Microprocessor suggestion

# 2nd internal solve

1. What is interrupt? Discuss the different types of interrupt in 8085 microprocessor.

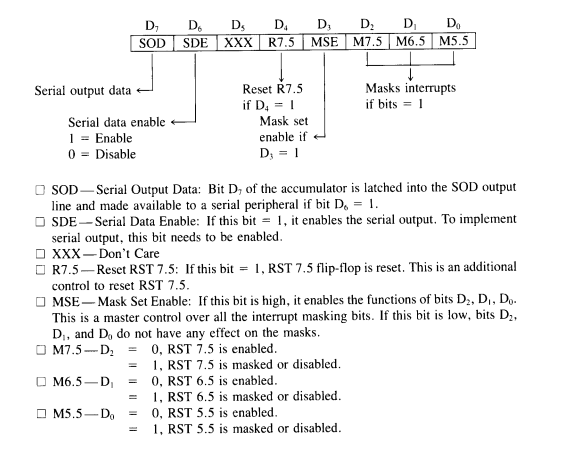
Ans. Interrupt is a mechanism by which an I/O or an instruction can suspend the normal execution of processor and get itself serviced. Generally, a particular task is assigned to that interrupt signal. In the microprocessor based system the interrupts are used for data transfer between the peripheral devices and the microprocessor.

Interrupts can be basically classified into two types: -

1. Maskable Interrupt (Can be delayed or Rejected)
2. Non-Maskable Interrupt (Cannot be delayed or rejected)

Interrupts can also be classified under vectored interrupt or non-vectored interrupt: -

1. Vectored interrupt: The address of the subroutine is hardwired.
2. Non-Vectored interrupt: The address of the subroutine is externally provided by the device.
3. Write down bit pattern of SIM instruction? What is the content for accumulator for SIM instruction to enable RST 5.5 interrupt of Intel 8085 microprocessor.

Ans.

The accumulator content should be 00001110 in order to enable RST 5.5 instruction.

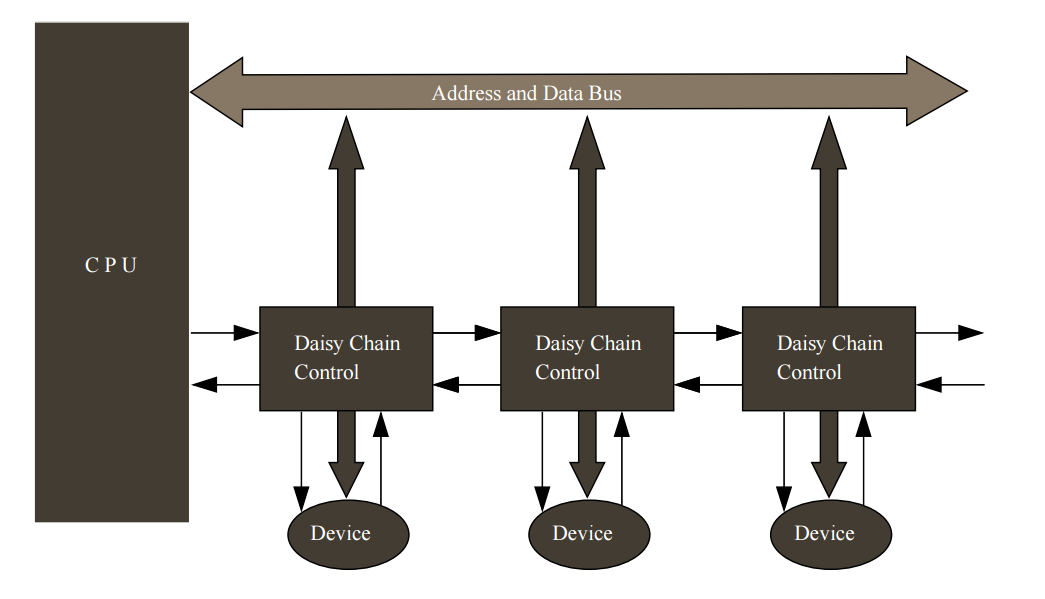
1. How a process “Daisy Chain” is used to resolve the multiple interrupts? Explain.

