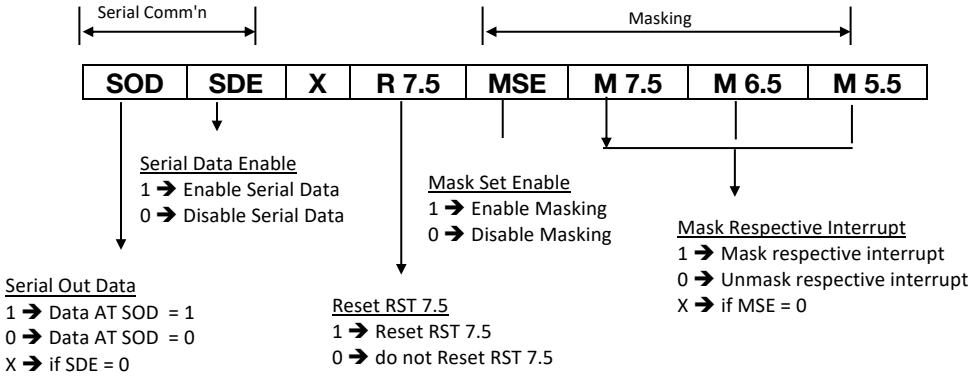
SIM & RIM INSTRUCTIONS

SIM: SET INTERRUPT MASK

SIM is a multipurpose instruction.
It is used for the following…

- To **Mask** or Un-Mask the **RST7.5, RST6.5 and RST5.5** interrupts.
- To send the data out serially (bit - by - bit) through the **SOD** line of the µP.
- To **reset RST7.5** interrupt irrespective of whether it is masked or not.



Method of execution:

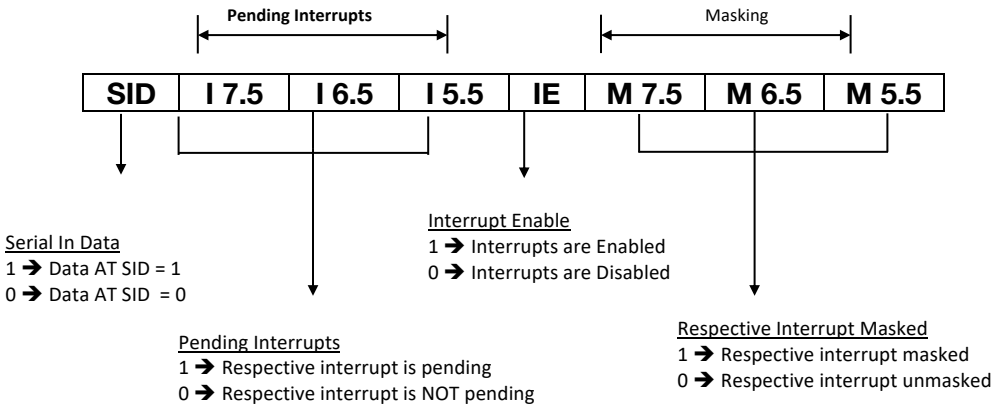
- The appropriate byte is formed and **loaded into** the **Accumulator**.
- Then the **SIM** Instruction is **executed**.
- The µP reads the contents of the accumulator in the above order.

> **Note**
>
> If We have disabled interrupts using DI, then whatever masking we do in SIM is of no use. First, we must enable interrupts using EI instruction. Only Then the masking pattern we give in SIM will come into effect.

RIM: READ INTERRUPT MASK

This instruction is used for the following purposes:

- To **read the Interrupt Mask** of the µP.
- To accept data serially through the **SID** pin.
- To see the **"Pending Interrupts"** of the µP.



Pending Interrupts:

Pending interrupts are those interrupts, which are waiting to be serviced.
An interrupt becomes pending as a higher priority interrupt is currently being serviced.
RIM Instruction indicates the Pending Status of RST7.5, RST6.5 and RST5.5.

Method of execution:

- Then the **RIM** Instruction is **executed**.
- The **µP loads** the appropriate byte into the **Accumulator.**
- The programmer reads the contents of the Accumulator.

Special Note:

If you are learning this by piracy, then you are not my student. You are simply a thief!
#PoorUpbringing

SQUARE WAVE PROGRAM USING SIM

Write a program to generate a SQUARE-WAVE of 1 KHz using SOD pin of 8085.

Soln:

```
BACK:  MVI   A, 40H  ; SIM Command = 0100  0000
       SIM
       CALL  DLAY
       MVI   A, C0H  ; SIM Command = 1100  0000
       SIM
       CALL  DLAY
       JMP   BACK
```

For a square wave of 1 KHz, the time period is 1 msec.
Hence the required delay is of 0.5 msec.
Assume 8085 is working at 3 MHZ

```
DLAY:  MVI   B, XXH      ; 7 T-states … … … Count is calculated later
BACK:  DCR   B           ; 4 T-states … … … Decrement Count
       JNZ   BACK        ; 10T (true) / 7T (false)
       RET               ; 10T-states
```

T_D = MT + $[(Count)_d$ x NT] – 3T
Here MT = Time outside the loop = 17T
NT = Time inside the loop = 14T
T_D = 17T + $[(Count)_d$ x 14T] – 3T

Required T_D = 0.5 msec = 0.5 x 10^{-3} sec
1T = 0.333 µsec = 0.333 x 10^{-6} sec

Substituting the above values we get:
$0.5x10^{-3} = 17x(0.333x10^{-6}) + [(Count)_d$ x $14x(0.333x10^{-6})] – 3x(0.333x10^{-6})$

Count = 6AH

Special Note:

If you are learning this by piracy, then you are not my student. You are simply a thief!
#PoorUpbringing

Bharat Acharya Education

Learn...
8085 | 8086 | 80386 | Pentium | 8051 | ARM7 | COA

Fees: 1199/- | Duration: 6 months | Activation: Immediate | Certification: Yes
Free: PDFs of Theory explanation, VIVA questions and answers, Multiple-Choice Questions

Start Learning... NOW!
www.BharatAcharyaEducation.com

Order our Books here...

8086 Microprocessor

Link: https://amzn.to/3qHDpJH

8051 Microcontroller

Link: https://amzn.to/3aFQkXc

Official WhatsApp number:

+91 9136428051