공작기계 고장 실시간 추적 · 진단기술 (개발사례 및 특징)

김동훈 | 한국기계연구원

1. 개 요

본 기술은 기존의 비효율적인 공작기계 유지보수 문제점을 인식하고 최근의 공작기계의 컨트롤러 환경이 PC 기반의 개방형 타입으로 변화하고 사용자 정의 응용모듈의 실장이 가능하게 되는 추세를 감안하여, PLC와 CNC에 의해 순차제어되는 공작기계의 운용 및 조작상의 고장을 정의하고 이의 발생 원인을 신속하고 효과적으로 찾아 초보 작업자나 내외부의 보수요원에게 실시간으로 제공하는 개방형 공작기계의 고장원인 추적 · 진단시스템 및 그 방법을 제공한다.

2. 기술 특징

본 기술은 공작기계의 고장 진단 시스템 및 방법에 관한 것으로 PLC순차 제어 정보를 나타내는 PLC로직으로부터, 각 출력신호가 어떤 입력신호와 출력신호의 연산관계에 있는지를 알려주는 스위칭 펑션 리스트를 형성하는 단계; 상기 형성된 스위칭 펑션 리스트를 결합하여 특정 출력에 대한 원인이 되는 입력신호와 출력신호의 관련성을 나타내는 수식을 도출하는 스위칭 펑션 형성 단계; 상기 스위칭 펑션 단계에서 형성된 수식을, 연산이 수행되는 단계별 수식으로 분해하여, 상기 각 단계별 수식을 형성하는 스텝 스위칭 펑션들을 형성하는 단계; 상기 형성된 스텝 스위칭 펑션을 역순으로 추적하여 고장원인을 도출하는 단계를 포함하는 것을 특징으로 한다.

이상과 같은 본 기술을 이용하면, PLC와 CNC에 의해 순차제어되는 공작기계의 운용 및 조작상의 고장의 원인을 신속하고 효과적으로 추적하여 초보 작업자나 내외부의 보수요원에게 실시간으로 제공할 수 있다. 표 1에 개발된 공작기계 고장원인 검출을 위한 추적 · 진단시스템의 특징을 제시하였다.

3. 기술동향 및 차이점

최근 공작기계의 컨트롤러 환경이 PC 기반의 개방형 타입으로 변화하고 사용자 정의 응용모듈의 실장이 가능함에 따라, PLC와 CNC에 의해 순차제어되는 공작기계의 운용/조작상의 고장을 정의하고 이의 발생 원인을 효과적

으로 찾아 초보 작업자나 내외부의 보수요원에게 실시간으로 제공해야 하는 필요성이 증대되고 있다.

일례로 고장이 발생하거나 기계가 작동을 하지 않을 때, 최근에 도입된 CNC는 알람 발생 등 어느 정도 자체적인 감시진단 기능을 가지고 있지만, 이는 부분적이며 제한적이다. 다시 말하면, Emergency 알람의 경우에만 보더라도, E-Push 버튼, Z축 스테이지가 모션 운동 범위를 벗어나 위치함, 공작 기계의 Cycle Start가 안되는 CSD 알람, 잘못된 운용모드 설정 등 비숙련 작업자의 운영/조작상 모든 발생 가능한 오류 등의 다양한 원인에 의해, Emergency CNC 알람이 울리게 된다.

따라서, 이러한 다양한 가능성 중에서 이 Emergency 알람의 원인을 찾아서 복구 조치를 하기 위해서는 경험이

표 1. 본 연구에서의 고장원인 검출 · 진단내용과 기존 방법의 비교

	예전방법	최근방법	본 연구의 개선된 방법	참조
배경	– 과거CNC가 폐쇄형 타입이라서 (예 : FANUC) 고장 종류 및 진단이 제한적 – 벤더 종속적 방법만 사용	– 현재 CNC가 개방형 구조를 가진 CNC로 대체중, 안정성 확보 필요 – CNC자체 고장과 CNC와 PLC사이는 고정어드레스 영역만 책임을 두고 안정화 진행 중	– CNC 자체 고장(30%) 뿐 아니라 전체 고장의 70%이상을 차지하는 운용상 발생고장을 중심으로 설계 – 고장종류 및 그 원인을 진단하여 현장자체보수 및 원격관리 지원	– CNC 벤더, 공작기계 벤더, 현장작업자의 고유 영역 및 안정성 때문에 등한시 되던 운용상 / 조작상 고장의 종류 및 특히 고장원인을 상세히 진단해 주는 기능이 CNC에 내부기능으로 실장된 특징 – PLC의 출력신호가 다시 입력으로 순환되는 등 10가지 정도의 PLC와 CNC사이의 모든 입출력 논리적 관례를 모델화 (SF, SSF 진단모델) 하는데 어려움이 있었으나, 이를 해결하였음 〈출력 2개가 다시 입력 예〉 *G9.00 = X0.00+R0.00* *R0.00 = X1.00+G9.00 +R0.01* *R0.01 = X2.00+R0.00*
특징	– CNC의 기능을 단순화시켜 고장의 발생 가능성 최소화 – 감지센서, 제어릴레이, 보조 액츄에이터 등의 연결시 CNC 벤더의 전문요원의 투입이 필요 – 고장발생시 블랙박스 타입의 컨트롤러라서 대부분 경우에 현장에서의 자체해결이 어려움 – 대부분 외부보수요원 투입 해결	– 감지센서, 제어릴레이, 보조액츄에이터 등의 불량 및 파손으로 운용중 고장이 빈번히 발생 – 특히 CNC벤더의 고정어드레스 영역외, 현장에서 임의로 추가 수정되는 독자어드레스 영역에서 발생 원인고장은 CNC자체에서 해결이 안됨 – 초보작업자의 조작오류도 빈번히 발생 – 1차적으로 PLC/CNC 숙련된 경험자가 해결, 2차적으로 외부보수요원 투입 해결	– 운용상 발생하는 머신의 무작동(Disability) 상태를 대상 고장으로 정의 (예 : ESE/CSD/MRD) – 그 외의 고장의 종류를 임의로 등록 및 수정할 수 있도록 다중화 – 고장 발생원인은 CNC와 PLC에 연관된 모든 입출력 신호가 대상 – 감지센서, 제어릴레이, 보조 액츄에이터 등의 불량 및 파손을 즉시 검지 및 그 원인을 정확히 통보 – 초보작업자의 조작오류로도 즉시 검지해 원인과 대처방안 제시 – 초보작업자도 현장에서 해결 및 원격지 이력관리도 가능	
대응시간 (100기준)	– CNC 고장과 운용조작상 고장 모두 외부보수요원 의존도 100	– 고장원인 파악시간 : 80 – 고장복구 조치시간 : 20	– 고장원인 파악시간 :10 – 고장복구 조치시간 : 20내	– 신속한 현장자체해결 – 원격기보수 병행가능
	Turbotek 창원공장 CNC 공작기계 A/S 센터 / 화천기계테스트 기준			

