



1과목 · 전자계산기 일반

핵심 18.상시, 15.상시, 14.상시, 12.상시, 09.7, 08.2, 06.10, 05.7, 04.4, 02.7

001 컴퓨터의 특징

| | |
|------|--|
| 대용량성 | 많은 양의 데이터를 처리 및 보관 |
| 범용성 | 컴퓨터를 여러 가지 용도로 사용 |
| 호환성 | 하나의 하드웨어나 소프트웨어를 컴퓨터의 기종에 관계 없이 여러 컴퓨터에서 사용 가능 |
| 정확성 | 사용자의 요구 기능을 충족시키는 정도 |
| 신뢰성 | 주어진 환경에서 고장 없이 담당 기능을 원활하게 수행하는 정도 |

핵심 18.상시, 17.상시, 16.상시, 15.상시, 14.상시, 13.상시, 12.상시, 13.1, 11.7, 11.4, 10.10, 10.7, 10.1, 09.9, ...

002 중앙처리장치(CPU)

중앙처리장치는 사람의 두뇌와 같이 컴퓨터 시스템에 부착된 모든 장치의 동작을 제어하고 명령을 실행하는 장치로, 제어장치, 연산장치, 레지스터로 구성된다.

제어장치

- 컴퓨터에 있는 모든 장치들의 동작을 지시하고 제어하는 장치이다.
- 제어장치는 프로그램 카운터(PC), 명령어 레지스터(IR), 부호기(제어신호 발생기), 명령어 해독기, 번지 해독기 등으로 구성되어 있다.
- 명령어의 실행은 인출(Fetch) 단계, 간접(Indirect) 단계, 실행(Execute) 단계, 인터럽트(Interrupt) 단계를 거친다.

잠깐만요! 인출(Fetch) 단계

주기억장치로부터 명령어를 읽어 들여 해석하는 단계

- 제어장치의 명령 실행 순서
 - 프로그램 카운터에 저장된 주소(값)를 번지 레지스터에 옮긴다.
 - 명령어를 주기억장치로부터 인출한다.
 - 프로그램 카운터를 증가시킨다.
 - 명령 코드를 명령 레지스터로 옮긴다.
 - 명령 레지스터의 내용을 해독하여 실행한다.

연산장치

- 제어장치의 명령에 따라 실제로 연산을 수행하는 장치이다.
- 연산장치는 가산기, 감산기, 누산기(AC, Accumulator), 보수기, 데이터 레지스터, 오버플로우 검출기, 시프트 레지스터 등으로 구성되어 있다.

잠깐만요! 연산 수행을 나타내는 단위

- LIPS : 초당 1개의 연산 수행
- KIPS : 초당 1,000개의 연산 수행
- MIPS : 초당 1,000,000개의 연산 수행

마이크로프로세서

제어장치, 연산장치, 레지스터가 하나의 대규모 집적회로 칩(IC)에 내장된 장치로, 개인용 컴퓨터(PC)에서 중앙처리장치로 사용되며, 클럭 주파수와 내부 버스의 폭으로 성능을 평가한다.

- 마이크로프로세서는 설계 방식에 따라 RISC와 CISC로 구분된다.
- RISC 방식은 명령어의 종류가 적어 전력 소비가 적고, 속도가 빠르지만 복잡한 연산을 수행하기 위해 명령어들을 반복·조합해서 사용해야 하므로 레지스터를 많이 필요로 하고, 프로그램도 복잡하다.
- CISC 방식은 명령어의 종류가 많아 전력 소비가 많고 명령어 설계가 어려워 고가이지만 레지스터를 적게 사용하므로 프로그램이 간단하다.
- 다음은 RISC와 CISC의 차이점이다.

| 구분 | RISC | CISC |
|---------|------------|-------------|
| 명령어 | 적음 | 많음 |
| 명령어의 길이 | 고정 | 가변 |
| 실행 사이클 | 단일 | 다중 |
| 주소 지정 | 간단 | 복잡 |
| 레지스터 | 많음 | 적음 |
| 전력 소모 | 적음 | 많음 |
| 처리 속도 | 빠름 | 느림 |
| 프로그래밍 | 복잡함 | 간단함 |
| 용도 | 서버, 워크스테이션 | 개인용 컴퓨터(PC) |



핵심

18.상시, 17.상시, 16.상시, 15.상시, 14.상시, 13.상시, 12.상시, 11.7, 11.4, 11.3, 11.2, 10.10, 10.7, 10.3, 09.9, ...

003 레지스터

- CPU 내부에서 처리할 명령어나 연산의 중간 결과값 등을 일시적으로 기억하는 임시 기억 장소이다.
- 연산에 사용되는 데이터 및 연산의 중간 결과를 레지스터에 저장하는 이유는 연산 속도를 향상시키기 위해서다.
- 레지스터는 메모리 중에서 속도가 가장 빠르다.
- 레지스터에 새로운 데이터가 전송되면 기존에 있던 내용은 지워지고 새로운 내용만 기억된다.

| | |
|---|--|
| 프로그램 카운터, 프로그램 계수기(PC; Program Counter) | 다음 번에 실행할 명령어의 번지를 기억하는 레지스터 |
| 명령 레지스터(IR; Instruction Register) | 실행중인 명령의 내용을 기억하는 레지스터 |
| 누산기 (AC; Accumulator) | 연산된 결과를 일시적으로 저장하는 레지스터로 연산의 중심이 됨 |
| <ul style="list-style-type: none"> • 상태 레지스터 (Status Register), • PSW(Program Status Word Register) • 플래그 레지스터 (Flag Register) | <ul style="list-style-type: none"> • CPU에서 명령이 실행되는 순서를 제어하거나 특정 프로그램에 관련된 컴퓨터 시스템의 상태를 나타내고 유지하기 위한 제어 워드로서 실행중인 CPU의 상황을 PSW라고 함 • 오버플로, 언더플로, 자리올림, 인터럽트 등의 PSW를 저장하고 있는 레지스터 • 제어장치와 연산장치의 실행 순서를 제어하기 위해 사용되는 레지스터 |
| 메모리 주소 레지스터 (MAR; Memory Address Register) | 기억장치를 출입하는 데이터의 번지를 기억하는 레지스터 |
| 메모리 버퍼 레지스터 (MBR; Memory Buffer Register) | <ul style="list-style-type: none"> • 기억장치를 출입하는 데이터가 잠시 기억되는 레지스터 • 입·출력장치의 동작 속도와 전자계산기 내부의 동작 속도를 맞추는 데 사용되는 레지스터 • 버퍼 레지스터라고도 함 |
| 인덱스 레지스터 (Index Register) | 주소의 변경, 서브루틴 연결 및 프로그램에서의 반복 연산의 횟수를 세는 레지스터 |
| 데이터 레지스터 (Data Register) | 연산에 사용될 데이터를 기억하는 레지스터 |
| 시프트 레지스터 (Shift Register) | 클럭 펄스(Clock Pulse)에 의해 기억되는 내용을 왼쪽 또는 오른쪽으로 1비트씩 자리를 이동시키는 레지스터 |

핵심

14.상시, 12.상시, 08.7, 06.7, 06.4, 04.4, 04.2, 03.7

004 기본적인 논리함수

AND

- 입력 정보의 값이 모두 1일 때만 결과가 1이 된다.
- 입력되는 값이 A, B라면 A AND B 또는 A · B로 표현한다.

OR

- 입력 정보의 값 중 1개라도 1이면 결과가 1이 된다.
- 입력되는 값이 A, B라면 A OR B 또는 A+B로 표현한다.

NOT

- 입력되는 정보의 반대 값이 출력되며 입력되는 값이 항상 1개이다.
- 입력되는 값이 A라면, NOT A 또는 A' 또는 \bar{A} 로 표현한다.

핵심

17.상시, 15.상시, 14.상시, 13.상시, 12.상시, 11.10, 11.2, 09.9, 09.7, ...

005 불 대수의 기본 공식

- 멱등법칙 : $A + A = A$, $A \cdot A = A$
- 보수법칙 : $A + \bar{A} = 1$, $A \cdot \bar{A} = 0$
- 항등법칙 : $A + 0 = A$, $A + 1 = 1$, $A \cdot 0 = 0$, $A \cdot 1 = A$
- 드모르강 법칙 : $\overline{A + B} = \bar{A} \cdot \bar{B}$, $\overline{A \cdot B} = \bar{A} + \bar{B}$
- 교환법칙 : $A + B = B + A$, $A \cdot B = B \cdot A$
- 결합법칙 : $A + (B + C) = (A + B) + C$,
 $A \cdot (B \cdot C) = (A \cdot B) \cdot C$
- 분배법칙 : $A \cdot (B + C) = A \cdot B + A \cdot C$,
 $A + B \cdot C = (A + B) \cdot (A + C)$

핵심

18.상시, 17.상시, 16.상시, 15.상시, 14.상시, 13.상시, 11.7, 10.10, 10.7, ...

006 논리식의 간소화

불 대수의 기본 공식을 이용해 간소화한다.

- ① 합의 곱 표현을 곱의 합 표현으로 변환한다.
- ② 공통 인수를 뽑아 묶는다.
- ③ 멱등법칙, 보수법칙, 항등법칙 등의 기본 공식 형태로 유도하여 줄여 나간다.

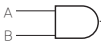
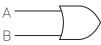
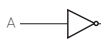
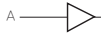

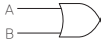
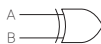

예제 다음 불 함수를 간략화하시오.

- $A + A \cdot B = A \cdot (1 + B) = A \cdot 1 = A$



- $A(A+B) = A \cdot A + A \cdot B = A + A \cdot B = A \cdot (1+B) = A \cdot 1 = A$
- $A + \bar{A} \cdot B = (A + \bar{A})(A + B) = 1 \cdot (A + B) = A + B$
- $A(\bar{A} + B) = A \cdot \bar{A} + A \cdot B = 0 + A \cdot B = A \cdot B$

핵심 18.상시, 17.상시, 16.상시, 15.상시, 14.상시, 13.상시, 12.상시, 11.10, 11.7, ...
007 논리 게이트

| 게이트 | 기 호 | 의 미 | 진리표 | 논리식 | | | | | | | | | | | | | | | |
|----------|---|---------------------------------|--|-----|---|---|---|---|---|---------------------------|---|---|---|---|---|---|---|---|---|
| AND |  | 입력 신호가 모두 1일 때 1 출력 | <table><tr><th>A</th><th>B</th><th>Y</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table> | A | B | Y | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | $Y = A \cdot B$ $Y = AB$ |
| A | B | Y | | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | | | | | | | | | | | | | | | | | |
| 0 | 1 | 0 | | | | | | | | | | | | | | | | | |
| 1 | 0 | 0 | | | | | | | | | | | | | | | | | |
| 1 | 1 | 1 | | | | | | | | | | | | | | | | | |
| OR |  | 입력 신호 중 1개만 1이어도 1 출력 | <table><tr><th>A</th><th>B</th><th>Y</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table> | A | B | Y | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | $Y = A + B$ |
| A | B | Y | | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | | | | | | | | | | | | | | | | | |
| 0 | 1 | 1 | | | | | | | | | | | | | | | | | |
| 1 | 0 | 1 | | | | | | | | | | | | | | | | | |
| 1 | 1 | 1 | | | | | | | | | | | | | | | | | |
| NOT, 인버터 |  | 입력된 정보를 반대로 변환하여 출력 | <table><tr><th>A</th><th>Y</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table> | A | Y | 0 | 1 | 1 | 0 | $Y = A'$ $Y = \bar{A}$ | | | | | | | | | |
| A | Y | | | | | | | | | | | | | | | | | | |
| 0 | 1 | | | | | | | | | | | | | | | | | | |
| 1 | 0 | | | | | | | | | | | | | | | | | | |
| BUFFER |  | 입력된 정보를 그대로 출력 | <table><tr><th>A</th><th>Y</th></tr><tr><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td></tr></table> | A | Y | 0 | 0 | 1 | 1 | $Y = A$ | | | | | | | | | |
| A | Y | | | | | | | | | | | | | | | | | | |
| 0 | 0 | | | | | | | | | | | | | | | | | | |
| 1 | 1 | | | | | | | | | | | | | | | | | | |
| NAND |  | NOT + AND, 즉 AND의 부정 | <table><tr><th>A</th><th>B</th><th>Y</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table> | A | B | Y | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | $Y = \overline{A \cdot B}$ $Y = \overline{AB}$ |
| A | B | Y | | | | | | | | | | | | | | | | | |
| 0 | 0 | 1 | | | | | | | | | | | | | | | | | |
| 0 | 1 | 1 | | | | | | | | | | | | | | | | | |
| 1 | 0 | 1 | | | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | | | | | | | | | | | | | | | | | |
| NOR |  | NOT + OR, 즉 OR의 부정 | <table><tr><th>A</th><th>B</th><th>Y</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table> | A | B | Y | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | $Y = \overline{A + B}$ |
| A | B | Y | | | | | | | | | | | | | | | | | |
| 0 | 0 | 1 | | | | | | | | | | | | | | | | | |
| 0 | 1 | 0 | | | | | | | | | | | | | | | | | |
| 1 | 0 | 0 | | | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | | | | | | | | | | | | | | | | | |
| XOR |  | 입력되는 값이 모두 같으면 0, 1개라도 다르면 1 출력 | <table><tr><th>A</th><th>B</th><th>Y</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table> | A | B | Y | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | $Y = A \oplus B$ $Y = \bar{A}B + A\bar{B}$ |
| A | B | Y | | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | | | | | | | | | | | | | | | | | |
| 0 | 1 | 1 | | | | | | | | | | | | | | | | | |
| 1 | 0 | 1 | | | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | | | | | | | | | | | | | | | | | |
| XNOR |  | NOT + XOR, 즉 XOR의 부정 | <table><tr><th>A</th><th>B</th><th>Y</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table> | A | B | Y | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | $Y = A \odot B$ $Y = \overline{A \oplus B}$ |
| A | B | Y | | | | | | | | | | | | | | | | | |
| 0 | 0 | 1 | | | | | | | | | | | | | | | | | |
| 0 | 1 | 0 | | | | | | | | | | | | | | | | | |
| 1 | 0 | 0 | | | | | | | | | | | | | | | | | |
| 1 | 1 | 1 | | | | | | | | | | | | | | | | | |

잠깐만요! AND와 OR 회로

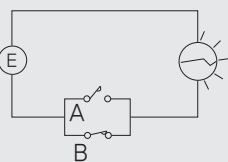
AND 회로($X = A \cdot B$)

2개의 입력 스위치가 ON ($A=1, B=1$)이 될때에만 불이 켜짐



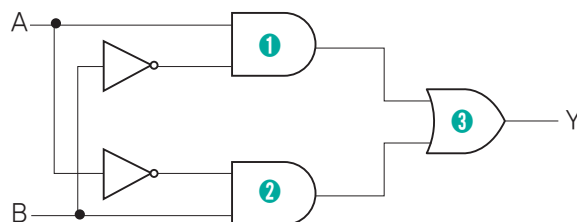
OR 회로($X = A + B$)

둘 중 1개 이상의 입력 스위치가 ON이 되면 불이 켜짐



핵심 18.상시, 17.상시, 16.상시, 15.상시, 14.상시, 13.상시, 12.상시, 11.10, 11.7, ...
008 논리회로의 이해

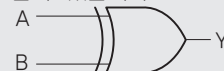
예제 1 다음 논리회로를 논리식으로 표현하시오.



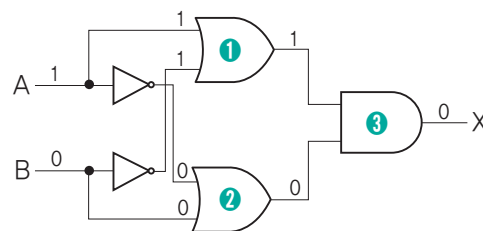
각각의 논리 게이트를 분리하여 논리식으로 표현한 후 1개의 논리식으로 합쳐 나갑니다.

- ① = $A \cdot \bar{B}$
- ② = $\bar{A} \cdot B$
- ③ = ① + ② = $A \cdot \bar{B} + \bar{A} \cdot B = A \oplus B$

그러므로 위의 논리회로는 아래와 같은 XOR 회로로 간략하게 표현할 수 있습니다.



예제 2 $A = 1, B = 0$ 입력 시 출력 X의 값은?



이런 문제는 위 그림과 같이 입력되는 값을 게이트 순서대로 대입한 후 출력을 구해서 계산해도 되고, 다음과 같이 게이트별로 분리해서 계산해도 됩니다.

- ① = $A + \bar{B} = 1 + \bar{0} = 1 + 1 = 1$
- ② = $\bar{A} + B = \bar{1} + 0 = 0 + 0 = 0$
- ③ = ① · ② = $1 \cdot 0 = 0$

∴ $X = 0$

핵심 18.상시, 17.상시, 16.상시, 15.상시, 14.상시, 13.상시, 12.상시, 11.10, 11.7, ...
009 반가산기(HA; Half Adder)

- 반가산기는 1Bit짜리 2진수 2개를 덧셈한 합(S)과 리올림수(C)를 구하는 회로이다.

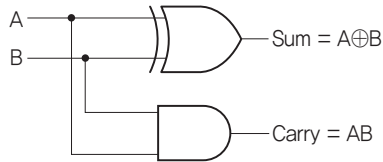
• 진리표

| A | B | S | C |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |



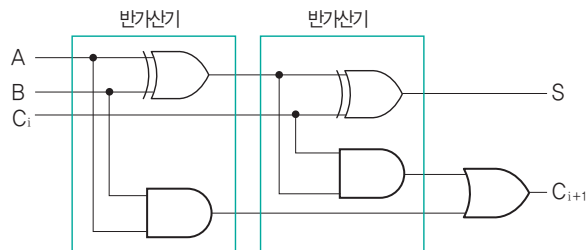
- 논리식 : $C = A \cdot B$, $S = \bar{A} \cdot B + A \cdot \bar{B} = A \oplus B$

논리회로



핵심 010 전가산기(FA; Full Adder)

- 전가산기는 뒷자리에서 올라온 자리올림수(C_i)를 포함하여 1Bit 크기의 2진수 3자리를 더하여 합(S_i)과 자리올림수(C_{i+1})를 구하는 회로이다.
- 전가산기는 2개의 반가산기(HA)와 1개의 OR 게이트로 구성된다.
- 전가산기는 3개의 입력선과 2개의 출력선을 갖는다.

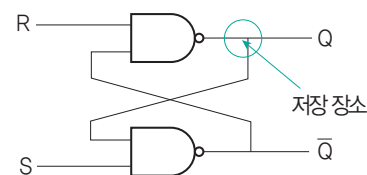


핵심 011 기타 조합논리회로

| | |
|------------------------------|--|
| 디코더(Decoder) | <ul style="list-style-type: none"> • n개의 입력선으로 입력된 값을 2^n개의 출력선으로 번역하는 회로로, 주로 AND 게이트로 구성됨 • 2진 코드를 다른 코드로 바꿀 때 사용함 |
| 인코더(Encoder) | <ul style="list-style-type: none"> • 인코더는 디코더의 반대 기능을 함 • 2^n개의 입력선으로 입력된 값을 n개의 출력선으로 코드화해서 출력하는 회로 • 특정 값을 여러 자리인 2진수로 변환하거나 특정 장치로부터 보내오는 신호를 여러 개의 2진 신호로 바꾸어 변환시키는 장치 • 특정한 장치에서 사용되는 정보를 다른 곳으로 전송하기 위하여 일정한 규칙에 따라 암호로 변환하는 장치 |
| 멀티플렉서(MUX; Multiplexer) | 2^n 의 입력선 중 1개를 선택하여 그 선으로부터 입력되는 값을 1개의 출력선으로 출력시키는 회로 |
| 디멀티플렉서(DeMUX; DeMultiplexer) | 1개의 입력선으로 들어오는 정보를 2^n 개의 출력선 중 1개를 선택하여 출력하는 회로 |

핵심 012 플립플롭(FF, Flip-Flop)

- 플립플롭은 전원이 공급되고 있는 한, 상태의 변화를 위한 신호가 발생할 때까지 현재의 상태를 그대로 유지하는 논리회로이다.
- 플립플롭 1개가 1Bit를 저장할 수 있다.
- 기본적인 플립플롭은 2개의 NAND 또는 NOR 게이트를 이용하여 구성한다.



- 플립플롭의 종류 : RS-FF, D-FF, JK-FF, T-FF

RS 플립플롭(Reset-Set FF)

플립플롭의 기본으로, S와 R선의 입력을 조절하여 임의의 Bit 값을 그대로 유지시키거나 무조건 0 또는 1의 값을 기억시키기 위해서 사용되는 플립플롭이다.

| S | R | $Q_{(t+1)}$ | 상태 | 암기 |
|---|---|-------------|----------|-------------|
| 0 | 0 | $Q_{(t)}$ | 상태 변화 없음 | 무(상태 변화 없음) |
| 0 | 1 | 0 | Reset | 공(항상 0) |
| 1 | 0 | 1 | Set | 일(항상 1) |
| 1 | 1 | 동작 안 됨 | 동작 안 됨 | 불(불가) |

JK 플립플롭

- RS FF에서 $S=R=1$ 일 때 동작되지 않는 결점을 보완한 플립플롭이다.
- 다른 모든 플립플롭의 기능을 대용할 수 있으므로 응용 범위가 넓고 집적회로화 되어 가장 널리 사용된다.

| J | K | $Q_{(t+1)}$ | 상태 | 암기 |
|---|---|-----------------|----------|-------------|
| 0 | 0 | $Q_{(t)}$ | 상태 변화 없음 | 무(상태 변화 없음) |
| 0 | 1 | 0 | Reset | 공(항상 0) |
| 1 | 0 | 1 | set | 일(항상 1) |
| 1 | 1 | $\bar{Q}_{(t)}$ | 반전 | 보(보수) |

T 플립플롭

- JK FF의 두 입력선을 묶어서 1개의 입력선으로 구성한 플립플롭이다.
- $T=0$ 인 경우는 변화가 없고, $T=1$ 인 경우에 현재의 상



태를 토글(Toggle)시킨다. 즉 원 상태와 보수 상태의 2가지 상태로만 서로 전환된다.

- 누를 때마다 ON, OFF가 교차되는 스위치를 만들고자 할 때 사용된다.

| T | $Q_{(t+1)}$ | 상태 |
|---|----------------------|----------|
| 0 | $Q_{(t)}$ | 상태 변화 없음 |
| 1 | $\overline{Q_{(t)}}$ | 반전 |

핵심 013 자료 구성의 단위

| | |
|------------------|---|
| 비트(Bit) | <ul style="list-style-type: none"> • 자료(정보) 표현의 최소 단위 • 2가지 상태(0과 1)를 표시하는 2진수 1자리 |
| 니블(Nibble) | <ul style="list-style-type: none"> • 4개의 비트(Bit)가 모여 1개의 니블(Nibble)을 구성함 • 4비트로 구성되며 16진수 1자리를 표현하기에 적합함 |
| 바이트(Byte) | <ul style="list-style-type: none"> • 문자를 표현하는 최소 단위로, 8개의 비트(Bit)가 모여 1Byte를 구성함 • 1Byte는 $256(2^8)$가지의 정보를 표현할 수 있음 • 주소 지칭의 단위로 사용됨 |
| 워드(Word) | <ul style="list-style-type: none"> • CPU가 한 번에 처리할 수 있는 명령 단위 • 반워드(Half Word) : 2Byte • 풀워드(Full Word) : 4Byte • 더블워드(Double Word) : 8Byte |
| 필드(Field) | <ul style="list-style-type: none"> • 파일 구성의 최소 단위 • 의미 있는 정보를 표현하는 최소 단위 |
| 레코드(Record) | <ul style="list-style-type: none"> • 하나 이상의 관련된 필드가 모여서 구성됨 • 컴퓨터 내부의 자료 처리 단위로서, 일반적으로 레코드는 논리 레코드(Logical Record)를 의미함 |
| 블록(Block) | <ul style="list-style-type: none"> • 1개 이상의 논리 레코드가 모여서 구성됨 • 각종 저장매체와의 입·출력 단위이며 물리 레코드(Physical Record)라고 함 |
| 파일(File) | 프로그램 구성의 기본 단위로, 여러 레코드가 모여서 구성됨 |
| 데이터베이스(Database) | <ul style="list-style-type: none"> • 여러 개의 관련된 파일(File)의 집합 • 관계형, 계층형, 망형 데이터베이스가 있음 |

핵심 014 진법 변환

10진수를 2진수, 8진수, 16진수로 변환

- 정수 부분 : 10진수의 값을 변환할 진수로 나누어 더 이상 나뉘지 않을 때까지 나누고, 몫을 제외한 나머지를 역순으로 표시함
- 소수 부분 : 10진수의 값에 변환할 진수를 곱한 후 결과의 정수 부분만을 차례대로 표기하되, 소수 부분이 0 또는 반복되는 수가 나올 때까지 곱하기를 반복함

예제 1 $(47.625)_{10}$ 를 2진수, 8진수, 16진수로 변환하기

• 정수 부분

| 2진수 | 8진수 | 16진수 |
|---|---|--|
| $\begin{array}{r} 2 \overline{) 47} \\ 2 \overline{) 23} \dots 1 \\ 2 \overline{) 11} \dots 1 \\ 2 \overline{) 5} \dots 1 \\ 2 \overline{) 2} \dots 1 \\ 1 \dots 0 \end{array}$ | $\begin{array}{r} 8 \overline{) 47} \\ 5 \dots 7 \end{array}$ | $\begin{array}{r} 16 \overline{) 47} \\ 2 \dots 15(F) \end{array}$ |
| $(47)_{10} = (101111)_2$ | $(47)_{10} = (57)_8$ | $(47)_{10} = (2F)_{16}$ |

• 소수 부분

| 2진수 | 8진수 | 16진수 |
|---|--|---|
| $\begin{array}{r} 0.625 \\ \times 2 \\ \hline 1.250 \\ \times 2 \\ \hline 0.50 \\ \times 2 \\ \hline 1.0 \end{array}$ | $\begin{array}{r} 0.625 \\ \times 8 \\ \hline 5.000 \end{array}$ | $\begin{array}{r} 0.625 \\ \times 16 \\ \hline 10(A).000 \end{array}$ |
| $(0.625)_{10} = (0.101)_2$ | $(0.625)_{10} = (0.5)_8$ | $(0.625)_{10} = (0.A)_{16}$ |
| $(47.625)_{10} \rightarrow (101111.101)_2$ | $(47.625)_{10} \rightarrow (57.5)_8$ | $(47.625)_{10} \rightarrow (2F.A)_{16}$ |

2진수, 8진수, 16진수를 10진수로 변환

정수 부분과 소수 부분을 나누어서 변환하려는 각 진수의 자릿값과 자리의 지수승을 곱한 결과값을 모두 더하여 계산한다.

