

◆ 특집 ◆ 첨단 생산시스템용 산업용 로봇 기술

산업용 양팔로봇 제어 SW 프레임 개발

Development of S/W Framework for the Industrial Dual-arm Robot

최태용^{1,✉}, 도현민¹, 박동일¹, 박찬훈¹, 김두형¹, 박경택¹
Taeyong Choi^{1,✉}, Hyun Min Do¹, Dong Il Park¹, Chanhun Park¹, Doohyung Kim¹, and Kyung-Taik Park¹

¹ 한국기계연구원 (Department of Robotics and Mechatronics, Korea Institute of Machinery & Materials)
✉ Corresponding author: taeyongc@kimm.re.kr, Tel: +82-42-868-7778

Manuscript received: 2013.7.9 / Accepted: 2013.7.31

Human rights at poor working condition is the severe problem in modern manufacturing system. The industrial dual-arm robot is being developed to meet these social issues fundamentally. The dual-arm robot can work instead of human workers. We developed the new dual-arm robot for manufacturing mobile phone and TV. It has advantages such as the solo controller for both arms, the human sized body and arms. The software platform for the industrial dual-arm robot is being developed which has strength in its convenience and intelligence compared to conventional the robot software platforms. Here the development of the dual-arm robot software platform is introduced.

Key Words: Dual-arm robot(양팔로봇), Assembly with robot(로봇기반 조립), Dual-arm robot controller(양팔로봇 제어)

1. 서론

기존 산업용 로봇의 경우 대표적으로 용접에 적용되어 큰 생산성 향상을 가져왔고, 이송이나 단순 조작 등에도 적용되어 왔다. 이런 산업용 로봇은 특수 목적에 맞게끔 전용 모델로 개발되거나, 범용성을 가진 경우에는 6축 시리얼 로봇이나 ADEPT로봇 형태 등과 같이 사람 팔과 유사한 모습으로 개발되어 왔다. 산업용 로봇은 기본적으로 고속·고정밀·고파워의 특징을 기반으로 단순·반복 작업에 주로 적용되어 왔다. 최근 이런 산업용 로봇의 한계를 극복하고 사람이 하는 복잡한 작업과 사람과 유사한 일을 하기 위해서 양팔로봇이 개발되고 있다.¹⁻³ 개발된 양팔로봇 들은 셀기반 조립라인에 주로 적용된 경우이며, 이 중 Yaskawa Motoman, Rethinkrobotics의 Baxter의 경우 이미 상

용화되어 판매하고 있고, ABB의 Frida는 상용화가 임박한 것으로 알려져 있다.

이런 기 개발된 양팔로봇들은 하드웨어적으로는 양팔로봇의 기본 기능에 적합하지만, 소프트웨어적으로는 한 팔 로봇의 확장에 지나지 않는 단점이 있다. 한 팔 로봇에서는 동일 궤적의 단순 반복이 주였기 때문에 궤적 교시만 하면 되었고, 연산량도 많지 않았다. 제어 소프트웨어는 임베디드형 하드웨어에 내장되어 판매되었고, 언어적으로는 전용 언어를 사용하였다. 이런 구조는 기본적으로는 산업용 로봇의 기본 역할인 단순 반복 작업을 정의하는 데는 충분하지만, 양팔로봇과 같이 양팔을 사용하는 복잡하고 외부 상황에 따라서 판단하고 동작을 하는 동작을 정의하기에는 부족하다.

본 연구에서는 이런 단점을 보완하기 위해서,

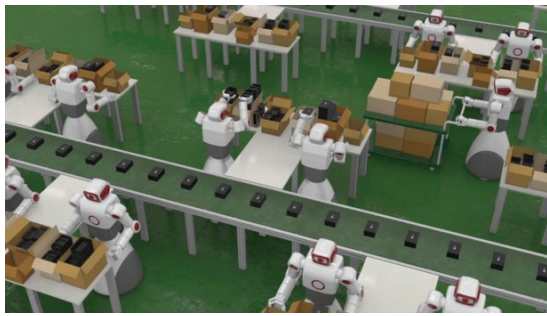


Fig.1 Cellular phone packaging process with the industrial dual-arm robot

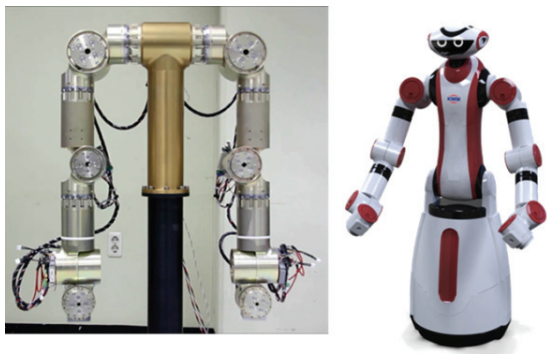


Fig. 2 Developed industrial dual-arm robot

개발 양팔로봇에 적용을 목적으로 양팔로봇 전용 언어를 만들고, 인지기반 동작을 가능하도록 하기 위해서 고수준의 알고리즘을 내장하도록 했다. 특히 고 수준의 알고리즘을 적용하기 위한 실시간 S/W프레임워크 구축에 대해서도 설명을 한다.

