

Microprocessor에 의한 자동창고의 入出庫 및 搬送

시스템의 Sequential Controller 설계

(Design of the Sequential Controller of Warehousing -Delivery and Unmanned Transportation System for Automated Warehouse System by Microprocessor)

朴 鍾 元*, 崔 成 勇*, 禹 廣 俊*

(Jong Won Park, Sung Yong Choi and Kwang Jun Woo)

要 約

本 論文은 자동창고의 入出庫 시스템을 설계하는데 있어 기존의 시스템에 비해 입출고 기능을 강화하였고, 시스템의 효율을 높이기 위하여 入出庫 시스템과 2대의 無人 搬送車를 이용하는 搬送 시스템을 계층적으로 구성하여 마이크로프로세서에 의하여 실현하였다.

시스템을 記述하는데 전체 시스템을 각 subsystem으로의 분할記述이 용이하고 각 subsystem간의 동기까지 쉽게 記述할 수 있는 GRAFCET를 이용하였으며, 記述된 전체 시스템의 마이크로프로세서에 의한 실험은 program module 개념을 정의하여 기존 마이크로프로세서(Z80) 명령어로 쉽게 프로그램을 작성할 수 있음을 보였다.

Abstract

This paper realizes the design of the sequential controller of a reinforced warehousing-delivery and unmanned transportation system for automated warehouse system. The system is composed of ware housing delivery and transportation system using two unmanned vehicles with hierachical structure. It is described by GRAFCET and realized by programmed logic with microprocessor.

Being described by GRAFCET, the system is able to divide it into subsystems and to synchronize them. Defining the concept of program module, one can easily program the system with the microprocessor instruction language.

I. 序 論

컴퓨터를 이용하여 창고에 관련된 시스템을 자동화 한 자동창고 시스템은 物量을 취급하는데 있어서 인력

의 절감, free location기능에 의한 보관효율의 향상, 선입선출기능에 의한 최적의 在庫관리, 流通구조의 合理化, 관리수준의 향상, 창고의 무인화 등 많은 효과를 가져오는 시스템으로서 최근의 공장 자동화 및 무인화 공장의 實現에 있어 필히 實現되어야 할 부분이다. 특히 자동창고 시스템의 入出庫 시스템 및 搬送 시스템은 物量의 흐름을 제어하는데 중요한 역할을 한다.^{1,2,3)}

*正會員, 檀國大學校 電子工學科

(Dept. of Elec. Eng., Dankook Univ.)

接受日字: 1986年 10月 13日

따라서 本 論文에서는 취급物量의 증가에 대해 보다 빠른 시간내에 入出庫 및 搬送업무를 처리할 수 있도록 出庫區를 자동창고 양측에 2 개로 확장하고 stacker crane (S. C) 이동시 주행이동(x축이동) 및 승강이동(y축이동)을 동시에 행하여 S. C의 이동거리를 최소로 하였으며, S. C의 1 회이동에 2 개의 pallet를 서로 다른 rack에 入出庫할 수 있도록 설계하고, 여기에 搬送 시스템을 연결하여 2 대의 無人 搬送車를 이용 물량의 흐름을 능동적으로 처리할 수 있도록 설계하였다.

이러한 sequential system의 설계는 원하는 사양을 일정한 형식을 통하여 記述한 후 記述된 언어 혹은 graph로 부터 다양한 實現과정을 거쳐 설계된다. 특히 産業 자동화에서와 같이 多數(수십~수천)의 入출력변수 및 상태변수가 요구되는 시스템을 설계하는데는;

-원하는 기능을 이해하기 쉽고 명확 간결하게 記述할 수 있어야 하며(記述의 용이성)

-記述된 결과로 부터 손쉽게 實現할 수 있어야 하며(實現의 용이성)

-또한 多數의 subsystem으로 분할 記述하여 설계할 수 있는 記述方法이 요구된다.

상기와 같은 요구조건은 古典인 記述方法(例로써 Huffman의 방법, flow chart 혹은 timing diagram)으로는 만족 시킬 수 없으며, 따라서 1970년대 후반에 개발된 graph에 의한 방법인 GRAFCET를 이용하여 system의 기능을 記述하였으며, 記述된 알고리즘으로부터 入出庫 시스템 및 搬送 시스템을 각각 subsystem으로 한 계층구조로서 Z80 마이크로프로세서에 의하여 實現하였다.

본 논문의 시스템記述에 사용된 GRAFCET의 구조적인 研究,⁵⁾ 다양한 産業 자동화에의 적용에 관한 研究⁶⁾ 및 GRAFCET에 의해 記述된 sequential system의 實現에 관한 研究^{7,8,9,10)} 등은 각각의 참고문헌에서 찾아볼 수 있다.

