ANALOG DEVICES

Concurrent Signal Processor

ADSP-21csp01

FEATURES

PERFORMANCE

- 20 ns Instruction Cycle Time from 25 MHz Crystal @ 5.0 V
- 50 MIPS Sustained Performance
- 24-Bit Address Bus with a Unified 16M Address Space
- 32 Flexible Data Registers Provide Local Variable Storage
- 64 Data Addressing Registers Support 16 Data Structures
- Background Registers Provide Single-Cycle Context Switch
- Multifunction Instructions Combine Memory Read or Write with Arithmetic Operation
- Single-Cycle Linked-List Update
- 64-Word, Selective Instruction Cache Provides Three Bus Performance
- Single-Cycle Arithmetic Execution
- Two 40-Bit Accumulators
- Powerdown Mode Featuring Low CMOS Standby Power Dissipation with Fast Recovery from Powerdown Condition
- Low Power Dissipation in Idle Mode
- Low Three-Cycle Interrupt Latency

INTEGRATION

- 20K Bytes of On-Chip RAM, Configured as:
  - 4K Words of On-Chip Program or Data RAM (24 Bits)
  - 4K Words of On-Chip Data RAM (16 Bits)
- Five-Channel DMA Controller
- Dual Purpose Program Memory for Both Instruction and Data Storage
- Independent ALU, Multiplier/Accumulator, & Barrel Shifter Computational Units
- Two Independent Data Address Generators Provide:
  - Pre-Modify and Post-Modify Addressing
  - Modification with a Constant
  - Circular/Modulo Addressing
- Powerful Program Sequencer Provides:
  - Zero Overhead Looping
  - Conditional Instruction Execution
- Programmable 16-Bit Interval Timer with Prescaler

SYSTEM INTERFACE

- 16-Bit DMA Port for High Speed Access to On-Chip Memory
- Four Memory Strobes & Separate I/O Memory Space Permits "Glueless" System Design
- Programmable Wait State Generation
- Acknowledge Pin Supports Asynchronous Memory Interface
- Two Synchronous Serial Ports with Companding Hardware, Four 8-Word FIFOs, Separate Receive and Transmit Clocks, DMA, and TDM Multichannel Support
- Automatic Booting of On-Chip Program Memory from Byte-Wide External Memory (e.g., EPROM) or through DMA Port
- 12 Programmable Flag Pins (6 Input and 6 Output) Provide Flexible System Signaling
- Four External Interrupts (Plus 12 Internal and Software Interrupts for a Total of 16)
- IEEE JTAG Standard 1149.1 Test and Emulation Port
- 160-Lead PQFP

GENERAL DESCRIPTION

The ADSP-21csp01 is a single chip DSP optimized for concurrent signal processing (CSP) and other high speed numeric processing applications. The ADSP-21csp01 combines high performance, high bandwidth, 16M address space, DMA port, and fast task switching support to provide efficient multisignal or multichannel processing. The ADSP-21csp01 processor is based on the architecture used for the ADSP-2100 Family. Although this architecture has been modified to improve the processor's performance and add new features, the ADSP-2100 Family code can be ported to the ADSP-21csp01.

The ADSP-21csp01 core architecture consists of three computational units, two data address generators, a program sequencer, and an instruction cache. The ADSP-21csp01 also has a programmable timer, extensive interrupt capabilities, two serial ports, a parallel DMA port, and on-chip memory. The on-chip memory is organized into a single, unified memory space containing four memory blocks with 2K locations in each block. Two blocks are 2K × 24 bits and can be used to store instructions or data, while the others are 2K × 16 bits and can be used to store data.

Fabricated in a high speed, low power CMOS process, the ADSP-21csp01 processor operates at 50 MHz with a 20 ns instruction cycle time. With its on-chip instruction cache, the processor can execute most instructions in a single cycle.

REV. 0

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.
Tel: 617/329-4700 World Wide Web Site: http://www.analog.com
Fax: 617/326-8703

The ADSP-21csp01's flexible architecture and comprehensive instruction set supports a high degree of parallelism. In one cycle the ADSP-21csp01 can perform all of the following operations:

- perform a computation
- perform one or two data moves
- update one or two data address pointers
- generate a program address
- fetch an instruction
- decode an instruction
- test an arithmetic condition
- saturate the accumulator

These operations take place while the processor continues to complete the following tasks:

- receive and transmit data through one or two serial ports
- receive or transmit data through the DMA port
- decrement the timer

Development Tools

A complete Windows*95 or WindowsNT based Integrated Design Environment (IDE) that includes software and hardware tools for system development supports the ADSP-21csp01. The ADSP-21csp01 Development Software includes C and assembly language tools that let you write code for the ADSP-21csp01 processor. The ANSI C compliant compiler generates ADSP-21csp01 assembly source code, while the C runtime library provides ANSI standard and custom DSP library routines. The ADSP-21csp01 assembler produces object code modules that the linker combines into an executable file. The simulator provides interactive instruction-level simulation and debugging with a Windows95 or WindowsNT interface. The IDE also generates PROM programmer compatible files and parallel DMA boot formatted files.

The EZ-ICE® in-circuit emulator allows full speed debugging of ADSP-21csp01 systems by providing a full range of emulation functions such as modification of memory and register values and execution breakpoints. The EZ-ICE emulator uses the IEEE 1149.1 JTAG test access port of the ADSP-21csp01 to monitor and control the target board processor during emulation. The EZ-LAB® development system (that includes a MAFE and 32K external memory) is a complete DSP system that executes EPROM-based programs in a stand-alone configuration for prototyping, or plugs into the PC's ISA bus for program development. The IDE provides you with a single command window for the simulator, EZ-LAB, and EZ-ICE.

ARCHITECTURE OVERVIEW

Figure 1 is a block diagram of the ADSP-21csp01 architecture. The processor contains three independent computational units: the ALU, the multiplier/accumulator with dual accumulators (MAC), and the shifter. The computational units process 16-bit data directly and have provisions to support multiprecision computations. The ALU performs a standard set of arithmetic, logic, and bit manipulation operations; division primitives are also supported. The MAC performs single-cycle multiply, multiply/add, and multiply/subtract operations. The shifter performs logical and arithmetic shifts, normalization, denormalization, and derive exponent operations. The shifter lets you efficiently implement numeric format control including multiword floating-point representations.

The computational units are fed by a set of 16 data registers or 16 background registers. For most instructions, you can use any data register as an input for any computation; this architectural feature provides a code structure that is flexible and easy to use. Multifunction and bit manipulation instructions, however, do have some register restrictions.

A powerful program sequencer and two dedicated data address generators (DAGs) ensure efficient use of these computational units. The sequencer supports most conditional jumps, subroutine calls, and returns in a single cycle. With internal loop counters and loop stacks, the ADSP-21csp01 supports zero overhead looping of code segments, which eliminates the need for jump instructions to maintain loops. You can also nest loops five levels deep.

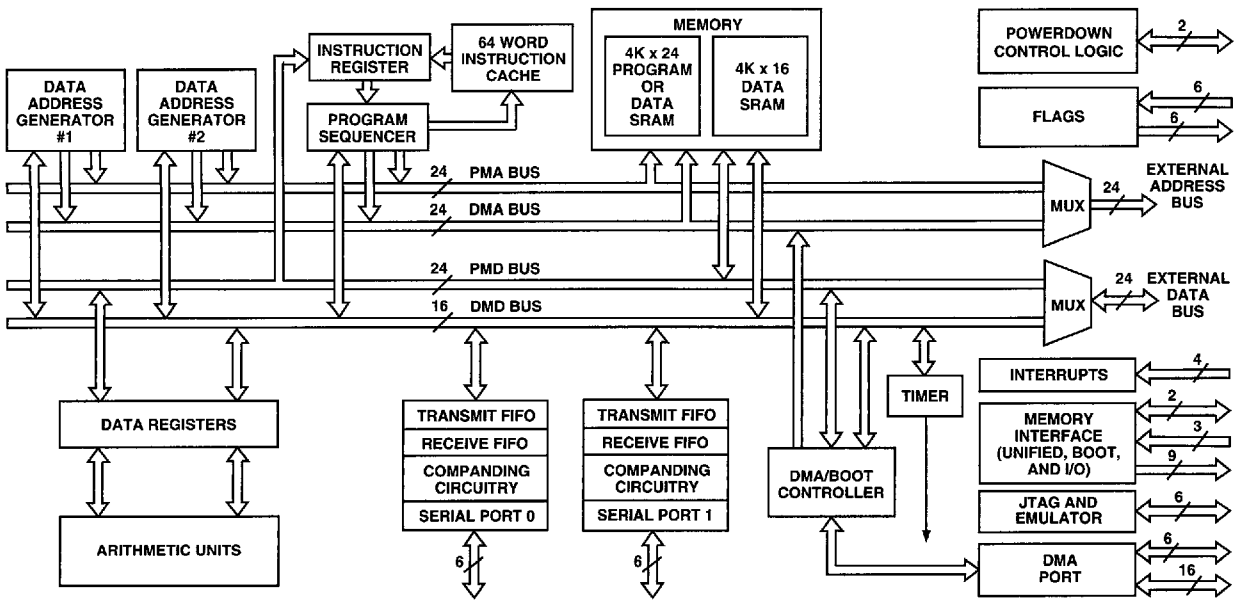


Figure 1. ADSP-21csp01 Block Diagram

*Windows is a registered trademark of Microsoft Corporation.
EZ-ICE and EZ-LAB are registered trademarks of Analog Devices, Inc.

Two data address generators provide addresses for simultaneous dual operand fetches. The DAGs can be used for pre- and post-modify direct addressing. Each DAG maintains and updates four address pointers. When a pointer is used to access data (indirect addressing), it can be pre- or post-modified by the value of one of four modify registers or by a direct modify value. A length value also may be associated with each pointer in a buffer length register to implement automatic modulo addressing for circular buffers. You can place the starting locations for circular buffers at any memory location by loading the starting address into the base register. The 32 DAG registers are shadowed by a set of 32 background registers that enable fast context switching.

The ADSP-21csp01 includes a 64-word on-chip instruction cache that enables three bus operation for fetching an instruction and two data values. The cache is selective—only the instructions whose fetches conflict with data accesses are cached. The cache lets the processor execute core, looped operations, such as digital filter multiply/accumulates and FFT butterfly processing, without bus contention.

Efficient data transfer comes from the DSP's four internal buses:

- Program Memory Address (PMA) Bus
- Program Memory Data (PMD) Bus
- Data Memory Address (DMA) Bus
- Data Memory Data (DMD) Bus

The two address buses (PMA, DMA) share a single external address bus, allowing memory to be expanded off-chip. The two data buses (PMD, DMD) also share a single external data bus.

The memory interface supports slow memories with programmable wait state generation and a hardware acknowledge pin (MACK). External devices can gain control of the processor's buses with the bus request/grant signals ($\overline{BR}$, $\overline{BG}$). The memory interface also supports a separate memory-mapped peripheral space with its own select ($\overline{IOMS}$) and acknowledge ($\overline{IOACK}$) pins.

In addition to the address and data bus for external memory connection, the ADSP-21csp01 has a 16-bit DMA port for connection to external systems. The DMA port has 16 data/address pins and six control pins. The DMA port provides the host system direct access to the DSP's on-chip and off-chip RAM. The ADSP-21csp01 also can be booted through the DMA port.

The ADSP-21csp01 responds to 16 interrupts including a master $\overline{RESET}$ signal. There can be up to four external interrupts and 12 internal interrupts generated by the timer, the serial ports, the DMA port, and powerdown circuitry.

The ADSP-21csp01 has a full set of background registers for all data and DAG registers (plus page registers) that provide a single-cycle context switch.

The ADSP-21csp01 also has two serial ports that provide a synchronous serial interface. The serial port receive and transmit channels can be synchronized separately to programmable internal or external serial clocks. The serial ports also support time division multiplexed channels. Each serial port has 8-word transmit and receive FIFOs to simplify data processing.

The ADSP-21csp01's DMA controller supports automatic data transfers to and from the serial ports and memory.

The ADSP-21csp01 has 12 flag pins: six input pins and six output pins. You can use the input flag pins to control software flow and you can use the output flag pins to control external hardware.

The ADSP-21csp01 has a programmable countdown interval timer that generates periodic interrupts. When the value of the count register reaches zero, the timer generates an interrupt, pulses or toggles the external timer expire pin (TIMEXP), and restarts the countdown.

PIN DESCRIPTIONS

Table I shows the pin definitions for the ADSP-21csp01 processor. Pins are identified as input (I), output (O), bidirectional (I/O), or configurable (I or O).

Table I. ADSP-21csp01 Pin Descriptions

Pin Name(s)	# of Pins	Input/ Output	Function
Address	24	O	Address Output Pins
Data	24	I/O	Data I/O Pins. Input only when reading boot memory; unused data lines may be left floating.
$\overline{RD}$	1	O	External Memory Read Enable Pin
$\overline{WR}$	1	O	External Memory Write Enable Pin
$\overline{MS}_{3-0}$	4	O	Memory Select Pins
$\overline{IOMS}$	1	O	I/O Select Pin
$\overline{BMS}$	1	O	Boot Memory Select Pin
IOACK	1	I	I/O Acknowledge Pin
MACK	1	I	Memory Acknowledge Pin
$\overline{BR}$	1	I or O	External Bus Request Input Pin. Input for Bus Master Mode. Output for Bus Slave Mode.
$\overline{BG}$	1	O or I	External Bus Grant Output Pin. Output for Bus Master Mode. Input for Bus Slave Mode.
$\overline{BGH}$	1	O	External Bus Hang Pin
$BMODE_{1-0}$	2	I	Boot Mode Control Pins
$\overline{BMAST}$	1	I	Bus Master/Bus Slave Control Pin
XTAL	1	I	1/2X External Quartz Crystal Input Pin
CLKIN	1	I	1/2X External Clock or Quartz Crystal Input Pin
CLKOUT	1	O	1X Processor Clock Output Pin
$\overline{RESET}$	1	I	Processor Reset Input Pin
$\overline{IRQ}_{3-0}$	4	I	External Interrupt Pins
$\overline{PWD}$	1	I	Powerdown Pin puts Device in Powerdown Mode.
PWDACK	1	O	Powerdown Acknowledge Pin
DT0	1	O	Data Transmit SPORT0 Pin
DR0	1	I	Data Receive SPORT0 Pin
TFS0	1	I or O	Transmit Frame Sync SPORT0 Pin
RFS0	1	I or O	Receive Frame Sync SPORT0 Pin
TCLK0	1	I or O	Transmit Clock SPORT0 Pin

Table I. ADSP-21csp01 Pin Descriptions (continued)

Pin Name(s)	# of Pins	Input/ Output	Function
RCLK0	1	I or O	Receive Clock SPORT0 Pin
DT1	1	O	Data Transmit SPORT1 Pin
DR1	1	I	Data Receive SPORT1 Pin
TFS1	1	I or O	Transmit Frame Sync SPORT1 Pin
RFS1	1	I or O	Receive Frame Sync SPORT1 Pin
TCLK1	1	I or O	Transmit Clock SPORT1 Pin
RCLK1	1	I or O	Receive Clock SPORT1 Pin
$FLIN_{5-0}$	6	I	Flag Input Pins
$FLOUT_{5-0}$	6	O	Flag Output Pins
TIMEXP	1	O	Timer Expired Pin
IAD_{15-0}	16	I/O	IDMA Address and Data Pins
$\overline{IWR}$	1	I	IDMA Write Enable Pin
$\overline{IRD}$	1	I	IDMA Read Enable Pin
$\overline{IS}$	1	I	IDMA Select Pin
$\overline{IACK}$	1	O	IDMA Acknowledge Pin
IAL	1	I	IDMA Address Latch Pin
IPGL	1	I	IDMA Page Latch Pin
TDI	1	I	JTAG Data Input Pin
TDO	1	O	JTAG Data Output Pin
TMS	1	I	JTAG Sync Pin
TCK	1	I	JTAG Clock Pin
$\overline{TRST}$	1	I	JTAG Reset Pin
$\overline{EMU}$	1	O	Emulator Event Pin
VDD	14		Power Supply Pins
GND	19		Ground Pins

MEMORY MAP & MEMORY INTERFACE

Figure 2 shows the ADSP-21csp01 memory map. The 8K words of on-chip RAM are divided into four separately bussed blocks. Block 1 and Block 2 are 2K × 24 bits each and can be used to store instructions or data. Block 3 and Block 4 are 2K × 16 bits each and are used only to store data.

The ADSP-21csp01 has 24 address lines that can address up to 16 million memory locations. The ADSP-21csp01 has a unified memory space; you can store 16-bit data at any valid memory location.

You can treat all 16M memory locations of the ADSP-21csp01 address range as a single memory space. Four memory select pins ($\overline{MS}_{3-0}$) let you divide the memory into four separate banks. These pins may also eliminate the need for external memory address decoding circuitry. Separating the memory selects can lower the overall system power when tying the $\overline{MS}_{3-0}$ pins to the chip selects on static RAMs. The processor uses the 2 MSBs of the address to generate the four memory strobes, $\overline{MS}_{3-0}$.

$\overline{MS}_{3-0}$ are not active when $\overline{IOMS}$ or $\overline{BMS}$ is active.

From a programmer's perspective, the ADSP-21csp01 has 16M words of linear instruction, paged data, and I/O space. The instruction space is considered to be linear because the user can jump or call to any address in the memory space without setting up a page register.

The ADSP-21csp01 provides three-bus bandwidth by supporting the following combinations of instruction and operand fetches:

- an instruction fetch from cache and two operand fetches from any two on-chip memory blocks

  or

- an instruction fetch from cache, one operand fetch from on-chip memory, and one operand fetch from off-chip memory.

