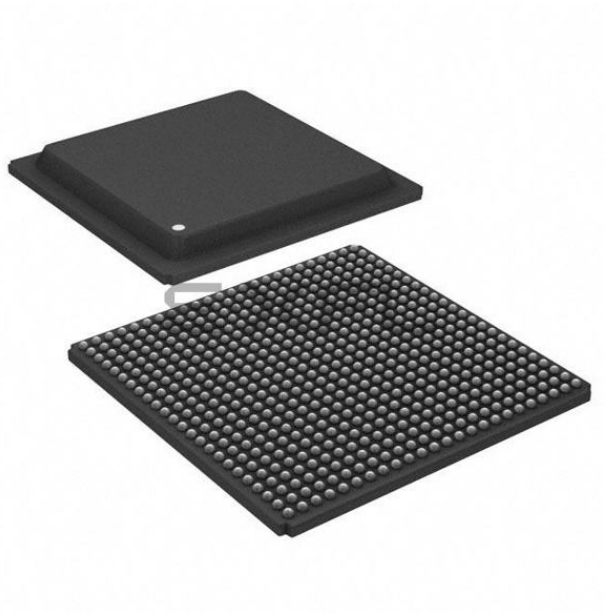


Welcome to E-XFL.COM

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	Obsolete
Type	Fixed/Floating Point
Interface	Host Interface, Link Port, Multi-Processor
Clock Rate	500MHz
Non-Volatile Memory	External
On-Chip RAM	3MB
Voltage - I/O	2.50V
Voltage - Core	1.05V
Operating Temperature	-40°C ~ 85°C (TC)
Mounting Type	Surface Mount
Package / Case	576-BBGA
Supplier Device Package	576-BGA-ED (25x25)
Purchase URL	https://www.e-xfl.com/product-detail/analog-devices/adsp-ts201sabp-050

The TigerSHARC DSP uses a Static Superscalar™† architecture. This architecture is superscalar in that the ADSP-TS201S processor's core can execute simultaneously from one to four 32-bit instructions encoded in a very large instruction word (VLIW) instruction line using the DSP's dual compute blocks. Because the DSP does not perform instruction re-ordering at runtime—the programmer selects which operations will execute in parallel prior to runtime—the order of instructions is static.

With few exceptions, an instruction line, whether it contains one, two, three, or four 32-bit instructions, executes with a throughput of one cycle in a 10-deep processor pipeline.

For optimal DSP program execution, programmers must follow the DSP's set of instruction parallelism rules when encoding an instruction line. In general, the selection of instructions that the DSP can execute in parallel each cycle depends on the instruction line resources each instruction requires and on the source and destination registers used in the instructions. The programmer has direct control of three core components—the IALUs, the compute blocks, and the program sequencer.

The ADSP-TS201S processor, in most cases, has a two-cycle 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.

In addition, the ADSP-TS201S processor supports SIMD operations two ways—SIMD compute blocks and SIMD computations. The programmer can load both compute blocks with the same data (broadcast distribution) or different data (merged distribution).

DUAL COMPUTE BLOCKS

The ADSP-TS201S processor has compute blocks that can execute computations either independently or together as a single-instruction, multiple-data (SIMD) engine. The DSP can issue up to two compute instructions per compute block each cycle, instructing the ALU, multiplier, shifter, or CLU to perform independent, simultaneous operations. Each compute block can execute eight 8-bit, four 16-bit, two 32-bit, or one 64-bit SIMD computations in parallel with the operation in the other block. These computation units support IEEE 32-bit single-precision floating-point, extended-precision 40-bit floating point, and 8-, 16-, 32-, and 64-bit fixed-point processing.

The compute blocks are referred to as X and Y in assembly syntax, and each block contains four computational units—an ALU, a multiplier, a 64-bit shifter, a 128-bit CLU—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), in sets of two (dual-aligned), or in sets of 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.
- Communications Logic Unit (CLU)—this 128-bit unit provides trellis decoding (for example, Viterbi and Turbo decoders) and executes complex correlations for CDMA communication applications (for example, chip-rate and symbol-rate functions).

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 3.6G FLOPS or 14.4G/s regular operations performance at 600 MHz
- 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 ALU (IALU)

The ADSP-TS201S processor has two IALUs that provide powerful address generation capabilities and perform many general-purpose integer operations. The IALUs are referred to as J and K in assembly syntax and have the following features:

- Provide memory addresses for data and update pointers
- Support circular buffering and bit-reverse addressing
- Perform general-purpose integer operations, increasing programming flexibility
- Include 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.

† Static Superscalar is a trademark of Analog Devices, Inc.

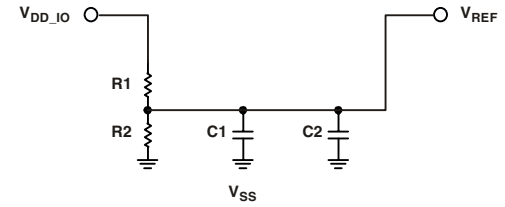
POWER DOMAINS

The ADSP-TS201S processor has separate power supply connections for internal logic (V_{DD}), analog circuits (V_{DD_A}), I/O buffer (V_{DD_IO}), and internal DRAM (V_{DD_DRAM}) power supply.

Note that the analog (V_{DD_A}) supply powers the clock generator PLLs. To produce a stable clock, systems must provide a clean power supply to power input V_{DD_A}. Designs must pay critical attention to bypassing the V_{DD_A} supply.

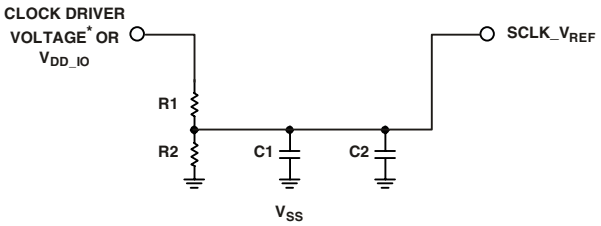
FILTERING REFERENCE VOLTAGE AND CLOCKS

Figure 6 and Figure 7 show possible circuits for filtering V_{REF}, and SCLK_V_{REF}. These circuits provide the reference voltages for the switching voltage reference and system clock reference.



R1: 2kΩ SERIES RESISTOR (±1%)
R2: 2.55kΩ SERIES RESISTOR (±1%)
C1: 1µF CAPACITOR (SMD)
C2: 1nF CAPACITOR (HF SMD) PLACED CLOSE TO DSP'S PINS

Figure 6. V_{REF} Filtering Scheme



R1: 2kΩ SERIES RESISTOR (±1%)
R2: 2.55kΩ SERIES RESISTOR (±1%)
C1: 1µF CAPACITOR (SMD)
C2: 1nF CAPACITOR (HF SMD) PLACED CLOSE TO DSP'S PINS
*IF CLOCK DRIVER VOLTAGE > V_{DD_IO}

Figure 7. SCLK_V_{REF} Filtering Scheme

DEVELOPMENT TOOLS

The ADSP-TS201S processor is supported with a complete set of CROSSCORE®† software and hardware development tools, including Analog Devices emulators and VisualDSP++®‡ development environment. The same emulator hardware that supports other TigerSHARC processors also fully emulates the ADSP-TS201S processor.

The VisualDSP++ project management environment lets programmers develop and debug an application. This environment includes an easy to use assembler (which is based on an algebraic syntax), an archiver (librarian/library builder), a linker, a loader, a cycle-accurate instruction-level simulator, a C/C++ compiler, and a C/C++ run-time library that includes DSP and mathematical functions. A key point for theses tools is C/C++ code efficiency. The compiler has been developed for efficient translation of C/C++ code to DSP assembly. The DSP has architectural features that improve the efficiency of compiled C/C++ code.

The VisualDSP++ debugger has a number of important features. Data visualization is enhanced by a plotting package that offers a significant level of flexibility. This graphical representation of user data enables the programmer to quickly determine the performance of an algorithm. As algorithms grow in complexity, this capability can have increasing significance on the designer's development schedule, increasing productivity. Statistical profiling enables the programmer to nonintrusively poll the processor as it is running the program. This feature, unique to VisualDSP++, enables the software developer to passively gather important code execution metrics without interrupting the real-time characteristics of the program. Essentially, the developer can identify bottlenecks in software quickly and efficiently. By using the profiler, the programmer can focus on those areas in the program that impact performance and take corrective action.

Debugging both C/C++ and assembly programs with the VisualDSP++ debugger, programmers can:

- View mixed C/C++ and assembly code (interleaved source and object information)
- Insert breakpoints
- Set conditional breakpoints on registers, memory, and stacks
- Trace instruction execution
- Perform linear or statistical profiling of program execution
- Fill, dump, and graphically plot the contents of memory
- Perform source level debugging
- Create custom debugger windows

The VisualDSP++ IDE lets programmers define and manage DSP software development. Its dialog boxes and property pages let programmers configure and manage all of the TigerSHARC processor development tools, including the color syntax highlighting in the VisualDSP++ editor. This capability permits programmers to:

- Control how the development tools process inputs and generate outputs
- Maintain a one-to-one correspondence with the tool's command line switches

The VisualDSP++ Kernel (VDK) incorporates scheduling and resource management tailored specifically to address the memory and timing constraints of DSP programming. These capabilities enable engineers to develop code more effectively,

† CROSSCORE is a registered trademark of Analog Devices, Inc.
‡ VisualDSP++ is a registered trademark of Analog Devices, Inc.

eliminating the need to start from the very beginning when developing new application code. The VDK features include threads, critical and unscheduled regions, semaphores, events, and device flags. The VDK also supports priority-based, preemptive, cooperative, and time-sliced scheduling approaches. In addition, the VDK was designed to be scalable. If the application does not use a specific feature, the support code for that feature is excluded from the target system.

