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	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	100-LQFP
Supplier Device Package	100-LQFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/analog-devices/adsp-2185bstz-133

- 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-226: ADSP-2191 DSP Host Port Booting
- EE-227: CAN Configuration Procedure for ADSP-21992 DSPs
- 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-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-2185: 16-bit, 33 MIPS, 5 v, 2 Serial Ports, Host Port, 80 KB RAM Data Sheet

Emulator Manuals

- ADSP-218X Family EZ-ICE Hardware Installation Guide

Integrated Circuit Anomalies

- ADSP-2185 Anomaly List for Revision 0.0

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
- VisualDSP++ 3.5 Loader 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-2185 Material Declaration
- PCN-PDN Information
- Quality And Reliability
- Symbols and Footprints

DISCUSSIONS

View all ADSP-2185 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.

Fabricated in a high speed, double metal, low power, 0.5 µm CMOS process, the ADSP-2185 operates with a 30 ns instruction cycle time. Every instruction can execute in a single processor cycle.

The ADSP-2185's flexible architecture and comprehensive instruction set allow the processor to perform multiple operations in parallel. In one processor cycle the ADSP-2185 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 and/or transmit data through the internal DMA port
- receive and/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-2185. 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-2185 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 Windows® 3.1 Control Software
- Stand-Alone Operation with Socketed EPROM
- EZ-ICE®* Connector for Emulator Control
- DSP Demo Programs

The ADSP-218x EZ-ICE®* Emulator aids in the hardware debugging of an ADSP-2185 system. The emulator consists of hardware, host computer resident software, and the target board connector. The ADSP-2185 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-ICE®*s. The ADSP-2185 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.

*All trademarks are the property of their respective holders.
*EZ-ICE and SoundPORT are registered trademarks of Analog Devices, Inc.

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-2185 functionality. For additional information on the architecture and instruction set of the processor, refer to the *ADSP-2100 Family User's Manual*. For more information about the development tools, refer to the *ADSP-2100 Family Development Tools Data Sheet*.

ARCHITECTURE OVERVIEW

The ADSP-2185 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-2185 assembly language uses an algebraic syntax for ease of coding and readability. A comprehensive set of development tools supports program development.

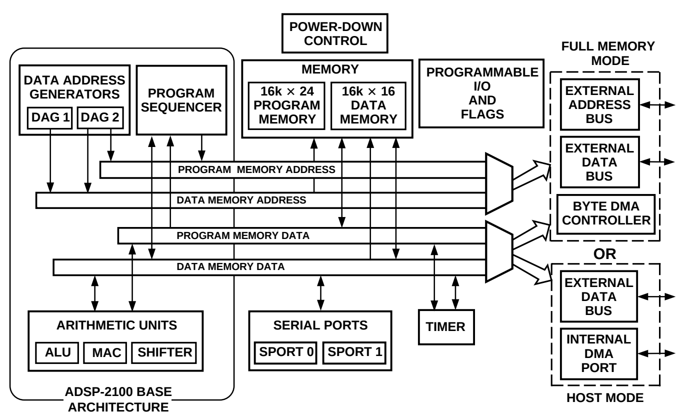


Figure 1. Block Diagram

Figure 1 is an overall block diagram of the ADSP-2185. 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-2185 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-2185 to fetch two operands in a single cycle, one from program memory and one from data memory. The ADSP-2185 can fetch an operand from program memory and the next instruction in the same cycle.

When configured in host mode, the ADSP-2185 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-2185 to continue running from on-chip memory. Normal execution mode requires the processor to halt while buses are granted.

The ADSP-2185 can respond to eleven interrupts. 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-2185 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-2185 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-2185 SPORTs. For additional information on Serial Ports, refer to the *ADSP-2100 Family User's Manual.*

- 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 pulse widths 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-2185 will be available in a 100-lead TQFP 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.

Common-Mode Pins

Pin Name(s)	# of Pins	Input/Output	Function
$\overline{\text{RESET}}$	1	I	Processor Reset Input
$\overline{\text{BR}}$	1	I	Bus Request Input
$\overline{\text{BG}}$	1	O	Bus Grant Output
$\overline{\text{BGH}}$	1	O	Bus Grant Hung Output
$\overline{\text{DMS}}$	1	O	Data Memory Select Output
$\overline{\text{PMS}}$	1	O	Program Memory Select Output
$\overline{\text{IOMS}}$	1	O	Memory Select Output
$\overline{\text{BMS}}$	1	O	Byte Memory Select Output
$\overline{\text{CMS}}$	1	O	Combined Memory Select Output
$\overline{\text{RD}}$	1	O	Memory Read Enable Output
$\overline{\text{WR}}$	1	O	Memory Write Enable Output
$\overline{\text{IRQ2}}$/	1	I	Edge- or Level-Sensitive Interrupt Request[1]
PF7		I/O	Programmable I/O Pin
$\overline{\text{IRQL0}}$/	1	I	Level-Sensitive Interrupt Requests[1]
PF5		I/O	Programmable I/O Pin
$\overline{\text{IRQL1}}$/	1	I	Level-Sensitive Interrupt Requests[1]
PF6		I/O	Programmable I/O Pin
$\overline{\text{IRQE}}$/	1	I	Edge-Sensitive Interrupt Requests[1]
PF4		I/O	Programmable I/O Pin
PF3	1	I/O	Programmable I/O Pin
Mode C/	1	I	Mode Select Input—Checked only During RESET
PF2		I/O	Programmable I/O Pin During Normal Operation
Mode B/	1	I	Mode Select Input—Checked only During RESET
PF1		I/O	Programmable I/O Pin During Normal Operation
Mode A/	1	I	Mode Select Input—Checked only During RESET
PF0		I/O	Programmable I/O Pin During Normal Operation
CLKIN, XTAL	2	I	Clock or Quartz Crystal Input
CLKOUT	1	O	Processor Clock Output
SPORT0	5	I/O	Serial Port I/O Pins
SPORT1/	5	I/O	Serial Port I/O Pins
$\overline{\text{IRQ1:0}}$			Edge- or Level-Sensitive Interrupts,
FI, FO			Flag In, Flag Out[2]
$\overline{\text{PWD}}$	1	I	Power-Down Control Input
PWDACK	1	O	Power-Down Control Output
FL0, FL1, FL2	3	O	Output Flags
VDD and GND	16	I	Power and Ground
EZ-Port	9	I/O	For Emulation Use