풍부한 숙련된 작업자나 외부 보수 요원의 투입이 필요하며, 그 때까지 공작기계는 무작동 상태로 다운될 수 밖에 없다.

종래의 공작기계 분야에서의 고장진단 시스템들은 공작기계 제작업체나 CNC 제작업체인 공급 벤더(Vendor)에서 자사의 제품들을 홍보하기 위한 일종의 전략으로 제공되기 때문에, 이러한 고장 진단 시스템의 기능이 제한적이며 일반적이지 못하다. 즉, 공작기계 제작업체는 기계에 중심을 두고, CNC 벤더는 컨트롤러에 중심을 두기 때문에, 고장의 진단보다는 모니터링하고자 하는 일부 정보를 설정하고 이것만 외부에 보여주는 시연 행사에 초점을 맞추는 정도이다. 이러한 이유는 산업이 세분화/전문화되는 현실에서 공작기계 벤더나 CNC 벤더가 각자의 영역에서 다른 영역으로의 융합이나 통합 작업을 하기가 어렵기 때문이다.

따라서 실제 공장이나 생산시스템 현장의 공작기계는 메이커인 밴더로부터 도입된 후, 그대로 사용되는 것이 아니라, 현장에서의 운용 목적에 따라 공작기계의 컨트롤러와 외부 보조장치나 수많은 센서 등과의 인터페이스 작업 등을 수행하게 되므로, 공작기계의 환경이 달라지게 된다. 이에 따라 감시 및 진단해야 될 데이터도 달라지게 된다.

결국, 현장의 특성에 따라 달라지는 이러한 실질적인 정보와 PLC와 CNC의 모든 내/외부 입출력 신호정보인 현장의 특성정보를 가지고 있는 PLC 래더를 기반으로 일반적이며 보편적인 방법으로 모델링하고 추출하며, 이를 바탕으로 고장이 발생했을 때, 고장의 종류 뿐만이 아니라 그 고장의 원인을 정확하며 신속하게 찾아서 내/외부의 관련자에게 알려주는 체계가 필요하다.

4. 구성 및 원리

먼저, 공작기계에서의 고장의 종류와 진단 절차에 대해 설명하기로 한다. 공작기계의 고장은 구성부품의 노화에 의한 기능저하와 운용상 발생하는 고장으로 분류가 가능하다. 본 연구에서는 운용상 발생하는 고장만을 다루고 있으며 이는 크게 두 가지로 나눌 수 있다.

첫째, CNC 고장

둘째, PLC 및 조작 · 운용상 발생되는 고장

CNC의 고장은 대부분의 CNC가 자기 진단기능을 가지고 있기 때문에 본 연구에서는 이를 이용한다. PLC의 운용상 발생하는 고장은 구조적 고장 (Structural fault), 조작상의 고장(Operational Fault) 그리고 조건상의 고장(Conditional Fault)으로 나눌 수 있고, 이러한 고장은 PLC의 순차제어 논리를 통해 진단이 가능하며 PLC상에서 발생되는 고장 중에서 70%가 조작상의 고장으로 보고 되고 있다. 이러한 PLC의 운용상 발생한 고장은 고장이 발생하여 가동이 정지되는 시간의 80%는 고장의 위치를 찾는데 그리고 20%가 고장을 수리하는데 소요된다. 따라서, 조작상의 고장을 효과적으로 정확하게 그리고 빨리 그 원인을 진단하는 것이 고장진단에 있어 가장 중요한 기술이라 할 수 있다.

따라서, 본 연구에서는 공작기계가 작동이 불가능한 상태를 조작상의 고장(Operational Fault)이라 정의했으며, 다음과 같이 3가지로 정의했다.

① ESE(Emergency Stop Error) : 비상정지가 발생하는 경우
② CSD(Cycle Start Disability) : 기동이 되지 않는 경우

③ MRD(Machine Ready Disability) : 기계준비가 되지 않는 경우

공작기계에서 고장의 원인을 진단하는 절차를 그림 1에 나타내었다.

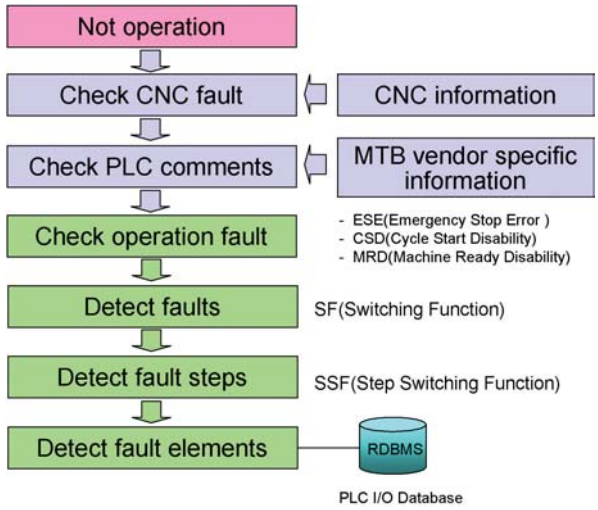

그림 1. 고장의 종류에 따른 진단 과정

공작기계가 고장이 발생하여 조작이 불가능한 상태가 되면 고장을 진단하게 된다. 고장진단은 CNC 고장부터 진단하게 되는데 CNC 고장인 경우 CNC로 부터 고장상태를 전달받게 된다. CNC 고장이 아닌 경우, PLC의 코멘트를 확인하게 된다. 이는 공작기계 메이커에서 특정기능에 대한 고장을 정리한 것으로서 이를 확인하게 된다. 이러한 고장이 아닌 경우, 운용상 발생하는 고장으로서 개발된 고장원인 추적진단 시스템(Fault Diagnosis System:FDS)을 통해 고장을 진단하게 된다.

