원전용 다양성 제어기기의 Functional Coverage 분석1)

김규철*, 오승록**, 최종균***, 홍승일*, 배일호**
*단국대학교 응용컴퓨터공학과
**단국대학교 전자공학과
*** 한국원자력원구원
e-mail:kckim@dku.edu

Functional Coverage Analysis of a Diversity Controller for Nuclear Power Plants

Kyuchull Kim*, Seung-Rok Oh**,Jong-Gyun Choi***, Seung-Il Hong*, Il-Ho Bae**
*Dept of Applied Computer Engineering, Dankook University
**Dept of Electrical Engineering, Dankook University
*** Korea Atomic Energy Research Institute

요 약

원자력 발전소의 오작동이나 사고는 인명이나 재산상의 큰 피해를 초래하므로 엄격한 안전 기준을 적용하고 있다. 따라서 원자력 발전소의 안전성과 관련된 원전용 제어기는 높은 수준의 신뢰도와 안전도가 요구된다. 이를 위해 디지털 방식의 제어기에 PLC 방식의 제어기와 PLD 방식의 제어기를 사용하여 다양성을 얻고 있다. 본 논문에서는 PLD 방식의 원전용 트립제어기의 Functional Verification을 위하여 RTL 수준의 설계에 대한 Functional Coverage 분석을 사용하였다. 테스트벤치는 System Verilog에서 제공되는 클래스에 기반한 구조적 테스트벤치를 작성하여 사용하였다.

1. 서론

최근 원자력 발전용 제어기는 유지보수 상의 문제 때문에 기존의 아날로그 방식 대신 디지털 방식으로 대체되고 있다[1]. 디지털 방식으로 마이크로프로세서에 기반한 PLC 방식과 CPLD나 FPGA 같은 PLD를 사용하는 PLD 방식의 제어기를 동시에 사용하여 공통모드 실패(Common Mode Failure)에 대비하기 위한 다양성을 얻고 있다. PLD 방식의 제어기는 여러 나라에서 시도 되었다. 우크라이나[2]와 미국[3], 캐나다[4]에서 FPGA 기반의 I&C (Instrumentation and Control) 플랫폼이 개발되어 Shutdown 시스템과 같은 안전장치에 적용되었다. 본 논문에서는 PLD 방식으로 설계된 원자력 발전용 트립 제어기의 설계를 확인하기 System Verilog에서 제공되는 클래스에 기반한 구조적 테스트벤치를 구축하여 Functional Coverage 분석을 수행하였다. 국내에서는 System Verilog를 기반으로 하는 구조적 테스트벤치에 의한 설계 확인은 최초로 수행되는 것으로 파악되고 있다.

2. 원전용 트립 제어기

원자로 보호 시스템(RPS : Reactor Protector System)은 원자로의 동작과 관련된 온도, 압력 등 여러 프로세스 파라미터가 위험 수준에 근접하면 프리트립(pretrip) 신호를 발생시켜 청각 및 시각적인 경고를 하고, 위험 수준에 도달하면 트립(trip) 신호를 발생시켜 원자로의 동작을 멈추게 한다[5].

원전용 트립제어기는 그림 1에 보인 바와 같이 트립결정(trip decision) 블록과 셋포인트 계산(setpoint calculation) 블록으로 구성되어 있다. 셋포인트 계산 블록은 셋포인트 알고리듬에 따라 프리트립 수준과 트립 수준을 계산한다. 트립결정 블록은 프로세스 파라미터가 프리트립 수준을 넘게 되면 프리트립 신호를 내보내고, 트립 수준을 넘게 되면 트립 신호를 발생시킨다. 원자로 제어 시스템은 프리트립 신호가 발생하면 시각, 청각적인 경고를 하게 되고, 트립 신호가 발생되면 원자로를 셧다운(shutdown) 시킨다.

