Multimedia means that the computer info can be represented through audio, video, graphics, animations etc.
Multimedia is concerned with computer controlled animation of text, graphics, drawings.

Multimedia is an application which makes a collection of multiple media sources.
Eg. Hypertext and Hypermedia.

Components of Multimedia

⇒ Capture Devices :- Mic, Video Recorder.
⇒ Storage Devices.
⇒ Communication Network.
⇒ Mobile System
⇒ Computer System and Smartphones.
⇒ Display Devices

Multimedia Data Input & format

* Text & Static Data (Time Independent)

Source
Static data :- Keyboard, speech, optical

Graphics :- Lines, polygons.
↳ Input Devices :- Styles, Graphic Tablet, Track ball, Mouse.

Page No.
Date

- Standards :- Open GL, GKS, PHIGS

Images :- Still images
↳ Represented by bitmap or grid of pixels.
Input :- Camera, Scanner.

They are stored 1 bit perpixel in B/W
8 bit " " Gray scale
24 " " " true colour

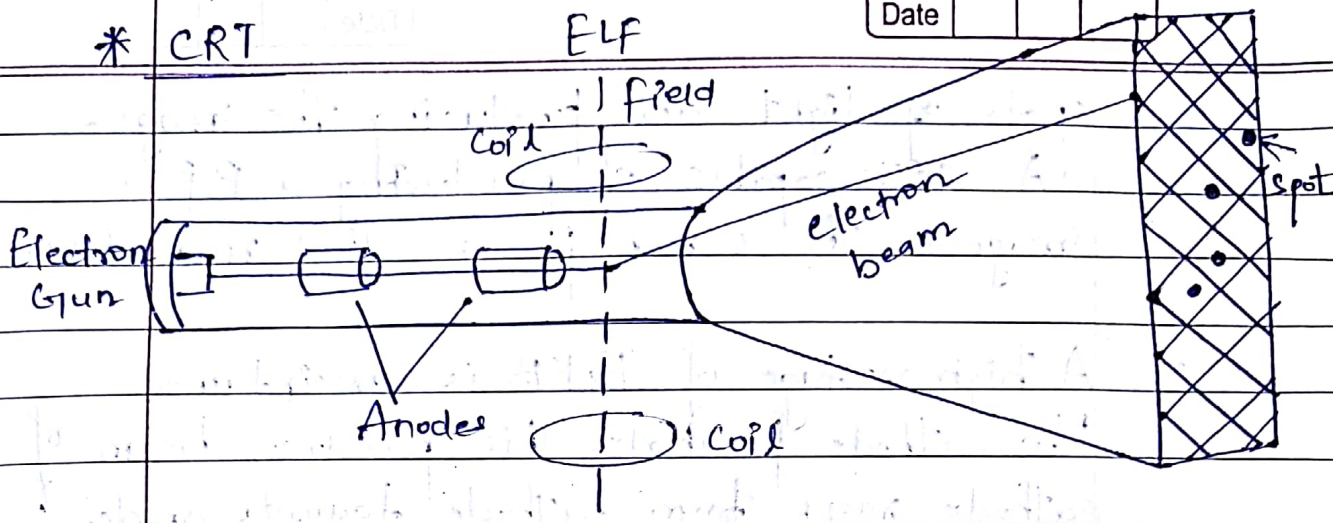
Audio signals :- Sync. analog signal
↳ input :- mic

Video :- can be series of single images
Eg. GIF

→ A majority of multimedia data is large and different.

Page No.
Date

* CRT



✓ The most common type of monitor in multimedia desktop system is picture tube called CRT (Cathode Ray Tube). It is essentially a vaccum sheet glass containing two electrode :- Cathode (-ve) & Anode (+ve)

✓ The front face of tube is created with phosphorus in the form of rectangular grid of large number of box. The material phosphour has a property of emmiting glow of light when it is hit by a charged particle like electron.

✓ The phosphor dots called pixels i.e. sort of picture elements that are responsible for producing an image on monitor. Other +ve electrodes called are G_1, G_2, G_3 that are located near the cathode, near the neck of CRT, two coils of wire known as the deflection coil.

✓ Electron beams are used, ~~to~~ generated from the cathode are made to hit phosphour & generating

spots of light and producing the image. A CRT capable of producing a B/W image on the screen is called monochrome.

A high voltage of 18KB is maintained b/w cathode & anode. This produces beam of cathode rays from cathode towards anode.

