ANALOG DEVICES

# DSP Microcomputer

# Preliminary Technical Data ADSP-2116ON

**SUMMARY**

**High-Performance 32-Bit DSP—Applications in Audio, Medical, Military, Graphics, Imaging, and Communication**

**Super Harvard Architecture—Four Independent Buses for Dual Data Fetch, Instruction Fetch, and Nonintrusive, Zero-Overhead I/O**

**Backwards-Compatible—Assembly Source Level Compatible with Code for ADSP-2106x DSPs**

**Single-Instruction-Multiple-Data (SIMD) Computational Architecture—Two 32-Bit IEEE Floating-Point Computation Units, Each with a Multiplier, ALU, Shifter, and Register File**

**Integrated Peripherals—Integrated I/O Processor, 4 M Bits On-Chip Dual-Ported SRAM, Glueless Multiprocessing Features, and Ports (Serial, Link, External Bus, and JTAG)**

**KEY FEATURES**

**95 MHz (10.5 ns) Core Instruction Rate**

**Single-Cycle Instruction Execution, Including SIMD Operations in Both Computational Units**

**570 MFLOPS Peak and 380 MFLOPS Sustained Performance (Based on FIR)**

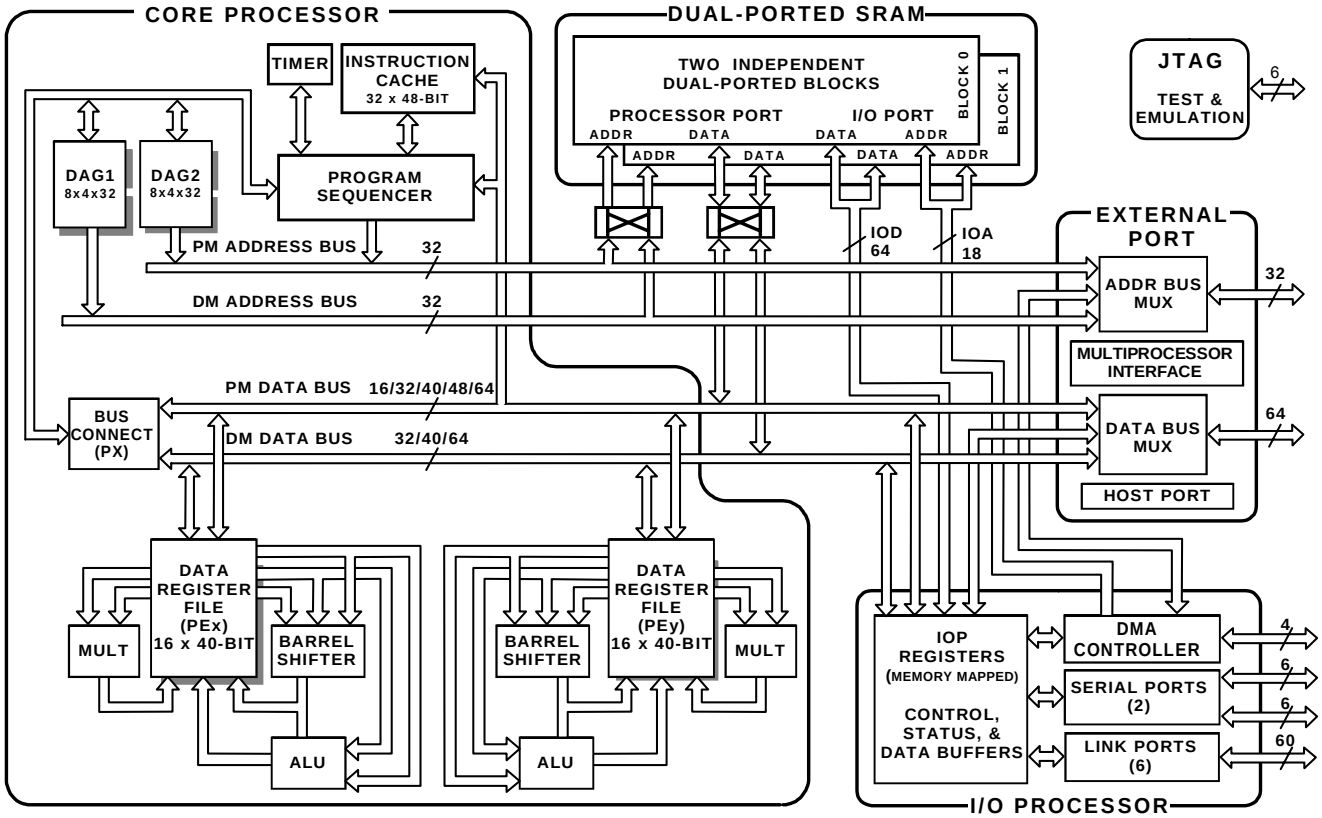
**Dual Data Address Generators (DAGs) with Modulo and Bit-Reverse Addressing**

**Zero-Overhead Looping and Single-Cycle Loop Setup, Providing Efficient Program Sequencing**

**IEEE 1149.1 JTAG Standard Test Access Port and On-Chip Emulation**

**400-Ball 27 × 27 mm Metric PBGA Package**

*Functional Block Diagram*




REV. PrA



**One Technology Way, P.O.Box 9106, Norwood, MA 02062-9106, U.S.A.**
**Tel:781/329-4700 World Wide Web Site: http://www.analog.com**
**Fax:781/326-8703**

**FEATURES (CONTINUED)**

**Single Instruction Multiple Data (SIMD)**
- **Architecture Provides:**
- **Two Computational Processing Elements**
- **Concurrent Execution—Each Processing Element Executes the Same Instruction, but Operates on Different Data**
- **Code Compatibility—at Assembly Level, Uses the Same Instruction Set as the ADSP-2106x SHARC DSPs**

**Parallelism in Buses and Computational Units Allows:**
- **Single-cycle Execution (with or without SIMD) of: A Multiply Operation, An ALU Operation, A Dual Memory Read or Write, and An Instruction Fetch**
- **Transfers Between Memory and Core at up to Four 32-Bit Floating- or Fixed-Point Words per Cycle**
- **Accelerated FFT Butterfly Computation Through a Multiply with Add and Subtract**

**4M Bits On-Chip Dual-Ported SRAM for Independent Access by Core Processor, Host, and DMA**

**DMA Controller supports:**
- **14 Zero-Overhead DMA Channels for Transfers Between ADSP-21160N Internal Memory and External Memory, External Peripherals, Host Processor, Serial Ports, or Link Ports**
- **64-Bit Background DMA Transfers at Core Clock Speed, in Parallel with Full-Speed Processor Execution**
- **665M Bytes/s Transfer Rate Over IOP Bus**
- **Host Processor Interface to 16- and 32-Bit Microprocessors**

**4G Word Address Range for Off-Chip Memory**

**Memory Interface Supports Programmable Wait State Generation and Page-Mode for Off-Chip Memory**

**Multiprocessing Support Provides:**
- **Glueless Connection for Scalable DSP Multiprocessing Architecture**
- **Distributed On-Chip Bus Arbitration for Parallel Bus Connect of up to Six ADSP-21160Ns plus Host**
- **Six Link Ports for Point-To-Point Connectivity and Array Multiprocessing**

**Serial Ports Provide:**
- **Two 47.5M Bits/s Synchronous Serial Ports with Companding Hardware**
- **Independent Transmit and Receive Functions**
- **TDM Support for T1 and E1 Interfaces**
- **64-Bit Wide Synchronous External Port Provides:**
- **Glueless Connection to Asynchronous and SBSRAM External Memories**
- **Up to 47.5 MHz Operation**

## GENERAL DESCRIPTION

The ADSP-21160N SHARC DSP is the second iteration of the ADSP-21160. Built in a 0.18 micron CMOS process, it offers higher performance and lower power consumption than its predecessor, the ADSP-21160M. Easing portability, the ADSP-21160N is application source code compatible with first generation ADSP-2106x SHARC DSPs in SISD (Single Instruction, Single Data) mode. To take advantage of the processor's SIMD (Single Instruction, Multiple Data) capability, some code changes are needed. Like other SHARCs, the ADSP-21160N is a 32-bit processor that is optimized for high performance DSP applications. The ADSP-21160N includes an 95 MHz core, a dual-ported on-chip SRAM, an integrated I/O processor with multiprocessing support, and multiple internal buses to eliminate I/O bottlenecks.

The ADSP-21160N introduces Single-Instruction, Multiple-Data (SIMD) processing. Using two computational units (ADSP-2106x SHARC DSPs have one), the ADSP-21160N can double performance versus the ADSP-2106x on a range of DSP algorithms.

Fabricated in a state of the art, high speed, low power CMOS process, the ADSP-21160N has a 10.5 ns instruction cycle time. With its SIMD computational hardware running at 95 MHz, the ADSP-21160N can perform 570 million math operations per second.

Table 1 shows performance benchmarks for the ADSP-21160N.

**Table 1. ADSP-21160N Benchmarks**

| Benchmark Algorithm | Speed |
|---|---|
| 1024 Point Complex FFT (Radix 4, with reversal) | 96 µs |
| FIR Filter (per tap) | 5.25 ns |
| IIR Filter (per biquad) | 21 ns |
| Matrix Multiply (pipelined) [3×3] × [3×1] | 47.25 ns |
| Matrix Multiply (pipelined) [4×4] × [4×1] | 84 ns |
| Divide (y/x) | 31.5 ns |
| Inverse Square Root | 47.25 ns |
| DMA Transfer Rate | 665M Bytes/s |

These benchmarks provide single-channel extrapolations of measured dual-channel processing performance. For more information on benchmarking and optimizing DSP code for single- and dual-channel processing, see Analog Devices's website.

The ADSP-21160N continues SHARC's industry-leading standards of integration for DSPs, combining a high-performance 32-bit DSP core with integrated, on-chip system features. These features include a 4M-bit dual ported SRAM memory, host processor interface, I/O processor that supports 14 DMA channels, two serial ports, six link ports, external parallel bus, and glueless multiprocessing.

The functional block diagram on page 1 shows a block diagram of the ADSP-21160N, illustrating the following architectural features:

- Two processing elements, each made up of an ALU, Multiplier, Shifter, and Data Register File
- Data Address Generators (DAG1, DAG2)
- Program sequencer with instruction cache
- PM and DM buses capable of supporting four 32-bit data transfers between memory and the core every core processor cycle
- Interval timer
- On-Chip SRAM (4M bits)
- External port that supports:
  - Interfacing to off-chip memory peripherals
  - Glueless multiprocessing support for six ADSP-21160N SHARCs
  - Host port
- DMA controller
- Serial ports and link ports
- JTAG test access port

Figure 1 shows a typical single-processor system. A multiprocessing system appears in Figure 4.

**ADSP-21160N Family Core Architecture**

The ADSP-21160N includes the following architectural features of the ADSP-2116x family core. The ADSP-21160N is code compatible at the assembly level with the ADSP-2106x and ADSP-21161.

***SIMD Computational Engine***

The ADSP-21160N contains two computational processing elements that operate as a Single Instruction Multiple Data (SIMD) engine. The processing elements are referred to as PEX and PEY, and each contains an ALU, multiplier, shifter, and register file. PEX is always active, and PEY may be enabled by setting the PEYEN mode bit in the MODE1 register. When this mode is enabled, the same instruction is executed in both processing elements, but each processing element operates on different data. This architecture is efficient at executing math-intensive DSP algorithms.

Entering SIMD mode also has an effect on the way data is transferred between memory and the processing elements. When in SIMD mode, twice the data bandwidth is required to sustain computational operation in the processing

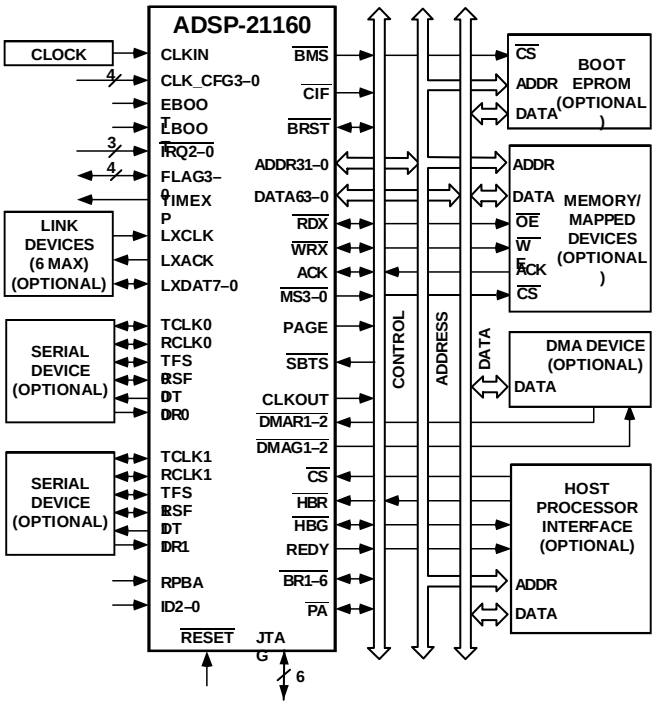



*Figure 1. Single-Processor System*

elements. Because of this requirement, entering SIMD mode also doubles the bandwidth between memory and the processing elements. When using the DAGs to transfer data in SIMD mode, two data values are transferred with each access of memory or the register file.

***Independent, Parallel Computation Units***

Within each processing element is a set of computational units. The computational units consist of an arithmetic/logic unit (ALU), multiplier, and shifter. These units perform single-cycle instructions. The three units within each processing element are arranged in parallel, maximizing computational throughput. Single multifunction instructions execute parallel ALU and multiplier operations. In SIMD mode, the parallel ALU and multiplier operations occur in both processing elements. These computation units support IEEE 32-bit single-precision floating-point, 40-bit extended precision floating-point, and 32-bit fixed-point data formats.

***Data Register File***

A general-purpose data register file is contained in each processing element. The register files transfer data between the computation units and the data buses, and store intermediate results. These 10-port, 32-register (16 primary, 16 secondary) register files, combined with the ADSP-2116x enhanced Harvard architecture, allow unconstrained data flow between computation units and internal memory. The registers in PEX are referred to as R0–R15 and in PEY as S0–S15.

***Single-Cycle Fetch of Instruction and Four Operands***
The ADSP-21160N features an enhanced Harvard architecture in which the data memory (DM) bus transfers data, and the program memory (PM) bus transfers both instructions and data (see the functional block diagram on page 1). With the ADSP-21160N's separate program and data memory buses and on-chip instruction cache, the processor can simultaneously fetch four operands and an instruction (from the cache), all in a single cycle.

***Instruction Cache***
The ADSP-21160N includes an on-chip instruction cache that enables three-bus operation for fetching an instruction and four data values. The cache is selective—only the instructions whose fetches conflict with PM bus data accesses are cached. This cache allows full-speed execution of core, providing looped operations such as digital filter multiply- accumulates and FFT butterfly processing.

***Data Address Generators with Hardware Circular Buffers***
The ADSP-21160N's two data address generators (DAGs) are used for indirect addressing and provide for implementing circular data buffers in hardware. Circular buffers allow efficient programming of delay lines and other data structures required in digital signal processing, and are commonly used in digital filters and Fourier transforms. The two DAGs of the ADSP-21160N contain sufficient registers to allow the creation of up to 32 circular buffers (16 primary register sets, 16 secondary). The DAGs automatically handle address pointer wraparound, reducing overhead, increasing performance, and simplifying implementation. Circular buffers can start and end at any memory location.

***Flexible Instruction Set***
The 48-bit instruction word accommodates a variety of parallel operations, for concise programming. For example, the ADSP-21160N can conditionally execute a multiply, an add, and subtract, in both processing elements, while branching, all in a single instruction.

**ADSP-21160N Memory and I/O Interface Features**
Augmenting the ADSP-2116x family core, the ADSP-21160N adds the following architectural features:

***Dual-Ported On-Chip Memory***
The ADSP-21160N contains four megabits of on-chip SRAM, organized as two blocks of 2M bits each, which can be configured for different combinations of code and data storage. Each memory block is dual-ported for single-cycle, independent accesses by the core processor and I/O processor. The dual-ported memory in combination with three separate on-chip buses allows two data transfers from the core and one from I/O processor, in a single cycle. On the ADSP-21160N, the memory can be configured as a maximum of 128K words of 32-bit data, 256K words of 16-bit data, 85K words of 48-bit instructions (or 40-bit data), or combinations of different word sizes up to four megabits. All of the memory can be accessed as 16-bit, 32-bit, 48-bit, or 64-bit words. A 16-bit floating-point storage format is supported that effectively doubles the amount of data that may be stored on-chip. Conversion between the 32-bit floating-point and 16-bit floating-point formats is done in a single instruction. While each memory block can store combinations of code and data, accesses are most efficient when one block stores data, using the DM bus for transfers, and the other block stores instructions and data, using the PM bus for transfers. Using the DM bus and PM bus in this way, with one dedicated to each memory block, assures single-cycle execution with two data transfers. In this case, the instruction must be available in the cache.

***Off-Chip Memory and Peripherals Interface***
The ADSP-21160N's external port provides the processor's interface to off-chip memory and peripherals. The 4G word off-chip address space is included in the ADSP-21160N's unified address space. The separate on-chip buses—for PM addresses, PM data, DM addresses, DM data, I/O addresses, and I/O data—are multiplexed at the external port to create an external system bus with a single 32-bit address bus and a single 64-bit data bus. The lower 32 bits of the external data bus connect to even addresses and the upper 32 bits of the 64 connect to odd addresses. Every access to external memory is based on an address that fetches a 32-bit word, and with the 64-bit bus, two address locations can be accessed at once. When fetching an instruction from external memory, two 32-bit data locations are being accessed (16 bits are unused). Figure 3 shows the alignment of various accesses to external memory.

The external port supports asynchronous, synchronous, and synchronous burst accesses. ZBT synchronous burst SRAM can be interfaced gluelessly. Addressing of external memory devices is facilitated by on-chip decoding of high-order address lines to generate memory bank select signals. Separate control lines are also generated for simplified addressing of page-mode DRAM. The ADSP-21160N provides programmable memory wait states and external memory acknowledge controls to allow interfacing to DRAM and peripherals with variable access, hold, and disable time requirements.

***DMA Controller***
The ADSP-21160N's on-chip DMA controller allows zero-overhead data transfers without processor intervention. The DMA controller operates independently and invisibly to the processor core, allowing DMA operations to occur while the core is simultaneously executing its program instructions. DMA transfers can occur between the ADSP-21160N's internal memory and external memory, external peripherals, or a host processor. DMA transfers can also occur between the ADSP-21160N's internal memory and its serial ports or link ports. External bus packing to

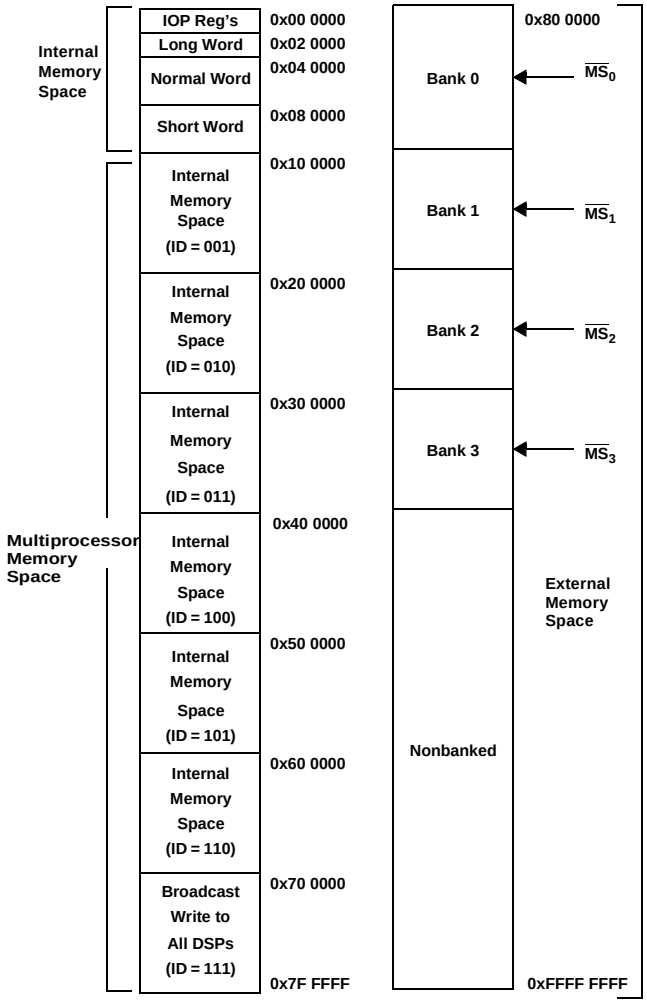



*Figure 2. ADSP-21160N Memory Map*

16-, 32-, 48-, or 64-bit words is performed during DMA transfers. Fourteen channels of DMA are available on the ADSP-21160N—six via the link ports, four via the serial ports, and four via the processor's external port (for either host processor, other ADSP-21160Ns, memory or I/O transfers). Programs can be downloaded to the ADSP-21160N using DMA transfers. Asynchronous off-chip peripherals can control two DMA channels using DMA Request/Grant lines ($\overline{\text{DMAR1–2}}$, $\overline{\text{DMAG1–2}}$). Other DMA features include interrupt generation upon completion of DMA transfers, two-dimensional DMA, and DMA chaining for automatic linked DMA transfers.

***Multiprocessing***

The ADSP-21160N offers powerful features tailored to multiprocessing DSP systems as shown in Figure 4. The external port and link ports provide integrated glueless multiprocessing support.

The external port supports a unified address space (see Figure 2) that allows direct interprocessor accesses of each ADSP-21160N's internal memory. Distributed bus arbitration logic is included on-chip for simple, glueless connection of systems containing up to six ADSP-21160Ns and a host processor. Master processor changeover incurs only one cycle of overhead. Bus arbitration is selectable as either fixed or rotating priority. Bus lock allows indivisible read-modify-write sequences for semaphores. A vector interrupt is provided for interprocessor commands. Maximum throughput for interprocessor data transfer is 380M bytes/s over the external port. Broadcast writes allow simultaneous transmission of data to all ADSP-21160Ns and can be used to implement reflective semaphores.

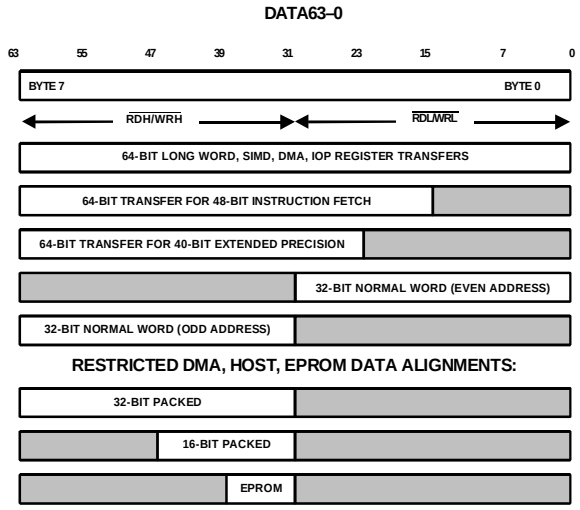



*Figure 3. ADSP-21160N External Data Alignment Options*

Six link ports provide for a second method of multiprocessing communications. Each link port can support communications to another ADSP-21160N. Using the links, a large multiprocessor system can be constructed in a 2D or 3D fashion. Systems can use the link ports and cluster multiprocessing concurrently or independently.

