MADE EASY
Leading Institute for ESE, GATE & PSUs

Detailed Solutions

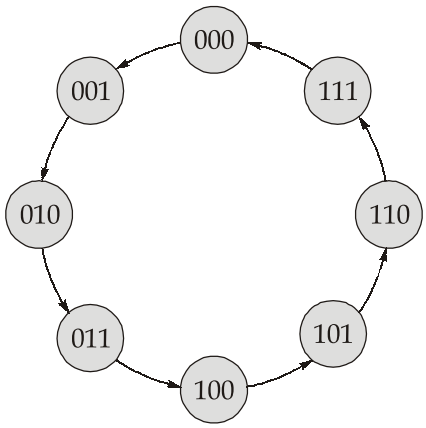
ESE-2025 Mains Test Series

E & T Engineering Test No : 2

Section A : Digital Circuits + Signals and Systems + Microprocessors & Microcontroller

Q.1 (a) Solution:

The state diagram for the 3-bit binary counter.



The state table for 3-bit binary counter,

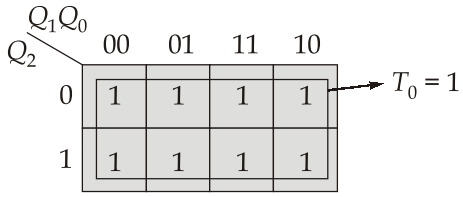
Present state			Next state		
Q_2	Q_1	Q_0	Q_2^+	Q_1^+	Q_0^+
0	0	0	0	0	1
0	0	1	0	1	0
0	1	0	0	1	1
0	1	1	1	0	0
1	0	0	1	0	1
1	0	1	1	1	0
1	1	0	1	1	1
1	1	1	0	0	0

The excitation table:

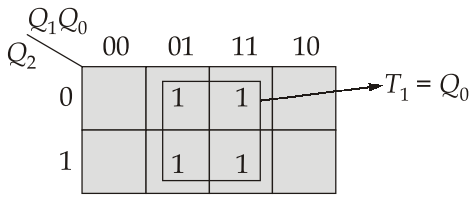
Output state transitions								
Present state			Next state			Flip flop inputs		
Q_2	Q_1	Q_0	Q_2^+	Q_1^+	Q_0^+	T_2	T_1	T_0
0	0	0	0	0	1	0	0	1
0	0	1	0	1	0	0	1	1
0	1	0	0	1	1	0	0	1
0	1	1	1	0	0	1	1	1
1	0	0	1	0	1	0	0	1
1	0	1	1	1	0	0	1	1
1	1	0	1	1	1	0	0	1
1	1	1	0	0	0	1	1	1

The Boolean expressions for flip-flop inputs T_2, T_1 and T_0 as

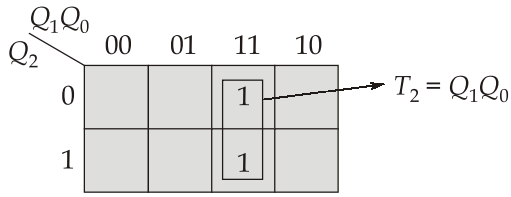
For T_0:



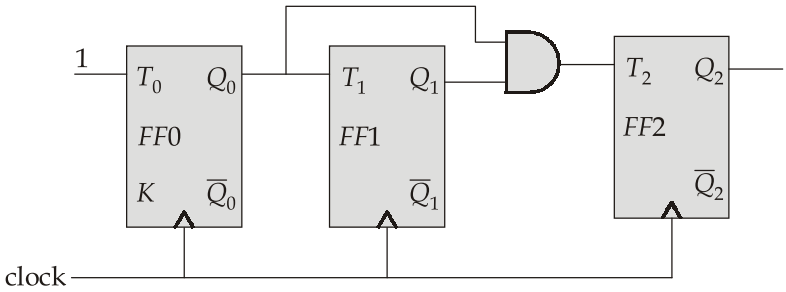
For T_1:



For T_2:



The logic diagram, since it is 3-bit binary counter we need 3FFs.



Q.1 (b) Solution:

(i) 8085 Microprocessor has 16 address lines A0-A15. For interfacing 512 bytes i.e. 2^9 bytes/address locations, 9 address lines are required. So A0-A8 can be used to directly connect to address bus of memory.

From the remaining address lines A9-A15, three address lines A15-A12 can be used as enable and input lines to 3 × 8 decoder for generating a chip select signal to memory as shown below. We assume the chip is selected i.e. $\overline{CS} = 0$ when the inputs to the decoder A15 = 0, A14 = 0, A13 = 0 and A12 = 0. A11-A9 can be considered as don't care signals. Thus, the memory address map is given by,

A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
0	0	0	0	X	X	X	0	0	0	0	0	0	0	0	0
0	0	0	0	X	X	X	1	1	1	1	1	1	1	1	1

Because of the don't care signals, the address range can be

0000 to 01FF

0200 to 03FF

0400 to 05FF

0600 to 07FF

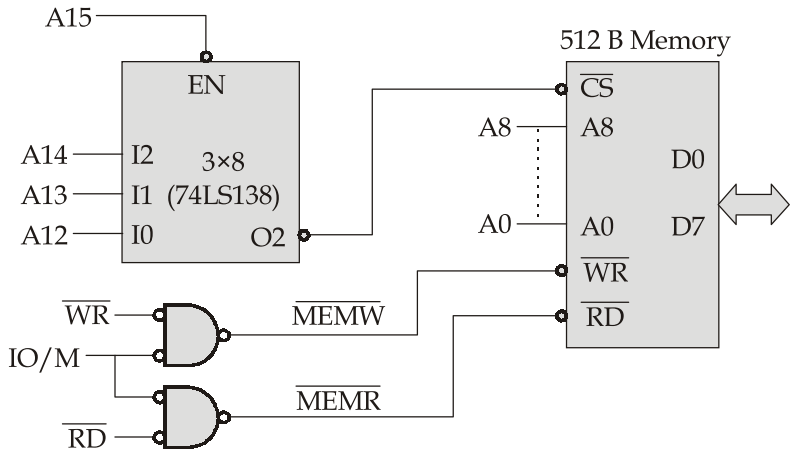
0800 to 09FF

0A00 to 0BFF

0C00 to 0DFF

0E00 to 0FFF

Address decoding circuit:



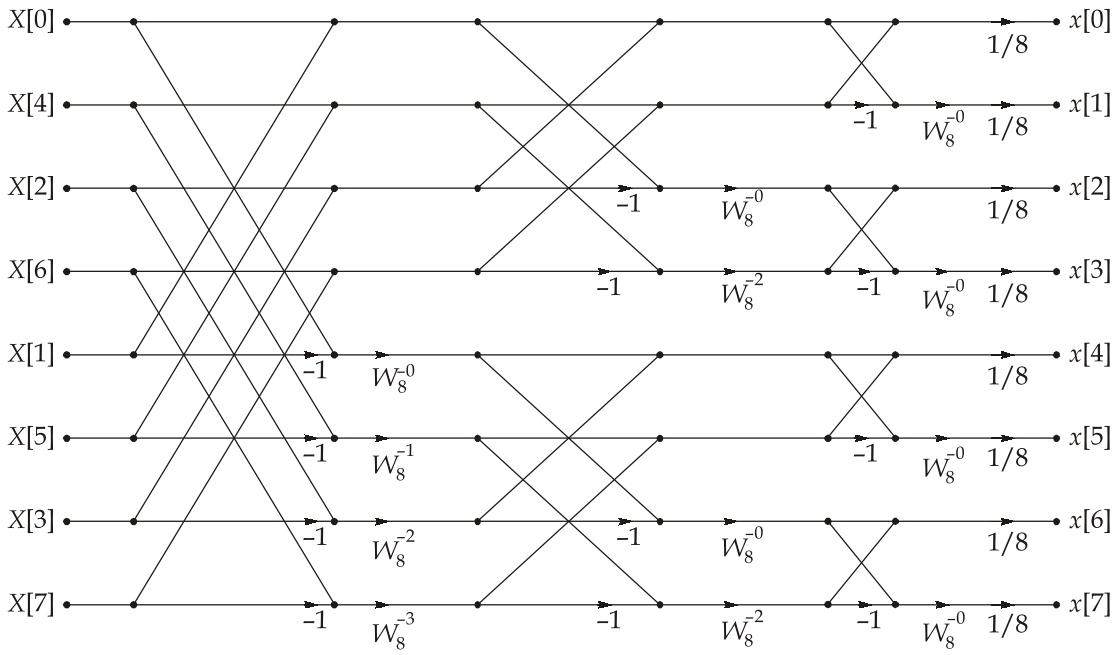
Address decoding circuit using 3 × 8 decoder

(ii) The 8085 assembly language program to move a block of data of 16 bytes starting from address location 2050H to another location starting from 2070H is given as below:

	Instruction	
	LXI H, 2050H;	Load starting address of the block into register pair HL
	LXI D, 2070H;	Load destination address into register pair DE
	LXI B, 10H;	Load counter (number of bytes to move i.e. 16_{10} = 10H) into register pair BC
L1:	MOV A, M;	Move data from source address pointed by HL to accumulator A
	STAX D;	Move data from accumulator A to destination address pointed by DE
	INX H;	Increment source address
	INX D;	Increment destination address
	DCX B;	Decrement counter
	JNZ L1;	Jump to L1 if counter is not zero
	HLT;	Halt the program

Q.1 (c) Solution:

(i)



(ii) **Statement:** If $x[n]$ is causal [i.e., $x[n] = 0$ for $n < 0$], then $x[0] = \lim_{z\to\infty} X(z)$

Proof: By definition of z-transform, we have

$$X(z) = \sum_{n=-\infty}^{+\infty} x[n]\, z^{-n}$$

Since $x[n] = 0$ for $n < 0$, we obtain

$$X(z) = \sum_{n=0}^{\infty} x[n]\, z^{-n}$$

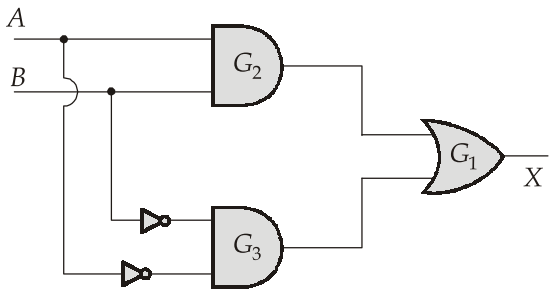
$$X(z) = x[0] + x[1]z^{-1} + x[2]z^{-2} + x[3]z^{-3} + \ldots\ldots\ldots$$

$$\Rightarrow \quad \lim_{z\to\infty} X(z) = \lim_{z\to\infty}\left[x[0] + x[1]z^{-1} + x[2]z^{-2} + x[3]z^{-3} + \ldots\ldots\ldots\right]$$

$$\Rightarrow \quad \lim_{z\to\infty} X(z) = x[0]$$

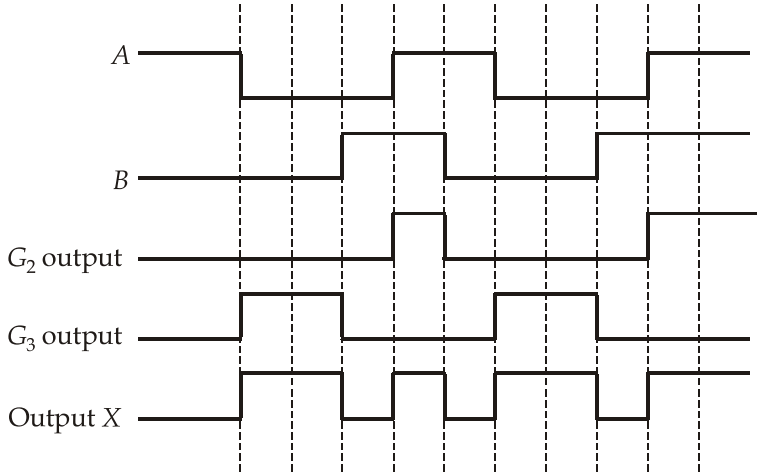
Q.1 (d) Solution:

(i) Given logic circuit,

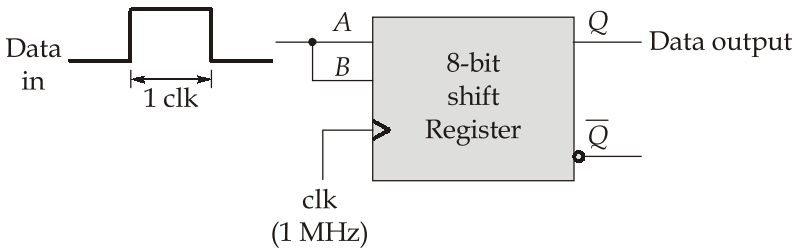


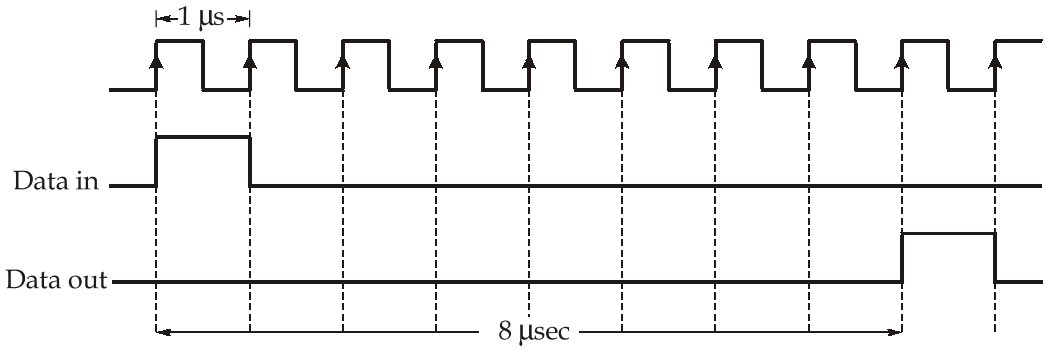
$$\text{Output, } X = AB + \overline{A}\,\overline{B} = A \odot B$$

When both inputs are high or low, the output X is HIGH.



