ANALOG DEVICES

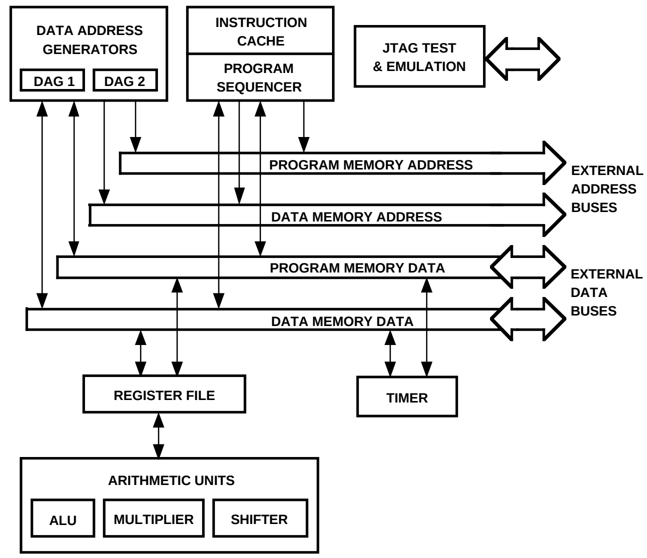
32/40-Bit IEEE Floating-Point DSP Microprocessor

ADSP-21020

FEATURES

- Superscalar IEEE Floating-Point Processor
- Off-Chip Harvard Architecture Maximizes Signal Processing Performance
- 30 ns, 33.3 MIPS Instruction Rate, Single-Cycle Execution
- 100 MFLOPS Peak, 66 MFLOPS Sustained Performance
- 1024-Point Complex FFT Benchmark: 0.58 ms
- Divide (y/x): 180 ns
- Inverse Square Root ($1/\sqrt{x}$): 270 ns
- 32-Bit Single-Precision and 40-Bit Extended-Precision IEEE Floating-Point Data Formats
- 32-Bit Fixed-Point Formats, Integer and Fractional, with 80-Bit Accumulators
- IEEE Exception Handling with Interrupt on Exception
- Three Independent Computation Units: Multiplier, ALU, and Barrel Shifter
- Dual Data Address Generators with Indirect, Immediate, Modulo, and Bit Reverse Addressing Modes
- Two Off-Chip Memory Transfers in Parallel with Instruction Fetch and Single-Cycle Multiply & ALU Operations
- Multiply with Add & Subtract for FFT Butterfly Computation
- Efficient Program Sequencing with Zero-Overhead Looping: Single-Cycle Loop Setup
- Single-Cycle Register File Context Switch
- 15 (or 25) ns External RAM Access Time for Zero-Wait-State, 30 (or 40) ns Instruction Execution
- IEEE JTAG Standard 1149.1 Test Access Port and On-Chip Emulation Circuitry
- 223-Pin PGA Package (Ceramic)

FUNCTIONAL BLOCK DIAGRAM



GENERAL DESCRIPTION

The ADSP-21020 is the first member of Analog Devices' family of single-chip IEEE floating-point processors optimized for digital signal processing applications. Its architecture is similar to that of Analog Devices' ADSP-2100 family of fixed-point DSP processors.

Fabricated in a high-speed, low-power CMOS process, the ADSP-21020 has a 30 ns instruction cycle time. With a high-performance on-chip instruction cache, the ADSP-21020 can execute every instruction in a single cycle.

The ADSP-21020 features:

- **Independent Parallel Computation Units**
  The arithmetic/logic unit (ALU), multiplier and shifter perform single-cycle instructions. The units are architecturally arranged in parallel, maximizing computational throughput. A single multifunction instruction executes parallel ALU and multiplier operations. These computation units support IEEE 32-bit single-precision floating-point, extended precision 40-bit floating-point, and 32-bit fixed-point data formats.
- **Data Register File**
  A general-purpose data register file is used for transferring data between the computation units and the data buses, and for storing intermediate results. This 10-port (16-register) register file, combined with the ADSP-21020's Harvard architecture, allows unconstrained data flow between computation units and off-chip memory.
- **Single-Cycle Fetch of Instruction and Two Operands**
  The ADSP-21020 uses a modified Harvard architecture in which data memory stores data and program memory stores both instructions and data. Because of its separate program and data memory buses and on-chip instruction cache, the processor can simultaneously fetch an operand from data memory, an operand from program memory, and an instruction from the cache, all in a single cycle.
- **Memory Interface**
  Addressing of external memory devices by the ADSP-21020 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-21020 provides programmable memory wait states, and external memory acknowledge controls allow interfacing to peripheral devices with variable access times.

REV. C

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.
Tel: 617/329-4700 Fax: 617/326-8703

- **Instruction Cache**
  The ADSP-21020 includes a high performance instruction cache that enables three-bus operation for fetching an instruction and two data values. The cache is selective—only the instructions whose fetches conflict with program memory data accesses are cached. This allows full-speed execution of core, looped operations such as digital filter multiply-accumulates and FFT butterfly processing.
- **Hardware Circular Buffers**
  The ADSP-21020 provides hardware to implement circular buffers in memory, which are common in digital filters and Fourier transform implementations. It handles address pointer wraparound, reducing overhead (thereby increasing performance) and simplifying implementation. Circular buffers can start and end at any location.
- **Flexible Instruction Set**
  The ADSP-21020's 48-bit instruction word accommodates a variety of parallel operations, for concise programming. For example, the ADSP-21020 can conditionally execute a multiply, an add, a subtract and a branch in a single instruction.

DEVELOPMENT SYSTEM

The ADSP-21020 is supported with a complete set of software and hardware development tools. The ADSP-21000 Family Development System includes development software, an evaluation board and an in-circuit emulator.

- **Assembler**
  Creates relocatable, COFF (Common Object File Format) object files from ADSP-21xxx assembly source code. It accepts standard C preprocessor directives for conditional assembly and macro processing. The algebraic syntax of the ADSP-21xxx assembly language facilitates coding and debugging of DSP algorithms.
- **Linker/Librarian**
  The Linker processes separately assembled object files and library files to create a single executable program. It assigns memory locations to code and to data in accordance with a user-defined architecture file that describes the memory and I/O configuration of the target system. The Librarian allows you to group frequently used object files into a single library file that can be linked with your main program.
- **Simulator**
  The Simulator performs interactive, instruction-level simulation of ADSP-21xxx code within the hardware configuration described by a system architecture file. It flags illegal operations and supports full symbolic disassembly. It provides an easy-to-use, window oriented, graphical user interface that is identical to the one used by the ADSP-21020 EZ-ICE Emulator. Commands are accessed from pull-down menus with a mouse.
- **PROM Splitter**
  Formats an executable file into files that can be used with an industry-standard PROM programmer.
- **C Compiler and Runtime Library**
  The C Compiler complies with ANSI specifications. It takes advantage of the ADSP-21020's high-level language architectural features and incorporates optimizing algorithms to speed up the execution of code. It includes an extensive runtime library with over 100 standard and DSP-specific functions.
- **C Source Level Debugger**
  A full-featured C source level debugger that works with the simulator or EZ-ICE emulator to allow debugging of assembler source, C source, or mixed assembler and C.
- **Numerical C Compiler**
  Supports ANSI Standard (X3J11.1) Numerical C as defined by the Numeric C Extensions Group. The compiler accepts C source input containing Numerical C extensions for array selection, vector math operations, complex data types, circular pointers, and variably dimensioned arrays, and outputs ADSP-21xxx assembly language source code.
- **ADSP-21020 EZ-LAB® Evaluation Board**
  The EZ-LAB Evaluation Board is a general-purpose, standalone ADSP-21020 system that includes 32K words of program memory and 32K words of data memory as well as analog I/O. A PC RS-232 download path enables the user to download and run programs directly on the EZ-LAB. In addition, it may be used in conjunction with the EZ-ICE Emulator to provide a powerful software debug environment.
- **ADSP-21020 EZ-ICE® Emulator**
  This in-circuit emulator provides the system designer with a PC-based development environment that allows nonintrusive access to the ADSP-21020's internal registers through the processor's 5-pin JTAG Test Access Port. This use of on-chip emulation circuitry enables reliable, full-speed performance in any target. The emulator uses the same graphical user interface as the ADSP-21020 Simulator, allowing an easy transition from software to hardware debug. (See "Target System Requirements for Use of EZ-ICE Emulator" on page 27.)

ADDITIONAL INFORMATION

This data sheet provides a general overview of ADSP-21020 functionality. For additional information on the architecture and instruction set of the processor, refer to the *ADSP-21020 User's Manual*. For development system and programming reference information, refer to the *ADSP-21000 Family Development Software Manuals* and the *ADSP-21020 Programmer's Quick Reference*. Applications code listings and benchmarks for key DSP algorithms are available on the DSP Applications BBS; call (617) 461-4258, 8 data bits, no parity, 1 stop bit, 300/1200/2400/9600 baud.

ARCHITECTURE OVERVIEW

Figure 1 shows a block diagram of the ADSP-21020. The processor features:

- Three Computation Units (ALU, Multiplier, and Shifter) with a Shared Data Register File
- Two Data Address Generators (DAG 1, DAG 2)
- Program Sequencer with Instruction Cache
- 32-Bit Timer
- Memory Buses and Interface
- JTAG Test Access Port and On-Chip Emulation Support

Computation Units

The ADSP-21020 contains three independent computation units: an ALU, a multiplier with fixed-point accumulator, and a shifter. In order to meet a wide variety of processing needs, the computation units process data in three formats: 32-bit fixed-point, 32-bit floating-point and 40-bit floating-point. The floating-point operations are single-precision IEEE-compatible (IEEE Standard 754/854). The 32-bit floating-point format is

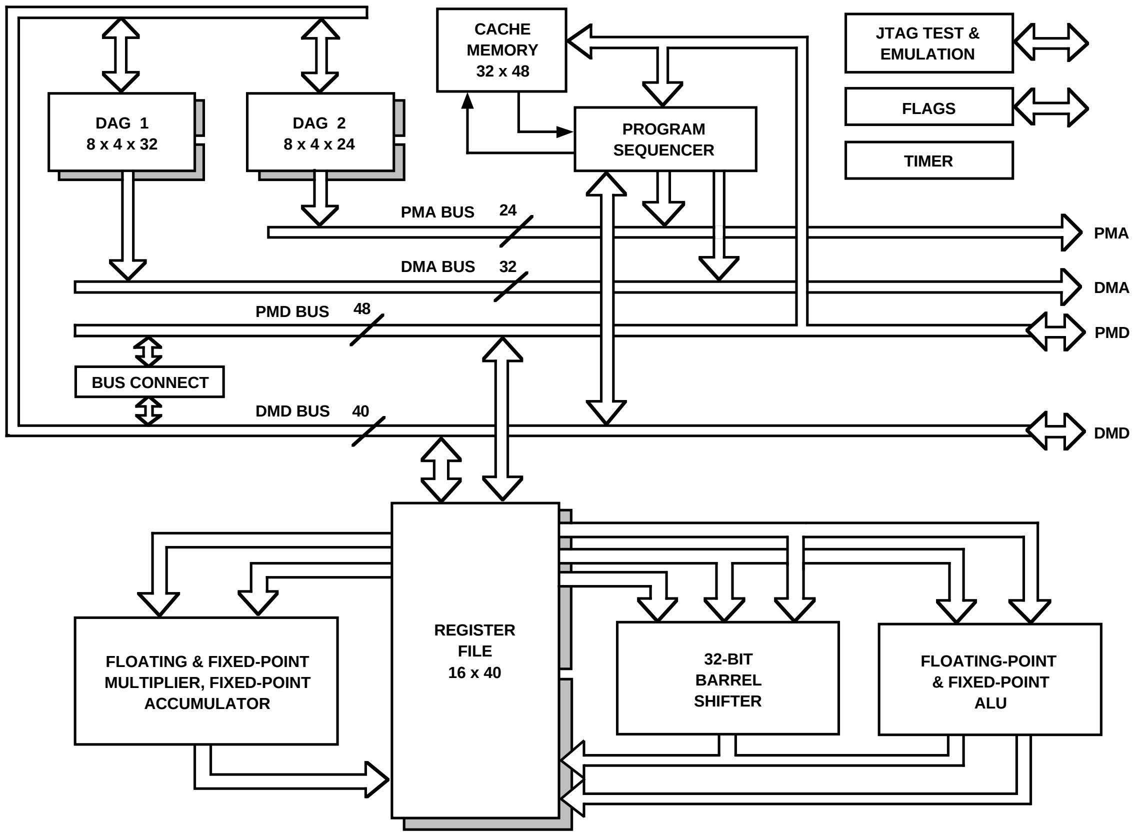


Figure 1. ADSP-21020 Block Diagram

the standard IEEE format, whereas the 40-bit IEEE extended-precision format has eight additional LSBs of mantissa for greater accuracy.

The multiplier performs floating-point and fixed-point multiplication as well as fixed-point multiply/add and multiply/subtract operations. Integer products are 64 bits wide, and the accumulator is 80 bits wide. The ALU performs 45 standard arithmetic and logic operations, supporting both fixed-point and floating-point formats. The shifter performs 19 different operations on 32-bit operands. These operations include logical and arithmetic shifts, bit manipulation, field deposit, and extract and derive exponent operations.

The computation units perform single-cycle operations; there is *no* computation pipeline. The three units are connected in parallel rather than serially, via multiple-bus connections with the 10-port data register file. The output of any computation unit may be used as the input of any unit on the next cycle. In a *multifunction* computation, the ALU and multiplier perform independent, simultaneous operations.

Data Register File

The ADSP-21020's general-purpose data register file is used for transferring data between the computation units and the data buses, and for storing intermediate results. The register file has two sets (primary and alternate) of sixteen 40-bit registers each, for fast context switching.

With a large number of buses connecting the registers to the computation units, data flow between computation units and from/to off-chip memory is unconstrained and free from bottlenecks. The 10-port register file and Harvard architecture of the ADSP-21020 allow the following nine data transfers to be performed every cycle:

- Off-chip read/write of two operands to or from the register file
- Two operands supplied to the ALU
- Two operands supplied to the multiplier
- Two results received from the ALU and multiplier (three, if the ALU operation is a combined addition/subtraction)

The processor's 48-bit orthogonal instruction word supports fully parallel data transfer and arithmetic operations in the same instruction.

Address Generators and Program Sequencer

Two dedicated address generators and a program sequencer supply addresses for memory accesses. Because of this, the computation units need never be used to calculate addresses. Because of its instruction cache, the ADSP-21020 can simultaneously fetch an instruction and data values from both off-chip program memory and off-chip data memory in a single cycle.

The data address generators (DAGs) provide memory addresses when external memory data is transferred over the parallel memory ports to or from internal registers. Dual data address generators enable the processor to output two simultaneous addresses for dual operand reads and writes. DAG 1 supplies 32-bit addresses to data memory. DAG 2 supplies 24-bit addresses to program memory for program memory data accesses.

