Welcome to E-XFL.COM

Understanding Embedded - DSP (Digital Signal Processors)

Embedded - DSP (Digital Signal Processors) are specialized microprocessors designed to perform complex mathematical computations on digital signals in real-time. Unlike general-purpose processors, DSPs are optimized for high-speed numeric processing tasks, making them ideal for applications that require efficient and precise manipulation of digital data. These processors are fundamental in converting and processing signals in various forms, including audio, video, and communication signals, ensuring that data is accurately interpreted and utilized in embedded systems.

Applications of Embedded - DSP (Digital Signal Processors)

Details	
Product Status	Active
Type	Fixed/Floating Point
Interface	Host Interface, Link Port, Multi-Processor
Clock Rate	250MHz
Non-Volatile Memory	External
On-Chip RAM	768kB
Voltage - I/O	3.30V
Voltage - Core	1.20V
Operating Temperature	-40°C ~ 85°C (TC)
Mounting Type	Surface Mount
Package / Case	484-BFBGA
Supplier Device Package	484-PBGA (19x19)
Purchase URL	https://www.e-xfl.com/product-detail/analog-devices/adsp-ts101sab2z000

ADSP-TS101S* PRODUCT PAGE QUICK LINKS

Last Content Update: 02/23/2017

COMPARABLE PARTS

View a parametric search of comparable parts.

EVALUATION KITS

- EZ-KIT Lite Evaluation Kit for ADSP-TS201S Processor
- USB-Based Emulator and High Performance USB-Based Emulator

DOCUMENTATION

Application Notes

- AN-911: A Detailed Guide to Powering the TigerSHARC Processors
- EE-104: Setting Up Streams with the VisualDSP Debugger
- EE-110: A Quick Primer on ELF and DWARF File Formats
- EE-120: Interfacing Assembly Language Programs to C
- EE-126: The ABCs of SDRAMemories
- EE-128: DSP in C++: Calling Assembly Class Member Functions From C++
- EE-143: Understanding DMA on the ADSP-TS101
- EE-147: Tuning C Source Code for the TigerSHARC® DSP Compiler
- EE-157: Explaining the Branch Target Buffer on the ADSP-TS101
- EE-159: Initializing DSP System & Control Registers From C and C++
- EE-167: Introduction to TigerSHARC® Multiprocessor Systems Using VisualDSP++™
- EE-169: Estimating Power For The ADSP-TS101S
- EE-174: ADSP-TS101S TigerSHARC® Processor Boot Loader Kernels Operation
- EE-175: Emulator and Evaluation Hardware Troubleshooting Guide for VisualDSP++ Users
- EE-176: Hardware Design Checklist For ADSP-TS101S TigerSHARC® Processors
- EE-178: The ADSP-TS101S TigerSHARC® On-chip SDRAM Controller
- EE-202: Using the Expert Linker for Multiprocessor LDFs
- EE-217: Updating the ADSP-TS101S TigerSHARC® EZ-KIT Lite™ Firmware
- EE-261: Understanding Jitter Requirements of PLL-Based Processors
- EE-263: Parallel Implementation of Fixed-Point FFTs on TigerSHARC® Processors
- EE-330: Windows Vista Compatibility in VisualDSP++ 5.0 Development Tools
- EE-332: Cycle Counting and Profiling
- EE-356: Emulator and Evaluation Hardware Troubleshooting Guide for CCES Users
- EE-68: Analog Devices JTAG Emulation Technical Reference

Data Sheet

- ADSP-TS101S: TigerSHARC Embedded Processor, 300 MHz, 6 Mbits, Data Sheet

Evaluation Kit Manuals

- ADSP-TS101S EZ-KIT Lite® Manual

Integrated Circuit Anomalies

- ADSP-TS101S TigerSHARC Anomaly List for Revision(s) 0.2, 0.4

Processor Manuals

- ADSP-TS101 TigerSHARC Processor Hardware Reference
- ADSP-TS101 TigerSHARC Processor Programming Reference
- TigerSHARC Processors: Manuals

Product Highlight

- General-Purpose TigerSHARC Processor Product Brief

Software Manuals

- VisualDSP++® 5.0 Assembler and Preprocessor Manual
- VisualDSP++® 5.0 C/C++ Compiler and Library Manual for TigerSHARC Processors
- VisualDSP++® 5.0 Kernel (VDK) Users Guide
- VisualDSP++® 5.0 Licensing Guide
- VisualDSP++® 5.0 Linker and Utilities Manual
- VisualDSP++® 5.0 Loader and Utilities Manual
- VisualDSP++® 5.0 Product Release Bulletin
- VisualDSP++® 5.0 Quick Installation Reference Card
- VisualDSP++® 5.0 Users Guide

SOFTWARE AND SYSTEMS REQUIREMENTS

- TigerSHARC Evaluation Kits

TOOLS AND SIMULATIONS

- ADSP-TS101 TigerSHARC BSDL File 19x19mm PBGA Package for Revision 0.4, (11/2006)
- ADSP-TS101 TigerSHARC BSDL File 27x27mm PBGA Package for Revision 0.4, (11/2006)
- ADSP-TS101: 19x19mm PBGA Silicon Revision 0.0 and 0.1 [BSDL Original File], 02/08/2001
- ADSP-TS101: 19x19mm PBGA Package Silicon Revision 0.2, [BSDL Original File], 09/09/2003
- ADSP-TS101: 27x27 PBGA Package Siicon Revision 0.2, [BSDL Original File], 09/09/2003
- ADSP-TS101: 27x27 PBGA Package Silicon Revision 0.0 and 0.1 [BSDL Original File], 10/12/2001
- Designing with BGA
- TigerSHARC Processors: Software and Tools
- ADSP-TS101S IBIS Datafile BGA Package

REFERENCE MATERIALS

Product Selection Guide

- ADI Complementary Parts Guide - Supervisory Devices and DSP Processors

Technical Articles

- A Software Solution for Chip Rate Processing in CDMA Wireless Infrastructure
- ADSP-TS101S MP System Simulation and Analysis
- Continuous Real-Time Signal Processing -- Comparing TigerSHARC and PowerPC Via Continuous cFFTs
- Rethinking Base Station, Baseband Processing for Wireless Communication
- SHARC Bites Back - The Memory Inside: TigerSHARC Swallows Its DRAM

White Papers

- ADSP-TS101S MP System Simulation and Analysis

DESIGN RESOURCES

- ADSP-TS101S Material Declaration
- PCN-PDN Information
- Quality And Reliability
- Symbols and Footprints

DISCUSSIONS

View all ADSP-TS101S EngineerZone Discussions.

SAMPLE AND BUY

Visit the product page to see pricing options.

The ADSP-TS101S, in most cases, has a two-cycle arithmetic execution pipeline that is fully interlocked, so whenever a computation result is unavailable for another operation dependent on it, the DSP automatically inserts one or more stall cycles as needed. Efficient programming with dependency-free instructions can eliminate most computational and memory transfer data dependencies.

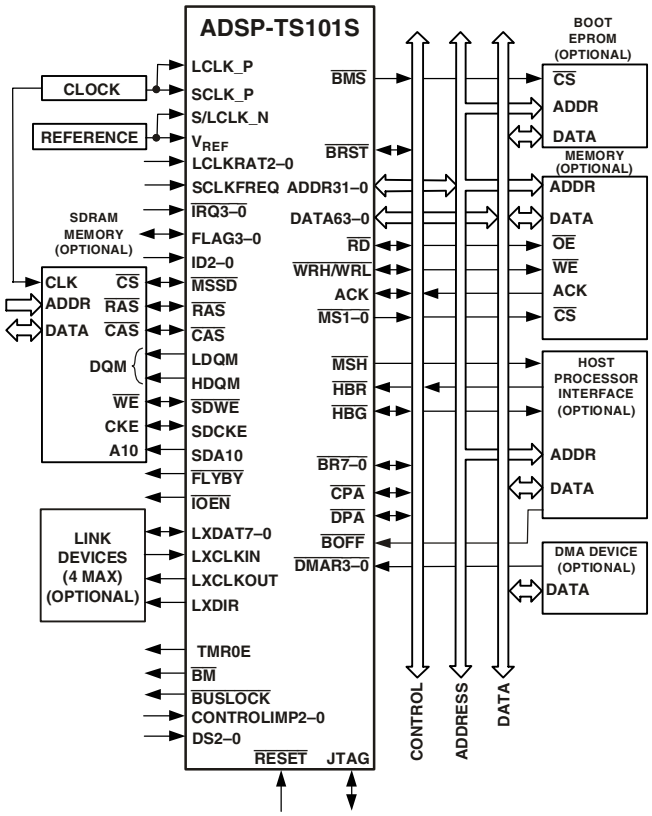

Figure 2. Single-Processor System with External SDRAM

In addition, the ADSP-TS101S supports SIMD operations two ways—SIMD compute blocks and SIMD computations. The programmer can direct both compute blocks to operate on the same data (broadcast distribution) or on different data (merged distribution). In addition, each compute block can execute four 16-bit or eight 8-bit SIMD computations in parallel.

DUAL COMPUTE BLOCKS

The ADSP-TS101S has compute blocks that can execute computations either independently or together as a SIMD engine. The DSP can issue up to two compute instructions per compute block each cycle, instructing the ALU, multiplier, or shifter to perform independent, simultaneous operations.

The compute blocks are referred to as X and Y in assembly syntax, and each block contains three computational units—an ALU, a multiplier, a 64-bit shifter, and a 32-word register file.

- Register file—each compute block has a multiported 32-word, fully orthogonal register file used for transferring data between the computation units and data buses and for storing intermediate results. Instructions can access the registers in the register file individually (word aligned), or in sets of two (dual aligned) or four (quad aligned).
- ALU—the ALU performs a standard set of arithmetic operations in both fixed- and floating-point formats. It also performs logic operations.
- Multiplier—the multiplier performs both fixed- and floating-point multiplication and fixed-point multiply and accumulate.
- Shifter—the 64-bit shifter performs logical and arithmetic shifts, bit and bit stream manipulation, and field deposit and extraction operations.
- Accelerator—128-bit unit for trellis decoding (for example, Viterbi and turbo decoders) and complex correlations for communication applications.

