

特 輯

韓國電子計算所の機能과 計算機

Korea Computer Center

梁 學 奎

I. 韓國電子計算所の紹介

1. 財団法人 韓國電子計算所の 設立趣旨

電子計算組織의 出現이 第2次 産業革命의 契機가 되어 人間의 精神勞動을 機械로 代替하는 時代가 到來하였다.

先進諸國에 있어서의 過去 10年間の 電子計算組織導入 및 活用實績에 依하던 10年前에 對比하여 40倍 乃至 千倍의 急成長을 하고 있어서 이로써 電子計算組織의 効率性和 必要性을 認定치 않을 수 없게 되었다.

또한 電子計算組織의 活用度 增加에 따라 現代經濟社會의 發展相도 이에 比例할 뿐더러 스히려 電子計算組織의 活用이 經濟 社會 軍事等 各部門의 開發과 發展을 先導하고 있는 實情에 있다.

現下 國家 經濟體制下에 있는 우리나라는 先進諸國이 電子計算組織을 動員하여 國際競爭에 臨하고 있음에 비추어 이의 活用을 통한 競爭力 強化가 切實히 要望된다.

또한 科學技術의 研究 開發에 있어서도 後進性을 脱皮하기 爲하여도 電子計算組織의 役割을 最大限度으로 利用한 時機에 마침내 到達하였다고 본다.

그러나 現下 韓國의 電子計算組織活用に 있어서 要員의 不足, 業務量에 比한 過剩設置의 傾向, 電子計算組織活用に 對한 抵抗等 許多한 問題點이 가토 늘어있다. 따라서 이와같은 問題點이 없스런 時點을 爲하여 電子計算組織의 活用은 時急하고 重大한 意義를 가졌다고 할수있다.

바로 이와같은 國家의 必要性和 意義를 俱現시키기 爲하여 韓國電子計算所를 發足한 것이니 二

役割과 使命을 다할 것을 期待한다.

2. 韓國電子計算所の 略歷

韓國生產總本部가 電子計算組織을 導入設置하여 韓國電子計算所가 發足하기 까지의 經過를 보면 아래와 같다.

1963年4月 KPC에 電子計算組織을 導入, 設置할 方針을 세움

1964年 10月 KPC와 日本 富士通(株)間에 FACOM-222의 貸貸借를 合議함

1965年 6月 KPC와 富士通(株)間에 FACOM-222 貸貸借契約을 締結함

1965年 9月 商工部長官과 經濟企劃院長官으로부터 電子計算組織導入의 推薦을 받음

1965年 11月 財務部長官으로부터 電子計算組織貸貸를 爲한 非居住者와의 外貨債權債務發生許可를 得함

1965年 9月 電子計算組織導入設置에 對한 技術의 問題로 富士通(株)의 貿易部次長 吉田嘉一氏 來韓함

1966年 10月 KPC 本部職員 3名을 日本 富士通(株)에 研修生으로 派遣함

1967年 1月 KPC에 電子計算所를 創設, 初代所長으로 李汝用氏가 就任함

1967年 3月 FACOM-222 電子計算組織이 仁川港에 到着함

1967年 5月 FACOM-222를 仁川에서 KPC에 運搬, 設置作業에 着手함

1967年 8月 FACOM-222 電子計算組織의 設置作業을 完了하고 試運轉을 開始함

1967年 10月 科學技術廳長官으로부터 民法 第32條에 依한 非營利 財團法人으로 韓國電子計算

所가 發足한

3. 韓國電子計算所의 人的構成

電子計算組織이 움직이고 있는 場面을 보고 있으면 意外로 사람이 적다는 것을 發見할 것이다. 그러나 이것은 큰 誤解이다. 왜냐 하면 눈에 보이지는 않지만 일하고 있는 사람이 많다는 것이다. 電子計算組織에 從事하는 人的構成은 管理者 System Analyst 혹은 Planner Programmer, Customer Engineer, Operator, Checker, Tape Librarian, Key Puncher, 一般事務職員 및 補助員 등이 있다.

그러나 아직도 發芽期에 있는 現韓國의 實情으로서의 위의 職務를 嚴密히 區分하지 못하고 있다. 當計算所의 人的構成으로 봐서는 理事長을 위시하여 計算所長 1人, 所長의 職務를 代理補佐하는 局長 1人과 System Analyst 겸 Programmer 18名, Customer Engineer 2名, Operator 1名, Tape Librarian 1名, Key Puncher 7名, 事務職員 3名이 있고 日本人技士로서 System Analyst 겸 Programmer가 1名 Customer Engineer가 1名이 있다. 이 중에서 特記할 것은 女性 Programmer 2名을 確保하고 있다는 事實이다.

