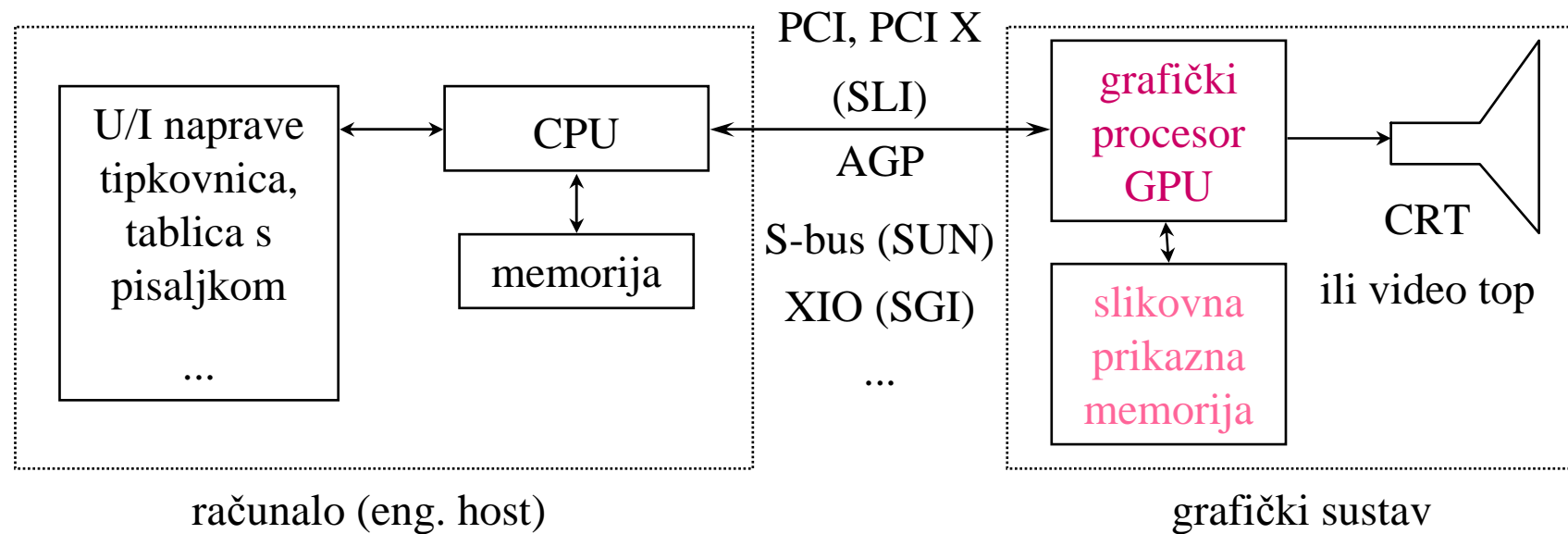


## 2 Računalna grafička oprema

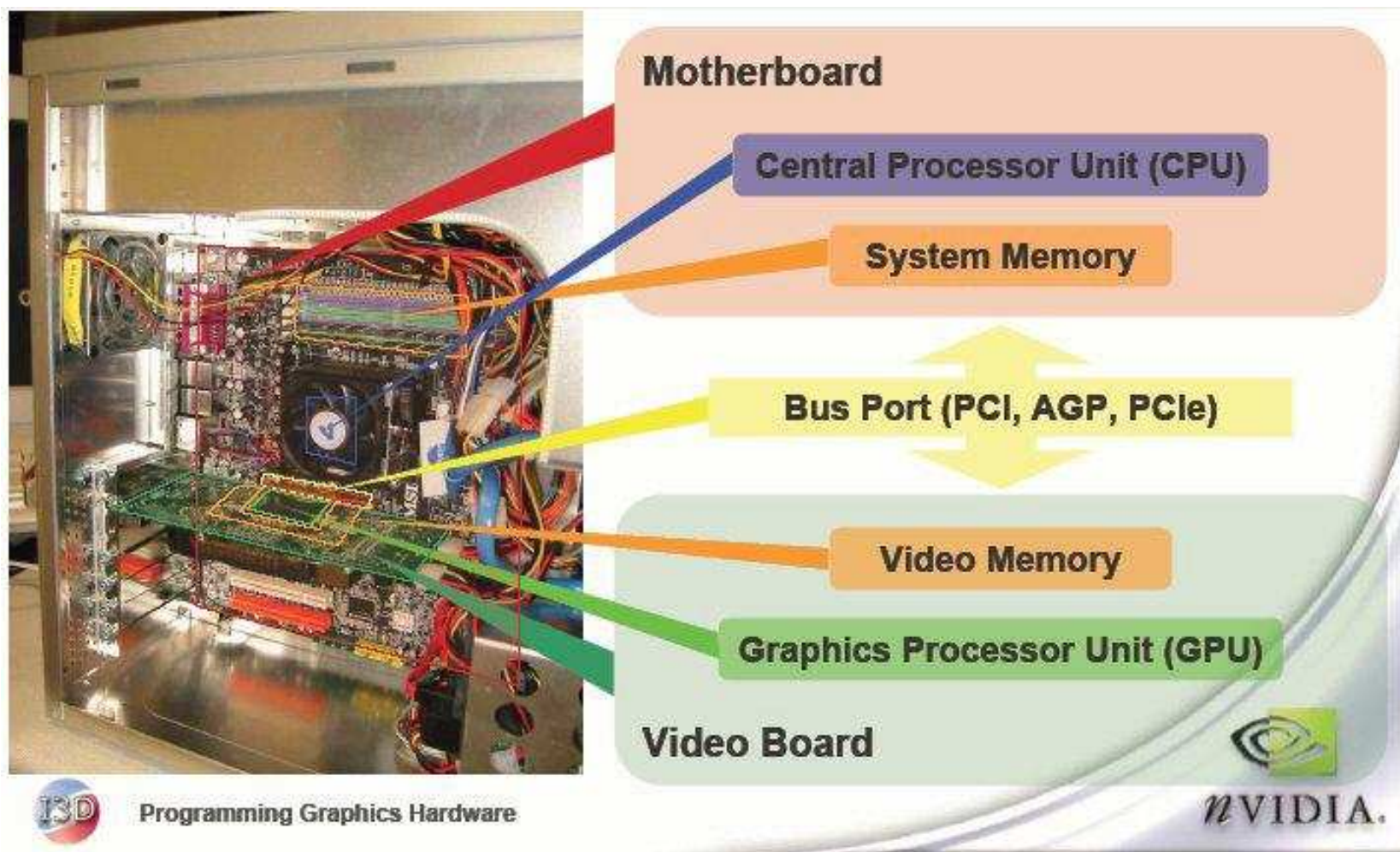
- sklopovska grafička oprema
  - grafički procesor GPU
    - rasterska prikazna procesna jedinica
    - vektorska prikazna procesna jedinica
  - ulazne grafičke naprave
  - izlazne grafičke naprave
- programska grafička oprema
  - knjižnica grafičkih rutina
  - grafička jezgra načinjena u okviru standarda (API), jezici za sjenčanje
  - gotovi programski paketi
    - za crtanje - CAD, animacije
    - za prikaz podataka

## 2.1 SKLOPOVSKA GRAFIČKA OPREMA

Povezanost grafičkog procesora s ostalim jedinicama sustava



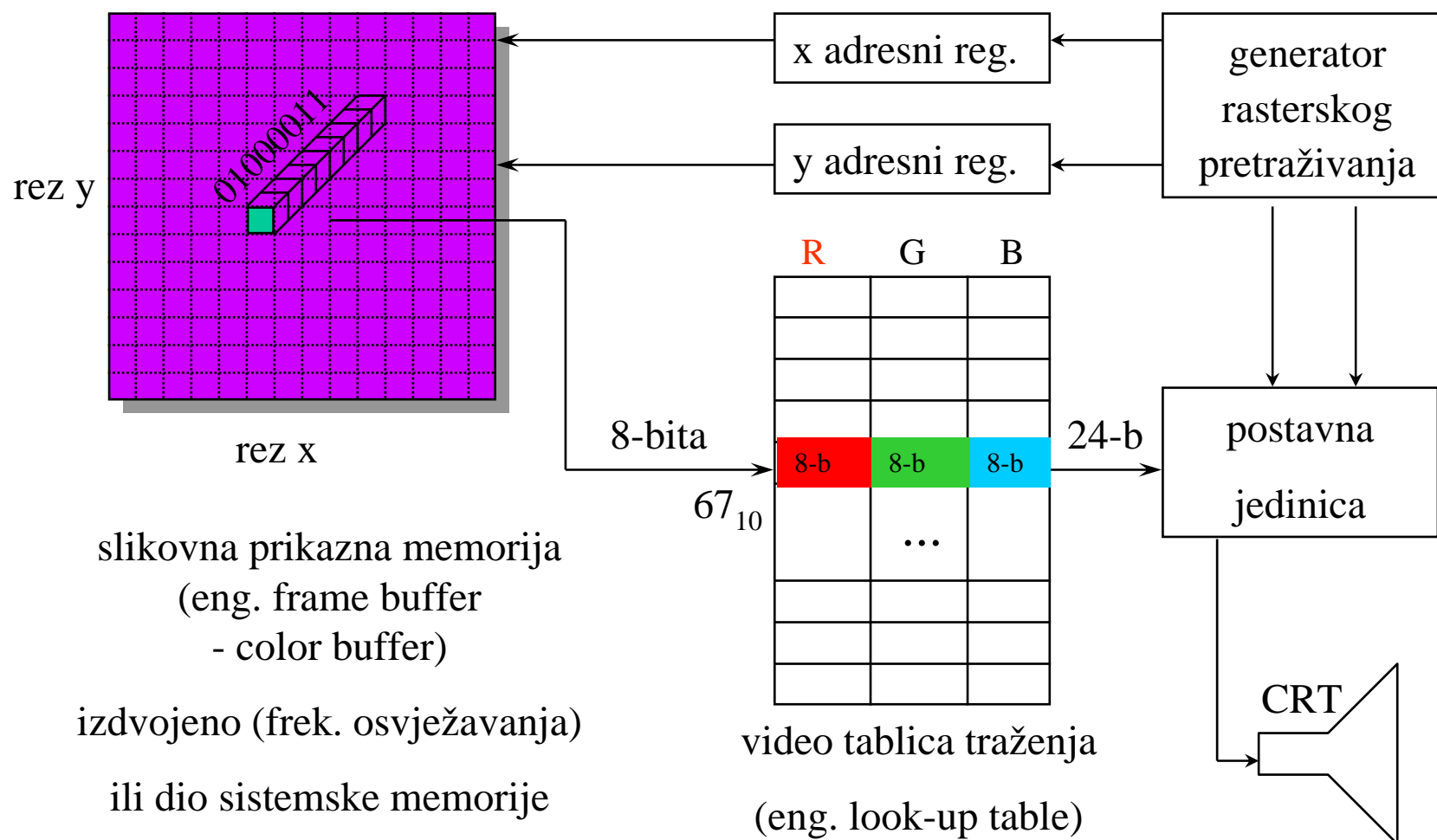
\* Primjer sklopovske grafičke opreme



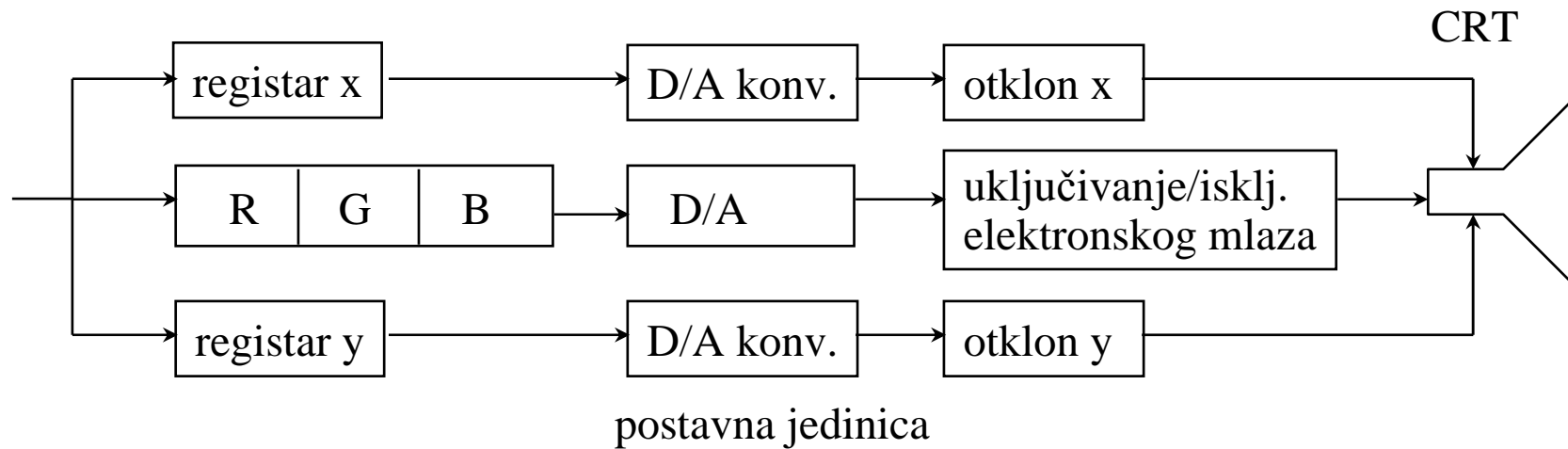
## 2.1.1 Grafički procesor GPU

### Funkcija rasterske prikazne procesne jedinice

1, 4, 8, 24, 32, .. 256 bita



LUT applet [http://www.hdm-stuttgart.de/~rk020/Files/Computeranimation/lut/applet/lut\\_applet.html](http://www.hdm-stuttgart.de/~rk020/Files/Computeranimation/lut/applet/lut_applet.html)



