

NoC 시스템에서 Deadlock과 패킷 drop을 감소를 위한 동적 Job Distribution 알고리즘에 관한 연구

정희원 김우주*, 이성희**, 황선영*

Dynamic Job Distribution Algorithm for Reducing Deadlock & Packet Drop Rate in NoC

Woo-Joo Kim*, Sung-Hee Lee**, Sun-Young Hwang* *Regular Members*

요 약

본 논문은 NoC 시스템에서 수행되는 멀티미디어 application에서 deadlock과 패킷 drop율을 최소화하기 위한 hybrid 구조의 네트워크 토폴로지와 job distribution 알고리즘을 제안한다. 제안된 NoC 시스템은 멀티미디어 application에 따라 패킷 drop율이 최소화될 수 있는 네트워크 토폴로지를 적용하여 sub-cluster에 배정하고, 수행되는 application은 sub-cluster 내의 네트워크 상황을 고려하여 동적 round-robin 방식의 스케줄링을 수행한다. 본 연구에서는 hybrid 네트워크 토폴로지와 동적인 job distribution을 통해 네트워크에 발생하는 트래픽을 분산시킴으로써 패킷 deadlock과 패킷 drop율을 최소화하는 알고리즘을 제시하였다. 제안된 시스템의 성능을 검증하기 위해 임베디드 시스템에서 사용되는 멀티미디어 application 중 MPEG4 동영상 재생, MPEG1 audio layer-3(MP3) 재생, GPS 위치 계산, OFDM 복조를 대상으로 실험하였다. 실험결과 제안된 job distribution 알고리즘에 적용된 시스템은 기존의 APSRA 알고리즘에 비해 패킷 drop율이 평균 13% 감소하고 면적은 2.7% 증가하였다. 또한 deadlock 감소의 비교 대상으로 흔히 사용되는 XY 알고리즘을 본 구조에 맞게 변경한 알고리즘에 대해 패킷 drop율은 평균 23.9% 감소하고 면적은 3.0% 증가하였다.

Key Words : Network-on-Chip, Packet drop rate, Deadlock, Topology, NoC

ABSTRACT

This paper proposes a dynamic job distribution algorithm in a hybrid NoC structure which can improve system network performance by reducing deadlock and packet drop rate for various multimedia applications. The proposed job distribution algorithm schedules every job to the sub-cluster where packet drop rate can be minimized for each multimedia application program. The proposed job distribution algorithm and network topology targets multimedia applications frequently used in modern embedded systems, such as MPEG4 and MP3 decoder, GPS positioning system, and OFDM demodulator. Experimental results show that packet drop rate was reduced by about 13.0%, and chip area was increased by about 2.7% compared to the APSRA algorithm. When compared to the XY algorithm popularly used for benchmarking, the packet drop rate was reduced by about 23.9%, while chip area was increased by about 3.0%.

※ 본 연구는 2008년도 「서울시 산학연 협력사업」의 「나노IP/SoC설계기술혁신사업단」의 지원으로 이루어졌습니다.

* 서강대학교 전자공학과 캐드 및 임베디드시스템 연구실(hwang@sogang.ac.kr), ** 삼성전자 DS총괄 메모리 SSD/CARD PJT
논문번호 : KIC2007-09-403, 접수일자 : 2007년 9월 11일, 최종논문접수일자 : 2008년 7월 7일

I. 서 론

Nano-electronic 시대에 접어들면서 천만 개 이상의 트랜지스터를 집적할 수 있게 되면서, 여러 기능 모듈과 코어들이 집적되어 하나의 시스템이 하나의 칩으로 구현되는 System-on-a-Chip(SoC)이 활발히 구현되고 있다^{[1][2]}. SoC의 성능 개선을 위한 최근 연구에서 많은 수의 코어와 IP 블록들이 집적되고 있고, 코어의 성능을 최대한 활용하기 위해 여러 코어와 IP 들이 멀티 프로세싱을 수행하게 되었다. 이로 인해 최근의 SoC에는 코어 간의 데이터 통신이나 공유 메모리 접근을 위한 트래픽이 증가하여 고속 데이터 통신에 관한 연구가 요구되었다. 코어의 처리 속도 증가에도 불구하고 트래픽 증가는 네트워크 오버헤드로 인해 데이터 전송을 지연시키게 되어 시스템 성능의 저하를 초래한다. 증가된 트래픽은 네트워크상에서 특정 구간에서 집중되어 시스템 전체의 성능이 저하하는 deadlock이 발생한다^[2]. 이러한 네트워크의 문제를 해결하기 위해 버스 사용 권한 획득을 위한 arbitration이 필요한 전통적인 버스 구조에 비해서 더욱 scalable한 네트워크 구조와 코어간 통신 대역폭을 넓히는 방안이 요구되었으며, 네트워크의 대역폭을 증가시킬 경우 wire에서의 전력 소모가 증가하고 시스템 면적이 증가되기 때문에 대역폭을 크게 증가시키지 않으면서도 충분한 throughput을 낼 수 있는 네트워킹 방법이 요구된다. 이러한 문제의 해결을 위하여 Network-on-Chip (NoC) 또는 on-chip 네트워크가 제안되었다^{[2][3]}. NoC에 관한 연구가 활발히 진행되면서 네트워킹의 오버헤드를 최소화하기 위해 네트워크 라우팅을 동적으로 처리하는 방식이 제안되었고, 트래픽 증가와 코어 증가에 따른 스위치의 면적 감소를 위해 워홀 라우팅 등의 방식이 연구되었다^{[4][5]}. 그리고 시스템에 집적되는 코어와 IP 간의 네트워크와 위치 등을 전력 소모, 면적, 네트워크 토폴로지 개선 등을 통한 처리 속도 증가를 위해 다양하게 변형하는 연구가 진행되고 있다^[5].