***Link Ports***

The ADSP-21160N features six 8-bit link ports that provide additional I/O capabilities. With the capability of running at 95 MHz rates, each link port can support 95M bytes/s. Link port I/O is especially useful for point-to-point interprocessor communication in multiprocessing systems. The link ports can operate independently and simultaneously. Link port data is packed into 48- or 32-bit words, and can be directly read by the core processor or DMA-transferred to on-chip memory. Each link port has its own double-buffered input and output registers. Clock/acknowledge handshaking controls link port transfers. Transfers are programmable as either transmit or receive.

***Serial Ports***

The ADSP-21160N features two synchronous serial ports that provide an inexpensive interface to a wide variety of digital and mixed-signal peripheral devices. The serial ports

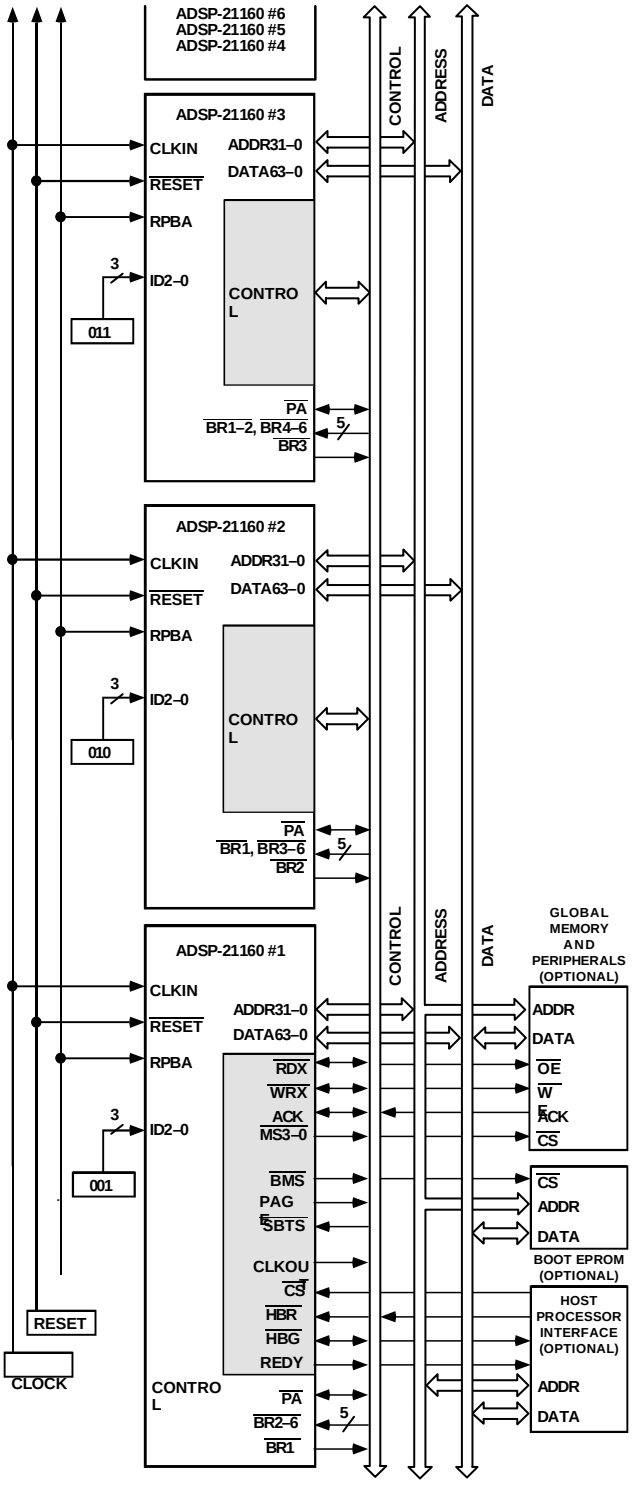



*Figure 4. Shared Memory Multiprocessing System*

can operate up to half the clock rate of the core, providing each with a maximum data rate of 47.5M bit/s. Independent transmit and receive functions provide greater flexibility for serial communications. Serial port data can be automatically transferred to and from on-chip memory via a dedicated DMA. Each of the serial ports offers a TDM multichannel mode. The serial ports can operate with little-endian or big-endian transmission formats, with word lengths selectable from 3 bits to 32 bits. They offer selectable synchronization and transmit modes as well as optional μ-law or A-law companding. Serial port clocks and frame syncs can be internally or externally generated.

### *Host Processor Interface*

The ADSP-21160N host interface allows easy connection to standard microprocessor buses, both 16-bit and 32-bit, with little additional hardware required. The host interface is accessed through the ADSP-21160N's external port and is memory-mapped into the unified address space. Four channels of DMA are available for the host interface; code and data transfers are accomplished with low software overhead. The host processor communicates with the ADSP-21160M's external bus with host bus request ($\overline{HBR}$), host but grant ($\overline{HBG}$), ready (REDY), acknowledge (ACK), and chip select ($\overline{CS}$) signals. The host can directly read and write the internal memory of the ADSP-21160N, and can access the DMA channel setup and mailbox registers. Vector interrupt support provides efficient execution of host commands.

### *Program Booting*

The internal memory of the ADSP-21160N can be booted at system power-up from an 8-bit EPROM, a host processor, or through one of the link ports. Selection of the boot source is controlled by the $\overline{BMS}$ (Boot Memory Select), EBOOT (EPROM Boot), and LBOOT (Link/Host Boot) pins. 32-bit and 16-bit host processors can be used for booting.

### *Phased Locked Loop*

The ADSP-21160N uses an on-chip PLL to generate the internal clock for the core. Ratios of 2:1, 3:1, and 4:1 between the core and CLKIN are supported. The CLK_CFG pins are used to select the ratio. The CLKIN rate is the rate at which the synchronous external port operates.

### *Power Supplies*

The ADSP-21160N has separate power supply connections for the internal ($V_{DDINT}$), external ($V_{DDEXT}$), and analog ($AV_{DD}$/AGND) power supplies. The internal and analog supplies must meet the 1.9 V requirement. The external supply must meet the 3.3 V requirement. All external supply pins must be connected to the same supply.

Note that the analog supply ($AV_{DD}$) powers the ADSP-21160N's clock generator PLL. To produce a stable clock, the system must provide an external circuit to filter the power input to the $AV_{DD}$ pin. Place the filter as close as possible to the pin. For an example circuit, see Figure 5. To prevent noise coupling, use a wide trace for the analog ground (AGND) signal and install a decoupling capacitor as close as possible to the pin.

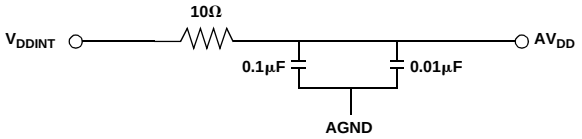



*Figure 5. Analog Power ($AV_{DD}$) Filter Circuit*

**Development Tools**

The ADSP-21160N is supported with a complete set of software and hardware development tools, including Analog Devices' emulators and VisualDSP++[1] development environment. The same emulator hardware that supports other ADSP-2116x DSPs, also fully emulates the ADSP-21160N.

The VisualDSP++ project management environment lets programmers develop and debug an application. This environment includes an easy-to-use assembler that is based on an algebraic syntax; an archiver (librarian/library builder), a linker, a loader, a cycle-accurate instruction-level simulator, a C/C++ compiler, and a C/C++ run-time library that includes DSP and mathematical functions. Two key points for these tools are:

- Compiled ADSP-2116x C/C++ code efficiency—the compiler has been developed for efficient translation of C/C++ code to ADSP-2116x assembly. The DSP has architectural features that improve the efficiency of compiled C/C++ code.
- ADSP-2106x family code compatibility—The assembler has legacy features to ease the conversion of existing ADSP-2106x applications to the ADSP-2116x.

Debugging both C/C++ and assembly programs with the VisualDSP++ debugger, programmers can:

- View mixed C/C++ and assembly code (interleaved source and object information)
- Insert break points
- Set conditional breakpoints on registers, memory, and stacks
- Trace instruction execution
- Perform linear or statistical profiling of program execution
- Fill, dump, and graphically plot the contents of memory
- Source level debugging
- Create custom debugger windows

The VisualDSP++ IDE lets programmers define and manage DSP software development. Its dialog boxes and property pages let programmers configure and manage all of the ADSP-2116x development tools, including the syntax highlighting in the VisualDSP++ editor. This capability permits:

- Control how the development tools process inputs and generate outputs.
- Maintain a one-to-one correspondence with the tool's command line switches.

Analog Devices' DSP emulators use the IEEE 1149.1 JTAG test access port of the ADSP-21160N processor to monitor and control the target board processor during emulation. The emulator provides full-speed emulation, allowing inspection and modification of memory, registers, and processor stacks. Nonintrusive in-circuit emulation is assured by the use of the processor's JTAG interface—the emulator does not affect target system loading or timing.

In addition to the software and hardware development tools available from Analog Devices, third parties provide a wide range of tools supporting the ADSP-2116x processor family. Hardware tools include ADSP-2116x PC plug-in cards. Third Party software tools include DSP libraries, real-time operating systems, and block diagram design tools.

***Designing an Emulator-Compatible DSP Board (Target)***

The White Mountain DSP (Product Line of Analog Devices, Inc.) family of emulators are tools that every DSP developer needs to test and debug hardware and software systems. Analog Devices has supplied an IEEE 1149.1 JTAG Test Access Port (TAP) on each JTAG DSP. The emulator uses the TAP to access the internal features of the DSP, allowing the developer to load code, set breakpoints, observe variables, observe memory, and examine registers. The DSP must be halted to send data and commands, but once an operation has been completed by the emulator, the DSP system is set running at full speed with no impact on system timing.

To use these emulators, the target's design must include the interface between an Analog Devices' JTAG DSP and the emulation header on a custom DSP target board.

***Target Board Header***

The emulator interface to an Analog Devices' JTAG DSP is a 14-pin header, as shown in Figure 6. The customer must supply this header on the target board in order to communicate with the emulator. The interface consists of a standard dual row 0.025" square post header, set on 0.1" × 0.1" spacing, with a minimum post length of 0.235". Pin 3 is the key position used to prevent the pod from being inserted backwards. This pin must be clipped on the target board.

[1]VisualDSP++ is a registered trademark of Analog Devices, Inc.

Also, the clearance (length, width, and height) around the header must be considered. Leave a clearance of at least 0.15" and 0.10" around the length and width of the header, and reserve a height clearance to attach and detach the pod connector.

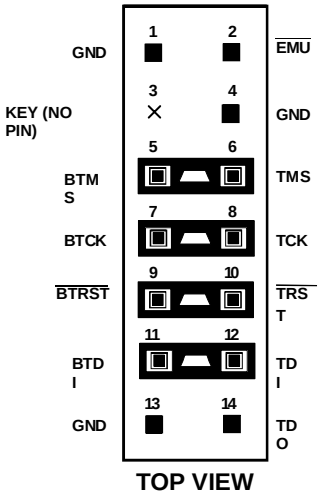



*Figure 6. JTAG Target Board Connector for JTAG Equipped Analog Devices DSP (Jumpers in Place)*

As can be seen in Figure 6, there are two sets of signals on the header. There are the standard JTAG signals TMS, TCK, TDI, TDO, $\overline{\text{TRST}}$, and $\overline{\text{EMU}}$ used for emulation purposes (via an emulator). There are also secondary JTAG signals BTMS, BTCK, BTDI, and $\overline{\text{BTRST}}$ that are optionally used for board-level (boundary scan) testing.

When the emulator is not connected to this header, place jumpers across BTMS, BTCK, $\overline{\text{BTRST}}$, and BTDI as shown in Figure 7. This holds the JTAG signals in the correct state to allow the DSP to run free. Remove all the jumpers when connecting the emulator to the JTAG header.

### *JTAG Emulator Pod Connector*

Figure 8 details the dimensions of the JTAG pod connector at the 14-pin target end. Figure 9 displays the keep-out area for a target board header. The keep-out area allows the pod connector to properly seat onto the target board header. This board area should contain no components (chips, resistors, capacitors, etc.). The dimensions are referenced to the center of the 0.25" square post pin.

### *Design-for-Emulation Circuit Information*

For details on target board design issues including: single processor connections, multiprocessor scan chains, signal buffering, signal termination, and emulator pod logic, see the *EE-68: Analog Devices JTAG Emulation Technical Reference* on the Analog Devices website—use site search on

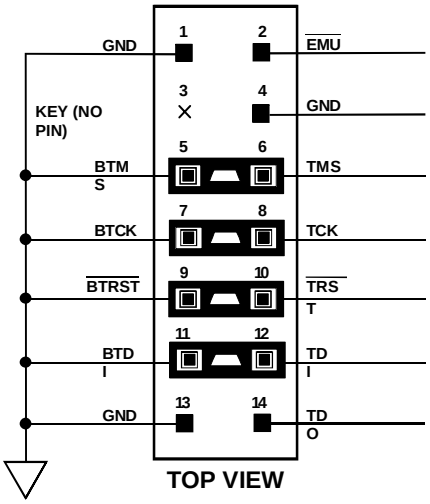



*Figure 7. JTAG Target Board Connector with No Local Boundary Scan*

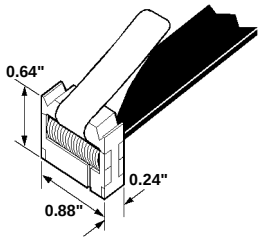



*Figure 8. JTAG Pod Connector Dimensions*

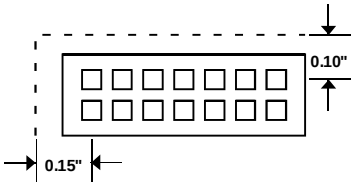



*Figure 9. JTAG Pod Connector Keep-Out Area*

"EE-68" (www.analog.com). This document is updated regularly to keep pace with improvements to emulator support.

**Additional Information**

This data sheet provides a general overview of the ADSP-21160N architecture and functionality. For detailed information on the ADSP-2116x Family core architecture and instruction set, refer to the ADSP-2116x *SHARC DSP Hardware Reference*.

**PIN FUNCTION DESCRIPTIONS**

ADSP-21160N pin definitions are listed below. Inputs identified as synchronous (S) must meet timing requirements with respect to CLKIN (or with respect to TCK for TMS, TDI). Inputs identified as asynchronous (A) can be asserted asynchronously to CLKIN (or to TCK for $\overline{\text{TRST}}$).

Tie or pull unused inputs to VDD or GND, except for the following:

- ADDR31–0, DATA63–0, PAGE, BRST, CLKOUT (ID2–0 = 00x) (NOTE: These pins have a logic-level hold circuit enabled on the ADSP-21160N DSP with ID2–0 = 00x)
- $\overline{\text{PA}}$, ACK, MS3–0, $\overline{\text{RDx}}$,$\overline{\text{WRx}}$, $\overline{\text{CIF}}$, $\overline{\text{EMU}}$, $\overline{\text{DMARx}}$, $\overline{\text{DMAGx}}$, TMS, $\overline{\text{TRST}}$, TDI (ID2–0 = 00x) (NOTE: These pins have a pull-up enabled on the ADSP-21160N DSP with ID2–0 = 00x)
- LxCLK, LxACK, LxDAT7–0 (LxPDRDE = 0) (NOTE: See Link Port Buffer Control Register Bit definitions in the *ADSP-21160 DSP Hardware Reference*).
- DTx, DRx, TCLKx, RCLKx (NOTE: These pins have a pull-up.)

The following symbols appear in the Type column of Table 2: A = Asynchronous, G = Ground, I = Input, O = Output, P = Power Supply, S = Synchronous, (A/D) = Active Drive, (O/D) = Open Drain, and T = Three-State (when $\overline{\text{SBTS}}$ is asserted, or when the ADSP-21160N is a bus slave).

**Table 2. Pin Function Descriptions**

| Pin | Type | Function |
|---|---|---|
| ADDR31–0 | I/O/T | External Bus Address. The ADSP-21160N outputs addresses for external memory and peripherals on these pins. In a multiprocessor system, the bus master outputs addresses for read/writes of the internal memory or IOP registers of other ADSP-21160Ns. The ADSP-21160N inputs addresses when a host processor or multiprocessing bus master is reading or writing its internal memory or IOP registers. A keeper latch on the DSP's ADDR31–0 pins maintains the input at the level it was last driven (only enabled on the ADSP-21160N with ID2–0 = 00x). |
| DATA63–0 | I/O/T | External Bus Data. The ADSP-21160N inputs and outputs data and instructions on these pins. Pull-up resistors on unused DATA pins are not necessary. A keeper latch on the DSP's DATA63-0 pins maintains the input at the level it was last driven (only enabled on the ADSP-21160N with ID2–0 = 00x). |
| $\overline{\text{MS3–0}}$ | O/T | Memory Select Lines. These outputs are asserted (low) as chip selects for the corresponding banks of external memory. Memory bank size must be defined in the SYSCON control register. The $\overline{\text{MS3–0}}$ outputs are decoded memory address lines. In asyn- chronous access mode, the $\overline{\text{MS3–0}}$ outputs transition with the other address outputs. In synchronous access modes, the $\overline{\text{MS3–0}}$ outputs assert with the other address lines; however, they de-assert after the first CLKIN cycle in which ACK is sampled asserted. |
| $\overline{\text{RDL}}$ | I/O/T | Memory Read Low Strobe. $\overline{\text{RDL}}$ is asserted whenever ADSP-21160N reads from the low word of external memory or from the internal memory of other ADSP-21160Ns. External devices, including other ADSP-21160Ns, must assert $\overline{\text{RDL}}$ for reading from the low word of ADSP-21160N internal memory. In a multiprocessing system, $\overline{\text{RDL}}$ is driven by the bus master. |
| $\overline{\text{RDH}}$ | I/O/T | Memory Read High Strobe. $\overline{\text{RDH}}$ is asserted whenever ADSP-21160N reads from the high word of external memory or from the internal memory of other ADSP-21160Ns. External devices, including other ADSP-21160Ns, must assert $\overline{\text{RDH}}$ for reading from the high word of ADSP-21160N internal memory. In a multiprocessing system, $\overline{\text{RDH}}$ is driven by the bus master. |

**Table 2. Pin Function Descriptions (Continued)**

| Pin | Type | Function |
|---|---|---|
| $\overline{\text{WRL}}$ | I/O/T | Memory Write Low Strobe. $\overline{\text{WRL}}$ is asserted when ADSP-21160N writes to the low word of external memory or internal memory of other ADSP-21160Ns. External devices must assert $\overline{\text{WRL}}$ for writing to ADSP-21160N's low word of internal memory. In a multiprocessing system, $\overline{\text{WRL}}$ is driven by the bus master. |
| $\overline{\text{WRH}}$ | I/O/T | Memory Write High Strobe. $\overline{\text{WRH}}$ is asserted when ADSP-21160N writes to the high word of external memory or internal memory of other ADSP-21160Ns. External devices must assert $\overline{\text{WRH}}$ for writing to ADSP-21160N's high word of internal memory. In a multiprocessing system, $\overline{\text{WRH}}$ is driven by the bus master. |
| PAGE | O/T | DRAM Page Boundary. The ADSP-21160N asserts this pin to signal that an external DRAM page boundary has been crossed. DRAM page size must be defined in the ADSP-21160N's memory control register (WAIT). DRAM can only be implemented in external memory Bank 0; the PAGE signal can only be activated for Bank 0 accesses. In a multiprocessing system PAGE is output by the bus master. A keeper latch on the DSP's PAGE pin maintains the output at the level it was last driven (only enabled on the ADSP-21160N with ID2–0 = 00x). |
| BRST | I/O/T | Sequential Burst Access. BRST is asserted by ADSP-21160N or a host to indicate that data associated with consecutive addresses is being read or written. A slave device samples the initial address and increments an internal address counter after each transfer. The incremented address is not pipelined on the bus. If the burst access is a read from host to ADSP-21160N, ADSP-21160N automatically increments the address as long as BRST is asserted. BRST is asserted after the initial access of a burst transfer. It is asserted for every cycle after that, except for the last data request cycle (denoted by $\overline{\text{RDx}}$ or $\overline{\text{WRx}}$ asserted and BRST negated). A keeper latch on the DSP's BRST pin maintains the input at the level it was last driven (only enabled on the ADSP-21160N with ID2–0 = 00x). |
| ACK | I/O/S | Memory Acknowledge. External devices can de-assert ACK (low) to add wait states to an external memory access. ACK is used by I/O devices, memory controllers, or other peripherals to hold off completion of an external memory access. The ADSP-21160N deasserts ACK as an output to add wait states to a synchronous access of its internal memory. A keeper latch on the DSP's ACK pin maintains the input at the level it was last driven (only enabled on the ADSP-21160N with ID2–0 = 00x). |
| $\overline{\text{SBTS}}$ | I/S | Suspend Bus and Three-State. External devices can assert $\overline{\text{SBTS}}$ (low) to place the external bus address, data, selects, and strobes in a high impedance state for the following cycle. If the ADSP-21160N attempts to access external memory while $\overline{\text{SBTS}}$ is asserted, the processor will halt and the memory access will not be completed until $\overline{\text{SBTS}}$ is deasserted. $\overline{\text{SBTS}}$ should only be used to recover from host processor and/or ADSP-21160N deadlock or used with a DRAM controller. |
| $\overline{\text{IRQ2–0}}$ | I/A | Interrupt Request Lines. These are sampled on the rising edge of CLKIN and may be either edge-triggered or level-sensitive. |
| FLAG3–0 | I/O/A | Flag Pins. Each is configured via control bits as either an input or output. As an input, it can be tested as a condition. As an output, it can be used to signal external peripherals. |
| TIMEXP | O | Timer Expired. Asserted for four CLKIN cycles when the timer is enabled and TCOUNT decrements to zero. |

