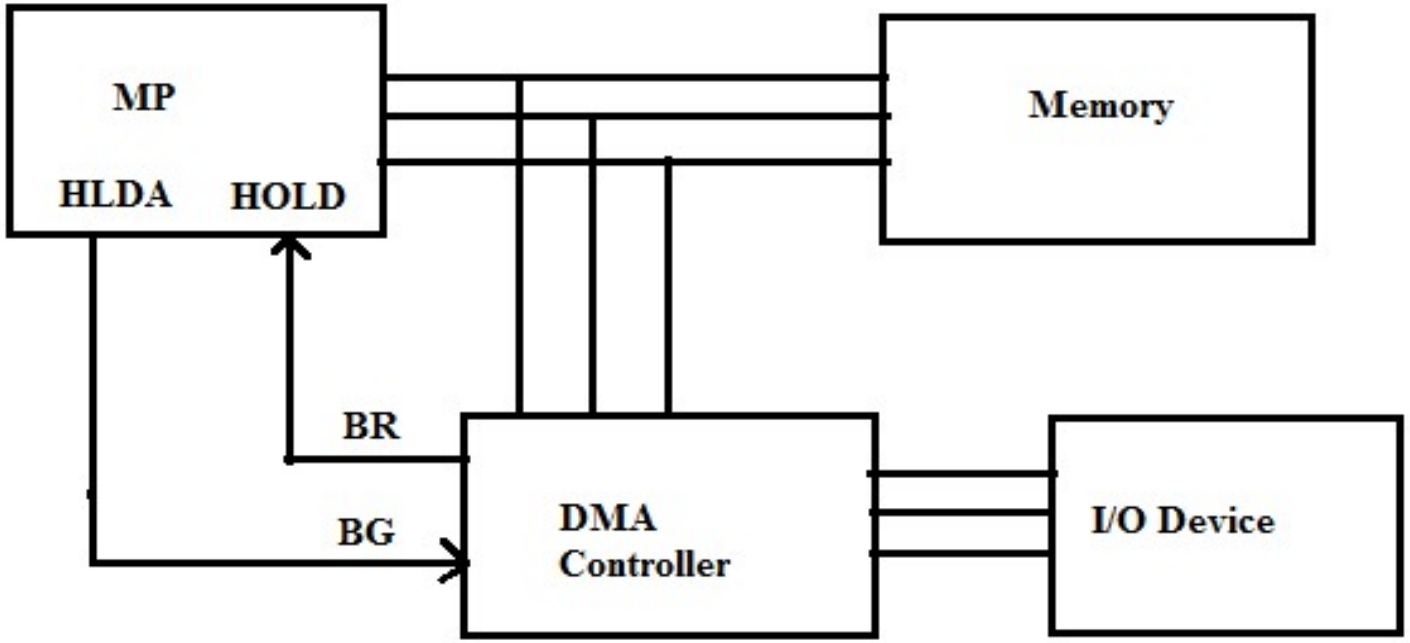
Direct Memory Access (DMA)/Interrupts

Introduction, Advantage and Application
8237 DMA Controller and Interfacing
Interrupt and its types

Direct Memory Access (DMA)



DMA - Data transfer between memory and I/O, controlled by an external circuit called DMA controller, without involvement of CPU.

- In 8085 MP two pins ***HOLD*** and ***HLDA*** are available for ***DMA*** operation.
- ***Step1:***

  DMA controller sends a request by making ***Bus Request (BR)*** control line high. When MP receives high signal to ***HOLD*** pin, it first completes the execution of current machine cycle, it takes few clock periods.

- ***Step 2:***

  Mp acknowledges ***DMA controller*** through ***HLDA*** pin. After receiving acknowledge through ***Bus Grant (BG)*** pin, the DMA controller takes control over system bus and transfers data directly between memory and I/O ***without involvement of CPU***.

- ***Step 3:***

  At the end of data transfer, the DMA controller terminates the request by sending low signal to ***HOLD*** pin and MP regains control over the system bus by making ***HLDA*** low.

