

論 文

GRAFCET로 記述된 順序 論理 시스템의 Microprogrammable Sequential Controller의 實現

正會員 禹 廣 俊* 正會員 李 範 薰**

Microprogrammable Sequential Controller Design of a Sequential Logic System Described by a GRAFCET

Kwang Joon WOO* , Bum Hoon LEE** *Regular Members*

要 約 기능이 향상된 microprogrammable Sequential Controller의 實現 방법을 제시한다. 고전적인 記述 방법을 개선한 GRAFCET로 부터 구성된 controller는 기존 programmable controller의 제약적인 처리속도, 메모리 용량, flexibility 및 프로그래밍의 용이성 등을 해결한다. 또한 제시된 controller는 기본 하드웨어의 구조는 변함없이 단지 microprogram의 변경 만으로써 새로운 시스템을 實現할 수 있다. 따라서 대규모의 입출력 변수를 갖는 industrial process 나 빠른 처리 속도를 요하는 power electronic converter 등의 controller 實現에 적합하다.

ABSTRACT The architecture and design of a highperformance general-purpose microprogrammable sequential controller are presented. Conventional sequential controllers are limited by speed, memory requirements, flexibility and programming ease. The proposed controller solves those limitations. It has a fixed hardware and is microprogrammable through firmware modification. It is especially designed for complex high-speed sequential controllers requiring large I/O capabilities, such as industrial process controller or power electronic conversion controller.

1. 序 論

順序 論理 回路는 例로써 자동 천공기의 제어를 위한 간단한 controller로부터 power electronic

converter나 industrial process의 제어를 위한 복잡한 controller에 이르기 까지 광범위하게 이용된다. 이러한 順序 論理 回路의 實現에는 flip-flop이나 multiplexer 혹은 decoder를 이용하는 hard-wired logic이 그 하나의 방법이고, ROM이나 PLA를 이용하는 microprogrammable controller가 또 다른 방법인 바 이는 前 방법에 비해 flexibility는 좋으나 근본적으로 하나의 ROM에 모든 제어 정보를 수록함으로써 시스템의 입출력수및

*, ** 檀國大學校理工大學電子工學科

Dept. of Electronic Engineering Dankook University,
Seoul 140, Korea.

論文番號 : 86 - 40 (接受 1986. 7. 22)

상태수가 증가하면 메모리의 크기에 제약을 받으며 또한 기본적으로 multiplexer에 의해 입력치 이 조건을 만족시킴으로서 입력수의 확장에 제약을 받는다¹⁾. 근자에는 제 3의 방법으로 마이크로프로세서나 PC (Programmable Controller) 를 이용하는 programmable sequential controller가 있는 바 이는 모든 정보를 직렬 처리 함으로써 처리 속도가 늦어 이러한 방법은 응답 속도가 늦은 industrial process의 제어에만 사용된다. 따라서 이러한 메모리의 크기나 처리 속도의 제약을 해결하기 위해 state diagram 혹은 flowchart로부터 상태의 천이에 관계되는 정보를 각각의 메모리에 분할 저장하여 이들 분할 메모리를 동시에 access 하는 회로를 부가하여 상기의 제약들을 극복하는 병렬 수행 방법이 제안되었다²⁾.

본 논문에서는 D.M. DIVAN et al⁽²⁾에 의해서 제안된 분할 메모리에 의한 병렬 수행 방법을 발전시켜 분할 메모리 수를 최소화 했으며, 입력천이 조건의 scan time 을 최소화 함으로써 처리 속도를 향상시켰다. 이와같은 분할 메모리에 의한 병렬 수행 기능은 program scan time 을 줄일 수 있고 또한 MOD-N counter, 레지스터 및 Adder에 의해 입력 천이 조건의 scan time 을 줄일 수 있는 구조는 기존 PC의 최대 약점인 처리 속도의 향상을 위한 하드웨어 구조의 새로운 한 방법을 제시한다. 또한 GRAFCET¹를 도입하여 sequential system의 하드웨어 구성을 표준화 즉 시스템의 변화시 하드웨어 구성의 변경없이 ROM 프로그램의 변경만으로써 실현 가능케 했으며 또한 ROM 프로그램을 GRAFCET로부터 용이하게 작성할 수 있게 하였다.

序論에 이어 2장에서는 GRAFCET의 기본적인 개념을 설명하며 3장에서는 메모리 분할 및 그 수행과 이를 기본으로 한 하드웨어 구성 및 마이크로프로그램의 작성을 보이며 이러한 구성이 subroutine의 처리도 용이하게 할 수 있음을 보인다. 또한 實例를 들어 제안된 방법의 實現 타당성을 보였으며 4장에서는 結論 및 앞으로의 연구과제를 논한다.

