

수직 다관절 로봇을 위한 시스템 프로그램

*
최 호 현, 고 경 철, 문 전 일, 임 계 영
금 성 산 전 (주) 연 구 소

A System Program for Revolute Type Manipulator

*
Ho-Hyon Choe, Kyong-Chul Koh, Jeon-Il Moon, Kye-Young Lim
Goldstar Industrial System Co., Ltd., R&D Lab.

ABSTRACT

In this paper, a system program is introduced for the effective operation of revolute type robot manipulator. The system program consists of several modes and specific functions. These modes and functions are flexible and easy to use because the structure is modular and conversational. This program had implemented and Verified the efficiency together with a five axes robot.

1. 서 론

로봇 시스템 프로그램은 로봇의 기능을 결정하는 소프트웨어로써 로봇 작업을 정의하는 프로그래밍부와 이를 해석하는 인터프리터부 그리고 좌표 연산을 통해 계획된 경로를 설정하여 실제 로봇을 실행시키는 플렉시백 및 원점조정모드, 제어기의 하드웨어에 관련된 서브평선으로 구성된다. 기존의 Robot 작업을 정의하는 프로그래밍 방법은 특정 로봇 언어를 사용하여, 플렉시백 방식으로 티칭된 위치변수간의 동작조건, 속도 그리고 입출력 상태, 분기조건등을 설정하는 언어 방식을 채택하고 있다[1]. 그러나 이같이 로봇 작업 전체 흐름(flow)을 로봇 언어로써 구성할때 이를 해석하는 시간(Interprete time)이 많이 소요되고 스텝마다의 로봇 동작이 지연되고, 입출력 응답성(responsibility)이 늦어지는 단점을 안고 있다. 따라서 본 논문에서는 로봇 전용 언어 방식이 갖는 단점을

해소하기위해 로봇 작업에 관련된 부분을 포맷(format)화된 테이블에 작성하고 흐름제어(flow control)에 관련 부분은 일반적인 고급언어(High-level language)를 사용하도록 하는 로봇 프로그래밍 체계를 구현하였다 [2]. 이러한 티칭구조를 갖는 시스템 프로그램은 이미 5축 수직다관절 로봇 개발과정에서[3] 실행한바 있으며 효율성 및 유연성이 입증된 바 있다. 본 논문에서는 이러한 티칭 구조에 대해 중점적으로 설명한다.

2. SYSTEM PROGRAM의 구성

시스템 프로그램의 기본적인 구성은 그림1과 같으며 운영체제와 관련된 소프트웨어와 하드웨어 관련 소프트웨어로 구분하여 하드웨어 변경에 따라 유연하게 적용하도록 한다. 개발 언어로서 PL/M-86을 이용하였으며, 모듈타 프로그램 방식을 이용하여 소프트웨어의 구조를 계층적 구조(Hierarchical)로 체계화 하고, 개발의 용이성 및 기능의 추가, 변경이 용이하도록 구성한다. 이 시스템 프로그램의 수행은 티칭박스와 CRT 터미널을 이용하여 사용자와 대화식으로 이루어지며 각 모드간의 변환은 메뉴 방식을 이용하여, 사용자의 조작이 편리하도록 설계한다. 시스템 프로그램의 크기는 48KByte로 Firm-Ware로써 EPROM에 기억되고, 로봇 작업 프로그램은 마그네틱 버블 메모리(MBM)에 저장한다.

