

論 文

마이크로컴퓨터를 이용한 PM스텝모터의 디지털位置制御시스템

正會員 成 源 基* 正會員 崔 宗 秀** 正會員 河 龍 守***

The Digital Position Control System for PM Step Motor Using Microcomputer

Weon Gi SEONG*, Jong Soo CHOI** and Yong Soo HA****Regular Members*

要 約 本 論 文 은 마이크로컴퓨터를 이용한 PM스텝모터의 디지털位置制御시스템으로 既存의 位置制御시스템의 多數의 로직回路를 줄이고 logic function을 소프트웨어로 代替하기 위한 알고리즘을 開發하고 이 알고리즘을 만족시키는 프로그램을 作成하여 EPROM에 내장시켜 디지털位置制御시스템을 構成하고 實驗을 통하여 시스템의 妥當性을 立證하였다.

ABSTRACT This paper presents the digital position control system for PM step motor using microcomputer. An algorithm is developed not only for the reduction of a large number of logiccircuits of the existing position control system but also for the substitution to software for logic functions. The program to satisfy this algorithm is studied out, which is written into the EPROM to make up the digital position control system. The justifiability of the system is proved by experiment.

1. 序 論

最近 制御分野에서 컴퓨터를 이용한 離散制御 시스템이 많이 研究되고 있으며 스텝모터는 離散入力信號에 따라 等角回轉運動을 하는 器機로 開루우프제어로서 높은 정도의 位置制御가 可能하여 NC, 로봇트머니퓰레이터, 프린터등의 디지털 제어에 적합한 長點을 가진다.

現在 digital circuit elements의 開發이 매우

빠른 속도로 발전됨에 따라 古典的인 아날로그 회로들을 하나의 단일칩내에 集積시키는 것이 可能하게되어 매우 경제적이며 유용하게 되었다. 스텝모터는 이러한 digital circuit elements의 發展과 더불어 많은 발전을 하였으며 應用分野도 매우 넓어졌다. 스텝모터의 離散制御에 대한 研究는 1957년 A. G. Thomas & JF. Fleischer⁽¹⁾ 등에 의해 시작되어 그 이후 많은 研究가 계속되어오고 있다^{(2),(3)}.

本 論文에서는 스텝모터중에서 구조가 간단하고 견고하며 높은 信賴度를 갖기 때문에 제어시스템에 널리 사용되는 PM(permanent magnet) 스텝모터를 마이크로 컴퓨터를 이용하여 既存의 디지털位置制御시스템을 改善하여 多數의 로직

* ** 中央大學校工科學電子工學科
Dept. of Electronic Engineering, Chungang University,
Seoul 151 Korea.
論文番號 : 85 - 29 (接受 1985. 8. 13)

회로를 줄이고 Logic function을 소프트웨어로 代替하기 위한 알고리즘을 開發하고 이 알고리즘을 만족시키는 프로그램을 作成하여 EPROM에 내장시켜 디지털위치제어시스템을 構成하고 實驗을 통해 본 시스템이 入力信號에 따라 높은 正確度를 가지고 位置制御가 됨을 보이교자 한다.

2. PM 스텝모터의 特性

1) 單-스텝特性

단일스텝 상태에서 스텝모터의 rotor의 움직임은 비틀림진자와 類似하다. stator flux에 의해 magnetic moment가 發生되며 이때 電流의 上昇率이 rotor의 최대운동에너지 入力を 決定한다. 負荷의 마찰력은 회전토크에 대한 減衰로서 나타나며 시스템의 관성력은 rotor와 負荷의 관성력의 합으로 나타난다. 증가하는 관성력은 rotor의 진동의 周波數를 낮추는 반면 增加된 電流上昇率은 진동의 크기를 증가시킨다¹⁴⁾. 명확히 진동의 크기와 settling time은 모터를 정지시킨 후 재회전시키거나 逆回轉시키는 시스템의 能力을 決定하는데 있어서나 대부분의 응용에서도 매우 重要한 要素이다.

2) 멀티스텝의 特性

連續되는 pulse train에 의한 PM 스텝모터의 動作은 첫째 입력펄스율, 둘째 stator winding 코일에서의 過渡電流, 셋째 rotor의 기계적 變數 등에 의해 많은 영향을 받는다 stator 電流의 過渡應答은 스텝모터의 入力 step rate가 增加됨에 따라 스텝모터의 최대발생토크에 영향을 미친다. 각 step command의 사이시간이 stator의 L/R 시정수에 접근하게 되면 電流는 期待하는 값에 도달하지 못하며 natural frequency와 모터의 발생토크가 減少한다. 또한 모터가 step command에 대해 反應할 수 있는 load torque의 총량 또한 감소하게 된다. 그러므로 一般的으로 스텝모터의 構動回路는 stator 과도전류에 대해 補償할 수 있도록 設計되어야 한다.

3) 角度的 正確度

典型的인 스텝모터의 正確도는 비축적성으로

$\pm 3\% \sim \pm 5\%$ 내에 있다. 스텝모터가 이러한 正確度를 갖는 것은 스텝모터의 特性인 특유의 對稱的 構造와 많은 극점에 대한 magnetic force의 平均성때문이다. 이러한 正確度에도 불구하고 정도가 매우 까다로운 應用分野에서 문제가 생기는 原因은 모터의 stiffness와 winding current의 不均型때문이다. stiffness는 스텝모터의 勵起된 안정위치로부터 rotor가 빗나갔을 때 發生되는 torque displacement에 나타난다.