예제 2 $(101111.101)_2$ 을 10진수로 변환하기

$$\begin{array}{cccccccc} (1 & 0 & 1 & 1 & 1 & 1 & . & 1 & 0 & 1)_2 \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ 2^5 & 2^4 & 2^3 & 2^2 & 2^1 & 2^0 & . & 2^{-1} & 2^{-2} & 2^{-3} \\ = 32 + 0 + 8 + 4 + 2 + 1 & . & 0.5 + 0 + 0.125 = 47.625 \end{array}$$

예제 3 $(57.5)_8$ 를 10진수로 변환하기

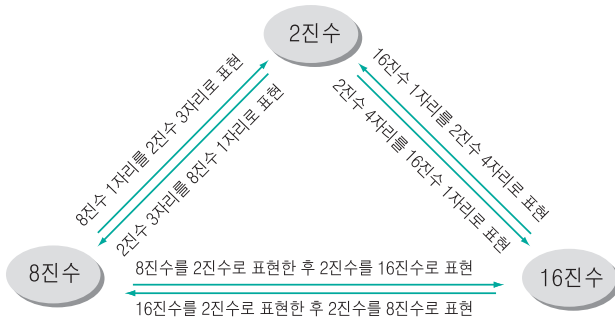
$$\begin{array}{ccc} (5 & 7 & . & 5)_8 \\ \times & \times & \times & \\ 8^1 & 8^0 & . & 8^{-1} \\ = 40 + 7 & . & 0.625 = 47.625 \end{array}$$



예제 4 $(2F.A)_{16}$ 를 10진수로 변환하기

$$\begin{array}{r} (2 \quad F \quad . \quad A)_{16} \\ \times \quad \times \quad \times \\ 16^1 \quad 16^0 \quad 16^{-1} \\ = 32 + 15 \quad . \quad 0.625 = 47.625 \end{array}$$

2진수, 8진수, 16진수 상호 변환



- 2진수를 8진수로 : 정수 부분은 소수점을 기준으로 왼쪽 방향으로 3자리씩, 소수 부분은 소수점을 기준으로 오른쪽 방향으로 3자리씩 묶어서 변환함
- 2진수를 16진수로 : 정수 부분은 소수점을 기준으로 왼쪽 방향으로 4자리씩, 소수 부분은 소수점을 기준으로 오른쪽 방향으로 4자리씩 묶어서 변환함
- 8진수, 16진수를 2진수로 : 8진수 1비트는 2진수 3비트로, 16진수 1비트는 2진수 4비트로 풀어서 변환함

예제 5 2진수를 8진수로 변환 2진수를 16진수로 변환

$$\begin{array}{ccccccc} 111001011.10101010 & 11001011.1010101000 \\ \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} \\ 7 & 1 & 3 & . & 5 & 2 & 12(C) \quad 11(B) \quad . \quad 10(A) \quad 8 \end{array}$$

자릿수를 맞추기 위해 0으로 채워짐

$$(111001011.10101010)_2 \rightarrow (713.52)_8 \quad (11001011.10101010)_2 \rightarrow (CB.A8)_{16}$$

※ 소수 부분의 자릿수가 부족할 경우 0으로 부족한 부분을 채워서 자리(8진수 3자리, 16진수 4자리) 수를 맞춥니다.

예제 6 8진수 또는 16진수를 2진수로 변환하기

$$\begin{array}{ccccccc} 7 & 1 & 3 & . & 5 & 2 & 12(C) \quad 11(B) \quad . \quad 10(A) \quad 8 \\ \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} \\ 111001011.10101010 & 11001011.1010101000 & & & & & \\ (713.52)_8 \rightarrow (111001011.10101010)_2 & (CB.A8)_{16} \rightarrow (11001011.10101010)_2 & & & & & \end{array}$$

※ 2진수를 8진수(3개씩 묶기), 16진수(4개씩 묶기)로 변환한 방법의 반대 방법으로 8진수(3개씩 풀기), 16진수(4개씩 풀기)를 2진수로 변환할 수 있습니다.

8진수, 16진수 상호 변환

- 8진수를 16진수로 : 8진수를 2진수로 변환한 후 2진수를 16진수로 변환함
- 16진수를 8진수로 : 16진수를 2진수로 변환한 후 2진수를 8진수로 변환함

예제 7 8진수 670.325를 16진수로 변환하기

① 8진수를 우선 2진수로 변환합니다.

$$\begin{array}{ccccccc} (6 & 7 & 0 & . & 3 & 2 & 5)_8 \\ \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} \\ (110 & 111 & 000 & . & 011 & 010 & 101)_2 \end{array}$$

② 2진수를 16진수로 변환합니다. 소수점을 기준으로 정수 부분은 왼쪽 방향으로, 소수 부분은 오른쪽 방향으로 4자리씩 묶어줍니다. 소수 이하 자릿수가 모자랄 경우 0으로 채워서 자리를 맞춥니다.

$$\begin{array}{ccccccc} (1 & 1011 & 1000 & . & 0110 & 1010 & 1000)_2 \\ \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} \\ (1 & 11(B) & 8 & . & 6 & 10(A) & 8)_{16} \\ \therefore (670.325)_8 \rightarrow (1B8.6A8)_{16} \end{array}$$

예제 8 16진수 F3.9D를 8진수로 변환하기

① 16진수를 우선 2진수로 변환합니다.

$$\begin{array}{ccccccc} (F & 3 & . & 9 & D)_{16} \\ \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} \\ (1111 & 0011 & . & 1001 & 1101)_2 \end{array}$$

② 2진수를 8진수로 변환합니다. 소수점을 기준으로 정수 부분은 왼쪽 방향으로, 소수 부분은 오른쪽 방향으로 3자리씩 묶어줍니다. 소수 이하 자릿수가 모자랄 경우 0으로 채워서 자리를 맞춥니다.

$$\begin{array}{ccccccc} (11 & 110 & 011 & . & 100 & 111 & 010)_2 \\ \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} & \underbrace{\hspace{1cm}} \\ (3 & 6 & 3 & . & 4 & 7 & 2)_8 \\ \therefore (F3.9D)_{16} \rightarrow (363.472)_8 \end{array}$$

핵심

17.상시, 16.상시, 14.상시, 12.상시, 11.10, 11.2, 10.10, 08.7, 08.2, 06.4, ...

015 보수

- 컴퓨터가 기본적으로 수행하는 가산을 이용하여 뺄셈을 수행하기 위해 사용한다.
- 1의 보수 구하기 : 주어진 각 자릿값을 0일 때는 1로, 1일 때는 0으로 변환
 10101 의 1의 보수 $\rightarrow 01010$
※ 2진수를 1의 보수로 구하는 게이트는 NOT이다.



• 2의 보수 구하기

- 방법 1 : 1의 보수를 구한 뒤 결과값에 1을 더함

$$\begin{array}{r} 10101 \text{의 } 2\text{의 보수} \\ 01010 \leftarrow 1\text{의 보수} \\ + \quad 1 \\ \hline 01011 \end{array}$$

- 방법 2 : 소수점의 위치에서 왼쪽 방향으로 첫 번째 1이 나올때까지는 그냥 쓰고 나머지는 반대로 씀

$$10101100 \text{의 } 2\text{의 보수} \rightarrow \boxed{01010} \boxed{100} \begin{array}{l} \text{그대로} \\ \text{쓰니다.} \end{array}$$

핵심 18.상시, 16.상시, 14.상시, 12.상시, 10.10, 08.7, 06.1, 04.2, 03.3, 00.10, 00.8

016 2진 연산

- 2진 정수 데이터의 표현에 사용된다.
- 표현할 수 있는 범위는 작지만 연산속도는 빠르다.
- 데이터 표현 시 첫 번째 비트를 부호 비트로 하여 양수는 0, 음수는 1로 표시한다.

| 표현 방식 | -10 | +10 |
|--------------|---|---|
| 부호화 절대치 | <ul style="list-style-type: none"> • 양수 10을 2진수로 표현한 후 부호 비트만 1로 바꿈 • 0000 1010 → 1000 1010 | |
| 부호화 1의 보수 방식 | <ul style="list-style-type: none"> • 양수 10을 2진수로 표현한 후 1의 보수를 구함 • 0000 1010 → 1111 0101 | <ul style="list-style-type: none"> • 양수는 표현 방식이 모두 같음 • 2진수로 바꾸어 줌 0000 1010 |
| 부호화 2의 보수 방식 | <ul style="list-style-type: none"> • 양수 10을 2진수로 표현한 후 2의 보수를 구함 • 0000 1010 → 1111 0110 | |

핵심 18.상시, 17.상시, 16.상시, 15.상시, 14.상시, 13.상시, 12.상시, 11.9, 11.7, 11.4, 10.10, 09.7, 08.1, 08.3, 07.7, ...

017 자료의 외부적 표현

| | |
|-----------------------|--|
| BCD(2진화 10진 코드) | <ul style="list-style-type: none"> • 6Bit 코드로 1개의 문자를 2개의 Zone 비트와 4개의 Digit 비트로 표현함 • $2^6 = 64$가지의 문자를 표현할 수 있음 |
| ASCII 코드 | <ul style="list-style-type: none"> • 7Bit 코드로 1개의 문자를 3개의 Zone 비트와 4개의 Digit 비트로 표현함 • $2^7 = 128$가지의 문자를 표현할 수 있음 • 통신 제어용 및 마이크로 컴퓨터의 기본 코드로 사용함 |
| EBCDIC(확장 2진화 10진 코드) | <ul style="list-style-type: none"> • 8Bit 코드로 1개의 문자를 4개의 Zone 비트와 4개의 Digit 비트로 표현함 • $2^8 = 256$가지의 문자를 표현할 수 있음 • 대형 기종의 컴퓨터에서 사용함 |

핵심

18.상시, 16.상시, 15.상시, 13.상시, 12.상시, 10.1, 09.3, 07.9, 07.7, 07.4, 06.10, 06.4, 05.10, 04.10, ...

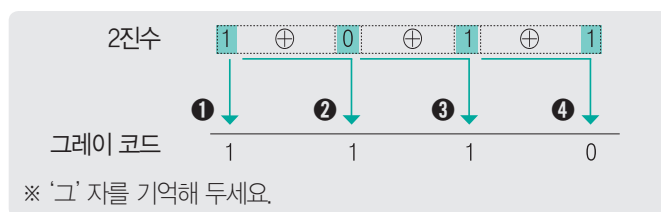
018 그레이 코드(Gray Code)

- BCD 코드의 인접하는 비트를 XOR 연산하여 만든 코드이다.
- 1Bit만 변화시켜 다음 수치로 증가시키기 때문에 하드웨어적인 오류가 적어 하드웨어 동작을 제어하기 적합하다.
- 연속되는 2개의 숫자를 표현한 코드에서 한 비트를 변경하면 새로운 코드가 되기 때문에 아날로그-디지털(A/D) 변환, 데이터 전송에 주로 사용한다.

2진수를 그레이 코드(Gray Code)로 변환하는 방법

- ① 첫 번째 그레이 비트(Gray Bit)는 2진수 비트를 그대로 내려쓴다.
- ② 두 번째 그레이 비트부터는 변경할 2진수의 해당 번째 비트와 그 왼쪽의 비트를 XOR 연산하여 쓴다.

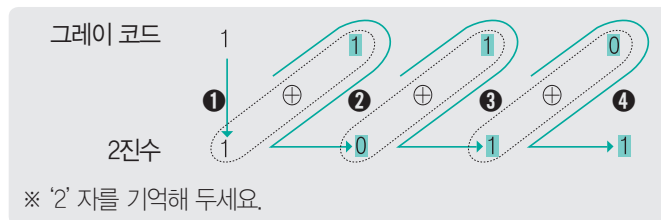
예제 1 2진수 1011을 그레이 코드(Gray Code)로 변환하시오.



그레이 코드(Gray Code)를 2진수로 변환하는 방법

- ① 첫 번째 2진수 비트는 그레이 비트(Gray Bit)를 그대로 내려쓴다.
- ② 두 번째 2진수 비트부터는 왼쪽에 구해 놓은 2진수 비트와 변경할 그레이 코드의 해당 번째 비트를 XOR 연산하여 쓴다.

예제 2 그레이 코드(Gray Code) 1110을 2진수로 변환하시오.





핵심 019 패리티 검사 코드 / 해밍 코드

패리티 검사 코드

- 코드의 오류를 검사하기 위해서 데이터 비트 외에 1Bit의 패리티 체크 비트를 추가하는 것으로 1Bit의 오류만 검출할 수 있다.
- 1의 개수에 따라 짝수(Even, 우수) 패리티와 홀수(Odd, 기수) 패리티 방법이 있다.

해밍 코드

- 오류를 스스로 검출하여 교정이 가능한 코드이다.
- 해밍 코드는 2Bit의 오류를 검출할 수 있고, 1Bit의 오류를 교정할 수 있다.
- 해밍 코드 중 1, 2, 4, 8, 16, ... 2ⁿ 번째 비트는 오류 검출을 위한 패리티 비트이다.

핵심 020 명령어의 구성

컴퓨터에서 실행되는 명령어는 크게 연산자가 표시되는 연산자(Operation Code)부와 연산의 수행에 필요한 자료의 정보가 표시되는 자료부(Operand)로 구성된다.

| 연산자(Operation Code)부 | 자료(Operand)부 |
|-----------------------------|--|
| 연산자 (Operation Code)부 | <ul style="list-style-type: none"> • 수행해야 할 동작에 맞는 연산자를 표시하며, 흔히 OP-Code부라고 함 • 연산자부의 크기(비트 수)는 표현할 수 있는 명령의 종류를 나타내는 것으로, nBit일 때 최대 2ⁿ개의 명령어를 사용할 수 있음 예 연산자부가 5Bit라면 최대 2⁵=32개의 명령어(연산자)를 사용할 수 있음 • 명령어 형식, 동작코드, 데이터 종류 등을 표시함 |
| 자료(Operand)부 | <ul style="list-style-type: none"> • 실제 데이터에 대한 정보를 표시하는 부분 • 주로 연산에 사용할 자료의 주소를 표시하므로 주소(Address)부라고 부르기도 함 • 기억장소의 주소, 레지스터 번호, 사용할 데이터, 명령어 순서 등을 표시함 |

핵심 021 연산자(Operation Code)의 기능

연산자(Operation Code)의 기능에는 함수 연산, 자료 전달, 제어, 입·출력 기능이 있다.

| | |
|----------|--|
| 함수 연산 기능 | 수치적인 산술 연산과 비수치적인 논리 연산이 있음 • 산술 연산 : ADD, SUB, MUL, DIV, 산술 Shift 등 • 논리 연산 : NOT, AND, OR, XOR, 논리적 Shift, Rotate, Complement, Clear 등 |
| 자료 전달 기능 | CPU와 기억장치 사이에서 정보를 교환하는 기능 |
| 제어 기능 | 명령의 실행 순서를 변경시킬 때 사용 |
| 입·출력 기능 | CPU와 I/O 장치, 또는 메모리와 I/O 장치 사이에서 자료를 전달하는 기능 |

잠깐만요 ! 피연산자의 수에 따른 연산자의 분류

- 단항 연산자(Unary Operator) : NOT, Complement, Shift, Rotate, MOVE 등
- 이항 연산자(Binary Operator) : 사칙연산, AND, OR, XOR, XNOR 등

핵심 022 연산

| | |
|----------|---|
| AND | <ul style="list-style-type: none"> • 특정 문자 또는 특정 비트를 삭제(Clear)시키는 연산으로 Masking 연산이라고도 함 • 삭제할 부분의 비트를 0과 AND시켜서 삭제하는 데, 대응시키는 0인 비트를 Mask Bit라고 함 |
| OR | <ul style="list-style-type: none"> • 특정 문자를 삽입하거나 특정 비트에 1을 세트시키는 연산으로 Selective Set 연산이라고도 함 • 삽입하거나 세트시킬 비트에 삽입할 문자 코드 또는 1을 OR 연산시킴 |
| XOR | <ul style="list-style-type: none"> • 2개의 데이터를 비교하거나, 특정 비트를 반전시킬 때 사용함 • 2개의 데이터를 XOR 연산하여 결과에 1비트라도 1이 있으면 서로 다른 데이터임 • 반전시킬 때는 반전시킬 비트와 1을 XOR시킴 |
| NOT | 각 비트의 값을 반전시키는 연산으로 보수를 구할 때 사용함 |
| 논리 Shift | <ul style="list-style-type: none"> • 왼쪽 또는 오른쪽으로 1bit씩 자리를 이동시키는 연산으로 데이터의 직렬 전송(Serial Transfer)에 사용함 • 삽입되는 자리는 무조건 0임 |
| Rotate | <ul style="list-style-type: none"> • Shift에서 밀려 나가는 비트의 값을 반대편 값으로 입력하는 연산임 • 문자 위치를 변환할 때 사용함 |
| MOVE | 레지스터에 기억된 자료를 그대로 다른 레지스터로 옮길 때 사용함 |



핵심 18.상시, 17.상시, 16.상시, 15.상시, 14.상시, 13.상시, 12.상시, 11.7, 11.4, 10.3, 10.1, 09.9, 09.7, ...

023 명령어 형식

| | |
|---------|---|
| 3주소 명령어 | <ul style="list-style-type: none"> Operand부가 3개로 구성되는 명령어 형식 여러 개의 범용 레지스터(GPR)를 가진 컴퓨터에서 사용함 연산 후 입력 자료가 변하지 않고 보존됨 명령어 한 개의 길이가 길어짐 |
| 2주소 명령어 | <ul style="list-style-type: none"> Operand부가 2개로 구성되는 가장 일반적으로 사용되는 명령어 형식 여러 개의 범용 레지스터를 가진 컴퓨터에서 사용함 연산의 결과는 주로 Operand 1에 저장되므로 Operand 1에 있던 원래의 자료가 파괴됨 |
| 1주소 명령어 | <ul style="list-style-type: none"> Operand부가 한 개로 구성되어 있음 AC(Accumulator, 누산기)를 이용하여 명령어를 처리함 |
| 0주소 명령어 | <ul style="list-style-type: none"> Operand부 없이 연산자(OP-Code)부만으로 구성됨 주소의 사용 없이 스택에 연산자와 피연산자를 넣었다 꺼내어 연산한 후 결과를 다시 스택에 넣으면서 연산하기 때문에 원래의 자료가 남지 않음 |

잠깐만요! 스택(Stack)

- 자료의 삽입 · 삭제 작업이 한쪽 방향에서만 가능할 수 있도록 할 당한 메모리의 일부입니다.
- 가장 나중에 삽입된 자료를 가장 먼저 삭제하는 후입선출(LIFO: Last In First Out) 방식으로 자료를 처리합니다.
- 스택에 자료를 삽입하는 명령은 Push이고, 스택에서 자료를 삭제하는 명령은 Pop입니다.

핵심 18.상시, 17.상시, 16.상시, 15.상시, 14.상시, 13.상시, 12.상시, 11.9, 11.7, 11.4, 11.2, 10.10, 10.7, 10.1, ...

024 주소지정방식

접근 방식에 따른 분류

| | |
|------------------------------------|--|
| 암시적 주소지정방식 (Implied Mode) | 주소를 지정하는 필드가 없는 0번지 명령어에서 Stack의 Top 포인터가 가리키는 Operand를 암시하여 이용함 |
| 즉치(즉시)적 주소지정방식 (Immediate Mode) | <ul style="list-style-type: none"> 명령어 자체에 오퍼랜드(실제 데이터)를 가지고 있는 방식 별도의 기억 장소를 액세스하지 않고 CPU에서 곧바로 자료를 이용할 수 있어서 실행 속도가 빠르다는 장점이 있음 |
| 직접 주소지정방식 (Direct Mode) | 명령의 주소부(Operand)에 있는 값이 실제 데이터가 기억된 번지(Address)를 지정하는 방식 |
| 간접 주소지정방식 (Indirect Mode) | <ul style="list-style-type: none"> 명령어의 주소부(Operand)가 지정하는 곳에 있는 값이 실제 데이터를 기억하는 또 다른 메모리의 번지를 지정하는 방식 최소한 주기억장치를 2회 이상 접근하여 데이터가 있는 기억장소에 도달함 메모리 참조 횟수가 2회 이상으로 가장 많이 필요함 |

계산에 의한 주소지정방식

- Operand부와 CPU의 특정 레지스터의 값이 더해져서 유효 주소를 계산하는 방식이다.
- 계산에 의한 주소지정방식은 대부분 주소의 일부분을 생략하는 약식주소이다.
- 사용하는 레지스터의 종류에 따라 상대, 베이스, 인덱스 주소지정방식으로 구분한다.

| 주소지정방식 | 설명 |
|-----------------------------------|---|
| 상대주소 (Relative Mode) | <ul style="list-style-type: none"> 유효주소 : 명령어의 주소 부분 + PC 명령어 자신의 기억장소를 기준으로 하여 데이터의 위치를 지정하는 방식 |
| 베이스 레지스터 (Base Register Mode) | <ul style="list-style-type: none"> 유효주소 : 명령어의 주소 부분 + Base Register 프로그램을 재배치(Relocation)할 때 이용함 |
| 인덱스 레지스터 (Index Register Mode) | <ul style="list-style-type: none"> 명령어의 주소 부분 + Index Register 주소지정에 2개의 레지스터를 사용하는 방식으로 순차적인 주소지정에 유리함 |

실제 기억공간 주소에 따른 구분

| | |
|----------------------------|---|
| 절대주소 (Absolute Address) | 임의의 기억장소에 대한 실제 주소(유효주소)로 기억장치 맨 처음부터 1Byte마다 0, 1, 2, 3, ...의 순서로 16진수의 번호가 차례대로 지정됨 |
| 상대주소 (Relative Address) | 기준주소를 기준으로 상대적으로 얼마만큼 떨어져 있는지 변위(Displacement; Offset)로 표현하는 주소로서, 상대주소는 절대주소로 변환해야만 실제 데이터에 접근할 수 있음 |

핵심 17.상시, 16.상시, 15.상시, 14.상시, 13.상시, 12.상시, 11.10, 10.10, 10.7, 10.3, 10.1, 09.9, 09.7, 09.3, 08.2, ...

025 채널

- 채널(Channel)은 주변장치에 대한 제어 권한을 CPU로부터 넘겨받아 CPU 대신 입 · 출력을 관리한다.
- 주기억장치와 입 · 출력장치의 중간에 위치한다.
- 입 · 출력장치와 CPU의 속도차로 인한 단점을 해결한다.
- CPU의 제어장치로부터 입 · 출력 전송을 위한 명령어를 받으면 CPU와는 독립적으로 동작하여 입 · 출력을 완료한다.
- 주기억장치에 기억되어 있는 채널 프로그램의 수행과 자료의 전송을 위하여 주기억장치에 직접 접근한다.



채널의 종류

| | |
|--------------------------------|---|
| Selector Channel (선택 채널) | <ul style="list-style-type: none"> 고속 입 · 출력장치(자기 디스크, 자기 테이프, 자기 드럼)와 입 · 출력하기 위해 사용함 특정한 한 개의 장치를 독점하여 입 · 출력함 |
| Multiplexer Channel (다중 채널) | <ul style="list-style-type: none"> 저속 입 · 출력장치(카드리더, 프린터)를 제어하는 채널로, Byte Multiplexer Channel이라고도 함 동시에 여러 개의 입 · 출력장치를 제어함 |
| Block Multiplexer Channel | <ul style="list-style-type: none"> 고속 입 · 출력장치를 제어하는 채널 동시에 여러 개의 입 · 출력장치를 제어함 |

026 DMA(직접 메모리 접근)/스풀링/버퍼링

DMA(직접 메모리 접근)

- DMA(Direct Memory Access)는 CPU의 참여 없이 입 · 출력장치와 메모리가 직접 데이터를 주고받는 것을 말한다.
- DMA 제어기는 작업이 끝나면 CPU에게 인터럽트 신호를 보내 작업이 종료됐음을 알린다.
- DMA 방식을 이용하면 CPU는 입 · 출력 작업에 참여하지 않고 다음 명령을 계속 처리하므로, 시스템의 전반적인 속도가 향상된다.

스풀링 / 버퍼링

- 스풀링(Spooling)은 프린터와 같은 저속의 입 · 출력장치와 고속의 CPU 간의 속도 차이를 해소하기 위한 방법이다.
- 버퍼링(Buffering)도 입 · 출력장치와 CPU 간의 속도 차이를 해결하기 위해 사용한다는 점에서 스푼링과 목적은 같지만 다음과 같은 점이 다르다.

| 구분 | 버퍼링 | 스풀링 |
|-------|--------|---------|
| 저장 장치 | 주 기억장치 | 보조 기억장치 |
| 운영 방식 | 단일 작업 | 다중 작업 |
| 구현 방식 | 하드웨어 | 소프트웨어 |

027 인터럽트

- 인터럽트(Interrupt)는 프로그램을 실행하는 도중에 예기치 않은 상황이 발생할 경우, 현재 실행중인 작업을 즉시 중단하고 발생한 상황을 우선 처리한 후 실행중이던 작업으로 복귀하여 계속 처리하는 것을 말한다. 일명 ‘끼어들기’라고도 한다.
- 인터럽트는 외부 인터럽트, 내부 인터럽트, 소프트웨어 인터럽트로 분류하는데, 외부나 내부 인터럽트는 CPU의 하드웨어에서의 신호에 의해 발생하고 소프트웨어 인터럽트는 명령어의 수행에 의해 발생한다.
- 동시에 하나 이상의 인터럽트가 발생하였을 때 먼저 서비스할 장치를 결정하는 인터럽트 우선순위 판별 방법에는 소프트웨어적인 방법으로 폴링, 하드웨어적인 방법으로 데이지 체인과 병렬 우선 처리 방식이 있다.
- 인터럽트가 발생하면 인터럽트를 처리하고 되돌아올 주소가 스택(Stack)에 저장된다.
- 하드웨어적인 인터럽트

| | |
|-------------------|-------------------------------------|
| 정전 인터럽트 | 정전이 되거나 전원의 이상에 의해 발생 |
| 기계 착오 인터럽트 | CPU의 기능적인 오류 동작에 의해 발생 |
| 외부(External) 인터럽트 | 입 · 출력장치, 타이밍장치, 오퍼레이터의 조작에 의해 발생 |
| 입 · 출력(I/O) 인터럽트 | 입 · 출력 조작의 종료 및 입 · 출력의 착오 등에 의해 발생 |

- 소프트웨어 인터럽트 : 프로그램 처리 중 명령의 요청에 의해 발생하는 것으로, 가장 대표적인 형태는 감시 프로그램을 호출하는 SVC(SuperVisor Call) 인터럽트가 있음

028 ROM(롬)

- 기억된 내용을 읽을 수만 있는 기억장치로서 일반적으로 쓰기는 불가능하다.
- 전원이 꺼져도 기억된 내용이 지워지지 않는 비휘발성 메모리이다.