II. GRAFCET에 의한 각 시스템의 記述

1. 入出庫 시스템의 記述

자동창고의 入出庫 시스템은 stacker crane을 이용하여 자동적으로 pallet를 rack에 入출고시키는 시스템이다. 본 시스템은 256개의 rack(x축16,y축16)을 갖는 부품창고의 入出庫 시스템으로서(rack수는 實現 program에 의해 임의로 조정가능), 入庫區는 rack번지(0,0)에 1개와 出庫區는 rack번지(0,0) 및 (F,0)에 즉 자동창고 양측에 各各 하나씩 2개로 확장하였으며(그림 1 참조), S. C의 1 회이동으로 서로 다른

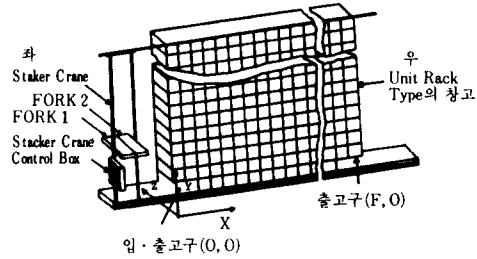


그림 1. 자동창고의 구조
Fig. 1. Structure of automated warehouse system.

rack에 있는 2 개의 pallet를 入출고할 수 있도록 하였으며, S. C의 주행 및 승강이동이 동시에 이루어지도록 기능을 확장하였다.

그림 1의 구조를 갖는 자동창고의 入出庫 시스템에

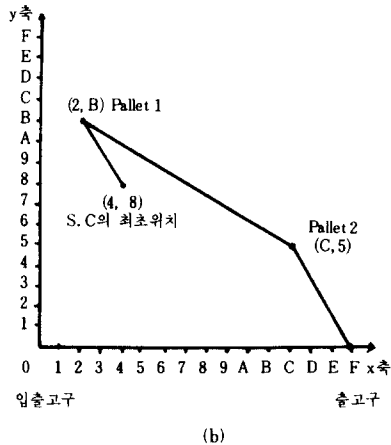
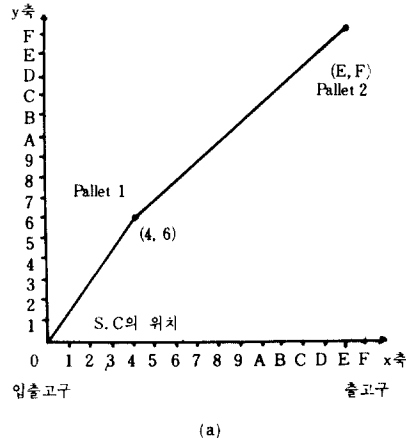


그림 2. 入출고 경로
(a) 入고경로 (b) 出고 경로

Fig. 2. Warehousing-delivery path.
(a) Warehousing path, (b) Delivery path.

대한 기능적 사양은 다음과 같으며 이상의 구조 및 기능적 사양을 갖는 입출고 시스템을 그림 3에 GRAFCET에 의해 記述하였으며 그림 3에서 사용한 actions 및 logic conditions의 심볼 및 의미는 표 1과 같다.

i) 임의의 위치에 있는 S.C는 입고명령어 의해 入庫區로 이동한 후 입고시킬 1개 또는 2개의 pallet를 지정된 rack에 입고시키고 입고시킨 위치에서 정지한다.

ii) 입고명령어 의해 S.C는 최소거리를 이동하기 위해 지정된 2개의 rack번지중 각각의 x축값을 비교하여 入庫區로 부터 가까운 rack으로 이동한다(그림 2(a)).

iii) 임의위치의 S.C는 출고명령어 의해 다음의 經路선택 알고리즘에 의해서 1개 또는 2개의 rack에 있는 pallet를 좌측(O, O) 또는 우측(F, O)출고구로 출고시킨다(그림 3의 step0, 1, 3~7) 經路선택 알고리즘(그림 2(b))

Step 1. 현재 S.C의 x축값과 출고시킬 rack의 x축값을 비교한다.

Step 2. 현재 S.C의 x축값과 출고시킬 또다른 rack의 x축값을 비교한다.