2. GRAFCET의 기본 개념

기본적인 定義 및 實例로써 GRAFCET의 기본 개념을 설명한다. 이에 관한 구체적인 연구는 참고문헌 (3~9)에서 찾아볼 수 있다.

2-1 定 義

GRAFCET란 sequential system의 서로 다른 action을 그래픽적으로 記述한 functional diagram으로서 step(정방향으로 표시), transition(bar로 표시)과 이들을 연결시키는 방향성 arc로써 구성된다. (그림 1). 이러한 그래프가 sequential system을 記述할 수 있도록 step은 수행할 일련의 action에 관계되며 transition은 다음 step으로의 evolution을 가능케 하는 logic condition에 관계된다.

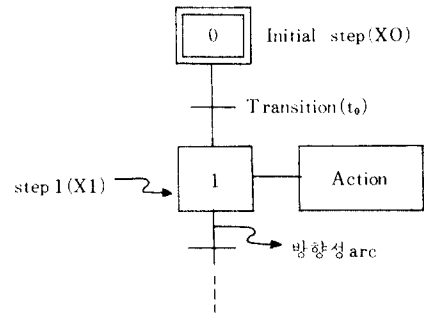


그림 1 GRAFCET의 구조
Structure of GRAFCET.

sequential system을 記述한 GRAFCET는 다음과 같은 evolution規則을 따른다. - Initialization은 시스템 최초의 action을 명시한 active step으로서 二重 정방향으로 표시한다. (그림 1 참조) - Transition은 直前の step들이 active되면 有効해진다. Transition이 有効하고 그리고 이 transition에 관계된 logic condition이 眞이 되면 다음 step으로 천이 된다. - 다음 step으로 천이된 結果로 해당 transition 直前の step들이 Inactivation 되고 直後の step들이 activation된다. - 동시에 다음 step으로 천이가 가능한 多數의 transition들은 同

1. GRAFCET : Le Graph de Commande Etape-Transition (Step 및 Transition으로 구성된 제어그래프)

時에 다음 step으로 천이된다. -동일한 step 이 activation과 inactivation이 同時に 이루어 질 경우 이 step은 activation되어 머문다. 이와같이 sequential system을 GRAFCET로 記述할 시에는 하나의 sequence (unique sequence)만으로 記述되는 경우, logic condition에 따라 각각의 sequence를 追求하는 선택 sequence를 포함하는 경우, 그리고 同時に 2개 이상의 sequence를 追求하는 並列 sequence를 포함하는 경우가 있다. 특히 並列 sequence는 독립적인 subsystem으로 분해할 수 있는 모든 system에 대해 항상 존재하며 이렇게 並列 sequence로 記述함으로써 system의 효율성을 높인다.

2-2 實 例

다음과 같은 仕樣을 갖는 脫脂槽를 GRAFCET에 의해 記述한다.

(1) 仕 樣

그림 2는 제품 표면에 있는 불순물을 제거하기 위한 시스템으로 불순물이 있는 제품은 탈지조에서 30초 동안 담금으로써 탈지한 후 하역 장소로 옮겨지며, 불순물이 없는 제품은 탈지 과정없이 단순히 하역 장소로 옮겨지고, 공급 장소에 제품을 담을 용기가 없을 경우에는 하역 장소에서 operator의 call 명령과 함께 용기를 크레인에 달

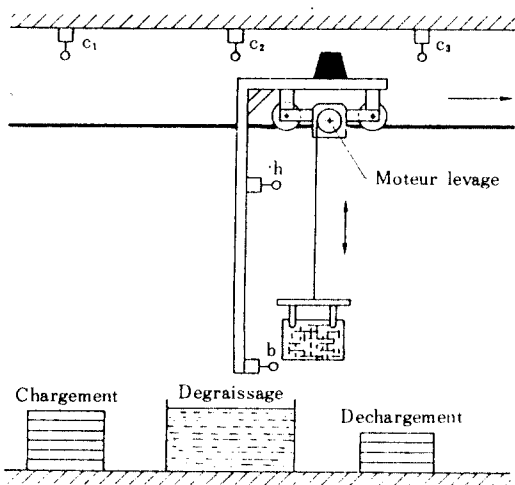


그림 2 탈지조 Washing pot.

