



(19)대한민국특허청(KR)
(12) 공개특허공보(A)

(51) 。 Int. Cl.

G06F 9/445 (2006.01)

G06F 9/46 (2006.01)

G06F 9/50 (2006.01)

G06F 9/38 (2006.01)

(11) 공개번호 10-2007-0058995

(43) 공개일자 2007년06월11일

(21) 출원번호 10-2006-0122423

(22) 출원일자 2006년12월05일

심사청구일자 2006년12월05일

(30) 우선권주장 JP-P-2005-00351012 2005년12월05일 일본(JP)

(71) 출원인 엔이씨 일렉트로닉스 가부시기가이샤
일본 211-8668 가나가와켄 가와사키시 나카하라쿠 시모누마베 1753

(72) 발명자 미나카미 데즈야
일본 가나가와켄 가와사키시 나카하라쿠 시모누마베 1753 엔이씨일렉
트로닉스 가부시기가이샤 나이

(74) 대리인 특허법인코리아나

전체 청구항 수 : 총 14 항

(54) 태스크를 동적으로 스케줄링할 수 있는 전용 신호 프로세서코어를 포함하는 프로세서 장치 및 그 태
스크 제어 방법

(57) 요약

프로세서 장치에서, 하나 이상의 범용 중앙 프로세싱 유닛 (11-1, 11-2, 11-3, 11A, 11B) 은 요청되어 새롭게 디스패치
된 태스크의 목적 코드를 메모리 (14a, 21a) 에 로드한다. 하나 이상의 전용 신호 프로세싱 유닛 코어 (12) 는 새롭게 디스
패치된 태스크의 발생 및 소멸을 동적으로 스케줄링하기 위해 메모리로부터 새롭게 디스패치된 태스크의 목적 코드를 다
운로드하고, 범용 중앙 프로세싱 유닛으로부터의 명령에 따라 현재 실행되는 태스크의 동작을 스케줄링한다.

대표도

도 1

특허청구의 범위

청구항 1.

요청되어 새롭게 디스패치된 태스크의 목적 코드를 메모리 (14a, 21a) 에 로드하도록 구성된 하나 이상의 범용 중앙 프로세싱 유닛 (11-1, 11-2, 11-3, 11A, 11B); 및

상기 새롭게 디스패치된 태스크의 발생 및 소멸을 동적으로 스케줄링하기 위해 상기 메모리로부터 상기 새롭게 디스패치된 태스크의 목적 코드를 다운로드하고, 상기 범용 중앙 프로세싱 유닛으로부터의 명령에 따라 현재 실행되는 태스크의 동작을 스케줄링하도록 구성된, 하나 이상의 전용 신호 프로세싱 유닛 코어 (12) 를 포함하는, 프로세서 장치.

청구항 2.

제 1 항에 있어서,

상기 각 명령의 포맷은 커맨드의 요청된 명령 콘텐츠 필드, 커맨드의 소스 필드, 커맨드의 결과 데이터의 수신지 필드, 및 상기 커맨드의 상기 요청된 명령 콘텐츠의 우선순위를 표시하는 필드에 의해 형성되는, 프로세서 장치.

청구항 3.

제 1 항에 있어서,

상기 커맨드가 전용 신호 프로세싱 요청 (REQ1, REQ2, REQ3) 인 경우, 상기 전용 신호 프로세싱 유닛 코어는 상기 새롭게 디스패치된 태스크의 각각의 목적 코드를 다운로드하는, 프로세서 장치.

청구항 4.

제 2 항에 있어서,

상기 커맨드가 프로세싱 시작 커맨드 (CMD1, CMD2, CMD3) 인 경우, 상기 전용 신호 프로세싱 유닛 코어는 상기 새롭게 디스패치된 태스크의 각각의 실행을 시작하는, 프로세서 장치.

청구항 5.

제 2 항에 있어서,

상기 커맨드가 프로세싱 종료 커맨드 (END1, END2, END3) 인 경우, 상기 전용 신호 프로세싱 유닛 코어는 상기 새롭게 디스패치된 태스크의 각각의 실행을 종료하는, 프로세서 장치.

청구항 6.

제 1 항에 있어서,

상기 커맨드가 전용 신호 프로세싱 요청 (REQ1, REQ2, REQ3) 인 경우, 상기 전용 신호 프로세싱 유닛 코어는 상기 새롭게 디스패치된 태스크의 각각의 목적 코드를 다운로드하고, 후속하여, 상기 새롭게 디스패치된 태스크의 각각의 실행을 시작하는, 프로세서 장치.

청구항 7.

제 1 항에 있어서,

상기 범용 중앙 프로세싱 유닛은 복수의 범용 프로세싱 유닛 코어 (11-1, 11-2, 11-3) 를 포함하며,
상기 범용 중앙 프로세싱 유닛 코어의 각각은,
중앙 프로세싱 유닛 (CPU1, CPU2, CPU3);
프로세서 엘리먼트 (PE1, PE2, PE3); 및
캐시 메모리 섹션 (CA1, CA2, CA3) 을 포함하는, 프로세서 장치.

청구항 8.

제 1 항에 있어서,
상기 범용 중앙 프로세싱 유닛은 대칭형 멀티프로세서 (symmetrical multiprocessor; 11B) 를 포함하며,
상기 대칭형 멀티프로세서는,
복수의 프로세서 엘리먼트 (PE1, PE2, PE3); 및
스누프 캐시 메모리 섹션 (SCM) 을 포함하는, 프로세서 장치.

청구항 9.

제 1 항에 있어서,
상기 범용 중앙 프로세싱 유닛은 단일 범용 프로세싱 유닛 코어 (11B) 를 포함하며,
상기 범용 중앙 프로세싱 유닛 코어는,
중앙 프로세싱 유닛 (CPU); 및
캐시 메모리 섹션 (CA1, CA2, CA3) 을 포함하는, 프로세서 장치.

청구항 10.

제 1 항에 있어서,
상기 메모리는 내부 메모리 (14) 의 공유 메모리 섹션 (14a) 을 포함하는, 프로세서 장치.

청구항 11.

제 1 항에 있어서,
상기 메모리는 외부 메모리 (21) 의 공유 메모리 섹션 (21a) 을 포함하는, 프로세서 장치.

청구항 12.

