

SystemC 를 이용한 PCI Express 종단장치 코어의 검증 모델 설계

김선욱, 김영우, 박경
한국전자통신연구원 컴퓨터시스템연구부

Design of PCI Express Endpoint Core Verification Model Using SystemC

Sunwook Kim, Youngwoo Kim, Kyoung Park
Computer System Research Department
Electronics and Telecommunications Research Institute
E-mail : {swkim99, bartmann, kyoung }@etri.re.kr

Abstract

In this paper, a design and experiment of PCI Express Core verification Model is described. The model targeting Endpoint core verification is designed by using newly-emerging SystemC which is a system design language based on a new C++ class library and simulation engine. In the verification model, we developed a SystemC Host System model which act as a Root Complex and Device Driver dedicated to the PCI Express Endpoint RTL Core. The test of Host System Model is guided by scenarios which implements and acts point of Device Driver and Root Complex and shows the result of simulation. Also, We present the full structure of verification model and Host model.

I. 서론

최근 반도체 기술의 발달로 마이크로프로세서 및 메모리, 로직, 아날로그, DSP 등과 같은 다양한 기능을 하나의 칩에 구현하는 개념의 SoC(System-On-Chip) 기술을 이용한 시스템 설계가 보편화되고 있다. 시스템의 복잡도가 높아지고 대형화되면서 기존의 시스템 설계 방법은 SoC 설계 요구를 만족시키지 못하고 있다. 또한 하드웨어 개발이 끝나는 시점에서 하드웨어 장치 드라이버의 개발을 시작하고, 그 후에 시스템 통합 및 시험,

검증의 절차를 거치는 기존의 설계 방법으로는 시스템 개발 기간을 단축시킬 수 없으며 이로 인해 제품에 대한 경쟁력 확보가 어렵다. 이에 면적 단위의 최적화를 요구하던 과거의 설계 방향에서 빠른 설계 기간을 요구하는 방향으로 변함에 따라 재사용 가능한 IP 기술과 하드웨어와 소프트웨어 공동 설계 및 검증 환경, 통합화 기술이 중심이 되고 있다. 특히 시스템이 복잡해지고 대형화되면서 정확한 시스템 수준 검증을 위해 소요되는 시간이 증가하여 설계 기간을 단축시키기 위한 새로운 시스템 설계 검증 환경에 대한 요구가 높아지고 있으며 하드웨어와 소프트웨어 통합 개발이 가능한 시스템 레벨의 설계 방법으로 C/C++ 기반의 하드웨어 설계 툴들과 방법론들이 등장하고 있다. SystemC 는 이러한 필요성에 따라 시스템 설계용으로 만들어진 표준 C++ 클래스 라이브러리로서 C++의 빠른 검증 능력, 확장성, 다양한 동작 기술 함수 제공 능력 등의 장점을 이용할 수 있으며, HDL(Hardware Description Language)의 Multi-Valued Logic 과 같은 하드웨어 동작 기술에 적합한 클래스들을 새로이 추가 구성하여 최상위 추상화 수준인 시스템 모델링으로부터 낮은 수준의 RTL(Register Transfer Level) 모델링이 단일 언어에서 가능하기에 새로운 시스템 설계 환경으로 그에 대한 관심이 높다[1].

본 논문에서는 하드웨어와 소프트웨어 통합 개발이 가능한 SystemC 를 사용하여 최근 소개된 새로운 시스

템 연결망 기술로서 고속 직렬 점대점(point-to-point) 패킷 전송 방식을 사용하는 PCI Express 종단장치 코어의 검증 모델을 설계하고 이를 활용한 동작 검증에 대하여 논의 한다.

서론에 이어 2 장에서는 PCI Express 연결망 기술에 대하여 간략히 설명하고 이어 SystemC 를 이용한 검증 모델의 설계를 다루도록 한다. 마지막으로 설계된 모델의 검증과 결과를 설명한다.

II. PCI Express 연결망 기술의 개요

PCI-Express 기술은 기존 병렬버스방식과 달리 직렬 기술을 채택한 입출력 연결망 기술이다. PCI-Express 기술은 종래 PCI 와 소프트웨어적인 호환성을 유지하면서 계층 버스 구조의 병목 현상을 해소하기 위하여, 스위치드 패브릭(switched fabric) 구조를 입출력 연결방식으로 채용한 것이 그 특징으로 현재 PCI-Express 1.0a 규격이 발표되어 있다.

