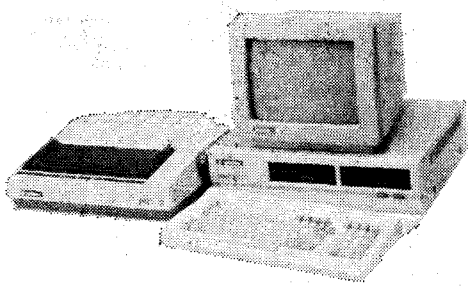


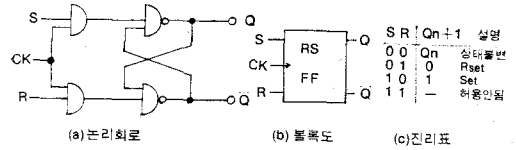
# 尖端技術 어디까지

## 컴퓨터 關聯發明의 技術

### 라. 基本的인 記憶回路

디지탈 記憶의 基本的인 電氣的 記憶裝置의 일例를 들면 〈그림 19〉와 같다.

〈그림 19〉S-R플립플롭




〈그림 19〉에서 S—R 플립플롭을 例로 들어 보았는데 이 외도 J—K, T, D플립플롭등 種類가 많다. 여기서 CK는 同期信號로서 보통 Clock이라 부른다. 이 信號가 있을 때만 플립플롭의 記憶 내용을 새로이 變化시킬 수 있다. 〈그림 19〉(C)의 眞理表에서 $Q_n$은 어떤 基準時間에서 出力信號이며 $Q_{n+1}$은 그 다음 週期에서의 出力信號이다. 이러한 플립플롭은 일단 記憶된 내용을 外部에서 變化시켜주지 않으면 항상 그대로 記憶하고 있다. 이러한 記憶裝置를 靜的 記憶裝置라고 한다(例를들어, SRAM). 이에 比하여 일단 記憶시켜 놓은 내용이라도 계속적인 記憶 維持를 위하여 再入力이 계속적으로 필요한 記憶裝置가 있는데 이를 動的 記憶裝置라 한다. (例를들어, DRAM)

電氣的 記憶裝置는 이와같은 基本的인 記憶 Cell을 集積시켜 놓은 것이라 할 수 있다. 이러한 電氣的 記憶裝置들은 電氣가 回路에 인가되어 있을 때만 記憶內容을 維持하고 電氣가 끊어지면 記憶內容이 消失된다. 이러한 記憶裝置를 휘발성 記憶裝置(volatile memory)라 하고 RAM이 이와 같은 記憶裝置이다.

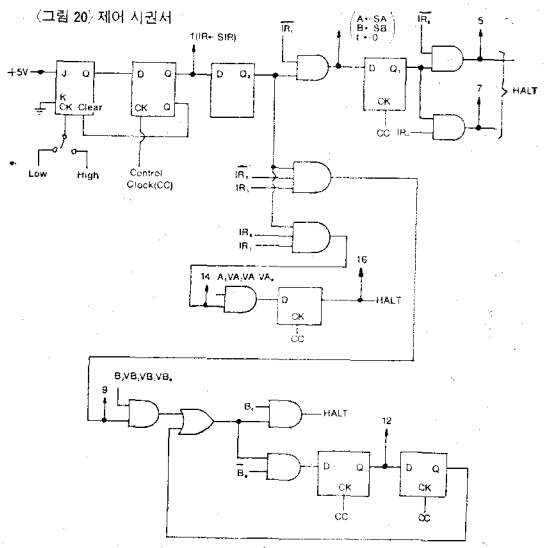
이에 對하여 電氣가 공급되지 않을 때에도 記憶內容이 消失되지 않는 記憶裝置가 있는데 이와 같은 것을 불휘발성 記憶裝置(nonvolatile memory)라고 하며 磁氣 記憶裝置(磁氣 disc, tape, 디스켈등)와 ROM등이 있다.

### 마. 制御 시퀀서(control sequencer)

컴퓨터에서 두뇌구실을 하는 判斷, 指令의 遂行등의 動作은 제어 시퀀서(control sequencer)에서 수행한다. 다시말하면 컴퓨터가 어떤 명령을 遂行하고자 할 때 그 명령 內容이 무엇인가를 判讀하고 그 判讀된 결과에 따라 記憶裝置, 演算裝置, 버스 制御裝置등을 使用하여 적절한 타이밍에 따라 명령內容의 處理를 진행시키는 裝置가 바로 制御 시퀀서이다.

