

# 로봇 프로그래밍을 위한 사용자 인터페이스 개발

## Development of a User-Interface for Robot Programming

\*정준호<sup>1</sup>, #국금환<sup>2</sup>, 김두형<sup>3</sup>

\*J. H. Jeong<sup>1</sup>, #K. H. Kuk(kkh@gnu.ac.kr)<sup>2</sup>, D. H. Kim<sup>3</sup>

<sup>1</sup>경상대학 제어계측공학과, <sup>2</sup>경상대학 제어계측공학과, <sup>3</sup>한국기계연구원 지능기계연구센터

Key words : Robot Programming, User Interface, Robot Language

### 1. 서론

로봇을 제작할 때 원하는 동작으로 제어하기 위하여 로봇 명령어를 설계한다. 즉 로봇언어를 개발하게 된다. 로봇언어는 로봇을 제작하는 회사마다 다르게 정의 되며, 로봇을 운용하는 사용자는 로봇의 기종에 따라 다른 명령어를 익혀 동작 제어 프로그램을 작성하게 된다. 이는 로봇 개발자나 운전자 모두에게 부담이 되며, 생산성이 떨어지는 비효율적인 절차이다<sup>1</sup>. 모든 로봇언어가 표준이 되는 기준을 바탕으로 개발되어진다면 이기종의 로봇이라도 용도별 동작 제어 체계는 같게 될 것이다. 따라서 로봇을 제작할 때 해당 로봇의 용도에 맞는 최적의 로봇언어를 설계하고 제어 프로그램을 작성하는데 도움을 줄 수 있는 사용자 인터페이스 시스템을 제시하고자 한다. 로봇 개발자는 본 시스템을 이용하여 원하는 용도의 로봇언어를 쉽게 설계할 수 있고, 본 시스템을 기반으로 개발되어진 로봇언어는 용도별로 유사성을 가지게 된다. 따라서 로봇 운용자는 각기 다른 기종의 로봇이라도 용도에 따라 언어체계가 유사성을 가지므로 로봇언어를 따로 배우지 않아도 동작 제어 프로그램을 작성하는데 있어 쉽게 접근이 가능할 것이다.

### 2. 로봇 언어 설계

로봇 운영 software를 개발하기 위해서는 로봇의 작업 환경을 알아야 하며, 로봇의 사용 목적, 용도 그리고 앞으로의 발전 가능성 및 전망에 대하여 이해하여야 한다. 제조업용 로봇의 작업 셀(workcell)<sup>2</sup>은 로봇과 주변장치들의 작업관계로 분류할 수 있으며, 로봇의 작업 환경은 목적과 용도에 따라 달라질 수 있으므로 환경의 변화에 따라 쉽게 적용하여 사용할 수 있는 시스템의 설계가 필요하다. 일본의 KAWASAKI 로봇에서 사용하는 AS 언어와 국내 삼성의 FARAL 로봇 언어를 비교하면 Table 1과 같이 분류 할 수 있다.

Table 1 Comparison of AS Language with FARAL Language

분류	AS언어	FARAL
동작 제어	17종(JMOVE, LMOVE, DELAY, HMOVE, ALIGN, JDEPART, LDEPART, C1MOVE, C2MOVE, ...)	15종(MOVE, MOVEA, MOVES, DEPART, DRAW, DRIVE, FIT, ROTATE, READY, ...)
속도 정도	7종(SPEED, ACCURACY, ACCEL, DECEL, BREAK, BRAKE, BSPEED)	19종(MSPD, GAIN, ACC, UL, LL, SPEED, INPOST, UNPOSA, ENVELA, RIGHT, LEFT, ...)
이진 신호 처리	20종(RESET, SIGNAL, PULSE, DLYSIG, RUNMASK, BITS, WAIT, EXTCALL, ON, ONI, ...)	10종(GOSIG, IFSIG, SDATA, SDEF, SIG, SIGIN, SIGOUT, SIGINT, PULSE, WAIT)
프로그램 제어	20종(IF, CALL, RETURN, WAIT, TWAIT, MVWAIT, DO...UNTIL, IF...HEN...ELSE...END, WHILE...DO...END, ...)	13종(FOR, NEXT, IF, ELSE, ENDIF, WHILE, ENDFOR, DO, UNTIL, GOTO, GOSUB, RETURN, STOP)
:	:	:

위의 명령어들을 분석해 보면 명령어도 같고 기능도 같거나, 명령어는 다르나 기능은 같으며, 명령어도 다르고 기능도 다르게 분류할 수 있다. 예를 들어, 동작제어 명령에서 AS언어와 FARAL 언어의 DELAY는 지정시간 로봇동작 정지명령이며, DRIVE는 지정

축의 단독동작 명령으로 명령어와 기능이 일치한다. AS언어의 LMOVE는 직선 보간 동작 명령, C1MOVE는 원호 보간 동작 명령이며 이에 해당하는 FARAL언어는 MOVES와 MOVEC로서 명령어는 다르나 기능은 같다. AS언어의 STABLE 명령은 축 일치 후 지정시간 로봇동작을 정지하는 기능이며, FARAL의 CONT 명령은 로봇 동작 보조 명령으로 각각의 동작을 연속으로 움직이도록 설정하는 명령으로 명령어와 기능이 다르나 다른 명령의 조합으로 인해 같은 기능을 구현할 수 있는 상황이다.

로봇 명령어는 사용 용도에 따라 여러 가지 방법으로 나눌 수 있으나 중량물 핸들링을 위해 좌표계 정의, 속도 정의, 홈핑 정의, 동작 명령, 대기 명령, 핸드 명령, 2진 신호제어 명령, 실행정의 명령 그리고, 프로그램 제어명령의 아홉 가지 그룹으로 정의 하였다. 로봇 명령어 정의는 다음과 같은 기준으로 정의하였다. 기존 ANSI C 문법과 호환되며, 로봇 제어를 위한 명령을 추가 한다. atoi, atof, itoa, ftoa 등 C 호환 기본 함수를 지원한다. 외부 기계 동작 상태를 관별할 수 있는 외부 I/O 기능을 제공한다. 명령어(operator)는 8자 이내로 한다. 인수(operand)는 가능한 규격화 한다. 명령어만으로 의미를 파악할 수 있어야 한다. 동일그룹의 명령어는 선두 2~4문자를 같은 문자열로 정의하여 그룹화 한다. Table 2는 이러한 기준에 의해 로봇 명령어를 설계하였다.

Table 2 Design of Robot Command

그룹	명령어
좌표계 정의	AXISJONT, AXISBASE, AXIS6END, AXISTOOL, AXISUSER, AXISABS, AXISINC
속도 정의	SPEED, ACCURACY, ACCEL, DECEL
홈핑 정의	HOME
동작 명령	MOVEP, MOVEV, MOVEC, MOVESIG, MOVEVIA
대기 명령	WAITTIME, WAITCOND
핸드 명령	HOPN, HCLS
2진 신호 제어	SIGRESET, SIGOUT, SIGIN, SIGHOLD, SIGREVS, SIGSET
실행 정의	SKIPON, SKIPOFF
프로그램 제어명령	IF-END, WHILE-END, DO-WHILE, FOR-NEXT, SWITCH-CASE-END, GOTO, BREAK, CONTINUE, EXIT

### 3. 사용자 인터페이스 설계

사용자 인터페이스 시스템의 구동 환경은 윈도우즈 XP 이상의 운영체제에서 사용 되어질 것이다. 윈도우 운영체제의 장점을 최대한 적용하여 최종 사용자가 쉽고 직감적으로 사용할 수 있는 에디터가 요구된다.

개발 도구로 Microsoft Visual C++ 8.0을 사용한다. 마이크로소프트사에서 제공하는 윈도우용 프로그램의 개발 도구로써 SDK(Software Development Kit)를 사용하며, 순수 API(Application Programming Interface) 함수만을 사용하여 개발한다. 이로써 시스템 차원에서 프로그램이 가능하며, 차후 시스템의 구성이 달라진다 하여도 쉽게 포팅이 가능하다.