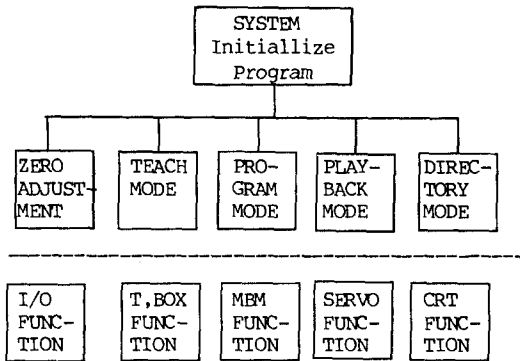


Fig. 1 Block Diagram of System Program

3. Teach Mode

이 모드에서는 전술한바와 같이 로봇 작업에 관련된 부분을 포맷화된 테이블에 작성하며, 작성된 테이블의 데이터는 로봇의 움직임을 순차적으로 정의하는 일종의 서브 작업의 의미를 갖는다. 테이블은 고유번호(0-99)를 가지며, 하나의 테이블은 100 Step까지 로봇 작업 포인트를 지정한다. 그림 2. 는 이러한 티치 테이블의 구조를 나타낸다.

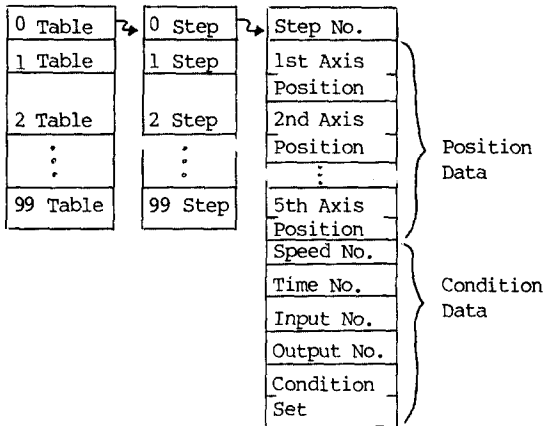


Fig. 2 Format of teach table

3.1 포지션 데이터 지정

작업 공간내의 작업 위치를 지정하는 방법에는 작업공간내에 좌표계를 설정하여, 작업지정(Point)의 좌표값을 입력하는 MDI(Manual Data Input) 방식과 로봇 자체를 수동으로 조작하여 작업 지점을 기억시키는 플레이백 티치 방식이 있으며 이때 사용되는 좌표계는 조인트(Joint)좌표계, 원통(Cylindrical)좌표계, 직각(Rectangular)좌표계등이 있다. 실제 테이블에 기억되는 포지션 데이터는 로봇 각축을 구동

하는 모터의 엔코더 펄스수를 16bit 2진수화한 값이며, 실제 플레이백 모드에서 수행될때 좌표연산 및 경로제어를 위한 실수(Floating Point)로 변환된다.

3.2 조건 데이터의 지정

티치 테이블의 매 스텝마다 5축의 위치 데이터로 작업 포인트가 결정되면, 스텝과 스텝간의 로봇 운동조건이 지정된다. 이러한 운동 조건은 로봇의 속도, 보간기능, 입출력 상태등 로봇의 운동상태의 다양한 조건을 결정하는 데이터로 작업조건에 따라 다양하게 선정된다. 그림 3. 에서 조건데이터의 포맷과 그 기능을 표시한다. 로봇의 속도와 타이머 기능은 256 단계 선정하며, 입출력 포트의 지정은 0-15까지 선정 할 수 있다.

