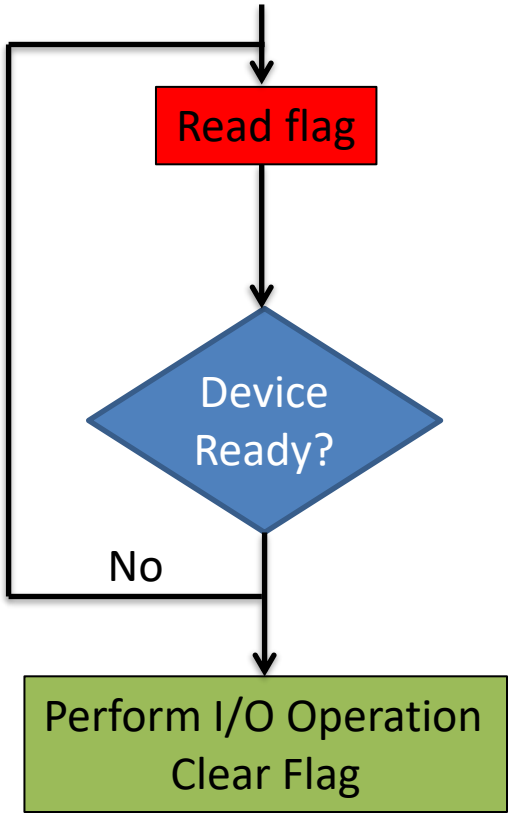
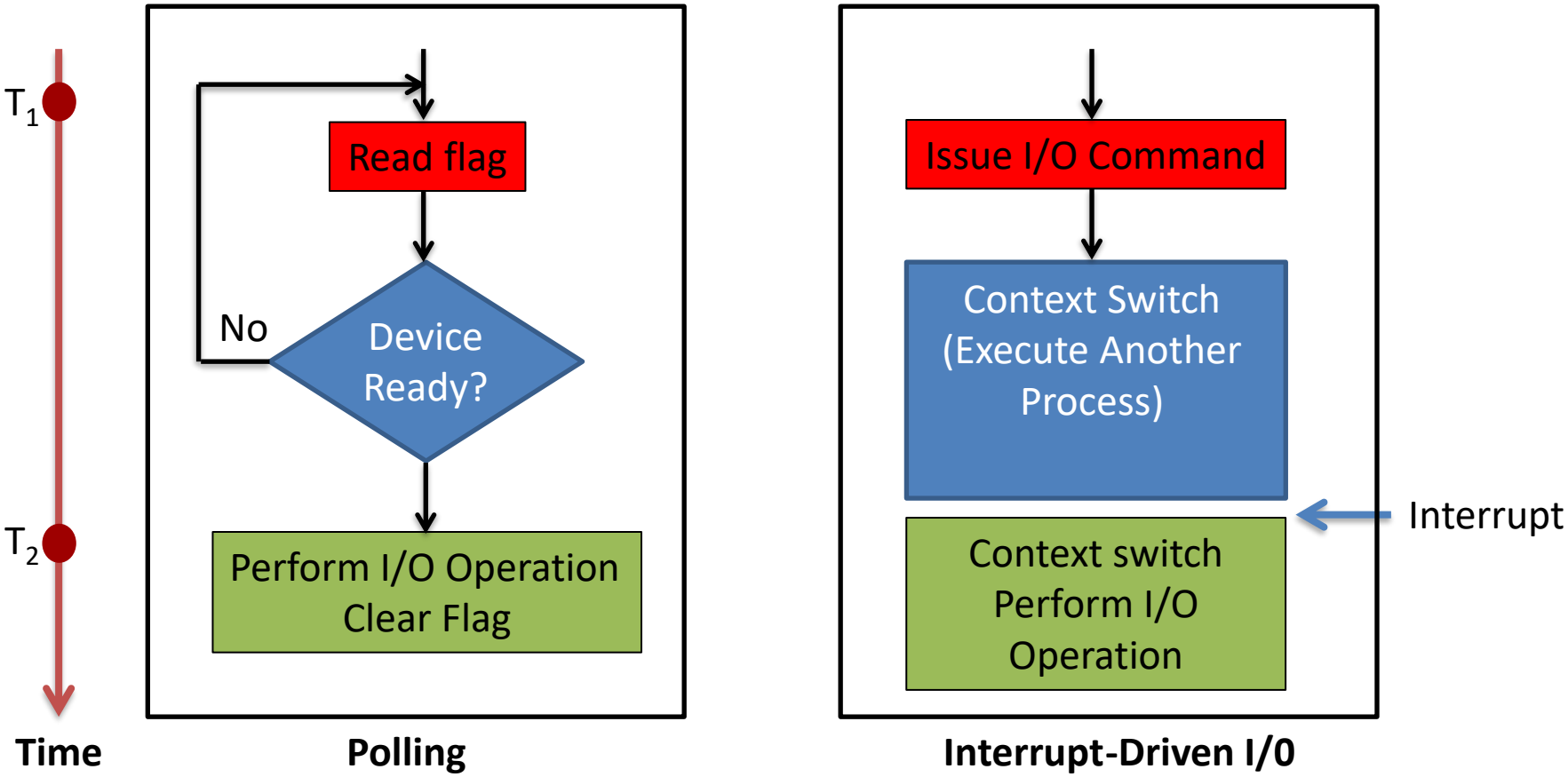
Pros and Cons of Polling



- Pros
  - Simple to implement in both software and hardware
- Cons
  - Occupies processor, preventing “useful” work
  - Very inefficient for data transfer (sending data to memory or sending data from memory via processor)
  - May miss important events when busy polling inactive devices or when doing something useful (and not polling)
- Summary
  - Micromanagement is too inefficient if the system is busy or has multiple devices

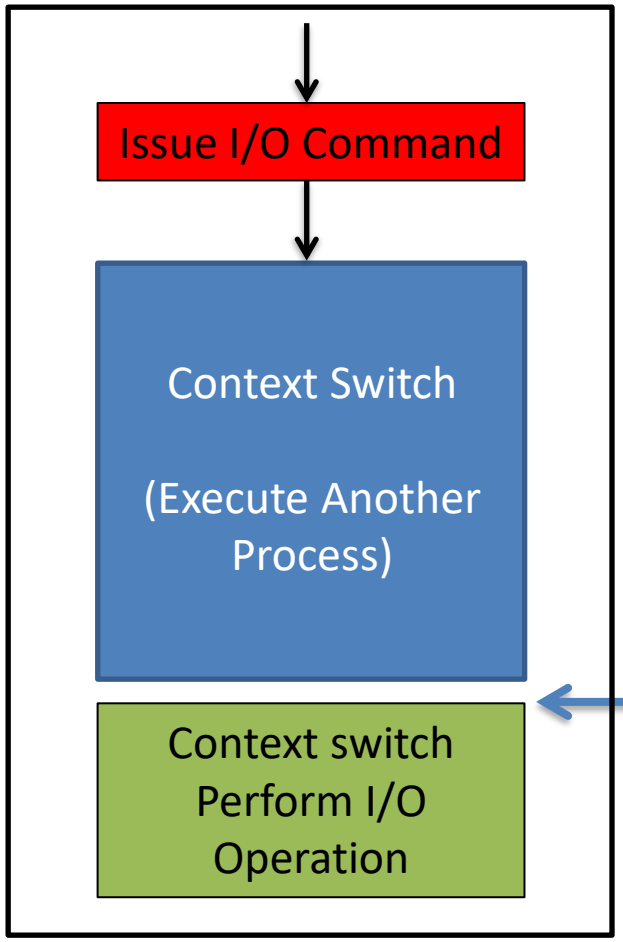
Slides are IP. Please do not copy, share, or post to website(s).

Interrupt-Driven I/O



Slides are IP. Please do not copy, share, or post to website(s).

Pros and Cons of Interrupt-Driven I/O



- Pros
  - CPU can perform useful work (e.g., think of telephone)
  - CPU can't miss important events
- Cons
  - Very inefficient for data transfer (time to process exception and perform I/O)

Slides are IP. Please do not copy, share, or post to website(s).

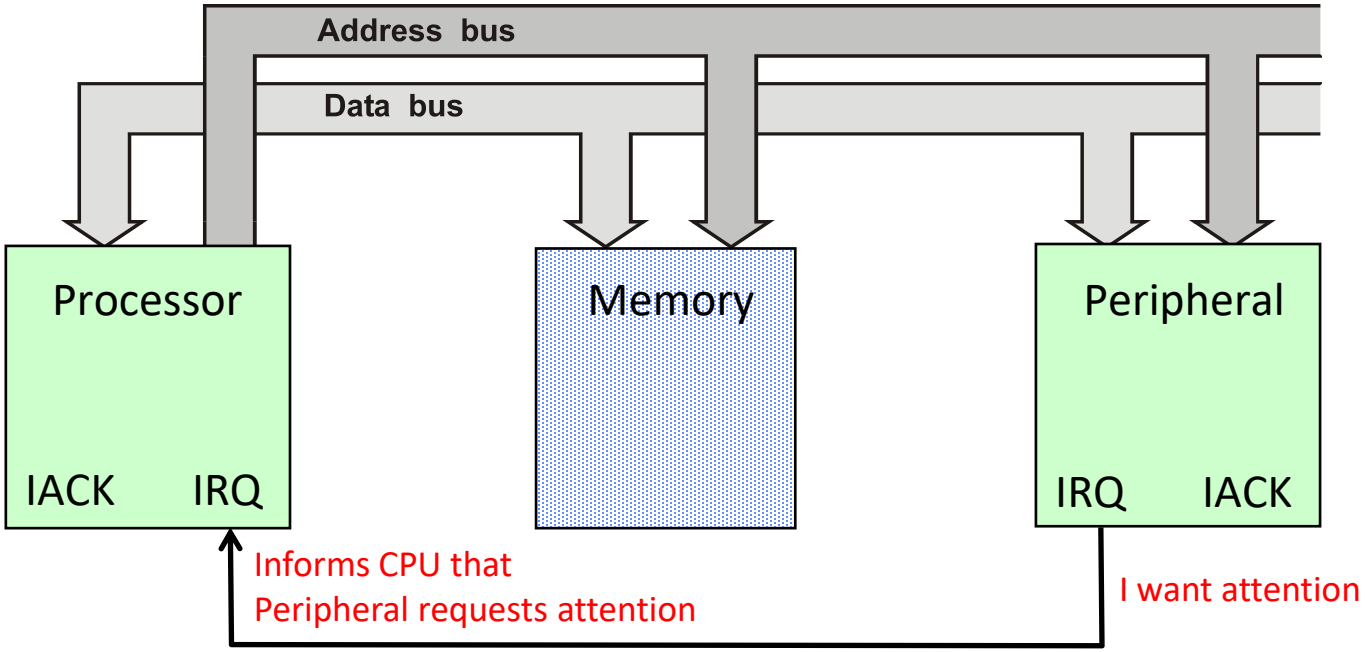
Interrupts in General

- An interrupt is an external, asynchronous signal to alert the operating system that some event or device needs attention
  - Causes may include
    - I/O Completions
    - External events
      - like mouse movement,
      - network interface card (NIC),
      - keyboard,
      - sensors, switches,
      - etc.
    - Timer expires on motherboard
      - Multi-tasking (OS course in W21)
    - Page fault
      - Virtual memory (OS course in W21)
    - Power Failure
      - Shutdown system in orderly fashion
    - Many more…

Slides are IP. Please do not copy, share, or post to website(s).