사용자 인터페이스 시스템은 Interface Define Robot Language Spec.과 Robot Command Generator, Control Sequence Editor, Object Code Generator, Target System으로 구성되어 있다.

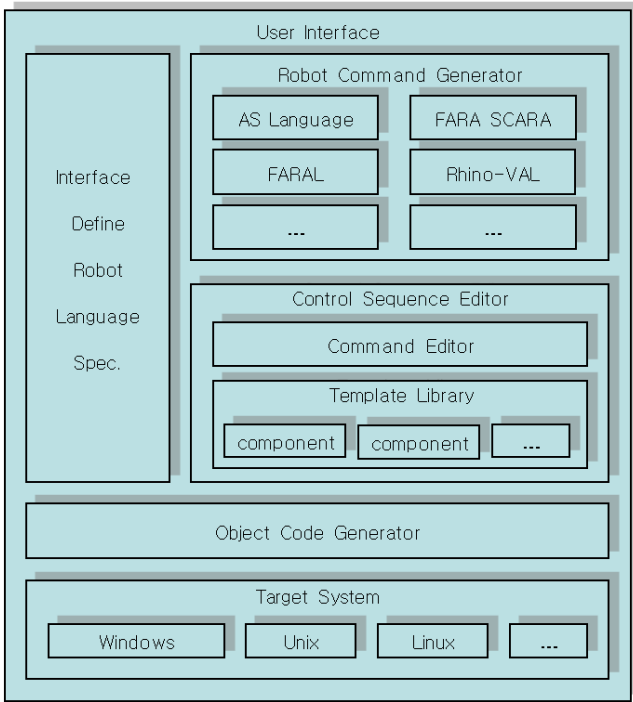


Fig.1 Design of a User Interface

Fig.1에서 Interface Define Robot Language Spec.은 본 논문에서 설계한 로봇 언어이며, 로봇 제어 프로그래밍시 사용되어지는 명령어이다. Robot Command Generator는 사용자가 기존에 사용해서 알고 있는 로봇 명령어 즉, 타 로봇에서 사용되고 있는 로봇 명령어를 User-Interface에 등록할 수 있는 구조를 제시하며, 이는 로봇 제어 프로그래밍시 본 논문에서 설계한 로봇 언어를 사용하지 않고 스스로 등록한 언어를 사용할 수 있는 구조이다. 또한 로봇 언어를 개발하고자 한다면 Robot Command Generator에 새로운 명령어를 정의하여 등록한 다음 제어 목적에 맞는 프로그램을 작성해 보면서 최적의 로봇언어를 설계하는 도구로 사용할 수 있을 것이다. Control Sequence Editor는 로봇 제어를 위한 프로그램 작성용 도구이다. 명령어 편집 기능과 공정별 표준 알고리즘을 제시하여 템플릿을 제공하며, 템플릿은 사용자의 요구에 따라 수정 변경이 가능하며, 사용자 스스로 새로운 템플릿을 정의하여 등록할 수 있다. Object Code Generator는 Target System에 따른 목적코드를 생성할 것이다.

4. 템플릿 설계

정의된 로봇 명령어를 컴포넌트화 함으로써 사용자가 제어 프로그램을 작성할 때 편리하고 효율적인 작업을 할 수 있도록 지원하고자 한다. 위에서 정의한 로봇 명령어는 일반적으로 사용되어지고 있는 산업용 로봇의 기능을 정의하였으며, 로봇마다 고유의 명령어를 정의하여 사용하고 있다. 어떤 로봇의 명령어를 사용해 본 사용자라면 자신이 알고 있는 명령어를 다음의 테이블에 등록할 수 있다.

이를 지원하기 위하여 사용할 명령어 분류명을 컴포넌트 분류 코드에 등록한다. 등록된 컴포넌트 분류 코드를 primary key로 갖는 명령어 집단을 로봇 언어 테이블에 사상시켜 추가한다. 컴포넌트 분류를 위한 데이터 포맷은 Fig.2와 같이 정의하였다.