• 롬(ROM)의 종류와 특징

| | |
|----------|---|
| Mask ROM | 제조 과정에서 미리 내용을 기억시킨 ROM으로, 사용자가 임의로 수정할 수 없음 |
| PROM | 특수 프로그램을 이용하여 한 번만 기록할 수 있으며, 이후엔 읽기만 가능한 ROM |
| EPROM | 자외선을 이용하여 기록된 내용을 여러 번 수정하거나 새로운 내용을 기록할 수 있는 ROM |
| EEPROM | 전기적인 방법을 이용하여 기록된 내용을 여러 번 수정하거나 새로운 내용을 기록할 수 있는 ROM |

▶▶▶ ① 주기억장치

중앙처리장치(CPU)가 직접 접근하여 데이터를 처리할 수 있는 기억장치(Memory)로, 현재 수행되는 프로그램과 데이터를 저장하고 주소(Address)에 의해 저장된 위치를 구분합니다.

핵심 18.상시, 17.상시, 16.상시, 14.상시, 13.상시, 11.9, 11.7, 11.2, 10.1, 08.7, 07.7, 04.10, 99.3

029 기타 메모리

| | |
|--------------------------------|---|
| 캐시 메모리 (Cache Memory) | <ul style="list-style-type: none"> 중앙처리장치(CPU)와 주기억장치 사이에 위치하여 컴퓨터의 처리 속도를 향상시키는 역할을 함 캐시 메모리로는 접근 속도가 빠른 정적 램(SRAM)을 사용함 |
| 가상 메모리 (Virtual Memory) | 보조기억장치(하드디스크)의 일부를 주기억장치처럼 사용하는 메모리 기법으로, 주기억장치보다 큰 프로그램을 불러와 실행해야 할 때 유용하게 사용됨 |
| 플래시 메모리 (Flash Memory) | <ul style="list-style-type: none"> EEPROM의 일종으로 비휘발성 메모리 MP3 플레이어, 개인용 정보단말기, 휴대전화, 디지털 카메라 등에 널리 사용됨 |
| 연관 메모리 (Associative Memory) | 주소를 참조하여 데이터를 읽어오는 방식이 아니라 저장된 내용의 일부를 이용하여 기억장치에 접근하여 데이터를 읽어오는 기억장치 |
| 버퍼 메모리 (Buffer Memory) | <ul style="list-style-type: none"> 두 개의 장치가 데이터를 주고받을 때 두 장치 간의 속도 차이를 해결하기 위해 중간에 데이터를 임시로 저장해 두는 공간으로, 키보드 버퍼, 프린터 버퍼 등이 있음 캐시 메모리도 일종의 버퍼임 |

핵심 17.상시, 15.상시, 13.상시, 12.상시, 11.2, 09.9, 07.4, 06.1, 04.2, 03.7, 02.7, 02.4, 02.1, 01.10, 01.4, 00.10, ...

030 보조기억장치 - 자기 테이프

- 자기 테이프에는 주소의 개념이 없고, 처음부터 차례대로 처리하는 순차처리(SASD; Sequential Access Storage Device)만 할 수 있는 대용량 저장 매체이다.
- 가격이 저렴하고 용량이 커서 자료의 백업용으로 많이 사용된다.

• 자기 테이프 관련 용어

- EOT/BOT : 자기 테이프의 시작/끝 위치
- BPI(Byte Per Inch) : 1인치에 기억할 수 있는 바이트 수, 기록 밀도라고 함
- GAP : 레코드와 레코드 사이의 데이터를 기록할 수 없는 공간
- IRG(Inter Record Gap) : 레코드와 레코드 사이의 갭
- IBG(Inter Block Gap) : 블록과 블록 사이의 갭
- Block : 한 개 이상의 논리 레코드의 집합으로 입·출력의 단위이며 물리 레코드라고도 함

블로킹

- 한 개 이상의 논리적 레코드를 묶어서 테이프에 기록하는 방식이다.

비블로킹(Unblocking)

| | | | | | | | | |
|-----|--------|-----|--------|-----|--------|-----|--------|-----|
| IRG | 논리 레코드 | IRG | 논리 레코드 | IRG | 논리 레코드 | IRG | 논리 레코드 | IRG |
|-----|--------|-----|--------|-----|--------|-----|--------|-----|

블로킹(Blocking)

| | | | | | | | | |
|-----|--------|--------|--------|-----|--------|--------|--------|-----|
| IBG | 논리 레코드 | 논리 레코드 | 논리 레코드 | IBG | 논리 레코드 | 논리 레코드 | 논리 레코드 | IBG |
|-----|--------|--------|--------|-----|--------|--------|--------|-----|

- 하나의 블록을 구성하는 논리 레코드의 개수를 블록화 인수(BF; Blocking Factor)라고 한다.
- 블로킹을 하면 블로킹을 하지 않았을 때에 비해 IRG의 수가 줄어들게 되므로 다음과 같은 장점이 있다.
 - 데이터의 처리 속도가 빨라진다.
 - 기억공간의 낭비가 줄어든다.
 - 입·출력 횟수가 감소한다.

핵심 18.상시, 17.상시, 14.상시, 13.상시, 12.상시, 11.9, 08.10, 08.7, 08.3, 07.1, 06.1, 05.4, 05.1, 03.10, 03.3, ...

031 보조기억장치 - 자기 디스크

- 자성 물질을 입힌 금속 원판을 여러 장 겹쳐서 만든 기억 매체로, 용량이 크고 접근 속도가 빠르다.
- 순차, 비순차(직접) 처리가 모두 가능한 DASD(Direct Access Storage Device) 방식으로 데이터를 처리한다.
- 구성 요소 : 읽기/쓰기 헤드(R/W Head), 디스크(Disk), 액세스 암(Access Arm)
- 자기 디스크 관련 용어
 - 트랙(Track) : 회전축(스핀들 모터)을 중심으로 데이터가 기록되는 동심원



- 섹터(Sector) : 트랙을 일정하게 나눈 구간으로 정보 저장 단위의 기본 단위
- 실린더(Cylinder) : 여러 장의 디스크 판에서 같은 위치에 있는 트랙의 모임
- 클러스터(Cluster) : 여러 개의 섹터를 모은 것으로, 운영체제가 관리하는 파일 저장의 기본 단위
- TPI(Tracks Per Inch) : 1인치(Inch)에 기록할 수 있는 트랙의 수로, 디스크의 기록 밀도 단위
- Seek Time(탐색 시간) : 읽기/쓰기 헤드가 지정된 트랙(실린더)에 도달하는 데 걸리는 시간
- Search Time(=Latency Time, 지연 시간) : 읽기/쓰기 헤드가 지정된 트랙(실린더)을 찾은 후 원판이 회전하여 원하는 섹터의 읽기/쓰기가 시작될 때까지의 시간
- Transmission Time(전송 시간) : 읽은 데이터를 주 기억장치로 보내는 데 걸리는 시간
- Access Time(접근 시간) : 데이터를 읽고 쓰는 데 걸리는 시간의 합(Seek Time + Search Time + Transmission Time)

2과목 · 패키지 활용

핵심

18.상시, 17.상시, 16.상시, 15.상시, 14.상시, 13.상시, 12.상시, 11.7, 10.10, 10.1, 09.3, 08.2, 07.7, 07.4, ...

032 데이터베이스

- 특정 조직의 기능을 수행하는 데 필요한 상호 관련된 데이터들의 모임이다.
- 데이터베이스의 장 · 단점

| 장 점 | 단 점 |
|--|---|
| <ul style="list-style-type: none"> • 데이터의 중복성 최소화 • 데이터의 공유 • 데이터의 일관성 유지 • 데이터의 무결성 유지 • 데이터의 논리적 · 물리적 독립성 • 데이터 저장 공간의 절약 • 데이터 보안성 유지 | <ul style="list-style-type: none"> • 전산화 비용 증가 • 데이터 유실 시 파일 회복이 어려움 • 시스템의 복잡화 • 처리 속도가 느림 |

- 데이터베이스의 설계 순서 : 요구 조건 분석 → 개념적 설계 → 논리적 설계 → 물리적 설계
- 데이터베이스 디자인 단계 순서 : 데이터베이스의 목적 정의 → 데이터베이스에서 필요한 테이블 정의 → 테이블에서 필요한 필드 정의 → 테이블 간의 관계 정의

핵심

18.상시, 17.상시, 16.상시, 15.상시, 14.상시, 13.상시, 12.상시, 11.4, 11.2, 10.10, 10.3, 10.1, 09.7, 09.1, ...

033 DBMS(데이터베이스 관리 시스템)

- DBMS(데이터베이스 관리 시스템)는 사용자와 데이터베이스 사이에 위치하여 데이터베이스를 관리하고, 사용자의 요구에 따라 정보를 생성해 주는 소프트웨어를 말한다.
- DBMS의 필수 기능

| | |
|-------|--|
| 정의 기능 | 데이터베이스에 저장될 데이터의 타입과 구조에 대한 정의와 데이터를 이용하는 방식을 정의하는 기능 |
| 조작 기능 | 데이터의 검색, 갱신, 삽입, 삭제 등을 체계적으로 처리하기 위해 데이터 접근 수단을 정의하는 기능 |
| 제어 기능 | 데이터의 정확성과 보안성을 유지하기 위한 무결성, 보안 및 권한 검사, 병행 제어 등의 기능을 정의하는 기능 |

DBMS 운용 시 고려사항

- 다수 사용자의 이용에 따른 시스템의 보안기능을 확보한다.
- 다양한 장애에 대비한 백업 파일을 확보한다.



- 효율적 검색지원을 위하여 데이터 구조의 표준화를 적극 추진한다.
- 효율적 운영 및 성능 최적화를 위한 관련 전문가를 확보해야 한다.

핵심 18.상시, 17.상시, 16.상시, 15.상시, 14.상시, 13.상시, 12.상시, 11.4, 11.2, 10.7, 10.3, 10.1, 09.3, 08.7, ...
034 스키마(외부, 개념, 내부)

- 스키마는 데이터베이스를 구성하는 개체, 속성, 관계 등 구조에 대한 정의와 이에 대한 제약 조건 등을 기술하는 것을 말한다.
- 스키마는 사용자의 관점에 따라 외부 스키마, 개념 스키마, 내부 스키마로 나뉜다.

| | |
|-------------------------------|--|
| 외부 스키마 (External Schema) | 일반 사용자나 응용 프로그래머의 관점에서 본 스키마 |
| 개념 스키마 (Conceptual Schema) | <ul style="list-style-type: none"> • 기관이나 조직체의 관점에서 본 스키마 • 데이터베이스 접근 권한, 보안 정책, 무결성 규칙에 대한 정의를 포함 |
| 내부 스키마 (Internal Schema) | 시스템 프로그래머나 시스템 설계자의 관점에서 본 스키마 |

핵심 18.상시, 17.상시, 15.상시, 14.상시, 12.상시, 11.4, 09.9, 09.7, 08.10, 06.4, 05.1, 03.7, 03.3, 02.1, 00.10
035 데이터베이스 사용자

| | |
|---|---|
| 데이터베이스 관리자 (DBA; Database Administrator) | <ul style="list-style-type: none"> • 데이터베이스 시스템을 관리하고 운영에 관한 모든 것을 책임지는 사람이나 그룹 • 데이터베이스의 스키마 정의 · 생성 · 삭제, 데이터 사전의 유지 관리 및 보안 조치, 저장 구조와 접근방법 선정, 시스템의 성능 분석 및 감시, 무결성 제약 조건 및 데이터 액세스 권한 지정, 시스템 문서화의 표준 지정, 복구 절차와 무결성 유지를 위한 대책 수립 등을 수행함 |
| 응용 프로그래머 | 일반 호스트 언어로 작성된 프로그램에 데이터 조작어(DML)를 삽입하여 만든 응용 프로그램을 통해서 데이터베이스에 접근하는 사람 |
| 일반 사용자 (End User) | 질의어를 사용하여 데이터베이스에 접근하는 사용자들 |

핵심 18.상시, 17.상시, 16.상시, 15.상시, 14.상시, 13.상시, 12.상시, 11.2, ...
036 관계형 데이터베이스의 구성 요소



- 테이블
 - 데이터들을 행과 열로 표현한 것
 - 튜플(레코드)의 집합
 - 관계형 데이터베이스에서는 릴레이션(Relation)이라고 함
 - 릴레이션 스킴(Scheme, 스키마, 구조)은 일정 수의 속성(Attribute)의 집합으로 구성됨
 - 릴레이션 스킴은 시간에 따라 불변의 특성을 가지고, 릴레이션 인스턴스는 동적인 특성을 가짐
- 튜플(Tuple) : 테이블의 행을 구성하는 개체(레코드)
- 속성(Attribute)
 - 테이블의 열을 구성하는 항목(Field)
 - 데이터베이스를 구성하는 가장 작은 단위
 - 개체의 성질이나 특성을 기술함
- 도메인(Domain) : 하나의 속성(Attribute)에서 취할 수 있는 값의 범위(예) 성별의 도메인은 '남', '여'임
- 차수(Degree) : 속성의 개수
- 기수(Cardinality) : 튜플의 개수

잠깐만요 ! 관계

두 개의 테이블에 속하는 원소들을 서로 연관시키기 위하여 하나의 쌍으로 연결하는 방법을 관계라 합니다.

핵심 18.상시, 17.상시, 15.상시, 12.상시, 11.7, 09.9, 09.1, 06.4, 03.10, 01.10, 00.10, 99.10, 99.7
037 키의 개념 및 종류

키는 데이터베이스에서 조건에 만족하는 레코드를 찾거나 순서대로 정렬할 때 기준이 되는 속성(Attribute)을 말한다.



| | |
|------------------------|---|
| 후보키 (Candidate Key) | <ul style="list-style-type: none"> 테이블을 구성하는 속성들 중에서 튜플을 유일하게 식별하기 위해 사용하는 속성들의 부분집합, 즉 기본키로 사용할 수 있는 속성들을 말함 후보키는 테이블에 있는 모든 레코드에 대해서 다음과 같은 성질을 만족해야 함 <ul style="list-style-type: none"> 유일성(Unique) : 하나의 키로 하나의 레코드만을 유일하게 식별할 수 있어야 함 최소성(Minimality) : 모든 레코드들을 유일하게 식별하는 데 꼭 필요한 속성으로만 구성되어야 함 |
| 기본키 (Primary Key) | <ul style="list-style-type: none"> 후보키 중에서 선택한 주키 한 릴레이션에서 특정 레코드를 유일하게 구별할 수 있는 속성 Null 값으로 둘 수 없음 기본키로 정의된 필드(속성)에는 동일한 값이 중복되어 저장될 수 없음 |
| 외래키 (Foreign Key) | 관계를 맺고 있는 테이블 R1, R2에서 테이블 R1이 참조하고 있는 테이블 R2의 기본키와 같은 R1 테이블의 속성을 외래키라고 함 |
| 대체키 (Alternate Key) | 후보키 중 기본키를 제외한 나머지 속성을 말함 |

잠깐만요 ! 널(Null)

데이터베이스에서 널(Null)이란 아직 알려지지 않았거나 모르는 값으로서, 해당 없음 등의 이유로 정보부재를 나타내기 위해 사용하는, 이론적으로 아무것도 없는 값을 의미합니다.

핵심 038 액세스의 기본

- 액세스는 데이터베이스(Database)를 구축하고, 데이터를 분류, 관리, 검색할 수 있으며 다양한 형태로 인쇄할 수 있는 데이터베이스 프로그램(DBMS)이다.
- 프로그래밍 언어를 모르는 사용자라도 각종 마법사와 제공된 기능을 이용하여 쉽게 데이터베이스를 구축하고, 관리할 수 있다.
- 액세스 프로그램의 기본 데이터 파일의 확장자는 액세스 2003 이전 버전은 *.mdb, 액세스 2007 이후 버전은 *.accdb이다.
- 액세스의 개체

| | |
|----------------|--|
| 테이블 (Table) | <ul style="list-style-type: none"> 데이터를 저장하고 관리하는 것으로, 데이터베이스에서 가장 기본이 되는 개체임 서로 다른 종류의 데이터로 저장된 필드를 가진 레코드로 구성되어 있음 |
|----------------|--|

| | |
|-----------------|--|
| 쿼리 (Query) | <ul style="list-style-type: none"> 하나 이상의 테이블로부터 일정한 기준에 따라 데이터를 선택 및 추출하는 방법을 제공함 관련된 여러 테이블을 연결하여 새로운 결과를 추출할 때도 유용하게 사용됨 데이터베이스 내에 저장되어 있는 데이터를 관리하는 명령어들의 집합임 |
| 폼 (Form) | <ul style="list-style-type: none"> 테이블이나 쿼리 데이터의 입 · 출력 화면을 작성하는 개체임 그래픽 화면을 사용한 입 · 출력 틀임 |
| 보고서 (Report) | <ul style="list-style-type: none"> 검색한 자료나 분석 자료의 출력물을 작성하는 개체임 많은 데이터를 분류 · 요약할 수 있으며 자료의 통계 · 분석을 위한 다양한 기능을 제공함 |
| 매크로 (Macro) | <ul style="list-style-type: none"> 반복적이고, 단순한 작업을 자동화하는 개체임 사용 빈도가 높거나 중요한 기능을 미리 매크로로 정의하여 사용함 |
| 모듈 (Module) | <ul style="list-style-type: none"> 복잡한 작업을 위해 VBA(Visual Basic for Applications)로 실제 프로그램을 작성하는 개체임 매크로에 비해 복잡한 작업을 처리하기 위해 프로그램을 직접 작성하는 것 |

핵심

18.상시, 17.상시, 16.상시, 15.상시, 14.상시, 13.상시, 12.상시, 13.4, 13.2, 13.1, 11.9, 11.7, 11.4, 10.10, 10.7, ...

039 SQL의 분류

- SQL은 사용 용도에 따라 DDL(데이터 정의어), DML(데이터 조작어), DCL(데이터 제어어)로 구분된다.
- DDL(데이터 정의어)
 - SCHEMA, DOMAIN, TABLE, VIEW, INDEX를 정의하거나 변경 또는 삭제할 때 사용하는 언어이다.
 - 데이터베이스 관리자나 데이터베이스 설계자가 사용한다.

| 명령어 | 기능 |
|--------|--|
| CREATE | Schema, Domain, Table, View, Index를 정의함 |
| ALTER | Table에 대한 정의를 변경하는 데 사용함 |
| DROP | <ul style="list-style-type: none"> Schema, Domain, Table, View, Index를 삭제함 옵션 <ul style="list-style-type: none"> CASCADE 옵션 : 삭제할 요소를 참조하는 다른 모든 개체를 함께 삭제함 RESTRICT : 삭제할 요소를 다른 개체가 참조중일 때는 삭제를 취소함 |

- DML(데이터 조작어)
 - 데이터베이스 사용자가 응용 프로그램이나 질의어를 통하여 저장된 데이터를 실질적으로 처리하는 데 사용하는 언어이다.



- 데이터베이스 사용자와 데이터베이스 관리 시스템 간의 인터페이스를 제공한다.

| 명령어 | 기능 |
|--------|---|
| SELECT | <ul style="list-style-type: none"> • 테이블에서 조건에 맞는 튜플을 검색함 • 형식 : SELECT ~ FROM ~ WHERE |
| INSERT | <ul style="list-style-type: none"> • 테이블에 새로운 튜플을 삽입함 • 형식 : INSERT INTO ~ VALUE |
| DELETE | <ul style="list-style-type: none"> • 테이블에서 조건에 맞는 튜플을 삭제함 • 형식 : DELETE ~ FROM ~ WHERE |
| UPDATE | <ul style="list-style-type: none"> • 테이블에서 조건에 맞는 튜플의 내용을 변경함 • 형식 : UPDATE ~ SET ~ WHERE |

• DCL(데이터 제어어)

- 데이터의 보안, 무결성, 데이터 회복, 병행 수행 제어 등을 정의하는 데 사용하는 언어이다.
- 데이터베이스 관리자가 데이터 관리를 목적으로 사용한다.

| 명령어 | 기능 |
|----------|---|
| COMMIT | 명령에 의해 수행된 결과를 실제 물리적 디스크로 저장하고, 데이터베이스 조작 작업이 정상적으로 완료되었음을 관리자에게 알려줌 |
| ROLLBACK | 데이터베이스 조작 작업이 비정상적으로 종료되었을 때 원래의 상태로 복구함 |
| GRANT | 데이터베이스 사용자에게 사용 권한을 부여함 |
| REVOKE | 데이터베이스 사용자의 사용 권한을 취소함 |

잠깐만요! 뷰(VIEW)

- 하나 이상의 기본 테이블로부터 유도되어 만들어진 가상 테이블입니다. 즉 기본 테이블은 데이터가 실제로 저장되지만, 이 가상 테이블은 물리적으로 구현되지는 않습니다.
- 필요한 데이터만 정의해서 처리할 수 있기 때문에 관리가 용이하고 명령문이 간단해집니다.
- 액세스에서는 쿼리를 View처럼 사용합니다.

핵심 040 SELECT문

• 기본 구문

```
SELECT [DISTINCT] 필드이름
FROM 테이블이름
[WHERE 조건식];
```

- SQL문에서는 대·소문자를 구분하지 않으며, 마지막에 ‘;’을 입력해 SQL문의 끝임을 알린다.

- DISTINCT : SELECT문에 ‘DISTINCT’를 입력하면 검색의 결과가 중복되는 레코드는 검색 시 한 번만 표시함
- 필드이름 : 테이블의 모든 필드를 검색할 경우에는 필드 이름 대신 ‘*’를 입력하고, 특정 필드들만 검색할 경우 필드와 필드는 쉼표(,)로 구분하여 표시함
- WHERE 조건식 : 조건을 입력하여 특정 조건에 맞는 레코드만 검색할 때 사용함
- 조건 연산자

| | |
|--------|--|
| 비교 연산자 | =, <, >, >=, <= |
| 논리 연산자 | NOT, AND, OR |
| LIKE | 대표 문자를 이용해 지정된 속성의 값이 문자 패턴과 일치하는 튜플만 검색함 |
| IN | 필드의 값이 IN 연산자의 수로 지정된 값과 같은 레코드만 검색하며, OR 연산을 수행한 결과와 같음 |

핵심 18.상시, 17.상시, 16.상시, 15.상시, 13.상시, 12.상시, 11.9, 10.10, 10.1, 07.4, 04.4, 03.3

041 정렬

```
SELECT [DISTINCT] 필드이름
FROM 테이블이름
[WHERE 조건식]
[ORDER BY 필드이름 정렬방식, ...];
```

- ORDER BY문 : 특정 필드를 기준으로 레코드를 정렬하여 검색할 때 사용함
- 정렬 방식 : ‘ASC’와 ‘DESC’가 있으며, ‘ASC’는 오름차순, ‘DESC’는 내림차순을 의미함. 정렬 방식을 지정하지 않으면(생략하면) 기본적으로 오름차순(ASC) 정렬이 수행됨

핵심 14.상시, 13.상시, 12.상시, 06.1, 02.7

042 그룹 지정

```
SELECT [DISTINCT] 필드이름
FROM 테이블이름
[WHERE 조건식]
[GROUP BY 필드이름]
[HAVING 그룹조건식];
```

- GROUP BY절 : 특정 필드를 기준으로 그룹화하여 검색할 때 사용함
- HAVING절 : 그룹에 대한 조건을 지정할 때 사용함



- 일반적으로 GROUP BY는 SUM, AVG, COUNT 같은 그룹 함수와 함께 사용된다.
- 그룹 함수의 종류

| | |
|------------|------------------|
| COUNT(속성명) | 그룹별 튜플 수를 구하는 함수 |
| MAX(속성명) | 그룹별 최대값을 구하는 함수 |
| MIN(속성명) | 그룹별 최소값을 구하는 함수 |
| SUM(속성명) | 그룹별 합계를 구하는 함수 |
| AVG(속성명) | 그룹별 평균을 구하는 함수 |

핵심 043 INSERT, DELETE, UPDATE문

삽입문(INSERT INTO ~)

INSERT INTO 테이블명(속성명1, 속성명2, ...)
VALUES (데이터1, 데이터2, ...);

- 기본 테이블에 새로운 튜플을 삽입할 때 사용한다.
- 대응하는 속성과 데이터는 개수와 데이터 형식이 일치해야 한다.
- 기본 테이블의 모든 속성을 사용할 때는 속성명을 생략할 수 있다.
- SELECT문을 사용하여 다른 테이블의 검색 결과를 삽입할 수 있다.

삭제문(DELETE FROM ~)

DELETE
FROM 테이블명
WHERE 조건;

- 기본 테이블에 있는 튜플(레코드)들 중에서 특정 튜플(레코드)을 삭제시킬 때 사용한다.
- 모든 튜플을 삭제할 때는 WHERE절을 생략한다.
- 모든 튜플을 삭제하더라도 테이블 구조는 남아 있기 때문에 디스크에서 테이블을 완전히 제거하는 DROP과는 다르다.

갱신문(UPDATE ~ SET ~)

UPDATE 테이블명
SET 속성명 = 데이터[, 속성명=데이터]
WHERE 조건;

기본 테이블에 있는 튜플들 중에서 특정 튜플의 내용을 변경시킬 때 사용한다.

