

# 02 CHAPTER

## 그래픽스 시스템과 사용자 인터페이스



### 2장. 학습 내용



- 그래픽스 출력 장치
- 래스터 스캔 시스템
- 그래픽 입력 장치
- 상호작용 입력 기법 (Interactive Techniques)
- 사용자 인터페이스 구현

## 2.1 그래픽스 출력 장치



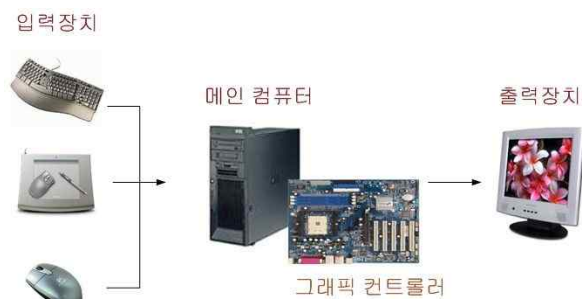
- 그래픽스 출력 장치
  - 그래픽스 시스템 하드웨어 구성
  - 음극선관 모니터의 구동원리
  - 모니터와 관련된 다양한 용어
  - 래스터 방식과 벡터 방식 모니터
  - 비월주사의 개념
  - 평면 디스플레이 장치 및 다양한 그래픽 출력 장치

3

## 그래픽스 시스템 하드웨어 구성



- 그래픽 시스템
  - 입력 장치, 메인 컴퓨터, 그래픽 컨트롤러, 출력장치
- 그래픽 컨트롤러
  - 그래픽 프로세서



[그림 2-1] 그래픽 하드웨어 구성

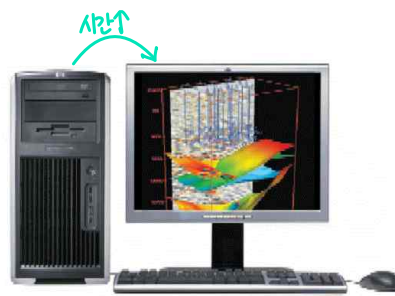
4

## 워크스테이션

- SGI FUEL
- HP XW 9300



[그림 2-2] SGI



[그림 2-3] HP

5

## 시스템 사양

- SGI FUEL

메인 프로세서	800 or 700MHz MIPS R16000A 프로세서
	4MB L2 캐쉬, 200MHz 프론트 사이드 버스
메인 메모리	512MB - 4GB DDR SDRAM
그래픽 컨트롤러	정점 처리 엔진, 영상/텍스처 처리 엔진,
및 그래픽 메모리	알파 채널, 더블 버퍼, 24 비트 자-버퍼, 8 비트 스텐실 버퍼, 332MB/128MB 그래픽 메모리 및 텍스처 메모리, 10 비트 DAC
중요 응용분야	캐드, 의료 영상, 자연과학 가시화, 지질 데이터 분석, 시각적 시뮬레이션

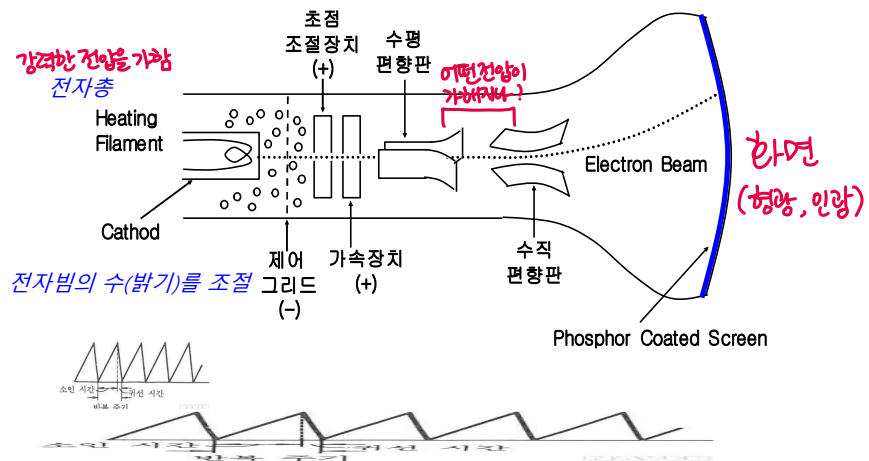
- HP XW 9300

메인 프로세서	(2.0 - 2.6 GHz) Single/Dual AMD Opteron 200 series 프로 세서 800MHz HyperTransparent 버스
메인 메모리	최대 16GB DDR SDRAM
그래픽 컨트롤러	NVIDIA Quadro NVS 280 for Professional 2D
및 그래픽 메모리	NVIDIA Quadro FX 3400 for High-end 3D

6

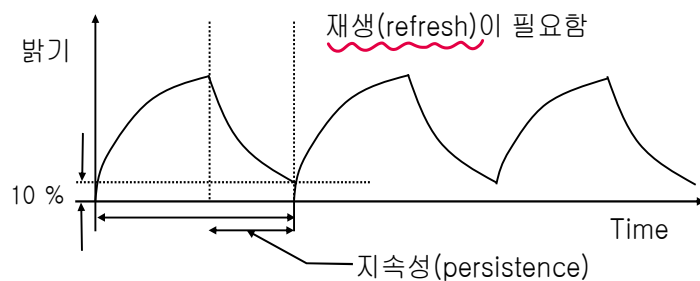
## 흑백 CRT 모니터 ~~시험~~

- CRT 모니터의 원리



7

- 인광물질(Phosphor)
- 정기적으로 재생(refresh)이 필요

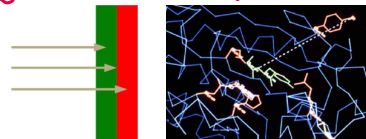


8

## 컬러 CRT 모니터

### • 빔 투과(Beam Penetration) 방식

- 스크린에 여러 종류의 인을 칠해 전자빔의 세기(투과 정도)에 따라 다른 색을 발광하도록 함.
- 출력할 수 있는 색이 제한적이므로 많이 사용되지 않음.



