

# 開閉回路의 論理設計(I)

高 明 三\*

— 차 례 —

- 1. 序 論
- 2. 디지털 시스템
  - 2-1 序 論
  - 2-2 數值系
  - 2-3 二進數의 演算
  - 2-4 2進化 10進法 및 誤差檢出코오드
  - 2-5 論理開閉回路素子

## 1. 序 論

開閉回路(switching circuit)는 일명 離散回路(discrete circuit), 論理回路(logical circuit) 혹은 組合·順序回路(combinational and sequential circuit)라 부르며 이는 連續回路理論과는 전혀 다른, 現代電氣工學의 基礎分野인 回路理論의 一部를 형성하고 있는 새로운 學問이다. 즉 連續回路理論에서 취급하는 定常狀態에서의 回路의 入出力信號(電壓, 電流 등)는 주어진 時間領域에서 變遷現象이 없는 연속적인 波形이었으나, 開閉回路理論에서는 그 回路의 入出力信號를 어떤 종류의 波形보다 오히려 離散的으로 정의된 入出力値로 對應시키거나 回路 자체가 주어진 時間領域에서 高速開閉現象(switching phenomena)으로 인한 有限個의 內部狀態로 나타내는 것이 그 특징이다.

약 1940년까지의 소위 連續回路理論은 주로 交流理論과 濾波器構成論 및 多端子 回路網의 波動理論 등이 그 주류를 이루고 있었고, 그후 1950年代까지는 能動回路論분야에 많은 進전을 가져 왔다. 1960年을 前後로 한 디지털 計算機의 出現과 그 후의 눈부신 發展은 과거 通信工學者 및 技術者들이 발전시킨 開閉回路理論을 體系化시키게 되었다. 情報社會 또는 自動化社會의 産業分野에 취업하고 있는 모든 技術者 또는 工學者들은 基礎工學의 基本理論의 하나<sup>1)</sup>인 開閉回路理論을 알고 있어야 하며, 1970年代에 사는 電氣 技術者에게는 특히 중요한 기초분야임을 강조하고 싶다.

앞으로 기술할 內容은 筆者가 지난 數年間 서울大學校 電氣工學科 및 成均館大學校 電子工學科 4學年學生들에게 강의한 內容을 現場電氣技術者(특히 計裝 및 制御分野)들이 이해할 수 있도록 다소 보충한 것이며 다음과 같은 順序에 따라 記述코저 한다.

- 1. 디지털시스템
- 2. 數學的豫備知識 및 應用
- 3. 組合開閉回路의 設計(I)
- 4. 組合開閉回路의 設計(II)
- 5. 順序開閉回路
- 6. 同期順序回路의 設計
- 7. 非同期順序回路의 設計
- 8. Threshold 論理回路
- 9. 不確定開閉回路
- 10. 論理시스템의 自動設計

## 2. 디지털시스템

### 2.1 序 論

디지털 計算機의 出現 이래, 일반적으로 디지털法(digital methods)은 技術的 및 商業的 努力의 모든 分野에서 급속히 應用되어 왔다. 만일 특별히 열거한다면 각종 형태의 디지털 計算機들은 가스, 鋼鐵, 시멘트 등의 제조과정에 혹은 道路, 鐵道 및 航空路의 管制라던가, 株流動過程과 保險記錄 등의 감독 또는 우주선의 軌道制御 등 그 應用分野는 무수히 많다. 1967年初에 이미 商業 및 科學分野를 除外한 순수한 process 工業에만 全世界적으로 설치된 process 制御 計算機는 약 1600臺에 이르렀다<sup>2)</sup>. 이와 並行하여 각종 IC素子의 出現은 덩가이면서 信賴성이 큰 minicomputer와 microprocessor의 폭넓은 應用과 數值制御工具, 安全保護裝置 및 通信系統의 制御와 같은 小시스템의 構成 및 運用에 digital 計算機 없는 digital技法의 應用을 加速화시켰다.

한편 digital system의 大型複雜化는 system 相互間의 효과적인 通信 및 情報傳送에 관한 문제를 새롭게 제기시켰다. 즉 digital system相互間, digital system과 analog system相互間 또는 人間-機械間의 경제적이면서 효율적인 信號의 授受問題인 境界問題

\* 正會員· 서울大學校 副教授(工博)·當學會編修理事

(interface problem)는 digital system에서 가장 중요한 문제의 하나이다. digital system에서의 境界問題의 하나라 볼 수 있는 計算機의 出入口裝置란 보더라도 typewriter로부터 현재는 可聽 및 可視型까지 登場하게 되었다.

小, 中 및 大型의 모든 종류의 digital system의 工學的 機能은 원천적으로 적당한 開閉回路와 이들 回路들의 結合에 관한 設計如何에 좌우된다.

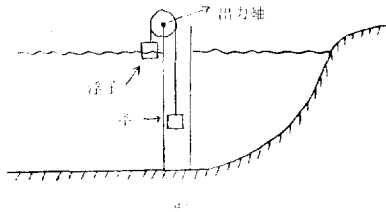
論理的 또는 組織的인 方法으로 여러 開閉回路들을 주어진 system의 入出力상태를 만조하도록 상호연결시키는 過程을 論理設計(logical design)라 한다. 이 論理設計過程에서 開閉回路專門家は 각종(電氣, 電子 및 機械的인) 標準 論理module를 開閉回路의 素子로 사용하여 앞으로 기술한 開閉理論의 原則에 입각하여 실지 設計를 수행한다. 論理設計過程으로

첫째는 問題의 명확한 設定과 대상 digital system에 대한 入出力特性和 機能에 관한 集合의 形成

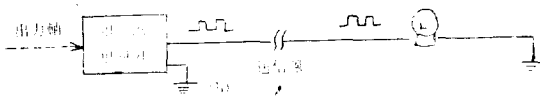
둘째는 費用, 信賴度, 補修의 難易度 및 사용할 開閉回路의 종류와 같은 구축條件을 고려한 最適의 hardware의 完成인

두가지 過程으로 구성된다.

그림 2.1은 遠隔 水位警報 digital system을 나타낸다.



(a) 水位檢出器



(b) 水位傳送 digital system

그림 2.1 遠隔水位警報 digital system.

즉 水位檢出器에서 檢出된 水位는 水位—電壓 變換를 거쳐 電壓으로 變換된 후 이를 다시 效果的인 遠隔 送信을 위해 2進數로 符號化한 후, 符號化된 信號에 대응되는 pulse를 pulse 발생기에서 발생, 이를 送信線을 통하여 遠隔地에 있는 中央水位管制所에 보내게 된다. 그림 2.1과 같은 system에 있어서 水位에 관한 情報를 어떻게 符號化할 것이며, 일단 符號化된 情報를 어떤 방식으로 發生하여 이를 遠隔地에 보낼 것인가

가 하는 등의 문제가 바로 開閉回路理論家가 해결해야 할 대상의 일부가 된다.

일반적으로 開閉理論은 大型 digital system의 設計에 效率的인 手段을 제공해 주지만 반드시 萬能的인 것은 못된다.

### 2.2 數值系

電氣的 혹은 機械的 裝置로 數值를 傳送하거나 다루기 위하여 이들 裝置들의 離散의 狀態 혹은 조건에 따라서 각각 다른 記號(10進數인 경우 0, 1, 2, ..., 9. 이고, 2進數인 경우 0, 1)를 나타내야 한다. 例를 들면 自動車의 走行計는 齒車列로 구성되어 있으며, 다섯자리의 10進數를 나타내는데 5個의 齒車列이 소요되며 각 齒車는 10개의 狀態를 나타내는 10개의 톱니가 있는 10 狀態裝置(ten-state device)라 볼 수 있다.

