

# 不完全 多重出力 函數를 利用한 BD프로그래머블 제어기

論 文
34~1~1

## BD Programmable Controller Utilizing an Incomplete Multiple Output Function

禹 廣 芳\* · 安 敏 良\*\* · 金 榮 日\*\*\* · 金 賢 起\*\*\*  
 (Kwang-Bang Woo · Min Ock An · Yeong-Il Kim · Hyun-Kee Kim)

### Abstract

In this study we described Binary - Decision Method Programmable Controller that is faster than conventional Programmable Controller (P. C) using the Boolean method. An algorithm for the optimal BD program was systematically developed by using the incomplete multiple output function and the residual characteristic function. The potential applications of BD programmable controller for the problems of industrial control were illustrated, which include on-off sequential control function and also the tasks involving continuous data processing and PID control. An example was presented that has shown how BD programmable controller and ICU (Industrial Control Unit) controller can be applied to the sequential logics.

### 1. 序 論

프로그래머블 제어기 (Programmable Controller)가 1970年度 초에 처음 소개된 이래 그 機能은 빠른 속도로 擴張되어 가고 있다. 初期에 프로그래머블 제어기 (P. C)는 진동기, 솔레노이드 (solenoid), 액츄에이터 (actuator) 등의 릴레이盤의 論理的 機能 즉, on-off 시퀀스를 처리하였으나 최근에는 데이터推定, 복잡한 數學的 알고리즘, 서보모우터의 制御, 스텝형 制御, 自己診斷, 시스템 制御 등이 向上된 機能을 갖춘 知能的인 制御器로서 발전되어 가고 있다.<sup>5) 8)</sup>

P. C에 내장되어 있는 CPU는 全 入出力裝置를 走査 (scan) 하는데 보통 3 ~ 200 (msec) 가 소요되는데 이는 입출력장치의 數와 P. C의 特性에 좌우된다. P. C가 간단한 on-off 機能을 가진 시퀀스 제어에서는 走査率 (scan rate)이 그렇게 큰 比重을 차지하지 않지만 보다 복잡한 PID (Propor-

tional - Integral - Differential) 제어나 아날로그 제어를 다루는 시스템에서는 매우 큰 比重을 차지한다.

大型 시스템에서 스캔레이트를 증가시키기 위해서 비트슬라이스 (bit slice), 二重마이크로프로세서 (dual microprocessor), 多重프로세서, 스마트인출력 (smart input/output), 등의 여러가지 방식이 있으나 그 중에서 특히 병렬 및 다중프로세서 방식에 대한 연구개발이 활발하게 추진되고 있으며 多重走査 (multi scan) 즉, 프로그램을 數개의 區間으로 나누어 各 區間을 별도로 走査하는 多重프로그래밍 방식은 복잡하고 긴 시간이 소요되는 계산도 다른 論理에 영향을 주지 않고 實行된다.

종래의 P. C의 경우 마이크로 프로세서의 연산기능을 이용하여 스위칭함수 (switching function)를 프로그래머블 (programmable) 하게 하였는데 이는 근본적으로 부울대수 (Boolean algebra)를 이용하여 CPU에 전달하는 방식을 취하였으므로 프로그램 스텝 (program step) 수가 많은 제어대상에서는 주사율 (scan rate)이 큰 比重을 차지하여 속도면에 있어서 병렬프로세스 (parallel process) 방식을 채택한 BD 프로그래머블 제어기보다 빠른 P. C를 얻기가 곤란 하였다.

\*正會員: 延世大 工大 電氣工學科 教授 · 工博  
 \*\*正會員: 大林工專大 電氣科 教授  
 \*\*\*正會員: 延世大 大學院 電氣工學科 碩士課程  
 接受日字: 1984年 8月 22日

병렬 접속에 의하여 신속하게 계산처리가 가능한 하드와이어드論理 (hardwired logic) 와 스토어드 프로그램 (stored program) 을 사용하여 프로그램修正이 용이하도록 마이크로프로세서를 채택한 프로그래밍論理 制御器 (programmable logic controller) 의 특징을 지닌 BD 프로그래머블 제어기는 프로세서의 싸이클타임 문제의 해결이 가능하도록 한다. 이러한 BD 제어기에 관한 하드웨어와 소프트웨어의 特性을 종합적으로 檢討하고자 한다.

## 2. BD 프로그래머블 제어기의 시스템구성

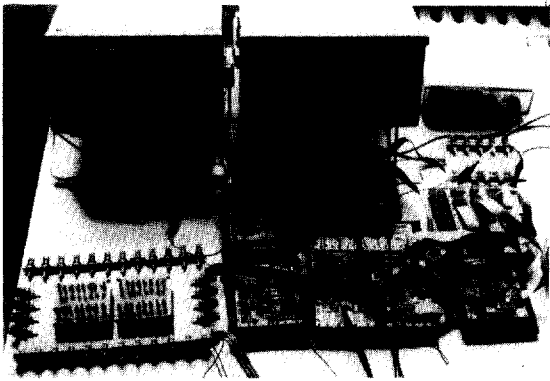


그림 1. BD 프로그래머블 제어기의 하드웨어 구성의 외관

Fig. 1. Picture of hardware architecture for the BD programmable controller.

