

- 
- Weitek Business: Solutions To Floating Point Intensive Problems
  - 500 Man-years Experience In Floating Point Chip Design & Marketing

Ken Stanley  
14 July 88

---

WEITEK



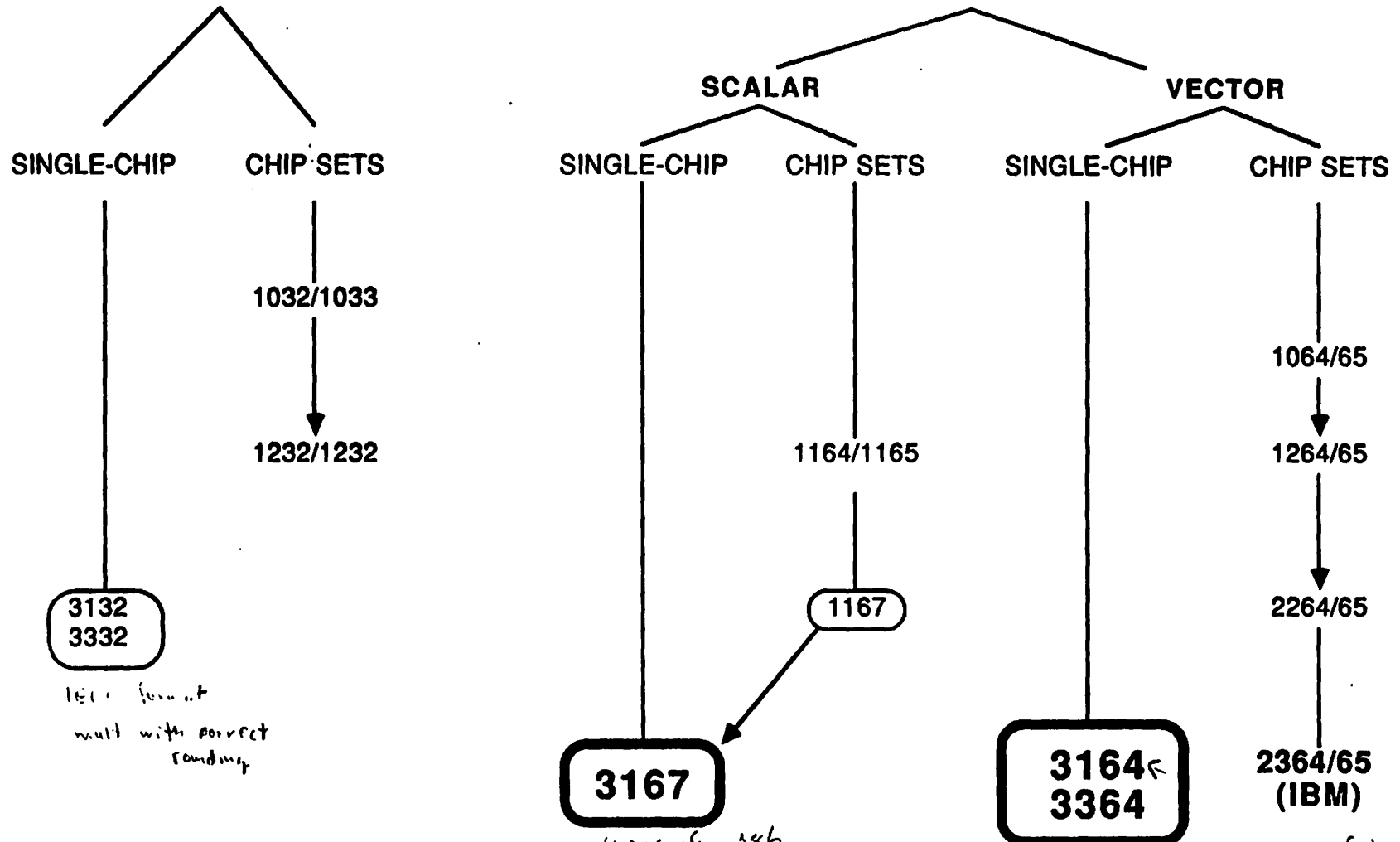
- 
- Weitek Makes Chips (Apples)
  - IEEE 754 Is A System Level Spec (Oranges)



# WEITEK FLOATING-POINT PRODUCT LINE

## 32-BIT

## 64-BIT



 Contains register file

---

All Generalizations Are False

---

WEITEK



---

## ALL WEITEK CHIPS

- Conform To IEEE Format (Except 2364/2365)  
*IEEE format*
- Allow Control Of Overflow Trap
- Do Not Support 80-bit Arithmetic
- Do Not Support De-Norms In Hardware
- Have Multiplier Arrays

*array is  
needed for*



---

## **WEITEK SINGLE CHIP SOLUTIONS**

- Attack System Cost
- Reduce Time To Market
- ansi C
- fortran 77
- CMOS
- 32 Registers
- 3132, 1167/3167, 3164/3364

---

**WEITEK**



---

## 3132

- 32-bits Only
- Graphics/DSP
- No Denorm, NaN or Infinity Support
- *multiply accumulate* **MAC** Operations Does Not Obey IEEE Rounding *it add  $\neq 0$*
- Ignores Underflow, Invalid, Inexact.  
Divide By Zero Gives Overflow
- Divide, and SQRT In Software (not IEEE)
- No Remainder Operation
- ~~All Four Rounding Modes~~

