Microprocessor & Interfaces

(4CS3-04)

UNIT 4

8085 Microprocessor Interfacing

Presented by: Prof. Shruti Sharma

Assistant Professor

Department of Computer Science Engineering

AIET, Jaipur

TOPIC:

8255

PROGRAMMABLE PERIPHERAL INTERFACE

Basic Interfacing Concept

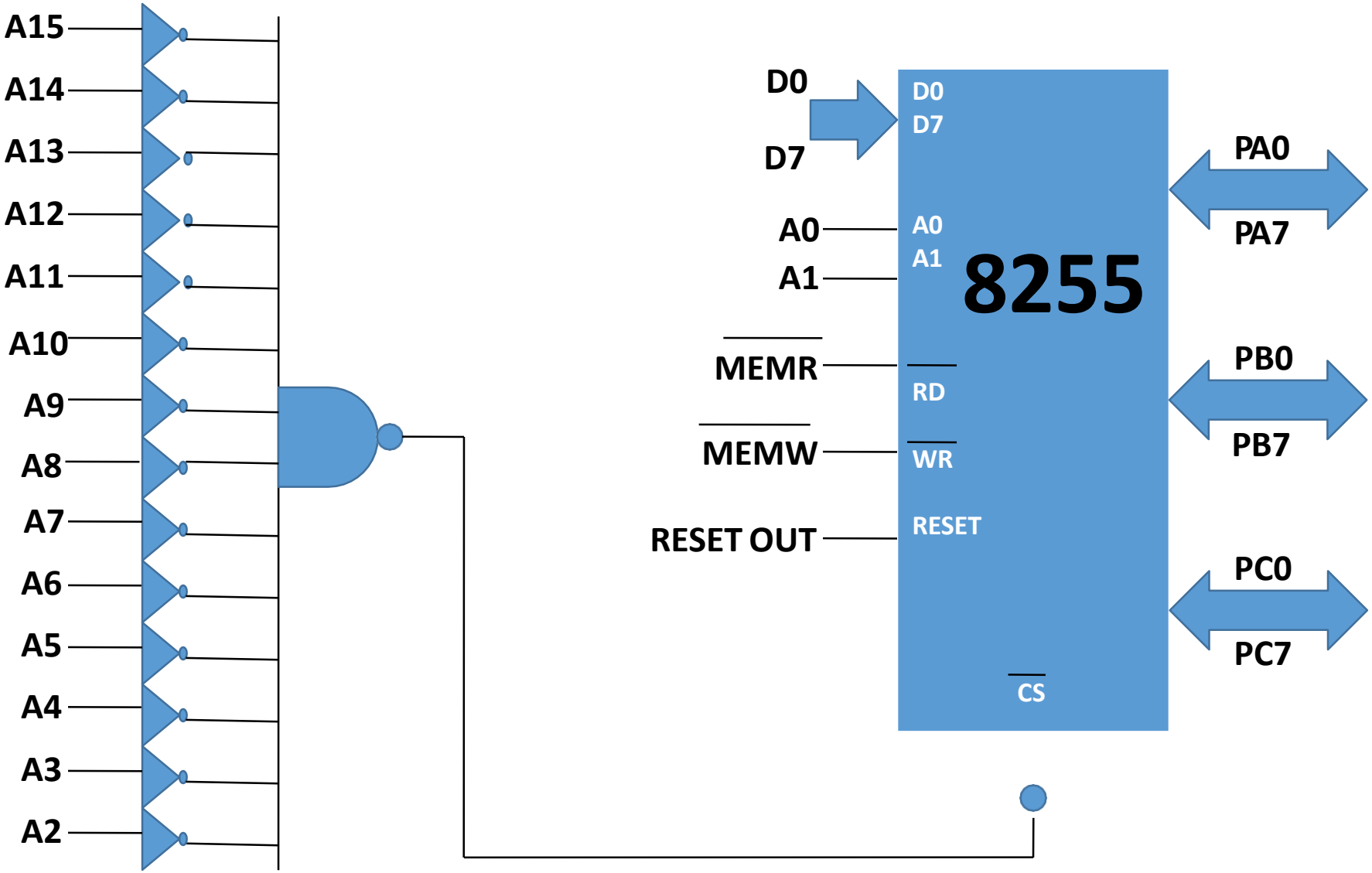
- **Any application of Microprocessor Based system Requires the transfer of data between external circuitry to the Microprocessor and Microprocessor to the External circuitry. User can give information (i.e. input) to the Microprocessor using keyboard and user can see the result or output information from the Microprocessor with the help of display.**

- **Hence interfacing is used to exchange information between two different applications/devices.**

Memory Mapped I/O

- **Device address is of 16 Bit. means A_0 to A_{15} lines are used to generate device address.**
- **MEMR and MEMW control signals are used to control read and write I/O operations.**
- **Data transfer is between Any register and I/O device.**
- **Maximum number of I/O devices are 65536.**
- **Decoding 16 bit address may requires more hardware.**
- **For e.g. MOV R M, ADD M,CMP M etc.**

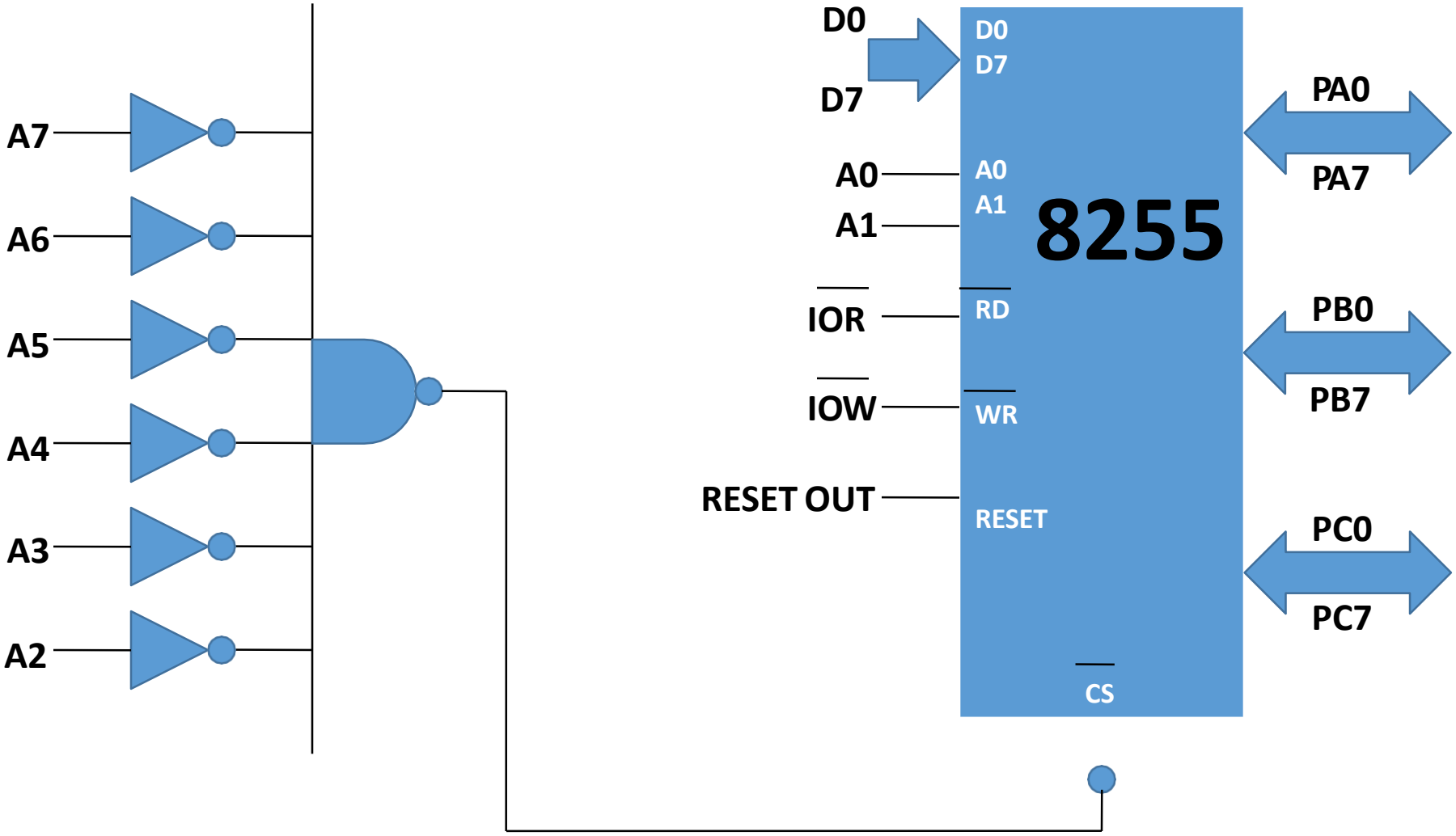
INTERFACING IN MEMORY MAPPED I/O



I/O Mapped I/O

- **Device address is of 8 Bit. means A_0 to A_7 or A_8 to A_{15} lines are used to generate device address.**
- **IOR and IOW control signals are used to control read and write I/O operations.**
- **Data transfer is between Accumulator and I/O device.**
- **Maximum number of I/O devices are 256.**
- **Decoding 16 bit address may requires less hardware.**
- **For e.g. IN, OUT etc.**

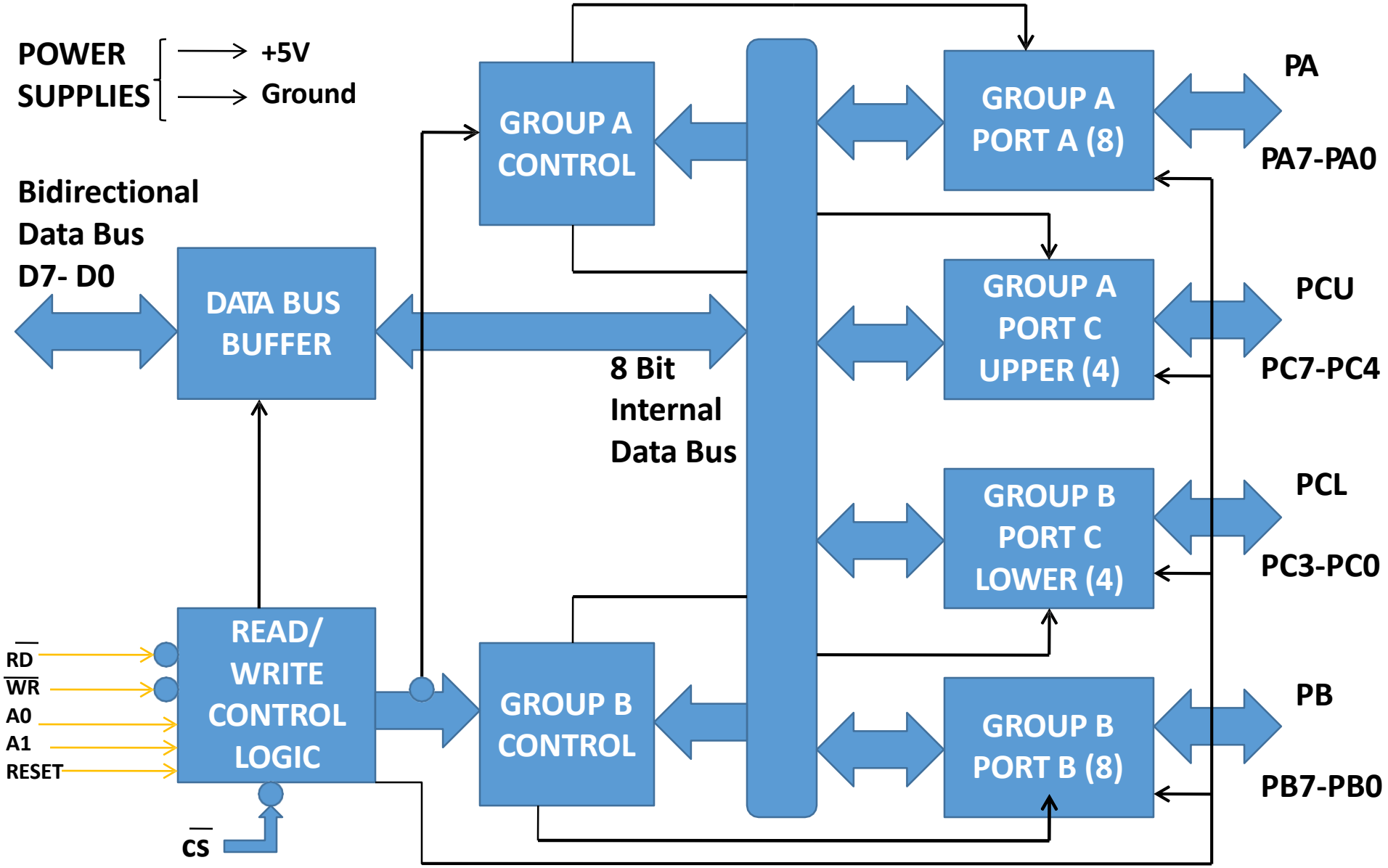
INTERFACING IN I/O MAPPED I/O



8255 PPI

- **The INTEL 8255 is a 40 pin IC having total 24 I/O pins. consisting of 3 numbers of 8 –bit parallel I/O ports (i.e. PORT A, PORT B,PORT C). The ports can be programmed to function either as a input port or as a output port in different operating modes. It requires 4 internal addresses and has one logic LOW chip select pin. Its main functions are to interface peripheral devices to the microprocessor. Basically used for parallel data transfer. operates in mainly two modes.**
- **(1) Bit Set Reset Mode (BSR Mode).**
- **(2) I/O Mode.**

Block Diagram of 8255 PPI

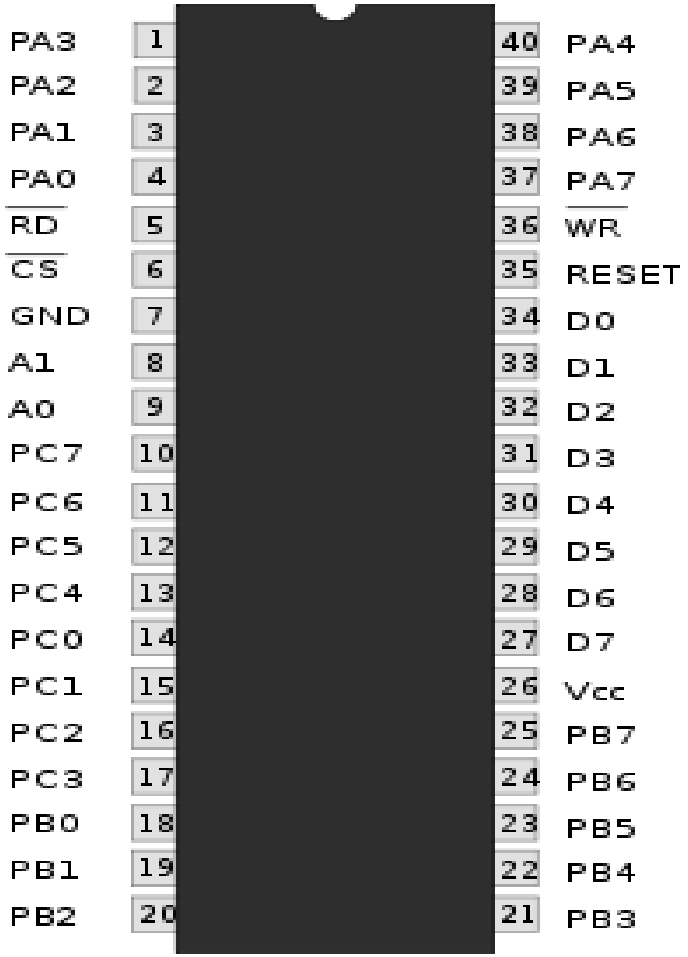


Function of Blocks

BLOCK	FUNCTION OF BLOCK
Data Bus Buffer	It is used to interface the internal data bus of 8255 to the system data bus by reading and writing operations.
Read/write Control logic	It accepts the input from the address bus and issues commands to the individual group blocks. also issues appropriate enabling signals to access the required data/control words/status words.
Port A	It can be programmed in three modes Mode0, Mode1 and Mode2.
Port B	It can be programmed in three modes Mode0 and Mode1.
Port C	It can be programmed for Bit Set/reset operation.