3. 디지털위치제어시스템의 構成

1) 컴퓨터 인터페이스의 선택

모터의 field 코일 및 구동회로와 센서들을 제외한 모든 素子들은 典型的인 binary logic으로 이루어져 있으며 理論적으로 두 부분으로 區分된다. 즉 Motion controller와 Processor로 나뉘어진다. 스텝모터가 관계되는한 controller와 processor 사이에 많은 interface level을 挿入시킬 수 있다. 一般的으로 controller에서 더 많은 logic circuits가 필요하면 할수록 마이크로 컴퓨터로부터 받는 監視는 적어지게 될 것이다. 즉 인터페이스를 마이크로컴퓨터 쪽에 치우치면 Controller 쪽의 logic circuits가 복잡해지고 반대로 Controller 쪽에 치우치면 logic circuits는 간단해지는 반면 마이크로 컴퓨터 쪽에 많은 프로그램이 내장되어야 함을 意味한다¹⁵⁾. 그림 1을 참조해 볼 때 Interface level을 어느 단계에 設定하여 system을 構成하느냐 하는 것은 경제성의 문제로 감부된다. 즉 logic function을 프로그램

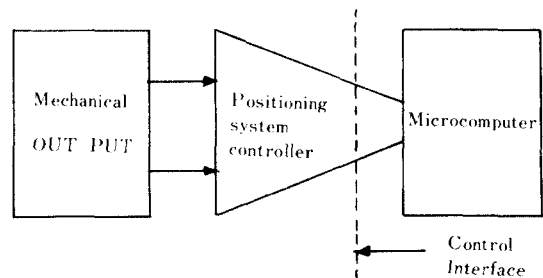


그림 1 컴퓨터 인터페이스의 선택
Computer interface selection.

으로 代替하여 메모리에 내장하는 것과 controller内部的 logic circuits와의 비교가 고려되어야 할 문제인 것이다. 本 論文에서는 interface level을 마이크로컴퓨터部가 logic function을 흡수하는 단계에 設定하여 시스템을 構成하였다.

2) 駆動回路의 設計

駆動回路의 設計는 스텝모터를 利用한 시스템을 構成하는데 있어서 가장 重要한 部分중의 하나

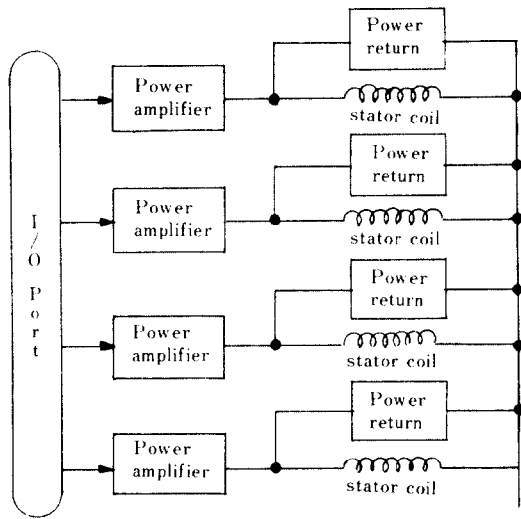


그림 2 스텝모터의 구동시스템 블록도
Block diagram of step motor driver.

이다. 負荷에 걸리는 動力의 관점에서 뿐만 아니라 가격과 전력의 소비 및 효율성의 관점에 있어서도 전체시스템의 성능은 구동시스템에서 많은 영향을 받는다. 그림 2는 스텝모터의 구동시스템을 나타낸 것이다. 이 시스템의 入力段에서는 마이크로컴퓨터의 인터페이스단으로부터 駆動信號을 받아서 電流를 增幅시켜 모터의 stator 코일을 勵磁시킨다. 또한 power return system은 한 step의 종단에 코일로부터 되돌아오는 電流를 제거하여 power TR의 손상을 막는 役割을 한다. 모터는 rotor magnetic field와 stator magnetic field 사이에 상호작용의 상태변화를 일으켜 回轉되는 것이다. 주로 스텝모터의 drive mode에는 2가지方式이 있으며 이것은 stator 코일을 어떤方式으로 勵磁시켜서 모터의 回轉力을 發生시키는가에 따라 나뉜다. 첫째 방식은 one phase mode라고 하며 한번에 한조의 stator pole들만 勵磁시키는 방식으로 그림 3에 나타내었다. 또 다른 하나는 two-phase mode이며 이 방식은 한번에 두 조의 stator pole들을 勵磁시키는 것이다⁶⁾. 이 두 방식 모두 N° 씩 增加하여 回轉하게 되지만 출발위치에서는 $N^\circ/2$ 의 차이를 갖게 된다. two phase mode를 그림 4에 나타냈다. 本 論文에서의 전체 시스템의 구성도는 그림 5와 같다.

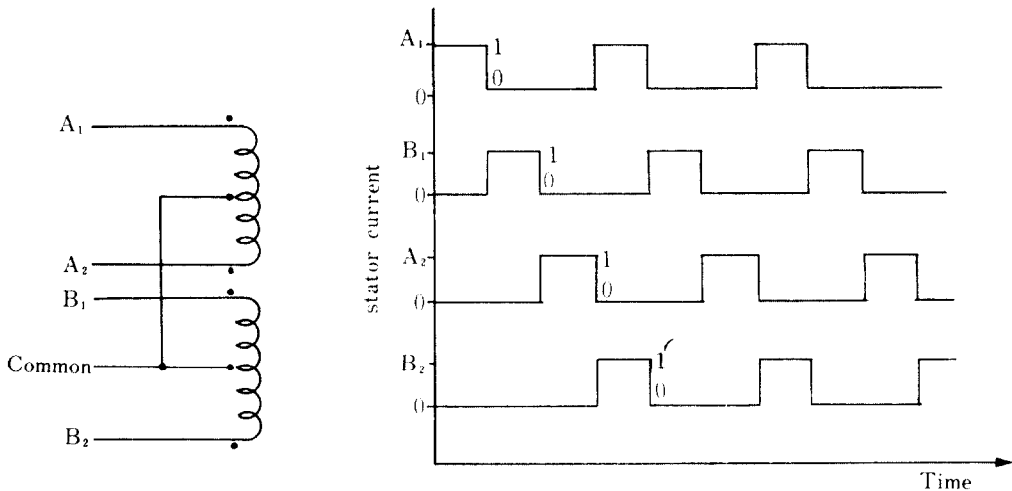


그림 3 one-phase mode의 선류파형
Current waveforms of one phase mode.