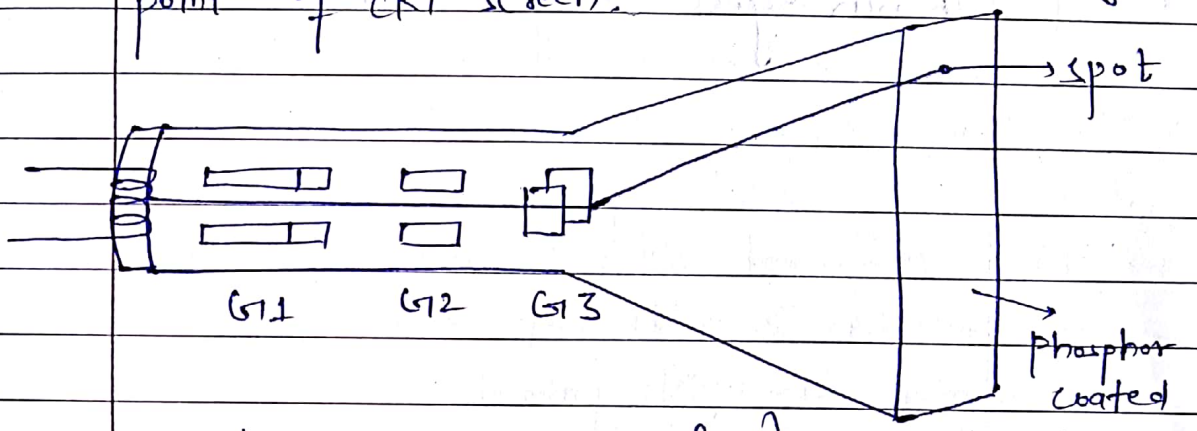
The beam of electron is controlled by three +ve terminals. The control gate G1 helps to control the amount of e⁻(s) in the beam.

The acceleration grid G2 provides accln of e⁻(s) in forward dirⁿ & focusing grid G3 focuses the beam to a single point be on the screen ahead, so that the dia of the beam is equal to the dia of single phosphour dot, a single glowing spot of light is created at the centre of screen.

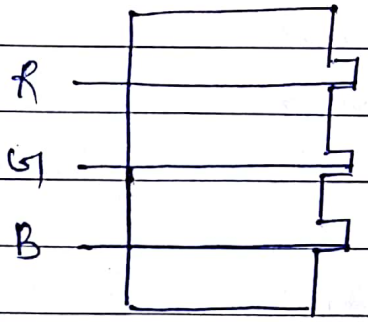
The spot of light is known as glowing pixels. When current flow through the collection coil an electric field produced interact with e⁻ beam (cathode rays). Thereby, deflecting in front of original path to another point Y.

One of the coils called horizontal deflection coil moves the beam horizontally across the screen & other coil called vertical

deflection coil move the beam vertically along ht. of screen. When both these coils are energised, required proportions of cathode rays can be moved in any direction. Thus generating single spot of light at any point of CRT screen.



Black and white (B & W)



RGB

✓ The working of color CRT is similar to monochrome CRT (B & W) except that here each pixel consist of three dots instead of one & is called triad. These processed phosphors produce light of color (Red, Green & Blue) & are called primary colours.

✓ Corresponding to the three dots, the cathode ray beam from the electrode also called

Page No.
Date

electron gun each of which falls on the corresponding dots in various intensity. They produce diff^n proportions of three elementing color lights which mixed togethes to create a specific color in our eyes.

Page No.
Date

Video Adapter Card is an extension card that act as an interface b/w processor & the monitor.

It accepts data from processor through comm^n channel which is earlier ISA (industry standard Architecture) were used but now a days, it is replaced by PCI (peripheral component interconnects).

✓ The card comm^n with monitor via adapter cable through 15 pin connectors. The digital data requires updating an image on the screen is generated by central & consist of RGB values of each pixel on the screen.

✓ RGB pixel is known as pixel attribute. value of each

✓ for 8 bit img, each pixel is represented digitally by 8 bit binary no. The 8 bit no's determine the intensity of coloured pixels.

✓ The adapter interprets these intensity values & translate them into corresponding voltage level to dry the electron gun.

