

논문 2007-44SD-4-4

다중 채널과 동시 라우팅 기능을 갖는 고성능 SoC 온 칩 버스 구조

(High Performance SoC On-chip-bus Architecture with Multiple
Channels and Simultaneous Routing)

이 상 현*, 이 찬 호**

(Sanghun Lee and Chanho Lee)

요 약

현재까지 다수의 버스 프로토콜과 구조가 발표되었지만, 대부분 공유 버스 구조를 가져 시스템 성능 저하의 원인이 되었다. 기존의 공유버스가 갖는 문제점들을 해결하기 위해 고성능의 버스 프로토콜인 SNP (SoC Network Protocol)와 버스 구조인 SNA (SoC Network Architecture)가 제안되었는데, 이를 수정/개선한 버스 구조를 제안하고자 한다. 개선된 SNA는 다중 마스터의 다중 버스 요청에 대해 다중 라우팅을 지원함으로써 성능을 향상시켰으며, 내부 라우팅 로직의 최적화로 면적을 감소시켰다. 또한 성능감소 없이 AMBA AHB 프로토콜과 완벽히 호환 가능한 XSNP(Extended SNP)를 인터페이스 프로토콜로 사용한다. 현재 라우팅 로직을 최적화하여 개선된 SNA의 하드웨어 복잡도가 크게 증가하지 않았고, 기존 SNP를 사용하는 IP는 호환성 문제나 성능 감소 없이 개선된 SNA를 통해 통신할 수 있다. 더불어, SNA는 AMBA AHB와 인터커넥트 버스 매트릭스를 대체할 수 있으며, 다중 채널을 동시에 보장하고 다양한 토폴로지를 지원가능 하도록 설계되어 사용하는 IP 수에 따라 설계자에 의해 다양한 토폴로지를 선택할 수 있다. 한편, SNA는 적은 수의 인터페이스 와이어를 가지기 때문에 오프 칩 버스로도 사용될 수 있다. 제안된 버스 구조는 시뮬레이션과 어플리케이션 동작을 통해 검증이 완료되었다.

Abstract

Up to date, a lot of bus protocol and bus architecture are released though most of them are based on the shared bus architecture and inherit the limitation of performance. SNP (SoC Network Protocol), and hence, SNA (SoC Network Architecture) which are high performance on-chip-bus protocol and architecture, respectively, have been proposed to solve the problems of the conventional shared bus. We refine the SNA specification and improve the performance and functionality. The performance of the SNA is improved by supporting simultaneous routing for bus request of multiple masters. The internal routing logic is also improved so that the gate count is decreased. The proposed SNA employs XSNP (extended SNP) that supports almost perfect compatibility with AMBA AHB protocol without performance degradation. The hardware complexity of the improved SNA is not increased much by optimizing the current routing logic. The improved SNA works for IPs with the original SNP at its best performance. In addition, it can also replace the AMBA AHB or interconnect matrix of a system, and it guarantees simultaneous multiple channels. That is, the existing AMBA system can show much improved performance by replacing the AHB or the interconnect matrix with the SNA. Thanks to the small number of interconnection wires, the SNA can be used for the off-chip bus system, too. We verify the performance and function of the proposed SNA and XSNP simulation and emulation.

Keywords : on-chip-network, on-chip-bus, SoC bus, SNA, crossbar router, AMBA AHB, SNP

I. 서 론

* 학생회원, 숭실대학교 전자공학과
(Dept. of Electronic Engr., Soongsil University)
** 정회원, 숭실대학교 정보통신전자공학부
(School of Electronic Engr., Soongsil University)
※ 본 연구는 숭실대학교 교내연구비 지원으로 이루어
졌음
접수일자: 2006년2월22일, 수정완료일: 2007년3월15일

공정기술과 EDA 툴의 발전에 따라서 하나의 실리콘 다이(die)에서 보다 많은 IP 블록의 통합이 기술적으로 가능하게 되었다. 이런 기술의 발달로 멀티미디어와 통신 등 연산 작업량이 많은 곳에서 필요로 하는 병렬 처

리 연산이 요구되고 있다. 다중 프로세서 SoC (System-on-Chip)에서는 데이터 통신과정에서 병목현상이 발생하며, 공유 통신 자원에 대한 접근을 동시에 요청할 때 시스템의 성능 저하를 막을 수 있는 효율적인 방법이 필요하게 된다. 이러한 문제는 기존의 SoC 버스가 공유버스(shared bus)구조를 가짐으로써 병목현상이 발생하기 때문이다.

현재 세계 시장의 70% 이상을 차지하고 있는 ARM 사의 AMBA AHB는^[1] 기존 타 버스에 비해 훨씬 간단한 프로토콜을 가지기 때문에 소규모 SoC에서는 비교적 쉽게 사용할 수가 있었다. 이에 따라 많은 IP 벤더들이 AHB 호환 IP를 개발하여 SoC 시장에서 AHB는 온 칩 버스의 대표자리를 굳혀 왔으나, 이처럼 간단한 버스 프로토콜은 SoC의 규모가 방대해지면서 그 효율성의 한계를 드러내고 있다. 즉, AHB는 단순한 SoC 설계에는 적합하지만 복잡한 SoC용으로는 그 기능이 부족한 것이다. 따라서 AHB의 문제점을 극복하는 새로운 SoC 버스 표준의 필요성이 절실하다.