FDS에서는 SF(Switching Function) 모델을 통해 고장이 발생한 항목을 찾은 후, SSF(Step Switching Function) 모델을 통해 고장이 발생한 스텝을 찾게 된다. 고장이 발생한 스텝을 찾은 후에는 PLC 입출력 DB 베이스를 이용해 고장이 발생한 요소를 찾게 된다. 결국, 운용상 발생하는 고장을 지능적으로 진단하기 위해서는 순차제어정보로부터 고장항목에 대한 연관성을 제공하는 SF와 SSF를 만들어야 한다.

순차제어정보는 PLC 로직으로 이루어진다. PLC 로직은 IL(Instruction List), ST(Structured Text), FBD(Function Block Diagram), LD(Ladder Diagram) 등 몇가지 표현방법이 있는데 LD가 가장 일반적으로 사용이 되고 있다. CNC의 순차제어에 사용되는 LD로부터 SF와 SSF의 생성은 SFG(Switching Function Generator)에 의해 형성된다.

이제, SF와 SSF모델에 대해 자세히 설명하기로 한다.

CNC 공작기계의 운전중에 발생하는 ESE, CSD, MRD의 원인을 진단하기 위해서는 순차제어를 수행하는 래더 다이어그램(LD)을 기반으로 해야 한다. 그러나 래더 다이어그램으로부터 고장을 진단하기 위해서는 많은 경험을 가진 보수 기술자를 필요로 한다. 본 연구에서는 전문가를 대신하여 지능적으로 고장을 진단하기 위한 효과적인 방법으로 입출력 신호간에 논리적 관계를 가지는 두가지 진단모델 SS, SSF를 제시한다.

이렇게 진단 모델을 만드는 주된 이유로는 각 공작기계 메이커마다 기계를 구성하는 방법에 고유한 특성을 가지기 때문에, 진단 방법에 대한 일반성을 부여하기 위해서이다. 두 진단 모델은 FDS의 고장진단 엔진으로 사용된다.

① SF(Switching Function) 모델

공작기계의 고장을 진단하기 위해 고장항목으로 정의한 원인을 논리적 모델로 표현한 것이다. 즉, 고장의 원인이 된 항목이 발생하기까지 관련된 모든 신호의 관련성을 나타내는 모델이다. 여기서 논리란 각 신호에 대한 'AND', 'OR', 'NOT' 등으로 구성된다. SF는 고장을 진단하기 위한 정적인 모델로서 이를 만족시키는 조건들은 많은 경우가 있을 수 있다. PLC가 하나의 동작을 완성시키기 위해 만족해야 하는 조건은 수학식1과 같이 정리할 수 있다.

$$c(t) = c_1(t) \cdot c_2(t) \cdot c_3(t) \cdots = \prod_j c_j(t) \qquad \text{(수학식1)}$$

여기서 c(t)를 t번째 스텝의 모든 조건의 조합상태라고 가정하고 t번을 마지막 스텝이라고 가정하면 c(t)를 스위칭 함수(Switching Function)라 정의한다.

② SSF(Step Switching Function) 모델

SF는 고장항목에 대한 정적모델로서 동작상태의 동적변화를 표현 할 수 없다. SSF는 고장이 발생한 원인을 만족시키는 조건들을 단계적으로 표현하여 고장이 발생한 스텝을 찾기 위한 모델이다. SSF는 동작 중 고장이 발생한 스텝을 효율적으로 찾을 수 있는 동적모델이다.

PLC로 제어되는 CNC 공작기계의 순차제어는 시퀀서(Sequencer)에 의해 수행된다. SSF는 기계상태와 시간순서에 따른 상태변화로 구성되어 있으며, 순차적인 기계작동 상태 변화를 기술한다. 각 스텝에서의 동작은 단순히 현재 스텝에서의 제어 명령에만 관계된 것이 아니라, 이전 스텝의 조건에도 관계된다. 현재 스텝은 이전 스텝이 종료되고 현재 제어명령이 전달된 후에 수행된다. 어떤 스텝이 종료되었는지 여부는 그 스텝 조건에 의해 결정된다. 따라서 SSF는 다음과 같이 구성될 수 있다.

식 (1)에서 c(t)가 만족하려면 가 순차적으로 만족을 해야한다. 즉, 은 그 스텝의 조건을 만족하고 다음 스텝이 시작될 수 있다는 조건을 의미한다. 는 그 스텝조건을 만족하지 못하였고, 다음 스텝을 실행할 수 없다는 것을 나타낸다. 따라서 각 스텝이 만족해야할 조건이 되는 를 스텝 스위칭 함수(Step Switching Function)라 정의한다.

다음으로, 고장진단 모델 생성기인 SFG(Switching Function Generator)에 대하여 설명하기로 한다.

PLC는 순차제어에 사용되는 릴레이(Relay), 타이머(Timer), 카운터(Counter) 등과 같은 기능을 마이크로프로세서를 이용해 통합시킨 장치이다. PLC에서 가능한 데이터 처리는 산술연산, 논리연산, 함수연산, 조절연산 등의 처리가 가능하기 때문에 CNC 공작기계에서도 순차제어를 담당하는 중요한 역할을 한다. CNC공작기계의 예가 그림 2에 나타나 있다.

