CPE/EE 421/521
Fall 2004
Chapter 6 – Exception Handling and the 68000

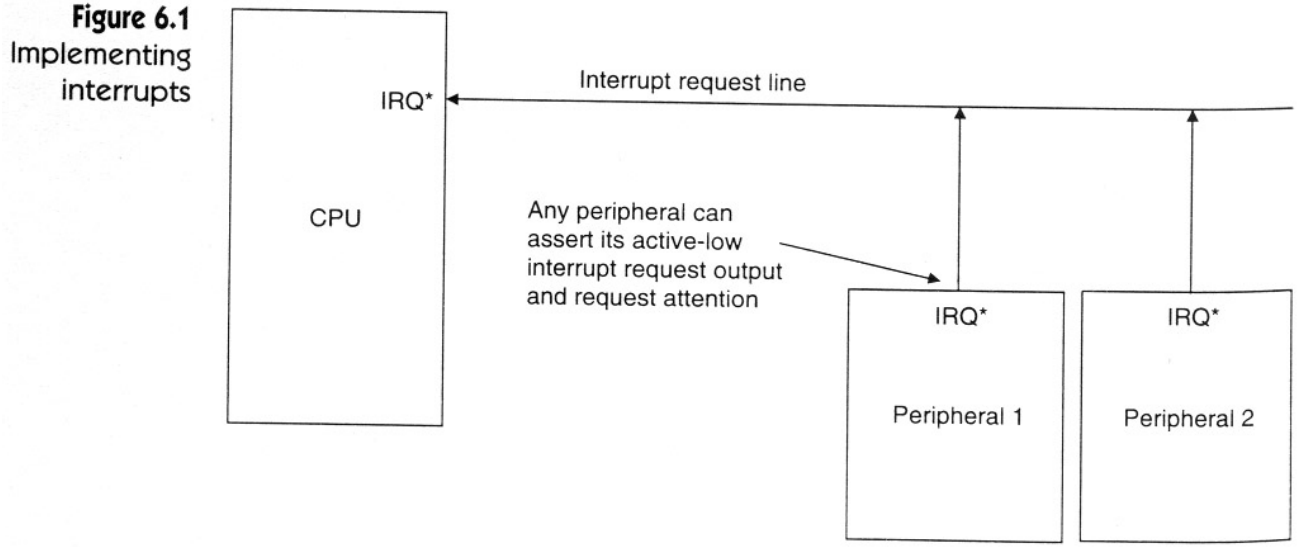
Dr. Rhonda Kay Gaede

UAH

Exception Handling and the 68000 – Overview

- Interrupts and exceptions are events that alter the normal execution of a program
  - Exception –
  - Interrupt –
- Exception examples
  - _____
  - _____
  - _____
- Interrupt examples
  - _____
  - _____
- Each type of exception has its own _____

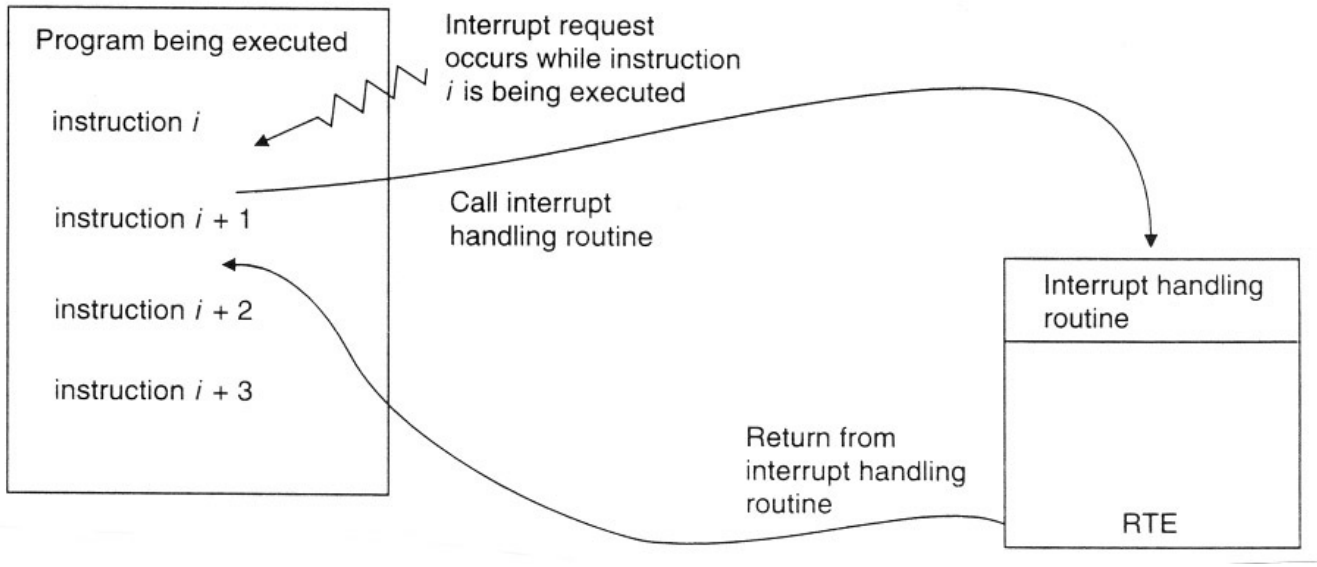
6.1 Interrupts – Interrupt Processing Mechanism



