

BD/Micro 프로그래머블 제어기와 이의 교통제어 시스템 응용

(Binary Decision/Micro Programmable Controller and Its Application to Traffic Control System)

禹 廣 芳*, 金 榮 日*, 安 敏 玉**

(Kwang Bang Woo, Yeong Il Kim and Min Ock An)

要 約

본 연구는 병렬 프로세스에 근거를 둔 BD 프로그래머블 제어기와 마이크로 컴퓨터를 통합한 하이브리드 BD/Micro 프로그래머블 제어기(Hybrid BD/Micro Programmable Controller)를 다루었다.

BD 프로그래머블 제어기 독자로는 수행할 수 없는 아날로그 신호의 발생, 데이터의 추정 등의 다양한 동작을 BD/Micro 프로그래머블 제어기로 확장시키므로써 시스템 제어의 기능을 수행토록 하였다. 이 BD/Micro P.C.를 이용하여 교통신호 제어기를 제작하고 이 제어기의 하드웨어 및 소프트웨어 개발이 검토되었다.

Abstract

In this paper, we investigate and characterize the parallel processing of a hybrid system interfacing binary decision (BD) programmable controller with a microcomputer. The generation of analog signals and estimation of data, which can not properly be performed by the BD programmable controller alone, have been done by the expanded controller utilizing BD/Micro P.C. In addition, implementation of its hardware and software is considered.

I. 序 論

프로그래머블 제어기(programmable controller)가 1970년도 초에 미국의 제너럴 모터사(GM)에서 처음으로 개발된 이래 그 기능은 비약적으로 발전되어 가고 있다.^{1), 1)} 종래의 시퀀스 콘트롤은 릴레이 로직

(relay logic)이나 타이머, 카운터 등의 주기능이었으나 최근에는 아날로그 입출력 취급이 가능하여 수치연산이나 데이터 처리, 서보 제어(servo control) 및 PID(Proportional-Integrated-Differential) 제어 그리고 시퀀스 제어 뿐만 아니라 피이드 백 제어도 가능하다. P.C.의 주 기능의 하나는 처리속도이다. P.C.에서는 다수의 입출력을 실시간(real time)으로 온라인 처리가 되어야 하기 때문에 처리속도가 늦으면 대응할 수가 없다. 이 처리시간은 스캐닝시간(scanning time)이라고도 하며 통상적인 P.C.에서는 보통 10~100ms 정도이다.

μP형 P.C는 그림 1에 보인바대로 상시 반복 프로그램 방식에 의거하여 프로그램 메모리내의 프로그램을 0스텝에서 최종스텝까지를 순차적으로 반복 실행한다.

*正會員, 延世大學校 電氣工學科
(Dept. of Electical Eng., Yonsei Univ.)

**正會員, 大林工業專門大學 電氣科
(Dept. of Elec. Eng., Daelim Tech. Junior College)

接受日字: 1985年 10月 11日

(本 研究는 韓國學術振興財團의 支援으로 이루어졌음.)

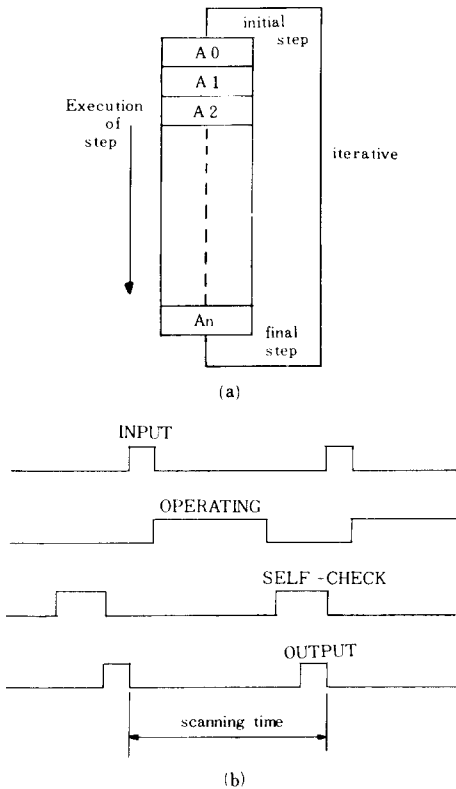


그림 1. μP 형 PC 의 상시 반복 연산 방식
 (a) 프로그램내에서 스텝의 실행
 (b) 타임 스케줄

Fig. 1. Iterative operation mode of μP based P. C
 (a) Execution step in the program memory,
 (b) Time schedule.

P. C의 주사율은 1/스캐닝 시간으로서 연산속도와 스텝수에 따라 결정된다.

μP 형 P. C의 경우 마이크로 프로세서의 연산기능을 이용하여 스위칭 함수를 프로그래머블하게 하였으며 이는 부울대수를 이용하여 CPU에 전달하는 형태이기 때문에 병렬 프로세스 방식을 채택한 BD(Binary Decision) 프로그래머블 제어기 보다 느린 속도의 P. C. 기능을 갖는다. 병렬 접속에 의해 신속한 계산처리가 가능한 하드 와이어드(hard-wired) 논리제어의 특징과 스토어드 프로그램(stored program) 방식의 μP 형 P. C.의 특징을 지닌 BD/Micro 프로그래머블 제어기는 순차제어와 반복 스위칭을 최적화한 BD 알고리즘을 이용한 펌웨어(firm ware)로 실현하여 스캐ليت, 즉 프로세서의 주기시간(cycle time) 문제를 보완해 준다. BD이론이 병렬프로세스 방식의¹⁸⁾ 근거가 되며 BD 프로그램이 부울스위칭(boolean switching) 회로에 대한 한가지 대안이 된다.

BD 이론 연구는 초기에 회로망, 스위칭 회로 및 병렬 처리 등의 분야에서 진행되었으며 BD 다이어그램의 최적화와 BD 알고리즘의 실행과 같은 그래프식 이론(graph-theoretical) 연구가 수행되었다.^{14, 5, 8)} 1976년 처음으로 Boute¹⁾이 등은 이러한 그래프식 이론 연구에서 제어를 목적으로 한 컴퓨터와 유사한 BD 장치의 실제 설계를 다루었으며 빠르고 극소화된 프로그래머블 제어기 개발이 큰 관심사로 되어 있다.

