



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2009-0089043
(43) 공개일자 2009년08월21일

(51) Int. Cl.

G06F 13/26 (2006.01) G06F 13/14 (2006.01)

(21) 출원번호 10-2008-0014373

(22) 출원일자 2008년02월18일

심사청구일자 2008년02월18일

(71) 출원인

인하대학교 산학협력단

인천 남구 용현동 253 인하대학교

(72) 발명자

윤영섭

서울 서초구 잠원동 한신아파트 8차 308동 801호

이국표

인천 남동구 만수동 벽산아파트 109동 2103호

(74) 대리인

황광연, 강동호, 오정환

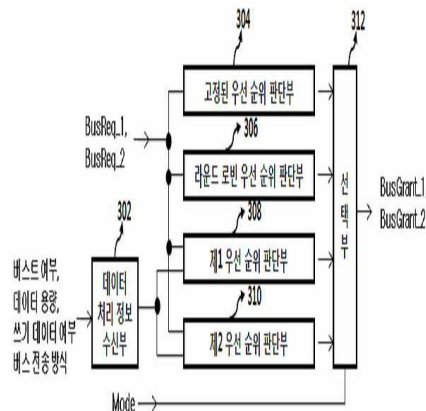
전체 청구항 수 : 총 5 항

(54) 버스 중재 방법 및 장치

(57) 요약

마스터의 데이터 처리 방식에 따라 버스의 우선 순위를 판단할 수 있는 본 발명의 일 실시예에 따른 버스 중재 방법은 마스터로부터 버스를 통해 송수신할 데이터의 처리 방식에 대한 정보를 수신하는 단계; 및 상기 데이터의 처리 방식에 대한 정보에 따라 상기 마스터의 상기 버스의 사용에 대한 우선 순위를 판단하는 단계를 포함하는 것을 특징으로 한다.

대표도



특허청구의 범위

청구항 1

마스터로부터 버스를 통해 송수신할 데이터의 처리 방식에 대한 정보를 수신하는 단계; 및

상기 데이터의 처리 방식에 대한 정보에 따라 상기 마스터의 상기 버스의 사용에 대한 우선 순위를 판단하는 단계;

를 포함하는 것을 특징으로 하는 버스 중재 방법.

청구항 2

제1항에 있어서,

상기 데이터의 처리 방식에 대한 정보는 버스트 여부, 데이터 크기, 버스 전송 방식, 및 쓰기 데이터 여부 중 적어도 하나를 포함하는 것을 특징으로 하는 버스 중재 방법.

청구항 3

제1항에 있어서,

상기 우선 순위는 상기 마스터의 종류 또는 상기 마스터에 할당된 시간 정보에 따라 판단되는 것을 특징으로 하는 버스 중재 방법.

청구항 4

제1항에 있어서,

상기 버스는 메모리 접근을 위한 버스인 것을 특징으로 하는 버스 중재 방법.

청구항 5

마스터로부터 버스를 통해 송수신할 데이터의 처리 방식에 대한 정보를 수신하는 데이터 처리 정보 수신부; 및

상기 데이터의 처리 방식에 대한 정보에 따라 상기 마스터의 상기 버스의 사용에 대한 우선 순위를 판단하는 우선 순위 판단부;

를 포함하는 것을 특징으로 하는 버스 중재 장치.

명세서

발명의 상세한 설명

기술분야

<1> 본 발명은 버스 중재 방법 및 장치에 관한 것으로서 보다 상세하게는 버스를 통해 송수신할 데이터의 처리 방식에 대한 정보에 따라 버스를 중재하는 방법 및 장치에 관한 것이다.

배경기술

<2> 하나의 칩에 여러 가지의 기능이 집적화된 SoC칩의 경우, 1개의 버스에 마스터(Master), 아비터(Arbiter), 메모리 등이 연결된 버스 아키텍처가 필요하다. 다수의 마스터들이 버스를 이용해서 메모리 접근 등의 동작을 할 경우, 아비터로부터 우선 순위에 따른 버스 우선권을 부여 받아 데이터를 처리하게 된다.

<3> 도 1의 일반적인 메모리 접근을 위한 버스 아키텍처를 참조하면, 1개의 버스(102)에 복수의 마스터들(104,106), 아비터(108), 메모리(110)가 연결되어 있을 경우, 아비터(108)는 동시에 여러 마스터들(104,106)이 메모리(110)로의 접근을 위해 버스(102)의 사용을 요청할 때, 우선 순위를 결정해야 하기 때문에, 칩의 성능에 결정적인 영향을 미치게 된다.

<4> 버스의 우선권을 결정하기 위해서 아비터(108)의 여러 가지 우선 순위 판단 방법이 연구되어 왔는데, 대표적인

것이 도 2a의 고정된 우선 순위(Fixed Priority) 판단 방법과 도 2b의 라운드-로빈 우선 순위(Round-Robin Priority) 판단 방법이다.

