

1.1 Scope.

This specification covers the detail requirements for a CMOS monolithic 32-bit and 64-bit IEEE Standard 754 format floating-point arithmetic and logic unit (ALU).

1.2 Part Number.

The complete part number per Table 1 of this specification is as follows:

Device	Part Number
-1	ADSP-3220SG/883B
-2	ADSP-3220TG/883B

1.2.3 Case Outline.

See Appendix 1 of General Specification ADI-M-1000: package outline: G-144A.

1.3 Absolute Maximum Ratings.

Supply Voltage	-0.3V to 7V
Input Voltage	-0.3V to V_{DD}
Output Voltage	-0.3V to V_{DD}
Operating Temperature Range (Ambient)	-55°C to +125°C
Storage Temperature Range	-65°C to +150°C
Lead Temperature (Soldering 10sec)	+300°C

1.5 Thermal Characteristics.

Maximum Thermal Resistance θ_{JC} : see MIL-M-38510, Appendix C.

ADSP-3220 – SPECIFICATIONS

Parameter	Symbol	Device	Design Limit @ +25°C	Sub Group 1	Sub Group 2, 3	Sub Group 9	Sub Group 10, 11	Test Condition ¹	Units
Digital Input High Voltage	V _{IH}	- 1, 2	2.0	2.0	2.0			V _{DD} = max	V min
Digital Input High Voltage, CLK and Asynchronous Controls (RESET, MSWSEL, OEN & IPORT0:1)	V _{IHA}	- 1, 2	3.0	3.0	3.0			V _{DD} = max	V max
Digital Input Low Voltage	V _{IL}	- 1, 2	0.8	0.8	0.8			V _{DD} = min	V max
Digital Output High Voltage	V _{OH}	- 1, 2	2.4	2.4	2.4			V _{DD} = min I _{OH} = - 1mA	V min
Digital Output Low Voltage*	V _{OL}	- 1, 2	0.4	0.6	0.6			V _{DD} = min I _{OL} = + 4mA	V max
Digital Input High Current	I _{IH}	- 1, 2	10	10	10			V _{DD} = max V _{IN} = + 5.0V	μA max
Digital Input Low Current	I _{IL}	- 1, 2	10	10	10			V _{DD} = max V _{IN} = 0.0V	μA max
Three-State Leakage Current	I _{OZH}	- 1, 2	50	50	50			V _{DD} = max High Z, V _{IN} = 0V or max	μA max
	I _{OZL}		50	50	50				
Supply Current*	I _{DD1}	- 1, 2	150	200	200			@ max Clock Rate, TTL Inputs All V _{IN} = 2.4V	mA max
	I _{DD2}	- 1, 2	50	60	60				
Clock Cycle	t _{CY}	- 1	125			150	150	Note 2	ns max
		- 2	100			125	125		
Clock LO	t _{CL}	- 1, 2	20			30	30	Note 2	ns min
Clock HI	t _{CH}	- 1, 2	20			30	30	Note 2	ns min
Data & Control Setup	t _{DS}	- 1	20			25	25	Note 2	ns min
		- 2	15			20	20		
Data & Control Hold	t _{DH}	- 1, 2	3			3	3	Note 2	ns min
Data Output Delay	t _{DO}	- 1	30			35	35	Note 2	ns max
		- 2	25			30	30		
Status Output Delay	t _{SO}	- 1	30			35	35	Note 2	ns max
		- 2	25			30	30		
MSWSEL-to-Data Delay	t _{ENO}	- 1	25			30	30	Note 2	ns max
		- 2	20			25	25		
Three-State Disable Delay*	t _{DIS}	- 1	18			25	25	Notes 2 & 3	ns max
		- 2	15			20	20		
Three-State Enable Delay	t _{ENA}	- 1	25			30	30	Notes 2, 3 & 4	ns max
		- 2	20			25	25		
RESET Setup*	t _{SU}	- 1, 2	25			25	25	Note 2	ns min
RESET Pulse Duration*	t _{RS}	- 1, 2	75			75	75	Note 2	ns min
Operation Time (With or Without Direct Operand Feed): 32-Bit ALU Operation	t _{OPD}	- 1	125			150	150	Note 2	ns max
		- 2	100			125	125		
		- 1	125			150	150		
		- 2	100			125	125		
64-Bit ALU Operation									

Table 1. (Continued on next page)

Parameter	Symbol	Device	Design Limit @ +25°C	Sub Group 1	Sub Group 2,3	Sub Group 9	Sub Group 10, 11	Test Condition ¹	Units
Hold Setup	t_{HS}	- 1	20			22	22	Note 2	ns min
		- 2	15			18	18		
Hold Hold	t_{HH}	- 1, 2	3			3	3	Note 2	ns min
Total Latency ⁵ (With Direct Operand Feed): 32-Bit ALU Operation	t_{LAD}	- 1	300			360	360	Note 2	ns max
		- 2	240			300	300		
64-Bit ALU Operation		- 1	363			435	435		
		- 2	290			363	363		

NOTES

*Indicates that a limit for this parameter has changed from REV. A.

¹ $T_A = +25^\circ\text{C}$; $V_{DD} = +4.5\text{V min to } +5.5\text{V max}$ (unless otherwise noted).

²Input levels are GND and +3.0V; $V_{DD} = +4.5\text{V}$, and timing transitions per Figures 1 through 8, measured at +1.5V.

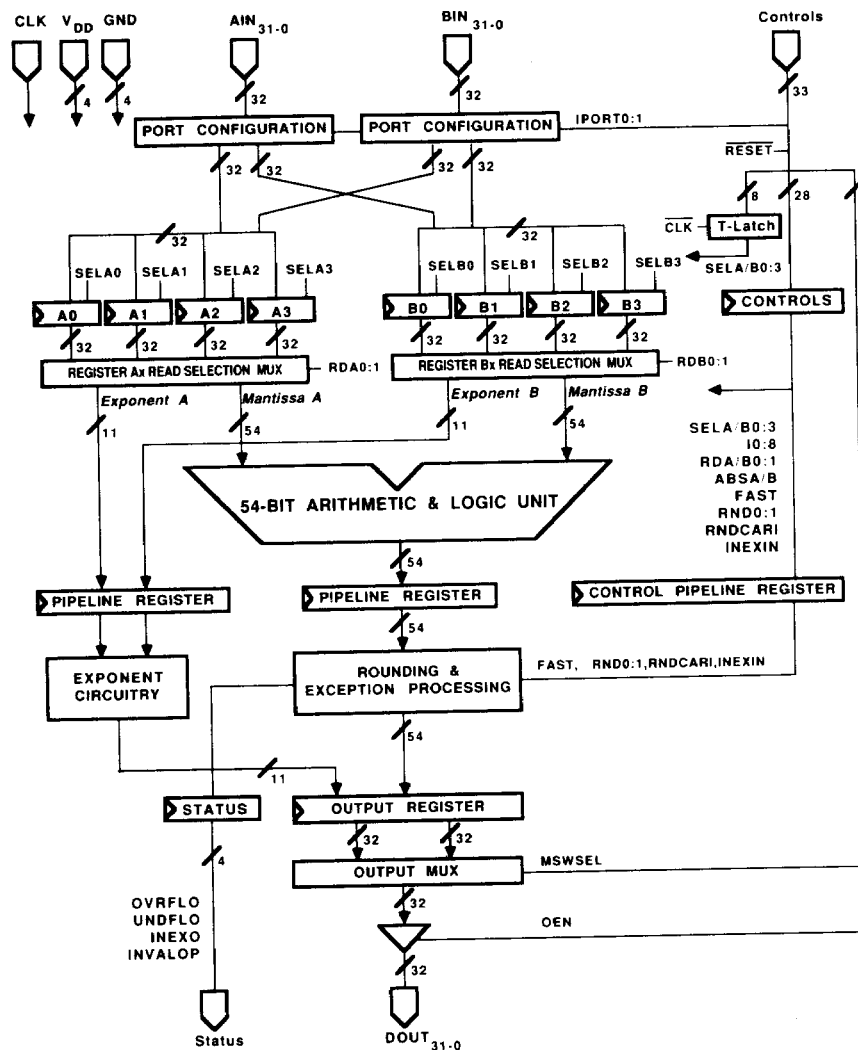
³Transitions measured per Figure 1.