Because the VDK is a library, a developer can decide whether to use it or not. The VDK is integrated into the VisualDSP++ development environment, but can also be used via standard command line tools. When the VDK is used, the development environment assists the developer with many error-prone tasks and assists in managing system resources, automating the generation of various VDK-based objects, and visualizing the system state, when debugging an application that uses the VDK.

VCSE is Analog Devices' technology for creating, using, and reusing software components (independent modules of substantial functionality) to quickly and reliably assemble software applications. It also is used for downloading components from the Web, dropping them into the application, and publishing component archives from within VisualDSP++. VCSE supports component implementation in C/C++ or assembly language.

Use the expert linker to visually manipulate the placement of code and data on the embedded system, view memory use in a color-coded graphical form, easily move code and data to different areas of the DSP or external memory with a drag of the mouse, and examine runtime stack and heap usage. The expert linker is fully compatible with existing linker definition file (LDF), allowing the developer to move between the graphical and textual environments.

Analog Devices DSP emulators use the IEEE 1149.1 JTAG test access port of the ADSP-TS201S processor to monitor and control the target board processor during emulation. The emulator provides full speed emulation, allowing inspection and modification of memory, registers, and processor stacks. Nonintrusive in-circuit emulation is assured by the use of the processor's JTAG interface—the emulator does not affect target system loading or timing.

In addition to the software and hardware development tools available from Analog Devices, third parties provide a wide range of tools supporting the TigerSHARC processor family. Hardware tools include TigerSHARC processor PC plug-in cards. Third party software tools include DSP libraries, real-time operating systems, and block diagram design tools.

EVALUATION KIT

Analog Devices offers a range of EZ-KIT Lite®† evaluation platforms to use as a cost-effective method to learn more about developing or prototyping applications with Analog Devices processors, platforms, and software tools. Each EZ-KIT Lite includes an evaluation board along with an evaluation suite of the VisualDSP++ development and debugging environment with the C/C++ compiler, assembler, and linker. Also included are sample application programs, power supply, and a USB cable. All evaluation versions of the software tools are limited for use only with the EZ-KIT Lite product.

The USB controller on the EZ-KIT Lite board connects the board to the USB port of the user's PC, enabling the VisualDSP++ evaluation suite to emulate the on-board processor in-circuit. This permits the customer to download, execute, and debug programs for the EZ-KIT Lite system. It also allows in-circuit programming of the on-board flash device to store user-specific boot code, enabling the board to run as a standalone unit, without being connected to the PC.

With a full version of VisualDSP++ installed (sold separately), engineers can develop software for the EZ-KIT Lite or any custom-defined system. Connecting one of Analog Devices JTAG emulators to the EZ-KIT Lite board enables high speed, nonintrusive emulation.

DESIGNING AN EMULATOR-COMPATIBLE DSP BOARD (TARGET)

The Analog Devices family of emulators are tools that every DSP developer needs in order to test and debug hardware and software systems. Analog Devices has supplied an IEEE 1149.1 JTAG test access port (TAP) on each JTAG DSP. The emulator uses the TAP to access the internal features of the DSP, allowing the developer to load code, set breakpoints, observe variables, observe memory, and examine registers. The DSP must be halted to send data and commands, but once an operation has been completed by the emulator, the DSP system is set running at full speed with no impact on system timing.

To use these emulators, the target board must include a header that connects the DSP's JTAG port to the emulator.

For details on target board design issues including mechanical layout, single processor connections, multiprocessor scan chains, signal buffering, signal termination, and emulator pod logic, see the *EE-68: Analog Devices JTAG Emulation Technical Reference* on the Analog Devices website (www.analog.com)—use the string "EE-68" in site search. This document is updated regularly to keep pace with improvements to emulator support.

ADDITIONAL INFORMATION

This data sheet provides a general overview of the ADSP-TS201S processor's architecture and functionality. For detailed information on the ADSP-TS201S processor's core architecture and instruction set, see the *ADSP-TS201 TigerSHARC Processor Hardware Reference* and the *ADSP-TS201 TigerSHARC Processor Programming Reference*. For detailed information on the development tools for this processor, see the *VisualDSP++ User's Guide for TigerSHARC Processors*.

† EZ-Kit Lite is a registered trademark of Analog Devices, Inc.

Table 6. Pin Definitions—External Port Arbitration

Signal	Type	Term	Description
$\overline{BR7–0}$	I/O	V_{DD_IO}[1]	Multiprocessing Bus Request Pins. Used by the DSPs in a multiprocessor system to arbitrate for bus mastership. Each DSP drives its own $\overline{BRx}$ line (corresponding to the value of its ID2–0 inputs) and monitors all others. In systems with fewer than eight DSPs, set the unused $\overline{BRx}$ pins high (V_{DD_IO}).
ID2–0	I (pd)	na	Multiprocessor ID. Indicates the DSP's ID, from which the DSP determines its order in a multiprocessor system. These pins also indicate to the DSP which bus request ($\overline{BR0}$–$\overline{BR7}$) to assert when requesting the bus: 000 = $\overline{BR0}$, 001 = $\overline{BR1}$, 010 = $\overline{BR2}$, 011 = $\overline{BR3}$, 100 = $\overline{BR4}$, 101 = $\overline{BR5}$, 110 = $\overline{BR6}$, or 111 = $\overline{BR7}$. ID2–0 must have a constant value during system operation and can change during reset only.
$\overline{BM}$	O	na	Bus Master. The current bus master DSP asserts $\overline{BM}$. For debugging only. At reset this is a strap pin. For more information, see Table 16 on Page 20.
$\overline{BOFF}$	I	epu	Back Off. A deadlock situation can occur when the host and a DSP try to read from each other's bus at the same time. When deadlock occurs, the host can assert $\overline{BOFF}$ to force the DSP to relinquish the bus before completing its outstanding transaction.
$\overline{BUSLOCK}$	O/T (pu_0)	na	Bus Lock Indication. Provides an indication that the current bus master has locked the bus. At reset, this is a strap pin. For more information, see Table 16 on Page 20.
$\overline{HBR}$	I	epu	Host Bus Request. A host must assert $\overline{HBR}$ to request control of the DSP's external bus. When $\overline{HBR}$ is asserted in a multiprocessing system, the bus master relinquishes the bus and asserts $\overline{HBG}$ once the outstanding transaction is finished.
$\overline{HBG}$	I/O/T (pu_0)	epu[2]	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, $\overline{MSH}$, $\overline{MSSD3–0}$, $\overline{MS1–0}$, $\overline{RD}$, $\overline{WRL}$, $\overline{WRH}$, $\overline{BMS}$, $\overline{BRST}$, $\overline{IORD}$, $\overline{IOWR}$, $\overline{IOEN}$, $\overline{RAS}$, $\overline{CAS}$, $\overline{SDWE}$, 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 it.
$\overline{CPA}$	I/O/OD (pu_od_0)	epu[2]	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. If not required in the system, leave $\overline{CPA}$ unconnected (external pull-ups will be required for DSP ID = 1 through ID = 7).
$\overline{DPA}$	I/O/OD (pu_od_0)	epu[2]	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. If not required in the system, leave $\overline{DPA}$ unconnected (external pull-ups will be required for DSP ID = 1 through ID = 7).

I = input; **A** = asynchronous; **O** = output; **OD** = open-drain output; **T** = three-state; **P** = power supply; **G** = ground; **pd** = internal pull-down 5 kΩ; **pu** = internal pull-up 5 kΩ; **pd_0** = internal pull-down 5 kΩ on DSP ID = 0; **pu_0** = internal pull-up 5 kΩ on DSP ID = 0; **pu_od_0** = internal pull-up 500 Ω on DSP ID = 0; **pd_m** = internal pull-down 5 kΩ on DSP bus master; **pu_m** = internal pull-up 5 kΩ on DSP bus master; **pu_ad** = internal pull-up 40 kΩ. For more pull-down and pull-up information, see Electrical Characteristics on Page 22.

Term (termination of unused pins) column symbols: epd = external pull-down approximately 5 kΩ to V_{SS}; epu = external pull-up approximately 5 kΩ to V_{DD_IO}, nc = not connected; na = not applicable (always used); V_{DD_IO} = connect directly to V_{DD_IO}; V_{SS} = connect directly to V_{SS}

[1] The $\overline{BRx}$ pin matching the ID2–0 input selection for the processor should be left nc if unused. For example, the processor with ID = 000 has $\overline{BR0}$ = nc and $\overline{BR7–1}$ = V_{DD_IO}.
[2] This external pull-up resistor may be omitted for the ID = 000 TigerSHARC processor.

Table 7. Pin Definitions—External Port DMA/Flyby

Signal	Type	Term	Description
$\overline{\text{DMAR3–0}}$	I/A	epu	DMA Request Pins. Enable external I/O devices to request DMA services from the DSP. In response to $\overline{\text{DMARx}}$, the DSP performs DMA transfers according to the DMA channel's initialization. The DSP ignores DMA requests from uninitialized channels.
$\overline{\text{IOWR}}$	O/T (pu_0)	nc	I/O Write. When a DSP DMA channel initiates a flyby mode read transaction, the DSP asserts the $\overline{\text{IOWR}}$ signal during the data cycles. This assertion makes the I/O device sample the data instead of the TigerSHARC.
$\overline{\text{IORD}}$	O/T (pu_0)	nc	I/O Read. When a DSP DMA channel initiates a flyby mode write transaction, the DSP asserts the $\overline{\text{IORD}}$ signal during the data cycle. This assertion with the $\overline{\text{IOEN}}$ makes the I/O device drive the data instead of the TigerSHARC.
$\overline{\text{IOEN}}$	O/T (pu_0)	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.

