

AIM 조립셀을 위한 시퀀스 생성기술에 관한 연구

Research of Sequence Generating Technology for AIM Assembly Cell

*#강성복¹, 김형태¹, 김승택¹, 정훈¹, 김종석¹, 조영준¹
 *#S. B. Kang(sbkang@kitech.re.kr)¹, H.T.Kim¹, S.T.Kim¹, H.Jeong¹, J.Kim¹, Y.J.Cho¹
¹ 한국생산기술연구원 생산시스템연구부

AIM(Agile and Intelligent Manufacturing), Sequence, Script, Script Analyzer, Control Software Module

1. 서론

조립생산시스템에서 유연하고 민첩한 지능형 생산 시스템의 필요성이 증대 되면서 다품종, 저가격화 그리고 초소형 초정밀화가 요구되는 제품을 안정된 품질로 조립생산이 가능한 생산시스템이 요구되어지고 있다. 특히 각 제품의 특성에 따라 전용장비가 아닌 Module화 된 장비를 추가하여 공정특성에 적합한 Assembly Cell을 신속하고 빠르게 구축하려는 연구가 국내외에서 활발하게 연구되고 있다. 국내에서는 비록 생산 시스템 분야는 아니지만 RUPI(Robot Unified Platform Initiative) 라는 이름으로 로봇의 프로그래밍 환경을 통합하려는 연구가 진행되고 있다. OS나 언어에 종속되지 않고 다양한 형태의 로봇을 하나의 플랫폼으로 제어할 수 있게 하려는 연구이다. 이런 통합 플랫폼은 다품종 초소형 제품의 생산시스템에 또한 필요하다. 본 연구는 이런 민첩형 생산시스템(AIM System) 구축을 돕기 위해 필요한 시퀀스 생성기술과 VM기반 시뮬레이션에 관한 연구이며, 민첩형 생산시스템 중 공정 및 제품의 변화가 심한 소형 다품종 생산시스템 구축을 위해 로봇마다 동일한 인터페이스를 가지는 Virtual Machine을 구동하고 공통된 형식의 프로그래밍 방법으로 다양한 생산시스템에 적용 가능한 스크립트를 이용한 시퀀스 생성을 통해 소형 다품종 제품의 생산을 가능하게 하고자 하였으며, 각제품별 다양한 공정에 맞는 시퀀스 생성을 통해 생산 기간을 단축시키고자 하였다.

2. 전용장비와 AIM 장비의 비교

전용장비의 경우 특정 공정에 최적화 되어있고 생산속도가 빠르다. 그러나 현재와 같은 다품종 생산시스템에서는 새로운 공정에 적용이 어렵고 또한 제품단가 상승으로 인한 가격경쟁력에서 매우 뒤떨어진다. 공정에 최적화하여 기구부 설계를 하고

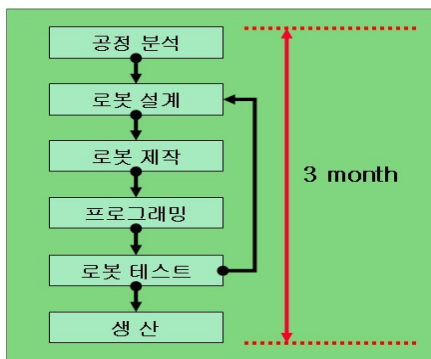


Fig. 1 Process Diagram of Exclusive Manufacturing System

제어소프트웨어를 제작했기 때문에 공정이 변경되면 로봇설계부터 소프트웨어 제작까지 모두 다시 설계해야한다. 이는 공정변화에 따라 소프트웨어 재설계를 통해 재활용하는게 아니라 새로운 로봇을 설계 및 제작함을 의미하며 또한 로봇의 재활용이 불가능하다는 것을 의미하기도 한다.

Fig. 1과 같이 공정분석 후 생산까지 모두 순차적으로 이루어지므로 제품생산 전 로봇기구부의 오류가 발생되면 로봇설계단계로 돌아가 처음부터 다시 설계과정을 거쳐야 한다. 따라서 제작기간이 AIM System 보다 오래 걸리며, 다품종생산에 필요한 빠른공

정변경이 불가능하다. 그로인해 시스템의 재활용이 어렵고, 잦은 공정의 변화에 따른 비용도 증가되어 생산설비의 채투자가 자주 발생하는 단점이 있다.

AIM System은 아래의 Fig. 2에서 보는바와 같이 공정분석 후 로봇설계 및 제작과 제어소프트웨어 즉 프로그래밍이 병행됨으로써 생산시스템의 구성이 빠르게 이루어지고, 전용장비와 달리 기존공정에 사용된 로봇의 재활용을 위한 모듈화와 제어 소프트웨어를 이용한 스크립트 생성을 통해 한달 안에 새로운 공정을 쉽게 변경 할 수 있는 유연성을 가지고 있다.