Using these features, the compute blocks can:

- Provide 8 MACs per cycle peak and 7.1 MACs per cycle sustained 16-bit performance and provide 2 MACs per cycle peak and 1.8 MACs per cycle sustained 32-bit performance (based on FIR)
- Execute six single-precision, floating-point or execute 24 fixed-point (16-bit) operations per cycle, providing 1,800 MFLOPS or 7.3 GOPS performance
- Perform two complex 16-bit MACs per cycle
- Execute eight trellis butterflies in one cycle

DATA ALIGNMENT BUFFER (DAB)

The DAB is a quad word FIFO that enables loading of quad word data from nonaligned addresses. Normally, load instructions must be aligned to their data size so that quad words are loaded from a quad-aligned address. Using the DAB significantly improves the efficiency of some applications, such as FIR filters.

DUAL INTEGER ALUS (IALUS)

The ADSP-TS101S has two IALUs that provide powerful address generation capabilities and perform many general-purpose integer operations. Each of the IALUs:

- Provides memory addresses for data and update pointers
- Supports circular buffering and bit-reverse addressing
- Performs general-purpose integer operations, increasing programming flexibility
- Includes a 31-word register file for each IALU

As address generators, the IALUs perform immediate or indirect (pre- and post-modify) addressing. They perform modulus and bit-reverse operations with no constraints placed on memory addresses for the modulus data buffer placement. Each IALU can specify either a single, dual, or quad word access from memory.

The IALUs have hardware support for circular buffers, bit reverse, and zero-overhead looping. Circular buffers facilitate efficient programming of delay lines and other data structures required in digital signal processing, and they are commonly used in digital filters and Fourier transforms. Each IALU provides registers for four circular buffers, so applications can set up a total of eight circular buffers. The IALUs handle address pointer wraparound automatically, reducing overhead, increasing performance, and simplifying implementation. Circular buffers can start and end at any memory location.

Because the IALU's computational pipeline is one cycle deep, in most cases, integer results are available in the next cycle. Hardware (register dependency check) causes a stall if a result is unavailable in a given cycle.

PROGRAM SEQUENCER

The ADSP-TS101S processor's program sequencer supports:

- A fully interruptible programming model with flexible programming in assembly and C/C++ languages; handles hardware interrupts with high throughput and no aborted instruction cycles.
- An eight-cycle instruction pipeline—three-cycle fetch pipe and five-cycle execution pipe—with computation results available two cycles after operands are available.
- The supply of instruction fetch memory addresses; the sequencer's instruction alignment buffer (IAB) caches up to five fetched instruction lines waiting to execute; the program sequencer extracts an instruction line from the IAB and distributes it to the appropriate core component for execution.
- The management of program structures and determination of program flow according to JUMP, CALL, RTI, RTS instructions, loop structures, conditions, interrupts, and software exceptions.
- Branch prediction and a 128-entry branch target buffer (BTB) to reduce branch delays for efficient execution of conditional and unconditional branch instructions and zero-overhead looping; correctly predicted branches that are taken occur with zero-to-two overhead cycles, overcoming the three-to-six stage branch penalty.
- Compact code without the requirement to align code in memory; the IAB handles alignment.

Interrupt Controller

The DSP supports nested and non-nested interrupts. Each interrupt type has a register in the interrupt vector table. Also, each has a bit in both the interrupt latch register and the interrupt mask register. All interrupts are fixed as either level sensitive or edge sensitive, except the $\overline{\text{IRQ3–0}}$ hardware interrupts, which are programmable.

The DSP distinguishes between hardware interrupts and software exceptions, handling them differently. When a software exception occurs, the DSP aborts all other instructions in the instruction pipe. When a hardware interrupt occurs, the DSP continues to execute instructions already in the instruction pipe.

Flexible Instruction Set

The 128-bit instruction line, which can contain up to four 32-bit instructions, accommodates a variety of parallel operations for concise programming. For example, one instruction line can direct the DSP to conditionally execute a multiply, an add, and a subtract in both computation blocks while it also branches to another location in the program. Some key features of the instruction set include:

- Enhanced instructions for communications infrastructure to govern trellis decoding (for example, Viterbi and turbo decoders) and despreading via complex correlations
- Algebraic assembly language syntax
- Direct support for all DSP, imaging, and video arithmetic types, eliminating hardware modes
- Branch prediction encoded in instruction, enables zero-overhead loops
- Parallelism encoded in instruction line
- Conditional execution optional for all instructions
- User-defined, programmable partitioning between program and data memory

ON-CHIP SRAM MEMORY

The ADSP-TS101S has 6M bits of on-chip SRAM memory, divided into three blocks of 2M bits (64K words × 32 bits). Each block—M0, M1, and M2—can store program, data, or both, so applications can configure memory to suit specific needs. Placing program instructions and data in different memory blocks, however, enables the DSP to access data while performing an instruction fetch.

The DSP's internal and external memory (Figure 3) is organized into a unified memory map, which defines the location (address) of all elements in the system. The memory map is divided into four memory areas—host space, external memory, multiprocessor space, and internal memory—and each memory space, except host memory, is subdivided into smaller memory spaces.

Each internal memory block connects to one of the 128-bit-wide internal buses—block M0 to bus MD0, block M1 to bus MD1, and block M2 to bus MD2—enabling the DSP to perform three memory transfers in the same cycle. The DSP's internal bus architecture provides a total memory bandwidth of 14.4G bytes per second, enabling the core and I/O to access eight 32-bit data words (256 bits) and four 32-bit instructions each cycle. The DSP's flexible memory structure enables:

- DSP core and I/O access of different memory blocks in the same cycle
- DSP core access of all three memory blocks in parallel—one instruction and two data accesses
- Programmable partitioning of program and data memory
- Program access of all memory as 32-, 64-, or 128-bit words—16-bit words with the DAB
- Complete context switch in less than 20 cycles (66 ns)

Host Interface

The ADSP-TS101S provides an easy and configurable interface between its external bus and host processors through the external port. To accommodate a variety of host processors, the host interface supports pipelined or slow protocols for accesses of the host as slave. Each protocol has programmable transmission parameters, such as idle cycles, pipe depth, and internal wait cycles.

The host interface supports burst transactions initiated by a host processor. After the host issues the starting address of the burst and asserts the $\overline{\text{BRST}}$ signal, the DSP increments the address internally while the host continues to assert $\overline{\text{BRST}}$.

The host interface provides a deadlock recovery mechanism that enables a host to recover from deadlock situations involving the DSP. The $\overline{\text{BOFF}}$ signal provides the deadlock recovery mechanism. When the host asserts $\overline{\text{BOFF}}$, the DSP backs off the current transaction and asserts $\overline{\text{HBG}}$ and relinquishes the external bus.

The host can directly read or write the internal memory of the ADSP-TS101S, and it can access most of the DSP registers, including DMA control (TCB) registers. Vector interrupts support efficient execution of host commands.

Multiprocessor Interface

The ADSP-TS101S offers powerful features tailored to multiprocessing DSP systems through the external port and link ports. This multiprocessing capability provides highest bandwidth for interprocessor communication, including:

- Up to eight DSPs on a common bus
- On-chip arbitration for glueless multiprocessing
- Link ports for point-to-point communication

The external port and link ports provide integrated, glueless multiprocessing support.

The external port supports a unified address space (see Figure 3) that enables direct interprocessor accesses of each ADSP-TS101S processor's internal memory and registers. The DSP's on-chip distributed bus arbitration logic provides simple, glueless connection for systems containing up to eight ADSP-TS101S processors and a host processor. Bus arbitration has a rotating priority. Bus lock supports indivisible read-modify-write sequences for semaphores. A bus fairness feature prevents one DSP from holding the external bus too long.

The DSP's four link ports provide a second path for interprocessor communications with throughput of 1G bytes per second. The cluster bus provides 800M bytes per second throughput—with a total of 1.8G bytes per second interprocessor bandwidth.

SDRAM Controller

The SDRAM controller controls the ADSP-TS101S processor's transfers of data to and from synchronous DRAM (SDRAM). The throughput is 32 or 64 bits per SCLK cycle using the external port and SDRAM control pins.

The SDRAM interface provides a glueless interface with standard SDRAMs—16M bit, 64M bit, 128M bit, and 256M bit. The DSP directly supports a maximum of 64M words × 32 bits of SDRAM. The SDRAM interface is mapped in external memory in the DSP's unified memory map.

EPROM Interface

The ADSP-TS101S can be configured to boot from external 8-bit EPROM at reset through the external port. An automatic process (which follows reset) loads a program from the EPROM into internal memory. This process uses 16 wait cycles for each read access. During booting, the $\overline{\text{BMS}}$ pin functions as the EPROM chip select signal. The EPROM boot procedure uses DMA Channel 0, which packs the bytes into 32-bit instructions. Applications can also access the EPROM (write flash memories) during normal operation through DMA.

The EPROM or flash memory interface is not mapped in the DSP's unified memory map. It is a byte address space limited to a maximum of 16M bytes (24 address bits). The EPROM or flash memory interface can be used after boot via a DMA.

DMA CONTROLLER

The ADSP-TS101S processor's on-chip DMA controller, with 14 DMA channels, provides zero-overhead data transfers without processor intervention. The DMA controller operates independently and invisibly to the DSP's core, enabling DMA operations to occur while the DSP's core continues to execute program instructions. The DMA controller performs DMA transfers between:

- Internal memory and external memory and memory-mapped peripherals
- Internal memory of other DSPs on a common bus, a host processor, or link port I/O
- External memory and external peripherals or link port I/O
- External bus master and internal memory or link port I/O

The DMA controller provides a number of additional features.

The DMA controller supports flyby transfers. Flyby operations only occur through the external port (DMA Channel 0) and do not involve the DSP's core. The DMA controller acts as a conduit to transfer data from one external device to another through external memory. During a transaction, the DSP:

- Relinquishes the external data bus
- Outputs addresses, memory selects ($\overline{\text{MS1–0}}$, $\overline{\text{MSSD}}$, $\overline{\text{RAS}}$, $\overline{\text{CAS}}$, and $\overline{\text{SDWE}}$) and the $\overline{\text{FLYBY}}$, $\overline{\text{IOEN}}$, and $\overline{\text{RD}}$/$\overline{\text{WR}}$ strobes
- Responds to ACK

DMA chaining is also supported by the DMA controller. DMA chaining operations enable applications to automatically link one DMA transfer sequence to another for continuous transmission. The sequences can occur over different DMA channels and have different transmission attributes.