Step 3. Step1과 Step2에서 구한값을 비교하여 작은값을 갖는 rack의 pallet를 먼저 취출한후 나머지

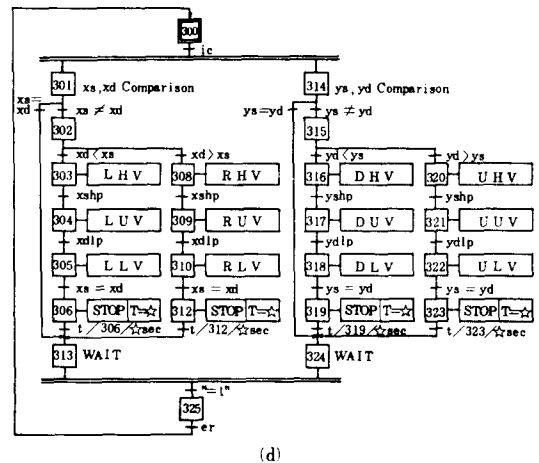
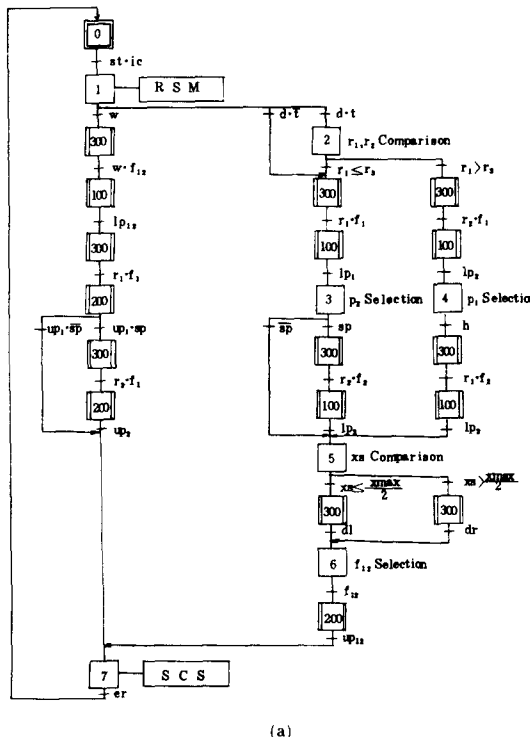
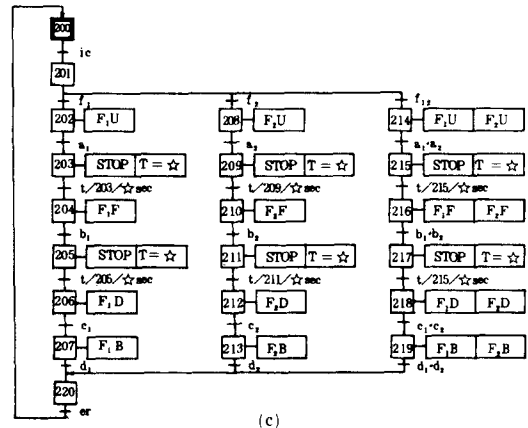
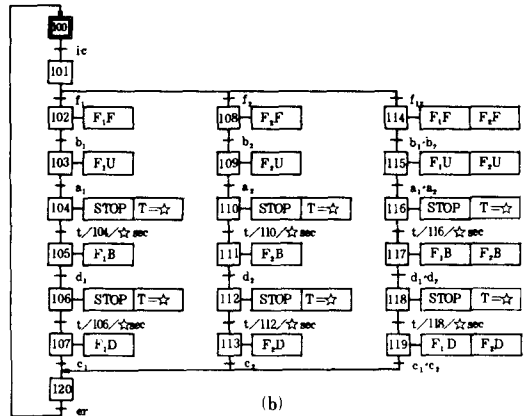


그림 3. 입출고 시스템의 GRAFCET

- (a) 입출고 시스템
 - (b) 포착 subroutine
 - (c) 배열 subroutine
 - (d) S.C의 이동 subroutine
- Fig. 3. GRAFCET of warehousing-delivery system.
 (a) Warehousing-delivery system.
 (b) Catch subroutine.
 (c) Arrangement subroutine.
 (d) Motion subroutine of S.C.

rack의 pallet를 취출한다.

Step 4. x축의 중앙값과 현재 S.C의 x축값을 비교하여 S.C의 x축값이 작거나 같으면 rack번지(O, O)로, 크면 rack번지(F, O)로 출고한다

iv) S.C가 rack에 pallet를 입출고시킬때 S.C의 이동거리 및 수행시간을 단축하기 위하여 2개의 fork를 이용하여 1개 또는 2개의 pallet를 개별 또는 동시에 포착, 배열한다(그림 3(b), (c)).

v) S.C는 이동시 이동거리 및 수행시간을 단축하기 위하여 주행이동 및 승강이동을 동시에 행한다. 이와같이 병렬 sequence를 하나의 controller에 의해 수행할 수 있음은 시스템의 response time이 충분히 크므로 가능하며, S.C의 이동시 관성에 따른 정확한 위치제어를 위해 rack전후에 加減速 위치 감지 sensor를 설치한다. (그림 3(d))

2. 搬送 시스템의 記述

본 시스템에서는 부품자동창고의 入出庫 시스템과

표 1. 입출고 시스템의 actions과 logic conditions의 심볼 및 그 의미

Table 1. Natures and symbols of actions and logic conditions for warehousing-delivery system.

* Actions

RSM	Memory에 입출고 정보 내장	Processor
SCS	Stacker Crane Stop	Motor
F _{1,2} A	Fork 1(2) 앞으로 이동	"
F _{1,2} B	Fork 1(2) 뒤로 이동	"
F _{1,2} U	Fork 1(2) 위로 이동	"
F _{1,2} D	Fork 1(2) 아래로 이동	"
LHV	왼쪽으로 가속	"
LUV	왼쪽으로 등속	"
LLV	왼쪽으로 감속	"
RHV	오른쪽으로 가속	"
RUV	오른쪽으로 등속	"
RLV	오른쪽으로 감속	"
UHV	위로 가속	"
UUV	위로 등속	"
ULV	위로감속	"
DHV	아래로 가속	"
DUV	아래로 등속	"
DLV	아래로 감속	"
STOP	STOP	"
T=☆	☆ Seconds동안 Activation	"