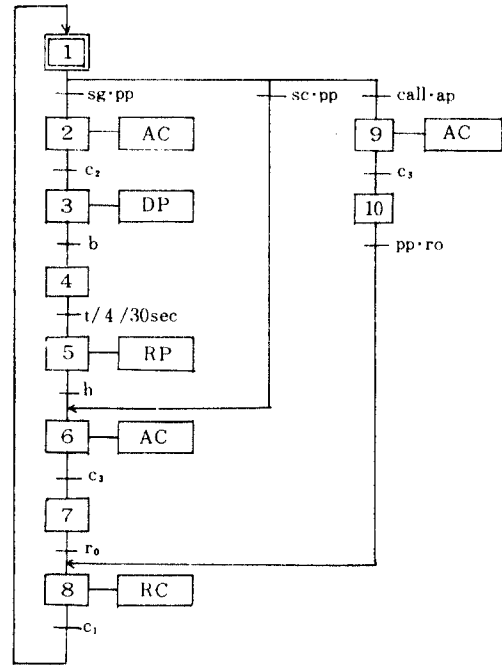


그림 3 탈지조의 GRAFCET GRAFCET of washing-pot.

아 공급 장소로 보낸다. 편의상 용기에 제품의 적재와 하역은 operator에 의해 수동으로 행하여지며 제품의 불순물 포함 여부도 역시 공급 장소에서 operator에 의해 구별된다.

(2) Functional GRAFCET의 구성

前述한 仕樣을 갖는 시스템을 GRAFCET로 記述하면 그림 3과 같으며 여기서 사용한 actions 및 input transition condition의 심볼 및 意味는 표 1과 같다.

표 1 Actions과 input condition의 심볼 및 意味 Natures and symbols of actions and input conditions.

Actions

AC	크레인 右로 이동	Motor
RC	크레인 左로 이동	"
DP	용기 하강	"
RP	용기 상승	"

Input conditions

sg	불순물 있는 제품 sequence 시작	operator
sc	불순물 없는 제품 sequence 시작	"
pp	容器 있음	"
ap	容器 없음	"
C ₁	위치 1	contact sw.
C ₂	위치 2	"
C ₃	위치 3	"
b	위치 b	"
h	위치 h	"
r ₀	복귀명령	operator
call	호출	"
t/4/ 30sec	step 4 (탈지) 시작후 30초 경과	counter

3. Microprogrammable Sequential controller의 구성

序論에서 언급한 바와 같이 programmable sequential controller는 -프로그램의 용이성- 처리 속도의 향상 -시스템의 확장성, 즉 多數(~수천)의 입출력 및 상태변수를 記述하고 實現할 수 있는 구조와 제어대상 system의 기능 변경시 controller의 순쉬운 변형등이 요구된다. GRAFCET로 記述된 sequential system의 기본 기능은 여러 step들 사이의 천이에 관계된 것이다. 이러한 천이는 입력 천이 조건 및 천이될 step을 정의함으로써 이루어지며 천이될 step에서 해당 output를 갖는다. 따라서 하나의 step에 관련되는 이상의 정보를 각개의 메모리(ROM)에 분할하며 이들 정보가 저장된 sequence를 추구해 갈 수 있도록 combinational circuit를 부가하여 실현한다. combinational circuit는 최초의 메모리만을 사용하도록 하고 또한 처리속도를 향상 시킬 수 있도록 구성한다. 또한 이상의 실현 방법을 GRAFCET로 부터 ROM program (microprogram)을 용이하게 작성할 수 있으며 제어 대상의 변경시 controller의 변형은 별도의 하드웨어 부가없이

microprogram의 변경에 의해서만 가능하도록 한다.

3-1 메모리 분할 기법

sequential system을 메모리에 의한 실현시 메모리의 효율적인 이용을 위한 데이터 구성 format이 필요하며 이러한 데이터는 CPU 설계시 이용되는 microinstruction 구성의 field 개념을 이용한다. 따라서 GRAFCET로 記述된 sequential system을 ROM에 의한 實現시 step을 중심으로 이에 필요한 데이터 format를 구성한다. 즉 active step에 관련된 정보를 다루기 위해서는 메모리를 다음과 같이 3개의 독립된 메모리 즉 block으로 분할한다. (i) 한 step에서 다음 step으로의 천이될 위하여 유효한 입력 천이 조건을 결정하기 위한 input transition condition block (ii) 입력천이 조건이 지성 됐을 시 천이될 다음 step를 지정해 주기 위한 next address block (iii) active step에 대한 action을 지정해 주기 위한 output block을 정의한다. 이상에서 정의된 분할 메모리를 GRAFCET와 대응시켜 보면 그림 4와 같다.

