

PC 기반 제어를 위한 Flowchart 활용 프로그래밍 환경의 개발

이희원*, 김기원, 민병권, 이상조(연세대학교 기계공학부), 김찬봉(터보테크)

Flowchart Programming Environment for Process Control

Hi-Won Lee*, Ki-Won Kim, Byung-Kwon Min, Sang-Jo Lee, Chan-Bong Kim

ABSTRACT

For agile production methods, manufacturing system requires development of a motion controller which has flexibility of general-purpose motion controller and productivity of specialized-purpose one. In this study we developed the Flowchart Programming development environment for Motion language and Process Control. The controller designed on this environment can be used as a general purpose motion controller of a machining tool. Design of control programming based on a flowchart has the advantage of reducing the time consumed and intuitive interface for users. We create the solution with the Microsoft Visio for the flowchart-based platform and OPC for the process communication..

Key Words : Motion Language, Visio, Process Control (공정 제어), Flowchart Programming (순서도 프로그래밍), PC-based Control (PC-기반 제어), OPC (OLE for Process Control)

1. 서론

정보기술과 인터넷을 통한 제어 및 자동화 분야에서 응용이 작업의 효율화 및 유연 생산 등에 널리 활용되는 추세이다. Motion Language 와 공정관리 소프트웨어는 공작기계의 제어가 전용 생산기계의 제어기에서 점차 범용 Motion Controller 의 형태로 바뀌어 가는 최근의 추세에 적용시키기 위하여 그래픽 기반의 모듈형 프로그래밍 환경이 필요하다.

그래픽 기반의 Motion Control 언어에 있어서는, Microsoft 의 Visio 를 활용하였다. Visio 는 사용자가 원하는 개념, 정보, 시스템 등을 drag & drop 만을 사용해 시각적으로 표현하고 정보를 전달하는 데 유용한 다이어그램 등을 작성할 수 있게 해주며, 사용자가 정의한 도형에 사용자가 임의로 정보를 포함시키고 모듈화된 프로그램을 이용하여 다이어그램에 포함된 정보를 활용할 수 있다. 본 연구에서는 Visio 의 이러한 특성을 이용하여 사용자가 그래픽만을 이용하여 기계의 동작을 표현할 수 있는 프로그래밍 환경을 개발하고 있다.

Microsoft Visio 는 개념, 정보 및 시스템을 시각

적으로 표현하고 정보를 전달하는 데 유용한 다이어그램 등을 작성할 수 있는 솔루션이다. 이번 연구에서는 Visio 의 기본적인 기능만을 그대로 이용하는 것이 아니라, 특정한 셰이프(shape)와 스텐실(stencil)을 직접 제작할 수 있는 Visio 내의 사용자 지정 솔루션 개발 방식을 통하여 공작 기계 및 기타 산업용 설비들의 운동을 구현하고 제어할 수 있는 Motion Language Programming Environment Prototype 을 개발하는 것을 목표로 한다.

Motion Language 중에서도 전체의 공정을 제어하기 위한 방법으로 기존의 Ladder Logic 표현 방법보다 더 직관적이고 표현이 단순한 Flowchart 를 사용하여 공정을 표현하는 방법을 위주로 연구를 진행했다. Flowchart 를 이용한 제어 프로그래밍은 그 직관적인 특징으로 디자인 시간을 줄이고 이해도를 높이며 시각적으로 진단 기능을 활용할 수 있는 장점이 있다.

2. Flowchart Programming

2.1 Flowchart 의 장점

PC-기반과 flowchart 프로그래밍을 이용한 제어

시스템을 사용하므로 얻는 이익은 크게 세 가지를 고려할 수 있다. 빠른 디자인 사이클과 기계의 다운타임을 줄이고 프로세스의 점진적 개선의 가능성을 들 수 있다.