II. 電子計算組織 (Electronic Data Processing System)

1. 概說

電子計算組織에는 電子的, 磁氣的, 機械的 構造를 갖고있는 5大機能을 有機的으로 連動시킴으로써 Data의 判讀, 記憶, 演算, 制御 및 印刷 (表 1)

機 構 名	主 要 機 械 名	機 能 的 概 要
入 力 裝 置	카드讀取機 (Card Reader) 紙帶一讀取機 (Paper Tape Reader) 光學的文字讀取機 (O. C. R)	原始 Data, 演算指示의 讀取
演 算 裝 置	中央處理裝置 (電算機本體: C. P. U.)	各種의 演算 (四則演算 및 論理演算)
記憶裝置 主記憶장치 補助記憶장치	磁氣 Drum, 磁氣 Disk, 磁氣 Tape 裝置	原始 Data, 中間演算結果의 記憶
制 御 裝 置	中央處理裝置 (電算機本體: C. P. U.) 操作卓 (Console)	讀取, 演算, 記憶, 記錄等 諸機能을 統一的으로 制御
出 力 裝 置	라인프린터 (Line Printer)	演算結果의 記錄, 印刷

의 諸作業을 自動的으로 高速度로서 할수있는 一連의 機械組織이다. 電子計算組織의 構成上의 機能을 보면 다음과 같다.

이 중에서 特히 制御, 演算, 記憶의 各機構는 電氣回路가 傳하는 Pulse에 依해서 操作되기 때문에 특히 高速으로 (1秒間에 백만回以上) 作動되고 또 磁心을 格子狀으로 配列한 素子板으로 이루어진 主記憶裝置는 좁은 空間內에서 大容量의 Data를 蓄積할수 있게 되어 있다. 電子計算組織은 이와같은 自動制御, 高速演算 및 大量記憶의 機能을 驅使함에 따라서 人力 手作業으로서는 相當한 工數를 要하는 科學技術計算이나 大量 Data의 分類集計 또는 選擇, 判定, 反復하는 過程을 包含하는 複雜한 事務處理等을 自動的 高速으로 正確하게 實行하게 된다.

이점은 一般 單能計算機나 事務機械에서 볼수 없는 機能이고 電子計算組織이 「人工頭腦」 또는 「電子頭腦」라 일컬어 지는 所以이기도 하다.

그러나 電子計算組織은 人間과 같이 스스로 生覺이나 判斷을 하지 못한다.

이와같은 것을 시키기 爲해서는 人間이 計算作業順序를 一定한 規則에 依해서 電子計算組織에게 仔細하고 正確하게 指示하여야 한다. 이것을 Program이라 한다. (Programming에 對해서는 追後 다시 論하겠다) 電子計算組織의 作業은 이와같은 一連의 Program의 指示에 依해서 行하여진다. EDPS는 Program에 依해서 指示된 計算手續을 忠實히 그리고 高速으로 遂行하지마는 그 以上の 것 (Program에 指示된 以外的의 것)은 處理하지 못한다. 더우기 Program이 잘못되었으면 結果는 잘못된 그대로 나온다.