Each memory block supports one access per cycle. Dual data accesses to 16-bit memory can be supported by using Block 3 and Block 4. When accessing Block 1 or Block 2 using the DMD bus, the upper 16 bits are transferred while the lower eight bits are stored in the PX register.

Memory Pages For Data Accesses

The memory space for data accesses is divided into 256 64K word pages. The internal memory is always in Page 0 which has 8K memory words available. There are four page registers used for memory addressing during data accesses. The 8 MSBs of the 24-bit data memory address bus is driven by one of the four page registers (DMPG1, DMPG2, DMAPAGE, and SPORTPAGE) and the 16 LSBs are driven from the instruction, DAG, or DMA controller. Page register selection is implicit in the operation type, as shown by the following sentences:

- Immediate address instructions and DAG1 instructions use the DMPG1 register.
- DAG2 instructions use the DMPG2 register.
- DMA controller accesses use the system control register (DMAPAGE 0) for DMA transfers and the serial port page register (SPORTPAGE) for serial port transfers.

Typically, all the page registers are loaded with the same page number; however, if you load the registers with different page numbers, you can increase the flexibility of the memory. For example:

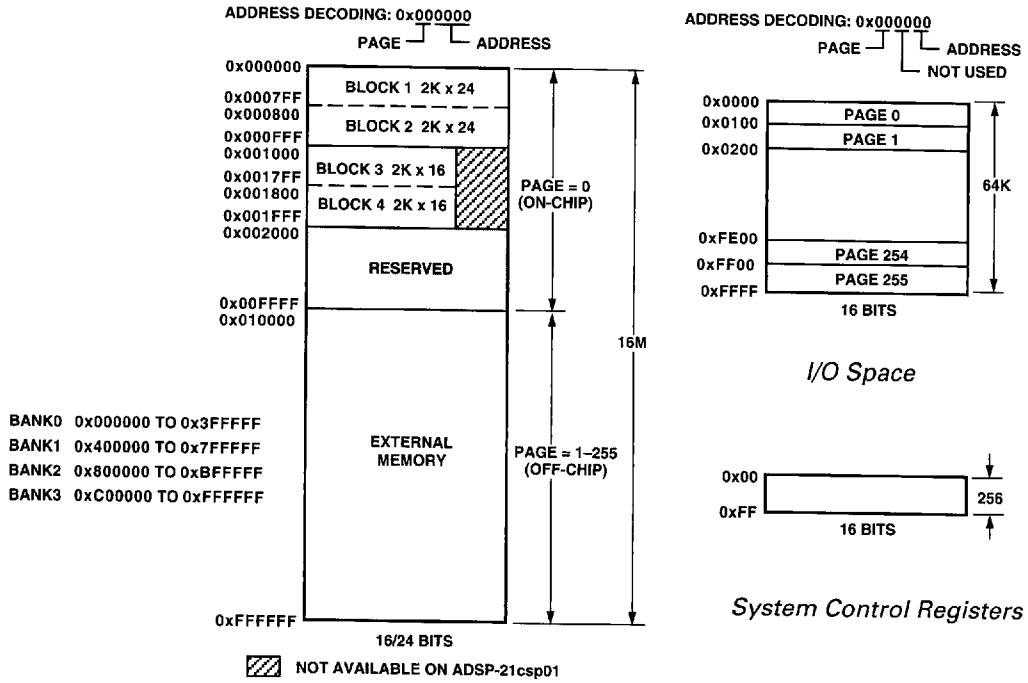
- The DMA space could be kept separate from the application data space.
- High speed copies from one page to another can be accomplished by exploiting the separate DMPG1 and DMPG2 registers attached to the two DAGs.

I/O Space

The I/O space is 256 pages of 256 locations and is externally accessed with special port I/O instructions. The IOPG register is used to form the eight MSBs (A23–A16) of the page address. The location is specified by the eight LSBs (A7–A0). During I/O operations the $\overline{IOMS}$ pin is asserted.

System Control Register Space

The system control register space is 256 linear locations that contain the system control registers. The DSP has specific instructions to access this space and to separate these registers from standard memory space. This feature simplifies the memory map, all standard memory becomes available for data.



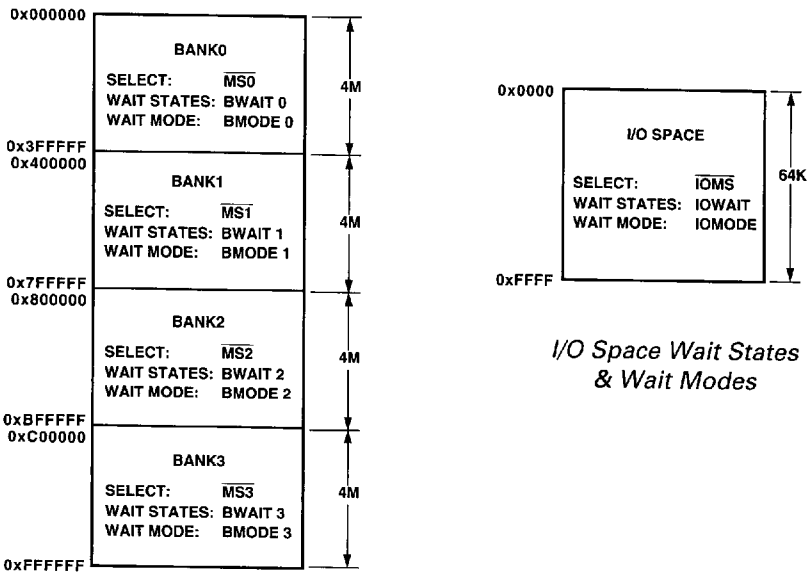


Figure 2. ADSP-21csp01 Memory Map

DIRECT MEMORY ACCESS

DMA Controller

The DMA controller transfers 16-bit and 24-bit words between ADSP-21csp01 peripherals or host CPU and the ADSP-21csp01's memory (internal or external). Registers in the DMA controller let you specify the source and destination in memory for DMA transfers from the serial ports and DMA port. Once the DMA is initialized, the DMA controller can automatically increment the address and begin the next transfer. The DMA controller supports booting during powerup.

The following DMA channels can be active simultaneously:

_ Bidirectional DMA Port

_ Serial Port 0 Transmit

_ Serial Port 0 Receive

_ Serial Port 1 Transmit

_ Serial Port 1 Receive

For each 16-bit or 24-bit word transferred, there is only one processor cycle of overhead.

Serial Ports

The ADSP-21csp01 processor includes two synchronous serial ports (SPORT0 and SPORT1) for peripheral and interprocessor communication.

The serial ports provide a complete synchronous serial interface with optional companding in hardware. Serial data can be automatically buffered in memory using the DMA controller. A wide variety of framed or frameless data transmit and receive modes of operation are available. Each SPORT can generate an internal or accept an external programmable transmit or receive serial clock. SPORTs can also generate internal or accept external frame synchronization signals.

Each serial port has a 6-pin interface consisting of the following signals:

Pin Name	Function
DT0 (or 1)	Data Transmit SPORT0 (or 1)
DR0 (or 1)	Data Receive SPORT0 (or 1)
RFS0 (or 1)	Receive Frame Synchronization SPORT0 (or 1)
TFS0 (or 1)	Transmit Frame Synchronization SPORT0 (or 1)
TCLK0 (or 1)	Transmit Clock SPORT0 (or 1)
RCLK0 (or 1)	Receive Clock SPORT0 (or 1)

Here is a brief list of the ADSP-21csp01 SPORT features:

- **Bidirectional**—each SPORT has a separate, double-buffered transmit and receive function.
- **Flexible Clocking**—each SPORT can use an external transmit or receive serial clock or generate its own clock internally.
- **Flexible Framing**—each SPORT section (receive and transmit) can operate with or without frame synchronization signals for each data word; with internally-generated or externally-generated frame signals; with active high or active low frame signals; with either of two pulse widths and frame signal timing.
- **Different Word Lengths**—each SPORT supports serial data word lengths from 3 to 16 bits.
- **Companding in Hardware**—each SPORT provides optional A-law and μ-law companding according to CCITT recommendation G.711.
- **Separate Transmit and Receive Companding Control**—lets companding control be independent for the receiver and the transmitter.
- **Flexible Interrupt Scheme**—receive and transmit functions can generate unique interrupts upon completion of a data word transfer.
- **Autobuffering with Single-Cycle Overhead**—each SPORT can automatically receive or transmit the contents of an entire data buffer with only one overhead cycle per data word; an interrupt is generated after the transfer of the entire buffer is completed.
- **Separate Transmit and Receive Clocks**—includes separate divide registers for the transmit and receive clocks and associated counters and temporary registers, and separate I/O control for the clock pins.
- **Transmit Empty Status**—serial port status bit to signify that transmission through the shift register is complete. This is in addition to the status that shows the transmit buffer is empty.
- **Dual Multichannel Modes**—multichannel capability is available on both SPORTs.
- **Multichannel Block Sizes of 1–32**—the ADSP-21csp01 has fully programmable block sizes for TDM systems.
- **Multichannel Companding Register**—SPORTs compand on a per channel basis in multichannel mode.
- **FIFO for Transmit and Receive Buffers**—each transmit and receive register (TX0, RX0, TX1, RX1) has an eight location trickle through FIFO. This feature makes the hardware less sensitive to software timing. You can program the various TX and RX interrupts to occur at any location in the FIFO. An internal register in the companding section makes the FIFO effectively nine locations long.

DMA Port

The DMA port provides an efficient means of communication between a host system and the ADSP-21csp01. The host uses the port to access on-chip and off-chip memory with only one DSP cycle per word overhead.

The DMA port has a 16-bit multiplexed address and data bus and supports 24-bit program memory. The DMA port is completely asynchronous and can be written to while the ADSP-21csp01 is operating at full speed.

The host accesses data through the DMA port in two phases. The first phase is DMA address and page latching. When the ADSP-21csp01 asserts an acknowledge, an external device can drive an 8-bit page designation and a 16-bit address onto the bus sequentially. The upper eight bits of the address specifies the page. Page 0, which is the default, specifies on-chip memory; pages 0x01–0xFF specify off-chip memory. When the host asserts the IPGL signal, the falling edge of the IAL signal latches the page into the IDMAPAGE register. The falling edge of the IAL signal latches the 16 bits of the address into the IDMAADR register when the host deasserts IPGL. The $\overline{IS}$ signal must be asserted to select the DSP during these cycles.

The second phase is the data transfer. Once the address is stored, the host can read data from or write data to the ADSP-21csp01's memory space. While the select line ($\overline{IS}$) is asserted, the host signals the ADSP-21csp01, with the appropriate read or write line ($\overline{IRD}$ and $\overline{IWR}$ respectively), that it requires a particular transaction. In either case, there is a one to three processor cycle delay for synchronization. The memory access consumes one additional processor cycle. The DMA address is automatically incremented by one address for each transfer.

Through the IDMAADR and IDMAPAGE registers, the DSP can specify the starting address and the 8-bit page of the DMA operation. Also, the IDMAADR and IDMAPAGE registers, in conjunction with the NEXTADR and NEXTPAGE registers, can be set in software for automatic roll-over to the next DMA transfer.

You can read large amounts of data more efficiently through the DMA read block mode. In this mode, internal prefetching reduces the synchronization delay from three processor cycles to one cycle when the host reads data from the ADSP-21csp01. When you use the DMA read block mode, all transfers in the block must be read; you cannot mix reads and writes.

The DMA port also supports 24-bit data transfers. The transfers require two DMA port data cycles. In the first cycle, IPGL is asserted with the appropriate read or write line ($\overline{IRD}$ or $\overline{IWR}$) and the 16 MSBs of data are transferred on IAD15–0. In the second cycle IPGL is deasserted and the eight LSBs of the data are transferred on IAD7–0. Only one processor overhead cycle is required for 24-bit data transfers.

Booting

On powerup, the ADSP-21csp01 can be booted from an 8-bit wide EPROM or from a host through the DMA port. Booting can also be initiated in software after reset. The destination for boot instructions can be internal or external memory.

Boot Mode Control

There are two pins that control booting, BMODE1 and BMODE0. These pins determine if the ADSP-21csp01 boots under software control and on $\overline{RESET}$ and they determine the source for the boot. Table II summarizes the boot configurations.

Table II. Boot Mode Control

BMODE1	BMODE0	RESET Function	Boot Source
0	0	Boot on $\overline{RESET}$	From Boot Memory (Byte-Wide EPROM)
0	1	Boot on $\overline{RESET}$	From DMA Port
1	0	Processor Runs from External Memory	From Boot Memory (Byte-Wide EPROM) Under Software Control
1	1	Reserved	

When BMODE1 is high, the software may still force a boot after $\overline{RESET}$.

DMA Port Booting

The ADSP-21csp01 can also boot programs through its DMA port. If BMODE0 = high and BMODE1 = low, the ADSP-21csp01 boots from the DMA port. This DMA feature can load as much on-chip memory as desired.

On $\overline{RESET}$, the host CPU writes to the DMA port as it does in any other DMA transfer. However, all bytes in the boot stream are considered to be data, so the port address and page select signals should not be active during booting. Program execution is held until code is written to memory location 0x0000 on any page.

Byte-Wide EPROM Booting

You can boot up to 256 pages (64K bytes each) from EPROM. When booting from an EPROM, the DSP uses the external memory bus to access the EPROM. The $\overline{RD}$ strobe and the $\overline{BMS}$ pins are used. The byte-wide EPROM is mapped onto bits 15–8 of the 24-bit data bus.

On $\overline{RESET}$, booting starts at address 0x000000 and continues through the EPROM until an end-of-stream marker is read. After $\overline{RESET}$, you can boot any boot page through software control by specifying the boot page number in the BOOTPAGE register. The upper eight address bits specify the boot page. Booting starts from address 0x0000 on each page.

The number of wait states for each EPROM access is specified by four bits in the System Control (SYSCNTL) register. On $\overline{RESET}$, the default is 15 wait states.

PROGRAM SEQUENCING

The ADSP-21csp01 program sequencer supports most jumps, calls, and returns in a single cycle. Jumps and calls may be conditional and may be indirect through an index register using the IJPG page register. JUMPs and CALLs within ±4K PC offset execute in a single cycle; long jumps and long calls (LJUMP and LCALL) require two instruction locations and an additional execution cycle.

You can use most branch instructions with or without the delayed-branch (dB) option. The delayed-branch instructions change the program flow after executing the two instructions that follow the branch instruction. Delayed-branch instructions do not require any extra cycles. The nondelayed-branch instructions change program flow immediately after the branch instruction, but you must insert two extra stall cycles in your code (for example, NOP instructions) for proper operation. (All long jumps and calls use nondelayed branching.)

The ADSP-21csp01 supports powerful zero-overhead looping. With a single setup instruction, your code can execute loops from one to 8192 instructions. These loops may be conditional based on computation status or loop count and you can nest loops up to five deep.

Interrupts

The interrupt controller lets the processor respond to the 16 possible interrupts, including $\overline{\text{RESET}}$, with minimum overhead. The ADSP-21csp01 provides four external interrupt pins, $\overline{\text{IRQ0}}$, $\overline{\text{IRQ1}}$, $\overline{\text{IRQ2}}$, and $\overline{\text{IRQ3}}$. The ADSP-21csp01 also supports loop stack overflow interrupts and three user-defined software interrupts, as well as internal interrupts generated by the timer, the serial ports, the DMA port, and powerdown circuitry. The interrupts are prioritized internally and individually maskable (except $\overline{\text{RESET}}$ and powerdown). You can also set bits in the interrupt control register to nest interrupts or mask interrupts globally. The external interrupt pins can be programmed to be either edge- or level-sensitive; level-sensitive is the powerup default. Table III shows the priorities and vector addresses of the interrupts.

Table III. Interrupt Vector Address & Interrupt Priority

Source of Interrupt	Interrupt Vector Address	IRPTL Bit
$\overline{\text{RESET}}$	0x0000	0 *(nonmaskable)*
(Powerup with PUCR = 1)	*(highest priority)*	
Powerdown	0x0004	1 *(nonmaskable)*
Stack Status	0x0008	2
$\overline{\text{IRQ3}}$	0x000C	3
Timer	0x0010	4
User Interrupt 2	0x0014	5
$\overline{\text{IRQ2}}$	0x0018	6
DMA/Host Interface Port Read	0x001C	7
$\overline{\text{IRQ1}}$	0x0020	8
SPORT0 Transmit	0x0024	9
SPORT0 Receive	0x0028	10
SPORT1 Transmit	0x002C	11
SPORT1 Receive	0x0030	12
$\overline{\text{IRQ0}}$	0x0034	13
User Interrupt 1	0x0038	14
User Interrupt 0	0x003C	15
	(lowest priority)	

Each interrupt has four vector locations for the interrupt service routine. $\overline{\text{RESET}}$ is the first interrupt and starts at address 0x0000.

The IRPTL register can be used to force or clear individual interrupts (except $\overline{\text{RESET}}$ and Powerdown). When a bit in the IRPTL register is set through software control, the interrupt will be serviced.

Interrupts, except $\overline{\text{RESET}}$ and Powerdown, can be masked or unmasked with the 16-bit IMASK register. Individual interrupt requests are logically ANDed with the bits in IMASK; the highest priority interrupt is then selected.

You can also access pending interrupt status. When read, a bit set in IRPTL designates a pending interrupt. There is a one-for-one correspondence between IRPTL bits and IMASK bits.

The interrupt control and mode extension register (ICNTL) is used to set the edge- or level-sensitivity of the external interrupt pins. ICNTL also includes the following bits:

- **Nesting Mode Enable**—This bit enables the nested interrupt mode of the ADSP-21csp01; a higher priority service request will interrupt the currently serviced interrupt routine. When this mode is disabled, pending interrupts are not serviced until the processor finishes servicing the current interrupt.
- **Global Interrupt Enable**—This bit is a global interrupt enable and disable bit. The global interrupt enable bit can be set and cleared through software control.