* Informations

st	Start	Push Button
ic	Initial Condition	"
w	Warehousing	End of Course
d	Delivery	"
f _{1,2}}	Fork 1(2) 선택	"
f _{12}}	Fork1, Fork2 선택	"
r _{1,2}}	Rack 1(2) 선택	"
t	Pallet 1, Pallet 2	"
t̄	Pallet 1 or Pallet 2	"
sp	Pallet 선택	"
sp̄	Pallet 선택	"
up _{1,2}}	Pallet 1(2) 배열	"
up _{12}}	Pallet 1, Pallet 2 배열	"
lp _{1,2}}	Pallet 1(2) 포착	"
lp _{12}}	Pallet 1, Pallet2 포착	"
dl	좌측 출고구 도착	"
dr	우측 출고구 도착	"
a _{1,2}}	Fork 1(2) 위	"
b _{1,2}}	Fork 1(2) 앞	"
c _{1,2}}	Fork 1(2) 아래	"
d _{1,2}}	Fork 1(2) 뒤	"
t/i/☆sec	☆ Seconds동안 Step i Activation	"
xs	Stacker Crane의 X축 위치	"
xd	X축 목적지	"
ys	Stacker Crane의 Y축 위치	"
yd	Y축 목적지	"
"= 1"	Always True	"
er	Sequence End	"

생산line을 연결하는데 2대의 無人 搬送車를 이용함으로써 시스템의 효율을 높이고 無人 搬送車가 이용하는 搬送路는 공통 operation part(그림 4의 EF구간)로 구성함으로써 공간의 이용율을 높이도록 하였다(그림 4 참조).

그림 4와 같이 설계된 搬送 시스템의 기능적 사양은 다음과 같으며 이상의 구조 및 기능적 사양을 갖는 반송시스템을 GRAFCET에 의한 記述은 그림 5(a), (b), (c)에 나타내었으며 그림 5에서 사용한 actions 및 logic conditions의 심볼 및 의미는 표 2와 같다.

i) 부품의 출고시 보다 빠른 반송을 위하여 入, 出庫區와 無人 搬送車를 동기시킨다. 즉 생산line에서 필요한 부품을 call하면 제 3의 동기 시스템에서 入出庫 시스템에 출고명령과 함께 대기점의 無人 搬送車에 걸

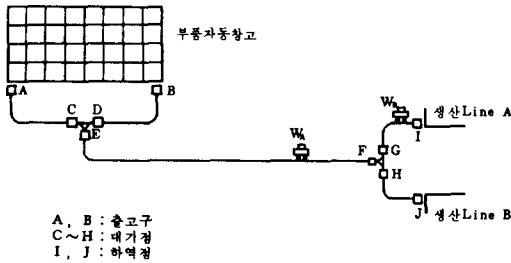


그림 4. 반송 시스템의 구성
Fig. 4. Construction of unmanned transportation system.

정된 出庫區로 이동하라는 명령을 준다(그림 5(a), (b)의 두번째 Step).

ii) 無人 搬送車는 生産line으로 공급될 부품을 반송한다. 이때 공통 operation part는 어느 한 시점에 2대의 無人 搬送車중 어느 한대만이 이용 가능하므로 wait step을 두어 이러한 기능을 처리한다. (그림 5(a), (b)의 step12, 15, 32, 35).

iii) 無人 搬送車가 生産line에 pallet를 배럴할 때는 시스템의 안정성을 고려하여 t초동안에 이루어지도록 한다(그림 5(a), (b)의 step11, 14, 31, 34).

iv) 반송업무를 마친 無人 搬送車는 다른 call이 없을 경우에는 대기점(C, D)에 돌아와 대기하도록 한다(그림 5(a), (b)의 11번째 step~15번째 step).

v) 無人 搬送車의 慣性문제를 해결하기 위하여 각 station(대기점, 입고구, 출고구, 하역점)에는 위치 sensor전후에 加減速을 위한 보조 sensor(s1, s2)를 설치하여 無人 搬送車의 출발 및 정지시 慣性문제를 해결한다(그림10(c)로 주어진 subroutine 50).

III. Subsystem에 의한 構成

1. Subsystem의 동기

Sequential system을 구성할 때 전체 시스템을 多數의 subsystem으로 나누어 구성한 후 각 subsystem을 동기시키면 시스템의 설계가 용이할 뿐만 아니라 시스템의 효율을 높일 수 있으므로 경제적인 시스템을 구성할 수 있다. 이와같은 subsystem에 의한 구성방법은;

-첫째방법으로 그림 6(a)와 같이 각 subsystem을 각각의 command part(controller)로 實現하며 이들 subsystem간을 동기시키기 위해서 두 subsystem간에 공통sequence를 두어 이 공통 sequence만을 各各의 subsystem이 공유함으로써 subsystem 各各은 독립성을 유지하며 subsystem간에 동기를 이루는 방법으로,