또한 기존 SoC 버스에서는 대부분 버스 대역폭이 고정되어 있기 때문에 새로운 IP가 추가되면 하나의 IP가 사용가능한 버스 대역폭이 감소하는 것이 일반적이다. 이러한 대역폭의 병목 등의 이유로 하나의 버스에서 사용될 수 있는 마스터의 수를 제한하였다. 더 많은 마스터를 사용하기 위해서는 복잡한 계층구조(hierarchical topology)를 갖게 된다. 개발자는 이러한 부분에 익숙하지 않기 때문에 프로토콜에 익숙해지기 위해서는 많은 시간이 필요하게 된다. 또한 계층구조를 사용했을 때 브리지를 거쳐서 통신을 해야 하기 때문에 데이터의 전달지연(propagation latency)이 증가하게 되며 하나의 집중화된 브리지(centralized bridge)는 공유버스 문제와 같이 병목문제를 발생시키는 요인이 될 수 있다. 한편 이런 문제를 해결하기 위해 만든 인터커넥트 매트릭스는 브리지에 의한 문제는 해결하였지만 면적이 크게 증가하고 기존 AHB 기반 시스템의 경우에는 재설계를 해야하는 문제가 있다.

기존 버스에 대한 대안 중 하나로 연구되고 있는 NoC (Network on Chip)의 경우도 아직까지 구현 형태가 지나치게 복잡하고 중재 알고리즘 처리를 위한 계산량이 많아 오히려 전력 소모 증가와 성능 저하를 가져올 수 있다. 이는 기존의 슈퍼컴퓨터나 서버 제작을 위한 네트워크 구현법을 그대로 모방하기 때문에 생기는 문제이다.^[2] Gurrier 등은 버스와 네트워크 구조를 사용할 때 장/단점에 대해서 언급했는데 네트워크 구조를

사용할 때 내부구조의 특성상 추가적인 잠복기(latency)가 발생하며, 상당한 실리콘 면적을 차지하게 된다는 것을 주요 단점으로 지적하였다.^[3]

대부분의 기존 SoC 버스구조는 중재 토폴로지(topology)를 갖는 다중 버스 분할 연결을 지원하고 동시에 적절한 확장성을 지원하도록 설계되었다. AMBA^[4], Wishbone^[5], CoreConnect^[6] 등이 적절한 예로 볼 수 있다. 이들은 중재정책 및 추가적인 몇몇 특징 이외에는 서로 유사한 구조를 가지고 있기 때문에 AHB 버스와 유사한 문제점을 갖고 있다.

따라서 이러한 문제점들을 해결하기 위해서 단순한 구조를 가지면서 동시에 다중 마스터에 대해서 다중 채널을 제공함으로써 대역폭을 확보하며, 버스에 추가되는 IP의 증가를 고려하여 확장이 용이한 SNP(SoC Network Protocol)와 SNA(SoC Network Architecture)가 제안되었다.^[7,8]

SNA는 주소와 대부분의 제어 신호를 데이터 선에 포함시키고 제어 신호선의 추가를 최소화 하도록 하여 연결선의 수를 대폭 줄인 SNP를^[8] 인터페이스 프로토콜로 사용한다. SNA 내부의 물리적 구성요소는 4N 개의 채널을 갖는 크로스바 라우터(Crossbar Router)와 이들의 상호연결을 중재하는 광역 중재자(Global Arbiter), 그리고 IP 또는 sub-system을 크로스바 라우터와 연결시키는 스위치 래퍼/브리지(Switch Wrapper/Bridge)로 구성된다. 그러나 사각형 모양의 크로스바 라우터로 인해 하나의 IP 증가로 크로스바 라우터 한 개가 증가해야 하는 문제와 다중 채널을 제공함에도 라우팅은 한 번에 하나씩만 이루어지는 단점이 있었다.

한편 다수의 버스 구조와 프로토콜이 발표되었지만, 기존에 존재하는 수많은 AHB 기반의 IP들을 무시할 수 없기 때문에 AHB 기반 IP를 SNA에 쉽게 연결할 수 있는 AHB-to-SNP 변환기가 필요하고 이를 위해 기존 SNP는 XSNP^[9]로 확장되었다. XSNP는 SNP 기반 IP와 함께 AHB 기반 IP와의 호환성을 유지할 수 있다.

본 논문에서는 다중 마스터의 다중 버스 요청을 동시에 처리함으로써 성능을 향상시키고 크로스바 라우터의 구조를 다양화하여 유연한 토폴로지를 갖는 개선된 SNA 구조를 제안한다. 기존 버스 구조에서는 다중 중재(arbitration)를 위해 다중 레이어 구조를 두어야했지만, 이러한 제약 없이 SNA의 라우팅 경로는 모든 IP간에 동시에 만들어질 수 있고 라우팅 로직이 개선되어 하드웨어 복잡도가 감소되었다. 또한 AMBA AHB와 호환성을 증대시키기 위해 큰 성능 감소 없이 호환 가

능한 XSNP를 인터페이스 프로토콜로 사용하고, 이에 기반한 프로토콜 변환기를 설계하였다. XSNP 기반의 SNA에서는 AHB 기반의 IP뿐 아니라 SNP 기반의 IP도 성능 감소 없이 동작이 가능하다. 따라서 기존 AMBA AHB 또는 인터커넥트 매트릭스를 대체할 수 있으며, 동시에 다중 채널을 보장한다. 또한 SNA는 기존 AMBA AHB I/O 수의 약 60%로 동등한 버스 대역폭을 지원할 수 있기 때문에 다중 칩 통신이 필요한 시스템에서 오프 칩 버스 구조로도 활용될 수 있는 특징을 가진다.

