18ECC203J – Module 3

8086 Interfacing with Memory and Programmable Devices

S – 1, 2, 3

S – 1

Semiconductor memory interfacing & Dynamic RAM interfacing

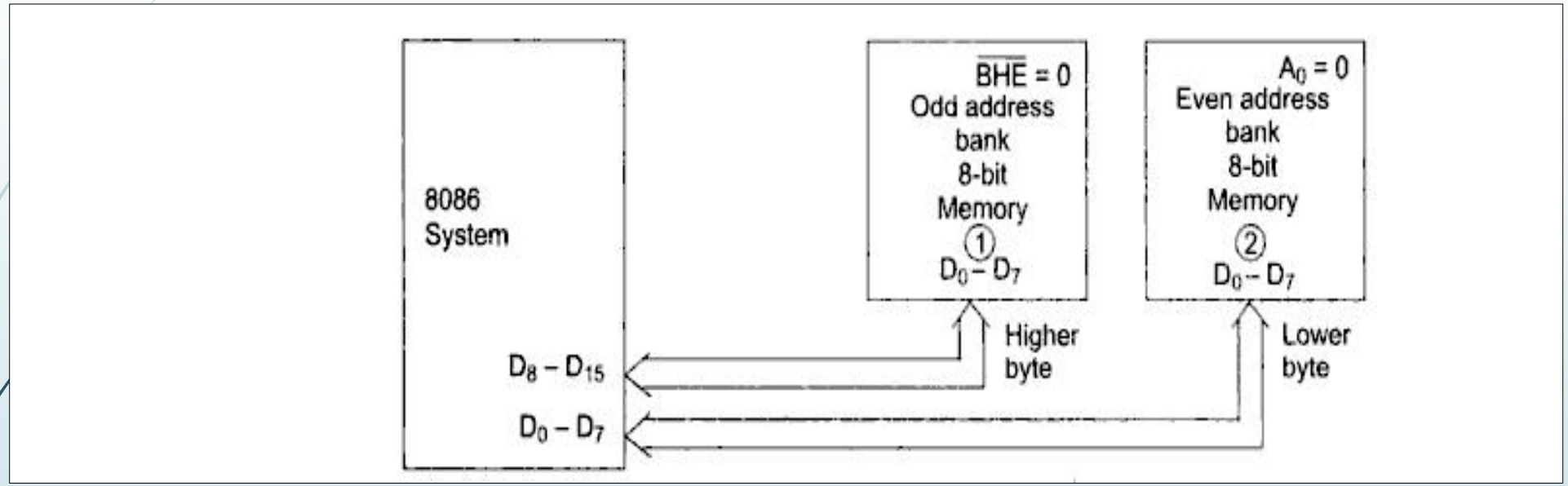
Semiconductor RAM

- The semiconductor RAM is broadly two types – Static RAM and Dynamic RAM.
- Semiconductor memory organization
  - Memory is organised as two-dimensional arrays of memory locations.
  - 4K * 8 or 4K byte memory contains 4096 locations. Each location contains 8 bits
  - Only one of the 4096 locations can be selected at a time.
  - Once selected, all bits are available in Data bus.
  - For addressing the 4K bytes of memory, 12 address lines are required.
  - N memory locations means, n address lines required i.e. $n = \text{Log}_2 N$.

    If the µP has n address lines, then it is able to address at the most N locations of memory, where $2^n = N$. If out of N locations only P memory locations are to be interfaced, then the least significant p address lines out of the n lines of µP are connected to the memory chip while the remaining (n-p) higher order address lines are used as inputs to the chip selection logic.
- The memory address depends upon the hardware circuit used for decoding the chip select ($\overline{CS}$). The output of the decoding circuit is connected with the $\overline{CS}$ pin of the memory chip.

RECAP:- Physical Memory Organisation

(Module 1, Session 6, Slide No. 12)



- Certain locations in memory are reserved for specific CPU operations. The locations from FFFF0H to FFFFFH are reserved for operations including jump to initialisation programme and I/O-processor initialisation.
- The locations 00000H to 003FFH are reserved for *interrupt vector table*.

Procedure to Interface RAM

1. Arrange the available memory chip so as to obtain 16- bit data bus width. The upper 8-bit bank is called as 'odd address memory bank' and the lower 8-bit bank is called as 'even address memory bank'.

2. Connect available memory address lines of memory chip with those of the µP and also connect the memory $\overline{RD}$ and $\overline{WR}$ inputs to the corresponding processor control signals. Connect the 16-bit data bus of the memory bank with that of the microprocessor 8086.

3. The remaining address lines of the µP, $\overline{BHE}$ and A_0 are used for decoding the required chip select signals for the odd and even memory banks. The $\overline{CS}$ of memory is derived from the o/p of the decoding circuit.

As a good and efficient interfacing practice, the address map of the system should be continuous as far as possible, i.e. there should not be no windows in the map and no fold back space should be allowed.

Example Problem

- **Problem:** Interface two 4K x 8 EPROMS and two 4K x 8 RAM chips with 8086. Select suitable maps.
- **Solution:** We know that, after reset, the IP and CS are initialized to form address FFFF0h. Hence, this address must lie in the EPROM. The address of RAM may be selected any where in the 1 MB address space of 8086, but we will select the RAM address such that the address map of the system is continuous.

Memory Map Table

Address (A_{XY})	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
FFFFFh	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	EPROM							8K x 8												
FE000h	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
FDFFFh	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1
	RAM							8K x 8												
FC000h	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0

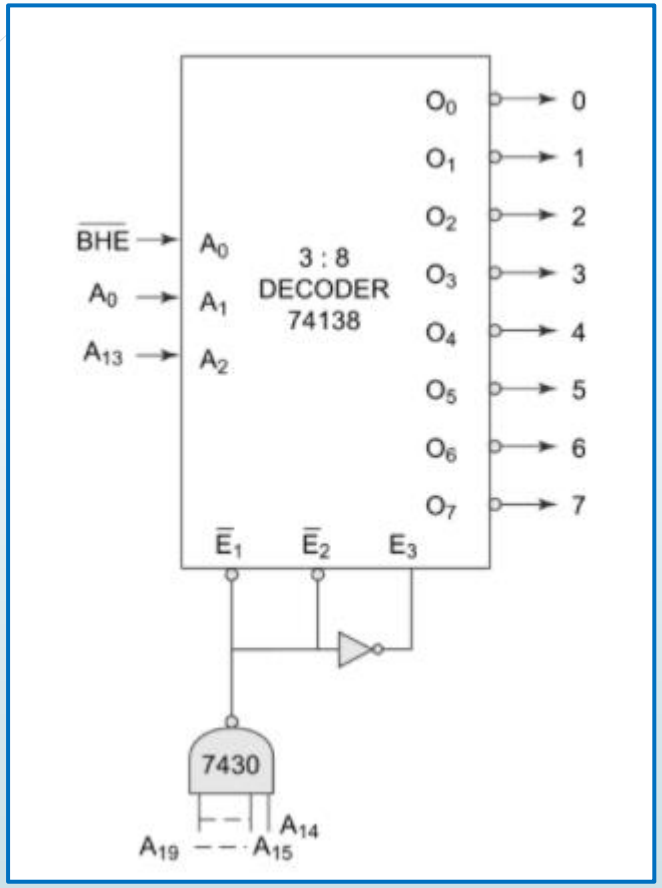
Solution (cont...)

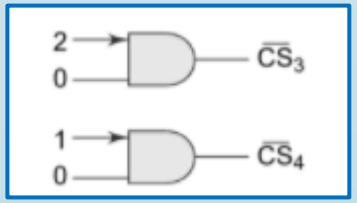
- Total 8K bytes of EPROM need 13 address lines $A_0 – A_{12}$ (since 2^{13} = 8K). Address lines $A_{13} – A_{19}$ are used for decoding to generate the chip select. The $\overline{BHE}$ signal goes low when a transfer is at odd address or higher byte of data is to be accessed.
- The memory system in this example contains in total four 4K x 8 memory chips.
- The two 4K x 8 chips of RAM and ROM are arranged in parallel to obtain 16-bit data bus width. If A0 is 0, i.e the address is even and is in RAM, then the lower RAM chip is selected indicating 8-bit transfer at an even address.
- If A_0 is 1, i.e., the address is odd and is in RAM, the $\overline{BHE}$ signal goes low, the upper RAM chip is selected, further indicating that the 8-bit transfer is at an odd address.
- If A_0 and $\overline{BHE}$ both are at 0, both the RAM and ROM chips are selected i.e. the data transfer is 16 bits.

