Welcome to E-XFL.COM

Understanding Embedded - DSP (Digital Signal Processors)

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

Applications of Embedded - DSP (Digital Signal Processors)

Details	
Product Status	Obsolete
Type	Fixed Point
Interface	Synchronous Serial Port (SSP)
Clock Rate	33.3MHz
Non-Volatile Memory	External
On-Chip RAM	80kB
Voltage - I/O	5.00V
Voltage - Core	5.00V
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	128-LQFP
Supplier Device Package	128-TQFP (14x20)
Purchase URL	https://www.e-xfl.com/product-detail/analog-devices/adsp-2181bstz-133

ADSP-2181* PRODUCT PAGE QUICK LINKS

Last Content Update: 02/23/2017

COMPARABLE PARTS

View a parametric search of comparable parts.

EVALUATION KITS

- EZ-ICE® Serial Emulator for ADSP-218x Processor Family

DOCUMENTATION

Application Notes

- AN-1: ADSP-21xx Legacy Application Notes
- AN-227: Digital Control System Design with the ADSP-2100 Family
- AN-227: Digital Control System Design with the ADSP-2100 Family
- AN-334: Digital Signal Processing Techniques
- AN-348: Avoiding Passive-Component Pitfalls
- AN-400: Considerations for Selecting a DSP Processor -- Why Buy the ADSP-2181?
- AN-415: ADSP-2181 IDMA Interface to Motorola MC68300 Family of Microprocessors
- AN-524: ADV601/ADV611 Bin Width Calculation in ADSP-21xx DSP
- AN-543: High Quality, All-Digital RF Frequency Modulation Generation with the ADSP-2181 and the AD9850 DDS
- EE-06: ADSP-21xx Serial Port Startup Issues
- EE-100: ADSP-218x External Overlay Memory
- EE-102: Mode D and ADSP-218x Pin Compatibility - the FAQs
- EE-103: Performing Level Conversion Between 5v and 3.3v IC's
- EE-104: Setting Up Streams with the VisualDSP Debugger
- EE-11: ADSP-2181 Priority Chain & IDMA Holdoffs
- EE-110: A Quick Primer on ELF and DWARF File Formats
- EE-115: ADSP-2189 IDMA Interface to Motorola MC68300 Family of Microprocessors
- EE-12: Interrupts and Programmable Flags on the ADSP-2185/2186
- EE-121: Porting Code from ADSP-21xx to ADSP-219x
- EE-122: Coding for Performance on the ADSP-219x
- EE-123: An Overview of the ADSP-219x Pipeline
- EE-124: Booting up the ADSP-2192
- EE-125: ADSP-218x Embedded System Software Management and In-System-Programming (ISP)
- EE-128: DSP in C++: Calling Assembly Class Member Functions From C++
- EE-129: ADSP-2192 Interprocessor Communication
- EE-130: Making Fast Transition from ADSP-21xx to ADSP-219x

- EE-131: Booting the ADSP-2191/95/96 DSPs
- EE-133: Converting From Legacy Architecture Files To Linker Description Files for the ADSP-218x
- EE-139: Interfacing the ADSP-2191 to an AD7476 via the SPI Port
- EE-142: Autobuffering, C and FFTs on the ADSP-218x
- EE-144: Creating a Master-Slave SPI Interface Between Two ADSP-2191 DSPs
- EE-145: SPI Booting of the ADSP-2191 using the Atmel AD25020N on an EZ-KIT Lite Evaluation Board
- EE-146: Implementing a Boot Manager for ADSP-218x Family DSPs
- EE-152: Using Software Overlays with the ADSP-219x and VisualDSP 2.0++
- EE-153: ADSP-2191 Programmable PLL
- EE-154: ADSP-2191 Host Port Interface
- EE-156: Support for the H.100 protocol on the ADSP-2191
- EE-158: ADSP-2181 EZ-Kit Lite IDMA to PC Printer Port Interface
- EE-159: Initializing DSP System & Control Registers From C and C++
- EE-164: Advanced EPROM Boot and No-boot Scenarios with ADSP-219x DSPs
- EE-168: Using Third Overtone Crystals with the ADSP-218x DSP
- EE-17: ADSP-2187L Memory Organization
- EE-18: Choosing and Using FFTs for ADSP-21xx
- EE-188: Using C To Implement Interrupt-Driven Systems On ADSP-219x DSPs
- EE-2: Using ADSP-218x I/O Space
- EE-21: AD1847/ADSP-2181 Daisy Chain Tips & Tricks
- EE-226: ADSP-2191 DSP Host Port Booting
- EE-227: CAN Configuration Procedure for ADSP-21992 DSPs
- EE-23: An AD1847/ADSP-2181 loopback example using a single index register for SPORT autobuffering
- EE-249: Implementing Software Overlays on ADSP-218x DSPs with VisualDSP++®
- EE-32: Language Extensions: Memory Storage Types, ASM & Inline Constructs
- EE-33: Programming The ADSP-21xx Timer In C
- EE-35: Troubleshooting your ADSP-218x EZ-ICE
- EE-356: Emulator and Evaluation Hardware Troubleshooting Guide for CCES Users
- EE-36: ADSP-21xx Interface to the IOM-2 bus
- EE-38: ADSP-2181 IDMA Port - Cycle Steal Timing
- EE-39: Interfacing 5V Flash Memory to an ADSP-218x (Byte Programming Algorithm)
- EE-4: ADSP-21xx Multi-channel Slot Assignments for the AD1847
- EE-48: Converting Legacy 21xx Systems To A 218x System Design
- EE-5: ADSP-218x Full Memory Mode vs. Host Memory Mode
- EE-60: Simulating an RS-232 UART Using the Synchronous Serial Ports on the ADSP-21xx Family DSPs
- EE-64: Setting Mode Pins on Reset
- EE-68: Analog Devices JTAG Emulation Technical Reference
- EE-71: Minimum Rise Time Specs for Critical Interrupt and Clock Signals on the ADSP-21x1/21x5
- EE-74: Analog Devices Serial Port Development and Troubleshooting Guide
- EE-78: BDMA Usage on 100 pin ADSP-218x DSPs Configured for IDMA Use
- EE-79: EPROM Booting In Host Mode with 100 Pin 218x Processors
- EE-82: Using an ADSP-2181 DSP's IO Space to IDMA Boot Another ADSP-2181
- EE-89: Implementing A Software UART on the ADSP-2181 EZ-Kit-Lite
- EE-90: Using the 21xx C-FFT Library
- EE-96: Interfacing Two AD73311 Codecs to the ADSP-218x