Page No.
Date

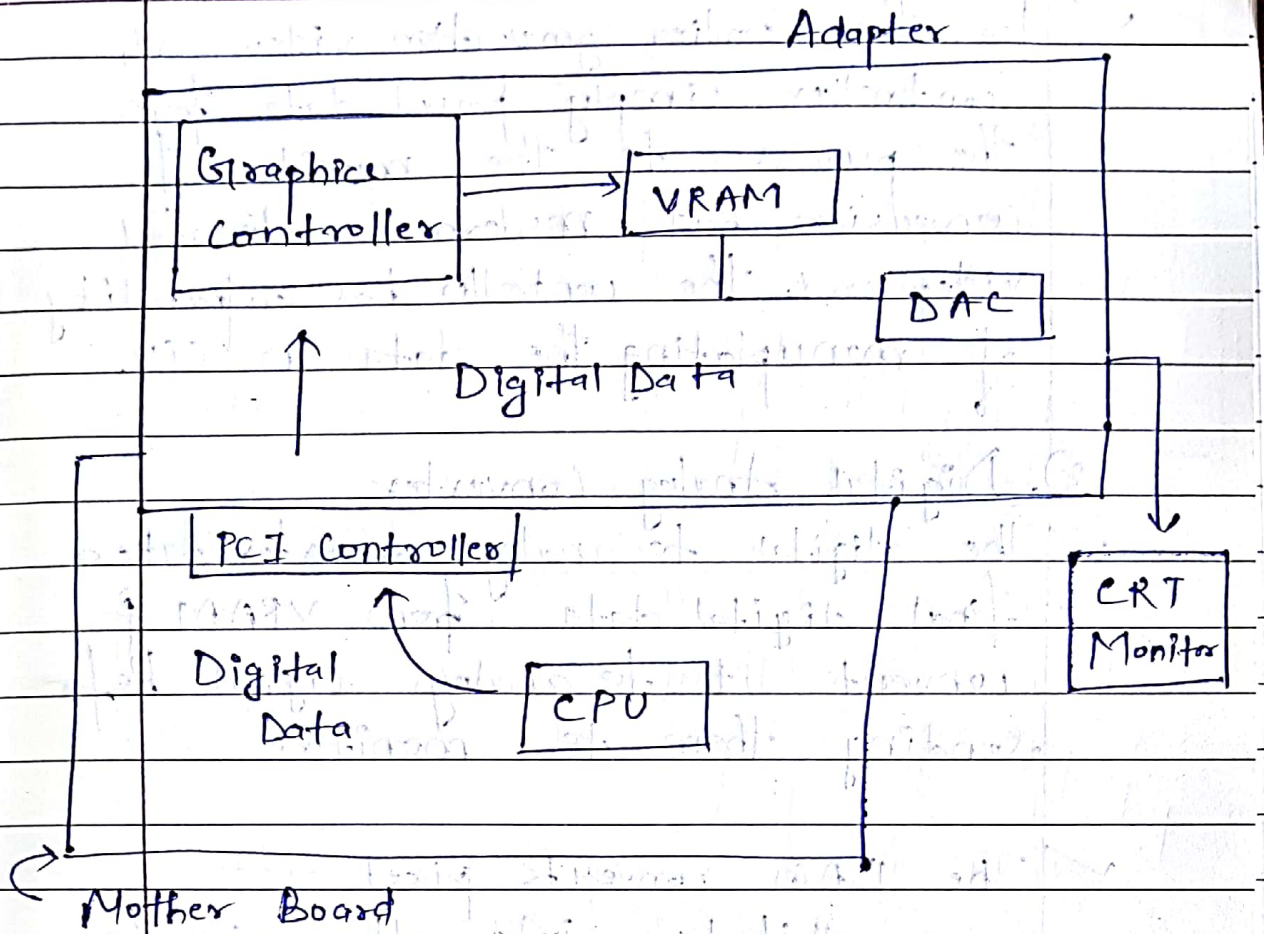
✓ The intensity signal along with synchronisation signal for positioning the e⁻(s) begin being at the location of pixels are fed to the monitor from the adapter through video cable.

✓ The earlier generation video card specifically transfer weak data to the monitor is called flat video cards.

Eg. Adapter.

✓ Modern video card can additionaly manuplate video attributes w/o help from processors.

✓ Due to this additional capability, the modern video cards activies computing load from processor and can display scenarios quickly due to presence of dedicated processor, card itself.
chip



Component of video card :-

1). Display Memory

✓ It is also known as video RAM (VRAM). A Bank of memory within the adapter card used for storing pixel attributes. Initially used for storing the image data from the CPU & later used by adapter to generate its signals. (RGB)

2). Graphic Controller

✓ A chip within the adapter card to coordinate the activities of all other components of the card.

Page No.
Date

✓ for the earlier generation video card, controller simply passed data from the processor to the monitor for conversion but modern accelerated video card, the controller has capability of manuplating the data in CPU.

3). Digital Analog Converter

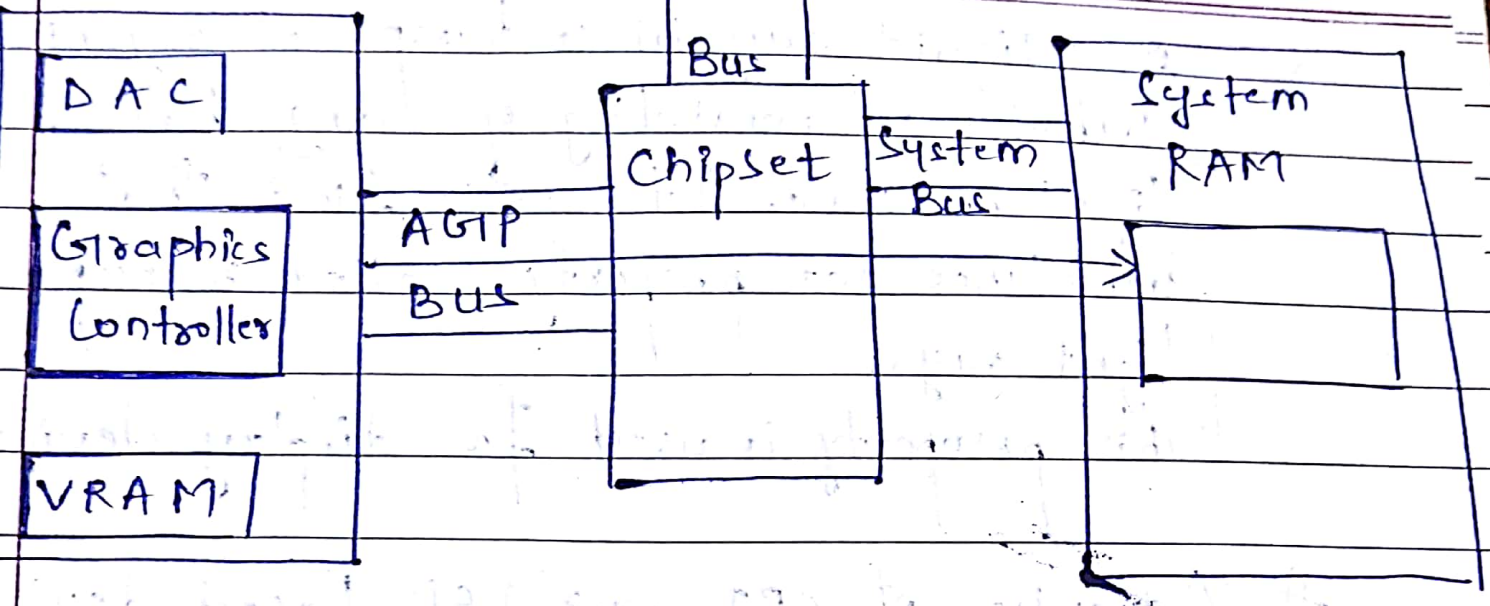
✓ The digital to analog converter take a final digital data from VRAM & converts it to analog signal before sending them to monitor.