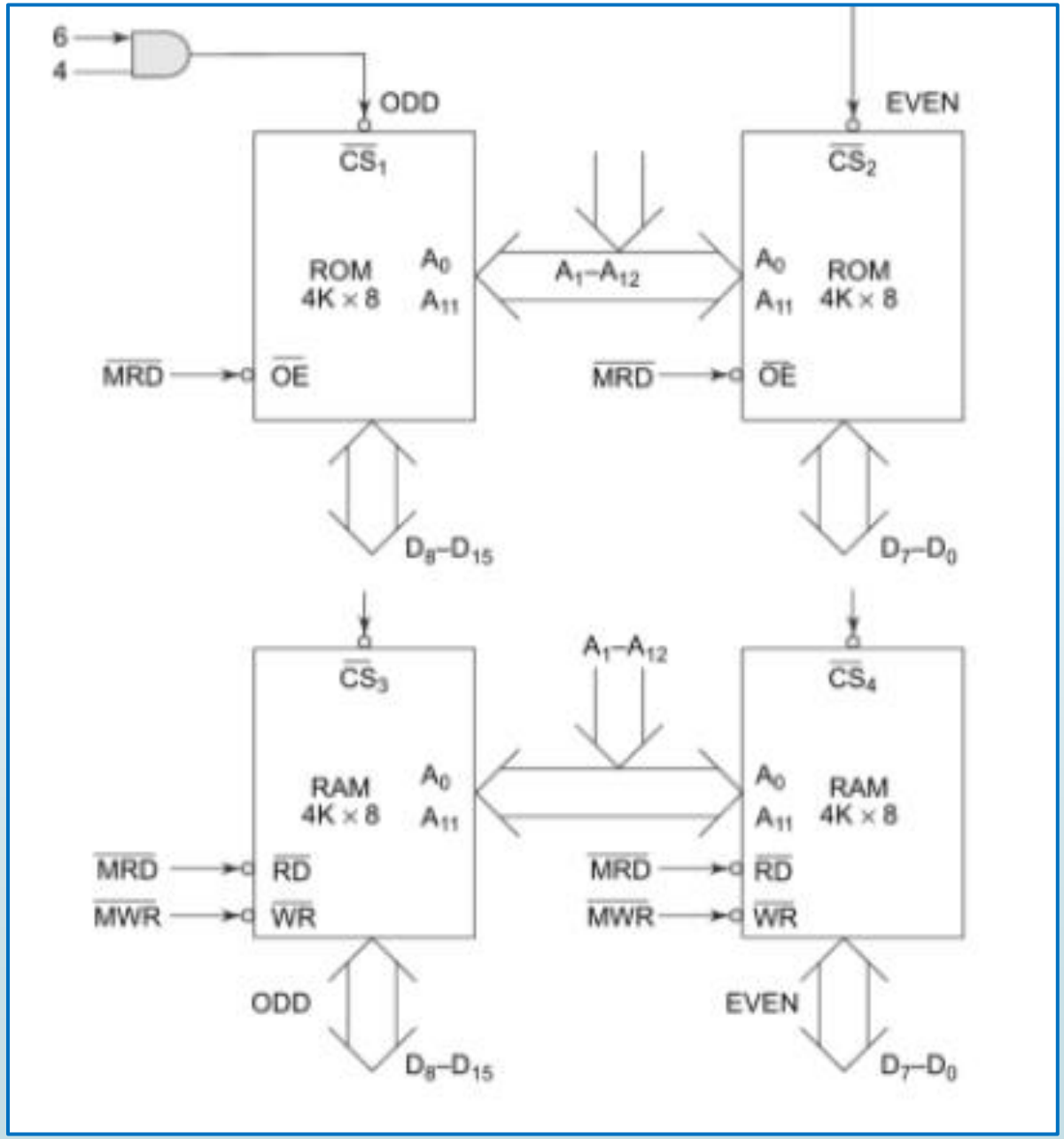
Address Map

Decoder I/P → Address/ $\overline{BHE}$ →	A_2 A_{13}	A_1 A_0	A_0 $\overline{BHE}$	Selection / Comments
Word transfer on $D_0 - D_{15}$	0	0	0	Even and odd addresses in RAM
Byte transfer on $D_7 - D_0$	0	0	1	Only even address in RAM
Byte transfer on $D_8 - D_{15}$	0	1	0	Only odd address in RAM
Word transfer on $D_0 - D_{15}$	1	0	0	Even and odd addresses in ROM
Byte transfer on $D_7 - D_0$	1	0	1	Only even address in ROM
Byte transfer on $D_8 - D_{15}$	1	1	0	Only odd address in ROM

Diagram


BHE
A0
A13
A0
A1
A2
3 : 8
DECODER
74138
O0
O1
O2
O3
O4
O5
O6
O7
0
1
2
3
4
5
6
7
E1
E2
E3
7430
A19
A15
A14


2
0
CS3
1
0
CS4


6
4
ODD
EVEN
CS1
CS2
ROM
4K × 8
A0
A11
A1–A12
MRD
OE
D8–D15
D7–D0
CS3
CS4
RAM
4K × 8
RD
WR
MWR
ODD
EVEN

Dynamic RAM

- Whenever a large capacity memory is required in a microcomputer system, the memory subsystem is generally designed using dynamic RAM (DRAM).
- Advantages of DRAM over Static RAM
  1. Higher packing density,
  2. lower cost and
  3. less power consumption
- A static RAM cell has 6 transistors while the dynamic RAM cell has 1 transistor and a capacitor. This leads to higher packaging density and low cost per unit.
- The function of capacitor is carried out by diode that operates in a reverse bias mode.

Dynamic RAM (Contd...)

- The reverse-biased diode has leakage current that tends to discharge the capacitor giving rise to the possibility of data loss. To avoid this, the stored data in the cell is refreshed in regular intervals. The process is called as **Refresh cycle**.
- The refresh activity is similar to reading the data from each and every cell of memory, independent of the requirement of microprocessor. During this refresh period all other operations related to the memory subsystem are suspended. Hence the refresh activity causes **loss of time**, resulting in **reduce system performance**.
- However keeping in view the advantages of dynamic RAM, like low power consumption, high packaging density and low cost, most of the advanced computing system are designed using dynamic RAM, at the cost of operating speed.
- A dedicated hardware chip called as **dynamic RAM controller** is the most important part of the interfacing circuit.

Refresh Cycle

- The Refresh cycle is different from the memory read cycle in the following aspects.

1. The memory address is not provided by the CPU address bus, rather it is generated by a refresh mechanism counter called as **refresh counter**.
2. Unlike memory read cycle, **more than one memory chip** may be enabled at a time so as to reduce the number of total memory refresh cycles.
3. The **data enable** control of the selected memory chip is deactivated, and data is not allowed to appear on the system data bus during refresh, as more than one memory units are refreshed simultaneously. This is to avoid the data from the different chips to appear on the bus simultaneously.
4. Memory read is either a **processor initiated** or an external bus master initiated and carried out by the refresh mechanism.

Refresh Cycle (2)

- Dynamic RAM is available in units of several **kilobits to megabits** of memory. This memory is arranged internally in a 2 – D matrix so that it will have *n* rows and *m* columns. The row address *n* and column address *m* are important for the **refreshing** operation.
- A 4 K bit dynamic RAM chip is internally arranged bit array of dimension 64 * 64 , i.e. 64 rows and 64 columns. The row address and column address will require 6 bits each (since $2^6 = 64$). These 6 bits for each row address and column address are generated by the refresh counter, during the refresh cycles.
- A complete row of 64 cells is refreshed at a time to minimize the refreshing time. Thus the refresh counter generate row addresses only. The row address are multiplexed, over lower order address lines.

Refresh Cycle (3)

- During refresh cycle is in process the refresh counter puts the row address over the address bus.
- During normal processor-initiated activities the address bus of the processor is connected to the address bus of DRAM.
- A timer, called refresh timer, derives a pulse for refreshing action after each refresh interval.
- **Refresh interval** can be qualitatively defined as the time for which a dynamic RAM cell can hold data charge level practically constant, i.e. no data loss takes place.
- If a dynamic RAM chip has 64 rows, then all the 64 rows are to refreshed in a single refresh interval.