Data Sheet

- ADSP-2181: 16-bit, 40 MIPS, 5v, 2 serial ports, host port, 80 KB RAM Data Sheet

Evaluation Kit Manuals

- ADSP-2181 EZ-KIT Lite® Evaluation System Manual

Integrated Circuit Anomalies

- ADSP-2181 Anomaly List for Revisions 0.0-4.0

Processor Manuals

- ADSP 21xx Processors: Manuals
- ADSP-218x DSP Hardware Reference
- ADSP-218x DSP Instruction Set Reference
- Using the ADSP-2100 Family Volume 2

Software Manuals

- CrossCore Embedded Studio 2.5.0 C/C++ Library Manual for SHARC Processors
- VisualDSP++ 3.5 Assembler and Preprocessor Manual for ADSP-218x and ADSP-219x DSPs
- VisualDSP++ 3.5 C Compiler and Library Manual for ADSP-218x DSPs

- VisualDSP++ 3.5 C/C++ Compiler and Library Manual for ADSP-219x Processors
- VisualDSP++ 3.5 Linker and Utilities Manual for 16-Bit Processors

SOFTWARE AND SYSTEMS REQUIREMENTS

- Software and Tools Anomalies Search

TOOLS AND SIMULATIONS

- ADSP-21xx Processors: Software and Tools

DESIGN RESOURCES

- ADSP-2181 Material Declaration
- PCN-PDN Information
- Quality And Reliability
- Symbols and Footprints

DISCUSSIONS

View all ADSP-2181 EngineerZone Discussions.

SAMPLE AND BUY

Visit the product page to see pricing options.

TECHNICAL SUPPORT

Submit a technical question or find your regional support number.

DOCUMENT FEEDBACK

Submit feedback for this data sheet.

ADSP-2181 can fetch an operand from program memory and the next instruction in the same cycle.

In addition to the address and data bus for external memory connection, the ADSP-2181 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-2181 to continue running from on-chip memory. Normal execution mode requires the processor to halt while buses are granted.

The ADSP-2181 can respond to 13 possible interrupts, eleven of which are accessible at any given time. There can be 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-2181 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, there are eight flags that are programmable as inputs or outputs and three flags that are always outputs.

A programmable interval timer generates periodic interrupts. A 16-bit count register (TCOUNT) is decremented every *n* processor cycles, 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-2181 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-2181 SPORTs. Refer to the *ADSP-2100 Family User's Manual, Third Edition* for further details.

- 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.

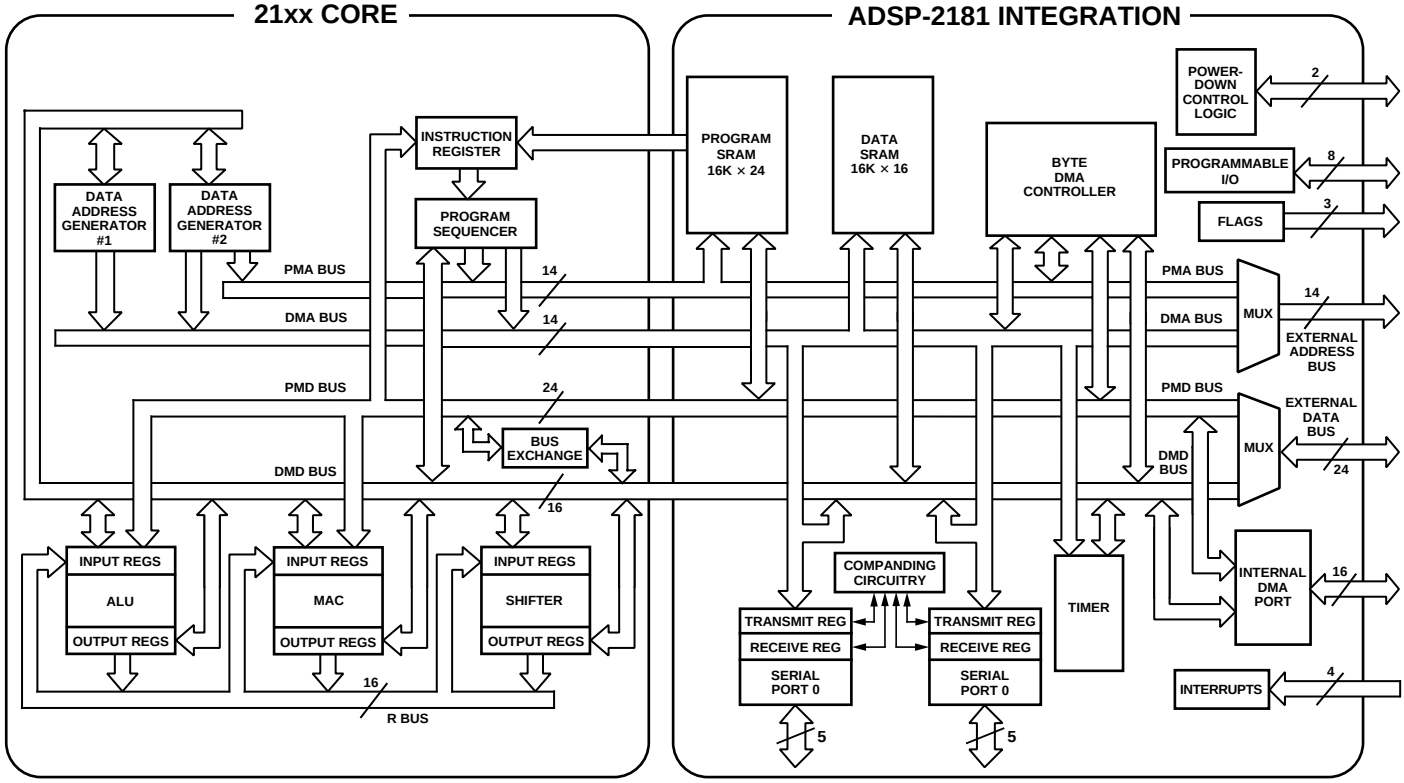


Figure 1. ADSP-2181 Block Diagram

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)

The IDMA Port provides an efficient means of communication between a host system and the ADSP-2181. 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-2181 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 address latch signal latches this value into the IDMAA register.

