MADE EASY
Leading Institute for ESE, GATE & PSUs

Detailed Solutions

ESE-2025 Mains Test Series

E & T Engineering Test No : 6

Section A : Advanced Electronics + Computer Organization and Architecture + Advanced Communication

Q.1 (a) Solution:

(i) The nominal etch time,

$$t_{\text{nominal}} = \frac{d_{ox}}{r_{ox}} = \frac{0.6}{0.4}\text{minute}$$

$$= 1.5 \text{ minutes.}$$

The overetch is done to make sure all the oxide is etched for the worst case condition; that means for the thickest oxide and the slowest etch rate.

$$d_{ox(\max)} = 0.6(1.05) = 0.63\ \mu\text{m}$$

$$r_{ox(\min)} = 0.4(0.95) = 0.38\ \mu\text{m}$$

The time taken to etch in the worst case condition,

$$t_{\max} = \frac{d_{ox(\max)}}{r_{ox(\min)}} = \frac{0.63}{0.38} = \frac{63}{38}\text{minutes}$$

The amount of overetch, in% time, can be given by

$$= \frac{t_{\max} - t_{\text{nominal}}}{t_{\text{nominal}}} \times 100$$

$$= \frac{\frac{63}{38} - 1.5}{1.5} \times 100 = 10.53\%$$

(ii) The maximum amount of Si that will be etched will occur when the oxide thickness is minimum and it is etched at the fastest etch rate. We have,

$$d_{ox(\min)} = 0.6(0.95) = 0.57\ \mu m$$
$$r_{ox(\max)} = 0.4 \times 1.05 = 0.42\ \mu m/\text{minute}$$

The time required, in this case, to etch the complete oxide layer is,

$$t_{ox(\min)} = \frac{d_{ox(\min)}}{r_{ox(\max)}} = \frac{0.57}{0.42} = \frac{57}{42}\text{minutes}$$

The etching is done for a time $t_{\max}$ obtained in part (i). The maximum duration of time that the Si is exposed to the etch is,

$$t_{Si(\max)} = t_{\max} - t_{ox(\min)} = \frac{63}{38} - \frac{57}{42}$$
$$= \frac{40}{133}\text{minutes}$$

For a maximum of 0.5 nm of Si to be etched,

$$t_{Si(\max)} \times r_{si} = 0.5\ \text{nm}$$

So,
$$r_{Si} = \frac{0.5\times133}{40}\text{nm/minute} = 1.6625\ \text{nm/minute}$$
$$r_{ox} = 0.4\ \mu m/\text{minute} = 400\ \text{nm/minute}$$

The required etch selectivity of oxide with respect to the Si is

$$S = \frac{r_{ox}}{r_{Si}} = \frac{400}{1.6625} = 240.601$$

Q.1 (b) Solution:

The Fibonacci series is the sequence where each number is the sum of the previous two numbers of the sequence.

C Program:

```
#include<stdio.h>
int main ()
{
unsigned int fib[100]; //Use integer array to store the fibonacci numbers
fib[0] = 1;
fib[1] = 1;
for (int i = 2; i < 100; i++)
```

```
{
fib[i] = fib[i – 1] + fib[i – 2]; //Generate first 100 Fibonacci numbers.
}
for (int i = 0; i < 100; i++)
{
printf ("%d\n", i, fib[i]); //Print the first 100 Fibonacci numbers
}
return 0;
}
```

Q.1 (c) Solution:

(i) Since the carrier frequency is 1 GHz. Thus, the wavelength λ is:

$$\lambda = \frac{c}{f} = \frac{3\times10^8}{10^9}\text{m} = \frac{3}{10}\text{m}$$

Hence, the Doppler frequency shift is:

$$f_D = \pm\frac{v}{\lambda} = \pm\frac{100\text{ km/hr}}{\frac{3}{10}\text{m}} = \pm\frac{100\times10^3\times10}{3\times3600}\text{Hz}$$

$$= \pm 92.5926\text{ Hz}$$

The plus sign hold when the vehicle travels towards the transmitter whereas the minus sign holds when the vehicle moves away from the transmitter.

(ii) The maximum difference in the Doppler frequency shift to track the vehicle movement in both directions, when the vehicle travels at speed 100 km/hr and f = 1 GHz, is

$$\Delta f_{D\max} = 2f_D = 185.1852\text{ Hz}$$

This should be the bandwidth of the Doppler frequency tracking loop.

(iii) For a transmitted signal bandwidth of 2 MHz centered at 1 GHz, the maximum Doppler frequency shift is obtained when f = 1 GHz +1 MHz and the vehicle moves towards the transmitter. In this case:

$$\lambda_{\min} = \frac{3\times10^8}{10^9+10^6}\text{m} = 0.2997\text{ m}$$

and therefore:

$$f_{D\max} = \frac{100\times10^3}{0.2997\times3600} = 92.6853\text{ Hz}$$

The minimum Doppler frequency is obtained when $f = 1$ GHz – 1 MHz. In this case,

$$\lambda_{max} = \frac{3\times10^8}{10^9 - 10^6} = 0.30003 \text{ m}$$

Thus, $$f_{D\min} = \frac{100\times10^3}{0.30003\times3600} = 92.5833 \text{ Hz}$$

Thus, the Doppler frequency spread is

$$B_d = f_{D\max} - f_{D\min} = 92.6853 - 92.5833 = 0.102 \text{ Hz}$$

Q.1 (d) Solution:

Given data, $h = 1200$ km

Transmitter frequency is 3.56 GHz.

(i) Velocity of satellite in orbit.

$$V_s = \sqrt{\frac{GM}{r(\text{km})}} = \sqrt{\frac{3.98\times10^{11}}{6378+1200}}$$

$$V_s = 7247.1 \text{ m/sec} = 7.247 \text{ km/sec}$$

(ii) The component of velocity towards an observer in the plane of the orbit as the satellite appears over the horizon is given by $V_r = V_s \cos\theta$, where θ is the angle between the satellite velocity vector and the direction of the observer at the satellite.

The angle θ can be found from simple geometry to be

$$\cos\theta = \frac{r_e}{r_e + h} = \frac{6378}{7578} = 0.8416$$

Hence, the component of satellite velocity toward the observer is

$$V_r = V_s \cos\theta$$
$$= 7.247 \times 0.8416$$
$$= 6.099 \text{ km/sec}$$

(iii) For a transmitter frequency of 3.56 GHz, the Doppler shift in the received signal is given as

$$\Delta f = \frac{V_r}{\lambda} = \frac{6099}{3\times10^8}\times3.56\times10^9$$

$$\Delta f = 72.374 \text{ kHz}$$

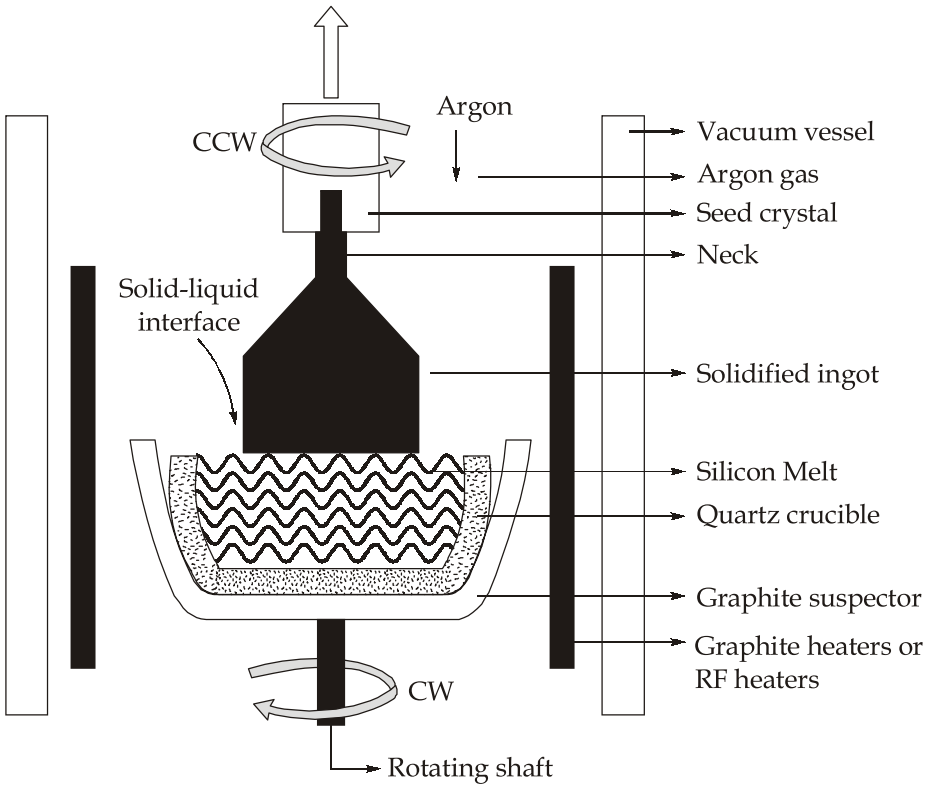
(iv) A Ka-band transmitter with frequency 25 GHz has a wavelength $\lambda = \frac{3\times10^8}{25\times10^9} = 0.012$ m. The corresponding Doppler shift at the receiver, is

$$\Delta f = \frac{V_r}{\lambda} = \frac{6099}{0.012}$$

$$\Delta f = 508.250 \text{ kHz}$$

Doppler shift at Ka band with a LEO satellite can be very large and requires a fast frequency tracking receiver.

Q.1 (e) Solution:



The Czochralski process (CZ) also known as "Crystal pulling" is a method of crystal growth used to obtain single silicon crystals.

It uses an apparatus called a crystal puller which has 3 main components:

- Furnace which includes a fused-silica (SiO_2) crucible, a graphite susceptor, a rotation mechanism, a heating element and a power supply.
- Crystal pulling mechanism which includes a seed holder and a rotation mechanism.
- Ambient control which includes a gas source (such as argon), a flow control and an exhaust system.

Electronic grade silicon (EGS) is placed in the crucible and the furnace is heated above the melting point of silicon. A suitably oriented seed crystal is suspended over the crucible in a seed holder. The seed is now inserted into the melt, part of it melts, but the tip of the remaining seed crystal still touches the liquid surface. It is then slowly withdrawn and progressive freezing at the solid-liquid interface yields a large single crystal.

During crystal growth, a known amount of dopant is added to the melt to obtain the desired doping concentration in the grown crystal. Boron and phosphorous are usually added.

Advantages:

This method is used to grow large single crystals. Thus, it is used extensively in the semiconductor industry.

There is no direct contact between the crucible walls and the crystal which helps to produce unstressed single crystal.

Drawback: The drawback is that at the high temperature, the inner liner of the crucible also starts melting and has to be replaced periodically.