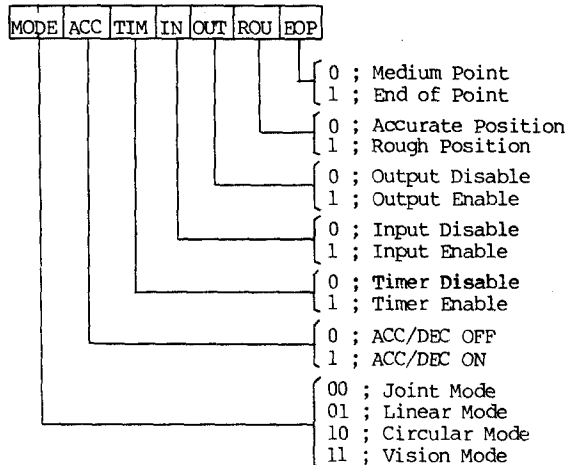


Fig. 3 Format of Condition Set

(1) 동작모드

- 각 스텝간의 운동시 보간기능을 선택하며, 각 모드는 다음과 같다.
 - 조인트 모드 : PTP(Point to Point) 모드로서 5축의 위치 변화가 가장 큰 축을 기준하여 동시에 운동이 완료하도록 배분한다.
 - 직선모드 : 두점간을 직선으로 연결한 경로를 추종하는 CP(Continuous Path) 운동을 수행한다
 - 원호모드 : 세점을 이용하여 원호 경로를 연산하여 수행한다.
 - VISION 모드 : 이 모드가 지정된 스텝이 (N) 스텝일때 (N+1) 스텝과 (N+2) 스텝으로 지정되는 좌표 프레임으로 부터, 비전시스템과의 통신을 통해 얻어진 오프셋 값을 연산하여 작업 위치를

연산하여 수행한다.

(2) 조위치 선정(Rough Position)

로봇 작업내에 장애물을 피하기 위한 경로를 티칭할때 사용하는 조건으로 운동시간을 최소화하는데 그 효과가 있다.

(3) 가속속 선정(Acceleration/Deceleration)

속도 프로파일(Velocity Profile)의 선택을 하는 기능으로 급격한 속도변화를 최소화하는 운동(Fine Motion)을 수행한다

(4) 입출력 조건 선정

각 스텝마다의 입출력 조건을 결정하는 기능으로 입력조건이 선정되면 해당 입력포트의 신호가 들어올때까지 기다리며, 출력 조건이 선정되면 해당출력포트에 출력한후 다음 스텝으로 진행한다.

(5) EOP 선정

티치테이블의 마지막 스텝을 알리는 플래그(Flag)로 EOP가 지정된 스텝의 위치까지 로봇이 운동이 수행되면, 하나의 티치 테이블로 정의된 일련의 작업은 완료된다.

3.3 티치 테이블의 수행

작성된 티치테이블은 1 스텝부터 EOP가 지정된 스텝까지 순차적으로 수행되며, 로봇의 작업을 여러 단위 작업으로 구분할때, 하나의 단위 작업은 하나의 티치테이블로 대응된다. 따라서 티치모드에서 작성된 일련의 테이블들은 프로그램 모드에서 일종의 서브루틴으로 간주되어 처리되며, 티치테이블 자체는 포맷화된 테이블에 작업위치, 속도, 작업조건 입출력 상태를 기록하고 있으므로 다른 명령어 해석이나 데이터 패치(Fetch) 없이 직접 해석되므로 빠른 응답성을 갖고 수행된다. 또한 스텝 별로 위치 및 입출력 상태를 체크할 수 있어 디버깅(Debugging)이 수월한 장점을 갖는다. 예를 들면 그림. 4와 같이 컨베이어로부터 조립 부품을 집는 작업의 경우, 총 4 스텝의 티치 테이블로 (표 1)과 같이 작성할 수 있다.

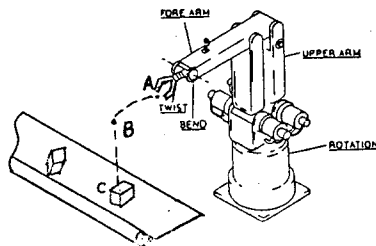


Fig. 4 Example of Robot Work

STEP NO	POSITION	SPEED	MODE	INPUT	OUTPUT	EOP
0	A	200	JOINT	OFF	OFF	OFF
1	B	50	LINEAR	ON	OFF	OFF
2	C	100	LINEAR	OFF	ON	OFF
3	B	100	JOINT	OFF	OFF	ON

(Table 1. Example of Teach Table)

(표 1)의 내용을 살펴보면, 로봇이 A 지점에서 출발하여 고속 조인트 모드로 컨베이어 근처인 B 지점까지 간후 컨베이어의 부품 감지 센서의 신호를 받고 C 지점까지 저속 직선 모드로 내려간후, 출력신호를 통해 로봇의 핸드를 작동시켜 부품을 집은 후 다시 중속 직선 모드로 B 지점까지 들어올리는 동작으로 하나의 단위의 작업을 정의하고 있다. 따라서 이렇게 구성된 티치 테이블들은 프로그램 모드에서 반복 선택 될 수 있어, 로봇의 작업 전체를 정의할 때 필요한 티칭 포인트 수를 절감할 수 있다.

