**Welcome to E-XFL.COM**

## Understanding Embedded - DSP (Digital Signal Processors)

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

## Applications of Embedded - DSP (Digital Signal Processors)

| Details | |
|---|---|
| Product Status | Active |
| Type | Fixed Point |
| Interface | Host Interface, Serial Port |
| Clock Rate | 40MHz |
| Non-Volatile Memory | External |
| On-Chip RAM | 20kB |
| Voltage - I/O | 5.00V |
| Voltage - Core | 5.00V |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 100-LQFP |
| Supplier Device Package | 100-LQFP (14x14) |
| Purchase URL | https://www.e-xfl.com/product-detail/analog-devices/adsp-2184bstz-160 |

The ADSP-21xx family DSPs contain a shadow bank register that is useful for single cycle context switching of the processor.

The ADSP-2184's flexible architecture and comprehensive instruction set allow the processor to perform multiple operations in parallel. In one processor cycle the ADSP-2184 can:

- Generate the next program address
- Fetch the next instruction
- Perform one or two data moves
- Update one or two data address pointers
- Perform a computational operation

This takes place while the processor continues to:
- Receive and transmit data through the two serial ports
- Receive or transmit data through the internal DMA port
- Receive or transmit data through the byte DMA port
- Decrement timer

**Development System**
The ADSP-2100 Family Development Software, a complete set of tools for software and hardware system development, supports the ADSP-2184. The System Builder provides a high level method for defining the architecture of systems under development. The Assembler has an algebraic syntax that is easy to program and debug. The Linker combines object files into an executable file. The Simulator provides an interactive instruction-level simulation with a reconfigurable user interface to display different portions of the hardware environment. A PROM Splitter generates PROM programmer compatible files. The C Compiler, based on the Free Software Foundation's GNU C Compiler, generates ADSP-2184 assembly source code. The source code debugger allows programs to be corrected in the C environment. The Runtime Library includes over 100 ANSI-standard mathematical and DSP-specific functions.

The EZ-KIT Lite is a hardware/software kit offering a complete development environment for the entire ADSP-21xx family: an ADSP-218x based evaluation board with PC monitor software plus Assembler, Linker, Simulator and PROM Splitter software. The ADSP-21xx EZ-KIT Lite is a low cost, easy to use hardware platform on which you can quickly get started with your DSP software design. The EZ-KIT Lite includes the following features:

- 33 MHz ADSP-2181
- Full 16-bit Stereo Audio I/O with AD1847 SoundPort® Codec
- RS-232 Interface to PC with Microsoft Windows® 3.1 Control Software
- EZ-ICE® Connector for Emulator Control
- DSP Demo Programs

The ADSP-218x EZ-ICE Emulator aids in the hardware debugging of an ADSP-2184 system. The emulator consists of hardware, host computer resident software, and the target board connector. The ADSP-2184 integrates on-chip emulation support with a 14-pin ICE-Port interface. This interface provides a simpler target board connection that requires fewer mechanical clearance considerations than other ADSP-2100 Family EZ-ICEs. The ADSP-2184 device need not be removed from the target system when using the EZ-ICE, nor are any adapters needed. Due to the small footprint of the EZ-ICE connector, emulation can be supported in final board designs.

The EZ-ICE performs a full range of functions, including:

- In-target operation
- Up to 20 breakpoints
- Single-step or full-speed operation
- Registers and memory values can be examined and altered
- PC upload and download functions
- Instruction-level emulation of program booting and execution
- Complete assembly and disassembly of instructions
- C source-level debugging

See Designing An EZ-ICE-Compatible Target System in the *ADSP-2100 Family EZ-Tools Manual* (ADSP-2181 sections), as well as the Target Board Connector for EZ-ICE Probe section of this data sheet, for the exact specifications of the EZ-ICE target board connector.

**Additional Information**
This data sheet provides a general overview of ADSP-2184 functionality. For additional information on the architecture and instruction set of the processor, refer to the *ADSP-2100 Family User's Manual, Third Edition*. For more information about the development tools, refer to the *ADSP-2100 Family Development Tools Data Sheet.*

## ARCHITECTURE OVERVIEW

The ADSP-2184 instruction set provides flexible data moves and multifunction (one or two data moves with a computation) instructions. Every instruction can be executed in a single processor cycle. The ADSP-2184 assembly language uses an algebraic syntax for ease of coding and readability. A comprehensive set of development tools supports program development.

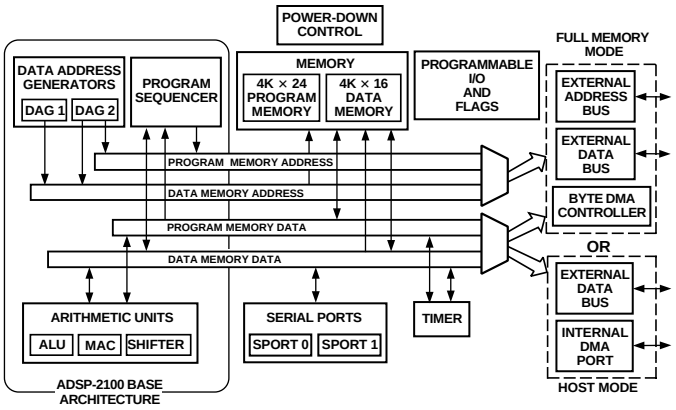



*Figure 1. Block Diagram*

Figure 1 is an overall block diagram of the ADSP-2184. The processor contains three independent computational units: the ALU, the multiplier/accumulator (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 and logic operations; division primitives are also supported. The MAC performs single-cycle multiply, multiply/add and multiply/subtract operations with 40 bits of accumulation. The shifter performs logical and arithmetic shifts, normalization, denormalization and derive exponent operations.

The shifter can be used to efficiently implement numeric format control including multiword and block floating-point representations.

The internal result (R) bus connects the computational units so the output of any unit may be the input of any unit on the next cycle.

A powerful program sequencer and two dedicated data address generators ensure efficient delivery of operands to these computational units. The sequencer supports conditional jumps, subroutine calls and returns in a single cycle. With internal loop counters and loop stacks, the ADSP-2184 executes looped code with zero overhead; no explicit jump instructions are required to maintain loops.

Two data address generators (DAGs) provide addresses for simultaneous dual operand fetches from data memory and program memory. Each DAG maintains and updates four address pointers. Whenever the pointer is used to access data (indirect addressing), it is post-modified by the value of one of four possible modify registers. A length value may be associated with each pointer to implement automatic modulo addressing for circular buffers.

Efficient data transfer is achieved with the use of five internal buses:

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

The two address buses (PMA and DMA) share a single external address bus, allowing memory to be expanded off-chip, and the two data buses (PMD and DMD) share a single external data bus. Byte memory space and I/O memory space also share the external buses.

Program memory can store both instructions and data, permitting the ADSP-2184 to fetch two operands in a single cycle, one from program memory and one from data memory. The ADSP-2184 can fetch an operand from program memory and the next instruction in the same cycle.

When configured in host mode, the ADSP-2184 has a 16-bit Internal DMA port (IDMA port) for connection to external systems. The IDMA port is made up of 16 data/address pins and five control pins. The IDMA port provides transparent, direct access to the DSPs on-chip program and data RAM.

An interface to low cost byte-wide memory is provided by the Byte DMA port (BDMA port). The BDMA port is bidirectional and can directly address up to four megabytes of external RAM or ROM for off-chip storage of program overlays or data tables.

The byte memory and I/O memory space interface supports slow memories and I/O memory-mapped peripherals with programmable wait state generation. External devices can gain control of external buses with bus request/grant signals ($\overline{BR}$, $\overline{BGH}$ and $\overline{BG}$). One execution mode (Go Mode) allows the ADSP-2184 to continue running from on-chip memory. Normal execution mode requires the processor to halt while buses are granted.

The ADSP-2184 can respond to eleven interrupts. There are up to six external interrupts (one edge-sensitive, two level-sensitive and three configurable) and seven internal interrupts generated by the timer, the serial ports (SPORTs), the Byte DMA port and the power-down circuitry. There is also a master $\overline{RESET}$ signal. The two serial ports provide a complete synchronous serial interface with optional companding in hardware and a wide variety of framed or frameless data transmit and receive modes of operation.

Each port can generate an internal programmable serial clock or accept an external serial clock.

The ADSP-2184 provides up to 13 general-purpose flag pins. The data input and output pins on SPORT1 can be alternatively configured as an input flag and an output flag. In addition, eight flags are programmable as inputs or outputs, and three flags are always outputs.

A programmable interval timer generates periodic interrupts. A 16-bit count register (TCOUNT) decrements every *n* processor cycle, where *n* is a scaling value stored in an 8-bit register (TSCALE). When the value of the count register reaches zero, an interrupt is generated and the count register is reloaded from a 16-bit period register (TPERIOD).

**Serial Ports**

The ADSP-2184 incorporates two complete synchronous serial ports (SPORT0 and SPORT1) for serial communications and multiprocessor communication.

Here is a brief list of the capabilities of the ADSP-2184 SPORTs. For additional information on Serial Ports, refer to the *ADSP-2100 Family User's Manual, Third Edition*.