齒車列대신 電氣的으로 10進法에 의한 數值를 나타내기 위하여는 dekastron, digitron 혹은 특별히 제작된 電氣的인 10狀態裝置를 사용하여야 하지만 非能率的이다. 현재 모든 digital system의 數值의 符號化는 2狀態(0, 1)만을 나타내는 2進數法을 이용하고 있다. 그 理由는 스위치(ON/OFF), 릴레이(접속/비접속) 트랜지스터(導通/不通)와 같은 高速 2狀態素子를 손쉽게 얻을 수 있고, 또한 주어진 數를 나타내는 素子의 個數가 最小가 되는 것은 3進法이지만 현재 3值을 나타내는 實用的인 적당한 素子가 없기 때문이다.

일반적으로 數 $N$ 는  $R$ 進法에서 다음과 같이, 多項式

$$N = N_I + N_F = a_n R^n + a_{n-1} R^{n-1} + \dots + a_2 R^2 + a_1 R^1 + a_0 + a_{-1} R^{-1} + \dots + a_{-m} R^{-m} \dots \dots \dots (2.1)$$

- 단  $N$ : 正의 實數
- $N_I$ : 整數部,  $N_F$ : 小數部
- $R$ : 基數(radix)
- $a$ : 指標(digit)

으로 표시된다. 단일 基數  $R$ 이 10이면 10進數를, 12이면 12進數를 각각 나타낸다. 10進數 148.5를 여러가지 進法으로 나타내면 다음과 같다.

10進數:  $R=10$ . 記號 0, 1, 2, 3, 4, 5, 6, 7, 8, 9

$$(148.5)_{10} = 1 \cdot 10^2 + 4 \cdot 10^1 + 8 \cdot 10^0 + 5 \cdot 10^{-1}$$

12進數:  $R=12$ . 記號 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, T, E

$$(148.5)_{12} = (104.6)_{12}$$

$$= 1 \cdot 12^2 + 0 \cdot 12^1 + 4 \cdot 12^0 + 6 \cdot 12^{-1}$$

8進數:  $R=8$ , 記號 0, 1, 2, 3, 4, 5, 6, 7

$$(148.5)_{10} = (224.4)_8$$

$$= 2 \cdot 8^2 + 2 \cdot 8^1 + 4 \cdot 8^0 + 4 \cdot 8^{-1}$$

3進數:  $R=3$ , 記號 0, 1, 2

$$(148.5)_{10} = (12111.111\dots)_3$$

$$= 1 \cdot 3^4 + 2 \cdot 3^3 + 1 \cdot 3^2 + 1 \cdot 3^1 + 1 \cdot 3^0 + 1 \cdot 3^{-1} + 1 \cdot 3^{-2} + 1 \cdot 3^{-3} + \dots$$

2進數:  $R=2$ , 記號 0, 1

$$(148.5)_{10} = (10010100.1)_2$$

$$= 1 \times 2^7 + 0 \times 2^6 + 0 \times 2^5 + 1 \times 2^4 + 0 \times 2^3$$

$$+ 1 \times 2^2 + 0 \times 2^1 + 0 \times 2^0 + 1 \times 2^{-1}$$

표 2.1은 1부터 12까지의 數値를 上記한 進法에 準하여 표시한 것이다.

표 2.1 數値의 進法比較等值

10進數	12進數	8進數	3進數	2進數
0	0	0	0	0
1	1	1	1	1
2	2	2	2	10
3	3	3	10	11
4	4	4	11	100
5	5	5	12	101
6	6	6	20	110
7	7	7	21	111
8	8	10	22	1000
9	9	11	101	1001
10	T	12	101	1010
11	E	13	102	1011
12	10	14	110	1100

進數相互間의 變換에 대하여는 이미 여러 處에 記述되어 있음으로 여기서는 간단히 설명한다.

지금 基數 S로 표시된 整數(N)<sub>S</sub>를 基數 R인 數(N)<sub>R</sub>로 變換시키기 위하여 우선 N을 R로 다음과 같이 계속 S進數法으로 나눈다. 즉

$$\begin{array}{l} R \overline{) N} \qquad \qquad \qquad \text{나머지} \\ R \overline{) N_1} \dots \dots \dots a_0 \text{ 最低位} \\ R \overline{) N_2} \dots \dots \dots a_1 \\ R \overline{) N_3} \dots \dots \dots a_2 \\ \vdots \\ R \overline{) N_r} \dots \dots \dots a_{r-1} \\ \underline{0} \dots \dots \dots a_n \text{ 最高位} \end{array}$$

즉  $N = RN_1 + a_0, N_1 = RN_2 + a_1, \dots, N_r = R0 + a_n$   
 따라서  $N = a_n R^n + a_{n-1} R^{n-1} + \dots + a_1 R + a_0$  (2.2)

예제 2.1

(653)<sub>10</sub>의 (N)<sub>5</sub>으로의 變換은

$$\begin{array}{l} 5 \overline{) 653} \qquad \qquad \text{나머지} \\ 5 \overline{) 130} \dots \dots \dots 3 \\ 5 \overline{) 26} \dots \dots \dots 0 \\ 5 \overline{) 5} \dots \dots \dots 1 \\ 5 \overline{) 1} \dots \dots \dots 0 \\ 0 \dots \dots \dots 1 \end{array}$$

$(N)_5 = (10103)_5 = 1 \times 5^3 + 1 \times 5^2 + 3 \times 5^1 + 0 \times 5^0 = (653)_{10}$

예제 2.2

(1606)<sub>10</sub>의 (N)<sub>12</sub>으로의 變換은

$$\begin{array}{l} 12 \overline{) 1606} \qquad \qquad \text{나머지} \\ 12 \overline{) 133} \dots \dots \dots 10 = T \\ 12 \overline{) 11} \dots \dots \dots 1 \\ 0 \dots \dots \dots 11 = E \end{array}$$

$(E1T)_{12} = E \times 12^2 + 1 \times 12^1 + T \times 12^0$   
 $= 11 \times 144 + 12 + 10 \times 1 = (1606)_{10}$

다음 식 (2.1)에서의 소수부 N<sub>F</sub>의 變換에 대하여 고려하자.

$N_F = a_{-1}R^{-1} + a_{-2}R^{-2} + a_{-3}R^{-3} + \dots$  (2.3)

에서 基數 R에 대한 계수 a<sub>-1</sub>, a<sub>-2</sub>, a<sub>-3</sub>을 구하기 위하여는 식 (2.3)의 양변을 R로 곱하면

$RN_F = a_{-1} + a_{-2}R^{-1} + a_{-3}R^{-2} + \dots$  (2.4)

따라서 RN<sub>F</sub>의 정수부는 곧 a<sub>-1</sub>임을 알 수 있다. 저수 R를 곱하면

$R(RN_F - a_{-1}) = a_{-2} + a_{-3}R^{-1} + a_{-4}R^{-2} + \dots$  (2.5)

에 의하여 a<sub>-2</sub>를 결정할 수 있다. 이러한 조작을 계속 하면 小數部의 모든 계수를 구하게 된다.

예제 2.3

(0.61)<sub>10</sub>의 (N<sub>F</sub>)<sub>2</sub>으로의 變換은

식 (2.4) 및 (2.5)로부터

$$\begin{array}{l} 2(0.61) = 1.22 \dots \dots a_{-1} = 1 \\ 2(0.22) = 0.44 \dots \dots a_{-2} = 0 \\ 2(0.44) = 0.88 \dots \dots a_{-3} = 0 \\ 2(0.88) = 1.76 \dots \dots a_{-4} = 1 \\ 2(0.76) = 1.52 \dots \dots a_{-5} = 1 \\ 2(0.52) = 1.04 \dots \dots a_{-6} = 1 \\ 2(1.04) = 0.08 \dots \dots a_{-7} = 0 \end{array}$$

즉

$(0.61)_{10} = 0.100110 \dots$   
 $= 1 \times (2)^{-1} + 0 \times (2)^{-2} + 0 \times (2)^{-3} + 1$   
 $\times (2)^{-4} + 1 \times (2)^{-5} + 0 \times (2)^{-6} + \dots$   
 $= \frac{1}{2} + \frac{1}{16} + \frac{1}{32} + \frac{1}{64} \dots$   
 $= 0.5 + 0.0625 + 0.03125 + 0.015625 + \dots$   
 $= 0.609375$

이와 같이 無限級數로 주어지는 경우에는 進法變換에 의한 誤差가 수반된다.