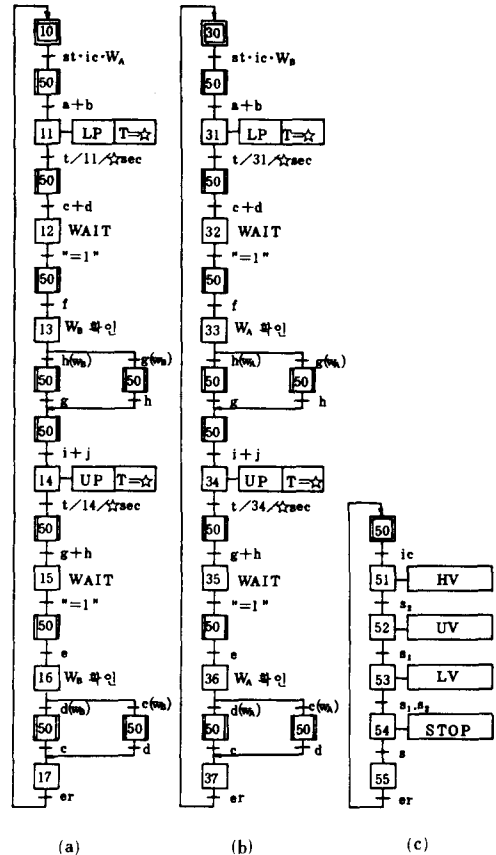


그림 5. 반송 시스템의 GRAFCET
(a) 무인 반송차 W_A
(b) 무인 반송차 W_B
(c) W_A, W_B 의 이동
Fig. 5. GRAFCET of unmanned transportation system.
(a) Unmanned vehicle W_A .
(b) Unmanned vehicle W_B .
(c) Motion of W_A and W_B .

상기와 같은 방법에 의한 實現은 operation part의 구조나 동작면에서 제약이 따르므로 각 subsystem을 서로 다른 방법으로 material化가 필요한 경우와 response time 혹은 maintenance의 용이성등에 따른 command part의 제약이 따르는 경우에 적합하다.

-두번째 방법으로는 그림 6(b)와 같이 각 subsystem을 각각의 command part로 實現하되 이들 subsystem간에는 직접적인 정보의 교환은 없고 이들 subsystem간의 동기를 위해서 제3의 동기 sequence를 구성하여 각 subsystem들은 기능연계상 필요한 어느 한 step으로 이 제3의 동기sequence를 통해서만 정보를 교환한다. 이와같이 계층적 구성방법은 시스템

표 2. 반송 시스템의 actions과 logic conditions의 심볼 및 그 의미

Table 2. Natures and symbols of actions and logic conditions for unmanned transportation system.

* Actions

LP	Pallet 적재	Motor
UP	Pallet 하역	"
HV	가속이동	"
UV	동속이동	"
LV	감속이동	"
STOP	STOP	"
T=☆	☆ Seconds동안 Activation	"

Informations		☆ Seconds동안
st	Start	Push Button
ic	Initial condition	"
a~j, s	각 Station	End of Course
s ₁	Sensor 1 감지	"
s ₂	Sensor 2 감지	"
s = 1"	Always True	"
er	Sequence End	"

의 확장이 용이하다.

2. 入出庫 시스템과 搬送 시스템의 계층적 구성

入出庫 시스템과 搬送 시스템의 無人 搬送車 W_A, W_B 사이를 제어하기 위해 계층적으로 구성된 전체적인 구조는 그림 7 과 같다.

그림 7 에서 2nd level부분의 동기 시스템은 host processor로서 작용하며 각 subsystem사이를 동기시켜준다. 入出庫 시스템, W_A, W_B를 각각 subsystem1, subsystem2, subsystem3 로 구성하였으며 동기 시스템과 각 subsystem사이를 동기시키기 위한 동기정보는 그림 7 에 표시하였다.

GRAFNET를 이용하여 그림 7 과 같은 구조를 갖는 시스템의 구성절차는 첫번째단계로 각 subsystem을 공통 sequence나 동기sequence를 고려하지 않고 보조 GRAFCET로 記述한다. 이는 앞에서 記述된 그림 3 (a)와 그림 5 (a), (b)와 같다. 두번째 단계는 기술된 보조GRAFCET를 가지고 subsystem사이의 동기를 위한 동기sequence를 도입하여 記述한다. 이를 그림8에 나타내었다. 그림 8 에서 step70과 step75로 이루어지는 loop는 동기sequence로서 subsystem 1 과 subsystem2, subsystem 1 과 subsystem 3 를 동기 시키는 sequence 이다. 세번째단계는 그림 8 의 각각의 subsystem에 대

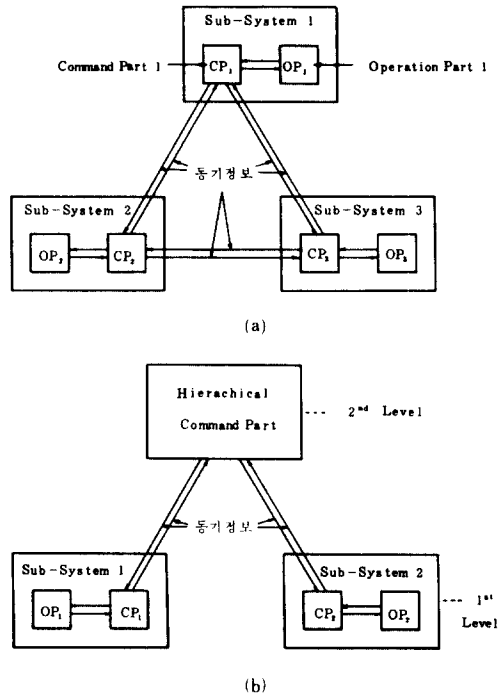


그림 6. Subsystem간의 동기
(a) 3 개의 subsystem의 기능적 구조
(b) 2nd-level 계층구조

Fig. 6. Synchronization of between subsystems.
(a) Functional structure of three subsystems.
(b) Hierarchical structure of 2nd-level.