- SPORTs are bidirectional and have a separate, double-buffered transmit and receive section.
- SPORTs can use an external serial clock or generate their own serial clock internally.
- SPORTs have independent framing for the receive and transmit sections. Sections run in a frameless mode or with frame synchronization signals internally or externally generated. Frame sync signals are active high or inverted, with either of two pulsewidths and timings.
- SPORTs support serial data word lengths from 3 to 16 bits and provide optional A-law and μ-law companding according to CCITT recommendation G.711.
- SPORT receive and transmit sections can generate unique interrupts on completing a data word transfer.
- SPORTs can receive and transmit an entire circular buffer of data with only one overhead cycle per data word. An interrupt is generated after a data buffer transfer.
- SPORT0 has a multichannel interface to selectively receive and transmit a 24- or 32-word, time-division multiplexed, serial bitstream.
- SPORT1 can be configured to have two external interrupts ($\overline{IRQ0}$ and $\overline{IRQ1}$) and the Flag In and Flag Out signals. The internally generated serial clock may still be used in this configuration.

## PIN DESCRIPTIONS

The ADSP-2184 is available in a 100-lead LQFP package. In order to maintain maximum functionality and reduce package size and pin count, some serial port, programmable flag, interrupt and external bus pins have dual, multiplexed functionality. The external bus pins are configured during RESET only, while serial port pins are software configurable during program execution. Flag and interrupt functionality is retained concurrently on multiplexed pins. In cases where pin functionality is reconfigurable, the default state is shown in plain text; alternate functionality is shown in italics.

When PMOVLAY is set to 1 or 2, external accesses occur at addresses 0x2000 through 0x3FFF. The external address is generated as shown in Table II.

**Table II.**

| PMOVLAY | Memory | A13 | A12:0 |
|---|---|---|---|
| 0 | Internal | Not Applicable | Not Applicable |
| 1 | External Overlay 1 | 0 | 13 LSBs of Address Between 0x2000 and 0x3FFF |
| 2 | External Overlay 2 | 1 | 13 LSBs of Address Between 0x2000 and 0x3FFF |

NOTE: Addresses 0x2000 through 0x3FFF should not be accessed when PMOVLAY = 0.

This organization provides for two external 8K overlay segments using only the normal 14 address bits, which allows for simple program overlays using one of the two external segments in place of the on-chip memory. Care must be taken in using this overlay space in that the processor core (i.e., the sequencer) does not take into account the PMOVLAY register value. For example, if a loop operation is occurring on one of the external overlays and the program changes to another external overlay or internal memory, an incorrect loop operation could occur. In addition, care must be taken in interrupt service routines as the overlay registers are not automatically saved and restored on the processor mode stack.

When Mode B = 1, booting is disabled and overlay memory is disabled the 4K internal PM cannot be accessed with MODE B = 1. Figure 5 shows the memory map in this configuration.

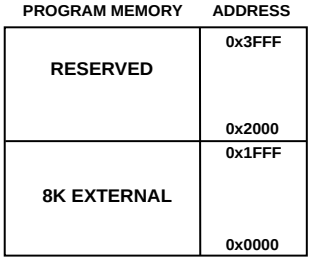



*Figure 5. Program Memory (Mode B = 1)*

**Data Memory**

The ADSP-2184 has 4K 16-bit words of internal data memory. In addition, the ADSP-2184 allows the use of 8K external memory overlays. Figure 6 shows the organization of the data memory.

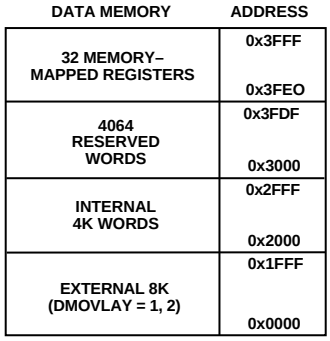



*Figure 6. Data Memory*

There are 4K words of memory accessible internally when the DMOVLAY register is set to 0. When DMOVLAY is set to 1 or 2, external accesses occur at addresses 0x0000 through 0x1FFF. The external address is generated as shown in Table III.

**Table III.**

| DMOVLAY | Memory | A13 | A12:0 |
|---|---|---|---|
| 0 | Internal | Not Applicable | Not Applicable |
| 1 | External Overlay 1 | 0 | 13 LSBs of Address Between 0x0000 and 0x1FFF |
| 2 | External Overlay 2 | 1 | 13 LSBs of Address Between 0x0000 and 0x1FFF |

This organization allows for two external 8K overlays using only the normal 14 address bits. All internal accesses complete in one cycle. Accesses to external memory are timed using the wait states specified by the DWAIT register.

**I/O Space (Full Memory Mode)**

The ADSP-2184 supports an additional external memory space called I/O space. This space is designed to support simple connections to peripherals or to bus interface ASIC data registers. I/O space supports 2048 locations. The lower eleven bits of the external address bus are used; the upper three bits are undefined. Two instructions were added to the core ADSP-2100 Family instruction set to read from and write to I/O memory space. The I/O space also has four dedicated three-bit wait state registers, IOWAIT0-3, that specify up to seven wait states to be automatically generated for each of four regions. The wait states act on address ranges as shown in Table IV.

**Table IV.**

| Address Range | Wait State Register |
|---|---|
| 0x000–0x1FF | IOWAIT0 |
| 0x200–0x3FF | IOWAIT1 |
| 0x400–0x5FF | IOWAIT2 |
| 0x600–0x7FF | IOWAIT3 |

**Composite Memory Select ($\overline{\text{CMS}}$)**

The ADSP-2184 has a programmable memory select signal that is useful for generating memory select signals for memories mapped to more than one space. The $\overline{\text{CMS}}$ signal is generated to have the same timing as each of the individual memory select signals ($\overline{\text{PMS}}$, $\overline{\text{DMS}}$, $\overline{\text{BMS}}$, $\overline{\text{IOMS}}$), but can combine their functionality.

Each bit in the CMSSEL register, when set, causes the $\overline{\text{CMS}}$ signal to be asserted when the selected memory select is asserted. For example, to use a 32K word memory to act as both program and data memory, set the $\overline{\text{PMS}}$ and $\overline{\text{DMS}}$ bits in the CMSSEL register and use the $\overline{\text{CMS}}$ pin to drive the chip select of the memory and use either $\overline{\text{DMS}}$ or $\overline{\text{PMS}}$ as the additional address bit.

The $\overline{\text{CMS}}$ pin functions as the other memory select signals, with the same timing and bus request logic. A 1 in the enable bit causes the assertion of the $\overline{\text{CMS}}$ signal at the same time as the selected memory select signal. All enable bits, except the $\overline{\text{BMS}}$ bit, default to 1 at reset.

**Byte Memory**
The byte memory space is a bidirectional, 8-bit-wide, external memory space used to store programs and data. Byte memory is accessed using the BDMA feature. The byte memory space consists of 256 pages, each of which is 16K × 8.

The byte memory space on the ADSP-2184 supports read and write operations as well as four different data formats. The byte memory uses data bits 15:8 for data. The byte memory uses data bits 23:16 and address bits 13:0 to create a 22-bit address. This allows up to a 4 meg × 8 (32 megabit) ROM or RAM to be used without glue logic. All byte memory accesses are timed by the BMWAIT register.

**Byte Memory DMA (BDMA, Full Memory Mode)**
The Byte memory DMA controller allows loading and storing of program instructions and data using the byte memory space. The BDMA circuit is able to access the byte memory space while the processor is operating normally and steals only one DSP cycle per 8-, 16- or 24-bit word transferred.

The BDMA circuit supports four different data formats that are selected by the BTYPE register field. The appropriate number of 8-bit accesses is done from the byte memory space to build the word size selected. Table V shows the data formats supported by the BDMA circuit.

**Table V.**

| BTYPE | Internal Memory Space | Word Size | Alignment |
|---|---|---|---|
| 00 | Program Memory | 24 | Full Word |
| 01 | Data Memory | 16 | Full Word |
| 10 | Data Memory | 8 | MSBs |
| 11 | Data Memory | 8 | LSBs |

Unused bits in the 8-bit data memory formats are filled with 0s. The BIAD register field is used to specify the starting address for the on-chip memory involved with the transfer. The 14-bit BEAD register specifies the starting address for the external byte memory space. The 8-bit BMPAGE register specifies the starting page for the external byte memory space. The BDIR register field selects the direction of the transfer. The 14-bit BWCOUNT register specifies the number of DSP words to transfer and initiates the BDMA circuit transfers.

BDMA accesses can cross page boundaries during sequential addressing. A BDMA interrupt is generated on the completion of the number of transfers specified by the BWCOUNT register. The BWCOUNT register is updated after each transfer so it can be used to check the status of the transfers. When it reaches zero, the transfers have finished and a BDMA interrupt is generated. The BMPAGE and BEAD registers must not be accessed by the DSP during BDMA operations.

The source or destination of a BDMA transfer will always be on-chip program or data memory, regardless of the values of Mode B, PMOVLAY or DMOVLAY.

When the BWCOUNT register is written with a nonzero value, the BDMA circuit starts executing byte memory accesses with wait states set by BMWAIT. These accesses continue until the count reaches zero. When enough accesses have occurred to create a destination word, it is transferred to or from on-chip memory. The transfer takes one DSP cycle. DSP accesses to external memory have priority over BDMA byte memory accesses.