The DMA controller also supports two-dimensional transfers. The DMA controller can access and transfer two-dimensional memory arrays on any DMA transmit or receive channel. These transfers are implemented with index, count, and modify registers for both the X and Y dimensions.

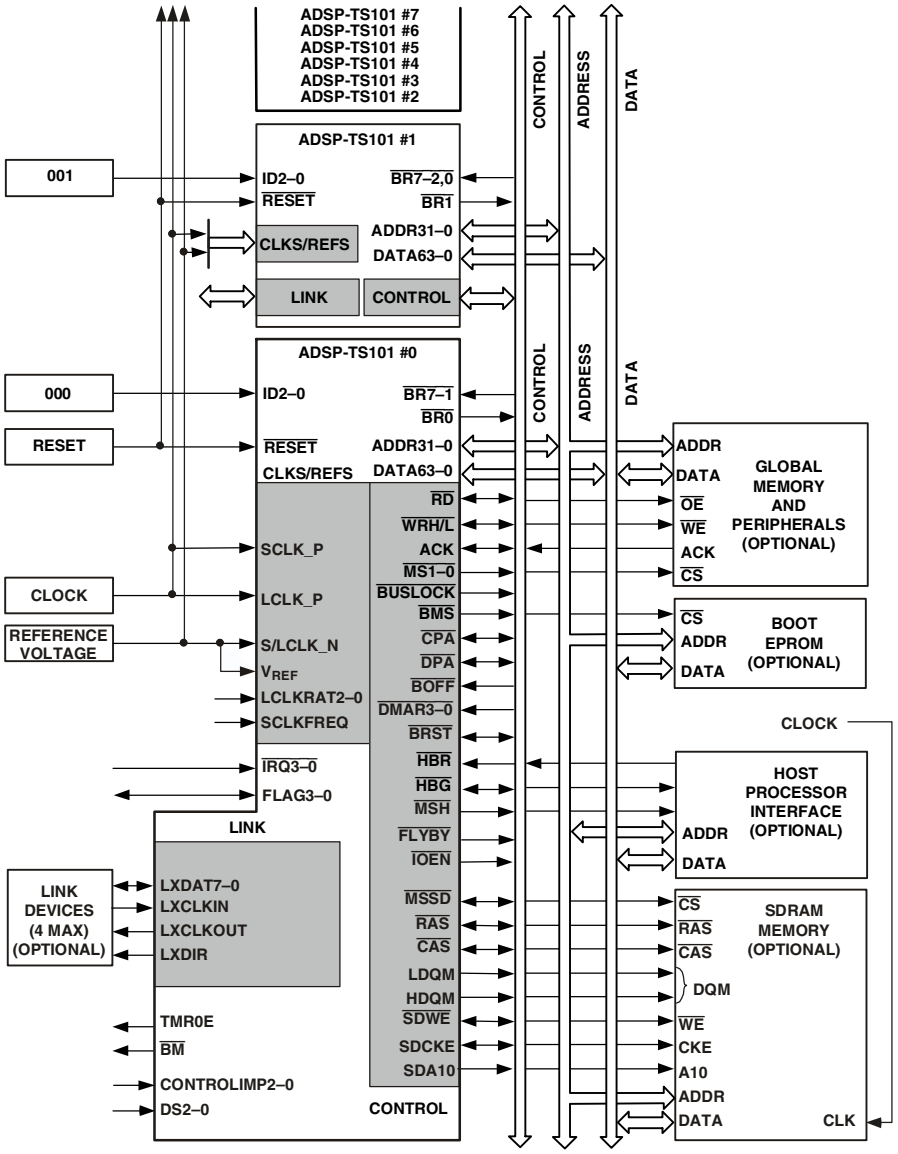

Figure 4. Shared Memory Multiprocessing System

The DMA controller performs the following DMA operations:

- External port block transfers. Four dedicated bidirectional DMA channels transfer blocks of data between the DSP's internal memory and any external memory or memory-mapped peripheral on the external bus. These transfers support master mode and handshake mode protocols.
- Link port transfers. Eight dedicated DMA channels (four transmit and four receive) transfer quad word data only between link ports and between a link port and internal or external memory. These transfers only use handshake mode protocol. DMA priority rotates between the four receive channels.
- AutoDMA transfers. Two dedicated unidirectional DMA channels transfer data received from an external bus master to internal memory or to link port I/O. These transfers only use slave mode protocol, and an external bus master must initiate the transfer.

LINK PORTS

The DSP's four link ports provide additional 8-bit bidirectional I/O capability. With the ability to operate at a double data rate—latching data on both the rising and falling edges of the clock—running at 125 MHz, each link port can support up to 250M bytes per second, for a combined maximum throughput of 1G bytes per second.

The link ports provide an optional communications channel that is useful in multiprocessor systems for implementing point-to-point interprocessor communications. Applications can also use the link ports for booting.

Each link port has its own double-buffered input and output registers. The DSP's core can write directly to a link port's transmit register and read from a receive register, or the DMA controller can perform DMA transfers through eight (four transmit and four receive) dedicated link port DMA channels.

Each link port has three signals that control its operation. LxCLKOUT and LxCLKIN implement clock/acknowledge handshaking. LxDIR indicates the direction of transfer and is used only when buffering the LxDAT signals. An example application would be using differential low-swing buffers for long twisted-pair wires. LxDAT provides the 8-bit data bus input/output.

Applications can program separate error detection mechanisms for transmit and receive operations (applications can use the checksum mechanism to implement consecutive link port transfers), the size of data packets, and the speed at which bytes are transmitted.

Under certain conditions, the link port receiver can initiate a token switch to reverse the direction of transfer; the transmitter becomes the receiver and vice versa.

TIMER AND GENERAL-PURPOSE I/O

The ADSP-TS101S has a timer pin (TMR0E) that generates output when a programmed timer counter has expired. Also, the DSP has four programmable general-purpose I/O pins (FLAG3–0) that can function as either single-bit input or output. As outputs, these pins can signal peripheral devices; as inputs, they can provide the test for conditional branching.

RESET AND BOOTING

The ADSP-TS101S has two levels of reset (see reset specifications Page 24):

- Power-up reset—after power-up of the system, and strap options are stable, the $\overline{\text{RESET}}$ pin must be asserted (low).
- Normal reset—for any resets following the power-up reset sequence, the $\overline{\text{RESET}}$ pin must be asserted.

The DSP can be reset internally (core reset) by setting the SWRST bit in SQCTL. The core is reset, but not the external port or I/O.

After reset, the ADSP-TS101S has four boot options for beginning operation:

- Boot from EPROM. The DSP defaults to EPROM booting when the $\overline{\text{BMS}}$ pin strap option is set low. See Strap Pin Function Descriptions on Page 19.
- Boot by an external master (host or another ADSP-TS101S). Any master on the cluster bus can boot the ADSP-TS101S through writes to its internal memory or through autoDMA.
- Boot by link port. All four receive link DMA channels are initialized after reset to transfer a 256-word block to internal memory address 0 to 255, and to issue an interrupt at the end of the block (similar to EP DMA). The corresponding DMA interrupts are set to address zero (0).
- No boot—Start running from an external memory. Using the "no boot" option, the ADSP-TS101S must start running from an external memory, caused by asserting one of the $\overline{\text{IRQ3–0}}$ interrupt signals.

The ADSP-TS101S core always exits from reset in the idle state and waits for an interrupt. Some of the interrupts in the interrupt vector table are initialized and enabled after reset.

LOW POWER OPERATION

The ADSP-TS101S can enter a low power sleep mode in which its core does not execute instructions, reducing power consumption to a minimum. The ADSP-TS101S exits sleep mode when it senses a falling edge on any of its $\overline{\text{IRQ3–0}}$ interrupt inputs. The interrupt, if enabled, causes the ADSP-TS101S to execute the corresponding interrupt service routine. This feature is useful for systems that require a low power standby mode.

CLOCK DOMAINS

As shown in Figure 5, the ADSP-TS101S has two clock inputs, SCLK (system clock) and LCLK (local clock).

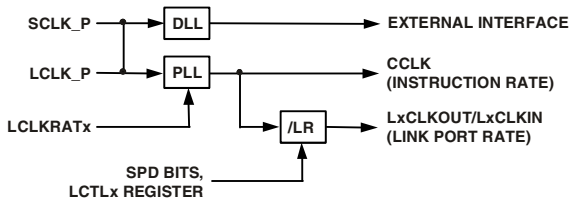

Figure 5. Clock Domains

These inputs drive its two major clock domains:

- SCLK (system clock). Provides clock input for the external bus interface and defines the ac specification reference for the external bus signals. The external bus interface runs at 1× the SCLK frequency. A DLL locks internal SCLK to SCLK input.
- LCLK (local clock). Provides clock input to the internal clock driver, CCLK, which is the internal clock for the core, internal buses, memory, and link ports. The instruction execution rate is equal to CCLK. A PLL from LCLK gener-

Table 6. Pin Definitions—External Port Arbitration (Continued)

Signal	Type	Term	Description
$\overline{HBG}$[3]	I/O/T (pu[2])	nc	Host Bus Grant. Acknowledges $\overline{HBR}$ and indicates that the host can take control of the external bus. When relinquishing the bus, the master DSP three-states the ADDR31–0, DATA63–0, $\overline{MSH}$, $\overline{MSSD}$, $\overline{MS1–0}$, $\overline{RD}$, $\overline{WRL}$, $\overline{WRH}$, $\overline{BMS}$, $\overline{BRST}$, $\overline{FLYBY}$, $\overline{IOEN}$, $\overline{RAS}$, $\overline{CAS}$, $\overline{SDWE}$, SDA10, SDCKE, LDQM and HDQM pins, and the DSP puts the SDRAM in self-refresh mode. The DSP asserts $\overline{HBG}$ until the host deasserts $\overline{HBR}$. In multiprocessor systems, the current bus master DSP drives $\overline{HBG}$, and all slave DSPs monitor $\overline{HBG}$.
$\overline{CPA}$	I/O (o/d)	See next column	Core Priority Access. Asserted while the DSP's core accesses external memory. This pin enables a slave DSP to interrupt a master DSP's background DMA transfers and gain control of the external bus for core-initiated transactions. $\overline{CPA}$ is an open drain output, connected to all DSPs in the system. The $\overline{CPA}$ pin has an internal 500 Ω pull-up resistor, which is only enabled on the DSP with ID2–0 = 0. If ID0 is not used, terminate this pin as either epu or nc. If ID7–1 is not used, terminate this pin as epu.
$\overline{DPA}$	I/O (o/d)	See next column	DMA Priority Access. Asserted while a high-priority DSP DMA channel accesses external memory. This pin enables a high-priority DMA channel on a slave DSP to interrupt transfers of a normal-priority DMA channel on a master DSP and gain control of the external bus for DMA-initiated transactions. $\overline{DPA}$ is an open drain output, connected to all DSPs in the system. The $\overline{DPA}$ pin has an internal 500 Ω pull-up resistor, which is only enabled on the DSP with ID2–0 = 0. If ID0 is not used, terminate this pin as either epu or nc. If ID7–1 is not used, terminate this pin as epu.

