

## 직교 좌표계형 Robot 의 구현에 관한 연구

전산 정보 처리과 박 해 암  
전 임 강 사

### I. 서 론

타이어는 그 구조적 특성에 의해 주행성, 제동성, 안전성등이 결정되므로 이를 조사, 분석하여 설계와 생산에 활용함으로써 보다 우수한 타이어를 얻을 수 있다.

타이어는 운용상 특성으로 정지상태에서는 역학적, 구조적인 특성을 조사, 분석할 수 없고 실제 운용하는 상황에서 이루어져야 한다.

이상적인 방법으로는 타이어의 표면을 작은 격자로 나누어 각 격자에 가속도계 등의 센서를 부착하여 회전상태에서 발생하는 원심력과 물체의 고유 주파수나 음압을 측정하여야 하지만 여러가지 기술적인 문제점이 따른다. 그러나 sound intensity mapping 방법을 이용하여 회전상태의 타이어에서 발생하는 소음을 측정하여 구조적 특성에 의해 발생하는 소음의 분포를 이용하여 소음원에 대한 원인을 쉽게 파악할 수 있다.

Metravib co.에서는 측정 타이어면 주위에 micro-phone array 를 폐곡면으로 구성하여 소음의 분포를 측정하고 있으나 micro-phone array 가 벽을 이루고 있어 소음이 반사나 회절을 일으켜 원음을 왜곡시킬 수 있다.

반면 그림 1 에서와 같이 무음향실에서 micro-phone 1 개만으로 측정좌표를 scanning 하도록 하면 이런 문제점을 해결할 수 있고 측정좌표를 세분화하면 보다 사실적인 측정을 수행할 수 있을 뿐만 아니라 다수의 micro-phone 을 사용하면 각 micro-phone 의 주파수 응답특성에 의한 오차로 발생하는 편차를 없앨 수 있다.

그러나 Metravib co.에서 시행하고 있는 방법에 비해 측정시간이 많이 소요되는 단점이 있지만 좌표의 이동과 소음의 측정 및 분석하는 시스템을 분산 처리 시스템으로 동기화 시키면 이를 보완할 수 있다.

일반적으로 로봇트는 사용목적에 따라 여러가지 형태와 기능을 갖고 있으며 산업현장에서 인간을 대신해서 산업활동에 참여하는 기계적인 매니퓰레이터 (manipulator) 를 “산업용 로봇트” 라고 한다.

산업용 로봇트는 모양과 구성면에서 분류할 때 polar 형, cylindrical 형, 직교좌표계형, jointed-arm형 등이 있고, 구조면에서 볼 때 base, body, arm, wrist, 그리고 물체에 직

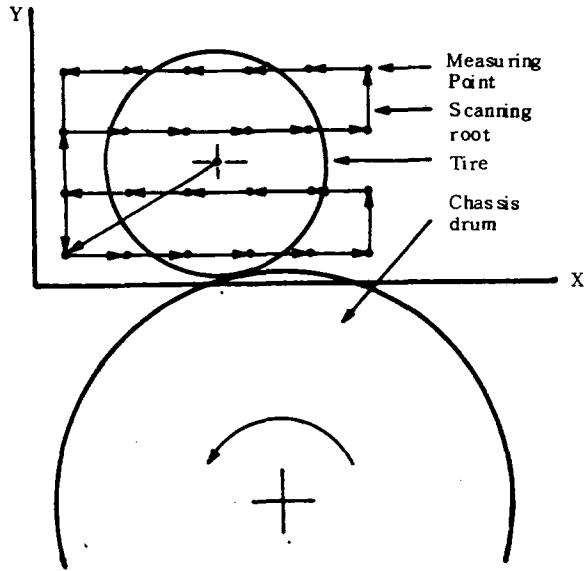


그림 1. Scanning 에 의한 측정

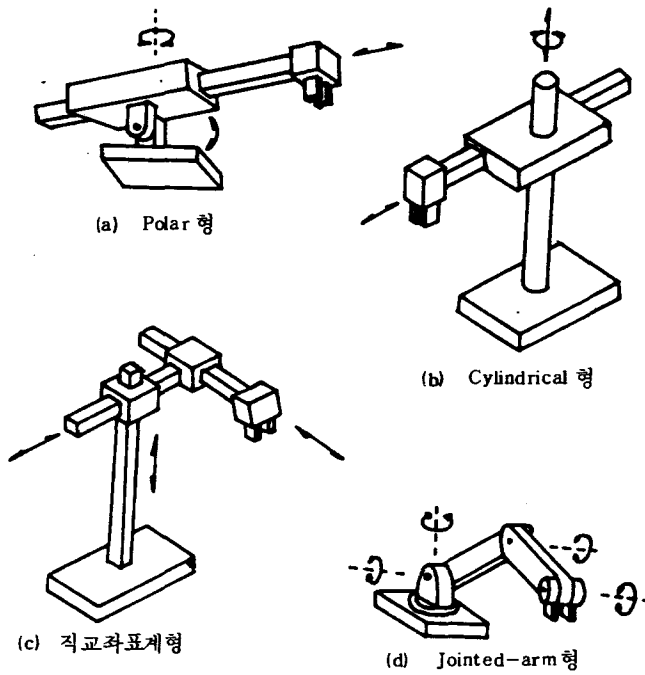


그림 2. 산업용 로봇의 기본형

접 작용하는 end-effector 와 이들이 결합하여 이루는 joint 등이 있다.

운동면에서는 회전운동 (revolute motion) 과 미끄럼 운동 (prismatic motion) 으로 나눌 수 있다.

기본적인 4 가지 형태의 로봇트는 기구적 구성요소로 인해 응용면에서는 각기 장점을 갖는데 반복운동도 (repeatability) 에는 직교좌표계형이, 도달면 (reach) 에는 polar 형과 jointed-arm 형이, 운송과 승강작업에는 cylindrical 형이 많이 이용된다.<sup>2)</sup>

본 연구에서는 그림 1 에서와 같이 타이어의 중심좌표와 각 축에 대해 측정하고자 하는 좌표의 point 갯수를 입력하여 각 좌표의 좌표값과 순서를 결정하고 이를 scanning 하면서 소음을 측정하는 직교좌표계형의 robot 를 IBM PC/AT (12MHZ) 와 Z80 의 co-operation 에 의해 설계, 제작하여 좌표이동의 정확성에 대하여 실험하고 검토하고자 한다.