2. 배경

기존 산업용 로봇은 다양한 자동화 공정에 적용되어 생산성 향상에 기여해 왔다. 하지만, 기술적 한계로 인하여 그 적용처를 조립, 포장 등으로 확대하는데 한계가 존재했다. IT기술의 발달에 따라 조립, 포장 등의 공정 요구사항이 많아지고 있음에도 현재는 사람에게 의존하고 있는 상황이다. 이런 한계를 극복하기 위해서 하드웨어적으로 사람과 유사한 양팔로봇을 조립·포장 등의 공정에 적용하고자 하는 다양한 연구가 있고,^{4,7} 한국기계연구원에서도 휴대폰 포장, LCD TV조립용 양팔로봇 AMIRO를 개발하고 있다.⁷

Fig. 1은 휴대폰 포장을 위한 셀라인에 개발 양

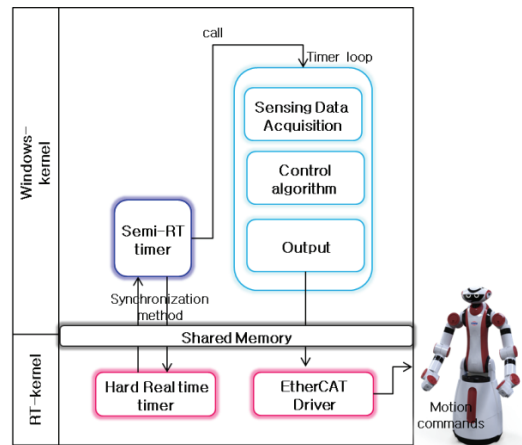


Fig. 3 Simplified structure of S/W framework

팔로봇을 적용한 것을 가정한 상상도이며, Fig. 2는 현재 개발중인 AMIRO를 보여주고 있다. 개발 로봇은 오른팔·왼팔 각각 7축의 시리얼 관절 구조를 가지며, 구동부의 경우에는 중공형 구조를 적용하여 유지보수가 쉽도록 구성하였다. 크기 또한 성인 남성과 유사한 크기로 제작하여 사람과 같이 일하는데 거부감이 없도록 했다.

3. S/W 프레임워크

3.1 제한사항

개발 양팔로봇을 운용하기 위한 소프트웨어는 기존 산업용 양팔로봇보다 많은 연산을 요구하는 알고리즘들이 사용된다. 목적으로 하는 IT제품 조립을 위해서는 실시간으로 대상물체의 위치를 인식하고, 양팔동시 동작시 양팔간의 충돌을 고려한 경로를 생성해야 하며, 양팔 14축 동기 제어를 수행 해야 한다. 더불어 개발 양팔로봇은 물건을 집을 때 떨어뜨리는 경우, 포장시 포장이 덜 된 경우 등의 공정상 오류를 모니터링하여 복구하는 기능까지 탑재된다. 이런 복잡한 알고리즘들은 정주기성을 지키며 수행되어야 하며, 기본적인 14축 동기 토크제어는 정주기성과 최소 1Khz이상의 제어주기를 가져야 한다. 개발 양팔로봇은 물체와의 접촉상황을 고려한 알고리즘이 다수 탑재되기 때문에 기본적으로 각 축은 토크제어를 사용한다.

개발하는 알고리즘은 요구기능이 많기 때문에 연산량이 많고 물리적 용량이 크다. 기능적으로 양팔 14축의 동기 제어뿐만 아니라, 14축에 대한 기구학, 양팔간의 충돌 모니터링, 대상물에 대한