The BDMA Context Reset bit (BCR) controls whether the processor is held off while the BDMA accesses are occurring. Setting the BCR bit to 0 allows the processor to continue operations. Setting the BCR bit to 1 causes the processor to stop execution while the BDMA accesses are occurring, to clear the context of the processor and start execution at address 0 when the BDMA accesses have completed.

**Internal Memory DMA Port (IDMA Port; Host Memory Mode)**
The IDMA Port provides an efficient means of communication between a host system and the ADSP-2184. The port is used to access the on-chip program memory and data memory of the DSP with only one DSP cycle per word overhead. The IDMA port cannot, however, be used to write to the DSP's memory-mapped control registers.

The IDMA port has a 16-bit multiplexed address and data bus and supports 24-bit program memory. The IDMA port is completely asynchronous and can be written to while the ADSP-2184 is operating at full speed.

The DSP memory address is latched and then automatically incremented after each IDMA transaction. An external device can therefore access a block of sequentially addressed memory by specifying only the starting address of the block. This increases throughput as the address does not have to be sent for each memory access.

IDMA Port access occurs in two phases. The first is the IDMA Address Latch cycle. When the acknowledge is asserted, a 14-bit address and 1-bit destination type can be driven onto the bus by an external device. The address specifies an on-chip memory location, the destination type specifies whether it is a DM or PM access. The falling edge of the IDMA address latch signal ($\overline{IAL}$) or the missing edge of the IDMA select signal ($\overline{IS}$) latches this value into the IDMAA register.

Once the address is stored, data can then either be read from or written to the ADSP-2184's on-chip memory. Asserting the select line ($\overline{IS}$) and the appropriate read or write line ($\overline{IRD}$ and $\overline{IWR}$ respectively) signals the ADSP-2184 that a particular transaction is required. In either case, there is a one-processor-cycle delay for synchronization. The memory access consumes one additional processor cycle.

Once an access has occurred, the latched address is automatically incremented and another access can occur.

Through the IDMAA register, the DSP can also specify the starting address and data format for DMA operation.

**Bootstrap Loading (Booting)**
The ADSP-2184 has two mechanisms to allow automatic loading of the internal program memory after reset. The method for booting is controlled by the Mode A, B and C configuration bits as shown in Table VI. These four states can be compressed into two-state bits by allowing an IDMA boot with Mode C = 1. However, three bits are used to ensure future compatibility with parts containing internal program memory ROM.

***BDMA Booting***
When the MODE pins specify BDMA booting, the ADSP-2184 initiates a BDMA boot sequence when $\overline{\text{RESET}}$ is released.

**Table VI. Boot Summary Table**

| MODE C | MODE B | MODE A | Booting Method |
|---|---|---|---|
| 0 | 0 | 0 | BDMA feature is used to load the first 32 program memory words from the byte memory space. Program execution is held off until all 32 words have been loaded. Chip is configured in Full Memory Mode. |
| 0 | 1 | 0 | No Automatic boot operations occur. Program execution starts at external memory location 0. Chip is configured in Full Memory Mode. BDMA can still be used but the processor does not automatically use or wait for these operations. |
| 1 | 0 | 0 | BDMA feature is used to load the first 32 program memory words from the byte memory space. Program execution is held off until all 32 words have been loaded. Chip is configured in Host Mode. Additional interface hardware is required. |
| 1 | 0 | 1 | IDMA feature is used to load any internal memory as desired. Program execution is held off until internal program memory location 0 is written to. Chip is configured in Host Mode. |

The BDMA interface is set up during reset to the following defaults when BDMA booting is specified: the BDIR, BMPAGE, BIAD and BEAD registers are set to 0; the BTYPE register is set to 0 to specify program memory 24-bit words; and the BWCOUNT register is set to 32. This causes 32 words of on-chip program memory to be loaded from byte memory. These 32 words are used to set up the BDMA to load in the remaining program code. The BCR bit is also set to 1, which causes program execution to be held off until all 32 words are loaded into on-chip program memory. Execution then begins at address 0.

The IDLE instruction can also be used to allow the processor to hold off execution while booting continues through the BDMA interface. For BDMA accesses while in Host Mode, the addresses to boot memory must be constructed externally to the ADSP-2184. The only memory address bit provided by the processor is A0.

***IDMA Port Booting***
The ADSP-2184 can also boot programs through its Internal DMA port. If Mode C = 1, Mode B = 0, and Mode A = 1, the ADSP-2184 boots from the IDMA port. The IDMA feature can load as much on-chip memory as desired. Program execution is held off until on-chip program memory location 0 is written to.

**Bus Request and Bus Grant**
The ADSP-2184 can relinquish control of the data and address buses to an external device. When the external device requires access to memory, it asserts the bus request ($\overline{\text{BR}}$) signal. If the ADSP-2184 is not performing an external memory access, it responds to the active BR input in the following processor cycle by:

- Three-stating the data and address buses and the $\overline{\text{PMS}}$, $\overline{\text{DMS}}$, $\overline{\text{BMS}}$, $\overline{\text{CMS}}$, $\overline{\text{IOMS}}$, $\overline{\text{RD}}$, $\overline{\text{WR}}$ output drivers,
- Asserting the bus grant ($\overline{\text{BG}}$) signal, and
- Halting program execution.

If Go Mode is enabled, the ADSP-2184 will not halt program execution until it encounters an instruction that requires an external memory access.

If the ADSP-2184 is performing an external memory access when the external device asserts the $\overline{\text{BR}}$ signal, it will not three-state the memory interfaces or assert the $\overline{\text{BG}}$ signal until the processor cycle after the access completes. The instruction does not need to be completed when the bus is granted. If a single instruction requires two external memory accesses, the bus will be granted between the two accesses.

When the $\overline{\text{BR}}$ signal is released, the processor releases the $\overline{\text{BG}}$ signal, reenables the output drivers and continues program execution from the point at which it stopped.

The bus request feature operates at all times, including when the processor is booting and when $\overline{\text{RESET}}$ is active.

The $\overline{\text{BGH}}$ pin is asserted when the ADSP-2184 is ready to execute an instruction but is stopped because the external bus is already granted to another device. The other device can release the bus by deasserting bus request. Once the bus is released, the ADSP-2184 deasserts $\overline{\text{BG}}$ and $\overline{\text{BGH}}$ and executes the external memory access.

**Flag I/O Pins**
The ADSP-2184 has eight general purpose programmable input/output flag pins. They are controlled by two memory mapped registers. The PFTYPE register determines the direction, 1 = output and 0 = input. The PFDATA register is used to read and write the values on the pins. Data being read from a pin configured as an input is synchronized to the ADSP-2184's clock. Bits that are programmed as outputs will read the value being output. The PF pins default to input during reset.

In addition to the programmable flags, the ADSP-2184 has five fixed-mode flags, FLAG_IN, FLAG_OUT, FL0, FL1 and FL2. FL0-FL2 are dedicated output flags. FLAG_IN and FLAG_OUT are available as an alternate configuration of SPORT1.

Note: Pins PF0, PF1 and PF2 are also used for device configuration during reset.

**INSTRUCTION SET DESCRIPTION**

The ADSP-2184 assembly language instruction set has an algebraic syntax that was designed for ease of coding and readability. The assembly language, which takes full advantage of the processor's unique architecture, offers the following benefits:

- The algebraic syntax eliminates the need to remember cryptic assembler mnemonics. For example, a typical arithmetic add instruction, such as AR = AX0 + AY0, resembles a simple equation.
- Every instruction assembles into a single, 24-bit word that can execute in a single instruction cycle.
- The syntax is a superset ADSP-2100 Family assembly language and is completely source and object code compatible with other family members. Programs may need to be relocated to utilize on-chip memory and conform to the ADSP-2184's interrupt vector and reset vector map.
- Sixteen condition codes are available. For conditional jump, call, return or arithmetic instructions, the condition can be checked and the operation executed in the same instruction cycle.
- Multifunction instructions allow parallel execution of an arithmetic instruction with up to two fetches or one write to processor memory space during a single instruction cycle.

**DESIGNING AN EZ-ICE-COMPATIBLE SYSTEM**

The ADSP-2184 has on-chip emulation support and an ICE-Port, a special set of pins that interface to the EZ-ICE. These features allow in-circuit emulation without replacing the target system processor by using only a 14-pin connection from the target system to the EZ-ICE. Target systems must have a 14-pin connector to accept the EZ-ICE's in-circuit probe, a 14-pin plug.

Issuing the chip reset command during emulation causes the DSP to perform a full chip reset, including a reset of its memory mode. Therefore, it is vital that the mode pins are set correctly PRIOR to issuing a chip reset command from the emulator user interface.

If using a passive method of maintaining mode information (as discussed in Setting Memory Modes), it does not matter that the mode information is latched by an emulator reset. However, if using the $\overline{\text{RESET}}$ pin as a method of setting the value of the mode pins, the effects of an emulator reset must be taken into consideration.

One method of ensuring that the values located on the mode pins is the one that is desired to construct a circuit like the one shown in Figure 7. This circuit will force the value located on the Mode A pin to Logic Low, regardless if it latched via the $\overline{\text{RESET}}$ or $\overline{\text{ERESET}}$ pin.