## II. System 분석

### 1. 좌표계의 고찰 및 평면좌표의 설정과 분석

좌표계의 공간상에서 두점 사이에 메니플레이터가 수행할 수 있는 경로 (path) 는 무한히 많으나<sup>1)</sup> 예측과 가시화의 용이성, 광범위한 응용성, 가장 짧은 거리의 제공, 적은 관성력 등에서 얻는 잇점등의 이유로 직선경로 (stright line path) 에 대한 연구가 많이 진행되고 있고 또한 실용화되고 있으나 수행하도록 제어하는 것은 어려운 문제로 간주되고 있다.<sup>2)</sup>

카테시안 (cartesian) 공간상에서 임의의 두점  $X_i$  와  $X_{\phi}$  사이의 직선경로는 식 1 ~ 식 3 과 같이 주어진다.<sup>1)</sup>

식 1 ~ 식 3 에서  $P(\lambda)$  와  $R(\lambda)$  는 위치벡터와 방위벡터이며  $\lambda = (t/t_d)$  는 normalized 시간,  $t_d$  는 주행시간을 나타낸다.

$$\hat{X}(\lambda) = \begin{bmatrix} R(\lambda) & P(\lambda) \\ 0 & 1 \end{bmatrix} \dots\dots\dots (1)$$

$$P(\lambda) = P_i + \lambda(p_{\phi} - p_i) \dots\dots\dots (2)$$

$$R(\lambda) = R_i \text{ Rot}(k, \lambda\theta) \dots\dots\dots (3)$$

벡터  $K$  와 각도  $\theta$  는  $\text{Rot}(K, \theta) = R_i^{-1} \cdot R_{\phi}$  에서 얻어지며  $X_i$  와  $X_{\phi}$  에 대응하는 조인트값을  $q_i$  와  $q_{\phi}$  라면 두 조인트값들을 선형적으로 보간시켜준 조인트값  $\hat{q}(\lambda)$  와 공간상의 경로  $\hat{x}(\lambda)$  는 식 4 와 식 5 같이 쓸 수 있다.

$$\hat{q}(\lambda) = q_i + \lambda(q_{\phi} - q_i) \dots\dots\dots (4)$$

$$\hat{x}(\lambda) = \begin{bmatrix} \hat{R}(\lambda) & \hat{P}(\lambda) \\ 0 & 1 \end{bmatrix} \dots\dots\dots (5)$$

여기서  $\hat{q}(\lambda) \in R^n$  (n자유도),  $\hat{p}(\lambda) \in R^3$ ,  $\hat{R}(\lambda) \in R^{3 \times 3}$ ,  $f(\cdot)$ 는 kinematics 방정식을 나타낸다.

두점  $x_i$ 와  $x\phi$ 의 조인트값  $q_i$ 와  $q\phi$ 를 선형적으로 보간시켜 보면 매니퓰레이터의 end-effector는 원래 설정했던 카테시안 직선경로에서 이탈하게 된다. 이에 발생하는 오차는 위치오차  $e_p(\lambda)$ 와 방위오차  $e_R(\lambda)$ 로 구별되며 이를 식 6, 식 7과 같이 쓸 수 있으며 좌표상으로 표현하면 그림 3과 같다.<sup>1),3),4)</sup>

$$e_p(\lambda) = | \hat{p}(\lambda) - p(\lambda) | \dots\dots\dots (6)$$

$$e_R(\lambda) = | \text{angle part of } R^{-1}(\lambda) \hat{R}(x) | \dots\dots\dots (7)$$

본 연구에서는 공간상의 좌표계가 아닌 평면상의 좌표계이므로 방위오차는 고려하지 않아도 무방하며 좌표이동시 허용된 위치오차는 최대 이동거리 2000mm를 1회 주행하였을 때 0.1mm로 위치오차의 누적은 허용된다는 가정하에서 임의의 두 좌표  $P_1$ 과  $P_2$ 를 설정하여  $P_1$ 에서  $P_2$ 로 이동한다고 가정하면 이를 평면좌표계에 그림 4과 같이 도식화할 수 있다.

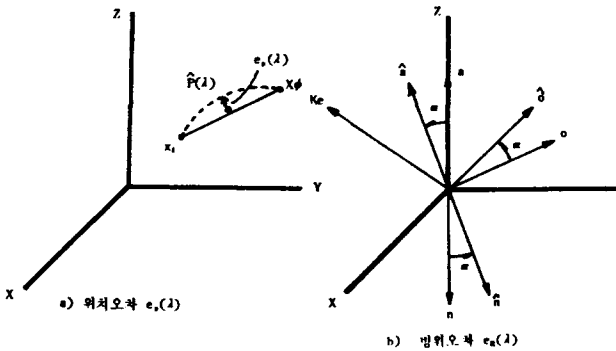


그림 3. 위치오차와 방위오차의 정의

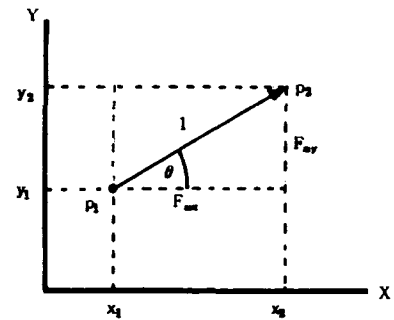


그림 4. X, Y 평면상에서 좌표의 이동

$P_1$ 에서  $P_2$ 로 이동하려면 각 축에 대한 방향(move-Direction; Dm)과 거리(move-difference; Fm)가 구해져야 되고, 속도(move-Velocity; Vm)는 각 축이 동시에 출발하여 도착하므로써 각 축에 대한 속도를 최적화 할 수 있다.

방향은 각 축에 대해  $x_2 - x_1$ 과  $y_2 - y_1$ 의 부호에 따르면, 거리는  $Fmx = |x_2 - x_1|$

과  $F_{my} = |y_2 - y_1|$ 의 값으로 결정된다. 또한 속도는 직선 L에 대한 기울기의 절대치로  $F_{mx}$ 와  $F_{my}$ 중 큰 값을  $F_{mh}$ , 작은 값을  $F_{ml}$ 이라 하고 최대속도  $V_{max}$ 를  $F_{mh}$ 축에 할당하고  $F_{ml}$ 축에는  $(F_{ml}/F_{mh}) * V_{max}$ 로 얻을 수 있다.

### 2. Stepping motor의 특성

스텝 모터의 회전 원리는 그림 5에서와 같이 고정자 철심에 감겨있는 코일  $L_1$ 에 전류가 흐르게 하면 고정자가 자화되고 영구자석으로 된 회전자자의 자료가 고정자의 자로와 일치할 때까지 이끌려서 회전하게 된다.