1-bit  
2 boje

4-bita (LUT)  
16 boja

8-bita (LUT)  
256 boja

24-bita  
16 777 216 boja

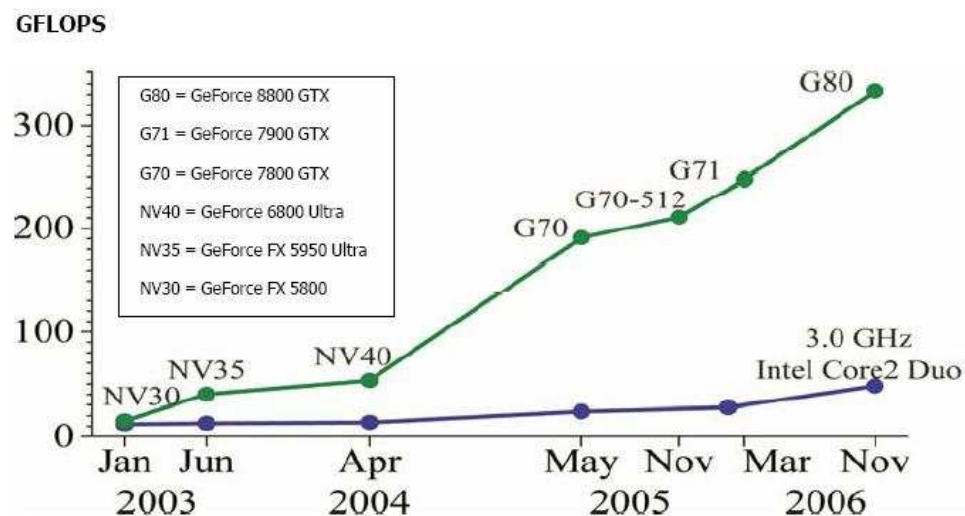
<http://www.cs.technion.ac.il/~cs234325/Applets/applets/dither/html/index.html>

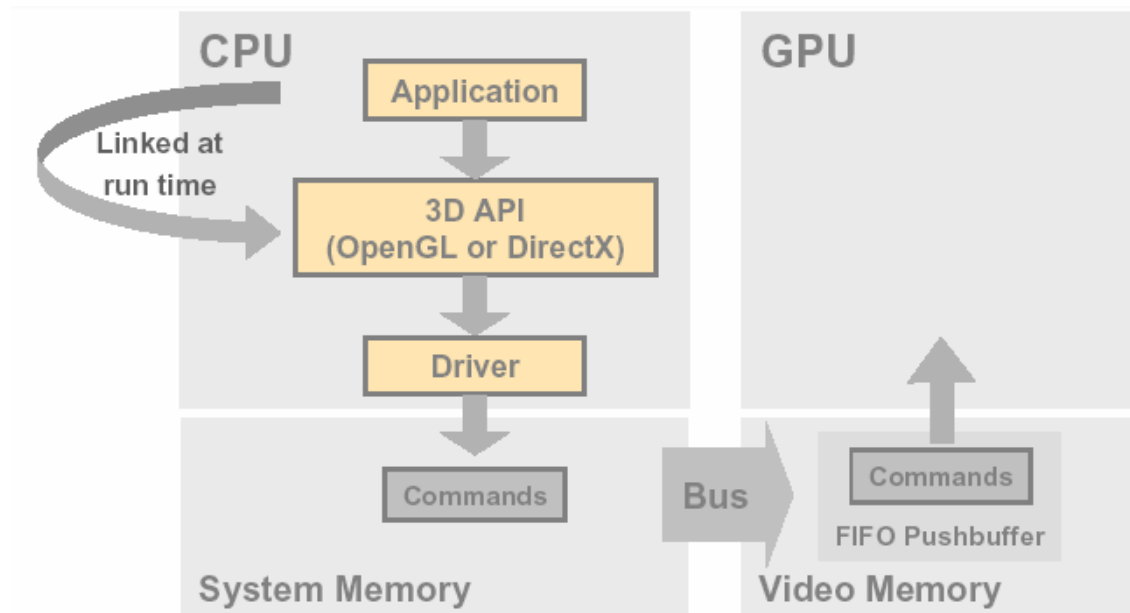
## Povijesni razvoj GPU-a

- profesionalno grafičko sklopovlje ~ razvoj zadnjih 30 godina
  - osobna računala
    - '95 tvrtka S3      kartica ViRGE, no naprednije mogućnosti spore
    - '96 tvrtka 3DFX    kartica Voodoo, 3D ubrzivačka kartica (nema 2D)
- do '99 sklopovski implementirane funkcije – postiže se velika brzina, no programirljivo sklopovlje (CPU) je fleksibilnije (ovisno o problemu u nekim slučajevima može biti brže)
- '99 važne grafičke funkcije sklopovski su podržane – GPU
  - '01 kartica GeForce3 podržava male **programe** u geometrijskoj fazi  
vrlo mali, jednostavne aritmetičke operacije (engl. vertex shader)
  - '02 programi za sjenčanje slikovnih elemenata, **floating point**  
dodaje se pristup teksturama (engl. pixel shader, fragment shader)  
još uvijek nema prave kontrole toka, postoje uvjetne naredbe ADDNZ  
ali ne i naredbe skoka JMP
  - '04 kartica GeForce6800 **kontrola toka** – naredbe skoka  
povećavanje broja cjevovoda

## Povijesni razvoj GPU-a - primjer

Generation	Year	Product Name	Process	Transistors	Antialiasing Fill Rate	Polygon Rate	Note
First	Late 1998	RIVA TNT	0.25 $\mu$	7 M	50 M	6 M	1
First	Early 1999	RIVA TNT2	0.22 $\mu$	9 M	75 M	9 M	2
Second	Late 1999	GeForce 256	0.22 $\mu$	23 M	120 M	15 M	3
Second	Early 2000	GeForce2	0.18 $\mu$	25 M	200 M	25 M	4
Third	Early 2001	GeForce3	0.15 $\mu$	57 M	800 M	30 M	5
Third	Early 2002	GeForce4 Ti	0.15 $\mu$	63 M	1200 M	60 M	6
Fourth	Early 2003	GeForce FX	0.13 $\mu$	125 M	2000 M	200 M	7





Primjer:

NVIDIA's – GeForce, nForce

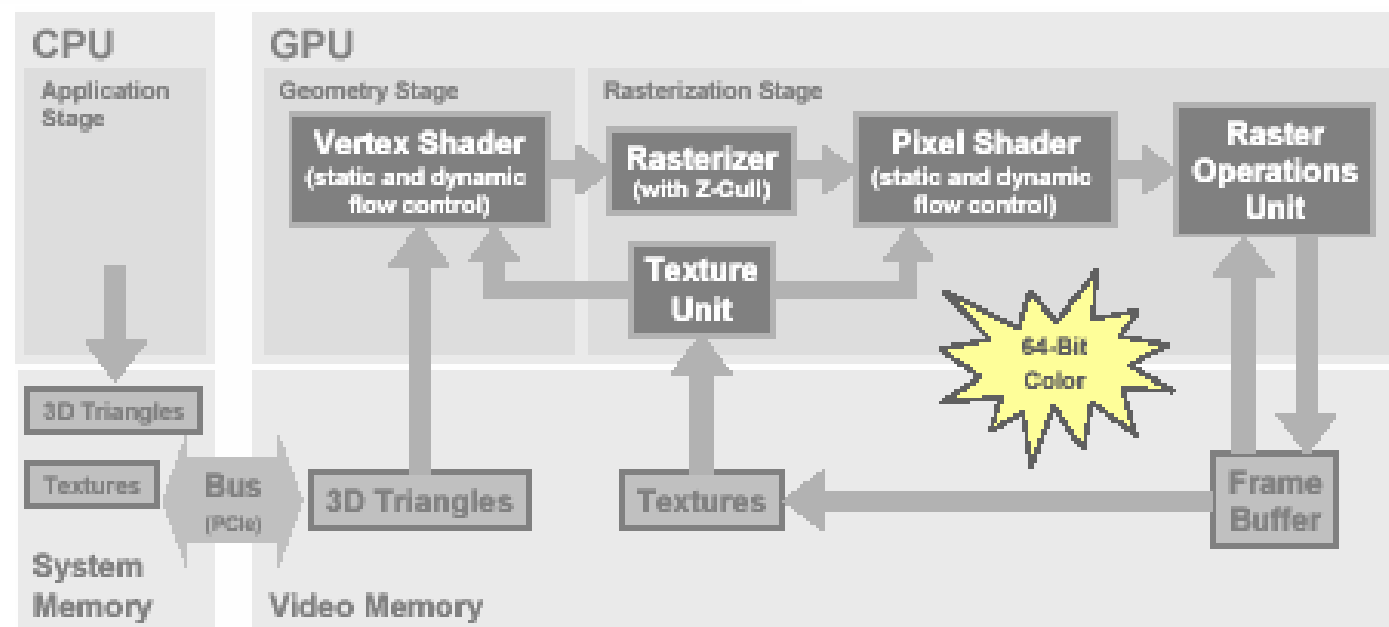
- Quadro

AMD(ATI) - Radeon

- FireGL

3Dlabs

Matrox





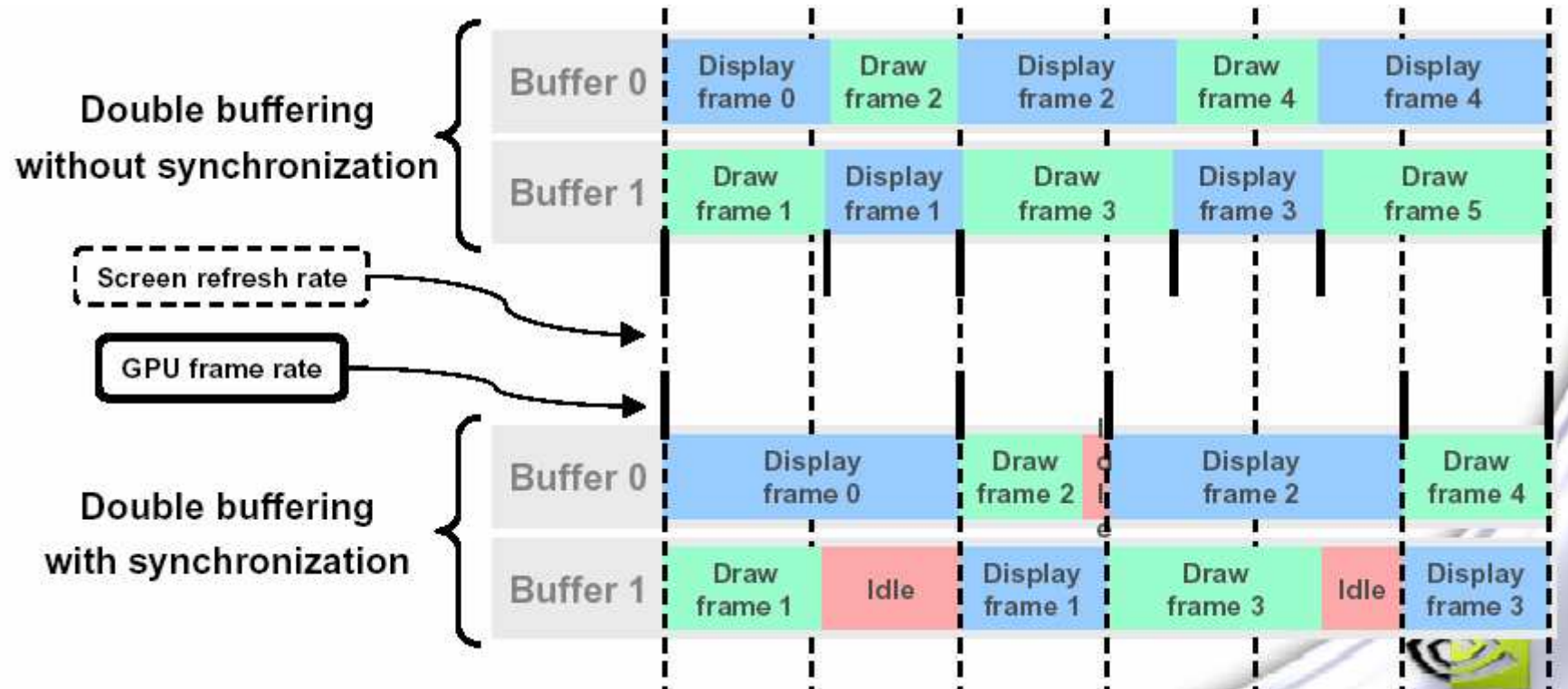
## Slikovna prikazna memorija (eng. frame buffer)

– memorija u koju se pohranjuje slika,  
iz te memorije se obavlja osvježavanje na zaslonu