- <5> 도 2a 참조하면, 고정된 우선 순위 판단 방법은 각각의 마스터에 대해 우선 순위를 1, 2, 3, 4 등으로 정해 놓고, 마스터가 동시에 접근했을 경우, 우선순위가 높은 마스터에게 버스를 건네주게 된다.
- <6> 그리고, 도 2b를 참조하면, 라운드-로빈 우선 순위 판단 방법은 마스터의 버스 우선 순위를 시간에 따라서 다르게 변화시켜서 골고루 버스 우선권을 주는 방식이다. 즉, 마스터에 일정시간을 할당하고, 할당된 시간이 지나면 그 마스터는 잠시 보류한 뒤 다른 마스터에게 버스 우선권을 주고, 해당 할당 시간이 지나면, 또 그 다음 마스터에게 버스 우선권을 주는 운영 방식이다.
- <7> 그런데, 고정된 우선 순위 판단 방법의 경우, 마스터가 처리하게 될 데이터의 특성 등을 고려하지 않으며, 단순히 마스터의 종류에 따라 우선 순위를 결정하므로, 데이터 처리 시간 측면에서 효율적이지 않은 문제점이 있었다.
- <8> 또한, 라운드-로빈 우선 순위 판단 방법의 경우, 고정된 우선 순위 없이 마스터에 할당된 시간 정보를 참조하여 우선 순위를 정하므로 빨리 처리해야 할 중요한 특정 데이터에 대한 처리가 늦어지는 문제점이 있다.

발명의 내용

해결 하고자하는 과제

- <9> 본 발명은 상술한 문제점을 해결하기 위한 것으로서, 1개의 버스를 공유하는 복수의 마스터들이 효율적으로 버스를 점유함으로써 데이터의 처리 시간을 줄일 수 있는 버스 중재 방법 및 장치를 제공하는 것을 기술적 과제로 한다.
- <10> 또한, 본 발명은 신속한 처리를 요하는 특정 데이터에 대한 우선 순위를 높이는 버스 중재 방법 및 장치를 제공하는 것을 다른 기술적 과제로 한다.

과제 해결수단

- <11> 상술한 목적을 달성하기 위한 본 발명의 일 측면에 따른 버스 중재 방법은 마스터로부터 버스를 통해 송수신할 데이터의 처리 방식에 대한 정보를 수신하는 단계; 및 상기 데이터의 처리 방식에 대한 정보에 따라 상기 마스터의 상기 버스의 사용에 대한 우선 순위를 판단하는 단계를 포함하는 것을 특징으로 한다.
- <12> 상술한 목적을 달성하기 위한 본 발명의 일 측면에 따른 버스 중재 장치는 마스터로부터 버스를 통해 송수신할 데이터의 처리 방식에 대한 정보를 수신하는 데이터 처리 정보 수신부; 및 상기 데이터의 처리 방식에 대한 정보에 따라 상기 마스터의 상기 버스의 사용에 대한 우선 순위를 판단하는 우선 순위 판단부를 포함하는 것을 특징으로 한다.

효 과

- <13> 상술한 바와 같이 본 발명에 따르면, 버스를 통해 송수신할 데이터의 처리 방식에 따라 우선 순위를 달리함으로써, 마스터의 데이터 처리 방식의 특성에 따라 발생하는 버스 사용의 비효율성을 제거하여 버스 아키텍처 전체의 데이터 처리 시간을 줄일 수 있는 버스 중재 방법 및 장치를 구현할 수 있는 효과가 있다.
- <14> 또한, 본 발명에 따르면, 마스터의 종류뿐 아니라, 마스터가 버스를 사용하여 수행하는 데이터 처리 방식의 중요도에 따라 우선 순위를 부여함으로써, 중요한 특정 데이터의 처리를 빠르게 처리할 수 있는 버스 중재 방법 및 장치를 구현할 수 있는 다른 효과가 있다.

발명의 실시를 위한 구체적인 내용

- <15> 이하 첨부된 도면을 참조하여 본 발명의 실시예에 대해 상세히 설명한다.
- <16> 도 3은 본 발명의 일 실시예에 따른 버스 중재 장치(아비터: Arbiter)를 나타낸 도면이다.
- <17> 도시된 바와 같이, 본 발명의 일 실시예에 따른 버스 중재 장치(이하, ‘아비터’라 한다)는 데이터 처리 정보 수신부(302), 고정된 우선 순위 판단부(304), 라운드 로빈 우선 순위 판단부(306), 제1 우선 순위 판단부(308), 제2 우선 순위 판단부(310), 및 선택부(312)를 포함하며, 프로세서, DSP블럭, DMA블럭 등의 마스터(Master: 미

도시)들이 메모리에 접근하는 경우, 버스 사용의 우선 순위를 결정한다.

