



Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!



Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832

Email & Skype: info@chipsmall.com Web: www.chipsmall.com

Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China



KEY FEATURES

- Up to 600 MHz, 1.67 ns instruction cycle rate
- 24M bits of internal—on-chip—DRAM memory
- 25 mm × 25 mm (576-ball) thermally enhanced ball grid array package
- Dual-computation blocks—each containing an ALU, a multiplier, a shifter, a register file, and a communications logic unit (CLU)
- Dual-integer ALUs, providing data addressing and pointer manipulation
- Dual-integer ALUs, providing data addressing and pointer manipulation
- Integrated I/O includes 14-channel DMA controller, external port, four link ports, SDRAM controller, programmable flag pins, two timers, and timer expired pin for system integration
- 1149.1 IEEE-compliant JTAG test access port for on-chip emulation
- Single-precision IEEE 32-bit and extended-precision 40-bit floating-point data formats and 8-, 16-, 32-, and 64-bit fixed-point data formats

KEY BENEFITS

- Provides high performance static superscalar DSP operations, optimized for telecommunications infrastructure and other large, demanding multiprocessor DSP applications
- Performs exceptionally well on DSP algorithm and I/O benchmarks (see benchmarks in [Table 1](#))
- Supports low overhead DMA transfers between internal memory, external memory, memory-mapped peripherals, link ports, host processors, and other (multiprocessor) DSPs
- Eases DSP programming through extremely flexible instruction set and high-level-language-friendly DSP architecture
- Enables scalable multiprocessing systems with low communications overhead
- Provides on-chip arbitration for glueless multiprocessing

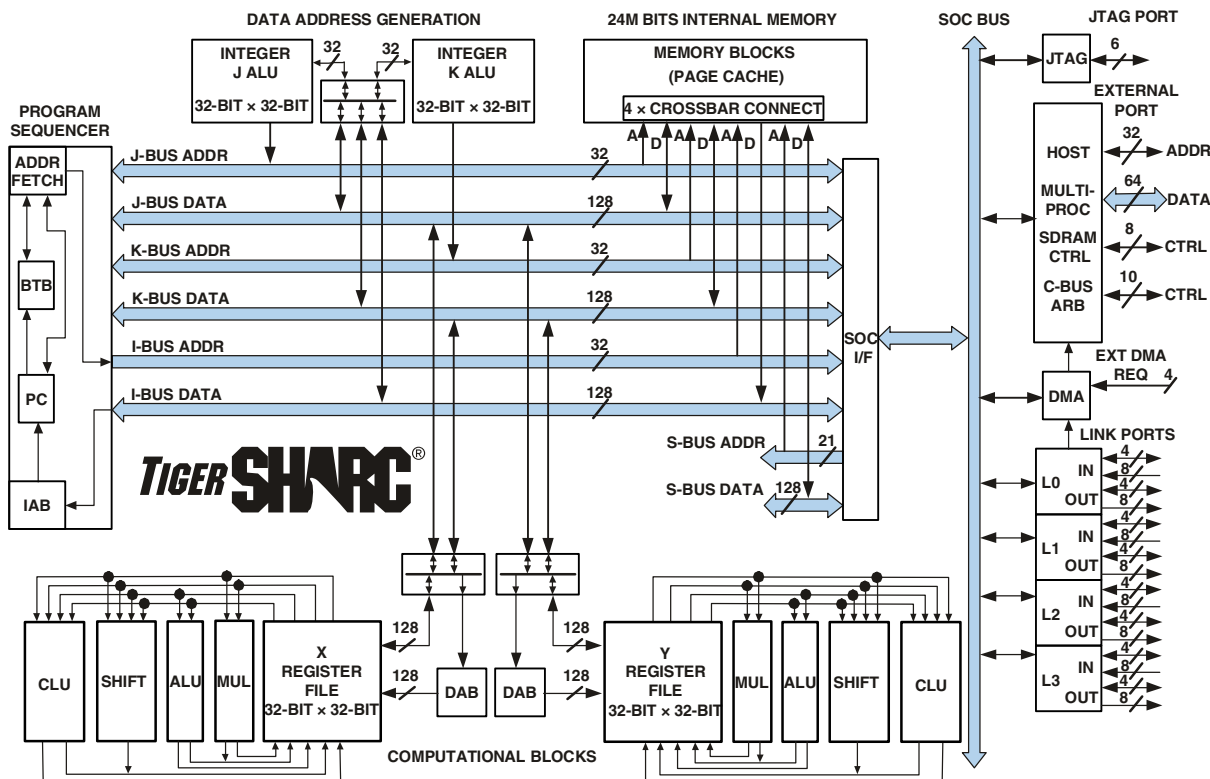


Figure 1. Functional Block Diagram

TigerSHARC and the TigerSHARC logo are registered trademarks of Analog Devices, Inc.

Rev. C

Information furnished by Analog Devices is believed to be accurate and reliable. However, no responsibility is assumed by Analog Devices for its use, nor for any infringements of patents or other rights of third parties that may result from its use. Specifications subject to change without notice. No license is granted by implication or otherwise under any patent or patent rights of Analog Devices. Trademarks and registered trademarks are the property of their respective owners.

TABLE OF CONTENTS

General Description	3
Dual Compute Blocks	4
Data Alignment Buffer (DAB)	4
Dual Integer ALU (IALU)	4
Program Sequencer	5
Interrupt Controller	5
Flexible Instruction Set	5
DSP Memory	5
External Port (Off-Chip Memory/Peripherals Interface)	6
Host Interface	7
Multiprocessor Interface	7
SDRAM Controller	7
EPROM Interface	7
DMA Controller	7
Link Ports (LVDS)	9
Timer and General-Purpose I/O	9
Reset and Booting	9
Clock Domains	9
Power Domains	10
Filtering Reference Voltage and Clocks	10
Development Tools	10
Evaluation Kit	11
Designing an Emulator-Compatible DSP Board (Target)	11
Additional Information	11
Pin Function Descriptions	12
Strap Pin Function Descriptions	20
ADSP-TS201S—Specifications	21
Operating Conditions	21
Electrical Characteristics	22
Package Information	23
Absolute Maximum Ratings	23
ESD Sensitivity	23
Timing Specifications	24
General AC Timing	24
Link Port Low Voltage, Differential-Signal (LVDS) Electrical Characteristics, and Timing	30
Link Port—Data Out Timing	31
Link Port—Data In Timing	34
Output Drive Currents	36

Test Conditions	37
Output Disable Time	37
Output Enable Time	38
Capacitive Loading	38
Environmental Conditions	40
Thermal Characteristics	40
576-Ball BGA_ED Pin Configurations	41
Outline Dimensions	45
Surface Mount Design	45
Ordering Guide	46

REVISION HISTORY

12/06—Rev. B to Rev. C

Applied Corrections to:

Figure 7, SCLK_VREF Filtering Scheme	10
Operating Conditions	21
Added On-Chip DRAM Refresh	27
Ordering Guide	46

GENERAL DESCRIPTION

The ADSP-TS201S TigerSHARC processor is an ultrahigh performance, static superscalar processor optimized for large signal processing tasks and communications infrastructure. The DSP combines very wide memory widths with dual computation blocks—supporting floating-point (IEEE 32-bit and extended precision 40-bit) and fixed-point (8-, 16-, 32-, and 64-bit) processing—to set a new standard of performance for digital signal processors. The TigerSHARC static superscalar architecture lets the DSP execute up to four instructions each cycle, performing 24 fixed-point (16-bit) operations or six floating-point operations.

Four independent 128-bit wide internal data buses, each connecting to the six 4M bit memory banks, enable quad-word data, instruction, and I/O access and provide 33.6G bytes per second of internal memory bandwidth. Operating at 600 MHz, the ADSP-TS201S processor’s core has a 1.67 ns instruction cycle time. Using its single-instruction, multiple-data (SIMD) features, the ADSP-TS201S processor can perform 4.8 billion, 40-bit MACS or 1.2 billion, 80-bit MACS per second. [Table 1](#) shows the DSP’s performance benchmarks.

