Integrated Device Technology, Inc.

32-BIT AND 64-BIT IEEE FLOATING-POINT MULTIPLIER AND ALU

IDT721264
IDT721265

FEATURES:

- Conforms to the requirements of IEEE Standard 754, 1985 version, for full 32-bit and 64-bit multiply and arithmetic operations
- Very high-speed operation
  - 16.7 megaflops (60ns) pipelined ALU operation (Add/Subtract/Convert/Compare)
  - 16.7 megaflops (60ns) pipelined 32-bit (single precision) multiplications
  - 8.0 megaflops (120ns) pipelined 64-bit (double precision) multiplications
- Full floating-point function arithmetic logic unit including:
  - Add
  - Subtract
  - Absolute Value
  - Compare
  - Conversion to and from two's complement integer
  - Performs 2-A to support Newton-Raphson division
- Low-power (500mW typical per device) operation
- Single 5 volt supply—no need for two supplies
- Advanced CEMOS™ technology
- Flexible system design
  - Three 32-bit ports allow two data inputs and one result output every clock cycle
  - One, two or three port architectures supported
  - Single phase, edge-triggered clock interface, fully registered TTL-compatible inputs and outputs
- Full commercial and military ranges available
- Standard 144-pin grid array package
  - Pin and functionally compatible with Weitek 1264/1265

DESCRIPTION:

The IDT721264 floating-point multiplier and the IDT721265 floating-point ALU provide high speed 32-bit and 64-bit floating-point processing capability.

The IDT721264/65 are fabricated using IDT's advanced CEMOS technology and are capable of a total flow-through multiply latency (time required from the input of the operand until the result can be used by another device) of 180ns for single precision and 270ns for double precision multiplications. This ultra-high-speed performance is achieved by combining both state-of-the-art CEMOS technology and advanced circuit design techniques.

For signal processing applications, where higher throughput speeds are required, operations including the function specification can be pipelined for single precision multiplications, new operands can be loaded and a product unloaded every 60ns, while double precision multiplies can be accomplished at a 120ns rating. The IDT721265 ALU executes all operations at a 60ns pipelined throughput. All operations, including the function specifications, are pipelined so there is no penalty for interleaving various operations. The on-chip pipeline is automatically advanced, using internal timers, so explicit pipeline flushing is not required.

The flexible two-chip set operates in full conformance with the requirements of IEEE standard 754, 1985 version. It performs operations on single (32-bit) and double (64-bit) precision operands, as well as conversion to 32-bit two's complement integers (IDT721265 only). The IDT721264/65 accommodates all rounding modes, infinity and reserved operand representations and the treatment of exceptions such as overflow, underflow, invalid and inexact operations. Exact conformance to the standards ensures complete software portability between prototype development and final application. A "FAST" mode eliminates the time penalty for denormalized numbers by substituting zero for a denormalized number.

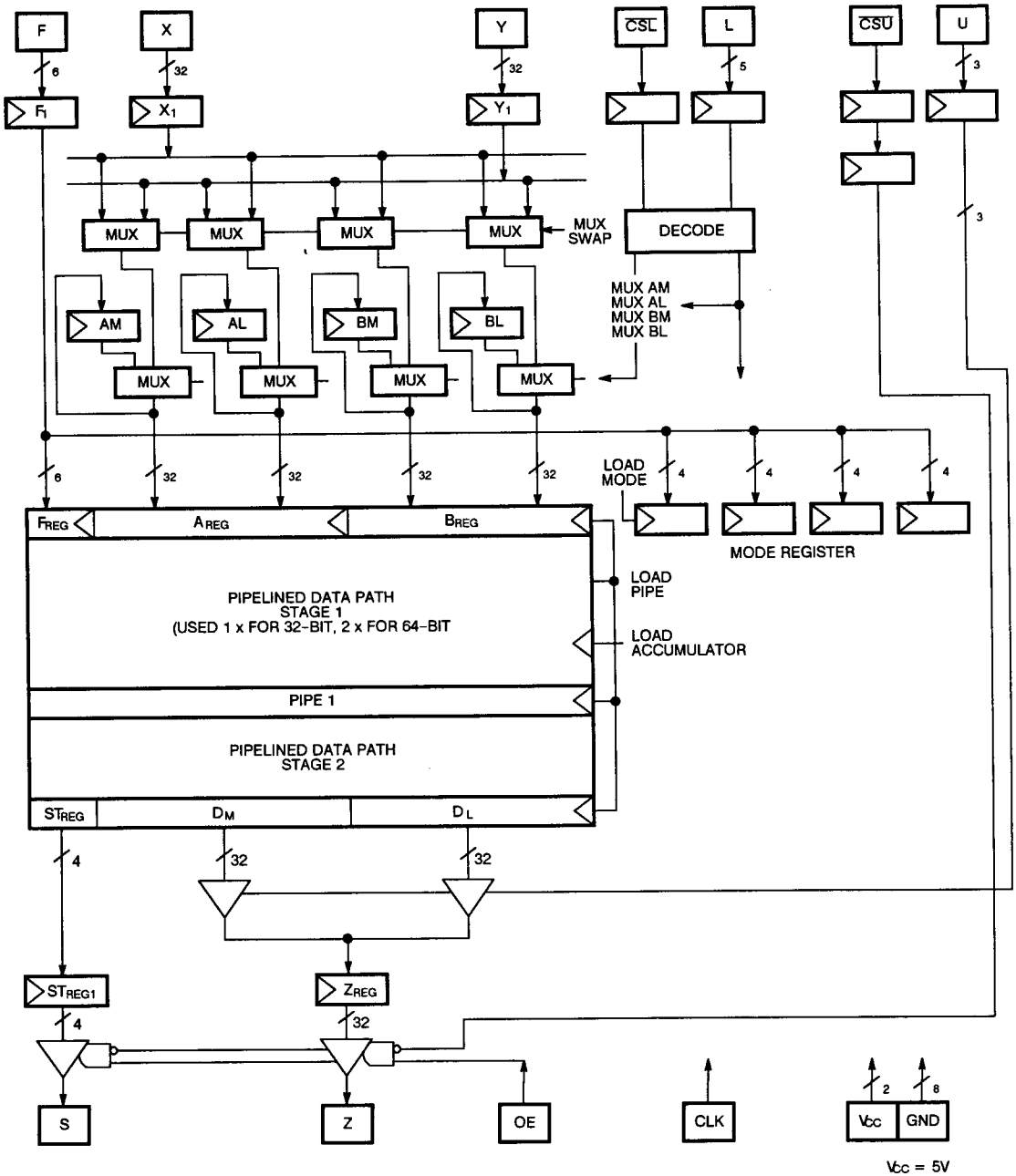
The flexible input/output architecture of these devices allows them to be used in systems with one, two or three 32-bit buses, or one 64-bit bus. Fully registered inputs and outputs, separately controlled, are loaded on each rising edge of the clock.

A 64-bit function control determines the arithmetic function to be performed while a 4-bit status output flags arithmetic exceptions and conditions. Both the function inputs and status outputs propagate along with the data to ease system design timing.

Military grade product is manufactured in compliance with the latest revision of MIL-STD-883C, Class B.

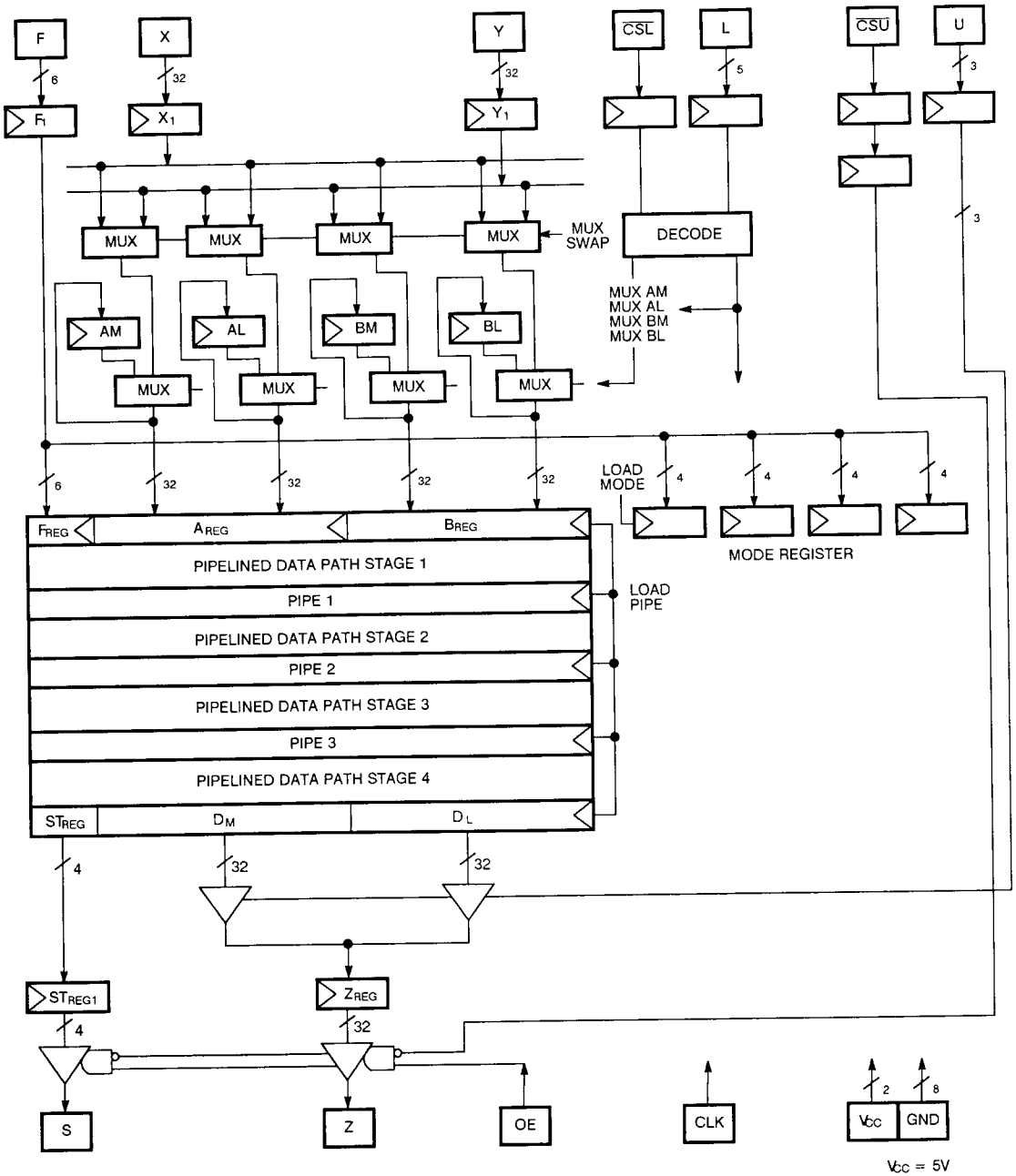
CEMOS is a trademark of Integrated Device Technology, Inc.