- During DMA operation, the processor is free to perform other task which does not need system bus.

DMA Data Transfer scheme

- **Block (Burst) mode of data transfer**
  - In this scheme, the I/O device withdraws the DMA request only after all the data bytes have been transferred.
- **Cycle stealing technique**
  - In this scheme the bytes are divided into several parts and after transferring every part the control of buses is given back to MPU and later stolen back when MPU does not need it.

8237 DMA Controller:

- Designed by ***Intel*** to transfer data at the ***fastest rate***.
- Allows the device to transfer the data directly to/from memory without any interference of the ***CPU***.
- The device requests the ***CPU*** to hold its data, address and control bus, so the device is free to transfer data directly to/from the memory.
- The ***DMA*** data transfer is initiated only after receiving ***HLDA*** signal from the ***CPU***.

DMA controller 8237 Interfacing

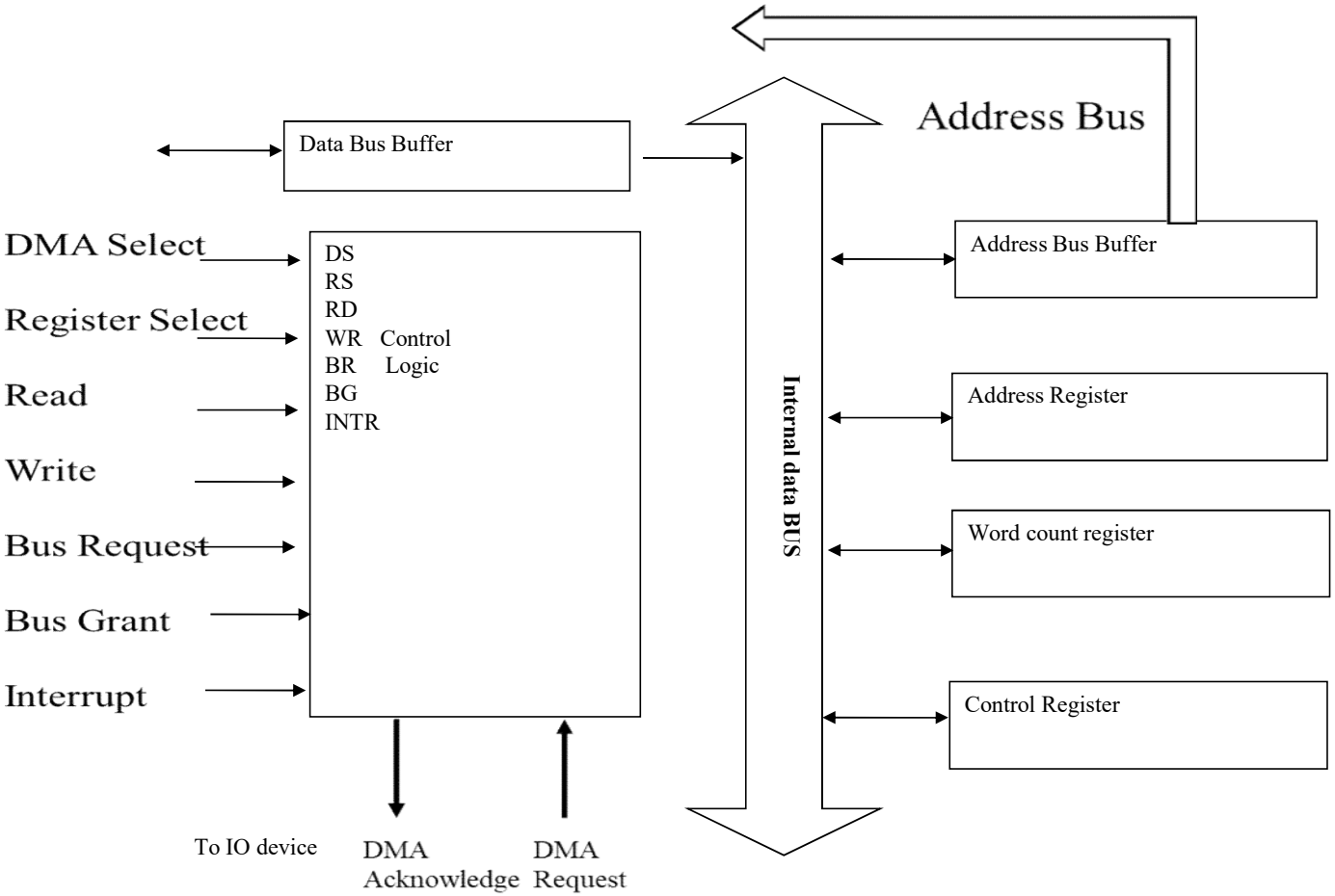
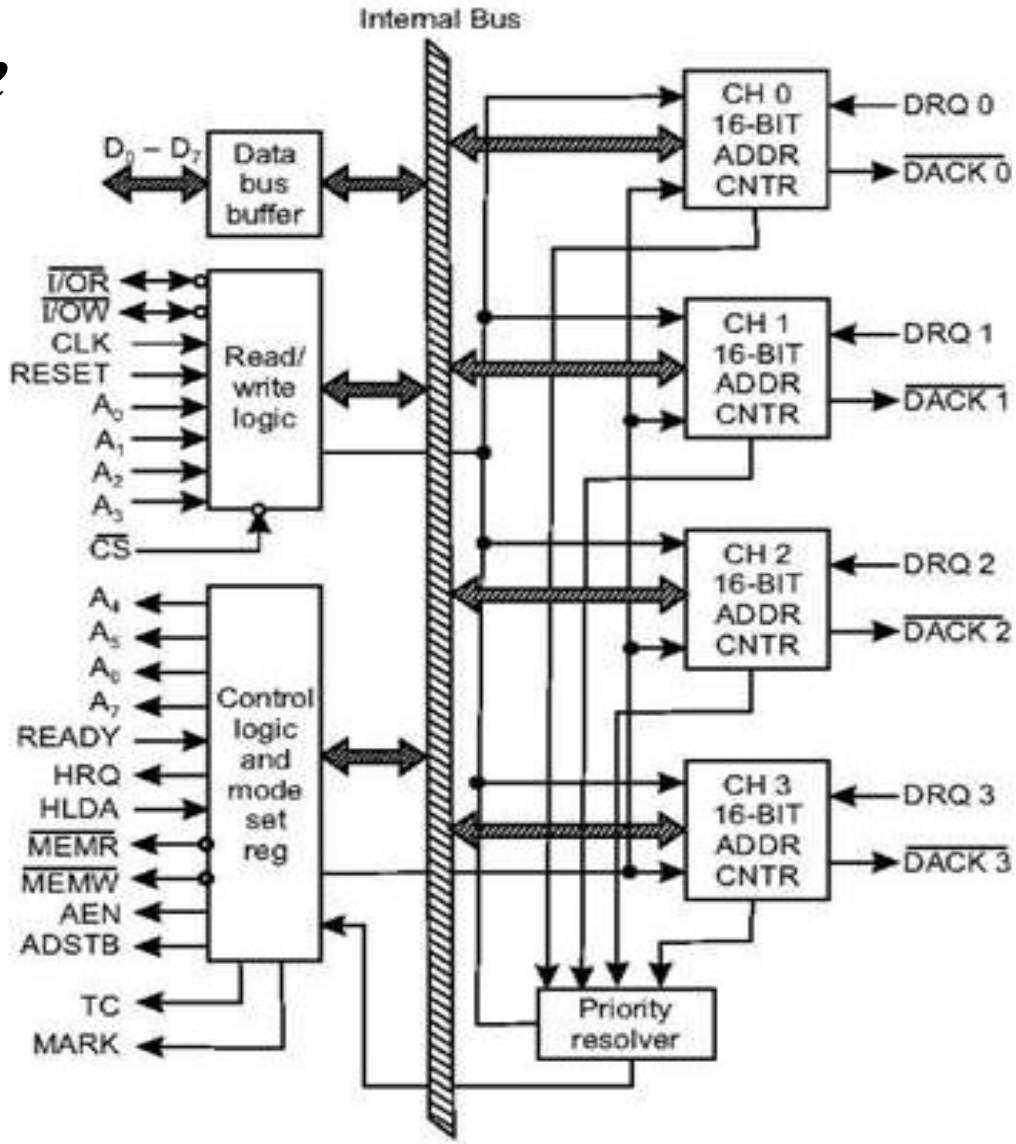