II. 기존 제안된 구조

기존 제안된 SNA는 다중 마스터에 대해 다중 채널을 동시에 제공하며, 버스 구조가 단순하고 AMBA AHB 호환성을 가진다는 주요 특징을 가진다. SNA는 다중 채널을 제공하는 크로스바 라우터(XR), 크로스바 라우터 간의 연결을 중재하는 광역 중재기(GA), IP 또는 지역버스를 갖는 서브 시스템(sub-system)과 크로스바 라우터 간의 연결을 구성하는 스위치 래퍼(SW)/스위치 브리지(SB)로 구성된다. 그림 1에서는 SNA의 2개의 크로스바 라우터와 5개의 스위치 래퍼, 하나의 스위치 브리지, 하나의 광역 중재기를 사용하여 시스템을 구성한 구조를 나타내고 있다. SNA는 크로스바 라우터의 토폴로지에 따라 유연성과 확장성을 모두 가질 수 있도록 다음 3 가지의 라우팅 모드를 지원한다. 같은

스위치 래퍼간의 IP 연결은 직접 라우팅 (Direct routing, ①), 같은 크로스바 라우터 내의 연결은 지역 라우팅 (Local routing, ②), 크로스바 라우터 간의 연결은 광역 라우팅 (Global routing, ③)으로 정의된다.^[7]

III. 개선된 구조

기존 SNA의 크로스바 라우터는 동시에 최대 4 채널을 제공할 수 있음에도 불구하고 구조적 복잡도를 낮추기 위해 특정 순간에 하나의 라우팅만 수행할 수 있게 설계되었다. 어떤 우선순위가 낮은 버스 요청자가 요구한 채널이 유희(idle)상태라 할지라도 최소한 한 사이클 이상의 다음 라우팅 순서가 돌아올 때까지 기다려야 한다. 기존 대부분의 버스 구조에서 가지는 버스 중재자가 이와 유사하게 동작한다. 이는 기존 버스 구조에서는 동시에 다중 채널을 제공할 수 없기 때문이다. 기존 버스 시스템에서도 종종 다중 채널을 필요로 하기 때문에 다중 레이어 시스템 설계를 한다. 이때 각 분리된 레이어에서는 독립된 중재가 진행된다. 서로 분리된 상태에서 아무 연관성 없이 중재가 진행되는 것으로 사실상 서로 독립된 시스템의 중재와 같다. 그리고 만약 서로 다른 레이어의 IP 간에 채널이 필요할 경우, 레이어간의 다중 채널을 항상 보장할 수 없기 때문에 레이어간 연결이 되지 않을 수도 있으며, 최악의 경우 전체 시스템이 교착상태(dead-lock)에 빠질 수 있다.

이에 반해, 제안된 SNA 구조는 모든 IP들이 서로 유기적으로 연관된 상태에서 다중 채널을 보장할 뿐만 아니라, 동시에 다중 라우팅을 지원할 수 있다. 크로스바 라우터의 크로스바 스위치 컨트롤러는 지역 라우팅과 광역 라우팅을 담당한다. 기존 SNA 구조에서는 동시에 여러 버스 요청이 발생하면 우선순위가 높은 요청부터 처리되었으며, 두 번째 우선순위를 가지는 요청은 채널이 가용한 조건이어도 그 다음 사이클 이후에 처리가 되었다. 그러나 제안된 SNA 구조에서는 다른 IP에 의해 채널이 사용되는 조건이 아니라면 동시에 발생하는 여러 버스 요청을 처리할 수 있게 설계 되었다. 그림 2에서 제안된 SNA 구조의 크로스바 컨트롤러의 내부구조를 보여주고 있다. 이 구조는 두 개의 라우팅 채널을 동시에 만들어 수 있다. 이와 같은 유사한 방법으로 더 많은 채널을 중재할 수 있게 설정될 수 있다. 우선순위로직(priority logic)은 각 버스 요청자로부터 요청 정보를 수신하여 미리 정의된 우선순위 정책에 의해 요청된 라우팅이 처리된다. 수신된 주소값과 (일반적으로 32

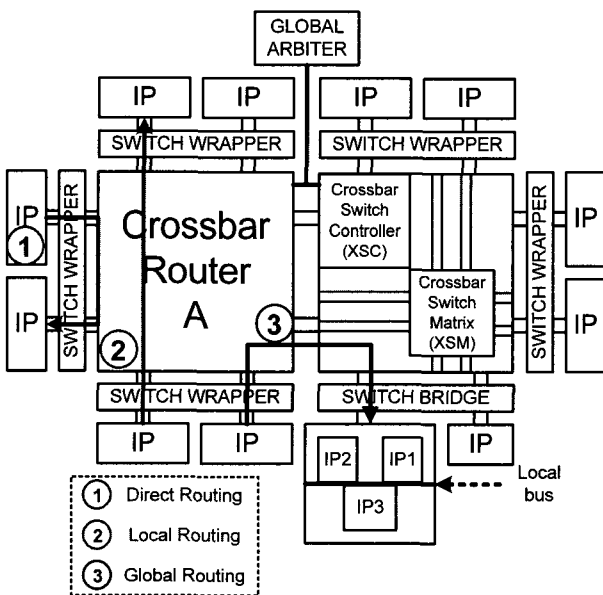


그림 1. SNA를 이용한 2-크로스바 라우터 시스템
Fig. 1. 2-crossbar router system of SNA.

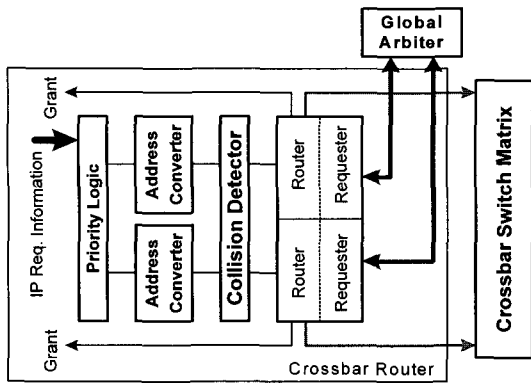


그림 2. 크로스바 스위치 컨트롤러의 내부 구조도
Fig. 2. Block diagram of crossbar switch controller.