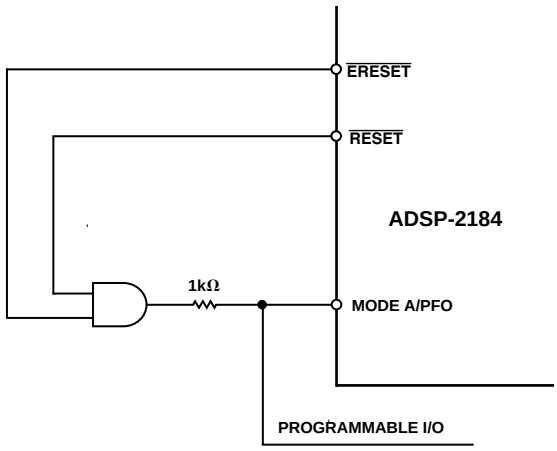



*Figure 7.*

See the *ADSP-2100 Family EZ-Tools* data sheet for complete information on ICE products.

The ICE-Port interface consists of the following ADSP-2184 pins:

| | | |
|---|---|---|
| $\overline{\text{EBR}}$ | $\overline{\text{EBG}}$ | $\overline{\text{ERESET}}$ |
| $\overline{\text{EMS}}$ | $\overline{\text{EINT}}$ | ECLK |
| ELIN | ELOUT | EE |

These ADSP-2184 pins must be connected only to the EZ-ICE connector in the target system. These pins have no function except during emulation, and do not require pull-up or pull-down resistors. The traces for these signals between the ADSP-2184 and the connector must be kept as short as possible, no longer than three inches.

The following pins are also used by the EZ-ICE:

| | |
|---|---|
| $\overline{\text{BR}}$ | $\overline{\text{BG}}$ |
| $\overline{\text{RESET}}$ | GND |

The EZ-ICE uses the EE (emulator enable) signal to take control of the ADSP-2184 in the target system. This causes the processor to use its $\overline{\text{ERESET}}$, $\overline{\text{EBR}}$ and $\overline{\text{EBG}}$ pins instead of the $\overline{\text{RESET}}$, $\overline{\text{BR}}$ and $\overline{\text{BG}}$ pins. The $\overline{\text{BG}}$ output is three-stated. These signals do not need to be jumper-isolated in your system.

The EZ-ICE connects to your target system via a ribbon cable and a 14-pin female plug. The female plug is plugged onto the 14-pin connector (a pin strip header) on the target board.

**Target Board Connector for EZ-ICE Probe**

The EZ-ICE connector (a standard pin strip header) is shown in Figure 8. You must add this connector to your target board design if you intend to use the EZ-ICE. Be sure to allow enough room in your system to fit the EZ-ICE probe onto the 14-pin connector.

# SPECIFICATIONS

## RECOMMENDED OPERATING CONDITIONS

| Parameter | B Grade Min | B Grade Max | Unit |
|---|---|---|---|
| $V_{DD}$ | 4.5 | 5.5 | V |
| $T_{AMB}$ | –40 | +85 | °C |

## ELECTRICAL CHARACTERISTICS

| Parameter | | Test Conditions | B Grade Min | B Grade Typ | B Grade Max | Unit |
|---|---|---|---|---|---|---|
| $V_{IH}$ | Hi-Level Input Voltage[1, 2] | @ $V_{DD}$ = max | 2.0 | | | V |
| $V_{IH}$ | Hi-Level CLKIN Voltage | @ $V_{DD}$ = max | 2.2 | | | V |
| $V_{IL}$ | Lo-Level Input Voltage[1, 3] | @ $V_{DD}$ = min | | | 0.8 | V |
| $V_{OH}$ | Hi-Level Output Voltage[1, 4, 5] | @ $V_{DD}$ = min, $I_{OH}$ = –0.5 mA | 2.4 | | | V |
| | | @ $V_{DD}$ = min, $I_{OH}$ = –100 µA[6] | $V_{DD}$ – 0.3 | | | V |
| $V_{OL}$ | Lo-Level Output Voltage[1, 4, 5] | @ $V_{DD}$ = min, $I_{OL}$ = 2 mA | | | 0.4 | V |
| $I_{IH}$ | Hi-Level Input Current[3] | @ $V_{DD}$ = max, $V_{IN}$ = $V_{DD}$ max | | | 10 | µA |
| $I_{IL}$ | Lo-Level Input Current[3] | @ $V_{DD}$ = max, $V_{IN}$ = 0 V | | | 10 | µA |
| $I_{OZH}$ | Three-State Leakage Current[7] | @ $V_{DD}$ = max, $V_{IN}$ = $V_{DD}$ max[8] | | | 10 | µA |
| $I_{OZL}$ | Three-State Leakage Current[7] | @ $V_{DD}$ = max, $V_{IN}$ = 0 V[8], $t_{CK}$ = 25 ns | | | 10 | µA |
| $I_{DD}$ | Supply Current (Idle)[9] | @ $V_{DD}$ = 5.0 | | 14 | | mA |
| $I_{DD}$ | Supply Current (Dynamic)[10, 11] | @ $V_{DD}$ = 5.0, $T_{AMB}$ = +25°C, $t_{CK}$ = 25 ns | | 60 | | mA |
| $C_I$ | Input Pin Capacitance[3, 6, 12] | @ $V_{IN}$ = 2.5 V, $f_{IN}$ = 1.0 MHz, $T_{AMB}$ = +25°C | | | 8 | pF |
| $C_O$ | Output Pin Capacitance[6, 7, 12, 13] | @ $V_{IN}$ = 2.5 V, $f_{IN}$ = 1.0 MHz, $T_{AMB}$ = +25°C | | | 8 | pF |

NOTES

[1] Bidirectional pins: D0–D23, RFS0, RFS1, SCLK0, SCLK1, TFS0, TFS1, A1–A13, PF0–PF7.

[2] Input only pins: $\overline{\text{RESET}}$, $\overline{\text{BR}}$, DR0, DR1, $\overline{\text{PWD}}$.

[3] Input only pins: CLKIN, $\overline{\text{RESET}}$, $\overline{\text{BR}}$, DR0, DR1, $\overline{\text{PWD}}$.

[4] Output pins: $\overline{\text{BG}}$, $\overline{\text{PMS}}$, $\overline{\text{DMS}}$, $\overline{\text{BMS}}$, $\overline{\text{IOMS}}$, $\overline{\text{CMS}}$, $\overline{\text{RD}}$, $\overline{\text{WR}}$, PWDACK, A0, DT0, DT1, CLKOUT, FL2-0, $\overline{\text{BGH}}$.

[5] Although specified for TTL outputs, all ADSP-2184 outputs are CMOS-compatible and will drive to $V_{DD}$ and GND, assuming no dc loads.

[6] Guaranteed but not tested.

[7] Three-statable pins: A0–A13, D0–D23, $\overline{\text{PMS}}$, $\overline{\text{DMS}}$, $\overline{\text{BMS}}$, $\overline{\text{IOMS}}$, $\overline{\text{CMS}}$, $\overline{\text{RD}}$, $\overline{\text{WR}}$, DT0, DT1, SCLK0, SCLK1, TFS0, TFS1, RFS0, RSF1, PF0–PF7.

[8] 0 V on $\overline{\text{BR}}$.

[9] Idle refers to ADSP-2184 state of operation during execution of IDLE instruction. Deasserted pins are driven to either $V_{DD}$ or GND.

[10] $I_{DD}$ measurement taken with all instructions executing from internal memory. 50% of the instructions are multifunction (types 1, 4, 5, 12, 13, 14), 30% are type 2 and type 6, and 20% are idle instructions.

[11] $V_{IN}$ = 0 V and 3 V. For typical figures for supply currents, refer to Power Dissipation section.

[12] Applies to LQFP package type.

[13] Output pin capacitance is the capacitive load for any three-stated output pin.

Specifications subject to change without notice.

# TIMING PARAMETERS

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| **Clock Signals and Reset** | | | | |
| *Timing Requirements*: | | | | |
| $t_{CKI}$ | CLKIN Period | 50 | 150 | ns |
| $t_{CKIL}$ | CLKIN Width Low | 20 | | ns |
| $t_{CKIH}$ | CLKIN Width High | 20 | | ns |
| *Switching Characteristics:* | | | | |
| $t_{CKL}$ | CLKOUT Width Low | 0.5 $t_{CK}$ – 7 | | ns |
| $t_{CKH}$ | CLKOUT Width High | 0.5 $t_{CK}$ – 7 | | ns |
| $t_{CKOH}$ | CLKIN High to CLKOUT High | 0 | 20 | ns |
| **Control Signals** | | | | |
| *Timing Requirements*: | | | | |
| $t_{RSP}$ | $\overline{\text{RESET}}$ Width Low[1] | 5 $t_{CK}$ | | ns |
| $t_{MS}$ | Mode Setup before $\overline{\text{RESET}}$ High | 2 | | ns |
| $t_{MH}$ | Mode Setup after $\overline{\text{RESET}}$ High | 5 | | ns |

NOTES
[1]Applies after power-up sequence is complete. Internal phase lock loop requires no more than 2000 CLKIN cycles assuming stable CLKIN (not including crystal oscillator start-up time).