〈그림 20〉은 制御 시퀀서의 일例이다.

〈그림 20〉 제어 시퀀서




### 바. Microprocessor 裝置(MPU)

中央處理裝置(CPU)가 컴퓨터 두뇌로서의 一般的인 명칭이라면 MPU는 보다 特別한 의미를 갖는다. 즉,

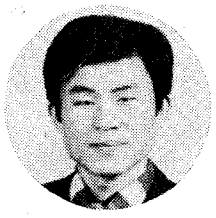
# 왔나
# 및 出願動向(2)

[redacted]
<特許廳 審査官>

目 次

1. 序 言
2. 애널로그(anolog)信號와 디지탈 信號
3. 애널로그 컴퓨터
4. **디지탈 컴퓨터**
5. 디지탈 컴퓨터 네트웍(network)
6. 디지탈 컴퓨터의 應用
7. 未來의 컴퓨터
8. 우리나라 컴퓨터 産業의 現況 및 展望
9. 國際 分類表에 의한 컴퓨터 關聯發明의 分類
10. 컴퓨터 關聯 發明의 出願 動向
11. 소프트웨어 關聯 發明의 特許性
12. 結 語
※ 參考文獻

<고딕은 이번號, 명조는 지난 및 다음號>



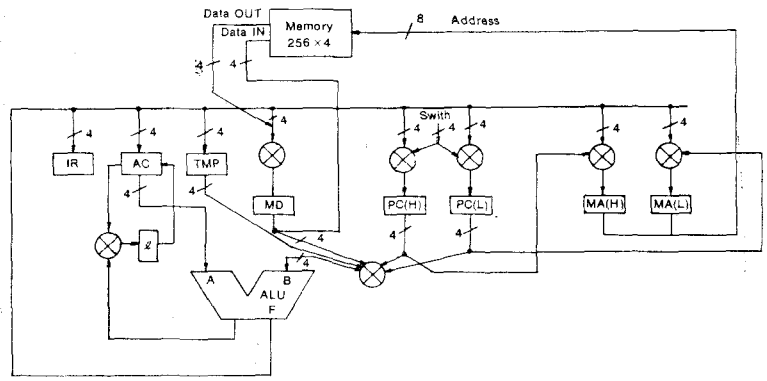
<許撥繩 審査官>

단일 Chip(또는 몇개의 Chip으로 구성된 멀티 chip)으로 構成되거나 비트數가 크지 않거나(현재까지 32bit multichip MPU가 출현해 있다) 혹은 기능이 보다 축 소된 경우가 많다. (現在에는 고도의 處理能力을 갖춘 MPU가 많이 出現되어 CPU와 MPU를 평면적으로 구분하기가 어렵고보통 같은 意味로 使用될 경우가 많다).

〈그림 21〉은 가장 基本的인 소형 컴퓨터의 CPU 또는 MPU의 블록다이어그램과 制御시퀀서를 나타낸 것이다.

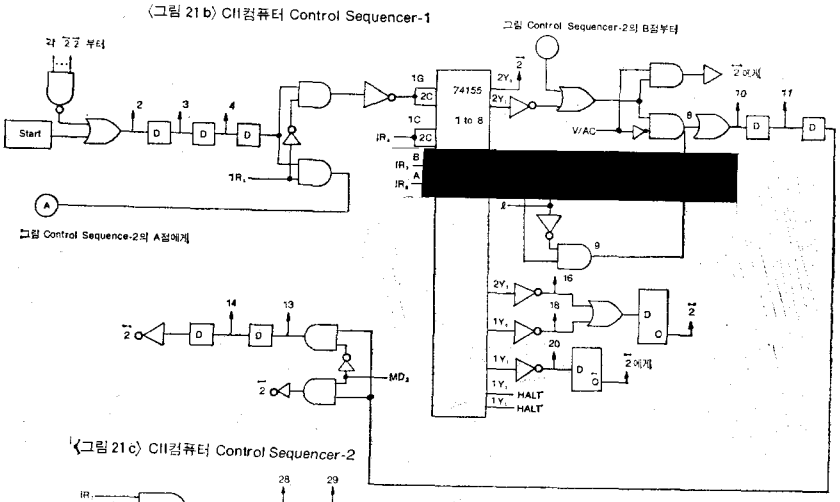
그림 22는 〈그림 21〉의 CPU의 動作과정을 表示하는 State diagram이다. 〈그림 21〉의 CPU動作은 外部에서 먼저 Start버튼을 눌러 스타트 信號를 주면 制御시퀀서는 〈그림 22〉의 State diagram의 順序에 따라 主記憶裝置(memory 256×4)로 부터 프로그램의 첫 명령(instruction)을 自動的으로 읽어서 명령 記憶裝置인 IR(Instruction Register)에 記憶시키고 IR에서 프로그램의 명령의 內容이 判別되면 이 명령의 意味에 따라 State diagram의 나머지 順序를 進行시킨다. 한 명령의 實行이 끝나면 制御시퀀서는 自動的으로 그 다음 명령을 主記憶裝置로 부터 읽어와 前述한 바와 같은

