

다중 로봇 협조제어를 위한 Controller의 설계기술

徐一弘*, 呂熙珠*, 金兌垣*, 高德用**, 曹廷鎬**

* 漢陽大學校 電子工學科 知能制御 및 로봇 研究室

** 三星航空 精密器機 研究所

I. 머리말

최근 생산성 향상을 통한 국제 경쟁력 강화를 위해 산업용 로봇을 응용한 다중 로봇 제어 시스템^[1,2]의 개발에 점점 더 관심이 집중되고 있다. 이러한 시스템의 잠재적인 응용이 보다 정교한 조립작업의 자동화를 포함하여 전 공장의 무인화까지를 목표로 하고 있으며 나아가서는 심해작업이나 우주공간에서 작업의 로봇화를 요구하고 있는 실정을 감안할때 이들 요구에 부응하기 위해서 가장 핵심적인 연구인 복수개 로봇 및 다중센서를 제어할 수 있는 다중 로봇 시스템 제어장치의 개발은 반드시 필요하게 될 것이다. 더군다나, 움직이는 장애물과 같은 환경의 변화를 고려할 때, 센서 정보에 의해 환경을 묘사하는 방법^[3,4]이 필요하다. 시각, 거리, 근접, 접촉 및 힘 센서 등과 같은 유용한 센서들에 의해서 다양한 정보를 얻을 수 있다. 따라서, 다중 로봇 제어 시스템을 운영하는 데 있어서 운동 계획, 협조제어 능력과 함께 특히 필요한 기능은 센서 정보의 융합과 통합으로 생각할 수 있다. 이 문제를 해결하기 위해서, 제어 시스템은 Multi-Robot이 협조제어를 할 수 있도록 해야 하고 다중 센서를 융합할 수 있어야 한다. 그러나, 현재 상품화된 거의 모든 로봇 제어 시스템은 기본적으로 사용자가 한 대의 로봇만을 사용할 수 있게 되어있고, 또한 적절한 제어 알고리즘과 계산 방법의 미흡으로 인해 효과적으로 다중 센서를 융합할 수 없다.

본 연구에서는, 다중 프로세서에 근거해서 다중 센서 통합 능력을 가지는 다중 로봇 협조제어 시스템을 구현하는 방법을 설명하고자 한다. 특히, 전체 제

어 시스템은 동시에 12축을 제어할 수 있으며 시각 센서, 힘 센서 그리고 외부 컨베이어의 위치 엔코더와 같은 환경에 적응할 수 있도록 설계했다. 또한, 이 제어 시스템은 사용자가 터치 박스, 풀 스크린 에디터를 가지는 고급 언어 그리고 메뉴 구동명령어—특히, 다중 로봇 제어를 위해 설계된 모든 것—를 경유해서 시스템과 통신을 할 수 있도록 해준다. 또한 본 제어시스템은 장애물 회피, 컨베이어 트래킹 그리고 다중로봇을 제어할 때 필요한 동기 운동과 같은 특수한 기능을 수행할 수 있다.

II. 다중 로봇 제어 시스템의 구성

1. 전체 시스템의 구성

본 연구에서는 다음과 같은 여러 개의 프로세서로 나누어 분산처리 구조를 갖도록 하여 확장성 및 유연성이 높은 시스템이 되도록 구성하였다.

먼저 시스템 전체를 관리하며 언어 및 로봇 동작의 교시 그리고 자기 진단등의 기능을 하는 Supervisory Processor, 물체의 위치와 자세 및 형태를 인식하는 Vision Processor, Man-Machine Interface를 위한 Display Processor, 그리고 로봇의 제어를 담당하는 로봇 제어 Processor 등으로 구성되어 있고, 이들 각각의 통신은 Bus Arbitration에 의한 VME Global Bus를 통해 공유 메모리(Common Memory)를 이용하여 수행한다.

특히 Supervisory System은 Real Time O.S.인 VxWorks를 사용하여 구현하였고 Servo 시스템은 Interrupt 방식을 사용하여 구현하였는 바, 전체 구성은 그림 1과 같다.