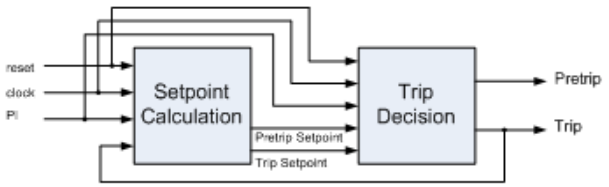

그림 1. 원전용 트립제어기의 구조

셋포인트 계산 알고리듬에는 Fixed setpoint, Rate-limited variable setpoint, Directional variable

1) 본 연구는 지식경제부 원자력 연구개발사업(원전기술혁신)의 지원으로 연구되었음.

setpoint의 세 가지 방식이 있다. 그리고 각 계산 알고리듬에는 Rising trip과 Falling trip의 두 가지 configuration이 있다. Rising trip configuration은 정상동작 범위가 트립수준보다 낮은 경우에 사용되고 Falling trip configuration은 정상동작 수준이 트립 수준보다 높은 경우에 사용된다. 그림 2는 Fixed setpoint trip 방식의 Rising configuration을 보이고 있다.

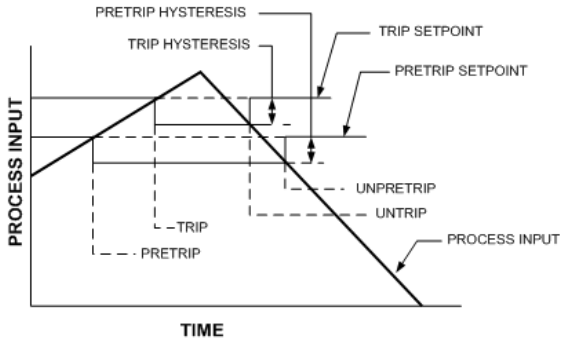

그림 2. Fixed Setpoint Trip Configuration

그림 2에서 untrip 수준은 트립이 일어난 후에 프로세스 파라미터가 정상으로 회복되어 안전한 상태로 들어가는 수준으로서 trip hysteresis 만큼 trip 수준과 차이가 있다. 그림 3은 UNTRIP, PRETRIP, TRIP의 세 상태를 가지고 있는 Rising configuration의 Fixed Setpoint 방식 트립결정 블록의 상태천이도를 보이고 있다. 각 상태마다 출력신호 TRIP과 PRETRIP이 결정된다. 상태 간의 천이 조건은 천이를 나타내는 화살표 옆에 표시되어 있다.

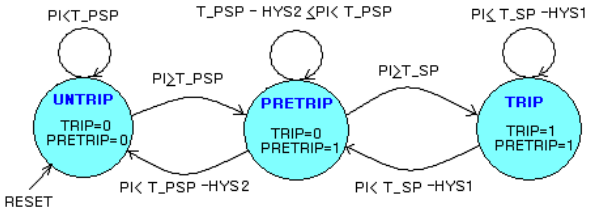

그림 3. Fixed Setpoint 방식 트립결정 블록의 상태천이도

3. System Verilog를 사용한 구조적 테스트벤치

셋포인트 계산 블록과 트립결정 블록은 Verilog로 RTL 수준에서 설계되었다. 그림 4는 RTL 수준으로 작성된 설계를 확인(verification)하기 위한 테스트벤치를 보이고 있다. 그림에서 DUT는 셋포인트 계산블록과 트립결정블록으로 구성되어 있다. DUT를 제외한 부분이 테스트벤치인데 테스트벤치는 클래스 개념을 지원하는 System Verilog로 작성되었다.

그림에서 테스트벤치는 드라이버(driver), 모니터(monitor), 스티뮬러스 생성기(stimulus generator), 스코어보드(scoreboard), 카버리지 컬렉터(coverage collector) 및 테스트 컨트롤러(test controller)로 구성되어 있다. 드라이버는 생성기가 만들어낸 스티뮬러스를 핀레벨 신호로 변환시켜 DUT에 인가하는 역할을 한다. 모니터는 DUT에서 나온 신호를 트랜잭션으로 변환시켜 스코어보드와 카버리지 컬렉터로 보낸다. 스코어보드는 생성기가 만들어낸 스티뮬러스에 대하여 DUT가 생성할 반응을 계산하여 모니터에서 나온 신호와 비교한다. 카버리지 컬렉터는 모니터에서 보내 온 트랜잭션(transaction)의 수를 계산한다. 테스트 컨트롤러는 스코어보드에서 받은 정보를 사용하여 시뮬레이션을 계속하거나 중지시킨다. System Verilog는 임의의 테스트 방식을 사용하여 임의의 테스트 케이스(test case)를 자동으로 생성하는 것을 지원한다[6].

