



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2007년09월21일
(11) 등록번호 10-0760976
(24) 등록일자 2007년09월17일

(51) Int. Cl.

G06F 7/00 (2006.01) G10L 19/00 (2006.01)

(21) 출원번호 10-2005-0070370
(22) 출원일자 2005년08월01일
심사청구일자 2006년01월27일
(65) 공개번호 10-2007-0015789
공개일자 2007년02월06일
(56) 선행기술조사문헌
JP2002026738 A
(뒷면에 계속)

(73) 특허권자

(주)펄서스 테크놀로지

경북 포항시 남구 효자동 산31 포항공과대학교 창업보육센터내

아주대학교산학협력단

경기도 수원시 영통구 원천동 산 5

(72) 발명자

선우명훈

서울특별시 서초구 반포동799

번지반포아파트68-501

오중훈

서울 양천구 신정동 312 목동 신시가지 아파트 924동1403호

문종하

경기도 수원시 팔달구 우만동 우만주공4차 아파트403-804호

(74) 대리인

함현경

전체 청구항 수 : 총 8 항

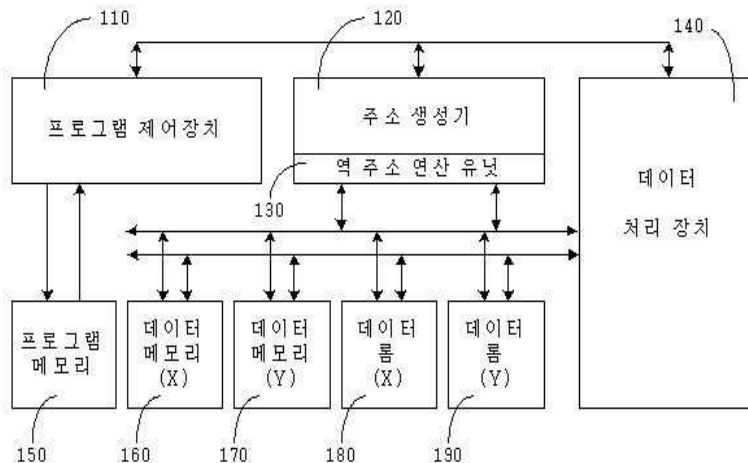
심사관 : 한선경

(54) 프로그래머블 프로세서에서 MPEG-2 또는 MPEG-4 AAC 오디오 복호 알고리즘을 처리하기 위한 연산 회로 및 연산 방법

(57) 요약

본 발명은 다채널 고음질 오디오 시스템에서 오디오 압축 알고리즘으로 사용되는 MPEG-2 AAC(Advanced Audio Coding) 또는 MPEG-4 AAC 알고리즘을 프로그래머블 프로세서에서 효율적으로 처리하기 위한 연산회로 및 연산방법에 관한 것이다. 이에 따르면, MPEG-2/4 AAC 알고리즘의 구현시 연산량의 비중을 많이 차지하는 IMDCT 과정을 효율적으로 수행할 수 있을 뿐만 아니라, 종래의 디지털 신호 처리 프로세서의 연산기 구조를 그대로 사용하고, 주소 생성기의 구조와 허프만 복호기 및 비트 처리 구조등을 추가하여 성능을 향상시킬 수 있으므로 프로그래머블 프로세서의 설계 및 변경이 용이하다.

대표도 - 도6



(56) 선행기술조사문헌
JP2002335230 A
KR1019970055597 A
KR1020010003385 A

특허청구의 범위

청구항 1

프로그램블 프로세서에서 MPEG-2 또는 MPEG-4 AAC 오디오 복호 알고리즘 연산을 처리하기 위하여,
 상기 MPEG-2 또는 MPEG-4 AAC 알고리즘의 연산 시작 신호를 발생시키고 상기 프로그램블 프로세서를 제어하는 프로그램 제어장치(110);
 상기 프로그램블 프로세서의 응용 프로그램이 저장되는 프로그램 메모리(150);
 상기 MPEG-2 또는 MPEG-4 AAC 알고리즘의 MDCT 또는 IMDCT 연산시 입력 데이터의 역주소를 생성시키기 위한 역주소 연산 유닛(130);
 연산을 위한 데이터가 저장되어 있는 데이터 메모리(160, 170);
 상기 역주소 연산 유닛(130)에서 생성된 역주소를 사용하여 상기 데이터 메모리(160, 170)의 주소를 계산하기 위한 주소 생성기(120);
 코사인 및 사인 데이터를 저장하고 있는 데이터 롬(180, 190);
 상기 데이터 메모리(160, 170) 및 롬 데이터를 이용하여 산술 논리 연산을 수행하는 데이터 처리장치(140); 및
 MPEG-2 또는 MPEG-4 디코딩 연산을 수행하기 위한 상태 레지스터;를 포함하는 것을 특징으로 하는 프로그램블 프로세서에서 오디오 복호 알고리즘을 처리하기 위한 연산 회로.

청구항 2

제 1항에 있어서,
 상기 데이터 처리 장치(140)는,
 데이터의 곱셈 연산 후 이를 누적시키는 2개의 곱셈 누적기;
 1개의 산술 논리 연산 장치;
 데이터 메모리의 값을 저장하는 입력 레지스터; 및
 연산 결과를 저장하고 다시 연산에 이용할 수 있도록 하는 누적기;를 포함하는 것을 특징으로 하는 프로그램블 프로세서에서 오디오 복호 알고리즘을 처리하기 위한 연산 회로.

