



**(19) 대한민국특허청(KR)**  
**(12) 공개특허공보(A)**

(11) 공개번호 10-2017-0084043  
(43) 공개일자 2017년07월19일

- (51) 국제특허분류(Int. Cl.)  
G06F 13/42 (2006.01) G06F 13/16 (2006.01)  
G06F 13/38 (2006.01)
- (52) CPC특허분류  
G06F 13/4291 (2013.01)  
G06F 13/1673 (2013.01)
- (21) 출원번호 10-2017-7011445
- (22) 출원일자(국제) 2015년11월09일  
심사청구일자 없음
- (85) 번역문제출일자 2017년04월26일
- (86) 국제출원번호 PCT/US2015/059661
- (87) 국제공개번호 WO 2016/077189  
국제공개일자 2016년05월19일
- (30) 우선권주장  
14/538,133 2014년11월11일 미국(US)

- (71) 출원인  
마이크로칩 테크놀로지 인코포레이티드  
미국 85224-6199 아리조나 찬들러 웨스트 찬들러  
블러바드 2355
- (72) 발명자  
캐서디, 제임스  
미국, 애리조나 85142, 쿤 크릭, 이스트 서니데일  
드라이브 19142  
페사벤토, 로드니  
미국, 애리조나 85226, 찬들러, 웨스트 에리 스트  
리트, 4790  
파블로프, 세르게이  
미국, 애리조나 85224, 찬들러, 700 노스 코로나  
도 스트리트 #2058
- (74) 대리인  
특허법인세신

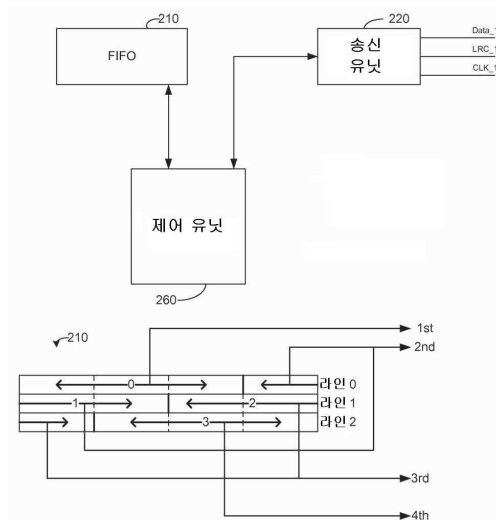
전체 청구항 수 : 총 32 항

(54) 발명의 명칭 멀티-채널 I<sup>2</sup>S 송신 제어 시스템 및 방법

(57) 요약

직렬 주변 장치 인터페이스는 I2S 송신 모드로 동작하도록 구성될 수 있다. 상기 인터페이스는: 데이터, 비트 클럭 및 좌/우 클럭 신호를 위해 외부 핀들과 연결된 송신 유닛; 복수의 메모리 라인들을 갖는 선입선출(FIFO; first-in-first-out) 버퍼; 및 2개의 메모리 라인들로부터 데이터 부분들을 판독하고 그것들을 송신 워드로 어셈블링하고 상기 어셈블링된 송신 워드를 상기 송신 유닛으로 포워딩하도록 동작 가능한 제어 유닛을 가지며, 상기 송신 유닛은 상기 어셈블링된 송신 워드를 상기 외부 데이터 핀을 통해 직렬로 송신하도록 구성된다.

대표도 - 도3



(52) CPC특허분류  
*G06F 13/385* (2013.01)

---

## 명세서

### 청구범위

#### 청구항 1

I<sup>2</sup>S 송신 모드로 동작하도록 구성 가능한 직렬 주변기기 인터페이스로서,

데이터, 비트 클럭 및 좌/우 클럭 신호를 위한 외부 핀들과 연결된 송신 유닛;

복수의 메모리 라인들을 포함하는 선입선출(FIFO; first-in-first-out) 버퍼; 및

2개의 메모리 라인들로부터 데이터 부분들을 판독하고, 상기 데이터 부분들을 송신 워드로 어셈블링하고, 그리고 상기 어셈블링된 송신 워드를 상기 송신 유닛으로 포워딩하도록 동작 가능한 제어 유닛을 포함하고,

상기 송신 유닛은 상기 어셈블링된 송신 워드를 상기 외부 데이터 핀을 통해 직렬로 송신하도록 구성되는, 직렬 주변기기 인터페이스.

#### 청구항 2

제 1 항에 있어서,

상기 FIFO 버퍼는 3개의 32-비트 메모리 라인들을 포함하는, 직렬 주변기기 인터페이스.

#### 청구항 3

제 1 항 또는 제 2 항에 있어서,

24-비트 동작 모드에서 상기 제어 유닛은,

제 1 메모리 라인에 액세스하여 상위 24 비트들을 판독하고 이것들을 상기 송신 유닛에 전송하고,

이후에 상기 제 1 메모리 라인에 액세스하여 하위 8 비트들을 판독하고 제 2 메모리 라인에 액세스하여 상위 16 비트들을 판독하고, 결합된 24 비트 워드를 상기 송신 유닛에 전송하고,

이후에 상기 제 2 메모리 라인에 액세스하여 하위 16 비트들을 판독하고 제 3 메모리 라인에 액세스하여 상위 8 비트들을 판독하고, 결합된 24 비트 워드를 상기 송신 유닛에 전송하고, 그리고

이후에 상기 제 3 메모리 라인에 액세스하여 하위 24 비트들을 판독하고 이것들을 상기 송신 유닛에 전송하도록 구성되는, 직렬 주변기기 인터페이스.

#### 청구항 4

제 3 항에 있어서,

16-비트 동작 모드에서 상기 제어 유닛은,

상기 제 1 메모리 라인에 액세스하여 상위 16 비트들을 판독하고 이것들을 상기 송신 유닛에 전송하고,

이후에 상기 제 1 메모리 라인에 액세스하여 하위 16 비트들을 판독하여 이것들을 상기 송신 유닛에 전송하고,

이후에 상기 제 2 및 제 3 메모리 라인에 대해 상기 액세스 및 전송을 반복하도록 구성되는, 직렬 주변기기 인터페이스.

#### 청구항 5

I<sup>2</sup>S 송신 모드로 동작하도록 구성 가능한 직렬 주변기기 인터페이스로서,

복수의 송신 유닛들 - 상기 송신 유닛들의 각각은 적어도 외부 데이터 핀과 연결됨 -;

관련된 선입선출(FIFO) 버퍼들 - 상기 FIFO 버퍼들의 각각은 복수의 메모리 라인들을 포함함 -; 및

2개의 메모리 라인들로부터 데이터 부분들을 판독하고, 상기 데이터 부분들을 송신 워드로 어셈블링하고, 그리고

고 상기 어셈블링된 송신 워드를 상기 송신 유닛으로 포워딩하도록 동작 가능한 제어 유닛을 포함하고, 상기 송신 유닛은 상기 어셈블링된 송신 워드를 상기 외부 데이터 핀을 통해 직렬로 송신하도록 구성되는, 직렬 주변기기 인터페이스.

**청구항 6**

제 5 항에 있어서,  
각각의 FIFO 버퍼는 3개의 32-비트 메모리 라인들을 포함하는, 직렬 주변기기 인터페이스.

**청구항 7**

제 5 항 또는 제 6 항에 있어서,  
상기 송신 유닛들 중 하나는 마스터 유닛으로 구성되고, 나머지 송신 유닛들은 슬레이브 유닛들로 구성되는, 직렬 주변기기 인터페이스.

**청구항 8**

제 7 항에 있어서,  
쓰기 동작들이 각각의 FIFO 버퍼 내의 동일한 어드레스에 데이터를 기록하도록 모든 FIFO 버퍼들 상에 입력 동작들이 수행되는, 직렬 주변기기 인터페이스.

**청구항 9**

제 5 항 내지 제 8 항 중 어느 한 항에 있어서,  
각각의 송신 유닛은 관련된 FIFO 버퍼로부터 단일 데이터 워드를 판독하고, 상기 단일 데이터 워드의 어드레스는 관련된 제어 레지스터 내에 정의되는, 직렬 주변기기 인터페이스.

**청구항 10**

제 5 항 내지 제 9 항 중 어느 한 항에 있어서,  
24-비트 동작 모드에서, 제 1 송신 유닛은 제 1 메모리 라인의 상위 24 비트들을 판독하고, 제 2 송신 유닛은 상기 제 1 메모리 라인의 하위 8 비트들과 제 2 메모리의 상위 16 비트들을 판독하고, 제 3 송신 유닛은 상기 제 2 메모리 라인의 하위 16 비트들과 제 3 메모리 라인의 상위 8 비트들을 판독하고, 그리고 제 4 송신 유닛은 관련된 FIFO 버퍼의 상기 제 3 메모리 라인의 하위 24 비트들을 판독하는, 직렬 주변기기 인터페이스.

**청구항 11**

제 5 항 내지 제 10 항 중 어느 한 항에 있어서,  
16-비트 동작 모드에서, 제 1 송신 유닛은 각각의 메모리 라인의 상위 16 비트들을 판독하고, 제 2 송신 유닛은 각각의 메모리 라인의 하위 16 비트들을 판독하는, 직렬 주변기기 인터페이스.

**청구항 12**

제 5 항 내지 제 11 항 중 어느 한 항에 있어서,  
상기 16-비트 동작 모드에서, 상기 FIFO 버퍼들은 단일의 32 비트 메모리 라인만을 사용하는, 직렬 주변기기 인터페이스.

**청구항 13**

제 5 항 내지 제 12 항 중 어느 한 항에 있어서,  
상기 FIFO 버퍼들 중 오직 하나만이 모든 4개의 송신 유닛들에 사용되는, 직렬 주변기기 인터페이스.

**청구항 14**

제 5 항 내지 제 13 항 중 어느 한 항에 있어서,

상기 24-비트 동작 모드에서 상기 제어 유닛은,

선택된 FIFO 버퍼의 상기 제 1 메모리 라인에 액세스하여 상위 24 비트들을 판독하고 이것들을 상기 제 1 송신 유닛에 전송하고,

이후에 상기 선택된 FIFO 버퍼의 상기 제 1 메모리 라인에 액세스하여 하위 8 비트들을 판독하고 상기 제 2 메모리 라인에 액세스하여 상위 16 비트들을 판독하고, 결합된 24 비트 워드를 상기 제 2 송신 유닛에 전송하고,

이후에 상기 선택된 FIFO 버퍼의 상기 제 2 메모리 라인에 액세스하여 하위 16 비트들을 판독하고 상기 제 3 메모리 라인에 액세스하여 상위 8 비트들을 판독하고, 결합된 24 비트 워드를 상기 제 3 송신 유닛에 전송하고, 그리고

이후에 상기 선택된 FIFO 버퍼의 상기 제 3 메모리 라인에 액세스하여 하위 24 비트들을 판독하고 이것들을 상기 제 4 송신 유닛에 전송하도록 구성되는, 직렬 주변기기 인터페이스.