Once the address is stored, data can either be read from or written to the ADSP-2181'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-2181 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-2181 has two mechanisms to allow automatic loading of the on-chip program memory after reset. The method for booting after reset is controlled by the MMAP and BMODE pins as shown in Table VI.

Table VI. Boot Summary Table

MMAP	BMODE	Booting Method
0	0	BDMA feature is used in default mode 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.
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.
1	X	Bootstrap features disabled. Program execution immediately starts from location 0.

BDMA Booting

When the BMODE and MMAP pins specify BDMA booting (MMAP = 0, BMODE = 0), the ADSP-2181 initiates a BDMA boot sequence when reset is released. 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 ADSP-2100 Family Development Software (Revision 5.02 and later) fully supports the BDMA booting feature and can generate byte memory space compatible boot code.

The IDLE instruction can also be used to allow the processor to hold off execution while booting continues through the BDMA interface.

IDMA Booting

The ADSP-2181 can also boot programs through its Internal DMA port. If BMODE = 1 and MMAP = 0, the ADSP-2181 boots from the IDMA port. 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.

The ADSP-2100 Family Development Software (Revision 5.02 and later) can generate IDMA compatible boot code.

Bus Request and Bus Grant

The ADSP-2181 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 ($\overline{BR}$) signal. If the ADSP-2181 is not performing an external memory access, then it responds to the active $\overline{BR}$ input in the following processor cycle by:

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

If Go Mode is enabled, the ADSP-2181 will not halt program execution until it encounters an instruction that requires an external memory access.

If the ADSP-2181 is performing an external memory access when the external device asserts the $\overline{BR}$ signal, then it will not three-state the memory interfaces or assert the $\overline{BG}$ 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 $\overline{BR}$ signal is released, the processor releases the $\overline{BG}$ signal, reenables the output drivers and continues program execution from the point where it stopped.

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

The $\overline{BGH}$ pin is asserted when the ADSP-2181 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-2181 deasserts $\overline{BG}$ and $\overline{BGH}$ and executes the external memory access.

Flag I/O Pins

The ADSP-2181 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-2181'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-2181 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.

INSTRUCTION SET DESCRIPTION

The ADSP-2181 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-2181'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-2181 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.

The ICE-Port interface consists of the following ADSP-2181 pins:

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

These ADSP-2181 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-2181 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:

$\overline{BR}$	$\overline{BG}$
GND	$\overline{RESET}$

The EZ-ICE uses the EE (emulator enable) signal to take control of the ADSP-2181 in the target system. This causes the processor to use its $\overline{ERESET}$, $\overline{EBR}$ and $\overline{EBG}$ pins instead of the $\overline{RESET}$, $\overline{BR}$ and $\overline{BG}$ pins. The $\overline{BG}$ output is three-stated. These signals do not need to be jumper-isolated in your system.

The EZ-ICE connects to the target system via a ribbon cable and a 14-pin female plug. The ribbon cable is 10 inches in length with one end fixed to the EZ-ICE. 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 7. 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.

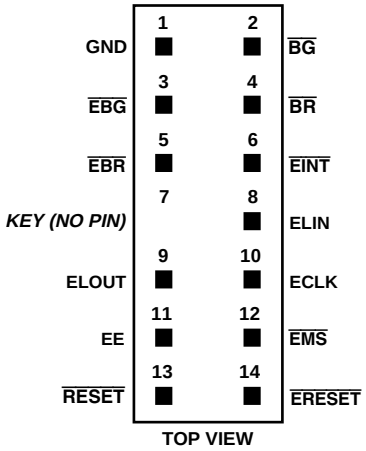


Figure 7. Target Board Connector for EZ-ICE

The 14-pin, 2-row pin strip header is keyed at the Pin 7 location—you must remove Pin 7 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 x 0.1 inches. The pin strip header must have at least 0.15 inch clearance on all sides to accept the EZ-ICE probe plug. Pin strip headers are available from vendors such as 3M, McKenzie and Samtec.

Target Memory Interface
For your target system to be compatible with the EZ-ICE emulator, it must comply with the memory interface guidelines listed below.

PM, DM, BM, IOM and CM
Design your Program Memory (PM), Data Memory (DM), Byte Memory (BM), I/O Memory (IOM) and Composite Memory (CM) external interfaces to comply with worst case device timing requirements and switching characteristics as specified in the DSP's data sheet. The performance of the EZ-ICE may approach published worst case specification for some memory access timing requirements and switching characteristics.

Note: If your target does not meet the worst case chip specification for memory access parameters, you may not be able to emulate your circuitry at the desired CLKIN frequency. Depending on the severity of the specification violation, you may have trouble manufacturing your system as DSP components statistically vary in switching characteristic and timing requirements within published limits.

Restriction: All memory strobe signals on the ADSP-2181 ($\overline{RD}$, $\overline{WR}$, $\overline{PMS}$, $\overline{DMS}$, $\overline{BMS}$, $\overline{CMS}$ and $\overline{IOMS}$) used in your target system must have 10 kΩ pull-up resistors connected when the EZ-ICE is being used. The pull-up resistors are necessary because there are no internal pull-ups to guarantee their state during prolonged three-state conditions resulting from typical EZ-ICE debugging sessions. These resistors may be removed at your option when the EZ-ICE is not being used.

Target System Interface Signals
When the EZ-ICE board is installed, the performance on some system signals changes. Design your system to be compatible with the following system interface signal changes introduced by the EZ-ICE board:

- EZ-ICE emulation introduces an 8 ns propagation delay between your target circuitry and the DSP on the $\overline{RESET}$ signal.
- EZ-ICE emulation introduces an 8 ns propagation delay between your target circuitry and the DSP on the $\overline{BR}$ signal.
- EZ-ICE emulation ignores $\overline{RESET}$ and $\overline{BR}$ when single-stepping.
- EZ-ICE emulation ignores $\overline{RESET}$ and $\overline{BR}$ when in Emulator Space (DSP halted).
- EZ-ICE emulation ignores the state of target $\overline{BR}$ in certain modes. As a result, the target system may take control of the DSP's external memory bus *only* if bus grant ($\overline{BG}$) is asserted by the EZ-ICE board's DSP.

Target Architecture File
The EZ-ICE software lets you load your program in its linked (executable) form. The EZ-ICE PC program can not load sections of your executable located in boot pages (by the linker). With the exception of boot page 0 (loaded into PM RAM), all sections of your executable mapped into boot pages are not loaded.

