



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2016-0100363
(43) 공개일자 2016년08월23일

- (51) 국제특허분류(Int. Cl.)
G06F 13/42 (2006.01) *G06F 1/04* (2006.01)
- (52) CPC특허분류
G06F 13/4221 (2013.01)
G06F 1/04 (2013.01)
- (21) 출원번호 10-2016-7019339
- (22) 출원일자(국제) 2014년12월17일
심사청구일자 없음
- (85) 번역문제출일자 2016년07월15일
- (86) 국제출원번호 PCT/US2014/070935
- (87) 국제공개번호 WO 2015/095382
국제공개일자 2015년06월25일
- (30) 우선권주장
61/917,895 2013년12월18일 미국(US)
14/572,680 2014년12월16일 미국(US)

- (71) 출원인
퀄컴 인코포레이티드
미국 92121-1714 캘리포니아주 샌 디에고 모어하우스 드라이브 5775
- (72) 벌명자
센고쿠 쇼이치로
미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
- (74) 대리인
특허법인코리아나

전체 청구항 수 : 총 30 항

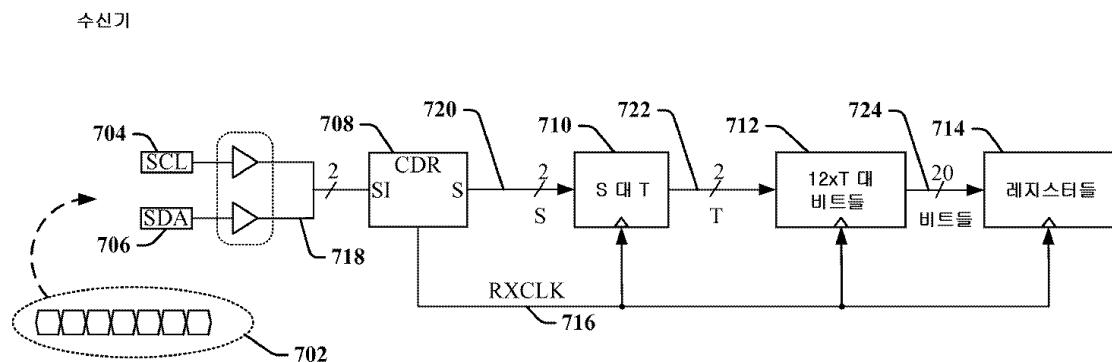
(54) 발명의 명칭 오직 수신기 클록에 의한 CCIe 수신기 로직 레지스터 기입

(57) 요 약

슬레이브 디바이스들에 대한 효율적인 슬립 및 웨이크업 메커니즘을 용이하게 하면서 프리-러닝 클록없이 슬레이브 디바이스의 레지스터들에 데이터가 기입될 수 있게 하는 메커니즘을 제공하는 방법들, 장치, 및 컴퓨터 프로그램 제품들이 설명된다. 수신기 디바이스는 공유 버스 상으로 복수의 심볼들을 수신하고, 복수의 심볼들의 심볼 대 심볼 트랜지션들에서 임베딩된 수신 클록 신호를 추출하고, 복수의 심볼들을 트랜지션 수로 변환하고, 트랜지션 수를 데이터 비트들로 변환하며, 오직 수신 클록 신호만을 사용하여 데이터 비트들의 적어도 부분을 하나 이상의 레지스터들에 저장할 수도 있다. 수신기 디바이스는 클록 신호의 제 1 사이클의 검출 시 다운 카운터를 시작하고, 다운 카운터가 미리정의된 값에 도달할 경우에 마커를 트리거링하며, 데이터 비트들의 적어도 부분을 레지스터들에 저장하기 위해 마커를 사용할 수도 있다.

대 표 도

700



(52) CPC특허분류
G06F 13/4295 (2013.01)

명세서

청구범위

청구항 1

수신기 디바이스에 의해 수행된 방법으로서,
 공유 버스 상으로 복수의 심볼들을 수신하는 단계;
 상기 복수의 심볼들의 심볼 대 심볼 트랜지션들에서 임베딩된 클록 신호를 추출하는 단계;
 상기 복수의 심볼들을 트랜지션 수로 변환하는 단계;
 상기 트랜지션 수를 데이터 비트들로 변환하는 단계; 및
 오직 상기 클록 신호만을 사용하여 상기 데이터 비트들의 적어도 부분을 하나 이상의 레지스터들에 저장하는 단계를 포함하는, 수신기 디바이스에 의해 수행된 방법.

청구항 2

제 1 항에 있어서,
 상기 심볼들은, 어떠한 2개의 순차적 심볼들도 동일한 값을 갖지 않도록 매 클록 사이클마다 트랜지션하는, 수신기 디바이스에 의해 수행된 방법.

청구항 3

제 1 항에 있어서,
 상기 수신기 디바이스는 상기 공유 버스에 커플링된 임의의 다른 디바이스들에게 통지하지 않고 슬립 모드에 독립적으로 진입하는, 수신기 디바이스에 의해 수행된 방법.

청구항 4

제 1 항에 있어서,
 상기 수신기 디바이스는 로컬 프리-러닝 클록의 사용없이 상기 데이터 비트들의 적어도 부분을 수신하여 상기 하나 이상의 레지스터들에 기입하는, 수신기 디바이스에 의해 수행된 방법.

청구항 5

제 1 항에 있어서,
 상기 수신기 디바이스는 수신기가 슬립 모드에 있는 동안 상기 데이터 비트들의 적어도 부분을 수신하여 상기 하나 이상의 레지스터들에 기입하는, 수신기 디바이스에 의해 수행된 방법.

청구항 6

제 1 항에 있어서,
 상기 트랜지션 수는 3진수로서 표현되는, 수신기 디바이스에 의해 수행된 방법.

청구항 7

제 1 항에 있어서,
 상기 트랜지션 수는 12 디지트 3진수인, 수신기 디바이스에 의해 수행된 방법.

청구항 8

제 1 항에 있어서,

상기 공유 버스는 카메라 제어 인터페이스 확장형 (CCIE) 버스인, 수신기 디바이스에 의해 수행된 방법.

청구항 9

제 1 항에 있어서,

상기 데이터 비트들의 적어도 부분은,

상기 클록 신호의 제 1 사이클의 검출 시에 다운 카운터를 시작하는 것,

상기 다운 카운터가 미리정의된 값에 도달할 경우에 마커를 트리거링하는 것, 및

상기 데이터 비트들의 적어도 부분을 레지스터들에 저장하기 위해 상기 마커를 사용하는 것

에 의해 상기 하나 이상의 레지스터들에 기입되는, 수신기 디바이스에 의해 수행된 방법.

청구항 10

제 9 항에 있어서,

상기 미리정의된 값은, 상기 클록 신호의 최종 클록 사이클이 도달될 경우에 발생하는, 수신기 디바이스에 의해 수행된 방법.

청구항 11

제 1 항에 있어서,

상기 트랜지션 수는 상기 클록 신호의 마지막 클록 사이클과 끝에서 두번째 클록 사이클 사이에서 상기 데이터 비트들로 변환되고; 그리고

상기 데이터 비트들의 적어도 부분은 상기 클록 신호의 마지막 클록 사이클에서 레지스터들에 저장되는, 수신기 디바이스에 의해 수행된 방법.

청구항 12

수신기 디바이스로서,

복수의 심볼들을 수신하기 위해 상기 수신기 디바이스를 공유 버스에 커플링하도록 적응된 버스 인터페이스;

하나 이상의 레지스터들; 및

상기 버스 인터페이스에 커플링된 수신기 회로를 포함하고,

상기 수신기 회로는,

상기 복수의 심볼들의 심볼 대 심볼 트랜지션들에서 임베딩된 클록 신호를 추출하고;

상기 복수의 심볼들을 트랜지션 수로 변환하고;

상기 트랜지션 수를 데이터 비트들로 변환하고; 그리고

오직 상기 클록 신호만을 사용하여 상기 데이터 비트들의 적어도 부분을 상기 하나 이상의 레지스터들에 저장하도록

구성되는, 수신기 디바이스.

청구항 13

제 12 항에 있어서,

상기 심볼들은, 어떠한 2개의 순차적 심볼들도 동일한 값을 갖지 않도록 매 클록 사이클마다 트랜지션하는, 수신기 디바이스.

청구항 14

제 12 항에 있어서,

상기 수신기 디바이스는 상기 공유 버스에 커플링된 임의의 다른 디바이스들에게 통지하지 않고 슬립 모드에 독립적으로 진입하도록 적응되는, 수신기 디바이스.

청구항 15

제 12 항에 있어서,

상기 수신기 디바이스는 로컬 프리-러닝 클록의 사용없이 상기 데이터 비트들의 적어도 부분을 수신하여 상기 하나 이상의 레지스터들에 기입하도록 적응되는, 수신기 디바이스.

청구항 16

제 12 항에 있어서,

상기 수신기 디바이스는 수신기가 슬립 모드에 있는 동안 상기 데이터 비트들의 적어도 부분을 수신하여 상기 하나 이상의 레지스터들에 기입하도록 적응되는, 수신기 디바이스.

청구항 17

제 12 항에 있어서,

상기 트랜지션 수는 3진수로서 표현되는, 수신기 디바이스.

청구항 18

제 12 항에 있어서,

상기 트랜지션 수는 12 디지트 3진수인, 수신기 디바이스.

청구항 19

제 12 항에 있어서,

상기 공유 버스는 카메라 제어 인터페이스 확장형 (CCIE) 버스인, 수신기 디바이스.

청구항 20

제 12 항에 있어서,

다운 카운터를 더 포함하고,

상기 수신기 디바이스는,

상기 클록 신호의 제 1 사이클의 검출 시에 상기 다운 카운터를 시작하는 것,

상기 다운 카운터가 미리정의된 값에 도달할 경우에 마커를 트리거링하는 것; 및

상기 데이터 비트들의 적어도 부분을 레지스터들에 저장하기 위해 상기 마커를 사용하는 것

에 의해 상기 데이터 비트들의 적어도 부분을 수신하여 상기 하나 이상의 레지스터들에 기입하도록 적응되는, 수신기 디바이스.

청구항 21

제 20 항에 있어서,

상기 미리정의된 값은, 상기 클록 신호의 최종 클록 사이클이 도달될 경우에 발생하는, 수신기 디바이스.

청구항 22

수신기 디바이스로서,

공유 버스 상으로 복수의 심볼들을 수신하는 수단;

상기 복수의 심볼들의 심볼 대 심볼 트랜지션들에서 임베딩된 클록 신호를 추출하는 수단;

상기 복수의 심볼들을 트랜지션 수로 변환하는 수단;

상기 트랜지션 수를 데이터 비트들로 변환하는 수단; 및

오직 상기 클록 신호만을 사용하여 상기 데이터 비트들의 적어도 부분을 레지스터들에 저장하는 수단을 포함하는, 수신기 디바이스.

청구항 23

제 22 항에 있어서,

상기 심볼들은, 어떠한 2개의 순차적 심볼들도 동일한 값을 갖지 않도록 매 클록 사이클마다 트랜지션하는, 수신기 디바이스.

청구항 24

제 22 항에 있어서,

상기 수신기 디바이스는 상기 공유 버스에 커플링된 임의의 다른 디바이스들에게 통지하지 않고 슬립 모드에 독립적으로 진입하도록 구성되는, 수신기 디바이스.

청구항 25

제 22 항에 있어서,

상기 데이터 비트들의 적어도 부분을 상기 레지스터들에 저장하는 수단은

로컬 프리-러닝 클록의 사용없이 상기 데이터 비트들의 적어도 부분을 하나 이상의 레지스터들에 기입하도록 구성되는, 수신기 디바이스.

청구항 26

제 22 항에 있어서,

상기 데이터 비트들의 적어도 부분을 상기 레지스터들에 저장하는 수단은

수신기가 슬립 모드에 있는 동안 상기 데이터 비트들의 적어도 부분을 하나 이상의 레지스터들에 기입하도록 구성되는, 수신기 디바이스.

청구항 27

제 22 항에 있어서,

상기 트랜지션 수는 12 디지트 3진수인, 수신기 디바이스.

청구항 28

제 22 항에 있어서,

상기 데이터 비트들의 적어도 부분을 상기 레지스터들에 저장하는 수단은

상기 클록 신호의 제 1 사이클의 검출 시에 다운 카운터를 시작하고,

상기 다운 카운터가 미리정의된 값에 도달할 경우에 마커를 트리거링하고, 그리고

상기 데이터 비트들의 적어도 부분을 레지스터들에 저장하기 위해 상기 마커를 사용하도록

구성되고,

상기 미리정의된 값은, 상기 클록 신호의 최종 클록 사이클이 도달될 경우에 발생하는, 수신기 디바이스.

청구항 29

제 22 항에 있어서,

상기 데이터 비트들의 적어도 부분을 상기 레지스터들에 저장하는 수단은

상기 트랜지션 수를 변환하는 것으로서, 상기 트랜지션 수는 상기 클록 신호의 마지막 클록 사이클과 끝에서 두 번째 클록 사이클 사이에서 상기 데이터 비트들로 변환되는, 상기 트랜지션 수를 변환하고; 그리고
상기 데이터 비트들의 적어도 부분을 상기 클록 신호의 마지막 클록 사이클에서 레지스터들에 저장하도록
구성되는, 수신기 디바이스.

청구항 30

명령들이 저장된 비-일시적인 머신 판독가능 저장 매체로서,
상기 명령들은, 적어도 하나의 프로세서에 의해 실행될 경우, 상기 적어도 하나의 프로세서로 하여금
공유 버스 상으로 복수의 심볼들을 수신하게 하고;
상기 복수의 심볼들의 심볼 대 심볼 트랜지션들에서 임베딩된 클록 신호를 추출하게 하고;
상기 복수의 심볼들을 트랜지션 수로 변환하게 하고;
상기 트랜지션 수를 데이터 비트들로 변환하게 하며; 그리고
오직 상기 클록 신호만을 사용하여 상기 데이터 비트들의 적어도 부분을 레지스터들에 저장하게 하는, 비-일시
적인 머신 판독가능 저장 매체.

발명의 설명

기술 분야

[0001]

관련 출원들에 대한 상호참조

[0002]

본 출원은 2014년 12월 16일자로 미국특허상표청에 출원된 미국 정규특허출원 제14/572,680호의 우선권 및 그 이익을 주장하고, 이 정규출원은 2013년 12월 18일자로 미국특허상표청에 출원된 미국 가특허출원 제61/917,895 호의 우선권 및 그 이익을 주장하며, 이를 양 출원의 전체 내용은 본 명세서에 참조로 통합된다.

[0003]

본 개시는 공유 버스 상으로의 효율적인 동작들을 가능하게 하는 것, 및 더 상세하게는, 트랜스코딩된 송신들 내에 클록을 임베딩함으로써 공유 버스 상으로의 송신 및/또는 수신을 간략화하는 것에 관한 것이다.

배경 기술

[0004]

(I²C 로드 또한 지칭되는) I²C 는 저속 주변기기들을 마더보드, 임베디드 시스템, 셀 폰, 또는 다른 전자 디바이스들에 접속하기 위해 사용되는 멀티-마스터 직렬 단일단 버스이다. I²C 버스는 7비트 어드레싱을 갖는 클록 (SCL) 및 데이터 (SDA) 라인들을 포함한다. 버스는 디바이스들에 대하여 2 개의 역할들을 갖는다: 즉, 마스터 및 슬레이브. 마스터 디바이스는 클록을 생성하고 슬레이브 디바이스들과 통신을 개시하는 노드이다. 슬레이브 디바이스는 마스터 디바이스에 의해 어드레싱될 경우, 클록을 수신하고 응답하는 노드이다. I²C 버스는 임의의 수의 마스터 디바이스들이 존재할 수 있음을 의미하는 멀티-마스터 버스이다. 추가로, 마스터 및 슬레이브 역할들은 (STOP 가 전송된 후에) 메세지를 사이에서 변경될 수도 있다. I²C 는 기본 타입들을 정의하며, 그 메시지들 각각은 START 로 시작하고 STOP 로 종료한다.

[0005]

카메라 구현의 이러한 맥락에서, 단방향 송신들은 센서로부터 이미지를 캡처하고 그러한 이미지 데이터를 기저 대역 프로세서에서의 메모리로 송신하기 위해 사용되는 한편, 제어 데이터는 기저대역 프로세서와 센서뿐만 아니라 다른 주변기기 디바이스들 간에 교환될 수도 있다. 일 예에 있어서, 카메라 제어 인터페이스 (CCI) 프로토콜은 기저대역 프로세서와 이미지 센서 (및/또는 하나 이상의 슬레이브 디바이스들) 간의 그러한 제어 데이터를 위해 사용될 수도 있다. 일 예에 있어서, CCI 프로토콜은 이미지 센서와 기저대역 프로세서 간에 I²C 직렬 버스 상으로 구현될 수도 있다.

[0006]

CCI 에 대한 확장, 소위 CCIe (카메라 제어 인터페이스 확장형) 는, 공유 버스 상으로의 송신을 위한 정보를 인코딩하도록 개발되었다. CCIe 는 공유 버스 상의 별도의 클록 라인을 구현하지 않는다. 대신, 송신된 트랜스코딩 정보 내에 클록을 임베딩한다. 하지만, 그러한 임베딩된 클록은 데이터의 수신을 위해 및/또는 동기화 목적으로 서빙할 수도 있으며, 이는, 그러한 데이터를 레지스터들에 저장하는 것을 허용하기에 불충분할 수도 있다.

[0007] 추가로, 슬레이브 디바이스들로 하여금 전력 절약 또는 슬립 모드로 진입하게 할 뿐만 아니라 마스터 디바이스로 하여금 데이터를 슬레이브 디바이스에 기입하게 하기 위한 메커니즘이 필요하다. 이는, 마스터 디바이스로 하여금 슬립 모드에 있는 슬레이브 디바이스들을 추적하게 함으로써 수행될 수도 있지만, 그러한 메커니즘은 원치않는 오버헤드를 부가한다.

[0008] 따라서, 슬레이브 디바이스들에 대한 효율적인 슬립 및 웨이크업 메커니즘을 용이하게 하면서 프리-러닝 (free-running) 클록없이 슬레이브 디바이스의 레지스터들에 데이터를 기입하는 것을 허용하기 위해 송신 내에 임베딩된 복원된 클록을 효율적으로 사용하는 솔루션이 필요하다.

발명의 내용

해결하려는 과제

[0009] 본 개시의 일 양태에 있어서, 슬레이브 디바이스들에 대한 효율적인 슬립 및 웨이크업 메커니즘을 용이하게 하면서 프리-러닝 클록없이 슬레이브 디바이스의 레지스터들에 데이터가 기입될 수 있게 하는 메커니즘을 제공하는 방법, 컴퓨터 프로그램 제품, 및 장치가 제공된다.

과제의 해결 수단

[0010] 특정 양태들에 있어서, 수신기 디바이스에 의해 수행된 방법은 공유 버스 상으로 복수의 심볼들을 수신하는 단계, 복수의 심볼들의 심볼 대 심볼 트랜지션들에서 임베딩된 클록 신호를 추출하는 단계, 복수의 심볼들을 트랜지션 수로 변환하는 단계, 트랜지션 수를 데이터 비트들로 변환하는 단계, 및 오직 클록 신호만을 사용하여 데이터 비트들의 적어도 부분을 하나 이상의 레지스터들에 저장하는 단계를 포함한다.

[0011] 일 양태에 있어서, 심볼들은, 어떠한 2개의 순차적 심볼들도 동일한 값을 갖지 않도록 매 클록 사이클마다 트랜지션한다.

[0012] 일 양태에 있어서, 수신기 디바이스는 공유 버스에 커플링된 임의의 다른 디바이스들에게 통지하지 않고 슬립 모드에 독립적으로 진입한다.

[0013] 일 양태에 있어서, 수신기 디바이스는 로컬 프리-러닝 클록의 사용없이 데이터 비트들의 적어도 부분을 수신하여 하나 이상의 레지스터들에 기입한다. 수신기 디바이스는 수신기가 슬립 모드에 있는 동안 데이터 비트들의 적어도 부분을 수신하여 하나 이상의 레지스터들에 기입할 수도 있다.