핵심 044 스프레드시트의 기초

- 입력 데이터에 대한 수치 계산과 처리 기능, 문서 작성 기능, 그래프 작성 기능, 데이터 관리 업무 등을 효율적으로 수행할 수 있도록 지원하는 응용 프로그램이다.
- 수치 계산과 관련된 업무에서 계산의 어려움과 비효율성을 개선하여 전표의 작성, 처리, 관리를 쉽게 할 수 있도록 한 것이다.
- 엑셀의 확장자는 엑셀 2003 이전 버전은 *.xls, 엑셀 2007 이후 버전은 *.xlsx입니다.
- 종류 : 엑셀(Excel), 혼민시트, 로터스(Lotus) 1-2-3, 퀵트로프로(Quattro pro) 등
- 기능

| 기능 | 설명 |
|--------|---|
| 문서 작성 | 입력된 데이터에 다양한 서식을 적용하여 문서 편집기를 사용한 것과 같이 문서를 작성함 |
| 수치 계산 | 수식과 함수를 이용하여 데이터 계산을 쉽게 처리함 |
| 차트 작성 | 입력한 데이터를 토대로 다양한 모양의 차트를 작성함 |
| 데이터 관리 | 다양한 데이터를 검색, 정렬, 추출하는 데이터 관리 기능을 제공함 |
| 매크로 | 반복적이거나 복잡한 단계를 수행하는 대량의 작업을 일괄적으로 자동 처리함 |

핵심 045 워크시트의 구성

- 데이터 작업이 이루어지는 기본 문서로, 행과 열이 교차되면서 만들어지는 셀로 구성되어 있다.
- 셀(Cell) : 행과 열이 교차되면서 만들어지는 사각형으로, 데이터가 입력되는 기본 단위
- 셀 포인터(Cell Pointer) : 작업이 이루어지는 셀을 나타내며, 현재 셀 포인터가 위치한 셀을 활성 셀(Active Cell)이라고 함
- 행 머리글 : 행의 맨 왼쪽에 숫자로 표시되어 있는 부분
- 열 머리글 : 열의 맨 위쪽에 알파벳으로 표시되어 있는 부분



- 시트 탭 : 통합 문서에 포함되어 있는 시트의 이름을 표시하는 부분으로, 시트 탭을 클릭하여 작업할 시트를 선택함

잠깐만요! 셀의 범위 지정

| 연속된 셀의 범위 지정 | 서로 떨어진 셀의 범위 지정 |
|---|---|
| 범위로 지정할 첫 번째 셀을 클릭한 후, Shift를 누른 상태에서 범위로 지정할 마지막 셀을 클릭 | 첫 번째 셀 범위를 지정한 후, 두 번째 셀 범위부터는 Ctrl을 누른 상태에서 원하는 셀을 클릭하거나 드래그 |

핵심 18.상시, 13.상시, 12.상시, 01.4, 01.1, 00.10, 00.8, 99.7

046 함수

- 함수는 약속된 값으로 정의된 인수를 사용하여 계산하는, 프로그램에 이미 정의된 수식을 말한다.
- 함수는 수식과 같이 등호(=), +, -로 시작해야 한다.

| 함수 | 설명 |
|------------------------|--------------------------------|
| SUM(인수1, 인수2, ...) | 인수의 합계 |
| AVERAGE(인수1, 인수2, ...) | 인수의 평균 |
| MAX(인수1, 인수2, ...) | 인수 중 가장 큰 값 |
| MIN(인수1, 인수2, ...) | 인수 중 가장 작은 값 |
| COUNT(인수1, 인수2, ...) | 인수 중 숫자가 들어 있는 셀의 개수 |
| MOD(인수1, 인수2) | 인수1을 인수2로 나눈 나머지 |
| LEFT(텍스트, 개수) | 텍스트의 왼쪽부터 지정한 개수만큼 표시 |
| MID(텍스트, 시작 위치, 개수) | 텍스트의 시작 위치부터 지정한 개수만큼 표시 |
| RIGHT(텍스트, 개수) | 텍스트의 오른쪽부터 지정한 개수만큼 표시 |
| NOW() | 현재 날짜와 시간을 표시 |
| TODAY() | 현재 날짜를 표시 |
| IF(조건, 인수1, 인수2) | 조건을 비교하여 참이면 인수1, 거짓이면 인수2를 실행 |
| AND(인수1, 인수2, ...) | 인수가 모두 참이면 참 |
| OR(인수1, 인수2, ...) | 인수 중 하나라도 참이면 참 |

잠깐만요! 참조

- 참조는 수식에서 워크시트의 특정 셀이나 셀 범위의 데이터 또는 결과값을 사용하기 위해 주소를 지정하는 것을 말합니다.
- 수식에 사용된 셀의 값이 변경되면 변경된 셀을 참조하는 수식의 값도 자동으로 재계산됩니다.

| 참조 대상 | 참조 방법 |
|------------------------------------|--------------|
| [A1]부터 [A5]까지의 셀 | A1:A5 |
| [A1] 셀 [B1] 셀 [C1] 셀 | A1, B1, C1 |
| 4행에 있는 모든 셀 | 4:4 |
| C열에 있는 모든 셀 | C:C |
| [A1]부터 [A5]까지의 셀과 [C1]부터 [C5]까지의 셀 | A1:A5, C1:C5 |

- 상대 참조 : 셀의 위치에 따라 상대적으로 셀 주소가 변하는 형태로, 수식을 복사하면 셀 주소가 수식이 복사된 셀의 상대적 위치로 변경됨 예 A1, B1
- 절대 참조 : 특정 셀의 주소를 고정시킬 때 사용하는 형태로, 수식을 다른 셀로 복사해도 셀 주소는 변경되지 않음 예 \$A\$1, \$B\$1
- 혼합 참조 : 상대 참조와 절대 참조를 혼합하여 사용함 예 \$A1, A\$1, \$B1, B\$1

핵심 18.상시, 17.상시, 16.상시, 15.상시, 14.상시, 13.상시, 11.7, 11.2, 10.7, 10.3, 10.1, 09.9, 09.7, ...

047 엑셀의 주요 기능

| | |
|-----|--|
| 매크로 | 엑셀에서 사용되는 다양한 명령들을 일련의 순서대로 기록해 두었다가 필요할 때마다 해당 키나 도구를 이용하여 호출하면 기록해 둔 처리 과정이 수행되도록 하는 기능 |
| 차트 | <ul style="list-style-type: none"> • 워크시트의 데이터를 막대나 선, 도형, 그림 등을 사용하여 시각적으로 표현한 것 • 차트를 이용하면 데이터의 추세나 유형 등을 쉽고 직관적으로 이해할 수 있으며, 많은 양의 데이터를 간결하게 요약할 수 있음 • 원본 데이터가 바뀌면 차트의 모양도 바뀜 • 차트 종류 <ul style="list-style-type: none"> - 꺾은선형 차트 : 일정 기간 동안의 데이터 변화 추세를 확인하는 데 적합함 - 막대형 차트 : 각 항목 간의 값을 막대의 길이로 비교·분석하는 데 적합함 - 원형 차트 : 전체 항목의 합에 대한 각 항목의 비율을 나타내고, 항상 한 개의 데이터 계열만 가지고 있으므로 축이 없음 |
| 정렬 | <ul style="list-style-type: none"> • 불규칙하게 입력된 데이터 목록을 특정 기준에 따라 재배열하는 기능 • 정렬 방식에는 오름차순과 내림차순이 있으며, 셀 값에 따라 정렬이 수행됨 • 영문자 대/소문자를 구분하여 정렬할 수 있는 기능을 제공하며, 오름차순으로 정렬하면 소문자가 우선순위를 갖음 |



| | |
|--------|---|
| 부분합 | <ul style="list-style-type: none"> • 많은 양의 데이터 목록을 그룹(필드)별로 분류하고, 각 그룹별로 계산을 수행하는 데이터 분석 도구임 • 부분합을 작성하려면 기준이 되는 필드를 반드시 오름차순이나 내림차순으로 정렬해야 함 |
| 피벗 테이블 | <ul style="list-style-type: none"> • 많은 양의 데이터를 한눈에 쉽게 파악할 수 있도록 요약·분석하여 보여주는 도구임 • 원본 데이터 목록의 행이나 열의 위치를 변경하여 다양한 형태로 표시할 수 있음 |
| 필터 | <ul style="list-style-type: none"> • 데이터 목록에서 설정된 조건에 맞는 데이터만을 추출하여 화면에 나타내는 기능 • 조건을 기술하는 방법에 따라 자동 필터와 고급 필터로 구분할 수 있음 |

핵심 048 프레젠테이션의 기초

- 기업의 제품 소개나 연구 발표, 회의 내용 요약 등 각종 그림이나 도표, 그래프 등을 이용하여 많은 사람에게 효과적으로 의미를 전달할 때 사용되는 응용 프로그램이다.
- 강연회나 세미나, 연구 발표, 교육안 등을 상대방에게 보다 효과적으로 전달하고자 할 때 사용되는 프로그램이다.
- 파워포인트의 확장자는 파워포인트 2003 이전 버전은 *.ppt이고, 파워포인트 2007 이후 버전은 *.pptx이다.
- 종류 : 파워포인트, 프리랜스, 훈민 프레젠테이션 등
- 프레젠테이션의 구성 요소

| | |
|------------|---|
| 슬라이드 | <ul style="list-style-type: none"> • 프레젠테이션을 구성하는 내용을 하나의 화면 단위로 나타낸 것 • 각 페이지의 기본 단위 |
| 개체(Object) | 한 화면을 구성하는 개개의 요소 |
| 시나리오 | 프레젠테이션의 흐름을 기획한 것 |
| 개요 | 시나리오에 의한 프레젠테이션의 실제 내용 |

잠깐만요 !

여러 슬라이드 보기 상태에서 슬라이드 선택하기

- 여러 개의 슬라이드를 연속적으로 선택하려면 첫 번째 슬라이드를 선택한 후 **Shift**를 누른 상태에서 마지막 슬라이드를 선택합니다.
- 서로 떨어져 있는 여러 슬라이드를 선택하려면 **Ctrl**을 누른 상태에서 슬라이드를 하나씩 선택합니다.

여러 도형 선택하기

- **Shift**를 누른 상태에서 도형을 하나씩 선택합니다.
- 선택할 도형이 포함되도록 마우스로 드래그하면 드래그한 범위 안의 도형들이 모두 선택됩니다.

3과목 · PC 운영체제

핵심 049 운영체제의 개념

운영체제의 정의

- 운영체제(OS; Operating System)는 컴퓨터 하드웨어와 일반 컴퓨터 사용자 또는 컴퓨터에서 실행되는 응용 프로그램의 중간에 위치하여 사용자들이 보다 쉽고 간편하게 컴퓨터 시스템을 이용할 수 있도록 제어하는 시스템 소프트웨어의 일종이다.
- 컴퓨터에게는 효율적인 자원 관리를, 사용자에게는 편리한 사용을 제공한다.
- 운영체제의 특성에는 신뢰성, 편리성, 효율성, 용이성 등이 있다.

운영체제의 목적

- 처리 능력 향상
- 반환 시간 단축
- 사용 가능도 향상
- 신뢰도 향상

운영체제의 평가 기준

| | |
|-----------------------------|-------------------------------------|
| 처리 능력 (Throughput) | 일정 시간 내에 시스템이 처리하는 일의 양 |
| 반환 시간 (Turn Around Time) | 시스템에 작업을 의뢰한 시간부터 처리가 완료될 때까지 걸린 시간 |
| 사용 가능도 (Availability) | 시스템을 사용할 필요가 있을 때 즉시 사용 가능한 정도 |
| 신뢰도(Reliability) | 시스템이 주어진 문제를 정확하게 해결하는 정도 |

운영체제의 기능

- 사용자와 컴퓨터 간의 인터페이스 제공
- 시스템의 효율적인 운영 및 관리
- 자원 스케줄링 및 주변장치 관리
- 사용자 간의 데이터 호환
- 각종 하드웨어와 네트워크 관리
- 데이터 공유 및 관리
- 초기 설정 기능 및 이식성 기능
- 시스템의 오류 검사 및 복구



• 운영체제의 종류

| | |
|--------------|----------------------|
| 단일 작업 처리 시스템 | DOS |
| 다중 작업 처리 시스템 | Windows, UNIX, LINUX |

핵심 17.상시, 15.상시, 14.상시, 13.상시, 12.상시, 11.9, 10.10, 10.7, 10.1, 09.3, 09.1, 08.10, 08.2, 07.9, 07.7, ...

050 운영체제의 구성

제어 프로그램(Control Program)

- 감시 프로그램(Supervisor Program) : 각종 프로그램의 실행과 시스템 전체의 작동 상태를 감시·감독하는 프로그램
- 작업 제어 프로그램(Job Control Program) : 어떤 업무를 처리하고 다른 업무로의 이행을 자동으로 수행하기 위한 준비 및 그 처리에 대한 완료를 담당하는 프로그램
- 자료 관리 프로그램(Data Management Program) : 주기억 장치와 보조기억장치 사이의 자료 전송, 파일의 조작 및 처리, 입·출력 자료와 프로그램 간의 논리적 연결 등 시스템에서 취급하는 파일과 데이터를 표준적인 방법으로 처리할 수 있도록 관리하는 프로그램

처리 프로그램(Processing Program)

- 언어 번역 프로그램(Language Translate Program) : 원시 프로그램(Source Program)을 기계어 형태의 목적 프로그램(Object Program)으로 번역하는 프로그램
- 서비스 프로그램(Service Program) : 사용자의 편리를 위해 시스템 제공자가 미리 작성하여 사용자에게 제공하는 것으로, 사용 빈도가 높은 프로그램
- 문제 프로그램(Problem Program) : 특정 업무 및 문제 해결을 위해 사용자가 작성한 프로그램

핵심 16.상시, 15.상시, 13.상시, 12.상시, 10.1, 08.3, 06.10, 04.7, 03.1, 02.10, 02.7, 02.4, 01.10, 01.7

051 언어 번역 프로그램

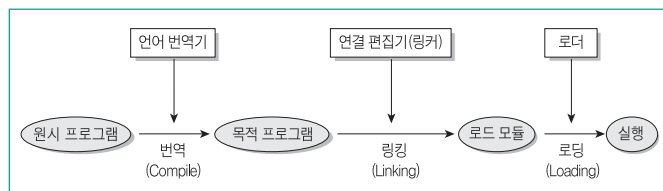
• 언어 번역 프로그램의 종류

| | |
|---------------------|---|
| 어셈블러 (Assembler) | 어셈블리어로 작성된 원시 프로그램을 기계어로 된 목적 프로그램으로 어셈블하는 언어 번역 프로그램 |
| 컴파일러 (Compiler) | <ul style="list-style-type: none"> • 고급 언어로 작성된 프로그램 전체를 목적 프로그램으로 번역한 후 링킹 작업을 통해 컴퓨터에서 실행 가능한 실행 프로그램을 생성함 • 컴파일러 언어 : FORTRAN, COBOL, PASCAL, C, C++, PL/1 등 |

인터프리터 (Interpreter)

- 고급 언어나 코드화된 중간 언어를 입력받아 목적 프로그램 생성 없이 직접 기계어를 생성, 실행해주는 프로그램
- 인터프리터 언어 : BASIC, SNOBOL, LISP, APL 등

• 언어 번역 과정



핵심 18.상시, 17.상시, 16.상시, 15.상시, 14.상시, 13.상시, 12.상시, 11.4, 10.10, 10.3, 09.3, 08.3, 07.4, 06.1, ...

052 서비스 프로그램

링커(Linker), 연결 편집기(Linkage Editor)

언어 번역 프로그램이 생성한 목적 프로그램들과 라이브러리, 또 다른 실행 프로그램(로드 모듈) 등을 연결하여 실행 가능한 로드 모듈을 만드는 프로그램이다.

정렬/합병 프로그램(Sort/Merge Program)

데이터를 일정한 기준으로 정렬하거나 정렬된 두개 이상의 파일을 하나로 합치는 프로그램이다.

라이브러리안(Librarian)

프로그램의 라이브러리를 유지·관리하는 프로그램이다.

유틸리티 프로그램(Utility Program)

사용자의 편의를 도모하기 위한 프로그램으로 텍스트 에디터, 디버거 등이 있다.

로더(Loader, Module Loader)

- 컴퓨터 내부로 정보를 들여오거나 로드 모듈을 디스크 등의 보조기억장치로부터 주기억장치에 적재하는 프로그램이다.
- 로더의 기능
 - 할당(Allocation) : 실행 프로그램을 실행시키기 위해 기억장치 내에 옮겨놓을 공간을 확보하는 기능
 - 연결(Linking) : 부 프로그램 호출 시 그 부 프로그램이 할당된 기억 장소의 시작 주소를 호출한 부분에 등록하여 연결하는 기능
 - 재배치(Relocation) : 디스크 등의 보조기억장치에 저장된 프로그램이 사용하는 각 주소들을 할당된 기억 장소의 실제 주소로 배치(변환)시키는 기능
 - 적재>Loading) : 실행 프로그램을 할당된 기억 공간에 실제로 옮기는 기능



잠깐만요! 재배치 로더(Loader)

로더의 기본 기능 4가지를 모두 수행하는 일반적인 기능의 로더를 의미합니다.

핵심 053 운영체제의 운용 방식

일괄 처리 시스템(Batch Processing System)

초기의 컴퓨터 시스템에서 사용된 형태로, 일정량 또는 일정 기간 동안 데이터를 모아서 한꺼번에 처리하는 방식이다.

실시간 처리 시스템(Real Time Processing System)

- 데이터 발생 즉시, 또는 데이터 처리 요구가 있는 즉시 처리하여 결과를 산출하는 방식이다.
- 우주선 운행이나 레이더 추적기, 핵물리학 실험 및 데이터 수집, 전화 교환장치의 제어, 은행의 온라인 업무 등 시간에 제한을 두고 수행되어야 하는 작업에 사용된다.

시분할 시스템(Time Sharing System)

- 하나의 프로세서가 CPU를 독점하는 것을 방지하기 위해 각각 하나의 시간 슬롯을 할당하여 동작하도록 하는 시스템으로, 라운드 로빈(Round Robin) 방식이라고도 한다.
- 한 시스템을 여러 명의 사용자가 공유하여 동시에 작업을 수행한다.

다중 프로그래밍 시스템(Multi-Programming System)

하나의 CPU와 주기억장치를 이용해 여러 개의 프로그램을 동시에 처리하는 방식이다.

다중 처리 시스템(Multi-Processing System)

컴퓨터 한 대에 두 개 이상의 CPU를 설치하여 병렬 처리하는 것으로, 한 시스템에서 여러 개의 처리 과정을 동시에 수행하는 것이다.

운영체제 운용 기법의 발달 과정

일괄 처리 시스템 → 실시간 처리 시스템 → 다중 프로그래밍 → 다중 처리 시스템 → 시분할 시스템 → 분산 운영체제

핵심 054 프로세스 관리

프로세스(Process)의 정의

- 프로세서(처리기, CPU)에 의해 처리되는 사용자 프로그램, 즉 실행중인 프로그램을 의미하며, 작업(Job) 또는 태스크(Task)라고도 한다.
- 실기억장치에 저장된 프로그램
- 프로세서가 할당되는 실체
- 운영체제가 관리하는 실행 단위

프로세스 주요 상태 및 관련 용어

- 준비(Ready) : 프로세스가 CPU를 할당받기 위해 기다리고 있는 상태로, 준비상태 큐에서 실행을 준비함
- 실행(Run) : 준비상태 큐에 있는 프로세스가 CPU를 할당받아 실행되는 상태
- 대기(Wait), 보류, 블록(Block) : 프로세스에 입·출력 처리가 필요하면 현재 실행중인 프로세스가 중단되고, 입·출력 처리가 완료될 때까지 대기하고 있는 상태
- 종료(Terminated, Exit) : 프로세서의 실행이 끝나고 프로세스 할당이 해제된 상태
- 디스패치(Dispatch) : 준비 상태에서 대기하고 있는 프로세스 중 우선순위가 가장 높은 프로세스가 CPU를 할당받아 실행 상태로 전이되는 과정
- Wake Up : 입·출력 작업이 완료되어 프로세스가 대기 상태에서 준비 상태로 전이되는 과정

핵심 055 프로세스 스케줄링

- 스케줄링(Scheduling)은 프로세스가 생성되어 실행될 때 필요한 시스템의 여러 자원을 해당 프로세스에게 할당하는 작업을 의미하며, 이를 수행하는 것을 스케줄러(Scheduler)라고 한다.
- 비선점(Non-preemptive) 스케줄링 : 이미 할당된 CPU를 다른 프로세스가 강제로 빼앗아 사용할 수 없는 스케줄링 기법

FCFS
= FIFO

- 준비상태 큐에 도착한 순서에 따라 차례로 CPU를 할당하는 기법
- 가장 먼저 CPU를 요청한 프로세스에게 가장 먼저 CPU를 할당하여 실행할 수 있음



| | |
|------|--|
| SJF | 준비상태 큐에서 기다리고 있는 프로세스들 중에서 실행 시간이 가장 짧은 프로세스에게 먼저 CPU를 할당하는 기법 |
| HRN | <ul style="list-style-type: none"> 실행 시간이 긴 프로세스에 불리한 SJF 기법을 보완하기 위한 것 대기 시간과 서비스(실행) 시간을 이용하는 기법 |
| 우선순위 | 준비상태 큐에서 기다리는 각 프로세스마다 우선순위를 부여하여 그 중 가장 높은 프로세스에게 먼저 CPU를 할당하는 기법 |

- 선점(Preemptive) 스케줄링 : 하나의 프로세스가 CPU를 할당받아 실행하고 있을 때 우선순위가 높은 다른 프로세스가 CPU를 강제로 빼앗아 사용할 수 있는 스케줄링 기법

| | |
|-------------------------|---|
| SRT | 현재 실행중인 프로세스의 남은 시간과 준비 상태 큐에 새로 도착한 프로세스의 실행 시간을 비교하여 가장 짧은 실행 시간을 요구하는 프로세스에게 CPU를 할당하는 기법 |
| 라운드 로빈 (Round Robin) | 규정 시간 또는 시간 조각(Slice)을 미리 정의하여 CPU 스케줄러가 준비상태 큐에서 정의된 시간만큼 각 프로세스에 CPU를 제공하는 시분할 시스템에 적절한 스케줄링 기법 |
| 다단계 큐 | 프로세스를 특정 그룹으로 분류할 수 있을 경우 그룹에 따라 각기 다른 준비상태 큐를 사용하는 기법 |
| 다단계 피드백 큐 | 특정 그룹의 준비상태 큐에 들어간 프로세스가 다른 준비상태 큐로 이동할 수 없는 다단계 큐 기법을 준비상태 큐 사이를 이동할 수 있도록 개선한 기법 |

핵심 056 교착상태(DeadLock)

- 다중 프로그래밍 상에서 2개의 프로세스가 실행중일 때 각 프로세스는 자신이 필요한 자원을 가지고 실행하다가 서로 자신이 점유하고 있는 자원을 포기하지 않은 상태에서 다른 프로세스가 자원을 요구하여 두 프로세스 모두 실행을 할 수 없게 되는 현상을 의미한다.
- 교착상태 발생의 필요 충분 조건

| | |
|-----------------------------|--|
| 상호 배제 (Mutual Exclusion) | 한 번에 한개의 프로세스만이 공유 자원을 사용할 수 있어야 함 |
| 점유와 대기 (Hold and Wait) | 최소한 하나의 자원을 점유하고 있으면서 다른 프로세스에 할당되어 사용되고 있는 자원을 추가로 점유하기 위해 대기하는 프로세스가 있어야 함 |

| | |
|---------------------------------|--|
| 비선점 (Non-preemption) | 다른 프로세스에 할당된 자원은 사용이 끝날 때까지 강제로 빼앗을 수 없어야 함 |
| 환형 대기, 순환 대기 (Circular Wait) | 공유 자원과 공유 자원을 사용하기 위해 대기하는 프로세스들이 원형으로 구성되어 있어 자신에게 할당된 자원을 점유하면서 앞이나 뒤에 있는 프로세스의 자원을 요구해야 함 |

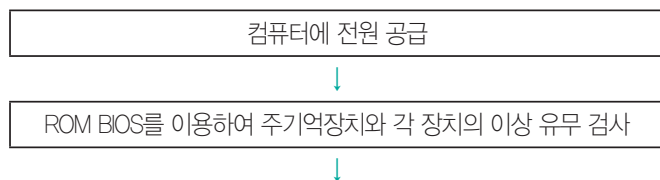
핵심 057 기억장치 - 교체 전략

- OPT(OPTimal replacement, 최적 교체) : 앞으로 가장 오랫동안 사용하지 않을 페이지를 교체하는 기법
- FIFO(First In First Out) : 각 페이지가 주기억장치에 적재될 때마다 그때의 시간을 기억시켜 가장 먼저 들어와서 가장 오래 있었던 페이지를 교체하는 기법
- LRU(Least Recently Used) : 계수기를 두어 가장 오랫동안 참조되지 않은 페이지를 교체하는 기법
- LFU(Least Frequently Used) : 사용 빈도가 가장 적은 페이지를 교체하는 기법
- MRU(Most Recently Used) : 사용 빈도가 가장 많은 페이지를 교체하는 기법
- NUR(Not Used Recently) : 최근에 사용하지 않은 페이지를 교체하는 기법

핵심 058 DOS의 부팅 과정

- 부팅(Booting, Bootstep)은 컴퓨터에 전원을 넣은 순간부터 DOS(시스템 프로그램)를 디스크로부터 주기억장치로 읽어 내어 컴퓨터를 이용할 수 있는 상태로 만들어주는 과정을 의미한다.
- 부팅에 반드시 필요한 시스템 파일에는 MS-DOS, SYS, IO.SYS, COMMAND.COM이 있다.
- 부팅 중 [F8]을 눌러 'CONFIG.SYS' 파일과 'AUTOEXEC.BAT' 파일의 실행 여부를 선택할 수 있다.