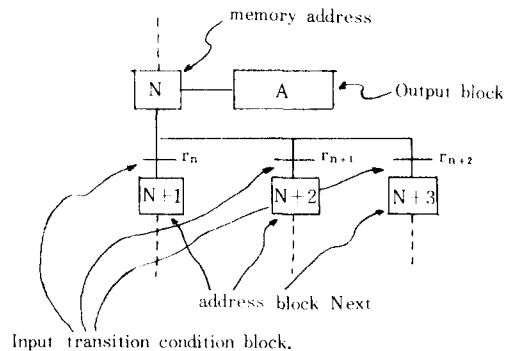


그림 4 GRAFCET와 분할메모리의 관계 relationship between GRAFCET and memory system.

그림 5는 메모리를 field로 분할한 경우 직렬 수행을 위한 실행 흐름도를 보였다. 하나의 active step에 관련된 모든 데이터 세트가 한 메모리 내에 있다면 이들의 정보를 처리하기 위해서는 최소한 3개의 메모리 access cycle이 요구된다. 그림 6은 동시에 Access할 수 있는 3개의 독립된 분할 메모리 내에 각각의 데이터 세트가

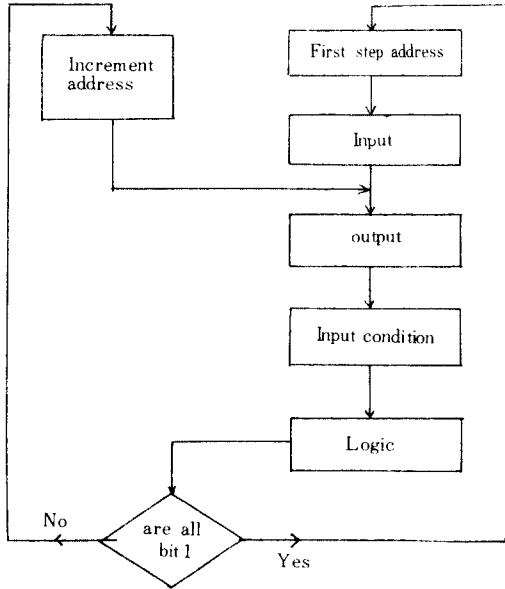


그림 5 메모리를 field로 분할한 상태의 직렬수행 flowchart for serial execution.

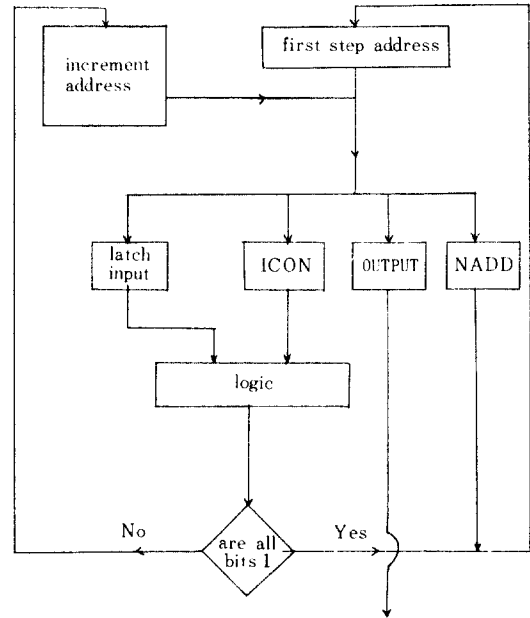


그림 6 독립된 분할메모리에 의한 병렬수행 flowchart for parallel execution.

있을 경우의 실행흐름도를 보였다. 이와같이 병렬 수행인 경우 처리 속도는 단지 1개의 메모리 access cycle에 의해 한정된다.

3-2 Microsequencer의 하드웨어 구성

State Graph⁽¹⁰⁾에서는 어느 한 시점에서 하나의 step만이 active되므로 이러한 active step은 sequential system에서 현재 상태에 대응되며 이와 같이 sequential system을 記述한 state graph를 ROM에 의한 實現시 그 하드웨어 구조는: (i) § 3-1에서 서술한 바와 같이 공통 address 버스를 중심으로 3개의 독립된 ROM 즉 입력 천이 조건을 감지하기 위한 input transition condition ROM (ICON ROM), next step의 address를 지정하는 next address ROM (NADD ROM) 및 active step의 action을 나타내는 microaction ROM (OUTPUT ROM) (ii) active step의 address를 나타내는 parallel-load counter로써 이 counter는 active step의 address로부터 scan에 의해 next step으로의 천이 조건인 입력 천이 조건을 감지할때 까지 scan을 행한다. 이와같이 1개의 counter로써 scan에

