

공작기계의 고장원인을 효과적으로 생성하기 위한 Switching Function Generator 개발

김동훈(한국기계연구원), 김도연(충남대 대학원), 김선호(한국기계연구원), 이은애, 한기상(터보테크(주))

Development of Switching Function Generator for Fault Reasons of CNC Machine Tool

D. H. Kim(KIMM), D. Y. KIM(CNU), S. H. Kim(KIMM), E. A. Lee, G. S. Han(Turbotek Co., Ltd.)

ABSTRACT

The fault analysis of the CNC machine tool which is controlled sequentially by PLC is generally based on ladder diagram. When machine tool has faults, it takes a lot of operator's experiences and times to identify logical relationship because ladder diagram is a step structured language. Therefore the technologies of finding out fault reasons automatically is necessary. In this paper, the SFG(Switching Function Generator) system is developed to analysis fault reasons correctly. The SF(Switching Function) and SSF(Step Switching Function) generated from SFG based on ladder diagram are experimented to identify the performance of SFG.

1. 서론

CNC(Computerized Numerical Control)와 PLC(Programmable Logic Controller)에 의해 제어되는 공작기계에서는 순차제어를 수행하는 PLC 래더 다이어그램을 기반으로 고장의 원인을 분석하는 것이 일반적이다⁽¹⁾. 그 이유는 CNC나 서보에서 고장이 발생하더라도 그 정보 처리는 대부분 PLC에서 이루어지기 때문이다. Fig 1에 나타내었듯이 공작기계를 제어하는 CNC는 다시 그 기능을 나누어 보면 CNC와 PLC로 분리된다. CNC는 보간과 같은 운동제어(Motion Control)가 주된 기능이며, PLC는 CNC 공작기계에서 순차제어를 담당하는 중요한 역할을 한다. 두 가지의 제어장치간에는 유기적인 신호처리를 수행하면서

기계를 동작시키게 된다. 특히 PLC는 순차제어에 사용되는 릴레이, 타이머, 카운터 등과 같은 기능을 마이크로프로세서를 이용해 통합시킨 장치로서 산술·논리연산 등의 여러 연산처리가 가능하다. 하지만 이러한 우수한 기능과 폭 넓은 사용성에도 불구하고, PLC는 프로그램 작성방법이 순차적인 제어에 의존하기 때문에 다음과 같은 두 가지 불편한 점이 있다^(2,3).

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

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

이러한 이유로 고장 수리시간(20%) 보다 고장 원인을 찾는 데 걸리는 시간(80%)이 일반적으로 몇배 더 걸린다. 따라서 비 숙련자도 고장 원인이 되는 논리관계를 수작업에 의존하지 않고 프로그램적으로 오류 없이 빨리 찾는 기술을 개발하는 것이 필요하다^(4,5).

본 연구에서는 PLC에 의해 순차제어가 이루어지는 CNC 공작기계에서 비숙련자도 고장 원인을 빠르고 정확하게 찾을 수 있는 기능을 지원하는 SFG(Switching Function Generator)를 개발하고자 했다. 고장의 범위는 일반적으로 고장의 70%를 차지하는 운용상의 고장을 대상으로 했으며, 공작기계는 (주)터보테크에서 생산하는 개방형 CNC가 적용된 I-I를 대상으로 하였다.⁽⁶⁾

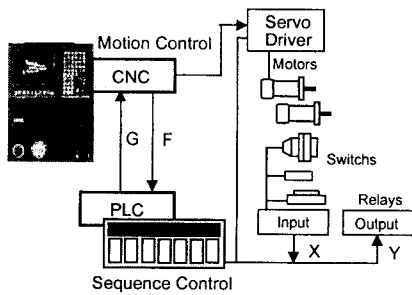


Fig. 1 Functional structure of CNC

2. 고장진단 모델

고장을 진단하기 위한 효과적인 방법으로 입출력 신호간에 논리적 관계를 가지는 두가지 진단 모델을 사용하였다⁽²⁾. 진단 모델을 만드는 주된 이유로는 각 공작기계 메이커마다 기계를 구성하는 방법에 고유한 특성을 가지기 때문에 진단 방법에 대한 일반성을 부여하기 위해서이다.