〈그림 21a. 4bit Cil Computer CPU Block Diagram

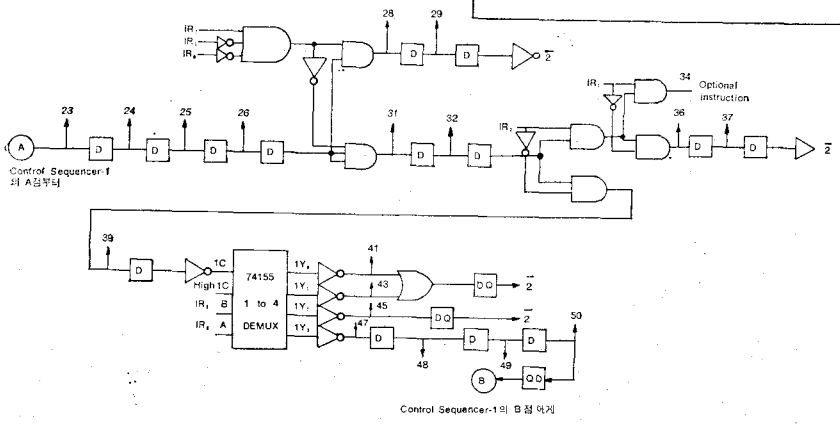


動作을 반복하고 프로그램상에 종료명령이 있던가 또는 外部로부터 종료명령이 있으면 制御시퀀서는 動作을 중지하고 State diagram의 첫번째 順序에서 정지하여 다음의 시작 명령을 기다리게 된다.

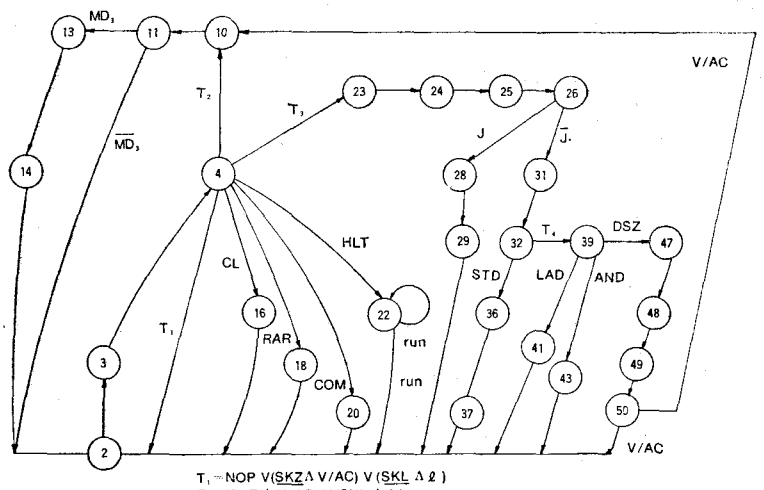
〈그림 21 b〉 CII컴퓨터 Control Sequencer-1




〈그림 21 c〉 CII컴퓨터 Control Sequencer-2




〈그림 22〉 CII Computer CPU State Diagram




$T_1$ = NOP V(SKZ Λ V/AC) V (SKL Λ ℓ )
$T_2$ = (SKZ Λ V/AC) V (SKL Λ ℓ )
$T_3$ = LAD V AND V ADP V J V STD V (Optimal Instruction 2 or 3)
$T_4$ = LAD V AND V ADD V DSZ

### 사. MPU의 應用

MPU가 凡用 컴퓨터에 使用되는 경우 CPU로 더 잘 알려져 있다. 그러나 MPU는 凡用 컴퓨터 이외에도 널리 應用되는데 例를 들면 自動販賣器, 소위 컴퓨터 TV, 냉장고, 세탁기, 선풍기 등과 그 밖에 自動制御를 目的으로 하는 곳에 널리 使用된다.