✓ The DAC converts pixel —
— attributes into voltage signals which can be applied to CRT to generate electron beams.

4). Accelerated Graphics Modes

✓ It removes pressure from existing PCI by providing additional high speed bus exclusively for transfering video data with the 1130 Mbps, with improvement of 1200 Mbps PCI.

✓ This allows large amount of data to be exchanged b/w processor & adapter card within short time.



DATA FLOW USING AGTP

Page No.
Date

✓ A liquid crystal is transparent organic substance consisting of long, broad, light molecule. It was find that substance has property called manuplating light rays.
This property is used for display devices.

Advantage of CRT over LCD based devices:-

✓ Less bulky & occupies approx. $\frac{1}{3}$rd of volume for comparable viewing areas.
That why it is suitable for portable devices like,
laptop, mobile phones.

✓ Power consumption :- It consumes much less power than CRT. So, can be operated by batteries as well.

→ LCD monitor is a collection of LCD elements, each element generating a pic. on screen analogus to phosphour dot on CRT screen.

→ LCD elements consist of 2 major components :- A layer of liquid crystal sandwiched b/w two finely grooved surfaces with their grooves ⊥ to each other

Date

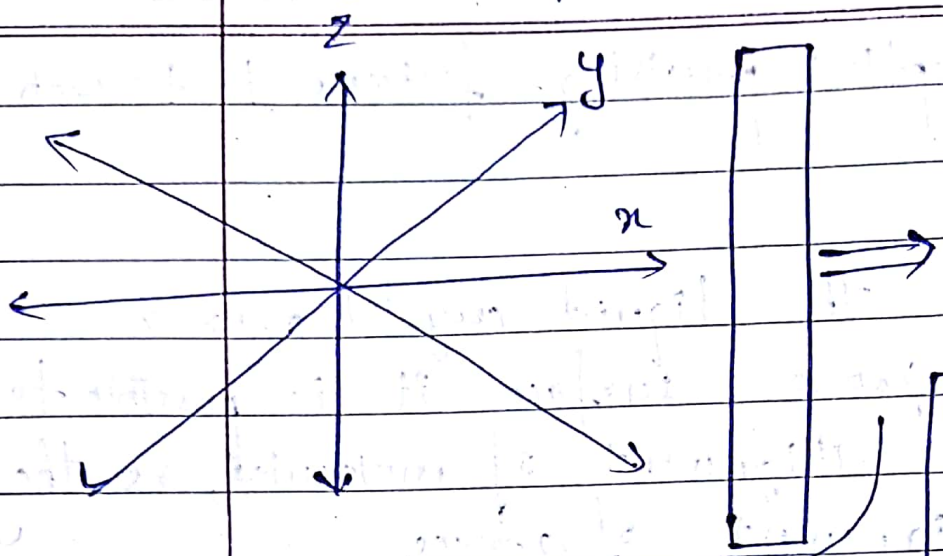
- Two optical polarising filters ⊥ to each other.

→ By following the liquid crystal over a finally groove surface it is possible to control the alligment of molecules, as they follow, allignment of grooves.

→ A layer of liquid crystal material is placed in the container with two finally grooves surfaces, whose groove surface ⊥ to each other.
Thus, molecules over these surfaces are ⊥ to each other & those which are at intermediate layers are twisted by intermediate angle.

→ An optical polarising filter act as a net of finally parallel lines blocking all lights except those, following in plane parallel to the lines.
Lines in this condition is said to be polarised.

Light source emitted from along all 3 planes.