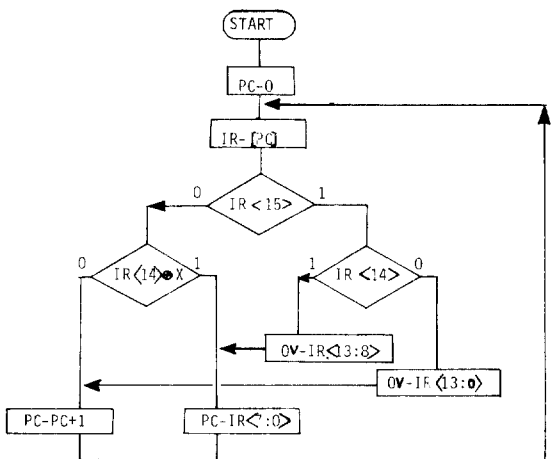


그림 2. BD 제어기의 명령어의 실행

Fig. 2. Execution of instruction in the BD controller.

BD 프로그래머블 제어기의 시스템 구성은 프로그램 메모리, 프로그램 카운터, 프리세트로직 (preset logic), 입력데이터선택타 (input data selector), 출력레지스터 (output register), 디스플레이 (display) 장치, 그리고 클럭 (clock) 을 포함한 5개의 블록 (block) 으로 구성되며 그림 1은 그 하드웨어의 외관을 보여준다.

BD 프로그래머블 제어기의 시스템은 3.1에서 기술한 ISP (Instruction Set Processor) 표기법으로 실행되며 그 시스템의 명령어 실행은 그림 2의 플로우차트 (flow chart) 에서 보여 준다. 또한 하드웨어의 設計要件으로는 CMOS 양립성, 직류전원장치, 자기진단 (自己診斷) 시스템, 병렬필드입출력 (parallel field input, output) 등이다.

Boute는 BD 프로그램을 이용하여 입출력 데이터를 포함한 각 프로세스 (process) 에서 반복실행하는 많은 수의 프로세스를 멀티플렉스 (multiplex) 하는 방법을 고안하였으며 프로세스의 수가 많은 경우에 시스템제어는 부울대수 (Boolean algebra) 를 채택한 제어가 유리하며 반면에 많은 변수 (variable) 를 포함한 프로세스의 수가 적은 경우에는 멀티프로그래밍 (multi programming) 方式을 채택한 BD 프로그래머블 제어기가 효율적이라는 것을 밝혔다.<sup>1)</sup>

소프트웨어 측면에서 불배 BD 프로그래머블 제어기는 릴레이래더다이아그램 (relay ladder diagram) 이나 진리표 (truth table) 로부터 직접 프로그램을 입력한 것이 아니고 최적한 BD 트리를 이용하여 알고리즘을 작성하는등 프로그램의 최적화 (optimization) 에 대한 전문적인 지식이 요구되며 범용 (general purpose) 으로 프로그램을 작성할 수 없는 단점을 지니고 있다.

### 2.1 하드웨어의 구성요소

本 BD 制御器의 제작은 Motorola CMOS 칩을 주로 사용하여 BD 器 (BD machine) 를 효율적으로 시뮬레이션 하였다.

5개의 블록을 각각 살펴보면 다음과 같다.<sup>3)</sup>

#### (a) 人力裝置

프로세스상태를 感知하기 위해서 32개의 입력변수 (input variable) IV < 31 : 0 > 가 사용되며 시뮬레이션하기 위해서 16개의 입력이 手動데이터 스위치에 연결되어 있다.<sup>11)</sup>

#### (b) 프로그램 記憶裝置

프로그램을 로우드 (load) 시킬때 BD 프로그램

命令語가 128 × 16비트 RAM안에 貯藏된다. 이 記憶裝置는 OP코우드의 명령어 (Instruction) IR < 15 : 14 > 에 따라서 점프 (jump) 하거나 필드출력으로 프로세스한다. 그러나 폭넓게 應用된 프로세스를 自體的으로 제어하는 경우에 記憶裝置는 ROM을 사용하여 프로그램을 휘엄웨어 (firmware) 로 施行한다.

(c) 制御裝置

9개의 手動制御 스위치가 사용되며 clock 시스템, 프로그램카운타, 1비트데이터 셀렉타가 포함된다.