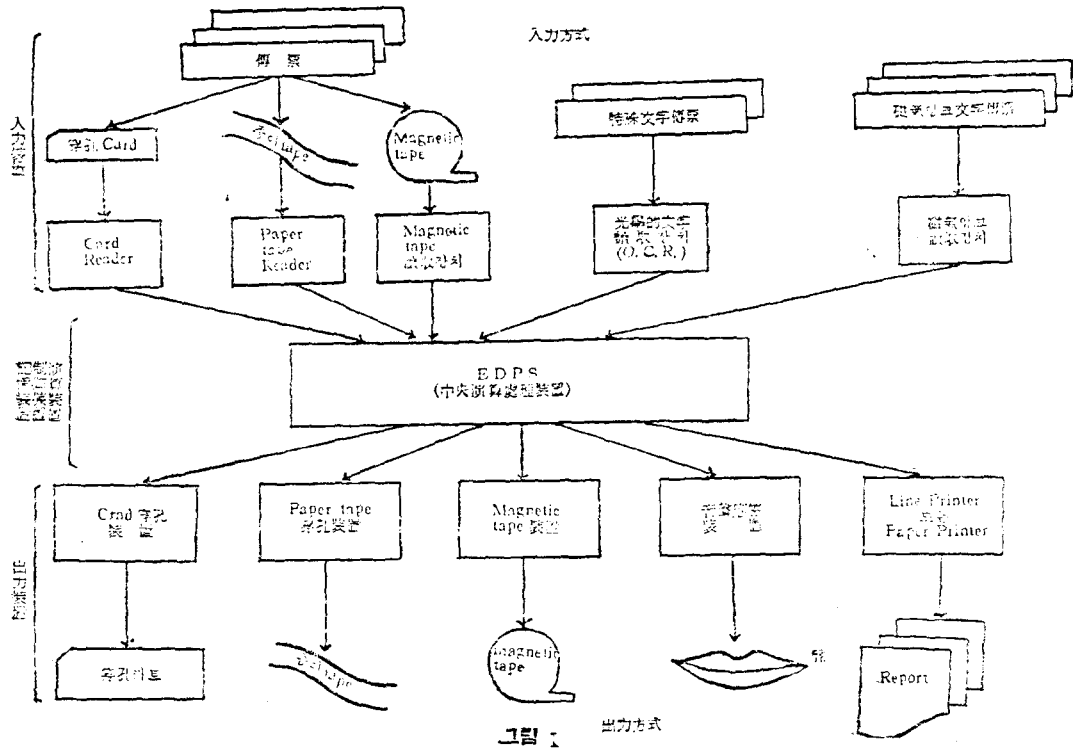


그림 : 出力方式

이런점에서 볼때 電算機의 記憶, 또는 計算手續이 人間の 思考過程과는 本質的으로 다르다는 것이다. 電算機가 記憶하는 情報은 모든 符號의 組合에 依해서 이룩되고 이에따라 行해지는 計算은 數學的演算과 論理的演算(比較, 分類, 選擇, 結合)에 限한다. 即 情報에 스스로 意味를 붙인다거나 直觀的洞察 또는 論理的 飛躍等은 一切 不可能하다. 따라서 電算機에 作業을 시키기 爲해서는 人間の 言語를 電算機의 言語(Program)로 번역 하기 前에 事務處理의 흐름과 計算方法을 여러가지 單純한 論理過程으로 分解해서 그것을 가지고 論理의 一貫性을 가지게끔 再構成하지 않으면 안된다.

아울러 이와같이 作成된 Program이 과연 電算機에 依해서 正確히 處理되는가 알리는가를 檢査(Debugging)해 보는것도 必要하다. 參考로 EDPS를 그림으로 보면 그림 1과 같다.

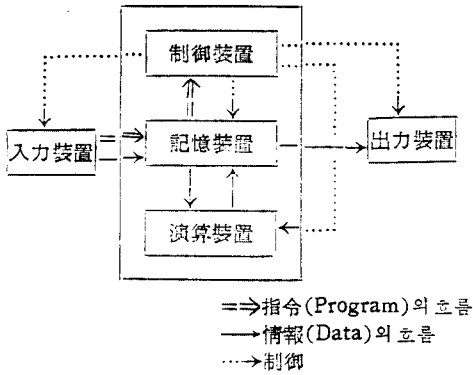
2. 構成 및 機能

前章에서 대충 말씀드렸거니와 EDPS는 構成面에서 크게 다섯가지로 나눈다. 即, 入力裝置(Input Unit), 記憶裝置(Memory 또는 Storage Unit), 制御裝置(Control Unit), 演算 및 論理判

斷裝置(Arithmetic & Logical Unit), 出力裝置(Output Unit)로 構成되어 있다. 이들 裝置들간의 關係는 어떤 型式의 機種이고 간에 各種의 情報(Data)는 入力裝置를 거쳐서 記憶裝置에 貯藏된다. 情報中에서 Program은 記憶裝置로부터 順次的으로 制御裝置에 送付되어 解讀되고 다시 指令이 되어서 4個의 裝置를 制御한다. Data는 制御裝置로부터 指令에 依해서 記憶裝置에서 演算裝置로 送付된다. 이중에서 處理된 Data는 出力裝置에 依해서 Print되거나 또는 카드나 tape에 穿孔된다. 電子計算機의 改良, 發展過程을 살펴 보면 보다 큰 大量記憶裝置, 보다 빠른 處理機能을 開發했고 入出力附屬裝置等에 對해서도 현저한 進歩가 期한다.