Q.2 (a) Solution:

Given data, Junction depth, $x_j = 0.5\ \mu m$

Doping concentration of substrate,

$$N_B = 10^{17}/cm^3$$

Solid solubility limit of boron,

$$N_0 = 2.6 \times 10^{20}\ atoms/cm^3$$

Surface concentration of boron for drive-in step,

$$N_S = 5 \times 10^{19}\ atoms/cm^3$$

Temperature used for pre-deposition

$$T_1 = 1100 + 273\ K = 1373\ K$$

Temperature used for drive-in

$$T_2 = 1200 + 273 = 1473\ K$$

Diffusion constant for boron, $D_0 = 11.8\ cm^2/sec$

Activation energy for boron, $E_a = 4.36\ eV$

Useful relations:

The concentration profile of the diffused atoms inside the substrate for the pre-deposition process can be given as,

$$N(x, t) = N_0 erfc\left[\frac{x}{2\sqrt{Dt}}\right]$$

where, $\text{erfc}(t)$ = Complementary error function

D = Diffusivity of boron = $D_e e^{-E_a/kT}$

t = process time for pre-deposition and

x = depth from the surface of the substrate

The concentration profile of the diffused atoms inside the substrate for drive-in process can be given as

$$N(x, t) = \frac{Q_0}{\sqrt{\pi Dt}} e^{\left(-\frac{x^2}{4Dt}\right)} = N_S e^{\left(-\frac{x^2}{4Dt}\right)}$$

where, Q_0 = Amount of solute per unit area present on the surface prior to drive-in.

$$Q_0 = 2N_0\sqrt{\frac{D_1 t_1}{\pi}}$$

Assume, D_1 = Diffusivity of the boron for pre-deposition process, t_1 = pre-deposition time, D_2 = Diffusivity of Boron for drive-in process and t_2 = drive-in time. After drive-in process, at junction depth (x_j), the concentration of solute (or diffused atoms) is equal to the initial doping concentration of the subrate (or background concentration).

i.e., $$\frac{Q_0}{\sqrt{\pi D_2 t_2}} e^{\left(-\frac{x_j^2}{4D_2 t_2}\right)} = N_B$$

Finding the values of D_1 and D_2:

For pre-deposition process,

$$D_1 = D_0 e^{\left(-\frac{E_a}{kT_1}\right)} = 11.8 e^{\left(\frac{-4.36}{8.62\times10^{-5}\times1373}\right)}$$

$$= 1.182 \times 10^{-15} \text{ cm}^2/\text{sec}$$

For drive-in process,

$$D_2 = D_0 e^{-\frac{E_a}{KT_2}} = 11.8 e^{\left(\frac{-4.36}{8.62\times10^{-5}\times1473}\right)}$$

$$= 1.442 \times 10^{-14} \text{ cm}^2/\text{sec}$$

Finding the drive-in time (t_2):

For drive-in process,

$$N(x,t) = \frac{Q_0}{\sqrt{\pi D_2 t_2}} e^{-\left(\frac{x^2}{4D_2t_2}\right)} = N_s e^{\left(\frac{-x^2}{4D_2t_2}\right)}$$

At $x = x_j$, $N(x,t) = N_B$

So, $N_s e^{\frac{-x_j^2}{4D_2t_2}} = N_B$

$$\frac{x_j^2}{4D_2t_2} = \ln\left(\frac{N_S}{N_B}\right)$$

$$t_2 = \frac{x_j^2}{4D_2 \ln\left(\frac{N_S}{N_B}\right)} \quad \text{...(i)}$$

By substituting the values of x_j, D_2, N_S and N_B in equation (i), we get

$$t_2 = \frac{(0.5\times10^{-4})^2}{4\times1.442\times10^{-14}\ln\left(\frac{5\times10^{19}}{10^{17}}\right)} \text{seconds}$$

$t_2 = 6974.306$ seconds

$= 116$ minutes 14 seconds

$= 1$ hr 56 minutes 14 seconds

Finding the process time of pre-deposition (t_1):

First, by using values of t_2 and N_S, we can calculate the value of Q_0 as follows,

$$N_S = \frac{Q_0}{\sqrt{\pi D_2 t_2}}$$

So, $Q_0 = N_S\sqrt{\pi D_2 t_2} = 5\times10^{19}\sqrt{\pi\times1.442\times10^{-14}\times6974.306}$

$= 8.887 \times 10^{14}/\text{cm}^3$

The value of Q_0 depends on pre-deposition process time, diffusivity, solid solubility limits as

$$Q_0 = 2N_0\sqrt{\frac{D_1t_1}{\pi}}$$

So, $$t_1 = \frac{\pi}{D_1}\left(\frac{Q_0}{2N_0}\right)^2 \quad ...(ii)$$

By substituting the values of D_1, Q_0 and N_0 in equation (ii), we get,

$$t_1 = \frac{\pi}{1.182\times10^{-15}}\left(\frac{8.887\times10^{14}}{2\times2.6\times10^{20}}\right)^2$$

$$= 7763.115 \text{ seconds}$$

$$\cong 7763 \text{ seconds}$$

$$= 2 \text{ hours } 9 \text{ minutes } 23 \text{ seconds}$$

So, the diffusion times required for both pre-deposition and drive-in processes are as follows:

Diffusion time of pre-deposition, $t_1 = 7763$ seconds

Diffusion time of drive-in, $t_2 = 6974$ seconds

Q.2 (b) Solution:

(i) Gyro frequency (also called electron cyclotron frequency) is the frequency at which free electrons spiral (or gyrate) around magnetic field lines in the Earth's ionosphere due to the lorentz force.

$$f_g = \frac{eB}{2\pi m}$$

where:

f_g = gyro frequency in Hz

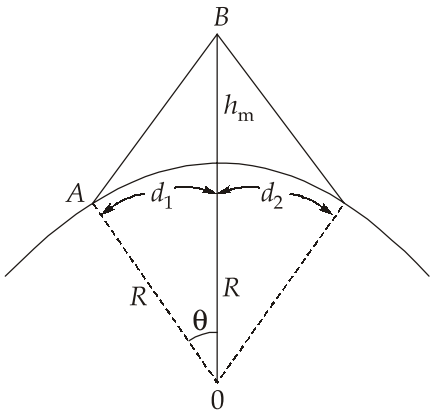
e = charge of electron (1.6×10^{-19}C)

B = magnetic field strength (in tesla)

m = mass of electron (9.1×10^{-31} kg)

- When a radio wave has a frequency near the gyro frequency, it resonates with spiral motion of electrons.
- This causes increased absorption and scattering of the wave energy by the electrons.
- The result is severe signal attenuation, making such frequencies inefficient or unusable for long range ionospheric communication.

The F-layer, acts as a reflective layer at a height h_m above the earth's surface.



The length of arc, $d_1 = d_2 = R\theta$

Maximum range in a single hop transmission is given by

$$d = d_1 + d_2 = 2R\theta$$

$$= 2R\cos^{-1}\left(\frac{R}{R+h_m}\right)$$

where R = radius of earth

h_m = height of the ionospheric layer

$$= 2\times 6370\cos^{-1}\left(\frac{6370}{6370+400}\right)$$

$$= 12740\cos^{-1}(0.9409)$$

$$= 12740\times 19.8°\times\frac{\pi}{180}$$

$$= 12740\times\frac{62.172}{180}$$

$$= 12740 \times 0.3454$$

$$d = 4400.396 \text{ km}$$

(ii) Given:

h = 350 km; $\mu = 0.8$, f_{muf} = 20 MHz

f = 15 MHz

The refractive index of the ionospheric layer is given by,

$$n = \sqrt{1-\frac{81N_{max}}{f^2}}$$

$$0.8 = \sqrt{1-\frac{81N_{max}}{f^2}}$$

$$0.64 = 1-\frac{81N_{max}}{f^2}$$

$$\frac{81N_{max}}{f^2} = 1 - 0.64 = 0.36$$

$$N_{max} = \frac{0.36\times f^2}{81}$$

$$= \frac{0.36(15\times10^6)^2}{81}$$

$$= 1 \times 10^{12}\ m^{-3}$$

Hence, critical frequency, $f_c = 9\sqrt{N_{max}} = 9\times10^6\ Hz$

When earth is flat, the range of propagation is given by

$$D_{skip} = 2h\sqrt{\left(\frac{f_{muf}}{f_c}\right)^2 - 1}$$

$$= 2\times350\sqrt{\left(\frac{20\times10^6}{9\times10^6}\right)^2 - 1}$$

$$= 700\sqrt{(2.22)^2 - 1}$$

$$= 700 \times 1.982$$

$$D_{skip} = 1387.41\ km$$

Q.2 (c) Solution:

(i) $$\%\text{Time CPU is blocked} = \frac{\text{Transfer time}}{\text{Transfer time + Preparation Time}}$$

Preparation time: The device is providing data at 1 Kbps. Since the size of data transfer is 16 bytes, thus to prepare 16 bytes, the time required can be calculated as,

10^3 bytes $\rightarrow$ 1 sec

1 byte $\rightarrow$ 1 m sec

So, 16 bytes $\rightarrow$ 16 msec

$$\text{Transfer time} = (10 \text{ cycles for initialization}) + (2 \times 16 \text{ cycles for transfer})$$
$$= (10 + 32) \text{ cycles}$$
$$= 42 \text{ cycles}$$
$$\text{1 cycle time} = \frac{1}{f_{clk}} = \frac{1}{(400 \times 10^3)} = 0.0025 \text{ msec}$$

So, $42 \text{ cycles} = 42 \times 0.0025 \text{ msec}$
$$= 0.105 \text{ msec}$$

$$\therefore \quad \%\text{ of time CPU blocked} = \frac{0.105}{(16 + 0.105)} = \frac{0.105}{16.105}$$
$$= 0.00652 \times 100 = 0.652\%$$

(ii) 1. FIFO: In FIFO, the cache block which is having the longest time stamp is replaced in case of a page miss.

4-miss	5-miss	7-miss	12-miss	4-hit	5-hit	13-miss	4-miss	5-miss	7-miss
			12	12	12	12	12	12	7
		7	7	7	7	7	7	5	5
	5	5	5	5	5	5	4	4	4
4	4	4	4	4	4	13	13	13	13

$$\text{Hit ratio} = \frac{2}{10} = 0.2$$

2. LRU: Least Recently Used (LRU) is a page replacement technique that replaces the least recently used page first i.e. the page that hasn't been used for the longest time from the frame in case of a page miss.

4-miss	5-miss	7-miss	12-miss	4-hit	5-hit	13-miss	4-hit	5-hit	7-miss
			12	12	12	12	12	12	7
		7	7	7	7	13	13	13	13
	5	5	5	5	5	5	5	5	5
4	4	4	4	4	4	4	4	4	4

$$\text{Hit ratio} = \frac{4}{10} = 0.4$$

Q.3 (a) Solution:

(i) Number of instructions = 250

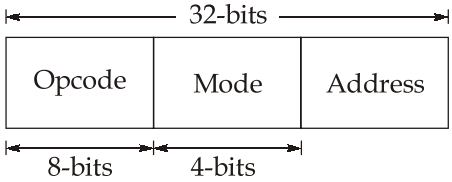
Thus, opcode bits = $\lceil \log_2 250 \rceil = 8$ bits

Number of modes = 10

Mode bits = $\lceil \log_2 10 \rceil = 4$ bits

Since each instruction is stored in one word, hence the size of the instruction is 32 bits.