Each DAG keeps track of up to eight address pointers, eight modifiers, eight buffer length values and eight base values. A pointer used for indirect addressing can be modified by a value in a specified register, either before (premodify) or after (postmodify) the access. To implement automatic modulo addressing for circular buffers, the ADSP-21020 provides buffer length registers that can be associated with each pointer. Base values for pointers allow circular buffers to be placed at arbitrary locations. Each DAG register has an alternate register that can be activated for fast context switching.

The program sequencer supplies instruction addresses to program memory. It controls loop iterations and evaluates conditional instructions. To execute looped code with zero overhead, the ADSP-21020 maintains an internal loop counter and loop stack. No explicit jump or decrement instructions are required to maintain the loop.

The ADSP-21020 derives its high clock rate from pipelined *fetch*, *decode* and *execute* cycles. Approximately 70% of the machine cycle is available for memory accesses; consequently, ADSP-21020 systems can be built using slower and therefore less expensive memory chips.

Instruction Cache

The program sequencer includes a high performance, selective instruction cache that enables three-bus operation for fetching an instruction and two data values. This two-way, set-associative cache holds 32 instructions. The cache is selective—only the instructions whose fetches conflict with program memory data accesses are cached, so the ADSP-21020 can perform a program memory data access and can execute the corresponding instruction in the same cycle. The program sequencer fetches the instruction from the cache instead of from program memory, enabling the ADSP-21020 to simultaneously access data in both program memory and data memory.

Context Switching

Many of the ADSP-21020's registers have alternate register sets that can be activated during interrupt servicing to facilitate a fast context switch. The data registers in the register file, DAG registers and the multiplier result register all have alternate sets. Registers active at reset are called *primary* registers; the others are called *alternate* registers. Bits in the MODE1 control register determine which registers are active at any particular time.

The primary/alternate select bits for each half of the register file (top eight or bottom eight registers) are independent. Likewise, the top four and bottom four register sets in each DAG have independent primary/ alternate select bits. This scheme allows passing of data between contexts.

Interrupts

The ADSP-21020 has four external hardware interrupts, nine internally generated interrupts, and eight software interrupts. For the external interrupts and the internal timer interrupt, the ADSP-21020 automatically stacks the arithmetic status and mode (MODE1) registers when servicing the interrupt, allowing five nesting levels of fast service for these interrupts.

An interrupt can occur at any time while the ADSP-21020 is executing a program. Internal events that generate interrupts include arithmetic exceptions, which allow for fast trap handling and recovery.

Timer

The programmable interval timer provides periodic interrupt generation. When enabled, the timer decrements a 32-bit count register every cycle. When this count register reaches zero, the ADSP-21020 generates an interrupt and asserts its TIMEXP output. The count register is automatically reloaded from a 32-bit period register and the count resumes immediately.

System Interface

Figure 2 shows an ADSP-21020 basic system configuration.

The external memory interface supports memory-mapped peripherals and slower memory with a user-defined combination of programmable wait states and hardware acknowledge signals. Both the program memory and data memory interfaces support addressing of page-mode DRAMs.

The ADSP-21020's internal functions are supported by four internal buses: the program memory address (PMA) and data memory address (DMA) buses are used for addresses associated with program and data memory. The program memory data (PMD) and data memory data (DMD) buses are used for data associated with the two memory spaces. These buses are extended off chip. Four data memory select (DMS) signals select one of four user-configurable banks of data memory. Similarly, two program memory select (PMS) signals select between two user-configurable banks of program memory. All banks are independently programmable for 0-7 wait states.

The PX registers permit passing data between program memory and data memory spaces. They provide a bridge between the 48-bit PMD bus and the 40-bit DMD bus or between the 40-bit register file and the PMD bus.

The PMA bus is 24 bits wide allowing direct access of up to 16M words of mixed instruction code and data. The PMD is 48 bits wide to accommodate the 48-bit instruction width. For access of 40-bit data the lower 8 bits are unused. For access of 32-bit data the lower 16 bits are ignored.

The DMA bus is 32 bits wide allowing direct access of up to 4 Gigawords of data. The DMD bus is 40 bits wide. For 32-bit data, the lower 8 bits are unused. The DMD bus provides a path for the contents of any register in the processor to be transferred to any other register or to any external data memory location in a single cycle. The data memory address comes from one of two sources: an absolute value specified in the instruction code (direct addressing) or the output of a data address generator (indirect addressing).

External devices can gain control of the processor's memory buses from the ADSP-21020 by means of the bus request/grant signals ($\overline{\text{BR}}$ and $\overline{\text{BG}}$). To grant its buses in response to a bus request, the ADSP-21020 halts internal operations and places its program and data memory interfaces in a high impedance state. In addition, three-state controls ($\overline{\text{DMTS}}$ and $\overline{\text{PMTS}}$) allow an external device to place either the program or data memory interface in a high impedance state without affecting the other interface and without halting the ADSP-21020 unless it requires a memory access from the affected interface. The three-state controls make it easy for an external cache controller to hold the ADSP-21020 off the bus while it updates an external cache memory.

JTAG Test and Emulation Support

The ADSP-21020 implements the boundary scan testing provisions specified by IEEE Standard 1149.1 of the Joint Testing Action Group (JTAG). The ADSP-21020's test access port and on-chip JTAG circuitry is fully compliant with the IEEE 1149.1 specification. The test access port enables boundary scan testing of circuitry connected to the ADSP-21020's I/O pins.

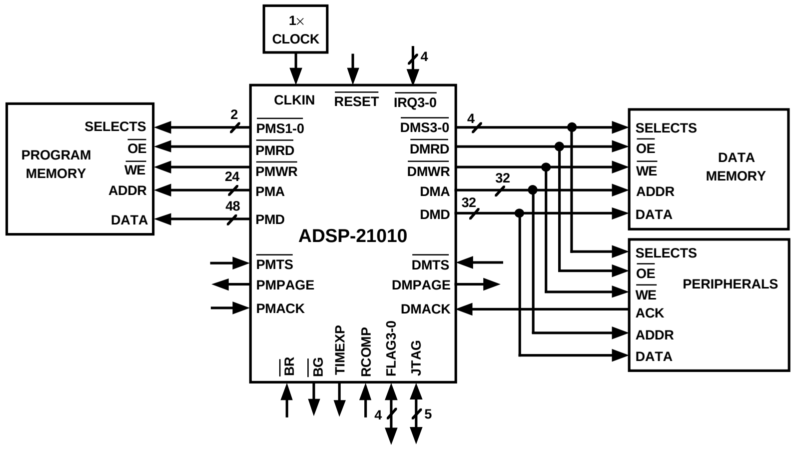


Figure 2. Basic System Configuration

The ADSP-21020 also implements on-chip emulation through the JTAG test access port. The processor's eight sets of break-point range registers enable program execution at full speed until reaching a desired break-point address range. The processor can then halt and allow reading/writing of all the processor's internal registers and external memories through the JTAG port.

PIN DESCRIPTIONS

This section describes the pins of the ADSP-21020. When groups of pins are identified with subscripts, e.g. PMD_{47-0}, the highest numbered pin is the MSB (in this case, PMD_{47}). Inputs identified as synchronous (S) must meet timing requirements with respect to CLKIN (or with respect to TCK for TMS, TDI, and $\overline{\text{TRST}}$). Those that are asynchronous (A) can be asserted asynchronously to CLKIN.

O = Output; I = Input; S = Synchronous; A = Asynchronous;
P = Power Supply; G = Ground.

Pin Name	Type	Function
PMA_{23-0}	O	Program Memory Address. The ADSP-21020 outputs an address in program memory on these pins.
PMD_{47-0}	I/O	Program Memory Data. The ADSP-21020 inputs and outputs data and instructions on these pins. 32-bit fixed-point data and 32-bit single-precision floating-point data is transferred over bits 47-16 of the PMD bus.
$\overline{\text{PMS}}_{1-0}$	O	Program Memory Select lines. These pins are asserted as chip selects for the corresponding banks of program memory. Memory banks must be defined in the memory control registers. These pins are decoded program memory address lines and provide an early indication of a possible bus cycle.
$\overline{\text{PMRD}}$	O	Program Memory Read strobe. This pin is asserted when the ADSP-21020 reads from program memory.
$\overline{\text{PMWR}}$	O	Program Memory Write strobe. This pin is asserted when the ADSP-21020 writes to program memory.
PMACK	I/S	Program Memory Acknowledge. An external device deasserts this input to add wait states to a memory access.
PMPAGE	O	Program Memory Page Boundary. The ADSP-21020 asserts this pin to signal that a program memory page boundary has been crossed. Memory pages must be defined in the memory control registers.
$\overline{\text{PMTS}}$	I/S	Program Memory Three-State Control. $\overline{\text{PMTS}}$ places the program memory address, data, selects, and strobes in a high-impedance state. If $\overline{\text{PMTS}}$ is asserted while a PM access is occurring, the processor will halt and the memory access will not be completed. PMACK must be asserted for at least one cycle when $\overline{\text{PMTS}}$ is deasserted to allow any pending memory access to complete properly. $\overline{\text{PMTS}}$ should only be asserted (low) during an active memory access cycle.
DMA_{31-0}	O	Data Memory Address. The ADSP-21020 outputs an address in data memory on these pins.
DMD_{39-0}	I/O	Data Memory Data. The ADSP-21020 inputs and outputs data on these pins. 32-bit fixed point data and 32-bit single-precision floating point data is transferred over bits 39-8 of the DMD bus.
$\overline{\text{DMS}}_{3-0}$	O	Data Memory Select lines. These pins are asserted as chip selects for the corresponding banks of data memory. Memory banks must be defined in the memory control registers. These pins are decoded data memory address lines and provide an early indication of a possible bus cycle.
$\overline{\text{DMRD}}$	O	Data Memory Read strobe. This pin is asserted when the ADSP-21020 reads from data memory.
$\overline{\text{DMWR}}$	O	Data Memory Write strobe. This pin is asserted when the ADSP-21020 writes to data memory.
DMACK	I/S	Data Memory Acknowledge. An external device deasserts this input to add wait states to a memory access.

Pin Name	Type	Function
DMPAGE	O	Data Memory Page Boundary. The ADSP-21020 asserts this pin to signal that a data memory page boundary has been crossed. Memory pages must be defined in the memory control registers.
$\overline{\text{DMTS}}$	I/S	Data Memory Three-State Control. $\overline{\text{DMTS}}$ places the data memory address, data, selects, and strobes in a high-impedance state. If $\overline{\text{DMTS}}$ is asserted while a DM access is occurring, the processor will halt and the memory access will not be completed. DMACK must be asserted for at least one cycle when $\overline{\text{DMTS}}$ is deasserted to allow any pending memory access to complete properly. $\overline{\text{DMTS}}$ should only be asserted (low) during an active memory access cycle.
CLKIN	I	External clock input to the ADSP-21020. The instruction cycle rate is equal to CLKIN. CLKIN may not be halted, changed, or operated below the specified frequency.
$\overline{\text{RESET}}$	I/A	Sets the ADSP-21020 to a known state and begins execution at the program memory location specified by the hardware reset vector (address). This input must be asserted (low) at power-up.
$\overline{\text{IRQ}}_{3-0}$	I/A	Interrupt request lines; may be either edge triggered or level-sensitive.
FLAG_{3-0}	I/O/A	External Flags. 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.
$\overline{\text{BR}}$	I/A	Bus Request. Used by an external device to request control of the memory interface. When $\overline{\text{BR}}$ is asserted, the processor halts execution after completion of the current cycle, places all memory data, addresses, selects, and strobes in a high-impedance state, and asserts $\overline{\text{BG}}$. The processor continues normal operation when $\overline{\text{BR}}$ is released.
$\overline{\text{BG}}$	O	Bus Grant. Acknowledges a bus request ($\overline{\text{BR}}$), indicating that the external device may take control of the memory interface. $\overline{\text{BG}}$ is asserted (held low) until $\overline{\text{BR}}$ is released.
TIMEXP	O	Timer Expired. Asserted for four cycles when the value of TCOUNT is decremented to zero.
RCOMP		Compensation Resistor input. Controls compensated output buffers. Connect RCOMP through a 1.8 kΩ ±15% resistor to EVDD. Use of a capacitor (approximately 100 pF), placed in parallel with the 1.8 kΩ resistor is recommended.
EVDD	P	Power supply (for output drivers), nominally +5 V dc (10 pins).
EGND	G	Power supply return (for output drivers); (16 pins).
IVDD	P	Power supply (for internal circuitry), nominally +5 V dc (4 pins).
IGND	G	Power supply return (for internal circuitry); (7 pins).
TCK	I	Test Clock. Provides an asynchronous clock for JTAG boundary scan.
TMS	I/S	Test Mode Select. Used to control the test state machine. TMS has a 20 kΩ internal pullup resistor.
TDI	VS	Test Data Input. Provides serial data for the boundary scan logic. TDI has a 20 kΩ internal pullup resistor.
TDO	O	Test Data Output. Serial scan output of the boundary scan path.
$\overline{\text{TRST}}$	I/A	Test Reset. 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-21020. $\overline{\text{TRST}}$ has a 20 kΩ internal pullup resistor.
NC		No Connect. No Connects are reserved pins that must be left open and unconnected.

INSTRUCTION SET SUMMARY

The ADSP-21020 instruction set provides a wide variety of programming capabilities. Every instruction assembles into a single word and can execute in a single processor cycle. Multifunction instructions enable simultaneous multiplier and ALU operations, as well as computations executed in parallel with data transfers. The addressing power of the ADSP-21020 gives you flexibility in moving data both internally and externally. The ADSP-21020 assembly language uses an algebraic syntax for ease of coding and readability.

The instruction types are grouped into four categories:
Compute and Move or Modify
Program Flow Control
Immediate Move
Miscellaneous

The instruction types are numbered; there are 22 types. Some instructions have more than one syntactical form; for example, Instruction 4 has four distinct forms. The instruction number itself has no bearing on programming, but corresponds to the opcode recognized by the ADSP-21020 device.

Because of the width and orthogonality of the instruction word, there are many possible instructions. For example, the ALU supports 21 fixed-point operations and 24 floating-point operations; each of these operations can be the compute portion of an instruction.

The following pages provide an overview and summary of the ADSP-21020 instruction set. For complete information, see the *ADSP-21020 User's Manual.* For additional reference information, see the *ADSP-21020 Programmer's Quick Reference.*

This section also contains several reference tables for using the instruction set.

- Table I describes the notation and abbreviations used.
- Table II lists all condition and termination code mnemonics.
- Table III lists all register mnemonics.
- Tables IV through VII list the syntax for all compute (ALU, multiplier, shifter or multifunction) operations.
- Table VIII lists interrupts and their vector addresses.