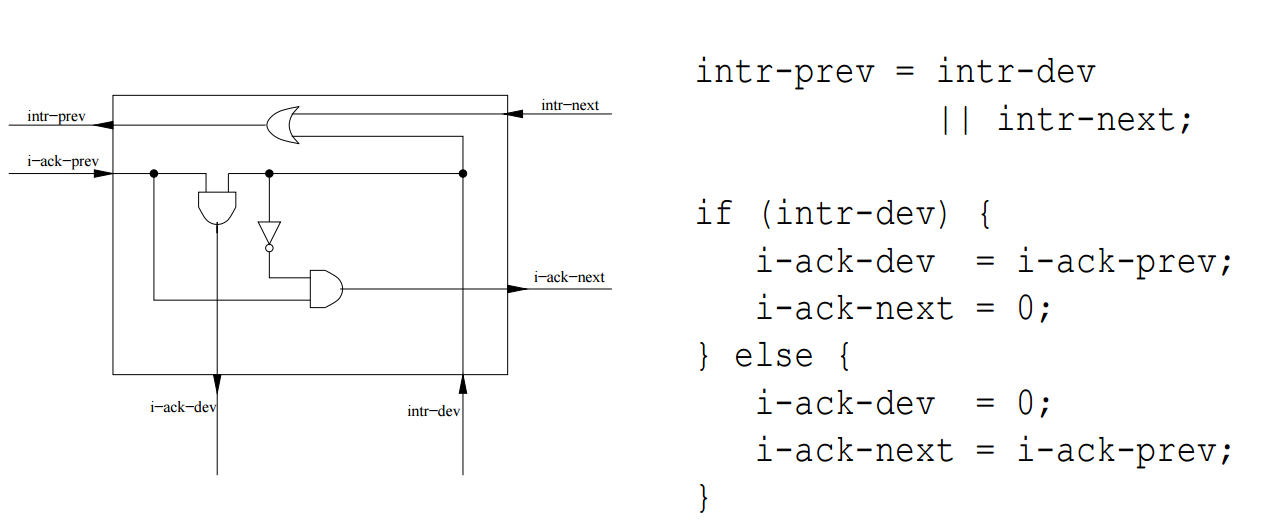
Ans. Daisy chaining is used for level sensitive interrupts, which act like a wired 'OR' gate. Any requesting device can take the interrupt line low, and keep it asserted low until it is serviced.

Because more than one device can assert the shared interrupt line simultaneously, some method must be employed to ensure device priority. This is done using the interrupt acknowledge signal generated by the processor in response to an interrupt request.

Each device is connected to the same interrupt request line, but the interrupt acknowledge line is passed through each device, from the highest priority device first, to the lowest priority device last.

After preserving the required registers, the microprocessor generates an interrupt acknowledge signal. This is gated through each device. If device 1 generated the interrupt, it will place its identification signal on the data bus, which is read by the processor, and used to generate the address of the interrupt-service routine. If device 1 did not request the servicing, it will pass the interrupt acknowledge signal on to the next device in the chain. Device 2 follows the same procedure, and so on.





1. What are the basic functions, which a DMA controller which a DMA controller is supposed to perform for DMA data transfer?

Ans.

During a block input byte transfer, the following sequence occurs as the data byte is sent from the interface to the memory:

1. The interface sends the DMA controller a request for DMA service.
2. A Bus request is made to the HOLD pin (active High) on the 8085 microprocessor and the controller gains control of the bus.
3. A Bus grant is returned to the DMA controller from the Hold Acknowledge (HLDA) pin (active High) on the 8086 microprocessor.
4. The DMA controller places contents of the address register onto the address bus.
5. The controller sends the interface a DMA acknowledgment, which tells the interface to put data on the data bus. (For an output it signals the interface to latch the next data placed on the bus.)
6. The data byte is transferred to the memory location indicated by the address bus.
7. The interface latches the data.
8. The Bus request is dropped, the HOLD pin goes Low, and the controller relinquishes the bus.
9. The Bus grant from the 8086 microprocessor is dropped and the HLDA pin goes Low.
10. The address register is incremented by 1.
11. The byte count is decremented by 1.
12. If the byte count is non-zero, return to step 1, otherwise stop.
13. Write the difference between memory mapped IO and IO mapped IO.