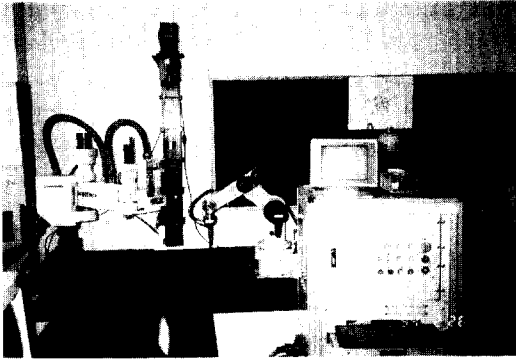


그림 1. 다중로봇을 위한 협조제어 시스템의 전체외관

2. 제어 시스템의 H/W 구성

본 연구에서는 여러개의 프로세서로 나누어 분산처리 구조를 갖도록하여 확장성 및 유연성이 높은 시스템을 구성하였다. 이에 대한 Hardware 구성은 그림 2와 같다.

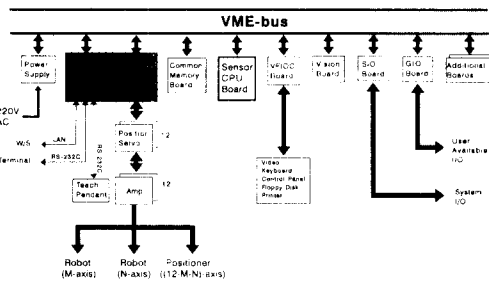


그림 2. 다중로봇 협조제어 시스템의 H/W 구조

3. 제어 시스템의 S/W 구성

개발된 Software는 C언어에 기반을 두고 모든 Program을 세분화하여 계층적 제어구조를 이루게 하고 수정 및 편집을 단위 Program Module별로 할 수 있게 하는 새로운 형태의 매니플레이터 레벨 로봇 제어언어를 설계하였다. 이에 대한 제어 시스템의 S/W 구조는 그림 3과 같다.

Ⅲ. 다중 로봇 제어 시스템의 기능별 구성

1. Supervisory System

Supervisory System을 설계하는 데 있어서, Supervisory는 각 Slave에 작업을 효과적으로 할당

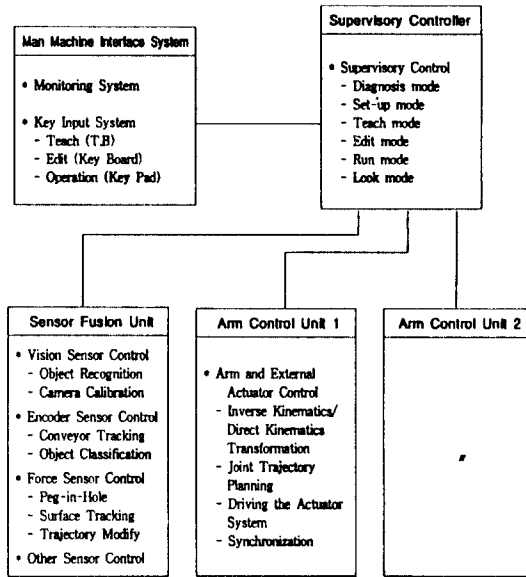


그림 3. 다중로봇 협조제어 시스템의 S/W 구조

하여 수행시켜야 하고, 또한 조작자에게는 로봇 언어와 같은 사용자에게 친숙한 Man-Machine Interfacing을 준비하여 시스템 Set-Up뿐만 아니라 운영상의 실수를 줄일 수 있도록 고려되어야 한다. 또한, 전체 시스템의 원활한 유지를 위해 자기 진단 기능이 필요하다. 이들 요건을 충족 시키기 위해서, 본 시스템에서는 다음과 같이 6가지 모드를 설정하였다.

- (1) Diagnosis Mode
- (2) Set-Up Mode
- (3) Teach Mode
- (4) Edit Mode
- (5) Look Mode
- (6) Run Mode

(1) Diagnosis Mode

진단 모드에서는 우선 Supervisory System을 메인 하드웨어 시스템(CPU Board, Common Memory Board, Motor Servo Modules), 주변 하드웨어 시스템(Floppy Disk Driver, Printer, Teaching Box)등의 상태를 확인하고, 그것들의 현 상태를 표시한다.