비트) 버스 요청 정보는 내부 ID 값으로 (일반적으로 3-5 비트) 변환된다. 이 ID 값은 광역 중재기로 전달되는 물리적 선의 수를 줄이는 목적으로 사용된다. 충돌 탐지기(collision detector)는 목적지가 같은지 여부를 체크하는 역할을 하며, 만약 같을 경우, 이 중 낮은 우선순위를 갖는 버스 요청 신호를 막음으로서 라우팅 충돌을 방지한다. 크로스바 스위치 컨트롤러는 하나 이상의 주소 변환기(address converter)와 요청기(requester)/라우터(router)를 포함하여 기존보다 부품수는 늘었지만 기존의 SNA 구조에 비해 주소 변환기의 로직을 단순화 시킴으로써 전체적으로 하드웨어 복잡도가 감소되었다. 기존 방식에서는 소스 정보와 목적지 정보를 LUT(look-up table)에 모두 저장하였다. 이를 목적지만을 고려하여 라우팅 하는 방법을 적용하여 구조적 복잡도를 보다 낮출 수 있었다. 즉, 모든 라우팅 패스에 대한 정보를 LUT에 포함하던 기존 방식을 스위치 래퍼에서 발생하는 버스 요청신호를 재사용하여 LUT 크기를 줄였다.

제안된 SNA 구조에서는 AMBA AHB와의 호환성을 유지하기 위해 XSNP를^[9] 인터페이스 프로토콜로 사용한다. XSNP는 SNP와의 호환성을 유지하면서도 AMBA AHB 프로토콜을 효과적으로 SNP로 변환시킬 수 있는 규격을 가지고 있다. 스위치 래퍼와 크로스바 스위치 매트릭스는 XSNP를 수용하기 위해 각 링크마다 페이즈(phase) 신호를 1 비트 증가시킴으로써 간단히 수정되었다. 보다 주목할 부분은 AMBA AHB 프로토콜을 SNP로 변환하기 위한 기존의 플로우 컨트롤러에서 AHB-to-XSNP/ XSNP-to-AHB 변환기(이하 프로토콜 변환기)로의 변화이다. 그림 3에서 프로토콜 변환기와 스위치 래퍼의 구조를 보여주고 있다. 하나의 프로토콜 변환기는 마스터와 슬레이브 인터페이스를 하

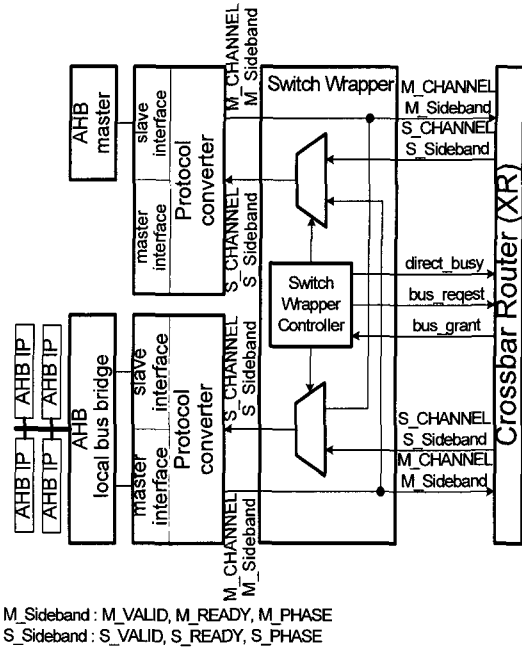


그림 3. 프로토콜 변환기와 스위치 래퍼의 구조도
Fig. 3. Architecture of protocol converter and switch wrapper.

나씩 가진다. 이는 마스터, 슬레이브, 마스터와 슬레이브, 버스 브리지를 사용하는데 편의를 도모한다. 만약, 하나의 변환기에 마스터와 슬레이브가 하나씩 연결되었다면 각각은 서로 통신할 수 없으며, 항상 특정 순간에는 둘 중 하나만 통신이 가능하다. 하나의 크로스바 라우터당 최대 16 개의 AMBA AHB IP (8 개의 마스터, 8개의 슬레이브)가 연결될 수 있으며, AHB 브리지를 이용하면 최대 8개의 지역 버스를 연결할 수 있다.

기존의 SNA가 사각형 구조의 크로스바 라우터만을 제공하여 하나의 크로스바 라우터로 시스템이 완벽하게 구성된 경우 하나의 IP를 추가하기 위해서는 하나의 크로스바 라우터를 추가하거나 하나의 링크를 스위치 브리지와 지역 버스를 갖도록 바꾸어야 했다. 그러나 제안된 SNA 구조는 크로스바 라우터를 이용하여 유연성 있는 토폴로지를 설정되어질 수 있다. 그림 4에서 다양한 구조의 SNA 토폴로지를 보여주고 있다. 그림 4(a)/4(b)와 같은 구조는 하나의 크로스바 라우터에 연결될 수 있는 IP수는 적지만 각각의 크로스바 라우터의 동작 속도는 빠르고 면적은 작아진다. 또한 크로스바 라우터의 포트수가 작아질수록 글로벌 라우팅의 횟수가 빈번해지기 때문에 전체 라우팅 사이클이 증가할 수 있으며, 이로 인해 광역 중재기에서 성능 병목현상이 발생할 수 있다. 반면, 그림 4(c)/4(d)와 같은 구조는 하나의 크로스바 라우터에 연결될 수 있는 IP수는 많지만

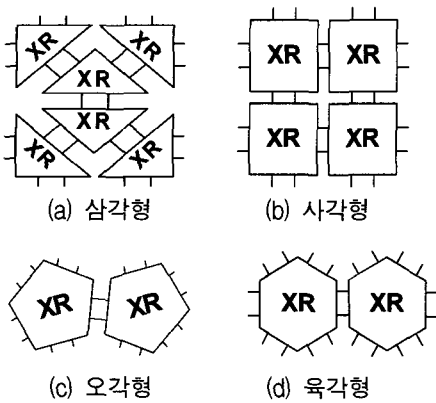


그림 4. 다양한 구조의 SNA 토폴로지
Fig. 4. Various architecture of SNA Topology.