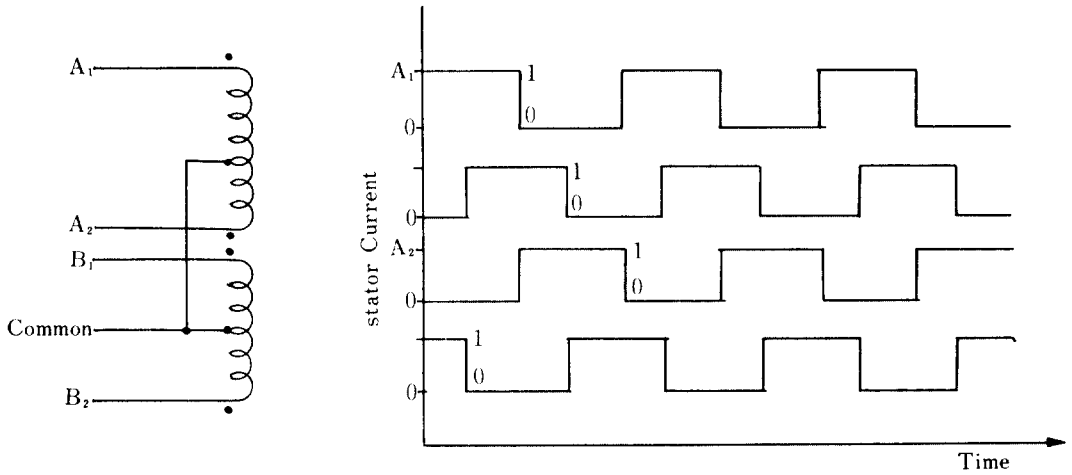


그림 4 Two phase mode의 전류파형
Current waveforms of two phase mode.

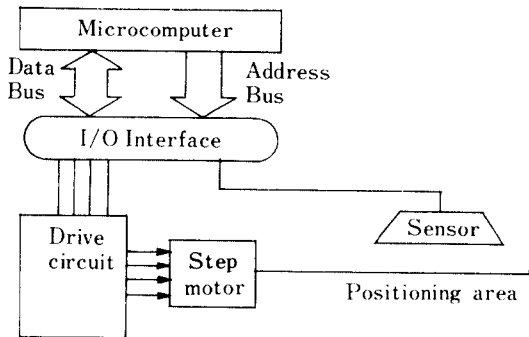


그림 5 시스템의 구성도
Block diagram of system.

Phase \ Step	A ₁	B ₁	A ₂	B ₂
1	1	1	0	0
2	0	1	1	0
3	0	0	1	1
4	1	0	0	1
5	1	1	0	0
⋮	⋮	⋮	⋮	⋮

그림 6 two phase mode 상태열
Two phase mode state sequence.

4. Positional Control logic function의 Software적 Algorithm

1) 回轉을 위한 서브루우틴

스텝모터를 回轉시키기 위한 stator의 勵磁方式으로 two phase mode를 사용했다. 앞절의 그림 4의 twophase mode에 대한 stator의 勵磁方式을 binary logic으로 나타내면 그림 6과 같다. 이것을 다시 상태변화로 표시하면 그림 7과 같다. 그림 7에서 s₁, s₂, s₃, s₄는 각각 state 1, state 2, state 3, state 4를 나타내며 이것은 任意로 붙여진 順序이다. 그러므로 위 4개의 狀態중 어느 것이나 初期의 狀態가 되어 순환될 수

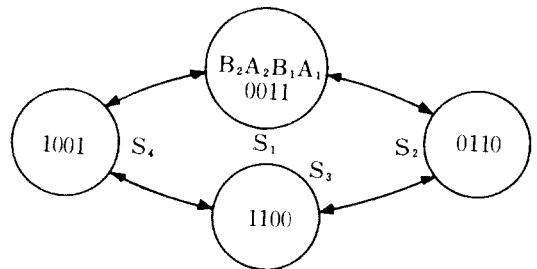


그림 7 stator 상태 변화도
Stator state diagram.

있다. 이러한 狀態의 變化는 4 비트 레지스터에 0011을 load시키고 Rotate Left 혹은 Rotate Right시킬 때의 狀態의 變化와 같음을 알 수 있다. 그러나 보통 8 비트 마이크로컴퓨터를 사용하게 되므로 8 비트 레지스터를 고려하면 001100

11의 데이터를 load시키고 같은 Rotate 명령을 實行하면 역시 下位 4 비트와 上位 4 비트의 상태 변화가 그림 7의 狀態變化圖와 같은 順序로 變化되므로 스텝모터를 回轉시키기 위해서는 I/O port를 통해 下位 4 비트나 上位 4 비트중의 하나를 驅動回路의 入力段으로 出力시키면 된다⁽⁷⁾ 또한 한 step出力후 다음 step을 出力시키기 까지의 時間은 결국 모터의 回轉速度에 關係되므로 Delay Subroutine을 追加하여 速度를 조절할 수 있다. 이러한 일련의 過定을 원하는 횟수만큼 反復하게 함으로서 shaft의 回轉角을 制御할 수 있다. 그림 8에 이러한 Algorithm을 flow chart로 보인다. 그림 8에서는 단순히 일정한 속도로 회전각을 制御하는 알고리즘을 제시하였다. 그러나 좀더 고려하지 않으면 안되는 문제점들이 있다. 그중의 하나가 Reference Position에 대한 것이다. 初期의 理想的인 tooth의 배열상태로부터 step動作이 시작되었다면 각