(2) Set-Up Mode

Set-Up 모드에서는 응용시에 제어 시스템의 융통성을 부여하기 위해서 여러가지 파라미터를 Set-Up 하거나 수정할 수 있다. 본 시스템에서는 각 로봇 파라미터(로봇 타입, 기구학 변수, Motor 정격), 좌표 원점 파라미터(홈 위치, 속도, 오프셋)와 시스템 파라미터(Base 좌표계, Tool 좌표계, 위치 레벨, 전용 I/O, 외부 장치)를 설정한다. 그래서 이들 입력

파라미터를 조정하여 어떤 Type의 로봇(X-Y-Z 직교형, 수평다관절형, 수직다관절형)도 제어할 수 있다.

(3) Teach Mode

로봇의 End-Effector를 정의된 여러 좌표계를 기준으로 이동시켜, 로봇의 "위치"와 "방향"을 Memory에 기억시키고 각 점의 이름을 할당하여 운동 계획에 사용한다. 본 연구에서는 여러 좌표계(Robot Base Coord, User Defined Coord, Tool Coord)를 준비해서 Teaching을 용이하게 하는 한편, CAD Data를 직접 입력하는 MDI(Manual Data Input)와 Motor의 전원을 끈채로 로봇을 직접 이동시켜 Teaching하는 Free Teach Mode가 있다.

Teaching Box는 두대의 로봇 및 외부축을 조작하고, 작업에 대한 궤적을 지시, 수정하며 실제로 로봇의 작업을 편집할 때 로봇 언어중 MOVE 관련 명령어(MOVJ, MOVL, MOVA, MOVG, ...)들을 편집, 등록, 수정할 수 있다. 아울러, 로봇의 상태 표시 기능으로 I/O Device의 상태를 표시, 제어할 수 있다.

(4) Edit Mode

본 시스템에서 사용하는 로봇 언어가 High Level Language에 가까우므로 본 에디터는 PC의 풀키(Full Key)를 사용하여 프로그램을 작성할 수 있고, 또한 많이 쓰이는 명령어의 편리를 위하여 로봇 언어 MACRO 기능을 갖추고 있다. 본 에디터는 Full Screen Editor로 Doubly Linked List Structure로 편집중인 Line을 가리키는 Pointer는 Text를 위한 Buffer와 시작과 끝을 가리키는 Pointer를 가지고 있다. 그리고, 본 에디터가 갖고 있는 주요기능으로 Search & Replace, Block Copy, Move, Delete, Block Print, File Print, File Read/Write등이 있다.

(5) Run Mode

Supervisory System은 프로그램된대로 자기 자신의 작업을 수행할 뿐만 아니라, 상태를 Monitoring하면서 각 Servo에 필요한 작업을 적절히 할당한다. 프로그램의 정확성을 확인하기 위해서, 세가지 형태로 프로그램을 수행할 수 있도록 하였다. 즉 단계별 동작 확인을 위한 단계별 수행(Step Run), 처음 단계에서 마지막 단계까지 동작 확인을 위한 사이클 수행(Cycle Run)과 동작 확인 후 반복 동작을 위한 자동

수행(Auto Run)이다. 충돌 회피와 두 대 팔의 협조 작업을 효과적으로 조작하기 위해, Supervisory는 궤적을 계획하고 계산하여, 이를 Common Memory를 통해 각 Servo System과 통신한다. 또, 연속 궤적제어의 부드러운 동작을 보장하기 위해 각 관절 동작이 계획되어질 때 자동적인 가감속이 고려되어야 한다. 이를 위해, 2차 이상 Shaping Filter가 직각 공간에서 궤적을 보장하기 위해 적용된 후 관절 궤적으로 변환된다.

(6) Look Mode

Look Mode에서는 크게 선처리와 후처리로 나눌 수 있다. 선처리는 카메라 선택(4 대중 한대를 선택), Image Grab, Snap, Threshold를 이용한 Binary Processing등의 기능이 있고, 후처리에서는 원도우 처리와 물체 학습, 저장 기능이 있다. 물체 학습은 각 물체의 특성에 따라 Chain Code, Projection, Matching등 여러가지 방법을 선택할 수 있다. 학습된 물체의 특징량은 그 이름과 함께 Common Memory에 저장된다. 이 데이터는 Supervisory CPU가 물체를 인식하여 동작할 때 쓰인다.