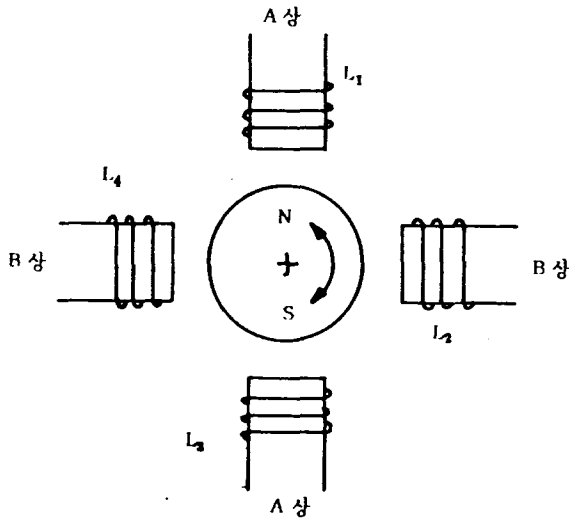


그림 5. Stepping Motor의 구동원리

또  $L_1$ 의 전류를 끊고  $L_2$ 에 전류를 흐르게 하면 회전자는 다시  $L_2$ 의 고정자의 자로와 일치할 때까지 이끌려서 회전하게 된다. 그림 5의  $L_1 \sim L_4$ 에 각각 1회씩 순차적으로 전류를 흐르게 하면  $90^\circ$ 씩 회전하여  $360^\circ$ 를 회전하게 된다. 여기에서 고정자와 회전자의 자극을 세분화하여 1 펄스당  $0.36^\circ \sim 1.8^\circ$ 까지 회전 각도를 얻을 수 있고 식 8과 같이 표현한다.<sup>6)</sup>

$$\alpha = \frac{360^\circ}{m \cdot nr} \dots\dots\dots (8)$$

$nr$  ; 相當 자극의 갯수,  $m$  ; 相數,  $\alpha$  ; 1 펄스당 회전각도.

스텝 모터의 구조적 측면에서 VR형 (Variable Reluctance Type), HB형 (HyBride type)

PM형 (PerManent type) 으로 분류하며, 여자방식에 따라 1상형, 2상형, 1-2상형, 3상형, 5상형 등으로 나눈다.<sup>5)</sup>

범용성을 갖는 2상형 스텝 모터에서 구동 펄스와 각 상의 timing chart의 관계를 그림 6과 같이 표현할 수 있으며 이를 16진수 값으로는 .....6H → 5H → 9H → AH → 6H .....로 쓸 수 있다.

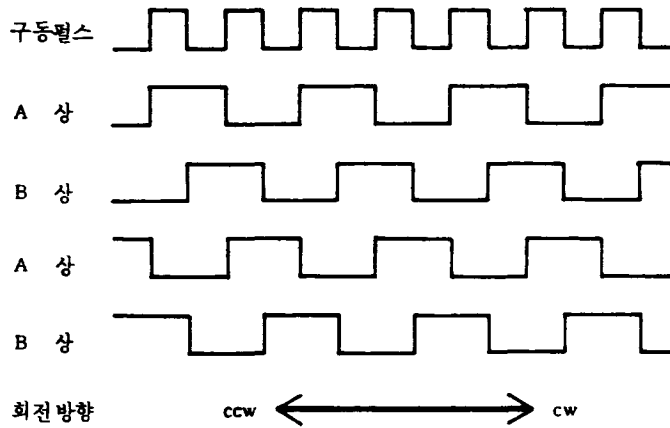


그림 6. 2상 스텝 모터의 Timing chart

스텝 모터에 한 개의 구동펄스가 가해진 후 안정을 이룰 때까지의 동적상태를 그림 7과 같이

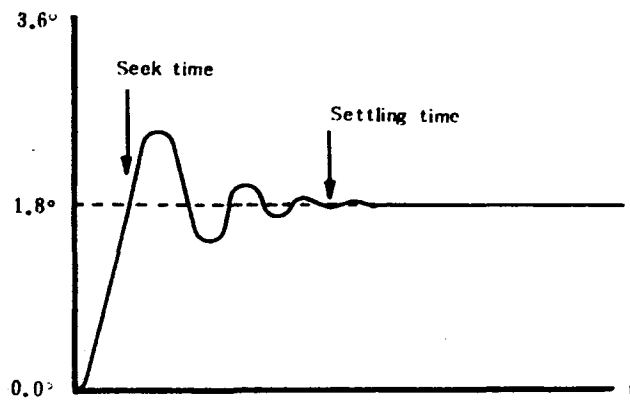


그림 7. 스텝 모터의 동적 특성

그릴 수 있으며 스텝 모터가 구동하여 정해진 회전각도를 처음으로 통과할 때까지의 시간(seek time)으로 부터 안정될 때까지의 시간(settling time)이 필요하고 안정된 후에 회전각도에 대한 오차는 보통 1% 이내이다. 반면 settling time 이전에 또 다른 구동 펄스를 인가하면 응답특성은 급격히 나빠지며 구동 펄스의 주기가 빨라지면 회전하지 못하고 oscillation을 일으킨다. 즉 settling time 보다 긴 주기로 구동펄스를 인가하면 1회전당 오차는 약  $1/3000^\circ$  이하로서 encoder 없이 정밀제어가 가능해진다.<sup>6)</sup>

### 3. 제어데이터와 처리데이터의 구성

그림 4에서 좌표 P<sub>1</sub>에서 P<sub>2</sub>로 이동하는데 필요한 거리, 방향, 속도의 자료인 제어데이터와 측정된 자료를 분석하기 위한 처리데이터로 구분할 수 있다.

본 연구에서 end-effector가 절대좌표의 원점에서 종점까지의 거리는 2000mm이며 이동기구로서 2상 스텝 모터와 pitch 간격이 10mm인 ball screw를 사용하므로 구동펄스의 최대값은 40,000으로 16bit의 길이를 갖으며 한개의 구동펄스로 이동할 수 있는 거리는 0.05mm이다.

방향은 좌우로 회전하므로 1bit이며 속도는 스텝 모터의 특성상 settling time을 짧게 하면 oscillation을 일으키게 되므로 최대 8bit인 255를 초과할 수 없게 된다.

처리데이터로는 선정된 좌표에서 Sampling한 tire noise에 관한 데이터와 그 절대좌표로 구분할 수 있다.

측정할 noise는 음압으로서 불규칙적이므로 1회 sampling만으로는 충분치 않고 수 회 sampling하여 산술평균한 값을 취하는 것이 타당하므로 매번 sampling한 자료를 보관할 장소와 산술평균한 자료를 보관할 장소가 필요하게 된다.

위에서 검토한 자료들에 대하여 논리적인 record로 설계하여 표현하면 그림 8과 같다.

좌표 데이터	index No.	Position X	Position Y						
제어 데이터	index No.	Drive Pulse X	Drive Pulse Y	Direction X	Direction Y	Speed X	Speed Y		
Noise 데이터	index No.	Noise 1	" 2	" 3	" 4	" 5	" 6		
	" 7	" 8	" 9	" 10	Average Noise				

그림 8. 제어 및 처리 데이터

### Ⅲ. System 구성과 Control flow

#### 1. Block diagram과 Control flow

2절에서 검토한 바와 같이 처리데이터의 획득 및 처리하는 작업과 측정하고자 하는 좌표의 자료인 제어데이터를 취급하는 것으로 분류된다.