- pohranjivanje slike `GL_COLOR_BUFFER`
- udaljenost od očišta `GL_DEPTH_BUFFER,`
- [http://olli.informatik.uni-oldenburg.de/Grafiti3/grafitiNav/flow8/page8.html#Ref\\_ID179](http://olli.informatik.uni-oldenburg.de/Grafiti3/grafitiNav/flow8/page8.html#Ref_ID179)
- dvostruki spremnik `GL_DOUBLE_BUFFER, GL_STEREO`
- <http://www1.ics.uci.edu/~frost/unex/JavaGraphics/course/Double.html>
- spremnik maske `GL_STENCIL_BUFFER`
- kombiniranje slike iz niza slika `GL_ACCUM_BUFFER`  
`GL_AUX_BUFFERS`

- spremnik teksture `GL_TEXTURE_1D`
  - `GL_TEXTURE_2D`
  - `GL_TEXTURE_3D`
  - (6 tekstura na kocki) `GL_TEXTURE_CUBE_MAP`
- posebna funkcija i spremnik za brisanje drugih spremnika – brzo
- određivanje broja bita u spremniku
- logičke operacije, operacije usporedbe, akumulacije/stapanja, antialias

## Sinkronizacija rada dvostrukog spremnika (engl. Double buffer)



<http://www.developer.com/repository/softwaredev/content/article/2000/06/20/SDtravisdblbuf/test1a.html>

## Upotreba spremnika – OpenGL

`glutInitDisplayMode (GLUT_RGB | GLUT_DOUBLE | GLUT_DEPTH);`

- inicijalizacija spremnika

- `GLUT_RGB` - spremnik boje

- `GLUT_DOUBLE` - dvostruki spremnik i osvježava se sa `glutSwapBuffers()`;  
inače ako ne koristimo dvostruki spremnik imamo `glFlush()`;

- `GLUT_DEPTH` - spremnik udaljenosti i nakon stvaranja prozora  
`window = glutCreateWindow ("Prozor");`

potrebno je omogućiti spremnik `glEnable(GL_DEPTH_TEST);`

- brisanje spremnika i zaslona

`glClear (GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);`

## 2.1.2 Izlazne grafičke naprave

- podjela izlaznih grafičkih naprava

jedinice za prikaz objekata (CRT, LCD, s plazmom, pisači, crtala)

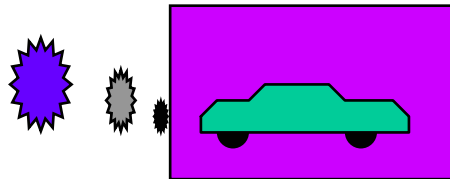
- vektorske
- rasterske <http://www.cs.unc.edu/~mcmillan/comp136/Lecture1/disptech.html>
- emitirajuće (CRT, s plazmom, OLED organske diode)
- ne emitirajuće (LCD - tekući kristali)
- osvježavajuće
- s pamćenjem
- jednobojne
- sivi klin
- višebojne

jedinice za izradu 3D objekata

- vektorski pristup (tokarilice, glodalice)
- sloj po sloj

## Usporedba vektorske i rasterske prikazne procesne jedinice

- Vektorska
  - nekadašnja izvedba
  - (ograničenost količine memorije)



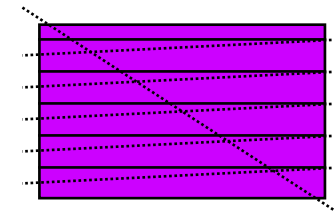
### prednosti

- točnost prikaza (ploteri)
- jednostavna promjena mjera

### nedostaci

- dugačka prikazna datoteka
  - popunjavanje poligona
- => problem osvježavanja

- Rasterska
  - danas uobičajeno



### prednosti

- veličina prikazne datoteke ne utječe na frekv. osvježavanja

### nedostaci

- potreba pretvorbe u diskretnu reprezentaciju
- => pogreška diskretizacije  
(eng. alias-sampling error  
nazubljene linije, moarè)

## Jedinice za prikaz (različite karakteristike)

- slika se pohranjuje u slikovnoj prikaznoj memoriji
- iz memorije podaci se prenose preko DAC do zaslona puno puta u sekundi
- važna je *brzina* osvježavanja zbog eksponencijalnog slabljenja intenziteta svjetla koje emitira fosfor, više kvantnih razina:
  - florescencija                      dio  $\mu\text{s}$  (snop uključen)
  - fosforescencija                    10-60  $\mu\text{s}$  (snop isključen)

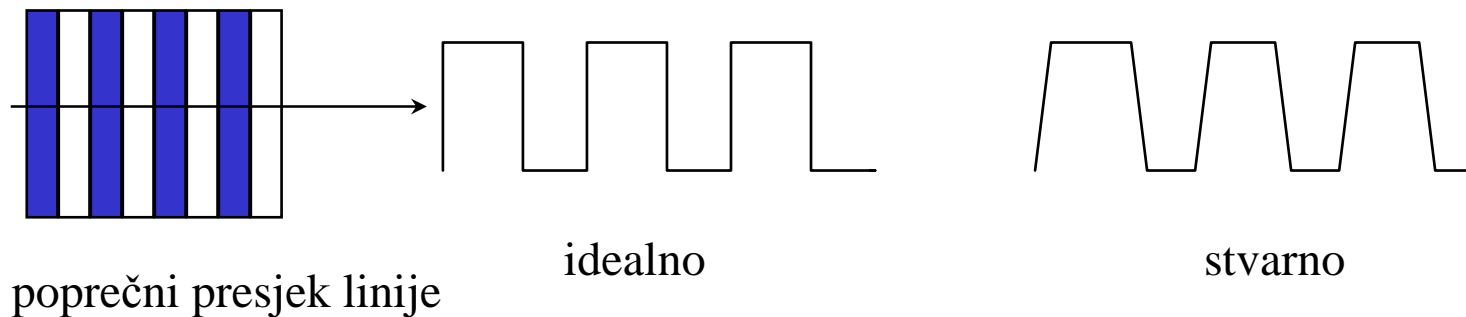
Visoka *perzistencija* znači da svjetlu treba dugo da oslabi (manje od 10% maksimalne vrijednosti), te se tada može sporije osvježavati

<= kontradiktorni zahtjev => [http://www.colorado.edu/physics/2000/tv/black\\_and\\_white.html](http://www.colorado.edu/physics/2000/tv/black_and_white.html)

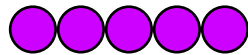
brzina animacije traži brži fosfor

- *dijagonala*
  - nazivna dijagonala 17''
  - vidljiva dijagonala 15,6-16,2''

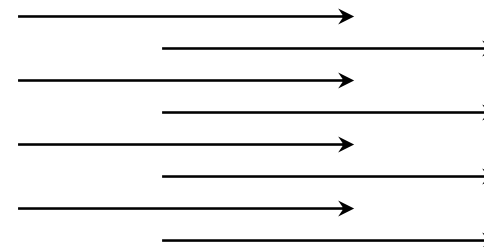
- *frekvencije osvježavanja*
  - vertikalna frekvencija (broj slika u sekundi) 60-160 Hz (85 Hz propisano VESA standardom)
  - horizontalna frekvencija (broj linija u sekundi) 30-100 kHz
  - frekvencija osvježavanja slikovnih elemenata (brzina paljenja i gašenja elektronskog snopa) 50-160 MHz - širina pojasa (engl. pixel rate) [http://www.colorado.edu/physics/2000/tv/moving\\_electrons.html](http://www.colorado.edu/physics/2000/tv/moving_electrons.html)
- *geometrijska svojstva*
  - kada prikazujemo kružnicu želimo da nema oblik elipse
- *razlučivost, zrnatost, rezolucija*
  - broj crnih/bijelih linija koje se mogu prikazati i odvojene su (obično se izražava po jedinici udaljenosti dpi)



- *razmak*
  - između točaka iste boje (eng. dot pitch) dijagonalno razmak između točaka 0,25-0,28 mm, razmak rupica na sitastoj maski
  - između pruga (eng. stripe pitch) horizontala udaljenost 0,21-0,28 mm, razmak na aperturnoj rešetki
- *veličina* jedne točke koja može biti načinjena (eng. dot, spot size)
- *adresibilnost* broj individualnih točaka (po inču) koji može biti načinjen - obrnuto proporcionalno udaljenosti
  - poželjno je da veličina točke bude veća od udaljenosti središta



- iscrtavanje s *preplitanjem*
    - (engl. interlaced /non interlaced)
- ako sporije iscrtavamo  
možemo iscrtati veću sliku



paran prolaz

neparan prolaz



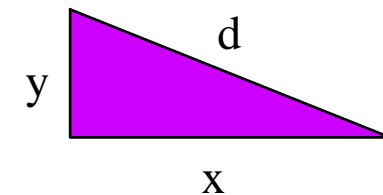
- *toplina boje* - spektar zračenja (eng. color temperature)
  - kada se crno tijelo zagrije na temperaturu 9300K ima identično zračenje monitoru
- *degauss*
  - uklanjanje statičkog naboja

- Različite karakteristike fosfora, DAC, elektronskog topa, sitaste maske, brzine i organizacije memorije utječu na konačne mogućnosti.

NPR:

vertikalna frekvencija - 76 Hz, NI  
 razlučivost - 1152x900  
 razmak pruga - 0,26 mm = 0,0103''

dijagonala  $d = 20''$   
 tipičan omjer slike  $\underline{y/x = 0,75}$



$$\Rightarrow 20^2 = x^2 + 0,75^2 x^2 \Rightarrow x = 16'' \Rightarrow 1550 \text{ slikovnih el.}$$

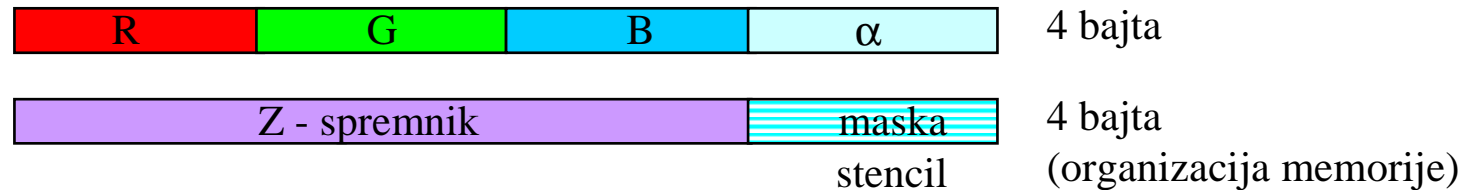
$$\Rightarrow \text{horizontalna frekvencija } 900 \times 76\text{Hz} + 10\% \quad \sim 76 \text{ kHz}$$

$$\Rightarrow \text{paljenje/gašenje elektronskog snopa} \\ 1152 \times 900 \times 76 + 30\% \quad \sim 100 \text{ MHz}$$

$$\Rightarrow \text{postavlja zahtjeve na brzinu D/A pretvorbe 3x8b} \\ \text{i vrijeme pristupa memoriji} \quad \sim 10 \text{ ns}$$

## Pojasna propusnost prema memoriji (engl. memory bandwidth)

NPR:



2 – pristupa (piši/čitaj) = 16 bajta

$$1280 \times 1024 \times 16 \text{ bajta} \times 60 \text{ fps} = 1,26 \text{ GB/sec.}$$

dubinska složenost (engl. depth complexity, engl. overdraw)

$$1280 \times 1024 \times 16 \text{ bajta} \times 60 \text{ fps} \times 3 = 3,78 \text{ GB/sec.}$$

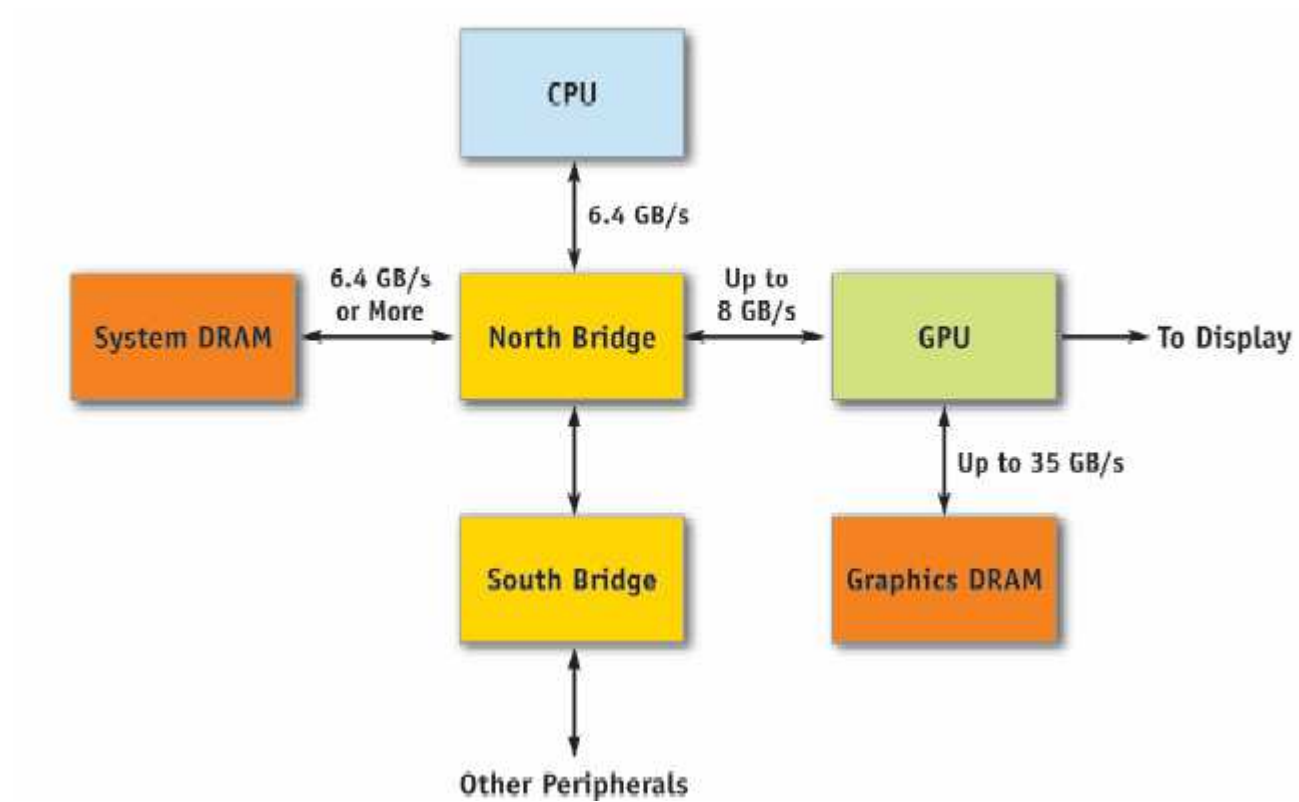
prikaz teksture – trilinearna interpolacija (8 vrhova  $\times$  4 bajta)

$$1280 \times 1024 \times (16 + 32) \text{ bajta} \times 60 \text{ fps} \times 3 = 11,32 \text{ GB/sec.}$$

antialias  $\times$  4 (engl. FSAA Full Screen Antialiasing)

$$1280 \times 1024 \times (16 + 32) \text{ bajta} \times 60 \text{ fps} \times 3 \times 4 = 45,3 \text{ GB/sec.}$$

