

# 컴퓨터 그래픽스

## 1. 컴퓨터 그래픽스 개요 및 시스템

2020년 2학기

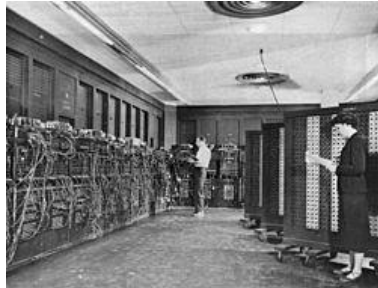
# 컴퓨터 그래픽스의 발전

- 컴퓨터 그래픽스 정의

- 컴퓨터 그래픽스는 컴퓨터를 이용해 그림을 생성하는 기술
- 영상처리 (Image Processing): 있는 그림을 개선하거나 인식하는 분야

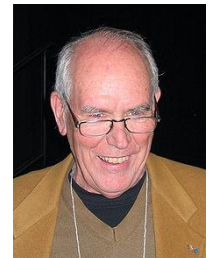
- 1960년대까지의 컴퓨터 그래픽스

- 1946년: 펜실베니아 대학의 모클리 와 애커트가 세계 최초의 진공관 컴퓨터인 에니악(ENIAC, Electronic Numerical Integrator and Computer) 발명



출처: 위키피디아

- 1962년: MIT의 Ivan Sutherland는 CRT 위에 라이트 펜으로 직접 그림을 그릴 수 있는 스케치 패드 (Sketchpad) 프로그램을 개발
- 1969년: ACM 학회에 그래픽스 협의회인 SIGGRAPH 발족

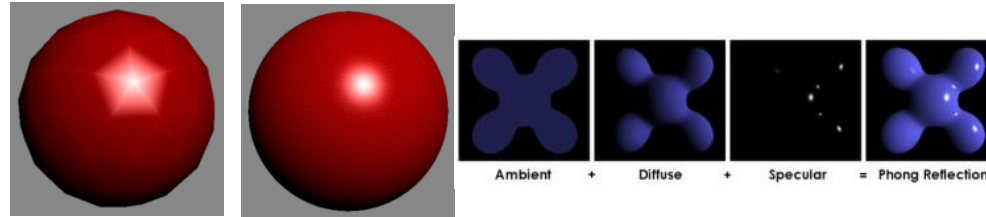


출처: 위키피디아

# 컴퓨터 그래픽스의 발전

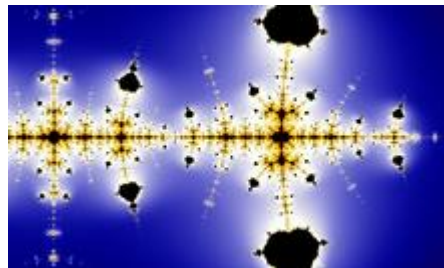
- 1970년대의 컴퓨터 그래픽스

- 본격적인 컴퓨터 그래픽스가 활용되는 시기
- 1972년: 래스터 그래픽스 등장
- 1974년: 유타 대학의 Bui Tuong Phong은 Phong shading 알고리즘 개발



출처: 위키피디아

- 1977년: 폴란드 수학자인 Benoit Mandelbrot가 [프랙털](#) 기하학 분야를 정립

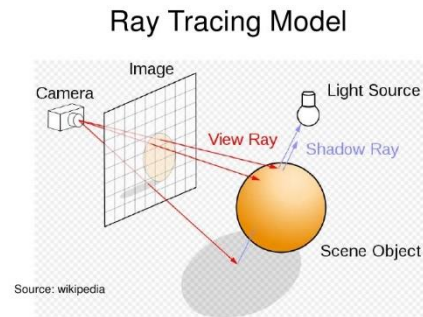


출처: 위키피디아

# 컴퓨터 그래픽스의 발전

- 1980년대의 컴퓨터 그래픽스

- 개인용 컴퓨터의 급속한 발전과 보급으로 컴퓨터 그래픽스가 대중화된 시기
- HCI 기술 발전(윈도우, 메뉴, 아이콘, 마우스 등)
- 1980년: 렌더링 테크닉인 [레이 트레이싱](#) (광선 추적 알고리즘)



