



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2014년03월18일
 (11) 등록번호 10-1375466
 (24) 등록일자 2014년03월11일

- (51) 국제특허분류(Int. Cl.)
G11C 7/22 (2006.01) *G11C 7/10* (2006.01)
- (21) 출원번호 10-2011-7017991
- (22) 출원일자(국제) 2009년07월09일
 심사청구일자 2011년12월02일
- (85) 번역문제출일자 2011년07월29일
- (65) 공개번호 10-2011-0113738
- (43) 공개일자 2011년10월18일
- (86) 국제출원번호 PCT/US2009/050044
- (87) 국제공개번호 WO 2010/080176
 국제공개일자 2010년07월15일
- (30) 우선권주장
 61/144,135 2009년01월12일 미국(US)
 (뒷면에 계속)
- (56) 선행기술조사문헌
 US20070133312 A1*
 US20010028599 A1*
 *는 심사관에 의하여 인용된 문헌

- (73) 특허권자
램버스 인코포레이티드
 미국 캘리포니아주 94089 선니베일 스위트 700 엔터프라이즈 웨이 1050
- (72) 발명자
웨어 프레데릭 에이
 미국 94022 캘리포니아주 로스 알토스 힐스 프레몬트 파인스 레인 13961
폴턴 존 더블유
 미국 27516 노스 캐롤라이나주 채플 힐 유니언 글로브 처치 로드 8720
팔머 로버트 이
 미국 27510 노스 캐롤라이나주 카보로 스위트 베이 플레이스 236
- (74) 대리인
특허법인코리아나

전체 청구항 수 : 총 14 항

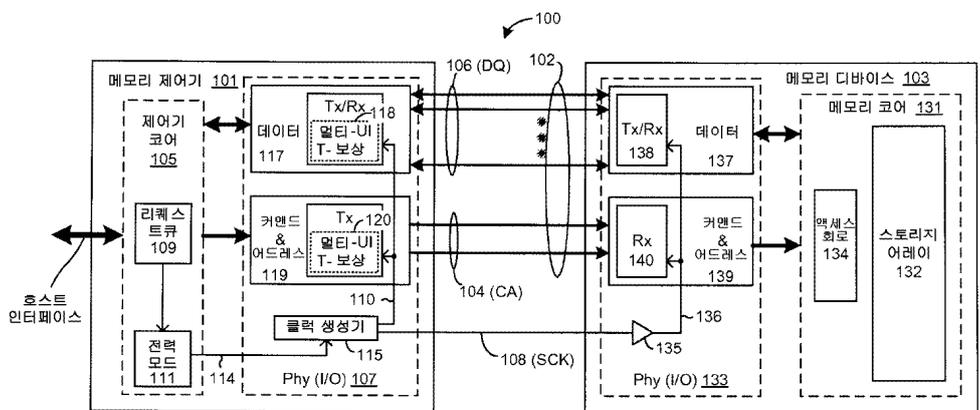
심사관 : 손운식

(54) 발명의 명칭 **다중 전력 모드를 갖는 메모리 컨트롤러 시스템**

(57) 요약

저-전력 시그널링 시스템에서, 집적 회로 디바이스는 소스-싱크로노스 타이밍 레퍼런스에 의해 수반되지 않은 정보-보유 심볼들의 고속 송신을 가능하게 하기 위해 함께 동작하는 개방 루프-클럭 분배 회로 및 송신 회로를 포함한다. 개방-루프 클럭 분배 회로는 외부-공급 클럭 신호에 응답하여, 송신 클럭 신호를 생성하고, 송신 회로는 송신 클럭 신호의 천이에 응답하여 외부 신호 라인 상에 심볼들의 시퀀스를 출력한다. 심볼들의 각각은 심볼 시간의 송신 회로의 출력에서 유효하고 송신 클럭 신호와 외부-공급 클럭 신호 사이의 위상 오프셋은 적어도 심볼 시간 만큼 드리프트하도록 허용된다.

대표도



(30) 우선권주장

61/156,872	2009년03월02일	미국(US)
61/177,467	2009년05월12일	미국(US)
61/177,478	2009년05월12일	미국(US)
61/177,596	2009년05월12일	미국(US)
61/177,599	2009년05월12일	미국(US)
61/177,606	2009년05월12일	미국(US)

특허청구의 범위

청구항 1

집적 회로 메모리 디바이스로서,

클럭 입력을 통해 제 1 클럭 신호를 수신하기 위한 클럭 수신기;

상기 클럭 수신기로부터 상기 제 1 클럭 신호를 수신하고 상기 제 1 클럭 신호의 천이들에 응답하여 상기 집적 회로 메모리 디바이스로부터 데이터 신호를 출력하기 위한 시그널링 회로로서, 상기 제 1 클럭 신호는 상기 데이터 신호에서 전달되는 데이터의 각 비트에 대한 각각의 천이를 가지며, 상기 데이터 신호에서 전달되는 데이터의 각 비트는, 각각의 비트 시간에 대해 상기 집적 회로 메모리 디바이스의 출력에서 유효하고, 상기 집적 회로 메모리 디바이스의 출력에서의 데이터 신호와 상기 클럭 입력에서의 상기 제 1 클럭 신호 사이의 위상 오프셋은 적어도 상기 비트 시간 만큼 드리프트되도록 허용되는, 상기 시그널링 회로;

메모리 제어기로부터의 제어 신호에 응답하여 상기 클럭 수신기의 동작을 디스에이블링하기 위한 제어 회로;

복수의 데이터 링크들을 통해 수신된 각각의 데이터 신호들을 수신하기 위한 복수의 수신기들;

상기 복수의 데이터 링크들을 통해 각각의 데이터 신호들을 출력하기 위한 복수의 송신기들로서, 상기 클럭 수신기로부터 상기 제 1 클럭 신호를 수신하고 데이터 신호를 출력하기 위한 상기 시그널링 회로를 포함하는, 상기 복수의 송신기들; 및

메모리 액세스 커맨드들을 수신하기 위한 커맨드 인터페이스를 포함하고,

상기 메모리 액세스 커맨드들은,

(i) 데이터 신호들이 상기 데이터 링크들을 통해 상기 메모리 디바이스 내에서 수신되고 상기 수신기들 내에서 증폭되는 대응하는 인터벌들을 지시 (indicate) 하는 메모리 쓰기 커맨드들, 및

(ii) 데이터 신호들이 상기 송신기들 내에서 증폭되고 상기 메모리 디바이스로부터 상기 데이터 링크들을 통해 출력되는 대응하는 인터벌들을 지시하는 메모리 읽기 커맨드들을 포함하고,

상기 제어 회로는,

상기 데이터 링크들을 통해 데이터 신호들이 수신되지 않는 제 1 인터벌 동안 상기 수신기들을 감소된 전력 모드 상태로 만들기 위한 회로, 및

상기 데이터 링크들을 통해 데이터 신호들이 출력되지 않는 제 2 인터벌 동안 상기 송신기들을 감소된 전력 모드 상태로 만들기 위한 회로를 포함하는, 집적 회로 메모리 디바이스.

청구항 2

제 1 항에 있어서,

상기 제 1 클럭 신호는 상기 메모리 제어기로부터 수신되는, 집적 회로 메모리 디바이스.

청구항 3

제 1 항에 있어서,

상기 메모리 제어기로부터 상기 메모리 디바이스로 메모리 액세스 리퀘스트들을 전달하는 하나 이상의 커맨드 신호들을 수신하기 위한 커맨드 신호 수신기를 더 포함하고,

상기 메모리 제어기로부터의 상기 제어 신호에 응답하여 상기 클럭 수신기의 동작을 디스에이블링하기 위한 상기 제어 회로는, 상기 제어 신호에 응답하여 상기 커맨드 신호 수신기의 동작을 디스에이블링하기 위한 회로를 포함하는, 집적 회로 메모리 디바이스.

청구항 4

제 1 항에 있어서,

상기 클럭 수신기는 바이어스 전류를 확립하기 위한 전류 소스를 포함하고, 상기 제어 회로는 상기 바이어스 전류의 흐름을 디스에이블링하기 위한 회로를 포함하는, 집적 회로 메모리 디바이스.

청구항 5

삭제

청구항 6

삭제

청구항 7

제 1 항에 있어서,

상기 제어 회로는 상기 제 1 인터벌 동안 상기 수신기들을 감소된 전력 모드 상태로 만들기 위해 상기 수신기들 내에서의 바이어스 전류를 디스에이블링하기 위한 회로를 포함하는, 집적 회로 메모리 디바이스.

청구항 8

삭제

청구항 9

제 1 항에 있어서,

제 1 인터벌 동안 상기 수신기들을 감소된 전력 모드 상태로 만들기 위한 상기 제어 회로는, 상기 제 1 인터벌에 대응하는 메모리 쓰기 커맨드가 수신되지 않았음을 결정할 때 상기 수신기들을 감소된 전력 모드 상태로 만들기 위한 회로를 포함하는, 집적 회로 메모리 디바이스.

청구항 10

제 9 항에 있어서,

제 2 인터벌 동안 상기 송신기들을 감소된 전력 모드 상태로 만들기 위한 상기 제어 회로는, 상기 제 2 인터벌에 대응하는 메모리 읽기 커맨드가 수신되지 않았음을 결정했을 때 상기 송신기들을 상기 감소된 전력 모드 상태로 만들기 위한 회로를 포함하는, 집적 회로 메모리 디바이스.

청구항 11

삭제

청구항 12

삭제

청구항 13

삭제

청구항 14

삭제

청구항 15

삭제

청구항 16

집적 회로 메모리 디바이스 내의 동작 방법으로서,

상기 집적 회로 메모리 디바이스의 클럭 입력을 통해 제 1 클럭 신호를 수신하는 단계;

상기 제 1 클럭 신호의 천이들에 응답하여 상기 집적 회로 메모리 디바이스로부터 데이터 신호를 출력하는 단계로서, 상기 제 1 클럭 신호는 상기 데이터 신호에서 전달된 데이터의 각 비트에 대한 각각의 천이를 갖고, 상기 데이터 신호에서 전달된 데이터의 각 비트는 각각의 비트 시간에 대해 상기 집적 회로 메모리 디바이스의 출력에서 유효하고, 상기 집적 회로 메모리 디바이스의 출력에서의 상기 데이터 신호와 상기 집적 회로 메모리 디바이스의 클럭 입력에서의 상기 제 1 클럭 신호 사이의 위상 오프셋은 적어도 상기 비트 시간 만큼 드리프트되도록 허용되는, 상기 데이터 신호를 출력하는 단계;

메모리 제어 컴포넌트로부터의 제어 신호에 응답하여 상기 제 1 클럭 신호의 수신을 디스에이블링하는 단계;

각각의 복수의 데이터 링크들을 통하여 복수의 데이터 신호 수신기들 내의 데이터 신호들을 수신하는 단계;

상기 데이터 링크들을 통해 데이터 신호들이 수신되지 않는 제 1 인터벌 동안 상기 데이터 신호 수신기들을 감소된 전력 모드 상태로 만드는 단계;

상기 데이터 링크들을 통해 데이터 신호들이 출력되지 않는 제 2 인터벌 동안 송신기 회로를 감소된 전력 모드 상태로 만드는 단계; 및

메모리 액세스 커맨드들을 수신하는 단계를 포함하고,

상기 메모리 액세스 커맨드들은,

(i) 상기 데이터 링크들을 통해 상기 메모리 디바이스 내에서 데이터 신호들이 수신되는 대응하는 인터벌들을 지시하는 메모리 쓰기 커맨드들, 및

(ii) 상기 데이터 링크들을 통해 상기 메모리 디바이스로부터 데이터 신호들이 출력되는 대응하는 인터벌들을 지시하는 메모리 읽기 커맨드들을 포함하고,

상기 제 1 클럭 신호의 천이들에 응답하여 상기 집적 회로 메모리 디바이스로부터 상기 데이터 신호들을 출력하는 단계는 상기 송신기 회로로부터 상기 데이터 링크들 중 하나를 통해 상기 데이터 신호를 출력하는 단계를 포함하는, 집적 회로 메모리 디바이스 내의 동작 방법.

청구항 17

제 16 항에 있어서,

상기 클럭 입력을 통해 상기 제 1 클럭 신호를 수신하는 단계는 상기 메모리 제어 컴포넌트로부터 상기 제 1 클럭 신호를 수신하는 단계를 포함하는, 집적 회로 메모리 디바이스 내의 동작 방법.

청구항 18

제 16 항에 있어서,

커맨드 신호 수신기를 통하여, 상기 메모리 제어 컴포넌트로부터 상기 메모리 디바이스로 메모리 액세스 리퀘스트들을 전달하는 하나 이상의 커맨드 신호들을 수신하는 단계를 더 포함하는, 집적 회로 메모리 디바이스 내의 동작 방법.

청구항 19

제 16 항에 있어서,

상기 제 1 클럭 신호를 수신하는 단계는 클럭 신호 수신기를 통해 상기 제 1 클럭 신호를 수신하는 단계를 포함하고, 상기 제 1 클럭 신호의 수신을 디스에이블링하는 단계는 상기 클럭 신호 수신기 내의 전력 소스를 디스에이블링하는 단계를 포함하는, 집적 회로 메모리 디바이스 내의 동작 방법.

청구항 20

삭제

청구항 21

삭제

청구항 22

제 16 항에 있어서,

상기 데이터 신호 수신기들을 감소된 전력 모드 상태로 만드는 단계는 상기 제 1 인터벌 동안 상기 데이터 신호 수신기들 내의 전력 소스를 디스에이블링하는 단계를 포함하는, 집적 회로 메모리 디바이스 내의 동작 방법.

청구항 23

삭제

청구항 24

제 16 항에 있어서,

상기 제 1 인터벌 동안 상기 데이터 신호 수신기들을 감소된 전력 모드 상태로 만드는 단계는 상기 제 1 인터벌에 대응하는 메모리 쓰기 커맨드가 수신되지 않았음을 결정하는 것에 응답하여 상기 데이터 신호 수신기들을 감소된 전력 모드 상태로 만드는 단계를 포함하는, 집적 회로 메모리 디바이스 내의 동작 방법.

청구항 25

제 24 항에 있어서,

상기 제 2 인터벌 동안 상기 송신기 회로를 감소된 전력 모드 상태로 만드는 단계는 상기 제 2 인터벌에 대응하는 메모리 읽기 커맨드가 수신되지 않았음을 결정하는 것에 응답하여 상기 송신기 회로를 감소된 전력 모드 상태로 만드는 단계를 포함하는, 집적 회로 메모리 디바이스 내의 동작 방법.

명세서

기술분야

- [0001] 우선권 주장
- [0002] 본 출원은 다음의 U.S. 특허 출원에 대한 우선권을 주장한다:
- [0003] (i) 2009년 1월 12일자로 출원되고, 발명이 명칭이 "4.3GB/S Mobile Memory Interface with Power-Efficient Bandwidth Scaling"인 U.S. 가특허출원 제61/144,135호; 및
- [0004] (ii) 2009년 3월 2일자로 출원되고, 발명이 명칭이 "Mesochronous Low-Power Signaling System." 인 U.S. 가특허출원 제61/156,872호
- [0005] 위의-표기된 특허 출원들은 참조에 의해 전체적으로 본원에 포함된다.
- [0006] 기술 분야
- [0007] 본 발명의 개시는 일반적으로 데이터 통신 시스템에 관한 것이고 보다 상세하게는 저-전력 어플리케이션에서의 고속 시그널링에 관한 것이다.

배경기술

- [0008] 메조크로노스 클럭 신호 (mesochronous clock signal) 는 종종 싱크로노스 (synchronous) 메모리 시스템에서 시간 시그널링 동작 (operation) 들에 사용된다. 같은 클럭 소스를 사용하여 송신/수신 타이밍을 메모리 제어기 및 메모리 디바이스 양쪽 모두 내에 제공함으로써, 주파수 드리프트 (frequency drift) 가 회피되어, 상대적으로 간단하고, 강인한 타이밍 배열을 발생시킨다. 하지만, 클럭 레퍼런스가 제어기와 메모리 사이 공간에 분배되므로, 2개 칩들의 클럭 도메인들은 일반적으로 서로에 대해 임의의 위상 오프셋 (arbitrary phase offset) 을 갖고 이는 동기식 통신을 가능하게 하기 위하여 보상되어야 한다. 클럭을 다양한 송신 및 수신 회로들로 팬 아웃 (fan-out) 하기 위해 각 칩 내에 제공되는 클럭 버퍼링 회로 (clock buffering circuitry) 에 대부분 기인하여, 복잡한 문제로, 칩-투-칩 (chip-to-chip) 위상 오프셋이 온도 및 전압에 따라 실질적으로 드리프트하는 경향이 있다.
- [0009] 많은 현대 메모리 시스템들은 스트로브 (strobe) 또는 다른 소스-싱크로노스 타이밍 신호 (source-synchronous

timing signal) 를 송신하여 수신 디바이스 내에 데이터 샘플링을 제어함으로써 칩-칩 위상 드리프트를 관리하며, 효과적으로 송신 디바이스의 클럭 도메인을 수신 디바이스로 확장한다. 불행하게도, 이 접근법은 (스트로브와 데이터 라인 사이의 전과 시간을 매칭시키기 위해) 추가의 신호 드라이브, 핀 및 정확히 라우팅된 신호 라인이 보통 요구되므로, 상당한 전력/비용 불이익을 겪는다.

[0010] 다른 접근법은 레퍼런스 클럭과 분배 클럭 사이에 정렬 (alignment) (즉 다양한 수신 및 송신 회로에 분배된 명목상 동일한-위상 클럭들의 다중도 (multiplicity)) 을 유지하기 위하여 각 메모리 디바이스 및 메모리 제어기 내에 PLL (phase-locked loop) 또는 DLL (delay-locked loop) 를 제공함으로써 드리프팅 위상 오프셋을 보상한다. 이 배열에 의해, 각각의 클럭-버퍼 지연 사이의 환경적으로 유발된 드리프트에도 불구하고 실질적으로 고정된 위상 관계가 칩들 사이에 유지될 수도 있다.

[0011] PLL/DLL 접근법은 소스-싱크로노스 배열의 많은 불이익 (특히 소중한 핀 (precious pin) 의 소모) 을 회피하지만, PLL 및 DLL 회로는 전력 소모적 (power hungry) 이고, (위상-고정을 유지하기 위하여) 심지어 유희 기간 동안에도 전력을 소비하고, 디스에이블링된, 전력-절약 상태에서부터 어웨이크닝 (awakening) 될 때 위상 고정 (phase-lock) 을 복원하기 위하여 추가의 전력과 상당한 시간을 필요로하는 경향이 있다. 모든 이들 단점들은 특히 이동 어플리케이션 (예를 들면, 셀폰, 랩톱 컴퓨터 등) 에서 특히 문제가 되고, 여기서 성능 요구 및 버스티 트랜잭션 프로파일 (bursty transaction profile) 은 고정-루프 동작을 디스에이블링하는 것을 어렵게 하고 게다가 고정-루프 회로의 큰 유희 전력은 소중한 배터리 수명을 고갈시킨다.

도면의 간단한 설명

[0012] 본원의 개시는 첨부 도면들에서 한정이 아닌 예로서 예시되고, 여기서 같은 참조 부호는 같은 엘리먼트들을 지칭한다:

- 도 1a 및 도 1b는 클럭-정지 저-전력 모드를 갖는 메모리 시스템의 일반화된 실시형태를 예시한다;
- 도 1c 및 1d는 도 1의 일시정지-가능-클럭 메모리 디바이스에서 예시적인 전력 소비 프로파일을 같은 사용 시나리오 (usage scenario) 하에서 연속적으로-클럭킹된 PLL/DLL-기반 메모리 디바이스를 위한 예시적인 전력-소모 프로파일과 대비한다;
- 도 2a는 메모리-측 및 제어기-측 I/O 회로 및 시스템 클럭킹 아키텍처의 실시형태를 보다 상세히 예시한다;
- 도 2b는 도 2a를 참조하여 설명된 메모리-측 타이밍 배열을 예시하며, 메모리 디바이스의 핀 (또는 다른 상호접속 구조) 에서 나타나는 시스템 클럭 신호 및 데이터 신호, 그리고 메모리-측 송신기에 적용된 버퍼링된 클럭 신호를 도시한다;
- 도 3a 및 도 3b는 도 2a 에서의 드리프트-보상 직병렬화기 (drift-compensating deserializer) 중 임의의 것을 구현하는데 사용될 수도 있는 드리프트-보상 직병렬화기의 실시형태 및 타이밍 도를 예시한다;
- 도 3c-도 3e 는 도 3a의 드리프트-보상 직병렬화기 내에 적용될 수도 있는 패킷-정렬 회로 (packet-alignment circuit) 의 실시형태 및 이를 조정하기 위한 방식을 예시한다;
- 도 3f 및 도 3g는 도 2a에 도시된 드리프트-보상 직렬화기 중 어느 것을 구현하는데 사용될 수도 있는 드리프트-보상 직렬화기의 실시형태 및 타이밍도를 예시한다;
- 도 3h는 도 3f의 드리프트-보상 직렬화기 내에 적용될 수도 있는 패킷-정렬 회로의 실시형태를 예시한다;
- 도 4a 및 도 4b는 직병렬화기 및 직렬화기 회로의 실시형태들을 각각 나타내고, 이들은 도 2a의 메모리 디바이스 내의 직병렬화기 및 직렬화기를 구현하는데 사용될 수도 있다;
- 도 5a는 도 2a의 실시형태 내에서 데이터 링크 (DQ0 및 DQ1) 을 위한 드리프트-보상 직병렬화기 내 수신 클럭 위상을 교정 (calibrating) 하는 것에 대한 예시적인 접근법을 예시한다;
- 도 5b는 다양한 데이터 선택 경로에 관한 상세가 없는 5a의 인트라-비트 클럭-위상 배열 (intra-bit clock-phase arrangement) 의 특정 실시형태를 예시한다;
- 도 5c는 최종 수신 클럭 위상을 결정하기 위한 접근법을 설명하며, 데이터-아이 슈무 (data-eye schmo) 에 관하여 통과-실패 바운더리 (pass-fail boundary) 및 예시적인 위상 선택기에 의해 선택된 일정 수의 클럭 위상 사이의 예시적인 관계를 나타낸다;

도 5d는 파인 (fine) 및 코스 (coarse) 데이터 아이 (data eye) 바운더리와, 주기적 타이밍 교정 동작 동안 드리프트를 트래킹 (tracking) 하는데 사용될 수도 있는 코스-데이터-아이 실패-바운더리와 파인-데이터-아이 센터 사이의 오프셋을 예시한다;

도 6a 및 도 6b는 드리프트-보상 직렬화기 교정의 예시적인 비트-정렬 (또는 패킷 프레임 조정 (packet-framing adjustment)) 스테이지를 예시한다;

도 6c는 도 2a의 다양한 시그널링 링크 내의 패킷 정렬 회로에 적용될 때, 제어기-코어 클록 도메인으로의 동시 전달을 위한 메모리 코어로부터 검색된 원래 멀티패킷 값의 부분을 형성하는 패킷들을 정렬하는 워드 레이턴시 값을 결정하기 위해 수행되는 예시적인 패킷 정렬 동작을 예시한다;

도 7a, 도 7b 및 도 8a-도 8c는 메모리 디바이스 내의 시그널링 링크의 각각의 쌍 사이의 크로스-커플링된 루프백 경로 (cross-coupled loopback path) 에 의존하는 예시적인 직렬화기 교정 절차를 예시한다;

도 9a 및 도 9b는 각각 드리프트-보상 직렬화기 및 드리프트-보상 직렬화기의 주기적 교정에 사용되는 동작들의 예시적인 시퀀스를 예시한다;

도 10a는 도 3a 내지 도 3c를 참조하여 설명된 6-비트 위상 조정 회로에 대응하는 정렬 카운터의 실시형태를 도시한다;

도 10b 및 도 10c는 주기적 타이밍 교정 회로의 실시형태 및 대응하는 상태 도를 예시한다;

도 11a는 도 2a의 실시형태 내에서 사용된 예시적인 클로킹 배열을 예시하며, 제어기 I/O 클럭과 메모리 디바이스에 포워딩되는 데이터-레이트 시스템 클럭을 위한 클럭 정지 로직을 명시적으로 나타낸다;

도 11b는 도 11a의 클럭킹 아키텍처의 클럭-정지 (또는 클럭 일시정지) 동작의 예시적인 타이밍 도이다;

도 11c 및 도 11d는 클럭-정지 로직 회로의 보다 상세한 실시형태 및 대응하는 타이밍 도를 예시한다;

도 11e-도 11g는 대안의 클럭-정지 아키텍처 및 대응하는 회로 및 타이밍 도를 예시한다;

도 12a는 클럭-정지 저 전력 모드로부터 진입 (entry) 및 종료 (exit) 를 포함하는 인터벌 동안 메모리 제어기에서의 클럭 신호, 클럭-인에이블 신호 및 커맨드/어드레스 신호의 예시적인 타이밍 도이다;

도 12b 및 도 12c는 메모리 디바이스의 관점으로부터 클럭-정지 모드 진입 및 종료를 예시한다;

도 13은 클럭-정지-인터벌로 하여금 코어 클럭 사이클의 비-진정수 (non-integral number) 에 대해 확장하도록 하는 대안의 실시형태에 따른 클럭-정지 진입 및 종료를 예시한다;

도 14a-도 14c는 메모리 제어기와 메모리 디바이스 내 코어 클럭 신호들 사이에 위상 오프셋의 조정을 가능하게 하는 위상-정렬 회로의 실시형태에 관한 것이다;

도 15a 및 도 15b는 주기적-타이밍-교정 모드에 진입 및 종료할 때 클럭 글리치 (clock glitch) 를 회피하기 위해 사용되는 예시적인 클럭-정지 동작을 예시한다;

도 16a-도 16f는 클럭 정지 없이 글리치리스 위상 점핑 (glitchless phase jumping) 을 가능하게 하는 주기적-타이밍 교정을 수행하는 대안의 방식에 관한 것이다;

도 17a는 단일 제어 IC 및 다중 메모리 IC를 갖는 일시정지-에이블-클럭 메모리 시스템의 실시형태를 예시한다;

도 17b는 도 2a에 도시된 메모리-측 I/O 인터페이스에 대응하는 인터페이스를 구현하는 모듈-장착 버퍼 IC를 갖는 일시정지-에이블-클럭 메모리 시스템의 실시형태를 예시한다;

도 18a는 도 1a 및 도 2a의 메모리 시스템 내에 이용될 수도 있는 티어드 전력 모드 (tiered power mode) 를 예시하는 예시적인 상태도이다;

도 18b는 도 2a의 실시형태에 대응하지만, 계속해서 감소된 전력 모드에서 회로 셧-다운 (shut-down) 에 관하여 추가의 상세를 나타내는 메모리 시스템 아키텍처를 예시한다.

도 18c는 감소된 전력 모드에서 디스에이블링될 수도 있는 바이어싱 회로를 갖는 차동 증폭기의 구현을 예시한다;

도 18d는 인커밍 메모리 쓰기 및 메모리 읽기 리퀘스트 각각에 대응하여 인에이블-쓰기 및 인에이블-읽기 신호

(EnW 및 EnR) 의 커맨드-기반 어썬션 (assertion) 을 예시하는 타이밍 도이다;

도 18e는 파워 다운 모드 진입 및 종료를 예시하는 타이밍 도이며, 종료는 메모리 쓰기 리퀘스트에 의해 트리거링된다.

발명을 실시하기 위한 구체적인 내용

- [0013] 메조크로노스 송신 및 수신 클럭이 메모리 액세스 트랜잭션 사이의 유휴 주기 (idle period) 동안 정지되고 재 시작될 수 있게 하는 스트로브리스 싱크로노스 메모리 시스템 (strobeless synchronous memory system) 이 수 개의 실시형태들로 개시된다. 이 동작에 의해, 유휴 주기 동안 전력 소비는 연속적으로-클럭킹된 설계에 관하여 극적으로 감소될 수도 있다. 또한, 유휴 시간은 종종 총계 (aggregate) 에서, 특히 전력-민감성 이동 디바이스에서 액티브 메모리 트랜잭션 시간 (액티브 시간) 을 훨씬 초과하기 때문에, 유휴-시간 전력 소비를 감소시키는 능력은 실질적으로 낮은 순 (net) 전력 소비를 낳을 수도 있다.
- [0014] 유휴-시간 클럭-정지 (또는 클럭 일시정지) 를 통해 이루어지는 실질적인 절력-절약에도 불구하고, 메조크로노스 시그널링 시스템에서 송신 및 수신 클럭을 정지시키는 것은 도전 (challenge) 의 캐스캐이딩 시퀀스 (cascading sequence) 를 가져온다. 우선, PLL은 일반적으로 위상-고정을 재확립하는데 지나치게 긴 시간을 필요로 하고 그 다음 심지어는 일반적으로 신뢰성있는 데이터-레이트 시그널링이 시작될 수도 있기 전에 위상 교정이 완료될 것을 요구하는 비교정 상태에서 재-고정하게 되므로, 메모리-측 PLL에서의 위상-고정의 손실은 즉각적인 성능 문제를 나타낸다. 그리고 게다가, 메모리-측 PLL의 제거는 압도적인 세트의 문제들을 나타내며, 메모리 디바이스 내의 광범위한 환경적으로-유발된 위상 드리프트와 송신 및 수신 클럭킹을 위해 메모리 디바이스 내에서 필요한 크리티컬 타이밍 에지의 손실로 시작된다. 즉, 관습적으로 온-메모리 PLL (on-memory PLL) 은 온도/전압-유발된 위상 드리프트를 보상하고 상대적으로 저-주파수 시스템 클럭의 주파수 (또는 위상의 수)를 체배 (multiplying) 함으로써 데이터-레이트 시그널링에 필요한 타이밍 에지를 제공하는 이중 기능을 수행한다.
- [0015] 이들 도전에도 불구하고, PLL/DLL 회로는 본원에 개시된 실시형태들에서의 메모리-디바이스 클럭킹 아키텍처로부터 생략되고 메모리 디바이스 타이밍 도메인의 위상은 메모리 제어기 타이밍 도메인에 관하여 자유롭게 드리프트하도록 허용된다. 또한, 복잡한 드리프트-보상 회로로 메모리 디바이스를 방해하는 대신, 메모리-제어기와 메모리-디바이스 타이밍 도메인 사이의 드리프팅 위상 오프셋이 메모리 제어기 내 회로에 의해 보상된다. 아래에 논의된 바처럼, 온-메모리 PLL의 부재시, 메모리-디바이스 위상 드리프트는 유닛 인터벌 (즉, 데이터 시그널링 레이트 또는 데이터 레이트의 역 (inverse), 및 비트 또는 심볼 송신에 배당된 시간 인터벌; 유닛 인터벌은 또한 여기서 비트 시간 또는 심볼 시간으로 지칭된다) 을 훨씬 넘어 확장될 수도 있으며, 타이밍 보상 노력 및 클럭 시작/정지 조정 (coordination) 에 실질적인 복잡성을 추가한다.
- [0016] 메모리-측 PLL/DLL의 생략 및 관습적인 온-메모리 PLL 기능들의 제 2 (second) 의 부수적인 손실-상대적으로 낮은 주파수 시스템 클럭 신호로부터의 데이터-레이트 타이밍 신호의 생성-은 시스템 클럭킹 배열 자체의 변화에 의해 저지된다. 보다 상세하게는, 저 주파수 시스템 클럭을 분배하고 그 다음 이 저 주파수 시스템 클럭이 온-메모리 PLL/DLL에 의해 주파수-체배 (또는 위상 분배) 되어 데이터-레이트 타이밍 에지를 제공해야 하는 것 대신에, 데이터-레이트 클럭 신호 그 자체가 시스템 클럭 신호로서 분배되어, 그에 의해 메모리 디바이스 내 주파수-체배 (또는 위상 분배) PLL/DLL 회로에 대한 필요성을 회피한다. 이 접근법은 더 높은-주파수 클럭의 온-칩 분배 및 송신에 수반된 잠재적으로 더 높은 전력 소비를 겪지만, 메모리-측 PLL/DLL의 생략은 관습적인 설계를 괴롭히는 고정 손실 (loss-of-lock) 고려사항들 (considerations) 을 제거하고, 비한정적인 예로서, 본원에 설명된 클럭-정지/시작 관리 회로 및 드리프트 보상 회로와 결합될 때, 무시가능한 성능 불이익으로 고속으로 진입되고 종료될 수도 있는 클럭-정지 저-전력 모드를 가능하게 한다. 궁극적으로, 버스티 메모리 액세스 프로파일 (예를 들면 셀폰 및 다른 이동 디바이스에서처럼, 액티브 메모리 액세스의 상대적으로 짧은 주기 중에 산재된 빈번한 유휴 주기) 을 나타내는 어플리케이션들에 대하여, 유휴-시간 전력 절약은 증가된 액티브-시간 전력 소비를 크게 증가하는 경향이 있고; 절약은 시스템에 있는 메모리 디바이스의 수만큼 배가된다.
- [0017] 도 1a 및 도 1b는 클럭-정지 저-전력 모드를 갖는 메모리 시스템 (100) 의 일반화된 실시형태를 예시한다. 메모리 시스템은 메모리 제어기 (101) 및 메모리 디바이스 (103)를 포함하고 이들은 시그널링 링크 (102) 및 시스템 클럭 링크 (108) 를 통해 서로 커플링된다. 메모리 제어기 자체는 제어기 코어 (105) 및 입력/출력 (I/O) 인터페이스 (107) (또는 PHY; 물리적 인터페이스) 를 포함하고 메모리 디바이스는 유사하게 메모리 코어 (131) 및 I/O 인터페이스 (133) 를 포함한다. 메모리 디바이스 및 메모리 제어기 내 I/O 인터페이스 (즉, "메모리-측" 및 "제어기-측" I/O 인터페이스) 는 하나 이상의 데이터 링크 (106) 를 통해 양-방향 데이터 전송

및 하나 이상의 커맨드/어드레스 (CA) 링크 (104) 를 통해 일-방향 커맨드 (또는 리퀘스트 또는 명령) 을 지원 하는 시그널링 회로 (117, 119, 137, 139) 를 포함한다. 제어기-측 I/O 인터페이스는 또한, 클럭 링크 (108) 을 통해 메모리 디바이스로 포워딩되고 클럭 버퍼 (135) 및 내부 클럭 경로 (136) 를 통해 메모리-측 시그널링 회로 (137, 139) 에 분배되는 시스템 클럭 신호 (시스템 클럭, SCK) 를 생성하는 클럭 생성기 (115) 를 포함한다. 클럭 생성기는 또한, 내부 클럭 경로 (110) 을 통해 제어기-측 시그널링 회로 (117, 119) 로 분배되는 제어기-측 클럭의 세트를 생성한다.

[0018] 메모리 디바이스 (103) 를 참조하면, 메모리 코어 (131) 은 하나 이상의 बैं크 (bank) 로 배열된 코어 저장 어레이 (132), 및 메모리 제어기로부터의 메모리 액세스 커맨드 및 어드레스에 응답하여 코어 저장 어레이에 대한 읽기 및 쓰기 액세스를 관리하기 위한 액세스 회로 (134) 를 포함한다. 아래에 설명된 실시형태들에서, 코어 저장 어레이는 데이터 손실을 피하기 위하여 때때로 리프레쉬 (refresh) 를 필요로 하는 DRAM (dynamic random access memory) 로 가정되지만, 사실상 임의의 저장 기술이 대안의 실시형태들에서 사용될 수도 있는데, SRAM (static random access memory) 및 다양한 형태의 비휘발성 메모리 (예를 들면, 플래시 메모리, 위상-변화 메모리 등) 을 포함하지만 이에 한정되지는 않는다. 사용된 저장 기술에 상관없이, 커맨드 링크 (104) (집합적으로 "커맨드 경로") 를 통해 메모리 디바이스에 전달되는 커맨드 및 어드레스 값 (커맨드/어드레스 또는 CA 값) 은 코어 저장 어레이 (132) 의 어드레스-특정 지역 내 데이터 검색 (메모리 읽기) 및 데이터 저장 (비휘발성 셀 프로그래밍을 포함하는, 메모리 쓰기) 동작을 수행하기 위해 사용된다. 검색된 데이터는 본원에서 "읽기 데이터" 로 지칭되고 데이터 링크 (106) (집합적으로, "데이터 경로") 를 통해 메모리 제어기로 리턴되고, 저장되거나 프로그래밍될 데이터 ("쓰기 데이터") 는, 반대로, 데이터 경로를 통해 메모리 제어기로부터 제공된다. 일부의 경우, 데이터-리스 커맨드 (data-less command), 이를 테면, (코어 저장 어레이 내의 저장 셀로부터 래칭 센스-증폭기 बैं크 (latching sense-amplifier bank) 로의 데이터 전송을 명령하는) 로우-액티베이션 커맨드 (row-activation command), 리프레쉬 커맨드 (refresh command), (예를 들면, 플래시 또는 다른 전기-이레이저블 비휘발성 메모리의 경우) 이레이즈 커맨드 (erase command) 및 다양한 구성 커맨드 및/또는 동작-모드 커맨드는 커맨드 경로를 통해 이슈 (issue) 될 수도 있다.

[0019] 도 1의 실시형태를 숙고하면, 메모리-측 클럭킹 배열의 다수의 특징들이 강조된다. 먼저, 클럭 버퍼 (135) 로부터 출력된 클럭 신호 (즉, 버퍼링된 클럭 신호) 는 시스템 클럭 신호의 위상-지연된 예이다; 주파수 체배 또는 다중-위상 클럭 생성이 메모리 디바이스 내에서 일어나지 않아서 시스템 클럭 신호의 주파수 자체가 데이터 송신 및 샘플링 레이트를 메모리-측 I/O 회로 내에 확립하고, 따라서 샘플링 레이트를 시그널링 링크 (102) 를 통해 확립한다. 따라서, 저 주파수 시스템 클럭을 분배하고 PLL/DLL 회로를 제공하여 클럭 주파수를 체배하거나 추가의 클럭 위상을 생성하는 것에 의해 데이터-레이트 클럭 신호를 생성하는 것과 반대로, 데이터-레이트 클럭 신호 자체 (즉, 데이터 링크를 통해 송신된 각 심볼에 대한 각각의 타이밍 예지를 포함하는 클럭 신호) 는 시스템 클럭 신호로서 메모리 디바이스에 공급된다. 이러한 접근법의 하나의 결과는 바람직한 이득을 이루기 위하여 클럭 버퍼 (135) 를 형성하는 증폭기들의 체인 (chain) 에서 추가의 버퍼 증폭기들이 요구될 수도 있어 (즉, 이득이 주파수에 따라 강해지는 경향이 있어, 추가의 이득 스테이지들이 더 높은 클럭 주파수에서 요구될 수도 있다), 저-주파수, 멀티-위상 클럭 신호의 분배와 반대로, 메모리 디바이스 전체에 걸쳐 데이터-레이트 클럭 신호를 분배하기 위해 추가의 전력을 필요로 한다. 위에 논의된 바와 같이, 종래의 클럭 분배 배열을 더 많은 전력을 소모할 수도 있는 것으로 교환하는 추정적 불이익에도 불구하고, 주파수-체배 PLL/DLL의 생략은 위상 고정-재획득과 연관된 보통 시간-지연 불이익을 발생시키지 않고서 액티브-모드 클럭 상태와 저-전력-모드 클럭-정지 상태 사이의 고속 천이를 가능하게 한다. 따라서, 무시가능한 성능 영향으로 (메모리 액세스 활동의 버스트 사이의) 상대적으로 짧은 유휴 주기 동안에도 클럭-정지 저 전력 모드가 진입될 수도 있다. 많은 어플리케이션에서 총계의 유휴 시간 (aggregate idle time) 이 총계의 액티브 메모리 액세스 시간 (aggregate active memory access time) 을 훨씬 초과하므로, 액티브-시간 전력에서의 약간의 증가를 희생하여 유휴 시간 동안의 실질적인 전력 감소가 전력 소비의 실질적인 순 감소를 낳을 수도 있다. 이 결과는 도 1c 및 도 1d에 그래프로 예시되어 있는데, 이들은 도 1의 일시정지-에이블-클럭 메모리 디바이스에서의 예시적인 전력 소비 프로파일을 동일한 사용 시나리오 (usage scenario) 하에서의 연속적으로-클럭킹된 PLL/DLL-기반 메모리 디바이스에 대한 예시적인 전력 소비 프로파일과 대비한다. 도시된 바처럼, 일시정지-에이블 클럭 메모리에서의 다소 더 높은 액티브-시간 전력에도 불구하고, 실질적으로 감소된 유휴-시간 전력 소비는, 메모리-측 타이밍 도메인을 시스템 클럭 신호의 위상으로 앵커링 (anchoring) 하기 위해 이용된 온-메모리 고정-루프에서의 큰 유휴 전력 소비를 겪는 연속적으로-클럭킹된 메모리에서 보다 훨씬 낮은 순 전력 소비를 낳는다.

[0020] 메모리-측 클럭킹 배열의 다른 특징은 클럭 분배 회로가 메모리 디바이스 내에서 완전히 개방된 루프라는 것이

다; 논의된 바처럼, 메모리-측 I/O 셀로 분배된 버퍼링된 클럭 신호와 시스템 클럭 신호 사이의 시간 변화 (즉, 드리프트) 위상 지연을 보상하기 위한 고정-루프 회로가 존재하지 않는다. 게다가, 시스템-클럭-버퍼링된 클럭 위상 지연의 규모 및 환경적 민감도 양쪽 모두는 클럭 버퍼 내에 제공된 증폭의 추가 스테이지들에 의해 증가되어 더 높은 주파수 데이터-레이트 클럭 신호를 처리한다. 즉, 클럭 버퍼 내의 각 증폭기 스테이지는 환경-의존성 (예를 들면, 온도-의존성 및/또는 전압-의존성) 전파 지연을 나타내는 경향이 있어서, 증폭기 스테이지의 추가는 순 시스템-클럭-버퍼-클럭 타이밍 스큐 (net system-clock-to-buffered-clock timing skew) 를 증가시킬 뿐만 아니라, 타이밍 스큐의 변경의 레이트 (즉, 드리프트 레이트) 를 증가시킨다. 버퍼링된 클럭 신호는 메모리-측 I/O 셀 내에서 시간 샘플링 및 송신 동작에 적용되기 때문에, 버퍼링된 클럭 신호의 드리프트 위상은 메모리 디바이스에 의해 송신된 읽기 데이터 신호들의 대응하는 위상 드리프트 (및 그러한 신호가 정확하게 수신되면 인커밍 읽기 데이터 신호에서 위상의 필요한 변화) 로서 표명된다. 마지막으로, 클럭 버퍼 지연은 수 비트 시간 정도일 수도 있고 온도 및 전압 코너 사이(즉, 최소 및 최대 허용가능 전압 및 온도 사이) 클럭 버퍼 지연에서의 순 변화는 용이하게 심볼 시간 (또는 비트 시간) 초과할 수도 있기 때문에, 송신 또는 수신 클럭 위상은 하나 이상의 비트 시간 바운더리를 지나 인접 비트 시간으로 드리프트할 수도 있다. 이것은 추가의 타이밍 복잡성을 생성하는데, 데이터 샘플링 시간이 비트 바운더리 사이에 적절히 센터링될 수도 있지만, 하나 이상의 전체 비트 시간에 의해 오프될 수도 있기 때문이다. 따라서, 그와 달리 정확히 수신된 데이터는 수신기-측 직렬화 회로에 의해 (본원에서 패킷으로 지칭되는) 데이터 비트들의 병렬 세트에 부적절하게 프레임될 수도 있다.