본 연구에서 Host는 제어데이터의 생성, 관리와 이동할 좌표의 자료를 각 축에 전달하는 일 외에 A/D converter로부터 noise 자료를 획득하고 관리, 분석, 보관하는 일을 담당한다.

Channel로 구성된 각 축은 Host로부터 좌표의 자료를 전달받아 축을 이동시키며 축의 이동상태, over run과 같은 위험한 상태 및 사용자의 interruption등을 감지하여 이에 필요한 조치를 취하고 이를 Host에 보고하는 등 작업을 분담함으로써 Host의 처리능률을 향상시킬 뿐만 아니라 각 장치의 module화를 가능케 한다.

이와 같은 작업을 분담하는 목적에 부합하도록 system을 구성하면 그림 9와 같다.

Host는 key-board로부터 측정할 tire의 규격과 각 축에 대해 등분할 숫자를 입력받으면, 규격으로 각 축의 길이를 계산하고 등분값에 의해 측정할 위치의 상대좌표와 갯수를 얻는다. 또한 tire의 중심좌표를 입력받아 각 축의 하한선과 상한선의 좌표값과 그림 1에서 좌표 데이터인 측정할 각 좌표값들과 제어데이터의 값들을 산출한다. 여기에서 drive pulse는 상대위치에 대한 변위량이며 이 값으로 방향과 속도의 자료를 얻고 A/D converter로부터 얻어질 자료인 noise값 즉 처리데이터를 보관할 배열을 마련한 후 첫번째 좌표값과 제어데이터를 각 channel에 전달하고 이동할 것을 지시한다. 이때 channel은 축을 즉시 이동시키기 시작하여 완료한 후 이를 host에 보고한다. 이동완료 보고를 받은 host는 A/D converter의 작동을 개시하여 자료 획득을 완전하게 완료한 후 다음 좌표값과 제어데이터를 각 channel에 전달하고 이동할 것을 명령한다.

축이 이동하는 동안에 앞에서 획득한 자료로 평균값과 좌표간의 noise에 관한 사이값을 보 간법에 의해 구하는 등 자료처리를 행하며 그 결과를 보조기억장치인 HDD에 보관한다.

이와 같은 System flow는 그림 10과 같고 자료전달 및 이동명령을 수행하는 subroutine은 그림 11과 같다.

한편 host로부터 자료를 전달받아 저장하고 수행을 담당하는 channel의 세부적인 작업으로는 축을 지정된 좌표로 지정된 속도에 의해 이동하는 것과 축의 over run, 사용자 또는 system의 interruption을 감지하는 일 외에 현재 주행중인 좌표 즉 남아있는 거리와 이동 완료등의 신호를 host에 보고하는 것으로 생각할 수 있다. 이를 flow chart로 그리면 그림 12와 같다.

Subroutine으로 구성된 driving routine에서는 CTC로부터 interrupt요청이 있으면 좌표값 즉 driving pulse 값을 1 감소하고 그림 9의 monostable에 한개의 pulse를 인가하