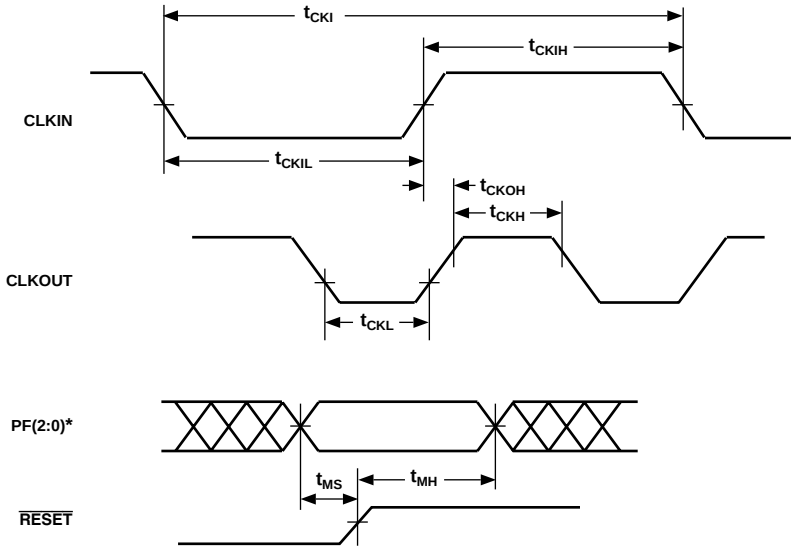



*Figure 9. Clock Signals*

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| **Bus Request–Bus Grant** | | | | |
| *Timing Requirements*: | | | | |
| $t_{BH}$ | $\overline{BR}$ Hold after CLKOUT High[1] | 0.25 $t_{CK}$ + 2 | | ns |
| $t_{BS}$ | $\overline{BR}$ Setup before CLKOUT Low[1] | 0.25 $t_{CK}$ + 17 | | ns |
| *Switching Characteristics*: | | | | |
| $t_{SD}$ | CLKOUT High to $\overline{xMS}$, $\overline{RD}$, $\overline{WR}$ Disable | | 0.25 $t_{CK}$ + 10 | ns |
| $t_{SDB}$ | $\overline{xMS}$, $\overline{RD}$, $\overline{WR}$ Disable to $\overline{BG}$ Low | 0 | | ns |
| $t_{SE}$ | $\overline{BG}$ High to $\overline{xMS}$, $\overline{RD}$, $\overline{WR}$ Enable | 0 | | ns |
| $t_{SEC}$ | $\overline{xMS}$, $\overline{RD}$, $\overline{WR}$ Enable to CLKOUT High | 0.25 $t_{CK}$ – 7 | | ns |
| $t_{SDBH}$ | $\overline{xMS}$, $\overline{RD}$, $\overline{WR}$ Disable to $\overline{BGH}$ Low[2] | 0 | | ns |
| $t_{SEH}$ | $\overline{BGH}$ High to $\overline{xMS}$, $\overline{RD}$, $\overline{WR}$ Enable[2] | 0 | | ns |

NOTES
$\overline{xMS}$ = $\overline{PMS}$, $\overline{DMS}$, $\overline{CMS}$, $\overline{IOMS}$, $\overline{BMS}$.
[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. Refer to the *ADSP-2100 Family User's Manual*, *Third Edition*, for $\overline{BR}$/$\overline{BG}$ cycle relationships.
[2]$\overline{BGH}$ is asserted when the bus is granted and the processor requires control of the bus to continue.

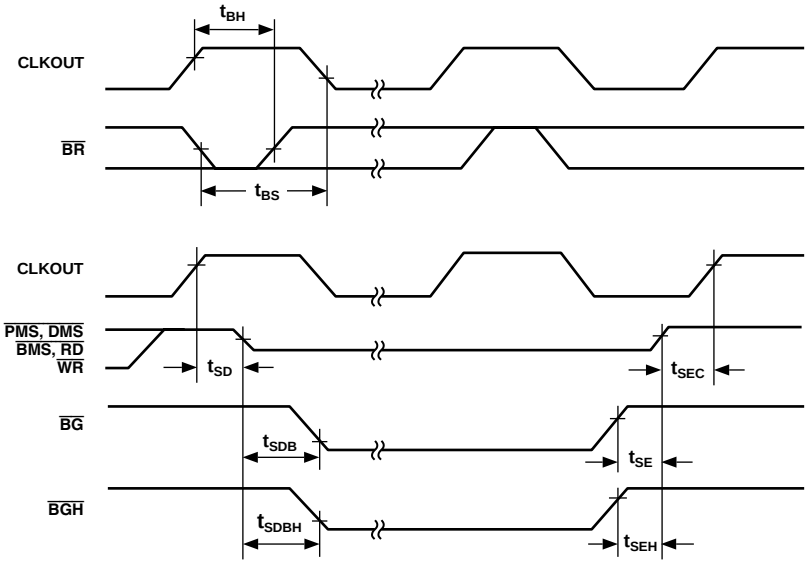



*Figure 11. Bus Request–Bus Grant*

# TIMING PARAMETERS

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| **Memory Read** | | | | |
| *Timing Requirements*: | | | | |
| $t_{RDD}$ | $\overline{RD}$ Low to Data Valid | | 0.5 $t_{CK}$ – 9 + w | ns |
| $t_{AA}$ | A0–A13, $\overline{xMS}$ to Data Valid | | 0.75 $t_{CK}$ – 12.5 + w | ns |
| $t_{RDH}$ | Data Hold from $\overline{RD}$ High | 1 | | ns |
| *Switching Characteristics*: | | | | |
| $t_{RP}$ | $\overline{RD}$ Pulsewidth | 0.5 $t_{CK}$ – 5 + w | | ns |
| $t_{CRD}$ | CLKOUT High to $\overline{RD}$ Low | 0.25 $t_{CK}$ – 5 | 0.25 $t_{CK}$ + 7 | ns |
| $t_{ASR}$ | A0–A13, $\overline{xMS}$ Setup before $\overline{RD}$ Low | 0.25 $t_{CK}$ – 6 | | ns |
| $t_{RDA}$ | A0–A13, $\overline{xMS}$ Hold after $\overline{RD}$ Deasserted | 0.25 $t_{CK}$ – 3 | | ns |
| $t_{RWR}$ | $\overline{RD}$ High to $\overline{RD}$ or $\overline{WR}$ Low | 0.5 $t_{CK}$ – 5 | | ns |

w = wait states × $t_{CK}$.
$\overline{xMS}$ = $\overline{PMS}$, $\overline{DMS}$, $\overline{CMS}$, $\overline{IOMS}$, $\overline{BMS}$.

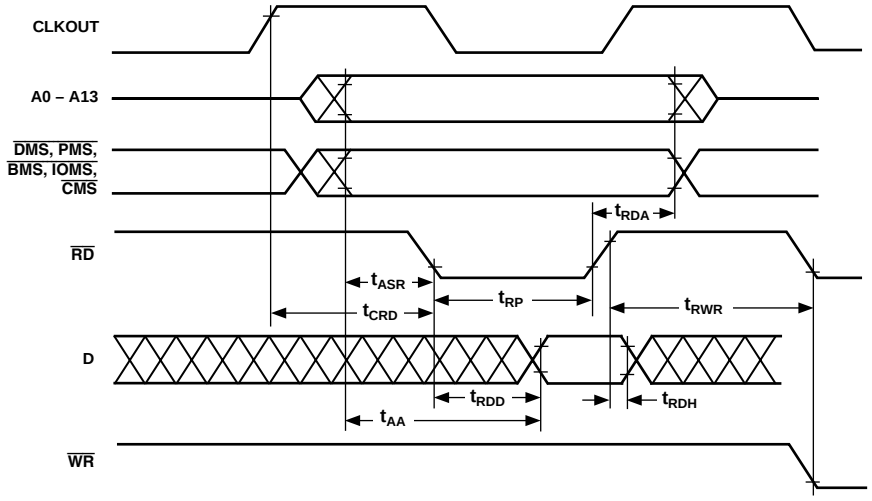



*Figure 12. Memory Read*

# TIMING PARAMETERS

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| **Serial Ports** | | | | |
| *Timing Requirements*: | | | | |
| $t_{SCK}$ | SCLK Period | 50 | | ns |
| $t_{SCS}$ | DR/TFS/RFS Setup before SCLK Low | 4 | | ns |
| $t_{SCH}$ | DR/TFS/RFS Hold after SCLK Low | 8 | | ns |
| $t_{SCP}$ | $SCLK_{IN}$ Width | 20 | | ns |
| *Switching Characteristics*: | | | | |
| $t_{CC}$ | CLKOUT High to $SCLK_{OUT}$ | 0.25 $t_{CK}$ | 0.25 $t_{CK}$ + 10 | ns |
| $t_{SCDE}$ | SCLK High to DT Enable | 0 | | ns |
| $t_{SCDV}$ | SCLK High to DT Valid | | 15 | ns |
| $t_{RH}$ | $TFS/RFS_{OUT}$ Hold after SCLK High | 0 | | ns |
| $t_{RD}$ | $TFS/RFS_{OUT}$ Delay from SCLK High | | 15 | ns |
| $t_{SCDH}$ | DT Hold after SCLK High | 0 | | ns |
| $t_{TDE}$ | TFS (Alt) to DT Enable | 0 | | ns |
| $t_{TDV}$ | TFS (Alt) to DT Valid | | 14 | ns |
| $t_{SCDD}$ | SCLK High to DT Disable | | 15 | ns |
| $t_{RDV}$ | RFS (Multichannel, Frame Delay Zero) to DT Valid | | 15 | ns |