---

## 1167/3167

- 32 + 64-bit IEEE Formats
- 386 Coprocessor
- User Controllable Traps On All Five Exceptions
- Cannot Restart After An Exception
- No Denorm Support (Hardware Or Software)
- All Four Rounding Modes
- ADD, SUB, MUL, DIV On Chip
- SQRT On 3167
- Remainder In Software
- Transcendental Library

All chips have  
32 registers of  
(16 double) single

Can't restart  
operands may  
have been modified  
trap doesn't say  
where trap occurred

---

WEITEK





---

## **WEITEK 3164/3364**

- First Single Chip To Make Possible Complete Compliance To The IEEE 754 Spec In A Pipelined Environment.
- Trap Handler Supplied To OEM's.

**WTL 3164 SINGLE-PORT  
AND WTL 3364 THREE-PORT  
64-BIT FLOATING-POINT  
DATA PATH UNITS**

**3X64 PRODUCT UPDATE**

**WEITEK**



# **OUTLINE**

**ARCHITECTURE AND FEATURES**

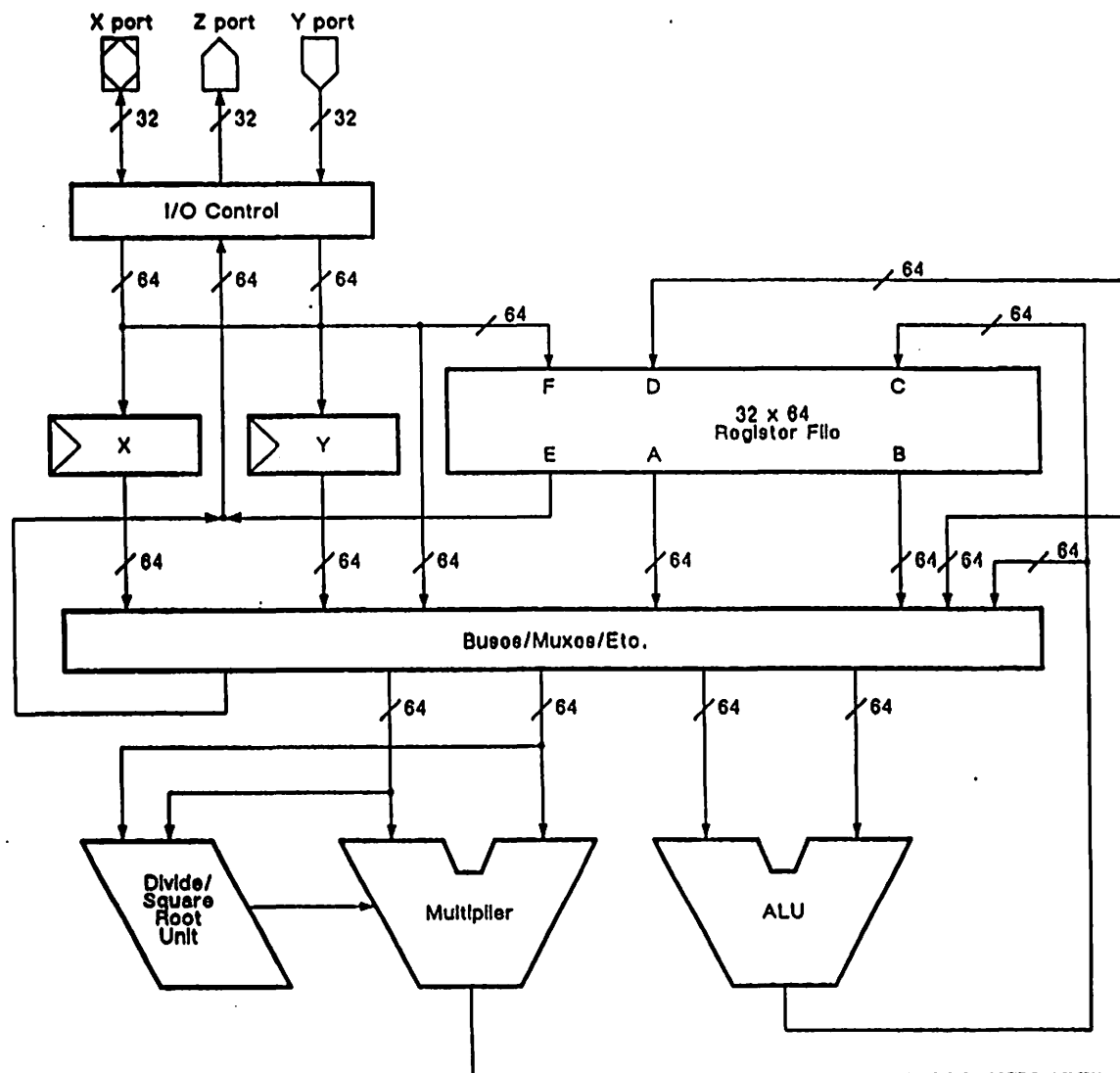
**IEEE COMPLIANCE**

**PERFORMANCE**

**STATUS AND SCHEDULE**

**FUTURES**

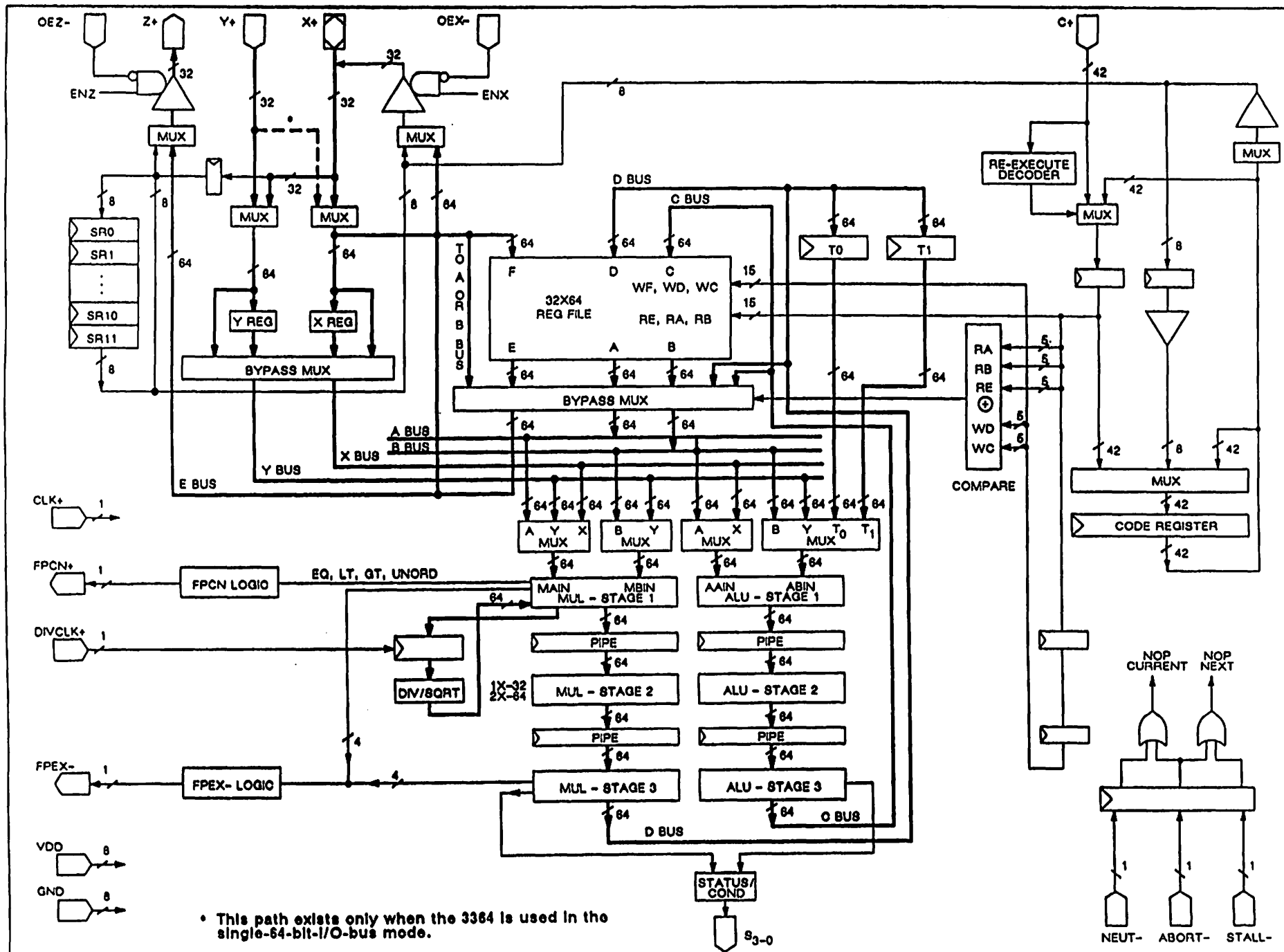




3X64 PRODUCT UPDATE

WEITEK

# WTL 3164 / 3364 BLOCK DIAGRAM



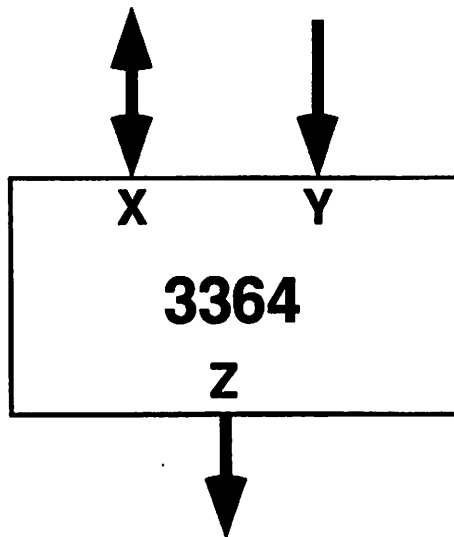
## **ARCHITECTURE**

- **64-BIT IEEE FLOATING-POINT DATA PATH**
- **THREE INDEPENDENT ARITHMETIC UNITS CAN OPERATE IN PARALLEL**
  - **64-BIT ALU**
  - **64-BIT MULTIPLIER**
  - **64-BIT DIVIDE/SQRT UNIT**
- **REGISTER FILE: SIX-PORT, 32-DEEP BY 64-BIT-WIDE**
  - **THREE READ PORTS**
  - **THREE WRITE PORTS**
  - **CAN BE BYPASSED ON LOADS, STORES, AND REGISTER-TO-REGISTER OPERATIONS**
- **INDEPENDENT LOAD/STORE**
- **SIX MAJOR INTERNAL 64-BIT BUSES**
- **EXTENSIVE STATUS AND CONTROL LOGIC**

