



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2011-0058575
(43) 공개일자 2011년06월01일

- | | |
|--|---|
| <p>(51) Int. Cl.
 <i>G06F 1/08</i> (2006.01) <i>G11C 7/22</i> (2006.01)
 <i>G06F 1/32</i> (2006.01) <i>G06F 13/38</i> (2006.01)</p> <p>(21) 출원번호 10-2009-0115414
 (22) 출원일자 2009년11월26일
 심사청구일자 없음</p> | <p>(71) 출원인
 삼성전자주식회사
 경기도 수원시 영통구 매탄동 416</p> <p>(72) 발명자
 윤재근
 경기 화성시 진안동 진안골마을주공10단지 1006동 301호
 정현욱
 경기 화성시 반송동 시범한빛마을한화꿈에그린아파트 236동 1003호
 (뒷면에 계속)</p> <p>(74) 대리인
 송윤호, 오세준, 권혁수</p> |
|--|---|

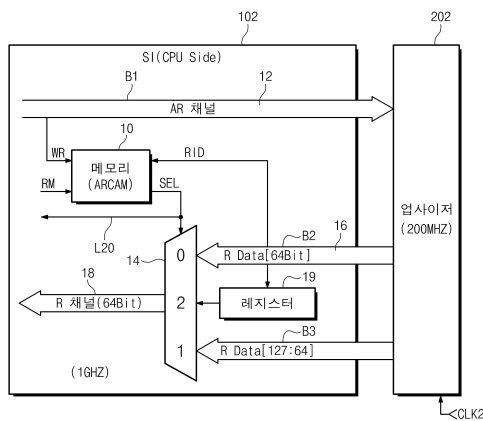
전체 청구항 수 : 총 10 항

(54) 데이터 프로세싱 시스템에서의 대역폭 동기화 회로 및 그에 따른 대역폭 동기화 방법

(57) 요약

데이터 프로세싱 시스템에서의 대역폭 동기화 회로 및 그에 따른 대역폭 동기화 방법이 개시된다. 그러한 대역폭 동기화 회로는, 업사이저와 리퀘스트 선택 공급부를 실시 예로서 구비한다. 업사이저는 제1 클럭보다 주파수가 낮은 제2 클럭에 응답하여 동작하며 미리 설정된 리퀘스트에서는 제1 데이터 비트폭 보다 큰 제2 데이터 비트폭으로 리드 데이터를 출력하는 싱크 언팩커를 포함한다. 또한, 리퀘스트 선택 공급부는, 상기 제1 데이터 비트 폭으로 리드 데이터가 입력되는 일반 리퀘스트에서는 상기 입력된 제1 데이터 비트 폭의 리드 데이터를 바이패스하고, 상기 미리 설정된 리퀘스트에서는 상기 제2 데이터 비트 폭으로 입력되는 상기 리드 데이터를 상기 제1 클럭에 응답하여 공급한다. 본 발명의 실시 예에 따르면, 대역폭 보틀넥을 해소하여 시스템 퍼포먼스가 개선된다.

대표도 - 도7



(72) 발명자

엄준형

서울 서초구 서초동 메이플라워 멤버스빌 803호

심성훈

경기 성남시 분당구 서현동 효자촌삼환아파트 501
동 1401호

홍성민

서울 구로구 고척동 98-93 (10/2) 성원빌라 202

정법철

경기 용인시 기흥구 농서동 산24번지 난초동319호

특허청구의 범위

청구항 1

프로세서 클럭에 따라 동작하는 싱크 팩커와 싱크 언팩커로 구성된 업사이저와;

상기 업사이저와 연결되며 상기 프로세서 클럭보다 낮은 주파수의 버스 클럭에 응답하여 상기 업사이저의 데이터에 대한 싱크 다운을 수행하는 싱크 다운부를 구비함을 특징으로 하는 데이터 프로세싱 시스템에서의 대역폭 동기화 회로.

청구항 2

제1항에 있어서, 상기 프로세서 클럭이 약 1GHz의 주파수를 가질 경우에 상기 버스 클럭은 약 200MHz의 주파수를 가짐을 특징으로 하는 데이터 프로세싱 시스템에서의 대역폭 동기화 회로.

청구항 3

제2항에 있어서, 상기 싱크 팩커는 라이트 어드레스 채널, 라이트 데이터 채널, 및 라이트 응답 채널에 싱크 팩킹을 수행하고, 상기 싱크 언팩커는 리드 어드레스 채널과 리드 데이터 채널에 대한 싱크 언팩킹을 수행함을 특징으로 하는 데이터 프로세싱 시스템에서의 대역폭 동기화 회로.

청구항 4

서로 다른 주파수의 제1,2 클럭에 각기 응답하여 동작하는 제1,2 싱크 팩커와;

상기 제1,2 클럭에 각기 응답하여 동작하는 제1,2 싱크 언팩커로 구성된 업사이저를 프로세서와 버스 사이에 구비함을 특징으로 하는 데이터 프로세싱 시스템에서의 대역폭 동기화 회로.

청구항 5

제4항에 있어서, 상기 제1 클럭이 400MHz일 경우에, 상기 제2 클럭은 200MHz임을 특징으로 하는 데이터 프로세싱 시스템에서의 대역폭 동기화 회로.

청구항 6

제5항에 있어서, 상기 제1 클럭은 CPU 사이드에서 상기 제2 클럭은 BUS 사이드에서 제공됨을 특징으로 하는 데이터 프로세싱 시스템에서의 대역폭 동기화 회로.

청구항 7

제1 클럭보다 주파수가 낮은 제2 클럭에 응답하여 동작하며 미리 설정된 리퀘스트에서는 제1 데이터 비트폭 보다 큰 제2 데이터 비트폭으로 리드 데이터를 출력하는 싱크 언팩커를 포함하는 업사이저와;

상기 제1 데이터 비트 폭으로 리드 데이터가 입력되는 일반 리퀘스트에서는 상기 입력된 제1 데이터 비트 폭의 리드 데이터를 바이패스하고, 상기 미리 설정된 리퀘스트에서는 상기 제2 데이터 비트 폭으로 입력되는 상기 리드 데이터를 상기 제1 클럭에 응답하여 공급하는 리퀘스트 선택 공급부를 구비함을 특징으로 하는 데이터 프로세싱 시스템에서의 대역폭 동기화 회로.

청구항 8

제7항에 있어서, 상기 리퀘스트 선택 공급부는 CPU 사이트에 형성됨을 특징으로 하는 데이터 프로세싱 시스템에서의 대역폭 동기화 회로.

청구항 9

제7항에 있어서, 상기 미리 설정된 리퀘스트는 랩 4 버스트 리드임을 특징으로 하는 데이터 프로세싱 시스템에서의 대역폭 동기화 회로.

청구항 10

제7항에 있어서, 상기 제1 데이터 비트 폭이 64비트인 경우에 상기 제2 데이터 비트 폭은 128비트임을 특징으로 하는 데이터 프로세싱 시스템에서의 대역폭 동기화 회로.

명세서

발명의 상세한 설명

기술분야

[0001] 본 발명은 데이터 프로세싱 시스템에 관한 것으로, 보다 상세하게는, 스마트 폰이나 네비게이션 디바이스 등과 같은 모바일 시스템에서의 대역폭 동기화에 관한 것이다.

배경기술

[0002] 스마트 폰들, 개인용 네비게이션 디바이스들, 휴대 인터넷기기, 휴대 방송기기, 또는 멀티미디어 기기 등과 같은 모바일 시스템에서는 다양한 어플리케이션(application)을 지원하기 위해 시스템 온 칩(SoC: System on Chip, 이하 "SoC" 라 칭함)상에 고 주파수로 동작되는 고성능의 모바일 응용 프로세서를 채택하고 있다.

[0003] 그러한 모바일 응용 프로세서는 연산과 프로그램 명령어 수행을 담당하기 때문에 모바일 SoC의 성능을 결정짓는 핵심 소자이다. 모바일 응용 프로세서는 무선통신, 개인용 네비게이션, 카메라, 포터블 게이밍, 포터블 음악/비디오 플레이어, 일체화된 모바일 TV 및 PDA 등과 같은 다양한 기능들의 집적을 인에이블(enable)하기 위해, L2 (레벨 2)캐시로서도 불리는 온칩 2차 캐시를 포함할 수 있다. 상기 L2 캐시의 부가는, 심각한 메모리 트래픽이 프로세서에 의해 발생되었을 때, 모바일 시스템의 성능을 증가시키는 인식된 방법이다.

