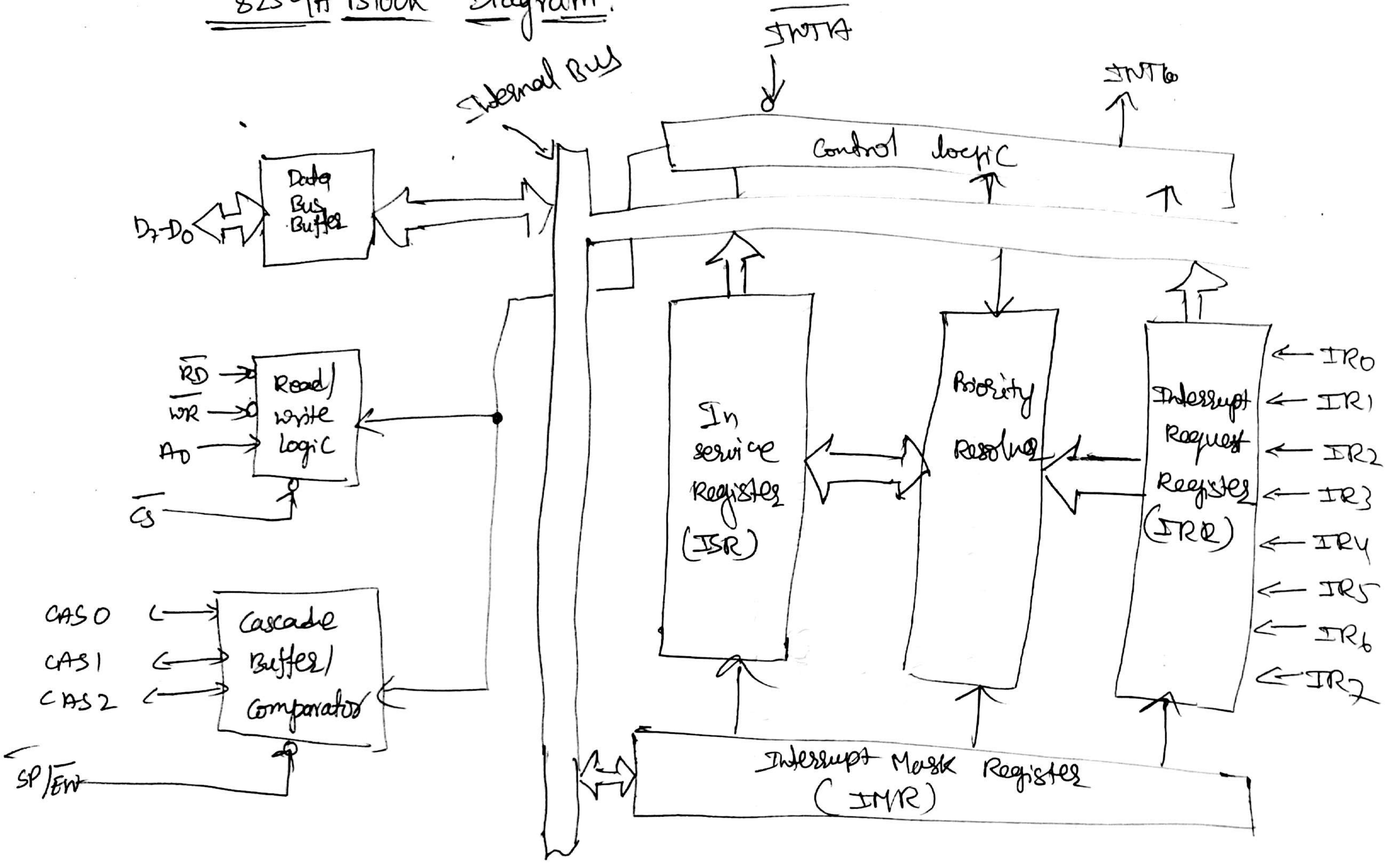
8259A Block Diagram:



→ 8259A consist of eight data bus lines from D_0 - D_7

→ The data bus is the path over which data are transferred b/n MPU & 8259A

→ The data can be command words, status information or interrupt type numbers.

→ The control signals WR & RD (active low) logic 0, signal the 8259A whether data to be written or read from its internal register

→ CS (active low) to enable the host interface enable

→ A_0 input involved in the selection of internal register is accessed during read & write operations

→ Interrupt request IR_0 - IR_7 inputs are issued by external devices for servie

→ CAS0 - CAS2 are known as cascade interface.

→ The cascade lines are used to cascaded system where ~~an~~ number of 8259 ICs are interconnected in master/slave configuration to expand the number of IR inputs from 8 to as high as 64

→ SP/EN (active low) is used to indicate the current device acts as a master or slave.