Interrupt Request Line

- Even the simplest processors have an Interrupt-Request (IRQ) line (wire) that is used by the peripherals to request attention

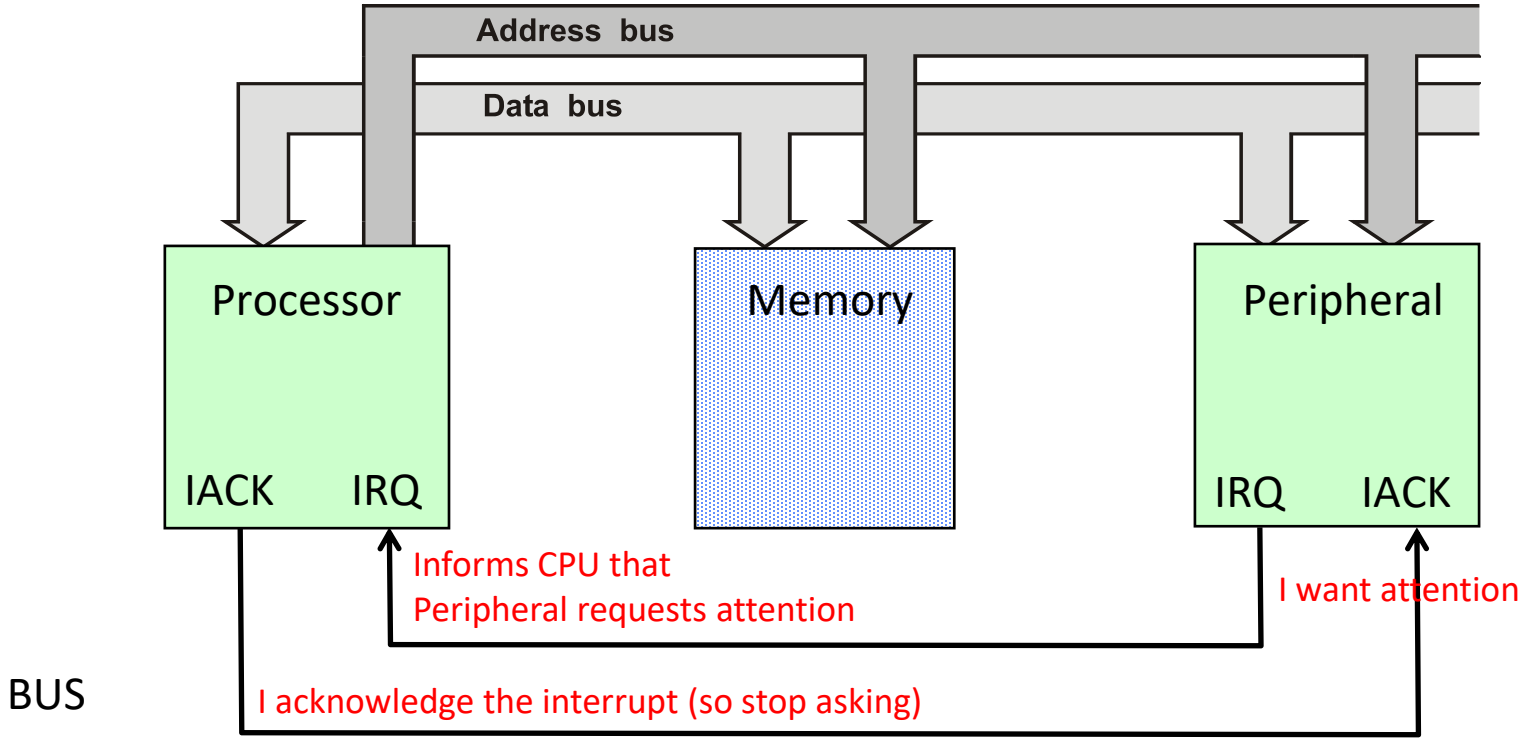


BUS

Slides are IP. Please do not copy, share, or post to website(s).

Interrupt Request Line

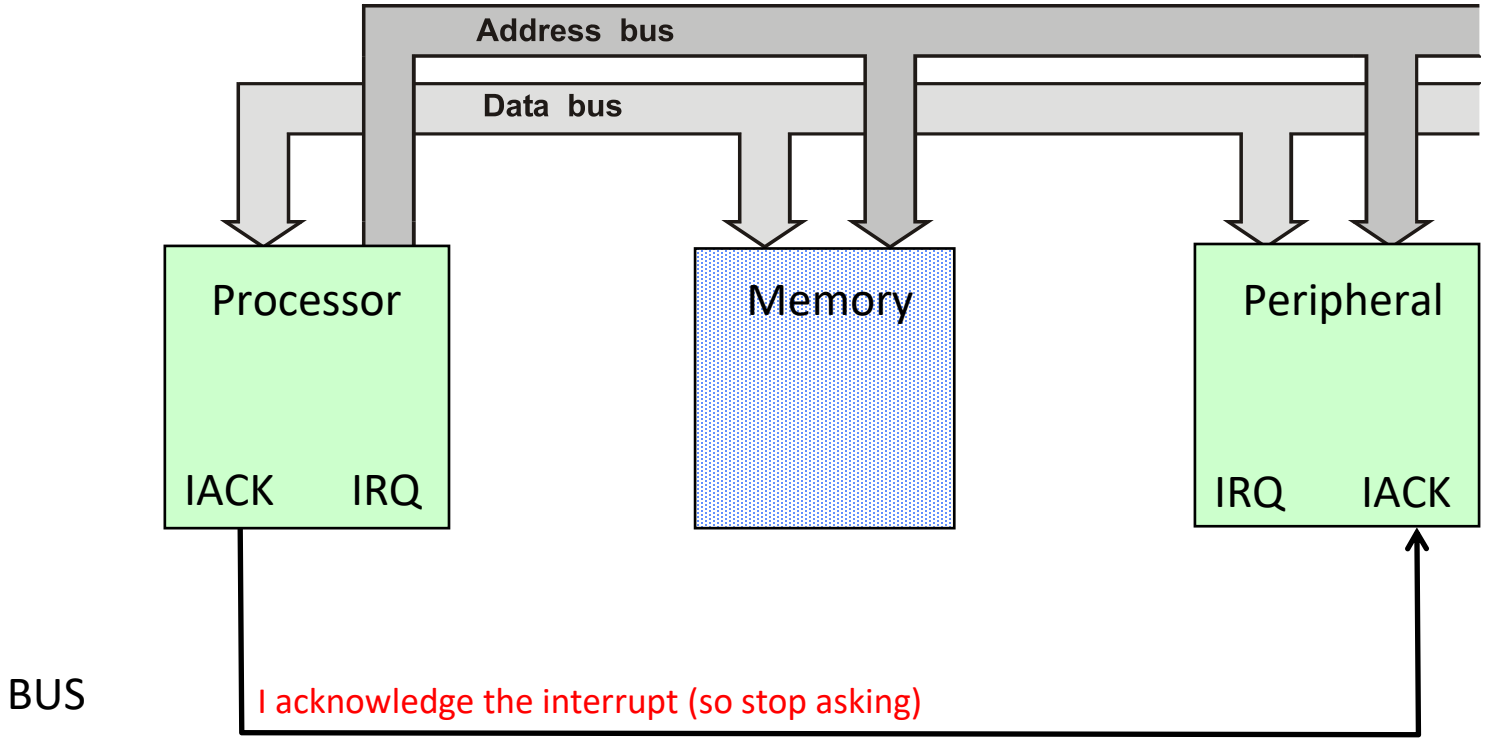
- Even the simplest processors have an Interrupt-Request (IRQ) line (wire) that is used by the peripherals to request attention



Slides are IP. Please do not copy, share, or post to website(s).

Interrupt Request Line

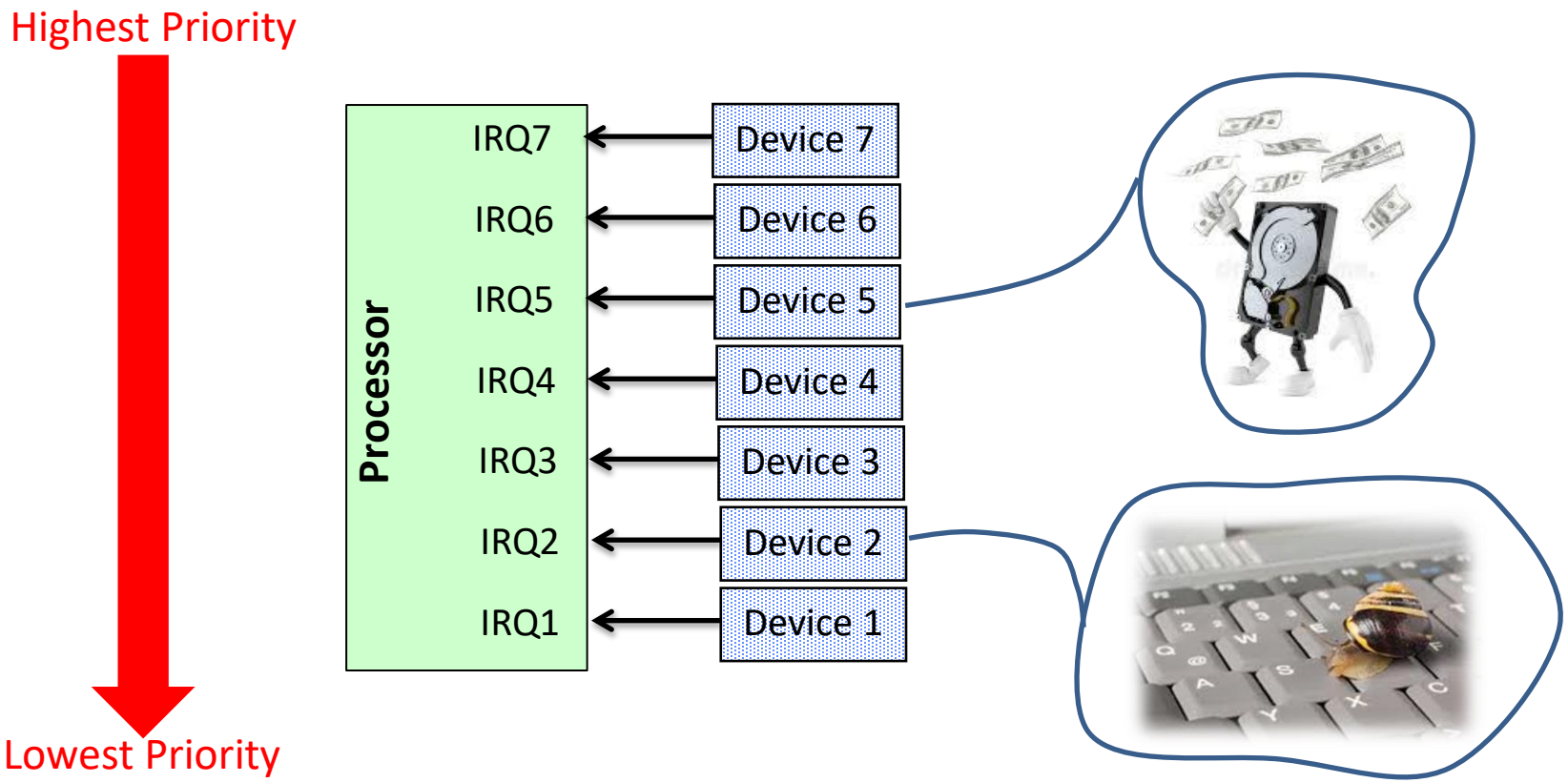
- Even the simplest processors have an Interrupt-Request (IRQ) line (wire) that is used by the peripherals to request attention



Slides are IP. Please do not copy, share, or post to website(s).

Multiple (Prioritized) Interrupts

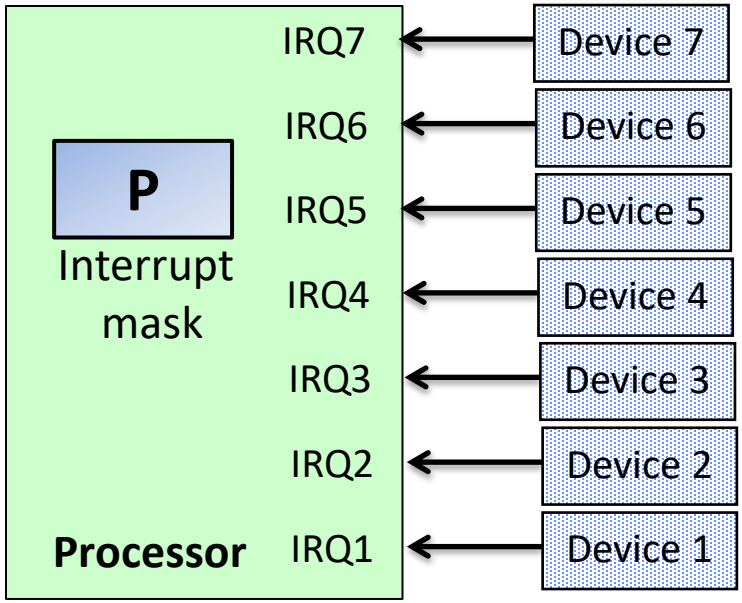
- Most computers have more than one interrupt request input with different priorities; the processor responds to the highest priority



Slides are IP. Please do not copy, share, or post to website(s).

When are Interrupts Acknowledged?

- The interrupt mask sets the level at or below which the interrupt will not be acknowledged
  - If a device interrupts at level *p*, and *p* is greater than the current level in the mask, the mask gets set to level *p*

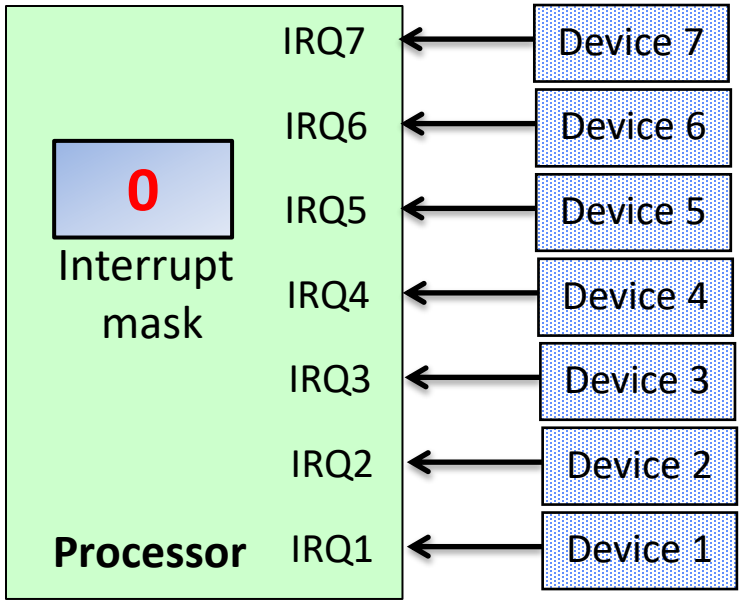


Slides are IP. Please do not copy, share, or post to website(s).