[0021] 메모리 디바이스 내의 클럭 분배 배열은 개방 루프일지라도, 그럼에도 불구하고 시스템-와이드 폐쇄 루프 타이밍 보상 구조는, 메모리 제어기와 메모리 디바이스 사이의 송신의 관점에서 수행되는 교정 동작 동안 위상, 비트 및 패킷 정렬 정보의 획득을 통하여 이루어짐에 유념해야 한다. 따라서, 멀티-컴포넌트 (멀티-IC) 폐쇄 루프는 시스템 클럭 신호의 메모리 디바이스로의 포워딩, 및 제어기-관리 타이밍 교정 동작을 통해 (메모리-측 송신 및 수신 회로에 적용되는) 포워딩된 클럭 신호의 메모리-측 위상을 나타내는 정보의 획득에서, 이루어진다.

[0022] **클럭-정지 저 전력 모드**

[0023] 도 1a를 계속해서 참조하면, 제어기 코어 (105) 는 (예를 들면, 프로세서 또는 다른 메모리 액세스 리퀘스터로부터) 호스트 인터페이스를 통해 수신된 메모리 액세스 리퀘스트를 큐잉하기 위한 트랜잭션 큐 (109) (또는 리퀘스트 큐), 및 트랜잭션 큐의 상태를 모니터링하는 전력-모드 제어기 (111) 를 포함한다. 트랜잭션 큐가 빈 상태가 될 때, 전력 모드 제어기는 최종 (즉, 마지막 디큐잉 (dequeuing) 된) 트랜잭션의 완료 전에 추가의 트랜잭션 리퀘스트가 수신 (및 큐잉) 되는지에 따라, 전력-모드 제어기는 저-전력 클럭-정지 모드에 진입할 준비를 하게 된다. 최종 트랜잭션을 완료하기 전에 추가의 트랜잭션 리퀘스트가 수신되지 않으면, 전력 모드 제어기는 클럭-인에이블 신호 (114) 를 디어써트 (deassert) 하여 (또는 일시정지 신호를 어써트 (assert) 하여) 시스템 클럭의 토글링 (toggling), 그리고 바람직하게는 (필요하지는 않지만) 제어기-측 시그널링 클럭을 서스펜딩 (suspending) 시킨다. 언어지는 클럭 정지 또는 클럭 일시정지는 메모리 디바이스 및 메모리 제어기 내에서 즉각적인 전력 절약을 낳는데, 메모리 측 및 제어기 측 I/O 회로들 내부의 모든 송신 및 수신 클럭이 토글링을 정지하고 따라서 쌍안정 (bi-stable) 로직 상태 사이의 전력-소비 범위를 통해 클럭 회로 (clocked circuitry) 를 구동하는 것을 피하게 되기 때문이다.

[0024] 도 1b는 클럭-정지 효과를 예시한다. 최종 메모리 트랜잭션이 클럭 사이클 "0" 에서 시작된다고 가정하면, 전력-모드 제어기는 빈 트랜잭션 큐를 주목하고 메모리 디바이스 및 제어기-측 I/O 회로의 내부 동작이 완료되는 시간까지 클럭 사이클의 카운팅을 시작한다. 이 예에서, 그 시간은 트랜잭션이 시작된 후 24 시스템-클럭 사이클에서, 그리고 따라서 시스템 클럭 사이클 24에서 일어난다. 그 직후, 이 경우에, 최종 비동작 (NOP) 커맨드의 메모리 디바이스로의 송신을 보장할 만큼 길게, 시스템 클럭 및 제어기 I/O 클럭들은 깨끗이 정지되고 로직 하이 (high) 또는 로우 (low) 상태로 남는다. 이 시점에서, 메모리 시스템은 유휴 상태이고 클럭-정지 저 전력 상태에 있다. 제어기 코어 내의 저-주파수 클럭은 계속 진동하고 따라서 나중에-제출된 트랜잭션 리퀘스트의 수신을 허용한다. 이 예에서, 트랜잭션은 시스템 클럭 사이클 44 직전 어느 시점에 큐잉된다. 따라서, 전력 모드 제어기는, 큐잉된 트랜잭션을 검출하며, 클럭 사이클 44에서 시그널링 클럭 (시스템 클럭 및 제어기-측 I/O 클럭) 을 재시작하고 비동작 커맨드를 메모리 디바이스에 전송될 수 있게 하고, 그 후에 코어 저장 어레이의 선택된 뱅크 (B) 로 지시된 액티브화 커맨드로서, 이 예에서, 도시된 액티브 커맨드 전송을 허용한다. 따라서, 전력-모드 제어기는 빈 트랜잭션 큐를 검출하는대로 메조크로노스 시그널링 클럭을 정지시키고 최종 트랜잭션이 완료될 만큼 길게 대기함으로써 메모리 액세스 트랜잭션 사이의 유휴 주기에서

의 전력 소비를 감소시키고, 새로 큐잉된 트랜잭션을 검출하는대로 시그널링 클럭을 재시작한다. 이 예에서, 클럭-정지 인터벌은 그렇지 않았으면 시스템 클럭 신호의 16회 사이클이 될 것을 넘어 확장되어, 그 시간 동안 전체 시스템 전력 소비를 현저히 낮춘다. 이 실제 어플리케이션에서, 심지어 수 밀리초의 유휴 주기 동안 시그널링 클럭을 정지시키는 것은 그렇지 않았으면 수백만 클럭 천이에 요구되는 전력 소비를 피하게 된다. 총계로 액티브 메모리 트랜잭션 시간을 실질적으로 초과하는 다수의 유휴 주기 동안 그 절약을 축적하면, 무시가능한 성능 불이익으로 실질적인 전력 절약을 낳게 된다.

[0025] 클럭킹 및 드리프트 보상

[0026] 도 2a는 메모리-측 및 제어기-측 I/O 회로 및 시스템 클럭킹 아키텍처의 실시형태를 보다 상세히 예시한다. 한정이 아닌 명료성을 위하여, 특정 수 및 타입의 시그널링 링크, 클럭 주파수 및 주파수 비, 및 직렬화 깊이 (serialization depth) 는 도 2a 및 이어지는 관련 도들에 도시되어 있다. 예를 들면 차동 시그널링 링크는 8개 데이터 링크 (DQ[0-7]), 2개 커맨드/어드레스 링크 (CA[0,1], 데이터 마스크 링크 (DM) 및 시스템 클럭 링크 (SCK) 각각을 구현하기 위해 제공되는 한편, 싱글-엔드형 링크 (single-ended link) 는 상대적으로 낮은 시그널링-레이트 측-밴드 링크 쌍 (SL [0,1]) 을 구현하기 위해 사용된다. 차동 링크들의 각각은 다르게는 싱글-엔드형 링크일 수도 있고 (및 그 역 또한 같다), 보다 많거나 보다 적은 링크들이 커맨드 경로 및/또는 데이터 경로를 구현하기 위해 사용될 수도 있고, (단일방향 커맨드 경로의 부분으로 고려될 수도 있는) 데이터 마스크 링크 및 연관 회로는 모두 생략될 수도 있다. 전용 측-밴드 링크는 또한 데이터 또는 커맨드 링크 중 하나 상의 아웃-오브-밴드 시그널링 (out-of-band signaling) 에 유리하게 생략될 수도 있다.

[0027] 클럭 주파수 및 비에 관하여, 시스템 클럭킹 아키텍처는 400 MHz 레퍼런스 클럭 신호 (REFCK1) 에 의해 구동되고 이는 PLL 회로 (161) 내에 8에 의해 체배되어 본원에서 다르게는 PCK8 또는 제어기-측 I/O 클럭 ("PCK8"에서의 "8"은 레퍼런스 클럭 주파수의 8× 체배를 나타낸다) 으로 지칭되는 3.2GHz 제어기-측 I/O 클럭 신호의 위상 분배 세트를 생성한다. 제어기-측 I/O 클럭을 구동하는 것에 더하여, 3.2GHz PLL 출력은 분할기 (165) 에서 2로 분할되어 시스템 클럭, SCK (또한 본원에서 PCK4로 지칭됨) 을 생성하고, 분할기 (163) 에서 8로 분할되어 시스템 클럭, 및 제어기-측 I/O 클럭에 위상 정렬되지만, 코어를 클럭킹하기 위해 감소된 주파수를 갖고 따라서 저-전력 로직 동작을 가능하게 하는 제어기-측 코어 클럭 신호 (PCK1) 을 발생시킨다. 모든 그런 경우에 있어서, 코어와 I/O 타이밍 도메인 사이에 다른 클럭 주파수 및 주파수 비가 사용될 수도 있다. 또한 동일한 주파수 클럭킹이 각 시그널링 링크에 관해 사용될지라도, 다르게는 상이한 I/O 클럭킹 주파수가 적용되어 상이한 클래스의 신호들에 대한 상이한 시그널링 레이트 (예를 들면 커맨드/어드레스 신호들의 하프-데이터-레이트 클럭킹) 를 이룰 수도 있다. 또한, 도시된 구현에서, 1.6GHz 시스템 클럭 주파수는 데이터 및 커맨드 링크에 대한 3.2Gb/s (초당 기가비트) 시그널링 레이트의 절반이다. 비록 때때로 본원에서 "하프 비트-레이트" 또는 "하프 심볼-레이트" 클럭 신호로 지칭되지만, 그럼에도 불구하고 시스템 클럭은 "데이터-레이트" 클럭 신호로 고려되는데, 각 사이클 내의 상승 및 하강 에지 (또는 차동 시스템 클럭 구현에서의 상보 신호들의 2개 180° -오프셋 상승 에지) 가 각각의 (1/3.2GHz) 데이터 인터벌에서 데이터를 송신 또는 샘플링하는데 사용될 수도 있기 때문이다. 비록, 이어지는 다수의 예시적인 실시형태들에서 하프-비트-레이트 (하프-심볼-레이트) 시스템 클럭이 포워드 (forward) 전달되지만, 다르게는 풀-비트-레이트 클럭 (이 예에서 3.2GHz) 가 시스템 클럭으로서 메모리 디바이스로 포워딩될 수도 있다.

[0028] 계속해서, 8-1 직렬화는 각 시그널링 링크를 통해 비트-직렬 송신을 위한 정보의 코어-공급 8-비트-와이드 패킷을 직렬화하기 위해 적용되고 대응하는 1-8 직병렬화 (deserialization) 는 카운터파트 코어 (counterpart core) 로의 전달을 위해 8-비트-와이드 데이터로 직렬 비트 시퀀스를 복원 (restore) 하기 위해 적용된다. 예를 들면, 8개 8-비트 패킷의 쓰기 데이터 (Wdata[0][0-7]-Wdata[7][0-7]) 은 400MHz 제어기 코어 클럭 (PCK1) 의 각 주기 동안 직렬화되고 각각의 8-비트 시퀀스에서 3.2Gb/s 데이터 레이트로 각 8개 데이터 링크, DQ[0-7]를 통해 송신되어, 따라서 3.2GB/s (초당 3.2기가바이트) 의 총 데이터 대역폭을 제공한다. 메모리 디바이스에서, 8-비트-길이 쓰기 데이터 패킷의 각각은 (비트-바이-비트 (bit-by-bit)) 샘플링되고 400MHz 메모리 코어 클럭 (MCK1) 의 사이클 시간 동안 병렬 패킷으로 변환되며, 따라서 제어기 코어와 같은 메모리 코어가 낮은 주파수 도메인에서 바이트-크기 패킷 (byte-sized packet) 의 데이터에 대해 동작되는 것을 가능하게 한다. 메모리 디바이스 내의 역 직렬화 (converse serialization) 및 메모리 제어기 내의 직병렬화는 메모리 디바이스로부터의 메모리 제어기로의 읽기 데이터 송신에서 수행되며, 따라서 메모리 코어로부터 제어기 코어로의 상대적으로 협소한, 8-링크 데이터 경로를 통해 3.2GB/s 데이터 전송을 가능하게 하는 한편, 양쪽 모두의 디바이스 코어들이 상대적으로 낮은 주파수 클럭 도메인 (이 예에서 400MHz) 에서 동작가능하게 한다. 비슷한 직렬화 및 직병렬화 동작은 커맨드/어드레스 링크 및 데이터 마스크 링크의 각각에 대해 단일 방향으로

수행된다. 모든 그러한 경우에서, 다른 직렬화 깊이 (즉, 패킷당 더 많거나 더 적은 비트) 는 (효과적으로 직렬화 또는 직병렬화를 포함하지 않고; 깊이=1을 포함하는) 링크의 어느 또는 모두에 대해, 일반적으로 코어-I/O 클러킹 비의 대응하는 변화를 이용하여 적용될 수도 있다.

[0029] **개방-루프 메모리-측 클럭 분배를 이용한 메조크로노스 클러킹**

[0030] 모든 시스템 타이밍 에지들은 공통 클럭 신호 (즉, 레퍼런스 클럭 신호, REFCK1으로부터 자체 도출되는, PLL의 출력) 로부터 도출되기 때문에, 시스템 내 다양한 클럭은 메조크로노스하다. 즉, 다양한 클럭들이 임의의 체배/분할을 처리한 후 동일한 주파수를 갖지만, 클럭이 메모리 제어기 및 메모리 디바이스 내 어플리케이션의 다양한 포인트들에 도달하는데 요구되는 상이한 전파 시간에 기인하여 잠재적으로 상이한 위상을 가진다. 일반적으로, 온-다이 (on-die) 또는 인터-칩 (inter-chip) 도체를 통한 그러한 전파 시간은 동작 시스템 온도 및 전압 범위에 대해 상대적으로 일정하게 남는다. 하지만, 메모리 제어기 및 메모리 디바이스 내 클럭 라인들을 구동하기 위해 제공된 버퍼 증폭기와 같은 액티브 컴포넌트를 통한 전파 시간은 환경적인 변화 (적어도, 온도 및 전압) 에 현저히 영향을 받고 따라서 그렇지 않았으면 다양한 분배 클럭들 사이에서 상대적으로 정상 (steady) 위상 관계 사이의 환경적으로 유발된 드리프트를 낳는 경향이 있다.

[0031] 메모리-측 클러킹 아키텍처를 참조하면, 특히 시스템 클럭이 버퍼 (223) 을 통해 수신되고 글로벌 클럭 라인 (230) 상으로 증폭기 (229) 에 의해 구동된다. 글로벌 클럭 라인을 구동하는데 필요한 상대적으로 큰 이득 때문에, 증폭기 (229) 는 다중 스테이지들을 포함하고, 이들의 각각은 상당한 환경적으로-민감한 전파 지연을 나타내는 경향이 있다. 시스템 클럭의 상대적으로 높은 주파수 (즉, 클럭은 온-메모리-PLL 설계의 낮은 시스템 클럭 주파수와 반대로, 워스트-케이스 데이터 신호 (worst-case data signal) 와 같은 상부 스펙트럼의 컴포넌트를 갖는다) 는 일반적으로 이 환경적 민감도 (sensitivity) 를 증가시키는데, 추가의 증폭기 스테이지들이 원하는 신호 이득을 이루는데 필요할 수도 있기 때문이다 (즉, 이득은 일반적으로 주파수가 증가함에 따라 롤링 오프한다). 따라서, 본원에서 메모리-측 I/O 클럭, 또는 MCK4 로도 지칭되는, 결과적인 버퍼링된 클럭 신호는 인커밍 시스템 클럭 신호에 관한 상당한 위상 지연을 나타낼 뿐만 아니라, 메모리 디바이스의 온도 및 전압 동작 범위에 대해 하나 이상의 유닛-인터벌 (비트 시간) 을 초과하는 드리프트를 발생시킬 수도 있는 환경적인 민감도를 나타낸다. 또한, 온-라인 PLL/DLL의 피드백 루프에서의 클럭 버퍼를 포함하는 것에 의해 드리프팅 증폭기 지연을 보상하는 전통적인 설계와는 대조적으로, 증폭된 시스템 클럭 신호 (즉, 버퍼링된 클럭 신호, MCK4) 의 개방-루프 분배는 클럭 증폭기 내 임의의 위상 드리프트가 메모리-측 송신 및 수신 클럭에서의 위상 드리프트로 직접 트랜스레이팅되고 따라서 메모리 디바이스에 의해 송신된 읽기 데이터 신호의 대응하는 위상 드리프트 (그리고 그러한 신호가 정확히 수신될 경우 인커밍 쓰기 데이터 신호에서 위상의 필요한 변화) 로서 표명됨을 의미한다. 마지막으로, 클럭 버퍼 지연 (즉, 엘리먼트 (223, 229) 를 통한 지연) 은 수 비트 시간 정도일 수도 있고 온도 및 전압 코너 사이(즉, 최소 및 최대 허용가능 전압 및 온도 사이) 클럭 버퍼 지연에서의 순 변화는 용이하게 비트 시간을 초과할 수도 있기 때문에, 송신 또는 수신 클럭 위상은 하나 이상의 비트 시간 바운더리를 지나 인접 비트 시간으로 드리프트할 수도 있다. 이것은 추가의 타이밍 복잡성을 생성하는데, 데이터 샘플링 시간이 비트 바운더리 (데이터 아이의 에지) 사이에 적절히 센터링될 수도 있지만, 정수의 비트 시간에 의해 오프될 수도 있기 때문이다. 결과로서, 그와 달리 정확히 수신된 데이터는 메모리-측 또는 제어기-측 직병렬화 회로에 의해 데이터 비트들 (예를 들면, 8-비트 패킷, 16-비트 패킷 등)의 병렬 패킷으로 부적절하게 프레임될 수도 있다.

[0032] 도 2b는 위에서 설명된 메모리-측 타이밍 배열을 예시하며, 도 2a의 메모리 디바이스의 핀 (또는 다른 상호접속 구조) 에서 나타나는 것처럼 시스템 클럭 신호 및 데이터 신호, 그리고 그리고 메모리-측 직렬화기 (235) (또는 싱글-비트 송신기) 에 적용된, 버퍼링된 메모리 I/O 클럭, MCK4을 도시한다. 도시된 바처럼, 메모리 I/O 클럭은 시스템 클럭에 관하여 시간-변화 지연을 나타내어, 메모리 I/O 클럭의 위상 및 따라서 데이터 링크 (DQ) 중 하나 상으로 구동된 읽기 데이터 신호의 위상은 시스템 클럭 신호에 관하여 자유롭게 드리프트한다. 보다 상세하게는, 시스템 클럭과 메모리 I/O 클럭 사이의 제 1 시간 지연 (또는 위상 오프셋) 은 제 1 전압 및 온도 포인트 (v0, t0) 에서 일어나고, 온도와 전압이 새로운 포인트 (v1, t1) 및 (v2, t2) 까지 시간에 대해 드리프트함에 따라, 시스템-클럭 대메모리-I/O-클럭 위상 오프셋은 뒤로 (drift-) 그리고 앞으로 (drift+) 일 비트 시간 이상 만큼 드리프트한다. 또한, 메모리 I/O 클럭의 예 및 싱글 데이터 링크 상의 위상 드리프트가 도시되어 있지만, 도시된 것으로부터 규모 및 방향에 있어 독립된, 비슷한 위상 드리프트가 타 데이터 링크에서 귀속되어 있을 수도 있다. 예를 들면, 시스템 클럭 신호에 대한 위상 드리프트는 데이터 링크에서 데이터 링크로 변할 수도 있는데, 예를 들면 각 시그널링 링크에 연관된 환경적으로-민감한 로컬 클럭 버퍼 및 그들이 도입할 수도 있는 잠재적으로 다른 전파 지연들에 기인한다.

[0033] 제어기-측 직렬화기/직병렬화기 회로 내 드리프트 보상

[0034] 도 2a의 실시형태에서, 타이밍 보상 회로는 제어기-측 직렬화기/직병렬화기 회로와 함께 제공되어 메모리-측 I/O 회로 내 자유롭게 드리프트하는 송신 및 수신 클럭 위상을 보상한다. 보다 상세하게는, 타이밍 보상 회로는 링크 바이 링크 기초 상에서 드리프트하는 메모리-측 I/O 타이밍 도메인으로 제어기-측 I/O 타이밍 도메인을 정렬하며, 인트라-비트 샘플링 위상 에러 뿐만 아니라, 메모리-측 위상 드리프트가 비트 바운더리를 교차할 때 발생하는 비트-시간 오정렬, 그리고 다양한 링크에서 상이한 비트 시간 오정렬에 의해 야기되는 링크-링크 패킷 오정렬을 보상한다. 사실상, 타이밍 보상 회로는 드리프트-트래킹 (drift-tracking) 송신 및 수신 클럭 위상을 각 제어기-측 I/O 회로 내에서 확립하고 이는 카운터파트 메모리-측 I/O 회로에서 수신 및 송신 클럭의 위상 드리프트를 보상하는데, 그렇지 않으면 메모리 제어기 또는 메모리 디바이스 중 어느 하나 내 I/O 회로와 코어의 클럭 도메인 사이에서 패킷이 전송될 때 도메인 교차 에러와 데이터 직렬화/직병렬화 에러 (즉, 시그널링 링크의 반대 측 상의 상이한 비트 바운더리에서의 패킷으로 비트를 프레임하는 것) 를 발생시킬 수도 있는 비트 바운더리를 가로지르는 드리프트를 포함한다.

[0035] 도 2a의 실시형태에서, 각 드리프트-보상 직병렬화기는 위상-선택 직병렬화기 (192) 를 포함하여 인트라-비트 위상 드리프트를 보상하고, 패킷/비트 정렬 회로 (194) 를 여기에 포함하여 비트 바운더리 (비트 정렬) 을 가로지르는 드리프트를 보상하고 제어기-코어로의 동기화 전송 (패킷 정렬) 을 위해 상이한 링크를 통해 수신된 패킷을 정렬한다. 드리프트 보상 직렬화기는 유사한 회로를 포함하여 메모리 디바이스로 흐르는 정보의 타이밍을 조정하며, 인트라-비트 조정(위상-선택 직렬화기 (191)), 및 비트/패킷 정렬 (193) 을 제공하여, 메모리 디바이스 내 링크-링크 패킷 정렬, 비트-프레이밍 및 적절히 타이밍된 샘플링을 위해 아웃고잉 데이터 스트림을 프리-스큐 (pre-skew) 한다.

[0036] 도 3a 및 도 3b는 도 2a에 도시된 드리프트-보상 직병렬화기 중 어느 것을 구현하는데 사용될 수도 있는 드리프트-보상 직병렬화기 (186) 의 실시형태 및 타이밍 도를 예시한다. 그에 따라, 도 2a에서 참조된 8개 직병렬화기 중 소정의 하나에 대한 전용인 각 입력 신호 및 출력 신호가 도 3a, 도 3b에서 인덱스 "[i]" 로 표시되어, 분리된 예의 동일 신호들이 나머지 7개의 직병렬화기 (즉, $i=0,1,2,\dots,7$) 로의 입력 또는 이들로부터 출력된다는 것을 나타낸다. 따라서, 직병렬화기 (186) 는 데이터 링크 DQ [i]에 커플링되어 직렬 데이터 신호를 수신하고 8-비트 와이드 데이터 패킷 Rdata[i][7:0] 을 출력한다. 또한 직병렬화기는 6-비트 위상-조정 신호 PhAdj[i][5:0] 및 3-비트 비트-조정 신호 BitAdj[i][2:0]를 수신한다. 직병렬화기는 또한, 모든 다른 직병렬화기와 함께, 제어기 코어 클럭, PCK1, 및 멀티-위상 제어기 I/O 클럭, PCK8 을 수신한다. 도시된 예에서, 제어기 I/O 클럭은 3-스테이지 링 오실레이터 (three-stage ring oscillator) 에 의해 생성되고, 따라서 PCK8 사이클 시간 내 위상 분배된 3개 차동 클럭 신호들의 세트를 출력한다. 다른 말로, 도 3 A의 실시형태에서, 제어기 I/O 클럭은 0° , 120° 및 240° 의 클럭 위상 그리고 180° , 300° 및 60° 의 그들의 보수 (complement) 를 포함하며, 따라서 6개의 클럭 위상들의 세트를 제공하며 이로부터, PCK8 사이클 내 어느 위상 오프셋 (즉, 클럭 위상 또는 위상 각도) 를 갖는 위상-시프트된 수신 클럭, RCK8[i]가 합성될 수도 있다. 일 구현에서, 예를 들면, 위상 인터포레이터 (phase interpolator) (271) 는 위상-인접 클럭 위상들의 6개의 가능한 쌍들 (즉, $0^\circ / 60^\circ$, $60^\circ / 120^\circ$, $120^\circ / 180^\circ$, $180^\circ / 240^\circ$, $240^\circ / 300^\circ$ 또는 $300^\circ / 0^\circ$) 중 하나를 선택함으로써 그리고 위상 조정 값의 최하위 3개 비트에 대응하여 선택된 클럭-위상 쌍 사이에서 인터폴레이팅 (interpolating) (또는 믹싱 (mixing)) 함으로써 6-비트 위상 조정 값의 최상의 3개 비트 (MSB) 에 응답하며, 따라서 위상 조정 값의 각각의 증가 또는 감소로 $60^\circ / 8$ 또는 7.5° 위상 단계 (또는 레졸루션 (resolution)) 을 제공한다. 보다 많거나 또는 보다 적은 클럭 위상들이 (선택가능한 클럭-위상 쌍들의 수를 충족시키는데 필요한 위상 선택 비트의 수의 대응하는 변화를 갖는) 대안의 실시형태에서 제공될 수도 있고/거나 더 파인하거나 더 코스한 위상 인터폴레이션이 제공될 수도 있다. 또한, 위상 인터폴레이터 (271) 자체는 임의의 타입의 위상 시프트 회로에 의해 구현될 수도 있는데, 비한정적인 예로, MSB-선택 위상 벡터들을 수신하도록 각각 커플링된 입력들, 공통으로 타이 (tie) 된 출력들 및 위상 조정 값의 최하위 3개-비트들의 상보 예 (complementary instance) 에 의해 제어된 각각의 구동 세기 (drive strength) 를 갖는 증폭기를 포함한다. 보다 일반적으로, 제어기 I/O 클럭, PCK 8에 관하여 선택가능한 위상 오프셋을 제공할 수 있는 임의의 타입의 회로가 대안의 실시형태에서 사용될 수도 있다. 마지막으로, 인터폴레이터 회로 토폴로지 (interpolator circuit topology) 에 상관 없이, 도 1a의 토폴로지 내 포함된 인터폴레이터 (또는 위상-시프팅) 회로는, 소스 제어기 I/O 클럭, PCK8이 정지될 때, 인터폴레이팅된 클럭 RCK8[i] 가 글리치-프리 (glitch-free) (즉, 무 단축 (런트 (runt)) 펄스 또는 무효 논리 레벨이) 되는 것을 가능하게 한다. 예로서, 몇몇 실시형태들에서, 인터폴레이팅된 클럭의 글리치-프리 시작 및 정지는 인터폴레이터 회로로의 PCK8[0°] 및 PCK8[180°] 과형의

1-사이클-지연 카피의 여분의 쌍의 분배에 의해 가능해진다. 비슷한 배열이 아래에서 도 3f를 참조하여 논의된 제어기-측 송신 클럭 위상들의 글리치-프리 시작 및 정지를 보장하는데 사용될 수도 있다.

[0037] 아래에 논의된 바처럼, 수신 클럭 위상은 값들의 범위를 통해 (또는 바이너리 또는 다른 서치 패턴을 통해) 위상 조정 값을 스텝핑 (stepping) 하여 비트-에러를 낚는 것들로부터 에러-프리 데이터 수신을 낚는 결과적인 클럭 위상들을 (즉, 실패 클럭 위상 (failing clock phase) 으로부터 통과 클럭 위상 (passing clock phase) 들을) 구별함으로써 초기에 교정될 수도 있다. 일 실시형태에서, 예를 들면, 데이터 아이의 개방 및 폐쇄 측 상의 (또는 하나의 데이터 아이의 폐쇄 측 상의 그리고 다음 데이터 아이의 개방 측 상의) 통과/실패 바운더리 상에 놓이는 클럭 위상들 (즉, 에러-프리 수신 및 비트 에러를 각각 낚는 인접 클럭 위상들) 이 식별되고 그러한 바운더리들 사이에 센터링된 위상이 교정 수신 클럭, RCK8[i] 으로서 선택된다. 그 후에, 수신 클럭 위상은, 바운더리 위상들이 같은 통과 (또는 실패) 결과를 낚는지를 확인하기 위해 바운더리 위상들을 리-테스팅 하고, 통과/실패 바운더리에서의 변화에 의해 표시된 임의의 드리프트에 대응하기 위해 최종 수신 클럭 위상에 대한 위상-조정 값을 증가 또는 감소시킴으로써 메모리-측 (또는 시스템-와이드) 위상 드리프트를 처리하도록 주기적으로 (또는 때때로) 조정될 수도 있다.

[0038] 플롭 스테이지 (Flop stage) (또는 래치 (latch)) (283) 는 수신 클럭 신호, RCK8[i] 의 천이에 응답하여 직렬로 로딩된 8-비트 시프트 레지스터를 형성한다. 프레임링 클럭 신호, RCK1[i] 는 수신 클럭 신호의 8개 사이클 마다 1번씩 사이클을 이루고, 병렬-출력 레지스터 (285) 속으로 병렬로 시프트 레지스터의 콘텐츠를 전송하는데 사용되어 1:8 직렬-병렬 변환을 이룬다. 수신 클럭 (RCK8[i]) 의 네가티브-고잉 에지 (negative-going edge) 를 카운팅하기 위한 (3-비트-와이드 레지스터 (273) 및 증가 로직 (275) 에 의해 형성된) 모듈로-8 카운터 (modulo-8 counter) 및 3-비트 모듈로-8 카운터 출력으로 3-비트 비트-조정 값 (RxBitAdj[2:0]) 을 가산하는 가산기 회로 (277) 를 포함하는, 비트 정렬 회로는 수신 클럭 신호와 프레임링 클럭 신호 사이 정렬에 대한 선택가능한 제어를 제공한다. 더 상세하게는, 비트-조정 값이 0 (즉, RxBitAdj[i][2:0] = 000b, 'b' 는 바이너리를 표시함) 이면, 카운터 값이 3에서 4로 (011b 에서 100b로) 천이할 때마다, 가산기 출력 (278) 의 MSB 는 하이가 되고, 2개 수신-클럭 사이클 후 (플롭 스테이지 (279, 281) 때문에), 병렬-출력 레지스터의 콘텐츠를 로딩하기 위하여 프레임링 클럭 (RCK1 [i]) 신호의 대응하는 하이-고잉 에지 (high-going edge) 를 트리거링한다. 비트-조정 시그널의 각 증가는 가산기 MSB (그리고 따라서 RCK1 [i]) 가 1 비트-시간 먼저 하이가 되게 하며, 따라서 모든 8개 RCK8[i] 사이클 중 어느 하나의 하강 에지 (falling edge) 로 RCK1[i] (또는 그의 하이-고잉 천이) 의 정렬을 가능하게 하고 따라서 직렬-병렬 프레임링이 인커밍 직렬 비트 스트림 내 8개 가능한 패킷-프레이밍 바운더리 중 어느 것으로 시프트될 수 있게 한다. 도시된 실시형태에서, RCK1 [i] 의 각 상승 에지 (rising edge) 는 RCK8[i] 신호의 하강 에지로 정렬되어, 병렬 레지스터로의 전송이 시프트 레지스터가 새로운 8-비트 패킷으로 로딩된 후 하프-RCK8[i] 사이클 (그리고 다음 패킷의 첫번째 비트가 시프트 레지스터로 로딩되기 전 하프 RCK8[i] 사이클) 에서 일어난다.

[0039] 도 3b는 위에 설명된 타이밍 배열을 예시하며, 다중 위상 제어기 I/O 클럭, PCK8, (그 중 0° 클럭 위상만이 도시됨) 및 위상-시프트된 수신 클럭, RCK8[i] 의 예로 시작하여, PCK8[0°] 에 대한 임의의 위상 오프셋 (288) 및 예시적인 위상 오프셋 (291) 을 가져 라인 DQ[i] 상의 인커밍 데이터 파형으로 쿼드러처 (quadrature) (즉, 비트-시간-센터링된) 정렬을 이룬다. 모듈로-8 카운터의 최상위-비트 출력 (즉, RCK1a[i]) 은, 하강 수신-클럭 에지와 정렬된 수신 클럭 신호 및 천이의 8개 사이클 마다 한번의 사이클을 이룬다. 논의된 바처럼, 프레임링 클럭 RCK1[i] 는 (직렬로 커플링된 플롭 스테이지 (279,281) 에 기인하여) 카운터 출력 후 N+2 수신-클럭 사이클을 천이하고 여기서 N 은 비트 조정 값, RxBitAdj[i][2:0]에 따라, 0 내지 7 범위이다. 따라서, 비트 조정 값이 0 (000b) 이면, 프레임링 클럭 신호는, 로 카운터 출력 (raw counter output) 후 2사이클 천이하고, 도시된 도에서, (임의의 넘버링된) 데이터 비트 12 가 시프트 레지스터의 백 엔드 (back end) 로 로딩된 후 하프 (half) 사이클 천이한다. 그에 따라, RxBitAdj[i][2:0]=000b으로, 8 비트, 넘버링된 5-12가 시프트 레지스터 플롭 (283)으로부터 병렬-출력 레지스터 (285) 로 병렬로 전송되며, 그러한 비트들을 비트 4 와 5, 그리고 12 와 13 사이의 시작 및 종료 비트 바운더리에 대한 패킷로서 각각 프레임링한다. 계속 예를 들면, RxBitAdj = 1 (001b) 이면, 비트 6-13이 패킷으로 프레임링되고, RxBitAdj = 2 (010b) 이면, 비트 7-14 가 패킷으로 프레임링되고 BitAdj = 7 (111b) 까지 그렇게 진행되고, 이 경우에 비트 12-19이 패킷으로 프레임링된다.

[0040] 계속 도 3a 및 도 3b를 참조하면, 코어 클럭 및 프레임링 클럭은 수신 클럭 및 제어기 I/O 클럭 사이의 인트라-비트 위상 오프셋 및 전체 수신 클럭 사이클 중 일정 수 (0 내지 7) 을 베이스 프레임링 클럭 위상 (RCK1a[i]) 에 가산함으로써 이루어지는 비트-와이즈 오프셋 (bit-wise offset) 에 기인하여 서로에 대해 임의의 위상을 가

짐을 알 수 있다. 따라서, 드리프트-보상 직병렬화기로부터 제어기 코어로의 데이터 전송은 프레이밍 클럭 도메인으로부터 제어기 코어 클럭 도메인으로 교차하는 클럭 도메인을 수반한다. 이 전송은 8개 드리프트-보상 직병렬화기의 각각 내에 존재할 수도 있는 잠재적으로 상이한 프레이밍 클럭 도메인에 의해 더 복잡해진다. 게다가, 메모리 제어기 (또는 동일 클럭 생성 회로를 공유하는 다중 동일-다이 또는 분리-다이 메모리 제어기) 가 2개 이상의 메모리 디바이스와 통신하면, 데이터-타이밍 가변성 (data-timing variability) 은 단일 메모리 디바이스에 대한 워스트-케이스 보다 심지어 더 커질 수도 있다. 따라서, 인트라-비트 샘플링 위상 조정을 위한 위상-조정 회로 및 패킷-프레이밍 바운더리를 제어하는 비트-정렬 회로에 더하여, 패킷 정렬 회로가 제공되어 제어기 코어 도메인 속으로의 동시 전송을 위한 각각의 데이터 링크를 통해 수신된 패킷들의 집합적 세트를 정렬한다. 즉, 8 패킷들이 정렬 상태에서 메모리 코어로부터 메모리-측 I/O 회로로 전송 되더라도, 다양한 데이터 링크 사이의 위상 차이는 메모리 제어기에서의 패킷들의 시간-차등형 도착 (time-staggered arrival) 을 발생시킬 수도 있고, 따라서, 서로에 대한 (그리고 제어기 코어 클럭, MCK1) 에 대한 상이한 비트 오프셋에서의 패킷들의 프레이밍을 발생시킬 수도 있다. 결과적으로, 후-도착 (later-arriving) (더 레이턴트 (latent) 한) 패킷과의 정렬을 위해 선-도착 (sooner-arriving) 패킷의 전송을 지연시키기 위한 메카니즘의 부재시, (예를 들면, 본 예에서 8-바이트 값인) 메모리 코어로부터 검색된 원래 멀티-패킷 메모리 워드의 구성 패킷들은 제어기 코어로의 전송시 2이상의 메모리 워드 중에 일시적으로 분산될 수도 있는 (즉, 구성 패킷들 사이의 메모리-측 타이밍 관계가 손실될 수도 있는) 것을 의미하는 다른 것들보다 먼저 코어 클럭 (PCK1) 의 래칭 에지 (latching edge) 에 관하여 하나 이상의 원래-정렬된 패킷들 (originally-aligned packets) 이 이용가능할 수도 있다. 따라서, 일 실시형태에서, 제어기 I/O 회로로부터 제어기 코어로의 패킷 전송에서 메모리-코어 패킷 정렬을 보장하기 위한 회로가 유지 (또는 복원) 된다. 도 3a의 실시형태에서, 예를 들면, 그러한 패킷 정렬 회로는 패킷-와이드 FIFO (packet-wide first-in-first-out) 버퍼 (287) 에 의해 구현되는 데, 이는 프레이밍 클럭 (또는 FIFO 클럭, FCK1 [i]) 으로 지칭되는 그의 1-비트-시간-어드밴싱된 버전 (one-bit-time-advanced version)) 에 의해 로딩되고, 제어기 코어 클럭, PCK1에 의해 언로딩되며, 워스트-케이스 타이밍 조건하에서 최대-레이턴트 및 최소-레이턴트 패킷-프레이밍 시간들 사이의 인터벌에 의해 스패닝 (spanning) 된 코어 클럭 사이클의 정수와 동일한 패킷의 수를 유지할 만큼 깊다.

[0041] 도 3c-도 3d 는 도 3a의 패킷-정렬 회로 (287) 를 구현하는데 사용될 수도 있는 FIFO-기반 패킷-정렬 회로 (290) 의 구현 형태 및 대응하는 타이밍 도를 예시한다. 패킷-정렬 회로 (290) 은 4개-패킷-깊이 버퍼 (299), 로드 회로 (291) 및 언로드 회로 (301) 을 포함한다. 로드 회로 (291) 는 2-비트 로드 카운트를 출력하는 모듈로-4 로드-카운터 (292) (즉, 카운트 시퀀스=0,1,2,3,0,1,..., 증가 로직 (293) 및 2-비트 레지스터 (294) 에 의해 구현됨), 패킷 조정 값 RxPktAdj[i][1:0] 를 로드 카운트에 가산하여, 로드 카운트가 0-3 프레이밍 클럭 사이클에 의해 어드밴싱될 수 있게 하는 (즉, 로드 카운트가 사실상 4개 가능한 초기 카운트 값들 중 어느 것으로 조정될 수 있게 하는) 2-비트 가산기 (295), 및 상승 FCK1 에지에 응답하여 인커밍 패킷, P[i][7:0]으로 로딩될 4-깊이 버퍼 (299) 내 4개 패킷 레지스터 중 하나를 선택하기 위해 가산기-조정 로드 카운트를 디코딩하는 2:4 디코더 (297) 를 포함한다. 사실상, 로드 회로 (291) 은 4-깊이 버퍼로의 "로드 포인터" (load pointer) 를 회전시키는 것을 구현하며, (3 (11b) 로부터 0(00b) 까지)에 대해 가산기-조정 카운트 (adder-adjusted count) 가 롤링 (rolling) 함에 따라 마지막 패킷 레지스터로부터 첫번째로 래핑 (wrapping) 하는 시퀀스에서 하나의 패킷 레지스터를 다른 하나 다음 선택하고, 가산기 (295) 는 포인터가 패킷-조정 값, RxPktAdj[i][1:0] 에 따라 임의의 시작 패킷 레지스터 포지션으로 어드밴싱될 수 있게 한다.

[0042] 도 3c를 계속 참조하면, 언로드 회로 (301) 는 코어 클럭 신호 (PCK1) 의 상승 에지에 응답하여 2-비트 카운트 시퀀스 또는 "언로드 카운트"를 생성하기 위한 (증가 로직 (303) 및 2-비트 레지스터 (304) 에 의해 형성된) 모듈로-4 언로드 카운터 (302), 및 언로드 카운트에 응답하여 4-깊이 버퍼 (SEL0-SEL3)의 패킷-레지스터 출력을, 차례로, 선택하기 위한 4:1 멀티플렉서 (305) 를 포함한다. 따라서, 로드 회로 (291) 는 FCK1 에 응답하여 (버퍼의 4개 패킷 레지스터 (299) 를 통해 순차적으로 회전하는) 라운드-로빈 방식 (round-robin fashion) 으로 패킷 레지스터를 로딩하고, 언로드 회로 (301) 는 로드 회로의 회전을 따르고, PCK1에 응답하여 라운드-로빈 방식으로 패킷 레지스터를 언로딩한다. 인커밍 패킷 조정 값은 로드 회로에 의해 구현된 회전 포인터가 원하는 수의 PCK1 클럭 사이클에 의해 언로드 회로에 의해 구현된 회전 포인터를 리딩 (leading) 하는 것을 가능하게 한다. 아래에 논의된 바처럼, 교정 동작이 수행되어 각 링크를 위한 FIFO 로딩 및 언로딩 사이의 최소 레이턴시를 결정하고, 그 다음 워스트-케이스 최소를 매칭시키기 위해 각 링크를 위한 로드-언로드 레이턴시를 설정함으로써 모든 링크들을 정렬시키게 수행될 수도 있다.