(ii) Given, serial-in-serial-out shift register,





When the given input pulse applied to the serial input, it enters the first stage on the triggering edge of the clock pulse.

It then shifted from stage to stage on each successive clock pulse until it appears on serial output n clock periods.

Given, clock frequency, $f_{clk} = 1$ MHz

the time period of clock,

$$T_{clk} = \frac{1}{f_{clk}} = 1\,\mu\text{sec}$$

For n-bit serial-in-serial-out shift register,

the time delay for output is $n \times T_{clk}$

$$\therefore \quad t_d = 8 \times 1 = 8\ \mu\text{sec}$$

Q.1 (e) Solution:

(i) Given,

$$y(t) = e^{-2t}u(t)$$

$$\text{System function, } H(s) = \frac{Y(s)}{X(s)} = \frac{s-1}{s+1}$$

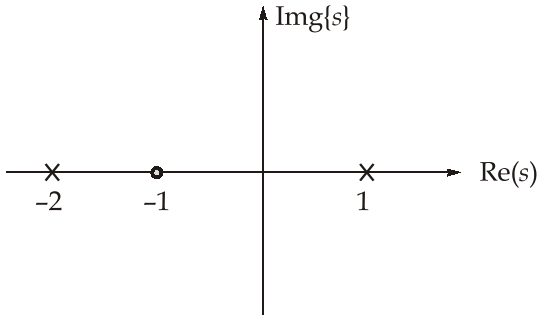
By taking Laplace transform of $y(t)$,

$$Y(s) = \frac{1}{s+2};\ \text{Re}\{s\} > -2$$

$$\therefore \quad X(s) = \frac{Y(s)}{H(s)} = \frac{\frac{1}{s+2}}{\frac{s-1}{s+1}}$$

$$\therefore \quad X(s) = \frac{s+1}{(s-1)(s+2)}$$

The pole-zero diagram of $X(s)$ is



Since the given system is a causal system, the ROC of the impulse response is right-sided. Thus, from the given $H(s)$, the ROC is $\text{Re}\{s\} > -1$. We know that the ROC of $Y(s)$ is at least the intersection of the ROCs of $X(s)$ and $H(s)$.

In the above case, we can choose the ROC of $X(s)$ to be either $-2 < \text{Re}\{s\} < 1$ (or) $\text{Re}\{s\} > 1$.

In both cases, we get the same ROC for $Y(s)$ because the poles at $s = -1$ and $s = 1$ in $H(s)$ and $X(s)$ respectively are cancelled out by zeros.

The partial fraction expansion of $X(s)$ is

$$X(s) = \frac{\frac{2}{3}}{s-1} + \frac{\frac{1}{3}}{s+2}$$

$\therefore$ Taking the ROC of $X(s)$ to be $-2 < \text{Re}\{s\} < 1$

We get, $$x(t) = \frac{2}{3}e^{t}\{-u(-t)\} + \frac{1}{3}e^{-2t}u(t)$$

Taking the ROC of $X(s)$ to be $\text{Re}\{s\} > 1$, we get

$$x(t) = \frac{2}{3}e^{t}u(t) + \frac{1}{3}e^{-2t}u(t)$$

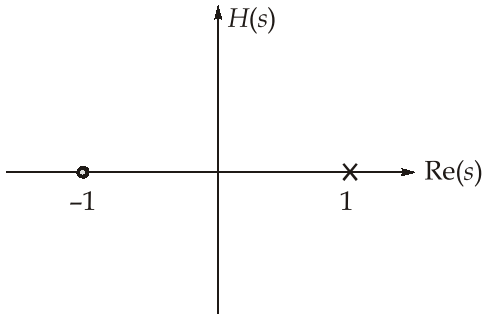
(ii) From the given information, we can write

$$H(s)Y(s) = X(s)$$

(Here, $X(s)$ is the output and $Y(s)$ is the input)

$$\text{Clearly, } H(s) = \frac{X(s)}{Y(s)} = \frac{s+1}{s-1}$$

The pole-zero plot of $H(s)$ is



Since the system is given to be stable, the ROC of the impulse response mustt include the $j\omega$ axis. Thus, the ROC of $H(s)$ has to be $\text{Re}\{s\} < 1$.

We can write,

$$H(s) = \frac{s+1-2+2}{s-1} = \frac{s-1}{s-1} + \frac{2}{s-1}$$

$$\therefore \quad H(s) = 1 + \frac{2}{s-1}$$

$$\therefore \quad h(t) = \delta(t) + 2e^{t}\{-u(-t)\}$$

Also $Y(s)$ has the ROC: $\text{Re}\{s\} > -2$

$\therefore$ $X(s)$ must have the ROC: $-2 < \text{Re}\{s\} < 1$

(i.e, the intersection of the ROCs of $Y(s)$ and $H(s)$).

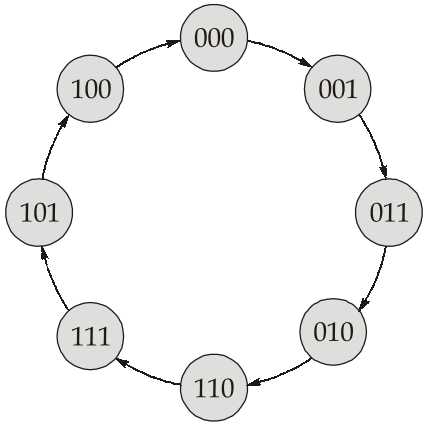
$$\therefore \quad x(t) = -\frac{2}{3}e^{t}u(-t) + \frac{1}{3}e^{-2t}u(t)$$

Q.2 (a) Solution:

The 3-bit Gray code truth table.

Binary Code			Gray Code		
Q_2	Q_1	Q_0	Q_{G2}	Q_{G1}	Q_{G0}
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	1
0	1	1	0	1	0
1	0	0	1	1	0
1	0	1	1	1	1
1	1	0	1	0	1
1	1	1	1	0	0

The state diagram for 3-bit Gray code:



The next state table for the 3-bit Gray code:

Present state			Next state		
Q_{G2}	Q_{G1}	Q_{G0}	Q^+_{G2}	Q^+_{G1}	Q^+_{G0}
0	0	0	0	0	1
0	0	1	0	1	1
0	1	1	0	1	0
0	1	0	1	1	0
1	1	0	1	1	1
1	1	1	1	0	1
1	0	1	1	0	0
1	0	0	0	0	0

Transition table for JK flip-flop:

Flip-flop Inputs		Transition outputs	
Q_N	Q_{N+1}	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

Present state inputs					
J_2	K_2	J_1	K_1	J_0	K_0
0	X	0	X	1	X
0	X	1	X	X	0
0	X	X	0	X	1
1	X	X	0	0	X
X	0	X	0	1	X
X	0	X	1	X	0
X	0	0	X	X	1
X	1	0	X	0	X

Implementing logic functions for present state J and K inputs:

J_2:

Q_{G2} \ $Q_{G1}Q_{G0}$	00	01	11	10
0				1
1	X	X	X	X

$\therefore \quad J_2 = Q_{G1}\,\overline{Q_{G0}}$

K_2:

Q_{G2} \ $Q_{G1}Q_{G0}$	00	01	11	10
0	X	X	X	X
1	1			

$\therefore \quad K_2 = \overline{Q_{G1}}\,\overline{Q_{G0}}$

J_1:

Q_{G2} \ $Q_{G1}Q_{G0}$	00	01	11	10
0		1	X	X
1			X	X

$\therefore \quad J_1 = \overline{Q_{G2}}\,Q_{G0}$

K_1:

Q_{G2} \ $Q_{G1}Q_{G0}$	00	01	11	10
0	X	X		
1	X	X	1	

$\therefore \quad K_1 = Q_{G2}\,Q_{G0}$

J_0:

Q_{G2} \ $Q_{G1}Q_{G0}$	00	01	11	10
0	1	X	X	
1		X	X	1

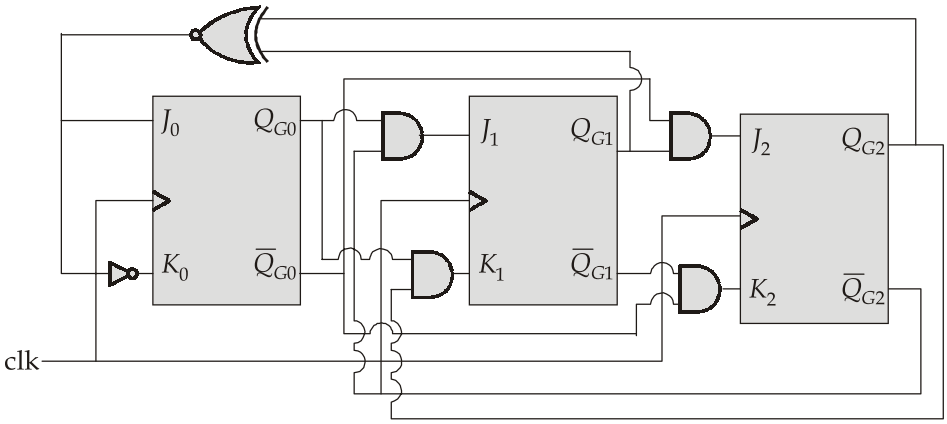
$$\therefore \quad J_0 = \overline{Q_{G2}}\,\overline{Q_{G1}} + Q_{G2}\,Q_{G1}$$

K_0:

Q_{G2} \ $Q_{G1}Q_{G0}$	00	01	11	10
0	X		1	X
1	X	1		X

$$\therefore \quad K_0 = \overline{Q_{G1}}\,Q_{G2} + \overline{Q_{G2}}\,Q_{G1}$$

$$= Q_{G2} \oplus Q_{G1}$$

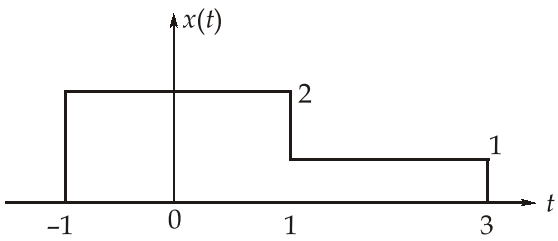


Q.2 (b) Solution:

(i) Given,

$$x(t) = \begin{cases} 2, & -1 \le t \le 1 \\ 1, & 1 < t \le 3 \\ 0, & \text{elsewhere} \end{cases}$$

and $$h(t) = 2\delta(t+1) + \delta(t+2)$$



Let $y(t)$ be the output of $x(t) * h(t)$

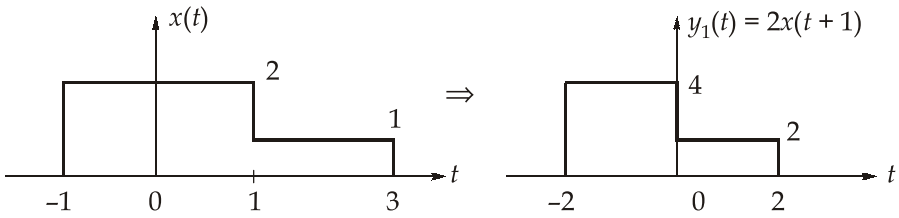
$$\therefore \quad y(t) = x(t) * [2\delta(t+1) + \delta(t+2)]$$
$$= 2(x(t) * \delta(t+1)) + (x(t) * \delta(t+2))$$

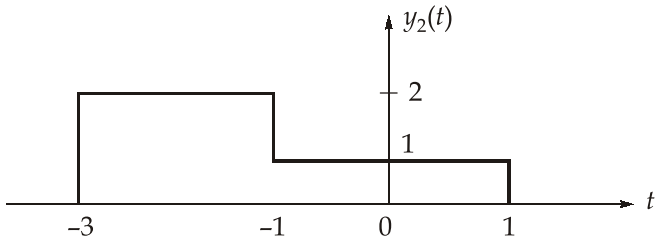
We know that,

$$x(t) * \delta(t - t_0) = x(t - t_0)$$

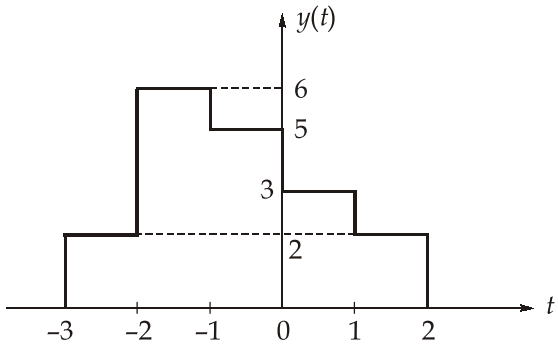
Thus, $$y(t) = 2x(t+1) + x(t+2)$$

Let $$y_1(t) = 2x(t+1)$$
$$y_2(t) = x(t+2)$$





$$\therefore \quad y(t) = y_1(t) + y_2(t)$$



$$y(t) = \begin{cases} 2; & -3 \le t \le -2 \\ 6; & -2 \le t \le -1 \\ 5; & -1 \le t \le 0 \\ 3; & 0 \le t < 1 \\ 2; & 1 \le t \le 2 \end{cases}$$