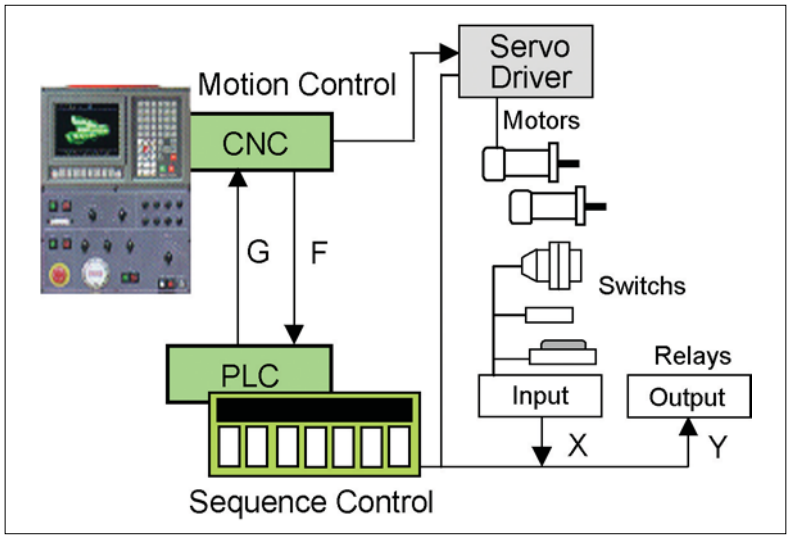

그림 2. CNC공작기계의 구성도

공작기계를 제어하는 CNC는 다시 그 기능을 나누어 보면 CNC와 PLC로 분리된다. CNC는 보간과 같은 운동제어(Motion Control)가 주된 기능이며, PLC는 각종 순차제어를 담당하게된다. 두 가지의 제어장치간에는 유기적인 신호처리를 수행하면서 기계를 동작시키게 된다. PLC는 프로그램 작성방법이 순차적인 제어에 의존하기 때문에 다음과 같은 두 가지의 불편한 점이 있었다.

첫째, 프로그램 구조가 스텝 구조를 가지기 때문에 프로그램을 개발할 때 발생하는 프로그램의 오류를 찾는데 어려움이 있었다. 즉, 프로그램의 디버깅(Debugging)과 변경(Modify)에 많은 어려움이 있었다.

둘째, 고장이 발생했을 때 고장의 원인이 되는 논리관계를 찾기 위해서는 많은 경험과 시간을 필요로 한다는 것이다.

위에서 정의한 SF 및 SSF를 수작업에 의해 구한다면 경험을 가진 전문가의 도움이 필요하게 되며 오류를 발생시킬 위험성이 높다. 본 연구에서는 SFG 개발을 통해 SF, SSF를 생성한다. SFG 시스템의 작업순서가 그림 3에 나타나 있다.

1단계는 LD(Ladder Diagram)을 BD(Binary Data)로 바꾸는 과정이다. 2단계에서는 BD를 IL(Instruction List)로 변환을 하고, 3단계에서는 2단계에서 해석한 IL을 이용해 각 출력신호들의 SF를 미리 정한 버퍼에 저장한다. 4단계

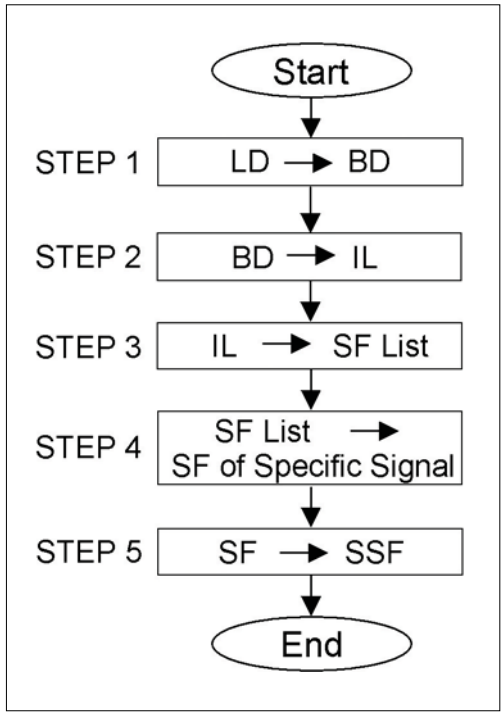

그림 3. SFG의 데이터 처리 흐름도

에서는 3단계에서 만들어낸 SF 리스트들을 이용해서 특정 고장에 대한 SF를 생성하게 된다. 5단계에서는 생성된 SF를 분해하여 SSF를 구하게 된다.

이제, SFG 시스템을 효과적으로 설명하기 위해 그림 4와 같은 예제 LD를 이용한다.

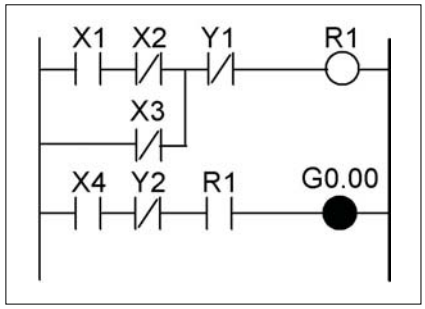

그림 4. 래더 다이어그램의 예

예제는 입력요소 X1, X2, X3, X4, 출력요소 Y1, Y2, 내부 릴레이 R1 그리고 고장을 정의하는 출력 코일 'G0.00'으로 구성된다. 논리구조는 2개의 'AND'와 2개의 'OR' 요소로 이루어진다. 'AND' 논리를 '*'로, 'OR' 논리를 '+'로 정리한다면 예제는 다음과 같이 표현이 가능하다.

R1=(((X1)*(-X2))+(-X3))*(-Y1)　　(수학식2)
G0.00=(X4)*(-Y2)*(R1)　　(수학식3)

위 두 식은 아래 식과 같이 줄일 수 있다.
G0.00=(X4)*(-Y2)*(((X1)*(-X2))+(-X3))*(-Y1)　(수학식4)

SFG는 수학식4와 같은 특정 고장원인에 대한 전후 논리관계를 생성하는 프로그램으로서, 다음과 같은 5단계로 이루어진다.

