ANALOG DEVICES

TIGER SHARC™ DSP Microcomputer

Preliminary Technical Data ADSP-TS001

KEY FEATURES

- Ultra high-performance 32-bit DSP optimized for computationally demanding, large signal processing tasks and multichannel applications
- Support for IEEE-compatible single-precision floating-point and extended-precision floating-point numeric formats and 8-, 16-, and 32-bit fixed-point numeric formats
- Static superscalar architecture combines and balances RISC, VLIW, and DSP functionality to provide:
  - Load and store architecture that supports 32-, 64-, or 128-bit data through direct or broadcast accesses
  - Static branch prediction
  - Interlocked, multiport register files; 128 registers
  - Instruction-level parallelism determined before runtime
  - I/O and internal memory capable of sustaining core rates
  - Hardware support for circular buffers, bit reverse, and zero-overhead looping

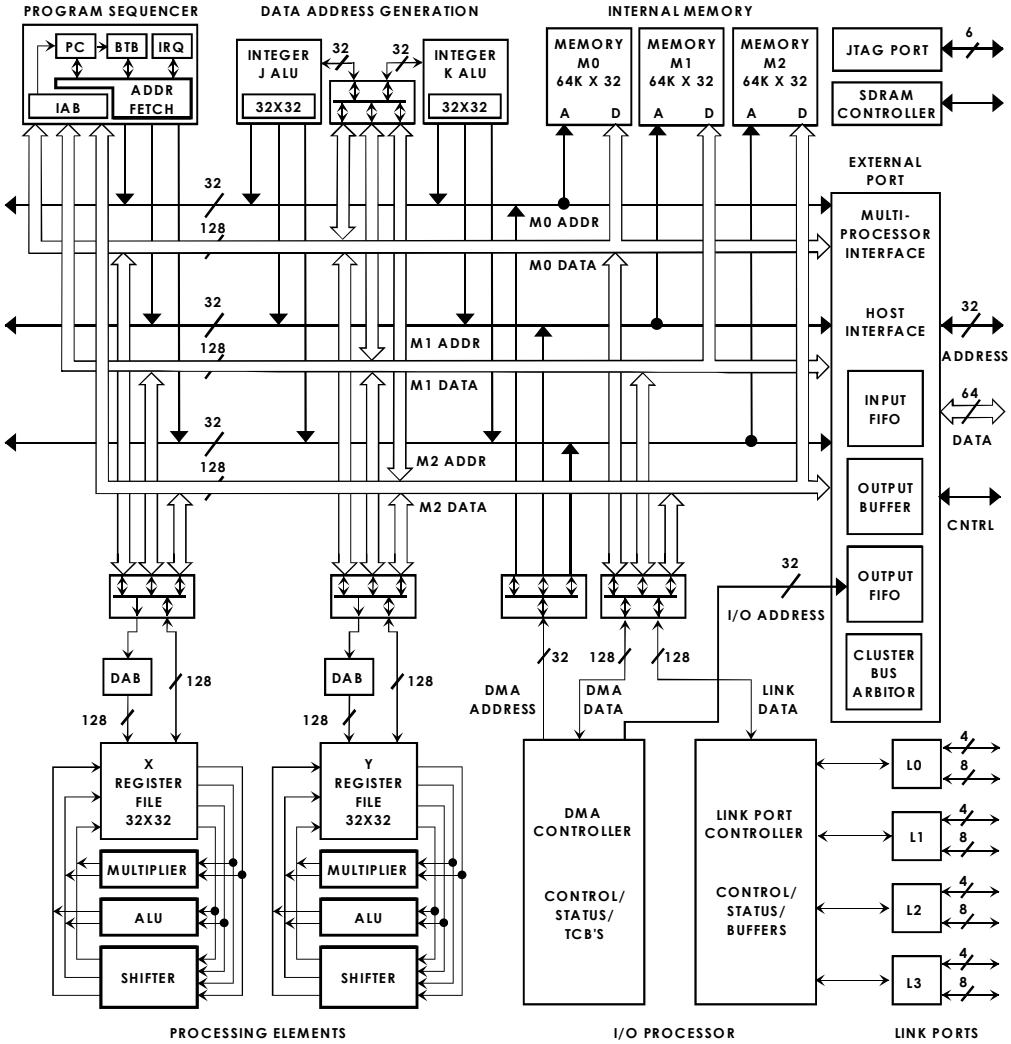


Figure 1 ADSP-TS001 block diagram

REV. PrC

One Technology Way
P.O. Box 9106
Norwood MA 02062-9106
U.S.A.

http://www.analog.com/dsp
Tel: 1-800-ANALOG-D
Fax: 1-781-461-3010

PERFORMANCE

- Runs at 150 MHz, 6.67ns instruction rate
- Executes from 1 to 4 instructions per cycle; maximum of 6 instructions
- Has internal bandwidth of 7.2 Gbytes/s and performs a complete context switch in <20 cycles (134 ns)
- Executes six single-precision floating-point or twenty-four 16-bit fixed-point operations per cycle (900 MFLOPS or 3.6 GOPS performance)
- Provides 8 MACs/cycle peak and 7.1 MACs/cycle sustained 16-bit performance and 2 MACs/cycle peak and 1.8 MACs/cycle sustained 32-bit performance (based on FIR)
- Executes 8 Add/Compare/Select (ACS) sequences for Viterbi algorithm in < 1 cycle
- Has an 8-cycle instruction pipeline (3-cycle fetch pipe and 5-cycle execution pipe)—computation results available 2 cycles after operation begins

FEATURES

- Fully interruptible programming model with flexible programming in Assembly and C languages
- Single-instruction multiple-data (SIMD) operations supported by two computation blocks, each with an ALU, multiplier, shifter, and 32-word register file
- Integrated peripherals include 6 Mbit on-chip SRAM, glueless multiprocessing features, and ports (link, external bus, and JTAG)
- 64-bit generalized bit manipulation unit
- Instruction set
  - Algebraic assembly language syntax
  - Directly supports all DSP, imaging, and video arithmetic types, eliminating hardware modes
  - Branch prediction encoded in instruction, enables zero-overhead loops
  - Parallelism encoded in instruction line
  - Predicated execution optional for all instructions
  - User defines partitioning between program and data memory
- 6 Mbit of on-chip SRAM memory
  - Divided into three 2 Mbit banks
  - Each bank connects to one of the three128-bit wide internal buses
  - Core can access 128 bits from each bank each cycle
  - Program stored in quad-words with no wasted space
  - Unified, linear memory map for all internal and external memory
- Dual computation blocks
  - Each has an ALU, multiplier, 64-bit shifter, and 32-word register file
  - Register files are fully orthogonal
  - Two-cycle computation pipeline provides results 2 cycles after an operation begins
  - Data Alignment Buffer (DAB) enables nonaligned accesses of memory, for efficient implementation of FIR filters
  - Each operates on one to eight data elements
  - Integer, fractional, signed, and unsigned data types, with optional saturation
  - Performs two complex, 16-bit MACs per cycle
- Dual integer ALUs (IALUs)
  - Provide memory addresses for data and update pointers
  - Support circular buffering and bit-reverse addressing
  - Perform general-purpose integer operations, increasing programming flexibility
  - Each has a 32-word register file
- 128-entry Branch Target Buffer (BTB) works in conjunction with branch prediction to provide zero-overhead looping

- Instruction Alignment Buffer (IAB) supports compact code by eliminating the requirement to align code in memory
- Two 32-bit interval timers
- External port
  - Connection to 32- or 64-bit wide external bus
  - Data transfer rate of 600 Mbytes/s over external bus
  - Glueless interface to SDRAM and other off-chip memory and other peripherals
- DMA controller
  - Supports fourteen DMA channels
  - Performs DMA transfers between internal memory and 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
- Multiprocessor interface supports up to eight DSPs on a common bus
  - On-chip arbitration for glueless multiprocessing
  - Globally accessible internal registers and memory
  - Semaphore support
- Four link ports
  - Hardware-controlled, full-duplex (one transmit and one receive port per link)
  - DMA data transfer rate of 150 Mbytes/s (double data rate)
- Four general-purpose flag I/O ports
- Standard 1149.1 IEEE JTAG test access port and on-chip emulation
- Debug functionality available through a debug monitor or the JTAG test access port
  - Three sets of programmable memory-address watch points, able to operate in parallel, enable users to design a set of conditions at which to halt and examine the state of the DSP
  - Eight trace buffers enable users to recreate the path taken by the program for the last four to eight branches
  - Performance monitors indicate the number of times a specified event occurs during a run
- User and Supervisor modes available for OS kernel and debug monitor support

GENERAL DESCRIPTION

The ADSP-TS001 TigerSHARC™ 32-bit DSP is a high-performance, static superscalar, next generation SHARC processor. It combines dual computation blocks with native support for 32- and 40-bit floating-point and 8-, 16-, 32-, and 64-bit fixed-point processing with very wide memory widths to set a new standard of performance for digital signal processors. Its static superscalar architecture enables the DSP each cycle to execute up to six instructions, performing twenty-four 16-bit fixed-point operations or six floating-point operations.

Three independent 128-bit wide internal data buses, each connecting to one of the three 2 Mbit memory banks, enable quad-word data, instruction, and I/O accesses and provide 7.2 Gbytes/sec of internal memory bandwidth. Operating at 150 MHz, the ADSP-TS001's core has a 6.67ns instruction cycle time. Using its SIMD features, the ADSP-TS001 can perform 1200 million 40-bit MACs or 300 million 80-bit MACs per second. Table 1 shows the DSP's performance benchmarks.

Table 1. ADSP-TS001 benchmarks (@ 150 MHz)

Benchmark	Speed	Clock Cycles
32-bit Algorithm 300 million MACs/s peak performance		
1024 Point Complex FFT (Radix 2)	69 us	10,300
50-tap FIR on 1024 input	183 us	27,500
• Single FIR MAC	3.7 ns	0.55
16-bit Algorithm 1200 million MACs/s peak performance		
256 Point Complex FFT (Radix 2)	7.3 us	1,100
50-tap FIR on 1024 input	48 us	7,200
• Single FIR MAC	0.93 ns	0.14
• Single Complex FIR MAC	3.8 ns	0.57
I/O		
DMA Transfer RateExternal port Link ports (each)	600 Mbytes/s 150 Mbytes/s	NA

Figure 1, "ADSP-TS001 block diagram," on page 1 shows the ADSP-TS001's architectural blocks:

- Two computation blocks, each consisting of an ALU, multiplier, 64-bit shifter, and 32-word register file
- Data Alignment Buffer (DAB)
- Two integer ALUs (IALUs), each with its own 32-word register file
- Program Sequencer with Instruction Alignment Buffer (IAB) and Branch Target Buffer (BTB)
- Three 128-bit internal data buses, each connecting to one of three 2Mbit memory banks
- On-chip SRAM (6Mbit)
- Two 32-bit interval timers
- External port that provides the interface to:
  - Host
  - Multiprocessor
  - Off-chip memory and memory-mapped peripherals
  - SDRAM
- DMA controller
- Link ports
- JTAG test access port

Figure 2, "ADSP-TS001 single-processor system with external SDRAM," on page 5 shows a typical single-processor system with external SDRAM. Figure 4, "ADSP-TS001 shared memory multiprocessing system," on page 11 shows a typical multiprocessor system.