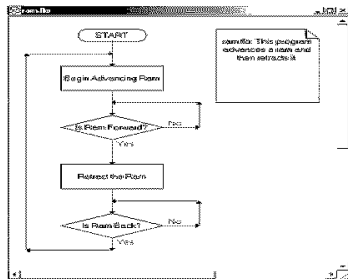


Fig 1. Flowchart Programming Environment

Flowchart 는 전체의 프로세스를 “Box”와 “Diamond” 블록의 조합으로 나누어서 수행하는 기능과 논리를 표현할 수 있다. 전체의 프로세스를 operation 을 수행하는 블록과 논리적 판단으로 분기와 선택적인 operation 을 만들어내는 블록으로 나누어서 단순화하고 그 내용을 기반으로 프로세스를 구성해 나감으로써 직관적인 제어 알고리즘을 구성할 수 있다.

단순히 직관적인 표현방법 이외에도 flowchart 프로그래밍을 통하여 제어 알고리즘의 설계와 구현을 용이하게 하는 이유는 재사용 가능한 모듈화된 프로그래밍 기법을 사용하기 때문이다. Flowchart 프로그래밍은 생성된 프로그램을 여러 경우에 재사용하기 용이한 기능을 제공한다. 동일한 혹은 유사한 operation 이 많은 제어 프로세스의 설계와 같은 경우에 기존에 생성되어 있는 코드를 잘 활용함으로써 개발시간과 개발비용을 단축할 수 있다. 이는 반복적으로 사용되는 코드의 신뢰도를 높이고 시스템의 적용도를 높이는 이점을 갖는다. 설계의 초기에 flowchart 를 통한 프로그래밍을 응용하여 function 들을 모듈화 하고 그 단위 별로 응용이 가능한 sub-program 을 통하여 구현될 수 있다. 이는 기존의 Ladder Logic 프로그래밍 방법보다 쉽게 이해될 수 있다.

Flowchart 프로그래밍 환경에서는 기본적으로 flowchart 를 이루는 블록들에 의하여 데이터와 프로세스가 캡슐화 되어있다고 볼 수 있다. 또 그 블록들을 큰 단위로 묶어서 super-block 을 구성하여 단위 모듈을 확장하는 기능을 통하여 모듈화를 가능하게 하며 이런 방법으로 sub-program 을 표현하는 모듈들로 전체의 프로세스를 구성할 수 있으며, 각각의 sub-program 들은 가지고 있는 파라미터를 통하여 입출력을 가능하다.

Flowchart 프로그래밍이 갖는 또 하나의 이점은

그래픽기반으로 이루어지는 프로세스의 상태 모니터링이 용이하다는 점에 있다. 프로세스가 흘러가는 상태가 PC 스크린을 통하여 직접적으로 표현됨으로 해서 기계의 상태를 조작자가 직관적으로 판단할 수 있도록 할 수 있다. 정해진 flowchart 에 따라서 기계 상태의 수정과 재제작 등이 가능할 수 있도록 한다. Flowcharts 가 직관적이고 이해해서 쉽기 때문에, flowchart 프로그래밍을 통한 PC-based 제어는 연속적인 과정 개선에 유용하게 사용될 수 있다. PLC 의 복잡함은 PC-based 제어의 단순함과 flowchart 프로그래밍 해법에 의해 바뀌어질 것이다

2.2 Visio 를 이용한 Flowchart 프로그래밍

정밀 공정 제어에 Microsoft 사의 Visio 를 기본적인 인터페이스의 기반으로 하는 연구를 진행하였다. Visio 는 일반적으로 각종 아이디어나 정보, 시스템을 시각화하여 의사전달을 도와주는 다양한 다이어그램 솔루션을 제공하는 프로그램이다. Visio 에서 만든 다이어그램을 사용하면 텍스트와 숫자만으로는 불가능했던 방식으로 분명하면서도 간략하게 그리고 효과적으로 정보를 가시화하고 전달할 수 있다. 또한 제공되는 다이어그램 솔루션뿐만 아니라 사용자가 자신만의 솔루션을 개발할 수 있는 환경을 제공한다.

이런 점을 이용하여 Visio 에서 VBA 프로그래밍을 통해 Visio Template 을 작성하였다. 이렇게 만들어진 Template 를 통하여 사용자에게 정밀 공정 제어에 필요한 인터페이스를 제공할 수 있다.