FUNCTIONAL BLOCK DIAGRAM
IDT721264 FLOATING-POINT MULTIPLIER

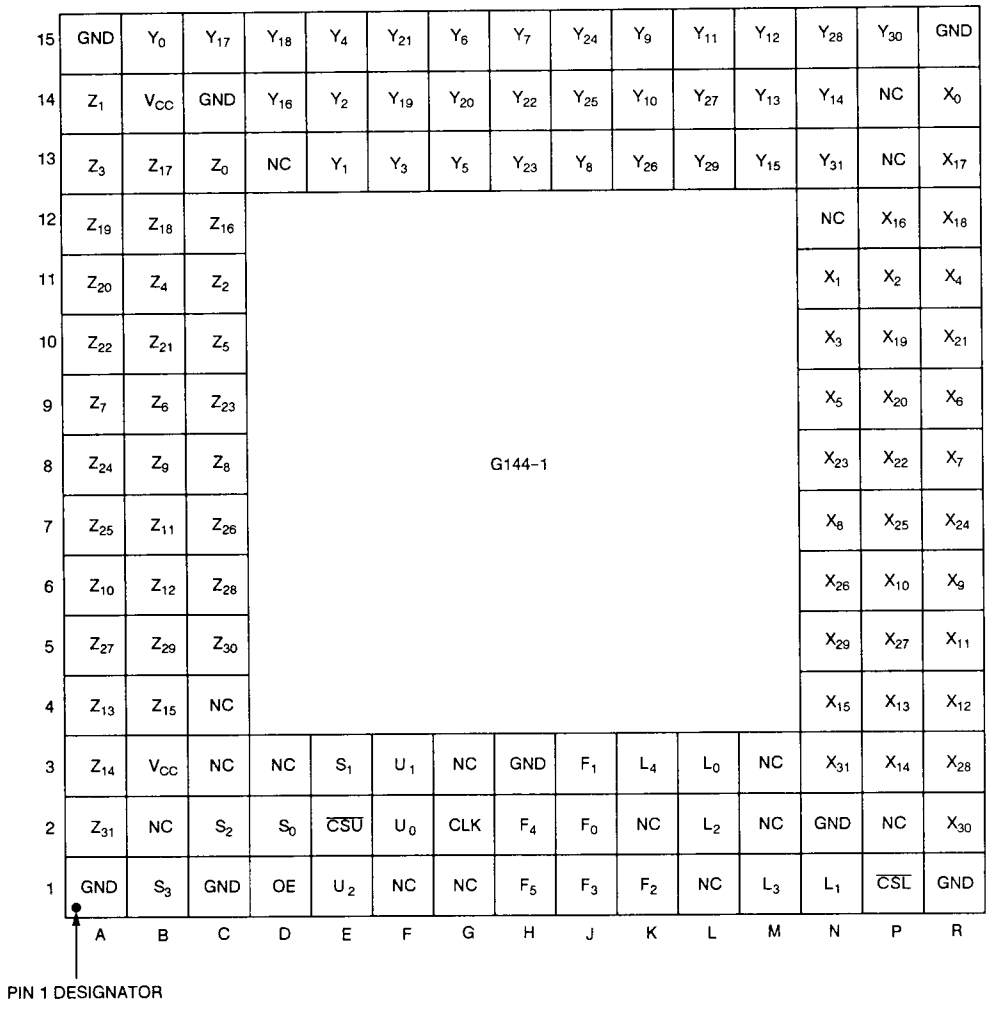


7

FUNCTIONAL BLOCK DIAGRAM
IDT721265 FLOATING-POINT ALU



PIN CONFIGURATION



PIN GRID ARRAY
TOP VIEW

ABSOLUTE MAXIMUM RATINGS[1]

SYMBOL	RATING	COMMERCIAL	MILITARY	UNIT
V_{TERM}	Terminal Voltage with Respect to GND	-0.5 to +7.0	-0.5 to +7.0	V
T_A	Operating Temperature	0 to +70	-55 to +125	°C
T_{BIAS}	Temperature Under Bias	-55 to +125	-65 to +135	°C
T_{STG}	Storage Temperature	-55 to +125	-65 to +155	°C
I_{OUT}	DC Output Current	50	50	mA

NOTE:
1. Stresses greater than those listed under ABSOLUTE MAXIMUM RATINGS may cause permanent damage to the device. This is a stress rating 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 reliability.

RECOMMENDED DC OPERATING CONDITIONS

SYMBOL	PARAMETER	MIN.	TYP.	MAX.	UNIT
V_{CCM}	Military Supply Voltage	4.5	5.0	5.5	V
V_{CC}	Commercial Supply Voltage	4.75	5.0	5.25	V
GND	Supply Voltage	0	0	0	V
V_{IH}	Input High Voltage	2.0	–	–	V
V_{IL}	Input Low Voltage	–	–	0.8	V

NOTE:
1. 1.5V under shoots are allowed for 10ns once per cycle.

DC ELECTRICAL CHARACTERISTICS

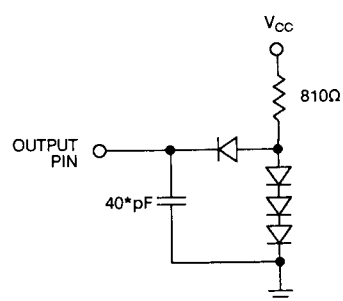
(Commercial V_{CC} = 5V ± 5%, T_A = 0°C to 70°C, Military V_{CC} = 5V ±10%, T_A = -55°C to +125°C)

SYMBOL	PARAMETER	TEST CONDITIONS	COMMERCIAL MIN.	COMMERCIAL MAX.	MILITARY MIN.	MILITARY MAX.	UNIT
$\|I_{LI}\|$	Input Leakage Current	V_{CC} = Max., V_{IN} = 0 to V_{CC}	–	10	–	20	µA
$\|I_{LO}\|$	Output Leakage Current	Hi Z, V_{CC} = Max., V_{OUT} = 0 to V_{CC}	–	10	–	20	µA
I_{CC}[1]	Operating Power Supply Current	Outputs Open, V_{CC} = Max.	–	100	–	120	mA
I_{CCQ}	Quiescent Power Supply Current	$V_{IN} \geq V_{CC}$ - 0.2V, $V_{IN} \leq$0.2V, V_{CC} = Max.	–	5	–	5	mA
I_{CC}/f (1,2)	Increase in Power Supply Current/MHz	V_{CC} = Max., f > 10MHz	–	4	–	6	mA/MHz
V_{OH}	Output High Voltage	V_{CC} = Min., I_{OH} = -2.0mA	2.4	–	2.4	–	V
V_{OL}	Output Low Voltage	V_{CC} = Min., I_{OL} = 8mA	–	0.4	–	0.4	V

NOTES:
1. I_{CC} is measured at 10MHz and V_{IN} = TTL voltages. For frequencies greater than 10MHz, the following equation is used for the commercial range: I_{CC} = 100 + 4(f-10)mA, where f = operating frequency in MHz. For the military range, I_{CC} = 120 + 6(f-10) where f = operating frequency in MHz.
2. For frequencies greater than 10MHz.

AC TEST CONDITIONS

Input Pulse Levels	GND to 3.0V
Input Rise/Fall Times	3ns
Input Timing Reference Levels	1.5V
Output Reference Levels	1.5V
Output Load	See Figure 1



***Includes scope and jig.**

Figure 1. Output Load

OPERATING CONDITIONS

(Commercial V_{CC} = 5V ± 5%, T_A = 0°C to 70°C, Military V_{CC} = 5V ±10%, T_A = -55°C to +125°C)

SYMBOL	PARAMETER	t_{CY} = 30ns MIN.	MAX.	t_{CY} = 40ns MIN.	MAX.	t_{CY} = 50ns MIN.	MAX.	t_{CY} = 60ns MIN.	MAX.	UNIT
t_{CY}	Clock Cycle Time	30	–	40	–	50	–	60	–	ns
t_{CH}	Clock HIGH Time	12	–	15	–	20	–	25	–	ns
t_{CL}	Clock LOW Time	12	–	15	–	20	–	25	–	ns
t_S	Input Set-up Time	11	–	13	–	15	–	15	–	ns
t_H	Input Hold Time	2	–	3	–	3	–	3	–	ns

AC ELECTRICAL CHARACTERISTICS

(Commercial V_{CC} = 5V ± 5%, T_A = 0°C to 70°C, Military V_{CC} = 5V ±10%, T_A = -55°C to

SYMBOL	PARAMETER	t_{CY} = 30ns MIN.	MAX.	t_{CY} = 40ns MIN.	MAX.	t_{CY} = MIN.	ns MAX.	t_{CY} = 60ns MIN.	MAX.	UNIT
t_{DO}	Output Delay Time	–	28	–		–	35	–	35	ns
t_{VO}	Output Data Valid	5	–	5		5	–	5	–	ns
t_{OZ}	Output Disable Time	–	25	–		–	35	–	35	ns
t_{ZO}	Output Enable Time	–	25	–		–	35	–	35	ns
t_{LA}	Total Latency Time IDT721265 ALU All Functions IDT721264 MPY 32-Bit Functions IDT721264 MPY 64-Bit Functions	– – –	270 180 270		360 240 360	– – –	450 300 450	– – –	540 360 540	ns ns ns
t_{OP}	Pipelined Time per Stage IDT721265 ALU All Functions IDT721264 MPY 32-Bit Functions IDT721264 MPY 64-Bit Functions	– – –	60 60 120	– – –	80 80 160	– – –	100 100 200	– – –	120 120 240	ns ns ns
t_{LAP}	Pipelined Total Latency IDT721265 ALU All Functions IDT721264 MPY 32-Bit Functions IDT721264 MPY 64-Bit Functions		360 210 300	– – –	480 280 400	– – –	600 350 500	– – –	720 420 600	ns ns ns
Array Time (IDT721265)	A, B register to Z register, pipeline registers transparent	–	100	–	120	–	150	–	180	ns
Array Time (IDT721264)	Time to Make One Pass Thro Multiplier Array for 64-Bit or 32	–	60	–	80	–	100	–	120	ns
t_{P64}	Time for a 64-Bit Resul go from the Pipeline Register t e Inp Register of the Z-Reg DL Transparent)	–	90	–	120	–	150	–	180	ns
t_{P32}	Time for a 32-B ge go from the Pipeline Re Input of the DM,	–	60	–	80	–	100	–	120	ns
t_{FLOW64}	Time Require 64-Bit Data to Make One Pa Throu e Array and the Tran peline Registers and Tr arent DL to the Input of the	–	120	–	160	–	200	–	240	ns
t_{FLOW32}	Req d for 32-Bit Data to Make One Pass Through the Array and the Transparent Pipeline Registers and Transparent DM, DL to the Input of the Z-Reg	–	120	–	160	–	200	–	240	ns

7

SIGNAL DESCRIPTIONS:

INPUTS:

X_{0-31} (Input Operand)
X operand inputs, 32-bit.

Y_{0-31} (Input Operand)
Y operand inputs, 32-bit.

CONTROL:

L_{0-4} (Load Control)
The input configuration of the IDT721264 and IDT721265 can be configured through the use of the 5-bit load control to specify the destination of the data from the X and Y inputs to the AM, AL, BM, BL registers or the arithmetic array.

F_{0-5} (Function Control)
The function configuration of the IDT721264 and IDT721265 can be configured through the use of the 6-bit function control to specify the operation to be performed. See Tables 3 and 4 for the specific function controls for the multiplier and ALU.

U_{0-2} (Unload Control)
The unload control (U_{0-2}) chooses the source of the output.

$\overline{CSL}$ (Input Enable)
When $\overline{CSL}$ is low, input ports X and Y are enabled. Input data buses may be shared with the ALU and multiplier with $\overline{CSL}$.

$\overline{CSU}$ (Synchronous Output Enable)
When $\overline{CSU}$ is low, output port Z is enabled. Therefore, microcode can control the three-stating of the Z output. Since $\overline{CSL}$ is pipelined, it takes effect 2 clock cycles after it is asserted.

$\overline{OE}$ (Asynchronous Output Enable)
When $\overline{OE}$ is high, output port Z is enabled. When $\overline{OE}$ is high, the Z output is enabled if $\overline{CSU}$ is low.