프로그램카운타의 내용은 制御器의 출력에 따라서 命令語 IR < 7 : 0 > 에 증가하거나 프라세트(preset) 한다. clock은 2가지 모우드를 갖는데 그 중의 한가지는 오퍼레이터가 디버깅 (debugging) 하기 위해서 필요하다.

(d) 出力裝置

14개의 필드출력은 OP코우드 IR < 15 : 14 > 에 따라서 並列로 램치내에 스트로우브 된다. 이들 14개의 出力중에서 IR < 13 : 8 > 은 브랜치 명령 (branch instruction) 을 하느중 操作된다.

(e) 디스플레이 裝置

프로그램을 로우드시키거나 實行하는 중에 指定된 番地의 記憶場所의 內容과 番地를 디스플레이 하기 위해서 37개의 LED를 사용한다. 오퍼레이터가 디버깅 하는 중에 制御順序를 확인하기 위해서 IR < 13 : 8 > 및 IR < 13 : 0 > 의 필드출력을 디스플레이 하는데도 역시 LED를 사용한다.

위에서 설명한 (a) ~ (e) 의 各 部分을 그림 3의 블럭선도에서 보여준다.

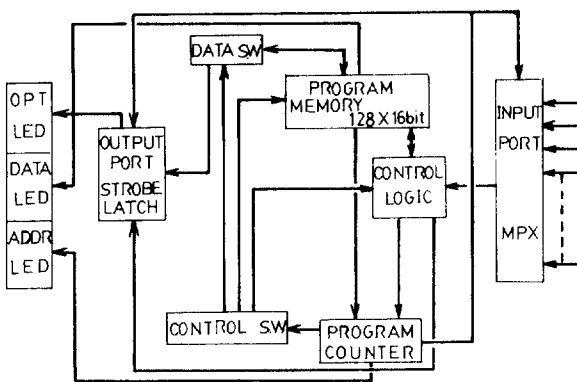


그림 3. BD 프로그램머블 제어기의 블럭선도

Fig. 3. Block diagram of the BD programmable controller.

2.2 不完全 多重出力函數(incomplete multiple output funtion) 및 레지듀얼特性函數 (residual characteristic function)에 관한 이론

BD 프로그램을 作成하기 이전에 BD 트리 (BD tree)의 개념을 도입하여 多重出力變數를 實行시키는 트리의 最少化 (minimization)가 이루어져야 한다.<sup>2)</sup>

不完全 多重出力函數를 갖는 最少의 BD 트리는 가장 적은 수의 판정노우드 (decision node) 를 갖는 함수를 실행하며 최적한 알고리즘을 제공해 준다. 또한 이 函數를 이용하여 트리를 合成하는 경우에 있어서 이 方法은 트리의 最適化를 체계적이고 용이하게 해결해 준다.

不完全 多重出力函數를 이용한 커버링트리 (covering tree)나 Karnaugh 도표를 이용한 트리의 합성방법에서 최적한 BD 트리의 세가지 조건 즉 包括性 (coverage), 分離性 (separability) 및 適合性 (compatibility)을 만족해야 하는데 임플리칸트 (implicant)의 수가 많은 경우에 적합성의 확인은 용이하지 않으므로 레지듀얼 특성 함수 (residual characteristic function)를 도입한 變數禁止 (Variable Inhibition)와 變數優先 (Variable Priority)의 법칙에 따른 2面型 (bidimensional form)을 작성하여 最適한 BD 트리의 조건을 만족시킨다.<sup>10)</sup>

3. BD 制御器의 소프트웨어에 관한 考察

最少의 BD 알고리즘을 얻기 위해서는 狀態變數의 合成을 통해서 비로서 最適한 트리가 가능하다. BD 프로그램을 最適化시키는 방법은 Cenry, Mange<sup>11)</sup> 등에 의해서 연구되었는데 최근에 BD 프로그램에 의해서 스위칭함수를 얻는 방법에 대한 관심이 상당히 높아가고 있다. 이러한 관심의 이유는 하드웨어와 소프트웨어의 문제를 포함한 광범위한 應用때문이다.

BD 다이어그램의 最適化에 관해서 Aker<sup>12)</sup>는 BD 프로그램을 사용하면 명령어의 수는 人力變數의 數를 초과하지 않는다고 結論을 내렸다.

本 論文에서는 不完全 多重出力函數 (incomplete multiple output function) 및 레지듀얼特性函數 (residual characteristic function)를 이용하여 最適한 알고리즘을 작성한다. 소프트웨어의 측면에서 볼때 마이크로프로그램은 IF - THEN - ELSE의 命令語를 사용하므로써 하드웨어적인 문제를 소프트웨어로 해결한다.

3.1 命令語의 型式

표 1에서 보는 바와 같이 16비트 命令語가 사용되며 ISP (Instruction - Set - Processor) 表記法이 이용된다.<sup>6)</sup>

표 1. 16비트 명령어의 형식

Table 1. Format of the 16bit instructions.