I = input; **A** = asynchronous; **O** = output; **OD** = open-drain output; **T** = three-state; **P** = power supply; **G** = ground; **pd** = internal pull-down 5 kΩ; **pu** = internal pull-up 5 kΩ; **pd_0** = internal pull-down 5 kΩ on DSP ID = 0; **pu_0** = internal pull-up 5 kΩ on DSP ID = 0; **pu_od_0** = internal pull-up 500 Ω on DSP ID = 0; **pd_m** = internal pull-down 5 kΩ on DSP bus master; **pu_m** = internal pull-up 5 kΩ on DSP bus master; **pu_ad** = internal pull-up 40 kΩ. For more pull-down and pull-up information, see Electrical Characteristics on Page 22.

Term (termination of unused pins) column symbols: epd = external pull-down approximately 5 kΩ to V_{SS}; epu = external pull-up approximately 5 kΩ to V_{DD_IO}, nc = not connected; na = not applicable (always used); V_{DD_IO} = connect directly to V_{DD_IO}; V_{SS} = connect directly to V_{SS}

Table 11. Pin Definitions—Link Ports

Signal	Type	Term	Description
LxDATO3–0P	O	nc	Link Ports 3–0 Data 3–0 Transmit LVDS P
LxDATO3–0N	O	nc	Link Ports 3–0 Data 3–0 Transmit LVDS N
LxCLKOUTP	O	nc	Link Ports 3–0 Transmit Clock LVDS P
LxCLKOUTN	O	nc	Link Ports 3–0 Transmit Clock LVDS N
LxACKI	I (pd)	nc	Link Ports 3–0 Receive Acknowledge. Using this signal, the receiver indicates to the transmitter that it may continue the transmission.
$\overline{\text{LxBCMPO}}$	O (pu)	nc	Link Ports 3–0 Block Completion. When the transmission is executed using DMA, this signal indicates to the receiver that the transmitted block is completed. The pull-up resistor is present on $\overline{\text{L0BCMPO}}$ only. At reset, the $\overline{\text{L1BCMPO}}$, $\overline{\text{L2BCMPO}}$, and $\overline{\text{L3BCMPO}}$ pins are strap pins. For more information, see Table 16 on Page 20.
LxDATI3–0P	I	V_{DD_IO}	Link Ports 3–0 Data 3–0 Receive LVDS P
LxDATI3–0N	I	V_{DD_IO}	Link Ports 3–0 Data 3–0 Receive LVDS N
LxCLKINP	I/A	V_{DD_IO}	Link Ports 3–0 Receive Clock LVDS P
LxCLKINN	I/A	V_{DD_IO}	Link Ports 3–0 Receive Clock LVDS N
LxACKO	O	nc	Link Ports 3–0 Transmit Acknowledge. Using this signal, the receiver indicates to the transmitter that it may continue the transmission.
$\overline{\text{LxBCMPI}}$	I (pd_l)	V_{SS}	Link Ports 3–0 Block Completion. When the reception is executed using DMA, this signal indicates to the receiver that the transmitted block is completed.

I = input; **A** = asynchronous; **O** = output; **OD** = open-drain output; **T** = three-state; **P** = power supply; **G** = ground; **pd** = internal pull-down 5 kΩ; **pu** = internal pull-up 5 kΩ; **pd_0** = internal pull-down 5 kΩ on DSP ID = 0; **pu_0** = internal pull-up 5 kΩ on DSP ID = 0; **pu_od_0** = internal pull-up 500 Ω on DSP ID = 0; **pd_m** = internal pull-down 5 kΩ on DSP bus master; **pu_m** = internal pull-up 5 kΩ on DSP bus master; **pu_ad** = internal pull-up 40 kΩ; **pd_l** = internal pull-down 50 kΩ. For more pull-down and pull-up information, see Electrical Characteristics on Page 22.

Term (termination of unused pins) column symbols: epd = external pull-down approximately 5 kΩ to V_{SS}; epu = external pull-up approximately 5 kΩ to V_{DD_IO}, nc = not connected; na = not applicable (always used); V_{DD_IO} = connect directly to V_{DD_IO}; V_{SS} = connect directly to V_{SS}

Table 12. Pin Definitions—Impedance Control, Drive Strength Control, and Regulator Enable

Signal	Type	Term	Description
CONTROLIMP0 CONTROLIMP1	I (pd) I (pu)	na na	Impedance Control. As shown in Table 13, the CONTROLIMP1–0 pins select between normal driver mode and A/D driver mode. When using normal mode (recommended), the output drive strength is set relative to maximum drive strength according to Table 14. When using A/D mode, the resistance control operates in the analog mode, where drive strength is continuously controlled to match a specific line impedance as shown in Table 14.
DS2, 0 DS1	I (pu) I (pd)	na	Digital Drive Strength Selection. Selected as shown in Table 14. For drive strength calculation, see Output Drive Currents on Page 36. The drive strength for some pins is preset, not controlled by the DS2–0 pins. The pins that are always at drive strength 7 (100%) include: $\overline{\text{CPA}}$, $\overline{\text{DPA}}$, TDO, $\overline{\text{EMU}}$, and $\overline{\text{RST\_OUT}}$. The drive strength for the ACK pin is always x2 drive strength 7 (100%).
ENEDREG	I (pu)	V_{SS}	Connect the ENEDREG pin to V_{SS}. Connect the V_{DD_DRAM} pins to a properly decoupled DRAM power supply.

I = input; **A** = asynchronous; **O** = output; **OD** = open-drain output; **T** = three-state; **P** = power supply; **G** = ground; **pd** = internal pull-down 5 kΩ; **pu** = internal pull-up 5 kΩ; **pd_0** = internal pull-down 5 kΩ on DSP ID = 0; **pu_0** = internal pull-up 5 kΩ on DSP ID = 0; **pu_od_0** = internal pull-up 500 Ω on DSP ID = 0; **pd_m** = internal pull-down 5 kΩ on DSP bus master; **pu_m** = internal pull-up 5 kΩ on DSP bus master; **pu_ad** = internal pull-up 40 kΩ. For more pull-down and pull-up information, see Electrical Characteristics on Page 22.

Term (termination of unused pins) column symbols: epd = external pull-down approximately 5 kΩ to V_{SS}; epu = external pull-up approximately 5 kΩ to V_{DD_IO}, nc = not connected; na = not applicable (always used); V_{DD_IO} = connect directly to V_{DD_IO}; V_{SS} = connect directly to V_{SS}

ADSP-TS201S—SPECIFICATIONS

Note that component specifications are subject to change without notice. For information on link port electrical characteristics, see Link Port Low Voltage, Differential-Signal (LVDS) Electrical Characteristics, and Timing on Page 30.

OPERATING CONDITIONS

Parameter	Description	Test Conditions	Grade[1]	Min	Typ	Max	Unit
V_{DD}	Internal Supply Voltage	@ CCLK = 600 MHz	060	1.14	1.20	1.26	V
		@ CCLK = 500 MHz	050	1.00	1.05	1.10	V
V_{DD_A}	Analog Supply Voltage	@ CCLK = 600 MHz	060	1.14	1.20	1.26	V
		@ CCLK = 500 MHz	050	1.00	1.05	1.10	V
V_{DD_IO}	I/O Supply Voltage		(all)	2.38	2.50	2.63	V
V_{DD_DRAM}	Internal DRAM Supply Voltage	@ CCLK = 600 MHz	060	1.52	1.60	1.68	V
		@ CCLK = 500 MHz	050	1.425	1.500	1.575	V
T_{CASE}	Case Operating Temperature		A	–40		+85	°C
T_{CASE}	Case Operating Temperature		W	–40		+105	°C
V_{IH1}	High Level Input Voltage[2, 3]	@ V_{DD}, V_{DD_IO} = Max	(all)	1.7		3.63	V
V_{IH2}	High Level Input Voltage[3, 4]	@ V_{DD}, V_{DD_IO} = Max	(all)	1.9		3.63	V
V_{IL}	Low Level Input Voltage[3, 5]	@ V_{DD}, V_{DD_IO} = Min	(all)	–0.33		+0.8	V
I_{DD}	V_{DD} Supply Current, Typical Activity[6]	@ CCLK = 600 MHz, V_{DD} = 1.20 V, T_{CASE} = 25°C	060		2.90		A
		@ CCLK = 500 MHz, V_{DD} = 1.05 V, T_{CASE} = 25°C	050		2.06		A
I_{DD_A}	V_{DD_A} Supply Current, Typical Activity	@ CCLK = 600 MHz, V_{DD} = 1.20 V, T_{CASE} = 25°C	060		25	55	mA
		@ CCLK = 500 MHz, V_{DD} = 1.05 V, T_{CASE} = 25°C	050		20	50	mA
I_{DD_IO}	V_{DD_IO} Supply Current, Typical Activity[6]	@ SCLK = 62.5 MHz, V_{DD_IO} = 2.5 V, T_{CASE} = 25°C	(all)		0.15		A
I_{DD_DRAM}	V_{DD_DRAM} Supply Current, Typical Activity[6]	@ CCLK = 600 MHz, V_{DD_DRAM} = 1.6 V, T_{CASE} = 25°C	060		0.28	0.43	A
		@ CCLK = 500 MHz, V_{DD_DRAM} = 1.5 V, T_{CASE} = 25°C	050		0.25	0.40	A
V_{REF}	Voltage Reference		(all)	(V_{DD_IO} × 0.56) ±5%			V
SCLK_V_{REF}	Voltage Reference		(all)	(V_{CLOCK_DRIVE} × 0.56) ±5%			V

[1] Specifications vary for different grades (for example, SABP-060, SABP-050, SWBP-050). For more information on part grades, see Ordering Guide on Page 46.