① SF(Switching Function) 모델

SF는 고장을 진단하기 위한 정적인 모델로서 이를 만족시키는 조건들은 많은 경우가 있을 수 있다. PLC가 하나의 동작을 완성시키기 위해 만족해야 하는 조건은 식 (1)과 같이 정리할 수 있다.

$$c(t) = c_1(t) \cdot c_2(t) \cdot c_3(t) \cdots = \prod_j c_j(t)$$

(1)

여기서 $c(t)$ 를 t 번째 스텝의 모든 조건의 조합상태라고 가정하고 t 번째를 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.

식 (1)에서 $c(t)$ 가 만족하려면 $c_1(t), c_2(t), c_3(t) \cdots$ 가 순차적으로 만족을 해야 한다. 여기서 각 스텝이 만족해야할 조건이 되는 $c_j(t)$ 를 스텝 스위칭 함수(Step Switching

Function)라 정의한다.

3. SFG 메커니즘

진단모델인 SF와 SSF의 생성을 SFG 시스템의 메커니즘을 통해 살펴보면, Fig 2와 같은 4단계의 과정을 거쳐 생성된다. 1단계는 바이너리 코드를 IL(Instruction List)로 변환하고 2단계는 1단계에서 만들어진 IL을 해석해서 각 출력신호들의 SF List를 만든다. 3단계는 SF List를 이용해서 특정 신호의 SF를 만들고, 4단계에서는 3단계에서 만들어진 SF를 이용해서 SSF(Step Switching Function)을 만든다.

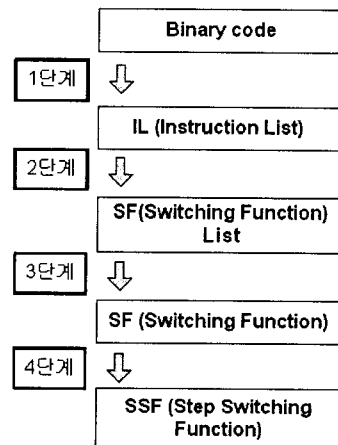


Fig 2 Flow chart for SFG

1단계 : Binary code → IL로의 변환

1단계는 래더 다이어그램으로부터 생성된 Binary code의 정보를 IL(Instruction List) 언어로 번역하는 단계이다. Binary code의 구조는 6개의 바이트가 하나의 명령어로 구성된다. Fig 3의 (a)와 (b)는 6개의 바이트가 하나의 명령어로 바뀌는 예를 보여준다. IL에서 AND는 '*', OR는 '+' 연산 논리를 의미하며, RD는 값을 Load/Read, RDS는 또 다른 연산을 위해 새로운 값을 Load/Read 하는 것을, ANDS는 앞서 논리적으로 연산된 묶음을끼리 '*', WR은 지금까지 연산된 값을 변수에 Write 하라는 것을 각각 의미이다⁽⁶⁾.

Binary code	IL (Instruction List)
00 01 83 1C 00 00	RD F 28.0
00 03 83 1C 00 01	AND F 28.1
00 03 83 1C 00 02	AND F 28.2
00 03 83 1C 00 03	AND F 28.3
00 05 85 FF 03 01	OR R 1023.1
00 07 83 00 00 01	RDS F 0.1
00 05 85 FF 03 00	OR R 1023.0
00 14 00 00 00 00	ANDS
00 03 82 00 00 00	AND Y 0.0
00 1E 85 00 00 01	WR R 0.1

Fig 3 Binary code → IL

2단계 : IL → SF List로의 변환

2단계에서는 1단계에서 제작된 IL을 이용하여 각 출력신호의 SF를 구하는 단계이다. 여기서 각 출력신호의 SF라고 하는 것은 각 출력신호가 어떤 입력과 출력의 연산관계에 있는지 알려주는 정보를 의미한다. 예를 들어, R0.1의 경우 IL에서 SF로 바뀌는 과정을 Fig 4에 나타내었다.