PCI Express 기술은 점대점 패킷 전송 방식의 프로토콜로서 Transaction, Data Link, Physical 계층 구조를 가지며 데이터의 송수신이 진행 됨에 따라 각 계층에서 사용될 정보를 처리하게 된다.그림 1 에서 보듯이 각 기능의 레이어화를 통하여 데이터 레이트의 변화와 무관하게 다른 레이어에서의 영향을 최소화할 수 있도록 하여 향후 확장이 용이하도록 하였다 [2].

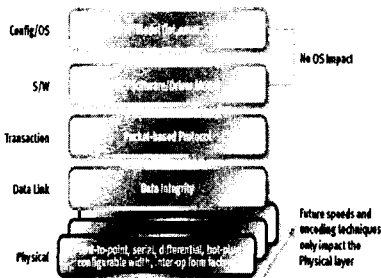


그림 1. PCI Express 의 계층 구조

Transaction 계층은 데이터의 전송을 주관하는 계층으로서, 데이터의 분해와 조립 (Segmentation And Reassemble) 기능을 담당하며 중간 계층인 Data Link Layer 는 연결되어있는 상대 Link 와의 올바른 TLP 전송을 위한 헤더 생성, 버퍼의 흐름제어 및 Link 의 관리를 위한 Data Link Layer Packet(DLLP)를 담당한다. 마지막으로 물리적인 link 로의 전송과 데이터의 직/병렬 변

환은 SerDes (Serializer and Deserializer)기능을 포함하고 있는 Physical 계층이 담당한다[3].

PCI Express interconnect 는 기존 방식과 달리 Root Complex (기존 PCI 의 Host Bridge 에 상응), Switch, Endpoint device 로 구성이 된다. 이들 device 는 Link 라 불리는 물리적인 연결을 통하여 이루어지며, Link 의 상위 연결 포트를 downstream 포트, 하위 연결 포트를 upstream 포트로 규정하고 있다. 실제 동작을 수행하는 종단 장치 (endpoint device)는 상호 Link 의 설정이 완료된 후, 설정 정보를 상위 장치인 Root Complex 로부터 받아 오으로써 PCI Express 장치로서 동작하게 된다 [4].

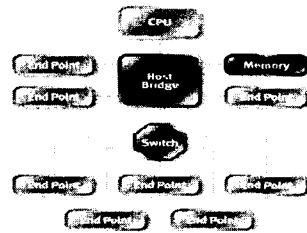


그림 2. PCI Express Topology

III. PCI Express 종단 장치 코어 검증 모델의 SystemC 모델링

3.1 SystemC 를 이용한 시스템 모델링

SystemC 는 Synopsys, Coware, Frontier Design 을 중심으로 구성된 OSCI(Open SystemC Initiative)에서 개발한 시스템 모델링 환경으로 C++ 클래스 라이브러리와 시뮬레이션 커널로 구성된다. 이는 하드웨어 개념인 병렬이나 반응 동작(reaction behavior), 하드웨어 타이밍, 비트/벡터의 데이터 형 등을 각각 클래스 형태로 제공 하처하여 기존의 C/C++ 컴파일러를 그대로 이용한 하드웨어 모델링이 가능케 한다.

SystemC 의 기본 블록인 모듈(module)은 다른 모듈이나 모듈안에 포함되는 프로세스를 가지는 계층적인 객체이며 프로세스(Process)는 C/C++의 함수와 같은 역할을 담당하여 동작 기능을 표현하며 모듈에 포함된다. 또한 모듈은 각 모듈간의 데이터 통신과 제어를 위해 연결 통로 역할을 하는 입출력 포트(Port)를 가지게 되며 모듈내의 프로세스간의 동기화를 맞추기 위해 클럭(Clock)을 사용하게 된다[5].

3.2 PCI Express 중단 장치 코어 검증 모델

본 논문에서는 Verilog HDL 을 사용하여 설계된 PCI Express 중단 장치 코어의 검증을 위하여 상위 장치인 Root Complex 의 역할 및 일부 PCI Express 장치 드라이버의 기능을 수행할 수 있는 Host 모델을 SystemC 를 이용하여 모델링하였으며 그림 3 은 본 논문에서 제시하는 PCI Express 중단 장치 검증 모델을 나타낸 그림이다.