- <18> 데이터 처리 정보 수신부(302)는 마스터들로부터 마스터의 데이터의 처리 방식에 대한 정보를 수신한다. 여기서, 데이터는 아비터를 통해 우선권을 부여 받아 마스터들이 공유하는 버스를 통해 송수신하고 처리하게 될 데이터이다. 일 실시예에 있어서, 데이터의 처리 방식에 대한 정보는 버스트(Burst) 여부, 데이터 크기, 쓰기 데이터 여부, 및 버스 전송 방식 중 적어도 하나를 포함한다.
- <19> 버스트 여부는 마스터들이 공유하는 버스를 통해 메모리에 접근하여 데이터를 메모리에 쓰는 경우, 메모리로부터 데이터를 읽는 경우, 또는 메모리에 명령을 송수신하는 경우에 데이터를 연속적으로 쓰거나 읽는지 또는 명령어를 연속적으로 송수신하는지에 대한 정보이다. 이하, 한번의 쓰기, 읽기 동작, 또는 특정 명령의 송수신 동작을 수행을 싱글(Single)이라 하고, 싱글의 반대되는 개념인 연속적인 해당 동작을 버스트라 하겠다.
- <20> 데이터 크기는 버스를 통해 메모리와 송수신할 데이터의 크기로서, 데이터가 워드(Word) 단위인지, 바이트(Byte) 단위인지 등을 나타낸다.
- <21> 쓰기 데이터 여부는 버스를 통해 전송하게 될 데이터가 메모리에 쓰여질 데이터인지 또는 쓰기 명령어인지에 대한 정보이다. 그리고, 버스 전송 방식은 데이터가 마스터와 메모리 사이에 특정 규칙을 가지고 전송되는 경우, 아비터가 그 특정 규칙을 식별할 수 있는 정보를 의미한다.
- <22> 고정된 우선 순위 판단부(304)는 마스터들로부터 버스 사용 요청을 수신하면, 마스터의 종류에 따라 우선 순위를 판단한다. 여기서, 판단 방법으로서 고정된 우선 순위 판단 방법을 사용한다. 이하, “마스터 1”의 버스 사용 요청을 “BusReq_1”이라 하고, “마스터 2”의 버스 사용 요청을 “BusReq_2”라 하겠다.
- <23> 고정된 우선 순위 판단부(304)는 마스터 1의 기 규정된 순위 점수가 마스터 2의 순위 점수보다 높고, 마스터 1 및 마스터 2로부터 버스 사용 요청을 수신한 경우, 마스터 1에게 먼저 버스 우선권을 부여하고, 마스터 1의 데이터 송수신이 끝난 후 마스터 2에게 버스 우선권을 부여한다.
- <24> 라운드 로빈 우선 순위 판단부(306)는 BusReq_1 및 BusReq_2를 수신하면, 마스터에 할당된 시간 정보를 참조하여 우선 순위를 판단한다. 여기서, 판단 방법으로서 라운드-로빈 우선 순위 판단 방법을 사용한다.
- <25> 라운드 로빈 우선 순위 판단부(306)는 BusReq_1 및 BusReq_2를 수신한 경우, 마스터 1에게 해당 할당 시간 동안 버스 우선권을 부여한 후, 할당 시간이 지나면, 마스터 2에게 버스 우선권을 부여한다. 우선권을 부여하는 순서는 마스터의 종류에 따라 달라진다.
- <26> 제1 우선 순위 판단부(308)는 BusReq_1 및 BusReq_2를 수신하면, 마스터의 종류 및 데이터의 처리 방식에 대한 정보에 따라 우선 순위를 판단한다. 여기서, 데이터의 처리 방식에 대한 정보는 데이터 처리 정보 수신부(302)를 통해 획득한 것이고, 버스트 여부, 데이터 크기, 쓰기 데이터 여부, 및 버스 전송 방식 중 적어도 하나를 포함한다.
- <27> 제1 우선 순위 판단부(308)는 고정된 우선 순위 판단 방법을 사용하여 마스터의 종류에 따라 일차적으로 우선 순위를 정한 후, 데이터의 처리 방식에 대한 정보에 따라 최종적인 마스터의 우선 순위를 결정한다.
- <28> 제2 우선 순위 판단부(310)는 BusReq_1 및 BusReq_2를 수신하면, 마스터의 종류, 마스터에 할당된 시간 정보, 및 데이터의 처리 방식에 대한 정보에 따라 우선 순위를 판단한다. 여기서, 데이터의 처리 방식에 대한 정보는 데이터 처리 정보 수신부(302)를 통해 획득한 것이고, 버스트 여부, 데이터 크기, 쓰기 데이터 여부, 및 버스 전송 방식 중 적어도 하나를 포함한다.
- <29> 제2 우선 순위 판단부(310)는 라운드-로빈 우선 순위 판단 방법을 사용하여 마스터의 종류 및 마스터에 할당된 시간 정보에 따라 일차적으로 우선 순위를 정한 후, 데이터의 처리 방식에 대한 정보에 따라 최종적인 마스터의 우선 순위를 결정한다.
- <30> 도 4a는 프로세서, DSP, DMA1, DMA2의 마스터가 버스 사용 요청을 한 경우, 제1 우선 순위 판단부(308)에 의해 판단된 우선 순위를 나타낸 도면이다.
- <31> 도 4a를 참조하면, 데이터의 처리 방식에 대한 정보가 버스트 여부인 경우, 마스터들의 우선 순위를 나타내고 있다. 버스트 여부가 “싱글”과 “버스트”로 나뉘게 됨으로써, 우선 순위의 종류는 마스터 종류의 두 배로 증가되며, 만약 마스터가 4개라면 우선 순위 개수가 8개가 된다. 결국, 버스트 여부에 따른 우선순위 요인까지 포함시키게 됨으로 해서, 우선순위 결정을 다양화할 수 있다.