NOTES

[1]Interrupt/Flag pins retain both functions concurrently. If IMASK is set to enable the corresponding interrupts, the DSP will vector to the appropriate interrupt vector address when the pin is asserted, either by external devices or set as a programmable flag.

[2]SPORT configuration determined by the DSP System Control Register. Software configurable.

Memory Interface Pins

The ADSP-2185 processor can be used in one of two modes, Full Memory Mode, which allows BDMA operation with full external overlay memory and I/O capability, or Host Mode, which allows IDMA operation with limited external addressing capabilities. The operating mode is determined by the state of the Mode C pin during RESET and cannot be changed while the processor is running.

Full Memory Mode Pins (Mode C = 0)

Pin Name	# of Pins	Input/Output	Function
A13:0	14	O	Address Output Pins for Program, Data, Byte and I/O Spaces
D23:0	24	I/O	Data I/O Pins for Program, Data, Byte and I/O Spaces (8 MSBs Are Also Used as Byte Memory Addresses)

Host Mode Pins (Mode C = 1)

Pin Name	# of Pins	Input/Output	Function
IAD15:0	16	I/O	IDMA Port Address/Data Bus
A0	1	O	Address Pin for External I/O, Program, Data, or Byte Access
D23:8	16	I/O	Data I/O Pins for Program, Data Byte and I/O Spaces
$\overline{\text{IWR}}$	1	I	IDMA Write Enable
$\overline{\text{IRD}}$	1	I	IDMA Read Enable
IAL	1	I	IDMA Address Latch Pin
$\overline{\text{IS}}$	1	I	IDMA Select
$\overline{\text{IACK}}$	1	O	IDMA Port Acknowledge

In Host Mode, external peripheral addresses can be decoded using the A0, $\overline{\text{CMS}}$, $\overline{\text{PMS}}$, $\overline{\text{DMS}}$, and $\overline{\text{IOMS}}$ signals

Setting Memory Mode

Memory Mode selection for the ADSP-2185 is made during chip reset through the use of the Mode C pin. This pin is multiplexed with the DSP's PF2 pin, so care must be taken in how the mode selection is made. The two methods for selecting the value of Mode C are active and passive.

Passive configuration involves the use a pull-up or pull-down resistor connected to the Mode C pin. To minimize power consumption, or if the PF2 pin is to be used as an output in the DSP application, a weak pull-up or pull-down, on the order of 100 kΩ, can be used. This value should be sufficient to pull the pin to the desired level and still allow the pin to operate as a programmable flag output without undue strain on the processor's output driver. For minimum power consumption during power-down, reconfigure PF2 to be an input as the pull-up or pull-down will hold the pin in a known state and will not switch.

Active configuration involves the use of a three-stateable external driver connected to the Mode C pin. A driver's output enable should be connected to the DSP's $\overline{\text{RESET}}$ signal such that it only drives the PF2 pin when $\overline{\text{RESET}}$ is active (low). After

RESET is deasserted, the driver should three-state, thus allowing full use of the PF2 pin as either an input or output.

To minimize power consumption during power-down, configure the programmable flag as an output when connected to a three-stated buffer. This ensures that the pin will be held at a constant level and not oscillate should the three-state driver's level hover around the logic switching point.

Interrupts

The interrupt controller allows the processor to respond to the eleven possible interrupts and reset with minimum overhead. The ADSP-2185 provides four dedicated external interrupt input pins, $\overline{IRQ2}$, $\overline{IRQL0}$, $\overline{IRQL1}$ and $\overline{IRQE}$ (shared with the PF7:4 pins). In addition, SPORT1 may be reconfigured for $\overline{IRQ0}$, $\overline{IRQ1}$, FLAG_IN and FLAG_OUT, for a total of six external interrupts. The ADSP-2185 also supports internal interrupts from the timer, the byte DMA port, the two serial ports, software and the power-down control circuit. The interrupt levels are internally prioritized and individually maskable (except power-down and reset). The $\overline{IRQ2}$, $\overline{IRQ0}$ and $\overline{IRQ1}$ input pins can be programmed to be either level- or edge-sensitive. $\overline{IRQL0}$ and $\overline{IRQL1}$ are level-sensitive and IRQE is edge-sensitive. The priorities and vector addresses of all interrupts are shown in Table I.