[0043] 도 3d는 링크 DQ[0] 및 DQ[7] 상의 예시적인 타이밍 데이터 타이밍 패턴을 위한 패킷-조정 값을 조정하는 효과를 예시한다. 보다 상세하게는, 제어기 코어 클럭 (PCK1) 을 레퍼런스로 사용하여, 링크 DQ[0] 을 위한

FIFO-로드 클럭이 PCK1 사이클의 부분 (fraction) 에 의해 PCK1을 래깅 (lagging) 하고, DQ[7] 을 위한 FIFO-로드 클럭은 대략 동일한 부분에 의해 PCK1을 리딩 (leading) 하는 것으로 가정된다. 또한, 설명을 위하여, 패킷 조정 값 00, 01, 10 및 11이 패킷 레지스터 출력 SEL0, SEL1, SEL2, 및 SEL3의 초기 선택을 각각 발생시키는 것으로 가정된다. 실제 동작에서, 로드 카운터 (292) 를 예정된 상태로 초기화하기 위한 회로의 부재 시, 패킷 조정 값들은 4개 가능한 초기 로드 카운터 상태 (00, 01, 10, 11) 중 어느 것에 의해 오프셋되는 초기 패킷 레지스터 출력 선택을 낳을 수도 있다.

[0044] 데이터 읽기 동작 (또는 교정 데이터 송신) 이 각 데이터 링크 상 패킷 (i) ("Pkt i") 를 포함하는 인커밍 패킷 시퀀스를 낳는 것을 가정하면, 그 다음 FCK1[0] 의 래깅 위상은, (예를 들면, 제어기 코어가 패킷 "i"의 리턴을 낳은 리퀘스트 또는 다른 송신을 이슈 (issue) 하기 때문에, N번째 PCK1 사이클의 시작을 마킹 (marking) 하는) PCK1의 상승 에지 N 직후 수신되고 패킷 조정 값, RxPktAdj[0][1:0]에 따라 4개 패킷 레지스터 (플롭0, 플롭1, 플롭2 또는 플롭3) 중 하나로 로딩되는 서브젝트 패킷 (subject packet) 을 발생시키게 된다. 즉, 패킷 조정 값이 00이면, 패킷 "i"는 (출력 SEL0 을 갖는) 플롭0 으로 로딩되고 거기에 4 FCK1 사이클 동안 남아 있다. 마찬가지로, 패킷 조정 값이 01, 10 또는 11이면, 패킷 "i"는 도시된 바처럼 플롭1 (SEL1), 플롭2 (SEL2) 또는 플롭3 (SEL3) 으로 로딩된다.

[0045] 설명을 위해, 언로딩된 포인터가 PCK1의 샘플링 (상승) 에지에서, 플롭0로 (즉, 패킷 레지스터 출력 SEL0이 멀티플렉서 (305) 에 의해 선택됨), (그리고 그 다음, PCK1 에지 N+1, N+2, N+3 에서 각각 플롭1, 플롭2, 플롭3로) 포인터 된다고 가정하고, 또한, 패킷 "i" 가 플롭0으로 로딩된다고 가정하면, 패킷이 PCK1 샘플링-에지 N 바로 다음 (그리고 플롭0이 코어 도메인에 언로딩된 바로 다음) 에 로딩되기 때문에, 거의 4개 전체 PCK1 사이클은 FCK1[0]의 상승 에지 0에서 플롭0으로 로딩 패킷 "i" 과 PCK1의 상승 에지 N+4에서의 플롭0으로부터의 언로딩 패킷 "i" (언로딩은 샘플링 표시자 (312) 로 도시) 사이에서 일어나야 한다. 코어 로직의 관점에서부터, 리퀘스트/커맨드 출력으로부터 (코어 도메인으로부터) 데이터 리턴으로 (다시 코어 도메인으로) 의 라운드-트립 레이턴시 (round-trip latency) 는 패킷 조정 값이 '00' 으로 설정(즉, (N+4)-(N+1)=3) 될 때 보다 '01' 로 설정될 때 3개 더 적은 코어 클럭 사이클을 필요로 했다. 사실, 본원에서 최소 링크 레이턴시로도 지칭되는, link[0] 을 위한 최소 라운드 트립 레이턴시는 패킷-조정=01에 대해 N+1 클럭 사이클이고, 패킷-조정 값이 증가됨에 따라 계속해서 더 큰 --N+2, N+3, N+4-- 상태가 되고 또한 패킷 레지스터 플롭2, 플롭3, 플롭0 각각에 대한 언로드 포인터 보다 앞서게 로드 포인터를 어드밴싱 (advancing) 한다.

[0046] 계속 해서 도 3d를 참조하면, FCK1[7] 의 로딩 에지가 PCK1의 플롭0 샘플링 에지 바로 전에 일어나기 때문에, 링크 DQ[7]에 대한 최소 링크 레이턴시는 'N' PCK1 사이클이고 링크 패킷-조정 값 (RxPktAdj[7][1:0]) 이 '00' 일 때 일어난다. 패킷 조정 값이 01, 10, 11으로 증가될 때, 링크 레이턴시는 N+1, N+2, N+3 에 대응하는 수의 PCK1 사이클 만큼 증가한다.

[0047] 도 3d의 예시적인 타이밍 도가 설명하는 바처럼, 상이한 링크들은 상이한 최소 링크 레이턴시를 나타낼 수도 있다. 그리고 또한, 각각의 데이터 링크 상의 i번째 패킷은 메모리 디바이스 코어로부터 검색된 (또는 교정 동작에서의 제어기 코어로부터 이슈된) 동일한 멀티-패킷 워드의 구성요소들이기 때문에, 코어 클럭 신호의 동일한 샘플링 에지에 대응하여 제어기 코어 도메인으로 i번째 패킷들을 모두 전송함으로써 i번째 패킷들 사이의 시간적 (temporal) 관계를 유지하는 것이 중요하다. 도 3d로부터 인식될 수 있는 바처럼, 이 "패킷-정렬" 동작은 사실상 모든 시그널링 링크에 대한 링크 레이턴시를, 그들 개별적인 최소 레이턴시가 무엇이든지 간에, 이퀄라이징 (equalizing) 하는 것이다.

[0048] 도 3e는 모든 데이터 링크에 걸쳐, 본원에서 최소 시스템 레이턴시로 지칭되는, 균일한 링크 레이턴시를 확립하는 예를 제공한다. 특히, 정보 (예를 들면, 커맨드, 데이터 마스크) 를 단일 방향으로 전달하는데 주로 사용되는 일부 시그널링 링크가 때때로 정보를 메모리 제어기로 리턴하는데 사용되는 경우, 이 동작은 일반적으로 모든 시그널링 링크로 확장될 수도 있다.

[0049] 초기에, 각 데이터 링크에 대한 링크 레이턴시 (본 예에서 읽기 데이터 레이턴시) 가 패킷-조정 값의 각 설정에 대해 결정된다. 이것은, 예를 들면, 각 링크 상에서, (상이하게-패턴화된 패킷들에 의해 선행 및 후행되는) 예정된 비트 패턴을 갖는 패킷을 수신하도록 배열하고, 그 다음 패킷이 수신되기 전에 일어나는 PCK1 사이클들의 수를 카운팅함으로써 이루어질 수도 있다. 예로서, (아래에서 더 자세히 설명되는) 일 실시형태에서 메모리 디바이스가 데이터-루프-백 모드 (data-loop-back mode) 에서 배치되며, 메모리-측 코어 인터페이스에서의 데이터를 루프백 (loop back) 하여 하나의 링크 (홀수 링크) 에 의해 송신된 데이터 패킷은 다른 하나 (예를 들면, 카운터파트 짝수 링크) 상에서 수신되고 따라서 각 상이한 패킷 조정 값에 대한 라운드-트립 레이턴시 결정

을 가능하게 한다. 다른 실시형태에서, 결정적 (예를 들면, 이전에 쓰여지거나 그렇지 않으면 예측가능한) 읽기 데이터 패턴의 읽기 커맨드 리퀘스트 리턴이 메모리 디바이스에 이슈되어, 각 링크 및 각 패킷 조정 값에 대한 (제어기 코어로부터의 읽기 커맨드의 출력으로부터 제어기 코어 내 예측 데이터의 획득으로) 라운드 트립 레이턴시 결정을 가능하게 한다. 어떻게 이루어지더라도, 링크-레이턴시 데이터의 세트가 획득되며, 각 링크에 대한 각 패킷 조정 값에 대한 상대 링크 레이턴시 (본 예에서 읽기 데이터 레이턴시) 값 (예를 들면, 코어 클럭 사이클의 수) 을 포함한다. 도 3e의 323에 도시된 예에서, 링크 레이턴시 데이터는 링크 DQ[1]에 대한 유사 데이터와 함께, 링크 DQ[0] 및 링크 DQ[7]에 대해 도 3d에 도시된 예시적인 링크 레이턴시를 반영한다.

도시된 바처럼, DQ[1]에 대한 링크 레이턴시는 링크 DQ[0]에 대한 것들과 매칭하지만, (2개 PCK1 사이클에 의해 회전된) 상이한 패킷 조정 값에서 일어나며, 적어도 하나의 실시형태에서, 로드 카운터와 언로드 카운터의 초기 상태는 완전히 임의적이라는 것을 설명한다.

[0050] 도 3e로 계속하면, 제어기 코어 내 프로세서 (또는 다르게는, 호스트 프로세서 또는 다른 업스트림 제어기) 는 325에서 각 링크에 대한 최소 링크 레이턴시 (본 예에서, DQ[0], DQ[1] 링크에 대해서는 N+1 PCK1 사이클 그리고 링크[7] 에 대해서는 N PCK1 사이클) 를 결정할 수도 있고 그 다음 327에서의 위스크-케이스 (즉, 최대) 링크 레이턴시에 기초하여 최소 시스템 레이턴시를 결정할 수도 있다. 도시된 실시형태에서, 예를 들면, 최소 시스템 레이턴시는, 이 경우에 N+1 PCK1 사이클인 개별 링크 레이턴시의 최대값이 되도록 결정된다. 그 후, 329에서, 각 링크 (RxPktAdj[i][1:0]) 에 대한 패킷 조정 값이 최소 시스템 레이턴시에 대응하는 값으로 (예를 들면 아래에 설명된 바와 같이 패킷 정렬 카운터 내에서) 프로그래밍된다. 따라서, 도시된 특정 예에서, 링크 DQ[0], DQ[1] 및 DQ[7] 에 대한 패킷 조정 값들이 '01', '11 ' 및 '01'으로 각각 프로그래밍되어 그러한 패킷-코어 전송을 최소 시스템 레이턴시로 정렬한다. 특히, DQ[7] (RxPktAdj[7] = '00') 에 대한 심지어 더욱 낮은 레이턴시 설정을 위한 기회에도 불구하고, 그 링크의 동작은, 사실상 더 느린 (더 레이턴트한) 링크로의 정렬을 이루기 위한 PCK1 사이클에 의해 지연됨에 유의한다.

[0051] 드리프트-보상 직렬화기 및 직병렬화기 회로 내에서 사용될 수도 있는 예시적인 위상-정렬, 비트-정렬 및 패킷-정렬을 설명했지만, 본원에 제시된 원리들로부터 이탈함이 없이 설명된 결과들을 이루기 위하여 수많은 다른 회로 구현들이 사용될 수도 있음을 유념해야 한다. 예를 들면, 다양한 타입들의 지연 회로 및 다른 타입들의 위상 시프팅 회로들이 원하는 수신 및 송신 클럭 위상들을 발생시키기 위해 사용될 수도 있다. 또한, 비트 정렬에 관하여, 도 3a 및 도 3f에 도시된 가산기 회로 (277, 345) 대신, 추가의 시프트 레지스터 스테이지들이 시프트 파이프라인 내 상이한 포인트들에서의 출력의 멀티플렉서 선택으로 제공될 수도 있다 (따라서 선택가능한 $n \cdot t_{bit}$ 지연을 이룬다, 여기서 'n'은 트레이싱 (traversing) 된 추가의 시프트 레지스터 스테이지들의 선택가능한 수이고, t_{bit} 는 비트-시간 인터벌이다). 마찬가지로, 패킷 정렬에 관하여, 추가의 병렬 레지스터가 상이한 워드 정렬의 선택을 가능하게 하기 위해 멀티플렉서와 함께 제공될 수도 있다. 보다 일반적으로는, FIFO 버퍼 배열 대신에, 사이클-스킵 (cycle-skip) 회로는 다수의 PCK1 에지들 (예를 들면, 도 3d에 도시된 바와 같은 N, N+1, N+2, N+3, N+4) 중 하나를 선택하여 단일 패킷 레지스터로부터 코어 도메인으로 데이터를 전송한다.

[0052] 도 3f 및 도 3g는 도 2a에 도시된 드리프트-보상 직렬화기 중 어느 것을 구현하는데 사용할 수도 있는 드리프트-보상 직렬화기 (185) 의 실시형태 및 타이밍 도를 예시한다. 도 3a의 드리프트-보상 직병렬화기와 같이, 드리프트-보상 직렬화기는 패킷 정렬, 비트 정렬 및 인트라-비트 타이밍 위상 조정 모두를 직병렬화기에 대한 역순으로 수행하기 위한 회로를 포함한다. 사실상, 드리프트-보상 직렬화기는 각 시그널링 링크의 패킷 (패킷-정렬) 을 서로에 대해, 각 패킷의 비트 (비트-정렬) 및 데이터-레이트 송신 클럭 신호의 인트라-비트 위상을 프리스큐하여 각 링크에 대한 데이터 송신을 정렬함으로써, 카운터파트 메모리-측 수신 회로가 원하는 인트라-비트 인스턴트 (intra-bit instant)에서 각 비트를 샘플링하고, 메모리 제어기에 의해 의도된 패킷-프레이밍에 따라 비트의 각 그룹을 패킷으로 프레이밍하고, 동일한 멀티-패킷 데이터 워드의 부분을 형성하는 모든 패킷들을 메모리 코어 도메인 속으로 싱크로니 (synchrony) 에서, 모두 위상 메모리-측 타이밍 보상 회로를 요함이 없이, 전송한다. 그에 따라, 패킷-정렬 FIFO 371은 제어기 코어 클럭 (PCK1) 에 대응하여 송신 데이터 패킷 (Tdata[i][7:0] 및 따라서 본 예에서 각기 8-비트 패킷) 의 시퀀스로 로딩되고, 디-프레이밍 (de-framing) 클럭 신호 (TCK1[i])의 버퍼 지연 예 (FCK1[i]) 에 응답하여 병렬 레지스터 (367) 로 언로딩 (즉, 패킷이 FIFO 또는 큐의 헤드로부터 팝핑 (popping) 됨) 되어, 상이한 링크들에 대해 제어기-코어-메모리-코어 전파 시간 차이를 보상하기 위해 필요할 때 제어기-코어로부터 동일한 멀티-패킷 워드로부터의 패킷들이 상이한 시간에서의 제어기 I/O 도메인으로 로딩될 수 있게 한다. 도 3a의 직병렬화기 내 프레이밍 클럭 신호 RCK1[i] 와 같은 방식으로 생성된 디-프레이밍 클럭 신호 TCK1[i] 에 응답하여 직렬-출력 시프트 레지스터 (365) 로 병렬 레지스터

(367)의 콘텐츠가 로딩된다. 즉, 디-프레이밍 클럭 신호는 비트-레이트 송신 클럭 신호 TCK8[i]를 (레지스터 (341) 및 증가 로직 (343)에 의해 형성된) 모듈로-8 카운터에서 8에 의해 분할하고, 가산기 (345)에서의 카운터 출력으로 3-비트 비트 조정 값을 가산함으로써 생성되어, 모듈로-8-카운터의 출력이 0 내지 7 범위의 값에 의해 오프셋될 수 있게 하고 따라서, 디-프레이밍이 8개 가능한 비트 바운더리 중 어느 것 상에서 일어날 수 있게 한다. TCK8[i]의 8개 사이클마다 한번 사이클을 이루는, 가산기 출력의 MSB는, 플롭 스테이지 (351)에서의 송신 클럭, TCK8[i]의 네가티브 고잉 에지 (negative going edge)와 동기화후, 디-프레이밍 클럭, TCK1[i]를 형성한다. 디-프레이밍 클럭은 네가티브-TCK8[i]-에지-트리거드 플롭-플롭 (353, 355, 357) 중 3개의 시퀀스를 통해 시프트되며, 최종 2개 플롭 스테이지 (357, 355)의 출력은 AND 게이트 (359)의 인버팅 및 논-인버팅 입력에 공급되어 단일-TCK8[i]-사이클 로드 펄스, LD[i]를 디-프레이밍 클럭 사이클마다 한번씩 생성한다. 로드 펄스는 직렬-아웃 시프트 레지스터 (365)내 플롭 스테이지의 로드-인에이블 입력에 공급되어, 로드 펄스가 하이 가 될 때, 병렬 레지스터 (367)의 콘텐츠는 직렬-아웃 시프트 레지스터 (365)로 로딩되고, 하프 TCK8[i] 사이클 후, (네가티브-에지-트리거드 플롭 스테이지 (361)에 기인하여), 출력 플롭 (363)으로 비트-바이-비트 시프트되고 DQ[i] 랭크 상으로 구동된다. 도 3a의 직병렬화기에서 처럼, 인터플레이터 (364) (또는 다른 클럭-위상 시프터)가 제공되어 송신 클럭 신호 TCK8[i]와 제어기 I/O 클럭, PCK8 사이의 교정된 인트라-비트 (또는 인트라-사이클) 타이밍 오프셋을 가능하게 한다. 드리프트-트래킹 위상 오프셋을 확립하고 조정하기 위해 적용된 교정 동작이 아래에서 설명된다. 도 3a의 드리프트-보상 직병렬화기를 참조하여 논의된 바처럼, 일부의 실시형태에서, 인터플레이팅된 클럭, TCK8[i]의 글리치-프리 시작 및 정지가 인터플레이터 회로 (364)로의 PCK8[0°] 및 PCK8[180°] 파형들의 1-사이클-자연-카피의 여분의 쌍의 분배에 의해 가능해지지만, 대안의 기법들이 글리치-프리 동작을 보장하기 위해 사용될 수도 있다.

[0053] 도 3g는 위에서 설명된 다양한 클럭, 제어 및 데이터 신호들 사이의 타이밍 관계를 예시한다. 보다 상세하게는, PCK8과 TCK8[i]도메인 사이의 임의의 위상 관계가 334에서 (멀티-위상 PCK8 클럭 신호의 오직 0° 클럭 위상만이 도시됨에 유의), 로드 펄스, LD[i]의 타이밍과 그의 비트 조정 신호, TxBitAdj[i][2:0]에 대한 의존성과 함께 도시되어, 직렬 출력 스트림 내 증가되게 비트-시프트된 포지션들에서의 송신을 위해 데이터의 소정 패킷을 디-프레이밍한다. 보다 상세하게는, 병렬 레지스터 내 데이터의 패킷은 비트 조정 값 TxBitAdj[i][2:0]에 따라 상이한 디-프레이밍 인터벌에서의 직렬-아웃 레지스터로 전송되어, 패킷 바운더리가 아웃고잉 직렬 비트스트림 내 비트-와이즈 시프트 (bit-wise shift)될 수 있게 한다. 즉, 비트 조정 값이 0 (TxBitAdj[i]=0, 또는 000b)이면, 병렬 레지스터 (367)내 데이터의 패킷이 비트 19 (임의로 할당된 숫자)의 송신의 종료시 직렬-아웃 시프트 레지스터 (365)로 로딩되고, 그 다음 비트 21-28로서 송신된다. TxBitAdj[i]=1인 경우, 패킷은 비트 20의 송신 종료시, 1비트 시간 후 직렬-아웃-시프트 레지스터로 로딩되고, 그 다음 비트 22-29로서 송신된다. 계속하면, TxBitAdj[i]=2,3,4,...,7인 경우, 병렬 레지스터로부터의 패킷이 TxBitAdj[i]=0인 경우보다 나중에 대응하는 수의 비트 시간 (즉, 2,3,4,..., 또는 7비트 시간 후)에 직렬-아웃 시프트 레지스터로 로딩되고, 그 다음 직렬 비트스트림 내 비트 23-30, 24-31, 25-32,..., 또는 28-35로서 대응하는 수의 비트 시간 후에 송신된다.

[0054] 도 3h는 도 3f의 패킷-정렬 회로 (371)을 구현하는데 사용될 수도 있는 FIFO-기반 패킷-정렬 회로 (380)의 구현형태를 예시한다. 패킷 정렬 회로는 일반적으로 도 3c-도 3e를 참조하여 설명된 것처럼 동작하지만, 역방향에서, 사실상, 메모리-측 코어로의 정렬된 전송을 보장하기 위해 필요할 때 컴패니언 패킷 (companion packet) (즉, 동일한 아웃고잉 데이터 워드 또는 커맨드 워드에 속하는 것들) 사이의 오-정렬 (mis-alignment)를 확립한다. 그에 따라, 패킷 정렬 회로 (380)는 (도 3h에서 각각의 출력 SEL0-SEL3로 나타낸) 패킷 레지스터 플롭0-플롭3을 갖는 4-깊이 FIFO 버퍼 (351)와 FIFO 버퍼를 로딩 및 언로딩하기 위한 로드 회로 (381) (또는 로드 포인터) 및 언로드 회로 (383) (또는 언로드 포인터)를 포함한다. 도시된 실시형태에서, 로드 회로 (381)은 (증가 로직 (385) 및 레지스터 (386)에 의해 형성된) 모듈로-4 카운터 (384), 및 일반적으로 도 3c의 로드 포인터 (291)의 대응하는 엘리먼트들과 동일하게 기능하지만, FCK1 대신에 PCK1[i]에 의해 클럭킹되는 2:4 디코더 (387)을 포함한다. 언로드 회로 (383)는 (증가 로직 (391) 및 레지스터 (392)에 의해 형성된) 모듈로-4 카운터 (390) 및, 일반적으로 도 3c의 언로드 포인터 (301)의 대응하는 컴포넌트를 참조하여 설명된 바와 같이 기능하지만, PCK1 대신에 FCK1[i]에 의해 클럭킹되는 4:1 멀티플렉서 (395)를 포함하고 2-비트 가산기 (393)를 포함하여 로드 시퀀스가 0, 1, 2, 또는 3 (0 내지 3) FCK1 샘플링 에지에 의해 어드밴싱되는 것을 가능하게 한다. 이러한 배열에 의해, FIFO 버퍼 (382)의 패킷 레지스터가 PCK1의 연속하는 에지들에 응답하여 회전 시퀀스에서 로딩되고 FCK1[i]의 연속하는 에지들에 응답하여 회전 시퀀스에서 언로딩되며, 로드-언로드 레이턴시는 모듈로-4 언로드 카운터 (390)의 출력에 가산되는 TxPktAdj[i][1:0] 값을 통해 조정가능하다. 그에 따라, 이전에 교정된 드리프트-보상 직병렬화기를 통해 (예를 들면, 루프백 또는 쓰기 및 읽

기 백을 통해) 송신된 데이터를 검색함으로써, 송신 패킷 조정 값의 각 설정에 대응하는 레이턴시 값들이 각 시그널링 링크에 대해 결정될 수도 있다; 최소 링크 레이턴시가 제어기-메모리 시그널링을 위한 시스템 링크 레이턴시를 확립하기 위해 확인 및 사용될 수도 있다. 그 후에, 시스템 링크 레이턴시 값은 각 시그널링 링크를 위한 송신 패킷 조정 값들을 프로그래밍 또는 그렇지 않으면 확립하는데 사용되어 직렬화시 균일한 정렬 및 메모리-측 코어 클럭 도메인으로의 전송을 보장할 수도 있다.

[0055] 도 4a 및 도 4b는 도 2a의 메모리 디바이스 내 직병렬화기 및 직렬화기 중 어느 것을 구현하는데 사용될 수도 있는, 직병렬화기 및 직렬화기 회로 (400, 415) 의 실시형태들을 각각 예시한다. 도시된 바처럼, 코어 메모리 클럭, MCK1은 패킷-프레이밍 및 디-프레이밍 클럭으로서 조정 없이 사용될 수도 있고, 다른 위상-조정 또는 비트-조정 회로는 제공될 필요가 없다. 또한, MCK4 신호는 하프 데이터 레이트에서 진동하므로, MCK4의 상승 및 하강 에지 양쪽 모두 (또는 MCK4의 상승 에지 및 상보 클럭, /MCK4 의 하강 에지 (또는 그 역도 같다) 가 메모리-측 직렬화기 및 직병렬화기 회로 내에서 시간 데이터 송신 및 수신에 사용되어, 데이터-레이트 타이밍을 이룰 수도 있다.

[0056] 도 4a의 예시적인 직병렬화기 (400) 에서, (쓰기 데이터, 커맨드/어드레스 정보, 교정 정보 등을 지닐 수도 있는) 인커밍 데이터 신호는 메모리-측 I/O 클럭, MCK4 의 상승 및 하강 에지에 대응하여 짝-데이터 플롭 (401) 및 홀-데이터 플롭 (403) 으로 각각 교번하여 클럭킹된다. 그 후, 짝-데이터 및 홀-데이터 플롭 내 캡처링된 데이터는 함께 짝-데이터 시프트 레지스터 (402) 및 홀-데이터 시프트 레지스터 (404) 로 시프트되며, 각 시프트 레지스터는, 이 8-비트 패킷 예에서, 4 플롭 스테이지를 갖는다. MCK4 신호의 4 사이클 마다 한번씩, 짝 및 홀 시프트 레지스터가 데이터의 전체 패킷으로 로딩된 후, MCK1의 상승 에지가 병렬-아웃 패킷 레지스터 (405) 내 (시프트 레지스터 (402, 404) 의 출력에서 병렬로 이용가능한) 데이터의 패킷을 래칭 (latching) 하는데 사용되어, 수신 데이터 Rdata[i][7:0] (예를 들면, 쓰기 데이터, 교정 데이터, 구성 데이터, 커맨드/어드레스 정보, 데이터-마스크 정보 등) 으로서 메모리 코어 도메인 인터페이스로의 패킷의 전송을 이룬다.

[0057] 도 4b의 예시적인 직렬화기 (415) 에서, 8-비트 송신 데이터 패킷, Tdata[i][7:0] 은 MCK1 사이클 마다 한번씩 생성된 로드 펄스 (430) 에 응답하여 4-스테이지, 2-비트-와이드 시프트 레지스터 (416) (이는 각각 패킷의 짝수 및 홀수 비트를 위한 단일-비트 시프트 레지스터의 쌍으로 볼 수도 있음) 으로 병렬-로딩된다. 그 후에, 시프트 레지스터의 헤드에서 (즉, 플롭 스테이지 R01에서) 2개 비트가, 비트의 다음 쌍이 다음 MCK4 사이클에서의 송신을 위해 포워드 시프트되기 전에, 소정 MCK4 사이클의 각각의 로우 및 하이 위상들에서 출력 드라이버로 적용 (그리고 따라서 링크 DQ[i] 상으로 구동) 된다. 도시된 바처럼, 플롭-플롭 (421) 이 MCK4 사이클의 하이 위상 동안 출력을 위해 제공되는 비트의 유지-시간 (hold-time) 을 보장하기 위해 제공되고, 충분한 유지 시간이 그와 달리 이용가능한 경우 생략될 수도 있다.

[0058] 로드 펄스 (430) 은 임의의 수의 방법으로 생성될 수도 있지만, 도시된 예에서는 플롭 (423) 및 AND 게이트 (425) 에 의해 생성된다. 보다 상세하게는, AND 게이트 (425) 는 인버팅 입력에서 플롭 (423) 의 출력과 논-인버팅 입력에서 MCK1 을 수신하고 따라서 코어 클럭, MCK1 의 각 상승 에지 다음 MCK4의 첫번째 사이클에 대해 확장되는 펄스를 생성한다. AND 게이트 (425) 로부터의 펄스는 로드 펄스 (430) 를 날기 위해 네가티브-MCK4-에지-트리거드 플롭 (429) 에서 하프 MCK4 사이클 후에 리-타이밍 (re-timing) 되기 전에 충분한 유지 시간을 보장하도록 플롭 (427) 에서 버퍼링된다. 로드 펄스 자체가 멀티플렉서 스테이지 (M01, M23 및 M45)에 공급되어, 시프트 레지스터의 컴포넌트 레지스터 (R01, R23 및 R45) 가 직렬화될 패킷의 구성 비트로 병렬-로딩 (즉, 비트 0, 1을 R01로; 비트 2, 3을 R23으로; 그리고 비트 4, 5를 R45로) 되는 한편, 패킷의 비트 6 및 7은 입력-스테이지 레지스터 R67 로 동시에 로딩되는 것을 가능하게 한다.

[0059] 대안의 실시형태에서, 다양한 구현 상세들이 도 4a 및 도 4b의 직렬화기 및 직병렬화기 회로 내에서 변화될 수도 있다. 예를 들면, 8:1 직렬화를 이루기 위해 (도시된 예에서 하프-비트-레이트 클럭) 데이터-레이트 클럭으로 시프트 레지스터를 클럭킹하는 대신에, 2:1 멀티플렉서 및 플롭-플롭 스테이지의 시퀀스가, 연속하는 스테이지에서, 선택 제공되어, MCK1 (스테이지 1) 의 하이 및 로우 위상에 응답하여 8-비트 패킷의 4-비트 부분을 교번하고, MCK4 (스테이지 2) 의 2-분할-예의 하이 및 로우 위상에 응답하여 각 4-비트 부분의 2-비트 부분을 교번하고, 출력 스테이지에서 MCK4의 하이 및 로우 위상들에 응답하여 각 2-비트 부위의 단일 비트들을 교번할 수도 있다. 유사한 배열이 1:8 직병렬화를 수행하는데 이용되며, 입력 스테이지에서 패킷의 2-비트 부분으로 개개의 비트를 결합하고, 제 2 스테이지에서 패킷의 4-비트 부위로 비트 쌍을 결합하고, 그 다음 제 3 스테이지에서 패킷의 4-비트 부위를 결합할 수도 있다. 메모리 제어기 내 드리프트-보상 직렬화기 및 직병렬화기는 마찬가지로 데이터-레이트 클럭에 의해 클럭킹된 시프트 레지스터 보다는 오히려 연속하는 2:1 멀티플렉싱 (또는 디멀티플렉싱) 스테이지로 구현될 수도 있다. 그 경우에, 비트 조정은 오프셋 값을 주파수-분할

된 로컬 클럭에 가산함으로써 이루어질 수도 있다.

[0060] 교정

[0061] 도 2a의 메모리 시스템에서, 각 드리프트-보상 직병렬화기 및 직렬화기 내 클럭 위상, 비트 정렬 및 패킷 정렬 회로의 교정은 교정되는 시그널링 링크를 통해 송신된 데이터에 기초하여 수행된다. 일 실시형태에서, 초기 교정 작업들이 수행되어 드리프트-보상 직병렬화기 내에서 그리고 그 다음 드리프트-보상 직렬화기내에서 신뢰성있는 동작을 확립하고, 그 후에 주기적 (또는 때때로의) 타이밍 교정 동작이 수행되어 메모리-측 타이밍 드리프트 (및 보다 일반적으로는, 시스템-와이드 타이밍 드리프트) 를 트래킹할 만큼의 빈도로 교정 설정을 증가되게 조정한다. 또한, 일 실시형태에서, 예를 들면, 도 3a에 도시된 드리프트-보상 직병렬화기 회로 (및 예를 들면 도 4b에 도시된 카운터파트 메모리-측 직렬화 회로) 가 그와 달리 단일방향성의 링크 (예를 들면, 제어/어드레스 및 데이터-마스크) 각각에 대해 제공되어, 동일한 교정 절차가 각 시그널링 링크에 대해 수행되는 것을 허용한다. 다른 실시형태에서, 메모리 디바이스로부터의 메모리 제어기로의 다른 리턴 경로 (예를 들면, 사이드밴드 (sideband) 링크) 가 단일 방향 제어기-메모리 링크를 교정하는데 사용될 수도 있다. 그 경우에 단일 방향 링크를 통해 제어기로부터 송신되는 데이터는 메모리 제어기로의 전달을 위한 다른 리턴 경로로 스위칭가능하게 라우팅되어, 메모리 제어기가 원래 송신된 데이터가 메모리 디바이스에 의해 적절히 수신되었는지를 결정가능하게 할 수도 있다.

[0062] 각 드리프트-보상 직병렬화기 및 직렬화기의 초기 교정에 관한 일반적인 접근법은 먼저 비트 정렬 (RxBitAdj[i] 및 TxBitAdj[i]) 그리고 그 다음 패킷 정렬 (RxPktAdj, TxPktAdj[i]) 이 뒤따르는 수신/송신 클럭 위상을 교정 (RxPhAdj[i] 및 TxPhAdj[i] 를 조정) 하는 것이다. 도 5a는, 예를 들면, 데이터 링크 (DQ0 및 DQ1) 을 위한 드리프트-보상 직병렬화기 내 수신 클럭 위상을 교정 (하고 그에 의해 신뢰성 있는 메모리-제어기 시그널링을 보장) 하는 방식을 예시하지만, 동일한 절차가 모든 시그널링 링크들에 대해 동시에 수행되는 것으로 이해되어야 한다. 교정-지원 회로로서, 메모리 제어기는 각 시그널링 링크에 대해 데이터-선택 (멀티플렉싱) 회로 및 매치 회로와, 인트라-비트 위상 정렬, 비트 정렬 및 워드 정렬을 위한 교정 데이터 패턴의 소스들을 포함한다. 또한 메모리 디바이스는 데이터-선택 회로 및 교정 데이터 소스와, 쌍의 링크들 사이 루프-백 상호접속을 포함하여 하나의 시그널링 링크를 통해 수신된 데이터가 쌍의 카운터파트 링크를 통해 메모리 제어기로 리턴가능하게 한다.

[0063] 도 5a로 계속하면, 메모리 제어기는 사이드-밴드 링크 (예를 들면, 도 2a에 도시된 사이드 बैं크, SL) 을 통해 메모리 디바이스로 교정 커맨드를 이슈하여, 데이터-선택기 (멀티플렉서) (477e 및 477o) 를 통해, 각 DQ[0] 및 DQ[1] 시그널링 링크를 통해 메모리 제어기로 송신된 교정 데이터의 소스로서 교정 데이터 소스 (471) 로부터 패턴 세트 A를 선택한다. 짝수 링크 DQ [0]에 대한 전체 교정 데이터 경로는 내부 데이터 교정 데이터 경로 (472) 로부터, 멀티플렉서 (477e, 473e) 를 통해 메모리-측 직렬화기 (235e) 로의 셰이딩 (shading) 된 데이터-흐름에 의해 예시되어 있다. 1.6GHz 메모리 I/O 클럭의 레이트의 2배에서 일어나고 따라서 각 차동 링크 상에서 3.2Gb/s 송신을 산출하는, 차동 메모리 I/O 클럭 (MCK4) 에서 상승 및 하강 에지에 응답하여, 결과적인 직렬 비트스트림이 송신된다. 메모리 제어기에서의 도착시, 비트 시퀀스는 위상-선택 직병렬화기 (192e) 및 비트/패킷 정렬 회로 (194e)에 제공되며 여기서 그것은 수신 클럭, RCK8[i] 및 프레이밍 클럭 신호, RCK1[i]의 초기 위상들 (예를 들면, RxPhAdj[i]=000000b 및 RxBitAdj[i]=000b) 에 응답하여 샘플링 및 직병렬화 (패킷으로 프레이밍) 되고, 초기 패킷 조정 값 (RxPktAdj[i]=00b) 에 따라 패킷-정렬된다. 일 실시형태에서, 패턴 세트 A를 형성하는 데이터 시퀀스는 예정되거나 적어도 예측가능하여 (예를 들면, 결정적으로 생성되어) 수신 데이터는 매치 회로 (453e) 에 제공되어 수신된 데이터가 예상 시퀀스에 매칭하는지를 결정할 수도 있다. 일 구현에서, 이 평가는 비트 만의 시퀀스에 관하여 수행되어, 매치/미스매치 결정은 개별 비트가 예러 없이 그리고 임의의 비트 또는 패킷 오정렬에 상관없이 샘플링되는지 여부에 대해서만 의존한다. 교정 데이터는 병렬 교정 경로를 통해 홀수 및 짝수 시그널링 링크 내에서 동시에 흐른다. 따라서, (DQ[1]) 으로 도시된 홀수 링크의 교정 데이터가 홀수 링크 (477o, 473o, 235o, 192o, 194o) 에 대한 I/O 회로 및 교정-지원 회로를 통해 진행하여 매치 회로 (453o) 에 도착한다.

[0064] 도 5b는 다양한 데이터 선택 경로에 관한 상세 없이 도 5a의 인트라-비트 클럭-위상 배열의 특정 실시형태를 예시한다. 도시된 바처럼, PRBS (pseudo-random bit-sequence) 회로 (501) 는, 패턴 세트 A로서, 시그널링 링크 (DQ[i]) 를 통한 송신을 위해 직렬화 (235) 되고 드리프트-보상 직병렬화기 (186) 내에서 수신되는 결정적인 비트 패턴을 생성한다. 매치 회로 (453) 내 상태 머신 (505) (또는 비트 시퀀서 또는 다른 제어 로직) 은 메모리-측 PRBS 회로 (501) 와 동일한 (즉, 동일한 PRBS 다항식을 계산하는) 방식으로 구현된 제어기-측 PRBS 회로 (503) 로 시드-인에이블 신호 ("시드" (Seed)) 를 초기에 어써팅 (asserting) 하여, 메모리-측과 제

어기-측 PRBS 회로가 동기화되는 것을 가능하게 한다. 인커밍 데이터 시퀀스가 초기 수신 클럭 위상에 의해 적절히 샘플링되면, 제어기-측 PRBS 회로의 출력은, 시딩 (seeding) (즉 PRBS 레지스터 체인으로 시프팅) 후, 메모리-측 PRBS의 출력을, 그것이 제어기-측 PRBS 입력으로서 나타나는 것처럼, 매칭한다. 따라서, 데이터 수신에 에러-프리 (error-free) 하면, 상태 머신은 시드 신호를 디어써팅 (deasserting) 할 수도 있고 이제 시딩된 제어기-측 PRBS 출력이 패턴 세트 A에 대응하는 수신 데이터 시퀀스를, 비트 포 비트 (bit for bit) 로, 계속 매칭하게 된다. 이러한 배열에 의해, 상태 머신 (505) 는 제어기-측 PRBS 출력이 송신된 데이터 시퀀스를 매칭하는지에 따라 (예를 들면, $RxPbAdj[i] = 000000b$ 에 의해 선택된) 초기 수신 클럭 위상에 대해 통과/실패 상태를 어세싱 (assessing) 할 수도 있다. 그 후에, 수신 클럭 위상이 (예를 들면, 바이너리 또는 다른 서지 패턴 다음에 또는 증가되게) 어드밴싱될 수도 있고 적절한 데이터 수신에 새로운 클럭 위상에서 리-테스팅된다. 데이터 아이의 어느 한쪽 단에서 또는 하나의 데이터 아이의 클로스 (close) 및 후행 데이터 아이의 오프닝 (opening) 에서 통과-실패 바운더리 (즉, 통과 및 실패 테스트 결과를 낳는 인접 클럭 위상들) 를 결정하는 것에 의해, 최종 클럭 위상은 통과 클럭 위상들의 범위 중에서, 예를 들면 2개 통과-실패 바운더리 사이의 중간값 (median)으로서, 선택될 수도 있거나 바운더리 중 하나로부터의 특정 오프셋에서, 아이 폭 (eye width) 에 따라 잠재적으로 선택될 수도 있다. 도 5c는 이러한 접근법을 설명하며, 예시적인 위상 선택기에 의해 선택된 48 클럭 위상들 사이의 예시적인 관계 (즉, 6개 클럭 위상들 중 2개를 선택하기 위한 3개 비트, 선택된 클럭-위상 쌍 사이의 인터플레이팅된 3개 비트) 와, (즉, 서로에 대해 오버레이 (overlay) 된 연속하는 데이터 아이들의 플롯인) 데이터-아이 슈무에 관한 통과-실패 바운더리를 나타낸다. 도시된 예에서, 위상-조정 세팅에 대한 실패 클럭 위상은 0-10 및 38-47 이고 통과 위상-조정 세팅은 11 내지 37이다. 따라서, 최종 수신 클럭 위상은 통과/실패 바운더리의 평균 또는 중간값, 이 경우에, 24의 위상 조정 값 $((10+38)/2$ 또는 $(11+37)/2$) 으로서 선택될 수도 있다. 다르게는, 통과-실패 바운더리들 사이의 차이는 유닛-인터벌 $((37-11) \times 7.5^\circ = 202.5^\circ)$ 의 측정으로서 사용되고 따라서, 때때로 본원에서 (반드시 정확하게 하프 유닛-인터벌 일 필요는 없지만) 하프-UI 오프셋으로도 지칭되는, 제한된 수의 에지-센터 오프셋들 사이에서 선택하는데 사용될 수도 있다. 이 접근법은 통과/실패 바운더리와 교정된 샘플링 포인트 (명목 데이터-아이 중간 포인트) 로 정렬된 클럭 위상들 사이의 고속 스위칭을 가능하게 한다. 그러한 "위상 점핑" (phase jumping)은 주기적 타이밍 교정 동작을 더 빠르게 하는데 특히 유용하다. 예를 들면, 아래에 논의된, 일 실시형태에서, 주기적 타이밍 교정은 수신 클럭 위상으로부터 통과-실패-바운더리 클럭 위상 ("바운더리 위상") (boundary phase) 으로의 위상-점핑에 의해 수행되어 마지막 타이밍 교정 동작 이래 통과-실패 바운더리가 드리프트했는지를 결정한다. 그러하면, 바운더리 위상은 드리프트의 방향에서 증가 또는 감소한다. 바운더리 위상을 업데이트한 후, 고정 위상 점프가 업데이트된 바운더리 위상에 대하여 수행되어, 대응하게 업데이트된 (증가 또는 감소) 수신 클럭 위상에 도착하며, 수신 클럭 위상에 대하여 주기적 타이밍 교정을 완료한다.