Pin Diagram of 8255 PPI

8255 Pin Diagram



Function of Pins

PIN	FUNCTION OF PIN
D0-D7 (Data Bus)	These are bidirectional, tri-state data bus lines are connected to the system data bus. They are used to transfer data and control word from microprocessor (8085) to 8255 or receive data or status word from 8255 to the 8085.
PA0-PA7 (Port A)	These are 8 Bit bidirectional I/O pins used to send data to output device and to receive data from input device. It functions as an 8 Bit data output latch/buffer when used in output mode and as an 8 Bit data input latch/buffer when used in input mode.
PB0-PB7 (Port B)	These are 8 Bit bidirectional I/O pins used to send data to output device and to receive data from input device. It functions as an 8 Bit data output latch/buffer when used in output mode and as an 8 Bit data input latch/buffer when used in input mode.

Function of Pins

PIN	FUNCTION OF PIN
PC0-PC7 (Port C)	These are 8 bit bidirectional I/O pins divided into two groups PCL (PC3-PC) and PCU (PC7-PC4).these groups can individually transfer data in or out when programmed for simple I/O, and used as handshake signals when programmed for handshake or bidirectional modes.
RD	When this pin is low, the CPU can read data in the ports or the status word through the data bus buffer.
WR	When this pin is low, the CPU can write data on the ports or in the control register through the data bus buffer.
CS	This pin can be enabled for data transfer operation between the CPU and 8255.
RESET	This pin is used to reset 8255.i.e control register gets cleared and all the ports are set to the input mode.

Function of Pins

PIN	FUNCTION OF PIN
A0-A1	The selection of input port and control word register is done by using A0 and A1 pins In conjunction with RD and WR pins.

$\overline{A1}$	$\overline{A0}$	$\overline{RD}$	$\overline{WR}$	CS	Operations
0	0	0	1	0	PORT A TO DATA BUS
0	1	0	1	0	PORT B TO DATA BUS
1	0	0	1	0	PORT C TO DATA BUS
0	0	1	0	0	DATA BUS TO PORT A
0	1	1	0	0	DATA BUS TO PORT B
1	0	1	0	0	DATA BUS TO PORT C
1	1	1	0	0	DATA BUS TO CONTROL REGISTER
x	x	x	x	1	DATA BUS TRI STATED
1	1	0	1	0	ILLEGAL CONDITION
x	x	1	1	0	DATA BUS TRI STATED

Operating Modes Of 8255

- **There are two main operational modes of 8255:**

(1)Input/output mode,

(2)Bit set/reset mode (BSR Mode).

I/O mode again classified into three types

- **(1) Mode 0,**
- **(2) Mode 1,**
- **(3) Mode 2.**

MODE 0

- **In this mode, the ports can be used for simple input/output operations without handshaking.**
- **If both port A and B are initialized in mode 0, the two halves of port C can be either used together as an additional 8-bit port, or they can be used as individual 4-bit ports.**
- **Since the two halves of port C are independent, they may be used such that one-half is initialized as an input port while the other half is initialized as an output port.**

The mode 0 has following features:

- **O/p are latched.**
- **I/p are buffered not latched.**
- **Port do not have handshake or interrupt capability.**

MODE 1

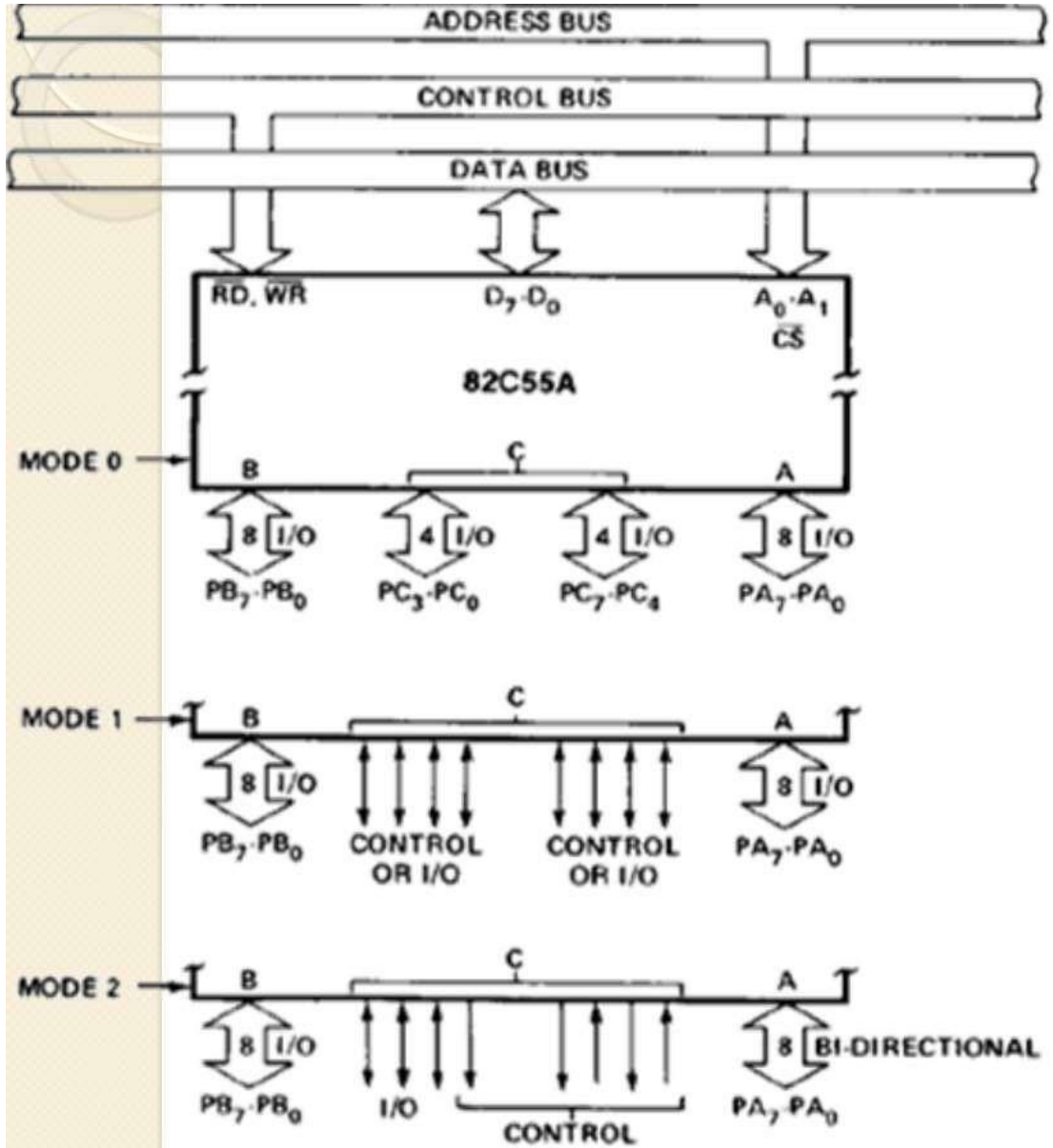
- **When we wish to use port A or port B for handshake (strobed) input or output operation, we initialize that port in mode 1.**
- **For port B in this mode (irrespective of whether is acting as an input port or output port), PC0, PC1 and PC2 pins function as handshake lines.**

The mode 1 has following features:

- **Two ports i.e. port A and B can be use as 8-bit i/o port.**
- **Each port uses three lines of port c as handshake signal and remaining two signals can be function as i/o port.**
- **Interrupt logic is supported.**
- **Input and Output data are latched.**

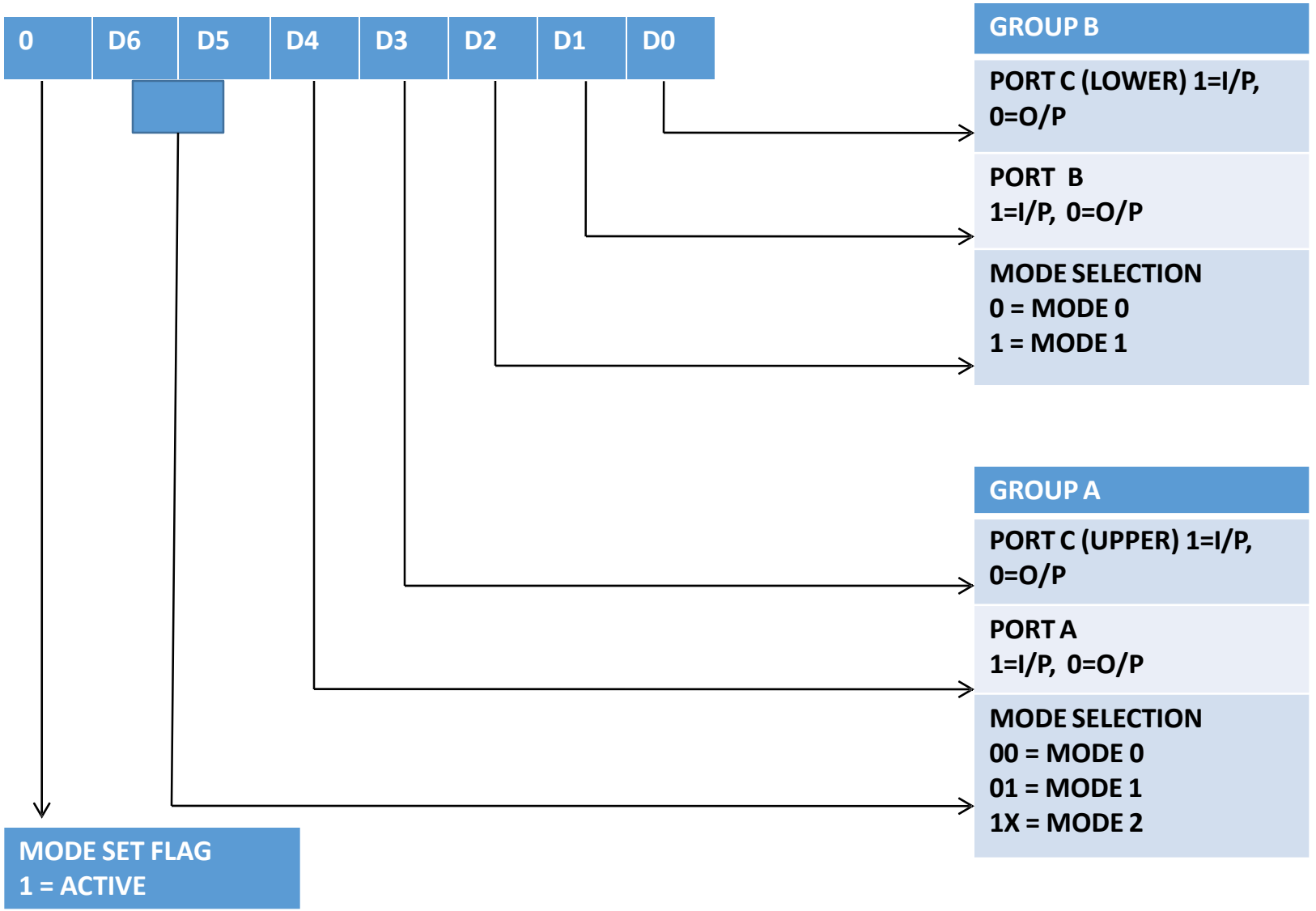
MODE 2

- Only group A can be initialized in this mode.
- Port A can be used for *bidirectional handshake* data transfer. This means that data can be input or output on the same eight lines (PA0 - PA7).
- Pins PC3 - PC7 are used as handshake lines for port A.
- The remaining pins of port C (PC0 - PC2) can be used as input/output lines if group B is initialized in mode 0.
- In this mode, the 8255 may be used to extend the system bus to a slave microprocessor.

