hardware table walk for TLB miss processing.

The STP1020A has separate on-chip instruction and data caches that provide fast access to code and data. The 20 KByte instruction cache is a 5-way set-associative and allows fetching 4 instructions on every access using a 128 bit wide bus. The data cache is 16 KBytes and is 4-way set-associative. Both caches are physical caches.

The STP1020A supports two different bus protocols: VBus and MBus. When MBus is selected, the STP1020A can be connected directly to MBus in either uniprocessor or multiprocessor configuration.

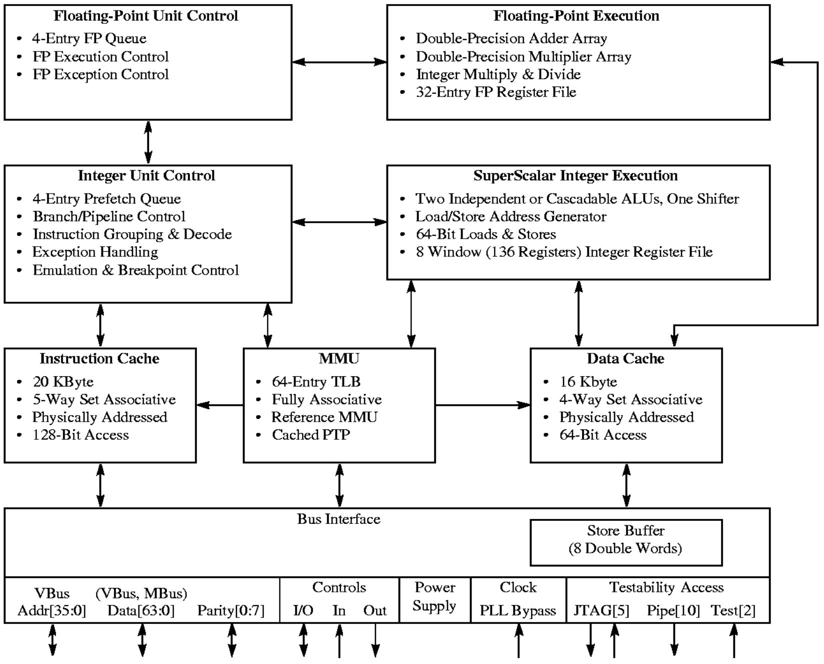


Figure 3. STP1020A SuperSPARC Functional Block Diagram

When VBus is selected, the STP1020A can be used with an external cache controller chip, STP1090, in uniprocessor or multiprocessor configuration. STP1090 chip supports both MBus or XBus interfaces and up to 2 MBytes of external cache. It is possible to design a different external cache controller which interfaces to yet another bus.

The STP1020A also integrates BIST (Built-In-Self-Test) logic, JTAG interface, and has features that support system and software debugging including hardware break-points.

MODES OF OPERATION

The STP1020A allows the system designer to select one of the two modes of operation. This selection is designed-in by statically connecting the $\overline{\text{CCMODE}}$ pin to Ground or V_{CC}. When the $\overline{\text{CCMODE}}$ pin is tied to Ground, the STP1020A operates with VBus interface. When the $\overline{\text{CCMODE}}$ pin is pulled high, the part operates with MBus interface. The selection of MBus or VBus is visible to software in MCNTL register as the “mb” bit.

VBus is a non-multiplexed synchronous bus. It is especially tailored to provide an efficient connection between the STP1020A, the STP1090(the external cache controller), and the external cache memories made up of synchronous SRAMs. It has a 36-bit address bus, and a 64-bit data bus. All transactions on the VBus are synchronized with the STP1020A clock. The arbiter for the VBus transactions is integrated on the STP1090 chip. The STP1090 supports up to 2 MBytes of external cache and multiprocessing.

In the VBus mode, the STP1020A provides an $\overline{\text{ADDR20}}$ signal besides the ADDR20 signal. The STP1020N, STP1020 do not drive this signal. The $\overline{\text{ADDR20}}$ is now driven out on a pin that used to be spare3. The $\overline{\text{ADDR20}}$ is useful in systems that incorporate 2 MB of external cache, and its integration onto the CPU eliminates an external inverter.

MBus is a SPARC International standard bus designed to function as a processor-independent bus between one or more processors and memory. It is a 64-bit multiplexed high-performance bus. It is fully synchronous with all the transfers controlled by an MBus clock. It supports block transfers in sizes up to 128 bytes with a peak transfer rate of 320 MBytes/ s. All transactions on the MBus are arbitrated by an external arbiter. The arbitration algorithm is not included in the MBus definition to allow flexibility in system design. MBus is defined for uniprocessor and multiprocessor systems. The uniprocessor form of MBus is termed “Level1”, and the multiprocessor version is called “Level2”.

VBus Mode Signals Description

Table 1 provides a description of all the STP1020A signals in the VBus mode of operation. The supply voltage signals are described in *Table 3*.

TABLE 1: Pin Descriptions - VBus Interface ($\overline{\text{CCMODE}}$ = L)

Signal	Type	Description
ADDR[35:0]	I/O	Physical address bus.
$\overline{\text{ADDR20}}$	O	Inverted physical address ADDR20. Eliminates an external inverter for 2MB cache systems.
$\overline{\text{ARDY}}$	I	This signal is an input to indicate that system logic is prepared to accept another address or bus cycle. This signal is active low. H = System not ready. L = System ready.
BURST	O	This signal is used to indicate that the current address on the bus is part of a burst bus cycle. H = Part of multi-cycle burst. L = Not part of multi-cycle burst.
$\overline{\text{BUSREQ}}$	O	Indicates VBus request by the processor. H = VBus not requested. L = VBus requested.
$\overline{\text{CCHBL}}$	O	This signal indicates that the current transaction is internally cacheable. H = Noncacheable transaction. L = Cacheable transaction.
$\overline{\text{CCMODE}}$ [1]	I	Cache controller mode. Selects the operation of the STP1020A for stand-alone operation, or for operation with a cache controller (such as the STP1090). The operation of the store buffer, data cache operation and the bus interface (VBus or MBus) are selected from this signal. This signal must be statically asserted and not changed during normal operation. H = MBus interface of operation is selected, data cache operates copy-back. L = VBus interface of operation is selected, data cache operates write-through.
$\overline{\text{CMDS}}$	I/O	Command strobe. Indicates the beginning of a bus cycle. When the STP1020A is not in bus master mode, as indicated by $\overline{\text{WGRT}}$ and $\overline{\text{RGRT}}$ being asserted, $\overline{\text{CMDS}}$ is used as an input to initiate external snoop transaction (including invalidates and demaps). H = Not a command word. L = VBus command word on ADDR35-ADDR00, $\overline{\text{CCHBL}}$, $\overline{\text{CSA}}$, $\overline{\text{DEMAP}}$, $\overline{\text{LDST}}$, SIZE1-SIZE0, $\overline{\text{SU}}$, $\overline{\text{RD}}$ and $\overline{\text{WR}}$. When the STP1020A is a bus master, it asserts this signal for the first cycle of VBus transactions. H = Not a command word. L = VBus command word on ADDR35-ADDR00, $\overline{\text{CCHBL}}$, $\overline{\text{CSA}}$, $\overline{\text{DEMAP}}$, $\overline{\text{LDST}}$, SIZE1-SIZE0, $\overline{\text{SU}}$, $\overline{\text{RD}}$, and $\overline{\text{WR}}$.
$\overline{\text{CSA}}$	O	This signal indicates that the current bus transaction is a control space access. It is asserted for the address space identifier (ASI) transactions to ASI space 0x02. H = Normal memory of ASI access. L = Control space access (to ASI 0x02).

TABLE 1: Pin Descriptions - VBus Interface ($\overline{\text{CCMODE}}$ = L) (Continued)

Signal	Type	Description
DATA[63:00]	I/O	Data bus.
$\overline{\text{DEMAP}}$	I/O	Asserted with $\overline{\text{CMDS}}$ to indicate demap cycle. As an input indicates an external demap cycle. When output: H = Normal command word. L = demap cycle system (system should remove TLB entries matching request). When input: H = Non-demap cycle. L = Demap cycle from system. The TLB entries matching request will be removed.
DPAR[0:7][1]	I/O	Data bus parity. When parity is enabled (by setting the parity bits in the MCNTL register), even parity is generated and checked. When parity is disabled, odd parity is generated but parity is not checked. DPAR0 is parity for bits DATA63-DATA56, etc., as listed: DPAR0: DATA63-DATA56 DPAR1: DATA55-DATA48 DPAR2: DATA47-DATA40 DPAR3: DATA39-DATA32 DPAR4: DATA31-DATA24 DPAR5: DATA23-DATA16 DPAR6: DATA15-DATA08 DPAR7: DATA07-DATA00
$\overline{\text{ERROR}}$	O	This signal indicates that the STP1020A has entered an error mode state and will take a watchdog reset trap. H = Normal operation. L = Error mode.
ESB	O	Execution strobe output. H = Programmed breakpoint event is occurring. L = Inactive.
IRL[3:0]	I	Interrupt request level. This field specifies the level of the highest priority interrupt request that is currently pending. If IRL3-IRL0 = 0000, no interrupts are pending. Level 15 (IRL3-IRL0 = 1111) is a NMI (disable all traps) Level 14 Highest maskable interrupt Level 1 Lowest maskable interrupt Level 0 No interrupts pending
$\overline{\text{LDST}}$	O	This signal indicates an atomic load/store (LDSTUB, LDSTUBA, SWAP or SWAPA) operation. It is equivalent to the logical OR of $\overline{\text{RD}}$ and $\overline{\text{WR}}$ signals. H = No LDST. L = Atomic Load/Store (LDST) cycle.

TABLE 1: Pin Descriptions - VBus Interface ($\overline{\text{CCMODE}}$ = L) (Continued)

<table>
<tr><th>Signal</th><th>Type</th><th>Description</th></tr>
<tr><td>$\overline{\text{MEXC}}$</td><td>I</td><td>This signal is encoded with $\overline{\text{RRDY}}$ or $\overline{\text{WRDY}}$ and with $\overline{\text{RETRY}}$ to indicate the type of acknowledgment.
<table>
<tr><th>$\overline{\text{MEXC}}$</th><th>$\overline{\text{RRDY}}$/$\overline{\text{WRDY}}$</th><th>$\overline{\text{RETRY}}$</th><th>Description</th></tr>
<tr><td>1</td><td>1</td><td>1</td><td>No Reply</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>Retry</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>Data Transfer Complete</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>Undefined Error (UD)</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>Bus Error (BE)</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>Timeout Error (TO)</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>Reserved</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>Reserved</td></tr>
</table></td></tr>
<tr><td>$\overline{\text{OE}}$</td><td>I/O</td><td>As an output, this signal controls the pipelined output enable of external cache SRAM. It is used as an input to prevent bus collisions.
H = SRAM outputs disables.
L = SRAM outputs enabled.</td></tr>
<tr><td>$\overline{\text{PEND}}$</td><td>I</td><td>This signal indicates that at least one outstanding write operation has not completed.
H = System has no incomplete write operations outstanding from this processor.
L = System has write operations that were issued by this processor that are not yet complete.</td></tr>
<tr><td>PIPE9</td><td>O</td><td>H = A valid memory reference occurred in the EO stage of the previous clock cycle.
L = No valid memory reference occurred in the EO stage of the previous clock cycle.</td></tr>
<tr><td>PIPE8</td><td>O</td><td>H = A valid floating-point operation occurred in the EO stage of the previous clock cycle.
L = No valid floating point operation occurred in the EO stage of the previous clock cycle.</td></tr>
<tr><td>PIPE7</td><td>O</td><td>H = A valid control transfer instruction was executed in the EO stage of the previous clock cycle.
L = No valid control transfer instruction was executed in the EO stage of the previous clock cycle.</td></tr>
<tr><td>PIPE 6</td><td>O</td><td>H = Indicates that no instructions were available when the group currently at the WB stage was in the DO stage.
L = Indicates that one or more instructions were available in this group.</td></tr>
<tr><td>PIPE5</td><td>O</td><td>H = The pipeline is being held by the data cache (generally processing a cache miss).
L = The pipeline is not being held by the data cache.</td></tr>
<tr><td>PIPE4</td><td>O</td><td>H = The pipeline is being held by the FPU (either queue is full or dependencies).
L = The pipeline is not being held by the FPU.</td></tr>
<tr><td>PIPE3</td><td>O</td><td>H = Indicates that the branch in EO stage of the previous cycle was taken.
L = Indicates that the branch in EO stage of the previous cycle was not taken.</td></tr>
<tr><td>PIPE2
PIPE1</td><td>O</td><td>Indicates the number of instructions in the EO stage of the current cycle.
<table>
<tr><th>PIPE2 - PIPE1</th><th>Instructions in E0 Stage</th></tr>
<tr><td>00</td><td>None</td></tr>
<tr><td>01</td><td>1</td></tr>
<tr><td>10</td><td>2</td></tr>
<tr><td>11</td><td>3</td></tr>
</table></td></tr>
<tr><td>PIPE0</td><td>O</td><td>H = Indicates that there is an exception or interrupt being signalled in the current cycle.
L = Indicates that there is no exception or interrupt being signalled in the current cycle.</td></tr>
</table>

TABLE 1: Pin Descriptions - VBus Interface ($\overline{\mathrm{CCMODE}}$ = L) (Continued)

Signal	Type	Description
$\overline{\mathrm{PLLBYP}}$ [1]	I	This pin is used to bypass the internal phase lock loop. When this pin is asserted, the external clock input will be routed directly to internal clock distribution with no delay compensation. H = PLL enabled. Normal operation. L = PLL disabled. No clock delay compensation.
$\overline{\mathrm{RD}}$	I/O	STP1020A drives $\overline{\mathrm{RD}}$ to qualify addresses on the VBus as READ cycles. It is also asserted with $\overline{\mathrm{WR}}$ for swap cycles and with $\overline{\mathrm{DEMAP}}$ for demap cycles. An an input, used for internal SRAM test only. H = Not a read cycle. L = Read (or load/store with $\overline{\mathrm{WR}}$ and $\overline{\mathrm{LDST}}$ low) cycle.
$\overline{\mathrm{RESET}}$	I	Reset. This causes an external reset for the STP1020A. At power-on, $\overline{\mathrm{RESET}}$ must be held low for at least 100 ms to all allow the PLL to stabilize. If the PLL is known to be stable, $\overline{\mathrm{RESET}}$ may be asserted for as short as 8 cycles. See reset operation. H = Normal operation. L = The STP1020A is externally reset.
$\overline{\mathrm{RETRY}}$	I	This signal is encoded along with $\overline{\mathrm{RRDY}}$ or $\overline{\mathrm{WRDY}}$ and with $\overline{\mathrm{MEXC}}$ to indicate the type of acknowledgment. See $\overline{\mathrm{MEXC}}$ table for description.
$\overline{\mathrm{RGRT}}$	I	This signal indicates that the STP1020A has been given a grant to use the VBus for read operations. H = VBus not available for read operations. L = VBus available for read operations.
$\overline{\mathrm{RRDY}}$	I	This signal indicates that incoming read data is valid. $\overline{\mathrm{RRDY}}$ may be connected to $\overline{\mathrm{WRDY}}$ when only a single ready signal is required. This signal is encoded with $\overline{\mathrm{MEXC}}$ and $\overline{\mathrm{RETRY}}$. See $\overline{\mathrm{MEXC}}$ table for description.
SIZE1 SIZE0	O	These bits indicate the transfer size of the current transaction. 00 = Byte 01 = Half word 10 = Word 11 = Doubleword
spare[2:0]	I	Not used. Should be tied high or left floating during normal chip operations.
$\overline{\mathrm{SRMTST}}$ [1]	I	Reserved: Factory test pin. It must be connected to V_{CC} for normal operation.
$\overline{\mathrm{SU}}$	O	This signal indicates that the current bus transaction is a supervisor transaction. H = User (unprivileged) transaction. L = Supervisor (privileged) transaction.
TCK [1]	I	JTAG test clock input.
TDI	I	JTAG test data input.
TDO	O	JTAG test data output.
$\overline{\mathrm{TEST}}$ [1]	I	This pin can be used for board level testing. H = Normal operation. L = All outputs except ESB and TDO are placed in a high-impedance state.
TMS [1]	I	JTAG test mode select input.