Type column symbols: A = asynchronous; G = ground; I = input; O = output; o/d = open drain output; P = power supply; pd = internal pull-down approximately 100 kΩ; pu = internal pull-up approximately 100 kΩ; T = three-state

Term (for termination) column symbols: epd = external pull-down approximately 10 kΩ to V_{SS}; epu = external pull-up approximately 10 kΩ to V_{DD-IO}, nc = not connected; au = always used.

[1] The internal pull-down may not be sufficient. A stronger pull-down may be necessary.
[2] See Electrical Characteristics on Page 20 for maximum and minimum current consumption for pull-up and pull-down resistances.
[3] The internal pull-up may not be sufficient. A stronger pull-up may be necessary.

Table 7. Pin Definitions—External Port DMA/Flyby

Signal	Type	Term	Description
$\overline{DMAR3–0}$	I/A	epu	DMA Request Pins. Enable external I/O devices to request DMA services from the DSP. In response to $\overline{DMARx}$, the DSP performs DMA transfers according to the DMA channel's initialization. The DSP ignores DMA requests from uninitialized channels.
$\overline{FLYBY}$[1]	O/T (pu[2])	nc	Flyby Mode. When a DSP DMA channel is initiated in $\overline{FLYBY}$ mode, it generates flyby transactions on the external bus. During flyby transactions, the DSP asserts $\overline{FLYBY}$, which signals the source or destination I/O device to latch the next data or strobe the current data, respectively, and to prepare for the next data on the next cycle.
$\overline{IOEN}$[1]	O/T (pu[2])	nc	I/O Device Output Enable. Enables the output buffers of an external I/O device for flyby transactions between the device and external memory. Active on flyby transactions.

Type column symbols: A = asynchronous; G = ground; I = input; O = output; o/d = open drain output; P = power supply; pd = internal pull-down approximately 100 kΩ; pu = internal pull-up approximately 100 kΩ; T = three-state

Term (for termination) column symbols: epd = external pull-down approximately 10 kΩ to V_{SS}; epu = external pull-up approximately 10 kΩ to V_{DD-IO}, nc = not connected; au = always used.

[1] The internal pull-up may not be sufficient. A stronger pull-up may be necessary.
[2] See Electrical Characteristics on Page 20 for maximum and minimum current consumption for pull-up and pull-down resistances.

Table 8. Pin Definitions—External Port SDRAM Controller

Signal	Type	Term	Description
$\overline{\text{MSSD}}$[1]	I/O/T (pu[2])	nc	Memory Select SDRAM. $\overline{\text{MSSD}}$ is asserted whenever the DSP accesses SDRAM memory space. $\overline{\text{MSSD}}$ is a decoded memory address pin that is asserted whenever the DSP issues an SDRAM command cycle (access to ADDR31:26 = 0b000001). MSSD in a multiprocessor system is driven by the master DSP.
$\overline{\text{RAS}}$[1]	I/O/T (pu[2])	nc	Row Address Select. When sampled low, $\overline{\text{RAS}}$ indicates that a row address is valid in a read or write of SDRAM. In other SDRAM accesses, RAS defines the type of operation to execute according to SDRAM specification.
$\overline{\text{CAS}}$[1]	I/O/T (pu[2])	nc	Column Address Select. When sampled low, $\overline{\text{CAS}}$ indicates that a column address is valid in a read or write of SDRAM. In other SDRAM accesses, $\overline{\text{CAS}}$ defines the type of operation to execute according to the SDRAM specification.
LDQM[1]	O/T (pu[2])	nc	Low Word SDRAM Data Mask. When LDQM is sampled high, the DSP three-states the SDRAM DQ buffers. LDQM is valid on SDRAM transactions when $\overline{\text{CAS}}$ is asserted and is inactive on read transactions. On write transactions, LDQM is active when accessing an odd address word on a 64-bit memory bus to disable the write of the low word.
HDQM[1]	O/T (pu[2])	nc	High Word SDRAM Data Mask. When HDQM is sampled high, the DSP three-states the SDRAM DQ buffers. HDQM is valid on SDRAM transactions when $\overline{\text{CAS}}$ is asserted and is inactive on read transactions. On write transactions, HDQM is active when accessing an even address in word accesses or is active when memory is configured for a 32-bit bus to disable the write of the high word.
SDA10[1]	O/T (pu[2])	nc	SDRAM Address bit 10 pin. Separate A10 signals enable SDRAM refresh operation while the DSP executes non-SDRAM transactions.
SDCKE[1, 3]	I/O/T (pu/pd[2])	nc	SDRAM Clock Enable. Activates the SDRAM clock for SDRAM self-refresh or suspend modes. A slave DSP in a multiprocessor system does not have the pull-up or pull-down. A master DSP (or ID = 0 in a single processor system) has a 100 kΩ pull-up before granting the bus to the host, except when the SDRAM is put in self-refresh mode. In self-refresh mode, the master has a 100 kΩ pull-down before granting the bus to the host.
$\overline{\text{SDWE}}$[1]	I/O/T (pu[2])	nc	SDRAM Write Enable. When sampled low while $\overline{\text{CAS}}$ is active, $\overline{\text{SDWE}}$ indicates an SDRAM write access. When sampled high while $\overline{\text{CAS}}$ is active, $\overline{\text{SDWE}}$ indicates an SDRAM read access. In other SDRAM accesses, $\overline{\text{SDWE}}$ defines the type of operation to execute according to SDRAM specification.

Type column symbols: A = asynchronous; G = ground; I = input; O = output; o/d = open drain output; P = power supply; pd = internal pull-down approximately 100 kΩ; pu = internal pull-up approximately 100 kΩ; T = three-state

Term (for termination) column symbols: epd = external pull-down approximately 10 kΩ to V_{SS}; epu = external pull-up approximately 10 kΩ to $V_{DD\text{-}IO}$, nc = not connected; au = always used.

[1] The internal pull-up may not be sufficient. A stronger pull-up may be necessary.
[2] See Electrical Characteristics on Page 20 for maximum and minimum current consumption for pull-up and pull-down resistances.
[3] The internal pull-down may not be sufficient. A stronger pull-down may be necessary.

Table 9. Pin Definitions—JTAG Port

Signal	Type	Term	Description
$\overline{\text{EMU}}$	O (o/d)	nc[1]	Emulation. Connected only to the DSP's JTAG emulator target board connector.
TCK	I	epd or epu[1]	Test Clock (JTAG). Provides an asynchronous clock for JTAG scan.
TDI[2]	I (pu[3])	nc[1]	Test Data Input (JTAG). A serial data input of the scan path.

Type column symbols: A = asynchronous; G = ground; I = input; O = output; o/d = open drain output; P = power supply; pd = internal pull-down approximately 100 kΩ; pu = internal pull-up approximately 100 kΩ; T = three-state

Term (for termination) column symbols: epd = external pull-down approximately 10 kΩ to V_{SS}; epu = external pull-up approximately 10 kΩ to $V_{DD\text{-}IO}$, nc = not connected; au = always used.

Table 15. Pin Definitions—Power, Ground, and Reference

Signal	Type	Term	Description
V_{DD}	P	au	V_{DD} pins for internal logic.
V_{DD_A}	P	au	V_{DD} pins for analog circuits. Pay critical attention to bypassing this supply.
V_{DD_IO}	P	au	V_{DD} pins for I/O buffers.
V_{REF}	I	au	Reference voltage defines the trip point for all input buffers, except $\overline{RESET}$, $\overline{IRQ3–0}$, $\overline{DMAR3–0}$, ID2–0, CONTROLIMP2–0, TCK, TDI, TMS, and $\overline{TRST}$. The value is 1.5 V ± 100 mV (which is the TTL trip point). V_{REF} can be connected to a power supply or set by a voltage divider circuit. The voltage divider should have an HF decoupling capacitor (1 nF HF SMD) connected to V_{SS}. Tie the decoupling capacitor between V_{REF} input and V_{SS}, as close to the DSP's pins as possible. For more information, see Filtering Reference Voltage and Clocks on Page 10.
V_{SS}	G	au	Ground pins.
V_{SS_A}	G	au	Ground pins for analog circuits.
NC			No connect. Do not connect these pins to anything (not to any supply, signal, or each other), because they are reserved and must be left unconnected.

Type column symbols: A = asynchronous; G = ground; I = input; O = output; o/d = open drain output; P = power supply; pd = internal pull-down approximately 100 kΩ; pu = internal pull-up approximately 100 kΩ; T = three-state

Term (for termination) column symbols: epd = external pull-down approximately 10 kΩ to V_{SS}; epu = external pull-up approximately 10 kΩ to V_{DD-IO}, nc = not connected; au = always used.

STRAP PIN FUNCTION DESCRIPTIONS

Some pins have alternate functions at reset. Strap options set DSP operating modes. During reset, the DSP samples the strap option pins. Strap pins have an approximately 100 kΩ pull-down for the default value. If a strap pin is not connected to an external pull-up or logic load, the DSP samples the default value during reset. If strap pins are connected to logic inputs, a stronger external pull-down may be required to ensure default value depending on leakage and/or low level input current of the logic load. To set a mode other than the default mode, connect the strap pin to a sufficiently stronger external pull-up. In a multiprocessor system, up to eight DSPs may be connected on the cluster bus, resulting in parallel combination of strap pin pull-down resistors. Table 16 lists and describes each of the DSP's strap pins.