- 1단계 : LD -> BD

1단계에서는 LD로 표현된 순차제어논리를 BD로 바꾸는 일을 수행하게 된다. 그림 4의 LD를 BD로 전환한 예는 다음과 같다.

02 00 00 00 00 01 81 01 00 00 00 04 81 02 00 00 00 08 81 03 00 00 00 15 00 00 00 00 00 04 82 01 00 00 00 1E 85 01 00 00 00 01 81 04 00 00 00 04 82 02 00 00 00 03 85 01 00 00 00 1E 84 00 00 00

- 2단계 : BD -> IL

2단계는 BD 정보를 IL 언어로 번역을 하는 단계이다. BD의 데이터 구조는 출력(WR)의 개수를 의미하는 앞부분 4개 바이트와 명령어를 의미하는 6개씩의 바이트로 구성된다. 이러한 규칙을 이용해 BD를 IL 언어로 변환한

결과는 다음과 같다.

```
RD X1
ANDN X2
RDNS X3
ORS
ANDN Y1
WR R1
RD X4
ANDN Y2
AND R1
WR G0.00
```

- 3단계 : IL -> SF 리스트

3단계는 2단계에서 작성한 IL을 이용해 모든 출력신호의 SF을 구하는 단계이다. 여기서 각 출력신호의 SF라고 하는 것은 각 출력신호가 어떤 입력과 출력의 연산관계에 있는지 알려 주는 정보를 의미한다.

R1=((X1*(-X2))+(-X3))*(-Y1)　　(수학식5)
G0.00=X4*(-Y2)*R1　　(수학식6)

- 4단계 : SF 리스트 -> 특정 SF

3단계에서 구한 SF 리스트에는 내부 릴레이 R 신호, CNC에서 PLC로 전달되는 F 신호 그리고 PLC에서 CNC로 전달되는 G 코드 등이 포함되어 있다. 특정출력에 대한 SF를 구하기 위해서는 이러한 신호의 연관성을 풀어야 한다. G0.00에 대한 SF가 수학식7에 나타나 있다.

G0.00=(X4)*(-Y2)*(((X1)*(-X2))+(-X3))*(-Y1)　(수학식7)

'G0.00'를 고장 트리로 표현하면 그림 5와 같이 표현이 된다. 그림 5를 참고하면, 'G0.00'의 원인이 되는 신호의 관련성은 3개의 'AND' 논리와 하나의 'OR' 논리로 이루어진다는 것을 알수 있다.

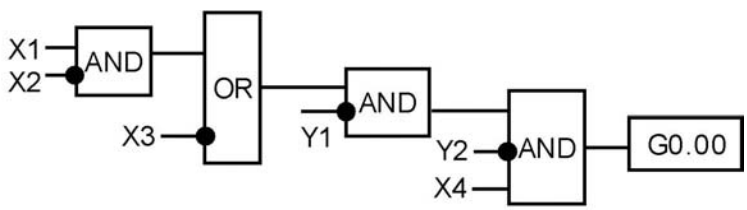

그림 5. G0.00의 고장 트리

- 5단계 : SF -> SSF

4단계에서 구한 SF를 분해하여 해당되는 SSF를 다음과 같이 구한다.

SSF1=(X1)*(-X2) (수학식8)

SSF2=SSF1 + (-X3) (수학식9)

SSF3=SSF2 * (-Y1) (수학식10)

SSF4=SSF3 * (-Y2) (수학식11)

SSF5=SSF4 * X4=G0.00 (수학식12)

이상에서 구한 수학식8 에서 수학식12의 SSF을 역으로 차례로 추적해 나가면, G0.00가 1이 된 원인을 추적해 나갈 수 있다. 이제, 구체적인 적용 사례를 통해 SFG를 이용한 고장진단 방법을 설명하기로 한다.

① SF(Switching Function) 모델의 형성

앞에서 설명된 SFG형성 방법을 적용하여, 본 출원인의 터보테크(주)에서 생산하는 I=I 형상조각기를 대상으로 SF를 구했다. 구해진 결과를 이용해 ESE에 대한 고장원인을 정리한 트리가 그림 6에 나타나 있다. 'G8.02'는 ESE가 발생하는 신호를 나타내며, 이러한 조건을 만족하는 논리로는 4개의 'OR' 조건 그리고 2개의 'AND' 조건을 가진다.

각 조건에는 다수의 신호조건이 연결된다. 그림에서 신호 끝에 도트로 표시한 것은 'NOT' 신호를 나타낸다. 그림에서 R은 내부 릴레이, G는 PLC에서 CNC로 입력되는 신호, F는 CNC에서 PLC로 입력되는 신호, X는 PLC 입력 신호 그리고 Y는 PLC 출력신호를 각각 의미한다. 이에 대한 각 신호 내용이 위의 표 2에 나타나 있다. ESE에