Table I. Interrupt Priority & Interrupt Vector Addresses

Source Of Interrupt	Interrupt Vector Address (Hex)
Reset (or Power-Up with PUCR = 1)	0000 (Highest Priority)
Power-down (Nonmaskable)	002C
$\overline{IRQ2}$	0004
$\overline{IRQL1}$	0008
$\overline{IRQL0}$	000C
SPORT0 Transmit	0010
SPORT0 Receive	0014
$\overline{IRQE}$	0018
BDMA Interrupt	001C
SPORT1 Transmit or $\overline{IRQ1}$	0020
SPORT1 Receive or $\overline{IRQ0}$	0024
Timer	0028 (Lowest Priority)

Interrupt routines can either be nested, with higher priority interrupts taking precedence, or processed sequentially. Interrupts can be masked or unmasked with the IMASK register. Individual interrupt requests are logically ANDed with the bits in IMASK; the highest priority unmasked interrupt is then selected. The power-down interrupt is nonmaskable.

The ADSP-2185 masks all interrupts for one instruction cycle following the execution of an instruction that modifies the IMASK register. This does not affect serial port autobuffering or DMA transfers.

The interrupt control register, ICNTL, controls interrupt nesting and defines the $\overline{IRQ0}$, $\overline{IRQ1}$ and $\overline{IRQ2}$ external interrupts to be either edge- or level-sensitive. The $\overline{IRQE}$ pin is an external edge-sensitive interrupt and can be forced and cleared. The $\overline{IRQL0}$ and $\overline{IRQL1}$ pins are external level-sensitive interrupts.

The IFC register is a write-only register used to force and clear interrupts.

On-chip stacks preserve the processor status and are automatically maintained during interrupt handling. The stacks are twelve levels deep to allow interrupt, loop and subroutine nesting.

The following instructions allow global enable or disable servicing of the interrupts (including power-down), regardless of the state of IMASK. Disabling the interrupts does not affect serial port autobuffering or DMA.

ENA INTS;

DIS INTS;

When the processor is reset, interrupt servicing is enabled.

LOW POWER OPERATION

The ADSP-2185 has three low power modes that significantly reduce the power dissipation when the device operates under standby conditions. These modes are:

- Power-Down
- Idle
- Slow Idle

The CLKOUT pin may also be disabled to reduce external power dissipation.

Power-Down

The ADSP-2185 processor has a low power feature that lets the processor enter a very low power dormant state through hardware or software control. Here is a brief list of power-down features. Refer to the *ADSP-2100 Family User's Manual*, "System Interface" chapter, for detailed information about the power-down feature.

- Quick recovery from power-down. The processor begins executing instructions in as few as 100 CLKIN cycles.
- Support for an externally generated TTL or CMOS processor clock. The external clock can continue running during power-down without affecting the lowest power rating and 100 CLKIN cycle recovery.
- Support for crystal operation includes disabling the oscillator to save power (the processor automatically waits approximately 4096 CLKIN cycles for the crystal oscillator to start or stabilize), and letting the oscillator run to allow 100 CLKIN cycle start-up.
- Power-down is initiated by either the power-down pin ($\overline{PWD}$) or the software power-down force bit.
- Interrupt support allows an unlimited number of instructions to be executed before optionally powering down. The power-down interrupt also can be used as a nonmaskable, edge-sensitive interrupt.
- Context clear/save control allows the processor to continue where it left off or start with a clean context when leaving the power-down state.
- The $\overline{RESET}$ pin also can be used to terminate power-down.
- Power-down acknowledge pin indicates when the processor has entered power-down.

Idle

When the ADSP-2185 is in the Idle Mode, the processor waits indefinitely in a low power state until an interrupt occurs. When an unmasked interrupt occurs, it is serviced; execution then continues with the instruction following the IDLE instruction. In Idle mode IDMA, BDMA and autobuffer cycle steals still occur.

Slow Idle

The IDLE instruction is enhanced on the ADSP-2185 to let the processor's internal clock signal be slowed, further reducing power consumption. The reduced clock frequency, a programmable fraction of the normal clock rate, is specified by a selectable divisor given in the IDLE instruction. The format of the instruction is

IDLE (n);

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

When the *IDLE (n)* instruction is used, it effectively slows down the processor's internal clock and thus its response time to incoming interrupts. The one-cycle response time of the standard idle state is increased by n, the clock divisor. When an enabled interrupt is received, the ADSP-2185 will remain in the idle state for up to a maximum of n processor cycles (n = 16, 32, 64 or 128) before resuming normal operation.

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

SYSTEM INTERFACE

Figure 2 shows typical basic system configurations with the ADSP-2185, two serial devices, a byte-wide EPROM and optional external program and data overlay memories (mode selectable). Programmable wait state generation allows the processor to easily connect to slow peripheral devices. The ADSP-2185 also provides four external interrupts and two serial ports or six external interrupts and one serial port.

Host Memory mode allows access to the full external data bus, but limits addressing to a single address bit (A0). Additional system peripherals can be added in this mode through the use of external hardware to generate and latch address signals.