각각의 크로스바 라우터의 동작 속도는 다소 감소하게 되고, 하나의 크로스바 라우터의 면적이 증가한다는 단점이 존재한다. 하지만 광역 라우팅의 빈도수가 감소하여 라우팅 사이클 수가 크게 증가하지 않기 때문에 라우팅으로 인한 성능 감쇠현상은 발생하지 않는다. 따라서 이러한 trade-off 관계를 이용하여 각 시스템의 특징에 적합한 구조를 선택하여 시스템을 구성할 수 있는 유연성을 지원한다. 또한 사각형 크로스바 라우터에서 IP가 하나 증가할 경우 크로스바 라우터를 추가하는 대신 오각형 크로스바 라우터를 이용하면 하나의 크로스바 라우터로 시스템이 구성되고 광역 중재기는 필요 없어진다.

SNA/SNP의 또 다른 장점은 오프칩 버스로도 사용 가능하다는 것이다. 만약 어떤 시스템이 단일 칩으로 구성되지 않고, PCB나 SiP (system-in-package) 기술을 사용하여 다중 칩으로 구현 된다면 AMBA AHB와 같은 온 칩 프로토콜에서 PCI와 같은 오프 칩 프로토콜로의 변환이 필요 할 것이다. 왜냐하면 대부분의 온칩 버스 구조는 오프 칩 버스로 부적절하기 때문이다. 예를 들어, AMBA AHB 인터페이스가 대략 130 개의 I/O 핀을 요구하므로 PCB나 칩 사이의 연결시 상당한 부담이 된다. 또한 어떤 시스템이 슬레이브로만 구성된 것이 아니라면 두 배 (약 260 개)의 I/O 핀이 요구되기 때문에 시스템 비용이 더욱 증가하게 된다. 그러나 XSNP는 레지스터 세그먼트 전송을 지원하며 76 개의 I/O 핀이 요구된다. 따라서 SNA는 오프칩 통신용 버스로 사용될 수 있는 장점을 가진다.

IV. 동작 및 검증

SNA를 SoC 플랫폼에 적용시키기 위해 AMBA

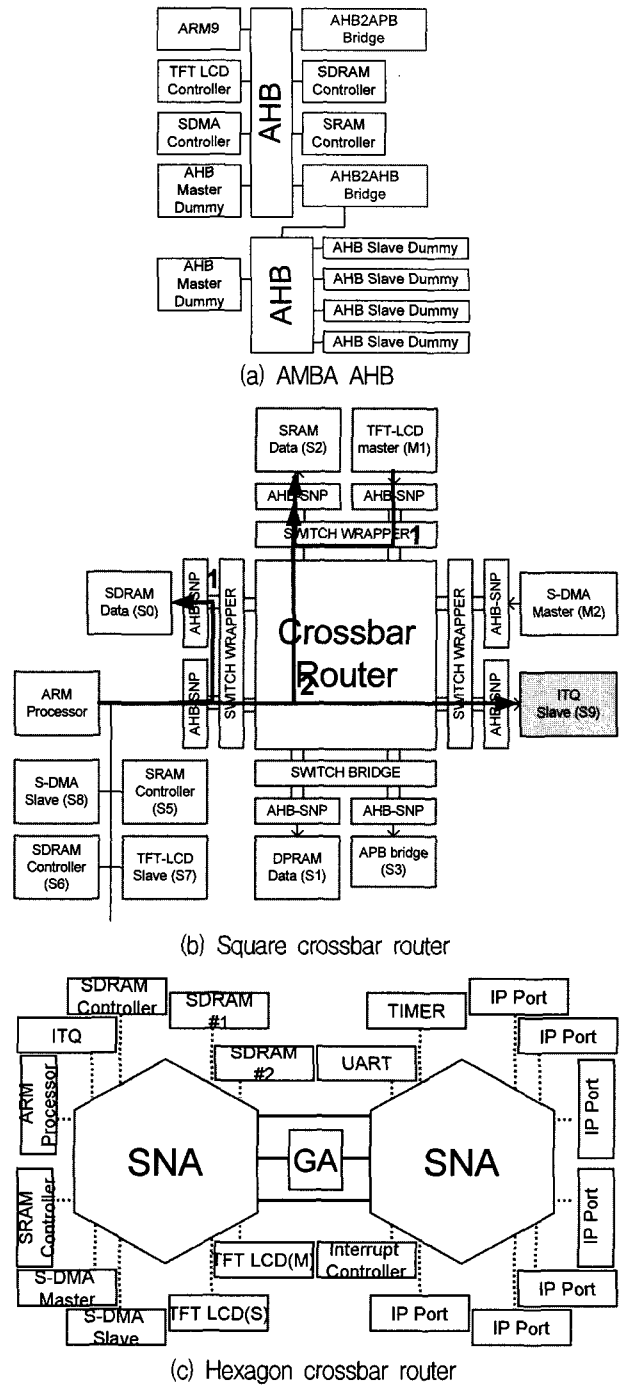


그림 5. 다양한 버스 기반의 SoC 플랫폼의 구조도.
Fig. 5. SoC Platforms using AHB and SNA.