- <32> 프로세서 또는 DSP는 싱글을 사용하는 경우, DMA1 또는 DMA2보다 우선 순위가 높다. 즉 마스터의 종류에 따라 일차적으로 우선 순위를 결정한다. 그러나, 프로세서 또는 DSP가 버스트를 사용하는 경우, 싱글을 사용하는 DMA1보다 우선 순위가 낮다. 따라서, 버스트 여부에 따라 최종적으로 우선 순위가 정해지게 된다.
- <33> 도 4b는 프로세서, DSP, DMA1, 및 DMA2의 마스터가 버스 사용 요청을 한 경우, 제2 우선 순위 판단부(310)에 의해 판단된 우선 순위를 나타낸 도면이다.
- <34> 도 4b를 참조하면, 데이터의 처리 방식에 대한 정보가 버스트 여부인 경우, 우선 순위를 나타내고 있다.
- <35> 각 마스터들은 할당 시간 동안 우선 순위를 갖게 되는데, 그 순서는 프로세서, DSP, DMA1, DMA2의 순서이다. 즉, 마스터의 종류에 따라 일차적으로 순서가 결정된다. 그러나, 각 마스터의 버스트 사용 여부에 따라 세부적인 우선 순위가 바뀌게 된다. 따라서, 버스트 여부에 따라 최종적으로 순서가 정해지게 된다. 여기서, 프로세서는 싱글을 사용하는 경우에 버스트를 사용하는 경우 보다 먼저 우선권을 부여 받게 된다.
- <36> 이하, 데이터의 처리 방식에 대한 정보가 버스트 여부일 경우, 도 5 내지 7을 통해 제1 우선 순위 판단부(308)의 구체적인 우선 순위 판단 방법을 살펴본다.
- <37> 도 5를 참조하면, 본 발명의 일 실시예에 따라 버스트 여부를 우선 순위의 판단에 이용하는 아비터가 실제로 버스의 성능을 향상 시키는 예를 나타내 위해서 버스의 동기화(Synchronization) 현상을 도시하였다. 마스터 1에 의해서 M11 명령(502)이 1 사이클 수행되면, 4 사이클 후에는 반드시 마스터 2에서 M21 명령(504)이 4사이클 이루어 지는 시스템이 있다고 가정하며, 마스터 1과 마스터 2는 서로 동기화 관계에 있다고 말할 수 있다.
- <38> 도 6a 및 도 6b는 이러한 도 5와 같은 동기화 관계에 있는 마스터 1과 마스터 2의 우선 순위에 따른 버스 성능의 차이를 나타내는 도면이다.
- <39> 도 6a에서는 싱글을 사용하는 마스터 1의 우선 순위가 버스트를 사용하는 마스터 2보다 높은 경우로서, 시간 1 구간에서 4구간을 제외하고 마스터 1 또는 마스터 2가 모든 시간에 효율적으로 버스를 점유하고 있다.
- <40> 마스터 1에 의해서 M11 명령(602a)가 수행되면, 이에 동기화되어 4 사이클 후에 마스터 2에 의해 M21 명령(606a)가 수행되어야 하지만, 마스터 1의 M12 명령(604a)이 우선권을 가지므로 먼저 수행되고, 그 다음에 M21 명령(606a)이 수행된다. 전술한 과정은 후속의 M13 명령(608a)과 M22 명령(610a) 사이에서도 마찬가지로 반복된다.
- <41> 반면에 도 6b에서는 싱글을 사용하는 마스터 1의 우선 순위가 버스트를 사용하는 마스터 2 보다 낮은 경우로서, 마스터 1의 명령들(602b,606b)이 수행된 후 마스터 2의 명령들(604b,608b)이 수행될 때까지, 반드시 4 사이클 동안 버스가 유휴상태를 유지한다.
- <42> 결국, 도 6a의 경우와 도6b의 경우를 비교해보면, 마스터의 종류와 버스트 여부에 따라 아비터에서 판단되는 우선 순위에 의해 버스의 성능이 현격하게 차이가 나타남을 확인할 수 있다.
- <43> 도 7a 및 도 7b는 마스터 1이 싱글과 버스트를 모두 사용하여 데이터 처리를 하는 경우를 나타내는 도면이다. 그리고, 도 7a 및 도 7b는 마스터 1의 명령이 싱글로 수행되는 경우, 4사이클 후에 마스터 2의 버스트 명령이 수행되는 것을 나타낸다.
- <44> 도 7a는 마스터 1의 우선 순위가 마스터 2 보다 높은 경우로서, 마스터 2의 버스 점유비율이 현격하게 떨어짐을 알 수 있다.
- <45> 도 7a에서, 마스터 1이 싱글 또는 버스트를 사용하는 모든 경우에 마스터 2보다 우선 순위가 높게 책정된다고 가정한다. 시간 0구간에서 M11 명령(702a)이 수행 되었지만, 이에 동기화된 마스터 2의 M21 명령(708a)은 우선 순위가 낮으므로, 마스터 1의 M12 명령(704a) 및 M13 명령(706a)가 수행된 이후에 수행되게 된다. M12 명령(704a)에 동기화된 마스터 2의 데이터 처리는 M14 명령(710a)의 수행이 끝난 후에야 시작될 수 있음을 쉽게 예상할 수 있다.
- <46> 반면에, 도 7b는 버스트 여부가 싱글일 경우 또는 버스트일 경우, 각각 마스터 1의 우선 순위를 다르게 책정한 것을 나타낸 도면이다.
- <47> 도 7b에서는 싱글을 사용하는 마스터 1의 우선 순위가 가장 높고, 다음으로 버스트를 사용하는 마스터 2의 우선 순위가 높으며, 버스트를 사용하는 마스터 1의 우선 순위가 가장 낮은 경우를 가정한다.
- <48> 도 7b를 참조하면, 마스터 1의 M11 명령(702b) 및 M12 명령(704b)에 상응하는 동기화 명령인 마스터 2의 M21 명