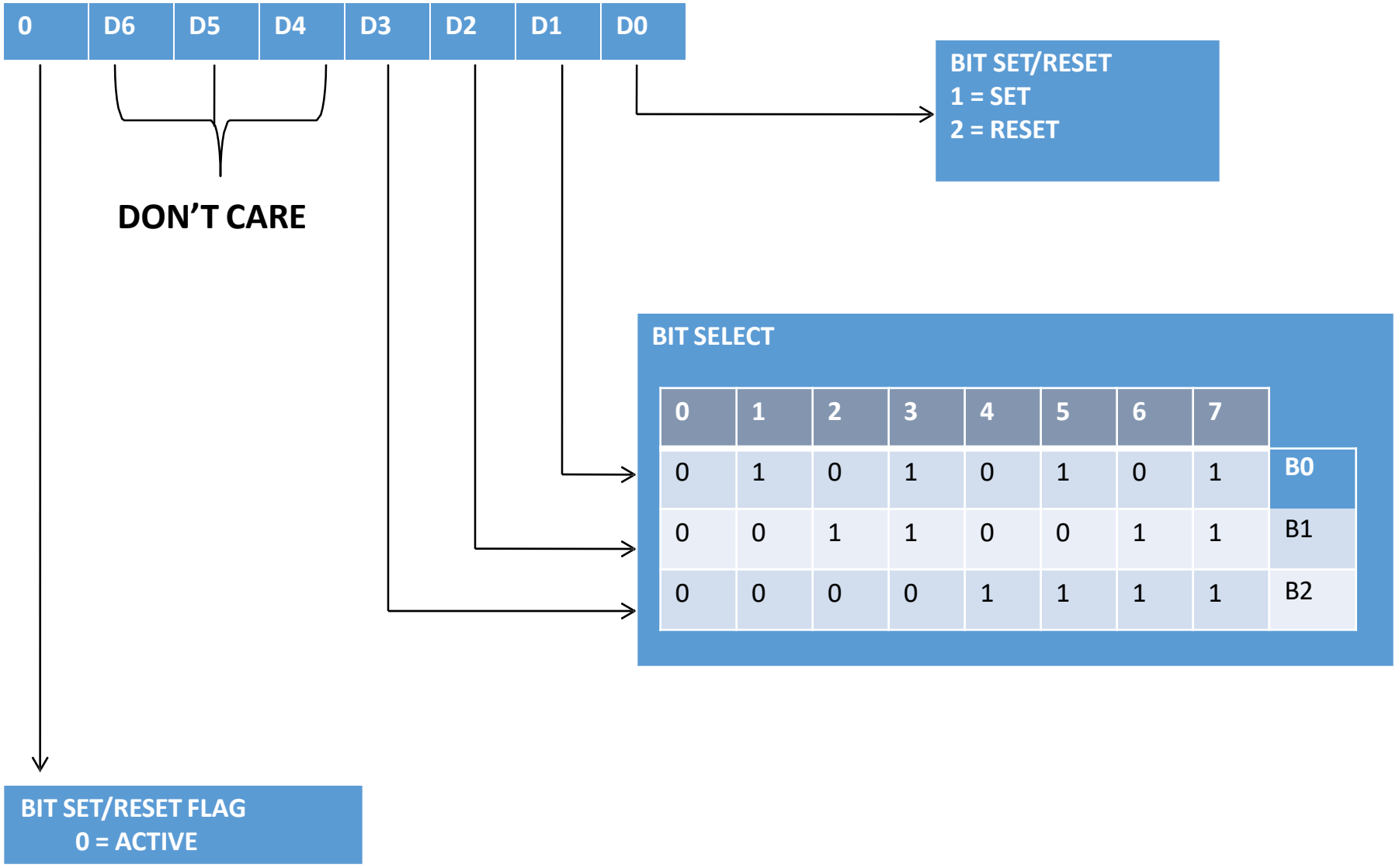


- Mode 0
  - Basic I/O
- Mode 1
  - Strobe I/O
- Mode 2
  - Bi-Dir Bus

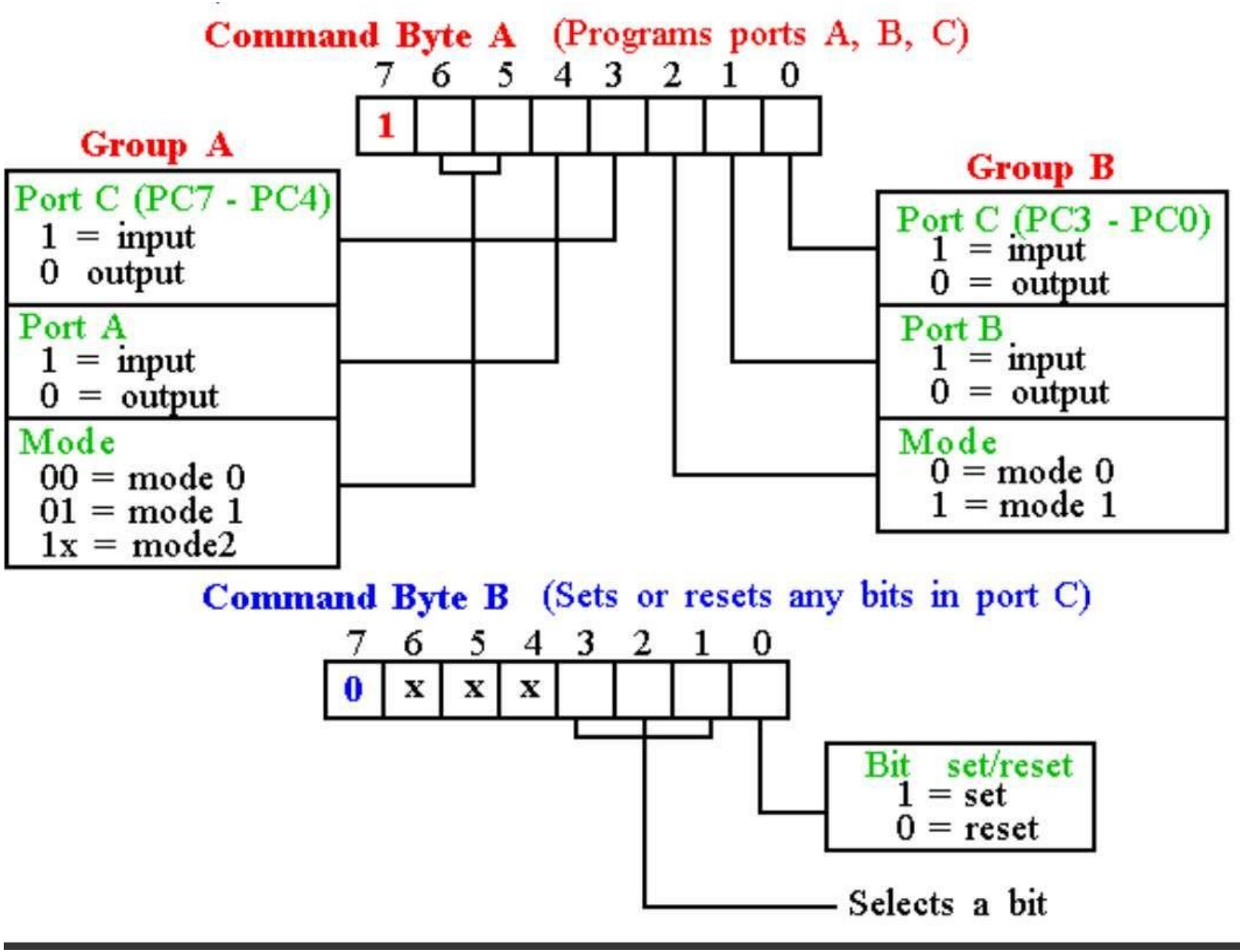
Control Word Format in I/O Mode



Control Word Format in BSR Mode

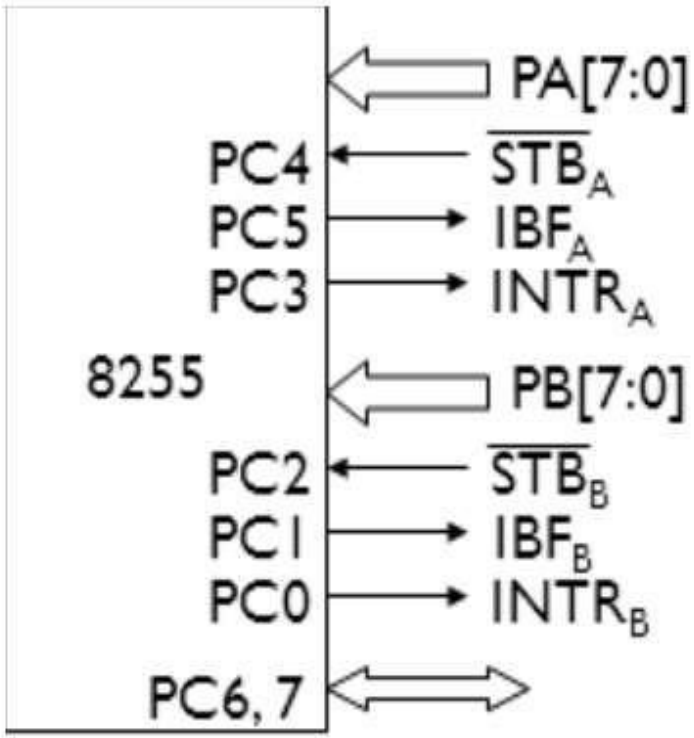


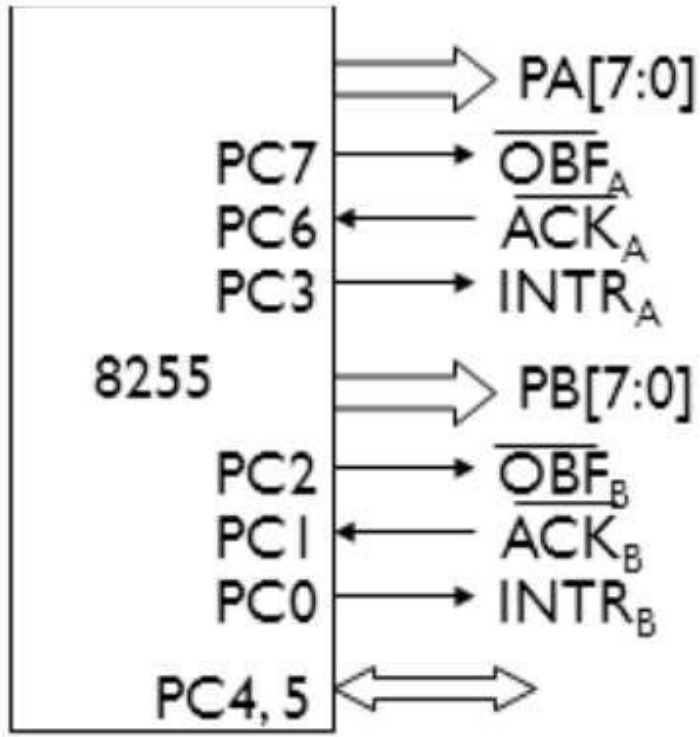
0	1	2	3	4	5	6	7	
0	1	0	1	0	1	0	1	B0
0	0	1	1	0	0	1	1	B1
0	0	0	0	1	1	1	1	B2


Command Byte A (Programs ports A, B, C)
7 6 5 4 3 2 1 0
1
Group A
Port C (PC7 - PC4)
1 = input
0 output
Port A
1 = input
0 = output
Mode
00 = mode 0
01 = mode 1
1x = mode2
Group B
Port C (PC3 - PC0)
1 = input
0 = output
Port B
1 = input
0 = output
Mode
0 = mode 0
1 = mode 1
Command Byte B (Sets or resets any bits in port C)
7 6 5 4 3 2 1 0
0 x x x
Bit set/reset
1 = set
0 = reset
Selects a bit

Mode 1:

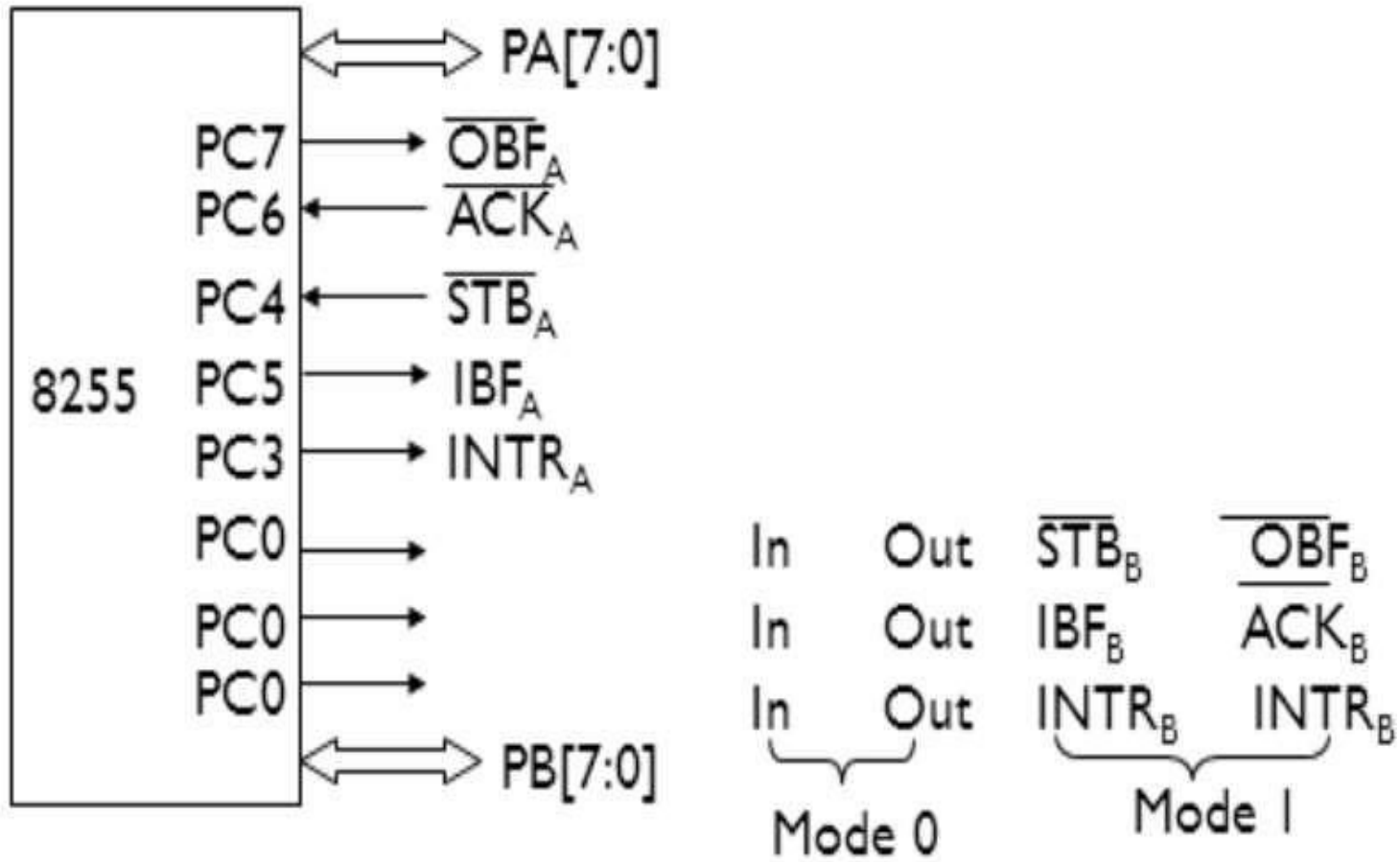
— Ports A and B are programmed as input or output ports
— Port C is used for handshaking





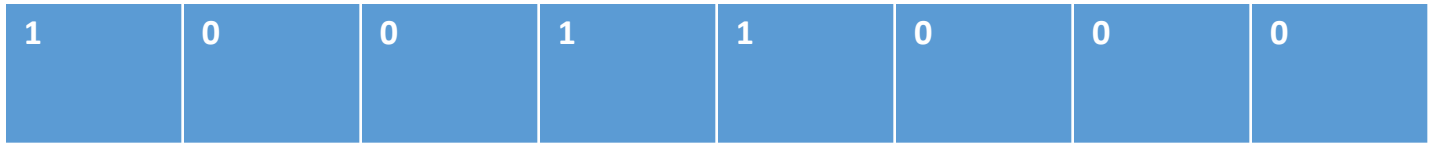
Mode 2:

- Port A is programmed to be bi-directional
- Port C is for handshaking
- Port B can be either input or output in mode 0 or mode 1



Write a program to initialize 8255 in the configuration below.(assume address of the CW register as 83H).
(1) Port A: simple input (2) Port B: simple output
(3) Port CL: output (4)Port CU: input

- **Solution**:



= 98H

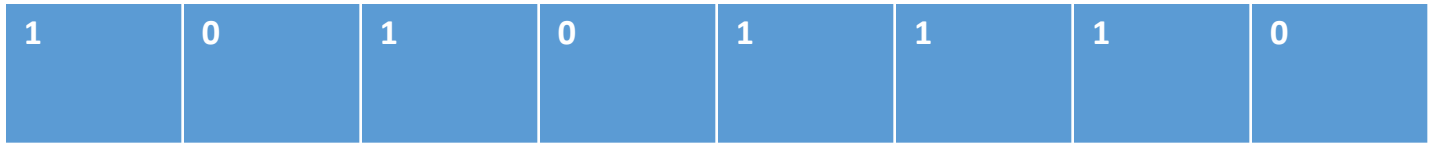
Program:

```
MVI A,98H  ; LOAD CONTROL WORD

OUT 83H    ; SEND CONTROL WORD
```

Write a program to initialize 8255 in the configuration below.(assume address of the CW register as 23H).
(1) Port A: output with handshake
(2) Port B: input with handshake
(3) Port CL: output (4)Port CU: input

- **Solution:**



= AEH

Program:

```
MVI A,AEH  ; LOAD CONTROL WORD

OUT 23H    ; SEND CONTROL WORD
```

Find the control word for the register arrangement of the ports of intel 8255 for mode 0 operation.