[2] V_{IH1} specification applies to input and bidirectional pins: SCLKRAT2–0, SCLK, ADDR31–0, DATA63–0, $\overline{RD}$, $\overline{WRL}$, $\overline{WRH}$, ACK, $\overline{BRST}$, $\overline{BR7–0}$, $\overline{BOFF}$, $\overline{HBR}$, $\overline{HBG}$, $\overline{MSSD3–0}$, $\overline{RAS}$, $\overline{CAS}$, SDCKE, $\overline{SDWE}$, TCK, FLAG3–0, DS2–0, ENEDREG.

[3] Values represent dc case. During transitions, the inputs may overshoot or undershoot to the voltage shown in Table 18, based on the transient duty cycle. The dc case is equivalent to 100% duty cycle.

[4] V_{IH2} specification applies to input and bidirectional pins: TDI, TMS, TRST, CIMP1–0, ID2–0, $\overline{LxBCMPI}$, LxACKI, $\overline{POR\_IN}$, $\overline{RST\_IN}$, $\overline{IRQ3–0}$, $\overline{CPA}$, $\overline{DPA}$, $\overline{DMAR3–0}$.

[5] Applies to input and bidirectional pins.

[6] For details on internal and external power calculation issues, including other operating conditions, see the *EE-170, Estimating Power for the ADSP-TS201S* on the Analog Devices website.

Table 18. Maximum Duty Cycle for Input Transient Voltage

V_{IN} Max (V)[1]	V_{IN} Min (V)[1]	Maximum Duty Cycle[2]
+3.63	–0.33	100%
+3.64	–0.34	90%
+3.70	–0.40	50%
+3.78	–0.48	30%
+3.86	–0.56	17%
+3.93	–0.63	10%

[1] The individual values cannot be combined for analysis of a single instance of overshoot or undershoot. The worst case observed value must fall within one of the voltages specified and the total duration of the overshoot or undershoot (exceeding the 100% case) must be less than or equal to the corresponding duty cycle.

[2] Duty cycle refers to the percentage of time the signal exceeds the value for the 100% case. This is equivalent to the measured duration of a single instance of overshoot or undershoot as a percentage of the period of occurrence. The practical worst case for period of occurrence for either overshoot or undershoot is $2 \times t_{SCLK}$.

ELECTRICAL CHARACTERISTICS

Parameter	Description	Test Conditions	Min	Max	Unit
V_{OH}	High Level Output Voltage[1]	@V_{DD_IO} = Min, I_{OH} = –2 mA	2.18		V
V_{OL}	Low Level Output Voltage[1]	@V_{DD_IO} = Min, I_{OL} = 4 mA		0.4	V
I_{IH}	High Level Input Current	@V_{DD_IO} = Max, V_{IN} = V_{IH} Max		20	µA
I_{IH_PU}	High Level Input Current	@V_{DD_IO} = Max, V_{IN} = V_{IH} Max		20	µA
I_{IH_PD}	High Level Input Current	@V_{DD_IO} = Max, V_{IN} = V_{DD_IO} Max	0.3	0.76	mA
$I_{IH\_PD\_L}$	High Level Input Current	@V_{DD_IO} = Max, V_{IN} = V_{IH} Max	30	76	µA
I_{IL}	Low Level Input Current	@V_{DD_IO} = Max, V_{IN} = 0 V		20	µA
I_{IL_PU}	Low Level Input Current	@V_{DD_IO} = Max, V_{IN} = 0 V	0.3	0.76	mA
$I_{IL\_PU\_AD}$	Low Level Input Current	@V_{DD_IO} = Max, V_{IN} = 0 V	30	100	µA
I_{OZH}	Three-State Leakage Current High	@V_{DD_IO} = Max, V_{IN} = V_{IH} Max		50	µA
I_{OZH_PD}	Three-State Leakage Current High	@V_{DD_IO} = Max, V_{IN} = V_{DD_IO} Max	0.3	0.76	mA
I_{OZL}	Three-State Leakage Current Low	@V_{DD_IO} = Max, V_{IN} = 0 V		20	µA
I_{OZL_PU}	Three-State Leakage Current Low	@V_{DD_IO} = Max, V_{IN} = 0 V	0.3	0.76	mA
$I_{OZL\_PU\_AD}$	Three-State Leakage Current Low	@V_{DD_IO} = Max, V_{IN} = 0 V	30	100	µA
I_{OZL_OD}	Three-State Leakage Current Low	@V_{DD_IO} = Max, V_{IN} = 0 V	4	7.6	mA
C_{IN}	Input Capacitance[2, 3]	@f_{IN} = 1 MHz, T_{CASE} = 25°C, V_{IN} = 2.5 V		3	pF

Parameter name suffix conventions: no suffix = applies to pins without pull-up or pull-down resistors, **_PD** = applies to pin types (pd) or (pd_0), **_PU** = applies to pin types (pu) or (pu_0), **_PU_AD** = applies to pin types (pu_ad), **_OD** = applies to pin types OD, **_PD_L** = applies to pin types (pd_l)

[1] Applies to output and bidirectional pins.

[2] Applies to all signals.

[3] Guaranteed but not tested.

Table 23. Reference Clocks—System Clock (SCLK) Cycle Time

Parameter	Description	SCLKRAT = 4×, 6×, 8×, 10×, 12×		SCLKRAT = 5×, 7×		Unit
		Min	Max	Min	Max	
t_{SCLK}[1, 2, 3]	System Clock Cycle Time	8	50	8	50	ns
t_{SCLKH}	System Clock Cycle High Time	0.40 × t_{SCLK}	0.60 × t_{SCLK}	0.45 × t_{SCLK}	0.55 × t_{SCLK}	ns
t_{SCLKL}	System Clock Cycle Low Time	0.40 × t_{SCLK}	0.60 × t_{SCLK}	0.45 × t_{SCLK}	0.55 × t_{SCLK}	ns
t_{SCLKF}	System Clock Transition Time—Falling Edge[4]	—	1.5	—	1.5	ns
t_{SCLKR}	System Clock Transition Time—Rising Edge	—	1.5	—	1.5	ns
t_{SCLKJ}[5, 6]	System Clock Jitter Tolerance	—	500	—	500	ps

[1] For more information, see Table 3 on Page 12.
[2] For more information, see Clock Domains on Page 9.
[3] The value of (t_{SCLK} / SCLKRAT2-0) must not violate the specification for t_{CCLK}.
[4] System clock transition times apply to minimum SCLK cycle time (t_{SCLK}) only.
[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.

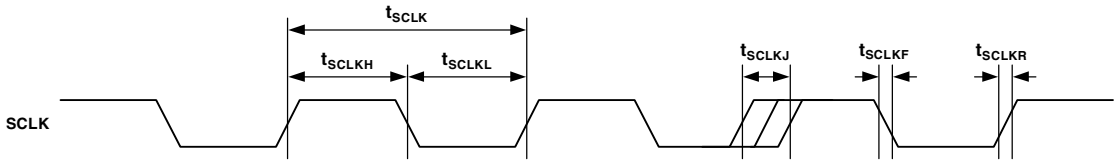


Figure 10. Reference Clocks—System Clock (SCLK) Cycle Time

Table 24. Reference Clocks—JTAG Test Clock (TCK) Cycle Time

Parameter	Description	Min	Max	Unit
t_{TCK}	Test Clock (JTAG) Cycle Time	Greater of 30 or t_{CCLK} × 4	—	ns
t_{TCKH}	Test Clock (JTAG) Cycle High Time	12	—	ns
t_{TCKL}	Test Clock (JTAG) Cycle Low Time	12	—	ns

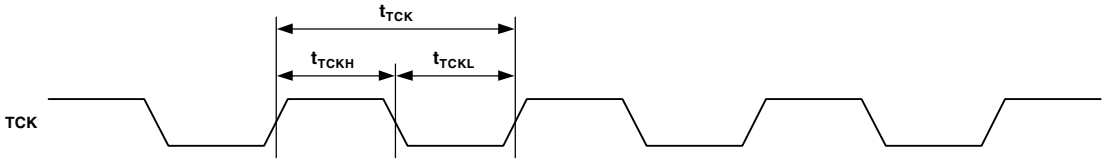


Figure 11. Reference Clocks—JTAG Test Clock (TCK) Cycle Time

Table 25. Power-Up Timing[1]

Parameter		Min	Max	Unit
Timing Requirement				
t_{VDD_DRAM}	V_{DD_DRAM} Stable After V_{DD}, V_{DD_A}, V_{DD_IO} Stable	>0		ms

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

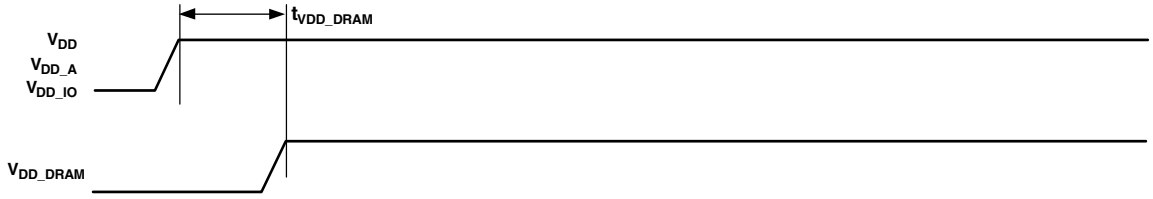


Figure 12. Power-Up Timing

Table 26. Power-Up Reset Timing

Parameter		Min	Max	Unit
Timing Requirements				
$t_{RST\_IN\_PWR}$	$\overline{RST\_IN}$ Deasserted After V_{DD}, V_{DD_A}, V_{DD_IO}, V_{DD_DRAM}, SCLK, and Static/ Strap Pins Stable	2		ms
$t_{TRST\_IN\_PWR}$[1]	$\overline{TRST}$ Asserted During Power-Up Reset	100 × t_{SCLK}		ns
Switching Characteristic				
$t_{RST\_OUT\_PWR}$	$\overline{RST\_OUT}$ Deasserted After $\overline{RST\_IN}$ Deasserted	1.5		ms

[1] Applies after V_{DD}, V_{DD_A}, V_{DD_IO}, V_{DD_DRAM}, and SCLK are stable and before $\overline{RST\_IN}$ deasserted.