의해 입력 천이조건을 만족시켜 address를 지정할 경우 2^n (n 은 counter bit數) 만큼의 scan time을 요한다⁽²⁾. 따라서 scan time을 줄이기 위해서는 순 시스템 내에서 임의의 한 step이 갖는 최대의 선택 sequence數 N 만큼을 count할 수 있는 MOD- N counter와 active step의 address를 나타내는 레지스터 및 이 두값의 adder로써 구성할 경우 scan time을 최소화 할 수 있다. (iii) step의 천이를 위한 입력 천이조건을 감지하는 logic circuit로서의 comparator (iv) control system內的 정보 교환을 동기시키기 위한 AND gates들로 구성되며 이상의 element들로 구성된 하드웨어 구성을 그림 7에 나타낸다.

3-3 Microprogram의 作成

§ 3-1의 그림 4와 같이 GRAFCET의 step number를 memory address와 一致시키면 이 address가 지정됐을 때 즉 step N 이 active되면 OUTPUT ROM은 해당하는 action을 지정하며 NADD ROM은 next step의 번지 $N+1$ 를 레지스터로 출력시킨다. 또한 동시에 ICON ROM은 next

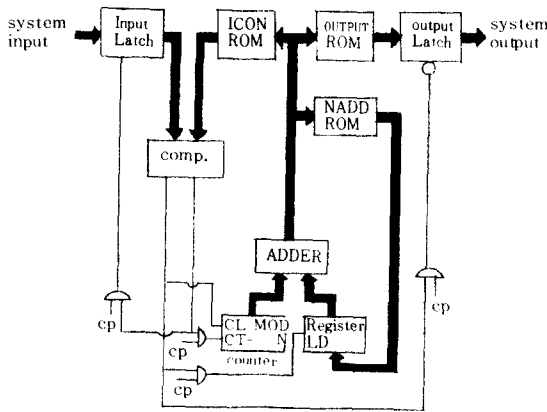


그림 7 microsequencer의 hardware 구조
hardware organization of microsequencer.

step으로의 입력 천이조건값을 comparator로 출력시킨다. 이때 input latch값이 ICON값과 일치하면 레지스터는 NADD의 값을 Adder로 보내며 (이 때 MOD-N counter는 clear 된다) 이 값으로 각 ROM의 번지를 지정한다. 만일 input latch값이 ICON값과 일치하지 않으면 MOD-N counter는 이 두값 즉 input latch값과 ICON이 일치 할 때까지 scan을 행한다. 이상에서와 같이 ROM의 address number는 step number와 일치하므로 step number를 정하기 위한 규칙은; 선택 sequence 들의 step number는 올림차순으로 하여야 하며 이들중 최초 step number는 前 step number보다 최소한 선택 sequence 數 만큼은 커야 한다. 상기와 같은 규칙에 따라 그림 4의 step number를 다시 쓰면 그림 8과 같다.

Microprogram의 작성 例로써 2장에서 實例로 주어진 脱筋操에 대한 GRAFCET(그림 3)에 대해서 실현 microprogram을 작성한다. 실현 하드웨어 구조는 §3-2의 그림 7에 주어졌으며 여기서 입력 천이 조건은 encoder를 통해 input latch에 입력된다. 따라서 주어진 시스템을 실현하기 위해서는 각각의 ROM (ICON ROM, NADD ROM 및 OUTPUT ROM) 프로그램만을 작성하면 된다. ROM 크기는 입력 천이조건이 11개이므로 2개의 8-3 encoder가 필요하므로 ICON ROM의 1 word는 4 bits (1개 encoder 출력값을 위한 3 bits와 2개 encoder를 구별키 위한 1 bit)를 갖

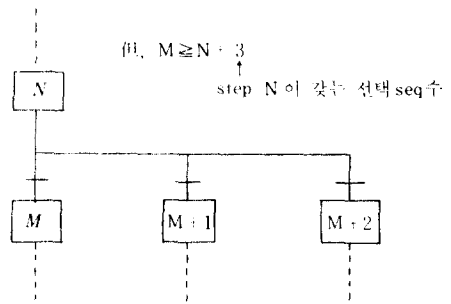


그림 8 그림 4를 새로이 step-numbering 한 GRAFCET
GRAFCET with renumbered steps.