**Table 2. Pin Function Descriptions (Continued)**

| Pin | Type | Function |
|---|---|---|
| $\overline{\text{HBR}}$ | I/A | Host Bus Request. Must be asserted by a host processor to request control of the ADSP-21160N's external bus. When $\overline{\text{HBR}}$ is asserted in a multiprocessing system, the ADSP-21160N that is bus master will relinquish the bus and assert $\overline{\text{HBG}}$. To relinquish the bus, the ADSP-21160N places the address, data, select, and strobe lines in a high impedance state. $\overline{\text{HBR}}$ has priority over all ADSP-21160N bus requests ($\overline{\text{BR6–1}}$) in a multiprocessing system. |
| $\overline{\text{HBG}}$ | I/O | Host Bus Grant. Acknowledges an $\overline{\text{HBR}}$ bus request, indicating that the host processor may take control of the external bus. $\overline{\text{HBG}}$ is asserted (held low) by the ADSP-21160N until $\overline{\text{HBR}}$ is released. In a multiprocessing system, $\overline{\text{HBG}}$ is output by the ADSP-21160N bus master and is monitored by all others. |
| $\overline{\text{CS}}$ | I/A | Chip Select. Asserted by host processor to select the ADSP-21160N. |
| REDY | O (O/D) | Host Bus Acknowledge. The ADSP-21160N deasserts REDY (low) to add waitstates to a host access when $\overline{\text{CS}}$ and $\overline{\text{HBR}}$ inputs are asserted. |
| $\overline{\text{DMAR1}}$ | I/A | DMA Request 1 (DMA Channel 11). Asserted by external port devices to request DMA services. |
| $\overline{\text{DMAR2}}$ | I/A | DMA Request 2 (DMA Channel 12). Asserted by external port devices to request DMA services. |
| ID2–0 | I | Multiprocessing ID. Determines which multiprocessing bus request ($\overline{\text{BR1}}$–$\overline{\text{BR6}}$) is used by ADSP-21160N. ID = 001 corresponds to $\overline{\text{BR1}}$, ID = 010 corresponds to $\overline{\text{BR2}}$, and so on. Use ID = 000 or ID = 001 in single-processor systems. These lines are a system configuration selection which should be hardwired or only changed at reset. |
| $\overline{\text{DMAG1}}$ | O/T | DMA Grant 1 (DMA Channel 11). Asserted by ADSP-21160N to indicate that the requested DMA starts on the next cycle. Driven by bus master only. |
| $\overline{\text{DMAG2}}$ | O/T | DMA Grant 2 (DMA Channel 12). Asserted by ADSP-21160N to indicate that the requested DMA starts on the next cycle. Driven by bus master only. |
| $\overline{\text{BR6–1}}$ | I/O/S | Multiprocessing Bus Requests. Used by multiprocessing ADSP-21160Ns to arbitrate for bus mastership. An ADSP-21160N only drives its own $\overline{\text{BRx}}$ line (corresponding to the value of its ID2–0 inputs) and monitors all others. In a multiprocessor system with less than six ADSP-21160Ns, the unused $\overline{\text{BRx}}$ pins should be pulled high; the processor's own $\overline{\text{BRx}}$ line must not be pulled high or low because it is an output. |
| RPBA | I/S | Rotating Priority Bus Arbitration Select. When RPBA is high, rotating priority for multiprocessor bus arbitration is selected. When RPBA is low, fixed priority is selected. This signal is a system configuration selection which must be set to the same value on every ADSP-21160N. If the value of RPBA is changed during system operation, it must be changed in the same CLKIN cycle on every ADSP-21160N. |
| $\overline{\text{PA}}$ | I/O/T | Priority Access. Asserting its $\overline{\text{PA}}$ pin allows an ADSP-21160N bus slave to interrupt background DMA transfers and gain access to the external bus. $\overline{\text{PA}}$ is connected to all ADSP-21160Ns in the system. If access priority is not required in a system, the $\overline{\text{PA}}$ pin should be left unconnected. |
| DTx | O | Data Transmit (Serial Ports 0, 1). Each DT pin has a 50 kΩ internal pull-up resistor. |
| DRx | I | Data Receive (Serial Ports 0, 1). Each DR pin has a 50 kΩ internal pull-up resistor. |

**Table 2. Pin Function Descriptions (Continued)**

| Pin | Type | Function |
|---|---|---|
| TCLKx | I/O | Transmit Clock (Serial Ports 0, 1). Each TCLK pin has a 50 kΩ internal pull-up resistor. |
| RCLKx | I/O | Receive Clock (Serial Ports 0, 1). Each RCLK pin has a 50 kΩ internal pull-up resistor. |
| TFSx | I/O | Transmit Frame Sync (Serial Ports 0, 1). |
| RFSx | I/O | Receive Frame Sync (Serial Ports 0, 1). |
| LxDAT7–0 | I/O | Link Port Data (Link Ports 0–5). Each LxDAT pin has a 50 kΩ internal pull-down resistor that is enabled or disabled by the LPDRD bit of the LCTL0–1 register. |
| LxCLK | I/O | Link Port Clock (Link Ports 0–5). Each LxCLK pin has a 50 kΩ internal pull-down resistor that is enabled or disabled by the LPDRD bit of the LCTL0–1 register. |
| LxACK | I/O | Link Port Acknowledge (Link Ports 0–5). Each LxACK pin has a 50 kΩ internal pull-down resistor that is enabled or disabled by the LPDRD bit of the LCOM register. |
| EBOOT | I | EPROM Boot Select. For a description of how this pin operates, see Table 3. This signal is a system configuration selection that should be hardwired. |
| LBOOT | I | Link Boot. For a description of how this pin operates, see Table 3. This signal is a system configuration selection that should be hardwired. |
| $\overline{\text{BMS}}$ | I/O/T | Boot Memory Select. Serves as an output or input as selected with the EBOOT and LBOOT pins; see Table 3. This input is a system configuration selection that should be hardwired. |
| CLKIN | I | Local Clock In. CLKIN is the ADSP-21160N clock input. The ADSP-21160N external port cycles at the frequency of CLKIN. The instruction cycle rate is a multiple of the CLKIN frequency; it is programmable at power-up. CLKIN may not be halted, changed, or operated below the specified frequency. |
| CLK_CFG3–0 | I | Core/CLKIN Ratio Control. ADSP-21160N core clock (instruction cycle) rate is equal to n × CLKIN where n is user-selectable to 2, 3, or 4, using the CLK_CFG3–0 inputs. For clock configuration definitions, see the *$\overline{\text{RESET}}$ & CLKIN* section of the *System Design* chapter of the *ADSP-21160 SHARC DSP Hardware Reference* manual. |
| CLKOUT | O/T | Local Clock Out. CLKOUT is driven at the CLKIN frequency by the current bus master. This output is three-stated when the ADSP-21160N is not the bus master, or when the host controls the bus ($\overline{\text{HBG}}$ asserted). A keeper latch on the DSP's CLKOUT pin maintains the output at the level it was last driven (only enabled on the ADSP-21160N with ID2–0 = 00x). |
| $\overline{\text{RESET}}$ | I/A | Processor Reset. Resets the ADSP-21160N to a known state and begins execution at the program memory location specified by the hardware reset vector address. The $\overline{\text{RESET}}$ input must be asserted (low) at power-up. |
| TCK | I | Test Clock (JTAG). Provides a clock for JTAG boundary scan. |
| TMS | I/S | Test Mode Select (JTAG). Used to control the test state machine. TMS has a 20 kΩ internal pull-up resistor. |
| TDI | I/S | Test Data Input (JTAG). Provides serial data for the boundary scan logic. TDI has a 20 kΩ internal pull-up resistor. |
| TDO | O | Test Data Output (JTAG). Serial scan output of the boundary scan path. |

**Table 2. Pin Function Descriptions (Continued)**

| Pin | Type | Function |
|---|---|---|
| $\overline{\text{TRST}}$ | I/A | Test Reset (JTAG). Resets the test state machine. $\overline{\text{TRST}}$ must be asserted (pulsed low) after power-up or held low for proper operation of the ADSP-21160N. $\overline{\text{TRST}}$ has a 20 kΩ internal pull-up resistor. |
| $\overline{\text{EMU}}$ | O (O/D) | Emulation Status. Must be connected to the ADSP-21160N emulator target board connector only. EMU has a 50 kΩ internal pull-up resistor. |
| $\overline{\text{CIF}}$ | O/T | Core Instruction Fetch. Signal is active low when an external instruction fetch is performed. Driven by bus master only. Three-state when host is bus master. |
| $V_{DDINT}$ | P | Core Power Supply. Nominally 1.9 V dc and supplies the DSP's core processor (40 pins). |
| $V_{DDEXT}$ | P | I/O Power Supply. Nominally 3.3 V dc (46 pins). |
| $AV_{DD}$ | P | Analog Power Supply. Nominally 1.9 V dc and supplies the DSP's internal PLL (clock generator). This pin has the same specifications as $V_{DDINT}$, except that added filtering circuitry is required. For more information, see Power Supplies on page 6. |
| AGND | G | Analog Power Supply Return. |
| GND | G | Power Supply Return. (83 pins) |
| NC | | Do Not Connect. Reserved pins that must be left open and unconnected (5 pins). |

**Table 3. Boot Mode Selection**

| EBOOT | LBOOT | $\overline{\text{BMS}}$ | Booting Mode |
|---|---|---|---|
| 1 | 0 | Output | EPROM (Connect $\overline{\text{BMS}}$ to EPROM chip select.) |
| 0 | 0 | 1 (Input) | Host Processor |
| 0 | 1 | 1 (Input) | Link Port |
| 0 | 0 | 0 (Input) | No Booting. Processor executes from external memory. |
| 0 | 1 | 0 (Input) | Reserved |
| 1 | 1 | x (Input) | Reserved |

# ADSP-2116ON SPECIFICATIONS

## RECOMMENDED OPERATING CONDITIONS

| Signal | Parameter[1] | C Grade | | K Grade | | Unit |
|---|---|---|---|---|---|---|
| | | Min | Max | Min | Max | |
| $V_{DDINT}$ | Internal (Core) Supply Voltage | 1.8 | 2.0 | 1.8 | 2.0 | V |
| $AV_{DD}$ | Analog (PLL) Supply Voltage | 1.8 | 2.0 | 1.8 | 2.0 | V |
| $V_{DDEXT}$ | External (I/O) Supply Voltage | 3.13 | 3.47 | 3.13 | 3.47 | V |
| $T_{CASE}$ | Case Operating Temperature[2] | –40 | +100 | 0 | 85 | °C |
| $V_{IH1}$ | High Level Input Voltage[3], @ $V_{DDEXT}$=Max | 2.2 | $V_{DDEXT}$+0.5 | 2.2 | $V_{DDEXT}$+0.5 | V |
| $V_{IH2}$ | High Level Input Voltage[4], @ $V_{DDEXT}$=Max | 2.3 | $V_{DDEXT}$+0.5 | 2.3 | $V_{DDEXT}$+0.5 | V |
| $V_{IL}$ | Low Level Input Voltage[3,4], @ $V_{DDEXT}$=Min | –0.5 | 0.8 | –0.5 | 0.8 | V |

[1] Specifications subject to change without notice.

[2]See Environmental Conditions on page 47 for information on thermal specifications.

[3]Applies to input and bidirectional pins: DATA63–0, ADDR31–0, $\overline{RDx}$, $\overline{WRx}$, ACK, $\overline{SBTS}$, $\overline{IRQ2–0}$, FLAG3–0, $\overline{HBG}$, $\overline{CS}$, $\overline{DMAR1}$, $\overline{DMAR2}$, $\overline{BR6–1}$, ID2–0, RPBA, $\overline{PA}$, BRST, TFS0, TFS1, RFS0, RFS1, LxDAT3–0, LxCLK, LxACK, EBOOT, LBOOT, $\overline{BMS}$, TMS, TDI, TCK, $\overline{HBR}$, DR0, DR1, TCLK0, TCLK1, RCLK0, RCLK1.

[4]Applies to input pins: CLKIN, $\overline{RESET}$, $\overline{TRST}$.

## ELECTRICAL CHARACTERISTICS

| Parameter[1] | | Test Conditions | C and K Grades | | Unit |
|---|---|---|---|---|---|
| | | | Min | Max | |
| $V_{OH}$ | High Level Output Voltage[2] | @ $V_{DDEXT}$=Min, $I_{OH}$=–2.0 mA[3] | 2.4 | | V |
| $V_{OL}$ | Low Level Output Voltage[2] | @ $V_{DDEXT}$=Min, $I_{OL}$=4.0 mA[3] | | 0.4 | V |
| $I_{IH}$ | High Level Input Current[4,5,6] | @ $V_{DDEXT}$=Max, $V_{IN}$=$V_{DD}$ Max | | 10 | µA |
| $I_{IL}$ | Low Level Input Current[4] | @ $V_{DDEXT}$=Max, $V_{IN}$=0 V | | 10 | µA |
| $I_{ILPU1}$ | Low Level Input Current Pull-Up1[5] | @ $V_{DDEXT}$=Max, $V_{IN}$=0 V | | 250 | µA |
| $I_{ILPU2}$ | Low Level Input Current Pull-Up2[6] | @ $V_{DDEXT}$=Max, $V_{IN}$=0 V | | 500 | µA |
| $I_{OZH}$ | Three-State Leakage Current[7,8,9,10] | @ $V_{DDEXT}$=Max, $V_{IN}$=$V_{DD}$ Max | | 10 | µA |
| $I_{OZL}$ | Three-State Leakage Current[7] | @ $V_{DDEXT}$=Max, $V_{IN}$=0 V | | 10 | µA |
| $I_{OZHPD}$ | Three-State Leakage Current Pull-Down[10] | @ $V_{DDEXT}$=Max, $V_{IN}$=$V_{DD}$ Max | | 250 | µA |
| $I_{OZLPU1}$ | Three-State Leakage Current Pull-Up1[8] | @ $V_{DDEXT}$=Max, $V_{IN}$=0 V | | 250 | µA |

# ELECTRICAL CHARACTERISTICS (CONTINUED)

| Parameter[1] | | Test Conditions | C and K Grades | | Unit |
|---|---|---|---|---|---|
| | | | Min | Max | |
| $I_{OZLPU2}$ | Three-State Leakage Current Pull-Up2[9] | @ $V_{DDEXT}$ = Max, $V_{IN}$ = 0 V | | 500 | µA |
| $I_{OZHA}$ | Three-State Leakage Current[11] | @ $V_{DDEXT}$ = Max, $V_{IN}$ = $V_{DD}$ Max | | 25 | µA |
| $I_{OZLA}$ | Three-State Leakage Current[11] | @ $V_{DDEXT}$ = Max, $V_{IN}$ = 0 V | | 4 | mA |
| $I_{DD-INPEAK}$ | Supply Current (Internal)[12] | $t_{CCLK}$ = 10.5 ns, $V_{DDINT}$ = Max | | 1400 | mA |
| $I_{DD-INHIGH}$ | Supply Current (Internal)[13] | $t_{CCLK}$ = 10.5 ns, $V_{DDINT}$ = Max | | 875 | mA |
| $I_{DD-INLOW}$ | Supply Current (Internal)[14] | $t_{CCLK}$ = 10.5 ns, $V_{DDINT}$ = Max | | 625 | mA |
| $I_{DD-IDLE}$ | Supply Current (Idle)[15] | $t_{CCLK}$ = 10.5 ns, $V_{DDINT}$ = Max | | 400 | mA |
| $AI_{DD}$ | Supply Current (Analog)[16] | @$AV_{DD}$ = Max | | 10 | mA |
| $C_{IN}$ | Input Capacitance[17,18] | $f_{IN}$ = 1 MHz, $T_{CASE}$ = 25°C, $V_{IN}$ = 2.5 V | | 4.7 | pF |

[1] Specifications subject to change without notice.

[2]Applies to output and bidirectional pins: DATA63–0, ADDR31–0, $\overline{MS3–0}$, $\overline{RDx}$, $\overline{WRx}$, PAGE, CLKOUT, ACK, FLAG3–0, TIMEXP, $\overline{HBG}$, REDY, $\overline{DMAG1}$, $\overline{DMAG2}$, $\overline{BR6–1}$, $\overline{PA}$, BRST, $\overline{CIF}$, DT0, DT1, TCLK0, TCLK1, RCLK0, RCLK1, TFS0, TFS1, RFS0, RFS1, LxDAT3–0, LxCLK, LxACK, $\overline{BMS}$, TDO, $\overline{EMU}$.

[3]See Output Drive Currents on page 45 for typical drive current capabilities.

[4]Applies to input pins: ACK, $\overline{SBTS}$, $\overline{IRQ2–0}$, $\overline{HBR}$, $\overline{CS}$, ID2–0, RPBA, EBOOT, LBOOT, CLKIN, $\overline{RESET}$, TCK, CLK_CFG3-0.

[5]Applies to input pins with internal pull-ups: DR0, DR1.

[6]Applies to input pins with internal pull-ups: $\overline{DMARx}$, TMS, TDI, $\overline{TRST}$.

[7]Applies to three-statable pins: DATA63–0, ADDR31–0, PAGE, CLKOUT, ACK, FLAG3–0, REDY, $\overline{HBG}$, $\overline{BMS}$, $\overline{BR6–1}$, TFSx, RFSx, TDO.

[8]Applies to three-statable pins with internal pull-ups: DTx, TCLKx, RCLKx, EMU.

[9]Applies to three-statable pins with internal pull-ups: $\overline{MS3–0}$, $\overline{RDx}$, $\overline{WRx}$, $\overline{DMAGx}$, $\overline{PA}$, $\overline{CIF}$.

[10]Applies to three-statable pins with internal pull-downs: LxDAT7–0, LxCLK, LxACK.

[11]Applies to ACK pulled up internally with 2 kΩ during reset or ID2–0 = 00x.

[12]The test program used to measure $I_{DD-INPEAK}$ represents worst case processor operation and is not sustainable under normal application conditions. Actual internal power measurements made using typical applications are less than specified. For more information, see Power Dissipation on page 45.

[13]$I_{DDINHIGH}$ is a composite average based on a range of high activity code. For more information, see Power Dissipation on page 45.

[14]$I_{DDINLOW}$ is a composite average based on a range of low activity code. For more information, see Power Dissipation on page 45.

[15]Idle denotes ADSP-21160N state during execution of IDLE instruction. For more information, see Power Dissipation on page 45.

[16]Characterized, but not tested.

[17]Applies to all signal pins.

[18]Guaranteed, but not tested.

**ABSOLUTE MAXIMUM RATINGS**

Internal (Core) Supply Voltage ($V_{DDINT}$)[1] –0.3 V to +2.3 V
Analog (PLL) Supply Voltage ($A_{VDD}$) .....–0.3 V to +2.3 V
External (I/O) Supply Voltage ($V_{DDEXT}$) ...–0.3 V to +4.6 V
Input Voltage.....–0.5 V to $V_{DDEXT}$+0.5 V
Output Voltage Swing .....–0.5 V to $V_{DDEXT}$+0.5 V
Load Capacitance.....200 pF
Junction Temperature under Bias.....130°C
Storage Temperature Range .....–65°C to +150°C
Lead Temperature (5 seconds) .....185°C

[1]Stresses greater than those listed above may cause permanent damage to the device. These are stress ratings only. Functional operation of the device at these or any other conditions greater than those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

**ESD SENSITIVITY**

**CAUTION:**

ESD (electrostatic discharge) sensitive device. Electrostatic charges as high as 4000V readily accumulate on the human body and test equipment and can discharge without detection. Although the ADSP-21160N 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 Specifications**

The ADSP-21160N's internal clock switches at higher frequencies than the system input clock (CLKIN). To generate the internal clock, the DSP uses an internal phase-locked loop (PLL). This PLL-based clocking minimizes the skew between the system clock (CLKIN) signal and the DSP's internal clock (the clock source for the external port logic and I/O pads).

The ADSP-21160N's internal clock (a multiple of CLKIN) provides the clock signal for timing internal memory, processor core, link ports, serial ports, and external port (as required for read/write strobes in asynchronous access mode). During reset, program the ratio between the DSP's internal clock frequency and external (CLKIN) clock frequency with the CLK_CFG3–0 pins. Even though the internal clock is the clock source for the external port, the external port clock always switches at the CLKIN frequency. To determine switching frequencies for the serial and link ports, divide down the internal clock, using the programmable divider control of each port (TDIVx/RDIVx for the serial ports and LxCLKD1–0 for the link ports).

Note the following definitions of various clock periods that are a function of CLKIN and the appropriate ratio control:

- $t_{CCLK} = (t_{CK}) / CR$
- $t_{LCLK} = (t_{CCLK}) \times LR$
- $t_{SCLK} = (t_{CCLK}) \times SR$

Where:

- LCLK = Link Port Clock
- SCLK = Serial Port Clock
- $t_{CK}$ = CLKIN Clock Period
- $t_{CCLK}$ = (Processor) Core Clock Period
- $t_{LCLK}$ = Link Port Clock Period
- $t_{SCLK}$ = Serial Port Clock Period
- CR = Core/CLKIN Ratio (2, 3, or 4:1, determined by CLK_CFG3–0 at reset)
- LR = Link Port/Core Clock Ratio (1, 2, 3, or 4:1, determined by LxCLKD)
- SR = Serial Port/Core Clock Ratio (wide range, determined by ×CLKDIV)

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, it is not meaningful to add parameters to derive longer times.

See Figure 33 under Test Conditions for voltage reference levels.

Switching Characteristics specify how the processor changes its signals. Circuitry external to the processor must be designed for compatibility with these signal characteristics. Switching characteristics describe what the processor will do in a given circumstance. 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.