即 磁氣 tape裝置의 讀取, Output時間의 스피드화 및 大容量化로부터 다시 入出力裝置의 高速化等이 그 例이다. 特히 현저한 業績은 大容量으로서 速度가 빠른 比較的 값이상 Random Access의 記憶裝置, 同一型의 標準活字일것 같은 活字 그대로 읽는 讀取機等이다. 그와 同時에 主要한 것은 Data의 收集裝置이다. 이것의 發展은 今後 漸次 計算機의 利用促進에 寄與할 바가 클 것이다.

(表2) EDPS의 基本的인 構造



各構成上의 機能을 살펴보면 表 3, 4, 5와 같다.

(表3) 入出力裝置

裝置	裝置名	速度	記錄密度
入力裝置 (Input)	Card Reader	200~1,600 枚/分	80, 90字/枚
	Paper Reader	200~1,000 字/초	10字/25mm
	Magnetic tape	10,000~150,000字/초	100~1,000 字/25mm
出力裝置 (Output)	Card穿孔機 穿孔機	100~500枚/分 10~200字/초	80, 90字/枚
	電動 Typewriter	10字/초	10字/25mm
	Line Printer	200~1,500 行/分	120, 136字/行
	Magnetic tape	10,000~150,000字/초	100~1,000字/25mm

(表4) 記憶裝置

裝置名	크기	容量(桁)	平均 Access time
主記憶裝置 (磁氣Core memory)	小形	2,000~20,000	0.2~10 μ s
	中形	20,000~100,000	
	大形	100,000~1,000,000	
補助記憶裝置 (磁氣Drum/Disk/tape)	磁氣Drum	24,000,000(1臺)	35~375ms
	磁氣Disk	12,000,000(1臺)	70~850ms
	磁氣tape	5,000,000~20,000,000(1卷)	

(表5) 演算速度

(時間은 大略值임)

演算種類	計算機 A	計算機 B	計算機 C
加 減 算	8 μ S	24 μ S	44 μ S
乘 算	24 μ S	229 μ S	440 μ S
除 算	42 μ S	550 μ S	1090 μ S

資料: 電子計算機 1967年

EDPS의 機能中에서 特別 重要한 面을 차지하는 記憶裝置에 對해서 말해보기로 한다.

Input된 Data나 Output된 Data, 計算에 必要한 定數, 計算途中의 變數 또는 諸命令等等을 記憶하는데는 두가지의 區別方式이 使用되고 있다. 即 語長固定方式(Fixed Word Length)과 語長可變方式(Variable Word Length)이다. 前者는 最初부터 數桁이 一單位(Word單位)로 되어있고 一般的으로 4000~10000word의 機械가 많고 後者는 記憶裝置의 가운데를 使用者의 目的에 應해서 自由로 區別해서 使用되는 것으로 桁單位의 것과 Bit 單位의 것이 있다. 容量은 數千桁(또는 Bit)에서 數十萬桁(또는 Bit)의 것이 있다 EDPS의 크기를 말할때는 通常 이 記憶裝置의 容量을 두고 區分한다. 또 記憶裝置는 人間的 記憶과는 다르지만 일단 記憶한 數字, 文字, 命令은 二 記憶한 同一場所(番地라고함 番地는 可變方式의 경우 1桁씩, 固定方式의 경우는 5~13桁을 單位로 해서 한 番地가 붙여짐)에 다른 것을 記憶하지 않는限 매버리지는 않는다. 勿論 그 場所의 內容을 몇번이고 Read해도 없어지는 일이 없다.

人間的 腦細胞는 約 140億 Bit로 되어 있다고 하나 앞으로 곧 開發될것으로 알려진 補助記憶裝置中에는 350億 Bit의 것이 있다고 하니 人間的 記憶能力을 數倍를 증가하게 되었다.

主記憶裝置의 構成은 Magnetic Core Matrix 素子로 되어있으며 Core는 直徑 1~2mm의 큰 Ring狀을 한 Ferrite로 되어있고 電流의 흐름에 따라 오른쪽으로 磁化되었느냐 왼쪽으로 磁化되었느냐에 依해서 2進法의 1 또는 0으로 成立된다. 그러던 다시 앞으로 되돌아 가서 EDPS의 發展過程에 對해서 考察해보자

3. EDPS의 歷史

英國의 C. Babbage(1791~1871)는 一次函數와 函數表를 自動的으로 計算할 수 있는 機械를 考案하였으나 製作에 失敗하였다. 그러나 研究를 거듭하여 더욱 複雜한 計算이 可能한 現代의 Digital電子計算機와 基本原理가 비슷한「Analytical Engine」을 製作하였다. 그後 美國 IBM

(International Business Machine Corp.)에서穿孔 Card用事務機械가 發展하던중 Relay에 依한 電話交換技術의 急速한 發達로 近代의 Digital型 計算機發達の 始初가 되었다.