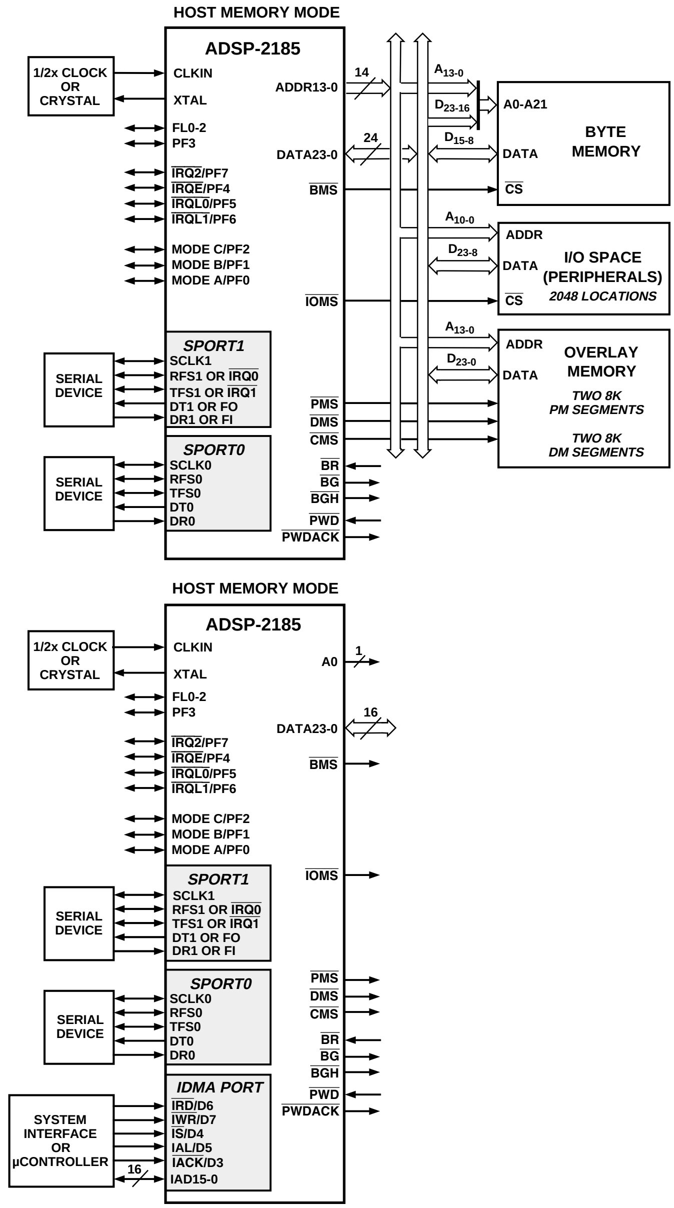


Figure 2. Basic System Configuration

Clock Signals

The ADSP-2185 can be clocked by either a crystal or a TTL-compatible clock signal.

The CLKIN input cannot be halted, changed during operation or operated below the specified frequency during normal operation. The only exception is while the processor is in the power-down state. For additional information, refer to Chapter 9, *ADSP-2100 Family User's Manual*, for detailed information on this power-down feature.

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

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 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. For BDMA accesses while in Host Mode, the addresses to boot memory must be constructed externally to the ADSP-2185. The only memory address bit provided by the processor is A0.

IDMA Port Booting

The ADSP-2185 can also boot programs through its Internal DMA port. If Mode C = 1, Mode B = 0 and Mode A = 1, the ADSP-2185 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 & Bus Grant

The ADSP-2185 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 (BR) signal. If the ADSP-2185 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 $\overline{PMS}$, $\overline{DMS}$, $\overline{BMS}$, $\overline{CMS}$, $\overline{IOMS}$, $\overline{RD}$, $\overline{WR}$ output drivers,
- Asserting the bus grant ($\overline{BG}$) signal and
- Halting program execution.

If Go Mode is enabled, the ADSP-2185 will not halt program execution until it encounters an instruction that requires an external memory access.

If the ADSP-2185 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-2185 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-2185 deasserts $\overline{BG}$ and $\overline{BGH}$ and executes the external memory access.

Flag I/O Pins

The ADSP-2185 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-2185'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-2185 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.

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.

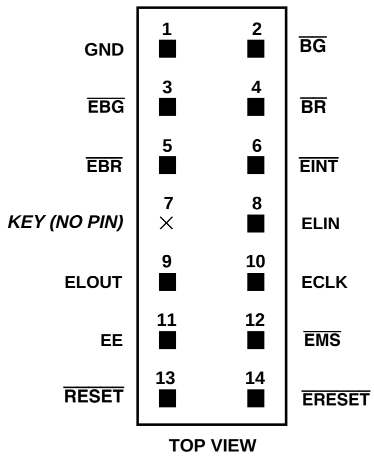


*Figure 7. Target Board Connector for EZ-ICE®**

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 × 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 this 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-2185 ($\overline{\text{RD}}$, $\overline{\text{WR}}$, $\overline{\text{PMS}}$, $\overline{\text{DMS}}$, $\overline{\text{BMS}}$, $\overline{\text{CMS}}$ and $\overline{\text{IOMS}}$) 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 change. 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 $\overline{\text{RESET}}$ signal.
- EZ-ICE®* emulation introduces an 8 ns propagation delay between your target circuitry and the DSP on the $\overline{\text{BR}}$ signal.
- EZ-ICE®* emulation ignores $\overline{\text{RESET}}$ and $\overline{\text{BR}}$ when single-stepping.
- EZ-ICE®* emulation ignores $\overline{\text{RESET}}$ and $\overline{\text{BR}}$ when in Emulator Space (DSP halted).
- EZ-ICE®* emulation ignores the state of target $\overline{\text{BR}}$ in certain modes. As a result, the target system may take control of the DSP's external memory bus only if bus grant ($\overline{\text{BG}}$) is asserted by the EZ-ICE®* board's DSP.