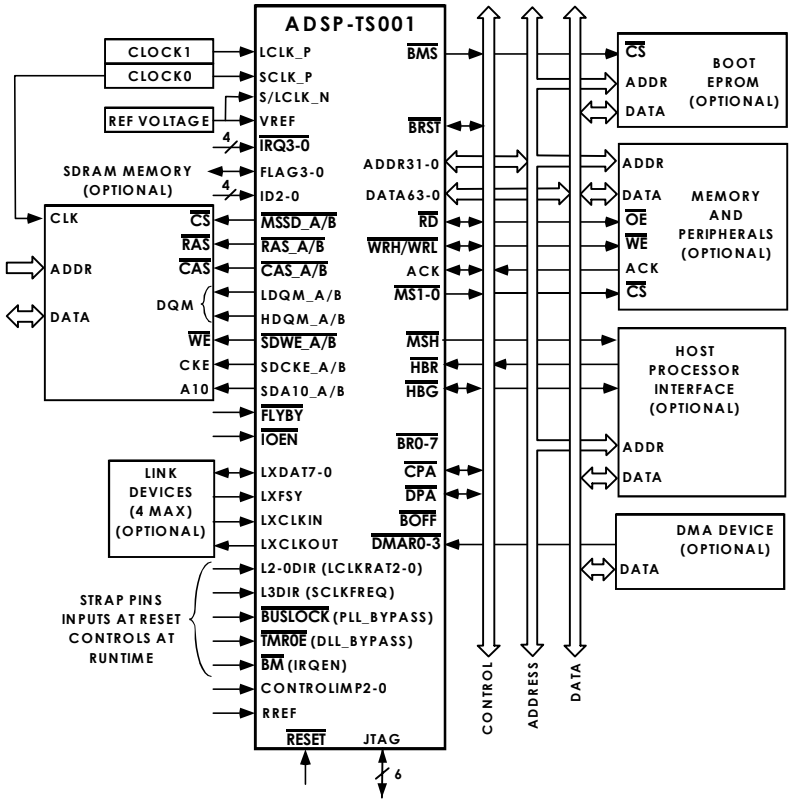


Figure 2 ADSP-TS001 single-processor system with external SDRAM

ADSP-TS001 TIGERSHARC CORE ARCHITECTURAL FEATURES

The ADSP-TS001 is a new family of DSPs. It is not code-compatible at the assembly level with other DSPs in the SHARC family. The ASDP-TS001's static superscalar architecture's features include:

- Instruction parallelism and SIMD operation
- Independent, parallel computation blocks, each with an ALU, multiplier, shifter, and register file
- Data alignment buffer (DAB)
- Integer ALUs for data addressing and general-purpose integer operations
- Program sequencer with an Instruction Alignment Buffer (IAB) and a Branch Target Buffer (BTB)
- Interrupts
- Flexible and comprehensive instruction set eliminates hardware modes

Instruction Parallelism and SIMD Operation

As a static superscalar DSP, the ADSP-TS001's core can execute simultaneously from one to four 32-bit instructions encoded in an instruction line. 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 an eight-deep processor pipeline.

The DSP has a set of instruction parallelism rules that programmers must follow when encoding an instruction line. In general, the selection of instructions the DSP can execute in parallel each cycle depends on the instruction line resources each requires and on the source and destination registers used. The programmer has direct control of three core components—the IALUs, the Computation Blocks, and the Program Sequencer.

The ADSP-TS001, 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 stall cycles. Efficient programming with dependency-free instructions can eliminate most computational and memory transfer data dependencies.

Instruction parallelism rules and data dependencies are documented in detail in the DSP's user's guide.

In addition, the ADSP-TS001 supports SIMD operations two ways—SIMD Compute Blocks and SIMD computations.The programmer can direct both Compute Blocks to operate on the same data (broadcast distribution) or on different data (merged distribution). In addition, each Compute Block can execute four 16-bit or eight 8-bit *SIMD computations* in parallel.

Independent, Parallel Compute Blocks

The ADSP-TS001 has two computation blocks, called Compute Blocks, that can operate either independently in parallel or 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, or shifter to perform independent, simultaneous operations.

The Compute Blocks are referenced as CBX and CBY, and each contains three computational units—an ALU, a multiplier, a 64-bit shifter—and a 32-word register file.

- Register File—Each Compute Block has a multiported 32-word register file used for transferring data between the computation units and data buses and for storing intermediate results. Instructions can access the registers in the register file individually (word-aligned), or in sets of two (dual-aligned) or four (quad-aligned).
- ALU—The ALU performs a standard set of arithmetic operations in both fixed- and floating-point formats. It also performs logic operations.
- Multiplier—The multiplier performs both fixed- and floating-point multiplication and fixed-point multiply and accumulate.
- Shifter—The 64-bit shifter performs logical and arithmetic shifts, bit and bitstream manipulation, and field deposit and extraction operations.

Data Alignment Buffer (DAB)

The DAB is a two 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.

IALUs

The ADSP-TS001 has two integer ALUs (IALUs) that provide powerful address generation capabilities and perform many general-purpose integer operations. Each IALU has a multiported 32-word register file.

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

The IALUs enable implementation of circular data buffers in hardware. 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, 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

To manage program structures and program flow, the program sequencer:

- Supplies addresses to memory for instruction fetches.

  The 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.

- Determines program flow according to JUMP, CALL, RTI, RTS instructions, loop structures, conditions, interrupts, and software exceptions.

- Using branch prediction and a 128-entry branch target buffer (BTB), reduces branch delays for efficient execution of conditional and unconditional branch instructions.

  Correctly predicted branches that are taken occur with zero overhead cycles, overcoming the 2- to 6-stage branch penalty.

- Decrements the loop counters.

- Handles hardware interrupts with high throughput and no aborted instruction cycles.

Interrupts

The DSP supports nested and nonnested interrupts. Each interrupt has a register in the interrupt vector table, the interrupt latch register, and the interrupt mask register. All interrupts are fixed as either level-sensitive or edge-sensitive, except the $\overline{IRQ}_{3-0}$ 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.

The instruction set directly supports all DSP, image, and video arithmetic types, including signed, unsigned, fractional, and integer data types. This frees the DSP from hardware modes and associated wasted cycles and simplifies compiler operation.

ADSP-TS001 MEMORY AND I/O INTERFACE FEATURES

The ASDP-TS001 includes these memory and I/O interface components and features:

- On-chip SRAM Memory
- Off-chip memory and peripherals interfaces through the external port
  - Host interface
  - Multiprocessing interface
  - SDRAM interface
  - EPROM interface
- DMA controller
- Link ports

On-Chip SRAM Memory

The ADSP-TS001 has 6Mbits of on-chip SRAM memory, divided into three blocks of 2Mbits (64K words × 32bits). Each block—M0, M1, and M2—can store program, data, or both, so applications can configure memory to suit their specific needs. However, placing program instructions and data in different memory blocks enables the DSP to access data while it performs an instruction fetch.

Each memory block connects to one of the 128-bit wide internal buses—block M0 to bus MD0, block M1 to bus MD1, and block M2 to bus MD2—enabling the DSP to perform three memory transfers in the same cycle. The DSP's internal bus architecture provides a total memory bandwidth of 7.2 Gbytes/second, enabling the core and I/O to access eight 32-bit data words (256 bits) and four 32-bit instructions each cycle.

The DSP's flexible memory structure enables:

- The DSP's core and I/O to access different memory blocks in the same cycle.
- The DSP's core to access all three memory blocks in parallel (one instruction and two data accesses)
- Applications to define the partition between program and data memory.
- Applications to access all memory as 32-, 64, or 128-bit words (and 16-bit words with DAB).

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, "ADSP-TS001 Memory Map," on page 9.

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.

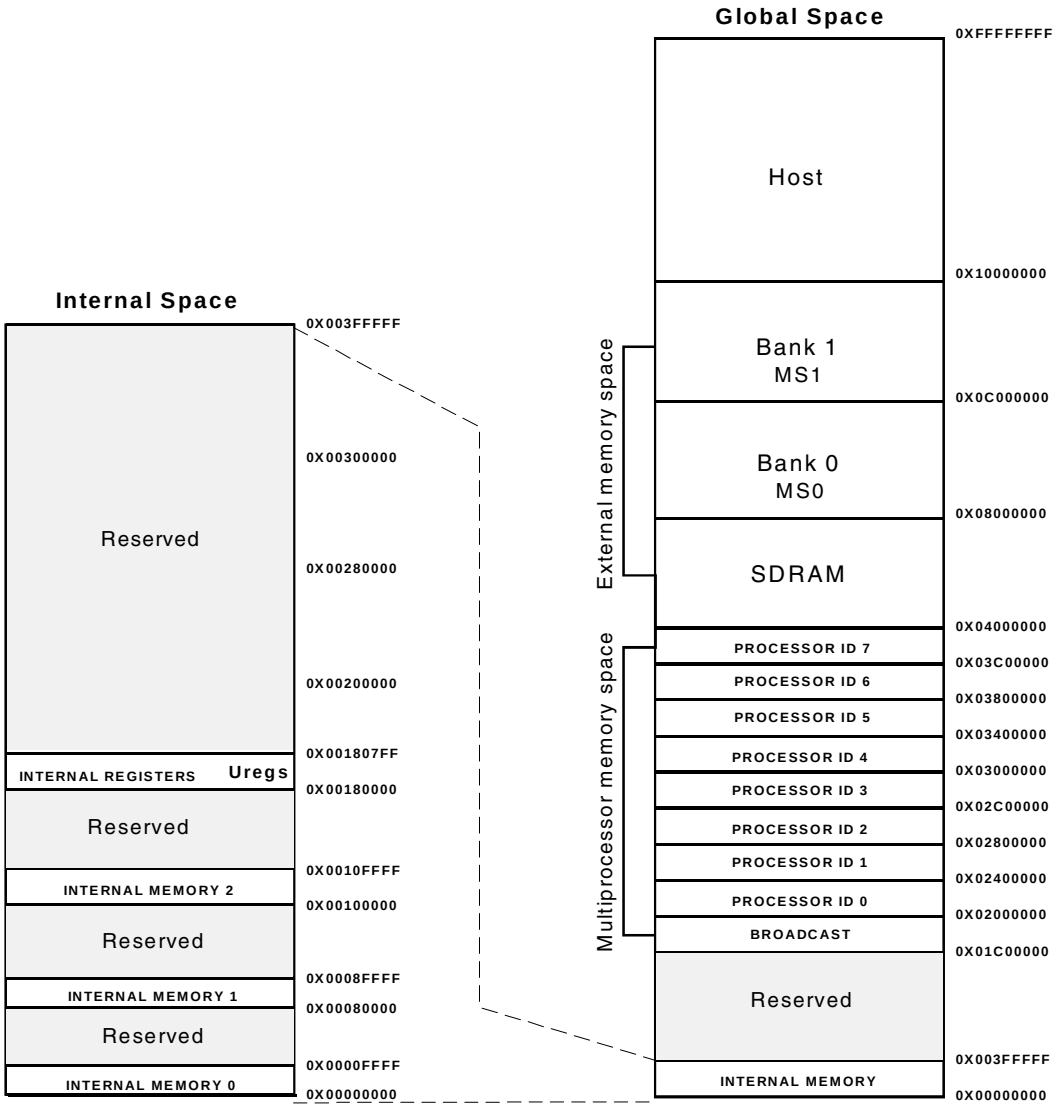


Figure 3 ADSP-TS001 Memory Map

Off-Chip Memory and Peripherals Interface

The ADSP-TS001's external port provides the DSP's interface to off-chip memory and peripherals. The 4-Gword address space is included in the DSP's unified address space. The separate on-chip buses—three 128-bit data buses and three 32-bit address buses—are multiplexed at the external port to create an external system bus with a single 64-bit data bus and a single 32-bit address bus.

