

불균형한 입력을 효과적으로 처리하는 유연한 혼합형 라우터 설계

정라미, 김성천
서강대학교 컴퓨터학과

Design of Flexible Hybrid Router to Process Unbalanced Input Effectively

Rami Jung, Sungchun Kim
Dept. of Computer Science, Sogang University

요 약

라우터의 기본적인 목적은 안정적으로 다량의 데이터를 전송하는 것이다. 현재 다양한 메시지를 효과적으로 처리하기 위한 여러 혼합형 라우터가 개발되고 있다. 이는 단순히 한가지 방식만 고수하는 것이 아니라 기존의 여러 기법을 혼합된 방식을 적용하는 것이다. 이러한 혼합형 기법은 기존의 단일 방식의 단점을 보완할 수 있어야 하며, 그에 따른 오버헤드를 감수할 수 있어야 한다. 이러한 목적으로 웜홀 스위칭(wormhole switching)과 파이프라인드 서킷 스위칭(Pipelined Circuit Switching)을 동시에 구현하기 위해 혼합형 라우터 구조가 제안되었다. 이 라우터는 두 스위칭 기법을 동시에 지원하여 다양한 메시지를 효과적으로 처리할 수 있는 특성이 있다. 그러나 이 구조는 각 스위칭 방식에 해당하는 내부 연결망을 독립적으로 구성함으로써 입력으로 들어오는 스위칭 비율이 불균형일 때 내부 자원을 효율적으로 사용할 수 없는 단점이 있다. 따라서 본 논문에서는 라우터의 내부 연결망을 공유하여 사용하는 새로운 혼합형 라우터를 제안하였다. 제안한 구조는 웜홀과 파이프라인드 서킷 스위칭을 지원하는 라우터로, 메시지를 전송할 때 내부 연결망을 서로 공유함으로써, 입력 메시지의 비율이 불균형할 때 효과적으로 자원을 이용할 수 있게 하였다. 시뮬레이션을 통하여 기존의 혼합형 라우터를 사용하는 것보다 더 높은 성능을 보인다는 것을 증명하였다.

제 1 장 서 론

현재 다양한 메시지를 효과적으로 처리하기 위한 여러 혼합형 라우터가 개발되고 있다. 이는 단순히 한가지 방식만 고수하는 것이 아니라 기존의 여러 기법을 혼합된 방식을 적용하는 것이다. 이러한 개념으로 기존에 제안된 하이브리드 라우터는 웜홀 스위칭과, 파이프라인드 서킷 스위칭[1]을 동시에 지원하는 혼합형 라우터이다.[2] 이 라우터는 웜홀 스위칭 하는 부분과 PCS 방식으로 하는 부분으로 이루어진 라우터로 해당 스위칭 부분을 분리하여 각 채널에 연결하여 메시지를 전송하는 혼합형 라우터이다.

이 하이브리드 라우터는 스위칭 방식에 따라 분리된 내부 연결망을 사용함으로써 두 가지 방식의 메시지 전송이 발생할 때 적절히 대처할 수 있게 하였다. 이는 또한 동시에 두 가지 방식의 메시지 전송도 가능하다는 장점이 있다. 그러나 이 라우터는 한 가지 방식의 메시지 전송 부하량이 증가해도 다른 방식의 내부 연결망을 사용하지 못한다는 단점이 존재한다.

이러한 내부 연결망의 효율적 사용을 위해 새로운 라우터 구조를 제안한다.

이 라우터는 웜홀과 PCS로 데이터를 전송할 때 내부 연결망을 서로 공유함으로써 로드의 불균형 발생시 효과적으로 자원을 이용할 수 있다. 이러한 유연한 라우터를 F-router라 정의한다.

기존의 방식이 각 스위칭 방식에 따라 고유한 스위치를 이용하는 데 반해, F-router는 스위치 내부 연결망을 서로 공유함으로써 스위칭 입력 로드의 불균형 발생시 효과적으로 자원을 이용할 수 있다. 파이프라인드 서킷 스위칭 방식은 웜홀 방식의 변형이므로 추가적인 보정 과정을 거치면 공유될 수 있는 가능성은 크다. 부가적인 연산 시간을 고려한 환경에서 제안된 라우터는 기존의 하이브리드 라우터보다 평균적으로 24%의 성능 향상을 보인다.

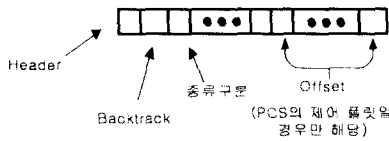
제 2 장 플렉서블 라우터(F-router)

본 논문에서 제안하는 F-router는 웜홀과 PCS로 데이터를 전송할 때 내부 연결망을 서로 공유함으로써 로드의 불균형이 발생시 효과적으로 자원을 이용할 수 있다. 이러한 F-router의 특징은 이미 설정된 경로의 재사용으로 인한 캐쉬 효과, 유연한 메시지 전송, 라우터의 크기 절감 효과, 내부 스위치 망의 사용 효율 증가로 설명할 수 있

다.