Instruction:



Address bits = 32 – (8 + 4)

= 32 – 12

= 20 bits

Maximum allowable memory size = $2^{20} \times 32$ bits

= $2^{20} \times 4$ bytes

= 4 MB

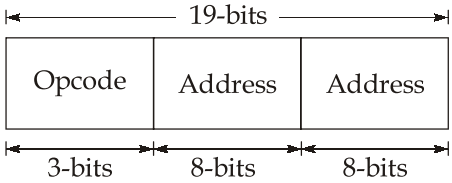
(ii) Given, Length of the instruction = 19 bits

Size of address field = 8 bits

For two address instructions,

Number of opcode bits = 19 – (2 × 8) = 3 bits

Instruction:



Total number of opcodes = $2^3 = 8$

Number of two address instructions = 6

Remaining opcodes for one address instruction

= (8 – 6) = 2

$\therefore$ Maximum possible one-address instructions = $2 \times 2^8 = 2^9 = 512$

Assume K as the number of one address instructions, then remaining opcodes for zero-address instructions = 512 – K

$\therefore$ Possible zero-address instruction = $(512 - K)2^8$

Given, $65536 = (512 - K)2^8$

$\Rightarrow$ $65536 = 131072 - 256\,K$

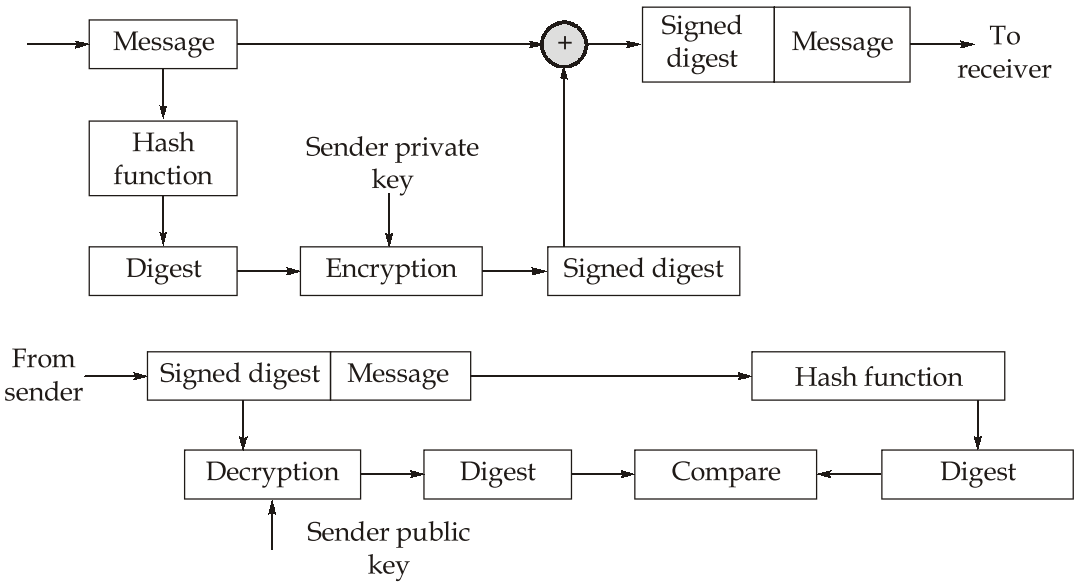
$\Rightarrow$ $K = \dfrac{65536}{256} = 256$

Q.3 (b) Solution:

A digital signature is a type of electronic signature that uses cryptographic techniques such as public and private key pairs to verify the authenticity and integrity of a message or document. In digital signature, the private and public keys of the sender are used. The sender uses it's private key and the receiver uses the public key of the sender to verify the signature's authenticity.

The public key is very inefficient in a cryptosystem if we are dealing with long messages. In a digital signature system our messages are normally long, but we have to use public keys. The solution is not to sign the message itself, instead we sign a digest of the message. A carefully selected message digest has a one-to-one relationship with the message. The sender can sign the message digest and the receiver can verify the message digest.

The digest is made out of the message at sender's site using hash function. The digest then goes through the signing process using sender's private key which is then send to the receiver along with the message.



At receiver, using the same public hash function, a digest is first created out from the message. Also, the signed digest is decrypted using the sender's public key. The message digest computed by receiver and the message digest (got by decryption) should be same for ensuring integrity.

Assurances provided by Digital Signature:

A digital signature can provide: message integrity, message authentication and non repudiation. A digital signature scheme does not provide confidential communication. If confidentiality is required, the message and the signature must be encrypted using either a secret-key or public-key cryptographic technique.

Message Integrity:

A digital signature ensures that the message has not been altered after it was signed. This is primarily achieved through the cryptographic hash function. Any change to the message, even a single bit, will result in a different message digest, and thus the signature authentication will fail.

Message Authentication:

The digital signature is designed in such a way that only the sender, with knowledge of the private key, can produce a valid signature. When a recipient verifies a digital signature using the sender's public key, a successful verification confirms that the message truly originated from the sender.

Non-Repudiation:

Digital signatures also provide non-repudiation, which means that the sender of the message cannot later deny having sent the message. This is because the digital signature is unique to the sender and is difficult to forge.

Q.3 (c) Solution:

(i) Given,

$$N = 10^{18} \text{ atoms/cm}^3$$

$$J = 5\ \mu\text{A/cm}^2$$

$$t = 20 \text{ minutes} = 20 \times 60 \text{ sec} = 1200 \text{ sec}$$

$$R_p = 2\ \mu\text{m}$$

$$\Delta R_p = 0.5\ \mu\text{m}$$

$$\text{Implantation Dose, } Q_0 = \frac{Jt}{q}$$

$$Q_0 = \frac{5\times10^{-6}\times1200}{1.6\times10^{-19}} = 3.75\times10^{16}\,\text{cm}^{-2}$$

The peak concentration is given by,

$$N_p = \frac{Q_0}{\Delta R_p \sqrt{2\pi}}$$

$$= \frac{3.75\times10^{16}}{0.5\times10^{-4}\sqrt{2\pi}}$$

$$\simeq 3 \times 10^{20}\ \text{cm}^{-3}$$

(ii) Given, $N = 5 \Rightarrow \omega_k = \frac{2\pi k}{N} = \frac{2\pi k}{5}$

$\therefore$ $\alpha = \frac{N-1}{2} = \frac{5-1}{2} = \frac{4}{2} = 2$

Normalization: $\omega_{c1} = \frac{2\pi f_{c1}}{f_s} = \frac{2\pi\times2}{8} = \frac{\pi}{2}\text{rad/s}$

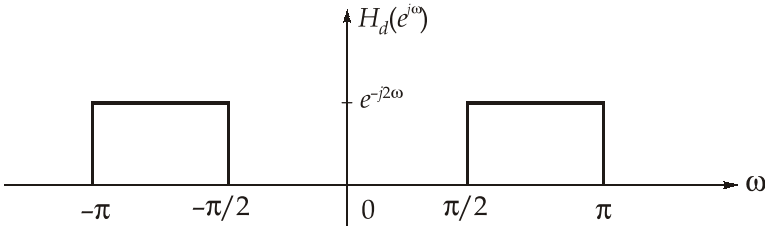
$$\omega_{c2} = \frac{2\pi f_{c2}}{f_s} = \frac{2\pi\times4}{8} = \pi\ \text{rad/s}$$

The given filter can be expressed as,

$$H_d(e^{j\omega}) = \begin{cases} e^{-j\alpha\omega}; & \frac{\pi}{2} \le \omega \le \pi \\ 0 \ ; & \text{otherwise} \end{cases}$$

$\Rightarrow$

$$H_d(e^{j\omega}) = \begin{cases} e^{-j2\omega}; & \frac{\pi}{2} \le \omega \le \pi \\ 0 \ ; & \text{otherwise} \end{cases}$$



Converting continuous spectrum to discrete spectrum,

$$H_d(e^{j\omega_k}) = H(k) = \begin{cases} e^{-j2\left(\frac{2\pi k}{5}\right)}; & \frac{\pi}{2} \le \frac{2\pi k}{5} \le \pi \\ & \frac{5}{4} \le k \le \frac{5}{2} \Rightarrow k = 2 \\ 0 \ ; & \text{otherwise} \end{cases}$$

For frequency sampling method, the impulse response is,

$$h[n] = \frac{1}{N}\left[H(0)+2\sum_{k=1}^{\frac{N-1}{2}}\text{Re}\,al\left(H(k)e^{\frac{j2\pi}{N}kn}\right)\right]; \text{odd } 'N'$$

For Band pass filter, $H(0) = 0$ and we have $k = 2$

$$\Rightarrow \quad h[n] = \frac{2}{N}\sum_{k=1}^{\frac{N-1}{2}}\text{Real}\left[H(k)\cdot e^{\frac{j2\pi}{N}kn}\right]; k=2$$

$$\Rightarrow \quad h[n] = \frac{2}{5}\sum_{k=1}^{\frac{5-1}{2}}\text{Real}\left[e^{\frac{-j4\pi k}{5}}\cdot e^{\frac{j2\pi}{5}kn}\right]; k=2$$

$$= \frac{2}{5}\sum_{k=1}^{2}\text{Real}\left[e^{\frac{-j2\pi}{5}(2-n)k}\right]; k=2$$

$$= \frac{2}{5}\sum_{k=1}^{2}\cos\frac{2\pi}{5}(2-n)k\ ; k=2$$

$$= \frac{2}{5}\cos\frac{2\pi}{5}(2-n)2$$

$$\Rightarrow \quad h[n] = \frac{2}{5}\cos\frac{4\pi}{5}(2-n)$$

$\therefore$ The filter coefficients for $N = 5$ are,

$$h[0] = \frac{2}{5}\cos\left[\frac{4\pi}{5}2\right] = 0.12361$$

$$h[1] = \frac{2}{5}\cos\left[\frac{4\pi}{5}(1)\right] = -0.3236$$

$$h[2] = \frac{2}{5}\cos\left[\frac{4\pi}{5}(0)\right] = 0.4$$

$$h[3] = \frac{2}{5}\cos\left[\frac{4\pi}{5}(-1)\right] = -0.3236$$

$$h[4] = \frac{2}{5}\cos\left[\frac{4\pi}{5}(-2)\right] = 0.12361$$

Q.4 (a) Solution:

For path loss exponent $n = 4$, the power received at a distance D from a transmitter can be given by,

$$P_r = P_0\left(\frac{D}{d_0}\right)^{-4}$$

For an N-cell reuse pattern, Co-channel reuse ratio is given by

$$Q = \frac{D}{r} = \sqrt{3N}$$

So, the power interference from the nearest co-channel is given by

$$P_r = P_0\left(\frac{D}{d_0}\right)^{-4}\Bigg|_{D=\sqrt{3N}r} = P_0\left(\frac{\sqrt{3N}\times r}{d_0}\right)^{-4}$$

P_0 = 1 mW for $d_0 = 1.5$ m

N = Number of cells in a cluster

r = Radius of each cell

Co-channel interference must be less than –100 dBm which is equal to $10^{-100/10}$ mW = 10^{-13} W.