[0014] 일 양태에 있어서, 트랜지션 수는 3진수로서 표현될 수도 있다. 일 예에 있어서, 트랜지션 수는 12 디지트 3진수일 수도 있다.

[0015] 일 양태에 있어서, 공유 버스는 카메라 제어 인터페이스 확장형 (CCIE) 버스이다.

[0016] 일 양태에 있어서, 데이터 비트들의 적어도 부분은, 클록 신호의 제 1 사이클의 검출 시 다운 카운터를 시작하는 것, 다운 카운터가 미리정의된 값에 도달할 경우에 마커를 트리거링하는 것, 및 데이터 비트들의 적어도 부분을 레지스터들에 저장하기 위해 마커를 사용하는 것에 의해 하나 이상의 레지스터들에 기입된다. 미리정의된 값은, 클록 신호의 최종 클록 사이클이 도달될 경우에 발생할 수도 있다.

[0017] 일 양태에 있어서, 트랜지션 수는 클록 신호의 마지막 클록 사이클과 끝에서 두번째 (penultimate) 클록 사이클 사이에서 데이터 비트들로 변환되고, 데이터 비트들의 적어도 부분은 클록 신호의 마지막 클록 사이클에서 레지스터들에 저장된다.

[0018] 특정 양태들에 있어서, 수신기 디바이스는 복수의 심볼들을 수신하기 위해 수신기 디바이스를 공유 버스에 커플링시키도록 적응된 버스 인터페이스, 하나 이상의 레지스터들, 및 버스 인터페이스에 커플링된 수신기 회로를 포함한다. 수신기 회로는 복수의 심볼들의 심볼 대 심볼 트랜지션들에서 임베딩된 클록 신호를 추출하고, 복수의 심볼들을 트랜지션 수로 변환하고, 트랜지션 수를 데이터 비트들로 변환하고, 그리고 오직 클록 신호만을 사용하여 데이터 비트들의 적어도 부분을 하나 이상의 레지스터들에 저장하도록 구성될 수도 있다.

[0019] 특정 양태들에 있어서, 수신기 디바이스는 공유 버스 상으로 복수의 심볼들을 수신하는 수단, 복수의 심볼들의 심볼 대 심볼 트랜지션들에서 임베딩된 클록 신호를 추출하는 수단, 복수의 심볼들을 트랜지션 수로 변환하는 수단, 트랜지션 수를 데이터 비트들로 변환하는 수단, 및 오직 클록 신호만을 사용하여 데이터 비트들의 적어도 부분을 레지스터들에 저장하는 수단을 포함한다.

[0020]

특정 양태들에 있어서, 며신 판독가능 저장 매체는 저장된 명령들을 갖는다. 저장 매체는 일시적인 및/또는 비-일시적인 저장 매체를 포함할 수도 있다. 명령들은 적어도 하나의 프로세서에 의해 실행될 수도 있고, 명령들은 적어도 하나의 프로세서로 하여금 공유 버스 상으로 복수의 심볼들을 수신하게 하고, 복수의 심볼들의 심볼 대 심볼 트랜지션들에서 임베딩된 클록 신호를 추출하게 하고, 복수의 심볼들을 트랜지션 수로 변환하게 하고, 트랜지션 수를 데이터 비트들로 변환하게 하고, 그리고 오직 클록 신호만을 사용하여 데이터 비트들의 적어도 부분을 레지스터들에 저장하게 할 수도 있다.

도면의 간단한 설명

[0021]

다양한 특징들, 특성, 및 이점들은 도면들과 함께 취해질 경우에 하기야 기재된 상세한 설명으로부터 더 명백하게 될 수도 있으며, 도면들에 있어서 유사한 참조 부호들은 전반에 걸쳐 대응하게 식별한다.

도 1 은 기저대역 프로세서 및 이미지 센서를 갖고 그리고 이미지 데이터 버스 및 멀티-모드 제어 데이터 버스를 구현하는 디바이스를 도시한 블록 다이어그램이다.

도 2 는 데이터 심볼들 내로의 클록 정보의 임베딩, 이에 의한, 데이터 송신들에 대한 양자의 I2C 와이어들 (SDA 및 SCL 라인들) 의 사용을 허용하는 것을 도시한다.

도 3 은 임베딩된 클록 정보로 트랜스코딩된 심볼들을 생성하는 송신기에서의 데이터 트랜스코딩의 일 예를 도시한 블록 다이어그램이다.

도 4 는 송신기에서 비트들이 트랜지션 수들로 변환되고 수신기에서 트랜지션 수들이 비트들로 변환되는 일 예를 도시한다.

도 5 는 트랜지션 수들과 심볼 시퀀스 간의 변환의 일 예를 도시한다.

도 6 은 순차적 심볼들과 트랜지션 수들 간을 변환하기 위해 사용될 수도 있는 변환 함수의 일 예를 도시한다.

도 7 은 공유 버스 상으로 송신된 심볼들로부터 디코딩된 데이터를 레지스터들에 기입하도록 구성된 수신기를 도시한다.

도 8 은 도 7 에 대응하고 그리고 심볼들로부터의 데이터의 디코딩 및 심볼 트랜지션들로부터의 클록의 복원을 도시한 타이밍 다이어그램이다.

도 9 는 심볼 트랜지션들로부터의 클록의 복원에 영향을 주는 특정 조건들을 도시한다.

도 10 은, 프리-러닝 클록없이 오직 심볼 트랜지션들로부터 복원된 클록만을 사용하여 공유 버스의 송신들로부터 디코딩된 데이터를 레지스터들에 기입하도록 구성된 수신기를 도시한다.

도 11 은, 프리-러닝 클록없이 오직 심볼 트랜지션들로부터 복원된 클록만을 사용하여 레지스터 기입 동작을 수행하고 그리고 12 디지트 3진수를 비트들로 변환하도록 구성될 수도 있는 수신기의 회로들을 도시한다.

도 12 는 도 10 및 도 11 에 도시된 수신기 및 회로들의 동작을 예시한 타이밍 다이어그램이다.

도 13 은, 슬레이브 디바이스가 어웨이크해야 할 필요없이 수신된 송신물로부터 추출된 클록을 이용하는 것 및 그 송신물로부터 데이터를 기입하는 것에 의해 공유 버스로부터의 송신물을 수신하도록 구성된 CCIe 슬레이브 디바이스의 일 예를 도시한다.

도 14 는 본 명세서에 개시된 하나 이상의 양태들에 따른 클록 복원 회로의 일 예를 도시한다.

도 15 는 도 14 의 예시적인 클록 복원 회로에 의해 생성된 특정 신호들의 타이밍의 일 예를 도시한다.

도 16 은 3진수 (베이스3 수치) 를 2진수로 변환하는 일반적인 예를 도시하며, 여기서, $\{T_{11}, T_{10}, \dots, T_2, T_1, T_0\}$ 에서의 각각의 트랜지션은 심볼 트랜지션 수이다.

도 17 은 2진수를 12 디지트 3진수로 변환하기 위한 방법의 일 예를 도시한다.

도 18 은 도 17 의 제산 및 모듈 연산들의 합성가능한 구현의 일 예를 도시한다.

도 19 는 본 명세서에 개시된 특정 양태들에 따라 적응될 수도 있는 프로세싱 회로를 채용한 장치의 일 예를 도시한 블록 다이어그램이다.

도 20 은, 공유 버스 상으로의 송신물을 수신하고 그러한 송신물 내의 그러한 데이터를 오직 그 송신물로부터 복원된 클록만을 사용하여 레지스터들에 저장하기 위해 슬레이브 디바이스 상에서 동작가능한 방법을 도시한다.

도 21 은 본 명세서에 개시된 특정 양태들에 따라 적응된 프로세싱 회로를 채용한 프로세싱을 채용한 장치에 대한 하드웨어 구현의 일 예를 도시한 다이어그램이다.

발명을 실시하기 위한 구체적인 내용

[0022]

다음의 설명에서, 실시형태들의 철저한 이해를 제공하기 위해 특정 상세들이 주어진다. 하지만, 실시형태들은 이들 특정 상세없이 실시될 수도 있음이 당업자에 의해 이해될 것이다. 예를 들어, 회로들은, 그 실시형태들을 불필요한 상세로 불명료하게 하지 않기 위해 복록 다이어그램들로 도시될 수도 있다. 다른 예들에 있어서, 널리 공지된 회로들, 구조들 및 기법들은 실시형태들을 불명료하게 하지 않기 위해 상세히 나타내지 않을 수도 있다.

[0023]

개관

[0024]

제 1 특징은 데이터 비트들을 3진수로 변환하는 방식을 제공한다. 그 후, 3진수는, 그 3진수의 최상위 유효 숫자를 트랜스코더로 먼저 전송하는 것에 의해 복수의 심볼들로 변환된다. 그 후, 복수의 심볼들은 버스 상으로 송신된다. 예를 들어, 복수의 심볼들은 카메라 제어 인터페이스 확장형 (CCIE) 버스 상으로 송신될 수도 있다.

[0025]

제 2 특징은, 수신된 송신물로부터 임베딩된 클록을 추출하고 (여분의 클록 사이클을 생성하기 위해 패딩된 필러 송신들을 사용하는 것 또는 프리-러닝 클록없이) 오로지 임베딩된 클록만을 사용하여 송신물 내의 데이터를 레지스터들에 저장하도록 구성된 수신기 디바이스를 위해 제공한다. 따라서, 수신기 디바이스는, 심지어 수신기 디바이스가 슬립 모드에 있을 경우에도 (어떠한 프리-러닝 클록도 이용가능하지 않을 경우) 정보를 수신 및 저장할 수 있다.

[0026]

심볼 코딩 및 CCIE 버스 상의 송신을 간략화하기 위한 예시적인 방법

[0027]

도 1 은 기저대역 프로세서 (104) 및 이미지 센서 (106) 를 갖고 그리고 이미지 데이터 버스 (116) 및 멀티-모드 제어 데이터 버스 (108) 를 구현하는 디바이스 (102) 를 도시한 복록 다이어그램이다. 도 1 이 카메라 디바이스 내의 멀티-모드 제어 데이터 버스 (108) 를 도시하지만, 이러한 제어 데이터 버스 (108) 는 다양한 상이한 디바이스들 및/또는 시스템들에서 구현될 수도 있음이 명확해야만 한다. 이미지 데이터는 이미지 데이터 버스 (116) (예를 들어, 고속 차동 DPHY 링크) 상으로 이미지 센서 (106) 로부터 기저대역 프로세서 (104) 로 전송될 수도 있다. 일 예에 있어서, 제어 데이터 버스 (108) 는 2 개의 와이어들, 즉, 클록 라인 (SCL) 및 직렬 데이터 라인 (SDA) 을 포함하는 I2C 버스일 수도 있다. 클록 라인 (SCL) 은 I2C 버스 (제어 데이터 버스 (108)) 상으로 모든 데이터 전송들을 동기화하기 위해 사용될 수도 있다. 데이터 라인 (SDA) 및 클록 라인 (SCL) 은 I2C 버스 (제어 데이터 버스 (108)) 상에서 모든 디바이스들 (112, 114, 및 118) 에 커플링된다. 이 예에 있어서, 제어 데이터는 제어 데이터 버스 (108) 를 통해 기저대역 프로세서 (104) 와 이미지 센서 (106) 뿐만 아니라 다른 주변기기 디바이스들 (118) 간에 교환될 수도 있다. I2C 에 대한 표준 클록 (SCL) 속도는 100 KHz 까지이다. I2C 고속 모드에서의 표준 클록 (SCL) 속도는 400KHz 까지이고, I2C 고속 모드 플러스 (Fm+) 에서는 1 MHz 까지이다. I2C 버스 상으로의 이를 동작 모드들은 카메라 어플리케이션들을 위해 사용될 경우, 카메라 제어 인터페이스 (CCIE) 모드로서 지정될 수도 있다.

[0028]

일 양태에 따르면, 개선된 동작 모드 (즉, 1 MHz 초과) 가 카메라 동작을 지원하기 위해 멀티-모드 제어 데이터 버스 (108) 상으로 구현될 수도 있다. I2C 버스 상으로의 이러한 개선된 동작 모드는, 카메라 어플리케이션들을 위해 사용될 경우, 카메라 제어 인터페이스 확장 (CCIE) 모드로 지정될 수도 있다. 이 예에 있어서, 기저대역 프로세서 (104) 는 마스터 디바이스/노드 (112) 를 포함하고, 이미지 센서 (106) 는 슬레이브 디바이스/노드 (114) 를 포함하며, 마스터 디바이스/노드 (112) 와 슬레이브 디바이스/노드 (114) 양자는 제어 데이터 버스 (108) 에 커플링된 다른 레거시 I2C 디바이스들의 적절한 동작에 영향을 주지 않고, 제어 데이터 버스 (108) 상으로 카메라 제어 인터페이스 확장 (CCIE) 모드에 따라 동작할 수도 있다. 일 양태에 따르면, 제어 데이터 버스 (108) 상으로의 이러한 개선된 모드는 CCIE 디바이스들과 레거시 I2C 슬레이브 디바이스들 간에 임의의 브릿지 디바이스없이 구현될 수도 있다.

[0029]

도 2 는 어떻게 클록이 데이터 심볼들 내에 임베딩될 수 있는지, 이에 의해, 데이터 송신들에 대한 양자의 I2C 와이어들 (즉, SDA 라인 및 SCL 라인) 의 사용을 허용하는 것을 도시한다. 일 예에 있어서, 클록은 트랜지

션 클록 트랜스코딩의 이용을 통해 임베딩될 수도 있다. 트랜지션 클록 트랜스코딩은, 연속적인 심볼들 사이에 통신 링크 (206) 의 시그널링 상태에서의 트랜지션이 존재하도록 (즉, 심볼 값이 매 심볼 사이클마다 트랜지션함) 오리지널 데이터를 심볼 데이터로 트랜스코딩하는 것을 수반할 수도 있다. 즉, 물리 링크 (와이어들) (206) 상으로 송신될 데이터 (204) 는, 물리 링크 (206) 의 시그널링 상태가 송신된 심볼들 (206) 의 매 심볼 사이클마다 변하도록 트랜스코딩될 수도 있다. 결과적으로, 오리지널 클록 (202) 은 매 심볼 사이클마다 심볼 상태들의 변경 시에 임베딩된다.

[0030] 수신기는 (송신된 심볼들 (206) 에서의) 각 심볼에서의 상태 트랜지션으로부터 클록 정보 (208) 를 복원하고, 그 후, 오리지널 데이터 (210) 를 획득하기 위해 송신된 심볼들 (206) 의 트랜스코딩을 역전시킨다. 이는 I2C 버스 (도 1에서 제어 데이터 버스 (108), SDA 라인 및 SCL 라인) 의 양자의 와이어들로 하여금 데이터 정보를 전송하는데 사용되게 한다. 추가로, 심볼 레이트는 배가될 수 있는데, 왜냐하면 셋업을 갖고 클록과 데이터 신호들 간에 시간을 유지할 필요가 더 이상 없기 때문이다.

[0031] 도 3 은 클록 신호를 트랜스코딩된 심볼들 내에 임베딩하기 위해, 송신기에서 데이터 비트들의 트랜스코딩된 심볼들로의 트랜스코딩을 위한 방법의 일 예를 도시한 블록 다이어그램이다. 송신기 (300) 에서, 입력 데이터 비트들 (304) 은 멀티-디지트 3진 (베이스 3) 수로 변환되며, 여기서, 각각의 디지트는 "트랜지션 수"로서 지정될 수도 있다. 그 후, 3진수는 물리 링크 (302) 의 클록 라인 (SCL) (312) 및 데이터 라인 (SDA) (314) 상으로 송신되는 (순차적) 심볼들의 세트로 변환된다. 일 예에 있어서, 2진 데이터의 오리지널 20 비트들은 비트 대 트랜지션 수 변환기 블록 (308) 에 입력되어 12 디지트 3진수로 변환된다. 12 디지트 3진수의 각각의 디지트는 "트랜지션 수" 를 나타낸다. 2개의 연속적인 트랜지션 수들은 동일한 값을 가질 수도 있다. 각각의 트랜지션 수는 트랜지션 대 심볼 블록 (310) 에서 순차적 심볼로 변환되어, 어떤 2개의 연속적인 순차적 심볼들도 동일한 값을 갖지 않도록 한다. 심볼 값 (및 와이어들 (312, 314) 의 시그널링 상태) 에서의 트랜지션이 순차적 심볼들의 모든 쌍에서의 심볼들 사이에서 보장되기 때문에, 순차적 심볼 트랜지션은 클록 심볼을 임베딩하도록 기능할 수도 있다. 그 후, 각각의 순차적 심볼 (316) 은 2 와이어 물리 링크 (302) 상으로 전송되며, 이 2 와이어 물리 링크는 SCL 라인 (312) 과 SDA 라인 (314) 을 갖는 I2C 버스를 포함할 수도 있다.

[0032] 도 4 는, 송신기 (400) 에서의 비트들로부터 트랜지션 수들로의 변환, 그 후, 수신기 (420) 에서의 트랜지션 수들로부터 비트들로의 변환의 일 예를 도시한다. 이러한 예는 12개 트랜지션 심볼들을 사용하여 2와이어 시스템에 대한 송신을 예시한다. 송신기 (400) 는 2진 정보 (비트들) 를 "비트들 대 $12 \times T$ " 변환기 (406) 으로 공급하여 12 심볼 트랜지션 수들 (T_{11} 내지 T_0) 을 생성한다. 수신기 (420) 는 12 심볼 트랜지션 수들 (T_{11} 내지 T_0) 을 수신하고, 이는 "12 $\times T$ 대 비트들" 변환기 (408) 로 공급되어 2진 정보 (비트들) 를 취출한다. 각각의 트랜지션 ($T_0 - T_{11}$) 에 대해 r 개의 가능한 심볼 트랜지션 상태들이 존재할 경우, 12개의 트랜지션들은 r^{12} 개의 상이한 상태들을 전송할 수 있다. 2와이어 버스에 대해, $r = 2^2 - 1$ 이다. 결과적으로, 트랜지션들 ($T_0 \dots T_{11}$) 은 $(2^2 - 1)^{12}$ 개의 상이한 상태들을 가질 수 있는 데이터를 포함한다. 결과적으로, $r = 4 - 1 = 3$ 이고, 상태들의 수 = $(4 - 1)^{12} = 531441$ 이다.

[0033] 12개 심볼 트랜지션 수들을 사용한 2와이어 시스템에 대한 이 예에 있어서, 각각의 트랜지션 (r) 에 대한 가능한 심볼 트랜지션들이 $3 (= 2^2 - 1)$ 이라고 가정될 수도 있다. 그룹에서의 심볼들의 수가 12 이면, 12 디지트 3진수 (베이스 3 수치): $T_{11}, T_{10}, \dots, T_2, T_1, T_0$ (각각의 $T_i: 0, 1, 2$) 이 사용될 수도 있다. 예를 들어, $\{T_{11}, T_{10}, \dots, T_2, T_1, T_0\} = \{2, 1, 0, 0, 1, 1, 0, 1, 0, 1, 2, 1\}$ 에 대해, 3진수는

[0034] 2100_1101_0121₃ (3진수)

$$= 2 \times 3^{11} + 1 \times 3^{10} + 0 \times 3^9 + 0 \times 3^8$$

$$+ 1 \times 3^7 + 1 \times 3^6 + 0 \times 3^5 + 1 \times 3^4$$

$$+ 0 \times 3^3 + 1 \times 3^2 + 2 \times 3^1 + 1 \times 3^0$$

$$= 416356 (0x65A64) 이다.$$

[0039] 이러한 방식으로, 12 트랜지션 수들은 수치로 변환될 수도 있다. 3진수 (2100_1101_0121₃) 가 예를 들어 도 3에서 트랜지션 수로서 사용될 수도 있어서, 각각의 정수는 순차적 심볼에 매핑될 수도 있거나 그 역도 성립함을 유의한다. 2100_1101_0121₃ 을 역순으로 전송할 경우, 트랜지션 수들은 전력의 감소하는 순서로 전송되며, 즉, T₁₁ 은 3¹¹ 에 의해 송신될 디지트이어서 제 11 전력이 되는 등등이다.