는다. (입력수의 확장은 例로써 8-3 encoder를 기본 element로 할 경우 이를 하나의 port로 취급하여 사용된 ports를 구별하는 bits만을 추가하면 되므로 입력수의 확장이 용이하다. 이때 ICON ROM bits 구성은 기본 단위의 encoder 출력값 및 port selection 값으로 구성된다). 또한 step 數가 12개(가상 step 2개 포함: 그림 9 참조)이므로 NADD ROM의 1 word는 4 bits를 갖는

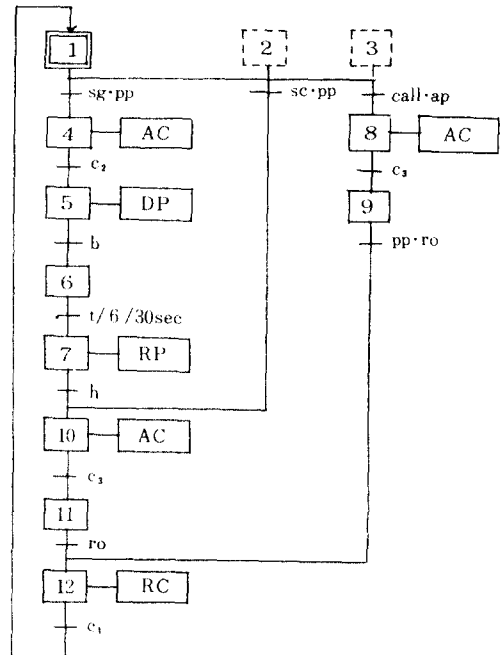


그림 9 그림 3을 새로이 step-numbering 한 GRAFCET
GRAFCET with renumbered steps
但, step 2, 3은 microprogram 작성을 알기 쉽게 하기 위한 가상 steps임

memory address (Reg. 값+MOD-N 값)	ICON ROM	NADD ROM	OUTPUT ROM
1	sg · pp	step 4	no action
2	sc · pp	" 10	"
3	call · pp	" 8	"
4	C ₂	" 5	AC
5	b	" 6	DP
6	t/4/30sec	" 7	no action
7	h	" 10	RP
8	C ₃	" 9	AC
9'	pp · ro	" 12	no action
10	C ₁	" 11	AC
11	ro	" 12	no action
12	C ₁	" 1	RC

그림10 심볼로 주어진 ROM program
Symbolic ROM program.

다. 또한 output 는 4 개이므로 output ROM 의 1 word는 4 bits를 가지며, 각 ROM 의 용량은 s-step 數인 12words가 필요하다. 또한 선택 sequen- ce 數는 step 1 에서 3 개를 가지므로 MOD-3 counter (count sequence는 0, 1, 2) 를 사용한다. microprogram을 작성하기 위해 §3-3에서 논한 바와 같이 memory address를 step number와 일치시키므로 이들 step number는 §3-3에서 논한 step numbering 규칙에 따라서 그림 9와 같이 step number를 다시 주어야 한다.

최초 step 1 이 active되며 input latch 값이 IC-ON (sg · pp) 에 일치하면 comparator로부터 레지스터의 load 신호에 따라 NADD의 값인 step 4로 천이되며 output AC를 출력시키나, 일치하지 않으면 MOD-3 counter는 CP에 따라 1이 증가되어 ROM address 값 2를 지정하며 이때 input latch 값이 ICON (sc, pp) 에 일치하면 next step 10으로 천이 되고 그렇지 않으면 MOD-3 counter는 다음 CP에 따라 1이 증가되어 ROM address 값 3을 지정하며 input latch 값과 해당 입력 천이조건과 일치할때 까지 input scan을 행한다. 이와같은 방법으로 step 간에 천이가 이루어

memory address	ICON ROM	NADD ROM	OUTPUT ROM
0001	0000	0100	0000
0010	0001	1010	0000
0011	0010	1000	0000
0100	0011	0101	0001
0101	0100	0110	0010
0110	0101	0111	0000
0111	0110	1010	1000
1000	0111	1001	0001
1001	1000	1100	0000
1010	0111	1011	0001
1011	1010	1100	0000
1100	1011	0001	0100

그림11 microcode化된 ROM program
Microcoded ROM program.

但-입력천이조건은 그림10의 순서대로 2개의 8-3 encoder 입력 pin 0로 부터 연결되며 ICON ROM의 MSB는 2개의 encoder를 구별하기 위한 bit인.

-OUTPUT ROM의 bit 0 (LSB)는 action AC, bit 1은 action DP, bit 2는 action RC, bit 3는 action RP에 각각 연결됨.

지며, 그림 9에 대한 각 ROM의 microprogram은 그림10의 과정을 거쳐 그림11과 같이 된다.