Write your target architecture file to indicate that only PM RAM is available for program storage, when using the EZ-ICE software's loading feature. Data can be loaded to PM RAM or DM RAM.

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) TQFP . . . . . +280°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; 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-2181 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-2181 features proprietary ESD protection circuitry to dissipate high energy discharges (Human Body Model). Per method 3015 of MIL-STD-883, the ADSP-2181 has been classified as 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—circuitry external to the processor must be designed for compatibility with these signal characteristics. Switching characteristics tell you what the processor will do in a given circumstance. You can also use switching characteristics to ensure that any timing requirement of a device connected to the processor (such as memory) is satisfied.

Timing Requirements apply to signals that are controlled by circuitry external to the processor, such as the data input for a read operation. Timing requirements guarantee that the processor operates correctly with other devices.

MEMORY TIMING SPECIFICATIONS

The table below shows common memory device specifications and the corresponding ADSP-2181 timing parameters, for your convenience.

Memory Device Specification	ADSP-2181 Timing Parameter	Timing Parameter Definition
Address Setup to Write Start	t_{ASW}	A0–A13, $\overline{xMS}$ Setup before $\overline{WR}$ Low
Address Setup to Write End	t_{AW}	A0–A13, $\overline{xMS}$ Setup before $\overline{WR}$ Deasserted
Address Hold Time	t_{WRA}	A0–A13, $\overline{xMS}$ Hold after $\overline{WR}$ Deasserted
Data Setup Time	t_{DW}	Data Setup before $\overline{WR}$ High
Data Hold Time	t_{DH}	Data Hold after $\overline{WR}$ High
$\overline{OE}$ to Data Valid	t_{RDD}	$\overline{RD}$ Low to Data Valid
Address Access Time	t_{AA}	A0–A13, $\overline{xMS}$ to Data Valid

$\overline{xMS}$ = $\overline{PMS}$, $\overline{DMS}$, $\overline{BMS}$, $\overline{CMS}$, $\overline{IOMS}$.

FREQUENCY DEPENDENCY FOR TIMING SPECIFICATIONS

t_{CK} is defined as $0.5t_{CKI}$. The ADSP-2181 uses an input clock with a frequency equal to half the instruction rate: a 16.67 MHz input clock (which is equivalent to 60 ns) yields a 30 ns processor cycle (equivalent to 33 MHz). t_{CK} values within the range of $0.5t_{CKI}$ period should be substituted for all relevant timing parameters to obtain the specification value.

Example: $t_{CKH} = 0.5t_{CK} – 7$ ns = 0.5 (25 ns) – 7 ns = 8 ns

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	$0.5t_{CK} - 7$		ns
t_{CKH}	CLKOUT Width High	$0.5t_{CK} - 7$		ns
t_{CKOH}	CLKIN High to CLKOUT High	0	20	ns
Control Signals				
Timing Requirement:				
t_{RSP}	$\overline{\text{RESET}}$ Width Low	$5t_{CK}$[1]		ns

NOTE
[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).

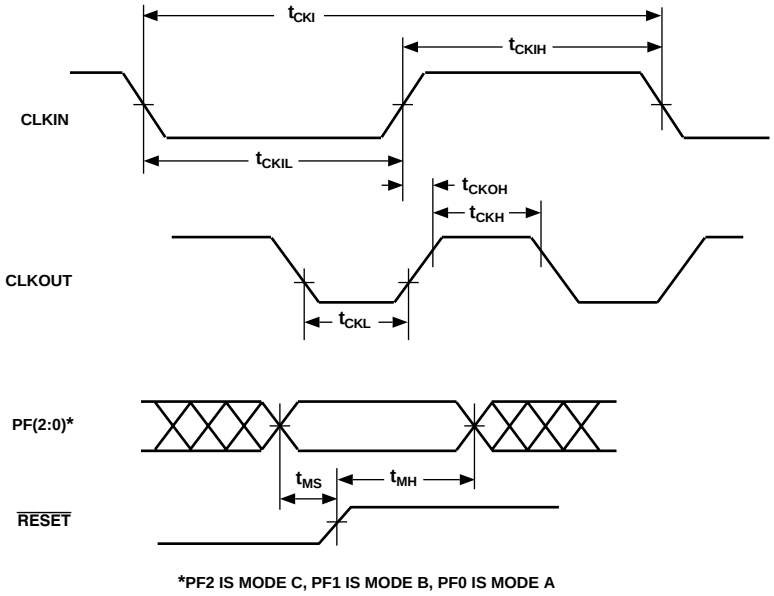


Figure 8. Clock Signals

Parameter		Min	Max	Unit
Interrupts and Flag				
Timing Requirements:				
t_{IFS}	$\overline{IRQx}$, FI, or PFx Setup before CLKOUT Low[1, 2, 3, 4]	$0.25t_{CK}$ + 15		ns
t_{IFH}	$\overline{IRQx}$, FI, or PFx Hold after CLKOUT High[1, 2, 3, 4]	$0.25t_{CK}$		ns
Switching Characteristics:				
t_{FOH}	Flag Output Hold after CLKOUT Low[5]	$0.5t_{CK}$ – 7		ns
t_{FOD}	Flag Output Delay from CLKOUT Low[5]		$0.5t_{CK}$ + 5	ns

NOTES
[1]If $\overline{IRQx}$ and FI inputs meet t_{IFS} and t_{IFH} setup/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]Edge-sensitive interrupts require pulsewidths greater than 10 ns; level-sensitive interrupts must be held low until serviced.
[3]$\overline{IRQx}$ = $\overline{IRQ0}$, $\overline{IRQ1}$, $\overline{IRQ2}$, $\overline{IRQL0}$, $\overline{IRQL1}$, $\overline{IRQE}$.
[4]PFx = PF0, PF1, PF2, PF3, PF4, PF5, PF6, PF7.
[5]Flag outputs = PFx, FL0, FL1, FL2, Flag_out4.