Instruction Cache

The instruction cache is a two-way set-associative cache that is 64 locations deep. The operation of the cache is completely transparent to the user. The purpose of the cache is to eliminate bus contention that may occur on the internal PM data bus, the external bus, or memory blocks because of a data access that attempts to use the same resource as the instruction fetch. The ADSP-21csp01 only caches instructions that conflict with memory accesses (selective caching).

The instruction cache has controls to allow for flushing (invalidating) the entire cache under program control. You can also freeze the contents of the instruction cache. When the cache is frozen, no new instructions can be written into the cache, but the processor continues to execute instructions saved in the cache.

The instruction cache is a two-way set-associative tagged cache (with 32 sets). Each entry consists of an address tag and an instruction word. A set consists of two entries and a least recently used (LRU) bit. The LRU bit is used to determine which entry in the set has been used or updated least recently. At processor reset, the cache is cleared.

Stacks

The ADSP-21csp01 PC and LOOP stacks are each 24 bits wide. Two stack registers, STACKA and STACKP, are used to accommodate the width of these stacks. When you use POP PC and POP LOOP commands, 24-bit values are written to these registers: STACKA holds the 16 LSBs and STACKP holds the 8 MSBs. When you use PUSH PC and PUSH LOOP commands, 24-bit values are read from these registers: STACKA yields the 16 LSBs and STACKP yields the 8 MSBs.

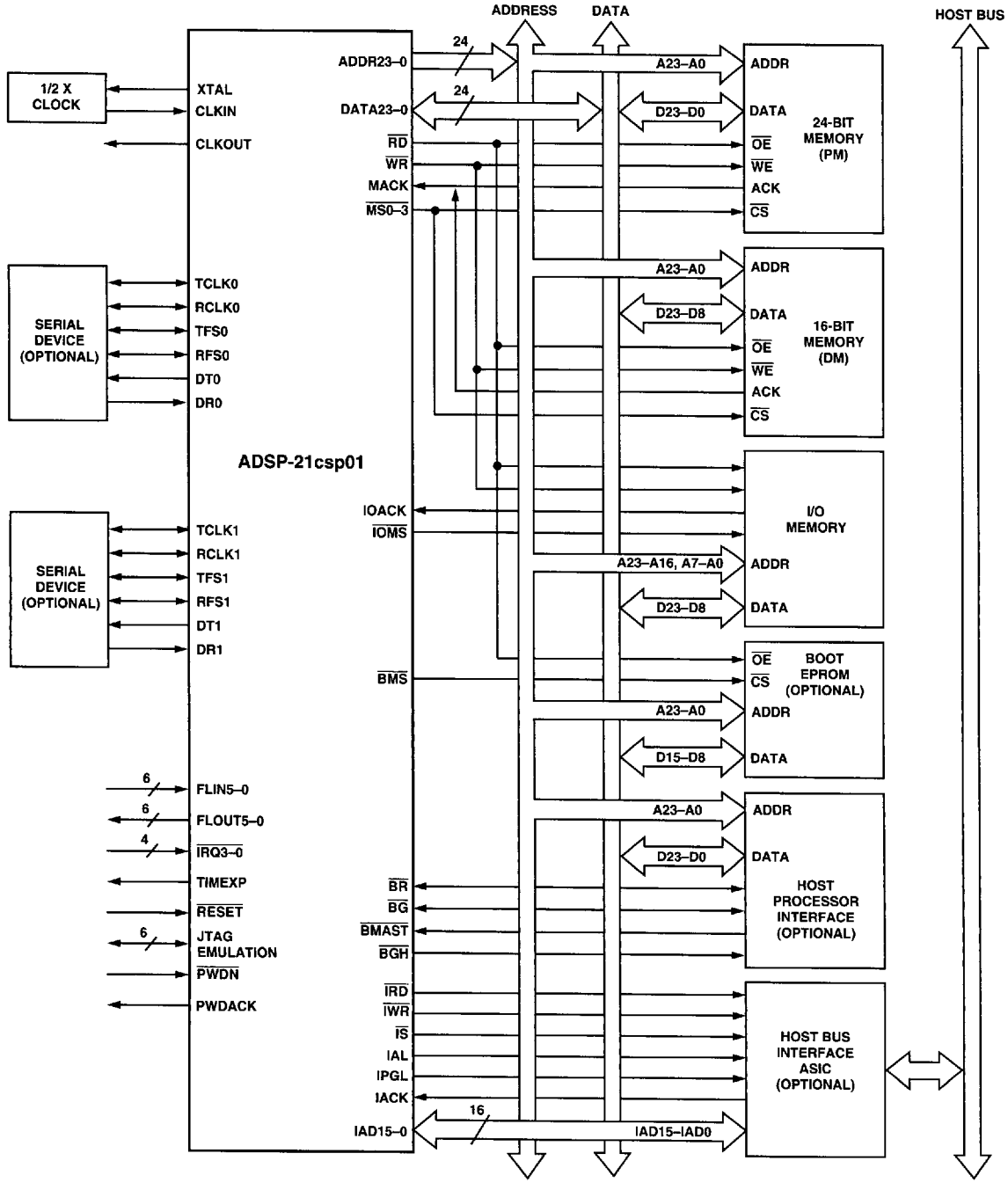


Figure 3. ADSP-21csp01 Basic System Configuration

Stack Depth
One additional stack location is included in the PC and status (STS) stacks for use with the emulator interrupt. In summary, the stack depths available to the user are:
_ PC = 31
_ STS = 15
_ LOOP = 5
_ CNTR = 5

SYSTEM INTERFACE

Figure 3 shows a basic system configuration that includes the ADSP-21csp01, two serial devices, external program and data memory, an external boot EPROM, a host processor interface, and host bus interface.

Clock Signals
The ADSP-21csp01 uses an input clock with a frequency equal to half the instruction rate; a 25 MHz input clock yields a 20 ns processor cycle (which is equivalent to 50 MHz). Normally, instructions are executed in a single processor cycle. All device timing is relative to the internal instruction clock rate indicated by the CLKOUT signal.

The CLKIN input cannot be halted, changed during operation, or operated below the specified frequency during normal operation. The only exception is while the processor is in the power-down state.

The ADSP-21csp01 can be clocked by a crystal or by a TTL-compatible clock signal.

If an external clock is used, it should be a TTL-compatible signal running at half the instruction rate. The signal is connected to the processor's CLKIN input. When an external clock is used, the XTAL input *must* be left unconnected.

Because the ADSP-21csp01 includes an on-chip oscillator circuit, an external crystal may be used. The crystal should be connected across the CLKIN and XTAL pins, with two capacitors connected as shown in Figure 4. Capacitor values depend on the crystal type and should be specified by the crystal manufacturer. A parallel-resonant, fundamental-frequency, microprocessor-grade crystal should be used.

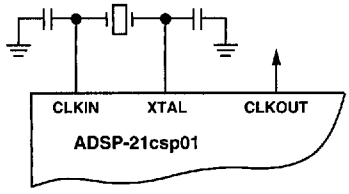


Figure 4. External Crystal Connections

A clock output (CLKOUT) signal is generated by the processor and is equal to the processor's instruction rate. The clock output can be held low (disabled) through software control by setting the CLKODIS bit in the System Control Register.

Bus Request, Bus Grant, and Bus Hang
The ADSP-21csp01 can be run in bus master or bus slave mode, depending on the state of the $\overline{\text{BMAST}}$ pin. When this pin is driven low (normal in single processor systems), the processor is the bus master; when it is driven high, the processor becomes a bus slave. Bus master/slave mode determines the function of the bus request, bus grant, and bus hang pins ($\overline{\text{BR}}$, $\overline{\text{BG}}$, and $\overline{\text{BGH}}$).

In bus master mode:

- $\overline{\text{BR}}$ is an INPUT that tells the ADSP-21csp01 that another device requests control of the external memory bus.
- $\overline{\text{BG}}$ is an OUTPUT whereby the ADSP-21csp01 grants control of the external memory bus to another device.

In bus slave mode:

- $\overline{\text{BR}}$ is an OUTPUT that tells other devices the ADSP-21csp01 requests control of the external memory bus.
- $\overline{\text{BG}}$ is an INPUT that grants the ADSP-21csp01 external memory bus control.

$\overline{\text{BGH}}$ (bus hang) lets other devices know that the ADSP-21csp01 wants to perform an external access, but it does not have control of the bus. Bus lock mode can give the ADSP-21csp01 exclusive access to the external bus. $\overline{\text{BGH}}$ is valid in both master and slave modes, but its function is affected by bus lock mode.

- In bus master mode, bus lock "locks out" bus granting to an external device. $\overline{\text{BG}}$ remains inactive after receiving $\overline{\text{BR}}$, even if external memory accesses are not currently underway. If the bus has already been granted when bus lock is enabled, $\overline{\text{BG}}$ stays active until $\overline{\text{BR}}$ is removed.
- In bus slave mode, bus lock automatically generates a bus request. The ADSP-21csp01 takes control of the bus on $\overline{\text{BG}}$ regardless of whether an external memory access is ongoing. $\overline{\text{BGH}}$ will not go active unless an external access is actually attempted before $\overline{\text{BG}}$ is returned.

Timer Expired
The ADSP-21csp01 has a timer expired (TIMEXP) output pin. When the timer is enabled, this pin pulses or toggles each time the timer expires. Bit 15 (EXPMODE bit) of the TSCALE register programs TIMEXP to pulse or toggle. Once the timer is enabled, TIMEXP is generated whether or not the timer interrupt is enabled or the interrupt is serviced. The TSCALE register is undefined after a reset. TIMEXP is low when the timer is disabled.

Terminating Unused Pins

Table IV shows the recommendations for terminating unused pins.

Table IV. ADSP-21csp01 Pin Terminations

Pin Name(s)	Unused Connection	# of Pins	Input/ Output	Comments
Address	Float	24	O	
Data	Float	24	I/O	
$\overline{RD}$	Float	1	O	
$\overline{WR}$	Float	1	O	
$\overline{MS3–0}$	Float	4	O	
$\overline{IOMS}$	Float	1	O	
$\overline{BMS}$	Float	1	O	
IOACK	HIGH	1	I	Active
MACK	HIGH	1	I	Active
$\overline{BR}$	HIGH	1	I	Master Mode, Default at $\overline{RESET}$, Inactive
	Float		O	Slave Mode
$\overline{BG}$	Float	1	O	Master Mode, Default at $\overline{RESET}$
	HIGH		I	Slave Mode, Inactive
$\overline{BGH}$	Float	1	O	
BMODE1–0	N/A	2	I	Follow Details in "Booting" Section
$\overline{BMAST}$	N/A	1	I	Follow Details in "Bus Request, Bus Grant, and Bus Hang" Section
XTAL	Float	1	I	Disable with Bit 15 in System Control Register
CLKIN	N/A	1	I	Follow Details in ß"Clock Signals" Section
CLKOUT	Float	1	O	Disable with Bit 13 in System Control Register
$\overline{RESET}$	N/A	1	I	
$\overline{IRQ3–0}$	HIGH	1	I	Inactive
$\overline{PWD}$	HIGH	1	I	Inactive
PWDACK	Float	1	O	
DT0	Float	1	O	
DR0	LOW or HIGH	1	I	
TFS0	LOW or HIGH	1	I	Input at Reset (Default)
	Float		O	
RFS0	LOW or HIGH	1	I	Input at Reset (Default)
	Float		O	
TCLK0	LOW or HIGH	1	I	Input at Reset (Default)
	Float		O	
RCLK0	LOW or HIGH	1	I	Input at Reset (Default)
	Float		O	
DT1	Float	1	O	
DR1	LOW or HIGH	1	I	
TFS1	LOW or HIGH	1	I	Input at Reset (Default)
	Float		O	
RFS1	LOW or HIGH	1	I	Input at Reset (Default)
	Float		O	
TCLK1	LOW or HIGH	1	I	Input at Reset (Default)
	Float		O	
RCLK1	LOW or HIGH	1	I	Input at Reset (Default)
	Float		O	
FLIN5–0	LOW or HIGH	6	I	
FLOUT5–0	Float	6	O	
TIMEXP	Float	1	O	
$\overline{IAD15–0}$	Float	16	I/O	
$\overline{IWR}$	HIGH	1	I	Inactive
$\overline{IRD}$	HIGH	1	I	Inactive
$\overline{IS}$	HIGH	1	I	Inactive
$\overline{IACK}$	Float	1	O	
IAL	LOW	1	I	Inactive
IPGL	LOW	1	I	Inactive
TDI	Float or HIGH	1	I	Pulled High Internally
TDO	Float	1	O	
TMS	Float or HIGH	1	I	Pulled High Internally
TCK	LOW	1	I	
$\overline{TRST}$	Float or HIGH	1	I	Pulled High Internally
$\overline{EMU}$	Float	1	O	

External Address and Data Bus Usage
Figure 5 shows how the ADSP-21csp01 uses the external address and data busses during data reads and writes.

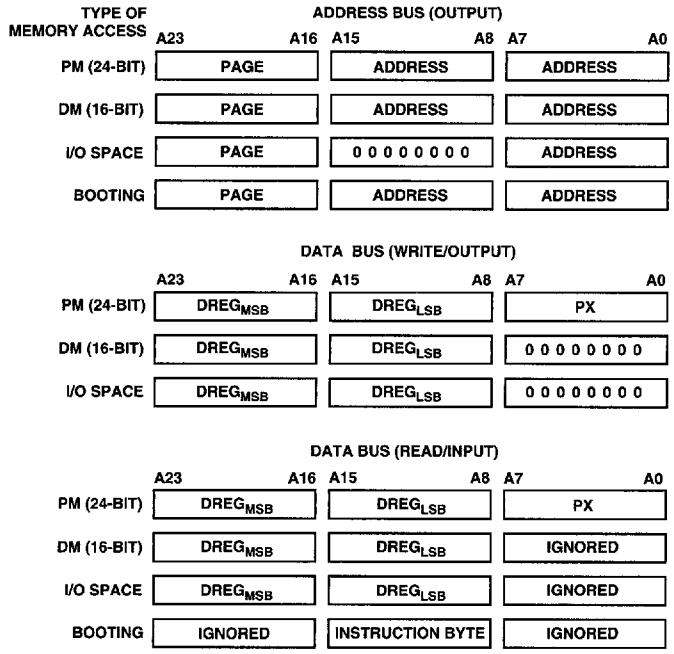


Figure 5. External Address and Data Bus Usage

DMA Port Address and Data Bus Usage
Figure 6 shows how the ADSP-21csp01 uses the DMA port address and data busses during data reads and writes.

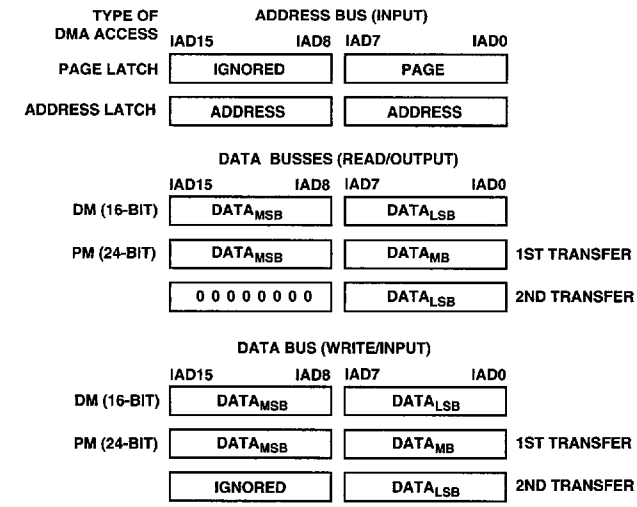


Figure 6. DMA Port Address and Data Bus Usage

Target Board Connector for EZ-ICE Probe
The ADSP-21csp01 EZ-ICE Emulator uses the IEEE 1149.1 JTAG test access port of the ADSP-21csp01 to monitor and control the target board processor during emulation. The EZ-ICE's probe requires the ADSP-21csp01's CLKIN (optional), TMS, TCK, $\overline{TRST}$, TD1, TD0, $\overline{EMU}$, and GND signals be made accessible on the target system through a 14-pin connector (a pin strip header) such as that shown in Figure 7. The EZ-ICE probe plugs directly onto this connector for chip-on-board emulation. You must add this connector to your target board design if you intend to use the ADSP-21csp01 EZ-ICE. The length of the traces between the connector and the ADSP-21csp01's JTAG pins should be as short as possible.

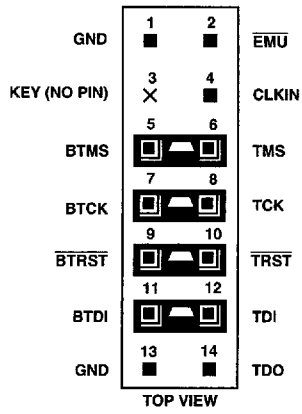


Figure 7. Target Board Connector for ADSP-21csp01 EZ-ICE Emulator (Jumpers in Place)

The 14-pin, 2-row strip header is keyed at the Pin 3 location—you must remove Pin 3 from the header. The pins must be 0.025 inch square and at least 0.20 inch in length. Pin spacing should be 0.1 × 0.1 inches. Pin strip headers are available from vendors such as 3M, McKenzie, and Samtec.

The BTMS, BTCK, $\overline{BTRST}$, and BTDI signals are provided so that the test access port can also be used for board-level testing. When the connector is not being used for emulation, place jumpers between the BXXX pins (Pins 5, 7, 9, and 11) and XXX pins (Pins 6, 8, 10, and 12) as shown in Figure 7. If you are not going to use the test access port for board testing, tie $\overline{BTRST}$ to GND and tie or pull up BTCK to V_{DD}. The $\overline{TRST}$ pin must be asserted after power-up (through $\overline{BTRST}$ on the connector) or held low for proper operation of the ADSP-21csp01. None of the BXXX pins are connected on the EZ-ICE probe.