부팅(Booting) 순서





부팅의 종류

| | |
|-------------------------|---|
| 웜 부팅 (Warm Booting) | Ctrl+Alt+Delete를 눌러 컴퓨터를 재부팅하는 소프트웨어적 부팅 |
| 콜드 부팅 (Cold Booting) | 컴퓨터 본체의 전원 스위치를 눌러 켜거나 본체의 리셋(Reset) 단추를 눌러 재부팅하는 것 |

059 DOS의 환경 설정 파일

- 도스(DOS)로 부팅될 때 자신에게 필요한 시스템 환경을 설정해 주는 파일로, 환경 설정 파일에는 CONFIG.SYS가 있다.
- 디스크의 동작 속도를 향상시켜 주는 버퍼/캐시를 설정할 수 있다.
- 키보드, 마우스, 기타 주변장치 활용 방법을 설정할 수 있다.
- COPY CON 명령으로 파일을 작성 및 수정할 수 있다.
- CONFIG.SYS 파일은 반드시 루트 디렉터리에 존재해야 실행된다.
- TYPE 명령으로 내용을 확인할 수 있다.
- 주요 환경 설정 명령어

| 명령어 | 기능 |
|-----------|--|
| LASTDRIVE | 드라이브의 최대 개수 지정 |
| FILES | 동시에 열 수 있는 파일의 수 지정 |
| BREAK | 프로그램 실행을 중지하는 Ctrl+C(Ctrl+Break)의 작동 여부 설정 |
| PAUSE | CONFIG.SYS 파일의 실행을 일시 중지시킴 |

| | |
|----------|------------------------------------|
| DEVICE | 마우스, 스캐너와 같은 장치 사용 시 장치 드라이버 연결 |
| BUFFERS | 시스템이 사용 가능한 버퍼 수 지정 |
| RAMDRIVE | 램의 일부를 드라이브처럼 할당하여 고속 처리가 가능하도록 해줌 |
| HIMEM | DOS가 연속 확장 메모리를 사용할 수 있도록 해줌 |
| EMM386 | 연장 메모리(EMS)를 사용할 수 있도록 해줌 |

060 DOS - 내부, 외부, 기본, 필터 명령어

내부 명령어

- 도스가 부팅될 때 COMMAND.COM이 실행됨과 동시에 주기억장치에 상주하는 명령어이다.
- 주기억장치에 상주하므로 언제든지 실행이 가능하다.
- 비교적 자주 사용되며 실행 과정이 간단하다.
- 경로(Path)와 관계없이 어떤 디렉터리에서도 실행이 가능하다.
- COMMAND.COM이 관리하는 명령어로 COMMAND.COM 파일이 주기억장치에 올려짐으로써 사용할 수 있다.
- 종류 : CLS, DIR, VER, COPY, DATE, TIME, MD, CD, RD, PROMPT, VOL, TYPE 등

외부 명령어

- 보조기억장치에 저장되어 있다가 사용자가 명령을 입력하면 주기억장치에 적재시킨 후 실행하는 명령어이다.
- 독립된 파일의 형태이며, DIR 명령으로 확인이 가능하다.
- 종류 : ATTRIB, FORMAT, CHKDSK, FDISK, LABEL, SCANDISK, DISKCOPY, XCOPY, DELTREE, SORT, SYS 등

기본 명령어

| | |
|--------|-------------------------|
| CLS | 현재 화면에 표시된 내용을 지움 |
| DATE | 현재 시스템의 날짜를 확인하거나 변경함 |
| TIME | 현재 시스템의 시간을 확인하거나 변경함 |
| VER | 현재 사용중인 DOS의 버전을 표시함 |
| PROMPT | DOS의 프롬프트를 여러 가지 형태로 변경 |



| | |
|-----|----------------------|
| VOL | 드라이브의 볼륨명과 일련번호를 표시함 |
|-----|----------------------|

필터 명령어

| | |
|------|-------------------------------|
| MORE | 내용을 한 화면씩 출력함 |
| FIND | 하나 또는 여러 개의 파일에서 특정한 문자열을 검색함 |
| SORT | 내용을 정렬하여 화면이나 파일로 출력함 |

핵심 061 DOS 명령어 - 디스크 관련 명령어

| | |
|----------|---|
| FORMAT | <ul style="list-style-type: none"> • 디스크에 데이터가 저장될 수 있도록 트랙과 섹터를 만드는 초기화 작업을 수행함 • FORMAT 명령의 옵션 <ul style="list-style-type: none"> - /S : 포맷한 후 시스템 파일을 복사하여 부팅 가능한 디스크로 만들 - /Q : 이미 사용하던 디스크의 빠른 포맷 - /V[:이름] : 포맷한 후 디스크에 볼륨명 부여 - /F : 용량을 지정하여 포맷 - /4 : 고밀도 드라이브(1.2M)에서 360KB 형식으로 포맷 |
| DISKCOPY | 원본 디스크에 있는 내용을 모두 다른 디스크로 복사함 |
| CHKDSK | 디스크의 상태를 점검하고 결과를 표시함 |
| SCANDISK | 하드디스크의 상태를 검사하며, 하드디스크에 존재하는 미미한 오류를 수정할 수 있음 |
| FDISK | 하드디스크를 논리적으로 여러 개의 디스크로 나누어 (파티션 설정), 각 볼륨이 서로 다른 드라이브 문자를 가진 별개의 드라이브로 동작하도록 설정함 |
| DEFRAG | 단편화되어 있는 파일의 저장 상태를 최적화하여 디스크의 작동 효율을 높임 |
| SYS | 도스의 시스템 파일을 디스크에 복사하는 것으로, 해당 디스크로 부팅이 가능함 |
| DISKCOMP | 동일한 디스크인지 서로 비교하며, DISKCOPY 명령으로 복사된 디스크가 정상적으로 복사되었는지 판별할 때 유용하게 사용함 |

핵심 062 DOS 명령어 - 디렉터리 관련 명령어

| | |
|----------------------|--|
| MD(Make Directory) | 새로운 디렉터리를 만드는 명령어로, 같은 디렉터리에 동일한 이름의 디렉터리를 생성할 수 없음 |
| RD(Remove Directory) | <ul style="list-style-type: none"> • 디렉터리를 삭제함 • 삭제하려는 디렉터리 안에 다른 디렉터리나 파일이 없어야 삭제할 수 있음 |

| | |
|----------------------|---|
| CD(Change Directory) | 다른 디렉터리로 이동함 |
| DELTREE | 디렉터리 안에 있는 파일과 디렉터리까지 모두 삭제함 |
| XCOPY | <ul style="list-style-type: none"> • 특정한 디렉터리 내의 모든 파일 및 하위 디렉터리까지 복사가 가능함 • 숨김 파일과 시스템 파일은 복사되지 않음 • XCOPY 명령의 옵션 <ul style="list-style-type: none"> - /P : 각 파일을 복사할 때마다 취소할 수 있도록 확인 메시지를 표시함 - /A : 파일에 저장 속성이 설정된 파일만 복사함 - /S : 비어 있지 않은 디렉터리와 하위 디렉터리를 모두 복사함 - /D : 특정 날짜 이후 변경된 파일만 복사 |
| PATH | <ul style="list-style-type: none"> • 실행 파일을 찾는 경로를 설정하거나 보여주는 데 사용됨 • 서로 다른 드라이브에 있는 파일도 PATH에 지정되면 검색이 가능함 • 찾고자 하는 파일이 현재의 디렉터리에 없을 때만 PATH에서 지정한 경로를 검색함 |

핵심 063 DOS 명령어 - 파일 관련 명령어

- DIR(DIRectory) : 디스크 내에 수록된 파일 및 디렉터리에 대한 정보를 표시함

| 옵션 | 기능 |
|----|--|
| /P | 목록을 한 화면 단위로 표시함 |
| /W | 한 줄에 5개씩 목록을 출력해주는 것으로, 화면에 가장 많은 파일을 표현할 수 있는 방식임 |
| /O | 지정한 정렬 방식으로 파일 목록을 표시하며 'O'로 지정하면 역순으로 표시함 <ul style="list-style-type: none"> • D(Date) : 날짜/시간순으로 정렬(빠른 순) • E(Extension) : 확장자순으로 정렬 • N(Name) : 파일의 이름순으로 정렬 • S(Size) : 파일의 크기(Byte)순으로 정렬(작은 것부터) |
| /S | 하위 디렉터리의 정보까지 표시함 |
| /A | 기록 속성이 설정된 목록을 표시함 |
| /H | 숨겨진 파일 목록을 표시함 |

- COPY : 파일을 지정한 곳에 복사하거나 여러 개의 파일을 결합함
- DEL = ERASE : 파일을 삭제함
- UNDELETE : DEL이나 ERASE를 사용하여 삭제한 파일을 복원함



- TYPE : 아스키 코드로 작성된 파일, CONFIG.SYS 파일, 배치 파일, COPY CON 명령으로 작성한 파일 등의 내용을 화면에 출력함
- REN : 파일의 이름을 변경함
- ATTRIB : 파일의 속성을 표시, 해제, 지정함

| 옵션 | 기능 |
|----|---------------|
| + | 속성을 지정할 때 사용함 |
| - | 속성을 해제할 때 사용함 |
| R | 읽기 전용 속성 |
| A | 저장/백업 속성 |
| S | 시스템 파일 속성 |
| H | 숨김 파일 속성 |

- FC : 2개의 파일을 비교하여 그 차이를 나타내며, 파일 복사 후 정확히 복사되었는지 확인할 때 사용함

핵심 064 WINDOWS의 특징

| | |
|--------------------------------------|---|
| 그래픽 사용자 인터페이스(GUI) 사용 | 키보드로 명령어를 직접 입력하지 않고, 아이콘이나 메뉴를 마우스로 선택하여 모든 작업을 수행하는 사용자 작업 환경(GUI)을 사용함 |
| 선점형 멀티태스킹 (Preemptive Multi-Tasking) | 운영체제가 각 작업의 CPU 이용 시간을 제어 하여 응용 프로그램 실행중 문제가 발생하면 해당 프로그램을 강제로 종료시키고, 모든 시스템 자원을 반환하는 멀티태스킹 운영 방식 |
| 32비트 또는 64비트 데이터 처리 | 이전 버전과의 호환을 위해 부분적으로 16비트 데이터 처리를 하나 대부분 32비트나 64비트 데이터 처리를 하므로 더 많은 양의 데이터를 빠르게 처리할 수 있음 |
| 플러그 앤 플레이 (PnP; Plug & Play) | <ul style="list-style-type: none"> • 컴퓨터 시스템에 새로운 하드웨어를 장착 하고 시스템을 가동시키면 자동으로 하드웨어를 인식하고 실행하는 기능 • 운영체제가 주변기기를 자동으로 인식하므로 시스템 환경을 사용자가 직접 설정 할 필요가 없음 |
| OLE(Object Linking and Embedding) | 다른 여러 응용 프로그램에서 작성된 문자나 그림 등의 개체(Object)를 현재 작성중인 문서에 자유롭게 연결(Linking)하거나 삽입(Embedding)하여 편집할 수 있게 하는 기능 |

| | |
|---------------|--|
| 255자의 긴 파일 이름 | <ul style="list-style-type: none"> • 파일 이름을 지정할 때 VFAT(Virtual File Allocation Table)를 이용하여 최대 255자 까지 지정할 수 있음 • 파일 이름에 공백을 포함할 수 있으며, 한 글은 127자까지 지정할 수 있음 |
| 향상된 네트워크 기능 | Windows는 운영체제 자체에서 여러 가지 프로토콜을 지원하므로 네트워크 구축 및 통신에 관련된 여러 가지 작업을 쉽게 할 수 있음 |
| DOS와 호환 | Windows는 DOS와 호환이 가능하여 기존에 사용하던 대부분의 DOS 응용 프로그램을 Windows에서도 그대로 사용할 수 있음 |

잠깐만요! 파일 시스템

- 보조기억장치에 저장되는 파일에 대해 수정, 삭제, 추가, 검색 등의 작업을 체계적으로 할 수 있도록 지원하는 관리 시스템을 말합니다.
- 파일 시스템에는 FAT(16), FAT32, NTFS가 있습니다.
- FAT32는 FAT에서 파생된 것으로 FAT에 비해 클러스터 크기가 작 으므로 하드디스크의 공간 낭비를 줄일 수 있습니다.
- NTFS의 장점
 - 성능, 보안, 안정성 면에서 뛰어난 고급 기능을 제공하며, 시스템 리소스를 최소화 할 수 있습니다.
 - 파일 및 폴더에 대한 액세스 제어를 유지하고 '표준 사용자' 계정을 지원합니다.

핵심 065 WINDOWS - 부팅 메뉴

| | |
|-----------------------|---|
| 안전 모드 | <ul style="list-style-type: none"> • 컴퓨터가 비정상적으로 작동될 때 컴퓨터에 발생한 문제를 해결하기 위해 사용하는 방식 • 컴퓨터 작동에 필요한 최소한의 장치만을 설정하여 부팅하므로 네트워크 관련 작업이나 사운드 카드, 모뎀 등은 사용할 수 없음 • 화면 모드 : 800×600 |
| 안전 모드 (네트워킹 사용) | 네트워크가 지원되는 안전 모드로 부팅하는 방식 |
| 안전 모드 (명령 프롬프트 사용) | 안전 모드로 부팅하되, GUI 환경이 아닌 DOS 모드로 부팅함 |
| 부팅 로깅 사용 | <ul style="list-style-type: none"> • 부팅 과정을 Ntbtlog.txt 파일에 기록하며 부팅하는 방식 • 문제가 있을 때 이 방법을 사용하여 부팅한 후 Ntbtlog.txt 파일을 열어 문제가 발생한 부분을 확인할 수 있음 |
| 저해상도 비디오 사용 (640×480) | <ul style="list-style-type: none"> • 화면 모드를 640×480 해상도로 설정하여 부팅하는 방식 • 그래픽 카드 드라이버를 새로 설치한 후 Windows가 제대로 실행되지 않을 때 유용 |



| | |
|--------------------------|--|
| 마지막으로 성공한 구성(고급) | <ul style="list-style-type: none"> 마지막으로 시스템이 문제없이 실행되고 종료되었을 때의 레지스트리 정보와 드라이버를 사용하여 부팅하는 방식 이 방식을 이용하면 마지막으로 부팅에 성공한 이후의 변경 사항은 모두 손실됨 |
| 디렉터리 서비스 복원 모드 | <ul style="list-style-type: none"> 디렉터리 컨트롤러에서만 사용 가능한 방식 디렉터리 서비스를 복원할 수 있도록 Active Directory를 실행하는 Windows 도메인 컨트롤러를 시작함 |
| 디버깅 모드 | 네트워크로 연결된 경우 컴퓨터 관리자에게 해당 컴퓨터의 디버그 정보를 보내면서 컴퓨터를 시작함 |
| 시스템 오류 시 자동 다시 시작 사용 안 함 | 시스템에 오류가 발생한 경우 시스템이 자동으로 다시 시작되지 않도록 지정함 |
| 표준 모드로 Windows 시작 | 한글 Windows의 기본 부팅 방식 |

066 WINDOWS - 마우스 사용법

| | |
|-----------------------|--|
| 클릭(Click) | <ul style="list-style-type: none"> 마우스 왼쪽 단추를 한 번 누르는 동작 아이콘이나 메뉴, 창을 선택할 때 사용함 |
| 더블클릭(Double Click) | <ul style="list-style-type: none"> 마우스 왼쪽 단추를 빠르게 두 번 누르는 동작 창이나 폴더를 열거나 응용 프로그램을 실행할 때 사용함 |
| 드래그 앤 드롭(Drag & Drop) | <ul style="list-style-type: none"> 마우스 왼쪽 단추를 누른 채 끌다가(드래그) 놓는(드롭) 동작 파일, 폴더 등을 이동하거나 복사할 때 또는 창의 크기를 조절할 때 사용함 바로 가기 아이콘을 만들 때에도 드래그 앤 드롭을 이용함 |
| 오른쪽 단추 클릭 | 마우스 오른쪽 단추를 한 번 누르는 동작으로, 바로 가기 메뉴를 나타낼 때 사용함 |

067 WINDOWS - 바로 가기 키(단축키)

| 바로 가기 키 | 기능 |
|-----------------------|-------------------------------------|
| 기능키 F1, F2, F3, F5 | 도움말 보기, 폴더/파일 이름 바꾸기, 찾기, 최신 정보로 고침 |
| Alt + →, ← | 현재 실행중인 화면의 다음 화면이나 이전 화면으로 이동함 |
| Alt + Esc | 현재 실행중인 프로그램들을 순서대로 전환함 |

| | |
|---------------------|---|
| Alt + Tab | <ul style="list-style-type: none"> 현재 실행중인 프로그램들의 목록을 화면 중앙에 나타냄 Alt를 누른 상태에서 Tab을 이용하여 이동할 작업 창을 선택함 |
| Alt + Enter | <ul style="list-style-type: none"> 선택된 항목의 등록 정보(속성)를 나타냄 도스 전체 화면과 창 형태를 번갈아가며 표시함 |
| Alt + Spacebar | 현재 열려 있는 창의 제어 상자(창 조절 메뉴)를 표시함 |
| Alt + F4 | <ul style="list-style-type: none"> 실행중인 창(Window)이나 응용 프로그램을 종료함 실행중인 프로그램이 없으면 시스템을 종료함 |
| Alt + Print Screen | 현재 작업중인 활성 창을 클립보드로 복사함 |
| Print Screen | 화면 전체를 클립보드로 복사함 |
| Ctrl + A | 폴더 및 파일을 모두 선택함 |
| Ctrl + Esc | [시작] 단추를 클릭한 것처럼 [시작] 메뉴를 표시함 |
| Ctrl + Alt + Delete | <ul style="list-style-type: none"> 한 번 누르기 : '프로그램 종료' 대화상자를 호출하여 문제가 있는 프로그램을 강제로 종료함 두 번 연속해서 누르기 : 시스템을 강제로 재부팅함 |
| Shift + Delete | 폴더나 파일을 휴지통을 거치지 않고 바로 삭제함 |
| Shift + F10 | 바로 가기 메뉴를 표시함 |
| Shift + CD 삽입 | Shift를 누른 상태에서 CD를 삽입하면 CD의 자동 실행 기능이 작동하지 않음 |

068 WINDOWS - 바탕 화면의 바로 가기 메뉴

- 바탕 화면 작업 시 자주 사용하는 명령을 메뉴로 구성한 것으로, 바탕 화면에서 마우스 오른쪽 단추를 클릭하면 표시된다.
- 바로 가기 메뉴에는 보기, 정렬 기준, 새로 고침, 새로 만들기, 화면 해상도, 가젯, 개인 설정 등이 있다.
- 바탕 화면에 있는 아이콘의 표시 유무를 지정할 수 있다.

잠깐만요! 바탕 화면의 정렬 기준

이름, 크기, 종류(항목 유형), 날짜, 자동 정렬 등이 있습니다.



핵심

18.상시, 17.상시, 16.상시, 14.상시, 13.상시, 12.상시, 11.2, 10.10, 10.7, 10.1, 08.10, 07.7, 06.7, 06.1, ...

069 WINDOWS - 바로 가기(단축) 아이콘

- 자주 사용하는 문서나 프로그램을 빠르게 실행시키기 위한 아이콘으로, 실제 실행 파일과 연결해 놓은 것이다.
- ‘단축 아이콘’이라고도 하며, 폴더나 파일, 디스크 드라이브, 다른 컴퓨터, 프린터 등 모든 개체에 대해 바로 가기 아이콘을 작성할 수 있다.
- 왼쪽 아랫부분에 화살표 표시가 있어 일반 아이콘과 구별된다.
- 사용자가 임의로 생성하거나 삭제할 수 있다.
- 확장자는 LNK이며, 컴퓨터에 여러 개 존재할 수 있다.
- 바로 가기 아이콘을 삭제하더라도 원본 파일은 삭제되지 않는다.
- 바로 가기 아이콘 만들기
 - 메뉴 이용 : Windows 탐색기의 [파일] → [바로 가기 만들기], [파일] → [새로 만들기] → [바로 가기] 선택
 - 바로 가기 메뉴 이용 : 개체를 선택한 후 바로 가기 메뉴에서 [바로 가기 만들기] 선택
 - 오른쪽 버튼으로 끌기 : 마우스 오른쪽 버튼으로 개체를 선택한 후 원하는 위치로 끌어다 놓으면 표시되는 바로 가기 메뉴에서 [여기에 바로 가기 만들기] 선택
 - **Ctrl+Shift+드래그** : 개체를 선택한 후 **Ctrl+Shift**를 누른 채 원하는 위치로 끌어다 놓음

핵심

18.상시, 17.상시, 16.상시, 15.상시, 13.상시, 11.7, 11.2, 10.10, 09.7, 07.4, 07.1, 05.7, 04.4, 02.10

070 WINDOWS - 작업 표시줄

- 작업 표시줄은 현재 실행되고 있는 프로그램 단추와 프로그램을 빠르게 실행하기 위해 등록한 고정 프로그램 단추 등이 표시되는 곳으로서, 기본적으로 바탕 화면의 맨 아래쪽에 있다.
- 작업 표시줄은 [시작] 단추, 빠른 실행 도구 모음, 고정된 프로그램 단추/실행중인 프로그램 단추가 표시되는 부분, 입력 도구 모음, 알림 영역(표시기), ‘바탕 화면 보기’ 단추로 구성된다.
- 작업 표시줄은 위치를 변경하거나 크기를 조절할 수 있다. 단, 크기는 화면의 1/2까지만 늘릴 수 있다.

‘작업 표시줄 및 시작 메뉴 속성’ 대화상자의 탭별 기능

| | |
|--------|---|
| 작업 표시줄 | <ul style="list-style-type: none"> • 작업 표시줄 잠금 : 작업 표시줄의 위치나 크기, 작업 표시줄에 표시된 도구 모음의 크기나 위치를 변경하지 못하도록 함 • 작업 표시줄 자동 숨기기 : 작업 표시줄이 있는 위치에 마우스를 대면 작업 표시줄이 나타나고 마우스를 다른 곳으로 이동하면 작업 표시줄이 사라짐 • 작은 아이콘 사용 : 작업 표시줄의 프로그램 단추들이 작은 아이콘으로 표시됨 • 화면에서의 작업 표시줄 위치 : 아래쪽, 왼쪽, 오른쪽, 위쪽 중에서 선택하여 작업 표시줄의 위치를 지정함 |
| 시작 메뉴 | <ul style="list-style-type: none"> • 시작 메뉴 사용자 지정 : 시작 메뉴에 표시할 항목, 표시할 최근 프로그램 수(최대 30개), 점프 목록에 표시할 최근 항목 수(최대 60개) 등을 지정함 • 전원 단추 동작 : 시작 메뉴의 전원 단추를 클릭하면 수행할 작업을 시스템 종료, 사용자 전환, 로그오프, 잠금, 다시 시작, 절전 등에서 선택하여 지정함 |
| 도구 모음 | 작업 표시줄에 표시할 도구 모음을 지정함 |

핵심

16.상시, 15.상시, 14.상시, 12.상시, 10.3, 08.7, 08.3, 06.10, 06.7, 05.7, 04.2, 03.10, 02.4, 02.4, 00.10

071 WINDOWS - 탐색기

- Windows 탐색기는 컴퓨터에 설치된 디스크 드라이브, 응용 프로그램 파일 및 폴더 등을 관리할 수 있는 곳으로, 파일이나 폴더, 디스크 드라이브에 관련된 모든 작업을 수행할 수 있다.
- Windows 탐색기는 컴퓨터의 파일과 폴더를 계층(트리) 구조로 표시한다.
- Windows 탐색기는 크게 탐색 창과 파일 영역(폴더 창), 두 부분으로 나누어져 있다.
- 탐색 창에는 컴퓨터에 존재하는 모든 폴더가 표시되고, 파일 영역(폴더 창)에는 탐색 창에서 선택한 폴더의 내용(하위 폴더, 파일)이 표시된다.
- Windows 탐색기의 [보기] 메뉴에서 큰 아이콘, 작은 아이콘, 간단히, 자세히 등을 선택하여 파일 영역의 아이콘 크기를 변경할 수 있다.
- ▶ 폴더 : 폴더 내에 또 다른 폴더, 즉 하위 폴더가 있음을 의미하며, 부분을 클릭하면 하위 폴더가 표시되고, 로 변경됨
- ▲ 폴더 : 하위 폴더까지 표시된 상태임을 의미하며, 부분을 클릭하면 하위 폴더가 숨겨지고 로 변경됨

※ Windows 98이나 Windows XP 버전에서는 ▶ 가 로 ▲ 가 로 표시됩니다.



핵심 18.상시, 17.상시, 16.상시, 15.상시, 14.상시, 13.상시, 12.상시, 11.9, 11.4, 10.1, 09.9, 09.7, 08.10, 08.3, ...
072 WINDOWS - 파일 / 폴더 선택 · 찾기

파일/폴더 선택

| | |
|-------------|--|
| 하나의 항목 선택 | 항목을 마우스 왼쪽 버튼으로 클릭함 |
| 연속적인 항목 선택 | <ul style="list-style-type: none"> 선택할 항목의 범위를 마우스로 드래그함 첫 항목을 클릭한 후 [Shift]를 누른 상태에서 마지막 항목을 클릭함 |
| 비연속적인 항목 선택 | [Ctrl] 을 누른 상태에서 선택할 항목을 차례로 클릭함 |
| 전체 항목 선택 | [편집] → [전체 선택]을 선택([Ctrl]+[A]) |
| 선택 항목 반전 | [편집] → [선택 항목 반전]을 이용하여 현재 선택된 항목을 해제하고 나머지 항목을 선택함 |

파일/폴더 찾기

- 특정 파일이나 폴더가 있는 위치를 모를 경우 빠르고 쉽게 파일이나 폴더가 있는 위치를 찾는 것이다.
- 파일과 폴더의 이름을 입력하여 찾을 수 있으며, 만능 문자(와일드 카드, *, ?)를 이용할 수 있다.
- 특정 문자열, 파일의 크기 및 유형, 수정한 날짜 등의 조건을 지정하여 파일을 찾을 수 있다.
- 하위 폴더까지 검색할지 여부를 지정할 수 있다.

핵심 18.상시, 16.상시, 15.상시, 14.상시, 13.상시, 12.상시, 10.1, 09.9, 09.7, 09.3, 09.1, 08.7, 08.2, 07.9, ...
073 WINDOWS - 파일 / 폴더 복사 · 이동 · 클립보드

파일/폴더 복사

| | |
|-------------|---|
| [편집] 메뉴 이용 | [편집] → [복사]를 선택한 후 복사할 위치에서 [편집] → [붙여넣기]를 선택함 |
| 바로 가기 키 이용 | [Ctrl]+[C] 를 누른 후 복사할 위치에서 [Ctrl]+[V] 를 누름 |
| 도구 모음 이용 | 를 선택한 후 복사할 위치에서 를 클릭함 |
| 키보드와 마우스 사용 | <ul style="list-style-type: none"> 같은 드라이브에서 : [Ctrl]을 누른 상태에서 마우스로 복사할 위치에 끌어다 놓음 다른 드라이브에서 : 마우스로 복사할 위치에 끌어다 놓음 |

파일/폴더 이동

| | |
|------------|--|
| [편집] 메뉴 이용 | [편집] → [잘라내기]를 선택한 후 이동할 위치에서 [편집] → [붙여넣기]를 선택함 |
| 바로 가기 키 이용 | [Ctrl]+[X] 를 누른 후 이동할 위치에서 [Ctrl]+[V] 를 누름 |

| | |
|-------------|---|
| 도구 모음 이용 | 를 선택한 후 복사할 위치에서 를 클릭함 |
| 키보드와 마우스 사용 | <ul style="list-style-type: none"> 같은 드라이브에서 : 마우스로 이동할 위치에 끌어다 놓음 다른 드라이브에서 : [Shift]를 누른 채 마우스로 이동할 위치에 끌어다 놓음 |

클립보드(Clipboard)

- 데이터를 일시적으로 보관해 두는 임시 저장 공간으로, 일종의 버퍼 역할을 하며 서로 다른 응용 프로그램 간에 데이터를 쉽게 전달할 수 있다.
- 클립보드의 내용은 여러 번 사용이 가능하지만, 가장 최근에 저장된 것 하나만 기억한다.
- 시스템을 재시작하면 클립보드에 저장된 데이터는 지워진다.
- 복사(**[Ctrl]+[C]**)하거나 잘라내기(**[Ctrl]+[X]**), 붙여넣기(**[Ctrl]+[V]**), 활성창 복사(**[Alt]+[Print Screen]**), 전체 창 복사(**[Print Screen]**) 시 사용되며, 클립보드에 저장된 데이터는 클립보드 표시기를 통해 볼 수 있다.
- 클립보드 표시기를 이용하여 클립보드의 내용을 삭제하거나 다른 이름으로 저장하여 사용할 수 있다.