⁴3ns minimum.

⁵Total latency = (data setup + processing + output delay of MSW) in Direct Operand Feed Mode.

Table 1.

3.2.1 Functional Block Diagram and Terminal Assignments.



Pin Assignments

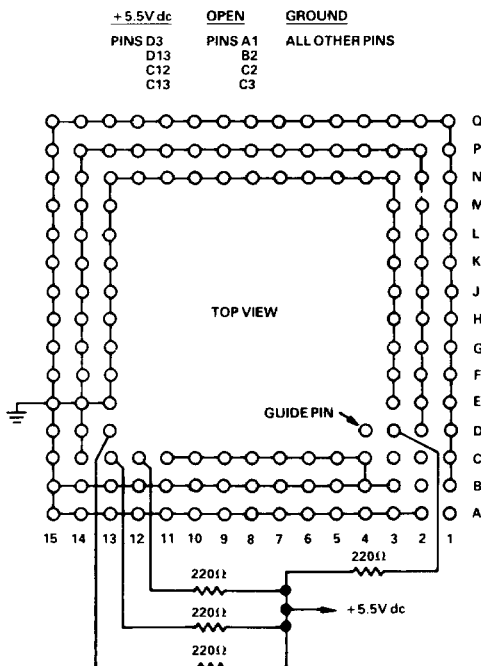
PIN	FUNCTION	PIN	FUNCTION
B2	DIN3	M12	DOUT28
B1	DIN2	M13	DOUT29
C2	DIN1	L12	DOUT30
C1	DIN0	L13	DOUT31
D2	WRAPB	K12	GND
D1	RDB0	K13	GND
E2	SELB0	J12	GND
E1	SELB1	J13	DENORM
F3	ABSB	H11	INVALOP
F2	DP	H12	OVRFLO
F1	SP	H13	UNDFLO
G2	CLK	G12	OEN
G3	RESET	G11	MSWSEL
G1	RND1	G13	SHLP
H1	RND0	F13	FAST
H2	RNDCARO	F12	ABSA
H3	VDD	F11	SELA1
J1	VDD	E13	SELA0
J2	VDD	E12	RDA0
K1	INEXO	D13	WRAPA
K2	DOUT0	D12	DIN31
L1	DOUT1	C13	DIN30
M1	DOUT2	B13	DIN29
L2	DOUT3	C12	DIN28
N1	N/C	A13	N/C
M2	DOUT4	B12	DIN27
N2	DOUT5	A12	DIN26
M3	DOUT6	B11	DIN25
N3	DOUT7	A11	DIN24
M4	DOUT8	B10	DIN23
N4	DOUT9	A10	DIN22
M5	DOUT10	B9	DIN21
N5	DOUT11	A9	DIN20
L6	DOUT12	C8	DIN19
M6	DOUT13	B8	DIN18
N6	DOUT14	A8	DIN17
M7	DOUT15	B7	DIN16
L7	DOUT16	C7	DIN15
N7	DOUT17	A7	DIN14
N8	DOUT18	A6	DIN13
M8	DOUT19	B6	DIN12
L8	DOUT20	C6	DIN11
N9	DOUT21	A5	DIN10
M9	DOUT22	B5	DIN9
N10	DOUT23	A4	DIN8
M10	DOUT24	B4	DIN7
N11	DOUT25	A3	DIN6
N12	DOUT26	A2	DIN5
M11	DOUT27	B3	DIN4
N13	N/C	A1	N/C

3.2.4 Microcircuit Technology Group.

This microcircuit is covered by technology group (105).

4.2.1 Life Test/Burn-In Circuit.

Steady state life test is per MIL-STD-883 Method 1005. Burn-in is per MIL-STD-883 Method 1015 test condition (B).



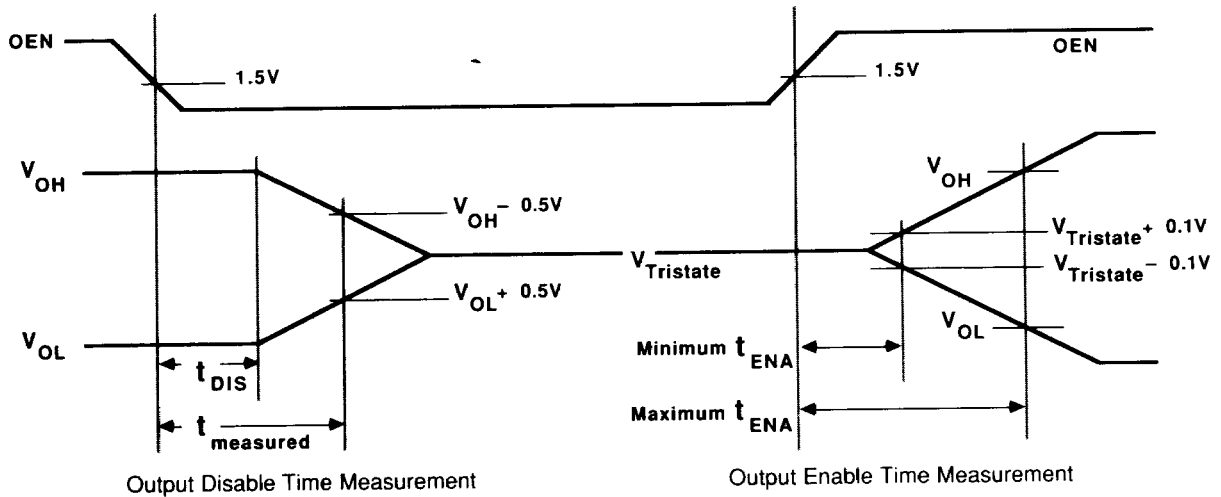


Figure 1. ADSP-3220 Three-State Disable and Enable Timing

Output disable time, t_{DIS} , is measured from the time the output enable control signal reaches 1.5 V to the time when all outputs have ceased driving. This is calculated by measuring the time, $t_{MEASURED}$, from the same starting point to when the output voltages have changed by 0.5 V toward +1.5 V. From the tester capacitive loading, C_L , and the measured current, i_L , the decay time, t_{DECAy} , can be approximated to first order by:

$$t_{DECAy} = \frac{C_L \cdot 0.5 \text{ V}}{i_L}$$

from which

$$t_{DIS} = t_{MEASURED} - t_{DECAy}$$

is calculated. Disable times are longest at the highest specified temperature.

The minimum output enable time, minimum t_{ENA} , is the earliest that outputs begin to drive. It is measured from the control signal OEN reaching 1.5 V to the point at which the fastest outputs have changed by 0.1 V from $V_{TRISTATE}$ toward their final output voltages. Minimum enable times are shortest at the lowest specified temperature.

The maximum output enable time, maximum t_{ENA} , is also measured from output enable control signal at 1.5 V to the time when all outputs have reached TTL input levels (V_{OH} or V_{OL}). This could also be considered as “data valid.” Maximum enable times are longest at the highest specified temperature.