4. 프로그램 모드

로봇에 주어지는 작업은 여러개의 단위작업으로 구성되고 이러한 단위 작업은 고유번호(0-99)를 갖는 티치 테이블에 작성 되었을때, 이러한 단위 작업의 순서 및 실행 조건, 반복 횟수를 정의할 필요가 있다. 이러한 역할을 수행하는 것이 바로 프로그램 모드이다. 본 연구에서는 이미 로봇의 운동과 관련 부분을 티치 테이블로 분리하였기 때문에 로봇의 전용 언어 체계를 설계할 필요성이 없으며, 단지 티치 테이블을 수행하거나 수정할 수 있는 통신 프로토콜(Protocol)만을 설계하면된다. 사용자는 일반적인 퍼스날 컴퓨터에서 로봇의 작업 프로그램을 작성 할 수 있으며, 이때 사용하는 언어는 퍼스날 컴퓨터에서 제공하는 고급 언어(High-Level Language)가 될 수 있는 것이다. 이러한 온라인 방식에 의한 호스트 컴퓨터와의 통신 방법과는 별도로 로봇 자체에서도 프로그래밍 능력을 갖는 (Stand-Alone System) 경우도 가능한데, 이 경우 일반적인 고급언어가 수행 될 수 있는 환경을 시스템 프로그램에 포함시켜야 한다.

4.1 온라인 모드(On-Line Mode)

호스트 컴퓨터와의 통신을 통해서 프로그램의 작성이나 수행이 이루어지는 모드로써, 다음과 같은 기능을 갖는다.

- (1) READ 명령 : 티치모드에서 작성한 티치 테이블 내용을 읽어들이므로써 호스트 컴퓨터가 로봇의 티칭 상태를 알고자 할때 사용.

- (2) WRITE 명령 : 호스트 컴퓨터가 티치 테이블을 직접 작성하거나 수정할 수 있는 명령
- (3) PERFORM 명령 : 테이블 번호를 지정하여, 해당 테이블에 기록된 작업을 수행하는 명령
- (4) INPUT 명령 : 로봇의 입력 포트 상태를 읽고자 할때 사용
- (5) OUTPUT 명령 : 로봇의 출력 포트를 지정하여 출력하는 명령
- (6) STATUS 명령 : 현재 로봇의 모드, 에러여부, 동작완료 등의 상태를 읽는 명령
- (7) A-CAL 명령 : 로봇의 원점 조정을 수행하는 명령
- (8) STOP 명령 : 로봇의 동작을 정지시키는 명령
- (9) START 명령 : 로봇의 동작을 시동시키는 명령

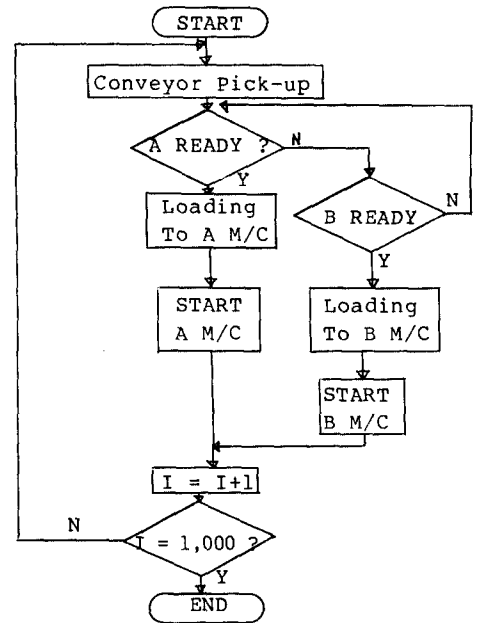
이와같이 로봇과 호스트 컴퓨터간의 통신 프로토콜은 RS-232C를 이용한 시리얼 인터페이스 하드웨어 로직을 통해 수행되며, 통신 에러를 피하기 위해 LRC 코드를 부여한다.