출처: 위키피디아

- 1982년: John Walker와 Dan Drake가 AutoCAD 소프트웨어 개발
- 1984년: GUI를 이용한 개인용 컴퓨터 Macintosh 개발
- 1985년: NES사가 가정용 게임기 Nintendo 개발
- 1986년: 애플의 스티브 잡스가 Lucas 필름으로부터 [PIXAR](#) 인수
- 1987년: IBM사에서 VGA 그래픽 카드 개발
- 1989년: PIXAR의 "[tin toy](#)" 가 아카데미상 수상

# 컴퓨터 그래픽스의 발전

- 1990년대의 컴퓨터 그래픽스

- 3차원 그래픽스의 발전, 인터넷 환경 발전
- 1990년: 오토데스크사의 Gary Yost는 3차원 컴퓨터 그래픽스를 위한 디자인 소프트웨어인 3D max
- 1992년: 실리콘 그래픽스사에서 OpenGL 사양 발표
- 1995년: Pixar사는 최초의 3D 애니메이션 “[Toy story](#)” 제작
- 1995년: 마이크로소프트사는 DirectX API 사양 발표
- 1999년: NVIDIA사는 GeForce256 GPU 개발

- 2000년대 이후의 컴퓨터 그래픽스

- 실시간 렌더링
- [물체의 사실감](#)과 자연스러움 증가
- 모바일 환경

## 2차원 그래픽스

- 2차원 그래픽스

- 점, 선, 원, 곡선 등과 같은 기본 도형을 이용하여 2차원 평면상에 구현된 디지털 이미지
- 결과물을 픽셀의 형태로 표현: 각 픽셀은 적색(Red), 녹색 (Green), 청색(Blue)의 농도 값을 배합
- 그래픽의 표현방법과 처리방식에 따라 벡터 그래픽스와 래스터 그래픽스로 구분

- 벡터 그래픽(Vector Graphics)

- 그래픽에 사용된 객체들을 점과 선을 이용해 수학적 함수로 표현하여 기억 공간에 저장하는 방식
- 파일의 크기가 래스터 그래픽 방식으로 저장한 것보다 작음
- 기하적 객체를 수식의 형태로 표현하므로 화면 확대 시에도 화질의 변화가 없음

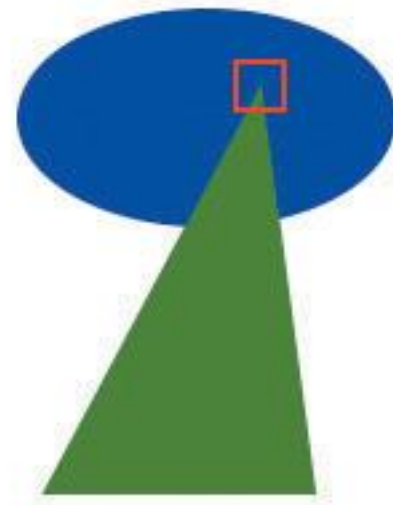
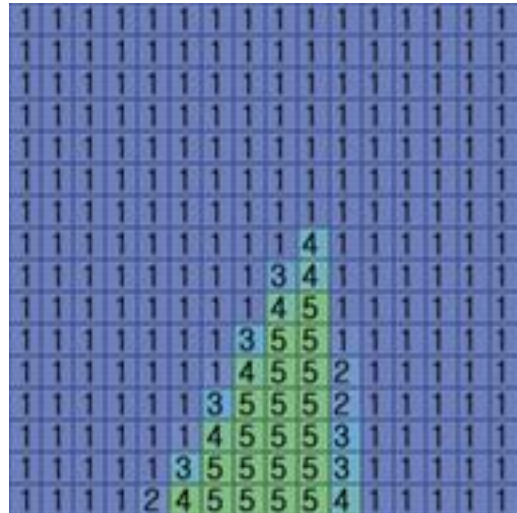
```
<?xml version = "1.0"?>  
<svg width="200" height="200">  
<ellipse cx="110" cy="50" rx="70" ry="40">  
style="fill:blue; stroke:blue; stroke-width:2"/>  
<polygon style="fill:green;stroke:green;"  
stroke-width:2" points="130,40 140,190 50,190"/>  
</svg>
```