When the system bus is configured for 64-bit operation, 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-TS001 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-TS001 provides an easy and configurable interface between its external bus and host processors. To accommodate a variety of host processors, the host interface supports pipelined or slow protocols for accesses of the host as slave. Each protocol has programmable transmission parameters, such as idle cycles, pipe depth, and internal wait cycles.

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

The host interface provides a deadlock recovery mechanism that enables a host to recover from deadlock situations involving the DSP. The $\overline{\text{BOFF}}$ signal provides the deadlock recovery mechanism. When the host asserts $\overline{\text{BOFF}}$, the DSP asserts $\overline{\text{HBG}}$ and relinquishes the external bus, even when buslock is enabled.

The host can directly read or write the internal memory of the ADSP-TS001, and it can access most of the DSP registers, including DMA control and TCB registers. Vector interrupts support efficient execution of host commands.

Multiprocessor Interface

The ADSP-TS001 offers powerful features tailored to multiprocessing DSP systems. The external port and link ports provide integrated, glueless multiprocessing support.

The external port supports a unified address space (see Figure 4, “ADSP-TS001 shared memory multiprocessing system,” on page 11) that enables direct interprocessor accesses of each ADSP-TS001’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-TS001s 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.

Maximum throughput for interprocessor data transfers is 600 Mbytes/second over the external port.

The DSP’s four link ports provide a second path for interprocessor communications.

SDRAM Interface

The SDRAM interface enables the ADSP-TS001 to transfer data to and from synchronous DRAM (SDRAM) at a throughput of one word per SCLK cycle.

The SDRAM interface provides a glueless interface with standard SDRAMs—16Mb, 64Mb, 128Mb, and 256Mb. The DSP supports directly a maximum of 64Mwords × 32b of SDRAM. The SDRAM interface is mapped in external memory in the DSP’s unified memory map.

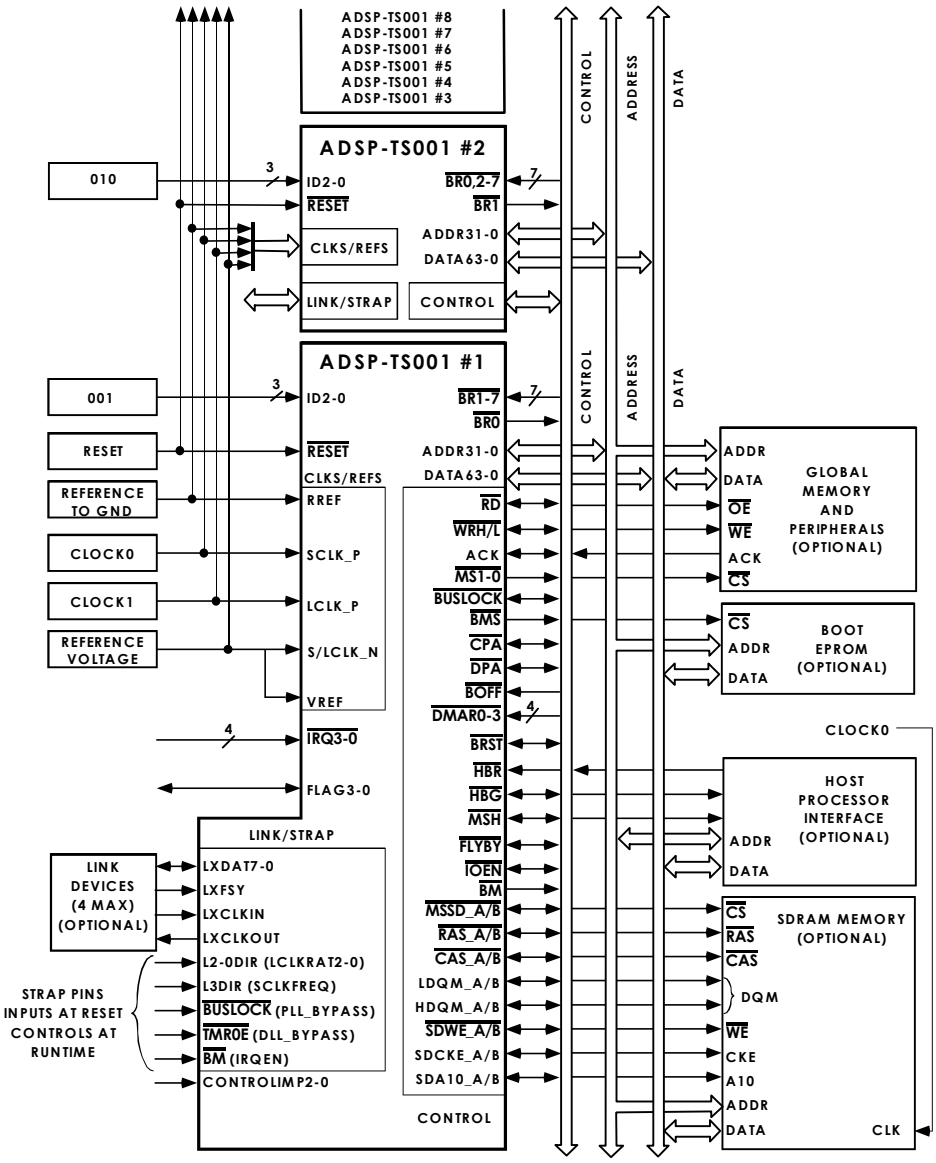


Figure 4 ADSP-TS001 shared memory multiprocessing system

EPROM Interface

The ADSP-TS001 can be configured to boot from external 8-bit EPROM at reset. An automatic process (which follows reset) loads a program from the EPROM into internal memory. This process uses sixteen wait cycles for each read access. During booting, the $\overline{\text{BMS}}$ pin functions 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 interface is not mapped in the DSP's unified memory map. It is a byte address space limited to a maximum of 16 Mbytes (twenty-four address bits).

DMA Controller

The ADSP-TS001's on-chip DMA controller, with fourteen 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 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 memory-mapped peripheral on the external bus. These transfers support master mode and handshake mode protocols.

- Link port transfers

  Eight dedicated DMA channels (four transmit and four receive) transfer quad-word data only between link ports and between a link port and internal or external memory. These transfers use handshake mode protocol only. 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 use slave mode protocol only, and an external bus master must initiate the transfer.

The DMA controller provides these additional features:

- Flyby transfers

  Flyby operations occur through the external port only (DMA channel 0) and do not involve the DSP's core. The DMA controller acts as a conduit to transfer data from one external device to another through external memory. During a transaction, the DSP relinquishes the external data bus and outputs addresses, memory selects ($\overline{MS}_{1-0}$), $\overline{FLYBY}$, $\overline{RD}/\overline{WR}$ 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 2-D 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

The DSP's four link ports provide additional eight-bit bidirectional I/O capability. With the ability to operate at a double data rate—latching data on both the rising and falling edges of the clock—running at 75 MHz, each link port can support up to 150 Mbytes/second, for a combined maximum throughput of 600 Mbytes/second.

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

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

Each link port has five signals that control its operation: LxCLKOUT and LxCLKIN implement clock/acknowledge handshaking. LxDIR indicates the direction of transfer and is used only when buffering the LxDAT signals; for example, using differential low-swing buffers for long twisted-pair wires. LxDAT provide the data bus 8-bit input/output. LxFSY provides frame synchronization to signal the completion of a block transfer.

The LxFSY signal supports Time Division Multiplex (TDM) applications. Frame synchronization enables applications to connect the link ports to devices that may not transmit bytes in multiples of quad-words, and to handle transfers in which the length of the data is unknown before transmission.

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

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

ADDITIONAL OPERATIONAL FEATURES

Timer and General-Purpose I/O

The ADSP-TS001 has a timer pin ($\overline{\text{TMROE}}$) that generates output when a programmed timer counter has expired and four programmable general-purpose I/O pins ($FLAG_{3-0}$) 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-TS001 has three levels of reset:

- Power-up reset—After power-up of the system and strap options are stable, the $\overline{\text{RESET}}$ pin must be asserted for at least 1 ms. after the power supply is stable.
- Normal reset—If the power and strap options were set more than 1 ms. before the reset, the $\overline{\text{RESET}}$ pin must be asserted for at least 100 SCLK cycles.
- Core reset—When setting the SQRST bit in SQCTL, the core is reset, but not the external port or I/O.

After reset, the ADSP-TS001 has four boot options for beginning operation:

- Boot from EPROM

  The DSP defaults to EPROM booting when the $\overline{\text{BMS}}$ pin is used as the strap option. When the $\overline{\text{BMS}}$ pin is not connected to V_{CC}, the mode is boot from EPROM.
- Boot by an external master (host or another ADSP-TS001)

Any master on the cluster bus can boot the ADSP-TS001 through various writes to its internal memory and registers.

- Boot by link port

  All four receive link DMA channels are initialized after reset to transfer a 256-word block to internal memory address 0 to 255, and to issue an interrupt at the end of the block (similar to EP DMA). The corresponding DMA interrupts are set to address zero (0).

- No boot—Start running from an external memory

  Using the 'no boot' option, the ADSP-TS001 must start running from an external memory, caused by asserting one of the $\overline{IRQ}$ interrupt signals.

The ADSP-TS001 core always exits to reset in the idle state and waits for an interrupt. Some of the interrupts in the IVT are initialized and enabled after reset.

Low-Power Operation

The ADSP-TS001 can enter a low-power sleep mode, in which its core does not execute instructions, reducing power consumption to a minimum. The ADSP-TS001 exits sleep mode when it senses a falling edge on any of its $\overline{IRQ}_{0-3}$ interrupt inputs. The interrupt, if enabled, causes the ADSP-TS001 to execute the corresponding interrupt service routine.

This feature is useful for systems that require a low-power standby mode.

Clock Domains

The ADSP-TS001 has two clock inputs that drive its two major clock domains:

- SCLK (system clock)

  Provides clock input for the external bus interface and defines the AC specification. The external bus interface runs at 1x the SCLK frequency. A PLL phase locks internal SCLK to SCLK input. The maximum SCLK frequency is ≤½ the internal DSP clock (CCLK) frequency.

- LCLK (local clock)

  Provides clock input to the internal clock driver, CCLK, which is internal clock for the core, internal buses, memory, and link ports. The instruction execution rate is equal to CCLK. A PLL from LCLK generates CCLK which is phase-locked. The LCLKRAT pins define the clock multiplication of LCLK to CCLK (see page 26). The link port clock is generated from CCLK via a software programmable divisor.

Connecting SCLK and LCLK to the same clock source and using an integer clock multiplication value provides predictable cycle-by-cycle operation, a requirement of fault-tolerant systems and some multiprocessing systems.

Power Supplies

The ADSP-TS001 has separate power supply connections for internal logic (V_{DD}), analog circuits (V_{DD_A}), reference resistor control (V_{DD_E}), and I/O buffer (V_{DD_IO}) power supplies. The internal (V_{DD}) and analog (V_{DD_A}) supplies must meet the 2.5V requirement. The reference resistor control (V_{DD_E}) and I/O buffer (V_{DD_IO}) supplies must meet the 3.3V requirement.

Note that the analog (V_{DD_A}) supply powers the clock generator PLL, and the reference resistor control (V_{DD_E}) powers the controlled impedance circuit. To produce a stable clock, you must provide an

external circuit to filter the power inputs to V_{DD_A} and V_{DD_E} as shown in Figure 5 below and Figure 6 on page 16.