*DSP Startup*

**Table 4. DSP Startup**

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| *Timing Requirements:* | | | | |
| $t_{RSTVDD}$ | $\overline{\text{RESET}}$ before $V_{DDINT}/V_{DDEXT}$ on | 0 | | ns |
| $t_{VDDRAMP}$ | $V_{DDINT}/V_{DDEXT}$ voltage ramp rate | 0.0018 | 9 | V/µs |
| $t_{VDDEVDD}$ | $V_{DDINT}$ on before $V_{DDEXT}$ | 0 | TBD | ms |
| $t_{CLKVDD}$ | CLKIN running before $V_{DDINT}/V_{DDEXT}$ | 5 | | ms |
| $t_{CLKRST}$ | CLKIN running before $\overline{\text{RESET}}$ de-asserted | 100 | | µs |
| $t_{PLLRST}$ | PLLCNTL[1] setup before $\overline{\text{RESET}}$ de-asserted | 20 | | µs |

[1]Applies to pins CLK_CFG3–0

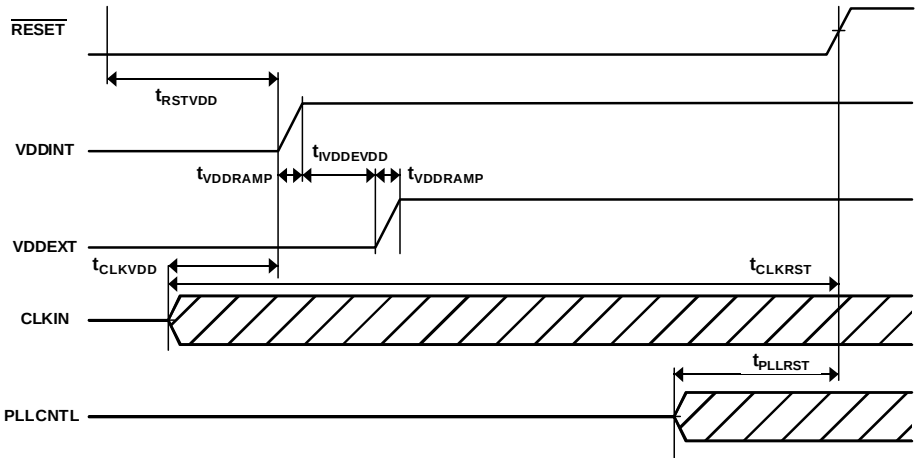



*Figure 10. DSP Startup*

***Clock Input***

**Table 5. Clock Input**

| Parameter | | 95 MHz | | Unit |
|---|---|---|---|---|
| | | Min | Max | |
| *Timing Requirements:* | | | | |
| $t_{CK}$ | CLKIN Period | 21 | 80 | ns |
| $t_{CKL}$ | CLKIN Width Low | 9.5 | 40 | ns |
| $t_{CKH}$ | CLKIN Width High | 9.5 | 40 | ns |
| $t_{CKRF}$ | CLKIN Rise/Fall (0.4 V–2.0 V) | | 3 | ns |

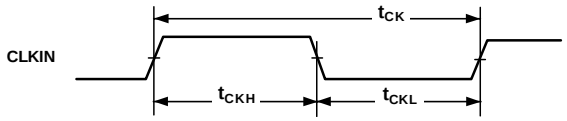



*Figure 11. Clock Input*

***Reset***

**Table 6. Reset**

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| *Timing Requirements:* | | | | |
| $t_{WRST}$ | $\overline{\text{RESET}}$ Pulsewidth Low[1] | $4t_{CK}$ | | ns |
| $t_{SRST}$ | $\overline{\text{RESET}}$ Setup Before CLKIN High[2] | 8 | | ns |

[1]Applies after the power-up sequence is complete. At power-up, the processor's internal phase-locked loop requires no more than 100 µs while $\overline{\text{RESET}}$ is low, assuming stable VDD and CLKIN (not including start-up time of external clock oscillator).
[2]Only required if multiple ADSP-21160Ns must come out of reset synchronous to CLKIN with program counters (PC) equal. Not required for multiple ADSP-21160Ns communicating over the shared bus (through the external port), because the bus arbitration logic automatically synchronizes itself after reset.

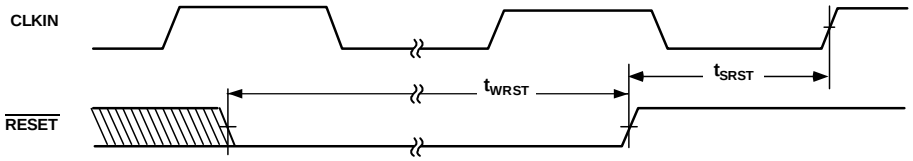



*Figure 12. Reset*

***Interrupts***

**Table 7. Interrupts**

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| *Timing Requirements:* | | | | |
| $t_{SIR}$ | $\overline{IRQ2–0}$ Setup Before CLKIN High[1] | 6 | | ns |
| $t_{HIR}$ | $\overline{IRQ2–0}$ Hold After CLKIN High[1] | 0 | | ns |
| $t_{IPW}$ | $\overline{IRQ2–0}$ Pulsewidth[2] | $2+t_{CK}$ | | ns |

[1]Only required for IRQx recognition in the following cycle.

[2]Applies only if $t_{SIR}$ and $t_{HIR}$ requirements are not met.

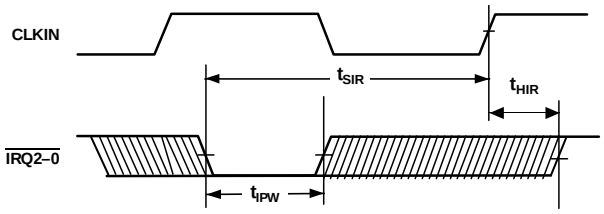



*Figure 13. Interrupts*

***Timer***

**Table 8. Timer**

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| *Switching Characteristic:* | | | | |
| $t_{DTEX}$ | CLKIN High to TIMEXP | 1 | 9 | ns |

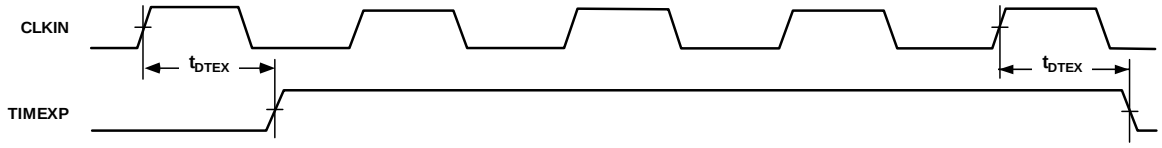



*Figure 14. Timer*

***Flags***

**Table 9. Flags**

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| *Timing Requirements:* | | | | |
| $t_{SFI}$ | FLAG3–0 IN Setup Before CLKIN High[1] | 4 | | ns |
| $t_{HFI}$ | FLAG3–0 IN Hold After CLKIN High[1] | 1 | | ns |
| $t_{DWRFI}$ | FLAG3–0 IN Delay After $\overline{RDx}/\overline{WRx}$ Low[1] | | 12 | ns |
| $t_{HFIWR}$ | FLAG3–0 IN Hold After $\overline{RDx}/\overline{WRx}$ Deasserted[1] | 0 | | ns |
| *Switching Characteristics:* | | | | |
| $t_{DFO}$ | FLAG3–0 OUT Delay After CLKIN High | | 9 | ns |
| $t_{HFO}$ | FLAG3–0 OUT Hold After CLKIN High | 1 | | ns |
| $t_{DFOE}$ | CLKIN High to FLAG3–0 OUT Enable | 1 | | ns |
| $t_{DFOD}$ | CLKIN High to FLAG3–0 OUT Disable | | $t_{CK}-t_{CCLK}+5$ | ns |

[1]Flag inputs meeting these setup and hold times for instruction cycle N will affect conditional instructions in instruction cycle N+2.

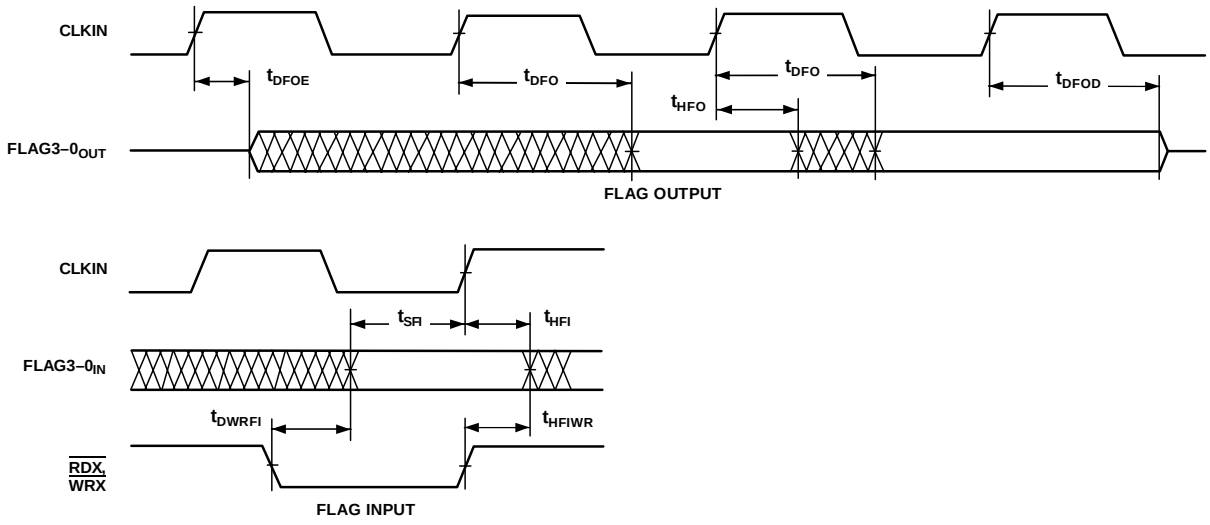



*Figure 15. Flags*

***Memory Read—Bus Master***

Use these specifications for asynchronous interfacing to memories (and memory-mapped peripherals) without reference to CLKIN. These specifications apply when the ADSP-21160N is the bus master accessing external memory space in asynchronous access mode. Note that timing for ACK, DATA, $\overline{RDx}$, $\overline{WRx}$, and $\overline{DMAG}$ strobe timing parameters only applies to asynchronous access mode.

**Table 10. Memory Read—Bus Master**

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| *Timing Requirements:* | | | | |
| $t_{DAD}$ | Address, $\overline{CIF}$, Selects Delay to Data Valid[1,2] | | $t_{CK} - 0.25t_{CCLK} - 11 + W$ | ns |
| $t_{DRLD}$ | $\overline{RDx}$ Low to Data Valid[1,3] | | $t_{CK} - 0.5t_{CCLK} + W$ | ns |
| $t_{HDA}$ | Data Hold from Address, Selects[4] | 0 | | ns |
| $t_{SDS}$ | Data Setup to $\overline{RDx}$ High[1] | 8 | | ns |
| $t_{HDRH}$ | Data Hold from $\overline{RDx}$ High[3,4] | 1 | | ns |
| $t_{DAAK}$ | ACK Delay from Address, Selects[2,5] | | $t_{CK} - 0.5t_{CCLK} - 12 + W$ | ns |
| $t_{DSAK}$ | ACK Delay from $\overline{RDx}$ Low[3,5] | | $t_{CK} - 0.75t_{CCLK} - 11 + W$ | ns |
| $t_{SAKC}$ | ACK Setup to CLKIN[3,5] | $0.5t_{CCLK} + 3$ | | ns |
| $t_{HAKC}$ | ACK Hold After CLKIN[3] | 1 | | ns |
| *Switching Characteristics:* | | | | |
| $t_{DRHA}$ | Address, $\overline{CIF}$, Selects Hold After $\overline{RDx}$ High[3] | $0.25t_{CCLK} - 1 + H$ | | ns |
| $t_{DARL}$ | Address, $\overline{CIF}$, Selects to $\overline{RDx}$ Low[2] | $0.25t_{CCLK} - 3$ | | ns |
| $t_{RW}$ | $\overline{RDx}$ Pulse width[3] | $t_{CK} - 0.5t_{CCLK} - 1 + W$ | | ns |
| $t_{RWR}$ | $\overline{RDx}$ High to $\overline{WRx}$, $\overline{RDx}$, $\overline{DMAGx}$ Low[3] | $0.5t_{CCLK} - 1 + HI$ | | ns |

W = (number of wait states specified in WAIT register) × $t_{CK}$.
HI = $t_{CK}$ (if an address hold cycle or bus idle cycle occurs, as specified in WAIT register; otherwise HI = 0).
H = $t_{CK}$ (if an address hold cycle occurs as specified in WAIT register; otherwise H = 0).

[1]Data Delay/Setup: User must meet $t_{DAD}$, $t_{DRLD}$, or $t_{SDS}$.

[2]The falling edge of $\overline{MSx}$, $\overline{BMS}$ is referenced.

[3]Note that timing for ACK, DATA, $\overline{RDx}$, $\overline{WRx}$, and $\overline{DMAG}$ strobe timing parameters only applies to asynchronous access mode.

[4]Data Hold: User must meet $t_{HDA}$ or $t_{HDRH}$ in asynchronous access mode. See Example System Hold Time Calculation on page 46 for the calculation of hold times given capacitive and dc loads.

[5]ACK Delay/Setup: User must meet $t_{DAAK}$, $t_{DSAK}$, or $t_{SAKC}$ for deassertion of ACK (Low), all three specifications must be met for assertion of ACK (High).

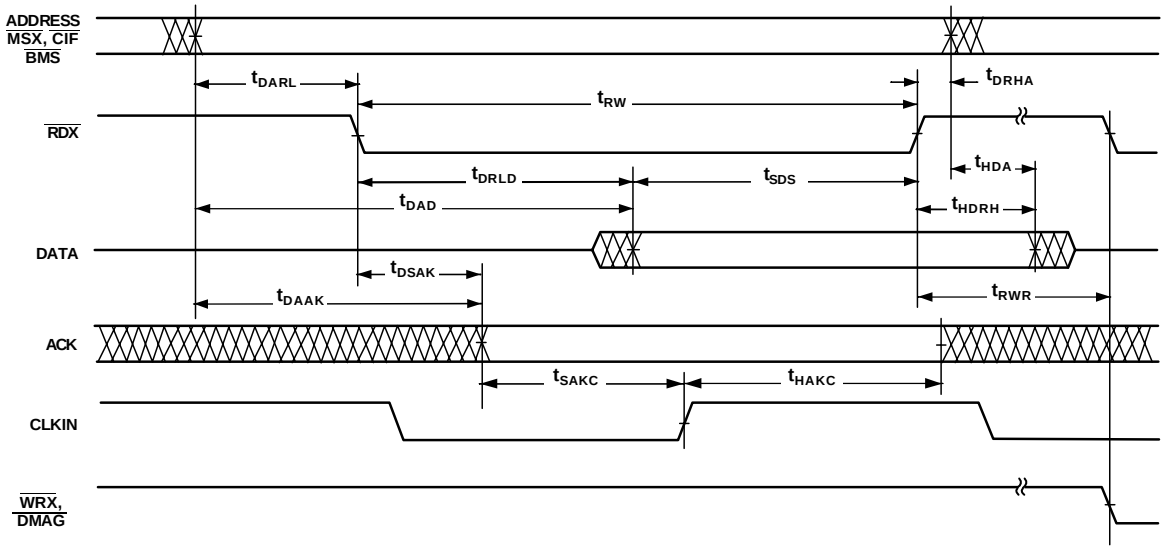



*Figure 16. Memory Read—Bus Master*

***Memory Write—Bus Master***

Use these specifications for asynchronous interfacing to memories (and memory-mapped peripherals) without reference to CLKIN. These specifications apply when the ADSP-21160N is the bus master accessing external memory space in asynchronous access mode. Note that timing for ACK, DATA, $\overline{RDx}$, $\overline{WRx}$, and $\overline{DMAG}$ strobe timing parameters only applies to asynchronous access mode.

**Table 11. Memory Write—Bus Master**

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| *Timing Requirements:* | | | | |
| $t_{DAAK}$ | ACK Delay from Address, Selects[1,2] | | $t_{CK}-0.5t_{CCLK}-12+W$ | ns |
| $t_{DSAK}$ | ACK Delay from $\overline{WRx}$ Low[1,3] | | $t_{CK}-0.75t_{CCLK}-11+W$ | ns |
| $t_{SAKC}$ | ACK Setup to CLKIN[1,3] | $0.5t_{CCLK}+3$ | | ns |
| $t_{HAKC}$ | ACK Hold After CLKIN[1,3] | 1 | | ns |
| *Switching Characteristics:* | | | | |
| $t_{DAWH}$ | Address, $\overline{CIF}$, Selects to $\overline{WRx}$ Deasserted[2,3] | $t_{CK}-0.25t_{CCLK}-3+W$ | | ns |
| $t_{DAWL}$ | Address, $\overline{CIF}$, Selects to $\overline{WRx}$ Low[2] | $0.25t_{CCLK}-3$ | | ns |
| $t_{WW}$ | $\overline{WRx}$ Pulse width[3] | $t_{CK}-0.5t_{CCLK}-1+W$ | | ns |
| $t_{DDWH}$ | Data Setup before $\overline{WRx}$ High[3] | $t_{CK}-0.5t_{CCLK}-1+W$ | | ns |
| $t_{DWHA}$ | Address Hold after $\overline{WRx}$ Deasserted[3] | $0.25t_{CCLK}-1+H$ | | ns |
| $t_{DWHD}$ | Data Hold after $\overline{WRx}$ Deasserted[3] | $0.25t_{CCLK}-1+H$ | | ns |
| $t_{DATRWH}$ | Data Disable after $\overline{WRx}$ Deasserted[3,4] | $0.25t_{CCLK}-2+H$ | $0.25t_{CCLK}+2+H$ | ns |
| $t_{WWR}$ | $\overline{WRx}$ High to $\overline{WRx}$, $\overline{RDx}$, $\overline{DMAGx}$ Low[3] | $0.5t_{CCLK}-1+HI$ | | ns |
| $t_{DDWR}$ | Data Disable before $\overline{WRx}$ or $\overline{RDx}$ Low | $0.25t_{CCLK}-1+I$ | | ns |
| $t_{WDE}$ | $\overline{WRx}$ Low to Data Enabled | $-0.25t_{CCLK}-1$ | | ns |

W = (number of wait states specified in WAIT register) × $t_{CK}$.
H = $t_{CK}$ (if an address hold cycle occurs, as specified in WAIT register; otherwise H = 0).
HI = $t_{CK}$ (if an address hold cycle or bus idle cycle occurs, as specified in WAIT register; otherwise HI = 0).
I = $t_{CK}$ (if a bus idle cycle occurs, as specified in WAIT register; otherwise I = 0).

[1]ACK Delay/Setup: User must meet $t_{DAAK}$ or $t_{DSAK}$ or $t_{SAKC}$ for deassertion of ACK (Low), all three specifications must be met for assertion of ACK (High).

[2]The falling edge of $\overline{MSx}$, $\overline{BMS}$ is referenced.

[3]Note that timing for ACK, DATA, $\overline{RDx}$, $\overline{WRx}$, and $\overline{DMAG}$ strobe timing parameters only applies to asynchronous access mode.

[4]See Example System Hold Time Calculation on page 46 for calculation of hold times given capacitive and dc loads.

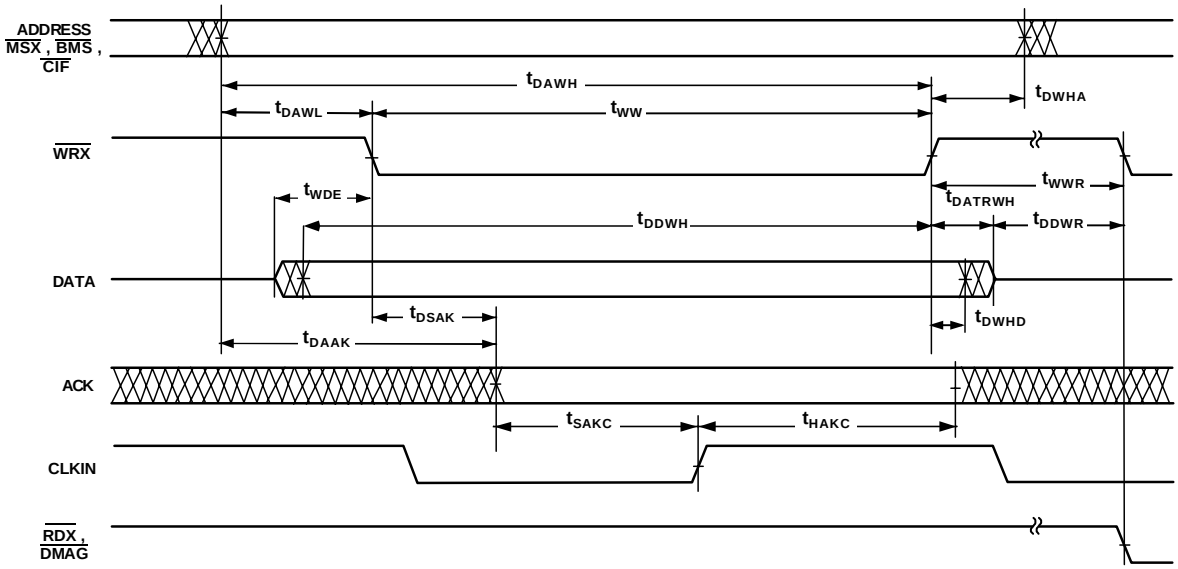



*Figure 17. Memory Write—Bus Master*

***Synchronous Read/Write—Bus Master***

Use these specifications for interfacing to external memory systems that require CLKIN—relative timing or for accessing a slave ADSP-21160N (in multiprocessor memory space). These synchronous switching characteristics are also valid during asynchronous memory reads and writes except where noted (see Memory Read—Bus Master on page 22 and Memory Write—Bus Master on page 24).

When accessing a slave ADSP-21160N, these switching characteristics must meet the slave's timing requirements for synchronous read/writes (see Synchronous Read/Write—Bus Slave on page 28). The slave ADSP-21160N must also meet these (bus master) timing requirements for data and acknowledge setup and hold times.

**Table 12. Synchronous Read/Write—Bus Master**

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| *Timing Requirements:* | | | | |
| $t_{SSDATI}$ | Data Setup Before CLKIN[1] | 5.5 | | ns |
| $t_{HSDATI}$ | Data Hold After CLKIN[1] | 1 | | ns |
| $t_{SACKC}$ | ACK Setup Before CLKIN[1] | $0.5t_{CCLK}+3$ | | ns |
| $t_{HACKC}$ | ACK Hold After CLKIN[1] | 1 | | ns |
| *Switching Characteristics:* | | | | |
| $t_{DADDO}$ | Address, $\overline{MS}$x, $\overline{BMS}$, BRST, $\overline{CIF}$ Delay After CLKIN | | 10 | ns |
| $t_{HADDO}$ | Address, $\overline{MS}$x, $\overline{BMS}$, BRST, $\overline{CIF}$ Hold After CLKIN | 1.5 | | ns |
| $t_{DPGO}$ | PAGE Delay After CLKIN | 1.5 | 11 | ns |
| $t_{DRDO}$ | $\overline{RDx}$ High Delay After CLKIN[1] | $0.25t_{CCLK}-1$ | $0.25t_{CCLK}+9$ | ns |
| $t_{DWRO}$ | $\overline{WRx}$ High Delay After CLKIN[1] | $0.25t_{CCLK}-1$ | $0.25t_{CCLK}+9$ | ns |
| $t_{DRWL}$ | $\overline{RDx}$/$\overline{WRx}$ Low Delay After CLKIN[1] | $0.25t_{CCLK}-1$ | $0.25t_{CCLK}+9$ | ns |
| $t_{DDATO}$ | Data Delay After CLKIN | | $0.25t_{CCLK}+9$ | ns |
| $t_{HDATO}$ | Data Hold After CLKIN | 1.5 | | ns |
| $t_{DACKMO}$ | ACK Delay After CLKIN[2] | 3 | 9 | ns |
| $t_{ACKMTR}$ | ACK Disable Before CLKIN[2] | −3 | | ns |
| $t_{DCKOO}$ | CLKOUT Delay After CLKIN | 2 | 5 | ns |
| $t_{CKOP}$ | CLKOUT Period | $t_{CK}-1$ | $t_{CK}$[3]$+1$ | ns |
| $t_{CKWH}$ | CLKOUT Width High | $t_{CK}/2-2$ | $t_{CK}/2+2$[3] | ns |
| $t_{CKWL}$ | CLKOUT Width Low | $t_{CK}/2-2$ | $t_{CK}/2+2$[3] | ns |

[1]Note that timing for ACK, DATA, $\overline{RDx}$, $\overline{WRx}$, and $\overline{DMAG}$ strobe timing parameters only applies to synchronous access mode.