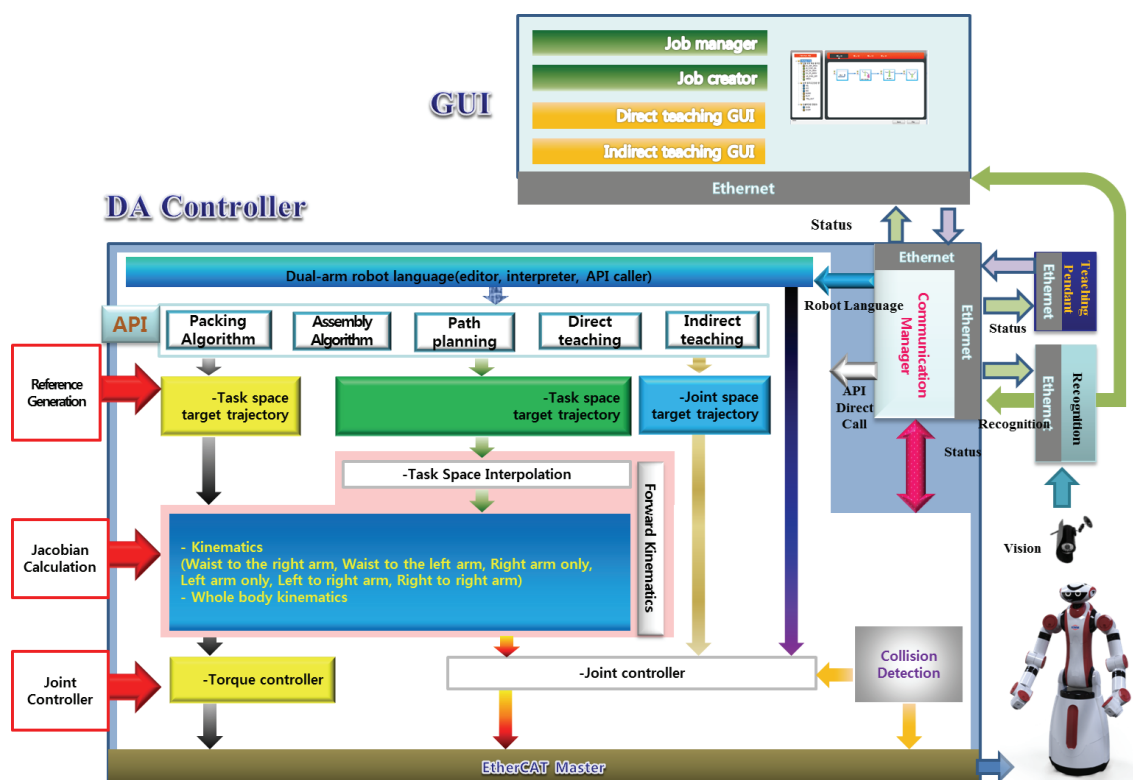


Fig. 4 Detail of the developed S/W framework. It contains many useful APIs for the dual-arm applications

영상처리, 오류상황 감지 등 무수한 지능형 기능들이 탑재되기 때문에, FPGA나 MCU등의 하드웨어에 탑재하는 것은 불가능하다. 본 연구에서는 필연적으로 PC기반 제어기를 적용하였다. 일반적으로 연구자, 개발자는 공통적으로 표준언어인 c/c++와 Microsoft windows 환경에 익숙하다. 이 중 Microsoft windows os의 경우에는 선점형 OS가 아니기 때문에 실시간성이 전혀 없다. 본 연구에서는 PC기반 제어기에서 실시간성을 확보하기 위해서 RTX기반 실시간 프레임워크를 구축하였다.

3.2 구현

실시간 코드 구현에 있어서 어려운 점은 실시간 코드이기 때문에 가지는 제한사항들이다. 예를 들어 실시간 코드에서는 메모리 allocation이나 release를 할 수 없다. 하지만, 일반적인 알고리즘 구현에서는 무수한 메모리 allocation과 release를 반복하며, 개발자가 변경할 수 없는 PC상의 시스템 library의 경우도 실시간성을 담보할 수 없는 것이 많다. 그래서, RTX⁸나 Intime⁹등의 상용 툴의 경우

시스템 library를 실시간성을 가진 것으로 새로 구성하여 제공한다.

본 연구에서는 realtime roboticslab 솔루션¹⁰을 활용하였다. Realtime roboticslab은 Fig. 3과 같이 내부에 RTX kernel을 포함하고 있다. Microsoft Windows 상에서 구동되는 RTX의 실시간 Timer를 인터럽트로 사용하여 윈도우 단에서 저 수준의 타이머를 구동하는 방식이다. RTX단에서 출력되는 타이머의 경우에는 1Khz세팅시 1%이내의 Jitter를 보장하지만, realtime roboticslab에서 구성한 windows os상의 timer는 10% 정도의 Jitter를 보여준다.