CAPACITIVE LOADING

Figures 9 and 10 show the capacitive loading characteristics of the ADSP-2185.

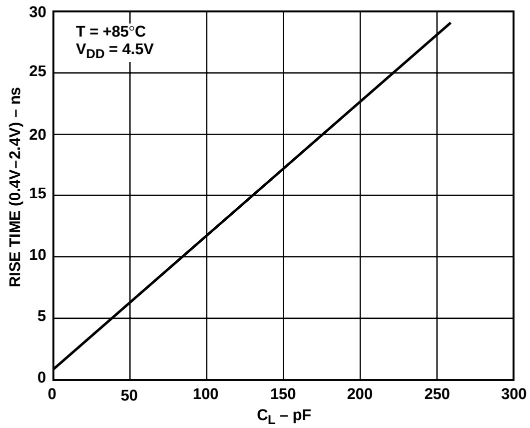


Figure 9. Typical Output Rise Time vs. Load Capacitance, C_L (at Maximum Ambient Operating Temperature)

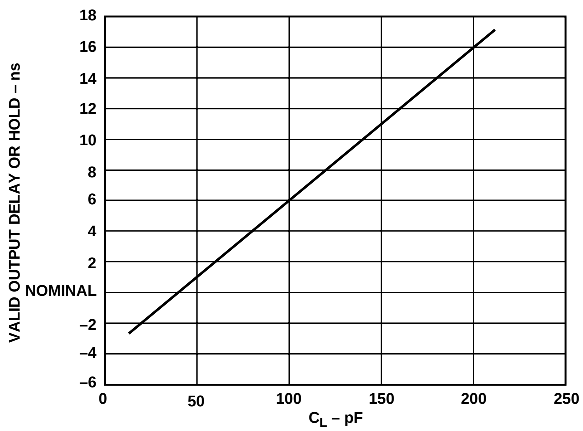


Figure 10. Typical Output Valid Delay or Hold vs. Load Capacitance, C_L (at Maximum Ambient Operating Temperature)

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:

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

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.

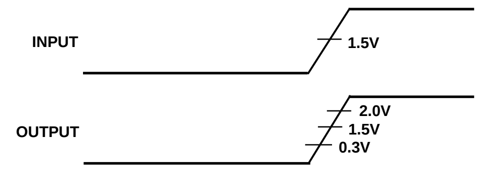


Figure 11. Voltage Reference Levels for AC Measurements (Except Output Enable/Disable)

Output Enable Time

Output pins are considered to be enabled when they 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.

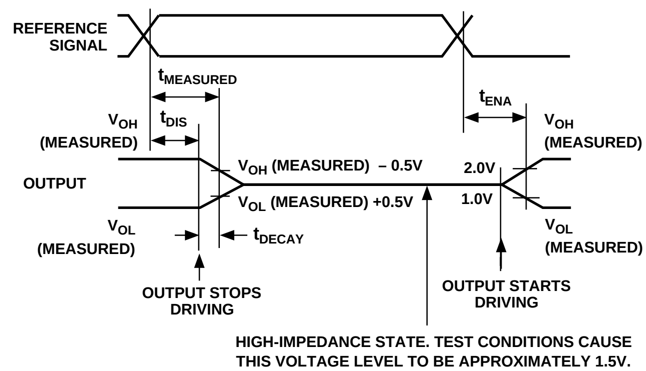


Figure 12. Output Enable/Disable

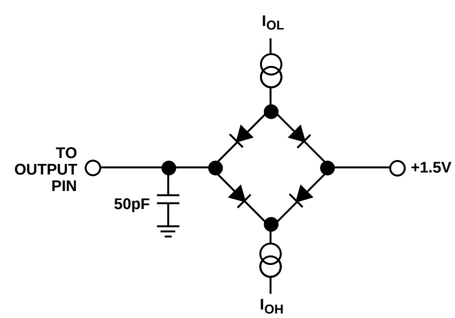


Figure 13. Equivalent Device Loading for AC Measurements (Including All Fixtures)

TIMING PARAMETERS

Parameter		Min	Max	Unit
Clock Signals and Reset				
Timing Requirements:				
t_{CKI}	CLKIN Period	60	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

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).