Page No.
Date

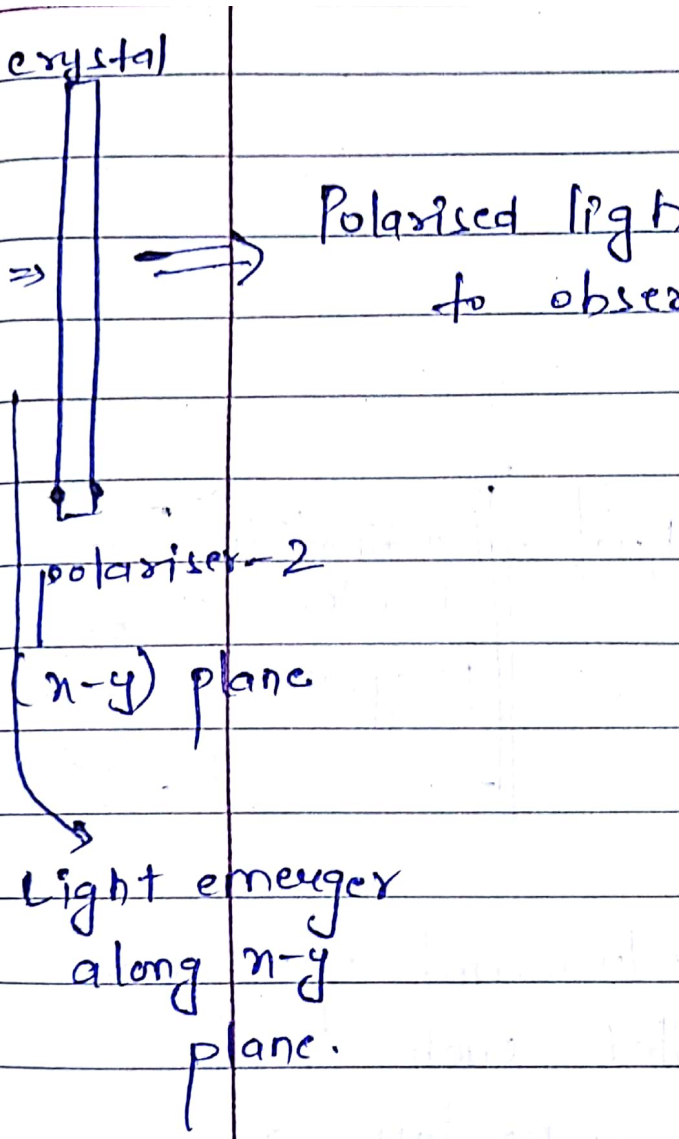
Light gets twisted from (x-z) plane to (x-y) plane by liquid Liquid Crystal whose molecular arrangement twists

polariser -1
(x-z) plane

Electrical contact open

single plane of light along x-z plane

⇒ A polariser can isolate a beam of light from collection of second polariser.



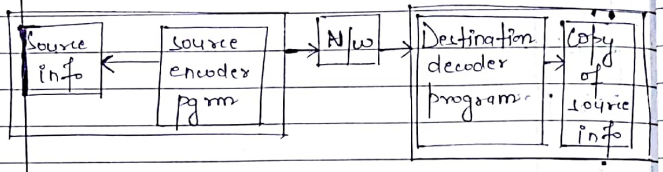
Types of LCD Devices :-

* Ranjan Parekh. [Thursday Test]

Page No.
Date

COMPRESSION

→ Source encoder
→ destination decoder



Lossless Compression :- reduce amt of source info to be transmitted such that during compression, there is no loss of information.

↳ → Reversible
↳ → transfer of text file over network.

Lossy Compression :- Aim is normally to reproduce an exact copy of source info after decompression but rather a version.
→ Ex transfer of digitised images & audio & video.

→ Entropy Encoding.
is lossless & independent of type of info being compressed.

↳ Run length encoding :- when source info comprises long substring of same character or binary digit
↳ String is transmitted in form of diff^n set of codewords.
↳ destination knows the set of codewords being used, it can interpret each codeword.
↳ o/p the no. of char or bit.
↳ Ex. Scanning typed docs : long strings of 0's and 1's.

000011111100111 — : o/p of scanner
0,4,2,5,0,2,1,2 — —
↳ individual decimal digit sent in binary form & is fixed no. of bits per codeword.

→ Statistical Encoding :- use of variable length codewords is not easy. The dest^n must know the set of codewords being used by the source.
↳ for decoding to be correct, prefix property is req'd : using set of variable length codewords ; shortlisted codeword for most frequently occuring symbol.

↳ Huffman encoding

↳ minimum avg. no. of bit reqd to transmit a particular source bno. stream is called entropy of the source.

Shannon formula:

$$\text{Entropy, } H = -\sum_{i=1}^{n} P_i \log_2 P_i$$

n → no. of diff. symbols in source stream.

P_i → probability of occurence.

→ Font

↳ font appearance

✓ font name: description of character appearance stored in font files.

✓ vector format: mathematical description of characters.

✓ true type: appearance of font remain same.

✓ bit map: character is described as collection of pixels.

→ Font size & style

↳ Changing the horizontal gap b/w the characters called kerning & vertical gap b/w of lines of text called leading.

Insertion of Text

↳ using keyboard

↳ copy & paste

↳ OCR

file formats

↳ Txt (text): data encoded using ASCII & unicode.

↳ Doc (document): not considered a doc exchange permit.

↳ RTF (Rich Text format): by microsoft cross platform document exchange, human readable codes similar to HTML code.

↳ PS (postscript): Page description language for desktop publishing. HLL that can describe contents of a page such that it can accurately display on o/p devices.