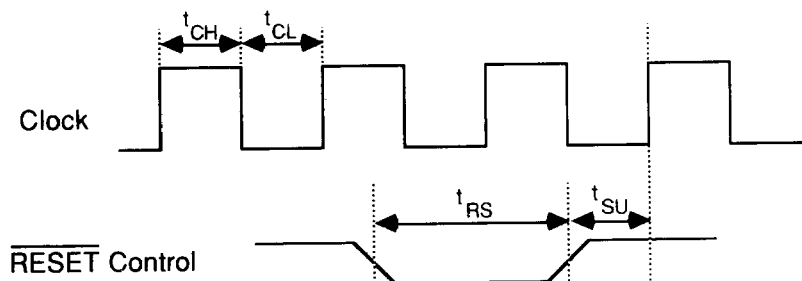
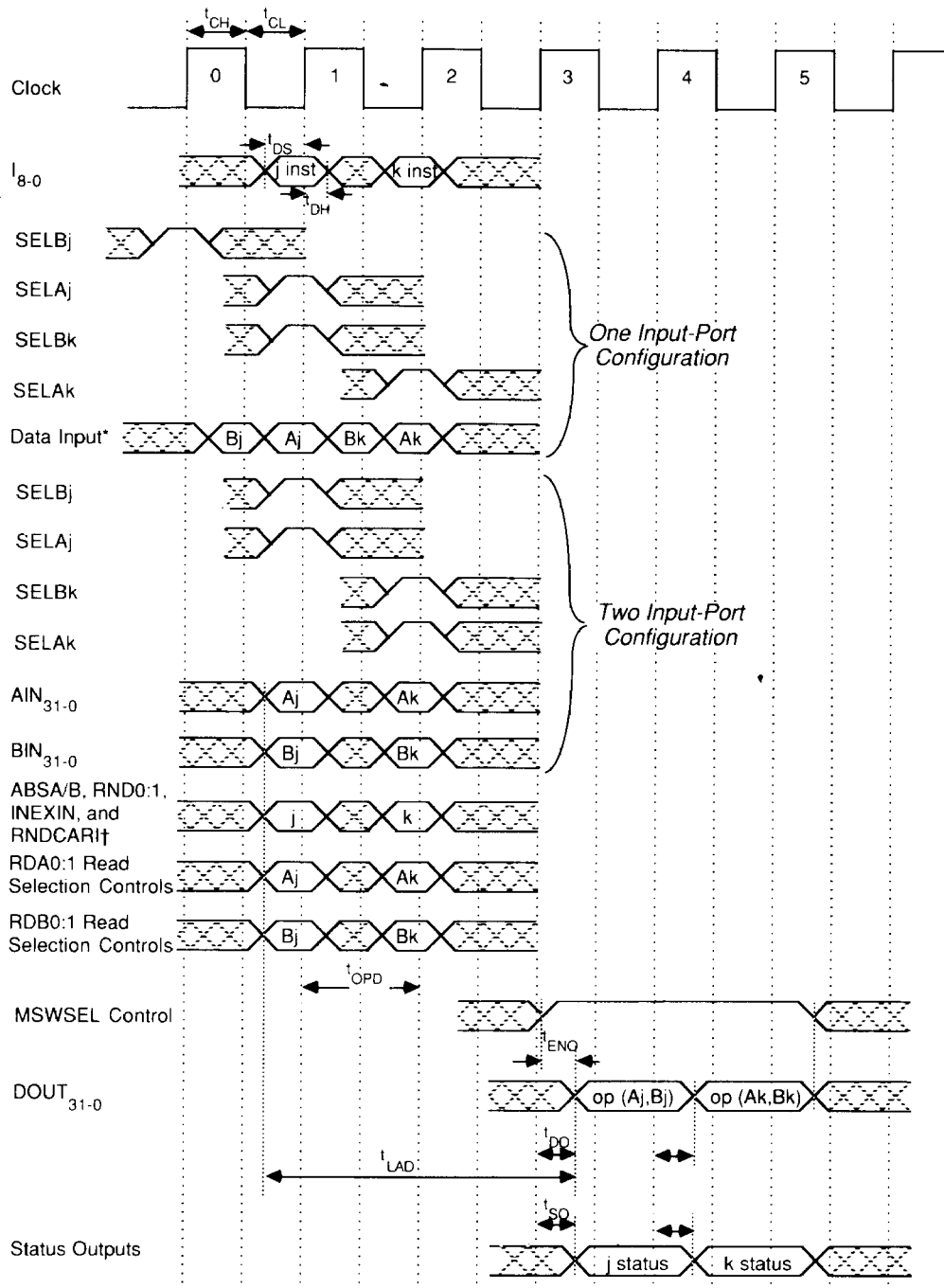


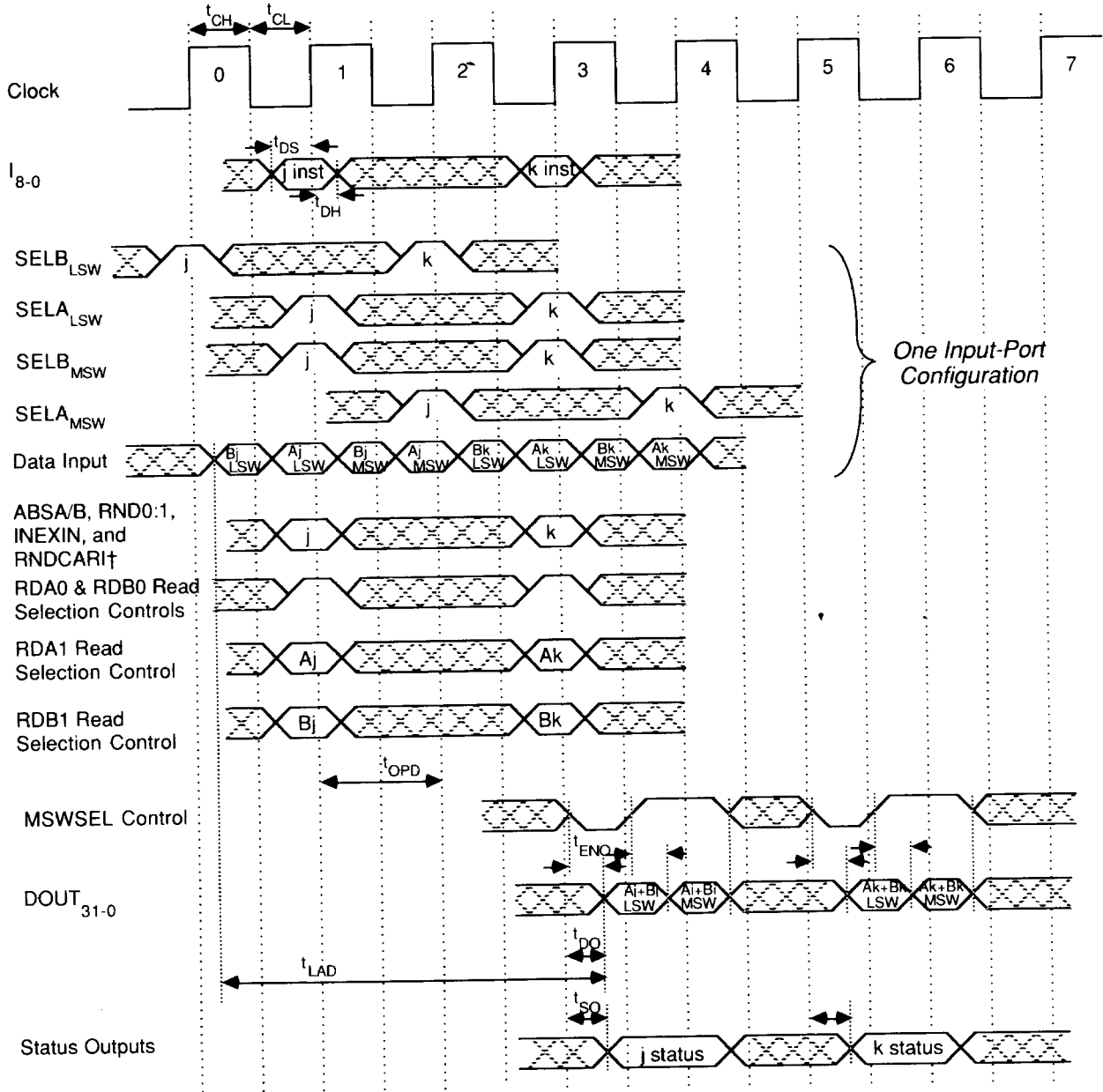
Figure 2. ADSP-3220 Reset Timing

ADSP-3220



† RNDCAI and INEXIN should be LO except for unwrap, division, and square root operations.