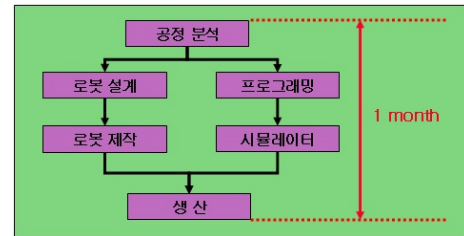


Fig. 2 Process Diagram of AIM System

3. AIM 제어소프트웨어를 이용한 시퀀스 생성 방법

3.1 시퀀스 생성 방법 및 구조

하위 레벨의 C++ 코드를 직접 구현하지 않고 제어 소프트웨어와 스크립트 분석기를 사용하여 사용자가 일상작업에서 사용하는 작업 중심의 자연어 스크립트를 정의하고 이를 이용하여 시퀀스를 생성한다.

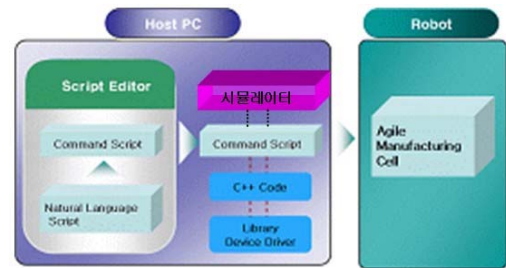


Fig. 3 Script Flow

Fig. 3에서처럼 스크립트 에디터에서 작업 중심의 자연어를 정의하고 이동경로를 입력하면 로봇제어언어에 가까운 명령어로 자동 변환해 주며, 제어소프트웨어 모듈을 통해 다양한 로봇을 제어하게 된다.

제어소프트웨어 구조는 크게 네가지로 나눌 수 있다. 자연어 스크립트와 명령어 스크립트를 생성하고 편집하는 스크립트 에디터, 명령어 스크립트를 인식하고 인수를 제어 소프트웨어 모듈에 전달하는 스크립트 분석기, 로봇 제어를 위한 구동 알고리즘이 포함된 제어 소프트웨어 모듈 그리고 제어소프트웨어를 통해 생성된 스크립트를 검증 및 시뮬레이션 할 수 있는 VM기반 시뮬레이터이다.

에디터를 통한 작업의 표현은 Table 1과 같이 일상작업에 사용되는 사용자 중심의 문장구조에서 동사를 강조하는 명령문 형태의 자연어이며 단어 및 구문분석기를 사용하여 이를 로봇제어언어와 유사한 명령어 스크립트로 변환된다.

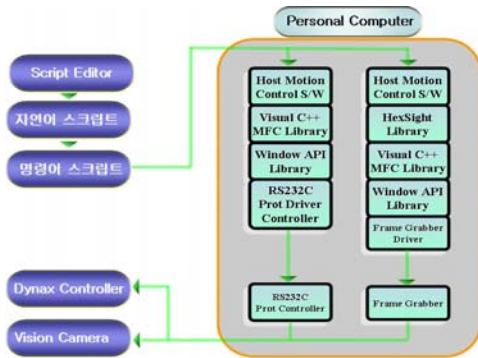


Fig. 4 Structure of Control S/W

스크립트 분석기는 명령어 스크립트의 명령어와 파라미터를 정확하게 구분 인식하여 제어 소프트웨어 모듈에 전달한다.

Table 1 Definition of Natural Language and Command Language

자연어	명령어	설명
Origin Return	ORGRT	초기위치로 이동
Approach	MOVEX, MOVEY	목표위치에 접근
	MOVEZ, MOBET	목표위치의 3차원(X,Y,Z) 좌표 입력
Pick-up	GRASP	부품 잡기
Move	MOVEX, MOVEY	목표위치로 이동
	MOVEZ, MOBET	목표위치의 3차원(X,Y,Z) 좌표 입력
Release	UNGRIP	부품 놓기
Identify	INPCT	부품의 판별
Align	ALIGN	부품조립을 위한 정렬
Insert	INSRT	부품을 목표 거리만큼 삽입

3.2 AIM 제어 소프트웨어를 이용한 시퀀스 생성 및 검증



Fig. 5 Programming Process of AIM Control S/W

제어소프트웨어를 이용한 시퀀스 생성과정은 공정분석을 통해 자연어 스크립트를 생성하고 생성된 자연어에는 목표 위치가 포함되어 있지 않으므로 작업을 수행하기 위해 이동해야 하는 위치 정보를 3차원 좌표(X,Y,Z)로 입력한다.

