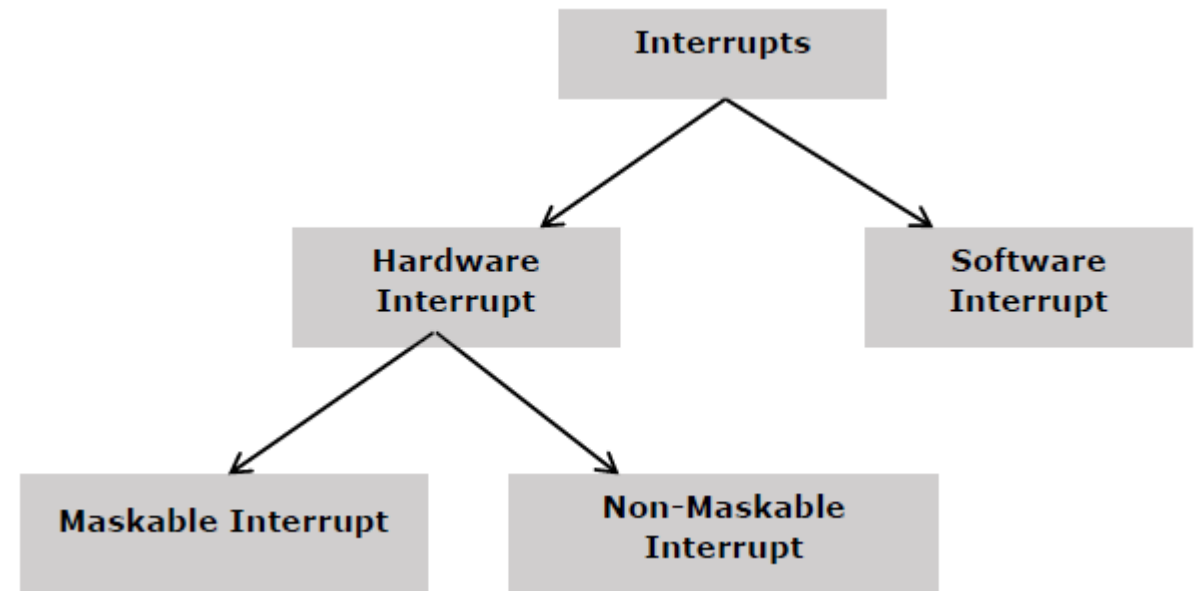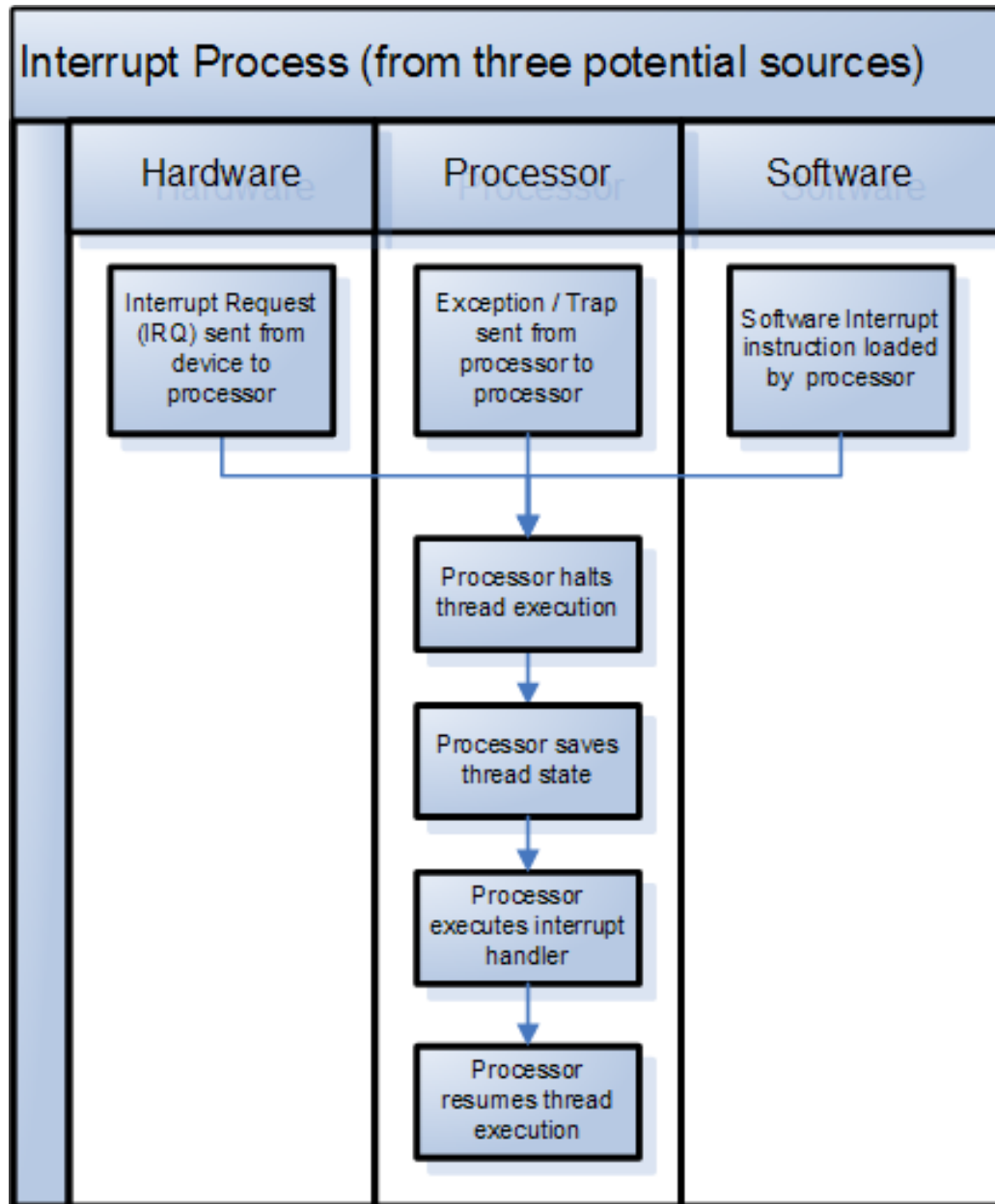