[0065] 일 실시형태에서, 주기적 타이밍 교정은 메모리-측 유지 동작 (예를 들면, DRAM 리프레쉬) 아래 감추어지거나 잠재적으로 짧은 유희 인터벌에서 수행되고 따라서 상대적으로 짧은 테스트 패턴 송신을 수반하여 통과-실패 바운더리의 드리프트의 검출을 가능하게 한다. 짧은 테스트 패턴은 (더 긴 의사랜덤한 (pseudo-random) 비트 시퀀스에 의해 나타낸) 더 랜덤한 데이터 시퀀스에 존재하는 제한된 수의 스펙트럼의 컴포넌트만을 포함할 수도 때문에, (상이한 통과/실패 바운더리를 갖는) 보다 개방된 데이터 아이는 초기 교정 동안 보다는 PTC (periodic timing calibration) 동안 인식될 수도 있다. 일 실시형태에서, PTC 통과-실패 바운더리는 수신 클럭의 초기 교정 후 결정되고 그 후에 (예를 들면, 바운더리 위상을 리테스팅하는 것에 의해) 드리프트를 검출하는데 사용된다. PTC 바운더리 클럭 위상은 드리프트에 응답하여 증가 및 감소함에 따라, 수신 클럭 위상이 대응하여 증가 및 감소되고, 그에 의해 PTC 실패 바운더리에 관하여 일정한 오프셋에서 수신 클럭 위상을 유지하고 시스템 드리프트를 보상한다. 이 동작은 도 5d에 개념적으로 예시되어 있는데, 이는 더 스펙트럼적으로 완료된 초기 교정 테스트 패턴으로부터 얻어진 보다 작은 데이터 아이 상에 오버레이된 스펙트럼적으로-제한된 PTC 테스트 패턴으로부터 얻어진 더 큰 데이터 아이와, PTC 바운더리 위상과 수신 클럭 위상 사이의 오프셋을 나타낸다. 유사한 동작이 (아래에 설명된) 송신 위상 교정 다음에 수행되며, 미세하게 튜닝된 송신 클럭 위상과 함께 PTC 바운더리 위상을 결정하고 주기적 송신 타이밍 교정 동안 고정 단계 (lock step) 에서의 그러한 클럭 위상들을 증가 및 감소시킬 수 있다.

[0066] 도 6a 및 도 6b는 드리프트-보상 직병렬화 교정의 예시적인 비트-정렬 (또는 패킷-프레이밍) 스테이지를 예시한다. 수신-클럭 교정과 마찬가지로, 비트-정렬 동작이 메모리 제어기로부터 메모리 디바이스로의 사이드-밴드 커맨드 (side-band command) 로 시작되어 예정된 교정 데이터 패턴; 이 경우, 패턴 세트 B: 예정된 비트 포지션에서의 솔리터리 (solitary) '1' 비트를 갖는 8-비트 패킷에 의해 형성된 프레이밍 패턴을 출력하는 것을 시작한다. 데이터 송신 경로는 (메모리-측 멀티플렉서 스테이지, 직렬화기, 데이터 링크, 제어기-측 직병렬

화기 및 비트/패킷 정렬 회로를 통과하여 매치 회로 (453) 에 도착하는 회색 윤곽에 의해 도시된 도 5a에서와 본질적으로 같지만, 각 링크에 대한 매치 회로의 출력은 도 5a의 수신 클럭 교정 동작에서처럼 인터플레이팅된 클럭 위상 대신 비트/패킷 정렬 회로로 전달된 비트 정렬 값 (RxBitAdj[i]) 을 조정한다. 도 6b는 예를 들면, 매치 회로 (453) 내 상태 머신 (505) 에 의해 수행된 전체 비트-정렬 동작을 예시한다. 도시된 바처럼, 상태 머신은 초기 프레이밍 값 (예를 들면, BitAdj = 000) 에서 인커밍 패킷을 평가하여 로직 '1' 비트가 패킷 내 의도된 비트 포지션에 나타나는지를 결정한다. 그렇지 않다면, 상태 머신은 실제 로직 '1' 비트 포지션과 요망되는 '1'비트 포지션 사이의 비트 오프셋을 결정하고, 그에 따라 비트 조정 값을 조정한다. 도 6b에 도시된 2개 비트 오정렬 (또는 패킷-프레이밍 에러) 예들 중 첫번째에서, 로직 '1' 비트는 원래 송신된 비트 포지션 0 (1000 0000b) 대신 비트 포지션 7 (0000 0001b) 에서 프레이밍된다. 그 경우에, 한정 상태 머신 (finite state machine) 은 1 비트의 비트 변위를 결정하고 그에 따라 비트 조정 값을 RxBitAdj[i] = 001b 로 설정하여, 인커밍 비트 시퀀스 내 로직 '1' 의 1-비트 우-시프트 (또는 프레이밍 바운더리에서의 1-비트 좌-시프트) 를 이루고 그에 의해 의도된 패킷-패킷 프레이밍을 확립한다. 제 2 오정렬 예들 중 두번째에서, 로직 '1' 는 비트 포지션 2에서 나타나서, 6 (RxBitAdj[i] = 110) 의 비트 조정이 적용되어 의도된 패킷 프레이밍 바운더리를 이룬다.

[0067] 도 6b 및 도 3a를 계속 참조하면, 비트 정렬이 코어 클럭 도메인에 관하여 프레이밍 클럭을 지연시키는 것에 의해 이루어져서, (수신 클럭의 요망되는 페이징 (phasing) 를 확립하기 위해 사용되는 위상-지연과 함께) 필요한 비트-시프트에 의존하여, 소정의 패킷이 상이한 시간에서 (즉, 코어 클럭 신호의 상이한 에지들에 응답하여) 코어 클럭 도메인으로의 전송을 위해 준비될 수도 있다. 따라서, 메모리 코어로부터 검색된 동일한 멀티-패킷 값의 부분을 형성하는 패킷들이, 정렬 없이, 상이한 코어 클럭 사이클에 응답하여 코어 도메인으로 전송될 수도 있다. 이러한 결과를 피하기 위하여, 도 6c에 도시된 패킷 정렬 동작이 수행되어, 다양한 링크들에 대한 패킷 정렬 회로에 적용될 때, 제어기-코어 클럭 도메인으로의 동시 전송을 위한 메모리 코어로부터 검색된 원래 멀티-패킷 값의 부분을 형성하는 패킷들을 정렬하는 패킷 레이턴시 값을 결정한다. 위상 및 비트 정렬과 마찬가지로, 패킷-정렬 교정은 메모리 제어기로부터 메모리 디바이스로의 커맨드로 시작되어 각 시퀀스의 4개 패킷들 사이 구별을 가능하게 하는 교정 데이터 소스 (패턴 세트 C) 를 선택한다. 예를 들면, 일 실시형태에서, 패턴 세트 C는 3개 0-값의 패킷 ("P0") 이 뒤따르는, 하나 이상의 '1' 비트 ("P1") 을 갖는 패킷을 포함하는 4-패킷 시퀀스이다. 그에 따라, 상태 머신 (505) 은 타 링크들의 것들에 관하여 P1 패킷 수신에서의 레이턴시를 결정할 수도 있는데, 예를 들면 로컬 패킷 레이턴시를 로직 회로에 출력하는 것에 의하고 로직 회로는 다른 링크들에 대하여 같은 신호를 수신하고 가장 레이턴트한 링크의 것에 관하여 로컬 패킷 레이턴시를 나타내는 "상대 레이턴시" 값을 리턴한다. 상태 머신 (505) 은 상대 레이턴시 표시에 따라 I/O-코어 패킷 전송 시간을 지연시킴으로써 응답하고 그에 의해 가장 레이턴트한 링크의 P1 전송 시간을 매칭시켜 P1 패킷이 모든 링크들에 대하여 동시에 (즉, 같은 PCK1 에지에 응답하여) 코어 도메인으로 전송된다.

[0068] 드리프트-보상 직렬화기 내 수신 클럭 위상, 비트 정렬 및 패킷 정렬을 교정한 후에, 비슷한 동작들이 수행되어 드리프트-보상 직렬화기 내 송신 클럭 위상, 비트 정렬 및 패킷 정렬을 교정한다. 일반적으로, 이들 동작들은 교정되는 드리프트-보상 직렬화기를 통해 메모리 제어기로부터 메모리 디바이스로 교정 데이터를 송신하며, 그 다음 앞서 교정된 드리프트-보상 직렬화기를 통해 메모리 디바이스로부터 다시 송신된 교정 데이터를 수신함으로써 수행된다. 보다 상세하게는, 일 실시형태에서, 데이터 루프백 경로가 메모리 디바이스 내에 제공되어 메모리 제어기에 의해 송신된 교정 데이터가 메모리 코어에 부담을 지우지 않고 메모리 제어기로 리턴되는 것을 가능하게 한다. 이 배열은 교정 데이터 리턴을 더 빠르게 하고 타이밍 교정 동작이 메모리 리프 레쉬 또는 메모리 코어 내 다른 오버헤드 동작하에서 부분적으로 또는 완전히 숨겨지는 것 (즉, 동시에 수행되는 것) 을 가능하게 한다. 다른 실시형태에서, 쓰기 및 뒤-읽기 (read-back) 동작이 데이터 링크의 전체-루프 테스트를 확립 (따라서 루프백 경로에 대한 필요를 회피) 하는데 사용될 수도 있으나 그러한 배열은 쓰기 및 읽기 동작이 시작될 수 있기 전 커맨드 경로가 일반적으로 동작할 준비가 갖추어질 필요가 있다는 점에서 전체 교정 절차를 복잡하게 만든다. 데이터 경로 교정에 앞서 (예를 들면, 사이드-밴드 링크를 통해 조정 (coordinating) 된 교정 동작을 통하여; 커맨드 경로를 통해 커맨드를 전송하고 커맨드가 적절히 수신되었는지를 평가하는) 신뢰성 있는 커맨드 경로 시그널링을 확립하는 것이 가능하지만, 이러한 복잡성은 루프백 접근을 통하여 회피된다.

[0069] 도 7a, 도 7b 및 도 8a-도 8c는 메모리 디바이스 내 시그널링 링크들의 각각의 쌍들 사이의 교차-커플링된 루프백 경로에 의존하는 예시적인 직렬화기 교정 절차를 예시한다. 일반적으로, 직렬화기 교정 절차는 직렬화기 교정과 동일한 시퀀스를 따르고, 클럭 위상 조정으로 시작하며, 비트 정렬 (패킷 프레이밍) 이 뒤따르고 그

다음 마지막으로 인터-링크 패킷 정렬이 뒤따른다.

[0070] 도 7a는 드리프트-보상 직병렬화기 내 송신 클럭 위상의 교정 동안 교정 데이터 흐름을 예시한다. 초기에, 교정 데이터 소스는 멀티플렉서 (451e) 를 통해 선택되고 초기 패킷-정렬, 비트-정렬 및 송신-위상 값 (예를 들면, 모든 값은 0) 을 사용하여 드리프트-보상 직렬화기 (191e, 193e) 내에서 직렬화된다. 데이터는 짝수의 시그널링 링크 (이 예에서 DQ[0]) 를 통해 전달되고, 카운터파트 메모리-측 직병렬화기 (236e) 내에서 수신 및 패킷화 (packetizing) 되고, 그 다음 루프백 경로 (240) 및 멀티플렉서 (473o) 을 통해, 짝/홀 링크 쌍 (즉, DQ[1]) 의 카운터파트 링크를 위해, 데이터 송신 회로 및 직렬화기로 라우팅된다. 계속해서, 데이터는 홀수 링크에 대하여 앞서 교정된 드리프트-보상 직병렬화기 (192o, 194o) 내에서 수신되고 매치 회로 (453o) 에 제공되고 이는, 차례로, 테스트 하의 송신 클럭 위상에 대해 통과-실패 결정을 내린다. 도 7b는 소스로부터 destination 으로로의 교정 데이터의 전체 흐름을 예시한다. 보다 상세하게는, 도시된 예에서, 교정 데이터 시퀀스는 제어기-측 PRBS 생성기 (571) 에 의해 소싱 (sourcing) 되고 테스트 하의 송신 위상 조정 세팅 (TxPhAdj[i]) 을 사용하여 메모리 디바이스로 송신된다. 메모리 디바이스 내 수신시, 교정 데이터 시퀀스는 스위칭가능하게-형성된 루프백 경로 (572) 를 통해 메모리 제어기로 루핑 백 (looping back) 하며, 메모리 제어기 내에 수신되고 그 다음 제어기-측 PRBS 체커 회로 (503) 으로 전달된다. 도시된 특정 실시형태에서, 같은 PRBS 체커 회로 (checker circuit) (503) 가 송신 및 수신 교정 동안에 이용되지만, 분리된 (또는 적어도 상이한 구성의) 체커 회로들이 다른 실시형태에서 사용될 수도 있다. 메모리-제어기 전송이 앞서 교정되었기 때문에, 테스트 하의 송신 클럭 위상에 대한 통과 또는 실패 결과는 제어기-측 송신 클럭 및 보상되지 않은 메모리-측 수신 클럭 신호 사이의 부적절한 위상 정렬로부터 얻어지는 것으로 가정될 수도 있다. 따라서, 송신 클럭 위상을 선형 (또는 바이너리-서치 또는 다른 서치) 시퀀스를 통해 증가시킴으로써, 송신 클럭 위상에 대한 통과-실패 바운더리가 식별될 수도 있고, 대응하는 송신 클럭 위상이 그러한 바운더리들 사이에서 중간 포인트로서, 또는 바운더리들 중 하나로부터의 예정된 오프셋으로서, (예를 들면 통과-실패 바운더리들 사이의 클럭 위상들의 범위에 기초하여) 선택된다. 짝수의 시그널링 링크의 각각에 대한 송신 클럭 위상이 완료된 후, 메모리 제어기는 커맨드를 사이드밴드 링크를 통해 이슈 (issue) 하여 멀티플렉싱 배열을 스위칭함으로써 홀수의 시그널링 링크에 의해 송신된 데이터가 (즉, 루프 백 경로 (242) 및 멀티플렉서 (473e) 를 포함하는) 짝수의 시그널링 링크를 통해 루프백되는 것을 가능하게 하고, 짝수의 시그널링 링크들 내의 매치 회로가 홀-링크 송신 클럭들의 위상을 조정하는 것을 가능하게 한다.

[0071] 짝 및 홀 송신 클럭 위상들이 교정된 후, 메모리 제어기는 또 다른 사이드밴드 커맨드를 이슈하여 드리프트-보상 직병렬화기에 대해 수행된 것들에 대응하는 비트 정렬 및 패킷 정렬 동작들의 준비로 짝-링크 직병렬화기 (236e) 와 홀-링크 직렬화기 (235o) 사이의 메모리-측 루프백 경로를 재확립한다. 이러한 배열은 전체 데이터 흐름 경로 (도 8a), 및 짝 루프 시그널링 경로 내 비트/패킷 정렬 회로에 제공된 비트-정렬 신호 (도 8b) 와 패킷-정렬 신호 (도 8c) 를 나타내는 도 8a, 도 8b 및 도 8c에 도시되어 있다. 짝수의 시그널링 링크 내 비트/패킷 정렬이 완료된후, 메모리-디바이스 데이터 멀티플렉서는 (예를 들면, 메모리 제어기로부터의 사이드밴드 링크 커맨드에 응답하여) 다시 스위칭되어 홀수 시그널링 링크 내 비트/패킷 정렬의 교정을 가능하게 한다.

[0072] **주기적 타이밍 교정**

[0073] 위상, 비트 및 패킷 정렬 동작들이 초기 타이밍 교정 노력에서 드리프트-보상 직렬화기/직병렬화기 회로에 대해 완료된 후, 액티브 메모리 동작들이 메모리 디바이스와 메모리 제어기 사이 전체 데이터-레이트 데이터 전송으로 개시될 수도 있다. 위에 논의된 바처럼, 메모리 디바이스 내 드리프트-보상 회로의 결여는 초기 위상 교정들이 온도 및 전압 (또는 다른 환경적인 요인들) 의 변화에 대응하여 요망되는 정렬로부터 벗어나 상대적으로 신속히 드리프트하고 따라서 상대적으로 빈번한 정정 (correction) 을 필요로 할 수도 있다는 것을 의미한다.

주기적 또는 때때로의 교정 동작들이 수행되어 이러한 정정을 제공한다.

[0074] 도 9a 및 도 9b는 각각 드리프트-보상 직렬화기 및 드리프트-보상 직병렬화기를 주기적으로 교정하는데 사용되는 동작들의 예시적인 시퀀스를 예시한다. 도 9a를 먼저 참조하면, 교정 데이터를 송신하기 전에, 메모리 제어기는 루프백-인에이블 커맨드를 새로 교정된 커맨드 경로를 통해 이슈하여 짝수 시그널링 링크로부터 홀수 시그널링 링크로의 루프백된 흐름을 확립하고 또한 클럭 선택 신호 (EdgeClkSel) 에 응답하여 송신 클럭의 위상을 시프트하여 통과/실패 바운더리에 대해 기록된 통과 클럭 위상을 매칭한다. 이 시점에서, 메모리 제어기는, 바운더리-위상 송신 클럭 신호에 응답하여 제한된 수의 시간 (예를 들면 하나의 패킷) 들로 송신될 수도 있는 값들의 예측가능하거나 예정된 시퀀스, 패턴 소스 F로부터 선택된 데이터를 송신한다. 패턴 소스는 메모리 디바이스의 카운터파트 직병렬화기 내에서 수신되며, (모든 링크 쌍들에서 동시에 일어나는) 테스트 하의 링크-쌍의 홀수 링크에 대한 직렬화기로 루프백되고 그 다음 메모리 제어기로 재송신되며 여기서 그것은 (앞서 교

정된 수신 클럭 위상을 사용하여) 드리프트-보상 직병렬화기 내에서 수신되고 예상 값과의 비교를 위하여 매치 회로 (473o) 에 공급된다. 이 포인트에서 패턴 미스매치 (mismatch) 가 검출되면, 메모리 제어기는 통과/실패 바운더리의 통과 위상의 방향으로 통과/실패 바운더리가 드리프트했다는 것을 추론할 수 있고 따라서 위상 드리프트를 트래킹 (tracking) 하는 방향에서 통과/실패 바운더리 값에 대해 기록된 클럭 위상을 증가시킨다.

그 후에, 이제-시프트된 통과/실패 바운더리에 관하여 예정된 오프셋에 의한 위상-점핑에 의해, 증가된 (조정된) 송신 클럭 위상이 이루어지고, 드리프트를 보상한다. 통과/실패 바운더리의 앞서 확인된 통과-위상에 대해 패턴 매치가 검출되면, 앞서 확인된 실패-위상이 테스트되어 (감소된 바운더리 위상을 사용하여 데이터 패턴 F의 송신을 반복하여) 제어기-메모리 위상이 반대 방향으로 드리프트 했는지를 결정한다. 실패-위상이 이제 통과하면, 실패-바운더리 방향의 드리프트가 추론되고, 통과/실패 바운더리에 대해 기록된 클럭 위상이 감소되어 위상 드리프트를 트래킹한다. 그 후에, 이제-감소된 통과/실패 바운더리로의 예정된 오프셋에 의한 위상-점핑에 의해, 감소된 (조정된) 송신 클럭 위상이 이루어지고, 드리프트를 보상한다. 패턴 매치가 통과-바운더리 위상에서 검출되고 패턴 미스매치가 실패-바운더리 위상에서 검출되면, 마지막 교정 동작 이래 통과/실패 바운더리는 드리프트하지 않은 것으로 여겨지고 따라서 송신 클럭 위상 또는 통과/실패 바운더리에 대한 위상 조정 값들의 변화가 기록되지 않는다.

[0075] 상술한 교정 접근법은 각 교정 업데이트 동안, 위상 증가, 위상 감소 또는 유지 (hold) 를 가능하게 하지만, 다른 실시형태에서, 각 교정 동작은 교정되는 (무 유지 상태) 클럭 위상을 증가 또는 감소시킬 수도 있어, 보다 간단한 제어 회로 (예를 들면, 상태 머신) 구현의 이익의 대가로 일부 클럭 디더링 (dithering) 을 겪을 수 있다. 그러한 실시형태에서, 수개의 교정 동작들의 결과들은 축적되고 그 다음 적용되어 다수결 보우트 (majority vote) 에 따라 위상 업데이트를 결정할 수도 있다. 어느 한쪽의 경우에, 바운더리와 샘플링 클럭 위상들 사이에서 오직 단일의 위상 점프가 이루어질 필요가 있다.

[0076] 제어기-측 송신 클럭 위상의 주기적 타이밍 교정이 짝수의 시그널링 링크에 대해 완료된 후, 메모리 제어기는 CA 링크를 통해 교정 커맨드를 이슈하여 데이터 멀티플렉싱 방향을 반대로 하고 교정 동작들의 동일한 시퀀스가 홀수 링크에 대해 수행된다.

[0077] 짝-링크 및 홀-링크 송신 클럭 위상들에 대한 주기적 타이밍 교정 동작들의 완료후, 대응하는 주기적 타이밍 교정 동작들이 도 9b에 도시된 바처럼 수행되어 짝 링크 및 홀-링크 수신 클럭 위상들을 조정한다. 일 실시형태에서, 송신 클럭 위상 조정에 이어 수신 클럭 교정이 뒤따라서 홀-링크 직병렬화기로부터 짝-링크 직렬화기로의 메모리-측 루프백 경로가 이미 확립된다. 그렇지 않으면, 메모리 제어기는 커맨드 경로를 통해 커맨드를 이슈하여 그 루프백 경로를 확립할 수도 있다. 어느 한쪽의 경우에, 홀-짝 루프백 경로가 확립되고 나면, 메모리 제어기는 패턴 F 데이터 (또는 다른 주기적 타이밍 교정 데이터) 를 홀수 시그널링 링크를 거쳐 (즉, 멀티플렉서 (451o) 를 거쳐, 패킷/비트 정렬 회로 (193o) 및 위상-시프팅 직렬화기 (191o) 을 통해) 송신하는 것을 시작하고 짝-링크 직병렬화기를 거쳐 루프백된 데이터를 수신한다. 송신 클럭과 마찬가지로, 수신 클럭 위상이 주기적 타이밍 교정 동안 앞서 기록된 통과/실패 바운더리로 오프셋되어 통과/실패 바운더리 이동했는지와 이동했으면 어느 방향으로 이동했는지를 결정한다. 즉, 통과-바운더리 위상이 여전히 통과 결과를 낳지만, 실패-바운더리 위상은 이제 실패 결과 대신 통과 결과를 낳으면, 통과-실패 바운더리에 대한 비트 조정 값들은 실패-바운더리 위상의 방향으로 시프트되어 위상 드리프트를 상쇄 (counteracting) 하며, 이제-시프트된 통과/실패 바운더리 및 수신 클럭 사이에 유지된 교정 위상 오프셋에 기인하여 수신 클럭 위상에서 대응하는 시프트를 낳는다. 통과-바운더리 위상이 실패 결과를 낳으면, 통과-실패 바운더리에 대한 비트 조정 값이 통과-바운더리 위상의 방향으로 시프트되며, 수신 클럭 위상에서 대응하는 시프트를 낳아 위상 드리프트를 상쇄한다. 통과/실패 바운더리에서 이동이 없으면, 수신 클럭 위상은 변화 없이 남겨진다. 송신 클럭 위상에 대해 논의된 바처럼, 다른 실시형태에서, 수신 클럭 위상은 각 업데이트 (즉, 무 유지 상태) 에서 증가되거나 감소될 수도 있다. 짝수 링크 수신 클럭에 대해 주기적 타이밍 교정이 완료된 후, 메모리 제어기는 메모리 디바이스로 커맨드를 이슈하여 데이터 루프백 접속을 반대로 하고 (짝수의 시그널링 링크들에 의해 송신된 데이터가 홀수의 시그널링 링크를 거쳐 루프백되는 것을 가능하게 하고) 따라서 같은 주기적 타이밍 교정 시퀀스가 홀수 링크 수신 클럭에 대해 수행되는 것을 가능하게 한다.

[0078] 계속 도 9a 및 도 9b를 참조하면, 다른 실시형태에서, 송신 타이밍 교정에 대해 루프백 배열을 사용하는 대신, (코어 저장 어레이 보다는 오히려 예정된 레지스터로 지시된 동작들을 포함하는) 메모리 쓰기 및 뒤 읽기 동작들이 수행되어 송신 클럭 위상의 주기적 교정 (조정) 을 이룰 수도 있다. 유사하게, (즉, 도 6a 를 참조하여 설명된 바와 같은) 메모리 읽기 동작들 및/또는 메모리-제어기 패턴 송신이 수신 클럭 위상을 주기적으로 교정 (조정) 하는데 사용될 수도 있다. 그러한 실시형태에서, 루프백 회로가 부분적 또는 전체적으로 생략될

수도 있고 모든 링크들은 다양한 링크 쌍들의 짝수 및 홀수의 링크들을 순차적으로 교정하는 것 보다는 오히려 동시에 잠재적으로 교정 가능할 수도 있다. 어느 한쪽의 경우에, 링크-바이-링크 (link-by-link) 드리프트 보상을 가능하게 하기 위한 동일한 위상, 비트 및 패킷 정렬 값들 상술한 바처럼 제어기-측 교정 회로 내에서 유지될 수도 있다.

[0079] 일 실시형태에서, 제어기-측 교정 회로에 의해 유지된 비트 조정 값들은 업/다운 정렬 카운터 내에서 유지되어 위상 조정 값에서의 오버플로우 (또는 언더플로우) 가 비트 조정 값으로 캐링 (carrying) (또는 borrowing) 하는 것을 가능하게 하고, 마찬가지로, 비트 조정 값에서의 오버플로우/언더플로우가 패킷 조정 값으로부터 캐링/borrowing하는 것을 가능하게 한다. 이 배열은 도 3a 및 도 3c를 참조하여 설명된 6-비트 위상 조정 회로에 대응하는 정렬-카운터 실시형태 (551) 로 도 10a에 도시되어 있다. 즉, 64개 가능한 위상 조정 값들 중 오직 48개만이 사용되어 (즉, 상위 3개 위상 조정 비트들이 6개 가능한 쌍의 클럭 위상들 중 하나를 선택하는데 사용되어 2개 위상-선택 값들을 미사용 상태로 남기고 따라서 16개의 전체 위상 조정 값들을 미사용 상태로 남겨) 위상-조정 필드 (553) 에 대해 모듈로 48 카운팅 (즉 47로부터 0까지의 증가 및 0으로부터 47까지의 감소) 을 이루는 회로가 위상 카운터 내에 제공된다. 그에 따라, 47의 위상 조정 값이 증가될 때, 얻어지는 롤오버 (rollover) 는 비트 조정 필드 (555) 로의 캐리 (carry) 를 만들어 내며, 소정 비트 시간 내 가장 레이턴트한 예지로부터 더 레이턴트한 비트 시간 내 최소 레이턴트 (가장 어드밴싱된) 예지로 클럭의 위상을 사실상 어드밴싱시킨다. 유사하게, 위상 조정 필드가 최대 카운트 (47) 에 이르고 비트 조정 필드 또한 최대 카운트 (7) 에 이르는 포인트로 위상을 어드밴싱시키는 대로, 클럭 예지에서의 다음 증가가 패킷 바운더리에 교차하여, 위상 조정 값 및 비트 조정 값이 효과적으로 0으로 재설정되고 패킷 조정 필드 (577) 가 증가되어, 더 레이턴트한 패킷 인터벌 내 제 1 비트 시간에서 가장 어드밴싱된 클럭 위상을 선택한다. 유사한 언더플로우가 일어나며, 언더플로잉 위상 조정 값 (0에서 47로의 감소) 에 응답하여 비트 조정 값으로부터 borrowing (감소) 시키며, 언더플로잉 비트 조정 값에 응답하여 패킷 조정 값으로부터 borrowing한다.

[0080] 계속 도 10a를 참조하면, 정렬 카운터 (551) 은 증가/감소 신호 ("inc/dec"), 로드 신호 ("load") 및 업데이트 신호 ("update") 를 수신하기 위한 입력과, 정렬 카운트값 (즉, 도시된 실시형태에서 12-비트 카운트 값) 이 카운트로 로딩되는 것을 가능하게 하기 위한 병렬 포트를 포함한다. 일 실시형태에서, 증가/감소 신호가, 주기적 타이밍 교정 동안 그리고 업데이트 신호의 트리거링 예지 (triggering edge) 에 응답하여 적용되어, 정렬 카운트 (카운터 콘텐츠) 를 증가 또는 감소시킨다. 로드 신호가 초기 교정 동안 어써팅되어 초기 교정 동안 결정된 정렬 값 (예를 들면 프로세서 또는 메모리 제어기 코어 내 다른 회로에 의해 결정된 아이 예지들 사이의 중간 또는 다른 통계적 센터 포인트) 이 정렬 카운터로 로딩되는 것을 가능하게 한다.

[0081] 도 10b는 주기적 타이밍 교정 동안 홀/짝 시그널링 링크 쌍; 본 예에서 DQ[0] 및 DQ[1]에 대한 송신 및 수신 클럭 위상들의 각각을 위하여 정렬 카운터를 업데이트하기 위해 이용될 수도 있는 회로 정렬을 예시한다. 도시된 바처럼, 각각의 4개 클럭 위상들에 대해 2개 정렬 카운터; 링크 마다 4개 정렬 카운터들 그리고 따라서 링크 쌍에 대해 8개 정렬 카운터들 (551_{Tx0}, 551_{Tb0}, 551_{Rx0}, 551_{Rb0}, 551_{Tx1}, 551_{Tb1}, 551_{Rx1}, 551_{Rb1}) 이 제공된다. 링크 DQ[0]을 참조하면, 예를 들면, (DQ[1]은 동일하게 구현됨), 송신-클럭 정렬 카운터 551_{Tx0} 이 제공되어 (비트 및 패킷 정렬을 포함하는) 송신 클럭 위상을 제어하는 한편, 송신-바운더리 정렬 카운터 551_{Tb0} 가 제공되어 대응하는 PTC 바운더리 위상 (즉, 스펙트럼적으로-제한되는 PTC 테스트 패턴이 적용될 때 통과/실패 바운더리에 위치하도록 결정된 송신 클럭 위상) 을 제어 (및 기록) 한다. 유사하게, 수신-클럭 정렬 카운터 551_{Rx0} 이 제공되어 수신 클럭 위상을 제어하고, 수신-바운더리 정렬 카운터 551_{Rb0} 가 제공되어 대응하는 PTC 바운더리 위상 (즉, PTC 테스트 패턴이 적용될 때 통과/실패 바운더리에 위치하도록 결정된 수신 클럭 위상) 을 제어 (및 기록) 한다.

[0082] 도시된 실시형태에서, 각각의 정렬 카운터 (집합적으로, 551) 는 상술한 초기 교정 시퀀스 동안 초기화된다. 일 구현에서, 예를 들면, 최종 교정 클럭 위상이 각 링크에 대해 결정될 때까지 클럭 위상들을 위한 정렬 카운터들이 제어기-코어 회로에 의해 반복적으로 병렬-로딩된다. 그 후에, (스펙트럼적으로-제한된 PTC 테스트 패턴에 응답하여 결정된) 통과-실패 바운더리에 대응하는 최종 바운더리 위상이 각 링크에 대해 결정될 때까지 유사하게 PTC 바운더리 위상들을 위한 정렬 카운터들이 제어기-코어 회로에 의해 반복적으로 병렬-로딩될 수도 있다.

[0083] 도 10b로 계속하면, 멀티플렉서가 (575로 도시된) 정렬 카운터들의 각 클럭-위상/바운더리-위상 쌍과 관련하여

제공되어, 클럭-위상 정렬 카운터 또는 바운더리-위상 정렬 카운터 중 어느 한쪽으로부터 정렬 카운트를 선택함으로써 대응하는 드리프트-보상 직렬화기 (185₀/185₁) 또는 드리프트 보상 직병렬화기 (186₀/186₁) 에 제공되게 된다. 위에서 설명한 교정-클럭 선택 신호 (CalClkSel) 가 제공되어 정렬-카운트 선택을 제어한다. 도시된 특정 예에서, 교정-클럭 선택 신호는 4-비트 신호이며, 각 비트는 정렬 카운터들의 4개 쌍들에 대해 멀티플렉서들 (575) 중 각각의 하나에 공급되어, 양쪽 모두의 데이터 링크의 직병렬화기 또는 직렬화기에 대하여 (PTC를 위한) 바운더리-위상 또는 (라이브 데이터 송신/수신을 위한) 클럭-위상의 선택을 가능하게 한다. 도시된 바처럼, (주기적 및/또는 초기 타이밍 교정과 관련하여 위에서 설명한 한정 상태 머신과 동일하거나 적어도 일부일 수도 있는) 상태 머신 (571) 은 또한 교정-클럭 선택 신호와, 비교 회로 (573) (예를 들면, 상술한 매치 회로의 일부) 의 출력을 수신한다. 도 10c (상태 머신 (571)의 예시적인 상태 도) 및 도 10b를 참조하면, 모든 교정-클럭 선택 비트가 디어썬팅된 상태로 남아 있는 한, 상태 머신은 동작 모드 (581) 로 남고 (예를 들면, 초기 및 주기적 타이밍 교정 동작들에 관하여 상술한 데이터 선택 멀티플렉서에 일반적으로 대응하는) 멀티플렉서 (576) 로 데이터 선택 신호들의 쌍을 출력하여 코어 데이터 레인 (lane), Tdata[0][7:0] 및 Tdata[1][7:0] 를 선택하여, DQ[0] 및 DQ[1] 직렬화기 (185₀, 185₁) 로 각각 전달된 송신 데이터 TxD[0] 및 TxD[1] 을 소싱 (sourcing) 한다. 교정-클럭 선택 비트들 중 어느 하나가 어썬팅되면 (즉, CalClkSel > 0000b) 이면, 상태 머신은 PTC (periodic timing calibration) 모드 (583) 으로 천이하고 여기서 PTC 테스트 패턴은 링크 직렬화기 (185₀/185₁)으로 전달된 송신 데이터의 소스로서 (멀티플렉서 (576)) 를 거쳐 선택되고 또한 제어 신호를 멀티플렉서 (574) 로 이슈하여 교정하의 링크에 따라, 평가를 위한 직병렬화기 (186₀) 또는 직병렬화기 (186₁) 의 출력을 선택한다. 즉, 링크 DQ[1]에 대응하는 2개 CalClkSel 비트들 중 어느 한쪽이 어썬팅되면, 상태 머신은 DQ[1] 직병렬화기로부터 데이터 (즉, RxD[1]) 를 선택하여 비교 회로 (573) 에 공급되게 된다. 그와 다른 경우, DQ[0] 직병렬화기로부터의 데이터 (RxD[0]) 이 비교 회로에 공급된다. PTC 테스트 패턴이 링크 직렬화기에 의해 송신되고, 링크 직병렬화기에 의해 수신되고 비교 회로 내 예상 값과 비교된 후, 상태 머신은 비교 결과가 통과 또는 실패 상태를 나타내는지에 따라, (하지만, 증가/감소 와 통과/실패 사이의 대응이 상황 요구 (circumstances dictate) 에 따라 반대로 할 수도 있지만) 각각 클럭-위상 증가 상태 (585) 또는 클럭 위상 감소 상태 (587) 중 어느 한쪽으로 천이한다. 증가 상태에서, 상태 머신은 증가/감소 출력, "inc/dec" 을 높여서, 증가 동작을 나타내고 (도 10a에 나타냈으나, 도시된 회로 구성을 불분명하게 하는 것을 피하기 위해 도 10b로부터 생략된) 업데이트 신호를 어썬팅하여 대응하는 바운더리-위상 및 클럭 위상 클럭 카운터가 증가되는 것을 가능하게 한다. 마찬가지로, 감소 상태에서, 상태 머신은 증가/감소 출력을 낮추어 감소 동작을 명시하고 업데이트 신호를 어썬팅하여 정렬 카운터의 CalClkSel-선택 쌍이 감소되는 것을 가능하게 한다. 이 동작에 의해, 초기 교정에서의 정렬 카운터로 로딩되는 클럭 위상 및 대응하는 PTC 바운더리 위상들이 각 시그널링 링크에 대해 샘플링 및 송신 클럭을 조정하기 위하여 PTC 바운더리 위상의 이동에 의해 표시된 타이밍 드리프트 정보를 사용하여, 함께 증가 및 감소된다.

[0084] **클럭 정지 저 전력 모드**

[0085] 도 11a는 도 2a의 실시형태 내에 사용되는 예시적인 클럭 배열을 예시하며, 제어기 I/O 클럭 (PCK8) 에 대한 클럭 정지 로직 (601) 및 클럭 버퍼 (603) 그리고 메모리로 포워딩된 데이터-레이트 시스템 클럭 (PCK4) 에 대한 클럭 정지 로직 (605) 및 클럭 버퍼 (607) 를 명시적으로 도시한다. PCK4 클럭-정지 로직 (605) 의 상세도 (610) 을 참조하면, 클럭-인에이블 신호 (ENPCK4) 가 제어기 코어 내 전력-모드 로직에 의해 어썬팅되고 디어썬팅 (본 예에서, 각각 로직 하이 및 로우) 되어 PCK4 클럭을 인에이블링 및 디스에이블링 (또는 시작 및 정지) 한다. 일 실시형태에서, 전력 모드 로직이 제어기 코어 클럭, PCK1에 의해 클럭킹되어, 어썬팅되거나 디어썬팅된 때, 클럭-인에이블 신호는 정수의 PCK1 사이클에 대해 어썬팅되거나 디어썬팅된 상태로 남는다. 이 배열은, 클럭-인에이블 신호가 낮추어져 클럭-정지 저 전력 모드를 확립하고 그 다음 나중에 높여져 클럭을 재시작할 때, 제어기 코어 클럭 (PCK1) 및 메모리 코어 클럭 (MCK1) 사이의 위상 관계가 유지되어, 심지어 클럭 정지 및 재시작을 통한 초기 교정 동안 확립된 비트 정렬 및 패킷 정렬을 보존한다. 본 특정 예에서, 제어기 코어 및 메모리 코어는 동일한 레이트에서 클럭킹됨에 유의한다. 다른 예에서, 제어기 코어 및 메모리 코어는 상이한 레이트에서 클럭킹됨 (그리고 따라서 상이한 깊이의 직렬화 및 직병렬화 파이프라인을 구동할) 수도 있다. 그 경우에, 클럭 정지 인터벌은 가장 긴 피리어드를 갖는 코어 클럭 신호의 정수로 제한될 수도 있다. 예를 들면, 제어기 코어 클럭 레이트가 800MHz로 증가되지만, 메모리 코어 클럭 레이트가 400MHz로 남으면, 클럭 정지 인터벌은 정수의 메모리 코어 클럭 사이클 (이 예에서 2.5nS) 로 제약될 수도 있어, 따라서 제어기 코어 클럭과 메모리 코어 클럭 사이의 위상이 유지되도록 보장하게 된다.