하나 이상의 범용 중앙 프로세싱 유닛 코어 (11-1, 11-2, 11-3, 11A, 11B) 및 하나 이상의 전용 신호 프로세싱 유닛 코어 (12) 를 포함하는 프로세서 장치를 위한 태스크 제어 방법으로서,

상기 범용 중앙 프로세싱 유닛 코어의 프로세스 (P1, P2, P3) 에 대응하는 전용 신호 프로세싱 요청 신호 (REQ1, REQ2, REQ3) 의 각각에 따라 상기 전용 신호 프로세싱 유닛 코어에 태스크의 목적 코드를 다운로드하는 단계;

상기 전용 신호 프로세싱 유닛 코어에 의해 상기 태스크의 실행을 시작하는 단계; 및

상기 전용 신호 프로세싱 유닛 코어가 상기 범용 중앙 프로세싱 유닛 코어로부터의 상기 전용 신호 프로세싱 요청 신호와 관련된 프로세싱 종료 신호 (END1, END2, END3) 의 각각을 수신한 경우에, 상기 태스크의 실행을 종료하는 단계를 포함하는, 태스크 제어 방법.

청구항 13.

제 12 항에 있어서,

상기 전용 신호 프로세싱 유닛 코어는, 상기 범용 중앙 프로세싱 유닛 코어로부터의 상기 전용 신호 프로세싱 요청 신호와 관련된 프로세싱 시작 신호의 각각의 수신에 응답하여 상기 태스크의 실행을 시작하는, 태스크 제어 방법.

청구항 14.

복수의 프로세서 엘리먼트 (11-1, 11-2, 11-3) 및 전용 신호 프로세싱 유닛 코어 (12) 를 포함하는 프로세서 장치에 대한 태스크 제어 방법으로서,

상기 전용 신호 프로세서 코어에 의해 상기 프로세서 엘리먼트 중 제 1 프로세서 엘리먼트의 제 1 프로세스 (P1) 로부터 요청되는 제 1 태스크를 실행하는 단계;

상기 전용 신호 프로세서 코어에 의해 상기 프로세서 엘리먼트 중 제 2 프로세서 엘리먼트의 제 2 프로세스 (P2) 로부터 요청되는 제 2 태스크를 실행하는 단계로서, 상기 제 1 태스크와 제 2 태스크는 병렬로 실행되는, 상기 제 2 태스크의 실행 단계; 및

상기 프로세서 엘리먼트 중 제 3 프로세서 엘리먼트의 제 3 프로세스 (P3) 에 대한 메모리를 해제하기 위해 상기 제 1 태스크의 실행을 종료하는 단계를 포함하는, 태스크 제어 방법.

명세서

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

본 발명은 하나 이상의 범용 중앙 프로세싱 유닛 (CPU) 코어 및 하나 이상의 디지털 신호 프로세싱 유닛 코어를 포함하는 프로세서 장치, 및 그 태스크 제어 방법에 관한 것이다.

최근에, 이동 전화는, 원칩 집적 회로에 의해 형성되는 단일 프로세서 장치로 결합되는, 원칩 집적 회로에 의해 형성되는 베이스밴드 프로세서 장치 및 원칩 집적 회로에 의해 형성되는 애플리케이션 프로세서 장치에 의해 구성된다.

원칩 장치에 의해 형성되는 종래 애플리케이션 프로세서는 하나 이상의 범용 중앙 프로세싱 유닛 (CPU) 코어, 및 소위 디지털 신호 프로세서 (DSP) 인 하나 이상의 전용 (specific) 신호 프로세싱 유닛 코어에 의해 구성된다. 예를 들어, 이동 전

화의 애플리케이션 프로세서에서, 범용 CPU 코어는 메일 표시 프로세싱 및 자바 (등록상표) 프로세싱과 같은 프로세싱을 수행하며, 전용 신호 프로세서 코어는 카메라 이미지의 데이터 압축 (JPEGenC/MPG4enc) 및 텔레비전 이미지의 데이터 확장 (MPEG4dec) 과 같은 프로세싱 (태스크) 을 수행한다.

종래 프로세서 장치 (일본공개특허공보 평7-287702A호 참조) 에서는, 범용 CPU 프로세서 코어 및 하나 이상의 전용 신호 프로세싱 유닛 코어 (DSP) 가 제공된다. 범용 CPU 코어는 미리 모든 가능한 태스크의 목적 코드를 메모리에 로드한다. 이후, 전용 신호 프로세싱 유닛 코어는 미리 메모리로부터 상술한 모든 목적 코드를 다운로드한다. 새롭게 디스패치된 태스크가 범용 CPU 코어에 의해 요청되는 경우, 새롭게 디스패치된 태스크에 대응하는 목적 코드 중 하나는 전용 신호 프로세싱 유닛 코어 중 하나에 의해 수행된다.

발명이 이루고자 하는 기술적 과제

상술한 종래 프로세서 장치에서, 목적 코드가 전용 신호 프로세싱 유닛 코어에 다운로드되지 않는 추가적인 태스크가 디스패치되면, 이러한 태스크를 실행하는 것은 불가능하다. 또한, 동시에 동작할 수 있는 복수의 전용 신호 프로세싱 유닛 코어는 제조 비용 및 전력 소비를 증가시킬 것이다.

일본 공개특허공보 평5-204828호는 범용 CPU 코어와 디지털 신호 프로세싱 유닛 코어 (DSP) 사이에 직접 메모리 접근 (direct memory access; DMA) 이 제공되는 프로세서 장치를 개시한다. 결과적으로, 범용 CPU 코어에 의해 디지털 신호 프로세싱 유닛 코어 (DSP) 에 요청되는 태스크는 그 능력 내로 제한된다.

본 발명에 의하면, 프로세서 장치에서, 하나 이상의 범용 중앙 프로세싱 유닛은 요청되어 새롭게 디스패치된 태스크의 목적 코드를 메모리에 로드한다. 하나 이상의 전용 신호 프로세싱 유닛 코어는, 새롭게 디스패치된 태스크의 발생 및 소멸을 동적으로 스케줄링하기 위해 메모리로부터 새롭게 디스패치된 태스크의 목적 코드를 다운로드하고, 범용 중앙 프로세싱 유닛으로부터의 명령에 따라 현재 실행되는 태스크의 동작을 동적으로 스케줄링한다.

본 발명은 첨부 도면을 참조하여 이하 개시되는 설명으로부터 명백히 이해될 것이다.