이들 기능 모두를 효과적으로 수행하기 위해, Supervisory 컨트롤러는 Real Time O.S.인 VxWorks를 사용하여 구현하였다.

2. Vision 시스템

비전 시스템은 VME 사양에 맞게 설계된 SVS900-DT Board를 사용하였는데, 이는 해상도가 512 × 512 이고, 명암도가 256단계이다. 본 비전시스템은 물체인식, 위치검출의 역할을 담당하고 있으며, 또한 최대 4대의 카메라를 사용할 수 있도록 함으로써 필요부분에 설치하여 효율을 높일 수 있도록 하였다. 이를 위한 제어명령어가 Table 1에 정의되어 있다.

3. 로봇 제어 시스템 (Arm Control System)

이 시스템은 Supervisory 시스템에서 받은 로봇 명령에 대해 Inverse Kinematics을 풀고, 이를 관절 운동으로 변환하여 Actuator 시스템을 구동시켜 실제로 로봇 Arm을 제어한다. 제어 장치는 Position Board와 Power Amp로 구성되어 있으며 최대 12축까지 제어할 수 있다. 이 장치는 매 Solution 시간마다 다음의 4가지 일을 수행한다.

- 1) 원하는 궤적 정보를 얻고 로봇의 현 상태를 알기 위해 Supervisory 통신

표 1. 다중로봇트 제어 언어 일람표

Group	Instruction	Structure(Syntax)
Basic Motion	MOVJ	movj/movl <robot> to <loc1>
	MOVL	with sp = <speed> pl = <position level>
	MOVC	move/mova <robot> via <loc1> <loc2>
	MOVA	with sp = <speed> pl = <position level>
	MOVJ	movi <robot> by <offset> with sp = <speed>
MOV_EXT	mov_ext <device> to <loc1> with sp = <speed>	
Master & Slave Motion	MS_MOVL	ms_movl <robot> to <loc1> with sp = <speed>
	MS_MOVI	ms_movi <robot> to <offset> with sp = <speed>
	MS_MOVC	ms_move/ms_move <robot> via <loc1> <loc2>
	MS_MOVA	with sp = <speed>
Collision Avoidance	COLL_AVOID	do coll_avoid
Conditional Instructions	ON condition	on <statement> do <statement>
	DO action	
Conveyor Tracking	CV_SPCBK	cvspchk cv #<num>
	CV_SYNC	cvsync <robot> with ul #<num> l = <num>
	MOVS	movs <robot> to <loc1> with sp = <speed>
CVEND	cvend	
Concurrent Motion	COBEGIN	cobegin
	COEND	motion instruction 1 motion instruction 2 coend
Special Motion	SET_PALLET	set_pallet <robot> for <loc1> <loc2>
	SET_SEG	set_seg x = <num> y = <num> z = <num>
	SFTON	sfton <robot>
	SFTOFF	sftoff <robot>
Coordinate Translation	TRANS_COORD	trans_coord <loc1> for <coord>
	REF_COORD	ref_coord <coord>
	RECOVER_COORD	recover_coord <robot>
	AFFIX	affix <object> to <robot>
	UNFIX	unfix <robot> for <robot>
MAKEPOSITION	makeposition TRANS, TOOL1 ... LOC	
Gripper Control	GRASP/RELEASE	grasp/release <robot> until force = <num>
		with width = <width>
	GRIP_UP	grip_up/grip_down <robot> with sp = <speed>
	GRIP_DOWN	disp = <displacement>
	GRIP_ROTATE	grip_rotate <robot> with sp = <speed>
	disp = <degree>	
TOOL	tool <tool#>	

- 2) 관절 궤적을 구하기 위해서 Inverse Kinematics 변환과 현재 위치 표시 위해 Direct Kinematics 변환
- 3) 관절 궤적 계획(Pulse Generation and Exponential Filtering)
- 4) 매 5msec당 펄스 명령을 분배

IV. 주변기기 및 센서인터페이스 시스템의 기능별구성

1. Man-Machine Interface System (VFIOC)
VFIOC (Video/Floppy/IO Card)는 VME 사양에 맞게 디자인된 Man-Machine Interface Board이다. 이는 PC Bus Converter Interface, Floppy Interface, Parallel Interface, Keyboard Interface, Keypad Interface의 Module을 내장하고 있으며, 이에 대한 Block Diagram은 그림 6과 같다.