[0004] 상기 SoC의 효과적인 설계를 위해서는 하나의 칩상에 집적된 복수의 IP(Intellecture Property)들간의 상호 통신을 위한 버스 시스템의 선택이 무엇보다 중요하다. 현재 알려진 대표적인 버스 시스템으로서는 ARM(Advanced RISC Machine)사의 AMBA(Advanced Microcontroller Bus Architecture)프로토콜을 기반으로 하는 AMBA 3.0 AXI(Advanced eXtensible Interface)버스 시스템이 있다.

[0005] SoC의 일부를 구성하는 DMAC(Direct Memory Access Controller), USB, PCI, SMC(Static Memory Controller), SCI(Smart Card Interface)등과 같은 주변기능 블록들은 제조 메이커의 개발 시간과 개발 인력 등과 같은 제약에 기인하여 외부에서 IP들로써 구입될 수 있다. 구입된 주변기능 블록 IP들은 CPU와 데이터 처리용 기능 블록들과 함께 칩 상에 집적(Integration)되어 SoC를 이루게 된다.

[0006] 모바일 응용 프로세서의 하이 퍼포먼스(High performance) 요구의 증가에 따라, SoC내의 CPU 및 캐시 컨트롤러(cache controller)의 동작 주파수(frequency)는 수 GHz로 증가되는 추세이다. 반면에, 버스(Bus) 시스템의 경우 주파수(frequency)를 수 GHz로 높이기 힘든 제약이 있으므로, 대역 폭을 맞추기 위해, 대신에 CPU의 데이터 버스 폭보다 넓은 데이터 버스 폭이 채용되어진다. 예를 들어, 1GHz의 동작주파수를 갖는 CPU의 데이터 버스 폭이 64비트라고 하면, 버스 시스템의 동작주파수는 200MHz로, 데이터 버스 폭은 128비트로 설계될 수 있다.

[0007] 캐시 컨트롤러(cache controller)와 연결 가능하며 64bit 데이터 버스폭과 1GHz 동작 주파수를 갖는 CPU와,

128bit 데이터 버스 폭과 200MHz 동작 주파수를 갖는 버스 시스템 사이에서 1GHz to 200MHz 동기(synchronize)를 수행하기 위한 싱크 다운 로직(syncdown logic)과 64bit to 128bit 업사이저(upsizer)회로가 채용되어질 수 있다.

[0008] 그러한 경우에, 동기화(synchronize)되는 싱크다운 포인트(syncdown point)에서 64bit 200MHz로 동작하는 부분이 약 1.6GBps로 되어, 약 8GBps를 갖는 CPU 나 약 3.2GBps를 갖는 업사이저에 비해, 대역폭 보틀넥(bandwidth bottleneck)으로 작용한다. 따라서, 고주파수(high frequency)CPU, 하이 데이터 폭(high data width) BUS 시스템 둘 모두의 성능이 저하될 수 있다.

[0009] 따라서, 모바일 시스템에서의 대역폭 보틀넥을 해소하여 시스템 퍼포먼스를 개선할 수 있는 대역폭 동기화 기술이 요망된다.

발명의 내용

해결 하고자하는 과제

[0010] 본 발명이 해결하고자 하는 기술적 과제는, 시스템 퍼포먼스를 개선할 수 있는 대역폭 동기화 회로를 제공함에 있다.

[0011] 본 발명이 해결하고자 하는 다른 기술적 과제는, 대역폭 보틀넥을 해소할 수 있는 데이터 프로세싱 시스템을 제공함에 있다.

[0012] 본 발명이 해결하고자 하는 또 다른 기술적 과제는, 고주파수 협대역 CPU와 저주파수 광대역 버스간의 대역폭 보틀넥을 해소할 수 있는 데이터 프로세싱 시스템에서의 대역폭 동기화 회로 및 그에 따른 대역폭 동기화 방법을 제공함에 있다.

과제 해결수단

[0013] 상기 기술적 과제를 달성하기 위하여, 본 발명의 실시 예의 일 양상에 따른 대역폭 동기화 회로는, 프로세서 클럭에 따라 동작하는 싱크 팩커와 싱크 언팩커로 구성된 업사이저와; 상기 업사이저와 연결되며 상기 프로세서 클럭보다 낮은 주파수의 버스 클럭에 응답하여 상기 업사이저의 데이터에 대한 싱크 다운을 수행하는 싱크 다운부를 구비한다.

[0014] 본 발명의 실시 예에서, 상기 프로세서 클럭이 약 1GHz의 주파수를 가질 경우에 상기 버스 클럭은 약 200MHz의 주파수를 가질 수 있다. 또한, 상기 싱크 팩커는 라이트 어드레스 채널, 라이트 데이터 채널, 및 라이트 응답 채널에 싱크 팩킹을 수행하고, 상기 싱크 언팩커는 리드 어드레스 채널과 리드 데이터 채널에 대한 싱크 언팩킹을 수행할 수 있다.

[0015] 본 발명의 실시 예의 다른 양상에 따른 대역폭 동기화 회로는, 서로 다른 주파수의 제1,2 클럭에 각기 응답하여 동작하는 제1,2 싱크 팩커와; 상기 제1,2 클럭에 각기 응답하여 동작하는 제1,2 싱크 언팩커로 구성된 업사이저를 프로세서와 버스 사이에 구비한다.

[0016] 본 발명의 실시 예에서 상기 제1 클럭이 400MHz일 경우에, 상기 제2 클럭은 200MHz일 수 있다. 또한, 상기 제1 클럭은 CPU 사이드에서 상기 제2 클럭은 BUS 사이드에서 제공될 수 있다.

[0017] 본 발명의 실시 예의 또 다른 양상에 따른 대역폭 동기화 회로는, 제1 클럭보다 주파수가 낮은 제2 클럭에 응답하여 동작하며 미리 설정된 리퀘스트에서는 제1 데이터 비트폭 보다 큰 제2 데이터 비트폭으로 리드 데이터를 출력하는 싱크 언팩커를 포함하는 업사이저와;

[0018] 상기 제1 데이터 비트 폭으로 리드 데이터가 입력되는 일반 리퀘스트에서는 상기 입력된 제1 데이터 비트 폭의 리드 데이터를 바이패스하고, 상기 미리 설정된 리퀘스트에서는 상기 제2 데이터 비트 폭으로 입력되는 상기 리드 데이터를 상기 제1 클럭에 응답하여 공급하는 리퀘스트 선택 공급부를 구비한다.

[0019] 본 발명의 실시 예에서, 상기 리퀘스트 선택 공급부는 CPU 사이드에 형성될 수 있으며, 상기 미리 설정된 리퀘스트는 랩 4 버스트 리드일 수 있다.