발명의 구성

본 발명에 따른 프로세서 장치의 제 1 실시형태를 도시한 도 1 에서, 프로세서 장치 (10) 는 3개의 범용 중앙 프로세싱 유닛 (CPU) 코어 (11-1, 11-2 및 11-3), 전용 신호 프로세싱 유닛 코어 (12), 인터럽트 제어기 (13), 및 원칩 메모리로 지칭되는 내부 랜덤 액세스 메모리 (RAM; 14) 를 포함하는 원칩 집적 회로에 의해 구성되며, 이들은 온칩 버스 (15) 에 의해 서로 접속된다. 또한, 프로세서 장치 (10) 는 온칩 버스 (15) 를 통해 외부 랜덤 액세스 메모리 (RAM; 21) 및 외부 읽기전용 메모리 (ROM; 22) 에 접속된다. ROM (22) 은 플래시 메모리로 대체될 수도 있다.

범용 CPU 코어 (11-1, 11-2 및 11-3) 는 개별 운영 체제 (operating system; OS) 의 제어하에 있다. 범용 CPU 코어 (11-1, 11-2 및 11-3) 의 각각은 하나의 중앙 프로세싱 유닛 (CPU1, CPU2 또는 CPU3), 하나의 프로세서 엘리먼트 (PE1, PE2 및 PE3) 및 하나의 캐시 메모리 섹션 (CM1, CM2 및 CM3) 에 의해 형성된다. 캐시 메모리 섹션 (CM1, CM2 및 CM3) 의 각각은 중앙 프로세싱 유닛 (CPU1, CPU2 및 CPU3) 에서 실행되는 명령, 테이블 데이터 등을 저장한다.

전용 신호 프로세싱 유닛 코어 (12) 는 프로세서 코어 섹션 (또는 DSP 코어 로직 섹션; 121) 및 캐시 메모리 섹션 (또는 DSP 코어 캐시 섹션; 122) 을 포함하는 풀 (full) 캐시형 디지털 신호 프로세서 (DSP) 이다. 이 경우에, 프로세서 코어 섹션 (121) 은 신호 프로세싱 엔진으로서 기능하고, 캐시 메모리 섹션 (122) 은 프로세서 코어 섹션 (121) 에서 실행되는 명령, 테이블 데이터 등을 저장한다.

내부 RAM (14) 및 외부 RAM (21) 은, 각각, 범용 CPU 코어 (11-1, 11-2 및 11-3) 및 전용 신호 프로세서 유닛 코어 (12) 용으로 보통 이용되는 공유 메모리 섹션 (14a 및 21a) 을 가진다.

도 1 의 프로세서 장치 (10) 가 이동 전화의 애플리케이션 프로세서로서 이용되는 경우, 범용 CPU 코어 (11-1, 11-2 및 11-3) 는 메일 표시 프로세싱 및 자바 (등록상표) 프로세싱과 같은 프로세싱을 수행하며, 전용 신호 프로세서 코어 (12) 는 카메라 이미지의 데이터 압축 (JPEGenC/MPG4enc) 및 텔레비전 이미지의 데이터 확장 (MPEG4dec) 과 같은 프로세싱을 수행한다. 도 1 에서, 전용 신호 프로세서 코어 (12) 에 의해 JPEG 목적 코드 또는 MPEG 목적 코드를 수행하기 위해, 범용 CPU 코어 (11-1, 11-2 및 11-3) 는 이들 목적 코드를 ROM (22) 으로부터 내부 RAM (14) 및/또는 외부 RAM (21) 의 공유 메모리 섹션 (14a 및/또는 21a) 에 미리 로드한다.

전용 신호 프로세서 유닛 코어 (12) 가 충분히 큰 명령 캐시 및 충분히 큰 데이터 캐시가 제공되는 풀 캐시형 디지털 신호 프로세서 (DSP) 이기 때문에, 종래 CPU 에서와 동일한 방법으로 프로세스 또는 태스크를 증가시키는 것이 가능하다. 이 경우에, 이러한 풀 캐시형 DSP는 프로세스 또는 태스크 스케줄링을 조종한다. 따라서, 이러한 풀 캐시형 DSP를 포함하는 작은 장치 또는 이동 전화의 소프트웨어 환경에서, 실행되는 모든 태스크는 미리 결정되고, DSP가 부팅되는 경우, 이들 모든 태스크의 목적 코드는 ROM (22) 으로부터 내부 RAM (14) 및/또는 외부 RAM (21) 의 공유 메모리 섹션 (14a) 및/또는 공유 메모리 섹션 (21a) 으로 전송된다.

또한, DSP의 운영 체제 (OS) 의 스케줄러는 새롭게 디스패치된 태스크를 동적으로 스케줄링할 수 있다. 즉, 스케줄러는 태스크의 동적 발생 및 소멸을 감독하여, 범용 CPU 코어 (11-1, 11-2 및 11-3) 로부터 요청되는 새롭게 디스패치된 태스크가 스케줄러에 등록되도록 하며, 동시에 현재 실행되는 태스크의 동작은 스케줄링된다. "디스패치"는 실행되는 프로세스 및 태스크에 프로세서 코어 섹션 (121) 의 동작 능력을 할당한다.

전용 신호 프로세싱 (태스크) 요청 커맨드와 같은 명령은 범용 CPU 코어 (11-1, 11-2 및 11-3) 로부터 전용 신호 프로세싱 유닛 코어 (12) 로 송신되고, 따라서 새롭게 디스패치된 태스크를 동적으로 스케줄링한다. 또한, 범용 CPU 코어 (11-1, 11-2 및 11-3) 로부터 전용 신호 프로세싱 유닛 코어 (12) 로 송신되는 프로세싱 시작 커맨드 및 프로세싱 종료 커맨드와 같은 명령은 현재 실행되는 태스크에 분배된다.

일반적으로, 하나의 명령 포맷은 커맨드의 요청된 명령 컨텐츠 필드, 커맨드의 소스 필드, 커맨드의 결과 데이터의 수신지 필드, 및 요청된 명령 컨텐츠의 우선순위를 표시하는 필드에 의해 형성된다.