8進數와 16進數는 2進數에 매우 유사하기 때문에 digital system에서 중요하다.

8=2<sup>3</sup> 이므로, 每 8進 digit는 3個의 bits(binary digits)에 대응된다. 즉

$(\underline{111} \ \underline{001} \ \underline{010} \ \underline{011} \ \underline{010} \ \underline{110} \ \underline{001})_2 = (7123.261)_8$

한편 16=2<sup>4</sup> 이므로, 각각의 16進 digit는 4個의 bits에 대응된다. 즉

$$\underbrace{(1101)}_D \underbrace{1011}_B \underbrace{10000110}_{8 \quad 6} \underbrace{1010 \quad 0001}_A \underbrace{\quad}_3 = (DB86.A3)_{16}$$

但 16進數는 기호 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F로 구성된다. 여기서 A, B, C, D, E, F의 기호는 아직 국제적으로 통일되어 있지 않으므로 임의로 어떤 기호를 사용하여도 무방하다.

2.3 二進數의 演算

2進數系는 이미 표 2·1에서 나타낸 바와 같이 0, 1의 두 記號만을 사용하므로, 스위치와 같은 2狀態素子로 모든 數值를 나타낼 수 있기 때문에 매우 경제적이다. 즉 同-數를 10進法으로 나타내는 것보다 그 자리 수가 길어 지지만, 그 數를 나타내는데 필요한 素子의 個數는 적어진다. 만일 10進數 10<sup>n</sup>을 2進數로 나타내거 위하여는

$$2^a \geq 10^n$$

$$b = \lceil n / \log_{16} 2 \rceil = n / 0.301 \quad (2.6)$$

을 만족하는 자리수의 약 3배의 자리수가 소요됨을 알 수 있다.

일반적으로 2進數系의 演算은 10進數系보다 더 간단하다.

2.3.1 加減算

2進法에서는 각 자리수에 "0"과 "1"의 두개의 숫자로 표현되므로, 加算에는 다음 네 가지만 있다.

$$\begin{array}{r} 0 \quad 0 \quad 1 \quad 1 \\ + 0 \quad + 1 \quad + 0 \quad + 1 \\ \hline 0 \quad 1 \quad 1 \quad 10 \end{array}$$

예제 2.4

(a) 2進數 10進數

$$\begin{array}{r} 101101 \dots\dots\dots 45 \\ + 10110 \dots\dots\dots 22 \\ \hline 100011 \dots\dots\dots 67 \end{array}$$

(b) 2進數 10進數

$$\begin{array}{r} 10101111 \dots\dots\dots 175 \\ 100101 \dots\dots\dots 37 \\ 111101 \dots\dots\dots 61 \\ + 110010 \dots\dots\dots 50 \\ \hline 101000011 \dots\dots\dots 323 \end{array}$$

(b)에서 第3列의 合은 1+1+1+1=10+1-1=11+1=100

가 되어 第5列의 자리수에 "1"이 加算됨을 주의하여야 한다.

減算에는 直接減算과 補數에 의한 減算法이 있다. 直接減算에는 다음 네 종류만 있다.

$$\begin{array}{r} 0 \quad 1 \quad 1 \quad 10 \\ - 0 \quad - 1 \quad - 0 \quad - 1 \\ \hline 0 \quad 0 \quad 1 \quad 01 \end{array}$$

예제 2.5

(b) 2進法 10進法

$$\begin{array}{r} 101110 \dots\dots\dots 46 \\ - 10001 \dots\dots\dots 17 \\ \hline 011101 \dots\dots\dots 29 \end{array}$$

(c)

$$\begin{array}{r} 111100 \dots\dots\dots 60 \\ - 110111 \dots\dots\dots 55 \\ \hline 000101 \dots\dots\dots 5 \end{array}$$

減算에는 위와 같이 직접 減算으로 실시하는 경우와 補數加算으로 減算을 하는 두가지 方法이 있다. 2進數의 補數에는 1에 대한 補數와 2에 대한 補數의 두가지가 있으며 각각 10進數의 9에 대한 補數 및 10에 대한 補數에 대응한다.

지금 네자리의 2進數 1000을 생각하자. 네자리의 2進數의 最大値는 1111이므로, 네자리의 2進數 1001의 1에 대한 補數는 1111-1001=0110 즉 0110이며, 2에 대한 補數는 10000-1001=0111 즉 0111이다. 표 1.2는 2進數의 補數의 對應表이다.

표 2.2 2進數의 補數

10 進 數	2 進 法	1에 대한補數	2에 대한補數
0	0000	1111	0000
1	0001	1110	1111
2	0010	1101	1110
3	0011	1100	1101
4	0100	1011	1100
5	0101	1010	1011
6	0110	1001	1010
7	0111	1000	1001
8	1000	0111	1000
9	1001	0110	0111

예제 2.6 補數에 의한 減算

우선 補數에 의한 10進數系의 減算 46-17=29를 고려하자. 10進數 17의 10에 대한 補數는 83(100-17=83)이고, 9에 대한 補數는 82(99-17=22)이다.

따라서

$$\begin{aligned} 46-17 &= 46 + (100-17-100) = 46 + \underbrace{(100-17)}_{10\text{에 대한補數}(83)} - 100 \\ &= 46-83-100 \\ &= 129-100=29 \end{aligned} \quad (2.7)$$

$$\begin{aligned} 46-17 &= 46 + (99-17-100-1) = 46 + \underbrace{(99-17)}_{9\text{에 대한補數}(82)} - 100 - 1 \\ &= 46+82-100-1 \\ &= 128-100+1=29 \end{aligned} \quad (2.8)$$

식 (2.7) 및 (2.8)에 의하여 46-17의 減算계산이 補數加算으로 변환되었음을 알 수 있다.

위와 같은 演算을 2進數系로 표시한다면 다음과 같이 補數加算으로 減算演算이 가능함을 알 수 있다.

(a) 1에 대한 補數에 의한 경우

$$\begin{array}{r} 00101110 \dots\dots\dots 46 \\ + 11101110 \dots\dots\dots -17\text{의 } 1\text{에 대한補數} \\ \hline 100011100 \dots\dots\dots \\ \dots\dots\dots 1 \\ \hline 00011101 \dots\dots\dots 29 \end{array}$$

(b) 2에 대한 補數에 의한 경우

$$\begin{array}{r}
 00101110 \dots\dots 46 \\
 + 11101111 \dots\dots -17 \text{의 2에 대한 補數} \\
 \hline
 100011101 \quad 29 \\
 \uparrow \\
 \text{무시}
 \end{array}$$

$$\begin{array}{r}
 1101100 \quad 108 \\
 11011)101101101101 \dots 27)2925 \\
 \underline{11011} \quad 27 \\
 100101 \quad 225 \\
 \underline{11011} \quad 216 \\
 101001 \quad 9 \\
 \underline{11011} \\
 11101 \\
 \underline{11011} \\
 1001
 \end{array}$$

2.3.2 乘除算

10進法에서의 九九算에 해당하는 2進法の 그것은 나 음 네가지 뿐이다.

$$\begin{array}{l}
 0 \times 0 = 0 \\
 0 \times 1 = 0 \\
 1 \times 0 = 0 \\
 1 \times 1 = 1
 \end{array}$$

예제 2.7

$$\begin{array}{r}
 1101101 \dots\dots 109 \\
 \times 1011 \dots\dots \times 11 \\
 \hline
 1101101 \quad 1199 \\
 1101101 \\
 0000000 \\
 1101101 \\
 \hline
 10010101111
 \end{array}$$

除算은 10進法の 除法과 똑같은 方法으로 실시한다.

예제 2.8

$$\begin{array}{r}
 (a) \quad \begin{array}{r} 0011 \\ 11)1001 \dots 3)9 \\ \hline 11 \\ \hline 11 \\ \hline 0 \end{array}
 \end{array}$$

2.4 2進化 10進法 및 誤差檢出 코오드(binary coded decimals and error-detecting code)

計算機 또는 digital system에서 취급하는 數値는 2進法이기 때문에 system의 入出口에서는 각각 10進法→2進法, 2進法→10進法の 變換이 필요하게 된다.