- Interrupt is an _____ event
- When an interrupt occur, the computer can:
  - _____
  - _____

6.1 Interrupts – Interrupt Service Actions

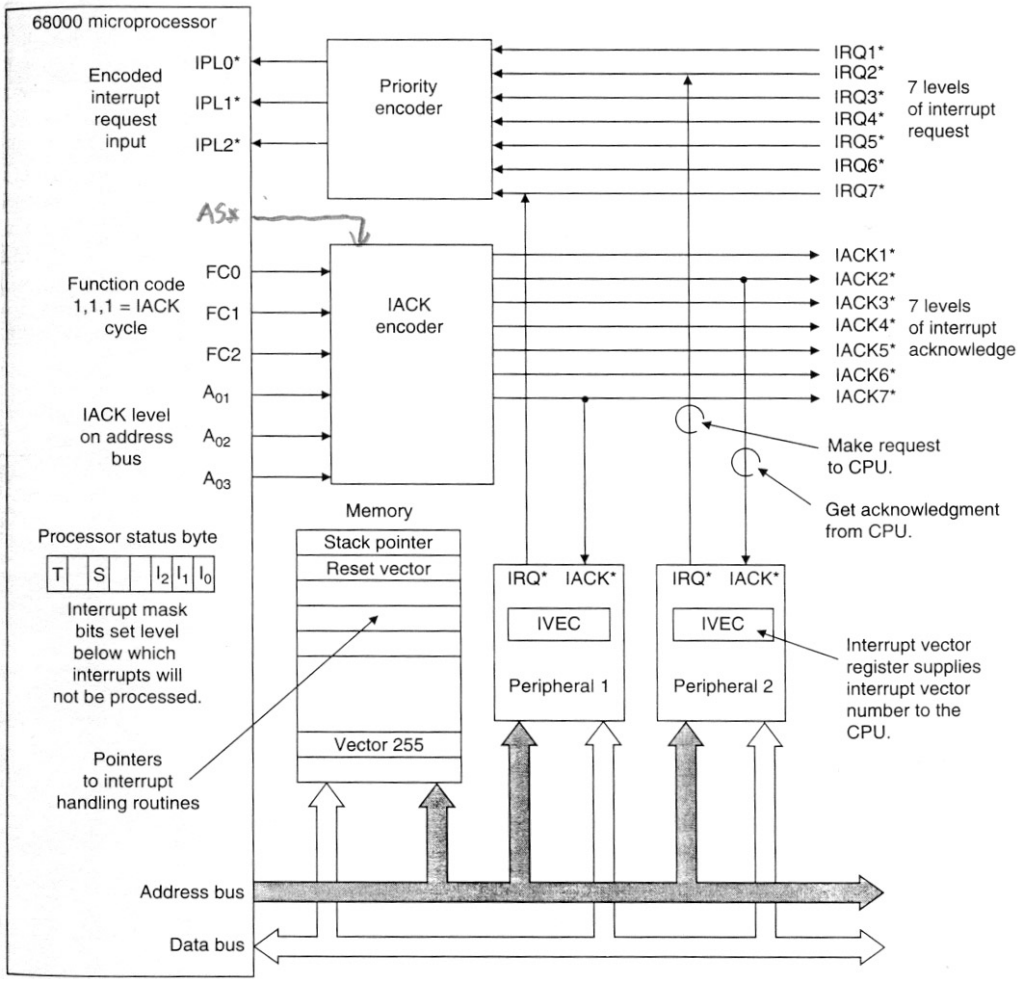
1. The computer completes its current machine-level instruction
2. The contents of PC is saved (on stack)
3. The state of the processor (status word) is saved on the stack
4. Jump to the location of the interrupt handling routine