The JTAG signals are terminated on the EZ-ICE probe as shown in Table V.

Table V. JTAG Signal Termination

Signal	Termination
TMS	Driven through 22 Ω Resistor (16 μA/–3.2 μA Driver)
TCK	Driven at 10 MHz through 22 Ω Resistor (16 μA/–3.2 μA Driver)
$\overline{TRST}$	Driven by Open-Drain Driver* (Pulled Up by On-Chip 50 kΩ Resistor)
TDI	Driven by 16 μA/-3.2 μA Driver
TD0	One TTL Load, No Termination
CLKIN	One TTL Load, No Termination (Optional Signal)
$\overline{EMU}$	4.7 kΩ Pull-Up Resistor, One TTL Load (Open Drain Output from ADSP-21csp01)

*$\overline{TRST}$ is driven low until the EZ-ICE probe is turned on by the EZ-ICE software (after the invocation command).

Figure 8 shows JTAG scan path connections for systems that contain multiple ADSP-21csp01 processors.

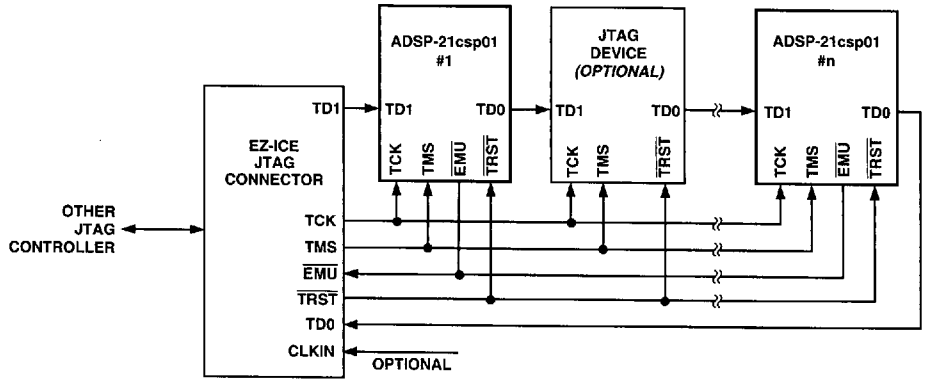


Figure 8. JTAG Scan Path Connections for Multiple ADSP-21csp01 Systems

Connecting CLKIN to Pin 4 of the EZ-ICE header is optional. The emulator only uses CLKIN when directed to perform operations such as starting, stopping, and single-stepping multiple ADSP-21csp01s in a *synchronous* manner. If you do not need these operations to occur synchronously on multiple processors, tie Pin 4 of the EZ-ICE header to ground.

If synchronous multiprocessor operations are needed and CLKIN is connected, clock skew between the multiple ADSP-21csp01 processors and the CLKIN pin on the EZ-ICE header *must be minimal.* If skew is too large, synchronous operations may be off by one cycle between processors. For synchronous multiprocessor operation, TCK, TMS, CLKIN, and $\overline{\text{EMU}}$ should be treated as critical signals in terms of skew, and should be laid out as short as possible on you board. If TCK, TMS, and CLKIN are driving a large number of ADSP-21csp01s (more than eight) in your system, treat them as a "clock tree" using multiple drivers to minimize skew. Figure 9 shows a JTAG clock tree for multiple ADSP-21csp01 systems.

If synchronous multiprocessor operations are not needed (i.e., CLKIN is not connected), just use the appropriate parallel termination on TCK and TMS. TD1, TD0, $\overline{\text{EMU}}$, and $\overline{\text{TRST}}$ are not critical signals in terms of skew.

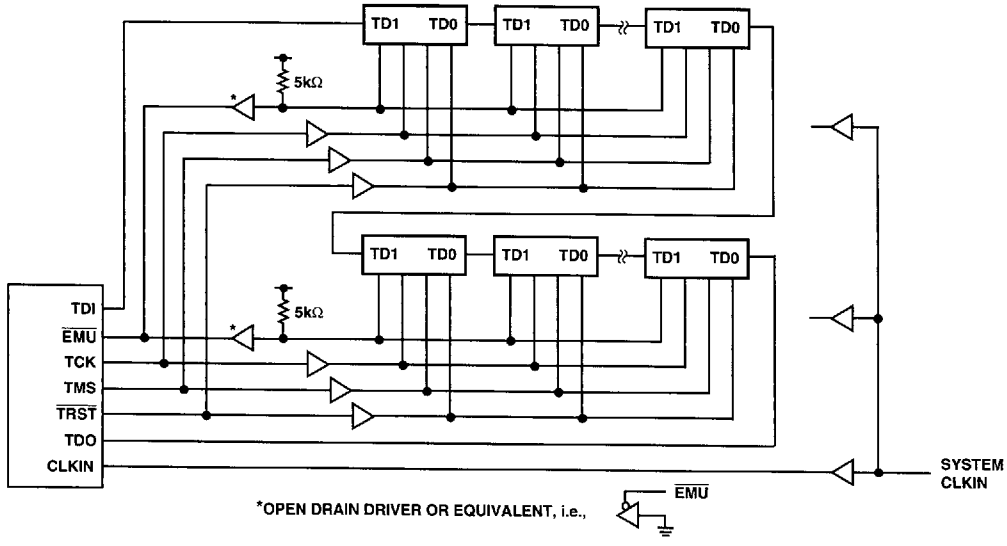


Figure 9. JTAG Clock Tree for Multiple ADSP-21csp01 Systems

LOW POWER OPERATION

The ADSP-21csp01 has three low power modes that significantly reduce the power dissipation when the device operates under standby conditions. These modes are:

- Idle
- Slow Idle
- Powerdown

The CLKOUT and XTAL pins may also be disabled, by clearing Bits 15 and 13 in the System Control register, to reduce external power dissipation.

Idle

When the ADSP-21csp01 is in the Idle Mode, the processor waits indefinitely in a low power state until an interrupt occurs. When an unmasked interrupt occurs, it is serviced, then execution continues with the next instruction after the *IDLE* instruction.

Slow Idle

The *IDLE* instruction on the ADSP-21csp01 lets the processor's internal clock signal be slowed during the *IDLE* instruction, further reducing power consumption. The reduced clock frequency, a programmable fraction of the normal clock rate, is specified by a selectable divisor given in the *IDLE* instruction. The format of the instruction is:

IDLE (n);

where n = 16, 32, 64, or 128. This instruction keeps the processor idle, but operating at the slower clock rate. While it is in this state, the processor's other internally-generated clock signals, such as SPORT clocks, CLKOUT, and timer clock, are reduced by the same ratio. The default form of the instruction, when no clock divisor is given, is the standard *IDLE* instruction.

When the *IDLE (n)* instruction is used, it effectively slows down the processor's internal clock and thus its response time to incoming interrupts. The one-cycle response time of the standard idle state is increased by n, the clock divisor. When an enabled interrupt is received, the ADSP-21csp01 will remain in the idle state for up to a maximum of n processor cycles (n = 16, 32, 64, or 128) before resuming normal operation.

When the *IDLE (n)* instruction is used in systems that have an externally generated serial clock, the serial clock rate must not be faster than the processor's reduced internal clock rate. Under these conditions, interrupts must not be generated at a faster rate than can be serviced, due to the additional time the processor takes to come out of the idle state (a maximum of n processor cycles).

Powerdown

The ADSP-21csp01 processor has a feature that lets the processor enter a very low power dormant state through hardware or software control. Here is a brief list of powerdown features.

- Quick recovery from powerdown.
- Support for crystal operation includes disabling the oscillator to save power.
- Powerdown is initiated by either the powerdown pin ($\overline{\text{PWD}}$) or the software powerdown force bit.
- Interrupt support allows instructions to be executed before optionally powering down. The powerdown interrupt also can be used as a non-maskable, level-sensitive interrupt.
- Powerup context reset (PUCR) allows the processor to continue where it left off or start with a clean context when leaving the powerdown state.
- The $\overline{\text{RESET}}$ pin also can be used to terminate powerdown.
- Powerdown acknowledge pin (PWDACK) indicates when the processor has entered powerdown.

INSTRUCTION SET DESCRIPTION

The ADSP-21csp01 assembly language uses an algebraic syntax that makes it easier for you to code and makes your code more readable. The sources and destinations of computations and data movements are written explicitly in each assembly statement, eliminating cryptic assembler mnemonics. All instructions, except LJUMP and LCALL, assemble into single 24-bit words. Most instructions execute in a single cycle. The instructions encompass a wide variety or instruction types along with a high degree of operational parallelism. There are five basic categories of instructions: computational instructions, data move instructions, multifunction instructions, program flow control instructions, and miscellaneous instructions. The complete instruction set is summarized in this section. The *ADSP-21csp01 User's Manual* contains a complete reference for the instruction set.

Hardware Condition Codes

The hardware condition codes are used to determine whether a conditional instruction, such as a jump, call, return, MAC saturation, or arithmetic operation, is performed. The 16 basic composite status conditions and their derivations are shown in Table VI. Since arithmetic status is latched into ASTAT at the end of a processor cycle, the condition logic represents conditions generated on the previous cycle.

Table VI. Hardware Condition Codes

Code	Status Condition	True If:
EQ	ALU Equal Zero	AZ = 1
NE	ALU Not Equal Zero	AZ = 0
GT	ALU Greater Than Zero	(AN .XOR. AV) .OR. AZ = 0
LE	ALU Less Than or Equal Zero	(AN .XOR. AV) .OR. AZ = 1
LT	ALU Less Than Zero	AN .XOR. AV = 1
GE	Greater Than or Equal Zero	AN .XOR. AV = 0
AV	ALU Overflow	AV = 1
NOT AV	NOT ALU Overflow	AV = 0
AC	ALU Carry	AC = 1
NOT AC	NOT ALU Carry Overflow	AC = 0
SWCOND	Software Condition True	
NOT SWCOND	Software Condition False	
MV	MAC Overflow	MV = 1
NOT MV	NOT MAC Overflow	MV = 0
NOT CE	NOT Counter Expired	CE = 0
FOREVER	Always	Always True

Software Condition Codes

The software condition codes let you use the six input flag pins to control conditional instructions. The condition code register (CCODE) is used to program the software conditions. The software condition codes also support the the ALU X input sign and shifter result register (SR) overflow conditions. The software condition codes are shown in Table VII.

Table VII. Software Condition Codes

CCODE Register	1010 SWCOND	1011 NOT SWCOND
0x00	FLIN0 High	FLIN0 Low
0x01	FLIN1 High	FLIN1 Low
0x02	FLIN2 High	FLIN2 Low
0x03	FLIN3 High	FLIN3 Low
0x04	FLIN4 High	FLIN4 Low
0x05	FLIN5 High	FLIN5 Low
0x06	Undefined	Undefined
0x07	Undefined	Undefined
0x08	X Input NEG	X Input POS
0x09	SV = 1	SV = 0
0x0A–0x0F	Undefined	Undefined

Instruction Set Summary

Key

UPPERCASE	Assembler keyword; exact syntax of instruction
[text]	Parts of the instructions in brackets are optional
x | y | z	Choose x, y, or z
[,...]	Any of the operations allowed by this instruction can be combined in any order, separated by commas
In, Mm	Index and modify registers for indirect addressing
xreg	X input; permissible registers depend on instruction
yreg	Y input; permissible registers depend on instruction
<data>	Immediate data value
<address>	Immediate address value
<cnst>	Constant value
<offset>	Offset value
condition	Conditions from Table V
dreg	Computation unit data register
qreg	Any register (except system control registers)
ALU	Any ALU instruction (except division)
MAC	Any multiply/accumulate instruction (except saturate)
SHIFT	Any shifter instruction (except shift immediate)

ALU Instructions

[IF condition] AR | AF = xreg + yreg | + yreg + c | + c | + constant | + constant + c;
– yreg |– yreg + c – 1 | + c – 1 |– constant | – constant – c + 1;
yreg – xreg |– xreg + c – 1 ;
– xreg + c – 1 |+ constant | + constant + c – 1;
yreg + 1 | – 1 ;
xreg AND | OR | XOR yreg | constant ;
NOT xreg | yreg | 0 ;
PASS xreg | yreg | 0 | 1 | – 1 | constant ;
— xreg | — yreg ;
ABS xreg ;
CLRBIT | SETBIT | TGLBIT | TSTBIT n OF xreg ;

DIVS yreg , xreg ;
DIVQ xreg ;

MAC Instructions

[IF condition] MR | SR = xreg * yreg ; (SS | SU | US | UU | RND) ;
MR | SR + xreg * yreg ; (SS | SU | US | UU | RND) ;
MR | SR — xreg * yreg ; (SS | SU | US | UU | RND) ;
MR | SR [(RND)] ;
0 ;

IF MV SAT MR ; | IF SV SAT SR ;

Shifter Instructions

[IF condition] SR = [SR OR] ASHIFT | LSHIFT | NORM xreg (HI | LO) ;
[IF condition] SE = EXP xreg (HI | LO | HIX) ;
[IF condition] SB = EXPADJ xreg ;
SR = [SR OR] ASHIFT | LSHIFT xreg BY < data > (HI | LO) ;

Move Instructions

qreg = < cnst > | qreg ;
dreg = DM | IO | REG (< address >) ;
DM | IO | REG (< address >) = dreg ;
dreg = DM (I_n , < offset > | < offset > , I_n) ;
DM (I_n , < offset > | < offset > , In) = dreg ;
qreg = DM | PM (I_n , M_m | M_m , I_n) ;
DM | PM (I_n , M_m | M_m , I_nn) = qreg ;
DM (I_{n0} , M_m | M_m , I_{n0}) = $qreg_0$, $qreg_0$ = I_{n0} ;

Multifunction Instructions

ALU | MAC* , xreg = DM (I_a, M_b) , yreg = PM (I_c, M_d) ;
xreg = DM (I_a, M_b) , yreg = PM (I_c, M_d) ;
ALU | MAC | SHIFT* , dreg = DM | PM (I_a, M_b) ;
DM | PM (I_a, M_b) = dreg , ALU | MAC | SHIFT* ;
ALU | MAC | SHIFT* , dreg = dreg ;

*All computation is unconditional; Division and Shift Immediate operations prohibited.

Program Flow Control Instructions

[IF condition] JUMP | CALL = (PC , < label > | < offset >) | (I_n) [(DB)] ;
[IF condition] RTI | RTS [(DB)] ;
[IF condition] [L]JUMP | [L]CALL < label > | < address > ;
DO < label > | < offset > | (PC , < label > | < offset >) [UNTIL condition] ;
IDLE [(n)] ;

Miscellaneous Instructions

MODIFY (I_n , < offset >) | (I_n , M_m) ;
ENA | DIS SEC_REG | BIT_REV | AV_LATCH | AR_SAT | M_MODE | TIMER | SEC_DAG | INT ;
PUSH | POP PC | LOOP | CNTR | STS ;
FLUSH CACHE ;
SETINT | CLRINT n ;
NOP ;

Example Code

The following example is a code fragment that performs the filter tap update for an adaptive (least-mean-squared algorithm) filter. Notice that the computations in the instructions are written like algebraic equations.

```
     SR=MX0*MY1(RND),MX0=DM(I2,M1);                    /* SR1=error*beta */
     MR=MX0*SR1(RND),AY0=DM(I6,M5);
     DO adapt UNTIL CE;
          AR=MR1+AY0,MX0=DM(I2,M1),AY0=PM(I6,M7);
adapt:    DM(I6,M6)=AR,MR=MX0*SR1(RND);
     MODIFY (I2,M3);                                   /* Point to oldest data */
     MODIFY (I6,M7);                                   /* Point to start of data */
```

SPECIFICATIONS

RECOMMENDED OPERATING CONDITIONS

Parameter		K Grade Min	K Grade Max	B Grade Min	B Grade Max	Unit
V_{DD}	Supply Voltage	4.5	5.5	4.5	5.5	V
T_{AMB}	Ambient Operating Temperature	0	+70	–40	+85	°C

ELECTRICAL CHARACTERISTICS

Parameter		Test Conditions	K/B Grades Min	K/B Grades Max	Unit
V_{IH}	Hi-Level Input Voltage[1, 2]	@ V_{DD} = max	2.0		V
V_{IH}	Hi-Level CLKIN $\overline{RESET}$, Testport, SPORTs Voltage[3]	@ V_{DD} = max	2.2		V
V_{IL}	Lo-Level Input Voltage[1, 4]	@ V_{DD} = min		0.8	V
V_{OH}	Hi-Level Output Voltage[1, 5, 6]	@ V_{DD} = min, I_{OH} = –0.5 mA	2.4		V
		@ V_{DD} = min, I_{OH} = –100 µA[7]	V_{DD} – 0.3		V
V_{OL}	Lo-Level Output Voltage[1, 5]	@ V_{DD} = min, I_{OL} = 2 mA		0.4	V
I_{IH}	Hi-Level Input Current[4]	@ V_{DD} = max, V_{IN} = V_{DD} max		10	µA
I_{IL}	Lo-Level Input Current[4]	@ V_{DD} = max, V_{IN} = 0 V		10	µA
I_{OZH}	Three-State Leakage Current[8]	@ V_{DD} = max, V_{IN} = V_{DD} max[9]		10	µA
I_{OZL}	Three-State Leakage Current[8]	@ V_{DD} = max, V_{IN} = 0 V[9]		10	µA
I_{DD}	Supply Current (Idle)[10, 11]	@ V_{DD} = max		55	mA
I_{DD}	Supply Current (Dynamic)[11]	@ V_{DD} = max, t_{CK} = 20 ns[12]		150	mA
I_{DD}	Supply Current (Powerdown)[11]	Lowest Power Mode[13] CLKIN Off		50	µA
I_{DD}	Supply Current (Powerdown)[11]	Lowest Power Mode[13] CLKIN Running		120	µA
C_I	Input Pin Capacitance[4, 7, 14]	@ V_{IN} = 2.5 V, f_{IN} = 1.0 MHz, T_{AMB} = +25°C		8	pF
C_O	Output Pin Capacitance[7, 8, 14, 15]	@ V_{IN} = 2.5 V, f_{IN} = 1.0 MHz, T_{AMB} = +25°C		8	pF

NOTES

[1] Bidirectional Pins: D0–D23, $\overline{BR}$, $\overline{BG}$, TFS0, TFS1, RFS0, RFS1, TCLK0, TCLK1, RCLK0, RCLK1, IAD0–IAD15.

