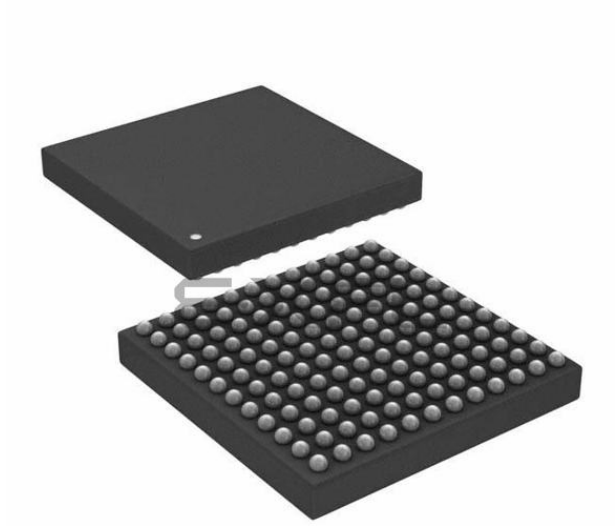

Welcome to E-XFL.COM

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	Host Interface, Serial Port
Clock Rate	75MHz
Non-Volatile Memory	External
On-Chip RAM	192kB
Voltage - I/O	3.30V
Voltage - Core	2.50V
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	144-LFBGA
Supplier Device Package	144-MiniBGA (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/analog-devices/adsp-2189mkca-300

ADSP-2189M* PRODUCT PAGE QUICK LINKS

Last Content Update: 02/23/2017

COMPARABLE PARTS

View a parametric search of comparable parts.

EVALUATION KITS

- EZ-KIT Lite Evaluation Kit for ADSP-218x Processor

DOCUMENTATION

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-524: ADV601/ADV611 Bin Width Calculation in ADSP-21xx DSP
- 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-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-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-2189M: DSP Microcomputer Data Sheet

Emulator Manuals

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

Evaluation Kit Manuals

- ADSP-2189M EZ-KIT Lite® Evaluation System Manual
- ADSP-218x DSP family and ADSP-2192 EZ-KIT Lite® Installation Procedure -Non-USB

Integrated Circuit Anomalies

- ADSP-2189M Anomaly List for Revision 0.0-0.4

Processor Manuals

- ADSP 21xx Processors: Manuals
- ADSP-218x DSP Hardware Reference
- ADSP-218x DSP Instruction Set Reference
- Using the ADSP-2100 Family Volume 1
- Using the ADSP-2100 Family Volume 2

Software Manuals

- 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 Component Software Engineering User's Guide for 16-Bit Processors
- VisualDSP++ 3.5 Getting Started Guide for 16-Bit Processors
- VisualDSP++ 3.5 Kernel VDK User's Guide for 16-Bit Processors
- VisualDSP++ 3.5 Linker and Utilities Manual for 16-Bit Processors
- VisualDSP++ 3.5 Loader Manual for 16-Bit Processors
- VisualDSP++ 3.5 User's Guide for 16-Bit Processors

SOFTWARE AND SYSTEMS REQUIREMENTS

- Software and Tools Anomalies Search

TOOLS AND SIMULATIONS

- ADSP-21xx Processors: Software and Tools
- ADSP-218xM IBIS Datafile (LQFP Package)

REFERENCE MATERIALS

Product Selection Guide

- ADI Complementary Parts Guide - Supervisory Devices and DSP Processors

DESIGN RESOURCES

- ADSP-2189M Material Declaration
- PCN-PDN Information
- Quality And Reliability
- Symbols and Footprints

DISCUSSIONS

View all ADSP-2189M 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, low power, CMOS process, the ADSP-2189M operates with a 13.3 ns instruction cycle time. Every instruction can execute in a single processor cycle.

The ADSP-2189M's flexible architecture and comprehensive instruction set allow the processor to perform multiple operations in parallel. In one processor cycle, the ADSP-2189M 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-2189M. 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-2189M 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-218x 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-218x
- Full 16-bit Stereo Audio I/O with AD1847 SoundPort® Codec
- RS-232 Interface to PC with Windows 3.1 Control Software
- EZ-ICE Connector for Emulator Control
- DSP Demo Programs

The ADSP-218x EZ-ICE® Emulator aids in the hardware debugging of an ADSP-2189M system. The emulator consists of hardware, host computer resident software and the target board connector. The ADSP-2189M integrates on-chip emulation support with a 14-pin ICE-Port interface. This interface provides a simpler target board connection that requires fewer mechanical clearance considerations than other ADSP-2100 Family EZ-ICEs. The ADSP-2189M device need not be removed from the target system when using the EZ-ICE, nor are any adapters needed. Due to the small footprint of the EZ-ICE connector, emulation can be supported in final board designs.

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

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

See "Designing An EZ-ICE-Compatible Target System" in the *ADSP-2100 Family EZ-Tools Manual* (ADSP-2181 sections) as well as the Designing an EZ-ICE compatible System 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-2189M functionality. For additional information on the architecture and instruction set of the processor, refer to the *ADSP-2100 Family User's Manual*, Third Edition. For more information about the development tools, refer to the ADSP-2100 Family Development Tools Data Sheet.

ARCHITECTURE OVERVIEW

The ADSP-2189M 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-2189M assembly language uses an algebraic syntax for ease of coding and readability. A comprehensive set of development tools supports program development.