	15	14	13	8	7	0
Branch Instruction	0	V	Input Variable Address	Next Instruction		
Output Instruction	1	V	Output Value	Output/Next Address		

ISP는 BNF (Backus Normal Form)처럼 여러 컴퓨터나 制御器의 제작에서 하드웨어의 동작과 구조를 묘사해 주는 간결하고 일관된 文法 (grammar) 이 사용되기 때문에 BD 프로그래머블 제어기의 동작과 구조를 서술하는데 이용된다. 하드웨어를 구성하는데 있어서 ISP 표기법은 다른 言語보다 응용범위가 넓은 설계도구 (design tool) 로서 하드웨어의 시뮬레이션 (simulation) 과 製作 (architecture) 외에 소프트웨어의 구성, 프로그램의 확인, 시스템제어등에 널리 應用된다.

ISP 표기법은

- (a) 定規명칭
- (b) 시스템의 規格 및 동작순서의 묘사 그리고
- (c) 별칭연산자 (alias operator) 를 사용한 규정된 명칭의 분리 및 생략등을 포함한다.

한 예로 Mp [127 : 0] < 15 : 0 > 은 命令語記憶裝置 (instruction memory) 를 나타내며 괄호 “ [ ] ” 는 기억장소에 있어서 배열 (array) 의 차원 (dimension) 을 묘사한다. 즉 명령어 기억장치는 0에서 127로 限定된 128개의 기억장소 (memory location) 를 의미하며 괄호 “ < > ” 는 각 기억장소가 16비트를 갖는다는 것을 나타낸다.

3.2 BD 프로그래머블 제어기에 사용된 ISP 表記

- OP 코우드 OP < 1 : 0 > := IR < 15 : 14 >
- 命令語 記憶裝置 Mp [ 127 : 0 ] < 15 : 0 >
- 命令語 IR < 15 : 0 >

• 命令語 番地 IA < 7 : 0 > := PC < 7 : 0 >

• 入力變數 IV < 31 : 0 >

• 選擇入力 X := IV < IR < 13 : 8 > >

• 出力變數 OV < 13 : 0 >

• IR < 15 > = 0, IR < 14 > ⊕ X = 1 => PC ← Mp [ IR < 7 : 0 > ]

• IR < 15 > = 0, IR < 14 > ⊕ X = 0 => PC ← PC + 1

• IR < 15 > = 1, IR < 14 > = 1 => PC ← Mp [ IR < 7 : 0 > ], OV < 13 : 8 > ← IR < 13 : 8 >

• IR < 15 > = 1, IR < 14 > = 0 => PC ← PC + 1, OV < 13 : 0 > ← IR < 13 : 0 >

4. 實驗方法 및 考察

그림 4에서 보는 바와 같이 電動밸브 (electric valve) 나 셔터 (shutter) 등의 正逆操作制御 (forward-reverse operational control) 에서 2개의 出力을 이루는 함수 Y (D, C, B, A) 와 Z (D, C, B, A) 를 얻기위한 組合論理와 順次論理를 BD트리로부터 作成하였다. 여기서 A, B, C, D는 각각 PB/UP (PUSH BUTTON/UP), LSU (LIMIT S.W UP), PB/DN (PUSH BUTTON/DOWN), LSD (LIMIT S.W DOWN) 이다.

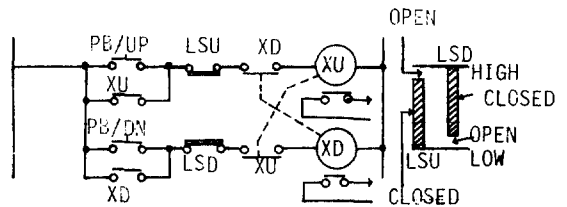


그림 4. 함수 Y, Z에 대한 正逆操作制御回路

Fig. 4. Circuit diagram of forward-reverse operational control for function Y, Z.

4.1 BD트리의 合成方法

出力 Y, Z 를 얻는데 필요한 Karnaugh 도표로부터 Meisel의 커버링표 (covering table) 를 작성하여 커버링트리 (covering tree) 를 표시하면

$$D C \bar{A} 01 - D C B A 01 - \bar{C} \bar{B} A 10 - \bar{D} C \bar{B} A 10 - \bar{C} \bar{A} 00 -$$

(12,14)                      (15)                      (1,9)                      (5)                      (0,2,8,10)

$$\overline{D}C\overline{A}_{00} - \overline{C}B A_{00} - \overline{D}CBA_{00} - DC\overline{B}A_{00}$$

(4,6)      (3,11)      (7)      (13)

이 커버링트리로부터 그림 5에서 보는 바와 같은 最適한 트리를 얻으며 이 트리가 最適한지를 증명하기 위한 세가지조건 포괄성, 분리성, 적합성을 만족하는가를 규명한다.