* Average no. of bit/codeword $= \sum_{i=1}^{n} N_i P_i$

Eg: Six diff. chars M, F, Y, N, O & I each is relative frequency of occurence of 0.25, 0.25, 0.15, 0.15 & 0.12 & resp.

If the encoding algo uses the following set of codewords,

M=10, F=11, Y=010, N=011, O=000, I=001

Compute:-

1) avg. no. of bits per codeword

Page No.
Date

(b) ?

$$\sum_{i=1}^{6} N_i P_i$$

(no. of bits for that char) × (4 chars having same value)

$= (0.25 \times 2) + (0.125 \times 3) \times 4$ $= 2(0.25) + 4(0.125)$

$+ 2(0.25)$ $= 0.5 + 0.500$

$= 0.5 + 0.5 + 12 \times 0.125$ $= 1$ ✓

$= 1 + (1.500) = 1 + 1.5 = 2.5$ ✓

(ii) Entropy of source :

$$= -\sum_{i=1}^{6} P_i \log_2(P_i)$$

$= -(2(0.25 \log_2 0.25) + 4(0.125 \log_2 0.125))$

$= 2.5$

(iii) min. no. of bits req. assuming fixed length codewords : 6 diff chars

: 3 bits

✓ (8 combinations)

Source encoding : exploits particular property of source info to produce alternative form of representation that is either a compressed version of original form or more amenable.

↳ Differential encoding

where amplitude of value or signals covers large range but diff. in amplitude b/w successive values/signals is small.

↳ set of smaller codewords used.

↳ can be lossy or lossless ; depends on no. of bits used to encode difference values.

↳ Transform encoding

→ transforming source info from one form into another, other form lending itself more readily to the application of compression

↳ no loss.

Text Compression :

✓ text is represented as strings of char selected from a defined set.

✓ Compression algo associated text must be lossless as a loss of any char can modify the meaning of complete string.

✓ Entropy encoding, statistical encoding.

uses single characters as basics of deriving an optimum set of codewords.	uses variable length strings of characters.
Eg. Huffman ; arithmetic coding algos.	Eg. Lempel-Ziv algo.

Page No.
Date

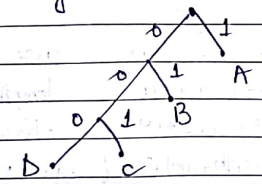
- Types of coding for text

Static	Dynamic
→ text to be compressed has known characteristics in terms of char used & their relative frequency of occurence.	→ for more general applications → optimum set of codewords likely to vary from one transfer to another.
→ optimum set of variable length codewords is derived, shortest codewords for most frequently occuring chars.	→ receiver is able to dynamically compute the same set of codewords being used at each transfer.

↳ Static Huffman Coding

Char string to be transmitted is first analysed & the character types & their relative frequency determined.

→ create an unbalanced tree
→ binary tree, branches having value 0 & 1 (left, right)



Page No.
Date

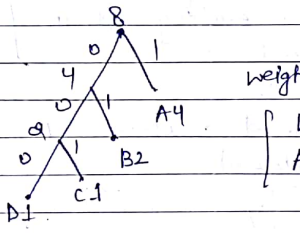
A = 1 (4)
B = 01 (2)
C = 001 (1)
D = 000 (0)

(AAAABBCD) ← no of bits

4×1 + 2×2 + 1×3 = 14 bits

char freq. + 3×1

→ codewords are determined by tracing the path from the root node out to each leaf and forming a string of binary values associated with each branch traced.



weight order =

D1	C1	2	B2	4
A4	8			

↳ optimum tree if the resulting list increments in weight order.

Every character is encoded has a variable no. of bits, the received bitstream must be interpreted in a bit oriented way.

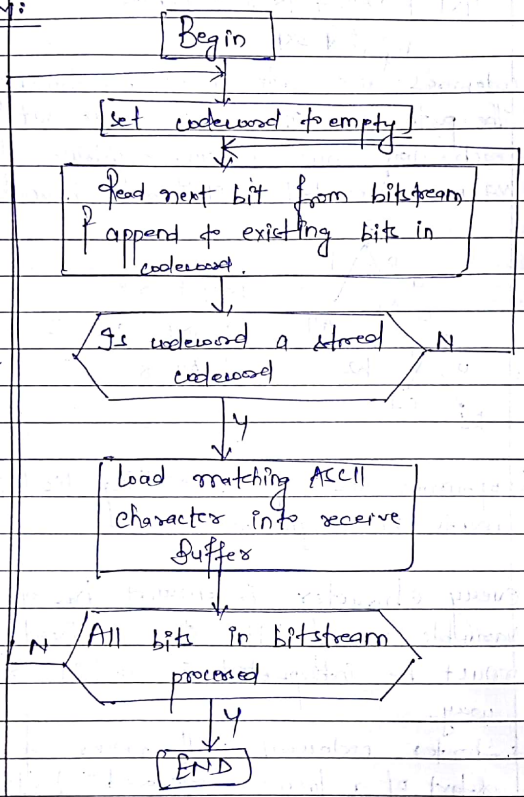
↳ shorter codeword will never form the start of a longer codeword : prefix property

Page No.
Date

Prefix Property:-

Received bitstream can be decoded simply by carrying out a recursive search bit by bit until valid codeword is found.