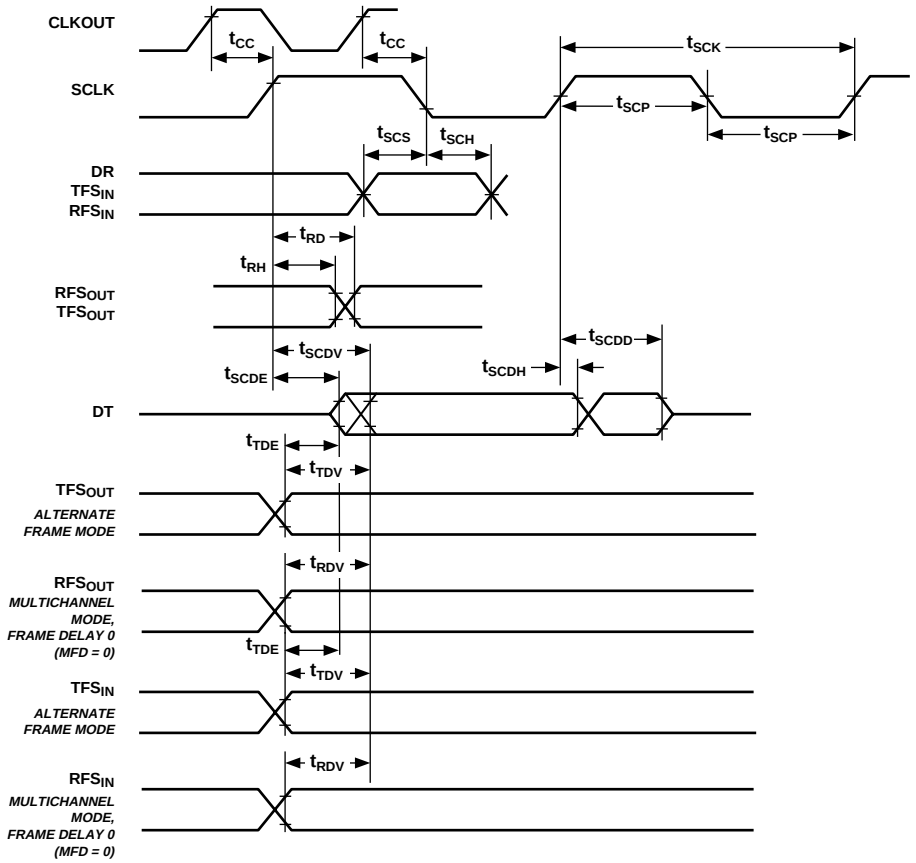



*Figure 14. Serial Ports*

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| **IDMA Address Latch** | | | | |
| *Timing Requirements*: | | | | |
| $t_{IALP}$ | Duration of Address Latch[1, 2] | 10 | | ns |
| $t_{IASU}$ | IAD15–0 Address Setup before Address Latch End[2] | 5 | | ns |
| $t_{IAH}$ | IAD15–0 Address Hold after Address Latch End[2] | 3 | | ns |
| $t_{IKA}$ | $\overline{IACK}$ Low before Start of Address Latch[2, 3] | 0 | | ns |
| $t_{IALS}$ | Start of Write or Read after Address Latch End[2, 3] | 3 | | ns |

NOTES
[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 $\overline{IWR}$ Low or $\overline{IRD}$ Low.

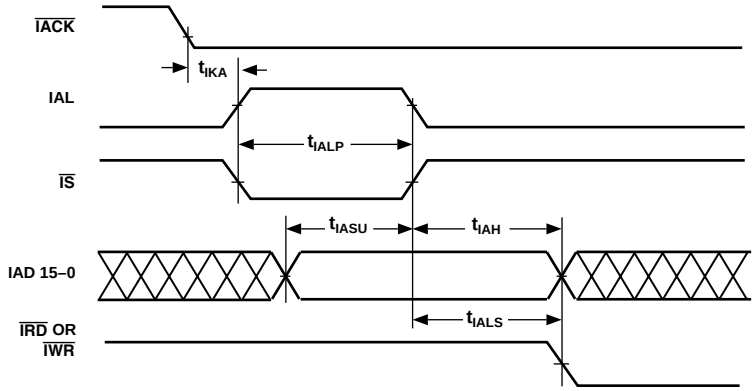



*Figure 15. IDMA Address Latch*

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| **IDMA Write, Long Write Cycle** | | | | |
| *Timing Requirements*: | | | | |
| $t_{IKW}$ | $\overline{IACK}$ Low before Start of Write[1] | 0 | | ns |
| $t_{IKSU}$ | IAD15–0 Data Setup before $\overline{IACK}$ Low[2, 3, 4] | 0.5 $t_{CK}$ + 10 | | ns |
| $t_{IKH}$ | IAD15–0 Data Hold after $\overline{IACK}$ Low[2, 3, 4] | 2 | | ns |
| *Switching Characteristics*: | | | | |
| $t_{IKLW}$ | Start of Write to $\overline{IACK}$ Low[4] | 1.5 $t_{CK}$ | | ns |
| $t_{IKHW}$ | Start of Write to $\overline{IACK}$ High | | 15 | ns |

NOTES
[1]Start of Write = $\overline{IS}$ Low and $\overline{IWR}$ Low.
[2]If Write Pulse ends before $\overline{IACK}$ Low, use specifications $t_{IDSU}$, $t_{IDH}$.
[3]If Write Pulse ends after $\overline{IACK}$ Low, use specifications $t_{IKSU}$, $t_{IKH}$.
[4]This is the earliest time for $\overline{IACK}$ Low from Start of Write. For IDMA Write cycle relationships, please refer to the *ADSP-2100 Family User's Manual*, *Third Edition*.

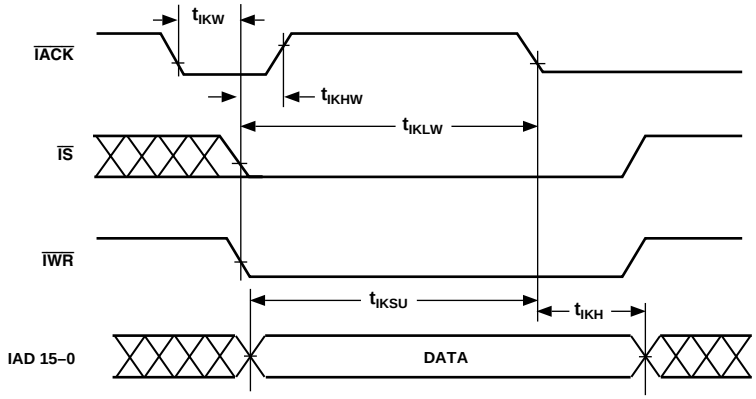



*Figure 17. IDMA Write, Long Write Cycle*

# TIMING PARAMETERS

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| **IDMA Read, Long Read Cycle** | | | | |
| *Timing Requirements*: | | | | |
| $t_{IKR}$ | $\overline{IACK}$ Low before Start of Read[1] | 0 | | ns |
| $t_{IRK}$ | End of Read after $\overline{IACK}$ Low | 2 | | ns |
| *Switching Characteristics*: | | | | |
| $t_{IKHR}$ | $\overline{IACK}$ High after Start of Read[1] | | 15 | ns |
| $t_{IKDS}$ | IAD15–0 Data Setup before $\overline{IACK}$ Low | 0.5 $t_{CK}$ – 10 | | ns |
| $t_{IKDH}$ | IAD15–0 Data Hold after End of Read[2] | 0 | | ns |
| $t_{IKDD}$ | IAD15–0 Data Disabled after End of Read[2] | | 10 | ns |
| $t_{IRDE}$ | IAD15–0 Previous Data Enabled after Start of Read | 0 | | ns |
| $t_{IRDV}$ | IAD15–0 Previous Data Valid after Start of Read | | 15 | ns |
| $t_{IRDH1}$ | IAD15–0 Previous Data Hold after Start of Read (DM/PM1)[3] | 2 $t_{CK}$ – 5 | | ns |
| $t_{IRDH2}$ | IAD15–0 Previous Data Hold after Start of Read (PM2)[4] | $t_{CK}$ – 5 | | ns |

NOTES
[1]Start of Read = $\overline{IS}$ Low and $\overline{IRD}$ Low.
[2]End of Read = $\overline{IS}$ High or $\overline{IRD}$ High.
[3]DM read or first half of PM read.
[4]Second half of PM read.