(1) PC Bus Converter Interface

PC Bus Converter Interface는 Motorola I/O Channel Bus의 Signal을 IBM PC/XT Bus의 Signal로 전환시키는데, 이때 Address Expansion Register를 사용하여 Address Window를 확장시켜 준다.

(2) Floppy Interface

Floppy Interface는 3.5" Floppy Drive와 IBM PC Data Format을 사용한다. 이의 구현은 VxWorks Kernel의 Read, Write 및 Format기능을 이용하였다.

(3) Keyboard Interface

Keyboard Interface는 IBM PC/AT Keyboard로부터 Data를 받아 Host로 보내는데, 이때 VxWorks Kernel을 이용하여 Read/Write 및 Status Report 기능을 제공한다.

(4) Keypad Interface

Keypad Interface는 16개의 Push Button을 인식하여 간이 Keyboard기능을 제공하는데, 이때 여러개의 Key가 눌러지는 경우, 최초의 Key Data를 Latch한다.

2. 다중 로봇트용 Multi-Functional Teach Box

본 연구에서 개발한 Teach Box는 두 대의 로봇트 및 외부축을 조작하고, 작업에 대한 궤적을 지시, 수

Group	Instruction	Structure(Syntax)
Boolean Logical Operators	AND/OR NOT/XOR	<variable> and <variable>
Relational Operators	<<,>>= ==,!=	<variable> == <variable>
Algebraic Operators	*,-,+/,	<variable> * <variable>
Control Instructions	IF, THEN ELSE, ENDIF	if (expression) then [else] ... endif
	FOR, STEP NEXT	for <Lvar> = <num> to <num> step = <num> ... next i<two_num>
	WHILE, WEND	while (expression) ... wend
	FCALL, FRET	fcall <function_name> [with <num>] ... fret
	JCALL	jcall <job_filename>
	PAUSE	pause if ln#10 = 1
Vision	SV_INT	
	SV_GRAB	
	SV_GLS	
	SV_THSET	sv_thset [with <num>]
	SV_SNAP	
	SV_RECOG	
USE CAMERA	use camera <num>	
SET SIZE	set size <num>	
SET GRAY	set gray <num>	
PRINT	print <string> on <x_position>, <y_position>	
Input/Output Control	PULSE	pulse #<port> <duration>
	DIN/DOUT	din/dout #<bit> <state>
	IN/OUT	in/out #<byte> <state>
	WAIT	wait until [time <reop> <num>] wait until [din #<port> <state>]
Sensor Related Instruction	TIME, DIST	
	FORCE LIMIT	limit #<number>
기 타	DEFINE	define <robot> master/slave
	END	
	SPEED	speed = <speed>

정하며 실제로 로봇의 작업을 편집할때 로봇 언어 중 70% ~ 80%를 사용하게 되는 MOVE 관련 명령어(MOVJ, MOVL, MOVA, MOVCL, ...)들을 편집, 등록, 수정할 수 있다. 아울러, 로봇의 상태 표시기능으로 I/O Device의 상태를 표시, 제어할 수 있으며 또, 로봇의 위치 및 자세를 사용자가 선택한 좌표계를 기준으로 나타낼 수 있는 등 기존의 단순한 Teaching Box 기능 보다 크게 향상시켰으며, Teaching Box의 Display를 위해 LCD를 이용하여 Man-Machine Interface 부분을 향상시켜 사용자로 하여금 보다 편리하게 명령어의 편집, 등록 및 수정등을 할 수 있게 하였다.

본 연구에서는 개발한 Teaching Box의 기능을 다음과 같이 크게 5가지로 대별된다.