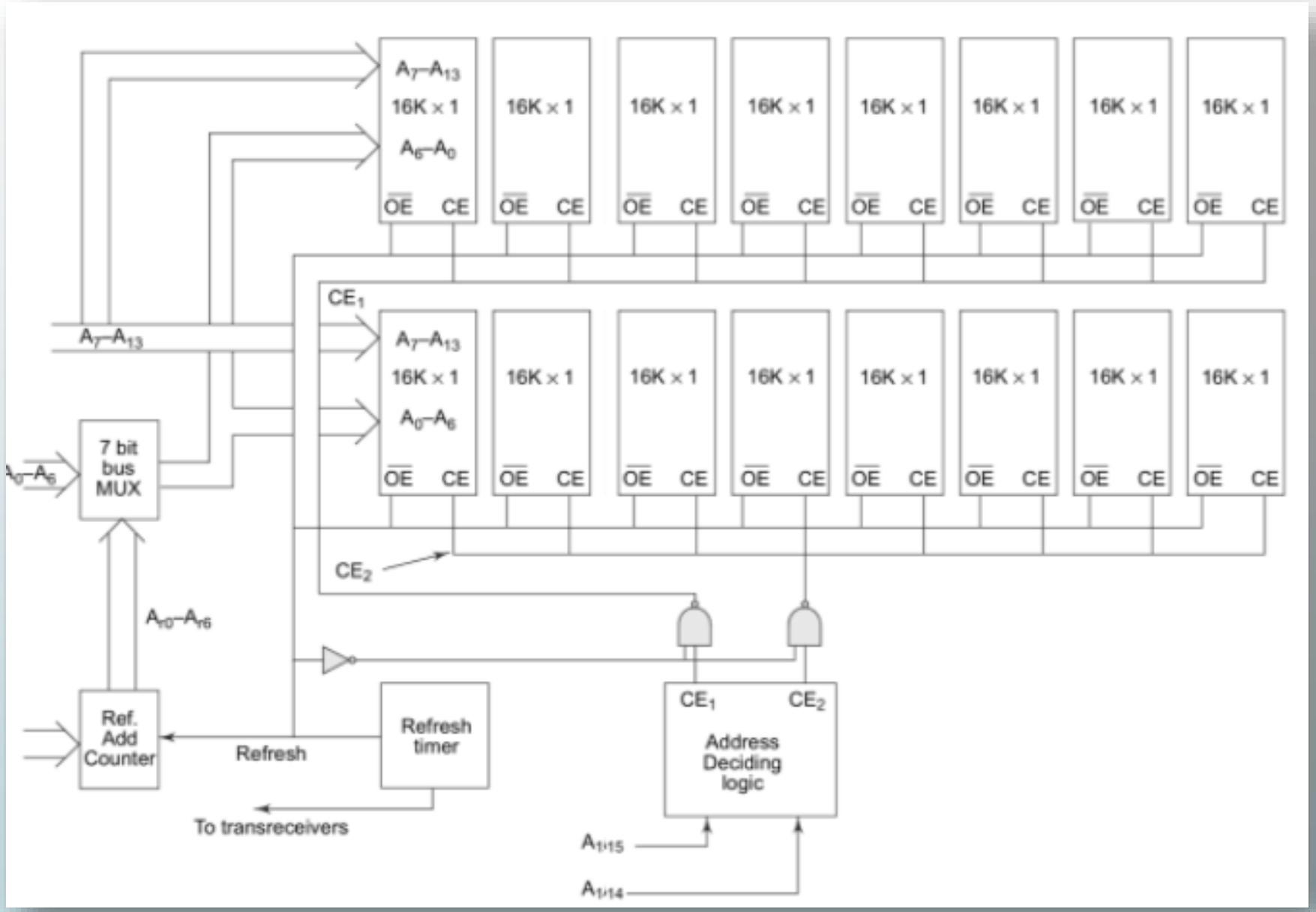
Refresh Cycle (4)

- This refresh interval depends upon the manufacturing technology of the dynamic RAM cell.
- It may range anywhere from 1 msec to 3 msec.
- **Example:** Find the frequency of refresh pulses for 4K DRAM arranged in 64 columns x 64 rows. Assume refresh time interval as 2 msec.

Refresh Time (per row) $t_r = \frac{2\ x\ 10^{-3}}{64}$ .... (1)

Refresh Frequency $f_r = \frac{64}{2\ x\ 10^{-3}}$ ... (2)

DRAM



Dynamic RAM refresh Logic

Refresh Cycle (5)

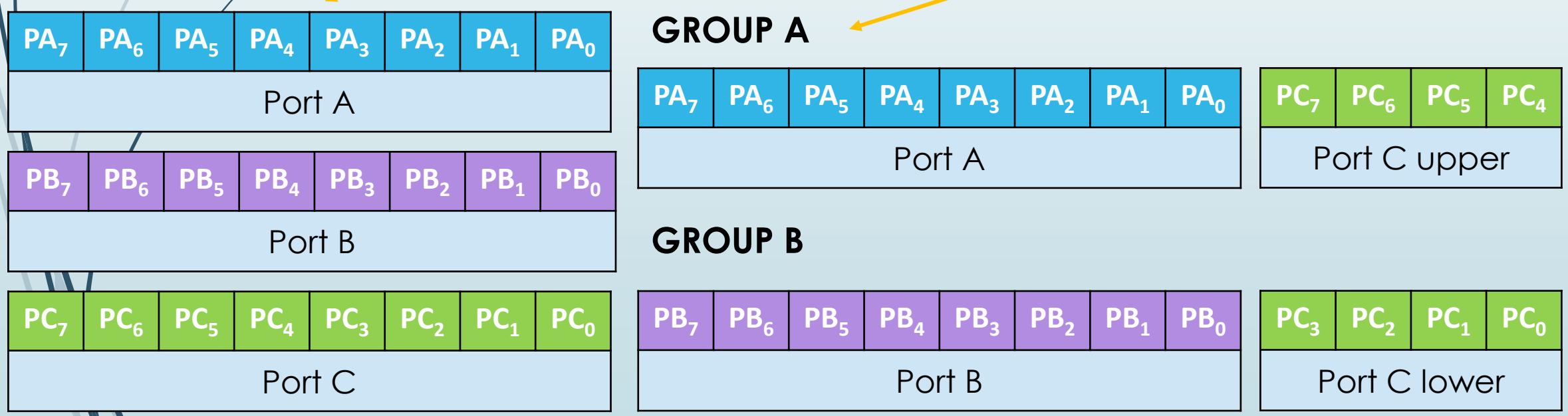
- Each chip is of 16K * 1-bit dynamic RAM cell array. The system contains two 16K byte dynamic RAM units. All the address and data lines are assumed to be available from an 8086-microprocessor system.
- The $\overline{OE}$ pin controls output data buffer of the memory chips. The Chip Enable (CE) pins are active high chip select signals.
- When the refresh cycle starts, $\overline{OE}$ and CE pins tend to go high. The high CE enables the memory chip for refreshing, while high OE prevents the data from appearing on the data bus.
- The 16K * 1-bit dynamic RAM has an internal array of 128*128 cells, requiring 7 bits for row address. The lower order seven lines A_0-A_6 are multiplexed with the refresh counter output A_{10}-A_{16}.

S – 2

Programmable Peripheral Interface 8255 & Interfacing 8255 with 8086 and programming

8255 – Ports

- The Parallel Input-Output Port chip 8255 is also called as Programmable Peripheral Input-Output Port. The Intel's 8255 is designed for use with Intel's 8-bit, 16-bit and higher capability microprocessors. It has 24 input/output lines which may be individually programmed in two groups of 12 lines each, OR or three groups of eight lines.



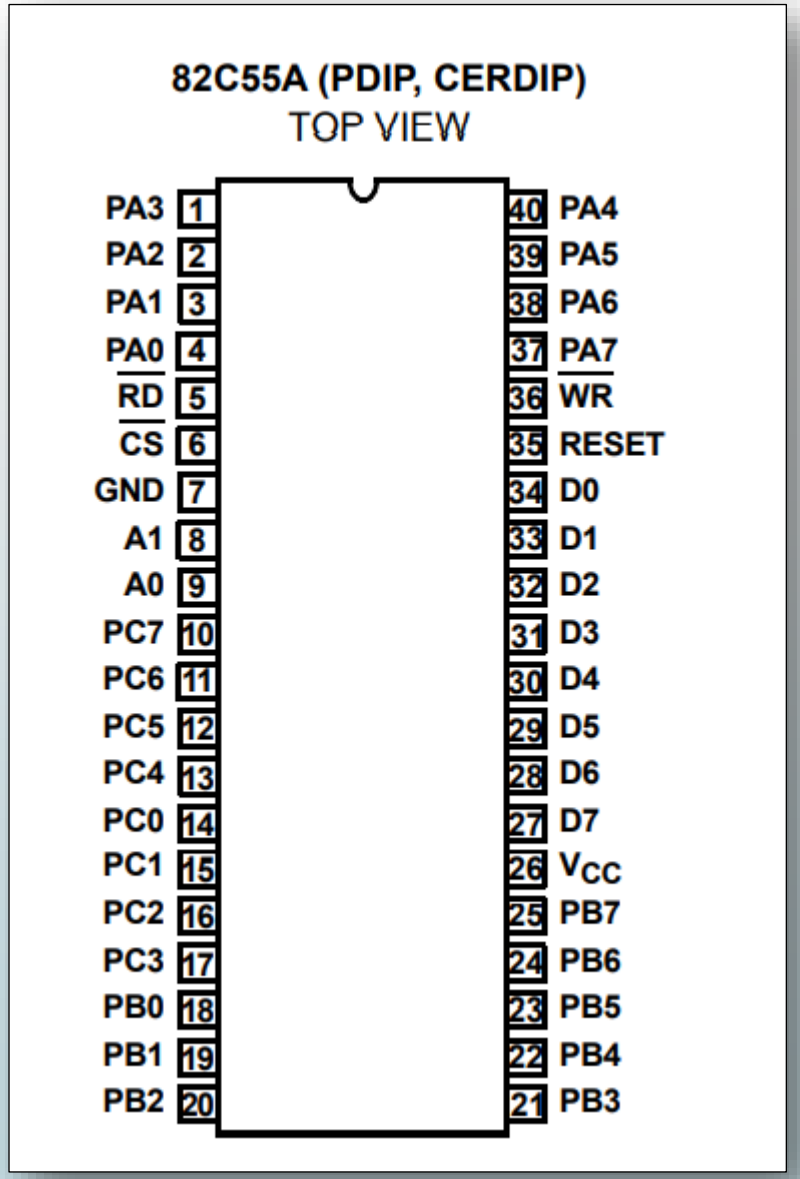
8255 – Ports (2)

- The two groups of I/O pins are named as Group A and Group B. Each of these two groups contains a subgroup of eight I/O lines called as 8-bit port and another subgroup of four lines or a 4-bit port. Thus, Group A contains an 8-bit port A along with a 4-bit port C upper.
- The port A lines are identified by symbols PA_0-PA_7 while the port C upper lines are identified as PC_7-PC_4. Similarly, Group B contains an 8-bit port B, containing lines PB_0-PB_7 and a 4-bit port C with lower bits PC_0- PC_3. The port C upper and port C lower can be used in combination as an 8-bit port C. Upper and lower port C are assigned the same address.
- Either Three 8- bit I/O ports or two 8-bit and two 4-bit C ports from 8255 are possible. All of these ports can function independently either as input or as output ports. This can be achieved by programming the bits of an internal register of 8255 called as Control Word Register (CWR).