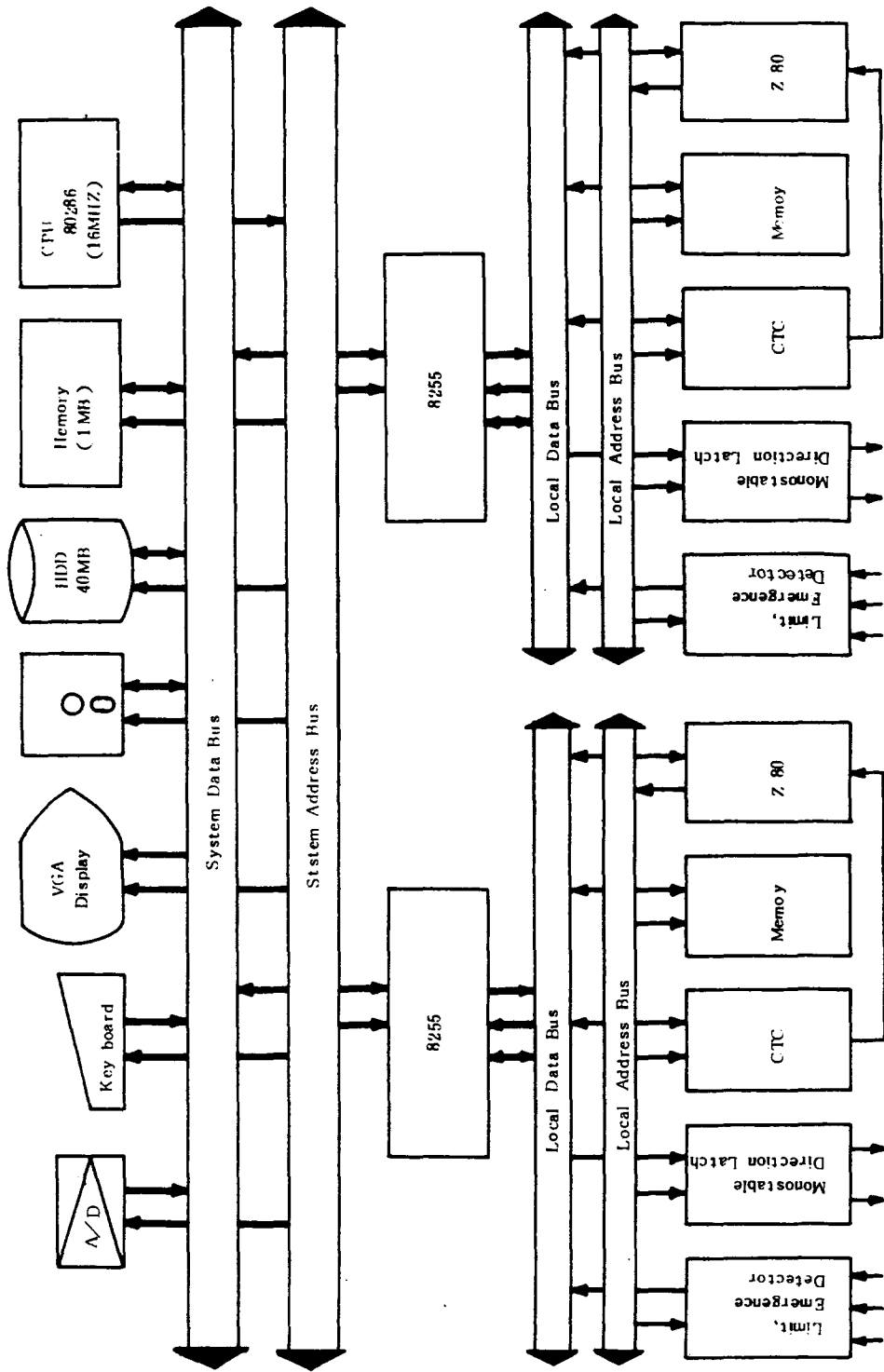


그림 9. 시스템의 구성

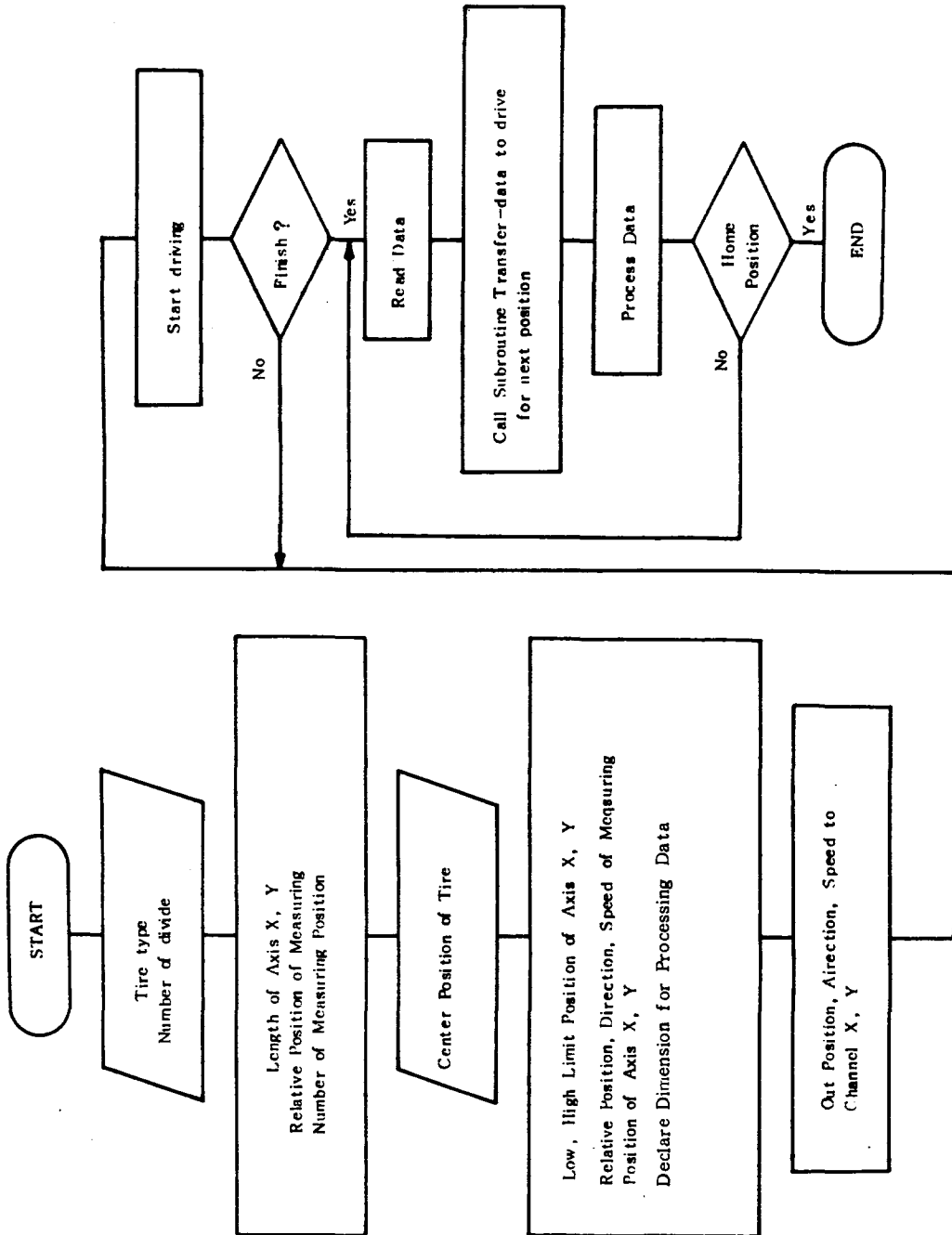


그림 10. 시스템 flow Chart

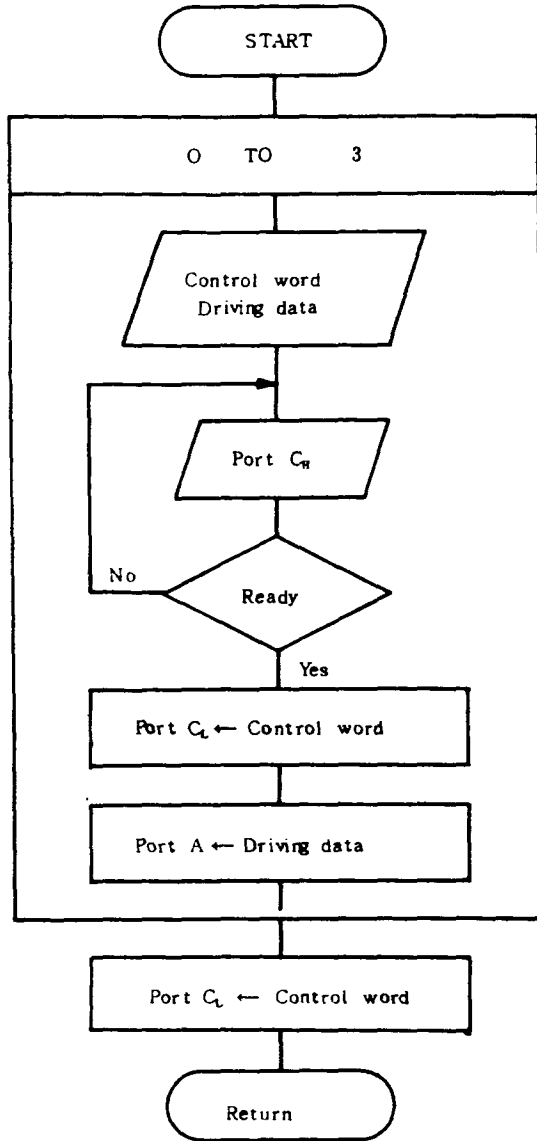


그림 11. Host로부터 Channel에 자료전달 flow chart

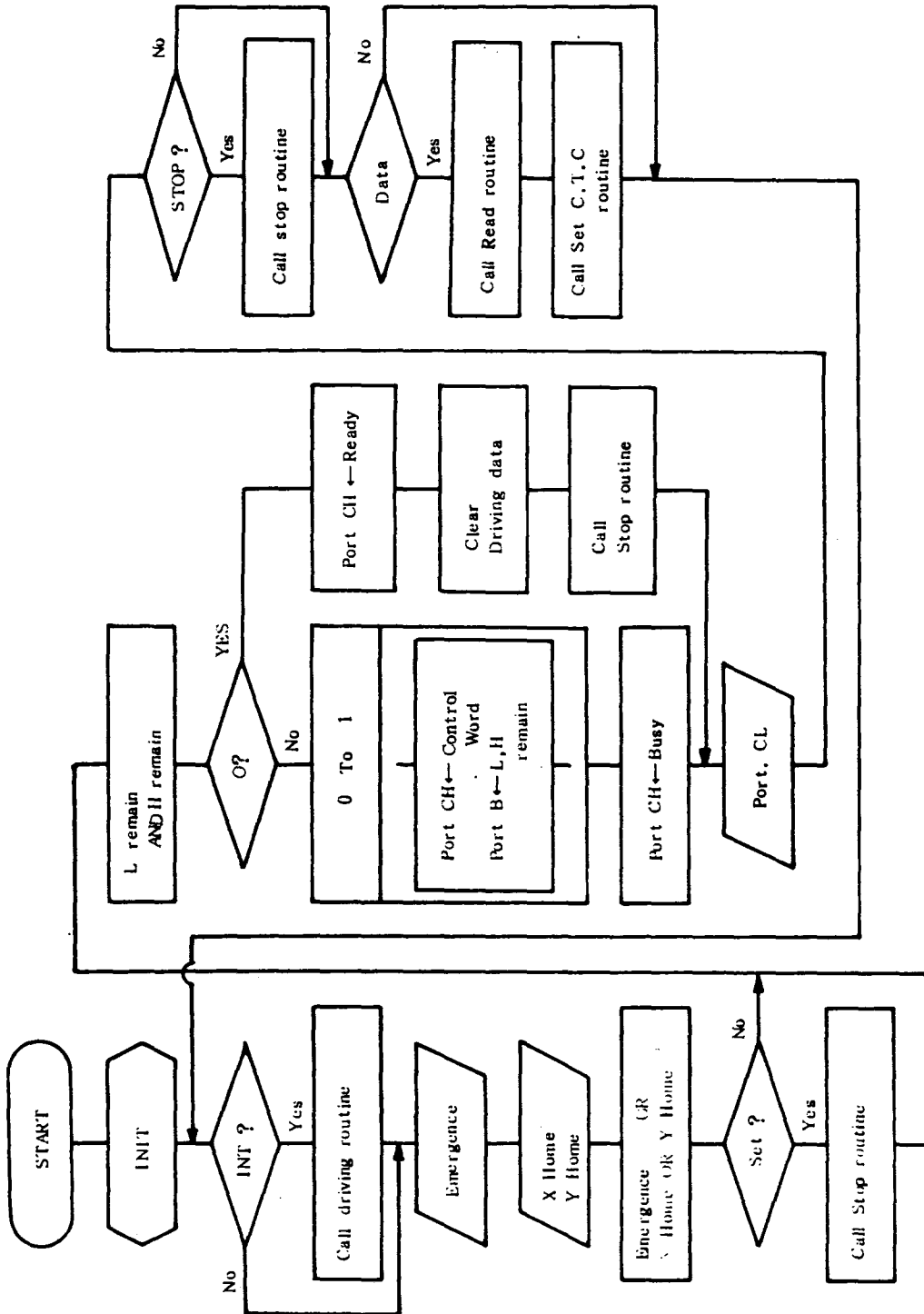


그림 12. Channel 의 flow chart

따라서 motor 구동회로 상태를 변환시켜 motor가 정해진 각도 만큼 회전하게 한다. 또 stop routine은 driving pulse 값을 stack에 push하고 CTC reset 및 interrupt를 disable한다. Read routine은 host에서 지령한 자료를 받아들여 보관하고 CTC에 set할 자료로 활용한다.

Channel이 수행하는 작업의 일괄적인 flow에 대해 검토하면, 초기에는 모든 자료가 0이므로 CTC 역시 아무런 동작을 하지 않아 interrupt를 요청하지 않고, motor 역시 정지되어 있으므로 home detector는 reset된다. 이동할 좌표가 없으므로 남은 거리 역시 0이므로 Ready 신호를 host에 보내고 좌표값을 0으로 clear함과 동시에 motor도 확실하게 정지시킨다. 이때 channel이 ready상태에 있음을 알아차린 host는 자료와 상태 code를 channel에 보낸다. channel은 상태 code를 분석하여 해당된 동작을 취하거나 할당된 장소에 자료를 저장한 후 CTC에 속도를 time constant로 set하여, set된 시간이 경과할 때마다 interrupt를 요청토록하여 좌표값 즉 상대좌표까지의 driving pulse 값을 감소토록 함으로써 속도와 거리에 맞게 이동한다. 또한 방향자료는 motor driver와 연결된 latch를 set한다. 이와 같은 과정을 반복하여 측정하고자 하는 좌표로 이동하면서 축의 상태를 점검하고 보고한다.