본 논문에서는 Lee¹³⁾ 등에 의하여 제안된 BD방식을 도입하여 프로그램 스텝 수를 단축시켜 속도를 개선할 수 있는 BD 프로그래머블 제어기능과 μP 형 제어방식에서의 펠치 주기시간(fetch cycle time)과 프로그램 주사시간(program scan time)을 최적화(optimal) 한 BD 알고리즘을 개발한다.

산업공정의 제어시스템에서 아날로그 신호의 발생 및 데이터 추정 등의 기능 확장을 위해서 BD 프로그래머블 제어기와 마이크로컴퓨터를 복합시킨 BD/Micro 프로그래머블 제어기를 교통제어시스템에 적용하여 그 제어기능을 검토하고자 한다. 각 교차로에 BD 프로그래머블 제어기를 채택한 교통신호 제어기를 사용하여 교차로를 연결시키고, 교통량 정보에 따르는 제어파라메타(control parameter)를 결정할 수 있도록 한다.

II. BD/Micro 프로그래머블 제어기의 시스템 구성

이 제어기의 시스템 구성에서 중앙처리장치는 프로그램 기억장치(ROM), 스크래치 패드 메모리(scratch pad memory), 아날로그 데이터 임출력장치, MPU(micro processor unit)로 구성된 마이크로 컴퓨터 시스템이 담당한다. BD 프로그래머블 제어기는 데이터, 어드레스 그리고 콘트롤 버스를 경유하여 MPU와 인터페이스 된다.¹²⁾

BD 프로그래머블 제어기에 사용되는 소프트웨어는 P. C.가 채택하고 있는 릴레이래더 다이어그램(relay ladder diagram)이나 진리표(truth table)로부터 직접 프로그램을 입력한 것이 아니고 최적 BD 트리(BD tree)를 이용하여 알고리즘을 작성하는 프로그램의 최적화(optimization)가 이루어진다.

1. BD 프로그래머블 제어기의 시스템 구성

이 제어기의 시스템 구성은 프로그램 메모리, 입력 및 출력장치, 제어장치, 디스플레이 장치의 5개의 블록으로 구성되며(그림2A, 2B) 하이브리드 BD/Micro 프로그래머블 제어기에서 BD 제어기는 PCU(programmable control unit)라 하는 주변장치의 일부분으로 구성된다.

BD 프로그래머블 제어기의 시스템 동작은 ISP(Ins-

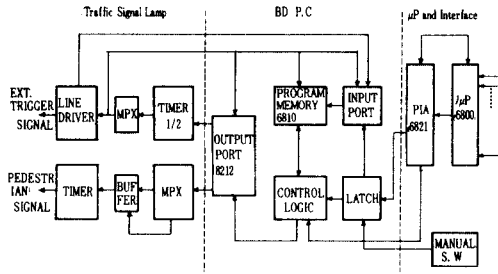


그림2(a). BD/Micro P.C와 교통신호등의 인터페이스 블록선도
 Fig.2(a). Interface block diagram of BD/Micro P.C and traffic signal lamps.

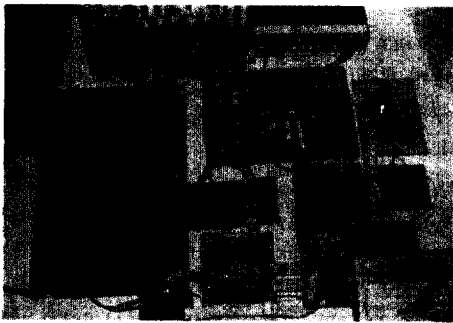


그림2(b). BD/Micro P.C.의 PCU의 하드웨어 구성의 외관
 Fig.2(b). Picture of hardware architecture for the PCU of BD/Micro programmable controller.

truction Set Processor) 표기법'으로 실행되며, 그 명령어 실행은 그림 3의 흐름도에서 보여준 바와 같이 처리된다.

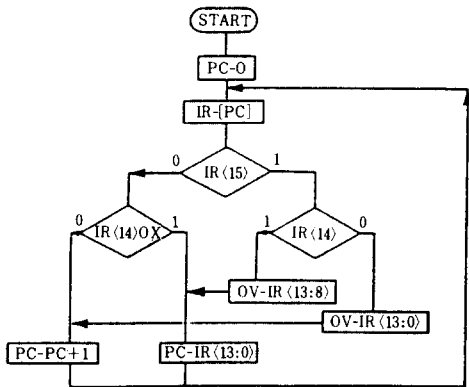


그림 3. BD 프로그래머블 제어기의 명령어 실행 플로우차트
 Fig. 3. Flow chart of execution of instruction in the BD programmable controller.

5 개의 블록을 각각 살펴보면 다음과 같다.

1) 입력 및 출력장치. 프로세스 상태를 감지하기 위해서 다수의 입력 변수가 사용되며 필드 출력은 OP 코우드에 따라서 병렬로 랫치내에 스트로브(strobe) 된다.

2) 프로그램 메모리·프로그램을 로우드(load)시킬 때 BD 프로그램 명령어가 RAM 안에 저장되며 이 기억장치는 OP 코우드의 명령어에 따라서 점프(Jump)하거나 필드출력으로 프로세스한다.

3) 제어장치·수동제어 스위치, 클럭 시스템, 프로그램 카운타 그리고 1비트 데이터 셀렉터가 포함된다. 프로그램 카운터의 내용은 제어기의 출력에 따라서 명령어에 증가(increment)하거나 프리세트(preset)한다.

4) 디스플레이 장치·프로그램을 로우드 시키거나 실행하는 중에 지정된 번지의 기억장소의 내용과 번지를 디스플레이 하기 위해서 LED를 사용한다.

2. PCU(Programmable Control Unit)의 구성

BD 프로그램은 MPU(Microprocessor Unit) 명령에 의해서 MPU의 주기억 장치에서 PCU의 명령어 기억장치로 전송되며 MPU는 동작 모드(operating mode)로 PCU의 제어를 실행한다.