기존의 deadlock 감소를 위한 연구는 라우팅 알고리즘을 선택하고, 선택된 알고리즘이 deadlock을 발생시키는지 검사한 후, deadlock이 발생할 수 없도록 추가적인 하드웨어나 라우팅 알고리즘을 더욱 개선하는 방식으로 진행되었다^[4]. 이는 NoC 시스템 설계와 함께 시스템에서 사용될 application에 따른 deadlock을 확인하고 이를 개선하는 알고리즘

을 설계에 반영하는 방식으로 deadlock이 발생할 상황을 미리 확인하고 설계과정에서 미리 해결할 수 있는 장점을 갖는다^[5]. 공유 네트워크 리소스에 대한 dependency graph를 통해 현재 dependency가 있는 리소스 사이에 상호 사용 요청이 발생하는 지점을 찾아낸다. 이러한 싸이클이 발생한 곳은 리소스 사용을 위해 패킷 대기 시간이 증가하게 되므로 deadlock이 발생할 가능성이 커진다. 따라서 공유 리소스간 사용 요청 싸이클이 형성된 리소스 또는 라우터 사이에 wire를 추가하거나 해당 패킷을 다른 라우터를 통해 우회 경로 사용을 유도하는 알고리즘을 통해 deadlock을 미리 막을 수 있다^{[5][6]}. 최근 NoC를 사용하는 임베디드 시스템은 동영상, 음악, GPS 시스템을 통한 네비게이션 등 많은 분야에서 활용되고 있다. 최근 임베디드 시스템은 PDA와 같이 하나의 기기를 활용하여 소프트웨어 설치를 통해 무선 통신, PMP, MP3P 등의 application 등을 실행하게 되며, 이러한 모든 application을 시스템 설계시에 예측하고 이에 대한 deadlock을 모두 예방하기 어렵다. 따라서 임베디드 시스템에서 사용되는 application의 특성에 따라 패킷 전송이 집중되는 구간을 분산시키는 다양한 구조를 가질 수 있도록 NoC 시스템의 네트워크 구조를 configurable 하게 설계하고, 실시간적으로 패킷 drop을 최소화시키는 알고리즘의 적용이 요구된다.

본 연구에서는 NoC 시스템에서 deadlock으로 인해 패킷이 drop되어 발생하는 시스템 성능 저하를 개선하기 위하여 NoC 시스템에서 수행되는 멀티미디어 application에 따라 패킷 drop이 적게 발생하는 네트워크 토폴로지로 sub-cluster를 구현하였다. 또한 job distributor를 통해 수행될 application을 패킷 drop이 적게 발생하는 sub-cluster로 각각 배정하고, 이후 수행되는 application은 sub-cluster 내의 네트워크 상황을 고려한 동적 round-robin 방식으로 할당 하는 알고리즘을 제안하였다. 설계 검증을 위해 MPEG-4, MP3, GPS 위치 계산, OFDM 복조를 사용하였으며, 4개의 application을 동시에 실행시켰을 때 발생하는 패킷 drop을 비교를 통해 deadlock 감소를 확인하였다.

본 논문의 II절에서는 관련 연구에 관해 기술하고, III절에서는 제안된 알고리즘에 대해 기술하며, IV절에서는 설계된 NoC 시스템에서의 deadlock 감소 결과를 기술한다. V절에서는 본 연구에 대한 결론을 제시한다.

II. 관련 연구

최근의 SoC에는 다수의 코어가 내장되고 다양한 IP 들이 집적되면서 코어들과 IP 간의 통신량이 증가하였다. 코어와 IP 간의 통신이 증가하면서 특정 위치에서 트래픽이 집중되어 네트워크 리소스 사용 대기 시간이 증가하고 패킷 drop이 발생하면서 코어의 성능 증가에 상관없이 시스템 전체의 성능을 저하시키는 deadlock 현상이 발생한다. 이를 해결하기 위해 NoC 기술과 deadlock free 라우팅 알고리즘이 제안되었다. 1절에서는 여러 코어들과 IP 들간의 네트워크 성능 개선을 위해 기존의 연구에서 제안된 NoC에 대해 설명하고, 2절에서는 임베디드 시스템에서 많은 application 각각에 대해 패킷 drop을 최소화 시키는 NoC 네트워크 토폴로지에 대해 설명한다. 2.3절에서는 NoC 시스템의 deadlock 현상과 deadlock free 알고리즘에 대해 설명한다.

2.1 Network-on-a-Chip

SoC 시스템에서 코어가 증가하고 IP 들간의 트래픽이 증가하면서 기존의 버스 구조로 트래픽을 고속으로 처리하기에 한계에 이르렀다. 증가된 트래픽을 시스템 성능 저하 없이 전송하기 위해서 다중 레이어 방식의 버스를 사용하기도 하나, 버스 사용 권한 획득을 위한 arbitration에 대한 오버헤드 등으로 인해 증가된 트래픽을 시스템의 성능 저하 없이 처리하기 어렵다. 이에 대한 해결책으로 제안된 NoC는 IP 코어간의 안정적인 통신, 오류 정정, scalable한 성능 등의 장점으로 인해 기존의 SoC 버스에 대한 대안으로서 등장하였다⁵⁾. NoC는 병렬 처리 컴퓨팅에서 사용되는 패킷 스위칭 네트워크와 비슷하며, 패킷의 시작점과 flit의 중점을 찾고, composing과 decomposing 등을 제어할 수 있는 상위 수준 프로토콜을 필요로 한다. NoC의 토폴로지는 2차원의 torus로 구성되며 source 라우팅이 패킷 전달을 위해 사용된다⁶⁾.

많은 양의 데이터를 빠른 속도로 전달하기 위해서는 wire의 대역폭을 키워야 한다. 대역폭 증가에 wiring에 따른 면적이 증가하고 전력 소비가 증가하여 NoC 시스템이 주로 사용되는 임베디드 시스템에서는 단점으로 작용된다. 패킷 전달에 사용되는 wire의 대역폭을 줄이고 면적을 감소시키기 위해 fat-free 토폴로지인 Scalable Programmable Integrated Network(SPIN)이 제안되었다⁷⁾. SPIN 토폴로지는 1 바이트 주소를 가지며 최대 256개의 노드를 갖도록

제안되었으며 SPIN 구조의 대역폭은 모든 트래픽이 트리의 root에 집중되어 있기 때문에 2등분되었으며 대역폭의 크기가 매우 작다⁷⁾. NoC 시스템에서 네트워크 구조는 패킷 전달 속도와 시스템 면적에 직접적인 영향을 주므로 패킷 전달에 효율적이며 면적을 최소화할 수 있는 네트워크 구성이 요구되며, 트래픽이 일부 네트워크 리소스에 집중되어 패킷 전달 불능에 빠지지 못하도록 트래픽을 분산시키는 알고리즘의 개발이 요구된다.