(ii) Given,

$$\frac{dy(t)}{dt}+10y(t) = \int_{-\infty}^{\infty} x(\tau)z(t-\tau)d\tau - x(t)$$

where $z(t) = e^{-t}u(t) + \delta(t)$

By taking Fourier transform,

$$j\omega\, Y(\omega) + 10Y(\omega) = X(\omega)Z(\omega) - X(\omega)$$

$$Y(\omega)[10 + j\omega] = X(\omega)[Z(\omega) - 1]$$

$$H(\omega) = \frac{Y(\omega)}{X(\omega)} = \frac{Z(\omega)-1}{10+j\omega}$$

Taking Fourier Transform of $z(t)$,

$$Z(\omega) = \frac{1}{1+j\omega}+1$$

$$Z(\omega) = \frac{2+j\omega}{1+j\omega}$$

$$H(\omega) = \frac{\frac{2+j\omega}{1+j\omega}-1}{10+j\omega} = \frac{(2+j\omega-1-j\omega)}{(1+j\omega)(10+j\omega)}$$

$$\therefore \quad H(\omega) = \frac{1}{(1+j\omega)(10+j\omega)}$$

By using partial fraction expansion,

$$\frac{1}{(1+j\omega)(10+j\omega)} = \frac{A}{1+j\omega}+\frac{B}{10+j\omega}$$

$$A = \left.\frac{1}{10+j\omega}\right|_{j\omega=-1} = \frac{1}{9}$$

$$B = \left.\frac{1}{1+j\omega}\right|_{j\omega=-10} = \frac{-1}{9}$$

$$\therefore \quad H(\omega) = \frac{\frac{1}{9}}{1+j\omega}+\frac{\frac{-1}{9}}{10+j\omega}$$

By taking inverse Fourier transform,

$$h(t) = \frac{1}{9}e^{-t}u(t) - \frac{1}{9}e^{-10t}u(t)$$

Q.2 (c) Solution:

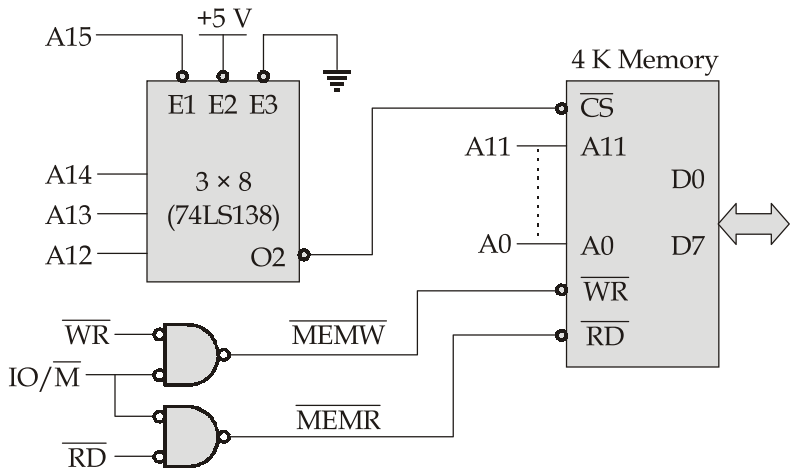
(i) 4 KB memory has $4 \times 2^{10} = 2^{12}$ address locations and thus, requires 12 address lines for addressing. But 8085 has 16 address lines (A0 - A15). Hence, four of address lines (A15 - A12) are used for address decoding to select the memory block.

Given that starting address for memory is A000H. So, for 4 KB memory, ending address becomes A000H + 0FFFH [4 KB] = AFFFH.

	A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
Start Address:	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
End Address:	1	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1

A0-A11 address lines are directly connected to address bus of memory chip. A12-A15 are used for generating chip select for memory chip.

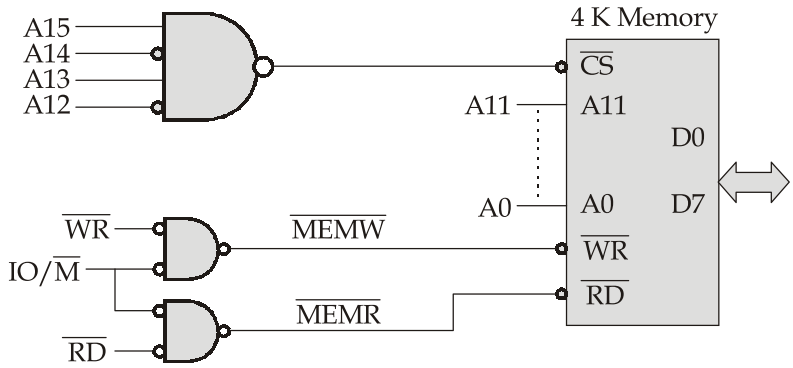
Address decoding circuit using 3 × 8 decoder.



Address decoding circuit using 3 × 8 decoder

A15 line is used for enabling 3 × 8 decoder chip. A12, A13, A14 lines are connected to 3 × 8 chip as inputs. When these lines are 010, output should be '0'. This is provided at O2 pin of 3 × 8 decoder which is connected to chip select input of memory.

Address decoding circuit using only NAND gates:



Address decoding circuit using NAND gates

A15, A14, A13, A12 inputs should be 1010, for enabling the chip. So, the address decoding circuit using NAND Gates is as shown above.

(ii) Algorithm:

1. Start the program by loading HL register pair with address of memory location.
2. Move the first 8-bit number to register B.
3. Increment the memory location and move the second 8-bit number to register C. It acts as the counter for how many times B will be added.
4. Add the current content of Register B to the content of the Accumulator (A). The sum is stored back in A.
5. Check for carry. If a carry is generated, increment the content of Register D. This accounts for the overflow into the higher byte.
6. Decrement the content of Register C and keep on adding the content of register B to the accumulator till C is zero.
7. Store the content of the Accumulator (A) which corresponds to the LSB of the final 16-bit result in memory.
8. Store the content of register D which corresponds to the MSB of the final 16-bit result in memory.
9. Terminate the program.

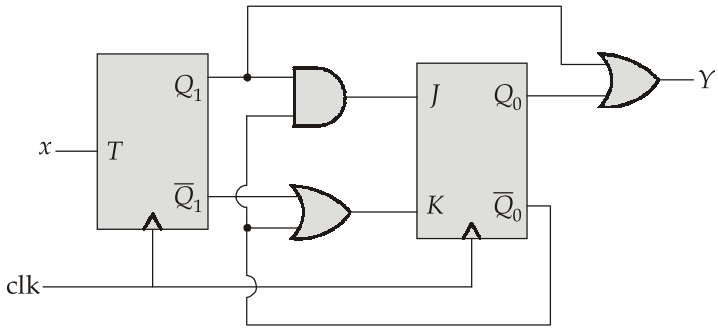
Program:

```
MVI     D, 00H      Initialize register D to 00H.
MVI     A, 00H      Initialize Accumulator content to 00H.
LXI     H, 4150H
MOV     B, M        Get the first number in B-register
INX     H
MOV     C, M        Get the second number in C-register
```

LOOP:	ADD	B	Add content of register A to register B.
	JNC	NEXT	JUMP on no carry to NEXT.
	INR	D	Increment content of register D.
NEXT:	DCR	C	Decrement content of register C.
	JNZ	LOOP	Jump to LOOP if C is not zero.
	STA	4152 H	Store the LSB of result in Memory.
	MOV	A, D	
	STA	4153 H	Store the MSB of result in Memory.
	HLT		Terminate the program.

Q.3 (a) Solution:

(i) Given sequential circuit,



We can write the following equations for the inputs.

$$T = x \text{ (input)}$$

$$J = Q_1\overline{Q_0}$$

$$K = \overline{Q_1} + \overline{Q_0}$$

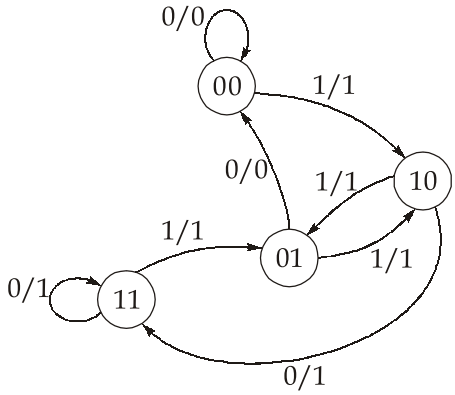
The next states for the above state equations,

Q_1	Q_0	$T = x$	$Q_1(t+1)$	J	K	$Q_0(t+1)$	Y
0	0	0	0	0	1	0	0
0	0	1	1	0	1	0	1
0	1	0	0	0	1	0	0
0	1	1	1	0	1	0	1
1	0	0	1	1	1	1	1
1	0	1	0	1	1	1	1
1	1	0	1	0	0	1	1
1	1	1	0	0	0	1	1

The state table is,

		Next state		Output (Y)	
Q_1	Q_0	$x = 0$	$x = 1$	$x = 0$	$x = 1$
0	0	00	10	0	1
0	1	00	10	0	1
1	0	11	01	1	1
1	1	11	01	1	1

The required state diagram is,



(ii) Given, $x[n] = a^n \sin(\omega_0 n)\, u[n]$

Let $g[n] = a^n\, u[n]$

Take z-transform on both sides,

$$G(z) = \frac{1}{1-az^{-1}};\ \text{ROC}: |z| > |a|$$

$$\therefore \quad x[n] = g[n] \cdot \sin(\omega_0 n)$$

$$= g[n]\left[\frac{e^{j\omega_0 n} - e^{-j\omega_0 n}}{2j}\right] = \frac{1}{2j} g[n] e^{j\omega_0 n} - \frac{1}{2j} g[n] e^{-j\omega_0 n}$$

Take z-transform on both sides,

$$X(z) = \frac{1}{2j}\left[G\left(\frac{z}{e^{j\omega_0}}\right) - G\left(\frac{z}{e^{-j\omega_0}}\right)\right];\ |z| > \left|e^{\pm j\omega_0}\right| \cdot |a|$$

$\because$ Scaling in z-domain property: $\alpha^n x[n] \leftrightarrow X\left(\frac{z}{\alpha}\right);\ \text{ROC}: |\alpha| R$

Where 'R' is the ROC of $x[n]$

$$\Rightarrow \quad X(z) = \frac{1}{2j}\left[\frac{1}{1-a\left(\frac{z}{e^{j\omega_0}}\right)^{-1}} - \frac{1}{1-a\left(\frac{z}{e^{-j\omega_0}}\right)^{-1}}\right];\ |z| > |a|$$

$$\Rightarrow \quad X(z) = \frac{1}{2j}\left[\frac{1}{1-a\,e^{j\omega_0}z^{-1}} - \frac{1}{1-a\,e^{-j\omega_0}z^{-1}}\right]$$

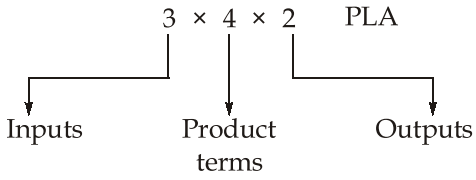
$$\Rightarrow \quad X(z) = \frac{1}{2j}\left[\frac{1-a\,e^{-j\omega_0}z^{-1}-1+a\,e^{j\omega_0}z^{-1}}{1-a\,e^{-j\omega_0}z^{-1}-a\,e^{j\omega_0}z^{-1}+a^2z^{-2}}\right]$$

$$\Rightarrow \quad X(z) = \frac{1}{2j}\left[\frac{az^{-1}\left(e^{j\omega_0}-e^{-j\omega_0}\right)}{1-a\,z^{-1}\left(e^{-j\omega_0}+e^{j\omega_0}\right)+a^2z^{-2}}\right]$$

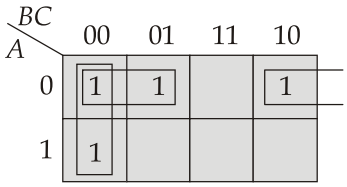
$$\Rightarrow \quad X(z) = \left[\frac{az^{-1}\sin\omega_0}{1-2a\,z^{-1}\cos\omega_0+a^2z^{-2}}\right];\ |z| > |a|$$

Q.3 (b) Solution:

(i) Given,

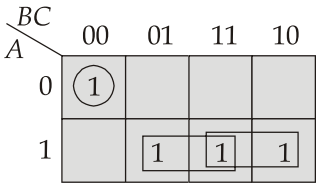


K-map for F_1:



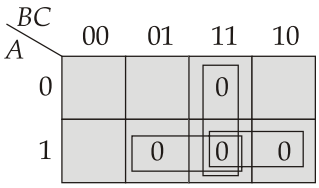
$$F_1 = \bar{B}\bar{C} + \bar{A}\bar{B} + \bar{A}\bar{C}$$

K-map for F_2:



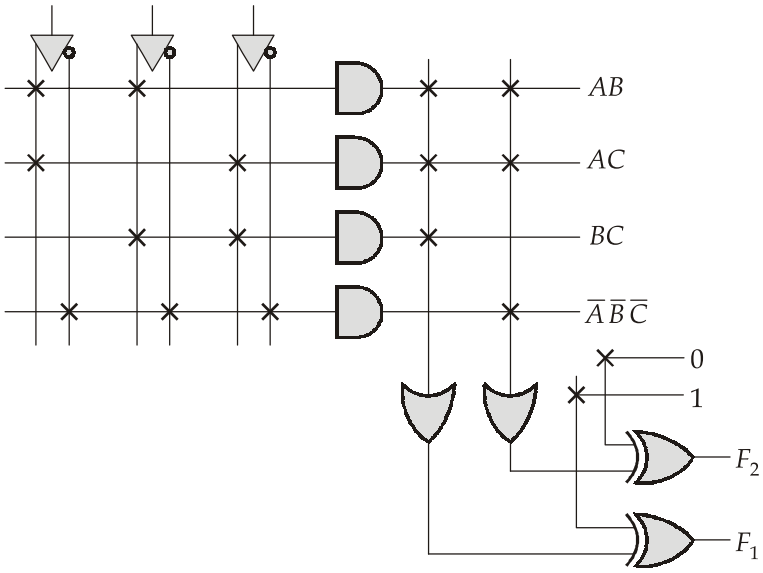
$$F_2 = AB + AC + \bar{A}\bar{B}\bar{C}$$

K-map for $\overline{F_1}$:



$$\overline{F_1} = AB + AC + BC$$

PLA implementation:



PLA programming table:

	Product term	Inputs			Outputs	
		A	B	C	F_1	F_2
1	AB	1	1	–	1	1
2	AC	1	–	1	1	1
3	BC	–	1	1	1	–
4	$\overline{A}\,\overline{B}\,\overline{C}$	0	0	0	–	1

(ii) Given,

$$\text{reference voltage, } V_R = 12 \text{ V}$$

$$\text{fixed count} = 010110_2 = (22)_{10}$$

$$N = 6 \text{ bit}$$

Discharge time of dual slope ADC,

$$T_2 = \frac{V_{in} \times T_1}{V_R}$$

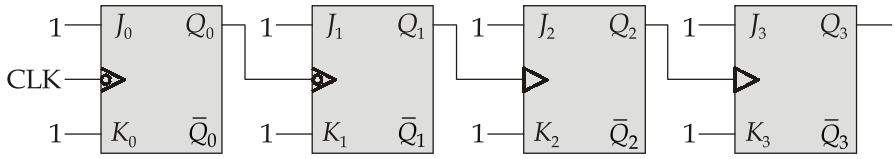
$$V_{\text{in, max}} = \frac{V_R \times T_2}{T_1}$$

where, $T_2 = 2^N \times T = 2^6 \times T$

$$V_{\text{in, max}} = \frac{(12) \times 2^6 \times T}{22T} = 34.9 \text{ V}$$

Q.3 (c) Solution:

(i)



Q_0 toggles only when clock changes from '1' to '0'. i.e., Q_0 toggles for every clock pulse.