## 2. Channel 및 상태 Code의 설계

Host와 channel 간에 자료를 송수신하기 위한 규칙에 관하여 고찰하면 host에서 channel에 지령하는 것과 channel에서 host로 보고하는 것으로 나눌 수 있다.

먼저 host에서 channel에 지령할 때, 현재 bus 상에 실려있는 자료의 유효여부, 방향, 속도, 거리 및 start 또는 stop을 구분하여 전달하여야 하며, 거리는 최대값이 40,000으로 16bit의 길이이므로 8bit씩 나누어 상위 byte와 하위 byte를 구분할 수 있어야 한다. 또한 channel은 주행중인 축의 좌표값과 상태, 그리고 수행중인지 대기상태인지를 알릴 수 있어야 한다.

t = Channel						Hest = Channel					
1	0	0	0	방	향	x	x	x	0	남은거리 I	
1	0	0	1	속	도	x	x	x	1	남은거리 II	
1	0	1	0	거리	L	x	0	0	x	normal drive	
4	0	1	1	거리	H	x	0	1	x	X home	
0	0	x	x	Stop		x	1	0	x	Y home	
0	1	x	x	Start		x	1	1	x	Emergency on	
1	0	x	x	data		0	x	x	x	Ready	
1	1	x	x	no mean		1	x	x	x	Busy	

그림 13. 송수신 상태 code

이를 code로 작성하여 그림 13과 같이 설계한다.

송수신 시차에 의하여 자료의 오류가 발생치 않도록 자료의 경로와 상태 code의 경로로 구분하여 8255의 port A와 port B에는 자료전달용으로 port C<sub>L</sub> 및 port C<sub>H</sub>에는 축의 이동과 정지 및 상태 code 용으로 할당한다.