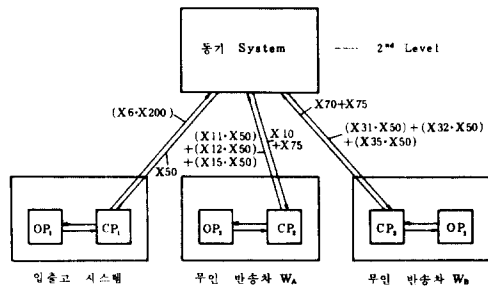


그림 7. 입출고 및 반송시스템의 계층구조
Fig. 7. Hierarchical structure of warehousing-delivery and unmanned transportation system.

해 command part를 구성한다. 이를 그림 9에 記述하였다. 그림 9 (d)는 각 subsystem사이의 동기를 위한 2nd level의 동기system의 GRAFCET이다. 이러한 구성절차에 의해 구성된 시스템은 각각 하나의 완전한 sequence를 이루게 된다.

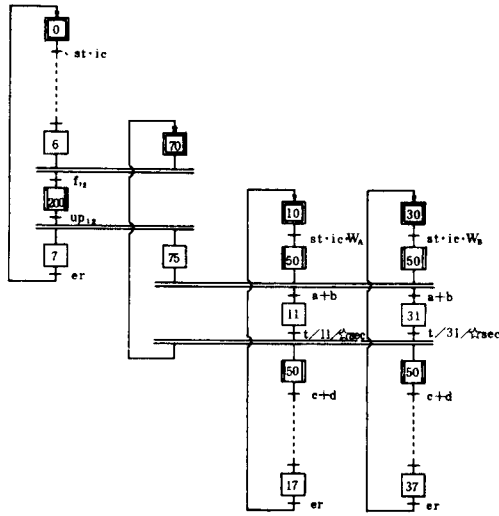


그림 8. 동기sequence를 고려한 입출고 및 반송 시스템의 GRAFCET

Fig. 8. GRAFCET of warehousing-delivery and unmanned transportation system considering the synchronization sequence.

IV. Microprocessor에 의한 實現

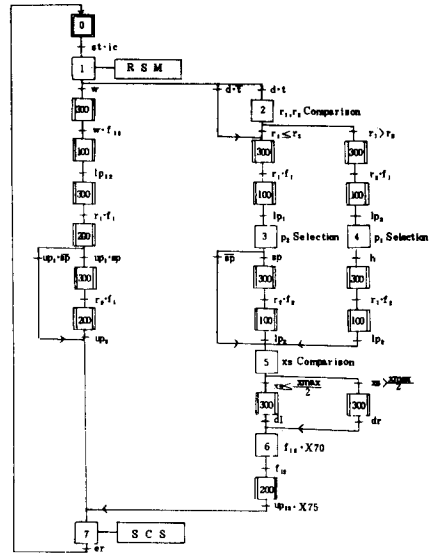
State graph^[10](병렬sequence가 없는 GRAFCET)에 의해 記述되는 sequential system은 어느 한 순간에 하나의 step만이 active되므로 이 active된 step에 대하여 다음 step으로의 천이를 위한 입력천이조건(transition)과 이에 해당하는 출력을 프로그램하면 된다.

이와같이 state graph를 프로그램에 의한 實現시 프로그램의 구조는 step수 만큼의 phase로 구분할 수 있으며, 하나의 step(P_i)에 대응되는 하나의 phase(φ_i)는 transition/next step의 쌍을 형성하는 control part(φ₁)와 출력변수에 대한 logic값(local combination)을 나타내는 operation part(φ₂)로 구분할 수 있으며 이들 phase들은 separator에 의해 구분된다(그림10참조). 따라서 state graph의 program에 의한 實現은 하나의 step에 대응하는 phase와 separator를 포함하는 하나의 program module의 집합으로 구성된다. 또한 security와 visualization등을 위하여 general combinational part를 추가할 수 있다.

이상의 개념으로부터 프로그램에 의한 實現을 위해 다음과 같은 명령어를 정의한다.

-TI(A) : 입력천이 조건을 테스트하는 언어로서 입력변수는 A번지로 나타낸다(test input).

-MP(A) : evolution condition이 만족될 경우 해당 A번지 step의 activation을 나타낸다(mark step).