Ans.

|  |  |
| --- | --- |
| MEMORY MAPPED IO | I/O mapped io |
| Memory mapped I/O is mapped into the same address space as program memory and/or user memory, and is accessed in the same way | Port mapped I/O uses a separate, dedicated address space and is accessed via a dedicated set of microprocessor instructions. |
| The advantage to this method is that every instruction which can access memory can be used to manipulate an I/O device. | The advantage to this system is that less logic is needed to decode a discrete address and therefore less cost to add hardware devices to a machine. |
| The disadvantage to this method is that the entire address bus must be fully decoded for every device. | A slight disadvantage because more instructions are required to accomplish the same task. For instance, if we wanted to test one bit on a memory mapped port, there is a single instruction to test a bit in memory, but for ports we must read the data into a register, then test the bit. |
|  |  |

1. Write instruction to multiply by 4 to a hex number 0A using RAL instruction.

Ans.

|  |  |
| --- | --- |
| OPCODE | OPERAND |
| MVI A | 0A |
| STC | - |
| CMC | - |
| RAL | - |
| RAL | - |
| RST 3 | - |

1. Write down the difference between RST and CALL instruction.

Ans.

1. RST is a 1 byte instruction and CALL is a 3 byte instruction.
2. RST instruction takes 3 machine cycle to complete whereas CALL instruction takes 5 machine cycle.
3. RST instruction does not saves the executing memory address in the stack whereas CALL instruction does in order to resume the execution back.
4. Write a program to save the value of the accumulator along with flags. Subtract 40H to the contents of B register. Restore the value of PSW.

Ans.

|  |  |
| --- | --- |
| OPCODE | OPERAND |
| LXI SP | 8000H |
| PUSH PSW |  |
| MOV A, B |  |
| SUI | 40H |
| MOV B, A |  |
| POP A |  |
| RST 3 |  |

1. The last address location of 16K Byte memory space is 9200H. What will be the first address location?

Ans. According to the given data the last memory location is 9200H. And the total memory capacity is 16K Byte. Thus considering each memory location to hold 8 bits or 1 bytes in total 16 x 1024 bytes needs to be subtracted from the above memory address.

In decimal number system 16 x 1024 = 16,384 bytes. But since the given memory location is in Hexadecimal we need to convert this into a Hex. Therefore 16,384 is equal to 4000H in HEX.

9 2 0 0

- 4 0 0 0

5 2 0 0 H

The first memory address is 5200H.

1. Write short notes on the following : -
2. Addressing modes in 8085
3. Explain BI and EU of 8086
4. Mode of operation of 8253
5. Addressing modes of 8051

Ans.

1. Addressing modes in 8085 microprocessor : -

In order to perform any operation by the microprocessor we need to give the corresponding instruction to it. In each instruction the programmer has to specify 3 things :

1. Operation to be performed
2. Address of source of data
3. Address of destination of result

The method by which the address of source data or the address of destination of result is given in the instruction is called Addressing Modes.

The microprocessor 8085 has 5 addressing modes:

1. Immediate Addressing Mode
2. Direct Addressing Mode
3. Register Addressing Mode
4. Indirect Addressing Mode
5. Implied Addressing Mode

Immediate Addressing Mode: Data is present in the instruction. The CPU does not needs to fetch the data from anywhere, it is given explicitly in the instruction.

Eg: - MVI A, 30H (30H is copied into the register A)

MVI B,40H(40H is copied into the register B).

Direct Addressing Mode: Used to accept data from outside devices to store in the accumulator or send the data stored in the accumulator to the outside device.

Eg: - LDA 3000H (The content at the location 3000H is copied to the register A).

Register Addressing Mode: Data is provided through the registers. The instruction specifies in which register the required data is present and the CPU fetches data from that particular register.