So, $$10^{-3}(\sqrt{3N}\,r)^{-4} < 1.5 \times 10^{-13}$$

$$(\sqrt{3N}\,r)^{-4} < 1.5 \times 10^{-10}$$

$$(\sqrt{3N}\,r)^{4} > 0.67 \times 10^{10}$$

$$r > \frac{286.1}{\sqrt{3N}}\text{m}$$

For a seven-cell cluster,

$$r > \frac{286.1}{\sqrt{21}}\text{m} \quad \Rightarrow \quad r > 62.43\text{ m}$$

For a four-cell cluster,

$$r > \frac{286.1}{\sqrt{12}}\text{m} \Rightarrow r > 82.59\text{ m}$$

Q.4 (b) Solution:

(i) Given data:

Receiver noise figure = 12 dB, cable loss = 8 dB

LNA gain = 60 dB and noise temperature 150 K

For the receiver,

$$F\,(\text{dB}) = 10\log_{10}F$$

$$12 = 10\log_{10}(F)$$

$$F = 10^{1.2} = 15.85$$

For the cable, $L = 10^{0.8} = 6.309$

$$F_{\text{cable}} = L$$

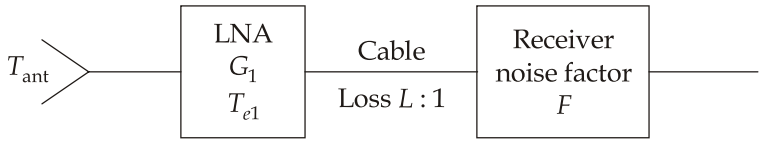
For the LNA, $G_1 = 10^6$

The total equivalent noise temperature referred to the input of a cascaded system in figure (a) is given by the Friis formula as,

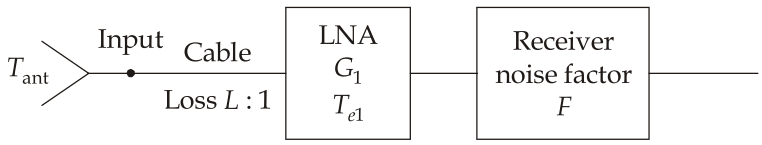
$$T_s = T_{\text{ant}} + T_{\text{LNA}} + \frac{(F_{\text{cable}} - 1)T_0}{G_1} + \frac{(F-1)T_0}{G_1 / L}$$

$$T_s = 45 + 150 + \frac{(6.309-1)290}{10^6} + \frac{(15.85-1)290}{10^6 / 6.309}$$

$$= 195.028\ \text{K} \cong 195\ \text{K}$$



(a)



(b)

For the cascaded system in figure (b), the cable precedes the LNA, and therefore, the equivalent noise temperature referred to the cable input is,

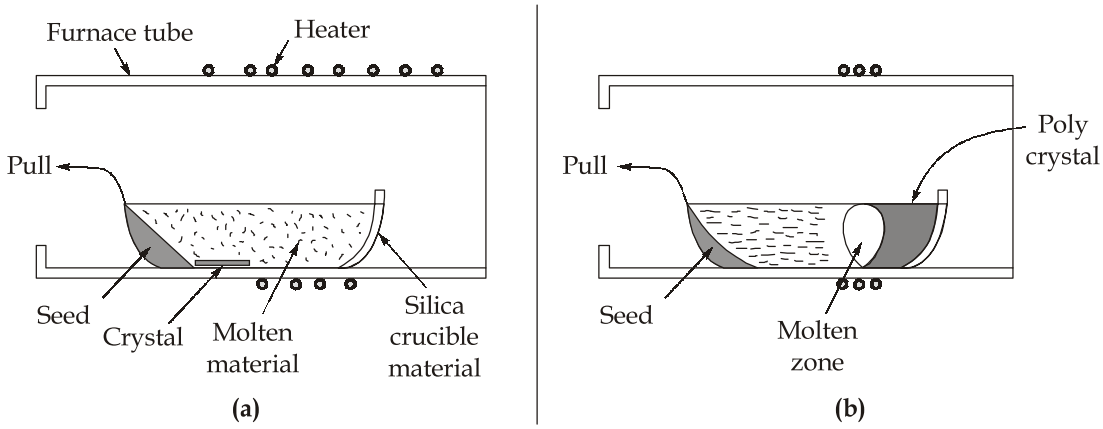
$$T_s = T_{\text{ant}} + (F_{\text{cable}} - 1)T_0 + \frac{T_{\text{LNA}}}{1/L} + \frac{(F-1)T_0}{G_1 / L}$$

$$T_s = 45 + (6.309-1)290 + 6.309\times 150 + \frac{6.309(15.85-1)\times 290}{10^6}$$

$$T_s = 2530.98\ \text{K}$$

(ii)

- Common technique for growing single crystal from the melt involves selective cooling of molten material so that the solidification occurs along a particular crystal direction as shown in below figure (a). Here the silica crucible containing molten Si or Ge is pulled slowly through the furnace.
- Consequently, the solidification begins at one end and slowly proceeds down the length of the bar. To increase crystal growth, a small seed crystal is placed at the end which is cooled first.
- Ge, GaAs and other semiconductor crystals are grown using this method.
- As shown in below figure (b), a small region of polycrystalline material is melted and the molten zone is moved down the crucible at such a rate that a crystal is formed behind the zone, as it moves. This method is not suitable for growth of Si, as it has high melting point and sticks to the crucible.



Q.4 (c) Solution:

(i) Given, 15000 rotation = 60 sec

$\Rightarrow$ 1 rotation = 0.004 sec

$$\Rightarrow \text{Average rotational delay} = \frac{0.004}{2} = 2 \text{ ms}$$

1. **Sequential access:** The file is stored as compactly as possible on the disk. That is, the file occupies all of the sectors on 5 adjacent tracks (5 tracks × 500 sectors/track = 2500 sectors).

   Average seek $\rightarrow$ 4 ms

   Average rotational delay $\rightarrow$ 2 ms

   $$\text{Read 500 sectors} \rightarrow \left(\frac{500 \times 1 \text{ rotation time}}{\text{Number of sectors per track}}\right) = 4 \text{ ms}$$

$\therefore$ Time to read the first track $\rightarrow$ 4 ms + 2 ms + 4 ms = 10 ms

Suppose that the remaining tracks can now be read with essentially no seek time. That is, the I/O operation can keep up with the flow from the disk. Then, at most, we need to deal with rotational delay for each succeeding track. Thus, each successive track is read in 2 + 4 = 6 ms.

To read the entire file.

$$\begin{aligned}\text{Total time} &= 10 + (4 \times 6)\\ &= 10 + 24\\ &= 34 \text{ ms} = 0.034 \text{ seconds}\end{aligned}$$

2. **Random access:** Accesses to the sectors are distributed randomly over the disk.

For each sector, we have

Average seek $\rightarrow$ 4 ms

Rotational delay $\rightarrow$ 2 ms

$$\text{Read 1 sectors} \rightarrow \frac{4\times10^{-3}}{500} = 0.008 \text{ ms}$$

Time to read the first track $\rightarrow$ 4 ms + 2 ms + 0.008 ms = 6.008 ms

$\therefore$
$$\begin{aligned}\text{Total time} &= 2500 \times 6.008\\ &= 15020 \text{ ms}\\ &= 15.02 \text{ seconds}\end{aligned}$$

It is clear that the order in which sectors are read from the disk has a tremendous effect on I/O performance.

(ii) 1. **Memory Address Register (MAR):** The MAR stores the memory address where the CPU wants to read data from or write data to. It's connected to the address bus, which carries the memory address. When the CPU needs to access memory, the address of the desired memory location is first loaded into the MAR.

2. **Memory Data Register (MDR):** The MDR holds the data that is being transferred between the CPU and the memory. It's connected to the data bus, which carries the data. When reading from memory, the data from the memory location addressed by MAR is loaded into the MDR, and then the CPU can access it. When writing to memory, the CPU writes data from the MDR to the memory location addressed by MAR.

Section B : Advanced Electronics + Computer Organization and Architecture + Advanced Communication

Q.5 (a) Solution:

(i) Cache misses occur when the processor attempts to access data that is not present in the cache. When this happens, the data needs to be fetched from the slower main memory, which results in increased latency. There are four primary types of cache misses:

1. **Compulsory Miss** (Cold start misses or first reference misses): It occurs when first time access to a block happens. In this miss, the block must be brought into the cache. Cache size and associativity makes no difference to the number of compulsory misses.
2. **Capacity Miss:** It occurs when program working set is much larger than the cache capacity. Blocks are being discarded from cache because cache cannot contain all blocks needed for program execution.
3. **Conflict Miss** (Collision misses or interference misses): In the case of set associative or direct mapped block placement strategies, conflict misses occur when several blocks are mapped to the same set or block frame.
4. **Coherence Miss (Invalidation misses):** It occur in systems with multiple processors or cores. These misses arise when a processor's cache holds stale data that has been modified by another processor or an external device.

(ii) Pipeline cycle time (t_p):

$$t_p = \max(5, 8, 6, 9, 7, 8) + \text{Intermediate register delay}$$
$$= 9 \text{ ns} + 1 \text{ ns} = 10 \text{ ns}$$

One instruction execution time in non-pipeline system,

$$t_n = 43 \text{ ns}$$

$$\therefore \quad \text{Pipeline time} = [(k + n - 1) + (\text{Stall cycles})]t_p$$
$$= \left[(6+1000-1)+\left(1000\times\frac{20}{100}\times 3\right)+\left(1000\times\frac{30}{100}\times 1\right)\right]10 \text{ ns}$$
$$= [1005 + 600 + 300]10\text{ns}$$
$$= 1905 \times 10 \text{ ns} = 19050 \text{ ns}$$

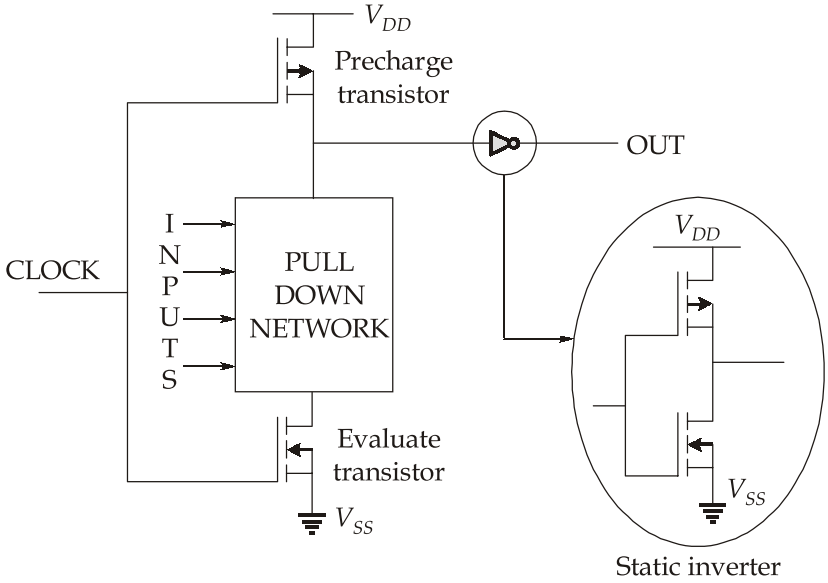
$$\text{Non-pipeline time} = n \times t_n$$
$$= 1000 \times 43 \text{ ns} = 43000 \text{ ns}$$