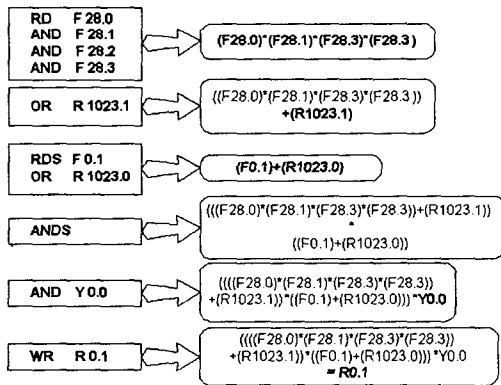


Fig 4 IL → SF List

3단계 : SF List → SF

2단계에서 구한 SF List의 SF에는 내부 릴레이를 의미하는 R 신호와 PLC에서 CNC로 전달되는 G 신호가 포함되어 있다. 특정출력신호에 대한 SF를 구하기 위해서는 이러한 신호들의 연관성을 풀어야 한다. R0.00 신호에 대한 SF를 구

하는 과정을 Fig 5에 나타내었다.

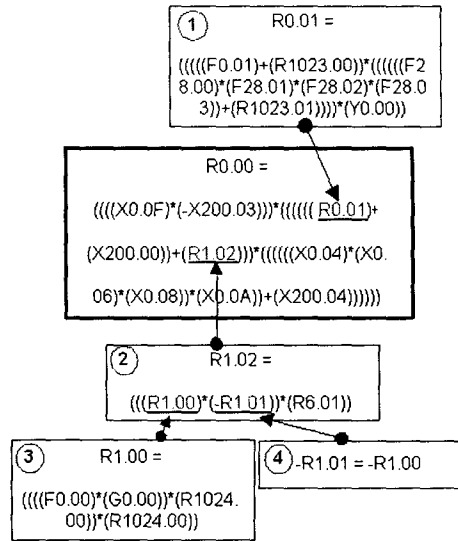


Fig 5 Procedure for SF

4단계 : SF → SSF

4단계에서는 3단계에서 구한 특정신호의 SF를 이용해서 SSF를 만든다. Fig 6과 같이 SF에서 SSF의 Step은 신호들의 연산이 '*'에서 '+'로 또는 '+'에서 '*'로 바뀔 때 결정된다.

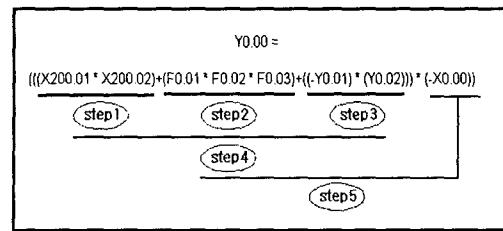


Fig 6 Step of SSF

4.SFG의 실행예제

다음은 이러한 알고리즘을 이용해서 (주)터보테크에서 생산하는 I=I 형상조각기의 ESE(Emergency Stop Error)에 대한 고장진단모델인 SF와 SSF를 SFG를 이용해서 구한 결과를 보여준다.

Fig 7은 ESE와 관련이 있는 레더 다이어그램이며 G8.02는 ESE가 발생하는 신호를 나타낸다.

이 레더 다이어그램의 정보를 가지고 있는 Binary code를 개발된 SFG에 입력하여 작업을 수행한 결과는 Fig 8과 같다. 왼쪽부터 Binary code, IL 및 SF List로의 변환된 결과이다.