When are Interrupts Acknowledged?

- The interrupt mask sets the level at or below which the interrupt will not be acknowledged
  - If a device interrupts at level *p*, and *p* is greater than the current level in the mask, the mask gets set to level *p*

User Program Running (can always be interrupted)

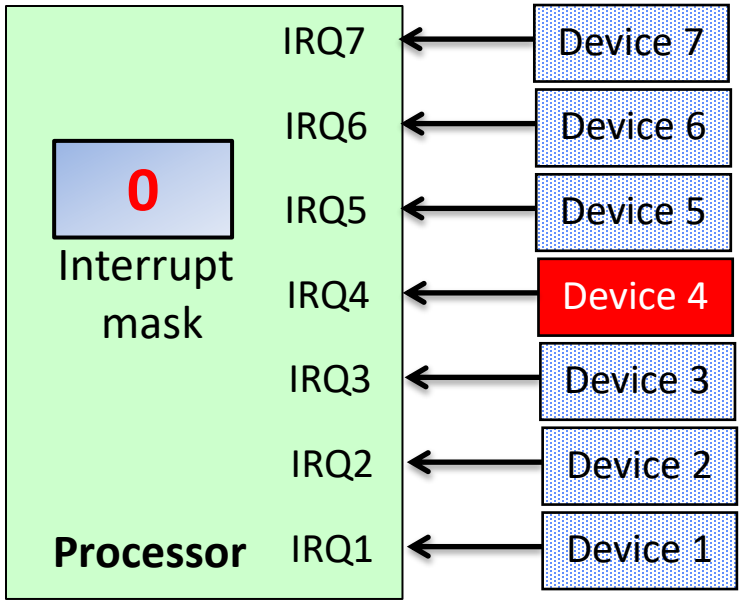


Slides are IP. Please do not copy, share, or post to website(s).

When are Interrupts Acknowledged?

- The interrupt mask sets the level at or below which the interrupt will not be acknowledged
  - If a device interrupts at level *p*, and *p* is greater than the current level in the mask, the mask gets set to level *p*

User Program Running (can always be interrupted)



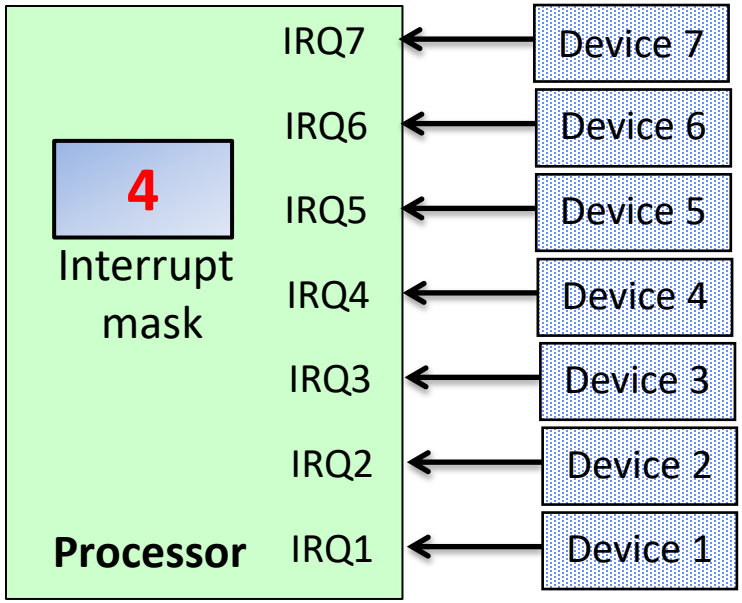
Suppose an interrupt is caused by the assertion of IRQ4 and no other interrupts are pending. The interrupt on IRQ4 will be serviced.

Slides are IP. Please do not copy, share, or post to website(s).

When are Interrupts Acknowledged?

- The interrupt mask sets the level at or below which the interrupt will not be acknowledged
  - If a device interrupts at level *p*, and *p* is greater than the current level in the mask, the mask gets set to level *p*

ISR 4 running



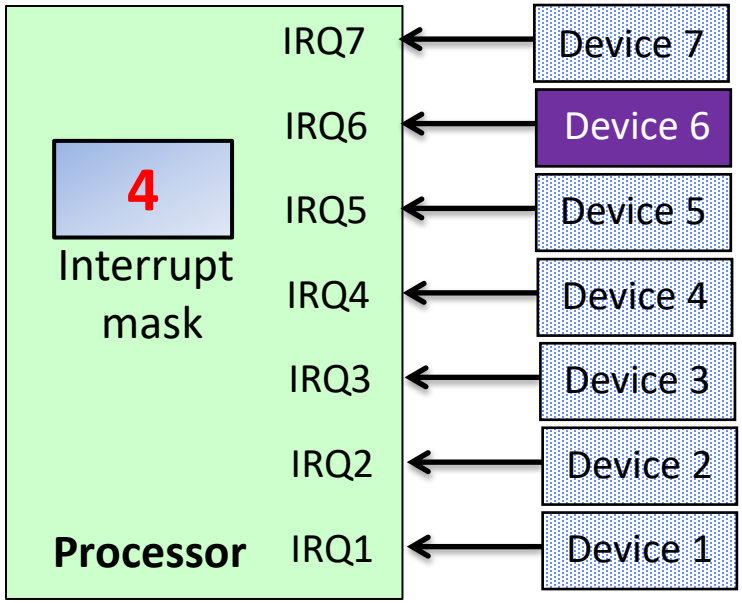
Suppose an interrupt is caused by the assertion of IRQ4 and no other interrupts are pending. The interrupt on IRQ4 will be serviced.

Slides are IP. Please do not copy, share, or post to website(s).

When are Interrupts Acknowledged?

- The interrupt mask sets the level at or below which the interrupt will not be acknowledged
  - If a device interrupts at level *p*, and *p* is greater than the current level in the mask, the mask gets set to level *p*

ISR 4 running



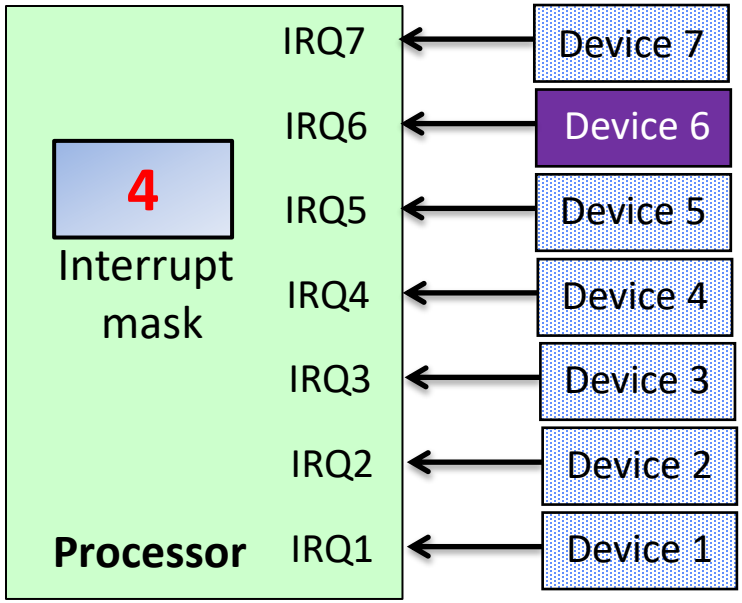
If an interrupt at a level higher than IRQ4 occurs, it will be serviced before the level 4 interrupt service routine is completed.

Slides are IP. Please do not copy, share, or post to website(s).

When are Interrupts Acknowledged?

- The interrupt mask sets the level at or below which the interrupt will not be acknowledged
  - If a device interrupts at level *p*, and *p* is greater than the current level in the mask, the mask gets set to level *p*

ISR 4 suspended



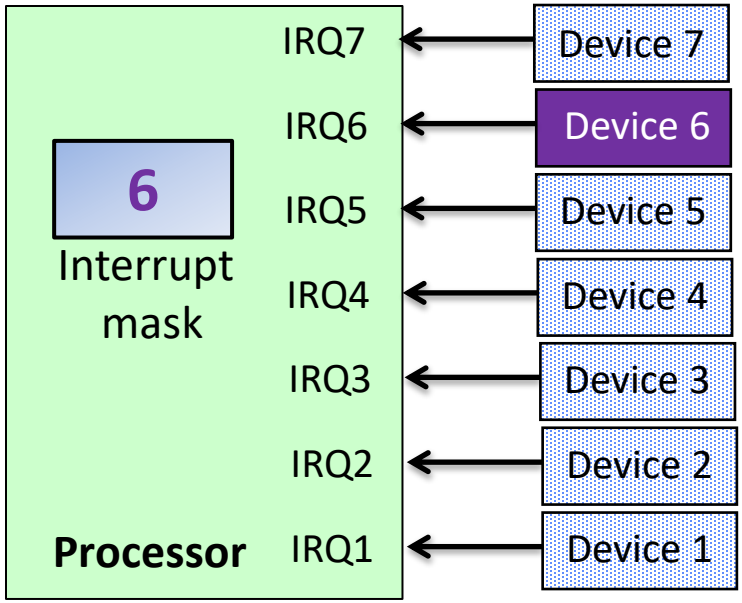
If an interrupt at a level higher than IRQ4 occurs, it will be serviced before the level 4 interrupt service routine is completed.

Slides are IP. Please do not copy, share, or post to website(s).

When are Interrupts Acknowledged?

- The interrupt mask sets the level at or below which the interrupt will not be acknowledged
  - If a device interrupts at level *p*, and *p* is greater than the current level in the mask, the mask gets set to level *p*

ISR 4 suspended



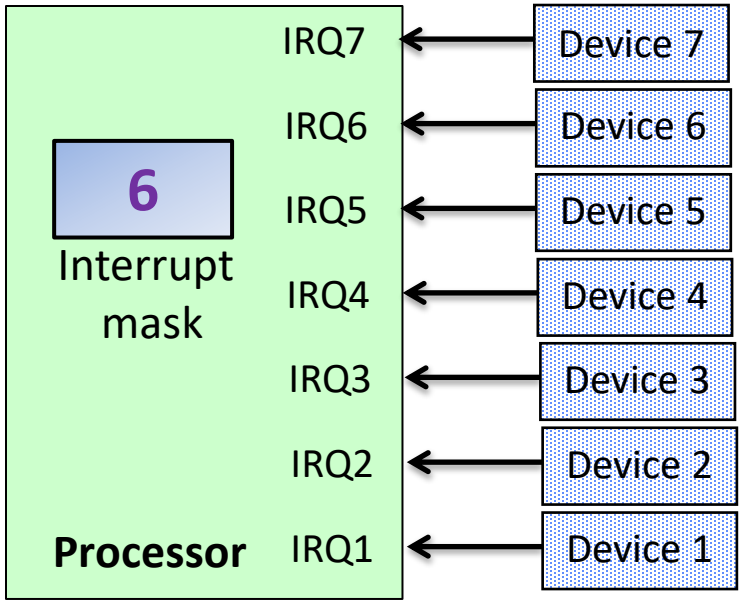
If an interrupt at a level higher than IRQ4 occurs, it will be serviced before the level 4 interrupt service routine is completed.

Slides are IP. Please do not copy, share, or post to website(s).

When are Interrupts Acknowledged?

- The interrupt mask sets the level at or below which the interrupt will not be acknowledged
  - If a device interrupts at level *p*, and *p* is greater than the current level in the mask, the mask gets set to level *p*

ISR6 running



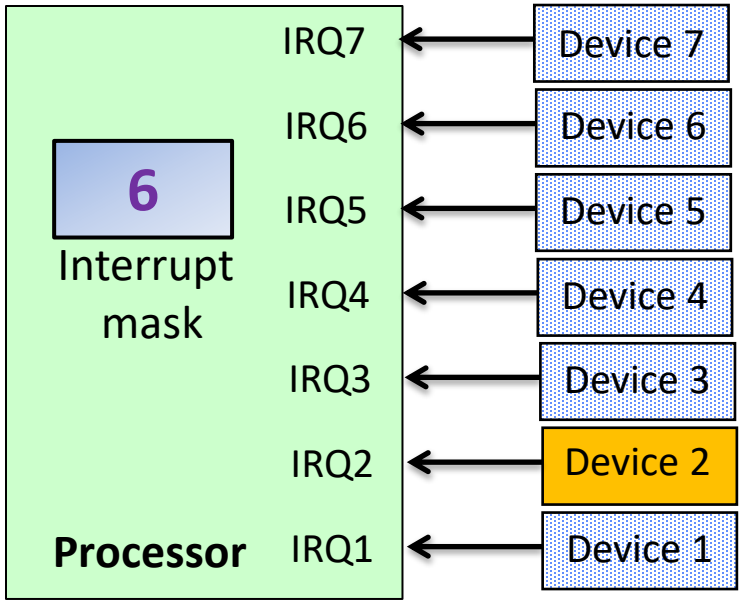
If an interrupt at a level higher than IRQ4 occurs, it will be serviced before the level 4 interrupt service routine is completed.

