(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(51) 。Int. Cl. [6]
G06F 13/38

(45) 공고일자 2002년08월21일
(11) 등록번호 10-0330294
(24) 등록일자 2002년03월14일

(21) 출원번호 10-1994-0025279
(22) 출원일자 1994년10월04일

(65) 공개번호 특1995-0012237
(43) 공개일자 1995년05월16일

(30) 우선권주장 08/133,413 1993년10월05일 미국(US)

(73) 특허권자 모토로라 인코포레이티드
미국, 일리노이 60196, 샤움버그, 이스트 앨공퀸 로드 1303

(72) 발명자 제임스지.게이
미국,텍사스78660,플루게빌,스플릿오크1103

(74) 대리인 이병호

심사관 : 김재욱

(54) 데이터프로세서제어방법및버스프로토콜

요약

버스 프로토콜은 신호를 사용하여 버스 이송 종료(BTT*)와 버스 허가 포기 (BGR*)를 지정한다. BTT*신호는 현재 버스 소유권을 보유하고 있는 버스 마스터에 의해 어서트 되는 출력으로서 버스 이송이 완료되고 버스 소유권이 다른 버스 마스터로 이송될 수 있음을 다른 가능한 버스 마스터에 지시한다. BGR*신호는 버스 마스터에 대한 입력이다.

BGR*이 어서트될 때 버스 중재기/제어기는 버스 허가 신호의 디어서트 후 에 로크된 시퀀스에 관계없이 버스 마스터가 가능한 한 신속하게 포기되어야 함을 현재 버스 마스터에 통보한다. BGR*이 디어서트되면, 버스 중재기는 현재 버스 마스터에 편리한 시간에 버스 마스터가 포기될 수 있음을 현재 버스 마스터에 통보한다. 일반적으로 BGR*은 현재 버스 소유권 이송의 긴급성을 지시하는 한 비트이다.

대표도
도 1

명세서

도면의 간단한 설명

제 1 도는 버스 보유 기간의 종료를 지시하기 위하여 사용된 종래 버스 프로토콜의 타이밍도.

제 2 도는 본 발명에 따른 버스 보유기간의 종료를 지시하기 위하여 사용된 버스 프로토콜의 타이밍도.

제 3 도는 제 2 도의 버스 프로토콜을 구현하기 위하여 사용된 시스템의 블럭도.

제 4 도는 버스 소유권의 종료가 곤란하고 마스터가 버스를 독점하는 공지의 버스 프로토콜의 타이밍도.

제 5 도는 로크된 버스 이송을 적절히 재시도하기 위하여 비동기 버스 허가(BG*: Bus Grant) 제어가 요구되는 공지의 다른 버스 프로토콜의 타이밍도.

제 6 도는 본 발명에 따른 버스 프로토콜.

* 도면의 주요부분에 대한 부호의 설명 *

TS*: 이송 개시 D[31:0] : 데이터 비트

TA*: 이송 긍정 응답

PM_BR*: 주 마스터 버스 요청(primary master bus request)

PM_BG*: 주 마스터 버스 허가(primary master bus grant)

AH_BR*: 부 마스터 버스 요청

AM_BG*: 부 마스터 버스 허가

PM_BTT*: 주 마스터 버스 보유 기간 종료

AM_BTT*: 부 마스터 버스 보유 기간 종료

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

발명의 분야

본 발명은 일반적으로 데이터 프로세서에 관한 것이며, 특히 데이터 프로세서가 다른 회로와 통신하게 하는 버스 프로토콜에 관한 것이다.

발명의 배경

버스 프로토콜의 성능은 집적 회로 및/또는 집적 회로 시스템의 전체 성능에 직접적으로 영향을 미친다. 공지의 버스 프로토콜이 안고 있는 문제점으로서 현재의 마스터가 동기 버스 시스템에서 일련의 연속 로크된 전송 시퀀스(back-to-back locked transfer sequences)를 실행하고 있을 때 버스 소유권(bus ownership)이 다른 마스터로 이전되는 점(point)을 외부 버스 중재기(bus arbiter)가 언제나 결정할 수 있는 것은 아니라는 것이다. 로크된 전송의 발생은 통상 공지 시스템에서 LOCK 신호로 지시되는데, 이 신호는 현재 마스터가 전송 또는 전송의 시퀀스를 원자화(atomic)(즉, 분리될 수 없게)되도록 하고자 하는 것을 나타낸다. 그러나, 마스터가 몇몇의 이러한 로크된 원자화 시퀀스 연속적인 열을 가지며 외부 중재기가 이러한 원자화 시퀀스의 인접부를 인지한다면, 외부 중재기가 상기 마스터로부터 버스 소유

권을 이전받아 상기 시퀀스사이의 인터페이스에서 더 필요한 마스터에 버스 소유권을 부여하는 것이 완전히 가능하게 된다.

공지의 버스 프로토콜은 어떤 버스 마스터가 일련의 긴 연속 로크된 전송을 실행하고 있는 시간 동안 다른 버스 마스터의 기아(starvation)를 방지할 수 없다. 제 4 도는 버스 클럭을 개입시키지 않고 2 개의 연속적인 로크된 시퀀스를 실행하는 동기 버스의 동작을 예시한다. 제 4 도로부터 알 수 있듯이, 외부 중재기는 단일, 다수-전송 로크된 시퀀스와 버스 클럭을 개입시키지 않는 다중 시퀀스 사이의 차이점을 결정할 수 없다. 이러한 문제점은, 상기 중재기가 다른 버스 마스터로 하여금 버스를 갖게 할 필요가 있으나, 중단없이 상기 링크된 로크 시퀀스의 로크된 특성을 보전할 필요가 있을 때 발생한다. 로크된 시퀀스를 파괴하지 않기 위해, 상기 중재기는 현재 버스를 사용하는 마스터에 BG(버스 허가)를 니게이트하기전에 LOCK 의 니게이션을 대기하도록 강제된다. LOCK 가 시퀀스 사이에서 니게이트되지 않으므로, 현재 마스터가 연속 로크된 시퀀스를 실행시킴으로써 버스를 효과적으로 독점할 수 있다.

공지의 버스 프로토콜은 로크된 시퀀스의 최종 전송의 종료를 통해 LOCK 가어서트될 것을 요구한다. LOCK 는 슬레이브가 어떤 전송에 대한 전송 재시도의 필요성을 신호할 수 있기 때문에 로크된 시퀀스내에 모든 전송에 대해 어서트된 상태를 유지해야 한다.

한가지 공지된 버스 프로토콜은 로크된 시퀀스 사이에 하나 또는 두개의 데드 클럭(dead clocks)을 부가함으로써 위에서 기술한 문제점을 해결하려고 하였다. 이러한 데드 클럭은 로크된 시퀀스의 파괴를 종료하거나, 긴급한 요청을 갖는 다른 마스터의 버스 대기 시간을 증가시키고 성능을 저하시키는 바람직하지 않은 부작용을 갖는 것으로 나타났다. 위에서 기술한 문제점을 더 해결하기 위하여, 버스 프로토콜은 최종 로크된 버스 사이클 상태 지시자(LOCKE)를 결합하여 중재기로 하여금 로크된 시퀀스의 최종 전송 동안에 BG 를 마스터에 니게이트 하게 하고 상기 전송에 중재를 중첩시킨다. 그러나, 버스 사이클 재시도 동안에 버스 트랜잭션의 로크된 완전성을 보전하기 위하여 외부 버스 중재지가 BG 를 비동기적으로 리어서트해야 하는 경우가 있다. 제 5 도는 이러한 경우를 도시한다. 제 5 도에서 버스 중재기는 LOCKE 가 어서트되고 BR(버스 요청)이 니게이트될 때 버스 사이클 2(BC2)내의 BG 를 니케이트한다. 그러나, 전송 재시도가 신호되고(TRA 어서트) 로크된 시퀀스의 파괴를 막기 위하여 외부 버스 중재기는 BG 를 비동기적으로 리어서트해야 한다. 이는 재시도 신호(TRA)가 매우 빠른 속도로 샘플화되어야 하기 때문에 위에서 기술한 문제점을 해결하는 외부 버스 중재기의 설계를 매우 어렵게 만든다. 이는 매우 높은 프로세서 클럭 주파수 에서는 가장 불가능하며 완전 동기화 설계에서는 불가능하다.

공지의 중첩된 버스 중재 프로토콜이 안고 있는 또 하나의 문제점은 제 2 마스터가 버스 마스터에 대한 비동기, 자동-타이밍된 또는 틱에 기초한 내부 논리 클럭에 의존하지 않고 버스 사용을 개시하는 동안 현재 버스 마스터에 의한 버스 보유기간의 종료를 지시한다는 것이다. 이러한 문제점을 다루는 공지의 프로토콜은 버스 비지(Bus Busy) 프로토콜이다.

버스 비지(BB) 프로토콜을 사용하는 매우 높은 버스 주파수에서 문제들이 발생한다. 제 1 도는 공지 시스템에서 버스 비지 프로토콜의 동작을 도시한다. 통상적으로 시스템내에서 주 마스터 버스 비지(PM_BB^*) 신호와 부 버스 마스타 버스비지 (AM_BB^*)신호는 서로 결합되고 저항기로써 Vdd 로 상승될 것이다. 그러나, 이러한 신호들은 단절되어 버스 지배권 이전시에 버스 비지 신호 경쟁에 대한 가능성을 도시한다. 제 1 도의 시스템은 틱(tick)에 기초한 내부 클럭을 사용하고, 턴 온 타임과 턴 오프 타임을 데드 틱(dead tick)으로 분리함으로써 상술한 문제점을 해결하려고 한다. 틱은 완전한 클럭 주기의 일부이다. 데드 틱 방법은 낮은 버스 주파수에서 유용하지만, 버스된 버스 비지 신호의 좁은 틱 시간폭과 RC 상수 때문에 높은 버스 주파수에서는 유용하지 않다.

