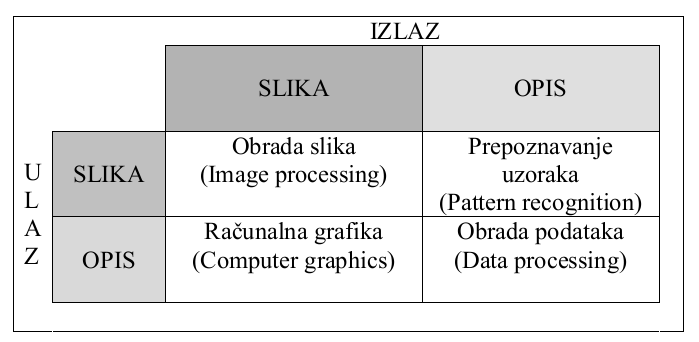
**ŠTO JE RAČUNALNA GRAFIKA?**

To su slike generirane računalom te alati (sklopovski i programski) potrebni za izradu slika. **Računalna grafika** (*computer graphics*) obuhvaća stvaranje, pohranu te uporabu modela i slika objekata. Ona se bavi sintezom slika na temelju računalnih modela stvarnih ili imaginarnih objekata. Modeli i objekti računalne grafike potječu iz različitih područja: prirode, znanosti, inženjerstva, arhitekture, apstraktnih područja,...

Obrnuti procesi analize scene i rekonstrukcije modela 2D i 3D objekata predmet su discipline koja se naziva **obrada slike** (*image processing*).

Obradu slike možemo podijeliti na podpodručja:

* **Poboljšanje slike** (*image enhancement*) - razvitak i primjena tehnika poboljšanja kvalitete slike (uklanjanje šuma) i povećanja kontrasta
* **Detekcija i prepoznavanja uzoraka** (*pattern detection and recognition*) - otkrivanje standardnih uzoraka na slici i pronalaženje devijacija (npr. optičko prepoznavanje alfanumeričkih znakova - optical character recognition, prepoznavanje otisaka prsta,...)
* **Analiza scene i računalni vid** (**scene analysis and computer vision**) - prepoznavanje i rekonstrukcija 3D modela scene na temelju više 2D slika (npr. industrijski robot sa senzorima za registriranje oblika, pozicije i boje dijelova na pokretnoj traci)

Računalna grafika i obrada slika bave se računalnom obradom slika i preklapaju se prvenstveno u dva područja:

* **Interaktivna obrada slika** - ulazni podaci zadaju se korištenjem izbornika i drugim grafičkim interakcijskim tehnikama kako bi se omogućilo i pomoglo vođenje različitih podprocesa za vrijeme kojeg se transformacije kombiniranih slika prikazuju u realnom vremenu. (primjer: skenirana fotografija dotjerana, odsječena i kombinirana s drugim prije objavljivanja).
* **Jednostavne operacije obrade slika** - često se koriste u računalnoj grafici kako bi pomogle u sintetiziranju slika modela. Neke transformacije i kombinacije sintetičkih slika uvelike ovise o operacijama obrade slike.

Osnovna značajka moderne računalne grafike je **interaktivnost**. Interaktivnost podrazumijeva mogućnost upravljanja sadržajem, strukturom i izgledom objekata i njihovih prikazanih slika od strane korisnika uz pomoć nekog ulaznog uređaja.

Interaktivna grafika omogućava:

* prikaz slika realnih objekata u 2D i 3D prostoru
* prikaz apstraktnih, sintetičkih objekata kao što su matematičke površine u višedimenzionalnim prostorima
* prikaz podataka bez konkretnih geometrijskih značajki (rezultati istraživanja).

Neki pojmovi RG:

* **Modeliranje** - predstavljanje oblika objekta
* **Renderiranje** - izgled objekta “Izraziti ili predstaviti nešto na određeni način” - LONGMAN Rječnik
* **Animacija** - kretanje objekata

Modeliranje

* Realni ili imaginarni objekti se opisuju pomoću matematike. Žičani model (wire frame) je najjednostavniji oblik modela.

Renderiranje

* Crtanje slike, boje, osvjetljenje, sjenčenje, tekstura površine, sjene te refleksije i prozirnost

Animacija

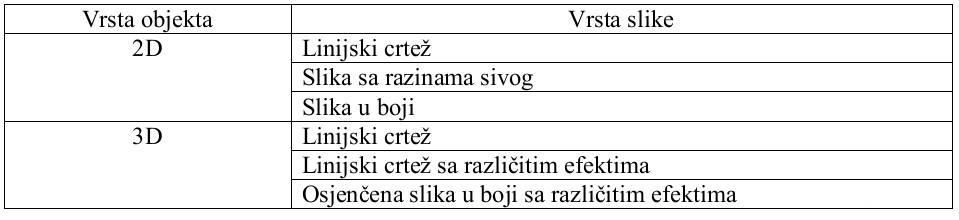
* Upravljanje pokretima objekta, fizikalni zakoni, biomehanika/kineziologija i specijalni efekti

Primjene računalne grafike: Film, simulacija, računalne igre, medicina (Medical Imaging), projektiranje pomoću računala (CAD), vizualizacija u znanosti (mogućnost gledanja izvan našeg vidnog područja), korisnička sučelja (GUI), interaktivno crtanje (brojčani podaci u slike; olakšava donošenje odluka), umjetnost, modeliranje terena i geografija, edukacija, uredska automatizacija i elektroničko izdavaštvo, upravljanje procesima, grafičko programiranje, prividna stvarnost,...

Klasifikacija primjena može biti prema :

1. tipu (dimenzionalnosti) objekta kojeg se prikazuje i vrsti dobivene slike.
2. tipu interakcije, koji određuje koliku kontrolu ima korisnik nad objektom i njegovom slikom.
3. ulozi slike, tj. koliko je zapravo slika svrha ili samo sredstvo za postizanje nekog cilja (umjetnost - slika je krajnji cilj, CAD – slika je samo prikaz geometrijskih svojstava objekta kojeg analiziramo).
4. logičkoj i vremenskoj vezi između objekata i njihovih slika (u nekom trenutku korisnik se bavi samo sa jednom slikom, sekvencom vremenski ovisnih slika, struktuiranim skupom objekata).

Klasifikacija primjena po tipu:



**ELEMENTI SLIKE**

**VEKTORSKI I GRAFIČKI RASTERSKI SUSTAVI**

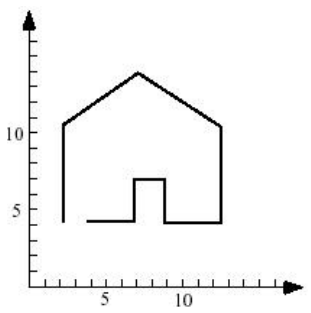
**KONCEPT INTERAKTIVNE GRAFIKE**

**Primitivi** su osnovni objekti od kojih se sastoji neka slika.

Primitivi mogu biti:

* polilinije
* tekst
* ispunjena područja (površine)
* rasterske slike

**POLILINIJE**

**Polilinija** je povezana sekvenca ravnih crta. Uvećanjem dijela slike se vidi da krivulja zapravo nije glatka već se sastoji od više ravnih segmenata. Slike napravljene od polilinija ponekad nazivamo i **linijski crteži**. Najjednostavnija polilinija je obična ravna crta koja je određena s početnom točkom (x1, y1) i krajnjom točkom (x2, y2). Primjer rutine za crtanje linije: drawLine (x1, y1, x2, y2). Specijalni slučaj:

segment se skraćuje na jednu točku: *drawDot (x1, y1)*. Polilinije možemo odrediti pomoću liste vrhova s njihovim koordinatama: (x0, y0), (x1, y1), (x2, y2), ..., (xn,yn). Na primjer, polilinija prikazana na slici zadana je nizom (2,4), (2,11), (6,14), (12,11), (12,4),... Naredba za crtanje polilinije npr. *drawPolyline (poly)* (*poly* - lista vrhova polilinije).

Važni atributi polilinija:

* boja
* debljina rubova
* način na koji su rubovi povučeni
* način na koji se rubovi spajaju

Uobičajeno je da svi rubovi jedne polilinije imaju iste atribute.

**TEKST**

Načini prikaza:

* tekstualni
* grafički

Tekstualni način:

* jednostavni ulaz/izlaz alfanumeričkih znakova
* koristi ugrađeni generator znakova
* znakovi se obično ne mogu postaviti proizvoljno na zaslonu

**Grafički način** (mod) prikaza omogućava prikaz većeg broja simbola bez ograničenja pozicije.

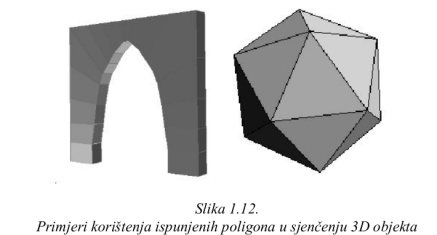
Primjer naredbe za ispis teksta: *drawString (x, y, string)*.

Uobičajeni atributi teksta:

* izgled
* boja
* veličina
* razmak
* orijentacija

Font je skup alfanumeričkih znakova specifična izgleda u određenom stilu i veličini.

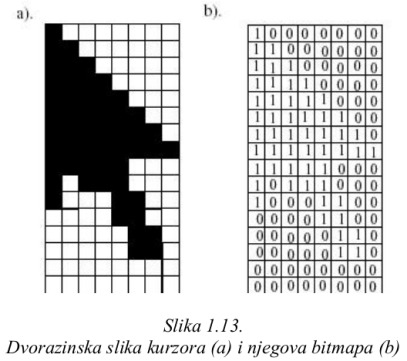
**ISPUNJENA PODRUČJA (POVRŠINE)**



Primitiv ispunjenog područja je obris ispunjen nekom bojom ili uzorkom. Rub ispunjenog područja je obično poligon iako to mogu biti i složenija područja. Primjer naredbe za crtanje ispunjenog poligona:

*fillPolygon (poly, pattern)*; *poly* su liste vrhova, *pattern* je opis uzorka. Pod atributima ispunjenih područja podrazumijevamo atribute rubova, uzorak i boju kojom ih ispunjavamo.

**RASTERSKE SLIKE**

Pravilno raspoređeni elementi fiksne veličine i oblika za koje je određena boja i nivo osvijetljenosti. **Elementi** su slikovni elementi ili pikseli. Ako su **slikovni elementi** dovoljno mali i gusto raspoređeni onda ih ljudsko oko neće prepoznati kao zasebne elemente, već će ih povezati u kontinuiranu sliku. Rasterska slika je u računalu pohranjena kao polje numeričkih vrijednosti. Ova matrica naziva se **bitmapa** (cb) ili **pixmapa** (boja). Slike se pohranjuje zapisivanjem pozicije, boje ili osvijetljenosti (razine sivog) svakog slikovnog elementa.

* **bitovna matrica** (*bitmap*) - matrica čiji elementi (1, 0) predstavljaju svjetloću (ili boju) odgovarajućih elemenata pravokutnog rasporeda osvjetljivih točaka zaslona (slikovnih elemenata) u dvorazinskom sustavu (informacijski kapacitet 1 bit/piksel)
* **matrica slikovnih elemenata** (*pixmap - pixel map*) - matrica čiji elementi predstavljaju boju odgovarajućih elemenata pravokutnog rasporeda osvjetljivih točaka zaslona (slikovnih elemenata) u višerazinskom sustavu (informacijski kapacitet n bit/piksel)

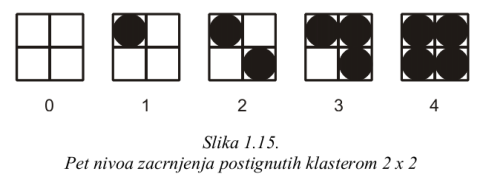
Memorijski zahtjevi:

* Crno-bijele slike: jedan bit (0 – bijelo, 1 – crno)
* 256 razina sivog – 8 bitova

Slike u boji još su memorijski zahtjevnije. Boja određenog slikovnog elementa određena je sa tri broja koja predstavljaju razinu crvene, zelene i plave boje čijim miješanjem nastaje željena boja

* «true color» mod – 24 bita za svaki slikovni element tj. po jedan bajt za svaku komponentu boje

Broj bita po slikovnom elementu - dubina boje.

Prilikom prikazivanja pohranjene slike, potrebno je isčitavati podatke za svaki slikovni element istim redoslijedom kojim su i zapisivani. Na ovaj način nije potrebno zapisivati poziciju svakog slikovnog elementa slike i pojednostavnjeno je zapisivanje i rekonstrukcija slike. Obično se slika sprema red po red, s lijeva na desno i od gore prema dole. Redove slikovnih elemenata nazivamo **rasterske scan linije**.

**Monokromatski rasteri** se također mogu koristiti za prikaz kontinuiranih tonskih prelaza korištenjem grupe tehnika pod nazivom polutonske transformacije (halftone approximation). Ove tehnike, u osnovi, koriste ograničenja prostorne razlučivosti oka na način da jedan slikovni element prikažu grupom slikovnih elemenata u bitmapi, takozvanim **klasterom piksela**.

**VEKTORSKI I RASTERSKI GRAFIČKI SUSTAVI**

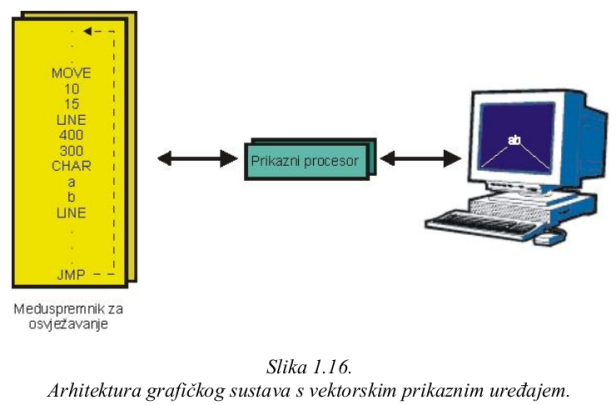
Ovisno o arhitekturi grafičkog sustava, možemo ih podijeliti na dvije osnovne grupe:

1. vektorski grafički sustavi (sustavi s proizvoljnom putanjom otklonjene zrake)
2. rasterski grafički sustavi (sustavi sa sekvencijalnom putanjom otklonjene zrake)

**Vektorski grafički sustavi**

Masovno u upotrebi od sredine šezdesetih do sredine osamdesetih godina 20 – og stoljeća (“vektor” = crta). Crta koja povezuje dvije (proizvoljno) odabrane točke na zaslonu osnovni je element grafičkog prikaza (random scan). Putanja zrake određena je slijedom naredbi iz prikazne liste ili prikaznog programa i povezuje krajnje točke pojedinih crta.

Tipični vektorski sustav sastoji se od sljedećih dijelova:

* prikazni procesor priključen kao U/I uređaj na glavni procesor (CPU)
* prikazna privremena memorija (display buffer memory)
* vektorski generator
* prikazni uređaj

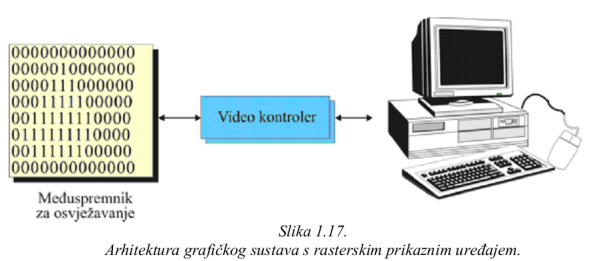
**Prikazna privremena memorija** ili međuspremnik za osvježavanje sadrži prikaznu listu ili prikazni program. Ta lista ili program sastoji se od niza naredbi za crtanje točaka i linija sa koordinatama krajnjih točaka (x, y) kao i naredbi za ispis karaktera. Naredbe za crtanje točaka, linija i karaktera interpretiraju se u prikaznom procesoru koji digitalne koordinate točaka prosljeđuje vektorskom generatoru. Vektorski generator pretvara digitalne koordinate u analogne vrijednosti napona za otklonski sustav katodne cijevi.