2.2 멀티미디어 application에 따른 패킷 drop 최소의 네트워크 토폴로지

NoC의 토폴로지는 구조에 따라 크게 단순 구조 토폴로지, 복합 구조 토폴로지의 두 가지로 나눌 수 있다. 단순 구조 토폴로지는 mesh, star, bus 구조 등으로 모든 코어와 IP 블록들이 한 가지 형태의 네트워크 구조를 갖는다. 그림 1은 global-star local-bus, global-star local-mesh, global-star local-star 등의 복합 구조 토폴로지를 보인다. 복합 구조 토폴로지는 global 네트워크 토폴로지와 local 네트워크 토폴로지로 구성되며, local/global 토폴로지 모두 각자의 토폴로지 타입과 프로토콜을 가질 수 있다. 즉, local 네트워크 토폴로지간이나 local-global 토폴로지 간에 토폴로지 타입이나 프로토콜을 각자 임의로 가질 수 있다⁸⁾⁹⁾.

NoC 시스템에서의 트래픽은 uniform random traffic과 locality factor를 가진 localized traffic으로

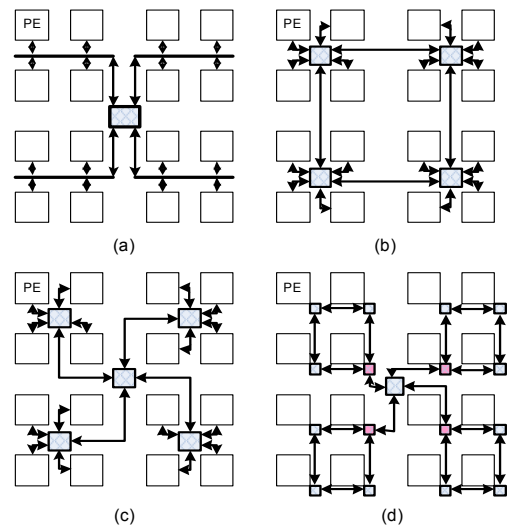


그림 1. NoC의 복합 구조 토폴로지(G: Globla, L:Local).
 (a) G-Star/L-Bus 구조, (b) G-Mesh/L-Star 구조,
 (c) G-Star/L-Star 구조, (d) G-Star/L-Mesh 구조

구분된다. Locality factor는 0에서 1사이의 수로 sub-cluster 내부의 트래픽과 전체 트래픽 사이의 비율을 의미하며, locality factor가 작으면 전체 네트워크의 트래픽이 많다는 뜻으로 NoC 시스템 전체에 병목현상이 발생할 수 있음을 의미한다^[8]. NoC 시스템에서 하나의 application에 해당하는 job을 하나의 sub-cluster 내에서만 처리하게 되면, 외부 네트워크 사용이 현저히 줄어들어 하나의 sub-cluster에서 deadlock 발생해도 다른 sub-cluster는 이에 영향을 받지 않는다.

최근 연구되는 무선 임베디드 시스템은 멀티태스킹을 이용하여 무선 데이터 통신, 동영상과 음성 재생 등 여러 멀티미디어 application 들이 동시에 수행한다. 멀티미디어 application 들은 자기 네트워크의 부하가 최소가 되는 네트워크 구조가 다르다^[10]. 멀티미디어 application에 대해 패킷 drop이 최소가 되는 네트워크 토폴로지는 동영상이나 음성 재생과 같은 입력 데이터가 랜덤하고 입력된 데이터에 따라 계산량이 달라질 수 있는 application에서는 mesh 구조가 적합하다. 또한 통신의 복조기와 같이 외부 모듈에서부터 입력되는 데이터가 시간적으로 불규칙하고 외부 모듈로의 통신이 많은 application의 경우에는 star 구조가 효율적이다. 이러한 멀티미디어 application에 따른 효율적인 구조를 고려하지 않고 일괄적인 네트워크 토폴로지를 사용할 경우 시스템의 네트워크 구조로 인해 많은 패킷 drop이 발생할 수 있다. 따라서 NoC 시스템의 local 네트워크와 global 네트워크는 임베디드 시스템에서 사용될 멀티미디어 application에 따라 서로 다른 네트워크 토폴로지를 가질 수 있도록 하는 hybrid 복합 구조를 사용하여 네트워크 오버헤드를 감소시켜야 한다. 수행되는 application이 모두 다른 여러 임베디드 시스템에 대해 각각 최적의 NoC 시스템을 칩으로 구현해두기는 사실상 불가능하다. NoC 시스템은 sub-cluster의 local 네트워크와 global 네트워크 모두 configurable 하게 설계되어, 임베디드 시스템 설계시 대상 시스템에 최소의 drop율을 가질 수 있도록 스위치를 변경할 수 있는 구조를 채택하여 성능을 향상시킬 수 있다. Configurable 한 스위치 회로가 채택된 NoC 시스템을 이용하여 임베디드 시스템 설계시 programming을 통해 대상 시스템에 최소 패킷 drop율을 갖는 구조로 스위치 회로를 변경하여 사용함으로써 칩 제작비용과 설계 시간 등 시스템 설계 비용을 줄일 수 있다^[11].

2.3 Deadlock 현상과 기존의 deadlock free 알고리즘

NoC 네트워크의 대표적인 문제점은 패킷의 전달 불가능이다. 한 코어에서 송신한 패킷이 목적지에 이르지 않고 중간에 drop 되거나 무한 루프에 빠져 목적지에 도달하지 못하는 경우가 발생할 수 있다^[8]. 라우터의 라우팅 테이블 등의 오류로 인해 패킷이 목적지에 전달되지 않고 라우터들 간에서만 전달되는 livelock이 발생할 수 있다. NoC 시스템 내에 존재하는 라우터 1(R1)과 라우터 2(R2)에서 R1, R2를 거쳐 최종 목적지에 전달되어야 하는 패킷을 R1은 R2로 전송하고 R2는 이를 다시 R1으로 전송하게 되는 경우가 발생할 수 있는데, 이 경우 해당 패킷을 R1과 R2가 서로 전달하기 때문에 무한 루프에 빠지게 된다. 이러한 livelock의 경우 패킷이 목적지에 전달되지 못해서 발생할 수 있는 오류 이외, 라우터의 처리 용량과 트래픽의 양이 지속적으로 증가되어 시스템 전체의 속도가 저하될 수 있다. 패킷이 무한 루프에 빠지지 않도록 하기 위해 최소 거리 라우팅 방법 등을 통해 라우터에서 목적지까지의 거리에 대한 정보를 라우터가 가지고 이에 따라 최소 길이 경로로 패킷을 전달하도록 해야 한다^{[5][6]}.