6.1 Interrupts – Interrupt Processing Facts

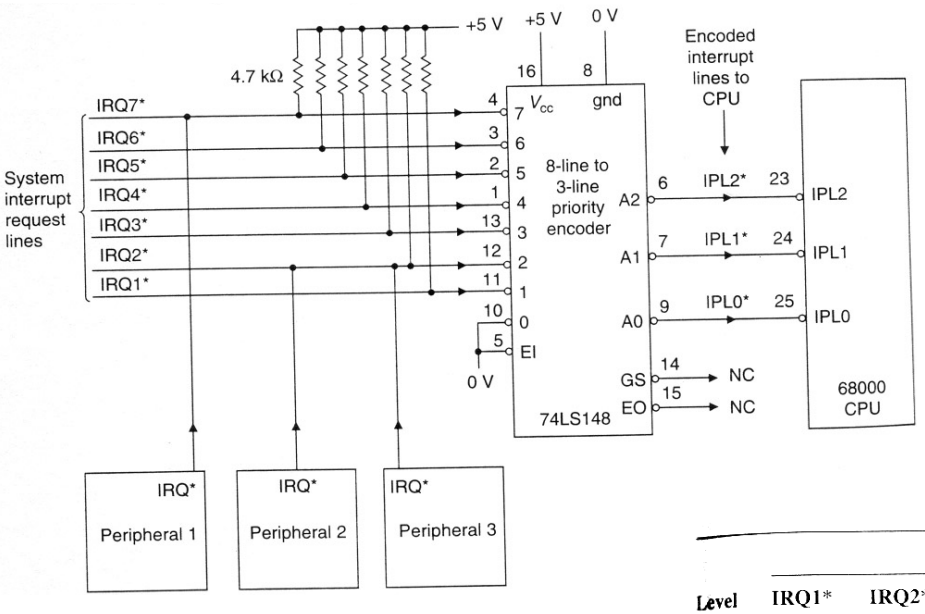
- The interrupt is _____ to the interrupted program
- Interrupt *request*:
  - Can be _____ or _____
  - When it is deferred, it is said to be masked
  - Special one: *nonmaskable interrupt request (NMI)*
  - The 68000 NMI: IRQ7 (MSP430: RST*/NMI pin)
- Prioritized interrupts
- Vectored interrupts
  - Requesting peripheral identifies itself, CPU doesn't have to poll the status of each device to discover the interrupter

6.1 Interrupts – 68K Interrupt Interface: Schematic



- A peripheral asserts output IRQ* that is connected to one of IRQ1* to IRQ7*, which are, in turn, input to a _____
- The priority level of the interrupt is compared to the level of interrupts currently being accepted _____
- When the 68000 decides to service an interrupt, ___ is output on FC2-FC0 and the _____ _____ is output on A3-A1
- FCs and As generate an _____ using a decoder
- When the peripheral receives an IACK, the device identifies itself by sending an _____ _____ to the CPU

6.1 Interrupts - 68K Interrupt Interface: Interrupt Encoding Details



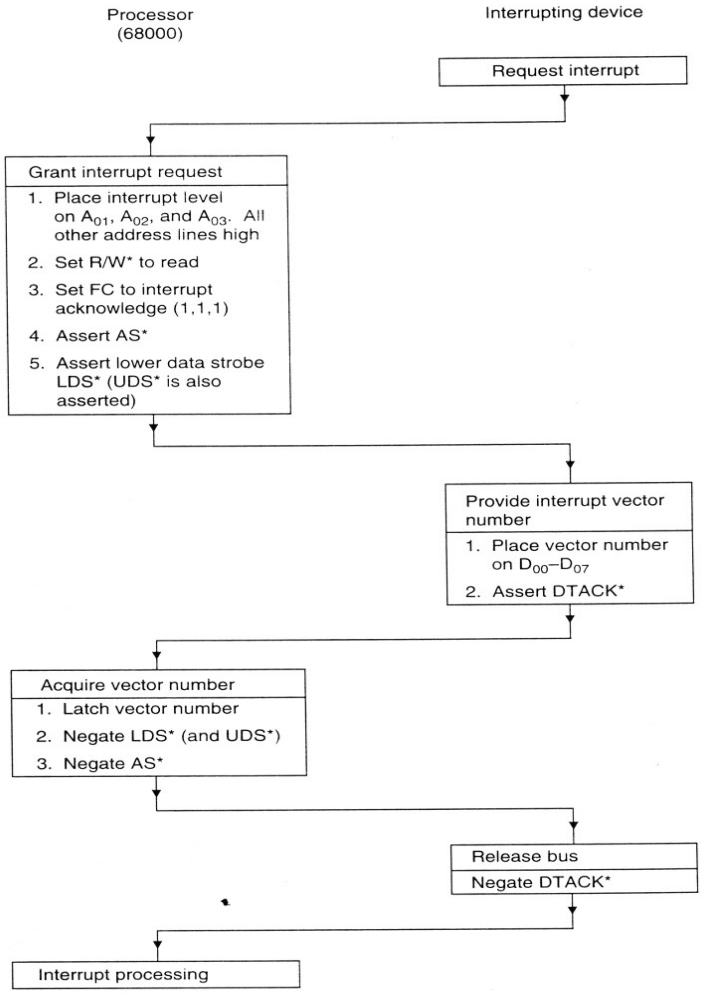
- Reset, bus error, address error, and trace exceptions take precedence over an interrupt