발명의 개요

본 발명은 전술한 결점들을 극복하고 다른 장점을 달성한다. 한 형태에서 본 발명은 버스 이전을 제어하는 버스 프로토콜 방법을 포함한다. 이 방법은 제 1 제어 신호를 어서트함으로써 버스 소유권을 마스터 회로에 허가하여 개시한다. 데이터는 제 2 제어 신호 어서션에 응답하여 마스터 회로와 마스터 회로 외부 장치 사이에서 전송된다. 제 3 제어 신호가 마스터 회로에 보내진다. 제 3 제어 신호는 제 1 제어 신호가 디어서트되어 버스 소유권이 포기되어야 함을 마스터 회로에 지시할 때 마스터 회로 버스 소유권이 현재 버스 사이클의 종료에서 포기되어야 하는지 또는 마스터 회로의 재량으로 종료되어야 하는지 여부를 마스터 회로에 지시한다. 제 4 신호가 어서트되어 마스터 회로가 버스 소유권을 종료했음을 마스터 회로에 지시한다. 제 3 신호 및 제 1 신호의 상태에 의해 결정되는 시간에 제 4 제어신호가 어서트된다.

다른 형태에서, 본 발명은 제 1 회로가 버스를 통해 제 2 회로와 인터페이스하게 하는 버스 프로토콜을 포함한다. 회로는 제 1 회로로부터 제 1 제어 신호를 출력으로서 제공하여 버스의 소유권이 요청됨을 지시한다. 수신용 회로는 제 1 회로에 버스 소유권이 부여되었는지 여부를 지시하는데 이용되는 제 1 회로에 대한 입력으로서 제 2 제어 신호를 수신한다. 공급용 회로는 제 1 회로로부터의 출력으로서 제 3 제어 신호를 발생하며, 여기서 제 1 제어 회로가 버스 소유권을 포기할 준비가 될 때만 제 3 제어 신호가 어서트된다.

또 다른 형태에서는 본 발명은 제 1 회로에 대한 입력 으로서 제어 신호를 수신하는 회로를 사용하여 버스를 통해 제 1 회로가 제 2 회로와 인터페이스하게 하는 버스 프로토콜을 포함한다. 제어 신호는 제 1 논리 상태로 세트되어, 제 1 회로에 의해 실행되는 현재 버스 사이클의 종료에서 제 1 회로가 버스 소유권을 포기해야 함을 지시한다. 제어 신호는 제 2 논리 상태로 세트되어, 제 1 회로에 의해 결정된 편리한 시간에 제 1 회로가 버스 소유권을 포기함을 지시한다.

본 발명은 첨부 도면과 함께 이하의 상세한 설명에 의해 명백히 이해될 것이다.

간단하고 명료하게 예시하기 위하여 도면에 예시된 소자들은 실제 크기로 도시되지 않는다. 예컨대, 몇몇 소자들의 크기는 명료하게 하기 위하여 다른 소자에 비해 상대적으로 크게 도시되었다. 또한 참조 번호는 대응 또는 유사한 소자를 나타내기 위해 도면들에서 반복된다.

양호한 실시예의 설명

여기에서 기술된 버스 동작 방법 및 버스 프로토콜은 제 2 마스터가 버스에 대한 비동기, 자기 시간, 틱(tick)에 기초한 내부 논리에 의존하지 않고 버스의 사용을 개시하는 동안 현재 버스 마스터에 의한 버스 보유 기간(bus tenure)의 종료를 지시하는 중첩된 버스 중재 시스템의 문제점을 해결한다. 제 1 도는 공지된 버스 프로토콜 및 방법의 타이밍 도를 예시한다. 제 1 도에서 신호는 다음과 같은 약자를 사용한다. CLK = 클럭(clock); TS * = 이송 개시(transfer start); R/W* = 판독/기록(read/write); Address & Attributes = 번지 비트 및 버스 속성 비트 (Address bits and bus attribute bits); D[31:0] = 데이터 비트(data bits); TA * = 이송 긍정 응답(transfer acknowledge); PM_BR * = 주 마스터 버스 요청 (primary master bus request); PM_BG * = 주 마스터 버스 허가 (primary master bus grant); PM_BB* = 주 마스터 버스 비지 (primary master bus busy); AM_BR * = 부 버스 마스터 요청; AM_BG* = 부 마스터 버스 허가 신호 명칭에서 |*| 기호는 신호가 액티브로우(active low) 신호임을 나타낸다.

제 1 도는 매우 높은 버스 주파수에서 Bus Busy(BB *) 신호를 이용하는 종래프로토콜에 발생하는 문제점을 예시한다. 통상적으로 시스템에서, 주 버스 마스터 Bus Busy(PM_BB *)와 부 버스 마스터 Bus Busy(AM_BB *) 신호는 서로 결합되고 저항기에 의해 Vdd 로 상승된다. 그러나, 이러한 신호는 버스 소유권 이송 시간에 Bus Busy(BB *) 신호 경합의 가능성을 나타내기 위해 제 1 도에서 분리되어 예시된다. 주 마스터는 틱에 기초한 내부 클럭을 사용하고, 버스 비지 신호(BB*)의 턴 온 시간과 턴 오프 시간을 데드 틱(dead tick)으로 분리 함으로써 BB *경쟁의 문제점을 해결하려한다. 이는 좁은 틱 시간 폭과 전송된 Bus Busy(BB *) 신호의 RC 상수 때문에 낮은 버스 주파수에서 효과가 있으나 높은 버스 주파수에서는 효과가 없다. 제 1 도에서, 영역 A 와 B 는 높은 주파수에서 하나의 BB *신호(PM_BB* 또는 AM_BB*)가 논리 1 을 유도하고 다른 BB*신호는 논리 0 을 유도하는 시간 영역을 도시한다. 이러한 버스 경쟁은 유용하지 못하다.

제 2 도는 본 발명에 따른 버스 보유기간 종료(BTT*: Bus Tenure Terminated) 프로토콜의 동작을 개시한다. 통상 주 버스 마스터 버스 보유기간 종료(PM_BTT*)와 부 버스 마스터 버스 보유기간 종료(AM_BTT*) 신호는 서로 결합되고 저항기로써 Vdd(공급 전압)로 상승시킨다. 그러나, 이러한 신호는 버스 소유권 이송 시간에 버스 보유기간 종료 신호 경쟁의 가능성이 없음을 예시하기 위하여 제 1 도에서는 분리되었다. 또한, 버스 보유기간 종료(BTT*) 신호의 타이밍은 버스 마스터의 비동기, 자기-시간, 틱에 기초한 내부 논리 클럭을 필요로 하지 않고 시스템클럭과 전적으로 동기화될 수 있다.

제 2 도를 참조하면, 용어 " 버스 사이클(bus cycle)과 " 이송 사이클 (transfer cycle)은 이송 개시(TS *: transfer start)가 어서트(asserted)될 때의 클럭 에지로부터 마지막 이송의 최종 종료 긍정응답 신호(TA *, TEA*또는 TRA*)가 샘플화될 때의 클럭 에지까지의 완전한 하나의 버스 동작 사이클을 의미한다. 한 버스 사이클내에서는 최대한 몇번의 데이터 이송이 일어날 수 있다. 용어 " 이송 (transfer)" 은 그 자체로 데이터가 프로세서로 이송되거나 또는 그로부터 이송될 때의 단일 클럭 주기를 일컫는다.

제 2 도에서 CLK 는 시스템 클럭을 나타낸다. TS*신호는 이송 개시 신호이다. 이송 개시(TS*) 신호는 번지선이 안정되는 클럭 사이클 동안 주 마스터에 의해 어서트되고, 다음 클럭 사이클에서 니게이트(negated)된다. TS *신호는 주 마스터가 버스의 소유권을 보유할 때 주 마스터에 대한 버스 사이클의 개시를 나타낸다. 다른 기능을 수행할 수도 있는 TS*신호는 주 마스터가 버스 마스터가 아닐 경우 주 마스터에 의해 검사되어 시스템내의 버스 지배권에 대한 내부 상태 정보를 유지한다. 버스상에서 버스 사이클을 실행하는 모든 마스터들은 그들의 TS*신호들이 함께 버스 상에서 이송되어야 한다. 주 마스터가 버스 마스터가 아닐 때, TS*는 주 마스터에 의해 통상적으로 3-상태(tri-state)가 된다.

제 2 도에서 R/W*신호는 주 마스터가 버스 소유권을 보유할 때 주 마스터에 의해 유도된다. 이 R/W*신호는 뒤이은 데이터 전송의 방향(마스터 외부로부터 판독 또는 마스터 외부로의 기록)을 나타낸다. R/W*신호는 판독 버스 사이클 동안에 하이(high)로 어서트되고 기록 버스 사이클 동안에 로드(low)로 어서트된다. 68060 이 버스 마스터가 아닐 때, 이 신호는 3-상태가 된다.

제 2 도에서 신호 TA*(또는 이송 에러 긍정응답 [TEA*]이나 또는 이송 재시도 긍정응답[TRA*]과 같은 등가 신호)가 버스 사이클의 종료를 지시하는 데 사용된다. 특별히, 주 마스터가 버스 마스터일 때(즉, 버스의 소유권을 보유할 때), TA*신호는 버스 사이클에서 전송되는 데이터가 유효하고 버스 판독 동작 동안에 샘플화된 준비가 되는 클럭에서 버스 마스터에 접속된 현재 슬레이브에 의해 어서트된다. 대안적 으로, TA*신호는 주 마스터로부터 전송되는 데이터가 버스 기록 동작 동안에 슬레이브에 의해 수용되는 클럭에서 현재 슬레이브에 의해 어서트된다. 주 마스터가 이송 긍정응답(TA*)으로 이전을 성공적으로 수용하기 위하여 TRA*와 TEA*는 TA*가 어스트될 때 모두 니게이트되어야 한다. 환언하면, 재시도 및 에러 버스 이송 신호는 TA*신호 보다 앞선다.

제 2 도에서, D[31:0] 신호는 현재 트랜잭션에 대한 데이터를 운반하는 양방향 신호이다. 버스트 버스 사이클(burst bus cycle) 동안에, 데이터 라인은 시분할 다중화되어 전체 버스트 요청의 모든 부분을 전승한다. 제 2 도에서, 번지 신호는 주 마스터가 버스 소유권을 보유할 때 주 마스터에 의해 출력된다. 번지 비트는 버스 이송의 제 1 항목의 번지를 나타낸다.