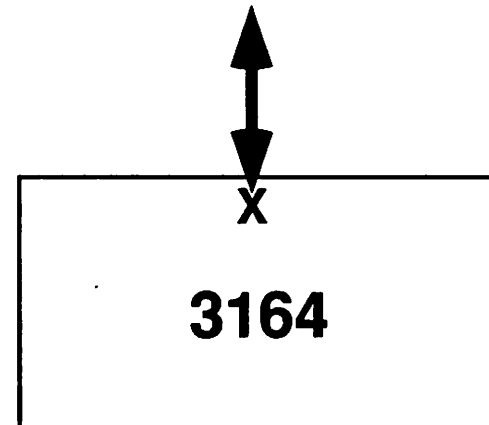
## ARCHITECTURE CONT'D

- FLEXIBLE I/O STRUCTURE

- X PORT: I/O
- Y PORT: INPUT ONLY
- Z PORT: OUTPUT ONLY



- THREE 32-BIT PORTS
- SINGLE 64-BIT I/O PORT



- SINGLE 32-BIT I/O PORT



## ARCHITECTURE CONT'D

- TWO-CYCLE REGISTER-TO-REGISTER LATENCY

- SINGLE-CYCLE THROUGHPUT FOR

- $X_i * Y_i \rightarrow Z_i$       and / or       $R_a + R_b \rightarrow R_c$

- $R_a * X_i + Y_i \rightarrow Z_i/R_c$

- $\sum X_i * Y_i$                       (requires 3364)

- $\sum (X_i + Y_i)$                       (requires 3364)

- DIVIDE/SQRT LATENCY

	<u>DIVIDE</u>	<u>SQRT</u>
64-BIT	17	30
32-BIT	10	16

- DIV/SQRT CAN OVERLAP WITH MULTIPLIER AND/OR ALU OPERATION



## **FULL FUNCTION**

- **64- AND 32-BIT FLOATING-POINT AND 32-BIT INTEGER INSTRUCTIONS**
  - **MULTIPLY, ADD, MULTIPLY-ADD, DIV, SQRT, ETC.**
- **64-BIT LOGICAL**
- **ABSOLUTE VALUE**
- **COMPARE**
- **MIN/MAX**
- **FORMAT CONVERSION**

## **FULLY INTERRUPTIBLE**

- **NEUT, STALL, ABORT**



## **IEEE COMPLIANCE**

- **FULL IEEE SUPPORT IN PIPELINED ENVIRONMENT**
  - **INCLUDING DIVIDE AND SQUARE ROOT**
- **ALL FOUR ROUNDING MODES**
- **DENORMALIZED NUMBER SUPPORT**
  - **FAST MODE IF DNRN SUPPORT NOT REQUIRED**
- **FULL STATUS AND CONDITION SUPPORT**
  - **FPEX PIN SIGNALS OCCURENCE OF ENABLED EXCEPTION**
  - **EXCEPTIONS**
    - **SOURCE: NAN, DNRN, DVZ, INV**
    - **RESULT: OVF, IOVF, UNF, INX**
  - **FLOATING-POINT CONDITION PIN REPORTS RESULTS OF COMPARE OPERATIONS: EQ, LT, GT, UNORDERED**

## **SIMPLE PROGRAMMING MODEL**

- **REGISTER-BASED PROGRAMMING MODEL**
- **MATCHED LATENCY FOR ALL OPERATIONS**
  - **EXCEPT DIVIDE AND SQRT**
- **BOTH SOURCES AND DESTINATIONS SPECIFIED ONLY ONCE -- WHEN INSTRUCTION IS ISSUED**
  - **DESTINATION ADDRESSES ARE DELAYED AUTOMATICALLY**
- **ALL INFORMATION NECESSARY FOR EXCEPTION HANDLING INCLUDED ON-CHIP**
  - **DEDICATED STATUS REGISTERS STORE THE STATUS AND REG FILE DESTINATION ADDRESSES**

## HIGH-LEVEL LANGUAGE SUPPORT

- AVAILABLE FOR BOTH 3164 AND 3364 WHEN USED IN XL SYSTEM
  - HLL COMPILERS (C, FORTRAN)
  - DEVELOPMENT SYSTEM
  - OTHER SUPPORT: DEBUGGERS, SIMULATORS
- PERFORMANCE OF XL-3164 IN XL8164 SYSTEM

	<u>-100</u>	<u>-080</u>
• LINPACK (MFLOPS) (HAND-CODED BLAS)		
SINGLE OR DOUBLE PRECISION	4.7	5.8
• WHETSTONES (MWHETS)		
SINGLE OR DOUBLE PRECISION	8.7	10.9
• DHRYSTONES	11,834	14,793