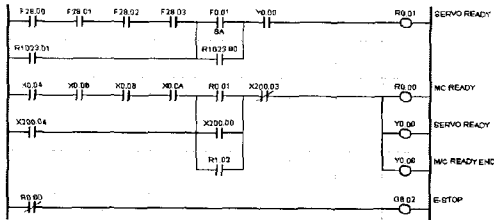


Fig 7 Ladder diagram for ESE

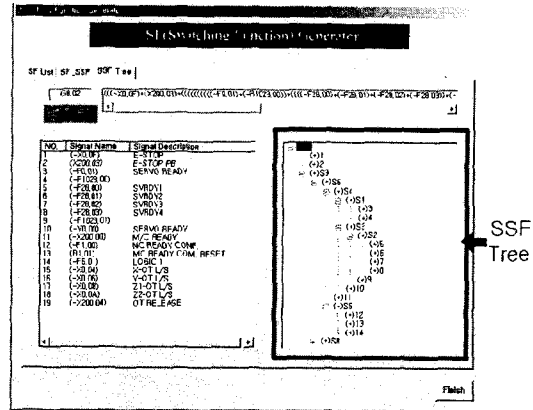


Fig 9 SSF Tree

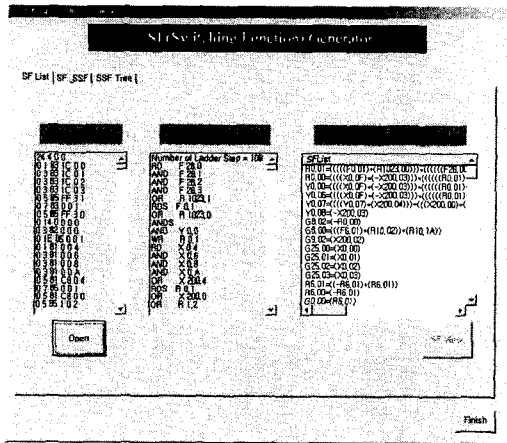


Fig 8 SFG main window

SFG에 의해 구해진 ESE에 대한 SF 모델은 다음과 같다.

$$(((\neg X0.0F) + (X200.03) + ((((((\neg F0.01) * (\neg R1023.00)) + (((\neg F28.00) + (\neg F28.01) + (\neg F28.02) + (\neg F28.03)) * (\neg R1023.01)))) + (\neg Y0.00))) * (\neg X200.00)) * ((\neg R1.00) + (R1.01) + (\neg R6.01)))) + (((\neg X0.04) + (\neg X0.06) + (\neg X0.08) + (\neg X0.0A)) * (\neg X200.04))))))$$

이를 SFG에서 단계별 표현기법으로 변환하면 Fig 9처럼 SSF Tree로 표현된다. SSF Tree내에서 's1', 's2'같이 's#'은 SSF의 "step #", 즉 step 번호를 의미하고, 's'없이 숫자만 있는 것은 Table 1에 정의된 입출력 신호를 의미한다. R은 내부 릴레이, G는 PLC에서 CNC로의 입력신호, F는 CNC에서 PLC로의 입력신호, X는 PLC 입력신호, Y는 PLC 출력신호를 각각 의미한다.

Table 1 Signal description (NC : Normal Close)

번호	신호	설명
1	-X0.0F	E-STOP
2	X200.03	E-STOP PUSH BOTTON
3	-F0.01	SERVO READY
4	-R1023.00	
5	-F28.00	SVRDY1
6	-F28.01	SVRDY2
7	-F28.02	SVRDY3
8	-F28.03	SVRDY4
9	-R1023.01	
10	-Y0.00	SERVO READY
11	-X200.00	M/C READY
12	-R1.00	NC READY CONF.
13	R1.01	MC READY COM. RESET
14	-R6.01	LOGIC1
15	-X0.04	X-OVER TRAVEL LIMIT SW(NC)
16	-X0.06	Y-OVER TRAVEL LIMIT SW(NC)
17	-X0.08	Z1-OVER TRAVEL LIMIT SW(NC)
18	-X0.0A	Z2-OVER TRAVEL LIMIT SW(NC)
19	-200.04	OT RELEASE

Fig 10은 SSF를 Tree형식으로 보는 과정을 단계별로 Step을 하나씩 펼치고 있다. 단계마다 있는 박스는 Step을 차례로 펼쳐 가는 과정을 나타낸 것이다. 이렇게 제작된 SSF Tree를 블록 다이어그램의 형태로 나타내면 Fig 11 같은 형태가 된다. 그림에서 신호선 끝의 도트는 'NOT' 신호를 나타낸다. 결과적으로 ESE의 SSF 모델은 AND 3개, OR 3개 총 6개의 스텝 블록을 가지는 것을 알 수 있다.