CLK (Clock)
The clock input.

OUTPUTS:

Z_{0-31} (Result)
The Z result output, 32-bit, three-state.

S_{0-3} (Status)
The 4-bit status output indicates any exceptions which resulted from multiplier or ALU operations.

POWER SUPPLY:

V_{CC} (Power Supply)
Two power supply pins, 5V.

GND (Ground)
Eight ground pins, 0V.

GENERAL OPERATING MODES

Both the Multiplier and the ALU are architected identically with two input ports and one output port that surround the pipelined arithmetic array. The function control (6-bits) controls the selection of the arithmetic operations with the input and output ports controlled by a total of 8 bits of the load and unload control registers.

INPUT PORTS

The IDT721264 multiplier and the IDT721265 ALU have identical input and control structures that handle data on two 32-bit buses (X and Y). The on-chip registers (AL, AM, BL, BM) can be written from either of the buses or data can be passed from the inputs directly into the arithmetic unit.

Both devices can be used in a range of bus configurations for operations in both 32-bit and 64-bit by configuring the input data in combination with the high bandwidth output. Transfers of data input and output can be made at twice the pipeline rate. The input buses are fully registered and can be configured for one or two 32-bit inputs or one 64-bit output. These registers are loaded on each LOW-to-HIGH transition of the clock provided $\overline{CSL}$ is held LOW.

LOAD CONTROL

The Load Control (L_{0-4}) is used to transfer data from the input ports to the internal registers or the arithmetic array. L_0 controls the initiation of an operation. When this input is LOW only a data transfer occurs while, when it is held HIGH, data is transferred and an operation is begun. The sequence of events is as follows: two registers (AREG and BREG) are loaded from the specified AL, AM, BL, BM register and the X and Y ports and the FREG is loaded from port F while, on the next cycle, the specified operation in the FREG begins with the data already loaded into the AREG and BREG. The X and Y ports can be used as single operand operations and must be loaded into the AREG. The configuration of these ports can be accomplished by using the Mode bits M_{15} and M_{14} for 16-, 32- and 64-bit data. The most significant halves of the AREG and BREG must be loaded with any 32-bit operands.

UNLOAD CONTROL

The Unload Control (U_{0-2}) chooses whether the DM or DL register is sent to the Z register. The DM register stores the result of 32-bit floating-point operations. With 64-bit operations, the most significant 32 bits are stored in the DM register; the least significant half is stored in the DL register. A 32-bit result is sent from the Z register to the Z output port on each clock cycle.

TABLE 1. LOAD CONTROL TRUTH TABLE

L_4	L_3	L_2	L_1	L_0		LOAD OPERATION
0	0	0	0	0	(0)	(NOP)
0	0	0	0	1	(1)	AM, AL → AREG; BM, BL → BREG; F1 → FREG
0	0	0	1	0	(2)	Load Mode
0	0	0	1	1	(3)	—Reserved
0	0	1	0	0	(4)	Y1 → AL; X1 → BL
0	0	1	0	1	(5)	Y1 → AL; X1 → BL; AM, Y1 → AREG; BM, X1 → BREG; F1 → FREG
0	0	1	1	0	(6)	Y1 → AM; X1 → BM
0	0	1	1	1	(7)	Y1 → AM; X1 → BM; Y1, AL → AREG; X1, BL → BREG; F1 → FREG
0	1	0	0	0	(8)	X1 → BM; Y1 → BL
0	1	0	0	1	(9)	X1 → BM; Y1 → BL; AM, AL → AREG; X1, Y1 → BREG; F1 → FREG
0	1	0	1	0	(10)	X1 → AM; Y1 → AL
0	1	0	1	1	(11)	X1 → AM; Y1 → AL; X1, Y1 → AREG; BM, BL → BREG; F1 → FREG
0	1	1	0	0	(12)	X1 → AL; Y1 → BL
0	1	1	0	1	(13)	X1 → AL; Y1 → BL; AM, X1 → AREG; BM, Y1 → BREG; F1 → FREG
0	1	1	1	0	(14)	X1 → AM; Y1 → BM
0	1	1	1	1	(15)	X1 → AM; Y1 → BM; X1, AL → AREG; Y1, BL → BREG; F1 → FREG
1	0	0	0	0	(16)	Y1 → BM
1	0	0	0	1	(17)	Y1 → BM; AM, AL → AREG; Y1, BL → BREG; F1 → FREG
1	0	0	1	0	(18)	Y1 → BL
1	0	0	1	1	(19)	Y1 → BL; AM, AL → AREG; BM, Y1 → BREG; F1 → FREG
1	0	1	0	0	(20)	Y1 → AL
1	0	1	0	1	(21)	Y1 → AL; AM, Y1 → AREG; BM, BL → BREG; F1 → FREG
1	0	1	1	0	(22)	Y1 → AM
1	0	1	1	1	(23)	Y1 → AM; Y1, AL → AREG; BM, BL → BREG; F1 → FREG
1	1	0	0	0	(24)	X1 → BM
1	1	0	0	1	(25)	X1 → BM; AM, AL → AREG; X1, BL → BREG; F1 → FREG
1	1	0	1	0	(26)	X1 → BL
1	1	0	1	1	(27)	X1 → BL; AM, AL → AREG; BM, X1 → BREG; F1 → FREG
1	1	1	0	0	(28)	X1 → AL
1	1	1	0	1	(29)	X1 → AL; AM, X1 → AREG; BM, BL → BREG; F1 → FREG
1	1	1	1	0	(30)	X1 → AM
1	1	1	1	1	(31)	X1 → AM; X1, AL → AREG; BM, BL → BREG; F1 → FREG

7

LOAD SEQUENCES

32-BIT OPERATIONS WITH TWO 32-BIT PORTS

OPERATION	L_4	L_3	L_2	L_1	L_0	INST#
LOAD MODE[1]	0	0	0	1	0	(2)
Y1, AL → AREG; X1, BL → BREG; F1 → FREG; Y1 → AM; X1 → BM	0	0	1	1	1	(7)

NOTE:
1. If the mode does not change between operations, it does not need to be reloaded.

64-BIT OPERATIONS USING THE X AND Y PORTS AS A SINGLE 64-BIT PORT

OPERATION	L_4	L_3	L_2	L_1	L_0	INST#
LOAD MODE	0	0	0	1	0	(2)
X1 → AM; Y1 → AL	0	1	0	1	0	(10)
AM, AL → AREG; X1, Y1 → BREG; F1 → FREG; X1 → BM; Y1 → BL	0	1	0	0	1	(9)

TABLE 2. UNLOAD CONTROL TRUTH TABLE

CSU	U_2	U_1	U_0	EDGE #1	EDGE #2
0	0	0	0	$DM_{31-0} \rightarrow$ ZREG	STREG1 $\rightarrow S^+$, ZREG $\rightarrow Z^+$
0	0	1	0	DM_{31-16}, $DL_{31-16} \rightarrow$ ZREG	STREG1 $\rightarrow S^+$, ZREG $\rightarrow Z^+$
0	1	0	0	$DL_{31-0} \rightarrow$ ZREG	STREG1 $\rightarrow S^+$, ZREG $\rightarrow Z^+$
0	1	1	0	DM_{15-0}, $DL_{15-0} \rightarrow$ ZREG	STREG1 $\rightarrow S^+$, ZREG $\rightarrow Z^+$
1	X	X	X		S^+ and Z^+ Tri-Stated

TABLE 3. FUNCTION CONTROLS FOR FLOATING-POINT MULTIPLIER

F_2	F_1	F_0		OPERATION	MNEMONIC	DESCRIPTION
0	0	0	(0)	F32 x F32	MUL32	Single Multiply
0	0	1	(1)	F64 x F64	MUL64	Double Multiply
0	1	0	(2)	W32 x F32	MULAW32	Single Multiply, A Wrapped
0	1	1	(3)	W64 x F64	MULAW64	Double Multiply, A Wrapped
1	0	0	(4)	F32 x W32	MULBW32	Single Multiply, B Wrapped
1	0	1	(5)	F64 x W64	MULBW64	Double Multiply, B Wrapped
1	1	0	(6)	W32 x W32	MULABW32	Single Multiply, A & B Wrapped
1	1	1	(7)	W64 x W64	MULABW64	Double Multiply, A & B Wrapped

F_5	F_4	F_3		OPERATION	MNEMONIC	DESCRIPTION
0	0	0	(0)	A x B	MUL	Multiply
0	0	1	(1)	$\|A\|$ x B	MULABSA	B Times Magnitude of A
0	1	0	(2)	A x $\|B\|$	MULABSB	A Times Magnitude of B
0	1	1	(3)	$\|A\|$ x $\|B\|$	MULABSAB	Magnitude of A Times B
1	0	0	(4)	-(A x B)	MULNEG	Multiply and Negate
1	0	1	(5)	(-$\|A\|$) x B	MULNEGA	B Times Negative Value of A
1	1	0	(6)	A x (-$\|B\|$)	MULNEGB	A Times Negative Value of B
1	1	1	(7)	-($\|A\|$ x $\|B\|$)	MULNEGAB	Negative Value of A Times B

MULTIPLIER OPERATION: IDT721264

The IDT721264 has exception detection handling circuitry, a 56 x 28-bit multiplier array, an exponent adder circuit, a normalizing shifter and a rounding circuit for IEEE format adjustment.

The exception detection circuit is at the beginning of the multiplier. Exceptions can be Not-a-Number (NaN) or a denormalized input and timings are handled like normal numbers.

A clocked 54 x 28-bit multiplier array multiplies the mantissa portion of the floating-point number. A single precision multiply takes one pass through the array; a double precision multiply takes two passes. Each pass through the array takes two clock cycles.

Partial results are stored in the accumulator (an adder and a transparent latch). The cycle time determines the number of clock cycles required to complete a multiplication. A long cycle time requires fewer clock cycles to complete the operation. A clock time of 30ns calls for four cycles to perform a double precision multiply. In the first two clock cycles, the operands are multiplied in the array. On the second cycle, the accumulator register must be latched to retain the results. On the fourth cycle the accumulator register must be transparent so that the results are passed to the pipeline register. An accumulator timer can be set to latch or pass results one to four clock cycles after the beginning of the operation. The timer is reset at the beginning of each operation.

MULTIPLIER FUNCTION CONTROL

The multiplier controls are given in Table 3. The multiplier functions can be considered briefly in the following manner:

F_5	F_4	F_3	F_2	F_1	F_0
CS	MB	MA	WB	BA	SD

CS: 1-> Complement sign of result
MB: 1-> Magnitude of B
MA: 1-> Magnitude of A
WB: 1-> Operand B is wrapped
WA: 1-> Operand A is wrapped
SD: 0-> Single precision operation
SD: 1-> Double precision operation

The Mode Control bits are loaded from the F_{0-3} function control bits. The F_{5-4} function control bits determine which of the four 4-bit mode control subsets is loaded.

F_5	F_4	EDGE 0	EDGE 1
0	0	F → F1	F → M_{3-0}
0	1	F → F1	F → M_{7-4}
1	0	F → F1	F → M_{11-8}
1	1	F → F1	F → M_{15-12}

MULTIPLIER MODE CONTROL

IEEE OR FAST MODE

Mode bit M_0 controls the way denormalized numbers are handled. If M_0 is 0, the IEEE format is used. The multiplier generates denormalized operand exceptions and produces UNRM values on underflow exceptions. The denormalized operands are sent to the ALU to be wrapped; the wrapped numbers (WNRMs) can then be multiplied. The IEEE Compatibility section discusses this in detail.