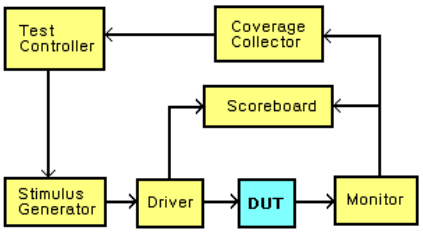

그림 4. 테스트벤치의 구조

4. Functional Coverage 분석

Functional verification은 RTL로 기술된 설계가 설계사양에 맞게 동작하는지를 확인하는 작업이다. 이를 위해 System Verilog를 지원하는 Mentor Graphics사의 QuestaSim이 시뮬레이터로 사용되었다. 그리고 구조적 테스트벤치를 작성하기 위해서는 QuestaSim 의 테스트벤치용 클래스 패키지인 AVM (Advaced Verification Methodology)을 사용하였다.

Functional verification을 하는 방법에는 사용되는 방법 중 하나가 Functional 카버리지를 구하는 것이다. 카버리지는 유한 공간에서 다양한 종류의 이벤트 발생 횟수를 측정하는 것이다. Functional 카버리지는 System Verilog에서 지원하는 기능 중의 하나이다. System Verilog 코드에 카버포인트(coverpoint)로 구성된 카버그룹을 작성하여 넣으면 카버그룹 내의 카버포인트가 만족되는지의 여부를 퍼센티지로 표현한다.

본 실험에서는 Functional 카버리지 분석에 세가지 카버그룹을 사용하였다. 첫 번째는 트립결정 블록과 셋포인트 계산 블록의 상태에 대한 것이고 두 번째는 각 상태에서의 천이에 관한 것이며 세 번째는 셋포인트에 대한 것이다. 다음 코드는 트립결정 블록의 상태에 대한 카버그룹의 한 예를 보이고 있다.

```
covergroup state_cvg @(posedge clk_1ms);
   option.per_instance = 1;
   state : coverpoint CURRENT_STATE {
      bins st[] = {NORMAL_STATUS, PRETRIP_STATUS,
                   TRIP_STATUS};
      bins illegal = default;
   }
   transition : coverpoint CURRENT_STATE {
      bins tr[] =  ( NORMAL_STATUS => NORMAL_STATUS),
                   (NORMAL_STATUS => PRETRIP_STATUS),
                   (PRETRIP_STATUS => PRETRIP_STATUS),
```

```
                    (PRETRIP_STATUS => TRIP_STATUS),
                    (PRETRIP_STATUS => NORMAL_STATUS),
                    (TRIP_STATUS => TRIP_STATUS),
                    (TRIP_STATUS => PRETRIP_STATUS) ;
    bins illegal = default;
  }
endgroup
```

카버포인트에 대해 bins를 정의하면 각 bin에 해당하는 값의 발생 횟수를 측정할 수 있다. bins 외에 ignore_bins와 illegal_bins를 정의할 수 있는데, ignore_bins를 정의하면 해당 값이 발생해도 카버리지에는 포함되지 않고 illegal_bins에 해당하는 값이 발생하면 오류 메시지를 발생시킬 수 있다.

그림 5는 트립제어기에 적용된 세 가지의 카버그룹 중 트립결정블록에 대한 상태 및 상태천이에 대한 카버리지를 퍼센티지로 보이고 있다. 모든 bin에서 100% 카버리지를 얻었다. 이는 테스트벤치에서 트립제어기에 인가된 스티뮬러스가 트립제어기의 모든 기능을 잘 시험하고 있음을 확인(verify)했음을 의미한다.