청구항 3

제 1항 또는 제 2항에 있어서,
 상기 역주소 연산 유닛(130)은,
 상기 프로그램 제어 장치(110)의 상태 레지스터에 저장된 MDCT 또는 IMDCT 연산의 포인트 수를 입력받아 제어 신호를 생성시키는 제어신호 생성기(201);
 상기 제어 신호에 따라 입력된 주소 레지스터의 하위 14비트를 역변환하는 14개의 인버터(202~215); 및
 상기 제어 신호에 따라 최종 주소값을 선택하는 14개의 2-입력 멀티플렉서(216~229);를 포함하는 것을 특징으로 하는 프로그램블 프로세서에서 오디오 복호 알고리즘을 처리하기 위한 연산 회로.

청구항 4

제 3항에 있어서,
 상기 제어신호 생성기(201)는,
 LSB측 6비트를 생성시키는 1개의 8-입력 AND 게이트(301); 및
 상기 시작 신호에 따라 MDCT 또는 IMDCT 포인트 수를 입력받아 MSB측 1비트를 찾기 위한 7개의 2-입력 OR 게이트(302~308);를 포함하는 것을 특징으로 하는 프로그램블 프로세서에서 오디오 복호 알고리즘을 처리하기 위

한 연산 회로.

청구항 5

제 1항에 있어서,
 상기 데이터 처리 장치(140)는,
 스몰 쉬프트 연산을 처리하기 위한 2개의 곱셈기(401, 402);
 1개의 산술 논리 연산 장치(409);
 최대값, 최소값 및 절대값을 처리하기 위한 연산기(410);
 데이터 버스 스위치(400);
 16개의 입력 레지스터(411);
 Saturation/Limit/Round를 위한 데이터 처리 유닛(407); 및
 4개의 누적기(408);를 포함하는 것을 특징으로 하는 프로그래머블 프로세서에서 오디오 복호 알고리즘을 처리하기 위한 연산 회로.

청구항 6

제 3항에 있어서,
 상기 역 주소 연산 유닛의 제어신호 생성기는, 시작 신호에 따라 MDCT/IMDCT 포인트 수를 입력 받아 MSB쪽의 1비트를 찾기 위한 2입력 OR게이트 7개;
 64포인트 MDCT/IMDCT 이상을 지원하므로, LSB쪽 6비트를 빠르게 생성하기 위한 8입력 OR게이트 1개; 및
 상기 역주소 연산 유닛의 제어 신호를 생성하기 위한 연결선;을 포함하는 것을 특징으로 하는 프로그래머블 프로세서에서 오디오 복호 알고리즘을 처리하기 위한 연산 회로.

청구항 7

MPEG-2/4 AAC 알고리즘의 연산량에 근거하여 필터뱅크에 사용되는 IMDCT 연산의 전처리를 위해 연산 신호를 인가하는 단계;
 하나의 주소 레지스터에서 특정 주소 생성 규칙에 따라 두 개의 주소를 생성하는 단계;
 데이터 메모리와 롬 메모리로부터 데이터를 읽어 오는 단계; 및
 병렬적으로 전처리 과정에 필요한 버터플라이 연산을 수행하는 단계를 포함하는 것을 특징으로 하는 프로그래머블 프로세서에서의 MPEG-2/4 AAC 연산 방법.

청구항 8

제 7항에 있어서,
 상기 MPEG-2/4 AAC 디코딩 연산의 IMDCT 연산은 LDPRE 및 LD4 연산 명령어가 사용되어 수행되는 것을 특징으로 하는 프로그래머블 프로세서에서의 MPEG-2/4 AAC 연산 방법.

명세서

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

<21> 본 발명은 다채널 고음질 오디오 시스템에서 오디오 압축 알고리즘으로 사용되는 MPEG-2 AAC 또는 MPEG-4 AAC 알고리즘의 복호 연산을 디지털 신호 처리 프로세서 또는 마이크로프로세서 등과 같은 프로그래머블 프로세서에

서 효율적으로 처리할 수 있는 연산 회로 및 연산 방법에 관한 것이다.

<22> 최근 다채널 고음질 오디오에 대한 수요가 증가하면서 디지털 다채널 압축 오디오 알고리즘에 대한 관심이 증가하고 있다. ISO/IEC (International Standards Organization/International Electrotechnical Commission)는 1998년에 디지털 오디오와 비디오에 관한 압축기술을 연구하기 위한 ISO/MPEG(Moving Pictures Expert Group)을 창설하였다. 이 그룹은 1994년부터 MPEG-1 스테레오 포맷과의 호환이 필요 없는 응용 분야에서 사용할 수 있는 새로운 표준화 작업을 시작하였는데, 표준화 과정에서 붙여진 명칭은 MPEG-2 NBC(Non-Backward Compatible)였다. 표준화 과정에 들어가기 전에 MPEG은 MPEG-1과 호환되는 MPEG-2 BC(Backward Compatible)를 Dolby의 AC-3, AT&T의 MPAC과 비교 시험하였고, 그 결과 역 방향 호환 조건을 없앴으로써 부호화기의 성능을 향상시킬 수 있다는 결론을 얻었다. MPEG-2 NBC는 5채널 전대역폭 오디오 신호가 384kbit/s이하의 비트율로 ITU-R(International Telecommunication Union, Radiocommunication Bureau)에서 정의한 “청각적으로 구분할 수 없는(indistinguishable)” 오디오 품질에 도달하는 것을 목표로 하였다. 이후 MPEG-2 NBC는 1997년 4월에 새로운 다채널 오디오 부호화 방식의 국제 표준으로 발표되었는데, 발표될 때 그 명칭이 MPEG-2 AAC(Advanced (ISO/IEC 13818-7)로 바뀌었다. MPEG-2 AAC는 이러한 배경에 의해 규격화된 부호화 방식으로 5채널의 신호를 320kbps(1채널당 64kbps)의 고음질로 부호화하는 오디오 부호화 방식이다.