령(706b) 및 M22 명령(708b)은 도 7a의 경우와 비교하여 빠르게 처리됨을 알 수 있다. 우선 순위가 제일 낮은 마스터 1의 M13 명령(710b)은 제일 나중에 수행되게 된다.

- <49> 결국, 마스터 2의 버스트 명령(706b,708b)을 최우선적으로 처리하고자 한다면, 도 7b와 같이 버스트 여부에 따라 마스터 1의 우선순위를 결정하는 것이 마스터 2의 버스트 명령(706b,708b)을 최우선적으로 처리할 수 있는 가장 효율적인 방법임을 알 수 있다.
- <50> 도 8은 마스터들의 데이터 처리 방식에 대한 정보가 버스트 여부, 데이터 크기, 및 쓰기 데이터 여부를 포함하는 경우, 마스터들에 대한 우선 순위 판단 결과를 나타내는 도면이다.
- <51> 도시된 바와 같이, 버스를 통해 전송할 데이터가 쓰기 데이터인지 여부, 데이터의 크기가 바이트 또는 워드 단위 인지 여부, 또는 메모리와의 버스트 접근 여부에 따라 마스터들의 우선 순위가 변화할 수 있다.
- <52> 다시 도 3을 참조하면, 선택부(312)는 고정된 우선 순위 판단부(304), 라운드 로빈 우선 순위 판단부(306), 제1 우선 순위 판단부(306), 및 제2 우선 순위 판단부(308)의 마스터 1 및 마스터 2에 대한 우선 순위 판단 결과 중에서 1개의 판단 결과에 따른 우선권 부여 신호를 출력한다. 상기 우선권 부여 신호는 BusGrant_1 또는 BusGrant_2로서, BusGrant_1은 마스터 1에게 우선권을 부여한다는 신호이고, BusGrant_2는 마스터 2에게 우선권을 부여한다는 신호이다.
- <53> 상기 4개의 우선 순위 판단 결과는 Mode 신호에 의해 선택된다. 일 예로, Mode 신호가 0인 경우, 고정된 우선 순위 판단부(304)의 판단 결과에 따라 우선권 부여 신호를 출력할 수 있고, Mode 신호가 3인 경우, 제2 우선 순위 판단부(308)의 판단 결과에 따라 우선권 부여 신호를 출력 할 수 있다.
- <54> 도 9는 본 발명의 일 실시예에 따른 아비터의 버스 중재 방법을 나타낸 순서도이다.
- <55> 먼저, 마스터들로부터 마스터의 데이터 처리 방식에 대한 정보를 수신한다(S902). 여기서, 데이터는 아비터가 관장하는 버스를 통해 전송할 데이터를 말한다.
- <56> 일 실시예에 있어서, 데이터 처리 방식에 대한 정보는, 버스트 여부, 데이터 용량, 버스 전송 방식, 및 쓰기 데이터 여부 중 적어도 하나를 포함할 수 있다. 일 실시예에 있어서, 아비터가 관장하는 버스는 메모리 접근을 위한 버스일 수 있다.
- <57> 다음으로, 아비터는 마스터들의 버스 점유에 대한 우선 순위를 마스터의 종류 또는 마스터에 할당된 시간 정보에 따라 1차적으로 판단한다(S904).
- <58> 다음으로, 마스터의 데이터 처리 방식에 대한 정보에 따라 마스터들의 버스 점유에 대한 최종 우선 순위를 판단한다(S906).
- <59> 본 발명이 속하는 기술분야의 당업자는 본 발명이 그 기술적 사상이나 필수적 특징을 변경하지 않고서 다른 구체적인 형태로 실시될 수 있다는 것을 이해할 수 있을 것이다.
- <60> 그러므로, 이상에서 기술한 실시예들은 모든 면에서 예시적인 것이며 한정적인 것이 아닌 것으로 이해해야만 한다. 본 발명의 범위는 상기 상세한 설명보다는 후술하는 특허청구범위에 의하여 나타내어지며, 특허청구범위의 의미 및 범위 그리고 그 등가 개념으로부터 도출되는 모든 변경 또는 변형된 형태가 본 발명의 범위에 포함되는 것으로 해석되어야 한다.