If M_0 is 1, the Fast mode is used in order to achieve the maximum performance, by eliminating the direct handling of denormalized numbers. The multiplier flushes denormalized operands (DNRMs) to zero and rounds underflow or unnormalized results (UNRMs) to zero. Mode bit M_2 must be set to zero in the Fast mode.

M_0	DESCRIPTION
0	IEEE Mode
1	Fast Mode

ROUNDING MODE

Mode bits M_1, M_2 and M_3 select the Rounding mode. Renormalization and IEEE rounding functions are performed between the pipeline register and the DM and DL registers.

M_3	M_2	DESCRIPTION
0	0	Round to nearest value or if a tie, round to even significand
0	1	Round to zero
1	0	Round towards positive infinity
1	1	Round towards negative infinity

PIPELINE CONFIGURATION

Mode bit M_4 controls whether the DM and DL registers are transparent. Mode bit M_5 controls whether the pipeline register is transparent.

M_4	DESCRIPTION
0	Transparent DM, DL
1	Latched DM, DL

M_5	DESCRIPTION
0	Transparent pipeline register
1	Latched pipeline register

ACCUMULATOR ADVANCE CONTROL

Mode bits M_{7-6} control the timing of the partial product accumulator. The accumulator is alternately latched and made transparent every N + 1 cycles, where N is the value of M_{7-6}. The accumulator timer is reset at the beginning of each operation. The accumulator timer is used to achieve maximum throughput.

M_7	M_6	DESCRIPTION
0	0	N = 1, Clock/1
0	1	N = 2, Clock/2
1	0	N = 3, Clock/3
1	1	N = 4, Clock/4

PIPELINE ADVANCE CONTROL

Mode bits M_{11-8} control the pipeline advance control of the pipeline registers. If M_{11-8} are all zeros, the pipeline registers will only be latched at the beginning of an operation. If M_{11-8} are non-zero values, N, the pipeline registers will be clocked at the beginning of every operation and every N cycles after the beginning of every operation. The internal pipeline advance timer is reset at the beginning of every operation.

For example, if N = 4 and operations are started on cycles 0, 6 and 10, pipeline advances will occur on cycles 0, 4, 6, 10, 14, 18 and so on. The pipeline advance control is used to achieve maximum throughput.

M_{11}	M_{10}	M_9	M_8	DESCRIPTION
0	0	0	0	N = 0, pipeline registers are latched
0	0	0	1	N = 1, pipeline registers are clocked 1 cycle after first operation
0	0	1	0	N = 2, pipeline registers are clocked 2 cycles after first operation
0	0	1	1	N = 3, pipeline registers are clocked 3 cycles after first operation
.	.	.	.	.
.	.	.	.	.
1	1	1	1	N = 15, pipeline registers are clocked 15 cycles after first operation

BUS BANDWIDTH CONTROL

Mode bits M_{13-12} are not used. Mode bits M_{15-14} control the input bus bandwidth of the X and Y input ports. When M_{15-14} are set to zero, the X1 and Y1 registers are loaded every clock cycle from the X and Y ports.

M_{15}	M_{14}	DESCRIPTION	DATA PATH
0	0	N = 1, 32-but bus	X-> X1; Y-> Y1
0	1	N = 2, Reserved	
1	0	N = 3, Unused	
1	1	N = 4, Unused	

ALU OPERATION: IDT721265

The IDT721265 ALU has five basic components: exception detection circuitry, a shifter to normalize the smaller of the two input operands, a 57-bit adder, a shifter to renormalize the result and IEEE rounding circuitry. The IDT721265 is easily considered as an ALU with multiple internal pipeline registers. The internal pipeline registers and the DM and DL registers can be made transparent by mode bits M_{7-4}.

The pipeline registers are clocked at the beginning of each operation and every N cycles thereafter, when N is given a value by mode bits M_{11-8}.

ALU FUNCTION CONTROL

The IDT721265's function controls are shown in Table 4. The IDT superset functions are highlighted in Table 5.

The Mode Control bits are loaded from the F_{3-0} function control bits. The F_{5-4} function control bits determine which of the four 4-bit mode control subsets is loaded.

F_5	F_4	EDGE 0	EDGE 1
0	0	F → F1	F1 → M_{3-0}
0	1	F → F1	F1 → M_{7-4}
1	0	F → F1	F1 → M_{11-8}
1	1	F → F1	F1 → M_{15-12}

ALU MODE CONTROL

IEEE OR FAST MODE

Mode bit M_0 controls the way denormalized numbers are handled. If M_0 is 0, the IEEE format is used. The ALU generates denormalized operand exceptions and produces UNRM values on underflow exceptions. The IEEE Compatibility section discusses this in detail.

If M_0 is 1, the Fast mode is used in order to achieve the maximum performance by eliminating the direct handling of denormalized numbers. The ALU flushes denormalized operands (UNRMs) to zero and rounds underflow or unnormalized results (UNRMs) to zero. Mode bit M_2 must be set to zero in the Fast mode.

M_0	DESCRIPTION
0	IEEE Mode
1	Fast Mode

ROUNDING MODE

Mode bits M_1, M_2 and M_3 select the Rounding mode. Renormalization and IEEE rounding functions are performed between the pipeline register and the DM and DL registers.

M_3	M_2	M_1	DESCRIPTION
0	0	0	Round to nearest value or if a tie, round to even significand
0	1	0	Round to zero
1	0	0	Round towards positive infinity
1	1	0	Round towards negative infinity
X	X	1	Round to zero, all cases

PIPELINE CONFIGURATION

Mode bits M_{7-4} determine which of the pipeline registers and DM and DL registers are made transparent. If the mode bit is low, the corresponding register is made transparent. If the mode bit is high, the register is latched by the rising edge of the clock. If the pipeline registers are made transparent, the latency time is reduced at the expense of slower overall throughput. The highest system throughput results from enabling all the pipeline registers.

Mode bit M_4 controls whether the DM and DL registers are transparent. Mode bits M_{7-5} control which of the pipeline registers are transparent.

M_7	M_6	M_5	M_4
PIPE1	PIPE2	PIPE3	STREG, DM, DL

PIPELINE ADVANCE CONTROL

Mode bits M_{11-8} control the pipeline advance control of the pipeline registers. If M_{11-8} are all zeros, the pipeline registers will only be latched at the beginning of an operation. If M_{11-8} are non-zero values, N, the pipeline registers will be clocked at the beginning of every operation and every N cycles after the beginning of every operation. The internal pipeline advance timer is reset at the beginning of every operation.

For example, if N = 4 and operations are started on cycles 0, 6 and 10, pipeline advances will occur on cycles 0, 4, 6, 10, 14, 18 and so on. The pipeline advance control is used to achieve maximum throughput.

M_{11}	M_{10}	M_9	M_8	DESCRIPTION
0	0	0	0	N = 0, pipeline registers are latched
0	0	0	1	N = 1, pipeline registers are clocked 1 cycle after first operation
0	0	1	0	N = 2, pipeline registers are clocked 2 cycles after first operation
0	0	1	1	N = 3, pipeline registers are clocked 3 cycles after first operation
.	.	.	.	.
.	.	.	.	.
1	1	1	1	N = 15, pipeline registers are clocked 15 cycles after first operation

BUS BANDWIDTH CONTROL

Mode bits M_{13-12} are not used. Mode bits M_{15-14} control the input bus bandwidth of the X and Y input ports. When M_{15-14} are set to zero, the X1 and Y1 registers are loaded every clock cycle from the X and Y ports.

M_{15}	M_{14}	DESCRIPTION	DATA PATH
0	0	N = 1, 32-but bus	X→ X1; Y →Y1
0	1	N = 2, Reserved	
1	0	N = 3, Unused	
1	1	N = 4, Unused	

7

TABLE 4. FUNCTION CONTROLS FOR ALU – IDT721265

F_5	F_4	F_3	F_2	F_1	F_0		OPERATION	DESCRIPTION
0	0	0	0	0	0	(0)	F32 - F32	Single Subtract
0	0	0	0	0	1	(1)	F64 - F64	Double Subtract
0	0	0	0	1	0	(2)	|F32 - F32|	Single ABS Subtract
0	0	0	0	1	1	(3)	|F64 - F64|	Double ABS Subtract
0	0	0	1	0	0	(4)	|F32| - |F32|	Single Subtract ABS (1)
0	0	0	1	0	1	(5)	|F64| - |F64|	Double Subtract ABS (1)
0	0	0	1	1	0	(6)		RESERVED
0	0	0	1	1	1	(7)		RESERVED
0	0	1	0	0	0	(8)	-F32 + 0	Single Negate
0	0	1	0	0	1	(9)	-F64 + 0	Double Subtract
0	0	1	0	1	0	(10)	2 - F32	Single 2-A (1)
0	0	1	0	1	1	(11)	2 - F64	Double 2-A(1)
0	0	1	1	0	0	(12)	-F32 - F32	Single 2's Complement of Addition (1)
0	0	1	1	0	1	(13)	-F64 - F64	Double 2's Complement of Addition (1)
0	0	1	1	1	0	(14)		RESERVED
0	0	1	1	1	1	(15)		RESERVED
0	1	0	0	0	0	(16)	F32 + F32	Single Addition
0	1	0	0	0	1	(17)	F64 + F64	Double Addition
0	1	0	0	1	0	(18)	|F32 + F32|	Single ABS Addition
0	1	0	0	1	1	(19)	|F64 + F64|	Double ABS Addition
0	1	0	1	0	0	(20)	|F32| + |F32|	Single Add ABS
0	1	0	1	0	1	(21)	|F64| + |F64|	Double Add ABS
0	1	0	1	1	0	(22)		RESERVED
0	1	0	1	1	1	(23)		RESERVED
0	1	1	0	0	0	(24)	F32 + 0	Single Pass
0	1	1	0	0	1	(25)	F64 + 0	Double Pass
0	1	1	0	1	0	(26)	F32 + |F32|	Single Mixed Addition (1)
0	1	1	0	1	1	(27)	F64 + |F64|	Double Mixed Addition (1)
0	1	1	1	0	0	(28)	|F32| + 0	Single Pass ABS
0	1	1	1	0	1	(29)	|F64| + 0	Double Pass ABS
0	1	1	1	1	0	(30)		RESERVED
0	1	1	1	1	1	(31)		RESERVED
1	0	0	0	0	0	(32)	COMP F32 - F32	Single Compare
1	0	0	0	0	1	(33)	COMP F64 - F64	Double Compare
1	0	0	0	1	0	(34)	F32 - |F32|	Single Mixed Subtract (1)
1	0	0	0	1	1	(35)	F64 - |F64|	Double Mixed Subtract (1)
1	0	0	1	0	0	(36)	COMP |F32| - |F32|	Single Compare ABS
1	0	0	1	0	1	(37)	COMP |F64| - |F64|	Double Compare ABS
1	0	0	1	1	0	(38)		RESERVED
1	0	0	1	1	1	(39)		RESERVED
1	0	1	0	0	0	(40)	COMP F32 - 0	Single Compare with Zero
1	0	1	0	0	1	(41)	COMP F64 - 0	Double Compare with Zero
1	0	1	0	1	0	(42)	-|F32| + 0	Single Negate of ABS (1)
1	0	1	0	1	0	(43)	-|F64| + 0	Double Negate of ABS (1)
1	0	1	1	0	0	(44)	-|F32| - |F32|	Single 2's Complement of Add ABS (1)
1	0	1	1	0	1	(45)	-|F64| - |F64|	Double 2's Complement of Add ABS (1)
1	0	1	1	1	0	(46)		RESERVED
1	0	1	1	1	1	(47)		RESERVED
1	1	0	0	0	0	(48)	U32 → D32 EX	Single Unwrap Exact
1	1	0	0	0	1	(49)	U64 → D64 EX	Double Unwrap Exact
1	1	0	0	1	0	(50)	D32 → W32	Single Wrap
1	1	0	0	1	0	(51)	D64 → W64	Double Wrap
1	1	0	1	0	0	(52)	U32 → D32 INX	Single Unwrap Inexact
1	1	0	1	0	1	(53)	U64 → D64 INX	Double Unwrap Inexact
1	1	0	1	1	0	(54)		RESERVED
1	1	0	1	1	1	(55)		RESERVED
1	1	1	0	0	0	(56)	F32 → I32	Single Fix
1	1	1	0	0	1	(57)	F64 → I64	Double Fix
1	1	1	0	1	0	(58)	I32 → F32	Single Float
1	1	1	0	1	0	(59)	I64 → F64	Double Float
1	1	1	1	0	0	(60)	F32 → F64	Single Convert to Double
1	1	1	1	0	1	(61)	F64 → F32	Double Convert to Single
1	1	1	1	1	0	(62)		RESERVED
1	1	1	1	1	1	(63)		RESERVED

NOTE:
1. IDT proprietary functions. Reserved functions in Weitek-Compatible mode.