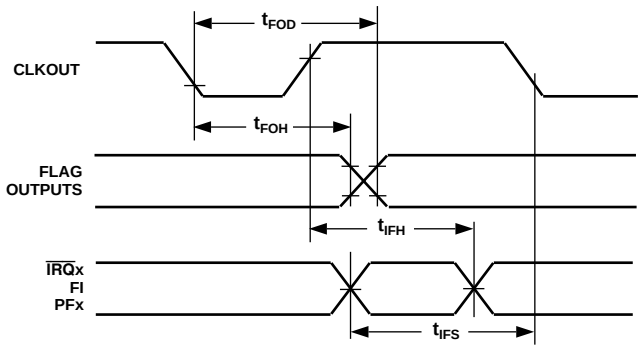


Figure 9. Interrupts and Flags

Parameter		Min	Max	Unit
Memory Read				
Timing Requirements:				
t_{RDD}	$\overline{RD}$ Low to Data Valid		$0.5t_{CK} - 9 + w$	ns
t_{AA}	A0–A13, $\overline{xMS}$ to Data Valid		$0.75t_{CK} - 10.5 + w$	ns
t_{RDH}	Data Hold from $\overline{RD}$ High	0		ns
Switching Characteristics:				
t_{RP}	$\overline{RD}$ Pulsewidth	$0.5t_{CK} - 5 + w$		ns
t_{CRD}	CLKOUT High to $\overline{RD}$ Low	$0.25t_{CK} - 5$	$0.25t_{CK} + 7$	ns
t_{ASR}	A0–A13, $\overline{xMS}$ Setup before $\overline{RD}$ Low	$0.25t_{CK} - 4$		ns
t_{RDA}	A0–A13, $\overline{xMS}$ Hold after $\overline{RD}$ Deasserted	$0.25t_{CK} - 3$		ns
t_{RWR}	$\overline{RD}$ High to $\overline{RD}$ or $\overline{WR}$ Low	$0.5t_{CK} - 5$		ns

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

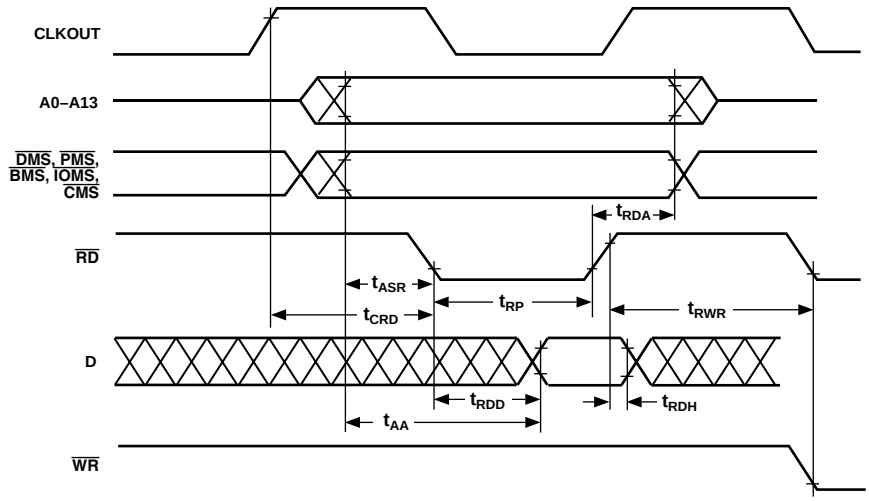


Figure 11. Memory Read

Parameter		Min	Max	Unit
Memory Write				
Switching Characteristics:				
t_{DW}	Data Setup before $\overline{WR}$ High	$0.5t_{CK} - 7 + w$		ns
t_{DH}	Data Hold after $\overline{WR}$ High	$0.25t_{CK} - 2$		ns
t_{WP}	$\overline{WR}$ Pulsewidth	$0.5t_{CK} - 5 + w$		ns
t_{WDE}	$\overline{WR}$ Low to Data Enabled	0		ns
t_{ASW}	A0–A13, $\overline{xMS}$ Setup before $\overline{WR}$ Low	$0.25t_{CK} - 4$		ns
t_{DDR}	Data Disable before $\overline{WR}$ or $\overline{RD}$ Low	$0.25t_{CK} - 4$		ns
t_{CWR}	CLKOUT High to $\overline{WR}$ Low	$0.25t_{CK} - 5$	$0.25\ t_{CK} + 7$	ns
t_{AW}	A0–A13, $\overline{xMS}$, Setup before $\overline{WR}$ Deasserted	$0.75t_{CK} - 9 + w$		ns
t_{WRA}	A0–A13, $\overline{xMS}$ Hold after $\overline{WR}$ Deasserted	$0.25t_{CK} - 3$		ns
t_{WWR}	$\overline{WR}$ High to $\overline{RD}$ or $\overline{WR}$ Low	$0.5t_{CK} - 5$		ns

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

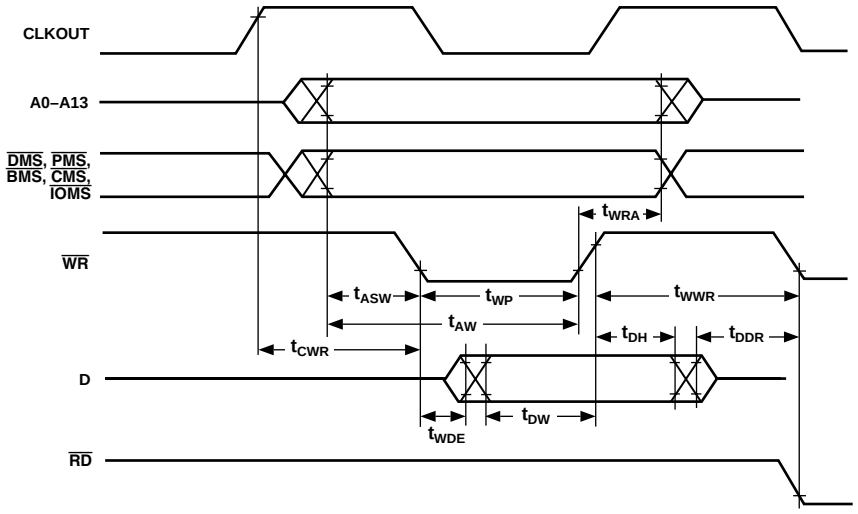


Figure 12. Memory Write

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]	2		ns
t_{IKA}	$\overline{IACK}$ Low before Start of Address Latch[1]	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.