The ideal power on sequence for the DSP is to provide power up of all supplies simultaneously. If there is going to be some delay between power up of the supplies, provide V_{DD_IO} first, then V_{DD}.

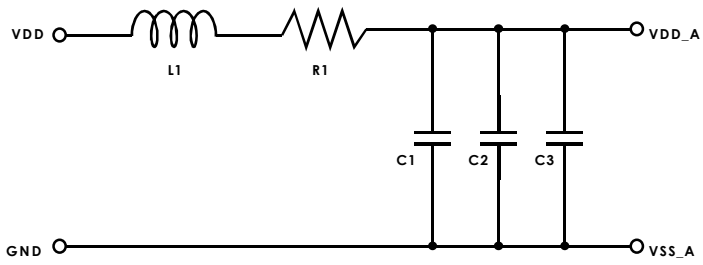
Filtering Circuits

V_{DD_A} Filter

This is the supply for the PLLs.

V_{DD_A} = 2.5V ±5%. The value need not necessarily be the same as VDD; however, it can be derived from V_{DD} by filtering.

Figure 5 shows a possible filtering scheme for V_{DD_A}.



L1: 10µH SERIES INDUCTOR
R1: 10Ω SERIES RESISTOR
C1: 10µF STORAGE CAPACITOR
C2: 100nF DECOUPLING CAPACITOR (SMD)
C3: 1nF DECOUPLING CAPACITOR (HF SMD) PLACED CLOSE TO DSP'S PINS

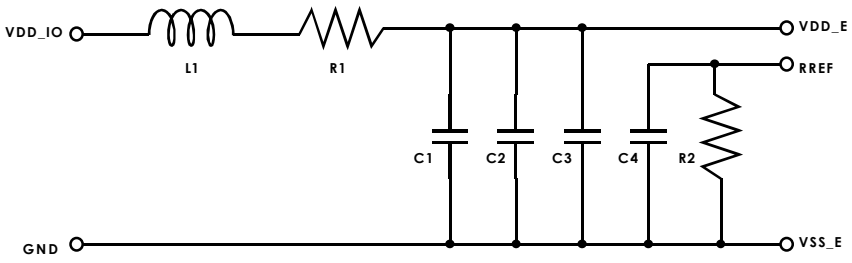
Figure 5 V_{DD_A} filtering scheme

V_{DD_E} Filter

This is the supply for the controlled impedance circuit.

V_{DD_E} = 3.3V ±5%. The value should be the same as the value of V_{DD_IO}. The user should derive V_{DD_E} by filtering V_{DD_IO}.

Figure 6 shows the filtering scheme for V_{DD_E}.



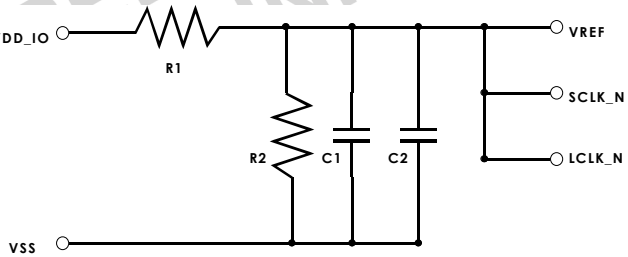
L1: 10µH SERIES INDUCTOR
R1: 10Ω SERIES RESISTOR
C1: 10µF STORAGE CAPACITOR
C2: 100nF DECOUPLING CAPACITOR (SMD)
C3: 1nF DECOUPLING CAPACITOR (HF SMD) PLACED CLOSE TO DSP'S PINS
C4: 1nF DECOUPLING CAPACITOR (HF SMD) PLACED CLOSE TO DSP'S PINS
R1: (5 × Z0), WHERE Z0 IS THE DESIRED IMPEDANCE OF ALL OUTPUT BUFFERS

Figure 6 V_{DD_E} filtering scheme

VREF, SCLK_N, & LCLK_N Filter

This circuit provides the reference voltage for the switching voltage, system clock, and local clock references.

Figure 7 shows the filtering scheme for VREF, SCLK_N, and LCLK_N.



R1: 2KΩ SERIES RESISTOR
R2: 1.67KΩ SERIES RESISTOR
C1: 1µF CAPACITOR (SMD)
C2: 1nF DECOUPLING CAPACITOR (HF SMD) PLACED CLOSE TO DSP'S PINS

Figure 7 VREF, SCLK_N, & LCLK_N filtering scheme

DEVELOPMENT TOOLS

The ADSP-TS001 is supported with a complete set of VisualDSP® software and hardware development tools, including the EZ-ICE® In-Circuit Emulator and development software.

Both the SHARC Development Tools family and the VisualDSP integrated project management and debugging environment support the ADSP-TS001. The VisualDSP project management environment enables you to develop and debug an application from within a single integrated program.

The SHARC Development Tools include an easy to use Assembler that is based on an algebraic syntax; an Assembly library/librarian; a linker; a loader; a cycle-accurate, instruction-level simulator; a C compiler; and a C run-time library that includes DSP and mathematical functions.

Debugging both C and Assembly programs with the VisualDSP debugger, you can:

- View mixed C and Assembly code.
- Insert break points.
- Set conditional breakpoints on registers, memory, and stacks.
- Trace instruction execution.
- Profile program execution.
- Fill and dump memory.
- Source level debugging.
- Create custom debugger windows.

The VisualDSP IDE lets you define and manage DSP software development. Its dialog boxes and property pages enable you to configure and manage all of the SHARC Development Tools, including the syntax highlighting in the VisualDSP editor. This capability lets you:

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

The EZ-ICE Emulator uses the IEEE 1149.1 JTAG test access port of the ADSP-TS001 processor to monitor and control the target board processor during emulation. The EZ-ICE provides full-speed emulation, allowing inspection and modification of memory, registers, and processor stacks. Non-intrusive 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 SHARC processor family. Hardware tools include SHARC PC plug-in cards, multiprocessor SHARC VME boards, and daughter and modules with multiple SHARCs' and additional memory. These modules are based on the SHARCPAC™ module specification. Third Party software tools include DSP libraries, real-time operating systems, and block diagram design tools.

ADDITIONAL INFORMATION

This data sheet provides a general overview of the ADSP-TS001's architecture and functionality. For detailed information on the ADSP-TS001's core architecture and instruction set, see the *TigerSHARC DSP Hardware Specification*. For detailed information on the development tools for this processor, see the *VisualDSP User's Guide and Reference for the ADSP-TS001TigerSHARC DSP*.

PIN FUNCTION DESCRIPTIONS

While most of the ADSP-TS001'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.

The output pins can be three-stated during normal operation. The DSP three-states all outputs during reset, allowing these pins to get to their internal pullup or pulldown state. Unused inputs may be left unconnected (floating). Some output pins (control signals) have a pullup or pulldown that maintain a known value during transitions between different drivers. Open drain outputs have a strong pullup (5KΩ) so they can operate as a wired-OR function.

Table 2. Pin definitions—clocks

Signal	Type	Description
LCLK_N	I	Local Clock Reference. Connect this pin to VREF as shown in Figure 7.
LCLK_P	I	Local Clock Input. DSP clock input. The instruction cycle rate = *n* x LCLK, where *n* is user-programmable to 2, 2.5, 3, 3.5, 4 or 5. LCLK cannot be halted, changed, or operated below the specified frequency.
SCLK_N	I	System Clock Reference. Connect this pin to VREF as shown in Figure 7.
SCLK_P	I	System Clock Input. The DSP's system input clock for cluster bus.
A = asynchronous; G = ground; I = input; O = output; o/d = open drain output; P = power supply; pd = internal pulldown 100 KΩ; pu = internal pullup 100 KΩ; S = synchronous; T = Three-State		

Table 3. Pin definitions—external port

Signal	Type	Description
$ADDR_{31-0}$	I/O/T	Address Bus. The DSP issues addresses for accessing memory and peripherals on these pins. In a multiprocessor system, the bus master outputs addresses for accessing internal memory or IOP registers of other ADSP-TS001s. The DSP inputs addresses when a host or another DSP accesses its internal memory or IOP registers.
$DATA_{63-0}$	I/O/T	External Data Bus. The DSP drives and receives data and instructions on these pins. Pull-up resistors on unused DATA pins are unnecessary.
$\overline{RD}$	I/O/T (pu)	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 Uregs. In a multiprocessor system, the bus master drives $\overline{RD}$. $\overline{RD}$ changes concurrently with ADDR pins.
$\overline{WRL}$	I/O/T (pu)	Write Low. $\overline{WRL}$ is asserted in two cases: When the ADSP-TS001 writes to an even address word of external memory or to another external bus agent; and when the ADSP-TS001 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.
A = asynchronous; G = ground; I = input; O = output; o/d = open drain output; P = power supply; pd = internal pulldown 100 KΩ; pu = internal pullup 100 KΩ; S = synchronous; T = Three-State		

Table 3. Pin definitions—external port (Cont'd)

Signal	Type	Description
$\overline{WRH}$	I/O/T (pu)	Write High. $\overline{WRH}$ is asserted when the ADSP-TS001 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.
$\overline{BMS}$	I/O/T (pu/pd)	Boot Memory Select. $\overline{BMS}$ is chip select for boot EPROM or flash memory. During reset, the DSP uses $\overline{BMS}$ as a strap pin for EPROM boot mode. When the DSP is configured to boot from EPROM, $\overline{BMS}$ is active during the boot sequence. Pulldown enabled during reset (asserted); pullup enabled after reset (deasserted). In a multiprocessor system, the DSP bus master drives $\overline{BMS}$. For details see "Reset and Booting" on page 13 and the EBOOT signal on page 26. At reset this is a strap pin, for more information, see Table 9.
$\overline{MS}_{1-0}$	O/T (pu)	Memory Select. $\overline{MS0}$ or $\overline{MS1}$ is asserted whenever the DSP accesses memory banks 0 or 1 respectively. $\overline{MS}_{1-0}$ are decoded memory address pins that change concurrently with ADDR pins: $\overline{MS}_0$: $ADDR_{31:26}$ = 0b000010 $\overline{MS}_1$: $ADDR_{31:26}$ = 0b000011 In multiprocessor systems, the master DSP drives $\overline{MS}_{1-0}$.
$\overline{MSH}$	O/T (pu)	Memory Select Host. $\overline{MSH}$ is asserted whenever the DSP accesses the host address space ($ADDR_{31:28} \neq$ 0b0000). $\overline{MSH}$ is a decoded memory address pin that changes concurrently with ADDR pins. In a multiprocessor system, the bus master DSP drives $\overline{MSH}$.
$\overline{HBR}$	I	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. When relinquishing the bus, the master DSP three-states the $ADDR_{31-0}$, $DATA_{63-0}$, $\overline{MSH}$, $\overline{MSSD}$, $\overline{MS}_{1-0}$, $\overline{RD}$, $\overline{WRL}$, $\overline{WRH}$, $\overline{BMS}$, $\overline{BRST}$, $\overline{FLYBY}$, $\overline{IOEN}$, $\overline{RAS}$, $\overline{CAS}$, $\overline{SDWE}$, SDCKE, LDQM and HDQM pins, and puts the SDRAM in self-refresh mode.
$\overline{HBG}$	I/O/T (pu)	Host Bus Grant. Acknowledges $\overline{HBR}$ and indicates that the host can take control of the external bus. 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.
ACK	I/O/T (pu)	Acknowledge. External devices can de-assert 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 de-assert ACK to add wait states to synchronous accesses of its internal memory.
$\overline{CPA}$	I/0/A (o/d)	Core Priority Access. Asserted while the DSP's core accesses external memory. This pin enables a slave DSP to interrupt a master DSP's background DMA transfers and gain control of the external bus for core-initiated transactions. $\overline{CPA}$ is an open drain output, connected to all DSPs in the system. The $\overline{CPA}$ pin has an internal 5 KΩ pullup resistor, which is only enabled on the DSP with ID=0. If not required in the system, leave $\overline{CPA}$ unconnected.