- A level 7 interrupt CAN interrupt level 7 interrupt

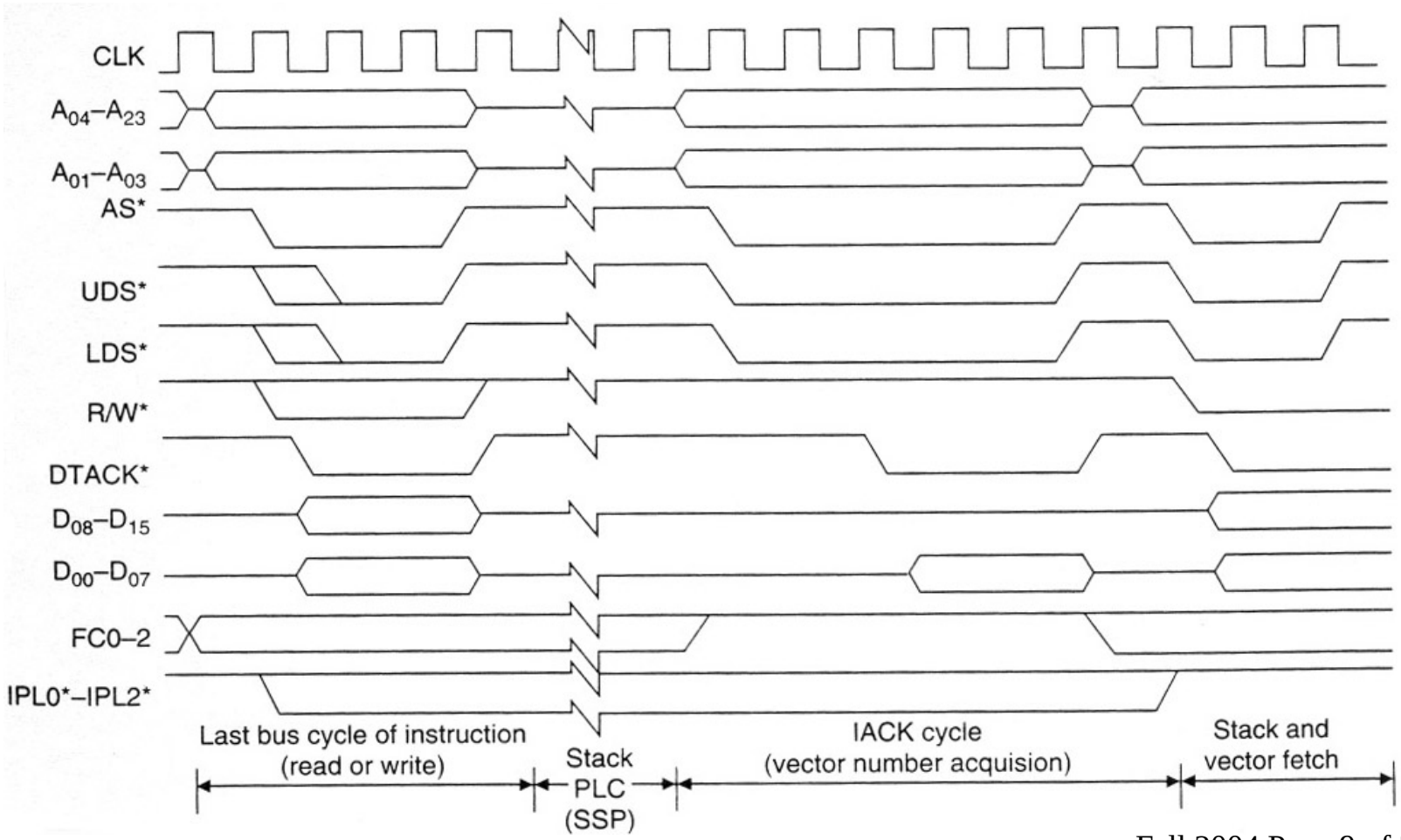
	Inputs							*Outputs*		
Level	**IRQ1***	**IRQ2***	**IRQ3***	**IRQ4***	**IRQ5***	**IRQ6***	**IRQ7***	**IPL2***	**IPL1***	**IPL0***
7	X	X	X	X	X	X	0	0	0	0
6	X	X	X	X	X	0	1	0	0	1
5	X	X	X	X	0	1	1	0	1	0
4	X	X	X	0	1	1	1	0	1	1
3	X	X	0	1	1	1	1	1	0	0
2	X	0	1	1	1	1	1	1	0	1
1	0	1	1	1	1	1	1	1	1	0
0	1	1	1	1	1	1	1	1	1	1