Nedostaci:

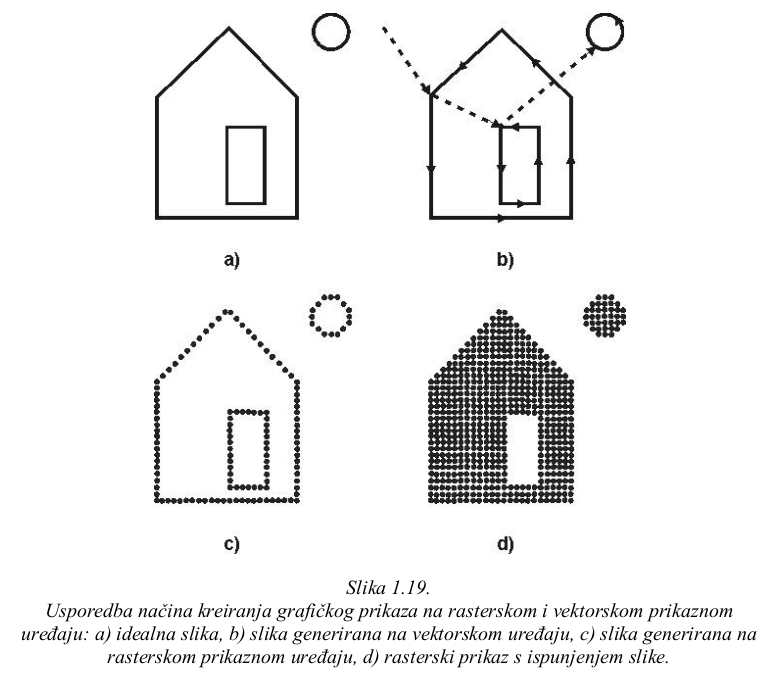
* Vektorski grafički sustavi nemaju mogućnost prikaza ispunjenih površina, manipulaciju bitovima i tablicama

Prednosti:

* Mogu ostvariti veće rezolucije od rasterskih sustava
* Mogu prikazivati glatke kose crte

**Rasterski grafički sustavi**

Početkom sedamdesetih godina 20-tog stoljeća dolazi do razvoja jeftine rasterske grafike zasnovane na televizijskoj tehnologiji. Rasterski prikazni uređaji pohranjuju primitivne oblike u prikaznu privremenu memoriju u obliku njihovih osnovnih sastavnih slikovnih elemenata - piksela. Cjelovita slika prikazuje se na rasteru koji predstavlja niz paralelnih horizontalnih redova slikovnih elemenata.



Za razliku od vektorskog prikaza, kod rasterske grafike potrebno je eksplicitno pohraniti cijelu sliku od N x M slikovnih elemenata. Realizacija putanje zrake kod rasterskog načina prikaza mnogo je jednostavnija i jeftinija nego što je to slučaj kod vektorskih uređaja gdje je potrebno osigurati vrlo veliku preciznost vektorskih generatora. Pojava jeftine RAM memorije - omogućila prevlast rasterske tehnologije početkom sedamdesetih godina.

Osnovne prednosti rasterske grafike:

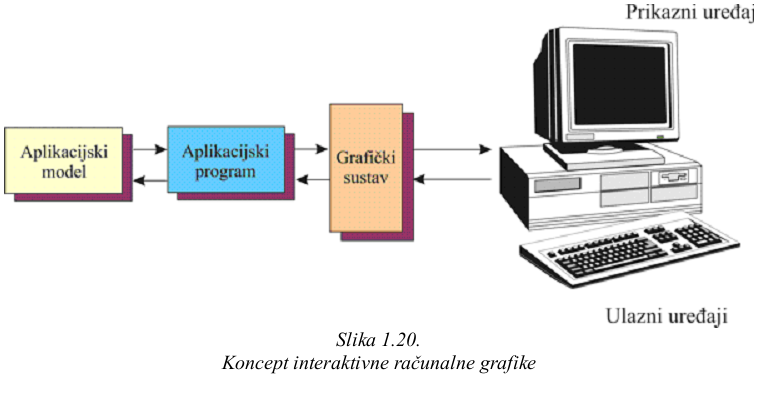
* cijena
* mogućnost prikaza ispunjenih područja
* neovisnost procesa osvježavanja slike o njenom sadržaju

Rasterski grafički sustavi nisu u stanju prikazati glatku, neprekinutu liniju ili krivulju već samo njihove aproksimacije.

**Aliasing** - pojava koja nastaje kao posljedica pogreške uzorkovanja.

Tehnike njenog odstranjivanja - **antialiasing** (primjena gradacije osvjetljenja susjednih slikovnih elemenata na rubovima primitiva).

**KONCEPT INTERAKTIVNE GRAFIKE**



Na sklopovskoj razini računalo prima ulazne informacije od interakcijskih uređaja i prenosi slike prikaznom uređaju.

Programska razina može se podijeliti u tri dijela:

* aplikacijski model
* aplikacijski program
* grafički sustav

**Aplikacijski model**

Aplikacijski model predstavlja podatke ili objekte koji se prikazuju na zaslonu. Funkcije aplikacijskog modela su sljedeće:

* sadrži sve podatke, objekte i odnose među njima koje koriste prikazni i interakcijski dio aplikacijskog programa ili negrafički moduli za obradu podataka;
* predstavlja objekte kombinacijom podataka i proceduralnih opisa neovisnih o prikaznom uređaju.

Aplikacijski model sadrži sljedeće:

* primitivne oblike (točka, crta, višekutni likovi u 2D ili 3D, različitih prostornih ploha u 3D...) od kojih je sastavljen model objekta;
* atribute objekata (vrsta crte, boja, struktura površine...);
* odnose među objektima i dijelovima objekata (povezivanje, spajanje...);
* podatke o položaju objekata i dijelova objekata.

**Aplikacijski program**

Prima i obrađuje informacije od aplikacijskog modela i korisnika. Razvija aplikacijski model i generira skup grafičkih izlaznih naredbi koje sadrže geometrijski opis onoga što treba prikazati kao i načina na koji se pojedini objekti prikazuju. Aplikacijski program pretvara opis dijela modela koji treba prikazati u pozive procedura ili naredbi grafičkog sustava koji se koristi za stvaranje slikovnog prikaza u dva koraka:

1. pretraživanje aplikacijske baze podataka i izlučivanje podataka nužnih za prikaz odabranog dijela modela,
2. pretvorba podataka u format prikladan za ulaz u grafički sustav.

**Grafički sustav**

Proizvodi sliku na temelju detaljnog opisa u obliku grafičkih naredbi koji generira aplikacijski program i prenosi ulaznu informaciju od korisnika aplikacijskom programu na obradu. Posreduje između aplikacijskog programa i prikaznog uređaja.

Zadaće:

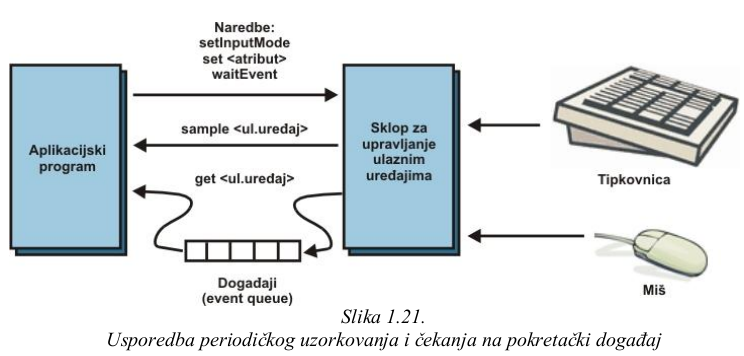
* izlazna transformacija - transformira objekt u aplikacijskom modelu u slikovni prikaz modela;
* ulazna transformacija - transformira korisničko djelovanje u ulaznu informaciju za aplikacijski program na temelju kojih aplikacijski program djeluje na promjenu modela i/ili slike.

Sastoji se od skupa izlaznih potprograma koji odgovaraju različitim primitivnim oblicima, atributima i drugim elementima. Ovi potprogrami tvore biblioteku grafičkih potprograma ili grafički paket i mogu se pozivati iz programskih jezika visoke razine (C, Pascal, LISP). Potprogrami pokreću prikazne uređaje i na taj način generiraju slikovni prikaz geometrijskih primitivnih oblika i atributa specificiranih aplikacijskim programom.

**Upravljanje interakcijom**

Aplikacijski program može od grafičkog sustava zatražiti:

1. Periodičko uzorkovanje ulaznih uređaja - Aplikacijski program periodički postavlja upit o trenutnoj vrijednosti logičkog ulaznog uređaja i nastavlja s radom. Uzorkovanje ove vrijednosti se obavlja bez obzira da li je došlo do promjene vrijednosti ulaznog uređaja).
2. Čekanje u određenom stanju na pokretački događaj (*interrupt - driven interaction*) - rad aplikacije se prekida nekim od događaja. Tipičan oblik interakcije: petlja pokretana događajima (*event-driven loop*).



Primjer pseudokoda petlje pokretane događajima:

stvori početni prikaz na temelju aplikacijskog modela

while(!kraj)

{ /\* korisnik nije odabrao opciju kraj \*/

omogući izbor naredbenih objekata

/\* program čeka beskonačno na korisnikovo djelovanje \*/

čekaj korisnikov izbor

switch(izbor)

{

obradi izbor i upotpuni naredbu ili izvrši upotpunjenu naredbu te promijeni model ili prikaz na

zaslonu po zahtjevu

}

}

**GRAFIČKI CJEVOVOD**

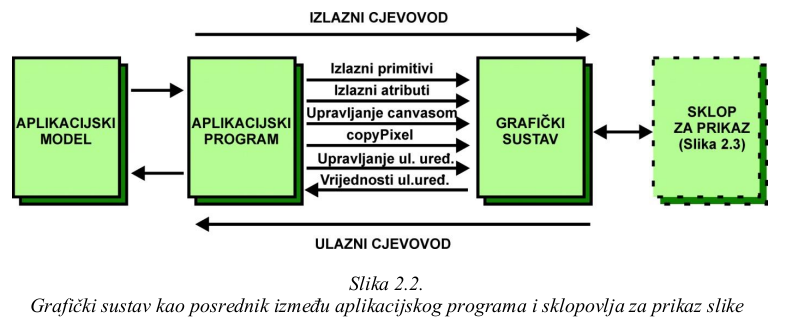
**OSNOVNI ALGORITMI RASTERSKE GRAFIKE**

**GRAFIČKI SUSTAV**

Sastoji se od skupa izlaznih potprograma koji odgovaraju različitim primitivnim oblicima, atributima i drugim elementima. Ovi potprogrami tvore biblioteku grafičkih potprograma ili grafički paket i mogu se pozivati iz programskih jezika visoke razine (C, Pascal, LISP). Potprogrami pokreću prikazne uređaje i na taj način generiraju slikovni prikaz geometrijskih primitivnih oblika i atributa specificiranih aplikacijskim programom.

**Grafički sustavi rasterske grafike**

* GKS, SRGP, PHIGS, OpenGL,...

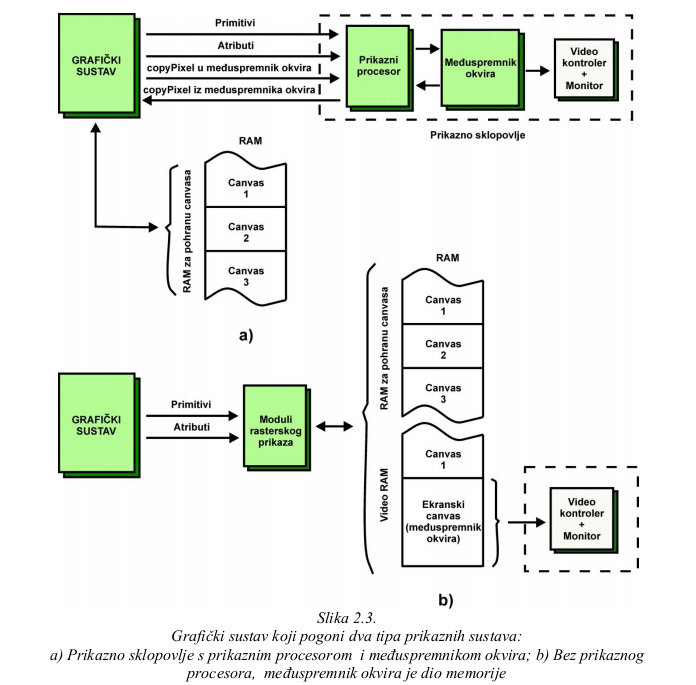
Grafički sustaviaproksimiraju matematičke (idealne) primitive opisanih vrhovima u Kartezijevu koordinatnom sustavu, pomoću skupova piksela odgovarajuće razine svjetloće i/ili boje. Ovi pikseli su pohranjeni kao bitovna matrica (bitmap) ili matrica slikovnih elemenata (pixmap) u memoriju centralnog procesora ili u međuspremnik okvira (frame buffer).

Procedure grafičkog sustava koje se odnose na generiranje primitiva specificiraju što treba generirati:

* **Atributske** procedure specificiraju kako će se generirati primitivi
* **copyPixel** procedura specificira kako će se slike modificirati
* procedure **upravljanja canvasom** specificiraju gdje će se generirati slike.

**Canvasi** su bitmape ili pixmape koje se pohranjuju u prikaznu memoriju. **Ekran** je također canvas koji je poseban samo po tome što je jedini canvas koji se prikazuje. Da bi se slika pohranjena u canvasu prikazala (a ne radi se o ekranskom canvasu), potrebno je da aplikacija taj canvas kopira u ekranski canvas. U nekom trenutku, samo jedan canvas može biti aktivan. To je canvas u koji se crtaju novi primitivi i na kojem se primjenjuju promjene atributa.

Grafički sustav (skup izlaznih podprograma koji sačinjavaju grafičku biblioteku) mora komunicirati s potencijalno velikim brojem različitih prikaznih sklopova. U svim arhitekturama prikaznih sustava centralni procesor mora biti u stanju čitati i zapisivati pojedinačne piksele u međuspremnik okvira. Također, poželjno je da može premještati blokove piksela u i iz međuspremnika okvira kako bi mogao primijeniti copyPixel (Izbornici, Windows, scroll,...).



**Prikazni sustavi s međuspremnicima okvira i prikaznim procesorima**

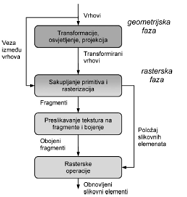
Rasterećenje:

* prikazni procesor obavlja rasterizaciju (scan conversion)
* prikazni procesor izravno obrađuje primitive i atribute koje je proslijedio grafički sustav.

Grafički sustav samo vrši konverziju u njegovu internu reprezentaciju primitiva, atributa i načina zapisa u formatima u koje prihvaća prikazna periferija koja iscrtava primitive.

**Prikazni sustavi bez prikaznog procesora**

Grafički sustav sam vrši rasterizaciju u međuspremnik okvira i neekranski canvas. Dijelovi memorije pripadaju međuspremniku okvira i pohranjuju canvase kojima upravlja grafički sustav. Ostatak memorije koriste ostali programi uključujući i sam grafički sustav.

**GRAFIČKI PROTOČNI SUSTAV (CJEVOVOD)**

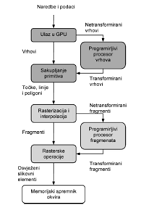
Sastoji se od niza stupnjeva koji obavljaju određene grafičke operacije po fiksnom redoslijedu. Na slici je prikazan vrlo pojednostavljeni grafički protočni sustav. Ulaz u grafički cjevovod čine koordinate vrhova i ostali parametri koji se postavljaju u grafičkoj aplikaciji. Svi ti parametri transformiraju se nizom matrica radi pravilnog smještanja u scenu, proračuna osvjetljenja i projekcije scene. To je “geometrijska faza”.