---

## WEITEK 3164/3364

*Pipe is 7 deep  
an can do mult & add  
in parallel*

- 32 + 64-Bit IEEE Formats
- Denorms Handled In Software
- User Controlled Traps On All Five IEEE Exceptions
- Status Regs Allow Recovery From Exceptions
- ADD, SUB, MUL, DIV, SQRT On Chip
- DREM In Software

---

WEITEK



# 3164/3364 EXCEPTION HANDLING

- Source Exceptions:

(Invalid, Divide-By-Zero)

$\infty / \infty$

- Stop Current Operation

- Denorm has to be handled in software

- Destination Exceptions

(Underflow, Overflow, Inexact)

- Operation Completes Before Trapping

- Correctly Rounded Result Supplied To User Trap Handler

mult  
 (1) wrap denorm  
 (2) wrap other input  
 (3) multiply  
 (4) underflows

product of 2 binary  
 denormed numbers is 0.

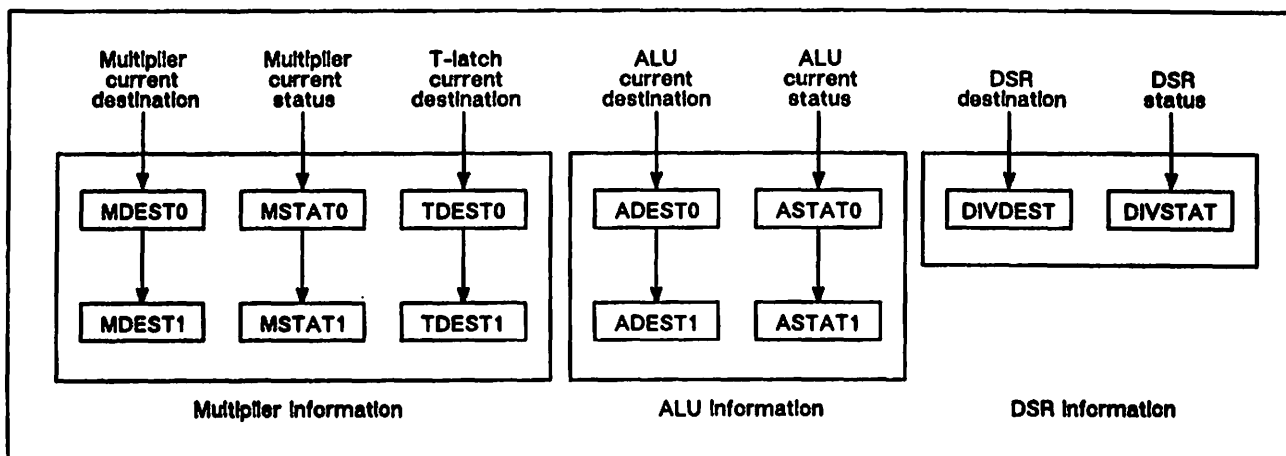
add/sub  
 wrap the denorm  
 R1 as input  
 then other  
 Do the add  
 in bits

Kahan says that denorm could  
 be done in hardware

No hardware in bit 11  
 ADD R0, R1, R2  
 MUL R2  
 software must  
 insert exp here

WEITEK





Operations' destination/status information and their storage in the status register

SR#	BIT#								COMMENTS
	7	6	5	4	3	2	1	0	
SR0	Multiplier Latency	FPEX-Sticky	0	0	Internal NEUT-on	Rounding Mode		Fast Mode	Modes
SR1	0 XL Software Underflow	Bypass on	FPEX-Delay	I/O Mode					Modes
SR2	NaN EN	INV EN	DVZ EN	DNRM EN	OVF EN	UNF EN	INX EN	IOVF EN	Trap Enables
SR3	NaN	INV	DVZ	DNRM	OVF	UNF	INX	IOVF	Sticky Bits
SR4	0	TDEST0		MDEST0					Destination
SR5	0	0	0	ADEST0					Destination
SR6	ASTAT0				MSTAT0				Status
SR7	0	0	0	DIVDEST					Destination
SR8	0	TDEST1		MDEST1					Destination
SR9	0	0	0	ADEST1					Destination
SR10	ASTAT1				MSTAT1				Status
SR11	FPEX-Taken	DSR in progress	FPCN	Carry	DIVSTAT				Status