TABLE 1: Pin Descriptions - VBus Interface ($\overline{\mathrm{CCMODE}}$ = L) (Continued)

Signal	Type	Description
$\overline{\mathrm{TRST}}$ [1]	I	JTAG reset input.
VCLK	I	Primary clock source.
VPLLRC	I	Phase locked loop filter capacitor. This pin should be connected to an external 0.1µF capacitor to ground.
$\overline{\mathrm{WE}}$[0:7]	O	These signals directly control the write enable signals of synchronous SRAM used for external cache. These signals are driven only when asserted; otherwise, they are tri-state. $\overline{\mathrm{WE}}$ bit ordering corresponds to the big-endian convention (i.e. $\overline{\mathrm{WE0}}$ is the write enable for byte 0) (DATA63-DATA56). H = SRAM read. L = SRAM write.
$\overline{\mathrm{WEE}}$	I	This pin is used to control the assertion of $\overline{\mathrm{WE7}}$-$\overline{\mathrm{WE0}}$ signals. H = May not drive $\overline{\mathrm{WE7}}$-$\overline{\mathrm{WE0}}$. L = May drive $\overline{\mathrm{WE7}}$-$\overline{\mathrm{WE0}}$.
$\overline{\mathrm{WR}}$	I/O	STP1020A drives $\overline{\mathrm{WR}}$ to qualify addresses on the bus as write cycle. It is asserted with $\overline{\mathrm{RD}}$ for swaps as well as demap cycles. An an input, this signal is used to qualify invalidation requests. H = Not a write cycle. L = Write (or load/store with $\overline{\mathrm{RD}}$ and $\overline{\mathrm{LDST}}$ low) cycle.
$\overline{\mathrm{WRDY}}$	I	This signal indicates that incoming read data is valid. $\overline{\mathrm{WRDY}}$ may be connected to $\overline{\mathrm{RRDY}}$ when only a single ready signal is required. The signal is encoded with $\overline{\mathrm{MEXC}}$ and $\overline{\mathrm{RETRY}}$. See $\overline{\mathrm{MEXC}}$ table for description.
$\overline{\mathrm{WGRT}}$	I	This signal grants the STP1020A bus access for write operations. $\overline{\mathrm{WGRT}}$ may be connected to $\overline{\mathrm{RGRT}}$ when only a single grant line is required. H = VBus not available for write operations. L = VBus available for write operations.
nu [1]		Not used in the VBus interface.

[1] These pins are pulled inactive with weak internal resistive pull-ups.

MBus Mode Signals Description

Table 2 provides a description of all the STP1020A signals in the MBus mode of operation. The supply voltage signals are described in *Table 3.*

TABLE 2: Pin Descriptions - MBus Interface ($\overline{\text{CCMODE}}$ = H)

Signal	Type	Description
$\overline{\text{AERR}}$ [1]	O	In error mode, the STP1020A will perform an automatic watchdog reset. Error mode is entered when any exception is taken with traps disabled (PSR.ET=0). This signal is driven only when asserted; otherwise, it is in tri-state. H = Normal operation. L = Error Mode.
$\overline{\text{CCMODE}}$ [2]	I	Cache controller mode. Selects the operation of the STP1020A for stand-alone operation, or for operation with a cache controller (such as the STP1090). The operation of the store buffer, data cache operation and the bus interface (VBus or MBus) are selected from this signal. This signal must be statistically asserted and not changed during normal operation. H = MBus interface of operation is selected, data cache operates copy-back. L = VBus interface of operation is selected, data cache operates write-through.
CLK	I	Primary clock source.
ESB	O	Execution strobe output. H = Programmed breakpoint event is occurring. L = Inactive.
MAD[63:00]	I/O	Multiplexed Command/Data.
$\overline{\text{MAS}}$	I/O	MBus address strobe. Asserted by the bus master when an MBus command word (containing address and control information) is on MAD63-MAD00. H = No command word. L = MBus command word on MAD63-MAD0.
$\overline{\text{MBB}}$	I/O	MBus busy. Asserted when there is any active transaction on MBus. H = MBus free. L = MBus busy.
$\overline{\text{MBG}}$	I	MBus grant. This is a dedicated (not bussed) signal from the MBus arbiter to this bus master. H = Not granted. The STP1020A may not initiate an MBus transaction. L = Granted. The STP1020A may initiate an MBus transaction as soon as MBus is free.
$\overline{\text{MBR}}$	O	MBus request. This is a dedicated (not bussed) signal from the STP1020A to the MBus arbiter. H = No request. L = Requesting to initiate a transaction on MBus.

TABLE 2: Pin Descriptions - MBus Interface ($\overline{\text{CCMODE}}$ = H) (Continued)

<table>
<tr><th>Signal</th><th>Type</th><th>Description</th></tr>
<tr><td>$\overline{\text{MERR}}$</td><td>I</td><td>MBus error. Encoded along with $\overline{\text{MRDY}}$ and $\overline{\text{MRTY}}$ to indicate acknowledge type (the type of error response).
<table>
<tr><th>$\overline{\text{MERR}}$</th><th>$\overline{\text{MRDY}}$</th><th>$\overline{\text{MRTY}}$</th><th>Description</th></tr>
<tr><td>H</td><td>H</td><td>H</td><td>Idle Cycle</td></tr>
<tr><td>H</td><td>H</td><td>L</td><td>Relinquish and Retry</td></tr>
<tr><td>H</td><td>L</td><td>H</td><td>Valid Data Transfer</td></tr>
<tr><td>H</td><td>L</td><td>L</td><td>Reserved</td></tr>
<tr><td>L</td><td>H</td><td>H</td><td>Bus Error (ERROR1)</td></tr>
<tr><td>L</td><td>H</td><td>L</td><td>Timeout Error (ERROR2)</td></tr>
<tr><td>L</td><td>L</td><td>H</td><td>Uncorrectable Error (ERROR3)</td></tr>
<tr><td>L</td><td>L</td><td>L</td><td>Retry</td></tr>
</table></td></tr>
<tr><td>MID[3:0] [1]</td><td>I</td><td>MBus Module ID. The identifier of this MBus device. Usually hardwired by the system. MID3 is the Most Significant Bit (MSB) and MID0 is the Least Significant Bit (LSB).</td></tr>
<tr><td>$\overline{\text{MIH}}$</td><td>I/O</td><td>Memory inhibit. Asserted by a snooping cache when it notices a coherent read of cache block it owns. Memory responds to this signal by ignoring the request.
H = No memory inhibit.
L = Inhibit memory. The snooping cache which asserted $\overline{\text{MIH}}$ will respond with the data in place of memory.</td></tr>
<tr><td>MIRL[3:0]</td><td>I</td><td>Interrupt request level. This field specifies the level of the highest priority interrupt request that is currently pending. If MIRL3-MIRL0 = 0000, no interrupts are pending.
Level 15 (MIRL3-MIRL0 = 1111) is a NMI (disable all traps)
Level 14 Highest maskable interrupt
Level 1 Lowest maskable interrupt
Level 0 No interrupts are pending</td></tr>
<tr><td>$\overline{\text{MRDY}}$</td><td>I/O</td><td>MBus ready. Encoded along with $\overline{\text{MERR}}$ and $\overline{\text{MRTY}}$ to indicate acknowledgment type (the type of error response). See table in $\overline{\text{MERR}}$ description.</td></tr>
<tr><td>$\overline{\text{MRTY}}$</td><td>I</td><td>MBus retry. Encoded along with $\overline{\text{MERR}}$ and $\overline{\text{MRDY}}$ to indicate acknowledgment type (the type of error response). See table in $\overline{\text{MERR}}$ description.</td></tr>
<tr><td>$\overline{\text{MSH}}$ [1]</td><td>I/O</td><td>Memory shared. Asserted by a snooping cache when it notices a coherent read of a cache block it is caching. Both caches will mark the data as shared.
H = No sharing.
L = Shared data.</td></tr>
<tr><td>$\overline{\text{PEND}}$ [2]</td><td>I</td><td>This signal is generally used in the VBus interface only. It indicates to the STP1020A that at least one outstanding write operation has not completed.
H = System has no write operations outstanding from this processor.
L = system has write operations that were issued by this processor that are not yet complete.</td></tr>
<tr><td>PIPE9</td><td>O</td><td>H = A valid memory reference occurred in the EO stage of the previous clock cycle.
L = No valid memory reference occurred in the EO stage of the previous clock cycle.</td></tr>
<tr><td>PIPE8</td><td>O</td><td>H = A valid floating-point operation occurred in the EO stage of the previous clock cycle.
L = No valid floating-point operation occurred in the EO stage of the previous clock cycle.</td></tr>
</table>

TABLE 2: Pin Descriptions - MBus Interface ($\overline{\text{CCMODE}}$ = H) (Continued)

<table>
<tr><th>Signal</th><th>Type</th><th>Description</th></tr>
<tr><td>PIPE7</td><td>O</td><td>H = A valid control transfer instruction was executed in the EO stage of the previous clock cycle.
L = No valid control transfer instruction was executed in the EO stage of the previous clock cycle.</td></tr>
<tr><td>PIPE6</td><td>O</td><td>H = Indicates that no instructions were available when the group currently at the WB stage was in theD0 stage.
L = Indicates that one or more instructions were available in this group.</td></tr>
<tr><td>PIPE5</td><td>O</td><td>H = The pipeline is being held by the data cache (generally processing a cache miss).
L = The pipeline is not being held by the data cache.</td></tr>
<tr><td>PIPE4</td><td>O</td><td>H = The pipeline is being held by the FPU (either queue is full or dependencies).
L = The pipeline is not being held by the FPU.</td></tr>
<tr><td>PIPE3</td><td>O</td><td>H = Indicates that the branch in EO stage of the previous cycle was taken.
L = Indicates that the branch in EO stage of the previous cycle was not taken.</td></tr>
<tr><td>PIPE2
PIPE1</td><td>O</td><td>Indicates the number of instructions in the EO stage of the current cycle:
<table>
<tr><th>PIPE2 - PIPE1</th><th>Instructions in E0 Stage</th></tr>
<tr><td>00</td><td>None</td></tr>
<tr><td>01</td><td>1</td></tr>
<tr><td>10</td><td>2</td></tr>
<tr><td>11</td><td>3</td></tr>
</table></td></tr>
<tr><td>PIPE0</td><td>O</td><td>H = Indicates that there is an exception or interrupt being signalled in the current cycle.
L = Indicates that there is no exception or interrupt being signalled in the current cycle.</td></tr>
<tr><td>$\overline{\text{PLLBYP}}$ [2]</td><td>I</td><td>This pin is used to bypass the internal phase lock loop. When this pin is asserted, the external clock input will be routed directly to internal clock distribution with no delay compensation.
H = PLL enabled. Normal operation.
L = PLL disabled. No clock delay compensation.</td></tr>
<tr><td>$\overline{\text{RSTIN}}$</td><td>I</td><td>Reset In. This causes an external reset for the STP1020A. At power-on,$\overline{\text{RSTIN}}$ must be held low for at least 100 ms to all allow the PLL to stabilize. If the PLL is known to be stable, $\overline{\text{RSTIN}}$ may be asserted for as short as 8 cycles. See reset operation.
H = Normal operation.
L = The STP1020A is externally reset.</td></tr>
<tr><td>spare3</td><td>O</td><td>Not used. Leave floating.</td></tr>
<tr><td>spare[2:0]</td><td>I</td><td>Not used. Should be tied high or left floating during normal chip operation.</td></tr>
<tr><td>TCK [2]</td><td>I</td><td>JTAG test clock input.</td></tr>
<tr><td>TDI [2]</td><td>I</td><td>JTAG test data input.</td></tr>
<tr><td>TDO</td><td>O</td><td>JTAG test data output.</td></tr>
<tr><td>$\overline{\text{TEST}}$ [2]</td><td>I</td><td>This pin can be used for board level training.
H = Normal operation.
L = All outputs except ESB and TDO are placed in a high-impedance state.</td></tr>
<tr><td>TMS [2]</td><td>I</td><td>JTAG test mode select input.</td></tr>
</table>

TABLE 2: Pin Descriptions - MBus Interface ($\overline{\text{CCMODE}}$ = H) (Continued)

Signal	Type	Description
$\overline{\text{TRST}}$	I	JTAG test reset input.
VPLLRC	I	Phase locked loop filter capacitor. This pin should be connected to an external 0.1µF capacitor to ground.
nu [2]	I/O	Not used for MBus.

1. These pins have an open drain.
2. These pins are pulled inactive with weak internal resistive pull-ups.

Table 3 gives the description of all the supply voltages in both MBus and VBus modes of operation.

TABLE 3: Pin Descriptions - Power Connections

Signal	Type	Description
V_{CCC}	I	V_{CC} for core logic.
V_{CCCLK} [1]	I	V_{CC} for clock and PLL.
V_{CCCI}	I	V_{CC} for input buffers.
V_{CCP}	I	V_{CC} for peripheral logic.
V_{SSC}	I	Ground for core logic.
V_{SSCLK}	I	Ground for clock and PLL.
V_{SSI}	I	Ground for input buffers.
V_{SSP}	I	Ground for peripheral logic.

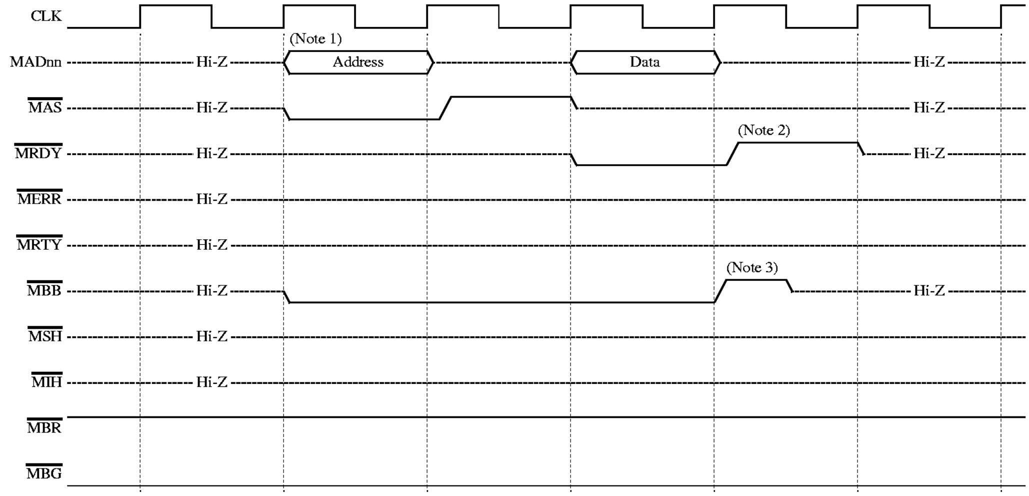
1. For stable operation of the phase lock loop (PLL), the filter circuit shown in ***Figure 20*** should be used.

MBUS TIMING

The MBus read, write and invalidate operations are explained in the following section.

MBus Single Read

The single read cycle transfers a byte, half-word, word, or a double-word. Big-endian word ordering is used (the least significant bytes in a word appear on the high bits of the bus according to SPARC standard). *Figure 4* shows an MBus single read operation.

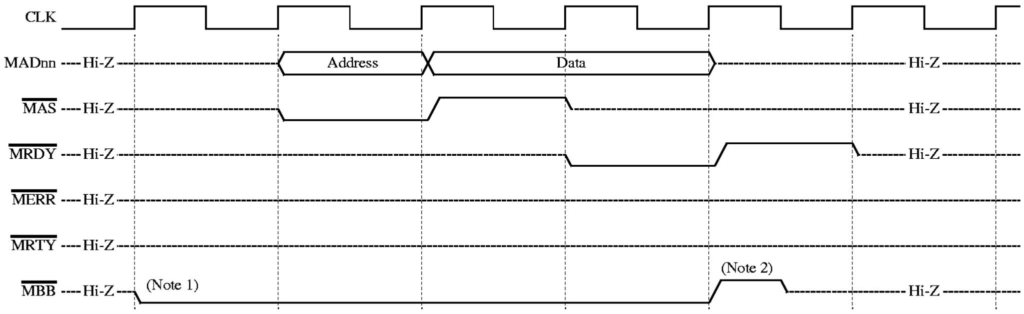


Notes: 1. MADnn lines are held to their previously driven state by system bus holders.
2. Control lines ($\overline{\text{MAS}}$, $\overline{\text{MRDY}}$, $\overline{\text{MERR}}$, $\overline{\text{MRTY}}$) are driven inactive for one clock before being released.
3. $\overline{\text{MBB}}$ is driven high for 1/2 clock cycle before being released.

Figure 4. MBus Single Read

MBus Single Write

Single write operations are queued in the STP1020A store buffer. As soon as the STP1020A receives a bus grant, the transactions will be issued on the bus. The processor will not wait during this time, unless the buffer fills. Bytes, half-words, words, and double words may all be stored, with big-endian ordering. Any errors are reported as deferred data store errors. *Figure 5* shows an MBus single write operation.