- 1) 교시 기능 및 축 조작 기능
- 2) 궤적의 확인, 실행 기능
- 3) 명령의 등록 및 편집, 수정 기능
- 4) 로봇의 상태 표시기능
- 5) 긴급사태시 조작 및 안전기능

3. 다중 센서 인터페이스 시스템

로봇 작업 내용이 복잡해지고, Motion 중심이 아닌 Task 중심의 프로그램을 하게 되면 로봇은 항상 외부 변화를 감시하고 이에 대응하는 동작을 수행해야 한다. 따라서 환경 변화를 감시하기 위해 센서가 필요하고 센서 신호에 의존하여 Trajectory Planning을 하는 Motion이나, Motion을 수정하는 기능이 필요하다.

본 시스템에서는 센서 처리 보드를 준비하여 16개의 Digital Input Signal을 Check하는 Din(Digital Input), Digital Signal을 출력하는 Dout(Digital Output), Conveyor Tracking을 위해 Conveyor Encoder 신호, Vision 신호등을 인식한다. Digital Signal 입력 장치로는 MACRO6744 Board, 출력 장치로는 MACRO6745 Board를 이용하였다.

4. Conveyor Tracking

Conveyor 동기 운전기능이란 Conveyor 이동량을 고려한 궤적계획에 의해서 Conveyor가 정지한 상태로 교시한 궤적을 동작중인 Conveyor위로 재현하는 기능이며, 궤적 추종을 위한 Conveyor의 이동량의 검출은 Encoder의 Feedback Pulse 누적치로 계산된다. 보정기능의 하나로 속도 변동에 따른 추종오차

를 줄이기 위하여 평균 속도를 일정 주기마다 ($T_s=40\text{msec}$) Sampling하여 Extrapolation을 통해 다음 주기에 가야할 양을 미리 계산하여 추종한다. 또한, 동기동작의 개시순간 추종지연에 따른 오차를 줄이기 위하여, 동기 개시시 추종지연시간 후의 오차량을 계산해 Conveyor 방향으로 추종오차량을 보정한다. 전체 Conveyor System구성은 그림 7과 같다. 사용된 Sensor는 위와같이 평균속도 측정용, 동기개시용 Limit Sensor가 있어 Conveyor상의 Object가 평균속도 측정 Limit Sensor에 닿았을 경우 기준이 되는 Conveyor의 속도를 구하고, 동기개시 Limit Sensor에 닿았을때 동기 동작이 시작된다.

V. 다중 로봇 시스템을 위한 로봇 제어 언어

본 연구에서 개발한 로봇 제어 언어는 한 대의 로봇을 위한 상용화된 로봇 언어를 포함하여 전체 87가지의 명령어를 제공하고 있다(Table 1). 우선, 관절운동(PTP)을 위한 MOVJ, 연속 궤도 운동(CP)을 위한 MOVL, 주어진 세점을 통과하는 원호운동을 위한 MOVCL, 현재 위치에서 주어진 오프셋 거리만큼 증분치 운동을 위한 MOVI와 같은 기본 운동 명령어를 제공한다. 그러나, 이들 기본 명령어만을 이용하여 두대의 로봇이 함께 일하는 것이 실제로는 어려우므로, 본 시스템에서는 Master & Slave 운동 명령어를 준비하였다. 한 대 로봇을 Master로 다른 한 대를 Slave로 정의하여, Slave 운동이 Master의 운동을 추종하도록 한다. 특히, MOVJ를 제외한 모든 기본 운동 명령어는 Table 1.에서 보는 바와 같이 Master/Slave 운동으로 사용할 수 있다. Master/Slave 운동은 두 로봇간의 Position Relation을 구하여 Relative Position을 유지하도록 Slave가 추종하는 방법과 Slave Robot에 장착한 F/T Sensor의 Force Feedback방법으로 추종할 수 있다.

또한 두대 로봇의 동시동작(Concurrent Motion)을 위하여 COBEGIN and COEND 명령어¹¹⁾를 준비하였다. COBEGIN과 COEND 선언 사이의 모든 동작명령어는 두대 로봇이 동시에 자기 자신의 컨트롤러에 의해 제어되는것처럼 동시에 수행

된다.