A = asynchronous; G = ground; I = input; O = output; o/d = open drain output; P = power supply;
pd = internal pulldown 100 KΩ; pu = internal pullup 100 KΩ; S = synchronous; T = Three-State

Table 3. Pin definitions—external port (Cont'd)

Signal	Type	Description
$\overline{DPA}$	I/0/A (o/d)	DMA Priority Access. Asserted while a high-priority DSP DMA channel accesses external memory. This pin enables a high-priority DMA channel on a slave DSP to interrupt transfers of a normal-priority DMA channel on a master DSP and gain control of the external bus for DMA-initiated transactions. $\overline{DPA}$ is an open drain output, connected to all DSPs in the system. The $\overline{DPA}$ pin has an internal 5 KΩ pullup resistor, which is only enabled on the DSP with ID=0. If not required in the system, leave $\overline{DPA}$ unconnected.
$\overline{DMAR}_{3-0}$	I/A	DMA Request Pins. Enable external I/O devices to request DMA services from the DSP. In response to $\overline{DMAR}$x, the DSP performs DMA transfers according to the DMA channel's initialization. The DSP ignores DMA requests from uninitialized channels. Input is edge-sensitive.
$\overline{FLYBY}$	O/T (pu)	$\overline{FLYBY}$. When a DSP DMA channel is initiated in $\overline{FLYBY}$ mode, it generates flyby transactions on the external bus. During flyby transactions, the DSP asserts $\overline{FLYBY}$, which signals the source or destination I/O device to latch the next data or strobe the current data, respectively, and to prepare for the next data on the next cycle.
$\overline{IOEN}$	O/T (pu)	I/O Device Output Enable. Enables the output buffers of an external I/O device for fly-by transactions between the device and external memory. Active on fly-by transactions.
ID_{2-0}	I (pd)	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 = $\overline{BR7}$ ID_{2-0} must have a constant value during system operation and can change during reset only.
$\overline{BR}_{7-0}$	I/O	Multiprocessing Bus Request Pins. Used by the DSPs in a multiprocessor system to arbitrate for bus mastership. Each DSP drives its own $\overline{BR}$x line (corresponding to the value of its ID_{2-0} inputs) and monitors all others. In systems with fewer than eight DSPs, set the unused $\overline{BR}$x pins high.
$\overline{BM}$	O	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 9.
$\overline{BRST}$	O/T (pu)	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.
$\overline{BOFF}$	I	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)	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 9.

A = asynchronous; G = ground; I = input; O = output; o/d = open drain output; P = power supply;
pd = internal pulldown 100 KΩ; pu = internal pullup 100 KΩ; S = synchronous; T = Three-State

Table 4. Pin definitions—JTAG

Signal	Type	Description
$\overline{EMU}$	O (o/d)	Emulation. Connected to the DSP's EZ-ICE target board connector only.
TCK	I	Test Clock (JTAG). Provides an asynchronous clock for JTAG boundary scan.
TDI	I (pu)	Test Data Input (JTAG). A serial data input of the boundary scan path. TDI has a 100 KΩ internal pullup resistor.
TDO	O	Test Data Output (JTAG). A serial data output of the boundary scan path.
TMS	I (pu)	Test Mode Select (JTAG). Used to control the test state machine. TMS has a 100 KΩ internal pullup resistor.
$\overline{TRST}$	I/A (pu)	Test Reset (JTAG). Resets the test state machine. $\overline{TRST}$ must be asserted after power up for proper JTAG operation. $\overline{TRST}$ has a 100 KΩ internal pullup resistor.

A = asynchronous; G = ground; I = input; O = output; o/d = open drain output; P = power supply;
pd = internal pulldown 100 KΩ; pu = internal pullup 100 KΩ; S = synchronous; T = Three-State

Table 5. Pin definitions—link ports

Signal	Type	Description
$L0DAT_{7-0}$	I/O	Link0 $Data_{7-0}$
$L1DAT_{7-0}$	I/O	Link1 $Data_{7-0}$
$L2DAT_{7-0}$	I/O	Link2 $Data_{7-0}$
$L3DAT_{7-0}$	I/O	Link3 $Data_{7-0}$
L0CLKOUT	O	Link0 Clock/Acknowledge Output
L1CLKOUT	O	Link1 Clock/Acknowledge Output
L2CLKOUT	O	Link2 Clock/Acknowledge Output
L3CLKOUT	O	Link3 Clock/Acknowledge Output
L0CLKIN	I	Link0 Clock/Acknowledge Input
L1CLKIN	I	Link1 Clock/Acknowledge Input
L2CLKIN	I	Link2 Clock/Acknowledge Input
L3CLKIN	I	Link3 Clock/Acknowledge Input

A = asynchronous; G = ground; I = input; O = output; o/d = open drain output; P = power supply;
pd = internal pulldown 100 KΩ; pu = internal pullup 100 KΩ; S = synchronous; T = Three-State

Table 5. Pin definitions—link ports (Cont'd)

Signal	Type	Description
L0DIR	O	Link0 Direction. 0 = input 1 = output At reset this is a strap pin, for more information, see Table 9.
L1DIR	O	Link1 Direction. 0 = input 1 = output At reset this is a strap pin, for more information, see Table 9.
L2DIR	O	Link2 Direction. 0 = input 1 = output At reset this is a strap pin, for more information, see Table 9.
L3DIR	O	Link3 Direction. 0 = input 1 = output At reset this is a strap pin, for more information, see Table 9.
L0FSY	I/O	Link0 Frame Synchronization. When asserted by the link master, LxFSY indicates completion of the frame.
L1FSY	I/O	Link1 Frame Synchronization. When asserted by the link master, LxFSY indicates completion of the frame.
L2FSY	I/O	Link2 Frame Synchronization. When asserted by the link master, LxFSY indicates completion of the frame.
L3FSY	I/O	Link3 Frame Synchronization. When asserted by the link master, LxFSY indicates completion of the frame.
A = asynchronous; G = ground; I = input; O = output; o/d = open drain output; P = power supply; pd = internal pulldown 100 KΩ; pu = internal pullup 100 KΩ; S = synchronous; T = Three-State		

Table 6. Pin definitions—power, ground, and reference

Signal	Type	Description
$CONTROLIMP_{2-0}$	I (pd)	Impedance Control. Every group of outputs has two controls: res_cont 0 = maximum drive pulse 1 = matched resistance pulse 0 = continuously matched impedance 1 = 2ns pulse for max resistance at the beginning of the transition For ADC (Address/Data/Controls) and LINK (all link port outputs) signals: CONTROLIMP ADC LINK res_cont pulse res_cont pulse 000 0 X 0 X 001 0 X 0 X 010 0 X 1 0 011 0 X 1 1 100 1 0 0 X 101 1 1 0 X 110 1 0 1 0 111 1 1 1 1
VDD	P	Vdd Pins for internal logic.
VDD_A[1]	P	Vdd Pins for analog circuits.
VDD_E[1]	P	VDD pin for reference resistor control.
VDD_IO	P	Vdd pins for I/O buffers.
VREF	I	Reference voltage defines the trip point for all input buffers (which are TTL-compatible). The value is 1.5V ± 100mv (which is the TTL trip point). VREF can be connected to a power supply or set by a voltage divider circuit. The voltage divider should have an HF decoupling capacitor (prepared 1nF HF SND) connected to VSS. Tie the decoupling capacitor between VREF input and VSS, as close to the DSP's pins as possible.
RREF	I	Reference resistor tied between RREF and VSS_E. For PCB, the value should be five times the value of matched resistance required for PCB traces. For example, if the target impedance is 50Ω, place a 250Ω resistor between RREF and VSS_E. Also, tie a decoupling capacitor (prepared 1nF HF SND) between RREF and VSS_E, as close to the DSP's pins as possible.
VSS	G	Ground Pins.
VSS_A	G	Ground Pins for analog circuits.
VSS_E	P	VSS pin for reference resistor control.

A = asynchronous; G = ground; I = input; O = output; o/d = open drain output; P = power supply;
pd = internal pulldown 100 KΩ; pu = internal pullup 100 KΩ; S = synchronous; T = Three-State

1. For details on deriving filtering circuits with the V_{DD_A} and V_{DD_E} pins, see "Filtering Circuits" on page 15.

Table 7. Pin definitions—flag I/O, IRQ, reset, and timer

Signal	Type	Description
$FLAG_{3-0}$	I/O/A (pd)	FLAG pins. Bidirectional input/output pins. Each pin can be configured individually for input or for output. $FLAG_{3-0}$ are inputs after power-up and reset. These pins are held at zero (0) value by a static 100KΩ pulldown resistor.
$\overline{IRQ}_{3-0}$	I/A (pu)	Interrupt Request. When asserted, the DSP generates an interrupt. Each of the $\overline{IRQ}_{3-0}$ pins can be independently set for edge-triggered or level-sensitive operation. After reset, these pins are disabled unless the $\overline{IRQ}$ strap option and interrupt vectors are initialized for booting.
$\overline{RESET}$	I/A	Reset. Sets the DSP to a known state and causes program fetch to start at an address specified by the reset vector register. $\overline{RESET}$ must be asserted a specified time according to the type of reset operation. For details, see "Reset and Booting" on page 13.
$\overline{TMR0E}$	O	Timer 0 expires. This output pulses for four cycles whenever timer 0 expires. At reset this is a strap pin, for more information, see Table 9.

A = asynchronous; G = ground; I = input; O = output; o/d = open drain output; P = power supply;
pd = internal pulldown 100 KΩ; pu = internal pullup 100 KΩ; S = synchronous; T = Three-State

Table 8. Pin definitions—SDRAM

Signal	Type	Description
$\overline{CAS}$_A	I/O/T (pu)	Column Address Select A. When sampled low at rising edge of the clock, $\overline{CAS}$ 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.
$\overline{CAS}$_B	O/T (pu)	Column Address Select B. Duplication of $\overline{CAS}$_A for drive strength, output only.
HDQM_A	O/T (pd)	High Word SDRAM Data Mask A. When sampled high, three-states the SDRAM DQ buffers. HDQM_A is valid on SDRAM transactions when $\overline{CAS}$ is asserted, and inactive on read transactions. On write transactions, HDQM_A 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..
HDQM_B	O/T (pd)	High Word SDRAM Data Mask B. Duplication of HDQM_A for drive strength, output only.
LDQM_A	O/T (pd)	Low Word SDRAM Data Mask A. When sampled high, three-states the SDRAM DQ buffers. LDQM_A is valid on SDRAM transactions when $\overline{CAS}$ is asserted, and inactive on read transactions. On write transactions, LDQM_A is active when accessing an odd address word on a 64-bit memory bus to disable the write of the low word.
LDQM_B	O/T (pd)	Low Word SDRAM Data Mask B. Duplication of LDQM_A for drive strength, output only.