Primjer: Pojasna propusnost prema memoriji (engl. memory bandwidth)



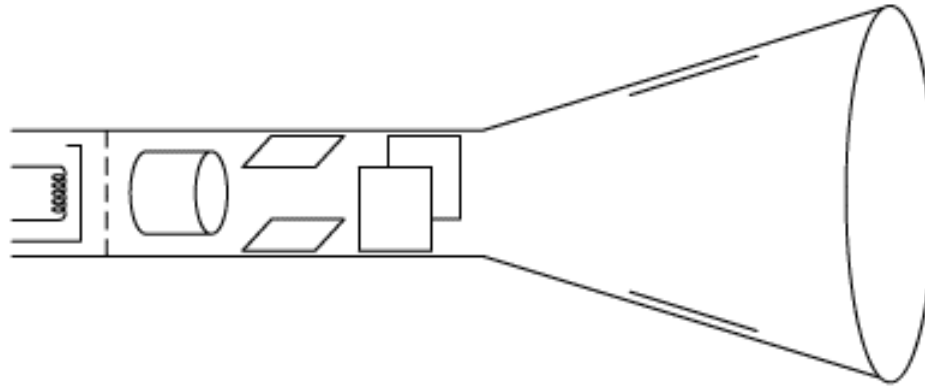
- rapoložive *memorije* ( 256 KB 1971 = 2 mil \$)
  - SDRAM, (interno paralelna organiz.) ~ 60 ns sljedeći ~10ns
  - SGRAM (synchronous graphics RAM, ima dodatne grafičke mogućnosti, može biti i dvopristupni)
  - DRAM (engl. dynamic)
  - VRAM (engl. dual port),
  - EDO RAM
  - RAMBUS
  - WRAM (engl. window)

DDR (engl. Double-Data-Rate) za neke od navedenih memorija postoji mogućnost (GDDR SDRAM ~ 1 ns)

- ostvarivanje potrebnog vremena pristupa (brzine)
  - FPM (eng. fast page mod) prisutna je adresa retka, potrebno je mijenjati samo adrese stupaca
  - paralelne organizacije, dohvaćanje u brze posmačne registre

## JEDINICE ZA PRIKAZ

- CRT (princip rada)
  - u elektronskom topu *žarna nit* grije *katodu* koja emitira snop elektrona
  - *kontrolna mrežica* određuje količinu elektrona koja će proći dalje i na taj način određuje svjetlinu
  - sustav za *fokusiranje* elektronskog snopa dinamički fokusira snop ovisno o položaju na zaslonu (defokusiran-mutna slika) teži se ravnom zaslonu (horizontalno, vertikalno)
  - horizontalni i vertikalni *otklonski sustav* otklanjaju snop
  - visoko pozitivna *metalizacija* (anoda) 15.000-20.000V ubrzava elektrone
  - sitasta *maska* ili aperturna rešetka



- *fosfor* naparen na staklo - prelazak u više kvantno energetska stanje a prilikom povratka elektrona emitira se energija u obliku *svjetla* određene valne duljine r, g, b  
obično postoje razlike u fosforu tako da ista slika izgleda različito na različitim monitorima
- miješanje valnih duljina => *oko* čovjeka
- utjecaj ambijentnog svjetla na svjetlinu i kontrast  
[http://www.sandlotscience.com/Contrast/Checker\\_Board\\_2.htm](http://www.sandlotscience.com/Contrast/Checker_Board_2.htm)
- kalibriranje boja
- na elektronski snop (elektro) magnetska polja imaju utjecaj

## CRT - FST (Flat Square Tube) s ravnom cijevi

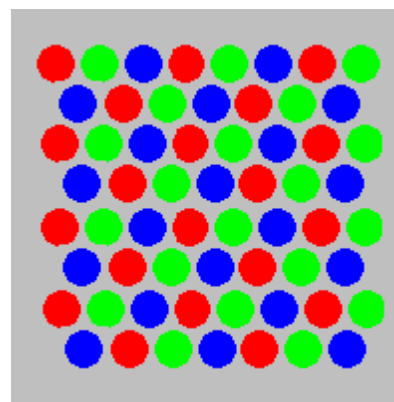
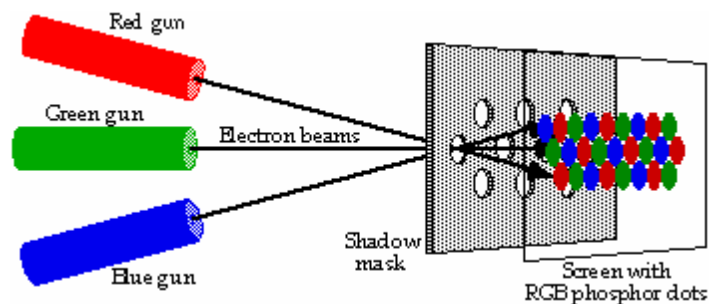
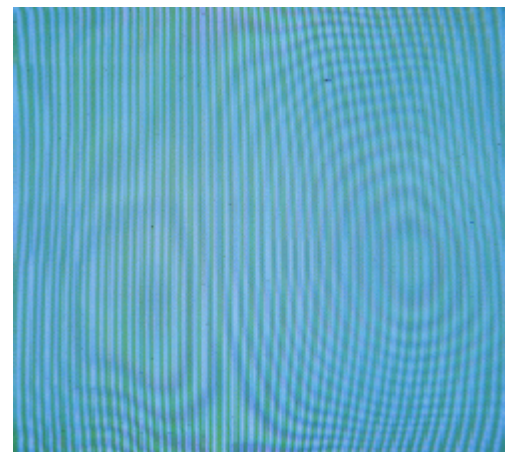
sitasta maska (eng. shadow mask)

- delta raspored fosfornih cijevi (topova)
- u nizu (eng. in line)
- sitasta maska se radi od legure invar, problem grijanja i naprezanja
- problem je vrlo preciznog fokusiranja snopa na pripadni fosfor

(neujednačena slika po površini zaslona, crveno-zeleni tragovi, moarè)

- veliki dio površine je zaklonjen maskom (~20% elektrona pogodi fosfor) pa je smanjena je svjetlina

[http://www.colorado.edu/physics/2000/tv/merging\\_color.html](http://www.colorado.edu/physics/2000/tv/merging_color.html)





## CRT - trinitron cijev

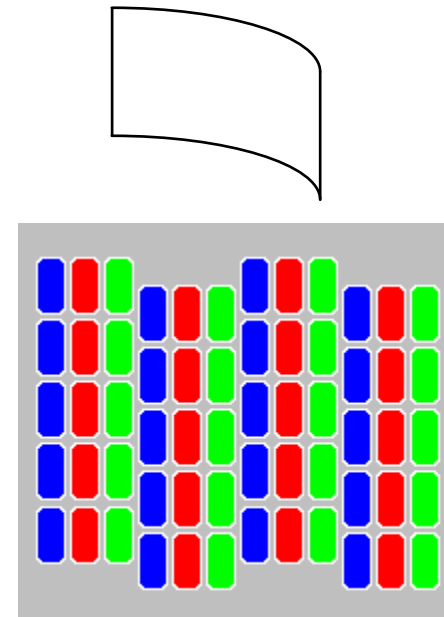
sa aperturnom rešetkom (eng. aperture grill)

- niz vertikalnih traka (žica)
- zauzimaju manju površinu pa je slika svjetlija, kontrasnija, vjernije boje
- prilikom rada rešetka se ugrije, pa se javlja problem deformacija i vibracija - dodaju se dvije žice od volframa za učvršćivanje (obično su teži zbog problema učvršćivanja, osjetljiviji na transport)
- horizontalna zakrivljenost
- 30-50% skuplji

Sony - 64 - koristio cijev chromatron za prvi TV u boji, ima niz vertikalnih žica

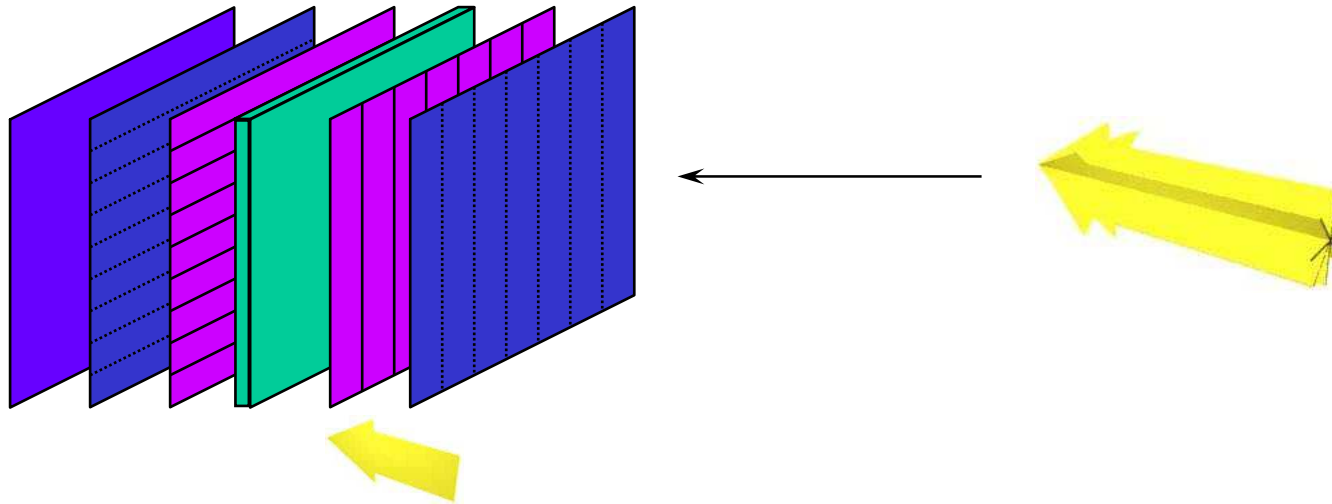
- 67 - načinjena cijev s tri elektronska topa i jednim sustavom za fokusiranje







- CRT - kombinacija prethodnih
  - NEC, KFC



## LCD prikazna jedinica s tekućim kristalima

- <http://www.colorado.edu/physics/2000/polarization/polarizationI.html>
- <http://www.colorado.edu/physics/2000/laptops/index.html>



- šest slojeva :
- |                           |   |
|---------------------------|---|
| reflektirajući sloj       |    |
| horizontalna polarizacija |  |
| horizontalne žičice       |  |
| sloj tekućih kristala     |  |
| vertikalne žičice         |  |
| vertikalna polarizacija   |  |

## princip rada

- materijal tekućih kristala je načinjen od dugačkih molekula
  - kada je kristal u *električnom polju* nema polarizirajuća svojstva na svjetlo koje dolazi, pa svjetlo ostaje vertikalno polarizirano i *ne prolazi* kroz horizontalnu polarizaciju
  - kada je kristal *nije* u električnom polju *zакreće* ravninu polarizacije za 90° iz vertikalne u horizontalnu
- <http://www.colorado.edu/physics/2000/laptops/index.html#demo>
- [http://www.colorado.edu/physics/2000/polarization/molecular\\_view.html](http://www.colorado.edu/physics/2000/polarization/molecular_view.html)
- TFT - (eng. thin film transistor) na svakom (x, y) ima tranzistor, služe kao aktivna memorija dok se stanje ne promijeni