Notes: 1. $\overline{\text{MBB}}$ is driven active one cycle before MAS during write and CI cycles.
2. $\overline{\text{MBB}}$ is driven inactive for 1/2 clock cycle before being released.

Figure 5. MBus Single Write

MBus Burst Read

Figure 6 shows a 32-byte burst read operation. A read operation can be performed on any size of data transfer that is specified by the SIZE bits. Read transactions support wrapping (critical word first ordering). Transactions involving fewer than eight bytes will have undefined data on the unused bytes.

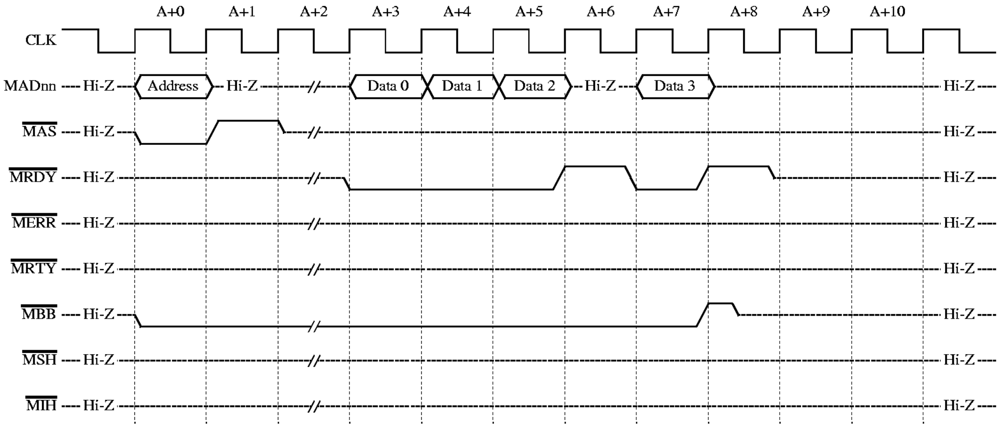
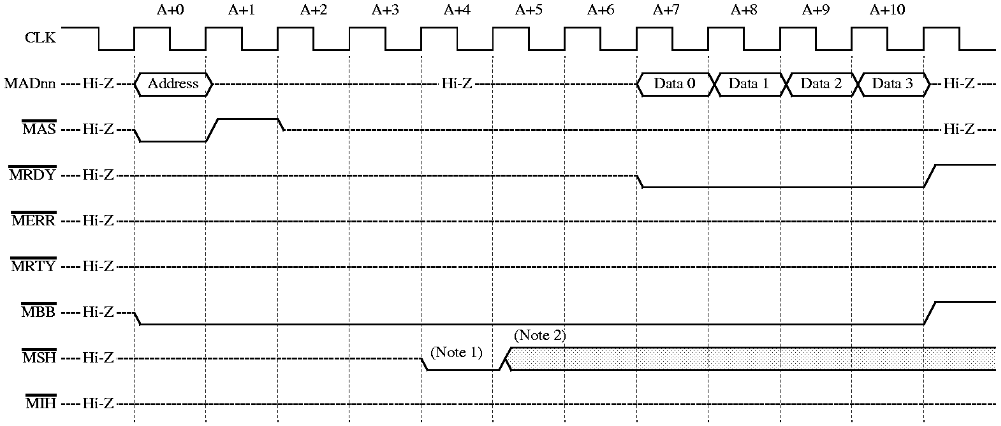


Figure 6. MBus Burst Read

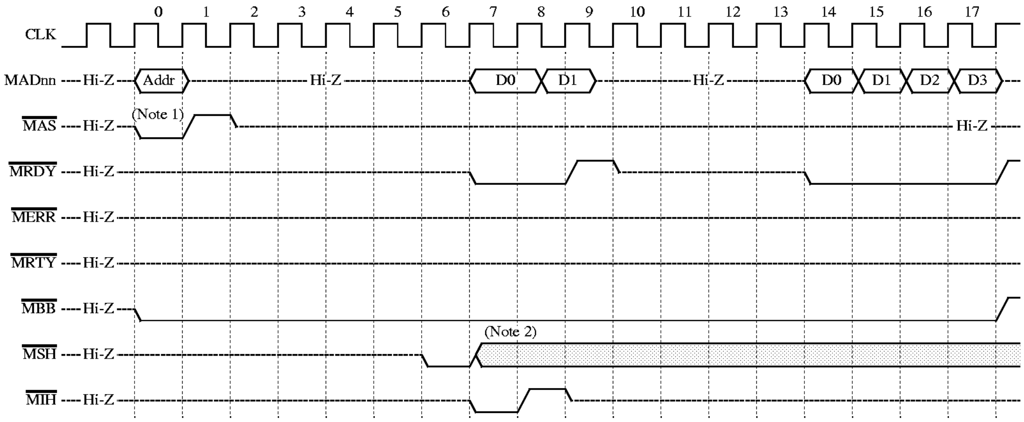
MBus Coherent Read

Coherent Read (CR) transactions are used to read data from the current owner. The owner may be memory or another cache. CR will be used for all on-board data cache load misses and all on-board instruction cache misses. If another cache owns the data, it will respond by asserting the $\overline{MIH}$ signal and providing the data. All CR transactions use critical-word-first ordering. The double-word that is needed first will be the starting address of the transaction. Double-words from memory must be returned in modulo 32-byte address order. Once the needed data arrives, the processor will use it immediately. *Figure 7* shows an MBus coherent read of shared data. Any processor that has a valid cached copy of data referenced by CR transactions must assert the $\overline{MSH}$ signal to indicate that the information is shared. The STP1020A can accept the assertion of $\overline{MSH}$ at any time until receipt of the first data word. If the data is owned by another cache, the STP1020A will ignore any data ready responses until four cycles beyond the assertion of $\overline{MIH}$. This allows memory controllers to begin transmitting data sooner. Memory controllers must not respond with data until a time equal to the maximum $\overline{\text{MIH}}$ assertion delay for any cache in the system. *Figure 8* shows an MBus coherent read of owned data.



Notes: 1. $\overline{\text{MSH}}$ may occur from A+2 to A+7.
2. $\overline{\text{MSH}}$ is an open drain signal. It is not driven inactive. The system pull-up resistor returns it to an inactive level.

Figure 7. MBUS Coherent Read of Shared Data



Notes: 1. Device is not the Master.
2. $\overline{\text{MSH}}$ is an open drain signal. It is not driven inactive. The system pull-up resistor returns it to an inactive level.

Figure 8. MBUS Coherent Read of Owned Data

MBus Coherent Invalidate

A Coherent Invalidate (CI) operation can only be performed on a block (32 bytes). All CI operations will be snooped by all snooping caches. If a Coherent Invalidate operation hits in a cache, that copy will be invalidated immediately, regardless of its state. Memory is responsible for the acknowledgment of the CI transaction. *Figure 9* shows a CI operation.

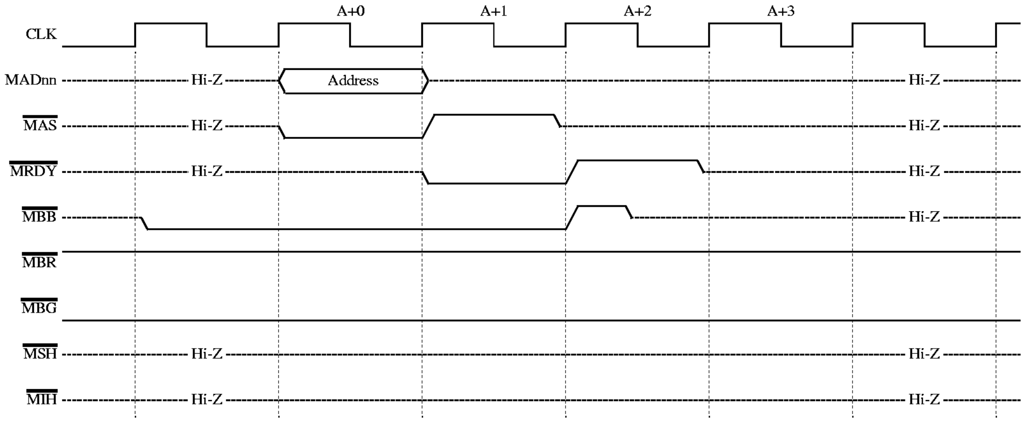


Figure 9. MBus Coherent Invalidate

Coherent Read and Invalidate

Since the MBus supports a write-invalidate type of cache-consistency protocol, a special Coherent Read and Invalidate (CRI) transaction that combines a CR transaction with the CI transaction was included to reduce the number of MBus Coherent transactions,. Caches that are performing CR transactions with the knowledge that they intend to immediately modify the data can issue this transaction.

Each CRI transaction will be snooped by all system caches. If the address hits and the cache does not own the block, that cache immediately invalidate its copy of this block, no matter what state the data was in. If the address hits and the cache owns the block, the block will assert $\overline{\text{MIH}}$ and supply the data. When the data has been successfully supplied, the cache will then invalidate its copy of this block.

$\overline{\text{MSH}}$ is not driven during the CRI transaction.

Coherent Write and Invalidate

A Coherent Write and Invalidate transaction combines a block write transaction with a CI transaction.

Each Coherent Write and Invalidate transaction will be snooped by all system caches. If the address hits, caches will invalidate their copies of this block, no matter what state the data was in. Neither $\overline{\text{MIH}}$ nor $\overline{\text{MSH}}$ is asserted for Coherent Write and Invalidate transactions. *Figure 10* shows a Coherent Write and Invalidate operation.

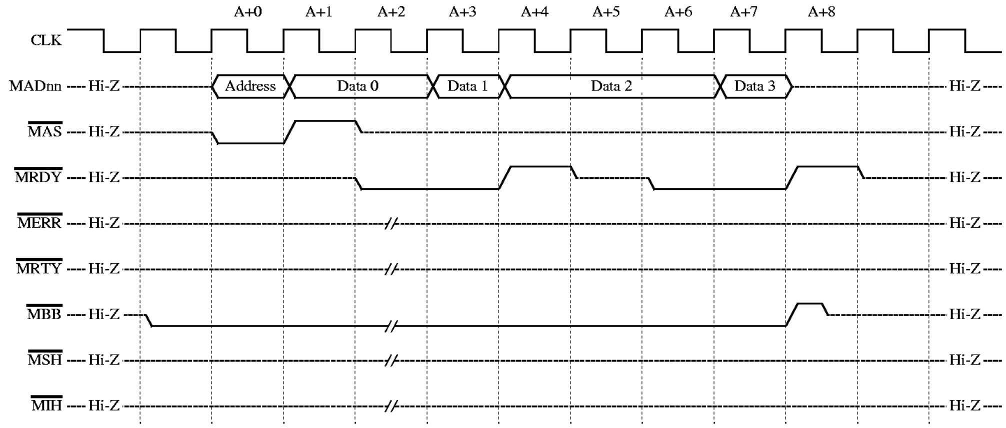


Figure 10. MBus Coherent Write and Invalidate

VBUS TIMING

The VBus read, write and invalidate operations are explained in the following section.

Cache Disabled/Non-Cacheable Single Read

Figure 11 shows a single read with the cache disabled. The external cache controller (STP1090) goes to the system bus to accomplish this operation. It deasserts $\overline{RGRT}$ to allow the STP1020A to complete pending write operations. When the data is available, the STP1090 negates grant, drives the data, and asserts $\overline{RRDY}$.

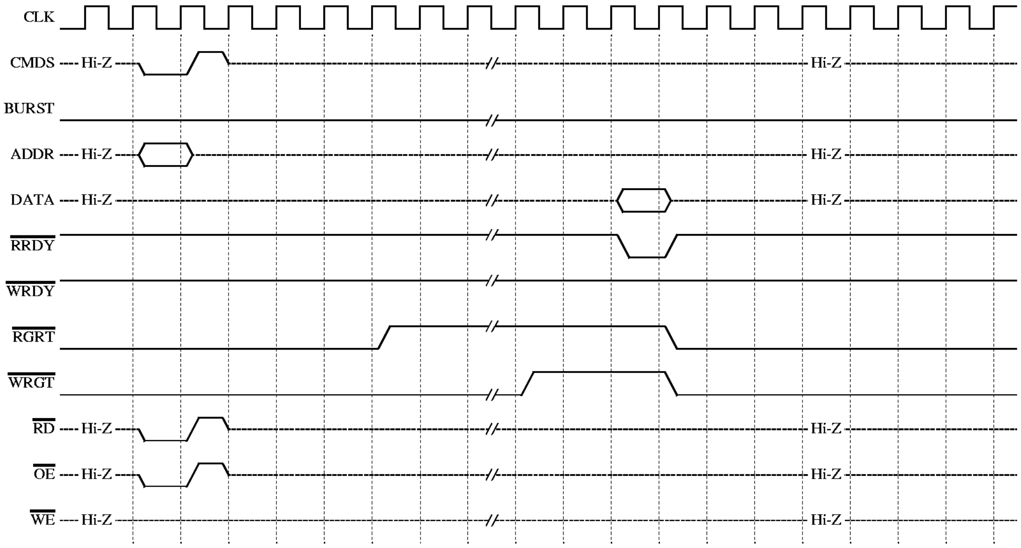


Figure 11. VBus Cache Disabled/Non-Cacheable Single Read

Cache Disabled Write (or Non-Cacheable) Write

Figure 12 shows a cache disabled write. The external cache controller (STP1090) terminates the VBus cycle by issuing a $\overline{\text{WRDY}}$ without asserting $\overline{\text{WEE}}$. A non-cacheable write would be identical.

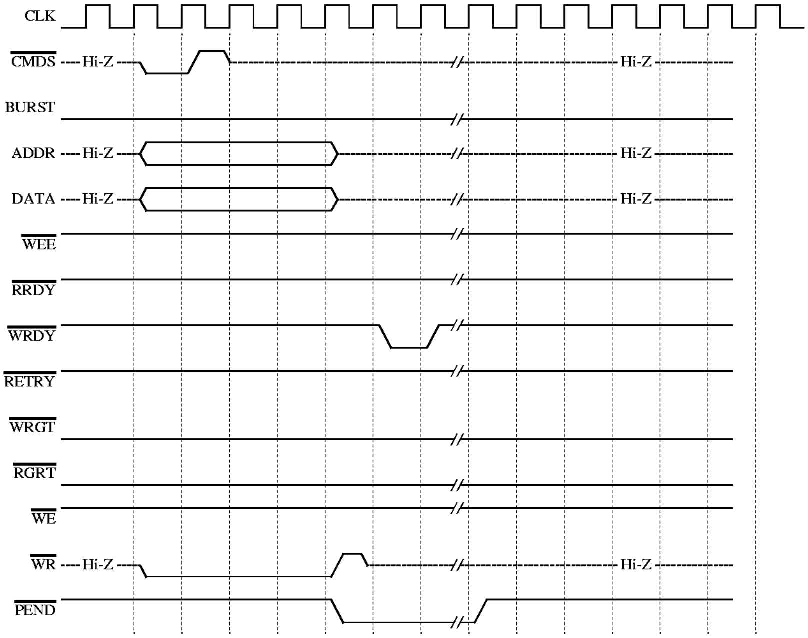


Figure 12. VBus Cache Disabled/Non-Cacheable Single Write

Cacheable Single Read Hit

Figure 13 shows a read by the STP1020A of a single cacheable word with an external cache hit. STP1020A asserts the address, cycle qualifiers, and the $\overline{\text{OE}}$ to SRAM. The STP1090 detects a tag match and issues a $\overline{\text{RRDY}}$ at the same time that the SRAMs drive data to STP1020A. The $\overline{\text{OE}}$ from STP1020A is delayed in the registers internal to the synchronous SRAMs, and the data is enabled two cycles after the $\overline{\text{OE}}$ is issued to the chip. Note that the partially bussed (not driven by the STP1020A for the entire cycle) VBus control signals are actively deasserted for 1/ 2 cycle before being released to the bus keepers.