AHB 기반 SoC 플랫폼을 먼저 구성하였다. 그림 5(a)는 FPGA 보드에 포팅된 AHB 기반의 SoC 플랫폼의 버스 구조를 보여주고 있다. 그림 5(b)는 AHB 기반의 버스 구조 플랫폼을 하나의 모듈로 제작된 단일 크로스바 라우터로 구성된 SNA 모듈과 프로토콜 변환기를 이용하여 AHB IP들을 위한 버스로 재구성하였다. 또한 CPU만 접근하는 환경설정을 위한 슬레이브는 CPU와 같은 지역 버스에 연결하였다. 그리고 SNA로 재구성된

AHB IP 기반 시스템에 H.264의 한 동작 요소인 ITQ(Inverse Transform and Quantization) IP(슬레이브)를 연결하였다. 여기에 JM 소프트웨어를 이용하여 H.264 디코더를 구현하였다. 즉, H.264 디코더 중에서 ITQ 부분만을 IP(하드웨어 블록)화 하고 디코더의 나머지 부분은 JM 소프트웨어에서 처리하도록 되어 있다. 그림 5(b)에서 구현된 H.264 디코더의 구조와 접근 경로(access path)를 보여주고 있다. 크로스바 라우터 하나를 이용하여 시스템을 구성할 경우 그림 5(b)와 같이 여러 개의 지역 버스를 사용해야 하지만 그림 5(c)와 같이 그림 4(d)의 토폴로지를 이용할 경우 로컬 버스를 구성하지 않고 IP 연결만으로 시스템을 구성할 수 있다. 또한 여유 IP 포트를 확보할 수 있기 때문에 보다 많은 수의 IP를 추가 연결할 수 있어 확장이 용이해진다. 이렇게 두 개 이상의 크로스바 라우터를 사용할 경우 크로스바 라우터간의 통신을 위해 광역 중재기(GA)가 추가적으로 필요하게 된다.

그림 6에서 제안된 SNA에서 동시에 라우팅 처리되는 과정을 타이밍도로 보여주고 있다. A에서 4 개의 버스 요청이 동시에 발생되어, B에서 우선순위가 높은 2 개의 라우팅이 처리되었다. B에서 처리된 라우팅보다 우선순위가 낮은 버스 요청은 한 사이클 후인 C에서 처리된다. 만약 동시에 버스 사용을 요청한 신호가 같은 목적지를 가리킬 경우 충돌 탐지기는 이를 탐지하여(D) 이중 우선순위가 높은 버스 요청을 먼저 처리한다(E). 높은 우선순위를 가진 요청자라도 기존 SNA 구조에서는 우선순위가 더 높은 글로벌 라우팅으로(F) 인해 3 사이클을 지연시켰지만(그림 6(b)), 제안된 SNA 구조에서는 버스 요청이 동시에 처리되기(F) 때문에 3 사이클 이득을 볼 수 있다.

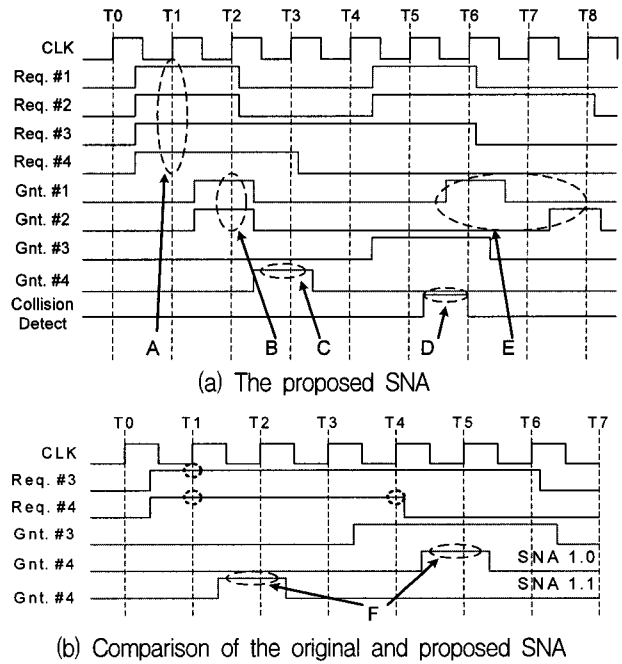


그림 6. 동시에 다중 라우팅이 처리되는 타이밍도.
Fig. 6. Timing diagram showing multiple routing in SNA.

개선된 SNA를 Xilinx FPGA에 구현하고 동작을 검증하였다. 그림 7에 쓰기 동작을 수행할 때 FPGA 내부에서 AHB와 XSNP의 신호 변환 과정을 Chipscope를 이용하여 측정된 결과를 보여주고 있다. 소프트웨어와 하드웨어 사이의 동작 사이클의 차이로 잠복기가 발생한다. 이 잠복기는 ARM 프로세서가 DRAM으로부터 명령어 정보를 읽어야 하기 때문에 발생하는 것이다. 그림 7에서 ARM_HADDR의 주소가 M_CHANNEL로 전달되고 다시 HADDR로 변환되고 있음을 보여준다. M_CHANNEL로의 변환은 마스터의 프로토콜 변환기에서, HADDR로의 변환은 슬레이브의 프로토콜 변환기