개발한 양팔로봇은 14축 제어 동기화를 위해서 EtherCAT 프로토콜을 사용한다. EtherCAT은 TCP스택의 실시간 버전으로, 별도의 카드를 필요로 하지 않고, 일반 Ethernet 카드를 이용하여 구성할 수 있는 장점이 있다.^{11,12} EtherCAT은 실시간 프로토콜이기 때문에, 실시간 OS상에서 구성이 되어야 하며, 본 개발에서도 EtherCAT driver는 RTX상에서 구성이 된다. EtherCAT으로 데이터를 주고 받기 위해서 RTX와 Windows사이에는 shared memory 구조

Table 1 Some examples of the dual-arm command

DA_CAP_JMOV	Simultaneous movement in the joint coordination of the both right and left arms
DA_CAP_MOV	Simultaneous movement in the cartesian coordination of the both right and left arms
DA_CP_LMOV	Both arm moves with keeping the relative distance between both arms
DA_CP_CMOV	Both arm moves around the goad position making a relative circle motion
LR_SYNC	Define the synchronization of the specific motions

Table 2 Some examples of the dual-arm command with Vision information

Pick_place_r (object, goal, time)	Pick the 'object' and put it in the 'goal' object using the right arm during 'time'
Pick_place_l (object, goal, time)	Pick the 'object' and put it in the 'goal' object using the left arm during 'time'
Delay(time)	Delay during 'time'

를 구성하여 데이터교환을 수행하였다.

좀 더 자세한 구성은 Fig. 4와 같다. 상위에서 사용자가 양팔로봇 명령어를 입력하면, 중간층의 API단에서 해석을 하고, 하위 단의 궤적 생성 및 움직임틀을 만들어 낸다. 오른쪽에는 로봇의 상태를 모니터링하는 영상처리 알고리즘이 위치해서 중간층의 API에서 영상정보를 요청하면 전송해 준다. API들은 양팔로봇의 공정에 특화된 포장·조립 등의 특화 API가 있으며, 교시관련 특화 API들로 구성된다. 최하부의 EtherCAT Master부분은 실제 RTX 단에서 구동하며, 그 상위 단은 semi-RT timer 영역에서 구동한다.

3.3 양팔로봇 언어

앞서 설명했듯이 양팔로봇은 기존 단일 팔과 달리 복잡한 작업을 수행하는 것이 보통이며, 그에 적합한 양팔로봇 언어가 필요하다. 특히 양팔 간의 동기화된 동작을 위해서는 사용자가 오른팔·왼팔을 각각 교시하는 것보다 내장된 명령어를 통해서 양팔의 상대적인 운동을 정의하는 것이 편하다. 개발 양팔로봇은 다수의 양팔의 상대운동을

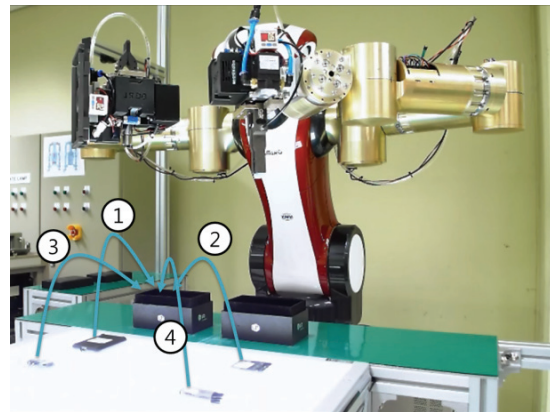


Fig. 5 Experiment with the developed dual-arm robot

정의하고 있어 교시시 매우 편리하다. Table 1은 그 중 몇 가지 예를 보여주고 있다. 기본적으로 오른팔, 왼팔 각각의 조인트를 입력하면 동시에 움직이도록 하는 동기화 조인트 명령어(DA_CAP_JMOV), Task space에서 양팔간의 상대거리를 유지하며 움직이도록 하는 명령어(DA_CP_LMOV)등을 구현하였다.

한편 영상정보를 이용한 명령어들도 구현하였다. 대상물의 ID와 목적지와 시간을 주면, 주어진 시간 동안 대상물을 집어서 목적위치로 옮겨주는 Pick_place_r, Pick_place_l같은 명령어도 구현이 되었다. Table 2를 참고하면 된다.

이런 명령어들은 개발된 S/W프로임워크 구조에서 가능한 것들이다. 복잡한 알고리즘을 요구하는 명령어 들이기 때문에, 기존의 PC가 아닌 임베디드 하드웨어에서 구현하는데 한계가 있다.

4. 실험

개발된 양팔로봇 AMIRO에 소프트웨어 플랫폼을 적용하여 간단한 실험을 수행하였다. 실험은 4가지 휴대폰 부품인 배터리 충전기, 배터리, 어댑터, 이어폰을 순서에 따라서 핸드폰 박스에 집어 넣은 것이다. 이 동작을 명령하기 위해서 개발된 S/W플랫폼에서는 Table 2에 주어진 명령어 3종만을 사용하였다.