- [0086] 도 11b는 도 11a 클럭킹 아키텍처의 클럭-정지 (또는 클럭 일시정지) 동작의 예시적인 타이밍 도이다. 나타낸 파형들은 메모리 제어기 및 메모리 디바이스 코어 클럭, PCK1 및 MCK1 과, 시스템 클럭, PCK4, 및 클럭 생성 회로에 내부적으로 존재하는 시스템 클럭의 버전, PCK4i을 포함한다. 또한 클럭-인에이블 신호, ENPCK4 및 리타이밍된 버전의 클럭-인에이블 신호, ENPCK4r이 또한 도시되어 있다.
- [0087] 본질적으로, 클럭-인에이블 신호는 내부 PCK4i 클럭을 게이팅 (gating) 하고 따라서 PCK4 시스템 클럭의 토글링을 인에이블링 (enabling) 또는 디스에이블링 (disabling) 중 어느 한쪽을 하게 하는데 사용된다. 제어기 코어 클럭 도메인 및 시스템 클럭 도메인이 서로에 관하여 임의의 위상 오프셋을 갖도록 허용되 (고 클럭-인에이블 신호가 PCK1의 에지에 관하여 임의의 위상 오프셋을 가질 수도 있) 기 때문에, 클럭 인에이블 신호는 PCK4i 클럭의 어느 상태 동안 상승 또는 하강할 수도 있고 따라서, 시스템 클럭을 게이팅하기 위해 직접 적용되면, PCK4i 클럭의 로직-하이 상태 동안에 클럭을 오프 또는 온으로 게이팅할 수도 있고, 그에 의해 시스템 클럭 라인 상의 바람직하지 않은 런트 (runt) (즉, 단축된) 펄스를 낳을 수도 있다. 이 결과 (런트 펄스 생성) 는 코어-도메인 클럭-인에이블 신호 (ENPCK4) 를 PCK4i 클럭 도메인으로 리타이밍하는 한편, 제어기 코어 클럭과 리타이밍된 클럭-인에이블 신호 (ENPCK4r) 의 상승 및 하강 에지들 사이의 고정 위상 오프셋을 유지하는 (상세도 (610) 에 도시된) 리-타이밍 로직 (611) 을 포함하는 것에 의해 도 11a의 클럭-정지 로직 내에서 회피된다. 보다 상세하게는, 도시된 실시형태에서, 시스템 클럭은 PCK4i 클럭의 로직-로우 위상 동안만 온 또는 오프로 게이팅되어, 런트 펄스가 생성되지 않고, 그리고 또한 리타이밍된 클럭-인에이블 신호의 디어써션 (deassertion) 및 어써션 (assertion) 사이의 시간 인터벌이 정수의 코어 클럭 사이클로서 유지되어, 메모리 코어 클럭과 제어기 코어 클럭 사이의 고정-보상된 위상 관계를 보존한다.
- [0088] 도 11a 및 도 11b로 계속하면, 리타이밍된 클럭-인에이블 신호는 게이트 (613) 에서 PCK4i로 AND되고, 따라서, 디어써팅된 때, PCK4 파형에 나타나는 것으로부터 정수의 사이클의 제어기 코어 클럭 (PCK1) 에 대응하는 일정수의 펄스를 블로킹 (blocking) 한다. 이 동작에 의해, PCK4는 게이트 오프 (토글링 방지; 억압) 되고 따라서 깨끗히 정지하고 정수의 PCK1 사이클 (본 예에서 1 사이클) 에 대응하는 인터벌 동안 정지상태로 남는다. PCK4가 메모리 디바이스 내에 수신되고 (예를 들면 버퍼 (223, 229) 에 의해 형성되는) 개방-루프 클럭 구동 회로를 통해 전파되어 데이터-레이트 클럭 신호, MCK4 와, 궁극적으로 메모리-측 송신 및 수신 클럭들을 (주파수 변화 없이) 낳으므로, PCK4의 깨끗한 정지 (또는 일시 정지) 는 그러한 클럭들의 대응되게 깨끗한 정지도 낳아서, 메모리-측 직렬화기 및 직렬화기의 동작을 깨끗이 서스펜딩 (suspending) 한다. 카운터파트 제어기-측 직렬화기 및 직렬화기의 동작들은 또한 클럭-정지 로직 (601) 내의 제어기-측 I/O 클럭의 정지에 의해 깨끗이 서스펜딩된다.
- [0089] 일 실시형태에서, 메모리-측 분할기 회로 (225) 는 (예를 들면 614로 도시된) 증가 로직 (616) 및 상태 레지스터 (621) 를 포함하는) 4-상태 모듈로-4 카운터에 의해 구현되고 여기서 최상위 비트는 MCK4 클럭의 매 2 사이클 후에 토글링하고 따라서 메모리-측 코어 클럭, MCK1으로서 출력된다. 모듈로-4 카운터는 일반적으로 (윤곽이 그려진 MSB로) 카운터 출력에서 도시된 4개 상태들 중 어느 것에서 파워 업 (power up) 될 수도 있음에 유의하면, MCK1 과 제어기 코어 클럭, PCK1 사이의 위상 관계가 MCK4 신호의 사이클 및 따라서 2 비트 시간에 의해 위상-이격된 4개 초기 위상 오프셋들 중 임의의 하나에 대해 취할 수도 있다. 도 2a (및 도 11a) 의 실시형태에서, PCK1 과 MCK4 사이의 이 초기 위상 관계는, 그것이 무엇이든지 간에, 드리프트-보상 직렬화기 및 직렬화기 내 비트 조정 및 패킷 조정 값들의 초기 교정에서 설명된다. PCK1은 PCK4/MCK4 신호의 정지 동안 계속 토글하기 때문에, PCK4-PCK1 클럭 비 (본 예에서 4) 와 같은 PCK4 펄스의 수를 억압하지 않는 PCK4/MCK4의 임의의 정지는 클럭 재시작시 PCK1과 MCK1 사이의 위상 관계를 변화시킬 것이고, 따라서 제어기-코어 클럭 도메인에 관하여 패킷 동기화 및 비트의 손실을 야기한다. 다른 한편으로는, 정수의 PCK1 사이클들에 대해 PCK4를 깨끗이 정지시킴으로써, 억압된 PCK4 (그리고 따라서 MCK4) 펄스의 수는 PCK4-PCK1 클럭 비와 같도록 보장되고 따라서 PCK1-MCK1 위상 관계를 유지할 것이고 그 위상 관계로 초기 교정 세팅이 정렬되고 따라서 적절히 프레이밍되고 패킷-정렬된 데이터가 클럭 재시작시 에러 없이 제어기 코어 도메인으로 전송되는 것을 가능하게 한다. 이 결과는 MCK4의 에지들과 함께 모듈로-2 카운터 (00, 01, 10, 11, 00, ...) 의 상태에 의해 도 11b에 예시되어 있고 클럭-정지 인터벌 동안 PCK4 클럭 펄스들의 $N \cdot (PCK4 - PCK1)$ 비) 의 억압은 클럭 재시작시 PCK1-MCK1 위상 관계를 보존함을 보인다 ('*'은 곱을 표시함).
- [0090] 도 11c 및 도 11d는 시스템-클럭 클럭-정지 로직 (605) 의 보다 상세한 실시형태 및 대응하는 타이밍 도를 예시한다. 클럭-스톱 로직은 도 11a의 게이트 (613) 에 대응하는 로직 AND 게이트 (613) 와, 플립-플롭 (631, 633, 635, 637, 639, 641, 645, 647) 에 의해 형성된 리-타이밍 로직, 로직 엘리먼트 (632, 634, 636), 및 멀티플렉서 (638, 643) 을 포함한다. 논의된 바처럼, 리-타이밍 회로는 제어기 코어로부터의 클럭-인에이블

신호, ENPCK4를 시스템 클럭, PCK4의 도메인으로 리타이밍하는 역할을 한다. 이 동작에서 초기 단계는 클럭-인에이블 신호를 코어 클럭 신호로 샘플링하고 따라서 ENPCK4 내 임의의 천이를 코어 클럭 신호의 천이로 정렬하고 클럭-인에이블 신호 인에이블 신호 (즉, 클럭-인에이블 샘플) 를 나타내는 신호가 적어도 하나의 코어-클럭 사이클에 대해 정상 (steady) 으로 유지되도록 보장하는 것이다. 게다가, 원-타임 로드-스킵 동작이 (로드-스킵 신호 (LD-SKIP) 에 응답하여) 시스템 초기화에서 수행되어 PCK 4 도메인에서 생성되고 PCK4c로 지칭되는 같은-주파수 클럭 신호에 관하여 PCK1 신호의 위상을 결정한다. 보다 구체적으로, 로드-스킵이 높여질 때, 멀티플렉서 (638) 는 PCK4c를 플롭 스테이지 (639) 의 입력으로 보내고, 그에 의해 PCK4c가 PCK1의 뒤따르는 상승 에지에 의해 샘플링되는 것을 가능하게 한다. 스킵 신호로도 지칭되는 플롭 스테이지 (639) 의 출력은 로드-스킵의 디어썬션에 의해 래칭되고, PCK4c가 PCK1 상승 에지에서 하이 또는 로우였는지 각각에 따라 로직 '1' 또는 로직 '0'이 될 것이다. PCK4c와 PCK1 사이의 위상 관계가 시스템 동작 동안 변화지 않기 때문에, 로드-스킵은 스킵 신호의 상태를 리졸빙 (resolving) 하기 위해, 파워 업 (또는 시스템 리셋) 에서, 오직 한번 어썬팅될 필요가 있다.

[0091] 계속해서, 클럭 인에이블 신호 ENPCK4가 PCK1의 상승 에지에 응답하여 플롭 스테이지 (637) 에 의해 샘플링되어 상승-PCK1-에지-정렬 클럭-인에이블 신호, ENPCK4a 를 생성하고 이는 정수의 PCK1 사이클 동안 같은 상태로 남도록 보장된다. ENPCK4a는 그 자체가 후행 하강 PCK1 에지에 응답하여 플롭 (641) 에서 샘플링되어 네가티브-PCK1-에지 정렬된 클럭-인에이블 신호, ENPCK4b를 생성하고, 또한 정수의 PCK1 사이클에 대해 같은 상태로 남도록 보장된다. 도 11d에 도시된 바처럼, 2개 PCK1-정렬 클럭-인에이블 신호, ENPCK4a 및 ENPCK4b 는 PCK1-정렬 클럭-인에이블 신호의 예들을 나타내고 이들은 같은 시간 인터벌 동안 유효하지만 다른 상황; 스킵 신호 (skip signal) 가 하이 (ENPCK4a) 인 한 경우와 스킵 신호가 로우 (ENPCK4b) 인 다른 경우이다. 그에 따라서, 스킵 신호의 상태에 따른 멀티플렉서 (643) 에서 2개 PCK1-정렬 클럭-인에이블 신호들 사이에서 선택함으로써, 같은 시작 및 정지 시간을 갖는 PCK1-정렬 클럭-인에이블 신호가 선택되고 어느 한쪽의 경우에 리-타이밍 플롭 스테이지 (645) 로 출력된다. 게다가, 선택된 PCK1-정렬 클럭-인에이블 신호는 리-타이밍 플롭 스테이지 (645) 의 트리거 입력 (trigger input)에 공급된 쿼드러썬어 클럭 신호 PCK4d (즉, PCK1 및 PCK4c와 같은 주파수를 갖지만, PCK4c에 관하여 쿼드러썬어 위상 관계를 갖는 클럭 신호) 의 상승 에지를 스패닝 (spanning) 하도록 보장된다. 마지막으로, PCK4d의 각 에지가 (플롭 (631) 에 의해) 데이터-레이트 클럭 PCK4i의 네가티브 고잉 에지에 응답하여 생성되기 때문에, 리-타이밍 플롭 스테이지 (645) 를 트리거링하고 따라서 PCK1-정렬 클럭-인에이블 신호를 샘플링하는데 사용되는 PCK4d의 상승 에지는 PCK4i가 로우가 된 직후에 일어난다. 이러한 설계에 의해, 셋업 및 유지 시간의 전체 (또는 거의 전체) PCK4i 클럭 사이클이 리-타이밍된 클럭-인에이블 신호, ENPCK4c가 다른 리-타이밍 플롭 스테이지 (647) 에서 PCK4i의 다음 하강 에지에 의해 샘플링되기 전에 제공되어 최종 리-타이밍 클럭-인에이블 신호, ENPCK4r 을 생성하는데, 이는 PCK4를 온 및 오프로 게이팅 (gating) 하는데 사용된다. 도 11d에 나타난 바처럼, 순 결과는 최종 리-타이밍된 클럭-인에이블 신호, ENPCK4r이 PCK4i의 로우-고잉 에지에 응답하여서만 그리고 정수의 PCK1 클럭 사이클이 마지막 상태 변화 이래 일어난 다음에만 상태를 변화시킨다는 것이다. 이 동작에 의해서, PCK4 출력 상의 문제가 되는 런트 펄스가 회피되고 PCK1과 PCK4 사이의 클럭 위상 관계가 클럭 정지 및 재-시작을 통해 유지되어, 메모리 제어기 내 드리프트-보상 직렬화기 및 직병렬화기 회로의 교정 상태를 보존한다.

[0092] 도 11c 및 도 11d에 도시된 기법 및 회로는 또한 제어기-측 I/O 클럭, PCK8에 대한 클럭-정지 로직 내에서 적용되어, PCK8 클럭 라인 상의 런트 펄스를 회피하고 클럭 정지 동안 디스에이블링 (또는 억압된) PCK8 펄스들의 수가, 2:1 클럭 비 처리, 디스에이블링된 PCK4 펄스들의 수와 매칭되도록 보장할 수도 있다.

[0093] 도 11e - 도 11g는 다른 클럭-정지 아키텍처 (650) 및 대응하는 회로 및 타이밍 도를 예시한다. 도 11a의 아키텍처에서 분리된 PCK4 및 PCK8 클럭-정지 회로에 대조적으로, 클럭-정지 아키텍처 (650) 는 단일 클럭-정지 로직 회로 (651) 을 포함하고 이는 내부 PCK8 클럭 위상 (PCK8i) 의 토글링을 디스에이블링하고 이는 최종 PCK8 클럭 위상과, 분주 (frequency division) 후에, 시스템 클럭 신호 PCK4를 낳는다. 클럭-정지 회로 (601, 605) 의 부재와, 다중 클럭-인에이블 신호 대신에 솔리터리 클럭-인에이블 신호 (ENPCK8/4) 의 제공을 제외하면, 아키텍처 (650) 의 기능성 엘리먼트는 일반적으로 도 11a에서 그들의 같은 넘버링의 카운터파트에 참조하여 설명된 것처럼 동작한다. 또한, 도 11a의 실시형태에서처럼, 제어기 코어 클럭, PCK1 은 PCK8i 클럭 위상 (그리고 따라서 PCK8 및 PCK4 클럭) 이 정지된 후에 계속 토글링할 수도 있다.

[0094] 도 11f는 도 11e의 클럭-정지 회로 (651) 를 구현하는데 사용될 수도 있는 클럭-정지 회로 (670) 의 실시형태를 예시한다. 도시된 바처럼, 제어기 코어 도메인으로부터의 클럭-인에이블 신호 (ENPCK 8/4) 는 플립 플롭 (671) 에서 베이스라인 PLL 출력 위상, PLL [0°] (코어 클럭에 관하여 8× 주파수를 갖는 클럭 위상) 에 응답

하여 샘플링되어, 클럭-인에이블 신호를 PLL-출력 클럭 도메인으로 리-타이밍된 인에이블 신호 (672) 로서 리-타이밍한다. 다른 리-타이밍 회로들이, PLL 출력 클럭 도메인으로 최종 리타이밍하기 전에 클럭-인에이블 신호를 타이밍 도메인의 시퀀스를 통해 전송하는 스테이지드 리-타이밍 회로 (staged re-timing circuit) 를 포함하는, 다른 실시형태들에서 클럭-인에이블 신호를 리-타이밍하는데 사용될 수도 있다. 리-타이밍된 인에이블 신호 (672) 는 PLL [0°] 의 하강 에지에 응답하여 샘플링되어 PLL [0°] 클럭 신호의 로직-로우 하프-사이클의 시작에서 클럭0-인에이블 신호 (clk0-en) 를 낮춘다. 멀티플렉서 (674) (또는 다른 선택기 회로) 는 대응하는 PCK8i 출력 (PCK8i[0°]) 을 PLL[0°] 으로부터 디커플링하고 PCK8i 출력을 접지로 커플링하여 출력을 로우로 유지하고 클럭 정지를 이룸으로써 낮추어진 클럭0-인에이블 신호에 응답한다. 리-타이밍된 클럭 인에이블 신호 (672) 는 유사하게 PLL[60°]의 하강 에지에 의해 샘플링되어 PLL[60°] 클럭 신호의 로직-로우 하프 사이클의 시작에서 클럭1-인에이블 신호 (clk1-en) 을 낮춘다. 멀티플렉서 (676) 는 PLL[60°] 으로부터 PCK8i[60°] 출력을 디커플링하고 출력을 접지에 커플링함으로써 낮추어진 클럭1-인에이블 신호에 응답한다. 마지막으로, (예를 들면, 클럭0-인에이블 신호의 버퍼-지연된 예에 의해 생성된) 리-타이밍된 클럭-인에이블 신호 (680) 의 보다 지연된 예는 PLL[120°] 의 하강 에지에 응답하여 샘플링되어 PLL [120°] 클럭 신호의 로직-로우 하프-사이클의 시작에서 클럭2-인에이블 신호 (clk2-en) 를 낮춘다. 멀티플렉서 (678) 는 PLL[120°] 으로부터 PCK8i[120°] 출력을 디커플링하고 출력을 접지로 커플링함으로써 낮추어진 클럭2-인에이블 신호에 응답한다. (대시의 윤곽으로 억압된 클럭 펄스를 나타내는) 도 11g의 셰이드된 클럭-정지 지역에 의해 도시된 바처럼, 낮추어진 클럭-인에이블 신호에 응답하여 PCK8i 클럭 위상들의 깨끗한 정지는 대응하여 PCK8[0°, 60°, 120°] 클럭 위상, PCK4 클럭 위상 및 따라서 메모리-측 클럭, MCK4 (및 MCK1, 미도시) 의 깨끗한 정지를 낳는다. 모든 정지된 (또는 일시 정지된 또는 디스에이블링된) 클럭의 깨끗한 재시작은 클럭-인에이블 신호 (ENPCK8/4) 를 높임으로써 유사하게 이루어진다. 즉, (각 PLL 출력 위상에 대한 셋업 및 유지 시간 요건을 만족시키는데 필요한 더 리-타이밍된) 클럭-인에이블 신호의 상승 에지는 리-인에이블링될 PLL 클럭 위상의 로우-고잉 에지에 응답하여 샘플링되며, 각 PLL 클럭 위상에 대한 로직-로우 인터벌의 시작에서 멀티플렉서 선택을 스위칭하여 PLL 클럭 위상의 PCK8i 클럭 노드로의 글리치리스 리-커플링 (glitchless re-coupling) 을 가능하게 한다. 비록 도 11f 및 도 11g에 구체적으로 도시되지는 않았지만, 0°, 60° 및 120° PLL 클럭 상보 예들 (180°, 240° 및 300°) 의 상보 예가 유사하게 각각 클럭0-인에이블, 클럭1-인에이블 및 클럭2-인에이블 신호의 상태에 따라 디스에이블링 및 인에이블링될 수도 있다. 또한, 상술한 다양한 실시형태들에서 처럼, 보다 많거나 보다 적은 PLL 출력 위상들이 다른 실시형태에서 생성될 수도 있다.

[0095] **클럭-정지 모드의 진입 및 종료-시스템 동작**

[0096] 일 실시형태에서, 클럭-정지 저 전력 모드는 메모리 제어기가 모든 리퀘스트된 메모리 트랜잭션을 완료하고 따라서 작업중이 아닐 때마다 진입된다. 일 실시형태에서, 이 유휴 상태는 헨딩 트랜잭션의 큐 ("트랜잭션 큐") 를 모니터링하는 메모리 제어기 코어 내의 전력 모드 로직에 의해 결정되고 따라서 트랜잭션 큐가 비워질 때마다 알려진다. 큐를 비우자마자 즉시 제어기 I/O 및 시스템 클럭을 정지시키기 보다는, 메모리 디바이스 및 제어기 I/O 회로의 견지로부터 적어도, 전력 모드 로직은 큐로부터 빼낸 마지막 트랜잭션 (즉, 최종 트랜잭션) 이 완료될 만큼 길게 적어도 대기하고, 그 다음 클럭 인에이블 신호, ENPCK4 및 ENPCK8 를 디어셔팅하여 제어기 I/O 및 시스템 클럭 신호 (각각, PCK8 및 PCK4) 를 깨끗이 정지시킨다.

[0097] 도 12a는 클럭-정지 저 전력 모드 진입 및 이로부터의 종료를 포함하는 인터벌 동안 메모리 제어기에서의 클럭 신호, 클럭-인에이블 신호 및 커맨드/어드레스 신호의 예시적인 타이밍 도이다. 클럭 신호는 제어기 코어 클럭 PCK1, 시스템 클럭 신호, SCK (메모리 제어기 내 PCK4), 및 제어기 I/O 클럭 PCK8 를 포함한다. 상술한 예시적인 실시형태들을 계속하면, 데이터 및 커맨드는 3.2Gb/s; 0.625nS 시스템 클럭 사이클 당 2 비트 및 2.5 nS (나노초) 코어 클럭 사이클 당 8비트로 송신된다. 이러한 배열에 의해, 8:1 직렬화 파이프라인이 확립되며, 아웃고잉 정보가 코어 클럭 신호의 소정 사이클 동안 8-비트 패킷 (즉 바이트) 으로서 각 드리프트-보상 직렬화기에 제공되는 한편, 앞서 제공된 패킷의 비트들이 그 동일한 코어 클럭 사이클 동안 각각의 비트-시간 (비트-시간 또는 $t_{BIT}=t_{PCK8}$ (PCK8 주기)) 에서 직렬로 송신된다. 따라서, 도 12a에 도시된 바처럼, $t_{PCK1}=t_{PKT}=4*t_{SCK}=8*t_{BIT}$ 이고 여기서 '*' 는 곱을 표시한다. 상이한 송신 주파수들, 클럭 비, 직렬화 비 및 패킷 사이즈가 다른 실시형태에서 선택될 수도 있다.

[0098] 메모리 제어기 내, 데이터의 패킷 및 커맨드/어드레스 (CA) 비트들이 8-비트 와이드 데이터 라인 및 CA 라인들 각각을 통해 I/O 회로에 공급된다. 일 실시형태에서, 각 메모리 액세스 커맨드 및 대응하는 어드레스가 2개의 8-비트 패킷들로 패킹되고 이것들은 따라서 단일 패킷-시간에서 2개의 CA 링크들 (CA[0] 및 CA[1]) 을 통해 송신될 수도 있다. 전송할 패킷이 남지 않았을 때, "NOP" 커맨드 패킷 (예를 들면, 제로-필드 패킷 (zero-

filled packets)) 으로서 표기되는, "무-동작" 커맨드가 커맨드 경로 (CA[0], CA[1]) 를 통해 메모리 디바이스로 송신되고, 제어기 코어는 커맨드 경로 상에 송신되는 마지막 메모리 액세스 커맨드 ("최종 커맨드") 의 완료까지 카운트다운을 시작한다. 카운트다운 동안, 클럭-정지 모드는 펜딩인 것으로 불려지고, 제어기 코어 내 전력-모드 로직은 프리-클럭-정지 상태 (pre-clock-stop state) 에 있고 여기서 모든 클럭들은 계속 토클링하여 최종 쓰기 동작의 경우에 메모리 코어 내에 쓰기 데이터가 저장되는데 필요하거나, 읽기 데이터가 메모리 코어로부터 리턴되어, 최종 메모리 읽기 커맨드의 경우에 제어기 I/O-코어 인터페이스에서 직렬화 및 제공되는데 필요한 타이밍 에지를 제공한다. 최종 커맨드와 연관된 모든 동작들이 메모리 디바이스 및 제어기 I/O 회로 내에 완료될 때까지 트랜잭션 큐 내에 새로운 트랜잭션이 큐잉되지 않으면, 전력-모드 로직은 클럭-인에이블 신호, ENPCK4 및 ENPCK8 를 시스템 클럭 및 제어기 I/O 클럭, PCK4 (SCK) 및 PCK8, 각각에 대해 디어썬트한다.

[0099] 도 12a 내, 클럭-정지 모드로의 진입이 16-비트 와이드 트랜잭션 큐 (T-Queue[15:0]) 로부터 제어기 I/O 회로로 8-비트 와이드 커맨드 라인 Cadata[0][7:0] 및 Cadata[1][7:0] 를 통해 최종 나머지 메모리 액세스 리퀘스트의 전송 (제거 또는 디큐잉) 으로 시작된다. 커맨드 데이터 라인들 그 자체가 패킷-정렬 회로 (즉, 패킷 정렬 FIFO 또는 스킵 회로) 내에서 구현될 수도 있고 이것은 소정 시그널링 링크에 대하여 코어 클럭 도메인으로부터 프레이밍 클럭 도메인으로의 교차를 가능하게 한다. 따라서, 각 연속하는 커맨드는, NOP 이든지 또는 메모리-액세스 커맨드 (OP) 이든지, 코어 클럭 (PCK1) 의 하강 에지에 응답하여 패킷-정렬 회로 내에서 포워딩되고, 교정된 (및 링크-특정) 수의 비트-시간들 후에 프레이밍 클럭 도메인으로 전송되고 그 다음 CA[0] 및 CA[1] 링크를 통해 송신을 위해 직렬화될 수도 있다. 따라서, "OP0," 으로 지정된, 최종 동작이 트랜잭션 큐로부터 커맨드 라인들로 시간 (702) 에서 전송되고, (하강 PCK1 에지에 응답하여) 시간 (704) 에서 시작하는 패킷-정렬 회로를 가로질러 포워딩되고 그 다음 패킷-정렬 회로로부터 직렬 시프트 레지스터로 (예를 들면 도 3d에 도시된 플립-스태이지 (315) 에 의해 형성된) 제어기-측 직렬화 회로 내에서 직렬화 지연, t_{SERIAL} 후에 전송되는데, 이는 디-프레이밍 클럭 신호 (도 3d에서의 TCK1[i]) 와 PCK1 사이의 비트-와이즈 오프셋을 더한 PCK1 사이클에 대응한다. 그 후에, 데이터는 직렬 시프트 레지스터로부터 비트 바이 비트 (bit by bit) 로 시프트되어 CA[0]/CA[1] 시그널링 링크를 통한 직렬 데이터 송신을 이룬다.

[0100] 상이한 시그널링 링크들 사이의 비트-가변성이 CA[0] 및 CA[1] 시그널링 링크에 대한 직렬화 지연들 사이의 4-비트-시간 차이에 의해 도 12a에서 강조되어 있다. (즉, PCK1 과 TCK1[i] 사이의) 링크 CA[0] 에 대해 디-프레이밍 클럭과 코어 클럭 사이의 비트-와이즈 오프셋이 제로여서, OP0의 로우-오더 패킷 (low-order packet) 이 트랜잭션 큐로부터 CA[0]패킷-정렬 회로로 전송된 후 1 PCK1 사이클 ($t_{SERIAL}=8$ 비트-시간)에 CA[0]링크에 대한 직렬 시프트 레지스터로 전송되고, 따라서 시간 (706) 에서 시작하는 CA[0] 시그널링 링크를 통해 비트 직렬로 송신된다. 따라서, 직렬화 지연, t_{SERIAL} 은 1 PCK1 사이클 또는 8 비트-시간이다. 대조적으로, 4-비트 오프셋이 (즉, PCK1 과 TCK1[i+1] 사이의) 링크 CA[1] 에 대해, 코어 클럭과 디-프레이밍 클럭 사이에 존재해서, 트랜잭션 큐로부터 CA[1] 패킷-정렬 회로로의 OP0의 하이-오더 패킷의 전송 다음에, 12-비트-시간 직렬화 지연이, 패킷이 (시간 (708) 에서 시작하는) CA[1] 시그널링 링크를 통해 패킷이 전송되기 전에, 경과 (elapsing) 또는 일어난다. 대체로, 8-비트-시간과 12-비트-시간 직렬화 지연들 사이의 차이는 OP0의 로우-및 하이-오더 패킷들의 제어기-측 송신 사이 4-비트-시간 오프셋 (또는 비트-가변성) 을 낳으며, CA[0] 와 CA[1] 링크에 대한 송신 클럭들 사이에 존재할 수도 있는 임의의 서브-비트-시간 위상 오프셋 (즉, TCK8[i] 과 TCK8[i+1] 사이의 위상 오프셋) 을 카운팅하지 않는다. 대체로, 비트-가변성 및 서브-비트 위상 오프셋은 연관된 커맨드/어드레스 패킷들 및 데이터 패킷들의 시간-스태거드 송신을 발생시켜 메모리-측 데이터 샘플링, 직렬화 및 I/O-코어 전송을 모두 메모리-측 코어 조정 회로 없이 가능하게 한다. 비록 도 12a에 도시되지 않았지만, 유사한 비트-가변성 및 서브-비트 위상 오프셋이 제어기-측 데이터 직렬화기 내에서 허용되어 메모리-측 코어-I/O 전송, 직렬화 및 데이터 송신을 메모리-측 클럭 조정 회로 없이 가능하게 한다.

[0101] 도 12a에 있는 클럭-정지 예로 계속해서, 링크 CA[0] 및 CA[1] 에 대해, 트랜잭션 큐로부터 커맨드 라인으로의 최종 메모리 액세스 동작 (즉, OP0) 을 전송할 때, 전력-모드 로직은 트랜잭션 큐가 빈 상태라는 것을 결정하고 따라서 시스템 클럭 및 제어기 I/O 클럭에 대한 클럭-인에이블 신호들의 디어썬트까지 카운트다운을 시작한다. 일 실시형태에서, 카운트다운 시간은 동작 특정되고 따라서 $t_{CA(OP)-EN}$ 으로서 명시되며, "OP" 는 메모리 액세스 리퀘스트의 성질을 나타낸다 (예를 들면, 프로그램 및 이레이즈 (erase) 시간과 같은 다른 비-DRAM 타입의 메모리 저장에 특정한 동작 시간들이 적용될 수 있지만, 로우 동작 (row operation), 이를테면 액티브화 (activate) 또는 프리차지 (precharge), 또는 칼럼 동작 (column operation), 이를 테면 메모리 읽기 또는 메모리 쓰기). 다르게는, 고정된 카운트다운 시간이, 수행되는 동작의 타입에 상관없이 적용될 수도 있다. 어느 하나의

경우에, 목표는 충분한 클럭킹 에지가 메모리 디바이스 및 제어기 I/O 회로에 제공되어 마지막 메모리 액세스 동작을 완료하는 것이다. 일반적으로, 트랜잭션 큐를 비우는 사이의 워스트-케이스 레이턴시는 메모리 읽기 동작에서 일어나고, 이는 (워스트-케이스 비트 가변성을 포함하는) 커맨드 직렬화 시간, 커맨드 경로를 통한 전파, (집합적으로 본원에서 CAS 레이턴시로 지칭되는) 메모리 디바이스의 직렬화 레이턴시 및 데이터 검색, 데이터 경로 상의 읽기 데이터 전파 시간, 및 마지막으로 제어기-측 데이터 직렬화 시간을 포함한다. 동작-특정 실시형태에서, 전력-모드 로직은 최종 동작에 기초하여 레지스터 뱅크 (또는 룩업 테이블) 를 인덱싱하고 그에 의해 카운트다운 값 (예를 들면, 클럭-인에이블 신호를 디어썬트하기 전에 일어나는 코어 클럭 사이클의 수) 를 검색할 수도 있다. 고정-카운트 실시형태에서, 카운트다운 값은, 메모리 읽기 동작을 완료하기 위하여 워스트-케이스 시간의 시스템 시작 기반 실행-시간 또는 생성-시간 또는 설계-시간 측정에서, 또는 시스템 생성 시간에서 원-타임 레지스터 (one-time register) 를 프로그래밍하거나 심지어 하드-와이어드 (hard-wired), 워스트-케이스 카운트 값을 구현함으로써 프로그래밍될 수도 있다.

[0102] 어떻게 구현되든, 새로운 메모리 액세스 리퀘스트가 클럭 정지까지의 카운트다운 동안 (즉, 전력 모드 로직이 클럭-정지-펜딩 모드에 있는 동안) 트랜잭션 큐로 삽입 (또는 그렇지 않으면 수신) 되면, 펜딩 클럭-정지는 중단되고 전력 모드 로직은 액티브 모드로 리턴되며, 빈 상태에 대해 트랜잭션 큐를 계속 모니터링한다. 그러나, 새로운 메모리 액세스 리퀘스트가 카운트다운 완료 전에 트랜잭션 큐 내에서 큐잉되지 않으면, 전력 모드 로직은 클럭-인에이블 신호, ENPCK4 및 ENPCK8 을 디어썬트하며, 따라서 클럭-정지 동작을 트리거링한다.

[0103] 위에 설명된 바처럼, ENPCK4 및 ENPCK8이 코어 클럭 도메인 내에서 생성되고 따라서 PCK4 및 PCK8 도메인 내에서 리타이밍되어 PCK4 및 PCK8 클럭들의 깨끗한 정지 (또는 일시정지 또는 디스에이블링) 을 보장한다. 게다가, PCK4 및 PCK8 도메인들 (즉, 시스템 클럭 및 제어기 I/O 클럭 도메인들) 이 (도 2a의 실시형태에서 처럼) 서로로부터 위상 오프셋되도록 허용되는 실시형태에서, 리-타이밍된 클럭-인에이블 신호, ENPCK4r 및 ENPCK8r 의 디어썬션 시간이 상이하야, PCK4 및 PCK8 클럭들에 대한 상이한 클럭 정지 시간을 발생시킬 수도 있다. 도 12a의 예시적인 타이밍 도에서, 클럭-정지 시간들에서의 이러한 변화는 PCK4 클럭-정지 시간과 PCK8 클럭-정지 시간 사이의 2 비트 시간 (2 PCK8 사이클, 1 PCK4 사이클) 오프셋에 의해 나타내어 진다. 즉, PCK8 클럭-정지 로직에서의 리-타이밍 지연, t_{C8DL} (또는 디스에이블 레이턴시) 이 PCK4 클럭-정지 로직에서의 리-타이밍 지연, t_{C4DL} 에서 보다 2 비트-시간 더 길다. 양쪽 모두의 클럭들이 정수의 PCK1 사이클 동안 정지되기 때문에, 같은 2-비트-시간 오프셋이 클럭 재-시작에서 적용되어 같은 수의 클럭 펄스들이, PCK8 및 PCK4 클럭 도메인에서, 임의의 클럭 비 (이 예에서는 2:1) 를 처리한 후, 생성된다.

[0104] 제어기 I/O 클럭을 정지시키는데 있어 하나의 중요한 도전은 다양한 제어기-측 직렬화기/직렬화기 회로 내에 허용되는 비트-가변성에 의해 설명된다. 즉, 비트 가변성이 (실제로, 카운터파트 비보상 메모리-측 타이밍 도메인과의 정렬을 이루기 위해 필요한 그러한 도메인들을 스테거링하는) 상이한 시그널링 링크들에 대한 제어기-측 타이밍 도메인들 사이에서 허용되기 때문에, 상이한 링크들에 대한 패킷 바운더리 자신이 오프셋된다. 클럭-정지 관점으로부터, 제어기 I/O 클럭이 어디에서 정지되든지, 하나 이상의 CA 패킷들이 오직 부분적으로 직렬화되며, 사실상 클럭 정지 인터벌 (예를 들면 711 및 712) 의 어느 한쪽의 측 상에 나타나는 부분들로 패킷을 분열 (fracture) 시킬 수도 있다. 하지만, PCK8이 정수의 PCK1 사이클들 동안 깨끗이 정지되기 때문에, 패킷 (712) 의 나머지 부분들이 클럭 재-시작 시에 적절히 직렬화되고 새로운 패킷이 제어기-측 디프레이밍 클럭 및 코어 클럭, PCK1 사이의 사전-확립된 관계를 반영하는 패킷 바운더리에서 디프레이밍되고 송신된다. 즉, 제어기 코어 클럭 (PCK1) 과 디-프레이밍 클럭 (예를 들면, TCK8[i]) 사이의 비트-와이즈 (및 인트라-비트 위상) 오프셋이 유지되어 임의의 클럭-정지-분열된 패킷의 나머지 비트들이 송신되고 클럭-정지가 일어나지 않은 것처럼 새로운 패킷이 디프레이밍된다. 시각적으로, 이것은 시작-클럭 바운더리를 따라 도 12a의 도를 슬라이싱하고 클럭-정지 바운더리와 일렬로 세워지게 남겨진 클럭 재-시작 후 나타나는 도의 부분을 시프팅함으로써 상상될 수도 있다. 알 수 있는 바처럼, 패킷 프레이밍 바운더리는 유지되어 모든 클럭-정지-분열된 패킷들이 클럭 재-시작시에 전부 만들어진다.

[0105] 지금까지 설명된 클럭-포워딩 아키텍처에 대해 고려하면, 임의의 수의 시스템 클럭 펄스들이 메모리 디바이스로 플라이트 (flight) (즉, 클럭 링크 상에서 전파하는) 상태일 수도 있기 때문에, 소정 시그널링 링크 상의 비트의 수신 또는 송신을 가능하게 하는데 사용되는 특정 시스템 클럭 에지가 일반적으로 제어기 I/O 클럭의 명목상 정렬된 에지에 관하여 시간적으로 오프셋될 것이다. 즉, 제어기 I/O 클럭 에지 및 시스템 클럭 에지가 제어기-측 클럭 생성기로부터 동시에 출력된다고 가정하면, I/O 클럭 에지가 일반적으로 제어기-측 I/O 회로 내 데이터 수신 이벤트를 타이밍하는데 적용되는 한편, 시스템 클럭 에지는 여전히 메모리 디바이스 또는 메모리 디바이스 I/O 회로로 엔-라우팅 (en-routing) 된다. 클럭-정지 관점으로부터, 이것은 시스템 클럭 및 제어기

I/O 클럭이 메모리 제어기에서 동시에 정지된 경우에도, 보다 긴 시스템 클럭 파이프라인이 드레인 (drain) 하는데 더 길게 걸리므로, 메모리 디바이스는 그럼에도 불구하고 (클럭 비를 처리하는) 제어기 I/O 회로 보다 더 많은 클럭 에지들을 겪게 된다. 그리고 비슷하게, 클럭 시작 관점으로부터, 시스템 클럭 및 제어기 I/O 클럭이 동시에 시작되면, 제어기 I/O 회로는 보다 긴 메모리-측 클럭 파이프라인에 기인하여 메모리-측 I/O 회로 전에 클럭 펄스를 수신하는 것을 시작한다. 이것은 패킷의 임의의 나머지 부분이 제어기 I/O 회로에 의해 송신될 수 있고, 시스템 클럭 에지들이 도착되어 인커밍 데이터를 샘플링하기 전 (또는 후), 메모리 디바이스에 도착할 수 있으므로 분열된 패킷을 관리하는 것에 대한 실질적인 도전을 제공한다. 보다 일반적으로, 클럭 재-시작시 송신된 임의의 커맨드 또는 데이터 패킷의 비트들은, 클럭킹 에지들이 그들의 수신을 타이밍하는데 이용가능하기 전 메모리 디바이스에 도착한다면 드롭 (drop) 될 수도 있다. 일 실시형태에서, 이 복잡성은 (i) 클럭-정지에 이르는 인터벌에 대한 그리고 클럭 재-시작 다음 인터벌에 대한 무-동작 (NOP 또는 no-op) 커맨드의 송신, 및 (ii) 제어기-측 코어 클럭 (PCK1) 과 메모리-측 코어 클럭 (MCK1) 사이의 위상 관계가 클럭-정지 인터벌을 통해 유지되도록 보장하는 것에 의해 관리된다. 첫째, 클럭-정지 인터벌 직전 및 후의 no-op 송신은 포워딩된-클럭 파이프라인이 필링 (filling) 될 때 의미있는 커맨드 또는 데이터가 드롭되지 않도록 보증한다. 즉, no-op 커맨드와 함께 데이터가 송신되지 않고, 메모리 액세스 커맨드가 명시되지 않으므로, 클럭 재시작 시 초기에 송신된 비트 손실은 전혀 문제가 되지 않는다. 둘째, PCK1을 MCK1 위상 관계로 유지함으로써, 제어기-측 프레이밍/디-프레이밍 클럭 신호와 초기 교정에서 확립된 메모리-측 프레이밍/디-프레이밍 클럭 신호들 사이의 관계가 클럭-재시작 시에 유지된다. 즉, 의미있는 (즉, no-op 가 아닌) 커맨드 (CA) 및 데이터가 궁극적으로 커맨드 및 데이터 경로를 통해 전송될 때, 커맨드 및 데이터는 수신 디바이스에 의해 적절히 프레이밍되며, 카운터-파트 프레이밍/디-프레이밍 클럭들을 재정렬할 필요 없이 시스템 동작이 계속되는 것을 가능하게 한다. 게다가, 메모리 디바이스 내 개방-루프 클럭 분배 아키텍처 때문에, 메모리-측 송신 및 수신 클럭들의 위상이 클럭-정지를 통해 실질적으로 변하지 않은 상태로 남아, 클럭-정지 이전의 제어기-측 직렬화기/직렬분리회기 회로 내 위치에서의 위상 조정이 클럭-재시작 후에 유효한 상태로 남아서, 클럭 재시작시 즉 시 및 신뢰성 있는 데이터 및 커맨드 송신을 가능하게 한다.

[0106] 도 12a로 계속해서, 클럭 정지 후에, 코어 클럭은 계속 실행 (즉 진동, 토클) 되어 제어기 코어는 호스트-리퀘스트된 메모리 트랜잭션을 계속 수신 및 큐잉할 수 있고 전력-모드 로직은 트랜잭션 큐를 계속 모니터링하여 새로운 트랜잭션 리퀘스트의 펜딩 여부 및 시기를 결정할 수 있다. 새로운 트랜잭션 리퀘스트가 큐잉되었음을 검출하자마자, 전력 모드 로직은 클럭-시작-펜딩 상태로 천이하고 후행의 상승 PCK1 에지에서의 클럭-인에이블 신호들 ENPCK4 및 ENPCK8 을 높은 (어썬트한) 다. PCK4 및 PCK8에 대한 클럭-정지 로직은 각각의 리타이밍 지연 (또는 인에이블-레이턴시), t_{C4EL} 및 t_{C8EL} 후 리-타이밍 클럭-인에이블 신호 ENPCK4r 및 ENPCK8r 를 높임으로써 코어-도메인 클럭-인에이블 신호의 어썬트에 응답한다. 도시된 특정 예에서, 디어썬- 및 리-어썬 리-타이밍 지연이 매칭 (즉, $t_{C4DL} = t_{C4EL}$ 및 $t_{C8DL} = t_{C8EL}$) 된다. 이것은 본 예에서처럼, ENPCK4 및 ENPCK8 신호들의 디어썬 시간이 정수의 PCK1 클럭 사이클인 한 그렇다. ENPCK4 또는 ENPCK8 디어썬 시간이 전체 수의 PCK1 사이클이 아닌 경우, 클럭-정지 리-타이밍 지연 및 클럭-시작 타이밍 지연이 매칭되지 않을 것이지만, 클럭-정지 로직은 대응하는 클럭-인에이블 신호를 리-타이밍하여 정수 PCK1 클럭-정지 인터벌을 실시할 것이다.

[0107] 리-타이밍 지연이 일어난 후, PCK4 및 PCK8 클럭-정지 회로는 리-타이밍된 클럭-인에이블 신호들, ENPCK4r 및 ENPCK8r 을 각각 높이며, 따라서 시스템 클럭 (SCK, PCK4) 및 제어기 코어 클럭이 토클링을 시작하는 것을 가능하게 한다. 논의된 바처럼, 제어기 코어는 재-시작 인터벌을 일정 수의 no-op 커맨드로 패딩 (padding) 하여 시스템 클럭 펄스가, 새로 큐잉된 메모리 트랜잭션 리퀘스트에 대응하는 메모리 커맨드를 송신하기 전에 제어기 I/O 회로에 도달하도록 보장한다. 따라서 ("OP1"으로 도시된) 새로운 트랜잭션 리퀘스트가 큐잉된 후 (본 예에서는, 2-사이클 지연 후) 일정 수의 코어 클럭 사이클까지 커맨드 레인으로 전송되지 않아서, no-op 커맨드들이 클럭-재시작시 송신된다. 전력-모드 로직은 새로 큐잉된 트랜잭션 리퀘스트, OP1을 검출하자마자 재-시작 카운트다운을 시작하며, OP1이 카운트다운 종료 전 1 PCK1 사이클 디큐잉되는 것을 가능하게 하며 (따라서 no-op 를 패딩하기 위한 시간을 제공하며), 그에 의해 재시작-카운트다운의 종결시 직렬화기로서의 전송을 위한 시간에서 커맨드 레인으로 OP1을 로딩한다. 그 후 (인터벌 $t_{EN-CA(OP)}$ 후), OP1 커맨드는 직렬화되고 CA[0] 및 CA[1] 링크를 통해 송신되며, 클럭 재시작 후 디-프레이밍 클럭 에지와 코어-클럭 에지 사이의 교정된 정렬 (그리고 따라서 링크-링크 비트 가변성) 을 유지한다.