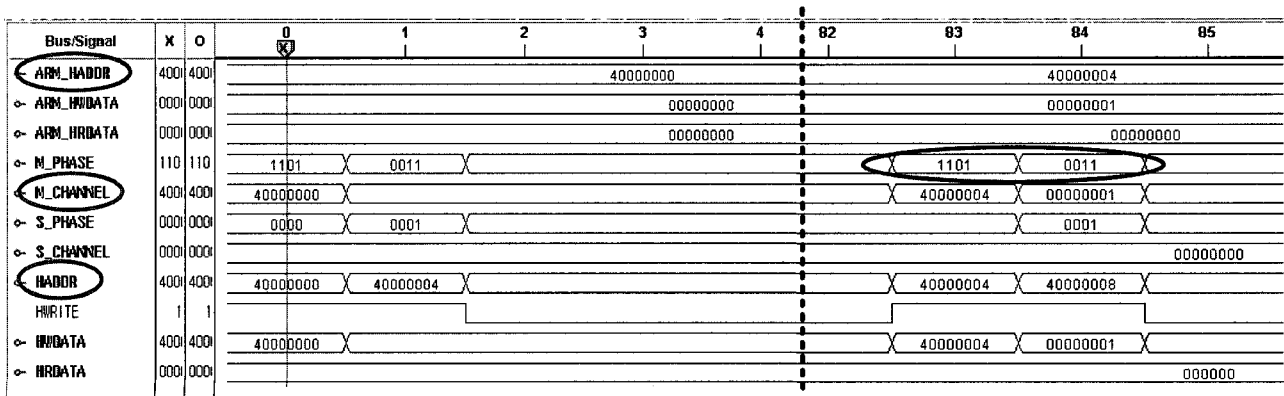


그림 7. FPGA 내부에서 AHB와 XSNP 신호의 변환과정을 측정된 결과 파형
Fig. 7. Captured signals of AHB-to-XSNP conversion in FPGA.

에서 각각 이루어진다. 채널의 정보를 나타내기 위해 M_PHASE 신호는 “1101” (쓰기 주소), “0011”(최종 데이터) 값을 가진다.

V. 설계 및 구현

표 1에서는 그림 4에 나타난 다양한 크로스바 라우터 타입을 설계하고 합성한 결과에 대한 비교 자료를 보여 준다. 합성조건은 모두 100MHz 동작주파수를 목표로 하였고, 0.25um CMOS 표준 셀 라이브러리를 사용하였다. 또한 하나의 크로스바 라우터의 면적만을 비교하였기 때문에 광역 라우팅 블록은 제외되었다. 비교란에는 하나의 크로스바 라우터를 구성하는 크로스바 스위치 컨트롤러(XSC)와 크로스바 스위치 매트릭스(XSM)의 면적을 기술하였다. 표 1에서 알 수 있듯이 전체적인 면적 증가는 지원 채널수의 증가로 인한 크로스바 스위치 매트릭스의 면적증가가 주요 원인임을 알 수 있다. 그리고 동작 주파수를 50 - 200MHz 사이에서 변화시키고 합성을 해도 이에 따른 면적의 변화는 거의 없다. 이는 버스가 로직 면적보다는 물리적 와이어 수의 증가가 더욱 중요하다는 것을 의미한다. 즉, 동일한 시스템을 구현할 경우 IP 간에 연결되는 인터커넥션 선의 수가 증가한다면 P&R(place and routing) 시간이 길어지고 물리적 선에 의한 면적이 증가하게 된다. 이는 버스 구조가 로직 기능보다는 물리적 선의 기능이 대부분이기 때문이다. 따라서 이러한 점을 고려하면 시스템 설계과정에서 trade-off 포인트를 쉽게 찾을 수 있다.

표 2에서 기존 SNA와 새롭게 제안된 SNA의 면적 정보를 비교 하였다. 전체적으로 기존에 제안되었던 구조에 비해 새로 제안된 구조가 약 10-18%정도의 면적이득이 있었음을 확인할 수 있었다. 합성조건은 0.25um CMOS 표준 셀 라이브러리에서 100MHz 동작속도를 목표로 하였으며, FPGA 구현 시에는 Xilinx Virtex2 디바이스를 이용하였다. 합성 모듈은 하나의 SNA 시스

표 1. 다양한 크로스바 라우터 타입에 따른 면적 비교

Table 1. Area comparison of various crossbar routers.

0.25um @100MHz	1 XR (gate counts)	비 고 (XSC + XSM)
Triangle	~ 3,900	1,100 + 2,800
Rectangle	~ 8,000	1,500 + 6,500
Pentagon	~ 13,300	2,200 + 11,100
Hexagon	~ 20,000	3,000 + 17,000

표 2. 기존 SNA와 새롭게 제안된 SNA의 면적 비교
Table 2. Area comparison of original and proposed SNA.

	Original SNA (4 SW + 1 XR)	Proposed SNA (4 SW + 1 XR)
Xilinx Virtex2 8000 (FPGA)	~ 2,320 LUT	~ 2,760 LUT
CMOS 0.25um @100MHz	~ 11,000 gates	~ 10,000 gates

표 3. AMBA 인터커넥트 매트릭스와 제안된 SNA와의 면적 비교

Table 3. Area comparison of AMBA Interconnect Matrix and the proposed SNA.

	AMBA Interconnect Matrix (8x8 port) [10]	Proposed SNA (8 Converters + 4 SW + 1 XR)
Xilinx Virtex2 (FPGA)	NA	~ 4,542 LUT
CMOS 0.25um @100MHz	~ 25,000 gates	~ 27,100 gates

템을 구성하기 위한 4 개의 스위치 래퍼와 1 개의 크로스바 라우터이며, 이는 상기 언급된 토폴로지 모양 중 그림 4(b)를 구성할 수 있다.