[2] Input Only Pins: IOACK, MACK, $\overline{RESET}$, $\overline{IRQ3–0}$, PWD, DR0, DR1, FLIN5–0, $\overline{IWR}$, $\overline{IRD}$, $\overline{IS}$, IAL, IPGL, BMODE0–1, BMAST, TDI, TMS, TCLK, $\overline{TRST}$.

[3] Testport: $\overline{TRST}$, TCK, TMS, TD0, TD1. SPORTS: RCLK0/1, TCLK0/1, RFS0/1, TFS0/1, DR0/1, DT0/1.

[4] Input Only Pins (with CLKIN): CLKIN, IOACK, MACK, $\overline{RESET}$, $\overline{IRQ3–0}$, PWD, DR0, DR1, FLIN5–0, $\overline{IWR}$, $\overline{IRD}$, $\overline{IS}$, IAL, IPGL, BMODE0–1, BMAST, TDI, TMS, TCLK, $\overline{TRST}$.

[5] Output Only pins: A0–A23, $\overline{RD}$, $\overline{WR}$, $\overline{MS3–0}$, $\overline{IOMS}$, $\overline{BMS}$, $\overline{BHG}$, CLKOUT, PWDACK, DT0, DT1, FLOUT5–0, TIMEXP, $\overline{IACK}$, TD0, $\overline{EMU}$.

[6] Although specified for TTL outputs, all ADSP-21csp01 outputs are CMOS compatible and will drive to V_{DD} and GND, assuming no dc loads.

[7] Guaranteed but not tested.

[8] High Impedance pins: A0–A23, D0–D23, $\overline{BR}$, $\overline{BG}$, $\overline{RD}$, $\overline{WR}$, $\overline{MS3–0}$, $\overline{IOMS}$, $\overline{BMS}$, DT0, DT1, TFS0, TFS1, RFS0, RFS1, TCLK0, TCLK1, RCLK0, RCLK1, IAD0–IAD15, $\overline{IACK}$, TD0, $\overline{EMU}$.

[9] 0 V on $\overline{BR}$, CLKIN active (to force three-state condition).

[10] Idle refers to ADSP-21csp01 state of operation during execution of IDLE instruction. Deasserted pins are driven to V_{DD} or GND. Current reflects device operation with CLKOUT disabled.

[11] Current reflects device operation with no output loads.

[12] V_{IN} = 0.4 V and 2.4 V. For typical figures for supply current, see "Power Dissipation" section.

[13] See the *ADSP-21csp01 Preliminary User's Manual* for details.

[14] Applies to PQFP package type.

[15] Output pin capacitance is the capacitive load for any three-state output pin.

Specifications subject to change without notice.

ABSOLUTE MAXIMUM RATINGS*

Supply Voltage . . . . . –0.3 V to +7 V
Input Voltage . . . . . –0.3 V to V_{DD} + 0.3 V
Output Voltage Swing . . . . . –0.3 V to V_{DD} + 0.3 V
Operating Temperature Range (Ambient) . . . –40°C to +85°C
Storage Temperature Range . . . . . –65°C to +150°C
Lead Temperature (5 sec) PQFP . . . . . +280°C

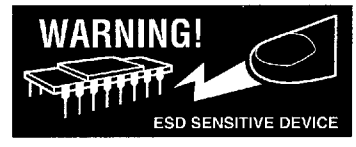
*Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

ESD SENSITIVITY

The ADSP-21csp01 is an ESD (electrostatic discharge) sensitive device. Electrostatic charges readily accumulate on the human body and equipment and can discharge without detection. Permanent damage may occur to devices subjected to high energy electrostatic discharges.

The ADSP-21csp01 features proprietary ESD protection circuitry to dissipate high energy discharges (Human Body Model). Per method 3015 of MIL-STD-883, the ADSP-21csp01 has been classified a Class 1 device.

Proper ESD precautions are recommended to avoid performance degradation or loss of functionality. Unused devices must be stored in conductive foam or shunts, and the foam should be discharged to the destination before devices are removed.



TIMING PARAMETERS

GENERAL NOTES

Use the exact timing information given. Do not attempt to derive parameters from the addition or subtraction of others. While addition or subtraction would yield meaningful results for an individual device, the values given in this data sheet reflect statistical variations and worst cases. Consequently, you cannot meaningfully add up parameters to derive longer times.

TIMING NOTES

Switching characteristics specify how the processor changes its signals. You have no control over this timing; it is dependent on the internal design. Timing requirements apply to signals that are controlled outside the processor, such as the data input for a read operation.

Timing requirements guarantee that the processor operates correctly with another device. Switching characteristics tell you what the device will do under a given circumstance. Also, use the switching characteristics to ensure any timing requirement of a device connected to the processor, such as memory, is satisfied.

MEMORY REQUIREMENTS

This chart links common memory device specification names and ADSP-21csp01 timing parameters for your convenience.

Parameter Name	Function	Common Memory Device Specification Name
t_{ASW}	A0–A23, $\overline{MS}_{3-0}$ Setup before $\overline{WR}$ Low	Address Setup to Write Start
t_{AW}	A0–A23, $\overline{MS}_{3-0}$ before $\overline{WR}$ Deasserted	Setup Address Setup to Write End
t_{WRA}	A0–A23, $\overline{MS}_{3-0}$ Hold after $\overline{WR}$ Deasserted	Address Hold Time
t_{DW}	Data Setup before $\overline{WR}$ High	Data Setup Time
t_{DH}	Data Hold after $\overline{WR}$ High	Data Hold Time
t_{RDD}	$\overline{RD}$ Low to Data Valid	$\overline{OE}$ to Data Valid
t_{AA}	A0–A23, $\overline{MS}_{3-0}$ $\overline{BMS}$ to Data Valid	Address Access Time

		50 MHz Min	50 MHz Max	Frequency Dependency Min	Frequency Dependency Max
CLOCK SIGNALS[1]					
Timing Requirements:					
t_{CKI}[1]	CLKIN Period	40			
t_{CKIL}	CLKIN Width Low	15			
t_{CKIH}	CLKIN Width High	15			
Switching Characteristics:					
t_{CKOH}	CLKIN High to CLKOUT High	0	15		
t_{CKL}	CLKOUT Width Low		12		
t_{CKH}	CLKOUT Width High		12		
CONTROL SIGNALS					
Timing Requirements:					
t_{RSP}	$\overline{\text{RESET}}$ Width Low	12_{TCK}[1, 2]			

NOTES

[1] t_{CK} is defined as 0.5 t_{CKI}. The ADSP-21csp01 uses an input clock with a frequency equal to half the instruction rate; a 25 MHz input clock (which is equivalent to 40 ns) yields a 20 ns processor cycle (equivalent to 50 MHz). t_{CK} values within the range of 0.5 t_{CKI} period should be substituted for all relevant timing parameters to obtain specification value. The DT derating allows specifications at other than 50 MHz processor frequencies (within the min-max range of the spec). DT is the difference between the actual processor cycle time and a processor cycle time of 20 ns (i.e., DT = t_{CK} – 20 ns).

[2] Applies after powerup sequence is complete. Internal phase lock loop requires no more than 2000 stable t_{CKI} cycles (not including crystal oscillator startup time).

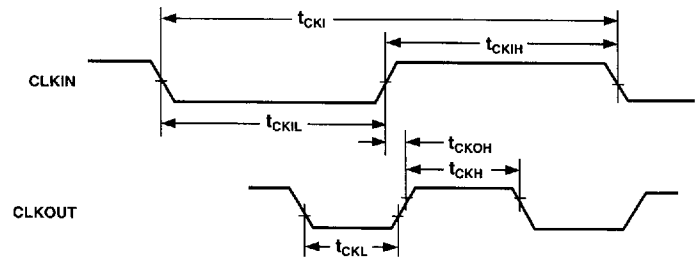


Figure 10. Clock Signals

		50 MHz Min	50 MHz Max	Frequency Dependency Min	Frequency Dependency Max
FLAGS AND INTERRUPTS					
Timing Requirements:					
t_{FIS}	FLINx Setup before CLKOUT Low[1, 2]	8		8 + 1/4 DT	
t_{FIH}	FLINx Hold after CLKOUT Low[1, 2]	0		0 – 1/4 DT	
t_{INTS}	$\overline{IRQx}$ Setup before CLKOUT Low[1, 3]	–2			
t_{INTH}	$\overline{IRQx}$ Hold after CLKOUT Low[1, 3]	10			
Switching Characteristics:					
t_{FOH}	FLOUTx Hold after CLKOUT High[4]	1			
t_{FOD}	FLOUTx Delay from CLKOUT High[4]		9		

NOTES
DT = t_{CK} – 20 ns
[1]If $\overline{IRQx}$ and FLINx inputs meet t_{FIS}, t_{FIH}, t_{INTS}, t_{INTH} setup and hold requirements, they will be recognized during the current clock cycle; otherwise, the signals will be recognized on the following cycle (Refer to "Interrupt Controller Operation" in the Program Control chapter of the User's Manual for further information on interrupt servicing).
[2] FLINx = FLIN0, FLIN1, FLIN2, FLIN3, FLIN4, FLIN5.
[3] $\overline{IRQx}$ = $\overline{IRQ0}$, $\overline{IRQ1}$, $\overline{IRQ2}$, $\overline{IRQ3}$.
[4] FLOUTx = FLOUT0, FLOUT1, FLOUT2, FLOUT3, FLOUT4, FLOUT5.

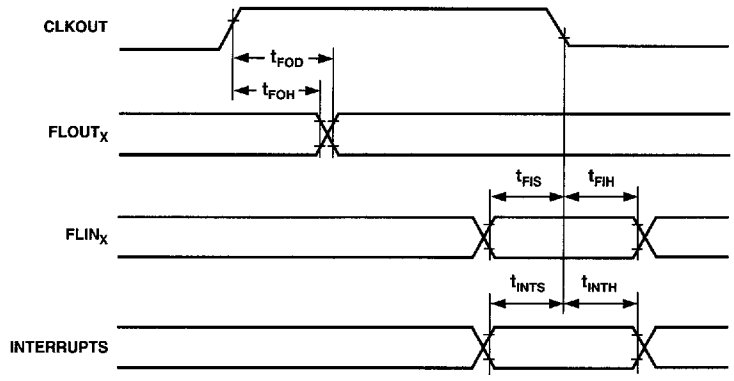


Figure 11. Flags and Interrupts

		50 MHz Min	50 MHz Max	Frequency Dependency Min	Frequency Dependency Max
MEMORY READ					
Timing Requirements:					
t_{RDD}	$\overline{RD}$ Low to Data Valid		8		8 + 5/6 DT + w
t_{AA}	Address to Data Valid[1]		10		10 + DT + w
t_{RDH}	Data Hold from $\overline{RD}$ High	4		0 + 1/6 DT	
t_{SAK}	Acknowledge Setup before CLKOUT Low[2]	2		2 + 1/4 DT	
t_{HAK}	Acknowledge Hold after CLKOUT Low[2]	4		4 + 1/4 DT	
Switching Characteristics:					
t_{RP}	$\overline{RD}$ Pulse Width	8		8 + 7/12 DT + w	
t_{CRD}	CLKOUT High to $\overline{RD}$ Low		8		8 + 1/6 DT
t_{ASR}	Address Setup before $\overline{RD}$ Low[1]	0		0 + 1/6 DT	
t_{RDA}	Address Hold after $\overline{RD}$ High[1]	3		3 + 1/4 DT	
t_{RWR}	$\overline{RD}$ High to $\overline{RD}$ or $\overline{WR}$ Low	7		7 + 5/12 DT	

NOTES
w = wait states × t_{CK}.
DT = t_{CK} – 20 ns.
[1]Address = A0–A23, $\overline{MS3–0}$, $\overline{IOMS}$, $\overline{BMS}$.
[2]Acknowledge = MACK, IOACK.

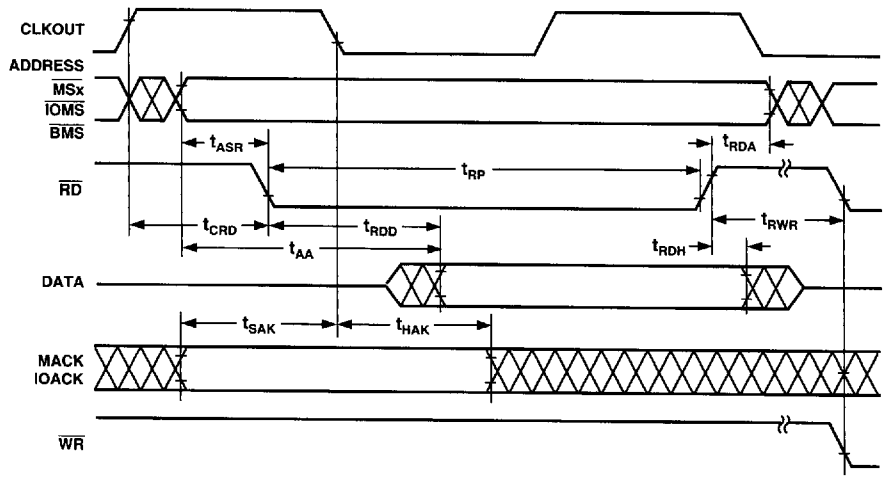


Figure 12. Memory Read

		50 MHz Min	50 MHz Max	Frequency Dependency Min	Frequency Dependency Max
MEMORY WRITE					
Timing Requirements:					
t_{SAK}	Acknowledge Setup before CLKOUT Low[1]	8		8 + 1/4 DT	
t_{HAK}	Acknowledge Hold after CLKOUT Low[1]	0		0	
Switching Characteristics:					
t_{DW}	Data Setup before $\overline{WR}$ High	7		7 + 3/4 DT + w	
t_{DH}	Data Hold after $\overline{WR}$ High	1		1 + 1/4 DT	
t_{WP}	$\overline{WR}$ Pulse Width	11		11 + 7/12 DT + w	
t_{WDE}	$\overline{WR}$ Low to Data Enabled	1		1 + 1/6 DT	
t_{ASW}	Address Setup before $\overline{WR}$ Low[2]	0		0 + 1/6 DT	
t_{DDR}	Data Disable before $\overline{RD}$ Low	2		2 + 1/6 DT	
t_{CWR}	CLKOUT High to $\overline{WR}$ Low		8		8 + 1/6 DT
t_{AW}	Address Setup before $\overline{WR}$ High[2]	11		11 + 3/4 DT + w	
t_{WRA}	Address Hold after $\overline{WR}$ High[1]	1		1 + 1/4 DT	
t_{WWR}	$\overline{WR}$ High to $\overline{RD}$ or $\overline{WR}$ Low	5		5 + 5/12 DT	
t_{DDW}	Data Disable from $\overline{WR}$ High		6		

NOTES
w = wait states × t_{CK}.
DT = t_{CK} – 20 ns.
[1]Acknowledge = MACK, IOACK.
[2]Address = A0–A23, $\overline{MS3–0}$, $\overline{IOMS}$, $\overline{BMS}$.

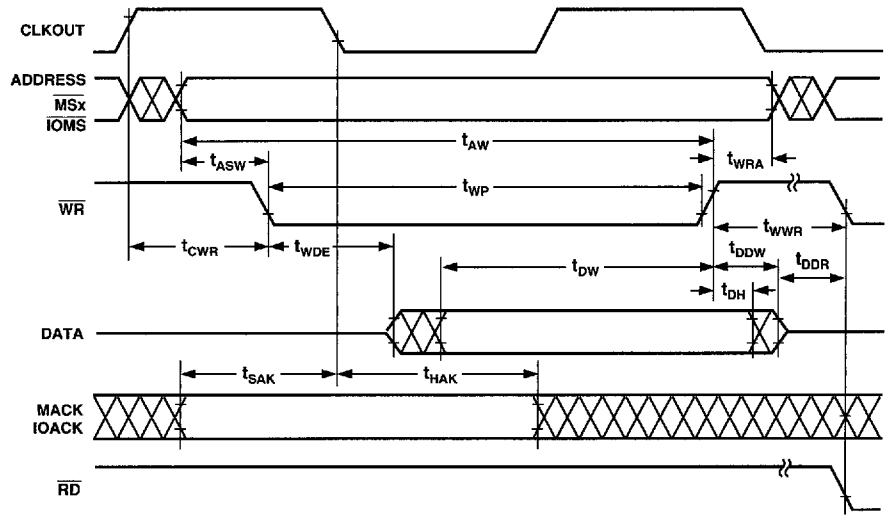


Figure 13. Memory Write

		50 MHz Min	50 MHz Max	Frequency Dependency Min	Frequency Dependency Max
BUS REQUEST, MASTER MODE					
Timing Requirements:					
t_{SBR}[1]	$\overline{BR}$ Setup before CLKOUT High	15		15 + 1/2 DT	
Switching Characteristics:					
t_{DBGL}	CLKOUT High to $\overline{BG}$ Low		17		17 + 1/2 DT
t_{DBGH}	CLKOUT High to $\overline{BG}$ High		17		17 + 1/2 DT
t_{DBHL}	CLKOUT High to $\overline{BGH}$ Low		6		
t_{DBHH}	CLKOUT High to $\overline{BGH}$ High		6		
t_{HZBG}	Memory Interface Disable to $\overline{BG}$ Low		–5		
t_{LZBG}	$\overline{BG}$ High to Memory Interface Enable		5		

NOTES
[1]$\overline{BR}$ is an asynchronous signal. If $\overline{BR}$ meets the setup hold requirements, it will be recognized during the current clock cycle; otherwise the signal will be recognized on the following cycle.