도 1 의 프로세서 장치의 동작, 특히 전용 신호 프로세싱 유닛 코어 (DSP; 12) 의 태스크 실행 및 태스크 스케줄링 동작은 도 2a, 2b 및 2c 및 도 3 을 참조하여 이하 설명한다. 여기서, 범용 CPU 코어 (11-1) 가 도 3 에 도시된 프로세스 (P1) 를 수행하는 경우에 범용 CPU 코어 (11-1) 에 대한 전용 신호 프로세싱 (태스크) 을 스케줄링 및 실행하는데 단계 201 내지 206이 이용되고, 범용 CPU 코어 (11-2) 가 도 3 에 도시된 프로세스 (P2) 를 수행하는 경우에 범용 CPU 코어 (11-2) 에 대한 전용 신호 프로세싱 (태스크) 을 스케줄링 및 실행하는데 단계 207 내지 212가 이용되며, 범용 CPU 코어 (11-3) 가 도 3 에 도시된 프로세스 (P3) 를 수행하는 경우에 범용 CPU 코어 (11-3) 에 대한 전용 신호 프로세싱 (태스크) 을 스케줄링 및 실행하는데 단계 213 내지 218이 이용된다. 또한, 범용 CPU 코어 (11-1, 11-2 및 11-3) 는 상술한 개별 전용 신호 프로세싱 (태스크) 의 목적 코드를 ROM (22) 으로부터 내부 RAM (14) 및/또는 외부 RAM (21) 의 공유 메모리 섹션 (14a 및/또는 21a) 로 미리 로드한다.

이하, 단계 201 내지 206을 설명한다.

먼저, 단계 201에서, DSP (12) 가 범용 CPU 코어 (11-1) 로부터 전용 신호 프로세싱 요청 커맨드 (REQ1) 를 수신하였는지 여부를 판정한다. DSP (12) 가 이런 전용 신호 프로세싱 요청 커맨드 (REQ1) 를 수신한 경우에만, 제어는 단계 202로 진행한다. 그렇지 않으면, 제어는 단계 205로 진행한다.

예를 들어, DSP (12) 가 전용 신호 프로세싱 요청 커맨드 (REQ1) 를 수신한 경우인 도 3 의 시간 t11에, 제어는 단계 201로부터, 범용 CPU 코어 (11-1) 에 대한 전용 신호 프로세싱 (태스크)(T1) 의 목적 코드를 공유 메모리 섹션 (14a 또는 21a) 으로부터 캐시 메모리 섹션 (122) 으로 다운로드하는 단계 202로 진행한다. 따라서, 전용 신호 프로세싱 (태스크) (T1) 은 DSP (12) 에서 동적으로 발생된다.

다음으로, 단계 203에서, DSP (12) 는 전용 신호 프로세싱 요청 커맨드 (REQ1) 와 관련된 범용 CPU 코어 (11-1) 로부터의 프로세싱 시작 커맨드 (CMD1) 를 기다린다. DSP (12) 가 이런 프로세싱 시작 커맨드 (CMD1) 를 수신한 경우에만, 단계 202에서 다운로드된 목적 코드를 이용하여 전용 신호 프로세싱 (T1) 의 실행을 시작하는 단계 204로 제어가 진행된다.

예를 들어, DSP (12) 가 프로세싱 시작 커맨드 (CMD1) 를 수신한 경우인 도 3 의 시간 t12에, 제어는 단계 203로부터 단계 204로 진행한다.

한편, 단계 205에서, DSP (12) 가 전용 신호 프로세싱 요청 커맨드 (REQ1) 와 관련된 범용 CPU 코어 (11-1) 로부터의 프로세싱 종료 커맨드 (END1) 를 수신하였는지 여부를 판정한다. DSP (12) 가 이러한 프로세싱 종료 커맨드 (END1) 를 수신한 경우에만, 제어는 단계 206으로 진행한다. 그렇지 않으면, 제어는 단계 207로 진행한다.

예를 들어, DSP (12) 가 프로세싱 종료 커맨드 (END1) 를 수신한 경우인 도 3 의 시간 t13에, 제어는 단계 205로부터, 전용 신호 프로세싱 (T1) 의 실행을 종료하는 단계 206으로 진행한다. 따라서, 캐시 메모리 섹션 (122) 의 메모리 영역은 해제되어, 전용 신호 프로세싱 (태스크)(T1) 이 동적으로 단절된다.

단계 204 또는 206에서의 제어는 단계 207로 진행한다.

단계 203이 생략될 수 있다. 이러한 경우에, 전용 신호 프로세싱 (태스크)(T1) 의 목적 코드가 단계 202에서 캐시 메모리 섹션 (122) 에서 다운로드된 직후, 목적 코드는 단계 204에서 수행된다.

이하, 단계 207 내지 212를 설명한다.

먼저, 단계 207에서, DSP (12) 가 범용 CPU 코어 (11-2) 로부터 전용 신호 프로세싱 요청 커맨드 (REQ2) 를 수신하였는지 여부를 판정한다. DSP (12) 가 이러한 전용 신호 프로세싱 요청 커맨드 (REQ2) 를 수신한 경우에만, 제어는 단계 208로 진행한다. 그렇지 않으면, 제어는 단계 211로 진행한다.

예를 들어, DSP (12) 가 전용 신호 프로세싱 요청 커맨드 (REQ2) 를 수신한 경우인 도 3 의 시간 t21에, 제어는 단계 207로부터, 범용 CPU 코어 (11-2) 에 대한 전용 신호 프로세싱 (태스크) 의 목적 코드를 공유 메모리 섹션 (14a 또는 21a) 으로부터 캐시 메모리 섹션 (122) 으로 다운로드하는 단계 208로 진행한다. 따라서, 전용 신호 프로세싱 (태스크)(T2) 은 DSP (12) 에서 동적으로 발생된다.

다음으로, 단계 209에서, DSP (12) 는 전용 신호 프로세싱 요청 커맨드 (REQ2) 와 관련된 범용 CPU 코어 (11-2) 로부터 프로세싱 시작 커맨드 (CMD2) 를 기다린다. DSP (12) 가 이러한 프로세싱 시작 커맨드 (CMD2) 를 수신하는 경우에만, 단계 208에서 다운로드된 목적 코드를 이용하여 전용 신호 프로세싱 (T2) 의 실행을 시작하는 단계 210으로 제어가 진행된다.

예를 들어, DSP (12) 가 프로세싱 시작 커맨드 (CMD2) 를 수신하는 경우인 도 3 의 시간 t22에, 제어는 단계 209로부터 단계 210으로 진행한다.

한편, 단계 211에서, DSP (12) 가 전용 신호 프로세싱 요청 커맨드 (REQ2) 와 관련된 범용 CPU 코어 (11-2) 로부터의 프로세싱 종료 커맨드 (END2) 를 수신하였는지 여부를 판정한다. DSP (12) 가 이러한 프로세싱 종료 커맨드 (END2) 를 수신한 경우에만, 제어는 단계 212로 진행한다. 그렇지 않으면, 제어는 단계 213으로 진행한다.