전송 메시지 형식

전송되는 플릿의 형식은 [그림 1]과 같이 구성된다



[그림 1] F-router 플릿 형식

제어 플릿은 일반적인 헤더 플릿, 인식 플릿, 그리고 경로상 문제가 발생했을 때 생성하는 백-트랙(back-track) 플릿, 이렇게 세 가지 경우가 존재한다. 이 세 가지 플릿을 구별하기 위해 첫 2 비트를 검사하여, 10으로 시작하면 헤더 플릿, 01로 시작하면 인식 플릿, 11로 시작하면 백-트랙 플릿으로 구분한다.

그런데 제안하는 방식에서는 보내는 플릿이 워홀 방식의 제어 플릿인지 서킷방식의 플릿인지 구분하는 비트가 더 필요하다. 이를 본 논문에서 C라 정의하였다. C가 0 이면 워홀 스위칭 방식이고, 1이면 PCS 방식이다. 또한 X라 표시된 부분은 어떤 값을 취하여도 된다는 것(don't care)을 의미한다.

그리고 Offset 부분은 PCS방식에서만 사용하는 것으로 설정된 경로에 문제가 발생했을 경우, 최대 재설정 횟수를 정의한 것이다 이와 같은 정의를 정리하면 [표 1]과 같이 표현될 수 있다.

플릿 종류		형식
데이터 플릿		00X...XXXX
제어 플릿	헤더 플릿	10CX...XXX
	ACK 플릿	01CX...XXX
	back-track 플릿	111X...XXX

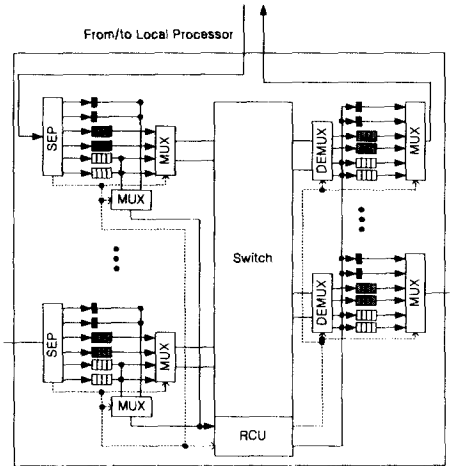
[표 1] 플릿의 종류와 형식

F-router의 구조

제안하려는 라우터의 구조는 [그림 2]와 같이 표현된다.

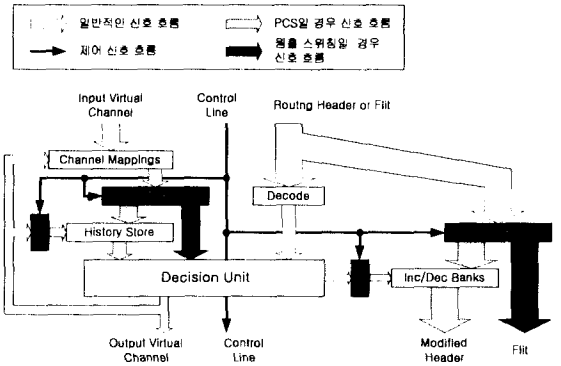
음영 처리된 버퍼는 PCS를 위한 데이터 버퍼, 검은 버퍼는 PCS를 위한 제어 플릿 버퍼, 흰색 처리된 버퍼는 워홀 스위칭을 위한 버퍼이다. 각 버퍼는 플릿 단위의 메시지를 저장할 수 있다.

실선의 화살표는 메시지의 이동을 표시하고 점선은 제어 신호의 이동을 의미한다. 입력으로 들어오는 제어 플릿은 SEP에 의해 스위칭 형식이 구별되어 해당 버퍼로 들어간다. SEP는 기존의 LCU의 기능에 스위칭 방식 구분의 기능도 포함시킨 것이다. 또한 SEP에 의해 구분된 제어 신호는 지속적으로 여러 MUX와 RCU에 영향을 주어 반복적인 계산으로 인한 추가적인 지연이 발생하지 않도록 한다. RCU에 의해 설정된 경로를 통하여 메시지를 전송하고, 입력부와 유사한 방식으로 출력부를 제어하여 메시지를 출력한다. 또한 출력부의 MUX는 LCU의 기능에 두 스위칭 방식을 구분하는 기능을 추가한 것이다.



[그림 2] F-router 구조

[그림 3]은 [그림 2]에 표시된 RCU의 블록 다이어그램이다. F-router의 RCU는 기존의 워홀 스위칭 방식이나 PCS가 두 가지 입력을 필요로 하는 것 외에 추가적으로 SEP에서 연산한 스위칭 구분 신호가 요구된다. 경로 설정 알고리즘은 기본적으로 워홀과 PCS의 동일한 경로 설정 알고리즘을 사용하며, 두 스위칭 방식의 상이한 연산은 MUX와 스위치를 사용하여 구분하였다. 이로 인해 각 경우에 필요한 정보를 전송하도록 하였다.