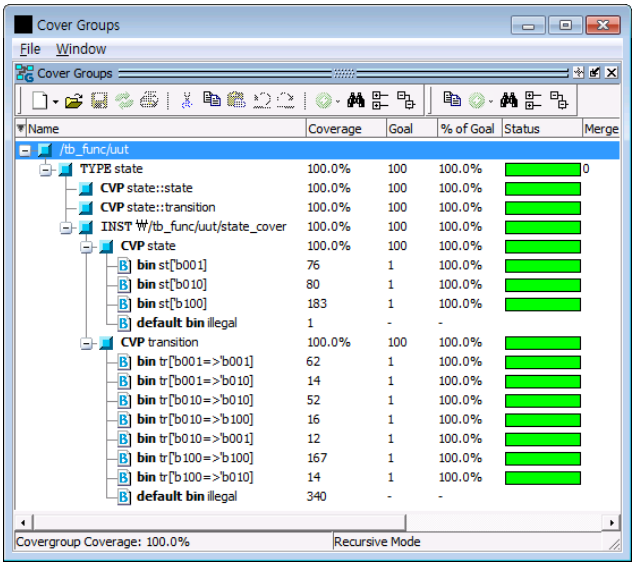

그림 5. Functional Coverage 분석

그림에서 보이는 그래프는 카버리지를 시각적으로 표현한 것으로 정해진 기준을 달성하면 초록색으로 표시되어 통과를 나타내고 기준에 미달하면 붉은 색으로 표시되어 불합격을 나타낸다. 카버리지를 만족시키지 못한 카버포인트를 테스트케이스의 'hole'이라고 하는데, Functional 카버리지 분석 결과 'hole'이 발견되면 이 'hole'을 없앨 수 있는 새로운 테스트케이스를 생성하여 카버리지를 정해진 수준으로 올릴 수 있다. Functional 카버리지가 달성됨은 설계 사양이 확인계획(verification plan)에 따라 충분히 확인되었음을 의미한다.

5. 결론

본 연구에서는 Verilog로 설계된 원전용 트립제어기에 대하여 Functional Verification을 수행하기 위해 System Verilog로 구조적 테스트벤치를 작성하여 Functional Coverage 분석을 수행하였다. Functional Coverage 분석 결과 RTL 수준으로 설계된 트립제어기가 설계 사양에 맞게 잘 동작됨을 확인하였다. 본 연구에서의 Functional 카버리지의 목표는 100%로 설정하였다. 이 방식은 RTL 설계의 Functional Verification에 아주 유용하고 편리하게 사용될 수 있을 것이다.

참고문헌

[1] Patrick Salaün, Frederic Daumas, Thuy Nguyen, and Claude Esmenjaud, "FPGA/ASIC: A promising technology for future of I&C Systems in power industry," 6th American Nuclear Society International Topical Meeting on NPIC&HMIT, April, Knoxville, Tennessee, USA, 2009.

[2] Vyacheslav Kharchenko, "Experience of RPC <<Radiy>> is designing, manufacturing and implementation of FPGA-based NPP I&C systems," 1st Workshopon The Applications of Field-Programmable Gate Arrays in Nuclear Power Plants, October, Chatou, France, 2008.

[3] Bernard F. Dittman, "Regulatory Experience with a FPGA-based Digital I&C Review," 2nd Workshopon The Applications of Field-Programmable Gate Arrays in Nuclear Power Plants, September, Kirovagrad, Ukraine, 2009.

[4] Jingke She and Jin Jiang, "Application of FPGA to Shutdown system No. 1 in CANDO", 6th American Nuclear Society International Topical Meeting on NPIC&HMIT, April, Knoxville, Tennessee, USA, 2009.

[5] Seop Hur, Don Hoon Kim, "A New Automatic Periodic Test Method for the Digital Reactor Protection System," 6th American Nuclear Society International Topical Meeting on NPIC&HMIT, April, Knoxville, Tennessee, USA, 2009.

[6] Chris Spear, "SystemVerilog for Verification: A Guide to Learning the Testbench Language Features," 2nd Edition, Springer, 2008.