이와 같이 MPU가 모든 機器의 自動制御에 널리 使用되는 理由는 전술한 바와 같이 계획된 順序를 정확히 反復 실행해주면서 外部조건에 따른 判斷과 간단한 演算을 行할 수 있을 뿐만 아니라 프로그램의 變更에 따라 制御 動作을 간단히 변경시킬 수 있는 잇점이 있으며 價格도 비싸지 않기 때문이다.

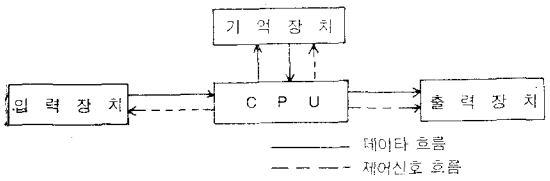
이와 같은 MPU는 보통 One-chip으로 構成되어 비용이 싸고 使用이 편리하며 처리능력의 향상을 위해 bit수를 증가 시킴에 따라 multi-chip이 출현되고 있다.

참고로 MPU의 發達과정은 그림 23과 같다.

### 아. 디지탈 컴퓨터의 基本構成

전형적인 現代의 컴퓨터 基本構成은 그림 24와 같다.

〈그림 24〉 컴퓨터의 기본 구성

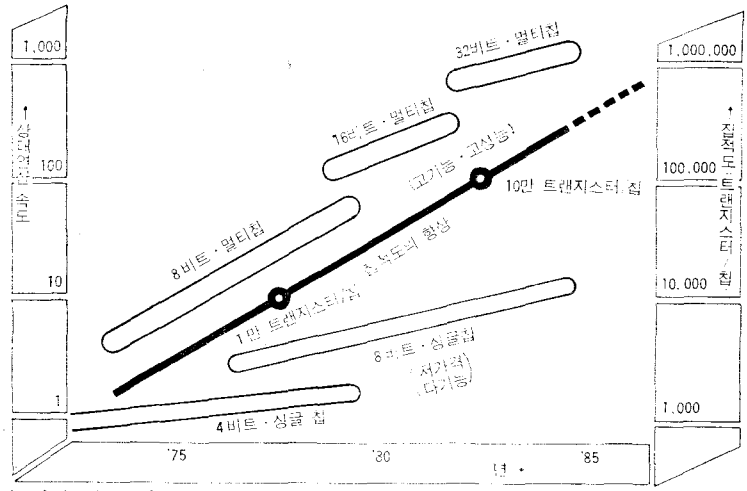



ㄱ) **中央處理裝置(CPU)**—演算과 制御技能을 갖고 간단한 記憶도 할 수 있다. 演算에는 四則演算과 論理演算이 포함되며, 制御에는 判斷과 命令기능을 포함한다. 中央處理裝置는 입력된 프로그램內容을 판독하여 그 內容을 해석하고 이에 의하여 演算과 制御技能을 수행하는 컴퓨터의 두뇌에 해당한다.

ㄴ) **入·出力製置(I/O)**—入力裝置는 中央處理裝置에 프로그램이나 데이타를 入力시키기 위한 裝置이다. 이러한 입력장치는 천공카드를 읽을 수 있는 카드 reader나, 종이 테이프를 읽을 수 있는 장치, 또는 오늘날 널리 普及되어 있는 CRT터미날의 키이보드 등이 될 수 있다. 出力裝置는 中央處理裝置에 의하여 處理된 데이타나 메모리에 記憶된 데이타의 出力을 기록하기 위한 裝置로서 프린터나 CRT터미날의 모니터가 될 수 있다.

ㄷ) **記憶裝置**—記憶裝置에는 主기억장치와 補助기억장치로 나눌 수 있다. 主기억장치는 半導體 메모리 裝置인 RAM으로 거의 構成되며 이 메모리의 特徵은 고속 access가 可能하다는 것이다. 따라서 주로 CPU와 밀접한 관계를 갖는다. 補助기억장치는 否휘발성 메모리로서 磁氣코어, 磁氣디스크 磁氣테이프, 磁氣디스켈 등으로 값싸고 대용량을 記憶할 수 있는 메모리이다. 主기억장치의 補助로 使用된다.

〈그림 23〉 마이크로 프로세서의 진보




### 2) 發達過程

1642年 프랑스의 數學者 Pascal이 인류 최초로 數의 加減이 가능한 計算裝置를 고안한 이래 1830年英國의 數學者 Charles Babbage에 의해 천공카드를 利用

한 現代的 의미의 機械式계산기 "Analytical Engine"이 만들어 졌다. Babbage가 고안한 이 機械式 계산기는 記憶(memory), 制御(control), 演算(arithmatic)및 入·出力(I/O)의 모든 條件을 갖춘 것으로 오늘날 컴퓨터의 基本개념과 동일한 것이다.