$\therefore$ Speed-up of this pipeline as compared to the corresponding non-pipeline system is,

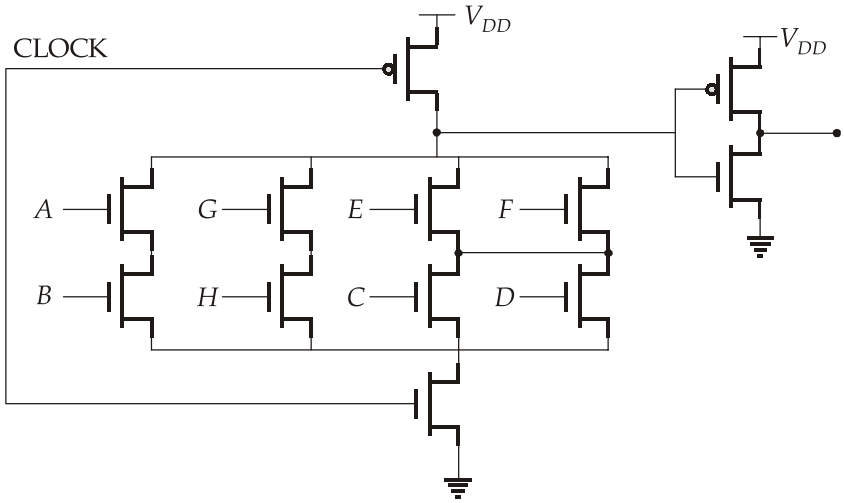
$$S = \frac{\text{Non-pipeline time}}{\text{Pipeline time}} = \frac{43000}{19050} = 2.2572$$

Q.5 (b) Solution:

Domino logic is a CMOS-based evolution of dynamic logic techniques consisting of a dynamic logic gate cascaded into a static CMOS inverter. It differs from static logic by including a clock signal to speed up performance. In CMOS dynamic logic gates, the gate output is precharged to the power supply voltage while the clock is off (the 'precharge' phase), and then is evaluated to the correct logic state while the clock is on (the 'evaluation' phase) by draining the relevant NMOS transistors in the pull-down network. The block diagram of domino CMOS logic is as shown below:



The domino logic version of the given conventional CMOS logic is as shown below:



Q.5 (c) Solution:

(i) We have, $NA = n_1\sqrt{2\Delta}$

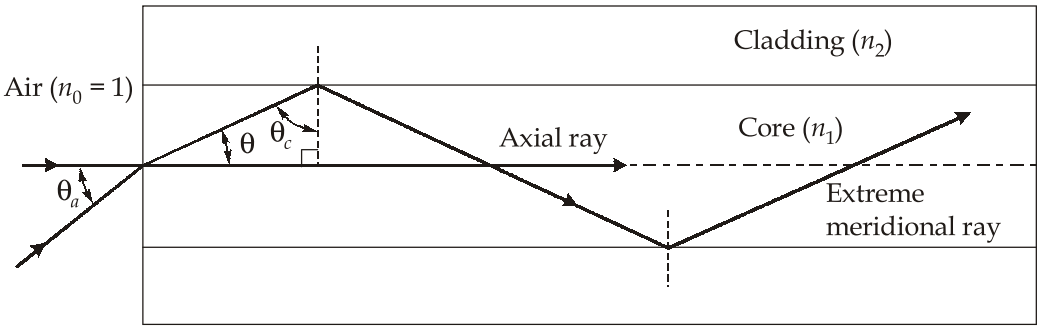
Squaring on both sides, we get:

$$(NA)^2 = n_1^2(2\Delta) \qquad ...(i)$$

The pulse broadening, ΔT is the maximum time delay difference between the fastest and slowest modes. The time taken for the axial ray to travel along a fiber of length L gives the minimum delay time,

$$T_{\min} = \frac{L}{c/n_1} = \frac{Ln_1}{c}$$

where n_1 is the refractive index of the core and c is the velocity of light in a vacuum.



The extreme meridional ray exhibits the maximum delay time T_{Max} where:

$$T_{\max} = \frac{L/\cos\theta}{c/n_1} = \frac{Ln_1}{c\cos\theta}$$

Using Snell's law of refraction at the core-cladding interface,

$$\sin\phi_c = \frac{n_2}{n_1} = \cos\theta$$

where n_2 is the refractive index of the cladding. Thus,

$$T_{\max} = \frac{Ln_1^2}{cn_2}$$

Thus, the pulse broadening ΔT is given by,

$$\Delta T = T_{\max} - T_{\min} = \frac{Ln_1^2}{cn_2}\left(\frac{n_1 - n_2}{n_1}\right)$$

$$\Delta T = \frac{Ln_1\Delta}{c} \qquad ...(ii)$$

From equation (i),

$$n_1\Delta = \frac{(NA)^2}{2n_1}$$

Substituting this value in equation (ii), we get,

$$\Delta T = \frac{L(NA)^2}{2n_1 c}$$ Hence proved.

(ii) For multimode fibre, critical radius of curvature is given by

$$R_{cm} = \frac{3n_1^2\lambda}{4\pi(n_1^2 - n_2^2)^{3/2}}$$

As we know, $$\Delta = \frac{n_1 - n_2}{n_1}$$

Rearranging we get:

$$n_2 = n_1(1 - \Delta) = 1.5(1 - 0.03)$$

$$\therefore \quad n_2 = 1.455$$

Substituting the values, we get,

$$R_{cm} = \frac{3\times(1.5)^2\times1.3\times10^{-6}}{4\pi\left\{(1.5)^2 - (1.455)^2\right\}^{3/2}}\text{m}$$

On solving, we get,

$$R_{cm} = 14.4 \times 10^{-6}\text{ m} = 14.4\ \mu\text{m}$$

Q.5 (d) Solution:

(i) **1.** We have the value of material dispersion coefficient given as:

$$D_{\text{mat}} = \left|\frac{\lambda}{c}\frac{d^2n}{d\lambda^2}\right|$$

or $$D_{\text{mat}} = \frac{1}{\lambda c}\left|\lambda^2\frac{d^2n}{d\lambda^2}\right|$$

Substituting the given values, we get:

$$D_{\text{mat}} = \frac{1}{1.3\times10^{-6}\text{m}\times3\times10^8\text{m/sec}}\times0.03$$

$$\therefore \quad D_{\text{mat}} = \frac{0.03}{1310\text{ nm}\times3\times10^5\text{km/sec}}$$

$$\therefore \quad D_{\text{mat}} = \frac{0.03}{1310\times3\times10^5}\text{ sec/nm-km}$$

$\therefore \quad D_{mat} = 7.63 \times 10^{-11}$ sec/nm-km

$D_{mat} = 76.3$ ps/nm-km

2. RMS pulse broadening per kilometer due to material dispersion is:

ΔT_{mat} (per km) $= \Delta T_s \times D_{mat}$

where ΔT_s is the rms spectral width of source

$\therefore \quad \Delta T_{mat}$ (per km) $= 15 \times 76.3$ ps/km

or ΔT_{mat} (per km) $= 1144.5$ ps/km

ΔT_{mat} (per km) $= 1.14$ ns/km

(ii) 1. For graded index fiber to act in single mode operation, normalized frequency is given by:

$$V_g = 2.405\left(\frac{\alpha+2}{\alpha}\right)^{1/2} = 2\pi\left(\frac{a}{\lambda}\right)_g (NA) \quad ...(i)$$

For single mode step index fiber, normalized frequency is:

$$V_s = 2.405 = 2\pi\left(\frac{a}{\lambda}\right)_s \cdot (NA) \quad ...(ii)$$

Dividing equations (i) and (ii), we get

$$\left(\frac{\alpha+2}{\alpha}\right)^{1/2} = \frac{\left(\frac{a}{\lambda}\right)_g}{\left(\frac{\alpha}{\lambda}\right)_s}$$

For parabolic refractive index profile, $\alpha = 2$

$\therefore \quad \left(\frac{a}{\lambda}\right)_g = \sqrt{2}\left(\frac{a}{\lambda}\right)_s$

or $\left(\frac{a}{\lambda}\right)_g = 1.4\left(\frac{a}{\lambda}\right)_s$

2. In a step-index fiber, for single-mode transmission, the normalised frequency is given by 2.405 i.e.

$$V = 2.405 = \frac{2\pi a}{\lambda} \times N.A = \frac{2\pi a}{\lambda} \times \sqrt{n_1^2 - n_2^2}$$

The radius of core,

$$a = \frac{2.405\lambda}{2\pi(n_1^2 - n_2^2)^{1/2}}$$

$$a = \frac{2.405\times1.5\times10^{-6}}{2\pi\times\sqrt{(1.49)^2-(1.48)^2}}$$

$$a = \frac{2.405\times1.5\times10^{-6}}{1.0828} = 3.33\times10^{-6}\,\text{m} = 3.33\,\mu\text{m}$$

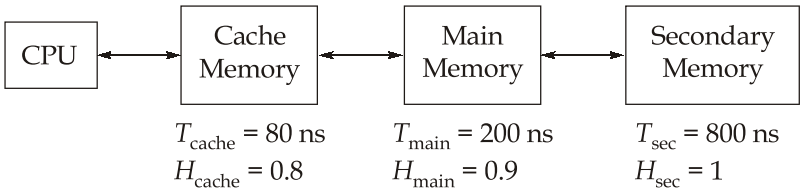
Thus, Fiber core diameter is $2a$ i.e., 6.66 μm.