<23> 도 1은 종래 기술에 따른 MPEG-2 AAC 오디오 복호화 알고리즘의 과정을 나타낸 블록도이다. 도 1을 참조하면, MPEG-2 AAC 오디오 알고리즘은 384kbit/s이하의 비트율로 원음과 “청각적으로 구분할 수 없는” 오디오 음질을 제공하기 위해 고해상도 필터 뱅크, 예측 부호화, 음압 스테레오 부호화, TNS(Temporal noise shaping), 허프만 부호화(Huffman coding) 등을 결합시켜 사용한다. 이러한 MPEG-2 AAC 오디오 압축 알고리즘은 MDCT(Modified Discrete Cosine Transform)를 이용한 변환 부호화(Transform coding) 방식으로서, 변환된 신호를 압축하기 위해 심리 음향 모델을 기반으로 한 비트 할당 방식을 사용하여 변환된 신호를 압축한다.

<24> 한편, MPEG-2 AAC 오디오 시스템은 음질, 메모리 사용량 및 전력 요구량 사이의 손익을 고려하여 메인 프로파일(Main profile), LC 프로파일(Low complexity profile) 및 SSR 프로파일(Scalable sampling rate)의 3가지 프로파일을 지원한다.

<25> 먼저, 메인 프로파일은 주어진 비트율에서 최상의 음질을 제공하는 것으로서 이득 제어(Gain control) 톨을 제외한 AAC의 모든 톨이 사용된다. 메인 프로파일은 후술하게 될 LC 프로파일의 비트 스트림을 복호화할 수 있다.

<26> 다음으로, LC 프로파일은 일반적으로 가장 많이 사용되는 것으로, 예측 톨과 이득 제어 톨은 사용되지 않고 TNS의 차수도 제한되어 사용된다. LC 프로파일의 메모리의 사용량과 요구 전력은 메인 프로파일과 비교하여 상당히 작다는 것이 특징인데, 음질은 상대적으로 우수한 편이다.

<27> 마지막으로, SSR(Scalable sampling rate) 프로파일은 LC 프로파일에 이득 제어 톨을 추가한 것이다. 예측 톨은 사용되지 않고, TNS(Temporal noise shaping)의 차수와 대역폭도 제한되어 사용된다. SSR 프로파일은 메인 프로파일과 LC 프로파일보다 복잡도(Complexity)가 낮지만 주파수 가변 신호를 제공하는 장점이 있다.

<28> 고음질 오디오 압축 부호화 및 복호화 시스템의 가장 기본적인 요소는 시간 영역의 신호를 내부적인 시간 주파수 표현으로 변환하거나 또는 그 역변환을 수행하는 것이다. MPEG-2 또는 MPEG-4 AAC에서는 상기 변환 과정이 MDCT와 IMDCT(Inverse MDCT)에 의해 수행되는데, 상기 MDCT와 IMDCT는 시간 영역에서의 에일리어싱 제거(TDAC, Time Domain Aliasing Cancellation)라 불리는 기법이 적용된다.

<29> 상기 변환 부호화 과정은 도 2에 도시된 바와 같이 LC 프로파일 전체 연산량의 약 48%를 차지한다. AAC 오디오 복호기에서 사용되는 IMDCT는 하기의 수학적 식 1과 같다.

$$x(i) = \sum_{k=0}^{\frac{N-1}{2}} X(k) \cos\left[\frac{\pi}{2N} \left(2i+1 + \frac{N}{2}\right) (2k+1)\right], \text{ for } 0 \leq i \leq N-1$$

<30>

<31> 여기에서, N은 IMDCT의 연산 포인트 수, i는 시간 영역의 샘플 인덱스 값, k는 주파수 영역의 샘플 인덱스 값을 각각 나타낸다. 수학적식1에 나타난 바와 같이 IMDCT의 결과 값인 x(i) 샘플 하나를 얻기 위해서는 X(k)cos(·)값을 N/2 번 누적하여 더해야 한다. 상기 변환 부호화 과정을 수행하기 위해 수학적식1에 나타난 IMDCT의 정의식을 그대로 구현하는 것을 IMDCT의 직접 구현이라 한다. 한편, AAC에서 사용되는 IMDCT 연산 포인트 수는 장변환

(long block)에서 2048포인트, 단변환(short block)에서 256포인트가 각각 사용된다.