Slides are IP. Please do not copy, share, or post to website(s).

When are Interrupts Acknowledged?

- The interrupt mask sets the level at or below which the interrupt will not be acknowledged
  - If a device interrupts at level *p*, and *p* is greater than the current level in the mask, the mask gets set to level *p*

ISR6 running



However, an interrupt generated by IRQ2 will be stored pending the completion of ISR6 and the resumption and completion of ISR4.

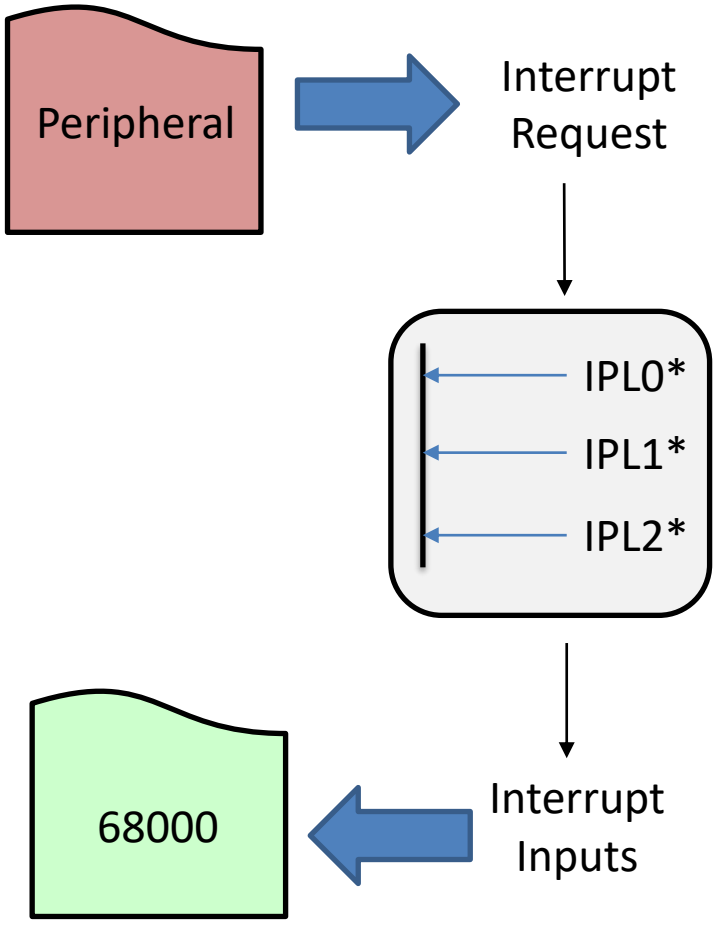
Slides are IP. Please do not copy, share, or post to website(s).

Summary

- Interrupts are prioritized
  - User programs can always be interrupted since they run with the interrupt mask set to zero
  - A lower priority interrupt-service routine can always be interrupted by a higher-priority interrupt request
  - A higher priority interrupt-service routine cannot be interrupted by a lower-priority interrupt request
- The interrupt mask sets the level at or below which the interrupt will not be acknowledged

Slides are IP. Please do not copy, share, or post to website(s).

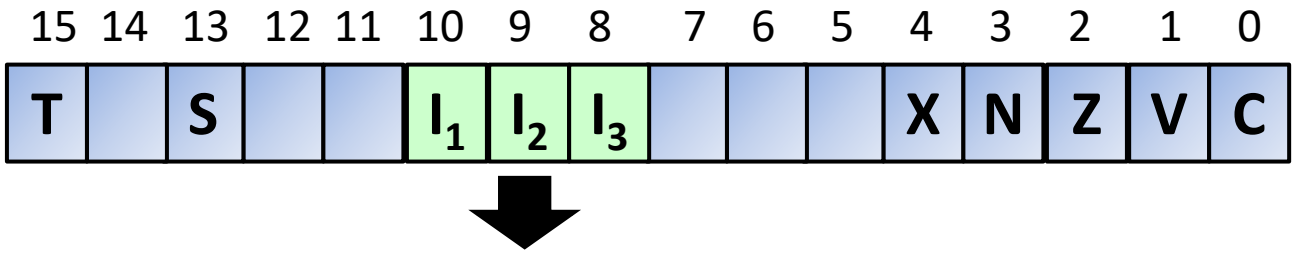
Interrupt Request Signals



Priority Level	IPL2*	IPL1*	IPL0*	Comment
7	Lo	Lo	Lo	Highest
6	Lo	Lo	Hi	
5	Lo	Hi	Lo	
4	Lo	Hi	Hi	
3	Hi	Lo	Lo	
2	Hi	Lo	Hi	
1	Hi	Hi	Lo	Lowest
0	Hi	Hi	Hi	None

Slides are IP. Please do not copy, share, or post to website(s).

Interrupt Mask



Status Register

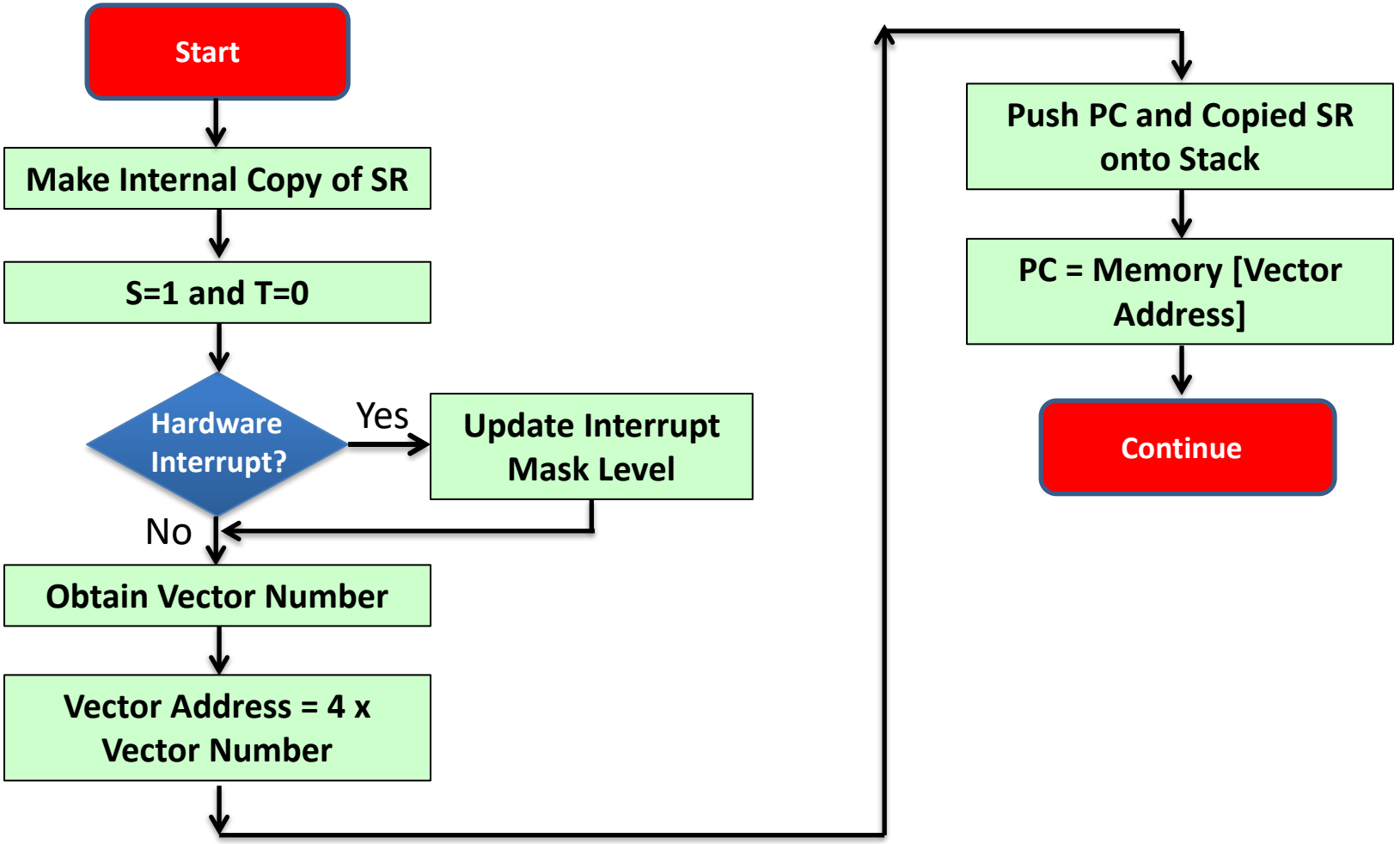
IRQ	Interrupt Mask	Recognized Levels
7 (highest)	111	7
6	110	7
5	101	6,7
4	100	5,6,7
3	011	4,5,6,7
2	010	3,4,5,6,7
1 (lowest)	001	2,3,4,5,6,7
No interrupt	000	1,2,3,4,5,6,7

Non-maskable interrupt

All interrupts are recognized. User programs run with mask 000

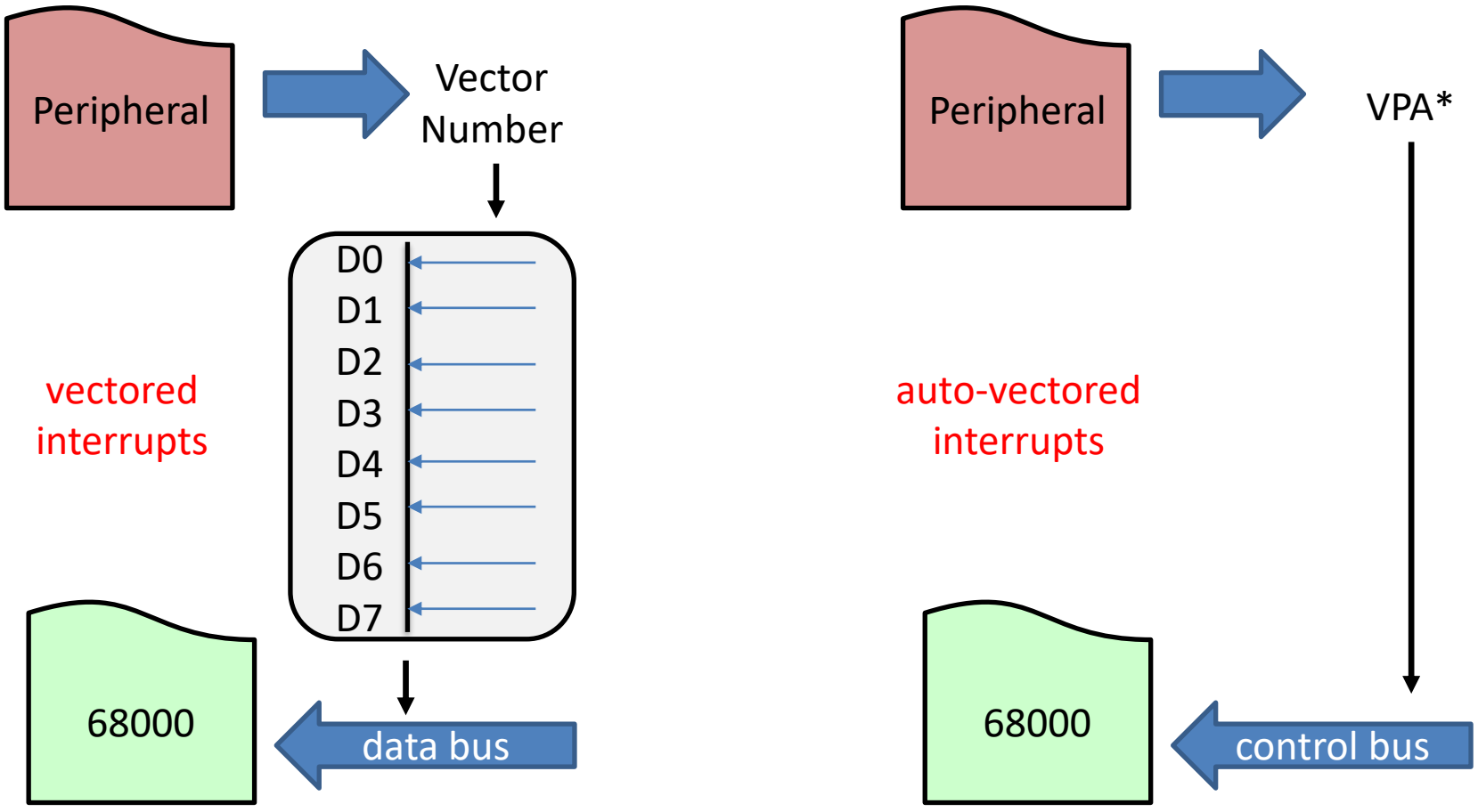
Slides are IP. Please do not copy, share, or post to website(s).