Note: 0 = low level
1 = high level
X = don't care

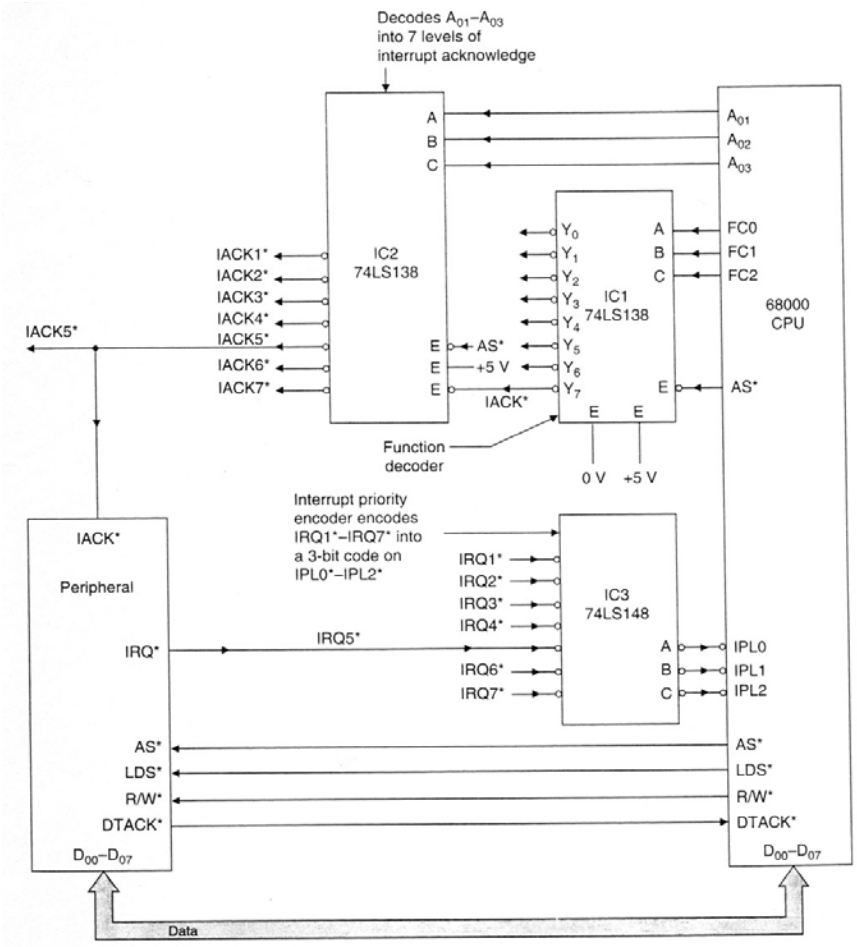
6.1 Interrupts – Processing the Interrupts: IACK* Sequence



6.1 Interrupts - Processing the Interrupts: Interrupt Timing Diagram



6.1 Interrupts – Vectored Interrupts



6.1 Interrupts – Exception Vectors

- A vector is associated with each type of exception
  - Vector is the 32-bit absolute address of the appropriate _____
- 256 exception vectors, 32 bits (4 bytes) each, extending from address $00 0000 to $__ _____
- Vectors 0-63 : EXCEPTIONS
- Vectors 64-255 : INTERRUPT HANDLING ROUTINES
- Difference between the reset vector and all other exceptions:
  - It requires 2 longwords
  - Located in SP space (FC = 110); others are in SD space (FC = 101)

6.1 Interrupts – Reset Facts

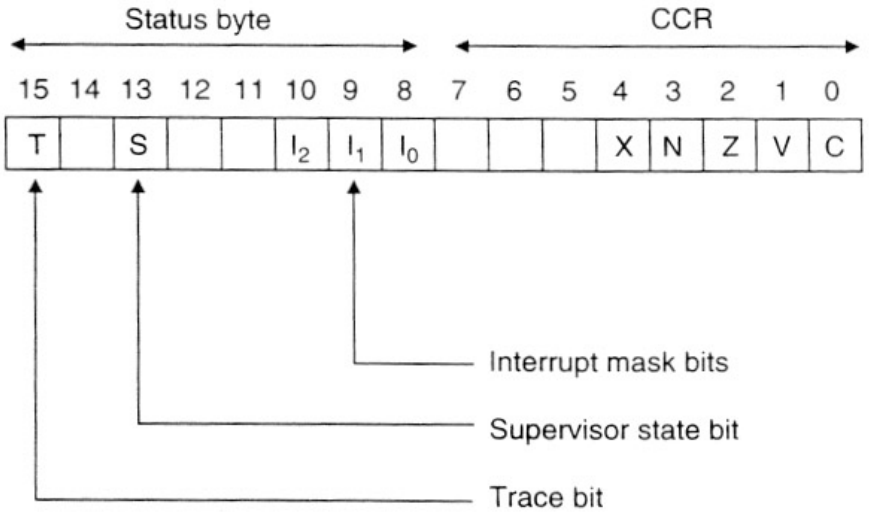
- When the RESET* pin is asserted for the appropriate duration:
  - SR = $2700
  - SSP is loaded with the longword @ $00 0000
  - PC is loaded with the longword @ $00 0004