## 2차원 그래픽스

### - 래스터 그래픽(Raster Graphics)

- 래스터 그래픽 출력장치에 표시하기 위한 그래픽 데이터를 픽셀단위로 기억 공간에 저장
- 저장된 파일의 크기는 출력장치의 해상도에 비례
- 화면을 확대하면 화질이 떨어짐



# 3차원 그래픽스

- 3차원 그래픽스

- 컴퓨터에 저장된 모델의 기하학적 데이터를 이용해 3차원으로 표현한 뒤 2차원 적 결과물로 처리, 출력하는 컴퓨터 그래픽

- 3차원 그래픽 생성과정 :

- 물체의 기하학적인 형상을 모델링(Modeling)
    - 3차원 물체를 2차원 평면에 투영(Projection)
    - 생성된 3차원 물체에 색상과 명암을 부여(Rendering)



# 3차원 그래픽스

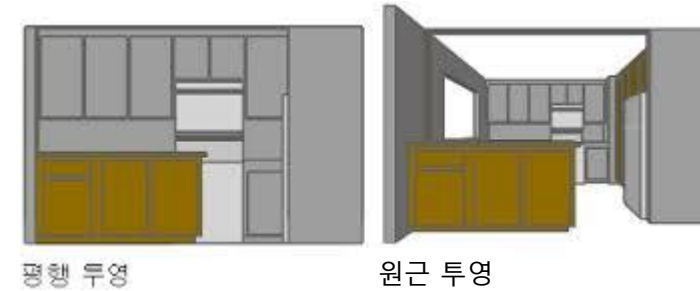
- 모델링(Modeling) 과정
  - 3차원 좌표계에서 물체의 모양을 표현하는 과정
    - 와이어프레임(Wireframe)모델
    - 다각형 표면(Polygon Surface)모델
    - 솔리드(Solid)모델링
    - 3차원 스캔에 의한 모델링



# 3차원 그래픽스

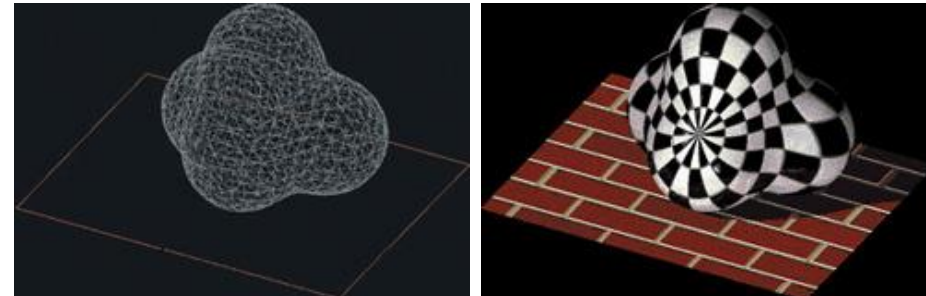
- 투영(Projection) 과정

- 3차원 객체를 2차원 화면에 투영
  - 평행 투영법, 원근 투영법



- 렌더링(Rendering) 과정

- 색상과 명암의 변화와 같은 3차원적인 질감을 더하여 현실감을 추가하는 과정
  - 은면의 제거(Hidden Surface Removal)
  - 셰이딩(Shading), 텍스처 매핑(Texture Mapping), 그림자(Shadow)
  - 광선 추적법(Ray Tracing)



# 컴퓨터 그래픽스 활용

- CAD (Computer Aided Design)
  - 부품설계 및 도면작성(Drafting), 기계설계
  - VLSI 설계, 전자회로 설계
  - Communication network, Water/Electricity supply system
  - 자동차, 비행기, 선박의 설계: Wireframe model
  - 건축 설계



# 컴퓨터 그래픽스 활용

- 컴퓨터 애니메이션과 시뮬레이션(Simulation)
  - 프레임들의 빠른 연속적인 디스플레이
  - Education, Training(Flight simulator), Physical system 의 Behavior 연구