10進↔2進變換은 매우 복잡한 회로를 필요로 하며, 자리수의 증가로 그 복잡성도 증대된다. 10進法↔2進法の 變換을 비교적 쉽게 할 수 있고, 또한 2進法の 특징을 살리는 것으로 2進化10進法이 있다. 이 2進化10進法은 10進法の 각자리의 數値를 2進數로 나타낸 것으로, 예를 들면 10進數 9873은

$$\begin{array}{cccc}
 \underbrace{1001}_9 & \underbrace{1000}_8 & \underbrace{0111}_7 & \underbrace{0011}_3
 \end{array}$$

와 같이 된다. 10進法の 한자리는 최대 9이므로 각 자리는 4 bit로 표시될 수 있다.

小型桌上計算機인 경우에는 變換이 쉬운 2進化10進法을 採用한 것이 많으나(실지 演算시에는 별도 補正을 강구하여야 한다. 본격적인 計算機에서는 演算速度

表 2.3 10進法과 各 code의 對應表

10進法	2 進法	Binary coded decimal			Excess three code	Gray code	2 out of 5 code	Bi-quinary code
		8421	7421	5211				
0	0000	0000	0000	0600	0011	0000	00011	01 00001
1	0001	0001	0001	0001	0100	0001	00101	01 00010
2	0010	0010	0010	0100	0101	0011	00110	01 00100
3	0011	0011	0011	0110	0110	0010	01001	01 01000
4	0100	0100	0100	0111	0111	0110	01010	01 10000
5	0101	0101	0101	1000	1000	0111	01100	10 00001
6	0110	0110	0110	1001	1001	0101	10001	10 00010
7	0111	0111	1000	1011	1010	0100	10010	10 00100
8	1000	1000	1001	1110	1011	1100	10100	10 01000
9	1001	1001	1010	1111	1100	1101	11000	10 10000
10	1010							
11	1011							
12	1100							
13	1101							
14	1110							
15	1111							

또는 시스템의 機能(예를들면, digital shaft position encoder)을 고려하여 표 2.3과 같은 여러가지 code가 사용된다. 表 2.3이외에도 5421 code, 2421code, 및 3321 code등 여러가지 있다.

7421 code는 1의 개수가 최고 2회만 나타나는 특징이 있고, 5211 및 Excess 3 code는 減算등에 필요로 하는 9에 대한 補數를 만들때 항상 0과 1을 교환하면 되는 利點이 있다. 한편 Gray code는 인접 digit間에 한개의 bit만 변함으로, shaft position encoder 등에 이용된다.

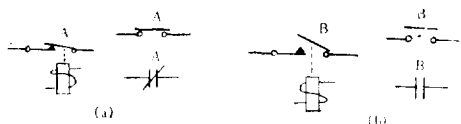
2 out of 5 code 및 Bi-quinary code는 0~9중 어느 數值도 전부 5 bits 또는 7 bits중 "1"이 2個있는 것이 특징이며, 이는 "1"이 한개 혹은 3개 나타나므로써 오차를 손쉽게 檢出할 수 있는 利點이 있다.

誤差檢出 code는 odd parity check 8421 code도 있다. 이것은 8421 code의 각각의 "1"의 개수가 奇數가 되도록 1을 더 부가한 5bit로 된 code이다. Hamming<sup>9)</sup>은 message내에 있는 單一 또는 多重誤差를 檢出하거나 교정하는 code에 관하여 발표하였고 Diamond<sup>10)</sup>은 多重誤差를 檢出 할 수 있는 더 効果的인 code를 발표하였으나 이에 관한 소개는 여기서 省略하기로 한다.

### 2.5 論理開閉回路素子

digital계산기, 自動process制御 및 計裝시스템은 일반적으로 이미 주어진 指示대로 入力側에서의 勵起에 따라서 그 機能을 수행하게 되는데 이러한 機能들은 AND, OR, NAND, NOT, 및 NOR와 같은 基本開閉素子 및 이들의 複合作用에 의하여 수행된다.

그림 2.2는 N. O. (normally open) 스위치와 N. C. (normally close) 스위치 및 이들의 기호를 표시한다. N. O. 스위치는 信號電流가 없을 때는 OFF상태로 있다고 信號가 들어오면 ON되는 스위치이며, N. C. 스위치는 이의 반대이다.

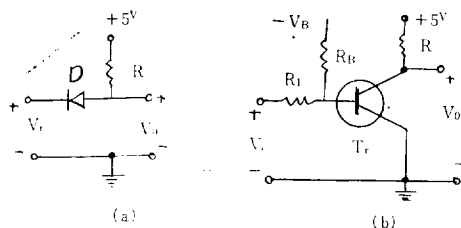


(a) N. C. 스위치의 기호 (b) N. O. 스위치의 기호

그림 2.2 N. C. 및 N. O. 스위치의 구조 및 기호

그림 2.2에서 A, B는 스위치의 상태를 나타내는 論理值이며, 만일 N. C. 스위치에서  $\bar{A}$ 라하면 이는 open 상태를, N. O. 스위치에서  $\bar{B}$ 라하면 ON상태를 각각 의미한다. 특히 論理開閉回路인 경우 스위치의 ON는 "1"로, OFF는 "0"으로 나타내는 것을 관례로 삼고 있다.

그림 2.3은 半導體를 이용한 스위치회로이다. 이의 動作原理는 다음과 같다.



(a) 다이오드 스위치 (b) 트랜지스터 스위치

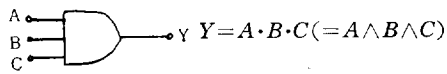
그림 2.3 半導體 스위치

그림 (a)에서 만일 入口電壓  $V_i$ 이  $V_i \geq 5$ 을 만족하면 다이오드 D는 逆 bias되어 D는 open상태가 되며, 出口電壓  $V_o$ 는  $V_o = 5[V]$ 가 되지만  $V_i < 5$ 이면 順 bias가 되어,  $V_o = 0[V]$ 가 된다. 즉 出口電壓은  $V_i$ 의 크기 여하에 따라  $V_o = 0$  혹은  $5[V]$ 가 됨을 알 수 있다. 스위치立場에서 보면  $V_o = 0$ 는 ON,  $V_o = 5$ 는 OFF을 의미한다. 한편 그림 (b)에서  $V_i = 0$ 이면  $V_b$ 에 의하여 트랜지스터  $T_r$ 의 emitter base間은 逆 bias 즉  $T_r$ 는 CUT-OFF領域에 속하게 되어 出口電壓  $V_o$ 는 약  $+5[V]$ 이지만,  $V_i > 0$ 이고  $R_1$ 과  $R_b$ 의 적절한 선정으로 큰 collector電流가 흐르므로  $T_r$ 는 飽和狀態가 되어  $V_o = 0$ 이 되어  $T_r$  자체가 스위치作用을 함을 알 수 있다. 오늘날 論理開閉素子는 대부분 diode와 transistor로 구성되며 IC技術로 제작된다.

1972년에 IC digital製品の 45%는 TTL(transistor-transistor Logic), 30%는 DTL(diode-transistor logic), 나머지 25%는 RTL(resistor transistor logic), DCTL(direct-coupled transistor logic) ECL(emitter-coupled logic) 및 MOSL(metal-oxide-semiconductor logic) 등이 차지하였다.