6.2 Privileged States and the 68000 – The Status Register

S When set, the *supervisor-state bit* indicates that the 68000 is in its supervisor state. When clear, S indicates that the 68000 is in its user state.

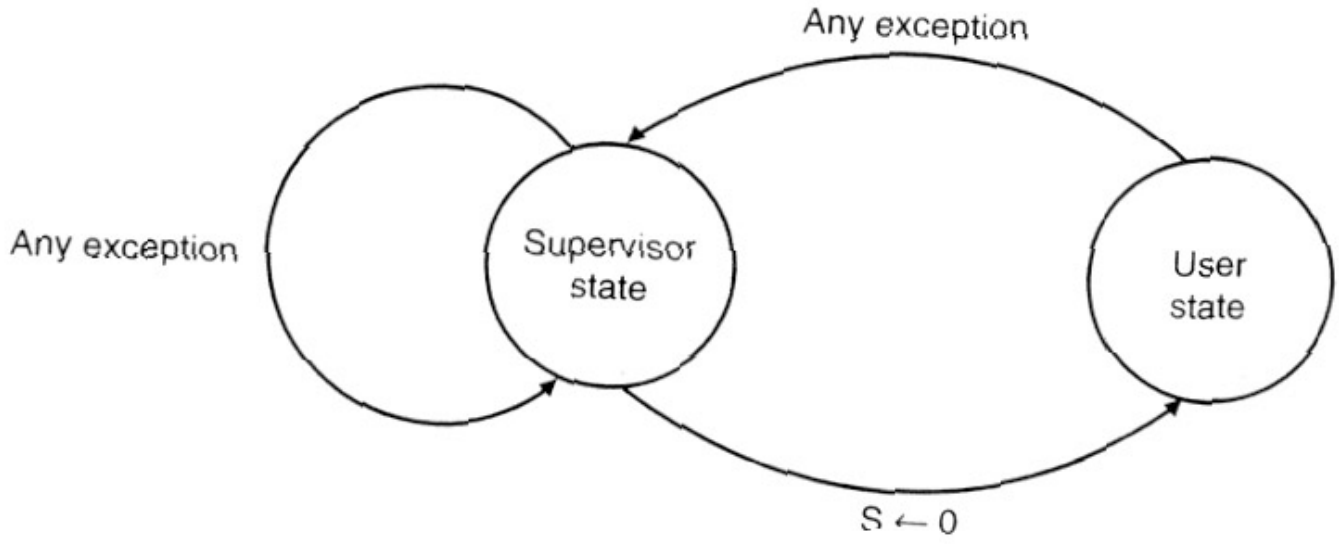
T When the *trace bit* is clear, the 68000 operates normally. When T = 1, the 68000 generates a *trace exception* after the execution of each instruction. The trace exception is used to debug programs.

I_2, I_1, I_0 The interrupt mask bits, I_2, I_1, and I_0 indicate the level of the current interrupt mask (i.e., 0 to 7).