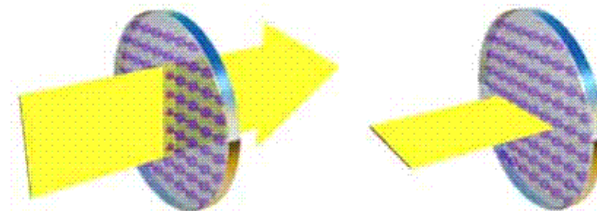
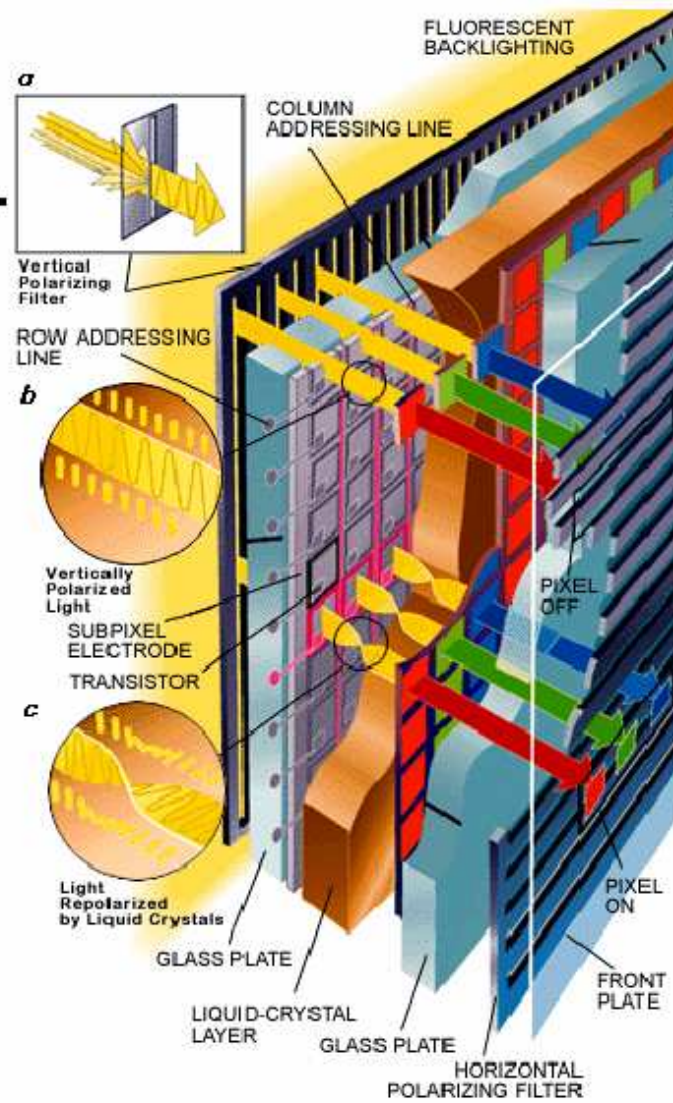
prednosti - lagani, mala potrošnja, mali po z-osi,

nedostaci - nisu izvor svjetlosti no može se koristiti stražnje osvjetljenje za projekcije, spora promjena slike, kut gledanja je ograničen, osjetljivi na pritisak i visoku temperaturu

upotreba

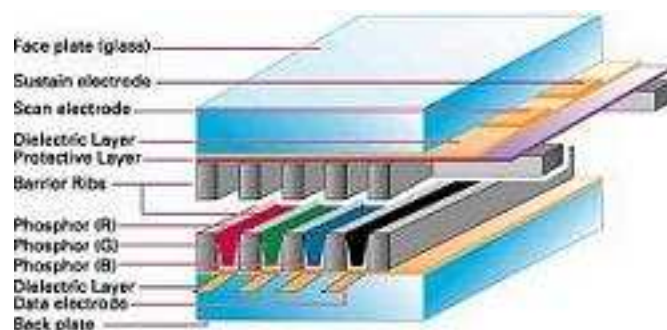
- prijenosna računala
- projektori [http://www.colorado.edu/physics/2000/laptops/laptop\\_screen.html](http://www.colorado.edu/physics/2000/laptops/laptop_screen.html)
- HMD

# LCDs



## Prikazna jedinica s plazmom

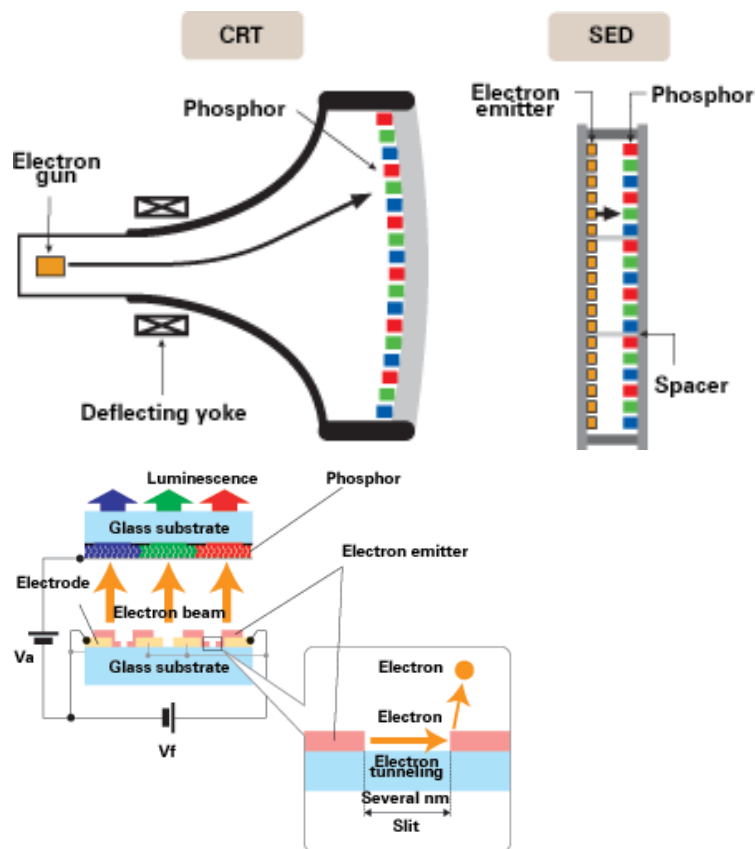
- kod CRT prikaznih jedinica velika je dubina po z - osi i tehnološki je ograničena veličina
- LCD prikazne jedinice nisu izvor svjetlosti
- na mjestu ukrštanja elektroda je adresirano mjesto zatim dolazi do ionizacije xenon/neon (xenon/neon XeNe) plina, to izaziva ultravioletno zračenje koje aktivira fosfor - svjetlo (nije pasivni uređaj)
- jedinice s plazmom mogu imati veličinu ~ 40'', 61'' (-100'')
- nedostatci – veliki slikovni elementi (1 mm, CRT 0,2 mm),  
vakuuum u malim fluorescentnim cijevima – deblje staklo  
velika potrošnja (40'' ~ 300W) uz slabu svjetlinu (~ 1/3 CRT),



Nove tehnologije:

- OLED (engl. Organic Light-Emitting Diode Arrays)
- DMD / DLP Digital Micromirror Devices /Digital Light Processing
- SED (engl. Surface-conduction Electronemitter Display)

3D PRIKAZ



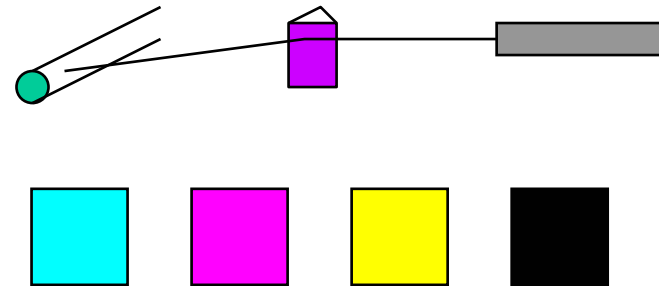
*OLED Display Screen (from Universal Display Corp.)*



# Jedinice za izradu 3D objekata

po uzoru na pisače (printeri)

- matrični
- laserski
- ink-jet
- termo
  - CMYK (više prolaza)



- izrada objekata sloj po sloj
    - 3D pisači (ZPrinter)
    - uređaji za stereolitografiju
- (važno za brzu izradu prototipa)



- po uzoru na crtala (ploteri)
  - s pisaljkom (PEN)
    - optimiranje praznog hoda, akceleracija
    - prikaz karata
  - elektrostatski
    - negativno nabijeni papir, pozitivno nabijena tinta
    - brži, manje kontrastni od crtala s pisaljkom
- TOKARILICE, GLODALICE
  - izrada trodimenzijskih objekata



### 2.2.3 Ulazne grafičke naprave

- tablica (engl. tablet) s pisaljkom, na dodir osjetljiva ploča
  - kapacitivna sprega, elektromagnetska, zvučna, naponski gradijent
- miš
  - mehanički, mehaničko optički, optički
- 3D zvučno pero, svjetlosno pero
- sustavi za 3D uzorkovanje
  - ultrazvučno
  - CT, PET (računalna tomografija)
  - MR (magnetska rezonancija)
  - laserskim snopom - mogućnost uzorkovanja boje i temperature

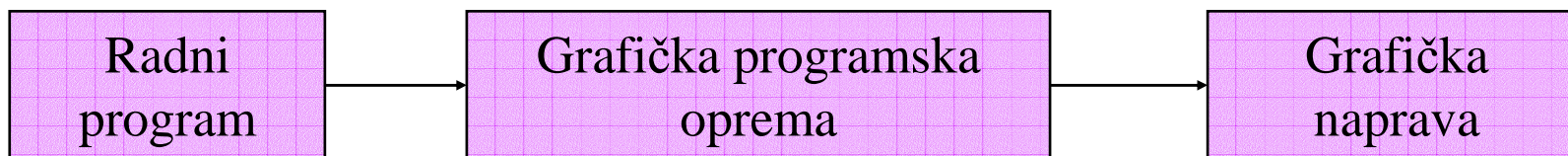
<http://www.cyberware.com/>

Primjeri objekata:

[http://www.cc.gatech.edu/projects/large\\_models/](http://www.cc.gatech.edu/projects/large_models/)



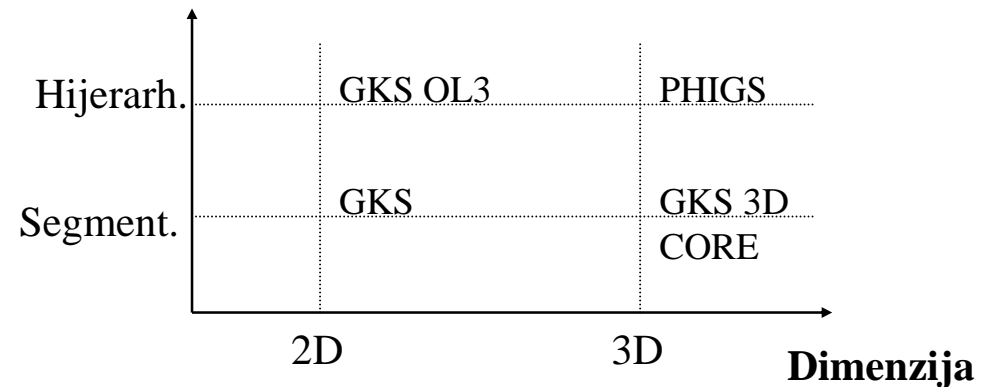
## 2.2 PROGRAMSKA GRAFIČKA OPREMA



- Knjižnica grafičkih rutina (grafičke rutine koje se pozivaju iz nekog višeg programskog jezika s atributima C, C++). Teži se da ova knjižnica bude načinjena prema specifikaciji API-a, tj. prema nekom standardu.
  - “+” neovisnost radnog programa o sklopovskoj opremi
  - “-” obično se ne može ostvariti potpuna iskorištenost sklopovske opreme
- Standardima je propisano
  - API - prema kojima se načine grafičke biblioteke
    - OpenGL-Mesa, DirectX Direct3D, Phigs-PEX
  - zapisi
    - slika TIF, GIF, BMP, JPEG, HPGL, PS (rasterski, vektorski)
    - niza slika GIF, video AVI, MOV, WMV, MPG, MP4, SWF, RM
    - scene, objekti DXF, MAX, 3DS, WRL-vrml, PLY, OBJ

- Grafičke jezgre načinjene u okviru standarda
  - 3D CORE (Core Graphics System)
    - 1979. ACM SIGGRAPH (Association for Computing Machinery Special Interest Group on Graphics)
  - GKS (Graphics Kernel System)
    - ISO 88, 94, 97, 98, 99 (International Standards Organization)
    - ANSI 85 (American National Standards Institute)
  - PHIGS (Programmer's Hierarchical Interactive Graphics System), PHIGS+ za pseudorealističan prikaz
    - ISO 90, 97
    - ANSI 88
  - VRML (Virtual Reality Modelling Language)
    - ISO 97, 98, 99

**Strukt. slike**



- Osim službenih standarda postoje “de facto” ili industrijski standardi
  - 93’ GL, OpenGL SGI
  - 95’ Direct 3D Microsoft
  - X Window System’s Xlib MIT PEX
  - RenderMan Pixar
  - PostScript Adobe
  - OpenFlight

Komercijalno su ovi standardi značajniji od službenih standarda jer se jednostavnije mogu mijenjati.