- **Port A: Output, Port B: Output,**
- **Port CU: Output, Port CL: Output**

Solution:

1	0	0	0	0	0	0	0

= 80H

The control word register for the above ports of intel 8255 is 80H.

Find the control word for the register arrangement of the ports of intel 8255 for mode 0 operation.

- **Port A: Input, Port B: Input,**
- **Port CU: Input, Port CL: Input**

Solution:

1	0	0	1	1	0	1	1

= 9BH

The control word register for the above ports of intel 8255 is 9BH.

TOPIC:

8254

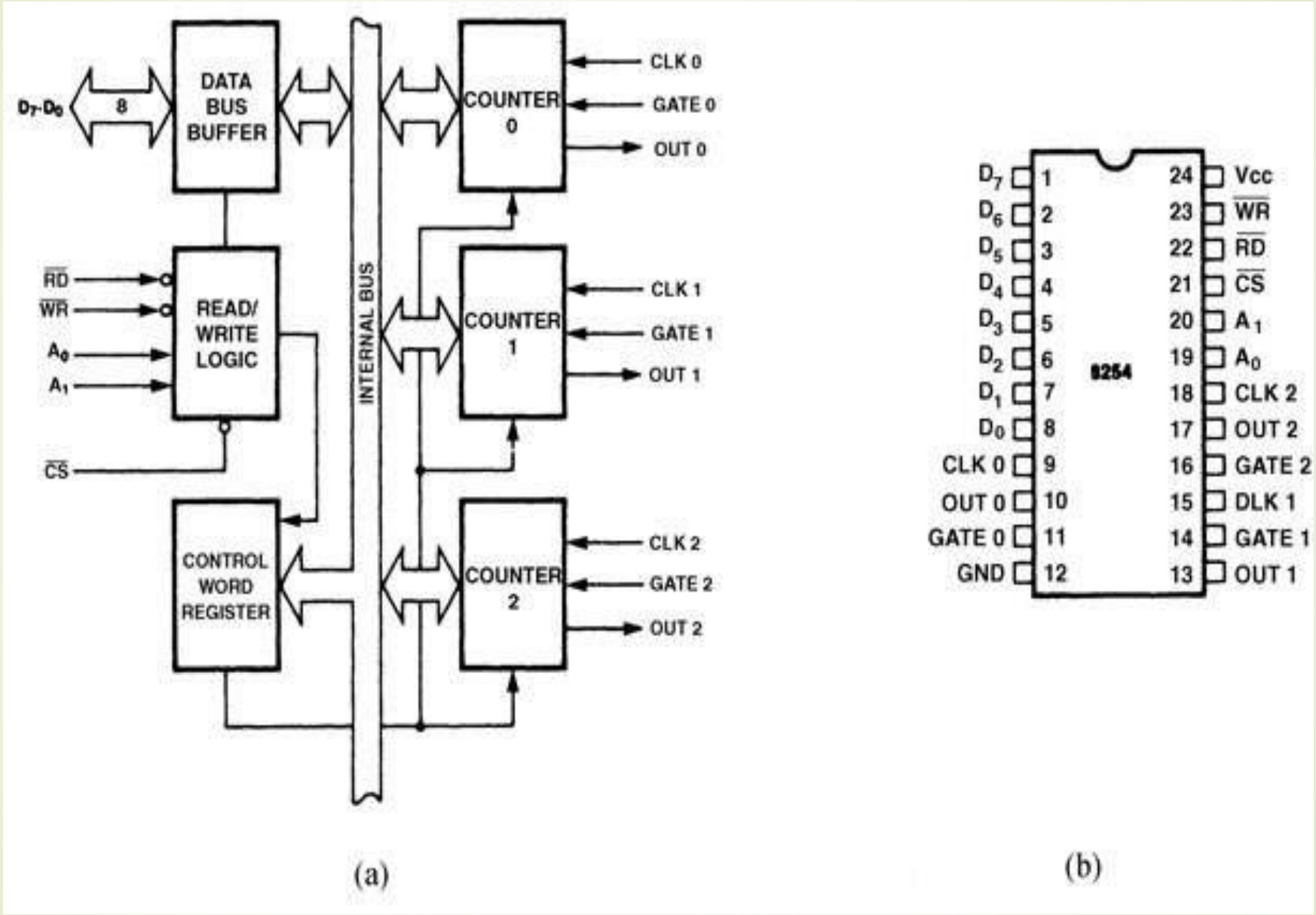
PROGRAMMABLE INTERVAL TIMER

Features of 8254 Timer

- It has 3 independent 16 bit down counters.
- Counters can be programmed in 6 different modes.
- Counting facility in both BCD and Binary number systems.
- It has powerful command called READ BACK COMMAND which allows the user to check the count value,programmed mode and current mode and current status of the counter.
- Operating frequency range is

  For 8253:- upto 2.6 MHz

  For 8254:- upto 10 MHz


DATA BUS BUFFER
8
D7-D0
READ/ WRITE LOGIC
RD
WR
A0
A1
CS
CONTROL WORD REGISTER
INTERNAL BUS
COUNTER 0
COUNTER 1
COUNTER 2
CLK 0
GATE 0
OUT 0
CLK 1
GATE 1
OUT 1
CLK 2
GATE 2
OUT 2
(a)
8254
D7 1
D6 2
D5 3
D4 4
D3 5
D2 6
D1 7
D0 8
CLK 0 9
OUT 0 10
GATE 0 11
GND 12
24 Vcc
23 WR
22 RD
21 CS
20 A1
19 A0
18 CLK 2
17 OUT 2
16 GATE 2
15 DLK 1
14 GATE 1
13 OUT 1
(b)

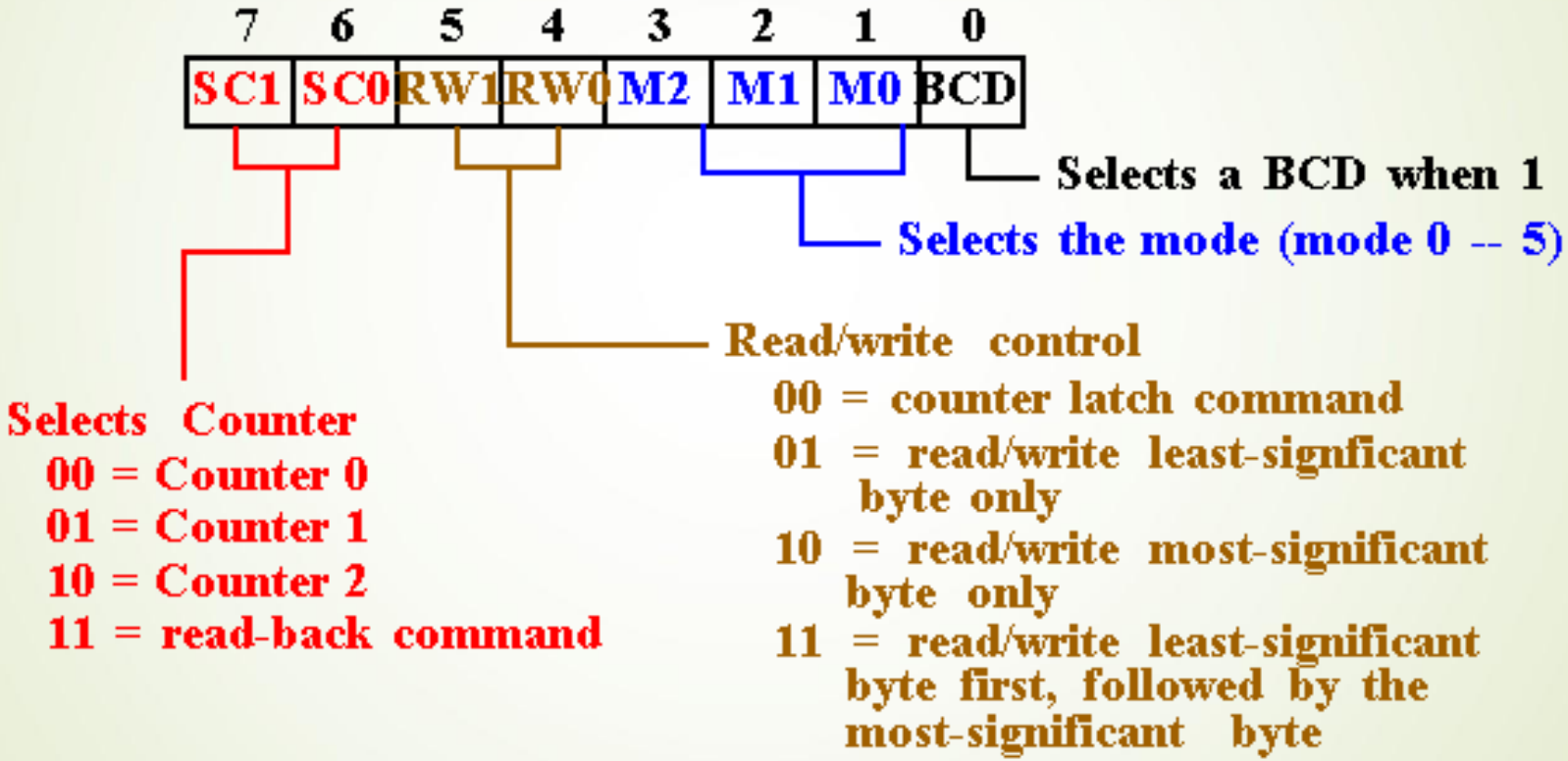
8254 Functional Description

Figure shows the pin-out of the 8254, a higher-speed version of the 8253, and a diagram of one of the three counters.

Each timer contains:

- *a CLK input which provides the basic operating frequency to the timer*
- *a gate input pin which controls the timer in some modes*
- *an output (OUT) connection to obtain the output of the timer*

8254 Control word

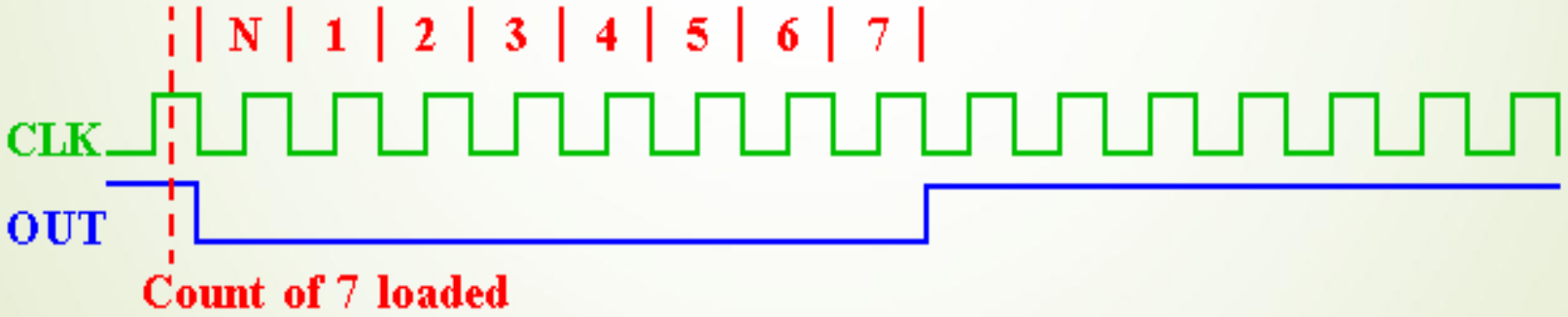


8254 Programming

- Each counter may be programmed with a count of 1 to FFFFH.
  - Minimum count is 1 all modes except 2 and 3 with minimum count of 2.
- Each counter has a program control word used to select the way the counter operates.
  - If two bytes are programmed, then the first byte (LSB) stops the count, and the second byte (MSB) starts the counter with the new count.

Mode 0: Interrupt on terminal count

- The output becomes a logic 0 when the control word is written
-and remains there until N plus the number of programmed counts
-When the Gate goes low the counter pauses. It resumes when Gates becomes high again



MODE 1:- Hardware retriggerable one shot.

The triggering must be done through the GATE input by sending a 0-to-1 pulse to it.

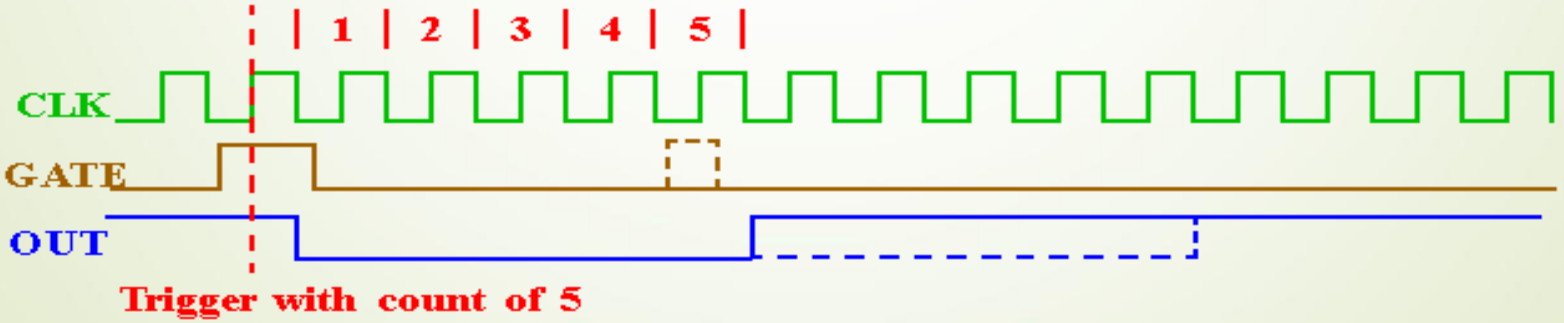
Steps:

1) Load the count register

2)A 0-to-1 pulse must be sent to the GATE input to trigger the count.

3)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.

4)If during the activation, a retriggered happened, then restart the down counting.