### • 색상별 전자총 방식

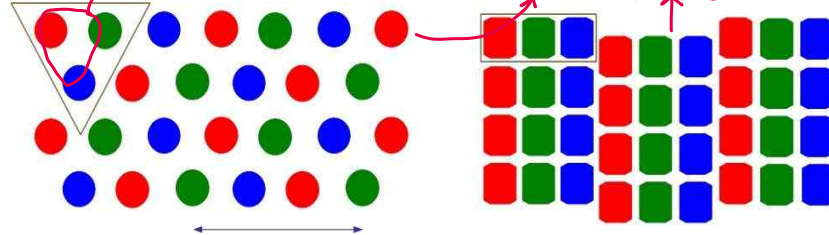
- 전자총 세 개를 사용하고, 스크린의 각 화소에 R, G, B를 각각 발광하는 인광물질을 칠해 해당 전자총이 각각 R, G, B 영역을 정확히 맞추는 방법.
- 대부분의 모니터에서 사용됨.

9

## 색상별 전자총 방식

### • 트라이어드 방식, 스트라이프 방식

*전자빔이 이렇게 맞도록 있음 (해명: 샷다운 마스크, 애퍼처그릴)*



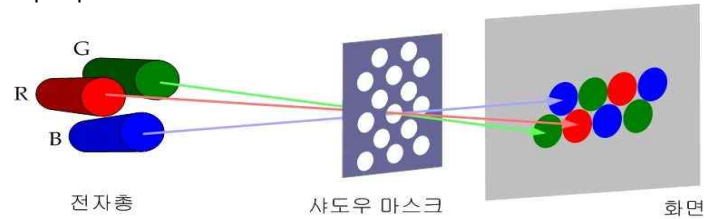
(a)  
트라이어드

(b)  
스트라이프

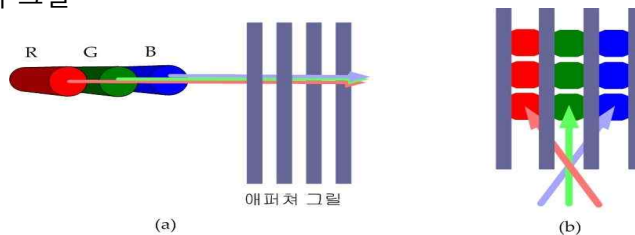
10

## 화면 선명도

- 샤도우 마스크



- 애퍼처 그릴

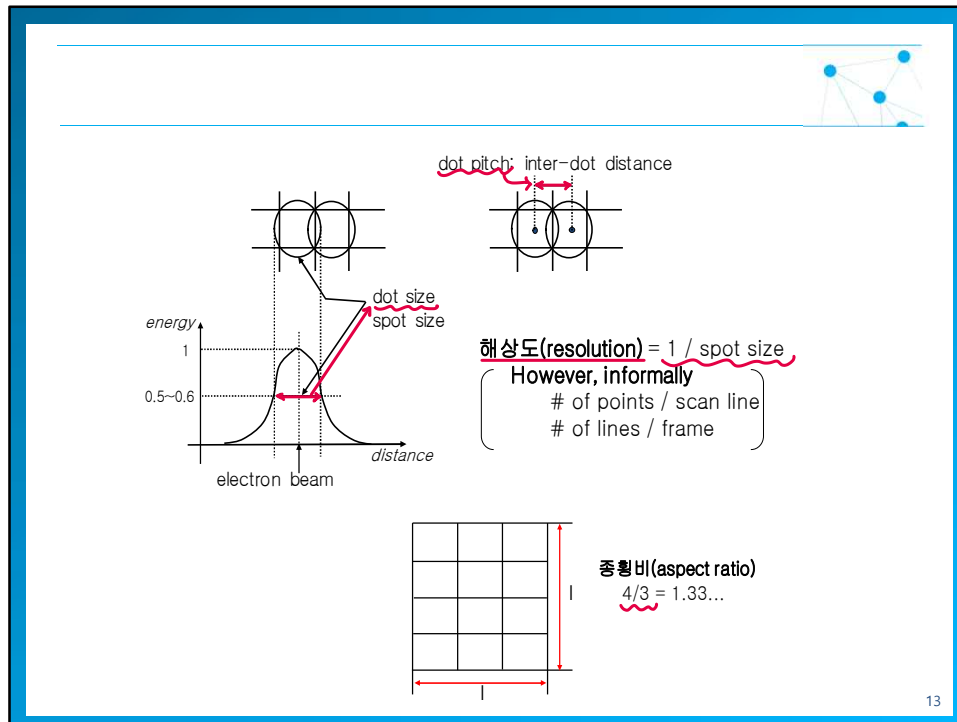


11

## 모니터 관련 용어

- 점의 크기(dot size) 픽셀이 작아나 크다
  - 인점의 에너지가 50~60%로 떨어지는 범위
- 도트 피치(dot pitch)
  - 인접한 인점의 거리
  - 화면의 선명도(sharpness)와 관련
- 해상도(resolution)  $w \times h$ 
  - 점의 크기의 역수 점의 크기
  - 통상적으로 모니터의 화소의 개수(모니터의 수직 점의 수  $w$  x 수평 점의 개수  $h$ )를 말하는 용어로 사용
- 종횡비(aspect ratio)
  - 동일한 길이의 수직선과 수평선을 만들기 위해 필요한 점의 수의 비율
  - 전체 모니터의 가로와 세로 화소 수의 비율을 말하는 용어로 흔히 사용

12



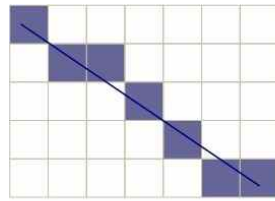
13

- 화면 재생(refresh)
- 재생율(refresh rate)
  - 그림을 화면에 다시 그리는 회수
- 벡터(vector display) 래스터(raster display)
  - TV나 PC 모니터와 같이 현재 우리가 사용하는 대부분의 모니터는 래스터 방식
  - 전자 계측 장치인 오실로스코프 등에서 벡터 방식을 흔히 사용

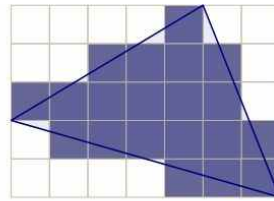
14

## 래스터 그래픽 디스플레이

- 화소단위 컬러링
- 계단현상 = Jaggies = Alias
- 래스터 화(Rasterization) = 스캔 변환(Scan Conversion)
  - 물체 좌표에서 화면 좌표로 *Just, 화면 구성*
  - 부동소수 좌표에서 정수 좌표로



(a)