예를 들면, DSP (12) 가 상술한 프로세싱 종료 커맨드 (END2) 를 수신한 경우인 도 3의 시간 t23에, 제어는 단계 211로부터, 전용 신호 프로세싱 (T2) 의 실행을 종료하는 단계 212로 진행한다. 따라서, 캐시 메모리 섹션 (122) 의 메모리 영역은 해제되어, 전용 신호 프로세싱 (태스크)(T2) 이 동적으로 단절된다.

단계 210 또는 212에서의 제어는 단계 213으로 진행한다.

단계 209는 생략될 수 있다. 이러한 경우에, 전용 신호 프로세싱 (태스크)(T2) 의 목적 코드가 단계 208에서 캐시 메모리 섹션 (122) 에서 다운로드된 직후, 목적 코드는 단계 210에서 수행된다.

이하, 단계 213 내지 218을 설명한다.

먼저, 단계 213에서, DSP (12) 가 범용 CPU 코어 (11-3) 로부터 전용 신호 프로세싱 요청 커맨드 (REQ3) 를 수신하였는지 여부를 판정한다. DSP (12) 가 이러한 전용 신호 프로세싱 요청 커맨드 (REQ1) 를 수신한 경우에만, 제어는 단계 214로 진행한다. 그렇지 않으면, 제어는 단계 217로 진행한다.

예를 들어, DSP (12) 가 전용 신호 프로세싱 요청 커맨드 (REQ3) 를 수신한 경우인 도 3 의 시간 t31에, 제어는 단계 213으로부터, 범용 CPU 코어 (11-3) 에 대한 전용 신호 프로세싱 (태스크)(T3) 의 목적 코드를 공유 메모리 섹션 (14a 또는 21a) 으로부터 캐시 메모리 섹션 (122) 으로 다운로드하는 단계 214로 진행한다. 따라서, 전용 신호 프로세싱 (태스크) (T3) 은 DSP (12) 에서 동적으로 발생된다.

다음으로, 단계 214에서, DSP (12) 는 전용 신호 프로세싱 요청 커맨드 (REQ3) 와 관련된 범용 CPU 코어 (11-3) 로부터의 프로세싱 시작 커맨드 (CMD3) 를 기다린다. DSP (12) 가 이러한 프로세싱 시작 커맨드 (CMD3) 를 수신한 경우에만, 단계 214에서 다운로드된 목적 코드를 이용하여 전용 신호 프로세싱 (T3) 의 실행을 시작하는 단계 216으로 제어가 진행된다.

예를 들어, DSP (12) 가 프로세싱 시작 커맨드 (CMD3) 를 수신한 경우인 도 3 의 시간 t32에, 제어는 단계 215로부터 단계 216으로 진행된다.

한편, 단계 217에서, DSP (12) 가 전용 신호 프로세싱 요청 커맨드 (REQ3) 와 관련된 범용 CPU 코어 (11-3) 로부터의 프로세싱 종료 커맨드 (END3) 를 수신하였는지 여부를 판정한다. DSP (12) 가 이러한 프로세싱 종료 커맨드 (END3) 를 수신한 경우에만, 제어는 단계 218로 진행된다. 그렇지 않으면, 제어는 단계 201로 복귀한다.

예를 들어, DSP (12) 가 프로세싱 종료 커맨드 (END3) 를 수신한 경우인 도 3 의 시간 t33에, 제어는 단계 217로부터, 전용 신호 프로세싱 (T3) 의 실행을 종료하는 단계 218로 진행된다. 따라서, 캐시 메모리 섹션 (122) 의 메모리 영역은 해제되어, 전용 신호 프로세싱 (태스크)(T3) 은 동적으로 단절된다.

단계 216 또는 218에서의 제어는 단계 201로 복귀한다.

단계 215는 생략될 수 있다. 이러한 경우에, 전용 신호 프로세싱 (태스크)(T3) 의 목적 코드가 단계 214에서 캐시 메모리 섹션 (122) 에서 다운로드된 직후, 목적 코드는 단계 216에서 수행된다.

도 3 에서, 전용 신호 프로세싱 (태스크)(T1 및 T2) 은 시간 t22로부터 시간 t13으로 병렬로 수행되고, 또한, 전용 신호 프로세싱 (태스크)(T2 및 T3) 은 시간 t32로부터 시간 t23으로 병렬로 수행된다. 이러한 경우에, 전용 신호 프로세싱 (태스크)(T1 및 T2) 의 합계에 요청되는 성능 및 전용 신호 프로세싱 (T2 및 T3) 의 합계에 요청되는 성능이 모두 DSP (12) 의 제한 성능보다 더 낮으면, 프로세싱의 양이 동적으로 증가하는 경우에도, 성능은 거의 변동하지 않을 것이다.

본 발명에 따른 프로세서 장치의 제 2 실시형태를 도시한 도 4 에서, 도 1 의 범용 중앙 CPU 코어 (11-1, 11-2 및 11-3) 는, 3개의 프로세서 엘리먼트 (PE1, PE2 및 PE3) 및 스누프 캐시 메모리 섹션 (SCM) 에 의해 형성되는 대칭형 멀티프로세서 (symmetrical multiprocessor; SMP) 인 범용 CPU 코어 (11A) 로 대체된다. 범용 CPU 코어 (11A) 는 하나의 운영 체제의 제어 (OS) 하에 있다. 스누프 캐시 메모리 섹션 (SCM) 은 각각 프로세서 엘리먼트 (PE1, PE2 및 PE3) 중 하나에 대한 캐시 블록 (미도시) 을 포함한다. 온칩 버스 (15) 의 메모리 액세스는 스누프 캐시 메모리 섹션 (SCM) 에 의해 모니터링되어, 스누프 캐시 메모리 섹션 (SCM) 의 캐시 블록 사이에서 데이터의 일관성을 유지하도록 한다.

도 4 의 프로세서 장치 (10) 의 태스크 스케줄링 동작 및 태스크 실행은 도 1 의 프로세서 장치와 유사하다. 이러한 경우에, 프로세서 엘리먼트 (PE1, PE2 및 PE3) 의 PE 수와 무관한 범용 CPU 코어 (11A) 에서 실행되는 각 개별 프로세스 또는 스레드 (thread) 는 전용 신호 프로세싱 요청을 발생시켜, 각각의 전용 신호 프로세싱 (태스크) 이 독립적으로 실행되도록 한다.

본 발명에 따른 프로세서 장치의 제 3 실시형태를 도시한 도 5 에서, 도 1 의 범용 중앙 CPU 코어 (11-1, 11-2 및 11-3) 는 단일 CPU 및 캐시 메모리 섹션 (CM) 을 포함하는 범용 CPU 코어 (11B) 로 대체된다. 범용 CPU 코어 (11B) 는 하나의 운영 체제 (OS) 의 제어하에 있다.