제 2 도에서, 버스 사이클 속성 신호는 버스상에서 일어나는 전송의 형태를 지시하는데 사용된다. 예를 들면, 이러한 신호는 16 비트 또는 32 비트 전송을 나타낸다. 유저 또는 슈퍼바이저 모드 액세스는 특정될 수 있다. 캐쉬(Caches)가 적절히 제어될 수 있다. 메모리 운용 장치가 액세스될 수 있다. 데이터 전송 크기가 결정될 수 있다. 전송된 긴 워드(longword) 내에서 바이트 선택이 일어날 수 있다. 버스 로킹(bus lockiug)이 결정될 수 있다. 일반적으로, 많은 전송 속성 기능 및 신호가 종래 기술에 존재하며, 그중 전부 또는 일부가 제 2 도에 사용될 수 있다.

제 2 도의 PM_BR*신호는 주 마스터 또는 주 프로세서에 대한 버스 요청 신호이다. PM_BR*신호는 주 마스터로부터 외부 버스 중재기로 전송되고 언제든지 어서트되거나 니게이트 될 수 있다. PM_BR*은 주 마스터에 의해 어서트되어 버스 사용의 필요성을 나타내며, 주 마스터가 더 이상 버스를 필요로 하지 않을 때까지 어서트된 상태를 유지한다. 이러한 어서션(assertion) 시간은 주 마스터가 버스 마스터일 때(즉, 버스의 소유권을 보유할 때)의 시간과 주 마스터가 버스 마스터가 아닐 때(즉, 버스의 소유권을 보유하지 않을 때)의 시간을 포함한다.

주 마스터에 의한 버스 사용의 필요성은 버스 중재기가 버스를 PM_BG* 신호를 거쳐 주 마스터에 허가하기 전에 내부 상태의 변화에 기인하여 사라진다. 이것은 PM_ER* 이 어떤 시간 주기 동안에 어서트되고 PM_BR* 가 외부 버스 중재기에 의해 어서트되기 전에 니게이트될 경우에 가능하다. 그래서, 외부 버스 중재기는 버스 마스터가 PM_BG* 를 주 마스터에 어스트할 때 주 마스터가 버스의 사용을 요청하지만 실제로는 버스 지배권을 넘겨받지는 못하는 경우를 다루어야 한다. 이러한 경우는 PM_BG* 가 외부 버스 중재기에 의해 어스트된 후 주 마스터가 PM_BR* 을 니게이트하고 다음 클럭 에지상에서 TS* 를 어서트하지 않음을 인지함으로써 검출될 수있다. 주의할 사항으로서, 위와 같은 경우는 PM_BG* 가 주 마스터로 어서트되고 주 마스터가 PM_BR* 를 여전히 어서트 하지만 TS* 를 어서트하지 않은 경우가 아니라는 것이다. 주 마스터가 PM_BG* 를 수신한 후 PM_BR* 이 어서트된 상태에서 주 마스터에 의한 TS* 어서션이 지연되는 것은 명령 버스 요청 동작을 선취한 오퍼랜드 버스동작에 의한 특별한 내부 조건에 기인한다. 이러한 경우에, 주 마스터는 PM_BR* 이 어서트된 상태인 한 여전히 버스를 사용하려고 한다.

제 2 도는 PM_BG* 신호를 예시한다. PM_BG* 신호는 외부 중재기에 의해 주 마스터에 부여된 버스 허가 신호로서 현재 마스터가 현재 버스 사이클을 종료하고 제어 신호로 그 버스를 포기하자 마자 주 마스터가 그 버스를 제어할 수 있음을 나타낸다. 버스를 포기시키는 제어 신호는 이하에서 기술된다. PM_BG* 가 중재기에 의해 니게이트될 때, 버스 사이클의 로크된 시퀀스가 니게이트된 BGR* (Bus Grant Relinquish)과 함께 진행하지 않는다면 주 마스터는 현재 버스 사이클이 완료되자마자 제어 신호를 거쳐 버스를 포기한다. BGR* 신호는 다음 그림과 함께 이하에서 기술된다. 버스 사이클의 로크된 시퀀스가 니게이트된 BGR* 과 함께 진행할 때 주 마스터는 원자화 로크된 버스 사이클의 시퀀스를 완료하고, 주 마스터가 PM_BTT* (전송 종료 제어 신호)를 어서트하고 PM_BB* (제 2 도에는 예시되어 있지 않으나 제 1 도에 예시된 버스 비지 신호)를 니게이트함으로써 버스를 포기함을 지시한다. 일반적으로 PM_BTT* 신호만이 버스 사이클의 종료를 결정하는데 필요하며 PM_BB* 가 프로세서 계열 호환성에 제공된다.

BTT* 신호가 강조되기 때문에 BGR* 신호는 제 2 도에 예시되어 있지 않다. BGR* 신호는 여기에서 간략하게 설명되고 이후에 상세히 설명된다. 제 2 도의 BG* 신호의 동작을 잘 이해할 수 있도록 하기 위해 여기에 BGR* 을 간단히 기술한다. BGR* (버스 허가 포기 제어 : Bus Grant Relinquish Control) 신호는 PM_BG* 에 대한 수식자(qualifier)로서, PM_BG* 가 니게이트될 때 버스 소유권을 포기하는 필요도를 주 마스터에 지시한다. BGR* 신호는 BG* 가 니게이트될 때 원자화 로크된 버스 시퀀스 동안에 주 마스터가 동작하는 방식에 영향을 미친다. BGR* 의 어서션은 비록 주 마스터가 일련의 버스 사이클을 서로 로크시키려 하더라도 BG* 가 니게이트될 때 현재 버스 사이클의 최종 이송에서 주 마스터가 버스 소유권을 포기하여야 함을 주 마스터에 지시한다. BG* 가 니게이트될 때 니게이트된 BGR* 은 로크된 버스 사이클의 전체 열이 종료되어 원자화된 시퀀스의 파괴가 발생하지 않음을 보장한 후에만 로크된 버스 조건(LOCK* 제어 신호가 어서트됨: 도시 안됨) 아래에서 주 마스터 버스 소유권이 포기되어야 함을 주 마스터에 지시한다.

환언하면, BGR* 이 어서트될 때, PM_BG* 이 니게이트되면 주 마스터는 로크된 시퀀스 등에 관계없이 가능한 신속하게 버스를 포기한다. 니게이트되는 BGR* 신호는 버스 종료의 긴급성을 덜 나타내고, 주 마스터는 적절한 종료 시간이 결과적으로 발생할 때까지 버스상에서 전송을 계속할 수 있다.

PM_BTT* 신호는 제 2 도에 예시되어 있다. PM_BTT* 신호는 한 클럭 주기동안 어서트되어 외부 중재기가 PM_BG* 를 프로세서에 니게이트한 후에 주 마스터가 버스보유기간을 포기하였음을 시스템내의 다른 마스터에 지시한다. PM-BTT* 신호는 어서션 후 즉시 한 클럭 주기동안 니게이트된다. 다른 모든 시간에는 PM_BTT* 는 3상태이다. 제 2 도는 주 마스터와 부 마스터가 출력 핀들(output pins)을 구동하는 방법을 도시한다. 제 2 도의 출력 핀들중의 일부는 서로 결합되어 하나의 통합된 신호를 생성한다. 예를 들면, PM_BTT* 는 통상 시스템내에서 AM_BTT* 와 다른 BTT* 핀들에 결합된다. 주 마스터는 버스 마스터가 아닐 때 PM_BTT* 를 입력으로 샘플화 하여 내부 상태를 유지하고, 언제 버스 지배권을 넘겨받을 것인지를 계산해 둔다. 버스 마스터는 PM_BTT* 를 구동하여 주 마스터가 버스 소유권을 종료하고 있음을 시스팀내의 다른 모든 부 마스터에 지시한다. 그러므로, 프로세서에 대한 BTT* 신호는 때때로 다른 마스터가 버스의 소유권을 포기하고 있음을 신호하는 입력이다. 또는 BTT* 는 프로세서에 의해 버스 소유권이 종료되고 있음을 지시하는 출력으로서 사용된다.

이러한 내부 상태 정보를 적절히 유지하기 위하여 버스상의 모든 마스터는 주 마스터와 다른 버스 마스터가 결합 TS*_BTT* 쌍을 기억할 수 있도록 서로 버스된 그 TS* 신호들이 함께 버스로 전송되어야 한다. 예를 들면, BTT* 신호가 발생하고 그 후에 TS* 신호가 어서트되지 않으면 버스는 현재 소유권을 잃는다. BTT* 신호가 발생하고 그후에 TS* 신호가 어서트되면 버스는 다른 마스터에 의해 사용된다(즉, 소유된다). 그러므로, TS* 신호와 BTT* 신호 양자를 감시하는 것이 버스 상태를 결정하는데 중요하다.

제 2 도의 프로토콜에서, 주 마스터는 제 1 도의 프로세서에 대한 호환 신호를 제공하기 위하여 BB* 신호 (제 2 도에 도시안됨)를 공급한다. 그러나, 양호한 형태에서 제 2 도의 BTT* 신호 및 프로토콜이나 제 1 도의 BB* 신호 및 프로토콜 이선택적으로 사용되며, 둘 다 사용되어서는 안 된다. BTT* 신호는 버스 경합이 감소되어 더 큰 유연성을 제공하여 바람직하다. 사용되지 않은 신호(바람직하게는 BB*)는 저항기를 통해 니게이트된 상태까지 계속적으로 상승되어야 한다. 여기에서 논의된 것처럼 BTT* 프로토콜은 BB* 프로토콜 보다 높은 동작 주파수와 최대 버스 속도에서 고성능을 달성한다.

제 2 도에서, AM_BTT* 신호, AM-BR* 신호, AM_BG* 신호는 각각 PM_BTT*, PM_BR*, PM_BG* 와 동일한 방식으로 기능한다. 유일한 차이점은 한 집단의 신호들이 주 마스터용 으로 기능하고 반면에 다른 집단의 신호들은 부 마스터용으로 기능한다는 것이다.