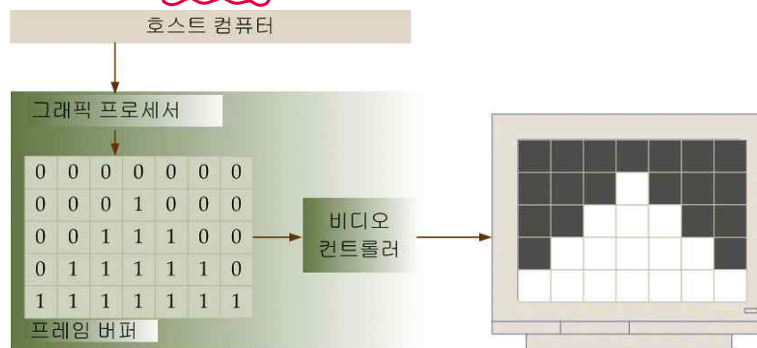


(b)

15

## 프레임 버퍼

- Frame Buffer = Color Buffer = Video Memory
- DA 변환기 *Digital → Analog*
  - 프레임 버퍼(디지털 정보) -> 아날로그 전압
  - 거의 동시(*매우 빠름*)



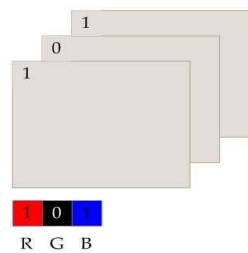
16



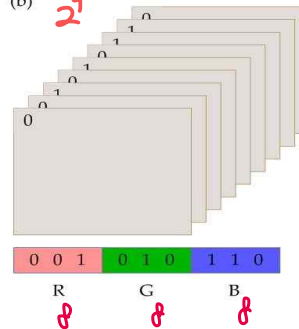
## 비트 평면 레스터방식

- 24비트 (R, G, B) = (8, 8, 8)
  - R, G, B 각각에 대해 256 회색도(Gray Level)
  - 총 몇 컬러?  $2^n$   $n=비트수$
- 1280 by 1024, 총 512 컬러
  - 프레임 버퍼 용량은?

(a)  $2^3$  8가지색



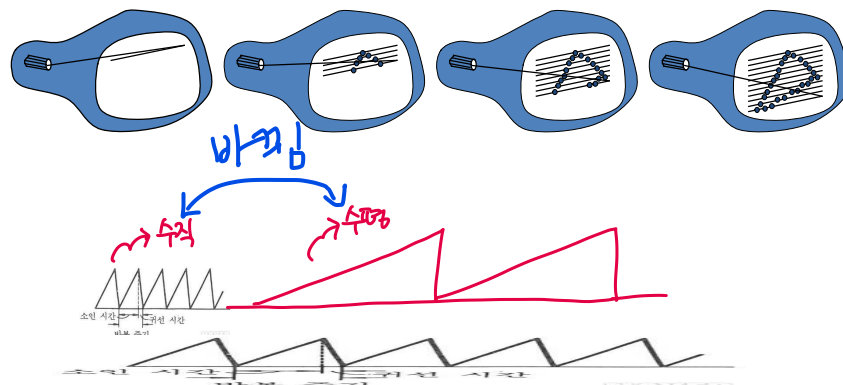
(b)  $2^9$



17

## 주사선(Scan Line)

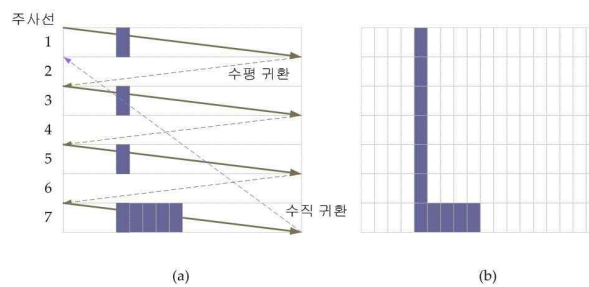
- 래스터 모니터: 전자총의 움직임이 일정하게 제한됨



18

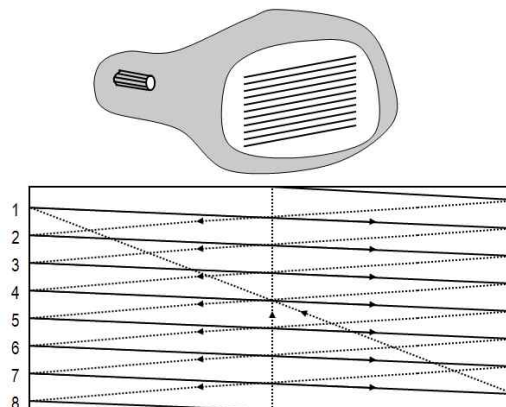
## 인터레이싱 $\Leftrightarrow$ non-Interlacing (고급 하드웨어 성능↑)

- 비월주사, <sup>홀수행/짝수행의 차례</sup> Interlacing, Interleaving)
- Hz: 30 Full Frame/Sec 보다는 60 Half Frame/Sec
- NTSC TV 표준
  - 60 Half Frame/Sec, 525 Scan Line, 4:3 Aspect Ratio



19

- 장점**
- 시각적으로 느껴지는 화면 재생률을  $\uparrow$  높일 수 있다.



20

## Non-Interlacing vs. Interlacing

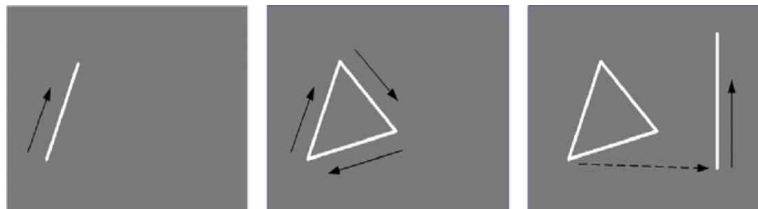


21

## 벡터 그래픽 디스플레이

- Vector Display, Calligraphic Display, Stroke Display, Random Display
- 화소개념 없음
  - 무한 해상도 (전자빔의 폭 -> 해상도)
  - 에일리어싱 없음
  - 전자총의 움직임