8255 – Ports (3)

- The 8-bit data bus buffer is controlled by the read/write control logic. The read/write control logic manages all of the internal and external transfers of both data and control words.
- $\overline{RD}$, $\overline{WR}$, A_1, A_0 and RESET are the inputs provided by the microprocessor to the READ/ WRITE control logic of 8255. The 8-bit, 3-state bidirectional buffer is used to interface the 8255 internal data bus with the external system data bus.
- This buffer receives or transmits data upon the execution of input or output instructions by the microprocessor. The control words or status information is also transferred through the buffer.

Pin Diagram



Pin Diagram of 8255

Signal Description

- PA_7 – PA_0 : These are eight port A lines that acts as either latched output or buffered input lines depending upon the control word loaded into the CWR.
- PC_7 – PC_4 : Upper nibble of port C lines. They may act as either output latches or input buffers lines. This port also can be used for generation of handshake lines in mode 1 or mode 2.
- PC_3 – PC_0 : These are the lower port C lines; other details are the same as PC_7 – PC_4 lines.
- PB_7 – PB_4 : These are the eight port B lines which are used as latched output lines or buffered input lines in the same way as port A.

Signal Description (2)

- $\overline{RD}$: This is the input line driven by the microprocessor and should be low to indicate read operation to 8255.
- $\overline{WR}$: This is an input line driven by the microprocessor. A low on this line indicates write operation.
- $\overline{CS}$: This is a chip select line. If this line goes low, it enables the 8255 to respond to RD and WR signals, otherwise RD and WR signal are neglected.
- A_1-A_0: These are the address input lines and are driven by the µP. These lines A1-A0 with $\overline{RD}$, $\overline{WR}$ and $\overline{CS}$ from the following operations for 8255.
  - These address lines (A_1-A_0) are used for addressing any one of the four registers, i.e. three ports and a Control Word Register (CWR).

8255 – Addressing Ports

A1	A0	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	INPUT OPERATION (READ)
0	0	0	1	0	Port A → Data Bus
0	1	0	1	0	Port B → Data Bus
1	0	0	1	0	Port C → Data Bus
1	1	0	1	0	Control Word → Data Bus
					OUTPUT OPERATION (WRITE)
0	0	1	0	0	Data Bus → Port A
0	1	1	0	0	Data Bus → Port B
1	0	1	0	0	Data Bus → Port C
1	1	1	0	0	Data Bus → Control
					DISABLE FUNCTION
X	X	X	X	1	Data Bus → Three-State
X	X	1	1	0	Data Bus → Three-State

Image Courtesy : 8255 data sheet

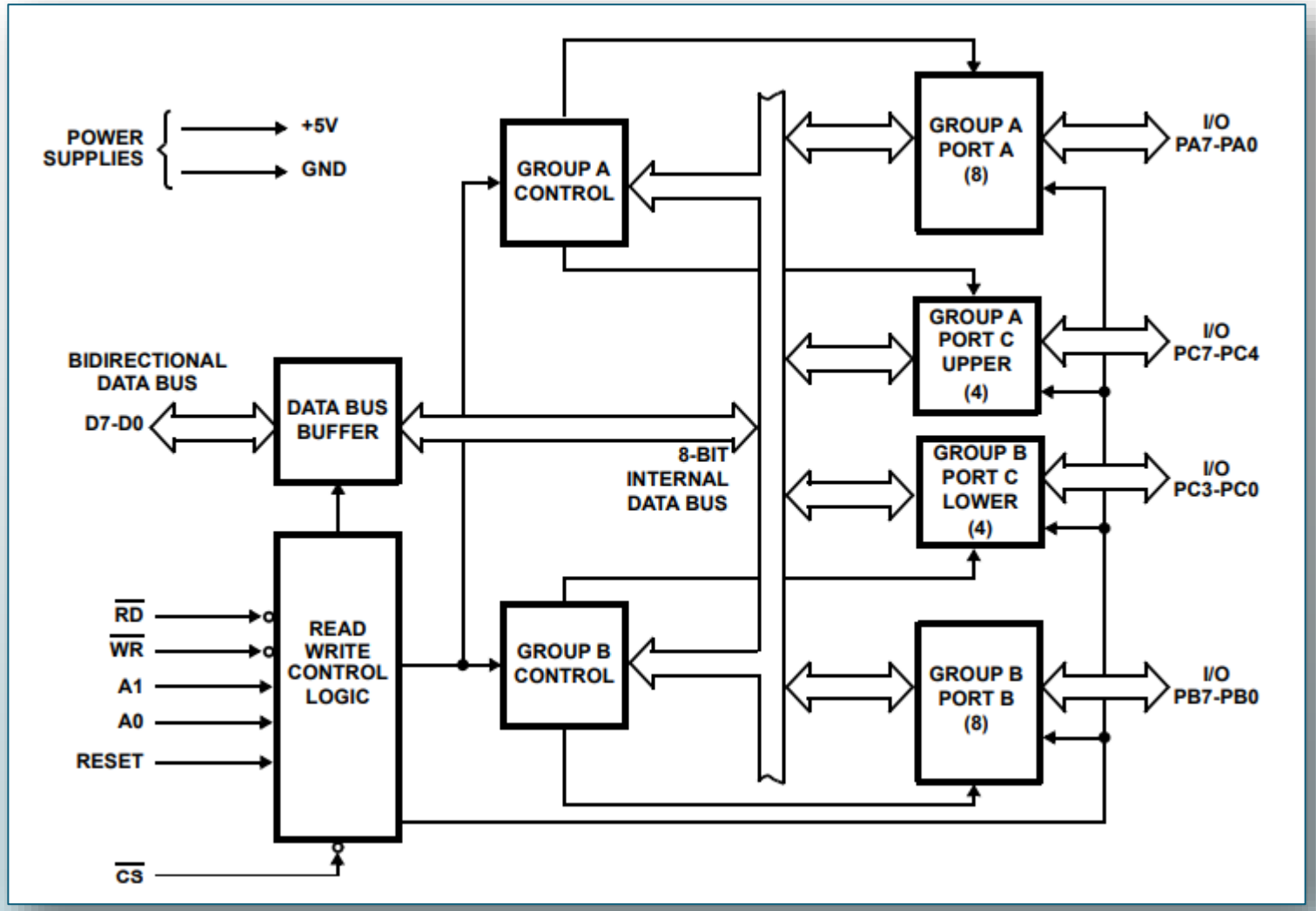
Signal Description (3)

- In case of 8086 systems, if the 8255 is to be interfaced with lower order data bus, the A_0 and A_1 pins of 8255 are connected with A_1 and A_2 respectively.
- D_0-D_7: These are the data bus lines those carry data or control word to/from the microprocessor.
- RESET: A logic high on this line clears the control word register of 8255. All ports are set as input ports by default after reset.
- It has a 40 pins of 4 groups.

8255 – Internal Architecture

SECTIONS

1. Data bus
2. Read Write control logic
3. Group A and Group B controls
4. Port A, B and C



8255 Internal Architecture

Architecture – Explanation

1. **Data bus buffer**: This is a tristate bidirectional buffer used to interface the 8255 to system data bus. Data is transmitted or received by the buffer on execution of input or output instruction by the CPU.
   - Control word and status information are also transferred through this unit.
2. **Read/Write control logic**: This unit accepts control signals ($\overline{RD}$, $\overline{WR}$) and also inputs from address bus and issues commands to individual group of control blocks (Group A, Group B).
3. **Group A and Group B controls**: These block receive control from the CPU and issues commands to their respective ports.
   - Group A - PA and PCU (PC_7 –PC_4) Group B - PCL (PC_3 – PC_0)
   - CWR can only be written into no read operation of the CWR is allowed.

Architecture – Explanation (2)

4. **Port A, B and C**
   - Port A: This has an 8 bit latched/buffered O/P and 8 bit input latch. It can be programmed in 3 modes – mode 0, mode 1, mode 2
   - Port B: This has an 8 bit latched / buffered O/P and 8 bit input latch. It can be programmed in mode 0, mode 1.
   - Port C : This has an 8 bit latched input buffer and 8 bit output latched/buffer. This port can be divided into two 4 bit ports and can be used as control signals for port A and port B. it can be programmed in mode 0.

BSR and IO Mode

- These are two basic modes of operation of 8255.
- I/O mode and Bit Set-Reset (BSR) mode.
- In BSR mode only port C (PC_0-PC_7) can be used to set or reset its individual port bits. In I/O mode, the 8255 ports work as **programmable I/O ports.** Under the I/O mode of operation, mode 0, mode 1 and mode 2 is possible
- **BSR Mode**: In this mode any of the 8-bits of port C can be set or reset depending on D0 of the control word. The bit to be set or reset is selected by bit select flags D3, D2 and D1 of the CWR.