state에 따른 stator와 rotor의 tooth 배열상태는 1대1 對應으로 定義될 수 있을 것이다. 이러한 각 state에 따른 정해진 rotor의 位置를 reference position이라 한다. 이러한 reference position이 있으므로 rotor의 現在位置에 따라 初期의 state logic을 정해야 한다. 또한 모터가 한動作이 끝나게 되면 메모리에 그때의 state logic을 기억시켜 두고 다음 動作이 이어질때 기억된 State logic으로 부터 信號를 發生해야 한다. 本 論文에서는 accumulator A에 state logic을 load시켜 Rotate Right, 혹은 Rotate Left시키고 이것을 I/O Port에 出力시킴으로서 모터를 驅動하도록 構成하였으므로 일련의 動作이 수행되면 그때의 A의 內容을 기억시켜두도록 하였다. 또한 positioning area 상에서 Arm의 位置를 항상 tracking하고 있어야 한다. 왜냐하면 positioning을 하기 위해 Arm과 Target와의 Distance를 계산하고 움직여야 할 Direction을 決定하기 위해 필요한 要素이기 때문이다. 그러므로 Arm의 tracking하는 pointer를 設定하여 1 step Rotation 시킬때마다 Increment 혹은 Decrement를 수행하도록 하면 한 動作이 끝나면 이 pointer Register의 內容을 정해진 memory 영역에 保管하도록 한다. 이러한 점들을 고려한 Rotation Algorithm을 그림 9에 나타낸다.

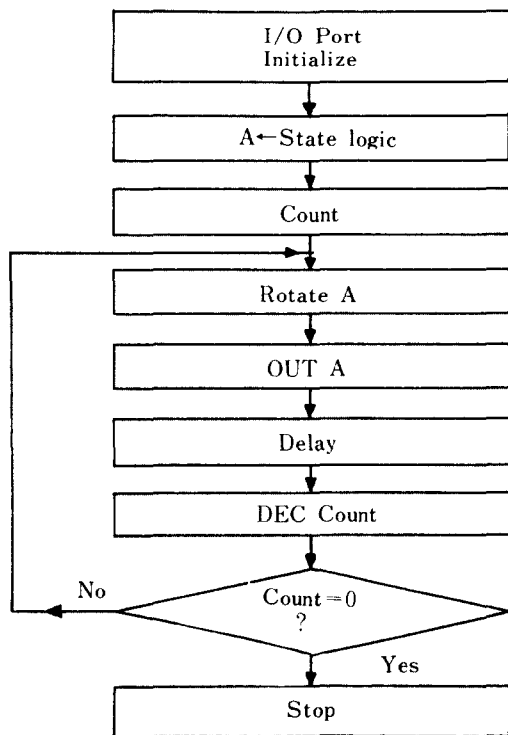


그림 8 기본적인 Rotation Algorithm.
Basic rotation algorithm.

2) Delay Subroutine

step과 step 사이의 時間은 모터의 回轉速度와 關係된다. 그러므로 프로그램으로 모터의 回轉速度를 制御할 수 있는 方法은 Rotation program loop 속에 Delay Subroutine을 挿入하는 것이다. 이 Delay Subroutine을 形成하기 위해서 우선 카운터 레지스터를 設定하고 거기에 遲延할 時間에 相當하는 카운터數를 load시키고 카운터의 內容이 0이 될 때까지 100p를 돌게 함으로써 실제 時間遲延의 效果를 얻었다. 한번의 100p를 遂行하는데 걸리는 時間을 약 10μsec라고 하면 16비트의 카운터를 使用하는 경우 最大loop數는 $2^{16} - 1 = 65535$ 가 되며 걸리는 時間은 약 0.65sec이다. 이것은 각step사이의 時間간격을 조절하기

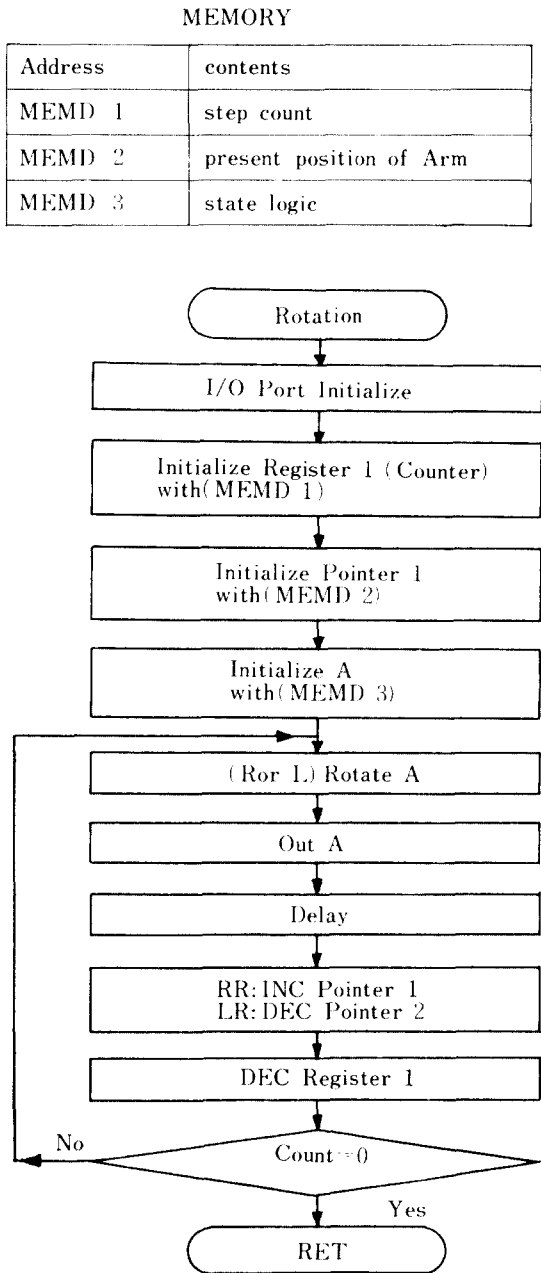


그림 9 회전 알고리즘
Rotation algorithm.