How are Interrupts and Exceptions Processed?



Slides are IP. Please do not copy, share, or post to website(s).

Vectored versus Auto-Vectored Interrupts



Slides are IP. Please do not copy, share, or post to website(s).

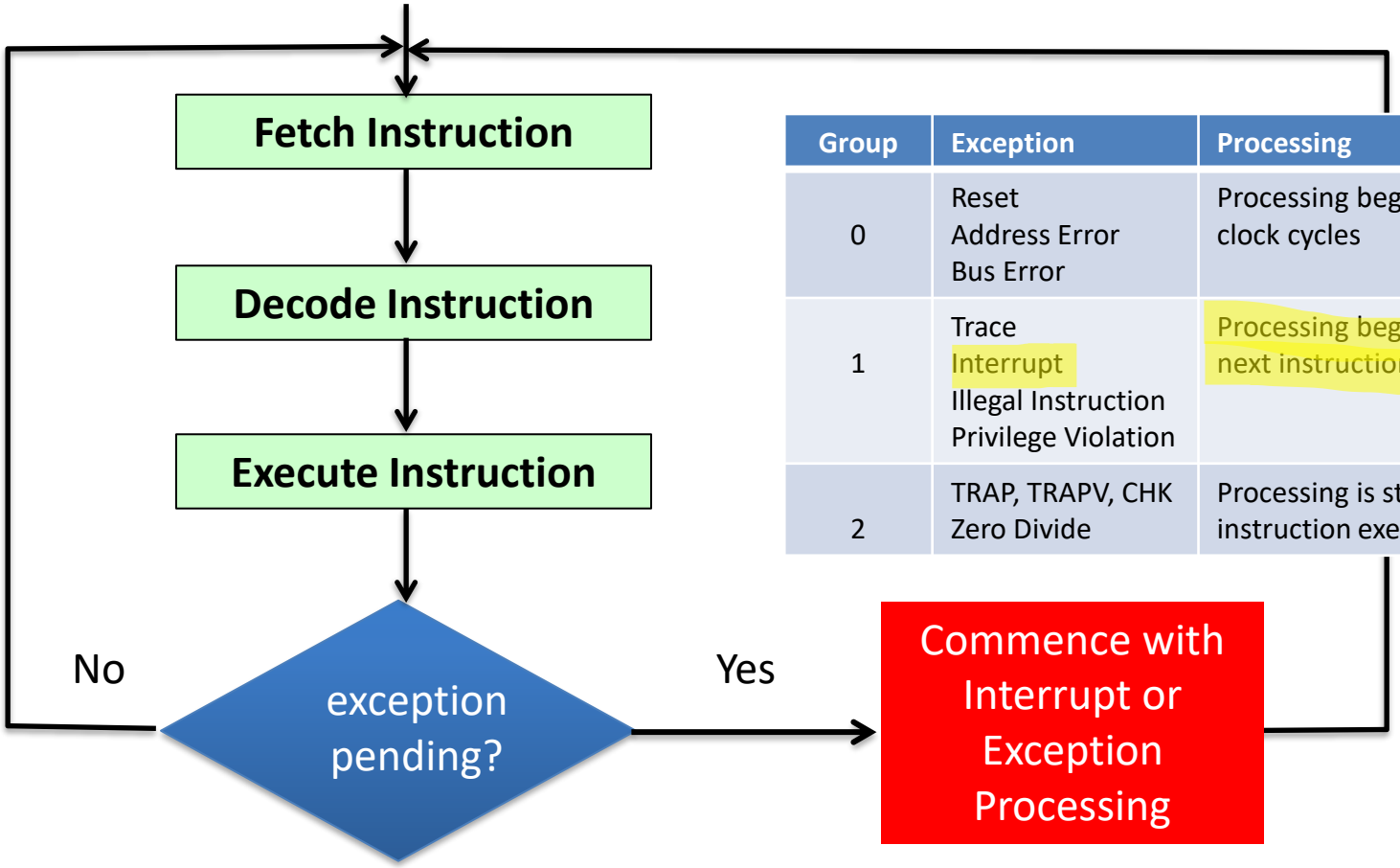
Review: 68000 Vector Table

vector number (Decimal)	address (Hex)	assignment
0	0000	RESET: initial supervisor stack pointer (SSP)
1	0004	RESET: initial program counter (PC)
2	0008	bus error
3	000C	address error
4	0010	illegal instruction
5	0014	zero divide
6	0018	CHK instruction
7	001C	TRAPV instruction
8	0020	priviledge violation
9	0024	trace
10	0028	1010 instruction trap
11	002C	1111 instruction trap
12*	0030	not assigned, reserved by Motorola
13*	0034	not assigned, reserved by Motorola
14*	0038	not assigned, reserved by Motorola
15	003C	uninitialized interrupt vector
16-23*	0040-005F	not assigned, reserved by Motorola
24	0060	spurious interrupt
25	0064	Level 1 interrupt autovector
26	0068	Level 2 interrupt autovector
27	006C	Level 3 interrupt autovector
28	0070	Level 4 interrupt autovector
29	0074	Level 5 interrupt autovector
30	0078	Level 6 interrupt autovector
31	007C	Level 7 interrupt autovector
32-47	0080-00BF	TRAP instruction vectors**
48-63	00C0-00FF	not assigned, reserved by Motorola
64-255	0100-03FF	user interrupt vectors

NOTES:
* No peripheral devices should be assigned these numbers
** TRAP #N uses vector number 32+N

Slides are IP. Please do not copy, share, or post to website(s).

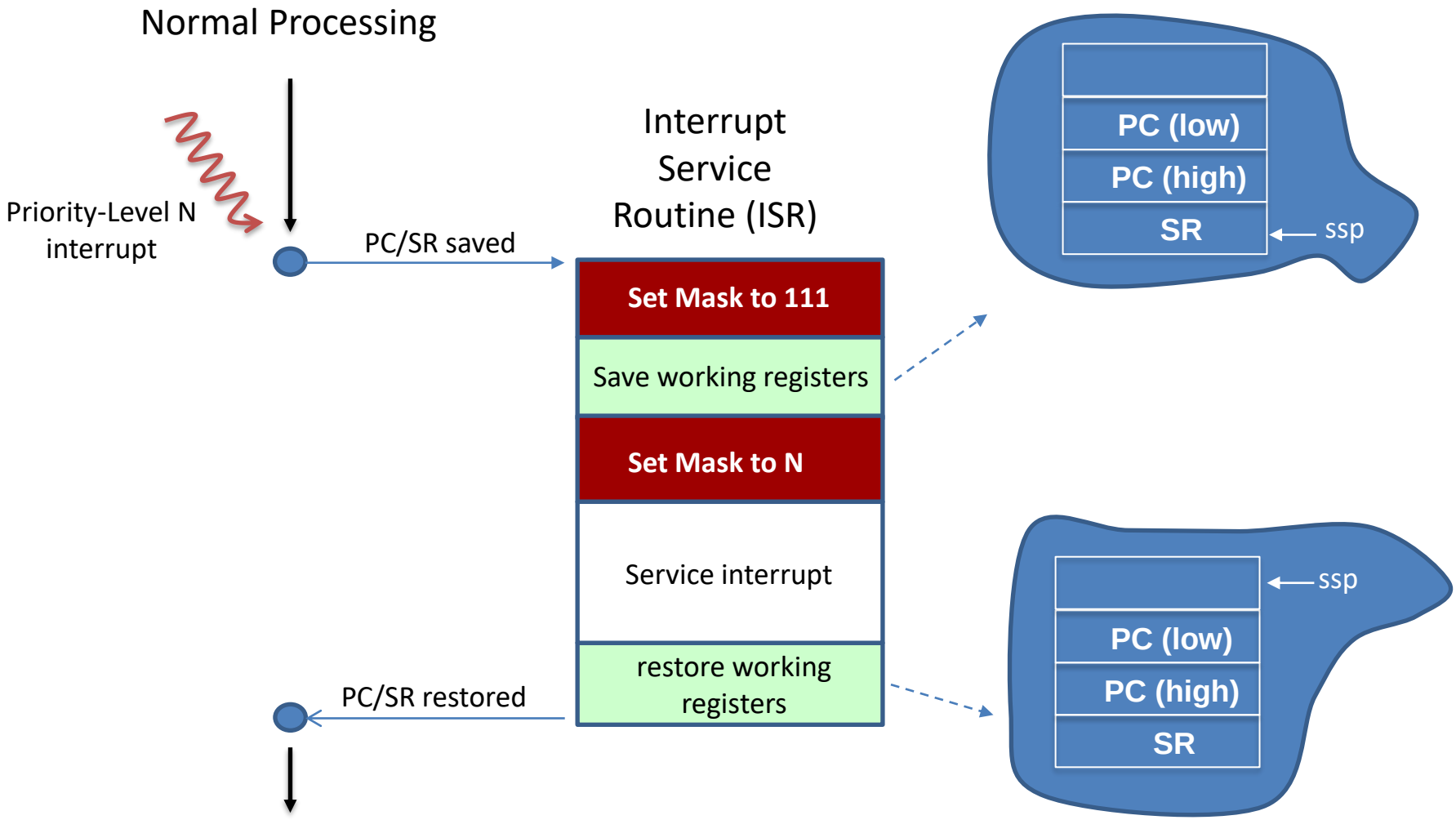
When are Interrupts Processed?



Group	Exception	Processing
0	Reset Address Error Bus Error	Processing begins within 2 clock cycles
1	Trace Interrupt Illegal Instruction Privilege Violation	Processing begins before the next instruction
2	TRAP, TRAPV, CHK Zero Divide	Processing is started by normal instruction execution

Slides are IP. Please do not copy, share, or post to website(s).

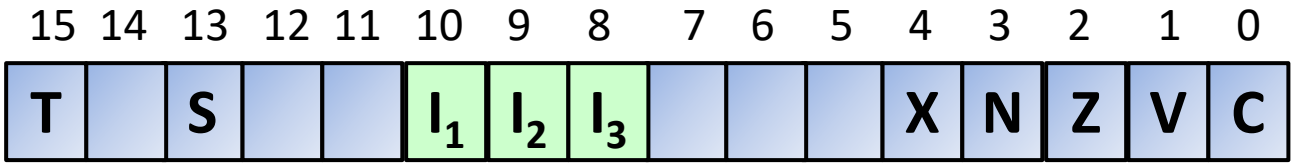
How are Interrupts are Serviced?



Slides are IP. Please do not copy, share, or post to website(s).

Problem

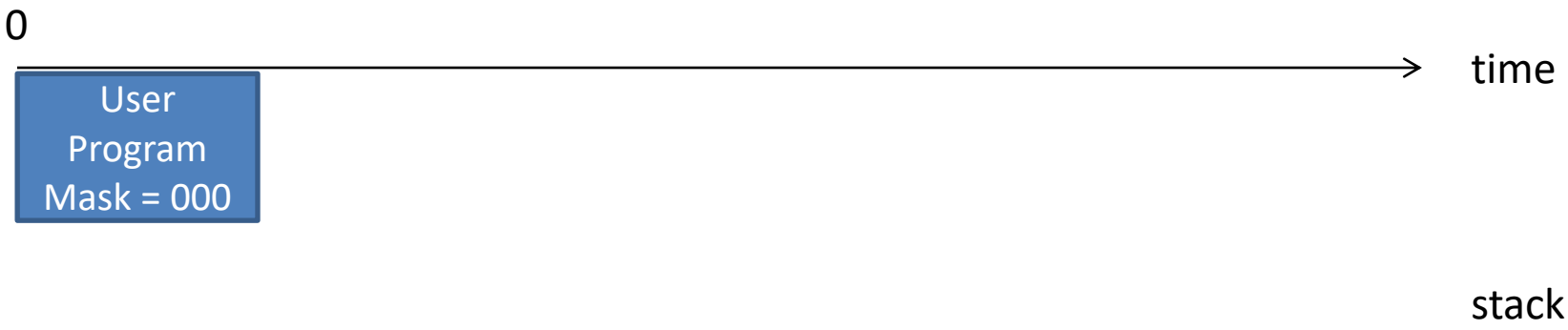
- Show how you would set the interrupt mask to level 5 without changing any of the other bits in the status register. Assume that the processor is in supervisor state.



Slides are IP. Please do not copy, share, or post to website(s).