Table 16. Pin Definitions—I/O Strap Pins

Signal	On Pin …	Description
EBOOT	$\overline{BMS}$	EPROM boot. 0 = boot from EPROM immediately after reset (default) 1 = idle after reset and wait for an external device to boot DSP through the external port or a link port
IRQEN	$\overline{BM}$	Interrupt Enable. 0 = disable and set $\overline{IRQ3–0}$ interrupts to level sensitive after reset (default) 1 = enable and set $\overline{IRQ3–0}$ interrupts to edge sensitive immediately after reset
TM1	L2DIR	Test Mode 1. 0 = required setting during reset. 1 = reserved.
TM2	TMR0E	Test Mode 2. 0 = required setting during reset. 1 = reserved.

Table 22. Reference Clocks—System Clock (SCLK) Cycle Time

Parameter	Description	Min	Max	Unit
t_{SCLK}[1, 2, 3, 4]	System Clock Cycle Time	10	25	ns
t_{SCLKH}	System Clock Cycle High Time	$0.4 \times t_{SCLK}$	$0.6 \times t_{SCLK}$	ns
t_{SCLKL}	System Clock Cycle Low Time	$0.4 \times t_{SCLK}$	$0.6 \times t_{SCLK}$	ns
t_{SCLKJ}[5, 6]	System Clock Jitter Tolerance		500	ps

[1] For more information, see Table 3 on Page 12.
[2] For more information, see Clock Domains on Page 9.
[3] LCLK_P and SCLK_P must be connected to the same source.
[4] The value of (t_{SCLK} / LCLKRAT2-0) must not violate the specification for t_{CCLK}.
[5] Actual input jitter should be combined with ac specifications for accurate timing analysis.
[6] Jitter specification is maximum peak-to-peak time interval error (TIE) jitter.

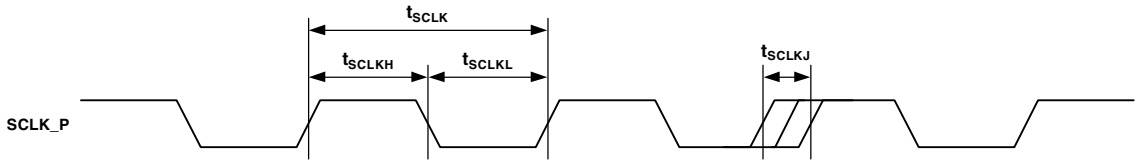

Figure 11. Reference Clocks—System Clock (SCLK) Cycle Time

Table 23. Reference Clocks—Test Clock (TCK) Cycle Time

Parameter	Description	Min	Max	Unit
t_{TCK}	Test Clock (JTAG) Cycle Time	Greater of 30 or $t_{CCLK} \times 4$		ns
t_{TCKH}	Test Clock (JTAG) Cycle High Time	12.5		ns
t_{TCKL}	Test Clock (JTAG) Cycle Low Time	12.5		ns

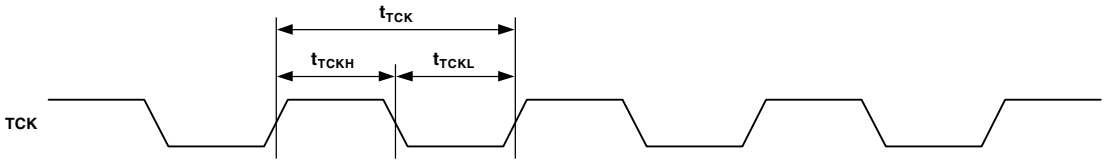

Figure 12. Reference Clocks—Test Clock (TCK) Cycle Time

Table 24. Power-Up Timing[1]

Parameter		Min	Max	Unit
Timing Requirement				
t_{VDD_IO}	V_{DD_IO} Stable and Within Specification After V_{DD} and V_{DD_A} Are Stable and Within Specification	>0		ms

[1] For information about power supply sequencing and monitoring solutions, please visit http://www.analog.com/sequencing.

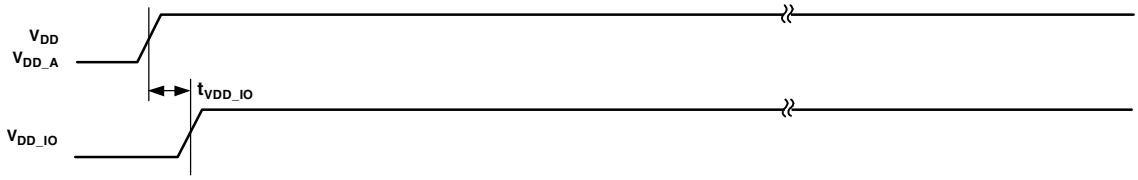

Figure 13. Power-Up Sequencing Timing

Table 27. AC Signal Specifications (for SCLK <16.7 ns) (All values in this table are in nanoseconds)

Name	Description	Input Setup (min)	Input Hold (min)	Output Valid (max)[1]	Output Hold (min)	Output Enable (min)[2]	Output Disable (max)[2]	Reference Clock
ADDR31–0	External Address Bus	2.6	0.5	4.2	1.0	0.9	2.5	SCLK
DATA63–0	External Data Bus	2.6	0.5	4.2	1.0	0.9	2.5	SCLK
$\overline{\text{MSH}}$	Memory Select Host Line			4.2	1.0	0.9	2.5	SCLK
$\overline{\text{MSSD}}$	Memory Select SDRAM Line	2.6	0.5	4.2	1.0	0.9	2.5	SCLK
$\overline{\text{MS1–0}}$	Memory Select for Static Blocks			4.2	1.0	0.9	2.5	SCLK
$\overline{\text{RD}}$	Memory Read	2.6	0.5	4.2	1.0	0.9	2.5	SCLK
$\overline{\text{WRL}}$	Write Low Word	2.6	0.5	4.2	1.0	0.9	2.5	SCLK
$\overline{\text{WRH}}$	Write High Word	2.6	0.5	4.2	1.0	0.9	2.5	SCLK
ACK	Acknowledge for Data	2.6	0.5	4.2	1.0	0.9	2.5	SCLK
SDCKE	SDRAM Clock Enable	2.6	0.5	4.2	1.0	0.9	2.5	SCLK
$\overline{\text{RAS}}$	Row Address Select	2.6	0.5	4.2	1.0	0.9	2.5	SCLK
$\overline{\text{CAS}}$	Column Address Select	2.6	0.5	4.2	1.0	0.9	2.5	SCLK
$\overline{\text{SDWE}}$	SDRAM Write Enable	2.6	0.5	4.2	1.0	0.9	2.5	SCLK
LDQM	Low Word SDRAM Data Mask			4.2	1.0	0.9	2.5	SCLK
HDQM	High Word SDRAM Data Mask			4.2	1.0	0.9	2.5	SCLK
SDA10	SDRAM ADDR10			4.2	1.0	0.9	2.5	SCLK
$\overline{\text{HBR}}$	Host Bus Request	2.6	0.5					SCLK
$\overline{\text{HBG}}$	Host Bus Grant	2.6	0.5	4.2	1.0	0.9	2.5	SCLK
$\overline{\text{BOFF}}$	Back Off Request	2.6	0.5					SCLK
$\overline{\text{BUSLOCK}}$	Bus Lock			4.2	1.0	0.9	2.5	SCLK
$\overline{\text{BRST}}$	Burst Access	2.6	0.5	4.2	1.0	0.9	2.5	SCLK
$\overline{\text{BR7–0}}$	Multiprocessing Bus Request	2.6	0.5	4.2	1.0			SCLK
$\overline{\text{FLYBY}}$	Flyby Mode Selection			4.2	1.0	0.9	2.5	SCLK
$\overline{\text{IOEN}}$	Flyby I/O Enable			4.2	1.0	0.9	2.5	SCLK
$\overline{\text{CPA}}$ [3, 4]	Core Priority Access	2.6	0.5	5.8			2.5	SCLK
$\overline{\text{DPA}}$ [3, 4]	DMA Priority Access	2.6	0.5	5.8			2.5	SCLK
$\overline{\text{BMS}}$[5]	Boot Memory Select			4.2	1.0	0.9	2.5	SCLK
FLAG3–0[6]	FLAG Pins			4.2	1.0	1.0	4.0	SCLK
$\overline{\text{RESET}}$[4, 7]	Global Reset							SCLK
TMS[4]	Test Mode Select (JTAG)	1.5	1.0					TCK
TDI[4]	Test Data Input (JTAG)	1.5	1.0					TCK
TDO	Test Data Output (JTAG)			6.0	1.0	1.0	5.0	TCK_FE[8]
$\overline{\text{TRST}}$[4, 7, 9]	Test Reset (JTAG)							TCK
$\overline{\text{BM}}$[5]	Bus Master Debug Aid Only			4.2	1.0			SCLK
$\overline{\text{EMU}}$[10]	Emulation			5.5			5.0	TCK or LCLK
JTAG_SYS_IN[11]	System Input	1.5	11.0					TCK
JTAG_SYS_OUT[12]	System Output			16.0				TCK_FE[8]
ID2–0[9]	Chip ID—Must Be Constant							
CONTROLIMP2–0[9]	Static Pins—Must Be Constant							
DS2–0[9]	Static Pins—Must Be Constant							
LCLKRAT2–0[9]	Static Pins—Must Be Constant							
SCLKFREQ[9]	Static Pins—Must Be Constant							