제 2 도에서, BTT* (Bus Tenure Terminated) 프로토콜 개념은 제 1 도의 BB * 프로토콜과는 다르다. BTT* 프로토콜은 버스 사용을 일정하게 지시하지는 않지만 마스터가 버스를 사용하는 시간을 단순히 플래그한다. BTT* 신호는 다른 마스터가 버스 소유권을 이전받기에 안전한 때를 검출하는 데 필요한 신호이다. BTT* 는 마스터가 버스를 사용하고 있을 때 항상 3-상태이며, 한 클럭 주기 동안에 어서트되고 한 클럭 주기 동안에 니게이트된 다음 다른 버스 종료 시점까지 다시 3-상태가 된다. 버스 사용을 대기하는 마스터는 이전의 마스터가 버스 사용을 종료함을 인지하는 BTT* 의 한 클럭 어서션을 검출하기만 하면 된다. BTT* 는 제 1 도의 BB* 신호와 달리 하나의 클럭 동안 내내 하이로 동기적으로 구동되어 신호가 3-상태로 될때 발생하는 긴 RC 지연을 방지한다는 것에 유의한다. 제 2 도에서, 하나의 마스터가 BTT* 를 구동하여 니게이트시킨 다음 3-상태로 한 시간과 다른 마스터가 BTT* 를 어서트하기 시작하는 시간 사이에는 항상 최소한 하나의 클럭이 존재하고, 따라서 제 1 도의 BB* 달리 경쟁이 방지된다는 것에도 유의해야 한다.

일반적으로, BTT* 프로토콜 개념에 의해 버스 마스터의 비동기, 자기-타이밍, 틱에 기초한 내부 논리에 의존하지 않고 동기 버스 구조에 대하여 진정한 중첩된 버스 중재가 발생할 수 있게 된다. BTT* 프로토콜은 BB* 프로토콜이 매우 높은 버스 주파수에서 가지는 경쟁의 문제점을 해결한다.

제 3 도는 제 2 도의 BTT* 프로토콜을 구현하는데 이용되는 시스템을 예시한다. 제 3 도는 몇 개의 실선을 사용하여 상호 접속을 예시한다. 제 3 도의 파선은 제 3 도의 요소의 상호 접속 구성의 변화 또는 상호 접속에의 부가를 나타낸다. 제 3 도는 주 마스터(10), 대체 마스터(12), 버스 중재기(14)를 가지며 특정 방식으로 상호 접속된다. 제 3 도는 제어 버스, 데이터 버스와 번지 버스를 예시한다. 제어 버스는 2 이상의 제어 신호를 상호 접속하고 2 이상의 제어 신호 또는 이와 유사한 신호를 경로 지정한다. 신호 BR*, BGR*, BG*, LOCK*, LOCKE* 는 비록 이러한 신호들이 제어 버스에 접속되는 것으로 제 3 도에 예시되지는 않았지만 제어 버스와 선택적으로 접속될 수 있다. 제 3 도에서 버스들은 각 버스 마스터에 대한 각 요청/허가 라인을 사용하여 중앙 버스 중재기(14)에 의해 중재된다(arbitrated). 중재 프로토콜(즉, 어느 프로세서가 버스 소유권을 보유하는가를 결정하는 규약)은 버스 이전 활동과 동시에 동작할 수 있어 버스 소유권 이전이 큰 성능 저하없이 가능하게 된다.

제 3 도는 버스 중재 신호가 다중 마스터 시스템에서 버스 중재기/제어기로 경로지정(route)될 수 있는 방법을 기술한 한 블럭도의 일예를 예시한다. 제 3 도는 주 마스터 버스 요청(PM_BR*) 입력과 부 마스터 버스 요청(AM_BR*) 입력을 예시한다. BR* 신호는 버스 중재기(14)에 접속된다. PM_BGR* (주 마스터 버스 허가 포기)과 AM_BGR* (부 마스터 버스 허가 포기)는 상기 중재기(14)에 접속되어 있는 것으로 예시된다. PM_BG* (버스 허가)와 AM_BG* (버스 허가)는 상기 중재기(14)에 접속되어 있는 것으로 예시된다. LOCK* 와 LOCKE* 신호는 마스터(10, 12)에 의해 출력되고 상기 중재기(14)에 접속된다. 제어 신호 BTT* (버스 이전 종료), TS* (이전 개시)와 BB* (버스 비지)는 마스터(10. 12) 사이에서 접속되고 제어 버스에 접속된다. 데이터 라인은 각 마스터(10, 12)로부터 데이터 버스까지 접속되고 번지선들은 각 마스터(10, 12)로부터 어드레스 버스까지 접속된다. R/W* (판독/기록), TA* (이송 긍정응답), TEA* (이송 에러 긍정 응답), TRA* (이송 재시도 긍정응답) 다른 버스 제어 신호와 같은 다른 모든 제어 신호는 제 3 도에 도시된 바와 같이 제어 버스에 접속된다. TS* 는 프로세서가 버스 마스터가 아닐 때 검사되어 버스 지배권에 대한 내부상태를 유지한다. 버스상에서 버스 사이클을 실행 할 수 있는 모든 마스터는 TS* 신호들이 양호한 형태로 함께 버스 이송되고 접속되도록 해야 한다.

제 3 도는 그 소자들에 대한 모든 클럭을 발생시키는 클럭 발생기 회로(16)를 예시한다. 하나의 클럭 또는 다수의 클럭이 사용될 수 있다.

제 3 도 시스템내의 중재는 이송 개시(TS*)로 부터 최종 이송 종료(TA*, TRA*, TEA*) 까지의 버스 사이클이 비록 너무 빨리 종료되더라도 분리될 수 없는 원자화된 실재(entity)라는 개념과 현재 버스 소유권에 기초한다. 여기에서 사용되는 " 버스" 는 하나 또는 그 이상의 데이터 버스, 번지 버스, 소유권이 할당될 수 있는 어떤 제어 신호를 총칭한다는 것에 유의한다. 버스가 주 마스터(10)에 의해 현재 소유되면, 주 마스터가 BTT*를 어서트할 때(그렇지 않으면 BB*를 니게이트할때) 최종 버스 사이클을 완료한 후에만 다른 마스터(즉, 부 마스터)에 의해 소유될 수 있다.

버스가 주 마스터(10)에 의해 현재 소유되지 않으면, 주 마스터(10)는 버스를 필요로 하자마자 BR*신호를 어서트한다. BG*의 어서션이 버스 중재기(14)로부터 수신되고 버스 이송된 BTT*의 한 클럭 주기 어서션이 검출(또는, 대안적으로는 BB*의 니게이션이 상승 클럭 에지에서 검출) 되자마자 버스 지배권이 인수되며, 이전의 마스터(예컨대, 비록 다른 마스터가 시스템내에 있더라도 부 마스터(12))가 그 보유기간을 종료하고 버스를 포기함을 지시한다. BG*가 수신될 때 주 마스터 (10)가 여전히 버스를 사용할 필요가 있는 경우, 주 마스터는 버스 지배권을 인수하고, TS*를 어서트하고 버스 사이클을 개시한다. 주 마스터(10)가 시간상의 특정한 순간에 버스를 더 이상 사용할 필요가 없는 경우에 주 마스터(10)는 내부 상태에 기인하여 BR*신호를 니게이트 한다. 주 마스터(10)는 BR*신호를 니게이트하고 동시에 버스가 한 버스 사이클 동안에만 필요한 경우에 TS*신호를 어서트한다. BR*은 하나 이상의 사이클이 주 마스터(10)에 의해 필요한 경우에 더 긴 시간 주기동안 보유된다.

BTT*신호(또는 BB*)는 시스템내의 모든 부 마스터에 접속되어 주 마스터(10) 에 의한 버스 보유기간의 종료를 알린다. BTT*신호는 버스에 대한 소유권을 상실하고 버스 사용을 종료할 준비가 된 후에 주 마스터(10)에 의해 어서트 된다. 최종 버스 사이클의 최종 종료 긍정 응답 후에 주 마스터가 버스 소유권을 상실한때, 주 마스터(10)는 한 클럭 주기동안 BTT*를 어서트하고 한 클럭 주기동안 BTT*를 니게이트하고, BTT*를 3-상태로 한다(제 2 도 참조). 외부 버스 중재기(14)가 BG 를 부 마스터에 어서트함으로써 버스를 부 마스터(예컨대, 부 마스터(12)로 허가하는 경우 BTT*프로토콜을 사용하여 부 마스터는 부 마스터가 주 마스터(10)로부터 BTT*의 어서션을 검출한 후에 바로 다음 클럭 에지상의 버스 사이클을 개시할수 있다.

이전의 마스터는 BTT*를 구동하여 니게이트시킬 수 있으며, 동시에 현재 마스터가 BTT*신호를 3-상태로 유지하므로 버스 사이클을 개시한다(제 2 도 참조). 부 마스터가 보유 기간을 종료할 때까지 이러한 프로토콜에서 BTT*를 구동하지 않으므로, 모든 마스터 BTT*신호를 서로 결합시키는데 충돌이 없다. 즉, 제 1 도에 지시한 것과 같은 버스 비지(B*) 프로토콜에 대한 경쟁이 없다. 제 3 도에서 마스터(10, 12)는 비록 BB*중재 프로토콜이 제 1 도에 도시된 것보다 약간의 타이밍 차이를 갖지만 대안적으로 BB*중재 프로토콜을 제공한다. BB*프로토콜이 제공되어 개량된 BTT*프로토콜을 갖지 않는 구형 시스템이나 프로세서와 용이하게 호환할 수 있도록 한다.

BB*프로토콜을 이용한 프로세서가 버스 사용을 종료 할 때, BB*를 한 시간주기동안 구동시켜 니게이트한 다음 3-상태화 하여야 한다. 고속 버스 환경에서 BB*프로토콜 보다는 위에서 기술한 BTT*프로토콜의 사용이 더 효율적이다. 이는 BTT*를 구동하는 방법(어서트하고, 니게이트하고, 3-상태화 하는 것)이 클럭과 동기화될 수 있기 때문이다. 버스 중재기는 BB*프로토콜 시스템내의 BB*신호 또는 BTT*프로토콜 시스템 내의 BTT*를 샘플화할 필요가 없으므로 BB*프로토콜 시스템에서 BTT*시스템으로 전환할 때 변환될 필요는 없고 버스 소유권을 결정하기 위해 단지 BR*, BG*와 선택적으로 LOCK*를 사용하기만 하면 된다. 어떠한 마스터도 버스 소유권을 양수 받기에 적절한 시간을 결정하기 위해 단지 BB*또는 BTT*와 TS*를 샘플화할 필요가 있다.