핵심 18.상시, 17.상시, 16.상시, 15.상시, 14.상시, 13.상시, 12.상시, 11.9, 11.2, 10.7, 10.3, 09.3, 08.10, 08.7, ...
074 WINDOWS - 휴지통 사용하기

- 삭제된 파일이나 폴더가 임시 보관되는 장소를 말한다.
- 크기는 기본적으로 드라이브 용량의 5~10%가 설정되어 있으며, 최대 100%까지 임의로 설정할 수 있다.
- 휴지통에 보관된 파일이나 폴더는 복원이 가능하며, 복원하기 전에는 사용할 수 없다.
- 휴지통 안에 있는 모든 항목을 삭제하려면 [파일] → [휴지통 비우기]를 선택한다.
- 휴지통 비우기를 수행하고 나면 복원할 수 없다.
- 휴지통 속성에서는 휴지통에 관련된 다음과 같은 여러 사항을 설정할 수 있다.
 - 휴지통의 크기를 드라이브마다 다르게 또는 모두 동일한 크기로 설정할 수 있다.
 - 파일이나 폴더를 삭제할 때 휴지통을 거치지 않고, 바로 삭제하도록 설정할 수 있다.
 - 파일이나 폴더가 삭제될 때마다 확인 메시지가 표시되도록 설정할 수 있다.



- 일반적으로 삭제된 항목은 휴지통에 임시 보관되지만 다음과 같은 경우에는 휴지통을 거치지 않고 바로 삭제되므로 복원이 불가능하다.
 - 플로피디스크, USB 메모리, DOS 모드, 네트워크 드라이브에서 삭제된 항목
 - [Shift]를 누르고 삭제 명령을 실행한 경우
 - 휴지통 속성에서 '파일을 휴지통에 버리지 않고 삭제할 때 바로 제거'를 선택한 경우
 - 같은 이름의 항목을 복사/이동 작업으로 덮어쓴 경우

핵심 17.상시, 16.상시, 15.상시, 14.상시, 13.상시, 12.상시, 11.7, 10.1, 09.3, 09.1, 08.2, 07.9, 07.4, 06.10, ...
075 WINDOWS - 제어판

| | |
|----------------------|---|
| 네트워크 또는 네트워크 및 공유 센터 | 어댑터, 프로토콜, 서비스, 클라이언트 등의 네트워크 구성 요소 추가/제거 및 컴퓨터 이름 지정 |
| 인터넷 옵션 | 인터넷 연결, 인터넷 자료 관리, 보안 영역 설정 등 인터넷 사용과 관련된 사항 설정 |
| 암호 또는 사용자 계정 | 컴퓨터의 자원 관리를 위한 암호를 설정하고, 한 PC를 사용하는 여러 사용자가 바탕 화면을 다르게 사용할 수 있도록 설정 |
| 장치 및 프린터 | 컴퓨터에 연결된 장치 및 프린터의 확인 및 추가, 속성 설정 |
| 전원 관리 또는 전원 옵션 | 절전 기능 설정으로 하드웨어 보호 및 전기 절약 |
| 글꼴 | 글꼴 추가/제거 등의 글꼴 관리 |
| 사운드 또는 소리 | 시작음, 종료음, 경고음과 같이 상황에 따라 다르게 나는 효과음 설정 |
| 국가별 설정 | 숫자, 날짜와 시간, 통화 등 국가별로 다른 기호나 단위 표시 방식 설정 |
| 날짜/시간 | 컴퓨터의 날짜와 시간 지정 |
| 프로그램 추가/제거 | <ul style="list-style-type: none"> • 응용 프로그램이나 Windows 구성 요소의 추가/제거 및 시동 디스크 작성 • 프로그램 추가/제거를 이용하여 프로그램을 제거하면 Windows가 작동하는데 영향을 미치지 않도록 깨끗하게 삭제됨 |
| 새 하드웨어 추가 | 설치되어 있는 하드웨어를 검색하고 새 하드웨어 설치 |
| 프로그램 및 기능 | 응용 프로그램 및 Windows 기능의 제거/변경/복구, 설치된 업데이트 확인 |
| 개인 설정 | 바탕 화면의 배경 그림, 창 색, 소리, 화면 보호기, 바탕 화면 아이콘, 마우스 포인터 등을 설정 |
| 디스플레이 | 바탕 화면의 배경 그림, 화면 배색, 화면 보호기, 해상도 등 화면의 표현 형식 설정 |

| | |
|-----------|--|
| 시스템 | 시스템에 설치된 모든 하드웨어 장치의 이상 여부 및 충돌 상태 점검 |
| 내게 필요한 옵션 | 신체가 불편한 사용자가 컴퓨터를 편리하게 사용하기 위한 여러 가지 옵션(키보드, 마우스, 사운드, 화면 표시 등) 설정 |
| 접근성 센터 | 컴퓨터 보기, 키보드 및 마우스 사용, 기타 입력 장치 사용 등을 보다 쉽게 사용할 수 있도록 설정 |
| 마우스 | 원손잡이용 마우스 단추 설정, 마우스 포인터 모양이나 더블클릭 속도 등 마우스 관련 사항 설정 |
| 키보드 | 키 재입력 시간, 반복 속도, 커서 깜빡임 속도 등 키보드 관련 사항 설정 |
| 멀티미디어 | <ul style="list-style-type: none"> • 오디오, 비디오, MIDI 출력, CD 재생 등 멀티미디어 장치 관련 사항 설정 • 작업 표시줄에 볼륨 조절 표시 |

핵심 16.상시, 14.상시, 13.상시, 10.10, 09.1, 08.10, 08.3, 07.7, 04.4, 02.7, 02.1, 01.4, 01.1
076 WINDOWS - 보조프로그램

보조프로그램의 종류

| | |
|-------------------|---|
| 오피스 관련 보조프로그램 | <ul style="list-style-type: none"> • 문서 작성, 그림 편집, 계산 등 일반 업무 처리에 사용하는 보조프로그램 • 종류 : 그림판, 계산기, 메모장, 워드패드, 스티커 메모, 수학적 입력란, 사용자 정의 문자 편집기, 문자표, 클립보드 표시기 등 |
| 엔터테인먼트 관련 보조 프로그램 | <ul style="list-style-type: none"> • 소리나 동영상 파일 등 멀티미디어를 재생할 때 사용하는 보조프로그램 • 종류 : 녹음기, Windows Media Player, CD 재생기, 볼륨 조절 등 |
| 통신 관련 보조프로그램 | <ul style="list-style-type: none"> • 통신망에 접속하거나 네트워크를 구성할 때 사용하는 보조프로그램 • 종류 : 원격 데스크톱 연결, 하이퍼터미널, 케이블 직접 연결, 전화 접속 네트워크 |
| 시스템 도구 관련 보조 프로그램 | <ul style="list-style-type: none"> • 시스템을 최적화 상태로 관리하기 위해 사용하는 보조프로그램 • 종류 : 디스크 정리, 디스크 조각 모음, 디스크 공간 늘림 |
| 게임 관련 보조프로그램 | <ul style="list-style-type: none"> • 간단한 게임을 할 때 사용하는 보조프로그램 • 종류 : 지뢰찾기, 카드놀이, 프리셀, 핀볼, 하트 등 |

메모장

- 특별한 서식이 필요 없는 간단한 텍스트(ASCII 형식) 파일을 작성할 수 있는 문서 작성 프로그램이다.
- 텍스트(.TXT) 형식의 문서만을 열거나 저장할 수 있다.
- 메모장에서는 그림, 차트 등의 OLE 개체를 삽입할 수 없다.



- 문서 전체에 대해서만 글꼴의 종류, 속성, 크기를 변경할 수 있다.

워드패드

- 메모장과 달리 글꼴, 글머리표, 단락 등의 다양한 서식을 적용할 수 있는 문서 작성 프로그램이다.
- 작성한 문서는 DOC, RTF, TXT 등의 확장자로 저장할 수 있다.
- RTF, TXT, DOC, WRI 등의 확장자를 가진 문서를 불러올 수 있다.
- 날짜 및 시간, OLE 개체를 삽입할 수 있다.

그림판

- 간단한 그림을 작성하거나 수정하기 위한 보조프로그램이다.
- 기본(Default) 저장 형식은 .BMP 형식의 비트맵 파일이다.
- 그림판에서는 BMP, GIF, JPG, PCX 형식의 파일을 편집할 수 있다.
- 그림판에서 작성 또는 편집한 그림은 Windows 바탕 화면의 배경으로 사용할 수 있다.

핵심 17.상시, 15.상시, 11.9, 09.1, 05.10, 99.10

077 스플 기능

- 스플(Simultaneous Peripheral Operation On-Line)이란 중앙처리장치와 같이 처리 속도가 빠른 장치와 프린터와 같이 처리 속도가 느린 장치들 간의 처리 속도 문제를 해결하기 위해 사용하는 기능이다.
- 스플링은 인쇄할 내용을 먼저 하드디스크에 저장하고 백그라운드 작업으로 CPU의 여유 시간에 틈틈이 인쇄하기 때문에, 프린터가 인쇄 중이라도 다른 응용 프로그램을 실행하는 포그라운드 작업이 가능하다.

핵심 17.상시, 16.상시, 15.상시, 14.상시, 13.상시, 12.상시, 11.7, 11.4, 10.3, 10.1, 09.7, 09.1, 08.10, 08.7, 08.3, ...

078 WINDOWS - 시스템 유지 보수 및 최적화 도구

- 종류 : 디스크 검사, 디스크 조각 모음, 백업, 디스크 정리, 디스크 공간 늘림

주요 도구

| | |
|-----------|---|
| 디스크 검사 | <ul style="list-style-type: none"> • 디스크의 손상된 부분을 점검하여 발견된 오류를 복구함 • 네트워크 드라이브, CD-ROM 드라이브는 디스크 검사를 수행할 수 없음 • 컴퓨터의 전원이 갑자기 중단되어 다시 부팅될 때는 디스크 검사가 자동으로 수행됨 • 디스크 검사는 시스템의 성능 향상을 위해 정기적으로 실행하는 것이 좋음 • 디스크 검사 결과 창에서 확인 가능한 정보 : 전체 디스크 공간, 불량 섹터, 폴더 수, 숨겨진 파일 용량과 파일 수, 사용자 파일 수, 사용할 수 있는 공간, 각 할당/전체 할당 단위, 사용할 수 있는 할당 단위 수 |
| 디스크 조각 모음 | <ul style="list-style-type: none"> • 단편화(Fragmentation)로 인해 여기저기 분산되어 저장된 파일들을 연속된 공간으로 최적화시켜 디스크의 접근 속도를 향상시키는 기능이 있음 • 디스크를 효율적으로 사용하기 위해 파일을 정리하는 것으로, 디스크의 용량 증가와는 관계가 없음 • Windows가 지원하지 않는 형식으로 압축된 프로그램, CD-ROM 드라이브, 네트워크 드라이브는 디스크 조각 모음을 수행할 수 없음 • 디스크 조각 모음을 수행하면 디스크 공간의 최적화가 이루어져 접근 속도와 안정성이 향상됨 • 디스크 조각 모음을 수행하는데 걸리는 시간은 볼륨에 있는 파일의 수, 파일 크기, 조각난 양 등에 따라 달라질 수 있음 • 디스크 조각 모음을 수행하는 동안 다른 작업을 할 수는 있지만, 모든 작업을 중지한 상태에서 수행해야 효율적으로 조각 모음을 할 수 있음 |

핵심 18.상시, 17.상시, 16.상시, 15.상시, 14.상시, 13.상시, 12.상시, 11.9, 11.7, 11.2, 10.7, 10.3, 09.7, 09.1, ...

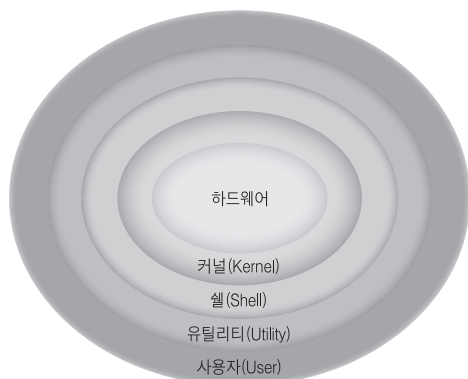
079 UNIX의 특징

- UNIX는 주로 서버용 컴퓨터에서 사용되는 운영체제이다.
- 시분할 시스템(Time Sharing System)을 위해 설계된 대화식 운영체제로, 소스가 공개된 개방형 시스템(Open System)이다.
- 대부분 C 언어로 작성되어 있어 이식성이 높으며 장치, 프로세스 간의 호환성이 높다.
- 크기가 작고 이해하기가 쉽다.
- 다중 사용자(Multi-User), 다중 작업(Multi-Tasking)을 지원한다.
- 많은 네트워킹 기능을 제공하므로 통신망(Network) 관리용 운영체제로 적합하다.
- 계층적 트리(Tree) 구조의 파일 시스템을 갖는다.
- CUI(Character User Interface) 환경을 제공한다.



- 하나 이상의 작업을 백그라운드에서 수행하므로 여러 작업을 동시에 처리할 수 있다.

080 UNIX 시스템의 구성



시스템의 구성

커널(Kernel)

- UNIX의 가장 핵심적인 부분이다.
- 컴퓨터가 부팅될 때 주기억장치에 적재된 후 상주하면서 실행된다.
- 하드웨어를 보호하고, 프로그램과 하드웨어 간의 인터페이스 역할을 담당한다.
- 프로세스(CPU 스케줄링) 관리, 기억장치 관리, 파일 관리, 입·출력 관리, 데이터 전송 및 변환 등 여러 가지 기능을 수행한다.

셸(Shell)

- 사용자의 명령어를 인식하여 프로그램을 호출하고 명령을 수행하는 명령어 해석기이다.
- 명령을 해석하여 커널로 처리할 수 있도록 전달해주는 명령 인터프리터로, 단말장치를 통하여 사용자로부터 명령어를 입력받는다.
- DOS의 COMMAND.COM과 같은 기능을 수행한다.
- 주기억장치에 상주하지 않고, 명령어가 포함된 파일 형태로 존재하며 보조기억장치에서 교체 처리가 가능하다.
- 공용 셸(Bourne Shell, C Shell, Korn Shell)이나 사용자 자신이 만든 셸을 사용할 수 있다.

| | |
|----------|---|
| Bourne 셸 | <ul style="list-style-type: none"> • UNIX 시스템의 최초의 셸 • 기본 프롬프트는 \$임 |
|----------|---|

| | |
|--------|---|
| C 셸 | <ul style="list-style-type: none"> • 대부분 가장 많이 사용되는 셸로, C 언어와 유사함 • 기본 프롬프트는 %임 |
| Korn 셸 | Bourne 셸을 기반으로 C 셸의 기능을 추가한 것 |

유틸리티(Utility)

- 일반 사용자가 작성한 응용 프로그램을 처리하는 데 사용한다.
- DOS에서의 외부 명령어에 해당한다.
- 유틸리티 프로그램에는 에디터, 컴파일러, 인터프리터, 디버거 등이 있다.

잠깐만요! UNIX 시스템 편집기(Editor)

UNIX 시스템이 제공하는 편집기에는 vi, ed, ex, emacs, pico, joe 등이 있습니다.

081 UNIX - 시스템 및 프로세스 관련 명령어

| 명령어 | 기능 |
|--------|---|
| kill | <ul style="list-style-type: none"> • 현재 실행중인 프로세스를 종료(삭제)함 • 편집 상태에서는 한 줄 전체를 지움 |
| fork | 새로운 프로세스를 생성(하위 프로세스 호출, 프로세스 복제)함 |
| exec | 새로운 프로세스를 수행함 |
| finger | 현재 시스템에 등록되어 있는 사용자 정보를 조회함 |
| ps | <ul style="list-style-type: none"> • 현재 작업중인 프로세스의 상태 정보를 표시함 • 프로세스의 이름, 명령어 이름, 프로세스 ID 번호, 제어 단말기와 소유주를 포함하는 속성의 목록, 수행된 시간 등을 표시함 |
| ping | 특정 시스템과 접속이 안 될 경우 네트워크 상의 문제를 진단함 |
| login | UNIX 작업을 위해 사용자 ID와 비밀번호를 입력받아 사용자를 확인함 |
| logout | UNIX 시스템에 대한 접속을 종료함(Ctrl+D, exit) |
| passwd | 로그인할 때 필요한 비밀번호를 설정 또는 변경함 |
| who | <ul style="list-style-type: none"> • 로그인한 사용자에게 관한 정보를 표시함 • 단말기 이름, 로그인 이름, 로그인 일시, 사용중인 단말기 번호 등을 표시함 |
| mount | 파일 시스템을 마운팅하거나 마운팅을 해제함 |



핵심 17.상시, 16.상시, 15.상시, 14.상시, 13.상시, 12.상시, 11.4, 10.1, 09.9, 09.7, 09.3, 09.1, 08.10, 08.2, ...
082 UNIX – 디렉터리 관련 명령어

| UNIX 명령어 | DOS 명령어 | 기능 |
|----------|---------|-----------------------------|
| pwd | | 현재 작업중인 디렉터리 경로를 화면에 출력함 |
| ls | dir | 현재 작업중인 디렉터리의 모든 파일을 표시함 |
| mkdir | md | 디렉터를 생성함 |
| rmdir | rd | 디렉터를 제거함 |
| cd | cd | 현재 작업중인 디렉터리에서 다른 디렉터리로 이동함 |

핵심 18.상시, 17.상시, 16.상시, 15.상시, 14.상시, 13.상시, 11.9, 11.7, 11.2, 10.7, 09.9, 08.10, 08.7, 08.3, 07.7, ...
083 UNIX – 파일 관련 명령어

| UNIX 명령어 | DOS 명령어 | 기능 |
|----------|-------------|-------------------------------------|
| creat | | 파일 생성 |
| open | | 파일을 사용할 수 있는 상태로 준비 시킴 |
| cp | copy, xcopy | 파일 복사 |
| rm | del | 파일 삭제 |
| cat | type | 파일의 내용을 화면에 표시함 |
| chmod | attrib | 파일의 보호 모드를 설정하여 파일의 사용 허가를 지정함 |
| chown | | 소유자 변경 |
| find | | 파일 찾기 |
| mv | | 파일을 이동시키거나 이름을 변경함 |
| cmp | | 두 파일을 비교하여 차이가 나는 바이트 위치와 행 번호를 표시함 |

4과목 · 정보 통신 일반

핵심 16.상시, 15.상시, 04.4, 03.7, 03.1, 02.1, 01.7, 00.10, 00.3, 99.10
084 정보 통신의 정의

- 컴퓨터와 통신 기술의 결합에 의해 통신 처리 기능과 정보 처리 기능은 물론 정보의 변환, 저장 과정이 추가된 형태의 통신으로 이용약관은 정보통신관계법령으로 규정하고 있다.
- 일반적으로 정보 통신의 범주에 포함되는 데이터 통신을 정보 통신 또는 컴퓨터 통신이라고 한다.
- 정보 통신 = 전기 통신(정보 전송) + 컴퓨터(정보 처리)

| | |
|------------|---|
| 전기 통신 | <ul style="list-style-type: none"> 전기 통신(Telecommunication)은 전기적인 신호를 전달 매체로 하여 아날로그 형태의 음성 정보를 송·수신하는 것을 말함 전기 통신은 1837년 모스 전신기의 발명으로 시작되었음 |
| 데이터 통신 | <ul style="list-style-type: none"> 컴퓨터의 발달을 배경으로 하여 생겨난 것으로, 컴퓨터와 각종 통신 기기(단말기) 사이에서 디지털 형태로 표현된 2진 정보(0과 1)를 송·수신하는 것을 말함 ITU-T의 정의 : 정보를 기계로 처리하거나 처리한 정보를 전송하는 것 데이터 통신 = 데이터 전송 기술 + 데이터 처리 기술 |
| 정보 통신의 3요소 | <ul style="list-style-type: none"> 정보원(Source) : 정보를 입력받아 전송하는 장소 수신원(Destination) : 전송된 정보를 수신하는 장소 전송 매체(Transmission Medium) : 정보원과 수신원을 연결하는 매체(통신 회선) |

핵심 17.상시, 13.상시, 07.9, 07.1, 05.10, 03.10, 03.1, 99.3
085 정보 통신 시스템의 특징

- 정보 통신은 정보 통신 시스템에 의해 구현되는 것으로 정보 통신의 특징은 엄격히 구분하면 정보 통신 시스템의 특징이라 할 수 있다.
- 고속 · 고품질의 전송이 가능하다.
- 고도의 오류 제어 방식을 사용하여 시스템의 신뢰도가 높다.
- 대형 컴퓨터와 대용량 파일을 공동으로 이용할 수 있다.
- 분산 처리가 가능하다.
- 대용량 · 광대역 전송이 가능하다.
- 거리와 시간의 한계를 극복한다.



- 통신 비밀을 유지하기 위한 보안 시스템의 개발이 필요하다.
- 정보 통신을 운용하기 위해 소프트웨어 기술이 필요하다.

핵심 086 정보 통신 시스템의 기본 구성

- 데이터 전송계 : 단말장치(DTE), 데이터 전송 회선(신호 변환장치(DCE), 통신 회선), 통신 제어장치(CCU)
- 데이터 처리계 : 컴퓨터(하드웨어, 소프트웨어)

| | |
|--|---|
| 단말장치(DTE; Data Terminal Equipment) | <ul style="list-style-type: none"> • 단말장치는 데이터 통신 시스템과 외부 사용자의 접속점에 위치하여 최종적으로 데이터를 입·출력하는 장치임 • 입·출력 기능, 전송 제어 기능, 기억 기능을 수행함 |
| 신호 변환장치 (DCE; Data Circuit Equipment) | 신호 변환장치는 컴퓨터나 단말장치의 데이터를 통신 회선에 적합한 신호로 변경하거나, 통신 회선의 신호를 컴퓨터나 단말장치에 적합한 데이터로 변경하는 신호 변환 기능을 수행하며, 통신 회선의 종류에 따라 사용되는 기기가 달라짐 |
| 통신 회선 | <ul style="list-style-type: none"> • 통신 회선은 단말장치에 입력된 데이터 또는 컴퓨터에서 처리된 결과가 실질적으로 전송되는 전송 선로임 • 유선 매체 : 꼬임선, 동축 케이블, 광섬유 케이블 등 • 무선 매체 : 라디오파, 지상 마이크로파, 위성 마이크로파 등 |
| 통신 제어장치 (CCU; Communication Control Unit) | <ul style="list-style-type: none"> • 통신 제어장치는 통신 회선과 주 컴퓨터 사이에 위치하여 이들을 전기적으로 연결하고 각종 제어 기능을 수행하는 장치임 • 기능 : 전기적 결합, 회선 제어, 전송 제어, 동기 및 오류 제어 등 |
| 컴퓨터(Host) | <ul style="list-style-type: none"> • 단말장치에서 보낸 데이터를 처리하는 곳으로, 처리된 데이터는 다시 통신 회선을 통해 단말장치로 전달됨 • 정보 통신 시스템의 구성 요소 중 일반적으로 고장 발생률(에러율)이 가장 낮음 • 하드웨어 : 중앙처리장치, 주변장치 • 소프트웨어 : 시스템 소프트웨어(운영체제), 응용 소프트웨어 |

- 온라인 시스템의 3대 구성 요소
단말장치(단말기), 통신 회선(전송 매체), 전송 제어장치
- 전처리기(FEP, Front End Processor)
- 통신 제어장치의 일종으로 여러 단말장치들의 전송

- 신호를 하나의 고속 통신 회선으로 호스트(중앙) 컴퓨터에 연결한다.
- 통신 회선 및 단말장치 제어, 메시지의 조립과 분해, 전송 메시지 검사, 오류(Error) 검출 등을 수행하므로, 컴퓨터의 부담이 적어진다.

핵심 087 정보 통신 시스템의 처리 형태

오프라인 시스템(Off-Line System)

- 단말장치와 컴퓨터가 통신 회선으로 직접 연결되어 있지 않은 형태로, 중간에 사람 혹은 기록 매체가 개입된다.
- 자료 입력과 데이터의 처리를 위한 정보 운반을 사람이 직접 수행한다.
- 단말장치와 컴퓨터 사이에 데이터를 임시로 보관할 기록 매체가 필요하다.

온라인 시스템(On-Line System)

- 데이터가 발생한 단말장치와 데이터를 처리할 컴퓨터가 통신 회선을 통해 직접 연결된 형태로, 데이터 송·수신 중간에 사람 혹은 기록 매체가 개입되지 않는다.
- 정보 통신 업무의 대부분을 차지하는 실시간 처리가 요구되는 작업에 주로 사용된다.
- 단말장치, 중앙처리장치, 통신 제어장치, 통신 회선 등으로 구성된다.

일괄 처리 시스템(Batch Processing System)

- 처리할 데이터를 일정 양이나 일정 시간 동안 모아두었다가 한꺼번에 전달하여 처리하고, 처리된 결과 또한 일정 양이나 일정 시간 동안 모아두었다가 전달하는 방식이다.
- 급여 관리, 긴급을 요하지 않는 과학 기술 업무 등에 사용된다.

실시간 처리 시스템(Real Time Processing System)

- 데이터가 발생한 즉시 처리하여 그 결과를 되돌려주는 방식이다.
- 은행 업무, 예약 업무, 각종 조회 업무 등에 사용된다.



| | |
|--------------------------------------|--|
| 거래 처리 (Transaction Processing) | <ul style="list-style-type: none"> 발생된 거래 상황을 직접 컴퓨터에 입력하여 처리한 후 그 결과를 바로 전달하는 방식 은행의 입·출금 업무, 증명서 발급 업무, 각종 예약 업무 등에 활용 |
| 질의/응답 (Inquiry/ Response) | <ul style="list-style-type: none"> 중앙의 데이터베이스에 대량의 정보를 저장한 후 필요한 정보를 질의하면 그에 맞는 응답을 즉시 전달하는 방식 각종 정보 검색(신용(Credit) 점검, 은행 잔고 검색 등), 주식 시세, 일기 예보 등에 활용 |
| 메시지 교환 (Message Switching) | <ul style="list-style-type: none"> 단말장치로부터 입력된 데이터를 요구된 특정 단말장치로 전달하는 방식 은행의 계좌 이체 업무 등에 활용 |

시분할 처리 시스템(Time Sharing System)

- 컴퓨터를 사용할 수 있는 시간을 일정하게 쪼개 여러 대의 단말장치가 정해진 시간(Time Slice) 동안 번갈아 가며 컴퓨터를 사용하는 방식이다.
- 컴퓨터의 처리 속도가 빨라짐에 따라 가능해진 방식으로, 단말장치가 정해진 시간 동안 컴퓨터를 사용한 후 다시 사용하기까지의 쉬는 시간이 짧아 사용자는 컴퓨터를 혼자 독점하여 사용하는 것처럼 느끼게 된다.
- 단말장치 사용자와 컴퓨터 사이의 정보 전송이 대화하듯이 수행되므로, 대화형 처리 방식이라고도 한다.