BSR Mode

Port C bits can be SET or RESET

Example

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	0	0	0	1	1	0	1

Port C: 6^{th} bit is SET

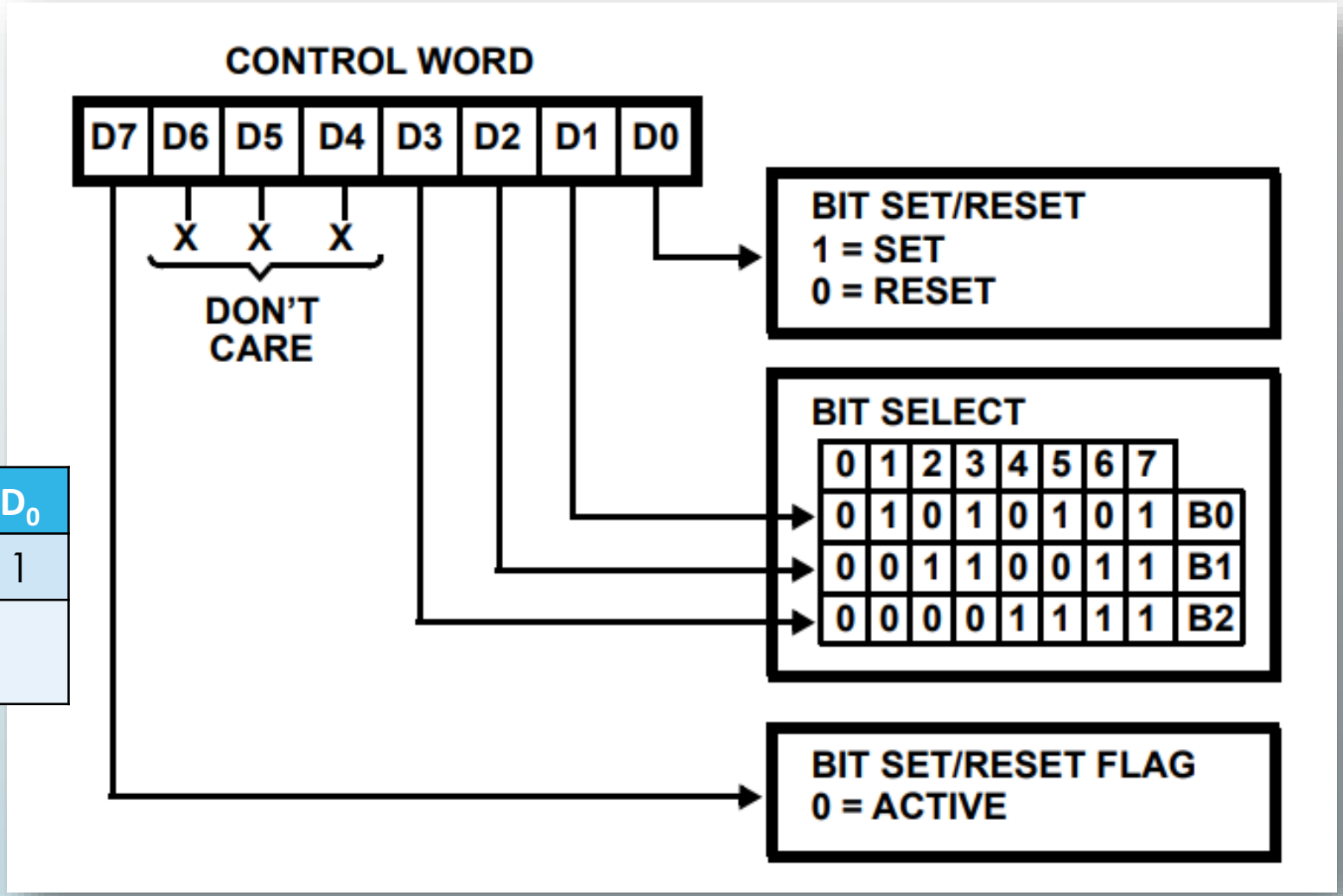


Image Courtesy: 8255 Data sheet

I/O Mode CWR

Example

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	0	0	1	1	0	0	0
IO mode	Mode 0		Port A input	Port C (up) input	Mode 0	Port B output	Port C (low) output

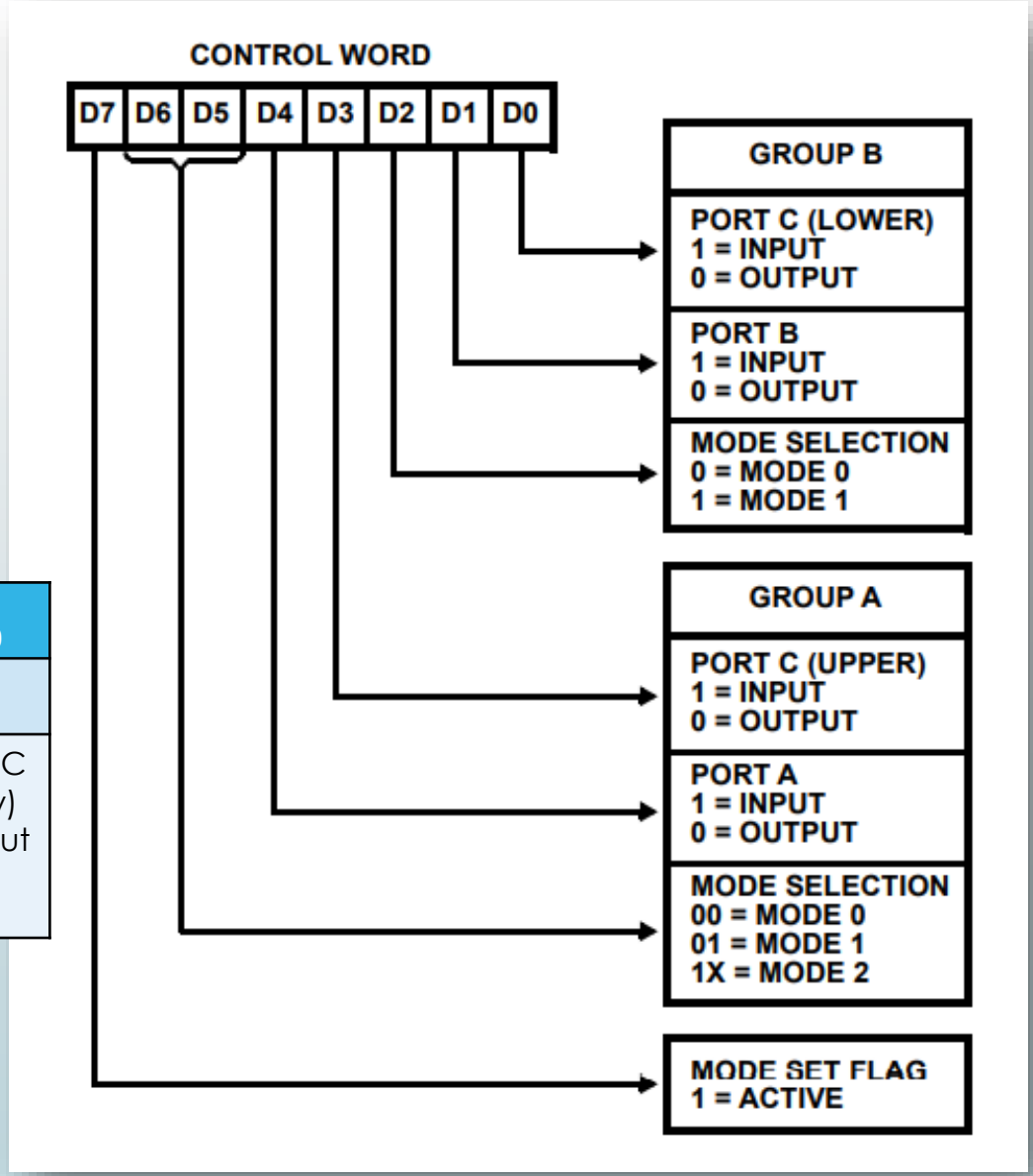


Image Courtesy: 8255 Data sheet

Mode 0 (Basic I/O mode)

- This mode is also called as basic Input/Output mode. This mode provides simple input and output capabilities using each of the three ports. Data can be simply read from and written to the input and output ports respectively, after appropriate initialisation.

- **The salient features**

  1. Two 8-bit ports (port A and port B)and two 4-bit ports (port C upper and lower) are available. The two 4-bit ports can be combinedly used as a third 8-bit port.
  2. Any port can be used as an input or output port.
  3. Output ports are latched. Input ports are not latched.
  4. A maximum of 4 ports are available. so, 16 I/O configuration are possible.

Mode 1 (Strobed input/output mode)

- In this mode the **handshaking** control the input and output action of the specified port. Port C lines PC_0-PC_2, provide strobe or handshake lines for port B. This group which includes port B and PC_0-PC_2 is called as group B for Strobed data input/output. Port C lines PC_3-PC_5 provide strobe lines for port A. This group including port A and PC_3-PC_5 from group A. Thus, port C is utilized for generating handshake signals.