주 마스터(10)가 암시(implicit) 버스 소유권을 가지고 있으며 필요한 버스사이클 전부를 종료한 경우에는 BTT*는 BG*가 니게이트되고 주 마스터(10)가 버스에서 이탈될 때까지 BTT*는 3-상태를 유지한다(또는 대안으로서 BB*가 어서트된 상태를 유지한다). 이러한 경우에 주 마스터가 BG*의 니게이션을 검출한 후 바로다음 주기에 주 마스터는 한 클럭 주기동안 BTT*를 어서트한다(또는 대안으로서 단일 클럭(CLK) 주기동안 BB*를 니게이트한다). 주 마스터(10)에 버스가 주어지나 TS*를 어서트함으로써 실제로는 버스를 사용하지 않는 암시 버스 소유권의 경우에 주 마스터(10)는 BG*가 니게이트될 때 BTT*를 어서트하지 않는다(또는 대안으로서 BB*를 어서트하고 니게이트 한다).

BTT*의 전이는 버스가 이양되는 적절한 시간을 결정 하기 위하여 BTT*프로토콜을 사용하는 시스템에서 마스터에 의해 탐지되어야 한다. BG*의 니게이션 이전, 도중과 이후의 BTT*전이는 또한 버스 허가(BG*)가 마스터로 고정되지 않는 경우에는 마스터에 의해 로그될(logged) 필요가 있으며, 얼마 동안은 어떠한 마스터도 버스를 사용하지 않는다. 이러한 경우에, 어떠한 마스터도 이전의 마스터가 즉시 버스를 제어하기에 안전함을 암시하는 버스의 사용을 이미 종료했음을 지시하는 상태 정보를 보유할 것이 요구된다.

외부 리셋 후, RSTI* (외부 리셋핀)의 니게이션으로 초기화되고 BG*는 어서트된 상태에서 RSTI*의 니게이션과 주 마스터(10)가 버스 사이클을 개시하려 할 때의 클럭 사이클 시간 사이에 다른 마스터에 의한 TS *의 어서션의 없다면 주 마스터 (10)는 다른 마스터(즉, 부 마스터)에 의한 BTT *의 어서션을 대기하지 않고 버스의 지배권을 이양받기 시작하여 버스 활동을 개시한다. 다른 마스터(즉, 부 마스터)가 위 시간 사이에서 버스 활동(TS *어서트)을 개시하면, 비록 주 마스터(10)가 버스 허가 지시(BG*어서트)를 받더라도 주 마스터(10)는 버스 지배권/소유권을 이양받기 전에 BTT*가 상기 다른 마스터에 의해 어서트되기를 기다린다.

BG*가 중재기(14)에 의해 니게이트될 때, 주 마스터(10)는 버스 사이클의 로크된 시퀀스가 니게이트된 BGR *와 함께 진행하지 않으며 현재 버스 사이클이 완료되자마자 버스를 포기한다. 이러한 경우에 주 마스터(10)는 원자화 로크된 버스사이클의 시퀀스를 완료하고, 번지 및 다른 버스 사이클 속성이 유휴 상태(idled)일 때의 클럭의 한 클럭 주기동안 LOCK*가 니게이트로 구동되고, 다음 클럭 주기동안 LOCK*를 3 상태로 하며 BTT*를 어서트함으로써 버스를 포기한다(제 6 도 참조). BGR *신호는 BG*에 대한 수식자(qualifler)로서 BG *가 니게이트될 때 버스소유권을 포기할 필요정도를 주 마스터(10)에 지시한다. BGR*은 주로 BG*가 니게이트될 때 원자화 로크된 시퀀스 동안 주 마스터(10)가 동작하는 방식에 영향을 미친다.

제 3 도의 주 마스터(10)는 비록 프로세서가 버스 사이클의 로크된 시퀀스를 지시하고 있더라도 BG*의 니게이션을 검출 한다면 버스 소유권이 주 마스터(10)로부터 제거되고 다른 버스 마스터에 부여되도록 하는 중재 프로토콜을 가진다. LOCK*신호가 주 마스터(10)에 의해 제공되어 프로세서가 버스 사이클의 현재 세트가 함께 로크되게(즉, 분리할 수 없게) 하려는 경향이 있음을 지시한다. 그러나, 이것은 BGR*신호의 시스템 버스 중재기의 제어에 의해 강제되거나 무시된다. 외부 버스 중재기(14)에 의한 BG*니게이션과 BGR*어서션은 비록 프로세서가 원자화 버스사이클의 로크된 시퀀스를 실행하고 있더라도 현재 버스 사이클이 종료되자 마자 프로세서로 하여금 버스를 포기하게 한다. 주 마스터(10)가 로크된 버스 사이클의 시퀀스를 실행하고 있을 때 BGR*과 BG*모두가 니게이트되면 주 마스터(10)는 원자화 로크된 버스 사이클의 전 세트를 종료하고, 원자화 로크된 버스 사이클의 완료점에서 버스를 포기하며, 그래서 원자화 시퀀스의 파괴를 방지한다. 주 마스터(10)는 한 세트의 연속 원자화 로크된 버스 사이클을 실행할 수 있으며, 그러므로, 외부 마스터가 버스 사이클의 그 인접부를 검출하여 BG*를 니게이트할 안전한 시간을 결정 할 수 없다(제 4 도 참조). 이는 BGR*이 니게이트될 때 회피된다. BGR*이 니게이트될 때, 주 마스터(10)는 원자화 로크된 버스 사이클의 현재 세트의 최종 버스 사이클을 종료하고 버스를 포기하며, 이렇게 함으로써 원자화 로크된 버스 사이클의 중단을 방지하지만 모든 연속 원자화 로크된 버스 사이클이 실행되기 전에 버스를 해제한다. 그러므로 버스 가용도가 개선된다.

BGR*프로토콜의 대안으로서, 주 마스터(10)는 계열 호환성을 위해 LOCKE *신호를 보유한다. 주 마스터(10)는 버스 사이클의 로크된 시퀀스의 최종 버스 사이클 동안 LOCKE*신호를 이용하여 외부 중재기로 하여금 버스상의 연속 로크된 시퀀스 사이의 경계를 검출하도록 하는 프로세서 계열의 일부이다. 프로세서 계열중 하나의 프로세서에 이용된 외부 중재기는 LOCKE*상태 신호를 이용하여 로크된 시퀀스를 파괴하지 않고 BG*를 제거하는 안전한 시간을 결정할 수 있으며 중재가 로크된 시퀀스의 최종 이전에 중첩되게 한다. 그러나, 로크된 시퀀스의 최종 버스 사이클동안 LOCKE *는 어서트되고 BG*는 니게이트된 재시도 긍정응답 종료(TRA*)는 외부 버스 중재기의 비동기 논리가 버스 사이클의 종료 전에 BG*를 다시 어서트하도록 요구하여 상기 로크된 시퀀스의 분리 또는 중단을 방지한다. BGR*프로토콜을 사용하여 주 마스터(10)가 그 자체로 버스 소유권을 포기하는 적절한 시간을 결정케 함으로써, 이러한 문제점이 방지되고 외부 버스 중재기 설계가 단순화된다.

로크된 시퀀스는 통상적으로 비교와 교체(CAS: compare and swap) 명령과 같은 명령이 요구되며, 선택적으로 이동 제어(MOVEC: move control) 명령, 테스트와 세트(TAS: test and set) 명령이 요구된다. 버스 사이클의 로크된 시퀀스에 대해, 주 마스터(10)는 제 1 버스 사이클의 TS *에 함께 LOCK*를 어서트하고, 메모리 테이블 워크(tablewalks)내 히스토리 정보의 갱신과 동시에 TAS 및 CAS 명령을 실행하는 동안, 그리고 버스 제어 레지스터에 LOCK 비트를 세트하고 나서 리세트하는 MOVEC 명령을 실행한 후에 최종 버스 사이클의 최종 이전에 대해 최종 종료 긍정응답을 하고 이어서 LOCK*를 니게이트한다. 중재기(14)가 LOCK*와 BGR*에 관하여 설계되는 방식에 따라서 LOCK*와 LOCKE*신호의 사용은 로크된 시퀀스 동안 버스 중재의 중첩을 방지하는 효과가 있다. 로크된 시퀀스의 지속기간 동안 LOCK*를 어서트된 상태로 계속 유지함으로써 상기 시퀀스의 최종 버스 사이클은 재시도 될 수 있고 여전히 로크 상태를 유지한다.

버스가 계속 소유되지만 이전의 사용후 유휴 상태에 있는 시간동안 프로세서계열중 다른 프로세서와 같이 주 마스터(10)는 번지 및 속성 라인상의 최종 버스 사이클로부터 정보를 계속적으로 보유하고 구동한다. 그러나 프로세서군의 다른 프로세서와는 달리 암시 버스 소유권의 경우에는, 프로세서(10)에 버스는 허가되었지만 아직 사이클을 실행하지 않는 동안 프로세서는 번지 및 속성 라인을 구동하지 않으며 버스 사이클이 실제로 개시될 때까지 3-상태를 유지한다.

BGR* 신호의 사용은 현재 마스터가 동기 버스 시스템에서 일련의 연속 로크된 전송 시퀀스를 실행하고 있을 때 버스 소유권이 다른 마스터로 이송되는 시점을 결정할 수 없는 외부 버스 중재기의 문제점을 해결한다. 로크된 전송의 발생은 통상적으로 LOCK* 신호로 지시되는데 이 신호는 현재 마스터가 전송 또는 전송의 시퀀스를 원자화(즉, 분리할 수 없게) 시키려는 경향이 있음을 나타낸다. 그러나, 현재 마스터가 몇몇의 이러한 원자화 로크된 시퀀스 연속을 가지고 외부 중재기가 이러한 원자화 시퀀스의 인접부를 인지한다면 외부 중재기가 상기 시퀀스의 인터페이스에서 마스터로부터 버스 소유권을 이양받아 더 필요한 마스터에 소유권을 부여하는 것이 완전히 가능하다.