- [0020] 본 발명의 실시 예에서, 상기 제1 데이터 비트 폭이 64비트인 경우에 상기 제2 데이터 비트 폭은 128비트일 수 있다.
- [0021] 본 발명의 실시 예의 또 다른 양상에 따른 대역폭 동기화 회로는, 제1 클럭보다 주파수가 낮은 제2 클럭에 응답하여 동작하며 미리 설정된 리퀘스트에서는 제1 데이터 비트폭 보다 큰 제2 데이터 비트폭으로 데이터를 출력하는 싱크 언팩커를 포함하는 업사이저와;
- [0022] 상기 제1 데이터 비트 폭으로 데이터가 입력되는 일반 리퀘스트에서는 상기 입력된 제1 데이터 비트 폭의 데이터를 바이패스하고, 상기 미리 설정된 리퀘스트에서는 상기 제2 데이터 비트 폭으로 입력되는 상기 데이터를 상기 제1 클럭에 응답하여 공급하는 리퀘스트 선택 공급부를 구비한다.
- [0023] 본 발명의 실시 예에서, 상기 제2 데이터 비트 폭으로 입력되는 데이터는 상기 업사이저에서 출력되는 리드 데이터이며, 상기 제1 데이터 비트 폭의 2배일 수 있다. 또한, 상기 미리 설정된 리퀘스트는 랩 4 버스트 리드에 관련된 코멘드일 수 있다.
- [0024] 본 발명의 실시 예의 또 다른 양상에 따른 데이터 프로세싱 시스템은, L2 캐시 컨트롤러와 연결되는 CPU와; 제1 클럭보다 주파수가 낮은 제2 클럭에 응답하여 동작하며 미리 설정된 리퀘스트에서는 제1 데이터 비트폭 보다 큰 제2 데이터 비트폭으로 리드 데이터를 출력하는 싱크 언팩커를 포함하는 업사이저와, 상기 제1 데이터 비트 폭으로 리드 데이터가 입력되는 일반 리퀘스트에서는 상기 입력된 제1 데이터 비트 폭의 리드 데이터를 바이패스하고 상기 미리 설정된 리퀘스트에서는 상기 제2 데이터 비트 폭으로 입력되는 상기 리드 데이터를 상기 제1 클럭에 응답하여 공급하는 리퀘스트 선택 공급부를 포함하며, 상기 L2 캐시 컨트롤러를 통해 상기 CPU와 인터페이스 버스간에 연결되는 대역폭 동기화 회로와; 상기 인터페이스 버스에 연결된 주변기능 블록 IP들을 구비한다.
- [0025] 본 발명의 실시 예에서, 상기 주변기능 블록 IP들은, DMAC, USB, PCI, SMC, 및 SCI 중의 적어도 둘 이상을 포함할 수 있으며, 상기 인터페이스 버스는 AXI 버스일 수 있다.
- [0026] 본 발명의 실시 예에서, 상기 업사이저는 64 비트 랩 4 버스트 리드의 경우에 128비트의 리드 데이터를 200MHz 사이클에 하나씩 공급할 수 있다.
- [0027] 본 발명의 실시 예의 또 다른 양상에 따라, L2 캐시 컨트롤러와 연결되는 CPU와 주변기능 블록 IP들이 연결되는 인터페이스 버스 간에서의 대역폭 동기화 방법은, 상기 CPU 사이드를 제1 클럭으로 구동하고, 상기 인터페이스 버스에 연결된 업사이저를 제2 클럭으로 구동하는 단계와; 업사이저 사이드에서는 노말 리드의 경우에 제1 데이터 비트 폭의 리드 데이터를 상기 제2 클럭에 맞추어 출력하고, 64 비트 랩 4 버스트 리드의 경우에 제2 데이터 비트 폭의 리드 데이터를 상기 제2 클럭에 맞추어 출력하는 단계와; CPU 사이드에서는 상기 제1 데이터 비트 폭의 리드 데이터가 입력되는 경우에는 상기 리드 데이터를 바이패스 하고 상기 제2 데이터 비트 폭의 리드 데이터가 입력되는 경우에는 상기 리드 데이터를 상기 제1 클럭에 맞추어 2사이클 동안에 공급하는 단계로 이루어질 수 있다.
- [0028] 본 발명의 실시 예에서, 상기 제1 데이터 비트 폭이 64비트인 경우에 상기 제2 데이터 비트 폭은 128비트일 수 있으며, 상기 제1 클럭이 약 1GHz의 주파수를 가질 경우에 상기 버스 클럭은 약 200MHz의 주파수일 수 있다.

효 과

- [0029] 본 발명의 실시예의 대역폭 동기화 회로 구성에 따르면, CPU와 버스 사이의 동기화 보틀넥이 최소화 또는 감소된다.
- [0030] 따라서, 그러한 대역폭 동기화 회로를 SoC에 채용하는 경우에 데이터 처리 시스템의 제조 원가가 낮아질 수 있으며 SoC의 동작 성능도 높아진다.

발명의 실시를 위한 구체적인 내용

- [0031] 위와 같은 본 발명의 목적들, 다른 목적들, 특징들 및 이점들은 첨부된 도면과 관련된 이하의 바람직한 실시예들을 통해서 쉽게 이해될 것이다. 그러나 본 발명은 여기서 설명되는 실시 예에 한정되지 않고 다른 형태로 구체화될 수도 있다. 오히려, 여기서 소개되는 실시 예는, 이해의 편의를 제공할 의도 이외에는 다른 의도 없이, 개시된 내용이 보다 철저하고 완전해질 수 있도록 그리고 당업자에게 본 발명의 사상이 충분히 전달될 수 있도록

록 하기 위해 제공되는 것이다.

- [0032] 본 명세서에서, 어떤 소자 또는 라인들이 대상 소자 블록에 연결 된다 라고 언급된 경우에 그것은 직접적인 연결뿐만 아니라 어떤 다른 소자를 통해 대상 소자 블록에 간접적으로 연결된 의미까지도 포함한다.
- [0033] 또한, 각 도면에서 제시된 동일 또는 유사한 참조 부호는 동일 또는 유사한 구성 요소를 가급적 나타내고 있다. 일부 도면들에 있어서, 소자 및 라인들의 사이즈는 기술적 내용의 효과적인 설명을 위해 과장되어 나타날 수 있다.
- [0034] 여기에 설명되고 예시되는 각 실시 예는 그것의 상보적인 실시 예도 포함될 수 있음을 유의하라.
- [0035] 먼저, 도 1은 본 발명의 실시 예에 적용 가능한 대역폭 동기화 회로의 블록 구성도이다.
- [0036] 도면을 참조하면, 업사이저(200)가 슬레이브 인터페이스(100)와 버스 매트릭스(300)간에 설치된 것이 보여진다. 상기 업사이저(200)는 대역폭 동기화를 수행하기 위해 예를 들어 64비트의 데이터를 128비트의 데이터로 확장하는 데이터 익스팬더(expander)의 업그레이드된 기능을 수행할 수 있다. 도 1에서, 상기 슬레이브 인터페이스(100)는 64bit 데이터 버스폭과 1GHz 동작 주파수를 갖는 CPU 측(side)에 연결될 수 있고, 상기 버스 매트릭스(300)는 128bit 데이터 버스 폭과 200MHz 동작 주파수를 갖는 버스 시스템이 될 수 있다. 상기 버스 매트릭스(300)는 멀티 레이어 버스 매트릭스(Multi Layer Bus Matrix)구조를 가질 수 있다.
- [0037] 도 1에서 AW는 라이트 어드레스 채널 신호들, W는 라이트 데이터 채널 신호들, B는 라이트 응답 채널 신호들, AR은 리드 어드레스 채널 신호들, 및 R은 리드 데이터 채널 신호들을 각기 나타낸다. 또한, SI는 슬레이브 인터페이스를, MI는 마스터 인터페이스를 각기 나타낸다.
- [0038] 도 2는 도 1중 업사이저(또는 익스팬더)와 연결된 버스 구조의 예를 보여주는 데이터 프로세싱 시스템의 일부 블록 구성도이다.
- [0039] 도 2를 참조하면, 데이터 프로세싱 시스템의 구성 블록(500)내의 업사이저(200)를 기준으로 상부는 협(narrow) AXI 버스, 하부는 광 AXI 버스로 되어 있음을 알 수 있다. 상기 업사이저(200)는 협 버스 라인(BN)으로 인가되는 32비트, 32비트, 64비트의 데이터를 64비트, 128비트, 128비트로 각기 확장하여 광 버스 라인(BW)으로 제공하는 역할을 할 수 있다.
- [0040] 도 3은 본 발명의 제1 실시 예에 따른 대역폭 동기화 회로의 세부적 블록 구성도이고, 도 4는 도 3중 싱크 다운부의 예를 보여주는 구체 블록도이다.
- [0041] 이하에서는 도 3 및 도 4를 참조로 제1 실시 예가 설명될 것이다.
- [0042] 도 3에서, 업사이저(200)와 싱크 다운부(250)는 대역폭 동기화 회로를 구성한다.
- [0043] 상기 업사이저(200)는 프로세서 클럭(CLK1)에 따라 동작하는 싱크 팩커(220)와 싱크 언팩커(240)로 구성된다. 상기 싱크 팩커(sync packer:220)는 제1,2 싱크 메모리(21,23)와 싱크 팩킹 제어기(25)로 구성되며, 상기 싱크 언팩커(240)는 제3,4 싱크 메모리(41,43), 제1,2 선택기(42,44), 및 싱크 언팩킹 제어기(45)로 구성된다. 상기 싱크 팩커(220)는 라이트 어드레스 채널, 라이트 데이터 채널, 및 라이트 응답 채널에 싱크 팩킹을 수행하며, 상기 싱크 언팩커(240)는 리드 어드레스 채널과 리드 데이터 채널에 대한 싱크 언팩킹을 수행할 수 있다.
- [0044] 제1 싱크 메모리(21)는 상기 싱크 팩킹 제어기(25)의 제어에 응답하여 라이트 어드레스 채널의 어드레스를 저장하고, 그 저장된 어드레스를 업사이징하여 싱크 다운부(250)로 출력한다.
- [0045] 상기 제2 싱크 메모리(23)는 상기 싱크 팩킹 제어기(25)의 제어에 응답하여 라이트 데이터 채널의 데이터를 저장하고, 그 저장된 데이터를 업사이징하여 싱크 다운부(250)로 출력한다. 여기서, 64비트로 저장된 데이터는 싱크 다운부(250)로 제공시 상기 업사이징에 의해 128비트로서 인가될 수 있다.
- [0046] 상기 제3 싱크 메모리(41)는 상기 싱크 언팩킹 제어기(45)의 제어에 응답하여 리드 어드레스 채널의 어드레스를 저장하고, 그 저장된 어드레스를 업사이징하여 선택기(42)를 통해 싱크 다운부(250)로 출력한다.
- [0047] 상기 제4 싱크 메모리(43)는 상기 싱크 언팩킹 제어기(45)의 제어에 응답하여 리드 데이터 채널의 데이터를 저장하고, 그 저장된 데이터를 선택기(44)를 통해 슬레이브 인터페이스(100)로 출력한다. 여기서, 128비트로 저장된 데이터는 슬레이브 인터페이스(100)로 제공될 시 64비트로서 인가될 수 있다.
- [0048] 상기 제1,2,3,4 싱크 메모리들(21,23,41,43)은 모두 선입선출 기능을 가지는 FIFO(First In First Out)메모리

로 구현될 수 있다.