Table 1. General-Purpose Algorithm Benchmarks at 600 MHz

Benchmark	Speed	Clock Cycles
32-bit algorithm, 1.2 billion MACS/s peak performance		
1K point complex FFT ¹ (Radix 2)	15.7 μs	9419
64K point complex FFT ¹ (Radix 2)	2.33 ms	1397544
FIR filter (per real tap)	0.83 ns	0.5
[8 × 8][8 × 8] matrix multiply (complex, floating-point)	2.3 μs	1399
16-bit algorithm, 4.8 billion MACS/s peak performance		
256 point complex FFT ¹ (Radix 2)	0.975 μs	585
I/O DMA transfer rate		
External port	1G bytes/s	n/a
Link ports (each)	1G bytes/s	n/a

¹ Cache preloaded

The ADSP-TS201S processor is code compatible with the other TigerSHARC processors.

The Functional Block Diagram on [Page 1](#) shows the ADSP-TS201S processor’s architectural blocks. These blocks include:

- Dual compute blocks, each consisting of an ALU, multiplier, 64-bit shifter, 128-bit CLU, and 32-word register file and associated data alignment buffers (DABs)
- Dual integer ALUs (IALUs), each with its own 31-word register file for data addressing and a status register
- A program sequencer with instruction alignment buffer (IAB) and branch target buffer (BTB)

- An interrupt controller that supports hardware and software interrupts, supports level- or edge-triggers, and supports prioritized, nested interrupts
- Four 128-bit internal data buses, each connecting to the six 4M bit memory banks
- On-chip DRAM (24M bit)
- An external port that provides the interface to host processors, multiprocessing space (DSPs), off-chip memory-mapped peripherals, and external SRAM and SDRAM
- A 14-channel DMA controller
- Four full-duplex LVDS link ports
- Two 64-bit interval timers and timer expired pin
- An 1149.1 IEEE-compliant JTAG test access port for on-chip emulation

[Figure 2 on Page 3](#) shows a typical single-processor system with external SRAM and SDRAM. [Figure 4 on Page 8](#) shows a typical multiprocessor system.

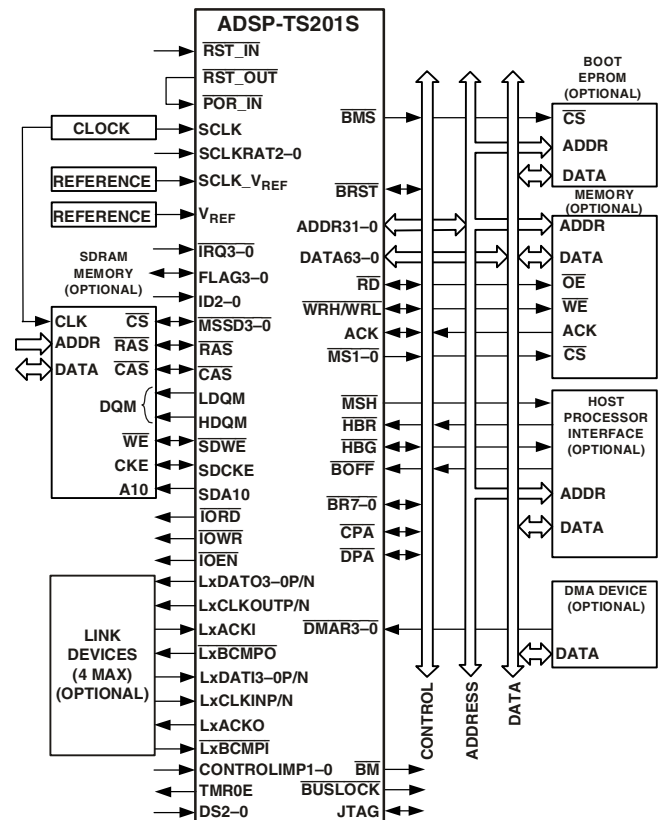


Figure 2. ADSP-TS201S Single-Processor System with External SDRAM

ADSP-TS201S

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 multiplexed 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.

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-TS201S processor's program sequencer supports the following:

- 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
- A 10-cycle instruction pipeline—four-cycle fetch pipe and six-cycle execution pipe—computation results available two cycles after operands are available
- 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
- Management of program structures and program flow determined 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 occur with zero overhead cycles, overcoming the five-to-nine stage branch penalty
- Compact code without the requirement to align code in memory; the IAB handles alignment

Interrupt Controller

The DSP supports nested and nonnested 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 $\overline{\text{IRQ}}3\text{--}0$ 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:

- CLU 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
- Eliminates toggling DSP hardware modes because modes are supported as options (for example, rounding, saturation, and others) within instructions
- Branch prediction encoded in instruction; enables zero-overhead loops
- Parallelism encoded in instruction line
- Conditional execution optional for all instructions
- User-defined partitioning between program and data memory

DSP MEMORY

The DSP's internal and external memory is organized into a unified memory map, which defines the location (address) of all elements in the system, as shown in [Figure 3](#).

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.

The ADSP-TS201S processor internal memory has 24M bits of on-chip DRAM memory, divided into six blocks of 4M bits (128K words \times 32 bits). Each block—M0, M2, M4, M6, M8, and M10—can store program instructions, 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. Each memory segment contains a 128K bit cache to enable single cycle access to internal DRAM.

The six internal memory blocks connect to the four 128-bit wide internal buses through a crossbar connection, enabling the DSP to perform four memory transfers in the same cycle. The DSP's internal bus architecture provides a total memory bandwidth of

ADSP-TS201S

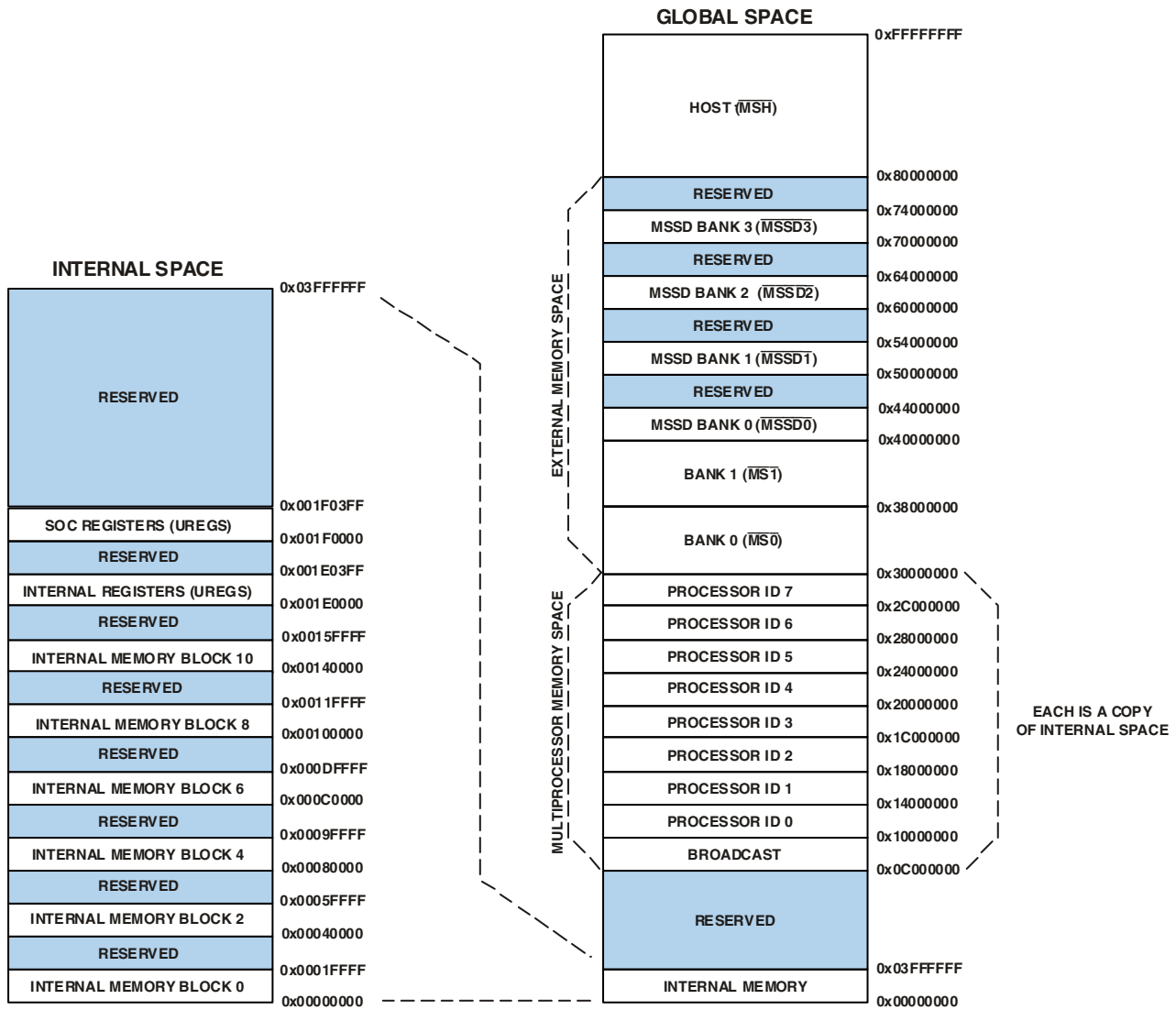


Figure 3. ADSP-TS201S Memory Map

33.6G bytes per second, enabling the core and I/O to access eight 32-bit data-words and four 32-bit instructions each cycle. The DSP's flexible memory structure enables:

- DSP core and I/O accesses to different memory blocks in the same cycle
- DSP core access to 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

EXTERNAL PORT (OFF-CHIP MEMORY/PERIPHERALS INTERFACE)

The ADSP-TS201S processor's external port provides the DSP's interface to off-chip memory and peripherals. The 4G word address space is included in the DSP's unified address space.

The separate on-chip buses—four 128-bit data buses and four 32-bit address buses—are multiplexed at the SOC interface and transferred to the external port over the SOC bus to create an external system bus transaction. The external system bus provides a single 64-bit data bus and a single 32-bit address bus. The external port supports data transfer rates of 1G byte per second over the external bus.

The external bus can be configured for 32-bit or 64-bit, little-endian operations. When the system bus is configured for 64-bit operations, the lower 32 bits of the external data bus connect to even addresses, and the upper 32 bits connect to odd addresses.

The external port supports pipelined, slow, and SDRAM protocols. Addressing of external memory devices and memory-mapped peripherals is facilitated by on-chip decoding of high order address lines to generate memory bank select signals.

The ADSP-TS201S processor provides programmable memory, pipeline depth, and idle cycle for synchronous accesses; and external acknowledge controls to support interfacing to pipelined or slow devices, host processors, and other memory-mapped peripherals with variable access, hold, and disable time requirements.

Host Interface

The ADSP-TS201S processor provides an easy and configurable interface between its external bus and host processors through the external port (see [Figure 4](#)). To accommodate a variety of host processors, the host interface supports pipelined or slow protocols for ADSP-TS201S processor access of the host as slave or pipelined for host access of the ADSP-TS201S processor 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-TS201S processor, 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-TS201S processor offers powerful features tailored to multiprocessing DSP systems through the external port and link ports (see [Figure 4](#)). This multiprocessing capability provides the 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-TS201S 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-TS201S 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 4G bytes per second. The cluster bus provides 1G byte per second throughput—with a total of 4.8G bytes per second interprocessor bandwidth (limited by SOC bandwidth).

SDRAM Controller

The SDRAM controller controls the ADSP-TS201S processor's transfers of data to and from external synchronous DRAM (SDRAM) at a throughput of 32 bits 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, 256M bit, and 512M bit. The DSP supports directly a maximum of four banks of 64M words \times 32 bits of SDRAM. The SDRAM interface is mapped in external memory in each DSP's unified memory map.

EPROM Interface

The ADSP-TS201S processor can be configured to boot from an 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-TS201S 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, external memory, and memory-mapped peripherals; the internal memory of other DSPs on a common bus, a host processor, or link port I/O; between external memory and external peripherals or link port I/O; and between an external bus master and internal memory or link port I/O. 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

ADSP-TS201S

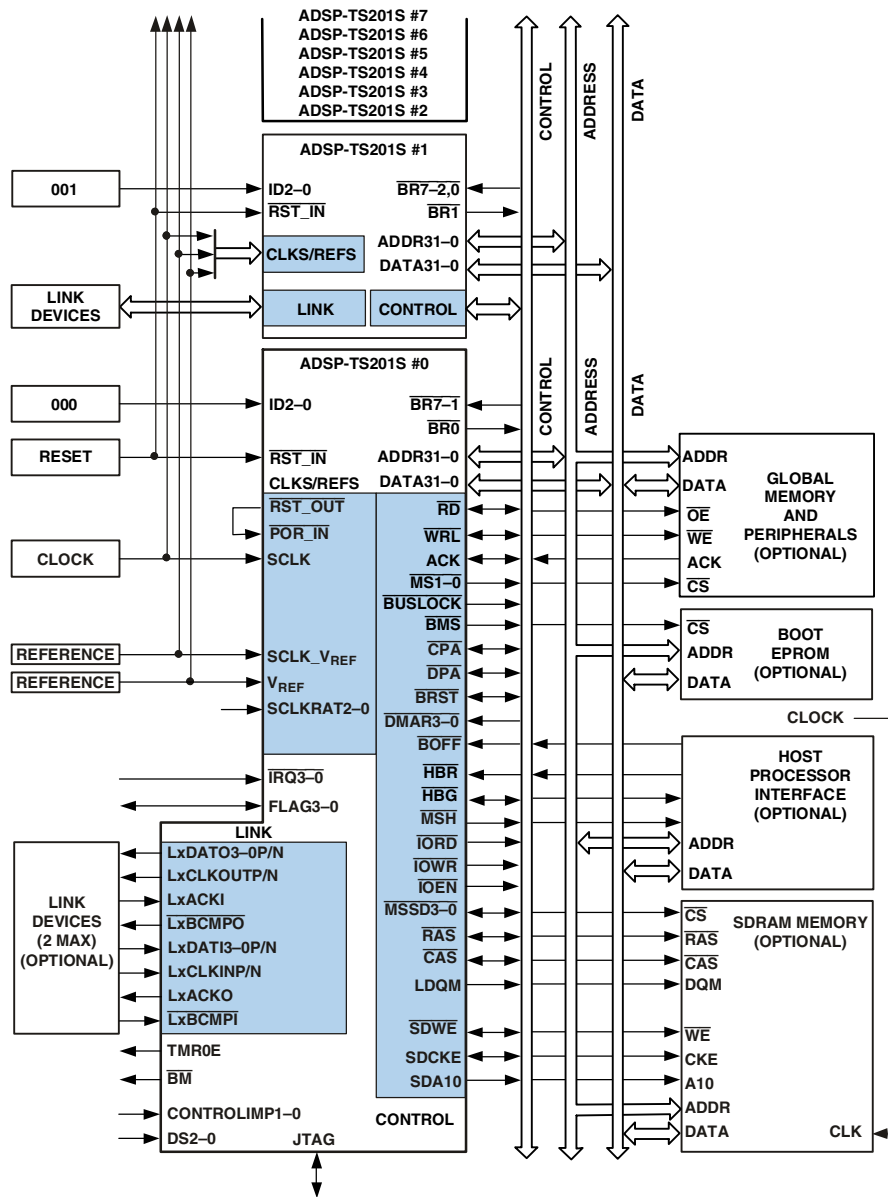


Figure 4. ADSP-TS201S Shared Memory Multiprocessing System

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.

The DMA controller provides these additional features:

- 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 an I/O device to external SDRAM memory.

During a transaction, the DSP relinquishes the external data bus; outputs addresses and memory selects (MSSD3-0); outputs the IORD, IOWR, IOEN, and RD/WR strobes; and responds to ACK.

- DMA chaining. 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.
- 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.

LINK PORTS (LVDS)

The DSP's four full-duplex link ports each provide additional four-bit receive and four-bit transmit I/O capability, using low voltage, differential-signal (LVDS) technology. With the ability to operate at a double data rate—latching data on both the rising and falling edges of the clock—running at up to 500 MHz, each link port can support up to 500M bytes per second per direction, for a combined maximum throughput of 4G 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 triple-buffered quad-word input and double-buffered quad-word 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 direction has three signals that control its operation. For the transmitter, LxCLKOUT is the output transmit clock, LxACKI is the handshake input to control the data flow, and the $\overline{\text{LxBMPPO}}$ output indicates that the block transfer is complete. For the receiver, LxCLKIN is the input receive clock, LxACKO is the handshake output to control the data flow, and the $\overline{\text{LxBMPPI}}$ input indicates that the block transfer is complete. The LxDATO3–0 pins are the data output bus for the transmitter and the LxDATI3–0 pins are the input data bus for the receiver.