Q_1 toggles only when Q_0 changes from '1' to '0'.

Q_2 toggles only when Q_1 changes from '0' to '1'.

Q_3 toggles only when Q_2 changes from '0' to '1'.

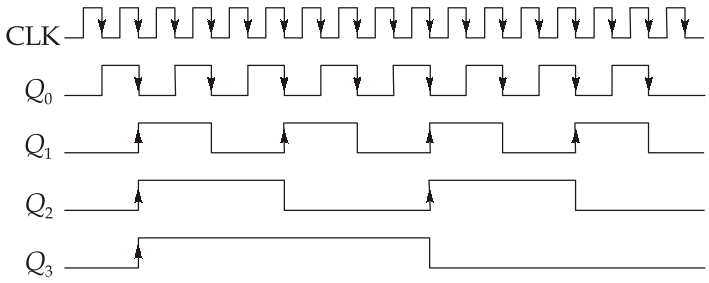
	Q_3	Q_2	Q_1	Q_0	Decimal equivalent
Initially	0	0	0	0	0
1^{st} clk →	0	0	0	1	1
2^{nd} clk →	1	1	1	0	14
3^{rd} clk →	1	1	1	1	15
4^{th} clk →	1	1	0	0	12
5^{th} clk →	1	1	0	1	13
6^{th} clk →	1	0	1	0	10
7^{th} clk →	1	0	1	1	11
8^{th} clk →	1	0	0	0	8
9^{th} clk →	1	0	0	1	9
10^{th} clk →	0	1	1	0	6
11^{th} clk →	0	1	1	1	7
12^{th} clk →	0	1	0	0	4
13^{th} clk →	0	1	0	1	5
14^{th} clk →	0	0	1	0	2
15^{th} clk →	0	0	1	1	3
16^{th} clk →	0	0	0	0	0

(ii)

$$\text{Frequency of } Q_0 = \frac{f_{clk}}{2} = \frac{160\text{ kHz}}{2} = 80\text{ kHz}$$

$$\text{Frequency of } Q_2 = \frac{f_{clk}}{8} = \frac{160\text{ kHz}}{8} = 20\text{ kHz}$$

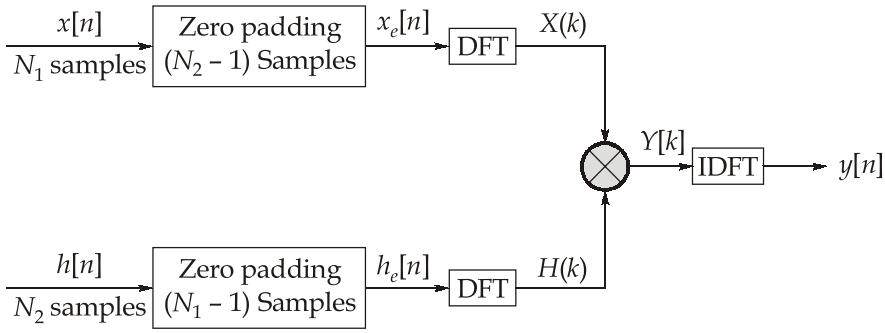
(iii)



Q.4 (a) Solution:

(i) The circular convolution of two sequences of lengths N_1 and N_2 respectively, can be made equal to the linear convolution of the two sequences by zero padding both the sequences, so that they both consist of $(N_1 + N_2 - 1)$ samples as shown below.

Then find the DFT of both the sequences having $(N_1 + N_2 - 1)$ samples individually. Multiply the DFTs of both the sequences and find the inverse DFT of the obtained product to get the convolved sequence.



Given, $x[n] = \{1, 1, 1\} \rightarrow N_1 = 3$ samples

$h[n] = \{1, 1\} \rightarrow N_2 = 2$ samples

$\therefore \quad N_1 + N_2 - 1 = 3 + 2 - 1 = 4$ samples

Pad $(N_2 - 1)$ zeros to $x[n]$ and pad $(N_1 - 1)$ zeros to $h[n]$. We get,

$$x_e[n] = \{1, 1, 1, 0\}$$

$$h_e[n] = \{1, 1, 0, 0\}$$

DFT of $x_e[n]$ and $h_e[n]$ is calculated as below,

$$X(k) = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1+1+1+0 \\ 1-j-1+0 \\ 1-1+1+0 \\ 1+j-1+0 \end{bmatrix} = \begin{bmatrix} 3 \\ -j \\ 1 \\ j \end{bmatrix}$$

$$H(k) = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1+1+0+0 \\ 1-j+0+0 \\ 1-1+0+0 \\ 1+j+0+0 \end{bmatrix} = \begin{bmatrix} 2 \\ 1-j \\ 0 \\ 1+j \end{bmatrix}$$

Take sample by sample product to get $Y(k)$:

$$\begin{aligned} Y(k) &= X(k) \cdot H(k) \\ k = 0 : Y(0) &= X(0) \cdot H(0) = (3) \cdot (2) = 6 \\ k = 1 : Y(1) &= X(1) \cdot H(1) = (-j) \cdot (1 - j) = (-1 - j) \\ k = 2 : Y(2) &= X(2) \cdot H(2) = (1) \cdot (0) = 0 \\ k = 3 : Y(3) &= X(3) \cdot H(3) = (j) \cdot (1 + j) = (-1 + j) \end{aligned}$$

$$\therefore \quad y[n] = \frac{1}{N}[W^{-1}] \times [Y(k)] = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & j & -1 & -j \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & j \end{bmatrix} \begin{bmatrix} 6 \\ -1-j \\ 0 \\ -1+j \end{bmatrix}$$

$$= \frac{1}{4} \begin{bmatrix} 6-1-j+0-1+j \\ 6-j+1+0+j+1 \\ 6+1+j+0+1-j \\ 6+j-1+0-j-1 \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 4 \\ 8 \\ 8 \\ 4 \end{bmatrix}$$

$$\Rightarrow \quad y[n] = \{1, 2, 2, 1\}$$

(ii) We know, the Discrete Fourier Series (DFS) coefficients of signal $x[n]$ is given by,

$$C_k = \frac{1}{N} \sum_{n=0}^{N-1} x[n] e^{-j\omega_0 kn}$$

Where, 'N' is the fundamental period and 'ω_0' is the fundamental frequency,

$$\omega_0 = \frac{2\pi}{N}$$

$$\Rightarrow \quad C_k = \frac{1}{N} \sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi}{N}kn} \qquad \text{...(i)}$$

We know, the Discrete Fourier Transform (DFT) of signal $x[n]$ is,

$$X(k) = \sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi}{N}kn} \qquad \text{...(ii)}$$

On comparing equations (i) and (ii), the relationship between C_k and $X(k)$ is,

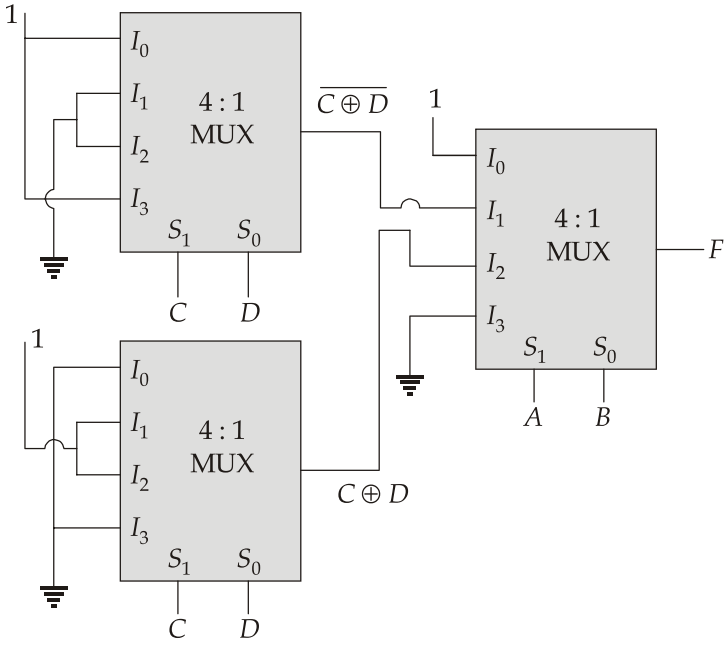
$$C_k = \frac{X(k)}{N}$$

Q.4 (b) Solution:

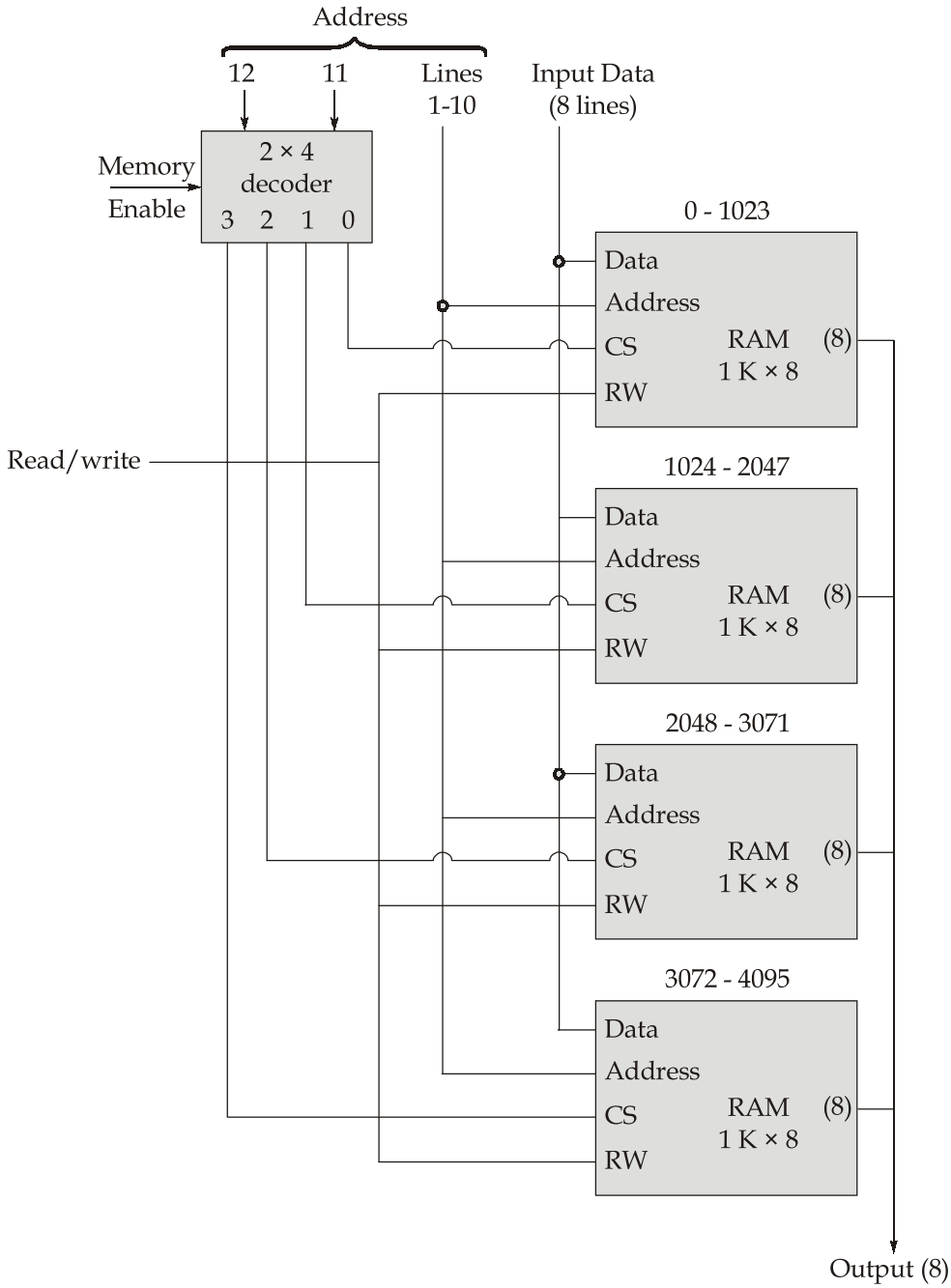
(i) $F(A, B, C, D) = \Sigma m(0, 1, 2, 3, 4, 7, 9, 10)$

Consider a 4 : 1 MUX with select lines as AB.