COMPUTE AND MOVE OR MODIFY INSTRUCTIONS

```
1.                     compute,    | DM(Ia, Mb) = dreg1 | ,        | PM(Ic, Md) = dreg2 |  ;
                                   | dreg1 = DM(Ia, Mb) |          | dreg2 = PM(Ic, Md) |
2.   IF condition      compute;
3a.  IF condition      compute,    | DM(Ia, Mb) |  = ureg ;
                                   | PM(Ic, Md) |
3b.  IF condition      compute,    | DM(Mb, Ia) |  = ureg ;
                                   | PM(Md, Ic) |
3c.  IF condition      compute,    ureg = | DM(Ia, Mb) |  ;
                                          | PM(Ic, Md) |
3d.  IF condition      compute,    ureg = | DM(Mb, Ia) |  ;
                                          | PM(Md, Ic) |
4a.  IF condition      compute,    | DM(Ia, <data6>) |  = dreg ;
                                   | PM(Ic, <data6>) |
4b.  IF condition      compute,    | DM(<data6>, Ia) |  = dreg ;
                                   | PM(<data6>, Ic) |
4c.  IF condition      compute,    dreg = | DM(Ia, <data6>) |  ;
                                          | PM(Ic, <data6>) |
4d.  IF condition      compute,    dreg = | DM(<data6>, Ia) |  ;
                                          | PM(<data6>, Ic) |
5.   IF condition      compute,    ureg1 = ureg2 ;
6a.  IF condition      shiftimm,   | DM(Ia, Mb) |  = dreg ;
                                   | PM(Ic, Md) |
6b.  IF condition      shiftimm,   dreg = | DM(Ia, Mb) |  ;
                                          | PM(Ic, Md) |
7.   IF condition      compute,    MODIFY | (Ia, Mb) |  ;
                                          | (Ic, Md) |
```

PROGRAM FLOW CONTROL INSTRUCTIONS

```
8.   IF condition    | JUMP |  | <addr24>         |     ( | DB     | ) ;
                     | CALL |  | (PC, <reladdr6>) |       | LA     |
                                                          | DB, LA |

9.   IF condition    | JUMP |  | (Md, Ic)         |     ( | DB     | ) , compute  ;
                     | CALL |  | (PC, <reladdr6>) |       | LA     |
                                                          | DB, LA |

11.  IF condition    | RTS |   ( | DB     | ) , compute  ;
                     | RTI |   ( | LA     |
                                 | DB, LA |

12.  LCNTR =         | <data16> | , DO | <addr24>            |   UNTIL LCE ;
                     | ureg     |      | (<PC, <reladdr24>)  |
13.                               DO | <addr24>            |   UNTIL termination ;
                                     |   (PC, <reladdr24>) |
```

(DB) Delayed branch
(LA) Loop abort (pop loop PC stacks on branch)

IMMEDIATE MOVE INSTRUCTIONS

```
14a. |DM(<addr32>)|  = ureg  ;
     |PM(<addr24>)|
14b. ureg =          |DM(<addr32>)|  ;
                     |PM(<addr24>)|
15a. |DM(<data32>, Ia)|  = ureg;
     |PM(< data24>, Ic)|
15b. ureg =          |DM(<data32>, Ia)|  ;
                     |PM(<data24>, Ic)|
16.  |DM(Ia, Mb)|  = <data32>;
     |PM(Ic, Md)|
17.  ureg = <data32>;
```

MISCELLANEOUS INSTRUCTIONS

```
18.  BIT          |SET|      sreg <data32>;
                  |CLR|
                  |TGL|
                  |TST|
                  |XOR|
19a. MODIFY       |(Ia, <data32>)| ;
                  |(Ic, <data32>)|
19b. BITREV        (Ia, <data32>)  ;
20.  |PUSH|     LOOP  ,        |PUSH|     STS ;
     |POP |                    |POP |
21. NOP ;
22. IDLE ;
```

Table I. Syntax Notation Conventions

Notation	Meaning
UPPERCASE	Explicit syntax—assembler keyword (notation only; assembler is not case-sensitive and lowercase is the preferred programming convention)
;	Instruction terminator
,	Separates parallel operations in an instruction
italics	Optional part of instruction
| between lines |	List of options (choose one)
<data*n*>	*n*-bit immediate data value
<addr*n*>	*n*-bit immediate address value
<reladdr*n*>	*n*-bit immediate PC-relative address value
compute	ALU, multiplier, shifter or multifunction operation (from Tables IV-VII)
shiftimm	Shifter immediate operation (from Table VI)
condition	Status condition (from Table II)
termination	Termination condition (from Table II)
ureg	Universal register (from Table III)
sreg	System register (from Table III)
dreg	R15-R0, F15-F0; register file location
Ia	I7-I0; DAG1 index register
Mb	M7-M0; DAG1 modify register
Ic	I15-I8; DAG2 index register
Md	M15-M8; DAG2 modify register

Table II. Condition and Termination Codes

Name	Description
eq	ALU equal to zero
ne	ALU not equal to zero
ge	ALU greater than or equal to zero
lt	ALU less than zero
le	ALU less than or equal to zero
gt	ALU greater than zero
ac	ALU carry
not ac	Not ALU carry
av	ALU overflow
not av	Not ALU overflow
mv	Multiplier overflow
not mv	Not multiplier overflow
ms	Multiplier sign
not ms	Not multiplier sign
sv	Shifter overflow
not sv	Not shifter overflow
sz	Shifter zero
not sz	Not shifter zero
flag0_in	Flag 0
not flag0_in	Not Flag 0
flag1_in	Flag 1
not flag1_in	Not Flag 1
flag2_in	Flag 2
not flag2_in	Not Flag 2
flag3_in	Flag 3
not flag3_in	Not Flag 3
tf	Bit test flag
not tf	Not bit test flag
lce	Loop counter expired (DO UNTIL)
not lce	Loop counter not expired (IF)
forever	Always False (DO UNTIL)
true	Always True (IF)

In a conditional instruction, the execution of the entire instruction is based on the specified condition.

Table III. Universal Registers

Name	Function
Register File	
R15–R0	Register file locations
Program Sequencer	
PC*	Program counter; address of instruction currently executing
PCSTK	Top of PC stack
PCSTKP	PC stack pointer
FADDR*	Fetch address
DADDR*	Decode address
LADDR	Loop termination address, code; top of loop address stack
CURLCNTR	Current loop counter; top of loop count stack
LCNTR	Loop count for next nested counter-controlled loop
Data Address Generators	
I7–I0	DAG1 index registers
M7–M0	DAG1 modify registers
L7–L0	DAG1 length registers
B7–B0	DAG1 base registers
I15–I8	DAG2 index registers
M15–M8	DAG2 modify registers
L15–L8	DAG2 length registers
B15–B8	DAG2 base registers
Bus Exchange	
PX1	PMD-DMD bus exchange 1 (16 bits)
PX2	PMD-DMD bus exchange 2 (32 bits)
PX	48-bit PX1 and PX2 combination
Timer	
TPERIOD	Timer period
TCOUNT	Timer counter
Memory Interface	
DMWAIT	Wait state and page size control for data memory
DMBANK1	Data memory bank 1 upper boundary
DMBANK2	Data memory bank 2 upper boundary
DMBANK3	Data memory bank 3 upper boundary
DMADR*	Copy of last data memory address
PMWAIT	Wait state and page size control for program memory
PMBANK1	Program memory bank 1 upper boundary
PMADR*	Copy of last program memory address
System Registers	
MODE1	Mode control bits for bit-reverse, alternate registers, interrupt nesting and enable, ALU saturation, floating-point rounding mode and boundary
MODE2	Mode control bits for interrupt sensitivity, cache disable and freeze, timer enable, and I/O flag configuration
IRPTL	Interrupt latch
IMASK	Interrupt mask
IMASKP	Interrupt mask pointer (for nesting)
ASTAT	Arithmetic status flags, bit test, I/O flag values, and compare accumulator
STKY	Sticky arithmetic status flags, circular buffer overflow flags, stack status flags (not sticky)
USTAT1	User status register 1
USTAT2	User status register 2

*read-only
Refer to User's Manual for bit-level definitions of each register.

Table IV. ALU Compute Operations

Fixed-Point	Floating-Point
Rn = Rx + Ry	Fn = Fx + Fy
Rn = Rx – Ry	Fn = Fx – Fy
Rn = Rx + Ry, Rm = Rx – Ry	Fn = Fx + Fy, Fm = Fx – Fy
Rn = Rx + Ry + CI	Fn = ABS (Fx + Fy)
Rn = Rx – Ry + CI – 1	Fn = ABS (Fx – Fy)
Rn = (Rx + Ry)/2	Fn = (Fx + Fy)/2
COMP(Rx, Ry)	COMP(Fx, Fy)
Rn = –Rx	Fn = –Fx
Rn = ABS Rx	Fn = ABS Fx
Rn = PASS Rx	Fn = PASS Fx
Rn = MIN(Rx, Ry)	Fn = MIN(Fx, Fy)
Rn = MAX(Rx, Ry)	Fn = MAX(Fx, Fy)
Rn = CLIP Rx BY Ry	Fn = CLIP Fx BY Fy
Rn = Rx + CI	Fn = RND Fx
Rn = Rx + CI – 1	Fn = SCALB Fx BY Ry
Rn = Rx + 1	Rn = MANT Fx
Rn = Rx – 1	Rn = LOGB Fx
Rn = Rx AND Ry	Rn = FIX Fx BY Ry
Rn = Rx OR Ry	Rn = FIX Fx
Rn = Rx XOR Ry	Fn = FLOAT Rx BY Ry
Rn = NOT Rx	Fn = FLOAT Rx
	Fn = RECIPS Fx
	Fn = RSQRTS Fx
	Fn = Fx COPYSIGN Fy

Rn, Rx, Ry R15–R0; register file location, fixed-point
Fn, Fx, Fy F15–F0; register file location, floating point

Table V. Multiplier Compute Operations

$$\begin{vmatrix} \text{Rn} \\ \text{MRF} \\ \text{MRB} \end{vmatrix} = \text{Rx} * \text{Ry} \left(\begin{vmatrix} S \\ U \end{vmatrix} \begin{vmatrix} S \\ U \end{vmatrix} \begin{vmatrix} F \\ I \\ FR \end{vmatrix} \right) \qquad \text{Fn} = \text{Fx} * \text{Fy}$$

$$\begin{vmatrix} \text{Rn} = \text{MRF} \\ \text{Rn} = \text{MRB} \\ \text{MRF} = \text{MRF} \\ \text{MRB} = \text{MRB} \end{vmatrix} + \text{Rx} * \text{Ry} \left(\begin{vmatrix} S \\ U \end{vmatrix} \begin{vmatrix} S \\ U \end{vmatrix} \begin{vmatrix} F \\ I \\ FR \end{vmatrix} \right) \qquad \begin{vmatrix} \text{Rn} = \text{MRF} \\ \text{Rn} = \text{MRB} \\ \text{MRF} = \text{MRF} \\ \text{MRB} = \text{MRB} \end{vmatrix} - \text{Rx} * \text{Ry} \left(\begin{vmatrix} S \\ U \end{vmatrix} \begin{vmatrix} S \\ U \end{vmatrix} \begin{vmatrix} F \\ I \\ FR \end{vmatrix} \right)$$

$$\begin{vmatrix} \text{Rn} = \text{SAT MRF} \\ \text{Rn} = \text{SAT MRB} \\ \text{MRF} = \text{SAT MRF} \\ \text{MRB} = \text{SAT MRB} \end{vmatrix} \begin{vmatrix} (SI) \\ (UI) \\ (SF) \\ (UF) \end{vmatrix} \qquad \begin{vmatrix} \text{Rn} = \text{RND MRF} \\ \text{Rn} = \text{RND MRB} \\ \text{MRF} = \text{RND MRF} \\ \text{MRB} = \text{RND MRB} \end{vmatrix} \begin{vmatrix} (SF) \\ (UF) \end{vmatrix}$$

$$\begin{vmatrix} \text{MRF} \\ \text{MRB} \end{vmatrix} = 0$$

$$\begin{vmatrix} \text{MRxF} \\ \text{MRxB} \end{vmatrix} = \text{Rn} \qquad \text{Rn} = \begin{vmatrix} \text{MRxF} \\ \text{MRxB} \end{vmatrix}$$

Rn, Rx, Ry	R15–R0; register file location, fixed-point
Fn, Fx, Fy	F15–F0; register file location, floating-point
MRxF	MR2F, MR1F; MR0F; multiplier result accumulators, foreground
MRxB	MR2B, MR1B, MR0B; multiplier result accumulators, background

$$\left(\begin{vmatrix} \text{x-input} \end{vmatrix} \begin{vmatrix} \text{y-input} \end{vmatrix} \begin{vmatrix} \text{data format,} \\ \text{rounding} \end{vmatrix} \right)$$

S	Signed input
U	Unsigned input
I	Integer input(s)
F	Fractional input(s)
FR	Fractional inputs, Rounded output
(SF)	Default format for 1-input operations
(SSF)	Default format for 2-input operations

Table VI. Shifter and Shifter Immediate Compute Operations

Shifter	Shifter Immediate
Rn = LSHIFT Rx BY Ry	Rn = LSHIFT Rx BY<data8>
Rn = Rn OR LSHIFT Rx BY Ry	Rn = Rn OR LSHIFT Rx BY<data8>
Rn = ASHIFT Rx BY Ry	Rn = ASHIFT Rx BY<data8>
Rn = Rn OR ASHIFT Rx BY Ry	Rn = Rn OR ASHIFT Rx BY<data8>
Rn = ROT Rx BY RY	Rn = ROT Rx BY<data8>
Rn = BCLR Rx BY Ry	Rn = BCLR Rx BY<data8>
Rn = BSET Rx BY Ry	Rn = BSET Rx BY<data8>
Rn = BTGL Rx BY Ry	Rn = BTGL Rx BY<data8>
BTST Rx BY Ry	BTST Rx BY<data8>
Rn = FDEP Rx BY Ry	Rn = FDEP Rx BY <bit6>: <len6>
Rn = Rn OR FDEP Rx BY Ry	Rn = Rn OR FDEP Rx BY <bit6>:<1en6>
Rn = FDEP Rx BY Ry (SE)	Rn = FDEP Rx BY <bit6>:<1en6> (SE)
Rn = Rn OR FDEP Rx BY Ry (SE)	Rn = Rn OR FDEP Rx BY <bit6>:<1en6> (SE)
Rn = FEXT Rx BY Ry	Rn = FEXT Rx BY <bit6>:<1en6>
Rn = FEXT Rx BY Ry (SE)	Rn = FEXT Rx BY <bit6>:<1en6> (SE)
Rn = EXP Rx	
Rn = EXP Rx (EX)	
Rn = LEFTZ Rx	
Rn = LEFTO Rx	