BGR* 신호는 이런한 부가된 기능을 허여한다. 그러므로, BGR* 신호는 현재 버스 마스터가 일련의 긴 연속 로크된 전송을 실행하고 있는 동안 다른 버스 마스터의 기아를 방지하는데 사용될 수 있다. 제 4 도는 버스 클럭을 개입시키지 않고 두개의 연속적인 로크된 시퀀스를 실행하는 공지의 버스 프로토콜의 동작을 도시한다. 제 4 도로부터 알 수 있듯이 외부 중재기는 단일 로크된 시퀀스, 다수-전송 로크된 시퀀스, 또는 버스 클럭을 개입시키지 않는 다중 시퀀스 사이의 차이점을 판정할 수 없다. 독립적인 로크된 시퀀스의 인접부를 판정 할 수 없는 점은 중재기가 다른 버스 마스터로 하여금 버스를 소유케할 필요가 있고 링크된 로크 시퀀스의 로크된 특성을 중단없이 보전할 필요성이 있는 때에 하나의 문제점이다.

로크된 시퀀스의 파괴를 방지하기 위해 버스 중재기는 BG* (버스 허가)를 현재 버스를 사용하고 있는 마스터에 니게이트 하기 전에 LOCK* 의 니게이션을 대기하도록 강제된다. LOCK* 가 시퀀스 사이에서 니게이트되지 않으므로 현재 마스터는 연속 로크된 시퀀스를 실행함으로써 버스를 효과적으로 독점할 수 있다. 제 4 도의 버스 프로토콜은 로크된 시퀀스에서 마지막 전송의 종료를 통하여 LOCK* 가 어서트될 것을 요구한다. LOCK* 신호는 슬레이브 장치가 어떤 버스 이전에 대한 이전 재시도의 필요성을 신호하기 때문에 로크된 시퀀스내의 모든 전송동안 어서트된 상태를 유지해야 한다.

위에서 기술한 결정의 문제점은 주 프로세서(10)의 선행하는 프로세서군에도 존재한다. 상기 선행하는 프로세서 군에 고려되는 가능한 해결책은 로크된 시퀀스사이에 1 또는 2 의 데드 클럭을 부가하는 것이다. 그러나, 로크된 시퀀스의 파괴를 종료시키거나 긴급한 요청에 대해 다른 마스터의 대기 시간을 증가시키고 성능을 저하시키는 불필요한 부작용이 발생한다. 결정의 문제점을 해결하기 위하여 종래 프로세서는 최종 로크된 버스 사이클 상태 지시기(LOCKE* 신호)를 그 설계에 부가하여 중재기로 하여금 로크된 시퀀스의 최종 전송 및 전송의 중첩 중재 동안 BG* 를 마스터로 니게이트하도록 한다. 그러나, 버스 사이클 재시도 동안에 외부 버스 중재기가 버스 트랜잭션의 로크된 완전성을 보전하기 위하여 BG* 를 비동기적으로 리어서트해야만 하는 경우가 있다.

제 5 도는 이러한 경우를 예시한다. 제 5 도에서 LOCKE* 가 어서트되고 BR* (버스 요청)이 니게이트될 때 버스 중재기가 버스 사이클 2(BC 2)에서 BG* 를 니게이트한다. 그러나, 이송 재시도가 슬레이브 장치에 의해 신호된다(TRT* 어스트됨). 이는 이송 재시도 신호(TRA) 역시 새로 샘플화되어야 하므로 이러한 문제점을 해결하는 외부 버스 중재기의 설계를 매우 어렵게 만든다. 이것은 매우 높은 주파수에서 거의 불가능하다.

주 마스터(10)는 버스 허가 포기(BGR*) 입력(LOCKE* 출력이 필요없음)을 제공함으로써, 위에서 기술한 문제점을 내부적으로 해결하여 주 마스터(10) 자체가 버스를 포기할 시점을 결정할 수 있게 한다. 제 6 도에서 BGR* 입력은 주 마스터(10)의 버스를 포기할 필요정도를 통보한다. 외부 중재기가 BG* 를 니게이트 하는 동안에 또는 그 이후에 BGR* 이 어서트되면(제 6 도의 버스 사이클 5(BC5) 참조), 주 마스터(10)는 비록 현재 사이클이 버스 사이클의 로크된 시퀀스의 일부라 할지라도 현재 사이클의 종료에서 무조건 버스를 포기한다. 그러므로, 클럭은 BGR* 신호의 상태에 기인하여 파괴된다. 외부 버스 중재기가 BG* 를 니게이트할 때 BGR* 이 니게이트되면(제 6 도, 버스 사이클 2 와 3(BC2/BC3 참조), 주 마스터(10)는 로크된 시퀀스 전체를 성공적으로 완료할 때 까지 대기하며, 그렇게 함으로써 로크된 시퀀스의 원자성(atomicity)을 보장하고 이전 재시도 조건(TRA*)에 관해 위에서 논의한 문제를 방지한다.

제 6 도는 BGR*에 관해 주 마스터(10)의 동작을 예시한다. 외부 중재기 (14)는 BGR*이 주 마스터(10)에 니게이트되어 부 마스터에 버스를 허가한다. 따라서, 주 마스터(10)는 버스를 포기하기 전에 일련의 로크된 버스 사이클 BC1, BC2, BC3(버스 사이클 1, 2, 3)를 종료한다. 버스 사이클 5(BC5)에서 주 마스터(10)는 로크된 버스 사이클의 시퀀스를 재실행하지만 이경우 BGR*은 주 마스터(10)에 어서트 되고 BG*는 니게이트된다. 그러므로 주 마스터(10)는 현재 버스 사이클의 종료에서 버스를 즉시 포기하여 로크된 전송 시퀀스를 선취한다.

일반적으로, BGR* 프로토콜은 외부 버스 중재기에 요구되는 비동기 논리의 문제점을 전부 해결한다. 게다가, BGR* 프로토콜은 시스템의 각 마스터가 버스 소유권을 결정하기 위해 2 개의 서로 다른 버스 허가 입력의 상태를 샘플화할 필요없이 요구되는 비동기 논리의 문제를 해결한다. 2 신호 버스 허가 방법(미국 특허 제 5,127,089 호 참조)은 버스 마스터 내부에 불필요한 논리의 오버헤드(overhead)를 가지며 시스템상에 중재기로부터 각 버스 마스터로 다른 신호를 경로지정하여 버스 소유권을 결정하고 외부 버스 중재기의 설계를 복잡하게 만든다. BGR* 프로토콜에서, 각 버스 마스터에 독립적으로 결합된 각각의 버스 허가 신호(BG*)는 버스 소유권을 결정하는데 사용되며 반면에 버스 허가 포기(BGR*) 신호는 BG*가 니게이트 될 때 버스 허가(BG*) 신호의 의미와 우선 순위를 식별한다. 이러한 프로토콜/방법은 역시 보다 더 단순한 외부 버스 중재기 설계를 가능케 한다.

요약하면, 여기에 제안된 프로토콜은 BG*가 니게이트 될 때 마이크로프로세서에 버스 소유권 포기의 긴급성을 지시하는 BGR* 신호를 제공한다. BG*가 니게이트될 때 BGR*이 어서트되면 버스 제어기/중재기(14)는 주 마스터(10)가 로크된 버스 시퀀스와 관계없이 가능한 한 신속하게 버스 소유권을 포기해야만 함을 주 마스터(10)에 지시한다. BG*가 주 마스터(10)에 니게이트될 때 BGR*이 니게이트되면, 주 마스터(10)는 버스를 포기하기에 가장 적절한 시간을 결정한다. 통상 버스 사이클이 로크되지 않으면 가장 적절한 시간은 다음 버스 전송 후나 완전한 로크된 전송 시퀀스의 종료이다.

여기에 기술한 프로토콜은 버스 소유권을 최종으로 보유한 마스터가 그 버스 소유권을 종료하고 있음을 다른 마스터에 지시하는 BTT* 신호도 제공한다. 여기에 기술한 BTT*는 공지의 BB* 프로토콜 등의 다른 프로토콜 보다 뛰어나다. BTT* 신호는 버스 중재 동안 버스 데이터 경쟁을 야기하는 계속적으로 어서트된 BB* 신호와 달리 일련의/ 단일의 전송 최종 버스 사이클이 완료된 때 어서트된다.

본 발명은 특정한 실시예를 참고하여 예시되고 기술 되었지만 본 분야의 숙련자에게는 본 발명의 변형이나 개량이 가능하다. 예를 들면, 여기 기술된 제어 비트는 명칭은 다르지만 동일한 기본 기능을 가질 수 있다. 버스, 번지 버스, 데이터 버스, 제어 신호 버스의 크기와 다른 값들은 설계에 따라 변화될 수 있다. 어떤 마스터 수, 즉, N 마스터(N 은 양의 정수)가 버스 중재기(14)에 접속될 수 있다. 다른 형태에서, BGR*신호는 버스 중재기(14)의 배타적 출력이 아니라 시스템의 다른 마스터에 의한 출력으로서 공급될 수 있다. 따라서, 본 발명은 여기에 예시된 특정 형태에만 한정되지 않으며, 부가된 특허청구 범위에 의해 본 발명의 사상과 범위내의 모든 변형을 포함한다.

(57) 청구의 범위

청구항 1.

버스 제어기로 버스 이송을 제어하는, 마이크로프로세서에 의해 실현되는(microprocessor implemented) 버스 프로토콜 방법에 있어서,

버스 제어기로부터 제 1 제어 신호를 어서트(asserting)함으로써 마스터 회로에 버스 소유권을 허가하는 단계로서, 상기 허가는 상기 제1 제어 신호 상에서만 조절되는, 상기 마스터 회로에 버스 소유권을 허가하는 단계와,

버스 이송이 개시됨을 지시하는 제 2 제어 신호의 어서션(assertion)에 응답하여 상기 마스터 회로와 그 마스터 회로의 외부 장치간에 데이터를 전송하는 단계와,

상기 제 1 제어 신호가 디어서트될 때 현재 버스 사이클의 종료에서 상기 마스터 회로 버스 소유권이 포기되거나 상기 제 1 제어 신호가 디어서트될 때 마스터 회로의 재량으로 마스터 회로 버스 소유권이 종료되어야 하는지 여부를 상기 마스터 회로에 지시하는 제 3 제어 신호를 상기 버스 제어기로부터 상기 마스터 회로로 전송하는 단계와,

상기 버스 소유권이 포기되어야 함을 상기 마스터 회로에 지시하기 위하여 상기 제 1 제어 신호를 디어서트(deassert ing)하는 단계와,

상기 제 3 제어 신호에 의해 결정된 바와 같이 상기 현재 버스 사이클의 종료에서 또는 상기 마스터 회로의 재량으로 상기 버스 소유권을 포기하는 단계와,

상기 제 3 제어 신호와 제 1 제어 신호의 상태에 의해 결정되는 시간에 상기 마스터 회로에 의해 어서트되는 제 4 제어 신호를 어서트함으로써 상기 마스터 회로가 버스 소유권을 종료함을 마스터 회로의 외부에 통신하는 단계로서, 상기 제 4 제어 신호는 상기 마스터 회로가 버스 소유권을 가진 동안에는 3-상태화되고, 또한 상기 마스터 회로가 버스 소유권을 가지지 않은 동안에 3-상태화되며, 버스 소유권이 상기 마스터 회로에 대하여 허가된 상태와 비허가 상태 사이에서 전이되고 있는 때에만 어서트되는, 상기 마스터 회로의 외부에 통신하는 단계를 포함하는, 버스이송을 제어하는, 마이크로프로세서에 의해 실현되는 버스 프로토콜 방법.