PCU의 각 부분은 다음과 같다. (그림 4)

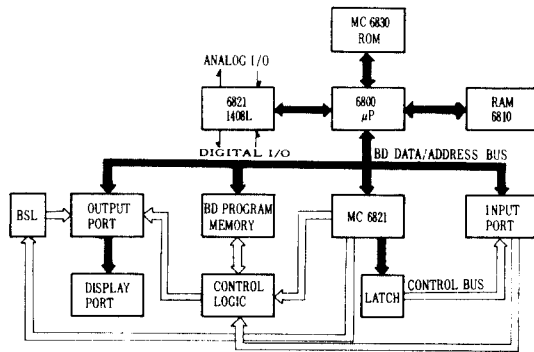


그림 4. BD/Micro 프로그래머블 제어기의 블록선도
 Fig. 4. Block diagram of BD/Micro programmable controller.

1) 입력장치·64개의 입력변수 IV(63:0)가 프로세스 입력, 피이드백(feedback)입력, 테스트 입력의 3가지로 구분된다.

2) 출력장치·뱅크선택논리(bank select logic)에 의해서 출력장치는 영향을 받는다. (그림 5)

3) 제어장치·8개의 명령어가 실행되며 각각 ISP 표기법으로 표시되며 명령어 형식은 그림 6과 같다.

4) BD-마이크로컴퓨터 인터페이스·16비트 데이터

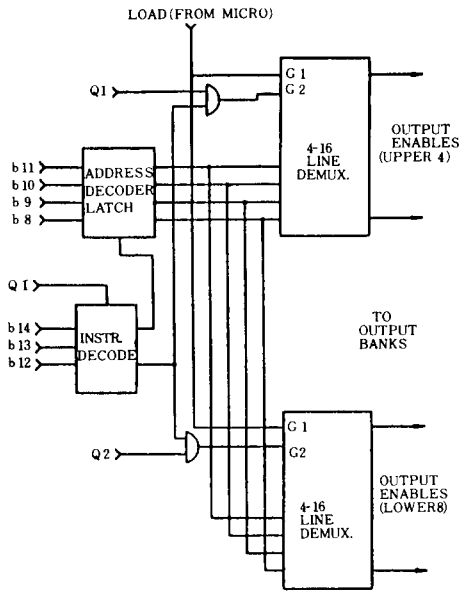


그림 5. 뱅크 선택로직의 구성
Fig. 5. Bank select logic.

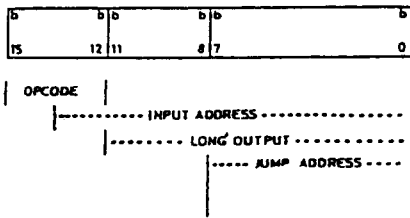


그림 6. BD/Micro프로그램머블 제어기의 명령어형식
Fig. 6. BD/Micro programmable controller instruction format.

버스와 8비트 콘트롤 버스는 PCU와 마이크로 컴퓨터 (μc)에 각각 접속되며 MC 6821 PIA에서 온 16비트 데이터 버스는 메모리에 접속된다. μc 내에 있는 8비트 제어용 레지스터는 3상태 (tri-state) 제어신호를 발생한다. 그림 7은 BD- μc 인터페이스를 보여 준다.¹¹²⁾

5) 뱅크 선택로직에 (bank select logic) · PCU의 출력명령어는 뱅크 선택로직에 의해서 12개 또는 4개의 출력래치 (output latch)에 영향을 준다.

3. PCU의 ISP 표기

ISP는 BNF(Backus Normal Form) 처럼 여러 컴퓨터나 제어기의 제작에서 하드웨어의 동작과 구조를 묘사해 주는 간결하고 일관된 문법(grammar)이 사용되기 때문에 BD 프로그래머블 제어기의 동작과 구조를 서술하는데 이용된다.⁵⁾ 하드웨어를 구성하는데 있어서 ISP 표기법은 다른 언어보다 응용 범위가 넓은 설계

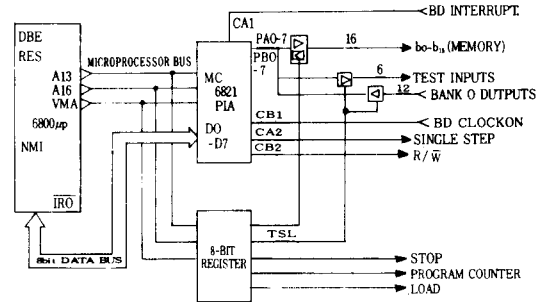


그림 7. BD/Micro프로그램머블 제어기의 데이터 및 콘트롤버스 구성도
Fig. 7. Scheme of data and control bus in BD/Micro programmable controller.

도구 (design tool)로서 하드웨어의 시뮬레이션 (simulation)과 제작 (architecture) 외에 소프트웨어의 구성, 프로그램의 확인, 시스템 제어 등에 널리 응용된다.

• ISP 표기법은

- (a) 정규명칭
- (b) 시스템의 규격 및 동작순서의 묘사 그리고
- (c) 별칭연산자 (alias operator)를 사용한 규정된 명칭의 분리 및 생략 등을 포함한다.

한 예로 $M_P[255:0] \langle 15:0 \rangle$ 은 명령어 기억장치 (instruction memory)를 나타내며 괄호 “[]”는 기억장소에 있어서 배열 (array)의 차원 (dimension)을 묘사한다. 즉 명령어 기억장치는 0에서 255로 한정된 256개의 기억장소 (memory location)를 의미하며 괄호 “ $\langle \rangle$ ”는 각 기억장소가 16비트를 갖는다는 것을 나타낸다.