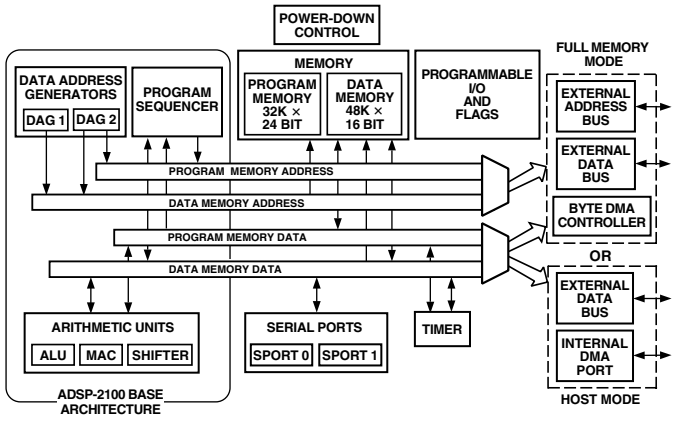

Figure 1. Functional Block Diagram

Figure 1 is an overall block diagram of the ADSP-2189M. 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.

EZ-ICE and SoundPort are registered trademarks of Analog Devices, Inc.

The internal result (R) bus connects the computational units so that 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-2189M 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-2189M to fetch two operands in a single cycle, one from program memory and one from data memory. The ADSP-2189M can fetch an operand from program memory and the next instruction in the same cycle.

In lieu of the address and data bus for external memory connection, the ADSP-2189M may be configured for 16-bit Internal DMA port (IDMA port) 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-2189M to continue running from on-chip memory. Normal execution mode requires the processor to halt while buses are granted.

The ADSP-2189M 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-2189M 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 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-2189M 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-2189M SPORTs. For additional information on Serial Ports, refer to the *ADSP-2100 Family User's Manual*, Third Edition.

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

PIN DESCRIPTIONS

The ADSP-2189M will be available in a 100-lead LQFP package. In order to maintain maximum functionality and reduce package size and pin count, some serial port, programmable flag, interrupt and external bus pins have dual, multiplexed functionality. The external bus pins are configured during $\overline{RESET}$ 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

Table II. ADSP-2189M Modes of Operation

MODE D	MODE C	MODE B	MODE A	Booting Method
X	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.[1]
X	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.
0	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. $\overline{\text{IACK}}$ has active pull-down. (**REQUIRES ADDITIONAL HARDWARE**).
0	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. $\overline{\text{IACK}}$ has active pull-down.[1]
1	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; $\overline{\text{IACK}}$ requires external pull-down. (**REQUIRES ADDITIONAL HARDWARE**).
1	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. $\overline{\text{IACK}}$ requires external pull-down.[1]

NOTE
[1]Considered as standard operating settings. Using these configurations allows for easier design and better memory management.

The master reset sets all internal stack pointers to the empty stack condition, masks all interrupts and clears the MSTAT register. When $\overline{\text{RESET}}$ is released, if there is no pending bus request and the chip is configured for booting, the boot-loading sequence is performed. The first instruction is fetched from on-chip program memory location 0x0000 once boot loading completes.

Power Supplies

The ADSP-2189M has separate power supply connections for the internal (V_{DDINT}) and external (V_{DDEXT}) power supplies. The internal supply must meet the 2.5 V requirement. The external supply can be connected to either a 2.5 V or 3.3 V supply. All external supply pins must be connected to the same supply. All input and I/O pins can tolerate input voltages up to 3.6 V regardless of the external supply voltage. This feature provides maximum flexibility in mixing 2.5 V and 3.3 V components.

MODES OF OPERATION

Setting Memory Mode

Memory Mode selection for the ADSP-2189M 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 10 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-statable 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). When $\overline{\text{RESET}}$ 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 will not oscillate should the three-state driver's level hover around the logic switching point.

$\overline{\text{IACK}}$ Configuration

Mode D = 0 and in host mode: $\overline{\text{IACK}}$ is an active, driven signal and cannot be wire OR-ed.

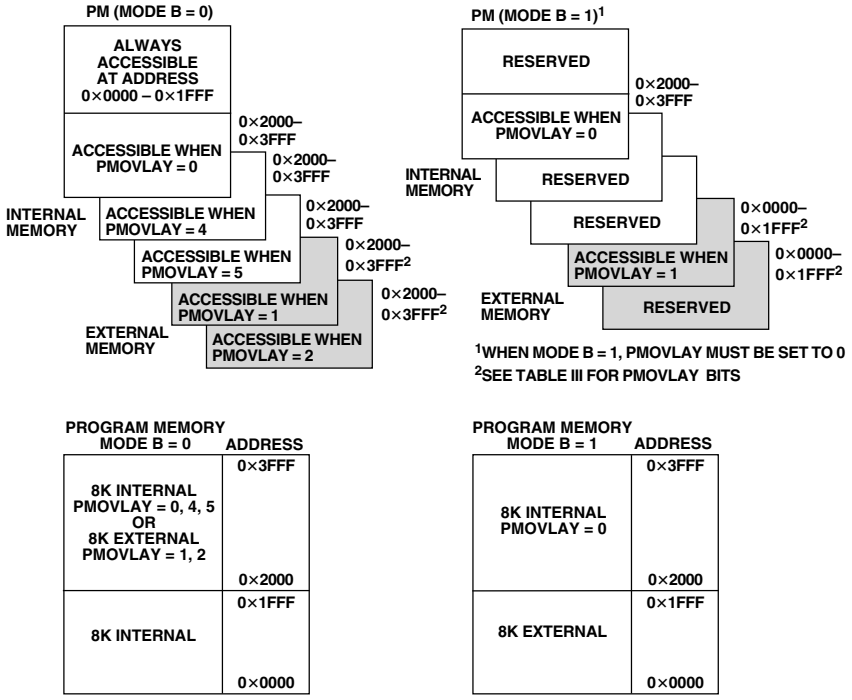

Figure 4. Program Memory

Mode D = 1 and in host mode: $\overline{\text{IACK}}$ is an open source and requires an external pull-down, but multiple $\overline{\text{IACK}}$ pins can be wire OR-ed together.