“Rasterska faza”:

1. Podfaza sakupljanja primitiva (iz pojedinih vrhova slažu se geometrijski elementi) i rasterizacija (određivanje prekrivenosti geometrijskih elemenata slikovnim elementima – pikselima)
2. Podfaza preslikavanja tekstura na fragmente (potencijalne slikovne elemente)
3. Podfaza rasterskih operacija koje uključuju razne testove koje prolazi fragment da bi se utvrdilo da li će se zapisati u memoriju.

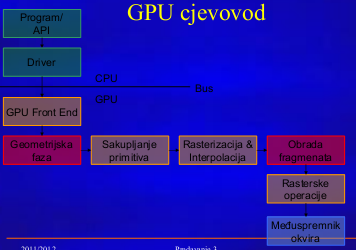
Ako se fragment zapiše u memoriju, točnije memorijski premnik okvira, on će postati likovni element koji se prikazuje na zaslonu. Memorijski spremnik okvira se sastoji od nekoliko spremnika (spremnik boje, dubine, maske, akumulacijski, itd.).

**GRAFIČKI CJEVOVOD PROGRAMIRLJIVOG SKLOPOVLJA**

****

Jedini dodaci su procesor vrhova i procesor fragmenata. Procesor vrhova zamjenjuje transformacije vrhova izvedene sklopovski ostvarenim funkcijama (ne može se koristiti oboje istovremeno). Sve transformacije se moraju izvesti na procesoru vrhova ili sklopovski ostvarenim funkcijama. Analogno vrijedi i za procesor fragmenata. Procesori vrhova i fragmenata mogu raditi nezavisno, odnosno, ako je jedan uključen, drugi ne mora biti. Dio grafičkih operacija i dalje isključivo izvodi pomoću sklopovski ostvarenih funkcija (dio sakupljanja primitiva i rasterizacije te interpolacije, kao i dio rasterskih operacija).

**GPU CJEVOVOD**

****

* Program - Napisani program
* API - Sučelje za OpenGL ili DirectX
* GPU Front End - Prima naredbe i podatke od drivera
* Obrada vrhova (geometrijska faza)
  + Uobičajeno: transformacije
  + Programabilno
* Ispitivanje dubine
  + Provjeri međuspremnik okvira: postoje li već manje dubine (Z-Buffer)
  + Ograničena programabilnost
* Miješanje (Blending)
  + Koristi alfa kanal za kombiniranje boja koje se već nalaze u međuspremniku okvira
  + Ograničena programabilnost

**OSNOVNI ALGORITMI RASTERSKE GRAFIKE**

**Rasterski prikaz**

Osnovni problem je crtanje kose ravne crte širine jednog piksela na dvorazinskom rasterskom prikaznom uređaju.

**Rasterski prikaz ravnih crta**

Algoritmi za rasterski prikaz ravnih crta proračunavaju koordinate piksela na 2D rasteru koji su najbliži idealnoj, beskonačno uskoj ravnoj crti.

Poželjne značajke prikazane ravne crte uključuju:

* jednolik intenzitet crte čitavom dužinom,
* jednolik intenzitet crte neovisno o nagibu,
* mogućnost proizvoljnog odabira širine crte,
* mogućnost proizvoljnog oblikovanja krajeva.

*Osnovni inkrementalni algoritam*

Osnovni inkrementalni algoritam je najjednostavniji pristup. Proračun piksel po piksel na osnovi eksplicitnog oblika jednadžbe pravca.

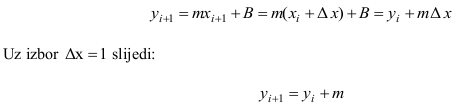
Algoritam:

1. proračun nagiba pravca (|m| < 1)
2. proračun nove vrijednosti xi povećanjem prethodne vrijednosti za 1
3. proračun vrijednosti
4. isticanje piksela (xi, *Round(yi)*) gdje je *Round(yi)* = *Floor (0.5 + yi)*.

Odabire točke koje su najmanje udaljene od idealne crte. Nije učinkovit jer se u svakoj iteraciji obavljaju operacije množenja i zbrajanja s realnim ili (racionalnim) brojevima kao i poziv funkcije *Floor*.

*Digitalni diferencijalni analizator (DDA)*

On izbjegava množenje realnih brojeva. Izraz za proračun vrijednosti koordinate y :



Ako je |m| > 1 mijenjaju se uloge x i y koordinata u algoritmu.

Nedostatak:

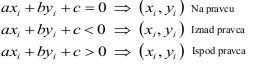
* zbog nepreciznosti prikaza realnih brojeva u digitalnim procesorima dodavanje ne sasvim točne vrijednosti unosi kumulativnu pogrešku i otklon od stvarne vrijednosti

Za kratke crte razina pogreške je prihvatljiva.

*Algoritam središnje točke*

Pretpostavka: nagib pravca između 0 i 1. Za pravce s nagibima izvan ovog intervala primjenjuje se tehnika zrcaljenja oko koordinatnih osi. Pravac se preslikava u odgovarajući pravac s nagibom unutar navedenog intervala. Samo inkrementiranje i cjelobrojno procesiranje.

Koristimo implicitni oblik pravca:



U svakoj iteraciji utvrdimo da li je sred.točka iznad ili ispod pravca. Ispitaj vrijednost od:



di - varijabla odluke u i – tom koraku

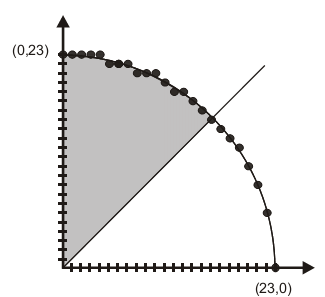
**Rasterski prikaz kružnice**

Osnovna jednadžba kružnice sa ishodištem koordinatnog sustava smještenom u njenom središtu je:



Eksplicitno izrazimo y:



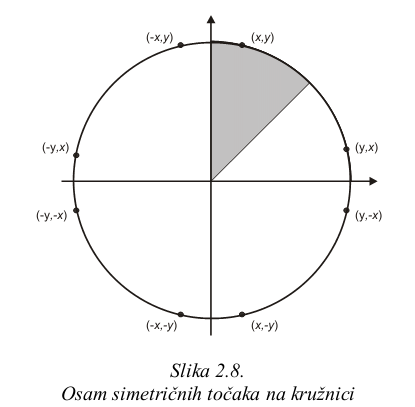
1. Nacrtamo četvrtinu kružnice
2. Povećavanjem vrijednosti x-a od 0 do r u jediničnim koracima izračunavamo +y rješenje za svaki korak.

Nedostatci:

* računski zahtjevna uzastupna izračunavanja kvadratnog korijena
* kako vrijednost x-a postaje bliža krajnjoj vrijednosti r, točke na kružnici će biti sve razmaknutije

Drugi pristup:

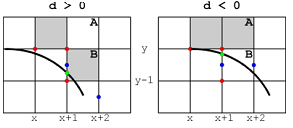
* crtanjem točaka s koordinatama (rcosα, rsinα) povećavanjem vrijednosti kuta od 0° do 90°
* Ovaj način je također spor

*Osmosmjerna simetrija*

Poboljšanje rasterskog prikazivanja kružnice može se postići većim korištenjem simetrije ovog primitiva. Ukoliko znamo da točka (x, y) leži na kružnici, možemo na trivijalan način izračunati ostalih sedam točaka kružnice kako je prikazano na slici 2.8. Dakle, za potpuno određivanje svih potrebnih točaka kružnice potrebno je izračunati samo točke koje leže na jednom luku kružnice s vršnim kutem od 45°.

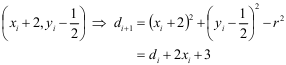
*Algoritam središnje točke za kružnice*

Koristimo simetriju pa razmatramo samo segment od 45°, npr. drugi oktant. Kao i kod tehnike središnje točke za rasterski prikaz ravne crte, strategija se zasniva na odabiru koji je od dva piksela bliži kružnici tako što se procijeni funkcija na sredini između ta dva piksela.

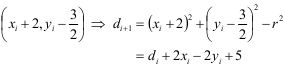


–Ako je di  < 0 izabiremo piksel A, inače B

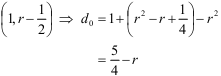
Ponovo, na osnovu izbora točke A ili B se može koristiti za izračun nove vrijednosti di+1. Ako je izabran A onda slijedeća središnja točka ima vrijednost varijable odluke:



Inače, ako je bio izabran B varijabla odluke se izračunava iz:



Ako pretpostavimo da je radijus cjelobrojna vrijednost, prvi nacrtani piksel će biti (0, r) a početna vrijednost za varijablu odluke je:



**ISPUNJAVANJE PRIMITIVNIH OBLIKA**

**ODREZIVANJE**

**ISPUNJAVANJE PRIMITIVNIH OBLIKA**

Primitivni oblici:

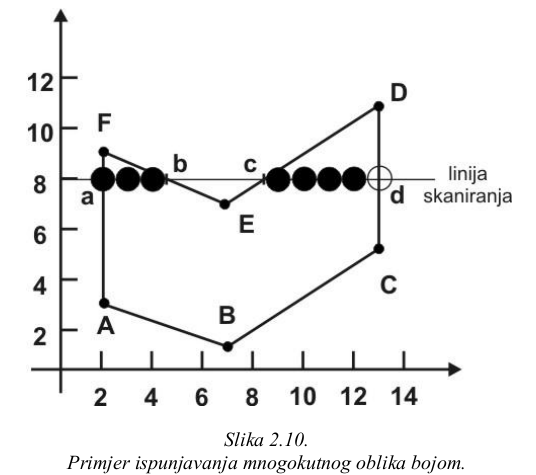
* Pravokutnici
* Kružnice
* Elipse
* Mnogokuti

Osnovni koraci:

* Izbor piksela koje treba ispuniti (ovisi o obliku primitiva)
* Izbor vrijednosti koju treba pridijeliti odabranim pikselima (ovisi o načinu ispunjavanja: bojom ili uzorkom)

*Izbor piksela za ispunjavanje:*

1. skeniranje po horizontalnim crtama (retcima piksela),
2. proračun presjecišta horizontalne crte i primitivnog oblika inkrementalnim algoritmom (odabiru se točke koje leže unutar oblika),
3. poredavanje presjecišta po veličini x koordinate,
4. ispunjavanje odsječaka koji leže unutar primitivnog (između neparnih i parnih presjecišta).



Primjer ispunjavanja mnogokutnog oblika bojom:

a, d – cjelobrojni

b, c - necjelobrojni

Zbog mogućih različitih pristupa, četvrti korak naveden u izboru piksela nekog primitiva potrebno je proširiti i pojasniti. Pitanja na koja je potrebno odgovoriti su:

4.1. Ako je zadan jedan presjek sa proizvoljnom vrijednošću x-a, kako određujemo s koje strane presjeka je unutrašnji piksel?

4.2. Kako se ponašamo u slučaju presjeka na cjelobrojnim vrijednostima koordinata piksela?

4.3. Što radimo sa specijalnim slučajem 4.2 za zajedničke vrhove?

4.4. Što radimo sa specijalnim slučajem 4.2 kada vrhovi određuju horizontalni rub?

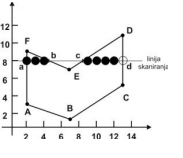
*4.1. Ako je zadan jedan presjek sa proizvoljnom vrijednošću x-a, kako određujemo s koje strane presjeka je unutrašnji piksel?*

Rješenje:

Ako se približavamo necjelobrojnom presjeku koji je s desne strane i nalazimo se unutar poligona, x koordinatu presjeka ćemo zaokružiti na manju vrijednost. Ako se nalazimo izvan poligona, zaokružit ćemo x koordinatu presjeka na veću vrijednost kako bi odabrani piksel bio unutar poligona.

*4.2. Kako se ponašamo u slučaju presjeka na cjelobrojnim vrijednostima koordinata piksela?*

Rješenje:

Primjenjujemo kriterij koji se koristi kod izbjegavanja konflikata zajedničkih rubova pravokutnika: Ako piksel koji se nalazi najviše lijevo u odsječku ima cjelobrojnu x koordinatu, definiramo ga kao unutrašnjeg; Ako piksel koji se u odsječku nalazi najviše desno ima cjelobrojnu x koordinatu, definiramo ga kao vanjskog.

*4.3. Što radimo sa specijalnim slučajem 4.2 za zajedničke vrhove?*

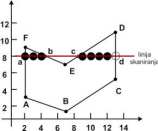
Rješenje:

Ubrajamo ymin vrh ruba pri računanju parnosti, ali ne i vrh ymax. Stoga, ymax vrh iscrtavamo samo ako je to ymin vrh susjednog ruba.

Primjer: vrh A ubraja se jedan put u računanju parnosti jer je to ymin vrh ruba FA ali ymax vrh ruba AB.

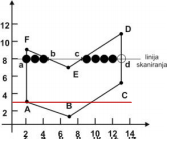
*4.4. Što radimo sa specijalnim slučajem 4.2 kada vrhovi određuju horizontalni rub?*

Rješenje:

Za horizontalne rubove, željeni efekt je da, kao i kod pravokutnika, donji rubovi budu nacrtani, a gornji ne. Ovo će biti automatski zadovoljeno ako ne ubrajamo vrhove rubova s obzirom da oni nisu ni ymin ni ymax vrhovi.

Primijenimo navedena pravila na liniju skeniranja 8 prikazanu na slici 2.10.:

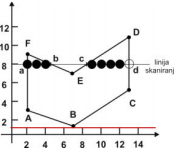
Popunjavamo piksele od točke a, piksela (2,8), do prvog piksela koji se nalazi lijevo od točke b, piksela (4,8) i od prvog piksela desno od točke c, piksela (9,8), do 1 piksela lijevo od točke d, piksela (12,8).



Linija skeniranja 3:

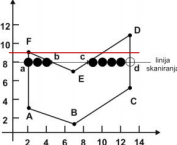
Vrh A ubrajamo jednom jer je to ymin vrh ruba FA ali i ymax vrh ruba AB.

Ovo uzrokuje neparnost pa iscrtavamo odsječak do 1 piksela lijevo od presjeka s rubom CB, gdje brojač parnosti postaje paran i odsječak završava.



Linija skeniranja 1:

Oba ruba AB i BC imaju ymin vrh u točci B, pa se ona stoga dva puta ubraja u brojaču parnosti te daje paran rezultat. Ovaj vrh se ponaša kao nulti odsječak – ulaz u vrh, crtanje piksela, izlaz iz vrha. Lokalni minimum iscrtava jedan piksel (u slučaju lokalnog max. ne iscrtavamo niti jedan piksel).



Linije skeniranja 9:

Vrh F dijele rubovi FA i EF.

Oba vrha su ymax vrhovi i stoga ne utječu na brojač parnosti, pa rezultat ostaje neparan.

U slučaju ispunjavanja primitivnog oblika uzorkom algoritam se upotpunjuje dijelom koji upravlja izborom vrijednosti pojedinog piksela.Vrijednosti pojedinih piksela preuzimaju se određenim redoslijedom iz dijela memorije u kojem je definiran uzorak.

**ODREZIVANJE LINIJA**

Primitivi (linije, trokuti, poligoni) ne leže u cijelosti unutarprozora promatranja (viewport).

Zašto odrezati?

* Rasterizacija dijelova slike koji su izvan vidljive slike je loša ideja
* Gubljenje vremena na rasterizaciju piksela van prozora

**ODREZIVANJE KRAJNJIH TOČAKA**

Ako su x koordinate rubova odreznog pravokutnika xmin i xmax, a y koordinate rubova na ymin i ymax, tada trebaju biti zadovoljene četiri nejednadžbe da bi točka (x,y) bila unutar pravokutnika:



Ukoliko nisu zadovoljene sve četiri nejednakosti, točka je izvan odreznog pravokutnika.