[2]Applies to broadcast write, master precharge of ACK.

[3]Applies only when the DSP drives a bus operation; CLKOUT held inactive or three-state otherwise, For more information, see the System Design chapter in the ADSP-2116x *SHARC DSP Hardware Reference*.

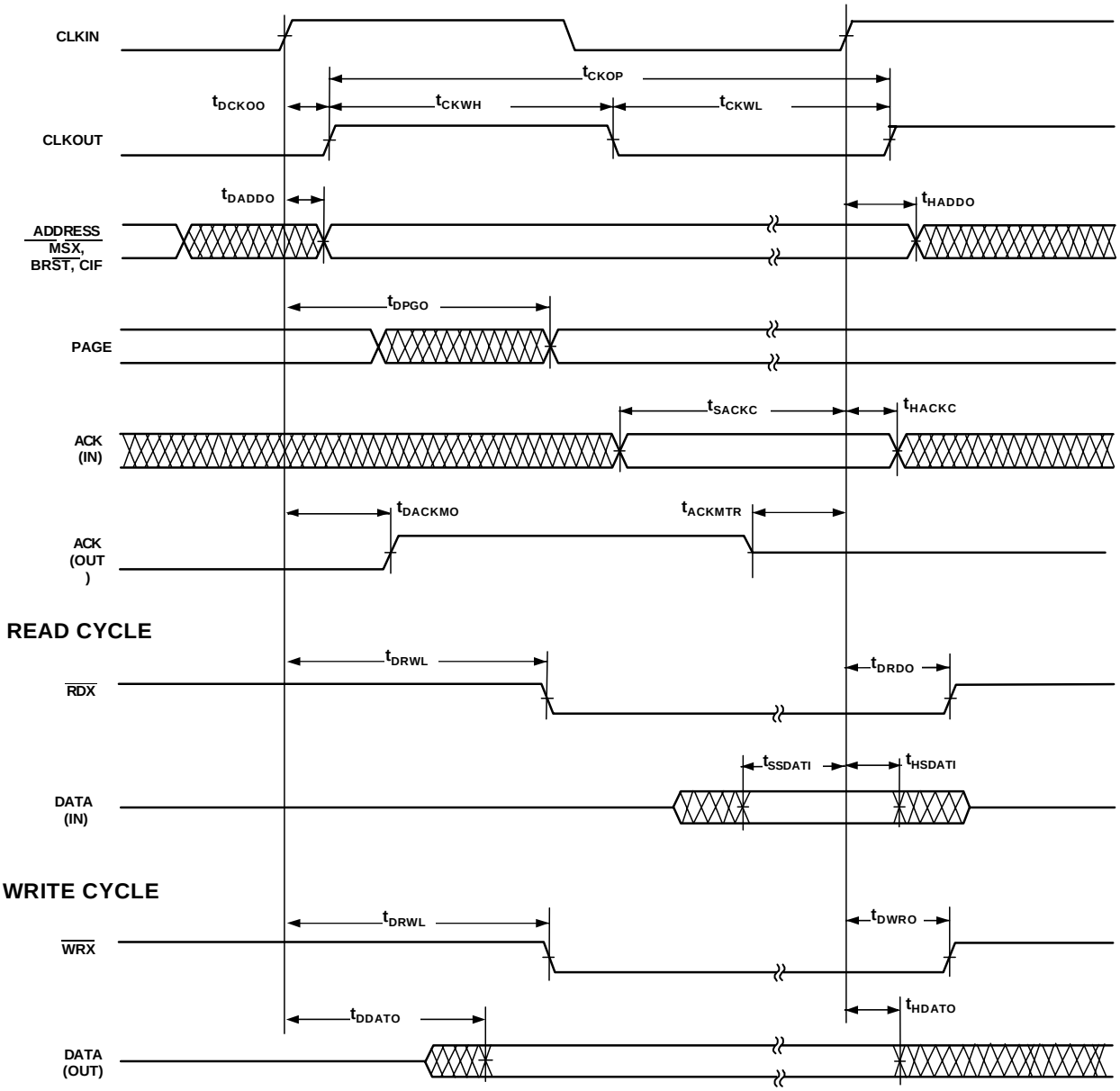



*Figure 18. Synchronous Read/Write—Bus Master*

***Synchronous Read/Write—Bus Slave***
Use these specifications for ADSP-21160N bus master accesses of a slave's IOP registers or internal memory (in multiprocessor memory space). The bus master must meet these (bus slave) timing requirements.

**Table 13. Synchronous Read/Write—Bus Slave**

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| *Timing Requirements:* | | | | |
| $t_{SADDI}$ | Address, BRST Setup Before CLKIN | 5 | | ns |
| $t_{HADDI}$ | Address, BRST Hold After CLKIN | 1 | | ns |
| $t_{SRWI}$ | $\overline{RDx}/\overline{WRx}$ Setup Before CLKIN | 5 | | ns |
| $t_{HRWI}$ | $\overline{RDx}/\overline{WRx}$ Hold After CLKIN | 1 | | ns |
| $t_{SSDATI}$ | Data Setup Before CLKIN | 5.5 | | ns |
| $t_{HSDATI}$ | Data Hold After CLKIN | 1 | | ns |
| *Switching Characteristics:* | | | | |
| $t_{DDATO}$ | Data Delay After CLKIN | | 12.5 | ns |
| $t_{HDATO}$ | Data Hold After CLKIN | 1.5 | | ns |
| $t_{DACKC}$ | ACK Delay After CLKIN | | 10 | ns |
| $t_{HACKO}$ | ACK Hold After CLKIN | 1.5 | | ns |

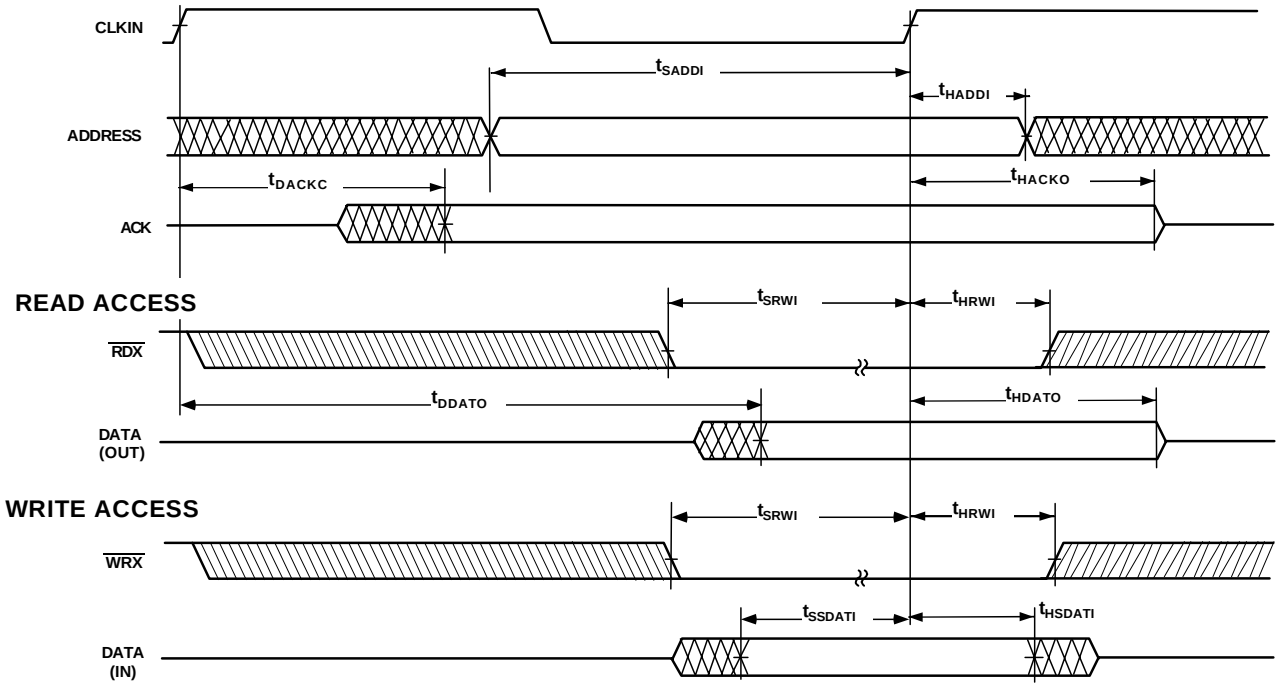



*Figure 19. Synchronous Read/Write—Bus Slave*

***Multiprocessor Bus Request and Host Bus Request***
Use these specifications for passing of bus mastership between multiprocessing ADSP-21160Ns ($\overline{BRx}$) or a host processor ($\overline{HBR}$, $\overline{HBG}$).

**Table 14. Multiprocessor Bus Request and Host Bus Request**

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| *Timing Requirements:* | | | | |
| $t_{HBGRCSV}$ | $\overline{HBG}$ Low to $\overline{RDx}$/$\overline{WRx}$/$\overline{CS}$ Valid | | 19 | ns |
| $t_{SHBRI}$ | $\overline{HBR}$ Setup Before CLKIN[1] | 6 | | ns |
| $t_{HHBRI}$ | $\overline{HBR}$ Hold After CLKIN[1] | 1 | | ns |
| $t_{SHBGI}$ | $\overline{HBG}$ Setup Before CLK/=']IN | 6 | | ns |
| $t_{HHBGI}$ | $\overline{HBG}$ Hold After CLKIN High | 1 | | ns |
| $t_{SBRI}$ | $\overline{BRx}$, $\overline{PA}$ Setup Before CLKIN | 9 | | ns |
| $t_{HBRI}$ | $\overline{BRx}$, $\overline{PA}$ Hold After CLKIN High | 1 | | ns |
| $t_{SPAI}$ | $\overline{PA}$ Setup Before CLKIN | 9 | | ns |
| $t_{HPAI}$ | $\overline{PA}$ Hold After CLKIN High | 1 | | ns |
| $t_{SRPBAI}$ | RPBA Setup Before CLKIN | 6 | | ns |
| $t_{HRPBAI}$ | RPBA Hold After CLKIN | 2 | | ns |
| *Switching Characteristics:* | | | | |
| $t_{DHBGO}$ | $\overline{HBG}$ Delay After CLKIN | | 7 | ns |
| $t_{HHBGO}$ | $\overline{HBG}$ Hold After CLKIN | 2 | | ns |
| $t_{DBRO}$ | $\overline{BRx}$ Delay After CLKIN | | 8 | ns |
| $t_{HBRO}$ | $\overline{BRx}$ Hold After CLKIN | 1.5 | | ns |
| $t_{DPASO}$ | $\overline{PA}$ Delay After CLKIN, Slave | | 8 | ns |
| $t_{TRPAS}$ | $\overline{PA}$ Disable After CLKIN, Slave | 1.5 | | ns |
| $t_{DPAMO}$ | $\overline{PA}$ Delay After CLKIN, Master | | $0.25t_{CCLK}+9$ | ns |
| $t_{PATR}$ | $\overline{PA}$ Disable Before CLKIN, Master | $0.25t_{CCLK}-5$ | | ns |
| $t_{DRDYCS}$ | REDY (O/D) or (A/D) Low from $\overline{CS}$ and $\overline{HBR}$ Low[2] | | $0.5t_{CK}$ | ns |
| $t_{TRDYHG}$ | REDY (O/D) Disable or REDY (A/D) High from $\overline{HBG}$[2] | $t_{CK}+25$ | | ns |
| $t_{ARDYTR}$ | REDY (A/D) Disable from $\overline{CS}$ or $\overline{HBR}$ High[2] | | 11 | ns |

[1]Only required for recognition in the current cycle.

[2](O/D) = open drain, (A/D) = active drive.

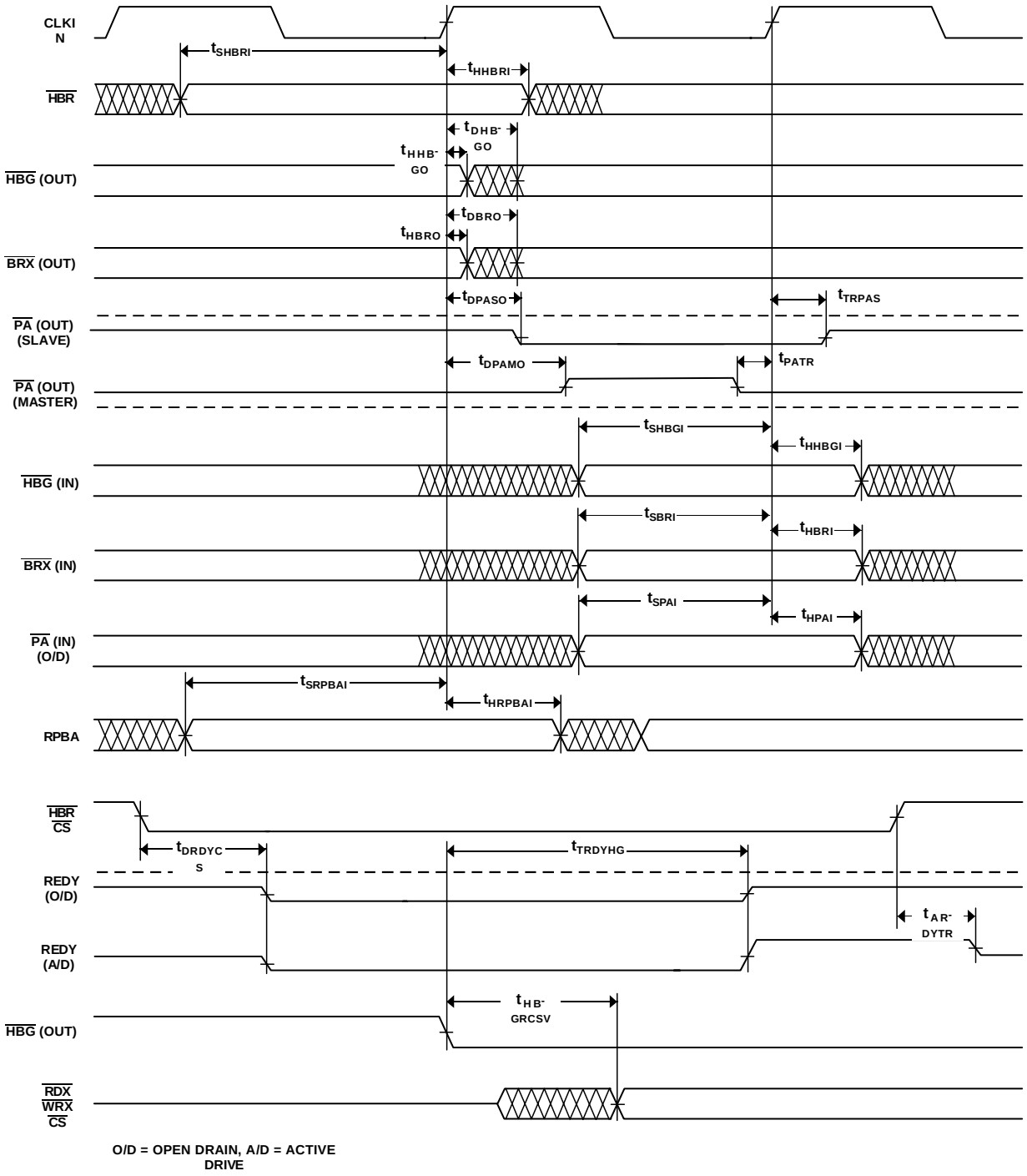



*Figure 20. Multiprocessor Bus Request and Host Bus Request*

***Asynchronous Read/Write—Host to ADSP-21160N***
Use these specifications (Table 15 and Table 16) for asynchronous host processor accesses of an ADSP-21160N, after the host has asserted $\overline{CS}$ and $\overline{HBR}$ (low). After $\overline{HBG}$ is returned by the ADSP-21160N, the host can drive the $\overline{RDx}$ and $\overline{WRx}$ pins to access the ADSP-21160N's internal memory or IOP registers. $\overline{HBR}$ and $\overline{HBG}$ are assumed low for this timing

**Table 15. Read Cycle**

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| *Timing Requirements:* | | | | |
| $t_{SADRDL}$ | Address Setup/$\overline{CS}$ Low Before $\overline{RDx}$ Low | 0 | | ns |
| $t_{HADRDH}$ | Address Hold/$\overline{CS}$ Hold Low After $\overline{RDx}$ | 2 | | ns |
| $t_{WRWH}$ | $\overline{RDx}$/$\overline{WRx}$ High Width | 5 | | ns |
| $t_{DRDHRDY}$ | $\overline{RDx}$ High Delay After REDY (O/D) Disable | 0 | | ns |
| $t_{DRDHRDY}$ | $\overline{RDx}$ High Delay After REDY (A/D) Disable | 0 | | ns |
| *Switching Characteristics:* | | | | |
| $t_{SDATRDY}$ | Data Valid Before REDY Disable from Low | 2 | | ns |
| $t_{DRDYRDL}$ | REDY (O/D) or (A/D) Low Delay After $\overline{RDx}$ Low | | 10 | ns |
| $t_{RDYPRD}$ | REDY (O/D) or (A/D) Low Pulsewidth for Read | $t_{CK}$ | | ns |
| $t_{HDARWH}$ | Data Disable After $\overline{RDx}$ High | 2 | 6 | ns |

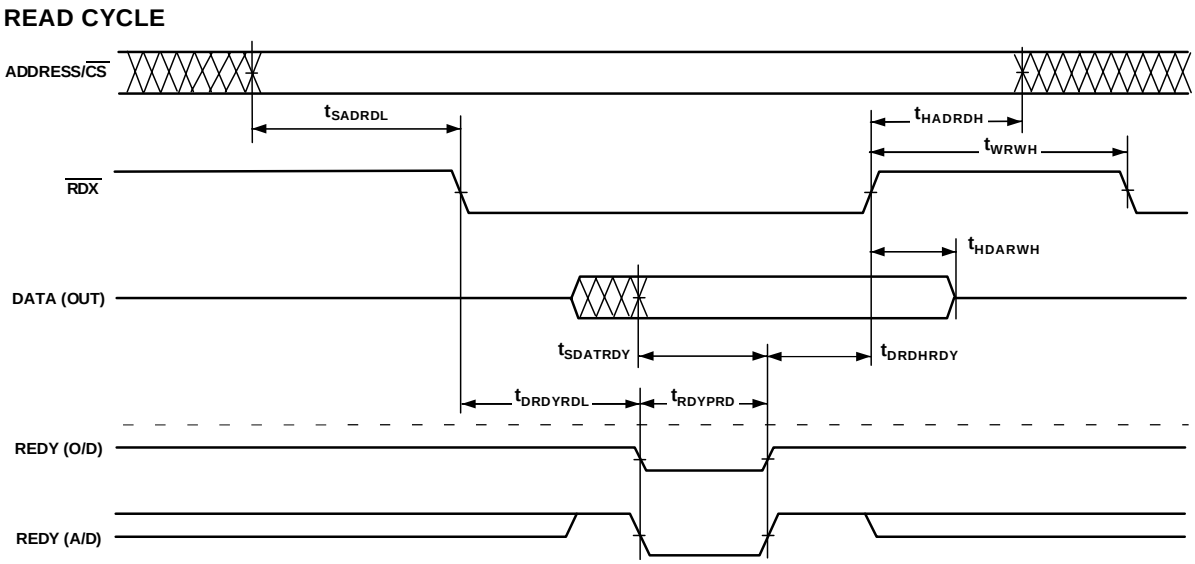



*Figure 21. Read Cycle (Asynchronous Read—Host to ADSP-21160N)*

**Table 16. Write Cycle**

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| *Timing Requirements:* | | | | |
| $t_{SCSWRL}$ | $\overline{CS}$ Low Setup Before $\overline{WRx}$ Low | 0 | | ns |
| $t_{HCSWRH}$ | $\overline{CS}$ Low Hold After $\overline{WRx}$ High | 0 | | ns |
| $t_{SADWRH}$ | Address Setup Before $\overline{WRx}$ High | 6 | | ns |
| $t_{HADWRH}$ | Address Hold After $\overline{WRx}$ High | 2 | | ns |
| $t_{WWRL}$ | $\overline{WRx}$ Low Width | 7 | | ns |
| $t_{WRWH}$ | $\overline{RDx}$/$\overline{WRx}$ High Width | 5 | | ns |
| $t_{DWRHRDY}$ | $\overline{WRx}$ High Delay After REDY (O/D) or (A/D) Disable | 0 | | ns |
| $t_{SDATWH}$ | Data Setup Before $\overline{WRx}$ High | 5 | | ns |
| $t_{HDATWH}$ | Data Hold After $\overline{WRx}$ High | 4 | | ns |
| *Switching Characteristics:* | | | | |
| $t_{DRDYWRL}$ | REDY (O/D) or (A/D) Low Delay After $\overline{WRx}$/$\overline{CS}$ Low | | 11 | ns |
| $t_{RDYPWR}$ | REDY (O/D) or (A/D) Low Pulsewidth for Write | 12 | | ns |

**WRITE CYCLE**

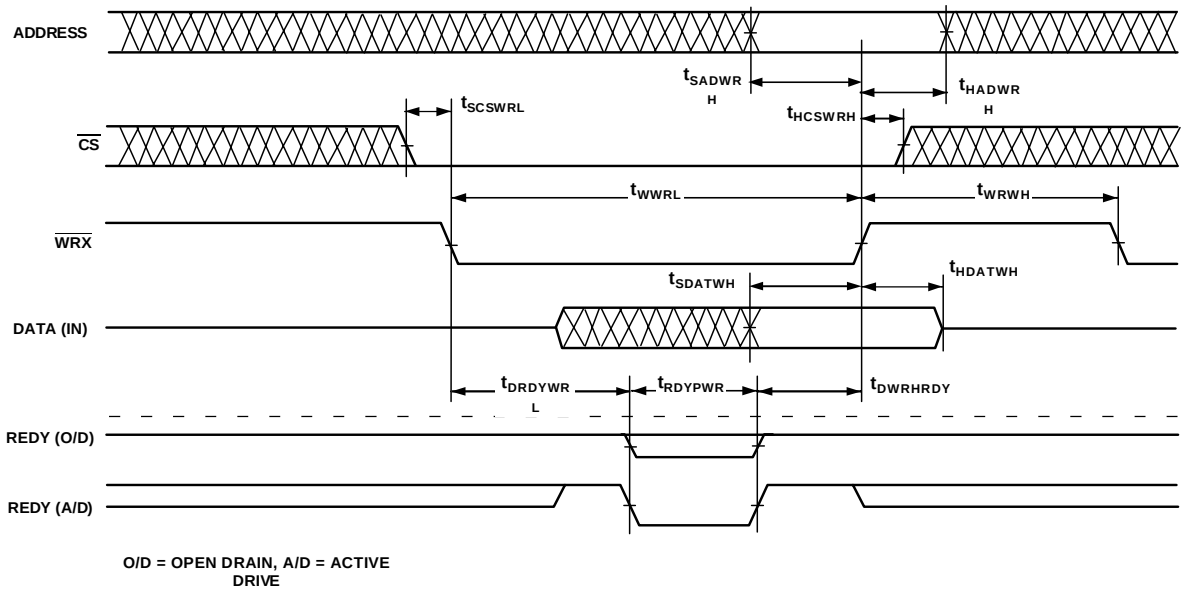



*Figure 22. Write Cycle (Asynchronous Write—Host to ADSP-21160N)*

***Three-State Timing—Bus Master and Bus Slave***
These specifications show how the memory interface is disabled (stops driving) or enabled (resumes driving) relative to CLKIN and the $\overline{\text{SBTS}}$ pin. This timing is applicable to bus master transition cycles (BTC) and host transition cycles (HTC) as well as the $\overline{\text{SBTS}}$ pin.