- 프로그램 카운터/PC (7:0)
- 명령어 기억장치/MP[255:0] (15:0)
- 명령어/IR (15:0)
- OP 코우드/OP (3:0) : (15:12)
- 출력뱅크/OB (15:0) (11:0)
- 출력브랜치번지/OBA (3:0)
- 클럭인터럽트로직 (clock interrupt logic)/CIL (2:0)
- $IR15=0, IR \langle 14 \rangle \oplus \times = 1 \Rightarrow PC \leftarrow M_P[IR \langle 7:0 \rangle]$
- $IR \langle 15 \rangle = 0, IR \langle 14 \rangle \oplus \times = 0 \Rightarrow PC \leftarrow PC + 1$
- $OP \langle 3:0 \rangle = 1000 \Rightarrow PC = PC + 1, OV \langle 11:0 \rangle \quad IR \langle 11:0 \rangle$
- $OP \langle 3:0 \rangle = 1100 \Rightarrow PC \leftarrow M_P[IR \langle 7:0 \rangle], OV \langle 11:8 \rangle \quad IR \langle 11:8 \rangle$
- $OP \langle 3:0 \rangle = 1110 \Rightarrow PC \leftarrow M_P[IR \langle 7:0 \rangle], OBA \langle 3:0 \rangle : IR \langle 11:8 \rangle$

- $OP\langle 3:0 \rangle = 1010, ST\langle 0 \rangle = 1 \Rightarrow PC = PC + 1$
- $OP\langle 3:0 \rangle = 1010, ST\langle 0 \rangle = 0$
 $\Rightarrow CIL\langle 2:0 \rangle = IR\langle 15:13 \rangle$
- $OP\langle 3:0 \rangle = 1101$
 $\Rightarrow PC \leftarrow M_P[IR\langle 7:0 \rangle], OV\langle 13:12 \rangle$
 $\leftarrow IR\langle 13:12 \rangle$
- $OP\langle 3:0 \rangle = 1111 = PC \leftarrow M_P[IR\langle 7:0 \rangle]$

III. BD/Micro 프로그래머블 제어기의 소프트웨어 고찰

BD 프로그램 작성을 위하여 BD트리의 개념을 도입하여 다출력 변수를 실행시키는 트리의 최소화가 실행되어야 한다. 프로그램을 최적화시키는 방법은 Cenry¹¹⁾ 등의 의하여 연구되었으며 최근에 BD 프로그램에 의해서 최적화된 논리설계를 얻는 방법에 대하여 관심이 상당히 높아지고 있다.

본 논문에서는 상태도(state diagram)를 이용하여 최적 알고리즘을 작성한다. 소프트웨어 측면에서는 BD 프로그램은 IF-Then-Else의 명령어를 사용하므로써 하드웨어 적인 문제를 소프트웨어로 해결한다.

BD 프로그램을 작성하기 이전에 BD 트리의 개념을 도입하여 다중출력함수를 갖는 최소의 BD 트리는 가장 적은 수의 판정 노드(decision node)를 가진 함수를 실행하며 최적 알고리즘을 제공해 준다.¹²⁾

불완전 다중 출력함수를 이용한 커버링 트리(covering tree), Karnaugh도를 이용한 트리, 그리고 상태도를 이용한 트리의 합성방법에서 최적 BD 트리의 세 가지 조건 즉 포괄성(coverage), 분리성(separability) 및 적합성(compatibility)을 만족해야 하는데 임플리칸트(implicant)의 수가 많은 경우에 적합성의 확인은 용이하지 않으므로 레지듀얼(residual) 특성함수를 도입하여 최적 BD 트리의 조건을 만족시킨다.¹¹⁾

IV. 교통신호 제어 시스템의 구성과 운용

BD/Micro 프로그래머블 제어기의 제어기능 연구를 위하여 제어 대상으로서 교통신호 제어기로 이용되는 경우를 검토한다. 제어할 교차로는 남북방향의 간선도로와 동서방향의 지선도로로서 남북(N-S) 간선 도로는 동시에 동작하는 좌회전 표시(화살표: ARROW)를 갖는다.

1. 교통신호 제어 시스템의 개요

제어기는 두가지 모드(mode) 즉, 순차모드와 반응 모드 중에서 어느 하나만 동작하는데 순차모드에서 상태도의 시계 방향으로 제어기가 각각의 상태를 순차적 단계로 실행한다. 이 모드에서 제어장치는 교통량에

상관없이 계속 반복 실행된다.

교차로에서의 제어기 실행 상태는 순차모드이면 시계 방향의 상태로 진행하며 반응 모드이고 요구신호(REQUEST SIGNAL)가 있으면 어느 상태에서든지 화살표(ARROW) 방향을 따라 가장 짧은 경로를 거쳐 진행한다. 동시에 두개의 요구신호가 있을 때 순차 모드의 순서에 따라 요구(request)가 차례대로 수행한다.

많은 요구가 동시에 발생하면 제어기는 어떤 요구가 우선(priority)인가를 판단하고 선택해야 한다. 이러한 요구는 매 싸이클마다 한번씩 실행될 것이다. 이것은 막중한 교통량으로 인한 어떤 요구라도 최우선 방향으로 실행되도록 하기 위해서다.

2. 상태도(state diagram)의 합성

상태도¹⁶⁾는 어떤 문제와 일의 수행과정을 명확하게 해주는 한 가지 방법으로서 완전한 형태로 최대의 정보를 제시해 준다. 그림 8은 제어하고자 하는 교차로의 완전한 상태도로서 제어기의 상태를 묘사한다. 그림에서처럼 교통신호기는 8개의 상태를 갖는다. 어느 한 상태에서 다른 상태로 진행해 나가는 조건들은 그림에서 두개의 상태를 연결하는 선(line)에 표시된 것(label)과 같다.

상태의 우선권은 다음 방법으로 지정한다. 어느 한 상태에서 취할 수 있는 경로가 여러개 있다면 제어기는 가장 인접한 상태의 경로로 진행할 것이다.¹⁹⁾ 즉 상태도의 원(circle)외곽의 시계방향 경로로 계속 진행될 것이다.

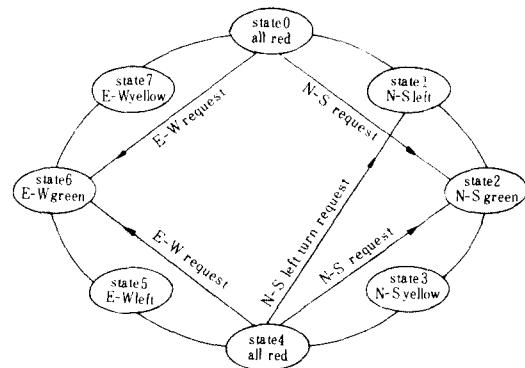


그림 8. 교통신호 제어기의 상태도
 Fig. 8. State diagram of traffic signal controller.