Q.5 (e) Solution:

(i) Given,

$$T_{cache} = 80\text{ ns}, H_{cache} = 0.8$$
$$T_{main} = 200\text{ ns}, H_{main} = 0.9$$
$$T_{sec} = 800\text{ ns}, H_{sec} = 1$$

A Hierarchical memory organization is given as shown below,

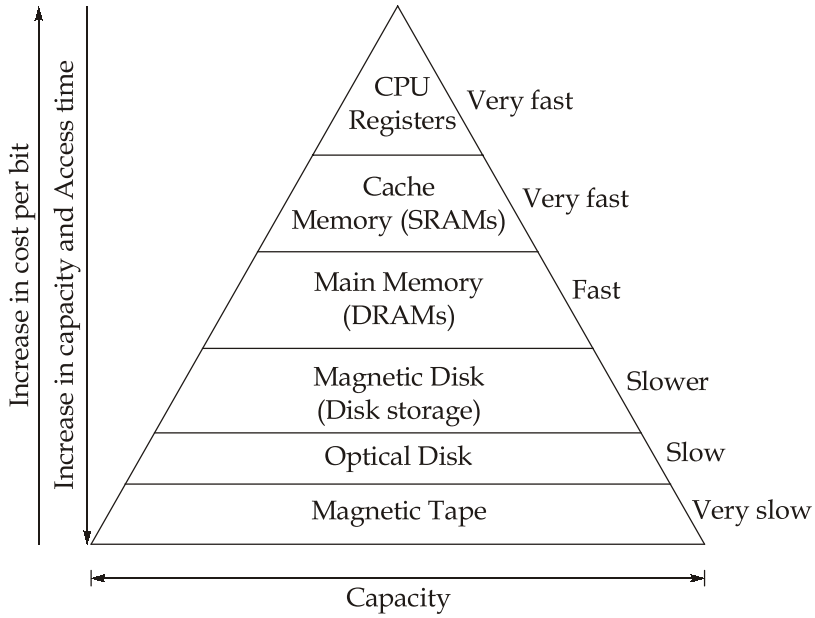


The average memory access time of memory system is,

$$\text{AMAT} = (H_{cache}\times T_{cache}) + [(1-H_{cache})\times H_{main}\times(T_{cache}+T_{main})] + [(1-H_{cache})\times(1-H_{main})\times(T_{cache}+T_{main}+T_{sec})]$$
$$= (0.8\times80\times10^{-9}) + [(1-0.8)\times0.9\times(80+200)\times10^{-9}] + [(1-0.8)\times(1-0.9)\times(80+200+800)\times10^{-9}]$$
$$\Rightarrow \text{AMAT} = (64+50.4+21.6)\times10^{-9}$$
$$\Rightarrow \text{AMAT} = 136\text{ ns}$$

(ii) In the Computer System Design, Memory Hierarchy is an enhancement to organize the memory such that it can minimize the access time. The memory hierarchy was developed based on a program behavior known as locality of references. Memory references are generated by the CPU for either instruction or data access. These accesses tend to be clustered in certain regions in time, space and ordering.

There are multiple levels present in the memory, each one having a different size, different cost, etc. Some types of memory like cache, and main memory are faster as compared to other types of memory but they are having a little less size and are also costly whereas some memory has a little higher storage value, but they are a little slower. The figure below demonstrates the different levels of memory hierarchy.



*As one goes down the hierarchy, the following occur:

(a) Decreasing cost per bit.

(b) Increasing capacity.

(c) Increasing access time.

(d) Decreasing frequency of access of the memory by the processor.

Q.6 (a) Solution:

Given that

$$EIRP = 57 \text{ dBW}$$

$$\text{Downlink carrier frequency} = 12.5 \text{ GHz}$$

$$\frac{C}{N_0} = 85 \text{ dB-Hz}$$

$$\text{Data rate} = 10 \text{ Mb/s}$$

Required E_b/N_0 at the receiving earth terminal = 10 dB

$$\text{Distance, } R = 41{,}000 \text{ km, } \eta = 55\%, \ T = 310°\text{K}$$

The expression for carrier to noise ratio at the receiver is as follows:

$$\frac{C}{N_0} = \frac{P_t G_t}{KB}\left[\frac{\lambda}{4\pi d}\right]^2 \cdot \frac{G_T}{T}$$

In this expression, $P_t G_t = EIRP$, $d = R$ and $\left[\frac{\lambda}{4\pi d}\right]^2 = \frac{1}{\text{Path loss}}$

$$\therefore \quad \left[\frac{C}{N_0}\right]_{dB} = [EIRP]_{dB} - [\text{Path loss}]_{dB} + [G_r/T]_{dB} - [K]_{dB} - [B]_{dB} \quad \text{...(i)}$$

Step I: To find path loss:

$$\text{Path loss} = \left[\frac{4\pi R}{\lambda}\right]^2$$

But, $$\lambda = \frac{c}{f} = \frac{3\times10^8}{12.5\times10^9} = 0.024 \text{ m}$$

$$\therefore \quad \text{Path loss} = \left[\frac{4\pi\times41000\times10^3}{0.024}\right]^2 = 4.608 \times 10^{20}$$

$$= 10 \log[4.608 \times 10^{20}] = 206.64 \text{ dB}$$

Step-II: To find $[K]$ and $[B]$

$$K = \text{Boltzman's constant} = 1.38 \times 10^{-23}$$

$$\therefore \quad [K] = 10\log_{10}(1.38 \times 10^{-23}) = -228.6 \text{ dB}$$

And $$B = \text{bandwidth}$$

$$= \frac{R_b}{2} = \frac{1}{2} \times \text{data rate} = 5 \text{ MHz}$$

$$\therefore \quad [B]_{dB} = 10 \log_{10}(5 \times 10^6) = 67 \text{ dB}$$

Step-III: To find $[G_r/T]$

Substituting all the values into equation (i), we get

$$85 = 57 - 206.64 + [G_r/T] + 228.6$$

Note that since C/N_0 is specified in dB-Hz, we have not substituted the value of B.

$$\therefore \quad [G_r/T] = 6.04 \text{ dB}$$

$$\therefore \quad 10\log_{10}[G_r/T] = 6.04$$

$$\therefore \quad [G_r/T] = 4.018$$

Step-IV: To find gain of receiving antenna:

$$T = 310°\text{K}$$

$$\therefore \quad G_r = 4.018 \times 310 = 1245.55$$

Step-V: Find the diameter of antenna:

$$G_r = \eta\left(\frac{\pi D}{\lambda}\right)^2 \quad \text{where } D = \text{diameter of the dish antenna}$$

$$1245.55 = 0.55\left[\frac{\pi\times D}{0.024}\right]^2$$

$$\therefore \quad D = 0.3635 \text{ m}$$

Q.6 (b) Solution:

(i) There are mainly two types of scaling used in VLSI technology.

1. Constant Field Scaling
2. Constant Voltage Scaling

1. Constant Field Scaling: Electric field in the MOSFET is preserved while dimensions are scaled by a factor of 'S'. To achieve this, potential must also be scaled by a factor of 'S'. So, power supply voltage and all terminal voltages are scaled by factor 'S', but the scaling of voltages may not be very practical in many cases.

In particular, the peripheral and interface circuitry may require certain voltage levels for all inputs and output voltages which in turn would necessitate multiple power supply voltages. This is a limitation of constant field scaling.

2. Constant Voltage Scaling: In constant voltage scaling, all dimensions of the MOSFET are reduced by a factor of 'S' but power supply and terminal voltages are kept constant.

In constant voltage scaling, drain current density and power density are increased which may eventually cause serious reliability problem for scaled transistor such as electro-migration, hot carrier-carrier degradation, oxide break-down and electrical overstress.

Parameters	Before scaling	After scaling	
		Constant field scaling	Constant voltage scaling
Channel length	L	L/S	L/S
Channel width	W	W/S	W/S
Gate oxide thickness	t_{ox}	t_{ox}/S	t_{ox}/S
Junction depth	X_j	X_j/S	X_j/S
Power supply voltage	V_{DD}	V_{DD}/S	V_{DD}
Threshold voltage	V_T	V_T/S	V_T
Doping densities	N_A, N_D	SN_A, SN_D	S^2N_A, S^2N_D
Oxide capacitance	C_{ox}	C_{ox}/S	C_{ox}/S
Drain current	I_D	I_D/S	SI_D
Power dissipation	P_D	P_D/S^2	$S.P_D$
Power density	P_D/Area	P_D/Area	S^3P_D/Area

(ii) Given, $t = 2$ hours

$A = 0.2\ \mu m$

$B = 0.5\ \mu m^2/hr$

Initial oxide thickness $= 0 \Rightarrow t_i = 0$

Using Deal Grove Model, the relation between final oxide thickness and initial oxide thickness is,

$$\frac{t_{ox}^2 - t_i^2}{B} + \frac{t_{ox} - t_i}{B/A} = t$$

$$\left(t_{ox}^2 - t_i^2\right) + A(t_{ox} - t_i) = Bt$$

$$\Rightarrow \left(t_{ox}^2 - 0\right) + (0.2)(t_{ox} - 0) = (0.5)(2)$$

$$\Rightarrow t_{ox}^2 + 0.2t_{ox} - 1 = 0$$

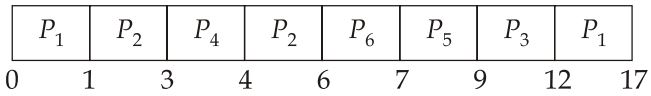
On solving, $t_{ox} = 0.905, -1.105$

$\Rightarrow t_{ox} = 0.905\ \mu m$

Q.6 (c) Solution:

(i) In Shortest Remaining Time First (SRTF) scheduling algorithm, the process with the least time left to finish is executed. The running process will continue until it finishes or a new process with a shorter remaining time arrives.

GANTT chart:



Ready Queue:

Time	Processes in Ready state (BT)
0	$P_1(6)$
1	$P_1(5), P_2(4)$
2	$P_1(5), P_2(3), P_3(3)$
3	$P_1(5), P_2(2), P_3(3), P_4(1)$
4	$P_1(5), P_2(2), P_3(3), P_5(2)$
5	$P_1(5), P_2(1), P_3(3), P_5(2), P_6(1)$
6	$P_1(5), P_3(3), P_5(2), P_6(1)$

Process	Arrival Time (AT)	Burst Time (BT)	Completion Time (CT)	Turn around time (TAT = CT – AT)	Waiting Time WT = (TAT – BT)
P_1	0	6	17	17	11
P_2	1	4	6	5	1
P_3	2	3	12	10	7
P_4	3	1	4	1	0
P_5	4	2	9	5	3
P_6	5	1	7	2	1

The average waiting time is calculated as below:

$$\text{Average waiting time} = \frac{\text{Sum of waiting time of all processes}}{\text{Number of processes}}$$

$$= \frac{(11+1+7+0+3+1)}{6} = \frac{23}{6} = 3.833 \text{ ms}$$

(ii)

Concurrency	Parallelism
1. Concurrency is the task of running and managing the multiple computations at the same time.	1. While parallelism is the task of running multiple computations simultaneously.
2. Concurrency is achieved through the interleaving operation of processes on the central processing unit (CPU) or in other words by the context switching.	2. While it is achieved through multiple central processing units (CPUs).
3. Concurrency can be done by using a single processing unit.	3. While this can't be done by using a single processing unit. It needs multiple processing units.
4. Concurrency increases the amount of work finished at a time.	4. While it improves the throughput and computational speed of the system.
5. Concurrency deals lot of things simultaneously.	5. While it do lot of things simultaneously.
6. Concurrency is the non-deterministic control flow approach.	6. While it is deterministic control flow approach.
7. In concurrency, debugging is very hard.	7. While in this debugging is also hard but simple than concurrency.