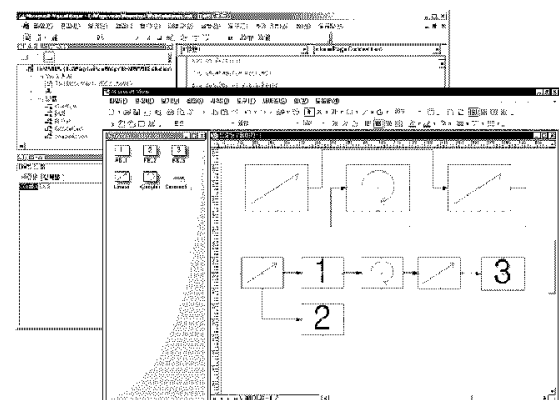


Fig 2. Visio Development Environment

Flowchart 를 Visio 상에서 구현하기 위해서는 flowchart 의 흐름이 Visio 상에서 가능하여야 한다. Visio 는 open architecture 로 프로그램 개발을 위해 많은 개체들을 제공한다. 개체들은 flowchart 상의 관점에서 보면 크게 shape 개체와 connect 개체로 나누어져 있다. Shape object 는 Visio 상에서 일반적

으로 사각형 마름모 등의 형태를 가진 도형들을 말하고 connect object 는 shape object 들이 만나서 생기는 연결지점의 object 를 말한다.

본 연구를 통해서 flowchart 를 통한 정밀 공정 제어 환경을 위하여 Visio 상에서 shape 들의 실제적인 흐름이 만들고 판단문을 통하여 flowchart 가 논리적 값에 따라 흐름을 제어 할 수 있게 되었다.

3. Flowchart 와 OPC 의 연결 구현

3.1 OPC(OLE for Process Control)

OPC (OLE for Process Control)는 프로세스에 사용되는 다양한 장비들 간의 통신을 위한 국제 산업 표준이다. 이런 표준화는 장비의 모니터링 및 제어 소프트웨어의 개발이 하드웨어 제조사에 종속되어 발생하는 호환성, 상호 이식성과 같은 문제점들을 개선하기 위한 것이다. OPC 를 표준을 따르면 다른 업체 사이에도 하드웨어와 소프트웨어의 개발이 가능하고 엔드 유저에 따라 하나의 하드웨어에 다양한 소프트웨어 중 자신에게 맞는 제품을 선택하여 사용할 수 있다. 또한 제품의 개발, 유지, 개선이 쉽게 이루어 질 수 있다. OPC 표준의 적용은, 자동화 분야의 엔드 유저, 소프트웨어 개발사, 하드웨어 제조사에게 다양한 이득을 제공한다.

Flowchart 의 지속적인 개발은 OPC 규격에 따라 표준화된 장비와 이런 장비들을 이용하는 공정 제어에 쓰일 범용 정밀 제어 환경의 개발을 목표로 하여 진행되었다. Flowchart 방식의 정밀 공정 제어 역시 사용자에게 쉽고 직관적인 인터페이스를 제공할 것이고 OPC 사용은 실제 장비와의 연결을 쉽게 할 수 있다.

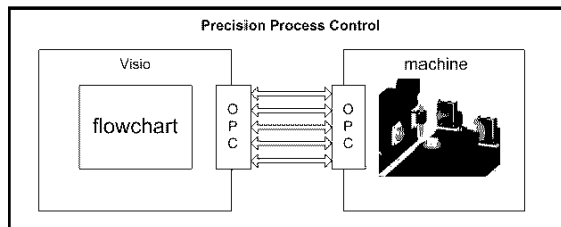


Fig 3. Schematic Diagram of OPC and Process Control

3.2 절 OPC 와 Flowchart 의 연결

VBA 프로그래밍을 통해 Visio 에서 기본적인 기능을 만들어 놓은 flowchart 에 각각의 shape 들이 OPC 장비와 대응할 수 있도록 연구가 진행되었다. 이를 위해서 OPC client 프로그래밍이 필요하다. OPC client 프로그램은 OPC server 로부터 데이터를 읽어오고 쓸 수 있는 것을 기본 기능으로 한다.