Rn, Rx, Ry R15-R0; register file location, fixed-point
<bit6>:<len6> 6-bit immediate bit position and length values (for shifter immediate operations)

Table Vll. Multifunction Compute Operations

Fixed-Point

Rm=R3-0 * R7-4 (SSFR), Ra=R11-8 + R15-12
Rm=R3-0 * R7-4 (SSFR), Ra=R11-8 – R15-12
Rm=R3-0 * R7-4 (SSFR), Ra=(R11-8 + R15-12)/2
MRF=MRF + R3-0 * R7-4 (SSF), Ra=R11-8 + R15-12
MRF=MRF + R3-0 * R7-4 (SSF), Ra=R11-8 – R15-12
MRF=MRF + R3-0 * R7-4 (SSF), Ra=(R11-8 + R15-12)/2
Rm=MRF + R3-0 * R7-4 (SSFR), Ra=R11-8 + R15-12
Rm=MRF + R3-0 * R7-4 (SSFR), Ra=R11-8 – R15-12
Rm=MRF + R3-0 * R7-4 (SSFR), Ra=(R11-8 + R15-12)/2
MRF=MRF – R3-0 * R7-4 (SSF), Ra=R11-8 + R15-12
MRF=MRF – R3-0 * R7-4 (SSF), Ra=R11-8 – R15-12
MRF=MRF – R3-0 * R7-4 (SSF), Ra=(R11-8 + R15-12)/2
Rm=MRF – R3-0 * R7-4 (SSFR), Ra=R11-8 + R15-12
Rm=MRF – R3-0 * R7-4 (SSFR), Ra=R11-8 – R15-12
Rm=MRF – R3-0 * R7-4 (SSFR), Ra=(R11-8 + R15-12)/2
Rm=R3-0 * R7-4 (SSFR), Ra=R11-8 + R15-12,
Rs=R11-8 – R15-12

Floating-Point

Fm=F3-0 * F7-4, Fa=F11-8 + F15-12
Fm=F3-0 * F7-4, Fa=F11-8 – F15-12
Fm=F3-0 * F7-4, Fa=FLOAT R11-8 by R15-12
Fm=F3-0 * F7-4, Fa=FIX R11-8 by R15-12
Fm=F3-0 * F7-4, Fa=(F11-8 + F15-12)/2
Fm=F3-0 * F7-4, Fa=ABS F11-8
Fm=F3-0 * F7-4, Fa=MAX (F11-8, F15-12)
Fm=F3-0 * F7-4, Fa=MIN (F11-8, F15-12)
Fm=F3-0 * F7-4, Fa=F11-8 + F15-12,
Fs=F11-8 – F15-12

Ra, Rm	Any register file location (fixed-point)
R3-0	R3, R2, R1, R0
R7-4	R7, R6, R5, R4
R11-8	R11, R10, R9, R8
R15-12	R15, R14, R13, R12
Fa, Fm	Any register file location (floating-point)
F3-0	F3, F2, F1, F0
F7-4	F7, F6, F5, F4
F11-8	F11, F10, F9, F8
F15-12	F15, F14, F13, F12
(SSF)	X-input signed, Y-input signed, fractional inputs
(SSFR)	X-input signed, Y-input signed, fractional inputs, rounded output

Table VIII. Interrupt Vector Addresses and Priorities

No.	Vector Address (Hex)	Function
0	0x00	Reserved
1*	0x08	Reset
2	0xl0	Reserved
3	0xl8	Status stack or loop stack overflow or PC stack full
4	0x20	Timer=0 (high priority option)
5	0x28	$\overline{\text{IRQ3}}$ asserted
6	0x30	$\overline{\text{IRQ2}}$ asserted
7	0x38	$\overline{\text{IRQ1}}$ asserted
8	0x40	$\overline{\text{IRQ0}}$ asserted
9	0x48	Reserved
10	0x50	Reserved
11	0x58	DAG 1 circular buffer 7 overflow
12	0x60	DAG 2 circular buffer 15 overflow
13	0x68	Reserved
14	0x70	Timer=0 (low priority option)
15	0x78	Fixed-point overflow
16	0x80	Floating-point overflow
17	0x88	Floating-point underflow
18	0x90	Floating-point invalid operation
19–23	0x98-0xB8	Reserved
24–31	0xC0–OxF8	User software interrupts

*Nonmaskable

RECOMMENDED OPERATING CONDITIONS

Parameter		K Grade Min	K Grade Max	B Grade Min	B Grade Max	T Grade Min	T Grade Max	Unit
V_{DD}	Supply Voltage	4.50	5.50	4.50	5.50	4.50	5.50	V
T_{AMB}	Ambient Operating Temperature	0	+70	–40	+85	–55	+125	°C

Refer to Environmental Conditions for information on thermal specifications.

ELECTRICAL CHARACTERISTICS

Parameter		Test Conditions	Min	Max	Unit
V_{IH}	Hi-Level Input Voltage[1]	V_{DD} = max	2.0		V
V_{IHCR}	Hi-Level Input Voltage[2, 12]	V_{DD} = max	3.0		V
V_{IL}	Lo-Level Input Voltage[1, 12]	V_{DD} = min		0.8	V
V_{ILC}	Lo-Level Input Voltage[2]	V_{DD} = max		0.6	V
V_{OH}	Hi-Level Output Voltage[3, 11]	V_{DD} = min, I_{OH} = –1.0 mA	2.4		V
V_{OL}	Lo-Level Output Voltage[3, 11]	V_{DD} = min, I_{OL} = 4.0 mA		0.4	V
I_{IH}	Hi-Level Input Current[4, 5]	V_{DD} = max, V_{IN} = V_{DD} max		10	µA
I_{IL}	Lo-Level Input Current[4]	V_{DD} = max, V_{IN} = 0 V		10	µA
I_{ILT}	Lo-Level Input Current[5]	V_{DD} = max, V_{IN} = 0 V		350	µA
I_{OZH}	Tristate Leakage Current[6]	V_{DD} = max, V_{IN} = V_{DD} max		10	µA
I_{OZL}	Tristate Leakage Current[6]	V_{DD} = max, V_{IN} = 0 V		10	µA
I_{DDIN}	Supply Current (Internal)[7]	t_{CK} = 30–33 ns, V_{DD} = max, V_{IHCR} = 3.0 V, V_{IH} = 2.4 V, V_{IL} = V_{ILC} = 0.4 V		490	mA
I_{DDIDLE}	Supply Current (Idle)[8]	V_{DD} = max, V_{IN} = 0 V or V_{DD} max		150	mA
C_{IN}	Input Capacitance[9, 10]	f_{IN} = 1 MHz, T_{CASE} = 25°C, V_{IN} = 2.5 V		10	pF

NOTES

[1]Applies to: PMD47–0, PMACK, $\overline{PMTS}$, DMD39–0, DMACK, $\overline{DMTS}$, $\overline{IRQ}$3–0. FLAG3–0, $\overline{BR}$, TMS, TDI.

[2]Applies to: CLKIN, TCK.

[3]Applies to: PMA23–0, PMD47–0, $\overline{PMS}$1–0, $\overline{PMRD}$, $\overline{PMWR}$, PMPAGE, DMA31–0, DMD39–0, $\overline{DMS}$3–0, $\overline{DMRD}$, $\overline{DMWR}$, DMPAGE, FLAG3–0, TIMEXP, $\overline{BG}$.

[4]Applies to: PMACK, $\overline{PMTS}$, DMACK, $\overline{DMTS}$, $\overline{IRQ}$3–0, $\overline{BR}$, CLKIN, $\overline{RESET}$, TCK.

[5]Applies to: TMS, TDI, $\overline{TRST}$.

[6]Applies to: PMA23–0, PMD47–0, $\overline{PMS}$1–0, $\overline{PMRD}$, $\overline{PMWR}$, PMPAGE, DMA31–0, DMD39–0, $\overline{DMS}$3–0, $\overline{DMRD}$, $\overline{DMWR}$, DMPAGE, FLAG3–0, TDO.

[7]Applies to IVDD pins. At t_{CK} = 30–33 ns, I_{DDIN} (typical) = 230 mA; at t_{CK} = 40 ns, I_{DDIN} (max) = 420 mA and I_{DDIN} (typical) = 200 mA; at t_{CK} = 50 ns, I_{DDIN} (max) = 370 mA and I_{DDIN} (typical) = 115 mA. See "Power Dissipation" for calculation of external (EVDD) supply current for total supply current.

[8]Applies to IVDD pins. Idle refers to ADSP-21020 state of operation during execution of the IDLE instruction.

[9]Guaranteed but not tested.

[10]Applies to all signal pins.

[11]Although specified for TTL outputs, all ADSP-21020 outputs are CMOS-compatible and will drive to V_{DD} and GND assuming no dc loads.

[12]Applies to $\overline{RESET}$, $\overline{TRST}$.

ABSOLUTE MAXIMUM RATINGS*

Supply Voltage . . . . –0.3 V to +7 V
Input Voltage . . . . –0.3 V to V_{DD} + 0.3 V
Output Voltage Swing . . . . –0.3 V to V_{DD} + 0.3 V
Load Capacitance . . . . 200 pF
Operating Temperature Range (Ambient) . . –55°C to +125°C
Storage Temperature Range . . . . –65°C to +150°C
Lead Temperature (10 seconds) CPGA . . . . +300°C

*Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

ESD SENSITIVITY

The ADSP-21020 features proprietary input protection circuitry to dissipate high energy discharges (Human Body Model). Per method 3015 of MIL-STD-883, the ADSP-21020 has been classified as a Class 3 device, with the ability to withstand up to 4000 V ESD.

Proper ESD precautions are strongly recommended to avoid functional damage or performance degradation. Charges readily accumulate on the human body and test equipment and discharge without detection. Unused devices must be stored in conductive foam or shunts, and the foam should be discharged to the destination socket before devices are removed. For further information on ESD precautions, refer to *Analog Devices' ESD Prevention Manual.*

TIMING PARAMETERS

General Notes

See Figure 15 on page 24 for voltage reference levels. Use the exact timing information given. Do not attempt to derive parameters from the addition or subtraction of others. While addition or subtraction would yield meaningful results for an individual device, the values given in this data sheet reflect statistical variations and worst cases. Consequently, you cannot meaningfully add parameters to derive other specifications.

Clock Signal

Parameter		K/B/T Grade 20 MHz Min	Max	K/B/T Grade 25 MHz Min	Max	B/T Grade 30 MHz Min	Max	K Grade 33.3 MHz Min	Max	Unit
Timing Requirement:										
t_{CK}	CLKIN Period	50	150	40	150	33	150	30	150	ns
t_{CKH}	CLKIN Width High	10		10		10		10		ns
t_{CKL}	CLKIN Width Low	10		10		10		10		ns

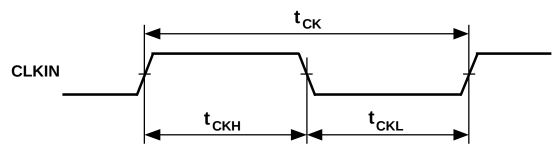


Figure 3. Clock

Reset

Parameter		K/B/T Grade 20 MHz Min	Max	K/B/T Grade 25 MHz Min	Max	B/T Grade 30 MHz Min	Max	K Grade 33.3 MHz Min	Max	Frequency Dependency* Min	Max	Unit
Timing Requirement:												
t_{WRST}[1]	RESET Width Low	200		160		132		120		$4t_{CK}$		ns
t_{SRST}[2]	RESET Setup before CLKIN High	29	50	24	40	21	33	19	30	29 + DT/2	30	ns

NOTES

$DT = t_{CK} - 50$ ns

[1]Applies after the power-up sequence is complete. At power up, the Internal Phase Locked Loop requires no more than 1000 CLKIN cycles while RESET is low, assuming stable V_{DD} and CLKIN (not including clock oscillator start-up time).

[2]Specification only applies in cases where multiple ADSP-21020 processors are required to execute in program counter lock-step (all processors start execution at location 8 in the same cycle). See the Hardware Configuration chapter of the *ADSP-21020 User's Manual* for reset sequence information.

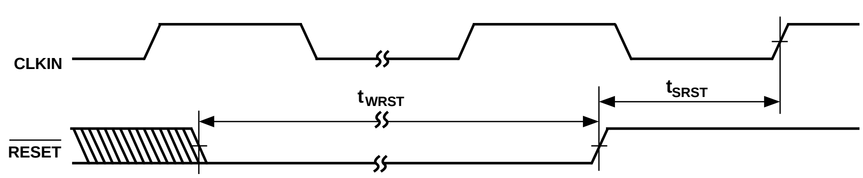


Figure 4. Reset

Interrupts

	K/B/T Grade		K/B/T Grade		B/T Grade		K Grade				
	20 MHz		25 MHz		30 MHz		33.3 MHz		Frequency Dependency*		
Parameter	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Unit**
Timing Requirement:											
t_{SIR} $\overline{\text{IRQ}}$3-0 Setup before CLKIN High	38		31		25		23		38 + 3DT/4		ns
t_{HIR} $\overline{\text{IRQ}}$3-0 Hold after CLKIN High	0		0		0		0			ns	
t_{IPW} $\overline{\text{IRQ}}$3-0 Pulse Width	55		45		38		35		t_{CK} + 5		ns

NOTE
*DT = t_{CK} – 50 ns
Meeting setup and hold guarantees interrupts will be latched in that cycle. Meeting the pulse width is not necessary if the setup and hold is met. Likewise, meeting the setup and hold is not necessary if the pulse width is met. See the Hardware Configuration chapter of the *ADSP-21020 User's Manual* for interrupt servicing information.