<32> 상기 수학적식1과 같이 IMDCT의 연산을 위해 직접 구현 방법이 사용될 수도 있지만, 통상적으로는 2^N 포인트의 IMDCT 연산에 대해 하드웨어 구현이 가장 용이하고 연산량이 작은 IMDCT 구현 알고리즘인 N/4 포인트 IFFT(Inverse Fast Fourier Transform)를 이용하여 연산되는 고속 IMDCT 알고리즘이 사용된다. 이러한 고속 IMDCT 알고리즘은 하기의 수학적식2 및 수학적식3에 따라 두 단계로 구성된다.

$$y(n) = \left[\sum_{k=0}^{\frac{N}{4}-1} \left\{ \left(X\left(\frac{N}{2}-2k-1\right) + j \cdot X(2k) \right) e^{j \frac{2\pi}{N} \left(k + \frac{1}{8} \right)} \right\} e^{j \frac{2\pi}{N/4} n} \right] e^{j \frac{2\pi}{N} \left(n + \frac{1}{8} \right)}$$

<33>

$$\begin{aligned} x(2n) &= -y_i\left(\frac{N}{8}+n\right), & x(2n+1) &= y_r\left(\frac{N}{8}-n-1\right) \\ x\left(\frac{N}{4}+2n\right) &= -y_r(n), & x\left(\frac{N}{4}+2n+1\right) &= y_i\left(\frac{N}{4}-n-1\right) \\ x\left(\frac{N}{2}+2n\right) &= -y_r\left(\frac{N}{8}+n\right), & x\left(\frac{N}{2}+2n+1\right) &= y_i\left(\frac{N}{8}-n-1\right) \\ x\left(\frac{3N}{4}+2n\right) &= y_i(n), & x\left(\frac{3N}{4}+2n+1\right) &= -y_r\left(\frac{N}{4}-n-1\right) \quad \text{for, } 0 \leq n < \frac{N}{8} \end{aligned}$$

<34>

<35> 상기 수학적식2에서 $\sum_{k=0}^{\frac{N}{4}-1} \{g\} e^{j \frac{2\pi}{N/4} nk}$ 은 N/4 포인트 IFFT 연산이다. 그리고, $(g) e^{j \frac{2\pi}{N} \left(k + \frac{1}{8} \right)}$ 와

$[g] e^{j \frac{2\pi}{N} \left(n + \frac{1}{8} \right)}$ 는 각각 IFFT 연산의 전처리, 후처리 과정이 된다.

<36> 또한, 상기 수학적식3은 역인터리빙(de-interleaving) 과정으로 $y_r = \text{real}\{y(n)\}$, $y_i = \text{image}\{y(n)\}$ 를 각각 의미한다.

<37> 일반적으로 대부분의 범용 DSP에서는 2^N 포인트 IMDCT를 적은 연산량으로 처리하기 위해 N/4 포인트 IFFT를 사용하는 고속 IMDCT 알고리즘을 사용한다.

<38> AAC의 IMDCT 연산 과정을 단계별로 나타낸 블록도인 도 3을 참조하면, 주파수 영역의 입력 신호 X(k)는 고속 IMDCT의 전처리 과정을 처리하기 위해 X(N/2-2k-1)와 X(2k) 두 개의 값을 이용하여 복소수 값인 X(N/2-2k-1) + jX(2k)을 만들게 된다. 즉, 전처리 과정을 위해 실수값으로 구성된 입력 신호 X(k)는 특정한 주소 생성 방식에 의해 복소수 값인 X(N/2-2k-1) + jX(2k)가 된다.

<39> 범용 DSP칩에서는 메모리에 저장된 X(k)값을 X(N/2-2k-1) + jX(2k)의 복소수 값으로 직접 나타내기 위한 특정 명령어 및 하드웨어 구조를 지원하지 않는다. 따라서, 고속 IMDCT 연산의 전처리 과정을 수행하기 위해 메모리에 저장된 실수 X(k)데이터를 특정 주소 형식으로 이동시키는 데이터 이동 사이클이 전체 연산의 많은 부분을 차지한다.

<40> 상기 수학적식2에 나타낸 바와 같이, 일반적으로 256포인트의 IMDCT를 고속 알고리즘으로 처리할 경우에도 도 3에 도시된 IMDCT 알고리즘에 따라 전처리 과정에서는 입력 신호 샘플을 복소수로 만든 X(N/2-2k-1) + jX(2k) 값과

$$e^{j\frac{2\pi}{N}(k+\frac{1}{8})}$$

값을 곱하게 된다. 이 때, N은 IMDCT 포인트 수로 256이며, k는 입력 인덱스 값으로 0부터 63까지의 수를 각각 나타낸다. MPEG-2 또는 MPEG-4 AAC 오디오 압축 알고리즘에서 사용되는 IMDCT 포인트 수는 각각 장변환에서 2048포인트, 단변환에서 256포인트이므로 경우에 따라 위 수식의 파라미터 값은 변경될 수 있다.

<41> 256포인트 IMDCT의 전처리 과정에서 입력 샘플을 복소수화하기 위해 k=0일 때 $X(127) + jX(0)$, k=1일 때 $X(125) + jX(2)$, k=2일 때 $X(123) + jX(4)$ 순으로 k값에 따라 DSP칩의 메모리에 저장된 X(k)데이터를 코어의 데이터 처리 장치로 이동시켜 복소수 연산을 처리해야 한다. 그러나, 범용 DSP칩을 이용할 경우에는 입력 샘플을 이동시키기 위해 두 개의 주소 레지스터가 할당되어, 각각에 대해 하나는 후감소 주소모드 (Post 2 decrement addressing mode), 또 다른 하나는 후증가 주소모드 (Post 2 increment addressing mode)를 사용하여 각 데이터를 다른 사이클로 이동시켜야 한다. 즉, 하나의 버터플라이 연산을 처리하도록 롬 데이터를 제외한 오디오 데이터를 구성하기 위해, 두 개의 주소 레지스터를 사용하여 최소 2사이클 이상의 시간을 소모해야 한다. 일반적인 상용 DSP칩에서는 후 감소 및 후 증가 주소 모드를 대부분 지원하므로 이를 사용하여 주소 생성을 더 효율적으로 수행할 수 있지만, 복소수 생성에 필요한 두 개의 데이터를 동시에 이동시킬 수 없는 단점이 있다.