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.

TIMER AND GENERAL-PURPOSE I/O

The ADSP-TS201S processor has a timer pin (TMROE) that generates output when a programmed timer counter has expired, and 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-TS201S processor has three levels of reset:

- Power-up reset – after power-up of the system (SCLK, all static inputs, and strap pins are stable), the $\overline{\text{RST_IN}}$ pin must be asserted (low).
- Normal reset – for any chip reset following the power-up reset, the $\overline{\text{RST_IN}}$ pin must be asserted (low).
- DSP-core reset – when setting the SWRST bit in EMUCTL, the DSP core is reset, but not the external port or I/O.

For normal operations, tie the $\overline{\text{RST_OUT}}$ pin to the $\overline{\text{POR_IN}}$ pin.

After reset, the ADSP-TS201S processor has four boot options for beginning operation:

- Boot from EPROM.
- Boot by an external master (host or another ADSP-TS201S processor).
- Boot by link port.
- No boot—start running from memory address selected with one of the $\overline{\text{IRQ3-0}}$ interrupt signals. See Table 2.

Using the “no boot” option, the ADSP-TS201S processor must start running from memory when one of the interrupts is asserted.

Table 2. No Boot, Run from Memory Addresses

Interrupt	Address
$\overline{\text{IRQ0}}$	0x3000 0000 (External Memory)
$\overline{\text{IRQ1}}$	0x3800 0000 (External Memory)
$\overline{\text{IRQ2}}$	0x8000 0000 (External Memory)
$\overline{\text{IRQ3}}$	0x0000 0000 (Internal Memory)

The ADSP-TS201S processor 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.

For more information on boot options, see the *EE-200: ADSP-TS20x TigerSHARC Processor Boot Loader Kernels Operation* on the Analog Devices website (www.analog.com).

CLOCK DOMAINS

The DSP uses calculated ratios of the SCLK clock to operate, as shown in Figure 5. The instruction execution rate is equal to CCLK. A PLL from SCLK generates CCLK which is phase-locked. The SCLKRATx pins define the clock multiplication of SCLK to CCLK (see Table 4 on Page 12). The link port clock is generated from CCLK via a software programmable divisor, and the SOC bus operates at 1/2 CCLK. Memory transfers to external and link port buffers operate at the SOCCLK rate. SCLK also 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 the SCLK frequency. The maximum SCLK frequency is one quarter the internal DSP clock (CCLK) frequency.

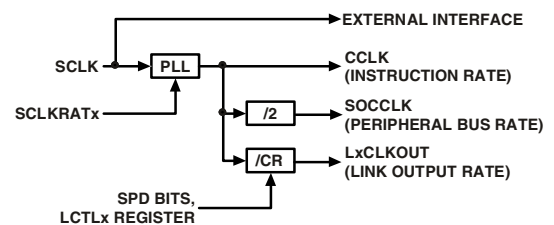


Figure 5. Clock Domains

ADSP-TS201S

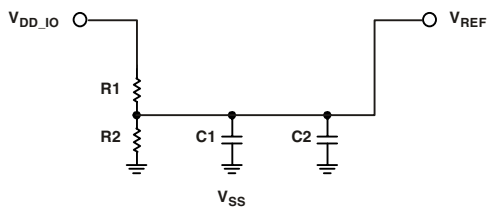
POWER DOMAINS

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.

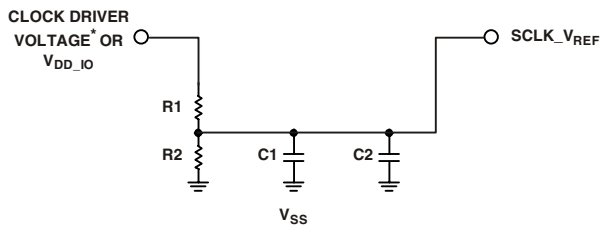
FILTERING REFERENCE VOLTAGE AND CLOCKS

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: 2k Ω SERIES RESISTOR ($\pm 1\%$)
R2: 2.55k Ω SERIES RESISTOR ($\pm 1\%$)
C1: 1 μ F CAPACITOR (SMD)
C2: 1nF CAPACITOR (HF SMD) PLACED CLOSE TO DSP'S PINS

Figure 6. V_{REF} Filtering Scheme



R1: 2k Ω SERIES RESISTOR ($\pm 1\%$)
R2: 2.55k Ω SERIES RESISTOR ($\pm 1\%$)
C1: 1 μ F 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.

[†] CROSSCORE is a registered trademark of Analog Devices, Inc.

[‡] VisualDSP++ is a registered trademark of Analog Devices, Inc.

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 these 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,

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, pre-emptive, 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.

ADSP-TS201S

PIN FUNCTION DESCRIPTIONS

While most of the ADSP-TS201S processor's input pins are normally synchronous—tied to a specific clock—a few are asynchronous. For these asynchronous signals, an on-chip synchronization circuit prevents metastability problems. Use the ac specification for asynchronous signals when the system design requires predictable, cycle-by-cycle behavior for these signals.

The output pins can be three-stated during normal operation. The DSP three-states all output pins during reset, allowing these pins to get to their internal pull-up or pull-down state. Some pins have an internal pull-up or pull-down resistor ($\pm 30\%$ tolerance) that maintains a known value during transitions between different drivers.

Table 3. Pin Definitions—Clocks and Reset

Signal	Type	Term	Description
SCLKRAT2-0	I (pd)	na	Core Clock Ratio. The DSP's core clock (CCLK) rate = $n \times \text{SCLK}$, where n is user-programmable using the SCLKRATx pins to the values shown in Table 4 . These pins may change only during reset; connect these pins to V_{DD_IO} or V_{SS} . All reset specifications in Table 25 , Table 26 , and Table 27 must be satisfied. The core clock rate (CCLK) is the instruction cycle rate.
SCLK	I	na	System Clock Input. The DSP's system input clock for cluster bus. The core clock rate is user-programmable using the SCLKRATx pins. For more information, see Clock Domains on Page 9.
$\overline{\text{RST_IN}}$	I/A	na	Reset. Sets the DSP to a known state and causes program to be in idle state. $\overline{\text{RST_IN}}$ must be asserted a specified time according to the type of reset operation. For details, see Reset and Booting on Page 9 , Table 25 on Page 26 , and Figure 13 on Page 26 .
$\overline{\text{RST_OUT}}$	O	na	Reset Output. Indicates that the DSP reset is complete. Connect to $\overline{\text{POR_IN}}$.
$\overline{\text{POR_IN}}$	I/A	na	Power-On Reset for internal DRAM. Connect to $\overline{\text{RST_OUT}}$.

I = input; **A** = asynchronous; **O** = output; **OD** = open-drain output; **T** = three-state; **P** = power supply; **G** = ground; **pd** = internal pull-down 5 k Ω ; **pu** = internal pull-up 5 k Ω ; **pd_0** = internal pull-down 5 k Ω on DSP ID = 0; **pu_0** = internal pull-up 5 k Ω on DSP ID = 0; **pu_od_0** = internal pull-up 500 Ω on DSP ID = 0; **pd_m** = internal pull-down 5 k Ω on DSP bus master; **pu_m** = internal pull-up 5 k Ω on DSP bus master; **pu_ad** = internal pull-up 40 k Ω . 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 k Ω to V_{SS} ; epu = external pull-up approximately 5 k Ω 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 4. SCLK Ratio

SCLKRAT2-0	Ratio
000 (default)	4
001	5
010	6
011	7
100	8
101	10
110	12
111	Reserved