- The salient features of mode 1 are listed as follows:

  1. Two groups – group A and group B are available for strobed data transfer.
  2. Each group contains one 8-bit data I/O port and one 4-bit control/data port.
  3. The 8-bit data port can be either used as input and output port. The inputs and outputs both are latched.
  4. Out of 8-bit port C, PC_0-PC_2 are used to generate control signals for port B and PC_3-PC_5 are used to generate control signals for port A. the lines PC_6, PC_7 may be used as independent data lines.

Mode 2 (Strobed bidirectional I/O)

- This mode of operation provides 8255 with an additional features for communicating with a peripheral device on an 8-bit data bus. Handshaking signals are provided to maintain proper data flow and synchronization between the data transmitter and receiver. The interrupt generation and other functions are similar to mode 1

- In this mode, 8255 is a bidirectional 8-bit port with handshake signals. The $\overline{RD}$ and $\overline{WR}$ signals decide whether the 8255 is going to operate as an input port or output port.

- The Salient features of Mode 2 of 8255 are listed as follows:
  1. The single 8-bit port in group A is available.
  2. The 8-bit port is bidirectional and additionally a 5-bit control port is available.

S – 3

Interfacing ADC with 8086 and programming & Interfacing DAC with 8086 and programming

Analog-to-Digital Converter (ADC)

- µP considers ADC as a input device. So, it sends Start of Conversation (SoC) signal (pulse of a specific duration) to ADC.
- Analog to digital conversion is a **slow** process. So, ADC will send a End of Conversion (EoC) signal to µP, when the conversion is over. Then µP can take the result from the buffer. EoC signal can trigger a interrupt in 8086 or µP can poll the EoC signal.
- Usually µP will use the IO ports of 8255 to interact with ADC. Like reading EoC signal from ADC and reading digital output of ADC.

Analog-to-Digital Converter (2)

- The time taken by the ADC from the active edge of SoC pulse till the active edge of EoC signal is called as the **conversion delay** of the ADC.
- Conversion delay: few **µs** (fast ADC) to few 100 **ms** (slow ADC).
- Most popular conversion techniques in integrated ADC chips
  - Successive Approximation Techniques
  - Dual Slope Integration Techniques

ADC Interfacing – General Algorithm

1. Ensure the stability of analog input, applied to the ADC.
2. Issue SoC pulse to ADC
3. Read EoC signal to mark the end of conversion processes.
4. Read digital data output of the ADC as equivalent digital output.
5. Analog input voltage must be constant at the input of the ADC right from the start of conversion till the end of the conversion to get correct results.

   This may be ensured by a **sample and hold circuit (S-H)** which samples the analog signal and holds it constant for a specific time duration. µP issue a hold signal to the S-H circuit.
6. If the applied input changes before the complete conversion process is over, the digital equivalent of the analog input calculated by the ADC may not be correct.

ADC 808 and 809 Characteristics

- The ADC chips 0808 and 0809 are 8-bit CMOS, successive approximation converters. The conversion delay is 100 µs at a clock frequency of 640 KHz.
- No need of external zero or full scale adjustments. They are inbuilt in the chip.
- These converters internally have a 3:8 analog multiplexer so that at a time 8 different analog conversion can be carried out. Use address lines - ADD A, ADD B, ADD C.
- In multichannel applications – The CPU drive the lines using output port. In single channel applications – the lines are hardwired to select the input.
- There are **unipolar** analog to digital converters, i.e. they are able to convert only positive analog input voltage to their digital equivalent. These chips do no contain any internal sample and hold circuit.

Table

Analog I/P selected	Address lines		
	C	B	A
I /P 0	0	0	0
I /P 1	0	0	1
I /P 2	0	1	0
I /P 3	0	1	1
I /P 4	1	0	0
I /P 5	1	0	1
I /P 6	1	1	0

Address mapping

NOTE

- The signal I/P_0 is referred as IN0 in Pin Diagram
- The signals O_0, O_2, … O_7 are referred as 2^{-8}, 2^{-7} … 2^{-1} in Pin Diagram