실제 로봇이 4개의 물건을 삽입하는 과정은 Fig. 6에서 볼 수 있다. 순차적 명령어에 의해서 배터리 충전기를 영상정보를 이용하여 위치를 찾고, goal로 정해진 휴대폰 박스에 집어 넣는다. 이어서 순차적으로 배터리를 영상정보로 위치를 찾고 역

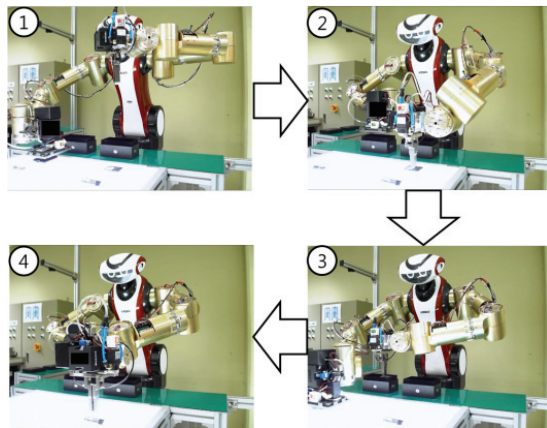


Fig. 6 Packing of the cellular phone parts

시 goal로 지정된 휴대폰 박스에 집어넣는다. 이렇게 해서 1-4까지의 과정을 거쳐 4개의 부품을 집어넣었다. 단순해 보이지만, 기존 단일 팔의 명령어 체계에서는 티칭 펜던트를 이용하여 각각의 팔의 궤적을 일일이 교시하고 양팔간의 동기화를 통하여 양팔동작을 이루어내는 복잡한 동작이다.

5. 결론

개발된 양팔로봇을 위한 소프트웨어플랫폼은 제어 실시간성을 유지하면서도 제어 알고리즘의 고수준화를 가능하게 하는 특징을 가지고 있다. 이런 개발된 소프트웨어 플랫폼의 특징을 이용하여 고수준의 지능형 양팔로봇 언어를 개발하고 이를 이용하여 복잡한 명령어 체계를 단순화 시킬 수 있었다.

참고문헌

1. ABB, "Dual-arm concept robot," <http://www.abb.com/cawp/abbzh254/8657f5e05ede6ac5c1257861002c8ed2.aspx.html>
2. YASKAWA Motoman robots, "Motoman Assembly Robot Models," <http://www.motoman.com/products/robots/assembly-robots.php#sthash.imtKmkUg.dpbs.html>
3. Rethink Robotic, "Baxter's Capabilities," <http://www.rethinkrobotics.com/index.php/products/baxter.html>
4. Krüger, J., Lien, T. K., and Verl, A., "Cooperation of human and machines in assembly lines," CIRP

Annals - Manufacturing Technology, Vol. 58, No. 2, pp. 628-646, 2009.

5. Sugi, M., Nikaïdo, M., Tamura, Y., Ota, J., Arai, T., Kotani, K., Takamasu, K., Seiichi, S., Suzuki, H., and Sato, Y., "Motion Control of Self-Moving Trays for Human Supporting Production Cell "Attentive Workbench"," Proc. of the IEEE International Conference on Robotics and Automation, pp. 4080-4085, 2005.
6. Morioka, M. and Sakakibara, S., "A new cell production assembly system with human-robot cooperation," CIRP Annals - Manufacturing Technology, Vol. 59, No. 1, pp. 9-12, 2010.
7. Park, C. H., Kyung, J. H., Choi, T. -Y., Do, H. M., Kim, B. -I., and Lee, S. -H., "Design of an modular actuation module for a dual arm robot manipulator," Proc. of Design of an modular actuation module for a dual arm robot manipulator, pp. 623-624, 2012.
8. IntervalZero RTX Real-Time Platform, "RTX at a glance," <http://www.intervalzero.com/products/rtx-at-a-glance.html>
9. Tenasys, "Intime for Windows," <http://www.tenasys.com/index.php/overview-ifw.html>
10. Simlab, "RealtimeRobotics," <http://simlab.co.kr/RealtimeRobotics.htm.html>
11. EtherCAT Technology Group, "EtherCAT Technology Group HOME," <http://www.ethercat.org>
12. Jansen, D. and Buttner, H., "Real-time ethernet the EtherCAT solution," Computing & Control Engineering Journal, Vol. 15, No. 1, pp. 16-21, 2004.