- [0049] 상기 싱크 다운부(250)는, 상기 업사이저(200)와 연결되며 상기 프로세서 클럭보다 낮은 주파수의 버스 클럭에 응답하여 상기 업사이저(200)의 출력에 싱크 다운을 수행할 수 있다.
- [0050] 도 3에서, 상기 프로세서 클럭이 약 1GHz의 주파수를 가진다고 하면, 상기 업사이저(200)에 인가되는 클럭(CLK1)도 프로세서의 클럭 도메인(domain)하에서 동작되기 때문에 약 1GHz로 주어질 수 있다. 한편, 상기 버스 클럭은 약 200MHz의 주파수를 가질 수 있다.
- [0051] 상기 프로세서(또는 CPU) 클럭은 SoC의 전류 소모를 최소화하기 위해 채용된 DVFS(Dynamic Voltage Frequency Scaling Controller)에 의해 제어되는 DVFS(Dynamic Voltage Frequency Scaling)클럭일 수 있다. 상기 DVFS의 채용에 의해 클럭 주파수는 다이내믹하게 제어될 수 있다.
- [0052] 도 4는 도 3중 싱크 다운부의 예를 보여주는 구체 블록도이다. 데이터의 저장을 위한 싱크 메모리(252), 매치 값을 저장하는 매치 밸류(254), 싱크 메모리(252)에 저장된 데이터가 매치 값과 매칭되는 지를 판단하는 매치(256), 상기 매치(256)의 활성화 신호(CLKEN)에 공통으로 응답하여 데이터를 래치 출력하는 제1,2 플립플롭들(258,259)로 이루어진 싱크 다운부(250)의 구현 예가 보여진다. 도 4에서, 출력 라인(L10)은 도 3에서의 AW, W, 및 AR을 한꺼번에 나타내고, 입력 라인(L20)은 도 3에서의 B, 및 R을 한꺼번에 나타낸다.
- [0053] 도 3과 같은 제1 실시 예의 경우에는 약 1GHz에서 고속 동작되는 업사이저를 만들기 어렵다는 제약과 고 주파수의 설계를 위해 파이프라인 구조를 삽입할 경우에 레이턴시(latency)에 악영향을 줄 수 있다는 제약이 있을 수 있다. 따라서, 그러한 제약들이 무시될 수 있는 경우에 제1 실시 예는 유용하다.
- [0054] 이하에서는 도 5 및 도 6을 참조로 제2 실시 예가 설명될 것이다.
- [0055] 도 5는 본 발명의 제2 실시 예에 따른 대역폭 동기화 회로의 세부적 블록 구성도이고, 도 6은 도 5의 업사이저의 구현 예를 보여주는 세부적 블록 구성도이다.
- [0056] 도 5를 참조하면, 데이터 프로세싱 시스템의 구성 블록(500)내에서, 대시 라인 바아(Ba1)를 경계로 제1 클럭(CLK1)과 제2 클럭(CLK2)에 응답하여 업사이징 기능을 수행하는 업사이저(210)의 블록 구성이 보여진다.
- [0057] 상기 업사이저(210)는 프로세서 측(100)과 버스 매트릭스(300)사이에 설치되며, 도 6에서, 서로 다른 주파수의 제1,2 클럭(CLK1,CLK2)에 각기 응답하여 동작하는 제1,2 싱크 팩커(222,224)와, 상기 제1,2 클럭(CLK1,CLK2)에 각기 응답하여 동작하는 제1,2 싱크 언팩커(242,244)로 구성된다. 상기 제1,2 싱크 팩커(222,224)는 1,2 싱크 메모리(21,23)를 공유적으로 가질 수 있다. 상기 제1 싱크 팩커(222)는 제1 싱크 팩킹 제어기(26)를 포함하며, 상기 제2 싱크 팩커(224)는 제2 싱크 팩킹 제어기(27)를 포함할 수 있다. 상기 제1,2 싱크 언팩커(242,244)는 3,4 싱크 메모리(41,43)를 공유적으로 가질 수 있다. 상기 제1 싱크 언팩커(242)는 제2 선택기(44)와 제1 싱크 언팩킹 제어기(46)를 포함하며, 상기 제2 싱크 언팩커(244)는 제1 선택기(42)와 제2 싱크 언팩킹 제어기(47)를 포함할 수 있다. 도 6에서 제1,2 싱크 팩킹 제어기들(26,27)과 제1,2 싱크 언팩킹 제어기(46,47)에서 보여지는 참조 문자 FSM_s는 파이널 스테이트 머시인_슬레이브를 의미하고, FSM_m는 파이널 스테이트 머시인_마스터를 의미한다.
- [0058] 상기 제1 클럭(CLK1)이 400MHz일 경우에, 상기 제2 클럭(CLK2)은 200MHz일 수 있다. 따라서, 상기 제1 클럭(CLK1)은 CPU 사이드에서, 상기 제2 클럭(CLK2)은 BUS 사이드에서 제공될 수 있다.
- [0059] 도 6에서 보여지는 업사이저(210)는 2부분으로 나뉘어져 서로 다른 주파수로 동작하는 것을 제외하면, 도 3의 업사이저(200)와 같다.
- [0060] 이와 같이, 64비트와 128비트로서 비트폭이 서로 다른 경우에, 업사이저(210)내의 제1 싱크 팩커(222)와 제1 싱크 언팩커(224)가 400MHz로 동작되도록 하고, 제2 싱크 팩커(224)와 제2 싱크 언팩커(244)가 200MHz로 동작되도록 하면, 대역폭 밸런스(Bandwidth balance)가 맞추어 진다.
- [0061] 그러나, 도 6에 따른 대역폭 동기화 회로의 경우에는, 200MHz 버스 클럭과 200MHz~1GHz DVFS 클럭 사이에서 400MHz 클럭이 추가로 필요하게 된다. 따라서, 추가 클럭의 생성은 업사이저 내의 하드웨어 부담을 가중시키고 업사이징 효율 저하를 유발할 수 있다. 결국, 제2 실시 예의 경우에는 추가 클럭의 생성에 대한 부담이 해소되는 경우에 대역폭 동기화 회로로서 유용하다.
- [0062] 도 7은 본 발명의 제3 실시 예에 따른 대역폭 동기화 회로의 세부적 블록 구성도이고, 도 8은 도 7중 업사이저의 일부를 보여주는 구체 블록도이다. 또한, 도 9는 도 7에 관련된 동작 타이밍도이다.