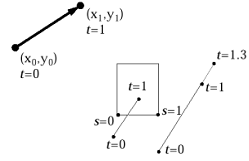
**ODREZIVANJE RAVNIH CRTA RJEŠAVANJEM SIMULTANIH JEDNADŽBI**

* Ako su obe krajnje točke unutra, može se “trivijalno prihvatiti”
* Ako je jedna krajnja točka unutra, a druga vani, odrezivanje
* Ako su obe krajnje točke vani, nije određeno

“Brute force” algoritam: rješavanjem simultanih jednadžbi oblika y = mx + b za ravnu crtu i četiri ruba odrezivanja. Ako postoji bar jedna točka koja leži na tim rubovima, crta siječe odrezni pravokutnik i djelomično se nalazi unutar njega.

Postupak možemo opisati u tri koraka:

1. Za svaku crtu i rub pravokutnika napišemo jednadžbe dvaju pravaca koje ih sadrže i presiječemo ih.
2. Nadalje, ispitujemo da li se točka presjeka nalazi i unutar rubova odreznog pravokutnika i unutar crte.
3. Ako je taj uvjet zadovoljen, postoji presjek crte i odreznog pravokutnika.

Parametarska formulacija segmenta ravne crte glasi:

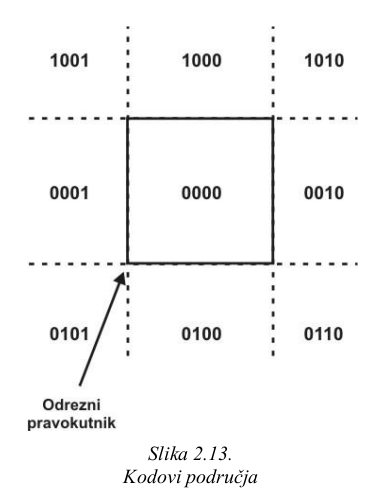
X = x0 + t(x1 – x0) 0 < t < 1

Y = y0 + t(y1 – y0)

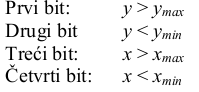
**P**(t) = P0 + t(P1 – P0)

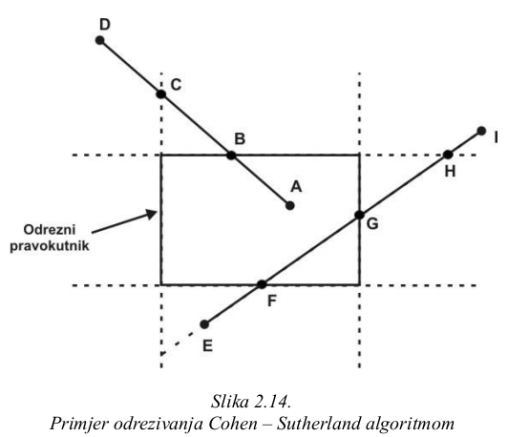
“true,” tj., unutrašnji presjek, ako srub i tcrta iz [0,1]

**COHEN – SUTHERLAND ALGORITAM ODREZIVANJA**



Ovo je vjerojatno najčešće korišteni algoritam za odrezivanje ravnih crta. Nije najučinkovitiji. Postoje brži algoritmi kao npr. Nichollov algoritam ili Cyrus – Beck algoritam.Cohen – Sutherland algoritam provodi početna ispitivanja linije kako bi utvrdio da li je moguće izbjeći proračune presjeka.



1. Zadana je linija s krajnjim točkama P0 = (x0, y0) i P1 = (x1, y1)
2. Izračunaj odgovarajuće 4-bitne kodove za krajnje točke. Ako su oba koda 0000, (logička ILI funkcija po bitovima kodova daje rezultat 0000 ) linija je u potpunosti unutar prozora: proslijedi krajnje točke rutini crtanja. Ako oba koda imaju 1 na istoj poziciji bita (logička I funkcija po bitovima kodova nije 0000), linija se nalazi izvan prozora => Može se trivijalno odbaciti.
3. Ako liniju ne možemo trivijalno odbaciti niti prihvatiti, to znači da bar jedna krajnja točka leži izvan prozora i da segment linije siječe rub prozora.Ova linija mora biti odsječena na rubu prozora prije nego što se proslijedi rutini za crtanje.
4. Ispitaj jednu krajnju točku, na primjer P0 = (x0, y0). Očitaj 4-bitni kod točke P0 sljedećim redom: Lijevo-Desno, Dole-Gore.
5. Kada se pojavi bit postavljanja (1), izračunaj presjek I odgovarajućeg ruba prozora sa linijom od P0 do P1. Zamijeni P0 sa I i ponovi algoritam.

Dužina AD:

Točka A ima kod 0000, a točka D ima kod 1001. Logička I funkcija ova dva koda je nula; stoga se inija ne može trivijalno odbaciti. Također, logička ILI funkcija kodova nije jednaka nuli pa liniju ne možemo trivijalno prihvatiti. Po redu ispitivanja, prvo koristimo gornji rub za odrezivanje AD na B. Algoritam potom ponovo izračunava kod točke B kao 0000. U sljedećoj iteraciji algoritma, AB se ispituje i trivijalno prihvaća i prikazuje.

Dužina EI:

Točka E ima kod 0100, a točka I kod 1010. Linija se ne može trivijalno prihvatiti niti odbaciti. E - vanjska točka => algoritam odrezuje liniju na donjem rubu prozora. Sada je linija EI odsječena na dužinu FI. Dužina FI se ispituje i ne može se trivijalno prihvatiti niti odbaciti. Točka F ima kod 0000 => algoritam izabire točku I kao vanjsku točku s obzirom da je njen kod 1010. Dužina FI se odsijeca do gornjeg ruba prozora, što rezultira novom dužinom FH. Dužina FH se ne može trivijalno prihvatiti niti odbaciti. Kod točke H = 0010 => sljedeća iteracija algoritma odsijeca do desnog ruba prozora, i dobijemo dužinu FG. Sljedeća iteracija algoritma ispituje FG, i ona se trivijalno prihvaća i prikazuje.

**ODREZIVANJE POLIGONA**

Odrezivanje poligona je složenije nego odrezivanje pojedinačnih linija

* Ulaz: poligon
* Izlaz: originalni poligon, novi poligon, ili ništa

Zašto je odrezivanje teško?

Mogući rezultati:



Slučaj težak za rješavanje:





**ALGORITAM SUTHERLAND-HODGEMAN**

* Podijeli-pa-vladaj strategija.
* Rješava niz jednostavnih i identičnih problema koji združeni rješavaju cijeli problem.
* Jednostavni problem je odrezivanje poligona s jednim beskonačnim rubom odrezivanja.
* Četiri ruba odrezivanja, od kojih svaki definira jednu granicu pravokutnika odrezivanja (uspješno odrezuju poligon s pravokutnikom odrezivanja.

Osnovna ideja: odrezivanje poligona s jednim beskonačnim rubom odrezivanja.



Osnovna ideja: odrezivanje poligona s jednim jednadžbom ruba. Nakon odrezivanja svih ravnina – poligon je potpuno obrađen.





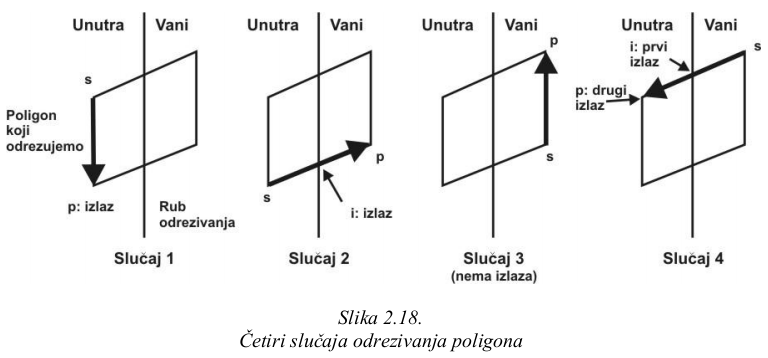
 ...

Ulaz/izlaz algoritma:

* Ulaz: lista poredanih vrhova poligona
* Izlaz: lista vrhova odrezanog poligona koja se sastoji od starih vrhova (možda) i novih vrhova (možda)

Osnovna rutina Sutherland-Hodgman:

* Idi oko poligona (od vrha do vrha)
* Tekući vrh ima poziciju p
* Prethodni vrh je imao poziciju s, i dodan je izlaznoj listi ako je to bilo potrebno



Četiri slučaja:

1. Rub poligona je u potpunosti unutar granica odrezivanja, vrh p se dodaje izlaznoj listi (p unutra, s unutra)
2. Točka presjeka i postaje izlaz kao vrh jer rub presijeca granicu. (s unutra, p vani)
3. Oba vrha su izvan granica pa nema izlaza. (s vani, p vani)
4. Točke presjeka i i p su obje dodane izlaznoj listi. (s vani, p unutra)

**GRAFIČKO SKLOPOVLJE**

**TEHNOLOGIJE ISPISA (HARDCOPY)**

To su sve tehnologije koje omogućavaju zapis statične digitalne slike na medij koji nije elektronski (pisači, ploteri,…).

Nemoguće povući jasnu liniju razgraničenja između pisača i plotera:

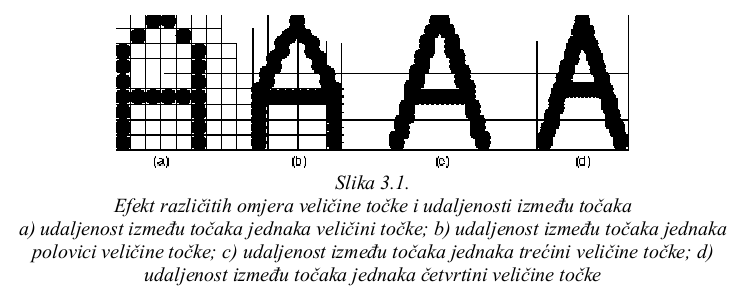
* pisači imaju mogućnost ravnopravnog tiskanja teksta i visokokvalitetnih grafičkih sadržaja
* ploteri bazirani na ink-jet ili elektrostatskoj tehnologiji bez problema mogu raditi kao klasični pisači
* dio hardcopy tehnologija, na potpuno isti način, koristi se i u pisačima i u ploterima (ink-jet tehnologija)

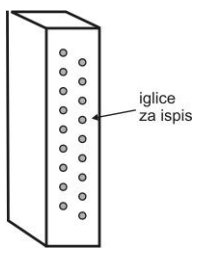
Ako baš moramo:

* Pisači su uređaji namijenjeni tiskanju teksta i grafičkih sadržaja koji uglavnom koriste medije (papir) do formata A2
* Ploteri su uređaji namijenjeni prvenstveno ispisu grafičkih sadržaja, a proizvode se i za veće formate papira (A1, A0 i više).

Ključne veličine kojima se opisuju uređaji za ispis su:

* Adresabilnost: broj točaka (piksela) koje uređaj može proizvesti po jedinici dužine, pri čemu nije obavezno da točke budu razdvojene jedna od druge.
* Udaljenost među točkama recipročna je adresabilnosti.
* Veličina točke: izražava se promjerom točke koju proizvodi uređaj.
* Rezolucija: maksimalan broj linija, naizmjenično crnih i bijelih, koje uređaj može proizvesti po jedinici dužine





**MATRIČNI PISAČI**

Jedna od prvih jeftinih tehnologija ispisa. Mehanizam za uvlačenje papira koji se sastoji od valjka i pomoćnih kotačića transportira papir kroz pisač dok glava za tiskanje ispisuje sadržaj na papir krećući se lijevo-desno. Glava sadrži od 7 do 24 iglice pokretane minijaturnim elektromagnetima koje udarom u papir preko pigmentirane trake (ribon) ostavljaju trag. Izgled glave za tiskanje matričnog pisača sa iglicama poredanim u dva stupca (slika). Niska kvaliteta tiskanja grafičkih sadržaja kao što su fotografije i crteži. Osnovni razlog je veličina točke. Moderni matrični pisači postižu adresabilnost do 360 dpi i po horizontali i po vertikali.

Zbog veličine točke, međutim, rezolucija se kreće od 50 do 100 dpi. Za postizanje tonskih prijelaza matrični pisači koriste uglavnom dithering tehnike S ovako niskom rezolucijom nije moguće postići zadovoljavajuće tonske prelaze.

Značajan problem: zagrijavanje glave za tiskanje:

Velika disipacija snage u elektromagnetima korištenim za pokretanje iglica -> glava se jako zagrijava (broj iglica i brzina tiskanja) -> Pregrijavanje glave -> promjena električkih osobina elektromagneta -> smanjenje preciznosti tiskanja.

Može doći i do pregorijevanja elektromagneta tj. do uništenja glave za tiskanje.

Ova tehnologija je ujedno i najbučnija. Matrični pisači su se danas zadržali samo tamo gdje je potrebno tiskati original i više kopija odjednom. Upotrebljavaju se i kao tzv. linijski pisači koji omogućavaju ekstremno brzi ispis velikih količina teksta. Linijski pisači primarno ispunjavaju zahtjev za brzinom a ne kvalitetom tiska, koriste glave sa 9 iglica.

**INK-JET PISAČI**

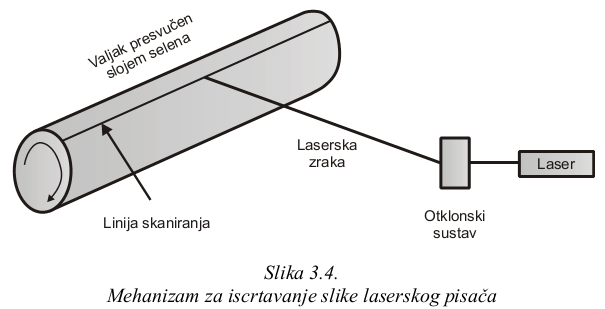
Nasljednik tehnologije matričnih pisača. Razlika u odnosu na matrične pisače je samo u konstrukciji glave za tiskanje. Umjesto iglica imamo cjevčice (mlaznice) kroz koje se sitne kapljice tinte izbacuju direktno na papir. Nakon izbacivanja kapljice tinte, cjevčica se ponovo puni iz rezervoara sa tintom.

Postoje dvije varijante ove tehnologije koje se razlikuju po načinu na koji ispumpavaju tintu na papir:

* "**Bubble-jet**“: koristi minijaturne grijače koji zagrijavanjem tinte u mlaznici stvaraju balončić (bubble). Prskanjem ovog balončića na papir se izbacuje kapljica tinte.
* **Piezoelektrični ink-jet** pisači koriste pločicu materijala (kvarc) koji mijenja svoje dimenzije kada se nađe u električnom polju. Dovođenjem napona na pločicu -> ona se širi tako da smanjuje prostor u mlaznici -> kapljica tinte izbacuje se na papir

Ne proizvode značajan višak toplote. Mlaznice ne trpe velika fizička naprezanja pa mogu biti i nekoliko desetina puta tanje od iglica u glavi matričnih pisača. Visoka adresabilnost (standardno 600 do 720 dpi, a bolji pisači i više od 1000 dpi). Manja veličina točke uz veću brzinu tiskanja od matričnih pisača (ne računajući linijske pisače). Rezultat: višestruko veća stvarna rezolucija u odnosu na matrične pisače.

Ugradnjom više stupaca mlaznica u glavu za tiskanje za više osnovnih boja dobiva se pisač u boji. Postoje modeli koji koriste samo tri osnovne boje (CMY) i modeli koji uz ove boje koriste i crnu boju (CMYK). Za postizanje tonskih i prelaza u boji, ovi pisači koriste dithering. Ink-jet pisači su i dalje relativno spori.