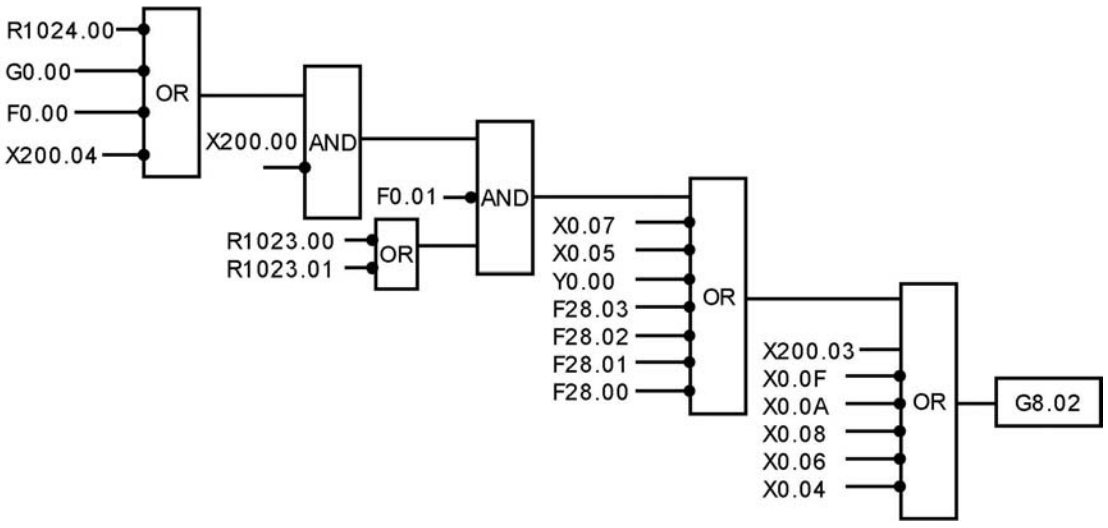

그림 6. ESE에 대한 고장 트리

표 2. ESE에 대한 데이터베이스

Address	Relay Comment
X0.07	S2 Alarm(B)
X0.05	S1 Alarm(B)
X0.0A	Z2 Over Travel Limit SW(B)
X0.08	Z1 Over Travel Limit SW(B)
X0.06	Y Over Travel Limit SW(B)
X0.04	X Over Travel Limit SW(B)
X200.00	Machine Ready
X200.03	E. Stop Push Button
X200.04	O. T. Release
X0.0F	E. Stop(B)
Y0.00	Servo Ready
F0.01	Servo Ready
F28.03	SVRDY4
F28.02	SVRDY3
F28.01	SVRDY2
F28.00	SVRDY1
F0.00	NC Ready
G0.00	PLC Run

대한 SF를 나타내면 수학식13과 같다.

SF(ESE)=(((((−R1024.00)+(−G0.00)+(−F0.00)+(−X200.04))*(−X200.00))*((−R1023.00)+(−R1023.01))*(−F0.01))+(−X0.07)+(−X0.05)+(−Y0.00)+(−F28.03)+(−F28.02)+(−F28.01)+(−F28.00))+(X200.03)+(−X0.0F)+(−X0.0A)+(−X0.08)+(−X0.06)+(−X0.04) (수학식13)

여기서 '+'는 'OR' 논리를 '*'는 'AND' 논리를 의미한다. 동일한 기계에 대해 CSD에 대한 고장원인을 정리한 트리가 그림 7에 나타나 있다.

'G9.00'은 사이클 스타트가 가능한 조건을 나타내며, 이러한 조건을 만족하는 논리로는 5개의 'OR' 조건 그리고 4개의 'AND' 조건을 가진다. 각 조건에는 다수의 신호조건이 연결된다. 이에 대한 각 신호 내용이 위의 표 3에 나타나 있다.

CSD에 대한 SF를 나타내면 수학식14와 같다.

SF(CSD)=(((((((−F17.00)+(−F17.01)+(−F17.02)+(−F17.03)+(−F18.00)+(−F18.01)+(−

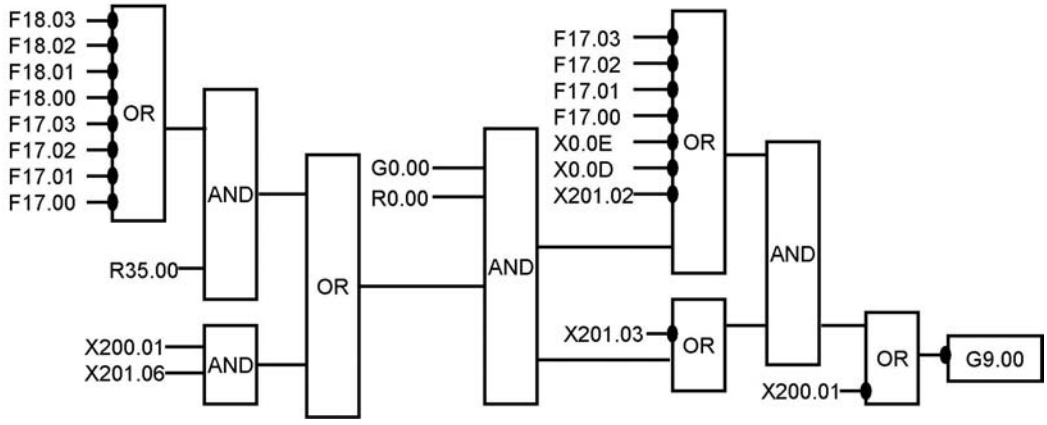

그림 7. CSD에 대한 고장 트리

표 3. CSD에 대한 데이터베이스

Address	Relay Comment
X0.0D	S1 Auto
X0.0E	S2 Auto
X200.01	Cycle Start
X200.02	Auto Mode
X200.03	MDI Mode
X200.06	JOG Mode
F17.00	X−Axis Ref. Finish
F17.01	Y−Axis Ref. Finish
F17.03	Z1−Axis Ref. Finish
F18.00	Z2−Axis Ref. Command
F18.01	X−Axis Ref. Position
F18.02	Y−Axis Ref. Position
F18.03	Z1−Axis Ref. Position
G0.00	PLC Run

F18.02)+(−F18.03))*(R35.00))+((X200.01)*(X201.06)))*(G0.00)*· (R0.00))+(−X0.0E)+(−X0.0D)+(−X201.02)+(−F17.03)+(−F17.02)+(−F17.01)+(−F17.00))*((((((−F17.00)+(−F17.01)+(−F17.02)+(−F17.03)+(−F18.00)+(−F18.01)+(−F18.02)+(−F18.03))*(R35.00))+((X200.01)*(X201.06)))* (G0.00)*(R0.00))+(−X201.03)))+(−X200.01)

(수학식14)

② SSF(Step Switching Function) 모델의 형성

이상의 G0.00의 예와 동일하게 SFG를 이용해 SSF를 구한다. 스텝 블록(Step Block)의 수는 'OR' 및 'AND'의 논리수와 같다. 즉, ESE는 6개의 스텝 블록을 가지며 CSD은 9개의 스텝 블록을 가진다. ESE에 대한 SSF를 나타내면 다음과 같다.