Bl
Di

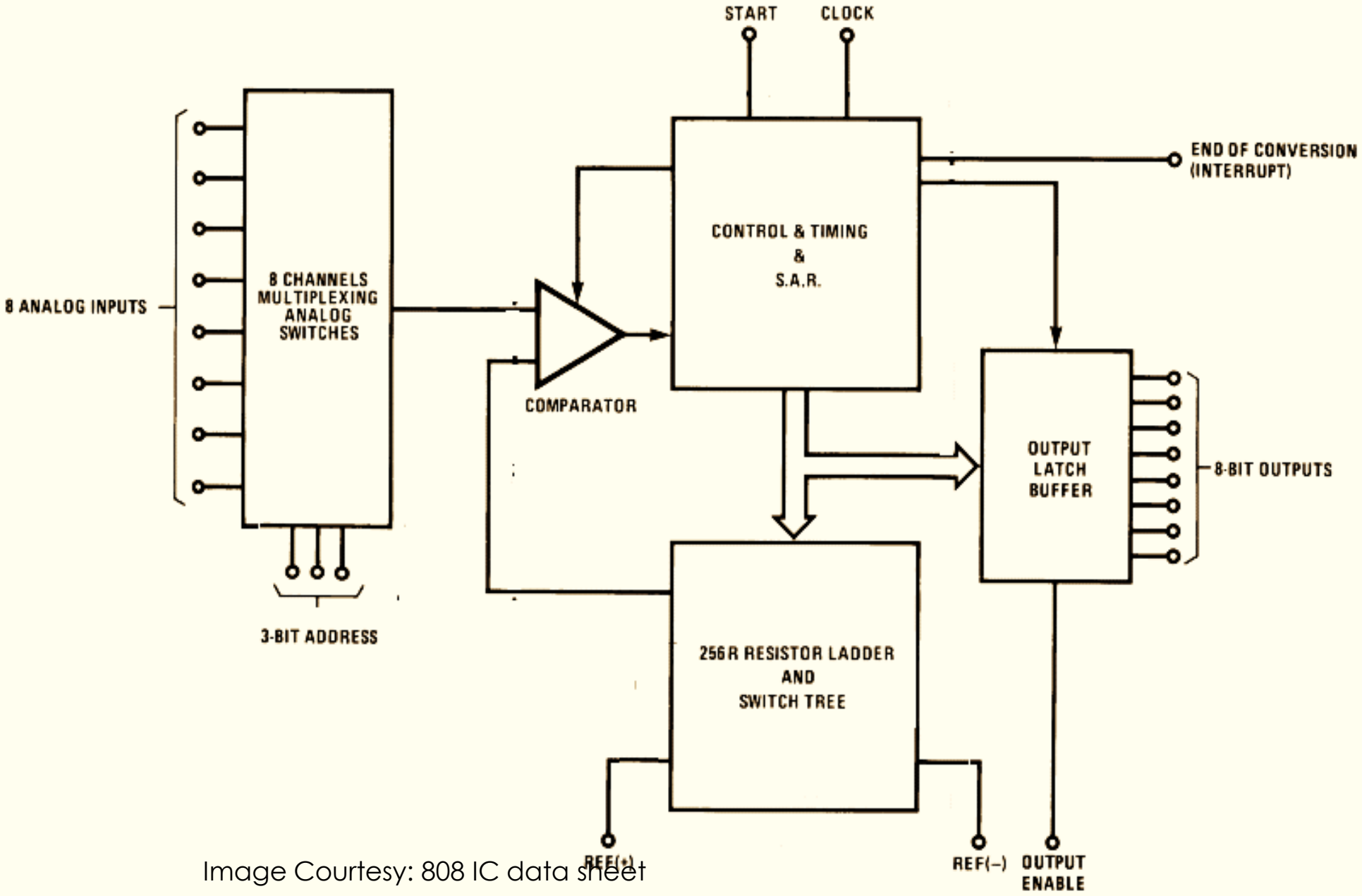


Image Courtesy: 808 IC data sheet

Pin Diagram

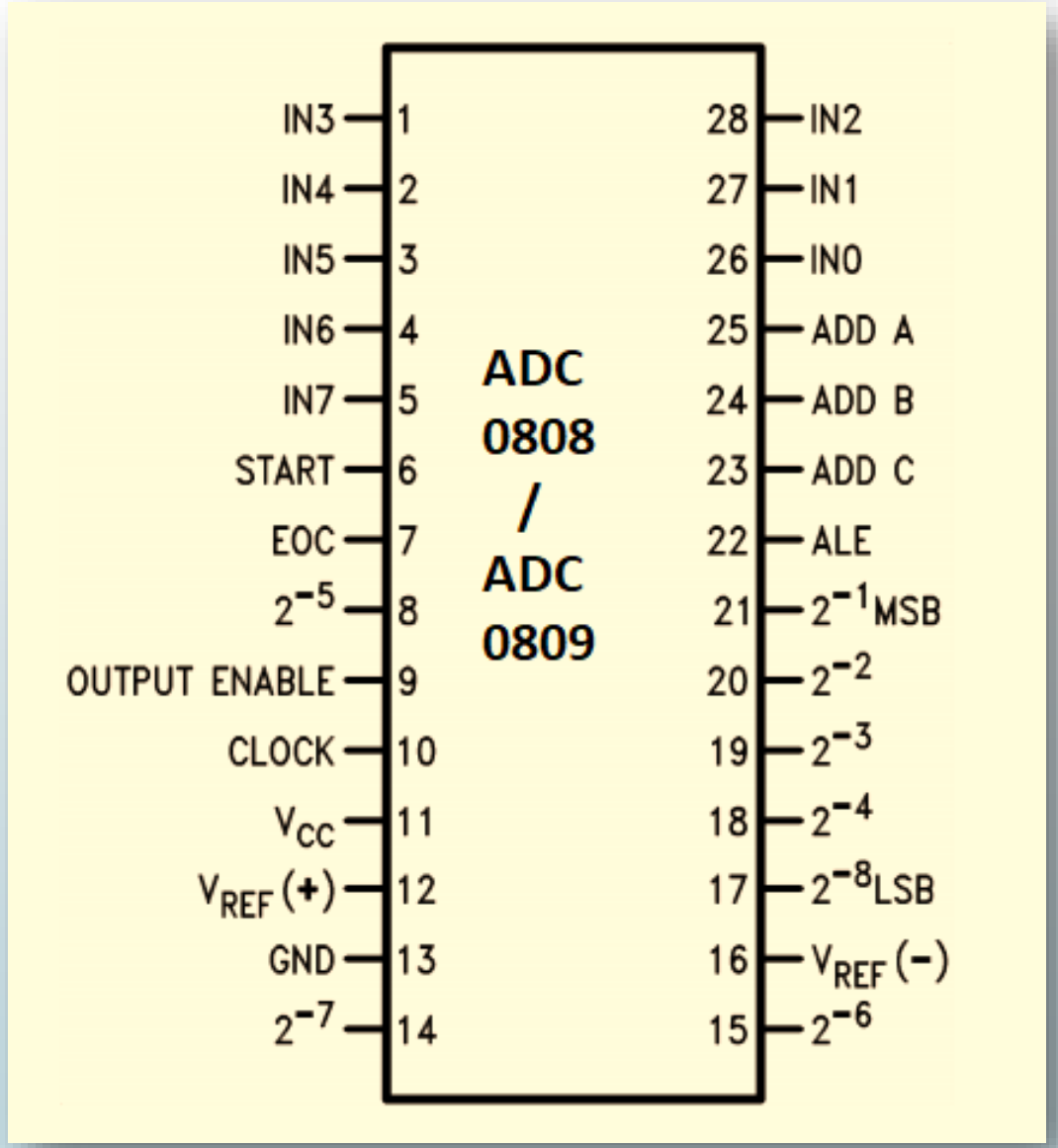


Image Courtesy: ADC 808 Data sheet

Example Interfacing Problem

- **Example:** Interfacing ADC 0808 with 8086 using 8255 ports. Use port A of 8255 for transferring digital data output of ADC to the CPU and port C for control signals. Assume that an analog input is present at I/P2 of the ADC and a clock input of suitable frequency is available for ADC.
- **Solution:** The analog input I/P2 is used and therefore address pins A,B,C should be 0,1,0 respectively to select I/P2. The OE and ALE pins are already kept at +5V to select the ADC and enable the outputs.
- Port C upper receives the EoC signal. Port C lower sends (output) SoC to the ADC.
- Port A acts as a 8-bit input data port to receive the digital data output from the ADC.

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	0	0	1	1	0	0	0
IO mode	Mode 0		Port A input	Port C (up) input	Mode 0	Port B output	Port C (low) output

CWR is holding a value 98h

Interfacing Diagram

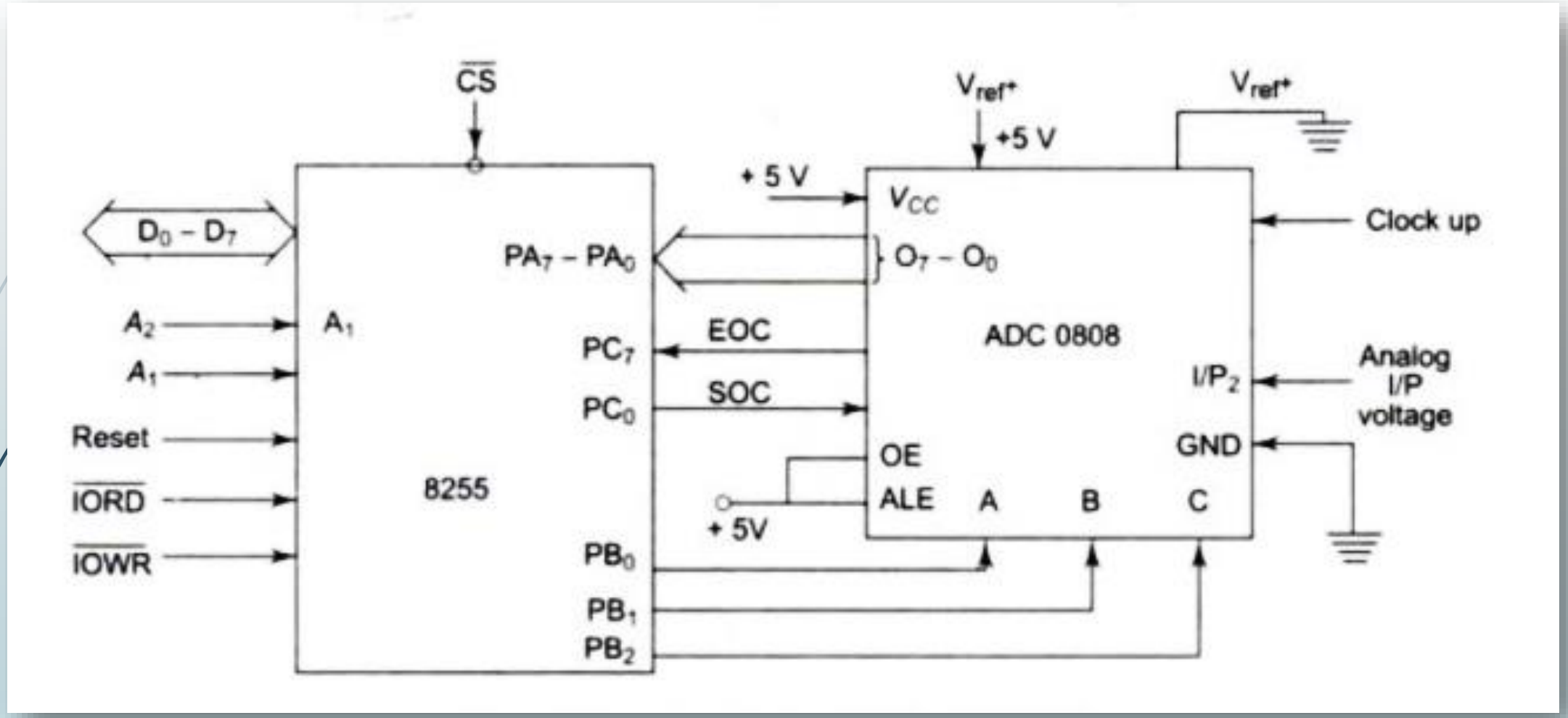


Image Courtesy: A. K. Ray Book

ALP to interface 8086 and 808

```
        MOV AL, 98h      ; initialise 8255 as OUT
        OUT CWR, AL      ; discussed above.
        MOV AL, 02h      ; Select I/P2 as analog  input
        OUT PortB, AL    ; PortB – Port address
        MOV AL, 00h      ; Give SoC pulse to ADC
        OUT PortC, AL    ; PortC – Port address
        MOV AL, 01h
        OUT PortC, AL
        MOV AL, 00h
        OUT PortC, AL
WAIT:   IN AL, PortC     ; Check for EoC by
        RCR              ; reading port C upper and
        JNC WAIT         ; rotating through carry.
        IN AL, PortA     ; If EoC, read digital equivalent in AL
        HLT              ; Stop.
```

Digital Analog Converters

- The digital to analog converters convert binary number into their equivalent voltages (analog).
- The DAC find applications in areas like digitally controlled gains, motors speed controls, programmable gain amplifiers etc. AD 7523
- 8-bit Multiplying DAC : This is a 16 pin DIP, multiplying digital to analog converter, containing R-2R ladder for D-A conversion along with single pole double thrown NMOS switches to connect the digital inputs to the ladder.

Digital Analog Converters (2)

- The supply range is from +5V to +15V, while V_{ref} may be anywhere between -10 V to +10 V. The maximum analog output voltage will be anywhere between -10 V to +10 V, when all the digital inputs are at logic high state.
- Usually a Zener is connected between OUT1 and OUT2 to save the DAC from negative transients. An operational amplifier is used as a current to voltage converter at the output of AD to convert the current output of AD to a proportional output voltage.
- It also offers additional drive capability to the DAC output.
- An external feedback resistor acts to control the gain. No need to connect any external feedback resistor, if no gain control is required.