Fig: A conceptual overview of DMA Controller

- The unit communicates with the MP via the data bus and control lines.
- The registers in the DMA are selected by the MP through the address bus by enabling the DS (DMA select) and RS (Register Select) inputs.
- The RD (read) and WR (write) inputs are bidirectional.
- When the bus grant (BG) = 0, the MP can communicate with the DMA registers through the data bus to read from or write to the DMA registers.
- When BG=1, the processor does not have control over the system buses and the DMA can communicate directly with the memory by specifying an address in the address bus and activating the RD or WR control.

- The DMA controller has three registers: ***an address register***, ***a word count register*** and ***a control register***.
- The ***address register*** contains an address to specify the *desired location in memory*. The address bits go though bus buffers into the address bus. The address register is incremented after each word that is transferred to memory.
- The ***word count register*** holds the *number of words* to be transferred. The register is decremented by one after each word transfer and internally tested for zero.
- The ***control register*** specifies the *mode of transfer*.

8257 Architecture



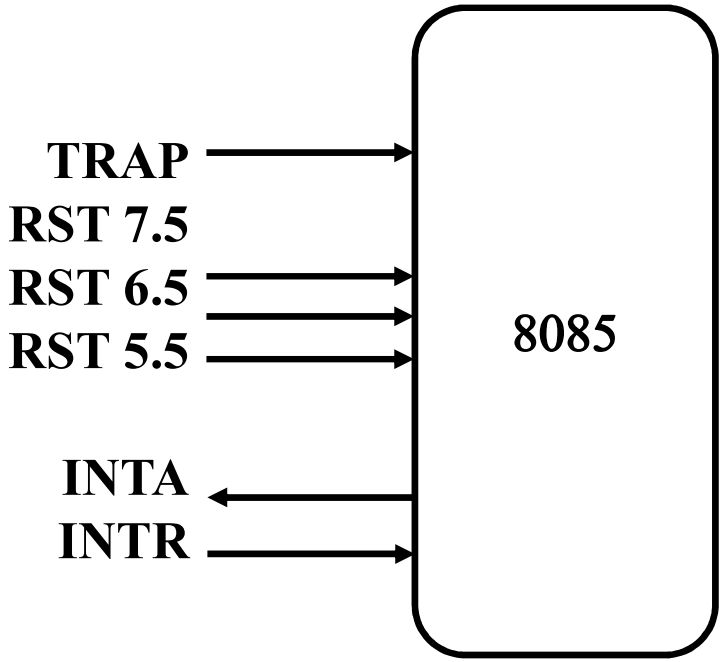
Features of 8257

- It has four channels which can be used over four I/O devices.
- Each channel has 16-bit address and 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 and verify transfer operations.
- It generates MARK signal to the peripheral device that 128 bytes have been transferred.
- It requires a single phase clock.
- Its frequency ranges from 250Hz to 3MHz.
- It operates in 2 modes, i.e., **Master mode** and **Slave mode**.

INTERRUPT

- Process where an external device can get the attention of the microprocessor..
- The process starts from the i/o device.
- An interrupt is considered to be an emergency signal that may be serviced.
- The microprocessor may respond to it as soon as possible.
- When the Microprocessor receives an interrupt signal, it suspends the currently executing program and jumps to an Interrupt Service Routine (ISR) to respond to the incoming interrupt.
- Each interrupt will most probably have its own ISR.

8085 Interrupt Pins and Priority:



Five interrupt pins in 8085 and one interrupt acknowledge (INTA) pin.