[0040] 2와이어 시스템 및 12 심볼 트랜지션 수들에 대하여 도 4에 도시된 예는 n와이어 시스템 및 m 심볼 트랜지션 수들로 일반화될 수도 있다. 하나의 T 당 r개의 가능한 심볼 트랜지션 상태들이 존재하면, T₀ 내지 T_{m-1}, 즉, m개의 트랜지션들이 r^m개의 상이한 상태들을 전송할 수 있고, 즉 $r = 2^n - 1$ 이다. 결과적으로, 트랜지션들 (T₀ … T_{m-1}) 은 $(2^n - 1)^m$ 개의 상이한 상태들을 가질 수 있는 데이터를 포함한다.

[0041] 도 5는 3진수들 (트랜지션 수) (502) 과 (순차적) 심볼들 (504) 사이에서 변환하기 위한 방식의 일 예를 도시한 다이어그램 (500)이다. 또한 트랜지션 수로서 지정되는 3진수, 즉, 베이스3 수치는 3개의 가능한 디지트들 또는 상태들, 즉 0, 1, 또는 2 중 하나를 가질 수 있다. 동일한 값이 2개의 연속적인 3진수들에서 나타날 수도 있지만, 어떠한 2개의 연속적인 심볼들도 동일한 값을 갖지 않는다.

[0042] 변환 함수가 도 6에 예시적으로 개시된다. 송신 측 (TX: T 대 S)에서, 로직은 $T_{tmp} = T = 0 ? 3 : T$ 이고 $C_s = P_s + T_{tmp}$ 이다. 즉, 트랜지션 수 (T) 는 0과 비교되고, T = 0 일 경우, T_{tmp}(T임시) 는 3과 동일하게 되고, 그렇지 않으면 (T 가 0과 동일하지 않을 경우) T_{tmp} 는 T 와 동일하게 된다. 그리고, 현재 심볼 (C_s) 은 이전 심볼 (P_s) 값 플러스 T_{tmp} 가 된다. 예를 들어, 제 1 사이클 (506)에서, T 는 2이고, 따라서 T_{tmp} 또한 2이며, 1 인 P_s에 의해, 새로운 C_s 는 이제 3이다.

[0043] 제 2 사이클 (508)에서, 트랜지션 수 1은 다음 사이클에서 입력되고, 트랜지션 수는 3이 아니며, 따라서 1의 T의 값이 3인 이전 심볼의 값에 가산된다. 가산의 결과, 4는 3 보다 크기 때문에, 롤오버된 수 0이 현재 심볼이 된다.

[0044] 제 3 사이클 (510)에서, 동일한 트랜지션 수 1이 입력된다. T가 1이기 때문에 T_{tmp} 또한 1이다. 변환 로직은 이전 심볼 0에 1을 가산하여 현재 심볼 1을 생성한다.

[0045] 제 4 사이클 (512)에서, 트랜지션 수 0이 입력된다. 변환 로직은, T가 제로일 때 T_{tmp} 가 3과 동일하게 되게 한다. 따라서, 3이 이전 심볼 1에 가산되어 현재 심볼 0을 생성한다 (가산의 결과, 4는 3 보다 크기 때문에, 롤오버된 수 0이 현재 심볼이 됨).

[0046] 결과적으로, 2개의 연속적인 3진 디지트들 (502)이 동일한 수들을 가지는 경우라도, 이러한 변환은 2개의 연속적인 심볼 수들이 상이한 상태 값들을 가짐을 보장한다. 이 때문에, 심볼들 (504)의 시퀀스에서의 보장된 심볼 트랜지션은 클록 신호를 임베딩하도록 기능할 수도 있고, 이에 의해, 데이터 송신들을 위한 I2C 버스에서 클록 라인 (SCL)을 자유롭게 할 수도 있다. 수신기 측 (Rx: S 대 T)에서, 로직은 역전된다: $T_{tmp} = C_s + 4 - P_s$ 및 $T = T_{tmp} = 3 ? 0 : T_{tmp}$ 이다.

[0047] 도 3을 다시 참조하면, 수신기 (320)에서, 프로세스는 트랜스코딩된 심볼들을 다시 비트들로 변환하도록 역전되고, 그 프로세스에서, 클록 신호가 심볼 트랜지션으로부터 추출된다. 수신기 (320)는 2와이어 물리 링크 (302) 상으로 순차적 심볼들 (322)의 시퀀스를 수신하고, 이 2와이어 물리 링크는 SCL 라인 입력 (324) 및 SDA 라인 입력 (326)에 접속된 I2C 버스일 수도 있다. 수신된 순차적 심볼들 (322)은 클록 데이터 복원 (CDR) 블록 (328)으로 입력되어, 클록 타이밍을 복원하고 트랜스코딩된 심볼들 (S) 을 샘플링한다. CDR (328)은 수신된 심볼들에 있어서 심볼 대 심볼 트랜지션들로부터 클록 신호 (336)를 복원할 수도 있다. 이러한 복원된 클록 (336)은 별도의 클록에 대한 필요성없이 수신기 컴포넌트들의 동작 및 추출된 비트들의 기입을 가능케 하도록 기능할 수도 있다. 그 후, 심볼 대 트랜지션 수 변환기 블록 (330)은 각각의 심볼을 트랜지션 수로 변환하며, 이 트랜지션 수는 현재 심볼과 직전 심볼 간의 차이를 나타내는 단일 디지트 3진수로서 표현될 수도 있다. 그 후, 트랜지션 수 대 비트들 변환기 (332)는 12 디지트 3진수로부터 오리지널 데이터의 20비트들 (304') 을 복구하기 위해 12개의 트랜지션 수들을 변환한다.

[0048] 본 명세서에 도시된 이러한 기법은 제어 버스 (108) (도 1)의 링크 레이트를 I2C 표준 버스가 제공하는 링크

레이트 초과로 증가시키는데 사용될 수도 있으며, 본 명세서에서 CCIe 모드로서 지정된다. 일 예에 있어서, 제어 데이터 버스 (108)에 커플링된 마스터 노드 및/또는 슬레이브 노드는, 표준 I2C 버스를 사용하여 가능한 것보다 동일한 제어 데이터 버스에 대해 더 높은 비트 레이트들을 달성하기 위해, 클록 신호를 (도 2 및 도 3에 도시된 바와 같은) 심볼 송신들 내에 임베딩하는 송신기들 및/또는 수신기들을 구현할 수도 있다. 다른 구현들에 있어서, 상이한 수의 데이터 비트들이 3진수 (베이스3 수치 시스템) 또는 다른 수치 베이스를 갖는 수로 인코딩될 수도 있음을 유의한다.

[0049] 도 6은 순차적 심볼들과 트랜지션 수들 간의 변환을 도시한 다이어그램이다. 이러한 변환은 이전 순차적 심볼 수 (P_s)로부터 현재 순차적 심볼 (C_s)로의 각각의 트랜지션을 트랜지션 수 (T)로 매핑한다. 송신기 디바이스에서, 트랜지션 수들은 순차적 심볼들로 변환되고 있다. 상대적 변환 방식이 사용되기 때문에, 트랜지션 수들은 어떠한 2개의 연속적인 순차적 심볼들 (604)도 동일하지 않을 것임을 보장한다.

[0050] 2와이어 시스템에 대한 일 예에 있어서, 4개의 순차적 심볼들 (S_0, S_1, S_2 , 및 S_3)에 할당된 4개의 원시 심볼들이 존재한다. 4개의 순차적 심볼들에 대해, 테이블 (602)은 현재 순차적 심볼 (C_s)이 어떻게 이전 순차적 심볼 (P_s)과, 현재 트랜지션 수 (T)에 기반한 임시 트랜지션 수 (T_{tmp})에 기초하여 할당될 수 있는지를 예시한다.

[0051] 이 예에 있어서, 트랜지션 수 (C_s)는 다음에 따라 할당될 수도 있다:

$$C_s = P_s + T_{tmp}$$

[0053] 여기서, $T_{tmp} = T == 0 ? 3 : T$ 이다.

[0054] 대안적으로 서술될 때, T 가 0과 동일하면, T_{tmp} 는 3이 되고, T 가 0이 아니면, T_{tmp} 는 T 와 동일하게 된다. 그리고 일단 T_{tmp} 가 계산되면, C_s 는 P_s 플러스 T_{tmp} 로 설정된다. 더욱이, 수신기단에서, 로직은 T 를 복원하기 위해 역전되며,

$$T_{tmp} = C_s + 4 - P_s \text{ 및 } T = T_{tmp} == 3 ? 0 : T_{tmp} \text{이다.}$$

[0056] 도 16은 3진수 (베이스3 수치)를 2진수로 변환하는 일반적인 예를 나타낸 식 (1600)을 포함하며, 여기서, $\{T_{11}, T_{10}, \dots, T_2, T_1, T_0\}$ 에서의 각각의 T 는 심볼 트랜지션 수이다.

[0057] 도 17은 2진수 (비트들)를 12 디지트 3진수 (베이스3 수치)로 변환하기 위한 예시적인 방법을 나타낸 식 (1700)을 포함한다. 각각의 디지트는, 3으로의 더 높은 디지트 계산으로부터의 나머지 (모듈러 연산의 결과)를 디지트 수의 거듭제곱으로 제산하고, 소수점 수치들을 버림으로써 계산될 수 있다.

[0058] 도 18은, 임의의 상업적 합성 툴들에 의해 합성가능할 수도 있는 도 17의 계산 및 모듈 연산들의 하나의 가능한 구현의 일 예를 나타낸 수학적 표현 (1800)이다.

임베딩된 수신 클록을 사용한 데이터의 추출

[0060] 도 7은, 도 3의 수신기 (320)과 비교가능할 수도 있는 수신기 (700)의 일 예를 도시한 블록 개략 다이어그램이다. 수신기 (700)는, 도 3에서 라인들 (324 및 326)에 대응할 수도 있는 신호 와이어들 (704, 706)을 포함하는 공유 버스로부터 수신된 심볼들 (702)의 시퀀스로부터 데이터 (724)를 추출하도록 구성될 수도 있다. CDR 회로 (708)는, 심볼들 (702)의 시퀀스를 나타내는 멀티 비트 신호 (718)에서 제공된 타이밍 정보로부터 도출된 수신 클록 (716) 및 샘플링된 심볼들 (720)을 제공할 수도 있다. 수신 클록 (716)은 또한, 심볼들 (702)의 시퀀스로부터 데이터를 추출하는 디코딩 로직 (710, 712)에 타이밍 정보를 제공하도록 사용될 수도 있다.

[0061] 일부 예들에 있어서, 수신 클록 (716)은, 오직 심볼들 (702)의 시퀀스에서의 심볼들 간의 트랜지션들로부터 도출된 클록 사이클들만을 사용하여, 추출된 데이터 (724)를 레지스터들 (714)에 기입하는데 사용될 경우에 문제가 존재할 수도 있다. 수신된 송신물 (702) 내의 심볼 대 심볼 트랜지션들로부터 추출된 수신 클록 (716)은 데이터 (724)를 디코딩하고 데이터 (724)를 레지스터들 (714)에 저장하기에 충분한 클록 사이클들을 제공하지 않을 수도 있다. 추출된 비트들을 저장을 위해 레지스터들 (714)에 기입하기 위한 최종 심볼 대 심볼 트랜지션 이후, 여분의 클록 사이클이 필요할 수도 있다. 일부 예들에 있어서, 충분한 클록 사이클들을 제공하기 위해 프리-러닝 클록이 사용될 수도 있다. 프리-러닝 클록의 사용은, 슬레이브 디바이스가

송신 전에 어웨이크함을 마스터 디바이스가 보장하는 것을 필요로 함을 그려한 프리-러닝 클록의 존재가 요구할 수도 있기 때문에 바람직하지 않을 수도 있다. 도 8 과 관련하여 논의된 조건들을 포함한 특정 조건들 하에서, 추출된 클록은, 심볼들 (702) 의 시퀀스로부터 데이터를 추출하고 그 데이터를 레지스터들 (714) 에 기입하기에 불충분할 수도 있다.

[0062] 도 8 은, 심볼들 내에서 인코딩된 데이터를 추출하는 것 및 심볼 트랜지션들로부터의 클록의 복원과 연관된 타이밍을 도시한 타이밍 다이어그램 (800) 이다. 시작 표시자 (S) (802) 가 선행되고, 심볼들 (806) 의 시퀀스가 2라인 버스 (704 및 706) 를 통해 송신된다. 심볼들 (702) 의 시퀀스를 나타낼 수도 있는 SI 신호 (718) 와 심볼들 간의 대응하는 트랜지션들 (722) 간의 관계가 다이어그램 (800) 에 도시된다. 수신기 클록 (RXCLK) (716) 이 심볼 대 심볼 트랜지션 (722) 으로부터 추출된다. 초기 클록 펄스 (806) 는, "시작 조건" 으로서 또한 지정될 수도 있는 시작 표시자 (S) (802) 에 대응한다. RXCLK (716) 상의 복수의 펄스들 (C1, C2, …, C12) 은, 어떠한 2개의 동일한 순차적 심볼들도 반복하지 않기 때문에, 연속적인 심볼들 (S₁₁, S₁₀, S₉, …, S₀) (718) 사이의 트랜지션들 (T₁₁, T₁₀, T₉ … T₀) (722) 로부터 추출될 수도 있다. 복수의 펄스들은, 인코딩된 심볼들 간의 제 1 트랜지션에 대응하는 제 1 펄스 (C1) (808) 로 시작하도록 고려될 수도 있다.

[0063] 끝에서 두번째 클록 사이클 (C11) (810) 이후 그리고 인코딩된 심볼들 간의 트랜지션에 대응하는 마지막 클록 사이클 (C12) (812) 이전, 최종 또는 마지막 심볼 (S₀) (804) 이 수신되고 나머지 심볼들 (S₁₁…S₁) 과 결합되어, 20개의 원시 데이터 비트들 (816) 이 마지막 클록 사이클 (C12) (812) 이 발생한 이후에 생성될 수도 있다. 오리지널 비트들이 원시 데이터 비트들 (816) 을 획득하기 위해 디코딩될 수 있는 것은 마지막 심볼 (예를 들어, 제 12 심볼 (S₀)) 의 수신 이후에만임을 유의한다. 마지막 클록 사이클 (814) 은 원시 데이터 비트들 (816) 또는 그 부분을 레지스터들 (714) 에 저장하기 위한 것이다.

[0064] 일 예에 있어서, 수신된 심볼들의 수는 12개이다. 12개의 심볼들은 20비트의 정보를 인코딩할 수도 있다 (예를 들어, 16개의 데이터 비트들 및 4개의 제어 비트들을 포함). 다른 예들에 있어서, 상이한 수의 심볼들이 상이한 수의 비트들을 인코딩하는데 사용될 수도 있다.

[0065] 도 9 는, 직렬 버스의 2개 라인들 (704 및 706) 의 상이한 시그널링 상태들에 대응하는 상이한 복원된 클록 조건들을 도시한 도면 (900) 이다. 4개의 상이한 케이스들 (912, 914, 916, 918) 이 제시된다. 4개 케이스들 (912, 914, 916, 918) 각각에 있어서, 최종 심볼 (S₀) 은 마지막 클록 (제 12) 클록 펄스 (902) 상에서 수신된다. 4개 케이스들 (912, 914, 916, 918) 은 최종 심볼의 4개의 가능한 값들을 커버한다. 제 4 케이스 (918) 에 있어서, 최종 심볼에 대응하는 2개 라인들 (704 및 706) 의 시그널링 상태는 종료 셋업 조건 동안 2개 라인들 (704 및 706) 의 시그널링 상태와 동일하다. 이에 따라, 최종 심볼이 송신되는 심볼 주기 이후 2개 라인들 (704 및 706) 의 시그널링 상태에 대해 어떠한 트랜지션도 판측되지 않는다. 제 13 클록 펄스 (904) 는 첫번째 3개의 케이스들 (912, 914, 916) 에서 생성되고, 수신된 데이터를 레지스터들에 기입하기 위해 사용될 수도 있다. 제 13 클록 펄스는 제 4 케이스 (918) 에 부재하며, 수신기가 제 13 클록 펄스 (904) 의 가능성에 의존한다면 수신기는 수신된 데이터를 레지스터들에 신뢰성있게 기입하는 것이 방해될 수도 있다.

보장된 레지스터 기입 신호의 생성

[0066] 본 명세서에서 개시된 특정 양태들에 따라 그리고 도 10 내지 도 12 에 의해 도시된 바와 같이, 심볼들의 시퀀스로부터 디코딩된 데이터가 CClе 수신기 내의 레지스터들에 신뢰성있게 기입될 수 있게 하도록 워드 마커가 생성될 수도 있다. 도 10 은, 도 11 의 회로들 (1100, 1110, 및/또는 1130) 을 사용하여 SI 신호 (1020) 로부터의 데이터를 디코딩하도록 적응될 수도 있는 수신기 (1000) 의 일 예를 도시한 블록 개략 다이어그램이다.

도 12 에서 제공된 타이밍 다이어그램 (1200) 은 수신기 (1000) 및 회로들 (1100, 1110, 1130) 의 동작을 도시한다. SI 신호 (1020) 는, RXCLK (1016) 의 샘플링 에지가 발생하기 전에, S 대 T 디코더 (1010), T 대 비트들 디코더 (1012), 및 일부 예들에 있어서, 레지스터 데이터 디코더 (1114) 의 조합에 의해 프로세싱될 수도 있다. 이에 따라, 어드레싱된 레지스터 (1113) 는 직렬 버스로부터 수신된 신호들에 관련된 샘플링을 수행하고, 그 샘플링은 디코딩된 데이터에 대해 수행된다.

[0067] 수신기 (1000) 는, 도 3 에서 라인들 (324 및 326) 에 대응할 수도 있는 신호 와이어들 (1004, 1006) 을 포함하는 공유 버스로부터 수신된 심볼들 (1002) 의 시퀀스로부터 데이터를 추출하도록 구성될 수도 있다. 데이터는, CDR 회로 (1008) 에 의해 수신된 데이터로부터 복원된 수신 클록 (1016) 을 사용하여 레지스터 (1014) 에

기입될 수도 있다. 프리-러닝 클록은 요구되지 않는다. 수신 클록 (RXCLK) (1016) 은 또한, 심볼들 (1002) 의 시퀀스로부터 데이터를 추출하는 디코딩 로직 (1010, 1012) 에 타이밍 정보를 제공하도록 사용될 수도 있다. 이 회로에 있어서, 심볼들 대 트랜지션 수 변환기 (1010) 는 SI 신호 (1020) 를 수신 및 프로세싱 한다.

[0069] 수신기 (1000) 는, 도 9 에 도시된 제 4 케이스 (918) 에서 포함한, 수신된 데이터 (1024) 를 레지스터들 (1014) 에 기입하는 것을 일관성있게 허용하는 워드 마커를 생성하고 그리고 트랜지션 수 포지션 (DNCNT) (1123) 에 대한 인덱스를 유지하기 위한 하나 이상의 카운터들을 채용할 수도 있다. 레지스터 기입 동작은, 오로지 여분의 클록 사이클을 이용가능하게 할 목적으로 미사용/패딩 비트들을 삽입할 필요성없이 그리고 프리-러닝 클록 또는 슬레이브 디바이스 상의 다른 부가적인 클록에 대한 필요성없이, 마지막 복원된 클록 (RXCLK) (1016) 사이클에 대해 수행될 수도 있다.