대부분의 라우터는 패킷을 목적지로 정확하게 운반하기 위해 설계되나 특정 패스에서 라우터의 처리 속도에 비해 많은 패킷이 전달될 경우 큐의 용량에 넘어선 패킷은 목적지까지 전달하지 못하고 이를 폐기한다. 이러한 deadlock 현상이 생길 경우 전송 오류로 인해 많은 시간을 소비할 수 있다^{[5][6]}. 패킷 스위칭 네트워크에서는 패킷이나 플릿이 이웃 라우터로 전달될 때마다 라우터의 큐에 저장한다. Deadlock은 하나 이상의 패킷이 전달될 때 네트워크가 블록된 상태인 경우에 발생하게 되며 deadlock의 전형적인 예를 그림 2에 보인다. 그림 2는 4개의 라우터가 상호 연결된 mesh 구조의 NoC 시스템이며, 4개의 라우터는 목적지에 대한 라우팅 테이블을 통해 패킷을 목적지까지 전달한다. 채널 1(c1)을 사용 중인 패킷 1(p1)이 채널 2(c2)를 기다리고 있으며, c2는 채널 3(c3)을 사용하기 위해 대기 중인 패킷 2(p2)가 점유 중이다. 또한 c3은 채널 4(c4)의 사용을 대기 중인 패킷 3(p3)이 점유하고 있으며, p4는 c1의 사용을 대기하며 c4를 점유한다. 이 경우 모든 패킷은 진행되어야 할 채널이 순환성을 이루며 점유되어 있으므로 다음 라우터로 전달되지 못한다.

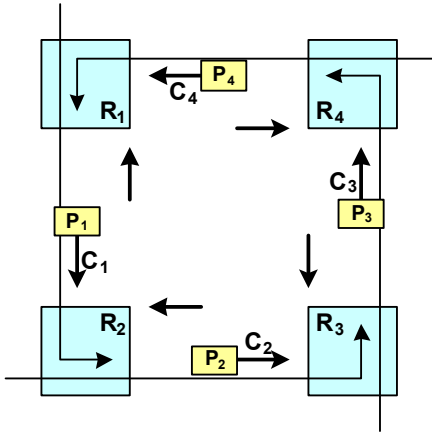


그림 2. Mesh 구조의 NoC에서 deadlock 발생의 예

패킷 전송 경로 상에서 패킷 전달이 순환성을 가지게 되어 발생하는 deadlock 외에도, 패킷 목적지의 라우터나 코어가 새로운 패킷을 수신할 수 없는 상태가 되어 패킷이 네트워크에 적체되는 deadlock이 발생할 수 있다. 패킷 수신 속도에 비해 코어의 처리 속도가 느리거나 job distribution 과 job scheduling의 오류가 생길 경우 이러한 deadlock이 발생한다⁶⁾. 코어의 local 메모리나 라우터나 네트워크 인터페이스 내부에 존재하는 큐 또는 버퍼의 용량이 커지면 처리속도에 비해 빠르게 수신된 패킷을 저장할 수 있으므로 deadlock의 발생을 막을 수 있으나, 메모리의 용량은 시스템의 면적과 전력 소모에 직접적으로 연관되어 있으므로 크기에 제한을 둘 수밖에 없다⁶⁾.

Deadlock을 줄이기 위해 기존의 연구에서는 NoC 시스템 설계 시에 결정된 NoC 시스템의 네트워크 구조에서 사용할 라우팅 알고리즘을 선택하고, 선택된 알고리즘이 수행될 application에서 deadlock을 발생시키는지 검사한 후, deadlock 발생이 예상되는 구간의 라우터 저장 용량이나 채널의 대역폭을 증가시킨다¹²⁾. 하드웨어가 추가에 따른 면적 오버헤드가 발생하므로 면적 증가 없이 패킷 drop이 최소화되도록 각각의 라우터에 라우팅 정보를 실시간으로 업데이트 하여 패킷을 deadlock 이 발생된 경로가 아닌 우회 경로로 전달하여 deadlock을 최소화한다⁶⁾. 그러나 이러한 기존의 방법은 패킷이 이동하는 경로 수정을 위해 일정 시간마다 라우터는 자신의 라우팅 정보를 다른 라우터에 전달해야 하므로 이에 따른 패킷이 증가하며, 우회 경로는 패킷 전달 hop 수가 증가하여 패킷 전송 시간이 지연되

며, 라우팅 정보 업데이트 주기 내에서 발생된 deadlock은 처리할 수 없다는 단점이 존재한다. 이러한 단점을 해결하기 위해 추가적인 하드웨어를 최소화하고 패킷 전달 경로 증가를 최소화하기 위해 NoC 시스템에서 수행될 job을 application의 특성에 따라 트래픽이 적게 발생시키는 코어로 할당하고, 패킷 전달 경로가 긴 코어 간의 트래픽을 줄이는 방법에 대한 연구가 요구된다.

III. Deadlock 감소를 위한 동적 Job Distribution 알고리즘

NoC 시스템의 중요 응용분야인 멀티미디어 application인 MPEG4, MP3, GPS 위치 계산, OFDM 복조를 기존의 단순, 복합 구조 토폴로지에 적용하여 drop율을 조사하였다. Deadlock이 감소되면 패킷 drop이 현저히 줄어들게 되므로 drop율 비교를 통해 deadlock 감소를 확인할 수 있다. 1절에서는 application에 따라 drop율을 최소화 시키는 hybrid 구조의 configurable NoC 토폴로지를 제시하고, 2절에서는 제안된 NoC 시스템에서 사용된 네트워킹 주소체계를 제시한다. 3절에서는 설계된 NoC 구조에서 deadlock을 최소화시키는 동적 job distribution 알고리즘을 제안한다.