088 유선 매체

전송 선로의 특성

- 1차 정수 : 저항(R), 정전용량(C), 인덕턴스(L), 누설 컨덕턴스(G)

잠깐만요 ① 무왜곡 전송

송신 측에서 전송한 신호가 일그러짐 없이 수신 측에 수신되는 무왜곡 전송의 조건은 $RC = LG$ 입니다.

- 2차 정수 : 감쇠정수, 위상정수, 전파정수, 특성 임피던스

꼬임선(이중 나선)

- 가장 최초로 사용된 통신 선로로 2개의 절연된 구리선을 서로 꼬아 하나의 전송 선로를 구성하며, 여러 쌍(Pair)의 꼬임선들을 하나로 묶어 케이블을 형성한다.
- 구리선을 꼬는 이유는 두 선 사이의 전기적 간섭을 최소화하기 위해서이다.
- 가격이 저렴하고, 설치가 간편하다.
- 거리, 대역폭, 데이터 전송률 면에서 제약이 많다.

- 다른 전기적 신호의 간섭이나 잡음에 영향을 받기가 쉽다.

동축 케이블

- 중심 도체를 플라스틱 절연체를 이용하여 감싸고, 이를 다시 외부 도체를 이용하여 감싸는 형태로 구성된다.
- 꼬임선에 비해 외부 간섭과 누화의 영향이 적다.
- 고주파 특성이 양호하며, 광대역 전송에 적합하다.
- CATV, 근거리 통신망, 장거리 전화 등에 다양하게 사용된다.

광섬유 케이블

- 석영유리를 원료로 하여 제작된 가느다란 광섬유를 여러 가닥 묶어서 케이블의 형태로 만든 것으로, 광 케이블이라고도 한다.
- 데이터를 전기 신호가 아닌 빛으로 바꾸어 빛의 전반사 원리를 이용하여 전송한다.
- 유선 매체 중 가장 빠른 속도와 높은 주파수 대역폭을 제공한다.
- 넓은 대역폭을 제공하므로 데이터의 전송률이 높다.
- 대용량, 장거리 전송이 가능하며, 가볍고 가볍다.
- 도청하기 어려워 보안성이 뛰어나다.
- 광섬유 케이블의 원료인 유리는 절연성이 좋아 전자 유도의 영향을 받지 않으므로(무유도성), 전자기적인 문제가 최소화되어 안정된 통신 및 누화 방지가 가능하다.

광통신의 3요소

| | |
|--------------------------|--------------------------------|
| 발광기 (LD; Laser Diode) | 전광 변환(전기 에너지 → 빛 에너지), 송신 측 요소 |
| 수광기 (PD; Photo Diode) | 광전 변환(빛 에너지 → 전기 에너지), 수신 측 요소 |
| 광심선(광케이블) | 중계부로 석영, 유리로 구성 |

광섬유 케이블의 구성

| | |
|---------------------------------|---|
| 코어(Core) | 빛이 전파되는 영역으로, 클래딩보다 높은 굴절률을 가짐 |
| 클래딩 (Cladding), 클래드(Clad) | 코어보다 약간 낮은 굴절률을 가지므로 코어의 빛이 외부로 빠져나가지 못하게 하고, 코어를 외부로부터 보호함 |
| 재킷(Jacket) | 습기, 마모, 파손 등의 위협으로부터 내부를 보호함 |



핵심 18.상시, 17.상시, 16.상시, 15.상시, 14.상시, 13.상시, 12.상시, 11.9, 11.2, ...

089 통신 속도 / 통신 용량

변조 속도

- 1초 동안 몇 개의 신호 변화가 있었는가를 나타내는 것으로, 단위는 Baud를 사용한다.
- 1개의 신호가 변조되는 시간을 T초라고 할 때 변조 속도는 $Baud = 1/T$ 이다.

신호 속도

- 신호 속도는 1초 동안 전송 가능한 비트의 수를 나타내는 것으로, 단위는 Bps(bit/sec)를 사용한다. 예를 들어, 2.4Kbps는 1초에 2,400개의 비트를 전송한다는 의미이다.
- 데이터 신호 속도는 다음과 같이 구할 수 있다.

$$\text{데이터 신호 속도(Bps)} = \text{변조 속도(Baud)} \times \text{변조 시 상태 변화 수}$$

- 데이터 신호 속도의 계산식에 의해 변조 속도를 다음과 같이 구할 수 있다.

$$\text{변조 속도(Baud)} = \text{데이터 신호 속도(Bps)} / \text{변조 시 상태 변화 수}$$

- 변조 시 상태 변화 수가 1Bit인 경우에는 Bps와 Baud가 같다.
- 변조 시 상태 변화 수 : 모노비트(Monobit) = 1비트, 디비트(Dibit) = 2비트, 트리비트(Tribit) = 3비트, 쿼드비트(Quadbit) = 4비트

전송 속도

- 단위 시간에 전송되는 데이터의 양(문자, 블록, 비트, 단어 수 등)을 나타낸다.
- 단위는 특별히 정해진 것은 없으나 보통 문자/초(분, 시간), 블록/초(분, 시간), 단어/초(분, 시간), 비트/초(분, 시간) 등과 같은 형식으로 나타낸다.

통신 용량

- 샤논(Shannon)의 정의

$$C = W \log_2(1+S/N) [\text{bps}]$$

- C : 통신 용량
- W : 대역폭(대역폭이 'Band Width'이므로 'W' 대신 'B'로도 사용)
- S : 신호 전력
- N : 잡음 전력

- 전송로의 통신 용량을 늘리기 위한 방법
 - 주파수 대역폭을 늘린다.
 - 신호 세력을 높인다.
 - 잡음 세력을 줄인다.

핵심

17.상시, 16.상시, 15.상시, 14.상시, 10.1, 09.3, 05.1, 03.10, 03.3, 03.1, 02.10, 02.7, 02.4, 01.10, 00.10, 00.8, ...

090 통신 방식

| | |
|---------------------|---|
| 단방향(Simplex) 통신 | 한쪽 방향으로만 전송이 가능한 방식 예) 라디오, TV |
| 반이중(Half-Duplex) 통신 | <ul style="list-style-type: none"> 양방향 전송이 가능하지만 동시에 양쪽 방향에서 전송할 수 없는 방식 2선식 선로를 사용하여 송신과 수신을 번갈아 전송 예) 무전기, 모뎀을 이용한 데이터 통신 |
| 전이중(Full-Duplex) 통신 | <ul style="list-style-type: none"> 동시에 양방향 전송이 가능한 방식 4선식 선로를 사용하며, 주파수 분할을 이용할 경우 2선식도 가능 전송량이 많고, 전송 매체의 용량이 클 때 사용 예) 전화, 전용선을 이용한 데이터 통신 |

핵심

18.상시, 17.상시, 15.상시, 14.상시, 13.상시, 12.상시, 11.7, 11.4, 11.2, 09.3, 08.2, 07.7, 07.4, 06.4, 05.1, 03.10, ...

091 신호 변환 방식 - 디지털 변조

디지털 변조란 디지털 데이터를 아날로그 신호로 변환하는 것을 의미하며, 모뎀(MODEM)을 이용한다.

| | |
|---|---|
| 진폭 편이 변조 (ASK; Amplitude Shift Keying) | <ul style="list-style-type: none"> 진폭 편이 변조는 2진수 0과 1을 서로 다른 진폭의 신호로 변조하는 방식임 진폭 편이 변조 방식을 사용하는 모뎀은 구조가 간단하고, 가격이 저렴함 신호 변동과 잡음에 약하여 데이터 전송용으로는 거의 사용되지 않음 |
| 주파수 편이 변조 (FSK; Frequency Shift Keying) | <ul style="list-style-type: none"> 주파수 편이 변조는 2진수 0과 1을 서로 다른 주파수로 변조하는 방식임 1,200bps 이하의 저속도 비동기식 모뎀에서 사용됨 주파수 편이 변조 방식을 사용하는 모뎀은 구조가 간단하고, 신호 변동과 잡음에도 강함 대역폭을 넓게 차지함 주파수 변조 지수 = 최대 주파수 편이 / 변조 신호 주파수 |



| | |
|--|---|
| 위상 편이 변조 (PSK; Phase Shift Keying) | <ul style="list-style-type: none"> 위상 편이 변조는 2진수 0과 1을 서로 다른 위상을 갖는 신호로 변조하는 방식임 한 위상에 1비트(2위상), 2비트(4위상), 또는 3비트(8위상)를 대응시켜 전송하므로, 속도를 증가시킬 수 있음 중·고속의 동기식 모뎀에 많이 사용함 종류 : 2위상 편이 변조(DPSK), 4위상 편이 변조(QPSK, QDPSK), 8위상 편이 변조(ODPSK) |
| 직교 진폭 변조, 진폭 위상 변조 (QAM; Quadrature Amplitude Modulation) | <ul style="list-style-type: none"> 진폭과 위상을 상호 변환하여 신호를 얻는 변조 방식임 제한된 전송 대역 내에서 고속 전송이 가능함(9,600bps) |

092 신호 변환 방식 - 펄스 코드 변조(PCM)

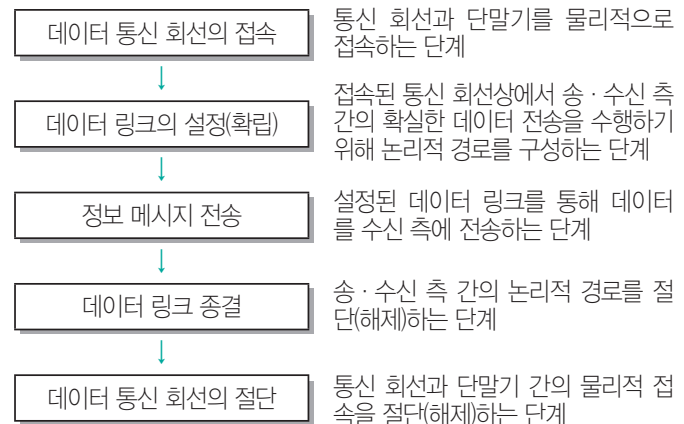
- 화상, 음성, 동영상 비디오, 가상 현실 등과 같이 연속적인 시간과 진폭을 가진 아날로그 데이터를 디지털 신호로 변환하는 것으로, 코덱(CODEC)을 이용한다.
- 누화의 영향을 거의 받지 않는다.
- 레벨 변동이 거의 없다.
- 점유 주파수 대역폭이 크다.
- 펄스 코드 변조(PCM) 순서 : 표본화 → 양자화 → 부호화 → 복호화 → 여과화

| | |
|---------------------|--|
| 표본화 (Sampling) | <ul style="list-style-type: none"> 음성, 영상 등의 연속적인 신호 파형을 일정 시간 간격으로 검출하는 단계로, 사논의 표본화 이론을 바탕으로 함 사논(나이퀴스트)의 표본화 이론 : 어떤 신호 f_0가 의미를 지니는 최고의 주파수보다 2배 이상의 속도로 균일한 시간 간격 동안 채집된다면, 이 채집된 데이터는 원래의 신호가 가진 모든 정보를 포함함 |
| 양자화 (Quantizing) | <ul style="list-style-type: none"> 표본화된 PAM 신호를 유한 개의 부호에 대한 대 표값으로 조정하는 과정 양자화 잡음 : 표본 측정값과 양자화 파형의 오차를 말하는 것으로, 주로 PCM 단국 장치에서 발생함 양자화 레벨 : PAM 신호를 부호화할 때 2진수로 표현할 수 있는 레벨(단계)을 말함 |
| 부호화 (Encoding) | 양자화된 PCM 펄스의 진폭 크기를 2진수(1과 0)로 표시하는 과정 |
| 복호화 (Decoding) | 수신된 디지털 신호, 즉 PCM 신호를 PAM 신호로 되돌리는 단계 |
| 여과화 (Filtering) | PAM 신호를 원래의 입력 신호인 아날로그 신호로 복원하는 과정 |

093 전송 제어 절차

전송 제어(Transmission Control)는 정확하고 원활한 데이터의 흐름을 위해 송·수신 측 간의 상호 확인 및 사전 처리를 수행하는 것을 말한다.

전송 제어 절차



094 전송 제어 문자

| | |
|--------------------------------|--------------------------------|
| SYN(SYNchronous idle) | 문자 동기 |
| SOH(Start Of Heading) | 헤딩의 시작 |
| STX(Start of TeXt) | 본문의 시작 및 헤딩의 종료 |
| ETX(End of TeXt) | 본문의 종료 |
| ETB(End of Transmission Block) | 블록의 종료 |
| EOT(End Of Transmission) | 전송 종료 및 데이터 링크의 해제 |
| ENQ(ENquiry) | 상대편에 데이터 링크 설정 및 응답 요구 |
| DLE(Data Link Escape) | 전송 제어 문자 앞에 삽입하여 전송 제어 문자임을 알림 |
| ACK(ACKnowledge) | 수신된 메시지에 대한 긍정 응답 |
| NAK(Negative Acknowledge) | 수신된 메시지에 대한 부정 응답 |



핵심 18.상시, 16.상시, 15.상시, 10.3, 10.1, 09.7, 09.1, 05.7, 04.4, 03.1, 00.3, 99.10, 99.7

095 오류 발생 원인

| | |
|----------------------------------|---|
| 감쇠(Attenuation) | 전송 신호 세력이 전송 매체를 통과하는 과정에서 거리에 따라 약해지는 현상 |
| 지연 왜곡 (Delay Distortion) | 하나의 전송 매체를 통해 여러 신호를 전달했을 때 주파수에 따라 그 속도가 달라짐으로써 생기는 오류 |
| 백색 잡음 (White Noise) | 전송 매체 내부에서 온도에 따라 전자의 운동량이 변화함으로써 생기는 잡음으로 열잡음이라고도 함 |
| 누화 잡음 = 혼선 (Cross Talk Noise) | 인접한 전송 매체의 전자기적 결합에 의해 다른 회선에 영향을 주는 상호 유도 작용에 의해 생기는 잡음 |
| 충격성 잡음 (Impulse Noise) | <ul style="list-style-type: none"> 회선 접촉 같은 외부적인 충격 또는 통신 시스템의 결함이나 파손 등의 기계적인 충격에 의해 생기는 잡음 순간적으로 일어나는 높은 진폭의 잡음으로, 비연속적이고 불규칙적인 진폭을 갖음 디지털 데이터를 전송하는 경우 중요한 오류 발생의 요인이 됨 |
| 위상 지터 잡음 (Phase Jitter Noise) | 전송 네트워크에서 전송 신호의 위상이 연속적으로 일그러지는 현상 |
| 위상 히트 잡음 (Phase Hit Noise) | 전송 네트워크에서 전송 신호의 위상이 불연속적으로 순간 변화가 일어나는 현상 |

핵심 17.상시, 16.상시, 15.상시, 14.상시, 11.7, 10.7, 09.3, 09.1, 08.3, 05.4, 01.7

096 자동 반복 요청(ARQ)

- 오류 발생 시 수신 측은 오류 발생을 송신 측에 통보하고, 송신 측은 오류 발생 블록을 재전송하는 모든 절차를 의미한다.
- ARQ 방식은 에러 검출 방식으로 체크섬, 패리티 검사, CRC 등을 사용한다.

| | |
|------------------------------|--|
| 정지-대기 (Stop-and-Wait) ARQ | <ul style="list-style-type: none"> 송신 측에서 한 개의 블록을 전송한 후 수신 측으로부터 응답을 기다리는 방식 구현 방법은 단순하지만, 전송 효율이 떨어짐 |
| 연속 (Continuous) ARQ | <ul style="list-style-type: none"> 연속적으로 데이터 블록을 보내는 방식 Go-Back-N ARQ : 오류가 발생한 블록 이후의 모든 블록을 재전송함 선택적 재전송(Selective Repeat) ARQ : 오류가 발생한 블록만을 재전송하는 방식으로, 수신 측에서 데이터를 처리하기 전에 원래 순서대로 조립해야 하므로, 더 복잡한 논리회로와 큰 용량의 버퍼가 필요함 |

적응적
(Adaptive)
ARQ

- 데이터 블록의 길이를 채널의 상태에 따라 그때그때 동적으로 변경하는 방식으로, 전송 효율이 제일 좋음
- 제어회로가 복잡하고, 비용이 많이 들어 현재는 거의 사용되지 않음

핵심 18.상시, 17.상시, 16.상시, 13.상시, 11.7, 11.4, 10.1, 07.1, 03.7, 02.10, 02.1, 01.1, 00.10, 99.10

097 오류 검출 방식

| | |
|----------------------------|--|
| 패리티(Parity) 검사 | <ul style="list-style-type: none"> 전송 비트들 중 값이 1인 비트의 개수가 짝수 또는 홀수가 되도록 패리티 비트를 부여함 <ul style="list-style-type: none"> 짝수(우수) 패리티 : 각 전송 비트 내에 1의 개수가 짝수가 되도록 하는 것으로, 주로 비동기식 전송에 사용 홀수(기수) 패리티 : 각 전송 비트 내에 1의 개수가 홀수가 되도록 하는 것으로, 주로 동기식 전송에 사용 가장 간단한 방식이지만, 2개의 비트에 오류가 동시에 발생하면 검출이 불가능함 오류를 검출만 할 수 있고, 수정은 하지 못함 |
| 해밍 코드 (Hamming Code) 방식 | <ul style="list-style-type: none"> 수신 측에서 오류가 발생한 비트를 검출한 후 직접 수정하는 방식 오류 검출은 물론 스스로 수정까지 하므로 자기 정정 부호라고도 함 1비트의 오류만 수정이 가능하며, 정보 비트 외에 잉여 비트가 많이 필요함 전송 비트 중 1, 2, 4, 8, 16, 32, 64, ... , 2ⁿ번째를 오류 검출을 위한 패리티 비트로 사용하며, 이 비트의 위치는 변하지 않음 송신한 데이터와 수신한 데이터의 각 대응하는 비트 중 서로 다른 비트의 수를 해밍 거리(Hamming Distance)라고 함 정정 가능한 최대 오류의 수 : $d_{min} \geq 2tc+1$ (d_{min} : 최소 해밍 거리, tc : 정정 가능 오류 수) |

핵심 18.상시, 15.상시, 14.상시, 13.상시, 12.상시, 09.9, 09.7, 08.10, 07.4, 06.10, 05.10, 05.7, 04.4, 04.2, ...

098 DTE / DCE 접속 규격

- 단말장치(DTE; Data Terminal Equipment)
 - 통신 시스템과 사용자의 접점에 위치하여 컴퓨터(Host)에 의해 처리될 데이터를 입력하거나 처리된 결과를 출력하는 기능을 한다.
 - 원격지에 설치된 입·출력장치이다.
- DTE/DCE 접속 규격이란 단말장치(DTE)와 회선 종단장치(DCE) 간의 접속을 정확하게 수행하기 위한 기계적, 전기적, 기능적, 절차적 조건을 사전에 정의해 놓은 규격으로, OSI 참조 모델의 물리 계층에 관계된다.



• DTE와 DCE 접속 특성

- DTE/DCE 접속 규격은 기계적, 전기적, 기능적, 절차적 특성을 고려하여 수행되어야 한다.
- 기계적 특성 : 접속 핀의 크기, 개수, 핀 사이의 간격 등에 관한 물리적 특성
- 전기적 특성 : 전압 레벨의 사용에 관한 특성
- 기능적 특성 : 각 핀마다 설정할 기능에 대한 특성
- 절차적 특성 : 신호의 전송 절차 등에 관한 특성

• 접속 규격 표준안

| | | |
|-------|----------|---|
| ITU-T | V 시리즈 | <ul style="list-style-type: none"> • 공중 전화 교환망(PSTN)과 같은 아날로그 통신망을 통한 DTE/DCE 접속 규격 • V.24 : 기능적, 절차적 조건에 대한 규정 • V.28 : 전기적 조건에 대한 규정 |
| | X 시리즈 | <ul style="list-style-type: none"> • 공중 데이터 교환망(PSDN)과 같은 디지털 통신망을 통한 DTE/DCE 접속 규격 • X.20 : 비동기식 전송을 위한 DTE/DCE 접속 규격 • X.21 : 동기식 전송을 위한 DTE/DCE 접속 규격 • X.25 : 패킷 전송을 위한 DTE/DCE 접속 규격 |
| EIA | RS-232C | <ul style="list-style-type: none"> • 공중 전화 교환망(PSTN)을 통한 DTE/DCE 접속 규격 • V.24, V.28, ISO 2110을 사용하는 접속 규격과 기능적으로 호환성을 가지며, 현재 가장 많이 사용됨 |
| | RS-449 | <ul style="list-style-type: none"> • 고속 데이터 통신을 위한 DTE/DCE 접속 규격 • RS-232C의 단점을 보완하기 위한 새로운 표준 • 거리에 제한이 없고, RS-232C에 비해 속도가 빠름 |
| ISO | ISO 2110 | <ul style="list-style-type: none"> • 공중 전화 교환망(PSTN)을 통한 DTE/DCE 접속 규격 • 주로 기계적 조건에 대한 규정임 |

• 주요 핀의 기능

| 핀 번호 | 핀 이름 | 기능 |
|------|--------------------------|---|
| 1 | FG(Frame Ground) | 보호용 접지 회로 |
| 2 | TXD(Transmitted Data) | <ul style="list-style-type: none"> • 송신 데이터 • 데이터를 송신하는 기능 |
| 3 | RXD(Received Data) | <ul style="list-style-type: none"> • 수신 데이터 • 데이터를 수신하는 기능 |
| 4 | RTS(Request To Send) | <ul style="list-style-type: none"> • 송신 요청 • DTE에서 DCE한테 송신을 요청하거나 DTE가 DCE에게 송신을 하려고 할 때 알리는 기능 |
| 5 | CTS(Clear To Send) | <ul style="list-style-type: none"> • 송신 준비 완료 • DCE에서 DTE한테 송신 준비 완료를 알리는 기능 |
| 6 | DSR(Data Set Ready) | <ul style="list-style-type: none"> • DCE 정상 상태 • DCE의 동작 상태를 알리는 기능(On : 동작, Off : 동작 안함) |
| 7 | SG(Signal Ground) | <ul style="list-style-type: none"> • 신호 접지 • 모든 신호의 기준 전압으로 되어 있음 |
| 8 | DCD(Data Carrier Detect) | <ul style="list-style-type: none"> • 수신선 신호 감지 • DCE가 선로 쪽으로부터 감지할 수 있는 신호를 수신하고 있음을 DTE에게 알리는 기능 |
| 20 | DTR(Data Terminal Ready) | <ul style="list-style-type: none"> • DTE 정상 상태 • DTE가 정상 동작을 하고 있음을 DCE에게 알리는 기능 |
| 22 | RI(Ring Indicator) | <ul style="list-style-type: none"> • 링 감지 신호 • 상대방으로부터 링 신호가 들어오고 있다는 것을 DTE에 알리는 기능 |

핵심 17.상시, 15.상시, 14.상시, 13.상시, 12.상시, 11.9, 11.4, 09.1, 08.7, 08.2, 07.7, 07.1, 06.7, 05.10, 05.1, ...

099 RS-232C

- DTE와 DCE(모뎀, 음향 결합기 등) 사이의 접속 규격으로는 RS-232C가 가장 많이 이용된다.
- 25핀으로 구성된 커넥터로 전송 거리는 15m 이하이다.
- OSI 참조 모델의 물리 계층에 관계된다.

핵심 16.상시, 13.상시, 12.상시, 10.10, 07.9, 07.7, 07.1, 06.7, 04.2, 03.10, 03.7, 02.7, 02.1, 01.10, 01.4, 00.3, 99.10

100 신호 변환장치 - 모뎀(MODEM)

- 컴퓨터나 단말장치로부터 전송되는 디지털 데이터를 아날로그 회선에 적합한 아날로그 신호로 변환하는 변조(MODulation) 과정과 그 반대의 복조(DEMODulation) 과정을 수행한다.
- 디지털 데이터를 공중 전화 교환망(PSTN)과 같은 아날로그 통신망을 이용하여 전송할 때 사용된다.
- 모뎀에 포함된 등화 회로는 신호의 전송중에 발생하는 신호의 감쇠 왜곡과 전송 지연 왜곡을 방지한다.
- 원거리 전송에 주로 이용된다.



- 기능 : 변 · 복조 기능, 자동 응답 기능, 반복 호출 기능, 자동 속도 조절 기능, 모뎀 시험 기능
- 분류 기준 : 변조 방식에 의한 분류, 동기 방식(동기식, 비동기식)에 의한 분류, 포트 수에 의한 분류, 통신 속도에 의한 분류

핵심 101 DSU / 음향 결합기 / 코덱

| | |
|-------------------------------|--|
| DSU (Digital Service Unit) | <ul style="list-style-type: none"> • DSU는 컴퓨터나 단말장치로부터 전송되는 디지털 데이터를 전송 회선에 적합한 디지털 신호로 변환하는 과정과 그 반대의 과정을 수행함 • 디지털 데이터를 공중 데이터 교환망(PSDN)과 같은 디지털 통신망을 이용하여 전송할 때 사용됨 |
| 음향 결합기 (Acoustic Coupler) | <ul style="list-style-type: none"> • 음향 결합기는 단말장치와 전화기를 연결하기 위한 모뎀의 일종임 • 전화기의 송 · 수화기를 음향 결합기에 결합시켜서 디지털 신호를 아날로그 신호로 변환한 후 전송하는 장치임 |
| 코덱 (CODEC, Coder/Decoder) | <ul style="list-style-type: none"> • 코덱은 아날로그 데이터를 디지털 통신 회선에 적합한 디지털 신호로 변환(Coder)하거나 그 반대의 과정(Decoder)을 수행함 • 펄스 코드 변조(PCM) 방식을 이용하여 데이터를 변환함 |

핵심 102 다중화기

다중화기의 특징

- 하나의 통신 회선에 여러 개의 단말장치가 동시에 접속하여 사용할 수 있도록 하는 장치이다.
- 여러 단말장치가 같은 장소에 위치하는 경우, 다중화 기능을 이용하여 전송로의 수를 감소시킬 수 있다.
- 다중화기는 주파수 분할 다중화기와 시분할 다중화기로 구분할 수 있다.
- 통신 회선을 공유함으로써 전송 효율을 높이고, 통신 회선의 경비를 줄일 수 있다.
- 다중화기의 종류에는 주파수 분할 다중화기(FDM), 시분할 다중화기(TDM), 파장 분할 다중화기(WDM) 등이 있다.