Q.7 (a) Solution:

(i) 1. Polling: (Software method)

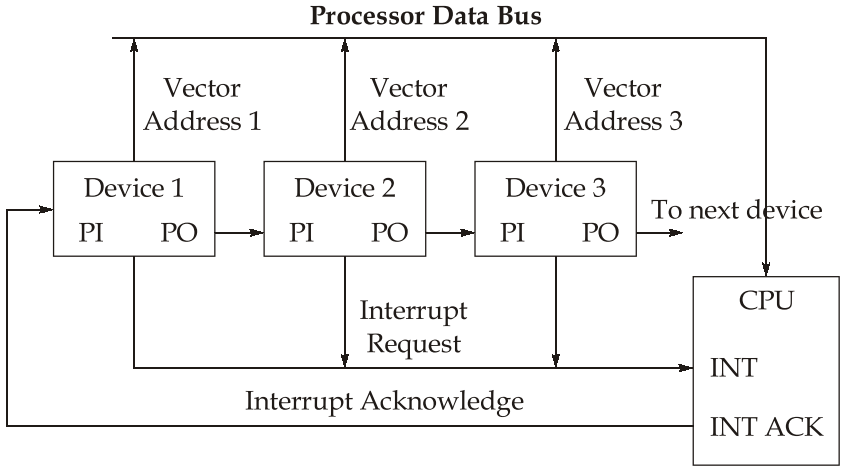
Polling is a software-based method of interrupt handling where the Central Processing Unit (CPU) continuously checks the status of various I/O (Input/Output) devices for the interrupt. In this method, all interrupts are monitored by branching to a service program. This program then checks with each device if it is the one generating the interrupt. The order of checking is determined by the priority that has to be set. The device having the highest priority is checked first and then devices are checked in descending order of priority. If the device is checked to be generating the interrupt, another service program is called which works specifically for that particular device.

Disadvantage: This method is quite slow. The CPU spends a significant amount of time continuously checking/polling devices, even if no device requires service. This can lead to wasted CPU cycles, especially in systems with many devices or infrequent I/O events.

2. Daisy chaining: (Hardware method)

The daisy-chaining method involves connecting all the devices that can request an interrupt in a serial manner. This configuration is governed by the priority of the devices.

The device with the highest priority is placed first followed by the second highest priority device and so on as shown below.



There is an interrupt request line which is common to all the devices and goes into the CPU.

- When no interrupts are pending, the line is in HIGH state. But if any of the devices raises an interrupt, it places the interrupt request line in the LOW state.

- The CPU acknowledges this interrupt request from the line and then enables the interrupt acknowledge line in response to the request.
- This signal is received at the PI (Priority in) input of device 1. If the device has not requested the interrupt, it passes this signal to the next device through its PO (priority out) output. (PI = 1 and PO = 1). However, if the device had requested the interrupt, (PI =1 and PO = 0).
  - The device consumes the acknowledge signal and block its further use by placing 0 at its PO (priority out) output.
  - The device then proceeds to place its interrupt vector address into the data bus of CPU.
  - The device puts its interrupt request signal in HIGH state to indicate its interrupt has been taken care of.
- If a device gets 0 at its PI input, it generates 0 at the PO output to tell other devices that acknowledge signal has been blocked. (PI = 0 and PO = 0)

Hence, the device having PI = 1 and PO = 0 is the highest priority device that is requesting an interrupt. Therefore, by daisy chain arrangement we have ensured that the highest priority interrupt gets serviced first and have established a hierarchy. The farther a device is from the first device, the lower its priority.

Disadvantage : This method suffers from "starvation" refers to a situation where lower-priority devices or tasks are unfairly denied access to resources because higher-priority tasks continuously interrupt the system.

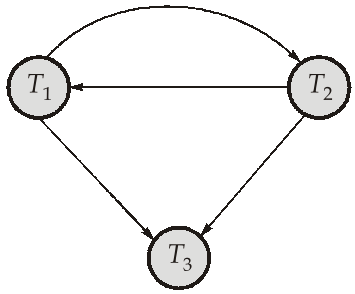
(ii) To draw the precedence graph,

1. Create a node T in the graph for each participating transaction in the schedule.
2. If a Transaction T_j executes a read_item (X) after T_i executes a write_item (X), draw an edge from T_i to T_j in the graph.
3. If a Transaction T_j executes a write_item (X) after T_i executes a read_item (X), draw an edge from T_i to T_j in the graph.
4. If a Transaction T_j executes a write_item (X) after T_i executes a write_item (X), draw an edge from T_i to T_j in the graph.

The schedule S is serializable if there is no cycle in the precedence graph.

$S_1 : W_2(x), W_1(x), R_3(x), W_2(y), R_3(y), R_3(z), R_2(x)$

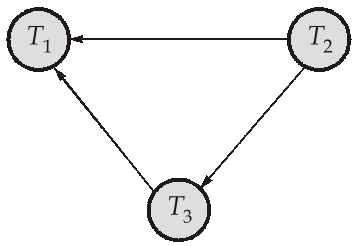
T_1	T_2	T_3
	$W_2(x)$	
$W_1(x)$		
		$R_3(x)$
	$W_2(y)$	
		$R_3(y)$
		$R_3(z)$
	$R_2(x)$	



Since precedence graph contain cycle, hence schedule S_1 is not conflict serializable.

$S_2 : R_3(z), W_2(x), W_2(y), R_1(x), R_3(x), R_2(x), R_3(y), W_1(z)$

T_1	T_2	T_3
		$R_3(z)$
	$W_2(x)$	
	$W_2(y)$	
$R_1(x)$		
		$R_3(x)$
	$R_2(x)$	
		$R_3(y)$
$W_1(z)$		



Since precedence graph contains no cycle, hence schedule S_2 is conflict serializable.

Q.7 (b) Solution:

(i) 1. Testability:

It is the property of a circuit that makes it easy to test. Testability is a design characteristic that add features in a chip to make the testing cost-effective. It allows for the status (normal, inoperable, degraded) of a device to be determined and the detection of faults within the device to be performed quickly to reduce both the test time and cost.

2. Yield:

It is ratio of number of good chips to the total number of fabricated chips on a wafer. Yield can be written as

$$\text{Yield} = \frac{\text{Number of good chips}}{\text{Total number of fabricated chips}}$$

The yield is divided into three subcategories:

1. Process yield: It refers to the yield related to the manufacturing process.
2. Functional yield: It refers to the percentage of chips that pass all the required functional tests.
3. Parametric yield: It refers to the percentage of chips that not only function correctly but also meet all their specified performance parameters

3. **Manufacturability:** Design for Manufacturability is a critical aspect of VLSI design that focuses on optimizing chip layouts to ensure high yield and performance during fabrication. With the increasing complexity of semiconductor devices, it helps in minimizing defects, improving reliability, and reducing manufacturing costs.

   There are different levels at which the design can be optimized to increase the manufacturability and hence, improve the yield. The levels are listed as

   1. Optimizing physical layout
   2. Minimising power dissipation
   3. Introducing redundancy

4. **Reliability:**

   - Reliability is concerned with projecting the lifetime of a component once it is placed into operation. The projected lifetime is the number of hours of operation that can be expected before a failure occurs.
   - It depends on the design and process conditions.
   - The major causes for chip reliability problems can be characterized into the following:
     (a) Electrostatic discharge (ESD) and electrical overstress (EOS)
     (b) Latchup in CMOS I/O and internal circuits.
     (c) Hot carrier induced aging.
     (d) Oxide breakdown.
     (e) Power supply drop and ground bouncing.
     (f) On-chip noise and cross talk.

(ii) Consider a hexagonal cell structure with radius $r = 1.4$ km.

The area of hexagon cell of radius r is = $1.5r^2\sqrt{3}$

$$\text{Area } (A) = 1.5(1.4)^2\sqrt{3} \qquad [\because r = 1.4 \text{ km}]$$

$$\text{Area, } A = 1.5(1.96)\sqrt{3} = 5.09 \text{ km}^2$$

Total area covered will be (A_T) = (Area) (Number of cells)

$$= 5.09 \times 34$$

$$A_T = 173.06 \text{ km}^2$$

For a reuse factor of $N = 7$, the number of channels in one cell is equal to,

$$\frac{\text{Number of traffic channels}}{7} = \frac{343}{7} = 49 \text{ channels}$$

The total capacity (C_T) = (Total number of channels) (Number of cells)

C_T = (49 channels) (34 cells) = 1666 channels

Hence, the system can handle 1666 concurrent calls.

Q.7 (c) Solution:

TCP/IP stands for Transmission Control Protocol/Internet Protocol. It has four layers as mentioned below:

Application layer
Transport layer
Network layer/Internet layer
Network Access layer

1. **Application Layer:** This is the top-most layer of TCP/IP protocol suite. This layer includes applications or processes that use transport layer services to deliver data to destination computer application layer. The application layer has various protocols that applications use to communicate with the transport layer. Some of the popular application layer protocols are:
   - HTTP (Hypertext transfer protocol)
   - FTP (File transfer protocol)
   - SMTP (Simple mail transfer protocol)
   - SNMP (Simple network management protocol) etc.
2. **Transport Layer:** This layer provides backbone to data flow between two hosts. There are many protocols that work at this layer but the two most commonly used protocols at transport layer are TCP and UPD.

TCP is used where reliable connection is required while UDP is used in case of unreliable connections.

TCP divides the data (coming from application layer) into proper sized chunks and then passes these chunks on the network. It acknowledges received packets, waits for the acknowledgment of the packets it sent and sets timeout to resend the packets if acknowledgments are not received in time.

UDP provides a comparatively simpler but unreliable service by sending packets from one host to another. UDP does not take any extra measures to ensure that the data sent is received by the target host or not.

3. **Network layer:** This layer is also known as internet layer. The main purpose of this layer is to organize or handle movements of data on network, i.e., routing of data over the network. The main protocol used at this layer is IP. While ICMP and IGMP are also used at this layer.
4. **Data link layer:** This layer is also known as network interface layer. This layer normally consist of device drivers and the network interface card attached to the system. Both the device drivers and the network interface card take care of the communication details with the media being used to transfer the data over the network. In most of the cases, this media is in the form of cables. Some of the famous protocols that are used at this layer include ARP (Address Resolution Protocol), PPP (Point to point protocol) etc.