[0108] 도 12b 및 도 12c는 메모리 디바이스의 관점으로부터 클럭-정지 모드 진입 및 종료를 예시한다. 클럭 재-시작 다음의 메모리 쓰기 동작을 예시하는 도 12b를 먼저 참조하면, 최종 동작이 시간 (720) 에서 수신 및 트랜잭

선되고, 시간 (722) 에서 클럭 정지까지 카운트다운이 뒤따른다. 도시된 카운트다운 인터벌이 상술한 바처럼 제어기 내 전력-모드 로직에 의해 실시되고 도 12b의 메모리-측 타이밍 도 상에 오버레이되어 간단히 클럭-정지 이벤트가 최종 커맨드, OP0의 도착 후에 펜딩함을 나타냄에 유의한다. 도시된 바처럼, no-op 커맨드의 시퀀스는 OP0 를 뒤따르며, 따라서 메모리-측 데이터 직렬화/직병렬화 회로 및 코어 로직이, 클럭 정지가 일어나기 전에 OP0에 의해 명시된 동작을 완료하는 것을 가능하게 한다. 카운트다운 인터벌이 일어난 후에, 클럭이 도시된 것처럼 정지하며, 메모리 디바이스의 클럭-정지 저 전력 동작을 이룬다. 클럭-정지가 CA 링크를 통해 프레이밍 바운더리와 일치하는 것으로 도시되어 있지만, 이것은 시스템 동작 하에서 요구되지 않음에 유의한다. 대신에, 임의의 수의 시스템 클럭 펄스가 (클럭 링크를 통한 웨이브 파이프라인의 깊이, 개방-루프 클럭 분배 아키텍처의 온-메모리 클럭 레이턴시에 의존하는) 메모리 디바이스로 엔 라우팅 (en-routing) 될 수도 있고 따라서 프레이밍 바운더리 사이의 구현-특정 (및/또는, 다수의 메모리 디바이스들이 존재하고 메모리 제어기로부터 이질적인 로케이션에 배치된 경우, 디바이스 로케이션-특정) 시간에서 클럭-정지를 낼 수도 있다.

위에 논의된 바처럼, 이러한 결과는 일 실시형태에서 no-op의 송신을 통해 처리되어 클럭이 정지할 때 메모리-측 직병렬화기/직렬화기 회로를 통해 진행하지 않도록 (즉, 분열되지 않도록) 보장한다. (이 예에서 시스템 클럭 사이클 (52) 에서) 클럭이 재-시작할 때, 하나 이상의 no-op 커맨드들이 수신되며, 시작 시퀀스를 패딩하여 메모리 액세스 커맨드 및/또는 데이터가 도착하기 전에 클럭 에지들이 메모리-측 I/O 회로 내에 도착하게 된다. 도시된 예에서, 쓰기-커맨드 (WR) 및 수신하는 뱅크 어드레스 (Ba, 메모리 코어 내 다수 메모리 뱅크들 중 적어도 하나를 선택함) 및 칼럼 어드레스 (Ca, 선택된 뱅크의 센스 증폭기 내 상주하는 데이터의 페이지 내 다수의 칼럼들 중 하나를 선택함) 의 수신 전에 적어도 하나의 전체 no-op 커맨드가 수신된다. 쓰기 커맨드의 등록후 시간, t_{WRD} (쓰기-커맨드-데이터) 에, 쓰기 데이터 패킷 및 대응하는 데이터 마스크 패킷들이 인터벌, t_{BL} (버스트-길이 또는 버스트 시간) 동안에 수신된다. 대체로, 전체의 32바이트 및 32 대응하는 마스크 비트가 수신되고 메모리 코어로 전송되어 쓰기 커맨드와 관련하여 명시된 (칼럼 오프셋에서 시작하며) 뱅크 내에서 쓰여진다.

[0109] 도 12c는 도 12b와 본질적으로 같지만, 메모리 읽기 동작 맥락의 클럭-정지-모드 진입/종료를 예시한다. 이 경우에, 시간 t_{CL} (칼럼-어드레스-스트로브 (CAS) 레이턴시) 가 (즉, 뱅크 어드레스 Ba 및 칼럼 어드레스 Ca로부터의 메모리 코어로부터 데이터를 읽기 위한 커맨드인) 메모리 읽기의 등록과 (인터벌, t_{BL} 동안 각 데이터 링크 상의 송신된 4개 직렬화된 패킷들인) 읽기 데이터의 32 바이트의 출력 사이에 경과된다.

[0110] 도 13은 클럭-정지 인터벌이 분수 (fractional) 또는 비-정수 (non-integral number) 의 코어 클럭 사이클에 걸쳐 연장되는 것을 허용하는 다른 실시형태에 따른 클럭-정지 진입 및 종료를 예시한다. 논의된 바처럼, 클럭-정지 인터벌을 정수의 코어 클럭 사이클로 한정하는 것은 메모리-측 코어 클럭이 클럭-정지 후 재시작할 때, 메모리-측 코어-클럭 (MCK1) 과 제어기 코어 클럭 (PCK1) 사이의 위상 관계가 유지되도록 보장한다. 도 2a의 실시형태에서 적어도, MCK1이, MCK4로부터 MCK1을 생성하는데 사용된 4-분할 회로의 4개 가능한 상태들에 따라, PCK1에 대한 4개 위상 관계들 중 하나를 가짐을 상기하면, 정수-코어-클럭 제한이 해제되면, 메모리 코어 클럭은 클럭 재-시작 시 제어기 코어 클럭에 관하여 4개의 가능한 위상 관계들 중 하나를 가질 수도 있음이 뒤따른다. 메모리-측 직렬화기/직병렬화기 회로의 견지로부터, 이것은 클럭-정지 인터벌의 지식의 부재시, 4개-패킷-프레이밍/디프레이밍 클럭들중 어느 하나가 적용될 수 있으며 각각이 MCK1에 관한 4개 상이한 $n*2$ 비트-시간 오프셋 (즉, MCK1에 관하여 0, 2, 4 또는 6 비트 시간에 의한 오프셋) 중 하나로 정렬된다. 따라서, 일 실시형태에서, 메모리 디바이스는 4:1 멀티플렉서를 포함하여 클럭 재-시작시 4 패킷-프레이밍/디-프레이밍 클럭들 중 하나의 선택을 가능하게 한다. 게다가, 클럭 재-시작시 0-값의 무-동작 커맨드를 송신하는 대신에, 메모리 제어기는 결합된 no-op, 클럭-정렬 커맨드를 송신하며 이는 도 13에서 "NCK" 커맨드로서 도시되어 있다. 예로서, 각 NCK 커맨드는 (예를 들면, "11 00 00 00" 인) NCK 패킷 내 예정된 비트 포지션에서 단일 쌍의 '1' 을 포함할 수도 있다. 클럭-시작 시 4개 가능한 프레이밍/디프레이밍 클럭들의 각각으로 인커밍 커맨드 스트림을 프레이밍하고, 4개의 상이하게 프레이밍된 패킷들과 예상된 NCK 패킷 값들을 비교함으로써, 예상된 NCK를 낚는 프레이밍 클럭은 포워드 진행하는 메모리-측 프레이밍/디-프레이밍 클럭으로서 선택될 수도 있다.

[0111] **칩-칩 코어-클럭 위상 오프셋 조정**

[0112] 도 11a를 참조하여 논의된 바처럼, 예정된 파워-온/리셋 상태를 실시하는 회로 부재시, 메모리-측 코어 클럭, MCK1를 (즉, 메모리-측 I/O 클럭 (MCK4) 를 4로 분할함으로써) 생성하도록 제공된 예시적인 모듈로-4 카운터 (225, 616) 는 4개 가능한 상태들 (00, 01, 10, 11) 중 어느 하나로 파워업되고 따라서 MCK1과 제어기-측 코어

클럭 (PCK1) 사이의 4개 가능한 위상 관계들 중 하나를 임의적으로 확립할 수도 있다. 각 MCK4 사이클이 2-비트 시간 스패닝 (spanning) 하기 때문에, 4개 가능한 위상 MCK1-PCK1 위상 관계들이 (시스템 클럭 링크 또는 클럭 버퍼 지연을 통한 전파에 기인한 임의의 위상 오프셋을 카운팅하지 않는) 서로로부터 같은 2-비트-시간 위상 오프셋들에서 이격된다. 일 실시형태에서, MCK1-PCK1 위상 관계가 파워-업 (또는 리셋) 에 설정되고 그 후에 메모리 제어기의 드리프트-보상 직렬화기/직렬분화기 회로 내 비트-정렬 및 패킷-정렬 회로의 교정을 통한 수정 없이 처리된다. 이것은 어떤 상황에서는 증가된 레이턴시를 야기할 수도 있기 때문에, 레이턴시 어드밴티지는, 다른 실시형태에서 이루어질 수도 있는데, 여기서 가장 레이턴트한 데이터 링크가 제어기-측 코어 클럭 도메인에 관하여 위상이 어드밴싱 (도 3d에서 FCK1[0]의 타이밍을 2 비트 시간 어드밴싱하는 것 상상) 되는 상태로 초기 교정 동안 MCK4 분할기 (225) 가 조정되고 따라서 위스트-케이스 링크 타이밍 및 확대하면 최소 읽기 레이턴시를 감소시킨다.

[0113] 도 14a는, 도 11a를 참조하여 설명된 (즉, 증가 로직 (616) 및 2-비트 레지스터 (621) 에 의해 형성된 모듈로-4 카운터 (614) 와 함께, 2-비트 코어-클럭 조정 값 (CoreCkAdj[1:0]) 을 카운터 출력으로 가산하여 클럭-분할기 출력을 생성하는 모듈로-4 가산기 (751) 를 포함하는 클럭 분할기의 실시형태를 예시한다. 이 배열에 의해, 클럭-분할기 출력은 (모듈로-4 카운터 (614) 의 파워-업/리셋에서 결정된) 어느 임의의 초기 값으로부터 4개 가능한 출력 상태 (00, 01, 10, 11) 중 어느 것으로 시프트되어, MCK1 (즉, 분할기 출력의 MSB) 의 위상이 PCK1 에 관하여 2-비트-시간 증가 (또는 PCK1의 쿼드러췌어 단계들) 에 의해 조정되는 것을 가능하게 할 수도 있다. 도 14b는 이러한 결과를 예시하며, (밀침자 MCK1으로 도시된) 코어-클럭 조정 값의 4개 세팅들 중 각각에 대해 PCK1에 관하여 MCK1의 4개 예시적인 위상들을 나타낸다. 이해를 용이하게 하기 위해, 모듈로-4 카운터 (614) 는 초기에 상태 "00b" 에서 파워 업되어, (655 에서 도시된 PCK4의 초기 상승 에지에 관하여 지연의 일정 주기 후 일어나는) MCK4의 초기 상승 에지에서, 분할기 출력은 '00'에서 '01'로, 또는 '01'에서 '10'으로 또는 '10'에서 '11'로 또는 '11'에서 '00'으로 천이하며, 이는 위상-조정가능 MCK1 을 생성하는 코어-클럭 조정 값의 상태에 의존한다. 도시된 바처럼, 코어-클럭 조정 값에서 각 증가의 순 효과는 PCK1에 관하여 MCK1을 2 비트 시간 만큼 어드밴싱하는 것이다.

[0114] 도 14c는 코어-클럭 조정 값, CoreCkAdj[1:0] 의 4개 설정들의 각각에 대해 PCK1 에지에 관하여 제어기-측 패킷-프레이밍 바운더리의 예시적인 정렬을 예시한다. 도시된 예에서, 읽기 데이터가 링크 DQ[7] 상의 최소 레이턴시와 링크 DQ[0] 상의 최대 레이턴시로 리턴되고, 게다가 DQ[0] 을 통해 도착하는 패킷들이 PCK1 샘플링 에지 바로 다음에 프레이밍되는 한편, DQ[7] 을 통해 도착하는 패킷들이 PCK1 샘플링 에지 바로 전에 프레이밍된다고 가정한다. 도 3c-도 3e를 참조하여 위에서 설명된 패킷 정렬 기법을 이용하는 실시형태에서, 시스템 읽기-레이턴시가 위스트-케이스 최소로 설정되고, 따라서 (즉 CoreCkAdj[1:0]='00'인) MCK1₀₀ 경우에 대해 도시된 링크 DQ[0] 의 N+1 레이턴시 (N+1 PCK1 사이클) 로 설정된다. 하지만, 메모리-측 코어 클럭의 위상을 2 비트-시간 만큼 어드밴싱함으로써, (즉, MCK₀₁로 도시된 바처럼), 모든 인커밍 패킷들이 PCK1 샘플링 에지에 관하여 2 비트-시간 먼저 도착하고, 따라서 에지 N+1 대신 PCK1의 N번째 샘플링 에지에 응답하여 샘플링되어, 시스템 읽기-레이턴시를 1 PCK1 클럭 사이클 만큼 감소시킨다 (즉, N PCK1 사이클의 시스템 읽기 레이턴시를 이룬다). MCK1이 또 하나의 2-비트 시간 인터벌 (MCK₁₀) 에 의해 어드밴싱될 때 유사한 결과가 얻어지며, 훨씬 더 많은 제어기-측 마진 (및 따라서 가능하게는 더 많은 드리프트 공차) 을 제공한다. 하지만, MCK1이 또 하나의 2-비트-시간 인터벌에 의해 어드밴싱될 때, (MCK₁₁로 도시), PCK1 직렬화 바운더리가 미싱되며 (즉 코어로부터의 데이터가 그러한 어드밴싱된 시간에서 직렬화에 대해 준비되지 않으며), 따라서 1-사이클 지연된 MCK1 에지에 관하여 데이터 직렬화 및 따라서 MCK1₀₀ 경우에서 보다 메모리 제어기에서의 훨씬 더 레이턴트한 도착을 야기한다.

[0115] 도 14c가 나타내는 바처럼, 감소된 시스템 레이턴시는 일부이지만 전부는 아닌 코어-클럭 조정 세팅으로 이루어질 수도 있다. 따라서, 일 실시형태에서, 각 코어-클럭 조정 세팅이 예를 들면 위에서 설명한 비트-정렬 및 패킷-정렬 동작을 실행함으로써 차례로 테스트되어, 각 세팅으로 이루어질 수 있는 최소 시스템 레이턴시를 결정한다. 하나 보다 많은 세팅이 같은 최소 시스템 레이턴시를 낳는 경우에, 같은 최소 시스템 레이턴시를 낳는 것들의 중앙 세팅 (median setting) 또는 다른 통계적 중심이 선택되어 어느 한쪽 방향으로 최대 드리프트 공차 (tolerance) 를 제공한다. 도 14c의 예에서, 같은 최소값을 낳는 2개 코어-클럭 조정 세팅들이 존재하기 때문에, 추가 정보가 모아져서 2개 세팅들 중 어느 것이 가장 큰 드리프트 공차 제공하는지를 결정 (하여 그 세팅을 교정 결과로 선택) 할 수도 있거나, 예정된 선택이 이루어질 수도 있다 (예를 들면, 항상 최고-값의 코어-클럭 조정, 또는 마지막 테스트된 조정을 선택하여 최소 레이턴시를 낳는다).

[0116] 도 14a를 다시 참조하면, 일 실시형태에서, 코어-클럭 조정 세팅 (CoreCkAdj[1:0]) 이 사이드-밴드 링크를 통해 메모리 디바이스로 통신되어, 세팅이 제어기-측 직병렬화기 교정의 종결시에 그리고 제어기-측 직렬화기 교정 전에 리바이싱 (revising) 되는 것을 가능하게 한다. 다르게는, 완전한 교정이 (직병렬화기 및 직렬화기에서) 수행되며 코어-클럭 조정이 뒤따르며, 필요하면 반복한다.

[0117] 메모리-측 코어 클럭 조정이 메모리-측 코어 클럭 및 제어기-측 코어 클럭의 위상들을 서로에 대해 시프트함을 고려하면, 같은 상대 위상 시프트가 메모리-측 코어 클럭 보다 제어기-측 코어 클럭의 위상을 시프트함으로써 다르게 이루어질 수도 있다는 결과가 된다. 일 실시형태에서, 예를 들면, 도 2a의 8-분할-회로 (163) 가 변경되어 PCK1의 위상이 8개 분할기 상태들 중 어느 것으로 어드밴스되는 것을 가능하게 하고 따라서 PCK1이 MCK1에 대해 시프트되는 것을 가능하게 한다. 다른 실시형태에서, 도 2a의 2-분할-회로 (165) 가 변경되어 PCK4의 위상이 하프 사이클 만큼 어드밴스 (사실상, 반전) 되는 것을 가능하게 한다. 게다가, 클럭 위상 시프팅 대신에, 조정 메카니즘이 제어기-측 PCK4 클럭 정지 로직 내에서 PCK4 펄스들의 (1, 2 또는 3) 의 어떤 수를 억압하여 MCK1 과 PCK1 사이의 초기 위상 관계를 설정함으로써 전술한 레이턴시 어드밴티지를 이룬다. 또 다른 실시형태에서, 위상-시프팅 회로 (예를 들면, 인터폴레이터) 가 PLL의 출력에 제공되어 포워드된 클럭이 (실행가능하고) 요망되는 레졸루션으로 위상-단계형이 되는 것을 가능하게 하여 전체 교정된 드리프트 공차로 감소된 시스템 레이턴시를 확립 (예를 들면 위상 세팅을 통해 스텝핑하거나 서치하여 최소 레이턴시 윈도우의 바운더리를 찾아내고 바운더리들 사이에 센터링된 최종 위상을 확립) 할 수도 있다.

[0118] **글리치리스 위상 점핑**

[0119] 일 실시형태에서, 위에서 설명한 클럭-정지 로직이 주기적 타이밍 교정 동작의 시작과 종료시에 이용되어 그렇지 않으면 위상 점핑 동안 제어기-측 수신 및 송신 클럭들에서 일어날 수도 있는 글리치들을 억압 (또는 마스킹) 한다. 즉, 도 15a에 도시된 바처럼, 소정 링크에 대한 데이터 샘플링 클럭의 위상, RCK8[i] 가 갑자기 라이브 데이터 (RCK8[i]_{LIVE}) 를 수신하는데 사용된 아이-센터링된 위상으로부터 타이밍 드리프트 (RCK8[i]_{PTC}) 를 검출하는데 사용된 바운더리 위상으로 갑자기 천이 (즉, CalClkSel 어썬션에 응답하여 위상-점핑) 할 때, 순 클럭 파형, RCK8[i]_{NET}에 의해 나타난 바처럼, 직병렬화 프레이밍 로직을 글리칭할 만큼 짧은 런트 클럭 펄스 (775) 가 클럭 라인 상에 나타날 수도 있다. 보다 상세하게는, 런트 펄스 (775) 가 (즉, 가능하게는 프레이밍 클럭을 생성하는데 사용된 카운터 회로에 의해 카운팅된, 가능하게는 그렇지 않은) 패킷-프레이밍 회로 내 비결정적 액션 (indeterminate action) 을 만들 정도로 지속 기간이 짧고 따라서 라이브 데이터 전송으로 리턴시 (즉, 주기적 타이밍 교정 종료시) 패킷 프레이밍 에러를 낳을 수도 있다. 일반적으로, 그러한 클럭 글리치 및 결과적인 로직 글리치들은 PTC 위상 점핑 동작 동안 제어기-측 코어를 억압함으로써 회피될 수도 있다.

[0120] 도 15b는 사전-PTC클럭-정지 동작 및 얻어지는 결과적인 논-글리칭 클럭 파형 (RCK8[i]_{NET}) 을 예시하는 타이밍도이다. 일반적으로, 단일-코어-클럭-사이클 클럭-정지 인터벌이 라이브 동작 (즉, 실행-시간 (run-time) 읽기 및 쓰기 데이터 전송) 및 PTC 동작 사이 각각의 천이에 삽입된다. 클럭-정지 인터벌은 라이브-모드와 PCT-모드 클럭 위상 선택들 사이의 천이가 수신 클럭이 디스에이블링되는 동안 이루어지는 것을 가능하게 하며, 클럭-정지 인터벌 동안 모든 다른 수신 클럭 펄스와 함께 임의의 잠재적인 런트 펄스를 억압하고 따라서 직병렬화 프레이밍 로직에 투명한 위상 점프를 만든다. 클럭 재시작시, 수신 클럭 펄스가 새로운 (PTC) 클럭 위상에도 불구하고, 프레이밍 로직에 의해 에러 없이 카운팅된다. 이 동작은 서수로 넘버링된 펄스에 의해 도 15b에 나타나 있으며, 펄스, 0, 1, 및 2는 라이브 모드 수신 클럭 위상 (RCK[i]_{LIVE}) 에 응답하여 카운팅되고 펄스 3, 4, 5, 6, 7 등은 PTC-모드 수신 클럭 위상 (RCK8[i]_{PTC}) 에 응답하여 카운팅된다. 라이브-모드 클럭 및 PTC-모드 클럭에서 억압된 펄스들은 회색 음영 (778) 으로 도시되어 있다. PTC 모드 클럭이 2개 클럭 위상들 사이의 오프셋을 설명하기 위해 라이브-모드 클럭과 동시에 토글링하는 것으로 도시되어 있음에 유의한다. 실제로, 오직 순 클럭 결과 RCK8[i]_{NET} 가 RCK8[i] 클럭 라인 상에 나타난다. 억압된 런트 펄스 (780) 를 포함하는, 순 억압된 펄스가 또한, 781에서 회색으로 도시되어 있다. 마지막으로, PTC-모드로부터 라이브-모드로의 리턴 위상 점프에서의 런트 펄스를 회피하기 위하여, 다른 단일-PCK1-사이클 클럭 정지 동작이 782에 도시된 것처럼 수행된다.

[0121] 도 16a-도 16f는 클럭 정지 없이 글리치리스 위상 점핑을 가능하게 하는 주기적-타이밍 교정을 수행하는 다른 방식에 관한 것이다. 일반적으로, 다른 접근법은 PTC-모드로 또는 PTC-모드로부터의 어느 천이시 하나 이상의 하프-비트-시간 (또는 하프-유닛-인터벌 (UI)) 위상 점프에 유리한 임의적 위상 점핑을 제거하는 완전히 상

이한 클럭-드리프트 검출을 수반한다. 아래에 논의된 바처럼, 하프-UI 만큼 전의 클럭 위상을 리딩하는 새로운 클럭 위상으로의 천이로 각 하프-UI 위상 점프를 제한함으로써, 모든 클럭 펄스들은 적어도 지속시간이 하프-UI가 되고 따라서 RCK8[i] 클럭 펄스의 펄스 폭 보다 더 짧지 않도록 보장된다. 마지막으로, 일 실시형태에서, PTC 모드로부터의 종료는 연속적인 코어-클럭 사이클에서 수행되는 3개 하프-UI 위상 점프들의 시퀀스를 수반하고 따라서 PTC 모드로 그리고 PTC 모드로부터의 천이에서 2개 유닛 인터벌 ($4 \times 0.5UI = 2UI$)의 전체 제어기-측 위상 어드밴스를 이룬다. 따라서, 메모리-측 프레이밍 및 디-프레이밍 로직에 관하여 패킷-프레이밍 동기화를 유지하기 위하여, 메모리 제어기는 PTC 모드로부터의 종료시에 프레이밍 클럭을 2 유닛 인터벌 (2-비트 시간) 만큼 지연시킨다.

[0122] 도 16a는 아이-오프닝 자체 동안 보다 데이터 아이들 사이 천이에서 캡처링된 시그널링 파형의 샘플들에 기초한 주기적 타이밍 교정을 예시한다. 일반적으로, 데이터 값들의 인커밍 시퀀스는 데이터-아이 중간점에서 수신 클럭 신호 (RCK) 에 응답하여 샘플링되어 전달된 데이터 값들 ($d_i, d_{i+1}, d_{i+2}, d_{i+3}, \dots$) 에 대응하는 데이터 샘플들의 시퀀스 ($s_i, s_{i+1}, s_{i+2}, s_{i+3}, \dots$) 을 낳을 수도 있다. 게다가, 시그널링 파형을 오버샘플링하며, 추가적으로 데이터 아이들 또는 "에지 샘플들" ($e_i, e_{i+1}, e_{i+2}, e_{i+3}, \dots$) 사이의 천이 (에지들) 에서 샘플들을 캡처링하는 것에 의해, 에지 샘플이 선행 또는 후행 데이터 샘플 중 어느 하나를 매칭하는 것에 실패할 때마다, 위상 정보가 획득될 수도 있다. 보다 상세하게는, 유닛 인터벌이, 정의에 의하면, 시그널링 파형에서 연속적인 에지들 사이의 시간이기 때문에, 파형이 유닛 인터벌 마다 2번, 데이터 샘플을 생성하기 위해 샘플링 클럭 신호에 대응하여 한번 그리고 에지 샘플을 생성하기 위해 샘플링 클럭 (또는 에지 클럭) 의 하프-UI-시프트된 버전에 대응하여 다시 한번 샘플링되면, 에지 샘플 및 그의 선행 또는 후행 데이터 샘플 사이의 불일치는, 사실상, 천이가 시그널링 파형 ('1'에서 '0'으로 또는 그 역 또한 같음) 에서 일어나고 이상적 에지 샘플링 포인트에 관하여, 각각 에지 샘플이 너무 후기 (late) 또는 너무 조기 (early) 에 캡처링된다는 표지이다. 이러한 결과는 도 16a에서, e_i 가 s_i 와 같지 않으면, 샘플은 s_i 에서 s_{i+1} 으로의 천이 후에 캡처링되고 따라서 후기; e_i 가 s_{i+1} 과 같지 않으면, 샘플은 s_i 에서 s_{i+1} 로의 천이 전에 캡처링되고 따라서 조기인 2개의 조기/후기 부등 (inequality) 에 의해 도시된다. 따라서, 위상 에러 정보는 소정 시간 인터벌을 통해 (또는 예정된 수의 에지 샘플들을 통해) 획득된 조기/후기 표지들 중 다수가 에지 클럭 (그리고 따라서 샘플링 클럭) 이 (조기/후기 표지들이 밸런스를 이루는) 이상적인 샘플링 포인트에 관하여 조기 또는 후기를 나타내는지 결정하고 그에 따라 에지 및 샘플링 클럭들의 위상을 조정함으로써 생성될 수도 있다.

[0123] 일 실시형태에서, 위에서-설명한 조기/후기 결정이 오버샘플링 없이 그리고 대신에 알려진 데이터 패턴을 송신하고 하프-UI-시프트된 수신 클럭 (즉, 에지 클럭) 을 사용하여 대응하는 에지 샘플들을 생성함으로써 주기적-타이밍-교정 (PTC) 동작 동안 이루어진다. 도 16b는 위상-에러 검출기 (801) 의 실시형태를 예시하는데, 이는 배타적 NOR (XNOR) 게이트 (803) 에서 한 세트의 에지 샘플들 (e_0, e_1, \dots, e_{n-1}) 을 알려진 데이터 샘플들 ($d_0, d_1, \dots, d_{n-1}, d_n$) 과 비교하며, 결과적인 샘플-조기/샘플-후기 (sE/sL) 를 보우팅 로직 (804) 에 공급한다. 일 실시형태에서, 보우팅 로직은 조합 로직 회로이고 이는 조기 표지 또는 후기 표지가 우세한지 (다수 보우트를 구성하는지) 에 따라 위상 증가/ 감소 신호 ("Inc/Dec") 를 생성하고 증가/감소 신호를 출력하여 수신 클럭 신호의 정렬 카운트를 업데이트한다.

[0124] 도 16c는 제어기-측 드리프트-보상 직병렬화기의 주기적 타이밍 교정 동안 위상 업데이트를 이루기 위해 수행되는 동작들의 예시적인 시퀀스를 예시한다. 821에서 시작하여, 수신 클럭이 하프 UI (0.5 UI) 만큼 위상-어드밴싱된다. 그 후에, 예정된 (또는 예측가능한) 테스트 데이터 패턴의 메모리-측 송신이 개시된다. 일 구현에서, 예를 들면, 사이드-링크 커맨드가 메모리 디바이스로 이슈되어 테스트 패턴 송신을 개시한다. 다르게는, 메모리 디바이스는 상술한 루프백 모드에 배치될 수도 있고, 테스트 패턴이 메모리 제어기에서 메모리 디바이스로 송신되고, 그 다음 루프백 동작에서 메모리 디바이스로부터 메모리 제어기로 재-송신된다. 어느 경우에도, 823에서, 메모리 제어기는 인커밍 테스트 패턴을 하프-UI 시프트된 수신 클럭으로 샘플링하여 825에서 에지 샘플들의 시퀀스를 생성한다. 에지 샘플들은 결정 블록 (827) 에서 평가되어 클럭-조기 표지 ($e_i < d_i$) 또는 클럭-후기 표지 ($e_i > d_{i+1}$) 가 우세한지를 결정한다. 클럭-조기 표지가 다수를 구성하면, 수신 클럭은 원하는 샘플링 포인트에 관하여 후기로 여겨지고, 대응하는 정렬 카운트가 828에서 감소되어 클럭 위상을 어드밴싱시킨다. 반대로, 클럭-후기 표지가 다수를 구성하면, 수신 클럭은 원하는 샘플링 포인트에 관하여 조기로 여겨지고, 그리고 대응하는 정렬 카운트가 829에서 증가되어 클럭 위상을 지체시킨다. 그 후에, 수신 클럭이 831에서 1.5UI 만큼 어드밴싱되어 사전-교정 위상을 복원한다. 일 실시형태에서, 이 위상 어드

밴스는 3개 0.5 UI 위상 어드밴스 동작들의 시퀀스에 의해 이루어져 도 16d 및 도 16e를 참조하여 아래에서 논의된 것처럼 사전-교정 위상을 복원한다. 마지막으로, 833에서, 2 비트-시간 지연이 프레이밍 클럭 생성기에 도입되어 수신 클럭의 순 2UI 위상 어드밴스로부터 발생하는 추가적인 펄스들을 보상한다. 이 동작은 도 16f를 참조하여 더 자세히 설명된다.

[0125] 도 16d는 위상-어드밴스 신호 ("Adv0.5UI)에 응답하여 글리치리스 0.5UI 위상 어드밴스를 제공하는 클럭-위상-시프팅 회로의 실시형태를 예시한다. 도시된 바처럼, 회로는 링-커플링된 쌍의 차동 에지-트리거드 플립-플롭 (841, 843) 을 포함하고 이는 비트-레이트 수신 클럭 (RCK8[i])의 상승 및 하강 (포지티브 및 네가티브) 에지들에 의해 각각 클럭킹된다. 포지티브-에지-트리거드 플립-플롭 (841)의 반전-비반전 출력들이 네가티브-에지-트리거드 플립-플롭 (843)의 대응하는 반전 및 비반전 입력들에 커플링되는 한편, 네가티브-에지-트리거드 플립-플롭 (843)의 반전 및 비반전 출력들이 포지티브-에지-트리거드 플립-플롭 (841)의 비반전 및 반전 입력들에 크로스-커플링된다. 이 배열에 의해, 비트-레이트 클럭 (RCK8[i])의 각 상승 에지에 응답하여 포지티브-에지-트리거드 플립-플롭 (841) 천이의 포지티브 (비-반전) 및 네가티브 (반전) 출력들은 2 비트-시간 마다 한번씩 사이클을 이루는 한편, 비트-레이트 클럭의 각 하강 에지에 응답하여 네가티브-에지-트리거드 플립-플롭 (843) 천이의 포지티브 및 네가티브 출력들은 2비트 시간 마다 한번씩 사이클을 이루지만, 포지티브-에지-트리거드 플립-플롭 (843)의 출력에 관하여 쿼드러췌어 관계 (하프-UI-오프셋) 를 이룬다. 따라서, 도 16e에 도시된 바처럼, 4 하프-비트-레이트 클럭 신호들, iCK_P 및 iCK_N (포지티브 및 네가티브 "동-위상 (in-phase)" 클럭) 및 qCK_P 및 qCK_N (포지티브 및 네가티브 "쿼드러췌어" 클럭) 이 생성되며, (즉 하프-비트-레이트 클럭 사이클의 1 사이클인) 2UI 인터벌 내 하프-UI 위상-오프셋에 의해 위상 분배된다. 도 16d에 도시된 바처럼, 4 클럭 신호들이 멀티플렉서 (847)의 입력 포트에 공급되고 2-비트 (모듈로-4) 카운터 (845)의 출력에 응답하여 출력에 대하여 선택된다. 일 실시형태에서, 카운터 (845)는 그레이-코드 카운터 (예를 들면, 카운트 시퀀스=00,01,11,10,00,...) 로 구현되어 출력 글리칭을 회피하고 위상-어드밴스 신호 (Adv0.5UI)에 응답하여 어드밴싱되어 시퀀스에서 다른 클럭 신호들을 선택함으로써 1 하프-비트-레이트 클럭으로부터 다음으로 위상 점프를 이룬다. 이 동작에 의해, 그리고 각 위상 점프를 1 하프-비트-레이트 클럭으로부터 하프-비트-레이트 클럭의 하프-UI-어드밴싱된 경우로 (즉, 화살표 850에 의해 도시된 것처럼, iCK_P로부터 qCK_N으로, qCK_N으로부터 iCK_N으로, iCK_N으로부터 qCK_P으로, 그리고 마지막으로 qCK_P 으로부터 다시 iCK_P으로)의 점프를 제한함으로써, 어드밴스 신호가 언제 증가되더라도, 위스트-케이스 (최단-지속시간) 런트 펄스가 851 및 853에서 도시된 것처럼 적어도 0.5UI 지속기간이 되도록 보장된다. 따라서, 모든 로직 회로들이 0.5UI-이격된 클럭 에지들에 결정적 응답이 가능 (즉, 데이터-레이트 주파수 클럭에 의해 클럭킹되는 것이 가능) 하도록 보장함으로써, 디터미넨트 (determinant), 글리치리스 회로 동작이 보장된다.

[0126] 계속 도 16d를 참조하면, (즉, 주기적 타이밍 교정이 완료된 후 라이브 동작을 복원하는데 일반적으로 바람직한) 소정 하프-비트-레이트 클럭으로부터 하프-UI-지연 클럭으로의 리턴 0.5UI 위상 점프는 같은 글리치리스 클럭 결과를 낳지 않음을 알 수 있다. 즉, 비결정적 지속기간의 런트 펄스들이 위상 점프가 언제 개시되는지에 따라 생성될 수 있다. 일 실시형태에서, 그러한 런트 펄스들이 연속하는 코어-클럭 사이클에서 실행된--전체 1.5UI- 3 추가 하프-UI 위상 점프의 시퀀스에 의해 원래 (사전-PTC) 클럭 위상으로 리턴을 이룸으로써 회피된다. 마지막으로, 0.5UI 위상 어드밴스 (하나는 PTC 동안 에지 클럭을 제공하며, 3개는 데이터 샘플링 클럭 위상을 복원한다)의 순 효과가 2 유닛 인터벌 만큼 결과적인 클럭의 위상을 어드밴싱하는 것이기 때문에, 비트 프레이밍 클럭을 생성하는데 사용되는 카운터 회로는 2 유닛 인터벌만큼 지연되어 메모리-측 패킷 프레이밍에 관하여 동기화를 유지한다. 이 효과는 도 16f에 개념적으로 도시되어 있는데, 4개 하프-UI 위상 점프들의 시퀀스가 카운터파트 메모리-측 클럭 (MCK4)에 관하여 제어기-측 클럭 (RCK4) 내 2 추가적인 비트-타이밍 에지들을 발생시킴을 도시한다. 일 실시형태에서, 프레이밍 클럭 지연 회로는 도 3a의 직병렬화기 내에 제공되어, PTC 모드로부터의 종료시 프레이밍 클럭 신호, RCK1 및 FCK1을 생성하는데 사용된 모듈로-8 카운터로부터 2를 감산하며, 따라서 적절한 패킷-프레이밍 바운더리를 복원한다.

[0127] 도 16d로 되돌아가면, 위상 점핑 회로의 하나의 결과는 하프-비트-레이트 수신 클럭, RCK4[i] 를 낳는 것임을 알 수 있다. 일 실시형태에서, 이 결과는 하프-비트-레이트 클럭의 상승 및 하강 에지들 양쪽 모두에 응답하여 클럭-인/클럭-아웃 데이터로 제어기-측 직렬화기/직병렬화기 회로를 리바이싱함으로써 수용된다. 일 실시형태에서, 예를 들면, 도 4a 및 도 4b의 하프-비트-레이트 직렬화기/직병렬화기 회로는 메모리 제어기 내에서 구현되며, MCK1 대신에 정렬-카운트-제어된 패킷-프레이밍 클럭을 적용한다.

[0128] 비록 도 16a-도 16f는 제어기-측 수신 클럭 타이밍을 참조하여 설명되었지만, 카운터파트 0.5UI 위상 점프 동작들 (및 송신-클럭 생성 회로)는 송신 클럭 위상의 주기적 타이밍 교정을 이루도록 실행될 수도 있다. 예를

들면, 0.5UI 만큼 송신 클럭 위상을 위상-어드밴싱하고, 그 다음 결과적인 메모리-측 캡처된 에지 샘플들을 (예를 들면 루프백을 통해) 수신함으로써, 같은 증가/감소 결정이 이루어질 수도 있으며, 이 경우, 에지 샘플이 초기-메모리-측 샘플링 인스턴트를 나타내면 송신 클럭 위상을 어드밴싱 (즉, 데이터 위상을 어드밴싱하고 효과적으로 메모리-측 샘플링 인스턴트를 지체) 시키며 에지 샘플들이 후기-메모리-측 샘플링 인스턴트를 나타내면 송신 클럭 위상을 감소시킨다. 유사하게, PTC로부터 종료시, 송신 클럭의 위상이 0.5UI 위상 점프의 시퀀스에서 1.5UI 만큼 어드밴싱되어 (이제 에지 드리프트에 따라 조정된) 사전-교전 송신 위상을 복원시킬 수도 있다.

마지막으로, 제어기-측 디프레이밍 클럭은 2의 카운트 만큼 지연되어 (메모리-측 타이밍에 관하여) 2개 추가 타이밍 에지들을 위해 수정할 수도 있는데 이는 4개 0.5UI 위상 점프로부터 발생된다.

[0129] **저-전력 클럭-정지 모드를 갖는 메모리 시스템의 시스템 어플리케이션**

[0130] 저-전력 클럭-정지 모드를 갖는 메모리 시스템들은 메모리 제어기 및 단일 메모리 디바이스의 맥락에서 지금까지 설명되었다. 그러한 타이트하게-커플링된 제어기/메모리 시스템들은 일정 수의 이동 어플리케이션들에서 사용될 수도 있지만, 다르게 단일 메모리 제어기 집적-회로 (제어기 IC) 는 다양한 아키텍처에서 배치된 다중 메모리 디바이스들 (메모리 IC) 들을 제어할 수도 있다. 또한 다중 메모리 제어기 채널은 단일 IC에서 구현될 수도 있으며, 각각은 분리된 그룹의 하나 이상의 메모리 IC 들을 제어하며 따라서 단일 클럭 회로가 다중 제어기-측 I/O 회로 및 개방-루프 메모리-측 클럭 분배 회로를 위한 클럭 신호들을 생성하는 것을 허용한다.

[0131] 도 17a는 단일 제어기 IC (751) 및 다중 메모리 IC (755_0 - 755_{n-1}) 을 갖는 일지정지-가능-클럭 메모리 시스템 (750) 의 실시형태를 예시한다. 도시된 실시형태에서, 메모리 디바이스 IC (집합적으로, 755) 는 메모리 모듈 (753) (일반적으로 백플레인 또는 마더보드에 분리성 접속을 위한 에지 커넥터를 가지며, 따라서 추가의 메모리 모듈이 삽입될 때 메모리 용량 확장을 허용하는 회로 보드) 상에 배치되며, 도 2a에 도시된 것처럼 개별적으로 I/O 인터페이스 및 개방-루프 클럭 분배 배열을 포함한다. 그 경우에, (752) 로 도시된 시그널링-링크 그룹들의 각각은 메모리 제어기와 메모리 디바이스들의 각각의 메모리 디바이스 사이에 포인트-포인트 접속을 포함할 수도 있고 전용 클럭, CA 및 데이터 링크(및 필요한 경우 데이터-마스킹) 를 포함할 수도 있다. 다르게는, 일부 또는 전부의 시그널링 링크들이 메모리 모듈의 모든 메모리 디바이스들에 분배될 수도 있다 (예를 들면, 클럭 링크는 멀티-드롭 (multi-drop) 방식으로 모든 메모리 디바이스들에 커플링되고/거나 커맨드 링크(들) 은 멀티-드롭 방식으로 모든 메모리 디바이스들에 커플링된다). 부가적으로, 각 시그널링 링크는 다중 메모리 디바이스에 커플링되며 (예를 들면, 데이터 링크 [0 내지 N-1] 에서 다중 메모리 모듈 (753) 의 각각 상에 제 1 메모리 IC에 커플링되며, 데이터 링크 [N+1 내지 2N] 은 메모리 모듈들의 각각 상의 제 2 메모리 IC에 커플링되는 것 등과 같이, 데이터 링크는 일정 수의 메모리 모듈을 가로질러 메모리 디바이스들의 슬라이스에 커플링된다) 따라서, 멀티-드롭 데이터 및/또는 커맨드 경로를 확립할 수도 있다. 후자의 경우에, 추가의 타이밍 보상 값들이 소정 메모리 액세스 트랜잭션에 대해 선택된 메모리 모듈 (또는 같은 모듈 상의 2개 이상의 그러한 그룹들로부터 선택된 메모리 디바이스들의 그룹) 에 따라 적용될 수도 있다. 그 경우에, 패킷, 비트 및 위상 조정 값들이, 소정 메모리 액세스 트랜잭션에 대해 타겟팅된 메모리 디바이스들의 그룹에 따라 동적으로 스위칭되며, 정렬 레지스터들의 분리된 세트가 각 그룹에 대해 유지된다.