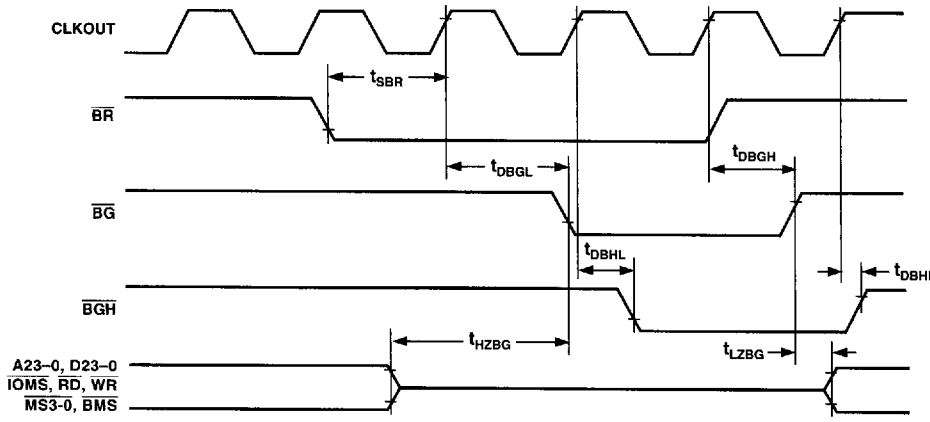


Figure 14. Bus Request, Master Mode

		50 MHz Min	50 MHz Max	Frequency Dependency Min	Frequency Dependency Max
BUS REQUEST, SLAVE MODE					
Timing Requirements:					
t_{SBG}	$\overline{BG}$ Setup before CLKOUT High	16		16 + 1/2 DT	
Switching Characteristics:					
t_{DBRL}	CLKOUT High to $\overline{BR}$ Low		7		
t_{DBRH}	CLKOUT High to $\overline{BR}$ High		6		
t_{DBHLS}	CLKOUT High to $\overline{BGH}$ Low		7		
t_{DBHHS}	CLKOUT High to $\overline{BGH}$ High		6		
t_{HZBR}	Memory Interface Disable to $\overline{BR}$ High	0			
t_{BGLZ}	$\overline{BG}$ Low to Memory Interface Enable	7			

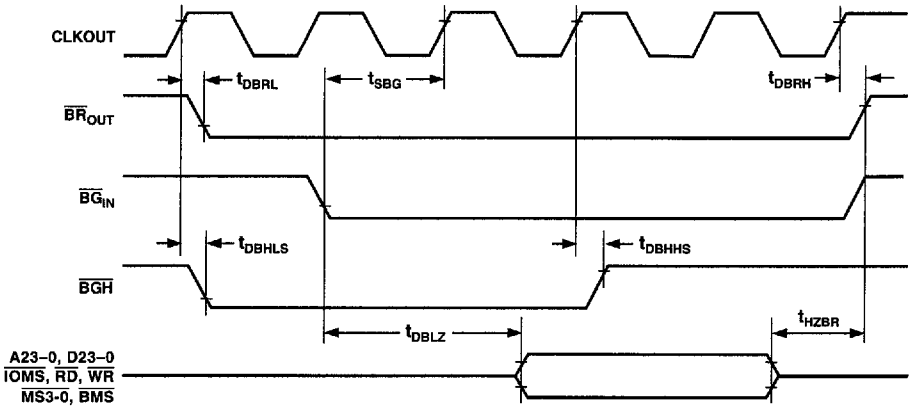


Figure 15. Bus Request, Slave Mode

		50 MHz		Frequency Dependency	
		Min	Max	Min	Max
SERIAL PORT DATA RECEIVE, INTERNAL CLOCK					
Timing Requirements:					
t_{RSCSI}	Input RFS, DR Setup before RCLK Low[1, 2, 3]	9			
t_{RSCHI}	Input RFS, DR Hold after RCLK Low[1, 2, 3]	0			
Switching Characteristics:					
t_{CCD}	RCLK Delay from CLKOUT High[1]		9		
t_{RFHI}	Output RFS Hold after RCLK High[1, 2]	0			
t_{RFDI}	Output RFS Delay from RCLK High[1, 2]		8		

NOTES
[1]Where RCLK = RCLK0, RCLK1.
[2]RFS = RFS0, RFS1.
[3]DR = DR0, DR1.

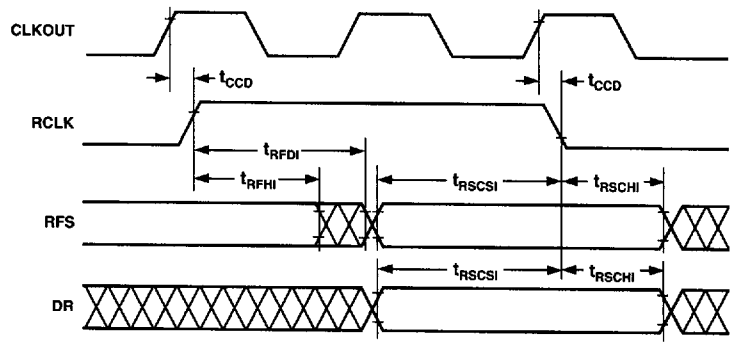


Figure 16. Serial Port Data Receive, Internal Clock

		50 MHz Min	50 MHz Max	Frequency Dependency Min	Frequency Dependency Max
SERIAL PORT DATA RECEIVE, EXTERNAL CLOCK					
Timing Requirements:					
t_{RSCSE}	Input RFS, DR Setup before RCLK Low[1, 2, 3]	2			
t_{RSCHE}	Input RFS, DR Hold after RCLK Low[1, 2, 3]	5			
t_{SCLKWE}	RCLK Pulse Width[1]	15			
t_{SCLKP}	RCLK Period[1]	40			
Switching Characteristics:					
t_{RFHE}	Output RFS Hold after RCLK High[1, 2]	4			
t_{RFDE}	Output RFS Delay from RCLK High[1, 2]		13		

NOTES
[1]Where RCLK = RCLK0, RCLK1.
[2]RFS = RFS0, RFS1.
[3]DR = DR0, DR1.

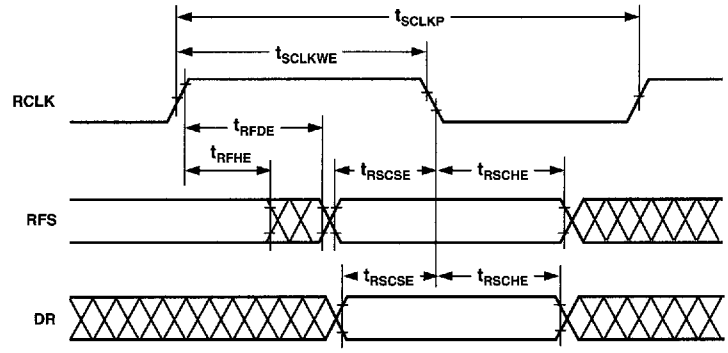


Figure 17. Serial Port Data Receive, External Clock

		50 MHz Min	50 MHz Max	Frequency Dependency Min	Frequency Dependency Max
SERIAL PORT DATA TRANSMIT, INTERNAL CLOCK					
Timing Requirements:					
t_{TSCSI}	Input TFS Setup before TCLK Low[1, 2]	9			
t_{TSCHI}	Input TFS Hold after TCLK Low[1, 2]	0			
Switching Characteristics:					
t_{CCD}	TCLK Delay from CLKOUT High[1]		9		
t_{SCLKWI}	TCLK Pulse Width[1]	15			
t_{TFHI}	Output TFS Hold after TCLK High[1, 2]	0			
t_{TFDI}	Output TFS Delay from TCLK High[1, 2]		5		
t_{SCDEI}	TCLK High to DT Enabled[1, 3]	0			
t_{SCDVI}	TCLK High to DT Valid[1, 3]		8		
t_{TDE}	TFS (Alternate Mode) to DT Enabled[2, 3]	–3			
t_{TDV}	TFS (Alternate Mode) to DT Valid[2, 3]		8		

NOTES
[1]TCLK = TCLK0, TCLK1.
[2]TFS = TFS0, TFS1.
[3]DT = DT0, DT1.

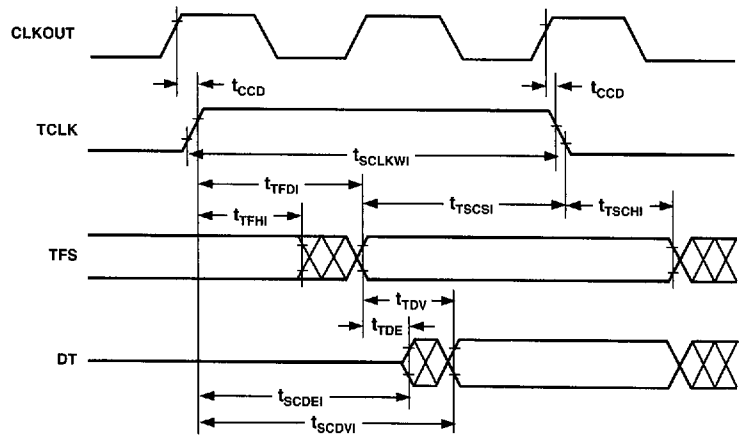


Figure 18. Serial Port Data Transmit, Internal Clock

		50 MHz		Frequency Dependency	
		Min	Max	Min	Max
SERIAL PORT DATA TRANSMIT, EXTERNAL CLOCK					
Timing Requirements:					
t_{TSCSE}	Input TFS Setup before TCLK Low[1, 2]	2			
t_{TSCHE}	Input TFS Hold after TCLK Low[1, 2]	3			
t_{SCLKWE}	TCLK Pulse Width[1]	15			
t_{SCLKP}	TCLK Period[1]	40			
Switching Characteristics:					
t_{TFHE}	Output TFS Hold after TCLK High[1, 2]	4			
t_{TFDE}	Output TFS Delay from TCLK High[1, 2]		12		
t_{SCDEE}	TCLK High to DT Enabled[1, 3]	8			
t_{SCDVE}	TCLK High to DT Valid[1, 3]		14		
t_{TDE}	TFS (Alternate Mode) to DT Enabled[2, 3]	–3			
t_{TDV}	TFS (Alternate Mode) to DT Valid[2, 3]		8		

NOTES
[1]TCLK = TCLK0, TCLK1.
[2]TFS = TFS0, TFS1.
[3]DT = DT0, DT1.

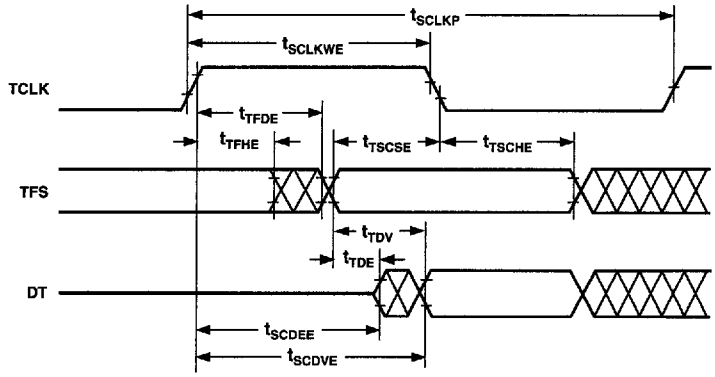


Figure 19. Serial Port Data Transmit, External Clock

		50 MHz Min	50 MHz Max	Frequency Dependency Min	Frequency Dependency Max
SERIAL PORT MULTICHANNEL, INTERNAL CLOCK					
Timing Requirements:					
t_{RSCSMI}	DR, RFS_{IN} Setup to RCLK Low	9			
t_{RSCHMI}	DR, RFS_{IN} Hold after RCLK Low	0			
Switching Characteristics:					
t_{CCD}	RCLK Delay from CLKOUT High[1]		9		
t_{RSCDEI}	RCLK High to DT Enabled[1, 2]	0			
t_{RSCDVI}	RCLK High to DT Valid[1, 2]		9		
t_{RSCDHI}	DT Hold after RCLK High[1, 2]	0			
t_{RSCDDI}	RCLK High to DT Disabled[1, 2]		3		
t_{RFHMI}	RFS_{OUT} Hold after RCLK High		3		
t_{RFDMI}	RFS_{OUT} delay after RCLK High	0			
t_{MTFHI}	TFS Hold after RCLK High[1, 3]	0			
t_{MTFDI}	RCLK High to TFS Valid[1, 3]		5		
t_{RDV}	RFS_{IN} to DT Valid[2, 4] (MFD = 0)		13		
t_{RFV}	RFS_{IN} to TFS Valid[3, 4] (MFD = 0)		13		

NOTES
[1]RCLK = RCLK0, RCLK1.
[2]DT = DT0, DT1.
[3]TFS = TFS0, TFS1.
[4]RFS = RFS0, RFS1.

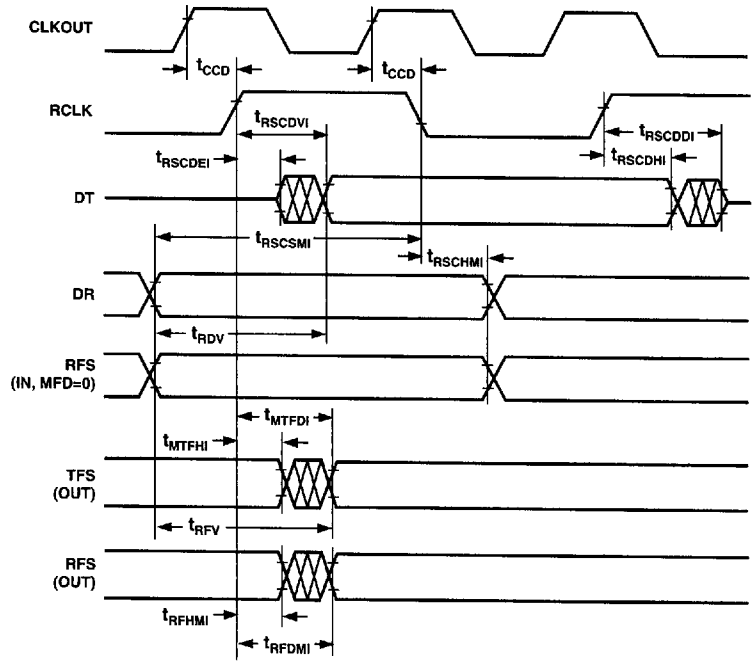


Figure 20. Serial Port Multichannel, Internal Clock

		50 MHz		Frequency Dependency	
		Min	Max	Min	Max
SERIAL PORT MULTICHANNEL, EXTERNAL CLOCK					
Timing Requirements:					
t_{RSCSME}	RFS_{IN}, DR Setup to RCLK Low	1			
t_{RSCHME}	RFS_{IN}, DR Hold after RCLK Low	5			
Switching Characteristics:					
t_{RSCDEE}	RCLK High to DT Enabled[1, 2]	6			
t_{RSCDVE}	RCLK High to DT Valid[1, 2]		16		
t_{RSCDHE}	DT Hold after RCLK High[1, 2]	0			
t_{RSCDDE}	RCLK High to DT Disabled[1, 2]		13		
t_{MTFHE}	TFS Hold after RCLK High[1, 2]	4			
t_{MTFDE}	RCLK High to TFS Valid[1, 3]		16		
t_{RDV}	RFS_{IN} to DT Valid[2, 4] (MFD = 0)		13		
t_{RFV}	RFS_{IN} to TFS Valid[3, 4] (MFD = 0)		13		
t_{RFHME}	RFS_{OUT} Hold after RCLK High	4			
t_{RFDME}	RFS_{OUT} Delay after RCLK High		15		

NOTES
[1]RCLK = RCLK0, RCLK1.
[2]DT = DT0, DT1.
[3]TFS = TFS0, TFS1.
[4]RFS = RFS0, RFS1.

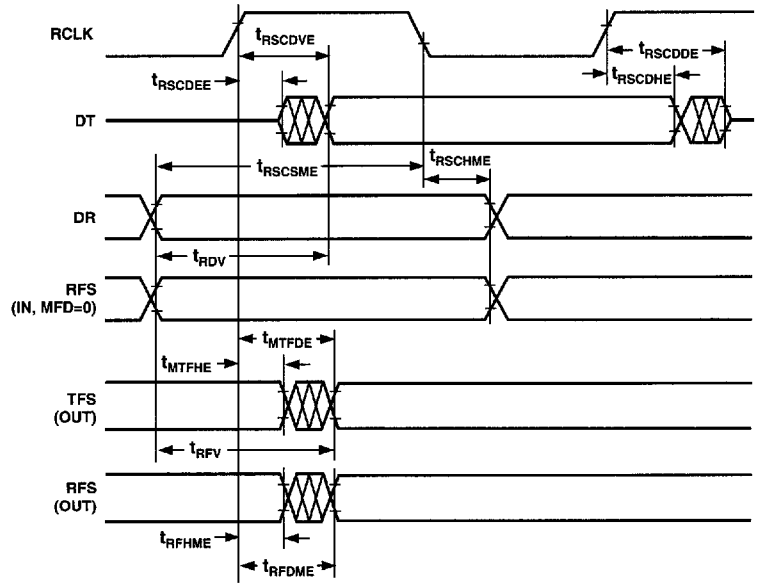


Figure 21. Serial Port Multichannel, External Clock

		50 MHz Min	50 MHz Max	Frequency Dependency Min	Frequency Dependency Max
IDMA ADDRESS LATCH					
Timing Requirements:					
t_{IKA}	$\overline{IACK}$ Low before Start of Address Latch[1]	0			
t_{IALP}	Duration of Address Latch[1, 2]	12			
t_{IASU}	IAD Setup before End of Address Latch[2]	3			
t_{IAH}	IAD Hold after End of Address Latch[2]	2			
t_{IPGSA}	IPGL Setup before End of Address Latch[2]	2			
t_{IPGHA}	IPGL Hold after End of Address Latch[2]	2			
t_{IALS}	Start of Write or Read after End of Address Latch[2, 3]	10			
t_{IALW}	Address Latch End before next Address Latch Start[1, 2]	5			

NOTES
When writing the full 24-bit page and address, the page must be written first (IPGL High), followed by the address (IPGL Low).
[1]Start of Address Latch = $\overline{IS}$ low and IAL high.
[2]End of Address Latch = $\overline{IS}$ high or IAL low.
[3]Start of Write or Read = $\overline{IS}$ low and $\overline{IWR}$ or $\overline{IRD}$ low.