A = asynchronous; G = ground; I = input; O = output; o/d = open drain output; P = power supply;
pd = internal pulldown 100 KΩ; pu = internal pullup 100 KΩ; S = synchronous; T = Three-State

Table 8. Pin definitions—SDRAM (Cont'd)

Signal	Type	Description
$\overline{\text{MSSD}}$_A	I/O/T (pu)	Memory Select SDRAM A. $\overline{\text{MSSD}}$ is asserted whenever the DSP accesses SDRAM memory space. $\overline{\text{MSSD}}$ is a decoded memory address pin that is asserted whenever the DSP issues an SDRAM command cycle (access to $ADDR_{31:26}$ = 0b000001). In a multiprocessor system, the master DSP drives $\overline{\text{MSSD}}$.
$\overline{\text{MSSD}}$_B	O/T (pu)	High Word SDRAM Data Mask B. Duplication of $\overline{\text{MSSD}}$_A for drive strength, output only.
$\overline{\text{RAS}}$_A	I/O/T (pu)	Row Address Select A. When sampled low at the rising edge of the clock, $\overline{\text{RAS}}$ 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.
$\overline{\text{RAS}}$_B	O/T (pu)	Row Address Select B. Duplication of $\overline{\text{RAS}}$_A for drive strength, output only.
SDA10_A	O/T (pu)	SDRAM A10_A. SDRAM Address bit 10 pin. Separate A10 signals enable SDRAM refresh operation while the DSP executes non-SDRAM transactions.
SDA10_B	O/T (pu)	SDRAM A10_B. Duplication of SDA10_A for drive strength, output only.
SDCKE_A	I/O/T (pu/pd)	SDRAM Clock Enable A. Activates the SDRAM clock for SDRAM self-refresh or suspend modes. A slave DSP in a multiprocessor system does not have the pullup or pulldown. A master DSP (or ID=0 in a single processor system) has a 100 KΩ pullup before granting the bus to the host, except when the SDRAM is put in self refresh mode. In self refresh mode, the master has a 100 KΩ pulldown before granting the bus to the host.
SDCKE_B	O/T (pu/pd)	SDRAM Clock Enable B. Duplication of SDCKE_A for drive strength, output only. A slave DSP in a multiprocessor system does not have the pullup or pulldown. A master DSP (or ID=0 in a single processor system) has a 100 KΩ pullup before granting the bus to the host, except when the SDRAM is put in self refresh mode. In self refresh mode, the master has a 100 KΩ pulldown before granting the bus to the host.
$\overline{\text{SDWE}}$_A	I/O/T (pu)	SDRAM Write Enable A. When sampled low at rising edge of the clock and $\overline{\text{CAS}}$ is active, $\overline{\text{SDWE}}$_A indicates an SDRAM write access. When sampled high at rising edge of the clock and $\overline{\text{CAS}}$ is active, $\overline{\text{SDWE}}$_A indicates an SDRAM read access. In other SDRAM accesses, $\overline{\text{SDWE}}$_A defines the type of operation to execute according to SDRAM specification.
$\overline{\text{SDWE}}$_B	O/T (pu)	SDRAM Write Enable B. Duplication of $\overline{\text{SDWE}}$_A for drive strength, output only.

A = asynchronous; G = ground; I = input; O = output; o/d = open drain output; P = power supply;
pd = internal pulldown 100 KΩ; pu = internal pullup 100 KΩ; S = synchronous; T = Three-State

Some pins have another function at reset. Strap options set DSP operating modes. During reset, the DSP samples the strap option pins. Strap pins have a 100 KΩ pull-down for the default value. If a strap pin is not connected to an external pull-up or logic load, the DSP samples the default value during reset. To set a mode other than the default mode, connect the strap pin to a 10 KΩ external pull-up. Table 9 lists and describes each of the DSP's strap pins. If strap pins are used as drivers to logic inputs, a 4.7 KΩ pulldown or pullup is required.

Table 9. I/O Strap Pins

Signal	On Pin…	Description
DLL_BYPASS	$\overline{\text{TMR0E}}$	Timer 0 Output Bypass. 0 = DLL works normally (default) 1 = If set during reset, SCLK input bypasses the DLL and connects directly to the DSP's internal I/O clock
EBOOT	$\overline{\text{BMS}}$	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 In systems that do not boot from an EPROM, short $\overline{\text{BMS}}$ to VDD_IO or (to retain access to flash memory) connect $\overline{\text{BMS}}$ to VDD_IO with a pullup resistor.
IRQEN	$\overline{\text{BM}}$	Interrupt Enable. 0 = disable and set $\overline{\text{IRQ}}$ interrupts to level-sensitive after reset (default) 1 = enable and set $\overline{\text{IRQ}}$ interrupts to edge-sensitive immediately after reset
$\text{LCLKRAT}_{2\text{-}0}$	L0DIR L1DIR L2DIR	LCLK Ratio. The DSP's core clock (instruction cycle rate) = $n \times LCLK$, where n is user-programmable to 2, 2.5, 3, 3.5, 4, or 5. Ratio LCLKRAT2 LCLKRAT1 LCLKRAT0 2 0 0 0 2.5 0 0 1 3 0 1 0 3.5 0 1 1 4 1 0 0 5 1 0 1
PLL_BYPASS	$\overline{\text{BUSLOCK}}$	Bus Lock Bypass. 0 = PLL works normally (default) 1 = If set during rest, LCLK input bypasses the PLL and connects directly to the DSP's internal core clock
SCLKFREQ	L3DIR	SCLK Frequency. Indicates to the SCLK deskew PLL the SCLK frequency range. 0 = 50 to 100 MHz (default) 1 = <50 MHz

DESIGNING AN EZ-ICE COMPATIBLE SYSTEM

This application note describes how to design the interface between an Analog Devices JTAG DSP and the emulation header on a custom DSP target board. The environment described uses an Analog Devices (ADI) SHARC DSP as an example. However, this information is applicable for all of ADI's JTAG DSPs.

The White Mountain DSP-Analog Devices family of emulators are tools that every DSP developer needs to test and debug their hardware and software system. ADI has supplied an IEEE 1149.1 JTAG Test Access Port (TAP) on each JTAG DSP. The emulator uses the TAP to access the internals of the DSP, allowing the developer to load code, set breakpoints, observe variables, observe memory, examine registers, etc. The DSP must be halted to send data and commands, but once an operation is completed by the emulator, the DSP system is set running at full speed with no impact on system timing.

The last thing an engineer wants to do when testing a new hardware design is debug the emulation port of their target board. Since the emulation port in many cases is the only vehicle to test a new target board, it needs to work flawlessly. The guidelines set forth in this application note are designed to help eliminate possible JTAG emulation port problems.

Target Board Connector

The emulator interface to an ADI JTAG DSP is a 14-pin header, as shown in Figure 8. The customer must supply this header on their target board in order to communicate with the emulator. The interface consists of a standard dual row 0.025" square post header, set on 0.1" x 0.1" spacing, with a minimum post length of 0.235". Pin 3 is the key position used to prevent the pod from being inserted backwards. This pin must be clipped on the target board.

Also, the clearance (length, width, and height) around the header must be considered. Leave a clearance of at least 0.15” and 0.10” around the length and width of the header, and reserve a height clearance to attach and detach the pod connector. See “White Mountain DSP Pod Designs” on page 31 for detailed drawings of the pod connector.

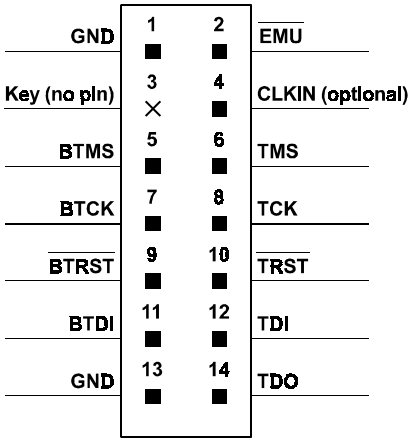


Figure 8 Emulator Interface for ADI JTAG DSPs

As shown in Figure 8, the interface has two sets of signals on the header. The standard JTAG signals TMS, TCK, TDI, TDO, $\overline{\text{TRST}}$, $\overline{\text{EMU}}$, and SCLK_P are used for emulation purposes (via an emulator). Secondary JTAG signals BTMS, BTCK, $\overline{\text{BTRST}}$, and BTDI are used for optional board-level (boundary scan) testing. These "B" signals would connect to a separate on-board JTAG

boundary scan controller if used. Most customers will never use the "B" signals. If these signals are not used, tie all of them to ground, except BTCK. Pull BTCK up to V_{DD_IO} (V_{CC} in Figure 9) using a 4.7KΩ resistor, as shown in Figure 9.

When the emulator is not connected to this header, place jumpers across BTMS, BTCK, $\overline{\text{BTRST}}$, and BTDI as shown in Figure 9. This holds the JTAG signals in the correct state to allow the DSP to run freely. Remove all the jumpers when connecting the emulator to the JTAG header.

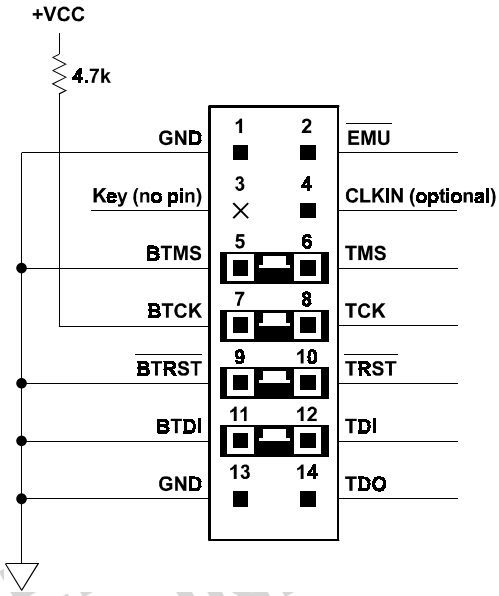


Figure 9 JTAG target board connector with no local boundary scan (V_{CC}=V_{DD_IO})

Table 10 lists the state of each standard JTAG signal.

Table 10. State of Standard JTAG Signals

Signal	Description	Emulator	DSP
TMS	Test Mode Select	O	I
TCK	Test Clock (10 MHz)	O	I
$\overline{\text{TRST}}$	Test Reset	O	I
TDI	Test Data In	O	I
TDO	Test Data Out	I	O
$\overline{\text{EMU}}$	Emulation Pin	I	I/O (Open Drain)
CLKIN (Optional)	DSP Clock Input	I	I
I = input; I/O = input/output; O = output			

The CLKIN signal is the clock signal line that connects to all DSPs (typically 30 MHz or greater). For synchronous DSP operation to work correctly, the CLKIN signal on all DSPs must be the same signal and the skew between them must be minimal—see the *TigerSHARC DSP Hardware Specification* for more details on CLKIN.

Note that the CLKIN signal presently is not used by the emulator and can cause noise problems if connected to the JTAG header. If you are experiencing noise from this signal, clip this pin on the 14-pin JTAG header, or, since the CLKIN signal is not used by the emulator, simply tie the CLKIN pin to ground on the 14-pin JTAG header (do not connect it to the DSP's CLKIN signal).

The final connections between a single DSP target and the emulation header (within 6 inches) are shown in Figure 10. A 4.7KΩ pull-up resistor has been added on TCK, TDI, and TMS for increased noise resistance.