MEMORY ARCHITECTURE

The ADSP-2189M provides a variety of memory and peripheral interface options. The key functional groups are Program Memory, Data Memory, Byte Memory and I/O. Refer to the following figures and tables for PM and DM memory allocations in the ADSP-2189M.

Program Memory

Program Memory, *Full Memory Mode* is a 24-bit-wide space for storing both instruction op codes and data. The ADSP-2189M has 32K words of Program Memory RAM on chip and the capability of accessing up to two 8K external memory overlay spaces using the external data bus.

Program Memory, *Host Mode* allows access to all internal memory. External overlay access is limited by a single external address line (A0). External program execution is not available in host mode due to a restricted data bus that is 16-bits wide only.

Table III. PMOVLAY Bits

PMOVLAY	Memory	A13	A12:0
0, 4, 5	Internal	Not Applicable	Not Applicable
1	External Overlay 1	0	13 LSBs of Address Between 0x2000 and 0x3FFF
2	External Overlay 2	1	13 LSBs of Address Between 0x2000 and 0x3FFF

Data Memory

Data Memory, *Full Memory Mode* is a 16-bit-wide space used for the storage of data variables and for memory-mapped control registers. The ADSP-2189M has 48K words on Data Memory RAM on-chip. Part of this space is used by 32 memory-mapped registers. Support also exists for up to two 8K external memory overlay spaces through the external data bus. All internal accesses complete in one cycle. Accesses to external memory are timed using the wait-states specified by the DWAIT register and the wait-state mode bit.

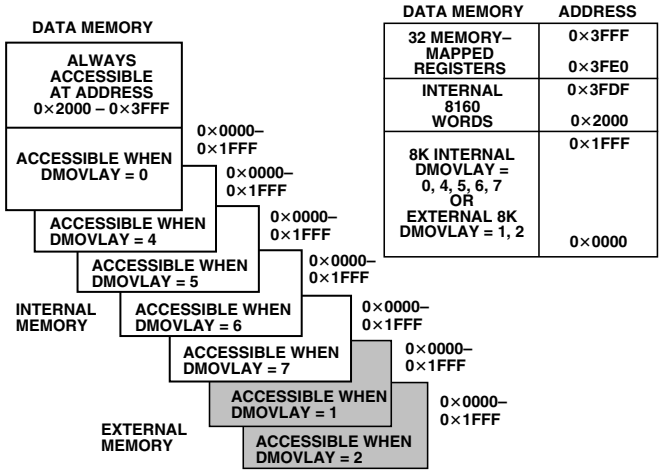

Figure 5. Data Memory Map

Data Memory, *Host Mode* allows access to all internal memory. External overlay access is limited by a single external address line (A0).

Table IV. DMOVLAY Bits

PMOVLAY	Memory	A13	A12:0
0, 4, 5, 6, 7	Internal	Not Applicable	Not Applicable
1	External Overlay 1	0	13 LSBs of Address Between 0x2000 and 0x3FFF
2	External Overlay 2	1	13 LSBs of Address Between 0x2000 and 0x3FFF

Memory Mapped Registers (New to the ADSP-2189M)
The ADSP-2189M has three memory mapped registers that differ from other ADSP-21xx Family DSPs. The slight modifications to these registers (Wait-State Control, Programmable Flag and Composite Select Control and System Control) provide the ADSP-2189M's wait-state and $\overline{\text{BMS}}$ control features.

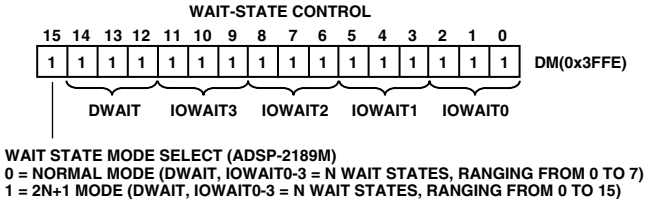

Figure 6. Wait-State Control Register (ADSP-2189M)

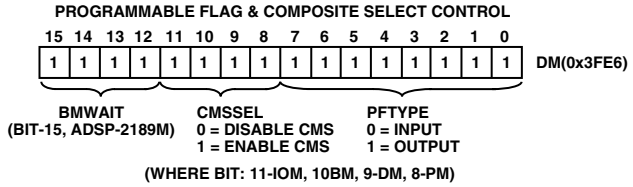

Figure 7. Programmable Flag and Composite Select Control Register

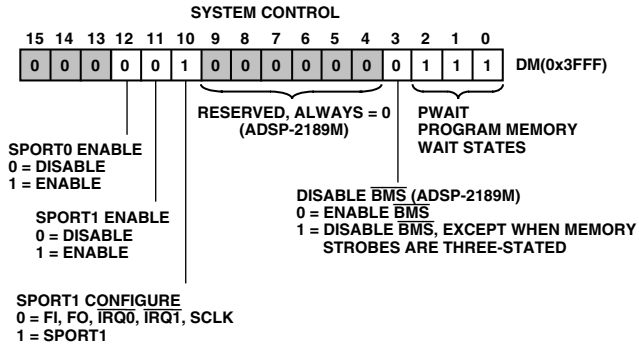

Figure 8. System Control Register

I/O Space (Full Memory Mode)
The ADSP-2189M supports an additional external memory space called I/O space. This space is designed to support simple connections to peripherals (such as data converters and external registers) or to bus interface ASIC data registers. I/O space supports 2048 locations of 16-bit-wide data. The lower eleven bits of the external address bus are used; the upper three bits are undefined. Two instructions were added to the core ADSP-2100 Family instruction set to read from and write to I/O memory space. The I/O space also has four dedicated three-bit wait-state registers, IOWAIT0–3, which, in combination with the wait-state mode bit, specify up to 15 wait-states to be automatically generated for each of four regions. The wait-states act on address ranges as shown in Table V.