[1] The output valid (max) value in this column applies for the standard 30 pF capacitive load used in testing. To see how output valid varies with capacitive loading, see Figure 40 on Page 36.
[2] The external port protocols employ bus IDLE cycles for bus mastership transitions as well as slave address boundary crossings to avoid any potential bus contention. The apparent driver overlap, due to output disables being larger than output enables, is not actual.
[3] $\overline{CPA}$ and $\overline{DPA}$ pins are open drains and have 0.5 kΩ internal pull-ups.
[4] These input pins have Schmitt triggers and therefore do not need to be synchronized to a clock reference. These synchronous specifications only apply for recognition in the current clock reference cycle.
[5] This pin is a strap option. During reset, an internal resistor pulls the pin low.
[6] For input specifications, see Table 21.
[7] For additional requirement details, see Reset and Booting on Page 9.
[8] TCK_FE indicates TCK falling edge.
[9] These pins may change only during reset; recommend connecting it to V_{DD_IO}/V_{SS}.
[10] Reference clock depends on function.
[11] System inputs are: $\overline{IRQ3–0}$, $\overline{BMS}$, LCLKRAT2–0, SCLKFREQ, $\overline{BM}$, TMR0E, FLAG3–0, ID2–0, $\overline{BRST}$, $\overline{WRH}$, $\overline{WRL}$, $\overline{RD}$, $\overline{MSSD}$, SDCKE, $\overline{SDWE}$, $\overline{CAS}$, $\overline{RAS}$, ADDR31–0, DATA63–0, $\overline{DPA}$, $\overline{CPA}$, $\overline{HBG}$, $\overline{BOFF}$, $\overline{HBR}$, ACK, $\overline{BR7–0}$, L0CLKIN, L0DAT7–0, L1CLKIN, L1DAT7–0, L2CLKIN, L2DAT7–0, L2DIR, L3CLKIN, L3DAT7–0, DS2–0, CONTROLIMP2–0, $\overline{RESET}$, $\overline{DMAR3–0}$.
[12] System outputs are: $\overline{BMS}$, $\overline{BM}$, $\overline{BUSLOCK}$, TMR0E, FLAG3–0, $\overline{FLYBY}$, $\overline{IOEN}$, $\overline{MSH}$, $\overline{BRST}$, $\overline{WRH}$, $\overline{WRL}$, $\overline{RD}$, $\overline{MS1–0}$, HDQM, LDQM, $\overline{MSSD}$, SDCKE, $\overline{SDWE}$, $\overline{CAS}$, $\overline{RAS}$, ADDR31–0, DATA63–0, $\overline{DPA}$, $\overline{CPA}$, $\overline{HBG}$, ACK, $\overline{BR7–0}$, L0CLKOUT, L0DAT7–0, L0DIR, L1CLKOUT, L1DAT7–0, L1DIR, L2CLKOUT, L2DAT7–0, L2DIR, L3CLKOUT, L3DAT7–0, L3DIR, $\overline{EMU}$.

Table 28. AC Signal Specifications (for 16.7 ns <SCLK <25 ns) (All values in this table are in nanoseconds)

Name	Description	Input Setup (min)	Input Hold (min)	Output Valid (max)[1]	Output Hold (min)	Output Enable (min)[2]	Output Disable (max)[2]	Reference Clock
ADDR31–0	External Address Bus	2.8	0.5	4.2	0.8	0.3	2.5	SCLK
DATA63–0	External Data Bus	2.8	0.5	4.2	0.8	0.3	2.5	SCLK
$\overline{MSH}$	Memory Select Host Line			4.2	0.8	0.3	2.5	SCLK
$\overline{MSSD}$	Memory Select SDRAM Line	2.8	0.5	4.2	0.8	0.3	2.5	SCLK
$\overline{MS1–0}$	Memory Select for Static Blocks			4.2	0.8	0.3	2.5	SCLK
$\overline{RD}$	Memory Read	2.8	0.5	4.2	0.8	0.3	2.5	SCLK
$\overline{WRL}$	Write Low Word	2.8	0.5	4.2	0.8	0.3	2.5	SCLK
$\overline{WRH}$	Write High Word	2.8	0.5	4.2	0.8	0.3	2.5	SCLK
ACK	Acknowledge for Data	2.8	0.5	4.2	0.8	0.3	2.5	SCLK
SDCKE	SDRAM Clock Enable	2.8	0.5	4.2	0.8	0.3	2.5	SCLK
$\overline{RAS}$	Row Address Select	2.8	0.5	4.2	0.8	0.3	2.5	SCLK
$\overline{CAS}$	Column Address Select	2.8	0.5	4.2	0.8	0.3	2.5	SCLK
$\overline{SDWE}$	SDRAM Write Enable	2.8	0.5	4.2	0.8	0.3	2.5	SCLK
LDQM	Low Word SDRAM Data Mask			4.2	0.8	0.3	2.5	SCLK
HDQM	High Word SDRAM Data Mask			4.2	0.8	0.3	2.5	SCLK
SDA10	SDRAM ADDR10			4.2	0.8	0.3	2.5	SCLK
$\overline{HBR}$	Host Bus Request	2.8	0.5					SCLK
$\overline{HBG}$	Host Bus Grant	2.8	0.5	4.2	0.8	0.3	2.5	SCLK
$\overline{BOFF}$	Back Off Request	2.8	0.5					SCLK
$\overline{BUSLOCK}$	Bus Lock			4.2	0.8	0.3	2.5	SCLK
$\overline{BRST}$	Burst Access	2.8	0.5	4.2	0.8	0.3	2.5	SCLK
$\overline{BR7–0}$	Multiprocessing Bus Request	2.8	0.5	4.2	0.8			SCLK
$\overline{FLYBY}$	Flyby Mode Selection			4.2	0.8	0.3	2.5	SCLK
$\overline{IOEN}$	Flyby Mode I/O Enable			4.2	0.8	0.3	2.5	SCLK
$\overline{CPA}$ [3, 4]	Core Priority Access	2.8	0.5	5.8			2.5	SCLK
$\overline{DPA}$ [3, 4]	DMA Priority Access	2.8	0.5	5.8			2.5	SCLK
$\overline{BMS}$[5]	Boot Memory Select			4.2	0.8	0.3	2.5	SCLK
FLAG3–0[6]	FLAG Pins			4.2	1.0	1.0	4.0	SCLK

Table 28. AC Signal Specifications (for 16.7 ns <SCLK <25 ns) (All values in this table are in nanoseconds) (Continued)

Name	Description	Input Setup (min)	Input Hold (min)	Output Valid (max)[1]	Output Hold (min)	Output Enable (min)[2]	Output Disable (max)[2]	Reference Clock
$\overline{\text{RESET}}$[4, 7]	Global Reset							SCLK
TMS[4]	Test Mode Select (JTAG)	1.5	1.0					TCK
TDI[4]	Test Data Input (JTAG)	1.5	1.0					TCK
TDO	Test Data Output (JTAG)			6.0	1.0	1.0	5.0	TCK_FE[8]
$\overline{\text{TRST}}$[4, 7, 9]	Test Reset (JTAG)							TCK
$\overline{\text{BM}}$[5]	Bus Master Debug Aid Only			4.2	0.8			SCLK
$\overline{\text{EMU}}$[10]	Emulation			5.5			5.0	TCK or LCLK
JTAG_SYS_IN[11]	System Input	1.5	11.0					TCK
JTAG_SYS_OUT[12]	System Output			16.0				TCK_FE[8]
ID2–0[9]	Chip ID—Must Be Constant							
CONTROLIMP2–0[9]	Static Pins—Must Be Constant							
DS2–0[9]	Static Pins—Must Be Constant							
LCLKRAT2–0[9]	Static Pins—Must Be Constant							
SCLKFREQ[9]	Static Pins—Must Be Constant							

[1] The output valid (max) value in this column applies for the standard 30 pF capacitive load used in testing. To see how output valid varies with capacitive loading, see Figure 40 on Page 36.

[2] The external port protocols employ bus IDLE cycles for bus mastership transitions as well as slave address boundary crossings to avoid any potential bus contention. The apparent driver overlap, due to output disables being larger than output enables, is not actual.

[3] $\overline{\text{CPA}}$ and $\overline{\text{DPA}}$ pins are open drains and have 0.5 kΩ internal pull-ups.

[4] These input pins have Schmitt triggers and therefore do not need to be synchronized to a clock reference. These synchronous specifications only apply for recognition in the current clock reference cycle.

[5] This pin is a strap option. During reset, an internal resistor pulls the pin low.

[6] For input specifications, see Table 21.

[7] For additional requirement details, see Reset and Booting on Page 9.

[8] TCK_FE indicates TCK falling edge.

[9] These pins may change only during reset; recommend connecting it to V_{DD_IO}/V_{SS}.

[10] Reference clock depends on function.

[11] System inputs are: $\overline{\text{IRQ3–0}}$, $\overline{\text{BMS}}$, LCLKRAT2–0, SCLKFREQ, $\overline{\text{BM}}$, TMR0E, FLAG3–0, ID2–0, $\overline{\text{BRST}}$, $\overline{\text{WRH}}$, $\overline{\text{WRL}}$, $\overline{\text{RD}}$, $\overline{\text{MSSD}}$, SDCKE, $\overline{\text{SDWE}}$, $\overline{\text{CAS}}$, $\overline{\text{RAS}}$, ADDR31–0, DATA63–0, $\overline{\text{DPA}}$, $\overline{\text{CPA}}$, $\overline{\text{HBG}}$, $\overline{\text{BOFF}}$, $\overline{\text{HBR}}$, ACK, $\overline{\text{BR7–0}}$, L0CLKIN, L0DAT7–0, L1CLKIN, L1DAT7–0, L2CLKIN, L2DAT7–0, L2DIR, L3CLKIN, L3DAT7–0, DS2–0, CONTROLIMP2–0, $\overline{\text{RESET}}$, $\overline{\text{DMAR3–0}}$.

[12] System outputs are: $\overline{\text{BMS}}$, $\overline{\text{BM}}$, $\overline{\text{BUSLOCK}}$, TMR0E, FLAG3–0, $\overline{\text{FLYBY}}$, $\overline{\text{IOEN}}$, $\overline{\text{MSH}}$, $\overline{\text{BRST}}$, $\overline{\text{WRH}}$, $\overline{\text{WRL}}$, $\overline{\text{RD}}$, $\overline{\text{MS1–0}}$, HDQM, LDQM, $\overline{\text{MSSD}}$, SDCKE, $\overline{\text{SDWE}}$, $\overline{\text{CAS}}$, $\overline{\text{RAS}}$, ADDR31–0, DATA63–0, $\overline{\text{DPA}}$, $\overline{\text{CPA}}$, $\overline{\text{HBG}}$, ACK, $\overline{\text{BR7–0}}$, L0CLKOUT, L0DAT7–0, L0DIR, L1CLKOUT, L1DAT7–0, L1DIR, L2CLKOUT, L2DAT7–0, L2DIR, L3CLKOUT, L3DAT7–0, L3DIR, $\overline{\text{EMU}}$.

Link Ports Data Transfer and Token Switch Timing

Table 31, Table 32, Table 33, and Table 34 with Figure 17, Figure 18, Figure 19, and Figure 20 provide the timing specifications for the link ports data transfer and token switch.