청구항 2.

하나의 버스를 통해 제 1 회로를 제 2 회로와 인터페이스 하게 하는 버스 프로토콜을 가진 데이터 처리 시스템에 있어서,

버스 소유권이 요청됨을 지시하기 위하여 제 1 제어 신호를 상기 제 1 회로로부터의 한 출력으로서 공급하는 회로와,

상기 제 1 회로에 버스 소유권이 허가되었는지 여부를 지시하는 데 사용되는 제 2 제어 신호를 상기 제 1 회로로의 한 입력으로서 수신하는 회로와,

상기 제 1 회로가 버스 소유권을 포기할 준비가 될 때에만 어서트되는 제 3 제어 신호를 상기 제 1 회로로부터의 출력으로서 공급하는 회로로서, 상기 제 3 제어 신호는 상기 제 1 회로가 상기 버스의 제어권을 가진 제 1 시간 주기 동안에 3-상태화되고, 또한 상기 제 1 회로가 상기 버스의 제어권을 가지지 않는 제 2 시간주기 동안에 3-상태화되며 , 상기 제 1 시간 주기로부터 상기 제 2 시간 주기로의 시간 전이하는 2 클럭 사이클 시간 주기에서 시간 순차로 또한 상호 배타적으로 고전압과 저전압 둘 다에 어서트되는, 상기 제 3 제어 신호를 상기 제 1 회로로부터의 출력으로서 공급하는 회로를 포함하는 데이터 처리 시스템.

청구항 3.

하나의 버스를 통해 제 1 회로를 제 2 회로와 인터페이스 하게 하는 버스프로토콜에 있어서,

버스 소유권이 요청됨을 지시하기 위하여 상기 제 1 회로로부터의 출력으로서 제 1 제어 신호를 공급하는 수단과,

상기 제 1 회로에 버스 소유권이 허가되었는지 여부를 지시하는데 사용되는 제 2 제어 신호를 상기 제 1 회로로의 입력으로서 수신하는 수단과,

상기 제 1 회로에 의해 실행되는 현재 버스 사이클의 종료에서 상기 제 1 회로가 버스 소유권을 포기해야만 함을 지시하기 위하여 제 1 논리 상태로 세트되고, 상기 제 1 회로에 의해 결정된 편리한 시간에 상기 제 1 회로가 버스 소유권을 포기함을 지시하기 위해 제 2 논리 상태로 세트되는 제 3 제어 신호를 상기 제 1 회로의 입력으로서 수신하는 수단을 포함하는 버스 프로토콜.

청구항 4.

하나의 버스를 통해 제 1 회로를 제 2 회로와 인터페이스 하게 하는, 마이크로프로세서에 의해 실현되는 버스 프로토콜에 있어서,

버스 소유권이 요청됨을 지시하기 위하여 제 1 제어 신호를 상기 제 1 회로로부터의 출력으로서 공급하는 수단과,

상기 제 1 회로에 버스 소유권이 허가되었는지 여부를 지시하는데 사용되는 제 2 제어 신호를 상기 제 1 회로로의 입력으로서 수신하는 수단과,

상기 제 1 회로에 의해 실행되는 현재 버스 사이클의 종료에서 상기 제 1 회로가 버스 소유권을 포기해야만 함을 지시하기 위하여 제 1 논리 상태로 세트되고, 상기 제 1 회로에 의해 결정된 편리한 시간에 상기 제 1 회로가 버스 소유권을 포기함을 지시하기 위해 제 2 논리 상태로 세트되는 제 3 제어 신호를 상기 제 1 회로의 입력으로서 수신하는 수단과,

상기 제 1 회로가 버스 소유권을 포기할 준비가 될 때에만 어서트되는 제 4 제어 신호를 상기 제 1 회로로부터의 출력으로서 공급하는 수단으로서, 상기 제 4 제어 신호는 상기 제 1 회로가 상기 버스의 제어권을 가진 제 1 시간 주기 동안에 3-상태화되고, 또한 상기 제 1 회로가 상기 버스의 제어권을 가지지 않는 제 2 시간 주기 동안에 3-상태화되며, 상기 제 1 시간 주기로부터 상기 제 2 시간 주기로의 시간 전이하는 2 클럭 사이클 시간 주기에서 시간 순차로 또한 상호 배타적으로 고전압과 저전압 둘 다에 어서트되는, 상기 제 4 제어 신호를 상기 제 1 회로로부터의 출력으로서 공급하는 수단을 포함하는, 마이크로프로세서에 의해 실현되는 버스 프로토콜.

청구항 5.

하나의 버스를 통해 제 1 회로를 제 2 회로와 인터페이스하게 하는 버스프로토콜 방법에 있어서,

상기 제 1 회로가 버스 소유권을 요청하고 있음을 지시하기 위해 상기 제 1회로로부터 제 1 제어 신호를 공급하는 단계와,

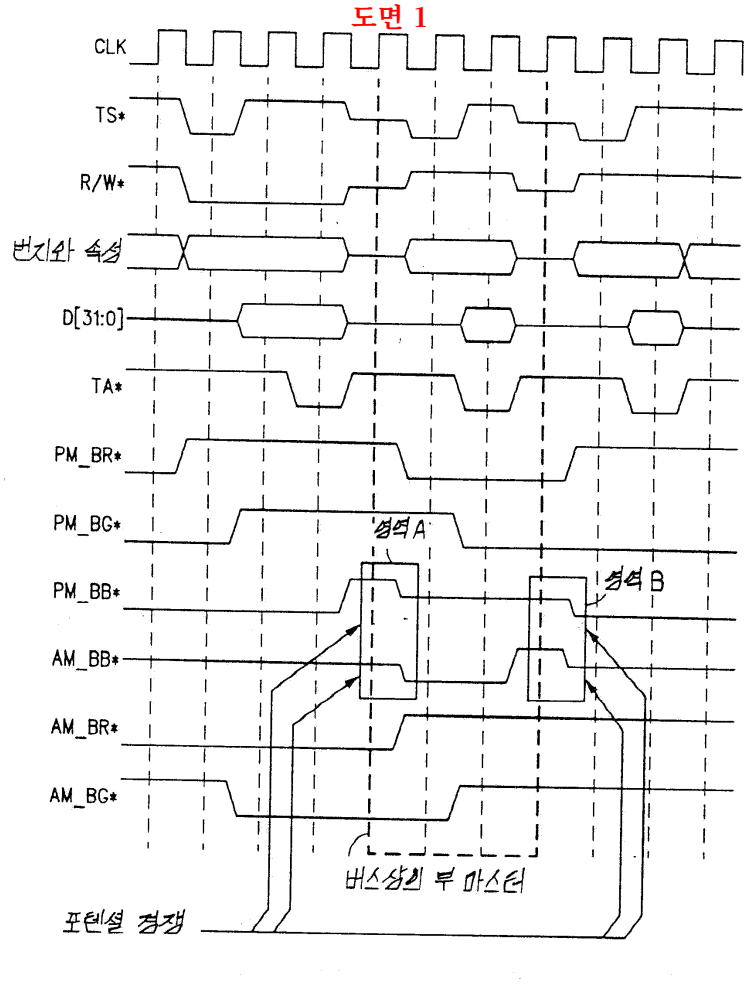
상기 제 1 회로에 버스 소유권이 허가되었는지 여부를 지시하는데 사용되는 제 2 제어 신호를 상기 제 1 회로에 공급하는 단계와,

상기 제 1 회로로부터 제 3 제어 신호를 공급하는 단계를 포함하고,

상기 제 3 제어 신호는 상기 제 1 회로에 의해 개시된 버스 사이클을 완료한 후에 어서트되며 상기 제 1 회로가 버스 소유권을 포기할 준비가 될 때에만 상기 제 1 회로에 의해 어서트되고, 상기 버스 사이클이 실행될 때 3-상태로 되는 버스프로토콜 방법.