Table 5. Pin Definitions—External Port Bus Controls

Signal	Type	Term	Description
ADDR31–0	I/O/T (pu_ad)	nc	Address Bus. The DSP issues addresses for accessing memory and peripherals on these pins. In a multiprocessor system, the bus master drives addresses for accessing internal memory or I/O processor registers of other ADSP-TS201S processors. The DSP inputs addresses when a host or another DSP accesses its internal memory or I/O processor registers.
DATA63–0	I/O/T (pu_ad)	nc	External Data Bus. The DSP drives and receives data and instructions on these pins. Pull-up or pull-down resistors on unused DATA pins are unnecessary.
\overline{RD}	I/O/T (pu_0)	epu ¹	Memory Read. \overline{RD} is asserted whenever the DSP reads from any slave in the system, excluding SDRAM. When the DSP is a slave, \overline{RD} is an input and indicates read transactions that access its internal memory or universal registers. In a multiprocessor system, the bus master drives \overline{RD} . \overline{RD} changes concurrently with ADDR pins.
\overline{WRL}	I/O/T (pu_0)	epu ¹	Write Low. \overline{WRL} is asserted in two cases: when the ADSP-TS201S processor writes to an even address word of external memory or to another external bus agent; and when the ADSP-TS201S processor writes to a 32-bit zone (host, memory, or DSP programmed to 32-bit bus). An external master (host or DSP) asserts \overline{WRL} for writing to a DSP's low word of internal memory. In a multiprocessor system, the bus master drives \overline{WRL} . \overline{WRL} changes concurrently with ADDR pins. When the DSP is a slave, \overline{WRL} is an input and indicates write transactions that access its internal memory or universal registers.
\overline{WRH}	I/O/T (pu_0)	epu ¹	Write High. \overline{WRH} is asserted when the ADSP-TS201S processor writes a long word (64 bits) or writes to an odd address word of external memory or to another external bus agent on a 64-bit data bus. An external master (host or another DSP) must assert \overline{WRH} for writing to a DSP's high word of 64-bit data bus. In a multiprocessing system, the bus master drives \overline{WRH} . \overline{WRH} changes concurrently with ADDR pins. When the DSP is a slave, \overline{WRH} is an input and indicates write transactions that access its internal memory or universal registers.
ACK	I/O/T/OD (pu_od_0)	epu ¹	Acknowledge. External slave devices can deassert ACK to add wait states to external memory accesses. ACK is used by I/O devices, memory controllers, and other peripherals on the data phase. The DSP can deassert ACK to add wait states to read and write accesses of its internal memory. The pull-up is 50 Ω on low-to-high transactions and is 500 Ω on all other transactions.
\overline{BMS}	O/T (pu_0)	na	Boot Memory Select. \overline{BMS} is the chip select for boot EPROM or flash memory. During reset, the DSP uses \overline{BMS} as a strap pin (EBOOT) for EPROM boot mode. In a multiprocessor system, the DSP bus master drives \overline{BMS} . For details, see Reset and Booting on Page 9 and the EBOOT signal description in Table 16 on Page 20 .
$\overline{MS1-0}$	O/T (pu_0)	nc	Memory Select. $\overline{MS0}$ or $\overline{MS1}$ is asserted whenever the DSP accesses memory banks 0 or 1, respectively. $\overline{MS1-0}$ are decoded memory address pins that change concurrently with ADDR pins. When ADDR31:27 = 0b00110, $\overline{MS0}$ is asserted. When ADDR31:27 = 0b00111, $\overline{MS1}$ is asserted. In multiprocessor systems, the master DSP drives $\overline{MS1-0}$.
\overline{MSH}	O/T (pu_0)	nc	Memory Select Host. \overline{MSH} is asserted whenever the DSP accesses the host address space (ADDR31 = 0b1). \overline{MSH} is a decoded memory address pin that changes concurrently with ADDR pins. In a multiprocessor system, the bus master DSP drives \overline{MSH} .
\overline{BRST}	I/O/T (pu_0)	epu ¹	Burst. The current bus master (DSP or host) asserts this pin to indicate that it is reading or writing data associated with consecutive addresses. A slave device can ignore addresses after the first one and increment an internal address counter after each transfer. For host-to-DSP burst accesses, the DSP increments the address automatically while \overline{BRST} is asserted.

I = input; **A** = asynchronous; **O** = output; **OD** = open-drain output; **T** = three-state; **P** = power supply; **G** = ground; **pd** = internal pull-down 5 k Ω ; **pu** = internal pull-up 5 k Ω ; **pd_0** = internal pull-down 5 k Ω on DSP ID = 0; **pu_0** = internal pull-up 5 k Ω on DSP ID = 0; **pu_od_0** = internal pull-up 500 Ω on DSP ID = 0; **pd_m** = internal pull-down 5 k Ω on DSP bus master; **pu_m** = internal pull-up 5 k Ω on DSP bus master; **pu_ad** = internal pull-up 40 k Ω . 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 k Ω to V_{SS} ; epu = external pull-up approximately 5 k Ω 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 This external pull-up may be omitted for the ID = 000 TigerSHARC processor.

ADSP-TS201S

Table 6. Pin Definitions—External Port Arbitration

Signal	Type	Term	Description
$\overline{\text{BR7-0}}$	I/O	$V_{\text{DD_IO}}^1$	Multiprocessing Bus Request Pins. Used by the DSPs in a multiprocessor system to arbitrate for bus mastership. Each DSP drives its own $\overline{\text{BRx}}$ line (corresponding to the value of its ID2–0 inputs) and monitors all others. In systems with fewer than eight DSPs, set the unused $\overline{\text{BRx}}$ pins high ($V_{\text{DD_IO}}$).
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 ($\overline{\text{BR0}}-\overline{\text{BR7}}$) to assert when requesting the bus: 000 = $\overline{\text{BR0}}$, 001 = $\overline{\text{BR1}}$, 010 = $\overline{\text{BR2}}$, 011 = $\overline{\text{BR3}}$, 100 = $\overline{\text{BR4}}$, 101 = $\overline{\text{BR5}}$, 110 = $\overline{\text{BR6}}$, or 111 = $\overline{\text{BR7}}$. ID2–0 must have a constant value during system operation and can change during reset only.
$\overline{\text{BM}}$	O	na	Bus Master. The current bus master DSP asserts $\overline{\text{BM}}$. For debugging only. At reset this is a strap pin. For more information, see Table 16 on Page 20 .
$\overline{\text{BOFF}}$	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 $\overline{\text{BOFF}}$ to force the DSP to relinquish the bus before completing its outstanding transaction.
$\overline{\text{BUSLOCK}}$	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 .
$\overline{\text{HBR}}$	I	epu	Host Bus Request. A host must assert $\overline{\text{HBR}}$ to request control of the DSP's external bus. When $\overline{\text{HBR}}$ is asserted in a multiprocessing system, the bus master relinquishes the bus and asserts $\overline{\text{HBG}}$ once the outstanding transaction is finished.
$\overline{\text{HBG}}$	I/O/T (pu_0)	epu ²	Host Bus Grant. Acknowledges $\overline{\text{HBR}}$ and indicates that the host can take control of the external bus. When relinquishing the bus, the master DSP three-states the $\overline{\text{ADDR31-0}}$, $\overline{\text{DATA63-0}}$, $\overline{\text{MSH}}$, $\overline{\text{MSSD3-0}}$, $\overline{\text{MS1-0}}$, $\overline{\text{RD}}$, $\overline{\text{WRL}}$, $\overline{\text{WRH}}$, $\overline{\text{BMS}}$, $\overline{\text{BRST}}$, $\overline{\text{IORD}}$, $\overline{\text{IOWR}}$, $\overline{\text{IOEN}}$, $\overline{\text{RAS}}$, $\overline{\text{CAS}}$, $\overline{\text{SDWE}}$, SDA10, SDCKE, LDQM, and HDQM pins, and the DSP puts the SDRAM in self-refresh mode. The DSP asserts $\overline{\text{HBG}}$ until the host deasserts $\overline{\text{HBR}}$. In multiprocessor systems, the current bus master DSP drives $\overline{\text{HBG}}$, and all slave DSPs monitor it.
$\overline{\text{CPA}}$	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. $\overline{\text{CPA}}$ is an open drain output, connected to all DSPs in the system. If not required in the system, leave $\overline{\text{CPA}}$ unconnected (external pull-ups will be required for DSP ID = 1 through ID = 7).
$\overline{\text{DPA}}$	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. $\overline{\text{DPA}}$ is an open drain output, connected to all DSPs in the system. If not required in the system, leave $\overline{\text{DPA}}$ 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 k Ω ; pu = internal pull-up 5 k Ω ; pd_0 = internal pull-down 5 k Ω on DSP ID = 0; pu_0 = internal pull-up 5 k Ω on DSP ID = 0; pu_od_0 = internal pull-up 500 Ω on DSP ID = 0; pd_m = internal pull-down 5 k Ω on DSP bus master; pu_m = internal pull-up 5 k Ω on DSP bus master; pu_ad = internal pull-up 40 k Ω . 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 k Ω to V_{SS} ; epu = external pull-up approximately 5 k Ω to $V_{\text{DD_IO}}$; nc = not connected; na = not applicable (always used); $V_{\text{DD_IO}}$ = connect directly to $V_{\text{DD_IO}}$; V_{SS} = connect directly to V_{SS}

¹ The $\overline{\text{BRx}}$ pin matching the ID2–0 input selection for the processor should be left nc if unused. For example, the processor with ID = 000 has $\overline{\text{BR0}} = \text{nc}$ and $\overline{\text{BR7-1}} = V_{\text{DD_IO}}$.