# External interruption
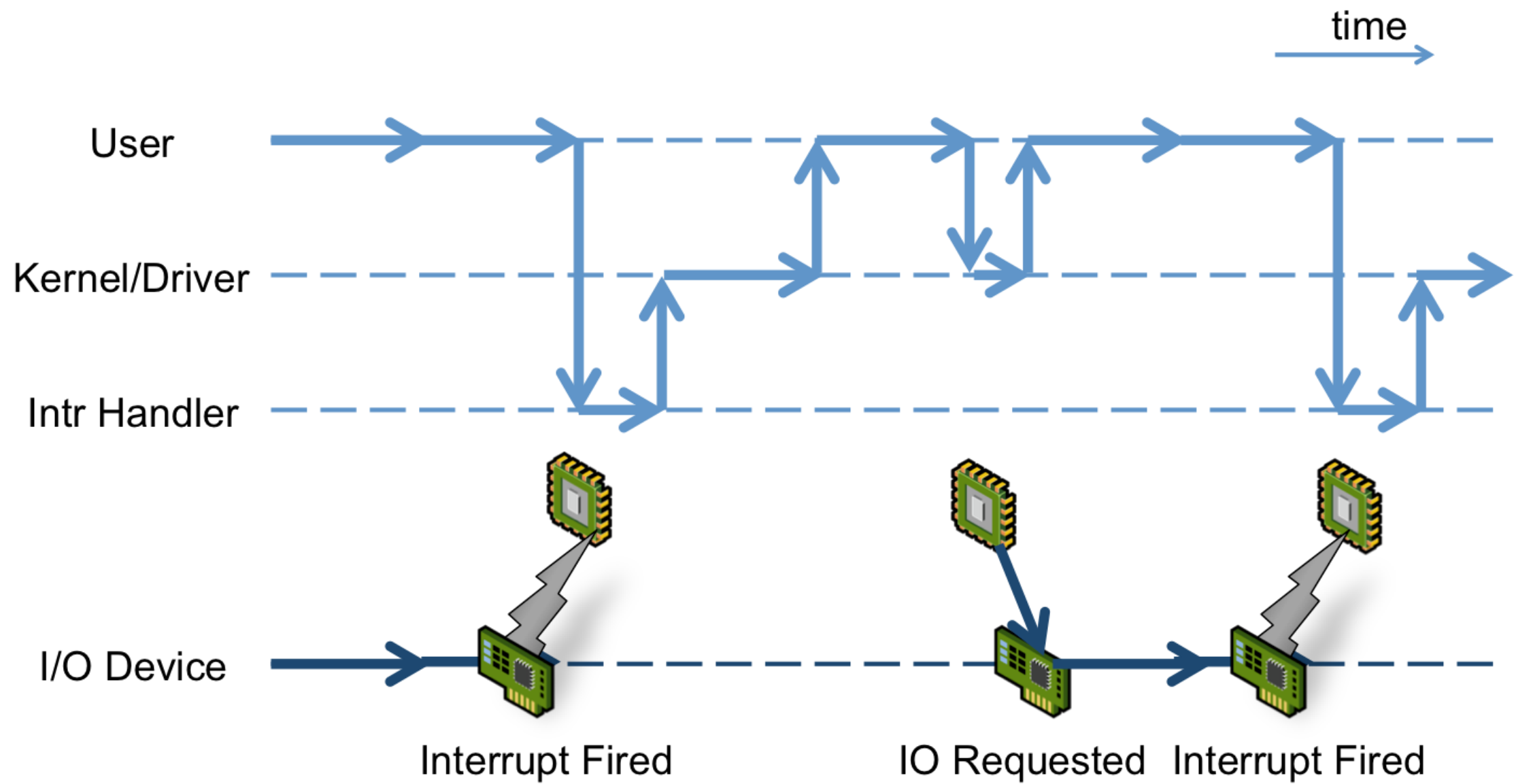