$SSF(ESE)_{16}=(-R1024.00)+(-G0.00)+(-F0.00)+(-X200.04)$ (수학식15)

$SSF(ESE)_{26}=SSF_{16}\cdot(-X200.00)$ (수학식16)

$SSF(ESE)_{36}=(-R1023.00)+(-R1023.01)$ (수학식17)

$SSF(ESE)_{46}=SSF_{26}\cdot SSF_{36}*(-F0.01)$ (수학식18)

$SSF(ESE)_{56}=SSF_{46}+(-X0.07)+(-X0.05)+(-Y0.00)+(-F28.03)+(-F28.02)+(-F28.01)+(-F28.00)$ (수학식19)

$SSF(ESE)_{66}=SSF_{56}+X200.03+(-X0.0F)+(-X0.0A)+(-X0.08)+(-X0.06)+(-X0.04)=G8.02$ (수학식20)

표 2를 참조하여 수학식15 즉, $SSF(ESE)_{16}$이 1이 되는 조건을 정리해 보면, R1024.00, G0.00, F0.00 그리고 X200.04 중 1개 이상의 신호가 NOT이 되는 조건이다. 이러한 조건이 만족되어 $SSF(ESE)_{16}=1$이 되면 ESE를 발생시키는 6개의 스텝 블록 중 하나의 스텝블록이 만족하는 조건이 되는 것이다.

동일하게 CSD에 대한 SSF를 나타내면 다음과 같다. 9개의 스텝 블록으로 구성된다.

$SSF(CSD)_{19}=(-F17.00)+(-F17.01)+(-F17.02)+(-F17.03)+(-F18.00)+(-F18.01)+(-F18.02)+(-F18.03)$ (수학식21)

$SSF(CSD)_{29}=SSF_{19}*R35.00$ (수학식22)

$SSF(CSD)_{39}=X200.01*X201.06$ (수학식23)

$SSF(CSD)_{49}=SSF29+SSF39$ (수학식24)

$SSF(CSD)_{59}=SSF49*G0.00*R0.00$ (수학식25)

$SSF(CSD)_{69}=SSF_{59}+(-X0.0E)+(-X0.0D)+(-X201.02)+(-F17.03)+(-F17.02)+(-F17.01)+(-F17.00)$ (수학식26)

$SSF(CSD)_{79}=SSF_{59}+(-X201.03)$ (수학식27)

$SSF(CSD)_{89}=SSF_{69}*SSF_{79}$ (수학식28)

$SSF(CSD)_{99}=SSF_{89}+(-X200.01)$ (수학식29)

CSD가 발생한 경우, 그 고장의 원인을 추적하기 위해 SSF 모델을 적용하여 고장이 발생한 스텝을 추적하게 된다. C(t)를 t 번째 스텝의 모든 조건이 조합된 상태라고 가정한다. 수학식24를 나타내는 스텝 4를 현재의 스텝이라고 가정하면, 스텝 4의 만족조건은 수학식23을 나타내는 스텝 3이 1이 되는 상태이다. 스텝 3으로부터 현재의 동

작 스텝4의 만족조건은 다음과 같이 표현 가능하다.

$$C(t-1) = X200.01 \cdot X201.06 = 1 \qquad (수학식30)$$

기능적으로 본다면 조그 모드에서 사이클 스타트(Cycle Start)가 온(On)되게 되면 CSD가 된다는 의미가 된다. 이것이 만족되면 스텝 4의 실행조건이 만족된다.

$$C(t) = SSF_{29} + SSF_{39} \qquad (수학식31)$$

이러한 원리에 의해 스텝 하나 하나씩 그 조건을 체크하면서 고장을 찾게 된다.

이러한 스텝 상관관계에 의해 고장이 발생한 스텝을 찾은 후에는 그 스텝에서 어떤 요소가 고장이 발생했는가를 찾게 된다. 이러한 고장요소를 찾는데는 미리 구성되어 있는 데이터 베이스를 활용하게 된다. 표 2와 3은 ESE와 CSD에 대한 고장요소를 DB로 정리한 것이다. 각 스텝 단위에서의 고장요소의 검출은 고장발생시 저장된 PLC의 각 접점상태의 확인을 통해 이루어진다.

이상에서는 본 기술의 바람직한 실시예를 설명하였지만, 본 기술은 상기의 실시 예들에만 한정되는 것이 아니라 본 기술의 요지를 벗어나지 않는 범위 내에서 다양한 개량, 변경, 대체 또는 부가하여 실시할 수 있는 것임은 당업자라면 용이하게 이해할 수 있을 것이다.