[0070] 도 10 을 계속 참조하면, 도 11 에 도시된 특정 회로들 (1100, 1110, 1130) 은, 오직 3진수에 대응하는 송신으로부터 복원된 클록 (1016) 만을 사용하여 추출된 비트들의 레지스터 기입 동작을 수행하고 그리고 12 디지트 3진수를 비트들로 변환하도록 적응되거나 구성될 수도 있다. 도 3 을 다시 참조하면, 20 비트들의 오리지널 데이터 (304) 는 3진 트랜지션 수로 변환되고, 그 후, 이 트랜지션 수는 12개 순차적 심볼들 (316) 로 변환된다 (즉, 트랜스코딩됨). 트랜스코딩된 심볼들 (316) 은 버스 (302) 상으로 송신된다. 수신 디바이스 (320) (예를 들어, 슬레이브 디바이스) 는 송신물 (316) 을 수신하고, 클록 복원 및 심볼 샘플링을 수행하여 트랜스코딩된 심볼들을 다시 3진수로 변환하며, 이 3진수는 그 후 3진수를 다시 오리지널 20 비트 이진 데이터로 변환하는 회로 (332) 와 같은 하나 이상의 회로들에 공급된다.

[0071] 제 1 회로 (1100) 는 12 트랜지션 수들로부터 20개 원시 비트들 (1108) 을 추출하도록 적응될 수도 있다. 3진 가중치들 (1102) 은 멀티플렉서 (1104) 를 제어하기 위해 DNCNT (1123) 를 사용하여 선택되며, 여기서, DNCNT (1123) 는 트랜지션 수 포지션을 나타낸다. 12 트랜지션 수들은 트랜지션 수들의 시퀀스에서 (예를 들어, 도달 시간에 관련된 시퀀스에서) 그 대응하는 포지션에 기초하여 결정된 순서로 프로세싱될 수도 있으며, 이 트랜지션 수들의 시퀀스는 카운터, 레지스터, 또는 다른 인덱스 회로부에 의해 제공될 수도 있는 DNCNT (1123) 의 값에 의해 표시될 수도 있다. 3진 가중치들 (1102) 은, 20개 원시 비트들 (1108) 이 추출될 수 있도록 3진 가중치들 (1102) 을 직렬화하는데 사용되는 단일 출력 멀티플렉서 (1104) 에 대한 입력들로서 제공된다. 20개 원시 비트들 (1108) 은 16개의 데이터 비트들 및 4개의 제어 비트들을 포함할 수도 있다. 제 2 멀티플렉서 (1106) 는 $T_i \times 3^i$ 연산을 위한 승산기로서 기능하고, 도 10 의 심볼 대 3진 블록 (1010) 으로부터의 2비트 출력을 나타내는 신호 (1022) 에 의해 트리거링 및/또는 제어된다. RXCLK (1016) 에 의해 트리거링된 제 1 플립 플롭 (1125) 은, 3진수가 디코딩되거나 또는 3진 가중치들 (1102) 로부터 원시 비트들 (1108) 로 변환됨에 따라, 일시적인 비트들을 누적하는데 사용된다. 끝에서 두번째 클록 펄스 (C11) (1217) 이후에 수신되는 마지막 심볼 (S_0) (1221) (도 12 참조) 의 발생은 제 1 플립 플롭 (1125) 을 트리거링하여 제 2 멀티플렉서 (1106) 에 의해 출력된 마지막 3진 가중치 (1103) 로부터 비트들에 부가될 수집된 일시적인 비트들을 출력함을 유의한다. 결과적으로, 원시 비트들 (1108) (예를 들어, 도 10 에서의 데이터 (1024)) 은 유효한 값을 유지하고, 마지막 심볼 (S_0) (1221) 이 끝에서 두번째 클록 사이클 (C11) (1217) 이후 하지만 마지막 클록 사이클 (C12) (1219) 이전에 입력된 이후에 이용가능하다.

[0072] 제 2 회로 (1110) 는, 모든 심볼들이 수신될 경우에 워드 마커 (1122) 를 획득하도록 기능할 수도 있다. 수신기 클록 (1120) 의 시작 표시자 (1118) 를 검출할 시, 다운 카운터의 값 (DNCNT) (1123) 은, RXCLK (1016) 상의 각각의 펄스와 함께, 0xB hex 로부터 끝에서 두번째 클록 (C11) (1217) 에서 제로 (0x0 hex) 로, 그 후, 마지막 클록 (C12) (1219) 에서 0xF hex 로 감소한다. 워드 마커 (1122) 상의 펄스 (1215) 는 다운 카운터가 0x0 hex 에 도달할 경우에 트리거링된다. 워드 마커 (1122) 는 제 3 회로 (1130) 에 대한 입력으로서 기능하여 데이터 비트들을 레지스터들에 기입하는 것을 가능하게 한다. DNCNT (1123) 는 또한, 입력 "B" (첫 번째 3진 가중치 (1105)) 로 시작하여 입력 "0" (마지막 3진 가중치 (1103)) 으로 카운트 다운하는 멀티플렉서 (1104) 로부터의 입력 신호를 선택하도록 기능함을 유의한다.

[0073] 제 3 회로 (1130) 는 디코딩된 비트들을 제 2 플립 플롭 또는 레지스터들 (1113) 에 기입하도록 구성된 회로의 일 예를 도시한다. 어드레스 디코더 (1112) 는 어드레스 정보의 17 비트들을 수신하고 그것을 디코딩한다. 유사하게, 데이터 디코더 (1114) 는 20개의 원시 비트들 (1108) 을 수신하고 그것들을 디코딩하여, 예를 들어, 4개의 제어 비트들이 제거된 이후 16개의 데이터 비트들을 획득한다. 워드 마커 (1122) 의 펄스 (1215) 가 트리거링되고 어드레스가 디코딩될 경우, 데이터 디코더 (1114) 에 의해 제공되는 디코딩된 데이터는 플립

플롭들 또는 레지스터 (1113)에 저장될 수도 있다. 이러한 제 3 회로 (1130)는 워드 마커 (1122)를 효과적으로 사용하여, 마지막 클록 사이클 (C12) (1219)에서 제 2 플립 플롭 또는 레지스터들 (1113)로의 기입을 트리거링한다.

[0074] 끝에서 두번째 클록 사이클 (C11) (1217)에서, DNCNT (1123)는 0xB hex로부터 0x0 hex로 감분되었고, 워드 마커 (1122)는 로직 로우로부터 로직 하이로 (즉, 펄스 (1215)의 시작) 트랜지션한다. 마지막 클록 사이클 (C12) (1219)에서, 제 2 플립 플롭 또는 레지스터 (1113)가 인에이블되고, 디코딩된 데이터 비트들을 이제 반송하는 16비트 버스를 저장한다.

[0075] 이러한 접근법은 슬레이브 디바이스 상에서 러닝 클록없이 수신된 데이터 비트들을 플립 플롭들 또는 레지스터 (1113)에 저장하는 것을 허용한다. 결과적으로, 슬레이브 디바이스는 마스터 디바이스에게 통지하지 않고 슬립 모드에 진입할 수 있다. 즉, 어떠한 별도의 메커니즘도, 슬레이브 디바이스가 슬립 모드에 진입한 때를 마스터 디바이스가 통지받도록 요구되지 않는다 (예를 들어, "슬레이브 슬립 요청"은 슬레이브 디바이스로부터 필요하지 않음). 임베딩된 클록이 슬레이브 디바이스로 하여금 송신된 비트들을 수신하게 하고 제 3 회로 (1130)가 슬레이브 디바이스로 하여금 어웨이크되게 할 필요성없이 부가적인 클록을 생성하기 때문에, 마스터 디바이스는, 슬레이브 디바이스가 (예를 들어, 프리-러닝 클록에 대한 필요성없이) 슬립상태이거나 슬립 모드일 경우라도 데이터를 슬레이브 디바이스 레지스터에 기입할 수 있다. 일부 구현들에 있어서, 슬레이브 디바이스는 기입된 레지스터 데이터를 사용하여 그 기능의 부분 또는 모두를 조건부로 웨이크업할 수도 있다.

따라서, 마스터 디바이스는, 슬레이브 디바이스에 데이터를 전송하거나 기입하기 전에 슬레이브 디바이스가 어웨이크인지 또는 슬립하고 있는지 여부를 알아야 하는 것은 아니다. 부가적으로, 슬레이브 디바이스는 마스터 디바이스에게 통지하지 않고 슬립 모드에 독립적으로 진입할 수도 있다.

[0076] 도 12의 타이밍 다이어그램 (1200)은 심볼들 내에서 인코딩된 데이터의 수신, 심볼 트랜지션들로부터 클록의 복원 뿐만 아니라 오직 복원된 클록만을 사용하여 수신된 데이터의 레지스터들로의 기입 동작을 완료하는데 사용되는 생성된 신호들의 타이밍을 도시한다. 시작 표시자 (S) (1206)가 선행되고, 심볼들 (1020)의 시퀀스가 2라인 버스 (1004, 1006)를 통해 송신된다. 심볼들 (1020)의 시퀀스 및 심볼들 간의 대응하는 트랜지션들 (1022)이 도시된다. 수신기 클록 (1016)이 심볼 대 심볼 트랜지션들 (1022)로부터 추출된다. 초기 클록 펄스 (1211)는 I2C 사양에 기술된 시작 조건과 같은 시작 조건 또는 시작 표시자에 대응한다. 복수의 클록 펄스들 (C1, C2, …, C12)은, 어떠한 2개의 동일한 순차적 심볼들도 반복하지 않기 때문에, 연속적인 심볼들 ($S_{11}, S_{10}, S_9, \dots, S_0$) 사이의 트랜지션들 ($T_{11}, T_{10}, T_9 \dots T_0$)로부터 추출될 수도 있다.

[0077] 이러한 예에서, DNCNT (1123)는 12개의 사이클들을 카운트 다운하기 위해 사용되고, 그 사이클들 각각은 수신기 클록 (RXCLK) (1016)의 로우-대-하이 트랜지션으로부터 로우-대-하이 트랜지션으로의 사이클에 대응한다. DNCNT (1123)는 제 1 클록 사이클 (1213)이 검출된 후에, 그리고 마지막 사이클 (1219)이 검출될 때까지 감분된다. DNCNT (1123)가 0x0 hex에 도달할 경우, 펄스 (1215)는 워드 (WORD) 신호 (1122)에서 트리거링된다.

[0078] 끝에서 두번째 클록 사이클 (C11) (1217) 이후 그리고 마지막 클록 사이클 (C12) (1219) 이전, 최종 또는 마지막 심볼 (S_0) (1221)이 수신되고 나머지 심볼들 ($S_{11} \dots S_1$)과 결합되어, 원시 데이터 비트들 (1108)이 마지막 클록 사이클 (C12) (1219)이 발생할 경우에 이용가능하다. 오리지널 비트들이 원시 데이터 비트들 (1108)을 획득하기 위해 디코딩될 수 있는 것은 마지막 심볼 (예를 들어, 제 12 심볼 (S_0))의 수신 이후임을 유의한다. 그 후, 마지막 클록 사이클 (1219)은 원시 데이터 비트들 (1108) 또는 그 부분을 레지스터들 (1113)에 저장하기 위해 사용된다. 이는 오로지 임베딩된 클록 (예를 들어, 심볼 대 심볼 트랜지션들로부터 복원된 클록)만을 사용하여 그리고 수신기 (슬레이브) 디바이스에서 외부의 또는 프리-러닝 클록의 사용없이 데이터 (1222)를 수신하고, 디코딩하고, 그리고 저장하는 것을 허용한다. 이는 여분의 심볼들 또는 비트들을 패딩하거나 삽입할 필요없이 달성을 유의한다. 일 예에 있어서, 수신된 심볼들의 수는 12개이다. 12개의 심볼들은 20비트의 정보를 인코딩할 수도 있다 (예를 들어, 16개의 데이터 비트들 및 4개의 제어 비트들을 포함). 다른 예들에 있어서, 상이한 수의 심볼들이 상이한 수의 비트들을 인코딩하는데 사용될 수도 있다.

[0079] 도 13은, 슬레이브 디바이스가 어웨이크해야 할 필요없이 수신된 송신물로부터 추출된 클록을 이용하는 것 및 그 송신물로부터 데이터를 기입하는 것에 의해 공유 버스로부터의 송신물을 수신하도록 구성될 수도 있는 CCIe 슬레이브 디바이스 (1302)의 일 예를 도시한 블록 다이어그램 (1300)이다. 슬레이브 디바이스 (1302)는 공유 버스 (1304 및 1306)에 커플링된 수신기 회로 (1308) 및 송신기 회로 (1310)를 포함한다. 제어 로직

(1314) 은 수신기 회로 (1308) 및/또는 송신기 회로 (1310) 를 선택적으로 활성화/비활성화하도록 기능하여 슬레이브 디바이스가 공유 버스 (1304 및 1306) 상으로 수신하거나 송신하도록 할 수도 있다. 슬레이브 디바이스 (1302) 는 또한, 슬레이브 디바이스로부터의 송신물에 대한 정보를 캡처하거나 수집하는 센서 디바이스를 포함할 수도 있다.

[0080] 수신기 회로 (1308) 는 본 명세서에 개시된 특정 양태들에 따라 심볼 대 심볼 트랜지션들로부터 수신기 클록 (RXCLK) 을 추출할 수도 있는 클록 데이터 복원 회로 (1312) 를 포함할 수도 있다. 수신기 회로 (1308) 는 또한, 오직 수신된 데이터 송신물로부터의 추출된 클록만을 사용하고 추출된 클록의 지연들을 도입하는 것 없이, 공유 버스 상에서 수신된 데이터를 디코딩하고 추출하고, 그러한 데이터를 레지스터들 (1318) 에 저장하기 위해 제 1 회로 (1100), 제 2 회로 (1110), 및/또는 제 3 회로 (1130) (도 11) 중 하나 이상을 포함할 수도 있다. 제 1 회로 (1100), 제 2 회로 (1110), 및/또는 제 3 회로 (1130) (도 11) 는 하나의 회로 내에 통합되거나 또는 상이한 모듈들 또는 서브-시스템들 중에 분산될 수도 있음을 유의한다.

[0081] 클록 생성기 (1320) 는 슬레이브 디바이스 (1302) 내에 존재할 수도 있지만, 오직 슬레이브 디바이스로부터의 데이터의 송신 및/또는 다른 슬레이브 디바이스 동작, 예컨대 센서 디바이스들에 의한 모션 검출 또는 온도 측정만을 위해 사용된다.

[0082] 도 14 는 본 명세서에 개시된 하나 이상의 양태들에 따른 CDR 회로 (1400) 의 일 예를 도시하고, 도 15 는 CDR 회로 (1400) 에 의해 생성된 특정 신호들의 타이밍의 일 예를 도시한다. CDR 회로 (1400) 는 클록 정보가 송신된 심볼들의 시퀀스들에 임베딩되는 CCIE 송신 방식에서 사용될 수도 있다. CDR 회로 (1400) 는 CDR (328) (도 3) 또는 CDR (1312) (도 13) 로서 사용될 수도 있다. CDR 회로 (1400) 는 CCIE 2라인 버스 (324 및 326) 로부터 수신된 심볼들 (1510, 1512) 에 대한 셋업 시간을 최대화하도록 구성되는 아날로그 지연 엘리먼트들 (1408a, 1412 및 1426) 을 포함한다. CDR 회로 (1400) 는 비교기 (1404), 세트-리셋 래치 (1406), 제 1 지연 엘리먼트 (1408a) 를 포함하는 원샷 엘리먼트 (1408), 제 2 아날로그 지연 엘리먼트 (1412), 제 3 아날로그 지연 엘리먼트 (1426) 및 레벨 래치 (1410) 를 포함한다. 비교기 (1404) 는 심볼들 (1510 및 1512) 의 스트림을 포함하는 입력 신호 (SI) (1420) 를, SI 신호 (1420) 의 레벨-래치된 인스턴스인 신호 (S) (1422) 와 비교할 수도 있다. 비교기는 비교 신호 (NE) (1414) 를 출력한다. 세트-리셋 래치 (1406) 는 비교기 (1404) 로부터 NE 비교 신호 (1414) 를 수신하고, 비교 신호의 필터링된 버전 (NEFLT) (1416) 을 출력한다. 제 1 아날로그 지연 디바이스 (1408a) 는 NEFLT 신호 (1416) 의 필터링된 버전을 수신할 수도 있고, NEFLT 신호 (1416) 의 지연된 인스턴스인 신호 (NEDEL 신호) (1428) 를 출력한다. 동작시, 원샷 로직 (1408) 은 NEFLT 신호 (1416) 및 지연된 NEDEL 신호 (1428) 를 수신하고, NEFLT 신호 (1416) 에 의해 트리거링된 펄스 (1506) 를 포함하는 신호 (NE1SHOT) (1424) 를 출력한다.

[0083] 제 2 아날로그 지연 디바이스 (1412) 는 NE1SHOT 신호 (1424) 를 수신하고, IRXCLK 신호 (1418) 를 출력하며, IRXCLK 신호 (1418) 는 제 3 아날로그 지연 엘리먼트 (1426) 를 사용하여 출력 클록 신호 (1430) 를 생성하는데 사용될 수도 있다. 출력 클록 신호 (1430) 는 S 신호 (1422) 에서 래치된 심볼들을 디코딩하기 위해 사용될 수도 있다. 세트-리셋 래치 (1406) 는 IRXCLK 신호 (1418) 의 상태에 기초하여 리셋될 수도 있다. 레벨 래치 (1410) 는 SI 신호 (1420) 를 수신하고 레벨-래치된 S 신호 (1422) 를 출력하며, 여기서 레벨 래치 (1410) 는 IRXCLK 신호 (1418) 에 의해 인에이블된다.

[0084] 제 1 심볼 값 S_1 (1510) 이 수신되고 있는 경우, 이는 SI 신호 (1420) 가 그 상태를 변경하는 것을 시작하게 한다. SI 신호 (1420) 의 상태는, 인터-와이어 스큐, 신호 오버슈트, 신호 언더슈트, 크로스토크 등등에 기인하여 이전 심볼 S_0 (1502) 로부터 제 1 심볼 S_1 (1510) 로의 신호 트랜지션에서 중간의 또는 불확정의 상태가 발생할 수도 있을 가능성으로 인해, S_1 신호 (1510) 의 상태와 상이할 수도 있다. NE 신호 (1414) 는 비교기 (1404) 가 SI 신호 (1420) 와 S 신호 (1422) 간에 상이한 값을 검출할 때 하이로 트랜지션하며, 이는 세트-리셋 래치 (1406) 가 비동기로 세팅되게 한다. 이에 따라, NEFLT 신호 (1416) 는 하이로 트랜지션하고, 이러한 하이 상태는, IRXCLK (1418) 가 하이가 될 경우에 세트-리셋 래치 (1406) 가 리셋될 때까지, 유지된다. IRXCLK (1418) 는 NEFLT 신호 (1416) 의 상승에 대한 지연된 응답에서 하이 상태로 트랜지션하고, 여기서, 그 지연은 아날로그 지연 엘리먼트 (1412) 에 부분적으로 기인할 수 있다.

[0085] SI 신호 (1420) 에서의 중간 상태들은 무효한 데이터로서 간주될 수도 있고, 심볼 S_0 (1502) 의 심볼 값의 짧은 주기를 포함할 수도 있으며, 이를 중간 상태들은, 비교기 (1404) 의 출력이 짧은 시간 주기들 동안 로우 상태 쪽으로 리턴하기 때문에 NE 신호 (1414) 에 스파이크들 또는 트랜지션들 (1538) 을 야기할 수도 있다. 스파

이크들 (1538) 은 세트-리셋 래치 (1406) 에 의해 출력된 NEFLT 신호 (1416) 에 영향을 미치지 않는데, 이는 세트-리셋 래치 (1406) 가 NEFLT 신호 (1416) 를 출력하기 전에 NE 신호 (1414) 상의 스파이크들 (1538) 을 효과적으로 차단 및/또는 필터링하기 때문이다.

[0086] 원샷 회로 (1408) 는 NEFLT 신호 (1416) 의 상승 예지 이후에 NE1SHOT 신호 (1424) 에서 하이 상태를 출력한다. 원샷 회로 (1408) 는 NE1SHOT 신호 (1424) 가 로우 상태로 리턴하기 전에, 지연 P 주기 (1516) 동안 NE1SHOT 신호 (1424) 를 하이 상태로 유지한다. NE1SHOT 신호 (1424) 에서의 결과적인 펄스 (1506) 는, 아날로그 지연 S 엘리먼트 (1412) 에 의해 야기된 지연 S 주기 (1518) 이후에 IRXCLK 신호 (1418) 로 전파한다. IRXCLK 신호 (1418) 의 하이 상태는 세트-리셋 래치 (1406) 를 리셋하고, NEFLT 신호 (1416) 는 로우로 트랜지션한다. IRXCLK 신호 (1418) 의 하이 상태는 또한, 레벨 래치 (1410) 를 인에이블하고, SI 신호 (1420) 의 값이 S 신호 (1422) 로서 출력된다.