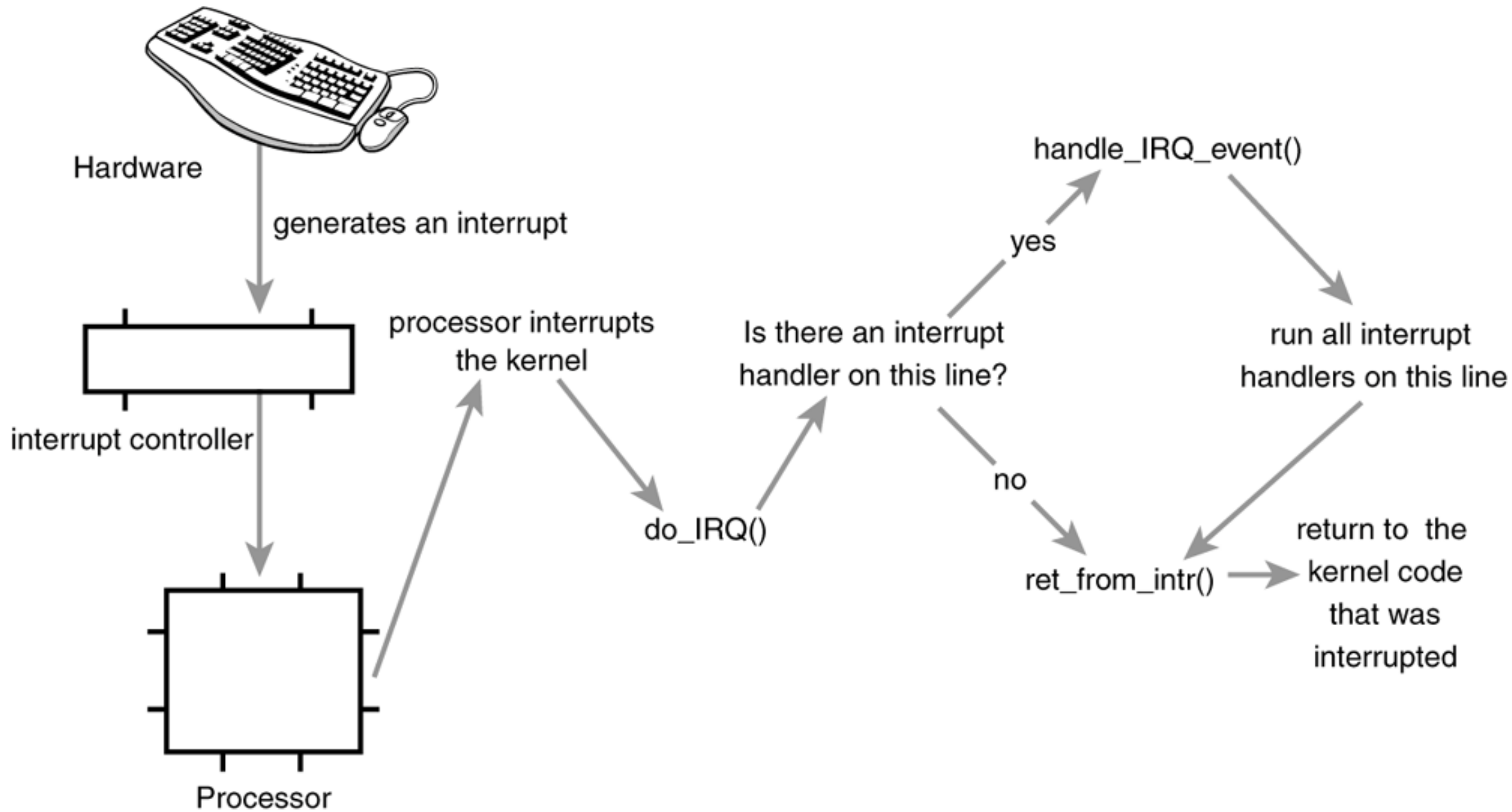
KukarcevKS, RCSE 4167 2018