<42> 현재 다채널 고음질 오디오 처리를 위한 상용 신호처리 프로세서(DSP)는 SHARC DSP의 ASDSP-21065L, Cirrus Logic의 CS49300, CS49500, TI (Texas Instrument)의 TMSc55x, TMSc64x, TMSc67x 시리즈, LSI Logic의 ZSP40x, CLARKSPUR의 CD2450, CD2480, 필립스 TriMedia의 TM-1300, PNX1500, Tensilica의 Xtensa 등이 있다. 뿐만 아니라, ARM사의 ARM9M, ARM9E 등도 AAC 처리가 가능하다. 이러한 상용 DSP 또는 프로세서는 대부분 다채널 또는 스테레오 채널에 대한 LC 프로파일까지 지원하고, TI TMSc67x, LSI Logic ZSP 시리즈, SHARC DSP ASDSP-21065L 등은 AAC의 메인 프로파일까지 지원이 가능하다.

<43> 일반적인 오디오 처리를 위한 상용 DSP는 24비트 또는 32비트의 데이터 표현 범위를 가지고 있고, 다채널 오디오 처리가 가능하도록 충분한 메모리 공간 또는 외부 오디오 신호의 I/O가 자유롭게 되도록 설계되어 있다. 또한, 대부분의 다채널 오디오용 DSP는 5.1채널이상의 오디오 데이터를 실시간으로 처리하기 위해 많은 하드웨어 자원을 병렬적으로 처리하는 방법을 사용한다. 예를 들어, SHARC DSP의 ASDSP-21065L 프로세서는 SIMD (Single instruction multiple data) 와 SISD (Single instruction single data) 처리가 모두 가능한 슈퍼 하버드 구조를 가지며, 많은 하드웨어 자원들이 병렬적으로 처리 가능하다. 또한, TMS320c64x, TMS320c67x, TM-1300, PNX1500 등은 VLIW (Very long instruction word) 프로세서로 비교적 많은 하드웨어 자원을 소프트웨어 인 컴파일러를 사용한 프로그램 제어를 통해 하드웨어를 병렬적으로 처리한다. 즉, 상용 DSP칩 개발사가 출시하는 대부분의 오디오 전용 DSP는 DSP 연산 코어가 슈퍼 하버드 구조 및 VLIW 구조를 갖고, 기본적으로 다양한 오디오 알고리즘에 대해 고속으로 동작할 수 있는 많은 연산기 및 하드웨어 자원을 가지고 있는 경우가 많다. 또한, DSP코어보다는 주변 장치가 오디오 I/O처리 등에 전용화되어, 오디오 신호 처리 연산을 위한 전용 명령어가 아닌 오디오 신호의 입/출력을 위한 주변 장치 제어가 가능하도록 전용의 명령어들이 존재하는 경우가 많다.

<44> 그러나, 이와 같은 대부분의 상용 DSP 코어는 그 구조의 특성상 상대적으로 코어의 크기 및 전력 사용량이 크며, SoC(System on a Chip)로 칩을 구현할 경우 그 크기로 인해 구현의 효율성이 떨어지는 단점이 있었다.

발명이 이루고자 하는 기술적 과제

<45> 이와 같은 문제점을 해결하기 위한 본 발명은 고음질 다채널 오디오 시스템에서 MPEG-2 AAC 또는 MPEG-4 AAC 알고리즘을 프로그래머블 프로세서에서 구현할 경우 고속의 고품질 오디오 신호 처리에 적합하면서 작은 칩 크기와 적은 전력 소모를 갖는 효율적인 오디오 복호 연산을 수행하는 연산 방법 및 회로를 제공하는 것을 목적으로 한다.

발명의 구성 및 작용

<46> 이를 구현하기 위해 본 발명에 따른 프로그래머블 프로세서에서 MPEG-2 또는 MPEG-4 AAC 오디오 복호 알고리즘 연산을 처리하기 위한 연산 회로는 MPEG-2 또는 MPEG-4 AAC 알고리즘의 연산 시작 신호를 발생시키고 상기 프로그래머블 프로세서를 제어하는 프로그램 제어장치, 프로그래머블 프로세서의 응용 프로그램이 저장되는 프로그램 메모리, MPEG-2 또는 MPEG-4 AAC 알고리즘의 MDCT 또는 IMDCT 연산시 입력 데이터의 역주소를 생성시키기 위한 역주소 연산 유닛, 연산을 위한 데이터가 저장되어 있는 데이터 메모리, 역주소 연산 유닛에서 생성된 역주소를 사용하여 데이터 메모리의 주소를 계산하기 위한 주소 생성기, 코사인 및 사인 데이터를 저장하고 있는 데이터 롬, 상기 데이터 메모리, 롬 데이터를 이용하여 산술 논리 연산을 수행하는 데이터 처리장치 및 MPEG-2 또

는 MPEG-4 디코딩 연산을 수행하기 위한 상태 레지스터를 포함하는 것을 특징으로 한다.