§3-2에서 제시한 microprogrammable controller가 하드웨어의 부가없이 단지 microprogram에 의해서만 sub-routine을 처리할 수 있음을 그림12에 주어진 例題로써 설명한다.

Subroutine을 포함한 sequence를 처리 할 때는 return address에 관계된 stack 및 stack pointer와의 논리적인 처리를 위한 하드웨어가 필요하나 여기서는 §3-2에서 주어진 기본적인 하드웨어는 변화없이 단지 microprogram에 의해서만 처리할 수 있음을 보인다. 즉 subroutine의 initial step은 main-routine에서 사용된 횟수만큼 그림 13과 같이 sub-routine에 가상 선택 sequence를 갖는다.

따라서 그림13에서 step11이 active되고 입력 천이 조건 t₁₁이 만족되면 ROM address 100으로부터 시작되는 sub-routine이 시작된다. 이후 sub-routine 마지막 step107이 active되고 입력 천이 조건 t₁₀₃이 만족되면 sub-routine 최초 step100

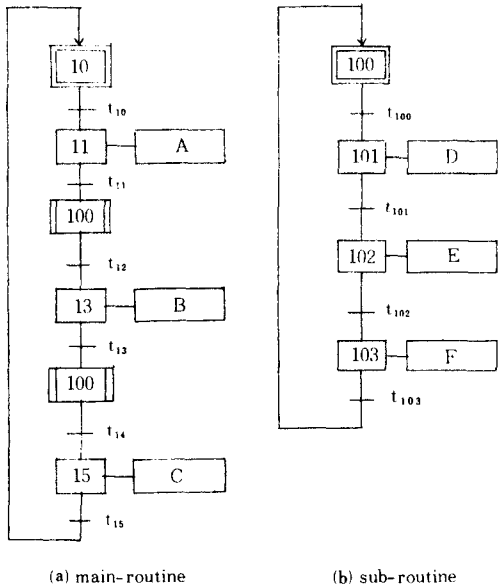


그림12 Sub-routine 을 포함한 GRAFCET 의 예
Example of GRAFCET having sub-routine.

(step 100은 no action임)에서 입력 천이 조건을 scan 하며 입력 천이 조건 t_{12} 가 만족되면 (이때 memory address는 scan에 의해 101을 지정한다) main routine의 step 104로 천이된다. 두번째 sub-routine도 첫번째와 같이 처리되며 이에 대한 microprogram은 그림14와 같이 된다.

메모리Address	ICON ROM	NADD ROM	OUTPUT ROM
10	10 (t_{10})	11	00 (no action)
11	11 (t_{11})	100	0A
104	13 (t_{13})	100	0B
105	15 (t_{15})	10	0C
100	100 (t_{100})	103	00 (no action)
101	12 (t_{12})	104	00 (no action)
102	14 (t_{14})	105	00 (no action)
103	101 (t_{101})	106	0D
106	102 (t_{102})	107	0E
107	103 (t_{103})	100	0F

그림14 Sub-routine 을 포함할 시 심볼로 표시된 ROM program
Symbolic ROM program with sub-routine.

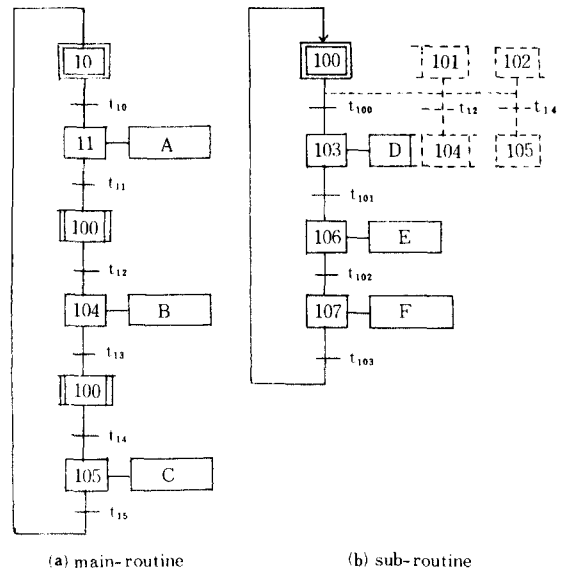


그림13 그림 10을 새로이 step numbering한 GRAFCET
GRAFCET with renumbered steps.
 - step 101, 102는 microprogramming을 알기쉽게 하기 위한 가상 steps.
 - step 104, 105는 가상 선택 sequence을 나타냄