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	e <u>pu</u>	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 (<u>pu_0</u>)	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 (<u>pu_0</u>)	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 (<u>pu_0</u>)	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 k Ω ; **pu** = internal pull-up 5 k Ω ; **pd_0** = internal pull-down 5 k Ω on DSP ID = 0; **pu_0** = internal pull-up 5 k Ω on DSP ID = 0; **pu_od_0** = internal pull-up 500 Ω on DSP ID = 0; **pd_m** = internal pull-down 5 k Ω on DSP bus master; **pu_m** = internal pull-up 5 k Ω on DSP bus master; **pu_ad** = internal pull-up 40 k Ω . 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 k Ω to V_{SS} ; epu = external pull-up approximately 5 k Ω 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

Table 8. Pin Definitions—External Port SDRAM Controller

Signal	Type	Term	Description
$\overline{\text{MSSD3-0}}$	I/O/T (pu_0)	nc	Memory Select SDRAM. $\overline{\text{MSSD0}}$, $\overline{\text{MSSD1}}$, $\overline{\text{MSSD2}}$, or $\overline{\text{MSSD3}}$ is asserted whenever the DSP accesses SDRAM memory space. $\overline{\text{MSSD3-0}}$ are decoded memory address pins that are asserted whenever the DSP issues an SDRAM command cycle (access to ADDR31:30 = 0b01—except reserved spaces shown in Figure 3 on Page 6). In a multi-processor system, the master DSP drives $\overline{\text{MSSD3-0}}$.
$\overline{\text{RAS}}$	I/O/T (pu_0)	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, it defines the type of operation to execute according to SDRAM specification.
$\overline{\text{CAS}}$	I/O/T (pu_0)	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, it defines the type of operation to execute according to the SDRAM specification.
LDQM	O/T (pu_0)	nc	Low Word SDRAM Data Mask. When sampled high, three-states the SDRAM DQ buffers. LDQM is valid on SDRAM transactions when $\overline{\text{CAS}}$ is asserted, and 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	O/T (pu_0)	nc	High Word SDRAM Data Mask. When sampled high, three-states the SDRAM DQ buffers. HDQM is valid on SDRAM transactions when $\overline{\text{CAS}}$ is asserted, and inactive on read transactions. On write transactions, HDQM is active when accessing an even address in word accesses or when memory is configured for a 32-bit bus to disable the write of the high word.
SDA10	O/T (pu_0)	nc	SDRAM Address Bit 10. Separate A10 signals enable SDRAM refresh operation while the DSP executes non-SDRAM transactions.
SDCKE	I/O/T (pu_m/ pd_m)	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 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 pull-down before granting the bus to the host.
$\overline{\text{SDWE}}$	I/O/T (pu_0)	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.

I = input; **A** = asynchronous; **O** = output; **OD** = open-drain output; **T** = three-state; **P** = power supply; **G** = ground; **pd** = internal pull-down 5 k Ω ; **pu** = internal pull-up 5 k Ω ; **pd_0** = internal pull-down 5 k Ω on DSP ID = 0; **pu_0** = internal pull-up 5 k Ω on DSP ID = 0; **pu_od_0** = internal pull-up 500 Ω on DSP ID = 0; **pd_m** = internal pull-down 5 k Ω on DSP bus master; **pu_m** = internal pull-up 5 k Ω on DSP bus master; **pu_ad** = internal pull-up 40 k Ω . 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 k Ω to V_{SS} ; epu = external pull-up approximately 5 k Ω 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 9. Pin Definitions—JTAG Port

Signal	Type	Term	Description
EMU	O/OD	nc ¹	Emulation. Connected to the DSP's JTAG emulator target board connector only.
TCK	I	epd or epu ¹	Test Clock (JTAG). Provides an asynchronous clock for JTAG scan.
TDI	I (pu_ad)	nc ¹	Test Data Input (JTAG). A serial data input of the scan path.
TDO	O/T	nc ¹	Test Data Output (JTAG). A serial data output of the scan path.
TMS	I (pu_ad)	nc ¹	Test Mode Select (JTAG). Used to control the test state machine.
TRST	I/A (pu_ad)	na	Test Reset (JTAG). Resets the test state machine. TRST must be asserted or pulsed low after power up for proper device operation. For more information, see Reset and Booting on Page 9 .

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 See the reference on [Page 11](#) to the JTAG emulation technical reference EE-68.

Table 10. Pin Definitions—Flags, Interrupts, and Timer

Signal	Type	Term	Description
FLAG3–0	I/O/A (pu)	nc	FLAG pins. Bidirectional input/output pins can be used as program conditions. Each pin can be configured individually for input or for output. FLAG3–0 are inputs after power-up and reset.
IRQ3–0	I/A (pu)	nc	Interrupt Request. When asserted, the DSP generates an interrupt. Each of the IRQ3–0 pins can be independently set for edge-triggered or level-sensitive operation. After reset, these pins are disabled unless the IRQ3–0 strap option and interrupt vectors are initialized for booting.
TMR0E	O	na	Timer 0 expires. This output pulses whenever timer 0 expires. At reset, this is a strap pin. For more information, see Table 16 on Page 20 .

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

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_I0}	Link Ports 3-0 Data 3-0 Receive LVDS P
LxDATI3-0N	I	V _{DD_I0}	Link Ports 3-0 Data 3-0 Receive LVDS N
LxCLKINP	I/A	V _{DD_I0}	Link Ports 3-0 Receive Clock LVDS P
LxCLKINN	I/A	V _{DD_I0}	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_I)	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 k Ω ; **pu** = internal pull-up 5 k Ω ; **pd_0** = internal pull-down 5 k Ω on DSP ID = 0; **pu_0** = internal pull-up 5 k Ω on DSP ID = 0; **pu_od_0** = internal pull-up 500 Ω on DSP ID = 0; **pd_m** = internal pull-down 5 k Ω on DSP bus master; **pu_m** = internal pull-up 5 k Ω on DSP bus master; **pu_ad** = internal pull-up 40 k Ω ; **pd_I** = internal pull-down 50 k Ω . 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 k Ω to V_{SS}; epu = external pull-up approximately 5 k Ω to V_{DD_I0}; nc = not connected; na = not applicable (always used); V_{DD_I0} = connect directly to V_{DD_I0}; V_{SS} = connect directly to V_{SS}

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

Signal	Type	Term	Description
CONTROLIMP0	I (pd)	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 .
CONTROLIMP1	I (pu)	na	
DS2_0	I (pu)	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: $\overline{\text{CPA}}$, $\overline{\text{DPA}}$, $\overline{\text{TDO}}$, $\overline{\text{EMU}}$, and $\overline{\text{RST_OUT}}$. The drive strength for the ACK pin is always x2 drive strength 7 (100%).
DS1	I (pd)	na	
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 k Ω ; **pu** = internal pull-up 5 k Ω ; **pd_0** = internal pull-down 5 k Ω on DSP ID = 0; **pu_0** = internal pull-up 5 k Ω on DSP ID = 0; **pu_od_0** = internal pull-up 500 Ω on DSP ID = 0; **pd_m** = internal pull-down 5 k Ω on DSP bus master; **pu_m** = internal pull-up 5 k Ω on DSP bus master; **pu_ad** = internal pull-up 40 k Ω . 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 k Ω to V_{SS}; epu = external pull-up approximately 5 k Ω to V_{DD_I0}; nc = not connected; na = not applicable (always used); V_{DD_I0} = connect directly to V_{DD_I0}; V_{SS} = connect directly to V_{SS}

Table 13. Impedance Control Selection

CONTROLIMP1-0	Driver Mode
00 (recommended)	Normal
01	Reserved
10 (default)	A/D Mode
11	Reserved

Table 14. Drive Strength/Output Impedance Selection

DS2-0 Pins	Drive Strength ¹	Output Impedance ²
000	Strength 0 (11.1%)	26 Ω
001	Strength 1 (23.8%)	32 Ω
010	Strength 2 (36.5%)	40 Ω
011	Strength 3 (49.2%)	50 Ω
100	Strength 4 (61.9%)	62 Ω
101 (default)	Strength 5 (74.6%)	70 Ω
110	Strength 6 (87.3%)	96 Ω
111	Strength 7 (100%)	120 Ω

¹ CONTROLIMP1 = 0, A/D mode disabled.