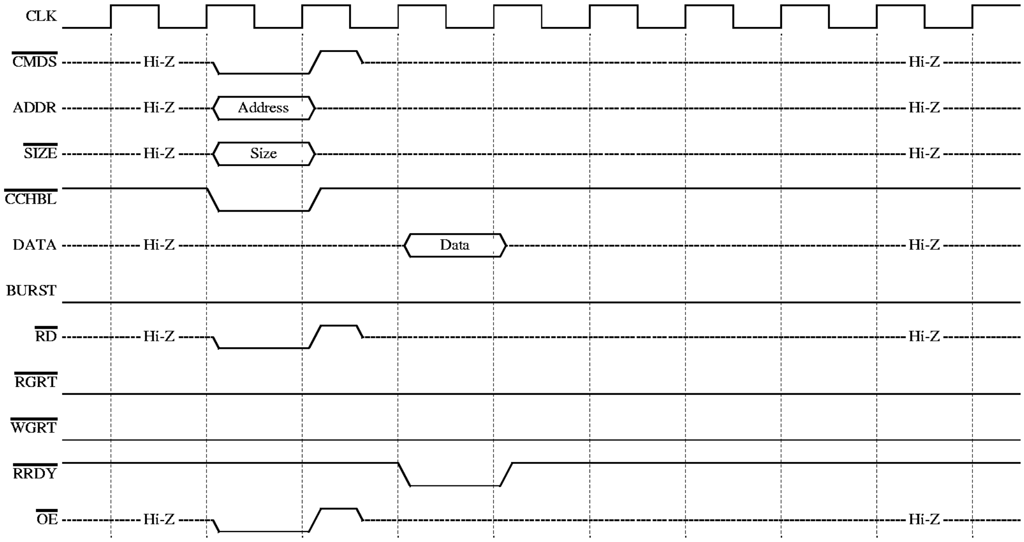


Figure 13. VBus Cacheable Single Read Hit

Cacheable Single Read Miss

Figure 14 shows a cacheable single-read miss. The STP1090 detects that a tag mismatch occurs and issues a cycle to the system bus to obtain data to fill the external cache. It removes $\overline{\text{RGRT}}$ to allow STP1020A to proceed with any write operation it may have had pending. When the system bus returns the requested data block, the STP1090 removes the bus grant to STP1020A (negates $\overline{\text{WGRT}}$) to obtain access to the SRAMs. The STP1090 writes the data into the SRAMs. The STP1090 issues a $\overline{\text{RRDY}}$ to STP1020A, as the data word requested (by STP1020A read) is driven on the DATA lines (while the data is being written into SRAMs).

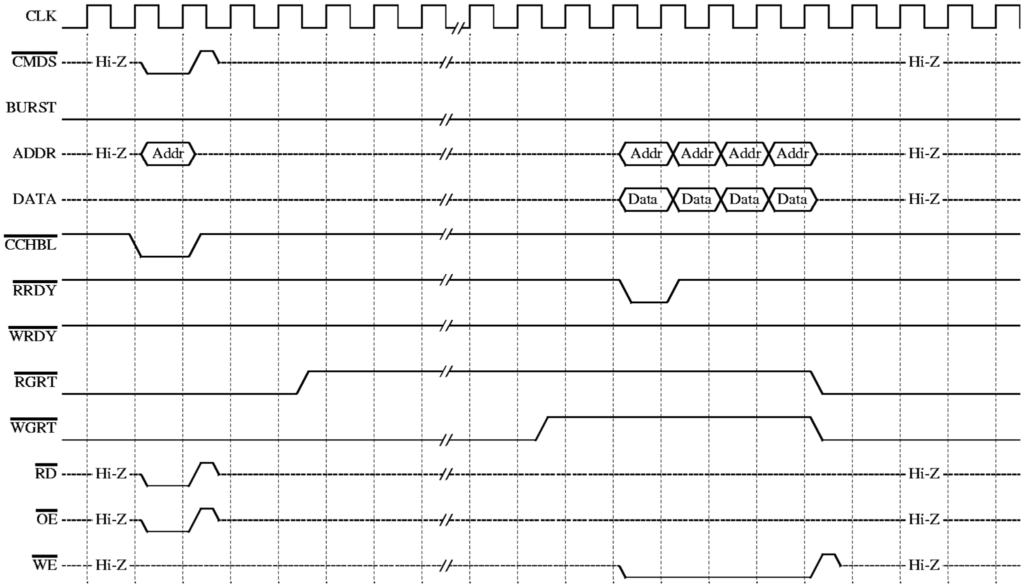


Figure 14. VBus Cacheable Single Read Miss

Burst Read Hit

Figure 15 shows a burst-read hit. As with a cacheable single-read hit, the STP1090 functions mainly to time the cycle by asserting $\overline{\text{RRDY}}$ as the SRAM provides the data.

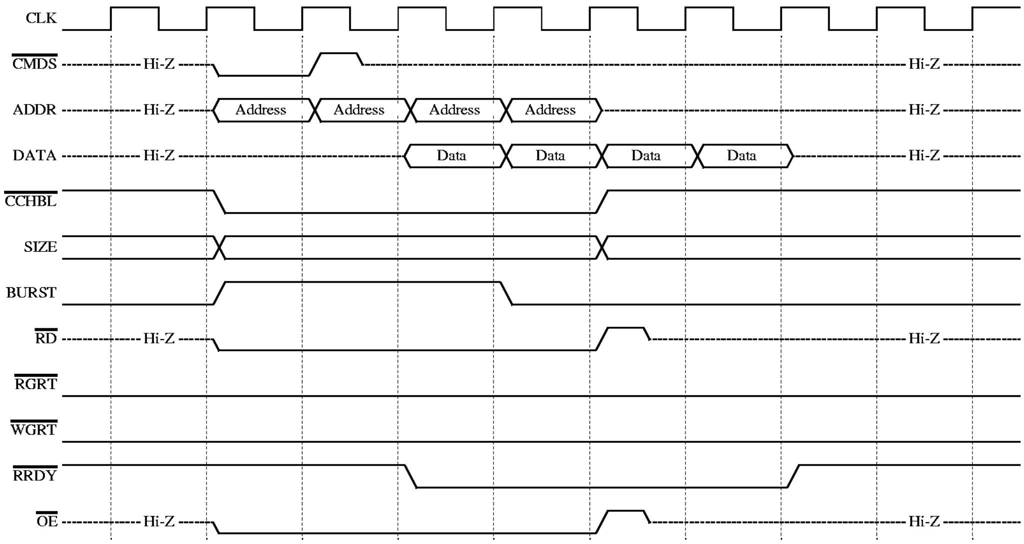


Figure 15. VBus Burst Read Hit

Burst Read Miss

Figure 16 shows a burst read miss. The external cache controller (STP1090) removes $\overline{\text{RRGT}}$ to indicate that the cycle is in progress and that STP1020A can proceed with an outstanding write if one is pending. When the data returns from the system bus, the STP1020A writes it into the SRAM and asserts $\overline{\text{RRDY}}$ when the requested data is on the VBus. Note that, in *Figure 16*, the STP1090 is in XBus configuration, and consequently the block size is 64 bytes. Only 32 bytes are sent to STP1020A, while all 64 bytes are stored in SRAM. Also note that with critical word first ordering, the data returned starts from the index into the block for the requested doubleword, continues to the last index, and then wraps from index 0 to the starting index minus 1.

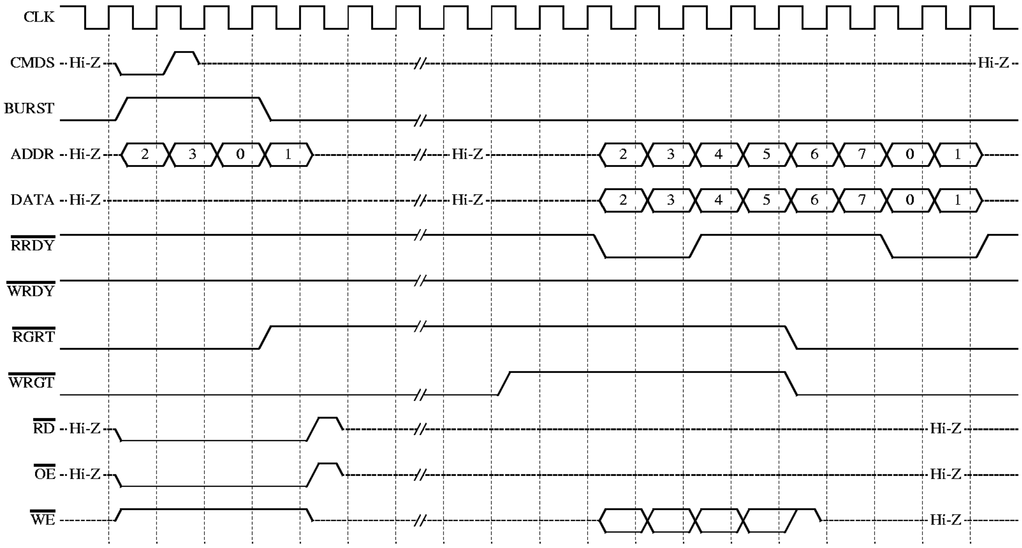


Figure 16. VBus Burst Read Miss

Cacheable Single Write Hit

Figure 17 shows a cacheable single-write hit. The STP1090 asserts $\overline{\text{WEE}}$ at the CMD + 2 cycle (i.e., two cycles after $\overline{\text{CMDS}}$) to allow the assertion of the write data (DATA, DPAR) and the write strobes ($\overline{\text{WE7}}$-$\overline{\text{WE0}}$). The STP1090 asserts the $\overline{\text{WRDY}}$ in the following cycle ($\overline{\text{CMDS}}$ + 3).

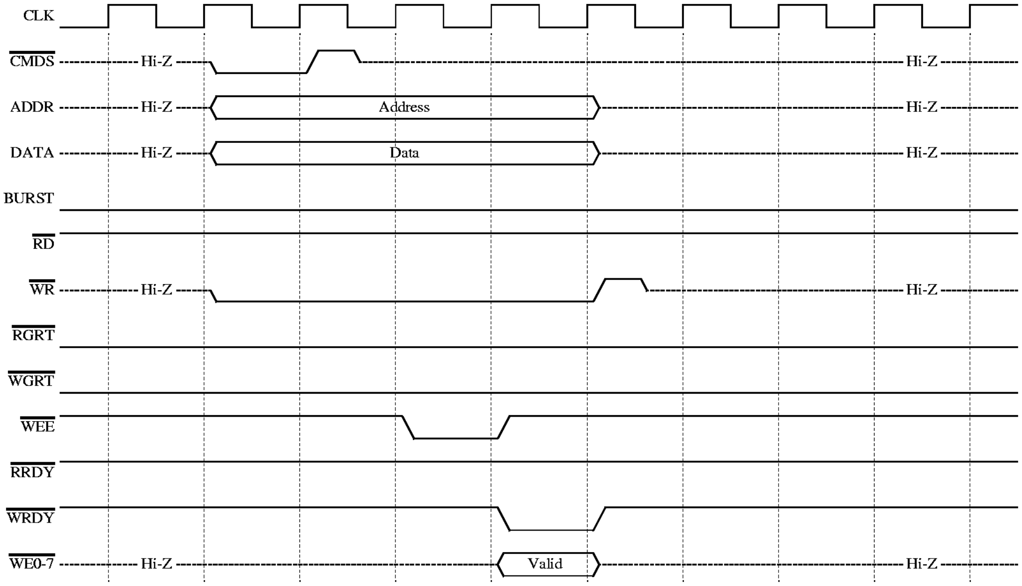


Figure 17. VBus Cacheable Single Write Hit

Cacheable Burst Write Hit

Figure 18 shows a burst write hit. It is basically the same except that $\overline{\text{WRDY}}$ is asserted for each data doubleword written in the burst. The STP1020A deasserts BURST one cycle before the last write. Each of the individual writes in the burst from the STP1020A may be from one to eight bytes and may be at any address within the cache block. The number of consecutive writes may be of arbitrary length. If the external cache controller (STP1090) needs the VBus while a burst write cycle is occurring, it can deassert the $\overline{\text{WRGT}}$ signal to terminate the burst cycle prematurely. When the STP1020A reacquires the VBus, it continues the burst write from where it was interrupted.

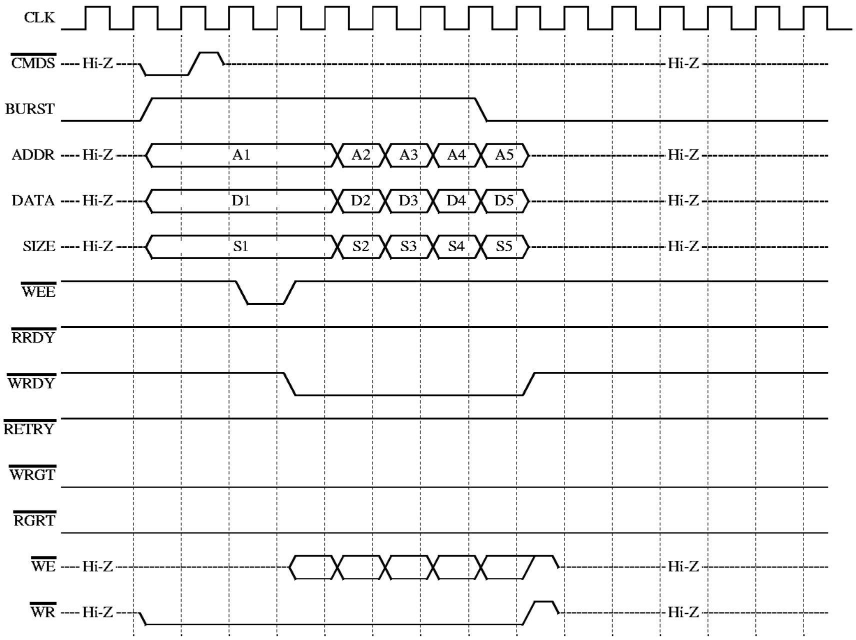


Figure 18. VBus Cacheable Burst Write Hit

Cache Invalidate

Figure 19 shows an invalidate. The external cache controller (STP1090) first removes the STP1020A from the VBus by revoking the $\overline{\text{RGRT}}$ and $\overline{\text{WGRT}}$ bus grants; it then asserts the address, $\overline{\text{WR}}$ and $\overline{\text{CMDS}}$. Multiple invalidates may occur consecutively. Invalidates may also occur when the STP1090 has obtained the VBus for SRAM reads or writes.

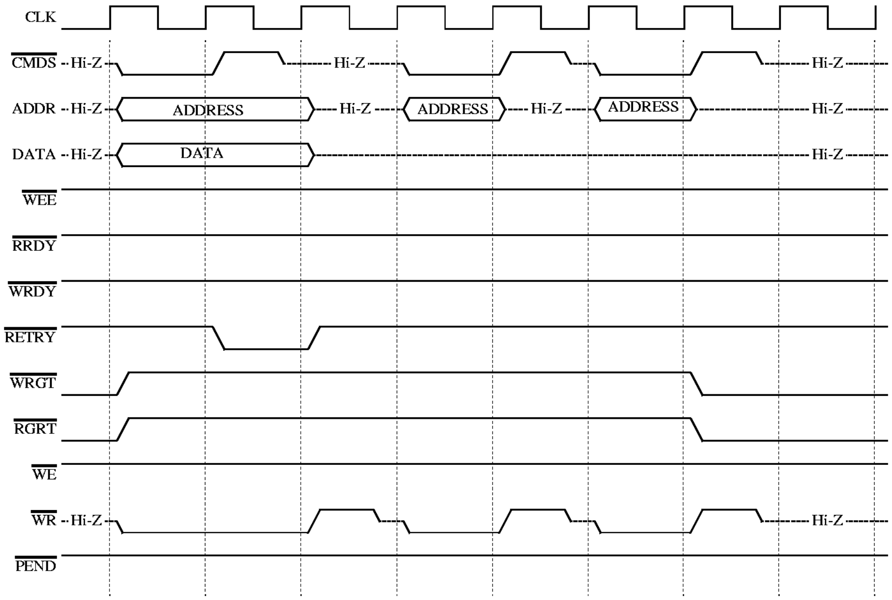


Figure 19. VBus Invalidation

Clock Operation

Proper clocking is essential at high operating frequencies. In order to reduce system clock skew, a phase lock loop (PLL) is implemented on-chip. For testing and other purposes, a PLL bypass mechanism is provided. When the $\overline{\text{PLLBYP}}$ signal is active (low), the PLL circuitry will be completely bypassed.

Phase Lock Loop Operation

The PLL operates by constantly measuring internal clock routing and gate delays and internally generating a clock that is effectively ahead of the external clock by an amount equal to the internal delay. This reduces clock skew to the internal logic. Prior to normal operation, the PLL must be allowed time to stabilize. To assure stabilization, $\overline{\text{RESET}}$ should be active for 100 ms (milliseconds).

The input clock to the STP1020A must never be stopped or changed from its normal periodic operation while the PLL is enabled. Doing so will cause PLL instability and unpredictable operation. To ensure proper operation of the PLL clock, V_{CCCLK} and V_{SSCLK} should be filtered of system noise. *Figure 20* shows a recommended filter circuit.