출처: Walt Disney pictures

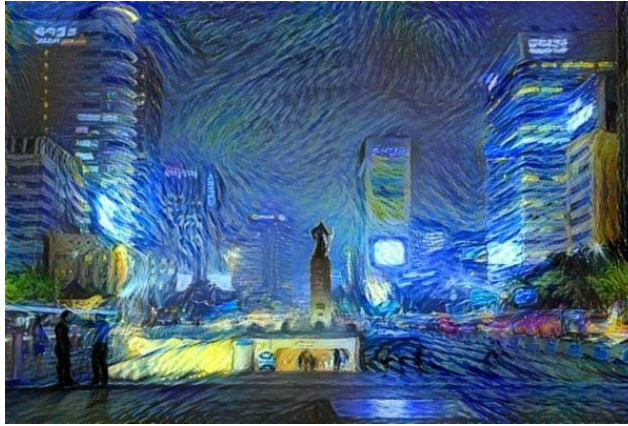


출처: Microsoft Flight Simulator

# 컴퓨터 그래픽스 활용

- 컴퓨터 디자인 및 아트

- 상업 디자인(Commercial art)
- 창작 미술(Creative art)



출처: 조선비즈  
(구글 딥드림을 활용해 고호의 화풍으로 그린 광화문 그림)

- 게임 및 엔터테인먼트

- 영화, 게임, 뮤직 비디오, TV 프로그램 등

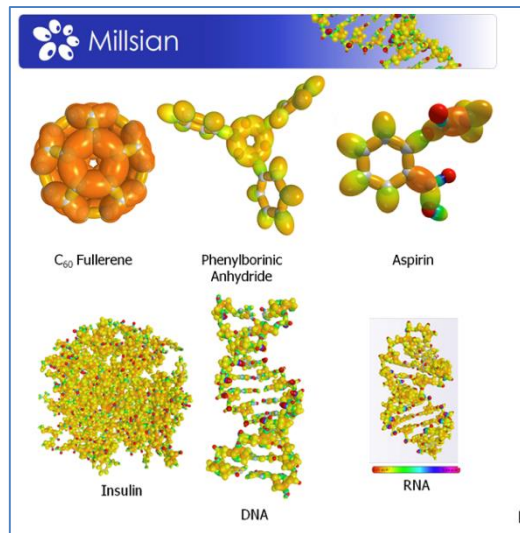


출처:  
<http://animationandvideogames.blogspot.com/2013/11/motion-capture-methods.html>

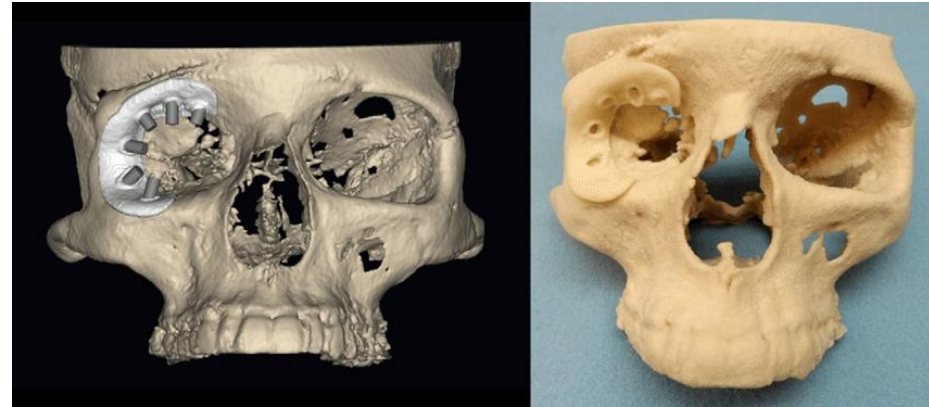


# 컴퓨터 그래픽스 활용

- 프레젠테이션 및 데이터 시각화(Data Visualization)
  - 프레젠테이션 그래픽스(Presentation Graphics): Graph, Chart, Business graphics, Project management
  - Computer generated model (Visualization): Physical, Financial, Economic model



출처: brilliantlightpower.com



출처: ResearchGate (Virtual surgical planning and 3D printing in prosthetic orbital reconstruction with percutaneous implants: A technical case report)