Table V. Wait-States

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

Composite Memory Select ($\overline{\text{CMS}}$)
The ADSP-2189M has a programmable memory select signal that is useful for generating memory select signals for memories mapped to more than one space. The $\overline{\text{CMS}}$ signal is generated to have the same timing as each of the individual memory select signals ($\overline{\text{PMS}}$, $\overline{\text{DMS}}$, $\overline{\text{BMS}}$, $\overline{\text{IOMS}}$) but can combine their functionality.

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

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

Byte Memory Select ($\overline{\text{BMS}}$)
The ADSP-2189M's $\overline{\text{BMS}}$ disable feature combined with the $\overline{\text{CMS}}$ pin lets you use multiple memories in the byte memory space. For example, an EPROM could be attached to the $\overline{\text{BMS}}$ select, and an SRAM could be connected to $\overline{\text{CMS}}$. Because $\overline{\text{BMS}}$ is enabled at reset, the EPROM would be used for booting. After booting, software could disable $\overline{\text{BMS}}$ and set the $\overline{\text{CMS}}$ signal to respond to $\overline{\text{BMS}}$, enabling the SRAM.

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

The byte memory space on the ADSP-2189M supports read and write operations as well as four different data formats. The byte memory uses data bits 15:8 for data. The byte memory uses data bits 23:16 and address bits 13:0 to create a 22-bit address. This allows up to a 4 meg × 8 (32 megabit) ROM or RAM to be used without glue logic. All byte memory accesses are timed by the BMWAIT register and the wait-state mode bit.

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

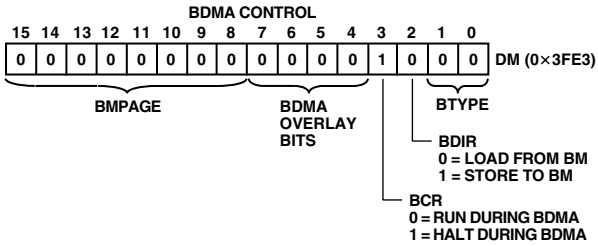

Figure 9. BDMA Control Register

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

Table VI. Data Formats

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

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

BDMA accesses can cross page boundaries during sequential addressing. A BDMA interrupt is generated on the completion of the number of transfers specified by the BWCOUNT register.

The BWCOUNT register is updated after each transfer so it can be used to check the status of the transfers. When it reaches zero, the transfers have finished and a BDMA interrupt is generated. The BMPAGE and BEAD registers must not be accessed by the DSP during BDMA operations.

The source or destination of a BDMA transfer will always be on-chip program or data memory.

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.

The BDMA overlay bits specify the OVLAY memory blocks to be accessed for internal memory.

The BMWAIT field, which has four bits on ADSP-2189M, allows selection of up to 15 wait-states for BDMA transfers.

Internal Memory DMA Port (IDMA Port; Host Memory Mode)
The IDMA Port provides an efficient means of communication between a host system and the ADSP-2189M. 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. A typical IDMA transfer process is described as follows:

1. Host starts IDMA transfer.
2. Host checks $\overline{\text{IACK}}$ control line to see if the DSP is busy.
3. Host uses $\overline{\text{IS}}$ and IAL control lines to latch either the DMA starting address (IDMAA) or the PM/DM OVLAY selection into the DSP's IDMA control registers. If Bit 15 = 1, the value of bits 7:0 represent the IDMA overlay: Bits 14:8 must be set to 0. If Bit 15 = 0, the value of bits 13:0 represent the starting address of internal memory to be accessed and Bit 14 reflects PM or DM for access.
4. Host uses $\overline{\text{IS}}$ and $\overline{\text{IRD}}$ (or $\overline{\text{IWR}}$) to read (or write) DSP internal memory (PM or DM).
5. Host checks $\overline{\text{IACK}}$ line to see if the DSP has completed the previous IDMA operation.
6. Host ends IDMA transfer.

Bus Request and Bus Grant
The ADSP-2189M 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-2189M is not performing an external memory access, 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-2189M will not halt program execution until it encounters an instruction that requires an external memory access.

If the ADSP-2189M is performing an external memory access when the external device asserts the $\overline{BR}$ signal, 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 at which 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-2189M requires the external bus for a memory or BDMA access, but is stopped. The other device can release the bus by deasserting bus request. Once the bus is released, the ADSP-2189M deasserts $\overline{BG}$ and $\overline{BGH}$ and executes the external memory access.

Flag I/O Pins
The ADSP-2189M 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-2189M'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-2189M 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, PF2 and PF3 are also used for device configuration during reset.

INSTRUCTION SET DESCRIPTION

The ADSP-2189M 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-2189M'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-2189M has on-chip emulation support and an ICE-Port, a special set of pins that interface to the EZ-ICE. These features allow in-circuit emulation without replacing the target system processor by using only a 14-pin connection from the target system to the EZ-ICE. Target systems must have a 14-pin connector to accept the EZ-ICE's in-circuit probe, a 14-pin plug.