1940年 S. B. William이 約 400個의 Relay와 10個의 Cross Bar Switch를 使用하여 「Model I」이라는 電子計算機를 完成하게 되었다.

1944年 IBM에서는 Harvard大學 H. H. Aiken 教授의 協力を 얻어 「Mark I」을 製造하여 實際로 電子計算機製作이 始作되었다. 「Mark I」은 3000個의 Relay와 4馬力모타 72個의 Addition Counter 60個의 Constant Register와 Arithmetic Control Tape로 構成되었다. 이때 同時에 IBM에서는 작은 Calculator 即 Pluggable Sequence Relay Calculator를 만들었다.

Bell電話研究所에서는 1950年度에 「Medel VI」을, 1952年度에는 Harvard 大學協助로 Relay 代身 Tube (電子管)를 利用하여 「Mark II」과 「Mark VI」를 製作하였던 것이다.

4. EDPS의 發展

1946年 Pensilvania 大學의 Moore School에서 電子管을 使用한 計算機를 完成한것이 바로 ENIAC (Electronic Numerical Integrater and Calculator)인데 이것이 現代의 EDPS의 호시라 할수 있다.

ENIAC의 記憶裝置回路는 Tube Flip-Flop를 使用하고 이에 使用된 總 tube數는 18,800個, 所要電力 120kw의 巨大한 것이었다. 10桁의 2數字 덧셈을 1秒間에 5400回, 곱셈은 300回 計算可能하며 그때까지 發明된 Relay式 計算機보다도 100餘倍의 計算速度를 내어 當時로서는 劃期的인 것이었다.

ENIAC을 만든 T. P. Eckert와 J. W. Mauchly가 會社를 設立하여 만든것이 BINAC(Binary Automatic Computer)이며 그後 Remington Rand Corp와 合併하여 UNIVAC(Universal Automatic Computer)을 1951年度에 美國 統計局 때문에 만들게 되었다. 한편 英國 Cambridge 大學의 M. V. Wilker가 1949年 EDSAC을, Manchester 大學의 W. Kilburn 教授가 Brown管을

利用한 高速記憶裝置를 利用한 電子計算機를 Ferrati會社가 달아서 製作하였다.

以後 계속 알을 다루어 研究改良하여 Tube에서 固體回路(Transister, Diode)로, 다시 最近에 와서 集積回路(Integrated Circuit)의 發明과 더불어 이를 使用한 電子計算機를 製作하여 各分野에서 活用되고 있다.

지금까지의 發展過程을 通해서 보면 大體로 Tube를 利用한 第一世代, Transistor를 利用한 第二世代로 區分하고 1964年4月 IBM社에서 「IBM 360」의 發表를 契機로 電子計算機産業에 있어서는 超小形集積回路나 其他 超小形部品를 使用하기 始作한 바로호로 第三世代로 突入하였던 것이다. 미국이나 RCA社의 「Spectra 70」같은 高性能, 新機種이란 點에서 世界電子計算機産業界에는 急激한 變化를 가져왔다.

EDPS의 發展에 따른 世代別 差異를 보면 表6과 같다.

(表6)

項目 \ 世代	1 世代	2 世代	3 世代	未 來
年 代	1944~1958	1958~1964	1964~?	
速度單位	(ms) milli Sec.	(us) micro Sec.	(ns) nano Sec.	(uns) pico Sec.
機內要素	Tube	Transistor Diode	Integrated Circuit	?
獨立性	매우적음	적 음	많 음	매우많음
重要度	Hardware	Software Hardware	Software Application Hardware	?
信賴性	적 음	조금많음	많 음	매우많음
用 途	科學計算	科學計算 Data Processing	Real Time Time Sharing	?
數 用	적 음		차차 많아짐	

參考: 1. 第3世代 技術의特徵

- (1) 超小形集積回路의使用
- (2) 信賴性의 向上, 高速化, 記憶裝置의 大容量化
- (3) Family의 形成
- (4) 새로운 Program管器의 確立

2. 電子計算機 System의 Cost配分의 推移(世代別)

(表7)

裝 置 \ 年 代	1954	1963	1972
中央處理裝置	80%	60%	25%

入出力裝置	10%	15%	25%
Data File	10%	15%	20%
Image File	○	○	15%
傳 送 系	○	○	15%

資料：日本電子工業協會電子計算機政策部會

Ⅲ. Hardware와 Software

앞서 說明한 EDPS의 構成要素와 機能을 總稱해서 Hardware라하고 이들의 構成要素는 人間이 주는 命令에 依해서만 各機能을 發揮할수 있다. 이 命令(Program)에도 여러가지 있는데 이들을 모두 말해서 Software라하며 電子計算組識은 이들 兩者가 完備해야만 完全한 것이된다.