### OpenGL - SIGGRAPH

- 2001. OpenML – integracija i sinkronizacija 3D grafike s video i audio zapisima (Media - rich programming, Khronos group)
- 2003. OpenES – podrška za ugrađene sustave (embedded 3D graphics)
- 2004. OpenGL 2.0

### Jezici i tehnologije za paralelno programiranje

Brook – programski jezik – programiranje tokova (stream program)

- <http://graphics.stanford.edu/projects/brookgpu/index.html>

CUDA paralelno obavljanje operacija na različitim podacima

- <http://developer.nvidia.com/object/cuda.html>

## Jezici za sjenčanje (engl. shading languages)

- programiranje grafičkog sklopovlja korištenjem jezika više razine (kako ne bi morali programirati u asembleru za karticu)

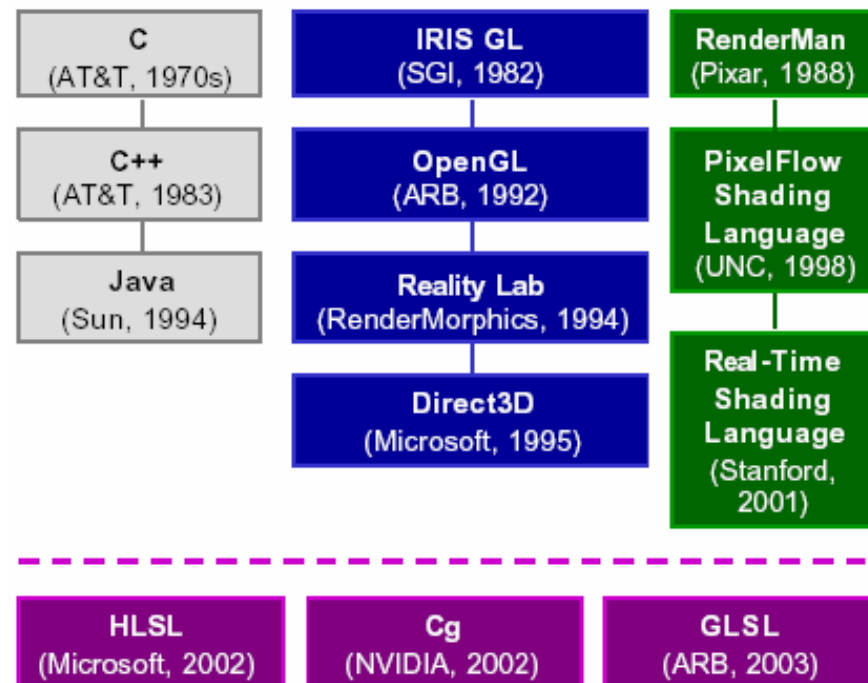
HLSL (engl. High-Level Shading Languages) – Direct3D, Microsoft, '02.

CG (engl. C for graphics) – OpenGL, Direct3D, NVidia, '02.

GLSL (engl. The OpenGL Shading Language) – open standard, ARB, '03.

- Alati** za izradu programa u SL FX-composer (NVidia – HLSL)

[http://developer.nvidia.com/object/fx\\_composer\\_home.html](http://developer.nvidia.com/object/fx_composer_home.html)



## Usporedba jezika za sjenčanje (engl. Shading Languages):

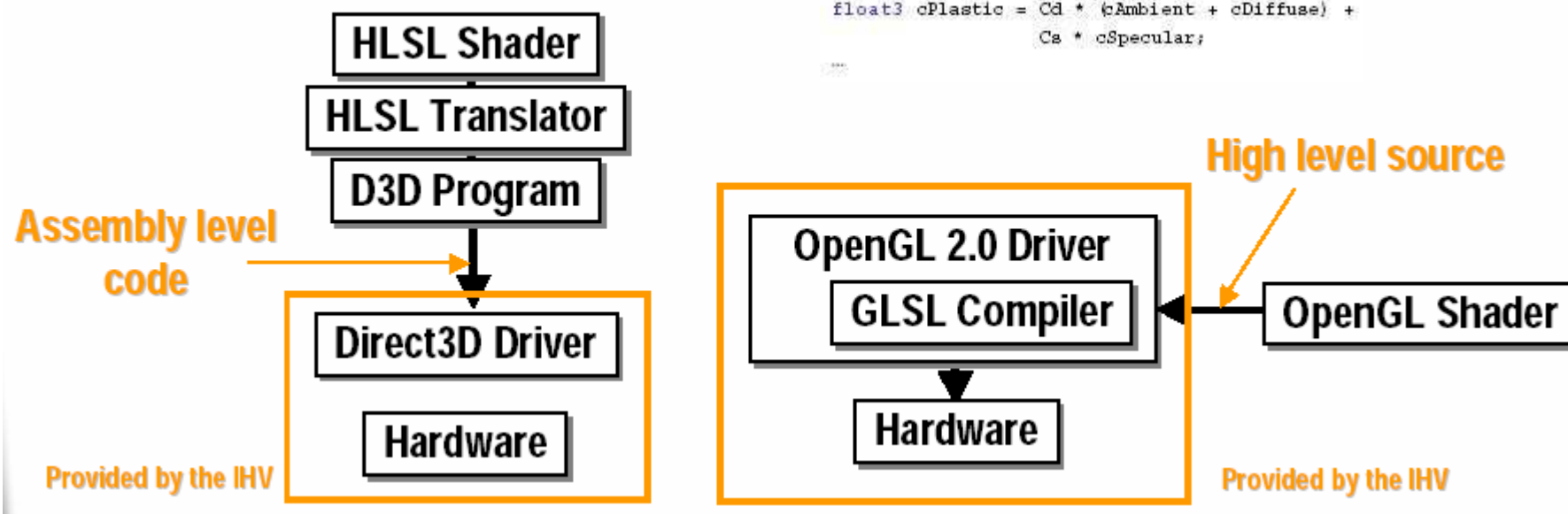
- HLSL Direct3D
- GLSL OpenGL

### Assembly

```
...
DP3 R0, c[11].xyzx, c[11].xyzx;
RSQ R0, R0.x;
MUL R0, R0.x, c[11].xyzx;
MOV R1, c[3];
MUL R1, R1.x, c[0].xyzx;
DP3 R2, R1.xyzx, R1.xyzx;
RSQ R2, R2.x;
MUL R1, R2.x, R1.xyzx;
ADD R2, R0.xyzx, R1.xyzx;
DP3 R3, R2.xyzx, R2.xyzx;
RSQ R3, R3.x;
MUL R2, R3.x, R2.xyzx;
DP3 R2, R1.xyzx, R2.xyzx;
MAX R2, c[3].z, R2.x;
MOV R2.z, c[3].y;
MOV R2.w, c[3].y;
LIT R2, R2;
...
```

### High-Level Language

```
...
float3 cSpecular = pow(max(0, dot(Nf, H)),
    phongExp).xxx;
float3 cPlastic = Cd * (cAmbient + cDiffuse) +
    Cs * cSpecular;
...
```



## OpenGL (engl. Open Graphics Library)

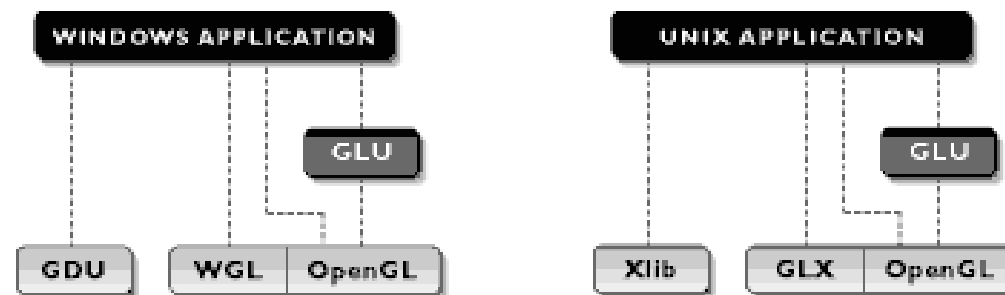
<http://www.opengl.org/>

[state.pdf](#)

- IrisGL - SGI temelj za OpenGL, 1992. industrijski standard
- engl. state machine - koji kontrolira skup specifičnih operacija crtanja 2D/3D (definira kontekst za prikaz)
- OpenGL se temelji na FrameBuffer-u no u svom konceptu ne podržava grafičke ulazno izlazne naprave kao što su miš ili tipkovnica
- programsko sučelje prema grafičkom sklopovlju, neovisan o platformi tj. o OS-u i grafičkom sučelju (engl. window system)

### Dodatne biblioteke

- rad u prozoru
  - WGL - Microsoft Windows,
  - GLX - X Windows system, na Unix-u
  - AGL - Apple Macintosh



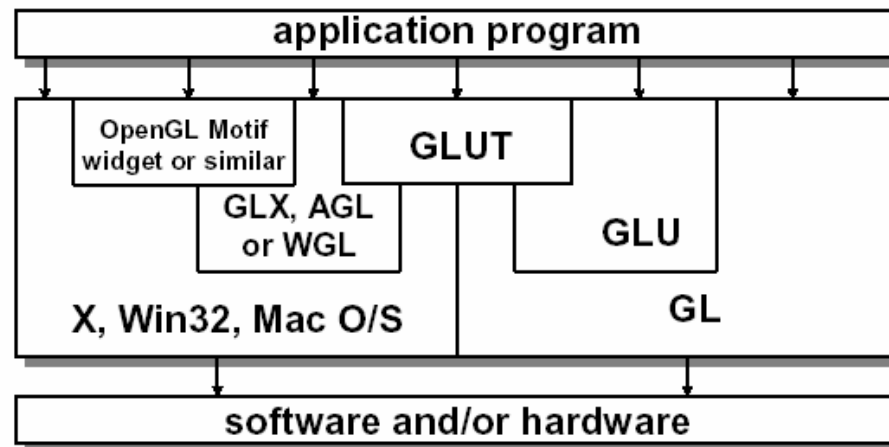
- GLU Utility Library (pomaže u modeliranju i nekim operacijama s prozorima)
  - objekti (kugla, cilindar, čajnik)
  - dijeljenje poligona, NURBS
- GLUT (OpenGL Utility Toolkit, truly cross-platform)
  - neslužbeni dio OpenGL, pisanje prenosivih aplikacija koje rade u okruženju prozora
  - pojednostavljuje stvaranje prozora, rukovanje događajima (engl. events)

```
#include <GL/gl.h>
```

```
#include <GL/glu.h>
```

```
#include <GL/glut.h>      // uključuje gl.h i glu.h pa ih nije potrebno navoditi
```

- GLEW (OpenGL Extension Wrangler Library) <http://glew.sourceforge.net/>
  - olakšava korištenje OpenGL ekstenzija
- OpenGL Performer – olakšava pisanje složenijih aplikacija

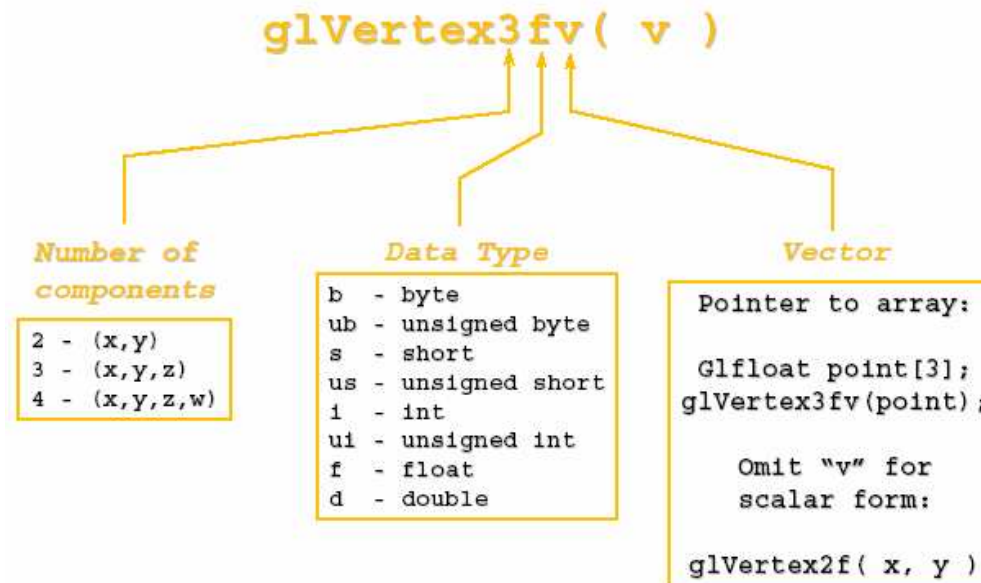




## OpenGL

[shapes.exe](#)

- definirani su osnovni tipovi podataka zbog prenosivosti
  - GLfloat, GLint ...
- primjer naredbe u OpenGL-u



## Programčić koji crta kvadrat:

```
#include <GL/glut.h>
```

```
void crtaj() {  
    glClear(GL_COLOR_BUFFER_BIT);  
    glLoadIdentity();  
    glOrtho(0, 4, 0, 4, -1, 1);  
    glBegin(GL_POLYGON);  
        glVertex2i(1, 1);  
        glVertex2i(3, 1);  
        glColor3f(0.5, 0, 0.5);  
        glVertex2i(3, 3);  
        glVertex2i(1, 3);  
    glEnd();  
    glFlush();  
}
```