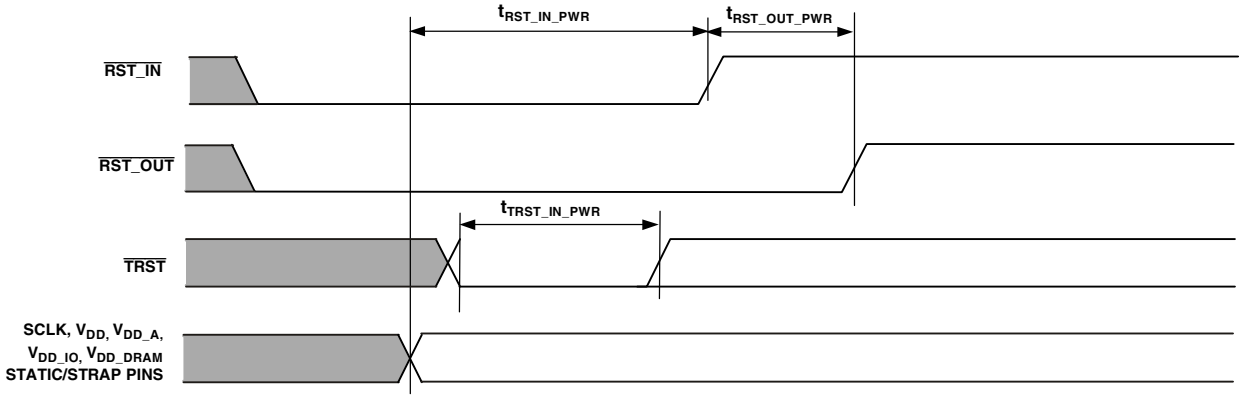


Figure 13. Power-Up Reset Timing

Table 27. Normal Reset Timing

Parameter		Min	Max	Unit
Timing Requirements				
t_{RST_IN}	$\overline{RST\_IN}$ Asserted	2		ms
t_{STRAP}	$\overline{RST\_IN}$ Deasserted After Strap Pins Stable	1.5		ms
Switching Characteristic				
t_{RST_OUT}	$\overline{RST\_OUT}$ Deasserted After $\overline{RST\_IN}$ Deasserted	1.5		ms

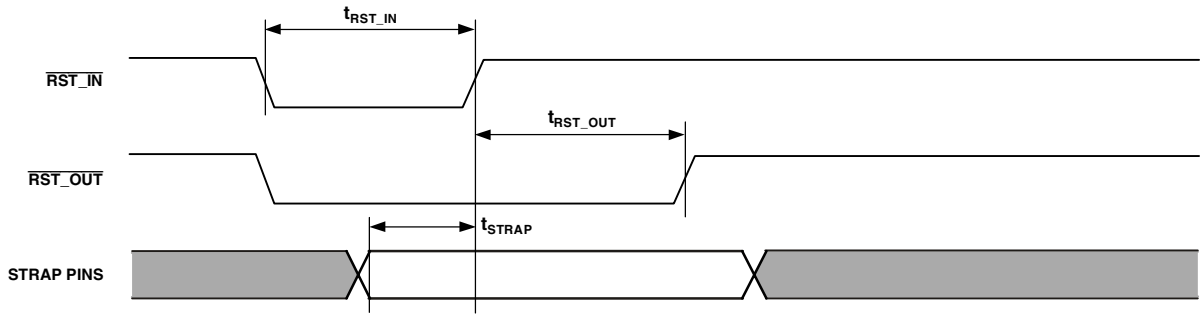


Figure 14. Normal Reset Timing

Table 28. On-Chip DRAM Refresh[1]

Parameter		Min	Max	Unit
Timing Requirement				
t_{REF}	On-chip DRAM Refresh Period		1.56	µs

[1] For more information on setting the refresh rate for the on-chip DRAM, refer to the *ADSP-TS201 TigerSHARC Processor Programming Reference.*

Table 29. AC Signal Specifications (Continued)

(All values in this table are in nanoseconds.)

Name	Description	Input Setup (Min)	Input Hold (Min)	Output Valid (Max)	Output Hold (Min)	Output Enable (Min)[1]	Output Disable (Max)[1]	Reference Clock
DS2–0[8]	Static Pins—Must Be Constant	—	—	—	—	—	—	—
SCLKRAT2–0[8]	Static Pins—Must Be Constant	—	—	—	—	—	—	—
ENEDREG	Static Pins—Must Be Connected to V_{SS}	—	—	—	—	—	—	—
STRAP SYS[9, 10]	Strap Pins	1.5	0.5	—	—	—	—	SCLK
JTAG SYS[11, 12]	JTAG System Pins	+2.5	+10.0	+12.0	–1.0	—	—	TCK

[1] The external port protocols employ bus IDLE cycles for bus mastership transitions as well as slave access boundary crossings to avoid any potential bus contention. The apparent driver overlap, due to output disables being larger than output enables, is not actual.
[2] For input specifications on FLAG3–0 pins, see Table 21.
[3] These input pins are asynchronous and therefore do not need to be synchronized to a clock reference.
[4] For additional requirement details, see Reset and Booting on Page 9.
[5] $\overline{\text{RST\_IN}}$ clock reference is the falling edge of SCLK.
[6] TDO output clock reference is the falling edge of TCK.
[7] Reference clock depends on function.
[8] These pins may change only during reset; recommend connecting it to V_{DD_IO}/V_{SS}.
[9] STRAP pins include: $\overline{\text{BMS}}$, $\overline{\text{BM}}$, $\overline{\text{BUSLOCK}}$, TMR0E, $\overline{\text{L1BCMPO}}$, $\overline{\text{L2BCMPO}}$, and $\overline{\text{L3BCMPO}}$.
[10] Specifications applicable during reset only.
[11] JTAG system pins include: $\overline{\text{RST\_IN}}$, $\overline{\text{RST\_OUT}}$, $\overline{\text{POR\_IN}}$, $\overline{\text{IRQ3–0}}$, $\overline{\text{DMAR3–0}}$, $\overline{\text{HBR}}$, $\overline{\text{BOFF}}$, $\overline{\text{MS1–0}}$, $\overline{\text{MSH}}$, SDCKE, LDQM, HDQM, $\overline{\text{BMS}}$, $\overline{\text{IOWR}}$, $\overline{\text{IORD}}$, $\overline{\text{BM}}$, $\overline{\text{EMU}}$, SDA10, $\overline{\text{IOEN}}$, $\overline{\text{BUSLOCK}}$, TMR0E, DATA63–0, ADDR31–0, $\overline{\text{RD}}$, $\overline{\text{WRL}}$, $\overline{\text{WRH}}$, $\overline{\text{BRST}}$, $\overline{\text{MSSD3–0}}$, $\overline{\text{RAS}}$, $\overline{\text{CAS}}$, $\overline{\text{SDWE}}$, $\overline{\text{HBG}}$, $\overline{\text{BR7–0}}$, FLAG3–0, L0DATOP3–0, L0DATON3–0, L1DATOP3–0, L1DATON3–0, L2DATOP3–0, L2DATON3–0, L3DATOP3–0, L3DATON3–0, L0CLKOUTP, L0CLKOUTN, L1CLKOUTP, L1CLKOUTN, L2CLKOUTP, L2CLKOUTN, L3CLKOUTP, L3CLKOUTN, L0ACKI, L1ACKI, L2ACKI, L3ACKI, L0DATIP3–0, L0DATIN3–0, L1DATIP3–0, L1DATIN3–0, L2DATIP3–0, L2DATIN3–0, L3DATIP3–0, L3DATIN3–0, L0CLKINP, L0CLKINN, L1CLKINP, L1CLKINN, L2CLKINP, L2CLKINN, L3CLKINP, L3CLKINN, L0ACKO, L1ACKO, L2ACKO, L3ACKO, ACK, $\overline{\text{CPA}}$, $\overline{\text{DPA}}$, $\overline{\text{L0BCMPO}}$, $\overline{\text{L1BCMPO}}$, $\overline{\text{L2BCMPO}}$, $\overline{\text{L3BCMPO}}$, $\overline{\text{L0BCMPI}}$, $\overline{\text{L1BCMPI}}$, $\overline{\text{L2BCMPI}}$, $\overline{\text{L3BCMPI}}$, ID2–0, CTRL_IMPD1–0, SCLKRAT2–0, DS2–0, ENEDREG.
[12] JTAG system output timing clock reference is the falling edge of TCK.