#### 2.5.1 AND素子

그림 2.4는 AND素子の 부호, 그 眞值表 및 실제회로를 나타내며, 그림 2.5는 relay, diode 및 transistor로 구성된 AND素子を 각각 나타낸다. 眞值表에서



ABC	Y
0 0 0	0
0 0 1	0
0 1 0	0
0 1 1	0
1 0 0	0
1 0 1	0
1 1 0	0
1 1 1	1

그림 2.4 AND기호 및 眞值表

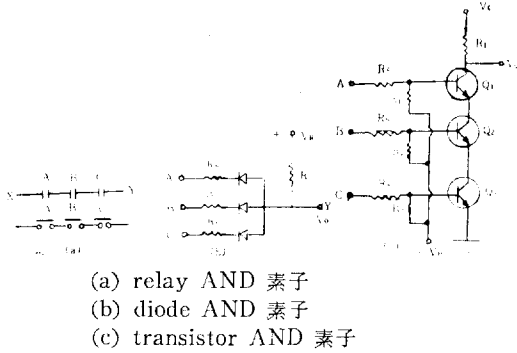


그림 2.5 relay 및 diode AND 素子

1 및 0은 해당 入出口 端子에서의 信號의 有無를 의미한다. 그림 2.6 (a)인 경우 relay A, B, C는 直列이므로, 모든 relay가 동시에 close되는 경우에 한해서만 XY間이 도통되므로 AND조건을 만족한다.

한편 同圖 (b)인 경우 入口端子 A, B, C중 어느 한端子에서의 電壓이  $V_R$ 보다 적으면 그端子와 出口 Y間은  $R_S$ 를 거쳐 順 bias되어 短絡狀態( $R_S \ll R$ 라 가정함)가 되어 出口 Y에서의 電壓은 0에 가까워진다. 반면 A, B, C의 모든 端子에서의 電壓이  $V_R$ 보다 크거나 같으면 세 diode는 逆 bias되어  $V_0 = V_R$ 가 되므로 결국 AND의 眞值表조건을 만족함을 알 수 있다. 마찬가지로 그림 (c)에서 CUT OFF以下로 bias된 상태에서 入口端子 A, B, C에 동시에 電壓이 인가되면  $Q_1, Q_2, Q_3$ 는 飽和가 되어  $V_0$ 는  $V_C$ 에서 0으로 변한다. 이들 AND 素子에서의 入出口值의 상태는 그림 2.7과 같다.

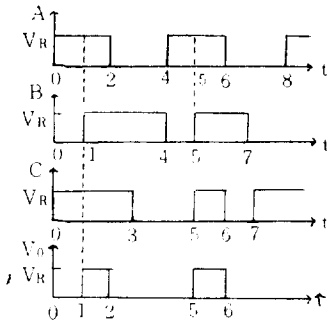


그림 2.6 AND素子の 入出口值

2.5.2 OR素子

그림 2.8은 그림 2.4와 같이 OR素子の 記號 및 그 眞值表를 나타내며 그림 2.9는 그림 2.5와 같이 relay, diode 및 transistor로 구성된 OR素子이며, 眞值表의 0, 1은 AND의 경우와 마찬가지로다.

relay 및 diode OR 素子에서 入出口值의 時間空間에서의 상태는 그림 2.10 (a)와 같다. 즉 (b)도인 경우 A, B, C 중 어느 한 入口端子에 信號(正電壓)가인

$Y = A + B + C$   
 $(= A \vee B \vee C)$

A	B	C	Y
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

그림 2.7 OR 기호 및 眞值表

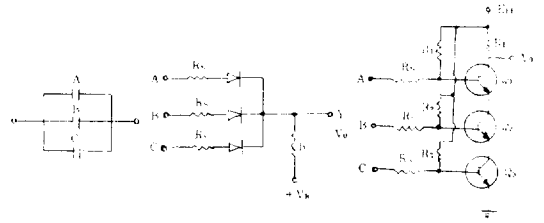


그림 2.8 relay 및 diode OR 素子

가 되면  $R_S \ll R$ 인 조건하에서는 해당 diode는 順 bias되어 出口 Y에서는 印加된 信號電壓에 아주 가까운 값으로 나타나게 된다. 한편 (c)도에서는  $Q_1, Q_2, Q_3$ 가 共히 飽和되도록 bias 電壓 및  $R_1, R_2, R_3$  및  $R_L$ 를 선정 한 후 어느 하나의 transistor 信號가 인가되면 CUTOFF가 되어 出口電壓  $v_0'$ 는 0에서  $V_{bb}$ 로 변하게 된다.

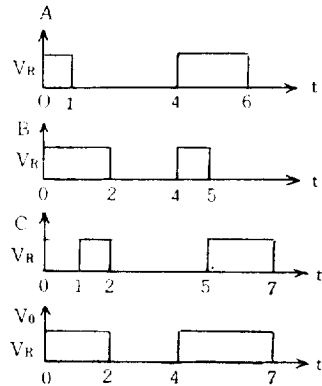


그림 2.9 OR素子에서의 入出口值

2.5.3 NOT 素子

NOT素子は INVERTER素子라고도 부르며 그림 2.11은 NOT素子の 기호 및 眞值表이다. 즉 入出口에서의 値가 상호 反對가 된다.

그림 2.12는 relay 및 transistor NOT素子로써, relay NOT素子인 경우 勵磁 coil에 信號電流 A가 존재하면(즉  $A=1$ ) normally closed스윙치는 열리고

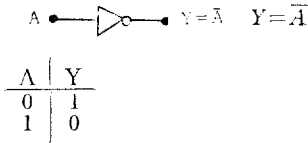


그림 2.10 NOT 素子の 기호 및 眞值表

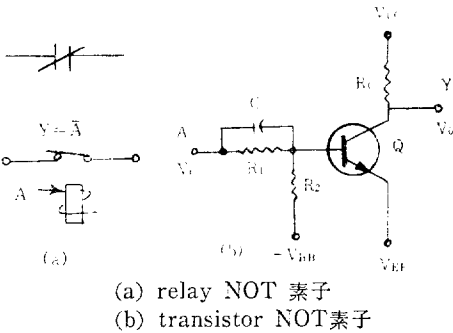


그림 2.11 Relay 및 Transistor NOT 素子

(즉  $Y=0=\overline{A}$ ),  $A=0$ 이면  $Y=1$ 이 된다. 한편 Transistor NOT 素子の 動作原理은 다음과 같다. 지금  $v_i = 0$ 시 transistor Q가 CUT-OFF 되도록 回路素子  $R_1$ ,  $R_2$  및  $R_C$ 를 선정하면,  $v_o = V_{CC}$ 가 될 것이다. 반대로  $v_i \geq | -V_{SB} |$ 이면 Q는 飽和狀態가 되어  $v_o = V_{EE}$ 가 되므로 NOT 眞值表를 만족한다.  $R_1$  양단의 C는 NOT 素子の 過渡應答을 改善하는 역할을 한다. 즉 capacitor C는 信號가 論理狀態를 NOT 조건으로 급변할 때 base內에 저장된 小數荷電子(minority-carrier charge)의 이동을 돕는다. 보통 C의 값은 100 pF 정도이며 정확한 값은 transistor에 의하여 결정된다.

그림 2.13은 NOT 素子の 入出口値를 나타낸다.

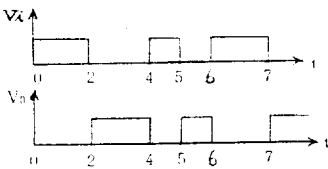


그림 2.12 NOT 素子の 入出口値

2.5.4 NAND 素子

AND 機能의 逆을 NAND라 하며, diode와 transistor의 論理組合(DTL), 혹은 transistor와 transistor의 論理組合(TTL)으로 再現한다. 現在 IC化된 DTL 또는 TTL등 여러가지가 있다. 그림 2.14는 NAND 素子の 부호 및 眞值表이다.

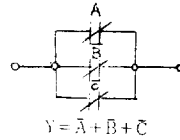
그림 2.15는 relay로 구성된 NAND 素子이고, 그림 2.16은 DTL NAND 素子の 基本型이다.