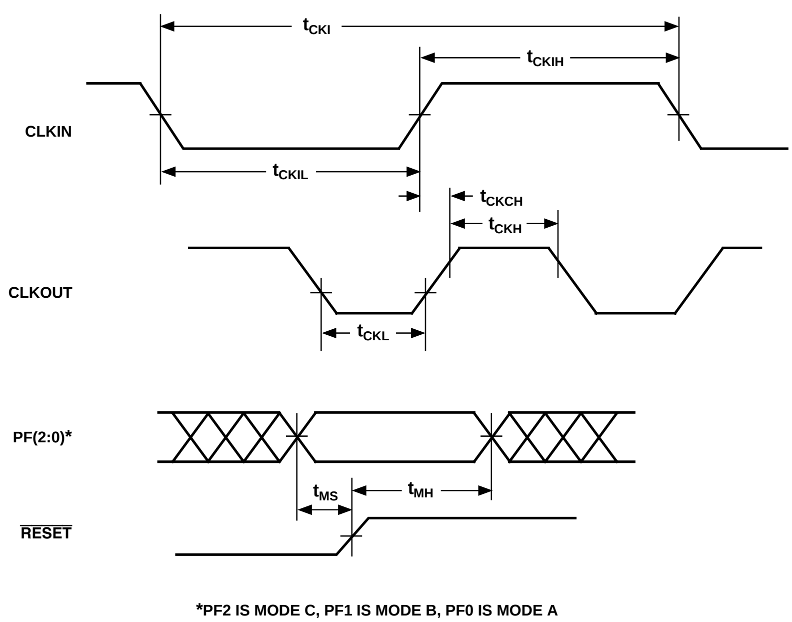


Figure 14. Clock Signals

Parameter		Min	Max	Unit
Interrupts and Flag				
Timing Requirements:				
t_{IFS}	$\overline{IRQx}$, FI, or PFx Setup before CLKOUT Low[1, 2, 3, 4]	$0.25\ t_{CK} + 15$		ns
t_{IFH}	$\overline{IRQx}$, FI, or PFx Hold after CLKOUT High[1, 2, 3, 4]	$0.25\ t_{CK}$		ns
Switching Characteristics:				
t_{FOH}	Flag Output Hold after CLKOUT Low[5]	$0.5\ t_{CK} - 7$		ns
t_{FOD}	Flag Output Delay from CLKOUT Low[5]		$0.25\ t_{CK} + 5$	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 *ADSP-2100 Family User's Manual* for further information on interrupt servicing.)

[2]Edge-sensitive interrupts require pulse widths 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_out[4].

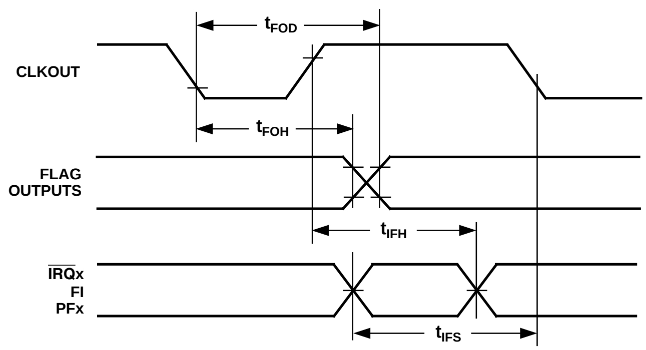


Figure 15. Interrupts and Flags

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} – 10.5 + w	ns
t_{RDH}	Data Hold from $\overline{RD}$ High	0		ns
Switching Characteristics:				
t_{RP}	$\overline{RD}$ Pulse Width	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}
$\overline{xMS}$ = $\overline{PMS}$, $\overline{DMS}$, $\overline{CMS}$, $\overline{IOMS}$, $\overline{BMS}$

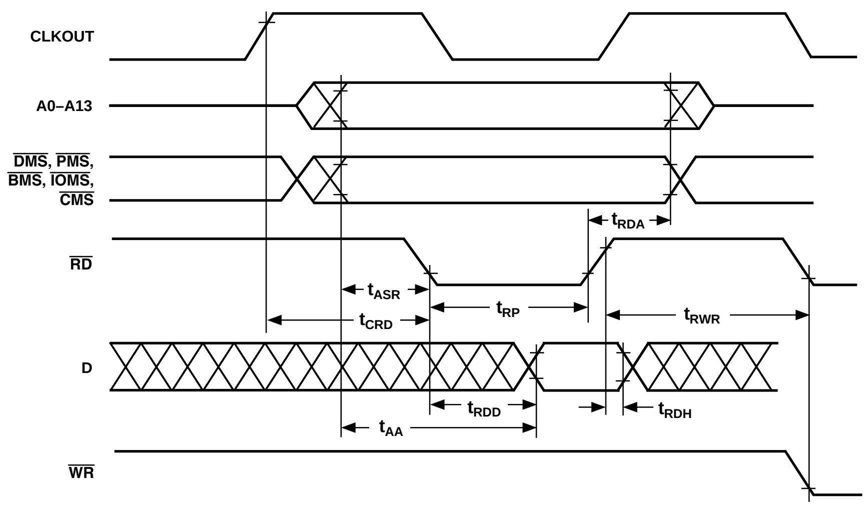


Figure 17. Memory Read

Parameter		Min	Max	Unit
Memory Write				
Switching Characteristics:				
t_{DW}	Data Setup before $\overline{WR}$ High	$0.5\ t_{CK} - 7 + w$		ns
t_{DH}	Data Hold after $\overline{WR}$ High	$0.25\ t_{CK} - 2$		ns
t_{WP}	$\overline{WR}$ Pulse Width	$0.5\ t_{CK} - 5 + w$		ns
t_{WDE}	$\overline{WR}$ Low to Data Enabled	0		ns
t_{ASW}	A0–A13, $\overline{xMS}$ Setup before $\overline{WR}$ Low	$0.25\ t_{CK} - 6$		ns
t_{DDR}	Data Disable before $\overline{WR}$ or $\overline{RD}$ Low	$0.25\ t_{CK} - 7$		ns
t_{CWR}	CLKOUT High to $\overline{WR}$ Low	$0.25\ t_{CK} - 5$	$0.25\ t_{CK} + 7$	ns
t_{AW}	A0–A13, $\overline{xMS}$, Setup before $\overline{WR}$ Deasserted	$0.75\ t_{CK} - 9 + w$		ns
t_{WRA}	A0–A13, $\overline{xMS}$ Hold after $\overline{WR}$ Deasserted	$0.25\ t_{CK} - 3$		ns
t_{WWR}	$\overline{WR}$ High to $\overline{RD}$ or $\overline{WR}$ Low	$0.5\ t_{CK} - 5$		ns