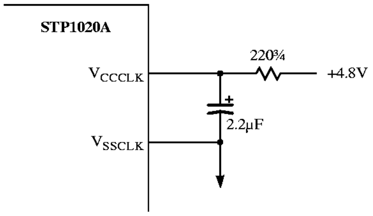


Figure 20. Typical Phase Lock Loop (PLL) Filter Circuit

Important Note: It is essential that the JTAG TAP controller be reset prior to or at the same time as $\overline{\text{RESET}}$ in order for the PLL to begin initialization. The TAP controller may be initialized either by asserting the $\overline{\text{TRST}}$ pin, or by asserting the TMS pin for five consecutive cycles of TCK (test clock). If this reset does not occur, the PLL clock feedback loop may not be established, and unpredictable operation may result. Whenever the JTAG interface is not in use by a particular system, asserting the $\overline{\text{TRST}}$ signal statically is strongly recommended.

Input Clock Requirements

The STP1020A can tolerate most clean stable clock sources when the PLL is enabled. With the PLL enabled, the STP1020A uses only the rising edge of the incoming clock. Internally, the STP1020A multiplies, then divides the clock to provide a stable 50% duty cycle clock. Input duty cycle must be at least 25% (either high or low). When the PLL is bypassed, care must be taken to provide a 50% duty cycle clock. Pin timings for operation with PLL bypassed are not fully defined.

Note: Operation in a system with the PLL bypassed is not recommended or fully specified.

RESET OPERATION

The STP1020A is reset from one of three sources: hardware reset, BIST reset, and watchdog reset. Reset from any source enables boot mode in the MMU, takes a reset trap, and starts executing from 0xFF000000.

In order for the STP1020A to properly reset care must be taken in the system implementation. In particular, JTAG operation may effect the STP1020A's ability to reset. The JTAG test access port (TAP) controller should be in the reset state when hardware reset is asserted.

Hardware Reset

Hardware reset is initiated external to the processor by asserting the $\overline{\text{RESET}}$ signal (named RSTIN in the MBus interface). Several actions are taken following a hardware reset. The STP1020A spends several hundred cycles initializing internal logic. In particular, during this time the cache column redundancy repair circuits are configured. As soon as $\overline{\text{RESET}}$ is asserted, the STP1020A will tri-state all signals immediately (except for TDO and ESB). All external logic should monitor $\overline{\text{RESET}}$ to ensure the validity of control signals. Appendix B shows what actions are taken by hardware reset.

At power-on, $\overline{\text{RESET}}$ must be held low for at least 100 ms to allow the PLL to stabilize. Once the PLL has stabilized, $\overline{\text{RESET}}$ must be asserted for an additional 16 cycles. If further reset operations are required beyond the initial power-on reset, $\overline{\text{RESET}}$ need only be asserted for a total of eight cycles.

The STP1020A implements internal RAM redundancy to increase component yield. A portion of this redundancy must be initialized each time the component is reset (hardware reset only). This initialization takes approximately 520 cycles, and is internally timed. These cycles begin once the $\overline{\text{RESET}}$ signal is deasserted and before the processor tries to fetch its first instruction. During this time the bus will be inactive, and the STP1020A will tri-state all I/O signals. All bus requests will be inactive, and the STP1020A will execute its first bus cycle approximately 525 cycles after $\overline{\text{RESET}}$ is deasserted.

Immediately after hardware reset and redundancy repair are complete, the STP1020A will execute a reset trap. This trap will cause the processor to enter boot mode (MCNTL.BT is set) and begin execution at virtual address 0xFF000000. This will force a READ-SINGLE bus operation at physical address 0xFF000000. The upper eight bits are set as a result of boot mode. In response to the read single operation, system logic should supply two valid SPARC instructions on the 64-bit data bus (in accordance with either MBus or VBus protocols). Once these instructions have entered the pipeline, another read single request for the next two instructions will appear on the bus. The physical addresses requested by these reads will be contiguous until a control transfer instruction is executed (normally within two or three instructions).

At power-on reset, the STP1020A will execute in single instruction execution mode. Multiple instruction per cycle execution is enabled by setting the MIX bit in the action register (ASI 0x4C).

BIST Reset

The built-in self test (BIST) logic generates a second type of reset, which is nearly identical to the hardware reset. The BIST operations can be requested either by software (with a STA), or via the JTAG interface. When BIST is complete, an internal reset is automatically generated. A JTAG reset must be generated for the JTAG logic to be reset. This can be done by entering the TAP test logic reset state either by asserting TMS for five consecutive TCK cycles or asserting TRST.

Watchdog Reset

In addition to the hardware reset, there is an internally generated reset referred to as a watchdog reset. This reset is caused by entry into error mode (trapping with ET bit of PSR set to 0).

To allow recovery from many error mode conditions, the only MMU control bit affected by a watchdog reset is the boot mode (BT) bit of the MCNTL register. The error mode (EM) bit of the MFSR is set to indicate that this is a watchdog reset, as opposed to a hardware reset. Breakpoints are cleared at watchdog reset.

When using the VBus interface, the STP1020A issues an error mode bus cycle, causing the cache controller (or external system logic) to record the occurrence of error mode. The completion of this bus cycle causes watchdog reset. Once the above actions have been completed, a reset trap will be generated.

The table in Appendix B gives the state of the internal registers after a hardware, BIST, or a Watchdog reset.

DEBUG SUPPORT

The STP1020A has Built-In Self Test (BIST) logic on-chip. BIST is a quick check for device integrity and not an exhaustive proof of the device function. Many types of device faults will be detected by an incorrect signature value after a BIST. There are two types of BIST: short and long versions. The long BIST operation, though a more exhaustive check of the logic than the short BIST, is not supported [1].To initiate BIST, and STA instruction to ASI 0x39 is issued. Stores to virtual addresses 0x0, and 0x100 select short and the unsupported long BISTs respectively. Once initiated, the internal logic controls the BIST operation. An external reset aborts the BIST operation. When the sequence completes, an internal reset is generated [2] . Then the BIST status and the signature can be read with loads from virtual addresses 0x100, and 0x0 with ASI 0x39.

The STP1020A also provides several device pins to monitor processor operation, and control processor signals. These pins are External Strobe (ESB) pin, PIPE9-0 pins, and the $\overline{\text{TEST}}$ pin. The ESB pin allows an external device to be triggered when an internal breakpoint is detected. This pin operates under software control to provide a programmable external synchronization pulse. It may be triggered by code, data, or cycle count breakpoint detection. The PIPE9-0 pins are provided to monitor the internal state of the processor and can be used to aid in system hardware and software debug. The $\overline{\text{TEST}}$ pin, when asserted, will tri-state the outputs of all the output buffers, except the TDO and ESB. This allows external test equipment to control the device's signals. Processor state is not assured after assertion of $\overline{\text{TEST}}$.

---

1. Long BIST is not verified during manufacturing test. Long BIST signatures are not provided.
2. After BIST completes, the STP1020A generates an internal reset. This internal reset behaves similar to the hardware reset. One of the consequences of this reset is that MCNTL gets initialized; in particular, the parity enable bit of the MCNTL register gets reset so that the STP1020A starts generating odd parity on the pins. This internally generated reset is not seen by a cache controller such as the STP1090.

The STP1020A provides five-signals for supporting the IEEE 1149.1 standard JTAG serial scan interface. This interface is used for manufacturing fault coverage testing, periphery and interconnect testing, Built-In Self Test (BIST), and remote debugging environment.

ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings [1]

Symbol	Parameter	Rating	Units
V_{CC}	Supply voltage range	0 to 5.5	V
V_I	Input voltage range	-0.5 to V_{CC} + 0.5	V
V_O	Output voltage range	-0.5 to V_{CC} + 0.5	V
I_{IK}	Input clamp current ($V_I < 0$ or $V_I > V_{CC}$)	±20	mA
I_{OK}	Output clamp current ($V_O < 0$ or $V_O > V_{CC}$)	±50	mA
	Current into any output in the low state	96	mA
T_{STG}	Storage temperature	-65 to 150	°C

1. Operation of the device at values in excess of those listed above will result in degradation or destruction of the device. All voltages are defined with respect to ground. Functional operation of the device at these or any other conditions beyond those indicated under " recommended operating conditions " is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

Recommended Operating Conditions [1]

Symbol	Parameter		Min	Typ	Max	Units
V_{CC} [2]	Supply voltage		4.75	4.8	4.85	V
V_{SS} [3]	Ground		–	0	–	V
V_{IH}	Input high voltage [4]	All except CLK/VCLK	2.0	–	V_{CC} + 0.3	V
		CLK/VCLK	2.2	–	V_{CC} + 0.3	V
V_{IL}	Input low voltage [4]		-0.3	–	0.8	V
I_{OH}	Output high current		–	–	-2.0	mA
I_{OL}	Output low current	All outputs except $\overline{\text{MSH}}$	–	–	2.2	mA
		$\overline{\text{MSH}}$	–	–	8.0	mA
T_J	Operating junction temperature		–	–	85	°C
T_A	Operating ambient temperature		0	–	[5]	°C

1. The Vcc should be regulated within a tolerance of 1% of the typical 4.8V.
2. V_{CC} includes V_{CCC}, V_{CCCLK}, V_{CCI}, V_{CCp}.
3. V_{SS} includes V_{SSC}, V_{SSCLK}, V_{SSI}, V_{SSP}.
4. V_{IH} (max) and V_{IL} (min) are characterized but not tested during manufacturing test.
5. Maximum ambient temperature is limited by air flow such that the maximum junction temperature does not exceed T_J.

DC Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Units
V_{OH}	Output high voltage [1]	I_{OH} = Max, V_{CC} = Min	2.4	–	–	V
V_{OL}	Output low voltage	I_{OL} = Max, V_{CC} = Max	–	–	0.4	V
I_{CC}	Supply current	V_{CC} = Max, f=50 MHz $t_{w(VCLK)}$ or $t_{W(CLK)}$ = Min	–	–	2.5	A
		V_{CC} = Max, f=60 MHz $t_{w(VCLK)}$ or $t_{W(CLK)}$ = Min	–	–	2.8	A
I_{CCQ}	Quiescent power supply current	V_{CC} = Max, $V_I = V_{SS}$ or V_{CC}	–	425	–	mA
I_{OZ}	High-impedance output current	V_{CC} = Max, V_O = 2.4V	–	–	20	µA
		V_{CC} = Max, V_O = 0.4V	–	–	-20	µA
I_I	Input current	$V_I = V_{SS}$ to V_{CC}, Inputs with pullups	–	–	250	µA
		$V_I = V_{SS}$ to V_{CC}, All other inputs	–	–	50	µA
C_I	Input capacitance [2]		–	5	–	pF
C_O	Output capacitance [2]		–	10	–	pF

1. Open drain pins $\overline{\text{AERR}}$ and $\overline{\text{MSH}}$ not driven high.
2. This specification is provided as an aid to board design. This specification is not assured during manufacturing testing.

AC Characteristics: VBus Timing - Setup and Hold[1]

Symbol	Parameter	Signals	50 MHz		60 MHz[2]		Unit
			Min	**Max**	**Min**	**Max**	
t_{su}(VA)	VBus setup to VCLK	ADDR35-ADDR0, $\overline{\text{ADDR20}}$	4.5	–	4.0	–	ns
t_{su}(VD)	VBus setup to VCLK	DATA63-DATA0, DPAR0-DPAR7	4.5	–	4.0	–	ns
t_{su}(VC1)	VBus setup to VCLK	$\overline{\text{RD}}$, $\overline{\text{WR}}$, $\overline{\text{CMDS}}$, $\overline{\text{DEMAP}}$	7.5	–	4.0	–	ns
t_{su}(VC2)	VBus setup to VCLK	$\overline{\text{WEE}}$, $\overline{\text{WRDY}}$, $\overline{\text{RRDY}}$, $\overline{\text{WGRT}}$, $\overline{\text{RGRT}}$, $\overline{\text{MEXC}}$, $\overline{\text{RETRY}}$	7.5	–	7.0	–	ns
t_{su}(IRL)	VBus setup to VCLK	IRL (synchronous)[3]	7.5	–	4.0	–	ns
t_{su}(OE)	VBus setup to VCLK	$\overline{\text{OE}}$	4.5	–	4.0	–	ns
t_{su}(ARDY)	VBus setup to VCLK	$\overline{\text{ARDY}}$	7.5	–	7.0	–	ns
t_{su}(PEND)	VBus setup to VCLK	$\overline{\text{PEND}}$ setup to VCLK	7.5	–	4.0	–	ns
t_h(VA)	VBus hold from VCLK	ADDR35-ADDR0, $\overline{\text{ADDR20}}$	0.5	–	0.5	–	ns
t_h(VD)	VBus hold from VCLK	DATA63-DATA0, DPAR0-DPAP7	0.5	–	0.5	–	ns
t_h(VC1)	VBus hold from VCLK	$\overline{\text{RD}}$, $\overline{\text{WER}}$, $\overline{\text{CMDS}}$, $\overline{\text{DEMAP}}$	0.5	–	0.5	–	ns
t_h(VC2)	VBus hold from VCLK	$\overline{\text{WEE}}$, $\overline{\text{WRDY}}$, $\overline{\text{RRDY}}$, $\overline{\text{WGRT}}$, $\overline{\text{RGRT}}$, $\overline{\text{MEXC}}$, $\overline{\text{RETRY}}$	0.5	–	0.5	–	ns
t_h(IRL)	VBus hold from VCLK	IRL (synchronous)[3]	0.5	–	0.5	–	ns
t_h(OE)	VBus hold from VCLK	OE	0.5	–	0.5	–	ns
t_h(ARDY)	VBus hold from VCLK	$\overline{\text{ARDY}}$	0.5	–	0.5	–	ns
t_h(PEND)	VBus hold from VCLK	$\overline{\text{PEND}}$	0.5	–	0.5	–	ns

1. VBus Timings are preliminary based on initial characterization, and are subject to change.
2. For optimum 60 MHz operation, allow a minimum margin of 1.5 ns for module / motherboard layout design.
3. IRL are asynchronous inputs. Times given are minimum to assure synchronous operation.

AC Characteristics: VBus Timing - Switching Characteristics [1] [2]

Symbol	Parameter	Test Conditions	50 MHz		60 MHz [3]		Unit
			Min	Max	Min	Max	
t_p(VA)	Propagation delay, VCLK to ADDR35-ADDR0, $\overline{\text{ADDR20}}$	I_{OH} = Max I_{OL} = Max C_L = 35pF V_{load} = 2.25V	–	14.0	–	10.0	ns
t_p(VD)	Propagation delay, VCLK to DATA63-DATA0, DPAR0-DPAR7		–	14.0	–	10.0	ns
t_p(VC1)	Propagation delay, VCLK to $\overline{\text{RD}}$, $\overline{\text{WR}}$, $\overline{\text{CMDS}}$, $\overline{\text{DEMAP}}$		–	11.5	–	8.5	ns
t_p(VC3)	Propagation delay, VCLK to BURST, $\overline{\text{CCHBL}}$, $\overline{\text{CSA}}$, $\overline{\text{LDST}}$, $\overline{\text{SU}}$		–	11.5	–	8.5	ns
t_p(SIZE)	Propagation delay, VCLK to SIZE1-SIZE0		–	11.5	–	10.0	ns
t_p(OE)	Propagation delay, VCLK to $\overline{\text{OE}}$		–	11.5	–	8.5	ns
t_p(WE)	Propagation delay, VCLK to $\overline{\text{WE7}}$-$\overline{\text{WE0}}$		–	14.0	–	11.0	ns
t_p(ERR)	Propagation delay, VCLK to $\overline{\text{ERROR}}$		–	11.5	–	8.5	ns
t_p(BUSR)	Propagation delay, VCLK to $\overline{\text{BUSREQ}}$		–	12.5	–	12.5	ns
t_{oh}(VA)	Output hold time, VCLK to ADDR35-ADDR0, $\overline{\text{ADDR20}}$		1.0	–	1.0	–	ns
t_{oh}(VD)	Output hold time, VCLK to DATA63-DATA0, DPAR0-DPAR7		2.0	–	1.0	–	ns
t_{oh}(VC1)	Output hold time, VCLK to $\overline{\text{RD}}$, $\overline{\text{WR}}$, $\overline{\text{CMDS}}$		1.0	–	1.0	–	ns
t_{oh}(DEMAP)	Output hold time, VCLK to $\overline{\text{DEMAP}}$		1.0	–	1.0	–	ns
t_{oh}(VC3)	Output hold time, VCLK to BURST, $\overline{\text{CCHBL}}$, $\overline{\text{CSA}}$, $\overline{\text{LDST}}$, $\overline{\text{SU}}$		1.0	–	1.0	–	ns
t_{oh}(SIZE)	Output hold time, VCLK to SIZE1-SIZE0		1.0	–	1.0	–	ns
t_{oh}(OE)	Output hold time, VCLK to $\overline{\text{OE}}$		2.0	–	1.0	–	ns
t_{oh}(WE)	Output hold time, VCLK to $\overline{\text{WE7}}$-$\overline{\text{WE0}}$		1.5	–	1.0	–	ns
t_{oh}(ERR)	Output hold time, VCLK to $\overline{\text{ERROR}}$		2.0	–	1.0	–	ns
t_{oh}(BUSR)	Output hold time, VCLK to $\overline{\text{BUSREQ}}$		2.0	–	1.0	–	ns

1. Switching characteristics are given with maximum number of outputs simultaneously switching.
2. VBus Timings are preliminary based on initial characterization, and are subject to change.
3. For optimum 60 MHz operation, allow a minimum margin of 1.5 ns for module / motherboard layout design.