A	B	C	D	F	Input to 4 : 1 MUX
0	0	0	0	1	$I_0 = 1$
0	0	0	1	1	
0	0	1	0	1	
0	0	1	1	1	
0	1	0	0	1	$I_1 = \overline{C \oplus D}$
0	1	0	1	0	
0	1	1	0	0	
0	1	1	1	1	
1	0	0	0	0	$I_2 = C \oplus D$
1	0	0	1	1	
1	0	1	0	1	
1	0	1	1	0	
1	1	0	0	0	$I_3 = 0$
1	1	0	1	0	
1	1	1	0	0	
1	1	1	1	0	



(ii)



The 4 K word memory requires 12-bit address. The 10 least significant bits of the address are applied to the address input of all four chips. The other two most significant bits are applied to a 2 × 4 decoder. The four outputs of the decoder are applied to the CS inputs of each chip. The memory is disabled when the memory enable input of the decoder is equal to 0. This causes all four outputs of the decoder to be in the 0 state and none of the chips are selected. When the decoder is enabled address bits 12 and 11 determine the particular chip that is selected.

If bits 12 and 11 are equal to 00, the first RAM chip is selected. The remaining 10 address bits select a word within the chip in the range from 0 to 1023. The next 1024 words are selected from the second RAM chip with a 12-bit address that starts with 01 and followed by the ten bits from the common address lines.

Q.4 (c) Solution:

(i) **Algorithm:** The algorithm for converting a BCD number into its binary equivalent first unpacks the BCD number into two digits tens digit (upper nibble) and units digit (lower nibble). The tens digit is multiplied by 10D or 0AH in hexadecimal form, and then the unit digit is added with the result. The result is stored in 9001 H.

Program

```
MOV DPTR, # 9000H ;  Point the data pointer to the location of the BCD number.
MOVX A, @ DPTR    ;  Move the BCD number to the accumulator.
MOV R0, A         ;  Copy BCD number to R0 register.
ANL A, # 0FH      ;  Mask the higher-order four bits.
MOV R1, A         ;  Save the (lower nibble) in register R1.
MOV A, R0         ;  Bring the data again to register A.
ANL A, # F0H      ;  Mask the lower-order four bits.
SWAP A            ;  Bring the higher four bits (upper nibble) to the lower
                     four bits.
MOV B, # 0AH      ;  Store 10D in register B.
MUL AB            ;  Multiply the tens digit of BCD number by 10.
ADD A, R1         ;  Add the result with the lower nibble (unit digit).
INC DPTR          ;  Increment the data pointer to the next location 9001H.
MOVX @ DPTR, A    ;  Save the result at the location 9001H pointed by the data
                     pointer.
WAIT : SJMP WAIT  ;  Terminate program execution.
```

(ii) Given, $\sum_{i=1}^{10} i = 1 + 2 + 3 + \ldots\ldots\ldots + 9 + 10$

Algorithm:

1. Initialize a counter (CX) to 10 = 000AH.
2. Initialize a sum register (AX) to 0.

3. Loop while CX is less than or equal to 10:
   - Add the current value of the counter (CX) to the sum (AX)
   - Decrement the counter (CX).
4. The final sum will be in AX.

Program:

```
START:
                MOV     CX, 0001H     ;   CX = 1
                MOV     AX, 0000H     ;   AX = 0
CONTINUE :      ADD     AX, CX        ;   AX ← AX + CX
                INC     CX            ;   CX ← CX + 1
                CMP     CX, 000AH     ;   Compare CX content with 10
                JBE     CONTINUE      ;   Jump if contents of CX is below or equal
                                          to '10'.
                HLT
END :
```

Section B : Digital Circuits + Signals and Systems + Microprocessors & Microcontroller

Q.5 (a) Solution:

(i) Given, $f(t) = 2e^{-t}\cos 10t - t^4 + 6e^{-(t-10)};\ t > 0$

Let, $f_1(t) = 2e^{-t}\cos 10t\, u(t)$

$f_2(t) = -t^4 u(t)$

$f_3(t) = 6e^{-(t-10)}u(t)$

Thus, $f(t) = f_1(t) + f_2(t) + f_3(t)$

Taking Laplace transform,

$$F(s) = F_1(s) + F_2(s) + F_3(s)$$

For $f_1(t)$;

We know that, $\cos \omega_0 t\, u(t) \xleftrightarrow{L} \frac{s}{s^2 + \omega_0^2};\ \text{Re}\{s\} > 0$

$$e^{s_0 t}x(t) \xleftrightarrow{L} X(s - s_0)$$

$\therefore$ $$\cos 10t\, u(t) \xleftrightarrow{L} \frac{s}{s^2 + 100}$$

$$e^{-t}\cos 10t\,u(t) \xleftrightarrow{L} \frac{s+1}{(s+1)^2+100};\ \text{Re}\{s\} > -1$$

$$2e^{-t}\cos 10t\,u(t) \xleftrightarrow{L} \frac{2(s+1)}{(s+1)^2+100};\ \text{ROC}_1 : \text{Re}\{s\} > -1$$

For $f_2(t)$;

We know that, $t^k u(t) \xleftrightarrow{L} \frac{k!}{s^{k+1}};\ \text{Re}\{s\} > 0$

$$-t^4 u(t) \xleftrightarrow{L} \frac{-4!}{s^5};\ \text{Re}\{s\} > 0$$

$$-t^4 u(t) \xleftrightarrow{L} \frac{-24}{s^5};\ \text{ROC}_2 : \text{Re}\{s\} > 0$$

For $f_3(t)$:

$$\begin{aligned} f_3(t) &= 6e^{-(t-10)}u(t) \\ &= 6e^{-t}\cdot e^{10}\cdot u(t) \\ &= 6e^{10}\,e^{-t}u(t) \end{aligned}$$

$$\therefore \quad 6e^{10}\cdot e^{-t}u(t) \xleftrightarrow{L} 6e^{10}\frac{1}{s+1};\ \text{ROC}_3 : \text{Re}\{s\} > -1$$

From (i), (ii) and (iii),

$$F(s) = \frac{2(s+1)}{(s+1)^2+100} - \frac{24}{s^5} + \frac{6e^{10}}{s+1};\ \text{ROC} : \text{ROC}_1 \cap \text{ROC}_2 \cap \text{ROC}_3 : \text{Re}\{s\} > 0$$

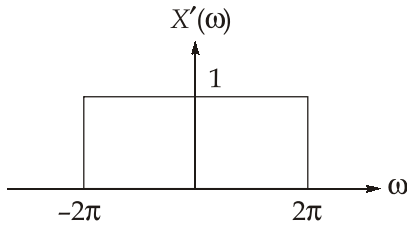
(ii) Given, $x(t) = \dfrac{\sin(2\pi t)}{\pi(t-1)}$

We know that, $A\,\text{rect}\left(\dfrac{t}{T}\right) \longleftrightarrow AT\dfrac{\sin(\omega T/2)}{\omega T/2}$

Using the principle of duality,

$$AT\frac{\sin(tT/2)}{tT/2} \longleftrightarrow 2\pi A\,\text{rect}\left(-\frac{t}{T}\right) = 2\pi A\,\text{rect}\left(\frac{t}{T}\right)$$

Thus, $x'(t) = \dfrac{\sin(2\pi t)}{\pi t} \longleftrightarrow$



$$x'(t) = \frac{\sin(2\pi t)}{\pi t} \longleftrightarrow u(\omega+2\pi) - u(\omega-2\pi)$$

We have, $$x(t) = \frac{\sin(2\pi t)}{\pi(t-1)} = \frac{\sin(2\pi t - 2\pi)}{\pi(t-1)} = \frac{\sin(2\pi(t-1))}{\pi(t-1)} = x'(t-1)$$

We know that, $$x(t) \xleftrightarrow{F.T} X(\omega)$$

$$x(t-1) \xleftrightarrow{F.T} e^{-j\omega}X(\omega)$$

Therefore, $$x(t) = \frac{\sin(2\pi(t-1))}{\pi(t-1)} \xleftrightarrow{F.T} e^{-j\omega}\left[u(\omega+2\pi) - u(\omega-2\pi)\right]$$

$\therefore$ $$X(\omega) = e^{-j\omega}[u(\omega + 2\pi) - u(\omega - 2\pi)]$$

Q.5 (b) Solution:

(i) For the given 4-bit R-$2R$ ladder converter output voltage,

$$V_0 = V_R \cdot \frac{R_f}{R}\left[\frac{b_1}{2^1} + \frac{b_2}{2^2} + \frac{b_3}{2^3} + \frac{b_4}{2^4}\right]$$

Given that, $V_R = 10$ V

$R = 5$ kΩ

value of 1 LSB = 1 volt

$$1 = 10 \times \frac{R_f}{5\times10^3} \cdot \left[\frac{1}{2^4}\right]$$

$$R_f = \frac{5\times10^3 \times 2^4}{10}$$

$$= 500 \times 2^4 = 8000\ \Omega$$

$$R_f = 8\ \text{k}\Omega$$

(ii) For binary value of 1000

$$b_1 = 1;\ b_2 = b_3 = b_4 = 0$$

$\therefore$ $$8 = 10 \times \frac{R_f}{5\times10^3}\left[\frac{1}{2}\right]$$

$$R_f = \frac{8\times2\times5\times10^3}{10} = 8\ \text{k}\Omega$$

(iii) Thus for getting a full scale voltage of 10 V,

$$b_1 = b_2 = b_3 = b_4 = 1$$

$$\frac{R_f \times 10}{5\times 10^3}\left[\frac{1}{2}+\frac{1}{2^2}+\frac{1}{2^3}+\frac{1}{2^4}\right] = 10$$

$$\frac{R_f}{5\times 10^3}[0.9375] = 1$$

$$R_f = \frac{5\times 10^3}{0.9375} = 5.333\ \text{k}\Omega$$

Q.5 (c) Solution:

We know, $$y[n] = x[n] * h[n] = \sum_{k=-\infty}^{+\infty} x[k]\cdot h[n-k]$$

Given, $x[n] = \left(\frac{1}{2}\right)^{(-n-1)} u[-n-1] = \left(\frac{1}{2}\right)^{-1}\left(\frac{1}{2}\right)^{-n} u[-n-1] = 2\left(\frac{1}{2}\right)^{-n} u[-n-1]$

and $$h[n] = u[n-1]$$

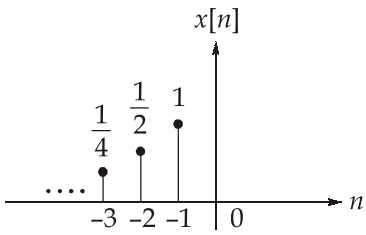
Limits of $x[n]$: $-\infty \le n \le -1$

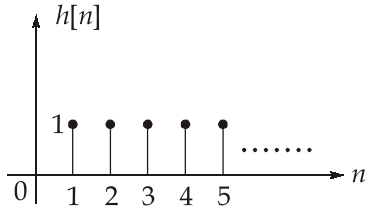
Limits of $h[n]$: $1 \le n \le \infty$

The range of the convoluted signal of two discrete-time signals is the sum of the range of the individual signals. Thus,

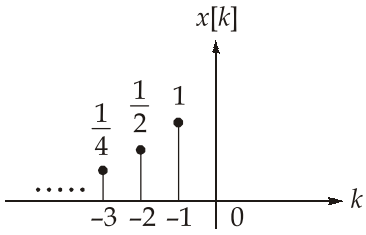
Limits of $y[n]$: $(-\infty + 1) \le n \le (-1 + \infty) = -\infty \le n \le \infty$

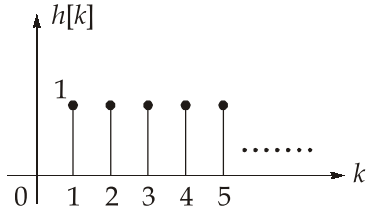
Plot $x[n]$ and $h[n]$:



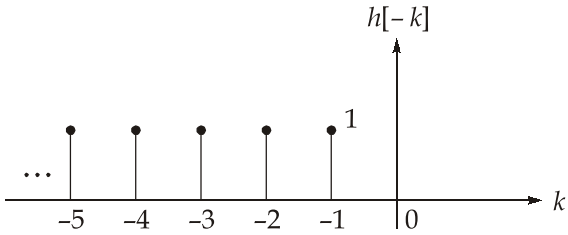


Change the n-axis to k-axis:

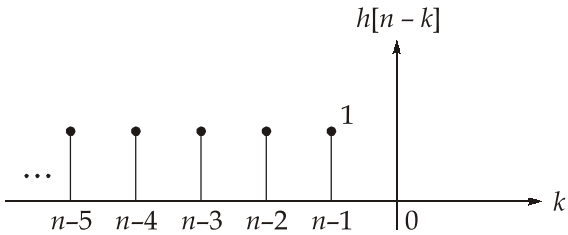




Find time reversal plot of $h[k]$:



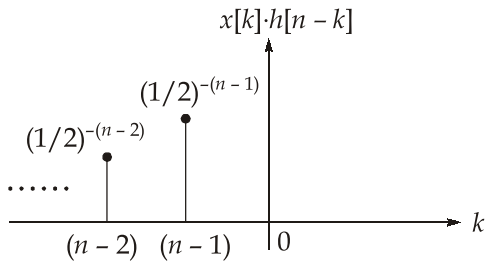
Plot $h[n-k]$:



Case (i): $(n-1) < -1$

$\Rightarrow$ $n < 0$

$$y[n] = \sum_{k=-\infty}^{+\infty} x[k]\cdot h[n-k]$$



$\Rightarrow$ $$y[n] = \sum_{k=-\infty}^{n-1} 2\left(\frac{1}{2}\right)^{-k}\cdot(1)$$

$$y[n] = 2\sum_{k=-\infty}^{n-1}\left(\frac{1}{2}\right)^{-k}$$

Put $(n-1)-k=p$

Limits: $k = n-1 \Rightarrow p = 0$

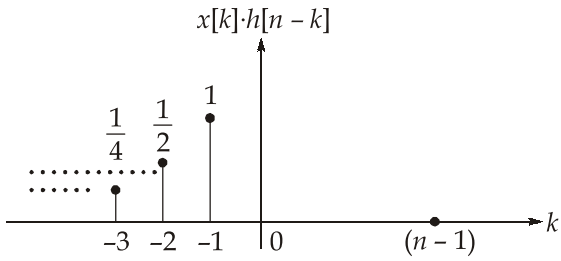
$k = -\infty \Rightarrow p = \infty$

$\therefore$ $$y[n] = 2\sum_{p=\infty}^{0}\left(\frac{1}{2}\right)^{p-(n-1)} = 2\left(\frac{1}{2}\right)^{-(n-1)}\sum_{p=\infty}^{0}\left(\frac{1}{2}\right)^{p}$$

$$= 2(2)^{(n-1)}\sum_{p=0}^{\infty}\left(\frac{1}{2}\right)^{p} = 2^{n}\sum_{p=0}^{\infty}\left(\frac{1}{2}\right)^{p}$$

$$= 2^{n}\left[\frac{1}{1-1/2}\right] = 2^{n}[2] = 2^{n+1}$$

Case (ii): $(n-1) \geq -1 \quad \Rightarrow \quad n \geq 0$



$$y[n] = \sum_{k=-\infty}^{+\infty} x[k]\cdot h[n-k]$$

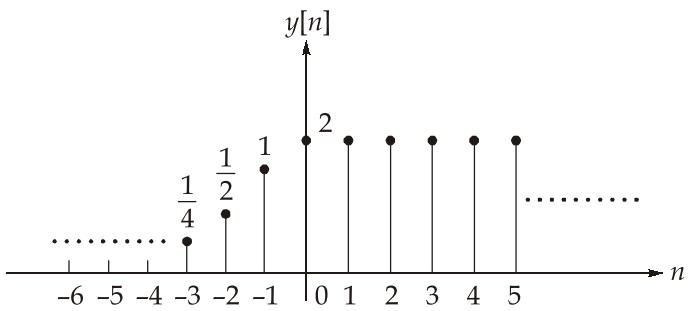
$$\Rightarrow \quad y[n] = \sum_{k=-\infty}^{-1} 2\left(\frac{1}{2}\right)^{-k}\cdot(1)$$

Put $-k = p$

$$\Rightarrow \quad y[n] = \sum_{p=\infty}^{1} 2\left(\frac{1}{2}\right)^{p} = 2\sum_{p=1}^{\infty}\left(\frac{1}{2}\right)^{p} = 2\left[\sum_{p=0}^{\infty}\left(\frac{1}{2}\right)^{p} - 1\right]$$

$$= 2\left[\frac{1}{1-\frac{1}{2}}\right] - 2 = 4 - 2 = 2$$

$$\therefore \quad \boxed{\begin{aligned} y[n] &= 2^{n+1} \quad ; \quad n < 0 \\ &= 2 \quad ; \quad n \geq 0 \end{aligned}}$$



Q.5 (d) Solution:

1. **$\overline{\text{BHE}}$/S7 :** The bus high enable $(\overline{\text{BHE}})$ pin is used in the 8086 to enable the most significant data bus (D_{15} – D_8) during a read/write operation. It goes low for the data transfer over D_{15}-D_8 and is used to derive chip selects of odd address memory bank or peripherals.

2. **MN/$\overline{\text{MX}}$**: The MN/$\overline{\text{MX}}$ pin is used to select either the minimum mode (single processor) or the maximum mode (multiprocessor) operation for the 8086. This is achieved by connecting this pin to either +5V directly (for minimum mode) or to the ground (for maximum mode).

3. **$\overline{\text{TEST}}$:** The $\overline{\text{TEST}}$ pin is an input that is tested by the WAIT instruction. If the $\overline{\text{TEST}}$ pin is at logic 0, the WAIT instruction functions as a NOP (no operation) instruction. If the $\overline{\text{TEST}}$ pin is at logic 1, the WAIT instruction waits for the $\overline{\text{TEST}}$ pin to become logic 0. This pin is often connected to the BUSY pin of the 8087 (numeric coprocessor) to perform floating-point operations.

4. **READY :** This input is used to insert wait states into the timing cycle of the 8086 such that it is extended by a number of clock periods. If the READY pin is at logic 1, it has no effect on the operation of the microprocessor. If it is at logic 0, the 8086 enters the wait state and remains idle. This pin is used to interface the slowly operating peripherals with the 8086.

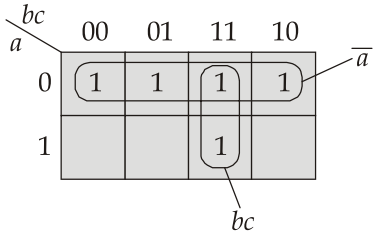
5. **RESET:** This input causes the 8086 to reset, if it is held at logic 1 for a minimum of four clocking periods. Whenever the 8086 is reset, CS and IP are initialized to FFFFH and 0000H, respectively, and all other registers are initialized to 0000H. This causes the 8086 to begin executing instructions from the memory address FFFF0H and IF flag is cleared.

6. **INTR:** The interrupt request (INTR) is a level-triggered hardware interrupt, which depends on the status of IF. When IF = 1, if INTR is held high (i.e., logic 1), the 8086 gets interrupted. When IF = 0, INTR is disabled.

Q.5 (e) Solution:

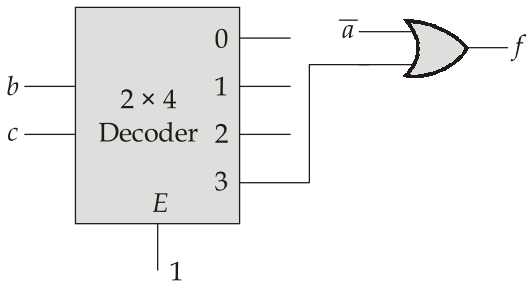
(i) Given, Boolean function, $f(a, b, c) = \Sigma m(0, 1, 2, 3, 7)$

By using three variable K-map,



$\therefore \quad f = \overline{a} + bc$

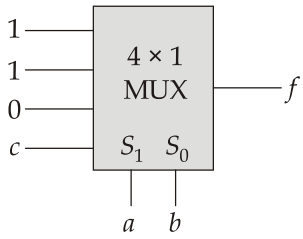
By using 2 × 4 decoder and OR gates.



(ii) Given Boolean function, $f(a, b, c) = \Sigma m(0, 1, 2, 3, 7)$

By using 4 × 1 multiplexer

input	I_0	I_1	I_2	I_3
$\overline{c}$	(0)	(2)	4	6
c	(1)	(3)	5	(7)
	1	1	0	c



$\therefore \quad f = \overline{a}\,\overline{b} + \overline{a}\,b + abc$

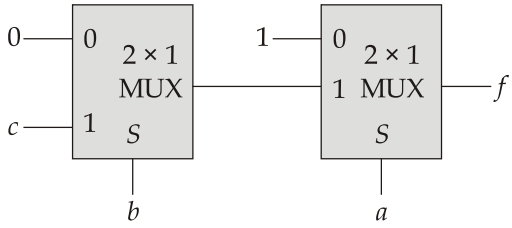
$= \overline{a}(\overline{b} + b) + abc$

$f = \overline{a} + bc$

(iii) $f(a, b, c) = \Sigma m(0, 1, 2, 3, 7)$

$\therefore \quad f = \overline{a} + bc$

By using a minimal network of 2 × 1 multiplexers



Q.6 (a) Solution:

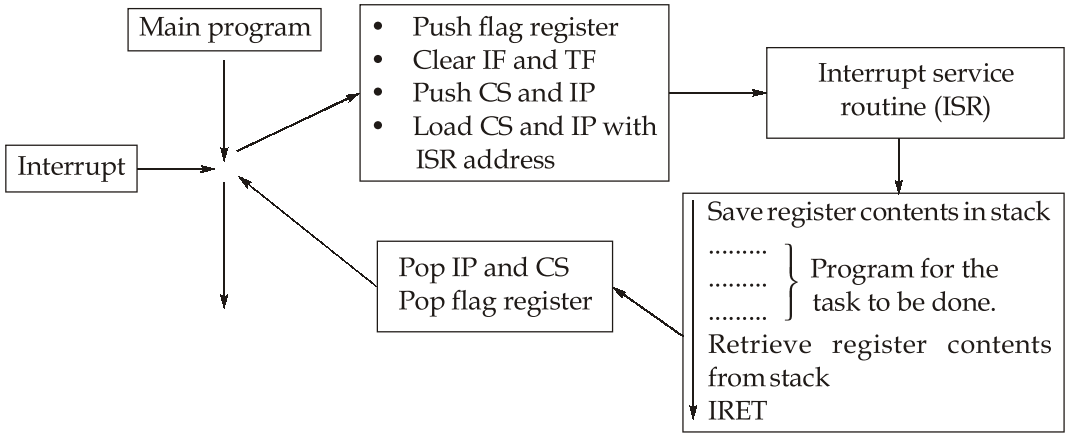
(i)

MOV AL, 29 H	;	Move the first BCD data to AL.
ADD AL, 98 H	;	Add the second BCD data to first BCD data.
DAA	;	Decimal-adjust AL to get the result in BCD form.
MOV BX, 2000 H	;	BX ← 2000 H
MOV DS, BX	;	Initialize DS with 2000 H (DS holds the base address for the data segment)
MOV [3000H], AL	;	Store the content of AL, which is the lower byte of the result in the memory.
JC L1	;	If the carry flag is 1, go to L1
MOV [3001 H], 00H	;	Store 00H in the memory, since the carry is 0.
JMP L2	;	Go to L2
L1 : MOV [3001H], 01H	;	Store 01H in the memory, since the carry is 1.
L2 : HLT	;	Stop

(ii) After executing each instruction in a program, the 8086 checks if any interrupt has been requested. If an interrupt has been requested, the 8086 processes it by performing the following series of steps:

1. Pushes the content of the 16-bit flag register onto the stack to preserve the status of the interrupt (IF) and trap flags (TF), by decrementing the stack pointer (SP) by 2.
2. Disables the INTR interrupt by clearing Interrupt Flag (IF) in the flag register.
3. Resets TF in the flag register, to disable the single step or trap interrupt.
4. Pushes the content of the code segment (CS) register onto the stack by decrementing SP by 2.

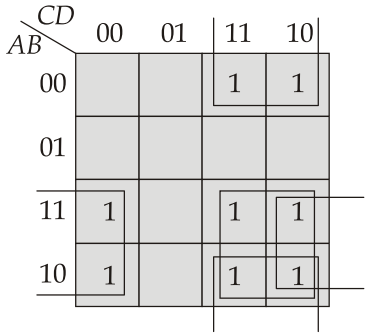
5. Pushes the content of the instruction pointer (IP) onto the stack by decrementing SP by 2.
6. Performs an indirect far jump to the start of the interrupt service routine (ISR) corresponding to the received interrupt. An IRET instruction at the end of the interrupt service routine returns execution to main program.



Processing of an interrupt by the 8086

Q.6 (b) Solution:

(i) Given, $f(A, B, C, D) = \Sigma m(2, 3, 8, 10, 11, 12, 14, 15)$

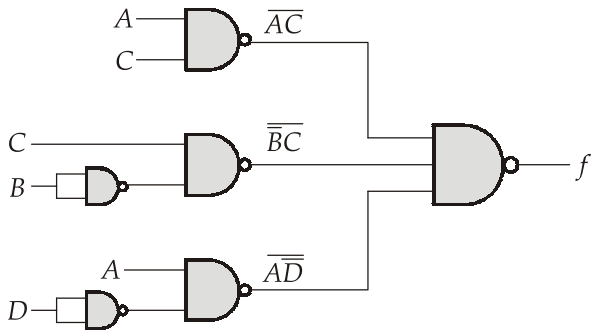


$\therefore \quad f = AC + A\bar{D} + \bar{B}C$

By using NAND gates, realizing above minimized expression,

$$\bar{\bar{f}} = \overline{\overline{AC + A\bar{D} + \bar{B}C}}$$

$$= \overline{\overline{AC} \cdot \overline{A\bar{D}} \cdot \overline{\bar{B}C}}$$



(ii) Truth table for the given logic circuit:

S_2	S_1	S_0	Y	Y
0	0	0	D_0	$\overline{S_0}D_0$
0	0	1	D_1	S_0D_1
0	1	0	D_2	$\overline{S_0}D_2$
0	1	1	D_3	S_0D_3
1	0	0	D_4	$\overline{S_0}D_4$
1	0	1	D_5	S_0D_5
1	1	0	D_6	$\overline{S_0}D_6$
1	1	1	D_7	S_0D_7

Let output of 2 × 1 MUXs are Y_0, Y_1, Y_2 and Y_3 respectively.