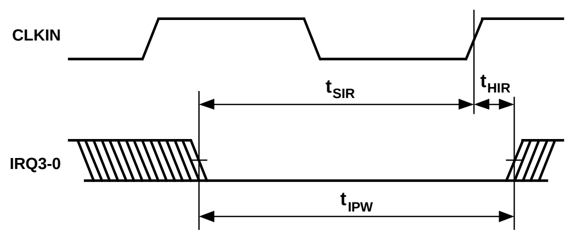


Figure 5. Interrupts

Timer

	K/B/T Grade		K/B/T Grade		B/T Grade		K Grade				
	20 MHz		25 MHz		30 MHz		33.3 MHz		Frequency Dependency*		
Parameter	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Unit**
Switching Characteristic:											
t_{DTEX} CLKIN High to TIMEXP		24		24		24		24			ns

NOTE
*DT = t_{CK} – 50 ns

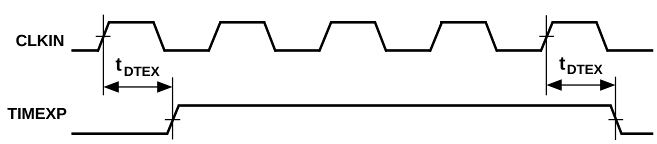


Figure 6. TIMEXP

Flags

Parameter		K/B/T Grade 20 MHz Min	Max	K/B/T Grade 25 MHz Min	Max	B/T Grade 30 MHz Min	Max	K Grade 33.3 MHz Min	Max	Frequency Dependency* Min	Max	Unit
Timing Requirement:[1]												
t_{SFI}	$FLAG3\text{-}0_{IN}$ Setup before CLKIN High	19		16		14		13		19 + 5DT/16		ns
t_{HFI}	$FLAG3\text{-}0_{IN}$ Hold after CLKIN High	0		0		0		0				ns
t_{DWRFI}	$FLAG3\text{-}0_{IN}$ Delay from $\overline{xRD}$, $\overline{xWR}$ Low		12		8		5		3		12 + 7DT/16	ns
t_{HFIWR}	$FLAG3\text{-}0_{IN}$ Hold after $\overline{xRD}$, $\overline{xWR}$ Deasserted	0		0		0		0				ns
Switching Characteristic:												
t_{DFO}	$FLAG3\text{-}0_{OUT}$ Delay from CLKIN High		24		24		24		24			ns
t_{HFO}	$FLAG3\text{-}0_{OUT}$ Hold after CLKIN High	5		5		5		5				ns
t_{DFOE}	CLKIN High to $FLAG3\text{-}0_{OUT}$ Enable	1		1		1		1				ns
t_{DFOD}	CLKIN High to $FLAG3\text{-}0_{OUT}$ Disable		24		24		24		24			ns

NOTES

*DT = t_{CK} – 50 ns

[1]Flag inputs meeting these setup and hold times will affect conditional operations in the next instruction cycle. See the Hardware Configuration chapter of the *ADSP-21020 User's Manual* for additional flag servicing information.

x = PM or DM.

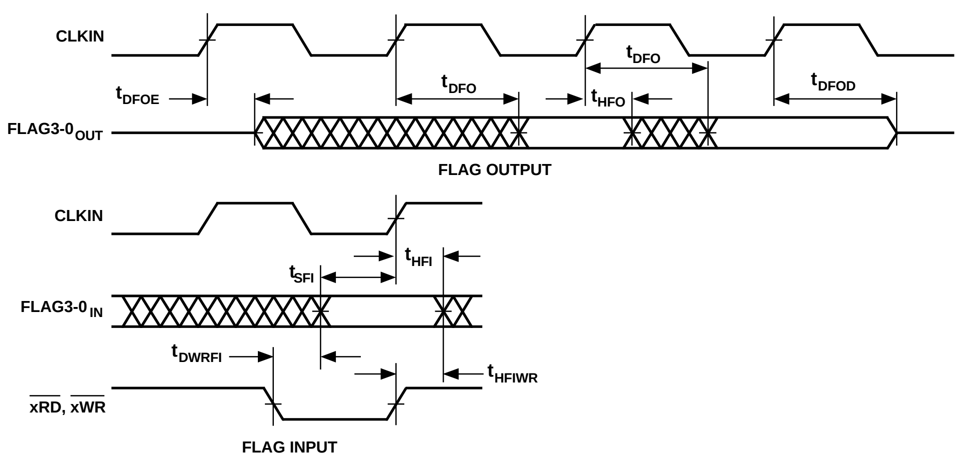


Figure 7. Flags

Bus Request/Bus Grant

		K/B/T Grade		K/B/T Grade		B/T Grade		K Grade				
		20 MHz		25 MHz		30 MHz		33.3 MHz		Frequency Dependency*		
Parameter		**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Unit**
Timing Requirement:												
t_{HBR}	$\overline{BR}$ Hold after CLKIN High	0		0		0		0				ns
t_{SBR}	$\overline{BR}$ Setup before CLKIN High	18		15		13		12		18 + 5DT/16		ns
Switching Characteristic:												
t_{DMDBGL}	Memory Interface Disable to $\overline{BG}$ Low	–2		–2		–2		–2				ns
t_{DME}	CLKIN High to Memory Interface Enable	25		20		16		15		25 + DT/2		ns
t_{DBGL}	CLKIN High to $\overline{BG}$ Low		22		22		22		22			ns
t_{DBGH}	CLKIN High to $\overline{BG}$ High		22		22		22		22			ns

NOTES
*DT = t_{CK} – 50 ns.
Memory Interface = PMA23-0, PMD47-0, $\overline{PMS1\text{-}0}$, $\overline{PMRD}$, $\overline{PMWR}$, PMPAGE, DMA31-0, DMD39-0, $\overline{DMS3\text{-}0}$, $\overline{DMRD}$, $\overline{DMWR}$, DMPAGE.
Buses are not granted until completion of current memory access.
See the Memory Interface chapter of the *ADSP-21020 User's Manual* for $\overline{BG}$, $\overline{BR}$ cycle relationships.

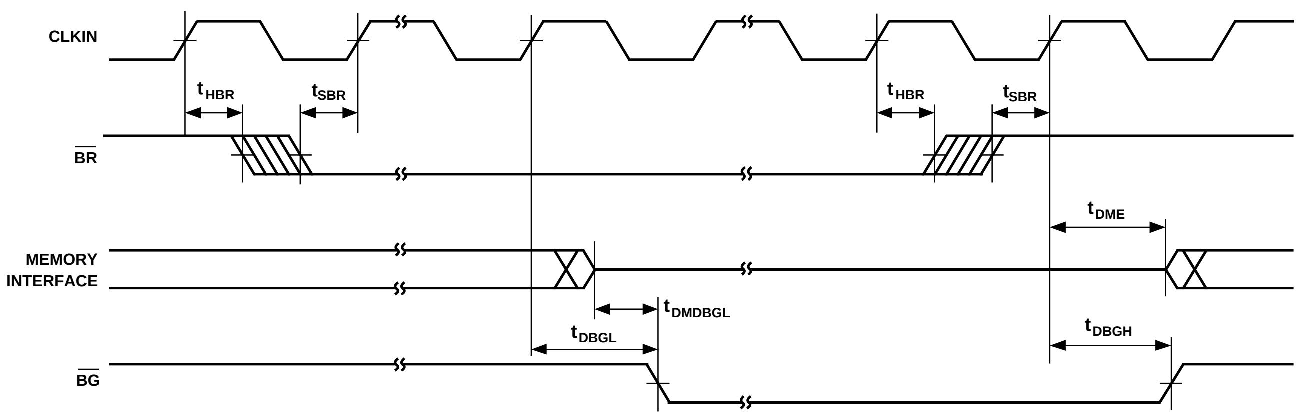


Figure 8. Bus Request/Bus Grant

External Memory Three-State Control

Parameter		K/B/T Grade 20 MHz Min	K/B/T Grade 20 MHz Max	K/B/T Grade 25 MHz Min	K/B/T Grade 25 MHz Max	B/T Grade 30 MHz Min	B/T Grade 30 MHz Max	K Grade 33.3 MHz Min	K Grade 33.3 MHz Max	Frequency Dependency* Min	Frequency Dependency* Max	Unit
Timing Requirement:												
t_{STS}	$\overline{xTS}$, Setup before CLKIN High	14	50	12	40	10	33	9	30	14 + DT/4	t_{CK}	ns
t_{DADTS}	$\overline{xTS}$ Delay after Address, Select		28		19		13		10		28 + 7DT/8	ns
t_{DSTS}	$\overline{xTS}$ Delay after $\overline{XRD}$, $\overline{XWR}$ Low		16		11		7		6		16 + DT/2	ns
Switching Characteristic:												
t_{DTSD}	Memory Interface Disable before CLKIN High	0		–2		–4		–5		DT/4		ns
t_{DTSAE}	$\overline{xTS}$ High to Address, Select Enable	0		0		0		0				ns

NOTES

*DT = t_{CK} – 50 ns.

$\overline{xTS}$ should only be asserted (low) during an active memory access cycle.

Memory Interface = PMA23-0, PMD47-0, $\overline{PMS1\text{-}0}$, $\overline{PMRD}$, $\overline{PMWR}$, PMPAGE, DMA31-0, DMD39-0, $\overline{DMS3\text{-}0}$, $\overline{DMRD}$, $\overline{DMWR}$, DMPAGE.

Address = PMA23-0, DMA31-0. Select = $\overline{PMS1\text{-}0}$, $\overline{DMS3\text{-}0}$.

x = PM or DM.

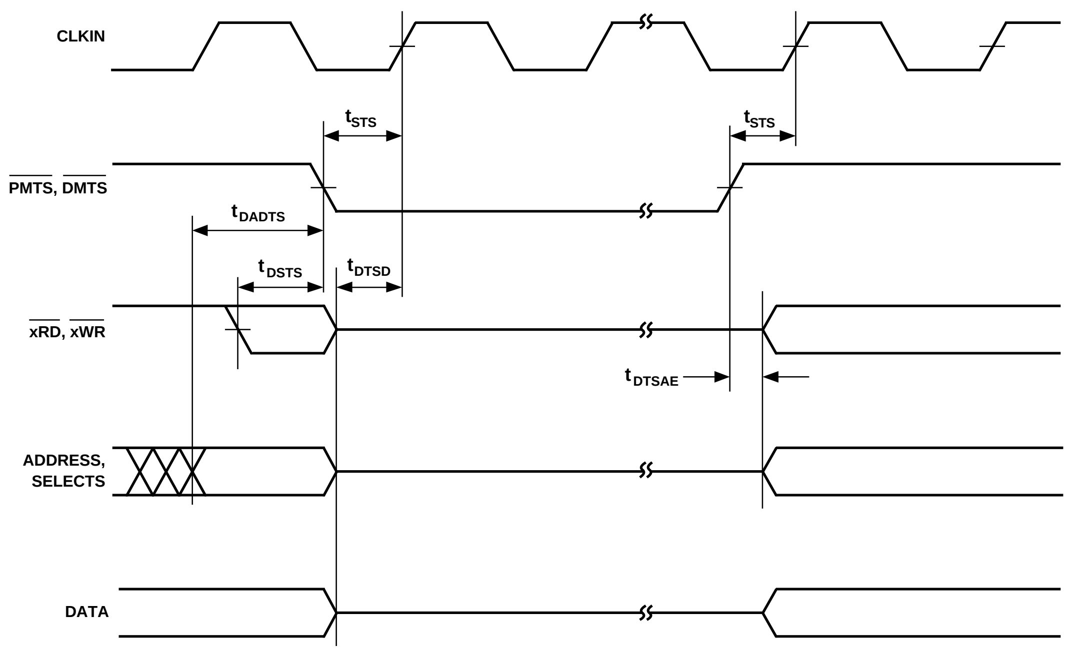


Figure 9. External Memory Three-State Control

Memory Read

Parameter		K/B/T Grade		K/B/T Grade		B/T Grade		K Grade				
		20 MHz		25 MHz		30 MHz		33.3 MHz		Frequency Dependence*		
		Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Unit
Timing Requirement:												
t_{DAD}	Address, Select to Data Valid		37		27		20		17		37 + DT	ns
t_{DRLD}	x$\overline{RD}$ Low to Data Valid		24		18		13		11		24 + 5DT/8	ns
t_{HDA}	Data Hold from Address, Select	0		0		0		0				ns
t_{HDRH}	Data Hold from x$\overline{RD}$ High	–1		–1		–1		–1				ns
t_{DAAK}	xACK Delay from Address		27		18		12		9		27 + 7DT/8	ns
t_{DRAK}	xACK Delay from x$\overline{RD}$ Low		15		10		6		5		15 + DT/2	ns
t_{SAK}	xACK Setup before CLKIN High	14		12		10		9		14 + DT/4		ns
t_{HAK}	xACK Hold after CLKIN High	0		0		0		0				ns
Switching Characteristic:												
t_{DARL}	Address, Select to x$\overline{RD}$ Low	8		4		2		0		8 + 3DT/8		ns
t_{DAP}	xPAGE Delay from Address, Select		1		1		1		1			ns
t_{DCKRL}	CLKIN High to x$\overline{RD}$ Low	16	26	13	24	12	22	11	21	16 + DT/4	26 + DT/4	ns
t_{RW}	x$\overline{RD}$ Pulse Width	26		20		15		13		26 + 5DT/8		ns
t_{RWR}	x$\overline{RD}$ High to x$\overline{RD}$, x$\overline{WD}$ Low	17		13		11		9		17 + 3DT/8		ns

NOTES

*DT = t_{CK} – 50 ns

x = PM or DM; Address = PMA23-0, DMA31-0; Data = PMD47-0, DMD39-0; Select = $\overline{PMS1\text{-}0}$, $\overline{DMS3\text{-}0}$.