**청구항 15**

제 5 항 내지 제 13 항 중 어느 한 항에 있어서,

16-비트 동작 모드에서 상기 제어 유닛은,

선택된 FIFO 버퍼의 각각의 메모리 라인에 액세스하여 상위 16 비트들을 판독하고 이것들을 상기 제 1 송신 유닛에 전송하고,

이후에 상기 선택된 FIFO 버퍼의 각각의 메모리 라인에 액세스하여 하위 16 비트들을 판독하고 이것들을 상기 제 2 송신 유닛에 전송하도록 구성되는, 직렬 주변기기 인터페이스.

**청구항 16**

제 5 항 내지 제 15 항 중 어느 한 항에 있어서,

상기 16-비트 동작 모드에서, 상기 FIFO 버퍼는 단일의 32 비트 메모리 라인만을 사용하는, 직렬 주변기기 인터페이스.

**청구항 17**

$I^2S$  송신 모드로 동작하도록 구성 가능한 직렬 주변기기 인터페이스를 동작시키는 방법으로서,

$I^2S$  모드로 동작하도록 상기 직렬 주변기기 인터페이스를 구성하는 단계 - 상기 직렬 주변기기 인터페이스는 데이터, 비트 클럭 및 좌/우 클럭 신호를 위한 외부 핀들과 연결된 송신 유닛을 포함함 -;

복수의 메모리 라인들을 포함하는 선입선출(FIFO) 버퍼를 제공하는 단계;

2개의 메모리 라인들로부터 데이터 부분들을 판독하고 상기 데이터 부분들을 송신 워드로 어셈블링하고 그리고 상기 어셈블링된 송신 워드를 상기 송신 유닛으로 포워딩하도록, 상기 직렬 주변기기 인터페이스를 제어하는 단계; 및

상기 어셈블링된 송신 워드를 상기 외부 데이터 핀을 통해 직렬로 송신하는 단계를 포함하는, 방법.

**청구항 18**

제 17 항에 있어서,

상기 FIFO 버퍼는 3개의 32-비트 메모리 라인들을 포함하는, 방법.

**청구항 19**

제 17 항 또는 제 18 항에 있어서,

24-비트 동작 모드에서 상기 방법은,

제 1 메모리 라인에 액세스하여 상위 24 비트들을 판독하고 이 비트들을 상기 송신 유닛에 전송하는 것,

이후에 상기 제 1 메모리 라인에 액세스하여 하위 8 비트들을 판독하고 제 2 메모리 라인에 액세스하여 상위 16

비트들을 판독하고, 결합된 24 비트 워드를 상기 송신 유닛에 전송하는 것,

이후에 상기 제 2 메모리 라인에 액세스하여 하위 16 비트들을 판독하고 제 3 메모리 라인에 액세스하여 상위 8 비트들을 판독하고, 결합된 24 비트 워드를 상기 송신 유닛에 전송하는 것, 그리고

이후에 상기 제 3 메모리 라인에 액세스하여 하위 24 비트들을 판독하고 이 비트들을 상기 송신 유닛에 전송하는 것을 포함하는, 방법.

**청구항 20**

제 19 항에 있어서,

16-비트 동작 모드에서 상기 방법은,

상기 제 1 메모리 라인에 액세스하여 상위 16 비트들을 판독하고 이 비트들을 상기 송신 유닛에 전송하는 것,

이후에 상기 제 1 메모리 라인에 액세스하여 하위 16 비트들을 판독하고 이 비트들을 상기 송신 유닛에 전송하는 것,

이후에 상기 제 2 및 제 3 메모리 라인에 대해 상기 액세스 및 전송 단계들을 반복하는 것을 포함하는, 방법.

**청구항 21**

IS 송신 모드로 동작하도록 구성 가능한 직렬 주변기기 인터페이스를 동작시키기 위한 방법으로서,

복수의 송신 유닛들을 제공하는 단계 - 상기 송신 유닛들의 각각은 적어도 외부 데이터 핀과 연결됨 -;

관련된 선입선출(FIFO) 버퍼들을 제공하는 단계 - 상기 FIFO 버퍼들의 각각은 복수의 메모리 라인들을 포함함 -;

2개의 메모리 라인들로부터 데이터 부분들을 판독하는 단계;

상기 데이터 부분들을 송신 워드로 어셈블링하는 단계;

상기 어셈블링된 송신 워드를 상기 송신 유닛으로 포워딩하는 단계; 및

상기 어셈블링된 송신 워드를 상기 외부 데이터 핀을 통해 직렬로 송신하는 단계를 포함하는, 방법.

**청구항 22**

제 21 항에 있어서,

각각의 FIFO 버퍼는 3개의 32-비트 메모리 라인들을 포함하는, 방법.

**청구항 23**

제 21 항 또는 제 22 항에 있어서,

상기 송신 유닛들 중 하나를 마스터 유닛으로 구성하고 나머지 송신 유닛들을 슬레이브 유닛들로 구성하는 단계를 포함하는, 방법.

**청구항 24**

제 23 항에 있어서,

쓰기 동작들이 각각의 FIFO 버퍼 내의 동일한 어드레스에 데이터를 기록하도록 모든 FIFO 버퍼들 상에 입력 동작들을 수행하는 것을 포함하는, 방법.

**청구항 25**

제 21 항 내지 제 24 항 중 어느 한 항에 있어서,

각각의 송신 유닛은 관련된 FIFO 버퍼로부터 단일 데이터 워드를 판독하고, 상기 단일 데이터 워드의 어드레스는 관련된 제어 레지스터 내에 정의되는, 방법.

**청구항 26**

제 21 항 내지 제 25 항 중 어느 한 항에 있어서,

24-비트 동작 모드에서, 제 1 송신 유닛은 제 1 메모리 라인의 상위 24 비트들을 판독하고, 제 2 송신 유닛은 상기 제 1 메모리 라인의 하위 8 비트들과 제 2 메모리의 상위 16 비트들을 판독하고, 제 3 송신 유닛은 상기 제 2 메모리 라인의 하위 16 비트들과 제 3 메모리 라인의 상위 8 비트들을 판독하고, 그리고 제 4 송신 유닛은 관련된 FIFO 버퍼의 상기 제 3 메모리 라인의 하위 24 비트들을 판독하는, 방법.

**청구항 27**

제 21 항 내지 제 26 항 중 어느 한 항에 있어서,

16-비트 동작 모드에서, 제 1 송신 유닛은 각각의 메모리 라인의 상위 16 비트들을 판독하고, 제 2 송신 유닛은 각각의 메모리 라인의 하위 16 비트들을 판독하는, 방법.

**청구항 28**

제 21 항 내지 제 27 항 중 어느 한 항에 있어서,

상기 16-비트 동작 모드에서, 상기 FIFO 버퍼들은 단일의 32 비트 메모리 라인만을 사용하는, 방법.

**청구항 29**

제 21 항 내지 제 28 항 중 어느 한 항에 있어서,

상기 FIFO 버퍼들 중 오직 하나만이 모든 4개의 송신 유닛들에 사용되는, 방법.

**청구항 30**

제 21 항 내지 제 29 항 중 어느 한 항에 있어서,

24-비트 동작 모드에서 상기 방법은,

선택된 FIFO 버퍼의 상기 제 1 메모리 라인에 액세스하여 상위 24 비트들을 판독하고, 이 비트들을 상기 제 1 송신 유닛에 전송하는 것,

이후에 상기 선택된 FIFO 버퍼의 상기 제 1 메모리 라인에 액세스하여 하위 8 비트들을 판독하고 상기 제 2 메모리 라인에 액세스하여 상위 16 비트들을 판독하고, 결합된 24 비트 워드를 상기 제 2 송신 유닛에 전송하는 것,

이후에 상기 선택된 FIFO 버퍼의 상기 제 2 메모리 라인에 액세스하여 하위 16 비트들을 판독하고 상기 제 3 메모리 라인에 액세스하여 상위 8 비트들을 판독하고, 결합된 24 비트 워드를 상기 제 3 송신 유닛에 전송하는 것, 그리고

이후에 상기 선택된 FIFO 버퍼의 상기 제 3 메모리 라인에 액세스하여 하위 24 비트들을 판독하고 이 비트들을 상기 제 4 송신 유닛에 전송하는 것을 포함하는, 방법.

**청구항 31**

제 21 항 내지 제 30 항 중 어느 한 항에 있어서,

16-비트 동작 모드에서 상기 방법은,

선택된 FIFO 버퍼의 각각의 메모리 라인에 액세스하여 상위 16 비트들을 판독하고, 이 비트들을 상기 제 1 송신 유닛에 전송하는 것, 그리고

이후에 상기 선택된 FIFO 버퍼의 각각의 메모리 라인에 액세스하여 하위 16 비트들을 판독하고 이 비트들을 상기 제 2 송신 유닛에 전송하는 것을 포함하는, 방법.

**청구항 32**

제 21 항 내지 제 31 항 중 어느 한 항에 있어서,

상기 16-비트 동작 모드에서, 상기 FIFO 버퍼는 단일의 32 비트 메모리 라인만을 사용하는, 방법.

**발명의 설명**

**기술 분야**

[0001] 본 개시는 동기식 직렬 버스에 관한 것으로, 특히 I<sup>2</sup>S 유형의 동기식 직렬 버스에 관한 것이다.

**배경 기술**

[0002] 클럭 및 데이터가 개별 라인들에서 송신되는 다양한 동기식 직렬 프로토콜들이 존재한다. 동기식 직렬 인터페이스의 가장 일반적인 구현들 중 하나는 입력 및 출력을 위한 개별 데이터 라인들, 클럭 라인, 옵션의 선택 라인 및/또는 슬레이브 선택 라인을 포함하는 직렬 주변기기 인터페이스(SPI) 버스이다.

[0003] I<sup>2</sup>S 버스는 비슷한 개수의 라인들을 사용하지만 서로 다른 송신 프로토콜을 제공한다. 마이크로컨트롤러들은 SPI 인터페이스가 I2S 인터페이스를 에뮬레이션할 수 있도록 개선된 SPI 프로토콜을 종종 실행한다. I2S 프로토콜은 클럭 라인 상의 비트 클럭 신호(BCLK)와 별도의 워드 클럭 라인을 사용한다. 워드 클럭은 종종 좌/우 클럭 신호(LRCLK)로 지칭된다. 일반적으로, LRCLK의 각 에지(edge)에서, 좌측 또는 우측 채널의 데이터 워드는 비트 클럭 신호를 사용하여 직렬로 송신된다. I2S 프로토콜은 오디오 데이터에 최적화되어 있다. 구성에 따라, 16, 24 또는 32 비트의 오디오 데이터 워드가 디바이스들 간에 전송된다. 특히, 24 비트 모드는 종종 32 비트를 사용하여 에뮬레이션되거나 또는 일부 데이터 처리가 올바르게 구현되는 것을 필요로 한다.