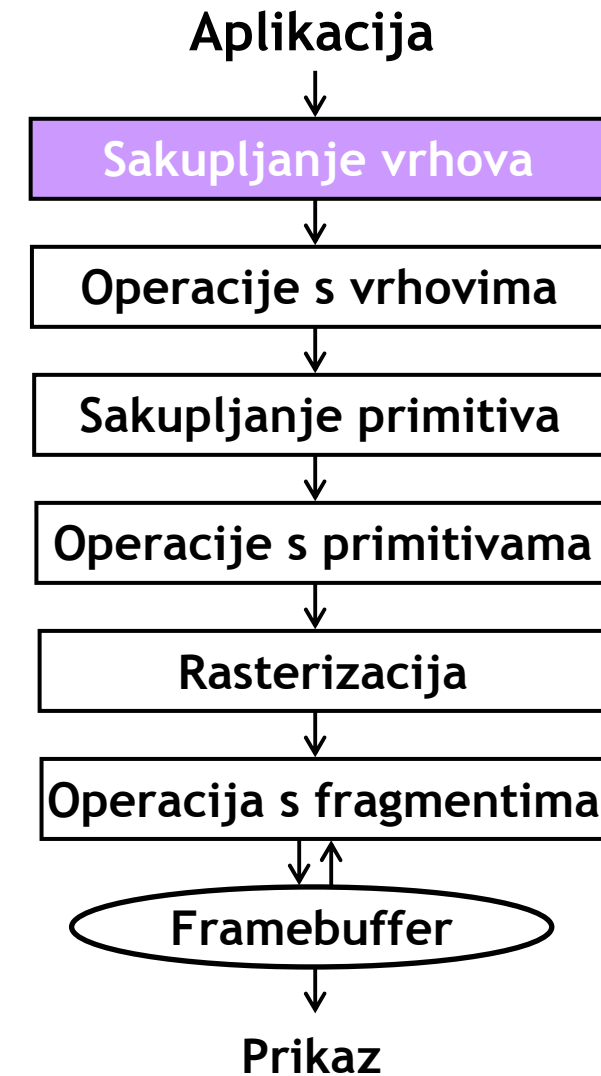
```
int main(int argc, char** argv) {  
    glutInit(&argc, argv);  
    glutInitDisplayMode (GLUT_RGBA);  
    glutCreateWindow ("kvadrat");  
    glutDisplayFunc (crtaj);  
    glutMainLoop();  
}
```

konverzija u internu  
reprezentaciju npr:

`glVertex2i(3, 3);` 1 u float  
z, w - inicijalizira na 0 1  
postavlja stanje npr. boju

interna reprezentacija:

```
struct {  
    float x,y,z,w; // 3, 3, 0, 1  
    float r,g,b,a; // 0.5, 0, 0.5, 1  
} vertex;
```



## Operacije s vrhovima (procesor vrhova):

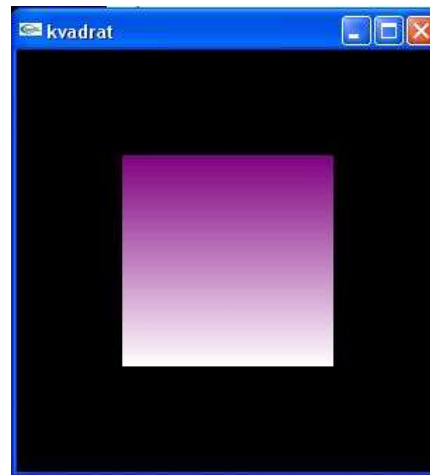
- transformiranje vrhova  
(množenje transformacijskom matricom)
- proračun osvjetljenja u vrhovima
- proračun koordinata tekstone
- ...

skalira koordinate  
obzirom na veličinu prozora

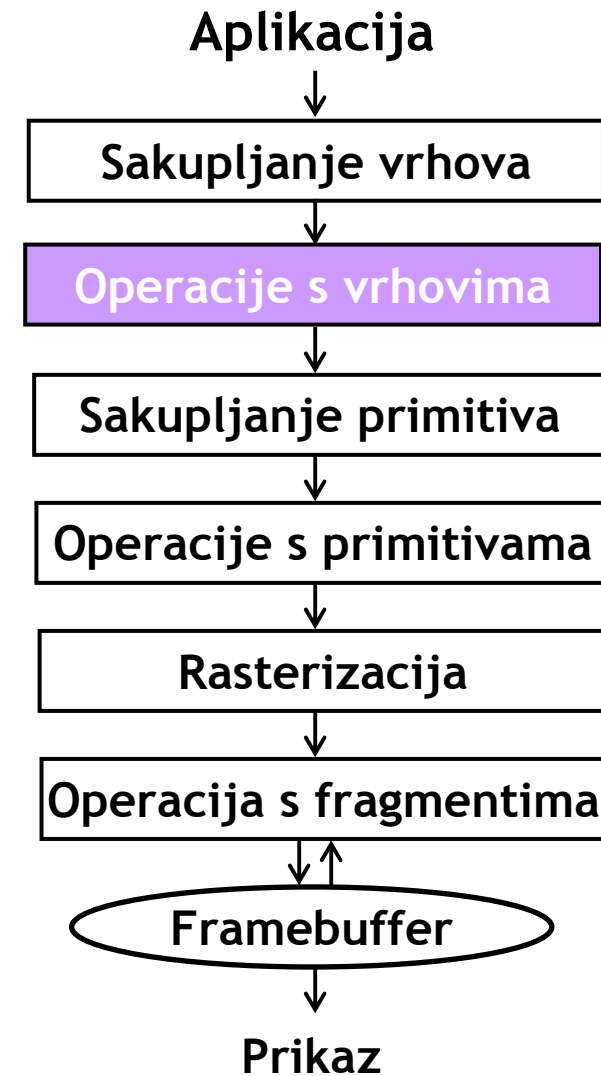


### U našem slučaju:

- skaliranje koordinata na veličinu prozora  
`glLoadIdentity();`  
`glOrtho(0, 4, 0, 4, -1, 1);`
- veličine koje nismo odredili  
npr. dimenzije prozora su predefiniране  
(default npr. 300×300)



Ž. M, ZEMRIS, FER



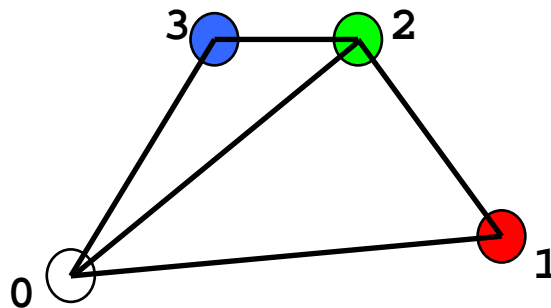
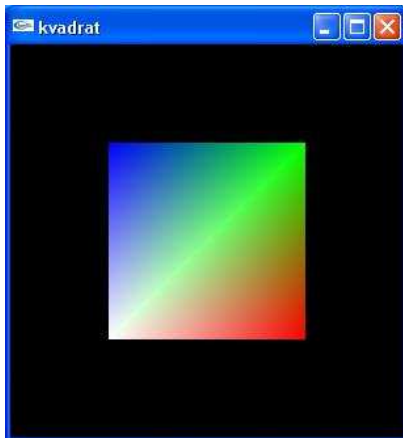
## Sakupljanje primitiva:

- povezivanje vrhova primitivama
- primitive su točke, linije, trokuti, niz trokuta, ...

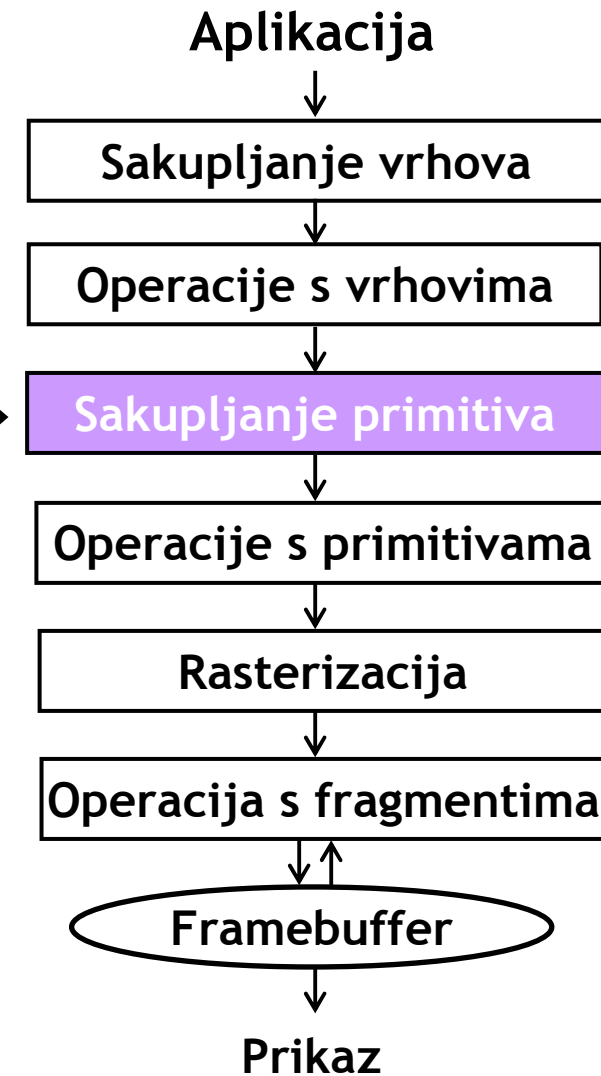
```
glBegin(GL_POLYGON);  
glColor3f(1, 1, 1);  
glVertex2i(1, 1);  
glColor3f(1, 0, 0);  
glVertex2i(3, 1);  
glColor3f(0, 1, 0);  
glVertex2i(3, 3);  
glColor3f(0, 0, 1);  
glVertex2i(1, 3);  
glEnd();
```

### U našem slučaju:

- poligon se rastavlja na niz trokuta



rastavljanje poligona  
na niz trokuta



## Operacije s primitivama:

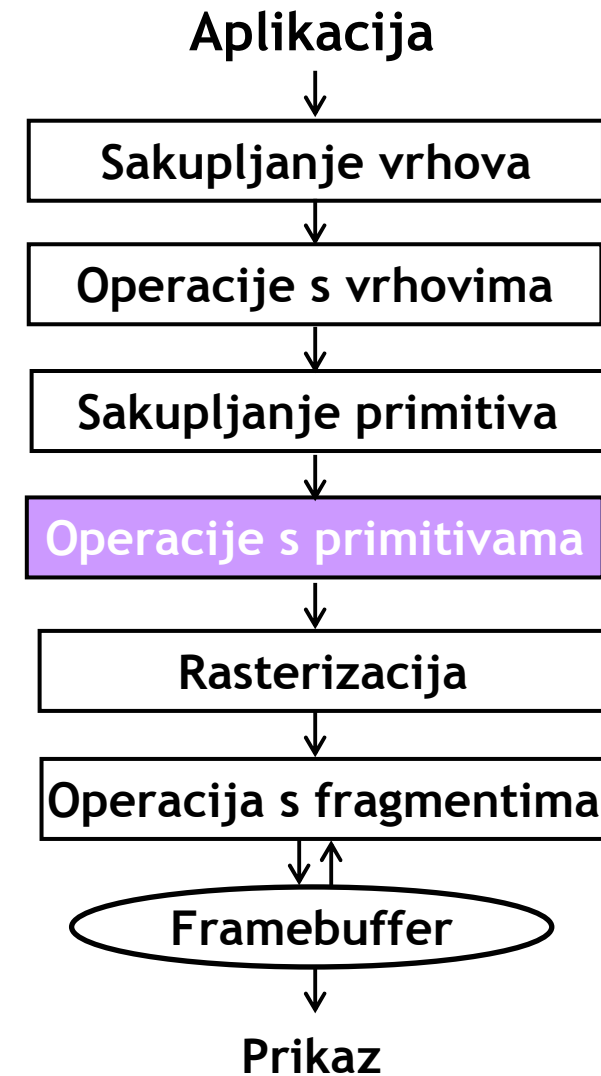
- odsijecanje obzirom na prozor  
odnosno prema piramidi pogleda (frustum surfaces)
- uklanjanje stražnjih poligona (Culling)

odsijecanje (clipping)



## U našem slučaju:

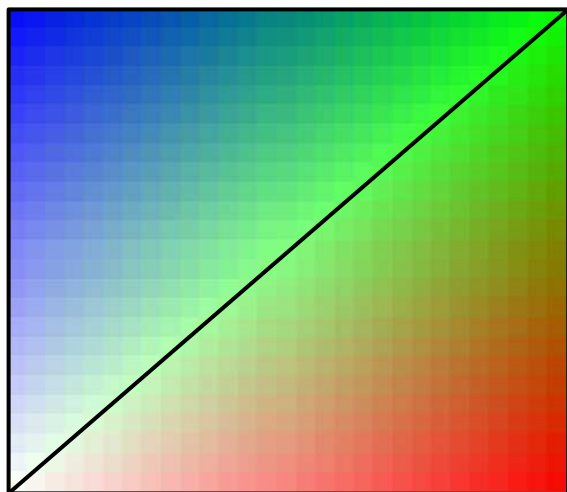
- ništa (samo se provjerava)
- ako bi neki vrh bio van prozora tada se odsijeca



## Rasterizacija:

- određuje se koji slikovni elementi čine primitivu
- stvaranje fragmenta za svaki sl. el.
- pridruživanje atributa (npr. boja) svakom fragmentu