에는 적당하나 system의 動作중 일시성지했다가 다시 動作을 連續하는 경우에는 좀 더 긴 wait loop가 必要하게 된다. 이에 대한 고려로 두개의 loop를 갖는 Delay subroutine을 그림 10에 보인다.

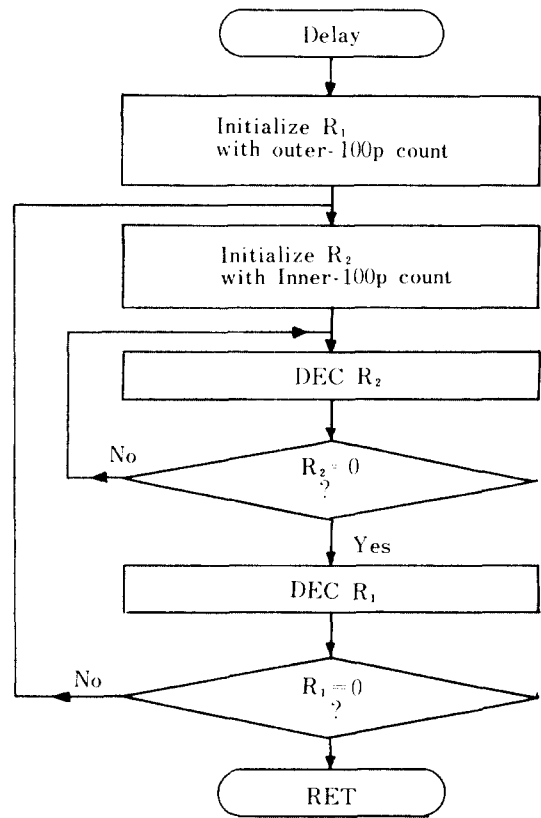


그림 10 지연 서브루우틴
Delay subroutine.

3) 방향과 거리 서브루우틴

本節에서는 입력port에서 step數로 변환된 target position을 받아들여 現在의 Arm의 位置와 의 연산에 의해 방향과 거리를 구하게 된다. 그림11에 位置制御領域을 나타내었다.

각각의 다른 서브루우틴들끼리의 상호연결을 容易하게 하기 위해 入力port로부터 데이터를 받아들이는 過程과 그 데이터를 처리하여 displacement를 구하는 過程은 따로 분리되어 프로그래밍되어야 한다. 그림12에 入力port로부터 데이터를 받아들여 設定된 메모리領域에 保管하는 過程을 나타내었다. 앞에서 받아들인 데이터를 가지고 現在의 Arm의 位置와의 연산에 의해 方向과 거리를 구하는 過程을 그림13에 나타낸다. 그림13에서 R Rotate Sub와 L Rotate Sub는 각각 Right Rotation Subroutine과 Left Ro-

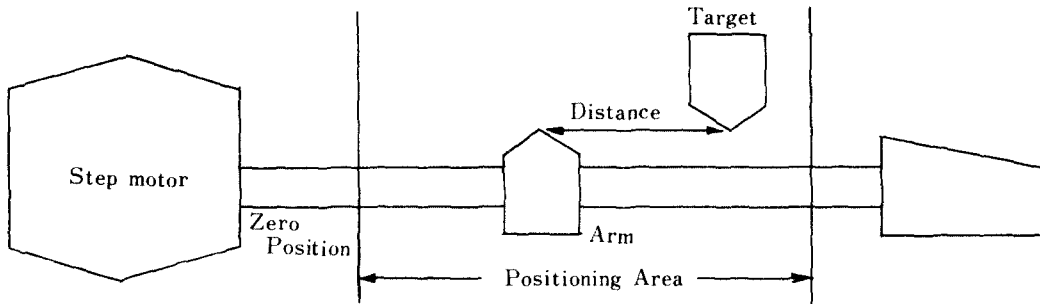


그림11 位置制御領域
Positioning area.

MEMORY	
Address	Contents
MEMD 6	Target Position

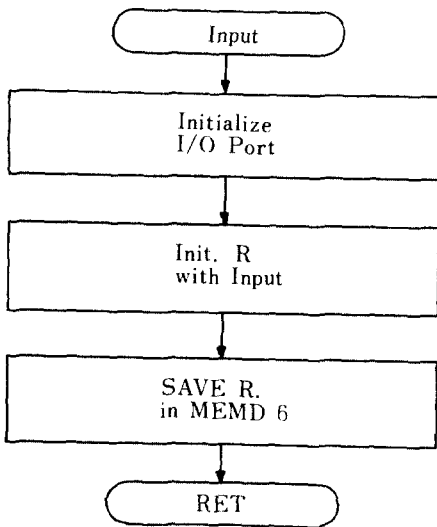


그림12 入力 데이터
Data input.

tion Subroutine을 나타낸다. 즉 방향과 거리가 계산되면 바로 該當되는 Rotation Sub를CALL하도록 되어 있다.

4) 連續的인 位置制御 서브루틴

그림11에서 몇개의 positioning point를 假定하고 이들을 連續的으로 順序에 따라 지적하는

動作을 假想할 수 있다. 이러한 動作에 대한 프로그램을 짜기 위해 알고리즘을 세우기로 한다. 앞의 Direction & Distance Sub에서 入力된 Target position Data는 特定된 메모리에 保管되어지고 그메모리번지를 참조하여 연산이 되어 自動的으로 Rotation Subroutine으로 넘어가도록 되어 있는 것을 利用하도록 한다. 즉 일련의 point들은 step數로 定義되어 特定된 메모리에 positioning順序에 따라 save시켜두고 이 데이터들 중 첫번째것을 Target Position Data가 save되는 特定된 메모리에 넣고 Direction & Distance Subroutine을 CALL하면 自動的으로 Rotation 動作까지 이어지게 된다. 이렇게 한점에 대한 positioning 動作이 끝나면 다시 다음 point에 대한 Data를 Target Position Data가 save되는 特定한 메모리에 넣고 같은 動作을 反復하게 되며 일련의 連續的인 positioning 動作이 完了된다. 이러한 알고리즘을 그림14에 보인다.