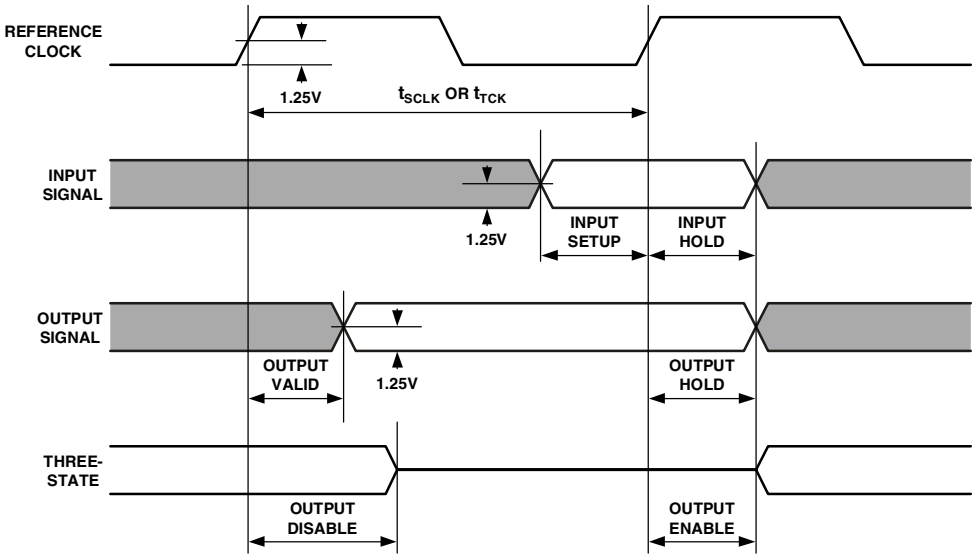


Figure 15. General AC Parameters Timing

Link Port Low Voltage, Differential-Signal (LVDS) Electrical Characteristics, and Timing

Table 30 and Table 31 with Figure 16 provide the electrical characteristics for the LVDS link ports. The LVDS link port signal definitions represent all differential signals with a V_{OD} = 0 V level and use signal naming without N (negative) and P (positive) suffixes (see Figure 17).

Table 30. Link Port LVDS Transmit Electrical Characteristics

Parameter	Description	Test Conditions	Min	Max	Unit
V_{OH}	Output Voltage High, V_{O_P} or V_{O_N}	R_L = 100 Ω		1.85	V
V_{OL}	Output Voltage Low, V_{O_P} or V_{O_N}	R_L = 100 Ω	0.92		V
$\|V_{OD}\|$	Output Differential Voltage	R_L = 100 Ω	300	650	mV
I_{OS}	Short-Circuit Output Current	V_{O_P} or V_{O_N} = 0 V		+5/– 55	mA
		V_{OD} = 0 V		±10	mA
V_{OCM}	Common-Mode Output Voltage		1.20	1.50	V

Table 31. Link Port LVDS Receive Electrical Characteristics

Parameter	Description	Test Conditions	Min	Max	Unit
$\|V_{ID}\|$	Differential Input Voltage	$t_{LDIS}/t_{LDIH} \geq 0.20$ ns	250	850	mV
		$t_{LDIS}/t_{LDIH} \geq 0.25$ ns	217	850	mV
		$t_{LDIS}/t_{LDIH} \geq 0.30$ ns	206	850	mV
		$t_{LDIS}/t_{LDIH} \geq 0.35$ ns	195	850	mV
V_{ICM}	Common-Mode Input Voltage		0.6	1.57	V

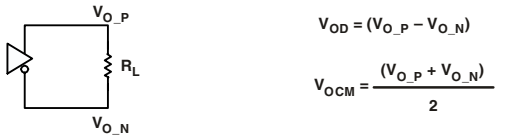


Figure 16. Link Ports—Transmit Electrical Characteristics

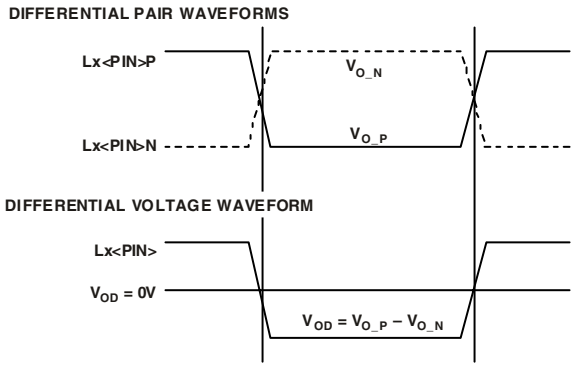


Figure 17. Link Ports—Signals Definition

Link Port—Data Out Timing

Table 32 with Figure 18, Figure 19, Figure 20, Figure 21, Figure 22, and Figure 23 provide the data out timing for the LVDS link ports.

Table 32. Link Port—Data Out Timing

Parameter	Description	Min	Max	Unit
Outputs				
t_{REO}	Rising Edge (Figure 19)		350	ps
t_{FEO}	Falling Edge (Figure 19)		350	ps
t_{LCLKOP}	LxCLKOUT Period (Figure 18)	Greater of 2.0 or $0.9 \times LCR \times t_{CCLK}$ [1, 2, 3]	Smaller of 12.5 or $1.1 \times LCR \times t_{CCLK}$ [1, 2, 3]	ns
t_{LCLKOH}	LxCLKOUT High (Figure 18)	$0.4 \times t_{LCLKOP}$ [1]	$0.6 \times t_{LCLKOP}$ [1]	ns
t_{LCLKOL}	LxCLKOUT Low (Figure 18)	$0.4 \times t_{LCLKOP}$ [1]	$0.6 \times t_{LCLKOP}$ [1]	ns
t_{COJT}	LxCLKOUT Jitter (Figure 18)		±150 [4, 5, 6]	ps
			±250 [7]	ps
t_{LDOS}	LxDATO Output Setup (Figure 20)	$0.25 \times LCR \times t_{CCLK} - 0.10 \times t_{CCLK}$ [1, 4, 8]		ns
		$0.25 \times LCR \times t_{CCLK} - 0.15 \times t_{CCLK}$ [1, 5, 6, 8]		ns
		$0.25 \times LCR \times t_{CCLK} - 0.30 \times t_{CCLK}$ [1, 7, 8]		ns
t_{LDOH}	LxDATO Output Hold (Figure 20)	$0.25 \times LCR \times t_{CCLK} - 0.10 \times t_{CCLK}$ [1, 4, 8]		ns
		$0.25 \times LCR \times t_{CCLK} - 0.15 \times t_{CCLK}$ [1, 5, 6, 8]		ns
		$0.25 \times LCR \times t_{CCLK} - 0.30 \times t_{CCLK}$ [1, 7, 8]		ns
t_{LACKID}	Delay from LxACKI rising edge to first transmission clock edge (Figure 21)		$16 \times LCR \times t_{CCLK}$ [1, 2]	ns
t_{BCMPOV}	$\overline{\text{LxBCMPO}}$ Valid (Figure 21)		$2 \times LCR \times t_{CCLK}$ [1, 2]	ns
t_{BCMPOH}	$\overline{\text{LxBCMPO}}$ Hold (Figure 22)	$3 \times TSW - 0.5$ [1, 9]		ns
Inputs				
t_{LACKIS}	LxACKI low setup to guarantee that the transmitter stops transmitting (Figure 22)			
	LxACKI high setup to guarantee that the transmitter continues its transmission without any interruption (Figure 23)	$16 \times LCR \times t_{CCLK}$ [1, 2]		ns
t_{LACKIH}	LxACKI High Hold Time (Figure 23)	0.51		ns

[1] Timing is relative to the 0 differential voltage (V_{OD} = 0).
[2] LCR (link port clock ratio) = 1, 1.5, 2, or 4. t_{CCLK} is the core period.
[3] For the cases of t_{LCLKOP} = 2.0 ns and t_{LCLKOP} = 12.5 ns, the effect of t_{COJT} specification on output period must be considered.
[4] LCR= 1.
[5] LCR= 1.5.
[6] LCR= 2.
[7] LCR= 4.
[8] The t_{LDOS} and t_{LDOH} values include LCLKOUT jitter.
[9] TSW is a short-word transmission period. For a 4-bit link, it is $2 \times LCR \times t_{CCLK}$. For a 1-bit link, it is $8 \times LCR \times t_{CCLK}$ ns.

Link Port—Data In Timing

Table 33 with Figure 24 and Figure 25 provide the data in timing for the LVDS link ports.