### 3. Channel의 구현

앞에서 검토한 바와 같이 channel이 수행하는 기능을 만족하도록 그림 14와 같이 회로를 결선하고 표 1과 같이 주기억장치와 입출력의 memory map을 구성하였다.

표 1. Channel의 Memory Map

Program	0000H ~ 1FFFH	Data	2000H ~ 2FFFH
Initialize	0000H ~ 00FFH	Direction	2000H
Main	0100H ~ 02FFH	Speed	2001H
Read RTN	0300H ~ 03FFH	Drive Pulse L	2002H
Scan RTN	0400H ~ 04FFH	Drive Pulse H	2003H
Stop RTN	0500H ~ 05FFH	Stack	2FFFH
Drive RTN	0600H ~ 06FFH		
BIOS	1F00H ~ 1FFFH		

I/O	CTC	CH0	10H	8255 Port A	20H	Drive Direction	30H
		CH1	11H	Port B	21H	Pulse	31H
		CH2	12H	Port C <sub>L</sub>	22H	Interrupt &	
		CH3	13H	Port C <sub>H</sub>	23H	Limit, Home	32H

CTC는 System clock 1MHz를 pre-scaler, CH0, CH1으로 분주하여 250Hz를 얻고, host로부터 전달받은 속도는 최대치가 FFH이므로 이를 2의 보수로 취하여 CH2의 time constant register에 set하여 counter mode로 설정한다.

Host로부터 start 명령을 받으면 counter가 down count하여 0이 될때 interrupt service를 요청하면 CPU는 interrupt mode 2에 의해 service routine에서 구동펄스를 motor driver에 보내고 거리의 값을 감소한 후 그 결과가 0이면 ready mode로 다음 좌표의 자료와 명령을 받을 수 있도록 한다.

또한 사용자의 interrupt는 NMI mode로 emergence stop에 사용하고 home point sense, limit point touch 등은 interrupt mode 1에 의해 처리한다.

### 4. 실험

본 연구의 직교좌표계형 robot는 IBM PC/AT(12MHz)를 host로하고 channel processor

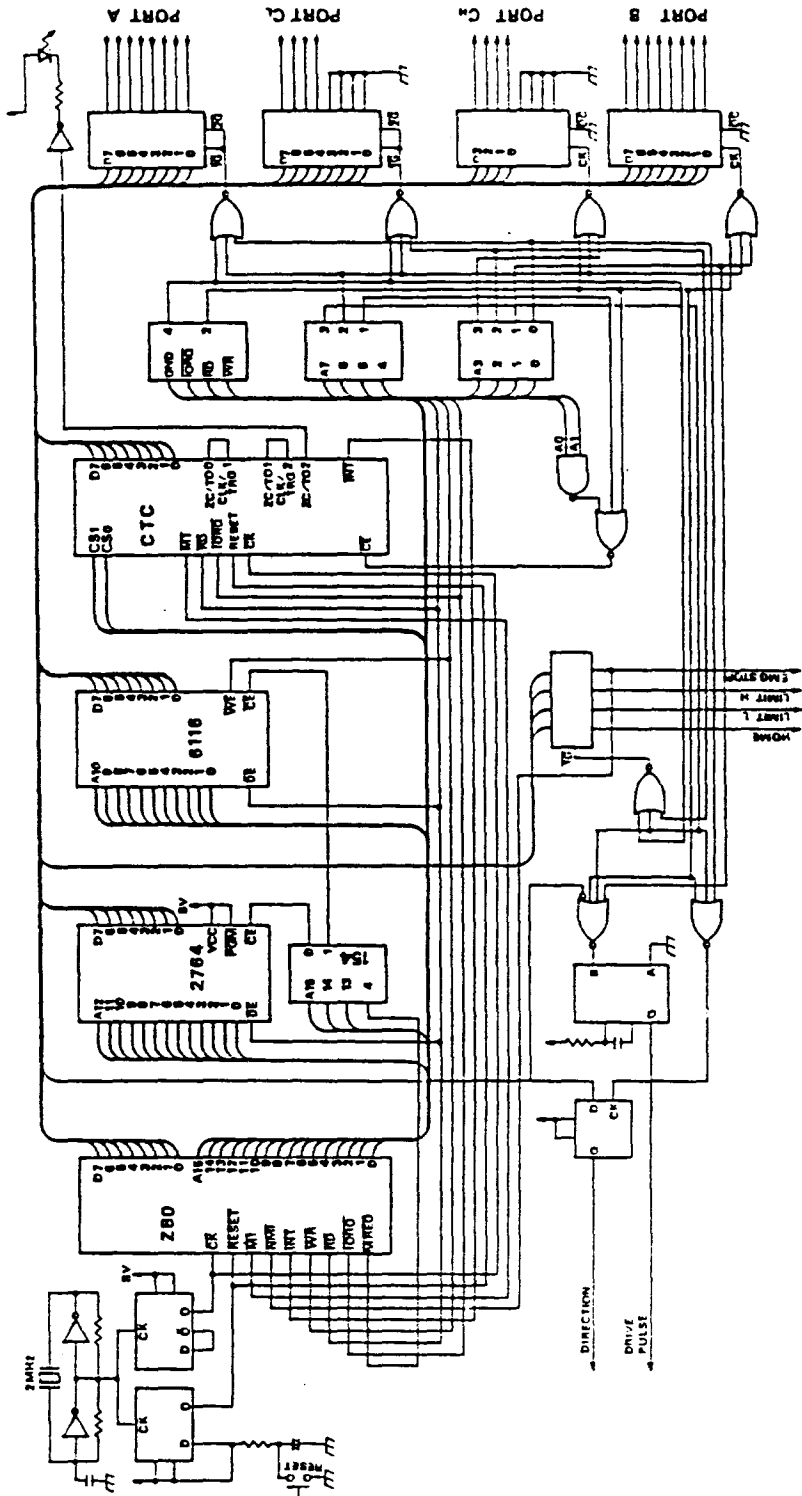


그림 14. Channel 의 구성

는 Z80 으로 구성하였고, host 가 관리하는 8255 와 Z80 이 점유하는 buffer 와 latch 로 interface 하였다.

System 을 관리, 제어, 처리하는 host 의 사용언어로는 Turbo pascal, version 5.0 과 channel 을 작동, 관리하는 Z80 은 Z80 assembly language 를 사용하여 그림 15 와 같이 실험하였다.

본 연구에서 제작한 robot 는 각 축의 주행거리가 최대 2000mm 로 plotting 기구의 설치 어려움으로, 짧은 구간에서는 endeffector 에 pen 을 설치하여 직접 plotting 하였고 전 구간은 vernier caliper 로 실측하였다.