Pin No.	Name	Type	Priority	ISR Location
6	*TRAP*	*Vectored*	*Highest*	*CALL 0024H (3-byte call)*
7	*RST 7.5*	*Vectored*		*CALL 003CH (3-byte call)*
8	*RST 6.5*	*Vectored*		*CALL 0034H (3-byte call)*
9	*RST 5.5*	*Vectored*		*CALL 002CH (3-byte call)*
10	*INTR*	*Non-Vectored*	*Lowest*	*RST (Restart instructions) – 1 byte call*

- **Vector Interrupt:**
  - In this type of interrupt, Processor knows the address of Interrupt. In other word processor knows the address of interrupt service routine.
  - The examples of vector interrupt are RST 7.5, RST 6.5, RST 5.5, TRAP.
- **Non-Vector Interrupt:**
  - In this type of interrupt, Processor do not know the address of Interrupt.
  - It should be given externally.
  - The device should send the address of interrupt service routine to the processor for performing Interrupt.
  - The example of Non-vector interrupt is INTR.
- When a device interrupts, it actually wants the MP to give a service which is equivalent to asking the MP to call a subroutine. This subroutine is called ISR (Interrupt Service Routine).

Maskable and Non-Maskable Interrupt:

- **Maskable interrupts:**
  - Can be enabled or disabled by software; means we can enable or disable the interrupt by sending appropriate instruction like EI OR DI respectively.
  - INTR, RST 7.5, RST 6.5, and RST 5.5 are the examples of Maskable Interrupt OF 8085.
  - Level or edge triggered.
- **Non-Maskable interrupts:**
  - Cannot be enabled or disabled by sending any instruction.
  - TRAP interrupt is the non-maskable interrupt for 8085. It means that if an interrupt comes via TRAP, 8085 will have to recognize the interrupt we cannot mask it.
  - Used in critical power failure condition.
  - Both level and edged triggered.

Software and hardware interrupt:

- **Software Interrupt:**
  - Instruction based Interrupt which is completely controlled by software.
  - Programmer can use this instruction to execute interrupt in main program.
  - There are eight software interrupts available in 8085 microprocessor. See the example with their hex code and vector address.

Instruction	Corresponding HEX code	Vector addresses
RST 0	C7	0000H
RST 1	CF	0008H
RST 2	D7	0010H
RST 3	DF	0018H
RST 4	E7	0020H
RST 5	EF	0028H
RST 6	F7	0030H
RST 7	FF	0038H

- **Hardware Interrupt:**
  - Interrupt can be requested in hardware pin of microprocessor 8085.
  - There are mainly six dedicated pins available for interrupt purpose:
    - ✓ TRAP,
    - ✓ RST 7.5, RST 6.5, RSTV5.5,
    - ✓ INTR, INTA (it is not an interrupt pin but it is used to send acknowledgement of the interrupt request getting from other interrupt pin.)