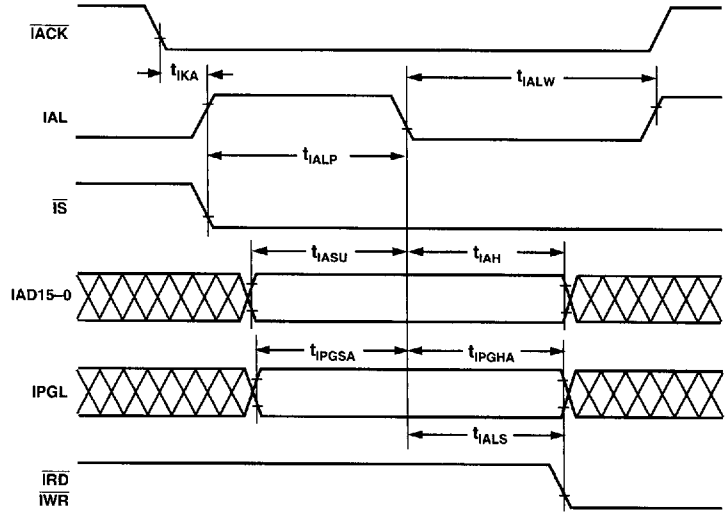


Figure 22. IDMA Address Latch

		50 MHz Min	50 MHz Max	Frequency Dependency Min	Frequency Dependency Max
IDMA READ					
Timing Requirements:					
t_{IKR}	$\overline{IACK}$ Low before Start of Read[1]	0			
t_{IRK}	$\overline{IACK}$ Low before End of Read[2]	0			
t_{IPGSR}	IPGL Setup before End of Read[2]	1			
t_{IPGHR}	IPGL Hold after End of Read[2]	3			
Switching Characteristics:					
t_{IKHR}	$\overline{IACK}$ High after Start of Read[1]		8		
t_{IKLR}	$\overline{IACK}$ Low after Start of Read[1]	65		65 + 3 DT	
t_{IRDE}	IAD Data Enabled after Start of Read[1]	0			
t_{IKDS}	IAD Data Setup before $\overline{IACK}$ Low	7		7 + 1/2 DT	
t_{IKDH}	IAD Data Hold after End of Read[2]	0			
t_{IKDD}	IAD Data Disabled after End of Read[2]		10		

NOTES
When reading 24-bit data, the 16 MSBs are read first (IPGL High), followed by the eight LSBs (IPGL Low) on IAD7–0.
[1]Start of Read = $\overline{IS}$ Low and $\overline{IRD}$ Low.
[2]End of Read = $\overline{IS}$ High or $\overline{IRD}$ High.

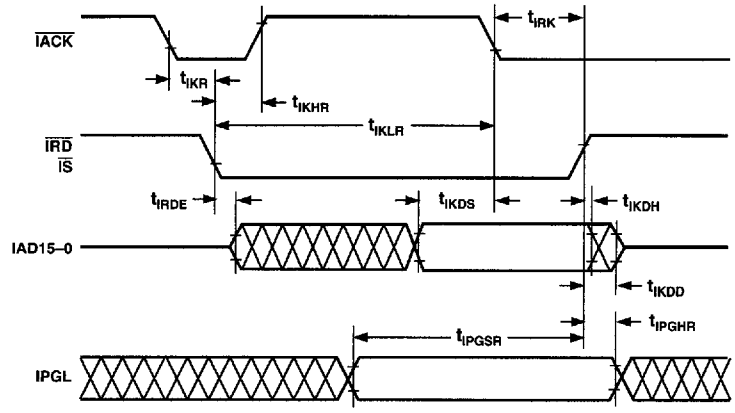


Figure 23. IDMA Read

		50 MHz Min	50 MHz Max	Frequency Dependency Min	Frequency Dependency Max
IDMA READ, BLOCK MODE					
Timing Requirements:					
t_{IKR}	$\overline{IACK}$ Low before Start of Read[1]	0			
t_{IRK}	$\overline{IACK}$ Low before End of Read[2]	12			
t_{IPGSR}	IPGL Setup before End of Read[2]	1			
t_{IPGHR}	IPGL Hold after End of Read[2]	3			
Switching Characteristics:					
t_{IKHR}	$\overline{IACK}$ High after Start of Read[1]		8		
t_{IKLRB}	$\overline{IACK}$ Low after Start of Read, Block Mode[1]	25		25 + DT	
t_{IRDE}	IAD Data Enabled after Start of Read[1]	0			
t_{IKDS}	IAD Data Setup before $\overline{IACK}$ Low	7		7 + 1/2 DT	
t_{IKDH}	IAD Data Hold after End of Read[2]	0			
t_{IKDD}	IAD Data Disabled after End of Read[2]		10		

NOTES
When reading 24-bit data, the 16 MSBs are read first (IPGL High), followed by the eight LSBs (IPGL Low) on IAD7–0.
[1]Start of Read = $\overline{IS}$ Low and $\overline{IRD}$ Low.
[2]End of Read = $\overline{IS}$ High or $\overline{IRD}$ High.

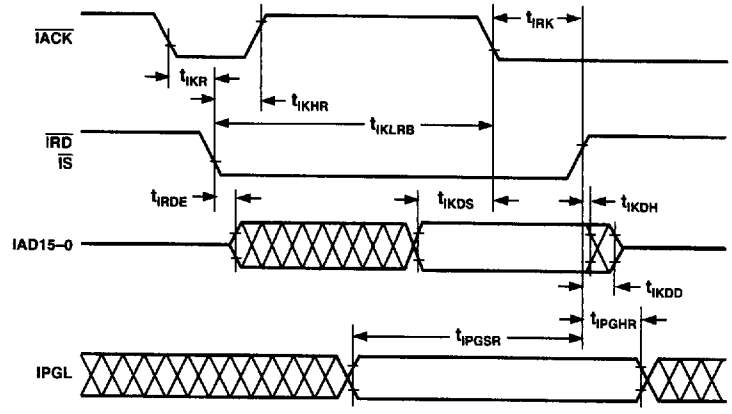


Figure 24. IDMA Read, Block Mode

		50 MHz Min	50 MHz Max	Frequency Dependency Min	Frequency Dependency Max
IDMA WRITE, LONG WRITE CYCLE					
Timing Requirements:					
t_{IWR}	$\overline{IACK}$ Low before End of Write[1]	0			
t_{IKW}	$\overline{IACK}$ Low before Start of Write[2]	0			
t_{IKSU}	IAD Data Setup before $\overline{IACK}$ Low	17			17 + 1/2 DT
t_{IKH}	IAD Data Hold after $\overline{IACK}$ Low	0			0 – 1/2 DT
t_{IPGSK}	IPGL Setup before $\overline{IACK}$ Low	17			17 + 1/2 DT
t_{IPGHK}	IPGL Hold after $\overline{IACK}$ Low	–10			–10 – 1/2 DT
Switching Characteristics:					
t_{IKHW}	Start of Write to $\overline{IACK}$ High[2]		8		
t_{IKLW}	Start of Write to $\overline{IACK}$ Low[2]	25		25 + DT	

NOTES
When writing the full 24-bit page and address, the page must be written first (IPGL High), followed by the address (IPGL Low).
[1]End of Write = $\overline{IS}$ High or $\overline{IWR}$ High.
[2]Start of Write = $\overline{IS}$ Low and $\overline{IWR}$ Low.

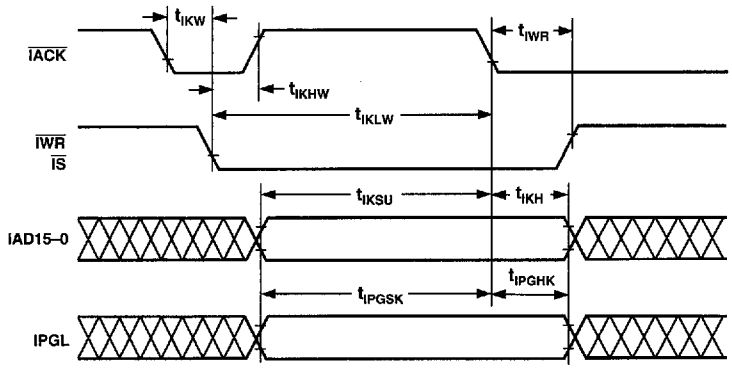


Figure 25. IDMA Write, Long Write Cycle

		50 MHz Min	50 MHz Max	Frequency Dependency Min	Frequency Dependency Max
IDMA WRITE, SHORT WRITE CYCLE					
Timing Requirements:					
t_{IKW}	$\overline{IACK}$ Low before Start of Write[1]	0			
t_{IWP}	Duration of Write	5			
t_{IDSUS}	IAD Data Setup before End of Write, Short[2]	0			
t_{IDHS}	IAD Data Hold after End of Write, Short[2]	3			
t_{IPGSWS}	IPGL Setup before End of Write, Short[2]	1			
t_{IPGHWS}	IPGL Hold after End of Write, Short[2]	3			
Switching Characteristics:					
t_{IKHW}	Start of Write to $\overline{IACK}$ High[1]		8		

NOTES
When writing the full 24-bit page and address, the page must be written first (IPGL High), followed by the address (IPGL Low).
[1]Start of Write = $\overline{IS}$ Low and $\overline{IWR}$ Low.
[2]End of Write = $\overline{IS}$ High or $\overline{IWR}$ High.

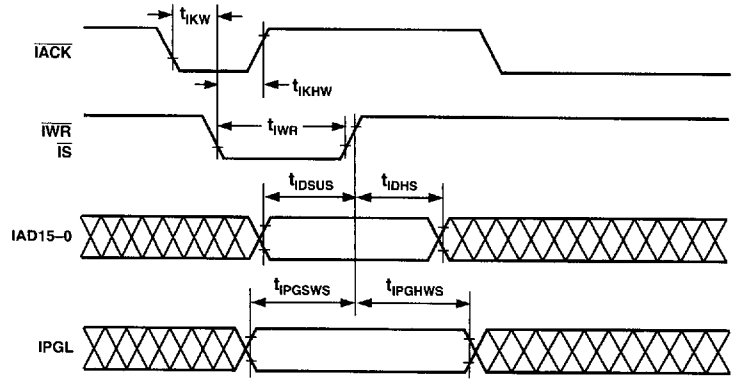


Figure 26. IDMA Write, Short Write Cycle

		50 MHz Min	50 MHz Max	Frequency Dependency Min	Frequency Dependency Max
IDMA ACKNOWLEDGE ENABLE					
Switching Characteristics:					
t_{SKE}	$\overline{IS}$ Low to $\overline{IACK}$ Enabled	0			
t_{SKD}	$\overline{IS}$ High to $\overline{IACK}$ Disabled		14		

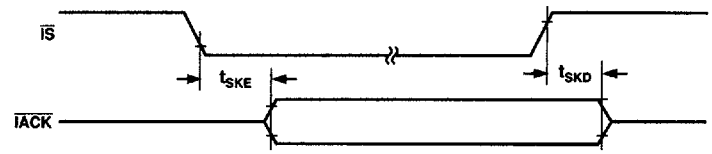


Figure 27. IDMA Acknowledge Enable

		50 MHz Min	50 MHz Max	Frequency Dependency Min	Frequency Dependency Max
TIMER					
Switching Characteristics:					
t_{DTEX}	TIMEXP Delay from CLKOUT High		7		

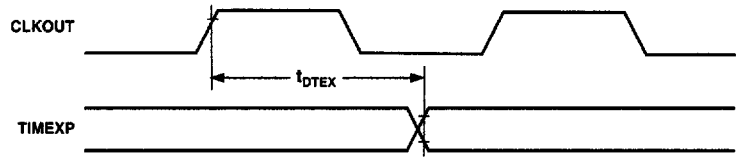


Figure 28. Timer

		50 MHz		Frequency Dependency	
		Min	Max	Min	Max
JTAG					
Timing Requirements:					
t_{TCK}	TCK Period	40			
t_{STAP}	TMS,TDI Setup before TCK High	3			
t_{HTAP}	TMS,TDI Hold after TCK High	4			
t_{SSYS}	System Input Setup before TCK High	18			
t_{HSYS}	System Input Hold after TCK High	0			
Switching Characteristics:					
t_{DTDO}	TDO Delay after TCK Low		10		
t_{DSYS}	System Output Delay after TCK Low		15		

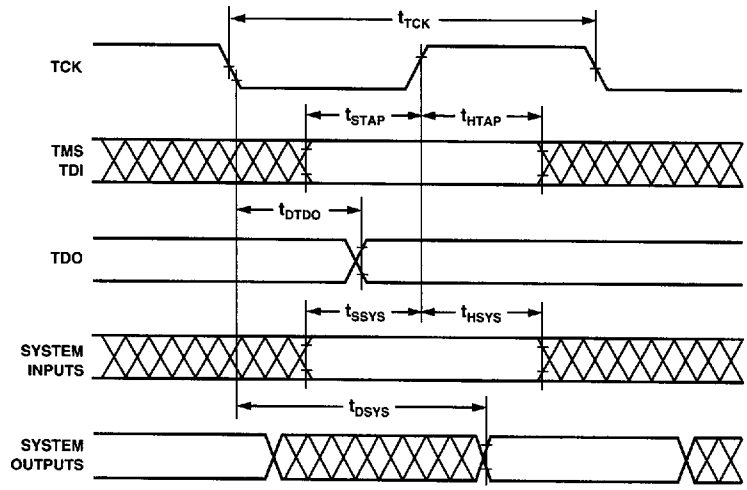


Figure 29. JTAG

ENVIRONMENTAL CONDITIONS

Ambient Temperature Rating:

$T_{amb} = T_{CASE} - (PD \times \Theta_{CA})$

T_{CASE} = Case temp in °C

PD = Power dissipation in W

Θ_{CA} = Thermal resistance (case-to-ambient)

Θ_{JA} = Thermal resistance (junction-to-ambient)

Θ_{JC} = Thermal resistance (junction-to-case)

Package	Θ_{JA}	Θ_{JC}	Θ_{CA}
PQFP	35°C/W	7°C/W	28°C/W

Power Dissipation

To determine total power dissipation in a specific application, the following formula should be applied for each output:

$$C \times V_{DD}^2 \times f$$

C = load capacitance, f = output switching frequency.

Example:
In an application where external data memory is used and no other outputs are active, power dissipation is calculated as follows:

Assumptions:

- External data memory is accessed every cycle with 50% of address pins switching.
- External data memory writes occur every other cycle with 50% of address pins switching.
- Each address and data pin has a 10 pF total load at the pin.
- The application operates at V_{DD} = 5.0 V and t_{CK} = 20 ns.

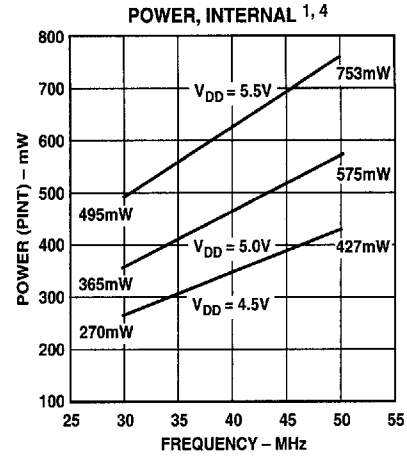
Total Power Dissipation = $P_{INT} + (C \times V_{DD}^2 \times f)$

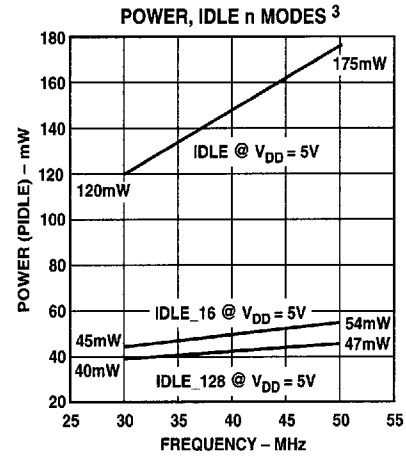
P_{INT} = internal power dissipation, from Power vs. Frequency graph.