3. 제어기의 프로그램 작성 및 실행과정

간선 도로망의 교통제어에서 교통량에 따라서 적절하게 선택해야 할 제어신호 인자(control signal parameter)는 신호주기(cycle length), 분할비(offset) 등

이 있는데 이들 값은 교통량 감지기에 의해서 감지된 각 방향의 교통량에 따라 결정된다. 따라서 교통제어기로 BD 프로그래머블 제어기를 쓸 경우에 이 제어기는 이진수 판정에 의한 순차제어를 실행하고, 수차연산 처리능력은 μc 를 인터페이스시켜 교통량에 따른 제어신호인자의 변화를 주어야만 원활한 도로교통망의 신호체제를 가질 수 있다.

시물레이션에 필요한 외부회로는 3개의 신호등 타이머이고 이들 타이머는 3개의 BD 제어기 출력에 의해서 시작된다.

반면에 3개의 캐리아아웃(carry out) 입력은 타이밍(timing)의 끝남을 지시한다. 한 개의 입력 스위치는 순차모드나 반응모드를 선택하고 다른 2개의 스위치는 반응모드에서 동서, 좌회전의 요구의 것을 다룬다. 7개의 LED들은 두 방향에서 적색, 녹색, 황색 등과 녹색 좌측 방향 표시등을 시물레이션 한다. 그 외에 표 3에서 처럼 3개의 타이머 출력과 3개의 상태출력은 6개의 LED에서 표시된다. 표 1은 이 제어기의 입출력신호를 보여주며 표 2는 외부 타이머의 하드웨어 시스템 및 입출력 신호관계를 보여준다.

그림 8의 상태에서 8개의 상태마다 각각의 BD트리를 작성한 것이 그림 9에서 보여준다.

마이크로 컴퓨터 MET 6802 D2에는 3개의 PIA(Peripheral Interface Adapter) 중에서 한개는 입출력용 LED 디스플레이 및 16진 키보드 인터페이스로 쓰이며 두개는 사용자(USER) 용으로 할당되어 있다.

6821 PIA(U 20)의 절대번호는 다음과 같다.

- PORT A(PDRA/DDRA).....\$ 8004
- CONTROL REGISTER(CRA).....\$ 8005
- PORT B(PDRB/DDR).....\$ 8006
- CONTROL REGISTER(CRB).....\$ 8007

마이크로 컴퓨터의 메모리는 주로 \$ 2000번지부터 사용한다. 6821 PIA(U 20)의 PORT A의 절대번호는 \$ 8004로 이것을 입력 상태로 정의하여 스위치(SW-3)의 셋팅 상태를 읽어들이어 PORT B에 인터페이

표 1. 교통 신호 제어기에 대한 BD/Micro프로그램머블 제어기의 입출력

Table 1. BD/Micro programmable controller I/O for the traffic signal controller.

INPUTS		OUTPUTS	
SIGNAL DESCRIPTION		SIGNAL DESCRIPTION	
T_{NSO}		T_{NSI}	
T_{EWO}		T_{EWI}	
T_{YO}	TIMER OUTPUTS	T_{YI}	TIMER TRIGGERS
NSR	NORTH-SOUTH REQUEST	R_{NS}	
		Y_{NS}	
		G_{NS}	NORTH-SOUTH LAMPS
EWR	EAST WEST REQUEST	R_{EW}	
		Y_{EW}	
		G_{EW}	EAST-WEST LAMPS
MOD	"SMART"/"DUMB"MODE	S_0	
		S_1	
		S_2	AUTOMATA STATE DISPLAY

표 2. 교통신호 제어기의 외부타이머 구성

Table 2. External timer for the traffic signal controller.

DEVICE	FUNCTION	INPUT SIGNAL (TRIGGER)	OUTPUT SIGNAL (RESET/EXPIRED)
TIME 1	NORTH SOUTH GREEN	T_{NSI}	T_{NSO}
TIME 2	EAST-WEST GREEN AND LEFT ARROW	T_{EWI}	T_{EWO}
TIME 3	SAEETY RED AND YELLOW	T_{YI}	T_{YO}

스된 BD 제어기에 출력시켜 그림 10과 같이 BD 프로그램을 실행한다.¹¹⁾ BD 트리로부터 엔트리 순서별로 프로그램을 작성해 보면 그림 11과 같다.

V. 결과 고찰

1. ICU(Industrial Control Unit)와 BD/Micro 프

표 3. 교통신호 제어기의 출력실행

Table 3. Output implementation for the traffic signal controller.

SIGNAL	T_{NSI}	T_{EWI}	T_{YI}	S_2	S_1	S_0				
DATA BIT	D_{13}	D_{12}	D_{11}	D_{10}	D_9	D_8				
SIGNAL	R_{NS}	Y_{NS}	G_{NS}	R_{EW}	Y_{EW}	G_{EW}	G_{NSL}	G_{EWL}		
DATA BIT	D_7	D_6	D_5	D_{14}	D_3	D_2	D_1	D_0		
SIGNALS					T_{NSO}	T_{EWO}	T_{YO}	MOD	NSL	EW
ASSIGNED FIELD INPUTS (63-0)					34	33	32	0	1	2
MULTIPLEXER ADDRESS (D13-8)					100100	100011	100010	000000	000001	000010