**LASERSKI PISAČI**

* prihvatljiva cijena
* velika brzina
* nisku cijenu po tiskanoj kopiji
* nedostižna kvalitetu tiska teksta i crno-bijele grafike

Tehnologija se zasniva na osobini nekih materijala (selen) da mijenjaju stanje vodljivosti pod utjecajem svjetlosti**.**

Selen je poluvodič koji kada nije osvijetljen ima osobine izolatora, a kada se osvijetli postaje vodič električne struje. U svakom laserskom pisaču nalazi se metalni valjak presvučen slojem selena, sa elektromotorom koji ga vrti oko osi. Prolaskom ispod elektrostatskog češlja na visokom pozitivnom potencijalu, površina selena se pozitivno nabije. Sve dok je selen neosvijetljen, ovaj naboj ostaje na površini valjka.

Drugi bitan dio laserskog pisača: laser sa otklonskim sustavom. Kada laserska zraka pogodi neku točku na površini valjka, selen u toj točki postaje vodljiv i pozitivni naboj sa površine se uzemlji preko metalnog jezgra valjka. Na površini valjka tako dobijemo nenabijenu tj. nenaelektriziranu točku. Tako je moguće na selenskom valjku iscrtati sliku (raster) koja se sastoji od naelektriziranih i nenaelektriziranih točaka.

Otklonski sustav (ogledalce pokretano elektromotorom) "vozi" laserska zraka lijevo-desno, od kraja do kraja valjka. Kada zraka dođe do točke koja treba ostati nabijena laser se gasi, a pali se ponovo kada otklonski sustav dođe do točke koju treba osvijetliti. Kada se na ovaj način iscrta cijeli red točaka, valjak se okreće za jedan red točaka i iscrtava se sljedeći red. Površina valjka dalje prolazi kroz posudu sa negativno nabijenim obojenim mikronskim prahom (tonerom).

Pozitivno nabijene točke na valjku privući će toner, dok će nenabijene točke ostati čiste. Na ovaj način na valjku dobijemo sliku obojenu u točkama koje nisu osvijetljene laserom. Toner se sa valjka, blagim pritiskom, prenosi na papir koji se kreće istom brzinom kao i površina valjka. Konačni rezultat - monokromatska rasterska slika na papiru. Da bi se toner fiksirao, papir sa slikom prolazi ispod grijača koji na trenutak rastopi toner. Rastopljene čestice tonera vežu se međusobno u kompaktnu masu koja se hlađenjem lijepi za površinu papira.

Ova tehnologija zahtjeva da se proces ispisa ne prekida, što znači da cijeli raster koji treba ispisati mora biti na raspolaganju pisaču prije nego što ispis počne. Podaci: niz nula i jedinica koje upravljaju paljenjem i gašenjem lasera, i moraju se odgovarajućom brzinom dostavljati mehanizmu koji upravlja laserom. Obim ovih podataka povećava se sa povećanjem adresabilnosti pisača.

Povećanje adresabilnosti i brzine ispisa zahtjeva brže isporučivanje podataka mehanizmu za upravljanje laserom. U ovu vrstu pisača ugrađuje se brza memorija (međuspremnik - buffer) čiji kapacitet mora biti dovoljno veliki da u nju stane cijela bit-mapa koja se tiska. Ako međuspremnik nije dovoljno veliki, pisač nije u stanju tiskati cijelu sliku. Povećanje formata, adresabilnosti i brzine ispisa -> povećanje kapaciteta i brzine međuspremnika -> skuplji pisači.

Većina proizvođača ovaj problem djelimično rješava upotrebom tehnika OnLine kompresije podataka, što u većini slučajeva dozvoljava korištenje međuspremnika manjeg kapaciteta.

*Laserski pisači u boji*

Izvedbe:

1. Sa tri ili četiri mehanizma za tiskanje, za tri (CMY) odnosno četiri (CMYK) osnovne boje
2. Pisači kod kojih se papir provlači kroz isti mehanizam tri ili četiri puta, za svaku osnovnu boju po jednom, uz korištenje tonera odgovarajuće boje.

Prva izvedba je brža i skuplja. Tonski i kolor prelazi se dobijaju screening-om ili error diffusion dithering-om.

Dva faktora koja određuju veličinu točke laserskih pisača:

* debljina laserske zrake (nije ograničavajući faktor)
* kvaliteta tonera.

1. Mala naelektrizirana točka na površini selena nije u stanju privući suviše velika zrnca tonera, što dovodi do gubljenja pojedinačnih piksela na završnom otisku.
2. Ako su zrnca tonera prekrupna, naelektrizirana površina selena ne privlači dovoljnu količinu tonera, što se manifestira nedovoljnim "zacrnjenjem" obojenih površina na papiru

Cijena tonera je uglavnom niža od cijene tinte korištene u ink-jet pisačima za istu količinu tiskanog materijala. Zbog potrebe za međuspremnikom za smještaj cijele slike te zbog problema vezanih za fokusiranje i precizno vođenje laserske zrake preko jako širokih selenskih bubnjeva, laserski pisači se obično ne izrađuju za tiskanja na formate veće od A3. Relativno velika potrošnja električne energije i zahtjev za velikom snagom izvora električne energije (1-2 kW), što je uglavnom posljedica rada grijača za "pečenje" tonera na papiru.

Laserski pisači danas standardno postižu brzinu ispisa od desetak strana u minuti (ppm). Bolji modeli i više od 40 ppm. Rezolucija se kod boljih modela približava adresabilnosti i kreće se od standardnih 600 ili 1200 dpi do više od 2000 dpi.

**LED PISAČI**

LED pisači koriste istu tehnologiju kao laserski, osim što umjesto lasera koriste niz svjetlećih dioda (LED) postavljen duž selenskog valjka. Ove diode se mogu pojedinačno paliti i gasiti. Svjetlost iz niza dioda fokusira se na selenski valjak na kojem iscrtava cijeli red piksela odjednom (izbijanjem naboja kao i kod laserskih pisača).

Horizontalna rezolucija zavisi od gustoće niza svjetlećih dioda. Ova tehnologija se počela razvijati nešto nakon laserske. Postignuta rezolucija, brzina i kvaliteta ispisa su nešto niži nego kod laserskih pisača. Sa druge strane, ovi su pisači jeftiniji i robusniji od laserskih, a sama tehnologija je pogodnija za razvoj pisača većeg formata. Izrađuju se u crno-bijelim i kolor varijantama.

**ELEKTROSTATSKI PISAČI I PLOTERI**

1. Gusti češalj napravljen od vrlo tankih metalnih iglica ispod kojeg se konstantnom brzinom provlači papir na koji se tiska
2. Promjene potencijala na iglicama češlja, red po red, oslikavaju raster na papiru, sastavljen od negativno naelektriziranih i nenaelektriziranih točaka.
3. Papir zatim prolazi kroz posudu sa pozitivno nabijenim tonerom, čije čestice se lijepe za negativno nabijene točke na papiru.
4. Toner zagrijavanjem fiksira za papir.

* Velika brzina i kvaliteta ispisa
* Uglavnom se koristi u ploterima za ispise velikih formata (papir širine 2 metra i više)
* Za postizanje tonskih prijelaza koriste se tehnike ditheringa

**PLOTERI S PERIMA**

Jedna od najstarijih tehnologija korištenih za kvalitetan ispis vektorke grafike, prije svega tehničkih crteža. I danas nezamjenjivi u arhitektonskim i konstruktorskim biroima i svuda gdje postoji potreba za tehničkim crtežima velikog formata, visoke preciznosti i vrhunske kvalitete.

Iscrtavaju likove povlačeći pero sa tintom ili tušem po papiru. Glava koja drži pero montira se na posebnu šinu (kran) duž koje se može kretati iznad papira. Ovo daje glavi za iscrtavanje jedan stupanj slobode (mogućnost kretanja duž jedne osi iscrtavanja). Drugi stupanj slobode (kretanje glave duž druge ose iscrtavanja) osigurava se na različite načine.

Jedna od često korištenih izvedbi je sa kranom koji se kreće iznad plohe na kojoj stoji papir ("flatbed" ploteri). Druga izvedba koristi se u tzv. "drum" tj. doboš ploterima i sastoji se od fiksno postavljenog krana sa glavom za iscrtavanje i okretnog bubnja preko kojeg se zateže papir. Bubanj povlači papir ispod glave za iscrtavanje. Obje ove izvedbe koriste se za plotere velikih formata.

Proces iscrtavanja sastoji se od sljedećih postupaka:

1. glava sa perom se dovodi iznad mjesta na papiru na kojem je potrebno započeti iscrtavanje lika
2. glava spušta pero na papir
3. mehanizmi pokreću glavu i kran ili papir prema parametarskim ili vremenskim funkcijama koje opisuju iscrtavani lik
4. nakon iscrtavanja cijelog lika glava podiže pero sa papira i proces se ponavlja za sljedeći lik (ako je lik previše složen, iscrtava se kao više jednostavnijih)

Današnji ploteri sa perima standardno mogu koristiti 16 različitih pera istovremeno, a bolji modeli i više. Pripadaju grupi random-scan uređaja. Neprimjenjivi su za iscrtavanje bilo čega drugog osim čisto vektorske grafike. Proizvode se uglavnom za iscrtavanje na medije velikih formata (A1, A0 i više).

Usko polje primjene rezultira malim serijama u proizvodnji a time i vrlo visokim cijenama ovih uređaja. Sporiji su od elektrostatskih i većine ink-jet plotera, ali to nadoknađuju nedostižnom kvalitetom iscrtavanja.

**GRAFIČKO SKLOPOVLJE II**

**TEHNOLOGIJE ZASLONA**

Najzastupljenije tehnologije zaslona:

* Zasloni s katodnom cijevi (*CRT - Cathode Ray Tube*)
* Zasloni s tekućim kristalom (*LCD - Liquid Crystal Display*)
* Plazma zasloni (*PDP – Plasma Display Panel*)

Ostale prikazne tehnologije su ili značajno manje zastupljene u upotrebi ili su u eksperimentalnoj fazi (npr. OLED).

**SUSTAVI SA KATODNOM CIJEVI (CRT)**

Ista tehnologija kao i kod televizijskih prijemnika, osciloskopa. Princip rada CRT prikaznog sustava zasniva se na osobini nekih materijala da emitiraju svjetlost kada ih pogodi snop elektrona. Intenzitet emitirane svjetlosti zavisi osim od karakteristika upotrijebljenog materijala i od broja i brzine elektrona koji pogađaju taj materijal.

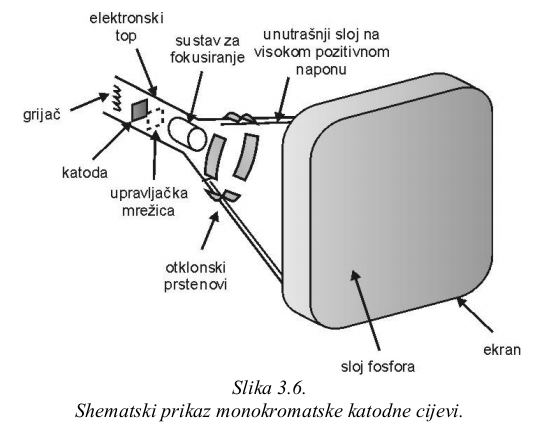
1. Kada elektron pogodi atom nekog materijala, predaje mu dio svoje energije
2. Višak energije akumulira se u atomu tako što elektroni iz elektronskih ljuski prelaze u viša energetska stanja tj. u elektronske ljuske sa većom energijom
3. Ovakvo stanje elektrona je nestabilno i on teži da se vrati na niži energetski nivo, pri čem u se oslobađa kvant energije, foton
4. Pojava emitiranja svjetlosti koja nastaje kao posljedica vraćanja jako nestabilnih elektrona u svoje ljuske naziva se flourescencija, za razliku od fosforescencije koju izaziva vraćanje nešto stabilnijih elektrona na niže energetske nivoe

* **Fluorescencija**: javlja se jako brzo nakon pobude atoma i traje vrlo kratko
* **Fosforescencija**: malo kasni i traje mnogo duže od fluorescencije

Najčešće se ove dvije pojave javljaju zajedno.

* **Perzistencija materijala**: vrijeme koje protekne od prestanka pobude do pada intenziteta emitirane svjetlosti na 10% početne tj. maksimalne vrijednosti.

Materijal koji se koristi za emitiranje svjetlosti kod katodnih cijevi je baziran na fosforu.Većina emitirane svjetlosti kod fosfora je posljedica fosforescencije, dok je fluorescencija gotovo zanemariva.



Monokromatska katodna cijev sastoji se od:

* elektronskog topa
* sustava za fokusiranje
* otklonskog sustava
* sustava za ubrzavanje elektrona
* zaslona ili ekrana

Elektronski top sastoji se od:

* katode
* grijača koji zagrijava katodu
* upravljačke mrežice

Iscrtavanje slike na ekranu: otklonski sustav prevlači elektronsku zraku s lijeva na desno, red po red, odozgo prema dole. Naponom na upravljačkoj mrežici regulira se intenzitet snopa elektrona, čime se na odgovarajućim točkama ekrana dobije jače ili slabije osvjetljenje. Kako otklonski sustav pomiče elektronsku zraku, na ekranu se iscrtava slika sastavljena od svjetlećih i tamnih točaka. Elektronska zraka pogađa svaku točku ekrana samo jedan kratki trenutak, nakon čega je otklonski sustav pomiče na sljedeću točku.

Ako želimo zadržati sliku na ekranu - treba osvježiti ekran. **Kritična frekvencija fuzije** je frekvencija osvježavanja pri kojoj ljudsko oko prestaje uočavati treperenje KFF ovisi o:

1. perzistenciji
2. ambijentalnom osvjetljenju
3. intenzitetu i boji emitirane svjetlosti
4. varira od čovjeka do čovjeka

Povećavanjem perzistencije fosfornog materijala koji se koristi za prekrivanje ekrana katodne cijevi dobili bismo mirniju sliku čak i kod nižih frekvencija osvježavanja. Dobro za statične slike, ne za dinamične. U modernim računalnim zaslonima, stabilna, netreperava slika se radije dobije povećanjem frekvencije osvježavanja.

Razlikujemo dvije frekvencije osvježavanja:

* horizontalnu
* vertikalnu

**Horizontalna** frekvencija osvježavanja je broj redova ekrana koji katodna cijev iscrta u jedinici vremena. **Vertikalna** frekvencija osvježavanja je frekvencija osvježavanja cijelog ekrana. To je broj koji pokazuje koliko puta u sekundi katodna cijev obnovi cijelu sliku na ekranu. **Bandwidth** ili pojasna širina je maksimalna frekvencija paljenja i gašenja elektronskog topa katodne cijevi.Katodna cijev je uređaj koji na svom ekranu prikazuje rastersku sliku.

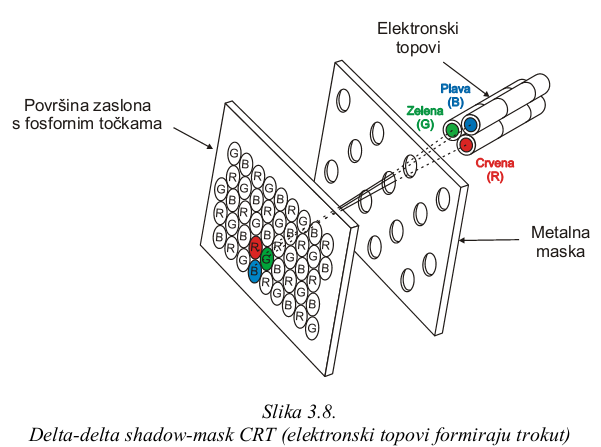
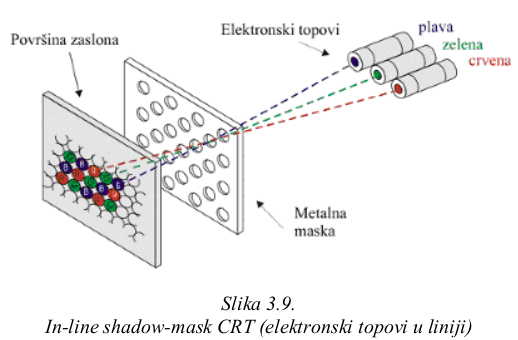
Rasterski uređaj karakteriziraju:

* adresabilnost
* rezolucija
* veličina piksela
* veličina ekrana

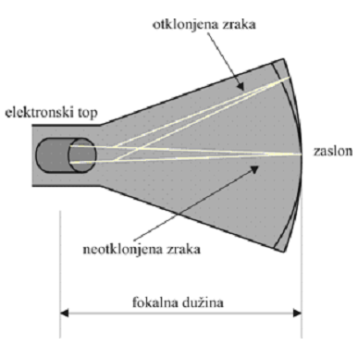
**KATODNE CIJEVI U BOJI**

**Katodna cijev** je uređaj koji emitira svjetlost. Za prikaz boja izabran je RGB model boja. Na ekranu katodne cijevi moraju postojati fosforne točke koje emitiraju tri osnovne boje iz RGB modela (crvenu, zelenu i plavu). Miješanjem intenziteta svjetlosti koju emitiraju te točke dobija se bilo koja boja iz spektra mogućih boja. Točke moraju biti dovoljno sitne i grupirane u trijade.