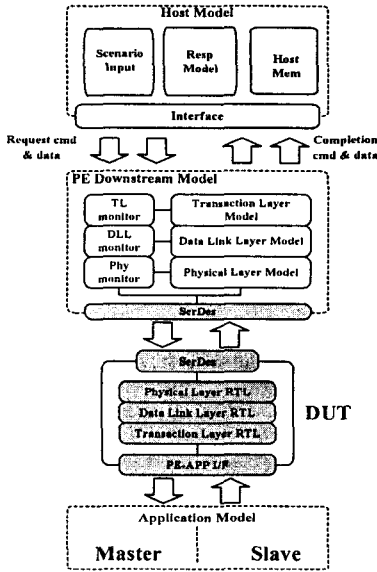


그림 3. Verification Mode 의 전체 개념도

Host 모델은 장치 드라이버가 구동하고 있는 호스트 플랫폼의 동작을 모사하기 위하여 데이터의 송수신을 위한 버퍼 메모리, downstream 포트 모델을 통한 패킷의 전송과 수신을 위한 인터페이스, 데이터의 송신을 위한 시나리오 모듈, 수신된 패킷에 대하여 응답을 생성하는 응답기 모듈로 구현하였다.

실질적인 PCI Express 패킷을 생성/수신하는 부분인 Downstream 모델은 PCI Express 프로토콜을 처리하기 위한 각 계층에 대한 행위 모델과 각 계층 모델에서의 데이터 흐름과 프로토콜의 오류여부를 확인하기 위한 모니터 모듈로 구성하였다.

Application 모델은 중단장치에 연결되는 장치의 기능 모델로서 PCI Express 프로토콜에 따른 응답을 생성함으로써 검증하고자 하는 중단장치(DUT)의 동작을 확인할 수 있도록 구성하였다.

3.3 Host 모델의 SystemC 모델링

SystemC 를 이용해 설계한 Host 모델은 PCI Express 중단 장치 코어 검증을 위한 PCI Express 연결망의 설정과 각종 검증 Transaction 을 생성/검증하는 주체이며 각 Transaction 별 구현 함수를 통한 시나리오 집합의 작성 및 구성된 시나리오 집합을 Downstream 모델에게 요청한다. 반대로 중단 장치 코어로부터 Host Memory 또는 IO 영역에 대한 read, write request 를 받아 처리하며 처리 결과를 Host 메모리 또는 IO 영역에 저장하거나 Downstream 모델에게 전송한다.

호스트 모델을 구성하는 모듈은 크게 네가지로 나누어 진다. 호스트 메모리를 모델링한 호스트 메모리 모듈과 PCI Express Transaction 별 구현 함수를 통한 검증 시나리오 집합을 구동시키는 시나리오 입력 모듈 그리고 중단 장치로부터의 transaction 요구를 처리하고 처리 결과를 downstream 모델에 전송하는 응답기 모듈로 나누어 진다. 마지막으로 호스트 모델의 구성 모듈에 대한 instance 를 통해 통합 모델링을 구현하는 호스트 모델 탑 모듈이 있다. 또한 각 모듈에서 수행하는 프로세스는 SystemC 에서 제공하는 세가지 유형의 프로세스 중에서 Clocked Thread 프로세스를 사용하여 클럭의 상승 에지에 동기되어진다. 표 1 은 각 모듈에 구현된 프로세스의 특징 및 동작을 정리한 것이며 그림 4 는 호스트 모델의 모듈 및 프로세스 구성을 나타낸다.

표 1. 선언된 Module 및 Process 의 특징

모듈명	프로세스명
SC_MODULE (Scen_Input)	SC_CTHREAD(do_scenario, clk_pos()) : CThread 프로세스로 생성되어지며 내부 클럭의 positive edge 에 동기화된다. : 검증 시나리오 집합을 생성 및 수행
SC_MODULE (resp_model)	SC_CTHREAD(init_resp_model, clk_pos()) : CThread 프로세스로 생성되어지며 내부 클럭의 positive edge 에 동기화된다. : 응답기 모듈의 초기화 수행 SC_CTHREAD(run_write_request, clk_pos()) : CThread 프로세스로 생성되어지며 내부 클럭의 positive edge 에 동기화된다. : 중단 장치에서 생성된 write 요청을 처리
	SC_CTHREAD(run_read_request, clk_pos()) : CThread 프로세스로 생성되어지며 내부 클럭의 positive edge 에 동기화된다. : 중단 장치에서 생성된 read 요청을 처리
SC_MODULE (Memory)	SC_CTHRED(do_mem, clk_pos()) : CThread 프로세스로 생성되어지며 내부 클럭의 positive edge 에 동기화된다. : 각 모듈에서 요청한 memory read /write 요청을 처리