TABLE 5. IDT721265 ALU SUPERSET FUNCTION CONTROLS

F_5	F_4	F_3	F_2	F_1	F_0		OPERATION	DESCRIPTION
0	0	0	1	0	0	(4)	|F32| - |F32|	Single Subtract Absolute Value
0	0	0	1	0	1	(5)	|F64| - |F64|	Double Subtract Absolute Value
0	0	1	0	1	0	(10)	2 - F32	Single 2 - A
0	0	1	0	1	1	(11)	2 - F64	Double 2 - A
0	0	1	1	0	0	(12)	-F32 - F32	Single 2's Complement of Addition
0	0	1	1	0	1	(13)	-F64 - F64	Double 2's Complement of Addition
0	1	1	0	1	0	(26)	F32 + |F32|	Single Mixed Addition
0	1	1	0	1	1	(27)	F64 + |F64|	Double Mixed Addition
1	0	0	0	1	0	(34)	F32 - |F32|	Single Mixed Subtract
1	0	0	0	1	1	(35)	F64 - |F64|	Double Mixed Subtract
1	0	1	0	1	0	(42)	-|F32| + 0	Single Negate of Absolute Value
1	0	1	0	1	1	(43)	-|F64| + 0	Double Negate of Absolute Value
1	0	1	1	0	0	(44)	-|F32| - |F32|	Single 2's Complement of Add Absolute Value
1	0	1	1	0	1	(45)	-|F64| - |F64|	Double 2's Complement of Add Absolute Value

RESULTS STATUS

The 4-bit Result Status (S_{3-0}) indicates any exceptions or conditions of the results of a floating-point operation. Comparison conditions are shown on S_{3-0} when comparison operations are performed. Exception status is shown on S_{3-0} when exceptions occur on an operation. Table 6 details the results status indicators.

TABLE 6. STATUS TRUTH TABLE

S_3^+	S_2^+	S_1^+	S_0^+		COMPARISON CONDITION	EXCEPTION STATUS
0	0	0	0	(0)	Equal	Result = +0 or -0, exact
0	0	0	1	(1)	Less than	Result = +infinity or -infinity, exact
0	0	1	0	(2)	Greater than	Result finite and < > 0, exact
0	0	1	1	(3)		Result finite and < > 0, inexact
0	1	0	0	(4)		– Not used
0	1	0	1	(5)		Overflow & inexact
0	1	1	0	(6)		Underflow
0	1	1	1	(7)		Underflow & inexact
1	0	0	0	(8)		Operand A is denormalized
1	0	0	1	(9)		Operand B is denormalized
1	0	1	0	(10)		Operands A & B are denormalized
1	0	1	1	(11)		– Not used
1	1	0	0	(12)		Operand A is NAN
1	1	0	1	(13)		Operand B is NAN
1	1	1	0	(14)		Operands A & B are NAN
1	1	1	1	(15)	Unordered	Invalid Operation

During certain operations, more than one exception may occur. The higher priority exception will be indicated on the result status output.

PRIORITY	EXCEPTION
Highest	Operands A & B are NAN
	Operand A is NAN
	Operand B is NAN
	Invalid Operation
	Operands A & B are Denormalized
	Operand A is Denormalized
	Operand B is Denormalized
	Underflow & Inexact
	Underflow
	Overflow & Inexact
Lowest	Result is Finite < > 0, Inexact

TIMING

The IDT721264 and IDT721265 are designed to be able to operate as pipelined processors, to maximize systems throughput or to be able to operate as flow-through processors to minimize the latency period from input to output result. The following figures explain the various timing constraints for pipelined and flow-through modes in both single and double precision operations.

7

MULTIPLIER TIMINGS

64-BIT MAXIMUM PIPELINED THROUGHPUT

For maximum throughput for 64-bit multiplications, the accumulator timer should be set to clock/2 (M_{7-6} = 1), the pipeline advance control should be set to four (M_{11-8} = 4), the pipeline registers should be latched (M_5 = 1) and the DM and DL registers should be transparent (M_4 = 0). The timing is shown in Figure 3.

The first cycle after the inputs and function code are loaded, the 64-bit multiplication begins. Two cycles later, the accumulator stores the partial product of the first pass through the array. Two cycles later, the unrounded results are latched into the pipeline register. Three cycles later (t_{P64}), the result is available on the output port.

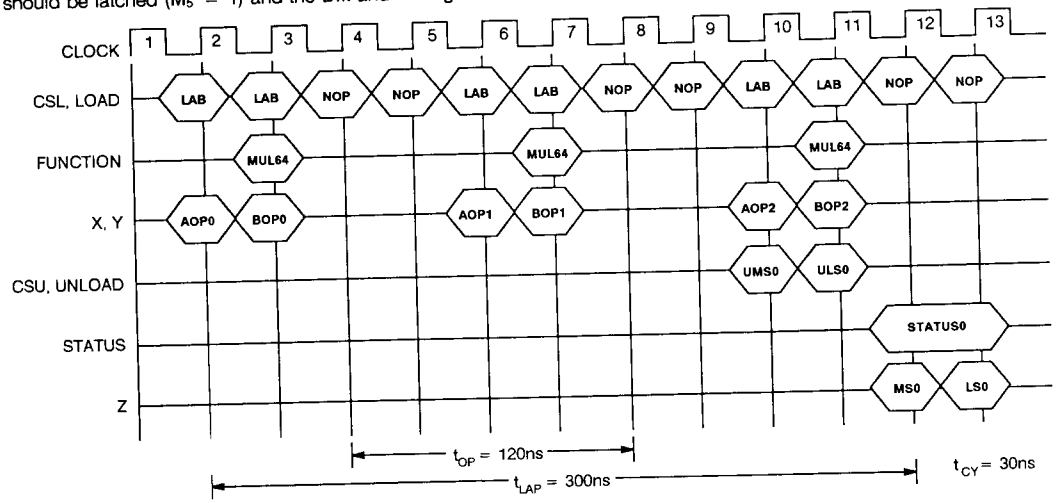


Figure 3. IDT721264 64-bit Pipelined Operation Timing

32-BIT MAXIMUM PIPELINED THROUGHPUT

For maximum throughput for 32-bit multiplications, the accumulator timer should be set to clock/1 (M_{7-6} = 0), the pipeline advance control should be set to two (M_{11-8} = 2), the pipeline registers and the DM and DL register should be latched (M_5 = M_4 = 1). The result of the multiplication will be stored on the most significant half of the DM register. The timing is shown is Figure 4.

The first cycle after the inputs and function code are loaded, the 32-bit multiplication begins. Two cycles later, the unrounded results are latched into the pipeline register. Two cycles later, the results are at the input of the DM register. One cycle later (t_{P32}), the result is available at the input of the ZREG and can be output on the following cycle.

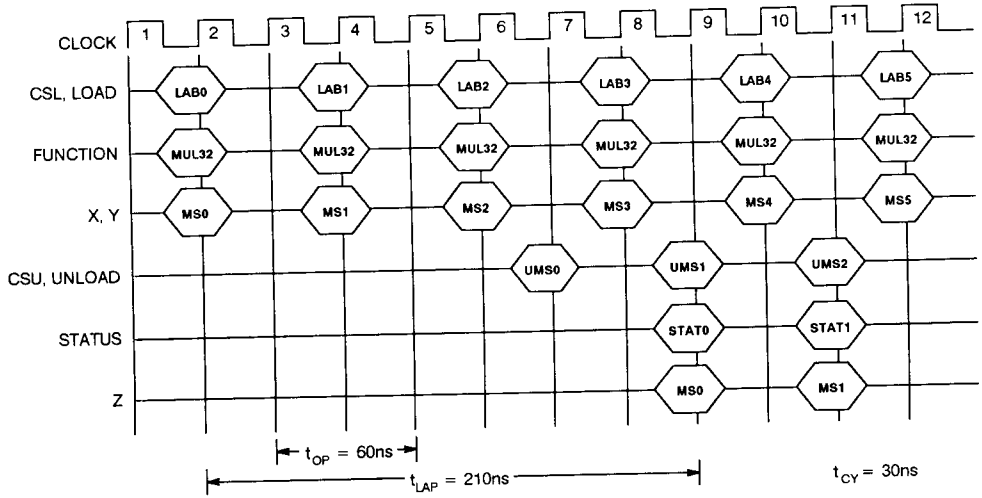


Figure 4. IDT721264 32-bit Pipelined Operation Timing

64-BIT MINIMUM LATENCY FLOW-THROUGH

For minimum latency for 64-bit multiplications, the accumulator timer should be set to clock/2 (M_{7-6} = 1), the pipeline advance control should be set to zero (M_{11-8} = 0) and the pipeline registers and DM and DL registers should be transparent (M_5 = M_4 = 0). The timing is shown is Figure 5.

The first cycle after the inputs and function code are loaded, the 64-bit multiplication begins. Two cycles later, the accumulator stores the partial product of the first pass through the array. Four cycles later (t_{FLOW64}), the result is available at the input of the ZREG and can be output on the following cycle.

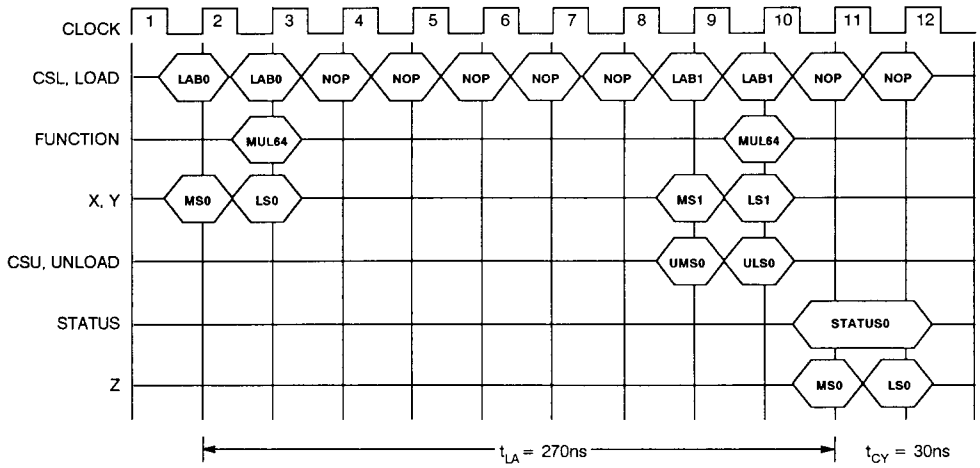


Figure 5. IDT721264 64-bit Flow-through Operation Timing

32-BIT MINIMUM LATENCY FLOW-THROUGH

For minimum latency for 32-bit multiplications, the accumulator timer should be set to clock/2 (M_{7-6} = 1), the pipeline advance control should be set to zero (M_{11-8} = 0) and the pipeline registers and DM and DL registers should be transparent (M_5 = M_4 = 0).