표 3에서는 AMBA 인터커넥트 매트릭스[10]와의 면적 정보를 비교하였다. 두 설계물 모두 100MHz의 동작속도를 목표로 하였으나 공정이 달라 제시된 게이트 수는 다소 차이가 있을 수 있다. 인터커넥트 매트릭스는 기존 AHB의 공유버스에서 오는 문제점들을 해결하기 위해 다중 레이어를 쉽게 지원할 수 있게 개발된 버스 모델이다. 표 3에서 비교된 인터커넥트 매트릭스는 8 개의 마스터와 8 개의 슬레이브 IP를 수용할 수 있다. 일반적으로 AMBA 인터커넥트 매트릭스는 마스터와 슬레이브 포트 수가 고정되기 때문에 토폴로지의 유연성이 다소 떨어지게 된다. 반면에 SNA는 마스터/슬레이브 구분이 없이 임의의 링크에 연결 가능하고 다양한 토폴로지를 가지고 있어 시스템 구성에 유연성이 크다. 또한 제안된 SNA와 약 8%의 면적차이를 보이고 있지만, 실제 버스는 로직 크기 자체만으로 우열을 가리기는 어렵다. [11]에서 언급했듯이 실제 버스에는 와이어의 길이(면적)가 포함된다. 이는 AHB가 130 개 이상의 I/O를 요구하고 IP 사이에 먼 거리를 달려야 하는 반면 SNA는 76 개의 I/O를 가지고 선이 각 IP와 버스 블록 사이의 짧은 거리만 연결되면 되므로 AHB에 비해 요구되는 물리적 면적이 작다고 할 수 있다.

VI. 결 론

본 논문에서 기존 SNA 구조를 개선시킨 새로운 SNA 구조를 제안하였다. 제안된 구조는 다중 채널을 보장하면서 동시에 다중 라우팅을 처리할 수 있다. 따라서 라우팅의 병렬성을 증가시켜 다중 채널 지원의 특징을 살렸다. 또한 성능 감소없이 AMBA AHB와 호환 가능한 XSNP^[9]를 인터페이스 프로토콜로 사용하여 기존 AHB IP를 문제없이 연결할 수 있게 되었다. 제안된 SNA 구조의 추가된 특징들로 인해 복잡한 하드웨어 구조가 필요함에도 불구하고, 내부 구조의 최적화로 인해 면적 증가를 막을 수 있었으며, 구현한 결과 AMBA 인터커넥트 매트릭스보다 약 8% 정도 큰 면적을 차지한다. 그러나 AHB에서 사용하는 와이어 수보다 40% 정도 적은 I/O 수를 사용하기 때문에 백엔드(back-end) 작업에서 추가되는 물리적 와이어 면적을 줄일 수 있다. 또한 다양한 크로스바 라우터 구조를 설계하여 시스템의 특징에 따른 다양한 토폴로지를 지원할 수 있게 하였으며, 성능/동작 검증을 위해 H.264 플랫폼 시스템을 구성하여 FPGA에서 검증하였다. 또한, SNA를 이용하여 시스템을 구현할 경우 SiP와 같은 다중 칩 형태의 시스템에서 76 개의 I/O를 이용해 온 칩과 오프 칩 통신을 직접 할 수 있으며, 유연성과 쉬운 확장성 특징을 갖는 토폴로지 지원이 가능하다.

참 고 문 헌

- [1] Inside the New Computer Industry, issue 138, Jan. 2001.
- [2] F. G. Moraes, N. L. V. Calazans, A. V. de Mello, L. H. Möller, L. C. Ost, "Hermes: an Infrastructure for Low Area Overhead Packet-switching Networks on Chip," *Integration the VLSI Journal*, 38(1), Oct. 2004, pp. 69-93.
- [3] P. Gurrier, A. Greiner, "A Generic Architecture for On-Chip Packet-Switched Interconnections," *Proceedings of the conference on Design, Automation and Test in Europe*, Paris, France, pp. 250-256, March 2000.
- [4] ARM, "AMBA Specification, Revision 2.0", 1999.
- [5] W. Peterson, "WISHBONE SoC Architecture Specification, Revision B.2", Silicore Corporation, 2001
- [6] IBM, "CoreConnect Bus Architecture", 1999.
- [7] S. Lee, C. Lee, H-J. Lee, "A New Multi-Channel On-Chip-Bus Architecture for System-On-

Chips," *IEEE SoC Conference*, Santa Clara, CA., pp. 305-308, September 2004.

- [8] J. Lee, H-J. Lee, and C. Lee, "SNP: A New Communication Protocol for SoC," in *Int'l Conf. on Communications, Circuits and Systems*, Chengdu, China, pp. 1419-1423, June 2004.
- [9] E-S. Kim, S. Lee, C. Lee, H-J. Lee, "A High Performance AMBA-compatible SoC Network Protocol for SoC Network Architecture," *IT-SoC Conference*, Seoul, Korea, pp. 252-255, November 2005.
- [10] ARM, ARM DDI 0170A, August 1999.
- [11] P. P. Pande, A. Ivanov, R. Saleh, "Structured interconnect architecture: a solution for the non-scalability of bus-based SoCs," *Proceedings of the 14th ACM Great Lakes symposium on VLSI*, pp.192-195, Boston, MA, USA, April 2004.

저 자 소 개

이 상 헌(학생회원)

제42권 SD편 제2호 참조

현재 숭실대학교 대학원 전자공학과 박사과정

이 찬 호(정회원)

제42권 SD편 제2호 참조

현재 숭실대학교 정보통신전자공학부 부교수