주파수 분할 다중화기
(FDM; Frequency Division Multiplexer)

- 통신 회선의 주파수를 여러 개로 분할하여 여러 대의 단말기가 동시에 사용할 수 있도록 한 것임
- 주파수 편이 변조 방식을 사용함
- 전송에 필요한 대역폭보다 전송 매체의 유효 대역폭이 큰 경우에 사용함
- 시분할 다중화기에 비해 구조가 간단하고 가격이 저렴함
- 대역폭을 나누어 사용하는 각 채널들 간의 상호 간섭을 방지하기 위한 보호 대역(Guard Band)이 필요함
- 저속(1,200bps 이하)의 비동기식 전송, 멀티 포인트(Multi-Point) 방식에 적합함
- 전송된 신호는 대역 여파기(필터)를 통해 각각의 신호(주파수)로 분리됨
- 아날로그 신호 전송에 적합함

시분할 다중화기
(TDM; Time Division Multiplexer)

- 통신 회선의 대역폭을 일정한 시간 폭(Time Slot)으로 나누어 여러 대의 단말장치가 동시에 사용할 수 있도록 한 것임
- 채널당 고정된 프레임의 구성하여 전송하는 방식임
- 디지털 회선에서 주로 이용하며, 대부분의 데이터 통신에 사용됨
- 주파수 분할 다중화기에 비해 고속 전송이 가능함
- 시분할 다중화기에는 동기식 시분할 다중화기와 비동기식 시분할 다중화기가 있음

핵심 103 통신 프로토콜

- 서로 다른 기기들 간의 데이터 교환을 정확하고 원활하게 수행할 수 있도록 표준화한 통신 규약이다.
- 통신을 제어하기 위한 표준 규칙과 절차의 집합으로 하드웨어와 소프트웨어, 문서를 모두 규정한다.
- 프로토콜의 기능 : 단편화와 재결합, 캡슐화, 흐름 제어, 오류 제어, 동기화, 순서 제어, 주소 지정, 다중화, 경로 제어, 전송 서비스
- 프로토콜의 기본 요소

| | |
|---------------|--|
| 구문(Syntax) | 전송하고자 하는 데이터의 형식, 부호화, 신호 레벨 등을 규정 |
| 의미(Semantics) | 두 기기 간의 효율적이고 정확한 정보 전송을 위한 협조 사항과 오류 관리를 위한 제어 정보를 규정 |
| 시간(Timing) | 두 기기 간의 통신 속도, 메시지의 순서 제어 등을 규정 |



핵심

18.상시, 17.상시, 16.상시, 15.상시, 13.상시, 12.상시, 11.9, 10.10, 10.1, 09.9, 07.9, 07.1, 06.10, 06.10, ...

104 OSI 7계층

- 다른 시스템 간의 원활한 통신을 위해 ISO(국제표준화 기구)에서 제안한 통신 규약(Protocol)이다.
- 개방형 시스템(Open System) 간의 데이터 통신 시 필요한 장비 및 처리 방법 등을 7단계로 표준화하여 규정했다.
- OSI 7계층 : 하위 계층(물리 계층 → 데이터 링크 계층 → 네트워크 계층) → 상위 계층(전송 계층 → 세션 계층 → 표현 계층 → 응용 계층)

| | |
|----------------------------------|--|
| 물리 계층 (Physical Layer) | <ul style="list-style-type: none"> • 전송에 필요한 장치 간의 실제 접속과 절단 등 기계적, 전기적, 기능적, 절차적 특성을 정의함 • RS-232C, X.21 등의 표준이 있음 |
| 데이터 링크 계층 (Data Link Layer) | <ul style="list-style-type: none"> • 2개의 인접한 개방 시스템들 간에 신뢰성 있고 효율적인 정보 전송을 할 수 있도록 함 • 흐름 제어, 오류 제어 등을 수행함 • HDLC, ADCCP, LLC, LAPB, LAPD 등의 표준이 있음 |
| 네트워크 계층 (Network Layer, 망 계층) | <ul style="list-style-type: none"> • 개방 시스템들 간의 네트워크 연결을 관리하는 기능과 데이터의 교환 및 중계 기능을 함 • 데이터의 안전한 전송을 위해 논리적 링크(가상 회로)를 설정함 • X.25, IP 등의 표준이 있음 |
| 전송 계층 (Transport Layer) | 논리적 안정과 균일한 데이터 전송 서비스를 제공함으로써 종단 시스템(End-to-End) 간에 투명한 데이터 전송을 가능하게 함 |
| 세션 계층 (Session Layer) | 개체들 간의 관련성을 유지하고 대화 제어를 담당하는 계층임 |
| 표현 계층 (Presentation Layer) | <ul style="list-style-type: none"> • 응용 계층으로부터 받은 데이터를 세션 계층에 보내기 전에 통신에 적당한 형태로 변환하고, 세션 계층에서 받은 데이터는 응용 계층에 맞게 변환하는 기능을 함 • 데이터 암호화, 데이터 압축 등을 수행함 |
| 응용 계층 (Application Layer) | <ul style="list-style-type: none"> • 사용자(응용 프로그램)가 OSI 환경에 접근할 수 있도록 서비스를 제공함 • 전자 사서함(SMTP), 파일 전송(FTP), 원격 접속(TELNET) 등의 서비스를 제공함 |

핵심

18.상시, 17.상시, 16.상시, 15.상시, 14.상시, 13.상시, 12.상시, 11.9, ...

105 망의 구성 형태

| | |
|------------------|---|
| 성형(Star, 중앙 집중형) | <ul style="list-style-type: none"> • 중앙에 중앙 컴퓨터가 있고, 이를 중심으로 단말 장치들이 연결되는 중앙 집중식의 네트워크 구성 형태임 • 포인트 투 포인트(Point-to-Point) 방식으로 회선을 연결함 • 각 단말장치들은 중앙 컴퓨터를 통하여 데이터를 교환함 |
| 링형(Ring, 루프형) | <ul style="list-style-type: none"> • 컴퓨터와 단말장치들을 서로 이웃하는 것끼리 포인트 투 포인트(Point-to-Point) 방식으로 연결시킨 형태임 • 분산 및 집중 제어 모두 가능함 • 단말장치의 추가/제거 및 기밀 보호가 어려움 |
| 버스형(Bus) | <ul style="list-style-type: none"> • 1개의 통신 회선에 여러 대의 단말장치가 연결되어 있는 형태임 • 물리적 구조가 간단하고, 단말장치의 추가와 제거가 용이함 • 기밀 보장이 어렵고, 통신 회선의 길이에 제한이 있음 |
| 계층형(Tree, 분산형) | <ul style="list-style-type: none"> • 중앙 컴퓨터와 일정 지역의 단말장치까지는 하나의 통신 회선으로 연결시키고, 이웃하는 단말장치는 일정 지역 내에 설치된 중간 단말장치로부터 다시 연결시키는 형태임 • 분산 처리 시스템을 구성하는 방식임 |
| 망형(Mesh) | <ul style="list-style-type: none"> • 모든 지점의 컴퓨터와 단말장치를 서로 연결한 형태로, 노드의 연결성이 높음 • 많은 단말장치로부터 많은 양의 통신을 필요로 하는 경우에 유리함 • 모든 노드를 망형으로 연결하려면 노드의 수가 n 개일 때, $n(n-1)/2$개의 회선이 필요함 |

잠깐만요! 분산 처리 시스템

분산 처리 시스템이란 지역적으로 분산된 여러 대의 컴퓨터를 연결하여 작업을 분담하여 처리하는 방식으로 다음과 같은 특징이 있습니다.

- 현장 업무를 신속하고 효율적으로 처리할 수 있습니다.
- 네트워크의 확장 및 다양한 입력 방식의 채택이 용이합니다.
- 신뢰성은 높으나 보안성이 취약합니다.
- 자원의 공유가 가능하며, 장애(고장) 발생 시 장애가 발생한 노드에만 국한됩니다.



106 네트워크 관련 장비

| | |
|-----------------|--|
| 허브(Hub) | 한 사무실이나 가까운 거리의 컴퓨터들을 연결하는 장치로, 각 회선을 통합적으로 관리하며, 신호 증폭 기능을 하는 리피터의 역할도 포함함 |
| 리피터 (Repeater) | 전송되는 신호가 전송 선로의 특성 및 외부 충격 등의 요인으로 인해 원래의 형태와 다르게 왜곡되거나 약해질 경우 원래의 신호 형태로 재생하여 다시 전송하는 역할을 수행함 |
| 브리지 (Bridge) | LAN과 LAN을 연결하거나 LAN 안에서의 컴퓨터 그룹(세그먼트)을 연결하는 기능을 수행함 |
| 라우터 (Router) | 브리지와 같이 LAN과 LAN의 연결 기능에 데이터(패킷) 전송의 최적 경로를 선택할 수 있는 기능이 추가된 것으로, 서로 다른 LAN이나 LAN과 WAN의 연결도 수행함 |
| 게이트웨이 (Gateway) | <ul style="list-style-type: none"> 프로토콜 구조가 전혀 다른 네트워크의 연결을 수행함 데이터 형식 변환, 주소 변환, 프로토콜 변환 등을 수행함 |
| 랜카드 (LAN Card) | <ul style="list-style-type: none"> 컴퓨터와 네트워크를 연결하는 기능을 수행함 이더넷 카드, 네트워크 어댑터라고도 함 |

107 패킷 교환망

- 메시지를 일정한 길이의 패킷으로 잘라서 저장 및 전송하는 방식이다.
- 패킷은 장애 발생 시의 재전송을 위해 패킷 교환기에 일시 저장되었다가 곧 전송되며 전송이 끝난 후 폐기된다.
- 하나의 회선을 여러 사용자가 공유할 수 있으므로 회선 이용률이 높다.
- 수신 측에서 분할된 패킷을 재조립해야 한다.
- 응답 시간이 빠르므로 대화형 응용이 가능하다.
- 통신량의 제어를 통한 망의 안전성을 높일 수 있다.
- 전송 속도와 코드 변환이 가능하다.
- 전송 시 교환기, 회선 등에 장애가 발생하여도 다른 정상적인 경로를 선택하여 우회할 수 있다.
- 통신망에 의한 패킷의 손실이 있을 수 있다.
- 공중 데이터 망을 통한 음성(아날로그) 전송보다 공중 데이터 교환망을 통한 데이터(디지털) 전송에 더 적합하다.

108 LAN(근거리 통신망)

- 광대역 통신망과는 달리 학교, 회사, 연구소 등 한 건물이나 일정 지역 내에서 컴퓨터나 단말장치들을 고속 전송 회선으로 연결하여 프로그램 파일 또는 주변장치를 공유할 수 있도록 한 네트워크 형태이다.
- 단일 기관의 소유, 제한된 지역 내의 통신으로, 외부 통신망의 제약을 받지 않는다.
- 광대역 전송 매체의 사용으로 고속 통신이 가능하므로 방송 형태로 서비스가 가능하다.
- 공유 매체를 사용하므로 경로 선택 없이 매체에 연결된 모든 장치로 데이터를 전송할 수 있다.
- 오류 발생률이 낮으며, 네트워크에 포함된 자원을 공유할 수 있다.
- 네트워크의 확장이나 재배치가 쉽다.
- 다양한 디지털 정보 전송이 가능하다.
- 전송 매체로 꼬임선, 동축 케이블, 광섬유 케이블 등을 사용한다.
- 망의 구성 형태에 따라 성형, 버스형, 링형, 계층형(트리형)으로 분류할 수 있다.
- LAN의 계층 구조는 물리 계층과 데이터 링크 계층으로 나뉜다.

| | |
|-----------|--|
| 물리 계층 | OSI 7계층의 물리 계층과 동일한 기능을 제공 |
| 데이터 링크 계층 | <ul style="list-style-type: none"> 하위 계층인 매체 접근 제어(MAC) 계층과 상위 계층인 논리 링크 제어(LLC) 계층으로 나뉨 매체 접근 제어(MAC) 방식의 종류 : CSMA, CSMA/CD, 토큰 버스, 토큰 링 |

• LAN의 확장형

| | |
|---------------------------------|---|
| MAN (Metropolitan Area Network) | <ul style="list-style-type: none"> 도시형 통신망이라고 하며, 약 50km 반경 이내의 도시, 변화가, 대단위 아파트 단지 등을 대상으로 구성하는 통신망 단일 기관 내에서만 구성하는 LAN의 제약과 가까운 거리에 있는 시스템 간에도 호스트 컴퓨터를 일일이 거치므로 비용 낭비 및 능률 저하가 발생하는 WAN(광역 통신망)의 단점을 해소하기 위한 통신망 |
| WAN (Wide Area Network) | <ul style="list-style-type: none"> 일반적으로 제3자에 의해 제공되고 운영되는 공중망(Public Network) LAN이 여러 개 모여서 그들 간에 고속 전송이 가능한 전용 회선으로 연결된 광역 통신망 |



• IEEE 802의 주요 표준 규격

| | | | |
|-------|-------------|--------|--------------|
| 802.1 | 전체의 구성 | 802.5 | 토큰 링 방식 |
| 802.2 | 논리 링크 제어 계층 | 802.6 | 도시형 통신망(MAN) |
| 802.3 | CSMA/CD 방식 | 802.11 | 무선 LAN |
| 802.4 | 토큰 버스 방식 | 802.15 | 블루투스 표준 |

핵심 109 위성 통신망

- 지상에서 쏘아올린 마이크로 주파수를 통신 위성을 통해 변환·증폭한 후 다른 주파수로 지상에 송신하는 방식이다.
- 위성 통신에서 사용하는 주파수 대역은 3~30GHz의 극초단파(SHF; Super High Frequency)이다.
- 통신 위성은 지구 적도면 상공 약 36,000km 정도 높이의 정지궤도 상에 위치하여 지구의 자전 속도로 운행한다. 정지궤도 상에 위치하므로 정지 위성이라고도 한다.
- 대역폭이 넓어 고속·대용량 통신이 가능하고, 통신 비용이 저렴하다.
- 오류율이 적어 고품질의 정보 전송이 가능하다.
- 데이터 전송 시 반드시 통신 위성을 거쳐야 하므로, 전송 지연시간이 길다.
- 수신용 안테나만 있으면 누구나 통신 내용을 수신할 수 있는 방송망 형태이므로 보안성이 취약하다.
- 사용 주파수가 높아질수록 기상 현상(비, 눈 등)에 의한 신호의 감쇠가 심하다.

핵심 110 다중 접속 방식

| | |
|---|---|
| FDMA(주파수 분할 다중 접속, Frequency Division Multiple Access) 방식 | 주파수 대역을 일정 간격으로 나누어 여러 사용자가 각기 주어진 주파수 대역에 채널을 설정하는 방식 |
| TDMA(시분할 다중 접속, Time Division Multiple Access) 방식 | 주파수의 시간폭을 여러 개로 나누어서 각 시간폭에 대해 채널을 설정하는 방식 |
| CDMA(코드 분할 다중 접속, Code Division Multiple Access) 방식 | FDMA와 TDMA의 혼합 방식으로, 여러 사용자가 시간과 주파수를 공유하면서 서로 다른 코드를 부여한 신호를 확산하여 보내고 수신 측에서는 동일한 코드로 확산된 데이터만을 골라 원래의 신호로 재생하는 방식 |

핵심 111 인터넷

인터넷의 개요

- 인터넷(Internet)이란 TCP/IP 프로토콜을 기반으로 하여 전 세계 수많은 컴퓨터와 네트워크들이 연결된 광범위한 컴퓨터 통신망이다.
- 인터넷은 미 국방성의 ARPANET에서 시작되었다.
- 인터넷은 유닉스 운영체제를 기반으로 한다.

인터넷의 주소 체계

| | |
|-------------------------|---|
| IP 주소(IPv4) | <ul style="list-style-type: none"> • 인터넷에 연결된 모든 컴퓨터의 자원을 구분하기 위한 고유한 주소 • 숫자로 8비트씩 4부분, 총 32비트로 구성되어 있으며, 각 부분은 점(.)으로 구분함 • IP 주소는 네트워크 부분의 길이에 따라 A 클래스에서 E 클래스까지 총 5단계로 구성되어 있음 |
| IPv6 | <ul style="list-style-type: none"> • 현재 사용하고 있는 IP 주소 체계인 IPv4가 더 이상 주소를 지정할 수 없을 정도로 포화 상태에 이르게 됨에 따라 이에 대한 대책으로 128비트의 IPv6이 개발되었음 • 숫자로 16비트씩 8부분, 총 128비트로 구성되어 있으며, 각 부분을 16진수로 표현하고 콜론(:)으로 구분함 |
| 도메인 네임 | <ul style="list-style-type: none"> • 숫자로 된 IP 주소를 사람이 이해하기 쉬운 문자 형태로 표현한 것 • 호스트 컴퓨터명, 소속 기관 이름, 소속 기관의 종류, 소속 국가명 순으로 구성되며, 왼쪽에서 오른쪽으로 갈수록 상위 도메인을 의미함 |
| DNS(Domain Name System) | 도메인 네임을 컴퓨터가 이해할 수 있는 IP 주소로 변환하는 역할을 하는 시스템 |



핵심 18.상시, 15.상시, 14.상시, 09.9, 08.3, 05.7, 05.1, 02.7

112 인터넷 서비스 / 검색 엔진의 연산자

인터넷 서비스

| | |
|--|---|
| WWW(World Wide Web) | <ul style="list-style-type: none"> • 텍스트, 그림, 동영상, 음성 등 인터넷에 존재하는 다양한 정보를 거미줄처럼 연결해 놓은 종합 정보 서비스 • HTTP 프로토콜을 사용하는 하이퍼텍스트 기반으로 되어 있음 • WWW를 효과적으로 검색할 수 있도록 도와주는 프로그램을 웹 브라우저(Web Browser)라고 함 <p>※ HTTP(Hyper Text Transfer Protocol) : 하이퍼 텍스트 문서를 전송하기 위해 사용되는 프로토콜</p> |
| 전자 우편 (E-Mail) | <ul style="list-style-type: none"> • 인터넷을 통해 다른 사람과 편지뿐만 아니라 그림, 동영상 등 다양한 형식의 데이터를 주고 받을 수 있도록 해주는 서비스 • 전자 우편에 사용되는 프로토콜 <ul style="list-style-type: none"> - SMTP : 메일 전송에 사용 - POP3 : 메일 수신에 사용 - MIME : 웹 브라우저가 지원하지 않는 각종 멀티미디어 파일의 내용을 확인하고 실행시켜 주는 프로토콜 |
| FTP(File Transfer Protocol = 파일 전송 프로토콜) | 컴퓨터와 컴퓨터 또는 컴퓨터와 인터넷 사이에서 파일을 주고받을 수 있도록 하는 원격 파일 전송 프로토콜 |

검색 엔진의 연산자

- 검색 엔진을 이용하여 데이터를 검색할 때, 연산자를 이용하면 정보를 더욱 쉽게 찾을 수 있다.
- 검색 엔진 연산자의 연산 순위는 NEAR → NOT → AND → OR 순이다.
- AND, OR, NOT과 같은 연산자를 불 연산자라고 한다.

| | |
|-------------|---------------------------------|
| AND(&, 그리고) | 두 단어가 동시에 포함된 정보만을 검색 |
| OR(+, 또는) | 2개의 단어 중 1개라도 포함된 정보를 검색 |
| NOT | 지정된 단어를 포함하고 있는 정보는 제외하고 검색 |
| NEAR | 단어의 순서를 무시하고, 인접한 거리에 있는 정보를 검색 |

핵심 17.상시, 15.상시, 12.상시, 06.4, 05.10, 03.10, 00.10, 99.7

113 비디오텍스(Viedotex)

- 각종 정보를 모아 데이터베이스(DB)를 구축하고, 전화망을 통해 TV나 단말장치에 접속하여 필요한 정보를 문자나 그림의 형태로 검색할 수 있도록 하는 서비스이다.
- 서비스 내용으로는 정보 검색, 거래 처리, 메시지 전달, 예약 업무, 원격 감시 서비스 등이 있다.
- 대화형 양방향 미디어로서, 요구하는 정보를 즉시 제공할 수 있다.

핵심 15.상시, 14.상시, 12.상시, 09.7, 08.3, 09.1, 05.1, 03.7, 03.3, 03.1, 02.10, 02.1

114 기타 뉴미디어

| | |
|----------------------------|--|
| HDTV(High Definition TV) | <ul style="list-style-type: none"> • 기존의 TV 주사선을 2배 정도 늘린 1,050~1,250의 주사선을 제공하여 선명한 화상과 양질의 음성을 제공하는 TV • 위성 TV 방송, TV 회의 등에서 새로운 매체의 단말장치로 사용 |
| VOD(Video On Demand) | <ul style="list-style-type: none"> • 다양한 정보의 데이터베이스를 구축하여 사용자가 요구하는 정보를 원하는 시간에 볼 수 있도록 전송하는 뉴미디어 서비스임 • 정보 제공자의 선택에 의해 정보를 서비스하는 것이 아니라 사용자의 선택에 의해 정보를 제공함 |
| 텔레매틱스 (Telematics) | <ul style="list-style-type: none"> • Telecommunication(통신)과 Informatics(정보 과학)의 합성어로 차량, 항공, 선박 등의 운송 수단 이동 중에 정보가 제공되는 무선 데이터 서비스임 • 통신과 정보처리를 결합한 새로운 비전화계 단말 장치에 의한 통신 서비스임 |
| 텔레미터링 (Telemetry) | 통신망을 통해 통신 회선이 사용되지 않는 심야 시간을 이용하여 공중 시스템 및 시설 시스템을 점검하는 서비스 |
| 텔레텍스트 (Teletext) | <ul style="list-style-type: none"> • TV 화면과 화면 사이의 귀선 시간을 이용하여 TV 방송과 함께 문자나 도형 정보를 제공하는 문자 다중 방송 • 일기 예보, 프로그램 안내, 방송되는 프로의 세부 설명, 교통 안내 등 방송국에서 제공하는 정보를 일방적으로 수신하는 형태 |
| IPTV(Internet Protocol TV) | <ul style="list-style-type: none"> • 인터넷을 통해 방송이나 다양한 정보를 TV로 제공하는 방송 서비스 • 인터넷과 TV가 융합된 디지털 컨버전스(Digital Convergence)의 한 유형임 |



핵심 17.상시, 16.상시, 14.상시, 13.상시, 12.상시, 10.1, 08.10, 07.7, 07.4, 06.1, 05.7, 01.7

115 멀티미디어의 표준화

JPEG

- 정지 영상 압축의 국제 표준 방식이다.
- 인터넷에서 그림을 전송할 때 많이 사용된다.
- JPEG은 무손실 압축 방식과 손실 압축 방식으로 분류되며, 주로 손실 압축 방식이 사용된다.
 - 무손실 압축 방식 : 압축과 복원의 과정을 거쳐도 원래의 정보를 보존할 수 있는 방식으로, 압축률은 낮지만 영상 품질이 좋음
 - 손실 압축 방식 : 데이터에서 중복되는 내용을 제거하여 압축률을 높이는 것으로, 복원한 데이터가 압축 전의 데이터와 완전히 일치하지 않음

동영상 압축 표준

| 형식 | 특징 |
|---|--|
| AVI(Audio Visual Interleaved) | <ul style="list-style-type: none"> • Windows의 표준 동영상 파일 형식 • Windows에서 기본적으로 지원하므로 별도의 하드웨어 장치 없이 재생할 수 있음 • Windows Media Player를 이용하여 재생할 수 있음 |
| DVI(Digital Video Interface) | <ul style="list-style-type: none"> • Intel 사가 개발한 동영상 압축 기술 • 디지털 TV를 위한 압축 기술이었지만, Intel 사에 의해 멀티미디어 분야의 동영상 압축 기술로 발전되었음 |
| 퀵 타임(Quick Time MOV) | <ul style="list-style-type: none"> • Apple 사가 개발한 동영상 압축 기술 • JPEG의 압축 방식을 사용 • Windows에서 재생하려면 Quick Time for Windows를 설치해야 함 |
| MPEG(Moving Picture Experts Group) | <ul style="list-style-type: none"> • 동영상 전문가 그룹에서 제정한 동영상 압축 기술에 대한 국제 표준 규격 • 동영상뿐만 아니라 오디오도 압축할 수 있음 • 프레임 간의 연관성을 고려하여 중복 데이터를 제거함으로써 압축률을 높이는 손실 압축 기법을 사용함 • MPEG-Video, MPEG-Audio, MPEG-System으로 구성됨 • MPEG-1, MPEG-2, MPEG-4, MPEG-7, MPEG-21의 규격이 있음 |
| ASF(Advanced Streaming Format) / WMV(Windows Media Video) | <ul style="list-style-type: none"> • 인터넷을 통해 오디오, 비디오 및 생방송 수신 등을 지원하는 마이크로소프트사의 통합 멀티미디어 형식으로, 스트리밍을 위한 표준 기술 규격 • 용량이 작고, 음질이 뛰어나 주로 스트리밍 서비스를 하는 인터넷 방송국에서 사용됨 • WMV는 ASF보다 최신 버전으로, ASF와 사용하는 코덱이 다름 |

| | |
|-----------------------------|--|
| DivX(Digital Video Express) | <ul style="list-style-type: none"> • 동영상 압축 고화질 파일 형식으로, 비표준 동영상 파일 형식 • MPEG-4와 MP3를 재조합한 것으로, 이 형식의 동영상을 보려면 소프트웨어와 코덱이 필요함 • 비표준 동영상 파일 형식이기 때문에 확장자는 AVI를 사용함 |
| ram | <ul style="list-style-type: none"> • 리얼 미디어라는 비디오 스트림 방식에서 사용되는 파일 포맷 • 실시간 전송이 가능하며, 파일을 내려 받으면서 재생할 수 있음 • Real Player를 설치해야 사용할 수 있음 |

잠깐만요 ① 코덱(CODEC)

- 전송 및 보관을 위해 대용량의 동영상 및 사운드 파일을 압축(COmpress)하거나 압축을 푸는(DECompress) 데 사용되는 모든 기술, 도구 등을 통칭하는 말입니다.
- 사용하는 소프트웨어마다 코덱이 다르므로 해당 소프트웨어에 맞는 코덱을 설치해야 합니다.