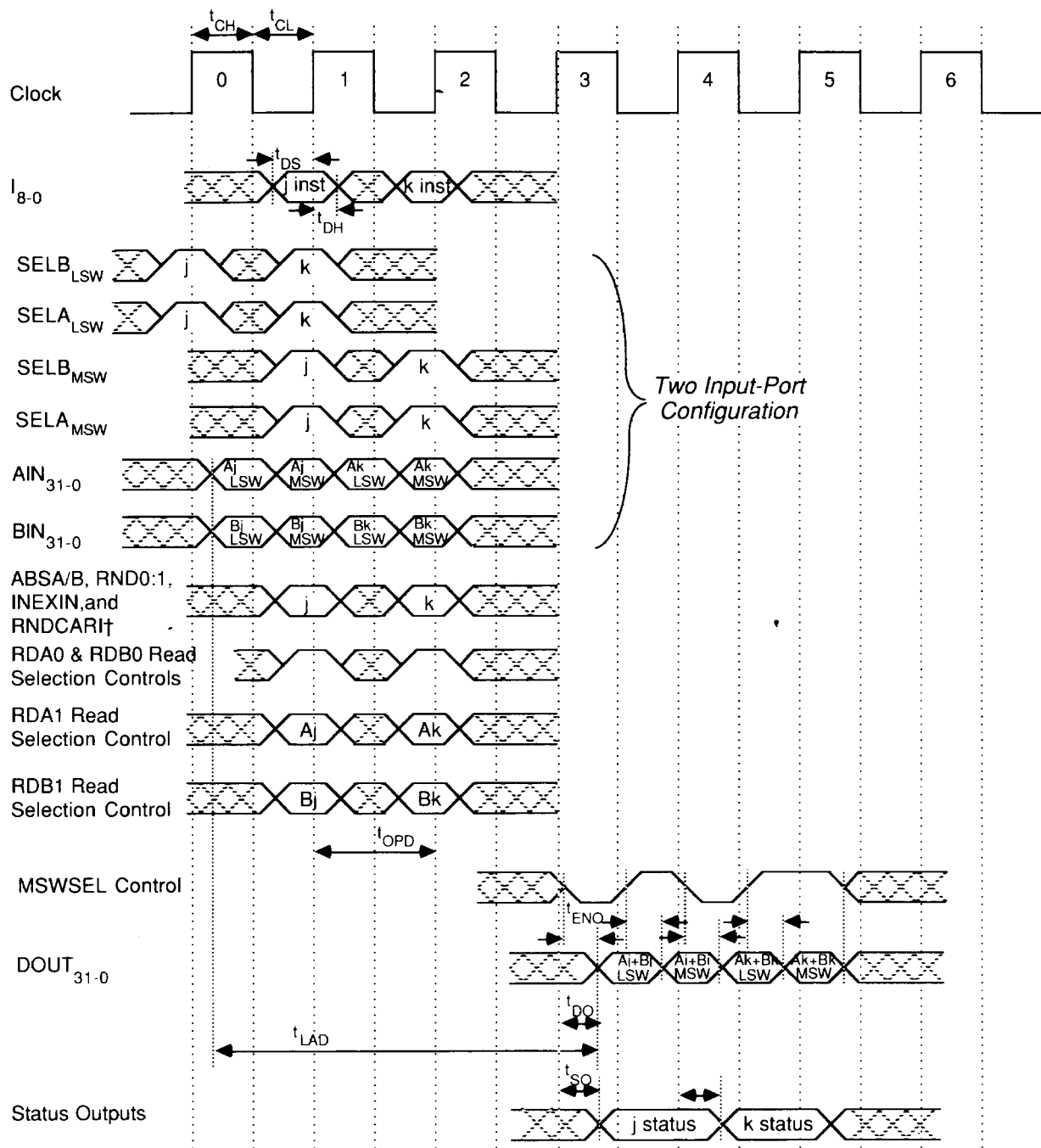
Figure 3. ADSP-3220 32-Bit Single-Precision Floating-Point Logical, and Fixed-Point ALU Operations



† RNDCA1 and INEXIN should be LO except for unwrap, division, and square root operations.

Figure 4. ADSP-3220 64-Bit Double-Precision Floating-Point ALU Operations – One-Port Configuration

ADSP-3220



† RNDCAI and INEXIN should be LO except for unwrap, division, and square root operations.

Figure 5. ADSP-3220 64-Bit Double-Precision Floating-Point ALU Operations – Two-Port Configuration

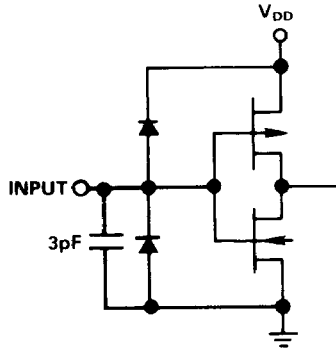


Figure 6. Equivalent Input Circuits

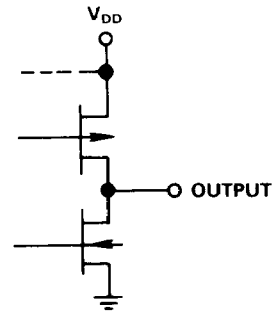


Figure 7. Equivalent Output Circuits

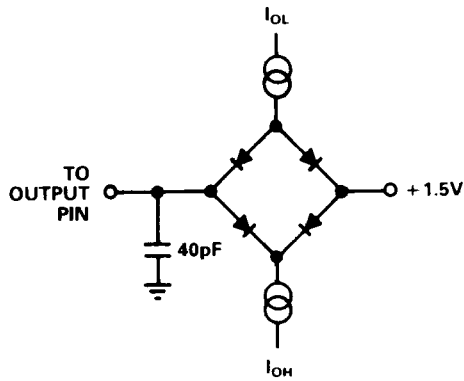


Figure 8. Normal Load for AC Measurements

ADSP-3220

5.0 ADSP-3220 Floating Point ALU Instructions

For all operations involving two operands, the operands may be in SP Floating point format, DP Floating point, or 32-bit FIXED point, but both operands must be of the same data type.

Note that for all arithmetic operations (SP, DP, and FIXED), the ABSA and ABSB controls can be used with the input operands, causing the absolute value of the respective operand to be used in the operation.

ADSP-3220 Fixed Point Arithmetic Operations

INSTRUCTION	OPCODE										COMMENT
	I ₈	I ₇	I ₆	I ₅	I ₄	I ₃	I ₂	I ₁	I ₀		
IADD	0	0	1	0	0	0	0	1	1	Integer Add (A + B)	
ISUBA	0	0	1	0	0	0	1	1	1	Integer Subtract (B - A)	
ISUBB	0	0	1	0	0	1	0	1	1	Integer Subtract (A - B)	
IADDWC	0	0	1	0	1	0	0	1	1	Integer Add With Carry (A + B) ¹	
ISUBWBA	0	0	1	0	1	0	1	1	1	Integer Subtract With Borrow (B - A) ¹	
ISUBWBB	0	0	1	0	1	1	0	1	1	Integer Subtract With Borrow (A - B) ¹	
INEGA	0	0	1	0	0	0	1	0	1	Perform Twos Complement Negation of the A Operand	
INEGB	0	0	1	0	0	1	0	1	0	Perform Twos Complement Negation of the B Operand	
IADDAS	0	0	1	1	0	0	0	1	1	Absolute Value of (A + B)	
ISUBAAS	0	0	1	1	0	0	1	1	1	Absolute Value of (B - A)	
ISUBBAS	0	0	1	1	0	1	0	1	1	Absolute Value of (A - B)	

ADSP-3220 Logical Operations

INSTRUCTION	OPCODE										COMMENT
	I ₈	I ₇	I ₆	I ₅	I ₄	I ₃	I ₂	I ₁	I ₀		
COMPLA	0	0	0	0	0	0	1	0	1	Ones Complement the A Operand	
COMPLB	0	0	0	0	0	1	0	1	0	Ones Complement the B Operand	
PASS A	0	0	0	0	0	0	0	0	1	Passes the A Operand (selected with RDA1, RDA0) unmodified to the output.	
PASS B	0	0	0	0	0	0	0	1	0	Passes the B Operand (selected with RDB1, RDB0) unmodified to the output.	
A AND B	0	0	0	0	1	0	0	1	0	Bitwise AND	
A OR B	0	0	0	1	0	0	0	1	0	Bitwise OR	
A XOR B	0	0	0	1	1	0	0	1	0	Bitwise Exclusive OR	
NOP	0	0	0	0	0	0	0	0	0	No Operation. Preserves status flags but does not preserve the contents of the output register.	
CLR	1	0	0	0	0	0	0	0	0	Clear. Sets the UNDFLO, OVERFLO, INVALOP, and INEXO flags to zero. The Output Register result is not meaningful.	