Table 29. Link Ports—Transmit

Parameter		Min	Max	Unit
Timing Requirements				
t_{CONNS}[1]	Connectivity Pulse Setup	$2 \times t_{CCLK} + 3.5$		ns
t_{CONNS}[2]	Connectivity Pulse Setup	8		ns
t_{CONNIW}[3]	Connectivity Pulse Input Width	$t_{LxCLK\_Tx} + 1$		ns
t_{ACKS}	Acknowledge Setup	$0.5 \times t_{LxCLK\_Tx}$		ns
Switching Characteristics				
t_{LxCLK_Tx}[4]	Transmit Link Clock Period	$0.9 \times LR \times t_{CCLK}$	$1.1 \times LR \times t_{CCLK}$	ns
t_{LxCLKH_Tx}[1]	Transmit Link Clock Width High	$0.33 \times t_{LxCLK\_Tx}$	$0.66 \times t_{LxCLK\_Tx}$	ns
t_{LxCLKH_Tx}[2]	Transmit Link Clock Width High	$0.4 \times t_{LxCLK\_Tx}$	$0.6 \times t_{LxCLK\_Tx}$	ns
t_{LxCLKL_Tx}[1]	Transmit Link Clock Width Low	$0.33 \times t_{LxCLK\_Tx}$	$0.66 \times t_{LxCLK\_Tx}$	ns
t_{LxCLKL_Tx}[2]	Transmit Link Clock Width Low	$0.4 \times t_{LxCLK\_Tx}$	$0.6 \times t_{LxCLK\_Tx}$	ns
t_{DIRS}	LxDIR Transmit Setup	$0.5 \times t_{LxCLK\_Tx}$	$2 \times t_{LxCLK\_Tx}$	ns
t_{DIRH}	LxDIR Transmit Hold	$0.5 \times t_{LxCLK\_Tx}$	$2 \times t_{LxCLK\_Tx}$	ns
t_{DOS}[1]	LxDAT7–0 Output Setup	$0.25 \times t_{LxCLK\_Tx} - 1$		ns
t_{DOH}[1]	LxDAT7–0 Output Hold	$0.25 \times t_{LxCLK\_Tx} - 1$		ns
t_{DOS}[2]	LxDAT7–0 Output Setup	Greater of 0.8 or $0.17 \times t_{LxCLK\_Tx} - 1$		ns
t_{DOH}[2]	LxDAT7–0 Output Hold	Greater of 0.8 or $0.17 \times t_{LxCLK\_Tx} - 1$		ns
t_{LDOE}	LxDAT7–0 Output Enable	1		ns
t_{LDOD}[5]	LxDAT7–0 Output Disable	1		ns

[1] The formula for this parameter applies when LR is 2.
[2] The formula for this parameter applies when LR is 3, 4, or 8.
[3] LxCLKIN shows the connectivity pulse with each of the three possible transitions to "Acknowledge." After a connectivity pulse low minimum, LxCLKIN may [1] return high and remain high for "Acknowledge," [2] return high and subsequently go low (meeting t_{ACKS}) for "Not Acknowledge," or [3] remain low for "Not Acknowledge."
[4] The Link clock Ratio (LR) is 2, 3, 4, or 8 as set by the SPD bits in the LCTLx register. The maximum LxCLK is 125 MHz. LR = 2 may not be used when CCLK ≥ 250 MHz.
[5] This specification applies to the last data byte or the "Dummy" byte that follows the verification byte if enabled. For more information, see the *ADSP-TS101* TigerSHARC *Processor Hardware Reference*.

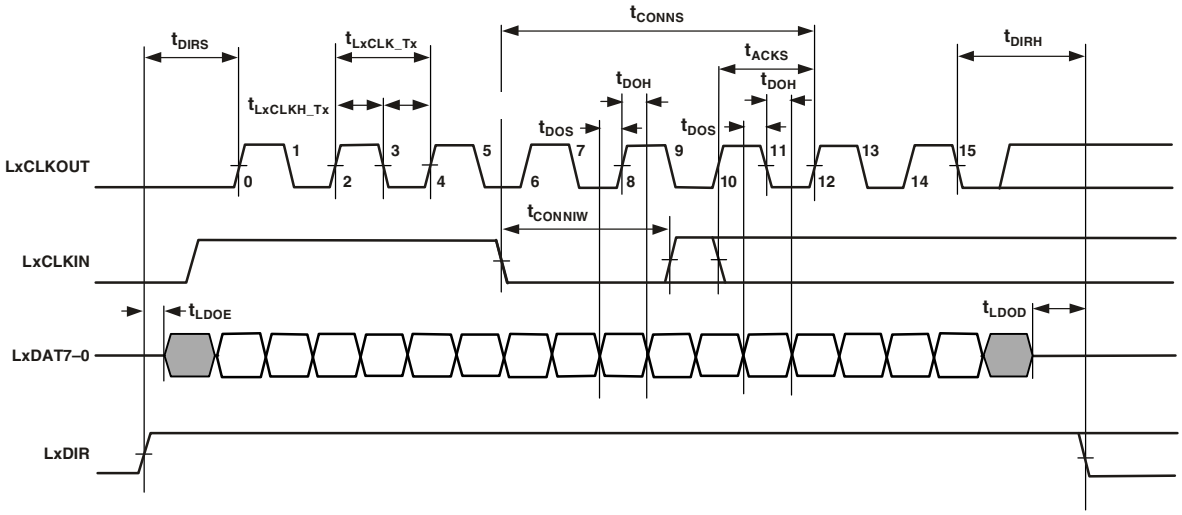

Figure 17. Link Ports—Transmit

Table 30. Link Ports—Receive

Parameter		Min	Max	Unit
Timing Requirements				
t_{LxCLK_Rx}[1, 2]	Receive Link Clock Period	$0.9 \times LR \times t_{CCLK}$	$1.1 \times LR \times t_{CCLK}$	ns
t_{LxCLKH_Rx}[3]	Receive Link Clock Width High	$0.33 \times t_{LxCLK\_Rx}$	$0.66 \times t_{LxCLK\_Rx}$	ns
t_{LxCLKH_Rx}[4]	Receive Link Clock Width High	$0.4 \times t_{LxCLK\_Rx}$	$0.6 \times t_{LxCLK\_Rx}$	ns
t_{LxCLKL_Rx}[3]	Receive Link Clock Width Low	$0.33 \times t_{LxCLK\_Rx}$	$0.66 \times t_{LxCLK\_Rx}$	ns
t_{LxCLKL_Rx}[4]	Receive Link Clock Width Low	$0.4 \times t_{LxCLK\_Rx}$	$0.6 \times t_{LxCLK\_Rx}$	ns
t_{DIS}	LxDAT7–0 Input Setup	0.6		ns
t_{DIH}	LxDAT7–0 Input Hold	0.6		ns
Switching Characteristics				
t_{CONNV}	Connectivity Pulse Valid	0	$2.5 \times t_{LxCLK\_Rx}$	ns
t_{CONNOW}	Connectivity Pulse Output Width	$1.5 \times t_{LxCLK\_Rx}$		ns

[1] The link clock ratio (LR) is 2, 3, 4, or 8 as set by the SPD bits in the LCTLx register.
[2] The maximum LxCLK is 125 MHz. LR = 2 may not be used when CCLK ≥ 250 MHz.
[3] The formula for this parameter applies when LR is 2.
[4] The formula for this parameter applies when LR is 3, 4, or 8.

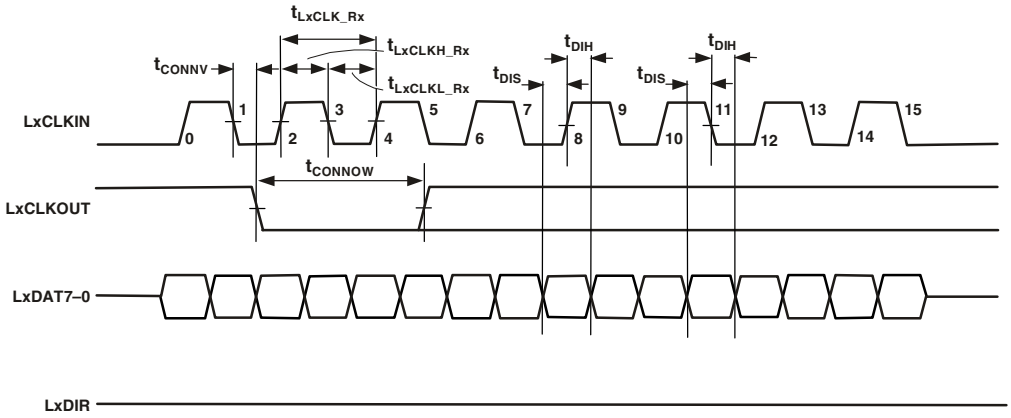

Figure 18. Link Ports—Receive

Table 31. Link Ports—Token Switch, Token Master

Parameter		Min	Max	Unit
Timing Requirements				
t_{REQI}	Token Request Input Width	$5.0 \times t_{LxCLK\_Rx}$		ns
t_{TKRQ}	Token Request from Token Enable[1]		$3.0 \times t_{LxCLK\_Tx}$	ns
Switching Characteristics				
t_{TKENO}	Token Switch Enable Output	$8.0 \times t_{LxCLK\_Tx}$		ns
t_{REQO}	Token Request Output Width[2]	$6.0 \times t_{LxCLK\_Tx}$		ns

[1] For guaranteeing token switch during token enable.
[2] LxCLKOUT shows both possible responses to the token request: [1] a "Token Grant" (LxCLKOUT remains high), and [2] a "Token Regret" (LxCLKOUT goes low).

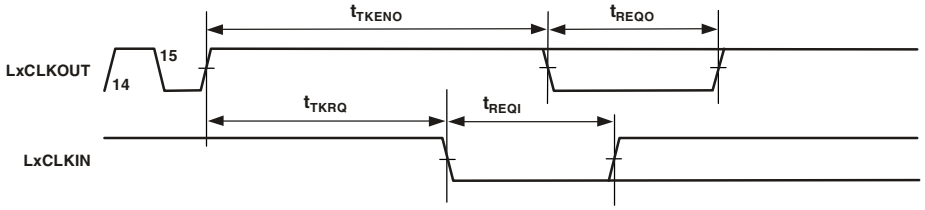

Figure 19. Link Ports—Token Switch, Token Master

Table 32. Link Ports—Token Switch, Token Requester

Parameter		Min	Max	Unit
Timing Requirements				
t_{TKENI}[1]	Token Switch Enable Input	$8.0 \times t_{LxCLK\_Rx}$		ns
Switching Characteristics				
t_{REQO}	Token Request Output Width[2]	$6.0 \times t_{LxCLK\_Rx}$		ns

[1] Required whenever there is a break in transmission.
[2] LxCLKOUT shows both possible responses to the token request: [1] a "Token Grant" (LxCLKOUT remains high), and [2] a "Token Regret" (LxCLKOUT goes low).