본 시스템은 Positioner 혹은 Conveyor 시스템과 같은 외부 장치를 포함하여 동시에 12축까지 제어할 수 있기 때문에, 두대 로봇 사이의 충돌 회피와 외부 장치와의 동기를 고려하여야 한다. 이를 위해, 조건운동 명령어를 ON condition DO action의 문법으로 제공하였다. 그러한 조건적 명령어(Conditional Motion)는 위에 기술한 모든 운동 명령어에 사용할 수 있다. 예를 들어, 두대의 로봇 사이의 가능한 충돌회피를 하기 위해, 거리 조건과 함께 COLL-AVOID 명령어를 다음과 같이 사용할 수 있다.

```
cobegin
  movl robot1 to loc1 with sp = 60
  movl robot2 to loc2 ON dist < 10.0 DO
    coll_avoid
coend
```

특히, 일반적인 3-차원 충돌회피^{[2], [13]}는 실시간에 구현하기가 어렵기 때문에 본 시스템에서는 단지 두대의 수평다관절형 로봇에만 적용되는 충돌 회피 기능을 구현하였다. 알고리즘은 다음과 같다.

만약 두대 로봇 사이의 거리가 주어진 조건보다 작으면(충돌 가능성이 있으면) i) 두 대 로봇이 다른 방향으로 움직이는 경우에는, Master 로봇은 목표를 향해 움직이고 Slave 로봇은 충돌이 회피될 때까지 안전한 지점까지 회피하고, ii) 두 대의 로봇이 같은 방향으로 움직이는 경우에는 우선 Master 로봇이 움직이고, 뒤 따라오는 로봇은 잠시동안 기다린다(Wait).

이러한 동작 명령어는 16점 디지털 입력 신호를 확인하는 DIN 조건, 동작 시간을 확인하는 TIME 조건, 입력 센서신호에 근거로 시스템이 궤적을 다시 계획하는 MOVS 동작, 시스템을 멈추는 STOP 동작, 16점 디지털 신호를 출력하는 DOUT과 같이 이용할 수 있다

전술한 동시 동작 명령어와 조건문장을 구현하기 위해서, Real Time O.S. VxWorks를 채택했다. 특히, 모든 동작 명령어는 프로세스로써 정의한다. COBEGIN 과 COEND 문장인 경우, 모든 프로세스는 COBEGIN과 COEND 사이에서 Create 되고 Spawn 된다. Conditional Motion인 ON condition DO action 문장의 경우에는, Motion

을 수행하다가 Condition을 만족하면 Motion을 정지시키고 Action을 수행한다.

전술한 동작 명령어 외에 Teach Point 근처를 연속적으로 경유하는 Position Level과 좌표계를 변환하여 작업하게 하는 Trans-Coord와 기준 좌표계를 새로 설정하는 Ref-Coord등을 이용한 Job Shift 기능, Pallet를 위한 Palletizing 기능등이 있다. 이외에도 Tool 제어, 논리 연산, Loop 제어, 산술 연산 명령어 등이 구현되어 있다. (Table 1 참조)

VI. 맺음말

지금까지 Multi-Tasking Real Time O.S인 VxWorks를 기본으로 하여 다중센서융합(Multi-Sensor Fusion)능력을 갖는 다중로봇 협조제어 시스템의 구현에 대하여 살펴 보았다. 본 제어시스템은 두대 로봇의 제어에 필요한 장애물 회피, 조건 동작(Conditional Motion)혹은 동시동작(Concurrent Motion)과 Device와의 동기 Motion(Conveyor Tracking)을 수행할 수 있게 구현하였고, 몇몇 작업을 통해 우수성을 입증하였다.

앞으로 본 연구와 관련한 추후과제로는 1) 자유도가 6 관절형인 수직다관절 Manipulator를 위한 충돌회피 알고리즘의 개발 2) TwoArm Robot의 상대 위치를 위한 Auto-Calibration System의 개발 3) CAD Based Trajectory 생성등이 있다.

※ 본 연구는 상공부 공업기반 기술개발사업의 연구비 지원으로 수행 되었음.