NOTE

¹The Carry bit in fixed point arithmetic is latched internally, for one-cycle only, to act as a Carry-In pin for the operations. Add with Carry or Subtract with Borrow. These instructions are used in performing 64-bit addition or subtraction. See note on Special Instructions in the commercial data sheet.

Table 2. ADSP-3220 Instruction Set

ADSP-3220 Single-Precision Floating Point Operations

INSTRUCTION	OPCODE										COMMENT
	I ₈	I ₇	I ₆	I ₅	I ₄	I ₃	I ₂	I ₁	I ₀		
SADD	1	1	1	0	0	0	0	1	1	Floating Point SP Add (A + B)	
SSUBB	1	1	1	0	0	0	1	1	1	Floating SP Subtract (A - B)	
SSUBA	1	1	1	0	0	1	0	1	1	Floating SP Subtract (B - A)	
SCOMP	1	1	1	0	0	1	1	1	1	SP Compare. Operation is (A - B); Output Result is (A - B); Flags OVRFLO and UNDFLO generated conditional on Greater Than or Less Than conditions, respectively. If equal, output is zero and UNDFLO and OVRFLO are both LOW.	
SADDAS	0	1	1	0	0	0	0	1	1	SP Absolute Value of (A + B)	
SSUBBAS	0	1	1	0	0	0	1	1	1	SP Absolute Value of (A - B)	
SSUBAAS	0	1	1	0	0	1	0	1	1	SP Absolute Value of (B - A)	
FIX A	0	1	1	0	0	1	1	0	1	Convert 32-bit Single Precision Floating Point A operand → 32-bit Integer	
FIX B	0	1	1	0	0	1	1	1	0	Convert 32-bit Single Precision Floating Point B operand → 32-bit Integer	
FLOATA	0	1	1	1	0	0	1	0	1	Convert 32-bit FIXED A operand → 32-bit Single Precision Floating Point	
FLOATB	0	1	1	1	0	0	1	1	0	Convert 32-bit FIXED B operand → 32-bit Single Precision Floating Point	
DOUBLEA	0	1	1	1	0	1	1	0	1	Convert 32-bit Single Precision A operand → 64-bit Double Precision Floating ¹	
DOUBLEB	0	1	1	1	0	1	1	1	0	Convert 32-bit Single Precision B operand → 64-bit Double Precision Floating ¹	
SPASSA	0	1	1	1	1	0	0	0	1	Pass SP A operand. A NAN is passed unmodified but causes the INVALOP exception	
SPASSB	0	1	1	1	1	0	0	1	0	Pass SP B operand. A NAN is passed unmodified but causes the INVALOP exception	
SWRAPA	0	1	1	1	0	0	0	0	1	Converts SP denormalized A operand → Wrapped format ²	
SWRAPB	0	1	1	1	0	0	0	1	0	Converts SP denormalized B operand → Wrapped format ²	
SUNWRAPA	0	1	1	0	1	0	0	0	1	Converts Wrapped A operand → SP denormalized format ²	
SUNWRAPB	0	1	1	0	1	0	0	1	0	Converts Wrapped B operand → SP denormalized format ²	
SSIGN ³	0	1	1	1	1	1	1	0	1	Transfers sign of the B operand to the A operand. Result is Sign B, Expt A, Fraction A.	
SXSUB ³	0	1	1	1	1	1	0	0	1	Exponent subtract. Subtracts the B exponent from the A exponent. Result is Sign A, (Expt A - Expt B), Fraction A. NANs and Infinities are operated on directly without flagging the INVALOP exception. If the Expt of the Result underflows to -127 or less, then the Output Result returned is Zero. If the Expt of the Result overflows to +128, then the Output Result returned is Infinity.	
SITRN ³	0	1	1	0	1	0	1	0	1	Integer Shift and Truncation. The A mantissa with hidden bit restored (24 bits) is treated as a right-justified unsigned integer and shifted down logically by the unbiased B exponent. ⁴ The 32-bit right-justified result appears in the MSW and is a left-zero-filled unsigned-magnitude integer. The shift value given by the unbiased B exponent is interpreted as an unsigned number; negative unbiased exponents cause very large down shifts.	

NOTES

¹Conversion from FIXED to Double format are always exact. The 32-bit FIXED source operand must be fed from one of the registers A0, A2, B0, or B2 only.

²For information on handling denormals, wrapped, and unwrapped numbers, see Application Note on "Handling IEEE Exceptions."

³An unbiased exponent is the exponent after the proper exponent bias has been subtracted.

Table 2. ADSP-3220 Instruction Set (Continued)

ADSP-3220

ADSP-3220 Double-Precision Floating Point Operations

INSTRUCTION	OPCODE										COMMENT
	I ₈	I ₇	I ₆	I ₅	I ₄	I ₃	I ₂	I ₁	I ₀		
DADD	1	1	0	0	0	0	0	1	1	DP Add (A + B)	
DSUBB	1	1	0	0	0	0	1	1	1	DP Subtract (A - B)	
DSUBA	1	1	0	0	0	1	0	1	1	DP Subtract (B - A)	
DCOMP	1	1	0	0	0	1	1	1	1	DP Compare. Operation is (A - B); Output Result is (A - B); Flags OVRFLO and UNDFLO generated conditional on Greater Than or Less Than conditions, respectively. If equal, output is zero and UNFLO and OVRFLO are both LOW.	
DADDAS	0	1	0	0	0	0	0	1	1	DP Absolute Value of (A + B)	
DSUBBAS	0	1	0	0	0	0	1	1	1	DP Absolute Value of (A - B)	
DSUBAAS	0	1	0	0	0	1	0	1	1	DP Absolute Value of (B - A)	
DFIX A	0	1	0	0	1	1	1	0	1	Convert 64-bit Double Precision A operand → 32-bit Integer	
DFIX B	0	1	0	0	1	1	1	1	0	Convert 64-bit Double Precision B operand → 32-bit Integer	
DFLOAT A ¹	0	1	0	1	0	0	1	0	1	Convert 32-bit FIXED A operand → 64-bit Double Precision Floating Point	
DFLOAT B ¹	0	1	0	1	0	0	1	1	0	Convert 32-bit FIXED B operand → 64-bit Double Precision Floating Point	
SINGLE A	1	1	0	0	1	1	1	0	1	Convert 64-bit Double Precision A operand → 32-bit Single Precision	
SINGLE B	1	1	0	0	1	1	1	1	0	Convert 64-bit Double Precision B operand → 32-bit Single Precision	
DPASS A	0	1	0	1	1	0	0	0	1	Pass DP A operand. A NAN is passed unmodified but causes the INVALOP exception.	
DPASS B	0	1	0	1	1	0	0	1	0	Pass DP B operand. A NAN is passed unmodified but causes the INVALOP exception.	
DWRAP A	0	1	0	1	0	0	0	0	1	Converts DP denormalized A operand → Wrapped format ²	
DWRAP B	0	1	0	1	0	0	0	1	0	Converts DP denormalized B operand → Wrapped format ²	
DUNWRAP A	0	1	0	0	1	0	0	0	1	Converts Wrapped A operand → DP denormalized format ²	
DUNWRAP B	0	1	0	0	1	0	0	1	0	Converts Wrapped B operand → DP denormalized format ²	
DSIGN	0	1	0	1	1	1	1	0	1	Transfers sign of the B operand to the A operand. Result is Sign B, Expt A, Fraction A.	
DXSUB	0	1	0	1	1	1	0	0	1	Exponent subtract. Subtracts the B exponent from the A exponent. Result is Sign A, (Expt A - Expt B), Fraction A. NANs and Infinities are operated on directly without flagging the INVALOP exception. If the Expt of the Result underflows to -1023 or less, then the Output Result returned is Zero. If the Expt of the Result overflows to +1024, then the Output Result returned is Infinity.	
DITRN	0	1	0	0	1	0	1	0	1	Integer Shift and Truncation. The A mantissa with hidden bit restored (53 bits) is treated as a right-justified unsigned integer and shifted down logically by the unbiased B exponent. ³ The 32-bit right-justified result appears in the MSW and is a left-zero-filled unsigned-magnitude integer. The shift value given by the unbiased B exponent is interpreted as an unsigned number; negative unbiased exponents cause very large down shifts.	