AC Characteristics: MBus Timing - Setup and Hold[1]

Symbol	Parameter	Signals	40 MHz		50 MHz		Unit
			Min	Max	Min	Max	
t_{su}(MAD)	MBus MAD lines setup to CLK	MAD63-MAD0	5.0	–	5.0	–	ns
t_{su}(MAS)	MBus bused setup to CLK	$\overline{\text{MAS}}$	6.0	–	5.0	–	ns
t_{su}(MC2)	MBus bused setup to CLK	$\overline{\text{MERR}}$, $\overline{\text{MRTY}}$	6.0	–	5.0	–	ns
t_{su}(MRDY)	MBus bused setup to CLK	$\overline{\text{MRDY}}$	6.0	–	5.0	–	ns
t_{su}(MBB)	MBus bused setup to CLK	$\overline{\text{MBB}}$	6.5	–	5.0	–	ns
t_{su}(MBG)	MBus point-to-point setup to CLK	$\overline{\text{MBG}}$	8.0	–	6.0	–	ns
t_{su}(MSH)	MBus $\overline{\text{MSH}}$ setup to CLK	$\overline{\text{MSH}}$	6.0	–	5.0	–	ns
t_{su}(MIH)	MBus $\overline{\text{MIH}}$ setup to CLK	$\overline{\text{MIH}}$	6.0	–	5.0	–	ns
t_{su}(MIRL)	MBus MIRL setup to CLK	MIRL3-MIRL0	6.0	–	6.0	–	ns
t_h(MAD)	MBus MAD lines hold from CLK	MAD63-MAD0	1.0	–	1.0	–	ns
t_h(MC1)	MBus bused hold from CLK	$\overline{\text{MAS}}$, $\overline{\text{MRDY}}$, $\overline{\text{MBB}}$	1.0	–	1.0	–	ns
t_h(MC2)	MBus point-to-point hold from CLK	$\overline{\text{MERR}}$, $\overline{\text{MRTY}}$	1.0	–	1.0	–	ns
t_h(MBG)	MBus Point-to-point hold from CLK	$\overline{\text{MBG}}$	1.0	–	1.0	–	ns
t_h(MSH)	MBus $\overline{\text{MSH}}$ hold from CLK	$\overline{\text{MSH}}$	1.0	–	1.0	–	ns
t_h(MIH)	MBus $\overline{\text{MIH}}$ hold from CLK	$\overline{\text{MIH}}$	1.0	–	1.0	–	ns
t_h(MIRL)	MBus MIRL hold to CLK	MIRL3-MIRL0	1.0	–	1.0	–	ns

1. MBus Timings are preliminary based on initial characterization, and are subject to change.

AC Characteristics: MBus Timing - Switching Characteristics [1] [2]

Symbol	Parameter	Test Conditions	40 MHz		50 MHz		Unit
			Min	Max	Min	Max	
t_p(MAD)	Propagation delay, CLK to MBus MAD63-MAD0	I_{OL} = Max I_{OH} = Max C_L = 35 pf V_{LOAD} = 2.25V	–	13.5	–	11.5	ns
t_p(MC1)	Propagation delay, CLK to MBus control $\overline{\text{MAS}}$, $\overline{\text{MRDY}}$, $\overline{\text{MBB}}$		–	13.5	–	10.0	ns
t_p(MIH)	Propagation delay, CLK to MBus control $\overline{\text{MIH}}$		–	13.5	–	10.0	ns
t_p(MBR)	Propagation delay, CLK to MBus point-to-point $\overline{\text{MBR}}$		–	12.5	–	12.0	ns
t_p(MSHHL)	Propagation delay, CLK to $\overline{\text{MSH}}$ high to low		–	13.5	–	10.0	ns
t_p(MSHLH)	Propagation delay, CLK to $\overline{\text{MSH}}$ low to high [3]		–	13.5	–	10.0	ns
t_p(AERRHL)	Propagation delay, CLK to $\overline{\text{AERR}}$ high to low		–	13.5	–	10.0	ns
t_p(AERRLH)	Propagation delay, CLK to $\overline{\text{AERR}}$ low to high [3]		–	13.5	–	10.0	ns
t_{oh}(MAD)	Output hold time, CLK to MBus MAD63-MAD0		2.5	–	1.5	–	ns
t_{oh}(MC1)	Output hold time, CLK to MBus control $\overline{\text{MAS}}$, $\overline{\text{MRDY}}$, $\overline{\text{MBB}}$		2.5	–	1.5	–	ns
t_{oh}(MIH)	Output hold time, CLK to MBus control $\overline{\text{MIH}}$		1.5	–	1.5	–	ns
t_{oh}(MBR)	Output hold time, CLK to MBus point-to-point $\overline{\text{MBR}}$		2.5	–	1.5	–	ns
t_{oh}(MSH)	Output hold time, $\overline{\text{MSH}}$ output hold from clock		2.5	–	1.5	–	ns
t_{oh}(AERR)	Output hold time, $\overline{\text{AERR}}$ output hold from clock		2.5	–	1.5	–	ns

1. Switching characteristics are given with maximum number of outputs simultaneously switching,
2. MBus Timings are preliminary based on initial characterization, and are subject to change.
3. These values are characterized, not tested. The timing depends on system loading.

Clock Timing [1]

Symbol	Parameter	50 MHz			60 MHz			Unit
		Min	**Typ**	**Max**	**Min**	**Typ**	**Max**	
t_w(VCLK)	VCLK pulse duration [2]	20	–	33	16.6	–	33	ns
	VCLK duty cycle	25	50	75	25	50	75	%
t_w(TCK)	TCK pulse duration	100	–	–	100	–	–	ns
	TCK duty cycle	–	50	–	–	50	–	%
t_w(CLK)	CLK pulse duration	20	–	33	20	–	33	ns
	CLK duty cycle	25	50	75	25	50	75	%

1. This is for the PLL enabled. If the PLL is disabled, the part supports a fully static design. The timing parameters are not assured, since this is not tested.
2. Switching characteristics are given with maximum number of outputs simultaneously switching.

JTAG and Miscellaneous Timing - Setup and Hold

Symbol	Parameter	Signal	50 MHz		60 MHz		Unit
			Min	**Max**	**Min**	**Max**	
t_{su}(RESET)	Setup to VCLK	$\overline{\text{RESET}}$ (synchronous) [1]	10	–	10	–	ns
t_{su}(TDI)	JTAG setup to TCK	TDI	10	–	10	–	ns
t_{su}(TMS)	JTAG setup to TCK	TMS	10	–	10	–	ns
t_h(RESET)	Hold from VCLK	$\overline{\text{RESET}}$ (synchronous) [1]	6	–	6	–	ns
t_h(TDI)	JTAG hold from TCK	TDI	20	–	20	–	ns
t_h(TMS)	JTAG hold from TCK	TMS	20	–	20	–	ns

1. RESET can occur anytime during a clock cycle. Time given is minimum to ensure synchronous operation.

JTAG and Miscellaneous Timing - Switching Characteristics [1]

Symbol	Parameter	Condition	50 MHz		60 MHz		Unit
			Min	**Max**	**Min**	**Max**	
t_p(TDO)	TCK (falling edge) to TDO	I_{OL} = Max I_{OH} = Max V_{LOAD} = 2.25V C_L = 35 pF	–	25	–	25	ns
t_{oh}(TDO)	TCK (falling edge) to TDO		5	–	5	–	ns
t_p(PIPE)	VCLK to PIPE9-PIPE0		–	14.5	–	14.5	ns
t_p(ESB)	VCLK to ESB		–	14.5	–	14.5	ns
t_{oh}(PIPE)	VCLK to PIPE9-PIPE0		0.5	–	0.5	–	ns
t_{oh}(ESB)	VCLK to ESB		0.5	–	0.5	–	ns

1. Switching characteristics are given with maximum number of outputs switching simultaneously.

PARAMETER MEASUREMENT INFORMATION

Load Circuit Parameters

Timing Parameters		C_{LOAD} [1] (pF)	I_{OL} (mA)	I_{OH} (mA)	V_{LOAD} (V)
t_{en}	t_{PZH}	35	2.2	-2.0	2.25
	t_{PZL}				
t_{dis}	t_{PHZ}	35	2.2	-2.0	2.25
	t_{PLZ}				
t_{PD}		35	2.2	-2.0	2.25
$t_{PD(MSH)}$		35	8.0	-2.0	2.25

1. C_{LOAD} includes probes and test fixture capacitance.

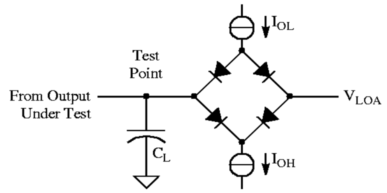
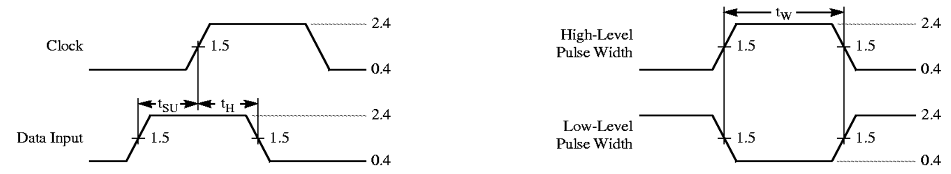
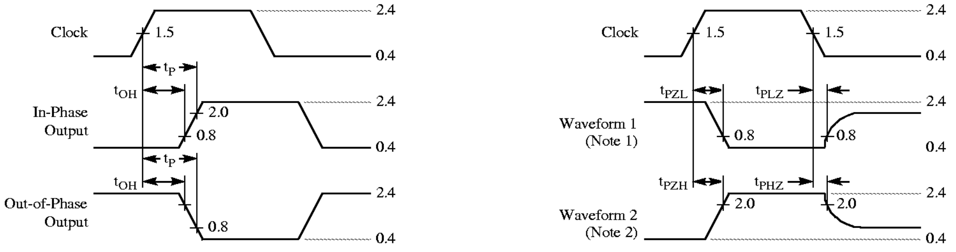


Figure 21. Load Circuit and Parameters



Setup and Hold Times

Pulse Duration



Propagation Delay Times

Enable and Disable Times, 3-State Outputs

1. Waveform 1 is for an output with internal conditions such that the output is low except when disabled by the output control.
2. Waveform 2 is for an output with internal conditions such that the output is high except when disabled by the output control. For t_{PLZ} and t_{PHZ}, V_{OL} and V_{OH} are specified values.

Figure 22. Voltage Waveforms

PIN ASSIGNMENTS

Table 4 and *Table 5* give the VBus and MBus pin assignments for 293-Pin Ceramic PGA Package.

TABLE 4: VBus Pin Assignments ($\overline{\text{CCMODE}}$ = L)

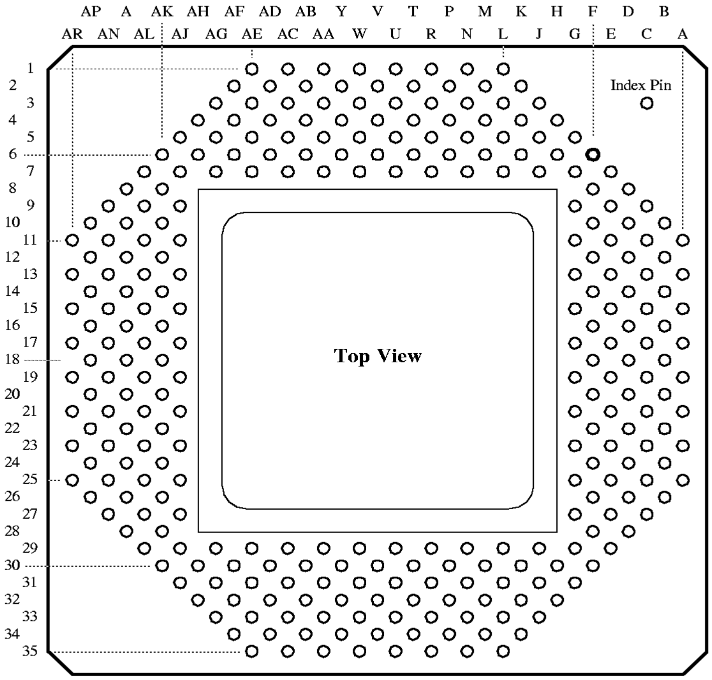
Pin	Signal Name	Pin	Signal Name	Pin	Signal Name	Pin	Signal Name	Pin	Signal Name	Pin	Signal Name
A11	PIPE8	E27	DATA63	L29	DATA53	V30	VSSP	AE29	DPAR2	AL11	BURST
A13	IRL0	E29	DATA60	L31	DATA52	V32	VSSI	AE31	$\overline{\text{WE0}}$	AL13	ADDR00
A15	IRL3	F6	VCCI	L33	DATA51	V34	VCCI	AE33	$\overline{\text{WE1}}$	AL15	ADDR04
A17	ADDR27	F8	DATA31	L35	DATA50	W1	DATA07	AE35	$\overline{\text{WE2}}$	AL17	ADDR08
A19	ADDR31	F10	VSSP	M2	VSSP	W3	DATA06	AF2	DPAR4	AL19	ADDR12
A21	ADDR35	F12	VCCP	M4	VSSC	W5	DATA04	AF4	VCCC	AL21	ADDR17
A23	$\overline{\text{TRST}}$	F14	VSSP	M6	VCCP	W7	DATA05	AF6	VSSP	AL23	ADDR21
A25	spare2	F16	VCCP	M30	VCCP	W29	DATA40	AF30	VSSP	AL25	$\overline{\text{CCHBL}}$
B10	PIPE5	F18	VSSP	M32	VSSC	W31	DATA39	AF32	VCCC	AL27	n/c
B12	VSSP	F20	VCCP	M34	VSSP	W33	DATA41	AF34	DPAR3	AL29	n/c
B14	VCCP	F22	VSSP	N1	DATA16	W35	DATA42	AG3	$\overline{\text{LDST}}$	AM8	VSSC
B16	VSSP	F24	VCCP	N3	DATA17	Y2	VSSP	AG5	$\overline{\text{DEMAP}}$	AM10	VCCC
B18	VCCI	F26	VSSP	N5	DATA19	Y4	VSSC	AG7	$\overline{\text{WGRT}}$	AM12	VSSC
B20	VSSP	F28	DATA61	N7	DATA18	Y6	VCCP	AG29	spare0	AM14	VCCC
B22	VCCP	F30	VCCI	N29	DATA49	Y30	VCCP	AG31	DPAR0	AM16	VSSC
B24	VSSP	G5	DATA29	N31	DATA48	Y32	VSSC	AG33	DPAR1	AM18	VSSI
B26	$\overline{\text{CCMODE}}$	G7	VSSI	N33	DATA47	Y34	VSSP	AH4	VSSC	AM20	VSSC
C3	n/c (index)	G9	DATA30	N35	DATA46	AA1	DATA03	AH6	$\overline{\text{RGRT}}$	AM22	VCCC
C9	PIPE3	G11	PIPE2	P2	VCCP	AA3	DATA02	AH30	$\overline{\text{ADDR20}}$	AM24	VSSC
C11	PIPE7	G13	PIPE6	P4	VCCC	AA5	DATA01	AH32	VSSC	AM26	VCCC
C13	ESB	G15	TDO	P6	VSSP	AA7	$\overline{\text{WE7}}$	AJ5	$\overline{\text{CMDS}}$	AM28	VSSC
C15	IRL2	G17	ADDR25	P30	VSSP	AA29	DATA37	AJ7	VSSI	AN9	$\overline{\text{WEE}}$
C17	ADDR26	G19	ADDR28	P32	VCCC	AA31	DATA35	AJ9	BUSREQ	AN11	$\overline{\text{RD}}$
C19	ADDR30	G21	ADDR33	P34	VCCP	AA33	DATA36	AJ11	$\overline{\text{MEXC}}$	AN13	ADDR01
C21	ADDR34	G23	TCK	R1	DATA15	AA35	DATA38	AJ13	$\overline{\text{WR}}$	AN15	ADDR05
C23	TMS	G25	$\overline{\text{RESET}}$	R3	DATA14	AB2	VCCP	AJ15	ADDR03	AN17	ADDR07
C25	$\overline{\text{SRMTST}}$	G27	DATA62	R5	DATA12	AB4	VCCC	AJ17	ADDR09	AN19	ADDR13
C27	spare1	G29	VSSI	R7	DATA13	AB6	VSSP	AJ19	ADDR11	AN21	ADDR16
D8	VSSC	G31	DATA59	R29	DATA45	AB30	VSSP	AJ21	ADDR18	AN23	ADDR20
D10	VCCC	H4	VSSC	R31	VCCCLK	AB32	VCCC	AJ23	ADDR22	AN25	$\overline{\text{SU}}$
D12	VSSC	H6	DATA28	R33	DATA44	AB34	VCCP	AJ25	$\overline{\text{ERROR}}$	AN27	nu
D14	VCCC	H30	DATA58	R35	VCLK	AC1	DATA00	AJ27	SIZE1	AP10	$\overline{\text{RETRY}}$
D16	VSSC	H32	VSSC	T2	VSSP	AC3	$\overline{\text{WE6}}$	AJ29	VSSI	AP12	VSSP
D18	VSSI	J3	DATA25	T4	VSSC	AC5	$\overline{\text{WE5}}$	AJ31	$\overline{\text{PEND}}$	AP14	VCCP
D20	VSSC	J5	DATA26	T6	VCCP	AC7	DPAR6	AK6	VCCI	AP16	VSSP
D22	VCCC	J7	DATA27	T30	VCCP	AC29	$\overline{\text{WE3}}$	AK8	$\overline{\text{WRDY}}$	AP18	VCCI
D24	VSSC	J29	DATA57	T32	VSSC	AC31	DATA33	AK10	VSSP	AP20	VSSP
D26	VCCC	J31	DATA56	T34	VSSP	AC33	DATA32	AK12	VCCP	AP22	VCCP
D28	VSSC	J33	DATA55	U1	DATA11	AC35	DATA34	AK14	VSSP	AP24	VSSP
E7	PIPE0	K2	DATA24	U3	DATA10	AD2	VSSP	AK16	VCCP	AP26	nu
E9	PIPE1	K4	VCCC	U5	DATA09	AD4	VSSC	AK18	VSSP	AR11	$\overline{\text{OE}}$
E11	PIPE4	K6	VSSP	U7	DATA08	AD6	VCCP	AK20	VCCP	AR13	ADDR02
E13	PIPE9	K30	VSSP	U29	DATA43	AD30	VCCP	AK22	VSSP	AR15	ADDR06
E15	IRL1	K32	VCCC	U31	VSSCLK	AD32	VSSC	AK24	VCCP	AR17	ADDR10
E17	ADDR24	K34	DATA54	U33	$\overline{\text{PLLBYP}}$	AD34	VSSP	AK26	VSSP	AR19	ADDR14
E19	ADDR29	L1	DATA20	U35	VPLLRC	AE1	$\overline{\text{WE4}}$	AK28	SIZE0	AR21	ADDR15
E21	ADDR32	L3	DATA21	V2	VCCI	AE3	DPAR7	AK30	VCCI	AR23	ADDR19
E23	TDI	L5	DATA22	V4	VSSI	AE5	DPAR5	AL7	$\overline{\text{RRDY}}$	AR25	ADDR23
E25	$\overline{\text{TEST}}$	L7	DATA23	V6	VSSP	AE7	$\overline{\text{CSA}}$	AL9	$\overline{\text{ARDY}}$		