- <47> 또한, 본 발명에 따른 MPEG-2/4 AAC 알고리즘을 프로그래머블 프로세서에 적용하여 효율적으로 처리하기 위한 연산 방법은 MPEG-2/4 AAC 알고리즘의 연산량에 근거하여 필터 뱅크에 사용되는 IMDCT 연산의 전처리를 위해 연산 신호를 인가하는 단계, 하나의 주소 레지스터에서 특정 주소 생성 규칙에 따라 두 개의 주소를 생성하는 단계, 데이터 메모리와 롬 메모리로부터 데이터를 읽어 오는 단계 및 병렬적으로 전처리 과정에 필요한 버터플라이 연산을 수행하는 단계를 포함하는 것을 특징으로 한다.
- <48> 이하, 첨부된 도면을 참조하여 본 발명을 상세히 설명하기로 한다.
- <49> 도 4는 본 발명에 따른 프로그래머블 프로세서의 구성을 설명하기 위한 구성도이다. 도 4에 도시된 바와 같이, 새로운 역주소 모드 (Inverse addressing mode)를 수행하게 되면 하나의 데이터 주소 레지스터와 롬 테이블 주소 레지스터 및 각 해당 비트를 역변환하는 간단한 비트 조작 회로만으로 고속 IMDCT 연산의 전처리에 필요한 복소수 샘플을 1사이클 안에 메모리에서 일반 레지스터로 이동시킬 수 있어 매우 효율적이다.
- <50> 다음 과정으로 고속 IMDCT 알고리즘에서 N/4 포인트 IFFT와 후처리 과정이 끝나면, 앞서 살핀 도 3에 도시된 바와 같은 데이터의 역 인터리빙 과정을 거쳐 최종 $x(n)$ 샘플이 출력된다. N포인트 고속 IMDCT 과정에서 데이터 역 인터리빙 과정의 이전 단계까지 출력된 데이터의 총 샘플수는 N개이다. 그러나, IMDCT 연산 과정에서 최종 출력 샘플 개수는 입력 샘플 개수의 2배가 되므로, 데이터 역 인터리빙 과정을 통해서 N개의 데이터를 2N개로 재정렬하여 구성하게 된다. 예를 들어, 256포인트의 고속 IMDCT 과정의 경우, 전처리, N/4 포인트 IFFT 및 후처리 과정을 통해 생성된 데이터는 총 256개이다. 후처리 이후 단계인 데이터 역 인터리빙 과정은 후처리 과정에서 생성된 256개의 데이터를 메모리에서 읽어서, 고속 IMDCT 알고리즘의 역 인터리빙 정의인 수학식3에 따라 처리되면 512개의 최종 샘플이 생성된다. 즉, 메모리에서 1개의 데이터를 읽어서(Memory read) 그 데이터가 수학식3에 따라 처리된 후, 처리된 하나의 데이터를 특정 메모리 주소로 두 번 쓰게(Memory write)된다. 이와 같이, 데이터 역 인터리빙 과정은 메모리에 저장된 샘플값을 특정한 규칙에 의해 재정렬시키는 과정이다. 상기 과정에서 DSP칩 내의 연산기는 거의 사용되지 않고, 저장된 데이터를 메모리에서 읽고 쓰는 과정이 대부분이다. 상용 DSP의 경우, 고속 IMDCT 알고리즘의 데이터 역 인터리빙 과정을 수행하기 위해 메모리 읽기 및 쓰기 명령어가 반복적으로 사용된다.
- <51> 도 5는 본 발명에 따른 역주소 생성 과정을 설명하기 위한 구성도로서, 메모리 읽기 및 쓰기 과정에서 동시에 많은 개수의 데이터를 효율적으로 이동시킬 수 있도록 개선된 주소 생성기 유닛의 구조를 나타낸 것이다. 이와 같은 새로운 구조를 적용하게 되면 추가적인 하드웨어 자원을 최소화하더라도 MPEG-2/4 AAC 알고리즘의 효율적인 연산이 구현될 수 있다. 개선된 구조는 적은 하드웨어로 메모리 읽기 4개를 일반 연산 명령어와 병렬적으로 수행할 수 있고, 또는 메모리 쓰기 2개를 일반 연산 명령어와 병렬적으로 수행할 수 있다. 새로운 구조를 위해 필요한 추가적인 하드웨어는 롬 테이블의 주소 생성을 위한 14비트 카운터 2개이다. 추가된 14비트 카운터는 롬 테이블의 크기에 최적화되어 있고, 하드웨어 크기는 매우 작다. 개선된 구조의 사용으로 고속 IMDCT 알고리즘의 역 인터리빙 과정 및 고속의 데이터 이동이 필요한 응용 프로그램에서 효율적으로 메모리 대역폭을 확보할 수 있다.
- <52> 도 6은 본 발명에 따른 주소 생성기의 구조를 설명하기 위한 구성도이다. MPEG-2/4 AAC 알고리즘의 연산 시작 신호를 발생시키고 프로그래머블 프로세서를 제어하는 프로그램 제어장치(110), 상기 프로그래머블 프로세서의 응용 프로그램이 저장되는 프로그램 메모리(150), MPEG-2/4 AAC 알고리즘의 MDCT/IMDCT 연산시 입력 데이터의 역주소 생성 모드를 지원하기 위한 역주소 연산 유닛(130), 상기 역주소 연산 유닛(130)에서 계산된 주소를 사용하여 데이터 메모리의 주소를 계산하는 주소 생성기(120), 데이터가 저장되어 있는 데이터 메모리(160, 170), 코사인 및 사인 데이터를 저장하고 있는 데이터 롬(180, 190) 및 상기 데이터 메모리(160, 170)와 데이터 롬(180, 190)의 데이터를 이용하여 산술 논리 연산을 수행하는 데이터 처리 장치(140)로 구성된다. 이 때, 상기 데이터 처리 장치(140)는 데이터의 곱셈 연산 후 이를 누적시키는 2개의 곱셈 누적기와 1개의 산술 논리 연산 장치, 데이터 메모리의 값을 저장하는 입력 레지스터, 연산 결과를 저장하고 다시 연산에 이용할 수 있도록 하는 누적기로 구성된다.
- <53> 본 발명에 따른 명령어 및 수식은 IMDCT 연산의 고속 알고리즘 사용시 전처리 과정에서 데이터 메모리로부터 연산 데이터를 특정 주소 생성 방식에 의해 읽어오도록 하는 LDPRE(Load for Pre-processing)와 IMDCT 연산의 후처리 과정과 데이터 역 인터리빙 과정에서 데이터 메모리와 롬으로부터 동시에 4개의 데이터를 읽어오는 LD4(Load 4 sources)이다. 상기 명령어를 이용하면 MPEG-2/4 AAC 알고리즘을 복호하기 위해 사용되는 프로그래머블 프로세서의 연산량이 기존의 프로그래머블 프로세서보다 줄어들어 그 연산이 효율적으로 수행되며, 상용