NOTES

¹Conversion from Single to Double format are always exact, so INEXO will always be LOW for these conversions.

²For information on handling denormals, wrapped, and unwrapped numbers, see Application Note on "Handling IEEE Exceptions."

³See note on Special Instructions in the commercial data sheet.

⁴An unbiased exponent is the exponent after the proper exponent bias has been subtracted.

Table 2. ADSP-3220 Instruction Set (Continued)

REV. B

Notations Used in the Tables Below:

- RN = Round to the Nearest Number -
- RP = Round toward Plus Infinity
- RM = Round toward Minus Infinity
- RZ = Round toward Zero
- NORM = Normalized number
- DNRM = Denormalized number. A Denormalized number is treated as zero internally.
- WNRM = Wrapped number. A wrapped number is a number with a normalized fraction and an exponent that has been decremented through zero to take on a twos complement negative value.
- UNRM = Unnormalized number. An Unnormal is an Underflowed and Wrapped Number. An UNRM can result from a multiplication of 1 or 2 Wrapped Numbers.
- NORM.MAX = Maximum Normalized Number Representable in the Destination Format
- NORM.MIN = Minimum Normalized Number Representable in the Destination Format
- OVF = Overflowed Number
- UNDF = Underflowed Number
- INV = Invalid Operand
- INF = Infinity
- OK = No Exception Status Generated

		B operand									
		ZERO		DNRM		NORM		INF		NAN	
A operand		result	status	result	status	result	status	result	status	result	status
	ZERO		ZERO ²		DNRM		NORM		INF		NAN
DNRM		DNRM		NORM DNRM ZERO		INF,NORM.MAX ¹ NORM DNRM	OVRFLO	INF		NAN	INVALOP
NORM		NORM		INF,NORM.MAX ¹ NORM DNRM	OVRFLO	INF,NORM.MAX ¹ NORM DNRM ZERO	OVRFLO	INF		NAN	INVALOP
INF		INF		INF		INF		INF ³ NAN ³	INVALOP	NAN	INVALOP
NAN		NAN	INVALOP	NAN	INVALOP	NAN	INVALOP	NAN	INVALOP	NAN	INVALOP

1. Either INF or NORM.MAX, depending on rounding mode.
2. $\pm \text{ZERO} \pm \text{ZERO} \Rightarrow \pm \text{ZERO}$
 $\pm \text{ZERO} \mp \text{ZERO} \Rightarrow + \text{ZERO}$ (RN, RZ, RP rounding modes)
 $\pm \text{ZERO} \mp \text{ZERO} \Rightarrow - \text{ZERO}$ (RM rounding mode)
3. $\pm \text{INF} \pm \text{INF} \Rightarrow \pm \text{INF}$
 $\pm \text{INF} \mp \text{INF} \Rightarrow + \text{NAN}$ (RN, RZ, RP rounding modes)
 $\pm \text{INF} \mp \text{INF} \Rightarrow - \text{NAN}$ (RM rounding mode)

Table 3. ADSP-3220 Floating-Point Addition/Subtraction (IEEE Mode)

ADSP-3220

		ZERO		DNRM		NORM		INF		NAN	
A operand	B operand	result	status	result	status	result	status	result	status	result	status
ZERO	ZERO ²	ZERO		ZERO		NORM		INF		NAN	INVALOP
DNRM	ZERO	ZERO		NORM ZERO		INF,NORM.MAX ¹ NORM ZERO	OVRFLO UNDFLO	INF		NAN	INVALOP
NORM	ZERO	ZERO		INF,NORM.MAX ¹ NORM ZERO	OVRFLO UNDFLO	INF,NORM.MAX ¹ NORM ZERO ZERO ⁴	OVRFLO UNDFLO	INF		NAN	INVALOP
INF	INF	INF		INF		INF		INF ³ NAN ³	INVALOP	NAN	INVALOP
NAN	NAN	INVALOP		NAN	INVALOP	NAN	INVALOP	NAN	INVALOP	NAN	INVALOP

1. Either INF or NORM.MAX, depending on rounding mode.
2. $\pm \text{ZERO} \pm \text{ZERO} \Rightarrow \pm \text{ZERO}$
 $\pm \text{ZERO} \mp \text{ZERO} \Rightarrow + \text{ZERO}$ (RN, RZ, RP rounding modes)
 $\pm \text{ZERO} \mp \text{ZERO} \Rightarrow - \text{ZERO}$ (RM rounding mode)
3. $\pm \text{INF} \pm \text{INF} \Rightarrow \pm \text{INF}$
 $\pm \text{INF} \mp \text{INF} \Rightarrow + \text{NAN}$ (RN, RZ, RP rounding modes)
 $\pm \text{INF} \mp \text{INF} \Rightarrow - \text{NAN}$ (RM rounding mode)
4. Exact result.

Table 4. ADSP-3220 Floating-Point Addition/Subtraction (FAST Mode)

Sign	HB	f22	f1	f0	Unbiased Expt	Source Name	Sign	i30	i29	i28	i27	i26	i25	i24	i23	i22	i17	i16	i15	i14	i13	i12	i11	i10	Rounding Modes	Status Flags			
0	1	X	X	X	2**	128	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	all	INVALOP		
0	1	0	0	0	2**	128	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	all	INVALOP		
0	1	0	0	0	2**	31	U*	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	all	OVRFLO		
0	1	1	1	1	2**	30	0	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	all			
0	1	1	1	1	2**	23	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	all			
0	1	0	0	0	2**	23	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	all			
0	1	1	1	1	2**	22	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	RN,RP	INEXO		
0	1	1	1	1	2**	22	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	RZ,RM	INEXO		
0	1	0	0	0	2**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	all			
0	1	1	1	1	2**	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	RN, RP	UNDFLO,INEXO		
0	1	1	1	1	2**	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	RZ,RM	UNDFLO,INEXO		
0	1	0	0	0	2**	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	RN,RP	UNDFLO,INEXO		
0	1	0	0	0	2**	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	RZ,RM	UNDFLO,INEXO	
0	1	0	0	0	2**	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	RP	UNDFLO,INEXO		
0	1	0	0	0	2**	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	RP	UNDFLO,INEXO	
0	1	0	0	0	2**	-126	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	RM,RN,RZ	UNDFLO,INEXO	
0	1	0	0	0	2**	-126	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	RP	UNDFLO,INEXO	
0	0	0	0	0	2**	-126	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	RM,RN,RZ	UNDFLO,INEXO	
0	0	0	0	0	2**	-126	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	RP	UNDFLO,INEXO	
0	0	0	0	0	2**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	RM,RN,RZ	UNDFLO,INEXO	
1	0	0	0	0	2**	-126	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	RM	UNDFLO,INEXO	
1	0	0	0	0	2**	-126	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	RP,RN,RZ	UNDFLO,INEXO
1	1	0	0	0	2**	-126	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	RM	UNDFLO,INEXO	
1	1	0	0	0	2**	-126	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	RP,RN,RZ	UNDFLO,INEXO
1	1	0	0	0	2**	-1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	RM	UNDFLO,INEXO	
1	1	0	0	0	2**	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	RM	UNDFLO,INEXO
1	1	0	0	0	2**	-1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	RM	UNDFLO,INEXO
1	1	0	0	0	2**	-1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	RM	UNDFLO,INEXO
1	1	0	0	0	2**	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	RP,RN,RZ	UNDFLO,INEXO
1	1	0	0	0	2**	-1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	RM	UNDFLO,INEXO
1	1	0	0	0	2**	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	RP,RZ	UNDFLO,INEXO
1	1	0	0	0	2**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	all	
1	1	1	1	1	2**	30	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	all		
1	1	0	0	0	2**	31	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	all	
1	1	0	0	0	2**	31	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	all	
1	1	0	0	0	2**	128	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	all		
1	1	X	X	X	2**	128	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	all	INVALOP	
1	1	X	X	X	2**	128	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	all	INVALOP	