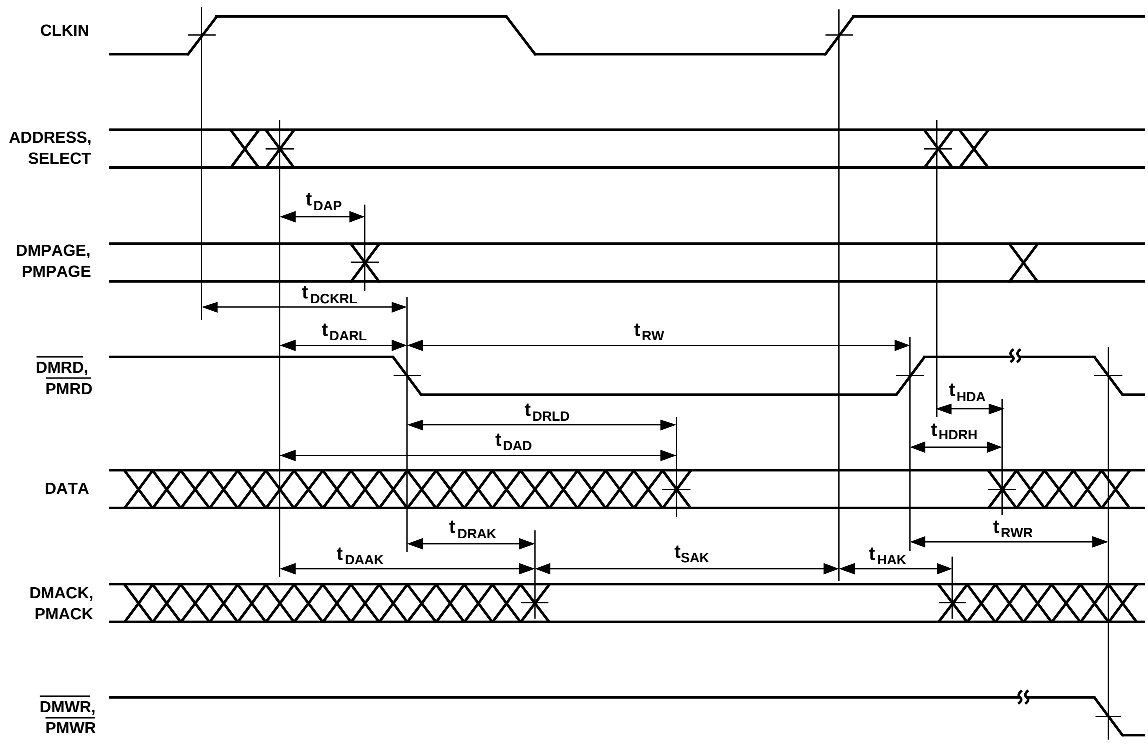


Figure 10. Memory Read

Memory Write

		K/B/T Grade		K/B/T Grade		B/T Grade		K Grade				
		20 MHz		25 MHz		30 MHz		33.3 MHz		Frequency Dependency*		
Parameter		Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Unit
Timing Requirement:							12					
t_{DAAK}	xACK Delay from Address, Select		27		18		6		9		27 + 7DT/8	ns
t_{DWAK}	xACK Delay from $\overline{\text{xWR}}$ Low		15		10	10			5		15 + DT/2	ns
t_{SAK}	xACK Setup before CLKIN High	14		12		0		9		14 + DT/4		ns
t_{HAK}	xACK Hold after CLKIN High	0		0				0				ns
Switching Characteristic:												
t_{DAWH}	Address, Select to $\overline{\text{xWR}}$ Deasserted	37		28		21		18		37+ 15DT/16		ns
t_{DAWL}	Address, Select to $\overline{\text{xWR}}$ Low	11		7		5		3		11 + 3DT/8		ns
t_{WW}	$\overline{\text{xWR}}$ Pulse Width	26		20		16		15		26 + 9DT/16		ns
t_{DDWH}	Data Setup before $\overline{\text{xWR}}$ High	23		18		14		13		23 + DT/2		ns
t_{DWHA}	Address, Select Hold after $\overline{\text{xWR}}$ Deasserted	1		0		0		0		1 + DT/16		ns
t_{HDWH}	Data Hold after $\overline{\text{xWR}}$ Deasserted[1]	0		–1		–1		–1		DT/16		ns
t_{DAP}	xPAGE Delay from Address, Select		1		1		1		1			ns
t_{DCKWL}	CLKIN High to $\overline{\text{xWR}}$ Low	16	26	13	24	12	22	11	21	16 + DT/4	26 + DT/4	ns
t_{WWR}	$\overline{\text{xWR}}$ High to $\overline{\text{xWR}}$ or $\overline{\text{xRD}}$ Low	17		13		10		8		17 + 7DT/16		ns
t_{DDWR}	Data Disable before $\overline{\text{xWR}}$ or $\overline{\text{xRD}}$ Low	13		9		7		5		13 + 3DT/8		ns
t_{WDE}	$\overline{\text{xWR}}$ Low to Data Enabled	0		–1		–1		–1		DT/16		ns

NOTES
*DT = t_C – 50 ns
See "System Hold Time Calculation" in "Test Conditions" section for calculating hold times given capacitive and DC loads.
x = PM or DM; Address = PMA23-0, DMA31-0; Data = PMD47-0, DMD39-0; Select = $\overline{\text{PMS1-0}}$, $\overline{\text{DMS3-0}}$.

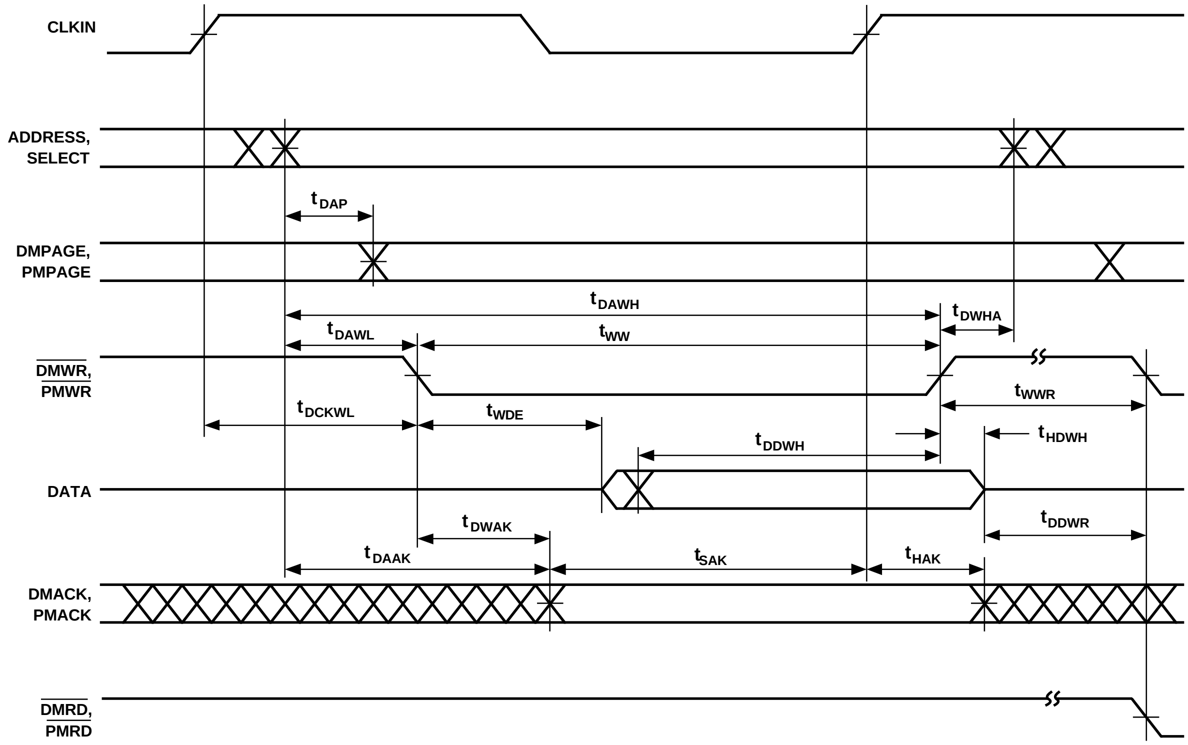


Figure 11. Memory Write

IEEE 1149.1 Test Access Port

		K/B/T Grade		K/B/T Grade		B/T Grade		K Grade				
		20 MHz		25 MHz		30 MHz		33.3 MHz		Frequency Dependency*		
Parameter		**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Unit**
Timing Requirement:												
t_{TCK}	TCK Period	50		40		33		30		t_{CK}		ns
t_{STAP}	TDI, TMS Setup before TCK High	5		5		5		5				ns
t_{HTAP}	TDI, TMS Hold after TCK High	6		6		6		6				ns
t_{SSYS}	System Inputs Setup before TCK High	7		7		7		7				ns
t_{HSYS}	System Inputs Hold after TCK High	9		9		9		9				ns
t_{TRSTW}	$\overline{\text{TRST}}$ Pulse Width	200		160		132		120				ns
Switching Characteristic:												
t_{DTDO}	TDO Delay from TCK Low		15		15		15		15			ns
t_{DSYS}	System Outputs Delay from TCK Low		26		26		26		26			ns

NOTES

*DT = t_C – 50 ns

System Inputs = PMD47-0, PMACK, $\overline{\text{PMTS}}$, DMD39-0, DMACK, $\overline{\text{DMTS}}$, CLKIN, $\overline{\text{IRQ3 0}}$, $\overline{\text{RESET}}$, FLAG3-0, $\overline{\text{BR}}$.

System Outputs = PMA23-0, $\overline{\text{PMS1-0}}$, $\overline{\text{PMRD}}$, $\overline{\text{PMWR}}$, PMD47-0, PMPAGE, DMA31-0, $\overline{\text{DMS1-0}}$, $\overline{\text{DMRD}}$, $\overline{\text{DMWR}}$, DMD39-0, DMPAGE, FLAG3-0, $\overline{\text{BG}}$, TIMEXP.

See the IEEE 1149.1 Test Access Port chapter of the *ADSP-21020 User's Manual* for further detail.

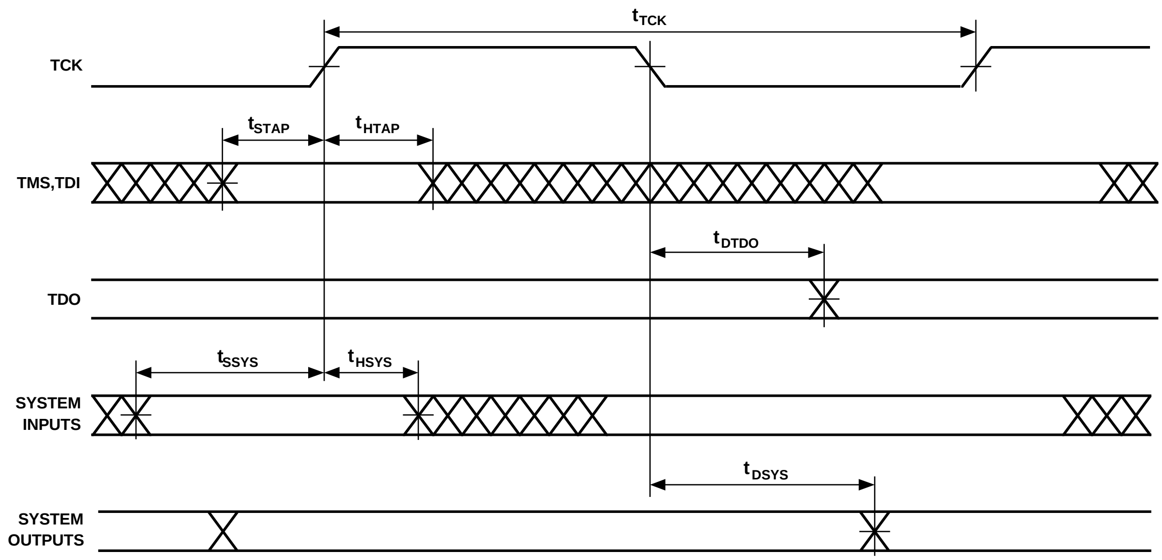


Figure 12. IEEE 1149.1 Test Access Port

TEST CONDITIONS

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. It can be approximated by the following equation:

$$t_{DECAY} = \frac{C_L \, \Delta V}{I_L}$$

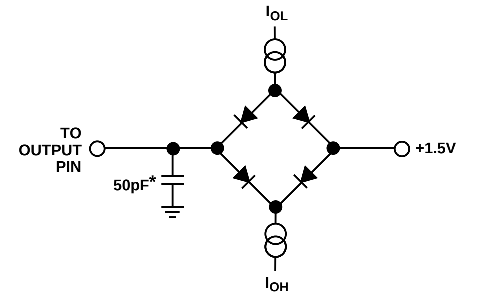
The output disable time (t_{DIS}) is the difference between $t_{MEASURED}$ and t_{DECAY} as shown in Figure 13. The time $t_{MEASURED}$) is the interval from when the reference signal switches to when the output voltage decays ΔV from the measured output high or output low voltage. t_{DECAY} is calculated with ΔV equal to 0.5 V, and test loads C_L and I_L.

Output Enable Time

Output pins are considered to be enabled when they have made a transition from a high-impedance state to when they start driving. The output enable time (t_{ENA}) is the interval from when a reference signal reaches a high or low voltage level to when the output has reached a specified high or low trip point, as shown in the Output Enable/Disable diagram. If multiple pins (such as the data bus) are enabled, the measurement value is that of the first pin to start driving.

Example System Hold Time Calculation

To determine the data output hold time in a particular system, first calculate t_{DECAY} using the above equation. Choose ΔV to be the difference between the ADSP-21020's output voltage and the input threshold for the device requiring the hold time. A typical ΔV 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_{HDWD} for the write cycle).



*AC TIMING SPECIFICATIONS ARE CALCULATED FOR 100pF DERATING ON THE FOLLOWING PINS: PMA23–0, PMS1–0, PMRD, PMWR, PMPAGE, DMA31–0, DMS3–0, DMRD, DMWR, DMPAGE

Figure 14. Equivalent Device Loading For AC Measurements (Includes All Fixtures)

Figure 15. Voltage Reference Levels For AC Measurements (Except Output Enable/Disable)

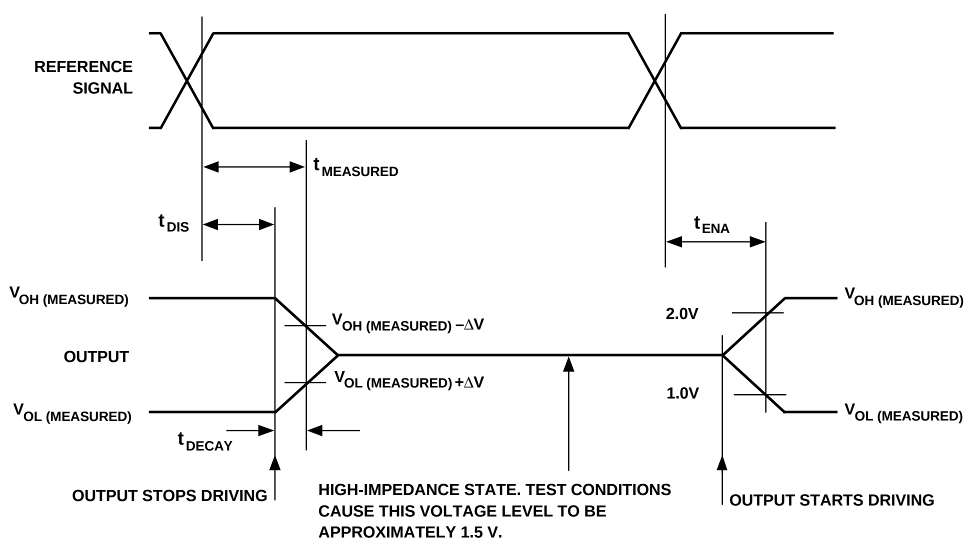
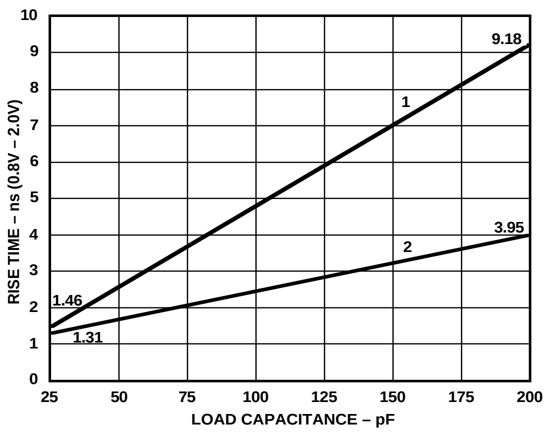


Figure 13. Output Enable/Disable

Capacitive Loading

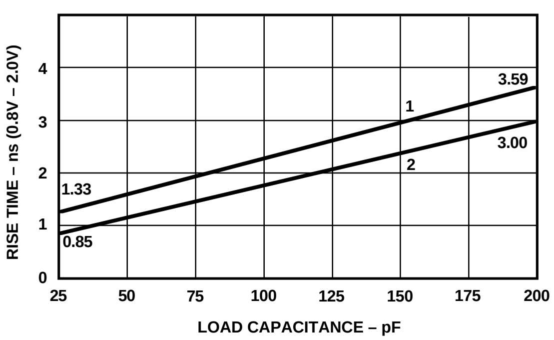
Output delays are based on standard capacitive loads: 100 pF on address, select, page and strobe pins, and 50 pF on all others (see Figure 14). For different loads, these timing parameters should be derated. See the Hardware Configuration chapter of the *ADSP-21020 User's Manual* for further information on derating of timing specifications.