Flowchart 를 통해 OPC 장비를 연결하는 것도 하나의 OPC client 프로그램을 만드는 것이다.

OPC 의 적용을 위해 많은 OPC 툴들이 사용되고 있는데 본 연구에서는 Softing 사의 OPC client DA Active X 컨트롤러를 사용하여 다음과 같은 프로그램을 개발하였다.

(1) Visio 와 OPC 의 초기화 : Visual Basic 에서 Microsoft Visio 와 OPC 를 초기화 하고 작동 가능한 상태로 만들어 준다.

(2) 프로그램 인터페이스 : OPC server name 과 Remote Host 를 입력하여 사용자가 OPC 를 세팅하는 부분과 OPC item 을 모니터하고 OPC Tree 구조에서 Visio 로 chart block 을 생성하는 2 개의 부분으로 구성되어 있다.

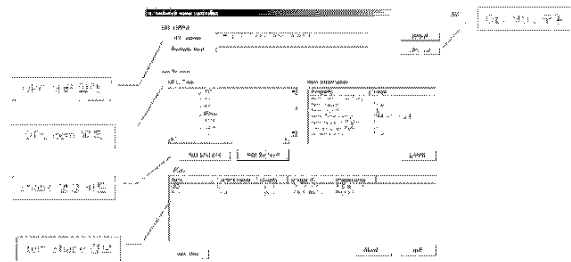


Fig 4. OPC Connection Programming

Softing OPC Active X 는 네트워크에 존재하는 OPC 를 검색하게 해주는 기능을 가지고 있다. 이 기능을 이용하여 네트워크상에서 원하는 OPC server name 을 선택하여 server 와 client 에 해당되는 flowchart 를 연결한다. 원격지의 OPC server 도 IP 입력을 통해 OPC server 와 연결이 가능하다.

Flowchart 는 현재 하나의 서버만 연결이 가능하게 개발되었다. 실제 공정을 제어하기 위해서는 여러 대의 OPC server 와 연결하여 제어할 수 있어야 한다. 복수의 OPC server 와의 연결은 앞으로의 프로그램 개선을 통해 구현될 것이다

원하는 item 을 선택한 후 OPC Tree 하단에 있는 블록 생성 버튼을 클릭하면 Visio 상에 선택된 형태의 블록이 생성된다. 현재 flowchart 에서는 이와 같은 방식으로 process block 과 decision block 의 두 가지 블록을 만들 수 있다.

OPC DA ActiveX 는 opcData_OnDataChanged 라는 함수를 제공하는데 이는 OPC server 의 item 값이 바뀌면 client 에서 item 값이 변했음을 알려주는 기능을 가지고 있다. 이 함수를 이용하여 Visio 상에서 shape 들을 통해 server 에서 바뀐 item 값을 모니터링 할 수 있게 하였다. 실제 정밀 공정 제어를

위해서는 장비와 연결된 OPC item 의 값을 읽어오는 것은 매우 중요하다. 사용자가 장비를 모니터링하는 것만 아니라 센서를 통해 공정 제어를 할 경우 분기문의 논리 값과 연결된 OPC Item 의 값을 제대로 읽어 와야 정확한 공정 제어가 가능한 것이다.

이미 판단문에 의한 flowchart 의 분기는 앞서서 만들어진 상태이다. 판단문으로 만들어진 OPC item 은 server 로부터 Boolean 값을 읽어와 그 값에 따라 flowchart 를 분기한다. 아래의 그림은 increment, BOOL 이라는 OPC item 의 값에 따라 공정이 분기되어 진행 되는 것을 보여준다.