# 컴퓨터 그래픽스 활용

- 가상 현실(VR), 증강 현실(AR), 혼합 현실(MR)
  - Virtual Reality: 컴퓨터 등을 사용하여 인공적인 기술로 만들어낸 실제와 유사한 환경을 만들어내는 기법
  - Augmented Reality: 가상 현실의 한 분야로 실제로 존재하는 환경에 가상의 사물이나 정보를 합성하여 실제로 존재하는 사물처럼 보이는 기법
  - Mixed Reality: 실제 세계를 디지털 세계와 혼합한 결과



영화 아이언맨 (출처: 네이버 영화)

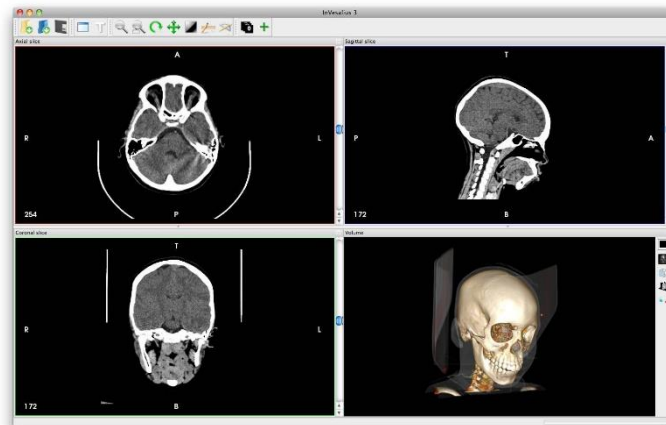


LG360 VR 체험 모습 (출처: LG전자 블로그)

[매직 리프가 공개한 MR](#)

# 컴퓨터 그래픽스 활용

- 전자 출판
  - Document Preparation System
- 공간 정보의 표현
  - 지리정보시스템(GIS: Geographic Information System)
  - 차량 주행안내 시스템(Car Navigation System)
- 이미지처리(Image Processing)
  - Feature Detection
  - Pattern Recognition
  - 3D Reconstruction(예: MRI, CT)



출처: <https://medevel.com/invesalius-3d-dicom/>



# 그래픽스 시스템

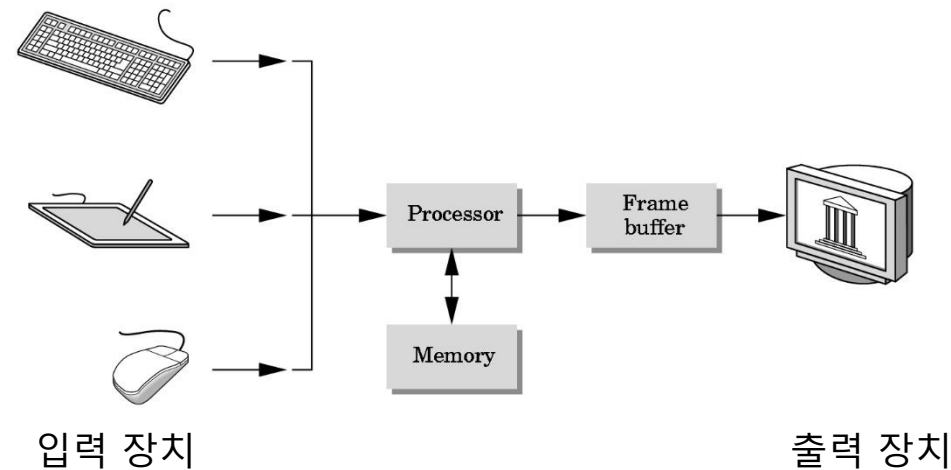
- 그래픽스 시스템 구성:

- 그래픽스 하드웨어:

- 그래픽 데이터를 입력 또는 생성하는 입력 장치
    - 결과물 출력하는 출력장치
    - 그래픽스 프로세서

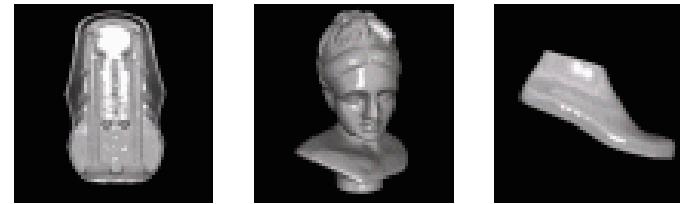
- 그래픽스 소프트웨어:

- 그래픽 데이터를 생성하고 처리하여 원하는 형태의 그래픽 결과물을 디스플레이 하기 위해서 사용되는 프로그램
    - 그래픽스 라이브러리 및 응용 프로그램



# 입력 장치

- 2차원 입력 장치
  - 마우스
  - 그래픽스 태블릿
  - 조이스틱
  - 라이트 펜
- 3차원 입력 장치
  - 3D 디지털타이저, 3D 스캐너
  - [모션 캡처 장치](#)



# 출력 장치

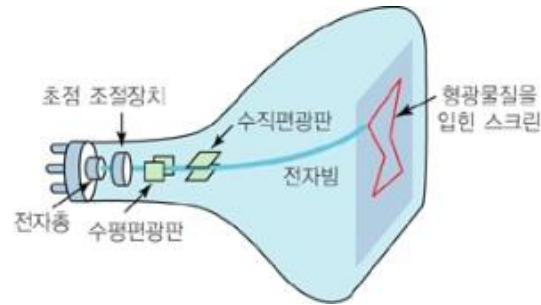
- 프린터

- 프린터
- 플로터
- 3D 프린터 ([샘플](#))

- 모니터

- 브라운관 모니터 (CRT, Cathode-Ray Tube)

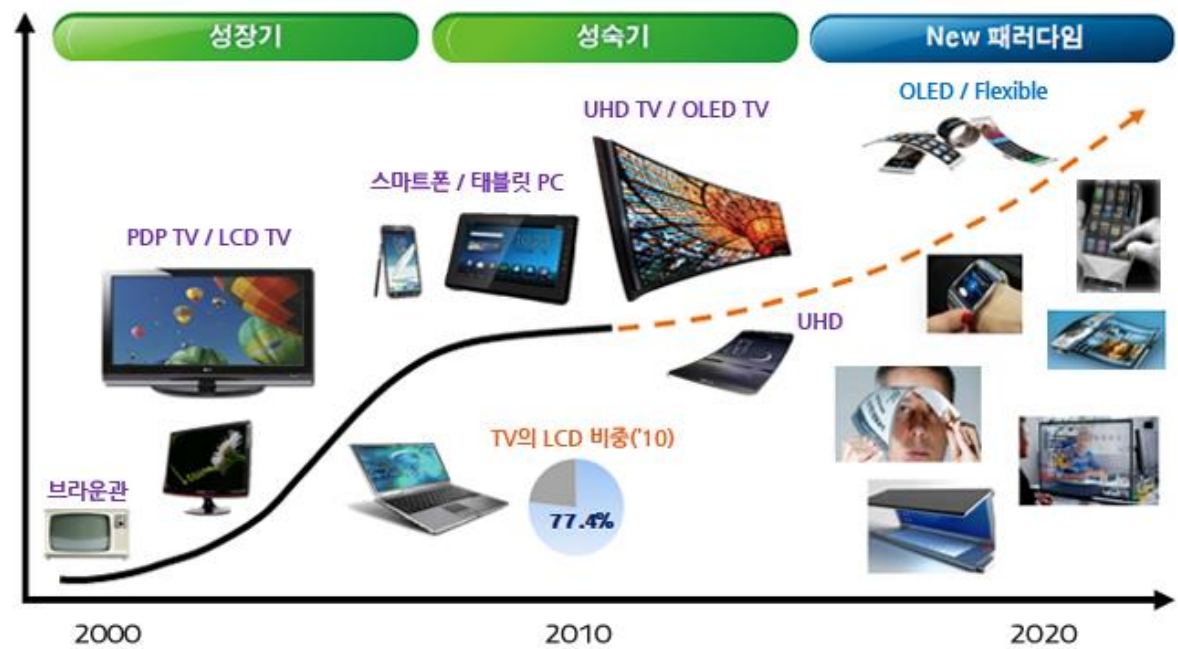
- 컴퓨터의 출력은 디지털 아날로그 변환기 (digital-to-analog converter)에 의해 수직, 수평 편광판 사이의 전압으로 변환, 충분한 양의 전자선이 형광 물질에 도달하면 CRT 표면에서 빛이 방출된다.