5. 實驗 및 檢討

實驗에 使用된 PM스텝모터는 8 pole 4phase이며 stator의 tooth數는 38개, rotor의 tooth數는 45이다. 또한 5[V] 1.5A/phase이며 1 step 당 2°씩 回轉하는 PM스텝모터이다. 實驗回路는 그림15에 나타내었다. 그림15에서 마이크로컴퓨터는 Z-80을 使用했으며 I/O port 로는 Z-80 PIO peripheral chip을 使用했다. 入力信號는 PIO의 A port를 使用하여 光TR에 光이 조사되는 狀態와 遮斷된 狀態에 따라 1과 0을

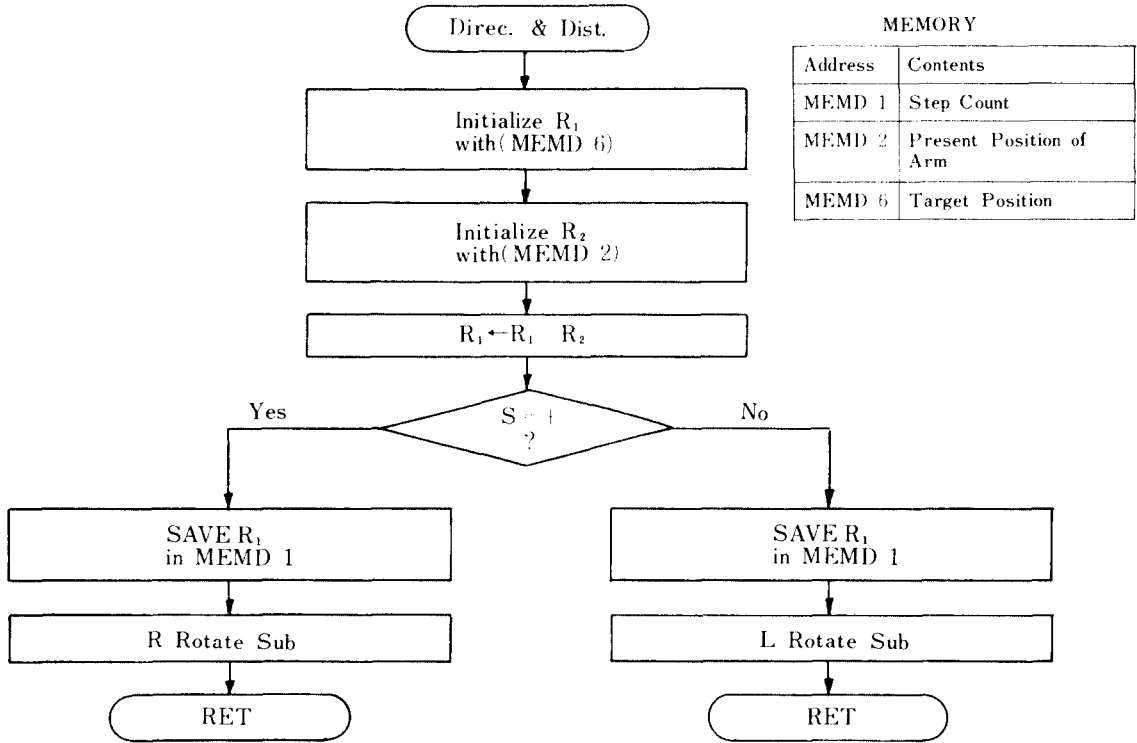


그림13 方向과 거리 서브루틴
Direction and distance subroutine.

感知시켜 位置정보를 주고 出力은 B port 로 보내 電流增幅을 시키기 위해 Q₁, Q₂, Q₃, Q₄ 는 Darlington TR MPSA 14를 사용하고 Q₅, Q₆, Q₇, Q₈은 2 SD635를 사용했다. 應答特性을 관측하기 위해 정밀한 角度的 눈금을 그려놓은 半徑10cm의 圓板을 모터의 軸에 닿지 않도록 끼우고 모터의 몸체에 固定시키고 軸에 指針을 부착시켜 rotor의 回轉角을 測定할 수 있게 했다. 入力 port A로 位置정보를 주고 two phase mode 로 그 對應結果를 관측하였다. 入力信號는 8 bit 로 데이터 A₇A₆A₅A₄A₃A₂A₁A₀에서 A₇은 回轉方向을 나타내며 1인 경우 시계方向, 0인 경우 반시계 방향으로 回轉하게 하였으며 A₆A₅A₄A₃A₂A₁A₀에 의해 거리정보를 주고 step당 遲延時間은 0.1초로 하여 實驗한 結果만족할만한 結果를 얻었다. 實驗데이터는 表 1에 나타내었다. 最大速度로 動作시킬 경우 step당 遲延時間은 0.0068(sec)이었으며 速度는 49(rpm)이다. 이 경우 stator A1에 걸리는 電壓과 電流의 波型은

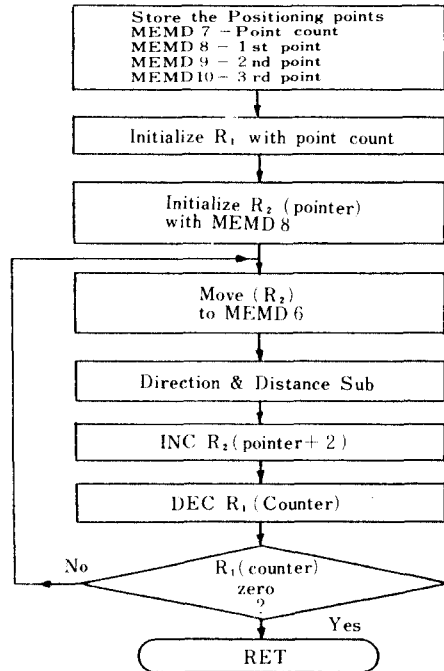


그림14 連續적인 位置制御 서브루틴
Sequential positioning subroutine.