[0132] 도 17b는 다른 메모리 시스템 실시형태를 예시하며, 이 경우에 도 2a에 도시된 메모리-측 I/O 인터페이스에 대응하는 인터페이스 (777) 를 구현하는 모듈-장착된 버퍼 IC (775) 를 갖는다. 이 배열에 의해, 클럭-정지된 저-전력 모드를 갖는 고속 시그널링 시스템이 메모리 제어기 (771) 와 버퍼 IC (775) 사이에서 구현될 수도 있으며, 더 많은 종래의 인터페이스 (729) 가 메모리 모듈 (773) 상의 버퍼 IC (775) 쪽에 배치된 메모리 디바이스 781_0 - 781_{N-1} , 782_0 - 782_{N-1} 와 버퍼 IC 사이에 구현된다. 일 실시형태에서, 예를 들면, 커맨드/어드레스 값들은 뱅크, 로우 및 칼럼 어드레스 뿐만 아니라, 개별 메모리 디바이스 781, 782, (또는 메모리 디바이스들의 그룹들) 의 어드레스를 포함하며, 이로 버퍼 IC (775) 가 커맨드를 포워딩하게 된다. 버퍼 IC는 추가적으로 데이터 입력/출력 버퍼를 포함하여 어드레스-선택된 메모리 디바이스 (또는 메모리 디바이스 그룹) 으로 최후의 분배를 위한 인커밍 쓰기 데이터, 및 읽기 데이터를 큐잉하여 메모리 제어기로 포워딩되게 할 수도 있다. 예로서, 일 실시형태에서, 버퍼 IC-메모리 디바이스 인터페이스는 링크 무결성 (integrity) 을 유지하는데 메모리 PLL/DLL을 필요로 하지 않는 상대적으로 느린 시그널링 인터페이스들이거나, 표준 스트로브-기반 시그널링을 사용하여 구현될 수도 있다.

[0133] **터어드 전력 모드**

[0134] 일 실시형태에서, 상술한 메조크로노스 저-전력 시그널링 시스템이 액티브 동작 모드 (액티브 모드) 및 상술한

클럭-정지 저-전력 모드 이외에 2개 다른 전력 모드: 신호 송신기 및 수신기 회로 내 바이어싱 전류 소스가 셧다운되는 파워다운 모드, 및 제어기-측 PLL (도 2a의 엘리먼트 (161)) 이 제어기 코어 내 로직 회로와 함께 디스에이블링될 수도 있는 딥 파워다운 모드를 지원한다. 모든 전력 모드들 사이의 천이는 제어기 코어로부터의 커맨드 트래픽에 응답하여 상술한 전력-모드 로직에 의해 관리될 수도 있다. (본원에서 전력 상태로도 지칭되는) 전력 모드는 전력 소비를 감소시키기 위해 종료 레이턴시를 증가시키는 것을 트레이드하는데 사용될 수도 있다. 다음 표 (표 1) 은 일 구현에서 메모리-제어기 전력 상태 성능을 요약하며, 액티브 모드 (P4) 와 3개 저-전력 모드를 나타낸다:

표 1

전력 모드	종료 레이턴시 (PCK1 사이클)	제어기 인터페이스 전력 vs. 피크 인터페이스 DQ 대역폭		
		2.7GB/s	3.2GB/s	4.3GB/s
P4 (액티브)	--	80.8	91.6	114.7
P3 (클럭-정지; 유틸)	0	15.0	15.7	17.4
P2 (파워다운)	10	4.6	5.2	6.5
P1 (딥 파워다운)	130	0.4	0.4	0.4

[0135]

[0136]

도시된 바처럼, P4 (액티브) 모드에서, 4.3GB/s (초당 기가바이트) DQ 대역폭이 114.7mW (3.3mW/Gb/s) 로 제공된다. P3 모드에서, 클럭 분배는 상술한 바처럼 일시 정지되고 DQ 출력 드라이버, 입력 증폭기, 및 데이터 샘플러가 추가적으로 디스에이블링될 수도 있다. P2 모드에서, 모든 송수신기들이 디스에이블링되며 (클럭 송신기 및 수신기 회로 포함), 오직 클럭 체배기만이 액티브가 된다. P1 모드에서, 오직 누설 전력이 소비된다. 각 전력 상태를 위한 엔트리 레이턴시는 (0 병렬 (PCK1) 클럭 사이클들의 최소값으로) 프로그래밍 가능할 수도 있으며, 상태 천이의 강화된 흐름 제어를 제공한다. 고속 전력-상태 천이 시간들은 피크 대역폭이 요구되지 않을 때 버스트 전송의 효율적인 사용을 가능하게 한다. 메모리 액세스 정책 및 트래픽 프로파일의 상세들이 전력 상태 이용 및 궁극 효율을 결정할 수도 있다. 위에서 논의한 바처럼, 제어기-메모리 시그널링 인터페이스가 유틸상태일 때, 클럭 분배를 그의 근원에서 싱크러너스하게 일시정지하고, 메모리 제어기 및 메모리 디바이스 양쪽 모두에서 다운스트림 회로를 깨끗이 멈추게 하고, 위 표에서 도시된 고속 전력-상태 천이 시간을 가능하게 함으로써 전력이 절약된다.

[0137]

다른 전력 모드들 사이의 천이는, 예를 들면, 호스트 프로세서 또는 호스트 제어기로부터의 명시적 전력-관련 제어 신호 및/또는 트랜잭션 큐 (109) 의 상태 (빈 또는 로딩된 상태) 에 기초하여 도 1a의 전력 모드 로직 (111) 에 의해 관리될 수도 있다. 일 실시형태에서, 예를 들면 도 18a에 대해 도시된, 전력 모드 로직 (111) 은, 메모리 액세스 리퀘스트 없이 시간이 증가함에 따라, 진행적으로 보다 낮은 전력 모드- 액티브 (P4) 으로 클럭-정지 (P3) 로 파워다운 (P2) 로 딥 파워다운(P1) 으로 천이하는 상태 머신을 포함한다. 따라서, 트랜잭션 큐가 먼저 비고 마지막 디큐잉된 트랜잭션이 완료될 때 (즉, 트랜잭션에 관련된 모든 I/O 동작들이 완료될 때), 전력 모드 로직은 액티브 상태에서부터 유틸 상태로 천이하며, 클럭-인에이블 신호(들) 을 디어써팅하여 시스템 클럭 신호 및 제어기 I/O 클럭 신호들을 일시정지한다. 그 후에, 트랜잭션 큐는 계속 예정된 또는 프로그래밍된 수의 메모리 액세스 사이클을 위해 빈 상태로 남아있는 경우, 전력 모드 로직은 유틸 상태에서부터 파워다운 상태로 천이할 수도 있고, 신호들을 이슈하여 메모리 디바이스 및 메모리 제어기 내 송신기 및 수신기를 디스에이블링한다. 트랜잭션 큐가 파워다운 모드 P2에 진입한 후에 연장된 시간 인터벌 (예를 들면, 다른 프로그램가능 시간 인터벌) 동안 빈 상태로 남으면, 또는 추가-감소된 전력 모드에 진입하기 위한 명시적 호스트 커맨드가 수신되면, 전력 모드 로직은 제어기-측 코어 내 회로 (예를 들면, 호스트-측 데이터 경로와 인터페이싱하기 위한 회로)와 함께 제어기-측 PLL의 동작을 디스에이블링함으로써 딥 파워다운 모드에 진입할 수도 있다. 제어기-코어 클럭이 PLL에 의해 생성되는 실시형태에서, 다른 클럭 소스가 호스트 프로세서 또는 호스트 제어기로부터 웨이크-업/파워-업 커맨드 또는 어느 액세스 리퀘스트에 응답하는데 필요한 회로로 (예를 들면 멀티플렉서를 통해) 스위칭가능하게 제공될 수도 있음에 유의한다. 또한, 도 2a에 도시된 바처럼, PCK8 신호를 분주함으로써 제어기 코어 클럭을 생성하는 대신에, 기준 클럭 신호 또는 그의 복원된 버전이 제어기-코어 클럭으로서 사용되며, 따라서 심지어 PLL 셧다운 후에도 코어-클럭 클럭 이용가능성을 보장할 수도 있

다.

- [0138] 계속 도 18a를 참조하면, 메모리 액세스 리퀘스트 또는 명시적 웨이크-업/파워-업 커맨드가 호스트 제어기/호스트 프로세서로부터 수신될 때, 전력 모드 로직은 제어기-측 PLL 및 다른 디스에이블링된 제어기-코어 회로를 턴 온함으로써 딥 파워 다운 상태 (P1) 으로부터 파워 다운 상태 (P2) 로 메모리 시스템을 천이하여 응답한다. 그 후에, 전력 모드 로직은 제어기 및 메모리 측 클럭 및 커맨드/어드레스 송신기 및 카운터-파트 메모리-측 수신기를 인에이블링 시킴으로써 파워 다운 모드 P2로부터 클럭-정지 상태 P3로 시스템을 천이한다. 마지막으로, 전력 모드 로직은 시스템 클럭 신호 및 제어기 I/O 클럭 신호를 토글링하는 것을 가능하게 함으로써 클럭-정지 (유휴) 상태 P3로부터 액티브 상태 P4로 시스템을 천이한다.
- [0139] 도 18b는 도 2a의 실시형태에 대응하지만, P4, P3, 및 P2 전력 모드에서 회로 셋-다운에 관하여 추가 상세들을 나타내는 메모리 시스템 아키텍처 (790) 를 예시한다. 먼저 메모리-측 I/O 회로 (793) 를 먼저 참조하면, 인에이블-읽기 및 인에이블-쓰기 신호 (EnR 및 EnW) 가 메모리 코어로부터 제공되어 데이터 및 마스크 라인 (231, 241) 에 대한 신호 수신기 (234) 및 데이터 링크 (231) 에 대한 신호 송신기 (233) 를 수행되는 칼럼 동작에 따라 선택적으로 인에이블링 및 디스에이블링한다. 즉, 읽기 데이터 또는 읽기-마스크가 수신되지 않는 액티브-모드 (P4) 메모리 읽기 동작 동안, 메모리 코어 로직 내 리퀘스트 디코딩 로직은 인에이블-쓰기 신호 (EnW) 를 낮추어 쓰기-데이터 및 쓰기-마스크 수신기 내 전력-소비 회로를 셋-오프시켜, 전력 소비를 감소시킨다. 유사하게, 읽기 데이터가 송신되지 않는 액티브-모드 메모리 읽기 동작 동안, 메모리 코어 로직은 인에이블-읽기 신호 (EnR) 를 낮추어서 읽기-데이터 송신기 내 전력-소비 회로를 셋 오프시킨다.
- [0140] 일 실시형태에서, I/O 증폭기 셋-다운이 데이터 신호들을 수신/송신하기 위해 제공된 차동 또는 싱글-엔드형 수신기/송신기 내 하나 이상의 바이어스 전류 소스(들) 을 디스에이블링함으로써 이루어진다. 도 18c는 그러한 수신기 또는 송신기의 일부를 형성할 수도 있는 차동 증폭기 (810) 의 예시적인 실시형태를 예시한다. 도시된 바처럼, 증폭기 (810) 는 패시브 또는 액티브 풀-업 로드 (811), 차동 커플링된 입력 트랜지스터 (813a/813b), 바이어싱 전류 소스 (815) 및 셋-다운 트랜지스터 (817) 를 포함한다. 인에이블-쓰기 또는 인에이블-읽기 신호 (총칭적으로, "En") 가 높여질 때, 셋-다운 트랜지스터 (817) 이 도통 상태로 스위칭되어 바이어싱 전류 소스 (815) 내 DC 바이어스 전류의 흐름을 가능하게 하며 따라서 차동 증폭기 (outP, outN)의 출력 노드들이 증폭기의 입력 노드 inP 및 inN에서 적용된 차동 신호에 따라 차동적으로 높여지고 낮추어지는 것을 가능하게 한다. 인에이블 신호가 낮추어질 때, 셋-다운 트랜지스터 (817) 가 실질적으로 비-도통 상태로 스위칭되어 DC 바이어스 전류의 흐름을 디스에이블링시키고, 따라서 증폭기를 감소된 전력 상태로 만든다. 다른 실시형태에서, 셋-다운은 예를 들면 바이어싱 전류 소스 (815) 내를 포함하지만 이에 한정되지는 않는 증폭기 (810) 내 다른 로케이션들에서 셋다운 트랜지스터 또는 다른 스위칭 엘리먼트를 포함함으로써 이루어질 수도 있다.
- [0141] 도 18b에 도시된 신호 수신기 (234) 및 송신기 (233) 는 인커밍 및 아웃고잉 신호들을 증폭하는 것 이상을 행할 수도 있고 따라서 도 18c의 예시적인 증폭기 이외의 (또는 그 대안의) 회로를 포함할 수도 있음에 유의해야 한다. 예를 들면, 수신기 및/또는 송신기 회로 ("수신기/송신기") 는 부가적으로 (예를 들면, 시그널링 링크 상에 전달된 스몰-스윙 신호들과 직렬화 회로로부터 수신되거나 직렬화 회로에 제공된 로직-레벨 신호들 사이에서 시프팅하는) 레벨-시프팅 동작을 수행할 수도 있다. 수신기/송신기는 타이밍된 샘플링/출력 동작의 수행, 전압 증폭/감쇠로 (또는 없이) 전류 드라이브의 증가; 슬루 레이트 제어의 제공, 전압 조절의 공급 등을 할 수도 있다. 이들 동작들의 어느 것 또는 모두는 정상-상태 ("DC") 전류 소스 또는 다른 전력-소비 회로를 사용할 수도 있는데, 이들은 인에이블 신호에 응답하여 신속하게 디스에이블링되고 인에이블링 (즉, 턴 오프 및 온) 될 수도 있다.
- [0142] 도 18d는 인커밍 메모리 쓰기 및 메모리 읽기 리퀘스트 각각에 응답하여 인에이블-쓰기 및 인에이블-읽기 신호 (EnW 및 EnR) 의 커맨드-기반 어썬션을 예시하는 타이밍 도이다. 메모리-측 I/O 클럭 신호 (MCK4), 커맨드/어드레스 신호, 데이터 마스크, 및 읽기/쓰기 데이터 신호들 모두는 도 12a-도 12c를 참조하여 위에서 설명한 것처럼 일반적인 타이밍 관계를 갖는다. 도시된 특정 커맨드 시퀀스에서, 뱅크 어드레스 (Ba) 및 칼럼 어드레스 (Ca) 를 포함하는 칼럼 쓰기 커맨드 (WR) 가 시간 (821)에서 메모리 디바이스 내에서, 예정된 시간, t_{WRD} , 후에 도착하는 대응하는 쓰기 데이터로 수신된다. 리퀘스트 디코딩 로직은 쓰기-인에이블 인터벌 (t_{WR-ENW}) 경과후에, 쓰기-인에이블 신호 (EnW) 를 높임으로써 쓰기 커맨드에 응답하며, 따라서 인커밍 쓰기 데이터에 앞서, 데이터-입력 수신기 (즉, 쓰기-데이터 및 데이터-마스크 수신기)의 동작을 인에이블링하며, 수신기가 안정화하도록 시간 t_{ENW-D} 를 제공한다. 쓰기 데이터가 수신된 후에, 다음 쓰기 리퀘스트가 수신되지 않고 따라서

바로-후행의 (즉, 백-투-백) 쓰기 데이터 수신에 스케줄링되지 않으면, 리퀘스트 디코딩 로직은 인에이블-쓰기 신호를 시간 인터벌 t_{D-ENW} 후에 디어써팅하여 데이터-입력 수신기를 감소된 전력 상태로 리턴시킬 수도 있다.

특정 시간 인터벌이 예시를 위한 목적으로만 도시되어 있음에 유의한다; 다른 t_{WRD} , T_{WR-ENW} , t_{ENW-D} 및 t_{D-ENW} 인터벌들이 다른 실시형태에서 구현될 수도 있다.

[0143] 계속 도 18d를 참조하면, 뱅크 어드레스 (Ba) 및 칼럼 어드레스 (Ca) 를 포함하는 칼럼 읽기 커맨드 (RD) 가 시간 (823)에서 메모리 디바이스 내에서, 예정된 시간, t_{CL} , 후에 출력될 대응하는 읽기 데이터로 수신된다. 리퀘스트 디코딩 로직은 읽기-인에이블 인터벌 (t_{RD-ENR}) 경과후에, 읽기-인에이블 신호 (EnR) 를 높임으로써 읽기 커맨드에 응답하며, 따라서 읽기-데이터 송신 전에, 데이터-출력 송신기 (즉, 읽기-데이터 송신기)의 동작을 인에이블링하며, 송신기가 안정화하도록 시간 t_{ENR-Q} 를 제공한다. 읽기 데이터가 출력된 후에, 다음 읽기 동작이 수신되지 않고 따라서 바로-후행의 (즉, 백-투-백) 읽기 데이터 송신이 스케줄링되지 않으면, 리퀘스트 디코딩 로직은 인에이블-읽기 신호를 시간 인터벌 t_{Q-ENR} 후에 디어써팅하여 데이터-출력 송신기를 감소된 전력 상태로 리턴시킬 수도 있다. 특정 시간 인터벌이 예시를 위한 목적으로만 도시되어 있음에 유의한다; 다른 t_{CL} , t_{RD-ENR} , t_{ENR-Q} 및 t_{Q-ENR} 인터벌들이 다른 실시형태에서 구현될 수도 있다.

[0144] 도 18b로 돌아가면, 트랜잭션 큐가 클럭-정지 저 전력 모드, P3 (즉, 유휴 모드) 로 천이한 후에 빈 상태로 계속 남아 있음을 결정할 때 제어기 코어 내 전력-모드 로직에 의해 파워다운 신호 또는 커맨드 (PD) 가 어써팅된다. 파워다운 신호가 전력 모드 드라이버 (795), 링크 (PM[1]), 및 수신기 (797) 를 통해 메모리 디바이스로 포워드되며, 수신기 (797) 에서 그것은 인에이블 로직 회로 (799) 내에서 수신되고 인에이블 로직 회로 (799) 는 응답하여 커맨드-인에이블 신호, EnCK/CA 를 낮춘다. 커맨드-인에이블 신호는 입력 수신기 (223a/223b)로 공급기로 그리고 시스템 클럭 인터페이스 (221) 및 커맨드/어드레스 인터페이스 (243) 내로 공급되고, 따라서 낮추어질 때, 대응하는 클럭 및 커맨드/어드레스 링크들에 대한 입력 수신기들을 디스에이블링하여, 본원에서 전력다운 모드, P2로 지칭되는, 추가-감소된 전력 상태를 확립한다.

[0145] 도 18e는 전력 모드 진입 및 종료를 예시하는 타이밍 도이며, 종료는 메모리 쓰기 리퀘스트에 의해 트리거링된다. 도시된 바처럼, 최종 커맨드 메모리 액세스 리퀘스트 (OP) 가 시간 (833) 에서 시작하여 수신되고, 클럭-정지 저 전력 모드로의 진입에 선행하는 (즉 클럭 사이클 (32) 에서) 인터벌 $t_{CA(OP)-CK}$ 동안 처리된다. 위에서 논의된 바처럼, $t_{CA(OP)-CK}$ 인터벌은 상이한 커맨드들에 대해 상이할 수도 있고 트랜잭션 큐로부터 디큐잉된 마지막 메모리 액세스 리퀘스트를 완료 (즉 그의 완료를 위해 클럭 에지들을 공급) 하는데 필요한 시간을 나타낸다. 메모리 읽기 동작은 예를 들면,로우 프리차지 커맨드 (row precharge command) 보다 완료하는데 더 많은 클럭 에지들을 필요로 할 수도 있다.

[0146] 트랜잭션 큐가 클럭 정지 후 예정된 또는 프로그래밍된 시간 인터벌 (t_{CK-PM}) 동안 계속 빈상태로 남으면, 제어기-측 전력-모드 로직이 파워다운 신호 (PD) 를 어써팅하며, 이는 커맨드-인에이블 신호, EnCK/CA의 디어써션을 짧은 시간 후(즉, 지연 t_{PM-EN} 후) 에 발생시켜서, 시스템-클럭 및 커맨드/어드레스 링크들에 대한 입력 수신기들을 디스에이블링하고 파워다운 모드를 확립한다.

[0147] 파워다운 모드 (P2) 는 커맨드/어드레스 경로를 통해 송신된 커맨드에 응답하여 대안으로 또는 부가적으로 진입될 수도 있음에 유의해야 한다. 그러한 배열은 파워다운 제어가 P4 또는 P3 모드에 있는 동안 다른 동작들을 나타내는 하나 또는 커맨드들로 (예를 들면, 엠베딩된 비트 또는 비트들로서) 포함되는 것을 허용한다. 커맨드 경로 상에서 수신된 커맨드를 통해 파워다운 모드로의 진입후, 파워다운 신호가 커맨드 경로 및 클럭 신호 수신기의 리-인에이블링을 트리거링하고 따라서 클럭-정지 모드 (P3) 로 다시 천이를 이룬다.

[0148] 제어기-측 트랜잭션 큐 내에서 새로운 메모리 액세스 리퀘스트가 큐잉될 때, 전력 모드 로직은 파워다운 신호를 낮추어 파워다운 모드로부터 클럭-정지 모드로 (즉, P2로부터 P3로) 천이를 가능하게 한다. 메모리-측 인에이블 로직 (도 18b의 799) 는 짧은 시간 후 (즉, 인터벌 t_{PM-EN}) 후 커맨드-인에이블 신호를 높여서 파워다운 신호의 디어써션에 응답하며 클럭 및 커맨드/어드레스 링크에 대한 입력 수신기를 인에이블링하고 따라서 액티브 모드로의 리턴을 위해 메모리-디바이스를 준비시킨다. 따라서, 커맨드-인에이블 신호가 높아진 후 시간 인터벌 (t_{EN-CK}) 에서, 시스템 클럭이 재시작되어 클럭-정지 모드로부터 다시 액티브 모드 (P3에서 P4) 로 메모리 디바이스를 천이한다. 그후에 바로, 액티브 모드로의 리턴을 트리거링한 메모리 액세스 리퀘스트가 커맨드 경로를 통해 메모리 디바이스 내에서 수신되며, 예정된 시간 후에 대응하는 데이터가 뒤따른다. 도시된 특

정 실시형태에서, 메모리 액세스 리퀘스트가 칼럼 쓰기 리퀘스트가어서 도시된 시간에서 쓰기 데이터를 수신하는 쓰기-데이터 수신기의 입력 증폭기를 인에이블링하도록 인에이블-쓰기 신호가 어써팅된다. 다르게는 메모리 액세스 리퀘스트가 도 18d에서 도시된 바처럼 칼럼 읽기 리퀘스트이거나 (이 경우에, 인에이블-읽기 신호가 시간 t_{RD-ENR} 후에 높여져서, 읽기-데이터 증폭기의 동작을 인에이블링 시킨다), 로우 액세스 리퀘스트일 수도 있다.

[0149] 도 18a-도 18e 를 참조하여 설명된 지원 회로 및 전력-모드 천이를 숙고하면, 전력 모드 제어에서 추가의 리파인먼트 (refinement) 가 제공될 수도 있음에 유념해야 한다. 예를 들면, 시스템 클럭 및 커맨드/어드레스 링크들을 위한 입력 수신기들이 단일 제어 신호에 의해 인에이블링되는 것으로 도시될지라도, 분리된 인에이블 신호들이 다른 실시형태에서 그러한 링크들을 위해 제공되며, 따라서 하나의 링크가 다른 하나 전 또는 후에 디스에이블링/인에이블링되는 것을 가능하게 하거나 링크들을 서로 독립적으로 인에이블링하는 것을 가능하게 한다. 게다가, 특히 그러한 버퍼들이 무시가능하지 않은 전류 (예를 들면, 전류 모드 로직)을 드로잉 (drawing) 하는 회로에 의해 구현되면, 클럭 링크 (또는 하나 이상의 추가 인에이블 신호) 를 위한 입력 수신기를 인에이블링시키기 위해 적용된 신호가 추가적으로 클럭 버퍼 (229) 및 (227) 에 공급될 수도 있다. 부가적으로, 전력 모드 신호가 전용 링크를 통해 공급되는 것으로 표시될지라도, 다르게는 파워다운 신호가 공유된 링크를 통해 송신 (예를 들면, 도 2에 도시된 사이드밴드 링크 상으로 시간-멀티플렉싱) 되어 핀 카운트를 감소시킬 수도 있다. 또한, 로직 회로는 제어기-측 회로 내에 제공되어 제어기-코어 클럭 신호 또는 다른 제어기-측 타이밍 신호로 파워다운 신호를 동기화할 수도 있다. 게다가, 도 18b에 특정적으로 도시되지는 않지만, 추가의 인에이블 신호들이 수행되는 동작 및 전력 모드에 따라 제어기 측 송신기 및 수신기 회로를 선택적으로 인에이블링하도록 제공될 수도 있다. 예를 들면, 인에이블-읽기 및 인에이블-쓰기 신호 (EnR, EnW) 에 대응하는 신호들이 제어기 코어 로직으로부터 데이터 수신기 및 데이터/마스크 송신기 회로 (예를 들면, 도 18b의 엘리먼트 (188), (187)) 로 제공되어 데이터/마스크 송신기를 메모리 쓰기 동작 동안 디스에이블링 (하고 데이터/마스크 송신기를 인에이블링) 한다. 또한 파워다운 신호가 커맨드/어드레스 직렬화기 (207) 및 어느 클럭 송신기 (175) 및 온-칩 클럭 분배 회로 (173) 내 커맨드/어드레스 송신기를 선택적으로 가능하게 하는데 사용될 수도 있으며, 따라서 메모리 제어기 내 파워다운 모드 (P2) 를 가능하게 하고 클럭-정지 동작 자체에 덧붙여 전력 절약을 제공한다. 제어기-측 인에이블-읽기/인에이블-쓰기 및 파워다운 신호들의 타이밍은 일반적으로 도 18d 및 18e에서 카운터-파트 메모리-측 신호들에 대해 도시된 타이밍에 대응한다.

[0150] 이전의 옵션 및 대안들 이외에, 클럭-정지 모드에서 파워다운 모드로의 천이 전에 일어나게 될 t_{CK-PM} 시간 인터벌이 시스템 동작 정책 또는 어플리케이션 디맨드에 따라 (예를 들면, 제어기-측 전력 모드 로직 내 레지스터로 프로그래밍) 프로그래밍가능하게 선택될 수도 있다. 보다 일반적으로는, 도 18d 및 18e에 도시된 모든 타이밍 인터벌들이 예시를 위한 목적으로만 제공되며, 반드시 스케일 대로 일 필요는 없고, 그리고 동작상의 요건을 만족시키기 위해 필요한 것처럼 변화할 수도 있다.

[0151] **물리적 실시형태들의 전자 표현**

[0152] 여기에 개시된 다양한 집적 회로들, 다이스 (dice) 및 패키지들이 컴퓨터 보조 설계 도구를 사용하여 설명될 수도 있고, 다양한 컴퓨터 관독가능 매체에서 구현된 데이터 및/또는 명령들로서, 거동의, 레지스터 전송, 로직 컴포넌트, 트랜지스터 레이아웃 지오메트리, 및/또는 다른 특징들에 관하여, 표현 (또는 표시) 될 수도 있다.

[0153] 하나 이상의 컴퓨터-관독가능 매체를 통해 컴퓨터 시스템 내에 수신될 때, 상술한 회로의 그러한 데이터 및/또는 명령-기반 표현들이, 순-리스트 생성 프로그램, 프레이스 및 라우트 프로그램 등을 포함하지만 이에 한정되지 않는 하나 이상의 다른 컴퓨터 프로그램들의 실행과 함께 컴퓨터 시스템 내에 프로레싱 엔티티 (예를 들면, 하나 이상의 프로세서) 에 의해 프로세싱되어 그러한 회로들의 물리적 발현 (physical manifestation) 의 이미지 또는 표시를 생성할 수도 있다. 그러한 표시 또는 이미지는 그 후에, 예를 들면 디바이스 방식 프로세스에서 회로들의 다양한 컴포넌트들을 형성하는데 사용되는 하나 이상의 마스크들의 생성을 가능하게 함으로써, 디바이스 제조에 사용될 수도 있다.

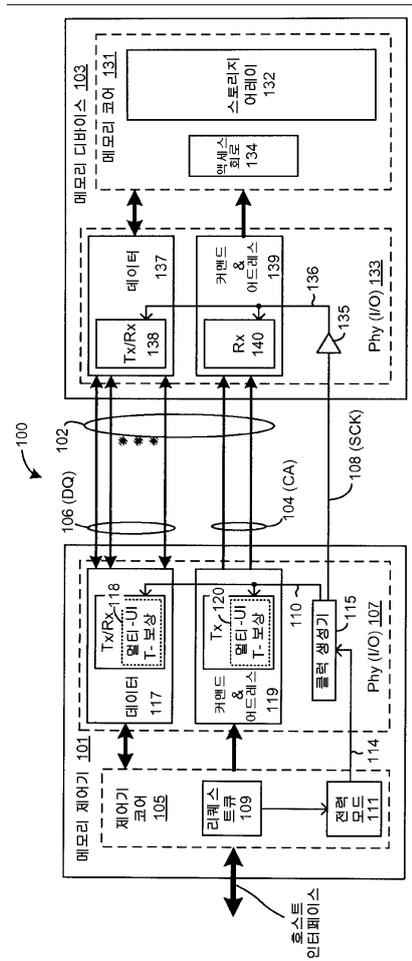
[0154] 이전의 설명 및 첨부 도면들에서, 특정 용어 및 도면 부호들은 본 발명의 완전한 이해를 제공하기 위해 제시되었다. 일부 예에서, 용어 및 부호들은 본 발명을 실시하는데 필요하지 않는 특정 상세들을 내포할 수도 있다. 예를 들면, 특정 수의 비터들, 신호 경로 폭, 시그널링 또는 동작 주파수, 컴포넌트 회로 또는 디바이스들 중의 어느 것이 다른 실시형태에서 상술한 것들과는 상이할 수도 있다. 다른 예들에서, 잘-알려진 회로들 및 디바이스들은 본 발명을 불필요하게 불명료하게 하는 것을 피하기 위해 블록 도 형태로 도시된다. 부가적으로, 회로 엘리먼트들 또는 블록들 사이의 상호 접속은 버스를 또는 단일 신호 라인으로서 도시될 수도

있다. 버스들의 각각은 다르게는 단일 신호 라인일 수도 있고, 단일 신호 라인들의 각각은 다르게는 버스들 일 수도 있다. 싱글-엔드형으로서 도시되거나 설명된 시그널링 경로들은 및 시그널들은 또한 차동일 수도 있으며, 그 역 또한 같다. 신호 구동 회로가 신호 드라이빙과 신호 수신 회로들 사이의 커플링된 신호 라인 상의 신호를 어써팅 (또는 명시적으로 언급되거나 문맥에 의해 나타내어지는 경우, 디어써팅) 할 때, 신호 구동 회로는 신호 수신 회로로 신호를 "출력" 한다고 불려진다. 표현 "타이밍 신호" 는 본원에서 집적 회로 디바이스 내 하나 이상의 액션들의 타이밍을 제어하고, 클럭 신호, 스트로브 신호 등을 포함하는 신호를 지칭하도록 사용된다. "클럭 신호" 는 본원에서 하나 이상의 집적 회로 디바이스들 상의 회로들 사이의 액션들을 조정하는데 사용되고 프리-러닝 (free-running) 및 게이트된 (gated (즉, 일시정지가능하거나 정지가능한)) 진동 신호들을 포함하는 주기적 타이밍 신호를 지칭하도록 사용된다. "스트로브 신호" 는 본원에서 스트로빙되는 디바이스 또는 회로에 대한 입력에서 데이터의 존재를 마킹하도록 천이하고 따라서 버스트 데이터 송신동안 주기성을 나타내지만, 그렇지 않으면 (파킹된 상태를 벗어난 천이 또는 다른 제한된 프리엠프 또는 포스트엠프 천이를 제외하고) 데이터 송신의 부재시 정상-상태로 남을 수 있는 타이밍 회로를 지칭한다. 용어 "커플링된" 은 본원에서 직접적인 연결 및 하나 이상의 개재 회로 또는 구조들을 통한 접속을 표현하도록 사용된다. 집적 회로 디바이스 "프로그래밍"은 예를 들면, 호스트 명령에 대응하여 디바이스 내 레지스터 또는 다른 저장 회로로 제어 값을 로딩하는 것과 따라서 디바이스의 동작상의 양태를 제어하는 것, 디바이스 구성을 확립하는 것 또는 원-타임 프로그래밍 동작을 통해 디바이스의 동작상의 양태를 제어하는 것 (예를 들면, 디바이스 생산 동안 구성 회로 내 퓨즈 (fuse) 를 블로잉 (blowing) 하는 것) 및/또는 디바이스의 하나 이상의 선택된 핀들 또는 다른 접속 구조를 (또한 스트레이핑 (strapping) 으로서 지칭되는) 레퍼런스 전압 라인들에 연결시키는 것을 포함하여 구체적 디바이스 구성 또는 디바이스의 동작 양태를 확립할 수도 있다. 용어 "예시적인" 및 "실시형태" 는 예를 표현하기 위해 사용되며, 바람직함 또는 요건을 표현하는 것은 아니다.

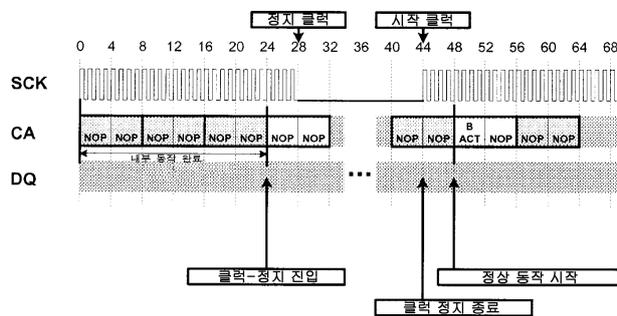
[0155] 본 발명의 특정 실시형태들을 참조하여 본 발명을 설명하였지만, 더 넓은 사상 및 범위로부터 이탈함이 없이 그에 대한 다양한 수정 및 변경들이 이루어질 수도 있음은 명백할 것이다. 예를 들면, 실시형태들 중 어느 것의 특징들 또는 양태들은 적어도 실행가능한 경우, 실시형태들 중 어느 다른 것과 함께 또는 그의 카운터파트 특징들 또는 양태들 대신에 적용될 수도 있다. 따라서, 명세서 및 도면은 제한적인 의미가 아니라 예시적인 것으로 간주되어야 한다.