Eg: - MOV B, A (the content of A is copied into the register B)

MOV A, C (the content of C is copied into the register A).

Indirect Addressing Mode: This means that the Effective Address is calculated by the processor. And the contents of the address (and the one following) is used to form a second address. The second address is where the data is stored.

Eg: - MOV A, M (data is transferred from the memory location pointed by the regiser to the accumulator)

Implied Addressing Mode: This mode doesn't require any operand. If address of source of data as well as address of destination of result is fixed, then there is no need to give any operand along with the instruction.

Eg: - CMA (Complements the content of the accumulator and saves it in the accumulator.)

1. Explain BI and EU of 8086 :

8086 microprocessor architecture divided in two parts first is execution unit and second is bus interface unit. Execution unit works all the calculation and manipulation work and bus interface unit work as data transfer from memory to microprocessor or ports and vice versa.

Execution Unit:

The execution unit has:

1. Control Unit
2. Instruction Decoder
3. ALU
4. General Registers
5. Flag Registers
6. Pointers
7. Index Registers

The EU is mainly responsible for executing the instructions of the program. The control unit co-ordinates all the other units of the processor. The ALU performs various arithmetic and logical operations over the data and the instruction decoder translates the instruction fetched from the memory into a series of actions that are carried out sequentially. And the general registers, flag registers and pointers acts as temporary locations to store the data on which actions are to be done and the results of those.

Bus Interface Unit:

The BIU has:

1. Instruction stream byte queue
2. A set of segment registers
3. Instruction pointer

This Bus Interface Unit handles all transfer of data and addresses on the buses for the EU(execution unit). This unit sends out addresses, fetches instructions from memory, reads data from ports and memory and writes data to ports and memory.

1. Explain the modes of 8253:

The 8253 chip has 6 mode of operation:

Mode 0 (Interrupt on terminal count):

The output is initially low, and remain low for the duration of the count if GATE=1. When the terminal count is reached, the output will go high and remain high until a new control word or new count number is loaded • Width of low pulse = N \* T, where T is clock period.  
If GATE becomes low at the middle of the count, the count will stop and the output will be low. The count resumes when the GATE becomes high again.

Mode 1 (HW triggered / programmable one shot):

The triggering must be done through the GATE input by sending a 0-to-1 pulse to it. In Mode 1, after sending the 0-to-1 pulse to GATE, OUT becomes low and stays low for a duration of N\*T, then becomes high and stays high until the GATE is triggered again – If during the activation, a retriggered happened, then restart the down counting.

Mode 2 (Rate Generator (Divide-by-N counter)):

In Mode2, if GATE=1, OUT will be high for N\*T, goes low only for one clock pulse, then counter is reloaded automatically, and the process continues indefinitely.

Mode 3 (Square Wave Rate Generator):

Initially output is high. For even count, counter is decremented by 2 on the falling edge of each clock pulse. When the counter reaches terminal count, the state of the output is changed and the counter is reloaded with the full count and the whole process is repeated. If the count is odd and the output is high the first clock pulse (after the count is loaded) decrements the count by 1. Subsequent clock pulses decrement the clock by 2. After timeout, the output goes low and the full count is reloaded. The first clock pulse (following the reload) decrements the count by 3 and subsequent clock pulse decrement the count by two. Then the whole process is repeated. In this way, if the count is odd, the output will be high for (n+1)/2 counts and low for (n-1)/2 counts

Mode 4 (Software triggered strobe):

The output will be initially high The output will go low for one CLK pulse after the terminal count (TC). If Gate is one the counting is enabled otherwise it is disabled. The Gate has no effect on the output. If a new count is written during counting, it will be loaded on the next CLK pulse and counting will continue from the new count. If the count is two byte then writing the first byte has no effect on counting and writing the second byte allows the new count to be loaded on the next CLK pulse.

Mode 6 (Hardware triggered strobe):

Similar to Mode4, except that the triggering must be done with the GATE input. The count starts only when a 0-to-1 pulse is sent to the GATE input. If GATE retriggered during the counting, it will restart the down counting