[0087] 비교기 (1404) 는 S_1 심볼 (1510) 에 대응하는 S 신호 (1422) 가 SI 신호 (1420) 의 심볼 S_1 심볼 (1510) 과 매칭하는 때를 검출하고, 비교기 (1404) 의 출력은 NE 신호 (1414) 를 로우로 구동한다. NE1SHOT 신호 (1424) 에서 펄스 (1540) 의 트레일링 에지는, 아날로그 지연 S 엘리먼트 (1412) 에 의해 야기된 지연 S 주기 (1518) 이후에 IRXCLK 신호 (1418) 로 전파한다. 새로운 심볼 S_2 (1512) 이 수신되고 있는 경우, SI 신호 (1420) 는 IRXCLK 신호 (1418) 의 트레일링 예지 이후에, 심볼 S_2 (1512) 에 대응하는 값으로의 그 트랜지션을 시작한다.

[0088] 일 예에서, 출력 클록 신호 (RXCLK) (1430) 는 제 3 아날로그 지연 엘리먼트 (1426) 에 의해 지연 R 주기 (1520) 만큼 지연된다. 출력 클록 신호 (1430) 및 S 신호 (1422) (데이터) 는 디코딩 회로들 (1100, 1110, 및/또는 1130) (도 11) 에 제공될 수도 있다. 디코딩 회로들 (1100, 1110, 및/또는 1130) (도 11) 은 출력 클록 신호 (1430) 또는 그 파생 신호를 사용하여 S 신호 (1422) 상의 심볼들을 샘플링할 수도 있다.

[0089] 도시된 예에 있어서, 다양한 지연들 (1522a-1522d) 은 다양한 회로들의 스위칭 시간들 및/또는 커넥터들에 기인하는 상승 시간들에 기인할 수도 있다. 디코딩 회로에 의한 심볼 캡처를 위해 적당한 셋업 시간들을 제공하기 위해, 심볼 사이클 주기 (t_{SYN}) 에 대한 타이밍 제약은 다음과 같이 정의될 수도 있다:

$$t_{dNE} + t_{dNEFLT} + t_{d1S} + \text{지연 } S + \text{지연 } P + \max(t_{HD}, t_{REC} - t_{dNE}) < t_{SYN}$$

[0090] 그리고 셋업 시간 (t_{SU}) 에 대한 타이밍 제약은 다음과 같을 수도 있다:

$$\text{최대 스큐 스펙} + t_{SU} < t_{dNE} + t_{d1S} + \text{지연 } S$$

[0091] 여기서:

[0092] t_{SYN} : 일 심볼 사이클 주기,

[0093] t_{SU} : IRXCLK (1418) 의 상승 (리딩) 예지로 참조되는, 레벨 래치들 (1410) 에 대한 SI (1420) 의 셋업 시간,

[0094] t_{HD} : IRXCLK (1418) 의 하강 (트레일링) 예지로 참조되는, 레벨 래치들 (1410) 에 대한 SI (1420) 의 유지 시간,

[0095] t_{dNE} : 비교기 (1404) 의 전파 지연,

[0096] t_{dRST} : IRXCLK (1418) 의 상승 (리딩) 예지로부터 세트-리셋 래치 (1406) 의 리셋 시간.

[0097] CDR 회로 (1400) 는, 수신기 디바이스 (예를 들어, 슬레이브 디바이스 (1302)) 가 프리-러닝 시스템 클록을 사용하지 않고 CCIE 인코딩된 심볼들을 디코딩하고 결과적인 비트들을 레지스터들에 저장할 수도 있음을 보장하기 위해, 아날로그 지연 회로들 (1408a, 1412 및 1426) 을 채용한다. 이에 따라, CCIE 슬레이브 디바이스 (1302) (도 13 참조) 는 CCIE 판독 커맨드에 응답할 때 시스템 클록으로서 송신 클록 (1320) 을 사용하도록 적응될 수도 있으며, 데이터를 수신할 때 또는 슬레이브 디바이스가 슬립상태일 때 CDR 생성된 클록 (1430) 이 사용될 수도 있다.

[0098] 도 19 는 본 명세서에 개시된 하나 이상의 기능들을 수행하도록 구성될 수도 있는 프로세싱 회로 (1902) 를 채용하는 장치에 대한 하드웨어 구현의 간략화된 예를 도시한 개념 다이어그램 (1900) 이다. 본 개시의 다양

한 양태들에 따르면, 본 명세서에 개시된 바와 같은 엘리먼트, 또는 엘리먼트의 임의의 부분, 또는 엘리먼트들의 임의의 조합은 프로세싱 회로 (1902)를 사용하여 구현될 수도 있다. 프로세싱 회로 (1902)는 하드웨어 및 소프트웨어 모듈들의 일부 조합에 의해 제어되는 하나 이상의 프로세서들 (1904)을 포함할 수도 있다. 프로세서들 (1904)의 예들은 마이크로프로세서들, 마이크로제어기들, 디지털 신호 프로세서들 (DSP들), 필드 프로그래밍가능 게이트 어레이들 (FPGA들), 프로그래밍가능 로직 디바이스들 (PLD들), 상태 면신들, 시퀀서들, 게이트형 로직, 이산 하드웨어 회로들, 및 본 개시 전반에 걸쳐 설명된 다양한 기능을 수행하도록 구성된 다른 적합한 하드웨어를 포함한다. 하나 이상의 프로세서들 (1904)은, 특정 기능들을 수행하고 그리고 소프트웨어 모듈들 (1916) 중 하나에 의해 구성, 증강, 또는 제어될 수도 있는 특수화된 프로세서들을 포함할 수도 있다. 하나 이상의 프로세서들 (1904)은 초기화동안 로딩되는 소프트웨어 모듈들 (1916)의 조합을 통해 구성되고, 동작 동안 하나 이상의 소프트웨어 모듈들 (1916)을 로딩 또는 언로딩함으로써 추가로 구성될 수도 있다.

[0101] 도시된 예에 있어서, 프로세싱 회로 (1902)는, 버스 (1910)에 의해 일반적으로 표현되는 버스 아키텍처로 구현될 수도 있다. 버스 (1910)는 프로세싱 회로 (1902)의 특정 애플리케이션 및 전체 설계 계약들에 의존하여 임의의 수의 상호접속 버스들 및 브리지들을 포함할 수도 있다. 버스 (1910)는 하나 이상의 프로세서들 (1904) 및 저장부 (1906)를 포함하는 다양한 회로들을 함께 링크시킨다. 저장부 (1906)는 메모리 디바이스들 및 대용량 저장 디바이스들을 포함할 수도 있고, 본 명세서에서 컴퓨터 관독가능 매체 및/또는 프로세서 관독가능 매체로서 지칭될 수도 있다. 버스 (1910)는 또한 타이밍 소스들, 타이머들, 주변기기들, 전압 레귤레이터들 및 전력 관리 회로들과 같은 다양한 다른 회로들을 링크시킬 수도 있다. 버스 인터페이스 (1908)는 버스 (1910)와 하나 이상의 트랜시버들 (1912) 간에 인터페이스를 제공할 수도 있다. 트랜시버 (1912)는 프로세싱 회로에 의해 지원되는 각각의 네트워킹 기술을 위해 제공될 수도 있다. 일부 예들에 있어서, 다수의 네트워킹 기술들은 트랜시버 (1912)에서 발견된 회로부 또는 프로세싱 모듈들 중 일부 또는 전부를 공유할 수도 있다. 각각의 트랜시버 (1912)는 송신 매체 상으로 다양한 다른 장치와 통신하는 수단을 제공한다. 장치들의 본성에 의존하여, 사용자 인터페이스 (1918) (예컨대, 키패드, 디스플레이, 스피커, 마이크로폰, 조이스틱)가 또한 제공될 수도 있고, 직접 또는 버스 인터페이스 (1908)를 통해 버스 (1910)에 통신가능하게 커플링될 수도 있다.

[0102] 프로세서 (1904)는 버스 (1910)를 관리하는 것 및 저장부 (1906)를 포함할 수도 있는 컴퓨터 관독가능 매체에 저장된 소프트웨어의 실행을 포함할 수도 있는 일반적인 프로세싱을 담당할 수도 있다. 이와 관련하여, 프로세서 (1904)를 포함하는 프로세싱 회로 (1902)가 본 명세서에 개시된 방법들, 기능들 및 기법들 중 임의의 것을 구현하는데 사용될 수도 있다. 저장부 (1906)는, 소프트웨어를 실행할 경우, 프로세서 (1904)에 의해 조작되는 데이터를 저장하기 위해 사용될 수도 있고, 소프트웨어는 본 명세서에 개시된 방법들 중 임의의 방법을 구현하도록 구성될 수도 있다.

[0103] 프로세싱 회로 (1902)에서의 하나 이상의 프로세서들 (1904)은 소프트웨어를 실행할 수도 있다. 소프트웨어는, 소프트웨어, 펌웨어, 미들웨어, 마이크로코드, 하드웨어 디스크립션 언어, 또는 기타 등등으로서 지칭되든 아니든, 명령들, 명령 세트들, 코드, 코드 세그먼트들, 프로그램 코드, 프로그램들, 서브프로그램들, 소프트웨어 모듈들, 어플리케이션들, 소프트웨어 어플리케이션들, 소프트웨어 패키지들, 루틴들, 서브루틴들, 오브젝트들, 실행가능물들, 실행 스크립트들, 절차들, 함수들, 알고리즘들 등을 의미하도록 넓게 해석될 것이다. 소프트웨어는 저장부 (1906)에 또는 외부 컴퓨터 관독가능 매체에 컴퓨터 관독가능한 형태로 상주할 수도 있다.

외부 컴퓨터 관독가능 매체 및/또는 저장부 (1906)는 비-일시적인 컴퓨터 관독가능 매체를 포함할 수도 있다. 비-일시적인 컴퓨터 관독가능 매체는, 예로서, 자기 저장 디바이스 (예를 들어, 하드 디스크, 플로피 디스크, 자기 스트립), 광학 디스크 (예를 들어, 컴팩트 디스크 (CD) 또는 디지털 다기능 디스크 (DVD)), 스마트 카드, 플래시 메모리 디바이스 (예를 들어, "플래시 드라이브", 카드, 스틱, 또는 키 드라이브), 랜덤 액세스 메모리 (RAM), 관독 전용 메모리 (ROM), 프로그래밍가능 ROM (PROM), 소거가능 PROM (EPROM), 전기적으로 소거가능 PROM (EEPROM), 레지스터, 착탈가능 디스크, 및 컴퓨터에 의해 액세스 및 관독될 수도 있는 소프트웨어 및/또는 명령들을 저장하기 위한 임의의 다른 적합한 매체를 포함한다. 컴퓨터 관독가능 매체 및/또는 저장부 (1906)는 또한, 예로서, 반송파, 송신선, 및 컴퓨터에 의해 액세스 및 관독될 수도 있는 소프트웨어 및/또는 명령들을 송신하기 위한 임의의 다른 적합한 매체를 포함할 수도 있다. 컴퓨터 관독가능 매체 및/또는 저장부 (1906)는 프로세싱 회로 (1902)에, 프로세서 (1904)에, 프로세싱 회로 (1902) 외부에 상주할 수도 있거나, 또는 프로세싱 회로 (1902)를 포함하는 다중의 엔터티들에 걸쳐 분산될 수도 있다. 컴퓨터 관독가능 매체 및/또는 저장부 (1906)는 컴퓨터 프로그램 제품에서 구현될 수도 있다. 예로서, 컴퓨터 프로그램 제품은 패키징 재료들에 컴퓨터 관독가능 매체를 포함할 수도 있다. 당업자는 전체 시스템에 부과된 전체

설계 제약들 및 특정 어플리케이션에 의존하여 본 개시 전반에 걸쳐 제시되는 설명된 기능을 최상으로 구현할 수 있는 방법을 인식할 것이다.

[0104] 저장부 (1906) 는, 본 명세서에서 소프트웨어 모듈들 (1916) 로서 지정될 수도 있는 로딩가능한 코드 세그먼트들, 모듈들, 어플리케이션들, 프로그램들 등에서 보유되고 및/또는 조직되는 소프트웨어를 보유할 수도 있다.

소프트웨어 모듈들 (1916) 의 각각은, 프로세싱 회로 (1902) 상에 설치되거나 로딩되고 하나 이상의 프로세서들 (1904) 에 의해 실행될 경우, 하나 이상의 프로세서들 (1904) 의 동작을 제어하는 런타임 이미지 (1914) 에 기여하는 명령들 및 데이터를 포함할 수도 있다. 실행될 경우, 특정 명령들은 프로세싱 회로 (1902) 가 본 명세서에서 설명된 특정 방법들, 알고리즘들 및 프로세스들에 따라 기능들을 수행하게 할 수도 있다.

[0105] 소프트웨어 모듈들 (1916) 의 일부는 프로세싱 회로 (1902) 의 초기화 동안 로딩될 수도 있고, 이를 소프트웨어 모듈들 (1916) 은 본 명세서에 개시된 다양한 기능들의 수행을 가능하게 하도록 프로세싱 회로 (1902) 를 구성할 수도 있다. 예를 들어, 일부 소프트웨어 모듈들 (1916) 은 프로세서 (1904) 의 내부 디바이스들 및/또는 로직 회로들 (1922) 을 구성할 수도 있고, 트랜시버 (1912), 버스 인터페이스 (1908), 사용자 인터페이스 (1918), 타이머들, 수학적 코프로세서들 등과 같은 외부 디바이스들로의 액세스를 관리할 수도 있다. 소프트웨어 모듈들 (1916) 은, 인터럽트 핸들러들 및 디바이스 드라이버들과 상호작용하고 그리고 프로세싱 회로 (1902) 에 의해 제공된 다양한 리소스들로의 액세스를 제어하는 제어 프로그램 및/또는 오퍼레이팅 시스템을 포함할 수도 있다. 리소스들은 메모리, 프로세싱 시간, 트랜시버 (1912) 로의 액세스, 사용자 인터페이스 (1918) 등을 포함할 수도 있다.

[0106] 프로세싱 회로 (1902) 의 하나 이상의 프로세서들 (1904) 은 다기능적일 수도 있고, 이에 의해, 소프트웨어 모듈들 (1916) 의 일부는 상이한 기능들 또는 동일한 기능의 상이한 인스턴스들을 수행하도록 로딩 및 구성된다.

하나 이상의 프로세서들 (1904) 은 추가로, 예를 들어, 사용자 인터페이스 (1918), 트랜시버 (1912) 및 디바이스 드라이버들로부터의 입력들에 응답하여 개시된 백그라운드 작업들을 관리하도록 적응될 수도 있다. 다수의 기능들의 수행을 지원하기 위해, 하나 이상의 프로세서들 (1904) 은 멀티태스킹 환경을 제공하도록 구성될 수도 있고, 이에 의해, 복수의 기능들의 각각은, 필요하거나 요구되는 바에 따라, 하나 이상의 프로세서들 (1904) 에 의해 서비스되는 작업들의 세트로서 구현된다. 일 예에 있어서, 멀티태스킹 환경은 상이한 작업들 사이에서 프로세서 (1904) 의 제어를 전달하는 시간공유 프로그램 (1920) 을 사용하여 구현될 수도 있고, 이에 의해, 각각의 작업은, 임의의 현저한 동작들의 완료시 및/또는 인터럽트와 같은 입력에 응답하여, 하나 이상의 프로세서들 (1904) 의 제어를 시간공유 프로그램 (1920) 으로 리턴한다. 작업이 하나 이상의 프로세서들 (1904) 의 제어를 가질 경우, 프로세싱 회로는 제어 작업과 연관된 기능에 의해 어드레싱되는 목적들을 위해 효과적으로 특수화된다. 시간공유 프로그램 (1920) 은 오퍼레이팅 시스템, 라운드-로빈 기반의 제어를 전송하는 메인 루프, 기능들의 우선순위화에 따라 하나 이상의 프로세서들 (1904) 의 제어를 할당하는 기능, 및/또는 하나 이상의 프로세서들 (1904) 의 제어를 핸들링 기능에 제공함으로써 외부 이벤트들에 응답하는 인터럽트 구동된 메인 루프를 포함할 수도 있다.

[0107] 도 20 은, 공유 버스 상으로의 송신물을 수신하고 그러한 송신물 내의 그러한 데이터를 오직 그 송신물로부터 복원된 클록만을 사용하여 레지스터들에 저장하기 위해 슬레이브 디바이스 상에서 동작가능한 방법의 플로우 차트 (2000) 이다. 예를 들어, 그 방법은 도 13 의 수신기 디바이스에 의해 구현될 수도 있다.

[0108] 블록 2002 에서, 복수의 심볼들이 공유 버스 상으로 수신될 수도 있다. 공유 버스는 CCIe 일 수도 있다. 심볼들은, 어떠한 2개의 순차적 심볼들도 동일한 값을 갖지 않도록 매 클록 사이클마다 트랜지션할 수도 있다.

[0109] 블록 2004 에서, 복수의 심볼들의 심볼 대 심볼 트랜지션들에서 임베딩된 클록 신호가 추출된다.

[0110] 블록 2006 에서, 복수의 심볼들이 트랜지션 수로 변환될 수도 있다. 트랜지션 수는 12 디지트 3진수일 수도 있다.

[0111] 블록 2008 에서, 트랜지션 수가 데이터 비트들로 변환될 수도 있다. 트랜지션 수는 클록 신호의 마지막 클록 사이클과 끝에서 두번째 클록 사이클 사이에서 데이터 비트들로 변환될 수도 있다.

[0112] 블록 2010 에서, 데이터 비트들의 적어도 부분은 오직 클록 신호만을 사용하여 하나 이상의 레지스터들에 저장될 수도 있다. 수신기 디바이스는 로컬 프리-러닝 클록의 사용없이 데이터 비트들의 적어도 부분을 수신하여 하나 이상의 레지스터들에 기입할 수도 있다. 수신기 디바이스는 수신기가 슬립 모드에 있는 동안 데이터 비트들의 적어도 부분을 수신하여 하나 이상의 레지스터들에 기입한다. 데이터 비트들의 적어도 부분은,

클록 신호의 제 1 사이클의 검출 시 다음 카운터를 시작하는 것, 다음 카운터가 미리정의된 값에 도달할 경우에 마커를 트리거링하는 것, 및 데이터 비트들의 적어도 부분을 레지스터들에 저장하기 위해 마커를 사용하는 것에 의해 하나 이상의 레지스터들에 기입된다. 미리정의된 값은, 클록 신호의 최종 클록 사이클이 도달될 경우에 발생할 수도 있다. 데이터 비트들의 적어도 부분은 클록 신호의 마지막 클록 사이클에서 레지스터들에 저장될 수도 있다.

[0113] 일 예에 있어서, 수신기 디바이스는 공유 버스에 커플링된 임의의 다른 디바이스들에게 통지하지 않고 슬립 모드에 독립적으로 진입할 수도 있다.