**발명의 내용**

**해결하려는 과제**

[0004] 특히 마이크로컨트롤러에서, I<sup>2</sup>S 인터페이스의 향상된 구현이 필요하다.

**과제의 해결 수단**

[0005] 일 실시예에 따르면, 직렬 주변기기 인터페이스는 I<sup>2</sup>S 송신 모드로 동작하도록 구성 가능하고, 데이터, 비트 클럭 및 좌/우 클럭 신호를 위한 외부 핀들과 연결된 송신 유닛; 복수의 메모리 라인들을 포함하는 선입선출(FIFO; first-in-first-out) 버퍼; 및 2개의 메모리 라인들로부터 데이터 부분들을 판독하고, 상기 데이터 부분들을 송신 워드로 어셈블링하고, 그리고 상기 어셈블링된 송신 워드를 상기 송신 유닛으로 포워딩하도록 동작 가능한 제어 유닛을 포함할 수 있으며, 여기서 상기 송신 유닛은 상기 어셈블링된 송신 워드를 상기 외부 데이터 핀을 통해 직렬로 송신하도록 구성된다.

[0006] 추가 실시예에 따르면, 상기 FIFO 버퍼는 3개의 32-비트 메모리 라인들을 포함할 수 있다. 추가 실시예에 따르면, 24-비트 동작 모드에서 상기 제어 유닛은, 제 1 메모리 라인에 액세스하여 상위 24 비트들을 판독하고 이것들을 상기 송신 유닛에 전송하고, 이후에 상기 제 1 메모리 라인에 액세스하여 하위 8 비트들을 판독하고 제 2 메모리 라인에 액세스하여 상위 16 비트들을 판독하고, 결합된 24 비트 워드를 상기 송신 유닛에 전송하고, 이후에 상기 제 2 메모리 라인에 액세스하여 하위 16 비트들을 판독하고 제 3 메모리 라인에 액세스하여 상위 8 비트들을 판독하고, 결합된 24 비트 워드를 상기 송신 유닛에 전송하고, 그리고 이후에 상기 제 3 메모리 라인에 액세스하여 하위 24 비트들을 판독하고 이것들을 상기 송신 유닛에 전송하도록 구성된다. 추가 실시예에 따르면, 16-비트 동작 모드에서 상기 제어 유닛은, 상기 제 1 메모리 라인에 액세스하여 상위 16 비트들을 판독하고 이것들을 상기 송신 유닛에 전송하고, 이후에 상기 제 1 메모리 라인에 액세스하여 하위 16 비트들을 판독하여 이것들을 상기 송신 유닛에 전송하고, 이후에 상기 제 2 및 제 3 메모리 라인에 대해 상기 액세스 및 전송을 반복하도록 구성된다.

[0007] 또 하나의 실시예에 따르면, 직렬 주변기기 인터페이스는 I<sup>2</sup>S 송신 모드로 동작하도록 구성 가능하고, 복수의 송신 유닛들 - 상기 송신 유닛들의 각각은 적어도 외부 데이터 핀과 연결됨 -; 관련된 선입선출(FIFO) 버퍼들 - 상기 FIFO 버퍼들의 각각은 복수의 메모리 라인들을 포함함 -; 및 2개의 메모리 라인들로부터 데이터 부분들을 판독하고, 상기 데이터 부분들을 송신 워드로 어셈블링하고, 그리고 상기 어셈블링된 송신 워드를 상기 송신 유닛으로 포워딩하도록 동작 가능한 제어 유닛을 포함할 수 있으며, 여기서 상기 송신 유닛은 상기 어셈블링된 송

신 워드를 상기 외부 데이터 핀을 통해 직렬로 송신하도록 구성된다.

[0008] 추가 실시예에 따르면, 각각의 FIFO 버퍼는 3개의 32-비트 메모리 라인들을 포함한다. 추가 실시예에 따르면, 상기 송신 유닛들 중 하나는 마스터 유닛으로 구성될 수 있고, 나머지 송신 유닛들은 슬레이브 유닛들로 구성된다. 추가 실시예에 따르면, 쓰기 동작들이 각각의 FIFO 버퍼 내의 동일한 어드레스에 데이터를 기록하도록 모든 FIFO 버퍼들 상에 입력 동작들이 수행된다. 추가 실시예에 따르면, 각각의 송신 유닛은 관련된 FIFO 버퍼로부터 단일 데이터 워드를 판독하고, 여기서 상기 단일 데이터 워드의 어드레스는 관련된 제어 레지스터 내에 정의된다. 추가 실시예에 따르면, 24-비트 동작 모드에서, 제 1 송신 유닛은 제 1 메모리 라인의 상위 24 비트들을 판독하고, 제 2 송신 유닛은 상기 제 1 메모리 라인의 하위 8 비트들과 제 2 메모리의 상위 16 비트들을 판독하고, 제 3 송신 유닛은 상기 제 2 메모리 라인의 하위 16 비트들과 제 3 메모리 라인의 상위 8 비트들을 판독하고, 그리고 제 4 송신 유닛은 관련된 FIFO 버퍼의 상기 제 3 메모리 라인의 하위 24 비트들을 판독한다. 추가 실시예에 따르면, 16-비트 동작 모드에서, 제 1 송신 유닛은 각각의 메모리 라인의 상위 16 비트들을 판독하고, 제 2 송신 유닛은 각각의 메모리 라인의 하위 16 비트들을 판독한다. 추가 실시예에 따르면, 상기 16-비트 동작 모드에서, 상기 FIFO 버퍼들은 단일의 32 비트 메모리 라인만을 사용한다. 추가 실시예에 따르면, 상기 FIFO 버퍼들 중 오직 하나만이 모든 4개의 송신 유닛들에 사용된다. 추가 실시예에 따르면, 상기 24-비트 동작 모드에서 상기 제어 유닛은, 선택된 FIFO 버퍼의 상기 제 1 메모리 라인에 액세스하여 상위 24 비트들을 판독하고 이것들을 상기 제 1 송신 유닛에 전송하고, 이후에 상기 선택된 FIFO 버퍼의 상기 제 1 메모리 라인에 액세스하여 하위 8 비트들을 판독하고 상기 제 2 메모리 라인에 액세스하여 상위 16 비트들을 판독하고, 결합된 24 비트 워드를 상기 제 2 송신 유닛에 전송하고, 이후에 상기 선택된 FIFO 버퍼의 상기 제 2 메모리 라인에 액세스하여 하위 16 비트들을 판독하고 상기 제 3 메모리 라인에 액세스하여 상위 8 비트들을 판독하고, 결합된 24 비트 워드를 상기 제 3 송신 유닛에 전송하고, 그리고 이후에 상기 선택된 FIFO 버퍼의 상기 제 3 메모리 라인에 액세스하여 하위 24 비트들을 판독하고 이것들을 상기 제 4 송신 유닛에 전송하도록 구성된다. 추가 실시예에 따르면, 16-비트 동작 모드에서 상기 제어 유닛은, 선택된 FIFO 버퍼의 각각의 메모리 라인에 액세스하여 상위 16 비트들을 판독하고 이것들을 상기 제 1 송신 유닛에 전송하고, 이후에 상기 선택된 FIFO 버퍼의 각각의 메모리 라인에 액세스하여 하위 16 비트들을 판독하고 이것들을 상기 제 2 송신 유닛에 전송하도록 구성된다. 추가 실시예에 따르면, 상기 16-비트 동작 모드에서, 상기 FIFO 버퍼는 단일의 32 비트 메모리 라인만을 사용한다.

[0009] 다른 또 하나의 실시예에 따르면,  $I^2S$  송신 모드로 동작하도록 구성 가능한 직렬 주변기기 인터페이스를 동작시키는 방법은:  $I^2S$  모드로 동작하도록 상기 직렬 주변기기 인터페이스를 구성하는 단계 - 상기 직렬 주변기기 인터페이스는 데이터, 비트 클럭 및 좌/우 클럭 신호를 위한 외부 핀들과 연결된 송신 유닛을 포함함 -; 복수의 메모리 라인들을 포함하는 선입선출(FIFO) 버퍼를 제공하는 단계; 2개의 메모리 라인들로부터 데이터 부분들을 판독하고 상기 데이터 부분들을 송신 워드로 어셈블링하고 그리고 상기 어셈블링된 송신 워드를 상기 송신 유닛으로 포워딩하도록, 상기 직렬 주변기기 인터페이스를 제어하는 단계; 및 상기 어셈블링된 송신 워드를 상기 외부 데이터 핀을 통해 직렬로 송신하는 단계를 포함할 수 있다.

[0010] 상기 방법의 추가 실시예에 따르면, 상기 FIFO 버퍼는 3개의 32-비트 메모리 라인들을 포함할 수 있다. 상기 방법의 추가 실시예에 따르면, 24-비트 동작 모드에서 상기 방법은, 제 1 메모리 라인에 액세스하여 상위 24 비트들을 판독하고 이 비트들을 상기 송신 유닛에 전송하는 것, 이후에 상기 제 1 메모리 라인에 액세스하여 하위 8 비트들을 판독하고 제 2 메모리 라인에 액세스하여 상위 16 비트들을 판독하고, 결합된 24 비트 워드를 상기 송신 유닛에 전송하는 것, 이후에 상기 제 2 메모리 라인에 액세스하여 하위 16 비트들을 판독하고 제 3 메모리 라인에 액세스하여 상위 8 비트들을 판독하고, 결합된 24 비트 워드를 상기 송신 유닛에 전송하는 것, 그리고 이후에 상기 제 3 메모리 라인에 액세스하여 하위 24 비트들을 판독하고 이 비트들을 상기 송신 유닛에 전송하는 것을 포함한다. 상기 방법의 추가 실시예에 따르면, 16-비트 동작 모드에서 상기 방법은, 상기 제 1 메모리 라인에 액세스하여 상위 16 비트들을 판독하고 이 비트들을 상기 송신 유닛에 전송하는 것, 이후에 상기 제 1 메모리 라인에 액세스하여 하위 16 비트들을 판독하고 이 비트들을 상기 송신 유닛에 전송하는 것, 이후에 상기 제 2 및 제 3 메모리 라인에 대해 상기 액세스 및 전송 단계들을 반복하는 것을 포함한다.