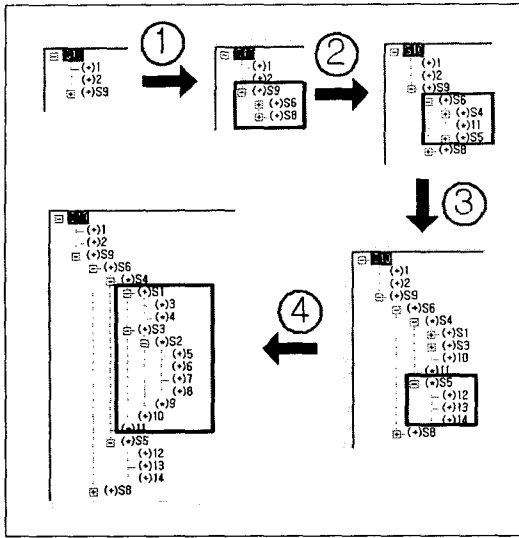


Fig 10 Expanding of SSF Tree step

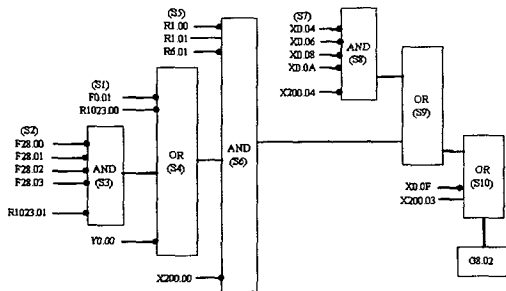


Fig 11 SSF Tree Block Diagram

5. 결 론

본 연구에서는 PLC에 의해 순차제어가 이루어지는 CNC 공작기계에서 래더 다이어그램으로부터 SF와 SSF라는 논리적 진단모델을 자동으로 생성시킬 수 있는 SFG를 개발하고, 실행예제를 통해 그 효용성을 제시했다. 개발된 SFG는 비숙련자도 공작기계의 고장의 원인을 빠르고 오류없이 찾을수 있는 방법을 제시해 준다. 또한, 개발된 시스템은 원격으로 고장을 진단하기 위한 시스템에도 효과적으로 응용이 가능한 기술이다.

참고문헌

1. 김선호, "공작기계의 원격감시, 원격고장진단

및 원격조작 기술," 한국정밀공학회지, Vol. 18, No. 10, pp.33-44, 2001.
 2. 김선호, 김동훈, 한기상, 김찬봉 "공작기계의 지능형 고장진단과 원격서비스 모델", 한국정밀공학회지, Vol. 19, No. 4, pp. 168-178, 2002.
 3. 김선호, 김동훈, 김도연외, "PLC로 제어되는 기계에서 Fault Tree를 효과적으로 생성하기 위한 LAT Tool 개발", 한국정밀공학회 추계학술대회 논문집 pp.442-445, 2001.
 4. Hu, W., Starr, A. G, and Leung A. Y. T., "Two diagnostic models for PLC controlled flexible manufacturing systems.," International J. of Machine tools & manufacture, Vol. 19, pp.1979-1991, 1999.
 5. 김형석, 장래혁, 권욱현, "고속 프로그램형 논리제어기 구현을 위한 래더 다이어그램 해석방법", 제어·자동화·시스템공학 논문집, Vol. 5, No. 1, pp. 33-38, 1999.
 6. TurboTek, Turbo HX-M 조작 매뉴얼, 2001.