The result of the multiplication will be stored on the most significant half of the DM register. The timing is shown is Figure 6.

The first cycle after the inputs and function code are loaded, the 32-bit multiplication begins. Four cycles later (t_{FLOW32}), the result is available at the input of the ZREG and can be output on the following cycle.

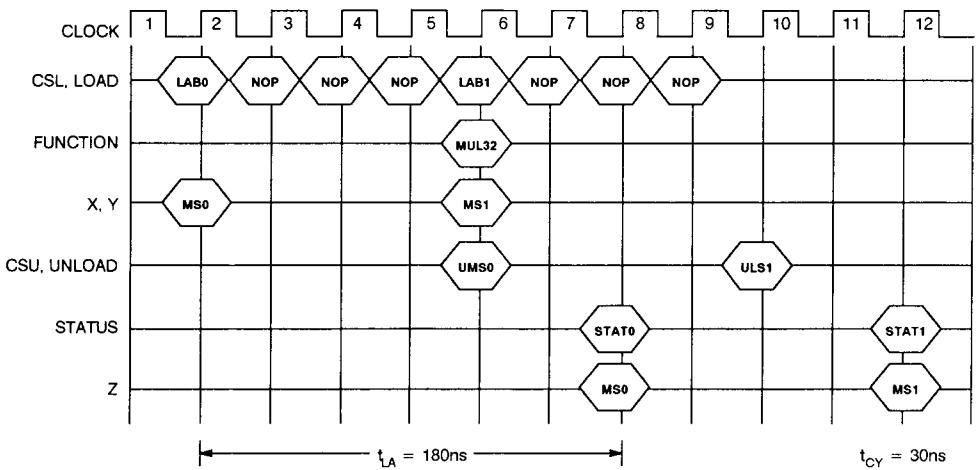


Figure 6. IDT721264 32-bit Flow-through Operation Timing

ALU TIMINGS

32-BIT AND 64-BIT MAXIMUM PIPELINED THROUGHPUT

The ALU has the same throughput for 32-bit and 64-bit operations. For maximum throughput, the pipeline advance control should be set to two (M_{11-8} = 2) and all pipeline registers should be enabled (M_7 = M_6 = M_5 = M_4 = 1). The timing is shown in Figure 7.

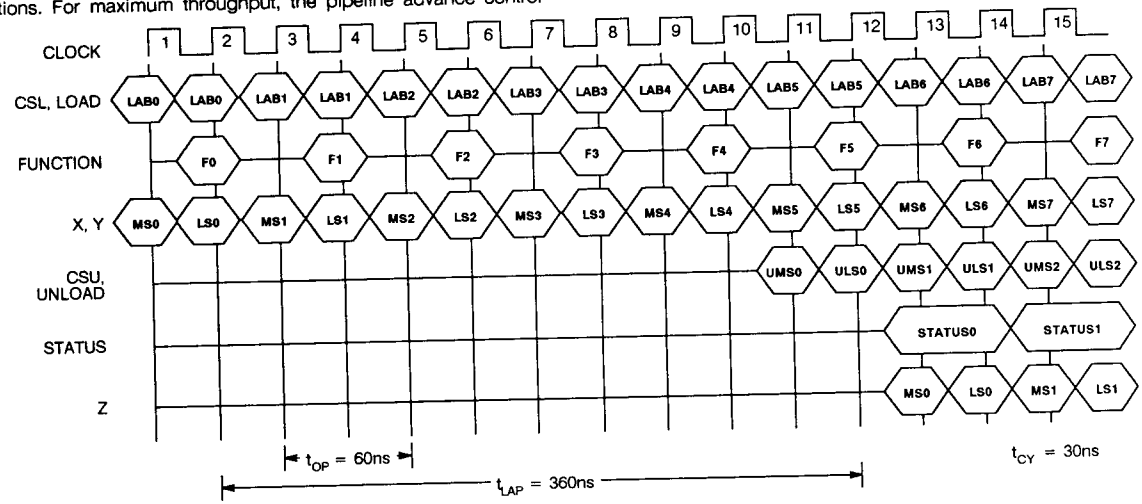


Figure 7. IDT721265 32-bit and 64-bit Pipelined Timing

32-BIT AND 64-BIT MINIMUM LATENCY FLOW-THROUGH

For minimum latency for ALU operations, the pipeline advance control should be set to zero (M_{11-8} = 0) and all pipeline registers should be disabled (M_7 = M_6 = M_5 = M_4 = 0). The timing is shown in Figure 8.

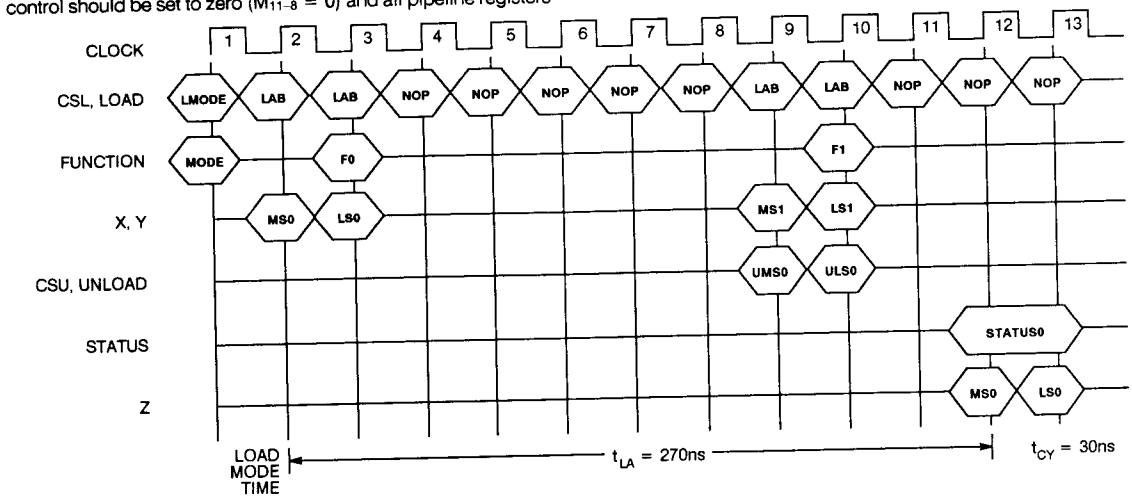
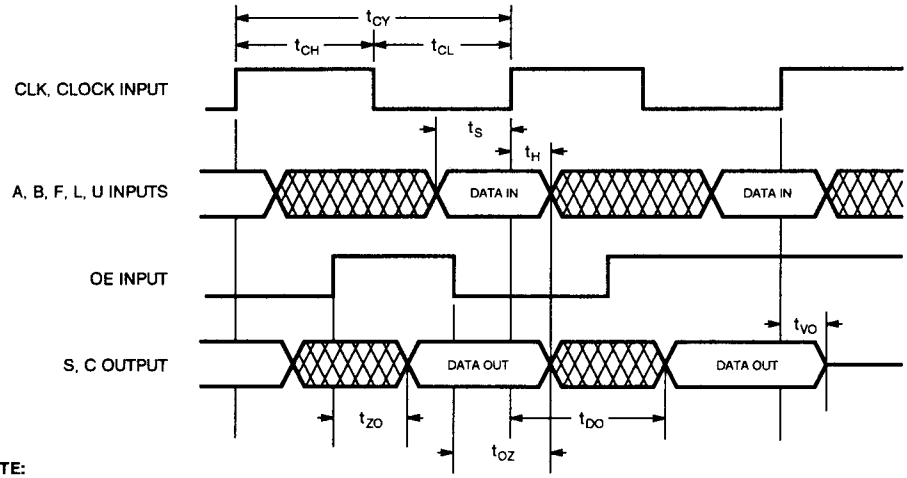


Figure 8. IDT721265 32-bit and 64-bit Flow-through Timing



NOTE:
1. CSU is LOW.

Figure 9. Input/Output Timing (1)

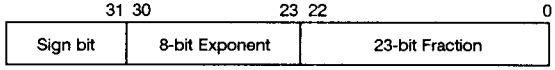
IEEE COMPATIBILITY

The IDT721264 and IDT721265 conform to the IEEE Standard 754, 1895 Version, which specifies floating-point processor data formats, rounding modes and exception handling. Many data formats are specified in the IEEE Standard 754: single precision, double precision, normalized numbers, denormalized numbers, wrapped numbers, zero and infinity. See Table 7. The Gradual Underflow section discusses how denormalized numbers (DNRMs) are handled.

DATA FORMATS

SINGLE PRECISION

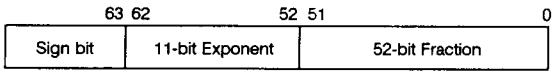
The chip set performs 32-bit and 64-bit IEEE standard floating-point operations. The 32-bit data format has a single sign bit, a 23-bit magnitude fraction field and an 8-bit exponent field in the following format:



Exponents for normalized single precision numbers range from 1 to 254. Exponents of zero and 255 are reserved for special operands. The exponent bias is + 127, which means that the exponent value is e-127. The fraction value is 1.f, where 1 is the hidden bit and f is the fraction. The single precision number can be represented as $(1)^S \times 2^{e-127} \times 1.f$.

DOUBLE PRECISION

The 64-bit data format has a single sign bit, a 52-bit magnitude fraction field and an 11-bit exponent field in the following format:



Exponents for normalized double precision numbers range from 1 to 2046. Exponents of zero and 2047 are reserved for special operands. The exponent bias is + 1023, which means that the exponent value is e-1023. The fraction value is 1.f, where 1 is the hidden bit and f is the fraction. The double precision number can be represented as $(1)^S \times 2^{e-1023} \times 1.f$.

NORMALIZED NUMBERS (NORM)

Most operations are performed on normalized numbers. In normalized single precision numbers where the exponent ranges from 1 (00000001) to 254 (11111110), the fraction is normalized and the hidden bit equals 1. This translates to a decimal number range from 10^{+38} to 10^{-38} for both positive and negative numbers and a precision of 7 decimal places.

In normalized double precision numbers where the exponent ranges from 1 to 2046, the fraction is normalized and the hidden bit equals 1. This translates to a decimal number range from 10^{+307} to 10^{-308} for both positive and negative numbers and a precision of 15 decimal places.

INFINITY

Infinity is defined as an exponent of 1 and a fraction of 0. IEEE Standard 754 defines both positive and negative infinity.

ZERO

Zero is defined as an exponent of 0, the hidden bit of 0 and a fraction of 0. IEEE Standard 754 defines both + 0 and -0.

DENORMALIZED NUMBERS (DNRM)

A denormalized number is defined with an exponent of 0, the hidden bit of 0 and a non-zero fraction. Only the ALU can directly handle denormalized numbers. To multiply two denormalized numbers, the operands must first be wrapped by the ALU, then sent to the multiplier for the multiplication of two wrapped (and normalized) numbers.

WRAPPED NUMBERS (WNRM)

A wrapped number is created by normalizing a denormalized number's fraction and subtracting the number of shifts from the exponent. The fraction is shifted left until the hidden bit is 1. The exponent equals one minus the number of shifts and is in two's complement format. Only the ALU can wrap denormalized numbers and the multiplier can operate on one or two wrapped numbers.