화면 갱신속도↓ ∴ 화면이 복잡할수록 느림

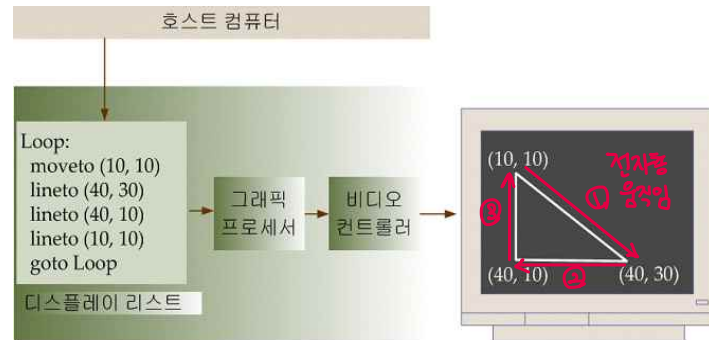


[그림 2-14] 벡터 그래픽 장치의 주사선 움직임

22

- 화소개념 없음

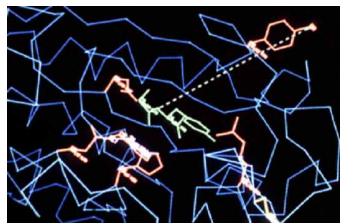
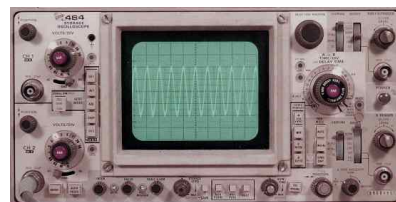
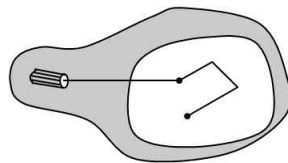
- 프레임 버퍼 ~~없음~~
- 디스플레이 리스트: 명령어 집합



23

## 벡터그래픽 장비

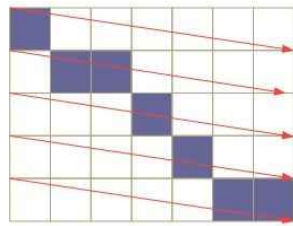
- 오실로스코프, 레이더, 플로터 등
- 컬러: 빔 투과 방식



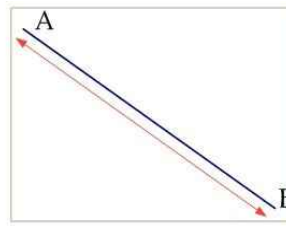
24

## 래스터 그래픽과 벡터 그래픽 비교

	래스터 그래픽	벡터 그래픽
에일리어스	발생	없음
채움 다각형	빠른 렌더링	느린 렌더링
래스터 변환	필수적	불필요
메모리	대용량 프레임 버퍼	소용량 디스플레이 리스트



(a)

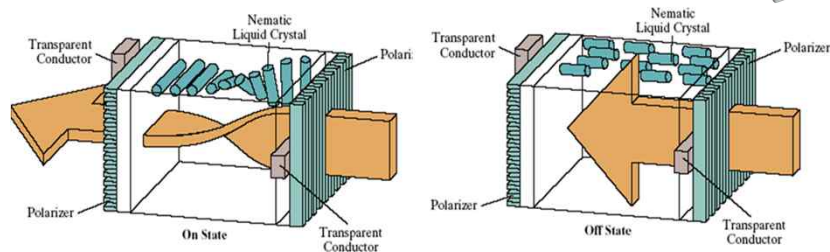


(b)

25

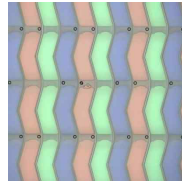
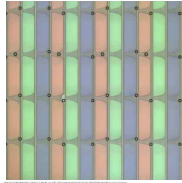
## 평판형 디스플레이(Flat Panel Display) 장치

- LCD (Liquid Crystal Display):
  - Liquid Crystal을 이용하여 빛을 on/off 시키는 원리(블라인드 커튼)를 사용함. 액정의 배열상태 변화에 따라 컬러 값을 만들 수 있음.
  - 자체 발광 방식이 아님. back light가 있어야 함

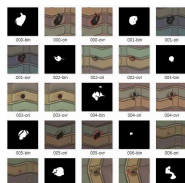
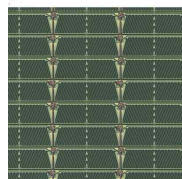
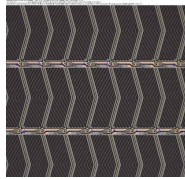


26

- LCD CF

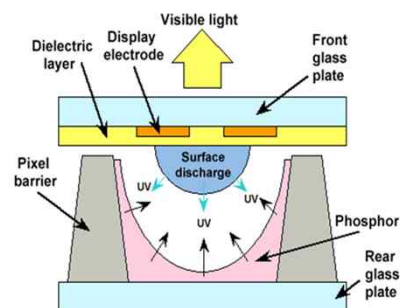


- TFT



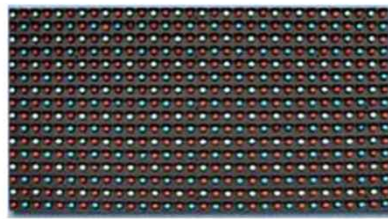
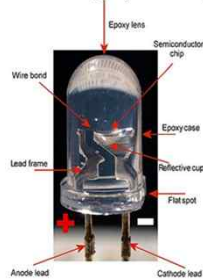
27

- PDP (Plasma Display Panel) **전력 소모**
  - 표면방전에 의한 번개 현상을 이용함(형광등과 같은 원리)
  - 자체 발광이 되며 대형 디스플레이 장치에 활용됨



28

- LED (Light-Emitting Diode)
  - 광 다이오드를 2차원으로 배열한 디스플레이
  - 화면이 매우 선명하고 내구성이 좋음
- LED의 구조와 RGB 및 LED 디스플레이 장치



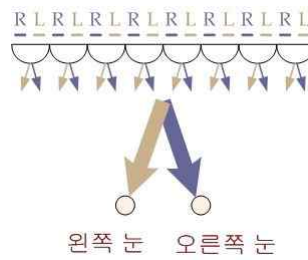
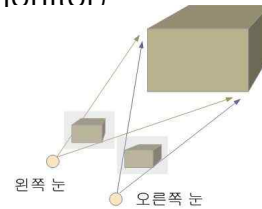
29

- OLED (Organic Light-Emitting Diode)
  - 유기 발광 다이오드로 빛을 내는 층이 유기 화합물로 되어 있음
  - LCD를 대체할 '꿈의 디스플레이'로 각광을 받음. AMOLED 등

30

- 3차원 모니터(Three-Dimensional Monitor)

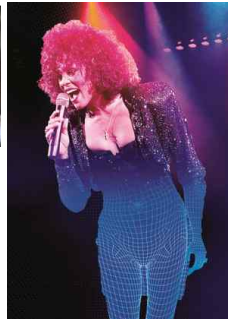
- 좌우 눈의 인식차이 좌우 눈 시야가 다름
- 편광 안경 또는 액정셔터 안경방식
- 렌즈 배열 방식 (무안경 방식)