産業혁명 이후 社會構造가 복잡해짐에 따라 관리해야 할 業務나 計算할 일이 비약적으로 增加함에 따라 대규모의 데이타를 高速으로 處理해줄 컴퓨터의 出現은 절실히 요망되었고, 이에 따라 1944年 Harvard大學의 Howard Aiken은 IBM社와 함께 3000개 이상의 릴레이와 여러개의 Gear를 使用한 Automatic Sequence Controlled Calculator인 "Mark-I"을 완성하였다. 그후 電子 技術의 비약적인 發達에 힘입어 1946年에 眞空管을 利用한 "ENIAC"이 開發되었고, 1951年에 개발된 "UNIVAC-1"에 이르러 商用化되기 시작하였다.

이와 같은 技術진보의 過程은 표 4—1과 같다.

### 3) 컴퓨터의 種類

컴퓨터는 그 규모와 성능 그리고 構成에 따라 마이크로 컴퓨터, 미니 컴퓨터, 수퍼 컴퓨터, 대형 컴퓨터 등으로 分類 되기도 한다.

가. 마이크로 컴퓨터는 보통 LSI로 된 One-Chip (혹은 Multi-Chip)의 마이크로 프로세서 장치(MPU)를 CPU로 使用하는데 1971年 Intel社에서 4bit짜리 MPU를 開發한 이래 오늘날에는 32bit까지 MPU가 개발되어 마이크로 컴퓨터와 미니 컴퓨터間의 구분이 事實上 무의미 하게 되어 가고있다.

나. 미니 컴퓨터 보통 32bit 이상의 중형 컴퓨터를 말하며 CPU의 技能이 마이크로 컴퓨터의 MPU와 달리 고정되어 있지않고 變更될 수 있다. CPU의 構成은 보통 단일 chip이 아니고 bit-slice라 불리우는 여러개의 반도체 IC의 집합으로 되어 있다.

다. 대형 컴퓨터는 보통 64bit이상의 컴퓨터로서 高速 大容量의 데이터 처리가 可能한 컴퓨터를 일컷는데 정확한 기준을 定하기는 어렵다.