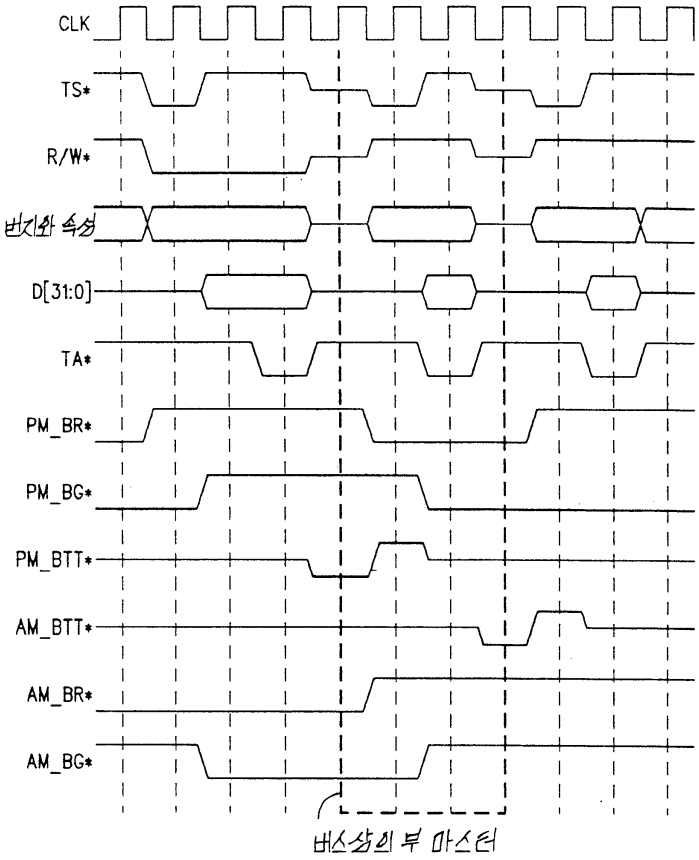
도면

도면 1

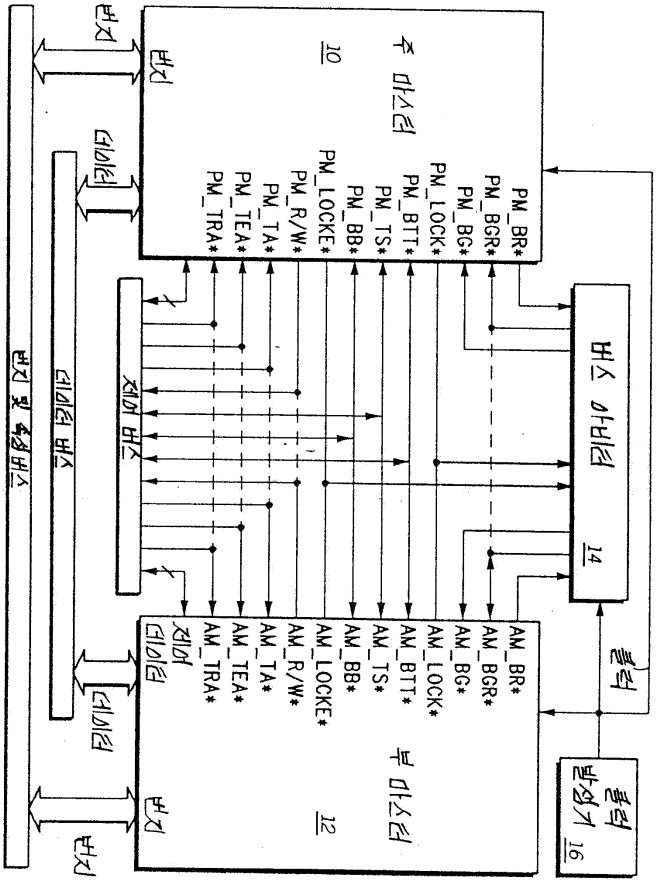

CLK
TS*
R/W*
버지와 속성
D[31:0]
TA*
PM_BR*
PM_BG*
PM_BB*
AM_BB*
AM_BR*
AM_BG*
영역A
영역B
버스상의 부 마스터
포텐셜 경쟁

- 종래기술 -

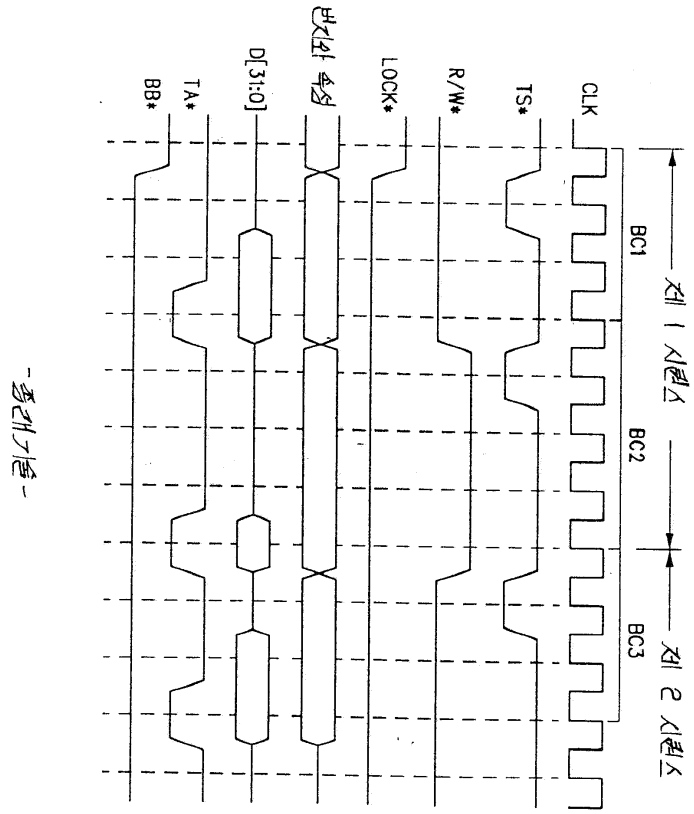
도면 2


CLK
TS*
R/W*
버저와 속성
D[31:0]
TA*
PM_BR*
PM_BG*
PM_BTT*
AM_BTT*
AM_BR*
AM_BG*
버스상의 부 마스터

도면 3

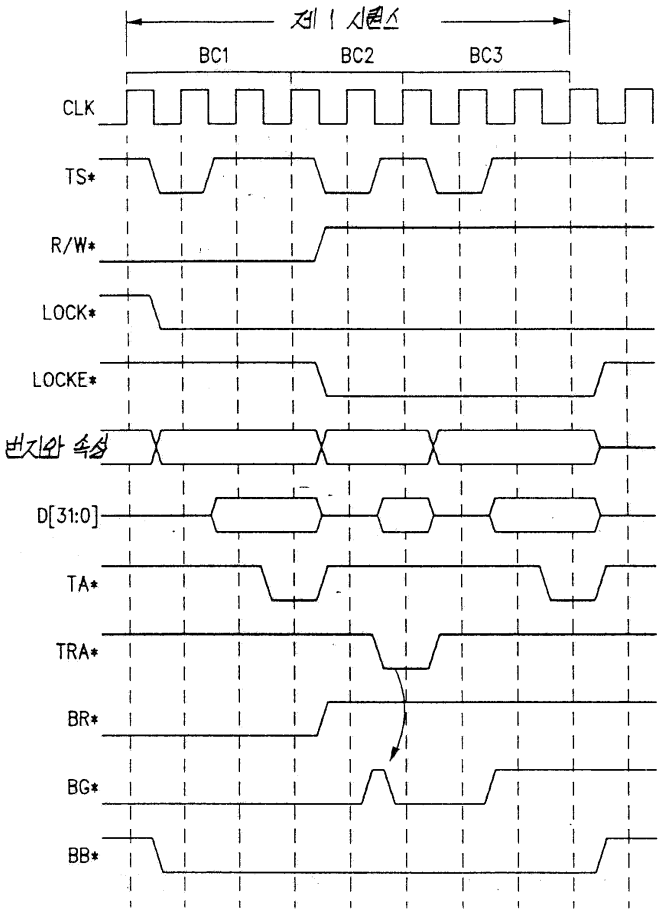

주 마스터
10
PM_BR*
PM_BGR*
PM_BG*
PM_LOCK*
PM_BTT*
PM_TS*
PM_BB*
PM_LOCKE*
PM_R/W*
PM_TA*
PM_TEA*
PM_TRA*
버스 아비터
14
클럭
클럭 발생기
16
부 마스터
12
AM_BR*
AM_BGR*
AM_BG*
AM_LOCK*
AM_BTT*
AM_TS*
AM_BB*
AM_LOCKE*
AM_R/W*
AM_TA*
AM_TEA*
AM_TRA*
제어 버스
제어 데이터
버퍼
버퍼
데이터
데이터
데이터 버스
버퍼 및 제어버스
버퍼

도면 4



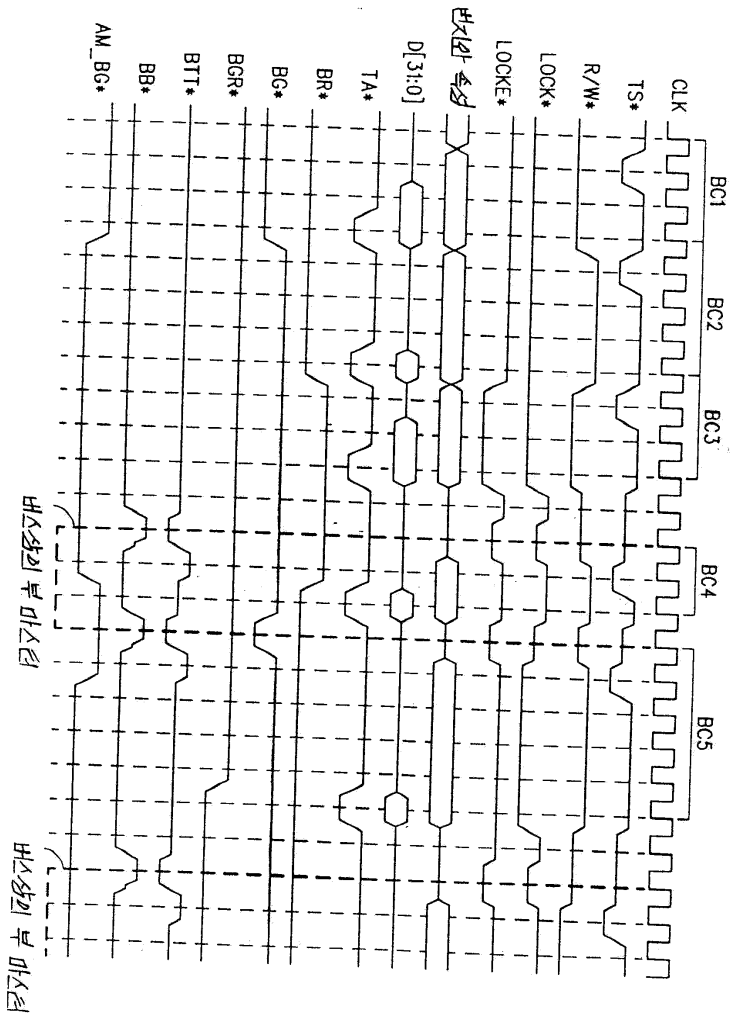
- 종래기술 -

도면 5


제 1 시퀀스
BC1
BC2
BC3
CLK
TS*
R/W*
LOCK*
LOCKE*
번지와 속성
D[31:0]
TA*
TRA*
BR*
BG*
BB*

-종래기술-

도면 6


CLK
TS*
R/W*
LOCK*
LOCKE*
변경된 속성
D[31:0]
TA*
BR*
BG*
BGR*
BTT*
BB*
AM_BG*
BC1
BC2
BC3
BC4
BC5
버스상의 뷰 마스터
버스상의 뷰 마스터