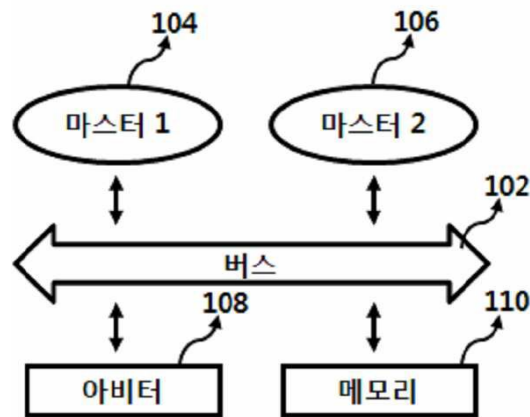
도면의 간단한 설명

- <61> 도 1의 일반적인 메모리 접근을 위한 버스 아키텍처.
- <62> 도 2a 및 도 2b는 일반적인 아비터의 우선 순위 판단 방법을 나타낸 도면.
- <63> 도 3은 본 발명의 일 실시예에 따른 버스 중재 장치를 나타낸 도면.
- <64> 도 4a 및 도 4b는 프로세서, DSP, DMA1, DMA2의 마스터가 버스 사용 요청을 한 경우, 마스터의 우선 순위를 나타낸 도면.
- <65> 도 5는 본 발명의 일 실시예에 따라 버스트 여부를 이용하는 아비터가 실제로 버스의 성능을 향상 시키는 예를 나타낸 도면.
- <66> 도 6a 및 도 6b는 마스터 1과 마스터 2의 우선 순위에 따른 버스 성능의 차이를 나타내는 도면.

- <67> 도 7a 및 도 7b는 싱글과 버스트를 사용하는 마스터 1과 마스터 2의 우선 순위에 따른 버스 성능의 차이를 나타내는 도면.
- <68> 도 8은 마스터들의 데이터 처리 방식에 대한 정보가 버스트 여부, 데이터 크기, 및 쓰기 데이터 여부를 포함하는 경우, 마스터들에 대한 우선 순위 판단 결과를 나타내는 도면.
- <69> 도 9는 본 발명의 일 실시예에 따른 아비터의 버스 중재 방법을 나타낸 순서도.
- <70> <도면의 주요부분에 대한 부호의 설명>
- <71> 302: 데이터 처리 정보 수신부 304: 고정된 우선 순위 판단부
- <72> 306: 라운드 로빈 우선 순위 판단부 308: 제1 우선 순위 판단부
- <73> 310: 제2 우선 순위 판단부 312: 선택부