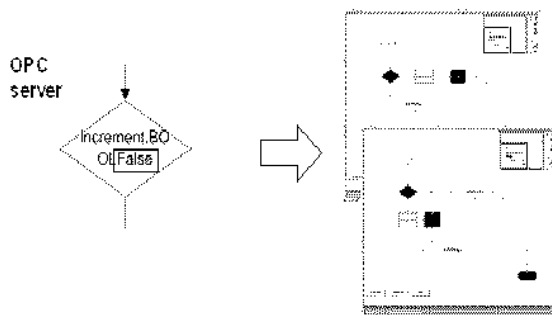


Fig 5. Branch by Decision block and OPC Data

OPC 를 연결하고 OPC item 을 Visio 상에 shape 으로 만들어 주는 프로그램은 앞서 만들어 놓은 Visio flowchart template 과 연결되어 있다. 프로그램을 실행하고 OPC item 으로부터 생성된 블록들을 원하는 공정에 맞게 연결한 후 하단의 실행 버튼을 클릭하면 flowchart 방식으로 공정이 실행 된다. 아래 그림의 예는 간단한 형태의 공정을 나타낸다. 각각의 OPC item 들 서버로부터 item value 를 받아와 Visio 상에서 변화되는 값을 보여준다. 따라서 Visio 화면 상에서 전체 공정이 진행되는 상태를 모니터링 할 수 있다.

4. 결론

IT 기반 제어시스템의 모듈형 S/W 기술을 개발하는데 필요한 Motion Language Programming 개발 환경을 구현하였다. Flowchart 를 이용하여 제어 프로그램을 디자인하는 방법은 그 직관적인 인터페이스를 통하여 기존의 Ladder Logic 보다 디자인의 소요 시간을 줄이고 가독성을 높이는 효과를 나타낸다. 또 공정의 모니터링을 위한 방법으로 사용될 수 있으므로 flowchart 를 이용한 프로그래밍 방법은 PC-기반의 모듈형 제어 프로그램으로 유리한 측면이 많다. Visio 를 기반으로 flowchart 를 작성하는 프로그램을 개발하였다. Visio 의 Shape 들을 flowchart

에 응용할 수 있기 때문에 다양한 제어형태를 구상할 수 있다. 또 각각의 블록들을 모듈화 하여 사용이 가능하기 때문에 재설계나 수정의 경우에도 용이한 장점이 있다. OPC 와 flowchart 의 연동을 통하여 OPC 환경으로 구성된 제어시스템을 쉽게 flowchart 에 연동시키도록 하였다. Visio 의 Shape 와 연동된 OPC Data 들은 flowchart 를 통하여 관리되고 제어하는 것이 가능하다. 이런 방식으로 Visio 에서 Flowchart 를 이용하고 OPC Data 를 연결할 수 있는 client 를 포함하는 프로그래밍 환경을 구축하였다. Flowcharts 가 직관적이고 이해해서 쉽기 때문에, flowchart 프로그래밍을 통한 PC-based 제어는 PLC 와 Ladder Logic 의 대안이 될 수 있을 것이다.

후 기

본 연구는 산업자원부에서 지원하는 중기거점사업 “IT 융합형 나노제어기 개발사업”의 지원에 의해 수행되었습니다.

참고문헌

1. Norbert Völker, Bernd J. Krämer , “Automated verification of function block-based industrial control systems,” Science of Computer Programming 42 , pp. 101-113, 2002
2. Euisu Park, Dawn M. Tilbury, Member, IEEE, and Pramod P. Khargonekar, Fellow, IEEE, “Modular Logic Controllers for Machining Systems: Formal Representation and Performance Analysis using Petri Nets”, IEEE Transactions on robotics and automation, vol. 15, No. 6, 1046-1061, December 1999
3. Naehyuck Chang, Wook Hyun Kwon, Jaehyun Park, “Hardware implementation of real-time Petri-net-based controllers,” Control Engineering Practice 6, 889-895, 1998
4. Automation, Production Systems, and Computer-Integrated Manufacturing 2nd , Mikell P. Groover, , 2001, Prentice Hall
5. Yeonsub Jung, Poonghyun Seong, Mancheol Kim, “A model for computerized procedures based on flowcharts and success logic trees,” Reliability Engineering & System Safety, vol. 26, 351-362, 2004
6. 황봉하, 김성기, 이희원, 박성준, 민병권 “IT 기반 제어시스템의 모듈형 S/W 기술개발”, 1 차년도 IT 기반 나노제어시스템 개발사업 Workshop, 2003
7. OPC - Fundamentals, Implementation, and Application 2nd rev., 2002, Huthig Verlag Heidelberg