What is it?

- Interrupt is a **signal** to the processor
- Emitted by **hardware** or **software**
- Indicates an **event**
- Always has **high-priority**
- Each interrupt has **interrupt handler**

Typical usage

- System timers
- Disk I/O
- Power-off signals
- Traps
- Network I/O
- Mouse movement
- Keyboard typing
- Different architecture emulation

Interrupt Process (from three potential sources)

Hardware	Processor	Software
Interrupt Request (IRQ) sent from device to processor	Exception / Trap sent from processor to processor	Software Interrupt instruction loaded by processor
	Processor halts thread execution	
	Processor saves thread state	
	Processor executes interrupt handler	
	Processor resumes thread execution	

Interrupts

Hardware Interrupt

Software Interrupt

Maskable Interrupt

Non-Maskable Interrupt

time

User

Kernel/Driver

Intr Handler

I/O Device

Interrupt Fired   IO Requested   Interrupt Fired

Hardware

generates an interrupt

interrupt controller

Processor

processor interrupts
the kernel

do_IRQ()

Is there an interrupt
handler on this line?

handle_IRQ_event()

yes

no

run all interrupt
handlers on this line

ret_from_intr()

return to the
kernel code
that was
interrupted

External interrupts

- It's an interrupt happens from outside interference.

1. From user
2. From peripherals
3. From hardware
4. From network

- Can be produces automatically, as a computer reads through program instructions.

System boots up

Initialization
(Initialize GPIO and
Interrupt Controller)

**Main
Program**
(Display the
Alphabet
Sequentially)

Interrupt

Returns

ISR
(Display which button
was pressed)