Time Sequence Showing Multiple Interrupts

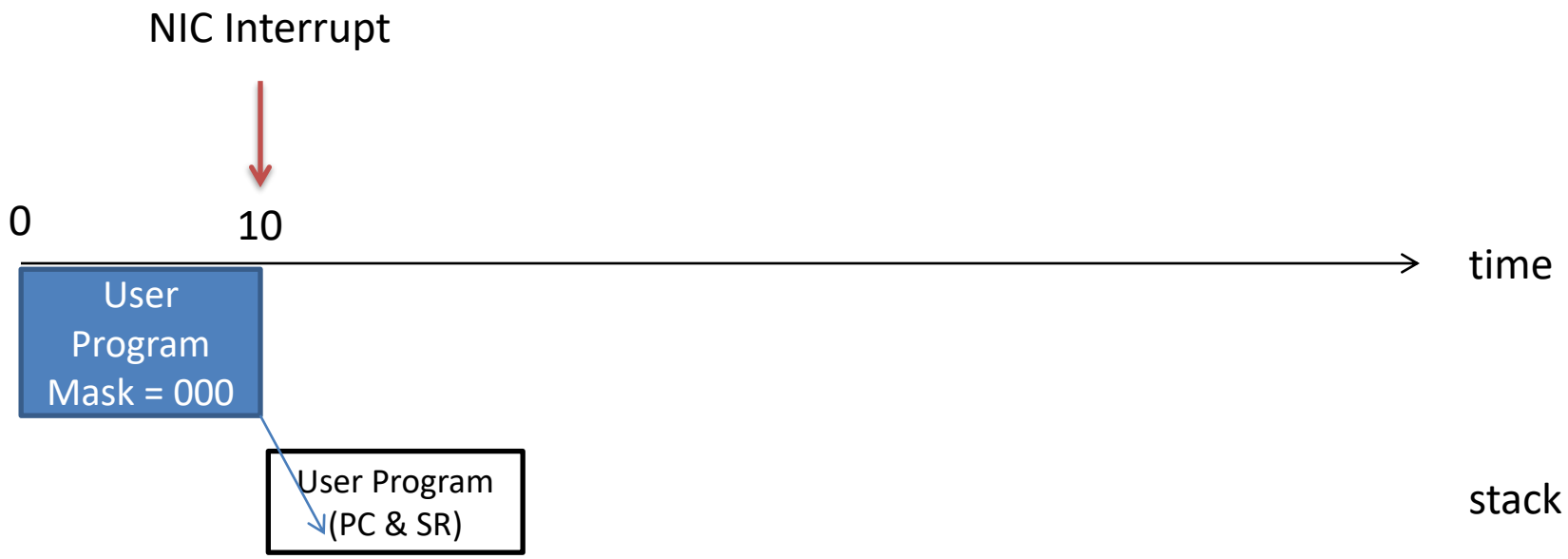
IRQ2 – Printer
IRQ4 – Disk
IRQ5 – NIC



Slides are IP. Please do not copy,
share, or post to website(s).

Time Sequence Showing Multiple Interrupts

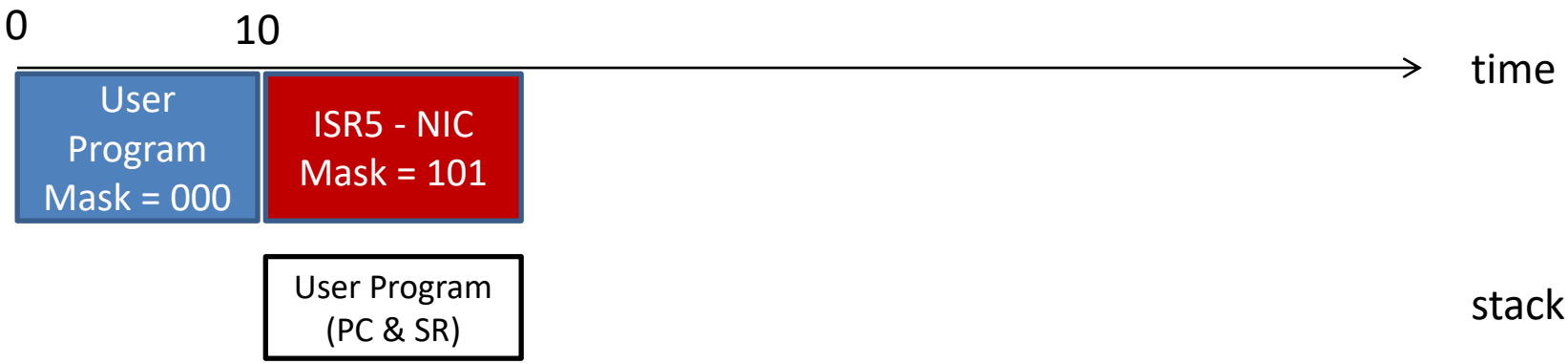
IRQ2 – Printer
IRQ4 – Disk
IRQ5 – NIC



Slides are IP. Please do not copy, share, or post to website(s).

Time Sequence Showing Multiple Interrupts

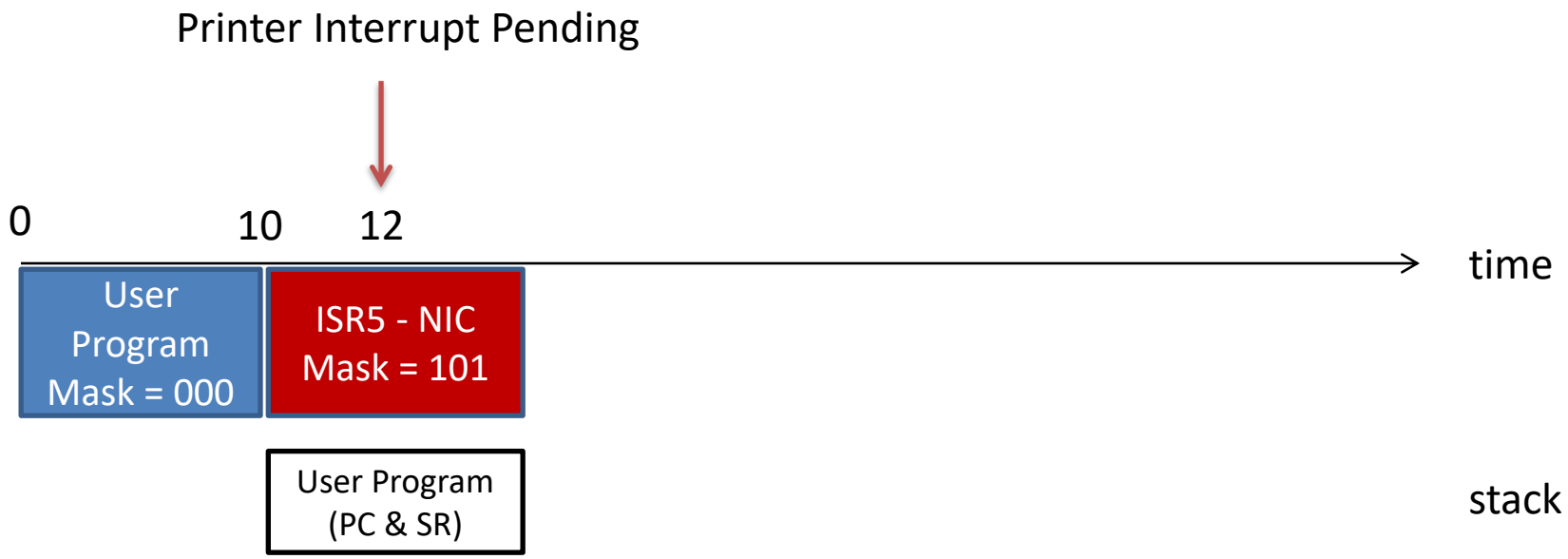
IRQ2 – Printer
IRQ4 – Disk
IRQ5 – NIC



Slides are IP. Please do not copy, share, or post to website(s).

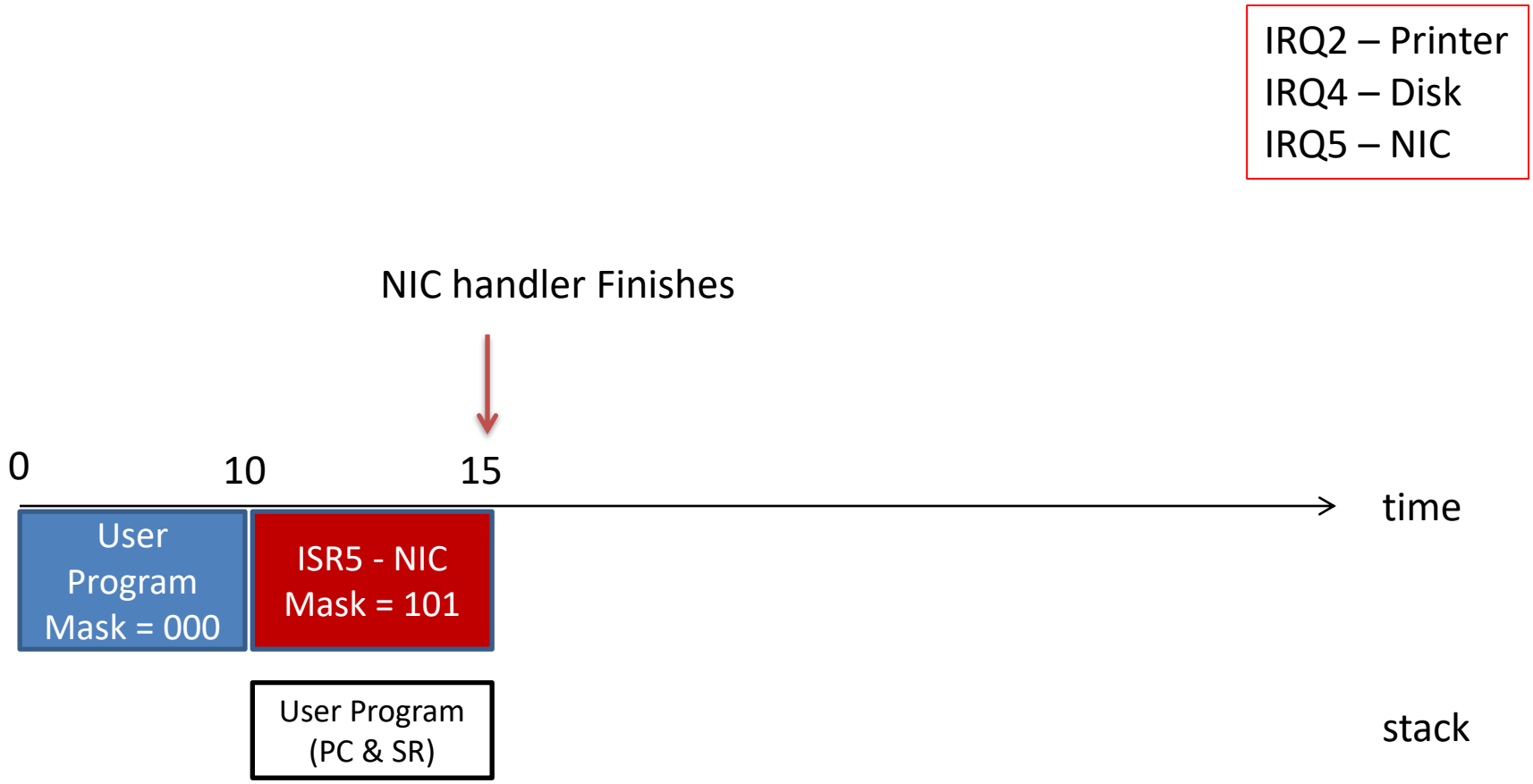
Time Sequence Showing Multiple Interrupts

IRQ2 – Printer
IRQ4 – Disk
IRQ5 – NIC



Slides are IP. Please do not copy, share, or post to website(s).