DSP보다 비교적 작은 하드웨어 지원이 요구된다.

- <54> 상기 프로그램 제어장치(110)는 종래 프로그래머블 프로세서에서 프로그램을 제어하는 역할을 수행하는 동시에 LDPRE 명령어를 디코딩하여 프로그램 제어장치 내의 상태 레지스터에서 MDCT/IMDCT의 연산 포인트를 상기 역 주소 생성 유닛(130)에 전달하고, 역주소 생성 유닛(130)과 주소 생성기(120)에 역 주소 모드의 시작을 알리게 된다.
- <55> 도 7은 본 발명에 따른 역주소 연산 유닛의 구성을 설명하기 위한 구성도로서, LDPRE 명령어를 지원하기 위한 역주소 연산 유닛의 내부 구성을 나타낸 것이다. 상기 역주소 연산 유닛은 MPEG-2/4 AAC 알고리즘의 필터뱅크 과정에서 고속의 IMDCT/MDCT를 효율적으로 처리하기 위해 사용된다. 역주소 연산 유닛을 상세하게 나타낸 도 7을 살펴보면, 상기 프로그램 제어장치의 상태 레지스터에 저장된 MDCT/IMDCT 연산의 포인트 수를 입력받아 제어 신호를 생성시키는 제어신호 생성기(201), 주소 레지스터의 하위 14비트를 역변환하는 인버터 14개(202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215), 주소값 선택을 위한 멀티플렉서 14개(216, 217, 218, 219, 220, 221, 222, 223, 224, 225, 226, 227, 228, 229) 및 연결선으로 구성된다.
- <56> 도 8은 본 발명에 따른 역주소 연산 유닛의 제어신호 생성기의 구성을 설명하기 위한 구성도로서, 내부의 제어신호 생성기를 상세하게 나타낸 것이다. 도 8에 도시된 입력 데이터는 MDCT/IMDCT 포인트 수의 상위 MSB(Most Significant Bit)측의 8비트이다. 출력 데이터인 제어 신호는 총 14비트로서 앞서 살핀 도 7의 멀티플렉서를 제어하는 신호로 사용된다. 상기 제어신호 생성기(201)의 내부는 8개의 입력 포트를 가지는 AND 게이트 1개(301)와 2개의 입력 포트를 가지는 OR 게이트 7개(302, 303, 304, 305, 306, 307, 308) 및 연결선으로 구성된다.
- <57> 상기 역주소 연산 유닛에서 데이터의 주소를 생성하는 방법은 프로그램 제어장치에서 LDPRE 명령어 디코딩이 끝나면 상태 레지스터에 저장된 IMDCT/MDCT 포인트 수를 제어신호 생성기(201)의 입력 포트에 상위 8비트만 보내는 단계, 제어신호 생성기에서 IMDCT/MDCT 포인트 수에 따라 제어신호 14비트가 생성되는 단계, 제어신호가 역주소 연산 유닛 내의 멀티플렉서 선택 신호로 입력되는 단계, 주소 데이터 14비트가 멀티플렉서를 거쳐 출력되는 단계를 포함하는 것을 특징으로 한다.
- <58> 상기 역주소 연산 유닛에서 생성된 역 주소값은 역연산되어 생성되기 전의 주소값과 함께 프로그래머블 프로세서의 주소 생성기 내에 위치한 오프셋 레지스터의 입력으로 사용되어 기본 베이스 주소와 함께 연산되어 주소로 사용된다.
- <59> 일반적인 상용 프로그래머블 프로세서는 고속 IMDCT 연산 알고리즘을 전처리하기 위해 두개의 데이터 주소값을 각각 생성시켜야 하며 그 중 하나의 오프셋 레지스터는 0부터 2씩 증가시켜야(+2 post increment)하고, 다른 오프셋 레지스터는 [포인트수/2]부터 2씩 감소시켜야(-2 post decrement) 한다. 이 때, 종래의 프로그래머블 프로세서는 각 주소 생성을 위해 산술 연산 장치 혹은 주소 생성기 내의 모듈로(Modulo) 연산 장치를 사용해야 하므로 본 발명에 따른 구조와 비교하여 연산량 및 전력 소모가 많게 되어 효율적이지 못하다.
- <60> 도 9는 본 발명에 따른 산술 논리 연산기 내부에서 비트 추출 과정을 설명하기 위한 구성도로서, MPEG-2/4 AAC 알고리즘의 복호 연산을 효율적으로 수행하기 위한 데이터 처리장치를 나타낸 것이다. 상기 데이터 처리장치(140)는 2개의 스몰 쉬프트(Small shift) 연산을 지원하는 2개의 곱셈 누적기(401, 402, 403, 404, 405, 406); 1개의 산술 논리 연산 장치; 최대값, 최소값 및 절대값을 처리하기 위한 연산기(409, 410); 데이터 버스 스위치(400); 16개의 입력 레지스터(411); Saturation/Limit/Round를 위한 데이터 처리 유닛(407); 및 4개의 누적기(408)로 구성된다.
- <61> 본 발명에 따른 곱셈 누적기는 누적기를 이용하기 위하여 곱셈기를 거치지 않고 버스 스위치에서 입력을 받을 수 있는 논리 네트워크 구조를 지원한다.
- <62> 데이터 처리 장치의 경우 메모리로부터 읽은 데이터를 16개의 입력 레지스터에 저장하여 사용하고, 역양자화 과정에서 나눗셈 및 곱셈 연산을 효율적으로 처리하도록 곱셈 및 덧셈 전후에 쉬프트 연산을 지원하는 스몰 쉬프트를 지원한다. 전체 데이터 비트수는 오디오 알고리즘에 효율적인 24비트로 구성하거나 디지털 오디오의 이퀄라이저(equalizer)와 같은 후처리 과정을 고성능으로 처리하기 위해 32비트로 구성할 수 있다.