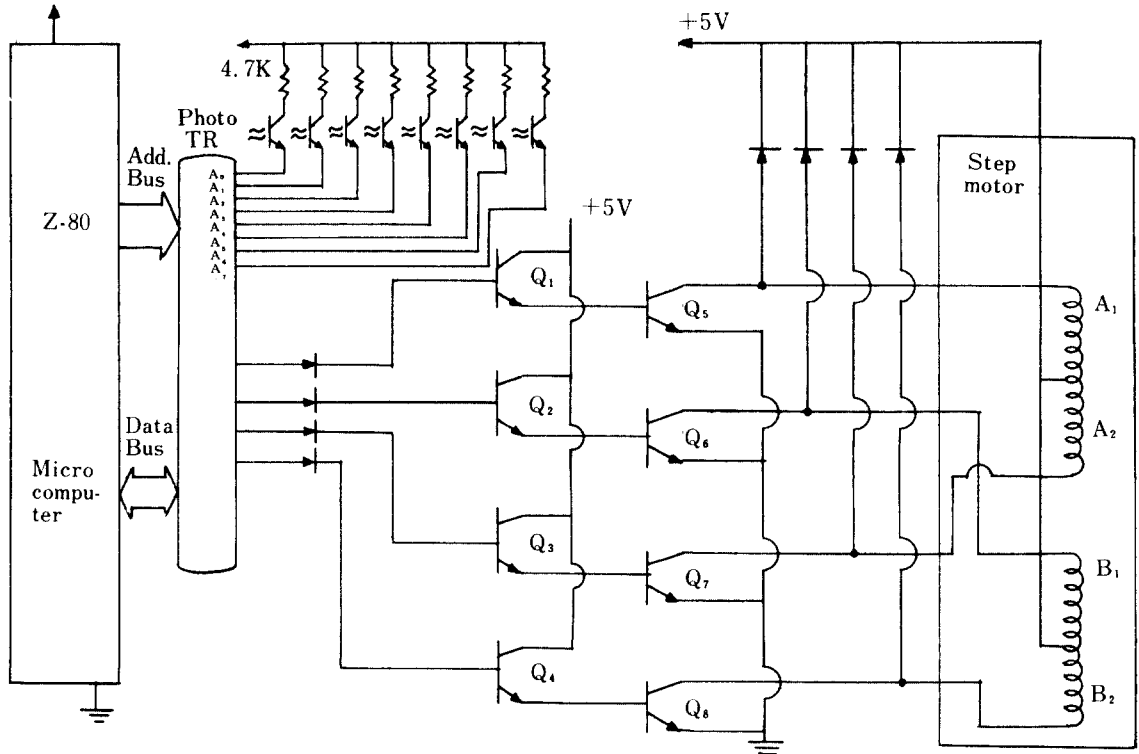


그림 15 實驗 回路
Experimental circuit.

표 1 實驗 데이터
Experimental data.

入 力 信 號	Phase Angle	Phase state
0000 0001	2°	S ₁
0000 0010	4°	S ₂
0000 0100	8°	S ₄
0000 1000	16°	S ₄
0001 0000	32°	S ₄
0010 0000	64°	S ₄
0100 0000	128°	S ₄
0111 1111	254°	S ₃
1000 0001	- 2°	S ₄
1000 0010	- 4°	S ₃
1000 0100	- 8°	S ₁
1000 1000	- 16°	S ₁
1001 0000	- 32°	S ₁
1010 0000	- 64°	S ₁
1100 0000	-128°	S ₁
1111 1111	-254°	S ₂

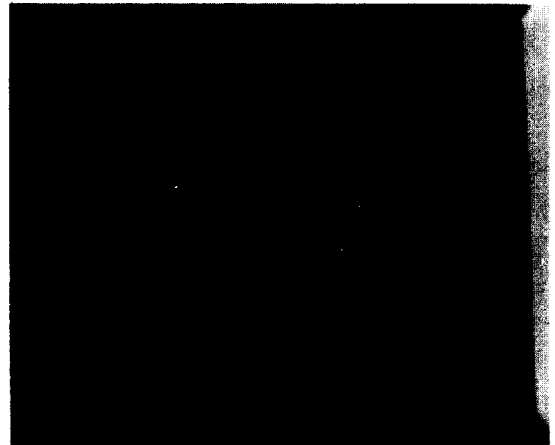


그림 16 stator A1의 電壓과 電流의 波型 (5V/div, 2msec/div)
Waveform of voltage & current by stator A1
(5V/div, 2msec/div).

그림 16과 같다.

또 stator A1, B1, A2, B2 사이에 인가되는 電壓의 位相關係를 관찰하기 위하여 stator A1, B1 사이의 電壓波型과 A₂, B₂ 사이의 電壓波型을 觀察한 結果 位相關係가 그림 4와 같음을 볼 수

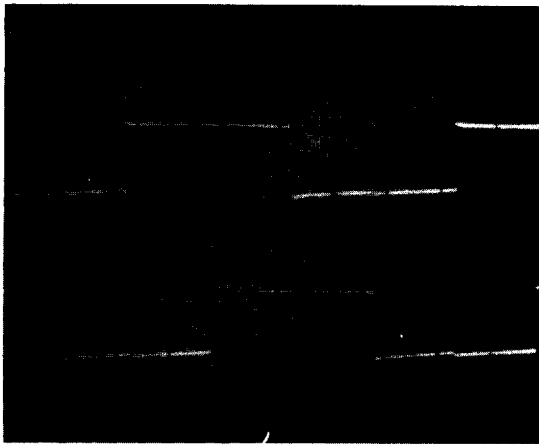


그림17 Stator A1, B1사이의電壓波型(5V/div, 2msec/div)
Voltage waveform of stator A1, B1(5V/div, 2msec/div.)