Issuing the chip reset command during emulation causes the DSP to perform a full chip reset, including a reset of its memory mode. Therefore, it is vital that the mode pins are set correctly PRIOR to issuing a chip reset command from the emulator user interface. If you are using a passive method of maintaining mode information (as discussed in Setting Memory Modes), then it does not matter that the mode information is latched by an emulator reset. However, if using the $\overline{RESET}$ pin as a method of setting the value of the mode pins, the effects of an emulator reset must be taken into consideration.

One method of ensuring that the values located on the mode pins are those desired is to construct a circuit like the one shown in Figure 12. This circuit forces the value located on the Mode A pin to logic high; regardless if it latched via the $\overline{RESET}$ or $\overline{ERESET}$ pin.

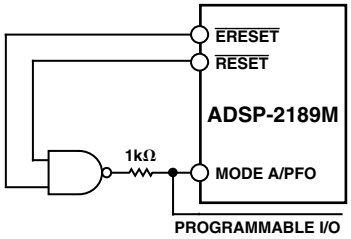

Figure 12. Mode A Pin/EZ-ICE Circuit

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

The ICE-Port interface consists of the following ADSP-2189M pins: $\overline{EBR}$, $\overline{EINT}$, EE, $\overline{EBG}$, ECLK, $\overline{ERESET}$, ELIN, $\overline{EMS}$, and ELOUT.

These ADSP-2189M 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-2189M 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: $\overline{BR}$, $\overline{BG}$, $\overline{RESET}$, and GND.

The EZ-ICE uses the EE (emulator enable) signal to take control of the ADSP-2189M 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 your target system via a ribbon cable and a 14-pin female plug. The female plug is plugged onto the 14-pin connector (a pin strip header) on the target board.

Target Board Connector for EZ-ICE Probe
The EZ-ICE connector (a standard pin strip header) is shown in Figure 13. 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.

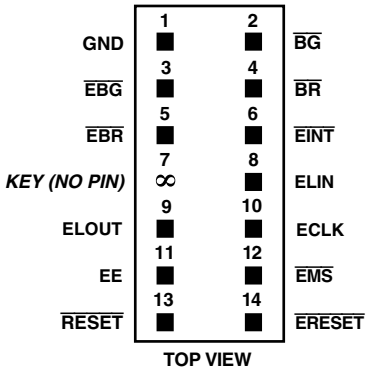

Figure 13. Target Board Connector for EZ-ICE

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 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-2189M ($\overline{RD}$, $\overline{WR}$, $\overline{PMS}$, $\overline{DMS}$, $\overline{BMS}$, $\overline{CMS}$, 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 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 $\overline{RESET}$ 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.

ABSOLUTE MAXIMUM RATINGS[1]

Parameter	Value Min	Max
Internal Supply Voltage (V_{DDINT})	–0.3 V	+3.0 V
External Supply Voltage (V_{DDEXT})	–0.3 V	+4.6 V
Input Voltage[2]	–0.5 V	+4.6 V
Output Voltage Swing[3]	–0.5 V	V_{DDEXT} + 0.5 V
Operating Temperature Range (Ambient)	–40°C	+85°C
Storage Temperature Range	–65°C	+150°C
Lead Temperature (5 sec) LQFP		+280°C

NOTES
[1]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.
[2]Applies to Bidirectional pins (D0–D23, RFS0, RFS1, SCLK0, SCLK1, TFS0, TFS1, A1–A13, PF0–PF7) and Input only pins (CLKIN, $\overline{\text{RESET}}$, $\overline{\text{BR}}$, DR0, DR1, $\overline{\text{PWD}}$).
[3]Applies to Output pins ($\overline{\text{BG}}$, $\overline{\text{PMS}}$, $\overline{\text{DMS}}$, $\overline{\text{BMS}}$, $\overline{\text{IOMS}}$, $\overline{\text{CMS}}$, $\overline{\text{RD}}$, $\overline{\text{WR}}$, $\overline{\text{PWDACK}}$, A0, DT0, DT1, CLKOUT, FL2-0, $\overline{\text{BGH}}$).

CAUTION

ESD (electrostatic discharge) sensitive device. Electrostatic charges as high as 4000 V readily accumulate on the human body and test equipment and can discharge without detection. Although the ADSP-2189M features proprietary ESD protection circuitry, permanent damage may occur on devices subjected to high energy electrostatic discharges. Therefore, proper ESD precautions are recommended to avoid performance degradation or loss of functionality.

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-2189M timing parameters, for your convenience.

Memory Device Specification	Parameter	Timing Parameter Definition[1]
Address Setup to Write Start	t_{ASW}	A0–A13, $\overline{\text{xMS}}$ Setup before $\overline{\text{WR}}$ Low
Address Setup to Write End	t_{AW}	A0–A13, $\overline{\text{xMS}}$ Setup before $\overline{\text{WR}}$ Deasserted
Address Hold Time	t_{WRA}	A0–A13, $\overline{\text{xMS}}$ Hold before $\overline{\text{WR}}$ Low
Data Setup Time	t_{DW}	Data Setup before $\overline{\text{WR}}$ High
Data Hold Time	t_{DH}	Data Hold after $\overline{\text{WR}}$ High
$\overline{\text{OE}}$ to Data Valid	t_{RDD}	$\overline{\text{RD}}$ Low to Data Valid
Address Access Time	t_{AA}	A0–A13, $\overline{\text{xMS}}$ to Data Valid

NOTE
[1]$\overline{\text{xMS}}$ = $\overline{\text{PMS}}$, $\overline{\text{DMS}}$, $\overline{\text{BMS}}$, $\overline{\text{CMS}}$ or $\overline{\text{IOMS}}$.