[0114] 도 21 은 프로세싱 회로 (2102) 를 채용하는 장치에 대한 하드웨어 구현의 간략화된 예를 도시한 개념 다이어그램 (2100) 이다. 프로세싱 회로는 통상적으로, 마이크로 프로세서, 마이크로제어기, 디지털 신호 프로세서, 시퀀서, 및 상태 머신 중 하나 이상을 포함할 수도 있는 프로세서 (2116) 를 갖는다. 프로세싱 회로 (2102) 는 버스 (2120) 에 의해 일반적으로 표현되는 버스 아키텍처로 구현될 수도 있다. 버스 (2120) 는 프로세싱 회로 (2102) 의 특정 애플리케이션 및 전체 설계 제약들에 의존하여 임의의 수의 상호접속 버스들 및 브리지들을 포함할 수도 있다. 버스 (2120) 는 프로세서 (2116), 모듈들 또는 회로들 (2104, 2106, 2108 및 2110), 직렬 버스 (2114) 의 커넥터들 또는 와이어들 상으로 통신하도록 구성가능한 라인 인터페이스 드라이버들 (2112), 심볼/3진수 변환 회로부와 협력하는 하나 이상의 레지스터들 (2122), 및 컴퓨터 판독가능 저장 매체 (2118) 에 의해 표현된 하나 이상의 프로세서들 및/또는 하드웨어 모듈들을 포함한 다양한 회로들을 함께 링크 시킨다. 버스 (2120) 는 또한, 당업계에 널리 공지되고 따라서 어떠한 추가로 설명되지 않을 타이밍 소스들, 주변기기들, 전압 레귤레이터들, 및 전력 관리 회로들과 같은 다양한 다른 회로들을 링크시킬 수도 있다.

[0115] 프로세서 (2116) 는 컴퓨터 판독가능 저장 매체 (2118) 상에 저장된 소프트웨어의 실행을 포함한 일반 프로세싱 을 책임진다. 소프트웨어는, 프로세서 (2116) 에 의해 실행될 경우, 프로세싱 회로 (2102) 로 하여금 임의의 특정의 장치에 대해 상기에서 설명된 다양한 기능들을 수행하게 한다. 프로세서 판독가능 저장 매체 (2118) 는 또한, 데이터 레인들 및 클록 레인들로서 구성될 수도 있는 커넥터들 (2114) 상으로 송신된 심볼들로부터 디코딩된 데이터를 포함하여, 소프트웨어를 실행할 경우 프로세서 (2116) 에 의해 조작되는 데이터를 저장하기 위해 사용될 수도 있다. 프로세싱 회로 (2102) 는 모듈들 (2104, 2106, 2108, 및 2110) 중 적어도 하나를 더 포함한다. 모듈들 (2104, 2106, 2108, 및 2110) 은 컴퓨터 판독가능 저장 매체 (2118) 에 상주/저장된, 프로세서 (2116) 에서 구동하는 소프트웨어 모듈들, 프로세서 (2116) 에 커플링된 하나 이상의 하드웨어 모듈들, 또는 이들의 일부 조합일 수도 있다. 모듈들 (2104, 2106, 2108 및/또는 2110) 은 마이크로제어기 명령들, 상태 머신 구성 파라미터들, 또는 이들의 일부 조합을 포함할 수도 있다.

[0116] 일 구성에 있어서, 무선 통신을 위한 장치 (2100) 는 직렬 버스 (2114) 상으로 복수의 심볼들을 수신하도록 구성된 모듈 및/또는 회로 (2104), 복수의 심볼들의 심볼 대 심볼 트랜지션들에서 임베딩된 클록 신호를 추출하도록 구성된 모듈 및/또는 회로 (2106), 복수의 심볼들을 트랜지션 수로 변환하도록 구성된 모듈 및/또는 회로 (2108), 트랜지션 수를 데이터 비트들로 변환하도록 구성된 모듈 및/또는 회로 (2108), 및 오직 클록 신호만을 사용하여 데이터 비트들의 적어도 부분을 레지스터들에 저장하도록 구성된 모듈 및/또는 회로 (2122) 를 포함한다.

[0117] 도면들에 도시된 컴포넌트들, 단계들, 특징들 및/또는 기능들 중 하나 이상은 단일 컴포넌트, 단계, 특징 또는 기능으로 재배열 및/또는 결합되거나, 또는 수개의 컴포넌트들, 단계들, 또는 기능들로 구현될 수도 있다. 부가적인 엘리먼트들, 컴포넌트들, 단계들, 및/또는 기능들이 또한, 본 명세서에 개시된 신규한 특징들로부터 일탈함없이 부가될 수도 있다. 도면들에 도시된 장치들, 디바이스들, 및/또는 컴포넌트들은 도면들에 설명된 방법들, 특징들, 또는 단계들 중 하나 이상을 수행하도록 구성될 수도 있다. 본 명세서에서 설명된 신규한 알고리즘들은 또한, 소프트웨어에서 효율적으로 구현되고/되거나 하드웨어에 임베딩될 수도 있다.

[0118] 추가로, 실시형태들은 플로우 차트, 플로우 다이어그램, 구조 다이어그램, 또는 블록 다이어그램으로서 도시된 프로세스로서 설명될 수도 있음을 유의한다. 플로우차트가 동작들을 순차적인 프로세스로서 기술할 수도 있지만, 동작들 중 다수는 병렬로 또는 동시에 수행될 수 있다. 부가적으로, 동작들의 순서가 재배열될 수도 있다. 프로세스는 그 동작들이 완료될 경우에 종료된다. 프로세스는 방법, 함수, 절차, 서브루틴, 서브프로그램 등에 대응할 수도 있다. 프로세스가 함수에 대응할 경우, 그 종료는 그 함수의 호출 함수 또는 메인 함수로의 반환에 대응한다.

[0119] 더욱이, 저장 매체는 판독 전용 메모리 (ROM), 랜덤 액세스 메모리 (RAM), 자기 디스크 저장 매체들, 광학 저장

매체들, 플래시 메모리 디바이스들 및/또는 정보를 저장하기 위한 다른 머신 관독가능 매체들을 포함하여 데이터를 저장하기 위한 하나 이상의 디바이스들을 나타낼 수도 있다. 용어 "머신 관독가능 매체"는 휴대용 또는 고정식 저장 디바이스들, 광학 저장 디바이스들, 무선 채널들, 및 명령(들) 및/또는 데이터를 저장, 포함, 또는 수록할 수 있는 다양한 다른 매체들을 포함하지만 이에 한정되지 않는다.

[0120] 더욱이, 실시형태들은 하드웨어, 소프트웨어, 펌웨어, 미들웨어, 마이크로코드, 또는 이들의 임의의 조합에 의해 구현될 수도 있다. 소프트웨어, 펌웨어, 미들웨어 또는 마이크로코드에서 구현될 경우, 필요한 작업들을 수행하기 위한 프로그램 코드 또는 코드 세그먼트들은 저장 매체 또는 다른 저장부(들)와 같은 머신 관독가능 매체에 저장될 수도 있다. 프로세서는 필요한 작업들을 수행할 수도 있다. 코드 세그먼트는 절차, 함수, 서브프로그램, 프로그램, 루틴, 서브루틴, 모듈, 소프트웨어 패키지, 클래스, 또는 명령들, 데이터 구조들, 또는 프로그램 스테이트먼트들의 임의의 조합을 나타낼 수도 있다. 코드 세그먼트는, 정보, 데이터, 인수들 (arguments), 파라미터들, 또는 메모리 콘텐츠를 전달 및/또는 수신함으로써 다른 코드 세그먼트 또는 하드웨어 회로에 커플링될 수도 있다. 정보, 인수들, 파라미터들, 데이터 등은 메모리 공유, 메시지 전달, 토큰 전달, 네트워크 송신 등을 포함하는 임의의 적합한 수단을 통해 전달되거나, 포워딩되거나, 또는 전송될 수도 있다.

[0121] 본 명세서에 개시된 예들과 관련하여 설명된 다양한 예시적인 논리 블록들, 모듈들, 회로들, 엘리먼트들, 및/또는 컴포넌트들은 범용 프로세서, 디지털 신호 프로세서 (DSP), 주문형 집적회로 (ASIC), 필드 프로그래밍가능 게이트 어레이 (FPGA) 또는 다른 프로그래밍가능 로직 컴포넌트, 이산 게이트 또는 트랜지스터 로직, 이산 하드웨어 컴포넌트들, 또는 본 명세서에서 설명된 기능들을 수행하도록 설계된 이들의 임의의 조합으로 구현 또는 수행될 수도 있다. 범용 프로세서는 마이크로프로세서일 수도 있지만, 대안적으로, 그 프로세서는 임의의 종래의 프로세서, 제어기, 마이크로 제어기, 또는 상태 머신일 수도 있다. 프로세서는 또한, 컴퓨팅 컴포넌트들의 조합, 예를 들어, DSP 와 마이크로프로세서의 조합, 다수의 마이크로프로세서들, DSP 코어와 결합된 하나 이상의 마이크로프로세서들, 또는 임의의 기타 다른 구성물로서 구현될 수도 있다.

[0122] 본 명세서에 개시된 예들과 관련하여 설명된 방법들 또는 알고리즘들은 하드웨어에서, 프로세서에 의해 실행 가능한 소프트웨어 모듈에서, 또는 이들 양자의 조합에서 프로세싱 유닛, 프로그래밍 명령들, 또는 다른 지시들의 형태로 직접 구현될 수도 있으며, 단일의 디바이스에 포함되거나 다중의 디바이스들에 걸쳐 분산될 수도 있다.

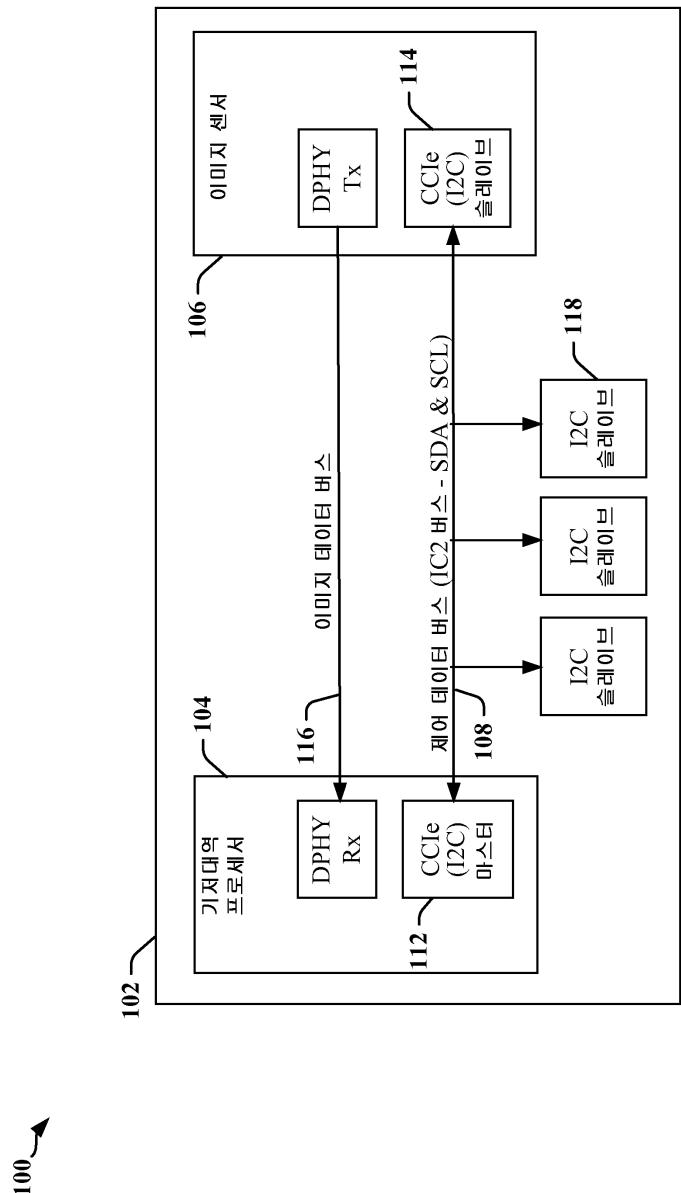
소프트웨어 모듈은 RAM 메모리, 플래시 메모리, ROM 메모리, EPROM 메모리, EEPROM 메모리, 레지스터들, 하드 디스크, 착탈가능 디스크, CD-ROM, 또는 당업계에 공지된 임의의 다른 형태의 저장 매체에 상주할 수도 있다. 저장 매체는, 프로세서가 저장 매체로부터 정보를 판독하고, 저장 매체로 정보를 기입할 수 있도록 프로세서에 커플링될 수도 있다. 대안적으로, 저장 매체는 프로세서에 통합될 수도 있다.

[0123] 당업자는 본 명세서에 개시된 실시형태들과 관련하여 설명된 다양한 예시적인 논리 블록들, 모듈들, 회로들, 및 알고리즘 단계들이 전자 하드웨어, 컴퓨터 소프트웨어, 또는 이들 양자의 조합으로서 구현될 수도 있음을 추가로 인식할 것이다. 하드웨어와 소프트웨어의 이러한 대체 가능성을 분명히 예시하기 위하여, 다양한 예시적인 컴포넌트들, 블록들, 모듈들, 회로들 및 단계들이 일반적으로 그들의 기능의 관점에서 상기 기술되었다. 그러한 기능이 하드웨어로 구현되는지 또는 소프트웨어로 구현되는지의 여부는 전체 시스템에 부과된 설계 제약들 및 특정 애플리케이션에 의존한다.

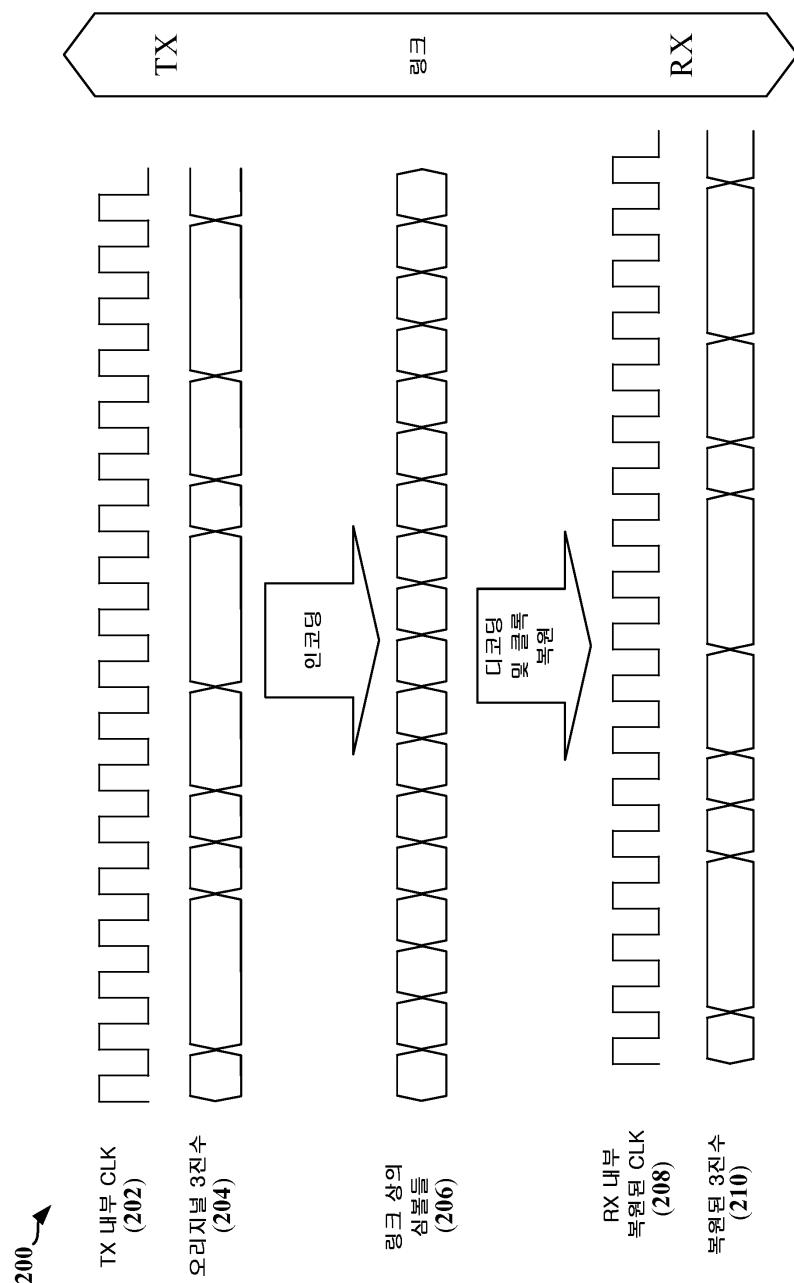
[0124] 본 명세서에서 설명된 본 발명의 다양한 특징들은 본 발명으로부터 일탈함없이 상이한 시스템들에서 구현될 수 있다. 전술한 실시형태들은 단지 예들일 뿐이고 본 발명을 제한하는 것으로 해석되지 않아야 함에 유의해야 한다. 실시형태들의 설명은 예시적인 것으로 의도되며, 청구항들의 범위를 한정하도록 의도되지 않는다. 그에 따라, 본 교시들은 다른 타입들의 장치들에 용이하게 적용될 수 있으며, 다수의 대안들, 수정들, 및 변동들은 당업자에게 명백할 것이다.