6.2 Privileged States and the 68000 – User/Supervisor Mode Transitions

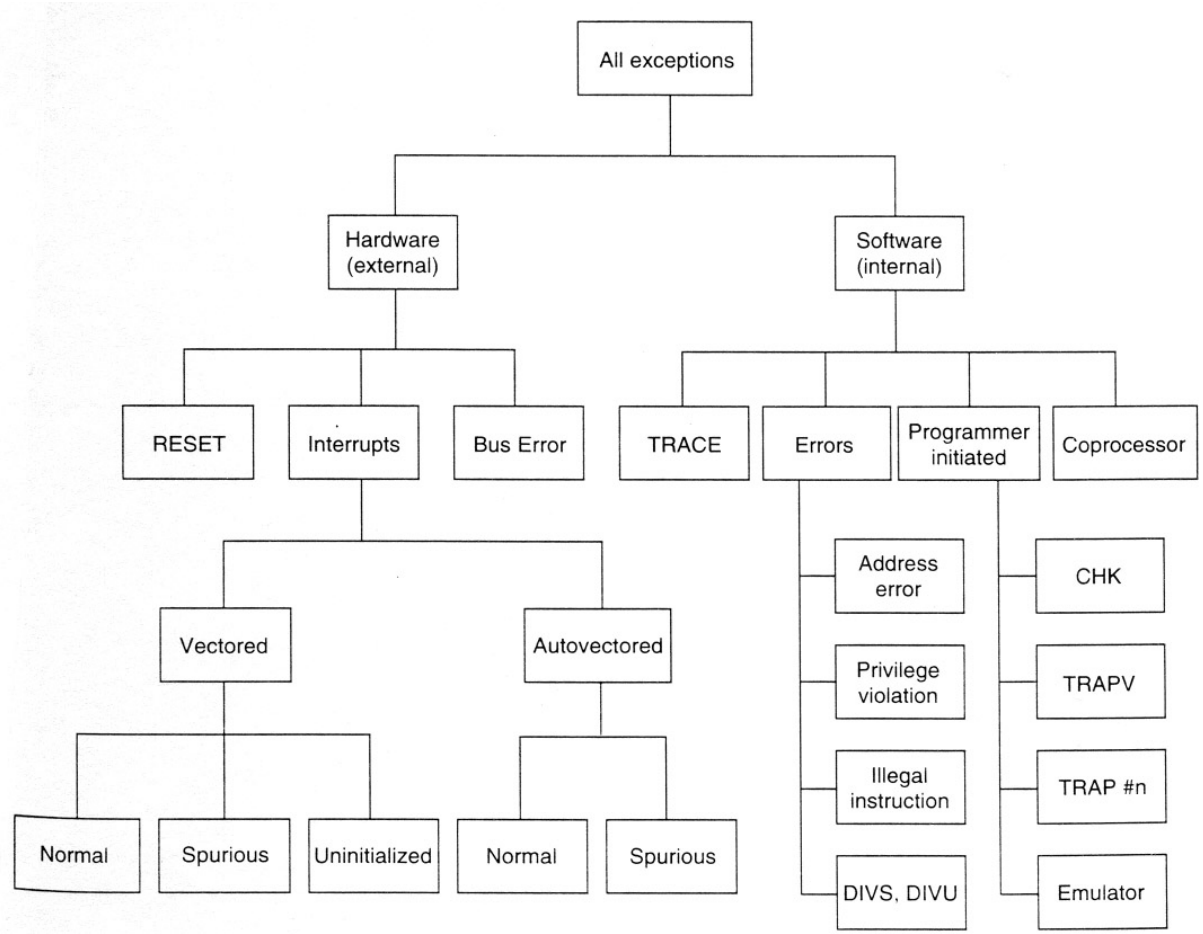
An exception always forces the 68000 into the supervisor state



6.3 Exception Processing

- Phase 1 – The 68000 makes a temporary internal copy of the pre-exception status register, turns on _____ and disables _____.
- Phase 2 – The vector number corresponding to the current exception is determined, then multiplied by 4 to calculate the value of a pointer to the _____ routine.
- Phase 3 – The current CPU _____ is saved on the stack pointed to by the Supervisor Stack Pointer in a data structure called an _____.
  - For _____, _____, and _____, this context consists of the program counter, the status register, the data access address and memory access type and function code
  - For all other exceptions, this context consists of only the _____ _____ and the _____
- Phase 4 – Load the _____ with the address of the first instruction of the exception handling routine.

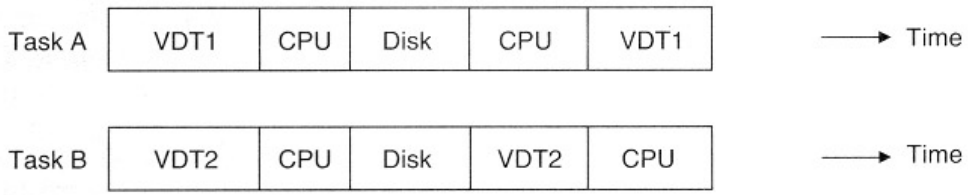
6.4 Exceptions Implemented by the 68K



6.5 Interrupts and Real-Time Processing

- Multitasking (multiprogramming)
  - concurrent execution
  - multiple tasks (processes)
  - resource sharing (multiple users using the same printer)
- Multiprocessing
  - parallel execution
  - multiple PROCESSORS!