**Trijada** je grupa sastavljena od crvene, zelene i plave točke koje su postavljene blizu jedna drugoj. Oko trijadu vidi kao jednu točku koja emitira svjetlost jednaku zbiru svjetlosti koju emitira svaka točka pojedinačno.



Kako bi se osiguralo da svaki elektronski top pogađa samo odgovarajuće točke na ekranu (crveni top samo crvene točke, itd.), neposredno prije udarca u zaslon elektroni prolaze kroz metalnu masku (shadow -mask).Uz veličinu točke, uvodi se još jedna bitna veličina, tzv. **dot-pitch** ili rastojanje između točaka. Ova veličina, u stvari, daje minimalno rastojanje između centara susjednih trijada.

Da bi se smanjilo izobličenje prikazane slike, idealno bi bilo da ekran katodne cijev i bude ravan. Da bi se sve točke ekrana nalazile u fokusu elektronskog zraka, ekran mora imati oblik isječka te sfere.

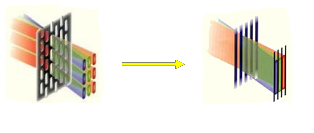
**PROBLEMI**

* Duža katodna cijev - ekran postaje ravniji bez narušavanja fokusa elektronske zrake - glomazno
  + Praktičnije rješenje: mijenjanje žarišne daljine elektronske zrake (kontrolom električnog polja u sustavu za fokusiranje) zavisno od točke na ekranu prema kojoj se usmjerava zraku.
  + Za ovo je potrebna dodatna elektronika za kontrolu otklonskog sustava i sustava za fokusiranje
* Znatan dio elektrona iz elektronskog topa udara u masku, izazivajući njeno zagrijavanje i izobličavanje materijala kao posljedicu zagrijavanja
* Deformacija metalne maske dovodi do toga da elektronski zrak djelimično ili potpuno promašuje točke na ekranu - nekvalitetna slika
* I još: da bi se osigurala svjetlija i kontrastnija slika potrebno je pojačati pobudu fosfornih tačaka na ekranu, a to se postiže ubrzavanjem elektrona - novo zagrijavanje

**RJEŠENJE**

1. Primijena "invar shadow -mask" katodnih cijevi - za izradu maske se koristi nova legura, "INVAR", vrlo visoke temperaturne stabilnosti. Upotreba ove legure omogućava veće brzine elektrona u snopu, čime se dobije svjetlija slika i bolji kontrast u odnosu na klasične katodne cijevi
2. TRINITRON (i njegova podvrsta DIAMONDTRON) i CHROMACLEAR kolor katodne cijevi (ne koriste klasične maske)

Trinitron cijevi koriste tanku rešetku napravljenu od vertikalno razapetih žica (*aperture-grille*). Fosforna masa se na ekran nanosi u vidu vertikalnih pruga (*stripes*) naizmjenično crvenih, zelenih i plavih. Elektronski topovi za tri osnovne boje poredani su u liniju.

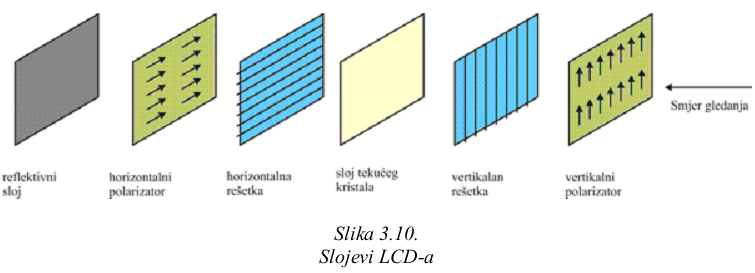


Trinitron koristi jedan sistem za fokusiranje za sve tri elektronske zrake.

Površina šupljina ovakve maske mnogo je veća u odnosu na masku u shadow-mask katodnim cijevima - više elektrona prolazi do ekrana – emitira više svjetlosti od klasičnih kolor katodnih cijevi – svijetle i jakim boje, dobar kontrast.

**TEHNOLOGIJA PRIKAZNOG UREĐAJA S TEKUĆIM KRISTALOM (LCD)**

Zasniva se na osobini materijala poznatog pod nazivom "tekući kristal" da rotira polaritet svjetlosti koja prolazi kroz njega. Tekući kristal je sastavljen od makromolekula koje u normalnom stanju zauzimaju spiralan položaj. U ovakvom stanju tekući kristal okreće polaritet svjetlosti koja prolazi kroz njega. Kada se tekući kristal stavi u električno polje, njegove molekule se poravnavaju i efekt uvrtanja polarizacije svjetlosti nestaje sve dok se električno polje ne ukloni.



*Princip rada*

1. Nepolarizirana svjetlost dolazi do sloja vertikalnog polarizatora ali samo vertikalno polarizirani dio svjetlosti prolazi kroz njega
2. Vertikalno polarizirana svjetlost prolazi kroz stakleni sloj sa vertikalnim adresnim linijama i kroz tekući kristal
3. Ako adresne linije nisu pod naponom - tekući kristal okreće polarizaciju svjetlosti pa horizontalno polarizirana svjetlost prolazi kroz horizontalni polarizator
4. Polarizirana svjetlost se reflektira na reflektivnom sloju i ponovo prolazi kroz horizontalni polarizator
5. Ponovnim prolaskom kroz sloj tekućeg kristala, polarizacija svjetlosti se okreće i njen vertikalno polarizirani dio prolazi vani kroz vertikalni polarizator.
6. Na LCD -u se vidi svijetla površina

Ako jednu horizontalnu adresnu liniju dovedemo na pozitivan potencijal, a jednu vertikalnu adresnu liniju na negativan potencijal, na mjestu njihovog presjeka stvorit će se električno polje. Molekule tekućeg kristala na tom mjestu će se poravnati, a polarizacija svjetlosti koja prolazi kroz ovu točku na LCD -u neće biti okrenuta -> vertikalno polarizirana svjetlost nedirnuta prolazi kroz tekući kristal - horizontalni polarizator je upija. Na tom mjestu na LCD –u vidjet ćemo tamnu točku.

Točka između pozitivno nabijene i negativno nabijene linije – zatamnjena, ostale osvjetljene. Slika se na LCD -u ne iscrtava točku po točku, nego red po red: Horizontalna adresna linija se dovodi na odgovarajući potencijal, a sve vertikalne adresne linije na čijem presjeku sa tom horizontalnom adresnom linijom treba zacrniti točku dovode se na suprotan potencijal. Nakon izvjesnog vremena, naponi se uklanjaju sa adresnih linija, a zatim se proces ponavlja za sljedeći red točaka.

Kontrola potencijala koji se dovode na vertikalne adresne linije - različiti nivoi zatamnjenja u svakoj točki prikazanog rastera. Među nabrojane slojeve može se dodati poseban filter koji omogućava prikaz slika u boji. Filter se sastoji od vertikalnih pruga koje propuštaju samo crvenu, samo zelenu ili samo plavu svjetlost. Ove pruge su grupirane u trijade i raspoređene tako da se ispod svake trake proteže jedna vertikalna adresna linija.

*Problemi*

* Svjetlost mora dva puta proći kroz niz slojeva, zbog čega je slika koju daje LCD dosta tamna i niskog kontrasta. Rješenje: uvođenje pozadinskog osvjetljenja umjesto zadnjeg, reflektivnog sloja.
* Tromost molekula tekućeg kristala: Da bi se molekule potpuno poravnale potrebno je tekući kristal izvjesno vrijeme zadržati u električnom polju -> ograničena brzina osvježavanja zaslona, koja opada sa porastom broja redova zaslona koje treba osvježiti
* Nakon uklanjanja električnog polja molekulama treba neko vrijeme za vraćanje u normalan položaj (zavisno od perzistencije, slično fosforu).
* Da bi se dobila stabilna slika, zaslon se mora osvježavati znatno brže neko što je molekulama potrebno da se vrate u normalno stanje. Ako to nije slučaj: slika na zaslonu je treperava, blijeda i slabog kontrasta.
* Problem treperenja se djelimično može riješiti povećanjem perzistencije tekućeg kristala, ali to zaslon čini tromim i nepogodnim za prikaz dinamičnih slika
* Drugo rješenje - povećanje brzine osvježavanja zaslona i to:

1. DUAL-SCAN" LCD (istovremeno osvježavanje dvaju redova tačaka – ekran iz dva dijela)
2. Pojačanje električnog polja kojim djelujemo na molekule tekućeg kristala

*Pojačanje električnog polja*

Na svaki presjek adresnih linija dodan je tranzistor koji služi kao pojačivački element za brzo izravnavanje molekula tečnog kristala kao i za kontrolu stupnja tog izravnavanja (kako bi se dobili različiti nivoi zatamnjenja) -> "ACTIVE MATRIX" ili TFT LCD. Upotreba tranzistora omogućila je mnogo veće brzine osvježavanja i kontrast usporediv sa kontrastom katodnih cijevi.

*Prednosti LCD-a*

* ekstremno mala potrošnja
* mala težina
* mala debljina
* ravan zaslon
* odsustvo bilo kakvog zračenja

Pasivni LCD-i su i izuzetno jeftini, ali kvaliteta prikaza nije ni blizu zadovoljavajućem. Kvaliteta prikaza TFT LCD-a je usporediva sa kvalitetom prikaza katodnih cijevi.

* CCFL LCD: fluorescentne CCFL lampe (6-20 redova) Upravljanje svjetiljkama zahtjevno
* Direct LED: izvor svjetlosti štedne LED diode (ravnomjerno po cijeloj površini ekrana). Svjetliji, bolji dinamički kontrast, manja potrošnja.
* EDGE LED LCD: bijele LED diode u rubovima ekrana.

Ne mogu potpuno zatamniti niti jedan dio na ekranu.

Matrični (segmentni) LED PRO: LED diode (po 8) u segmente postavljene po cijeloj površini ekrana. Npr 40” ekran - 168 segmenata. Segmenti mogu prilagođavati svjetlinu ili se potpuno ugasiti tako da je moguća potpuno crna boja ili jako visoka svjetlina. Vrijeme odziva – 0.5 ms.

**PLAZMA ZASLONI (PLASMA DISPLAY PANELS, PDP)**

Iako je "plazma" postao sinonim za tehnologiju ravnih ekrana, taj termin zapravo označava posebno tzv. "četvrto stanje tvari" (uz tri osnovna: čvrsto, tekuće i plinovito) koje sadrži velik broj iona (el. nabijene čestice) : negativno nabijenih elektrona i pozitivno nabijenih atoma koje utječu na njena električna svojstva i ponašanje. Većina plazmi, uključujući i onu koja se koristi u proizvodnji prikaznih uređaja također sadrži velik broj neutralnih čestica (nemaju naboj).

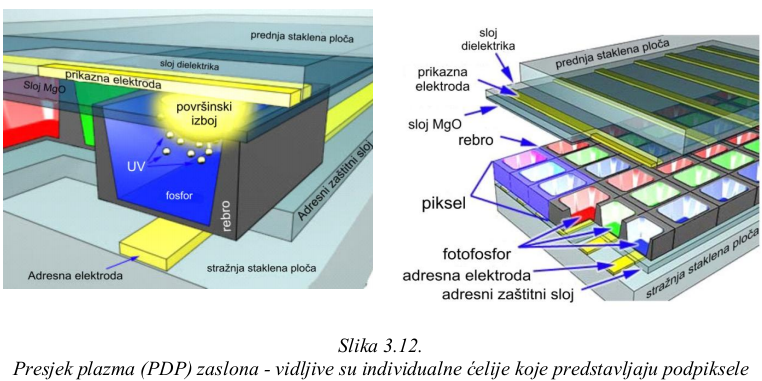
*Princip rada PDP-a*

Kada je plazma pobuđena električnom strujom, negativno nabijene čestice se ubrzano kreću ka pozitivno, a pozitivno nabijene ka negativno nabijenom području plazme. Pri tom se čestice neprestano sudaraju jedna s drugom. Ovi sudari pobuđuju atome plina u plazmi, stimulirajući ih da otpuste svjetlosne fotone.

Ti otpušteni UV fotoni nisu vidljivi ljudskom oku, međutim oni mogu pobuditi vidljive svjetlosne fotone. Unutrašnjost podpikselnih ćelija obložena je fosforom. Fosfor je kemijski element koji, kada je izložen svjetlu, otpušta određenu količinu svjetlosti. Kada ultraljubičasti foton pobudi atom fosfora u ćeliji, jedan od elektrona fosfora skače na viši energetski nivo i atom se zagrije. Pri povratku na njegov normalni energetski nivo, elektron otpušta energiju u vidu vidljivog svjetlosnog fotona.

*Unutrašnjost ekrana*

Između dvije staklene ploče smještene su stotine tisuća sićušnih ćelija u kojima su sadržani xenon i neon. Adresne i transparentne prikazne elektrode također su ukliještene između staklenih ploča : adresne elektrode uzduž stražnje, a prozirne prikazne elektrode uzduž prednje staklene ploče. Prozirne prikazne elektrode prekrivene su zaštitnim slojem magnezij-oksida i okružene dielektričnim izolatorom.

Obje skupine elektroda (adresne i prozirne prikazne) se protežu preko cijelog ekrana. Prikazne elektrode su složene u horizontalne retke duž ekrana, a adresne elektrode u vertikalne stupce. Skupa čine osnovnu mrežu.

Kako bi ionizirao plin u pojedinoj ćeliji, dovodi se naboj na elektrode koje se sijeku u toj ćeliji. Kad su te elektrode nabijene (naponskom razlikom između njih), struja poteče kroz plin u ćeliji, stvarajući ubrzan tok nabijenih čestica, što stimulira atome plina u ćeliji na otpuštanje ultraljubičastih fotona. Otpušteni ultraljubičasti fotoni reagiraju s fosfornim materijalom kojim je obložena unutrašnjost zidova ćelija. Pobuđeni fosfor tada svijetli ljudskom oku vidljivim bojama.