지금 電源 impedance  $R_s$ , 接合飽和電壓 및 順bias 시의 diode電壓降下를 무시한다. 즉 理想的인 diode 및 transistor로 가정한다. 만일 入口 A, B, C중 어느



A	B	C	Y
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

그림 2.13 NAND 素子の 부호 및 眞值表



$Y = \overline{A \cdot B \cdot C} = \overline{A} + \overline{B} + \overline{C}$

그림 2.14 relay NAND 素子

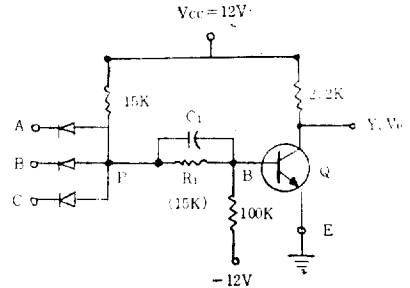


그림 2.15 DTL NAND 素子

하나라도 그 電壓이 0[V]이면 diode는 順 bias되어 點 P의 電位는 高이되어 base에서의 電壓  $V_B$ 는  $V_B < 0$ 가 되어 Q는 CUT OFF가 되므로  $v_o = 12[V]$ 가 된다.

한편 入口 A, B, C에 각각 電壓 12[V]을 인가하면 모든 diode는 逆 bias되고 transistor Q는 飽和狀態가 된다. 따라서  $v_o = 0[V]$ 가 되어 NAND의 眞值表를 만족함을 알 수 있다.

그림 2.17은 IC로 제작한 DTL NAND 素子이며 그림 2.16에서의 抵抗 대신 가능한 범위내에서 diode 혹은 transistor로 代置하였고, 抵抗値는 15KΩ에서 5KΩ로 單一電源 5V을 이용할 수 있게 한 것이다. 入口 A, B, C중 어느 하나가 0[V]이면 해당 入口에 연결된

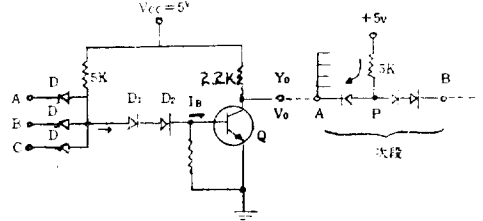


그림 2.16 IC型 DTL NAND 素子





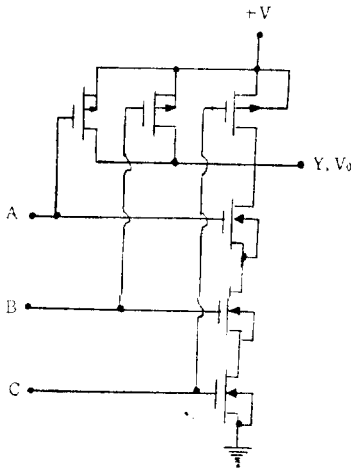
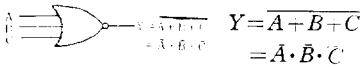


그림 2.22 CMOS NAND素子

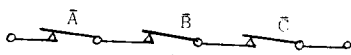
2.5.5 NOR素子

OR素子の逆을 NOR라하며, NAND와 마찬가지로 DTL, TTL, RTL 및 DCTL등 여러가지가 있다.



A	B	C	Y
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

그림 2.23 NOR素子の 부호 및 眞值表



$$Y = \overline{A \cdot B \cdot C} = \overline{A + B + C}$$

그림 2.24 Relay NOR素子

그림 2.23은 NOR素子の 부호 및 眞值表이다.

그림 2.24는 relay로 구성된 NOR素子이다. 한편 그림 2.16 자체는 負 NOR素子(入口信號電壓을 12[V]에서 0[V]로 변화시키는 경우이면, 眞值表의 0는 12[V] 1은 0[V]에 각각 대응됨)의 機能을 갖는다. 만일 그림 2.16에서 diode의 方向을 逆으로 하면 正 NOR素子(positive NOR gate, 眞值表의 0는 0[V], 1은 12[V]에 대응되고 入口信號는 0 [V]에서 12[V]로 변함)가 된다. 그림 2.25는 NOR素子에서의 入出口值이다.

그림 2.26은 RTL 正 NOR素子の 一例이며, 이는 silicon wafer에서의 空間占有率이 적기 때문에 매우 경제적이다.

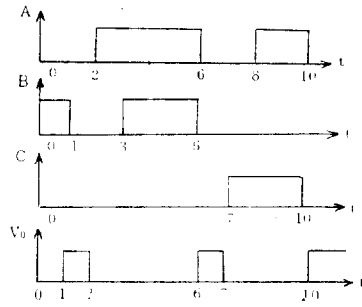
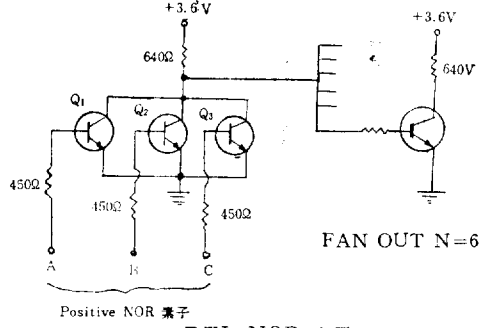


그림 2.25 NOR 入出口值



Positive NOR 素子  
그림 2.26 RTL NOR 素子

入口 A, B, C중 어느 한 入口에 높은 信號가 인가되면 해당 Q는 飽和가 되어  $v_0 \approx 0[V]$  그러나 모든 入口電壓이 零이면 Q는 CUT-OFF되어,  $v_0$ 는 높아져 NOR 眞值表를 만족함을 알 수 있다.

참고 문헌

- 1) Coates, etc.: An Undergraduate Computer Engineering Option for Electrical Engineering, Vol. 59, No. 6 pp. 854-859, Proc. of IEEE
- 2) Process Computer scorecard, Control Engng, 1967, Vol. 14, pp. 51-55.
- 3) Scott, N.R. Analog and digital computer technology 2nd ed. McGraw-Hill, 1970
- 4) Hamming, R. W. Error-detecting and error-correcting codes. Bell Syst. tech. J., 1950 Vol. 29, 147-160.
- 5) Diamond, J. M. Checking codes for digital computers. Proc. Inst. Radio Engrs, 1955, Vol. 43, 487-488.
- 6) Millman-Halkias: Integrated Electronics, Chapter 6, 1972, McGraw-Hill.

[Quiz 1]

디지털 회로에서 2진법이 사용되는 이유를 數를 나타내는 素子の 個數와 나타낼 수 있는 數의 가지수 의 立場에서 설명하라. (解答은 다음號에 나옴)

<계속>

# 海外論文紹介

## 最適負荷配分

H. H. Happ: Optimal Power Dispatch  
IEEE Trans, Power Apparatus System Vol.  
PAS-93, No. 3, May/June, 1974 pp. 820~830

送電損失을 고려한 電力系統의 最適負荷配分에 관한 새로운 計算法을 提案하고 있는데 이 方法에 따르면 이제까지 發表된 것보다 간단하게 求할 수 있고 또 그 收束特性도 우수하다고 밝히고 있다.

널리 알려진 바와 같이 最適의 負荷配分은 모든 發電機(m)에 대하여

$$\left(\frac{dF}{dP}\right)_m PF_m = \lambda \dots\dots\dots (1)$$

$$PF_m = \frac{1}{1 - \frac{\partial P_L}{\partial P_m}} \dots\dots\dots (2)$$

을 만족하는 값( $P_1, P_2, \dots, P_m, \dots$ )이다.