6.5 Interrupts and Real-Time Processing - Multitasking



- _____ (to schedule activities)
- _____ (to switch between tasks)

Resource	Activity					
	Slot 1	Slot 2	Slot 3	Slot 4	Slot 5	Slot 6
VDT1	Task A				Task A	
VDT2	Task B				Task B	
Disk			Task A	Task B		
CPU		Task A	Task B	Task A		Task B

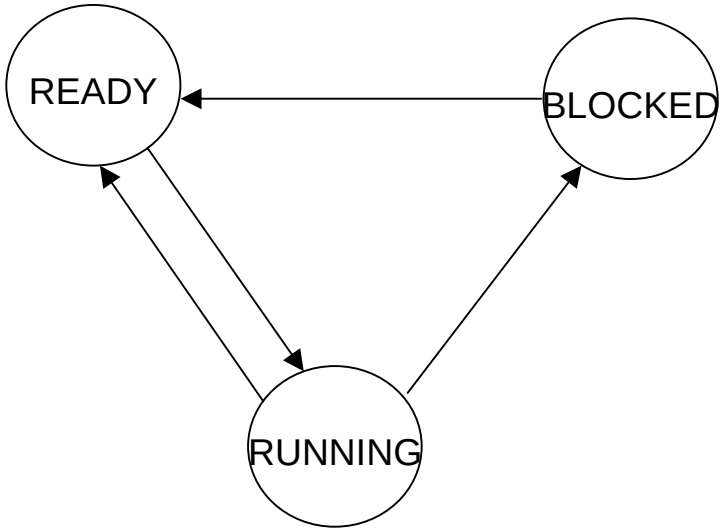
Time

6.5 Interrupts and Real-Time Processing - Real-Time Operating System

- Real time - meaningful time
  - fast enough to influence the system at that moment
    - space shuttle / chemical plant
- Real-time system
  - Optimizes the response time to events
  - Tries to use resources efficiently
- Multitasking system
  - Optimizes resource utilization
  - Tries to provide a reasonable response time

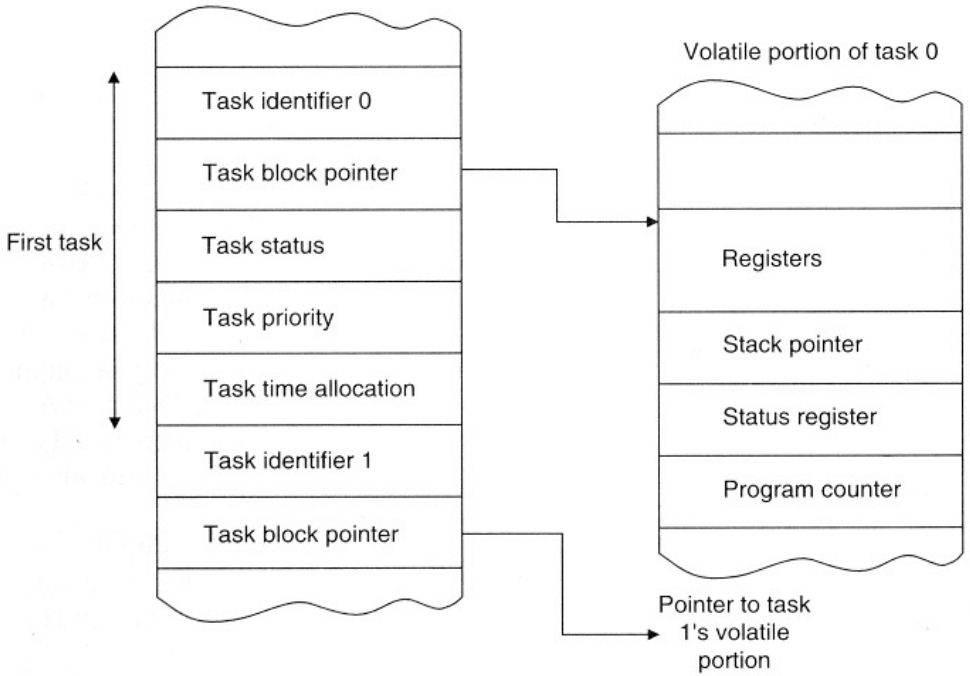
6.5 Interrupts and Real-Time Processing - Real-Time Kernel

- Scheduler is the *kernel*, *nucleus*, of a real-time OS
- Functions
  - a first-level interrupt handler
  - scheduler - the sequence in which tasks are executed
  - interprocess communication
- Task States
  - Ready
  - Running
  - Blocked (dormant)



6.5 Interrupts and Real-Time Processing - Tasks

- Volatile portion (PC, status, registers)
- Task control block (TCB
  - Task ID
  - Task block pointer
    - PC
    - SP
    - status register
    - other registers
  - Task status
    - run / ready / blckd
  - Task priority
  - Task time allocation
    - how many slots



6.5 Interrupts and Real-Time Processing - Exception Handling and Tasks

- Preemptive real-time OS:
  - RTC generates periodic interrupts
  - used by the kernel to locate and run the next task
- How to deal with other interrupts?
  - Service them independently, subject to priority
  - Integrate them into the real-time task structure