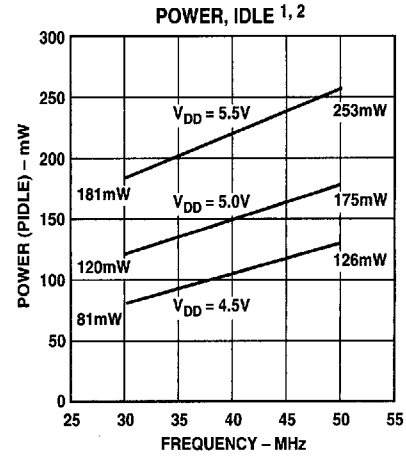
$C \times V_{DD}^2 \times f$ is calculated for each output:

	# of Pins	× C	× V_{DD}^2	× f	
Address, $\overline{MSB0}$	13	× 10 pF	× 5^2 V	× 50 MHz =	162.5 mW
Data Output, $\overline{WR}$	13	× 10 pF	× 5^2 V	× 25 MHz =	81.3 mW
$\overline{RD}$	1	× 10 pF	× 5^2 V	× 25 MHz =	6.25 mW
CLKOUT	1	× 10 pF	× 5^2 V	× 50 MHz =	12.5 mW
					262.55 mW

Total power dissipation for this example is P_{INT} + 262.55 mW.







NOTE:
VALID FOR ALL TEMPERATURE GRADES.

1 POWER REFLECTS DEVICE OPERATING WITH NO OUTPUT LOADS.

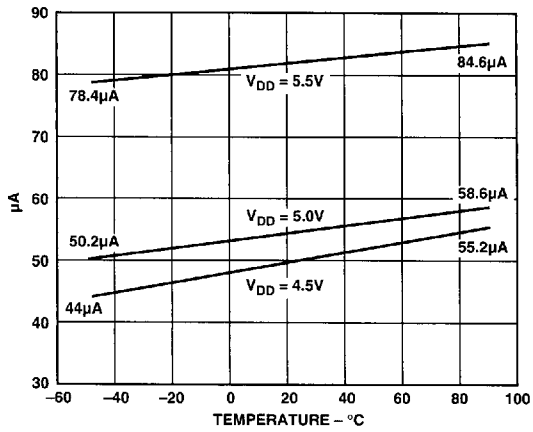
2 IDLE REFERS TO ADSP-21csp01 STATE OF OPERATION DURING EXECUTION OF IDLE INSTRUCTION. DEASSERTED PINS ARE DRIVEN TO V_{DD} OR GND.

3 TYPICAL POWER DISSIPATION AT 5.0V V_{DD} DURING EXECUTION OF IDLE n INSTRUCTION (CLOCK FREQUENCY REDUCTION).

4 I_{DD} MEASUREMENT TAKEN WITH ALL INSTRUCTIONS EXECUTING FROM INTERNAL MEMORY. 50% OF THE INSTRUCTIONS ARE MULTIFUNCTION (TYPE 1, 4, 12, 14), 30% ARE TYPE 2 AND TYPE 6, AND 20% ARE IDLE INSTRUCTIONS.

Figure 30. Power vs. Frequency

Figure 31 shows the power dissipation during Powerdown.



NOTE:
[1] REFLECTS ADSP-21csp01 OPERATING DURING POWERDOWN WITH CLKIN RUNNING.
[2] CURRENT REFLECTS DEVICE OPERATING WITH NO OUTPUT LOAD.
[3] FOR THE LOWEST POSSIBLE POWER CONSUMPTION, CLKIN SHOULD NOT BE RUNNING.

Figure 31. Power Supply Current (Typical) During Powerdown

CAPACITIVE LOADING

Figures 32 and 33 show the capacitive loading characteristics of the ADSP-21csp01. Figure 34 shows the equivalent device loading for ac measurements

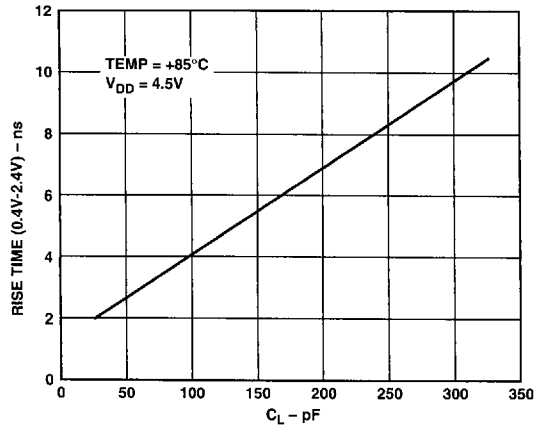


Figure 32. Typical Output Rise Time vs. Load Capacitance, C_L (at Maximum Ambient Operating Temperature)

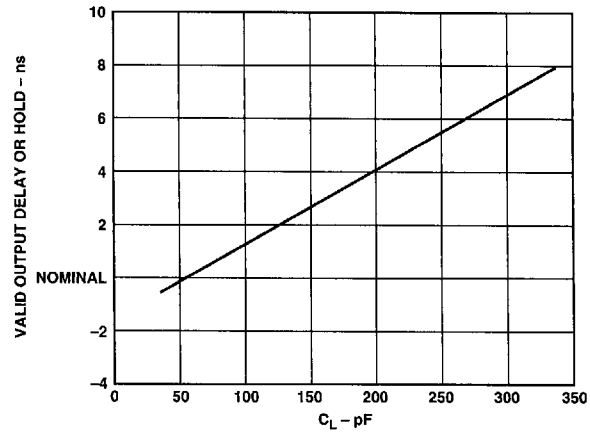


Figure 33. Typical Output Valid Delay or Hold vs. Load Capacitance, C_L (at Maximum Ambient Operating Temperature)

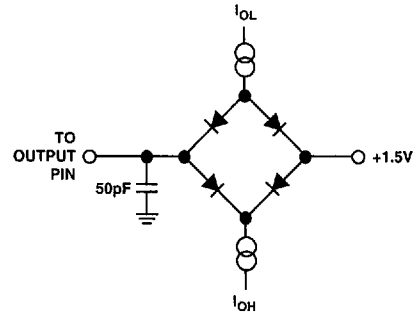


Figure 34. Equivalent Device Loading for AC Measurements (Includes all Fixtures)

TEST CONDITIONS

Output Disable Time

Output pins are considered to be disabled when they have stopped driving and started a transition from the measured output high or low voltage to a high impedance state (see Figure 35 for voltage reference levels for ac measurements). The output disable time (t_{DIS}) is the difference of $t_{MEASURED}$ and t_{DECAY}, as shown in the Output Enable/Disable diagram (Figure 36). The time is the interval from when a reference signal reaches a high or low voltage level to when the output voltages have changed by 0.5 V from the measured output high or low voltage. The decay time, t_{DECAY}, is dependent on the capacitive load, CL, and the current load, i_L, on the output pin. It can be approximated by the following equation:

$$t_{DECAY} = \frac{C_L \bullet 0.5V}{i_L}$$

from which

$$t_{DIS} = t_{MEASURED} - t_{DECAY}$$

is calculated. If multiple pins (such as the data bus) are disabled, the measurement value is that of the last pin to stop driving.

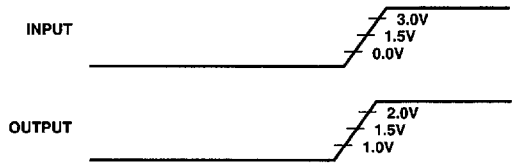


Figure 35. Voltage Reference Levels for AC Measurements (Except Output Enable/Disable)

Output Enable Time

Output pins are considered to be enabled when that have made a transition from a high-impedance state to when they start driving. The output enable time (t_{ENA}) is the interval from when a reference signal reaches a high or low voltage level to when a reference signal reaches a high or low voltage level to when the output has reached a specified high or low trip point, as shown in the Output Enable/Disable diagram (Figure 36). If multiple pins (such as the data bus) are enabled, the measurement value is that of the first pin to start driving.

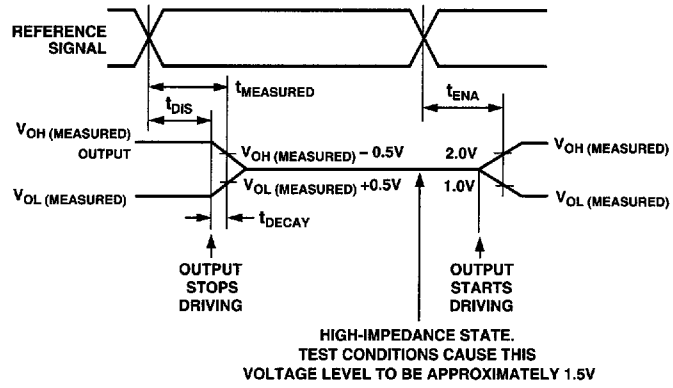
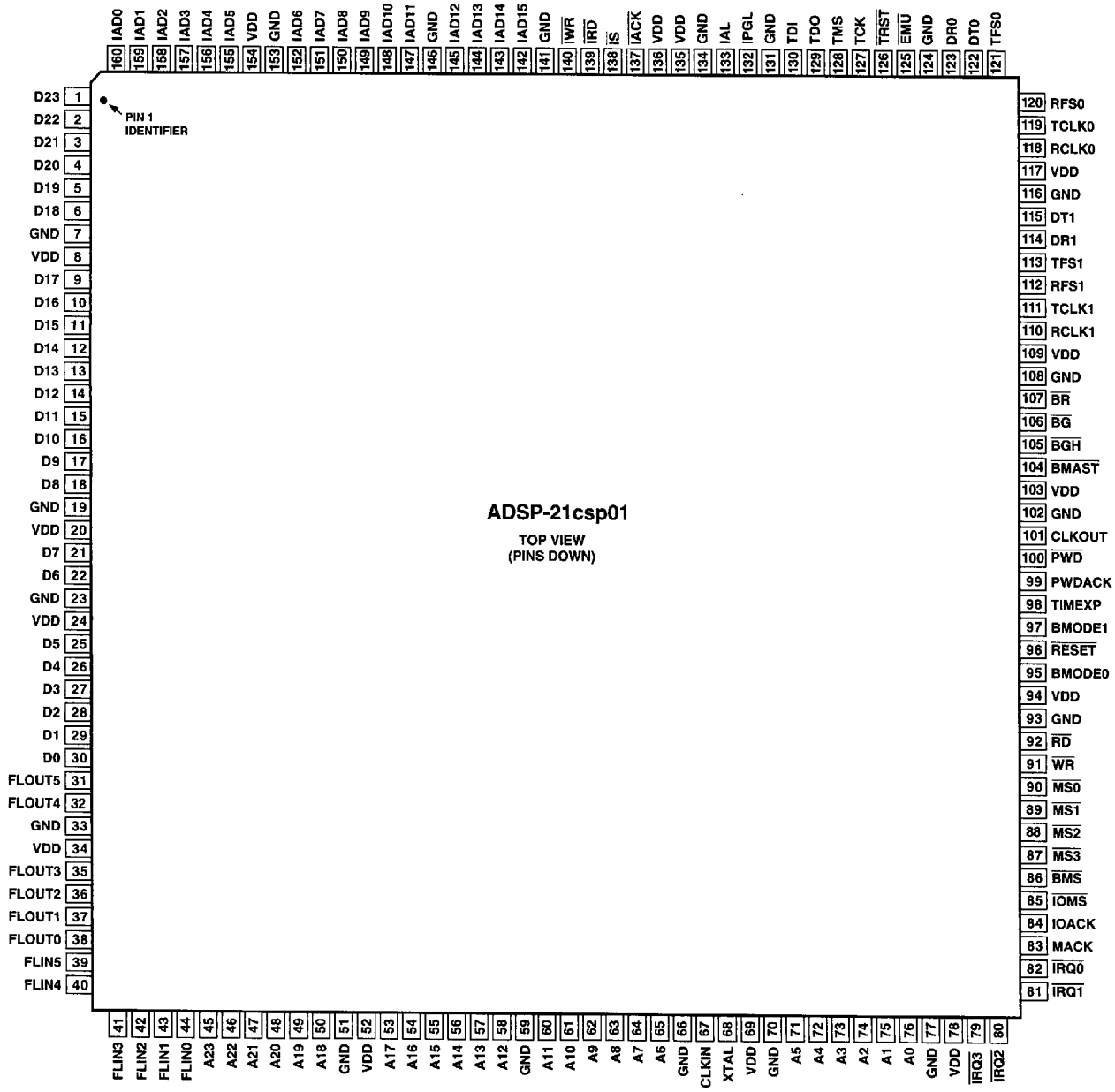


Figure 36. Output Enable/Disable

PIN CONFIGURATIONS

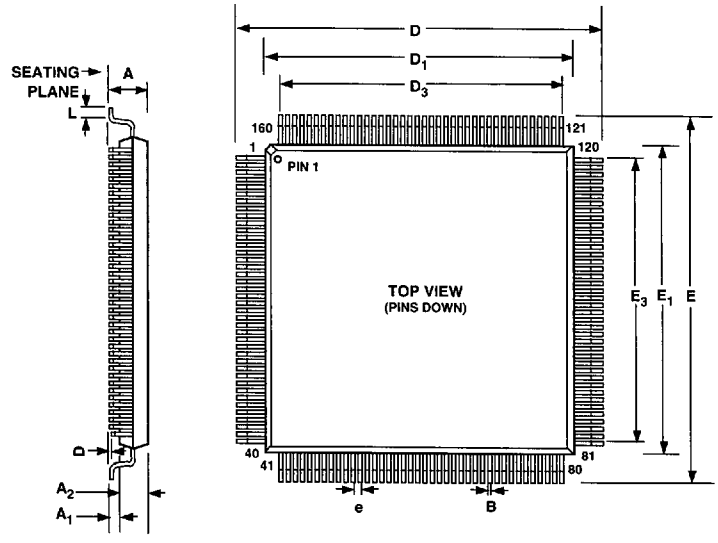


PQFP NUMBER	PIN NAME	PQFP NUMBER	PIN NAME	PQFP NUMBER	PIN NAME	PQFP NUMBER	PIN NAME
1	D23	41	FLIN3	81	$\overline{\text{IRQ1}}$	121	TFS0
2	D22	42	FLIN2	82	$\overline{\text{IRQ0}}$	122	DT0
3	D21	43	FLIN1	83	MACK	123	DR0
4	D20	44	FLIN0	84	IOACK	124	GND
5	D19	45	A23	85	$\overline{\text{IOMS}}$	125	$\overline{\text{EMU}}$
6	D18	46	A22	86	$\overline{\text{BMS}}$	126	$\overline{\text{TRST}}$
7	GND	47	A21	87	$\overline{\text{MS3}}$	127	TCK
8	VDD	48	A20	88	$\overline{\text{MS2}}$	128	TMS
9	D17	49	A19	89	$\overline{\text{MS1}}$	129	TDO
10	D16	50	A18	90	$\overline{\text{MS0}}$	130	TDI
11	D15	51	GND	91	$\overline{\text{WR}}$	131	GND
12	D14	52	VDD	92	$\overline{\text{RD}}$	132	IPGL
13	D13	53	A17	93	GND	133	IAL
14	D12	54	A16	94	VDD	134	GND
15	D11	55	A15	95	BMODE0	135	VDD
16	D10	56	A14	96	$\overline{\text{RESET}}$	136	VDD
17	D9	57	A13	97	BMODE1	137	$\overline{\text{IACK}}$
18	D8	58	A12	98	TIMEXP	138	$\overline{\text{IS}}$
19	GND	59	GND	99	PWDACK	139	$\overline{\text{IRD}}$
20	VDD	60	A11	100	$\overline{\text{PWD}}$	140	$\overline{\text{IWR}}$
21	D7	61	A10	101	CLKOUT	141	GND
22	D6	62	A9	102	GND	142	IAD15
23	GND	63	A8	103	VDD	143	IAD14
24	VDD	64	A7	104	$\overline{\text{BMAST}}$	144	IAD13
25	D5	65	A6	105	$\overline{\text{BGH}}$	145	IAD12
26	D4	66	GND	106	$\overline{\text{BG}}$	146	GND
27	D3	67	CLKIN	107	$\overline{\text{BR}}$	147	IAD11
28	D2	68	XTAL	108	GND	148	IAD10
29	D1	69	VDD	109	VDD	149	IAD9
30	D0	70	GND	110	RCLK1	150	IAD8
31	FLOUT5	71	A5	111	TCLK1	151	IAD7
32	FLOUT4	72	A4	112	RFS1	152	IAD6
33	GND	73	A3	113	TFS1	153	GND
34	VDD	74	A2	114	DR1	154	VDD
35	FLOUT3	75	A1	115	DT1	155	IAD5
36	FLOUT2	76	A0	116	GND	156	IAD4
37	FLOUT1	77	GND	117	VDD	157	IAD3
38	FLOUT0	78	VDD	118	RCLK0	158	IAD2
39	FLIN5	79	$\overline{\text{IRQ3}}$	119	TCLK0	159	IAD1
40	FLIN4	80	$\overline{\text{IRQ2}}$	120	RFS0	160	IAD0

OUTLINE DIMENSIONS

Dimensions shown in inches and (mm).

160-Lead Plastic Quad Flatpack (PQFP)



SYMBOL	MILLIMETERS			INCHES		
	MIN	TYP	MAX	MIN	TYP	MAX
A			4.07			0.160
A_1	0.25			0.010		
A_2	3.39	3.49	3.59	0.133	0.138	0.141
B	0.25	0.30	0.35	0.009	0.012	0.014
D, E	30.95	31.20	31.45	1.219	1.228	1.238
D_1, E_1	27.90	28.00	28.10	1.098	1.102	1.106
D_3, E_3	25.10	25.35	25.60	0.988	0.998	1.001
e	0.58	0.65	0.72	0.023	0.026	0.028
L	0.78	0.88	1.03	0.031	0.035	0.041
c			0.10			0.004

ORDERING GUIDE

Part Number*	Ambient Temperature Range	Instruction Rate (MHz)	Package Description	Package Option
ADSP21csp01KS200	0°C to +70°C	50	160-Lead PQFP	S-160
ADSP21csp01BS200	–40°C to +85°C	50	160-Lead PQFP	S-160

*S = Plastic Quad Flatpack

C2180–8–10/96

PRINTED IN U.S.A.