FREQUENCY DEPENDENCY FOR TIMING SPECIFICATIONS

t_{CK} is defined as $0.5t_{CKI}$. The ADSP-2189M uses an input clock with a frequency equal to half the instruction rate: a 37.50 MHz input clock (which is equivalent to 28 ns) yields a 13 ns processor cycle (equivalent to 75 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 $= 0.5$ (15 ns) $- 7$ ns $= 0.5$ ns

ENVIRONMENTAL CONDITIONS[1]

Rating Description	Symbol	Value
Thermal Resistance		
(Case-to-Ambient)	θ_{CA}	48°C/W
(Junction-to-Ambient)	θ_{JA}	50°C/W
(Junction-to-Case)	θ_{JC}	2°C/W

NOTE
[1]Where the ambient temperature rating (T_{AMB}) is:
$T_{AMB} = T_{CASE} - (PD \times \theta_{CA})$
T_{CASE} = Case temperature in °C
PD = Power dissipation in W.

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 $V_{DDEXT} = 3.3$ V and $t_{CK} = 15$ ns.

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

P_{INT} = internal power dissipation from Power vs. Frequency graph (Figure 15).

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

Parameters	# of Pins	× C	× V_{DDEXT}^2	× f	PD
Address, $\overline{DMS}$	8	10 pF	3.3^2 V	33.3 MHz	29.0 mW
Data Output, $\overline{WR}$	9	10 pF	3.3^2 V	16.67 MHz	16.3 mW
$\overline{RD}$	1	10 pF	3.3^2 V	16.67 MHz	1.8 mW
CLKOUT	1	10 pF	3.3^2 V	33.3 MHz	3.6 mW
					50.7 mW

Total power dissipation for this example is P_{INT} + 50.7 mW.

Output Drive Currents

Figure 14 shows typical I-V characteristics for the output drivers on the ADSP-2189M. The curves represent the current drive capability of the output drivers as a function of output voltage.

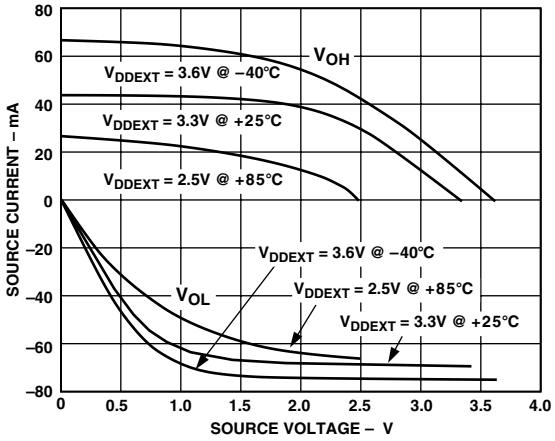

Figure 14. Typical Output Driver Characteristics

Parameter		Min	Max	Unit
Bus Request–Bus Grant				
Timing Requirements:				
t_{BH}	$\overline{BR}$ Hold after CLKOUT High[1]	$0.25t_{CK}$ + 2		ns
t_{BS}	$\overline{BR}$ Setup before CLKOUT Low[1]	$0.25t_{CK}$ + 10		ns
Switching Characteristics:				
t_{SD}	CLKOUT High to $\overline{xMS}$, $\overline{RD}$, $\overline{WR}$ Disable		$0.25t_{CK}$ + 8	ns
t_{SDB}	$\overline{xMS}$, $\overline{RD}$, $\overline{WR}$ Disable to $\overline{BG}$ Low	0		ns
t_{SE}	$\overline{BG}$ High to $\overline{xMS}$, $\overline{RD}$, $\overline{WR}$ Enable	0		ns
t_{SEC}	$\overline{xMS}$, $\overline{RD}$, $\overline{WR}$ Enable to CLKOUT High	$0.25t_{CK}$ – 3		ns
t_{SDBH}	$\overline{xMS}$, $\overline{RD}$, $\overline{WR}$ Disable to $\overline{BGH}$ Low[2]	0		ns
t_{SEH}	$\overline{BGH}$ High to $\overline{xMS}$, $\overline{RD}$, $\overline{WR}$ Enable[2]	0		ns

NOTES
$\overline{xMS}$ = $\overline{PMS}$, $\overline{DMS}$, $\overline{CMS}$, $\overline{IOMS}$, $\overline{BMS}$
[1]$\overline{BR}$ is an asynchronous signal. If $\overline{BR}$ meets the setup/hold requirements, it will be recognized during the current clock cycle; otherwise the signal will be recognized on the following cycle. Refer to the *ADSP-2100 Family User's Manual*, Third Edition, for $\overline{BR}$/$\overline{BG}$ cycle relationships.
[2]$\overline{BGH}$ is asserted when the bus is granted and the processor or BDMA requires control of the bus to continue.