31

- 홀로그래피

- 모든 방향에서 3차원 뷰가 만들어짐
- 레이저와 육각렌즈 사용



32





- HMD (Head Mounted Display)
  - 머리 탑재형 디스플레이 장치. 사용자의 머리에 장착해 눈앞에 직접 입체 영상을 출력함
- 하드 카피 장치
  - Printer
    - 비교적 작은 규모의 출력에 사용됨
    - 잉크젯 방식, 레이저 방식 등 다양한 방식이 있음
  - Plotter
    - 비교적 큰 출력물을 위해 사용됨
  - 3D-printer
    - 3차원으로 모델링 된 객체의 모형을 만들어 는 장치

33

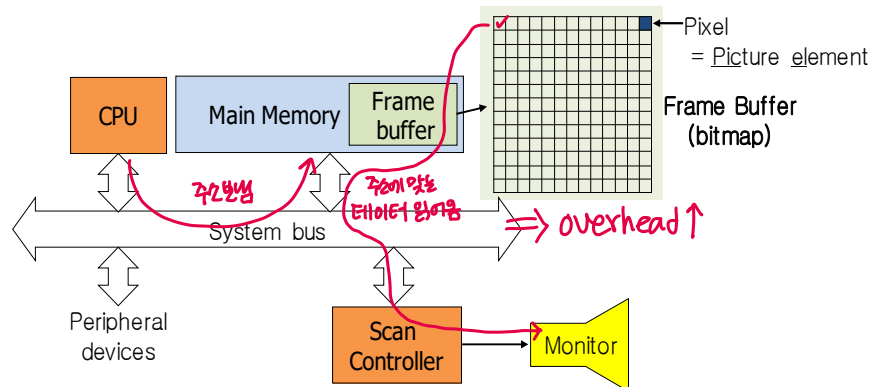


## 2.2 래스터 스캔 시스템

- 단순한 래스터 디스플레이 시스템
- 프레임 버퍼를 분리한 시스템
- 그래픽 가속기를 추가한 시스템

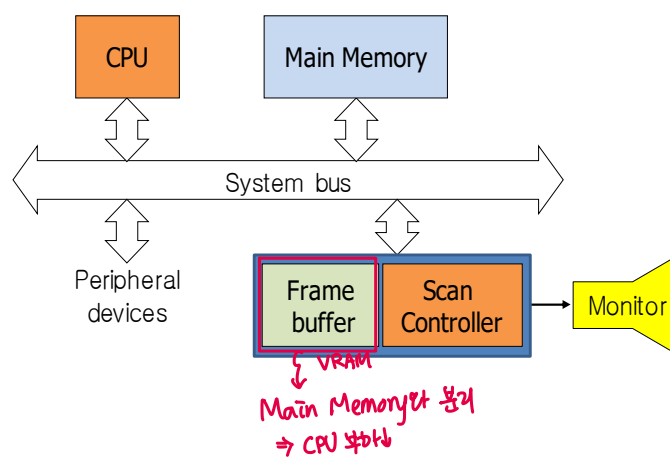
34

## 단순한 래스터 디스플레이 시스템



35

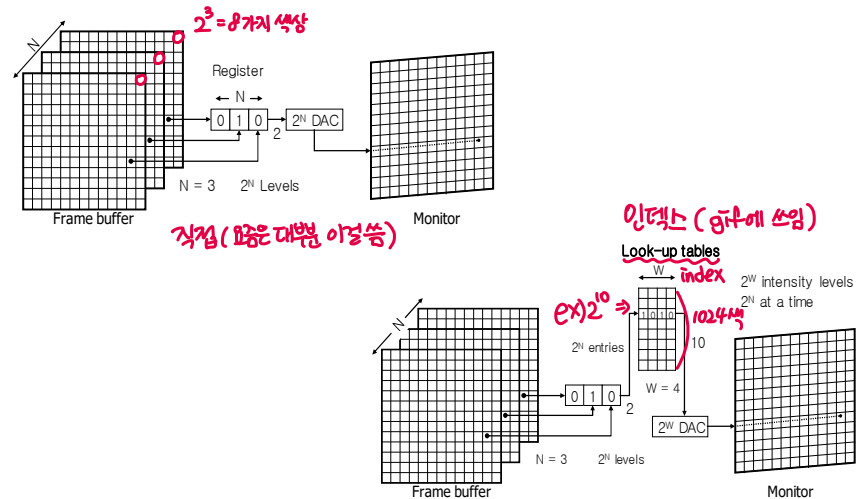
## 프레임 버퍼를 분리한 시스템



36



## 직접(RGB) 컬러 모드/인덱스(indexed) 컬러 모드



39

## 2.3 그래픽 입력 장치

- 물리적(physical) 장치



- 논리적(logical) 입력장치

- 입력을 논리적으로 취급
- 물리적 입력장치가 바뀌어도 프로그램은 동일

40

## 물리적 입력장치

- 마우스, 조이스틱, 트랙볼, 스페이스 볼
  - 상대입력 / 절대입력



[그림 5-1-4] 마우스

조이스틱

트랙볼

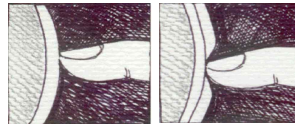
스페이스볼

- 태블릿, 스타일러스 펜
  - 크로스 헤어 커서
  - 디지털라이징



[그림 5-6] 태블릿 II

- 터치 패널
  - 광학 패널, 전기 패널



[그림 5-7] 광학

[그림 5-8] 전기

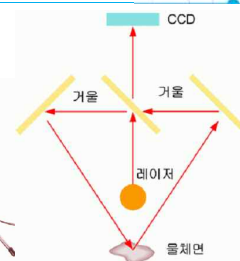
41

## 물리적 입력장치

- 3D 스캐너
  - 물체 표면의 X, Y, Z 좌표 인식
  - 레이저
  - 촬상소자(CCD)



[그림 5-9] 3차원 스캔 작업



[그림 5-10] 레이저스캐너

- 버튼 박스와 다이얼
  - 버튼 박스: 매크로 기능
  - 다이얼:
    - 물체에 대한 기하변환
    - 아날로그 방식



[그림 5-11] 버튼



[그림 5-12] 다이얼

42

## 논리적 입력장치



- 위치 장치(Locator)
  - 절대좌표 또는 상대좌표. 마우스, 키보드의 화살표 키, 트랙 볼
- 획 입력 장치(Stroke)
  - 일련의 연속 좌표. 마우스, 태블릿 커서.
- 문자열 장치(String)
  - 문자열 키보드
- 수량 장치(Valuator)
  - 회전각, 크기조절 비율 등 스칼라 값. 키보드, 마우스, 다이얼
- 선택 장치(Choice)
  - 메뉴, 서브메뉴, 메뉴옵션 선택. 마우스, 키보드, 터치 패널, 음성
- 지적 장치(Pick)
  - 물체를 선택. 마우스나 터치 패널