도 5의 프로세서 장치 (10) 의 태스크 스케줄링 동작 및 태스크 실행은 도 1 의 프로세서 장치 (10) 와 유사하다. 이러한 경우에, 범용 CPU 코어 (11B) 에서 실행되는 각 개별 프로세스 또는 스레드는 전용 신호 프로세싱 요청을 발생시켜, 각 전용 신호 프로세싱 (태스크) 이 독립적으로 실행되도록 한다.

요약하면, 본 발명의 특징은 다음과 같다.

1) 범용 CPU 코어 (11-1, 11-2, 11-3, 11A 및 11B) 는 DSP (12) 로 새롭게 디스패치된 태스크의 목적 코드를 ROM (22) 으로부터 내부 RAM (14) 및/또는 외부 RAM (21) 의 공유 메모리 섹션 (14a 및/또는 21a) 으로 로드한다.

2) DSP (12) 는 충분히 큰 명령 캐시 및 충분히 큰 데이터 캐시가 디스패치된 태스크를 수행하게 한다.

3) DSP (12)의 운영 체제(OS)는 범용 CPU 코어(11-1, 11-2, 11-3, 11A 및 11B)로부터의 전용 신호 프로세싱 요청 커맨드 및 프로세싱 종료 커맨드에 따라 전용 신호 프로세싱(태스크)의 동적 발생 및 소멸을 감독한다. 즉, 새롭게 디스패치된 전용 신호 프로세싱(태스크)이 스케줄링된다. 또한, 현재 실행되는 다른 전용 신호 프로세싱(태스크)의 동작이 스케줄링된다.

4) 범용 CPU 코어(11-1, 11-2, 11-3, 11A 및 11B)로부터 DSP(12)로 명령이 전송되어, 명령이 현재 실행되는 전용 신호 프로세싱(태스크)에 분배되도록 한다. 이러한 경우에, 하나의 명령 포맷은, 범용 CPU 코어(11-1, 11-2, 11-3, 11A 및 11B) 사이의 적절한 데이터 송수신을 가능하게 하는, 커맨드의 요청된 명령 콘텐츠 필드, 커맨드의 소스 필드, 커맨드의 결과 데이터의 수신지 필드, 및 요청된 명령 콘텐츠의 우선순위를 표시하는 필드에 의해 형성된다.

결과적으로, DSP(12)는, 소스, 즉 범용 CPU 코어(11-1, 11-2, 11-3, 11A 및 11B)중 하나로부터의 동적 요청에 따라 전용 신호 프로세싱(태스크)을 수행하여, 결과 데이터가 수신지, 즉 범용 CPU 코어(11-1, 11-2, 11-3, 11A 및 11B)중 하나에 송신될 수 있도록 한다. 이러한 경우에, 복수의 현재 실행되는 전용 신호 프로세싱(태스크)의 목적 코드뿐만 아니라 데이터가, 미스히트(mishit) 레이트가 작은 것으로 가정되는 DSP(12)의 캐시 메모리 섹션(122)에 다운로드되기 때문에, 현재 실행되는 전용 신호 프로세싱(태스크)의 수가 증가하더라도, 성능의 변동은 거의 변동되지 않는다.

또한, 전용 신호 프로세싱(태스크)의 다운로드, 실행 및 종료가 범용 CPU 코어로부터 각각, 전용 신호 프로세싱 요청 커맨드, 프로세싱 시작 커맨드 및 프로세싱 종료 커맨드에 의해 때때로 수행되기 때문에, 예상치 못한 전용 신호 프로세싱(태스크)이 용이하게 수행될 수 있다. 이러한 경우에, 이용가능한 메모리 양은 감소되어 전력 소비가 감소될 수 있다.

상술한 실시형태에서, 전용 신호 프로세싱 유닛 코어로서 오직 하나의 DSP가 제공되더라도, 하나 이상의 DSP는 전용 신호 프로세싱 유닛 코어 자체로 제공될 수 있다.

본 발명에 따른 프로세서 장치는 이동 전화의 애플리케이션 프로세서뿐 아니라 이동 전화의 베이스밴드 프로세서, 및 이동 전화의 애플리케이션 프로세서와 베이스밴드 프로세서에 의해 구성되는 단일 프로세서에 적용될 수 있다.

발명의 효과

본 발명에 의하면, 하나 이상의 범용 중앙 프로세싱 유닛 코어의 동적 요청에 따라 전용 신호 프로세싱을 수행하여, 결과 데이터가 범용 중앙 프로세싱 유닛 코어로 송신됨으로써, 현재 실행되는 전용 신호 프로세싱의 수가 증가하더라도, 성능의 변동을 최소화할 수 있다.

도면의 간단한 설명

도 1은 본 발명에 따른 프로세서 장치의 제 1 실시형태를 도시한 블록 회로 다이어그램.

도 2a, 2b 및 2c는 도 1의 프로세서 장치의 태스크 스케줄링 동작 및 태스크 실행을 설명하는 플로우차트.

도 3은 도 1의 프로세서 장치의 태스크 스케줄링 동작 및 태스크 실행을 설명하는 타이밍 다이어그램.

도 4는 본 발명에 따른 프로세서 장치의 제 2 실시형태를 도시한 블록 회로 다이어그램.

도 5는 본 발명에 따른 프로세서 장치의 제 3 실시형태를 도시한 블록 회로 다이어그램.