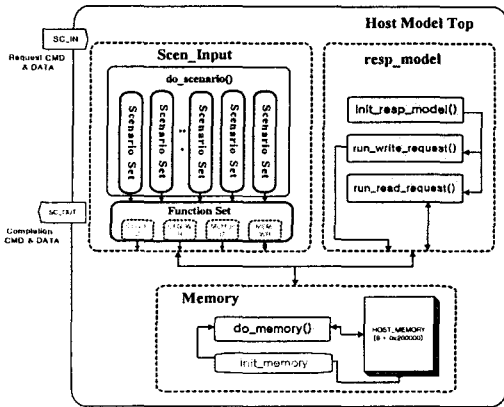


그림 4. Host model 의 구조

IV. 검증 모델 실험 결과

구현된 Host 모델에 대한 검증은 Testbench 의 역할을 수행하는 테스트 모듈을 추가하여 모델에 대한 검증을 진행하였다. 추가된 테스트 모듈은 configuration read/write 와 메모리 read/write 에 대한 시나리오 집합을 동작시키는 모듈(test_host_model)과 응답기 모듈에 대해 memory read/write 동작을 지시하는 제어 신호를 내보내는 모듈(test_resp)이다.

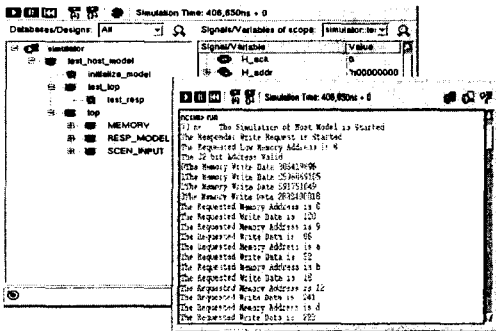


그림 5. 모듈의 계층 및 시물레이션 결과

test_host_model 에서는 호스트 모델 모듈에 클럭 및 초기화 신호, 시나리오 집합 동작 신호를 내보낸다. test_resp 모듈에서는 응답기 모듈을 동작시키기위한 제어 신호 및 동작에 필요한 address 및 data, 길이 등의 정보를 내보낸다. 그림 5 는 구현된 모듈의 계층 및 시물레이션에 의해 출력되는 결과값을 나타내며 그림 6 은 응답기 모듈의 memory write 수행 결과를 VCD 파형으로 나타낸 것이다.

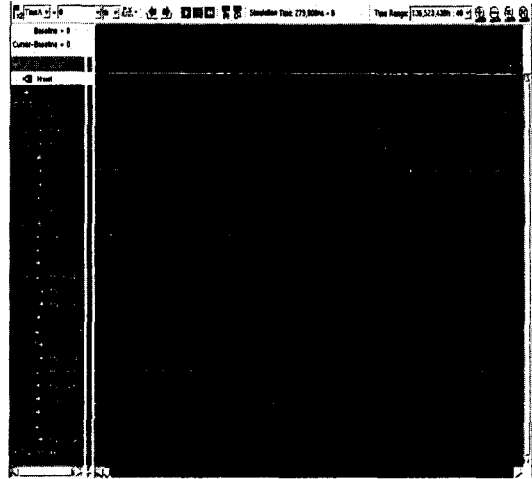


그림 6. 응답기 모듈의 수행 결과

V. 결론

본 논문에서는 Verilog HDL 을 사용하여 설계된 PCI Express 종단 장치 코어의 검증을 위하여 상위 장치인 Root Complex 의 역할 및 일부 PCI Express 장치 드라이버의 기능을 수행할 수 있는 Host 모델을 SystemC 를 이용하여 모델링하였다. 장치 드라이버의 일부 동작을 수행하는 Host 모델의 검증 결과 S/W 이지만 H/W 와 밀접하게 관련되어 있는 장치 드라이버의 기능을 SystemC 를 활용하여 모델링하였을 경우 H/W 및 S/W 의 동작 모두에 대해 PCI Express 종단장치 코어의 동작 검증을 수행할 수 있었다.

현재 검증 모델중 나머지 모델도 SystemC 를 활용한 모델링을 진행중이며 향후 PCI Express 종단 장치 코어에 대한 하드웨어와 소프트웨어 통합 검증을 실시할 계획이다.

참고문헌

- [1] Cadence Manual, "NC-SystemC Simulator Ver 5.0," 2003.
- [2] 김영우 외, "입출력 연결 규격의 기술 동향," 중간기술동향, 통권 1101 호, 2003. 6, pp.12~25.
- [3] Ajay Bhatt, "PCI Express Architectural Overview," PCI SIG Developer's Conference, 2002
- [4] PCI Express Base Specification 1.0a, PCI-SIG
- [5] 기안도, "C 언어에 기초한 시스템 모델링 언어 SystemC ver 0.1," <http://www.dynalith.com>, Sep. 2003.