3.1 Hybrid 구조의 Configurable NoC 토폴로지

기존의 NoC 시스템의 네트워크 토폴로지는 버스, mesh, star, point-to-point 등의 단순 구조와 star-bus, star-mesh, star-star 등의 여러 단순 구조 네트워크가 혼용된 복합 구조 토폴로지 등이 제안되어 처리 속도, 면적, 전력 소모에 관한 비교가 이루어 졌다. 단순 구조 토폴로지는 모든 코어와 메모리가 하나의 네트워크 구조를 가지며 동일한 주소 체계와 동일한 프로토콜을 이용하여 데이터 통신을 수행한다⁸⁾. 복합 구조 토폴로지는 일부 코어들을 sub-cluster로 묶고, 각 sub-cluster 간을 연결하는 구조를 갖는다. Sub-cluster 내부와 sub-cluster 간의 연결은 다른 구조를 가질 수 있으며, 별도의 주소 체계와 별도의 프로토콜을 가질 수 있다⁸⁾. 이를 통해 sub-cluster 간의 역할을 분할할 수 있으며, sub-cluster 내부의 데이터 통신과 외부의 데이터 통신을 구분하여 트래픽이 많이 발생하는 코어들 간의 통신 제어를 통해 칩 전체의 속도 저하를 줄일 수 있다는 장점을 갖는다⁸⁾. 또한, sub-cluster 설계 방법에 따라 코어간의 네트워크 인터페이스를 다르

게 설계할 수 있으며, wiring에서 면적 감소의 효과를 기대할 수 있다. 그러나 기존에 연구된 NoC의 토폴로지들은 NoC가 적용된 시스템에서 수행되는 application을 고려하지 않아 다양한 application의 멀티태스킹이 필요한 최근의 요구를 효과적으로 충족하지 못하는 단점이 존재한다.

최근의 NoC 네트워크 구조는 공유 메모리나 이웃 코어의 접근성 등의 이유로 복합구조 토폴로지를 갖는다. 복합구조 토폴로지의 NoC 시스템은 sub-cluster 에서의 local 네트워크와 sub-cluster 간의 global 네트워크는 상이한 프로토콜을 사용할 수 있으나 각각의 sub-cluster는 sub-cluster 간 통신에서 프로토콜 변환 과정을 줄이기 위해 대부분 동일한 프로토콜을 사용한다. 프로토콜이 상이한 경우에는 global 네트워크를 사용하는 패킷이 증가할수록 프로토콜 변환 과정으로 인해 NoC 시스템의 성능은 감소하게 된다. 이를 최소화시키기 위해 job scheduling을 통해 global 네트워크에서의 트래픽을 최소화하면 트래픽의 locality factor는 0.8 이상으로 커지게 되어 global 네트워크 사용량이 감소되어 전체 시스템에 미치는 영향은 크지 않다⁸⁾.

기존의 연구에 따르면 임베디드 시스템에서 사용되는 application 들은 고유 특성에 따라 각기 다른 네트워크 토폴로지를 갖는 NoC 시스템 상에서 최적의 성능을 나타낸다¹⁰⁾. NoC의 네트워크 토폴로지가 모두 공통의 토폴로지를 가질 경우 application 별 패킷 drop을 최소화하는 토폴로지를 적용하지 못하고 시스템 전체적으로 패킷 drop이 적은 하나의 토폴로지만을 적용하게 되어 application에 효과적인 구조를 가질 수 없다. 따라서 각각의 sub-cluster 들은 수행될 application에 따라 패킷 drop이 최소가 되는 토폴로지를 갖게 하고, 프로토콜 변환 과정을 감소시키기 위해 sub-cluster간의 트래픽을 최소화시켜야 한다. 최근 sub-cluster 들의 네트워크 토폴로지를 수행될 application에서 패킷 drop이 최소화되는 구조를 실험적으로 탐색하여 NoC 시스템을 구성하는 hybrid NoC 시스템이 연구되었다¹²⁾. Hybrid NoC 시스템은 여러 개의 sub-cluster를 application에 따라 각기 다른 구조의 네트워크로 구성되고 각각의 sub-cluster를 star 또는 mesh 형태로 구성된다. 임베디드 시스템에서 수행될 application은 NoC 시스템 설계시 모두 예측할 수 없으며 종류도 많기 때문에 모든 application에 대해 최적의 hybrid NoC 시스템을 칩으로 구현해 두기는 사실상 불가능하여 스위치에 대해 프로그래

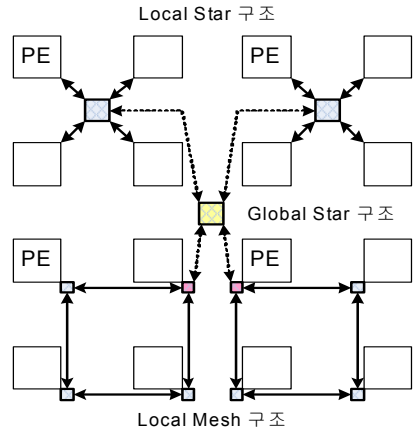


그림 3. 제안된 Configurable Hybrid NoC 시스템의 네트워크 토폴로지

밍을 통해 토폴로지를 결정할 수 있는 configurable 스위치를 사용한다¹¹⁾. 그림 3은 본 연구에서 설계한 configurable hybrid NoC 시스템의 네트워크 토폴로지를 보인다. 그림 3의 시스템은 sub-cluster의 구조로는 2개의 mesh와 2개의 star 구조를 갖고 global 네트워크는 star 로 구성되었다. 각 sub-cluster들이 서로 다른 토폴로지와 주소체계, 그리고 프로토콜을 사용하게 되므로 프로토콜 변환 과정을 최소화시켜 패킷 drop을 최소화시키는 알고리즘이 필요하며 이를 위한 알고리즘은 3.3절에 제시하였다.

3.2 네트워킹 주소 체계

구현된 NoC 시스템에서는 두 가지의 주소 체계를 사용한다. 16개 코어에 대한 4bit의 주소를 사용하지 않고 cluster 내부에서 사용하는 local 주소 2비트씩을 사용하며, 각각의 cluster는 2비트의 주소를 별개로 가지고 있다. 외부로 패킷을 보내거나 타 cluster에 패킷을 전달할 경우에는 각 cluster의 게이트웨이로 동작하는 라우터에서 해당하는 cluster의 주소를 추가하여 외부로 전송하게 된다. 이는 인터넷의 IPv4에서 부족한 주소를 최대한 활용하기 위한 방법으로 사용하는 NAT(Network Address Translation)방식과 비슷한 개념을 가지고 있다. 즉, 각 cluster 내부 통신에서는 2비트의 내부 주소를 사용하고 외부로 패킷을 전송할 때만 총 5비트의 주소로 번역해서 사용하는 방식이다.