ALGORITHM:



Page No.
Date

→ assumes a table of codeword at the received & it also holds the corresponding ASCII codewords

↳ Bitstream variable holds the received bitstream & codeword variable holds the bits in each clk.

Dynamic Huffman Coding

- encoder & decoder build the Huffman tree & hence the codeword table dynamically as the characters are being transmitted/received.
- If the char to be transmitted is currently present in the tree, its codewords is determined & sent in the normal way.
- If char isn't present or it is 1st occurence, it is transmitted in its uncompressed form.
- Encoder updates its Huffman tree by introducing the new char into the tree.
- receiver can determine char from the codeword & also make modifications in its tree copy, so that it can interpret the next codeword.

Page No.
Date

• Arithmetic Coding w=0.1, •=0.1

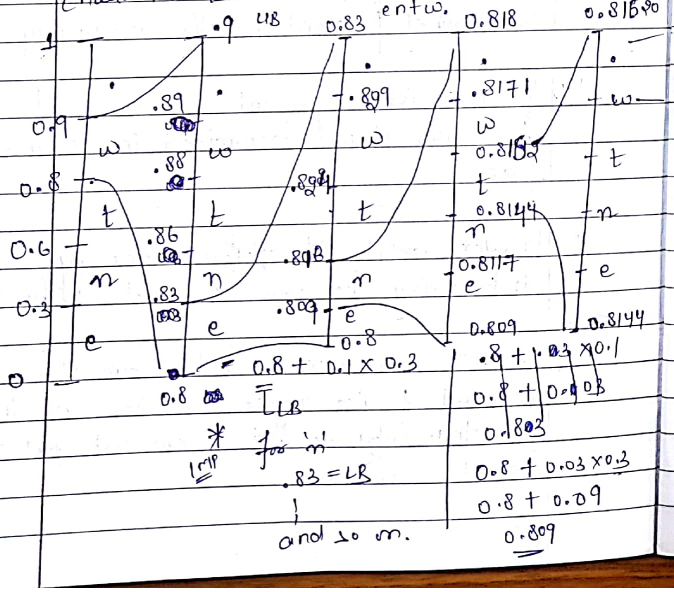
"went." e=0.3, n=0.3, t=0.2, ~~0.2~~

procedure:

1) Divide the numeric range 0 to 1 into no of diff^n symbol present in the msg.

2) Expand the first letter to be coded along with the range for the subdivided ~~for this range~~ into no. further of symbols.

3) Repeate the procedure, until termination character is encoded.



0.8 + 0.1 × 0.3

I_{LB}

* for 'n'

.83 = LB

and so on.

0.8 + 0.03 × 0.1

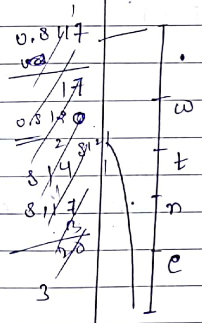
0.8 + 0.003

0.803

0.8 + 0.03 × 0.3

0.8 + 0.09

0.809



probability

Page No.
Date

✓ * lower limit + d (P of symbol)

(upper bound − lower bound)

~~0.009 × 0.3~~

0.809 + 0.009 × 0.3

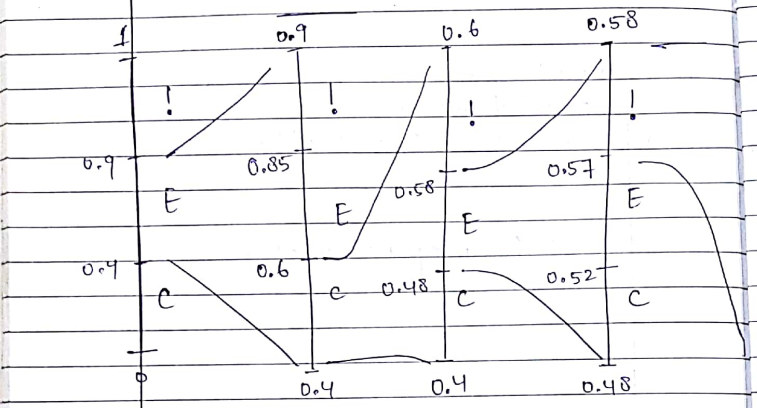
0.809 + 0.0097 = 0.8117

✓ Here, sequence matters.

Page No.
Date

Q. c = 0.4 E = 0.5 ! = 0.1

Decode, 0.572 → Range is decided acc. to this

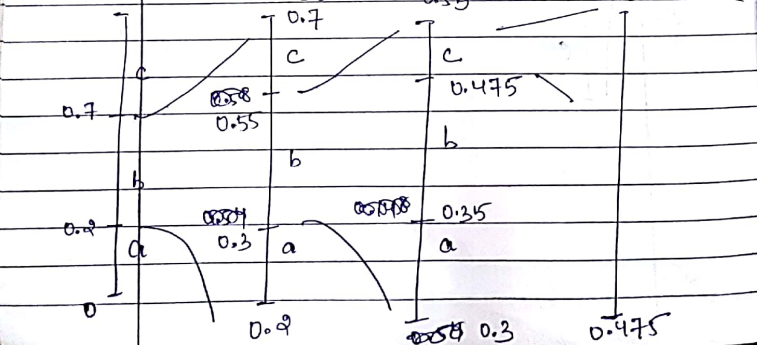


0.4 + 0.5 x 0.4
0.4 + 0.2 = 0.6

= ECE!