Vectored and Polled Interrupt

- **Vectored Interrupt:**

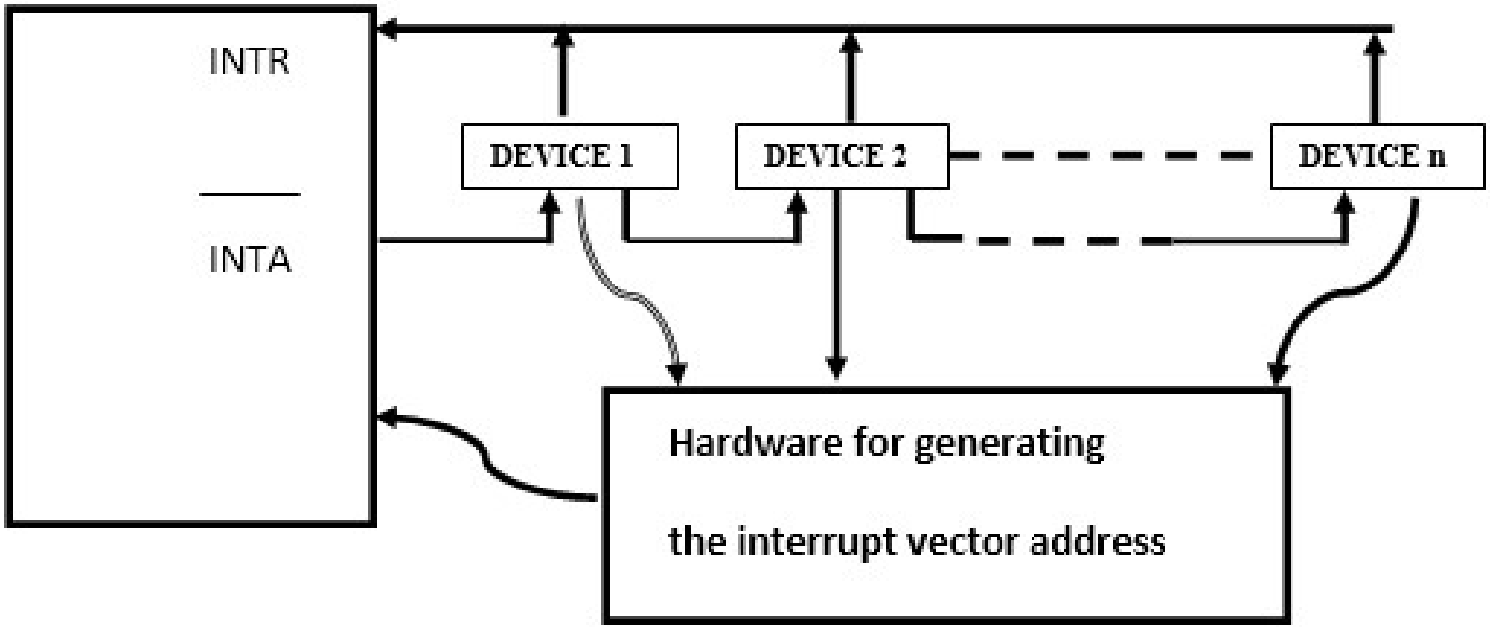


Fig: Vectored Interrupt

Polled Interrupt:

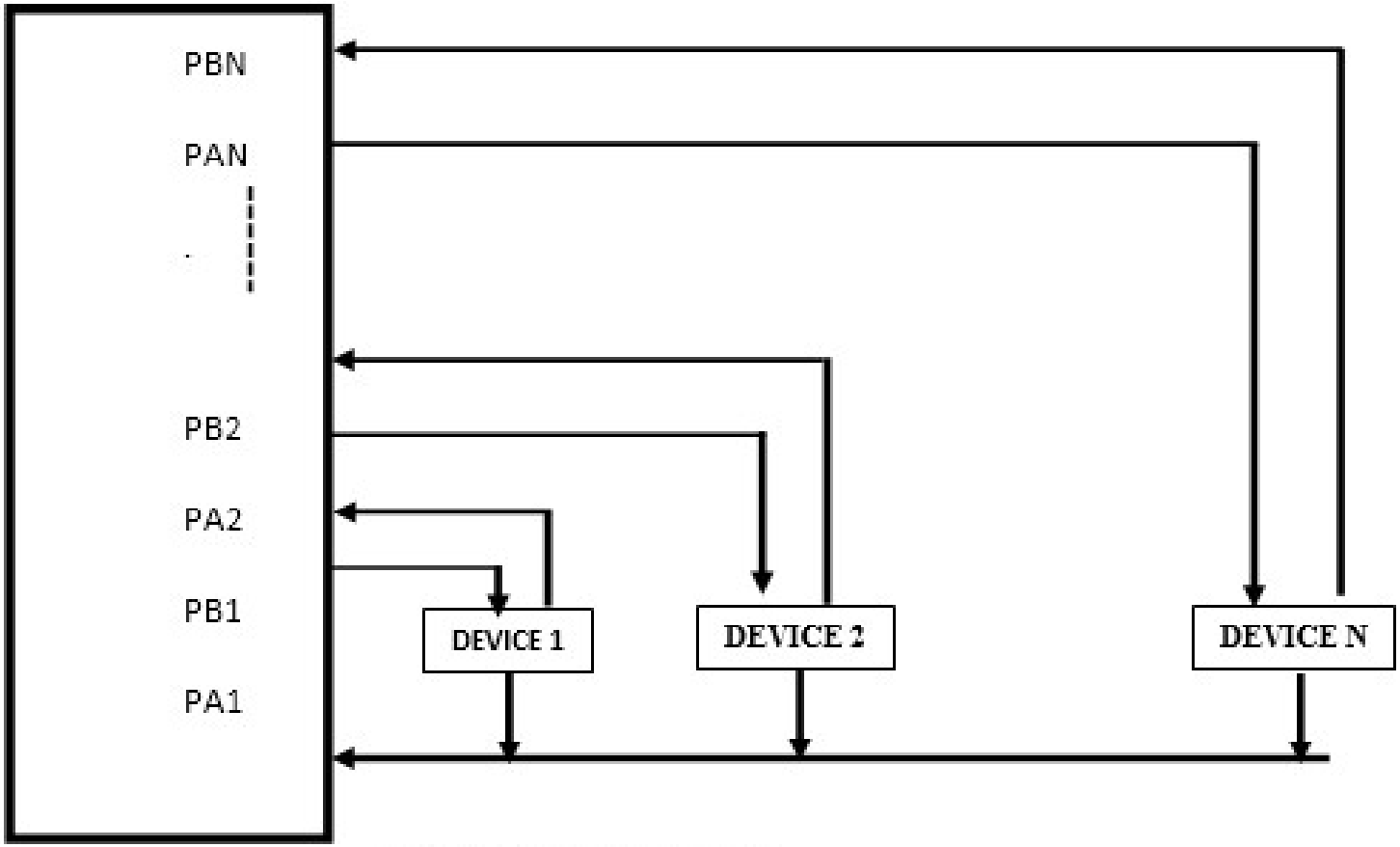


Fig: Polled Interrupt

8259 Interrupt Controller

- Programmable interrupt-managing device, specifically designed for use with the interrupt signals (INTR/INT) of the 8085 MP.
- Block diagram includes:
  - Control logic,
  - Registers for interrupt requests,
  - Priority resolver,
  - Cascade logic, and data bus.
  - The registers manage interrupt requests; the priority resolver determines their priority; cascade logic is used to connect additional 8259A devices.