Status register structure

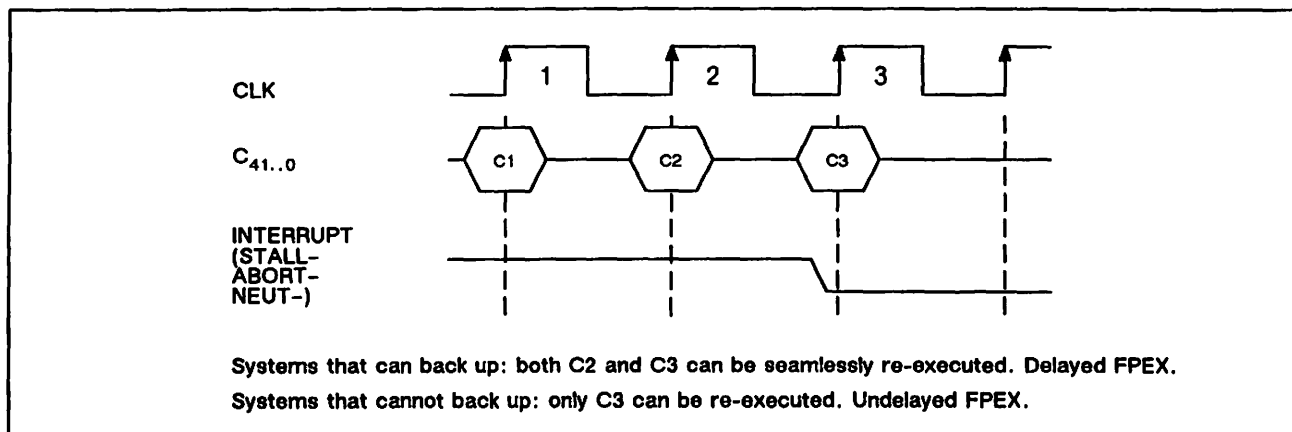
Non IEEE  
flags to  
software sum

Before you  
get underflow  
must pick  
denominator

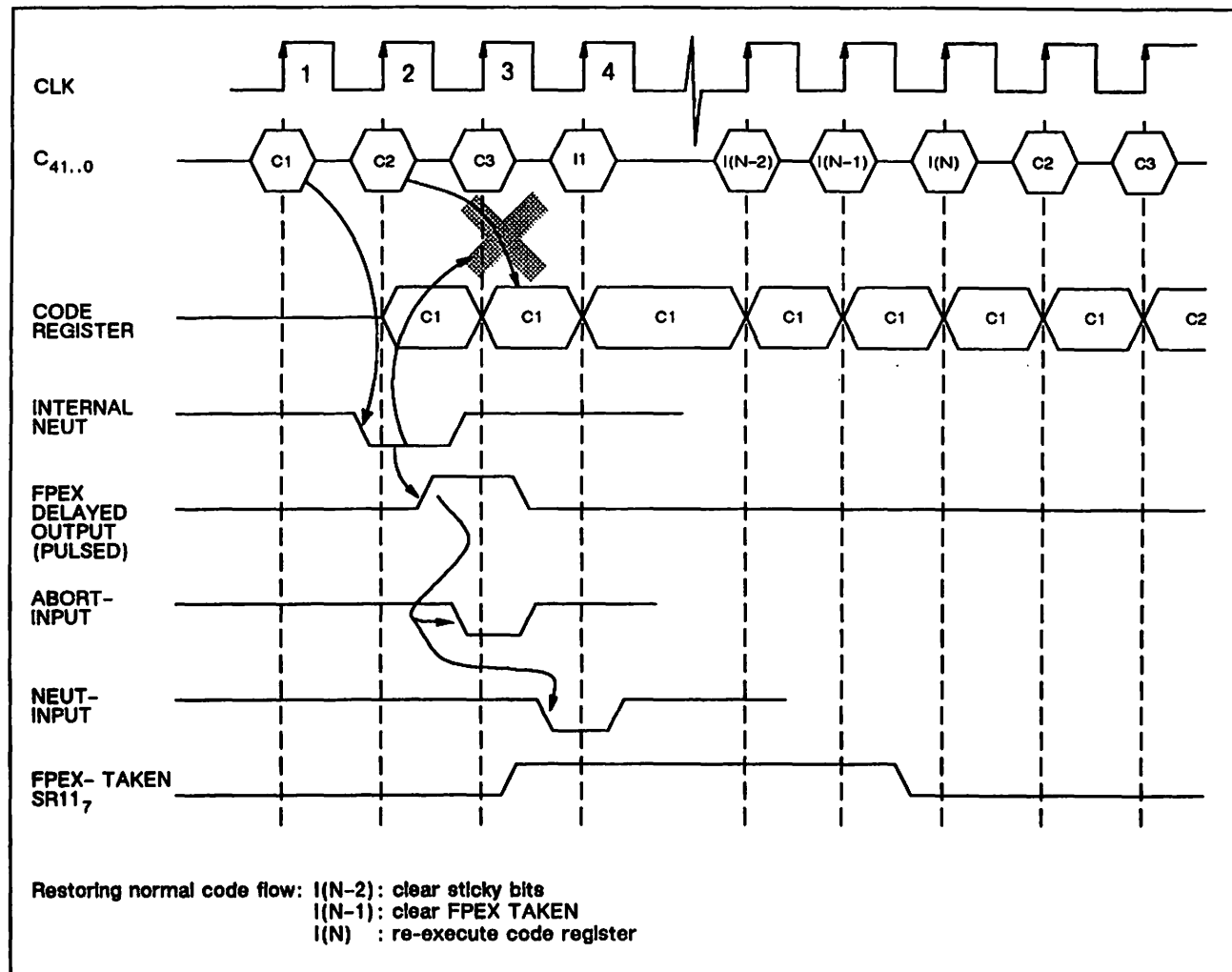
Two divided  
can't go to denum  
or other  
but denominator  
with chopping