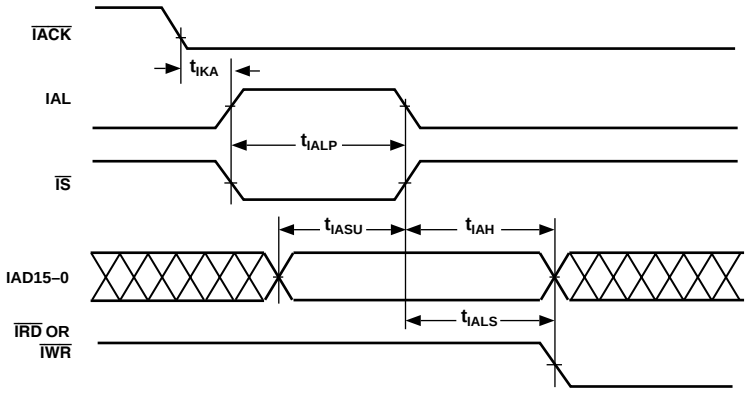


Figure 14. IDMA Address Latch

Parameter		Min	Max	Unit
IDMA Write, Short Write Cycle				
Timing Requirements:				
t_{IKW}	$\overline{IACK}$ Low before Start of Write[1]	0		ns
t_{IWP}	Duration of Write[1, 2]	15		ns
t_{IDSU}	IAD15–0 Data Setup before End of Write[2, 3, 4]	5		ns
t_{IDH}	IAD15–0 Data Hold after End of Write[2, 3, 4]	2		ns
Switching Characteristic:				
t_{IKHW}	Start of Write to $\overline{IACK}$ High		15	ns

NOTES
[1]Start of Write = $\overline{IS}$ Low and $\overline{IWR}$ Low.
[2]End of Write = $\overline{IS}$ High or $\overline{IWR}$ High.
[3]If Write Pulse ends before $\overline{IACK}$ Low, use specifications t_{IDSU}, t_{IDH}.
[4]If Write Pulse ends after $\overline{IACK}$ Low, use specifications t_{IKSU}, t_{IKH}.

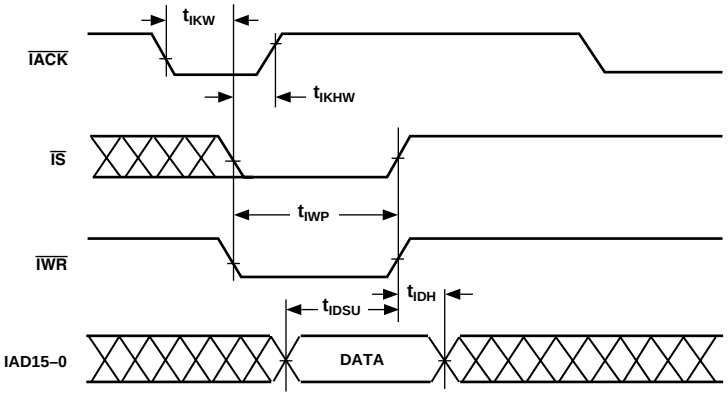


Figure 15. IDMA Write, Short Write Cycle

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]	$0.5t_{CK}$ + 10		ns
t_{IKH}	IAD15–0 Data Hold after $\overline{IACK}$ Low[2, 3]	2		ns
Switching Characteristics:				
t_{IKLW}	Start of Write to $\overline{IACK}$ Low[4]	$1.5t_{CK}$		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 User's Manual.

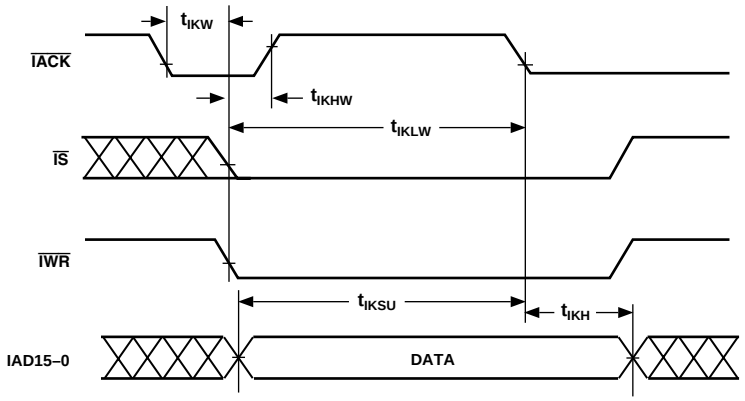


Figure 16. IDMA Write, Long Write Cycle

Parameter		Min	Max	Unit
IDMA Read, Long 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_{IKDS}	IAD15–0 Data Setup before IACK Low	$0.5t_{CK} - 10$		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]		12	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]	$2t_{CK} - 5$		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.

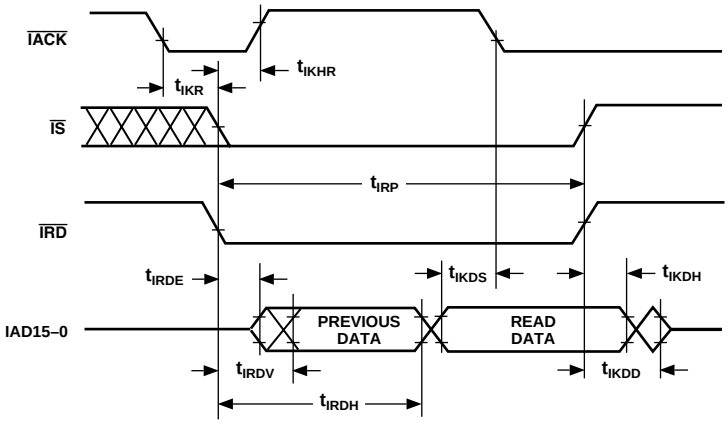


Figure 17. IDMA Read, Long Read Cycle

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]		12	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.

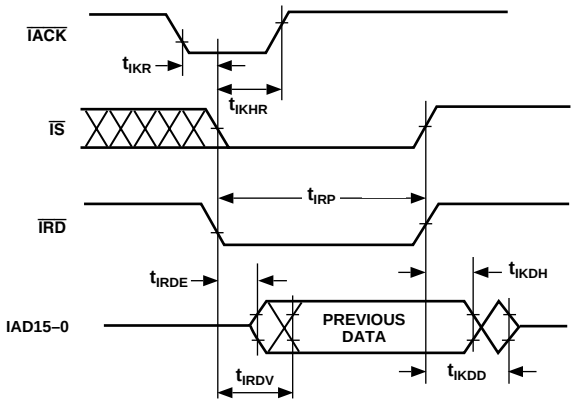


Figure 18. IDMA Read, Short Read Cycle

OUTPUT DRIVE CURRENTS

Figure 19 shows typical I-V characteristics for the output drivers of the ADSP-2181. The curves represent the current drive capability of the output drivers as a function of output voltage.