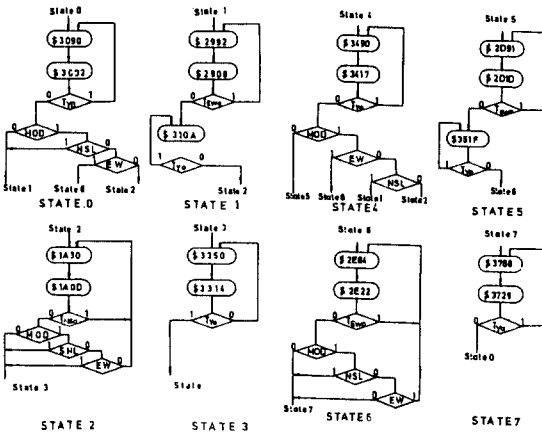


그림 9. 교통신호 제어기의 8 개 상태의 BD트리
Fig. 9. BD tree for 8 states of the traffic signal controller.

```

PAGE 001 TRAFFIC 1
00010 00001      NMM      TRAFFIC1
00020 00002      OPT
00030 00003      EQU      $B004   IPORT A
00040 00004      PIAC     EDU      $B005   ICRA
00050 00005      PIAB     EDU      $B006   IPORT B
00060 00006      PIBC     EDU      $B007   ICRB

00080 00008      ORG      $2000
00090 00009      2000     7F 8005   CLR      PIAC   ICRA2=0
00100 00010      2003     7F 8007   CLR      PIBC   ICRB2=0
00110 00011      2006     86 00     LDAA    #00   IPORT A=IN MODE
00120 00012      2008     87 8004   STAA   PIAB
00130 00013      200B     86 FF     LDAA    #FF   IPORT B=OUT MODE
00140 00014      200D     87 8006   STAA   PIAB
00150 00015      2010     86 04     LDAA    #04
00160 00016      2012     87 8005   STAA   PIAC
00170 00017      2015     87 8007   STAA   PIBC   ICRB2=1
00180 00018      2018     86 8004   INPUT  LDAA   PIAB   INPUT FROM SW1-3
00190 00019      201B     87 8006   STAA   PIAB   INPUT TO SDC
00200 00020      201E     20 FB 201B  BRA     INPUT
00210 00021      END

INPUT 2018 ,PIAA B004 PIAB 8006 PIAC 8005 PIBC 8007
    
```

그림 10. MPU의 BD 프로그램 실행
Fig. 10. BD program execution of MPU.

로그래머블 제어기의 프로그램 스텝의 비교

ICU(Industrial Control Unit)는 연속적인 비트 정보에서 판정을 필요로 하는 시스템에서 많이 사용되는 CMOS 프로세서이다.¹⁾ 그림 12는 실험에 사용된 ICU 교통신호 제어기의 프로그램 실행에 대한 플로우차트를 보여주며 표 4에서 라벨은 ICU 제어기의 입력을 나타낸다. 이 플로우차트에서 프로그램 스텝수를 그림 11에 나타난 바와 같이 BD 프로그램과 비교해 보면 다음과 같다.

구 분	BD (IMHZ)	ICU (IMHZ)
프로그램스텝수	42	134

이 제어기에 대한 결과 프로그램은 11개의 IF-THEN 구조로 되어 있으며 각각의 IF-THEN 구조는 그림 8에 나타난 화살표 방향(ARROW)이나 상태(state) 중

C	Address	Inputvariable	Truebranch	Falsebranch	
State 0	00	OUT:\$3090			
	01	OUT:\$3002			
	02	Tyo	00	03	
	03	MOD	04	06	
	04	NSL	06	05	
	05	EW	20	08	
	State 1	06	OUT:\$2992		
		07	OUT:\$2908		
	State 2	08	Tewo	06	09
		09	OUT:\$310A		
0A		Tyo	09	0E	
0B		OUT:\$1A30			
0C		OUT:\$1A0D			
0D		Tnsd	0B	0E	
0E		MOD	0F	12	
0F		NSL	12	10	
10		EW	12	11	
11		OUT:\$1A0D			
State 3	12	OUT:\$3350			
	13	OUT:\$3314			
State 4	14	Tyo	13	15	
	15	OUT:\$3490			
State 5	16	OUT:\$3417			
	17	Tyo	15	18	
	18	MOD	19	1B	
	19	EW	20	1A	
	1A	NSL	06	0B	
	1B	OUT:\$2D91			
	1C	OUT:\$2D1D			
	1D	Tewo	1B	1E	
	1E	OUT:\$331F			
	1F	Tyo	1E	20	
State 6	20	OUT:\$2E84			
	21	OUT:\$2E22			
	22	Tewo	20	23	
	23	MOD	24	27	
	24	NSL	27	25	
	25	EW	26	27	
	26	OUT:\$2E22			
	27	OUT:\$3788			
	28	OUT:\$3729			
	29	Tyo	27	00	

그림 11. 교통신호 제어기의 BD 프로그램
Fig. 11. BD program for the traffic signal controller.

표 4. 교통신호 제어기의 입출력
Table 4. Input/output listing of ICU traffic signal controller.

INPUT	Name	Function
0	RR	RESULT REGISTER
1	LR	LEFT TURN REQUEST
2	MOD	MODE SELECT
3	TMZ	CARRY OUT OF THE TIMER
4		FLAG BIT
5		FLAG BIT
6		FLAG BIT
7	NSR	NORTH-SOUTH GREEN REQUEST
8	EWR	EAST-WEST GREEN REQUEST.

에 하나를 실행한다.

2. BD/Micro P.C.와 기존 μ P형 P.C.의 비교

BD/Micro 프로그래머블 제어기와 기존 P.C.를 적용대상, 모듈화(modularity) 및 프로그래밍 용이성의 관점에서 비교해 보면 다음과 같다.

1) 적용대상

μ P를 채택한 PC는 병렬 프로세서 및 스캐레이트에 큰 비중을 두지 않는 산업용 제어 시스템에 많이 사용된다.

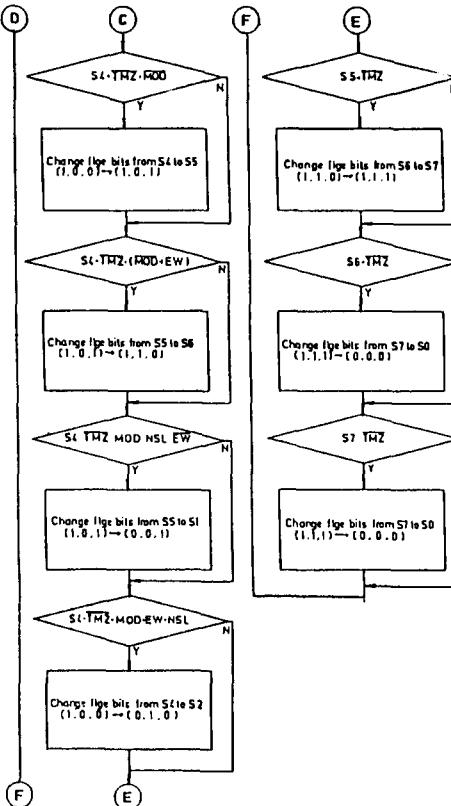
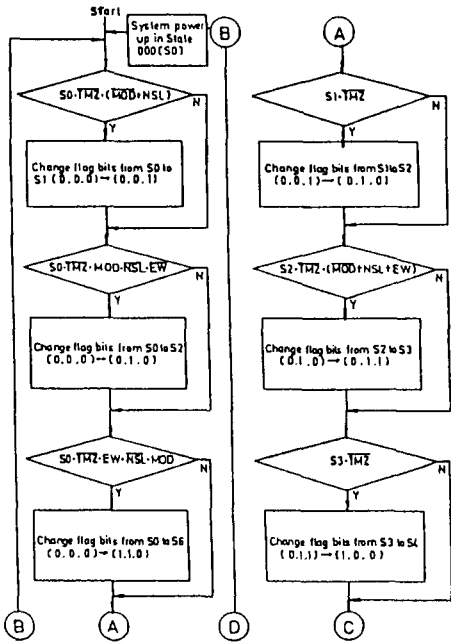
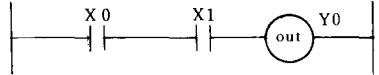