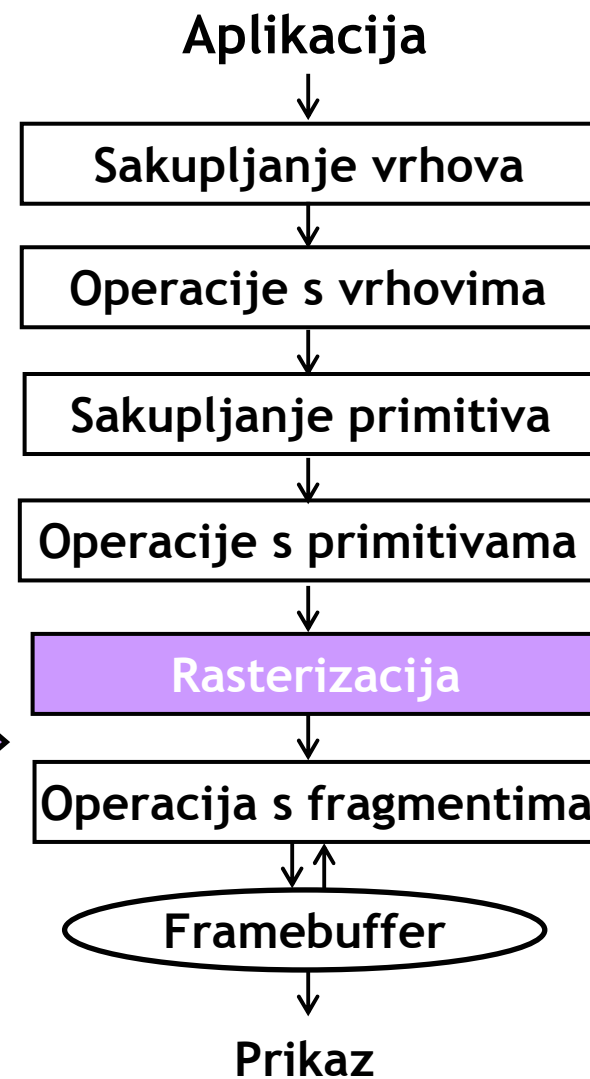
U našem slučaju:



```
struct {  
    float x,y,z,w;  
    float r,g,b,a;  
} vertex;
```

```
struct {  
    vertex v0,v1,v2  
} triangle;
```

```
struct {  
    short int x,y;  
    float depth;  
    float r,g,b,a;  
} fragment;
```



### Operacije s fragmentima (fragment shader):

- preslikavanje teksture, magla,
- proračun osvjetljenja po fragmentu  
(različiti **l**, **n** vektori prema izvoru i vektor normale za svaki slikovni element)

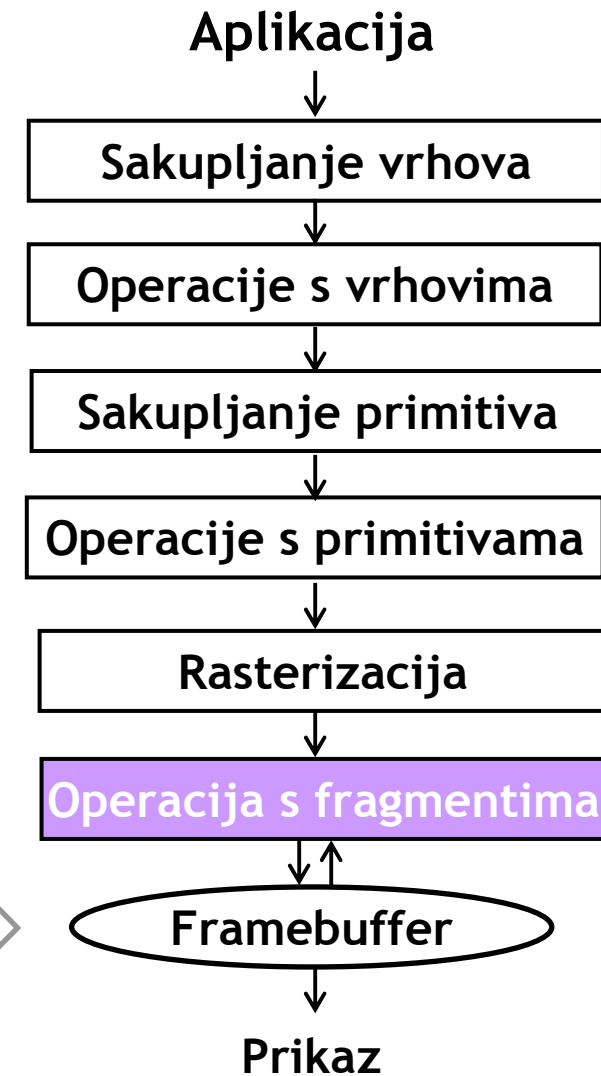
### Veza sa slikovnom memorijom (Framebuffer)

- konačna slika se gradi i sastavlja ovisno o Z-spremniku, miješanju boja (color blending), ...

### U našem slučaju:

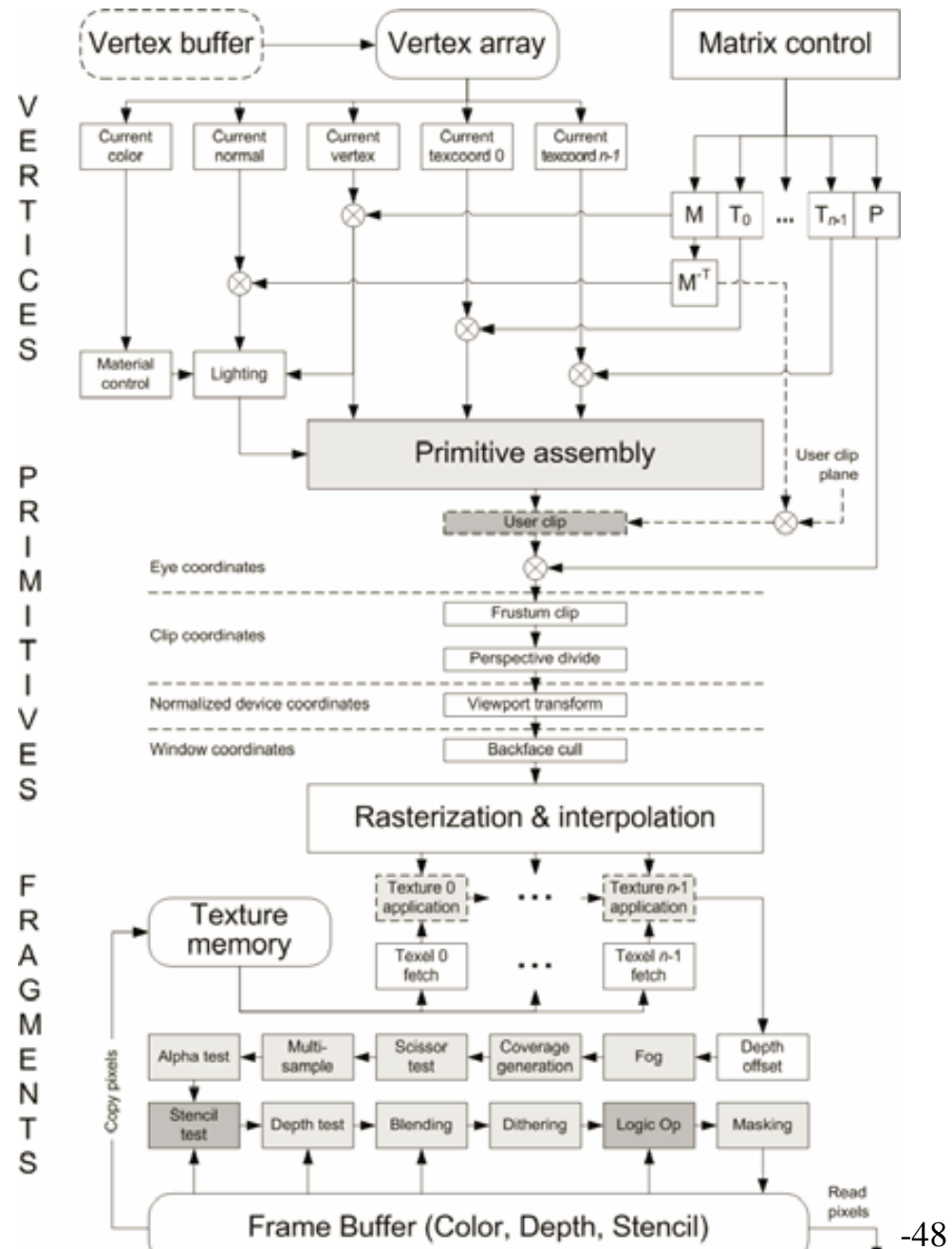
- ništa

```
struct {  
    int depth;  
    byte r,g,b,a;  
} pixel;
```



## Protočni sustav za OpenGL ES (ugrađene sustave)

- vrhovi
- primitive
- fragmenti





# Osnove GLUT-a

## struktura aplikacije

- konfiguracija i otvaranje prozora
- inicijalizacija OpenGL stanja
  - npr. boja pozadine, položaji izvora,
- funkcije povratnog poziva (engl. callback) – kada se dese događaji pozivaju se funkcije (osvježi prozor...)
  - prikaz – osvježavanje prikaza
  - promjena veličine prozora
  - ulazne naprave: miš, tipkovnica - rukovanje događajima (engl. events)
  - animacija
- ulazak u glavnu petlju procesiranja događaja
  - aplikacija prati događaje i rukuje (poziva funkcije ovisno o događaju)

primjer glavnog programa:

```
void main (int argc, char** argv)
{
    glutInitDisplayMode (GLUT_RGB | GLUT_DOUBLE); // konfiguracija i otvaranje prozora
    glutInitWindowSize (200, 300);
    glutInitWindowPosition (100, 100);
    glutCreateWindow ("Moj prozor");

    glClearColor( 0.0, 0.0, 0.0, 1.0 ); // inicijalizacija OpenGL stanja
    glClearDepth( 1.0 ); // inicijalizacija početnih stanja je često u potprogramu init();
    glEnable( GL_LIGHT0 );
    glEnable( GL_LIGHTING );
    glEnable( GL_DEPTH_TEST );

    glutDisplayFunc (crtaj); // funkcije povratnog poziva -- callback
    glutReshapeFunc (resize);
    glutKeyboardFunc (tipkovnica);
    glutIdleFunc (idle);

    glutMainLoop(); // glavna petlja procesiranja događaja
}
```

neke od funkcija povratnog poziva koje GLUT podržava:

- korisnik treba napisati ove funkcije

poziva se kada:

`glutDisplayFunc (crtaj);` – treba osvježiti slikovne elemente

`glutReshapeFunc ();` – se promijeni veličina prozora

`glutKeyboardFunc (tipkovnica);` – je pritisnuta tipka na tipkovnici

`glutMouseFunc (mojmis);` - je pritisnuta tipka na mišu

`glutMotionFunc ();` - je pritisnuta tipka na mišu i pomičemo miša

`glutPassiveMouseFunc ();` - pomičemo miša neovisno o tipkama miša

`glutIdleFunc ();` – kada se ništa drugo ne dešava – korisno u animaciji

primjer:

```
void crtaj (void)
{
    glClear (GL_COLOR_BUFFER_BIT);          // brisanje i obično glavno iscrtavanje
    glBegin (GL_TRIANGLES);
        glColor3ub(255, 0, 0); glVertex3f(-1.0, 0.0, 0.0);
        glColor3ub(0, 0, 0);   glVertex3f(0.0, 1.0, 0.0);
        glColor3ub(100, 0, 0); glVertex3f(0.0, 0.0, 1.0);
    glEnd ();
    glutSwapBuffers ();                      // ako je jedan spremnik onda je glFlush();
}
```

## Interaktivni rad - GLUT funkcije:

- kada je pritisnuta neka tipka tipkovnice poziva se funkcija:

`glutKeyboardFunc (tipkovnica);`

```
void tipkovnica (unsigned char tipka, int x, int y) // za tekući prozor pritisak tipke generira
{
    // ASCII znak – pokreće funkciju – prosljeđuje znak i x, y miša
    switch (tipka) {
        case 'r' : case 'R' :
            glColor3f (1,0,0); // promijenimo stanje – aktivna boja prikaza je crvena
            glutPostRedisplay();
            break;
        case 27 : exit (0);
            break;
    }
}
```

- pritisnuta tipka miša:

```
void mojmis (int button, int state, int x, int y)
{
    if (button == GLUT_LEFT_BUTTON && state == GLUT_DOWN)
        {...}
    else if (button == GLUT_RIGHT_BUTTON && state == GLUT_DOWN)
        {...}
}
```

## Animacija - GLUT funkcije:

- potrebno je osvježavati prikaz za svaki dt:

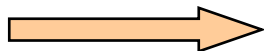
```
glutIdleFunc (idle);  
void idle (void)  
{  
    t += dt;  
    glutPostRedisplay();  
}
```

# Grafičko korisničko sučelje

## Fizičke naprave



Stisnuta tipka(e)  
Miš – pozicija  
- promjena, pomak  
- klik  
3D podaci ...



Osluškivanje procesa (notify process)

## Događaji (event)

- kontekstno ovisne akcije  
ovisno u kojem prozoru,  
na kojem mjestu  
i kada (poslije/prije neke akcije)

Podjela zaslona (layout), **logički elementi sučelja**

- padajući izbornici (menu)
- panel
- canvas

## Pokretanje funkcija (callback)

- npr. rotacijom u sceni  
možemo upravljati s  
tipkovnice ili mišem