7

UN-NORMALIZED NUMBERS (UNRM)

An un-normalized number results from an addition or multiplication which is smaller than the minimum representable normalized number. An un-normalized number has a wrapped exponent, a hidden bit of 1 and a normalized fraction. The smallest un-normalized number (UNRM.MIN) is the result of multiplying the two smallest denormalized numbers (DNRM.MIN).

NOT-a-NUMBER (NaN)

Not-a-Number is a special data format to flag data overflow or underflow, uninitialized operands and invalid operations (i.e., 0 x ∞). Not-a-Number has an exponent of all 1s and a non-zero fraction.

TABLE 7. IEEE SINGLE PRECISION FORMATS SUPPORTED BY THE IDT721264 AND IDT721265

OPERAND	EXPONENT	FRACTION	HIDDEN BIT	VALUE
NAN	255	ANY	N/A	NONE
INFINITY	255	ALL 0's	1	$(-1)^S \infty$
NORM.MAX	254	ALL 1's	1	$(-1)^S \times 2^{127} \times (2)$
NORM	1 to 254	ANY	1	$(-1)^S \times 2^{e-127} \times (1.f)$
NORM.MIN	1	ALL 0's	1	$(-1)^S \times 2^{-126} \times (1)$
DNRM.MAX	0	ALL 1's	0	$(-1)^S \times 2^{-126}$
DNRM	0	ANY	0	$(-1)^S \times 2^{-126} \times (0.f)$
DNRM.MIN	0	000...01	0	$(-1)^S \times 2^{-126} 2^{-23}$
WNRM.MAX	0	ALL 1's	1	$(-1)^S \times 2^{-126}$
WNRM	0 to (-22)	ANY	1	$(-1)^S \times 2^{e-127} \times (1.f)$
WNRM.MIN	-22	ALL 0's	1	$(-1)^S \times 2^{-149}$
UNRM.MAX	0	ALL 1's	1	$(-1)^S \times 2^{-126}$
UNRM. MIN	-171	ALL 0's	1	$(-1)^S \times 2^{-298}$
ZERO	0	ALL 0's	1	$(-1)^S$ 0

ROUNDING MODES

The chip set supports the four IEEE Standard 754 rounding modes: round to nearest, round toward zero, round toward plus infinity and round toward minus infinity. Biased rounding or unbiased rounding may occur. Biased rounding introduces a small offset in the direction of the bias. IEEE Standard 754 specifies positive bias, negative bias and bias toward zero. Unbiased rounding rounds toward the nearest representable number. If a number is halfway between two representable numbers, the number is rounded towards the nearest even number which averages the rounding up and down.

ROUND TO NEAREST (RN)

The result is rounded to the nearest representable number. If a number is halfway between two representable numbers, the number is rounded towards the nearest even number.

ROUND TOWARDS ZERO (RZ)

The result is rounded to the nearest representable number not greater in magnitude than the number.

ROUND TOWARD PLUS INFINITY (RP)

The result is rounded to the nearest representable number not less than the number.

ROUND TOWARD MINUS INFINITY (RM)

The result is rounded to the nearest representable number not greater than the number.

If the result of an operation is less than the minimum representable number, the underflow condition exists and is handled differently in the multiplier and ALU. In the Fast mode the underflow result is set to zero for both the multiplier and ALU. In the IEEE mode, the multiplier will not round the underflow result but will wrap it. The inexact status bit, S_0, is one if any of the truncated bits contains a one. The ALU can unwrap the result, using the "UNWRAP EXACT" or "UNWRAP INEXACT" depending on the value of S_0. The ALU status register will show whether the result is exact or inexact.

EXCEPTION HANDLING

The chip set performs exception handling according to the IEEE Standard 754. The status bits are pipelined synchronously with the operands and partial results. The status bits are stored in the STREG1 when the result is clocked into the output register until the rising edge of the next clock cycle.

The result of an ALU Compare operation is shown on the status output. A Compare result supersedes an exception status.

INEXACT (INX)

When the result of an ALU or multiplier operation losses accuracy, an Inexact status is shown. The ALU computes more that 23 fraction bits in a single precision and 53 bits in double precision. If any of the lesser significant bits equals 1, then an INX is signaled. In floating-point to fixed-point conversions, any loss of accuracy will signal INX. For normalized number operations, INX will not be signaled.

UNDERFLOW (UNF)

Underflow is asserted if a rounded result is less than the minimum normalized number. If the result is exactly zero, UNF will not be asserted.

OVERFLOW (OVF)

Overflow is asserted if a rounded result is greater than the maximum normalized number. The result is either infinity or the largest representable number and is a factor of the Round mode.

RESULT	ROUNDING MODE
+NORM.MAX (positive)	RM or RZ
-NORM.MAX (negative)	RP or RZ
+∞ (positive)	RN or RP
-∞ (negative)	RN or RM

Overflow is also asserted if the result of a floating-to-fixed operation overflows the 32-bit format.

INVALID OPERATION (INV)

The following are Invalid Operations (INV):

- One of the operands is a NaN
- 0 x ∞
- +∞ - +∞
- -∞ + +∞
- +∞ + -∞
- -∞ - -∞

OPERATIONS

The following tables represent different results obtained from different operand formats and rounding modes. Tables for both IEEE and Fast modes are shown. All results are in the "Result-Exception Status" format.

7

TABLE 8. FLOATING-POINT ADD/SUBTRACT ("FAST" MODE)

A/B	ZERO	DNRM	NRM	INF	NAN
ZERO	OK-ZERO(3)	OK-ZERO(3)	OK-NRM	OK-INF	INV-NAN
DNRM	OK-ZERO(3)	OK-ZERO(3)	OK-NRM	OK-INF	INV-NAN
NRM	OK-NRM	OK-NRM	OK-ZERO UNF-ZERO OK-NRM OK-INF(4)	OK-INF	INV-NAN
INF	OK-INF	OK-INF	OK-INF	OK-INF(1) INV-NAN(2)	INV-NAN
NAN	INV-NAN	INV-NAN	INV-NAN	INV-NAN	INV-NAN

NOTES:

1. +INF+INF → +INF
   -INF-INF → -INF
2. +INF-INF → NAN
   -INF+INF → NAN
3. +ZERO+ZERO → +ZERO (RN, RZ, RP, RM)
   -ZERO-ZERO → -ZERO (RN, RZ, RP, RM)
   +ZERO-ZERO → +ZERO (RN, RZ, RP)
   +ZERO-ZERO → -ZERO (RM)
   -ZERO+ZERO → +ZERO (RN, RZ, RP)
   -ZERO+ZERO → -ZERO (RM)
4. OVF will produce INF or MAX.NRM, depending upon the rounding mode
   +NRM.MAX if [(RM, RZ) AND (TRESULTS is +)]
   +NRM.MAX if [(RM, RZ) AND (TRESULTS is -)]
   +INF if [(RN, RP) AND (TRESULT is +)]
   -INF if [(RN, RM) AND (TRESULT is +)]

TABLE 9. FLOATING-POINT MULTIPLICATION ("FAST" MODE)

A/B	ZERO	DNRM	NRM	INF	NAN
ZERO	OK-ZERO	OK-ZERO	OK-NRM	INF-NAN	INV-NAN
DNRM	OK-ZERO	OK-ZERO	OK-NRM	OK-INF	INV-NAN
NRM	OK-ZERO	OK-ZERO	UNF-ZERO OK-NRM OK-INF(1)	OK-INF	INV-NAN
INF	INF-NAN	OK-INF	OK-INF	OK-INF INV-NAN	INV-NAN
NAN	INV-NAN	INV-NAN	INV-NAN	INV-NAN	INV-NAN

NOTES:
1. OVF will produce INF or MAX.NRM, depending upon the rounding mode
   + NRM.MAX if [(RM, RZ) AND (TRESULTS is +)]
   + NRM.MAX if [(RM, RZ) AND (TRESULTS is -)]
   + INF if [(RN, RP) AND (TRESULT is +)]
   -INF if [(RN, RM) AND (TRESULT is -)]

TABLE 10. FLOATING-POINT ADD/SUBTRACT (IEEE MODE)

A/B	ZERO	DNRM	NRM	INF	NAN
ZERO	OK-ZERO(3)	UNF-WNRM	OK-NRM	OK-INF	INV-NAN
DNRM	UNF-WNRM	UNF-WNRM OK-ZERO(3) OK-NRM	OK-NRM UNF-WNRM	OK-INF	INV-NAN
NRM	OK-NRM	UNF-WNRM OK-NRM OK-INF(4)	OK-ZERO UNF-WNRM OK-NRM OK-INF(4)	OK-INF	INV-NAN
INF	OK-INF	OK-INF	OK-INF	OK-INF(1) INV-NAN(2)	INV-NAN
NAN	INV-NAN	INV-NAN	INV-NAN	INV-NAN	INV-NAN

NOTES:
1. +INF+INF → +INF
   -INF-INF → -INF
2. +INF-INF → NAN
   -INF+INF → NAN
3. +ZERO+ZERO → +ZERO (RN, RZ, RP, RM)
   -ZERO-ZERO → -ZERO (RN, RZ, RP, RM)
   +ZERO-ZERO → +ZERO (RN, RZ, RP)
   +ZERO-ZERO → -ZERO (RM)
   -ZERO+ZERO → +ZERO (RN, RZ, RP)
   -ZERO+ZERO → -ZERO (RM)
4. OVF will produce INF or MAX.NRM, depending upon the rounding mode
   + NRM.MAX if [(RM, RZ) AND (TRESULTS is +)]
   + NRM.MAX if [(RM, RZ) AND (TRESULTS is -)]
   + INF if [(RN, RP) AND (TRESULT is +)]
   -INF if [(RN, RM) AND (TRESULT is +)]

TABLE 11. FLOATING-POINT MULTIPLICATION (IEEE MODE)

A/B	ZERO	DNRM	NRM	INF	NAN
ZERO	OK-ZERO	OK-ZERO	OK-ZERO	INF-NAN	INV-NAN
DNRM	OK-ZERO	OK-ZERO	DIN-ZERO	OK-INF	INV-NAN
NRM	OK-ZERO	OK-ZERO	UNF-ZERO OK-NRM OK-INF(1)	OK-INF	INV-NAN
INF	INF-NAN	OK-INF	OK-INF	OK-INF	INV-NAN
NAN	INV-NAN	INV-NAN	INV-NAN	INV-NAN	INV-NAN

NOTES:
1. OVF will produce INF or MAX.NRM, depending upon the rounding mode
   + NRM.MAX if [(RM, RZ) AND (TRESULTS is +)]
   + NRM.MAX if [(RM, RZ) AND (TRESULTS is -)]
   + INF if [(RN, RP) AND (TRESULT is +)]
   -INF if [(RN, RM) AND (TRESULT is +)]

TABLE 12. FLOATING-POINT COMPARE STATUS

INPUT A						INPUT B					
	U	U	U	U	U	NAN	U	U	U	U	U
	U	L	L	L	L	+INF	L	L	L	E	U
	U	L	L	L	L	+NRM	L	L	Note 1	G	U
	U	L	L	L	L	+DNRM	L	Note 1	G	G	U
	U	L	L	L	E	+ZERO	E	G	G	G	U
	NAN	-INF	-NRM	-DNRM	-ZERO		+ZERO	+DNRM	+NRM	+INF	NAN
	U	L	L	L	E	-ZERO	E	G	G	G	U
	U	L	L	Note 1	G	-DNRM	G	G	G	G	U
	U	L	Note 1	G	G	-NRM	G	G	G	G	U
	U	E	G	G	G	-INF	G	G	G	G	U
	U	U	U	U	U	NAN	U	U	U	U	U