- [0063] 도 7을 참조하면, 제2 클럭(CLK2)으로 구동되는 업사이저(202)와, CPU 사이트의 SI(102)내에서 메모리(10), 선택기(14), 및 레지스터(19)로 이루어진 리퀘스트 선택 공급부의 연결 구성이 보여진다. 상기 업사이저(202)와 상기 리퀘스트 선택 공급부는 본 발명의 제3 실시 예에 따른 대역폭 동기화 회로를 구성한다.
- [0064] 상기 업사이저(202)는 제2 클럭(예를 들어 200MHz)에 응답하여 동작하며 미리 설정된 리퀘스트(예를 들어 랩 4 버스트 리드 리퀘스트, 또는 랩 8 버스트 리드 리퀘스트)에서는 제2 데이터 비트폭(예를 들어 128비트)으로 리드 데이터를 출력하는 싱크 언팩커(242:도 8 참조)를 포함한다.
- [0065] 상기 리퀘스트 선택 공급부는 제1 데이터 비트 폭(예를 들어 64비트)으로 리드 데이터가 입력되는 일반 리퀘스트(인크리먼트 버스트 혹은 픽스드 버스트)에서는 상기 리드 데이터를 바이패스하고, 상기 랩(Wrap)4 버스트 리드 리퀘스트(burst read request)에서는 상기 제2 데이터 비트 폭(예를 들어 128비트)으로 입력되는 상기 리드 데이터를 상기 제2 클럭보다 주파수가 높은 제1 클럭(예를 들어 1GHz)에 응답하여 2클럭 사이클 동안 공급한다.
- [0066] 상기 메모리(10)는 랩(Wrap)4 버스트 리드 리퀘스트(burst read request)의 유무를 검출하는 회로 소자로서 어드레스 리드 콘텐츠 어드레서블 메모리(ARCAM)로 구성될 수 있다. 또한, 선택기(14)는 선택신호(SEL)의 활성화에 응답하여 버스라인들(B2,B3)을 통해 인가되는 총 128비트의 데이터를 수신하고, 그 수신된 총 128비트의 데이터를 제1 클럭(예를 들어 1GHz)에 맞추어 2회에 걸쳐 R채널(18)에 공급한다. 상기 선택기(14)는 상기 선택신호(SEL)가 비활성화된 경우(일반 리퀘스트)에는 상기 버스라인(B2)을 통해 인가되는 64비트의 데이터를 바이패스하여 상기 R채널(18)에 공급한다. 한편, 레지스터(19)는 리오더(reorder)를 위한 저장소자로서, 예를 들어, 랩 4 버스트 리드 리퀘스트(burst read request)시에 데이터가 순서대로 들어오지 않은 경우에 데이터를 순서대로 출력하기 위해 필요한 저장소자이다.
- [0067] 도 8을 참조하면, 일반적인 업사이저 내에 형성된 싱크 언팩커(242)의 구성이 보여진다. 제3,4 싱크 메모리(41,43), 제1,2 선택기(42,44), 랩(Wrap)4 버스트 리드 리퀘스트(burst read request)의 유무를 검출하기 위한 메모리(46), 및 싱크 언팩킹 제어기(45)로 구성된다. 상기 싱크 언팩커(242)는 리드 어드레스 채널(AR)과 리드 데이터 채널(R)에 대한 싱크 언팩킹을 수행할 경우에, 미리 설정된 리퀘스트(예를 들어 랩 4 버스트 리드 리퀘스트)에서는 버스라인들(B2,B3)을 통해 제2 데이터 비트폭(예를 들어 128비트)으로 리드 데이터를 출력한다. 또한, 랩 버스트 리드 리퀘스트가 아닌 경우에는 제4 싱크 메모리(43)에서 출력된 64비트의 데이터가 제2 클럭(예를 들어 200MHz)에 맞추어 출력된다. 상기 랩 4 버스트 리드 리퀘스트의 경우에 선택에 따라, 상기 제4 싱크 메모리(43)에서 128비트의 데이터가 출력되거나, 상기 제4 싱크 메모리(43)의 입력단에서 바이패스된 128비트의 데이터가 출력될 수 있다.
- [0068] 도 9에서, RDATA2a를 참조시 랩 4 버스트 리드 리퀘스트에서 상기 제2 데이터 비트폭(예를 들어 128비트)으로 리드 데이터가 출력되는 것이 나타난다. 상기 RDATA2a는 도 7의 선택기(14)에서 출력되는 데이터의 타이밍을 보여준다. 도 9에서 보여지는 CLK은 CPU의 클럭이다. CPU의 클럭은 예를 들어, 1GHz의 주파수를 갖는 상기 제1 클럭에 대응될 수 있다. 또한, ACLK은 AXI 버스 클럭으로서, 상기 200MHz의 주파수를 갖는 상기 제2 클럭에 대응될 수 있다. INCLKEN은 입력 클럭 인에이블 신호이다.
- [0069] 도 9에서, 타임 포인트 t1와 t2간의 구간은 CPU의 1클럭 주기에 대응된다. 또한 t3와 t4간의 구간도 상기 CPU의 1클럭 주기에 대응된다. RDATA1a를 참조시 랩 4 버스트 리드 리퀘스트에서는 버스 클럭(200MHz)의 1주기 동안에 128비트의 데이터가 수신되는 것이 보여진다. 상기 수신된 128비트의 RDATA1a는 상기 INCLKEN으로부터 변경된 인에이블 신호 INCLKEN_M에 따라 상기 CPU의 2클럭 주기 동안에 128비트의 데이터(a1과 a2를 합한 데이터)로서 출력된다.
- [0070] 한편, RDATA1는 랩 4 버스트 리드 리퀘스트가 아닌 경우에 64비트의 데이터가 수신되는 것을 나타내고, RDATA2는 상기 수신된 64비트의 데이터가 상기 CPU의 1클럭 주기 동안에 64비트의 데이터(a1)로서 출력되는 것을 보여준다.
- [0071] 이와 같이, 퍼포먼스 이슈에 영향을 주는 특정한 리퀘스트 예를 들어 랩 4 버스트 리드 리퀘스트인 경우에만 제2 클럭으로써 R 채널 데이터 비트 폭의 2배에 상응하는 데이터를 한꺼번에 보내고, 제1 클럭의 2주기 동안에 데이터를 공급하게 되면 대역폭의 밸런스가 효율적으로 유지된다.
- [0072] 제3 실시 예에서, 랩 4 버스트 리드 리퀘스트를 특정한 리퀘스트로 선정하는 이유는 도 10 및 도 11의 트레이스(trace) 분석 테이블들에 연유된다.
- [0073] 도 10 및 도 11은 프로세서의 동작 중 헤비 리퀘스트의 발생 빈도에 대한 일 예들을 보여주는 테이블들이다.