TABLE 5: MBus Pin Assignments ($\overline{\text{CCMODE}}$ = H)

Pin	Signal Name	Pin	Signal Name	Pin	Signal Name	Pin	Signal Name	Pin	Signal Name	Pin	Signal Name
A11	PIPE8	E27	MAD63	L29	MAD53	V30	VSSP	AE29	nu	AL11	nu
A13	MIRL0	E29	MAD60	L31	MAD52	V32	VSSI	AE31	nu	AL13	MID0
A15	MIRL3	F6	VCCI	L33	MAD51	V34	VCCI	AE33	nu	AL15	nu
A17	nu	F8	MAD31	L35	MAD50	W1	MAD07	AE35	nu	AL17	nu
A19	nu	F10	VSSP	M2	VSSP	W3	MAD06	AF2	nu	AL19	nu
A21	nu	F12	VCCP	M4	VSSC	W5	MAD04	AF4	VCCC	AL21	nu
A23	$\overline{\text{TRST}}$	F14	VSSP	M6	VCCP	W7	MAD05	AF6	VSSP	AL23	nu
A25	spare2	F16	VCCP	M30	VCCP	W29	MAD40	AF30	VSSP	AL25	nu
B10	PIPE5	F18	VSSP	M32	VSSC	W31	MAD39	AF32	VCCC	AL27	n/c
B12	VSSP	F20	VCCP	M34	VSSP	W33	MAD41	AF34	nu	AL29	n/c
B14	VCCP	F22	VSSP	N1	MAD16	W35	MAD42	AG3	nu	AM8	VSSC
B16	VSSP	F24	VCCP	N3	MAD17	Y2	VSSP	AG5	nu	AM10	VCCC
B18	VCCI	F26	VSSP	N5	MAD19	Y4	VSSC	AG7	$\overline{\text{MBG}}$	AM12	VSSC
B20	VSSP	F28	MAD61	N7	MAD18	Y6	VCCP	AG29	spare0	AM14	VCCC
B22	VCCP	F30	VCCI	N29	MAD49	Y30	VCCP	AG31	nu	AM16	VSSC
B24	VSSP	G5	MAD29	N31	MAD48	Y32	VSSC	AG33	nu	AM18	VSSI
B26	$\overline{\text{CCMODE}}$	G7	VSSI	N33	MAD47	Y34	VSSP	AH4	VSSC	AM20	VSSC
C3	n/c (index)	G9	MAD30	N35	MAD46	AA1	MAD03	AH6	nu	AM22	VCCC
C9	PIPE3	G11	PIPE2	P2	VCCP	AA3	MAD02	AH30	spare3	AM24	VSSC
C11	PIPE7	G13	PIPE6	P4	VCCC	AA5	MAD01	AH32	VSSC	AM26	VCCC
C13	ESB	G15	TDO	P6	VSSP	AA7	nu	AJ5	nu	AM28	VSSC
C15	MIRL2	G17	nu	P30	VSSP	AA29	MAD37	AJ7	VSSI	AN9	$\overline{\text{MBB}}$
C17	nu	G19	nu	P32	VCCC	AA31	MAD35	AJ9	$\overline{\text{MBR}}$	AN11	nu
C19	nu	G21	nu	P34	VCCP	AA33	MAD36	AJ11	$\overline{\text{MERR}}$	AN13	MID1
C21	nu	G23	TCK	R1	MAD15	AA35	MAD38	AJ13	nu	AN15	nu
C23	TMS	G25	$\overline{\text{RSTIN}}$	R3	MAD14	AB2	VCCP	AJ15	MID3	AN17	nu
C25	$\overline{\text{SRMTST}}$	G27	MAD62	R5	MAD12	AB4	VCCC	AJ17	nu	AN19	nu
C27	spare1	G29	VSSI	R7	MAD13	AB6	VSSP	AJ19	nu	AN21	nu
D8	VSSC	G31	MAD59	R29	MAD45	AB30	VSSP	AJ21	nu	AN23	nu
D10	VCCC	H4	VSSC	R31	VCCCLK	AB32	VCCC	AJ23	nu	AN25	nu
D12	VSSC	H6	MAD28	R33	MAD44	AB34	VCCP	AJ25	$\overline{\text{AERR}}$	AN27	$\overline{\text{MIH}}$
D14	VCCC	H30	MAD58	R35	CLK	AC1	MAD00	AJ27	nu	AP10	$\overline{\text{MRTY}}$
D16	VSSC	H32	VSSC	T2	VSSP	AC3	nu	AJ29	VSSI	AP12	VSSP
D18	VSSI	J3	MAD25	T4	VSSC	AC5	nu	AJ31	$\overline{\text{PEND}}$	AP14	VCCP
D20	VSSC	J5	MAD26	T6	VCCP	AC7	DPAR6	AK6	VCCI	AP16	VSSP
D22	VCCC	J7	MAD27	T30	VCCP	AC29	nu	AK8	$\overline{\text{MRDY}}$	AP18	VCCI
D24	VSSC	J29	MAD57	T32	VSSC	AC31	MAD33	AK10	VSSP	AP20	VSSP
D26	VCCC	J31	MAD56	T34	VSSP	AC33	MAD32	AK12	VCCP	AP22	VCCP
D28	VSSC	J33	MAD55	U1	MAD11	AC35	MAD34	AK14	VSSP	AP24	VSSP
E7	PIPE0	K2	MAD24	U3	MAD10	AD2	VSSP	AK16	VCCP	AP26	$\overline{\text{MSH}}$
E9	PIPE1	K4	VCCC	U5	MAD09	AD4	VSSC	AK18	VSSP	AR11	nu
E11	PIPE4	K6	VSSP	U7	MAD08	AD6	VCCP	AK20	VCCP	AR13	MID2
E13	PIPE9	K30	VSSP	U29	MAD43	AD30	VCCP	AK22	VSSP	AR15	nu
E15	MIRL1	K32	VCCC	U31	VSSCLK	AD32	VSSC	AK24	VCCP	AR17	nu
E17	nu	K34	MAD54	U33	$\overline{\text{PLLBYP}}$	AD34	VSSP	AK26	VSSP	AR19	nu
E19	nu	L1	MAD20	U35	VPLLRC	AE1	nu	AK28	nu	AR21	nu
E21	nu	L3	MAD21	V2	VCCI	AE3	nu	AK30	VCCI	AR23	nu
E23	TDI	L5	MAD22	V4	VSSI	AE5	nu	AL7	nu	AR25	nu
E25	$\overline{\text{TEST}}$	L7	MAD23	V6	VSSP	AE7	nu	AL9	$\overline{\text{MAS}}$		

PACKAGE INFORMATION

STP1020A 293-Pin PGA Package Layout

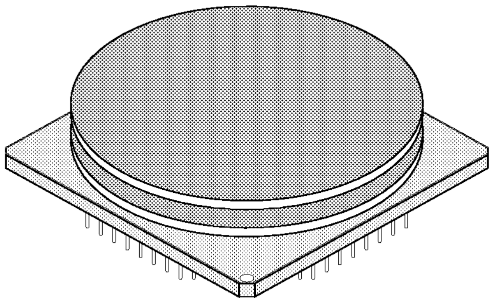


Thermal Resistance vs. Air Flow [1] [2] [3] [4]

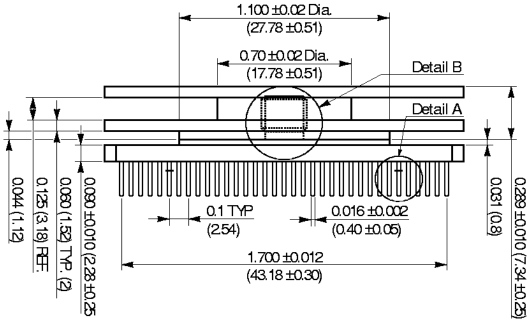
	Air Flow (ft./min)			
Symbol	**100**	**200**	**300**	**500**
$\varnothing_{JA}$ (°C/W)	6.8	4.7	3.7	2.5

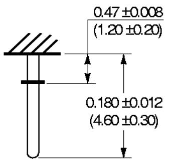
1. T_J can be calculated by: $T_J = T_A + P_d \times \varnothing_{JA}$. Maximum T_J is 85 °C.
2. The above $\varnothing_{JA}$ values are with disk-type heat-sink.
3. $\varnothing_{JC}$ = 0.6°C/W for the 293-pin PGA package. $\varnothing_{JC}$ is package dependent..
4. @50 MHz maximum power = 12.1 Watts.
   @60 MHz maximum power = 13.6 Watts.

293-Pin PGA Package Dimensions

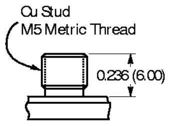


Note: 1. Standard parts do not have heat sinks. Heat Sinks should be ordered separately.
2. Required Torque for mounting heatsink is 2~3 lb-in. 15 lb-in = 1.7 Newton-meter.



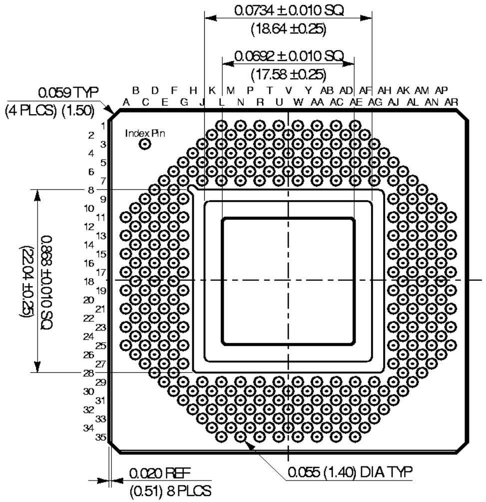


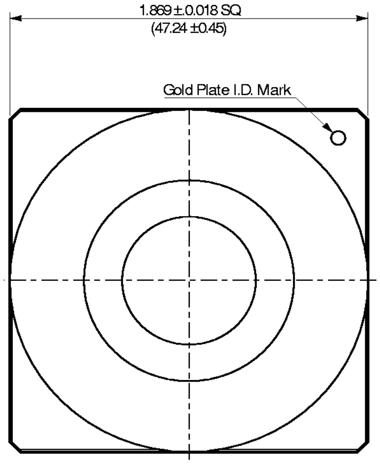
Detail A



Note: Set Calib. Torque Driver to 2-3 lb-in.
15 lb-in = 1.7 Newton-meter

Detail B





Dimensions in Inches (mm)

APPENDIX A

Address Space Identifier (ASI) Support

For each instruction access and each normal data access, the processor appends to the 32-bit memory address an 8-bit address space identifier, or ASI. The ASI encodes whether the processor is in supervisor or user mode, and whether the access is an instruction or data access. Supervisor programs can also make access to program-controlled address spaces by using the previleged Load/ Store Alternate instructions. These previleged Load/ Store instructions, contain implementation dependent ASI codes that specify the address space to be accessed. These address spaces can be the MMU registers, cache controller registers, processor state registers, and other processor or system-dependent values.The following table provides an assignment list of all the STP1020A ASIs.

TABLE 6: Assignments of STP1020A ASIs

ASI	Function	Access	Size
0x00 to 0x01	Reserved	–	–
0x02	Control space access	LD/ST	All
0x03	Reference MMU flush/probe	LD/ST	Single
0x04	Reference MMU registers	LD/ST	Single
0x05	Reserved	–	–
0x06	Reference MMU TLB diagnostics	LD/ST	Single
0x07	Reserved	–	–
0x08	User instruction	LD/ST	All
0x09	Supervisor instruction	LD/ST	All
0x0A	User data	LD/ST	All
0x0B	Supervisor data	LD/ST	All
0x0C	Instruction cache tags	LD/ST	Double
0x0D	Instruction cache data	LD/ST	Double
0x0E	Data cache tags	LD/ST	Double
0x0F	Data cache data	LD/ST	Double
0x10 to 0x1F	Reserved	–	-
0x20 to 0x2F	Reference MMU bypass	LD/ST	All
0x30	Store buffer tags	LD/ST	Double
0x31	Store buffer data	LD/ST	Double
0x32	Store buffer control	LD/ST	Single
0x33 to 0x35	Reserved	–	–
0x36	Instruction cache flash clear	ST	Single
0x37	Data cache flash clear	ST	Single
0x38	MMU breakpoint diagnostics	LD/ST	Double

TABLE 6: Assignments of STP1020A ASIs (Continued)

ASI	Function	Access	Size
0x39	BIST diagnostics	LD/ST	Single
0x3A to 0x3F	Reserved	–	–
0x40 to 0x41	Emulation Temps1-Temps2	LD/ST	Single
0x42 to 0x43	Reserved	–	–
0x44	Emulation data in	LD	Single
0x45	Reserved	–	–
0x46	Emulation data out	ST	Single
0x47	Emulation exit PC	LD/ST	Single
0x48	Emulation exit nPC	LD/ST	Single
0x49	Emulation counter value	LD/ST	Single
0x4A	Emulation counter mode	LD/ST	Single
0x4B	Emulation counter status	LD/ST	Single
0x4C	Action register	LD/ST	Single
0x4D to 0xFF	Unassigned	–	–

APPENDIX B

Register State After Reset

The following table shows the state of the internal registers after a reset.