Software의 種類와 範圍에 對해서는 여러가지 解析에 있어서 사람에 따라 다르나 크게 나누어 다음의 세가지로 나눈다. 即

- ① Mathematical Program
- ② System Program
- ③ Application Program

이 셋중에서 특히 System Program와 Application Program에 對해서 말해 보기로 한다.

1. System Program

EDPS는 漸次 高速, 大容量, 多岐入出力裝置, 多重處理等 Hardware의 機能이 增大해 가고 있다. 即 Card Reader가 생기면 Card Reading Program이, Magnetic Tape가 생기면 Magnetic Tape用 Program이 多重處理 EDPS이던 이의 活用을 爲한 Program이 必要하게 되는데 이와같이 Hardware의 機能을 더 有效하게 活用할수 있게 하는것이 System Program이다. 最近에는 많은 自動 Program이 開發되었는데 Autocoder, CCEOL(Common Business Oriented Language), ALGOL(Algorithmic Language), FORTRAN(Formular Translator), FASP, MACRO, PL-1, MAD 등이 그것이다. 또 EDPS가 高速化하게 되면 필수록 1일에 하는 種類가 많아지고 일의 種類가 바뀔때마다 磁氣 tape의 變更 switch의 set 등의 作業數도 많아진다. 그래서 그때마

다 EDPS를 停止시키는 時間을 減수있는데로 節約하기 爲해서 몇個人가의 相異한 일을 하나의 일과 같이 編集해서 最初 1回 Start하던 그다음으로 自動적으로 運轉해가기 爲한 Program 即 Monitor System이라는 것도 大型機가 되던 必要하게 된다.

2. Application Program

最近에는 새로운 應用을 爲한 專門問題用的 Program이 相當히 開發되었다. 今後로는 더욱 이러한 경향으로 기울어 질것이 豫想된다.

代表的인 것으로는 LP (Linear Program), PERT(Program Evaluation and Review Technique), SIP(Symbolic Input Program), CPM (Critical Path Method), GPSS, DYNAMO 등이 있다.

이와같이 넓은 意味를 가진 Software를 一括해 보면 다음과 같다.

Mathematical Program: ① 技術計算用 Sub-Routine으로 線型代數, 代數方程式, 數值微積分, 微積分方程式, 函數近似 等

② 統計 OR(Operation Research) 用 Sub-Routine으로 Regression Analysis, 分散分析, 要因分析, 相關係數, 標準偏差, 亂數發生等

③ 特殊算術演算으로 浮動小數點演算, 二倍精度演算, 複素數演算等

System Program: ① Input Output Program 으로 Card入出力用 Program, Print用 Program, Magnetic Tape用 Program等

② Data處理 System으로 Sorting Program, Merging Program, File Maintenance Program等

③ 自動Programming System으로 Assembler Compiler Generator等

④ Monitor System

⑤ Check System으로 Memory Print, Check Point等

Application Program: ① Linear Programming System ② Simulation System, ③ 統計處理 System, ④ 經營管理 System으로 日程計劃, 生産管理, 在庫管理, 需要豫測等

- ⑤ 工學用 System으로 土木, 建設, 化學工業, 原子力, 機械工學, 光學等
- ⑥ Information Retrieval System 등으로 區分해 볼수 있다.

IV. Programming

1. Prpgramming이란 ?

EDPS를 使用하여 Data를 處理할때 機械가 取할 動作을 定해 주는것이 프로그래밍이다. 따라서 Program이 잘 못되었으면 그릇된 結果가 나오거나 機械가 停止하게 되거나 한다. Program은 사람이 作成하는데 사람이 만든 Program에 따라서 機械의 能力이 充分히 發揮되느냐 안되느냐가 決定된다. Program의 作成段階는