(a)

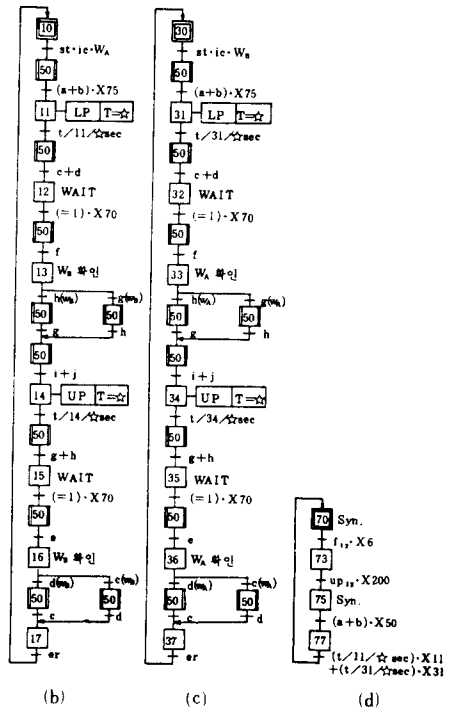


그림 9. 그림 8의 command part 실현

- (a) 입출고 시스템
- (b) 무인 반송차 W_A
- (c) 무인 반송차 W_B
- (d) 동기 시스템

Fig. 9. Realization of command part using Fig. 8.

- (a) Warehousing-delivery system.
- (b) Unmanned vehicle W_A.
- (c) Unmanned vehicle W_B.
- (d) Synchronization system.

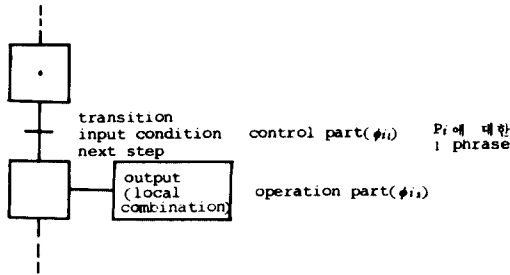


그림 10. Program에 의한 실현 개념도
Fig. 10. Diagram of realization concept by program.

-AO(A) : 번지A의 출력변수를 나타낸다(affection output).

-SP(A) : separator로서 partial combinational logic을 나타내는 것으로 이러한 언어의 정의는 프로그램의 변형을 용이하게 할 수 있다(separator).

이상의 기본적인 4개의 명령어는 하나의 program module을 형성하며 이 외에도 입력변수를 위해 logical OR, AND 및 번지A의 peripheral에 값V를 affection 하는 AP(A) 《V》가 추가 정의될 수도 있다.

이상과 같은 개념을 바탕으로 Z80 마이크로프로세서의 명령어로 변환시키면 다음과 같다.

-TI는 임의의 상태가 다음상태로 상태천이 하기 위해 만족해야 할 logic condition을 테스트하는 언어로서 마이크로프로세서의 레지스터에 내장된 값과 외부에서 입력되는 logic condition의 값이 일치하는지를 비교 판단하는 IN(input), CP(compare) 그리고 일치하지 않았을 때는 만족되는 논리상태의 값이 입력될 때까지 계속받아 들이는 JP(jump) 명령에 해당된다.

-MP는 단일 sequence에서는 자동적으로 다음상태로 상태천이하므로 지정할 필요는 없으나 선택 sequence 및 병렬 sequence에서는 다음상태 지정을 JP(jump) 명령으로 지정하여 준다.

-SP는 partial combinational logic이 필요한 경우 AND, OR등과 같은 logic instruction등이 해당된다.

이밖에 각 상태의 출력과 논리상태를 레지스터에 저장하는 LD(load) 명령이 필요하다.

GRAF CET에 대응하는 위의 각 명령들은 앞서 언급한 바와 같이 하나의 step 및 transition에 해당되는 하나의 program module을 이루게 되며, 이러한 program module개념을 이용하면 이미 記述된 GRAF CET로부터 직접 손쉽게 프로그램을 작성할 수 있어 programmed logic에 의한 시스템의 實現이 용이하다. 예로써 그림 5(c)를 Z80 명령어로 작성한 프로그램을 그림 11에 나타내었다.

```

MACRO-80 3.33 12-Sep-79 PAGE 1

0000 .
      .Z80
      ASEG
      ORG
      EQU 2000H
      EQU 80H
0000 LD A,4FH;PIO INITIALIZATION
0002 OUT (82H),A
0004 LD A,0FH
0006 OUT (83H),A
0008 LD C,81H
000A START: LD B,60H;INITIAL STEP
000C OUT (C),B
000E LD D,61H
0010 AA: IN A,(P);INITIAL CONDITION
0012 CP D
0014 JP NZ,AA
0016 LD B,61H
0018 OUT (C),B;HIGH VELOCITY
001A LD D,62H
001C BB: IN A,(P)
001E CP D
0020 JP NZ,BB
0022 LD B,62H
0024 OUT (C),B;UNIFORM VELOCITY
0026 LD D,63H
0028 CC: IN A,(P)
002A CP D
002C JP NZ,CC
002E LD B,63H
0030 OUT (C),B;LOW VELOCITY
0032 LD D,64H
0034 DD: IN A,(P)
0036 CP D
0038 JP NZ,DD
003A LD B,64H
003C OUT (C),B;MOTOR STOP
003E LD D,65H
0040 EE: IN A,(P)
0042 CP D
0044 JP NZ,EE
0046 LD B,65H
0048 OUT (C),B;SEQUENCE END
004A JP START
      END
    
```

그림 11. Z80명령어에 의한 무인 반송차 이동program
Fig. 11. Motion program of unmanned vehicle by Z 80instruction language.