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

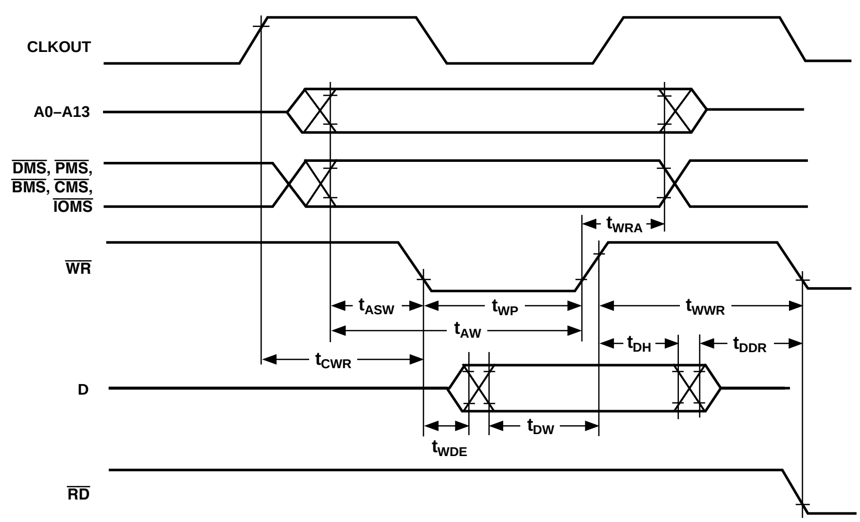


Figure 18. Memory Write

Parameter		Min	Max	Unit
IDMA Address Latch				
Timing Requirements:				
t_{IALP}	Duration of Address Latch[1, 3]	10		ns
t_{IASU}	IAD15–0 Address Setup before Address Latch End[3]	5		ns
t_{IAH}	IAD15–0 Address Hold after Address Latch End[3]	2		ns
t_{IKA}	$\overline{\text{IACK}}$ 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 = $\overline{\text{IS}}$ Low and IAL High.
[2]Start of Write or Read = $\overline{\text{IS}}$ Low and $\overline{\text{IWR}}$ Low or $\overline{\text{IRD}}$ Low.
[3]End of Address Latch = $\overline{\text{IS}}$ High or IAL Low.

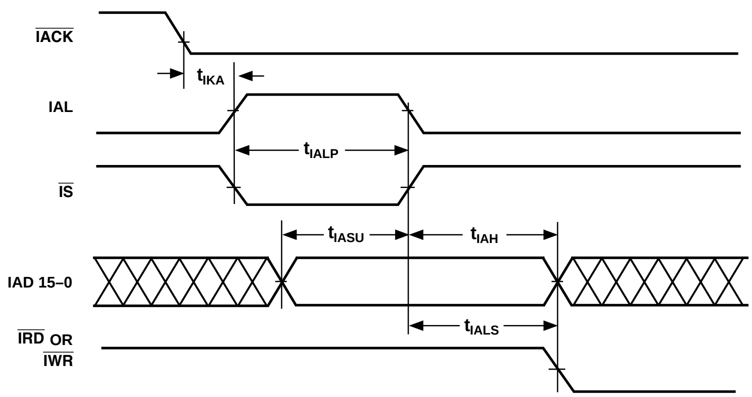


Figure 20. IDMA Address Latch

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*.

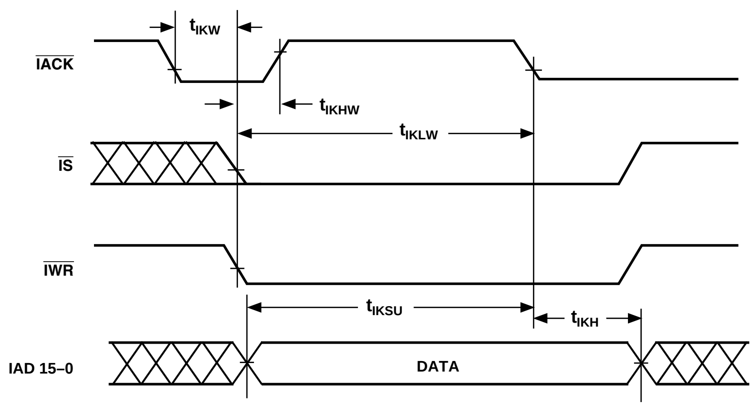


Figure 22. IDMA Write, Long Write Cycle

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[1]	15		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.