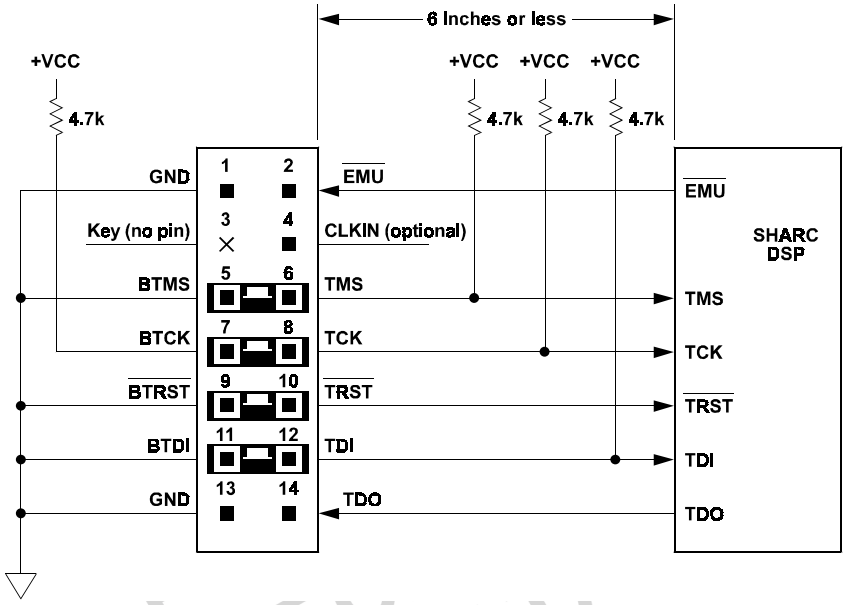


Figure 10 Single DSP Connection to JTAG Header (V_{CC}=V_{DD_IO})

Should your design use more than one DSP (or other JTAG device in the scan chain), or if your JTAG header is more than 6 inches from the DSP, use a buffered connection scheme as shown in Figure 11 (no local boundary scan mode shown). Using a buffer that has built in series resistors such as the 74ABT2244 family can help reduce ringing on the JTAG signal lines. For low voltage applications (3.3V), the 74ALVT, and 74AVC logic families are a good starting point. Also, note the position of the pull-up resistor on $\overline{\text{EMU}}$. This is required since the $\overline{\text{EMU}}$ line is an open drain signal.

Important: If you have more than one DSP (or JTAG device) on your target (in the scan chain), it is imperative that you buffer the JTAG header. Buffering keeps the signals clean and avoids noise problems that occur with longer signal traces.

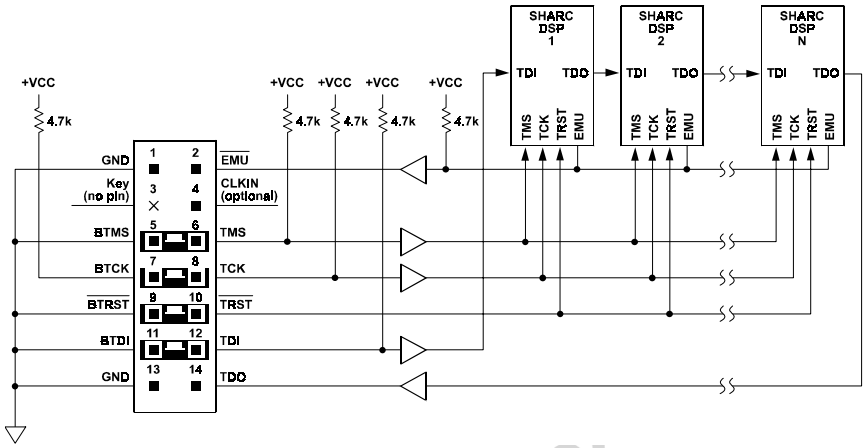


Figure 11 Multiple DSP Connection to JTAG Header (V_{CC}=V_{DD_IO})

Although the theoretical number of devices that can be supported (by the software) in one JTAG scan chain is quite large (50 devices or more), we recommended that you use no more than eight physical devices in one scan chain. (A physical device could however contain many JTAG devices such as inside a multichip module). This recommendation is due mostly to the transmission line effects that appear in long signal traces and is based on some field-collected empirical data. For large numbers of physical devices, the best approach is to break the chain into several smaller independent chains, each with their own JTAG header and buffer. If this is not possible, at least add some jumpers that can reduce the number of devices in one chain for debug purposes, and pay special attention in the layout stage for transmission line effects.

Layout Requirements

All JTAG signals (TCK, TMS, TDI, TDO, $\overline{\text{EMU}}$, $\overline{\text{TRST}}$, CLKIN) should be treated as critical route signals. This means paying special attention when you route these signals. Specify a controlled impedance requirement for each route (value depends on your circuit board—typically 50-75Ω). Keep crosstalk and inductance to a minimum on these lines by using a good ground plane and by routing away from other high noise signals, such as clock lines. Keep these routes as short and clean as possible, and keep the bused signals (TMS, TCK, $\overline{\text{TRST}}$, $\overline{\text{EMU}}$) as close to the same length as possible.

Power Sequence

The power-on sequence for your target and emulation system is as follows:

1. Apply power to the emulator first, then to the target board.

   This ensures that the JTAG signals are in the correct state for the DSP to run free.

   Upon power-on, the emulator drives the $\overline{\text{TRST}}$ signal low, keeping the DSP TAP in the test-logic-reset state, until the emulation software takes control.

2. Removal of power should be the reverse, so turn off power to the target board then to the emulator.

White Mountain DSP Pod Designs

This section describes emulator pod designs from White Mountain DSP:

- JTAG pod connector
- 3.3 pod logic

JTAG Pod Connector

White Mountain DSP emulator pod is a device that connects directly to the DSP target board 14-pin JTAG header. Check our web site for updates to this section for new emulator design details.

Figure 12 shows the Mountain ICE, Summit-ICE, Trek-ICE, Mountain-ICE/WS, Apex-ICE JTAG pod connector.

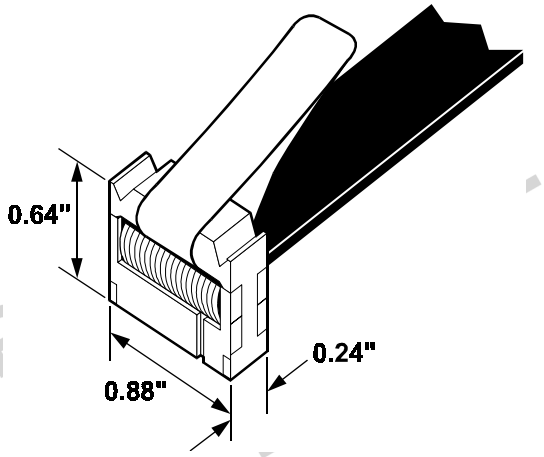


Figure 12 Dimensions of the JTAG pod connector at the 14-pin target end

Figure 13 shows the keep-out area that enables the pod connector to properly seat onto the target board header. This board area should contain no components (chips, resistors, capacitors, etc.). The dimensions are referenced to the center of the 0.25” square post pin.

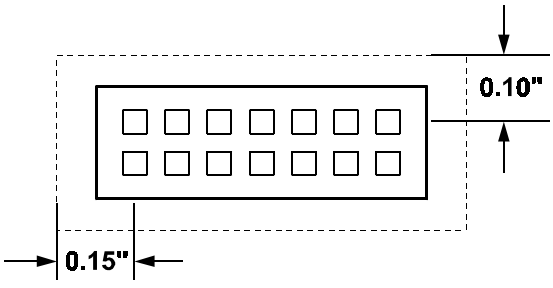


Figure 13 Keep-out area for a target board header

3.3V Pod Logic

A portion of the White Mountain DSP 3.3V emulator pod interface is shown in Figure 14. This figure describes the driver circuitry of the emulator pod. As can be seen, TMS, TCK, $\overline{\text{TRST}}$, and TDI are driven with a 10Ω series resistor. TDO and CLKIN are terminated with an optional 91/120Ω parallel terminator. $\overline{\text{EMU}}$ is pulled up with a 4.7KΩ resistor. The 74LVT244 chip drives the signals at 3.3V, with a maximum current rating of ±32mA.

Figure 14 shows the Mountain ICE, Summit-ICE, Trek-ICE, Mountain-ICE/WS, Apex-ICE 3.3V JTAG pod driver logic.

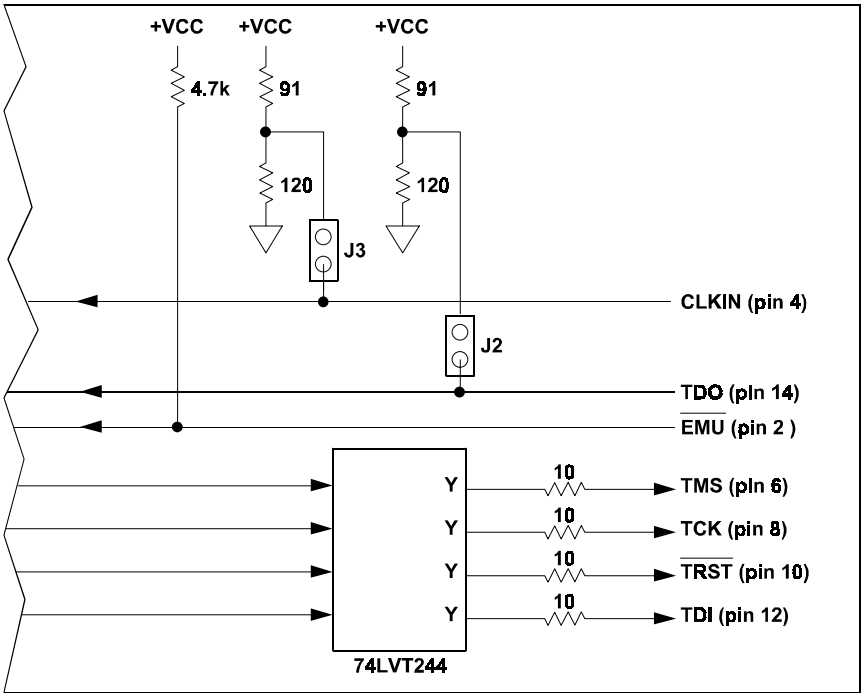


Figure 14 3.3V JTAG pod driver logic (V_{CC}=V_{DD_IO})

If needed, you can parallel terminate the TMS, TCK, $\overline{\text{TRST}}$, and TDI lines locally on your target board since these signals are driven by the pod with sufficient current drive (±32mA). To use the terminators on the TDO and CLKIN lines, you must have a buffer on your target board JTAG header and CLKIN signal. The DSP is incapable of driving the parallel terminator load directly with TDO. Assuming you have the proper buffers, to use the optional parallel terminators, just place a jumper on J2 or J3.

ADSP-TS001—SPECIFICATIONS

Note that component specifications are subject to change without notice.