² CONTROLIMP1 = 1, A/D mode enabled.

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

Signal	Type	Term	Description
V _{DD}	P	na	V _{DD} pins for internal logic.
V _{DD_A}	P	na	V _{DD} pins for analog circuits. Pay critical attention to bypassing this supply.
V _{DD_IO}	P	na	V _{DD} pins for I/O buffers.
V _{DD_DRAM}	P	na	V _{DD} pins for internal DRAM.
V _{REF}	I	na	Reference voltage defines the trip point for all input buffers, except SCLK, $\overline{\text{RST_IN}}$, $\overline{\text{POR_IN}}$, $\overline{\text{IRQ3-0}}$, $\overline{\text{FLAG3-0}}$, $\overline{\text{DMAR3-0}}$, $\overline{\text{ID2-0}}$, CONTROLIMP1-0, LxDATO3-0P/N, LxCLKOUTP/N, LxDATI3-0P/N, LxCLKINP/N, TCK, TDI, TMS, and $\overline{\text{TRST}}$. V _{REF} can be connected to a power supply or set by a voltage divider circuit as shown in Figure 6 . For more information, see Filtering Reference Voltage and Clocks on Page 10 .
SCLK_V _{REF}	I	na	System Clock Reference. Connect this pin to a reference voltage as shown in Figure 7 . For more information, see Filtering Reference Voltage and Clocks on Page 10 .
V _{SS}	G	na	Ground pins.
NC	—	nc	No Connect. Do not connect these pins to anything (not to any supply, signal, or each other). These pins are reserved and must be left unconnected.

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}

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 internal pull-up or pull-down for the default value. If a strap pin is not connected to an overdriving external pull-up, pull-down, or logic load, the DSP samples the default value during reset. If strap pins are

connected to logic inputs, a stronger external pull-up or 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 or pull-down. [Table 16](#) lists and describes each of the DSP's strap pins.

Table 16. Pin Definitions—I/O Strap Pins

Signal	Type (at Reset)	On Pin ...	Description
EBOOT	I (pd_0)	$\overline{\text{BMS}}$	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	I (pd)	$\overline{\text{BM}}$	Interrupt Enable. 0 = disable and set $\overline{\text{IRQ3-0}}$ interrupts to edge-sensitive after reset (default) 1 = enable and set $\overline{\text{IRQ3-0}}$ interrupts to level-sensitive immediately after reset
LINK_DWIDTH	I (pd)	TMR0E	Link Port Input Default Data Width. 0 = 1-bit (default) 1 = 4-bit
SYS_REG_WE	I (pd_0)	$\overline{\text{BUSLOCK}}$	SYSCON and SDRCON Write Enable. 0 = one-time writable after reset (default) 1 = always writable
TM1	I (pu)	$\overline{\text{L1BCMPO}}$	Test Mode 1. Do not overdrive default value during reset.
TM2	I (pu)	$\overline{\text{L2BCMPO}}$	Test Mode 2. Do not overdrive default value during reset.
TM3	I (pu)	$\overline{\text{L3BCMPO}}$	Test Mode 3. Do not overdrive default value during reset.

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

When default configuration is used, no external resistor is needed on the strap pins. To apply other configurations, a 500 Ω resistor connected to V_{DDIO} is required. If providing external pull-downs, do not strap these pins directly to V_{SS} ; the strap pins require 500 Ω resistor straps.

All strap pins are sampled on the rising edge of $\overline{\text{RST_IN}}$ (deassertion edge). Each pin latches the strapped pin state (state of the strap pin at the rising edge of $\overline{\text{RST_IN}}$). Shortly after deassertion of $\overline{\text{RST_IN}}$, these pins are reconfigured to their normal functionality.

These strap pins have an internal pull-down resistor, pull-up resistor, or no-resistor (three-state) on each pin. The resistor type, which is connected to the I/O pad, depends on whether $\overline{\text{RST_IN}}$ is active (low) or if $\overline{\text{RST_IN}}$ is deasserted (high).

[Table 17](#) shows the resistors that are enabled during active reset and during normal operation.

Table 17. Strap Pin Internal Resistors—Active Reset ($\overline{\text{RST_IN}} = 0$) vs. Normal Operation ($\overline{\text{RST_IN}} = 1$)

Pin	$\overline{\text{RST_IN}} = 0$	$\overline{\text{RST_IN}} = 1$
$\overline{\text{BMS}}$	(pd_0)	(pu_0)
$\overline{\text{BM}}$	(pd)	Driven
TMR0E	(pd)	Driven
$\overline{\text{BUSLOCK}}$	(pd_0)	(pu_0)
$\overline{\text{L1BCMPO}}$	(pu)	Driven
$\overline{\text{L2BCMPO}}$	(pu)	Driven
$\overline{\text{L3BCMPO}}$	(pu)	Driven

pd = internal pull-down 5 k Ω ; **pu** = internal pull-up 5 k Ω ;
pd_0 = internal pull-down 5 k Ω on DSP ID = 0;
pu_0 = internal pull-up 5 k Ω on DSP ID = 0

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, RD, WR, WRL, WRH, ACK, BRST, BR7-0, BOFF, HBR, HBG, MSSD3-0, RAS, CAS, SDCKE, 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, LxBCMPLI, LxACKI, POR_IN, RST_IN, IRQ3-0, CPA, DPA, 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.

ADSP-TS201S

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	@ $V_{DD_IO} = \text{Min}$, $I_{OH} = -2 \text{ mA}$	2.18		V
V_{OL}	Low Level Output Voltage ¹	@ $V_{DD_IO} = \text{Min}$, $I_{OL} = 4 \text{ mA}$		0.4	V
I_{IH}	High Level Input Current	@ $V_{DD_IO} = \text{Max}$, $V_{IN} = V_{IH} \text{ Max}$		20	μA
I_{IH_PU}	High Level Input Current	@ $V_{DD_IO} = \text{Max}$, $V_{IN} = V_{IH} \text{ Max}$		20	μA
I_{IH_PD}	High Level Input Current	@ $V_{DD_IO} = \text{Max}$, $V_{IN} = V_{DD_IO} \text{ Max}$	0.3	0.76	mA
$I_{IH_PD_L}$	High Level Input Current	@ $V_{DD_IO} = \text{Max}$, $V_{IN} = V_{IH} \text{ Max}$	30	76	μA
I_{IL}	Low Level Input Current	@ $V_{DD_IO} = \text{Max}$, $V_{IN} = 0 \text{ V}$		20	μA
I_{IL_PU}	Low Level Input Current	@ $V_{DD_IO} = \text{Max}$, $V_{IN} = 0 \text{ V}$	0.3	0.76	mA
$I_{IL_PU_AD}$	Low Level Input Current	@ $V_{DD_IO} = \text{Max}$, $V_{IN} = 0 \text{ V}$	30	100	μA
I_{OZH}	Three-State Leakage Current High	@ $V_{DD_IO} = \text{Max}$, $V_{IN} = V_{IH} \text{ Max}$		50	μA
I_{OZH_PD}	Three-State Leakage Current High	@ $V_{DD_IO} = \text{Max}$, $V_{IN} = V_{DD_IO} \text{ Max}$	0.3	0.76	mA
I_{OZL}	Three-State Leakage Current Low	@ $V_{DD_IO} = \text{Max}$, $V_{IN} = 0 \text{ V}$		20	μA
I_{OZL_PU}	Three-State Leakage Current Low	@ $V_{DD_IO} = \text{Max}$, $V_{IN} = 0 \text{ V}$	0.3	0.76	mA
$I_{OZL_PU_AD}$	Three-State Leakage Current Low	@ $V_{DD_IO} = \text{Max}$, $V_{IN} = 0 \text{ V}$	30	100	μA
I_{OZL_OD}	Three-State Leakage Current Low	@ $V_{DD_IO} = \text{Max}$, $V_{IN} = 0 \text{ V}$	4	7.6	mA
C_{IN}	Input Capacitance ^{2,3}	@ $f_{IN} = 1 \text{ MHz}$, $T_{CASE} = 25^\circ\text{C}$, $V_{IN} = 2.5 \text{ V}$		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.