여기서  $P_m$ : 發電機 m의 出力

$\left(\frac{dF}{dP}\right)_m$ : 發電機 m의 増分燃料費

$PF_m$ : 發電機 m의 Penalty factor

한편 送電損失  $P_L$ 는 B定數를 사용해서 近似的으로

$$P_L = \sum_{m,n} P_m B_{mn} P_n$$

처럼 ( $P_1 \sim P_m \sim$ )의 二次形式으로 求해지는데 여기서 係數行列 B를 얻는다는 것이 큰 문제였다. 그런데 새로 提案하는 이 方法에서는  $\partial P_L / \partial P_m$ 를 精確하게 다음式을 풀어서 얻도록 하고 있다.

$$\begin{matrix} \frac{\partial P_L}{\partial \theta} \\ \frac{\partial P_L}{\partial V} V \end{matrix} = \begin{matrix} \frac{\partial P}{\partial \theta} & \frac{\partial Q}{\partial \theta} \\ \frac{\partial P}{\partial V} V & \frac{\partial Q}{\partial V} V \end{matrix} \begin{matrix} \frac{\partial P_L}{\partial P} \\ \frac{\partial P_L}{\partial Q} \end{matrix} \dots (3)$$

(3)式的 左邊은 損失(電壓 E의 函數)

$$P_L + jQ_L = (E)'(YE)^*$$

를 실제로 偏微分해서 求하게 된다. (3)式的 右邊의 行列은 뉴턴·랩손(Newton Raphson)法의 潮流計算에 사용되는 야코비안(Jacobian) 行列을 轉置한 것이다. 이 결과 負荷配分計算은 潮流計算처럼 反復計算으로 求하게 되는데 그 收束特性은 Penalty factor 算定時

$$PF'_{new} = PF_{old} + \alpha(PF_{new} - PF_{old})$$

의 加速定數  $\alpha$ 에 좌우되지만 IEEE의 118 母線系에서 試算을 본 결과  $\alpha=0.5$ 에서 6회의 反復으로 收斂되었다. (宋吉永抄)

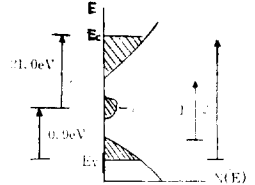
## 유리질액상 A<sub>1</sub>-S<sub>2</sub>합금의 전자적인 성질

Electronic Properties of Vitreous and Liquid A<sub>1</sub>-S<sub>2</sub> Alloys

C. H. Hurst and E. A. Davis. Journal of Non-Crystalline Solids 16, 1974.

이 論文은 A<sub>1</sub>를 30%~50% 포함하는 5種의 유리질 A<sub>1</sub>-S<sub>2</sub> 試料에 對한 導電率의 溫度依存性과 熱起電力, 光吸收等에 關한 實驗으로 電子의 性質을 究明하였다.

그림과 같이 이동단  $E_C - E_V$ 는 局在化되고  $E_F$ 는 에너지대 겹중양에 있는 Mott와 Davis의 狀態密度를 나타내는 에너지帶를 모형화하여  $\sigma = C \exp(-E_\sigma/kT)$ 인 式에서  $C=10^4 \Omega^{-1} \cdot \text{cm}^{-1}$ 인 定數이고 活性化 에너지  $E_\sigma$ 는  $E_F - E_V$ 에 해당되며



熱起電力  $S = \frac{k}{c} \left[ \frac{E_s}{kT} + B \right]$ 로 나타내는  $E_s$ 와  $E_\sigma$ 은 같다고 보아

$$E_s(\tau) = E_\sigma(\tau) = E_\sigma(0) - rT \text{ 로 두었다.}$$

$$\text{즉 } \sigma = C \exp(-E_\sigma(0)/kT)$$

인 關係를 利用하여  $A_{1x}S_{2100-x}$ 에서  $A_{12}S_{48}$ 때 移動度 係이 最少가 되고 펄리準位는 成分比率에 無關함을 밝혔다.

光學的인 吸收은 에너지 겹에 해당하는 스펙트럼 領域에서 잘코게나이드 유리는 Urbach 스펙트럼을 나타내고 吸收係數  $\alpha$ 는 周波數 W에 따라

$$\alpha = \alpha_0 \exp(T/W) \text{ 로 나타낸다.}$$

$A_{12}S_{48}$ 인 試料는  $\hbar W_0 = 1.75 \text{ eV}$  때 300K에서  $\alpha$ 가  $10^{-1} \text{ cm}^{-1}$ 에서  $10^4 \text{ cm}^{-1}$ 로 指數的으로 증가하고  $A_{1x}S_{2100-x}$ 에서  $x=50, 45, 40, 35, 30$ 때 活性化 에너지는 감소하다가 증가하고 도전율은 X가 클수록 증가하며 熱起電力은 X가 클수록 增加하는 경향을 밝혔다. (朴昌輝抄)

## 세멘트·프로세스의 計算機制御

J. Barber, et al.: Description of Process Control Progress Using On-line Computers. [IEEE Trans. Industr. Applic., Vol. IA-10, No. 1, Jan./Feb., 1974, p. 12-22]

이 論文은 세멘트·프란트에서의 DDC의 成功例에 關하여 기술한 것이다.

프로세스 제어를 계산기로 실시하는 경우, 그 對象 plant중, 어떤 process에서, 어떤 制御를 선택하는 것이 좋은가 하는 것이 제일 중요한 테마이다.

process control에 계산기를 도입함에 있어서 그

plant의 manager로부터 現場의 operator에 이르는 모든 人間이 計算機를 理解하며, 生産기一體가 되어 일을 하는 것에 만족감을 갖는 것이 중요하다. 이렇기 위하여는 生産기에 관한 훈련은 着實하게 하는 것이 중요한 점이다.

Martin Marietta Cement社에서 1966년에 Tulsa工場에 GEPAC-4040를 도입하여 analog의 setpoint 制御를 실시하다가 1969년에 Lyons工場에 LN 5200 (Computer 本體는 XDS製)를 설치하여 DDC를 하고 있다.

Lyons工場에서는, Cement plant를 Dryer, Raw Mill, Kiln-Cooler, Finish Mill의 네가지 分野로 나누어, 각각의 分野에서 process의 現象과 制御機能을 연결시키면서 Dryer 7 loop, Raw Mill 10 loop, Kiln Cooler 27 loop, Finish Mill 9 loop의 DDC loop의 동작을 해석하여 成功을 거두었다.

장래는 Hardware를 追加하여 그 기능을 높여 猶豫 決定이다. DDC는 柔軟성이 크기 때문에, 비교적 열가가 된다. Backup는 manual 操作으로 切換하게 하였다. DDC에 관한 문제의 하나는 data의 sampling率이며 analog에서의 data의 취급법과는 다른 생각을 하여야 한다. (高明三抄)

마이크로프로세서—그 現在와 未來

T. A. Lalotis: Microprocessors Present and Future [Computer, Vol. 7, No. 7, July, 1974, p. 20~24]

Microprocessor는 出現이래 아직 얼마되지 않았으나, 급속히 digital system의 設計部品으로서 成熟段階에 이르렀다. 이는 transistor以來의 劃期的인 일이다.

우선 microprocessor와 microcomputer와의 차이점을 명확히 할 필요가 있다. 「processor」란 말은 user가 推定한 program에 따라서 동작하며, data를 처리하는 部分을 의미하며, 「computer」란 말은 processor, memory, 및 入出力을 내포한 處理 system全體를 의미한다. 이런 뜻에서, 「microprocessor」는 「microcomputer」의 制御部分을 구성하는 部品이라고 정의할 수 있다.

microcomputer를 CPU, RAM, ROM, I/O의 各部分의 集합으로 생각하는 경우 일반적으로 여러개의 chip로 分割시킨다. 그 分割方法은 各社, 各機種에 따라서 다르다. 일반적으로 1個의 chip에 종합화하는 것은 열가지지만 융통성이 적다.

한편 복수 chip로 나누면 半導體面積이 커지기 때문에 많은 機能을 부여한다.

microprocessor의 용도는 電卓, 制御, data處理, 凡用 computer의 네 가지로 크게 분류된다. 前二者는

one chip의 최대 후보자이고, 後二者는 高性能, 高融通성이 요구되므로 복수 chip쪽이 더 적합하다.

현재의 microprocessor는 半導體 memory의 製造技術의 延長線上에서 memory chip를 擴張하기 위한 道具로서 出現되었다. 現在の microprocessor는 (1) 半導體의 제조 방법(pMOS)으로 제한되어, 그 속도는 劃期的인 것이 못되지만 (2) 그 性能에서는 用途가 充分하고 (3) MOS memory의 속도와 대등하다고 평가되고 있다. 한편 더 高速인 microprocessor에 대한 요구 있으므로, 「Nanoprocessor」도 말다귀는 아니지만 入出力機器의 制御등에 사용되기 때문에 단순한 구조로서도 充分하며, bipolar技術로 달성될 것이다. (高明三抄)

MOSFET를 사용한 샘플-홀드 회로

E. S. McVey & C. D. Fiskeaux; "A Sample-Hold Circuit using MOSFETs," (IEEE Transactions on Industrial Electronics and Control Instrumentation, Vol. IECI-21, No. 4, November 1974, pp. 262~263)

이 論文은 샘플-홀드 회로의 解析과 設計가 提示되고 더욱이 실제 회로를 자세히 나타내고 있다.