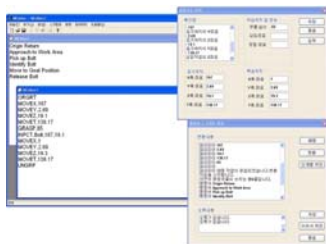


Fig. 6 Script Editor

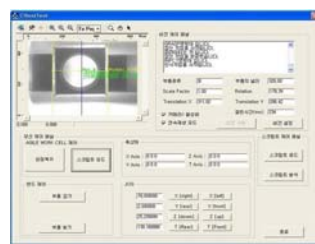


Fig. 7 Control S/W Module GUI

자연어 스크립트와 이동 경로를 저장한 후에 작업 Parameter를 포함한 명령어 스크립트로 변환하게 된다. 이러한 과정을 통해 자연어 스크립트에 축약되어 명시된 작업을 단위작업으로 풀어서 명령어를 생성하고 각 명령어는 제어 소프트웨어 모듈에 1:1 대응하여 로봇을 제어하게 된다.

스크립트 에디터와 제어 소프트웨어 모듈 GUI를 살펴보면 Fig.

6, Fig. 7과 같다

에디터는 작업을 생성 및 저장하는 에디터 기능과 이동경로 입력과 명령어 스크립트 변환 기능 등으로 이루어져 있다. 에디터 창을 통해 입력한 자연어 스크립트에 이동 경로를 입력한 후 명령어 스크립트로 변환 저장하게 된다. Fig. 7은 제어 소프트웨어 모듈의 GUI로 로봇의 동작시험을 위한 원점 복귀 기능과 각 축을 동작시킬 수 있는 JOG 기능을 포함하고 있다. 또한 비전 제어 패널에서는 카메라를 켜고 끄기 위한 아이콘과 HexSight를 동작시키고 설정을 불러올 수 있는 비전 동작과 비전 설정 버튼을 포함하고 있다. 스크립트 제어를 위해 명령어 스크립트를 불러오고 명령어 스크립트를 정해진 변수에 넣어 로봇을 제어하는 기능을 구현하였다.

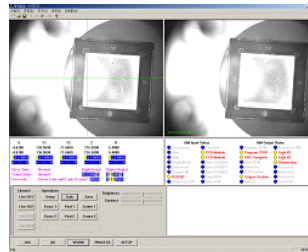


Fig. 8 Script Verification S/W



Fig. 9 AIM Simulator GUI

Fig. 8과 Fig. 9는 스크립트 에디터와 AIM 제어소프트웨어를 통해 생성된 스크립트를 검증하기 위한 스크립트 검증 소프트웨어와 VM기반의 로봇시뮬레이터이다. 스크립트 검증 소프트웨어는 스크립트 에디터와 제어소프트웨어를 통해 생성된 시퀀스를 다운로드하여 실제 로봇에 적용하여 검증하게 되며, 로봇의 설계 및 제작이 완료되지 않은 상태에서는 VM기반 AIM 시뮬레이터를 통하여 생성된 시퀀스를 다운로드하여 시뮬레이션하게 된다.

4. 결론

본 연구를 통하여 AIM 시스템을 제어하는 제어 언어의 확장성을 높이고 로봇의 설계 및 제작과 병행하여 하드웨어와 독립적으로 제어소프트웨어를 통해 시퀀스를 생성 및 검증하는 방법을 제안함으로써 다품종생산을 필요로 하는 AIM 시스템의 민첩성을 확보하고자 하였다. AIM 제어 소프트웨어를 통한 시퀀스 생성 및 검증, 시뮬레이션 방법은 C++ 코드와 비교하여 짧은 코드로 시퀀스를 생성하고 또한 검증할 수 있었다. 그 결과 다품종 대량생산, 저가격화 그리고 초소형 초정밀화가 요구되는 다양한 AIM 조립공정에 AIM 제어소프트웨어를 적용함으로써 숙련되지 않은 사용자가 쉽게 시퀀스를 생성 및 검증하고 빠르게 변화하는 모델 및 공정변경에 민첩하게 대응할 수 있을 것으로 예상된다.

참고문헌

- 1 Alfred A. Rizzi, Jay Gowdy, and Ralph L.Hollis, "Agile Assembly Architechure An Agent Based Approach to Modular Precision Assembly Systems", To appear in the Proceedings of the 1997 International Conference on Robotics and Automation
- 2 김화진, 조현보, 정무영, "An Intelligent Planner of Processing Equipment for CSCW-based Shop Floor Control in Agile Manufacturing", 한국경영과학회 학술대회 논문집, pp.185 ~ 192, 1995
- 3 Emanuele Carpanzano, Andrea Cataldo, Marco Dona, "Rapid prototyping test-bed of logic control solutions for reconfigurable manufacturing systems", 2005 IEEE
- 4 Chandimal Jayawardena, Keigo Watanabe and Kiyotaka Izumi, "Teaching a Tele-robot using Natural Language Commands", IEEE 2000