# 출력 장치

- 평판 디스플레이 모니터 (FPD, Flat Panel Display)
  - 전면 유리에 전자를 쏘서 표시하는 방식으로 CRT의 전자총을 얇은 판으로 대체하여 얇고 가벼운 디스플레이 장치, 열이 덜 발생하고 전력 소모가 작다.
  - LCD (Liquid Crystal Display)
    - 액정 디스플레이
    - 시계, 컴퓨터, TV, 자동차/항공기 속도표시판 등에 폭넓게 사용
    - 소비전력이 적고 가볍고 슬림해서 휴대하기 편리한 장점을 가지고 있으나 스스로 빛을 내지 못하고 시야각과 휘도가 좋지 않고 응답속도가 느린 단점
  - PDP (Plasma Display Panel)
    - 플라즈마 디스플레이
    - 플라즈마 방전 시 발생하는 자외선을 화상으로 표시
    - 발광형으로 선명한 표시가 가능, 고전력이 필요하다.
  - LED (Lighting Emitting Diode)
    - 발광 다이오드
    - 에너지가 가해졌을 때 가시광선이나 비가시광선(적외선)을 발산하는 다이오드로 처리속도, 전력소모, 수명 등에서 큰 장점
  - OLED (Organic Light Emitting Diode)
    - 유기발광다이오드
    - LCD에 비해 더 높은 대비와 더 나은 시야각

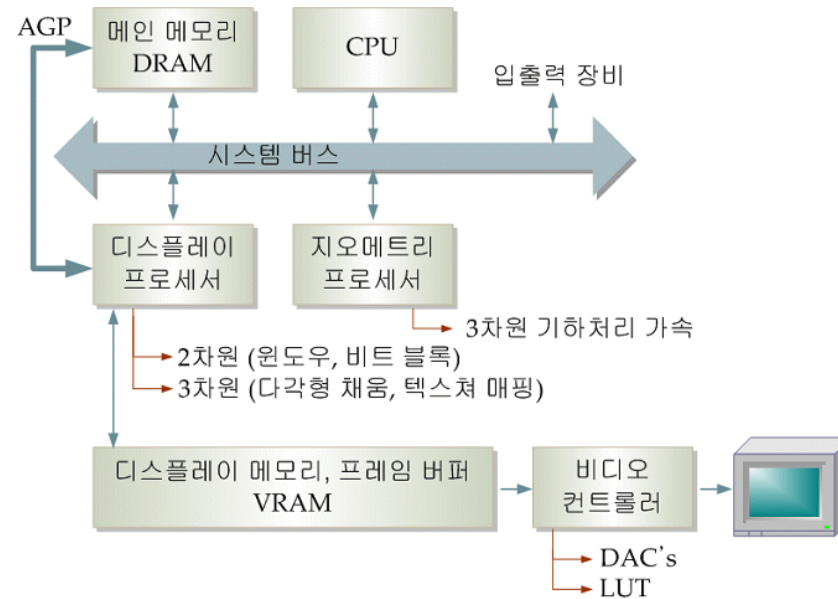
# 출력 장치



# 그래픽스 프로세서

- 그래픽스 프로세서

- 그래픽 데이터 처리: 컴퓨터에 의해 생성된 디지털 정보를 최종적으로 아날로그 신호로 변환하여 화면에 그림을 그리게 하는 역할 담당
- 그래픽 카드의 구조



# 그래픽스 프로세서

- 그래픽스 프로세서에 의해 그래픽스 프로그램이 처리되는 수행 방식
  - 랜덤 스캔 방식
    - 그래픽 처리 과정
      - 그래픽스 응용 프로그램 실행 → 디스플레이 파일 생성되어 시스템 메모리에 저장 → 출력 프로세서에 의해 매 활성주기마다 모니터에 출력
    - 벡터 모니터에서 그림이 처리되는 방식