Figures 16 and 17 show how the output rise time varies with capacitance. Figures 18 and 19 show how output delays vary with capacitance. Note that the graphs may not be linear outside the ranges shown.



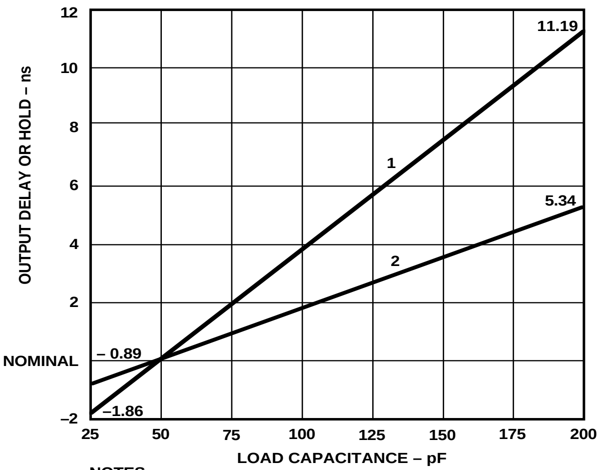
NOTES:
(1) OUTPUT PINS $\overline{BG}$, TIMEXP
(2) OUTPUT PINS PMD47–0, DMD39–0, FLAG3–0

Figure 16. Typical Output Rise Time vs. Load Capacitance (at Maximum Case Temperature)



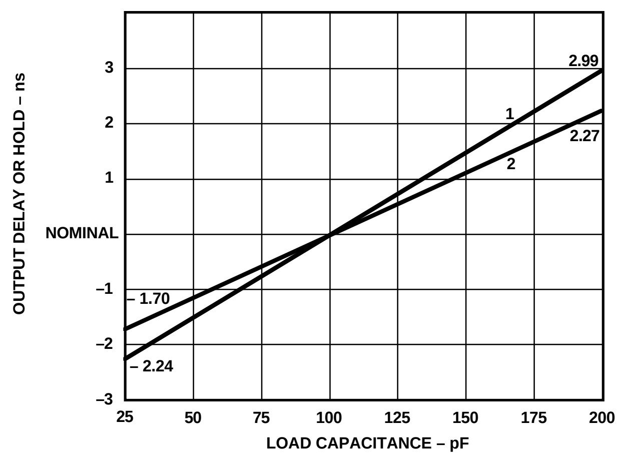
NOTES:
(1) OUTPUT PINS PMA23–0, $\overline{PMS1-0}$, PMPAGE, DMA31–0, $\overline{DMS3-0}$, DMPAGE, TDO
(2) OUTPUT PINS $\overline{PMRD}$, $\overline{PMWR}$, $\overline{DMRD}$, $\overline{DMWR}$

Figure 17. Typical Output Rise Time vs. Load Capacitance (at Maximum Case Temperature)



NOTES:
(1) OUTPUT PINS $\overline{BG}$, TIMEXP
(2) OUTPUT PINS PMD47–0, DMD39–0, FLAG3–0

Figure 18. Typical Output Delay or Hold vs. Load Capacitance (at Maximum Case Temperature)



NOTES:
(1) OUTPUT PINS PMA23–0, $\overline{PMS1-0}$, PMPAGE, DMA31–0, $\overline{DMS3-0}$, DMPAGE, TDO
(2) OUTPUT PINS $\overline{PMRD}$, $\overline{PMWR}$, $\overline{DMRD}$, $\overline{DMWR}$

Figure 19. Typical Output Delay or Hold vs. Load Capacitance (at Maximum Case Temperature)

ENVIRONMENTAL CONDITIONS

The ADSP-21020 is available in a Ceramic Pin Grid Array (CPGA). The package uses a cavity-down configuration which gives it favorable thermal characteristics. The top surface of the package contains a raised copper slug from which much of the die heat is dissipated. The slug provides a surface for mounting a heat sink (if required).

The commercial grade (K grade) ADSP-21020 is specified for operation at T_{AMB} of 0°C to +70°C. Maximum T_{CASE} (case temperature) can be calculated from the following equation:

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

where PD is power dissipation and θ_{CA} is the case-to-ambient thermal resistance. The value of PD depends on your application; the method for calculating PD is shown under "Power Dissipation" below. θ_{CA} varies with airflow and with the presence or absence of a heat sink. Table IX shows a range of θ_{CA} values.

Table IX. Maximum θ_{CA} for Various Airflow Values

Airflow (Linear ft./min.)	0	100	200	300
CPGA with No Heat Sink	12.8°C/W	9.2°C/W	6.6°C/W	5.5°C/W

NOTES

θ_{JC} is approximately 1°C/W.

Maximum recommended T_J is 130°C.

As per method 1012 MIL-STD-883. Ambient temperature: 25°C. Power: 3.5 W.

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 values involved. Internal power dissipation is calculated in the following way:

$$\mathbf{P_{INT} = I_{DDIN} \times V_{DD}}$$

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

1) the number of output pins that switch during each cycle (O),
2) the maximum frequency at which they can switch (f),
3) their load capacitance (C), and
4) their voltage swing (V_{DD}).

It is calculated by:

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

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 $1/(2t_{CK})$. The write strobes can switch every cycle at a frequency of $1/t_{CK}$. Select pins switch at $1/(2t_{CK})$, but 2 DM and 2 PM selects can switch on each cycle. If only one bank is accessed, no select line will switch.

Example:

Estimate P_{EXT} with the following assumptions:

- A system with one RAM bank each of PM (48 bits) and DM (32 bits).
- 32K × 8 RAM chips are used, each with a load of 10 pF.
- Single-precision mode is enabled so that only 32 data pins can switch at once.
- PM and DM writes occur every other cycle, with 50% of the pins switching.
- The instruction cycle rate is 20 MHz (t_{CK} = 50 ns) and V_{DD} = 5.0 V.

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

Pin Type	# Pins	% Switch	× C	× f	× V_{DD}^2	P_{EXT}
PMA	15	50	68 pF	5 MHz	25 V	0.064 W
$\overline{\text{PMS}}$	2	0	68 pF	5 MHz	25 V	0.000 W
$\overline{\text{PMWR}}$	1	—	68 pF	10 MHz	25 V	0.017 W
PMD	32	50	18 pF	5 MHz	25 V	0.036 W
DMA	15	50	48 pF	5 MHz	25 V	0.045 W
$\overline{\text{DMS}}$	2	0	48 pF	5 MHz	25 V	0.000 W
$\overline{\text{DMWR}}$	1	—	48 pF	10 MHz	25 V	0.012 W
DMD	32	50	18 pF	5 MHz	25 V	0.036 W

P_{EXT} =0.210 W

A typical power consumption can now be calculated for this situation by adding a typical internal power dissipation:

$$\mathbf{P_{TOTAL} = P_{EXT} + (5\ V \times I_{DDIN}\ (typ)) = 0.210 + 1.15 = 1.36\ W}$$

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. Also note that it is not common for a program to have 100% or even 50% of the outputs switching simultaneously.

Power and Ground Guidelines

To achieve its fast cycle time, including instruction fetch, data access, and execution, the ADSP-21020 is designed with high speed drivers on all output pins. Large peak currents may pass through a circuit board's ground and power lines, especially when many output drivers are simultaneously charging or discharging their load capacitances. These transient currents can cause disturbances on the power and ground lines. To minimize these effects, the ADSP-21020 provides separate supply pins for its internal logic (IGND and IVDD) and for its external drivers (EGND and EVDD).

To reduce system noise at low temperatures when transistors switch fastest, the ADSP-21020 employs compensated output drivers. These drivers equalize slew rate over temperature extremes and process variations. A 1.8 kΩ resistor placed between the RCOMP pin and EVDD (+5 V) provides a reference for the compensated drivers. Use of a capacitor (approximately 100 pF), placed in parallel with the 1.8 kΩ resistor, is recommended.

All GND pins should have a low impedance path to ground. A ground plane is required in ADSP-21020 systems to reduce this impedance, minimizing noise.

The EVDD and IVDD pins should be bypassed to the ground plane using approximately 14 high-frequency capacitors (0.1 μF ceramic). Keep each capacitor's lead and trace length to the pins as short as possible. This low inductive path provides the ADSP-21020 with the peak currents required when its output drivers switch. The capacitors' ground leads should also be short and connect directly to the ground plane. This provides a low impedance return path for the load capacitance of the ADSP-21020's output drivers.

If a V_{DD} plane is not used, the following recommendations apply. Traces from the +5 V supply to the 10 EVDD pins should be designed to satisfy the minimum V_{DD} specification while carrying average dc currents of [I_{DDEX}/10 × (number of EVDD pins per trace)]. I_{DDEX} is the calculated external supply current. A similar calculation should be made for the four IVDD pins using the I_{DDIN} specification. The traces connecting +5 V to the IVDD pins should be separate from those connecting to the EVDD pins.

A low frequency bypass capacitor (20 μF tantalum) located near the junction of the IVDD and EVDD traces is also recommended.

Target System Requirements For Use Of EZ-ICE Emulator
The ADSP-21020 EZ-ICE uses the IEEE 1149.1 JTAG test access port of the ADSP-21020 to monitor and control the target board processor during emulation. The EZ-ICE probe requires that CLKIN, TMS, TCK, $\overline{\text{TRST}}$, TDI, TDO, and GND be made accessible on the target system via a 12-pin connector (pin strip header) such as that shown in Figure 20. The EZ-ICE probe plugs directly onto this connector for chip-on-board emulation; you must add this connector to your target board design if you intend to use the ADSP-21020 EZ-ICE. Figure 21 shows the dimensions of the EZ-ICE probe; be sure to allow enough space in your system to fit the probe onto the 12-pin connector.

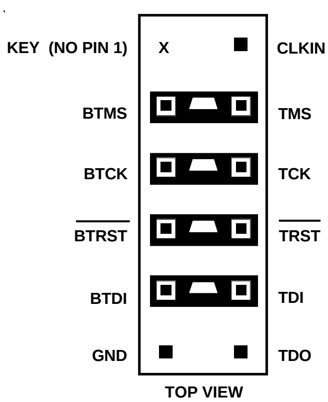


Figure 20. Target Board Connector for EZ-ICE Emulator (Jumpers In Place)

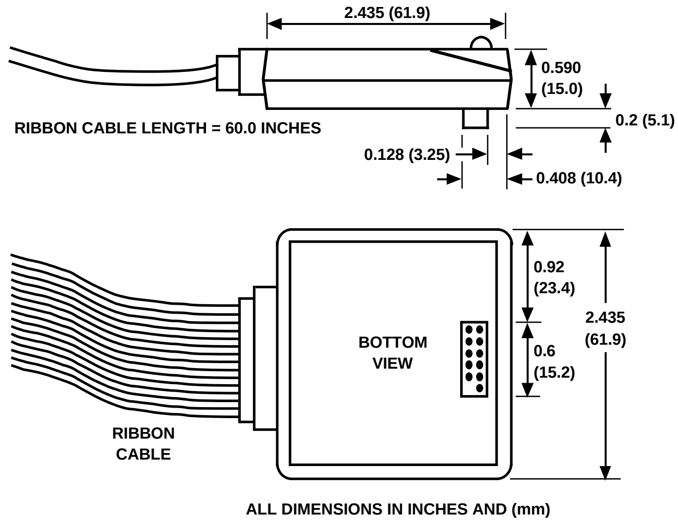


Figure 21. EZ-ICE Probe

The 12-pin, 2-row pin strip header is keyed at the Pin 1 location –you must clip Pin 1 off of the header. The pins must be 0.025 inch square and at least 0.20 inch in length. Pin spacing is 0.1 × 0.1 inches.

The tip of the pins must be at least 0.10 inch higher than the tallest component under the probe to allow clearance for the bottom of the probe. Pin strip headers are available from vendors such as 3M, McKenzie, and Samtec.

The length of the traces between the EZ-ICE probe connector and the ADSP-21020 test access port pins should be less than 1 inch. Note that the EZ-ICE probe adds two TTL loads to the CKIN pin of the ADSP-21020.

The BMTS, BTCK, $\overline{\text{BTRST}}$, and BTDI signals are provided so that the test access port can also be used for board-level testing. When the connector is not being used for emulation, place jumpers between the BXXX pins and the XXX pins as shown in Figure 20. If you are not going to use the test access port for board test, tie $\overline{\text{BTRST}}$ to GND and tie or pull up BTCK to VDD. The $\overline{\text{TRST}}$ pin must be asserted (pulsed low) after power up (through $\overline{\text{BTRST}}$ on the connector) or held low for proper operation of the ADSP-21020.