參 考 文 獻

[1] J. W. Roach and M. N. Boaz, "Coordinating the Motions of Robot arms in a Common Workspace," IEEE Journal of Robotics and Automation, Volume RA-3 No.5, October, pp 437-444, 1987.
 [2] R. Guptill and S. Paul, "Multiple Robotic Devices : Position Specification

- and Coordination." *IEEE International Conf. on Robotics and Automation*, pp 1655-1659, 1987.
- [3] A.A. Mangaser, Y. Wang, and E.S. Butner, "Concurrent Programming Support for a MultiManipulator Experiment on RIPS." *IEEE International Conf. on Robotics and Automation*, pp 853-859, 1989.
- [4] M.A. Turk, "A Vision System for Autonomous Land Vehicle Navigation." *IEEE Trans. Patt. Anal. Mach. Intell.*, Vol.10, No.3, pp 342-361, May 1988.
- [5] S. Venkatesan and C. Archibald, "Real time tracking in five degrees of freedom using two wrist-mounted laser range finders." *Proc. of IEEE Int. Conf. on Robotics and Automation*, pp 2004-2010, May 1990.
- [6] A.J.koivo, and N. Houshang, "Real-Time Vision Feedback for Serving Robotics Manipulator with Self-Tuning Controller." *IEEE Trans. on Sys. Man, Cybern.*, Vol.21, No.1, pp 134-141, January/February 1991.
- [7] M. A. Abid, R. O. Eason, and R.C.Gonzalez, "Autonomous Robotics Inspection and Manipulation Using Multisensor Feedback." *IEEE Computer*, Vol.24, No.4, pp 17-31, April 1991.
- [8] D.M.LYONS, and M.A.ARBIB, "A Formal Model of Computation for Sensor-Based Robotics." *IEEE Trans. on Robotics and Automation*, Vol.2, No.3, pp 280-293, June 1989.
- [9] Z.Bein, S.R.Oh, I.H. Suh, J.O. Kim, and Y.S. Oh, "Automatic Assembly for Microelectronic components." *IEEE Control Systems Magazine*, Vol. 9, No. 4, June 1989.
- [10] Motorola series in solidstate Electronics, VMEbus Specification Manual, 2nd printing Revision C.1 October 1985.
- [11] S.Mujtaba and R.Goldman : AL users' Manual, Stanford Artificial Intelligent Laboratory Memo, AIM-323, 1979.
- [12] R. A. Basta, R. Mehrotra, and M. R. Varanasi, "Detecting and Avoiding collisions between two robot arms in a common workspace." *Robot Control Theory and Application*, pp 185-192, 1988.
- [13] R.A.Basta, R.Mehrotra, and M.R. Varanasi "Collision Detection for Planning Collision Free Motion of two Robot Arms." *Proceeding of the IEEE International Conf. on Robotics and Automation*, 1988. 🌐

筆者紹介



徐 一 弘

1955年 4月 16日生

1977年 서울대 공대 전자공학과 졸업

1982年 한국 과학기술원 전기 및 전자공학과 졸업(공학박사)

1982年 대우중공업 기술 연구소 근무

1987年 ~ 1988年 미국 미시간대 객원 연구원

현재 한양대 공대 전자공학과 교수, 대한전기학회 편집위원

金 兌 垣

1965年 12月 28日生

1988年 2月 한양대학교 전기공학과 졸업

1990年 2月 한양대 대학원 전기공학과 졸업 (공학석사)

1990年 3月 ~ 현재 한양대 대학원 전자공학과 박사과정 재학중

주관심 분야 : 로봇트 버전, 뉴로퍼지

呂 熙 珠

1965年 5月 2日生

1988年 2月 한양대학교 전자공학과 졸업

1990年 2月 한양대 대학원 전자공학과 졸업 (공학석사)

1992年 3月 ~ 현재 한양대 대학원 전자공학과 박사과정 재학중

주관심 분야 : 로보틱스, 뉴로퍼지

高 德 用

현재 삼성항공 정밀기기 연구소 근무

曹 廷 鎬

현재 삼성항공 정밀기기 연구소 근무