**Table 17. Three-State Timing—Bus Slave, $\overline{\text{HBR}}$, $\overline{\text{SBTS}}$**

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| *Timing Requirements:* | | | | |
| $t_{STSCK}$ | $\overline{\text{SBTS}}$ Setup Before CLKIN | 6 | | ns |
| $t_{HTSCK}$ | $\overline{\text{SBTS}}$ Hold After CLKIN | 1 | | ns |
| *Switching Characteristics:* | | | | |
| $t_{MIENA}$ | Address/Select Enable After CLKIN | 1.5 | 9 | ns |
| $t_{MIENS}$ | Strobes Enable After CLKIN[1] | 1.5 | 9 | ns |
| $t_{MIENHG}$ | $\overline{\text{HBG}}$ Enable After CLKIN | 1.5 | 9 | ns |
| $t_{MITRA}$ | Address/Select Disable After CLKIN | $0.25t_{CCLK}-1$ | $0.25t_{CCLK}+4$ | ns |
| $t_{MITRS}$ | Strobes Disable After CLKIN[1] | $0.25t_{CCLK}-4$ | $0.25t_{CCLK}$ | ns |
| $t_{MITRHG}$ | $\overline{\text{HBG}}$ Disable After CLKIN | 3.5 | 8 | ns |
| $t_{DATEN}$ | Data Enable After CLKIN[2] | 1.5 | 10 | ns |
| $t_{DATTR}$ | Data Disable After CLKIN[2] | 1.5 | 5 | ns |
| $t_{ACKEN}$ | ACK Enable After CLKIN[2] | 1.5 | 9 | ns |
| $t_{ACKTR}$ | ACK Disable After CLKIN[2] | 1.5 | 5 | ns |
| $t_{CDCEN}$ | CLKOUT Enable After CLKIN | 1.5 | 9 | ns |
| $t_{CDCTR}$ | CLKOUT Disable After CLKIN | $t_{CCLK}-3$ | $t_{CCLK}+1$ | ns |
| $t_{MTRHBG}$ | Memory Interface Disable Before $\overline{\text{HBG}}$ Low[3] | $t_{CK}-6$ | $t_{CK}+2$ | ns |
| $t_{MENHBG}$ | Memory Interface Enable After $\overline{\text{HBG}}$ High[3] | $t_{CK}-5$ | $t_{CK}+5$ | ns |

[1]Strobes = $\overline{\text{RDx}}$, $\overline{\text{WRx}}$, $\overline{\text{DMAG}}$x.

[2]In addition to bus master transition cycles, these specs also apply to bus master and bus slave synchronous read/write.

[3]Memory Interface = Address, $\overline{\text{RDx}}$, $\overline{\text{WRx}}$, $\overline{\text{MSx}}$, PAGE, $\overline{\text{DMAG}}$x, and $\overline{\text{BMS}}$ (in EPROM boot mode).

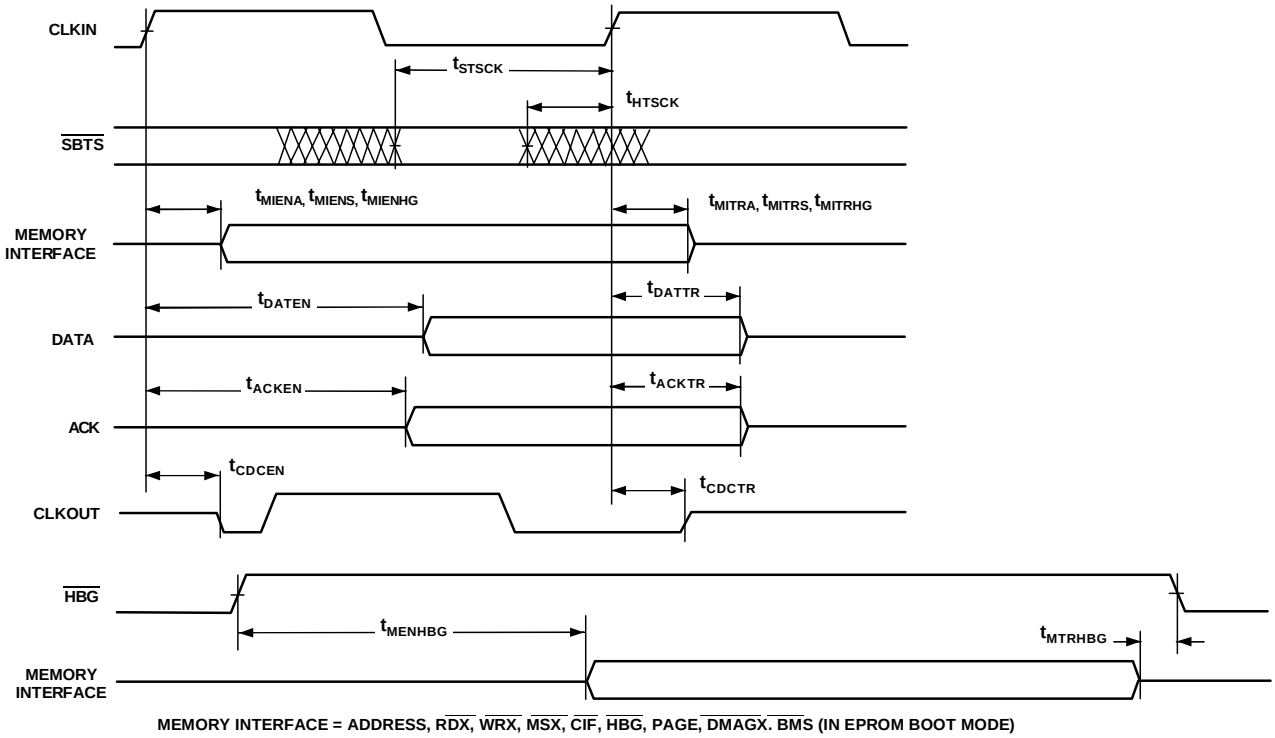



*Figure 23. Three-State Timing—Bus Slave, HBR, SBTS*

***DMA Handshake***

These specifications describe the three DMA handshake modes. In all three modes $\overline{\text{DMAR}}$ is used to initiate transfers. For handshake mode, $\overline{\text{DMAG}}$ controls the latching or enabling of data externally. For external handshake mode, the data transfer is controlled by the ADDR31–0, $\overline{\text{RDx}}$, $\overline{\text{WRx}}$, PAGE, $\overline{\text{MS3–0}}$, ACK, and $\overline{\text{DMAG}}$ signals. For Paced Master mode, the data transfer is controlled by ADDR31–0, $\overline{\text{RDx}}$, $\overline{\text{WRx}}$, $\overline{\text{MS3–0}}$, and ACK (not $\overline{\text{DMAG}}$). For Paced Master mode, the Memory Read-Bus Master, Memory Write-Bus Master, and Synchronous Read/Write-Bus Master timing specifications for ADDR31–0, $\overline{\text{RDx}}$, $\overline{\text{WRx}}$, $\overline{\text{MS3–0}}$, PAGE, DATA63–0, and ACK also apply.

**Table 18. DMA Handshake**

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| *Timing Requirements:* | | | | |
| $t_{SDRC}$ | $\overline{\text{DMARx}}$ Setup Before CLKIN[1] | 3 | | ns |
| $t_{WDR}$ | $\overline{\text{DMARx}}$ Width Low (Nonsynchronous)[2] | $0.5t_{CCLK}+1$ | | ns |
| $t_{SDATDGL}$ | Data Setup After $\overline{\text{DMAGx}}$ Low[3] | | $t_{CK}-0.5t_{CCLK}-7$ | ns |
| $t_{HDATIDG}$ | Data Hold After $\overline{\text{DMAGx}}$ High | 2 | | ns |
| $t_{DATDRH}$ | Data Valid After $\overline{\text{DMARx}}$ High[3] | | $t_{CK}+3$ | ns |
| $t_{DMARLL}$ | $\overline{\text{DMARx}}$ Low Edge to Low Edge[4] | $t_{CK}$ | | ns |
| $t_{DMARH}$ | $\overline{\text{DMARx}}$ Width High[2] | $0.5t_{CCLK}+1$ | | ns |
| *Switching Characteristics:* | | | | |
| $t_{DDGL}$ | $\overline{\text{DMAGx}}$ Low Delay After CLKIN | $0.25t_{CCLK}+1$ | $0.25t_{CCLK}+9$ | ns |
| $t_{WDGH}$ | $\overline{\text{DMAGx}}$ High Width | $0.5t_{CCLK}-1+HI$ | | ns |
| $t_{WDGL}$ | $\overline{\text{DMAGx}}$ Low Width | $t_{CK}-0.5t_{CCLK}-1$ | | ns |
| $t_{HDGC}$ | $\overline{\text{DMAGx}}$ High Delay After CLKIN | $t_{CK}-0.25t_{CCLK}+1.5$ | $t_{CK}-0.25t_{CCLK}+9$ | ns |
| $t_{VDATDGH}$ | Data Valid Before $\overline{\text{DMAGx}}$ High[5] | $t_{CK}-0.25t_{CCLK}-8$ | $t_{CK}-0.25t_{CCLK}+5$ | ns |
| $t_{DATRDGH}$ | Data Disable After $\overline{\text{DMAGx}}$ High[6] | $0.25t_{CCLK}-3$ | $0.25t_{CCLK}+1.5$ | ns |
| $t_{DGWRL}$ | $\overline{\text{WRx}}$ Low Before $\overline{\text{DMAGx}}$ Low | –1.5 | 2 | ns |
| $t_{DGWRH}$ | $\overline{\text{DMAGx}}$ Low Before $\overline{\text{WRx}}$ High | $t_{CK}-0.5t_{CCLK}-2+W$ | | ns |
| $t_{DGWRR}$ | $\overline{\text{WRx}}$ High Before $\overline{\text{DMAGx}}$ High[7] | –1.5 | 2 | ns |
| $t_{DGRDL}$ | $\overline{\text{RDx}}$ Low Before $\overline{\text{DMAGx}}$ Low | –1.5 | 2 | ns |
| $t_{DRDGH}$ | $\overline{\text{RDx}}$ Low Before $\overline{\text{DMAGx}}$ High | $t_{CK}-0.5t_{CCLK}-2+W$ | | ns |
| $t_{DGRDR}$ | $\overline{\text{RDx}}$ High Before $\overline{\text{DMAGx}}$ High[7] | –1.5 | 2 | ns |
| $t_{DGWR}$ | $\overline{\text{DMAGx}}$ High to $\overline{\text{WRx}}$, $\overline{\text{RDx}}$, $\overline{\text{DMAGx}}$ Low | $0.5t_{CCLK}-2+HI$ | | ns |
| $t_{DADGH}$ | Address/Select Valid to $\overline{\text{DMAGx}}$ High | 18 | | ns |
| $t_{DDGHA}$ | Address/Select Hold after $\overline{\text{DMAGx}}$ High | 1 | | ns |

W = (number of wait states specified in WAIT register) × $t_{CK}$.
HI = $t_{CK}$ (if data bus idle cycle occurs, as specified in WAIT register; otherwise HI = 0).

[1]Only required for recognition in the current cycle.

[2]Maximum throughput using $\overline{\text{DMARx/DMAGx}}$ handshaking equals $t_{WDR} + t_{DMARH} = (0.5t_{CCLK}+1) + (0.5t_{CCLK}+1) = 12.5$ ns (80 MHz). This throughput limit applies to non-synchronous access mode only.

[3]$t_{SDATDGL}$ is the data setup requirement if $\overline{\text{DMARx}}$ is not being used to hold off completion of a write. Otherwise, if $\overline{\text{DMARx}}$ low holds off completion of the write, the data can be driven $t_{DATDRH}$ after $\overline{\text{DMARx}}$ is brought high.

[4]Use $t_{DMARLL}$ if $\overline{\text{DMARx}}$ transitions synchronous with CLKIN. Otherwise, use $t_{WDR}$ and $t_{DMARH}$.

[5]$t_{VDATDGH}$ is valid if $\overline{\text{DMARx}}$ is not being used to hold off completion of a read. If $\overline{\text{DMARx}}$ is used to prolong the read, then $t_{VDATDGH} = t_{CK} - .25t_{CCLK} - 8 + (n \times t_{CK})$ where n equals the number of extra cycles that the access is prolonged.

[6]See Example System Hold Time Calculation on page 46 for calculation of hold times given capacitive and dc loads.

[7]This parameter applies for synchronous access mode only.

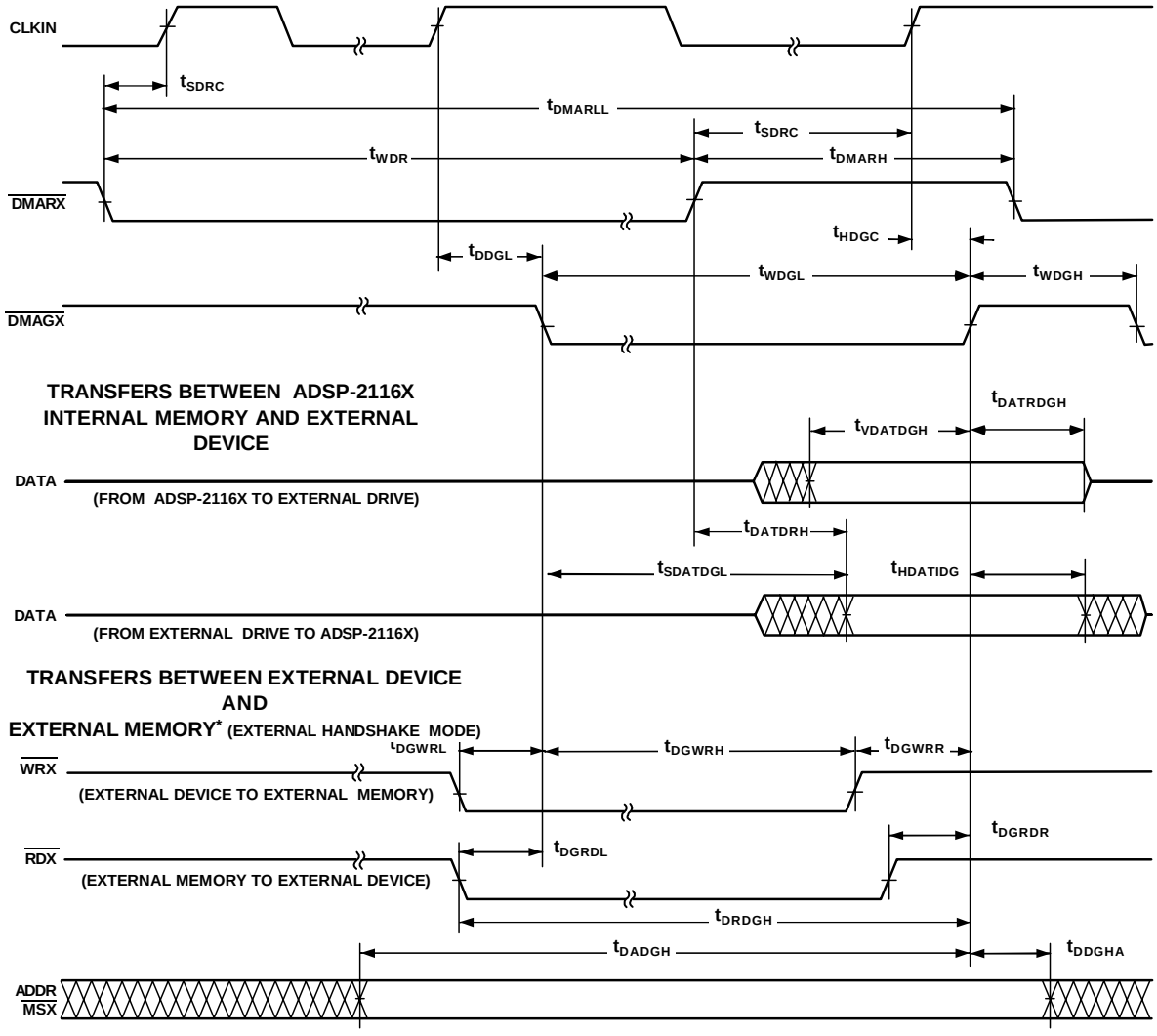



*Figure 24. DMA Handshake Timing*

***Link Ports***

Calculation of link receiver data setup and hold, relative to link clock, is required to determine the maximin allowable skew that can be introduced in the transmission path, between LDATA and LCLK. Setup skew is the maximum delay that can be introduced in LDATA, relative to LCLK (setup skew = $t_{LCLKTWH}$ minimum – $t_{DLDCH}$ – $t_{SLDCL}$). Hold skew is the maximum delay that can be introduced in LCLK, relative to LDATA (hold skew = $t_{LCLKTWL}$ minimum + $t_{HLDCH}$ – $t_{HLDCL}$).Calculations made directly from speed specifications result in unrealistically small skew times, because they include multiple tester guardbands.

Note that there is a two-cycle effect latency between the link port enable instruction and the DSP enabling the link port.

**Table 19. Link Ports—Receive**

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| *Timing Requirements:* | | | | |
| $t_{SLDCL}$ | Data Setup Before LCLK Low | 2.5 | | ns |
| $t_{HLDCL}$ | Data Hold After LCLK Low | 2.5 | | ns |
| $t_{LCLKIW}$ | LCLK Period | $t_{LCLK}$ | | ns |
| $t_{LCLKRWL}$ | LCLK Width Low | 4 | | ns |
| $t_{LCLKRWH}$ | LCLK Width High | 4 | | ns |
| *Switching Characteristics:* | | | | |
| $t_{DLALC}$ | LACK Low Delay After LCLK High[1] | 12 | 17 | ns |

[1]LACK goes low with $t_{DLALC}$ relative to rise of LCLK after first nibble, but doesn't go low if the receiver's link buffer is not about to fill.

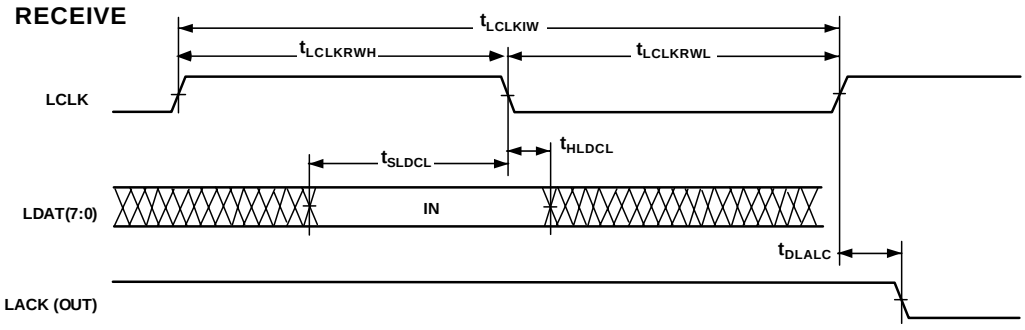



*Figure 25. Link Ports—Receive*

**Table 20. Link Ports—Transmit**

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| *Timing Requirements:* | | | | |
| $t_{SLACH}$ | LACK Setup Before LCLK High | 14 | | ns |
| $t_{HLACH}$ | LACK Hold After LCLK High | –2 | | ns |
| *Switching Characteristics:* | | | | |
| $t_{DLDCH}$ | Data Delay After LCLK High | | 6.0 | ns |
| $t_{HLDCH}$ | Data Hold After LCLK High | –2 | | ns |
| $t_{LCLKTWL}$ | LCLK Width Low | $0.5t_{LCLK}-.5$ | $0.5t_{LCLK}+.5$ | ns |
| $t_{LCLKTWH}$ | LCLK Width High | $0.5t_{LCLK}-.5$ | $0.5t_{LCLK}+.5$ | ns |
| $t_{DLACLK}$ | LCLK Low Delay After LACK High | $0.5t_{LCLK}+5$ | $^3/_2t_{LCLK}+11$ | ns |

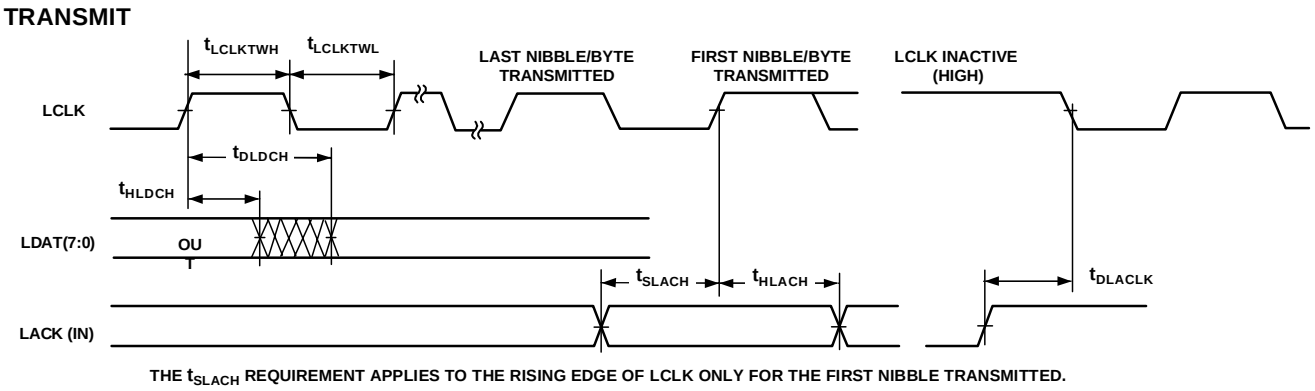



*Figure 26. Link Ports—Transmit*

***Serial Ports***

To determine whether communication is possible between two devices at clock speed n, the following specifications must be confirmed: 1) frame sync delay and frame sync setup and hold, 2) data delay and data setup and hold, and 3) SCLK width.