Q. Encode, a = 0.9, b = 0.5, c = 0.3

"bbc"



← (Q. Decode 0.49)

0.5 + 0.2 x 0.2 = 0.5 + 0.04
0.54

Page No.
Date

0.16 + 0.5 x 0.54 = 0.16 + 0
0.5 + 0.5 x 0.16 = 0.5 + 0.08 0
0.58

0.58
0.579
0.574
0.57

0.04 + 0.2 x 0.54
= 0.04 + 0.108 = 148
= 0.04 + 0.5 x 0.432
= 0.04 +

0.2 + 0.5 x 0.2 = 0.2 + 0.10
= 0.3

0.3 + 0.5 x 0.5 = 0.3 + 0.25
= 0.4

0.3 + 0.25 x 0.2 = 0.3 + 0.05
0.36

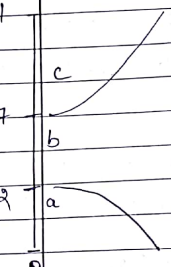
0.35 + 0.25 x 0.5 = 0.35 + 0.125
= 0.475

(0.475 − 0.55)
Required range

0.55 + 0.475 / 2
= 1.025 / 2 = 0.5125

0.57 + 0.01 x 0.4
0.57 + 0.004
0.574
.574 + 0.01 x 0.5
.574 + 0.005
9

0.49



ADAPTIVE DICTIONARY TECHNOLOGY

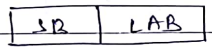
(1) LZ77 / LZ1 / Sliding window

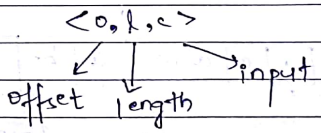
When the i/p type is taken for compression, it starts building the dictionary.
Thus the symbols, words & phrases which have been scanned previously appears again in the same set then a token corresponding to the entry of that phrase on the dictionary is generated.

- LZ77 scheme use that text window which is divided into 2 parts.

1) Scan Buffer / Search Buffer
2) Look Ahead Buffer

→ Search Buffer contains previously scanned text
→ Look Ahead Buffer contains upcoming text





Q. i/p cabracadabra with window size = 13 and LAB = 7.

⇒ SB = size − LAB
= 13 − 7 = (6)

Page No.
Date

Initially,

SB | LAB

| | | | | | c | a | b | ʜ | a | c | a | d a b ...

(data will keep moving on shifting)

<0,0,c>
<0,0,a>
<0,0,b>
<0,0,ʜ>
<3,1,c> → which didn't match d.

pst. ← length of pattern which matched.

→ One which is found, step for it is escaped.

→ check all duplicates, from right to left.

<2,1,d>

| | c | a | b | ʜ | a | c | a | | d | a | b | ʜ | a | |

a a c c

c | a | b | ʜ | a | c | a | d | a | b | ʜ | a | ʜ | ʜ |

found.

<7,4,ʜ>

length (abʜa)

Page No.
Date

| a | d | a | b | ʜ | a | ʜ | ʜ | a | ʜ | ʜ | a | d |

if continuous, we use look ahead buffer also.

<3,5,d>

c
c a
c a b
c a b ʜ
c a b ʜ a c
c a b ʜ a c a d
c a b ʜ a c a d a b ʜ a ʜ ← length given (only 2) so, after 1, rotate.
c a b ʜ a c a d a b ʜ a ʜ a
c a b ʜ a c a d a b ʜ o ʜ a d.

ʜ a ʜ
1 2 3
4 5

Q. SB=7, LAB=4

i/p abracadabrad

	7	6	5	4	3	2	1	1	2	3	4
							a	b	r	a	c
<0,0,a>						a	b	r	a	c	a
<0,0,b>					a	b	r	a	c	a	d
<0,0,r>			a	b	r	a	c	a	d	a	b
<3,1,c>		a	b	r	a	c	a	d	a	b	r
<2,1,d>	a	b	r	a	c	a	d	a	b	r	a
<7,4,d>	b	r	a	c	a	d	a	b	r	a	d

d

abracadabra

→ The search buffer contains the past of i/p msg being coded & look ahead buffer contains forth coming string of characters.

→ Size of SB & LAB remains fixed.

→ LAB is moved over the i/p string in forward dirn.

→ first character in LAB is checked if it is there in SB.

→ A pointer in search buffer do th. match from

→ The dist. from LAB is known as offset.

→ After the match, the ptr. is moved forward to check if next consecutive char match. from LAB.

→ The Total no. of consecutive characters matched is known as length of match.

→ The compressor attempts to search the longest match in LAB

→ After match has been found <offset, length, symbol>

symbol → corresponds to first char till which match is found.

⇒ Drawback: Searching for phrases is complex task and it is tedious affair as size of dictionary inc.