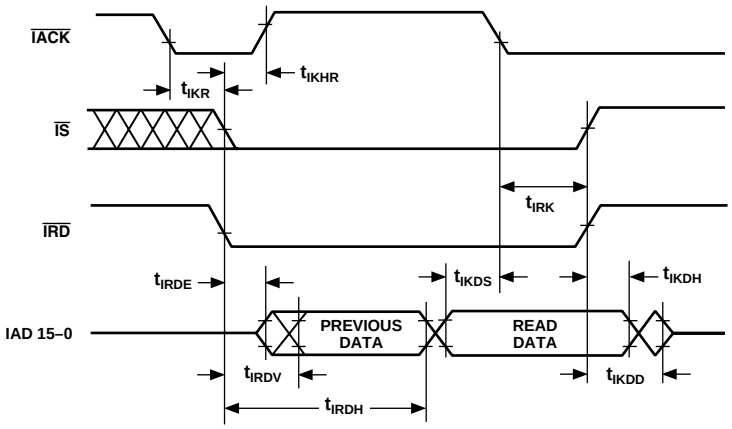



*Figure 18. IDMA Read, Long Read Cycle*

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| **IDMA Read, Short Read Cycle** | | | | |
| *Timing Requirements*: | | | | |
| $t_{IKR}$ | $\overline{IACK}$ Low before Start of Read[1] | 0 | | ns |
| $t_{IRP}$ | Duration of Read | 15 | | ns |
| *Switching Characteristics*: | | | | |
| $t_{IKHR}$ | $\overline{IACK}$ High after Start of Read[1] | | 15 | ns |
| $t_{IKDH}$ | IAD15–0 Data Hold after End of Read[2] | 0 | | ns |
| $t_{IKDD}$ | IAD15–0 Data Disabled after End of Read[2] | | 10 | ns |
| $t_{IRDE}$ | IAD15–0 Previous Data Enabled after Start of Read | 0 | | ns |
| $t_{IRDV}$ | IAD15–0 Previous Data Valid after Start of Read | | 15 | ns |

NOTES
[1]Start of Read = $\overline{IS}$ Low and $\overline{IRD}$ Low.
[2]End of Read = $\overline{IS}$ High or $\overline{IRD}$ High.

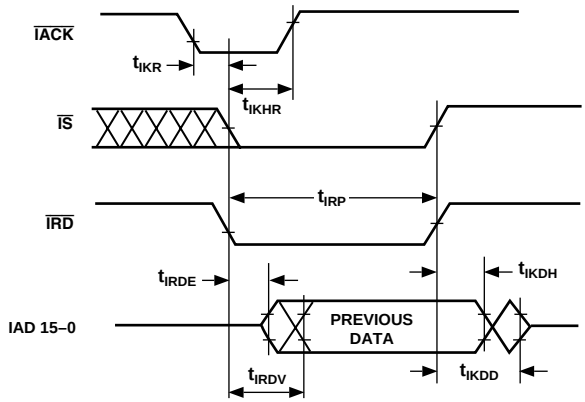



*Figure 19. IDMA Read, Short Read Cycle*

**OUTPUT DRIVE CURRENTS**

Figure 20 shows typical I-V characteristics for the output drivers of the ADSP-2184. The curves represent the current drive capability of the output drivers as a function of output voltage.

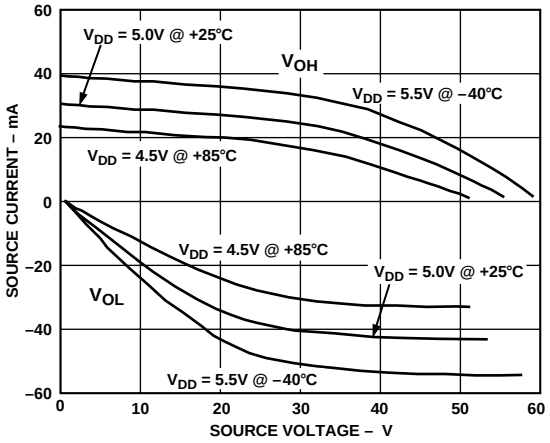



*Figure 20. Typical Drive Currents*

**POWER DISSIPATION**

To determine total power dissipation in a specific application, the following equation 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 the address pins switching.
- External data memory writes occur every other cycle with 50% of the data 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 $t_{CK}$ = 25 ns.

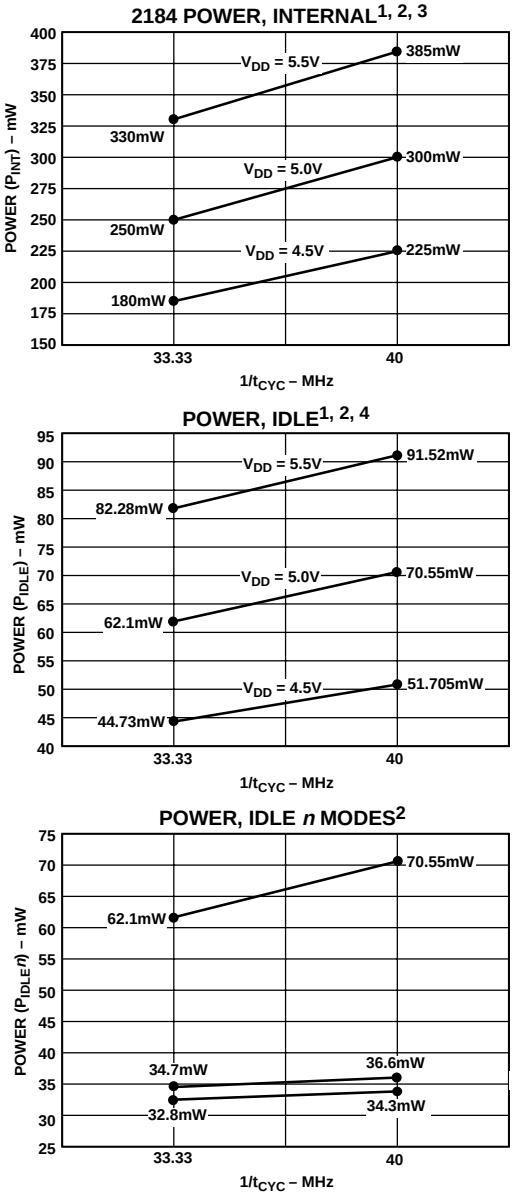
*Total Power Dissipation* = $P_{INT} + (C \times V_{DD}^2 \times f)$

$P_{INT}$ = internal power dissipation from Power vs. Frequency graph (Figure 21).

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

| | # of Pins | × C | × $V_{DD}^2$ | ×f | |
|---|---|---|---|---|---|
| Address, $\overline{DMS}$ | 8 | × 10 pF | × $5^2$ V | × 40 MHz = | 80 mW |
| Data Output, $\overline{WR}$ | 9 | × 10 pF | × $5^2$ V | × 20 MHz = | 45 mW |
| $\overline{RD}$ | 1 | × 10 pF | × $5^2$ V | × 20 MHz = | 5 mW |
| CLKOUT | 1 | × 10 pF | × $5^2$ V | × 40 MHz = | 10 mW |
| | | | | | 140 mW |

Total power dissipation for this example is PINT + 40 mW.




VALID FOR ALL TEMPERATURE GRADES.

[1]POWER REFLECTS DEVICE OPERATING WITH NO OUTPUT LOADS.

[2]$I_{DD}$ MEASUREMENT TAKEN WITH ALL INSTRUCTIONS EXECUTING FROM INTERNAL MEMORY. 50% OF THE INSTRUCTIONS ARE MULTIFUNCTION (TYPES 1, 4, 5, 12, 13, 14) 30% ARE TYPE 2 AND TYPE 6, AND 20% ARE IDLE INSTRUCTIONS.

[3]IDLE REFERS TO ADSP-2184 STATE OF OPERATION DURING EXECUTION OF IDLE INSTRUCTION. DEASSERTED PINS ARE DRIVEN TO EITHER $V_{DD}$ OR GND.

[4]TYPICAL POWER DISSIPATION AT 5.0V $V_{DD}$ AND $T_A$ = 25°C EXCEPT WHERE SPECIFIED.

*Figure 21. Power vs. Frequency*

## CAPACITIVE LOADING

Figures 22 and 23 show the capacitive loading characteristics of the ADSP-2184.

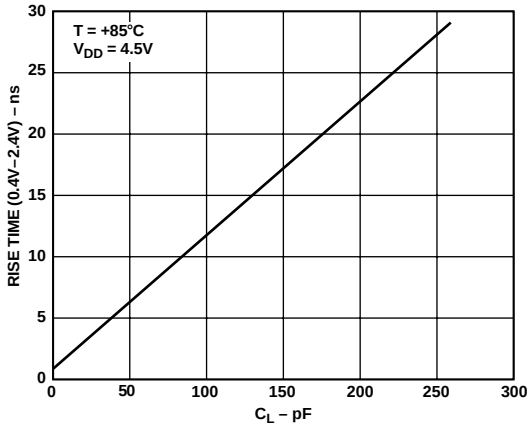



*Figure 22. Typical Output Rise Time vs. Load Capacitance, $C_L$ (at Maximum Ambient Operating Temperature)*

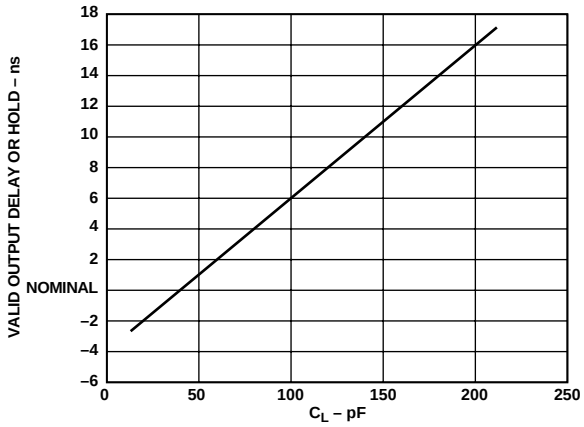



*Figure 23. Typical Output Valid Delay or Hold vs. Load Capacitance, $C_L$ (at Maximum Ambient Operating Temperature)*

## 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. The output disable time ($t_{DIS}$) is the difference of $t_{MEASURED}$ and $t_{DECAY}$, as shown in the Output Enable/Disable diagram. 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, $C_L$, and the current load, $i_L$, on the output pin. It can be approximated by the following equation:

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

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.