$\therefore$ $$Y_0 = \overline{S_0}D_0 + S_0D_1$$

$$Y_1 = \overline{S_0}D_2 + S_0D_3$$

$$Y_2 = \overline{S_0}D_4 + S_0D_5$$

$$Y_3 = \overline{S_0}D_6 + S_0D_7$$

$\therefore$ Output logic function

$$Y = \overline{S_2}\,\overline{S_1}\,Y_0 + \overline{S_2}\,S_1\,Y_1 + S_2\,\overline{S_1}\,Y_2 + S_2\,S_1\,Y_3$$

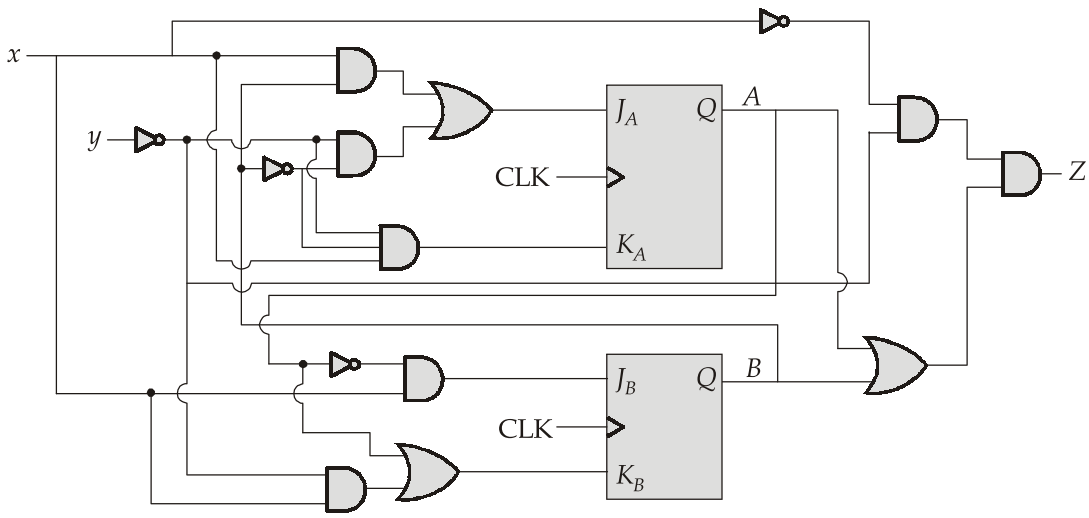
$$Y = \overline{S_2}\,\overline{S_1}(\overline{S_0}D_0 + S_0D_1) + \overline{S_2}\,S_1(\overline{S_0}\,D_2 + S_0D_3)$$

$$+ S_2S_1(\overline{S_0}\,D_4 + S_0D_5) + S_2S_1(\overline{S_0}\,D_6 + S_0D_7)$$

Q.6 (c) Solution:

Given, $J_A = Bx + \bar{B}\bar{y}$; $K_A = \bar{B}\,x\bar{y}$; $J_B = \bar{A}x$; $K_B = A + x\bar{y}$; $Z = A\bar{x}\,\bar{y} + B\bar{x}\,\bar{y}$

(i)



(ii)

Present state		Input		Next State		O/P	Flip flop inputs			
A	B	x	y	$A(t+1)$	$B(t+1)$	Z	J_A	K_A	J_B	K_B
0	0	0	0	1	0	0	1	0	0	0
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	1	0	1	1	1	1
0	0	1	1	0	1	0	0	0	1	0
0	1	0	0	0	1	0	0	0	0	0
0	1	0	1	0	1	0	0	0	0	0
0	1	1	0	1	0	0	1	0	1	1
0	1	1	1	1	1	1	1	0	1	0
1	0	0	0	1	0	0	1	0	0	1
1	0	0	1	1	0	0	0	0	0	1
1	0	1	0	0	0	0	1	1	0	1
1	0	1	1	1	0	1	0	0	0	1
1	1	0	0	1	0	0	0	0	0	1
1	1	0	1	1	0	0	0	0	0	1
1	1	1	0	1	0	0	1	0	0	1
1	1	1	1	1	0	1	1	0	0	1

(iii) For a JK flip flop,

$$Q(t+1) = J\bar{Q}(t) + \bar{K}Q(t)$$

State equations for A and B,

$$A(t+1) = J_A\bar{A} + \bar{K}_A A$$

$$= \left[Bx + \bar{B}\bar{y}\right]\bar{A} + \left(\overline{\bar{B}x\bar{y}}\right)A$$

$$= \bar{A}Bx + \bar{A}\bar{B}\bar{y} + A\left[B + \bar{x} + y\right]$$

$$= \bar{A}Bx + \bar{A}\bar{B}\bar{y} + AB + A\bar{x} + Ay$$

$$A(t+1) = \bar{A}\bar{B}\bar{y} + Ay + A\bar{x} + AB + Bx$$

$$B(t+1) = J_B\bar{B} + \bar{K}_B B$$

$$= \bar{A}x\bar{B} + \left(\overline{A + x\bar{y}}\right)B$$

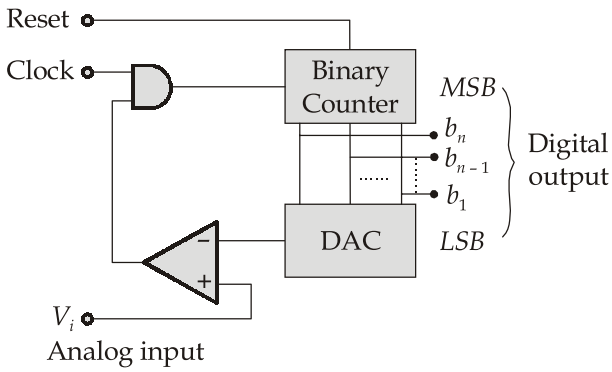
$$= \bar{A}x\bar{B} + B\left[\bar{A}\left(\overline{x\bar{y}}\right)\right]$$

$$= \bar{A}x\bar{B} + \bar{A}B(\bar{x} + y)$$

$$B(t+1) = \bar{A}\bar{B}x + \bar{A}B\bar{x} + \bar{A}By$$

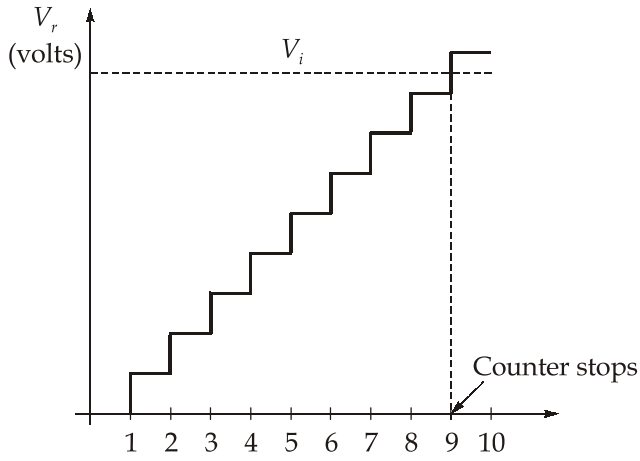
Q.7 (a) Solution:

(i) The counter type *ADC* is constructed using only one comparator with a variable reference voltage. The variable reference voltage can be obtained by a sequence or binary counter and a *DAC*. The block diagram for an n-bit counter type *ADC* is shown below,



The operation of the counter type ADC:

The n-bit binary counter is initially set to 0 by the reset switch which is normally active low. Therefore, the digital output is zero and the analog equivalent V_r is also 0. When reset signal is released (HIGH), the clock pulses gated through the AND gate are counted by the binary counter. The DAC converts the digital output to an analog voltage and connects it as the inverting input to the comparator. The output of the comparator enables the AND gate to pass the clock. The number of counted pulses increases with time and the analog output V_r from *DAC* is a rising staircase waveform.



The counting will continue until the reference voltage V_r equals and just rises more than V_i. Then the comparator output becomes low and this disables the AND gate from passing the clock. The counting steps at the instance $V_i < V_r$ and at that instant the digital output represents the analog input voltage V_i. Then the clock is inhibited, the counter stops its progress and the conversion is said to be complete. The numbers stored in the n-bit counter is the equivalent n-bit digital data for the given analog input voltage.

Advantages:

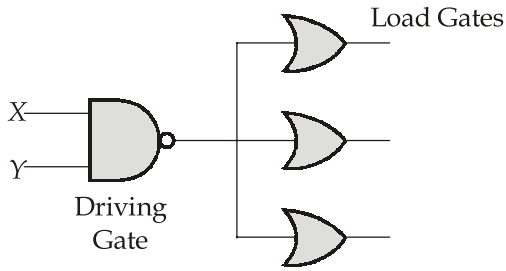
- The counter type *ADC* is very simple and needs less hardware compared to the simultaneous type *ADC*.
- This is suitable for digitizing applications with high resolution.

Disadvantages:

- In counter type A/D converter, the conversion time is very long, variable and proportional to the amplitude of the analog input voltage. Since the counter always counts from 0 through a normal sequence, a maximum of 2^n counts are required to convert a full-scale analog input voltage. Hence for an n-bit *ADC*,

the average conversion time is $\frac{2^n}{2} = 2^{n-1}$ times the clock period, which can be very long for large value of n.

(ii) Fan Out: When the output of a logic gate is connected to one or more inputs of other gates, a load on the driving gate is created. There is a limit to the number of load gate inputs that a given gate can drive. This limit is called the fan out of the gate.



1. High state noise margin $= V_{OH(\min)} - V_{IH(\min)}$
$= 2.7 - 2 = 0.7$ V

2. Low state noise margin $= V_{IL(\max)} - V_{OL(\max)}$
$= 0.8 - 0.4 = 0.4$ V

3. High state fan-out $= \frac{I_{OH}}{I_{IH}} = \frac{400}{25} = 16$

Low state fan-out $= \frac{I_{OL}}{I_{IL}} = \frac{8}{0.4} = 20$

Among these worst case fan-out will be taken, so number of NAND gate inputs that can be driven is 16.

Q.7 (b) Solution:

(i)
$$\begin{array}{r} 3441 \\ +4235 \\ \hline 7676 \end{array}$$

Since largest digit used is 7, it is true for any base > 7

(ii) $$\frac{142}{7} = 16$$

Assume base for the operation is b then

$$\frac{b^2 + 4b + 2}{7} = b + 6$$

$$b^2 - 3b - 40 = 0$$

$$\Rightarrow \quad b = 8$$

$$\therefore \quad \text{base (b)} = 8$$

(iii) $23 + 44 + 14 + 32 = 223$

Assume base for the operation is b

$$(2b + 3) + (4b + 4) + (b + 4) + (3b + 2) = (2b^2 + 2b + 3)$$

$$b^2 - 4b - 5 = 0$$

$$b = 5$$

(iv) $21 \times 16 = 366$

Assume base for the operation is b then,

$$(2b + 1)(b + 6) = 3b^2 + 6b + 6$$

$$2b^2 + 13b + 6 = 3b^2 + 6b + 6$$

$$b^2 - 7b = 0$$

$$b = 7$$

(v) $\dfrac{302}{20} = 12.1$

Assume base for operation is b,

$$\frac{3b^2 + 2}{2b} = b + 2 + \frac{1}{b}$$

$$\frac{3b^2 + 2}{2b} = \frac{b^2 + 2b + 1}{b}$$

$$3b^2 + 2 = 2b^2 + 4b + 2$$

$$b^2 - 4b = 0$$

$$b = 4$$

(vi) $\sqrt{51} = 6$

Assume base for the operation is b

$$5b + 1 = 36$$

$$5b = 35$$

$$b = 7$$

Q.7 (c) Solution:

(i) Let $H_M(e^{j\omega}) = |H(e^{j\omega})|$

The inverse DTFT of $H_M(e^{j\omega})$ is given by,

$$h_M[n] = \frac{1}{2\pi}\int_{-\pi}^{\pi} H_M\left(e^{j\omega}\right)e^{j\omega n}d\omega$$

$$= \frac{1}{2\pi}\int_{-\pi/3}^{\pi/3}(1)e^{j\omega n}d\omega = \frac{1}{2\pi}\left[\frac{e^{j\omega n}}{jn}\right]_{-\pi/3}^{\pi/3}$$

$$= \frac{1}{2\pi jn}\left[e^{j\pi/3 n} - e^{-j\pi/3 n}\right] = \frac{\sin \pi/3\, n}{\pi n}$$

If the group delay function,

$$\tau_g(\omega) = \frac{-d\underline{|H(e^{j\omega})}}{d\omega} = \alpha \text{ (where 'α' is a constant),}$$

then $\underline{|H(e^{j\omega})} = -\alpha\omega + \beta$ (where 'β' is a constant).

Since the impulse response $h[n]$ is real, the phase response $\underline{|H(e^{j\omega})}$ is an odd function, and hence for $\underline{|H(e^{j\omega})}$ to be an odd function, the constant $\beta = 0$.

Therefore, $\underline{|H(e^{j\omega})} = -\alpha\omega$

$$\Rightarrow H(e^{j\omega}) = \left|H(e^{j\omega})\right| e^{j\underline{|H(e^{j\omega})}} = H_M(e^{j\omega})\, e^{-j\alpha\omega}$$

Taking the inverse DTFT, we obtain

$$h[n] = h_M(n-\alpha)$$

$$\Rightarrow h[n] = \frac{\sin\frac{\pi}{3}(n-\alpha)}{\pi(n-\alpha)}$$

Given that, $\tau_g(\omega) = \frac{3}{2}$

$$\Rightarrow \frac{-d\underline{|H(e^{j\omega})}}{d\omega} = \frac{3}{2}$$

$$\Rightarrow \frac{-d}{d\omega}(-\alpha\omega) = \frac{3}{2}$$

$$\Rightarrow \alpha = \frac{3}{2}$$

$\therefore$ The real-valued impulse response of the filter is,

$$h[n] = \frac{\sin\frac{\pi}{3}\left(n-\frac{3}{2}\right)}{\pi\left(n-\frac{3}{2}\right)}$$

(ii) For a 5-coefficient FIR filter, if $x[n]$ is the input sequence, the output sequence is given by:

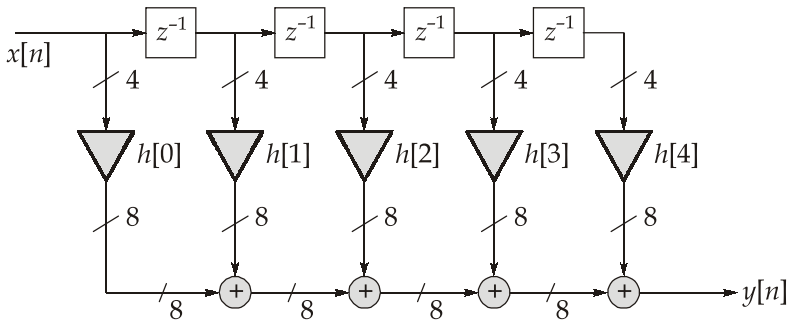
$y[n] = h[0]\,x[n] + h[1]\,x[n-1] + h[2]\,x[n-2] + h[3]\,x[n-3] + h[4]\,x[n-4]$

Taking z-transform on both the sides,

$$Y(z) = h[0]\,X(z) + h[1]\,z^{-1}\,X(z) + h[2]\,z^{-2}\,X(z) + h[3]\,z^{-3}\,X(z) + h[4]\,z^{-4}\,X(z)$$

$\therefore Y(z) = X(z)\,[h[0] + z^{-1}h[1] + z^{-2}h[2] + z^{-3}h[3] + z^{-4}\,h[4]]$

The above FIR filter can be implemented using direct-form structure as shown below:



The input $x[n]$ is available in 4-bit 2's complement representation, hence the input line is represented by a bus of 4-bit.