Table 11. Recommended operating conditions

Parameter	Test Conditions	K Grade		Unit
		Min	Max	
V_{DD} Internal Supply Voltage		2.375	2.625	V
V_{DD_A} Analog Supply Voltage[1]		2.375	2.625	V
V_{DD_IO} I/O Supply Voltage		3.15	3.45	V
V_{DD_E} Reference Resistor Supply Voltage[1]		3.15	3.45	V
T_{CASE} Case Operating Temperature		0	+85	°C
V_{IH} High-Level Input Voltage[2]	@ V_{DD} = max	2	V_{DD} + 0.5	V
V_{IL} Low-Level Input Voltage[2]	@ V_{DD} = min	−0.5	0.8	V
V_{REF} Voltage reference		1.4	1.6	V

1. For details on deriving filtering circuits with the V_{DD_A} and V_{DD_E} pins, see "Filtering Circuits" on page 15.
2. Applies to input and bidirectional pins: ACK, $ADDR_{31-0}$, $\overline{BOFF}$, $\overline{BR}_{7-0}$, $\overline{CAS}\_A$, $CONTROLIMP_{2-0}$, $\overline{CPA}$, $DATA_{63-0}$, $\overline{DMAR}_{3-0}$, $\overline{DPA}$, $FLAG_{3-0}$, $\overline{HBG}$, ID_{2-0}, $\overline{IRQ}_{3-0}$, $L0DAT_{7-0}$, $L1DAT_{7-0}$, $L2DAT_{7-0}$, L0CLKOUT, L1CLKOUT, L2CLKOUT, L3CLKOUT, L0CLKIN, L1CLKIN, L2CLKIN, L3CLKIN, L0FSY, L1FSY, L2FSY, L3FSY, LCLK_P, LCLK_N, $\overline{MSSD}\_A$, $\overline{RAS}\_A$, $\overline{RD}$, $\overline{RESET}$, SCLK_P, SCLK_IN, $\overline{SDWE}\_A$, TDI, TMS, TCK, $\overline{TRST}$, $\overline{WRL}$, $\overline{WRH}$.

Table 12 ABSOLUTE MAXIMUM RATINGS[1]

Parameter	Absolute Maximum Rating
Internal (Core) Supply Voltage (V_{DD})	-0.3 V to +3.0 V
Reference Resistor Supply Voltage (V_{DD_E})	-0.3 V to +4.6 V
Analog (PLL) Supply Voltage (V_{DD_A})	-0.3 V to +3.0 V
External (I/O) Supply Voltage (V_{DD_IO})	-0.3 V to +4.6 V
Input Voltage	-0.5 V to V_{DD_IO} + 0.5 V
Output Voltage Swing	-0.5 V to V_{DD_IO} + 0.5 V
Load Capacitance	200 pF
Storage Temperature Range	-65°C to +150°C
Lead Temperature (5 seconds)	TBD °C

1. Stresses greater than those listed above may cause permanent damage to the device. These are stress ratings only, and 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 13. Electrical characteristics

Parameter	Test Conditions	K Grades Min	Max	Units
V_{OH} High-Level Output Voltage[1]	@ V_{DD} = min, I_{OH} = -2 uA	2.4		V
V_{OL} Low-Level Output Voltage[1]	@ V_{DD} = min, I_{OL} = 4 uA		0.4	V
I_{IH} High-Level Input Current[2, 3]	@ V_{DD} = max, V_{IN} = V_{DD} max		10	uA
I_{IL} Low-Level Input Current[2]	@ V_{DD} = max, V_{IN} = 0V[6]		10	uA
I_{ILP}Low-Level Input Current[3]	@ V_{DD} = max, V_{IN} = 0V[6]			A
I_{OZH} Three-State Leakage Current[4, 5, 7]	@ V_{DD} = max, V_{IN} = V_{DD} max		10	uA
I_{OZL} Three-State Leakage Current [4, 6]	@ V_{DD} = max, V_{IN} = 0V		10	uA
I_{OZHP}Three-State Leakage Current[6]	@ V_{DD} = max, V_{IN} = V_{DD} max			A
I_{OZLO}Three-State Leakage Current[7]	@ V_{DD} = max, V_{IN} = 0V			A
I_{OZLS}Three-State Leakage Current[5]	@ V_{DD} = max, V_{IN} = 0V			A
C_{IN}Input Capacitance[8, 9]	fIN = 1MHz, T_{CASE} = 25C, V_{IN} = 2.5V			F

1. Applies to output and bidirectional pins: ACK, $ADDR_{31-0}$, $\overline{BM}$, $\overline{BMS}$, $\overline{BUSLOCK}$, $\overline{BR}_{7-0}$, $\overline{BRST}$, $\overline{CAS}$_A, $\overline{CAS}$_B, $DATA_{63-0}$, $\overline{DP}$, $\overline{EMU}$, $FLAG_{3-0}$, $\overline{FLYBY}$, $\overline{HBG}$, IHDQM_A, HDQM_B, IOEN, $L0DAT_{7-0}$, $L1DAT_{7-0}$, $L2DAT_{7-0}$, L0CLKOUT, L1CLKOUT, L2CLKOUT, L3CLKOUT, L0CLKIN, L1CLKIN, L2CLKIN, L3CLKIN, L0DIR, L1DIR, L2DIR, L3DIR, L0FSY, L1FSY, L2FSY, L3FSY, LDQM_A, LDQM_B, $\overline{MS}_{1-0}$, $\overline{MSSD}$_A, $\overline{RAS}$_A, $\overline{RAS}$_B, $\overline{RD}$, SDA10_A, SDA10_B, SDCKE_A, SDCKE_B, $\overline{SDWE}$_A, $\overline{SDWE}$_B, TDO, $\overline{TMROE}$, $\overline{WRL}$, $\overline{WRH}$.
2. Applies to input pins: $\overline{BOFF}$, CONTROLIMP$_{2-0}$, $\overline{DMAR}_{3-0}$, $\overline{HBR}$, ID_{2-0}, LCLK_P, LCLK_N, $\overline{RESET}$, SCLK_P, SCLK_N, TCK.
3. Applies to input pins with internal pullups: $\overline{IRQ}_{3-0}$, TDI, TMS, $\overline{TRST}$.
4. Applies to three-stateable pins: $ADDR_{31-0}$, $\overline{BUSLOCK}$, $DATA_{63-0}$, SDA10_A, SDA10_B.
5. Applies to three-stateable pins with internal pullups: ACK, $\overline{BRST}$, $\overline{BMS}$, $\overline{CAS}$_A, $\overline{CAS}$_B, $\overline{FLYBY}$, $\overline{HBG}$, $\overline{IOEN}$, $\overline{MS}_{1-0}$, $\overline{MSH}$, $\overline{MSSD}$_A, $\overline{MSSD}$_B, $\overline{RAS}$_A, $\overline{RAS}$_B, $\overline{RD}$, $\overline{SDWE}$_A, $\overline{SDWE}$_B, $\overline{WRL}$, $\overline{WRH}$.
6. Applies to three-stateable pins with internal pulldowns: $FLAG_{3-0}$, HDQM_A, HDQM_B, LDQM_A., LDQM_B, SDCKE_A, SDCKE_B.
7. Applies to open drain pins with 5KΩ pull ups: $\overline{CPA}$, $\overline{DPA}$.
8. Applies to all signals.
9. Guaranteed but not tested.

ESD Sensitivity

CAUTION: ESD (electrostatic discharge) sensitive device. Electrostatic charges as high as 4000V readily accumulate on the human body and test equipment and can discharge without detection. Although the ADSP-TS001 features proprietary ESD protection circuitry, permanent damage may occur on devices subjected to high-energy electrostatic discharges. Therefore, proper ESD precautions are recommended to avoid performance degradation or loss of functionality.

Pages 35 through 47 of this document cover timing and pinout for the ADSP-TS001. These pages are only available to registered TigerSHARC customers. Please contact DSP product support at:

dsp.support@analog.com
(with the subject: TigerSHARC Timing Info)

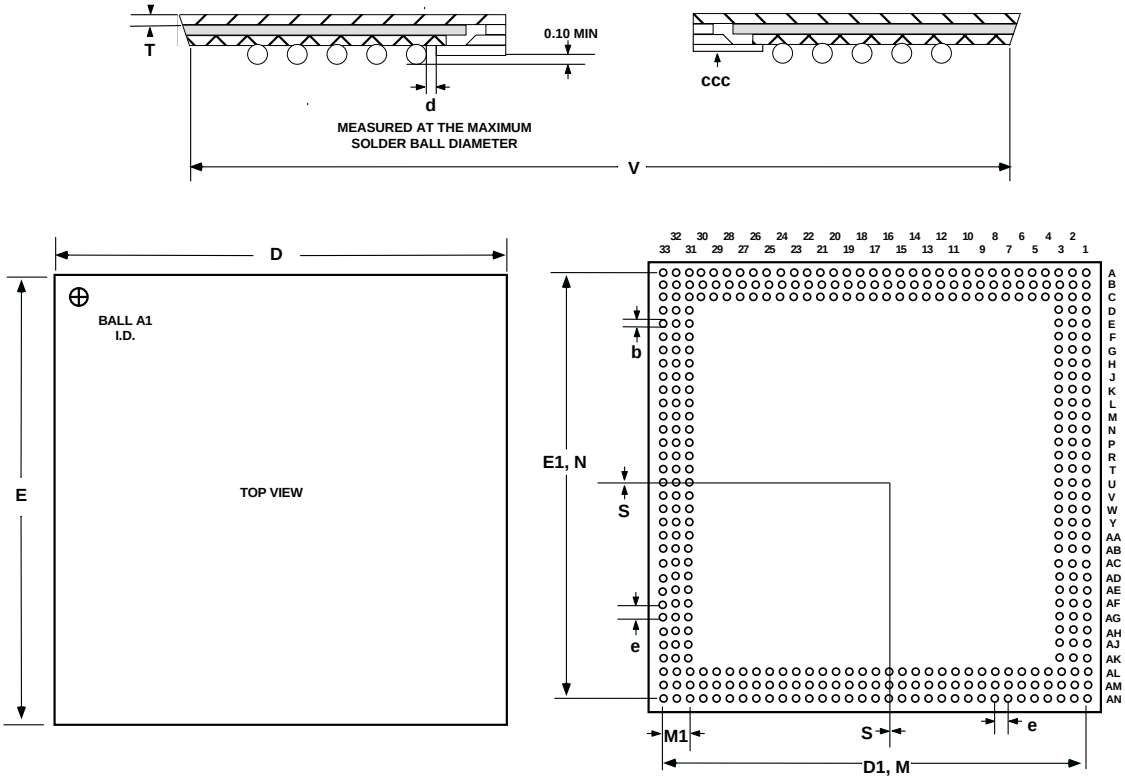
to request timing and pinout information.

For other TigerSHARC documentation, register through the Analog Devices DSP website at:

www.analog.com/dsp

OUTLINE DIMENSIONS—360 BALL SBGA PACKAGE

The package is 35mm x 35mm, 360 Ball SBGA with 3 rows of balls. Dimensions are in millimeters.



Body Size	35.0mm x 35.0mm Package			Body Size
Symbol	MIN.	NOM.	MAX.	Note
A			1.60	overall thickness
A1	0.40	0.50	0.60	ball height
A2	0.80	0.91	1.00	body thickness
D		35.00		body size
D1	32.90	33.00	33.10	ball footprint
E		35.00		body size
E1	32.90	33.00	33.10	ball footprint
M,N	33 x 33			ball matrix
M1	3			depth in rows
b	0.50	0.63	0.70	ball diameter
d	0.50			min distance encap to balls
e	1.00			ball pitch
ccc			0.20	encap flatness over die
ddd			0.20	coplanarity
S		0.00		solder ball placement
T	0.050	0.125	0.175	v-score web thickness
V	34.20		34.90	v-score bottom size

Ordering Guide

Table 19.

Part Number	Case Temperature	Instruction Rate (core clock)	On-chip SRAM	Operating Voltage
ADSP-TS001	0° C to 85° C	150 MHz	6Mbit	2.5 INT/3.3 EXT V