System long handle  
always traps  
on underflow

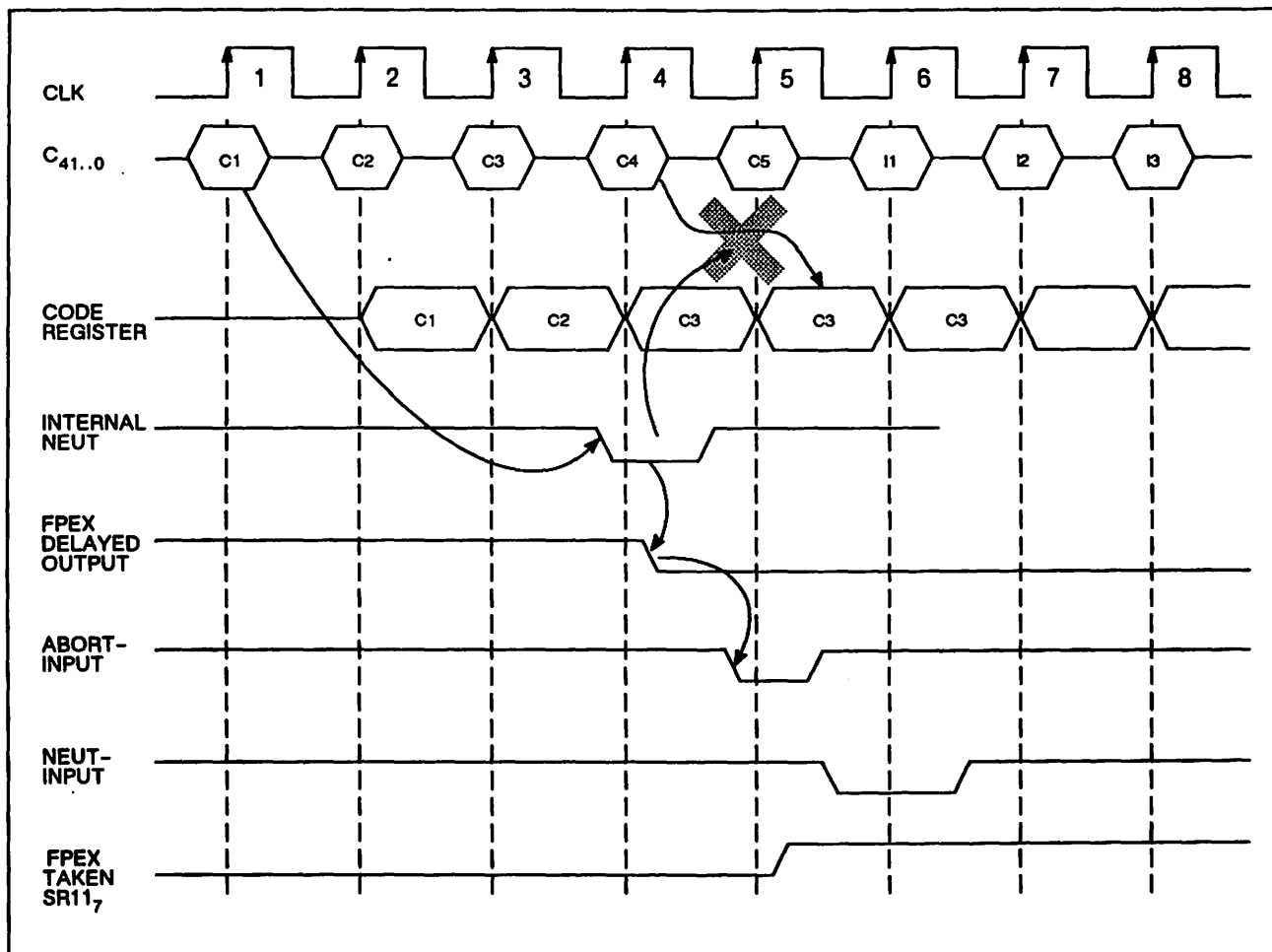




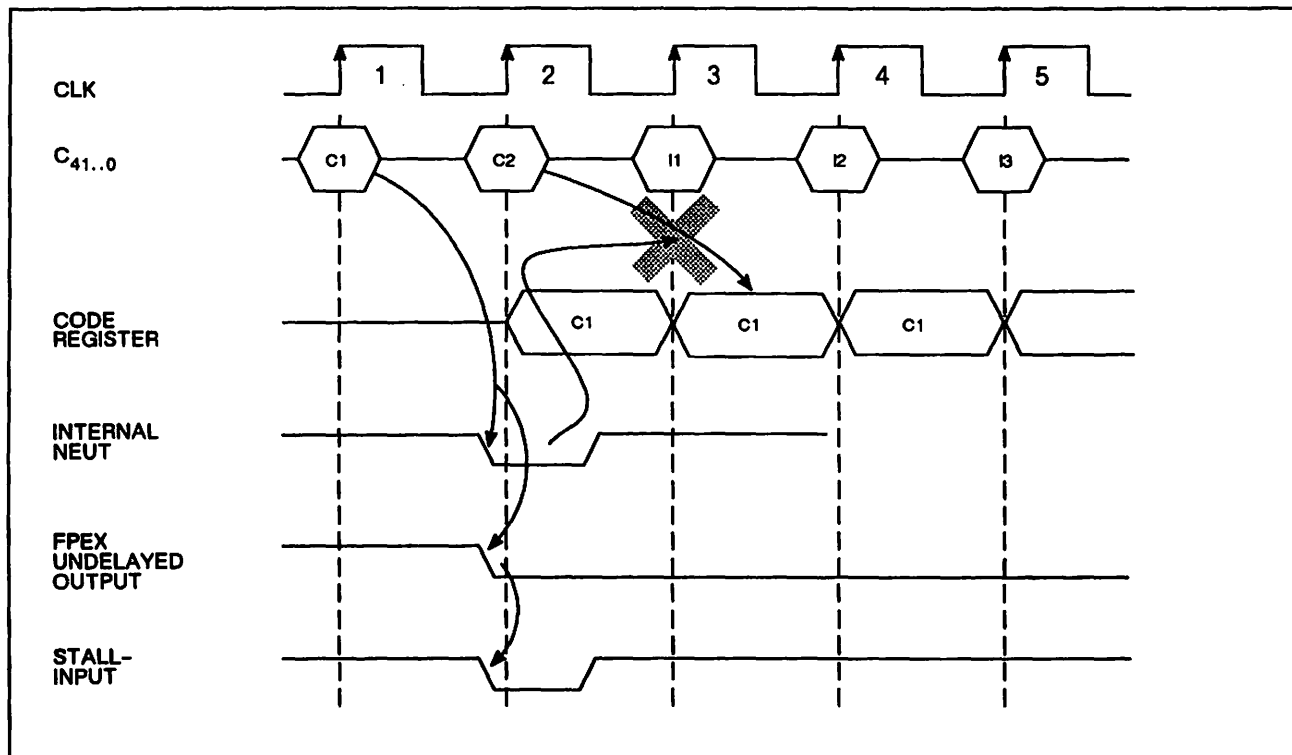
System types