Table 33. Link Port—Data In Timing

Parameter	Description	Min	Max	Unit
Inputs				
t_{LCLKIP}	LxCLKIN Period (Figure 25)	Greater of 1.8 or $0.9 \times t_{CCLK}$[1]	12.5	ns
t_{LDIS}	LxDATI Input Setup (Figure 25)	0.20[1, 2]		ns
		0.25[1, 3]		ns
		0.30[1, 4]		ns
		0.35[1, 5]		ns
t_{LDIH}	LxDATI Input Hold (Figure 25)	0.20[1, 2]		ns
		0.25[1, 3]		ns
		0.30[1, 4]		ns
		0.35[1, 5]		ns
t_{BCMPIS}	$\overline{\text{LxBCMPI}}$ Setup (Figure 24)	$2 \times t_{LCLKIP}$[1]		ns
t_{BCMPIH}	$\overline{\text{LxBCMPI}}$ Hold (Figure 24)	$2 \times t_{LCLKIP}$[1]		ns

[1] Timing is relative to the 0 differential voltage (V_{OD} = 0).
[2] $|V_{ID}|$ = 250 mV
[3] $|V_{ID}|$ = 217 mV
[4] $|V_{ID}|$ = 206 mV
[5] $|V_{ID}|$ = 195 mV

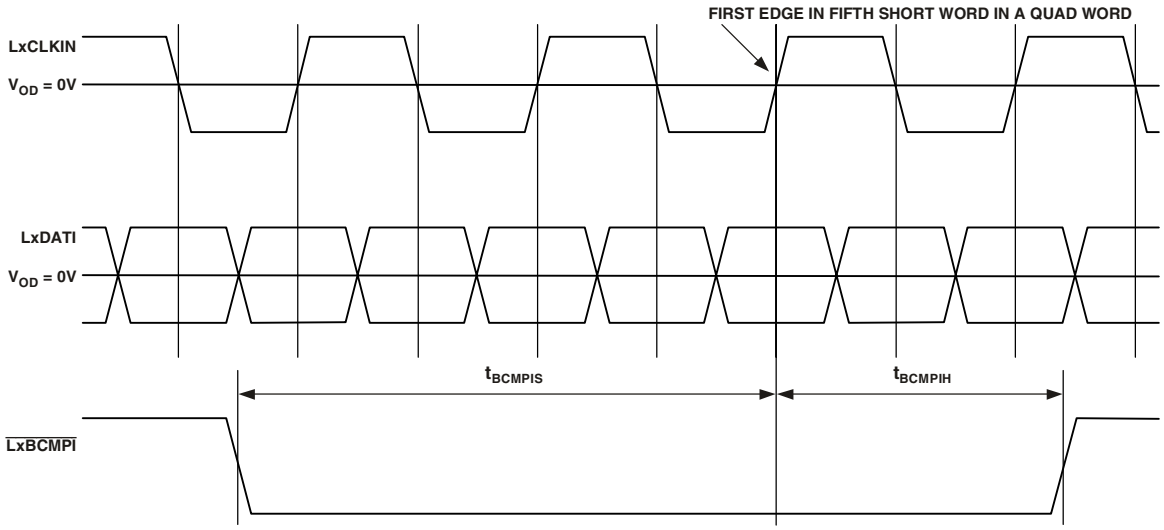


Figure 24. Link Ports—Last Received Quad Word

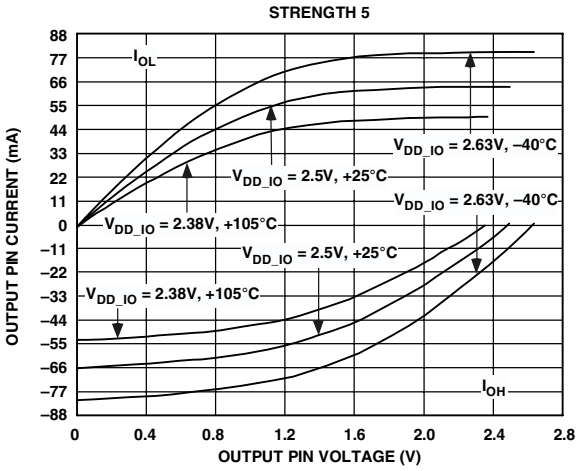


Figure 31. Typical Drive Currents at Strength 5

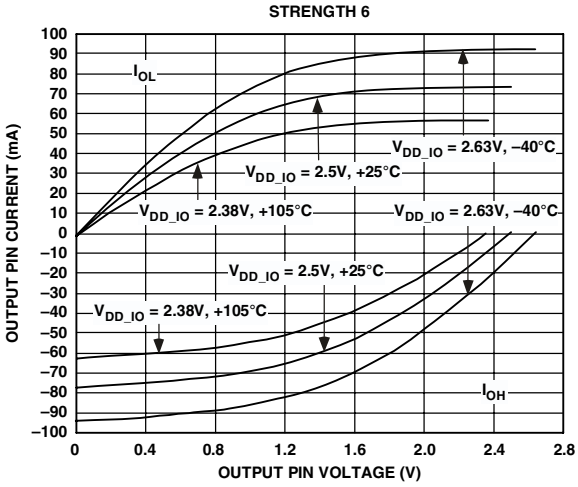


Figure 32. Typical Drive Currents at Strength 6

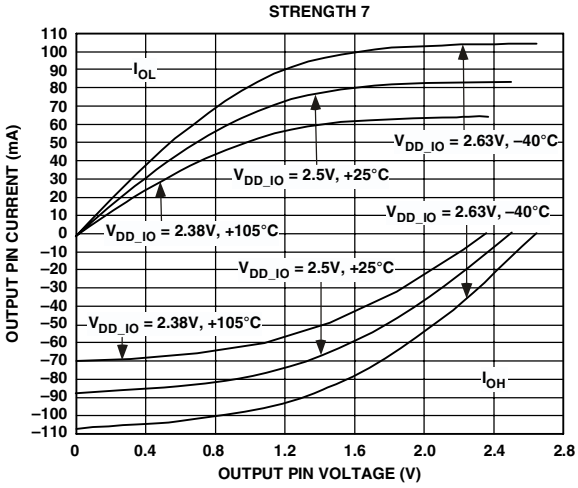


Figure 33. Typical Drive Currents at Strength 7

TEST CONDITIONS

The ac signal specifications (timing parameters) appear in Table 29 on Page 28. These include output disable time, output enable time, and capacitive loading. The timing specifications for the DSP apply for the voltage reference levels in Figure 34.

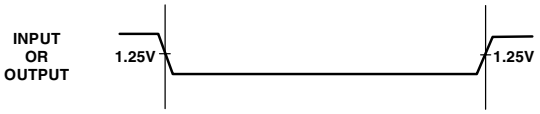


Figure 34. Voltage Reference Levels for AC Measurements (Except Output Enable/Disable)

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:

$$t_{DECAY} = (C_L \Delta V)/I_L$$

The output disable time t_{DIS} is the difference between $t_{MEASURED\_DIS}$ and t_{DECAY} as shown in Figure 35. 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. t_{DECAY} is calculated with test loads C_L and I_L, and with ΔV equal to 0.4 V.

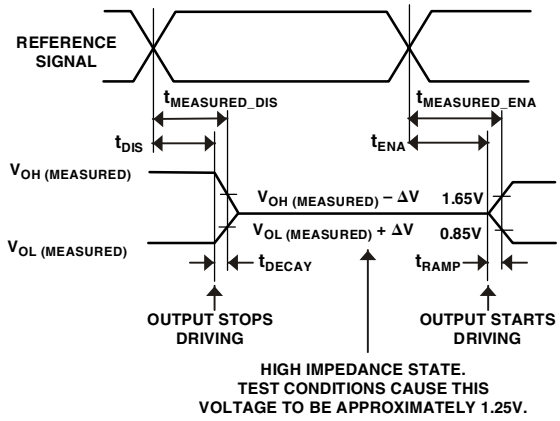


Figure 35. Output Enable/Disable

576-BALL BGA_ED PIN CONFIGURATIONS

Figure 46 shows a summary of pin configurations for the 576-ball BGA_ED package and Table 35 lists the signal-to-ball assignments.

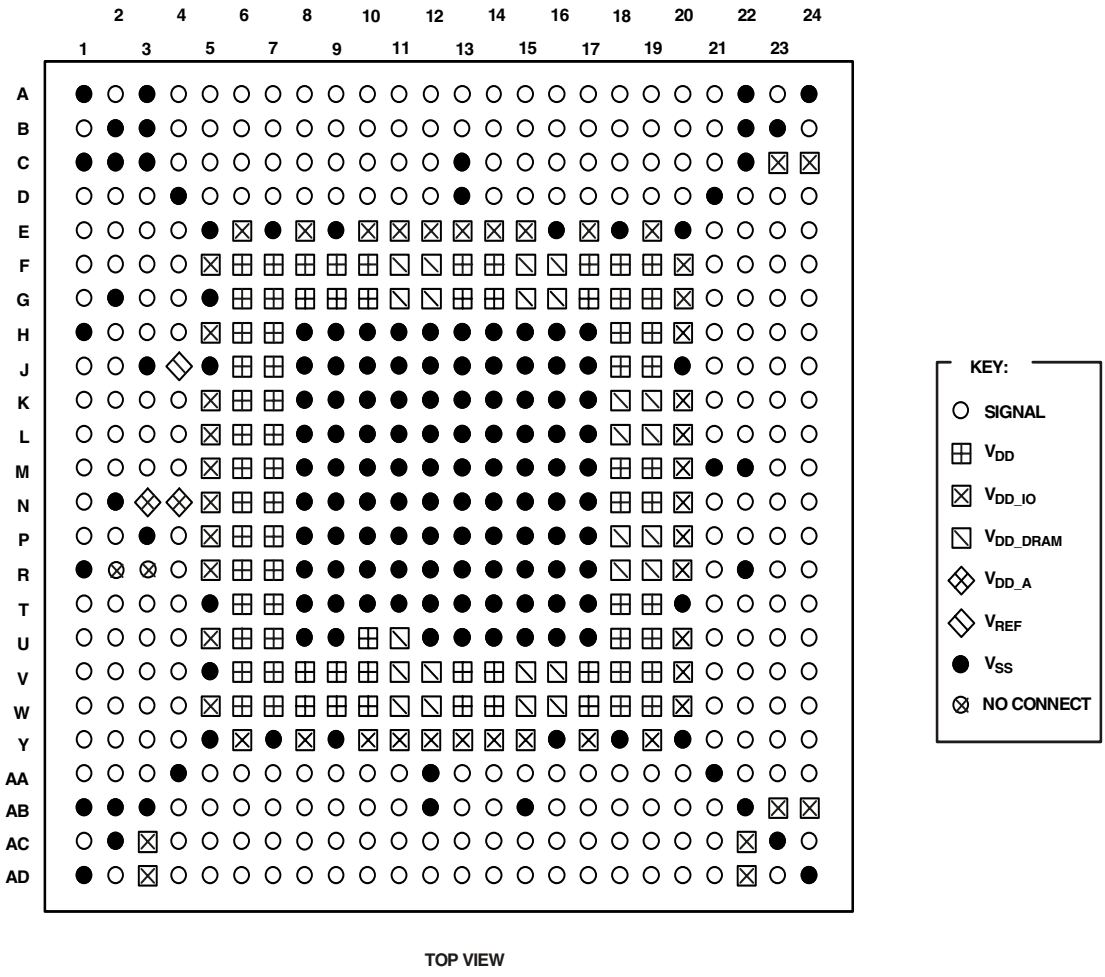


Figure 46. 576-Ball BGA_ED Pin Configurations[1] (Top View, Summary)

[1] For a more detailed pin summary diagram, see the *EE-179: ADSP-TS201S System Design Guidelines* on the Analog Devices website (www.analog.com).