[0011] 다른 또 하나의 실시예에 따르면,  $I^2S$  송신 모드로 동작하도록 구성 가능한 직렬 주변기기 인터페이스를 동작시키기 위한 방법은: 복수의 송신 유닛들을 제공하는 단계 - 상기 송신 유닛들의 각각은 적어도 외부 데이터 핀과 연결됨 -; 관련된 선입선출(FIFO) 버퍼들을 제공하는 단계 - 상기 FIFO 버퍼들의 각각은 복수의 메모리 라인들을 포함함 -; 2개의 메모리 라인들로부터 데이터 부분들을 판독하는 단계; 상기 데이터 부분들을 송신 워드로

어셈블링하는 단계; 상기 어셈블링된 송신 워드를 상기 송신 유닛으로 포워딩하는 단계; 및 상기 어셈블링된 송신 워드를 상기 외부 데이터 핀을 통해 직렬로 송신하는 단계를 포함할 수 있다.

[0012] 상기 방법의 추가 실시예에 따르면, 각각의 FIFO 버퍼는 3개의 32-비트 메모리 라인들을 포함할 수 있다. 상기 방법의 추가 실시예에 따르면, 상기 방법은 상기 송신 유닛들 중 하나를 마스터 유닛으로 구성하고 나머지 송신 유닛들을 슬레이브 유닛들로 구성하는 단계를 포함할 수 있다. 상기 방법의 추가 실시예에 따르면, 상기 방법은 쓰기 동작들이 각각의 FIFO 버퍼 내의 동일한 어드레스에 데이터를 기록하도록 모든 FIFO 버퍼들 상에 입력 동작들을 수행하는 것을 포함할 수 있다. 상기 방법의 추가 실시예에 따르면, 각각의 송신 유닛은 관련된 FIFO 버퍼로부터 단일 데이터 워드를 판독할 수 있고, 여기서 상기 단일 데이터 워드의 어드레스는 관련된 제어 레지스터 내에 정의된다. 상기 방법의 추가 실시예에 따르면, 24-비트 동작 모드에서, 제 1 송신 유닛은 제 1 메모리 라인의 상위 24 비트들을 판독하고, 제 2 송신 유닛은 상기 제 1 메모리 라인의 하위 8 비트들과 제 2 메모리의 상위 16 비트들을 판독하고, 제 3 송신 유닛은 상기 제 2 메모리 라인의 하위 16 비트들과 제 3 메모리 라인의 상위 8 비트들을 판독하고, 그리고 제 4 송신 유닛은 관련된 FIFO 버퍼의 상기 제 3 메모리 라인의 하위 24 비트들을 판독한다. 상기 방법의 추가 실시예에 따르면, 16-비트 동작 모드에서, 제 1 송신 유닛은 각각의 메모리 라인의 상위 16 비트들을 판독하고, 제 2 송신 유닛은 각각의 메모리 라인의 하위 16 비트들을 판독한다. 상기 방법의 추가 실시예에 따르면, 상기 16-비트 동작 모드에서, 상기 FIFO 버퍼들은 단일의 32 비트 메모리 라인만을 사용한다. 상기 방법의 추가 실시예에 따르면, 상기 FIFO 버퍼들 중 오직 하나만이 모든 4개의 송신 유닛들에 사용된다. 상기 방법의 추가 실시예에 따르면, 24-비트 동작 모드에서 상기 방법은, 선택된 FIFO 버퍼의 상기 제 1 메모리 라인에 액세스하여 상위 24 비트들을 판독하고, 이 비트들을 상기 제 1 송신 유닛에 전송하는 것, 이후에 상기 선택된 FIFO 버퍼의 상기 제 1 메모리 라인에 액세스하여 하위 8 비트들을 판독하고 상기 제 2 메모리 라인에 액세스하여 상위 16 비트들을 판독하고, 결합된 24 비트 워드를 상기 제 2 송신 유닛에 전송하는 것, 이후에 상기 선택된 FIFO 버퍼의 상기 제 2 메모리 라인에 액세스하여 하위 16 비트들을 판독하고 상기 제 3 메모리 라인에 액세스하여 상위 8 비트들을 판독하고, 결합된 24 비트 워드를 상기 제 3 송신 유닛에 전송하는 것, 그리고 이후에 상기 선택된 FIFO 버퍼의 상기 제 3 메모리 라인에 액세스하여 하위 24 비트들을 판독하고 이 비트들을 상기 제 4 송신 유닛에 전송하는 것을 포함한다. 상기 방법의 추가 실시예에 따르면, 16-비트 동작 모드에서 상기 방법은, 선택된 FIFO 버퍼의 각각의 메모리 라인에 액세스하여 상위 16 비트들을 판독하고, 이 비트들을 상기 제 1 송신 유닛에 전송하는 것, 그리고 이후에 상기 선택된 FIFO 버퍼의 각각의 메모리 라인에 액세스하여 하위 16 비트들을 판독하고 이 비트들을 상기 제 2 송신 유닛에 전송하는 것을 포함한다. 상기 방법의 추가 실시예에 따르면, 상기 16-비트 동작 모드에서, 상기 FIFO 버퍼는 단일의 32 비트 메모리 라인만을 사용한다.

**도면의 간단한 설명**

[0013] 도 1은 종래의 직렬 주변기기 인터페이스를 도시한 도면이다.  
 도 2는 다양한 실시예들의 다양한 동작 모드들에 따른 FIFO 버퍼들을 도시한 도면이다.  
 도 3은 하나의 실시예의 블록도를 도시한 도면이다.  
 도 4는 또 하나의 실시예의 블록도를 도시한 도면이다.  
 도 5 및 도 6은 도 4에 따른 실시예의 동작 모드들을 도시한 도면이다.  
 도 7 내지 도 10은 다양한 실시예들에 따른 동작 모드들의 흐름도들을 도시한 도면이다.

**발명을 실시하기 위한 구체적인 내용**

[0014] 도 1은 마이크로컨트롤러 내에 실장될 수 있는 종래의 SPI 인터페이스 주변 디바이스(100)를 나타낸다. 디바이스에 의해 외부에서 전송될 데이터는 내부 버스(110)를 통해 버퍼 레지스터(120)에 쓰여진다. 버퍼 레지스터는 도 1에 도시된 바와 같이 개별의 송신 및 수신 버퍼들을 갖는 선입선출(FIFO; first-in-first-out) 메모리(130)로서 구현될 수 있으며 임의의 적절한 크기를 가질 수 있다. 일부 실시예들에 따르면, 버퍼의 FIFO 기능은 인에이블 또는 디스에이블되도록 프로그램 가능하다. 입력 버퍼(120/130)는 외부 입력 핀들(SDIx)과 외부 출력 핀(SDOx)에 연결되는 전송 시프트 레지스터(140)와 결합된다. 'x'는 마이크로컨트롤러에 여러 개의 유닛들이 있을 수 있음을 나타낸다.

[0015] 슬레이브 선택 및 프레임 동기화 제어 유닛(150)이 제공되어 외부 슬레이브 선택 핀(SSx)과 결합되고, 레지스터

(140)의 출력부와 출력 핀(SDOx) 사이의 3-상태 버퍼를 제어할 수 있다. 클록 제어 유닛(160)은 시프트 레지스터(140)를 동작시키는 시프트 클록 신호를 제공한다. 클록 제어 유닛(160)은 또한 슬레이브 선택 및 프레임 동기화 제어 유닛(150)과 예지 선택 유닛(170)에 결합된다. 보드 레이트(baud rate) 생성기(180)는 다양한 내부 클록 신호들에 의해 프로그램 가능하게 구동될 수 있고, 제어 가능한 드라이버를 통해 외부 핀(SCKx)과 예지 선택 유닛(170)에 결합된다. SPI를 구성하는데 사용될 수 있는 다양한 제어 레지스터들이 실장될 수 있다.

[0016] 이 SPI 주변기기(100)는 마스터 또는 슬레이브 모드, 프레임된 동작, DSP 모드 등과 같은 다양한 동작 모드들로 정상 SPI 인터페이스로서 동작하도록 설계된다. 또한, 이 주변기기는 I<sup>2</sup>S 인터페이스로서 동작하도록 구성될 수 있다. 이 모드에서, SSx 핀은 LRCLK 핀으로서 사용되고, SCKx는 BCLK 라인으로서 동작하며, SDIx 및 SDOx는 데이터 신호 입/출력 핀들로서 동작한다. 유닛(100)이 마스터 혹은 슬레이브로서 동작되는지의 여부에 따라, 핀들(SSx 및 SCKx)은 각각 출력(마스터) 핀과 입력(슬레이브) 핀 중 어느 하나이다.

[0017] 일단 데이터 워드가 버퍼에 기록되면, 주변기기는 쓰여진 데이터를 시프트 레지스터를 통해 출력 핀(SDOx)으로 전송할 것이다. 상기 유닛이 마스터 디바이스로서 구성되면, 각각의 클록 신호들은 디바이스(100)에 의해 생성된다. 슬레이브로서 동작할 때, 디바이스(100)는 각각의 LRCLK 신호의 수신시 버퍼로부터 새로운 데이터 워드를 페치(fetch)한다. 상기 디바이스는 16, 24 또는 32 비트의 데이터 워드들을 전송하도록 프로그램 가능하며, 여기서 상기 FIFO는 복수의 32 비트 레지스터들을 갖도록 설계된다. 16 비트 워드들을 송신하도록 프로그램되면, 각 버퍼 레지스터의 하위 16 비트만이 송신되고 상위 16 비트는 무시될 것이다. 24 비트 워드들을 송신하도록 프로그램되면, 각 버퍼 레지스터의 하위 24 비트만이 송신되고 상위 8 비트는 무시될 것이다. 32 비트 워드들을 송신하도록 프로그램될 때에는, 각각의 전체 버퍼 레지스터가 송신된다. 따라서, 이러한 구현에는 오디오 데이터가 일반적으로 패킹 방식(packed manner)으로 대용량 저장 디바이스에 저장되기 때문에, 데이터의 일부 전처리(pre-processing)를 필요로 한다. 따라서, 32 비트 전송 모드만이 전처리가 필요로하지 않는데, 그 이유는 데이터가 메모리로부터 버퍼로 직접 기록될 수 있기 때문이다.