**Table 21. Serial Ports—External Clock**

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| *Timing Requirements:* | | | | |
| $t_{SFSE}$ | TFS/RFS Setup Before TCLK/RCLK[1] | 3.5 | | ns |
| $t_{HFSE}$ | TFS/RFS Hold After TCLK/RCLK[1,2] | 4 | | ns |
| $t_{SDRE}$ | Receive Data Setup Before RCLK[1] | 1.5 | | ns |
| $t_{HDRE}$ | Receive Data Hold After RCLK[1] | 4 | | ns |
| $t_{SCLKW}$ | TCLK/RCLK Width | 8 | | ns |
| $t_{SCLK}$ | TCLK/RCLK Period | $2t_{CCLK}$ | | ns |

[1]Referenced to sample edge.

[2]RFS hold after RCK when MCE = 1, MFD = 0 is 0 ns minimum from drive edge. TFS hold after TCK for late external TFS is 0 ns minimum from drive edge.

**Table 22. Serial Ports—Internal Clock**

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| *Timing Requirements:* | | | | |
| $t_{SFSI}$ | TFS Setup Before TCLK[1]; RFS Setup Before RCLK[1] | 8 | | ns |
| $t_{HFSI}$ | TFS/RFS Hold After TCLK/RCLK[1,2] | 1 | | ns |
| $t_{SDRI}$ | Receive Data Setup Before RCLK[1] | 6.5 | | ns |
| $t_{HDRI}$ | Receive Data Hold After RCLK[1] | 3 | | ns |

[1]Referenced to sample edge.

[2]RFS hold after RCK when MCE = 1, MFD = 0 is 0 ns minimum from drive edge. TFS hold after TCK for late external TFS is 0 ns minimum from drive edge.

**Table 23. Serial Ports—External or Internal Clock**

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| *Switching Characteristics:* | | | | |
| $t_{DFSE}$ | RFS Delay After RCLK (Internally Generated RFS)[1] | | 13 | ns |
| $t_{HOFSE}$ | RFS Hold After RCLK (Internally Generated RFS)[1] | 3 | | ns |

[1]Referenced to drive edge.

**Table 24. Serial Ports—External Clock**

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| *Switching Characteristics:* | | | | |
| $t_{DFSE}$ | TFS Delay After TCLK (Internally Generated TFS)[1] | | 13 | ns |
| $t_{HOFSE}$ | TFS Hold After TCLK (Internally Generated TFS)[1] | 3 | | ns |
| $t_{DDTE}$ | Transmit Data Delay After TCLK[1] | | 16 | ns |
| $t_{HDTE}$ | Transmit Data Hold After TCLK[1] | 0 | | ns |

[1]Referenced to drive edge.

**Table 25. Serial Ports—Internal Clock**

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| *Switching Characteristics:* | | | | |
| $t_{DFSI}$ | TFS Delay After TCLK (Internally Generated TFS)[1] | | 4.5 | ns |
| $t_{HOFSI}$ | TFS Hold After TCLK (Internally Generated TFS)[1] | –1.5 | | ns |
| $t_{DDTI}$ | Transmit Data Delay After TCLK[1] | | 7.5 | ns |
| $t_{HDTI}$ | Transmit Data Hold After TCLK[1] | 0 | | ns |
| $t_{SCLKIW}$ | TCLK/RCLK Width | $0.5t_{SCLK}-1.5$ | $0.5t_{SCLK}+1.5$ | ns |

[1]Referenced to drive edge.

**Table 26. Serial Ports—Enable and Three-State**

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| *Switching Characteristics:* | | | | |
| $t_{DDTEN}$ | Data Enable from External TCLK[1] | 4 | | ns |
| $t_{DDTTE}$ | Data Disable from External TCLK[1] | | 10 | ns |
| $t_{DDTIN}$ | Data Enable from Internal TCLK[1] | 0 | | ns |
| $t_{DDTTI}$ | Data Disable from Internal TCLK[1] | | 3 | ns |

[1]Referenced to drive edge.

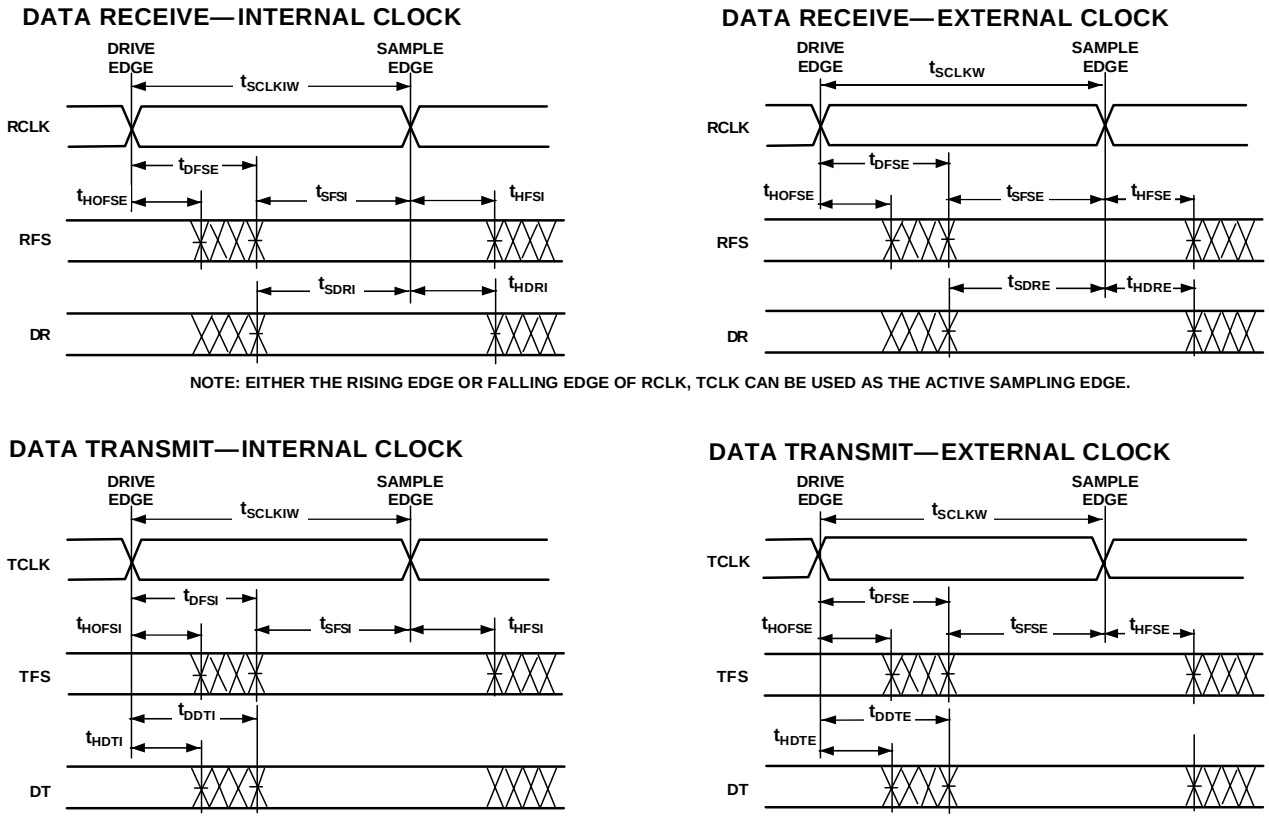



*Figure 27. Serial Ports*

**Table 27. Serial Ports—External Late Frame Sync**

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| *Switching Characteristics:* | | | | |
| $t_{DDTLFSE}$ | Data Delay from Late External TFS or External RFS with MCE = 1, MFD = 0[1] | | 13 | ns |
| $t_{DDTENFS}$ | Data Enable from late FS or MCE = 1, MFD = 0[1] | 1.0 | | ns |

[1]MCE = 1, TFS enable and TFS valid follow $t_{DDTLFSE}$ and $t_{DDTENFS}$.

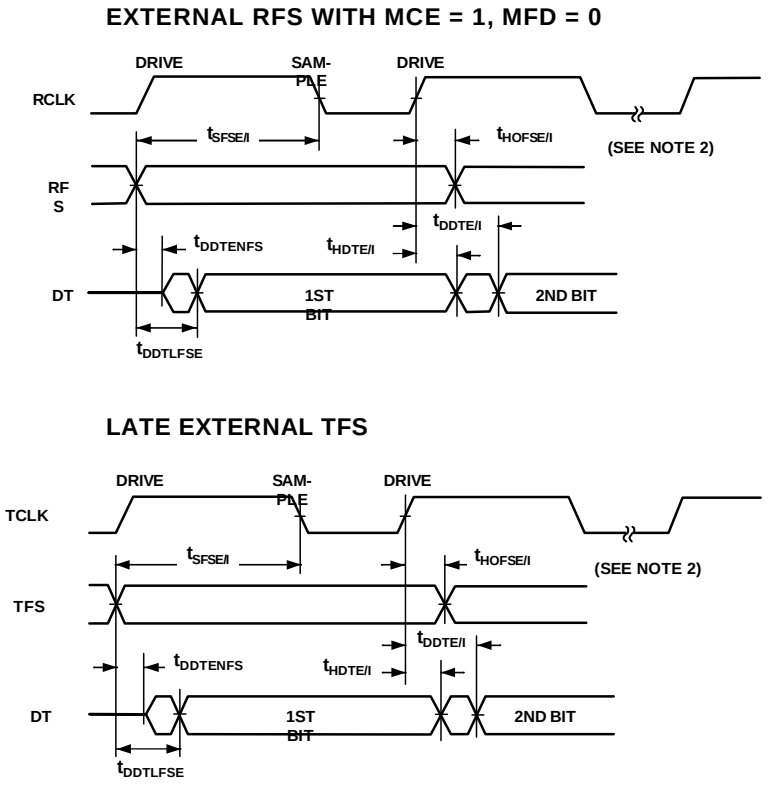



*Figure 28. External Late Frame Sync*

*JTAG Test Access Port and Emulation*

**Table 28. JTAG Test Access Port and Emulation**

| Parameter | | Min | Max | Unit |
|---|---|---|---|---|
| *Timing Requirements:* | | | | |
| $t_{TCK}$ | TCK Period | $t_{CK}$ | | ns |
| $t_{STAP}$ | TDI, TMS Setup Before TCK High | 5 | | ns |
| $t_{HTAP}$ | TDI, TMS Hold After TCK High | 6 | | ns |
| $t_{SSYS}$ | System Inputs Setup Before TCK Low[1] | 7 | | ns |
| $t_{HSYS}$ | System Inputs Hold After TCK Low[1] | 18 | | ns |
| $t_{TRSTW}$ | $\overline{\text{TRST}}$ Pulsewidth | $4t_{CK}$ | | ns |
| *Switching Characteristics:* | | | | |
| $t_{DTDO}$ | TDO Delay from TCK Low | | 13 | ns |
| $t_{DSYS}$ | System Outputs Delay After TCK Low[2] | | 30 | ns |

[1]System Inputs = DATA63–0, ADDR31–0, $\overline{\text{RDx}}$, $\overline{\text{WRx}}$, ACK, $\overline{\text{SBTS}}$, $\overline{\text{HBR}}$, $\overline{\text{HBG}}$, $\overline{\text{CS}}$, $\overline{\text{DMAR1}}$, $\overline{\text{DMAR2}}$, $\overline{\text{BR6–1}}$, ID2–0, RPBA, $\overline{\text{IRQ2–0}}$, FLAG3–0, $\overline{\text{PA}}$, BRST, DR0, DR1, TCLK0, TCLK1, RCLK0, RCLK1, TFS0, TFS1, RFS0, RFS1, LxDAT7–0, LxCLK, LxACK, EBOOT, LBOOT, $\overline{\text{BMS}}$, CLKIN, $\overline{\text{RESET}}$.

[2]System Outputs = DATA63–0, ADDR31–0, $\overline{\text{MS3–0}}$, $\overline{\text{RDx}}$, $\overline{\text{WRx}}$, ACK, PAGE, CLKOUT, $\overline{\text{HBG}}$, REDY, $\overline{\text{DMAG1}}$, $\overline{\text{DMAG2}}$, $\overline{\text{BR6–1}}$, $\overline{\text{PA}}$, BRST, $\overline{\text{CIF}}$, FLAG3–0, TIMEXP, DT0, DT1, TCLK0, TCLK1, RCLK0, RCLK1, TFS0, TFS1, RFS0, RFS1, LxDAT7–0, LxCLK, LxACK, $\overline{\text{BMS}}$.

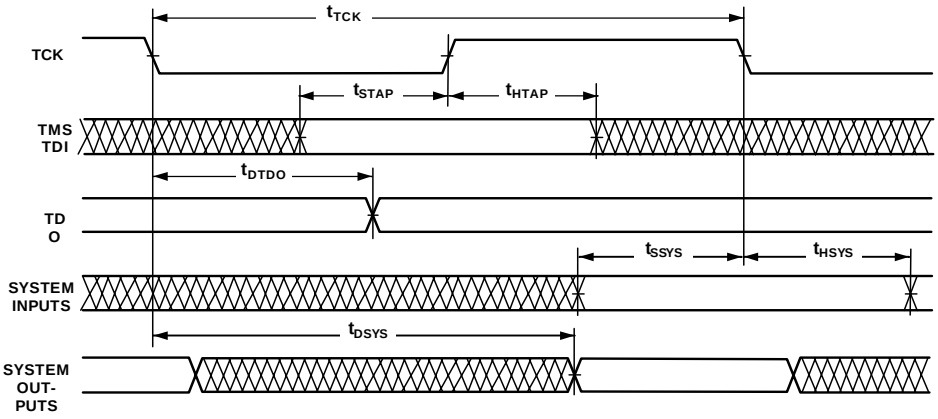



*Figure 29. IEEE 11499.1 JTAG Test Access Port*

### Output Drive Currents

Figure 30 shows typical I–V characteristics for the output drivers of the ADSP-21160N. The curves represent the current drive capability of the output drivers as a function of output voltage.

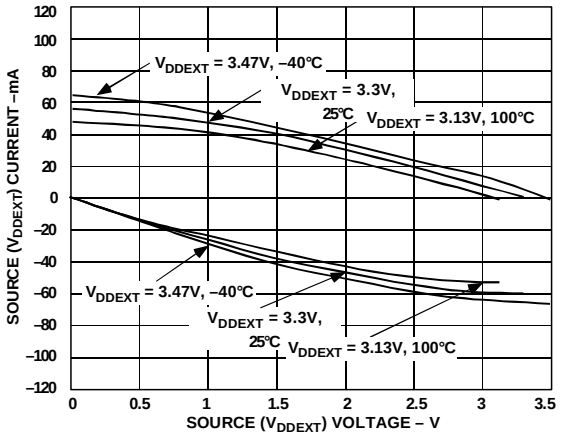



*Figure 30. ADSP-21160N Typical Drive Currents*

### Power Dissipation

Total power dissipation has two components, one due to internal circuitry and one due to the switching of external output drivers.

Internal power dissipation is dependent on the instruction execution sequence and the data operands involved. Using the current specifications ($I_{DDINPEAK}$, $I_{DDINHIGH}$, $I_{DDINLOW}$, $I_{DDIDLE}$) from Electrical Characteristics on page 14 and the current-versus-operation information in Table 29, engineers can estimate the ADSP-21160N's internal power supply ($V_{DDINT}$) input current for a specific application, according to the following formula:

$$\frac{\%\,\text{Peak} \times I_{DDINPEAK} + \%\,\text{High} \times I_{DDINHIGH} + \%\,\text{Low} \times I_{DDINLOW} + \%\,\text{Idle} \times I_{DDIDLE}}{I_{DDINT}}$$

The external component of total power dissipation is caused by the switching of output pins. Its magnitude depends on:

- the number of output pins that switch during each cycle (O)
- the maximum frequency at which they can switch (f)
- their load capacitance (C)
- their voltage swing (VDD)

and is calculated by:

$$P_{EXT} = O \times C \times V_{DD}^2 \times f$$

The load capacitance should include the processor's package capacitance ($C_{IN}$). The switching frequency includes driving the load high and then back low. Address and data pins can drive high and low at a maximum rate of $1/(2\,t_{CK})$. The write strobe can switch every cycle at a frequency of $1/t_{CK}$. Select pins switch at $1/(2\,t_{CK})$, but selects can switch on each cycle.

**Table 29. ADSP-21160N Operation Types vs. Input Current**

| Operation | Peak Activity[1] | High Activity[1] | Low Activity[1] |
|---|---|---|---|
| Instruction Type | Multifunction | Multifunction | Single Function |
| Instruction Fetch | Cache | Internal Memory | Internal Memory |
| Core Memory Access[2] | 2 per $t_{CK}$ cycle (DM×64 and PM×64) | 1 per $t_{CK}$ cycle (DM×64) | None |
| Internal Memory DMA | 1 per 2 $t_{CCLK}$ cycles | 1 per 2 $t_{CCLK}$ cycles | None |
| External Memory DMA | 1 per external port cycle (×64) | 1 per external port cycle (×64) | None |
| Data bit pattern for core memory access and DMA | Worst case | Random | N/A |

[1]Peak Activity=$I_{DDINPEAK}$, High Activity=$I_{DDINHIGH}$, and Low Activity=$I_{DDINLOW}$. The state of the PEYEN bit (SIMD versus SISD mode) does not influence these calculations.

[2]These assume a 2:1 core clock ratio. For more information on ratios and clocks ($t_{CK}$ and $t_{CCLK}$), see the timing ratio definitions on page 17.

**Example:** Estimate $P_{EXT}$ with the following assumptions:

- A system with one bank of external data memory—asynchronous RAM (64-bit)
- Four 64K × 16 RAM chips are used, each with a load of 10 pF
- External data memory writes occur every other cycle, a rate of 1/(2 $t_{CK}$), with 50% of the pins switching
- The bus cycle time is 47.5 MHz ($t_{CK}$ = 21 ns).

The $P_{EXT}$ equation is calculated for each class of pins that can drive:

**Table 30. External Power Calculations (3.3 V Device)**

| Pin Type | # of Pins | % Switching | × C | × f | × VDD$^2$ | = $P_{EXT}$ |
|---|---|---|---|---|---|---|
| Address | 15 | 50 | × 44.7 pF | × 24 MHz | × 10.9 V | = 0.088 W |
| MS0 | 1 | 0 | × 44.7 pF | × 24 MHz | × 10.9 V | = 0.000 W |
| $\overline{WRx}$ | 2 | – | × 44.7 pF | × 24 MHz | × 10.9 V | = 0.023 W |
| Data | 64 | 50 | × 14.7 pF | × 24 MHz | × 10.9 V | = 0.123 W |
| CLKOUT | 1 | – | × 4.7 pF | × 48 MHz | × 10.9 V | = 0.003 W |
| | | | | | | $P_{EXT}$ = 0.237 W |

A typical power consumption can now be calculated for these conditions by adding a typical internal power dissipation:

$$P_{TOTAL} = P_{EXT} + P_{INT} + P_{PLL}$$

Where:

- $P_{EXT}$ is from Table 30
- $P_{INT}$ is $I_{DDINT}$ × 1.9 V, using the calculation $I_{DDINT}$ listed in Power Dissipation on page 45
- $P_{PLL}$ is $AI_{DD}$ × 1.9 V, using the value for $AI_{DD}$ listed in ABSOLUTE MAXIMUM RATINGS on page 16

Note that the conditions causing a worst-case $P_{EXT}$ are different from those causing a worst-case $P_{INT}$. Maximum $P_{INT}$ cannot occur while 100% of the output pins are switching from all ones to all zeros. Note also that it is not common for an application to have 100% or even 50% of the outputs switching simultaneously.

**Test Conditions**

The test conditions for timing parameters appearing in ADSP-21160N specifications on page 14 include output disable time, output enable time, and capacitive loading.

***Output Disable Time***

Output pins are considered to be disabled when they stop driving, go into a high impedance state, and start to decay from their output high or low voltage. The time for the voltage on the bus to decay by ΔV is dependent on the capacitive load, $C_L$ and the load current, $I_L$. This decay time can be approximated by the following equation:

$$t_{DECAY} = (C_L \Delta V)/I_L$$

The output disable time $t_{DIS}$ is the difference between $t_{MEASURED}$ and $t_{DECAY}$ as shown in Figure 31. The time $t_{MEASURED}$ is the interval from when the reference signal switches to when the output voltage decays ΔV from the measured output high or output low voltage. $t_{DECAY}$ is calculated with test loads $C_L$ and $I_L$, and with ΔV equal to 0.5 V.

***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 (Figure 31). If multiple pins (such as the data bus) are enabled, the measurement value is that of the first pin to start driving.

***Example System Hold Time Calculation***

To determine the data output hold time in a particular system, first calculate $t_{DECAY}$ using the equation given above. Choose –V to be the difference between the ADSP-21160N's output voltage and the input threshold for the device requiring the hold time. A typical –V will be 0.4 V. $C_L$ is the total bus capacitance (per data line), and $I_L$ is the total leakage or three-state current (per data line). The hold time will be $t_{DECAY}$ plus the minimum disable time (i.e., $t_{DATRWH}$ for the write cycle).

***Capacitive Loading***

Output delays and holds are based on standard capacitive loads: 50 pF on all pins (see Figure 32). The delay and hold specifications given should be derated by a factor of 1.5 ns/50 pF for loads other than the nominal value of 50 pF. Figure 34 and Figure 35 show how output rise time varies with capacitance. Figure 36 graphically shows how output delays and holds vary with load capacitance. (Note that this graph or derating does not apply to output disable

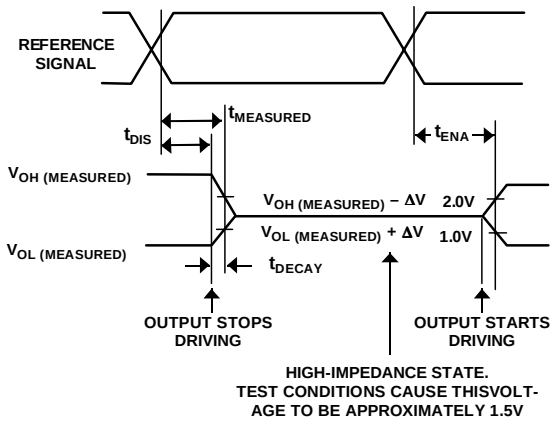



*Figure 31. Output Enable/Disable*

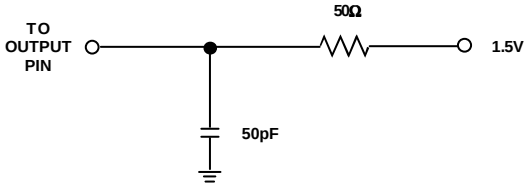



*Figure 32. Equivalent Device Loading for AC Measurements (Includes All Fixtures)*


*Figure 33. Voltage Reference Levels for AC Measurements (Except Output Enable/Disable)*

delays; see Output Disable Time on page 46.) The graphs of Figure 34, Figure 35, and Figure 36 may not be linear outside the ranges shown.

**Environmental Conditions**

The ADSP-21160NKB-95 and ADSP-21160NCB-TBD are provided in a 400-Ball Metric PBGA (Plastic Ball Grid Array) package.