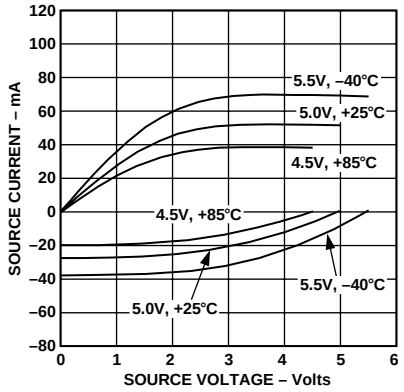


Figure 19. Typical Drive Currents

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:

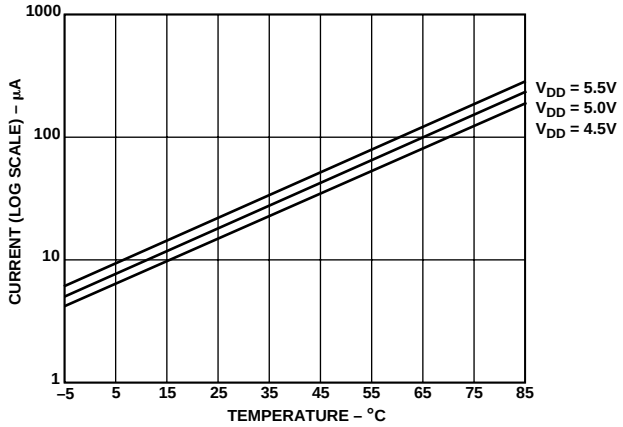
In an application where external data memory is used and no other outputs are active, power dissipation is calculated as follows:

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} = 30 ns.

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

P_{INT} = internal power dissipation from Power vs. Frequency graph (Figure 20).



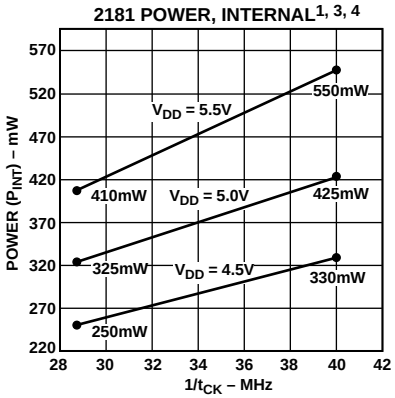
NOTES:
1. REFLECTS ADSP-2181 OPERATION IN LOWEST POWER MODE. (SEE "SYSTEM INTERFACE" CHAPTER OF THE *ADSP-2100 FAMILY USER'S MANUAL, THIRD EDITION,* FOR DETAILS.)
2. CURRENT REFLECTS DEVICE OPERATING WITH NO OUTPUT LOADS.

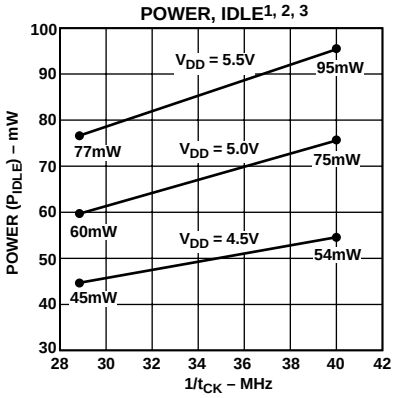
Figure 20. Power-Down Supply Current (Typical)

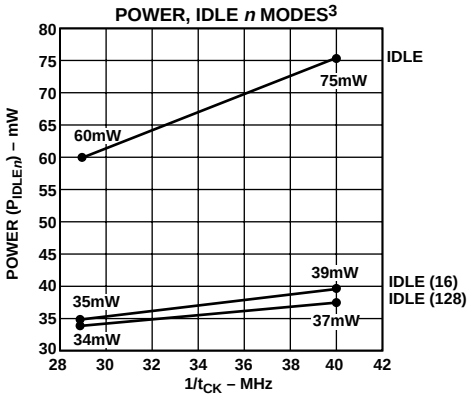
($C \times V_{DD}^2 \times f$) is calculated for each output:

	# of Pins	× C	× V_{DD}^2	× f		
Address, $\overline{DMS}$	8	× 10 pF	× 5^2 V	× 33.3 MHz	=	66.6 mW
Data Output, $\overline{WR}$	9	× 10 pF	× 5^2 V	× 16.67 MHz	=	37.5 mW
$\overline{RD}$	1	× 10 pF	× 5^2 V	× 16.67 MHz	=	4.2 mW
CLKOUT	1	× 10 pF	× 5^2 V	× 33.3 MHz	=	8.3 mW
						116.6 mW

Total power dissipation for this example is P_{INT} + 116.6 mW.







VALID FOR ALL TEMPERATURE GRADES.

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

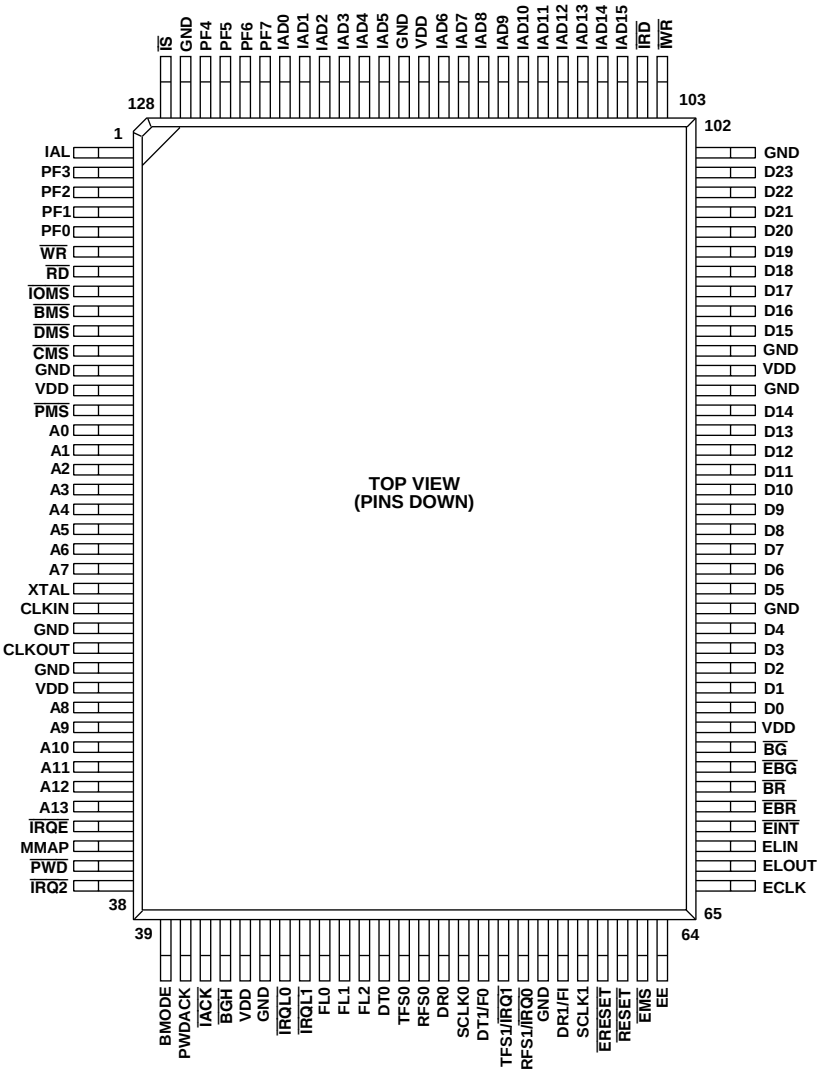
[2]IDLE REFERS TO ADSP-2181 STATE OF OPERATION DURING EXECUTION OF IDLE INSTRUCTION. DEASSERTED PINS ARE DRIVEN TO EITHER V_{DD} OR GND.