[0018] 그러나, 16 비트 및 24 비트 모드에서, 각각의 오디오 데이터는 패킹 방식으로 저장되는데, 다시 말해서, 24 비트들의 어떠한 정렬도 일어나지 않아서 메모리를 절약한다. 데이터가 32 비트 기록 동작을 사용하여 버퍼(120/130)에 직접 기록되면, 데이터는 정렬되지 않아 절단될 것이다. 실제로 어떠한 유용한 송신도 이루어질 수 없다. 따라서 데이터는 버퍼에 기록되기 전에 전처리되어야 한다. 특히 24 비트 오디오 데이터의 경우에는, 이것은 매우 시간이 걸리는 처리를 야기한다. 32 비트 판독 동작을 사용하여 메모리로부터 데이터를 검색하는 경우에는, 8 비트 또는 16 비트를 버퍼링하고 다음 판독 데이터(a following read)와 결합해야 한다. 대안적으로는, 8 비트 판독 데이터만이 사용될 수 있고, 24 비트는 3개의 연속적인 판독 동작들을 사용함으로써 어셈블링될 것이다. 어떤 식으로든, 메모리로부터 24 비트 오디오 데이터를 전송하려면, 다른 작업들에 필요할 수 있는 처리 능력(processing power)을 소모하는 약간의 처리가 필요할 것이다. 모든 16 비트 데이터의 정렬로 인해 보다 적은 처리 능력이 필요하더라도, 16 비트 오디오 데이터에 대해 유사한 동작들이 필요하다. 그러나, 각각의 16 비트 워드는 I<sup>2</sup>S 인터페이스의 적절한 동작을 보장하기 위해 32 비트 버퍼 레지스터(120/130)에 개별적으로 저장되어야 한다.

[0019] 도 2는 32 비트, 24 비트 및 16 비트의 패킹된(packed) 데이터를 메모리에 정렬하는 것을 도시한다. 데이터는 메모리 공간이 낭비되지 않도록 가장 효율적인 방법으로 저장된다. 따라서, 각각의 후속 데이터 워드는, 어떠한 메모리 공간 없이 선행 워드를 바로 뒤따라서 임의의 유형의 정렬 오배치(alignment misplacement)를 무시한다. 이것은 일반적으로 32 비트 시스템에서 32 비트 워드들의 경우 문제가 되지 않는데, 그 이유는 첫 번째 워드가 예 300에 표시된 것처럼 정렬되면 32 비트 데이터가 자동으로 정렬되기 때문이다. 따라서, 32 비트 명령어들로 메모리를 읽는 것은 단일의 쓰기 명령어로 각 워드를 FIFO에 배치하는 것을 가능케 한다.

[0020] 310의 예는 24 비트 데이터의 저장을 보여준다. 도 3에서 알 수 있는 바와 같이, 데이터 0 및 4만이 32 비트에 대해 정렬된다. 따라서, 몇 개의 8 비트 읽기 동작들 및 데이터의 병합, 결합된 16 비트 및 8 비트 읽기 명령어들, 또는 32 비트 읽기 명령어들 및 다음 24 비트 워드에 필요한 그것들의 일부들의 중간 저장이 FIFO를 올바르게 채우기 위해 필요하다.

[0021] 320의 예에 도시된 바와 같은 16 비트 데이터 폭의 경우, 모든 다른 워드는 32 비트 경계(boundary)에 정렬될 것이다. 따라서, 단일의 16 비트 읽기 명령어들 또는 중간 버퍼링을 갖는 32 비트 읽기 명령어가 필요하다.

[0022] 분명히, 310의 예에 도시된 24 비트 시나리오가 대부분의 처리를 초래한다. 도 3은 I<sup>2</sup>S 송신 능력을 갖는 개선된 SPI 인터페이스를 도시한다. 여기서, FIFO(210)는 SPI 송신 유닛(220)에 직접 연결되지 않고, 오히려

FIFO(210)에의 액세스는, 예를 들어 프로그램 가능한 동작 모드에 따라 데이터를 송신 유닛(220)에 제공하는 유한 상태 기계(finite state machine; FSM)를 포함할 수 있는 제어 유닛(260)에 의해 제어된다.

- [0023] 이 유닛이 I<sup>2</sup>S 모드로 동작하도록 프로그램될 때, 송신 유닛(220)은 도 3에 도시된 바와 같은 자신의 외부 연결부들을 사용하는데, 즉, 데이터 라인은 오디오 데이터용으로 사용되고, 클럭 라인은 비트 클럭 신호(BCLK)를 제공하고, 그리고 SSx 핀은 LRCLK 신호용으로 사용된다. 도 3의 하단은 SPI 인터페이스가 I2S 프로토콜을 이용하는 24 비트 오디오 모드로 동작한다고 가정한다. 따라서 FIFO(210)는 4개의 오디오 24 비트 데이터 워드들로 메모리로부터 직접 채워질 것이다. 일단 FIFO(210)가 3개의 32 비트 워드들로 채워지면, 제어 유닛은 도 3의 하단에 도시된 바와 같은 FIFO(210)에 액세스할 수 있다. FSM에의 제 1 액세스는 Line0[31..8]을 송신 유닛(220)으로 전송한다. 제 2 액세스는 Line1[31..16]과 연결된 Line0[7..0]을 전송한다. 제 3 액세스는 Line2[31..24]와 연결된 Line1[15..0]을 전송하고, 제 4 액세스는 Line2[23..0]를 전송한다. 그런 다음 시스템은 FIFO가 비어 있고 더 많은 데이터가 메모리로부터 버퍼에 전송될 수 있음을 마이크로프로세서에 알린다. 그러면 마이크로프로세서는 데이터를 처리하거나 재배치하지 않고 메모리로부터 다음 3개의 32 비트 워드들을 버퍼로 직접 전송할 수 있다.
- [0024] 16-비트 동작 모드에서, 상태 머신(260)은 FIFO(210)에 액세스하여 상위 워드(Linex[31..16])와 하위 워드(Linex[15..0])를 교대로 판독한다. 다시, 프로세서는 메모리로부터 검색된 데이터를 재정렬할 필요없이 최적의 속도를 위해 32-비트 읽기/쓰기 명령어들만을 사용하여 데이터를 직접 전송할 수 있다. 또 하나의 실시예에 따르면, 16-비트 동작 모드에서, FIFO(210)는 단일 메모리 라인만을 사용할 수 있다.
- [0025] 도 4는, 다수의 SPI 또는 I<sup>2</sup>S 인터페이스 유닛들(410, 420, 230, 240)이 멀티-채널 기능을 제공하기 위해 FIFO들(210-240)과 함께 동작하는 동기식 직렬 인터페이스 주변기기(400)의 다른 또 하나의 실시예를 도시한다. 이 예에서는 4개의 채널들이 제공된다. 그러나, 다른 실시예들에 따르면, 더 많거나 더 적은 채널들이 제공될 수 있다. 이러한 구현들에서는 각각의 FIFO(210-240)에 대한 FIFO 라인들의 수는 전송되는 오디오 데이터의 블록이 32 비트 메모리 구조에 대해 정렬되도록 그에 따라 증가될 수 있다. 24 비트 데이터 워드들이 32-비트 메모리 시스템에서 4개 워드마다 정렬되기 때문에, 32-비트 시스템에서 4개의 채널의 배수를 고려해야 할 수 있다. 그러나, 위에 논의된 개념은 또한, 최소한 2개의 24-비트 워드들을 저장하는 3개의 16-비트 FIFO 라인들 또는 더 많은 채널들에서는 그것의 배수를 필요로 할 16-비트 시스템에서도 사용될 수 있다. 높은 비트 시스템들은 세 개의 FIFO 라인들에 더 많은 24-비트 워드들을 저장할 수 있을 것이다.
- [0026] 다시, 제어 유닛(260)은, 이들 4개의 채널 유닛들을 구성하는데 사용될 수 있고, 버퍼 FIFO들(210-240)에 저장된 데이터의 전달을 조정하는(coordinate) 유한 상태 머신(FSM)을 포함할 수 있다. 각각의 유닛은 별개의 데이터, LRCD 및 CLK 라인들을 포함한다. 하나의 유닛, 예컨대 유닛(440)은 마스터로서 선택될 수 있고, 나머지 유닛들(410-430)은 슬레이브 디바이스로서 동작할 수 있다. 어떤 유닛이 마스터인지는 각각의 구현들에 따라 선택 가능하거나 또는 고정될 수 있다. 도 4에 도시된 바와 같이, 이 유닛들의 각각의 클럭 라인들은 마스터 유닛(440)이 유닛들(410-420)에 클럭 신호들을 제공하도록 서로 연결된다. 따라서, 4개의 채널들을 통한 오디오 신호들의 모든 송신들은 동기식으로 이루어진다.
- [0027] 외부적으로, 상기 디바이스는 단지 한 세트의 LRC 및 CLK 핀들을 제공할 수 있고, 슬레이브 SPI 유닛들(410-430)의 각각의 관련 핀들은 범용 입력/출력 핀 기능과 같은 다른 목적으로 이용 가능하다. 따라서, 디바이스가 도 4 내지 도 6에 도시된 다양한 모드로 동작할 때, 복수의 외부 핀들이 다른 기능을 위해 사용될 수 있다.
- [0028] 다양한 실시예들에 따르면, 제어 유닛(260)은 패킹된 데이터가 단일 버퍼 FIFO, 예를 들면 FIFO(240)로부터, 또는 각각의 송신 유닛(410-440)과 관련된 각각의 버퍼 FIFO(210-240)로부터 직접 전송될 수 있도록 구성될 수 있다. 제어기는 라인별로(line by line) 버퍼를 판독하도록 구성되고, 도 3과 관련하여 설명된 것과 유사한 방식으로, 저장된 데이터를 정확한(correct) I<sup>2</sup>S 유닛(410-440)으로 전송/조립하는 것을 조정하도록 구성된다. 일 실시예에서, 24 비트 동작 모드에서는, 4개의 24 비트 워드들이 동기식 송신을 위해 모든 4개의 채널들에 분배될 것이다. 마스터(440)가 송신을 트리거하기 때문에, 마스터(440)는 그것의 데이터를 수신하는 마지막 채널이다. 따라서, 일단 데이터가 마스터(440)로 전송되면, 마스터는 모든 다른 유닛들의 송신을 동시에 트리거하도록 모든 다른 유닛들에 각각의 클럭 신호들을 전송하는 송신을 시작한다. 이것은 별개의 명령이나 FIFO에의 기록 완료를 통해 수행될 수 있다.
- [0029] 도 5는, 시스템이 24-비트 오디오 데이터를 갖는 I<sup>2</sup>S 모드로 구성될 때, 도 4와 동일한 구성을 보다 상세히 도

시한다. 4 채널 24-비트 데이터의 경우에는 3 쿼드렛(quadlets) 또는 2 채널 16-비트 데이터의 경우에는 1 쿼드렛의 패킹된 32-비트 포맷으로 데이터를 디코딩하고 전송하기 위해 SPI 마스터 컨트롤러(240)와 최대 3개의 SPI 슬레이브들(210-230)이 제공된다. FIFO 데이터가 판독될 때, 마스터 및 슬레이브 결합들은 전송한 바와 같은 마스터 CLK 및 LRC(동기화 펄스)로 모든 4개 (또는 2개) 채널들에 대해 디코딩된 데이터를 동시에 송신한다. 전송한 바와 같이, 일부 실시예들에 따르면, 16-비트 동작 모드에서, FIFO들(210-240)은 도 5의 FIFO들(210-240)의 맨 위의 표현으로 표시된 바와 같이 단일의 메모리 라인만을 사용할 수 있다.