	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	
U	PMA17	PMA20	TMS	EGND	TCK	EVDD	RCOMP	EGND	PMACK	EVDD	$\overline{\text{PMWR}}$	EGND	PMD44	EGND	PMD40	PMD39	PMD35	PMD31	U
T	EGND	PMA19	PMA23	$\overline{\text{PMS1}}$	$\overline{\text{TRST}}$	$\overline{\text{DMWR}}$	DMACK	CLKIN	NC	NC	$\overline{\text{PMTS}}$	PMD45	PMD42	NC	PMD37	PMD32	PMD30	PMD27	T
S	PMA11	PMA14	PMA18	PMA22	PMPAGE	TDI	$\overline{\text{DMTS}}$	$\overline{\text{DMRD}}$	NC	$\overline{\text{PMRD}}$	PMD47	PMD43	PMD41	PMD36	PMD34	PMD28	PMD26	PMD21	S
R	EGND	PMA10	PMA15	PMA16	PMA21	$\overline{\text{PMS0}}$	TDO	IGND	$\overline{\text{RESET}}$	IVDD	PMD46	IGND	PMD38	PMD33	PMD29	PMD25	PMD23	EGND	R
P	PMA8	PMA9	PMA13	PMA12											PMD24	PMD22	PMD19	PMD18	P
N	EVDD	PMA5	PMA6	PMA7											PMD20	PMD17	PMD16	EVDD	N
M	PMA1	PMA4	PMA3	PMA2											PMD15	PMD14	PMD13	PMD12	M
L	EGND	PMA0	TIMEXP	IGND											IGND	PMD10	PMD11	EGND	L
K	EVDD	NC	$\overline{\text{IRQ2}}$	$\overline{\text{IRQ3}}$											PMD6	PMD7	PMD8	PMD9	K
J	EVDD	$\overline{\text{IRQ0}}$	$\overline{\text{IRQ1}}$	IVDD											IVDD	PMD2	PMD5	EVDD	J
H	EGND	FLAG2	FLAG0	FLAG1											DMD1	DMD0	PMD3	PMD4	H
G	FLAG3	DMA1	DMA0	IGND											IGND	DMD3	NC	EGND	G
F	DMA2	DMA3	DMA4	DMA5											DMD9	DMD6	PMD0	PMD1	F
E	DMA6	DMA7	DMA8	DMA10											DMD13	DMD10	DMD2	EGND	E
D	DMA9	DMA11	DMA12	DMA15	DMA19	DMA23	DMA27	IGND	$\overline{\text{DMS0}}$	IVDD	DMD36	DMD31	DMD27	DMD22	DMD17	DMD11	DMD5	DMD4	D
C	DMA13	DMA14	DMA18	DMA20	DMA24	DMA28	DMA31	$\overline{\text{DMS1}}$	NC	DMD38	DMD35	DMD30	DMD28	DMD24	DMD20	DMD15	DMD8	DMD7	C
B	DMA16	DMA17	DMA21	DMA25	DMA26	DMA30	DMPAGE	$\overline{\text{DMS3}}$	DMD39	DMD37	DMD33	DMD32	DMD26	DMD25	DMD21	DMD18	DMD14	DMD12	B
A	$\overline{\text{BR}}$	$\overline{\text{BG}}$	DMA22	EGND	DMA29	EVDD	$\overline{\text{DMS2}}$	EGND	DMD34	EVDD	DMD29	EGND	DMD23	EVDD	DMD19	EGND	DMD16		A
	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	

ADSP-21020

TOP VIEW
(PINS DOWN)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	
U	PMD31	PMD35	PMD39	PMD40	EGND	PMD44	EGND	$\overline{\text{PMWR}}$	EVDD	PMACK	EGND	RCOMP	EVDD	TCK	EGND	TMS	PMA20	PMA17	U
T	PMD27	PMD30	PMD32	PMD37	NC	PMD42	PMD45	$\overline{\text{PMTS}}$	NC	NC	CLKIN	DMACK	$\overline{\text{DMWR}}$	$\overline{\text{TRST}}$	$\overline{\text{PMS1}}$	PMA23	PMA19	EGND	T
S	PMD21	PMD26	PMD28	PMD34	PMD36	PMD41	PMD43	PMD47	$\overline{\text{PMRD}}$	NC	$\overline{\text{DMRD}}$	$\overline{\text{DMTS}}$	TDI	PMPAGE	PMA22	PMA18	PMA14	PMA11	S
R	EGND	PMD23	PMD25	PMD29	PMD33	PMD38	IGND	PMD46	IVDD	$\overline{\text{RESET}}$	IGND	TDO	$\overline{\text{PMS0}}$	PMA21	PMA16	PMA15	PMA10	EGND	R
P	PMD18	PMD19	PMD22	PMD24											PMA12	PMA13	PMA9	PMA8	P
N	EVDD	PMD16	PMD17	PMD20											PMA7	PMA6	PMA5	EVDD	N
M	PMD12	PMD13	PMD14	PMD15											PMA2	PMA3	PMA4	PMA1	M
L	EGND	PMD11	PMD10	IGND											IGND	TIMEXP	PMA0	EGND	L
K	PMD9	PMD8	PMD7	PMD6											$\overline{\text{IRQ3}}$	$\overline{\text{IRQ2}}$	NC	EVDD	K
J	EVDD	PMD5	PMD2	IVDD											IVDD	$\overline{\text{IRQ1}}$	$\overline{\text{IRQ0}}$	EVDD	J
H	PMD4	PMD3	DMD0	DMD1											FLAG1	FLAG0	FLAG2	EGND	H
G	EGND	NC	DMD3	IGND											IGND	DMA0	DMA1	FLAG3	G
F	PMD1	PMD0	DMD6	DMD9											DMA5	DMA4	DMA3	DMA2	F
E	EGND	DMD2	DMD10	DMD13											DMA10	DMA8	DMA7	DMA6	E
D	DMD4	DMD5	DMD11	DMD17	DMD22	DMD27	DMD31	DMD36	IVDD	$\overline{\text{DMS0}}$	IGND	DMA27	DMA23	DMA19	DMA15	DMA12	DMA11	DMA9	D
C	DMD7	DMD8	DMD15	DMD20	DMD24	DMD28	DMD30	DMD35	DMD38	NC	$\overline{\text{DMS1}}$	DMA31	DMA28	DMA24	DMA20	DMA18	DMA14	DMA13	C
B	DMD12	DMD14	DMD18	DMD21	DMD25	DMD26	DMD32	DMD33	DMD37	DMD39	$\overline{\text{DMS3}}$	DMPAGE	DMA30	DMA26	DMA25	DMA21	DMA17	DMA16	B
A		DMD16	EGND	DMD19	EVDD	DMD23	EGND	DMD29	EVDD	DMD34	EGND	$\overline{\text{DMS2}}$	EVDD	DMA29	EGND	DMA22	$\overline{\text{BG}}$	$\overline{\text{BR}}$	A
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	

ADSP-21020

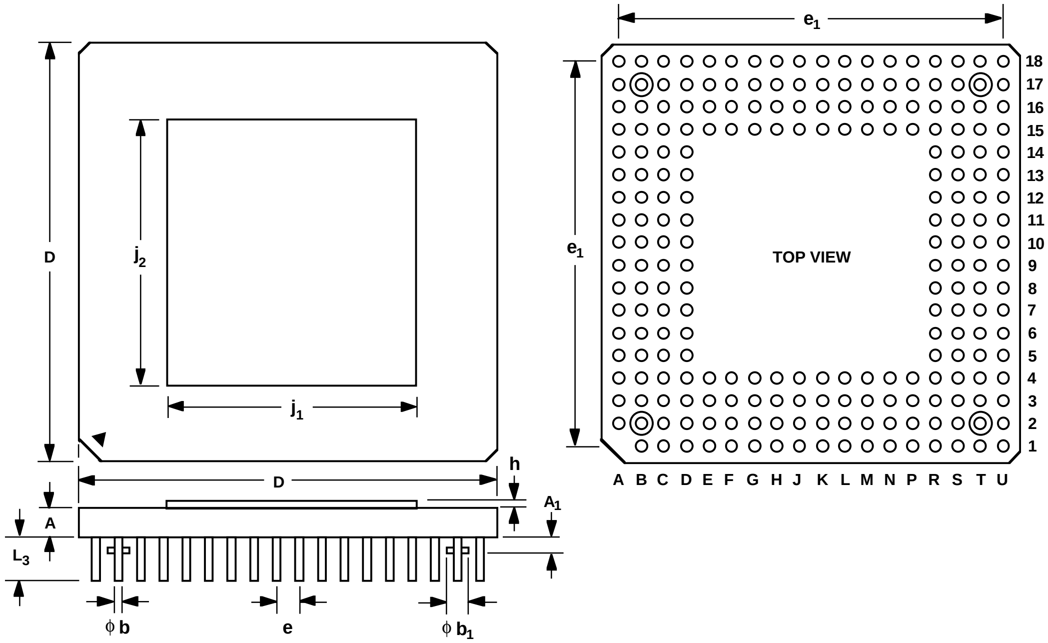
BOTTOM VIEW
(PINS UP)

PGA LOCATION	PIN NAME	PGA LOCATION	PIN NAME	PGA LOCATION	PIN NAME	PGA LOCATION	PIN NAME
G16	DMA0	B5	DMD25	K1	PMD9	L16	TIMEXP
G17	DMA1	B6	DMD26	L3	PMD10	U12	RCOMP
F18	DMA2	D6	DMD27	L2	PMD11	T11	CLKIN
F17	DMA3	C6	DMD28	M1	PMD12	T14	TRST
F16	DMA4	A8	DMD29	M2	PMD13	R12	TD0
F15	DMA5	C7	DMD30	M3	PMD14	S13	TDI
E18	DMA6	D7	DMD31	M4	PMD15	U16	TMS
E17	DMA7	B7	DMD32	N2	PMD16	U14	TCK
E16	DMA8	B8	DMD33	N3	PMD17	H18	EGND
D18	DMA9	A10	DMD34	P1	PMD18	A3	EGND
E15	DMA10	C8	DMD35	P2	PMD19	A7	EGND
D17	DMA11	D8	DMD36	N4	PMD20	A11	EGND
D16	DMA12	B9	DMD37	S1	PMD21	A15	EGND
C18	DMA13	C9	DMD38	P3	PMD22	E1	EGND
C17	DMA14	B10	DMD39	R2	PMD23	G1	EGND
D15	DMA15	D10	$\overline{\text{DMS0}}$	P4	PMD24	L1	EGND
B18	DMA16	C11	$\overline{\text{DMS1}}$	R3	PMD25	L18	EGND
B17	DMA17	A12	$\overline{\text{DMS2}}$	S2	PMD26	R1	EGND
C16	DMA18	B11	DMS3	T1	PMD27	R18	EGND
D14	DMA19	T13	$\overline{\text{DMWR}}$	S3	PMD28	T18	EGND
C15	DMA20	S11	$\overline{\text{DMRD}}$	R4	PMD29	U5	EGND
B16	DMA21	B12	DMPAGE	T2	PMD30	U7	EGND
A16	DMA22	S12	$\overline{\text{DMTS}}$	U1	PMD31	U11	EGND
D13	DMA23	T12	DMACK	T3	PMD32	U15	EGND
C14	DMA24	L17	PMA0	R5	PMD33	D11	IGND
B15	DMA25	M18	PMA1	S4	PMD34	G4	IGND
B14	DMA26	M15	PMA2	U2	PMD35	G15	IGND
D12	DMA27	M16	PMA3	S5	PMD36	L4	IGND
C13	DMA28	M17	PMA4	T4	PMD37	L15	IGND
A14	DMA29	N17	PMA5	R6	PMD38	R7	IGND
B13	DMA30	N16	PMA6	U3	PMD39	R11	IGND
C12	DMA31	N15	PMA7	U4	PMD40	A5	EVDD
H3	DMD0	P18	PMA8	S6	PMD41	A9	EVDD
H4	DMD1	P17	PMA9	T6	PMD42	A13	EVDD
E2	DMD2	R17	PMA10	S7	PMD43	J1	EVDD
G3	DMD3	S18	PMA11	U6	PMD44	J18	EVDD
D1	DMD4	P15	PMA12	T7	PMD45	N1	EVDD
D2	DMD5	P16	PMA13	R8	PMD46	N18	EVDD
F3	DMD6	S17	PMA14	S8	PMD47	U9	EVDD
C1	DMD7	R16	PMA15	R13	$\overline{\text{PMS0}}$	U13	EVDD
C2	DMD8	R15	PMA16	T15	$\overline{\text{PMS1}}$	K18	EVDD
F4	DMD9	U18	PMA17	U8	$\overline{\text{PMWR}}$	D9	IVDD
E3	DMD10	S16	PMA18	S9	$\overline{\text{PMRD}}$	J4	IVDD
D3	DMD11	T17	PMA19	S14	PMPAGE	J15	IVDD
B1	DMD12	U17	PMA20	T8	$\overline{\text{PMTS}}$	R9	IVDD
E4	DMD13	R14	PMA21	U10	PMACK	C10	NC
B2	DMD14	S15	PMA22	A17	$\overline{\text{BG}}$	S10	NC
C3	DMD15	T16	PMA23	A18	$\overline{\text{BR}}$	T10	NC
A2	DMD16	F2	PMD0	H16	FLAG0	T9	NC
D4	DMD17	F1	PMD1	H15	FLAG1	K17	NC
B3	DMD18	J3	PMD2	H17	FLAG2	T5	NC
A4	DMDI9	H2	PMD3	G18	FLAG3	G2	NC
C4	DMD20	H1	PMD4	J17	$\overline{\text{IRQ0}}$		
B4	DMD21	J2	PMD5	J16	$\overline{\text{IRQ1}}$		
D5	DMD22	K4	PMD6	K16	$\overline{\text{IRQ2}}$		
A6	DMD23	K3	PMD7	K15	$\overline{\text{IRQ3}}$		
C5	DMD24	K2	PMD8	R10	$\overline{\text{RESET}}$		

OUTLINE DIMENSIONS

Dimensions shown in inches and (mm).

223-Pin Ceramic Pin Grid Array



	INCHES		MILLIMETERS	
SYMBOL	MIN	MAX	MIN	MAX
A	0.084	0.102	2.11	2.59
A_1	0.40	0.60	1.02	1.52
ϕb	0.018 TYP		0.46 TYP	
ϕb_1	0.050 TYP		1.27 TYP	
D	1.844	1.876	46.84	47.64
e_1	1.700 TYP		43.18 TYP	
e	0.100 TYP		2.54 TYP	
L_3	0.172	0.188	4.37	4.77
h	0.020 TYP		0.500 TYP	
j_1	1.125	1.147	28.56	29.14
j_2	1.065	1.186	27.05	27.61

NOTE
When socketing the CPGA package, use of a low insertion force socket is recommended.

ORDERING GUIDE

Part Number*	Ambient Temperature Range	Instruction Rate (MHz)	Cycle Time (ns)	Package
ADSP-21020KG-80	0°C to +70°C	20	50	223-Lead Ceramic Pin Grid Array
ADSP-21020KG-100	0°C to +70°C	25	40	223-Lead Ceramic Pin Grid Array
ADSP-21020KG-133	0°C to +70°C	33.3	30	223-Lead Ceramic Pin Grid Array
ADSP-21020BG-80	–40°C to +85°C	20	50	223-Lead Ceramic Pin Grid Array
ADSP-21020BG-100	–40°C to +85°C	25	40	223-Lead Ceramic Pin Grid Array
ADSP-21020BG-120	–40°C to +85°C	30	33.3	223-Lead Ceramic Pin Grid Array
ADSP-21020TG-80	–55°C to +125°C	20	50	223-Lead Ceramic Pin Grid Array
ADSP-21020TG-100	–55°C to +125°C	25	40	223-Lead Ceramic Pin Grid Array
ADSP-21020TG-120	–55°C to +125°C	30	33.3	223-Lead Ceramic Pin Grid Array
ADSP-21020TG-80/883B	–55°C to +125°C	20	50	223-Lead Ceramic Pin Grid Array
ADSP-21020TG-100/883B	–55°C to +125°C	25	40	223-Lead Ceramic Pin Grid Array
ADSP-21020TG-120/883B	–55°C to +125°C	30	33.3	223-Lead Ceramic Pin Grid Array

*G = Ceramic Pin Grid Array.

C1601c–5–8/94

PRINTED IN U.S.A.