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

[4]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.

Figure 21. Power vs. Frequency

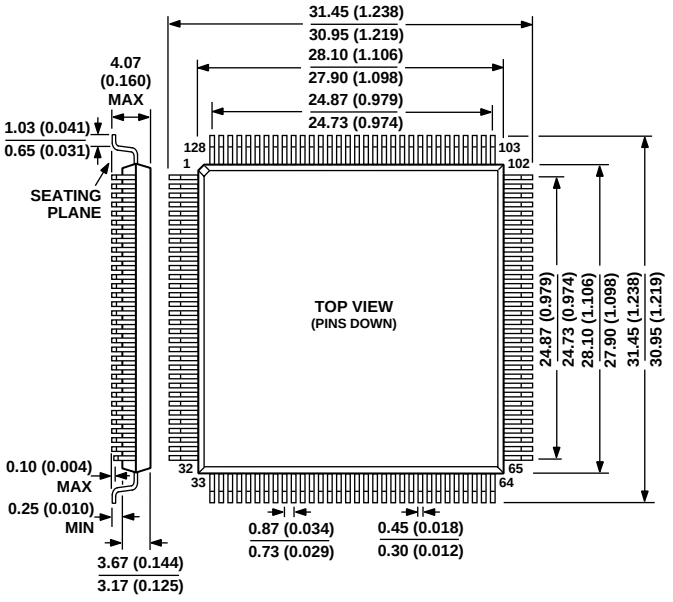
128-Lead TQFP Package Pinout



OUTLINE DIMENSIONS

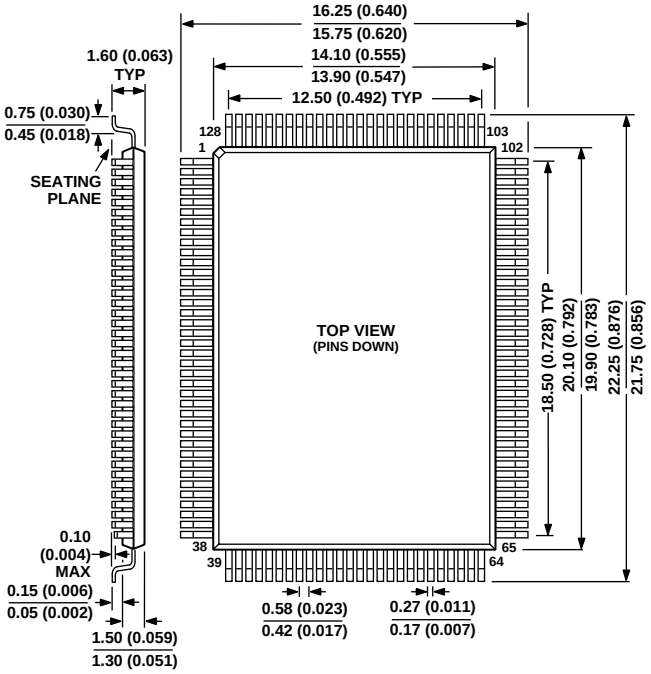
Dimensions shown in mm and (inches).

128-Lead Metric Plastic Quad Flatpack (PQFP) (S-128)



NOTE: THE ACTUAL POSITION OF EACH LEAD IS WITHIN .20 (.008) FROM ITS IDEAL POSITION WHEN MEASURED IN THE LATERAL DIRECTION. UNLESS OTHERWISE NOTED.

128-Lead Metric Thin Plastic Quad Flatpack (TQFP) (ST-128)



NOTE: THE ACTUAL POSITION OF EACH LEAD IS WITHIN .08 (.0032) FROM ITS IDEAL POSITION WHEN MEASURED IN THE LATERAL DIRECTION. UNLESS OTHERWISE NOTED.

ORDERING GUIDE

Part Number	Ambient Temperature Range	Instruction Rate (MHz)	Package Description	Package Options*
ADSP-2181KST-115	0°C to +70°C	28.8	128-Lead TQFP	ST-128
ADSP-2181BST-115	–40°C to +85°C	28.8	128-Lead TQFP	ST-128
ADSP-2181KS-115	0°C to +70°C	28.8	128-Lead PQFP	S-128
ADSP-2181BS-115	–40°C to +85°C	28.8	128-Lead PQFP	S-128
ADSP-2181KST-133	0°C to +70°C	33.3	128-Lead TQFP	ST-128
ADSP-2181BST-133	–40°C to +85°C	33.3	128-Lead TQFP	ST-128
ADSP-2181KS-133	0°C to +70°C	33.3	128-Lead PQFP	S-128
ADSP-2181BS-133	–40°C to +85°C	33.3	128-Lead PQFP	S-128
ADSP-2181KST-160	0°C to +70°C	40	128-Lead TQFP	ST-128
ADSP-2181KS-160	0°C to +70°C	40	128-Lead PQFP	S-128

*S = Plastic Quad Flatpack (PQFP), ST = Plastic Thin Quad Flatpack (TQFP).

C2041c–3–3/98

PRINTED IN U.S.A.