그림 4는 본 연구에서 사용한 패킷의 헤더를 보인다. OFLAG는 패킷의 목적지가 cluster 내부인지 외부인지를 구별하는데 사용되는 flag 비트이다, 그림 4(a)는 global 네트워크에서 사용되는 패킷 헤더

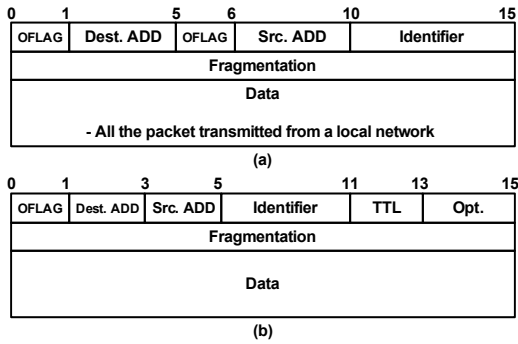


그림 4. 제안된 패킷 헤더
(a) Globa-net 용 패킷 헤더, (b) Local-net 용 패킷 헤더

로 4비트의 목적지 주소, 전송지 주소가 포함된다. 각 cluster의 스위치에서 OFLAG가 1로 세팅된 패킷은 sub-cluster 내로 보내지 않고 job distributor로 전달한다. Job distributor는 sub-cluster에서 패킷을 전달받으면 OFLAG를 뒤의 목적지 주소 4 비트를 확인하고 global 네트워크에서 사용하는 주소 헤더를 제거한 후 sub-cluster의 스위치로 전달한다. 목적지의 sub-cluster의 스위치는 OFLAG를 0으로 리셋하고 목적지 코어의 주소를 확인한 후 목적지로 전송한다. 6 비트의 identifier는 전송되는 패킷을 구별하는데 사용되고, fragmentation 필드는 하나의 데이터가 패킷 크기에 따라 나뉠 때 몇 번째 조각 인지를 구별하는데 사용된다. 그림 4(b)는 local 네트워크에서 사용하는 패킷 헤더로 2 비트 TTL(Time-to- Live)에 기입된 수 이상의 hop을 거치게 되면 패킷을 자동 삭제하게 된다. 이는 무한 루프에 빠지는 whitelock을 예방하기 위한 필드이다. 각 스위치나 라우터는 패킷을 전달하면서 TTL의 수를 1만큼 차감하여 기입한다. 추가 기능을 위해 3 비트의 옵션 필드가 지원된다.

3.3 동적 Job Distribution 알고리즘

임베디드 시스템에서 수행되는 멀티미디어 application에 따라 패킷 drop이 최소가 되는 sub-cluster의 네트워크 토폴로지가 결정되고 나면, 해당 application이 최적으로 수행되는 구조의 sub-cluster에서만 수행될 수 있도록 해야한다. 패킷 drop이 최소가 되는 sub-cluster에서 수행되어 네트워크 부하를 줄이게 되고, 타 sub-cluster로의 패킷 전송을 줄여 프로토콜 변환에 따른 오버헤드를 무시할 수 있기 때문에 반드시 각각의 멀티미디어 application은 자신에게 최적인 sub-cluster 내에서 수행되어야 한다. 이를 위해 본 연구에서는 최적

sub-cluster로의 job 매핑을 위해 job distribution 알고리즘을 제안하였다.

Job distributor는 수행될 각각의 멀티미디어 application과 해당 application이 최적으로 수행되는 sub-cluster의 주소를 연결한 lookup table을 가지고 있다. 수행될 job은 자신이 포함된 application을 나타내는 비트를 가지고 있으므로 job distributor는 입력된 job에 대해 해당 비트를 확인한 후 lookup에 따라 해당 sub-cluster로 할당하게 된다. 즉, 하나의 동영상 파일은 단 하나의 sub-cluster로만 할당되기 때문에 application 수행에 따른 데이터 전달이 sub-cluster 내에서만 이루어지며, global 메모리나 외부 입출력을 위한 job 수행시에만 global 네트워크를 사용하게 된다. 따라서 global 네트워크 사용이 감소하기 때문에 네트워크 트래픽의 locality factor가 커짐으로 인해 트래픽은 시스템 전체로 분산되며, deadlock 발생하더라도 전체 네트워크 시스템에 영향을 미치지 않고 sub-cluster 내에서만 영향을 미치게 되어 시스템 성능 저하를 최소화시킬 수 있다. 수행될 job이 job distributor로 입력되면, lookup table 참조를 통해 해당 application에 패킷 drop율이 최소가 되는 후보 sub-cluster들을 선택하고 이 중 idle 상태의 sub-cluster를 탐색하여 job을 할당하게 된다. Lookup table에 의해 제공된 모든 sub-cluster가 idle 상태가 아닌 경우, 나머지 sub-cluster에 대해 idle 상태인 sub-cluster를 찾아 job을 할당한다. 모든 sub-cluster가 idle 상태가 아닌 경우 lookup table에서 제공된 sub-cluster부터 나머지 sub-cluster 순으로 해당 sub-cluster의 버퍼 점유율이 70%가 넘지 않는 sub-cluster를 탐색하여 job을 할당하되 모든 sub-cluster의 스위치 버퍼가 70% 이상 점유되고 있을 경우에는 1번 sub-cluster에서부터 round-robin 방식을 이용하여 job을 할당한다.

IV. 실험 결과

본 연구의 NoC 시스템은 Verilog HDL로 설계되었다. 실험에 사용된 영상 데이터는 720*480 해상도의 MPEG4 코덱으로 인코딩되었으며, 음성 데이터는 128 Kbps 수준으로 압축된 MPEG1 audio layer-3 코덱으로 인코딩되었다. GPS 위치 데이터는 4채널 방식의 송신기를 기준으로 프레임당 1.5 Kb/sec로 50 bit/sec 전송율을 기준으로 설계하였다. OFDM 복조는 MPEG4에서 사용한 데이터를 미리 OFDM 방식으로 변조 후 사용하였다. 종류별로 각