컴퓨터의 성능은 一般的으로 데이터를 處理하는 速度로서 평가 될 수 있는데 그림 25는 이와 같은 觀點

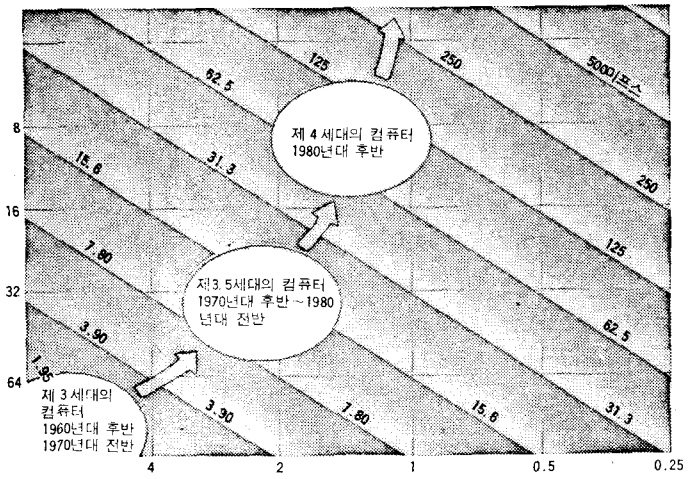
〈표 4—1〉 전자계산기의 세대별 발달과정

| 세 대 별 | 성 능 | 논 리 소 자 | 평균처리속도 |
|---|---|---|---|
| ① 제1세대 (1952~1957) | • 1952년 진공관 시대부터 업무에 까지 응용처리하기 시작한 실용화 시대의 COMPUTER가 됨.<br>• 집계·통계자료 및 일반사무중 간단한 업무만을 적용. (기계적인 속도) | 진 공 관 | 1/1000(Sec) (MILLE SECOND(MS)) |
| ② 제2세대 (1958~1964) | • TRANSISTOR의 개발로 COMPUTER의 성능향상과 신뢰도가 진공관 시대보다 100배이상 향상됨.<br>• MACNETIC CORE(기억소자)를 사용하기 시작.<br>• SYSTEM언어의 개발로 기계어에서 번역어로 변천함.<br>• DATA 통신 SYSTEM의 개발로 SERVICE업무가 개선됨.<br>• 적용; 생산관리, 원가관리, 회계관리. | 트랜지스터 | 1/100 (SEC) (MICRO SECOND [$\mu$S]) |
| ③ 제3세대 (1965~1970) | • IC(Inte grated Circuit)<br>• IBM社에서 개발<br>• IC회로의 크기는 1mm²에 TRANSISTOR 약 20개분과 콘덴서 약 40개분의 역할을 하는 회로로 구성.<br>• 컴퓨터는 소형화, 경량화, 고속화, 고신뢰성화 됨.<br>• ONE-MACHINE CONCEPT방식(전자적인 속도) | IC | 1/110억 (SEC) (NANO SECOND[NS] |
| ④ 제4세대 (1971~ ) | • LSI(Large Scale Integration)<br>• IC회로를 더욱 집약시켜 소형화함.<br>• I/O장치의 개선이 문제화 됨.<br>• MICRO COMPUTER, MINI COMPUTER 가 등장 | LSI (고밀도집적회로) | 1/1조(SEC) (PICO SECOND [PS]) |

에서 컴퓨터 성능 추이의 변천을 나타낸 것이다.

여기서 MIPS는 Million Instruction Per Second의 略字로서 1초간에 實行 가능한 命令의 數를 백만개 단위로 표시한 것이다.

〈그림 25〉 컴퓨터 성능의 추이



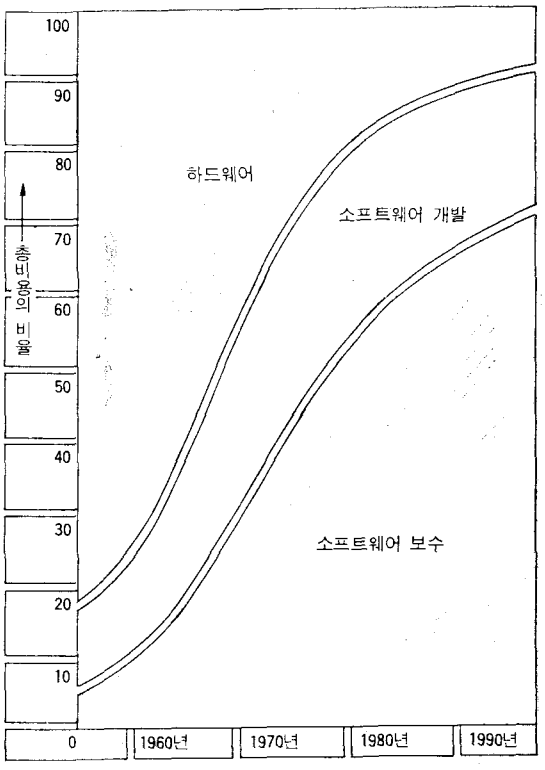

## 4) 소프트웨어

일반적으로 컴퓨터에서는 裝置인 하드웨어와 이를 運用하는 技術인 소프트웨어로 구분할 수 있다. 컴퓨터의 성능은 결국 하드웨어의 機械的 性能 그 자체와 이것의 利用技術인 소프트웨어에 의하여 決定된다. 따라서 일단 컴퓨터를 導入하고 난 후 그 효율적인 利用을 위해 소프트웨어 重要性은 더욱 커진다.

하드웨어는 오늘날 눈부신 技術進步에 힘입어 그 價格이 점점 떨어지고 있으나, 소프트웨어의 開發은 전적으로 高級 두뇌에만 의존하므로 컴퓨터의 普及이 확장되고 社會의 情報化 추세가 深化됨에 따라 그 價格이 점점 增加되어 가고 있는 실정이다.

그림 26은 전체시스템 開發 價格중에 소프트웨어가 점유하는 비율을 나타낸 것이다. 그림에서 알 수 있듯이 1980年에 이미 총 開發비용의 80%가 소프트웨어 費用이며 1990年에는 90%에 달할 것으로 보인다. 또 소프트웨어 開發費用중 초기개발에 使用되는 費用은 1/3이고 나머지 2/3는 소프트웨어 補修(확장개발 포함)에 소요됨을 알 수 있다. 〈계속〉

〈그림 26〉 시스템 개발비용 속에서 소프트 웨어가 점유하는 비율




發明하는 國民되어
先進祖國 앞당기자