도면

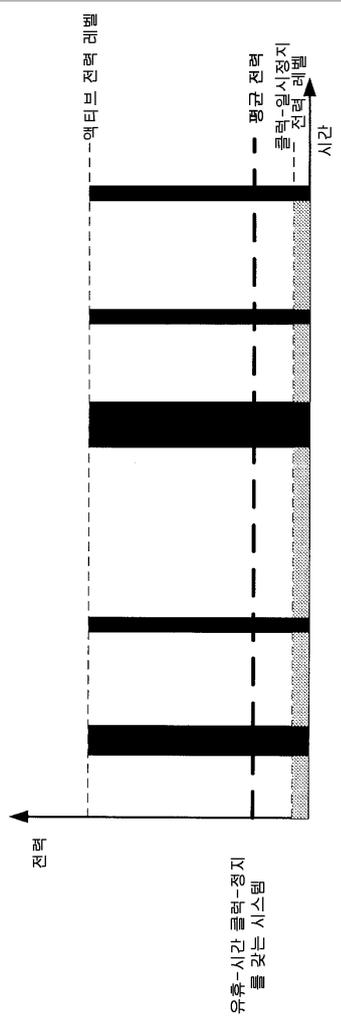
도면1a



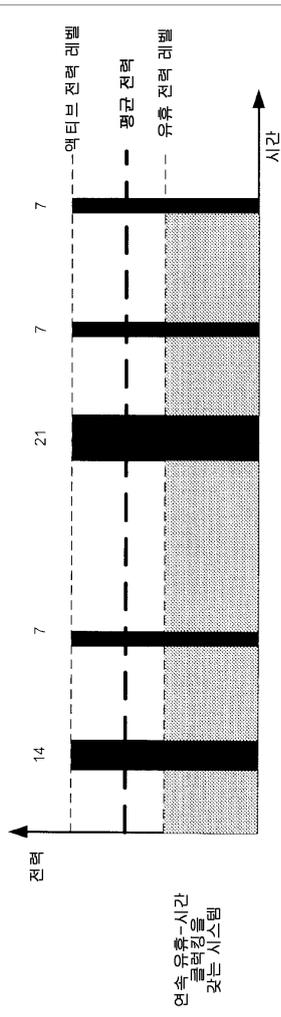
도면1b



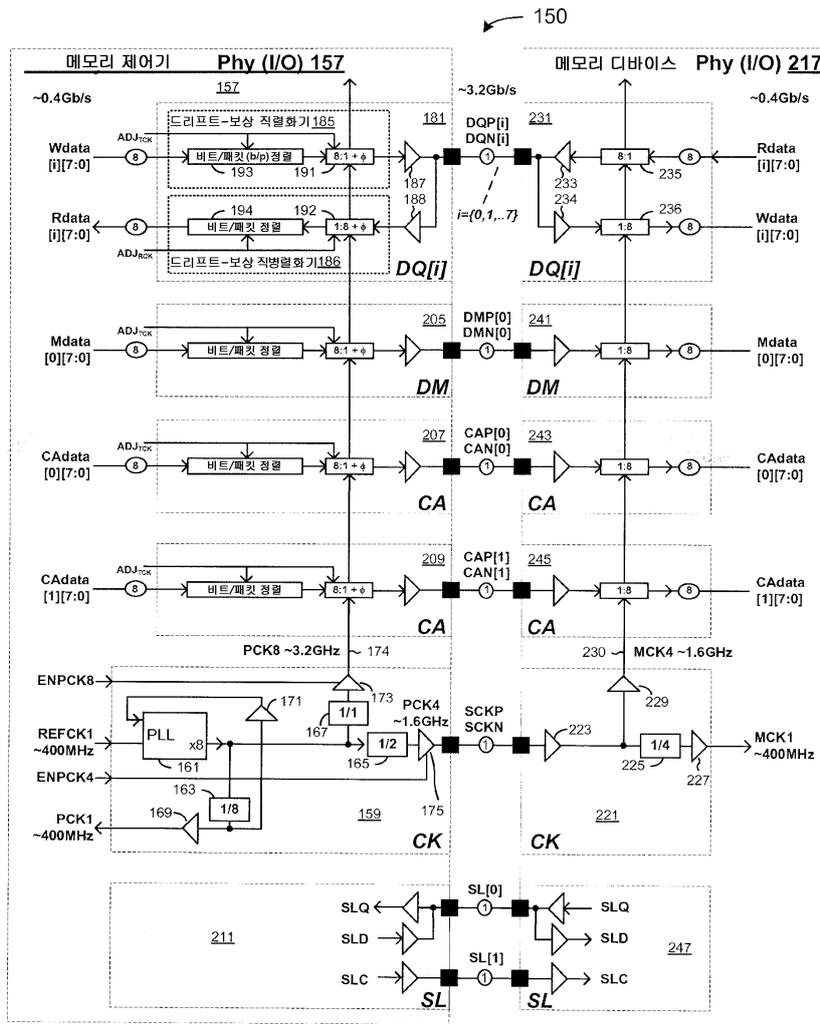
도면1c



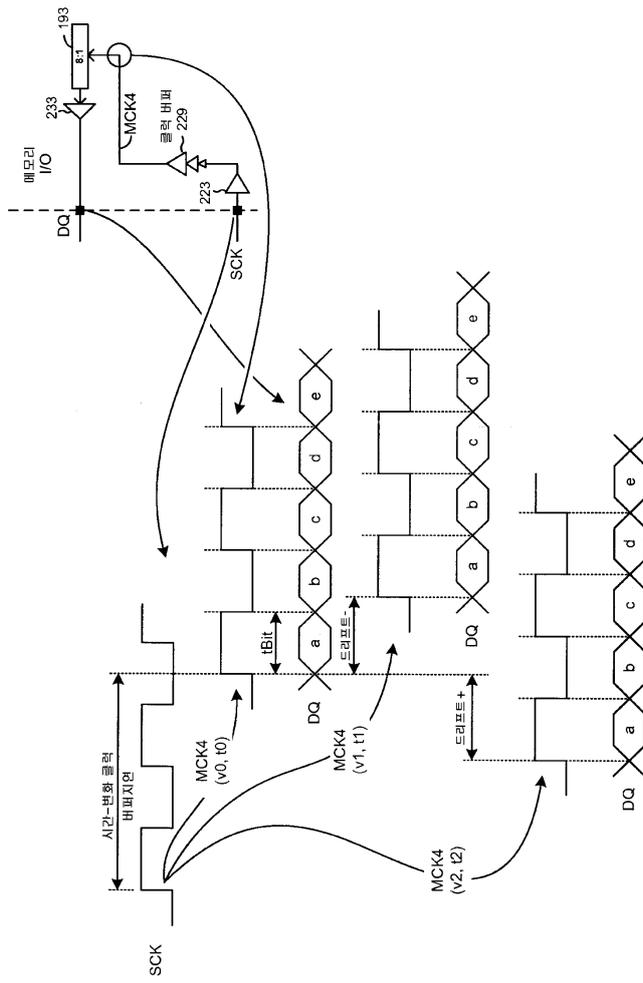
도면1d



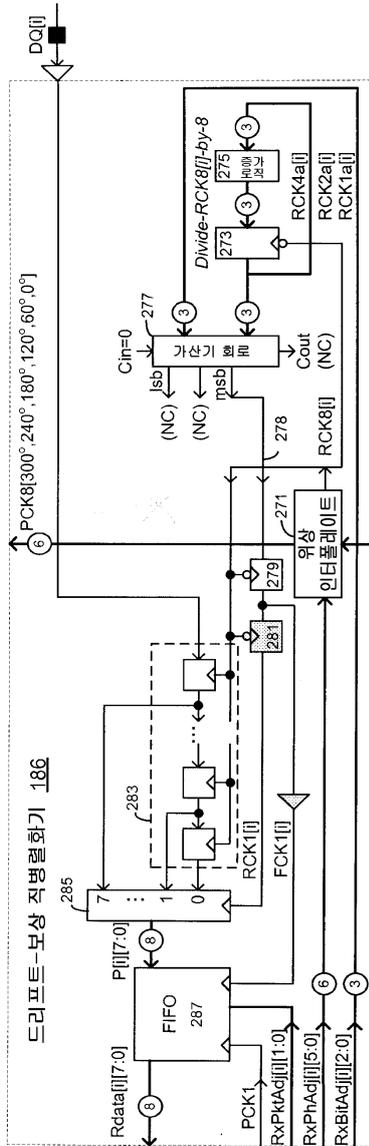
도면2a



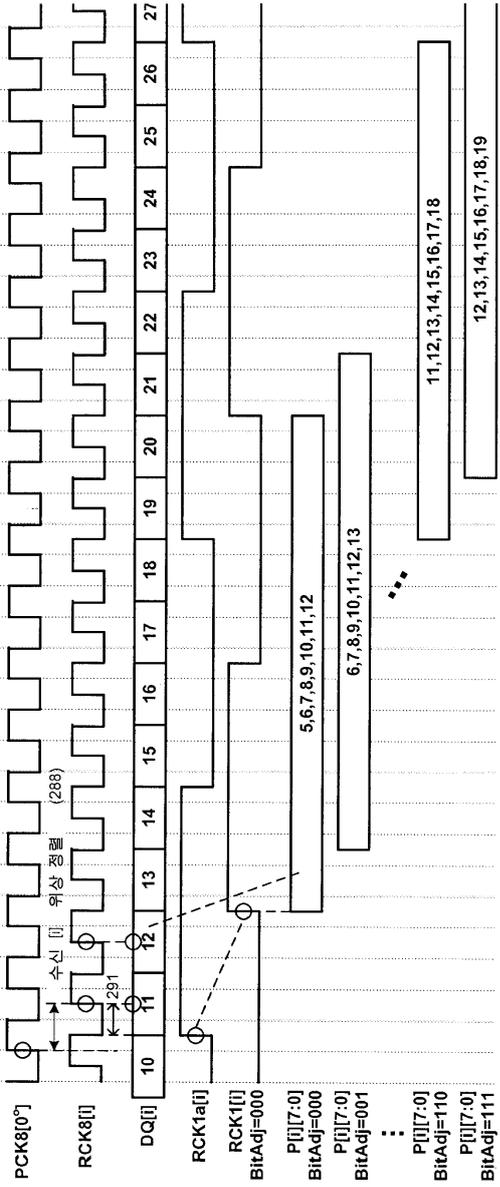
도면2b



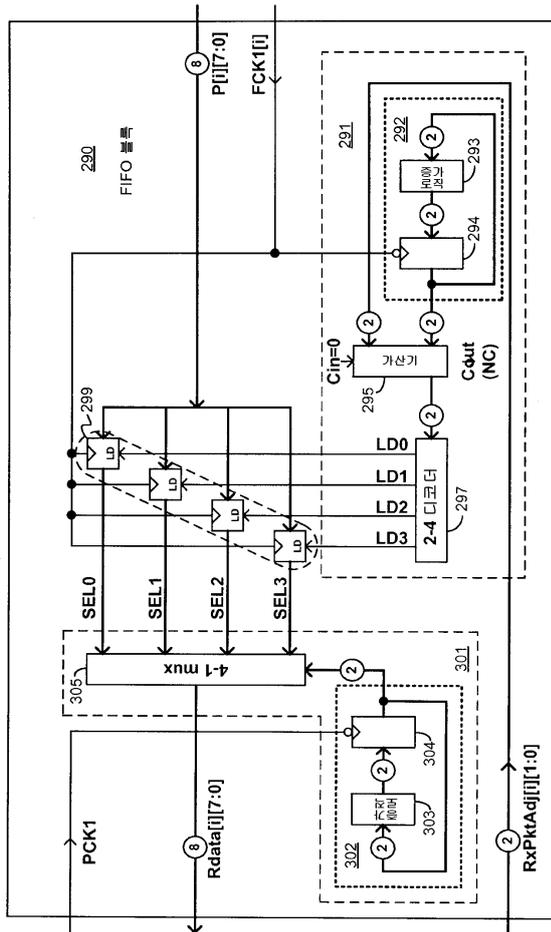
도면3a



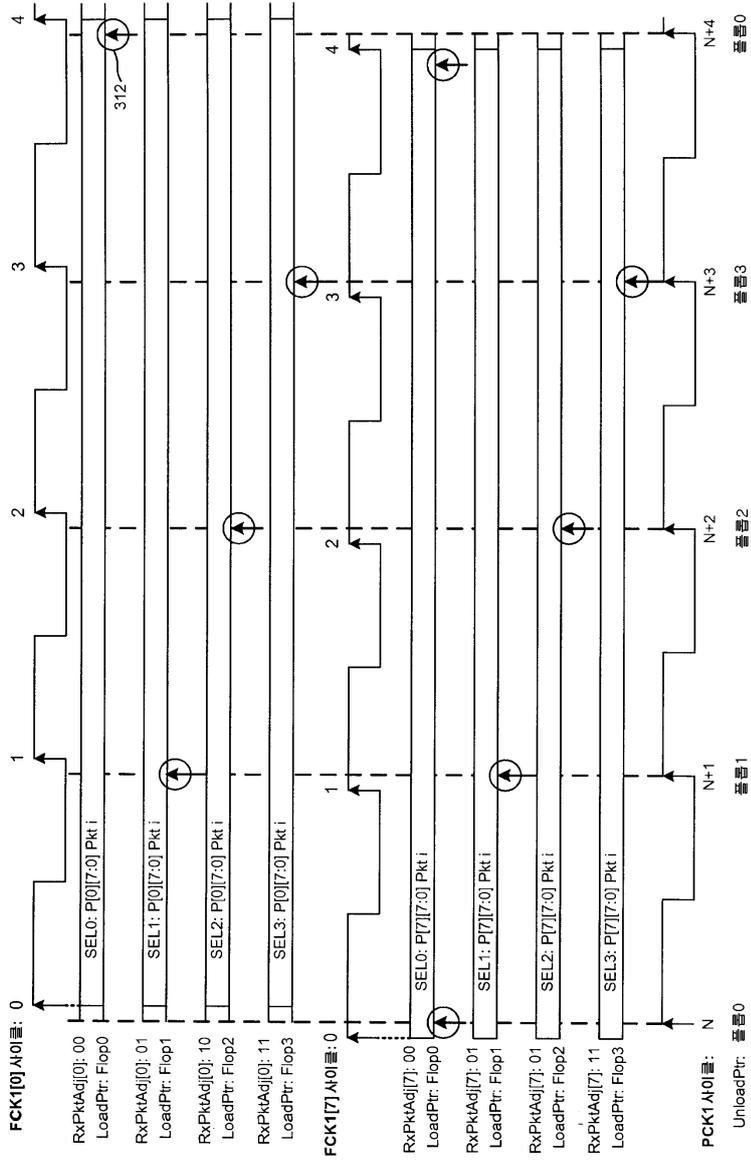
도면 3b



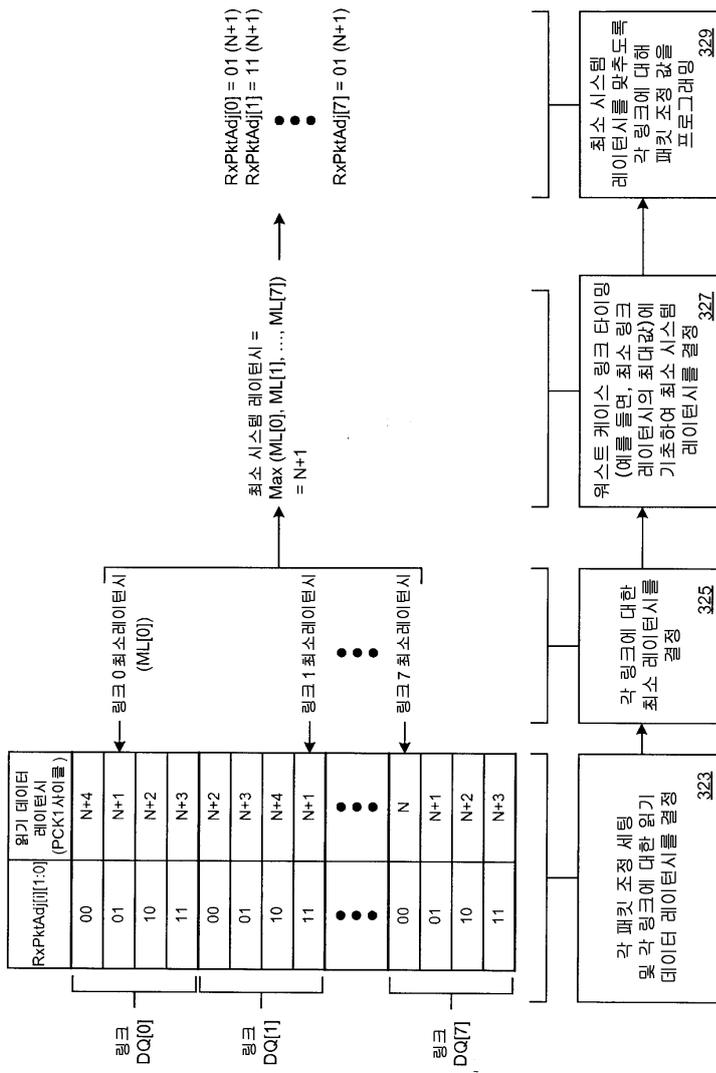
도면3c



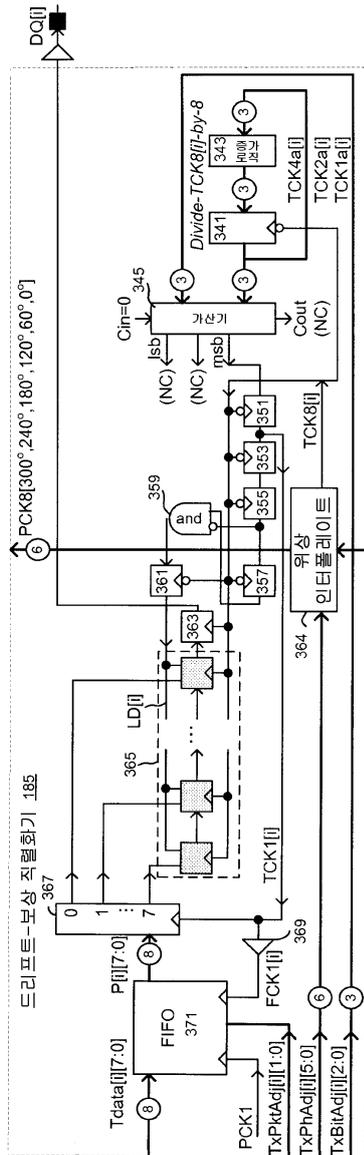
도면3d



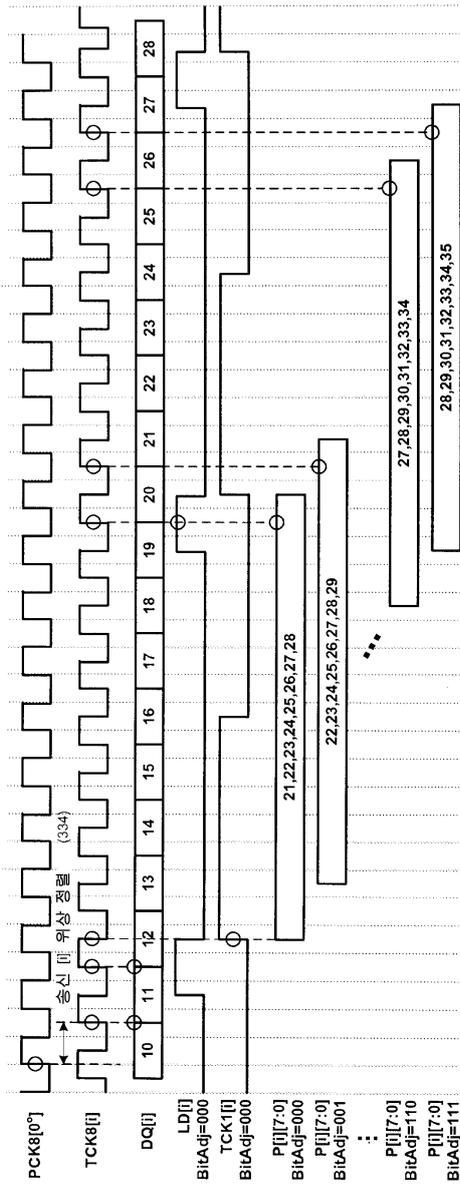
도면3e



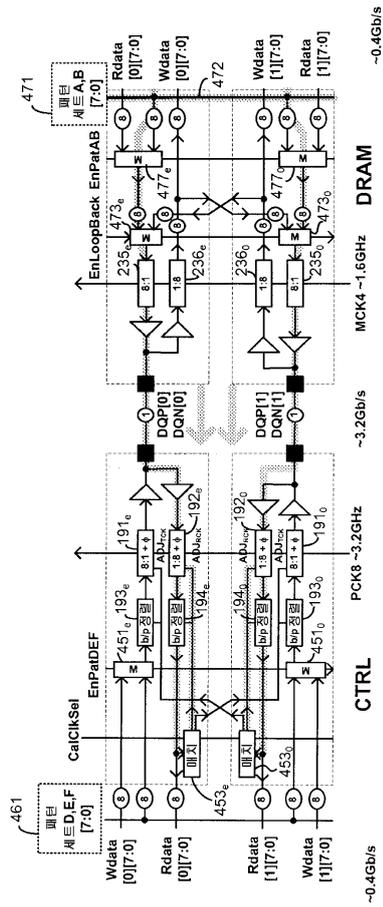
도면3f



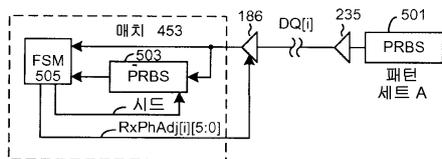
도면3g



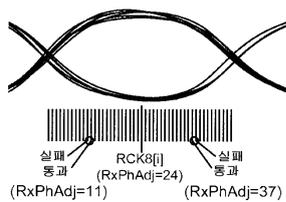
도면5a



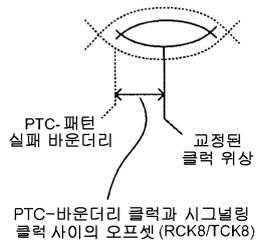
도면5b



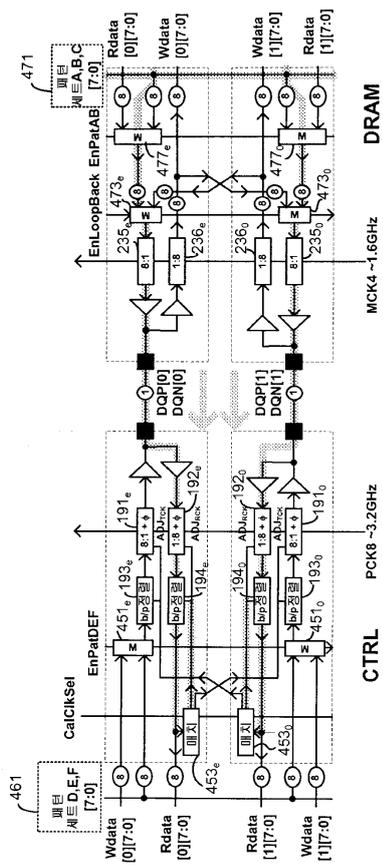
도면5c



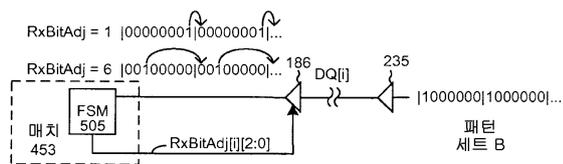
도면5d



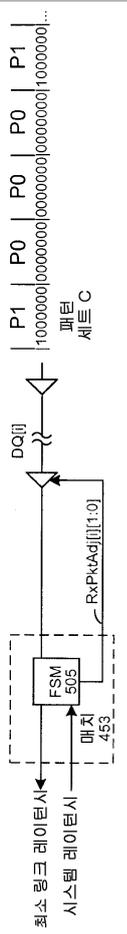
도면6a



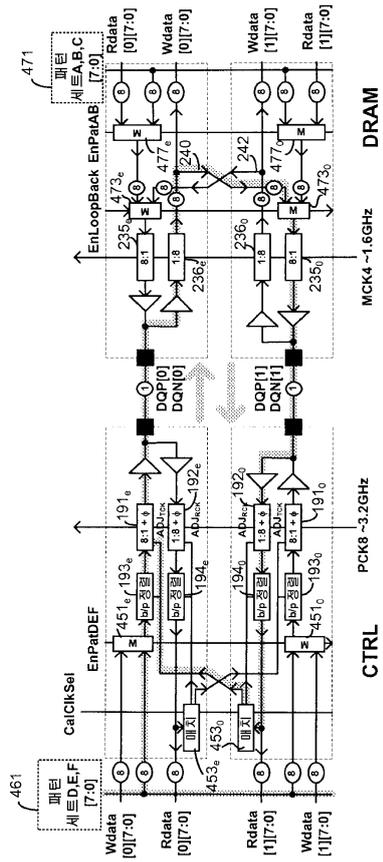
도면6b



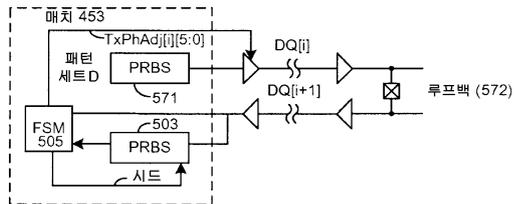
도면6c



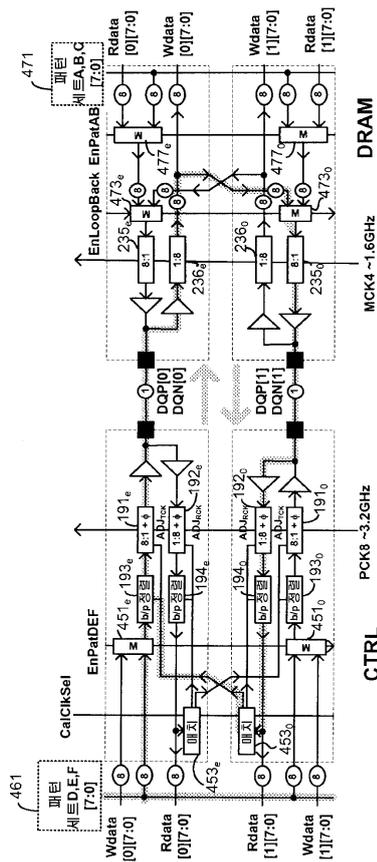
도면7a



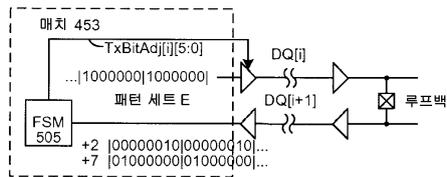
도면7b



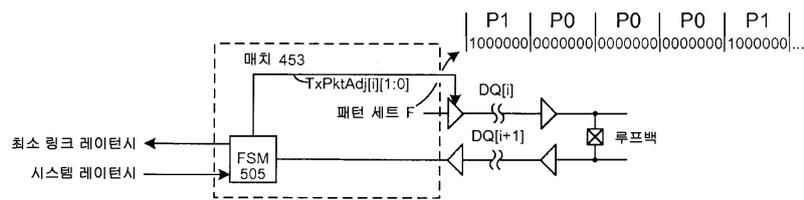
도면8a



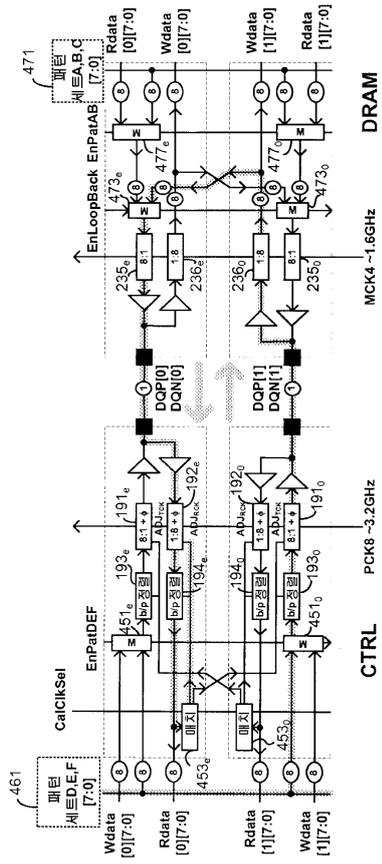
도면8b



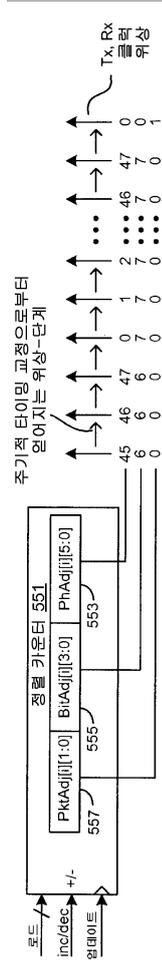
도면8c



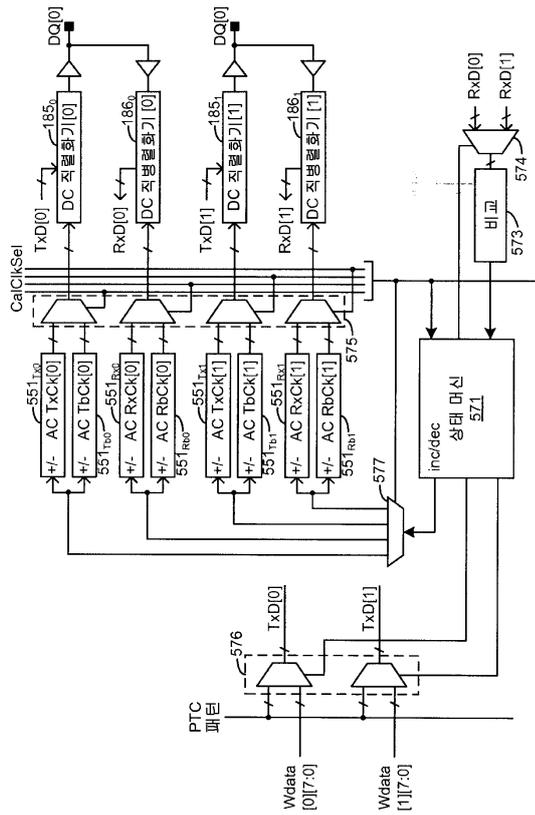
도면9b



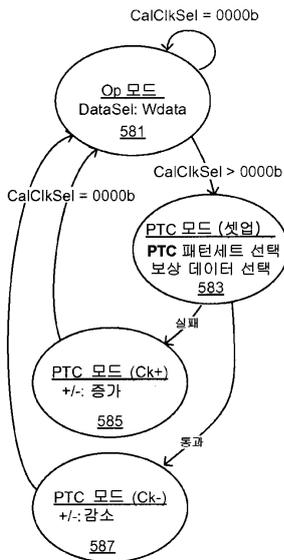
도면10a



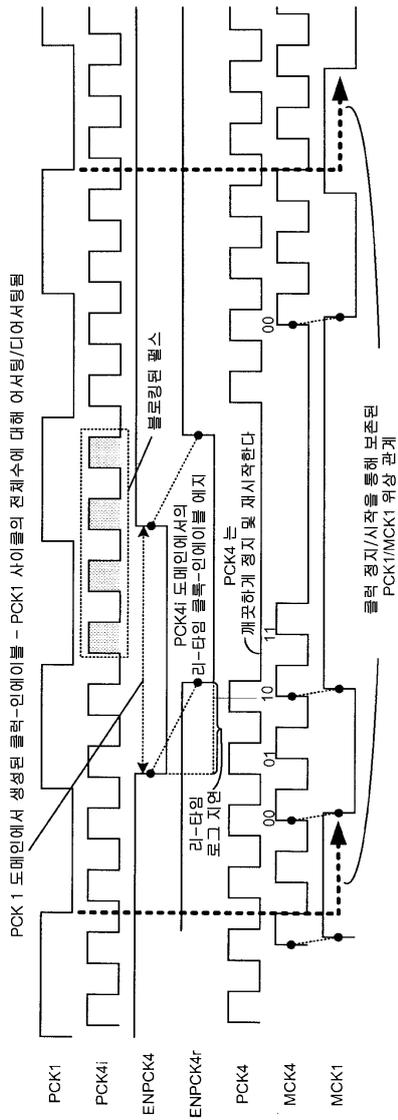
도면10b



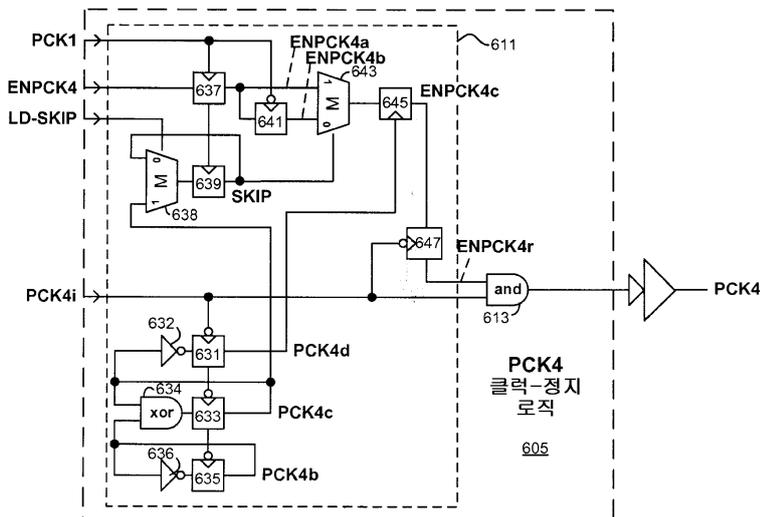
도면10c



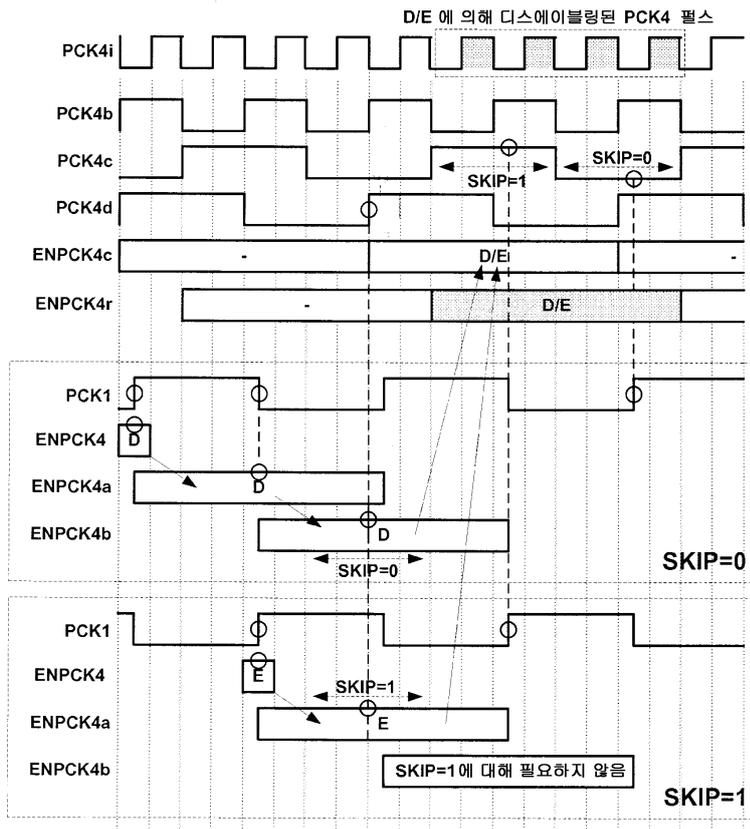
도면11b



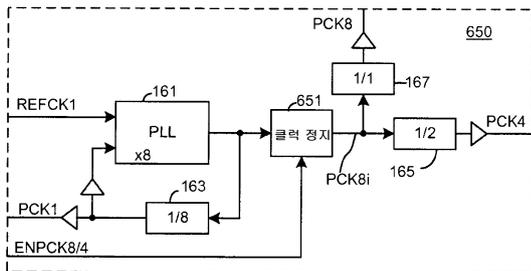
도면11c



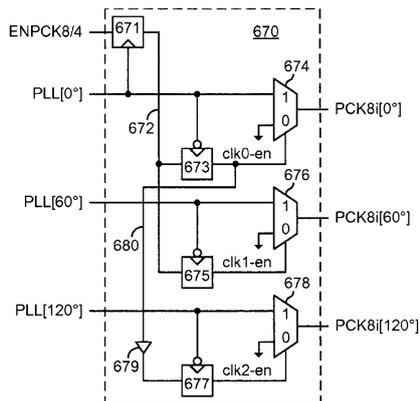
도면11d



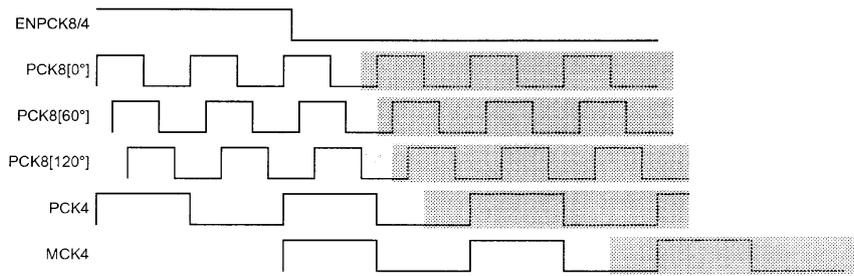
도면11e



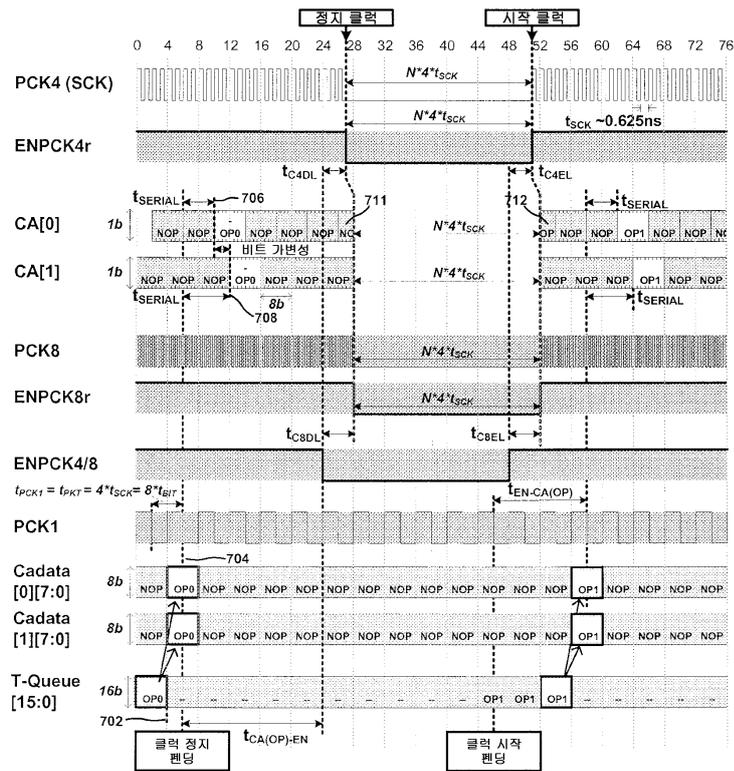
도면11f



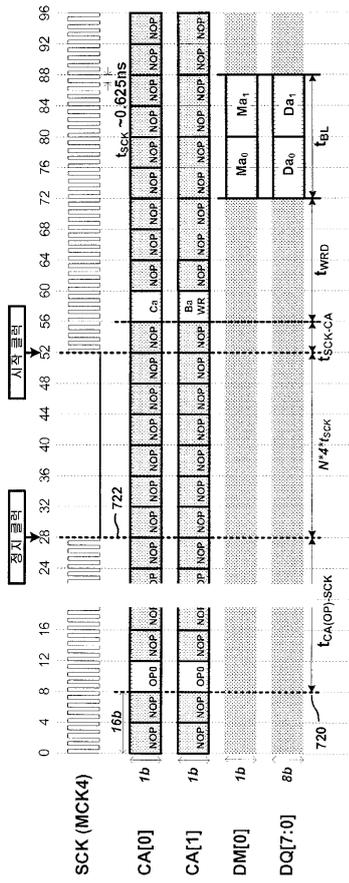
도면11g



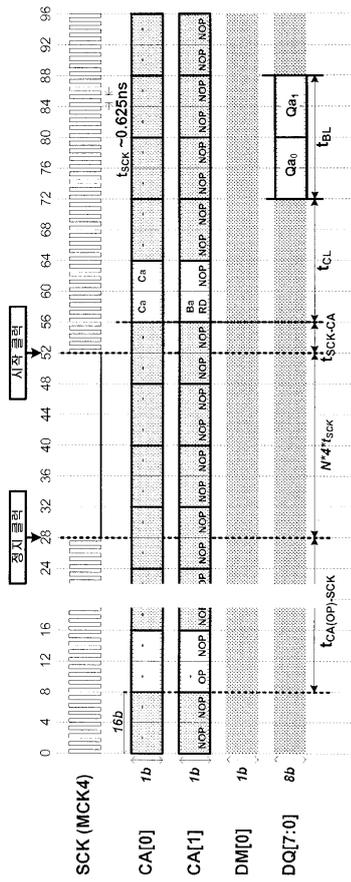
도면12a



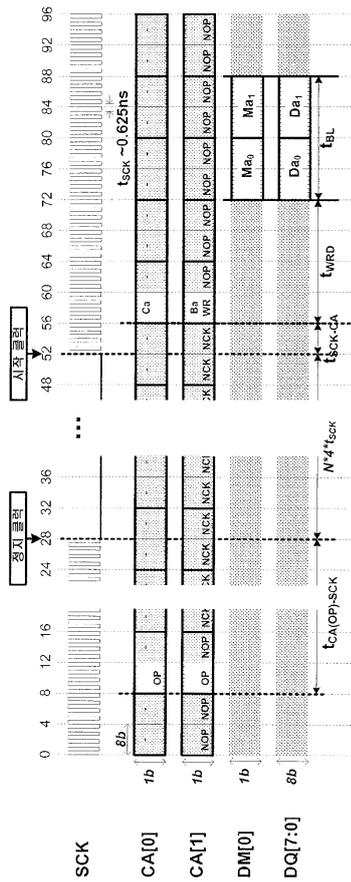
도면12b



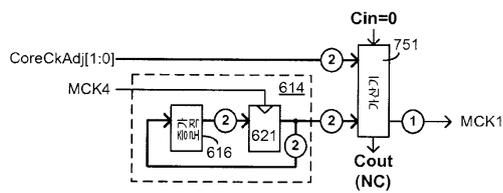
도면12c



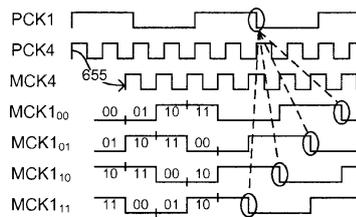
도면13



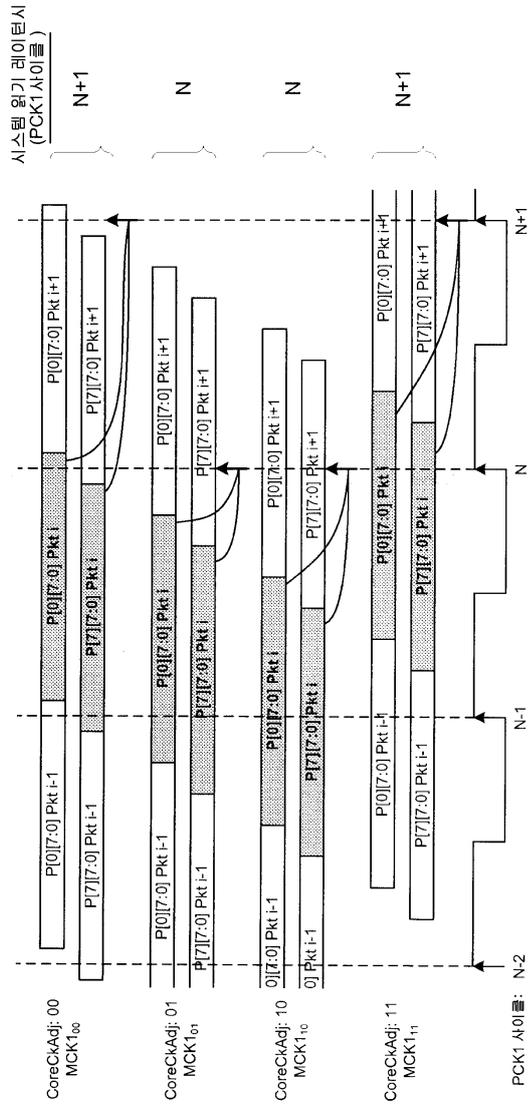
도면14a



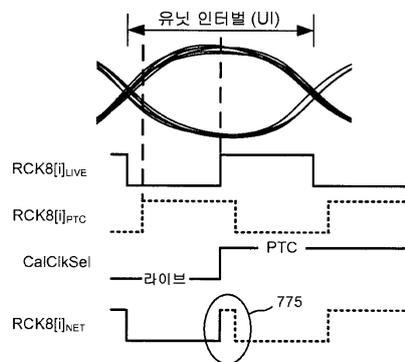
도면14b



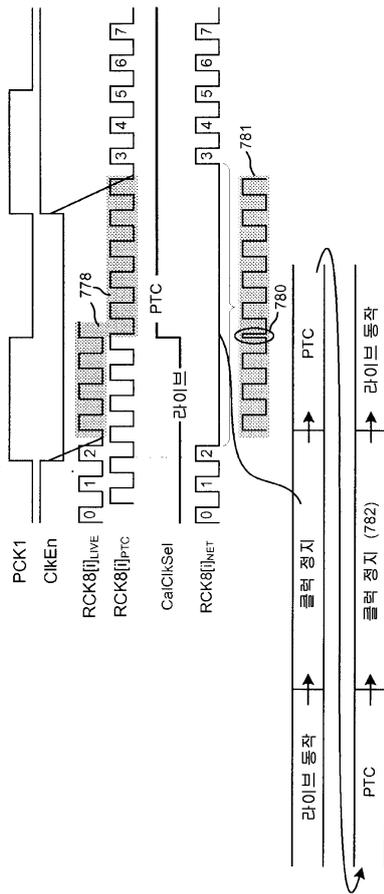
도면14c



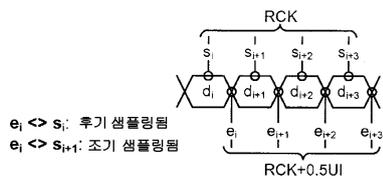
도면15a



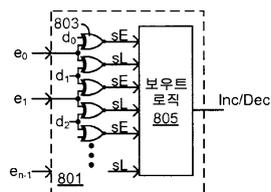
도면15b



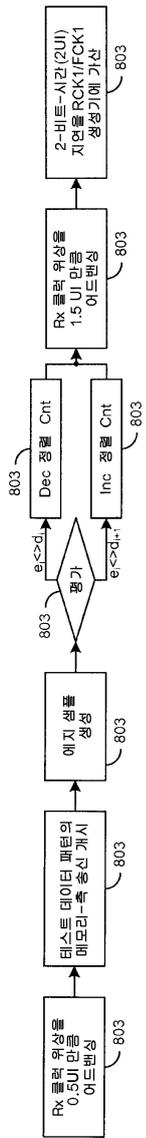
도면16a



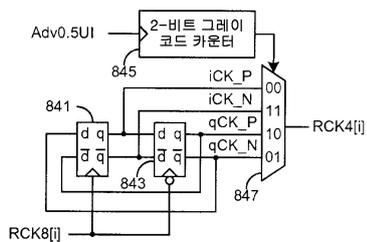
도면16b



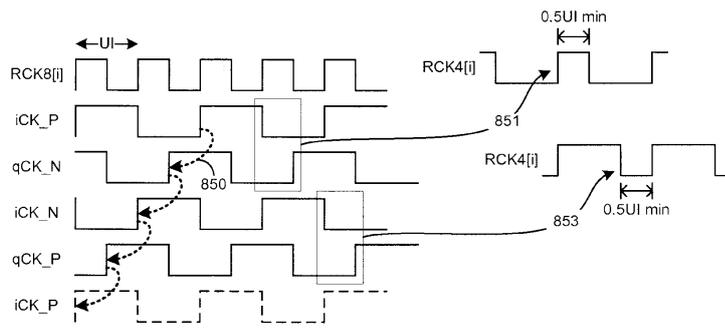
도면16c



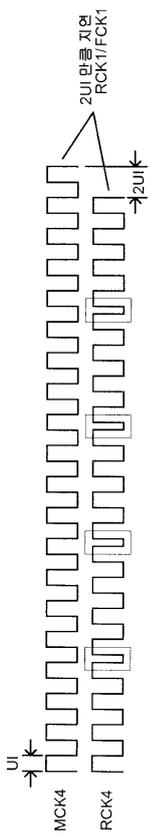
도면16d



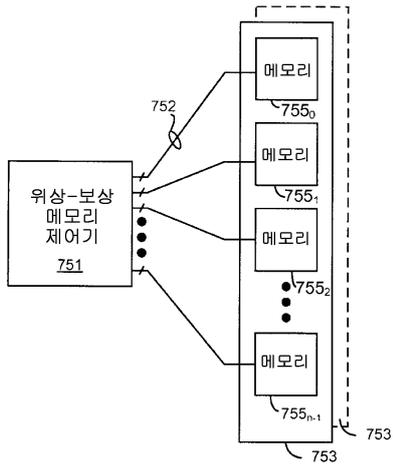
도면16e



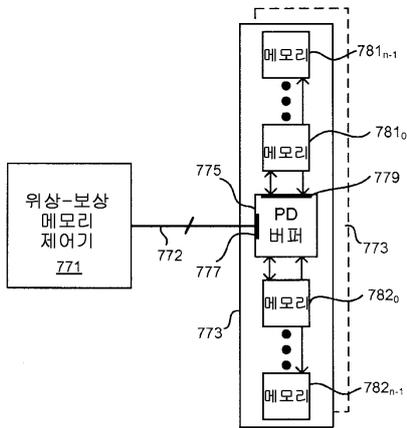
도면16f



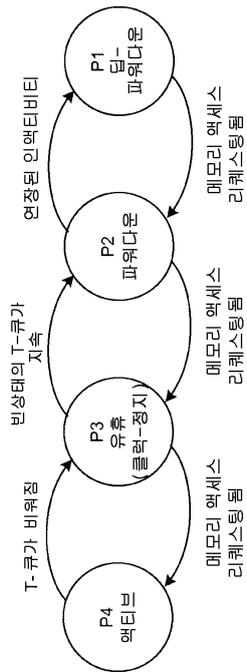
도면17a



도면17b



도면18a



도면18b