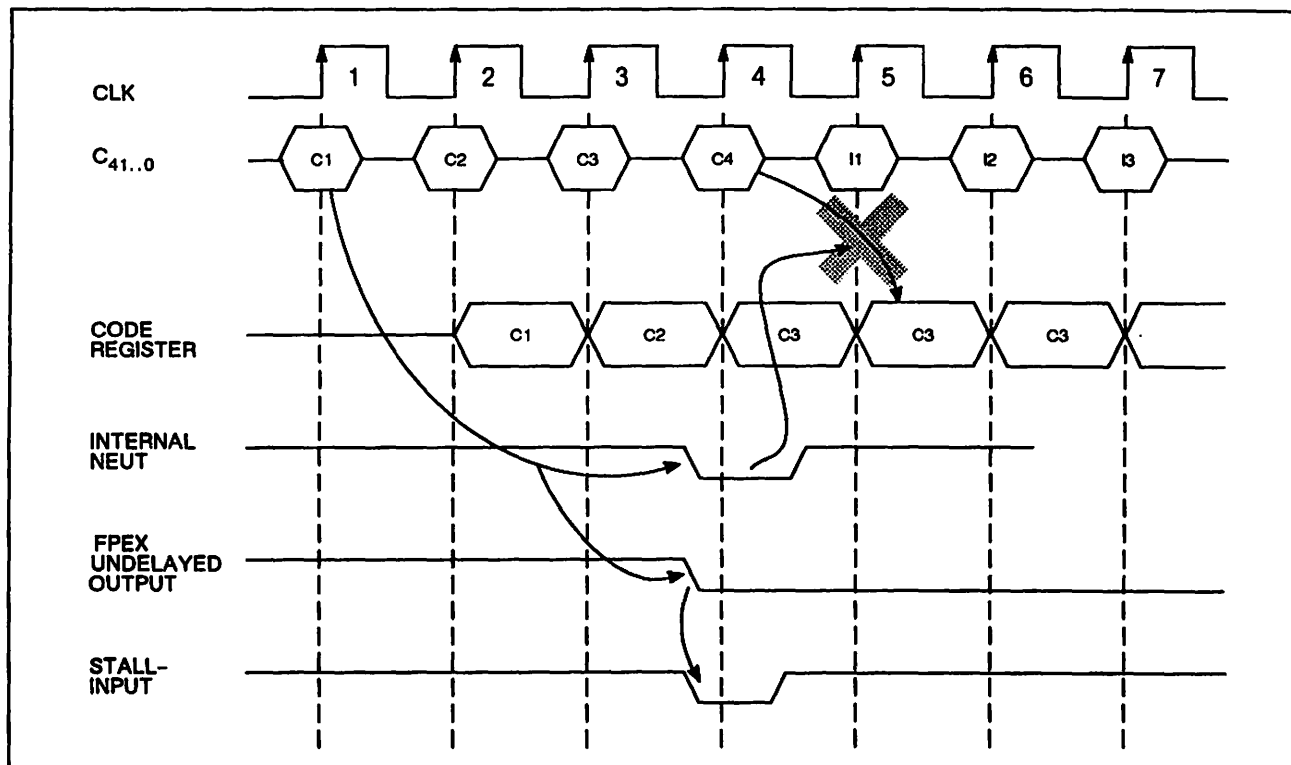
Source exception on C1



Result exception on C1



Source exception on C1, undelayed FPEX



Result exception on C1, undelayed FPEX