- [0074] 명령 실행의 아웃이나 멀티플 로드(load)와 같은 헤비 리퀘스트가 발생하는 경우에는 캐시 미스(miss) 케이스가 초래될 수 있어 시스템 동작 퍼포먼스가 저하된다. 본 발명의 실시 예에서는 캐시 미스 케이스의 빈도를 알기 위해 2가지의 경우에 대하여 CPU 트레이스의 분석을 행하였다.
- [0075] 먼저, 도 10을 참조하면, ARM(Advanced RISC Machine)사의 AXI 버스를 이용한 ARM1176PB_L2WA1loc_AXI.out의 경우에 헤비(heavy) 리퀘스트 빈도를 나타내는 테이블이 보여진다. 또한, 도 11을 참조하면, ARM1176PB_L2AWCACHEattr_AXI.out의 경우에 헤비 리퀘스트 빈도를 나타내는 테이블이 보여진다. 여기서, L2WA1loc는 L2 캐시의 얼로케이션을 의미하고, L2AWCACHEattr는 L2 캐시의 라이트 어드레스의 어트리뷰션을 나타낸다.
- [0076] 도 10에서는 AR 랩 버스트 4 리퀘스트 중에서 4 클럭 사이클 이내에 발생하는 트래픽 카운트는 17491이고, 이는 9.1%의 어큐물레이션 비율로 나타나는 것이 보여진다. 또한 도 11의 경우에는 AR 랩 버스트 4 리퀘스트 중에서 4 클럭 사이클 이내에 발생하는 트래픽 카운트는 14621이고, 이는 7.5%의 어큐물레이션 비율로 나타나는 것이 보여진다. 결국, 도 10의 경우에 전체에서 헤비 리퀘스트가 발생하는 빈도는 5.88%로 나타나고, 도 11의 경우에 전체에서 헤비 리퀘스트가 발생하는 빈도는 1.51%로 나타났다.
- [0077] 상기한 바와 같이, 제3 실시 예의 경우에는 대역폭 보틀넥을 해소하기 위해 퍼포먼스 크리티컬한 랩 4 버스트 리드 리퀘스트의 경우에 도 9의 RDATA1a와 같은 타이밍으로 데이터를 전송하고 RDATA2a와 같은 타이밍으로 데이터를 CPU 측에 공급함에 의해, 대역폭 보틀넥의 해소가 보다 효율적으로 달성된다.
- [0078] 도 12는 본 발명의 실시 예들에 따른 대역폭 동기화 회로를 채용한 모바일 시스템의 블록 구성도이다.
- [0079] 도면을 참조하면, L2 캐시를 갖는 CPU(500)와, AXI 버스(BUS1)를 통해 연결된 미디어 시스템(510), 모뎀(520), 메모리 컨트롤러(410), 부트 롬(430), 및 디스플레이 컨트롤러(440)가 보여진다. 상기 메모리 컨트롤러(410)에는 DRAM 또는 플래시 메모리 등과 같은 메모리(420)가 연결될 수 있으며, 상기 디스플레이 컨트롤러(440)에는 LCD 등과 같은 디스플레이(450)가 연결될 수 있다.
- [0080] 도 12에서 BUS1은 CPU 버스이고, BUS2는 메모리 버스이다. 상기 CPU(500)는 레벨 2(L2) 캐시이외에 레벨 원(L1)캐시(CACHE)를 내부적으로 포함할 수 있다. 상기 L1 캐시는 빈번히 액세스되어지는 데이터 및/또는 명령들을 저장하기 위해 사용된다. 상기 L2 캐시 메모리도 L1 캐시와 마찬가지로, 빈번히 액세스되어지는 데이터 및/또는 명령들을 저장할 수 있다.
- [0081] 도 12의 모바일 시스템은 스마트 폰들, 개인용 네비게이션 디바이스들, 휴대 인터넷기기, 휴대 방송기기, 또는 멀티미디어 기기로서 기능할 수 있다.
- [0082] 도 12와 같은 모바일 시스템에서, 도 7과 같은 대역폭 동기화 회로는 상기 CPU(500)의 블록과 AXI 버스 사이에 탑재될 수 있다.
- [0083] 이 경우에 도 7의 SI 블록(102)은 상기 CPU 사이트에 있으므로 제1 클럭(약 1GHz)으로 구동되고, 상기 AXI 버스에 연결된 업사이저(202)는 제2 클럭(약 200M Hz)으로 구동될 수 있다.
- [0084] 상기 업사이저(202)의 사이트에서는 노말 리드의 경우에 64비트의 리드 데이터를 상기 200MHz 클럭에 맞추어 출력하고, 64 비트 랩 4 버스트 리드의 경우에는 128비트의 리드 데이터를 상기 200MHz의 클럭에 맞추어 출력한다.
- [0085] 상기 CPU 사이트의 블록(102)에서는 노말 리드인 경우에는 상기 64비트의 리드 데이터를 바이패스 하여 R 채널(18)을 통해 CPU로 인터페이스하고, 64 비트 랩 4 버스트 리드의 경우에는 상기 128비트의 리드 데이터를 상기 1GHz의 클럭에 맞추어 2사이클 동안에 CPU로 공급한다.
- [0086] 이와 같이, 퍼포먼스 크리티컬한 랩 4 버스트 리드 리퀘스트의 경우에 한정하여 대역폭 동기화를 수행하면 업사이저 회로의 변경을 최소화하면서도 효율적으로 대역폭 보틀넥 문제가 해소되어, SoC를 채용하는 모바일 시스템의 동작 퍼포먼스가 개선된다. 또한, 모바일 시스템과 같은 데이터 처리 시스템의 제조 원가가 낮아질 수 있다.
- [0087] 본 발명의 실시 예들에서는 64 비트 고 주파수 CPU 서브 시스템(subsystem)과 128 비트 저 주파수 버스 인터페이스 사이에서 발생하는 대역폭 보틀넥(bandwidth bottleneck)의 해소에 관하여 주로 설명되었으나, 이에 한정됨이 없이 대역폭 보틀넥이 발생할 수 있는 데이터 처리 시스템의 경우에 본 발명의 실시 예들이 확장적으로 적용될 수 있음은 물론이다.
- [0088] 본 발명의 실시 예가 적용되는 모바일 시스템에서 프로세서들의 개수는 2개 이상으로 확장될 수 있다. 그러한

프로세서들은 마이크로프로세서, CPU, 디지털 신호 프로세서, 마이크로 콘트롤러, 리듀스드 명령 세트 컴퓨터, 콤플렉스 명령 세트 컴퓨터, 또는 그와 유사한 것이 될 수 있다.

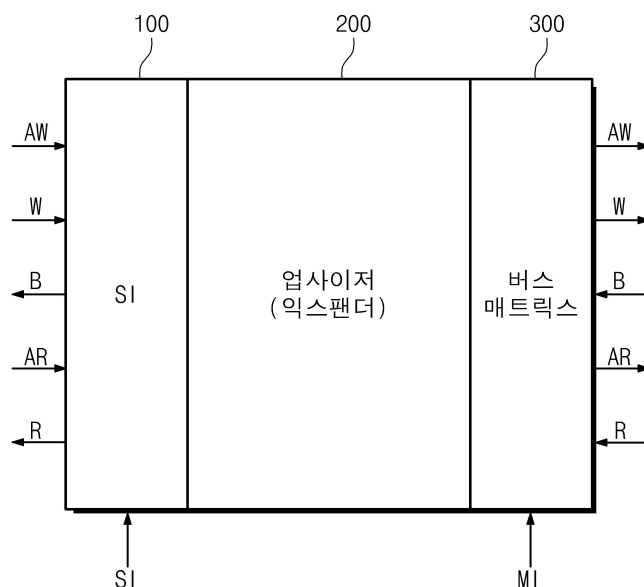
[0089] 상기한 설명에서는 본 발명의 실시예들을 위주로 도면을 따라 예를 들어 설명하였지만, 본 발명의 기술적 사상의 범위 내에서 본 발명을 다양하게 변형 또는 변경할 수 있음은 본 발명이 속하는 분야의 당업자에게는 명백한 것이다. 예를 들어, 사안이 다른 경우에 본 발명의 기술적 사상을 벗어나지 없이, CPU의 클럭이나 이와 연결되는 소자들의 배치순서 및 회로 구성을 다양하게 변형 또는 변경할 수 있음은 물론이다.

도면의 간단한 설명

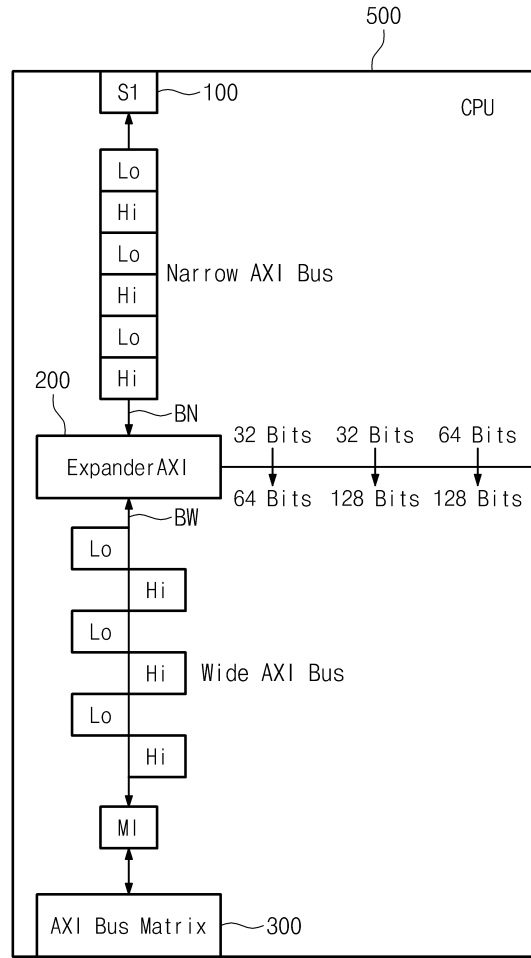
- [0090] 도 1은 본 발명의 실시 예에 적용 가능한 대역폭 동기화 회로의 블록 구성도
- [0091] 도 2는 도 1중 업사이저와 연결된 버스 구조의 예를 보여주는 데이터 프로세싱 시스템의 일부 블록 구성도
- [0092] 도 3은 본 발명의 제1 실시 예에 따른 대역폭 동기화 회로의 세부적 블록 구성도
- [0093] 도 4는 도 3중 싱크 다운부의 예를 보여주는 구체 블록도
- [0094] 도 5는 본 발명의 제2 실시 예에 따른 대역폭 동기화 회로의 세부적 블록 구성도
- [0095] 도 6은 도 5의 업사이저의 구현 예를 보여주는 세부적 블록 구성도
- [0096] 도 7은 본 발명의 제3 실시 예에 따른 대역폭 동기화 회로의 세부적 블록 구성도
- [0097] 도 8은 도 7중 업사이저의 일부를 보여주는 구체 블록도
- [0098] 도 9는 도 7에 관련된 동작 타이밍도
- [0099] 도 10 및 도 11은 프로세서의 동작 중 헤비 리퀘스트의 발생 빈도에 대한 일 예들을 보여주는 테이블들
- [0100] 도 12는 본 발명의 실시 예들에 따른 대역폭 동기화 회로를 채용한 모바일 시스템의 블록 구성도
- [0101] < 도면의 주요 부분에 대한 부호의 설명 >
- [0102] 100: 슬레이브 인터페이스 200,210: 업사이저
- [0103] 300: 버스 매트릭스 14: 선택기