* 도면의 주요부분에 대한 부호의 설명 *

11-1, 11-2, 11-3: 범용 중앙 프로세싱 유닛

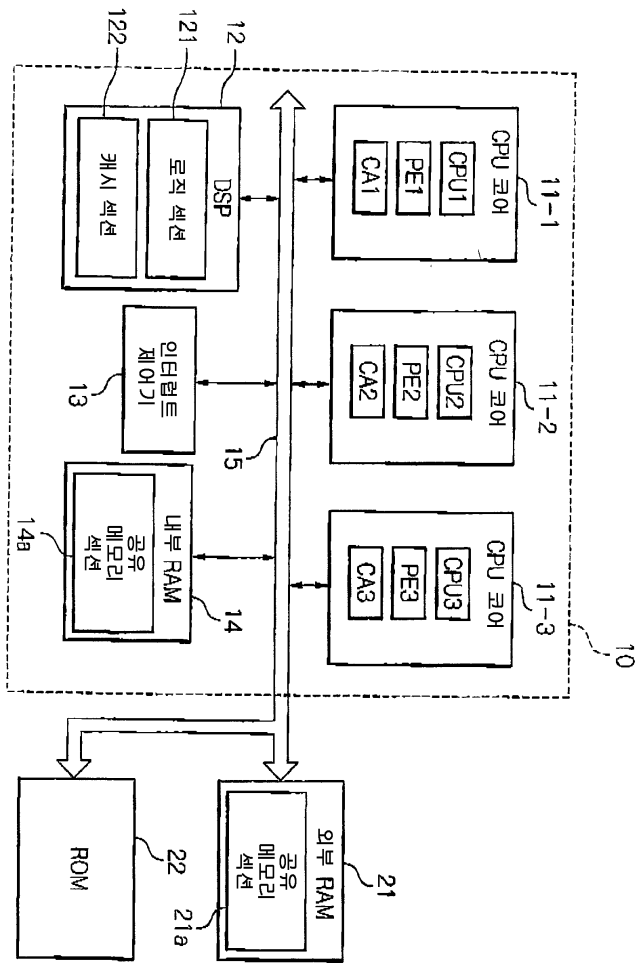
12: 전용 신호 프로세싱 유닛 코어

13: 인터럽트 제어기

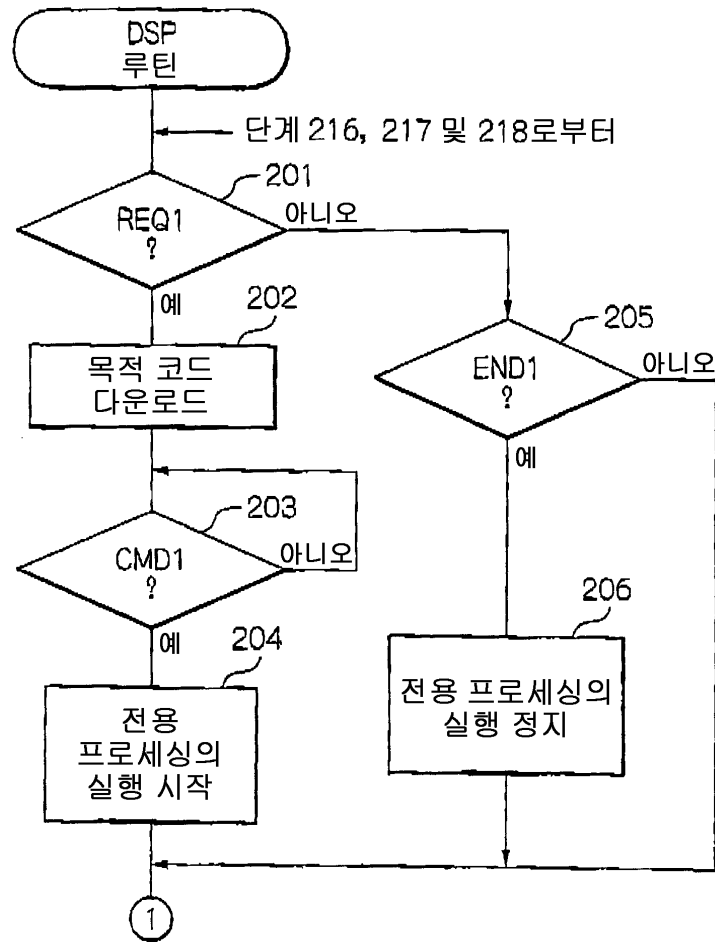
- 14: 내부 RAM
- 15: 온칩 버스
- 21: 외부 RAM
- 14a, 21a: 공유 메모리 섹션