그림 5의 BD 트리에서 모든 임플리칸트는 포괄성과 분리성을 만족함을 알 수 있으나 적합성을 만족하고 있는가는 쉽게 알 수 없으므로 레지듀얼 특성함수를 이용한 二面型을 작성하여 적합성을 증명한다.

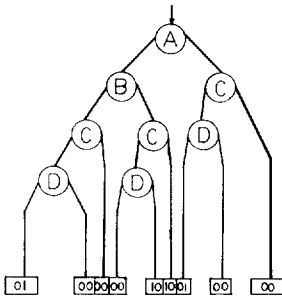


그림 5. 함수 Y와 Z에 대한 最適한 BD 트리  
Fig. 5. An optimal BD tree for output function Y, Z.

4.2 적합성을 얻기 위한 알고리즘

함수 Y와 Z에 대한 특성함수의 P1 (Prime Implicant)는 다음과 같다.

$$\{DC\overline{A}YZ, DCBAYZ, \overline{C}BAYZ, \overline{D}CBAYZ, \overline{C}AYZ, \overline{D}C\overline{A}YZ, \overline{C}BAYZ, \overline{D}CBAYZ, DC\overline{B}AYZ\}$$

출력 Y와 Z를 소거한 PI의 집합

$$\{DC\overline{A}, DCBA, \overline{C}BA, \overline{D}CBA, \overline{C}A, \overline{D}C\overline{A}, \overline{C}BA, \overline{D}CBA, DC\overline{B}A\}$$

Variable	D	C	B	A
weight wi	3	0	3	0

레지듀얼 특성함수의 변수법칙에 따라 관성변수를 A로 선택한다.

$$\overline{A} : \Phi (D, C, B, O, Y, Z) : \{DCYZ, \overline{C}YZ, \overline{D}CYZ\}$$

Variable	D	C
weight wi	1	0

$$\overline{C} : \Phi (D, O, B, O, Y, Z) : \{YZ\}$$

: 終端노우드

$$C : \Phi (D, I, B, O, Y, Z) : \{DYZ, \overline{D}YZ\}$$

: 終端노우드

$$A : \Phi (D, C, B, I, Y, Z) : \{DCBYZ, \overline{C}BYZ, \overline{D}C\overline{B}YZ, \overline{C}B\overline{Y}Z, \overline{D}C\overline{B}\overline{Y}Z, DC\overline{B}\overline{Y}\overline{Z}\}$$

Variable	D	C	B
weight wi	2	0	0

$$B : \Phi (D, C, I, I, Y, Z) : \{DCYZ, \overline{C}YZ, DCYZ, \overline{D}C\overline{Y}\overline{Z}\}$$

Weight wi	1	0
-----------	---	---

$$C : \Phi (D, I, I, I, Y, Z) : \{DYZ, \overline{D}YZ\}$$

:

$$\overline{C} : \Phi (D, O, I, I, Y, Z) : \{YZ\}$$

:

$$\overline{B} : \Phi (D, C, O, I, Y, Z) : \{\overline{C}YZ, \overline{D}CYZ, DCYZ\}$$

Variable	D	C
weight wi	1	0

$$\overline{C} : \Phi (D, O, O, I, Y, Z) : \{YZ\}$$

: 終端노우드

$$C : \Phi (D, I, O, I, Y, Z) : \{\overline{D}YZ, DYZ\}$$

: 終端노우드

4.3 二面型式(Bidimensional form)

4.2로부터 모든 브랜치는 중속변수를 갖는 레지

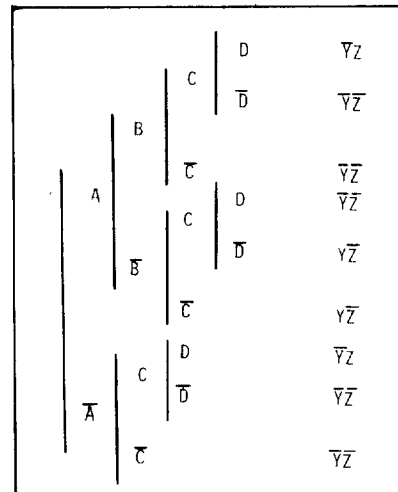


그림 6. 레지듀얼 특성함수에 의한 出力 Y와 Z의 二面型

Fig. 6. Bidimensional form of output Y, Z for residual characteristic function.

듀얼 특성함수에서 끝나며 2面型을 작성하면 그림 6과 같다.

모든 임플리컨트는 2面型으로부터 적합성을 만족하므로 그림 5에서의 BD 트리는 最適하다.

#### 4.4 프로그램의 作成 및 實行過程

##### 4.4.1 프로그램의 作成

最適화된 BD트리로부터 표 2의 명령어순서 (instruction sequence)를 작성하여 표 3과 같은 BD 프로그램을 얻는다.