- ① 問題 또는 作業의 分析
- ② Flow Chart의 作成
- ③ Coding
- ④ Debugging의 順序이다

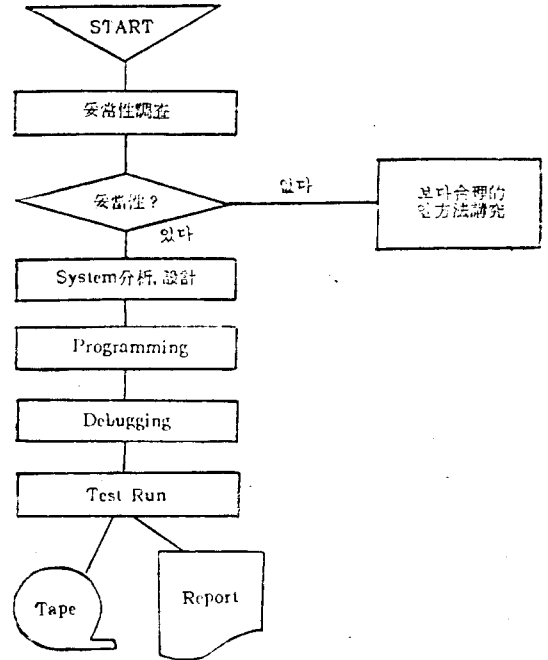
2. 問題 또는 作業의 分析

科學計算이나 事務處理에서나 問題를 處理할 EDPS의 能力和 問題의 特性을 考慮해서 作業이나 問題를 分析하여야 된다. 이를테면 科學計算의 境遇에는 微積分의 式은 數值解析의 手法에 依하여 加減乘除算으로 바꾸고 또 事務處理의 境遇에는 해야할 作業을 EDPS가 處理할수 있도록 分析한다.

3. Flow Chart의 作成

分析이 끝나면 問題解決의 段階를 大略적으로 Block Diagram으로 그린다 이것은 處理할 內容을 圖形으로 表現하는 것으로 節次의 흐름을 손쉽게 把握하고 誤謬의 發生을 防止하며 訂正을 容易하게 하는데 必要하다. Block Diagram을 作成하고 檢討한 後에 計算機가 할 일을 考慮하여 더 仔細하게 說明을 하는 Detailed Flow Chart를 만든다.

EDPS의 運轉順序를 Block Diagram으로 그려 보면 아래와 같다.



4. Coding

Detailed Flow Chart의 內容을 直接 機械가 읽을수 있도록 翻譯하는 일을 Coding이라 한다. Coding에는 機械語 Coding (Machine Coding)과 記號 Coding (Symbolic Coding), Compiler Coding이 있다. Machine Coding은 Programming中 가장 基本的, 原始의 方法이고 Software 面의 開發이 進步되지 않았던 數年前까지는 이 方法을 많이 使用했다. 이 方法으로 하면 各 EDPS는 各各의 獨特한 命令語와 命令形式이 있어서 그 命令語와 命令形式에 따라서 演算해야 할 順序로 Program한다. Symbolic Coding은 어떤 命令을 Symbol로써 表現하여 Coding하는 것이나 Machine Coding했을 경우보다 여러가지 便利한 點이 많고 Programmer의 誤謬나 負擔을 減輕된다. 作成할때는 記號 Coding으로하여 EDPS가 機械語로 번역하게 한다. 그러나 Machine Coding된 Program은 機械에 讀入되면 即時 演算할수 있으나 Symbolic Coding된 Program은 EDPS에 記憶시켜둔 번역 Processor인 Program에 依해서 一次 Machine Coding으로 翻譯되어야 하기 때문에 記憶容量을 많이 使用하고 時間이 더 많이 必要하다는 短點이 있기는 하지만 現代와 같이 大容量의 記憶裝置와 處理時間의 高速化는 이를 充分히 克服하고 있다.

Compiler Coding은 日常의 計算式이나 言語에 가장 가까운 表現方法이다. 例를들면 $A+B=C$ 라는 計算을 할때도 $A+B=C$ 라고 記入해도 機械는 알아보는 것이다. 二對身 翻譯 Program이 複雜해진다.

5. Debugging

作成된 Program을 實際 EDPS에 動作시켜서 誤謬가 있는가 없는가 檢討하는 作業을 Debugging이라 한다.

V. Data 處理 System

處理 System를 大別한다면 一括處理와 實時間處理로 分類한다.

1. 一括處理(Batch Process)

生産管理, 販賣管理, 購買事務, 給與計算, 料金計算, 經理, 各種統計, 技術計算 등에서 볼수 있는 가장 一般적인 方法은 Data를 어떤 週期로 取어서 計算機에 入力하여 一括處理하는 것이다.