***Thermal Characteristics***

The ADSP-21160N is specified for a case temperature ($T_{CASE}$). To ensure that the $T_{CASE}$ data sheet specification is not exceeded, a heatsink and/or an air flow source may be used. Use the center block of ground pins (PBGA balls: H8–13, J8–13, K8–13, L8–13, M8–13, and N8–13) to provide thermal pathways to the printed circuit board's

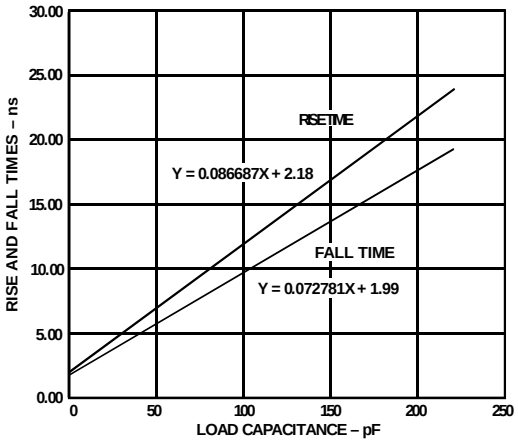



*Figure 34. Typical Output Rise Time (10%–90%, $V_{DDEXT}$ = Max) vs. Load Capacitance*

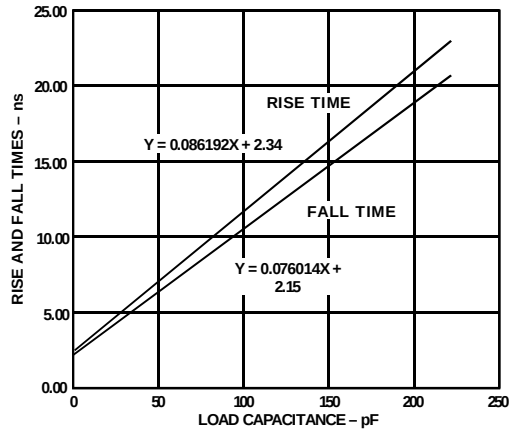



*Figure 35. Typical Output Rise Time (10%–90%, $V_{DDEXT}$ = Min) vs. Load Capacitance*

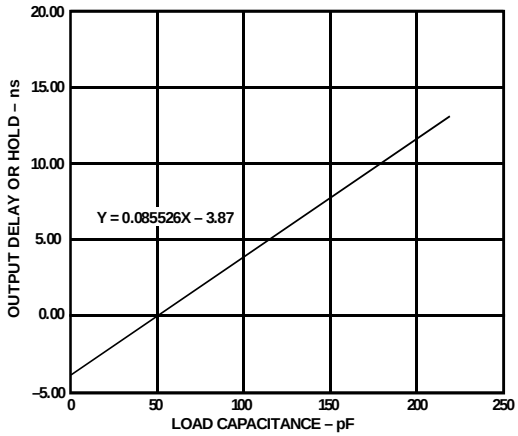



*Figure 36. Typical Output Delay or Hold vs. Load Capacitance (at Max Case Temperature)*

ground plane. A heatsink should be attached to the ground plane (as close as possible to the thermal pathways) with a thermal adhesive.

$$T_{CASE} = T_{AMB} + (PD \times \theta_{CA})$$

- $T_{CASE}$ = Case temperature (measured on top surface of package)
- *PD* = Power dissipation in W (this value depends upon the specific application; a method for calculating PD is shown under Power Dissipation).
- $\theta_{CA}$ = Value from Table 31.
- $\theta_{JB}$= 6.46°C/W

**Table 31. Airflow Over Package Versus $\theta_{CA}$**

| Airflow (Linear Ft./Min.) | 0 | 200 | 400 |
|---|---|---|---|
| $\theta_{CA}$ (°C/W)[1] | 12.13 | 9.86 | 8.7 |

[1]$\theta_{JC}$ = 3.6 °C/W.

### 400-BALL METRIC PBGA PIN CONFIGURATIONS

Table 32 lists the pin assignments for the PBGA package, and the pin configurations diagram on page 55 shows the pin assignment summary.

**Table 32. 400-ball Metric PBGA Pin Assignments**

| Pin Name | PBGA Pin# | Pin Name | PBGA Pin# | Pin Name | PBGA Pin# | Pin Name | PBGA Pin# |
|---|---|---|---|---|---|---|---|
| DATA[14] | A01 | DATA[22] | B01 | DATA[24] | C01 | DATA[28] | D01 |
| DATA[13] | A02 | DATA[16] | B02 | DATA[18] | C02 | DATA[25] | D02 |
| DATA[10] | A03 | DATA[15] | B03 | DATA[17] | C03 | DATA[20] | D03 |
| DATA[8] | A04 | DATA[9] | B04 | DATA[11] | C04 | DATA[19] | D04 |
| DATA[4] | A05 | DATA[6] | B05 | DATA[7] | C05 | DATA[12] | D05 |
| DATA[2] | A06 | DATA[3] | B06 | DATA[5] | C06 | $V_{DDEXT}$ | D06 |
| TDI | A07 | DATA[0] | B07 | DATA[1] | C07 | $V_{DDINT}$ | D07 |
| $\overline{TRST}$ | A08 | TCK | B08 | TMS | C08 | $V_{DDEXT}$ | D08 |
| $\overline{RESET}$ | A09 | $\overline{EMU}$ | B09 | TD0 | C09 | $V_{DDEXT}$ | D09 |
| RPBA | A10 | $\overline{IRQ2}$ | B10 | $\overline{IRQ1}$ | C10 | $V_{DDEXT}$ | D10 |
| $\overline{IRQ0}$ | A11 | FLAG3 | B11 | FLAG2 | C11 | $V_{DDEXT}$ | D11 |
| FLAG1 | A12 | FLAG0 | B12 | $V_{DDEXT}$ | C12 | $V_{DDEXT}$ | D12 |
| TIMEXP | A13 | $V_{DDEXT}$ | B13 | NC | C13 | $V_{DDINT}$ | D13 |
| $V_{DDEXT}$ | A14 | NC | B14 | TCLK1 | C14 | $V_{DDEXT}$ | D14 |
| NC | A15 | DT1 | B15 | DR1 | C15 | TFS0 | D15 |
| TFS1 | A16 | RCLK1 | B16 | DR0 | C16 | L1DAT[7] | D16 |
| RFS1 | A17 | RFS0 | B17 | L0DAT[7] | C17 | L0CLK | D17 |
| RCLK0 | A18 | TCLK0 | B18 | L0DAT[6] | C18 | L0DAT[3] | D18 |
| DT0 | A19 | L0DAT[5] | B19 | L0ACK | C19 | L0DAT[1] | D19 |
| L0DAT[4] | A20 | L0DAT[2] | B20 | L0DAT[0] | C20 | L1CLK | D20 |

**Table 32. 400-ball Metric PBGA Pin Assignments (Continued)**

| Pin Name | PBGA Pin# | Pin Name | PBGA Pin# | Pin Name | PBGA Pin# | Pin Name | PBGA Pin# |
|---|---|---|---|---|---|---|---|
| DATA[30] | E01 | DATA[34] | F01 | DATA[38] | G01 | DATA[40] | H01 |
| DATA[29] | E02 | DATA[33] | F02 | DATA[35] | G02 | DATA[39] | H02 |
| DATA[23] | E03 | DATA[27] | F03 | DATA[32] | G03 | DATA[37] | H03 |
| DATA[21] | E04 | DATA[26] | F04 | DATA[31] | G04 | DATA[36] | H04 |
| $V_{DDEXT}$ | E05 | $V_{DDEXT}$ | F05 | $V_{DDEXT}$ | G05 | $V_{DDEXT}$ | H05 |
| $V_{DDINT}$ | E06 | $V_{DDINT}$ | F06 | $V_{DDINT}$ | G06 | $V_{DDINT}$ | H06 |
| $V_{DDINT}$ | E07 | GND | F07 | GND | G07 | GND | H07 |
| $V_{DDINT}$ | E08 | GND | F08 | GND | G08 | GND | H08 |
| $V_{DDINT}$ | E09 | GND | F09 | GND | G09 | GND | H09 |
| $V_{DDINT}$ | E10 | GND | F10 | GND | G10 | GND | H10 |
| GND | E11 | GND | F11 | GND | G11 | GND | H11 |
| $V_{DDINT}$ | E12 | GND | F12 | GND | G12 | GND | H12 |
| $V_{DDINT}$ | E13 | GND | F13 | GND | G13 | GND | H13 |
| $V_{DDINT}$ | E14 | GND | F14 | GND | G14 | GND | H14 |
| $V_{DDINT}$ | E15 | $V_{DDINT}$ | F15 | $V_{DDINT}$ | G15 | $V_{DDINT}$ | H15 |
| $V_{DDEXT}$ | E16 | $V_{DDEXT}$ | F16 | $V_{DDEXT}$ | G16 | $V_{DDEXT}$ | H16 |
| L1DAT[6] | E17 | L1DAT[4] | F17 | L1DAT[2] | G17 | L2DAT[5] | H17 |
| L1DAT[5] | E18 | L1DAT[3] | F18 | L2DAT[6] | G18 | L2ACK | H18 |
| L1ACK | E19 | L1DAT[0] | F19 | L2DAT[4] | G19 | L2DAT[3] | H19 |
| L1DAT[1] | E20 | L2DAT[7] | F20 | L2CLK | G20 | L2DAT[1] | H20 |

**Table 32. 400-ball Metric PBGA Pin Assignments (Continued)**

| Pin Name | PBGA Pin# | Pin Name | PBGA Pin# | Pin Name | PBGA Pin# | Pin Name | PBGA Pin# |
|---|---|---|---|---|---|---|---|
| DATA[44] | J01 | CLK_CFG_0 | K01 | CLKIN | L01 | $AV_{DD}$ | M01 |
| DATA[43] | J02 | DATA[46] | K02 | CLK_CFG_1 | L02 | CLK_CFG_3 | M02 |
| DATA[42] | J03 | DATA[45] | K03 | AGND | L03 | CLKOUT | M03 |
| DATA[41] | J04 | DATA[47] | K04 | CLK_CFG_2 | L04 | GND | M04 |
| $V_{DDEXT}$ | J05 | $V_{DDEXT}$ | K05 | $V_{DDEXT}$ | L05 | $V_{DDEXT}$ | M05 |
| $V_{DDINT}$ | J06 | $V_{DDINT}$ | K06 | $V_{DDINT}$ | L06 | $V_{DDINT}$ | M06 |
| GND | J07 | GND | K07 | GND | L07 | GND | M07 |
| GND | J08 | GND | K08 | GND | L08 | GND | M08 |
| GND | J09 | GND | K09 | GND | L09 | GND | M09 |
| GND | J10 | GND | K10 | GND | L10 | GND | M10 |
| GND | J11 | GND | K11 | GND | L11 | GND | M11 |
| GND | J12 | GND | K12 | GND | L12 | GND | M12 |
| GND | J13 | GND | K13 | GND | L13 | GND | M13 |
| GND | J14 | GND | K14 | GND | L14 | GND | M14 |
| $V_{DDINT}$ | J15 | $V_{DDINT}$ | K15 | $V_{DDINT}$ | L15 | $V_{DDINT}$ | M15 |
| $V_{DDEXT}$ | J16 | $V_{DDEXT}$ | K16 | $V_{DDEXT}$ | L16 | $V_{DDEXT}$ | M16 |
| L2DAT[2] | J17 | $\overline{BR6}$ | K17 | $\overline{BR2}$ | L17 | PAGE | M17 |
| L2DAT[0] | J18 | $\overline{BR5}$ | K18 | $\overline{BR1}$ | L18 | $\overline{SBTS}$ | M18 |
| $\overline{HBG}$ | J19 | $\overline{BR4}$ | K19 | ACK | L19 | $\overline{PA}$ | M19 |
| $\overline{HBR}$ | J20 | $\overline{BR3}$ | K20 | REDY | L20 | L3DAT[7] | M20 |

**Table 32. 400-ball Metric PBGA Pin Assignments (Continued)**

| Pin Name | PBGA Pin# | Pin Name | PBGA Pin# | Pin Name | PBGA Pin# | Pin Name | PBGA Pin# |
|---|---|---|---|---|---|---|---|
| NC | N01 | DATA[49] | P01 | DATA[53] | R01 | DATA[56] | T01 |
| NC | N02 | DATA[50] | P02 | DATA[54] | R02 | DATA[58] | T02 |
| DATA[48] | N03 | DATA[52] | P03 | DATA[57] | R03 | DATA[59] | T03 |
| DATA[51] | N04 | DATA[55] | P04 | DATA[60] | R04 | DATA[63] | T04 |
| $V_{DDEXT}$ | N05 | $V_{DDEXT}$ | P05 | $V_{DDEXT}$ | R05 | $V_{DDEXT}$ | T05 |
| $V_{DDINT}$ | N06 | $V_{DDINT}$ | P06 | $V_{DDINT}$ | R06 | $V_{DDINT}$ | T06 |
| GND | N07 | GND | P07 | GND | R07 | $V_{DDINT}$ | T07 |
| GND | N08 | GND | P08 | GND | R08 | $V_{DDINT}$ | T08 |
| GND | N09 | GND | P09 | GND | R09 | $V_{DDINT}$ | T09 |
| GND | N10 | GND | P10 | GND | R10 | $V_{DDINT}$ | T10 |
| GND | N11 | GND | P11 | GND | R11 | $V_{DDINT}$ | T11 |
| GND | N12 | GND | P12 | GND | R12 | $V_{DDINT}$ | T12 |
| GND | N13 | GND | P13 | GND | R13 | $V_{DDINT}$ | T13 |
| GND | N14 | GND | P14 | GND | R14 | $V_{DDINT}$ | T14 |
| $V_{DDINT}$ | N15 | $V_{DDINT}$ | P15 | GND | R15 | $V_{DDINT}$ | T15 |
| $V_{DDEXT}$ | N16 | $V_{DDEXT}$ | P16 | $V_{DDEXT}$ | R16 | $V_{DDEXT}$ | T16 |
| L3DAT[5] | N17 | L3DAT[2] | P17 | L4DAT[5] | R17 | L4DAT[3] | T17 |
| L3DAT[6] | N18 | L3DAT[1] | P18 | L4DAT[6] | R18 | L4ACK | T18 |
| L3DAT[4] | N19 | L3DAT[3] | P19 | L4DAT[7] | R19 | L4CLK | T19 |
| L3CLK | N20 | L3ACK | P20 | L3DAT[0] | R20 | L4DAT[4] | T20 |

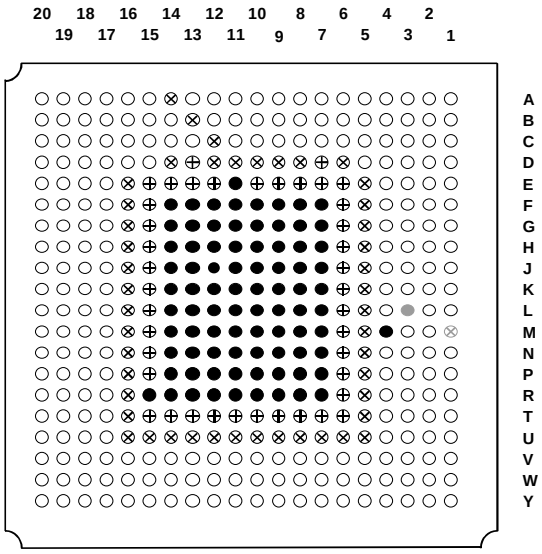
C02426–2.5–3/01(0)

PRINTED IN U.S.A.

**Table 32. 400-ball Metric PBGA Pin Assignments (Continued)**

| Pin Name | PBGA Pin# | Pin Name | PBGA Pin# | Pin Name | PBGA Pin# | Pin Name | PBGA Pin# |
|---|---|---|---|---|---|---|---|
| DATA[61] | U01 | ADDR[4] | V01 | ADDR[5] | W01 | ADDR[8] | Y01 |
| DATA[62] | U02 | ADDR[6] | V02 | ADDR[9] | W02 | ADDR[11] | Y02 |
| ADDR[3] | U03 | ADDR[7] | V03 | ADDR[12] | W03 | ADDR[13] | Y03 |
| ADDR[2] | U04 | ADDR[10] | V04 | ADDR[15] | W04 | ADDR[16] | Y04 |
| $V_{DDEXT}$ | U05 | ADDR[14] | V05 | ADDR[17] | W05 | ADDR[19] | Y05 |
| $V_{DDEXT}$ | U06 | ADDR[18] | V06 | ADDR[20] | W06 | ADDR[21] | Y06 |
| $V_{DDEXT}$ | U07 | ADDR[22] | V07 | ADDR[23] | W07 | ADDR[24] | Y07 |
| $V_{DDEXT}$ | U08 | ADDR[25] | V08 | ADDR[26] | W08 | ADDR[27] | Y08 |
| $V_{DDEXT}$ | U09 | ADDR[28] | V09 | ADDR[29] | W09 | ADDR[30] | Y09 |
| $V_{DDEXT}$ | U10 | ID0 | V10 | ID1 | W10 | ADDR[31] | Y10 |
| $V_{DDEXT}$ | U11 | ADDR[1] | V11 | ADDR[0] | W11 | ID2 | Y11 |
| $V_{DDEXT}$ | U12 | $\overline{MS1}$ | V12 | $\overline{BMS}$ | W12 | BRST | Y12 |
| $V_{DDEXT}$ | U13 | $\overline{CS}$ | V13 | $\overline{MS2}$ | W13 | $\overline{MS0}$ | Y13 |
| $V_{DDEXT}$ | U14 | $\overline{RDL}$ | V14 | $\overline{CIF}$ | W14 | $\overline{MS3}$ | Y14 |
| $V_{DDEXT}$ | U15 | $\overline{DMAR2}$ | V15 | $\overline{RDH}$ | W15 | $\overline{WRH}$ | Y15 |
| $V_{DDEXT}$ | U16 | L5DAT[0] | V16 | $\overline{DMAG2}$ | W16 | $\overline{WRL}$ | Y16 |
| L5DAT[7] | U17 | L5DAT[2] | V17 | LBOOT | W17 | $\overline{DMAG1}$ | Y17 |
| L4DAT[0] | U18 | L5ACK | V18 | L5DAT[1] | W18 | $\overline{DMAR1}$ | Y18 |
| L4DAT[1] | U19 | L5DAT[4] | V19 | L5DAT[3] | W19 | EBOOT | Y19 |
| L4DAT[2] | U20 | L5DAT[6] | V20 | L5DAT[5] | W20 | L5CLK | Y20 |

*400-ball Metric PBGA Pin Configurations (Bottom View, Summary)*

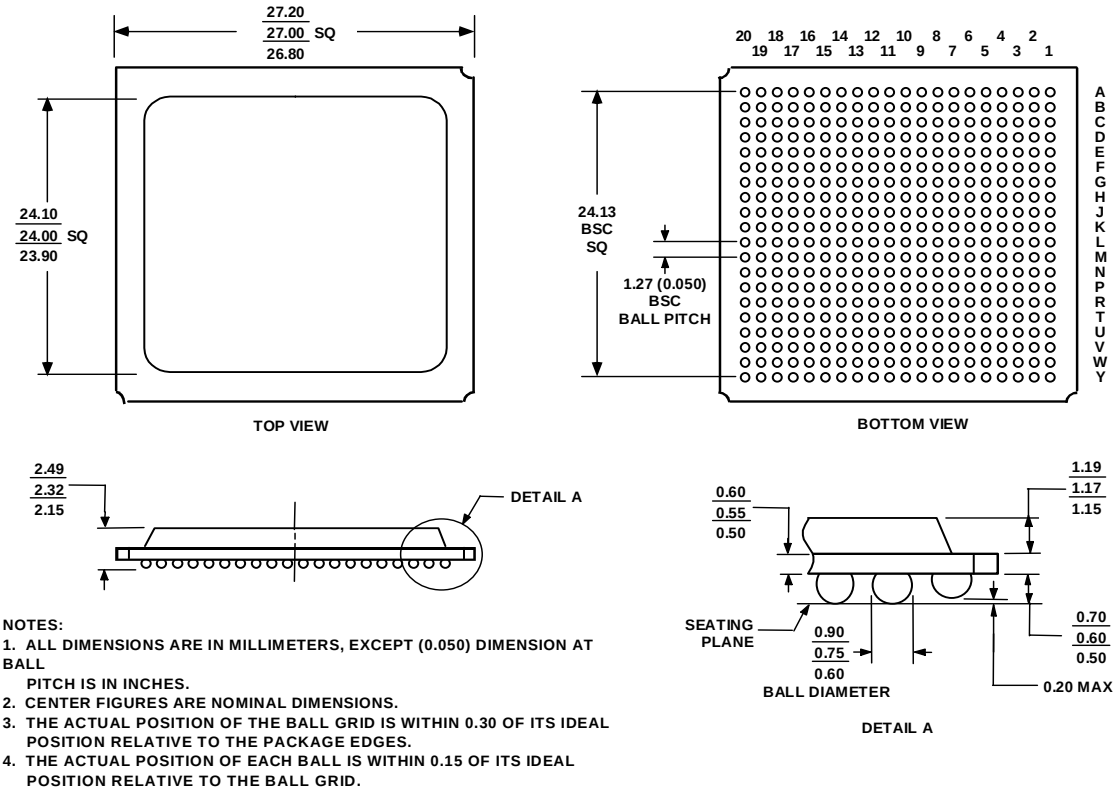



*** USE THE CENTER BLOCK OF GROUND PINS (PBGA BALLS: H8–13, J8–13, K8–13, L8–13, M8–13, AND N8–13) TO PROVIDE THERMAL PATHWAYS TO YOUR PRINTED CIRCUIT BOARD'S GROUND PLANE.**

## OUTLINE DIMENSIONS

The ADSP-21160N comes in a 27mm × 27mm, 400-ball
Metric PBGA package with 20 rows of balls.

*400-ball METRIC PBGA (B-400)*




NOTES:
1. ALL DIMENSIONS ARE IN MILLIMETERS, EXCEPT (0.050) DIMENSION AT BALL
   PITCH IS IN INCHES.
2. CENTER FIGURES ARE NOMINAL DIMENSIONS.
3. THE ACTUAL POSITION OF THE BALL GRID IS WITHIN 0.30 OF ITS IDEAL
   POSITION RELATIVE TO THE PACKAGE EDGES.
4. THE ACTUAL POSITION OF EACH BALL IS WITHIN 0.15 OF ITS IDEAL
   POSITION RELATIVE TO THE BALL GRID.

## ORDERING GUIDE

| Part Number[1] | Case Temperature Range | Instruction Rate | On-Chip SRAM | Operating Voltage |
|---|---|---|---|---|
| ADSP-21160NCB-TBD | –40°C to 100°C | TBD MHz | 4M bits | 1.9 INT/3.3 EXT V |
| ADSP-21160NKB-95 | 0°C to 85°C | 95 MHz | 4M bits | 1.9 INT/3.3 EXT V |

[1]B = Plastic Ball Grid Array (PBGA) package.