도면

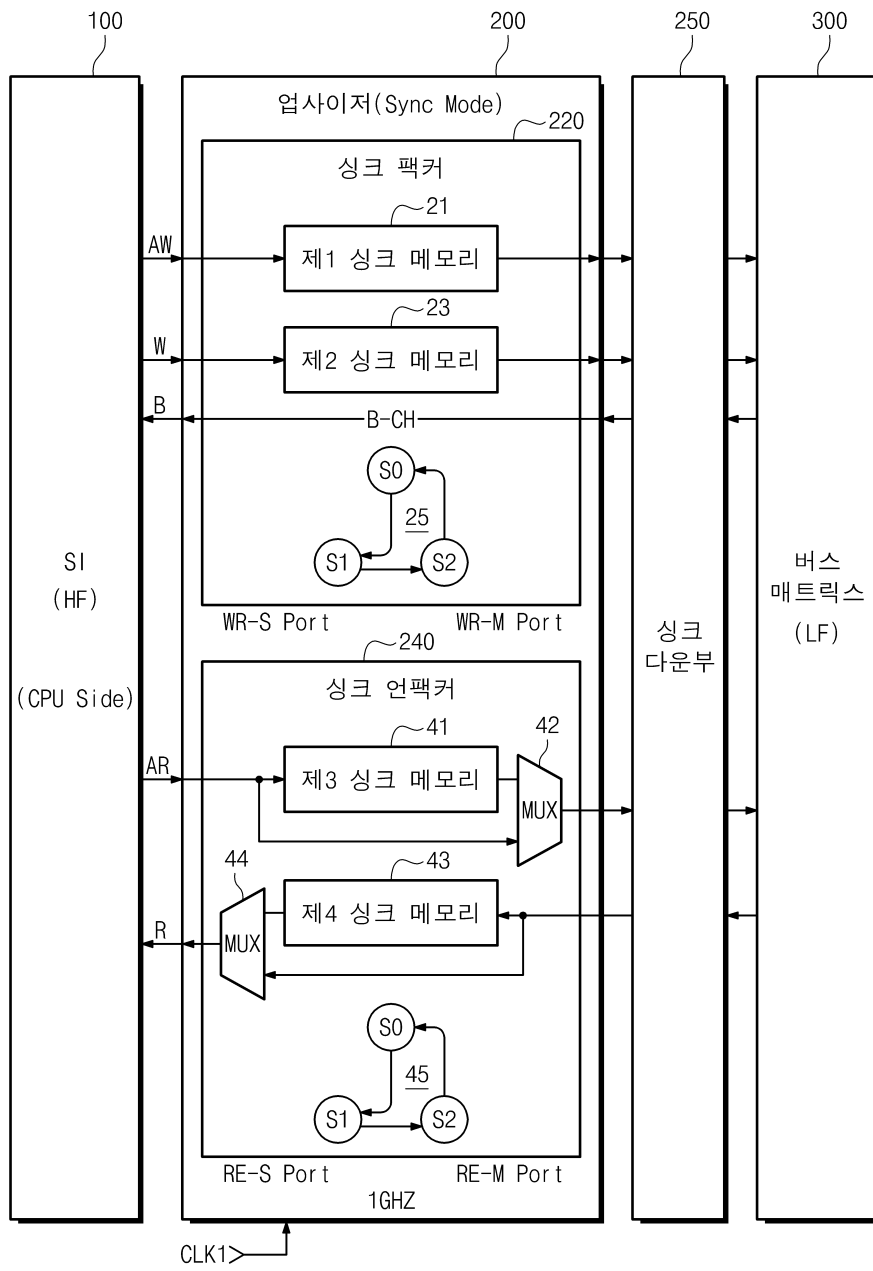
도면1



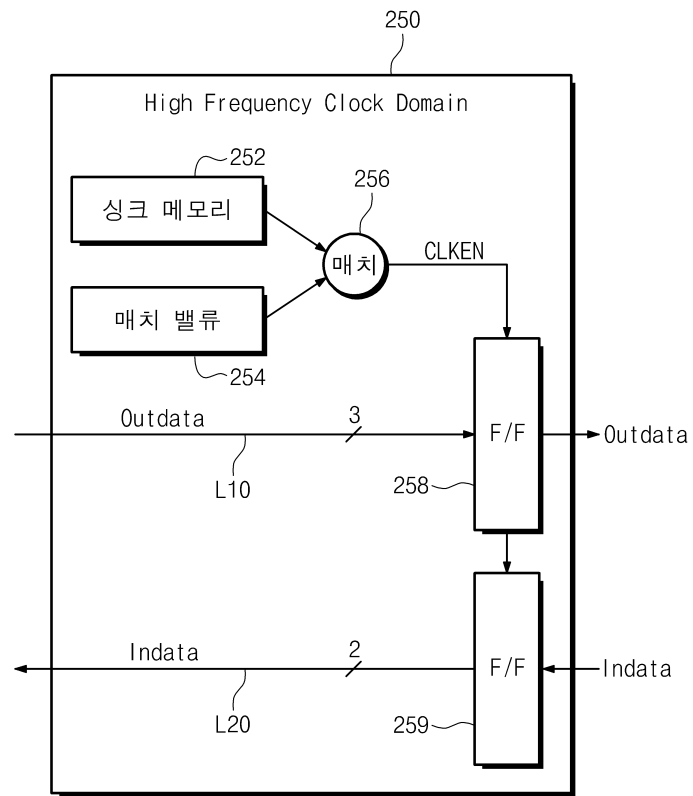
도면2



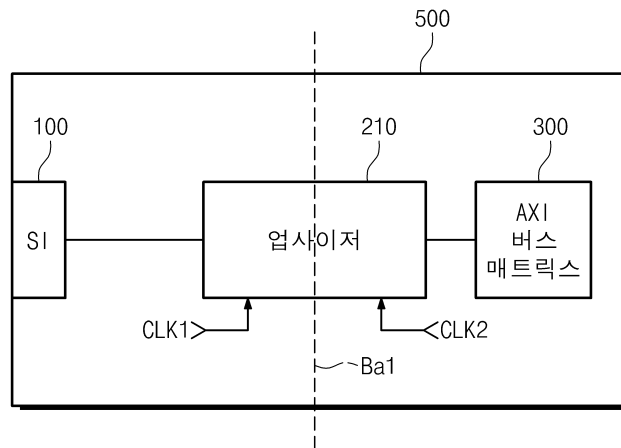
도면3



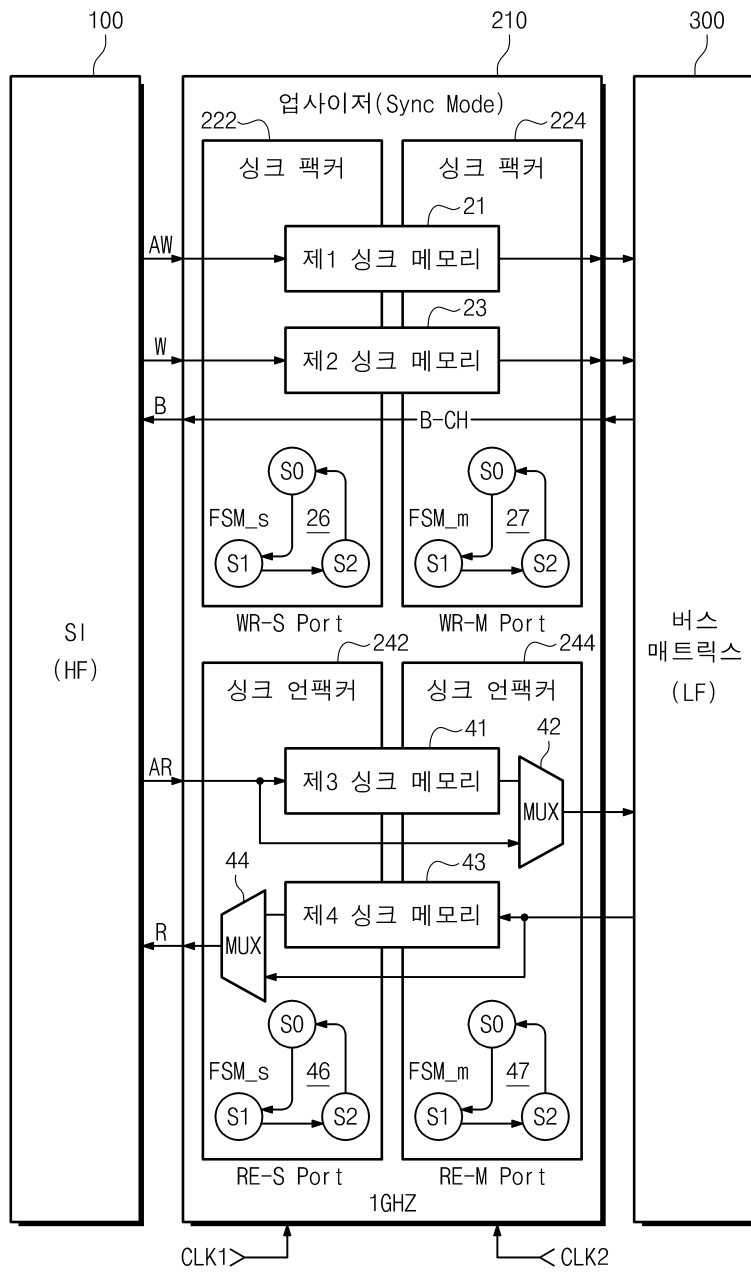
도면4



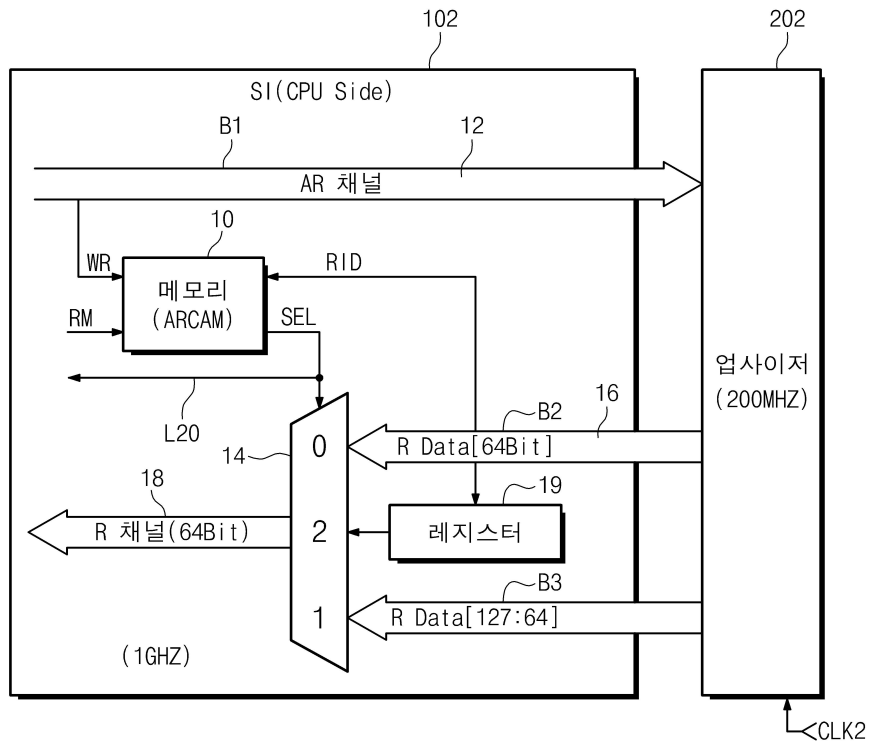
도면5



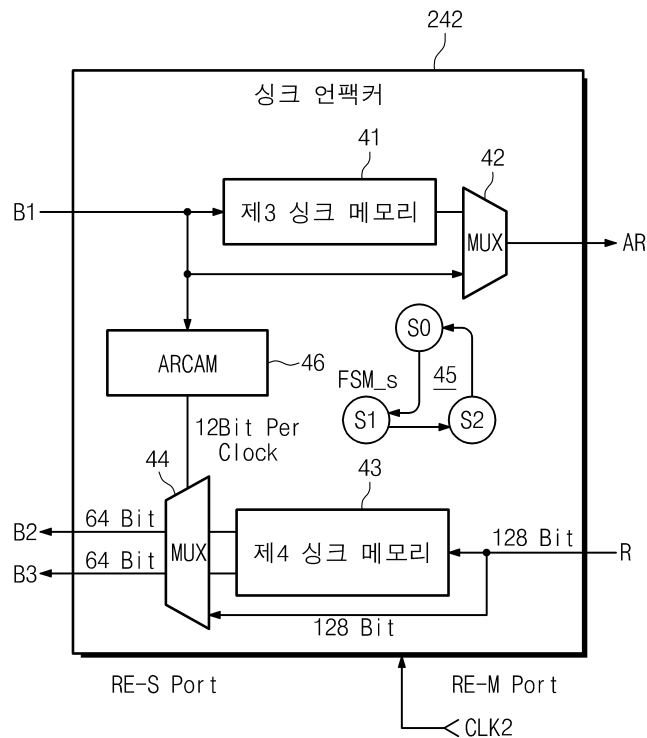
도면6



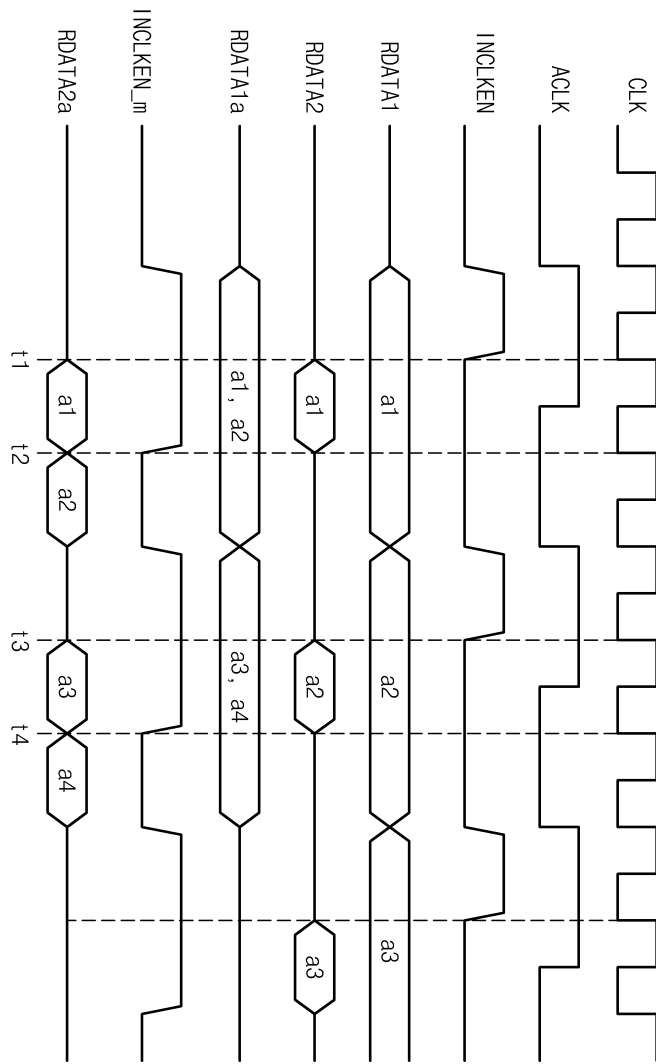
도면7



도면8



도면9



도면10

ARM1176PB_L2WAlloc_AXI.out			
Cycle from AR(n-1) to AR(n) (Wrap burst 4 read)	Traffic Count	Ratio(%)	Acc Ratio(%)
0	51948	2.9	2.9
1	49988	2.8	5.7
2	20729	1.2	6.9
3	21076	1.2	8.1
4	17491	1.0	9.1
5	9700	0.5	9.6
6	5917	0.3	10.0
7	7847	0.4	10.4
8	8582	0.5	10.9
9	4263	0.2	11.1
10이상	1578823	88.9	100
Total	1776364		
AR Total	1898231	106.9	
AW Total	845082	47.6	
W Total	2958550	166.6	

도면11

L2AWCACHEattr_AXI.out			
Cycle from AR(n-1) to AR(n) (Wrap burst 4 read)	Traffic Count	Ratio(%)	Acc Ratio(%)
0	14409	1.1	1.1
1	38617	2.9	3.9
2	17156	1.3	5.2
3	17099	1.3	6.5
4	14621	1.1	7.5
5	8697	0.6	8.2
6	4857	0.4	8.5
7	6393	0.5	9.0
8	5642	0.4	9.4
9	3886	0.3	9.7
10이상	1219169	90.3	100
Total	1350546		
AR Total	1483409	109.8	
AW Total	5245290	388.4	
W Total	6535315	483.9	

도면12