NOTE:
1. Equal, less than or greater than
   E – A = B
   L – A < B
   G – A > B
   U – Unordered

TABLE 13. CONVERT SINGLE TO DOUBLE

F32 → F64					
F32 OPERAND		F64 RESULT		STATUS	COMMENTS
077777	177777	077777 177777	177777 177777	12	NaN
077600	000000	077760 000000	000000 000000	1	+INF
077577	000000	043757 160000	177777 000000	2	+MAX.NRM
037600	000000	037760 000000	000000 000000	2	+1
000200	000000	034020 000000	000000 000000	2	+MIN.NRM
000177	177777	033757 140000	177777 000000	2	+MAX.DNRM
000000	000001	033240 000000	000000 000000	2	+MIN.DNRM
000000	000000	000000 000000	000000 000000	0	+ZERO
177600	000000	177760 000000	000000 000000	1	-INF
177577	177777	143757 160000	177777 000000	2	-MAX.NRM
140000	000000	140000 000000	000000 000000	2	-2
100200	000000	134020 000000	000000 000000	2	-MIN.NRM
100177	177777	133757 140000	177777 000000	2	-MAX.DNRM
100000	000001	133200 000000	000000 000000	2	-MIN.DNRM
100000	000000	100000 000000	000000 000000	0	-ZERO

TABLE 14. CONVERT DOUBLE TO SINGLE [1]

F64 → F32					
F64 OPERAND		F32 RESULT		STATUS	COMMENTS
077777 177777	177777 177777	077777	177777	12	NaN
077760 000000	000000 000000	077600	000000	1	+INF
077757 000000	177777 000000	077600	000000	5	+MAX.NRM OPERAND
043757 170000	177777 000000	077600	000000	5	+OVF RESULT
043757 160000	177777 000000	077577	177777	2	+MAX.NRM RESULT
043757 177777	177777 177777	077400	000000	3	+INEXACT
037760 000000	000000 000000	037600	000000	2	+1
034020 000000	000000 000000	000200	000000	2	+MIN.NRM RESULT
034017 160005	177777 000000	000200	000000	3	+MIN.NRM RESULT
033757 140000	177777 000000	000177	177777	6	+MAX.DNRM RESULT
033240 000000	000000 000000	000000	000001	6	+MIN.DNRM RESULT
033230 000000	000000 000000	000000	000001	7	+MIN.DNRM RESULT
033220 000000	000000 000000	000000	000000	7	+ZERO RESULT
000020 000000	000000 000000	000000	000000	7	+MIN.NRM OPERAND
000017 177777	177777 177777	000000	000000	7	+MAX.DNRM OPERAND
000000 000000	000000 000000	000000	000000	0	+ZERO
177760 000000	000000 000000	177600	000000	1	-INF
177757 177777	177777 177777	177600	000000	5	-MAX.NRM OPERAND
100000 000000	000000 000000	100000	000000	0	-ZERO

NOTE:
1. Round Mode = RN

7

TABLE 15. DOUBLE FLOAT

I32 → F64					
I32 OPERAND		F64 RESULT		STATUS	COMMENTS
077777	177777	040737 177700	177777 000000	2	+MAX OPERAND
077777	177776	040737 177600	177777 000000	2	–
000000	000002	040000 000000	000000 000000	2	+2
000000	000001	037760 000000	000000 000000	2	+1
000000	000000	000000 000000	000000 000000	0	ZERO
177777	177777	137760 000000	000000 000000	2	-1
177777	177776	140000 000000	000000 000000	2	-2
100000	000002	140737 177600	177777 000000	2	–
100000	000001	140737 177700	177777 000000	2	–
100000	000000	140740 000000	000000 000000	2	-MAX OPERAND

TABLE 16. SINGLE FLOAT [1]

I32 → F32					
I32 OPERAND		F32 RESULT		STATUS	COMMENTS
077777	177777	047400	000000	3	+MAX OPERAND
077777	177700	047400	000000	3	–
077777	177600	047377	177777	2	–
000000	000002	040000	000000	2	+2
000000	000001	037600	000000	2	+1
000000	000000	000000	000000	0	ZERO
177777	177777	137600	000000	2	-1
177777	177776	140000	000000	2	-2
100000	000000	147400	000000	2	-MAX OPERAND

NOTE:
1. Round Mode = RN

TABLE 17. DOUBLE FIX [1]

F64 → I32					
F64 OPERAND		I32 RESULT		STATUS	COMMENTS
077777 177777	177777 177777	077777	177777	12	NAN
077760 000000	000000 000000	077777	177777	5	+INF
077757 177777	177777 177777	077777	177777	5	+MAX.NRM OPERAND
040737 177740	177777 000000	077777	177777	5	+OVF
040737 177700	177777 000000	077777	177777	2	+MAX RESULT
040000 000000	000000 000000	000000	000002	2	+2
377760 000000	000000 000000	000000	000001	2	+1
037750 000000	000000 000000	000000	000001	3	+1
037740 000000	000000 000000	000000	000000	3	ZERO
000020 000000	000000 000000	000000	000000	3	+MIN.NRM
000000 000000	000000 000001	000000	000000	3	+MIN.NRM
000000 000000	000000 000000	000000	000000	0	+ZERO
177760 000000	000000 000000	100000	000000	5	-INF
177757 177777	177777 000000	100000	000000	5	-MAX.NRM OPERAND
140740 000100	000000 000000	100000	000000	5	-OVF
140740 000000	000000 000000	100000	000000	2	-MAX RESULT
140000 000000	000000 000000	177777	177776	2	-2
137760 000000	000000 000000	177777	177777	2	-1
137750 000000	000000 000000	177777	177777	3	-1
137740 000000	000000 000000	000000	000000	3	ZERO
100200 000000	000000 000000	000000	000000	3	-MIN.NRM
100000 000000	000000 000001	000000	000000	3	-MIN.DNRM
100000 000000	000000 000000	000000	000000	0	-ZERO

NOTE:
1. Round Mode = RN

TABLE 18. SINGLE FIX [1]

F32 → I32					
F32 OPERAND		I32 RESULT		STATUS	COMMENTS
077777	177777	077777	177777	12	NAN
077600	000000	077777	177777	5	+INF
077577	177777	077777	177777	5	+MAX.NRM
047400	000000	077777	177777	5	+OVF
047377	177777	077777	177600	2	+MAX RESULT
040000	000000	000000	000002	2	+2
037600	000000	000000	000001	2	+1
037500	000000	000000	000001	3	+1
000200	000000	000000	000000	3	+MIN.NRM
000000	000001	000000	000000	3	+MIN.DNRM
000000	000000	000000	000000	2	+ZERO
177600	000000	100000	000000	5	-INF
177577	177777	100000	000000	5	-MAX.NRM
147400	000001	100000	000000	5	-OVF
147400	000000	100000	000000	2	-MAX RESULT
140000	000000	177777	177776	2	-2
137600	000000	177777	177777	2	-1
137500	000000	177777	177777	3	-1
137400	000000	000000	000000	3	-ZERO
100200	000000	000000	000000	3	-MIN.NRM
100000	000001	000000	000000	3	-MIN.DNRM
100000	000000	000000	000000	0	-ZERO

NOTE:
1. Round Mode = RN

TABLE 19. DOUBLE WRAP DENORMALIZED VALUE

D64 → W64					
D64 OPERAND		W64 RESULT		STATUS	COMMENTS
000000 000000	000000 000001	076320 000000	000000 000000	2	+MIN.DNRM
000010 000000	000000 000000	000000 000000	000000 000000	2	–
000017 177777	177777 177777	000017 177777	177777 177776	2	+MAX.DNRM

TABLE 20. SINGLE WRAP DENORMALIZED VALUE

D32 → W32					
D32 OPERAND		W32 RESULT		STATUS	COMMENTS
100000	000001	172400	000000	2	-MIN.DNRM
100100	000000	100000	000000	2	–
100177	177777	100177	177776	2	-MAX.DNRM

TABLE 21. DOUBLE UNWRAP EXACT VALUE [1]

U64 → D64					
U64 OPERAND		D64 RESULT		STATUS	COMMENTS
000017 177777	177777 177777	000020 000000	000000 000000	3	+MIN.NRM RESULT
000000 000000	000000 000000	000010 000000	000000 000000	6	+UNF
077777 177777	177777 177777	000010 000000	000000 000000	7	–
076320 000000	000000 000000	000000 000000	000000 000001	6	+MIN.DNRM RESULT
076317 177777	177777 177777	000000 000000	000000 000001	7	–

NOTE:
1. Round Mode = RN

TABLE 22. SINGLE UNWRAP EXACT VALUE [1]

U32 → D32					
U32 OPERAND		D32 RESULT		STATUS	COMMENTS
100177	177777	100200	000000	3	–
100000	000000	100100	000000	6	–
177777	177777	100100	000000	7	–
172400	000000	100000	000001	6	–
172577	177777	100000	000002	7	–

NOTE:
1. Round Mode = RN

7

GRADUAL UNDERFLOW

The minimum normalized number has an exponent of one and a fraction field of zero. Zero has an exponent of zero and a fraction field of all zeros. This gives users the ability to deal with numbers between NORM.MIN and ZERO. These numbers are known as denormals. Their format is given in the number format section. The IEEE standard has specified gradual underflow as the way to handle denormals. Many of the features of the IDT721264 and IDT721265 are included to deal with denormals in a manner consistent with IEEE Standard 754. Since denormals are very close to zero, many applications can substitute zero for a denormal without a significant loss of accuracy. For these applications, a "FAST" mode is included which substitutes zero for all denormalized inputs to the IDT721264 and IDT721265. Zero is also inserted for all UNRM outputs in "FAST" mode.

For all arithmetic operations, the IDT721265 handles denormalized inputs directly as it would handle and other number.

Unfortunately, a floating-point multiplier must either operate exclusively on normalized numbers or suffer large cost and performance penalties in dealing directly with denormals. A normalized format that yields an equivalent to a given denormalized number is the wrapped format. The number format table shows the equivalence of wrapped and denormalized numbers. To translate a denormalized number to a wrapped number, the fraction is normalized (shifted up so that a one is in the hidden bit) and one is subtracted from the exponent for every position shifted. The IDT721264 can multiply correctly either two wrapped numbers or a wrapped and a normalized number. To understand the full procedure, consider the following case.

Assume one of the two input operands to the IDT721264 is a denormalized number. Four cycles after the input, the denorm exception is flagged. The denormalized operand must then be sent to the IDT721265 to be wrapped. Once wrapped, the operand can be sent back to the IDT721264 for multiplication. The result o the multiplication will either be a normalized number or a UNRM.

If the result is a UNRM, status bit S_0 indicates either UNF (if all the truncated bits are equal to zero) or UNF-INX (if any of the truncated bits is equal to one).

No rounding will occur regardless of the rounding mode specified.

The underflowed number may then be sent to the IDT721265 for "unwrapping". To unwrap a number, the fraction field is shifted right and the exponent incremented by one for each shift position. Status bit S_0 must be used to conditionally execute the "UNWRAP INEXACT" or "UNWRAP EXACT" instruction. The rounding must be performed in the ALU. The unwrapping may have three possible results:

RESULT	EXCEPTION	COMMENT
DNRM	UNF	When the denormalized result is exact. Note that this result is possible only if the UNWRAP EXACT instruction is possible (i.e., both the input and the result must be exact).
DNRM	UNF-INX	If either the "UNWRAP INEXACT" instruction is executed or if the result of the "UNWRAP INEXACT" instruction is inexact.
ZERO	INX	The result is zero, but the unwrapping has resulted in the loss of precision.

ORDERING INFORMATION