Pin Diagram

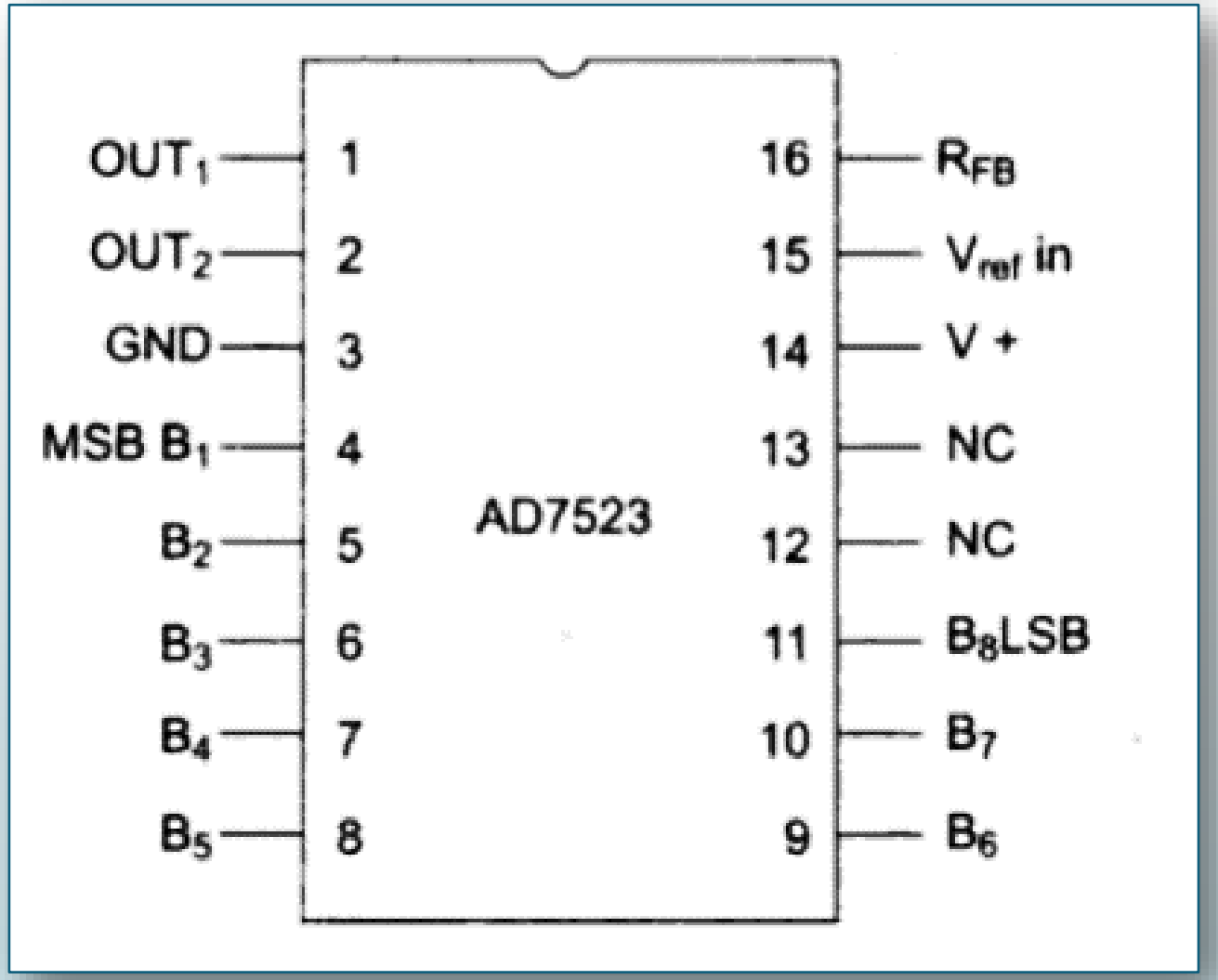
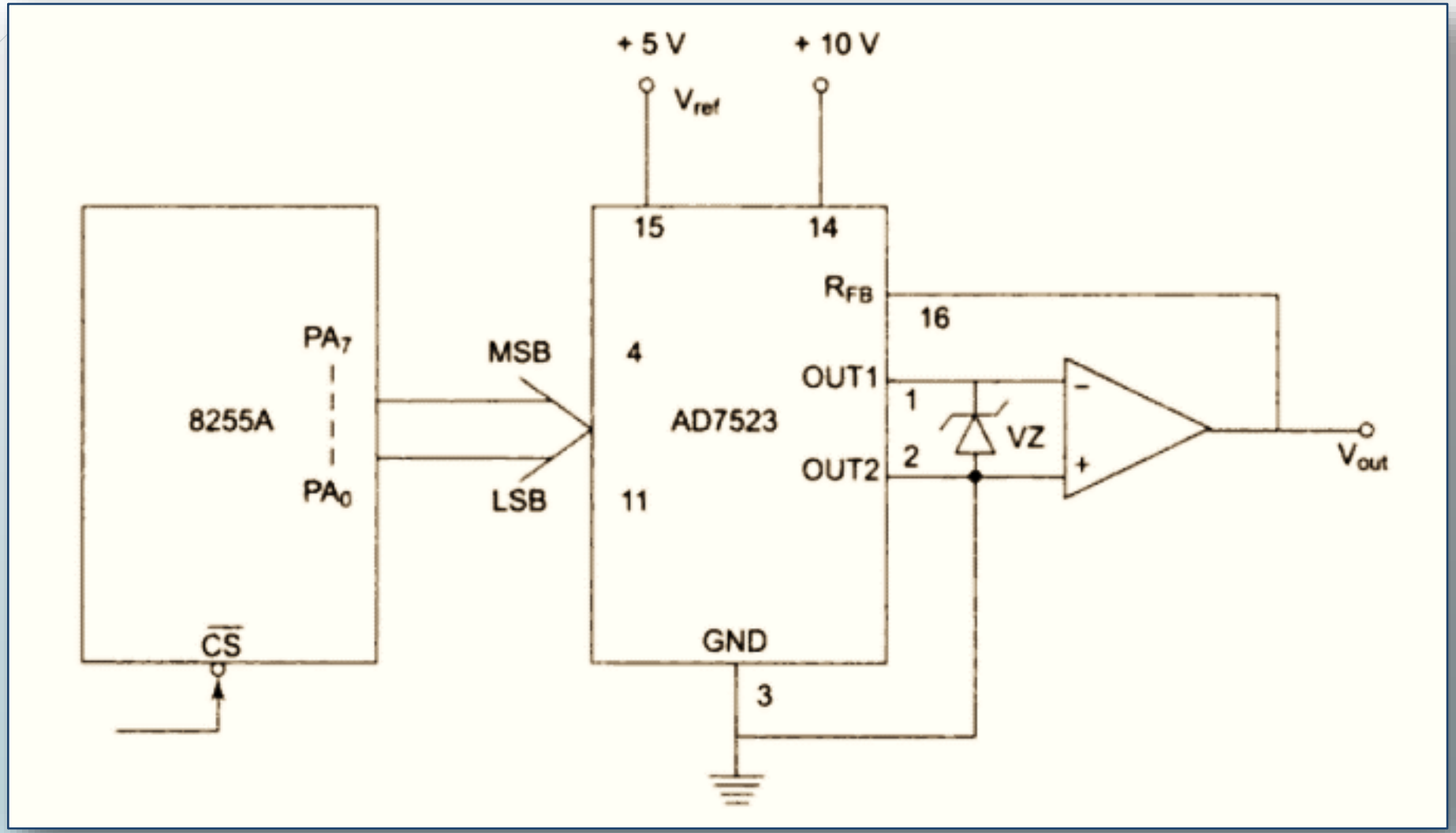


Image Courtesy: A.K. Ray book

8086 – AD7253 Interface Diagram



Code

- EXAMPLE: Interfacing DAC AD7523 with an 8086 CPU running at 8 MHz and write an ALP to generate a sawtooth waveform of period 1 ms with V_{max} 5V.

```
ASSUME CS: CODE
CODE SEGMENT
START :   MOV AL ,80h          ;make all ports output
          OUT CWR, AL
AGAIN :   MOV AL,00h           ;start voltage for ramp
BACK :    OUT PA, AL
          INC AL
          CMP AL, 0FFh
          JB BACK
          JMP AGAIN
     CODE ENDS
     END START
```

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	0	0	0	0	0	0	0
IO mode	Mode 0		Port A output	Port C (up) output	Mode 0	Port B output	Port C (low) output

CWR is holding a value 80h

ALP - Explanation

- Port A is initialized as the output port for sending the digital data as input to DAC. The ramp starts from the 0 V (analog), hence AL starts with 00H.
- To increment the ramp, the content of AL is increased during each execution of loop till it reaches F2H.
- After that the saw tooth wave again starts from 00H, i.e. 0 V (analog) and the procedure is repeated.
- The ramp period given by this program is precisely 1.000625 msec.
- Here the count F2H has been calculated by dividing the required delay of 1ms by the time required for the execution of the loop once. The ramp slope can be controlled by calling a controllable delay after the OUT instruction.

Learning Resource

[1] K. M. Bhurchandi and A. K. Ray, "Advanced Microprocessors and Peripherals – with ARM and an Introduction to Microcontrollers and Interfacing", Tata McGraw Hill, 3rd ed., 2015.

Note: Almost all figures and text content taken from the above stated book.

Thank You