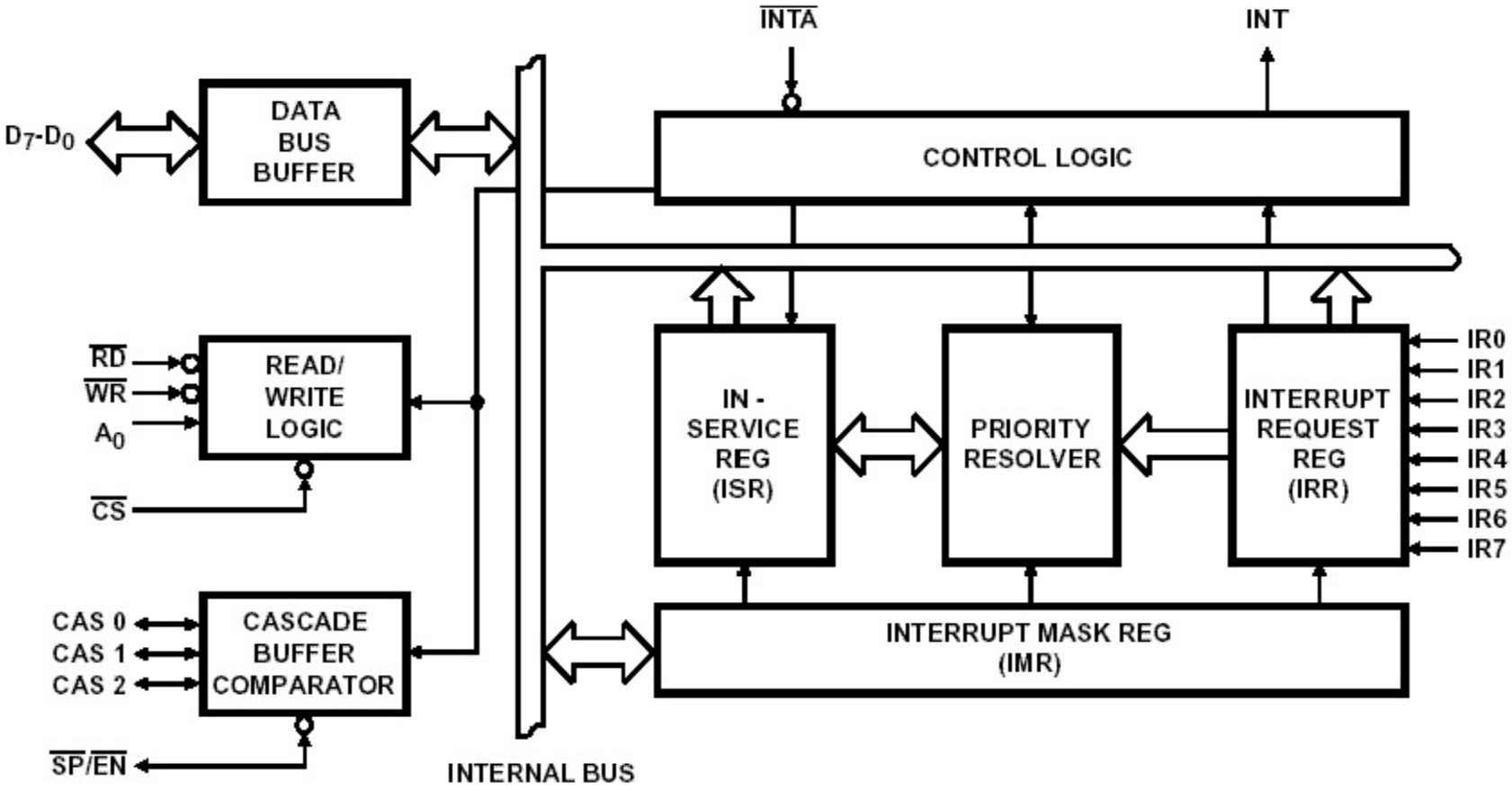


Fig: Block Diagram of 8259A PIC

The operation of 8259A:

i. One or more interrupt request lines go high requesting the service.

ii. The 8259A resolves the priorities and sends an INT signal to the MP.

iii. The MP acknowledges the interrupt by sending INTA(bar).

iv. After the INTA(bar) has been received, the op-code for the call instruction (CDH) is placed on the data bus.

v. Because of the CALL instruction, the MP sends two more INTA(bar) signals.

vi. At the first INTA(bar), the 8259A places the low-order 8-bit address on the data bus and at the second INTA(bar), it places the high-order 8-bit address of the interrupt vector. This completes the 3- byte CALL instruction.

vii. The program sequence of the MP is transferred to the memory location specified by the CALL instruction.

Priority modes

1. **Fully Nested mode**
   - IR0 has the highest priority and the following IR1, IR2, IR3…… etc. have the decreasing priorities.
2. **Automatic rotation mode**
   - First priority changes to the last after its service.
3. **Specific rotation mode**
   - This is user selectable or programmable, which means priority can be selected by programming.

Features

i. It manages 8 interrupt requests.

ii. It can vector an interrupt request anywhere in the memory map through program control without additional hardware for restart instructions. However, all 8 requests are spaced at the interval of either 4 locations or 8 locations.

iii. It can solve 8 levels of interrupt priorities in a variety of modes.

iv. With cascading additional 8259A devices, the priority scheme can be expanded to 64 levels.

v. The 8259A has the abilities such as reading the status and changing the interrupt mode during a program execution.

vi. It can mask each interrupt request individually.

vii. It can be set up to work with either the 8085 MP mode or the 8086/8088 MP mode.