그림 12. ICU 교통 신호 제어기의 플로우차트
 Fig. 12. Flow chart of the ICU traffic controller.

멀티 프로그래밍(multi-programming) 방식을 채택한 BD PC는 병렬 프로세스 및 높은 스케일레트를 요하는 시스템에서 같은 조건의 μ P형 PC보다 효율적이다. 이를 아래의 간단한 래더 다이어그램을 통해서 실험하여 비교 고찰해 보기로 한다.

여기서 μ P형 PC의 소오스프로그램은 z-80 어셈블리 언어로 실행한다.



이 릴레이 래더 다이어그램을 이용한 μ P형 PC의 사용자 프로그램(user program)과 소오스프로그램(source program)의 실행시간은 다음과 같다.

ADDRESS	INSTRUCTION	EXECUTION CYCLE OF SOURCE PROGRAM
0	LD/STRT X 0	10/100 μ S
1	AND X 1	10/100 μ S
2	OUT X 2	10/100 μ S

총실행시간 30/300/ μ S

BD P. C.를 이용하여 위의 래더 다이어그램을 실행시키면 다음과 같다.

ADDRESS	INPUT STATE	TRUE BRANCH	FALSE BRANCH	EXECUTION CYCLE
0	X 0	1	ELSE	1 μ S
1	X 1	2	ELSE	1 μ S
2	OUTPUT YO	GOTO 0		1 μ S

총실행시간 3 μ S

위의 도표와 같이 실행사이클은 각각의 클럭펄스에 의해 수행되어 프로그램 스캐닝 시간이 줄게 된다.

2) 모듈화(modularity)

기존 P. C.의 모듈성은 I/O 모듈, 시퀀스(sequencer) 프로그래밍 로우더(programming loader) 그리고 타이머/카운터 모듈로 구분되며, 제어 대상이 필요로 하는 신호의 유형에 따라 모듈을 선택할 수 있어 시스템의 확장 및 변경이 용이한 잇점을 지니고 있다.¹¹⁾ 반면에 BD/Micro 프로그래머블 제어기는 BD 제어기 자체가 일종의 프로세스 역할을 하므로 시스템의 확장을 위해서는 BD 제어기를 여러개 인터페이스 시켜야 하는 단점이 있다.

3) 프로그래밍 용이성

μ P형 PC는 위의 릴레이 래더 다이어그램에서 프로그램 로우더(Program loader)를 이용하여 사용자(USER)가 쉽게 다음과 같은 순서로 프로그램 할 수 있는 잇점이 있다.

Address	Instruction	Comment
0	LD/STRT X0	starting logical operation
1	AND X1	logical and operation
2	OUT Y0	output operation

이에 비해 BD/Micro PC는 불완전 다중 출력함수^[1], P함수^[2] 및 상태도 등을 이용하여 그림 9 처럼 최적한 BD 트리(BD tree)를 작성하여 프로그래밍해야 하는 등 프로그래밍에 대한 지식이 요구된다.

3. 교통신호 제어기의 응용

여러 개의 교차로가 있는 지역에서 원활한 차량소통을 위해서는 인접교차로에 설치되어 있는 교통신호 제어기와 연계시켜야 한다.

교통신호 제어기의 연계에 있어서 연계의 기준시간은 한예로 그림13의 교차로 5의 청신호 시작신호를 기준으로 하여 계통속도를 변경시키지 않고 교통량에 따라 4 개의 신호주기와 5 개의 분할비를 설정하였으며 교차로의 유입교통량이 많은 방향에 맞추어 2 개의 지시 시간을 부여하였다. 분할비는 표 5와 같고, 교통량이 많은 방향을 유입방향으로 할때 (a)는 서쪽과 북쪽을 차량의 유입방향으로 한 지시 시간이고 (b)는 동쪽과 남쪽을 차량의 유입방향으로 한 경우의 지시시간이다. 도로망에서 각 교차로의 제어 인자들을 각 교차로의 특성에 맞도록 결정하면 도로망은 난조를 일으킬 염려가 있으므로 한 도로망 상에 있는 주요 교차로를 선택하며, 이를 기준으로 모든 교차로의 신호가 조화를 이루도록 한다.

기준이 되는 교차에서 μC 는 교통량에 따른 청신호시간과 지시시간을 결정하여 각 BD 프로그래머블 교통신호 제어기에 전달한다. 최적 교통제어 시스템설계

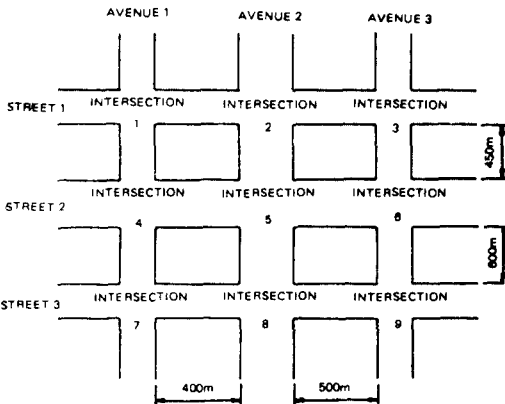


그림13. 도로교통망의 예
Fig. 13. The example of road network.

표 5. 교통신호 제어인자

Table 5. Traffic signal control parameter.