코드	로봇 언어	접두어	Link
----	-------	-----	------

Fig.2 Data Format for Component

[코드]는 로봇언어를 구별하기 위한 고유 ID이며, [로봇 언어]는 본 논문에서 정의한 명령어, [접두어]는 로봇 제어 프로그래밍시 소스코드에서 로봇언어를 판단하기 위한 구분자로서 인터프리터 시에 인터페이스 정의 로봇 언어 코드로 바꾸기 위한

용도이기도 하다. 컴포넌트 분류를 위한 데이터 포맷에 의해 실질적인 데이터는 Fig.3과 같이 Linked List 구조로 연결되며, 이는 필요시에 얼마든지 추가 및 삭제가 가능한 구조이다.



Fig.3 Linked List for Component

따라서 사용자는 계속적으로 새로운 로봇 언어를 추가 할 수 있으며, 이를 새로운 로봇 언어 개발시에도 개발자가 정의한 언어로 등록가능하며 여러 가지 테스트를 거쳐 효율성 판단에 의해 새로운 로봇언어를 설계하고 개발하는 용도로도 사용 가능하다. 로봇은 응용분야에 따라, 용도에 따라, 기능에 따라 등등으로 무엇을 기준으로 하는가에 따라 무수히 많은 분류를 할 수 있다. 또한 분류에 따라 로봇의 제어 내용은 모두 다를 것이며, 사용하는 명령어도 다르다. 사용자 입장에서 원하는 작업의 표준안을 템플릿으로 제공 받을 수 있고, 제공 받은 템플릿으로 작업 환경에 따라 필요한 부분만 수정하여 사용할 수 있으면, 사용자의 새로운 작업은 아주 쉽게 구현 가능할 것이다.

Fig.4는 이러한 용도로 설계된 템플릿의 데이터 포맷이다.

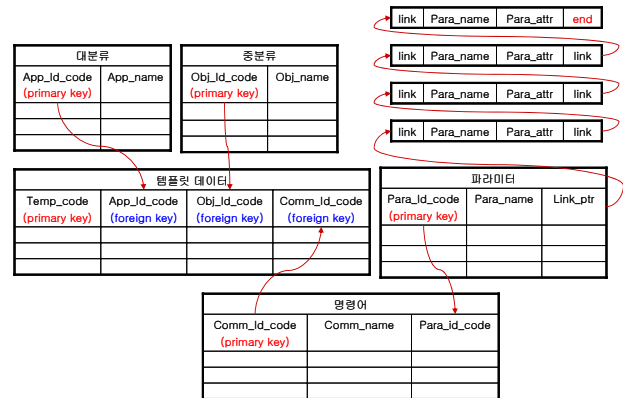


Fig.4 Data Format for Template

5. 결론

본 논문에서는 현재 산학연 협동으로 추진 중인 600kg급 초중량물 핸들링 로봇 개발 사업 내용 중 하나인, 로봇 운영 software 개발을 위하여 설계된 명령어 체계를 다른 용도의 로봇언어 개발을 위한 User-Interface로 제작하여 배포함으로써 로봇언어를 설계하는 도구로 사용할 수 있도록 하였다. 로봇 명령어를 응용 영역에 따른 용도별로 컴포넌트화 시켜 확장성과 호환성을 증가시키고, 템플릿을 제공함으로써 새로운 로봇을 개발하고자 할 때 쉽고 빠르게 개발 가능한 로봇언어 개발을 위한 사용자 인터페이스를 설계하여 구현하였다.

후기

본 논문은 산업자원부지원으로 지역산업기술개발사업으로 추진중인 ‘초중량물 핸들링 로봇 설계해석 및 제어기술 개발’과제로부터 지원받아 수행한 연구결과입니다.

참고문헌

- 오재홍, “로봇 개발용 소프트웨어 플랫폼에 관한 연구”, 중앙대학교, 석사학위논문, 2004. 12
- 하주한, 이기동, “자연언어 처리기법을 이용한 청소용 이동 로봇 자동 프로그래밍 시스템”, Journal of the Institute of Industrial Technology, Vol.32, No.1, 75-86, 2004.6