*Prednosti PDP-a*

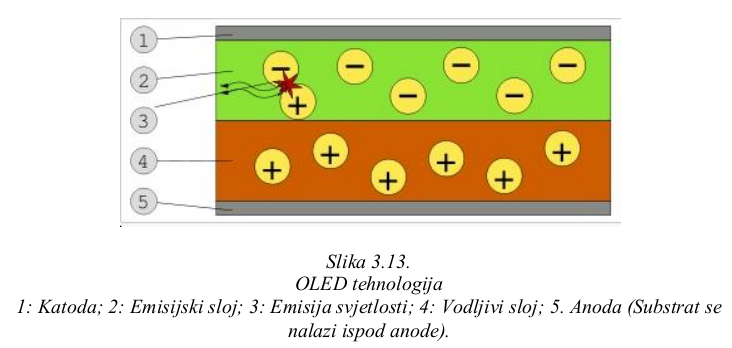
* Masovna proizvodnja ekrana većih dijagonala
* Bolje osvijetljenje
* Visok kontrast pri svim uvjetima osvijetljenja (i u tamnijim i u svijetlijim prostorijama)
* Vjernije, živopisnije boje pri reprodukciji slike
* Mogućnost postizanja tamnije nijanse crne boje nego kod LCD-a
* Besprijekoran prikaz brzo pokretne slike
* Dulji vijek trajanja i veća pouzdanost piksela

*Mane PDP-a*

* Krhki su, što otežava njihov transport i postavljanje.
* Veoma su skupi, iako su u klasi ekrana s većim dijagonalama trenutno jeftiniji od LCD-ova gledano po jedinici veličine
* Kod novijih PDP-ova “burn-in” efekt se često uzrokuje zbog polarizacije čestica plina, iako može biti poništen (ostavljanjem ekrana na "snježnom" ili statičnom kanalu određeno vrijeme čime se depolariziraju čestice plina.) LCD-ovi nisu podložni permanentnom "burn-in"-u.
* Osvijetljenje je najbolje tijekom prvih 2 000 sati rada. Svakim satom kasnije PDP postepeno gubi osvijetljenje.Kontroliranjem toka struje kroz ćelije može se regulirati intezitet svake podpikselne boje, i različitim kombinacijama crvene, zelene i plave proizvesti cijeli spektar boja.

**OLED ZASLONI (ORGANIC LIGHT EMITTING DISPLAY)**

Za razliku od LCD-ova koji su transmisijski ekrani, OLED ekrani su emisijski kao i CRTovi, za rad ne trebaju nikakvo pozadinsko osvjetljenje. S obzirom da sami emitiraju svjetlost, potrošnja energije je puno manja od LCD-ova, boje su vjernije i kut gledanja je znatno veći. Osnovu čine slojevi organskog materijala koji se nalaze između dvije metalne ploče. Kada se na metalne ploče dovede napon dolazi do emitiranja svjetlosti. Za prikaz u boji organski se sloj dodatno dopira malom količinom fluorescentnih molekula određene boje.



1. Napon na anodi pozitivniji od napona na katodi -> struja elektrona koji se kreću s katode prema anodi
2. Katoda daje elektrone u emisijski sloj, a anoda povlači elektrone iz vodljivog sloja (zbog djelovanja anode stvaraju se šupljine u vodljivom sloju)
3. Emisijski sloj postaje negativno nabijen, a vodljivi sloj postaje pozitivno nabijen zbog nastalih šupljina
4. Dolazi do rekombinacije
5. Rekombinacija se odvija bliže emisijskom sloju zbog svojstva organskih poluvodiča da su im šupljine pokretljivije od elektrona
6. Prilikom rekombinacije dolazi do pada energetskog nivoa elektrona kojeg prati emisija vidljive svjelosti (emisijski sloj)
7. Boja svjetlosti koju će OLED emitirati ovisi o vrsti organskih molekula u emisijskom sloju.

*Pasivni i aktivni zasloni*

Pasivni se koriste u jeftinijim uređajima koji ne zahtijevaju veliku brzinu obnove prikaza i kvalitetu, a aktivni se koriste u uređajima namijenjim multimediji.

S obzirom na brojne prednosti ovo je najverojatnija tehnologija koja će u budućnosti zamjeniti LCD zaslone u većini primjena.

**GRAFIČKO SKLOPOVLJE III**

**ANTIALIASING**

**RASTERSKI PRIKAZNI SUSTAVI**

Opisane tehnologije zaslona su rasterske (CRT, LCD). Za prikaz slike na bilo kojem od ovih zaslona potrebno je osigurati dodatni elektronski sklop koji će sliku zapisanu u memoriji računala prilagoditi i prikazati na zaslonu. Ovaj elektronski sklop naziva se video kontrolerom. Njegov zadatak: sinkronizacija i upravljanje procesom iscrtavanja i osvježavanja slike na zaslonu.

CRT, LCD - nemaju mogućnost pamćenja slike. Iscrtana slika se treba neprestano osvježavati

uz video kontroler potrebno je osigurati i dovoljno brze memorije za pohranjivanje slike koja se prikazuje na zaslonu. Ove komponente organizirane na odgovarajući način čine prikazni sustav.

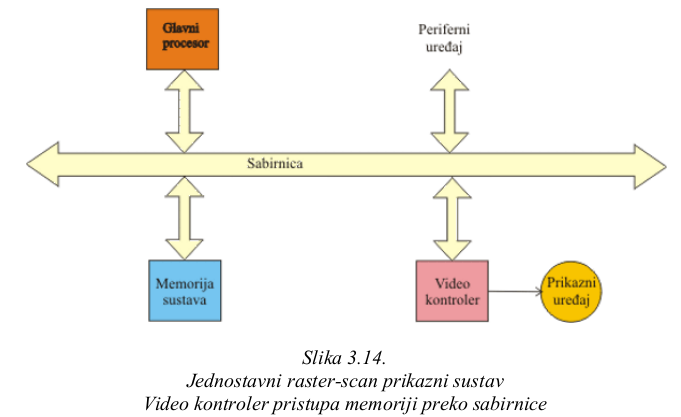
Osim video kontrolera i memorije, prikazni sustav može posjedovati i elektroniku koja rasterećuje CPU. Rasterećenje: poslovi obrade i prilagođavanje slike za prikaz na zaslonu (jedan ili više specijaliziranih procesora, tzv. prikazni procesori ili grafički procesori).

Osnovna razlika između raznih prikaznih sustava je u načinu raspoređivanja i organizacije ovih komponenti. Video kontroler je s jedne strane uvijek vezan sa zaslonom. Njegov zadatak je da zaslonu isporučuje piksel po piksel, red po red slike, brzinom kojom se ti redovi na zaslonu osvježavaju. Slika se nalazi zapisana u dijelu memorije koji se naziva međuspremnik okvira (frame buffer), odakle video kontroler preuzima informacije, prilagođava ih i šalje zaslonu.

**JEDNOSTAVNI RASTERSKI PRIKAZNI SUSTAV**

Osnovna svojstva ove arhitekture su:

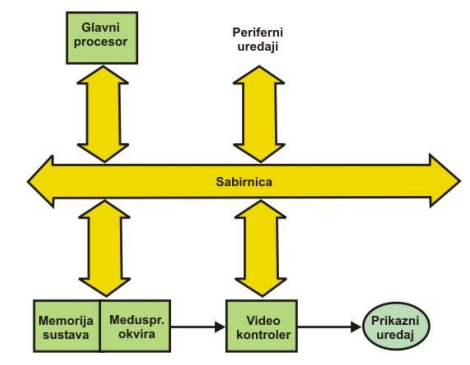
* odnos memorije i glavnog procesora je isti kao kod negrafičkih sustava,
* dio memorije služi kao bitovna matrica,
* aplikacijski program i grafički paket pohranjeni su u memoriji sustava, a izvodi ih glavni procesor,
* video kontroler prikazuje slike pohranjene u okvirnom međuspremniku,
* video kontroler koji sadrži generator adresa i otklonskih signala adresira lokacije međuspremnika okvira u memoriji, a podaci određuju intenzitet ili boju slikovnih elemenata



Nedostatak:

* Sporost i veliki broj pristupa memoriji
* Brzina sistemske sabirnice je ograničena, a uvijek se mora ostaviti dovoljno kapaciteta za rad ostalih komponenti računalnog sustava.

**JEDNOSTAVNI RASTERSKI PRIKAZNI SUSTAV (2)**



Druga jednostavna vrsta rasterskih prikaznih sustava (slika). U ovom slučaju, video kontroler može direktno pristupiti memoriji sustava, bez korištenja sabirnice.

Dvostruko portirana ili dvoulazna memorija – može se vezati istovremeno na dvije sabirnice.

1. Sa jedne strane: sistemska sabirnica preko koje joj pristupa glavni procesor (promijena sadržaja međuspremnika okvira)
2. S druge strane: brza sabirnica, preko koje joj pristupa video kontroler (omogućuje da CPU upravlja video kontrolerom).

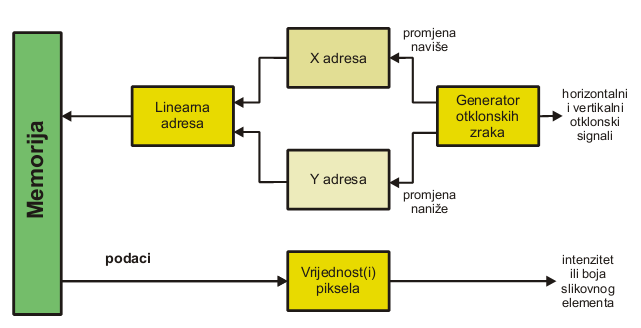
Omogućava veće rezolucije, više boja i brže osvježavanje.

Negativno: maksimalna veličina međuspremnika okvira je fiksna (mora se nalaziti u dvostruko portiranoj memoriji) - fizičko ograničenje veličine slike i broja boja koje se mogu prikazati.

**VIDEO KONTROLER**

Video kontroler ima zadaću stalnog osvježavanja zaslona. Da bi se izbjegao efekt treperenja osvježavanje se treba obavljati frekvencijom od minimalno 60 Hz. Ako se svi redci piksela u rasteru osvježavaju istovremeno: sustav bez prepletanja (noninterlaced). Ako se izmjenično osvježavaju parni i neparni redci u rasteru: sustav s prepletanjem (interlaced).

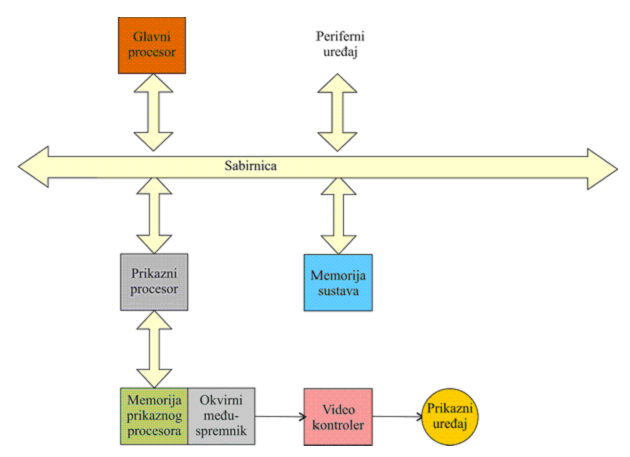
1. Generator horizontalnih i vertikalnih otklonskih signala usmjerava zraku na određeni slikovni element na zaslonu. Istovremeno generira i horizontalnu i vertikalnu adresu piksela u koordinatnom sustavu zaslona.
2. Na temelju tih koordinata - izračunava se adresa pripadajuće memorijske lokacije (s podacima o svjetloći/ boji piksela).
3. Postavljaju se parametri svjetlosnih izvora koji određuju svjetloću ili boju piksela.



**RASTERSKI PRIKAZNI SUSTAV SA ZASEBNIM PRIKAZNIM PROCESOROM**

Zasebni prikazni procesor preuzima grafičke funkcije kao što je pretvorba modela u rasterski prikaz. Procesorske funkcije su raspodijeljene izmedu CPU-a i GPU-a.

U sustavu su definirana tri odvojena memorijska područja:

* glavna memorija sustava
* memorija prikaznog procesora
* zasebni međuspremnik okvira

Memorija sustava sadrži:

* aplikacijski program
* grafički paket
* operacijski sustav

Memorija prikaznog procesora sadrži:

* podatke
* programe za stvaranje slike

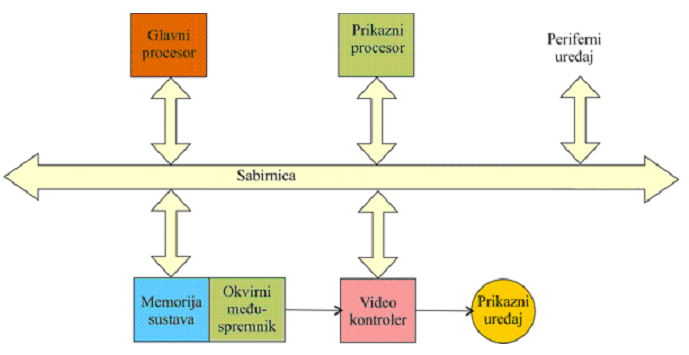
Slika se pohranjuje u okvirni međuspremnik. Ovakvom organizacijom memorije omogućen je istovremeni pristup odgovarajućim dijelovima memorije od strane dvaju procesora i video kontrolera.

Nedostatci:

* Potreba za međusobnom komunikacijom dvaju procesora i prijenosom sadržaja između različitih memorijskih prostora (Na primjer, jedan memorijski sadržaj priprema se i mijenju jednom memorijskom prostoru, a isčitava se iz drugog memorijskog prostora).

Ove komunikacije i prijenosi sadržaja mogu predstavljati značajno opterećenje za sustav u cijelosti.

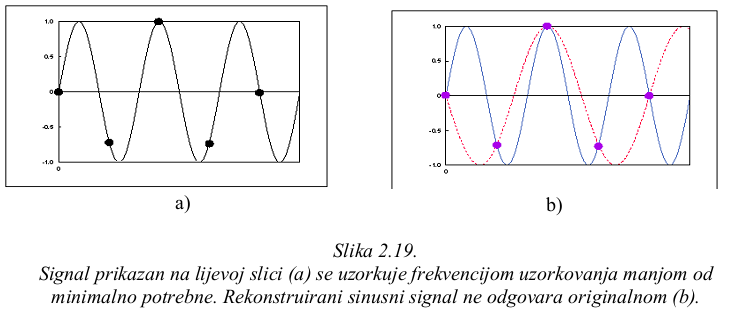
**RASTERSKI PRIKAZNI SUSTAV S JEDINSTVENIM MEMORIJSKIM PROSTOROM**



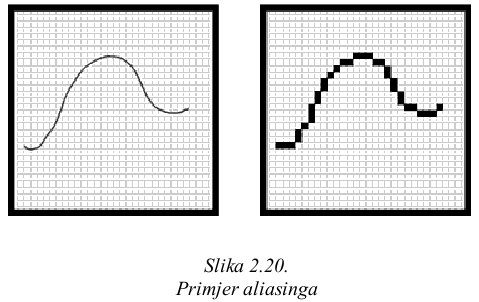
Nedostatci arhitekture rasterskog sustava sa zasebnim prikaznim procesorom mogu se izbjeći objedinjavanjem memorijskog adresnog prostora. U tom slučaju moguća je fleksibilna dodjela memorijskih prostora i pristupa što pojednostavnjuje i programiranje.

Nedostatci:

* Potreba za istovremenim pristupom memoriji od strane 2 procesora.
* Jedno rješenje : korištenje priručne memorije u samom CPU-u (cache memorija) - smanjuje broj i učestalost pristupa memoriji od strane CPU-a

**ANTIALIASING**

Pojam aliasing dolazi iz područja digitalnog procesiranja signala. Da bi uspješno uzorkovali signal, najviša frekvencija u tom signalu ne smije biti veća od Nyquistove. Aliasing nastaje ukoliko ulazni signal ipak sadrži veće rekvencije od Nyquistove, pa će se te frekvencije pogrešno rekonstruirati kao neke druge. Nyquistova frekvencija kod računalne grafike određena je rezolucijom tj. veličinom piksela.



Zašto nastaje aliasing?

* Da bi bilo koji podatak iz realnog svijeta mogli zapisati u računalo, moramo ga prvo digitalizirati, to jest pretvoriti u brojeve.
* Stvarni svijet je kontinuiran, a računalo može jedino raditi s diskretnim vrijednostima.
* Aliasing nastaje zbog uzorkovanja u diskretnim točkama sa preniskom razlučivošću.

Nazubljenost rubova objekata uklanja se povećanjem razlučivosti i tzv. FSAA (*full screen anti-aliasing*) metodama u koje spadaju metode naduzorkovanja, adaptivnog uzorkovanja te njihovih različitih varijacija (OGSS, RGSS, Multisampling). U stvarnom životu udaljeni objekti gube na detaljima zbog fizičkih ograničenja kao i zbog nesavršenosti oka. Zbog toga se koriste manje i detaljima siromašnije teksture za udaljene objekte.