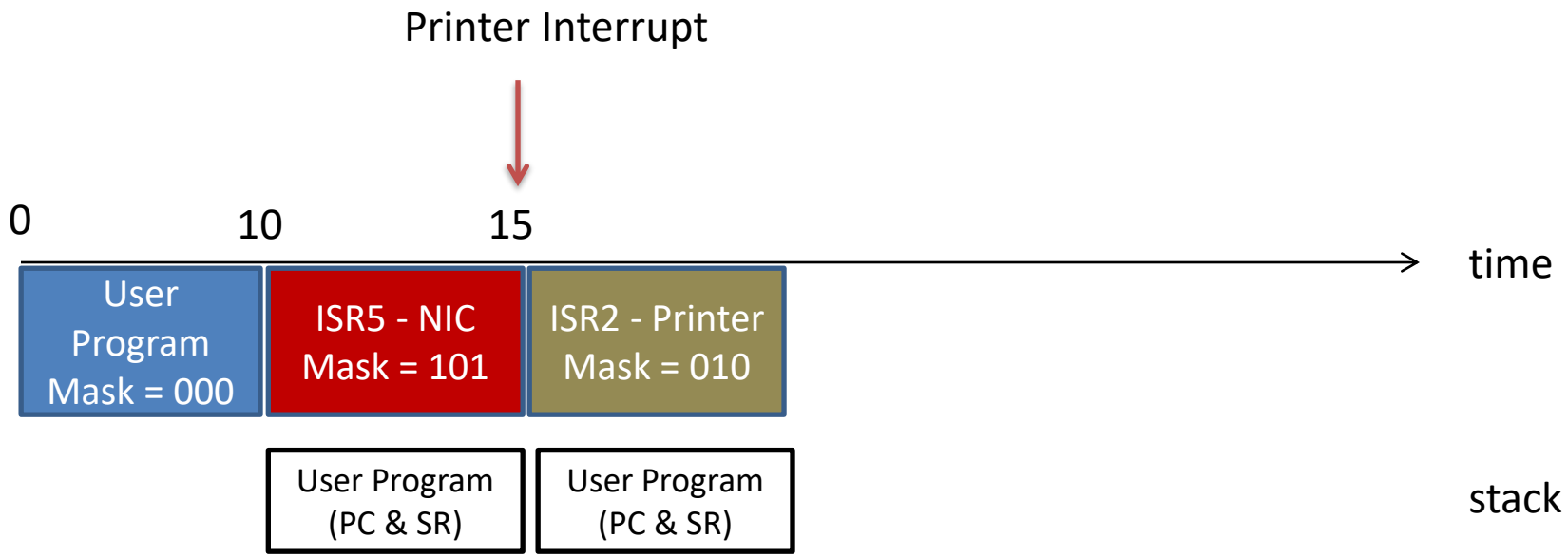
Time Sequence Showing Multiple Interrupts



Slides are IP. Please do not copy, share, or post to website(s).

Time Sequence Showing Multiple Interrupts

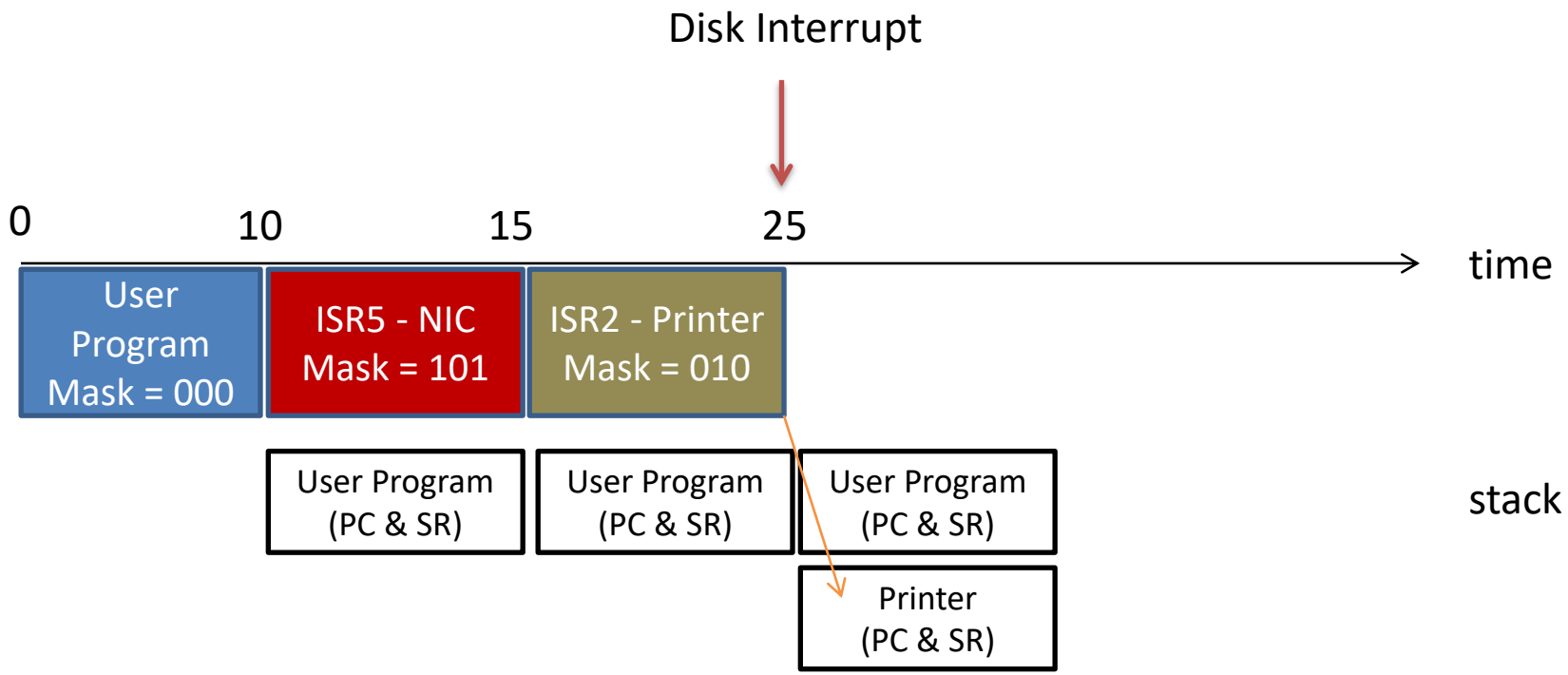
IRQ2 – Printer
IRQ4 – Disk
IRQ5 – NIC



Slides are IP. Please do not copy, share, or post to website(s).

Time Sequence Showing Multiple Interrupts

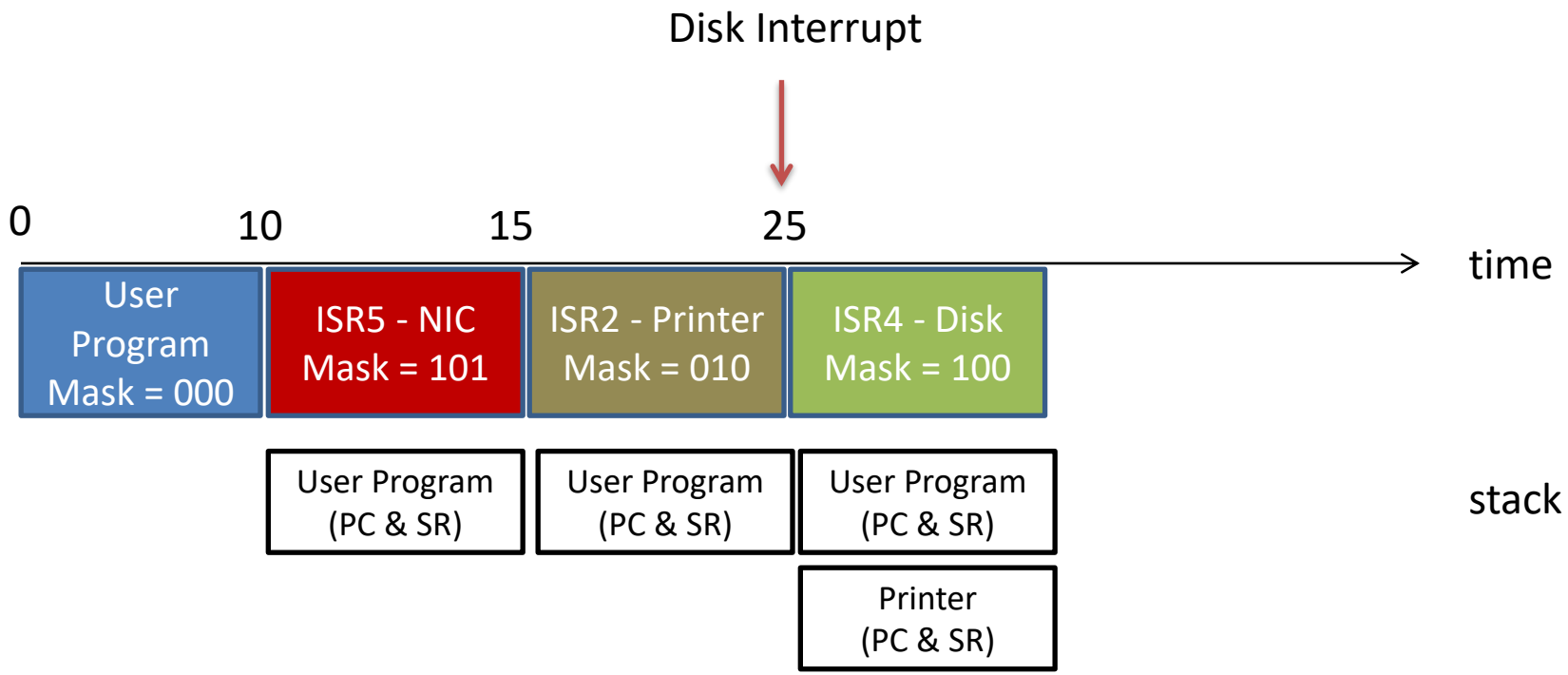
IRQ2 – Printer
IRQ4 – Disk
IRQ5 – NIC



Slides are IP. Please do not copy, share, or post to website(s).

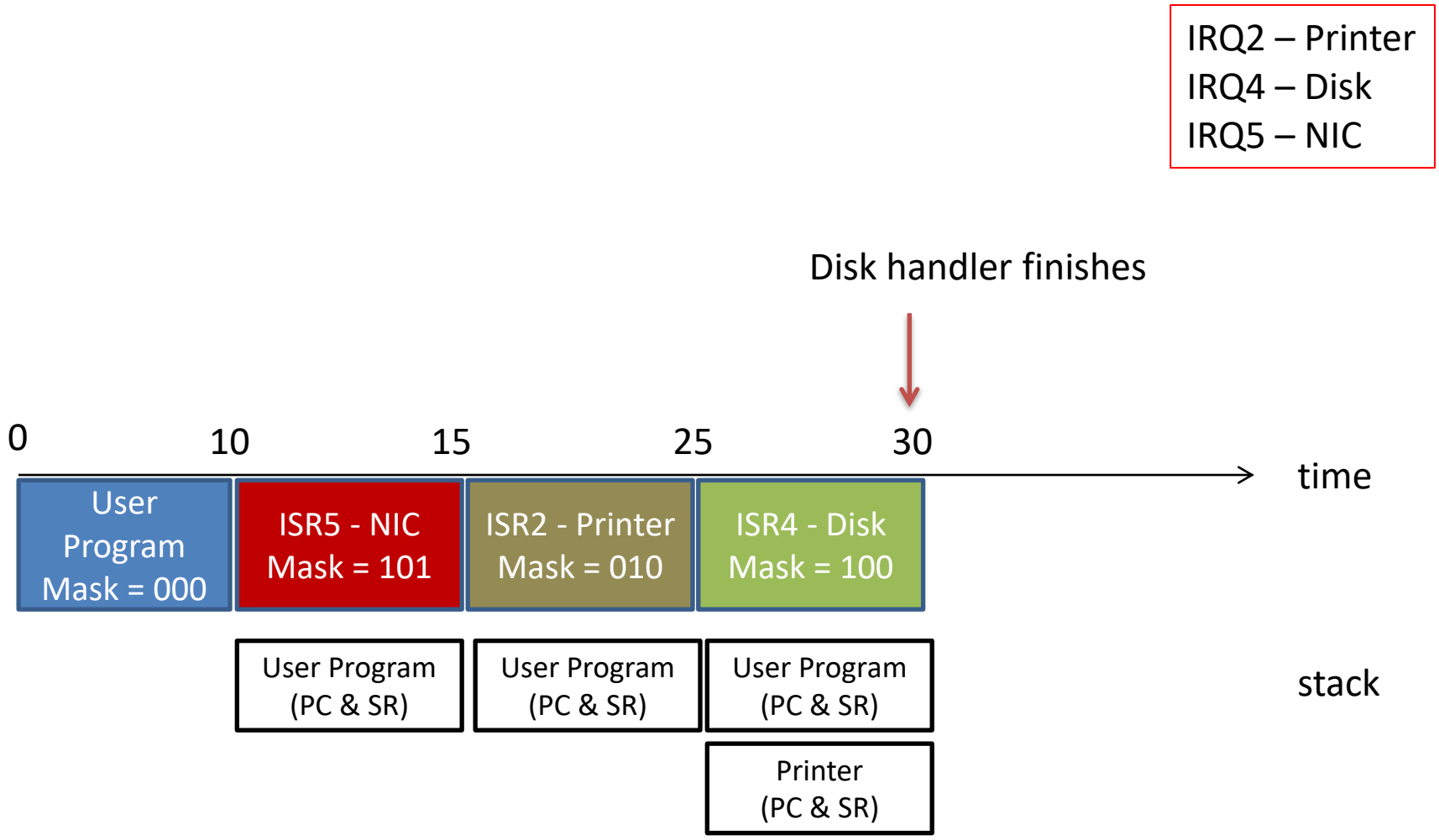
Time Sequence Showing Multiple Interrupts