여기에서 MOSFET는 종래 高度의 스위칭 速度와 낮은 "ON" 抵抗을 통한 적은 누설전류 때문에 널리 이용되었지만 良質의 샘플-홀드 회로에 이용 가능성도 提供하고 있다. (林相暉抄)

時系列解析을 이용한 휴먼-오퍼레이터 性能의 모델링

S. M. Shimmers; "Modeling of Human Operator Performance utilizing Time Series Analysis," (IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-4, No. 5, September 1974, pp. 446~458)

이 論文에서는 時系列 解析을 이용하여 실제 入出力 데이터로부터 휴먼-오퍼레이터를 모델링하는 것에 관한 새로운 방법이 시도되고 있다.

처음에 이 技法은 모델의 形態를 同定하고 실제 데이터에 基礎를 둔 同定된 모델의 變數를 推定하고 있다. 결과적으로 이것이 모델의 不適當性을 발견하게 되고 데이터에 관하여 모델 改善을 하게 해준다. 그리고 饋還系內에 人間要素의 모델을 決定하기 위한 時系列 技法 應用에 관하여 다루고 있다. 또한 여러가지 수준의 자극하에 人間 모델을 결정하는 接近 方法이 檢討되고 있다.

時系列法이 人間이나 어떤 다른 確定的 또는 確率的 過程이건간에 雜音으로 汚된 離散 觀測에 대한 모델링에 有用함을 보여주고 있다.

이 論文이 人間의 모델에 時系列 方法으로부터 最初로 試圖된 것이므로 어떤 觀點에서 새로이 밝혀진 結果가 人間 모델의 새로운 면을 나타내게 되고 問題點 解決에 도움이 될 것이다. (朴相禧抄)

遲延函數에 關한 近似式으로 부터도 同一한 過程에 依해서 필터가 定해진다. 이 方法으로 얻은 回歸形 필터는 安定性에 對해서 論하고 있으나 明確한 證明은 주어지지 않다. (李忠雄抄)

**A Direct Approximation Technique for Digital Filters and Equalizers**

G. C. Maenhout & W. Steenaert. IEEE Trans.,

Dirichlet條件을 滿足하는 周波數特性이 주어졌을 때 損失微分 또는 遲延函數를 만들어 各各 Fourier Sine, Cosine級數로 展開할 수 있는 경우에는 이것을 無回歸形이나 回歸形의 Digital filter의 損失微分, 遲延函數로 近似시킬 수 있는 것을 利用하여 返復計算이 없는 Digital filter의 設計法을 論하고 몇가지의 設計例를 들었다.

주어진 周波數特性  $F(jw)$ 에 對하여  $F(jw)=G(w)e^{j\theta(w)}$ 로 놓고 損失微分, 遲延函數를 各各  $dG(w)/dw$ ,  $\tau(w)=-d\theta(w)/dw$ 로 定義한다.  $\tau(w)$ 에 對해서도 同等하게 取扱할 수 있으므로 損失微分函數만 紹介한다.

$dG(w)/dw$ 는 Fourier sinc級數

$$\frac{dG(w)}{dw} = T \cdot \sum_{k=1}^{\infty} h_k \cdot \text{Sinc}kwT \dots\dots\dots(1)$$

로 展開되는 것으로 한다.

그런데 近似化한 필터의 펄스傳達函數를

$$F(Z) = \frac{P(S)}{Q(S)} \Big|_{e^{ST}=Z} \dots\dots\dots(2)$$

로 놓는다. 이 경우에 損失微分函數는

$$\frac{dG(w)}{dw} = \text{Im} \left[ \frac{Q'(s)}{Q(s)} - \frac{P'(s)}{P(s)} \right]_{s=jw} \dots\dots(3)$$

가 된다. 다음에  $Q'(s)/Q(s)$ 는

$$\frac{Q'(s)}{Q(s)} \approx -T \cdot (t_1 e^{-sT} + t_2 e^{-2sT} + \dots + t_n e^{-nsT}) \dots\dots\dots(4)$$

로 近似化된다. 마찬가지로  $P'(s)/P(s)$ 는  $e^{-sT}$ 의 冪級數에 展開되므로

$$\frac{P'(s)}{P(s)} \approx -T \cdot (v_1 e^{-sT} + v_2 e^{-2sT} + \dots + v_n e^{-nsT}) \dots\dots\dots(5)$$

가 된다. ( $m > n$ ). 式 (3), (4)와 (5)로부터

$$\frac{dG(w)}{dw} \approx T [(t_1 - v_1) \sin wT + \dots + (t_n - v_n) \sin nwT - v_m \sin mwT] \dots\dots(6)$$

을 얻을 수 있다. 式 (1)과 (6)의 처음의  $m$ 項을 比較하여  $\{t_i\}_{i=1}^m$ ,  $\{v_i\}_{i=1}^m$ 을 얻는다. 그런데  $\{t_i\}_{i=1}^m$ ,  $\{v_i\}_{i=1}^m$ 와  $\{a_i\}_{i=0}^m$ ,  $\{b_i\}_{i=1}^m$ 와의 사이에는 一定한 代數的인 關係가 있으므로 필터가 定해진다.

**不均一RC分布線路의 最適設計**

油田信一·高橋進一, 電子通信學會論文誌 A, Vol. 57-A, No.11 11月 1974. pp.786-793

RC分布線路의 形狀을 適當히 定하여 單一한 RC分布線路가 願하는 傳送特性을 갖도록 하는 設計方法을 考察하고 있다. 이 問題는 周波數域에서 常微分方程式系의 境界值最適化問題로서 定式化되고 特異最適制御問題와 一致한다. 따라서 制御理論으로 부터의 類推에 依하여 어떤 條件下에서는 RC分布線路의 最適形狀은 방향形으로 된다. 具體的인 形狀의 決定은 數理計劃法에 依存할 必要가 있으나 一般的인 勾配法을 使用하는 알고리즘과 방향形에 固定시키고 最適化하는 알고리즘을 考察하고 例로서 이 方法을 定位相角 어드미턴스 設計에 適用했다. (李忠雄抄)

**논리 회로의 해석을 위한 대수적 모델**

Giuseppe Fantauzzi: An algebraic model for the Analysis of Logical Circuits. IEEE Trans. Comp., Vol. C-23, No. 6 June 1974, pp.576-581

BOOLEAN 대수는 논리회로의 상세한 내용을 기술할 수 없다.

따라서 다가(多價) 논리대수에 관심을 두게 되었다. 이제까지는 3가대수와 METZE에 의한 4가대수가 고려되었으며 최근에는 LEWIS에 의하여 "5가대수"가 제안되었다.

이 논문에서는 기본적 전자 개폐동작시의 장애(Hazard) 및 전이(Transition)을 정확히 분석하기 위한 "9가대수"를 소개하였다.

9가대수는 크기가 정해지지 않은 지연(Delay)이 예상되는 개폐회로의 해석을 위한 도구로 쓰여지며, 여기에 사용되는 집합의 원소는  $0 \rightarrow 0$ ,  $0 \rightarrow 1$ ,  $1 \rightarrow 1$ ,  $1 \rightarrow 0$ 의 기본적 4가지 외에 이의 변형 및 Race조건을 더한 9가지로 되어 있다.

원소들은 이미 기술된 기본적 네가지의 전이는 장애가 없는 안정한 직선전이(straight transition)이고, 나머지는 같은 전이지만 장애가 수반되는 것들이다.

인용된 예에서는 개폐 gate의 기본회로인 inverter, NOR, OR, AND 및 NAND에 대한 각 원소들의 가능한 입력에 따른 출력을 조사하였다.