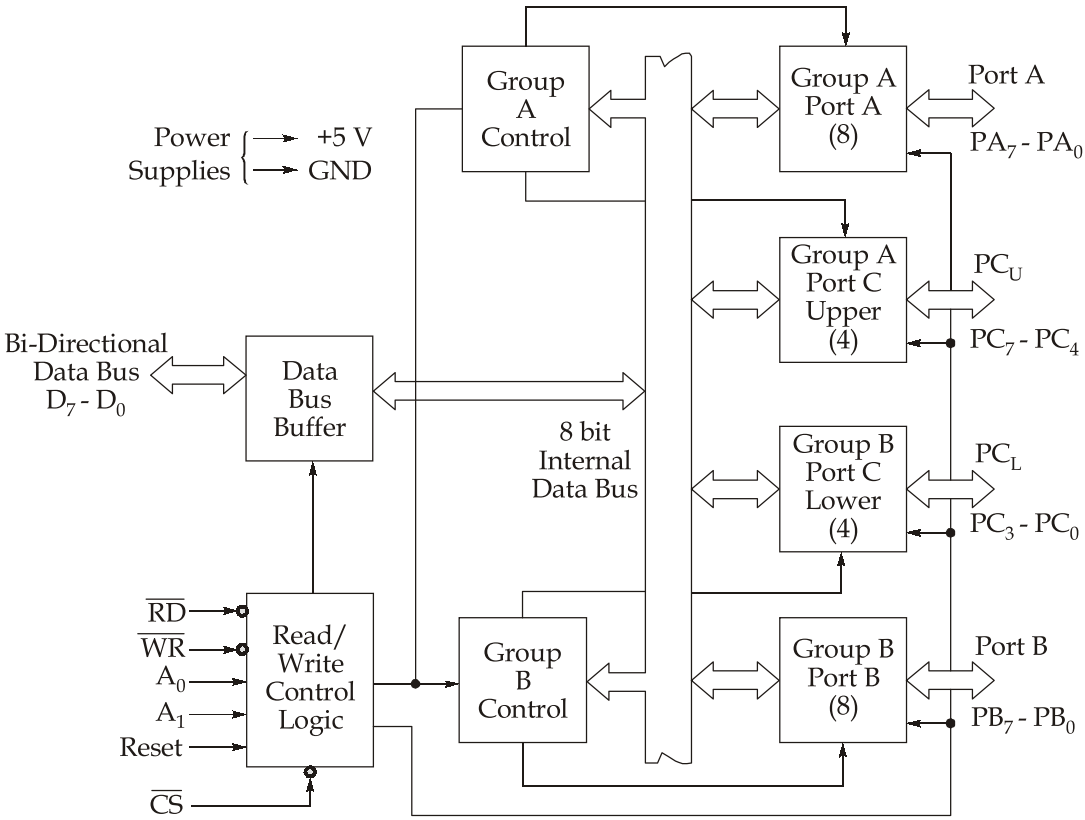
Assuming, the filter coefficients are also represented by 4 bits, the multiplier is then a 4 × 4 bit fixed point arithmetic multiplier giving 8-bit output. The adder adds two 8-bit fixed point words and hence, a 8-bit adder is required.

The above structure, therefore requires:

1. 4-delay elements: Each delay element consists of 4-bit PIPO register which produces a delay of 1 clock cycle.
2. Five 4 × 4 bit fixed point multipliers.
3. Four 8-bit adders.

Q.8 (a) Solution:

(i) Block diagram of 8255A

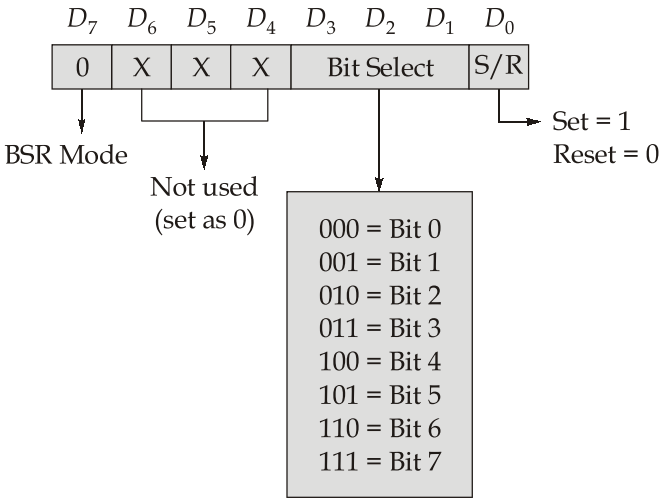


(ii) BSR mode is used to set or reset the bits of port C.

BSR mode in 8255A is concerned only with the eight bits of port C, which can be set or reset by writing an appropriate control word in the control register.

A control word with bit $D_7 = 0$ is recognised as a BSR control word, and it does not alter any previously transmitted control word with bit $D_7 = 1$; thus input-output operations of port A and port B are not affected by a BSR control word. In BSR mode, individual bits of port C can be used for applications such as an ON/OFF switch.

BSR control word :



(iii) The control word to set/reset the bits PC_7 and PC_3 can be obtained as below,

	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	
To set bit PC_7 =	0	0	0	0	1	1	1	1	= 0F H
To reset bit PC_7 =	0	0	0	0	1	1	1	0	= 0E H
To set bit PC_3 =	0	0	0	0	0	1	1	1	= 07 H
To reset bit PC_3 =	0	0	0	0	0	1	1	0	= 06 H

Given, the control register address = 83H.

Subroutine :

```
BSR :   MVI  A, 0FH   ;  Load byte in accumulator to set PC7
        OUT  83H      ;  PC7 set to 1
        MVI  A, 07H   ;  Load byte in accumulator to set PC3
        OUT  83H      ;  PC3 set to 1
        CALL DELAY    ;  Delay
        MVI  A, 06H   ;  Load byte in accumulator to reset PC3
        OUT  83H      ;  Reset PC3 to 0
        MVI  A, 0EH   ;  Load byte in accumulator to reset PC7
        OUT  83H      ;  Reset PC7 to 0
        RET
```

The delay program can be written as below:

```
DELAY :  MVI B, FFH
LOOP  :  DCR B
         JNZ LOOP
         RET
```

Q.8 (b) Solution:

(i) Given, fundamental period, $T = 3$

The Fourier series coefficient of an even function is also even. Thus,

From the fact 2, we can conclude that $x(t) = x(-t)$

We have, $$a_k = \frac{1}{T}\int_0^T x(t)e^{-jk\frac{2\pi}{T}t}$$

From the fact 1, $$a_k = a_{k+2}$$

then we require,

$$x(t)\cdot e^{-jk\times\frac{2\pi}{T}t} = x(t)e^{-j(k+2)\times\frac{2\pi}{T}t}$$

$$\Rightarrow \quad e^{j\frac{4\pi}{3}t} = 1 = e^{j2n\pi}$$

$$\Rightarrow \quad t = \frac{3n}{2}, \text{ where } n = 0, \pm1, \pm2, ...$$

Thus, $x(t)$ may have non-zero values only for $t = 0, \pm1.5, \pm3, \pm4.5.....$

From the fact 3,

$$\int_{-0.5}^{0.5} x(t)\,dt = 1,$$

We may conclude that $x(t) = \delta(t)$ for $-0.5 \le t \le 0.5$.

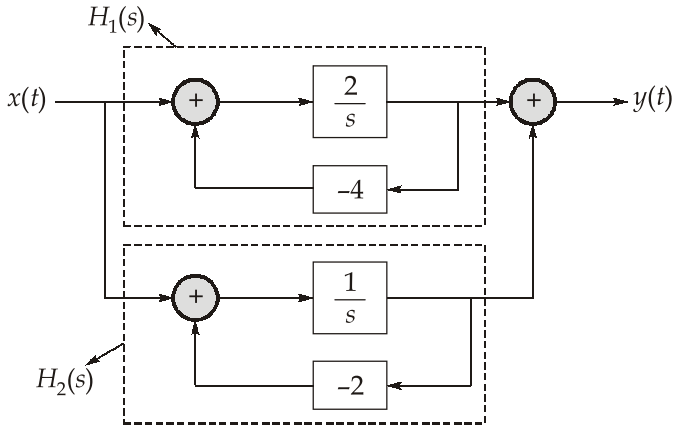
Also given $\int_{0.5}^{1.5} x(t)dt = 2$, thus we may conclude $x(t) = 2\delta\left(t - \frac{3}{2}\right)$ in the range $0.5 \le t \le 1.5$.

(Impulse must be within the given range)

Therefore, the periodic signal $x(t)$ with $T = 3$ can be written as,

$$x(t) = \sum_{k=-\infty}^{\infty} \delta(t-3k) + 2\sum_{k=-\infty}^{\infty} \delta\left(t-3k-\frac{3}{2}\right)$$

(ii) The given overall system may be treated as two feedback systems of the form shown below connected in parallel.



The transfer function of the upper feedback system is,

$$H_1(s) = \frac{\frac{2}{s}}{1+4\times\frac{2}{s}} = \frac{2}{s+8}$$

Similarly, the transfer function of the lower feedback system is,

$$H_2(s) = \frac{\frac{1}{s}}{1+2\left(\frac{1}{s}\right)} = \frac{1}{s+2}$$

The transfer function of the overall system is

$$H(s) = H_1(s) + H_2(s)$$

$$= \frac{2}{s+8} + \frac{1}{s+2}$$

$$H(s) = \frac{3s+12}{s^2+10s+16}$$

$\because$ $H(s) = \dfrac{Y(s)}{X(s)}$, we may write,

$$Y(s)[s^2 + 10s + 16] = X(s)[3s + 12]$$

$$s^2Y(s) + 10sY(s) + 16Y(s) = 3sX(s) + 12X(s)$$

Taking the inverse Laplace transform, we get the differential equation of the system as,

$$\frac{d^2y(t)}{dt^2} + 10\frac{dy(t)}{dt} + 16y(t) = 3\frac{dx(t)}{dt} + 12x(t)$$

Q.8 (c) Solution:

State table of Moore-machine,

Present State	Input X	Next State	Output Z
q_0	0	q_1	0
	1	q_2	0
q_1	0	q_1	0
	1	q_3	1
q_2	0	q_4	1
	1	q_2	0
q_3	0	q_4	1
	1	q_2	0
q_4	0	q_1	0
	1	q_3	1

Here, the states q_1 and q_4 are equivalent and the states q_2 and q_3 are equivalent.

So, reduced state table,

Present State	Input X	Next State	Output Z
q_0	0	q_1	0
	1	q_2	0
q_1	0	q_1	0
	1	q_2	1
q_2	0	q_1	1
	1	q_2	0

Excitation table: Let state $q_0 = 00$, $q_1 = 01$, $q_2 = 10$

Present State Q_1	Q_0	Input X	Next State Q_1^+	Q_0^+	J-K FF Inputs J_1	K_1	J_0	K_0	Output Z
0	0	0	0	1	0	X	1	X	0
0	0	1	1	0	1	X	0	X	0
0	1	0	0	1	0	X	X	0	0
0	1	1	1	0	1	X	X	1	1
1	0	0	0	1	X	1	1	X	1
1	0	1	1	0	X	0	0	X	0

K-Map for J_1:

Q_1 \ Q_0X	00	01	11	10
0	0	1	1	0
1	X	X	X	X

$J_1 = X$

K-Map for K_1:

Q_1 \ Q_0X	00	01	11	10
0	X	X	X	X
1	1	0	X	X

$K_1 = \overline{X}$

K-Map for J_0:

Q_1 \ Q_0X	00	01	11	10
0	1	0	X	X
1	1	0	X	X

$J_0 = \overline{X}$

K-Map for K_0:

Q_1 \ Q_0X	00	01	11	10
0	X	X	1	0
1	X	X	X	X

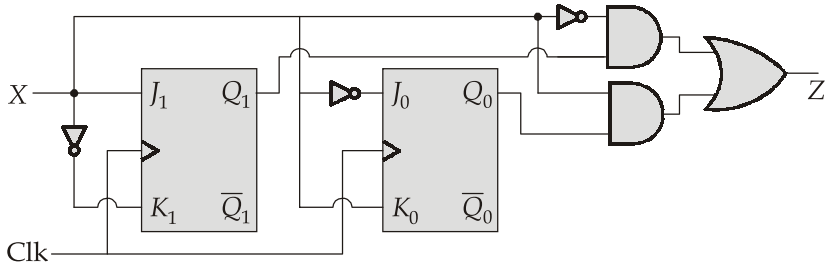
$K_1 = X$

K-Map for output Z:

Q_1 \ Q_0X	00	01	11	10
0	0	0	1	0
1	1	0	X	X

$Z = Q_0X + Q_1\overline{X}$

Circuit:



❍❍❍❍