→ INT is the interrupt request output of 8259A

→ It is applied directly to the INTR input of the 8086.

→ logic 1 is produced at this output whenever the interrupt controller receives a valid request from a ~~interr~~ interrupting device.

→ INTA (active low) is an input pin of the 8259A

→ This pin is used to receive acknowledgement

Interrupt Request Register (IRR)

-> Interrupt request register keeps track of which interrupt inputs are asking for service

-> If an interrupt input has an interrupt signal on it, then corresponding bit in the interrupt request register will be set

Interrupt Mask Register

-> Interrupt mask register is used to disable or enable the individual interrupt inputs.

-> Each bit in this register corresponds to the interrupt input with the same number

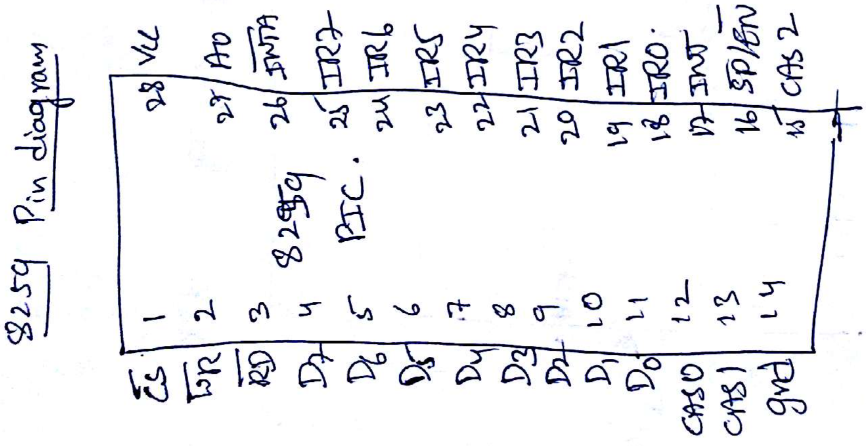
Interrupt service Register

-> The In-service register keep tracks of which interrupt inputs are currently being serviced

-> The corresponding bit will be set in the in service register

Priority Resolver:

-> The priority resolver acts as a judge" - that determine if and when an interrupt request on one of the IR inputs got serviced.