State	Street 1		Avenue 2		x
	Veh	Ped	Veh	Ped	
0*	R-2				1.7
1	A-20	DW-26			16.7
2	Y-4				3.4
3	G-30	**	R-62	DW-86	24.9
4	Y-4	Y-			3.3
5*					1.7
6			A-20		16.7
7	R-60	DW-74	Y-4		3.4
8			G-30	**	24.9
9			Y-4	DW-14	3.3

G:Green, Y:Yellow, R:Red
A:left turn arrow
W:Walk, DW:steady Don't walk
HDW:Flashing Don't walk
* :all-red interval
** :W-4, FDW-16

Cycle length 120(sec)
off set 3.6

(a) 신호주기와 분할비

(a) Cycle lengths and splits.

교 차 로	(a)	(b)
1	-63.5	61.4
2	-34.8	32.2
3	-0.6	57.5
4	-28.7	29.2
5	0	0
6	34.2	-34.7
7	12.5	-11.0
8	41.2	-40.2
9	75.4	-74.9

(sec)

(b) 제어기의 지시시간

(b) Offsets of each controller.

를 위하여 제어하고자 하는 지역의 교통흐름을 파악할 수 있는 교통흐름도, 교통량밀도, 경제적인 통로에 관한 연구가 이루어져, 가장 효율적인 교통망이 형성되도록 교통흐름을 유도할 수 있어야 하며 본 논문에서 제시한 BD/Micro P.C 교통신호제어기는 이와 같은 연구의 기반이 될 것으로 기대한다.

VI. 結 論

본 논문에서는 일반화된 μP 형 PC의 단점인 명령어 실행 싸이클 즉 프로그램 스캔타임(program scan time)의 문제점을 BD 알고리즘을 이용하여 펌웨어(Firmware)로 해결하였다.

아날로그 신호의 발생 및 데이터 추정등의 기능확장 및 원활한 도로 교통망의 신호체제가 되도록 교통량에 따른 제어 신호 인자의 변화를 주기 위한 BD P.C- μ 간의 인터페이스 장치(interface device) 등을 제작하였으며 PCU(Programmable Control Unit)의 오퍼레이팅 시스템(operating system)은 마이크로 컴퓨터의 메모리에 내장되도록 설계 제작하였다.

또한 교통신호 제어기의 제어 파라미터의 변화를 줄 수 있도록 독자적인 PCU를 인터페이스 시켰으며 더 나아가서 이 시스템이 μ 에 인터페이스 시킬 수 있는 PCU의 수 만큼의 교차로를 갖는 지역이나 간선도로에 확장시킬 수 있을 것으로 보고 앞으로는 제어 파라미터를 결정할 수 있는 교통망 이론과 실제의 도로망에서 적용해야 할 것이다.

BD/Micro PC의 앞으로의 과제를 살펴보면 산업용 시스템제어에서 적용할 수 있는 프로그래밍의 개발, MPU와 PCU의 모듈화의 개발 및 신뢰성을 높이기 위한 BD chip의 개발 등을 들 수 있다.

BD/Micro 프로그래머블 제어기는 단축된 논리경로를 갖기 때문에 그 응용 범위는 메모리 싸이클에 의해서 제한을 받는다. 이 제어기가 서보(servo) 및 로봇의 제어 시스템 응용 외에도 NC 선반의 제어, 대형 플랜트 설비의 순차 및 피이드백 제어(feed-back control), PID(Proportional Integrated Differential) 제어 등을 포함하는 여러 산업기기 응용 분야에서 광범위하게 활용될 것으로 기대된다.

参 考 文 献

- [1] E. Ceny, D. Mange, and E. Sanchez, "Synthesis of Minimal Demultiplexers, Multiplexers and Binary Decision Trees," *IEEE Trans. Computer*, vol. C-28, no. 7, July 1979, pp. 472-482.
- [2] R. Thayse, "P-Functions: A New Tool for the Analysis and Synthesis of Binary Programs," *IEEE Trans. Computer*, vol. c-30, no. 2, Feb. 1981, pp. 126-135.
- [3] E. Horowitz and A. Zorat, "Binary Tree as Interconnection Network," *IEEE Trans. Computer*, vol. c-30, no. 4, Apr. 1981, pp. 247-253.
- [4] R.T. Boute, "The Binary-Decision Machine as Programmable Controller," *Euromicro Newsletter*, vol. 1, no. 2, 1976, pp. 16-22.
- [5] R.L. Tabachnick, P.J. Zsombor-Murray, L. J. Vroomen, and T. Le-Ngoc, "Sequence Controllers with Standard Hardware and Custom Firmware," *IEEE Micro*, vol. 1, no.2, May 1981, pp. 9-25.
- [6] Charles H. Roth, Jr, "Fundamentals of Logic Design", West Publishing Co., 1976.
- [7] D. Siewiorck, "Introducing ISP," *Computer*, vol. 7, no. 12, Dec. 1974.
- [8] S.B. Akers, "Binary-Decision Diagrams," *IEEE Trans. Computers*, vol. c-27, no. 6, June 1978, pp. 509-516.
- [9] V. Gregory and B. Dellande, *MC 14500B Industrial Control Unit Handbook*, motorols SEMiconductor Products, Inc., Austin, Tex, 1977.
- [10] M6800 Microprocessor Application Manual, *Motorola Semiconductor Products, Inc.*, Austin, Tex., 1975.
- [11] R. Hudson, P.J. Zsombor-Murray and L.J. Vroomen. *Operating System for a Hybrid BD/ μ Programmable controller*. Proc. Twentieth ISMM Int'l Symp. On Mini and Microcomputers, July 1982, Cambridge, Mass., pp. 1-4.
- [12] Joseph D. Greenfield and William C. Wary, *Using μ ps and μ Cs the 6800 family*. John Wiley and Sons, 1981.
- [13] C.Y. Lee, *Representation of switching Circuits by Binary-Decision Programs*. Bell System, J., July, 1959, pp. 985-999.
- [14] B.K. Ledgerwood; *The Programmable Controller Makers Take a Look in the Mirror*. control Engineering, pp. 24-29, DEC, 1976.
- [15] Programmable controller, *Control and Instrumentation*, pp. 23-29, July 1981.