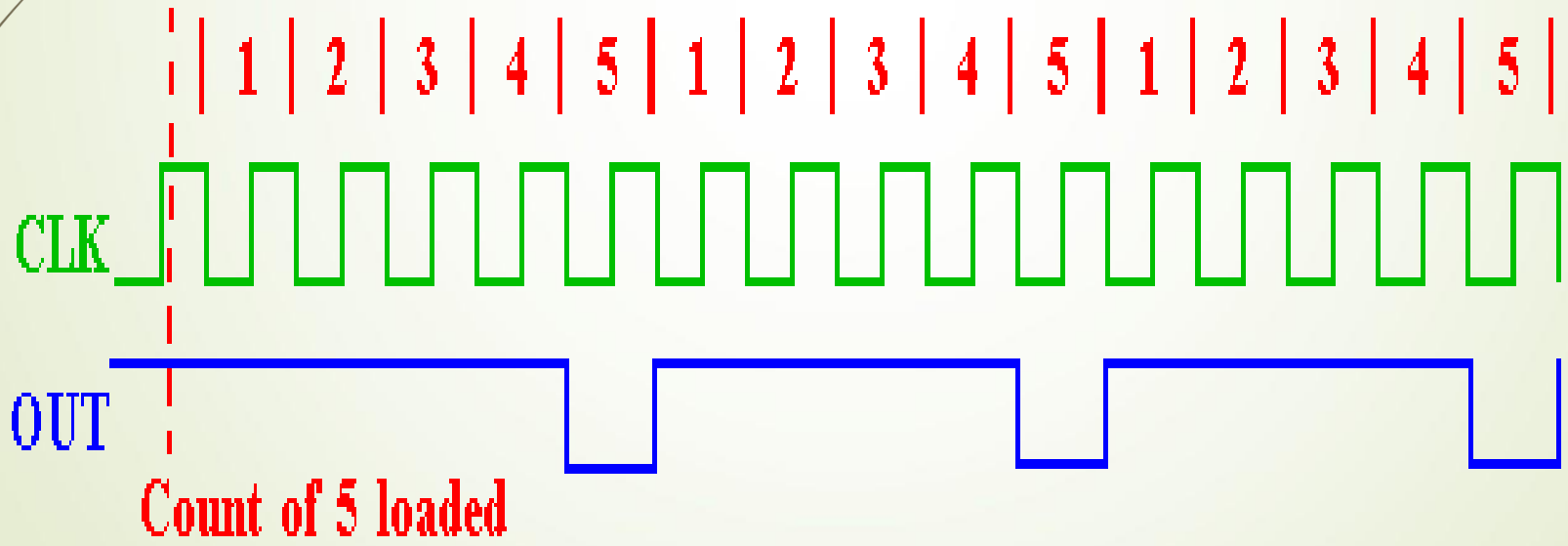


Mode 2: Rate Generator (Divide-by-N counter

*In Mode2, if GATE=1, OUT will be high for N*T, goes low only for one clock pulse, then counter is reloaded automatically,*

The cycle is repeated until reprogrammed or G pin set to 0.

Count =1 is illegal in this mode.

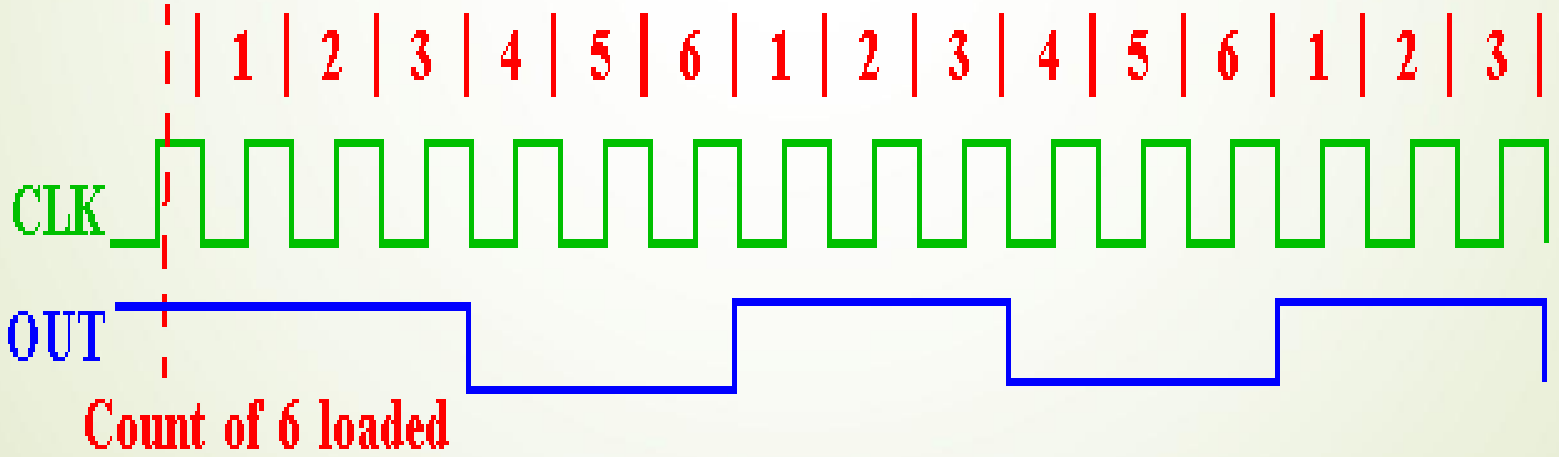


Mode 3: Square wave rate generator

Mode 3: *Generates a continuous square-wave with G set to 1.*

If count is even, 50% duty cycle otherwise OUT is high 1 cycle longer

High for (n+1)/2 & low for (n-1)/2 clock cycles

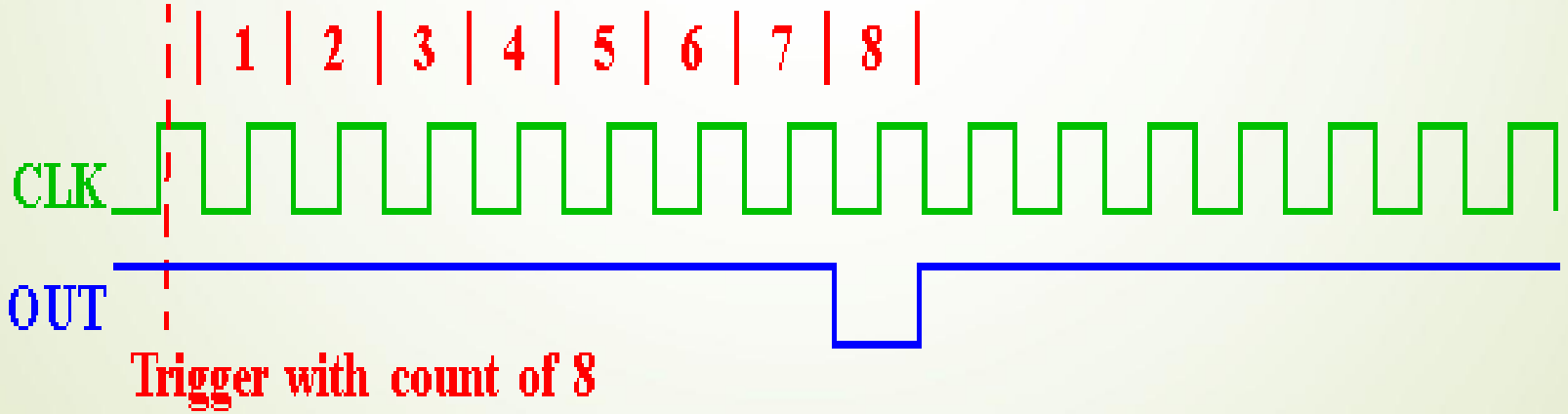


Mode 4: Software triggered strobe

In Mode4, *if GATE=1, the output will go high when loading the count, it will stay high for duration N*T.*

After the count reaches zero, it becomes low for one clock pulse, then goes high again and stays high until a new command word or new count is loaded

To repeat the strobe, the count must be reloaded



Mode 5: 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

Read Operations

- There are three possible methods for reading the counters:

1. a simple read operation
2. the Counter Latch Command
3. the Read-Back Command

1. Simple read operation :

The Counter which is selected with the A1, A0 inputs, the CLK input of the selected Counter must be inhibited by using either the GATE input or external logic.

Otherwise, the count may be in the process of changing when it is read, giving an undefined result.

2. Counter Latch Command:

- SC0, SC1 bits select one of the three counters
- Two other bits, D5 and D4, distinguish this command from a control word
- If a counter is latched and then, some time later, latched again before the count is read, the second counter latch command is ignored.
- The count read will be the count at the time the first counter latch command was issued.

A_1,A_0 = 11; CS = 0; RD = 1; WR = 0

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
SC1	SC0	0	0	X	X	X	X

SC1,SC0—specify counter to be latched

SC1	SC0	Counter
0	0	0
0	1	1
1	0	2
1	1	Read-Back Command

D5,D4—00 designates Counter Latch Command

X—don't care

NOTE:
Don't care bits (X) should be 0 to insure compatibility with future Intel products.

3. Read-back control command:

- The read-back control, word is used, when it is necessary for the contents of more than one counter to be read at a same time.
- Count : logic 0, select one of the Counter to be latched
- Status : logic 0, Status must be latched to be read status of a counter and is accessed by a read from that counter

$A0, A1 = 11 \quad \overline{CS} = 0 \quad \overline{RD} = 1 \quad \overline{WR} = 0$

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	1	$\overline{COUNT}$	$\overline{STATUS}$	CNT 2	CNT 1	CNT 0	0

D_5: 0 = Latch count of selected counter(s)
D_4: 0 = Latch status of selected counters(s)
D_3: 1 = Select Counter 2
D_2: 1 = Select Counter 1
D_1: 1 = Select Counter 0
D_0: Reserved for future expansion; Must be 0

Status register:

- Shows the state of the output pin
- Check the counter is in null state (0) or not
- How the counter is programmed

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
Output	Null Count	RW1	RW0	M2	M1	M0	BCD

D_7 1 = OUT Pin is 1
0 = OUT Pin is 0

D_6 1 = Null Count
0 = Count available for reading

D_5–D_0 Counter programmed mode

TOPIC:

8257

DIRECT MEMORY ACCESS

What is DMA

- A direct memory access (DMA) is an operation in which data is copied (transported) from one resource to another resource in a computer system without the involvement of the CPU.
- The task of a DMA-controller (DMAC) is to execute the copy operation of data from one resource location to another. The copy of data can be performed from:
  - I/O-device to memory
  - memory to I/O-device
  - memory to memory
  - I/O-device to I/O-device

Contd.

- A DMAC is an independent (from CPU) resource of a computer system added for the concurrent execution of DMA-operations.
- The DMAC replaces the CPU for the transfer task of data from the I/O-device to the main memory (or vice versa) which otherwise would have been executed by the CPU using the programmed input output (PIO) mode.
- The CPU initiates the transfer, does other operations while the transfer is in progress, and receives an interrupt from the DMA controller when the operation is done.

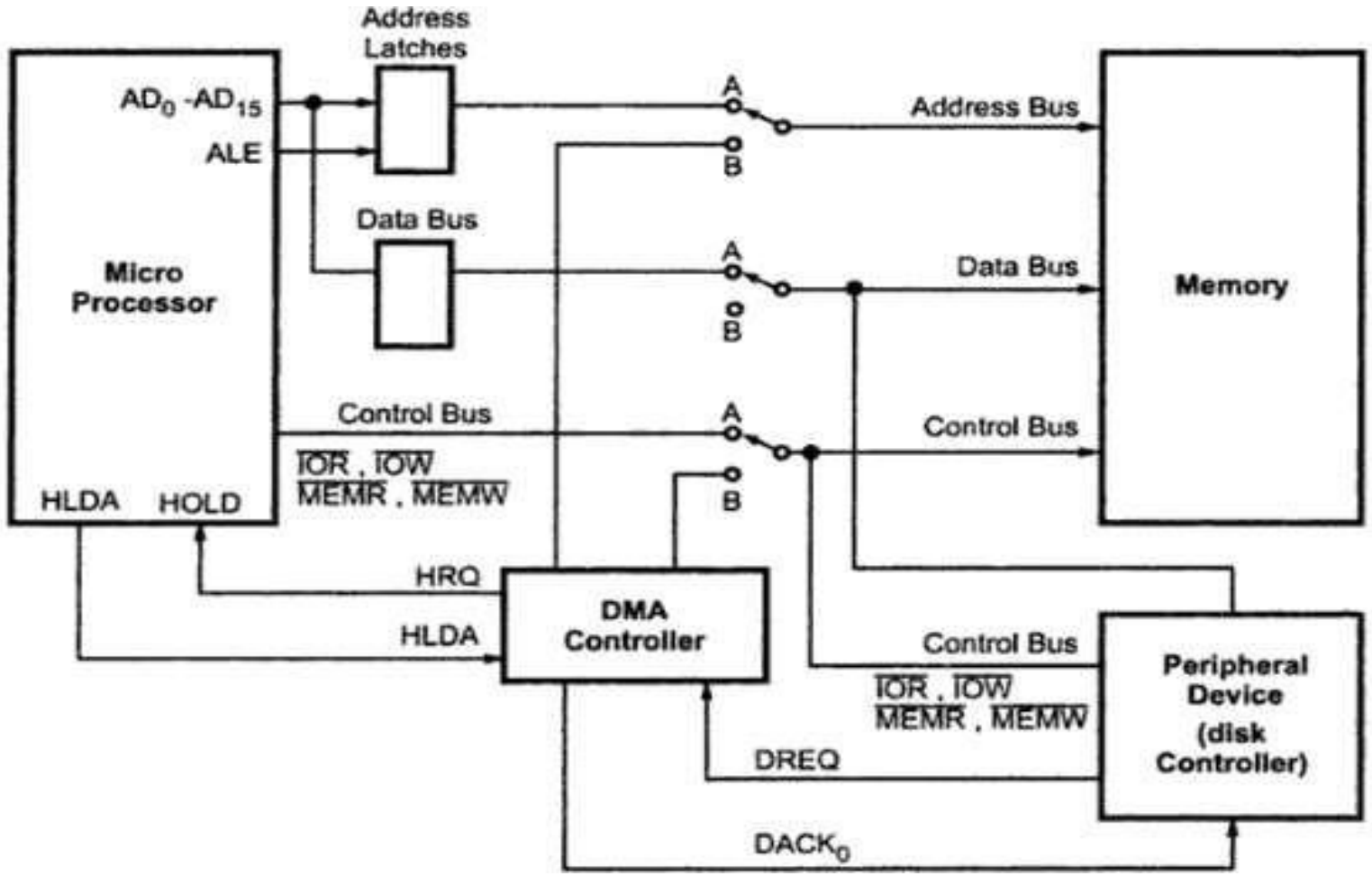
DMA Initialization

- DMA controllers require initialization by software. Typical setup parameters include the
  - base address of the source area,
  - base address of the destination area,
  - length of the block, and
  - whether the DMA controller should generate a processor interrupt once the block transfer is complete.