표 2. BD 프로그래머블 제어기의 命令語의 順序  
Table 2. Instruction sequences for the BD programmable controller.

ADDRESS	INPUT STATE	TRUE BRANCH	FALSE BRANCH
0	A	1	5
1	B	2	7
2	C	3	9
3	D	4	9
4	OUTPUT YZ=01	GO TO 0	
5	C	6	9
6	D	4	9
7	C	8	10
8	D	9	10
9	OUTPUT YZ=00	GO TO 0	
10	OUTPUT YZ=10	GO TO 0	

##### 4.4.2 프로그램의 實行過程

BD 제어기를 로우딩모우드 (loading mode)로 設定한 후 프로그램 카운터를 리세트 (reset)시킨다. 코딩 (coding)된 프로그램은 手動 스위치에 의해서 RAM의 命令語 記憶場所 안으로 로우딩된다.

11개의 명령어 프로그램이 로우딩된 후에 BD 制御器는 實行모우드로 設定된다. 프로그램 카운터를 0으로 clear시키고 1[MHZ] 클럭주파수로 처음 명령어로부터 프로그램의 실행을 시작한다. 이때 strobe pulse는 clock 주파수의 하강구간에서 동기이다.

### 5. ICU(Industrial Control Unit)와 BD제어기의 프로그램스텝의 比較

ICU는 연속적인 비트情報에서 判定을 필요로하는 시스템에서 많이 사용되는 CMOS 프로세서이다. 그림 7은 기본적인 ICU블럭선도를 보여준다.<sup>9)</sup>

프로그램스텝數와 實行時間에 관해서 BD프로그래머블 제어기와 ICU를 比較해 보면 표 3의 BD 프로그램에서 11개의 命令語가 必要한데 比하여 표 4의

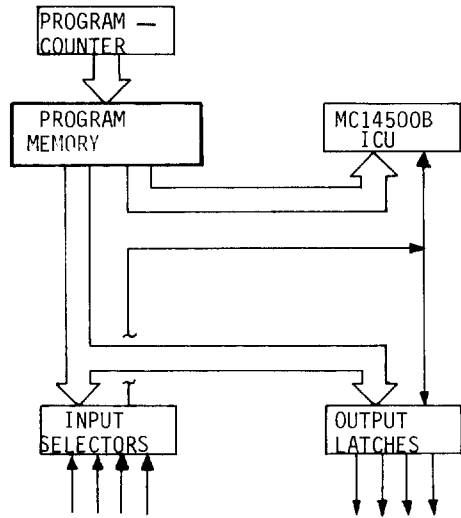


그림 7. 기본적인 ICU 계통도

Fig. 7. Basic ICU system.

표 3. 命令語順序로부터 작성된 BD 프로그램

Table 3. BD program using instruction sequences.

ADDRESS	INSTRUCTION	COMMENT
0	00000000 01 000000 00000101	TEST
1	00000001 01 000001 00000111	TEST
2	00000010 01 000010 00001001	TEST
3	00000011 01 000011 00001001	TEST
4	00000100 11 000001 00000000	OUTPUT
5	00000101 00 000010 00000110	TEST
6	00000110 00 000011 00000100	TEST
7	00000111 00 000010 00001000	TEST
8	00001000 00 000011 00001001	TEST
9	00001001 11 000000 00000000	OUTPUT
10	00001010 11 000010 00000000	OUTPUT

표 4. ICU 제어기의 命令語順序

Table 4. Instruction sequences for the ICU controller

ADDRESS	INSTRUCTION	ADDRESS	INSTRUCTION
0	LDC C	11	AND C
1	ANDC B	12	ANDC A
2	AND A	13	OR Z
3	STO Y	14	STO Z
4	LDC D	15	LD D
5	AND C	16	AND C
6	ANDC B	17	AND B
7	AND A	18	AND A
8	OR Y	19	OR Z
9	STO Y	20	STO Z
10	LD D		

ICU의 命令語順序는 21개의 명령어를 實行해야만 한다. 즉 BD 프로그래머블 제어기는 2개의 出力을 얻는데 3개의 스텝만으로서 가능한 것이다.