[그림 3] F-router의 RCU

제 3 장 성능 평가

본 논문에서는 Scientific and Engineering Software Inc.의 SES/Workbench Release 3.0을 사용하여 확률적 이산 사건 모델링으로 시뮬레이션을 수행하였다.[3]

통신망에서 가장 중요한 성능 평가 기준으로는 처리율과 평균 지연 시간이 있다. 이 두 기준으로 기존의 16×16 하이브리드 라우터(H-router)와 16×16 F-router를 비교하였고 변화되는 인자로는 작

업로드와 스위칭 비율로 설정하였다.

시뮬레이션을 위해 설정값을 [표 2]로 정리하였고 결과는 [표 3]부터 [표 7]로 산출되었다.

종류	설정값
하이브리드 라우터의 내부 크로스 바 지연 시간 (16×16 스위치일 경우)	0.72 ns
F-router의 내부 크로스 바 지연 시간 (32×32 스위치)	1.3 ns
내부 흐름 제어 시간	2.2 ns
출력 부분의 셋업 시간	0.8 ns
웜홀 스위칭의 플릿당 제어 시간	4.9 ns/flit
PCS 스위칭의 플릿당 제어 시간	3.1 ns/flit
F-router의 내부 흐름 제어 시간	2.4 ns

[표 2] 시뮬레이션을 위한 설정 값

작업로드	0.2	0.4	0.6	0.8	1	
처리율	H-Router	15.01	11.95	4.04	1.67	0.63
	F-Router	50	35.2	25.33	22.26	20.15
지연시간	H-Router	22	26.6	28.8	26.6	26.3
	F-Router	21.8	25.5	25	22	22.7

[표 3] 100 % PCS일 때 처리율(%)과 지연시간(ns)

작업로드	0.2	0.4	0.6	0.8	1	
처리율	H-Router	84.63	50.91	25.96	24.47	21.92
	F-Router	72.71	55.71	39.32	33.02	27.95
지연시간	H-Router	10.6	16.5	9.6	13.2	12.3
	F-Router	14.2	19	43.9	24.5	24

[표 4] 75 % PCS일 때 처리율(%)과 지연시간(ns)

작업로드	0.2	0.4	0.6	0.8	1	
처리율	H-Router	95.13	69.28	52.51	36.4	28.96
	F-Router	87.84	54.45	45.29	35.25	31.19
지연시간	H-Router	5.1	27.3	22.1	19.9	30.2
	F-Router	12.1	15.1	22.6	25.8	26.2

[표 5] 50 % PCS일 때 처리율(%)과 지연시간(ns)

작업로드	0.2	0.4	0.6	0.8	1	
처리율	H-Router	90.19	67.1	47.28	36.94	29.86
	F-Router	79.72	58.25	40.77	34.27	30.96
지연시간	H-Router	10.1	20.2	28.5	29.5	29.8
	F-Router	13.9	21.5	24.1	24.9	25.4

[표 6] 25 % PCS일 때 처리율(%)과 지연시간(ns)

작업로드	0.2	0.4	0.6	0.8	1	
처리율	H-Router	67.06	41.09	27.59	21.46	16.85
	F-Router	67.22	39.89	27.15	20.58	16.5
지연시간	H-Router	17.6	32.08	33.04	33.08	33.12
	F-Router	18.5	33.07	33.86	33.9	33.98

[표 7] 0 % PCS일 때 처리율(%)과 지연시간(ns)

제 5 장 결 론

시뮬레이션한 성능 평가 및 분석을 통하여 볼 때, F-router가 하이브리드 라우터보다 평균 10%의 처리율의 감소에서 63%의 처리율을 보이며, 전체 평균적으로는 24% 처리 향상을, PCS비율이 25%이하이거나 75%이상인 경우에는 34%성능 향상을 보였다. 또한 평균 지연 시간의 경우 F-router는 하이브리드 라우터에 비해 평균 7% 시간 감소에서 34%시간 증가를 가져오며 전체적인 평균 9% 평균 지연 시간의 증가를 초래하였다. 또한 PCS비율이 25%이하이거나 75%이상인 경우에는 평균 12% 시간 증가를 보였다.

입력으로 들어오는 메시지의 비율 차가 클 때 제한한 라우터의 성능이 기존의 하이브리드 라우터보다 증가하는 것을 알 수 있었다. 그리고 서킷 스위칭 메시지와, 웜홀 스위칭 메시지 량이 비슷할 경우, 라우터의 제어와 전송의 오버헤드로 하이브리드 라우터에 비해 처리율이 다소 떨어지는 현상을 보였다.

그러므로 웜홀과 서킷 스위칭 비율의 불균형이 심한 조건일 때 본 논문에서 제안하는 라우터가 유용하게 사용될 수 있다.

참 고 문 헌

- [1] James D. Allen, Patrick T. Gaughan, David E. Schimmel, and S. Yalamanchili, "Ariadne - an adaptive router for fault-tolerant mulicomputers," Proceedings 23st Int'l Symp. on Computer Architecture, pp. 278-288, April. 1994.
- [2] J. Duato, P. López, F. Silla and S. Yalamanchili, "A high performance router Architecture for interconnection networks," 1996 International Conference on Parallel Processing, pp. I-61 - I-68, 1996.
- [3] SES/Workbench Rel. 3.0, Scientific and Engineering Software, Inc., 1995