Serial Input/ Output

The 8085 has a SID (Serial input data) pin. You can use this input to receive serial data from a peripheral device.

The RIM instruction reads the interrupt mask into the accumulator (see the following figure).

SID Serial Input Data							

RIM ; A=

Bit 7 is the serial input data bit. This bit has nothing to do with the interrupt system; it included in the RIM instruction to avoid having to include an extra instruction for SID operations.

Each time a new bit arrives at the SID input, we can execute a RIM instruction. By isolating and saving this bit, we can convert a serial data stream into a parallel 8-bit word. If interrupt-driven I/O is used, a service subroutine is called each time a new bit is at the SID input. This subroutine would include a RIM plus rotate and store instructions for serial-to-parallel conversion.

PROGRAM FOR SERIAL INPUT DATA

```
        MVI B,0         ; CLEAR B REGISTER
        MVI C,08        ; PRESENT COUNTER TO 8
LOOP:   RIM             ; GET SID BIT
        ANI 80          ;  ISOLATE SID BIT
        ORA B           ; UPDATE PARALLEL WORD
        RRC             ; ROTATE RIGHT
        MOV B,A         ; SAVE A
        DCR C           ; COUNT DOWN
        JNZ LOOP        ; GO BACK IF NOT FINISHED
        RLC             ; ROTATE LEFT
        HLT
```

The SOD output pin can deliver a serial data stream to a peripheral device. The SIM instruction sets the interrupt mask. Bit 7, SOD, is latched into the SOD output pin only if bit6, SOE (SOD enable), is high. In other words, bit 6 acts like a switch for bit 7. (See the following figure)

SOD Serial Output Data	SOE Serial Output Enable						

As an example, if we want to send a high bit to the SOD output pin, we can use:

```
MVI A,0c0h
SIM
```

The MVI sets bits 7 and 6. The SIM then latches bit 7 into the SOD output pin. To send a low bit to the SOD output, we can use:

```
MVI A,40h
SIM
```

By using rotate and other instruction we can write a program that convert an 8-bit parallel word into a serial data stream at the SOD output. With interrupt-driven I/O, the service subroutine would include a SIM, rotates, and other instructions for parallel-to serial conversion.

PROGRAM FOR SERIAL OUTPUT DATA

```
      MVI A,4D      ; LOAD ASCII M
      MVI C,08      ; PRESENT COUNTER TO 8
LOOP: RRC           ; ROTATE LSB INTO MSB
      MOV B,A       ; SAVE A
      ANI 80        ; ISOLATE SOD BIT
      ORI 40        ; ENABLE SOE BIT
      SIM           ; LATCH SOD BIT
      MOV A,B       ; RESTORE A
      DCR C         ; COUNT DOWN
      JNZ LOOP      ; GO BACK IF NOT FINISHED
      HLT
```

Class Work

Receive 5 serial data byte and store them in memory addressed by 2080.

Address	HexCode	Label	Opcode	Operands	Comments
2000 2001 2002	 80 20		LXI	H,2080	; HL=2080
2003 2004	 05		MVI	D,05	; D=5
2005 2006 2007	 08 00	SBYTE:	LXI	B,008	; BC=008
2008		NEXT:	RIM		; RIM
2009 200A	 80		ANI	80	; A= A AND 80
200B			ORA	B	; A=A OR B
200C			RRC		; CY=BIT0
200D			MOV	B,A	; B=A
200E			DCR	C	; C=C-1
200F 2010 2011	 08 20		JNZ	NEXT	; IF Z=0 then PC=2008
2012			RLC		; BIT0=CY
2013			MOV	M,A	; M_{HL}=A
2014			INX	H	; HL=HL+1
2015			DCR	D	; D=D-1
2016 2017 2018	 05 20		JNZ	SBYTE	; IF Z=0 then PC=2005
2019			RST1		END