[0030] 도시된 바와 같이, 시스템은 일 실시예에서 "많은 FIFO들에의 하나의 쓰기"로 동작할 수 있으며, 여기서 3 또는 1 (32-비트) 쿼드렛 형태의 데이터가 모든 스파이(spy) FIFO들(210-240)에 쓰여진다. 이 실시예에 따르면, 이것은 모든 FIFO들이 동시에 어드레싱되어 각각이 동일한 데이터 값들을 얻게 될 하나의 기록 사이클로 행해지지만, 더 많은 처리 시간을 명백히 필요로 할 FIFO들(210-240)에의 개별 기록들로 수행될 수 있다. 일부 실시예들에 따르면, 사용자는 어떤 SPI 유닛(210-240)이 사용자-설정 가능한 SPI 레지스터 값으로써 FIFO 내의 각 쿼드렛으로부터 어떤 비트의 데이터를 얻는지를 정의할 수 있다. 하나의 SPI만이 마스터로 정의되고 나머지 SPI들은 슬레이브로 정의된다. 표 1은 유닛(210-250)의 가능한 설정들을 보여준다.

표 1

SLAVE_EN[2:0]
000 -- Master decode enable
001 -- Slave 1 decode enable
010 -- Slave 2 decode enable
011 -- Slave 3 decode enable
100 -- Master 16-bit decode [15:0] enable
101 -- Master 16-bit decode [31:16]enable
110 -- Slave 1 16-bit decode [31:16] enable
111 -- Slave 1 16-bit decode [15:0] enable

[0031]

[0032] FIFO로부터의 데이터 디코딩을 위한 유한 상태 머신(FSM)은 마스터 또는 슬레이브에 대해 동일하다. 사용자-정의된 SPI 레지스터 값들과 함께 FIFO 데이터를 판독하는 FSM 제어 유닛(FSM controls)은 쿼드렛으로부터의 어떤 비트의 데이터가 SPI 밖으로 시프트되어야 하는지를 결정한다. 마스터/슬레이브 SPI는, 마스터 SPI FSM이 데이터를 보내도록 명령할 때, CPU로부터의 신호에 의해 또는 자동 마스터모드로 데이터의 전송을 시작하도록 명령을 받는다.

[0033]

또 하나의 실시예에 따르면, "사이드 밴드 버스로 마스터 전용 FIFO 쓰기(Master only FIFO write with a side band bus)"가 도 6에 도시된 바와 같이 구현될 수 있다. 이 실시예에서, 3 또는 1 (32-비트) 쿼드렛의 형태의 데이터는 마스터 FIFO에만, 예를 들어 FIFO(240)에만 쓰여진다. 사용자는 어떤 슬레이브 SPI가 사용자-설정 가능한 SPI 레지스터 값(a user-settable SPI register value)으로써 FIFO(240) 내의 각 쿼드렛으로부터 어떤 비트의 데이터를 얻는지를 정의했다. 하나의 SPI만이 마스터로서 정의되고 나머지 SPI들은 슬레이브들로서 정의된다. FIFO(240)로부터의 데이터 디코딩을 위한 FSM은 마스터 또는 슬레이브에 대해 동일하지만, SLAVE(슬레이브)에서만 인에이블된다. 사용자-정의된 SPI 레지스터 값들과 함께 FIFO 데이터를 판독하는 FSM 제어 유닛은, 쿼드렛으로부터의 어떤 비트의 데이터가 SPI 밖으로 시프트되어야 하는지를 결정한다. 마스터/슬레이브 SPI는, 마스터 SPI FSM이 데이터를 보내도록 명령할 때, CPU로부터의 신호(CMD\_SPI\_GO)에 의해 또는 자동 마스터모드로 데이터의 전송을 시작하도록 명령을 받는다. 데이터 버스(MST\_SLV\_DATA[24:0])는 데이터의 송신을 위해 모든 SPI 유닛들에 연결된다. 제 2 버스(MST\_SLV\_BUS[2:0])는 표 2에 따른 제어 값들을 제공한다. 도 6에 도시된 바와 같이, 데이터 및 제어 신호들은 SPI 마스터 유닛(440)을 통해 채널링될 수 있다. 그러나, 다른 실시예들은 제어 유닛(260)에 의해 이 신호들을 제공할 수 있다. 전송한 바와 같이, 일부 실시예들에 따르면, 16-비트 동작 모드에서, FIFO(210)는 단일 메모리 라인만을 사용할 수 있다.

표 2

MST_SLV_BUS[2:0]
000 -- nothing default
1xx -- Master start transmit slave
x01 -- SPI slave#1 get data
x10 -- SPI slave#2 get data
x11 -- SPI slave#3 get data

[0034]

[0035]

위에서 설명한 모든 기능들은 적절히 결합될 수 있다. 예를 들면, 디바이스는 도 5 및 도 6에 도시된 바와 같은 제 1 및 제 2 동작 모드로 동작하도록 설계될 수 있다. FIFO들은, 예를 들면 3개보다 많은 메모리 라인들의 프로그램 가능한 크기를 갖도록 설계될 수 있다. 따라서, 예를 들어 하나 이상의 채널들을 갖는 32-비트 폭의 오디오 데이터를 사용하는 다른 동작 모드들은 더 큰 FIFO 크기에 의해 이익을 얻을 수 있다.

[0036]

도 7은 다양한 실시예들에 따른, 4개의 채널을 공급하는 24-비트 동작 모드의 3개의 연속 판독 동작들과 2개의 채널을 공급하는 16-비트 동작 모드에서의 단일 판독 동작을 포함하는 일반적인 흐름도를 도시한다. 주의: 대기 상태 동안 각각의 신호(cmd\_spi\_go)가 어서트될 때, FIFO의 다음 값들을 로드할 수 있다. 이것은 다음 FIFO 값이 송신 레지스터에 미리-로딩될 수 있도록 하기 위한 것이다. 이전 rdy\_tx\_reg\_data의 마지막 비트가 시프트-아웃되기 전에 SPI 시프터에 데이터를 등록하도록, 대기 상태 동안 cmd\_spi\_go 신호가 어서트될 때 시스템 클럭 주기와 동일한 펄스에서 master\_latch\_next\_data 신호가 어서트된다. 스톨(stall) 상태 동안 활성화된 후 FSM을 통해 처음으로 pre\_tx\_reg 및 rdy\_tx\_reg는 cmd\_spi\_go 펄스에서 동일한 값으로 로딩된다.

[0037]

도 8은 일 실시예에 따른 24-비트 동작 모드에 대한 흐름도를 도시한다. 도 10은 일 실시예에 따른, FIFO 내의 단일 메모리 라인이 동작하는 16-비트 동작 모드에 대한 흐름도를 도시한다. 주의: load\_next\_value는 tx\_reg의 시프트 시작시 시스템 클럭과 동일한 펄스에서 어서트된다. 따라서 다음 fifo 값들은 tx\_reg의 시프트에 필요하기 전에 pre\_tx\_reg에 미리-로딩될 수 있다. cpu로부터의 고우(go) 신호가 스톨 상태 동안 하이(high)이거나 또는 자동 모드에 있고 SPI 시프터가 시프트-아웃될 tx\_reg 데이터의 마지막 비트를 시프트-아웃 완료하기 전일 때에는, 시스템 클럭과 동일한 펄스에서 mst\_latch\_next\_data가 어서트된다. 스톨 상태에서 활성화된 후 상태 머신을 통해 처음으로 pre\_tx\_reg 및 tx\_reg는 latch\_next\_data와 동일한 값으로 로딩된다.

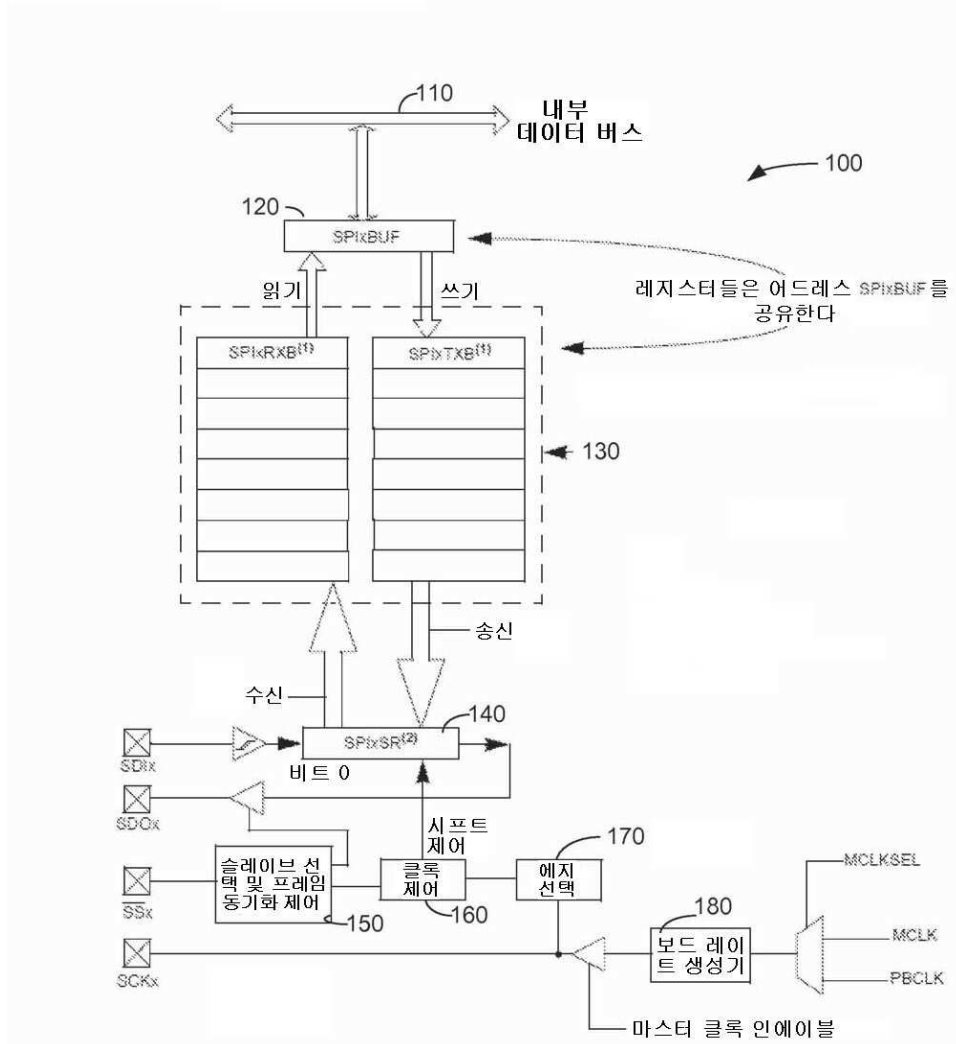
[0038]

도 9는 일 실시예에 따른, 4개의 채널들과 각각의 데이터 어셈블리 경로들을 갖는 24-비트 모드에 대한 흐름도를 도시한다. 주의: Load\_next\_value는 마스터 tx\_ms\_reg의 시프트 시작시 시스템 클럭과 동일한 펄스에서 어서트된다. 따라서 다음 fifo 값들은 tx\_xx\_reg의 시프트에 필요하기 전에 pre\_tx\_xx\_reg에 미리-로딩될 수 있다. cpu로부터의 go 신호가 스톨 상태 동안 하이이거나 또는 자동 모드에 있고 그리고 SPI 마스터 시프터가 시프트-아웃될 tx\_ms\_reg 데이터의 마지막 비트를 시프트-아웃 완료하기 전일 때에는, 시스템 클럭과 동일한 펄스에서 Latch\_next\_data가 어서트된다. 스톨 상태에서 활성화된 후 상태 머신을 통해 처음으로 pre\_tx\_xx\_reg 및 tx\_xx\_reg는 latch\_next\_data와 동일한 값으로 로딩된다.