도면

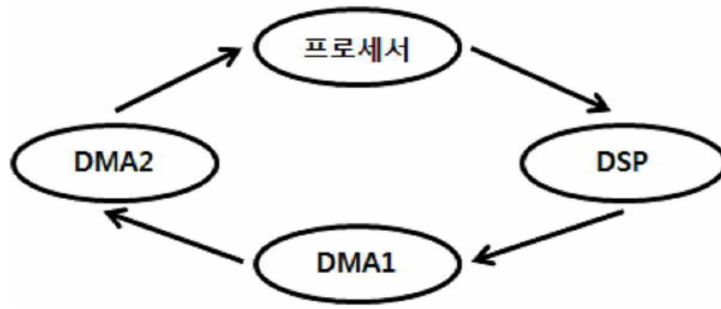
도면1



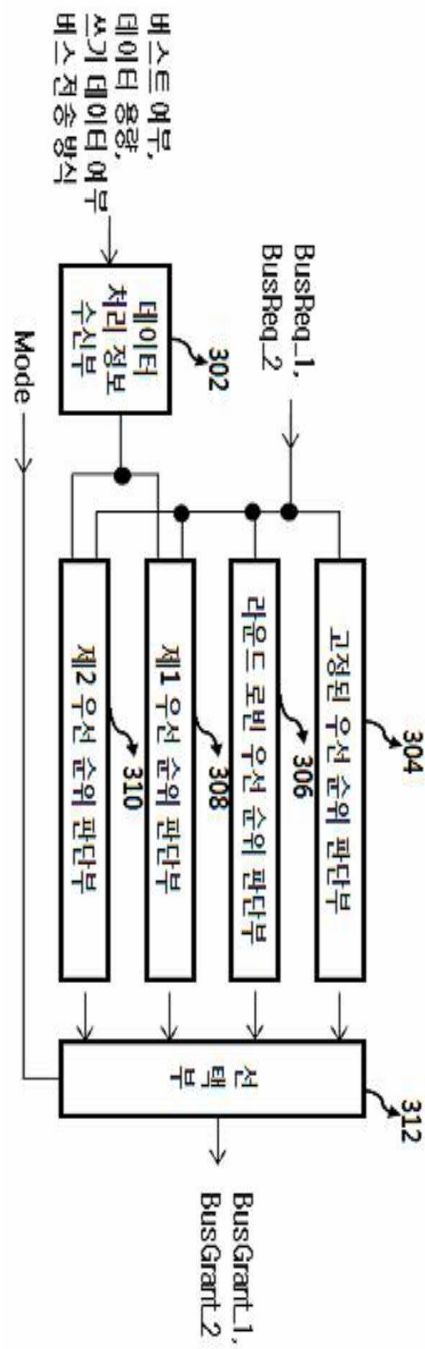
도면2a

우선 순위	마스터
1	프로세서
2	DSP
3	DMA1
4	DMA2

도면2b



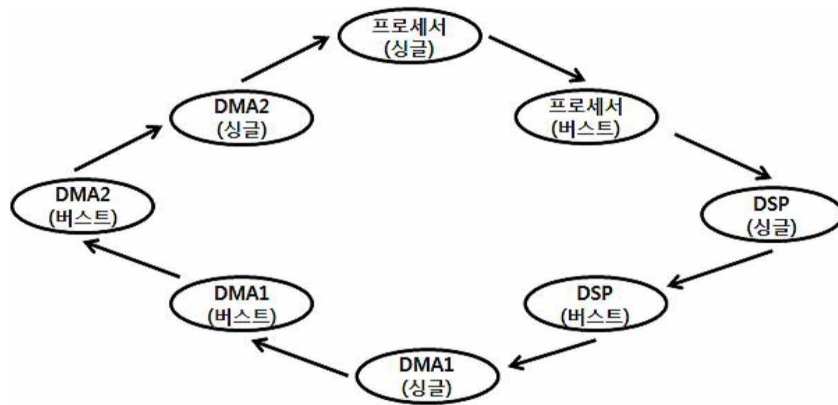
도면3



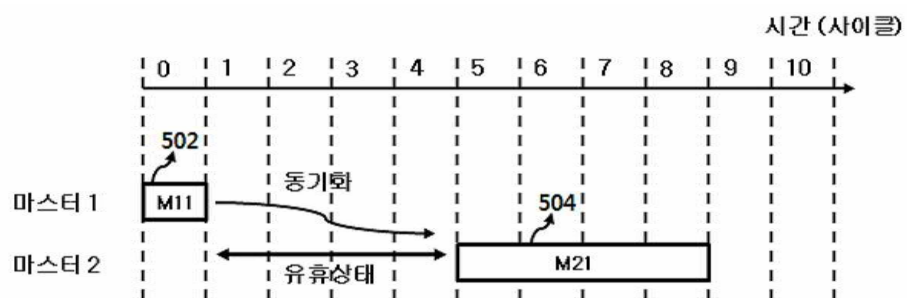
도면4a

우선 순위	마스터	버스트 여부
1	프로세서	싱글
2	DSP	싱글
3	DMA1	싱글
4	DMA1	버스트
5	프로세서	버스트
6	DSP	버스트
7	DMA2	싱글
8	DMA2	버스트