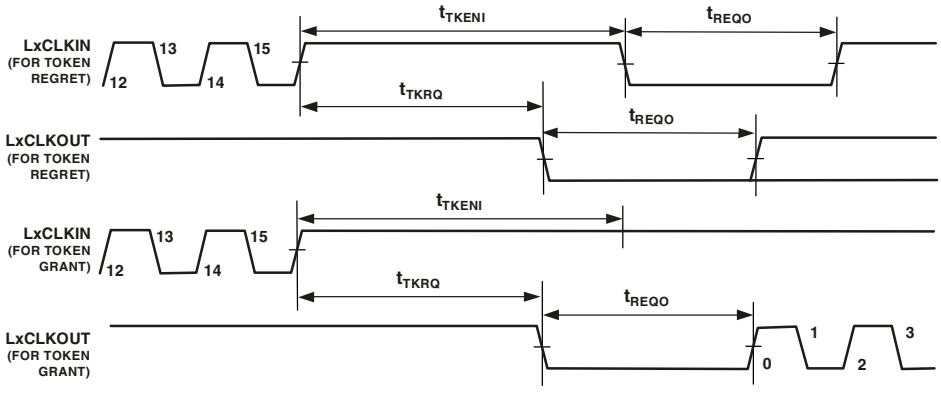

Figure 20. Link Ports—Token Switch, Token Requester

TEST CONDITIONS

The test conditions for timing parameters appearing in Table 29 on Page 29 and Table 30 on Page 30 include output disable time, output enable time, and capacitive loading. The timing specifications for the DSP apply for the voltage reference levels in Figure 29.

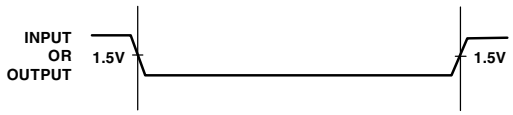

Figure 29. Voltage Reference Levels for AC Measurements (Except Output Enable/Disable)

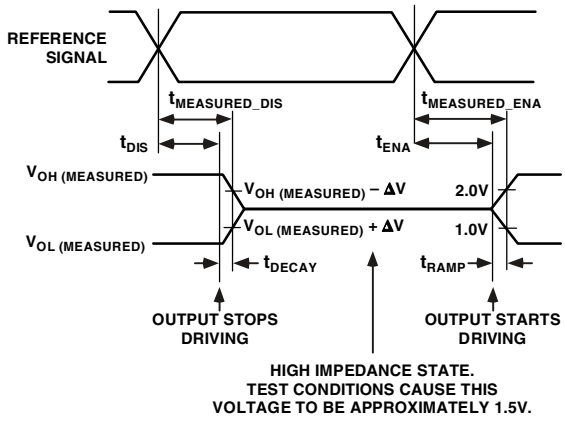

Figure 30. Output Enable/Disable

Output Disable Time

Output pins are considered to be disabled when they stop driving, go into a high impedance state, and start to decay from their output high or low voltage. The time for the voltage on the bus to decay by ΔV is dependent on the capacitive load, C_L and the load current, I_L. This decay time can be approximated by the following equation:

$$t_{DECAY} = \frac{C_L \Delta V}{I_L}$$

The output disable time t_{DIS} is the difference between $t_{MEASURED\_DIS}$ and t_{DECAY} as shown in Figure 30. The time $t_{MEASURED\_DIS}$ is the interval from when the reference signal switches to when the output voltage decays ΔV from the measured output high or output low voltage. The t_{DECAY} value is calculated with test loads C_L and I_L, and with ΔV equal to 0.5 V.

Output Enable Time

Output pins are considered to be enabled when they have made a transition from a high impedance state to when they start driving. The time for the voltage on the bus to ramp by ΔV is dependent on the capacitive load, C_L, and the drive current, I_D. This ramp time can be approximated by the following equation:

$$t_{RAMP} = \frac{C_L \Delta V}{I_D}$$

The output enable time t_{ENA} is the difference between $t_{MEASURED\_ENA}$ and t_{RAMP} as shown in Figure 30. The time $t_{MEASURED\_ENA}$ is the interval from when the reference signal switches to when the output voltage ramps ΔV from the measured three-stated output level. The t_{RAMP} value is calculated with test load C_L, drive current I_D, and with ΔV equal to 0.5 V.

Capacitive Loading

Figure 31 shows the circuit with variable capacitance that is used for measuring typical output rise and fall times. Figure 32 through Figure 39 show how output rise time varies with capacitance. Figure 40 graphically shows how output valid varies with load capacitance. (Note that this graph or derating does not apply to output disable delays; see Output Disable Time on Page 34.) The graphs of Figure 32 through Figure 40 may not be linear outside the ranges shown.

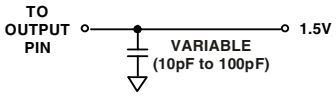

Figure 31. Equivalent Device Loading for AC Measurements (Includes All Fixtures)

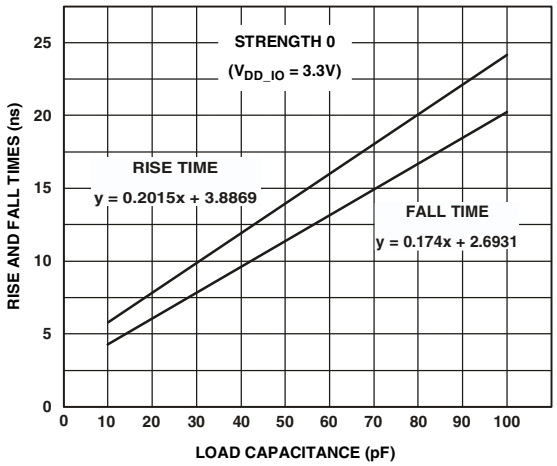

Figure 32. Typical Output Rise and Fall Time (10%–90%, V_{DD_IO} = 3.3 V) vs. Load Capacitance at Strength 0

Table 35. 484-Ball (19 mm × 19 mm) PBGA Pin Assignments (Continued)

Pin No.	Mnemonic	Pin No.	Mnemonic	Pin No.	Mnemonic	Pin No.	Mnemonic	Pin No.	Mnemonic
AA1	DATA44	AA10	L2DAT3	AA19	SDA10	AB6	DATA62	AB15	$\overline{BR5}$
AA2	DATA50	AA11	L2DAT7	AA20	ADDR10	AB7	L2DAT1	AB16	$\overline{BOFF}$
AA3	DATA47	AA12	$\overline{BR2}$	AA21	ADDR13	AB8	L2DAT2	AB17	ADDR3
AA4	DATA49	AA13	$\overline{BR6}$	AA22	ADDR15	AB9	L2DAT6	AB18	ADDR4
AA5	DATA51	AA14	$\overline{HBR}$	AB1	V_{SS}	AB10	L2CLKIN	AB19	ADDR6
AA6	DATA54	AA15	$\overline{DPA}$	AB2	DATA53	AB11	L2DIR	AB20	ADDR7
AA7	DATA57	AA16	ADDR2	AB3	DATA55	AB12	$\overline{BR0}$	AB21	ADDR9
AA8	DATA61	AA17	ADDR5	AB4	DATA56	AB13	$\overline{BR1}$	AB22	V_{SS}
AA9	L2DAT0	AA18	ADDR8	AB5	DATA59	AB14	$\overline{BR3}$		

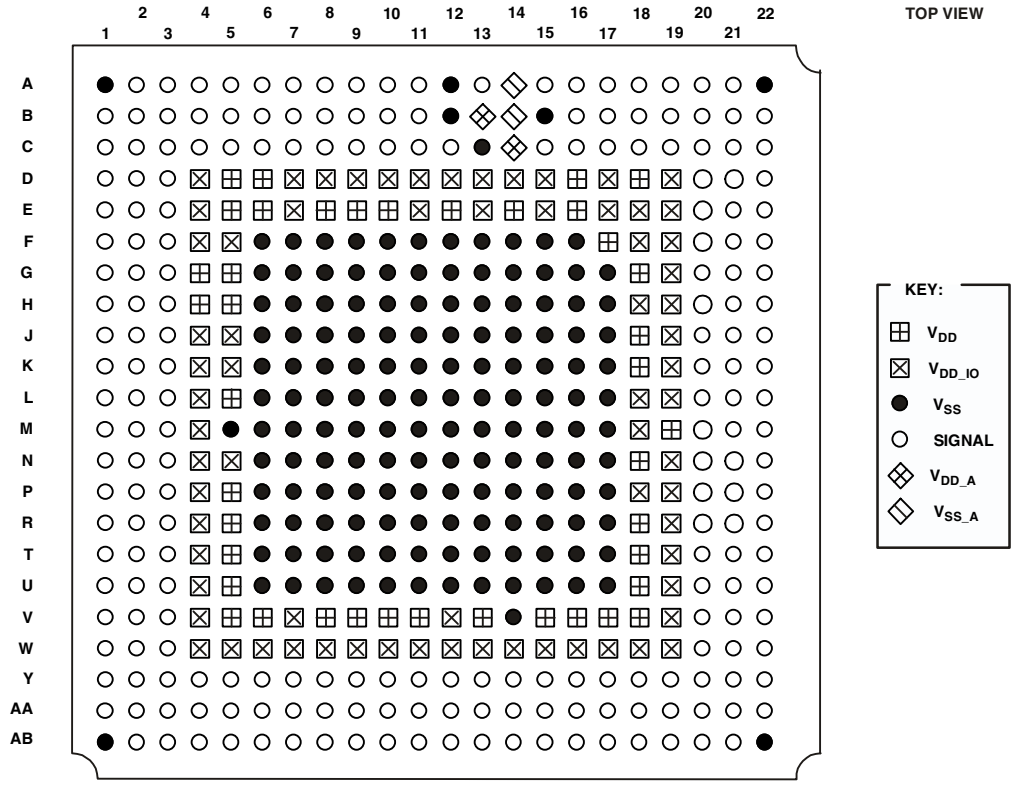

Figure 41. 484-Ball PBGA Pin Configurations (Top View, Summary)