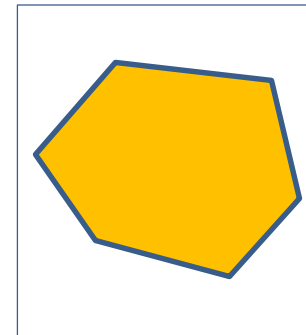
Application  
Program

```
#include <gl/glut.h>
...
void display ()
{
    glClear (GL_COLOR_BUFFER_BIT);
    ...
}
```

Display  
File program

```
glBegin(GL_POLYGON);
glVertex(0.1, 0.1, 0.1);
...
glEnd ();
...
```

Monitor



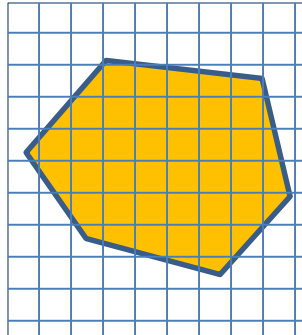
# 그래픽스 프로세서

- 그래픽스 프로세서에 의해 그래픽스 프로그램이 처리되는 수행 방식
  - 래스터 스캔 방식 (Raster Scan)
    - 그래픽 데이터 처리 과정
      - 그래픽스 응용 프로그램 실행 → 처리결과가 그래픽스 명령어 형태로 생성 → 명령어가 수행되어 픽셀 형태로 프레임 버퍼에 저장 → 주기적으로 모니터에 재생
    - 빠른 활성화를 위해 이중 프레임 버퍼를 사용
    - 래스터 모니터에 적용되는 방식으로 대부분의 그래픽스 프로세서 방식

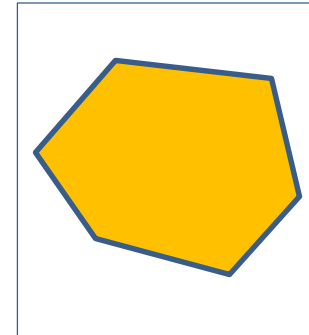
Application  
Program

```
#include <gl/glut.h>
...
void display ()
{
    glClear (GL_COLOR_BUFFER_BIT);
    ...
}
```

Frame  
Buffer



Monitor





# 그래픽스 프로세서

- GPU (Graphics Process Unit)

- 프레임 버퍼에 그래픽 이미지를 빠르게 생성하고 처리하기 위한 프로세서
- 병렬처리 프로세서를 사용
  - 그래픽 및 이미지 처리 알고리즘이 빠르게 처리, 조작 될 수 있게 구성
  - 작고 단순한 연산 코어를 가지고 병렬 연산을 수행
  - 1999년 NVIDIA사의 GeForce256에서 처음 사용
- 셰이더 기능을 지원
  - 셰이더: 하드웨어 연산 장치로 작은 프로그래밍이 가능한 코어 프로세서
  - 셰이딩 언어를 이용하여 병렬로 화면의 폴리곤들을 처리하며 고품질의 실시간 3차원 그래픽스 처리가 가능

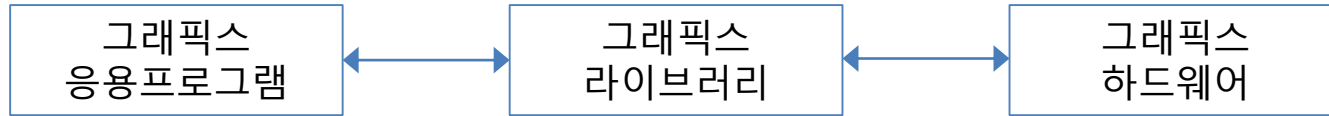


# 그래픽스 소프트웨어

- **그래픽스 소프트웨어**

- **그래픽스 라이브러리:**

- 그래픽스 응용 프로그램을 개발하는데 필요한 기능과 알고리즘을 제공하는 라이브러리
    - OpenGL, Direct3D 등



- **그래픽스 응용프로그램:**

- 특정한 분야의 그래픽을 제작하거나 특수한 목적으로 개발된 경우
    - CAD, 3D Max 등