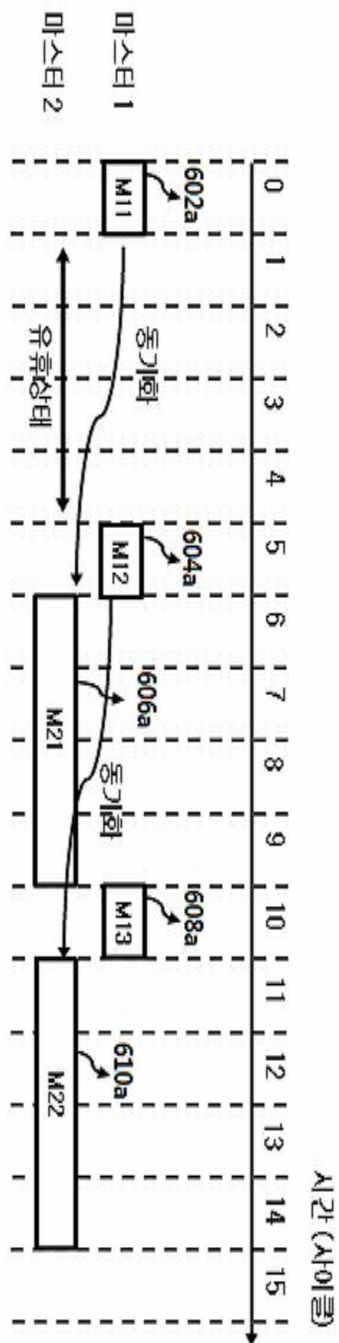
도면4b



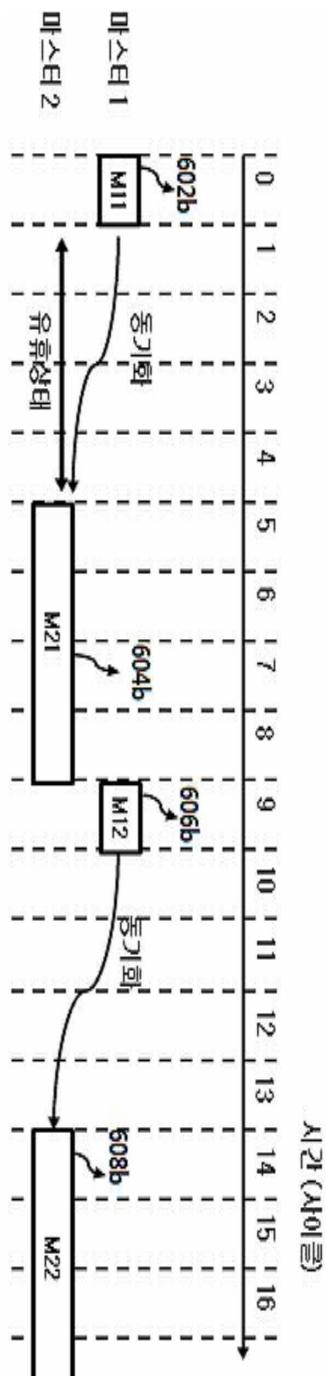
도면5



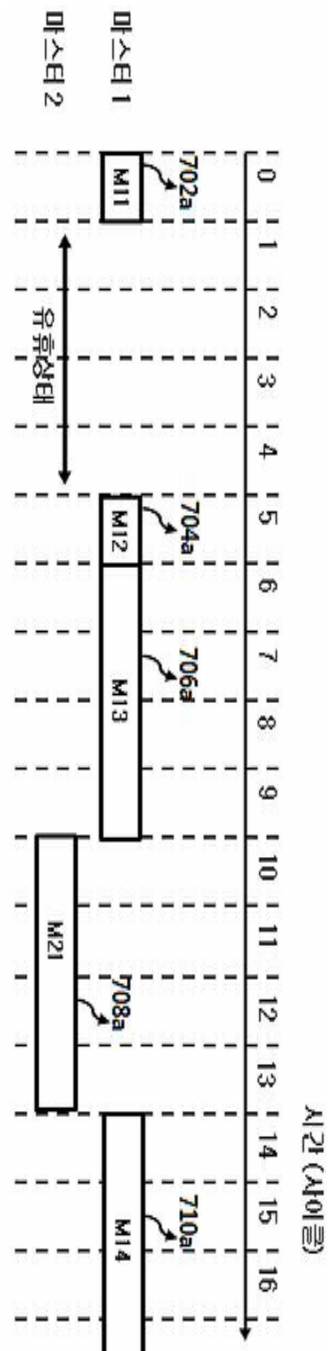
도면6a



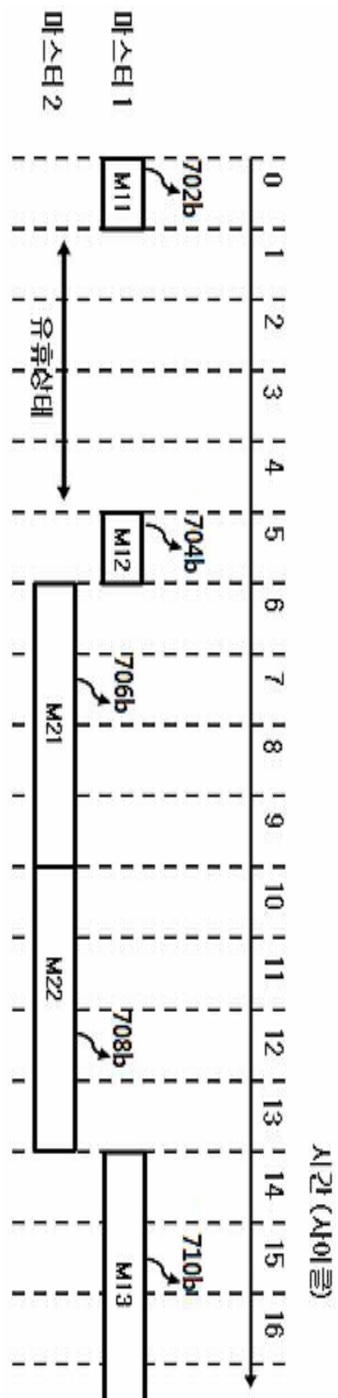
도면6b



도면7a



도면7b



도면8

우선 순위	마스터	버스트 여부	데이터 크기	쓰기 여부
1	프로세서	싱글	Word	쓰기
2	DSP	싱글	Byte	쓰기
3	DMA1	싱글	Half Word	읽기
4	DMA1	버스트	Word	쓰기
5	프로세서	버스트	Word	쓰기
6	DSP	버스트	Word	읽기
7	DMA2	싱글	Byte	읽기
8	DMA2	버스트	Byte	쓰기

도면9