4. 結 論

본 연구에서는 Sequential system의 design 에 요구되는 사양을 알기쉽고 간편하게 記述할 수 있는 GRAFCET를 이용하여 flexibility가 좋은 ROM을 사용한 microprogrammable sequential controller의 實現 方法을 提示했다. 提示된 實現 方法은 -입출력수를 용이하게 확장할 수 있으며, -입력천이 조건의 scan time을 줄일 수 있으며, -사용되는 ROM의 數가 최소화되며 -microprogram이 GRAFCET로 부터 직접 작성되며, -subroutine을 처리 하는데 별도의 하드웨어를 부가하지 않고 가상 선택 sequence의 개념을 이용하며 program에 의해서만 처리 가능함을 보였다. 특히 분할 메모리 기법에 의해 병렬 수행이 되어 program scan-time을 줄일 수 있고 또한 MOD-N counter와 레지스터에 의한 하드웨어구조는 입력 scan-time을 줄일 수 있다. 위에 제시된 방법으로 기존 PC의 최대 약점인 처리 속도를 향상시킬 수 있는 PC의 processor 구성방법

을 제시하였다. 따라서 제시된 방법은 대규모의 입출력 변수를 갖는 industrial process나 빠른 처리 속도를 요하는 power electronic converter 등의 controller 실현에 적합하다. 앞으로 병렬 sequence를 처리하는데 $W_{00}^{(0)}$ 에 의해 제시된 state-graph로 분할 후實現하는 방법 외에 본 논문에서 제시된 입력 scan time을 최소로 하는實現方法을 기본으로 병렬 sequence를 처리할 수 있다면 공간적으로 분리할 수 없는 병렬 sequence를 갖는 시스템에 대해서 대단히 효율적인 controller를 구성할 수 있으리라 기대된다.

參 考 文 獻

(1) W.I. Fletcher "An Engineering approach to digital design" Prentice-Hall, 1980.
 (2) D.M. Divan et al "Microprogrammable Sequential Controller" IEEE Proceedings, vol. 131 Pt. E. No 6. Nov. 1984.
 (3) B. Taconet et B. challot, programmation du GRAFCET

sur automate a langage Logique, a Relais ou Booleen, Le Nouvel Automatisme, pp. 44~45, Jan-Feb, 1979.
 (4) J.P. Cocquerez et J. Devars, "synthese d'automatismes sequentiels a laide de circuits logiques programmables," L'onde Electrique vol 59. no 4. pp 72~78, 1979.
 (5) E. Daclin et M. Blanchard, synthese des systemes Logiques. cepadues-Edition Toulouse. 1976.
 (6) G. Michel, C. Laurgeau et B. Espiau, Les Automates programmables Industriels. Dunod technique Paris, 1979.
 (7) S. Thelliez et J. M Toulotte, GRAFCET et Logique Industrielle programme. Eyrolles, Paris. 1982.
 (8) S. Thelliez et J.M Toulotte, Applications Industrielles du GRAFCET. Eyrolles. Paris. 1983.
 (9) J. C. Bossy, P. Brard, P. Faugere et C. Merlaud, Le GRAFCET a Pratique et Ses Applications Educalivre Paris, 1979.
 (10) K. J. Woo "Race-Free programmable synthesis of a Sequential system Described by a GRAFCET" 대한전자공학회지 Nov. 1984.
 (11) PLC-2/20 Programmable controller: programming and operation manual, ALLEN-BRADLEY.
 (12) J. H. Christewsen, O. J Struger" programmable controller software Architectures for advanced machine Diagnostics" IEEE Trans. Industry Application, vol, IA-21. no. 1 Jan/Feb. 1985.



禹 廣 俊(Kwang Joon Woo) 正會員
 1946年11月8日生
 1967. 3~1974. 2 : 漢陽大學校電子工學科(工學士)
 1975. 3~1977. 2 : 漢陽大學校大學院電子工學科(工學碩士)
 1977. 10~1980. 9 : Université Louis Pasteur de Strasbourg (D. E. A)
 1980. 10~1983. 1 : Institut National Polytechnique de Grenoble (Docteur Ingénieur)
 1983. 3~現在 : 檀國大學校 電子工學科(助教授)
 (Dept. of Electronic Engineering Dankook University)



李 範 薰(Bum Hoon Lee) 正會員
 1958年6月1日生
 1978. 3~1984. 8 : 檀國大學校 電子工學科(工學士)
 1984. 9~1986. 8 : 檀國大學校 大學院電子工學科(工學碩士)
 1986. 9~現在 : North Carolina 주립대학 在學中