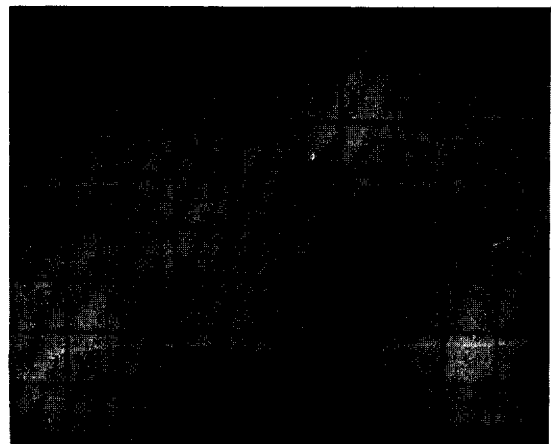


그림18 stator A2, B2사이의電壓波型(5V/div, 2msec/div)
Voltage waveform of stator A2, B2(5V/div, 2msec/div)

있었다. 波型을 각각 그림17과 그림18에 보인다. 最大速度 以上の 速度에서는 制御信號에 스텝 모터가 對應하여 回轉하지 못하는 것이 관찰 되었다. 또 遲延時間을 0.0068[sec]에서 增加시키 가며 stator에 흐르는 電流를 one phase mode와 two phase mode에 대해 實驗하였으며 그實驗 데이터를 表 2에 나타내었다.

표 2 one phase mode와 two phase mode의 stator電流
Stator Current of one phase mode & two phase mode.

遲延 時間 (sec)	one phase mode	two phase mode
0.0068	0.91[A]	1.59[A]
0.0136	1.22	2.15
0.0204	1.27	2.32
0.0272	1.32	2.43
0.0334	1.38	2.47
0.0408	1.40	2.53
0.0476	1.42	2.55
0.0543	1.43	2.56
0.0612	1.43	2.56
0.0668	1.46	2.60
0.0748	1.47	2.63
0.0816	1.47	2.63

이 結果 速度가 빠른 경우 電流가 적고 step당 遲延時間이 0.0748(sec)以上인 경우 one phase mode인 경우 1.47[A], two phase mode인 것

우 2.63[A]의 일정한 값으로 나타났으며 two phase mode의 경우가 動作이 安定되고 토오크가 크므로 位置制御에 적합함을 確認하였다.

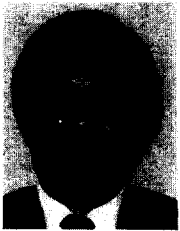
6. 結 論

最近에 들어 많은 디지털시스템이 開發되고 있으며 마이크로컴퓨터가 그중의 한부분으로 중요한 役割을 하고 있다. 그러나 마이크로컴퓨터의 活用性은 소프트웨어의 開發이 先行되었을 때 무한한 領域으로 擴大될 수 있다. 지금까지 本論文에서는 이러한 마이크로컴퓨터를 使用한 PM 스텝모터의 位置制御시스템을 構成하는데 있어서의 소프트웨어의 開發을 研究檢討하여 왔다. 既存의 복잡한 logic circuits를 제거하고 프로 그램으로서 그 logic function을 대신하도록 構成하여 시스템構成이 簡單해졌으며 시스템프로 그램을 變化시킴에 따라 시스템의 확장및 改造가 容易하게 되며 디지털制御시스템을 單一집으로 集積시킨 전용프로세서의 開發이 可能하게 되어 NC, 로봇, 프린터 등 여러 分野에 應用될 수 있을 것으로 期待된다.

参 考 文 献

(1) A. G. Thomas & J. E. Fleischer, "The power stepping motor a new digital actuator," Control Eng., vol. 4, pp. 74-81, January 1957.

- (2) T. R. Fredriksen, "Application of the closed-loop stepping motor," IEEE Trans. Automatic Control, vol. AC 13, no. 5, pp. 464-474, October 1968.
- (3) T. R. Fredriksen, "Design of digital control system with step motors," Symposium proceedings incremental motion control systems and devices. University of Illinois, pp. 290-315, March 1972.
- (4) D. J. Robinson & C. K. Taft, "A dynamic analysis of magnetic stepping motors," IEEE Trans. Industrial Electronics and Control Instrumentation, vol. IECI-16, pp. 111-125, September 1969.
- (5) Michael Andrews, "Programming microprocessor interface for control and instrumentation," Prentice-Hall, Inc. Engle wood cliffs New Jersey 07632, pp. 132-210, 1982.
- (6) P. A. Mohlbacher, "Coding scheme for efficient storage of stepper motor profiles," IBM Tec. Disclosure Bulletin, vol. 22, no. 5, pp. 215-224, October 1979.
- (7) Rodney Zaks, "Programming the Z-80," Zilog Inc. 1980.



成源基(Weon Gi SEONG) 正會員
1956年2月9日生
1975年3月~1980年2月:中央大學校電子工學科卒業(工學士)
1982年3月~現在:東國大學大學院電子工學科(博士課程)
1984年3月~現在:中央大學校電子工學科(講師)



崔宗秀(Jong Soo CHIO) 正會員
1949年5月3日生
1968年3月~1975年2月:仁荷大學校電氣工學科卒業(工學士)
1978年7月~1981年3月:日本Keio大學院卒業(工學博士)
1981年3月~1981年9月:日本Aloka(株)研究所 研究員
1981年10月~現在:中央大學校電子工學科 助教授



河龍守(Yong Soo HA) 正會員
1932年12月22日生
1952年~1956年:서울大學校電氣工學科卒業(工學士)
1982年~1984年:崇田大學校大學院卒業(工學碩士)
1972年~現在:中央大學校工科大學長