The 8257 offers three different modes of operation:

- (1) DMA read, which causes data to be transferred from memory to peripheral:
- (2) DMA write, which causes data to be transferred from peripheral to memory.
- (3) DMA verify, which does not actually involve the transfer of data.
  - When an 8257 channel is in the DMA verify mode, it will respond the same as described for transfer operations, except that no memory or I/O read/write control signals will be generated, thus preventing the transfer of data.
  - The 8257. however, will gain control of the system bus and will acknowledge the peripheral's DMA request for each DMA cycle.
  - The peripheral can use these acknowledge signals to enable an internal access of each byte of data block in order to execute some verification procedure, such as the accumulation of CRC (Cyclic Redundancy Code) checkword.

DMA Controller Operating in MPU



DMA Idle Cycle

- When the system is turned on, the switches are in the A position, so the buses are connected from the microprocessor to the system memory and peripherals. Microprocessor then executes the program until it needs to read a block of data from the disk.
- To read a block of data from the disk microprocessor sends a series of commands to the disk controller telling it to search and read the desired block of data from the disk into the disk buffer.
- When the disk controller is ready with the first byte of data in its buffer, it sends DMA request DRQ signal to the DMA controller.

Execution of a DMA-operation

- Initially, when any device has to send data between the device and the memory, the device has to send DMA request (DRQ) to DMA controller.
- The DMA controller sends Hold request (HRQ) to the CPU on HOLD line and waits for the CPU to assert the HLDA.
- Then the microprocessor 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 between the CPU, memory, and I/O devices.

DMA Active Cycle

- When DMA controller gets control of the buses, it sends the memory address where the first byte of data from the disk is to be written.
- It also sends DMA acknowledge, DACK signal to the disk controller device telling it to get ready for data transfer .
- Finally (in case of DMA write operation), it asserts both the IOR and MEMW signals on the control bus . Asserting the IOR signals enables the disk controller to output the data from the disk on the data bus and MEMW signal enables the addressed memory to accept data form the data bus.
- In this technique data is transferred directly from the disk controller to the memory location without passing through the CPU or the DMA controller.

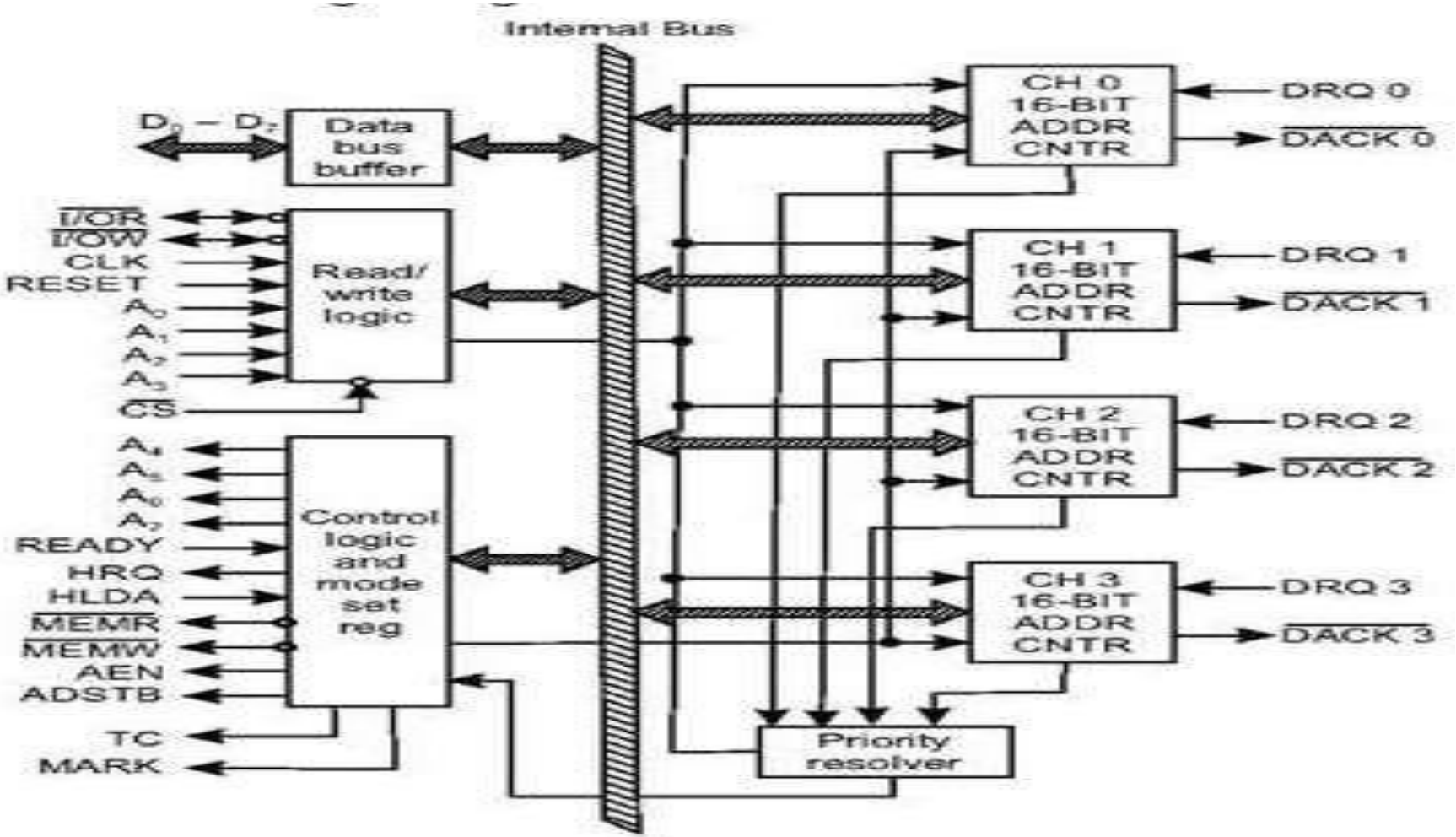
DMA Controller 8257

- 8257 is a programmable, 4 channel, DMA controller. Each channel can be programmed independently to transfer up to 64kb of data by DMA. Therefore we can interface 4 I/O devices with 8257.
- Each channel includes a 16-bit DMA address register and a 14-bit counter.
- DMA address register gives the address of the memory location and counter specifies the number of DMA cycles to be performed.
- As counter is 14-bit, each channel can transfer 2^{14} (16 kbytes) without intervention of microprocessor.
- It maintains the DMA cycle count for each channel and activates a control signal TC (Terminal count) to indicate the peripheral that the programmed number of DMA cycles are complete.

Contd.

- It provides another control signal MARK to indicate peripheral that the current DMA cycle is the 128^{th}cycle since the previous MARK output.
- It has priority logic that resolves the peripherals requests. The priority logic can be programmed to work in two modes, either in fixed mode or rotating priority mode.
- It allows data transfer in two modes : burst mode and cycle steal (single byte transfer) mode.
- It can execute three DMA cycles : DMA read, DMA write and DMA verify.
- It operates in two modes : slave and master , where in master mode it gains control over the system buses while in slave mode buses are under control of processor

8257 Architecture



The functional block diagram consists of

1. DMA channels
2. Data bus buffer
3. Read/Write logic
4. Control logic
5. Mode set Register
6. Status Register

1. DMA Channels

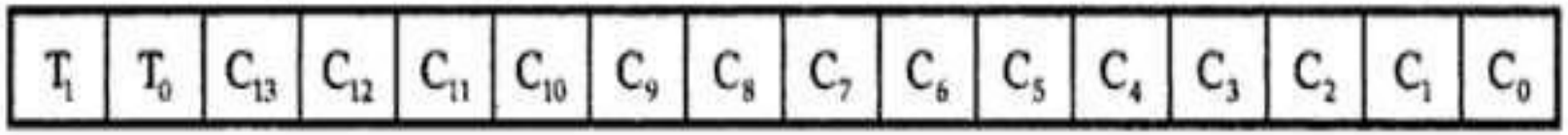
- The 8257 provides four separate DMA channels (labeled CH-0 to CH-3).
- Each channel includes two sixteen-bit registers:
  (1) DMA address register, and
  (2) Terminal count register
- Both registers must be initialized before channel is enabled.
- The DMA address register is loaded with the address of the first memory location to be accessed.
- The value loaded into the low-order 14-bits of the terminal count register specifies the number of DMA cycles minus one before the Terminal Count (TC) output is activated.
- In general, if the number of desired DMA cycles, load the value N-1 into the low-order 14-bits of the terminal count register.

DMA Channels

- The most significant two bits of the terminal count register specify the type of DMA operation for that channel.
- Each channel accepts DMA Request (DRQn) input and provides DMA Acknowledge (DACKn) output.
- **(DRQ0-DRQ3) : DMA Request**: These are individual asynchronous channel request inputs used by the peripherals to obtain DMA cycle.
- **(DACK0- DACK3) :- DMA Acknowledge:** An active low level on the acknowledge output informs the peripheral connected to that channel that it has been selected for DMA cycle.

Format of Terminal Count Register

- The most significant two bits of the terminal count register specify the type of DMA operation for that channel.



T_1	T_0	Type of operation
0	0	DMA verify cycle
0	1	DMA Write cycle
1	0	DMA READ cycle
1	1	Illegal

2. Data Bus Buffer

- This three-state, bi-directional, eight bit buffer interfaces the 8257 to the system data bus.

(D0-D7) :-

- Data Bus Lines: These are 8-bit bi-directional three-state lines.
- When the 8257 is being programmed by the CPU. Eight bits of data for DMA address register, terminal count register or the Mode Set register are received on the data bus.
- When the CPU reads DMA address register, terminal count register or the Status register, the data is sent to the CPU over the data bus.
- When 8257 is operating as Master, during a DMA cycle, it gains control over the system buses. In this mode, the 8257 sends out the 8 MSBs of the DMA address register of the channel being serviced on the D0-D7 pins at the starting of each DMA cycle to the 8212 latch. After this, the bus is released to handle the memory data transfer during the remaining DMA cycle.

3. Read/Write Logic

- When the CPU is programming or reading one of the 8257’s registers (i.e., when the 8257 is "slave" device on the system bus), the Read/Write Logic accepts the I/O Read or I/O Write signal, decodes the least significant four address bits, (A0-A3), and either writes the contents of the data bus into the addressed register (if I/OW is true) or places the contents of the addressed register onto the data bus (if I/OR is true).
- During DMA cycles (i.e., when the 8257 is the bus "master"), the Read/Write Logic generates the I/O read and memory write (DMA write cycle) or I/O Write and memory read (DMA read cycle) signals which control the data link with the peripheral that has been granted the DMA cycle.

Signals of Read/Write Logic

- **I/O Read:** An active-low, bi-directional three-state line. In the **"slave" mode**, it is an input which allows the 8-bit status register or the upper/lower byte of 16-bit DMA address register or terminal count register to be read.
- In the **"master" mode, I/O Read** is control output which is used to access data from peripheral during the DMA write cycle(reading data from I/O and write into memory).
- **I/O Write:** An active-low, bi-directional three-state line. In the **"slave" mode,** it is an input which allows the contents of the data bus to be loaded into the 8-bit mode set register or the upper/lower byte of 16-bit DMA address register or terminal count register.
- In the **"master" mode**, I/OW is control output which allows data to be output to peripheral during DMA read cycle (reading from memory and write into I/O).

Signals of Read/Write Logic

- **Clock Input**: Generally from an Clock Generator device) or Intel 8085A CLK output.
- **Reset:** An asynchronous input from 8085 which disables all DMA channels by clearing the mode register and tri-stated all control lines.
- **(A0-A3) :-** Address lines:- These least significant four address lines are bi-directional.
  - In the "slave" mode they are inputs which select one of the registers to be read or programmed.
  - In the "master" mode, they are outputs which constitute the least significant four bits of the 16-bit memory address generated by the 8257.

Signals of Read/Write Logic

- **Chip Select:**
  - It is active low, Chip select input line.
  - In the slave mode, it is used to select the chip.
  - In the master mode, it is ignored.

4.Control Logic

- This block controls the sequence of operations during all DMA cycles by generating the appropriate control signals and the 16-bit address that specifies the memory location to be accessed.
- **(A4-A7):-** Address Lines: These four address lines are tri-stated outputs which constitute bits through 4 to 7 of the 16-bit memory address generated by the 8257 during all DMA cycles.
- **Ready:** This asynchronous input is used to elongate the memory read and write cycles in the 8257 with wait states if the selected memory requires longer cycles.
- In master mode, When ready is high it is received the signal and when ready is low, it adds wait state between S1 and S3
- In slave mode ,this signal is ignored.