4.2 AUTO Mode

로봇 자체내에서 프로그래밍 할 수 있는 모드로, 전체 로봇 작업의 플로우 콘트롤(Flow Control)을 담당하며 설계된 언어는 PL/M 언어방식과 유사한 구조를 갖도록 하였다. 언어의 해석 방법은 인터프리터(Interprete) 방식을 채택하였다.

4.3 응용 프로그램

그림 5는 컨베이어로부터 부품을 공급받아 A, B 2대의 그라인딩 머신에 1,000개의 부품을 가공하는 작업의 플로우 차트(Flow Chart)를 나타낸다. 이 작업을 컨베이어 부품 픽업(Pick-up) 작업, A 기계 부품 로딩>Loading) 작업, B기계 부품 로딩 작업등의 3개의 단위작업으로 구분하여, 이 단위 작업들을 티치모드에서 티치테이블 No. 1 - No. 3에 정의했다고 가정하자. 이러한 단위 작업의 수행 순서 및 수행 조건 반복횟수는 프로그램 모드에서 작성된 프로그램에 의해 지정된다 (Table 2)



```

DO I=1, 1000;
PERFORM(1);          /* CONVEYOR PICK-UP */
1: IF INPUT(1)      /* A M/C Ready Check */
  THEN DO;
    PERFORM(2);    /* Load Material to A */
    OUT(2);        /* Start A Machine */
    END;          /* End of Do Block */
  ELSE IF INPUT(2) /* BM/C Ready Check */
    THEN DO;
      PERFORM(3);  /* Load Material to B */
      OUT(3);      /* Start B Machine */
      END;
    ELSE GOTO 1;   /* IF A,B are not Ready */
  END;            /* Until 1,000 Material */
END;              /* End of Program */
  
```

Table 2. Example of Program

5. 결 론

본 논문에서는 5축 다관절 로봇를 제어하는 시스템 프로그램의 설계에 있어, 특히 로봇 작업을 정의하는 티칭 체계에 대해 다루었다. 티치코드에서는 일반적인 로봇가 갖는 기능 즉 PTP 및 CP 운동의 수행, 속도 프로파일의 선정, 속도지정, 입출력 상태등의 조건 데이터와 플래이백 티칭 방법을 이용하여 지정되는 위치 데이터를 포맷화된 테이블에 작성하는 방식으로 로봇의 단위 작업을 정의하고 프로그램 모드에서는 일반적인 고급언어 방식의 언어 체계를 이용하여 전체 로봇 수행 작업을 정의한다. 이렇게 2개의 모드로 분리하는 티칭 방식은 다음과 같은 특성을 갖는다.

첫째, 로봇의 운동에 관련된 부분을 랭귀지 인터프리터에서 분리하여, 인터프리터의 부하(Burden)를 최소화 한다.

둘째, 티칭 포인트들이 랜덤(Random)하게 설정되지 않고 일련의 단위 작업을 의미하게 되어, 티칭 포인트들이 체계적인 구조를 갖게 된다.

셋째, 일반적인 고급언어(PASCAL, FORTRAN, BASIC 등)의 사용이 가능하여 호스트 컴퓨터를 이용한 네트워크(Network) 체계를 형성할 수 있다.

참고 문헌

1. Bruce Shimano, "VAL : Versatile Robot Programming and Control System" Proc. of COMPSAC 79, 1979
2. 고경철 외 5인, "VISION SYSTEM 기술개발", 금성사 중앙연구소, 1986
3. 문전일 외 5인, "수직다관절 로봇 개발완료 보고서", 금성사 중앙연구소, 1986