표 1. 기존 복합구조에서 제안된 알고리즘과 기존 알고리즘 패킷 drop을 비교

알고리즘 Appl.	복합구조 패킷 drop율(%) (G-Star/L-Mesh)						
	미적용		APSRA		XY알고리즘		제안된 구조
	패킷 drop율	drop 감소율	패킷 drop율	drop 감소율	패킷 drop율	drop 감소율	drop 감소율
MPEG4	21.4	40.2%	14.1	9.2%	14.7	12.9%	12.8
MP3	18.2	54.4%	10.8	23.1%	13.5	38.5%	8.3
GPS	20.7	56.5%	11.3	20.4%	13.9	35.3%	9.0
OFDM	21.1	53.1%	11.5	13.9%	12.8	22.7%	9.9
평균	20.3	50.7%	11.9	16.0%	13.7	27.0%	10.0

10개씩의 데이터에 대해 반복적으로 실험한 결과의 평균을 실험 결과로 사용하였다. NoC 시스템에서 일반적으로 패킷 drop이 발생했을 때 송신측에 ack를 보내어 재전송을 요구하는 프로토콜을 사용하나, 본 연구에서는 패킷 drop 발생을 확인하기 위해 재전송을 허용하지 않고 출력된 데이터를 입력 데이터로 변형시킨 후 drop된 패킷의 양을 확인하는 방식을 이용하였다. 여러 가지 네트워크 토폴로지를 가진 NoC 시스템에 기존 연구의 deadlock 감소 알고리즘과 본 연구에서 제안된 알고리즘과의 deadlock 감소 효과를 패킷 drop 감소를 통해 비교한 결과를 표 1과 표 2에 제시한다. 표 1은 각각의 sub-cluster들이 공통된 mesh 구조를 가지며, global 네트워크가 star 구조를 가지는 기존의 복합구조에서 기존의 deadlock 감소 알고리즘과 제안된 알고리즘과의 deadlock 감소와 패킷 drop율을 보인다.

제안된 알고리즘을 검증하기 위한 NoC 시스템의 복합구조로 global-star/ local-mesh 구조를 채택하였다. 실험결과에서 기존 복합구조인 global-star/ local-mesh 구조에서는 제안된 job distribution 알고리즘을 적용하지 않은 경우 패킷 drop율은 평균 20.3% 였으며, APSRA[6] 알고리즘이 적용된 경우의 패킷 drop율은 11.9% 였다. 또한 XY[13] 알고리즘이 적용된 경우에는 평균 패킷 drop율이 13.7% 로 측정되었다. 제안된 job distribution 알고리즘은 global-star/ local-mesh 복합 구조에서 기존 알고리즘에 비해 최대 27.0%의 패킷 drop이 감소되었다. 표 2는 네트워크 토폴로지에 따라 패킷 drop율이 감소하는 구조로 sub-cluster를 구성한 configurable hybrid 구조와 제안된 job distribution 알고리즘을 적용하여

표 2. Configurable Hybrid 구조에서 제안된 알고리즘과 기존 알고리즘 패킷 drop을 비교

알고리즘 Appl.	Configurable Hybrid 구조 패킷 drop율(%) Global Star/Local (2-mesh, 2-star)						
	미적용		APSRA		XY알고리즘		제안된 구조
	패킷 drop율	drop 감소율	패킷 drop율	drop 감소율	패킷 drop율	drop 감소율	drop 감소율
MPEG4	19.6	57.7%	9.2	9.8%	10.1	17.8%	8.3
MP3	17.3	67.1%	7.1	19.7%	7.3	21.9%	5.7
GPS	18.2	66.5%	7.8	21.8%	8.1	24.7%	6.1
OFDM	18.8	65.4%	6.8	4.4%	9.6	32.3%	6.5
평균	18.5	63.8%	7.7	13.0%	8.8	23.9%	6.7

표 3. 제안된 알고리즘 적용에 따른 면적 비교

	제안된 알고리즘	알고리즘 미적용	APSRA	XY
면적 (gates)	509,315.8	482,162.2	495,917.6	494,534.5
면적 증가율	-	5.6 %	2.7 %	3.0 %

(2-input NAND gate = 1)

기존의 deadlock 감소 알고리즘과의 비교 결과를 보인다.

Configurable hybrid 구조에는 application의 패킷 drop 특징에 따라 4개의 sub-cluster를 2개씩 star와 mesh 구조를 갖도록 하였다. 제안된 job distribution 알고리즘에 따라, star 구조에서 GPS 위치 계산과 OFDM 복조를, mesh 구조에서는 MPEG4와 MP3를 재생을 동시에 수행하였다. 멀티미디어 application에 따라 패킷 drop이 최고가 되는 네트워크 토폴로지를 갖도록 하는 configurable hybrid 구조에서는 제안된 job distribution 알고리즘이 적용되지 않은 경우 평균 18.5%, APSRA가 적용된 경우 평균 7.7%, XY가 적용된 경우에는 평균 8.8%의 패킷 drop율을 보였다. Configurable hybrid 구조에서는 평균 패킷 drop율이 최대 23.9% 감소하였다. Deadlock 감소 기법이 적용되지 않은 경우에 대해서는 50.7% 이상 패킷 drop을 감소시켰다.

본 연구에서 제안된 job distribution 알고리즘에 따라 설계된 NoC 시스템과 기존 알고리즘으로 설계된 시스템의 면적 비교 결과를 표 3에 보인다. 각각의 시스템은 Verilog HDL로 구현되었으며, Synopsys사의 Design Analyzer 를 사용하여 하이닉

스의 0.35 μ m 공정 라이브러리로 면적을 측정하였다. 또한 측정 단위는 2-input nand 크기를 1로 하는 게이트의 수로 지정하였다.

본 연구에서 설계한 hybrid 구조의 configurable NoC 토폴로지를 사용하고 각각의 deadlock 감소를 위한 여러 알고리즘을 적용하였다. 제안된 알고리즘이 적용된 NoC 시스템은 509,315.8 게이트, job distributor가 포함되지 않은 시스템은 482,162.2 게이트로 5.6 % 면적 증가를 보였다. 또한 APSRA가 적용된 경우는 495,917.6 게이트로 이에 비해 2.7 % 증가하였으며, XY가 적용된 경우는 530,488.5 게이트로 3.0 % 수준으로 면적이 증가하였다.