도면

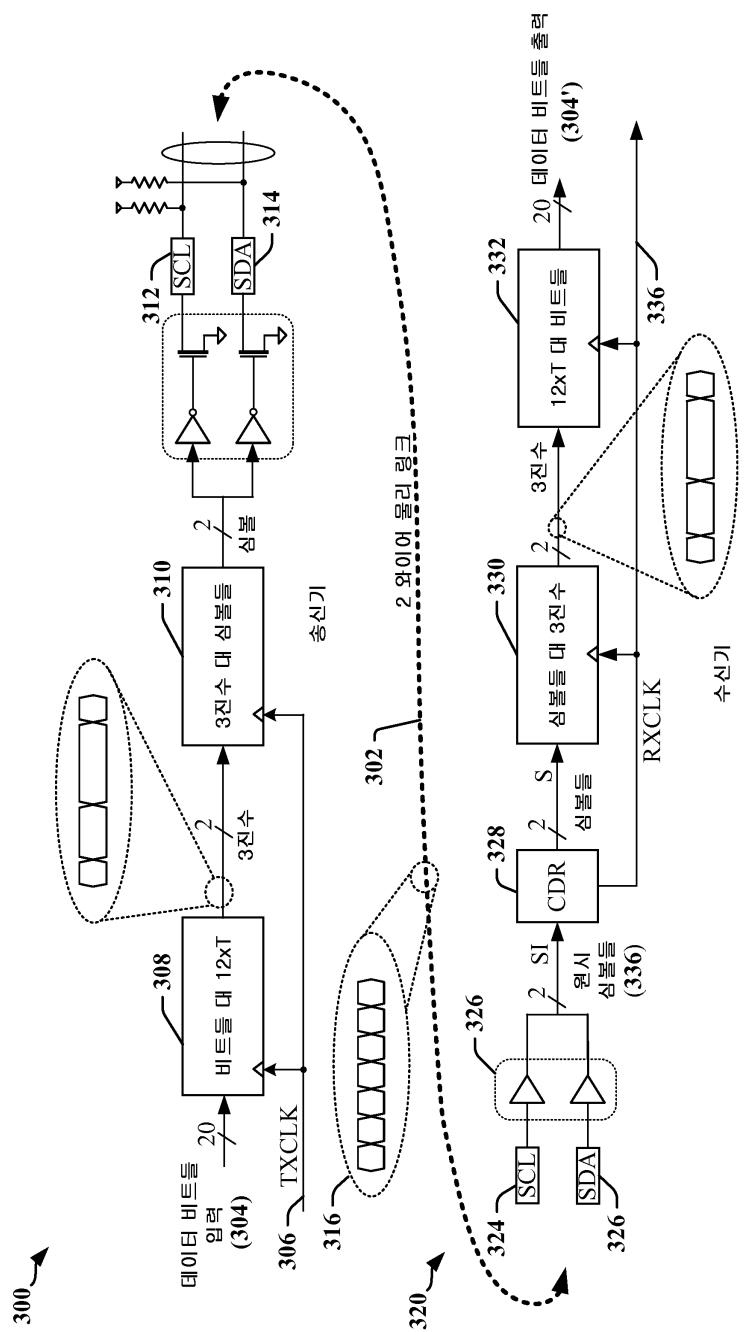
도면1



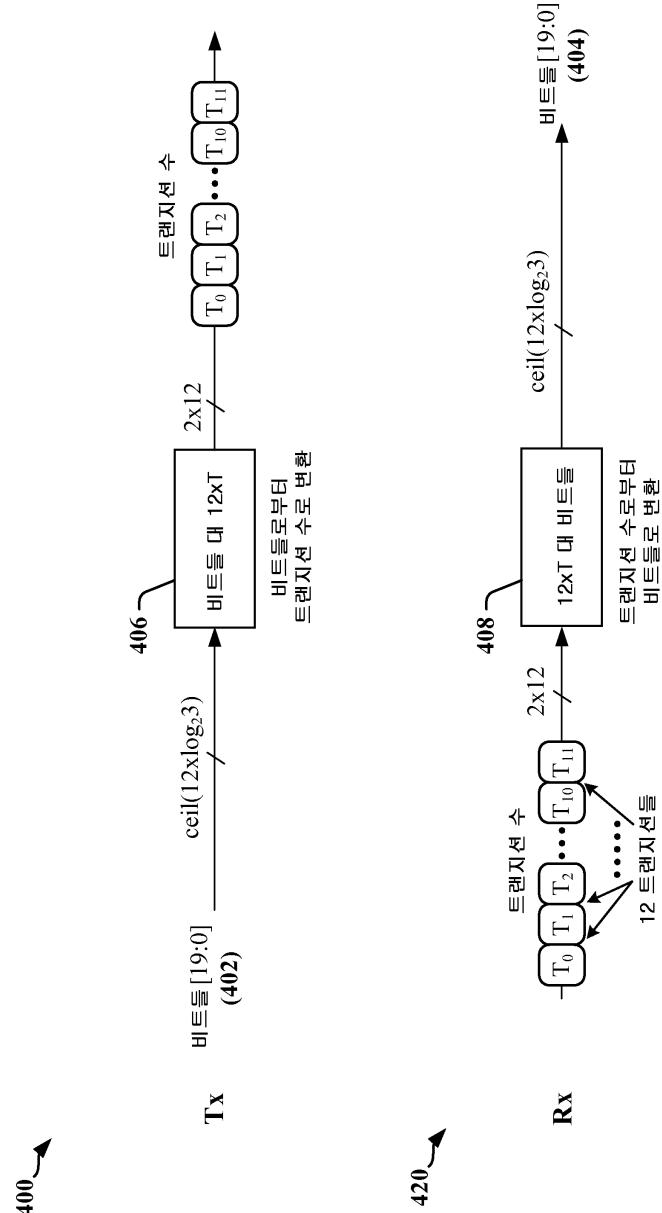
도면2



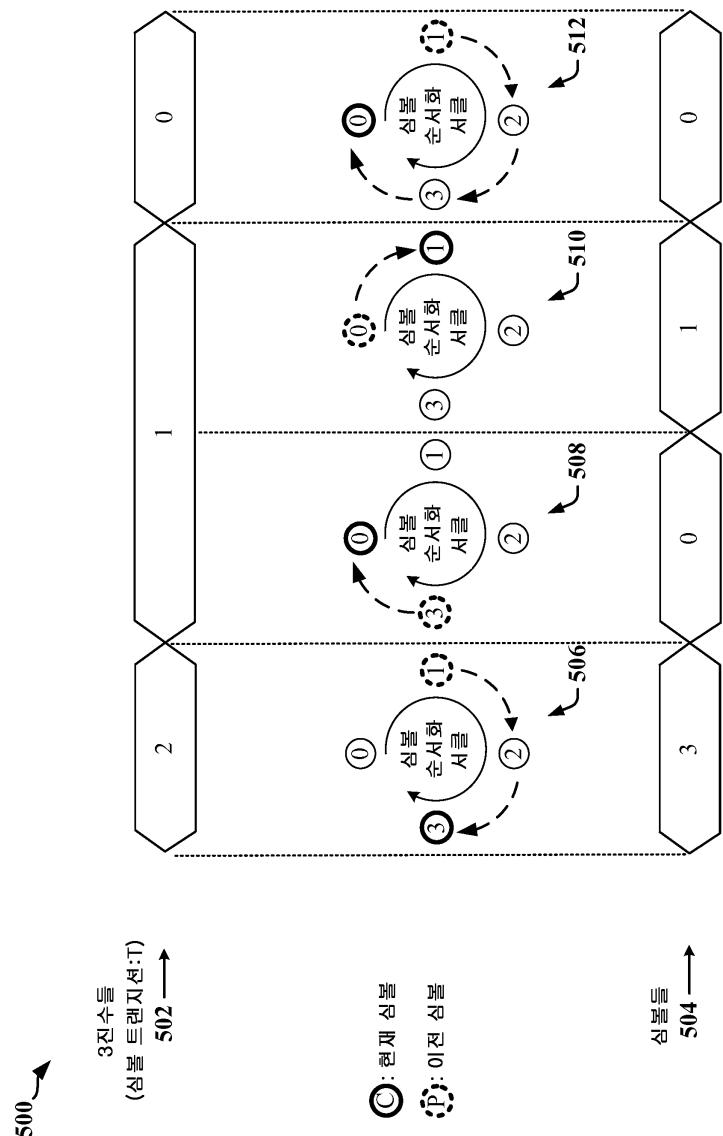
도면3



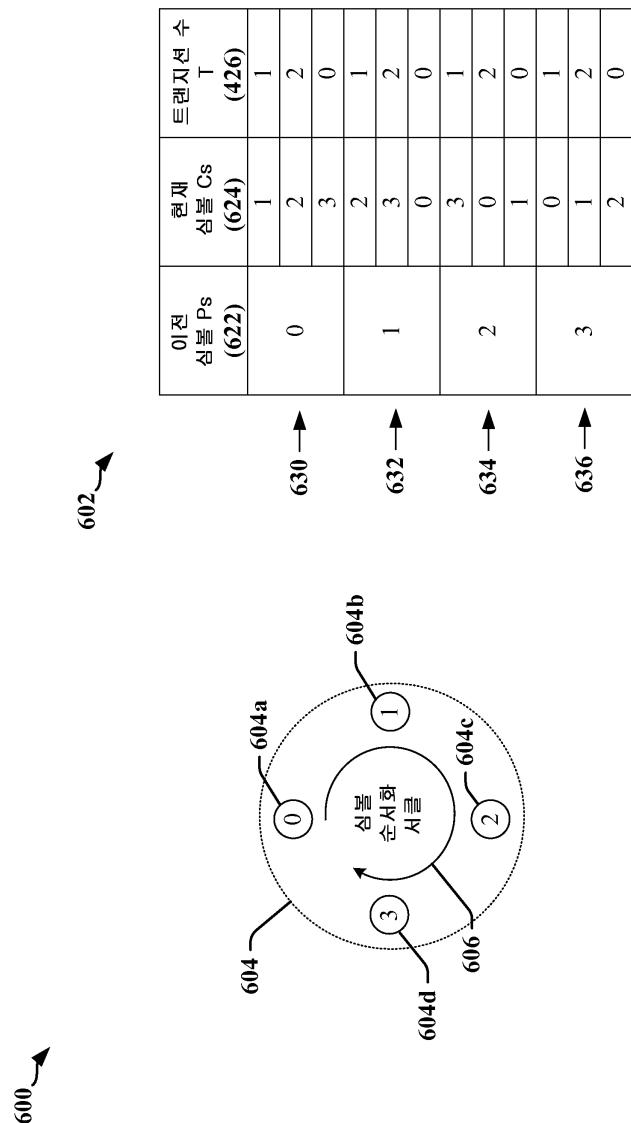
도면4



도면5



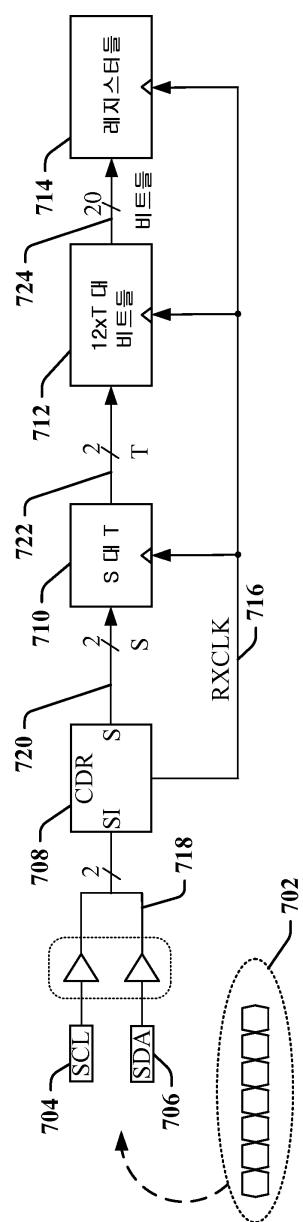
도면6



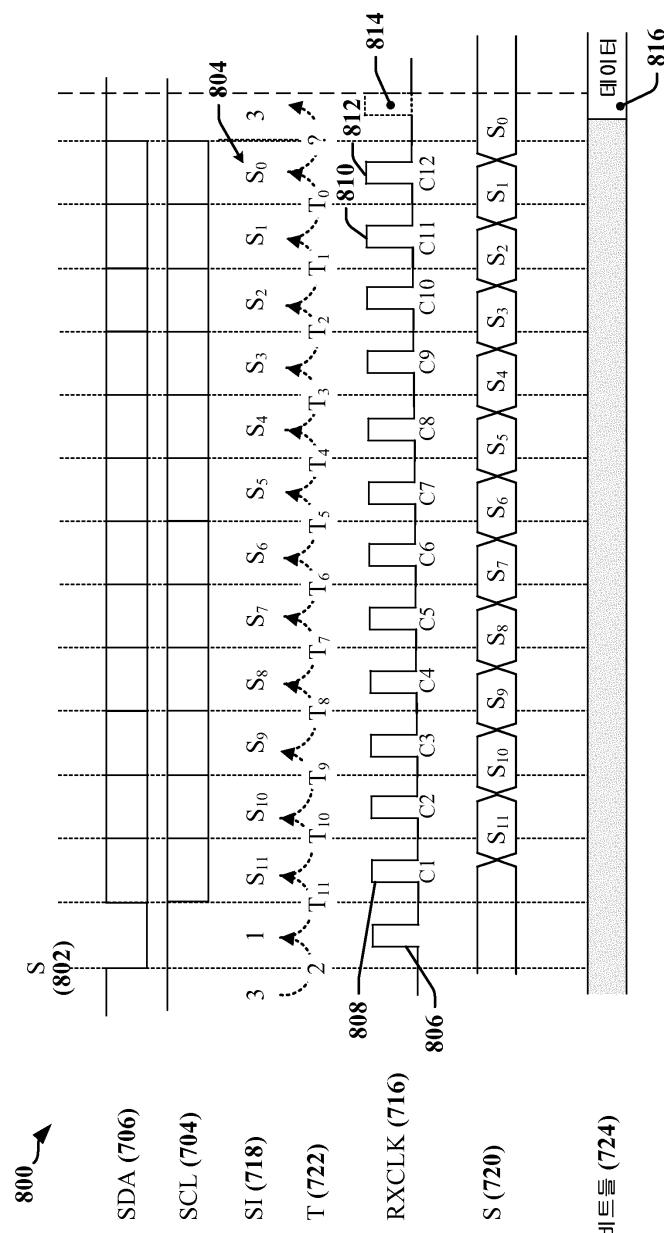
도면7

700 ↗

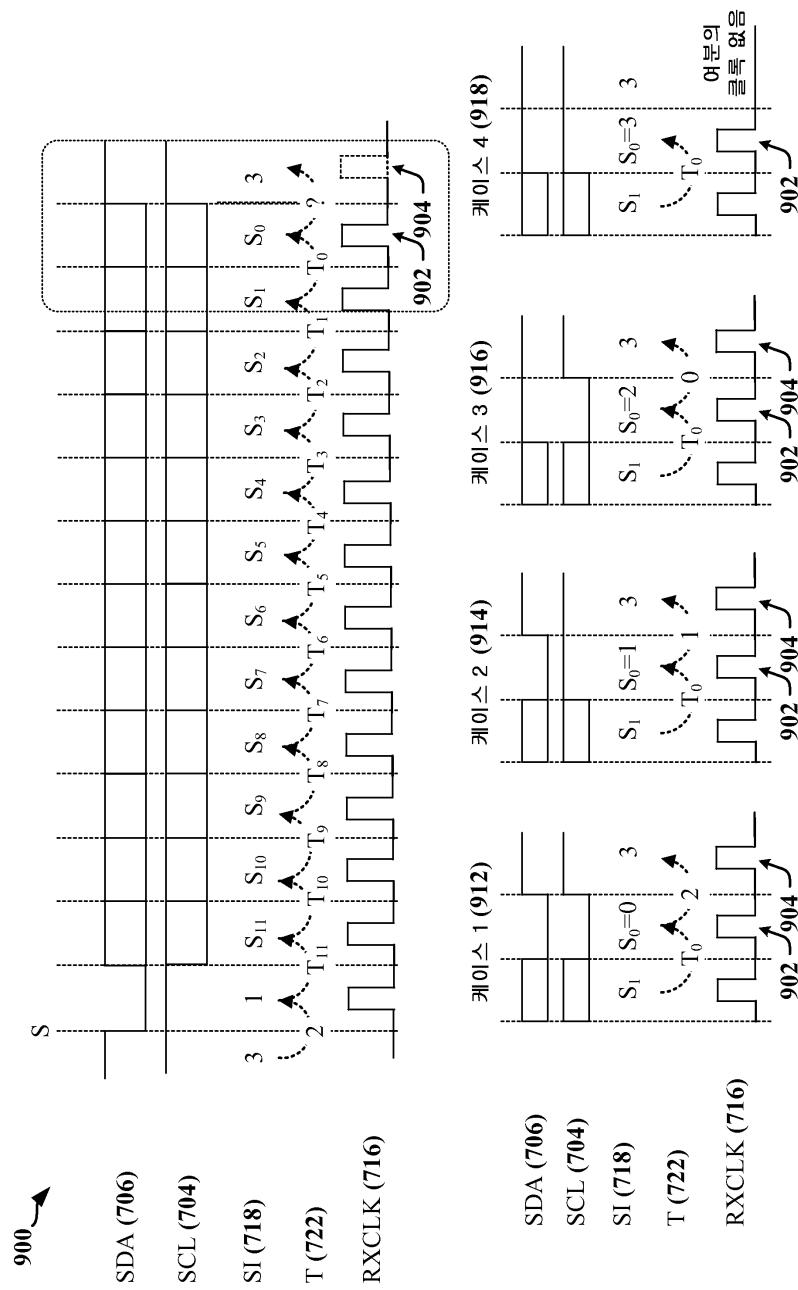
수신기



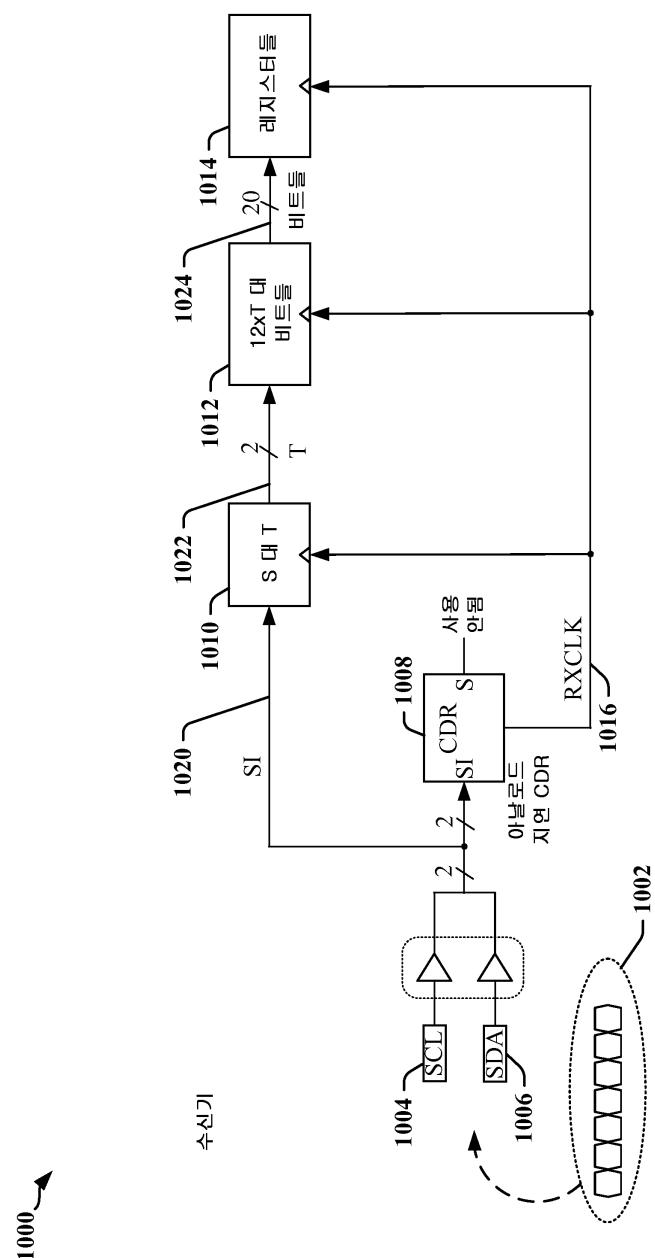
도면8



도면9

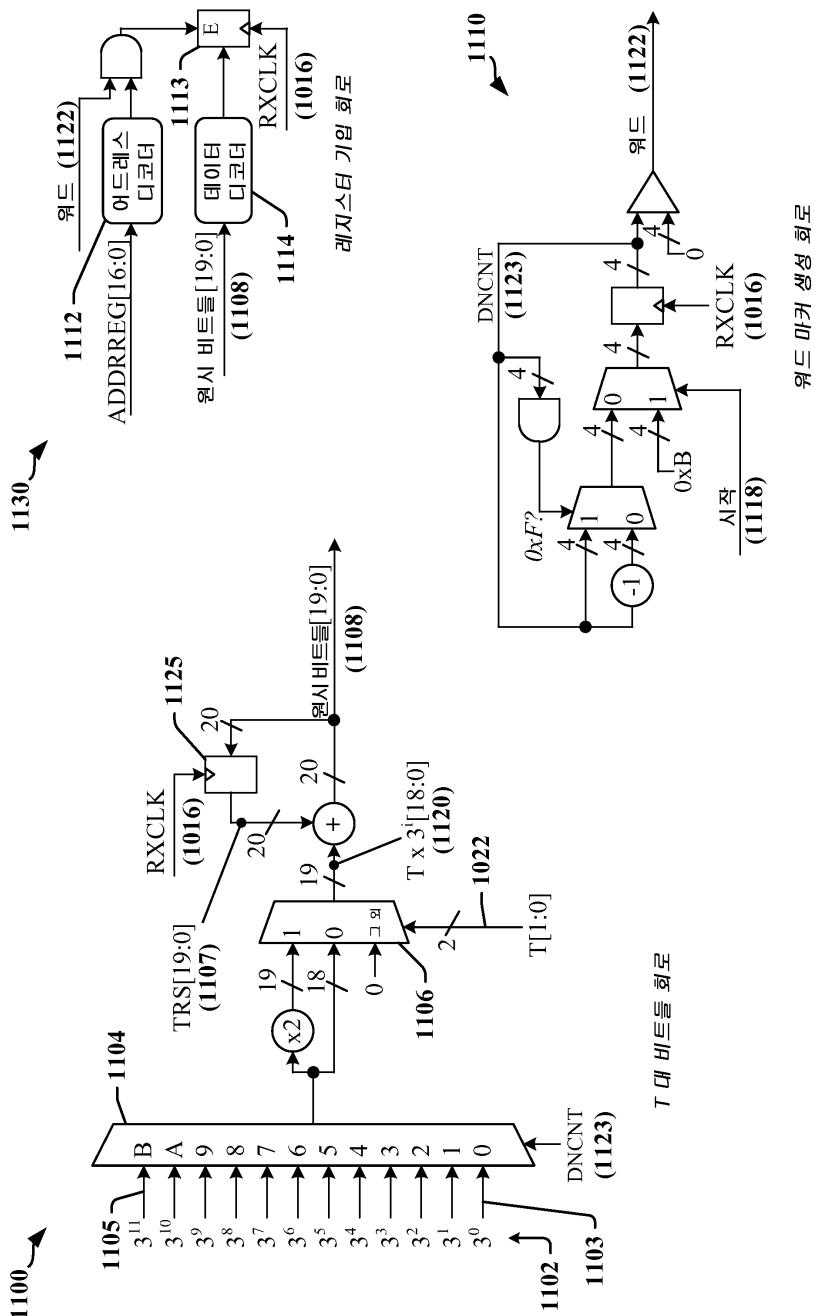


도면10

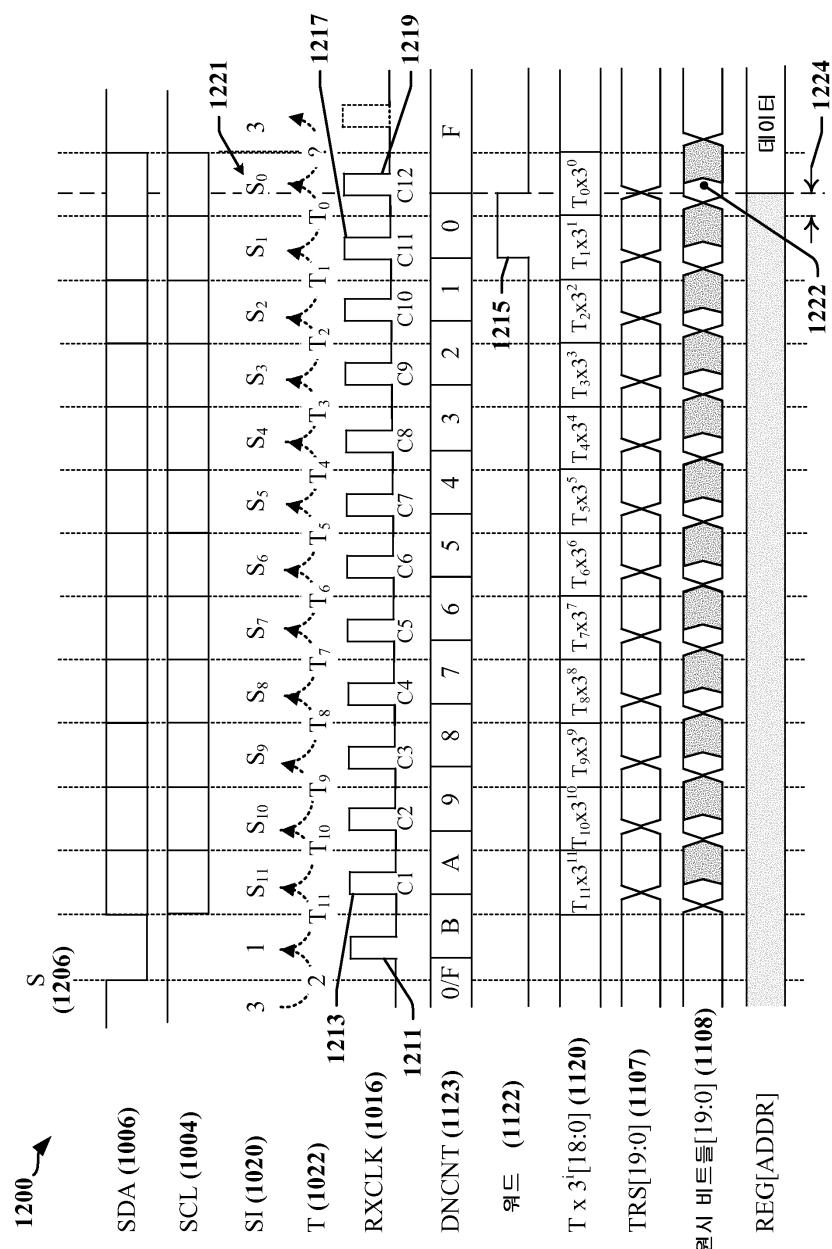


1000 ↗

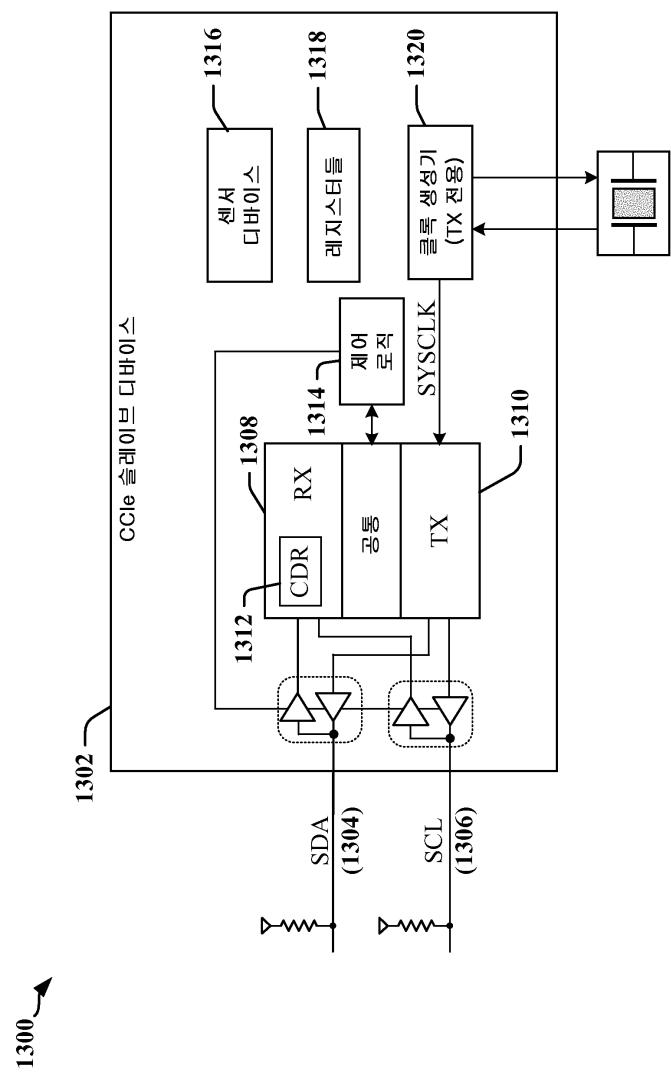
도면11



도면12

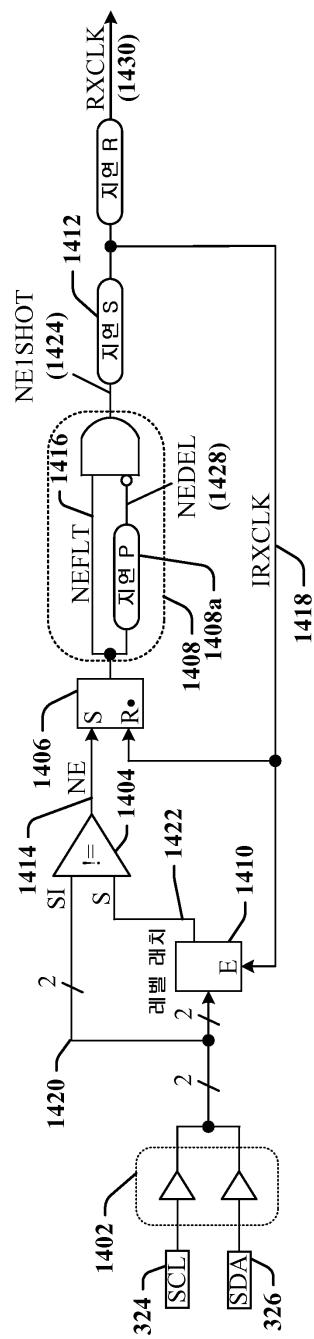


도면13



도면14

1400 ↗

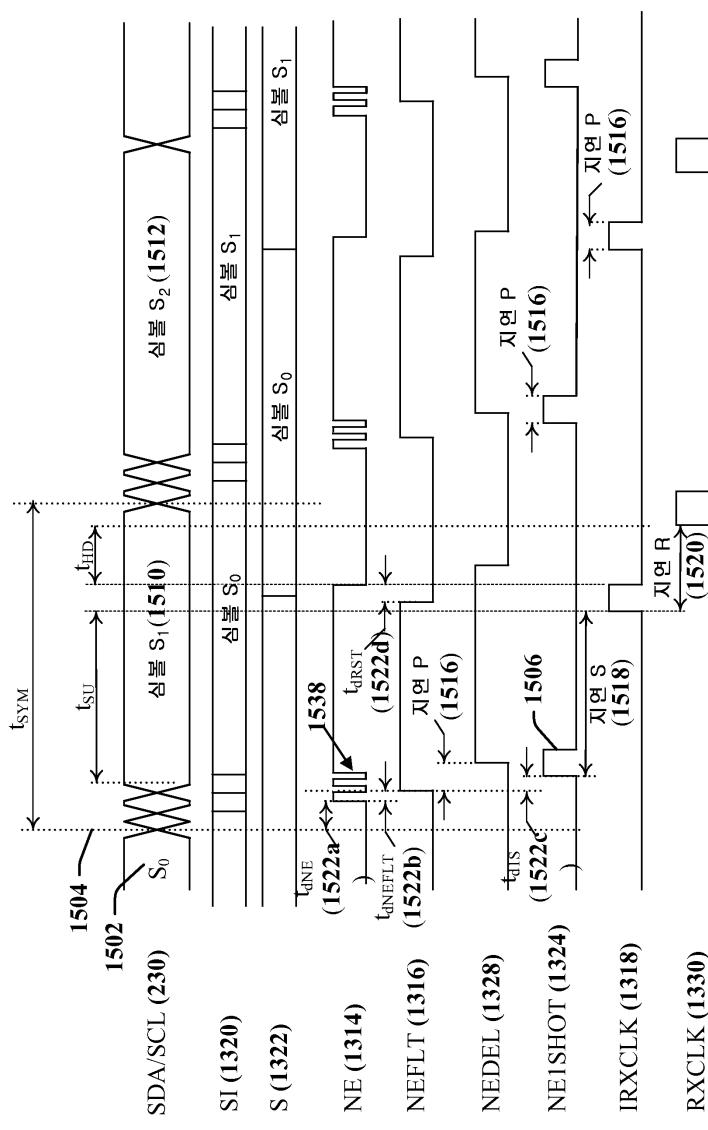


$$\max(\min E_{\text{주기}}, \min R_{\text{주기}}) < \text{지연 } P$$

내부 FF 셋업 최소 < 지연 R

도면15

1500 ↗



도면16

$$\begin{aligned}
 \text{비트들} = & T_{11} \times 3^{11} \\
 & + T_{10} \times 3^{10} \\
 & + T_9 \times 3^9 \\
 & + T_8 \times 3^8 \\
 & + T_7 \times 3^7 \\
 & + T_6 \times 3^6 \\
 & + T_5 \times 3^5 \\
 & + T_4 \times 3^4 \\
 & + T_3 \times 3^3 \\
 & + T_2 \times 3^2 \\
 & + T_1 \times 3^1 \\
 & + T_0
 \end{aligned}$$

1600 ↗

도면17

$$\begin{array}{ll}
 T_{11} = M_{11} / 3^{11}, & M_{11} = 비트율 \% 3^{11} \\
 T_{10} = M_{10} / 3^{10}, & M_{10} = M_{11} \% 3^{10} \\
 T_9 = M_9 / 3^9, & M_9 = M_{10} \% 3^9 \\
 T_8 = M_8 / 3^8, & M_8 = M_9 \% 3^8 \\
 T_7 = M_7 / 3^7, & M_7 = M_8 \% 3^7 \\
 T_6 = M_6 / 3^6, & M_6 = M_7 \% 3^6 \\