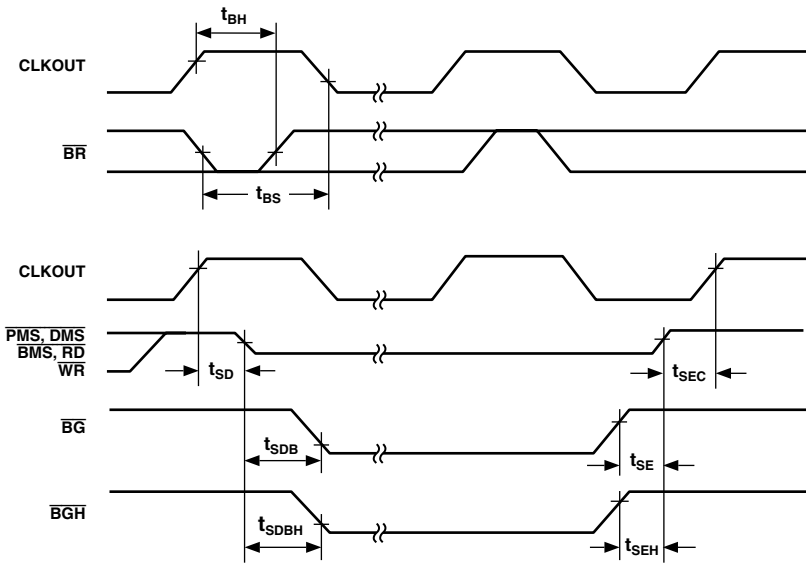

Figure 24. Bus Request–Bus Grant

Parameter		Min	Max	Unit
Memory Write				
Switching Characteristics:				
t_{DW}	Data Setup before $\overline{WR}$ High	$0.5t_{CK} - 4 + w$		ns
t_{DH}	Data Hold after $\overline{WR}$ High	$0.25t_{CK} - 1$		ns
t_{WP}	$\overline{WR}$ Pulsewidth	$0.5t_{CK} - 3 + w$		ns
t_{WDE}	$\overline{WR}$ Low to Data Enabled	0		ns
t_{ASW}	A0–A13, $\overline{xMS}$ Setup before $\overline{WR}$ Low	$0.25t_{CK} - 3$		ns
t_{DDR}	Data Disable before $\overline{WR}$ or $\overline{RD}$ Low	$0.25t_{CK} - 3$		ns
t_{CWR}	CLKOUT High to $\overline{WR}$ Low	$0.25t_{CK} - 2$	$0.25t_{CK} + 4$	ns
t_{AW}	A0–A13, $\overline{xMS}$, Setup before $\overline{WR}$ Deasserted	$0.75t_{CK} - 5 + w$		ns
t_{WRA}	A0–A13, $\overline{xMS}$ Hold after $\overline{WR}$ Deasserted	$0.25t_{CK} - 1$		ns
t_{WWR}	$\overline{WR}$ High to $\overline{RD}$ or $\overline{WR}$ Low	$0.5t_{CK} - 3$		ns

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

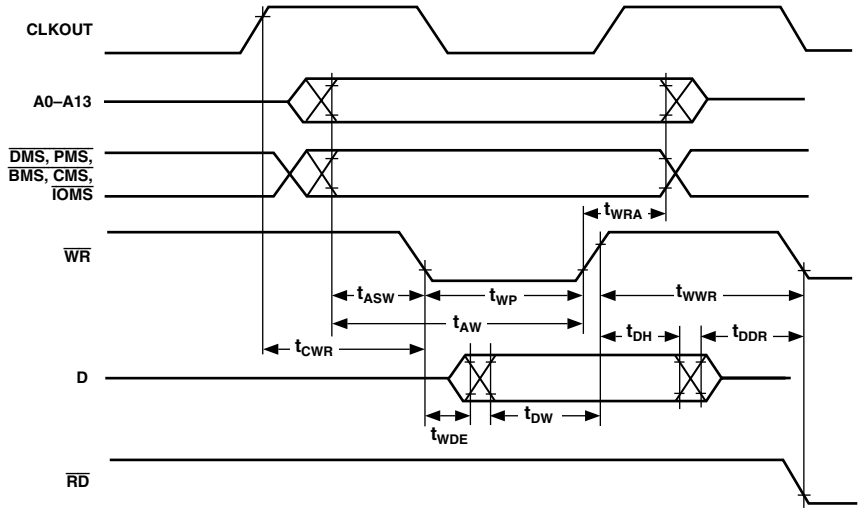

Figure 26. Memory Write

Parameter		Min	Max	Unit
Serial Ports				
Timing Requirements:				
t_{SCK}	SCLK Period	26.67		ns
t_{SCS}	DR/TFS/RFS Setup before SCLK Low	4		ns
t_{SCH}	DR/TFS/RFS Hold after SCLK Low	7		ns
t_{SCP}	SCLKIN Width	12		ns
Switching Characteristics:				
t_{CC}	CLKOUT High to SCLKOUT	$0.25t_{CK}$	$0.25t_{CK}$ + 6	ns
t_{SCDE}	SCLK High to DT Enable	0		ns
t_{SCDV}	SCLK High to DT Valid		12	ns
t_{RH}	TFS/RFS_{OUT} Hold after SCLK High	0		ns
t_{RD}	TFS/RFS_{OUT} Delay from SCLK High		12	ns
t_{SCDH}	DT Hold after SCLK High	0		ns
t_{TDE}	TFS (Alt) to DT Enable	0		ns
t_{TDV}	TFS (Alt) to DT Valid		12	ns
t_{SCDD}	SCLK High to DT Disable		12	ns
t_{RDV}	RFS (Multichannel, Frame Delay Zero) to DT Valid		12	ns