V. 결 론

본 연구에서 제안한 job distribution 알고리즘과 멀티미디어 application에 대한 configurable hybrid NoC 토폴로지를 적용하여 구현한 NoC 시스템을 패킷 drop율을 통해 기존의 알고리즘과 deadlock 감소를 비교 검증하였다. 대상 application을 최근 컴퓨터나 무선 임베디드 시스템에서 가장 많이 사용되는 멀티미디어 application인 MPEG4, MP3, GPS 위치 계산, OFDM 복조로 정하고 해당 application에 hybrid NoC 토폴로지와 job distribution을 적용할 경우 job distribution을 적용하지 않은 시스템에 비해 패킷 drop율이 평균 63.8% 감소되었다. 또한 configurable hybrid 구조를 적용하고 APSRA 알고리즘을 적용한 시스템에 비해 패킷 drop율이 평균 13.0%, XY 알고리즘을 적용한 시스템에 비해 평균 23.9% 감소되었다. 이는 NoC 시스템을 멀티미디어 application에 패킷 drop이 최소가 되는 네트워크 구조를 가진 상태에서 job distribution을 적용할 경우 시스템 전체에서 발생하는 패킷 drop이 최소화되고, 이는 deadlock 발생이 최소화됨을 의미한다. 또한 job distributor의 면적 증가량은 APSRA 알고리즘을 적용한 경우에 비해 면적이 2.7 % 증가하였으며, XY 알고리즘에 비해 3.0 % 증가하여 전체 시스템의 면적에 비추어볼 때 미미한 수준이다. 멀티미디어 application에 따라 최소 패킷 drop율을 갖는 sub-cluster를 구성하고 해당 sub-cluster에 job을 할당하며, 이후 입력되는 job에 대해 sub-cluster의 트래픽 수준에 따른 동적 job distribution을 통해 deadlock을 현저히 감소시킬 수 있다.

참 고 문 헌

- [1] 조재범, 유용호, 황선영, "임베디드 프로세서 코어 자동생성 시스템의 구축", 한국통신학회논문지, 제30권 6A호, pp.526-534, 2005년 6월.
- [2] W. Dally and B. Towles, "Route Packets, not Wires: On-Chip Interconnection Networks," in Proc. Design Automation Conf., Las Vegas, NV. pp.684-689, Nov. 2001.
- [3] L. Benini and G. Micheli, "Networks on Chips: A New SoC Paradigm," IEEE Computer, Vol.35, No.1, pp.70-78, Jan. 2002.
- [4] W. Dally and C. Seitz, "Deadlock-Free Message Routing in Multiprocessor Inter-connection Networks," IEEE Trans. on Computers, Vol. 36, No.5, pp.547-553, May 1987.
- [5] G. Micheli and L. Benini, Technology and Tools: Networks On Chips, Morgan Kaufmann, 2006.
- [6] M. Palesi, R. Holmark, and S. Kumar, "A Methodology for Design of Application Specific Deadlock-Free Routing Algorithms for NoC Systems," in Proc. 4th CODES+ISSS, Seoul, Korea, pp.142-147, Oct. 2006.
- [7] A. Adriahtenaina, H. Charlery, A. Greiner, L. Moritz, and C. Zeferino, "SPIN: A Scalable-Packet Switched On-Chip Micro-Network, in Proc. DATE '03, Messe Munich, Germany, pp.70-73, Mar. 2003.
- [8] A. Jouraku, M. Koibuchi, and H. Amano, "An Effective Design of Deadlock-Free Routing Algorithms Based on 2D Turn Model for Irregular Networks," IEEE Trans. on Parallel and Distributed Systems, Vol.18, No.3, pp.320-333, Mar. 2007.
- [9] W. Zhou, Y. Zhang, and Z. Mao, "An Application Specific NoC Mapping for Optimized Delay," in Proc. DTIS '06, Gammarth, Tunisia, pp.184-188, Sep. 2006.
- [10] L. Benini, "Application Specific NoC Design," in Proc. DATE '06, Munich, Germany, pp.491-495, Mar. 2006.
- [11] C. Hilton and B. Nelson, "A Flexible Circuit-switched NoC FPGA-Based Systems," in Proc. FPLA '05, Tucson, AZ, pp.191-196, Aug. 2005.
- [12] S. Bourduas and Z. Zilic, "A Hybrid Ring/Mesh

Interconnect for Networ-on-Chip Using Hierarchical Rings for Global Routing,” in Proc. NOCS 07’, Princeton, NJ, pp.195-204, May 2007.

- [13] E. Bolotin, A. Morgenshtein, I. Cidon, and A. Kolodny, “Automatic and Hardware- Efficient SoC Integration by QoS Network on Chip,” in Proc. 11th ICECS, Tel Aviv, Israel, pp.479-482, Dec. 2004.

김 우 주 (Woo-Joo Kim)

정회원



2001년 2월 서강대학교 전자공학과 졸업

2003년 2월 서강대학교 전자공학과 석사

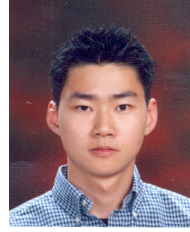
2003년 3월~현재 서강대학교 전자공학과 박사과정

<관심분야> NoC/SoC 시스템 설

계, 임베디드 시스템 설계, 네트워크 알고리즘

이 성 희 (Sung-Hee Lee)

정회원



2006년 2월 서강대학교 전자공학과 졸업

2008년 2월 서강대학교 전자공학과 석사

2008년 3월~현재 삼성전자 DS 총괄 메모리 SSD/CARD팀 연구원

<관심분야> NoC/SoC 시스템 설계, CAD, 네트워크 스위치 설계

김 우 주 (Woo-Joo Kim)

정회원



2001년 2월 서강대학교 전자공학과 졸업

2003년 2월 서강대학교 전자공학과 석사

2003년 3월~현재 서강대학교 전자공학과 박사과정

<관심분야> NoC/SoC 시스템 설

계, 임베디드 시스템 설계, 네트워크 알고리즘

황 선 영 (Sun-Young Hwang)

정회원



1976년 2월 서울대학교 전자공학과 졸업

1978년 2월 한국 과학원 전기 및 전자공학과 공학석사 취득

1986년 10월 미국 Stanford 대학 전자공학 박사학위 취득

1976~1981 삼성 반도체 주식회

사 연구원 팀장

1986~1989 Stanford 대학 Center 랙 Integrated System 연구소 책임 연구원

1989~1992 삼성전자(주) 반도체 기술 자문

2002년 4월~2004년 2월 서강대학교 정보통신대학 원장

1989년 3월~현재 서강대학교 전자공학과 교수

<관심분야> CAD 및 임베디드 시스템 설계, NoC/ SoC 시스템 설계, HW/SW co-design, DSP/VLSI 시스템 설계