**"U" denotes an undefined result.

Table 5. Conversion of 32-Bit Floating-Point to 32-Bit Twos-Complement Integer

Sign	HB	f23				f19				f0	Unbiased Expnt	Source Name	Sign	i30			Rounding Modes	Status Flags
		f21	f20	f19	f18	f17	f16	f15	f14					i29	i28	i27		
0	1	X	X	X	X	X	X	X	X	2**	1024	+ NAN	0	1	1	1	all	INVALOP
0	1	0	0	0	0	0	0	0	0	2**	1024	+ INF	0	1	1	1	all	INVALOP
0	1	0	0	0	0	0	0	0	0	2**	31		U*	U	U	U	all	OVRFLO
0	1	1	1	1	1	1	1	1	1	2**	30		U	U	U	U	RP,RN	OVRFLO,INEXO
0	1	1	1	1	1	1	1	1	1	2**	30		0	1	1	1	RZ,RM	INEXO
0	1	1	1	1	1	0	0	0	0	2**	30		U	U	U	U	RP,RN	OVRFLO,INEXO
0	1	1	1	1	1	0	0	0	0	2**	30		0	1	1	1	RZ,RM	INEXO
0	1	1	1	1	0	1	1	1	1	2**	30		U	U	U	U	RP	OVRFLO,INEXO
0	1	1	1	1	0	1	1	1	1	2**	30		0	1	1	1	RM,RN,RZ	INEXO
0	1	1	1	1	0	0	0	0	1	2**	30		U	U	U	U	RP	OVRFLO,INEXO
0	1	1	1	1	0	0	0	0	1	2**	30		0	1	1	1	RM,RN,RZ	INEXO
0	1	1	1	1	0	0	0	0	0	2**	30		0	1	1	1	all	
0	1	0	0	0	0	0	0	0	0	2**	0	one	0	0	0	1	all	
0	1	1	1	1	1	1	1	1	1	2**	-1	one - 1LSB	0	0	0	1	RN,RP	UNDFLO,INEXO
0	1	1	1	1	1	1	1	1	1	2**	-1	one - 1LSB	0	0	0	0	RZ,RM	UNDFLO,INEXO
0	1	0	0	0	0	0	0	0	1	2**	-1	1/2 + 1LSB	0	0	0	1	RN,RP	UNDFLO,INEXO
0	1	0	0	0	0	0	0	0	1	2**	-1	1/2 + 1LSB	0	0	0	0	RZ,RM	UNDFLO,INEXO
0	1	0	0	0	0	0	0	0	0	2**	-1	1/2	0	0	0	0	RP	UNDFLO,INEXO
0	1	0	0	0	0	0	0	0	0	2**	-1	1/2	0	0	0	1	RM,RN,RZ	UNDFLO,INEXO
0	1	0	0	0	0	0	0	0	0	2**	-1022	+ NORM.MIN	0	0	0	1	RP	UNDFLO,INEXO
0	1	0	0	0	0	0	0	0	0	2**	-1022	+ NORM.MIN	0	0	0	0	RM,RN,RZ	UNDFLO,INEXO
0	0	0	0	0	0	0	0	0	1	2**	-1022	+ DENORM.MIN	0	0	0	1	RP	UNDFLO,INEXO
0	0	0	0	0	0	0	0	0	1	2**	-1022	+ DENORM.MIN	0	0	0	0	RM,RN,RZ	UNDFLO,INEXO
0	0	0	0	0	0	0	0	0	0	2**	0	+ ZERO	0	0	0	0	all	
1	0	0	0	0	0	0	0	0	1	2**	-1022	- DENORM.MIN	1	1	1	1	RM	UNDFLO,INEXO
1	0	0	0	0	0	0	0	0	1	2**	-1022	- DENORM.MIN	0	0	0	0	RP,RN,RZ	UNDFLO,INEXO
1	1	0	0	0	0	0	0	0	0	2**	-1022	- NORM.MIN	1	1	1	1	RM	UNDFLO,INEXO
1	1	0	0	0	0	0	0	0	0	2**	-1022	- NORM.MIN	0	0	0	0	RP,RN,RZ	UNDFLO,INEXO
1	1	0	0	0	0	0	0	0	0	2**	-1	-1-2	1	1	1	1	RM	UNDFLO,INEXO
1	1	0	0	0	0	0	0	0	0	2**	-1	-1-2	0	0	0	0	RP,RN,RZ	UNDFLO,INEXO
1	1	0	0	0	0	0	0	0	1	2**	-1	-1-2 - 1LSB	1	1	1	1	RM,RN	UNDFLO,INEXO
1	1	0	0	0	0	0	0	0	1	2**	-1	-1-2 - 1LSB	0	0	0	0	RP,RZ	UNDFLO,INEXO
1	1	1	1	1	1	1	1	1	1	2**	-1	-one + 1LSB	1	1	1	1	RM,RN	UNDFLO,INEXO
1	1	1	1	1	1	1	1	1	1	2**	-1	-one + 1LSB	0	0	0	0	RP,RZ	UNDFLO,INEXO
1	1	0	0	0	0	0	0	0	0	2**	0	-one	1	1	1	1	all	
1	1	1	1	1	0	0	0	0	0	2**	30		1	0	0	1	all	
1	1	1	1	1	0	0	0	0	1	2**	30		1	0	0	0	RM	INEXO
1	1	1	1	1	0	0	0	0	1	2**	30		1	0	0	1	RP,RN,RZ	INEXO
1	1	1	1	1	0	1	1	1	1	2**	30		1	0	0	0	RM	INEXO
1	1	1	1	1	0	1	1	1	1	2**	30		1	0	0	1	RP,RN,RZ	INEXO
1	1	1	1	1	1	0	0	0	0	2**	30		1	0	0	0	RM,RN	INEXO
1	1	1	1	1	1	0	0	0	0	2**	30		1	0	0	1	RP,RZ	INEXO
1	1	1	1	1	1	1	1	1	1	2**	30		1	0	0	0	RM,RN	INEXO
1	1	1	1	1	1	1	1	1	1	2**	30		1	0	0	1	RP,RZ	INEXO
1	1	0	0	0	0	0	0	0	0	2**	31		1	0	0	0	all	
1	1	0	0	0	0	0	0	0	1	2**	31		1	0	0	0	RP,RN,RZ	INEXO
1	1	0	0	0	0	0	0	0	1	2**	31		U	U	U	U	RM	OVRFLO,INEXO
1	1	0	0	0	0	1	0	0	0	2**	31		1	0	0	0	RP,RZ	INEXO
1	1	0	0	0	0	1	0	0	0	2**	31		U	U	U	U	RM,RN	OVRFLO,INEXO
1	1	0	0	0	0	1	1	1	1	2**	31		1	0	0	0	RP,RZ	INEXO
1	1	0	0	0	0	1	1	1	1	2**	31		U	U	U	U	RM,RN	OVRFLO,INEXO
1	1	0	0	0	1	0	0	0	0	2**	31		U	U	U	U	all	OVRFLO
1	1	0	0	0	1	0	0	0	1	2**	31		U	U	U	U	all	OVRFLO,INEXO
1	1	0	0	0	0	0	0	0	0	2**	32		U	U	U	U	all	OVRFLO
1	1	0	0	0	0	0	0	0	0	2**	1024	- INF	1	1	1	1	all	INVALOP
1	1	X	X	X	X	X	X	X	X	2**	1024	- NAN	1	1	1	1	all	INVALOP

*"U" denotes an undefined result.
 NOTE: Heavy line indicates rounding boundary in source.