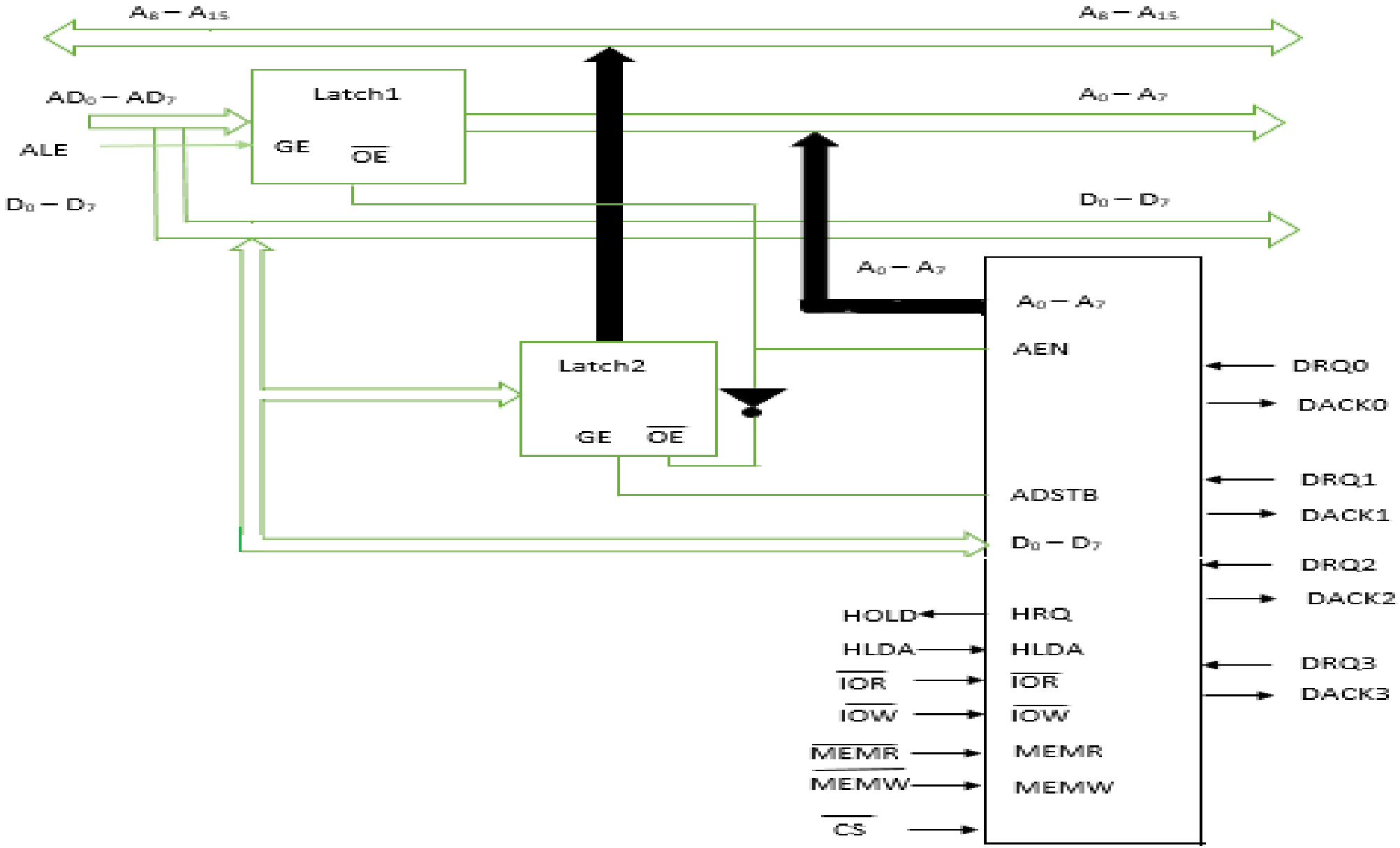
Control Signals of Control Logic

- **(HRQ)** Hold Request: This output requests control of the system bus. In systems with only one 8257, HRQ will normally be applied to the HOLD input on the CPU. HRQ must conform to specified setup and hold times.
- **(HLDA) Hold Acknowledge**: This input from the CPU indicates that the 8257 has acquired control of the system bus.
- **(MEMR) Memory Read:** This active-low three-state output is used to read data from the addressed memory location during DMA Read cycles.
- **(MEMW) Memory Write:** This active-low three-state output is used to write data into the addressed memory location during DMA Write cycles.
- **(ADSTB) Address Strobe: (**Equivalent to ALE pin of 8085), This line is connected with the latch of DMA to enable or disable it, when this line is 1, it strobes the most significant byte of the memory address into the latch from the data bus. So this pin is used to split data and address line from the DMA.

Control Signals of Control Logic

- **(AEN) Address Enable**: This output is used to disable (float) the System Data Bus and the System Control Bus.
- DMA generates 16 bit memory address, out of which the lower 8 bits are carried by A0- A7 pins of DMA , while the higher order address is carried by data lines (D0-D7) of DMA. So we need to demultiplex these data lines for which AEN pin is used.
- When DMA operates in the master mode, then AEN pin is used to disable the output from 8085's latch and to enable the output of DMA's latch, so the DMA can store the 8 MSB of the 16 bit memory address produced by DMA into its latch from where this higher order address goes to the higher order address bus and in the subsequent cycles DMA can carry data over its 8-bit data lines.
- In slave mode this pin is ignored.

INTERFACING OF 8257 WITH 8085



8257 Pin Description

- **DRQ_0–DRQ_3:** These are the four individual asynchronous channel DMA request inputs, which are used by the peripheral devices to obtain DMA services. When the rotating priority mode is selected, then DRQ0 will get the highest priority and DRQ3 will get the lowest priority among them.
- **$DACK_o$ – $DACK_3$:** These are the active-low DMA acknowledge lines, which updates the peripheral requesting device service about the status of their request by the CPU. These lines can also act as strobe lines for the requesting devices.
- **D_o – D_7 :** These are bidirectional, data lines which help to interface the system bus with the internal data bus of DMA controller In the Slave mode, command words are carried to 8257 and status words from 8257. In the master mode, the lines which are used to send higher byte of the generated address are sent to the latch. This address is further latched using ADSTB signal.

Contd.

- **IOR :** It is an active-low bidirectional tri-state input line, which helps to read the internal registers of 8257 by the CPU in the Slave mode. In the master mode, it also helps in reading the data from the peripheral devices during a memory write cycle.
- **IOW :** It is an active low bi-direction tri-state line, which helps in loading the contents of the data bus to the 8-bit mode register or upper/lower byte of a 16-bit DMA address register or terminal count register. In the master mode, it is used to load the data to the peripheral devices during DMA memory read cycle.
- **CLK:** It is a clock frequency signal which is required to perform internal operation of 8257.
- **RESET:** This signal is used to RESET the DMA controller by disabling all the DMA channels.
- **CS :** It is an active-low chip select line. In the Slave mode, it enables the read/write operations to/from 8257. In the master mode, it automatically disables the read/write operations to/from 8257.

Contd.

- **A_o - A_3 :** These are the four least significant address lines. In the slave mode, they perform as an input, which selects one of the registers to be read or written. In the master mode, they are the outputs which contain four least significant memory address output lines produced by 8257.
- **A_4 - A_7:** These are the higher nibble of the lower byte address generated by DMA in the master mode.
- **READY:** It is an active-high asynchronous input signal, which helps DMA to make ready by inserting wait states.
- **HRQ :** This signal helps to receive the hold request signal sent from the output device. In the slave mode, it is connected with a DRQ input line 8257. In Master mode, it is connected with HOLD input of the CPU.
- **HLDA :** It is the hold acknowledgement signal which indicates the DMA controller that the bus has been granted to the requesting peripheral by the CPU when it is set to 1.

Contd.

- **MEMR :** It is the low memory read signal, which is used to read the data from the addressed memory locations during DMA read cycles.
- **MEMW :** It is the active-low three state signal which is used to write the data to the addressed memory location during DMA write operation.
- **ADST :** This signal is used to convert the higher byte of the memory address generated by the DMA controller into the latches.
- **AEN :** This signal is used to disable the address bus/data bus.
- **TC:** It stands for 'Terminal Count', which indicates the present DMA cycle to the present peripheral devices.
- **MARK :** The mark will be activated after each 128 cycles or integral multiples of it from the beginning. It indicates the current DMA cycle is the 128th cycle since the previous MARK output to the selected peripheral device.
- **V_{cc} :** It is the power signal which is required for the operation of the circuit.

TOPIC:

8279 KEYBOARD & DISPLAY INTERFACE

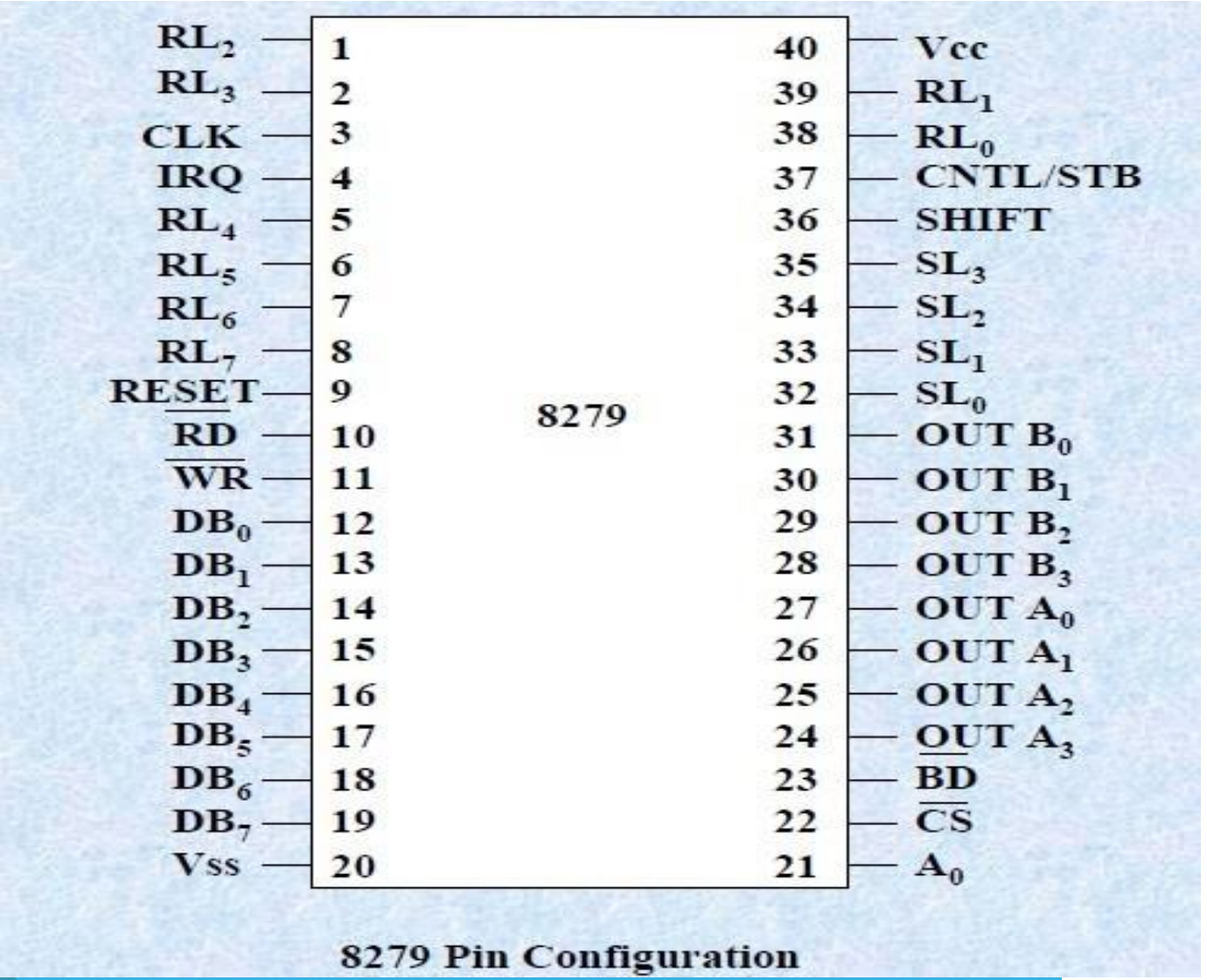
8279 KEYBOARD AND DISPLAY INTERFACING

Features:

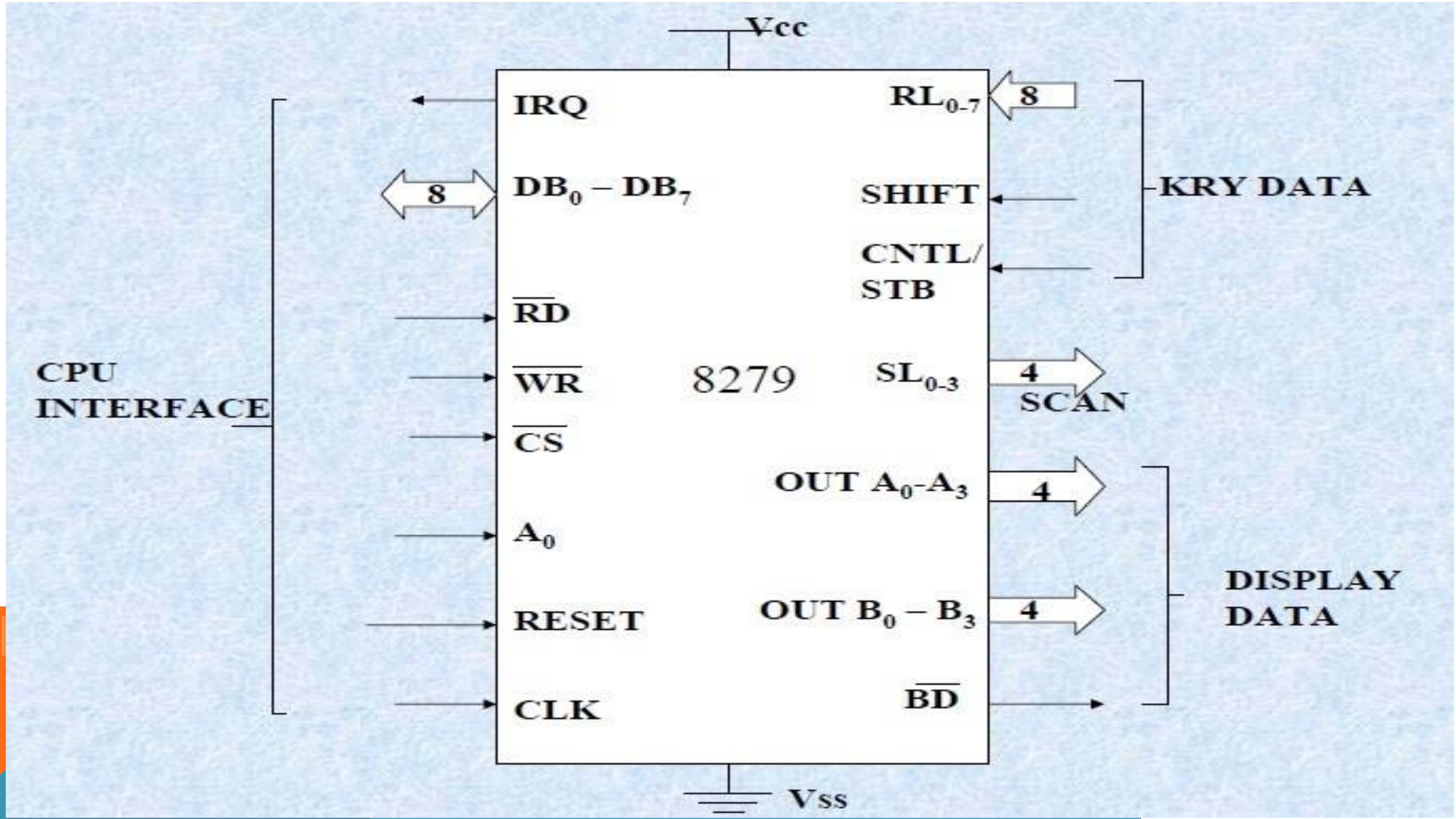
- It is designed by Intel.
- It is support 64 contact key matrix with two more keys “CONTROL” and “SHIFT”
- It provides 3 operating modes
  1. Scanned keyboard mode
  2.Scanned sensor matrix mode
  3.Strobed Input mode.
- It has inbuilt debounce key .

- It provides 16 byte display RAM to display 16 digits and interfacing 16 digits.
- It provides two output modes:
  1. Left entry (Typewriter type).
  2. Right entry (Calculator type).
- Simultaneous keyboard and display operation facility allows to interleave keyboard and display software.
- The interrupt output of 8279 can be used to tell CPU that the key press is detected, this eliminates the need of software polling.

PIN DIAGRAM OF 8279



8279 Pin Configuration


Vcc
IRQ
RL_{0-7}
8
8
$DB_0 - DB_7$
SHIFT
KRY DATA
CNTL/
STB
$\overline{RD}$
CPU
INTERFACE
$\overline{WR}$
8279
SL_{0-3}
4
SCAN
$\overline{CS}$
OUT A_0-A_3
4
A_0
DISPLAY
DATA
RESET
OUT $B_0 - B_3$
4
CLK
$\overline{BD}$
Vss

CPU INTERFACE PINS:

- **DB_0-DB_7 :** These are bidirectional data bus lines. The data and command words to and from the CPU are transferred on these lines.