43

## 위치 장치(Locator Device)



- 위치의 좌표 ( $x, y$ )를 그래픽스 응용 프로그램으로 입력
  - 사용자에게 현재의 위치를 피드백 시켜주는 것이 중요
  - 물리적 입력 장치: 마우스, 트랙볼, 태블릿, 터치스크린 등
  - **절대좌표**를 출력할 수도 있고 **상대좌표**를 출력할 수도 있다.
- **절대좌표** 입력 장치
  - 태블릿, 터치스크린 등
- **상대좌표** 입력 장치
  - 마우스, 트랙볼, 터치패드, 조이스틱, 데이터글러브 등

44

## 획 입력 장치(Stroke Device)



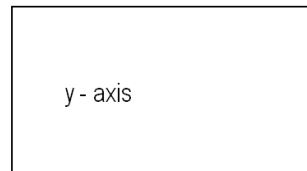
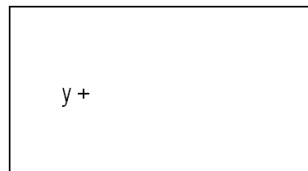
- 태블릿 펜, 마우스 커서 (드래그)  
• 연속적인 좌표의 입력을 위한 장치
- 위치 장치의 동작을 연속적으로 처리함으로써 구현
- 대응되는 물리적인 장치들도 위치 장치와 동일

45

## 문자열 장치 (String Device)



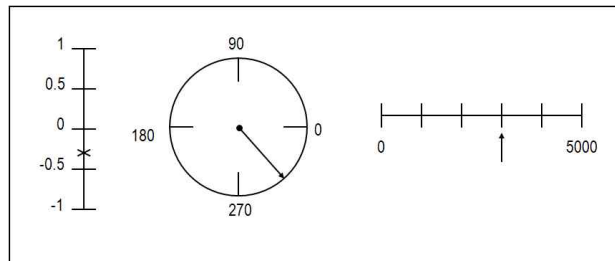
- 텍스트 문자열을 입력하기 위한 장치
- 대표적 장치: 키보드
  - 터치펜으로 자판의 글자를 선택
  - 음성 인식이나 영상 인식을 이용해 문자열을 판단하여 입력
- 모바일 기기의 급격한 보급으로 인해 이러한 모바일 장치에서 문자열을 효율적으로 입력하는 방법에 대한 요구가 증가



46

## 수량 장치(Valuator Device)

- 정해진 범위 내의 수량 값을 지정하기 위한 장치
  - 예) 전류나 전압을 조절할 수 있도록 만든 슬라이드 형이나 다이얼 형의 전위차계(Potentiometer) 등
- 소프트웨어
  - 슬라이드 바나 다이얼 형태의 사용자 인터페이스



47

## 선택 장치 (Choice Device)

메뉴 선택

- 메뉴에서와 같이 나열된 여러 항목 중 하나를 선택
- 대표적인 장치: PFK(Programmable Function Key)
  - 이러한 키를 이용하여 컴퓨터나 장치의 특정 기능을 선택
- 소프트웨어
  - 메뉴나 컨트롤키도 선택장치로 볼 수 있다.

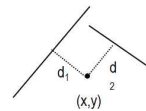
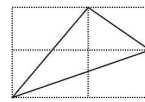
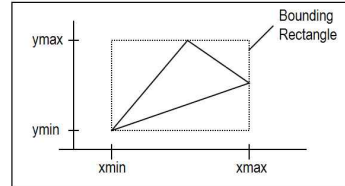
48



## 지적 장치 (Pick Device)

객체 선택

- 그림의 구성요소를 직접 선택하기 위한 장치
  - 예) 라이트 펜: 화면상에 디스플레이 된 그림의 구성요소로부터 나오는 빛을 감지하고 그것이 어느 객체에 해당하는지를 인식하여 그 객체를 선택
  - 마우스도 좋은 지적 장치
- 복잡한 알고리즘이 필요
  - 마우스 클릭 → 어떤 물체 선택?
    - 가까운 물체
    - 맨 위 layer의 물체
    - 사용자 선택



- 그래픽 편집기에서 물체를 선택할 때 고려할 점?

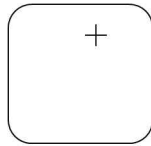
49

## 2.4 상호작용(Interactive) 입력 기법

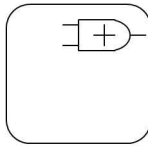
- Positioning
- Constraints
- Grids
- Gravity
- Rubber-band
- Sketching
- Dragging

50

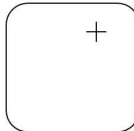
## Positioning 위치시킴



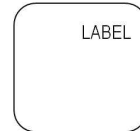
Position Cursor  
and Press Button



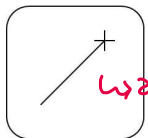
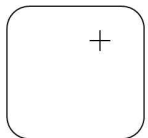
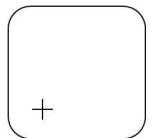
Object Displayed  
at Cursor Position



Press Button to  
Select Text Position



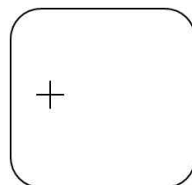
Text Displayed.  
Centered on  
Selected Position



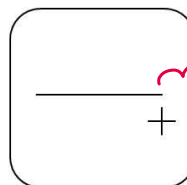
라인을  
positioning

51

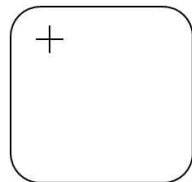
## Constraints 제한



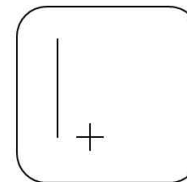
Press Button  
to Select First  
Endpoint