TABLE 7: Register State After Reset

Register	State After Rest		
	Hardware	**BIST**	**Watchdog**
Floating-Point Queue	Invalidated	Invalidated	Unchanged
Boot Mode (MCNTL.BT)	1 (Boot mode)	1 (Boot mode)	1 (Boot mode)
MMU Enable (MCNTL.EN)	0 (MMU disabled)	0 (MMU disabled)	Unchanged
No Fault (MCNTL.NF)	0 (Faults enabled)	0 (Faults enabled)	Unchanged
Data Cache Enable (MCNTL.DE)	0 (Data Cache disabled)	0 (Data Cache disabled)	Unchanged
Instruction Cache Enable (MCNTL.IE)	0 (Instruction Cache disabled)	0 (Instruction Cache disabled)	Unchanged
Store Buffer enable (MCNTL.SB)	0 (Store Buffer disabled)	0 (Store Buffer disabled)	Unchanged
MBus Mode (MCNTL.MB)	1/0 Depends on $\overline{\text{CCMODE}}$ pin	1/0 Depends on $\overline{\text{CCMODE}}$ pin	1/0 Depends on $\overline{\text{CCMODE}}$ pin
Parity Enable (MCNTL.PE)	0 (Parity disabled)	0 (Parity disabled)	Unchanged
Snoop Enable (MCNTL.SE)	0 (Snooping disabled)	0 (Snooping disabled)	Unchanged
Partial Store Ordering (MCNTL.PSO)	0 (TSO/Strong Ordering)	0 (TSO/Strong Ordering)	Unchanged
Alternate Cacheable (MCNTL.AC)	0 (Noncacheable)	0 (Noncacheable)	Unchanged
Table Walk Cacheable (MCNTL.TC)	0 (Noncacheable)	0 (Noncacheable)	Unchanged
Error Mode (MFSR.EM)	0 (Not an error mode, or watchdog reset)	0 (Not an error mode, or watchdog reset)	1 (A watchdog reset)
TLB Lock Bits	0 (All TLB lock bits cleared)	0 (All TLB lock bits cleared)	0 (All TLB lock bits cleared)
Multiple Instruction Mode (ACTION.MIX)	0 (Single Instruction Execution)	0 (Single Instruction Execution)	0 (Single Instruction Execution)
Breakpoints (MDIAG)	0 (All breakpoints disabled)	0 (All breakpoints disabled)	0 (All breakpoints disabled)
Program Counter	0 (PC=0x0, nPC=0x4)	0 (PC=0x0, nPC=0x4)	0 (PC=0x0, nPC=0x4)
BIST Status	00 (No BIST since reset)	01 or 10 (BIST run since reset)	Not Affected
Store Buffer Tags	Valid bits cleared	Valid bits cleared	Valid bits cleared
Store Buffer Contents	Contents Uninitialized	Contents Uninitialized	Contents Unchanged
Data Cache	Contents Uninitialized	Contents Uninitialized	Contents Unchanged

TABLE 7: Register State After Reset (Continued)

Register	State After Rest		
	Hardware	BIST	Watchdog
Instruction Cache	Contents Uninitialized	Contents Uninitialized	Contents Unchanged
Register File	Contents Uninitialized	Contents Uninitialized	Contents Unchanged
Processor Status Register (PSR)	S=1, ET=0, EC=0, Ver=0, Impl=4, PSR current window pointer Uninitialized	S=1, ET=0, EC=0, Ver=0, Impl=4, PSR current window pointer Uninitialized	S=1, ET=0, EC=0, Ver=0, Impl=4, PSR current window pointer Unchanged
Window Invalid Mask (WIM)	Uninitialized	Uninitialized	Unchanged
Fault Status Register (MFSR)	Uninitialized (except for MFSR error mode bit)	Uninitialized (except for MFSR error mode bit)	Unchanged (except for MFSR error mode bit)
Shadow Fault Status Register (MSFSR)	Uninitialized	Uninitialized	Unchanged
Emulation Facilities	Disabled	Disabled	Unchanged

APPENDIX C

Boundary Scan Description

The following table shows the ordering of the STP1020A boundary scan chain. Pin 0 is connected to TDI, while pin 291 is connected to TDO.

TABLE 8: The STP1020A Boundary Scan Bit Definition

Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
0	reset_in	42	data44_in	84	addr20*_out	126	addr08_in	168	wrdy_out	210	data10_in	252	data31_in
1	test_in	43	data44_out	85	size1_out	127	addr08_out	169	rrdy_in	211	data10_out	253	data31_out
2	srmtst_in	44	vck_in	86	size0_out	128	dpboe_out	170	wgrt_in	212	data11_in	254	pipe00-out
3	ccrdy_in	45	pllbyp_in	87	owner_in	129	addr07_in	171	rgrt_in	213	data11_out	255	pipe01_out
4	data63_in	46	data43_in	88	owner_out	130	addr07_out	172	cmds_in	214	data12_in	256	pipe02_out
5	data63_out	47	data43_out	89	shared_in	131	srboe_out	173	cmds_out	215	data12_out	257	pipe03_out
6	data62_in	48	data42-in	90	shared_out	132	addr06_in	174	demap_in	216	data13_in	258	pipe04_out
7	data62_out	49	data42_out	91	error_out	133	addr06_out	175	demap_out	217	data13_out	259	pipe05_out
8	data 61_in	50	data41_in	92	cchbl_out	134	allboe_out	176	csa_out	218	data14_in	260	pipe06_out
9	data61_out	51	data41_out	93	su_out	135	erboe_out	177	ldst_out	219	data14_out	261	pipe07_out
10	data60_in	52	data40_in	94	addr23_in	136	addr05_in	178	dpar4_in	220	data15_in	262	pipe08_out
11	data60_out	53	data40_out	95	addr23_out	137	addr05_out	179	dpar4_out	221	data15_out	263	pipe09_out
12	data59_in	54	data39_in	96	addr22_in	138	addr04_in	180	dpar5_in	222	data16_in	264	irl0_in
13	data59_out	55	data39_out	97	addr22_out	139	addr04_out	181	dpar5_out	223	data16_out	265	irl1_in
14	data58_in	56	data38_in	98	addr21_in	140	snboe_out	182	dpar6_in	224	data17_in	266	irl2_in
15	data58_out	57	data38_out	99	addr21_out	141	addr03_in	183	dpar6_out	225	data17_out	267	irl3_in
16	data57_in	58	data37_in	100	addr20_in	142	addr03_out	184	dpar7_in	226	data18_in	268	addr24_in
17	data57_out	59	data37_out	101	addr20_out	143	mbboe_out	185	dpar7_out	227	data18_out	269	addr24_out
18	data56_in	60	data36_in	102	addr19_in	144	addr02_in	186	we4_out	228	data19_in	270	addr25_in
19	data56_out	61	data36-out	103	addr19_out	145	addr02_out	187	we5_out	229	data19_out	271	addr25_out
20	data55_in	62	data35_in	104	addr18_in	146	mrboe_out	188	we6_out	230	data20_in	272	addr26_in
21	data55_out	63	data35_out	105	addr18_out	147	stboe_out	189	we7_out	231	data20_out	273	addr26_out
22	data54_in	64	data34_in	106	addr17_in	148	addr01_in	190	data0_in	232	data21_in	274	addr27_in
23	data54_out	65	data34_out	107	addr17_out	149	addr01_out	191	data0_out	233	data21_out	275	addr27_out
24	data53_in	66	data33_in	108	addr16_in	150	addr00_in	192	data1_in	234	data22_in	276	addr28_in
25	data53_out	67	data33_out	109	addr16_out	151	addr00_out	193	data1_out	235	data22_out	277	addr28_out
26	data52_in	68	data32_in	110	addr15_in	152	oe_in	194	data2_in	236	data23_in	278	addr29_in
27	data52_out	69	data32_out	111	addr15_out	153	oe_out	195	data2_out	237	data23_out	279	addr29_out
28	data51_in	70	we3_out	112	addr14_in	154	wr_in	196	data3_in	238	data24_in	280	addr30_in
29	data51_out	71	we2_out	113	addr14_out	155	wr_out	197	data3_out	239	data24_out	281	addr30_out
30	data50_in	72	we1_out	114	addr13_in	156	oeboe_out	198	data4_in	240	data25_in	282	addr31_in
31	data50_out	73	we0_out	115	addr13_out	157	rd_in	199	data4_out	241	data25_out	283	addr31_out
32	data49_in	74	dpar3_in	116	addr12_in	158	rd_out	200	data5_in	242	data26_in	284	addr32_in
33	data49_out	75	dpar3_out	11 7	addr12_out	159	burst_out	201	data5_out	243	data26_out	285	addr32_out
34	data48_in	76	dpar2_in	118	daboe_out	160	retry_in	202	data6_in	244	data27_in	286	addr33_in
35	data48_out	77	dpar2_out	119	addr11_in	161	wee_in	203	data6_out	245	data27_out	287	addr33_out
36	data47_in	78	dpar1_in	120	addr11_out	162	wee_out	204	data7_in	246	data28_in	288	addr34_in
37	data47_out	79	dpar_out	121	addr10_in	163	mexc_in	205	data7_out	247	data28_out	289	addr34_out
38	data46_in	80	dapr0_in	122	addr10_out	164	ardy_in	206	data8_in	248	data29_in	290	addr35_in
39	data46_out	81	dpar0_out	123	adboe_out	165	ardy_out	207	data8_out	249	data29_out	291	addr35_out
40	data45_in	82	pend_in	124	addr09_in	166	busreq_out	208	data9_in	250	data30_in		
41	data45_out	83	addr20*_in	125	addr09_out	167	wrdy_in	209	data9_out	251	data30_out		

STP1020A ERRATA

This errata applies to the following products:

- STP1020APGA-50 SuperSPARC CPU
- STP1020APGA-60 SuperSPARC CPU
- STP5010AMBUS-50 SuperSPARC Module
- STP5011BMBUS-50 SuperSPARC Module
- STP5022MBUS-50 SuperSPARC Module
- STP5011BMBUS-60 SuperSPARC Module
- STP5012BMBUS-60 SuperSPARC Module

Errata Categories:

System Design

These errata are deviations from the product specification that a system design engineer needs to be aware of. These would include bus protocol, mechanical and thermal errata.

Performance

These errata result only in a loss in performance. These errata do not cause functional errors.

Operating System

These errata only occur with privileged-mode or supervisor-mode instructions. Workarounds should be implemented in the operating system or boot PROM. The target audience is the operating system or system firmware engineer.

Application Software

These errata can occur with nonprivileged-mode or user-mode instructions. The target audience is anyone writing software for a platform using this part, including the end-user, independent software vendors, and operating system engineers.

Each erratum will have the following rating scale at the top of each description:

System Design	Performance	Operating System	Application

Erratum: Branch Always Instruction Bubble:

System Design	Performance	Operating System	Application

Description: (Ref. # 133394)

This erratum affects performance only. It induces an unnecessary pipeline bubble in common, unoptimized code for each iteration of an in-cache “while” loop. Code of the following form will expose the problem:

```
loop:
     . . .
     bicc exit_while_loop
     nop
     ba loop
     nop
```

There will always be a pipeline bubble after the Branch Always instruction’s nop before issuing the instruction at label “loop”.

Workaround:

This erratum can be avoided by optimizing the code. It is very unlikely that optimizing compilers would generate the code given above.

Erratum: Extraneous STBAR

System Design	Performance	Operating System	Application

Description: (Ref. # 148094)

In systems with external cache and using the PSO memory model, an extra STBAR may be executed if the corresponding store operation was retried. A very slight performance loss may be observed due to execution of an extra STBAR. Functionally there is no problem. The scenario is as follows:

When a store moves from the store buffer to the pins, a STBAR that is associated with that store gets executed by the processor. The effect of this STBAR is to prevent any subsequent stores from happening until PEND_ is deasserted. Now, if this store receives a RETRY, the store will be pushed back into the store buffer, together with its associated STBAR. However, the STBAR that is now in effect is not cleared. The result is that an extra STBAR has been inserted, and the store that needs to be retried will not appear again on the pins until PEND_ is deasserted. The original STBAR is still associated with the store, and will be recognized again when that store is reissued.

Workaround:

Whenever the logic communicating with the processor (Multi-Cache Controller or otherwise) forces a RETRY on a store, it needs to deassert PEND_ signal. The STP1090, STP1090A and STP1091 MXCCs deassert PEND_ on RETRYs as part of their standard protocol implementation.

Erratum: MFAR May Log Wrong Address During Bus Error

System Design	Performance	▨ Operating System	Application

Description: (Ref. # 148394)

If a bus error is reported while a table walk is pending, the MFAR register will record the value of the pending table walk address (in this case the value of the next Program Counter). The kernel will get confused as it checks for the address of the fault.

Workaround:

The exception handler can check to see if the contents of MFAR are equal to the next program counter. If they are equal, use the contents of the PC that is stored in the local register (stored automatically by the CPU) as the address that is causing the bus error.

Erratum: Snoop and Writes to ASI 0xe-0xf

System Design	Performance	▨ Operating System	Application

Description: (Ref. # 148494)

In MBus systems, when the processor is storing data into ASI 0xe or 0xf (data cache tags or data cache data), if a snoop comes in and hits in the exact line of the data cache which is being accessed by the ASI then the data in the tags or data fields can become corrupted.

Workaround:

Following an ASI write to 0xe or 0xf, a read to the same address should be performed to check if tags or data is corrupted.

Erratum: Branch to its Own Delay Slot

System Design	Performance	Operating System	▨ Application

Description: (Ref. # 108794)

If the target of a branch instruction is the delay slot of that branch instruction the processor may stall.

Workaround:

A compiler would never generate this code sequence. Hand-generated code should never use this code sequence.

Erratum: FSR[UFA] Bit Setting for FMUL

System Design	Performance	Operating System	[✓ Application]

Description: (Ref. # 118894)

If a FMULS or FMULD instruction that would result in both an inexact and an underflow result is immediately followed by a FMULS or FMULD instruction that has a normal result, then the UFA bit of the floating-point status register will not get set, even though the UFC bit was set for the first FMULS/ FMULD. If the two multiply instructions are not executed consecutively the UFA bit will be set correctly.

This erratum is only a problem for applications that explicitly examine the UFA status bit. It does not affect the resulting data.

Work around:

This erratum can easily be avoided by using the NXA instead of the UFA bit of the FSR. Another work around for this erratum is to enable the TEM bits (Trap Enable Mask) TEM.NXM or the TEM.UFM. When either one of these bits is set an exception will be taken, thus avoiding the execution of two consecutive fmuls/ fmuld.

Erratum: Simultaneous Snoop & STA to ASI 0x40-0x4C

System Design	Performance	[✓ Operating System]	Application

Description: (Ref. # 108994)

In all direct MBus systems in a multiprocessor configuration, a snoop to the processor that is currently in the middle of a STA instruction to ASI addresses 0x40-0x4c could result in the cache data from that snoop hit ending up in that ASI Register. ASI address 0x4c is the Action Register and ASI addresses 0x40-0x4b are either the Emulation or Breakpoints registers. In such a situation, the MIX bit (Super-Scaler Enable) in the Action Register could be set or cleared regardless of the previous data. This erratum cannot affect systems where the SuperSPARC processor is used with a Multi-Cache Controller, since there are no snoop hits in writeback systems.

Clearing of the MIX bit affects performance only, since the processor will proceed to run in single scalar mode for the rest of the current context. If either breakpoints or emulation is being used then there may be other problems relating to breakpoints during emulation.

Work around:

The software that writes to ASI 0x40-0x4c should rewrite the register if it has the wrong value. Once the correct value is in the register it will not be changed.

Erratum: 40 MHz MBus Output Hold times

[✓ **System Design**]	**Performance**	**Operating System**	**Application**

Description: (Ref. # 100094)

The output hold times (t_{oh}) for 40 MHz MBus operation shown on page 38 of the datasheet for the signals MAD, MC1, MBR, MSH and AERR are incorrect. The correct output hold times for these signals is 1.5 ns.

Workaround:

The system design must guarantee that components interfacing to the STP1020A part in direct MBus applications must support these output hold times.

ORDERING INFORMATION [1] [2]

Part Number	Speed	Description
STP1020APGA-50	50 MHz	Production Parts
STP1020APGA-60	60 MHz	Production Parts
STP1020HS	N/A	Disk-Fin Type Heat Sink

1. Standard parts do not have heat sinks. Heat sinks should be ordered separately.
2. Engineering Samples are marked -ES-.