예를들면, 이상에서는 PLC로직이 래더 다이아그램으로 표현되는 경우를 예를들어 본 기술을 설명하였으나, 이

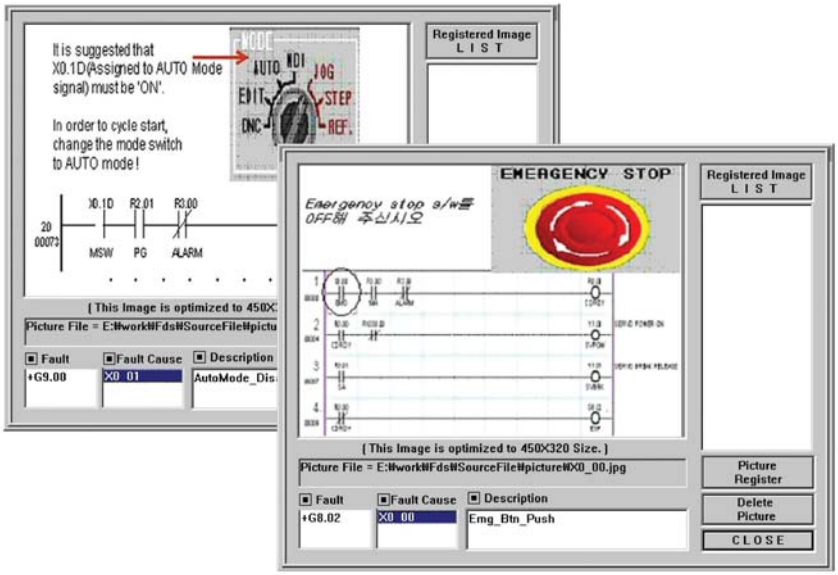

그림 8. CNC 컨트롤러에 실장된 FDS 운영화면

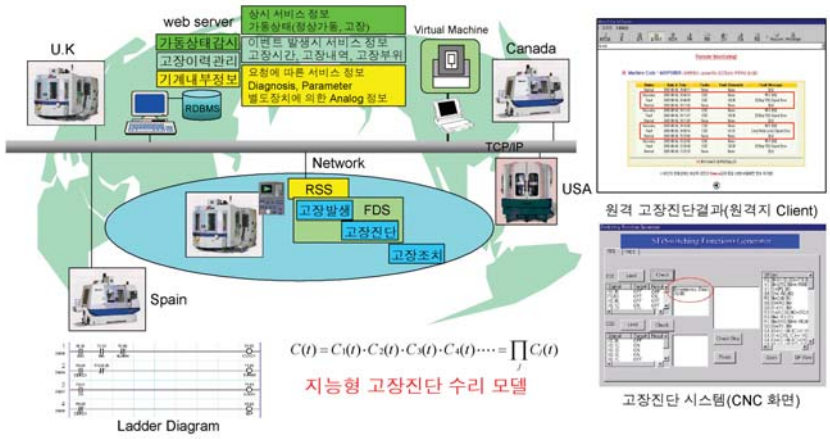

그림 9. 개방형 공작기계의 운용상의 고장에 대한 고장 원인 진단 및 원격 서비스

상에서 설명한 바와 같이 PLC로직이 IL(Instruction List)로 표현되는 경우에는 상기 단계3(IL->SF 리스트)에서 바로 시작하여 SF 및 SSF를 형성하는 것도 가능하다.

5. 적용분야 및 운용 예

개발된 고장원인 추적 · 진단기술은 공작기계 유지보수에 많은 비용과 인원이 필요한 분야에 최적의 솔루션을 제공한다. 즉, 현장에서 신속한 보수가 자동으로 가능하기에 내외부 보수요원의 추가 투입이 필요성이 많이 줄었다. 그림 8은 실제 CNC에 실장된 고장원인 진단시스템 운용 화면으로써 운용중 고장이 발생했을 때 자동으로 고장을 체크한다는 것을 의미한다. 여기서는 앞에서 설명된 SFG 알고리즘을 이용해서 ESE(Emergency Stop Error)가 발생했을 때 두가지 고장원인 추적모델인 SF와 SSF를 SFG를 이용해서 고장 원인을 찾은 결과를 보였다.

6. 기대효과 및 요약

공작기계의 유지보수성과 이용도 측면에서 볼 때, 본 기술개발에서 제안된 방법과 그 결과가 기계 비가동 시간을 줄이는데 있어서 공작기계 운용자와 유지보수관리자에게 매우 유용하게 된다. 지금까지 운용상의 고장이 발생했을 때 보수기사의 경험이나 PLC 래더 프로그램의 수작업 디버깅을 통해 고장의 원인을 찾을 때에 비하여, 빠르고 정확하게 현장에서의 자체적인 보수가 가능하게 된다.

본 기술개발에 따른 직접적인 효과는 다음과 같다.

- 공작기계의 운용상의 고장 원인을 찾는데 소요되는 기계 비가동 시간의 80%를 제거함으로 기계의 이용도를 높일 수 있다.
- 제안된 논리 모델을 이용한 고장진단시스템을 통해 고장원인을 정확하게 진단함으로 전문 보수요원의 필요

성이 많이 줄어든다.

- 기계의 상태 및 진단결과를 외부의 CNC 벤더에 실시간 보고함으로 원격지에서 종합적으로 보수정보 관리가 가능하였다.

이상과 같은 본 기술개발의 고장진단시스템은 PLC 래더 정보의 수식적인 등록과 운용상의 고장에 대한 출력신호의 선택이 가능하므로, 표준화된 PLC를 사용하는 다른 기계에서도 바로 적용할 수 있다. 또한, 본 기술은 개방형 CNC를 가지는 공작기계에서, 운용상 발생하는 고장을 지능적으로 진단하고 운용 및 진단상태를 인터넷을 기반으로 원격 서비스할 수 있는 시스템(그림 9참조)을 개발하는 것을 가능하게 한다. 이러한 원격감시 및 고장진단 시스템의 개발을 통하여 공작기계 제어기의 고기능화 및 IT 컨텐츠화가 가능하며, 수조원대의 공작기계 시장에서 년간 700억원대의 공작기계 개방형 제어기 수입대체 효과가 기대된다.

⚜ 참고 문헌

[1] Guasch, A., Quevedo, J. and Milne, R., 2000, "Fault diagnosis for gas turbines based on the control system Engineering", Application of Artificial Intelligence, Vol. 13, pp. 477-484.

[2] Hu, W., Starr, A.G., Zhou, Z. and Leung, A. Y. T., 2000, "A systematic approach to integrated fault diagnosis of flexible manufacturing systems", International Journal of Machine Tools & Manufacture, Vol. 40, pp. 1587-1602.

[3] Hu, W., Starr, A. G., Zhou, Z. and Leung, A., 2001, "An intelligent integrated system scheme for machine tool diagnostics", International Journal of Advanced Manufacturing Technology, Vol. 18, pp. 836-841.

[4] Kim, D. H., Kim, S. H., Koh, K. S., 2005, "A Scheme for an Internet-based Checking Method of Machine-Tools with Variant CNC Architecture", Journal of Mechanical Science and Technology, Vol. 19, No. 1, pp. 97-105.

[5] Rao, R. V. and Gandhi, O. P., 2002, "Failure cause analysis of machine tools using digraph and matrix methods", International Journal of Machine Tools & Manufacture, Vol. 42, pp. 521-528.

김 동 훈

· 한국기계연구원 지능기계연구센터 선임연구원
· 관심분야 : M2M기계복합지능화, Machine Vision
· E-mail : kdh680@kimm.re.kr