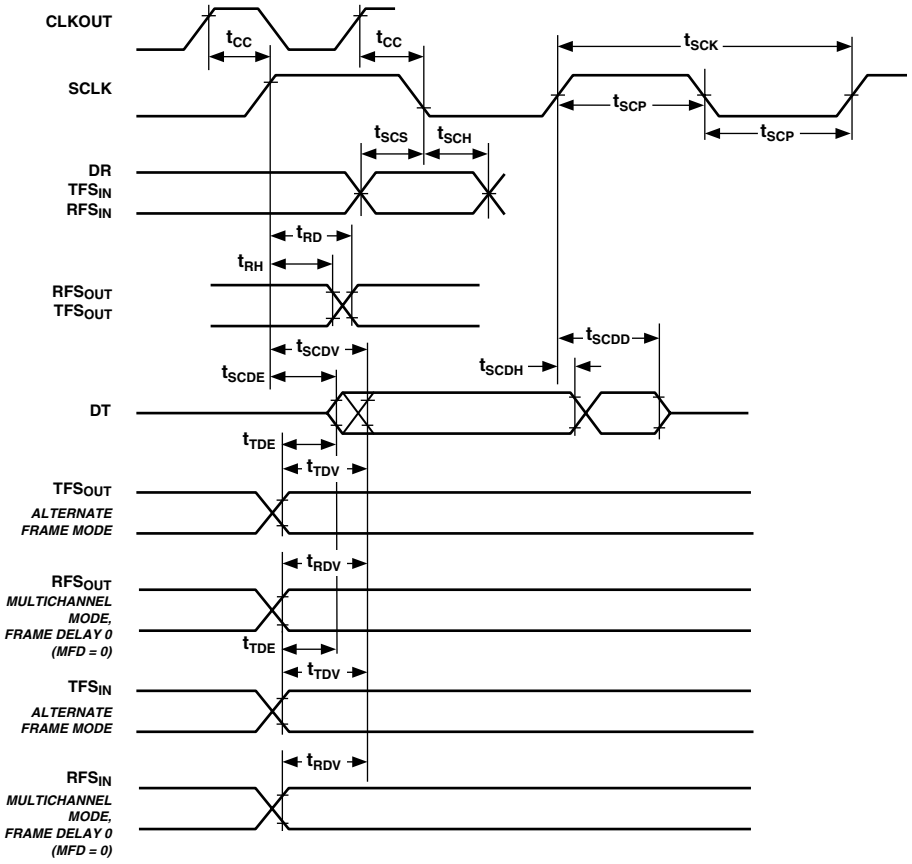

Figure 27. Serial Ports

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

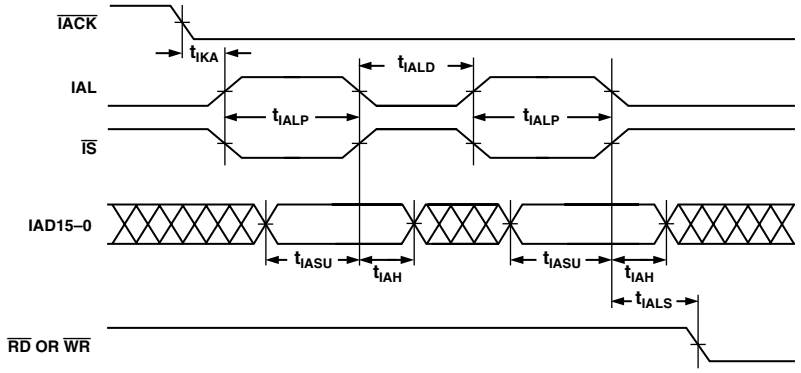

Figure 28. IDMA Address Latch

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 End of Write[2, 3, 4]	$0.5t_{CK}$ + 5		ns
t_{IKH}	IAD15–0 Data Hold after End of Write[2, 3, 4]	0		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		10	ns

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

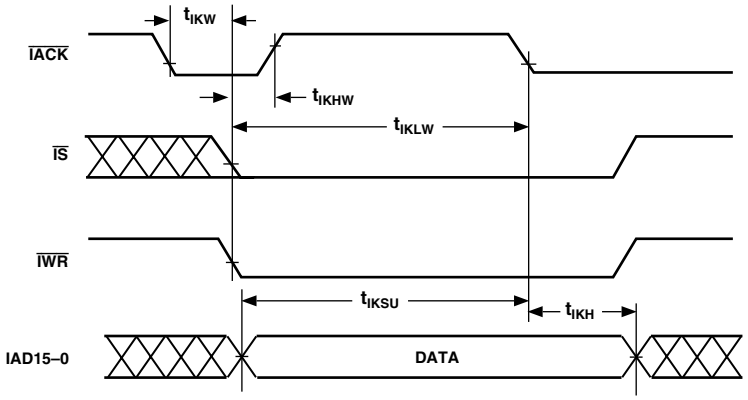

Figure 30. IDMA Write, Long Write Cycle

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	10		ns
Switching Characteristics:				
t_{IKHR}	$\overline{IACK}$ High after Start of Read[1]		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		10	ns

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

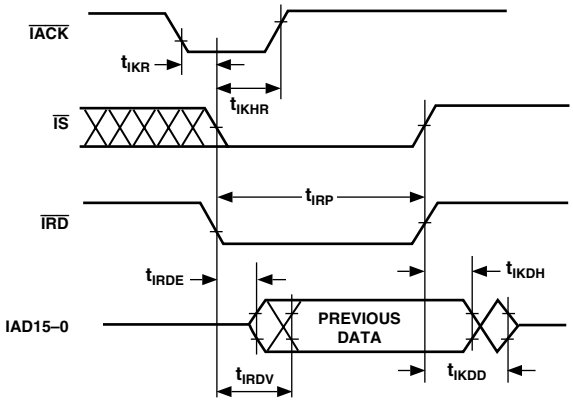

Figure 32. IDMA Read, Short Read Cycle

The ADSP-2189M package pinout appears in the following table. 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 $\overline{\text{RESET}}$.

PIN CONFIGURATION

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