 T_5 = M_5 / 3^5, & M_5 = M_6 \% 3^5 \\
 T_4 = M_4 / 3^4, & M_4 = M_5 \% 3^4 \\
 T_3 = M_3 / 3^3, & M_3 = M_4 \% 3^3 \\
 T_2 = M_2 / 3^2, & M_2 = M_3 \% 3^2 \\
 T_1 = M_1 / 3, & M_1 = M_2 \% 3 \\
 T_0 = M_0, &
 \end{array}$$

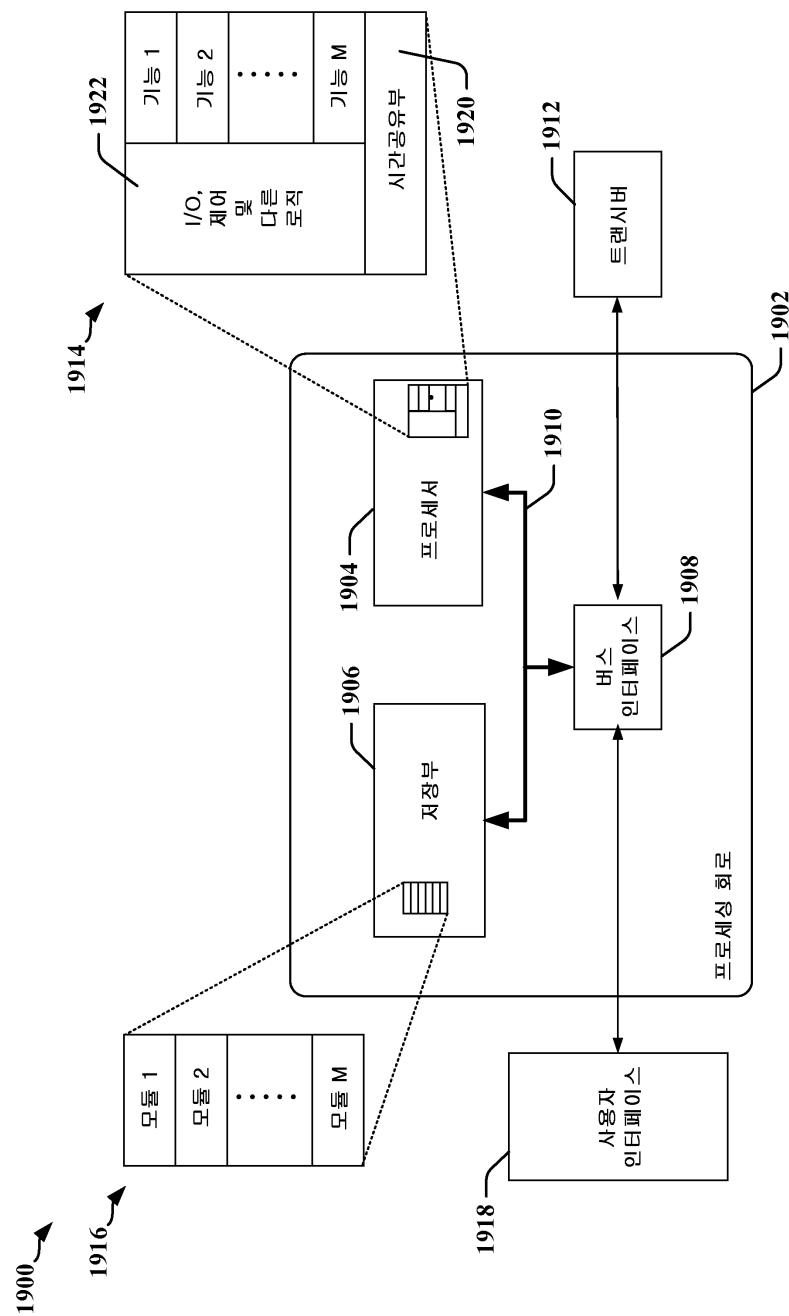
1700 ↗

도면18

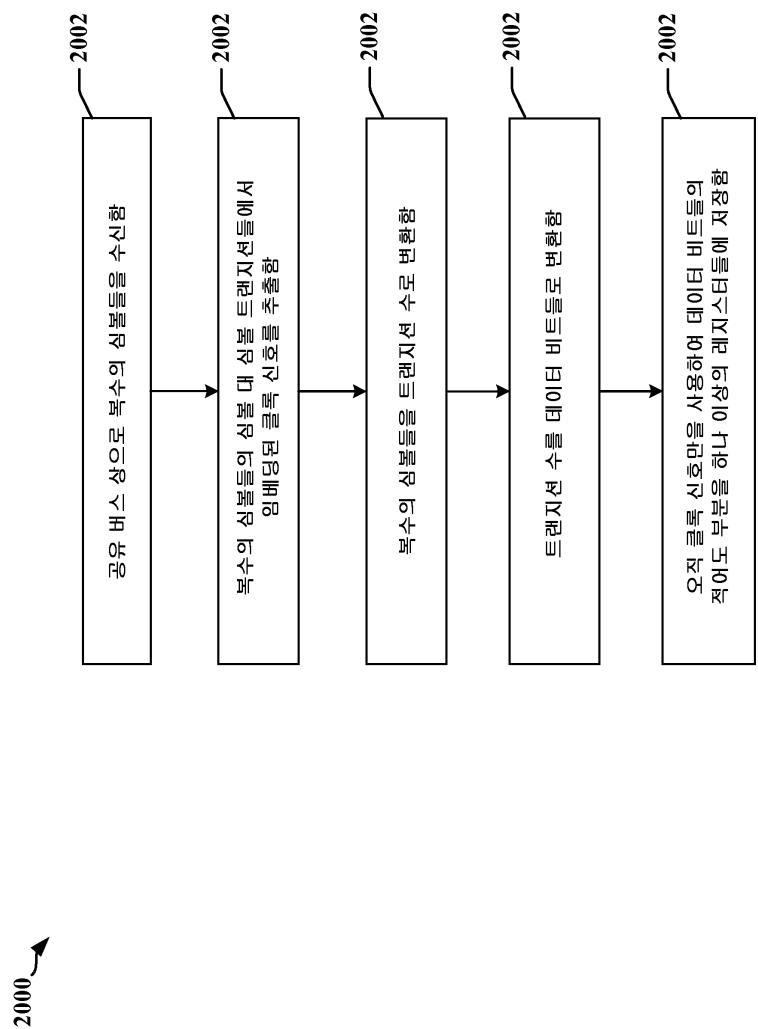
$T_{11} = (비트들 \geq 3^{11} \times 2) ? 2 : (비트들 \geq 3^{11}) ? 1 : 0, M_{11} = 비트들 - T_{11} \times 3^{11}$
 $T_{10} = (M_{11} \geq 3^{10} \times 2) ? 2 : (M_{11} \geq 3^{10}) ? 1 : 0, M_{10} = M_{11} - T_{10} \times 3^{10}$
 $T_9 = (M_{10} \geq 3^9 \times 2) ? 2 : (M_{10} \geq 3^9) ? 1 : 0, M_9 = M_{10} - T_9 \times 3^9$
 $T_8 = (M_9 \geq 3^8 \times 2) ? 2 : (M_9 \geq 3^8) ? 1 : 0, M_8 = M_9 - T_8 \times 3^8$
 $T_7 = (M_8 \geq 3^7 \times 2) ? 2 : (M_8 \geq 3^7) ? 1 : 0, M_7 = M_8 - T_7 \times 3^7$
 $T_6 = (M_7 \geq 3^6 \times 2) ? 2 : (M_7 \geq 3^6) ? 1 : 0, M_6 = M_7 - T_6 \times 3^6$
 $T_5 = (M_6 \geq 3^5 \times 2) ? 2 : (M_6 \geq 3^5) ? 1 : 0, M_5 = M_6 - T_5 \times 3^5$
 $T_4 = (M_5 \geq 3^4 \times 2) ? 2 : (M_5 \geq 3^4) ? 1 : 0, M_4 = M_5 - T_4 \times 3^4$
 $T_3 = (M_4 \geq 3^3 \times 2) ? 2 : (M_4 \geq 3^3) ? 1 : 0, M_3 = M_4 - T_3 \times 3^3$
 $T_2 = (M_3 \geq 3^2 \times 2) ? 2 : (M_3 \geq 3^2) ? 1 : 0, M_2 = M_3 - T_2 \times 3^2$
 $T_1 = (M_2 \geq 3 \times 2) ? 2 : (M_2 \geq 3) ? 1 : 0, M_1 = M_2 - T_1 \times 3$
 $T_0 = M_1$

1800 ↗

도면19



도면20



도면21