Ovaj proces se naziva "mip-mapping" (latinski: multum in parvam ili "mnogo u malo") jer često postoji mnogo različitih "mip-nivoa" na koje su pojedinačne slike tekstura skalirane za specifičnu udaljenost (LOD).

LOD postavke na video karticama koje se postavljaju od strane igara ili samog korisnika omogućavaju detaljniju sliku objekata uz manje usporavanje rada kartice. Ovaj povećani broj detalja nije fizički ispravan i često dovodi do aliasinga tekstura.

Aliasing tekstura je različit od aliasinga rubova koju pokušava ukloniti FSAA. Kod aliasinga tekstura, neki pikseli teksture dominiraju što za posljedicu ima da izgledaju pokretni umjesto da su stacionarni – to je takozvani efekt plivajućih piksela. Da bi uklonili ove efekte, prikazane slike je potrebno filtrirati nekom od mogućih metoda (npr. bilinearno filtriranje, trilinearno filtriranje, anizotropno filtriranje).

**ANTIALIASING : METODE**

Antialiasing metode mogu se kategorizirati na više načina. Jedna od kategorizacija: pred- i post- metode filtriranja. Pred-filtriranje se odnosi na uklanjanje frekvencija iz slike koje nadilaze Nyquistovu, prije obavljanja uzorkovanja. U osnovi, ovo znači primjenu nisko-pojasnog filtera na sliku i nakon toga uzorkovanje. Za svaki piksel se izračunava područje doprinosa pojedinih poligona.

Pred-filtriranje je jedina metoda koja u potpunosti uklanja efekte aliasinga. Pred-filtriranje se mora obaviti prije rasterizacije (uzorkovanja), i stoga ju nije tako jednostavno dodati standardnom rendereru. Metodepost-filtriranja razmatraju prekrivenost piksela samo u diskretnim točkama, umjesto izračunavanja točne površine.

Metode post-filtriranja se lakše hardverski implementiraju, a i dovoljno su brze za renderiranje u stvarnom vremenu. Zbog njihove diskretne naravi, one nikad u potpunosti ne mogu ukloniti aliasing. Sve što one mogu jest pomaknuti prag aliasinga više u frekvenciji dodavanjem subpiksel uzoraka, i po mogućnosti zamijeniti vidljivi aliasing za šum (pseudoslučajnim uzorkovanjem piksela) ili zamućenjem (blurring-interpoliranjem više subpiksela u jednu vrijednost).

*Metoda 1: povećanje razlučivosti.*

Kako aliasing nastaje prvenstveno zbog niske razlučivosti, jednostavna metoda je povećati razlučivost. Međutim, povećavanje razlučivosti zahtijeva sofisticiran hardware, to jest preciznije izlazne jedinice, i dodatnu procesorsku snagu.

Ovo nije prihvatljivo rješenje radi velikog povećanja cijene sustava. Na ovaj način ne uklanjamo u potpunosti aliasing jer je povećanje razlučivosti uvijek konačno, a slika može sadržavati beskonačno visoke frekvencije (npr. oštri rubovi).

*Metoda 2: Supersampling (naduzorkovanje, višestruko uzorkovanje)*

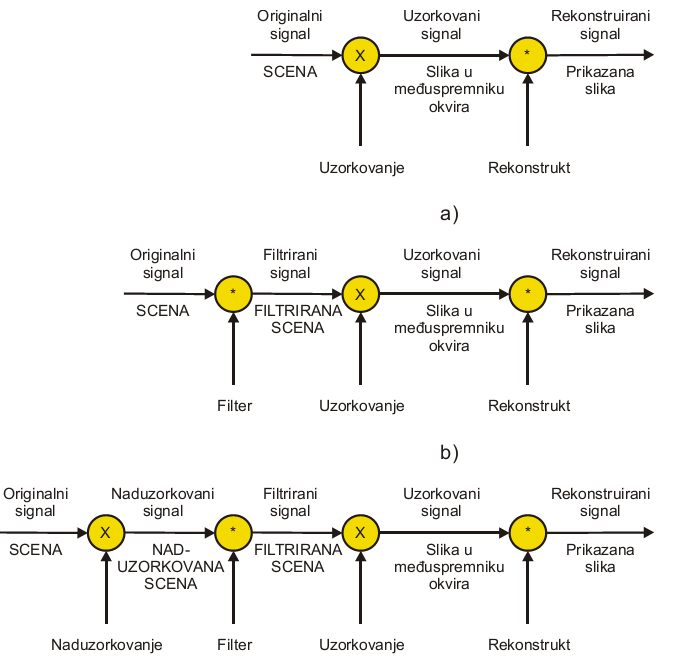
Često se koristi i izraz multisampling. Uzorkujemo sliku na većoj razlučivosti, a zatim filtrom uklonimo visoke frekvencije. Drugim riječima, uzimamo više uzoraka po svakom pikselu (na različitim mjestima unutar piksela), i zatim računamo srednju vrijednost dobivenih uzoraka.

Osim srednje vrijednosti često se koriste i drugi filteri. Prilikom korištenja supersamplinga iz slike ćemo uspješno otkloniti sve frekvencije do nove povećane gornje Nyquistove frekvencije (ona se poveća zbog povećane razlučivosti). Rub poligona može imati polutonove jer boju piksela određujemo iz srednje vrijednosti više uzoraka.

Na ovaj način ne uklanjamo u potpunosti aliasing jer je povećanje razlučivosti uvijek konačno, a slika može sadržavati beskonačno visoke frekvencije (npr. oštri rubovi). Ova metoda najčešće izgleda dobro zato što u foto realističnim slikama jako visoke frekvencije imaju jako male amplitude, pa se ne primjećuju. Ova metoda je procesorski vrlo zahtjevna (koliko puta povećamo razlučivost, toliko puta više moramo procesirati).

**OGSS** – *Ordered Grid SuperSampling*; najjednostavniji način Antialiasing objekata. Izvršava se renderiranjem dvije (2x) ili četiri (4x) puta više rezolucije, izračunavaju se prosječne vrijednosti okolnih piksela te slika vraća u početnu rezoluciju. Zamućuje teksture i sliku.

**RGSS** – *Rotated Grid SuperSampling*; renderira se više slika u istoj rezoluciji pod kutem od 30 stupnjeva u različitim smjerovima. Uspoređuje dijelove piksela i ujednačava ih. Ne zamućuje teksture.

**MultiSampling AA**; akumulacijski međuspremnik koristi algoritam koji uspoređuje boju iz originalnog uzorka za sve pod-uzorke (ovisno o modu – 2, 4, 8 uzoraka unutar jednog piksela) i izračunava idealnu boju za svaki pojedini uzorak.

*Rasterizacijski sustavi*

1. Konvencionalno uzorkovanje točaka
2. Idealni antialiasing s filtriranjem
3. Antialiasing u praksi – sa naduzorkovanjem i filtriranjem

*Adaptive supersampling*

* Kod metode adaptive supersamplinga (prilagodljivog naduzorkovanja) o razlučivosti uzorkovanja odlučujemo lokalno.
* Pri tome je potreban algoritam koji zna kad treba povećati razlučivost.
* Povećamo razlučivost naduzorkovanja na rubovima poligona i u vrhovima, jer se na tim mjestima pojavljuju nagli prijelazi.
* Ovo je najpopularnija metoda za rubove poligona (3D akceleratori).

*Metoda 3: Uzorkovanje površine (area sampling).*

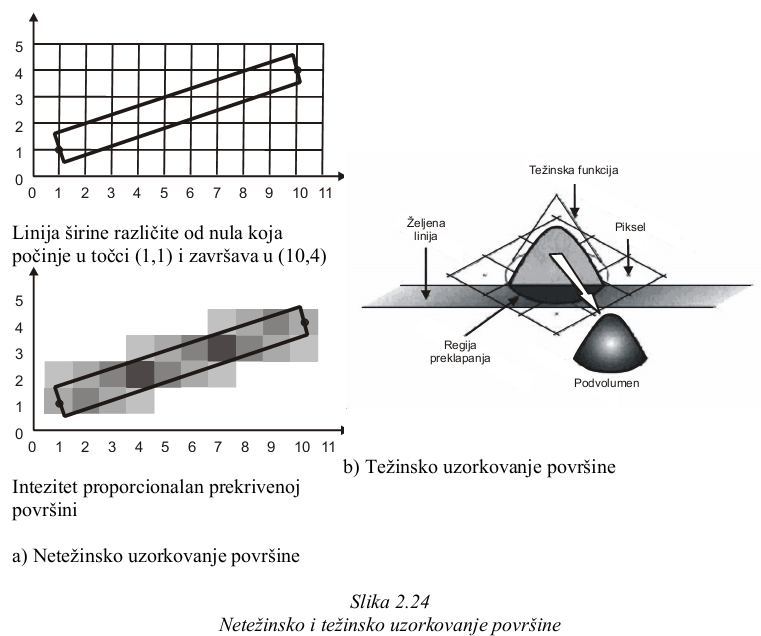
Kod ove metode visoke frekvencije se uklanjaju filtriranjem.

a) **Netežinsko uzorkovanje površine** (*unweighted area sampling*):

* Intenzitet piksela je proporcionalan prekrivenoj površini piksela.
* Pikseli u stvarnosti najčešće nisu kvadrat (ćelije CCD-a na digitalnim fotoaparatima, pikseli na monitoru...).
* Ljudska percepcija nije ustrojena da prepoznaje piksele kao kvadratične oblike (mi raspoznajemo linije i oblike)

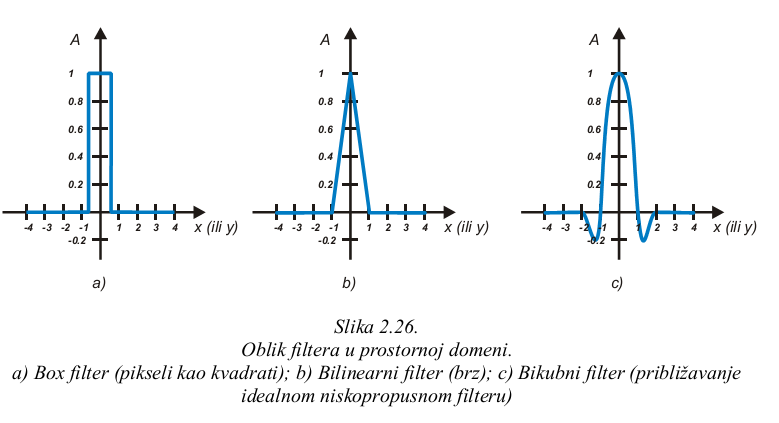
b) **Težinsko uzorkovanje površine** (*weighted area sampling*):

* Piksel je svjetliji u središtu nego na rubu.
* Zato u središte stavljamo veću "težinu".
* Izbor težinske funkcije zapravo je pitanje odabira optimalnog filtera.
* Filteri mogu biti procesorski vrlo zahtjevni i zato se najčešće koriste jednostavni filteri.



**ANTIALIASING : FILTERI**

* Box filter (netežinski)
* (sin x)/x - procesorski prezahtjevan jer nije ograničen u prostoru
* Gausov filter - dobra kvaliteta slike (izgleda zvonoliko). Također nije ograničen u prostoru, ali sa najčešće ograniči na nekoliko susjednih uzoraka.
* Brzi filteri: (oni se najčešće koriste)
  + Linearni – izgleda kao obrnuto slovo V. u svakoj dimenziji koristi dva susjedna uzorka, dakle potrebno je 4 uzorka za bilinearno filtriranje.
  + Kubni – u svakoj dimenziji koristi 4 susjedna uzorka, dakle potrebno je 16 susjednih uzoraka za bikubno filtriranje
* Bikubno filtriranje - Također dobro, ali sporije. U slučaju da smanjujemo sliku koristimo bikubno filtriranje. Oblik kubnog polinoma može biti različit, pa možemo imati "soft", "normal" i "sharp" bikubne filtere.
* Anizotropno filtriranje - Poboljšava filtriranje tako što pri ubacivanju novog piksela mijenja boju okolnih da bio bolje uklopljen. Ono omogućava postupno smanjenje veličine mip–mape s udaljenošću bez primjetnih linija razgraničenja različitih područja.



**POVIJESNI PREGLED**

1885 - Katodna cijev CRT (Cathode Ray Tube)

1887 - Edison patentira filmsku kameru

1926 - J.L. Baird: izum Televizije

1951 - MIT: rana vektorska grafika

1959 - Prvi CAD sistem (IBM)

1962 - Ivan Sutherland razvio je Sketchpad za izradu inženjerskih crteža direktno na zaslonu; prvi interaktivni grafički sustav

- SpaceWar izradio Steve Russell - Prva računalna igra

1963 - Doug Engelbart razvio je prvog miša

1966 - Prvi proizvod RG za tržište: Odyssey Pinball

1967 - Prvi simulator letenja u realnom vremenu (NASA)

1972 - ATARI i Pong

1974 - Intel razvija procesor the 8080.

1975 - Mandelbrot crta fraktale

- Bill Gates osniva Microsoft

1976 - Steve Jobs i Steve Wozniak osnivaju Apple.

1977 - Američka filmska akademija uvodi novu kategoriju:Visual Effects category for Oscars.

- Star Wars : oskar za specijalne efekte

1978 - Superman: oskar za specijalne efekte

1981 - IBM proizvodi prvi IBM PC (16 bit 8088 chip)

rane 1980-te - prva osobna računala, sa ugrađenim mogućnostima rasterskog prikaza (Apple MAC, IBM PC) dovela su do širokog korištenja bitmap grafike i interaktivne grafike.

1983 - Prva Coca-Cola reklama s polarnim medvjedom

1985 - GKS (Graphical Kernel System) – prvi usvojeni grafičstandard - razvoj prenosivih programa koji ne ovise o specifičnostima pojedinog prikaznog uređaja

- The Last Starfighter – prvi igrani film s realističnom računalnom animacijom detaljno izrađenih modela

~1985 - IBM PC uglavnom poslovne aplikacije u tekst modu (obrada teksta i sl.), grafika se postupno poboljšava

- CGA (Color Graphics Adapter) omogućava grafiku u 4 boje, EGA (Enhanced Graphics Adapter) omogućava grafiku u 16 boja i višu rezoluciju (640 x 350).

1988 - GKS – 3D

- PHIGS (Programmer's Hierarhical Interactive Graphic System )

1989 - Abyss prvi igrani film s uvjerljivom 3D animacijom likova.

1990 - Windows 3.0

1992 - PHIGS Plus

1993 - Myst

1994 - Playstation i N64

1995 - Quake (Id Software)

- Toy story: Prvi u potpunosti 3D računalom animirani film.

1996 - Independence Day: oskar za vizualne efekte

1997 - Titanic: oskar za vizualne efekte

1999 - The Matrix

2000 - Sony Playstation II

2002 - Microsoft: XBOX

- Oskar za najbolji animirani film : Shrek

2004 - Shrek 2

- Više od 330 HP radnih stanica koristili su digitalni umjetnici u interaktivnom postupku izrade filma.

- Radna stanica:

- Red Hat Linux 7.2 OS

- Dual-Intel Xeon 2.4 Ghz procesor, 2GB RAM

- NVIDIA Quadro 4 XGL grafička kartica

- Dual monitori

- Za renderiranje:

- 347 LP-1000 1.2 GHZ P3 dual procesor servera sa 2 GB RAM-a

- 433 ProLiant DL360 2.8 GHZ P4 dual procesor servera sa 4 GB RAM-a

2008 - Ratatouille (bez mocap); Pixar, Oskar za animirani film 2008

- Beowulf (nominiran); Sony Pictures Imageworks, Performance-capture (mocap), moguće pogledati u formatima: IMAX 3D, Real D i standardnom 2D formatu.

2010 /2011 - Up / Toy story 3; Pixar / Disney

2050 - ?