IRQ2 – Printer
IRQ4 – Disk
IRQ5 – NIC



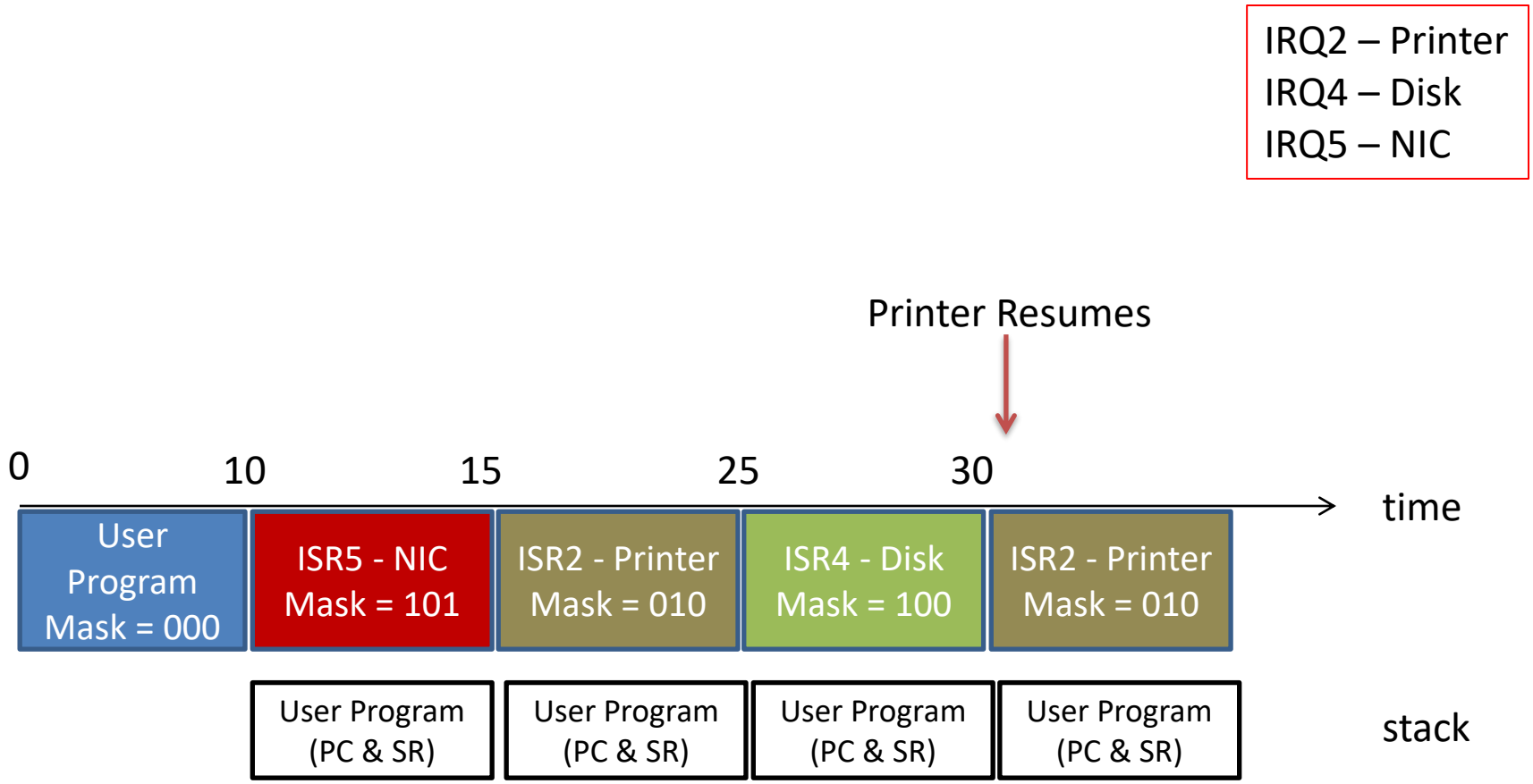
Slides are IP. Please do not copy, share, or post to website(s).

Time Sequence Showing Multiple Interrupts



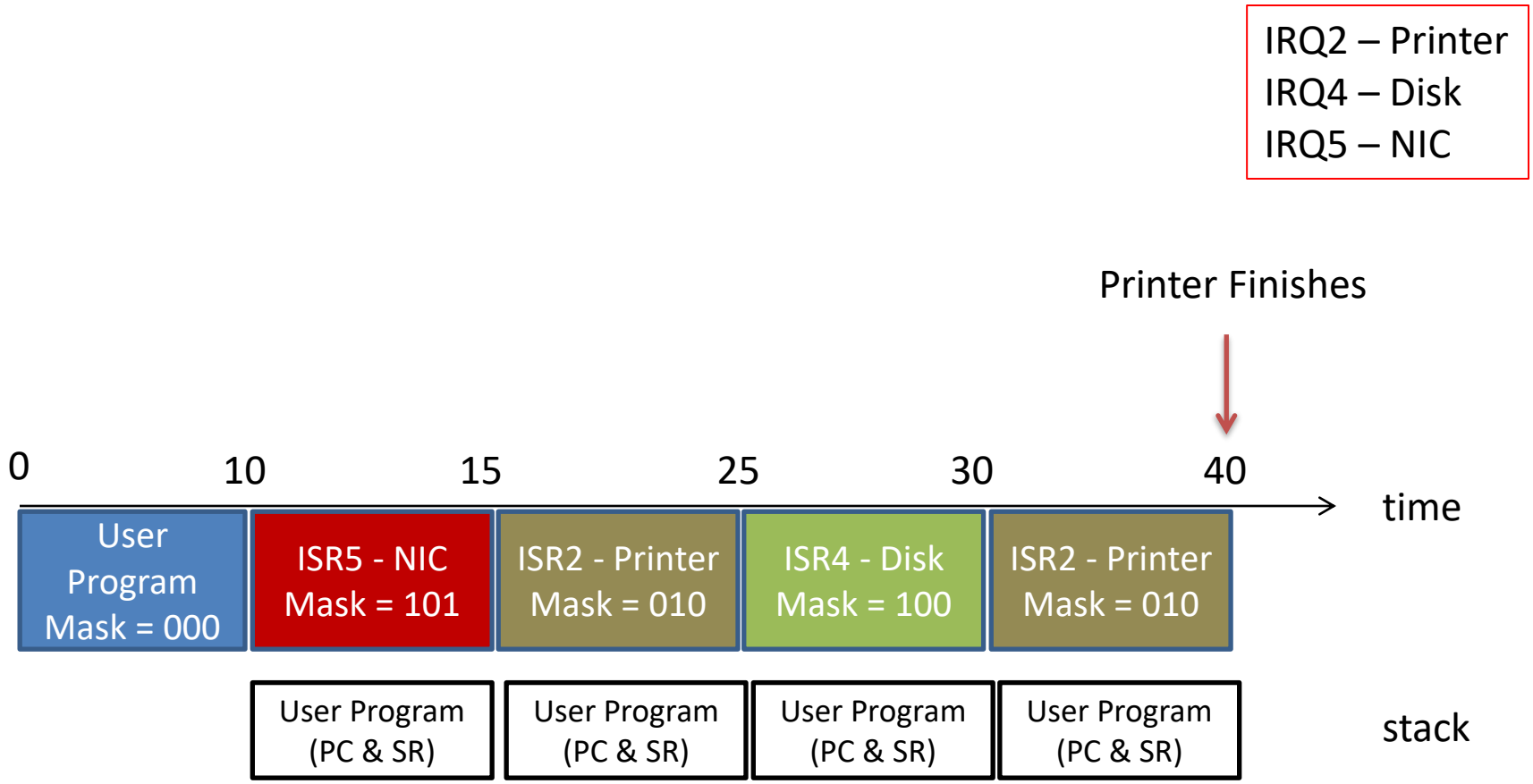
Slides are IP. Please do not copy, share, or post to website(s).

Time Sequence Showing Multiple Interrupts



Slides are IP. Please do not copy, share, or post to website(s).

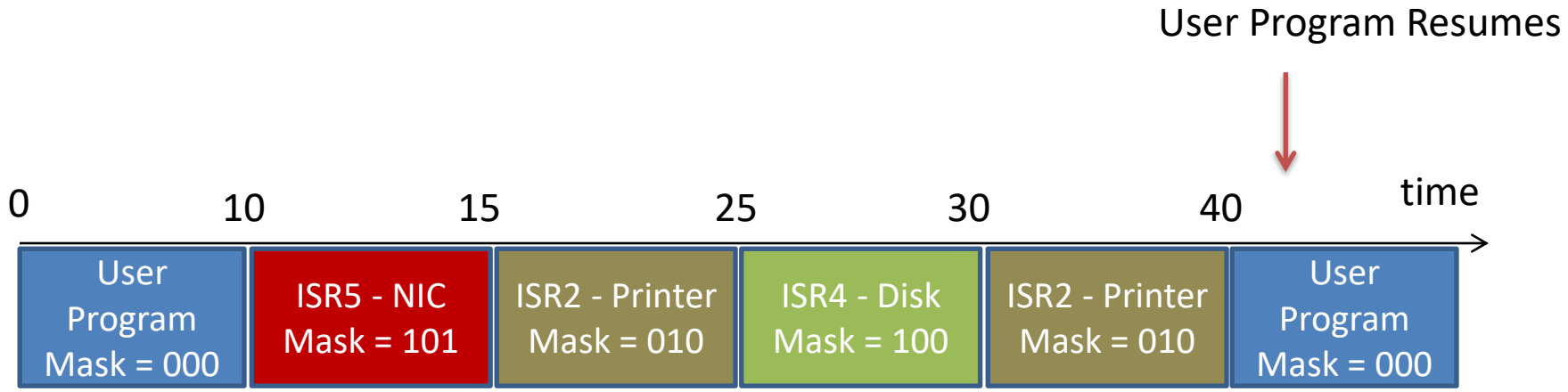
Time Sequence Showing Multiple Interrupts



Slides are IP. Please do not copy, share, or post to website(s).

Time Sequence Showing Multiple Interrupts

IRQ2 – Printer
IRQ4 – Disk
IRQ5 – NIC



Slides are IP. Please do not copy, share, or post to website(s).

Problem

- Write a level-5 autovector interrupt-service routine to display the message “Interrupts are cool!” on the Easy68K terminal. Then use a timer to cause a level-5 interrupt to occur once every second.

Slides are IP. Please do not copy, share, or post to website(s).

Review: 68000 Vector Table

vector number (Decimal)	address (Hex)	assignment
0	0000	RESET: initial supervisor stack pointer (SSP)
1	0004	RESET: initial program counter (PC)
2	0008	bus error
3	000C	address error
4	0010	illegal instruction
5	0014	zero divide
6	0018	CHK instruction
7	001C	TRAPV instruction
8	0020	priviledge violation
9	0024	trace
10	0028	1010 instruction trap
11	002C	1111 instruction trap
12*	0030	not assigned, reserved by Motorola
13*	0034	not assigned, reserved by Motorola
14*	0038	not assigned, reserved by Motorola
15	003C	uninitialized interrupt vector
16-23*	0040-005F	not assigned, reserved by Motorola
24	0060	spurious interrupt
25	0064	Level 1 interrupt autovector
26	0068	Level 2 interrupt autovector
27	006C	Level 3 interrupt autovector
28	0070	Level 4 interrupt autovector
29	0074	Level 5 interrupt autovector
30	0078	Level 6 interrupt autovector
31	007C	Level 7 interrupt autovector
32-47	0080-00BF	TRAP instruction vectors**
48-63	00C0-00FF	not assigned, reserved by Motorola
64-255	0100-03FF	user interrupt vectors

NOTES:
* No peripheral devices should be assigned these numbers
** TRAP #N uses vector number 32+N

Slides are IP. Please do not copy, share, or post to website(s).

Summary

- Interrupts are essentially a hardware generated function call
  - Asynchronous events
  - Must be acknowledged by the CPU
  - Processed at the end of the current instruction cycle
  - Causes current program execution to halt while ISR is executed
  - Normal program execution resumes at conclusion of ISR
- Interrupts are used for infrequent events that require a few bytes to be transferred between device and memory/processor
  - Keyboard, mouse, timer, etc.
- Interrupts versus polling
  - Polling wastes times waiting for the I/O device to become ready, whereas interrupts use the CPU only when they are ready
  - Polling is generally messy when multiple devices exist or system is busy, whereas interrupts allow the CPU to synchronize with outside events

Slides are IP. Please do not copy, share, or post to website(s).

Summary

- Interrupts are prioritized so that
  - User programs can always be interrupted, since they run with the interrupt mask set to zero
  - A lower priority interrupt-service routine can always be interrupted by a higher-priority interrupt request
  - A higher priority interrupt-service routine cannot be interrupted by a lower-priority interrupt request
- A non-maskable interrupt cannot be deferred
  - Used by most time-critical event, like shut down on power loss
- 68000 IRQs
  - A total of 7 levels encoded on 3 bus signals (IPL*0, IPL*1,IPL*2)
  - Level 7 is non-maskable interrupt (NMI)
  - Both vectored and auto-vectored interrupts supported

Slides are IP. Please do not copy, share, or post to website(s).