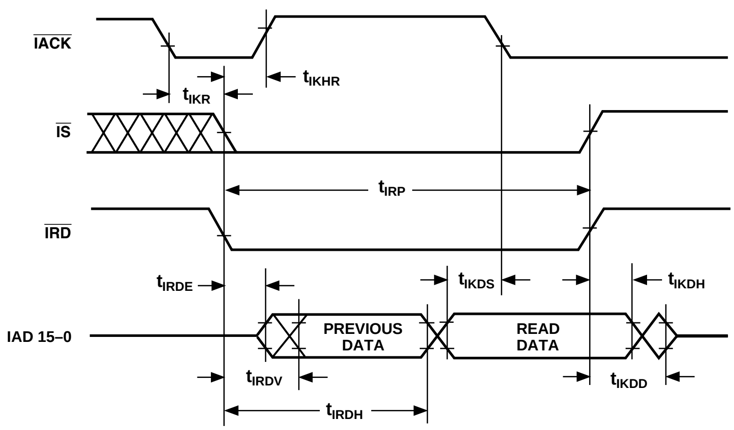
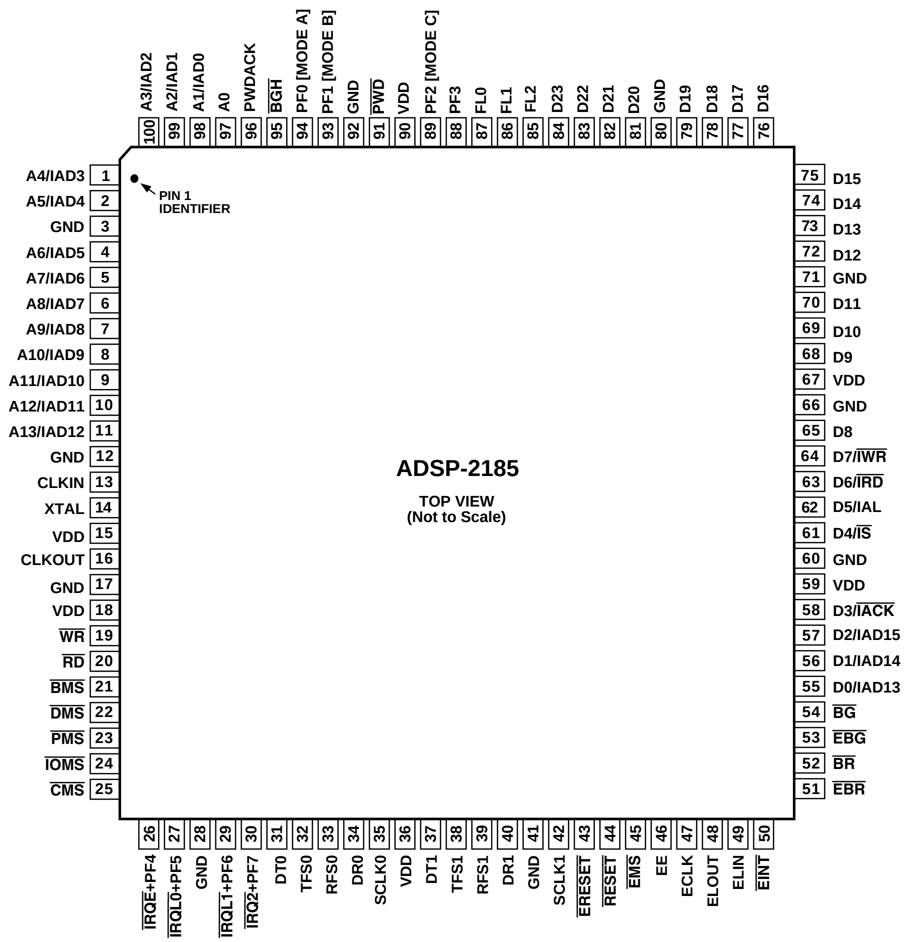


Figure 23. IDMA Read, Long Read Cycle

100-Lead TQFP Package Pinout



The ADSP-2185 package pinout is shown in the table below. Pin names in **bold** text replace the plain text named functions when Mode C = 1. A + sign separates two functions when either function can be active for either major I/O mode. Signals enclosed in brackets [ ] are state bits latched from the value of the pin at the deassertion of RESET.

TQFP Pin Configurations

TQFP Number	Pin Name	TQFP Number	Pin Name	TQFP Number	Pin Name	TQFP Number	Pin Name
1	A4/**IAD3**	26	IRQE + PF4	51	EBR	76	D16
2	A5/**IAD4**	27	IRQL0 + PF5	52	BR	77	D17
3	GND	28	GND	53	EBG	78	D18
4	A6/**IAD5**	29	IRQL1 + PF6	54	BG	79	D19
5	A7/**IAD6**	30	IRQ2 + PF7	55	D0/**IAD13**	80	GND
6	A8/**IAD7**	31	DT0	56	D1/**IAD14**	81	D20
7	A9/**IAD8**	32	TFS0	57	D2/**IAD15**	82	D21
8	A10/**IAD9**	33	RFS0	58	D3/**IACK**	83	D22
9	A11/**IAD10**	34	DR0	59	VDD	84	D23
10	A12/**IAD11**	35	SCLK0	60	GND	85	FL2
11	A13/**IAD12**	36	VDD	61	D4/**IS**	86	FL1
12	GND	37	DT1	62	D5/**IAL**	87	FL0
13	CLKIN	38	TFS1	63	D6/**IRD**	88	PF3
14	XTAL	39	RFS1	64	D7/**IWR**	89	PF2 [Mode C]
15	VDD	40	DR1	65	D8	90	VDD
16	CLKOUT	41	GND	66	GND	91	PWD
17	GND	42	SCLK1	67	VDD	92	GND
18	VDD	43	ERESET	68	D9	93	PF1 [Mode B]
19	WR	44	RESET	69	D10	94	PF0 [Mode A]
20	RD	45	EMS	70	D11	95	BGH
21	BMS	46	EE	71	GND	96	PWDACK
22	DMS	47	ECLK	72	D12	97	A0
23	PMS	48	ELOUT	73	D13	98	A1/**IAD0**
24	IOMS	49	ELIN	74	D14	99	A2/**IAD1**
25	CMS	50	EINT	75	D15	100	A3/**IAD2**

C2993–10–3/97
PRINTED IN U.S.A.