6. 結果考察

6.1 하드웨어의 考察

그림 8의 시한도표에서 보는 것처럼 데이터출력은 OP코우드 D14, D15에 의해서 變化한다. 本 實驗에서 데이터출력은 制御信號에서 발생한 스트로우브 펄스가 상승구간에 의해 操作(trigger) 될때 램치에 일단 저장되어 있다가 출력으로 傳達된다. 그림 9는 출력장치의 LED 디스플레이 부분의 블

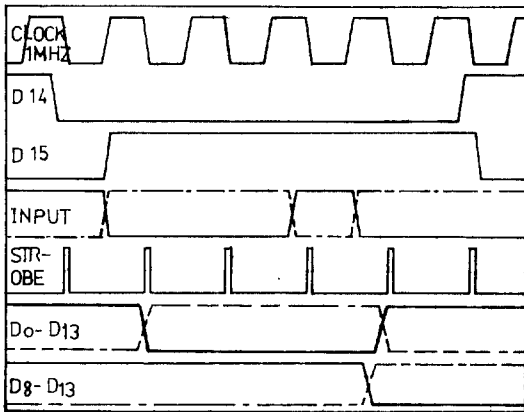


그림 8. 制御 및 出力裝置의 시한도표  
Fig. 8. Timing diagram for the control and output device.

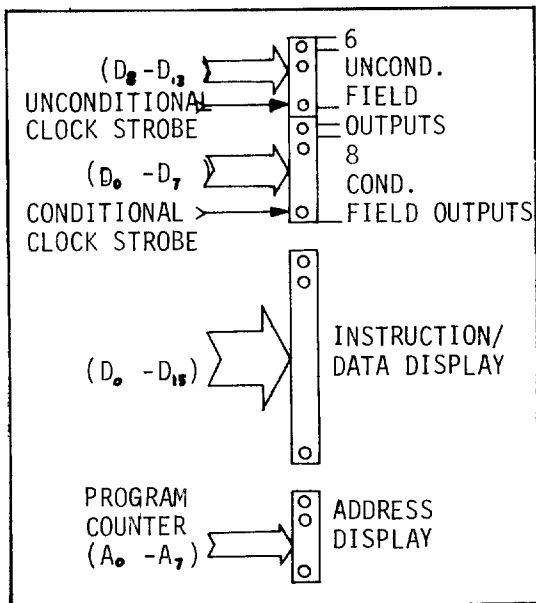


그림 9. 출력장치의 블럭선도  
Fig. 9. Block diagram of output device.

럭선도를 보여준다.

1비트선택 입력은 명령어 기억장치의 D < 13 : 8 > 에 의하여 변화하며 클럭주파수와는 비동기 (asynchronous) 이다.

本 實驗에서 프로그램의 入力 및 修正을 용이하도록 하기 위해서 수동제어스위치 (manual control S.W) 를 사용하여 모니터 프로그램 (monitor program) 을 수행하였으며 디버깅 (debugging) 하기 위해 사용한 클럭주파수는 0.5 [HZ] 로서 LED 디스플레이가 可能하도록 했다.

프로그램 스텝수가 많은 경우에 手動制御스위치가 모니터프로그램을 수행하기에는 어렵다는 결과를 얻었으며 이에 대한 대책으로서 ROM을 이용해야만 可能하다는 結論을 고찰하였다.

6.2 소프트웨어의 考察

最適한 BD 다이어그램을 얻는 方法은 不完全 多重出力函數의 P-函數와 G-函數等의 여러가지가 있다.<sup>7)</sup>

本 논문에서 提示된 方法을 살펴보기로 한다.

(a) 最少의 노우드수를 얻기위해서 어떤 방법을 화장시켜야 하는가?

— 레지뉴얼 特性函數를 이용하여  $m = b - 1$  의 式을 만족한다.

(b) 이러한 다이어그램을 一般化시키기 위해서 有用한 方法은 무엇인가?

— 키버링表를 作成하여 一般化시켰다.

(c) 여러 函數중에서 最適한 結果인지를 檢證하는 方法은?

— 바이나리트리의 3가지 特性중의 하나인 적합성을 만족하는가의 관계는 2面型을 作成하여 解決하였다.

6.3 BD 프로그래머를 제어기와 기존 P.C의 比較

BD 프로그래머를 제어기와 기존 P.C를 적용대상, universality, modularity, noise immunity 및 programming facility 등의 관점에서 비교해 보면 다음과 같다.

(a) 적용대상

마이크로프로세서를 채택한 P.C와 하드와이어드 (hard wired) 論理制御器는 명령프로세스가 불필요하고 스캔레이트에 큰 비중을 두지않는 제어시스템에서 많이 사용된다. 멀티프로그래밍 (multi programming) 방식을 채택한 BD 프로그래머를 제어기는 명령프로세스 및 증가된 스캔레이트를 要

하는 대형 시스템에서 같은 조건의 P.C나 하드와이어드 논리제어기 보다 효율적이다.

(b) 일반성 (universality)

기존의 P.C가 신뢰성 (reliability), 신축성 (flexibility), 편리성, 고장점검의 용이등의 일반적인 특징을 갖는데 비해 BD 프로그래머블 제어기는 모듈화 (modularity) 를 하기 힘든 점과 프로그램의 作成이 복잡한 단점을 지니고 있어 P.C 보다 편리성 및 신축성이 제한된다.