전 구간에 있어 주행오차는 실측상 측정하기 어려웠고 반복주행으로 발생할 수 있는 주행누적 오차도 측정할 수 없을 정도로 적어 실용상 만족할만 하였다. 그러나 타이어가 도우넛 형태이므로 Metravib CO. 에서 시행하는 방법과 같은 곡면에 대한 측정은 할 수 없었다.

앞으로 본 robot 에 Z축과 end-effector revolute 하는 4 자유도로 연구, 보완하면 tire noise 측정에 충분한 성능을 기대할 수 있다고 사료된다.

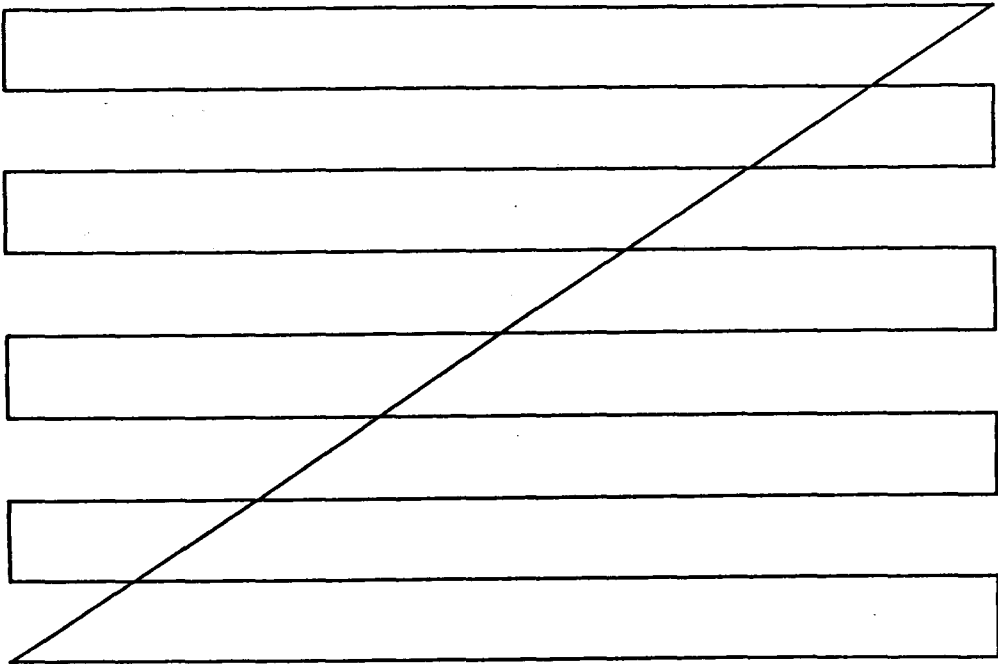


그림 15. 실험 결과



## V. 결 론

자동차용 Tire가 주행 상태에서 발생하는 noise를 측정하기 위해, host로는 IBM PC/AT와 channel processor로는 Z80을 사용하여 co-operation에 의한 직교좌표계형 robot를 분석, 설계, 제작하여 실험하고, 좌표 이동상태와 이동오차에 대해 측정하였던바 다음과 같은 결론을 얻었다.

1. 전 구간에서 이동오차와 주행누적 오차는 실측으로 측정할 수 없을 정도로 적었다.
2. 두 system간의 interface는 8255와 buffer 및 latch, 상태 code만으로도 정확한 송수신이 가능하였다.
3. CTC를 counter로 사용하여 사선주행시 각 축의 주행속도를 최적화할 수 있었다.
4. Co-operation에 의한 channel processor를 modul화하였고 host에서는 channel의 번지만을 할당하므로써 link가 가능하여 다축제어를 용이하게 구성할 수 있다.
5. Host를 관리, 제어하는 언어로써 Turbo Pascal과 channel을 제어하는 언어로 Z80 assembly language를 사용하였다.
6. End-effector가 revolute하고 Z축을 부가하여 4자유도를 갖는 system으로 연구, 보완하면 tire noise를 측정하는데 충분한 성능을 기대할 수 있을 것으로 사료된다.

본 연구를 위해 지원과 협조를 아끼지 않으신 (주) 금호 타이어 기술 연구소의 관계자들과 금천 Engineering CO.의 사장님께 감사사를 드린다.

## 참 고 문 헌

1. 오세현, 전홍일; "로봇 매니플레이터의 카르테시안 직선운동을 위한 경로점의 최소화," 전자공학회 논문집, 25(2), 144~146, (1988)
2. 전홍일; "로보틱스 이론," 전자공학회지, 15(6), 575~576, (1988)
3. 김태윤; 메카트로닉스 산업용 로봇 서울, 도서출판 생능, 128~150, (1991)
4. Wesley E. Snyder; *Industrial Robot Computer Interfacing and Control*, New Jersey, Prentice-Hall, 178~199, (1985)
5. 이종춘; 최첨단 제어용 모우터, 서울, 성안당, 13~39, (1988)
6. 유수엽; "스텝모터의 구동과 그 특성," 전자과학, 29(338), 194~198, (1987)
7. 남궁재찬; Robot공학의 기초, 서울, 기전연구소, (1990)
8. Carl Helmers; *Robotics Age*, New York NY, Robotics Age Magazine, (1981)
9. 임경환; 제어용 로봇, 서울, 집문당, (1988)
10. 한성국; IBM PC XT/AT 기술사전, 서울, 집문당, (1988)
11. 양홍석, 황희욱, 조동섭; 마이크로 컴퓨터, 서울, 동일출판사, (1987)
12. 이영돈, 이상훈, 최인열; Z80 Subroutine, 서울, 성안당, (1985)

**A Study on the Implementation of a Cartesian Coordinate  
Type Robot Using a IBM PC/AT & a Z80 for a Automobile  
Tire Noise Measuring**

**Hae Am, Park**

*Dept. of Computer and Information  
Processing*

*Kwangju Health Junior College*

*>Abstract<*

A general method to measure a noise of a automobile tire is used of a micro-phone array.

This method has the weakness that a walled micro-phone array causes a reflection and a diffraction, and so a sampled noise on a micro-phone array is distorted.

But this weakness is improved by a micro-phone which traces a course to a setted up measuring point on a made up cartesian coordinate.

This method requires that a micro-phone is accurately moved into a setted up measuring point.

In this paper, a cartesian coordinate type robot for automobile tire noise measuring to search a source of noise is implemented by a co-operated with IBM PC/AT (12MHz) as a host and a micro-processor Z80 as a channel for the precise movement.

This designed, manufactured and implemented robot system is experimented.

The result has been obtained the pretty good movement on the X,Y coordinate in the full range 2000mm.