Programme: B.Sc.

Subject : Computer Science.

Semester : V

Paper Code: CSD 104

Paper Title : Embedded Systems

Unit II : Interrupts

Module Name: Microprocessor Architecture ;Interrupt Basics Part I

Module No : 05

Contents

- ✓ Introduction to Microprocessor Architecture
- ✓ Basics of Interrupts
- ✓ ISR (Interrupt Service Routine)
- ✓ Categories and Types of interrupts

Microprocessor Architecture.

Assembly language: is a human readable form of instructions that a microprocessor understands.

Assembler: translates assembly language instructions into binary numbers, each binary language instruction turns into single instruction for the microprocessor.

Registers/General purpose registers: holds data for the processes currently executing in the microprocessor.

Program counter: this registers keeps a track of the address of next instruction to be executed by the microprocessor.

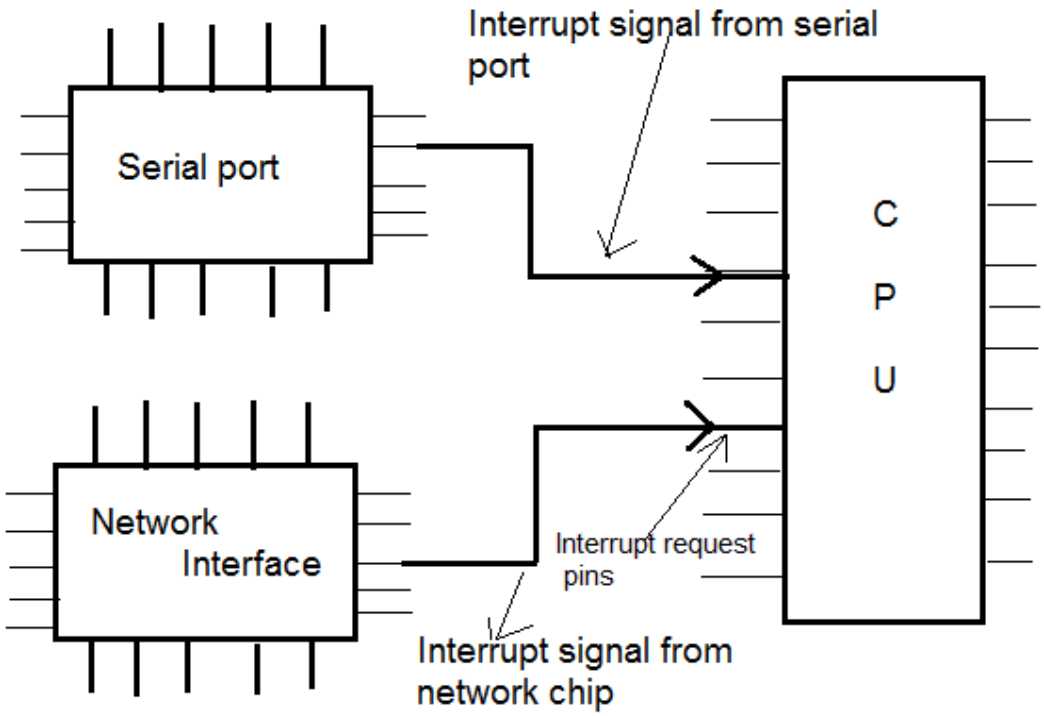
Interrupts: an interrupt is a signal to the processor emitted by hardware or software indicating an event that needs immediate attention.

Interrupt Service Routine (ISR) : fig 5.1

- **ISR**: is a subroutine written by a programmer to service the interrupting source.
- **Series of steps microprocessor follows when interrupt occurs:**
    1. Finish current instruction
    2. Save address of next instruction on stack
    3. Executes (DI) , disable interrupts
    4. Attends the interrupt:
        A. fetch address of interrupt routine and execute
        B. Acknowledge the interrupting device (INTA)
    5. Executes EI, enable interrupts
    6. Retrieve the saved address at step 2

7. Resumes execution.

Understanding ISR with the help of diagram. Fig. 5.2



In the above figure we can see that a microprocessor has various input pins, through which the hardware communicates with the microprocessor. In the figure we have the Serial port and the Network port.

Interrupts are triggered when certain events occur in the hardware.

e.g. when a serial chip wants to send data to a microprocessor and needs the microprocessor to read it from where it is stored inside the serial port chip, and store it in memory , it signals the microprocessor, asking for its attention to serve the signal.

The microprocessor understands that there is an interrupt to be handled, and performs the steps in fig. 5.1.

The ISR jumps to an **interrupt service routine** (**ISR**) or interrupt handler, there by serving the Serial port. Once the ISR is executed , the microprocessor resumes its execution by fetching the address of the next instruction to be executed from the stack. Similarly the

network when chip sends a interrupt, asking for microprocessors assistance for similar events.

Each of these pins of the chip is asserted when it requires service. these pins are attached to the input pins on the microprocessor know as Interrupt request; IRQ.

Category of interrupts:

1. Internal Interrupts / Software interrupts

   Namely: RST 0, RST 1, RST 2, RST 3, RST 4, RST 5, RST 6, RST 7

2. External interrupts/ hardware interrupts

   Namely: RST 7.5, RST 6.5, RST 5.5, TRAP, INTR

Internal interrupts are also referred to as a software interrupt. it is caused either by an exceptional condition or a special instruction in the instruction set which causes an interrupt when it is executed by the processor.

For example, if the processor's arithmetic logic unit runs a command to divide a number by zero, to cause a divide-by-zero exception, thus causing the computer to abandon the calculation or display an error message. Software interrupt instructions work similar to subroutine calls. Software interrupts are RST 0, RST 1, RST 2, RST 3, RST 4, RST 5, RST 6, RST 7

External interrupts are referred to as hardware interrupts. A hardware interrupt is an electronic alerting signal sent to the processor from an external device, like a disk controller or an external peripheral.

For example, when we press a key on the keyboard or move the mouse, they trigger hardware interrupts which cause the processor to read the keystroke or mouse position. Hardware interrupts are RST 7.5, RST 6.5, RST 5.5, TRAP, INTR

Types of Interrupts

1. Vectored interrupts
2. Non - vectored interrupts
3. Maskable interrupts
4. Non -Maskable interrupts

Vectored Interrupts: are those which have fixed vector address (starting address of sub-routine) and after executing these, program control is transferred to that address.

Namely : RST 7.5, RST 6.5, RST 5.5, TRAP (hardware interrupts)

: RST 0, RST 1, RST 2… RST 7. (software Interrupts)

Non-Vectored Interrupts: are those in which vector address is not predefined. The interrupting device gives the address of sub-routine for these interrupts. *INTR* is the only non-vectored interrupt in 8085 microprocessor.

Maskable Interrupts: are those which can be disabled or ignored by the microprocessor. *INTR, RST 7.5, RST 6.5, RST 5.5* are maskable interrupts in 8085 microprocessor.

Non-Maskable Interrupts: are those which cannot be disabled or ignored by microprocessor. *TRAP* is a non-maskable interrupt. It is used in critical power failure conditions.

Categorization of Interrupt. fig 5.3

Sr. No	Name	Priority	Masking
1	TRAP	HIGH	Non Maskable
2	RST 7.5	↓	Maskable
3	RST 6.5	↓	Maskable
4	RST 5.5	↓	Maskable
5	INTR	Least	Maskable