도면

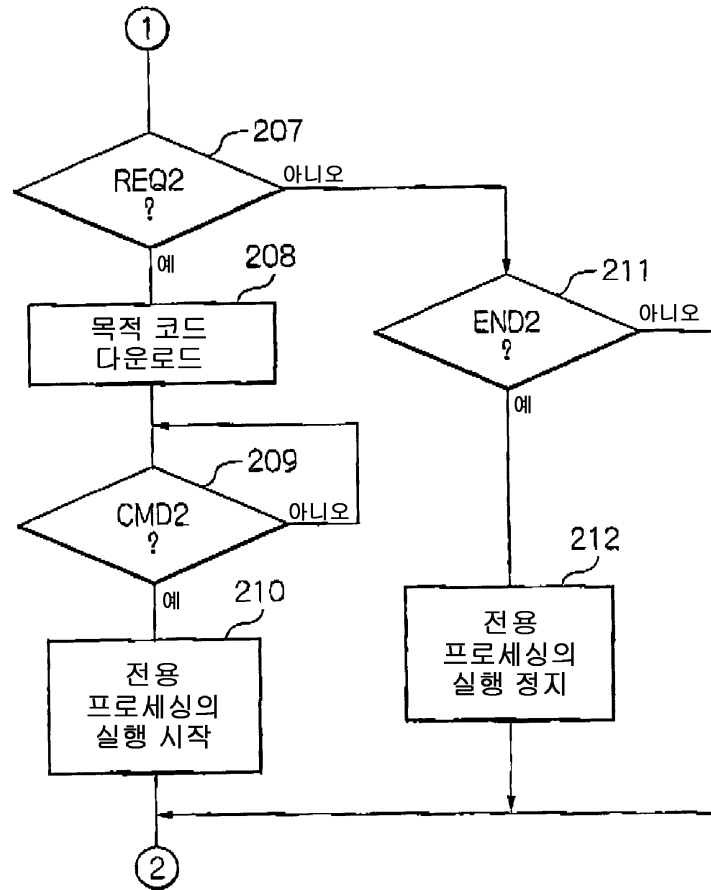
도면1



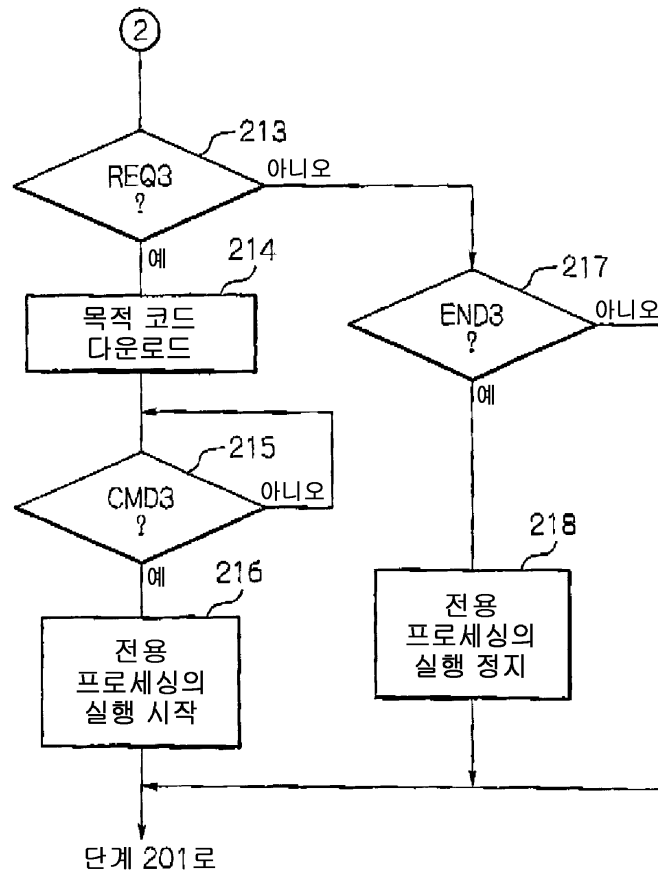
도면2a



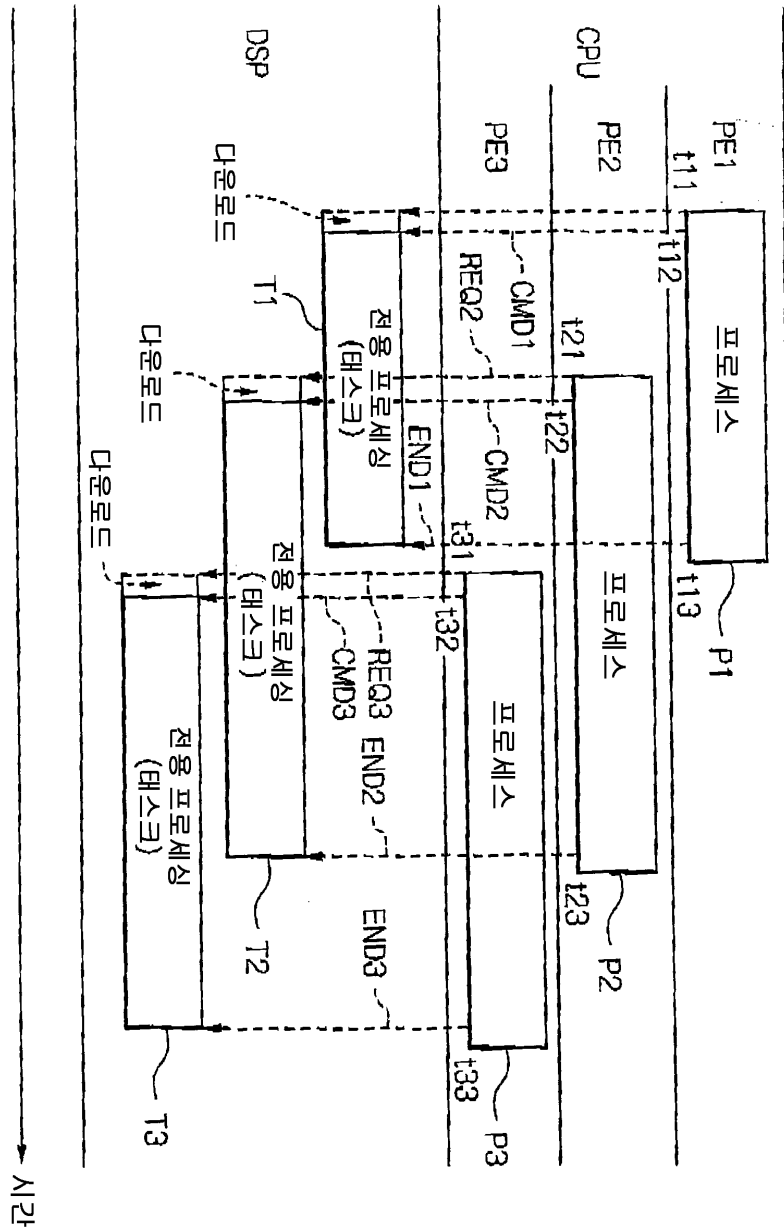
도면2b



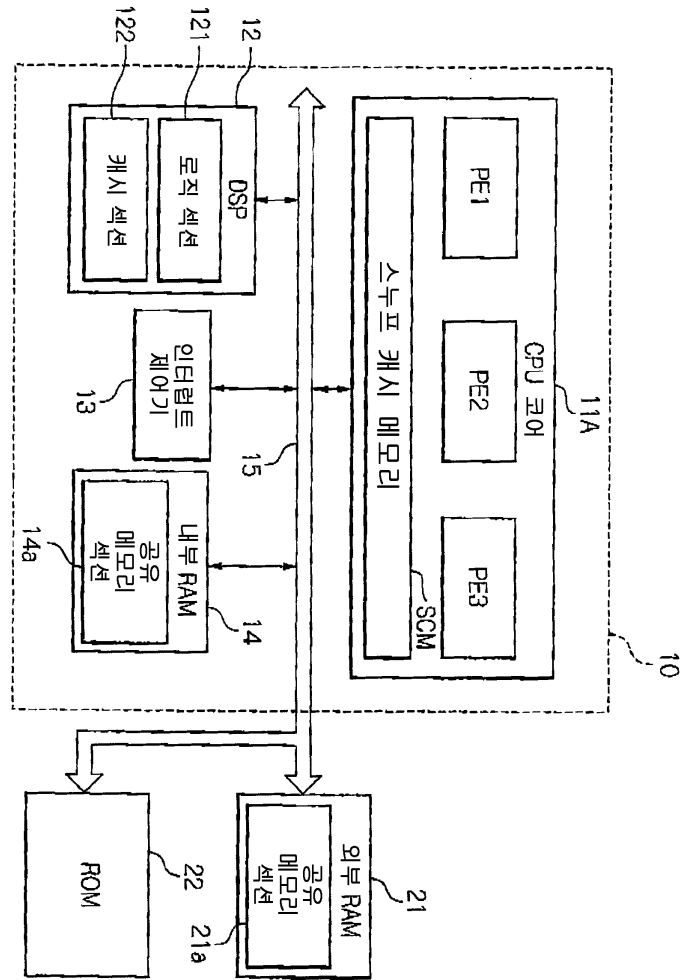
도면2c



도면3



도면4



도면5