Press Button  
to Select  
Second Endpoint



Press button  
to Select First  
Endpoint

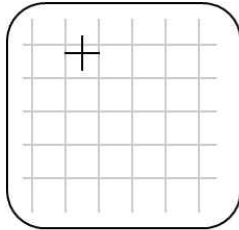


Press Button  
to Select  
Second Endpoint

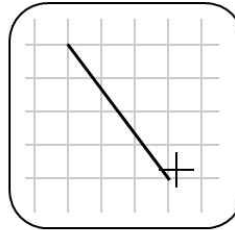
자동으로 수평선  
이것도 맞춰줌

52

## Grids 누름



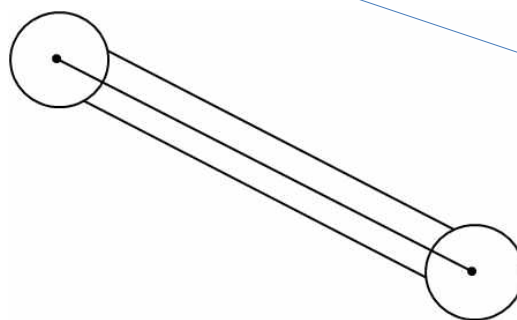
Press Button to  
Select First Endpoint



Press Button to  
Select Second Endpoint

53

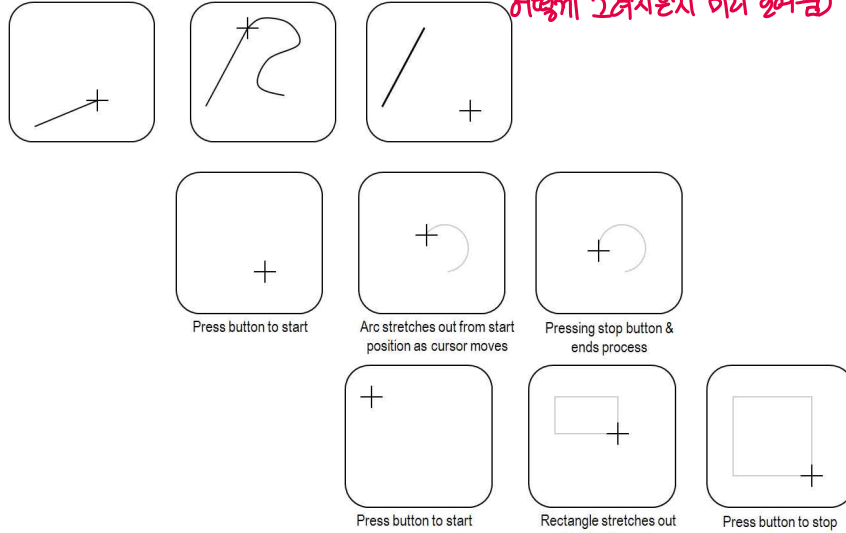
## Gravity 선을 선택할 때 근처로만 가도 대충 선이 붙어됨



54

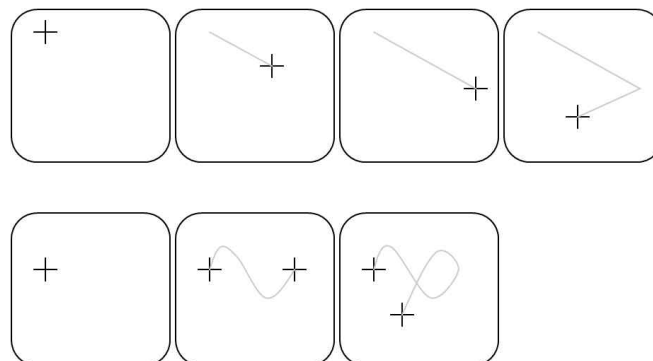
## Rubber-band

선을 그릴 때 피드백 (마우스 클릭을 놓지 않았을 때  
드래그된 상태에서 다음선이  
어떻게 그려지는지 미리 보여줌)

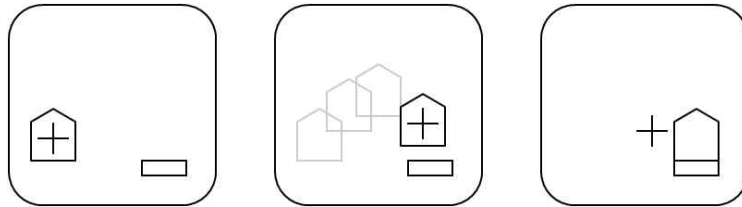


## Sketching

우클릭 전까지 라인이 계속이어서 그려짐



## Dragging 마우스로 객체를 잡고 움직임



57

## 2.5 사용자 인터페이스 구현



- 입력 모드
- 이벤트 기반 프로그래밍
- 선분 그리기를 위한 프로그램의 구조

58

## 입력 모드

- 입력장치로 응용프로그램에 값을 입력하는 과정을 메저와 트리거로 나눌 수 있음
  - 메저(Measure):** 입력장치가 응용프로그램에게 전달하는 값 *ex) dir*
  - 트리거(Trigger):** 전달하라는 신호를 의미한다. *ex) Enter키*

```

D:\저서-그래픽스\교재코드\05장-모델변환>dir + Enter
D 드라이브의 볼륨: DATA
볼륨 일련 번호: FA08-FD09

D:\저서-그래픽스\교재코드\05장-모델변환 디렉터리

2018-12-04 오후 04:57 <DIR> .
2018-12-04 오후 04:57 <DIR> ..
2018-12-04 오후 04:59 <DIR> 2,065 05장-모델변환.sln
2018-12-04 오후 04:59 <DIR> AnalogClock
2018-12-04 오후 04:59 <DIR> Debug
2018-12-04 오후 05:02 <DIR> MakeRobot

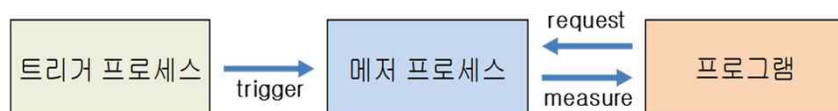
1개 파일 2,065 바이트
52개 디렉터리 588,031,770,624 바이트 남음

D:\저서-그래픽스\교재코드\05장-모델변환>
    
```

59

## (1) Request mode

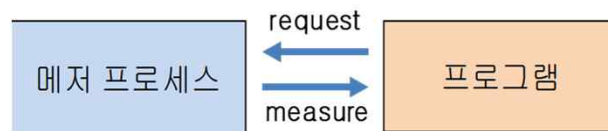
- 입력을 위해 항상 트리거를 요구하는 모드  $\Rightarrow$  `scanf("")`;
  - 예) C언어의 표준 입력함수
    - 이 모드의 입력을 받기 전까지 모든 처리가 멈춤
    - `getchar()` 함수가 호출되면 어떤 키를 누르고 엔터가 입력되어야 프로그램이 다음으로 넘어감.
    - 이때 엔터키가 트리거이고, 그 전에 입력한 키가 메저로 프로그램에 전달된다.