Table 6. Conversion of 64-Bit Double-Precision Floating-Point to 32-Bit Twos-Complement Integer

Sign	HB	f51	f30	f29	f28	f27	f1	f0	Unbiased Expt	Source Name	Sign	HB	f22	f1	f0	Unbiased Expt	Result Name	Rounding Modes	Status Flags	
0	1	X	X	X	X	X	X	X	2**	1024	0	1	1	1	1	2**	+NAN	all	INVALOP	
0	1	0	0	0	0	0	0	0	2**	1024	0	1	0	0	0	2**	+INF	all		
0	1	1	1	1	1	1	1	1	2**	1023	0	1	0	0	0	2**	+INF	RP,RN	OVRFLO,INEXO	
0	1	1	1	1	1	1	1	1	2**	1023	0	1	1	1	1	2**	+NORM.MAX	RZ,RM	OVRFLO,INEXO	
0	1	1	1	1	1	1	1	1	2**	127	0	1	1	1	0	2**	+INF	RP,RN	OVRFLO,INEXO	
0	1	1	1	1	1	1	1	1	2**	127	0	1	1	1	1	2**	+NORM.MAX	RZ,RM	INEXO	
0	1	1	1	1	1	1	1	1	2**	127	0	1	1	1	1	2**	+INF	RP	OVRFLO,INEXO	
0	1	1	1	1	1	1	1	1	2**	127	0	1	1	1	1	2**	+NORM.MAX	RM,RN,RZ	INEXO	
0	1	1	1	1	1	1	1	1	2**	127	0	1	1	1	1	2**	+NORM.MAX	all		
0	1	1	1	1	1	1	1	1	2**	127	0	1	1	1	1	2**	+NORM.MAX	RP	INEXO	
0	1	0	0	0	0	0	0	0	2**	-126	0	1	0	0	0	2**	+NORM.MIN	all		
0	1	1	1	1	1	1	1	1	2**	-127	0	0	1	1	1	2**	+NORM.MIN	RP,RN	INEXO	
0	1	1	1	1	1	1	1	1	2**	-127	0	0	1	1	1	2**	+DNRM.MAX	RZ,RM	UNDFLO,INEXO	
0	1	1	1	1	1	1	1	1	2**	-127	0	0	1	1	1	2**	+DNRM.MAX	all		
0	1	0	0	0	0	0	0	0	2**	-149	0	0	0	0	0	2**	+DNRM.MIN	all		
0	1	0	0	0	0	0	0	0	2**	-1022	0	0	0	0	1	2**	+DNRM.MIN	RP	UNDFLO,INEXO	
0	1	0	0	0	0	0	0	0	2**	-1022	0	0	0	0	0	2**	+ZERO	RM,RN,RZ	UNDFLO,INEXO	
0	0	1	1	1	1	1	1	1	2**	-1022	0	0	0	0	1	2**	+DNRM.MIN	RP	UNDFLO,INEXO	
0	0	1	1	1	1	1	1	1	2**	-1022	0	0	0	0	0	2**	+ZERO	RM,RN,RZ	UNDFLO,INEXO	
0	0	0	0	0	0	0	0	0	2**	-1022	0	0	0	0	1	2**	+DNRM.MIN	RP	UNDFLO,INEXO	
0	0	0	0	0	0	0	0	0	2**	-1022	0	0	0	0	0	2**	+ZERO	RM,RN,RZ	UNDFLO,INEXO	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	+ZERO	all		
1	0	0	0	0	0	0	0	0	2**	-1022	1	0	0	0	1	2**	-DNRM.MIN	RM	UNDFLO,INEXO	
1	0	0	0	0	0	0	0	0	2**	-1022	1	0	0	0	0	2**	-ZERO	RP,RN,RZ	UNDFLO,INEXO	
1	0	1	1	1	1	1	1	1	2**	-1022	1	0	0	0	1	2**	-DNRM.MIN	RM	UNDFLO,INEXO	
1	0	1	1	1	1	1	1	1	2**	-1022	1	0	0	0	0	2**	-ZERO	RP,RN,RZ	UNDFLO,INEXO	
1	1	0	0	0	0	0	0	0	2**	-1022	1	0	0	0	1	2**	-DNRM.MIN	RM	UNDFLO,INEXO	
1	1	0	0	0	0	0	0	0	2**	-1022	1	0	0	0	0	2**	-ZERO	RP,RN,RZ	UNDFLO,INEXO	
1	1	0	0	0	0	0	0	0	2**	-149	1	0	0	0	0	2**	-DNRM.MIN	all		
1	1	1	1	1	1	1	1	1	2**	-127	1	0	1	1	1	2**	-DNRM.MAX	all		
1	1	1	1	1	1	1	1	1	2**	-127	1	0	1	1	1	2**	-NORM.MIN	RM,RN	INEXO	
1	1	1	1	1	1	1	1	1	2**	-127	1	0	1	1	1	2**	-DNRM.MAX	RP,RZ	UNDFLO,INEXO	
1	1	0	0	0	0	0	0	0	2**	-126	1	1	0	0	0	2**	-NORM.MIN	all		
1	1	1	1	1	1	1	1	1	2**	127	1	1	1	1	1	2**	127	RM	INEXO	
1	1	1	1	1	1	1	1	1	2**	127	1	1	1	1	0	2**	127	RP,RN,RZ	INEXO	
1	1	1	1	1	1	1	1	1	2**	127	1	1	1	1	1	2**	127	all		
1	1	1	1	1	1	1	1	1	2**	127	1	1	1	1	1	2**	127	-NORM.MAX	RM	OVRFLO,INEXO
1	1	1	1	1	1	1	1	1	2**	127	1	1	1	1	1	2**	127	-NORM.MAX	RP,RN,RZ	INEXO
1	1	1	1	1	1	1	1	1	2**	127	1	1	1	1	1	2**	127	-INF	RM,RN	OVRFLO,INEXO
1	1	1	1	1	1	1	1	1	2**	127	1	1	1	1	1	2**	127	-NORM.MAX	RP,RZ	INEXO
1	1	1	1	1	1	1	1	1	2**	1023	1	1	0	0	0	2**	128	-INF	RM,RN	OVRFLO,INEXO
1	1	1	1	1	1	1	1	1	2**	1023	1	1	1	1	1	2**	127	-NORM.MAX	RP,RZ	OVRFLO,INEXO
1	1	0	0	0	0	0	0	0	2**	1024	1	1	0	0	0	2**	128	-INF	all	
1	1	X	X	X	X	X	X	X	2**	1024	1	1	1	1	1	2**	128	-NAN	all	INVALOP

NOTE: Heavy line indicates rounding boundary in source.

Table 7. Conversion of 64-Bit Double-Precision Floating-Point to 32-Bit Single-Precision Floating-Point (IEEE Mode)