Program module 개념을 이용해 작성한 프로그램이 해당 process의 sequence 처리를 위한 interface구조를 그림 12에 나타내었다. Z80 마이크로프로세서에 데이터를 입출력하기 위한 I/O소자로 PIO를 사용하였으며 PIO의 입력port에는 buffer 및 encoder를, 출력port에는 buffer 및 decoder등을 사용하였다. Encoder에의 입력은 GRAF CET의 evolution규칙, 즉 다음 step으로의 천이는 현재step이 activation되고 해당 입력 천이조건이 만족되어야 하므로, activation된 현재의 step(decoder의 출력)과 해당 입력천이조건을 ANDing하여 encoder에 입력하였다. 이와같이 입출고 및 반송시스템의 sequential controller의 구성을 위하여 1대의 host 마이크로프로세서와 3대의 sub 마이크로프로세서를 사용하였으며 각각의 마이크로프로세서에는 상기에서 언급한 논리소자를 부가하여 계층적으로 구성하였다.

V. 結 論

本 論文은 자동창고의 入出庫 시스템을 입고구 1개와 출고구 2개로 확장하였으며, S.C의 이동시 주행 및 승강이동을 동시에 행하며, S.C의 1회이동에 pallet를 2개까지 포착 배열할 수 있도록 설계하여 기존의 시스템에 비해 입출고 시간을 단축시켰다. 또

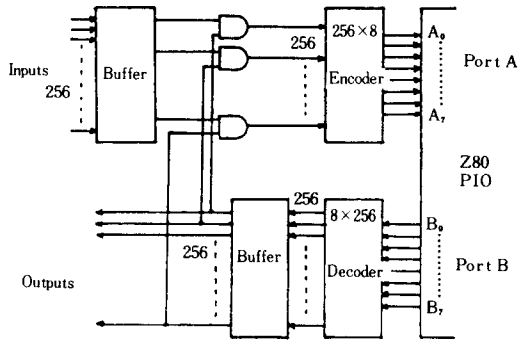


그림 12. Interface 구조
Fig. 12. Structure of interface.

한 시스템 효율을 높이기 위하여 설계된 入出庫 시스템과 2대의 無人 搬送車를 이용한 搬送 시스템을 계층적으로 構成하여 마이크로프로세서 (Z80)에 의하여 實現하였다.

시스템을 記述하는데 GRAFCET를 이용함으로써 전체 시스템을 각 subsystem으로 분할 記述이 용이함을 보였고 記述된 subsystem을 계층적으로 구성할 때 문제가 되는 각 subsystem간의 동기 정보까지 쉽게 記述할 수 있음을 보였다.

記述된 전체 시스템의 microprocessor에 의한 實現은 program module 개념을 정의함으로써 기존 마이크로프로세서 (Z80) 명령어로 쉽게 프로그램을 작성할 수 있음을 보였다.

본 시스템은 자동창고의 入出庫 시스템과 생산 line을 위주로 搬送 시스템을 설계 하였으나 재고관리 시스템 및 생산관리 시스템등으로 확장 연계함으로써 無人 工場의 實現에 많은 도움이 되리라 기대된다.

參 考 文 獻

[1] 양재규, "무인 반송차와 그 응용," 전자공학회지, vol. 13, no. 2, pp. 69-76, 4월, 1986년.

- [2] 溝淵博幸, "自動倉庫システムと 情報處理," 住友重機械技報, vol. 29, no. 87, pp. 96-101, Dec. 1981.
- [3] Robert E. Smith, "Robotic vehicles will perform tasks ranging from production retrieval to sub-assembly work in factory of future", *Industrial Engineering*, pp.62-72, Sep., 1983.
- [4] M. Moriss Mano, *Digital Logic and Computer Design*, Prentice Hall, 1979.
- [5] J. C. Bossy, P. Brard, P. Faugere et C. Merlaud, *Le GRAFCET a Partique et ses Applications*, Educavivre, Paris, 1979.
- [6] G. Michel, C. Lurgeau et B. Espiau, *Les Automates Programmables Industriels* Dunod technique, Paris, 1979.
- [7] B. Taconet et B. Challot, *Programmation du GRAFCET Sur Automate a Langage Logique a Relais ou Booléen*, Le Nouvel Automatism, pp.44-45, Jan.-Feb. 1979.
- [8] T. Maurin et M. Robin, *GRAFCET une Traduction Micro-Programmee*, *Mesure-Regulation-Quotomatisme*, pp.75-80, Mai, 1979.
- [9] S. Thelliez et J. M. Toulotte, *Applications Industrielles du GRAFCET*, Eyrolles, Paris, 1983.
- [10] 우광준, "GRAFCET로 기술된 순서논리 시스템의 race없는 program으로서의 합성," 전자공학회지, 제21권, 제6호, pp. 56-63, 11월, 1984년.
- [11] Rodney Zaks, *Programming the Z 80*, Sybex, 1980.