2. 實時間處理(Real Time Process)

座席豫約, 銀行業務 등에서 볼수있는 System으로서 應答의 適時性이 要求되는데 使用된다. 但 要求되는 應答速度는 業務에 따라 다르다. 普通의 境遇 Real Time Process는 Data의 發生場所와 電子計算機의 處理裝置가 直結되어 On line 으로서 情報의 受授가 實行된다. 이것을 On line Real Time Process라 한다.

VI. EDPS의 用途

EDPS는 거의 모든 業務를 遂行할수 있고 處理할수 없는 業務가 무엇인가를 묻는 것이 正確한 對答이 나올만큼 用途는 廣範圍하지마는 各分野別로 代表적인 것만 들추어 본다.

1. 經營分野

- ① 標準經營管理方式(MOS)
- ② 工事進行計劃
- ③ 標準原價管理制度
- ④ O.R., Simulation

2. 經濟分野

- ① 長期經濟開發計劃
- ② 景氣分析 및 豫測
- ③ 產業聯關分析
- ④ 物資需給計劃
- ⑤ 投資分析

3. 政府行政分野

- ① 租稅政策
- ② 年金制度業務
- ③ 糧穀管理
- ④ 情報檢索, 犯罪搜查
- ⑤ 輸送計劃
- ⑥ 貯金保險管理
- ⑦ 其他各種行政業務

4. 科學技術分野

- ① 氣象豫測
 - ② 送配電計算 및 其他電氣工學的인計算
 - ③ 道路網計劃
 - ④ 土積計算, 構造設計
 - ⑤ 工學設計, 其他科學計算
- 이 以外에 또 各種分析業務等을 處理할수 있다.

VII. 韓國電子計算所의 實績

昨年度 8月부터 稼動開始한 FACOM-222 汎用大型 電子計算組織으로 處理한 業務量은 各部門에 있어서 實로 多大하다고 하겠다.

1. 計算用役

- ① 韓國銀行의 外換統計
- ② 韓一銀行의 積金, 換業務
- ③ 大韓重石의 重石價格豫測, 原價豫測, 財務管理, 會計管理業務
- ④ 韓國電力의 昇壓配電方式에 依한 經濟性檢討, 潮流計算, 過渡安全度, 故障電流計算
- ⑤ 民主共和黨의 經濟豫測
- ⑥ 與韓化學의 資材管理業務
- ⑦ 數個大學 및 USOM의 回歸分析 및 統計業務

⑥ 其他 性格分析 등이 있다.

2. 妥當性調查 및 System分析用役

- ① 韓國銀行全業務의 System分析
- ② 政府 및 金融機關에 對한 EDPS 妥當性調査

3. 要員養成

- ① Programmer (基礎 및 高級包含) 400餘名
- ② Custom Engienre 38名
- ③ 公務員 EDPS 委託訓練指定機關으로 指定되어 現在 第一期 38名을 教育中에 있다.

以上 列擧한것 외에도 現在 政府, 國營企業體, 銀行 및 民間業體等에서 곧 着手를 서두르고 있는 바이다.

Ⅷ. 結 言

지금까지 EDPS의 概念과 Hardware 上으로본 構成과 機能 Software의 解說 發達過程과 處理 能力等을 未熟한데로 考察해 보았지만 EDPS는 어디까지나 人間이 事務處理의 道具로 必要에 依

해서 創造해 놓은 人間能力의 集約體인 하나의 器具에 지나지않는 程度에 不遇하며 스스로 生覺하거나 判斷하지 못한다. 그러나 여기에 人間의 意思를 주게되면 가장 充實하게 그리고 正確迅速하게 그것들을 處理하게 된다. 우리가 이러한 世紀의 寵兒인 電子計算組織을 運用하는데도 許多한 問題點들이 가로 놓여있다. 現在의 問題點도 가까운 將來에는 解決되리라 믿으면서 곧 EDPS 導入의 經營效果分析을 日本의 生産性本部調査에 依據해서 添言해 둔다.

効 果	比 率
1. 增大하는 事務量을 增員없이 處理한다	25%
2. 새로운 經營資料로 經營管理에 萬全을 期한다	18%
3. 合理化 및 改善, 標準化에 誘引한다	13%
4. 月決算을 遲滯없이 迅速處理把握한다	15%
5. 管理가 計劃的으로 可能하다	11%
6. 人員의 節減을 가져온다	9%
7. 經營者의 管理意識을 高揚시킨다	2%