60

## (2) Sample Mode

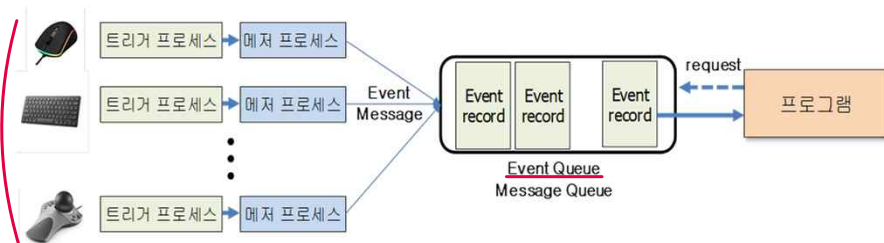
- 사용자의 트리거가 필요X는 모드
  - 사용자가 메저를 요구하면 무조건 현재의 메저를 바로 제공
  - 메저는 항상 준비가 된 상태 **마우스 움직임**
  - 예) **현재의 마우스 좌표** 등



61

## ★ 이번학기 계속 할 것 (3) Event Mode

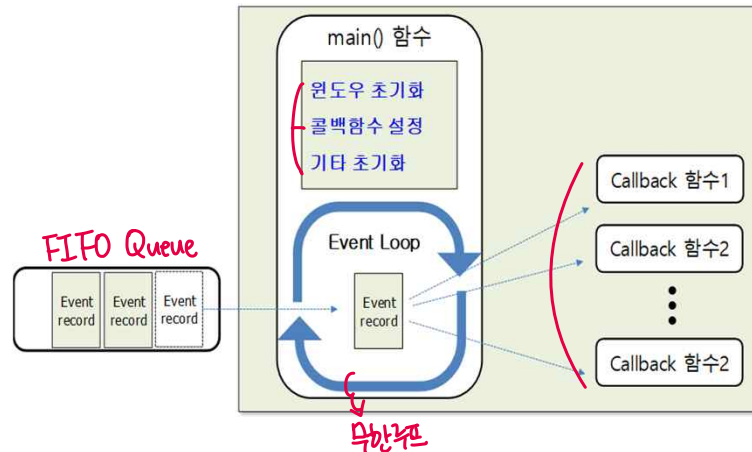
- 다양한 입력장치를 사용해 이벤트(event) 발생
- **이벤트 큐(event queue)**
  - 응용 프로그램은 가장 먼저 들어온 이벤트를 꺼내고 이벤트의 종류에 따라 관련 처리기(event handler)에서 처리
  - **이벤트 구동 시스템(event-driven system)**
    - MS 윈도우 운영체제



62

## 이벤트 기반 프로그래밍

- main()함수, callback 함수



63

- main()함수
  - 사용할 윈도우를 만들고 각종 초기화 작업을 한다.
  - 콜백함수를 설정
    - "어떤 이벤트가 오면 어느 함수를 실행해라"는 대응 관계
    - 이벤트 처리 함수는 추가로 구현
      - 키보드 입력, 마우스 클릭, 마우스 이동
      - 화면 갱신, 화면 크기 및 위치 변경
      - 메뉴 상태, 아이들(idle) 및 타이머 이벤트 등
  - 초기화가 끝나면 메시지 루프를 반복
    - 이벤트 큐에서 먼저 입력된 이벤트를 꺼내고
    - 종류에 따라 등록된 콜백함수를 호출하는 과정.
- 윈도우 환경의 구현방법
  - Windows API, MFC(Microsoft Foundation Class), GLUT

64



## 선분 그리기를 위한 프로그램의 구조



- 다음과 같은 동작으로 두 좌표를 입력할 수 있다.
  - 화면에서 마우스로 한 점을 누름(BUTTON\_DOWN)→한 좌표
  - 이 상태에서 다른 점으로 **마우스를 이동**
  - 현재 위치에서 마우스를 뽐(BUTTON\_UP)→ 다음 좌표

65

### 프로그램 2.1 선분 그리기를 위한 응용 프로그램 유사코드

```
1. 선분의 두 끝점 저장을 위한 변수 p1, p2 선언
2.
3. 화면갱신처리_함수() {
4.     전역 변수를 이용해 화면을 그리는 코드;
5.     배경색상으로 화면을 지운 후 선분 (p1,p2)를 그리는 코드;
6. }
7. 마우스클릭처리_함수(현재좌표) {
8.     만약 마우스 버튼이 눌러지는 이벤트이면
9.         좌표를 p1에 저장;
10.    만약 마우스가 떴지는 이벤트이면
11.        좌표를 p2에 저장;
12.    화면갱신 이벤트를 이벤트 큐에 추가하여 화면이 다시 그려지도록 함
13. }
14. // 주 함수: Program Entry Point (main() 또는 WinMain() 등)
15. 주함수() {
16.    윈도우의 크기, 위치 등 다양한 속성을 설정;
17.    현재 설정대로 새로운 윈도우를 만들;
18.    배경색상 등 다양한 초기화 코드 추가;
19.    콜백등록함수_마우스클릭( 마우스클릭처리_함수 );
20.    콜백등록함수_화면갱신( 화면갱신처리_함수 );
21.    기타 콜백함수 등록;
22.
23.    메시지_루프();
24. }
```

66

## 2장 연습문제



- [Lab 2-1] 사각형과 원을 그리는 과정을 프로그램 2.1과 같이 유사 코드로 구현해 보라. 선분과 유사한 방법으로 구현할 수 있을 것이다.
- [Lab 2-2] 선이나 사각형, 원 등 다양한 그래픽 객체를 그리는 편집기를 구현하려면 현재 모드가 어떤 객체를 그리는 모드인지가 결정되어 있어야 한다. 이를 결정하는 방법에는 어떤 것이 있는지를 MS사의 파워포인트 프로그램을 통해 알아보라.
- [Lab 2-3] 다각형을 그리기 위한 인터페이스를 설계해 보라. 다각형을 위해서는 다수의 좌표가 입력될 수 있어야 하며, 좌표의 입력이 끝난 것을 알릴 수 있어야 할 것이다.
- [Lab 2-4] 화면에 그려진 그래픽 객체를 이동하거나 회전 또는 신축(크기 변환)하기 위한 인터페이스를 설계해 보라.

67



**감사합니다!**

68