Address used at different layers:

1. Application layer : Application-specific address (i.e., domain name)
2. Transport layer: Port Address
3. Internet Layer: IP Address (IPv4 or IPv6 address)
4. Network Access Layer: Physical Address (e.g. MAC address)

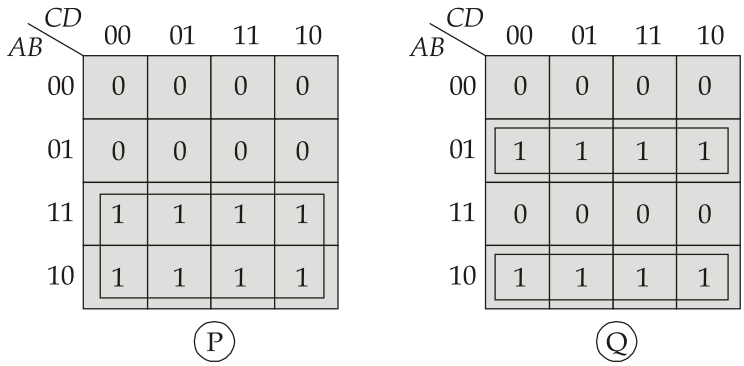
Q.8 (a) Solution:

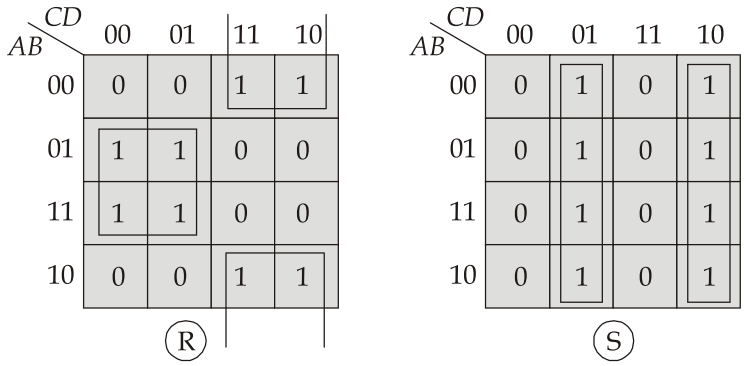
(i) Binary to gray code converter:

Truth table:

A	B	C	D	P	Q	R	S
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	1
0	0	1	1	0	0	1	0
0	1	0	0	0	1	1	0
0	1	0	1	0	1	1	1
0	1	1	0	0	1	0	1
0	1	1	1	0	1	0	0
1	0	0	0	1	1	0	0
1	0	0	1	1	1	0	1
1	0	1	0	1	1	1	1
1	0	1	1	1	1	1	0
1	1	0	0	1	0	1	0
1	1	0	1	1	0	1	1
1	1	1	0	1	0	0	1
1	1	1	1	1	0	0	0

Karnaugh maps for the four outputs P, Q, R and S are respectively shown below:





The minimized Boolean expressions are given by the following equations:

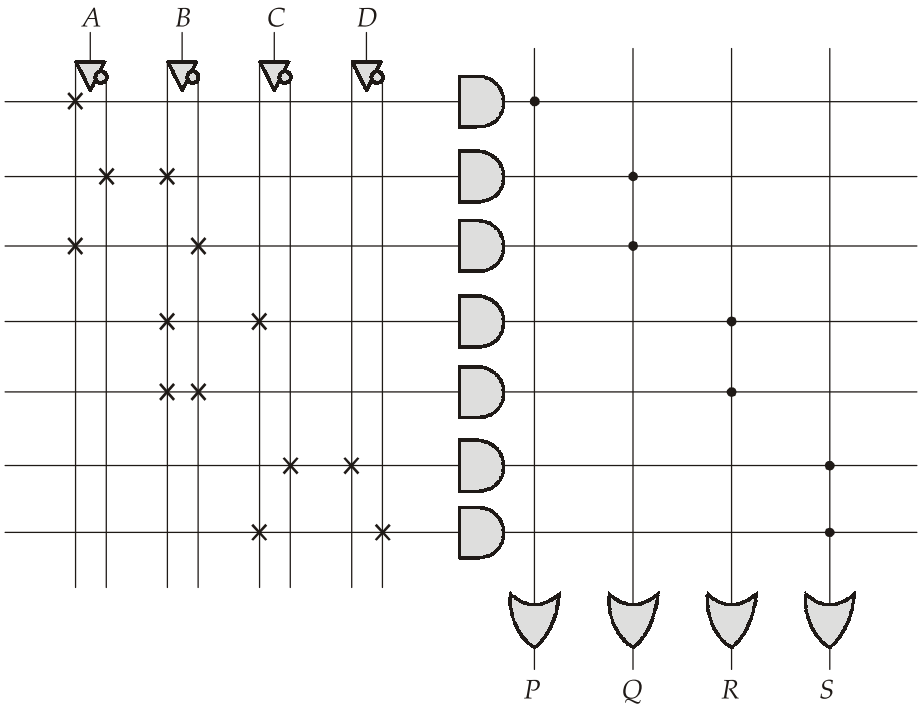
$$P = A$$

$$Q = \bar{A}B + A\bar{B}$$

$$R = B\bar{C} + \bar{B}C$$

$$S = \bar{C}D + C\bar{D}$$

PLA is a programmable logic device with a programmable AND array and a programmable OR array. The above functions have four inputs and 7 distinct product terms, therefore PLA device with 4 inputs and 7 AND gates in the programmable AND array is used. Further, to generate four outputs (P,Q,R,S), 4 OR Gates are used in the programmable OR array. The circuit is as shown below:



(ii) 1. **Controllability:** This is the ability to control the signal value at a node using only the input pins.

A node is called controllable if we are able to drive a value of 0 or 1 onto that node using primary inputs.

To test for a stuck-at fault at a specific node in a digital circuit, it's crucial to be able to control that node to both logic 0 and logic 1. This allows for the creation of test patterns that can expose the fault if it exists.

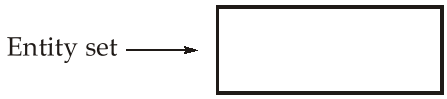
2. **Observability:** It is the ability of propagating a signal value at a node to one of output pins of the circuit so that it can be observed.

A node is called observable if the signal value at the node can be propagated to one of the primary outputs.

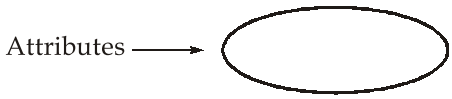
To propagate a stuck-at-fault at a node at the output, the node must be observable.

Q.8 (b) Solution:

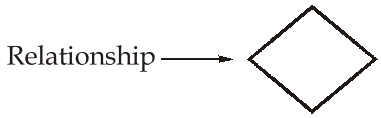
(i) 1. **Entity:** An entity is referred to as a real-world object. It can be a name, place, object, class, etc. These are represented by a rectangle in an ER diagram.



2. **Attributes:** An attribute can be defined as the description of the entity. These are represented by Ellipse in an ER Diagram. It can be Age, Roll Number, or Marks for a student.



3. **Relationship:** Relationships are used to define relations or association among different entities. In ER diagram, the relationship type is represented by a diamond connecting the entities with lines.



4. **Domain:** A domain represents the set of all possible values that an attribute can hold in a database. It defines the type, format etc. for the values associated with an attribute. For example, if an attribute is 'age', the domain would be a set of non-negative integers.

(ii) Cache size = 64 KB

Cache block size = 32B

$$\text{Number of cache lines} = \frac{\text{Cache size}}{\text{Cache block size}} = \frac{2^6 \times 2^{10}}{2^5} = 2^{11}$$

Since the cache is 8-way set associative $\Rightarrow$ Number of sets = $\frac{2^{11}}{2^3} = 2^8$

Hence, 28 bit physical address is broke down as,

Tag	Set Index	Block offset
15	8	5
= (28 – 8 – 5) = 15 bits	Number of sets = 2^8 Thus, 8 bits required	Cache block size = 32 B Thus, 5 bits required

Hence, Tag space in the line = 15 bits

Tag directory size = Number of lines × [Tag space + Controller bits]

$= 2^{11} \times [15 + 2 + 2 + 3]$

$= 2^{11} \times 22$

$= 22 \times 2^{11}$ bits

Q.8 (c) Solution:

(i) The first octet of an IP address determines its class in the classful IP addressing scheme. Each class (A, B, C, etc.) is defined by a specific allocation of bits for the network and host portions of the address. To determine the network address for a given IP address, the host bits are masked using a subnet mask.

First Octet (in decimal notation)	Address Class	Network Bits	Host Bits	Subnet Mask
0-127	Class A	8 bits	24 bits	255.0.0.0
128-191	Class B	16 bits	16 bits	255.255.0.0
192-223	Class C	24 bits	8 bits	255.255.255.0
224-239	Class D	–	–	Reserved for multicasting
240-255	Class E	–	–	Experimental

1. From the observation, the IP address 23.56.89.12 is a class A address. Therefore, the address is ANDed with the class A subnet mask to determine the network address.

IP address (decimal)	23	56	89	12
IP address (binary)	00010111	00111000	01011001	00001100
Default Mask	11111111	00000000	00000000	00000000
Result of ANDing	000010111	00000000	00000000	00000000
Network address	23	0	0	0

2. From observation, the IP address 133.45.78.65 is a class B address. Therefore, the address is ANDed with class B subnet mask to determine the network address.

IP address (decimal)	133	45	78	65
IP address (binary)	10000101	00101101	01001110	01000001
Default Mask	11111111	11111111	00000000	00000000
Result of ANDing	10000101	00101101	00000000	00000000
Network address	133	45	0	0

3. From observation, the IP address 201.150.47.19 is a class C address; therefore, the address is ANDed with class C subnet mask to determine the network address.

IP address (decimal)	201	150	47	19
IP address (binary)	11001001	10010110	00101111	00010011
Default Mask	11111111	11111111	11111111	00000000
Result of ANDing	11001001	10010110	00101111	00000000
Network address	201	150	47	0

(ii) 1. Determining the address class for an IP address in binary notation requires examining only the beginning bits of the left most octet.

- If the first bit is a 0, the IP address is class A.
- If the first two bits are 10, the IP address is class B.
- If the first three bits are 110, the IP address is class C.
- If the first four bits are 1110, the IP address is class D.
- If the first four bits are 1111, the IP address is class E.

In the leftmost octet of the given IP address, 11000000, first three bits are 110, therefore it is a class C address.

2. Determining the address class for an IP address in hexadecimal notation requires looking at only the first hex character of the left most octet.
   - If the first hex character is in the range: 0 to 7, the IP address is class A.
   - If the first hex character is in the range: 8 to B, the IP address is class B.
   - If the first hex character is C or D, the IP address is class C.
   - If the first hex character is E, the IP address is class D.
   - If the first hex character is F, the IP address is class E.

   For the IP address 8F 7C 2A 1B, the first hex character is 8, therefore it is a class B address.
3. Determining the address class for an IP address in dotted decimal notation requires looking at only the first dotted decimal number.
   - If the first dotted decimal number is in the range: 0 – 127, the IP address is class A.
   - If the first dotted decimal number is in the range: 128 – 191, the IP address is class B.
   - If the first dotted decimal number is in the range: 192 – 223, the IP address is class C.
   - If the first dotted decimal number is in the range: 224 – 239, the IP address is class D.
   - If the first dotted decimal number is in the range: 240 – 255, the IP address is class E.

   For dotted decimal IP address 172.31.0.1, the first decimal number falls within range 128-191 and therefore, it is a class B address.

❍❍❍❍