PACKAGE INFORMATION

The information presented in [Figure 8](#) provide details about the package branding for the ADSP-TS201S processors. For a complete listing of product availability, see [Ordering Guide on Page 46](#).



Figure 8. Typical Package Brand

Table 19. Package Brand Information

Brand Key	Field Description
t	Temperature Range
pp	Package Type
Z	Lead Free Option (optional)
ccc	See Ordering Guide
LLLLLLLLL-L	Silicon Lot Number
R.R	Silicon Revision
yyww	Date Code
vvvvv	Assembly Lot Code

ABSOLUTE MAXIMUM RATINGS

Stresses greater than those listed below may cause permanent damage to the device. These are stress ratings only. Functional operation of the device at these or any other conditions greater than those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Table 20. Absolute Maximum Ratings

Parameter	Rating
Internal (Core) Supply Voltage (V_{DD})	-0.3 V to +1.4 V
Analog (PLL) Supply Voltage ($V_{DD,A}$)	-0.3 V to +1.4 V
External (I/O) Supply Voltage ($V_{DD,IO}$)	-0.3 V to +3.5 V
External (DRAM) Supply Voltage ($V_{DD,DRAM}$)	-0.3 V to +2.1 V
Input Voltage ¹	-0.63 V to +3.93 V
Output Voltage Swing	-0.5 V to $V_{DD,IO}+0.5$ V
Storage Temperature Range	-65°C to +150°C

¹ Applies to 10% transient duty cycle. For other duty cycles see [Table 18](#).

ESD SENSITIVITY

	<p>ESD (electrostatic discharge) sensitive device. Charged devices and circuit boards can discharge without detection. Although this product features patented or proprietary circuitry, damage may occur on devices subjected to high energy ESD. Therefore, proper ESD precautions should be take to avoid performance degradation or loss of functionality.</p>
--	---

ADSP-TS201S

TIMING SPECIFICATIONS

With the exception of $\overline{\text{DMAR3-0}}$, $\overline{\text{IRQ3-0}}$, TMROE , and FLAG3-0 (input only) pins, all ac timing for the ADSP-TS201S processor is relative to a reference clock edge. Because input setup/hold, output valid/hold, and output enable/disable times are relative to a clock edge, the timing data for the ADSP-TS201S processor has few calculated (formula-based) values. For information on ac timing, see [General AC Timing](#). For information on link port transfer timing, see [Link Port Low Voltage, Differential-Signal \(LVDS\) Electrical Characteristics, and Timing on Page 30](#).

General AC Timing

Timing is measured on signals when they cross the 1.25 V level as described in [Figure 15 on Page 29](#). All delays (in nanoseconds) are measured between the point that the first signal reaches 1.25 V and the point that the second signal reaches 1.25 V.

The general ac timing data appears in [Table 22](#) and [Table 29](#). All ac specifications are measured with the load specified in [Figure 36 on Page 38](#), and with the output drive strength set to strength 4. In order to calculate the output valid and hold times for different load conditions and/or output drive strengths, refer to [Figure 37 on Page 38](#) through [Figure 44 on Page 39](#) (Rise and Fall Time vs. Load Capacitance) and [Figure 45 on Page 39](#) (Output Valid vs. Load Capacitance and Drive Strength).

The ac asynchronous timing data for the $\overline{\text{IRQ3-0}}$, $\overline{\text{DMAR3-0}}$, FLAG3-0 , and TMROE pins appears in [Table 21](#).

Table 21. AC Asynchronous Signal Specifications

Name	Description	Pulse Width Low (Min)	Pulse Width High (Min)
$\overline{\text{IRQ3-0}}^1$	Interrupt Request	$2 \times t_{\text{SCLK}}$ ns	$2 \times t_{\text{SCLK}}$ ns
$\overline{\text{DMAR3-0}}^1$	DMA Request	$2 \times t_{\text{SCLK}}$ ns	$2 \times t_{\text{SCLK}}$ ns
FLAG3-0^2	FLAG3-0 Input	$2 \times t_{\text{SCLK}}$ ns	$2 \times t_{\text{SCLK}}$ ns
TMROE^3	Timer 0 Expired	$4 \times t_{\text{SCLK}}$ ns	—

¹ These input pins have Schmitt triggers and therefore do not need to be synchronized to a clock reference.

² For output specifications on FLAG3-0 pins, see [Table 29](#).

³ This pin is a strap option. During reset, an internal resistor pulls the pin low.

Table 22. Reference Clocks—Core Clock (CCLK) Cycle Time

Parameter	Description	Grade = 060 (600 MHz)		Grade = 050 (500 MHz)		Unit
		Min	Max	Min	Max	
t_{CCLK}^1	Core Clock Cycle Time	1.67	12.5	2.0	12.5	ns

¹ CCLK is the internal processor clock or instruction cycle time. The period of this clock is equal to the system clock period (t_{SCLK}) divided by the system clock ratio (SCLKRAT2-0). For information on available part numbers for different internal processor clock rates, see the [Ordering Guide on Page 46](#).

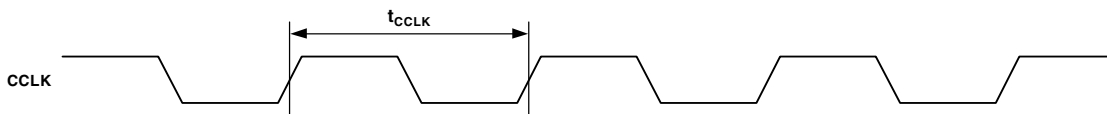


Figure 9. Reference Clocks—Core Clock (CCLK) Cycle Time

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	$0.40 \times t_{SCLK}$	$0.60 \times t_{SCLK}$	$0.45 \times t_{SCLK}$	$0.55 \times t_{SCLK}$	ns
t_{SCLKL}	System Clock Cycle Low Time	$0.40 \times t_{SCLK}$	$0.60 \times t_{SCLK}$	$0.45 \times t_{SCLK}$	$0.55 \times t_{SCLK}$	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.

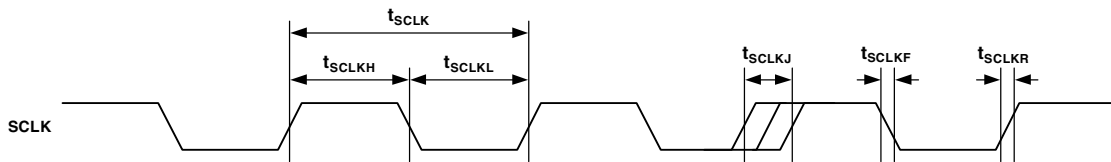


Figure 10. Reference Clocks—System Clock (SCLK) Cycle Time

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 $t_{CCLK} \times 4$	—	ns
t_{TCKH}	Test Clock (JTAG) Cycle High Time	12	—	ns
t_{TCKL}	Test Clock (JTAG) Cycle Low Time	12	—	ns

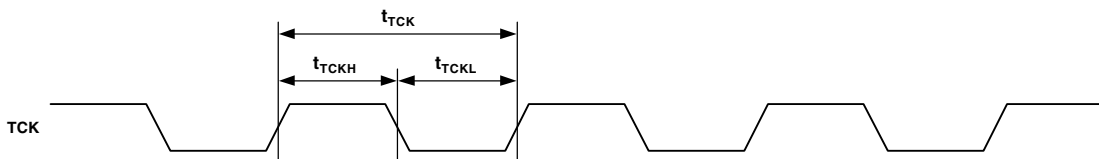


Figure 11. Reference Clocks—JTAG Test Clock (TCK) Cycle Time