- **RD, WR (Input / Output)**
  READ/WRITE: These input pins enable the data buffers to receive or send data over the data bus.
- **A0(Address lines) :** A high on this line indicates the transfer of a command or status information. A low on this line indicates the transfer of data. This is used to select one of the internal registers of 8279.

- **CS :** Chip Select – A low on this line enables 8279 for normal read or write operations. Other wise, this pin should remain high.
- **RESET :** This pin is used to reset 8279. A high on this line reset 8279. After resetting 8279, its in sixteen 8-bit display, left entry encoded scan, 2-key lock out mode. The clock prescaler is set to 31.
- **CLK :** This is a clock input used to generate internal timing required by 8279.

- **IRQ :** This interrupt output lines goes high when there is a data in the FIFO sensor RAM. The interrupt lines goes low with each FIFO RAM read operation but if the FIFO RAM further contains any key-code entry to be read by the CPU, this pin again goes high to generate an interrupt to the CPU.
- **Vss, Vcc :** These are the ground and power supply lines for the circuit.
- **SL0-SL3-Scan Lines :** These lines are used to scan the key board matrix and display digits. These lines can be programmed as encoded or decoded, using the mode control register.

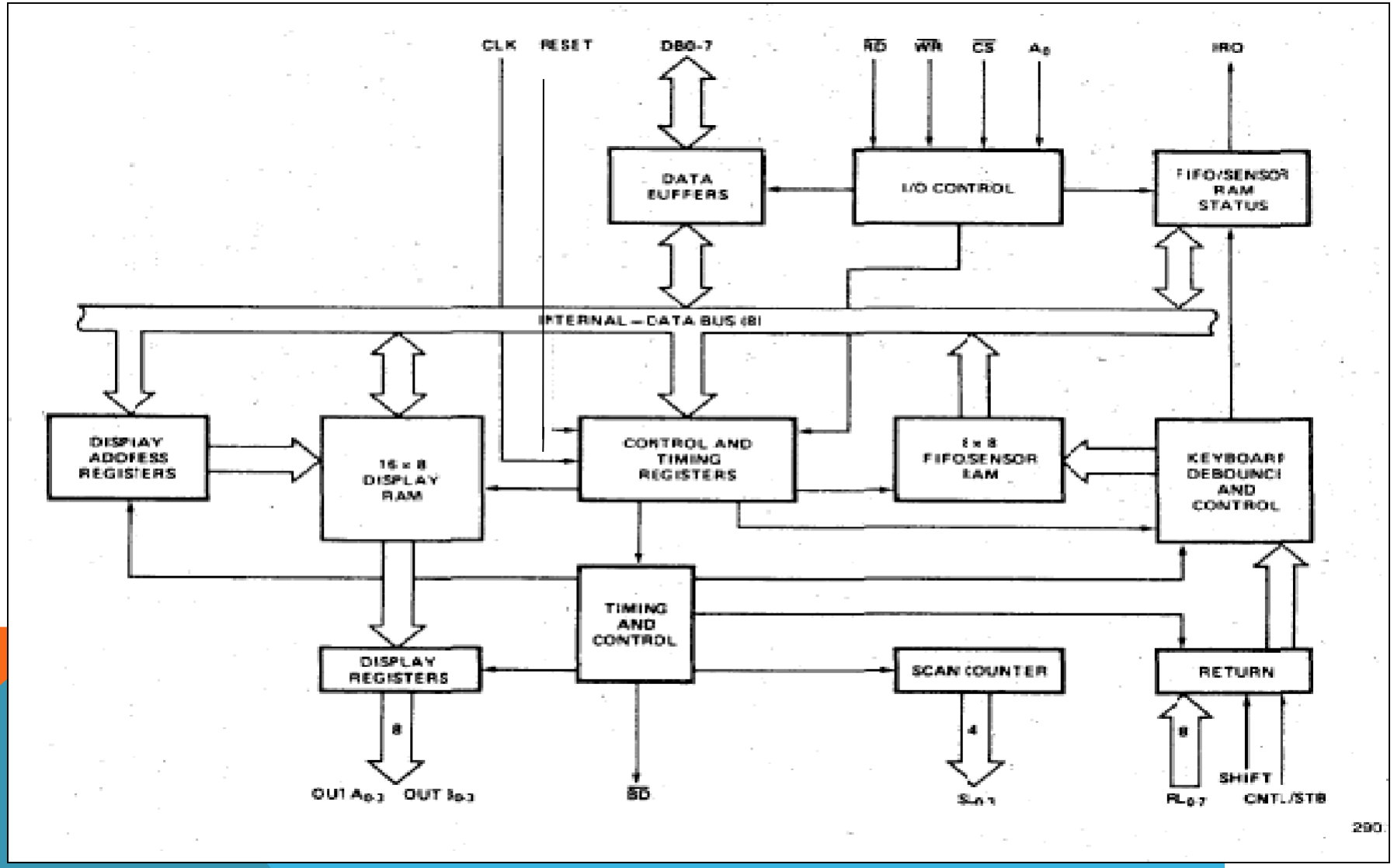
KEY BOARD DATA

- **RL0 - RL7 - Return Lines :** These are the input lines which are connected to one terminal of keys, while the other terminal of the keys are connected to the decoded scan lines. These are normally high, but pulled low when a key is pressed.
- **SHIFT :** The status of the shift input lines is stored along with each key code in FIFO, in scanned keyboard mode. It is pulled up internally to keep it high, till it is pulled low with a key closure.
- **CNTL/STB- CONTROL/STROBED I/P Mode :** In keyboard mode, this lines is used as a control input and stored in FIFO on a key closure. The line is a strobed lines that enters the data into FIFO RAM, in strobed input mode. It has an interrupt pull up. The lines is pulled down with a key closer.

DISPLAY DATA

- **OUT A0 – OUT A3 and OUT B0 – OUT B3 :** These are the output ports for two 16*4 or 16*8 internal display refresh registers. The data from these lines is synchronized with the scan lines to scan the display and keyboard. The two 4-bit ports may also as one 8-bit port.
- **BD – Blank Display :** This output pin is used to blank the display during digit switching or by a blanking closure.

BLOCK DIAGRAM



➢**It consists 4 main section.**

1. CPU interface and control section.
2. Scan section
3. Keyboard Section
4. Display section.

CPU INTERFACE AND CONTROL SECTION:

It consists of

1. Data buffers
2. I/O control
3. Control and timing registers.
4. Timing and control logic.

Data Buffers:

- 8-bit bidirectional buffer.
- Used to connect the internal data bus and external data bus.

I/O control:

- I/O control section uses the A0,CS,RD and WR signals to controls the data flow.
- The data flow is enabled by CS=0otherwise it is the high impedance state.
- A0=0 means the data is transferred.
- A0=1 means status or command word is transferred.

I/O CONTROL SIGNALS LISTED BELOW

A0	RD	WR	Interpretation
0	1	0	Data from CPU to 8279
0	0	1	Data from 8279 to CPU
1	1	0	Command word from CPU to 8279
1	0	1	Status word from 8279 to CPU

TIMING AND CONTROL REGISTERS:

- Store the keyboard and display modes and others operating condition programmed by the CPU.
- The modes are programmed by sending proper command $A_0=1$.

TIMING AND CONTROL:

- It consist timing counter chain.
- First counter is divided by N prescalar that can be programmed to give an internal frequency of 100 KHz.

THE OTHER COUNTER IS DIVIDE BY BASIC INTERNAL FREQUENCY

Parameter	Timings
Keyboard and time	5.1 m sec
Keyboard and debounce time	10.3 m sec
Key scan time	80 µ sec
Display scan time	10.3 m sec
Digit ON time	480 µ sec
Blanking time	160 µ sec
Internal clock time	10 µ sec

SCAN SECTION

- It has two modes,

1. Encoded mode
2. Decoded mode.

ENCODED MODE:

- It provide binary count from 0000 to 1111 by four scan lines(SC_3-SC_0)by active high inputs.
- It is externally decoded to provide 16 scan lines

- Display use all 16 lines to interface 16 digit 7 segment display.
- But keyboard use only 8 scan lines out of 16 lines.

DECODED MODE:

- In this mode ,the internal decoder decodes the least 2 significant bits.
- It is provide four possible combination from (SC_0-SC_3) such as 1110 ,1101 ,1011 and 0111.
- This four active low outputs line is used to directly to interface 4 –digit 7-segment display ,8*4 matrix keyboard

KEYBOARD SECTION

- This is consist of,
- Return buffers.
- Keyboard debounce control.
- FIFO / sensor RAM.
- FIFO / sensor RAM status.

RETURN BUFFERS:

- 8 return lines(RL_7-RL_0) are buffered and latched by when each row scan in scanned keyboard or sensor matrix mode.
- In strobed mode ,the contents of return lines are transferred to FIFO Ram.

KEYBOARD DEBOUNCE AND CONTROL:

- It is enabled only when keyboard mode is selected.
- In this mode , return lines are scanned whether any keys are closed in the row.
- If debounce circuit is detect any closed switch it wait about 10 msec.
- It is continued , the status of SHIFT and CONTROL keys are transferred into RAM.

FIFO/SENSOR RAM:

- This is a dual function of 8*8 RAM.
- In scanned key board mode and Strobed input mode , It is FIFO.
- Each new entry is written into successive RAM position and read in the order of entry.
- In sensor matrix mode it is a sensor RAM.
- Each sensor RAM is loaded with corresponding sensor RAM status.

- FIFO/SENSOR RAM status:
- This is used to tell the status of FIFO/SENSOR RAM.
- The status of logic also makes IRQ signal is High , When FIFO is empty.

DISPLAY SECTION:

It consists of,

1. Display RAM.
2. Display Address registers.
3. Display registers.

DISPLAY RAM:

- It is a 16*8 RAM.
- Which stores 16 digits display codes.
- It can be accessed by CPU directly.
- In Decoded mode,8279 uses only first four location of Display RAM.
- In Encoded mode,8279 uses only first eight location of Display RAM.
- And all 16 location for 16 digits display.

DISPLAY ADDRESS REGISTERS:

- Used to hold address of the byte currently write or read by the CPU and scan count value.
- In auto increment mode, address in the register is automatically incremented for each write or read.

DISPLAY REGISTERS:

- It is a Two 4-bit registers such as , A and B.
- They hold the bit patterns of character to be displayed.
- The content of display registers A and B can B blanked and inhibited individually.

OPERATING MODES

- It is two types,

1. Input modes.
2. Display modes.

INPUT MODES:

➢ It is basically 3 types,

1. Scanned keyboard.
2. Scanned sensor matrix.
3. Strobed mode.

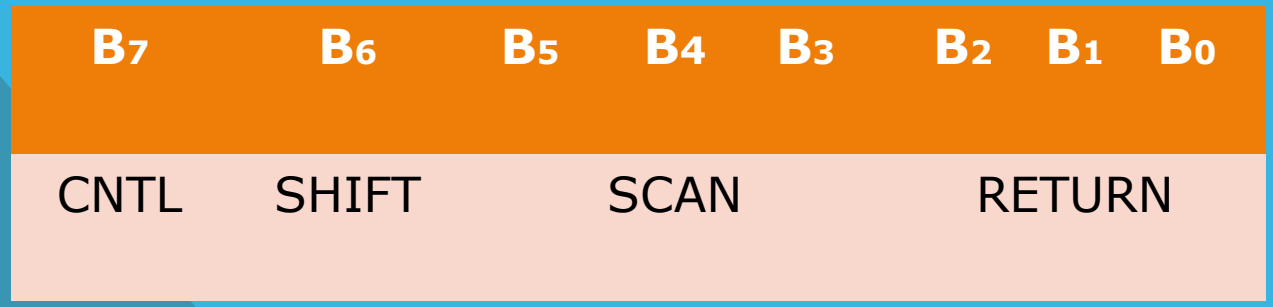
SCANNED KEYBOARD:

Key board can be scanned in two ways.

1.Encoded Scan 2.Decoded Scan.

ENCODED SCAN:

- In this scan, scan lines (SL_2-SL_0) are decoded externally to provide 8 scan lines.
- Additionally it provides 8 return lines.
- So the size of matrix keyboard is 8*8 (i.e Scan * Return)=64.
- When the key is pressed , it is stored the status of return lines , Scan lines ,SHIFT and CNTL/STB keys into FIFO RAM.
- The Scanned keyboard structure is,



Example:

- Find the key code for given condition below:

CNTL/STB SHIFT keys are open.

The pressed keys are to scan lines 2 and return lines 4.

SOLUTION:

B_7	B_6	B_5	B_4	B_3	B_2	B_1	B_0
1	1	0	1	0	1	0	0

- CNTL=1
- SHIFT=1
- Scan mode=010 (Scan line 2)
- Return mode=100 (Return line 4)
- Key code =D4 H

DECODED SCAN:

- In this mode ,internal decoder decodes the least significant bits of scan lines (SC3-SC0).
- That is provide the four combination such as 1110,1101,1011 and 0111.
- So the maximum size of keyboard is 8*4=32.
- The key code is similar to encoded code , only bit 5 (B_5) is always zero.

2-KEY LOCKOUT:

- In this mode, the two key depression is not allowed.
- When any key is depressed, the debounce logic is set and 8279 checks for any key depress next two scans.
- Three possible condition to avoid debouncing:

- **Condition 1:**

- If other key depression is not found during next two scan, it is a single key is depressed .Then the status of key code is entered into FIFO RAM along with the status of CNTL and SHIFT lines

- If FIFO RAM is empty , The CPU is entry the data.
- If FIFO RAM is full , The CPU does not entry the data.

Condition 2:

- If any other key depress is encountered , no entry to the FIFO can occur.
- When the key is released after that only Entry will be allowed.

Condition3:

- If the two key is pressed in simultaneously in a debounce cycle, both depression is not considered.

- **N-KEY ROLLOVER:**

➢Each key is depression is treated as independently from all others.

SCANNED SENSOR MATRIX:

➢In this mode , image of the sensor matrix is kept in the sensor RAM.

➢The status of sensor switches are input directly to the sensor RAM.

➢8279 scans row one by one and store the status of each row in the corresponding memory location.

STROBED INPUT MODE:

➢ The data is entered from Returned lines.

DISPLAY MODES

- It is basically two types,

1. Left entry (Type writer mode).
2. Right entry (Calculator mode).

LEFT ENTRY:

- In this mode , 8279 display characters from left to right.
- Like a typewriter.

AUTOINCREMENT IN LEFT ENTRY:

- In left entry mode , Autoincrement flag is set after each operation display RAM address is incremented.

RIGHT ENTRY:

- In right entry mode , Auto increment flag is set after each operation display
- RAM address is incremented.

AUTO INCREMENT IN RIGHT ENTRY:

- In this mode , 8279 display characters from Right to left.

  Like a Calculator.

Thank You