(c) modularity

기존 P.C의 모듈 (module) 구성은 대략 i/0 모듈, 시퀀서 (sequencer), 프로그래밍로더 (programming loader) 그리고 타이머 / 카운타모듈로 구분되며 제어대상이 필요로하는 신호 (signal) 의 類型에 따라 모듈을 선택할 수 있어 시스템의 확장 및 변경이 용이한 잇점을 지니고 있다. 반면에 BD 프로그래머블 제어기의 경우는 그 自體가 일종의 프로세서 역할을 하므로 모듈구성을 할수 없으며 시스템의 확장을 위해서는 BD 프로그래머블 제어기를 여러개 인터페이스 시켜야 하는 단점이 있다.

(d) noise immunity

기존 P.C와 마찬가지로 BD 프로그래머블 제어기가 시스템제어를 원활하게 수행하기 위한 대책을 보면

- 定電壓을 유지시키기 위한 스위칭회로의 설계,
- 전기적 노이즈 (noise) 를 제거시키기 위한 필터의 설계,
- 入出力電壓과 내부구동회로의 절연 (insulation) 을 위한 포토커플러 (photo coupler) 의 채택,
- 出力래치 (output latch) 에서의 비트신호 (bit signal) 을 외부로 구동시키기 위한 회로의 개발등을 들 수 있다.

(e) programming facility

기존 P.C는 플로우차트 및 릴레이래더다이아그램 (relay ladder diagram) 등을 이용하여 쉽게 프로그램할 수 있는 잇점을 지니고 있으나 BD 프로그래머블 제어기는 BD 트리를 最適하게 해주는 不完全 多重出力函数, P函数, 기타 여러가지 방법을 이용하여 알고리즘을 작성해야 하는등 프로그램의 最適化에 대한 지식이 요구된다.

7. 結 論

本 논문에서 BD 프로그래머블 제어기가 앞으로 開發해야할 課題를 살펴보면 다음과 같다.

- (1) 여러 素子로인한 傳播遲延時間 (propagation delay time) 의 減少에 관한 대책,
- (2) BD 프로그래머블 제어기의 機能을 확장 시키기 위한 마이크로컴퓨터 인터페이스에 관한 開發,
- (3) 最適한 BD 프로그램 컴파일러의 開發, 그리고
- (4) 組合論理 및 順次論理에 관한 프로그램의 체계적인 開發등을 들 수 있다.

BD 방식을 채택한 P.C는 단축된 論理經路 (logic path) 를 가지고 있기 때문에 그 응용범위는 메모리싸이클에 의해서 制限을 받는다. 또한 BD 프로그래머블 제어기가 서보제어기 또는 로봇의 제어 시스템 응용 이외에도 NC 旋盤의 制御, 대형플랜트 설비의 順次制御, PID 制御, 데이터推定, 아날로그信號의 發生等を 포함한 여러 산업기기 응용분야에서 광범위하게 활용될 것으로 기대된다.

參 考 文 獻

- 1) S. B Akers. "Binary-Decision Diagrams." IEEE Trans. computers. Vol. c -27, No. 6, June 1978
- 2) R. P. Voith "ULM implicants for minimization of universal logic module circuits." IEEE Trans. comput, Vol. c -26, pp. 417-424, May 1977.
- 3) Paul J. A. Zsombor -Murray, Louis J. Vroomen, Thole -Ngoc and peter H. Holck. "BD based programmable controller." IEEE Micro, INC, Aug. Oct, Dec, 1983
- 4) B. L. Beukema. "P.C of Data Transfer Rates for cycle steals Between Two processors." IBM Technical Disclosure Bulletin Vol. 24, No 9, pp. 4448 - 4451, Febraury 1982.
- 5) K. Rajaraman. "A Microprocessor -Based Time Sequence Controller for process Control Applications." IEEE Trans Ind. Electronics & Control Inst. Vol. IECI-28, No. 3, pp. 214 - 219, August 1981
- 6) Mario R. Barbacci "ISP: The Notation and It's Applications" IEEE Trans. computer, Vol. c-30, No. 1, pp. 25-32, January 1981
- 7) Andre Thayse. "P-Function : A New Tool for the Analysis and Synthesis of Binary programs." IEEE Trans. Computers. Vol. c -30, No. 2, pp. 126 - 134. Febraury 1981



- 8) M. Morris Mano, "Digital Logic Computer Design." prentice-Hall, 1979, pp. 114 - 121
- 9) V. Gragory, B. Dellande, etal, " Mc 14500 B ICU Hand book," Motorola Semiconductor products, Inc., Austin, Tx 74721 1977
- 10) E. Cenry, D. Mange and E. San che Z, "Synthesis of Minimal Binary Desision Trees." IEEE Trans. on comp. c-28, No. 7, July 1979
- 11) F. J. Hill "Digital System." Willey. 1978. pp. 489