Table 35. 576-Ball (25 mm × 25 mm) BGA_ED Ball Assignments (Continued)

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
U1	$\overline{\text{MSSD0}}$	V1	$\overline{\text{MSSD2}}$	W1	CONTROLIMP0	Y1	$\overline{\text{EMU}}$
U2	$\overline{\text{RST\_OUT}}$	V2	DS2	W2	ENEDREG	Y2	TCK
U3	ID2	V3	$\overline{\text{POR\_IN}}$	W3	TDI	Y3	TMR0E
U4	DS1	V4	CONTROLIMP1	W4	TDO	Y4	FLAG3
U5	V_{DD_IO}	V5	V_{SS}	W5	V_{DD_IO}	Y5	V_{SS}
U6	V_{DD}	V6	V_{DD}	W6	V_{DD}	Y6	V_{DD_IO}
U7	V_{DD}	V7	V_{DD}	W7	V_{DD}	Y7	V_{SS}
U8	V_{SS}	V8	V_{DD}	W8	V_{DD}	Y8	V_{DD_IO}
U9	V_{SS}	V9	V_{DD}	W9	V_{DD}	Y9	V_{SS}
U10	V_{DD}	V10	V_{DD}	W10	V_{DD}	Y10	V_{DD_IO}
U11	V_{DD_DRAM}	V11	V_{DD_DRAM}	W11	V_{DD_DRAM}	Y11	V_{DD_IO}
U12	V_{SS}	V12	V_{DD_DRAM}	W12	V_{DD_DRAM}	Y12	V_{DD_IO}
U13	V_{SS}	V13	V_{DD}	W13	V_{DD}	Y13	V_{DD_IO}
U14	V_{SS}	V14	V_{DD}	W14	V_{DD}	Y14	V_{DD_IO}
U15	V_{SS}	V15	V_{DD_DRAM}	W15	V_{DD_DRAM}	Y15	V_{DD_IO}
U16	V_{SS}	V16	V_{DD_DRAM}	W16	V_{DD_DRAM}	Y16	V_{SS}
U17	V_{SS}	V17	V_{DD}	W17	V_{DD}	Y17	V_{DD_IO}
U18	V_{DD}	V18	V_{DD}	W18	V_{DD}	Y18	V_{SS}
U19	V_{DD}	V19	V_{DD}	W19	V_{DD}	Y19	V_{DD_IO}
U20	V_{DD_IO}	V20	V_{DD_IO}	W20	V_{DD_IO}	Y20	V_{SS}
U21	L1CLKINN	V21	L1DATI3_N	W21	L1CLKON	Y21	L1DATO1_N
U22	L1CLKINP	V22	L1DATI3_P	W22	L1CLKOP	Y22	L1DATO1_P
U23	L1DATI1_N	V23	L1DATI2_N	W23	L1DATO3_N	Y23	L1DATO2_N
U24	L1DATI1_P	V24	L1DATI2_P	W24	L1DATO3_P	Y24	L1DATO2_P
AA1	FLAG2	AB1	V_{SS}	AC1	FLAG0	AD1	V_{SS}
AA2	FLAG1	AB2	V_{SS}	AC2	V_{SS}	AD2	ID1
AA3	$\overline{\text{IRQ3}}$	AB3	V_{SS}	AC3	V_{DD_IO}	AD3	V_{DD_IO}
AA4	V_{SS}	AB4	NC	AC4	TMS	AD4	$\overline{\text{TRST}}$
AA5	$\overline{\text{IRQ0}}$	AB5	$\overline{\text{IRQ2}}$	AC5	$\overline{\text{IOWR}}$	AD5	$\overline{\text{IORD}}$
AA6	$\overline{\text{IOEN}}$	AB6	$\overline{\text{IRQ1}}$	AC6	$\overline{\text{DMAR2}}$	AD6	$\overline{\text{DMAR3}}$
AA7	$\overline{\text{DMAR0}}$	AB7	$\overline{\text{DMAR1}}$	AC7	$\overline{\text{CPA}}$	AD7	$\overline{\text{DPA}}$
AA8	$\overline{\text{HBR}}$	AB8	$\overline{\text{HBG}}$	AC8	$\overline{\text{BOFF}}$	AD8	$\overline{\text{BUSLOCK}}$
AA9	$\overline{\text{L3BCMPO}}$	AB9	L3ACKI	AC9	L3DATO0_N	AD9	L3DATO0_P
AA10	L3DATO1_N	AB10	L3DATO1_P	AC10	L3CLKON	AD10	L3CLKOP
AA11	L3DATO3_N	AB11	L3DATO3_P	AC11	L3DATO2_N	AD11	L3DATO2_P
AA12	V_{SS}	AB12	V_{SS}	AC12	L3DATI3_N	AD12	L3DATI3_P
AA13	L3DATI2_N	AB13	L3DATI2_P	AC13	L3CLKINN	AD13	L3CLKINP
AA14	L3DATI1_N	AB14	L3DATI1_P	AC14	L3DATI0_N	AD14	L3DATI0_P
AA15	NC	AB15	V_{SS}	AC15	L3ACKO	AD15	$\overline{\text{L3BCMPI}}$
AA16	L2DATO0_N	AB16	L2DATO0_P	AC16	$\overline{\text{L2BCMPO}}$	AD16	L2ACKI
AA17	L2CLKON	AB17	L2CLKOP	AC17	L2DATO1_N	AD17	L2DATO1_P
AA18	L2DATO3_N	AB18	L2DATO3_P	AC18	L2DATO2_N	AD18	L2DATO2_P
AA19	L2CLKINN	AB19	L2CLKINP	AC19	L2DATI3_N	AD19	L2DATI3_P
AA20	L2DATI1_N	AB20	L2DATI1_P	AC20	L2DATI2_N	AD20	L2DATI2_P
AA21	V_{SS}	AB21	L2ACKO	AC21	L2DATI0_N	AD21	L2DATI0_P
AA22	$\overline{\text{L1BCMPO}}$	AB22	V_{SS}	AC22	V_{DD_IO}	AD22	V_{DD_IO}
AA23	L1DATO0_N	AB23	V_{DD_IO}	AC23	V_{SS}	AD23	$\overline{\text{L2BCMPI}}$
AA24	L1DATO0_P	AB24	V_{DD_IO}	AC24	L1ACKI	AD24	V_{SS}

[1] On revision 1.x silicon, the R2 and R3 balls are NC. On revision 0.x silicon, the R2 ball is SCLK, and the R3 ball is SCLK_V_{REF}. For more information on SCLK and SCLK_V_{REF} on revision 0.x silicon, see the *EE-179: ADSP-TS20x TigerSHARC System Design Guidelines* on the Analog Devices website (*www.analog.com*).

ORDERING GUIDE

Model	Temperature Range[1]	Instruction Rate[2]	On-Chip DRAM	Operating Voltage	Package Option	Package Description
ADSP-TS201SABP-060	–40°C to +85°C	600 MHz	24M bit	1.20 V_{DD}, 2.5 V_{DD_IO}, 1.6 V_{DD_DRAM}	BP-576	576-Ball BGA_ED
ADSP-TS201SABP-050	–40°C to +85°C	500 MHz	24M bit	1.05 V_{DD}, 2.5 V_{DD_IO}, 1.5 V_{DD_DRAM}	BP-576	576-Ball BGA_ED
ADSP-TS201SYBP-050	–40°C to +105°C	500 MHz	24M bit	1.05 V_{DD}, 2.5 V_{DD_IO}, 1.5 V_{DD_DRAM}	BP-576	576-Ball BGA_ED
ADSP-TS201SABPZ060[3]	–40°C to +85°C	600 MHz	24M bit	1.20 V_{DD}, 2.5 V_{DD_IO}, 1.6 V_{DD_DRAM}	BP-576	576-Ball BGA_ED
ADSP-TS201SABPZ050[3]	–40°C to +85°C	500 MHz	24M bit	1.05 V_{DD}, 2.5 V_{DD_IO}, 1.5 V_{DD_DRAM}	BP-576	576-Ball BGA_ED
ADSP-TS201SYBPZ050[3]	–40°C to +105°C	500 MHz	24M bit	1.05 V_{DD}, 2.5 V_{DD_IO}, 1.5 V_{DD_DRAM}	BP-576	576-Ball BGA_ED

[1] Represents case temperature.
[2] The instruction rate is the same as the internal processor core clock (CCLK) rate.
[3] Z = Pb-free part.

D04324-0-11/06(C)