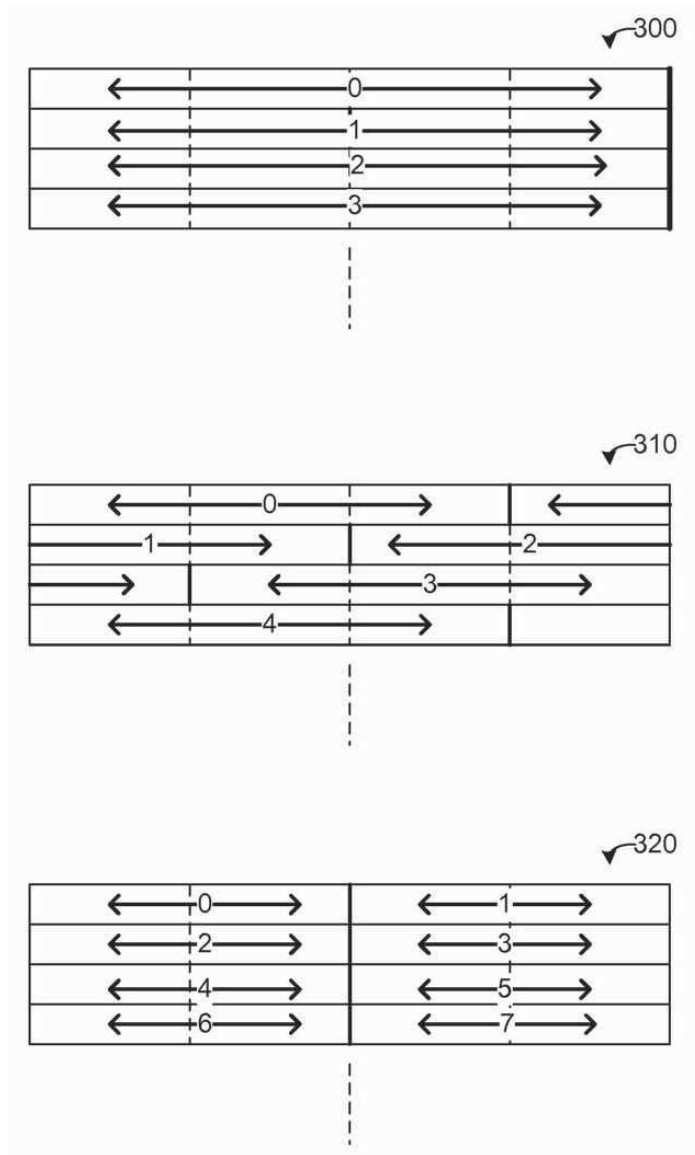
도면

도면1

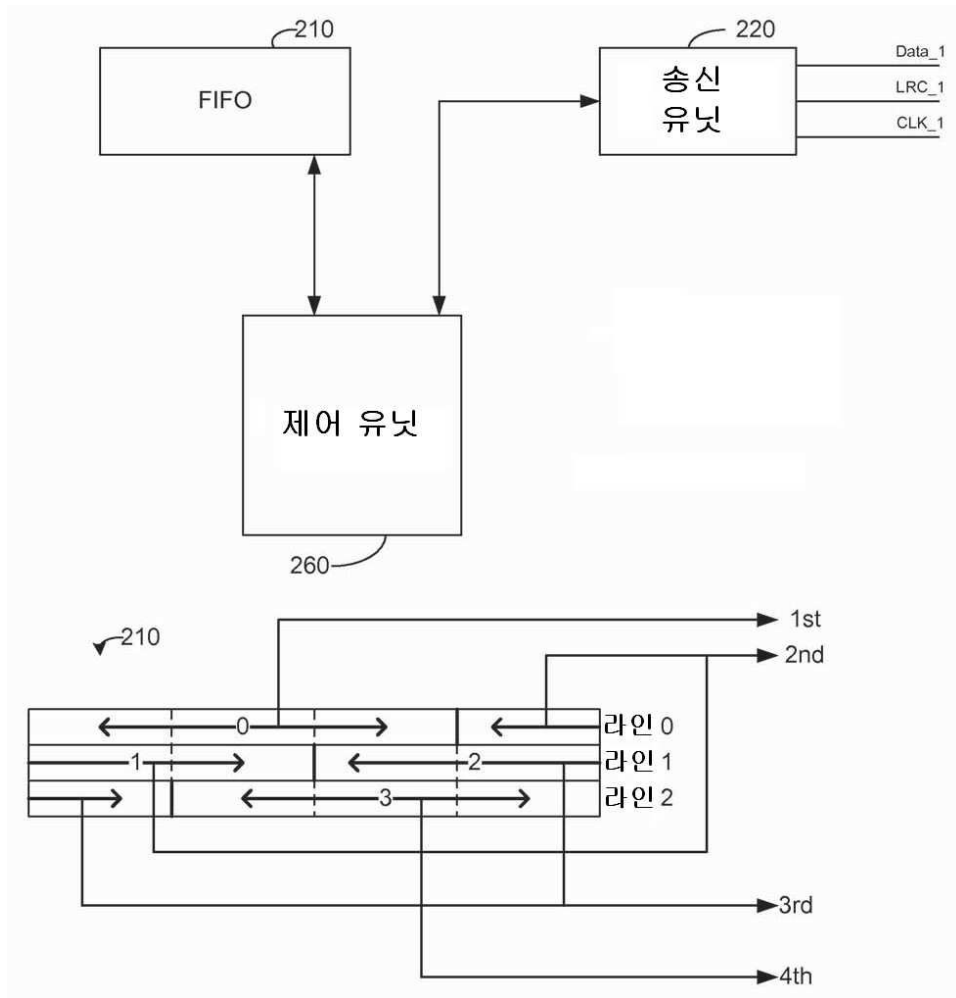
종래기술



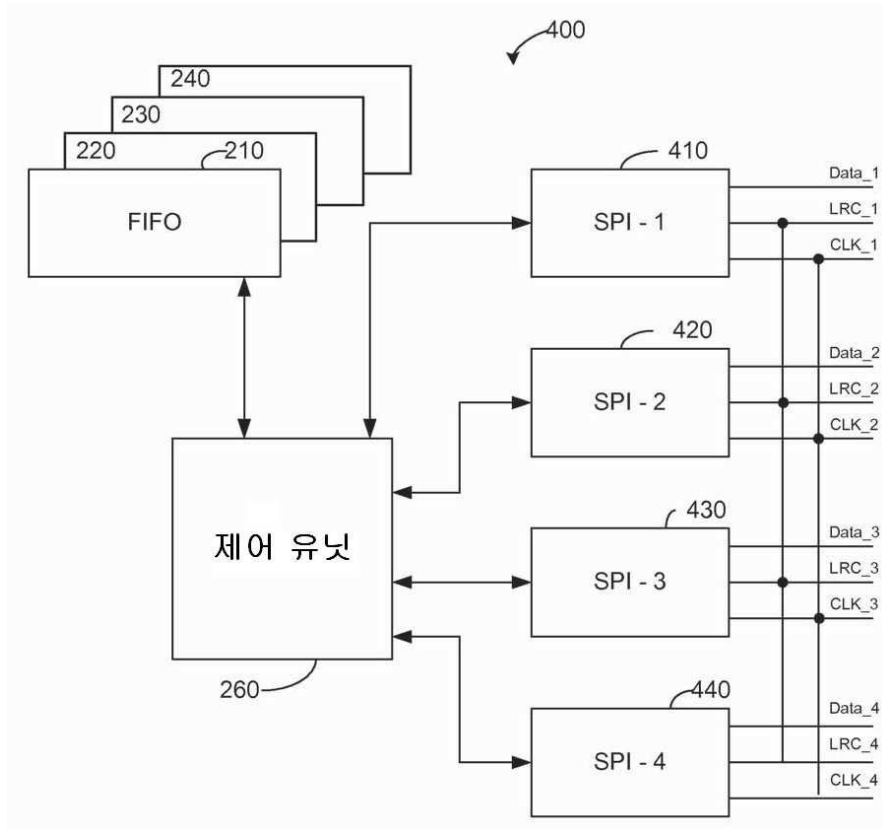
도면2



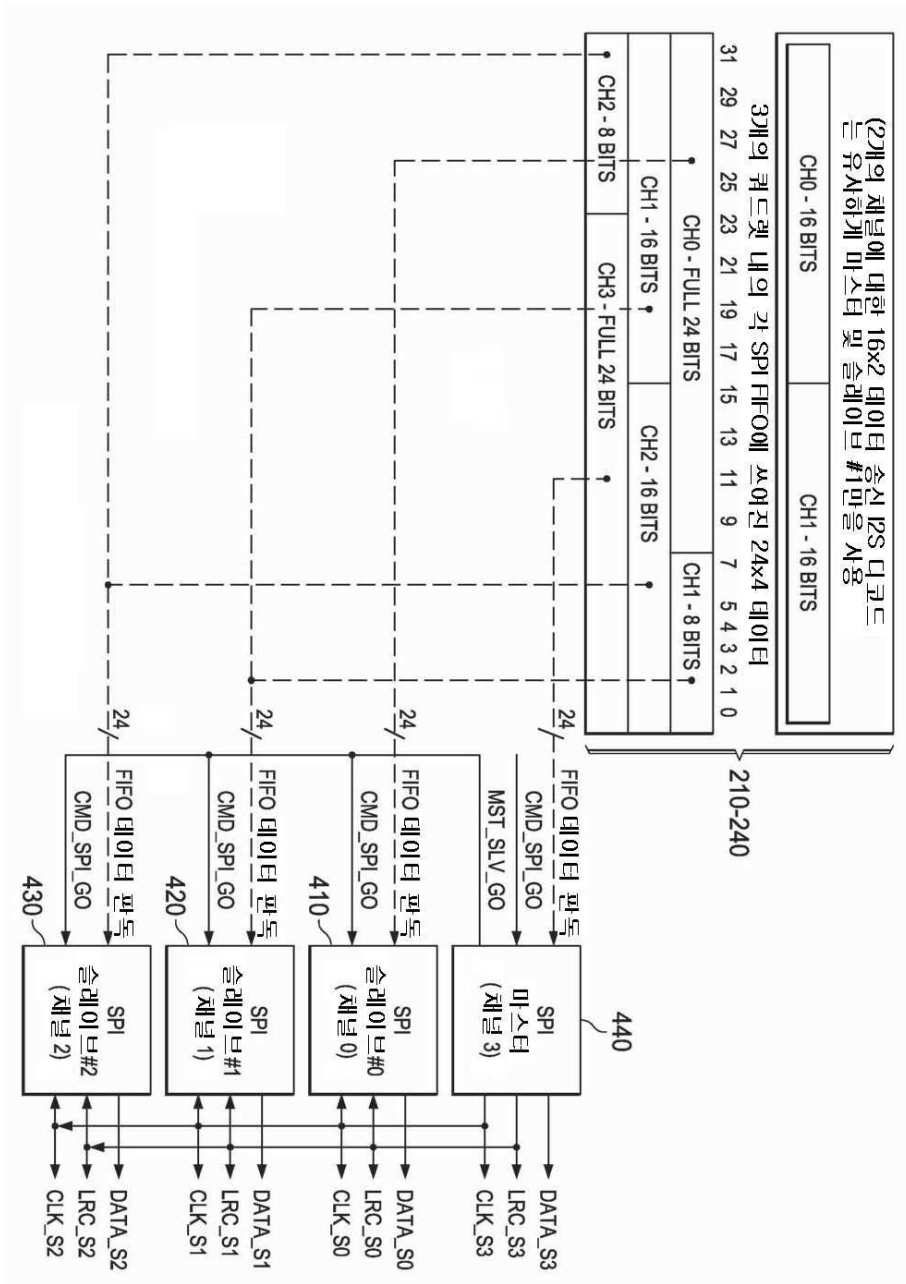
도면3



도면4

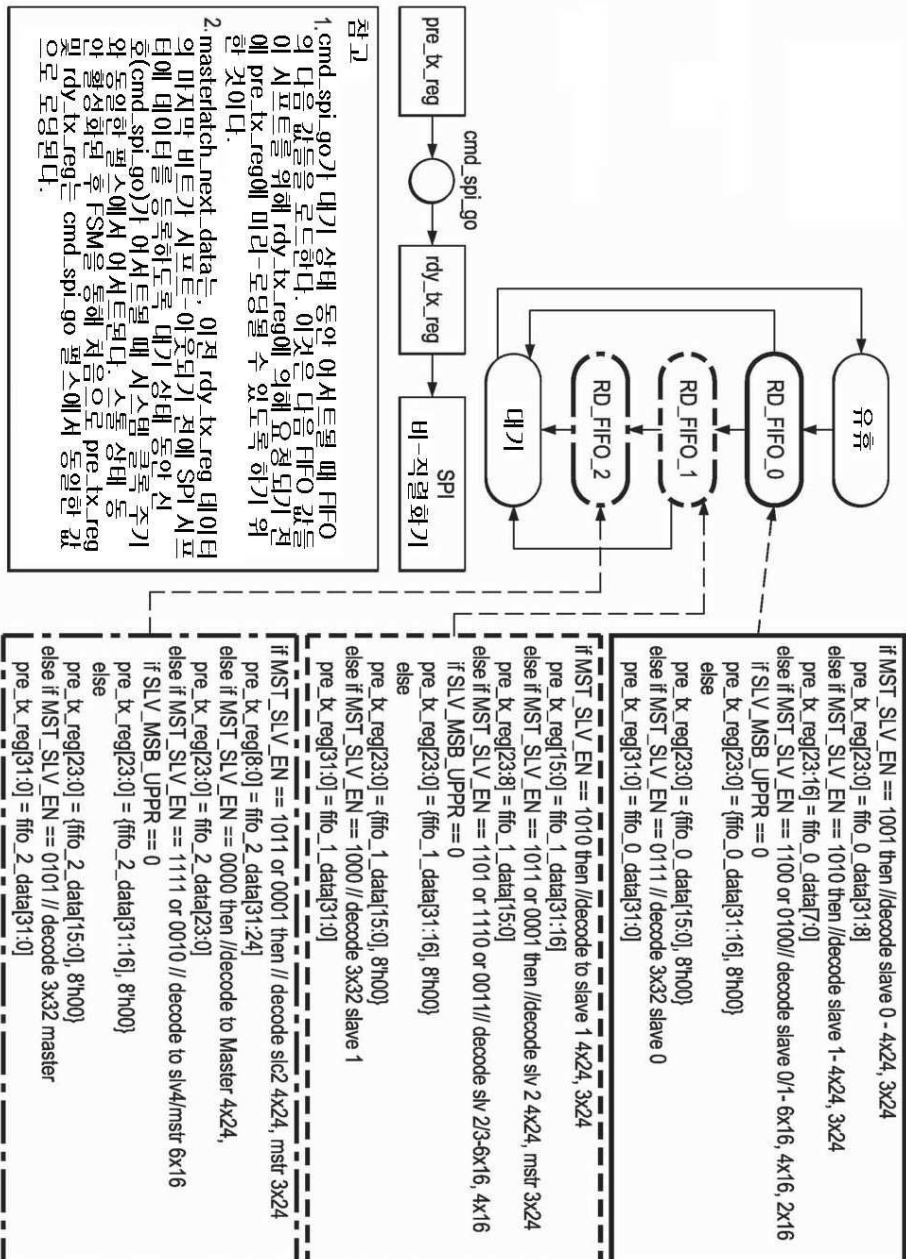


도면5



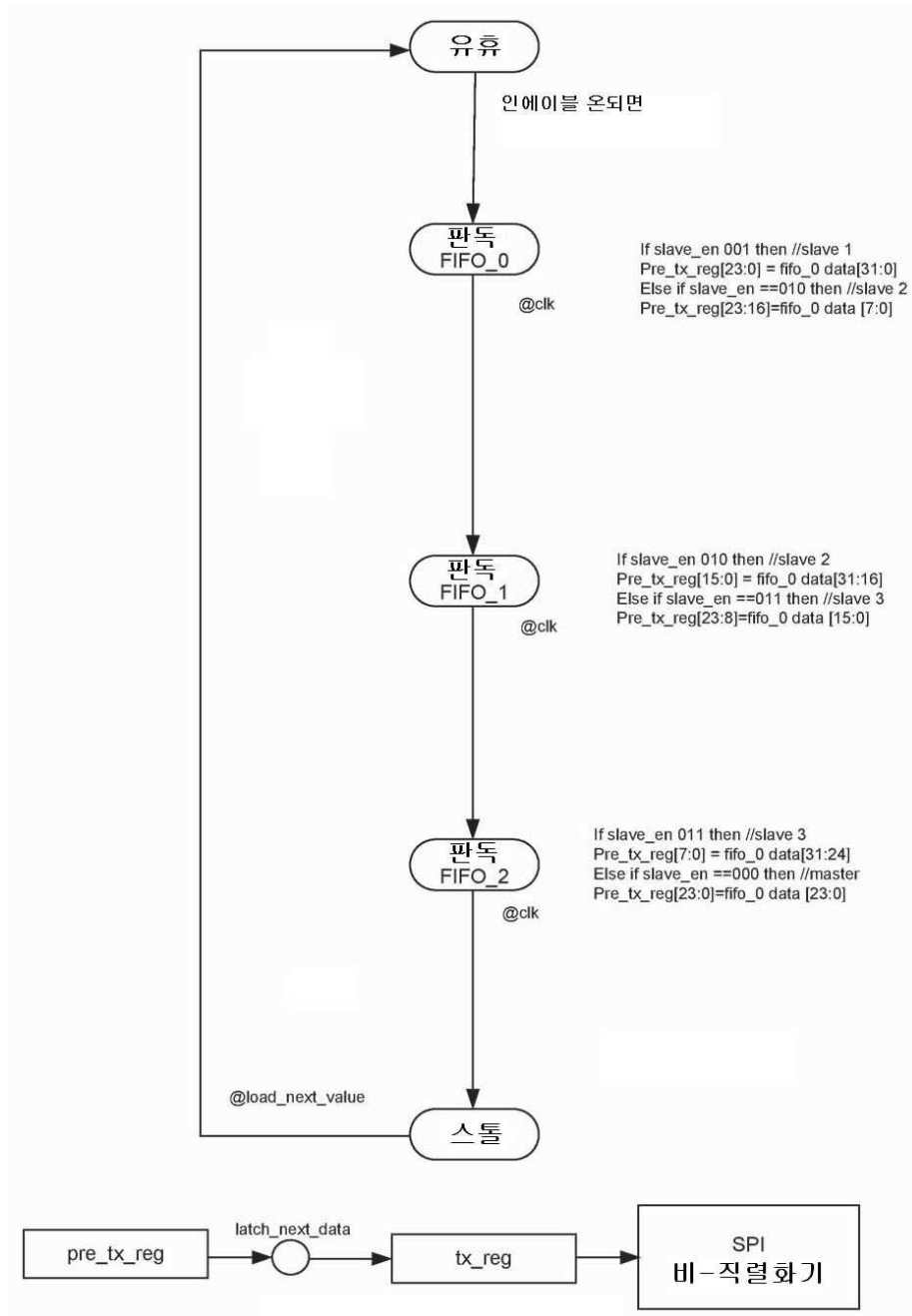


도면7



참고  
 1. cmd\_spi\_go가 대기 상태 동안 아서트될 때 FIFO의 다음 값을 로드한다. 이것은 다음 FIFO 값들이 시프트를 위해 rdy\_tx\_reg에 의해 요청되기 전에 pre\_tx\_reg에 비인-포딩될 수 있도록 하기 위한 것이다.  
 2. masterlatch\_next\_data는, 이전 rdy\_tx\_reg 데이터의 마지막 비트가 시프트-아웃되기 전에 SPI 시프트에 데이터를 등록하도록 대기 상태 동안 신호(cmd\_spi\_go)가 아서트된다. 스레드 상태 동안 동일한 펄스에서 아서트된다. 스레드 상태 동안 rdy\_tx\_reg는 cmd\_spi\_go 펄스에서 동일한 값으로 포딩된다.

도면8



도면9