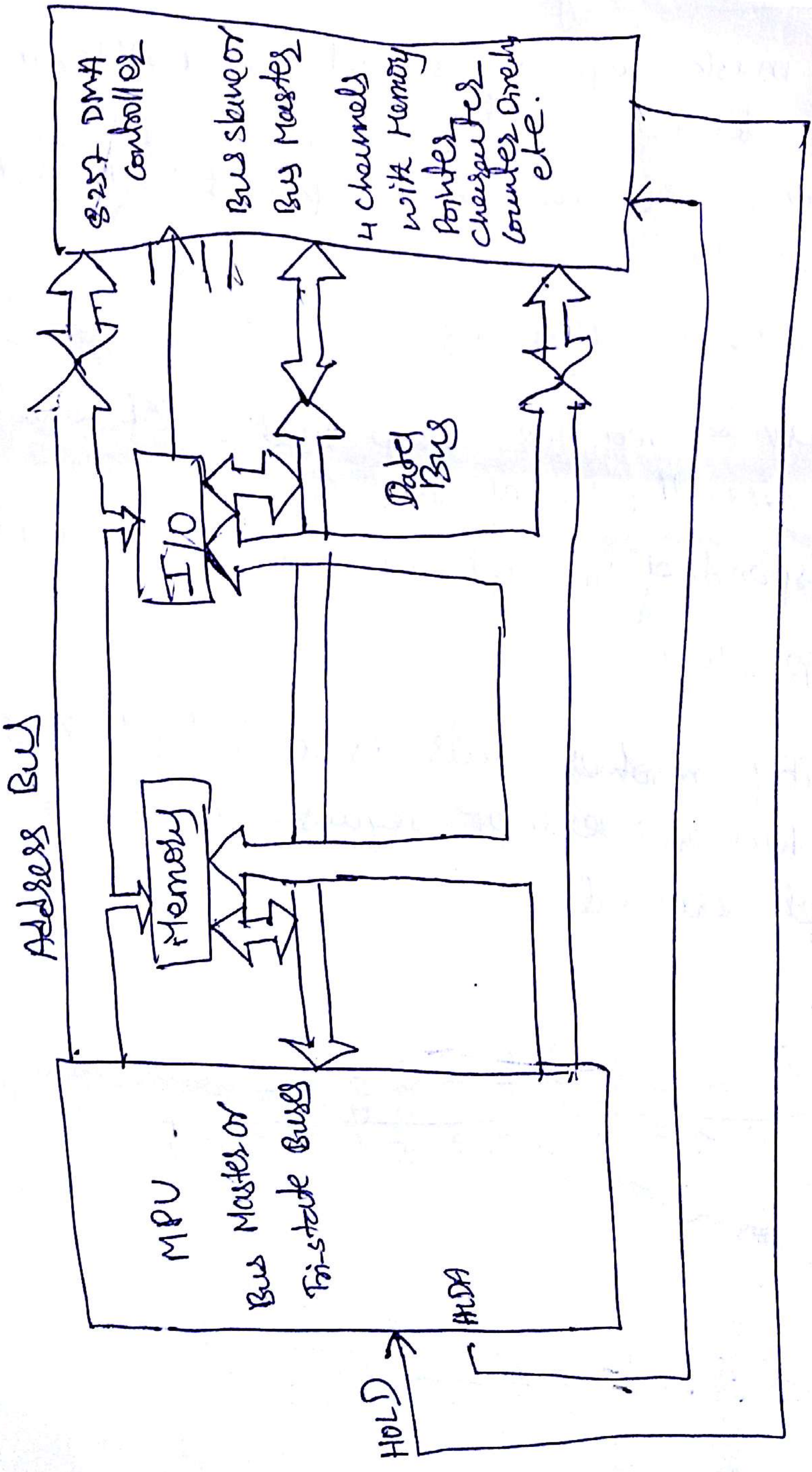


Dired Memory Access (DMA)

→ DMA is a process of communication or data transfer controlled by a external peripheral.

→ In situations in which the MP-controlled data transfer is too slow, the DMA is generally used.

e.g. data tranfer b/n a floppy disk & R/W memory of the system.


8257 DMA Controller
Bus slave or Bus Master
4 channels with Memory Pointer Register Counter Areh etc.
Data Bus
I/O
Address Bus
Memory
MPU
Bus Master or Tri-state Buss
HLDA
HOLD

DMA Data Transfer

-> The 8085 μp has two pins available for this type of I/O communication: HOLD (Hold) and HLDA (Hold Ack).

-> Conceptually this is an important technique; it introduces two new signals available on the 8082 — HOLD & HLDA.

-> DMA stands for Direct Memory Access.

-> It is designed by Intel to transfer data at the fastest rate.

-> It allows the device to transfer the data directly to/from memory without any interference of the CPU

How DMA Operations are performed?

-> Initially, when any device has to send data b/n the device & the memory, the device has to send DMA request (DRQ) to DMA controller.

-> The DMA controller sends Hold request (~~HRQ~~ HOLD) to the CPU and waits for the CPU to assert the HLDA

-> Then the μp tri-states all the data bus, address bus and control bus. The CPU leaves the control over bus and acknowledges the HOLD request through HLDA signal.

-> Now the CPU is in HOLD state and the DMA controller has to manage the operations over buses b/n the CPU memory & I/O devices

Features of 8257:

-> It has four channels which can be used over four I/O devices.

-> Each channel has 16-bit address & 14-bit counter

-> Each channel can transfer data up to 64kb

-> Each channel can be programmed independently

-> Each channel can perform read transfer, write transfer, & verify transfer operations.

-> It generates MARK signal to the peripheral device that 128 bytes have seen transferred.

-> It requires a single phase clock.

-> Its frequency ranges from 250 Hz to 3 MHz.

-> It operates in 2 modes, i.e.

Master mode
Slave "

-> DAA instruction

1-byte
4 -T states.

-> The contents of the accumulator are changed from a binary value to two 4-bit binary-coded decimal digits. This is the only instruction that uses the auxiliary flag (internally) to perform the binary-to-BCD conversion,

1. If the value of the low-order four bits (D_3-D_0) in the accumulator is greater than 9 or if AC flag is set, the instruction adds 06 to the low-order four bits.

2. If the value of the high-order four bits (D_7-D_4) in the accumulator is greater than 9 or if the carry flag is set, the instruction adds 6 (60) to the high-order four bits.

Eg: Add decimal 12_{BCD} to the accumulator, which contains 39_{BCD}.

39_{BCD} = 0011 1001

12_{BCD} = 0001 0010

0100 1011

4 B

The binary sum is 4BH. The value of the low-order four bits is larger than 9. Add 06 to the low-order four bits.

4B = 0100 1011

06 = 0000 0110

0101 0001

5 1

Eg: Add decimal 68_{BCD} to the accumulator, which contains 85_{BCD}

1000

85 = ~~0111~~ 0101

68 = 0110 1000

1110 1101

E D

= 1110 1101

0110 0110

$\boxed{1}$ 0101 0011

cy

45 = 0100 0101

38 = 0011 1000

0111 1101

0 0 0 0 0

7 D

1 1 1 0110

1000 0011

8 3

3/2/17 Absentees 1st period.

3, 6 10, 23, 22, 43, 52, 54, 58,