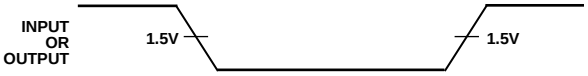



*Figure 24. Voltage Reference Levels for AC Measurements (Except Output Enable/Disable)*

### 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 the output has reached a specified high or low trip point, as shown in the Output Enable/Disable diagram. If multiple pins (such as the data bus) are enabled, the measurement value is that of the first pin to start driving.

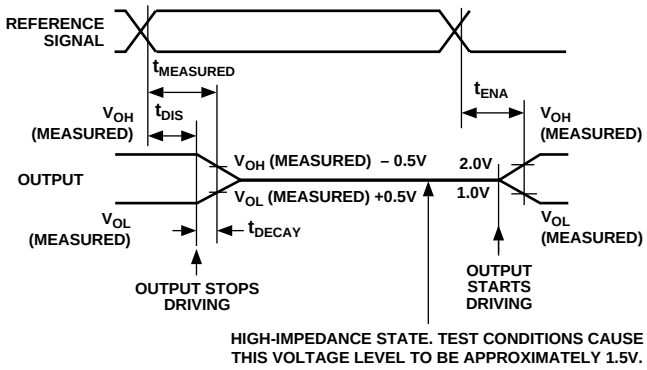



*Figure 25. Output Enable/Disable*

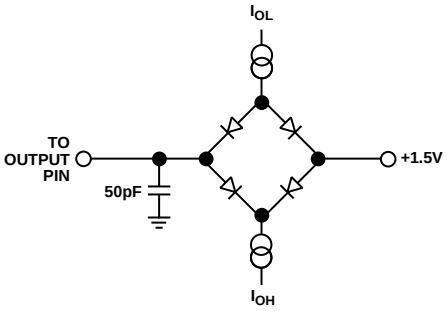



*Figure 26. Equivalent Device Loading for AC Measurements (Including All Fixtures)*

**ENVIRONMENTAL CONDITIONS**

Ambient Temperature Rating:

$T_{AMB}$ = $T_{CASE} - (PD \times \theta_{CA})$
$T_{CASE}$ = Case Temperature in °C
PD = Power Dissipation in W
$\theta_{CA}$ = Thermal Resistance (Case-to-Ambient)
$\theta_{JA}$ = Thermal Resistance (Junction-to-Ambient)
$\theta_{JC}$ = Thermal Resistance (Junction-to-Case)

| Package | $\theta_{JA}$ | $\theta_{JC}$ | $\theta_{CA}$ |
|---|---|---|---|
| LQFP | 50°C/W | 2°C/W | 48°C/W |

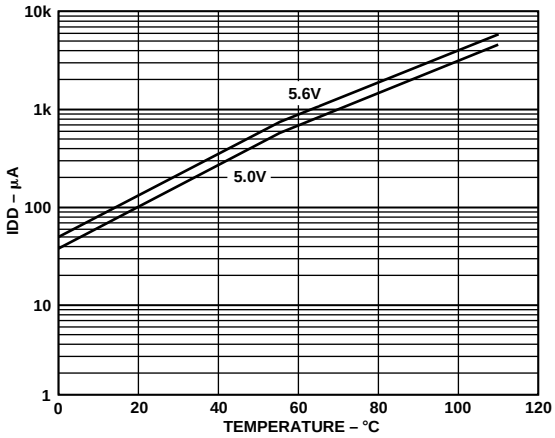
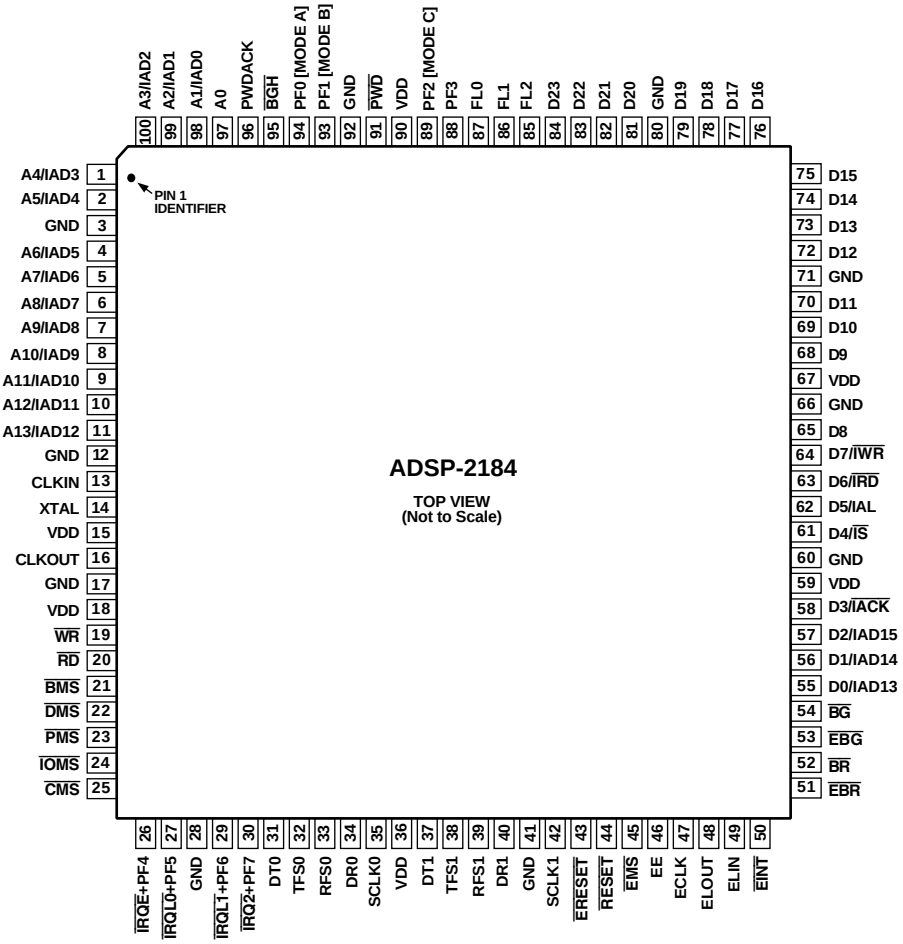



*Figure 27. Power-Down Supply Current*

**100-Lead LQFP Package Pinout**



**ORDERING GUIDE**

| Part Number | Ambient Temperature Range | Instruction Rate (MHz) | Package Description | Package Option* |
|---|---|---|---|---|
| ADSP-2184BST-160 | –40°C to +85°C | 40.0 | 100-Lead LQFP | ST-100 |

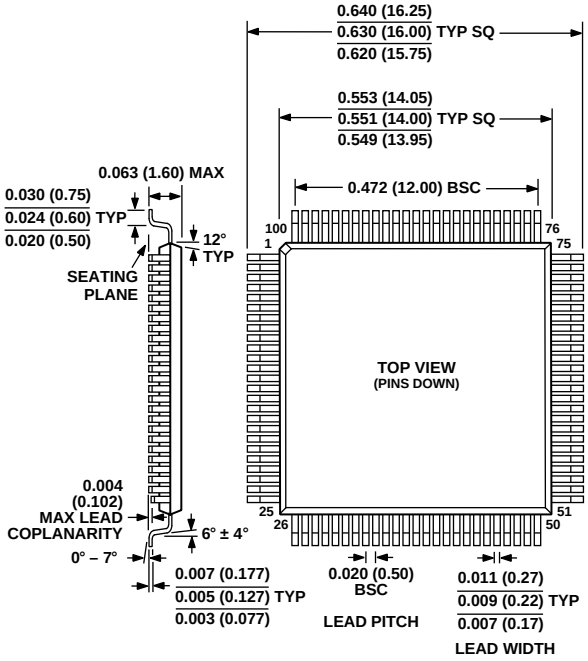
*ST = Plastic Thin Quad Flatpack (LQFP).

C3418–2–5/99

## OUTLINE DIMENSIONS

Dimensions shown in inches and (mm).

### 100-Lead Metric Thin Plastic Quad Flatpack (LQFP) (ST-100)




NOTE:
THE ACTUAL POSITION OF EACH LEAD IS WITHIN (0.08) 0.0032 FROM ITS IDEAL POSITION WHEN MEASURED IN THE LATERAL DIRECTION. CENTER FIGURES ARE TYPICAL UNLESS OTHERWISE NOTED

PRINTED IN U.S.A.