발명의 효과

- <63> 본 발명은 상술한 바와 같이 MPEG-2/4 AAC 알고리즘을 효율적으로 처리하기 위한 연산 회로 및 연산 방법을 제공하여, MPEG-2/4 AAC 알고리즘의 구현시 연산량의 비중을 많이 차지하는 IMDCT 과정을 효율적으로 수행할 수 있게 된다. 또한, 종래의 디지털 신호 처리 프로세서의 연산기 구조를 그대로 사용하고, 주소 생성기의 구조와

허프만 복호기 및 비트 처리 구조등을 추가하여 성능을 향상시킬 수 있으므로 프로그래머블 프로세서의 설계 및 변경이 용이하다.

【표 1】

<64>

구분	설명
LDPRE	ldpre GR0, AR0.x, ROM0A GR0 ← MEM[AR0.x], GR1 ← MEM[inversion of AR0.x], GR2 ← ROM0[ROM0A], GR3 ← ROM1[ROM0A]. 다음 사이클에서, AR0 주소값은 +2 증가, ROM0A는 +1 증가.
LD4	ld4 AR3.x+, ROM0A, AR4.y+, ROM1A GR0 ← MEM[AR3.x+], GR1 ← ROM0[ROM0A], GR2 ← MEM[AR4.y+], GR3 ← ROM1[ROM0B].

<65>

상기 표 1은 MPEG-2/4 AAC 알고리즘을 효율적으로 처리하기 위해 제안된 전용의 명령어들과 동작 내용을 상세히 나타낸 것이다. 제안된 프로그래머블 프로세서는 상기 전용의 명령어를 지원하도록 고안되어 있다.

【표 2】

<66>

고속 IMDCT 처리 과정	연산 사이클
전처리 과정(Pre-processing)	$[N/2 * 2] + 3$
N/4-포인트 IFFT	$(2N/2) * \log_2 N + 8$
후처리 과정(Post-processing)	$[N/2 * 2] + 6$
데이터 역인터리빙 과정	$[N/8 * 5] * 2 + 12$

<67>

상기 표 2는 MPEG-2/4 AAC 알고리즘의 필터 뱅크 과정인 IMDCT 처리를 고속 알고리즘을 이용하여 수행할 때 보이는 연산 사이클을 나타낸 것이다. 상기 표 2에서 알 수 있듯이, 제안된 프로그래머블 프로세서 구조로 2048 포인트 IMDCT를 처리할 경우에 하기의 수학적 4에 의해 총 11,294 사이클이 하나의 오디오 채널에 대해서 필요하게 된다.

$$=(2048+3)+(2048+6)+(5*2048/4+12)+(2048/4)*\log(2048/4)+9$$

$$=[(13*2048/4)+(2048/4)*\log(2048/4)+30]$$

$$=11,294 \text{ cycles}$$

【표 3】

<69>

프로세서	수행 시간	연산 사이클	MIPS
국내 오디오 전용 DSP	1.3312ms	53,248	n.a.
대만 오디오 전용 VLSI	n.a.	32,768	n.a.
TMS320c62x	n.a.	n.a.	7.5
ADSP-21060	9ms	n.a.	n.a.
본 발명의 구조	150.88us	22,588	1.0588

<70>

상기 표 3은 제안된 방법 및 하드웨어 구조를 사용하여 IMDCT 연산 수행시 종래의 프로그래머블 프로세서 칩과 연산 사이클, 수행 시간, MIPS(Million Instructions per Second)의 3가지 기준을 비교한 것이다. 일부 공개되

지 않은 항목은 제외하였으며, 성능 분석 결과 본 발명에서는 메모리에서 데이터를 효율적으로 이동시킬 수 있으므로 텍사스 인스트루먼트사의 TMS320c62x DSP 코어의 연산량에 14% 만을 가지고도 동일한 성능을 보이며, 국내 오디오 전용 DSP 코어 및 대만의 ASIC 칩과 비교하였을 때 기존의 칩에 비해 약 42.4% 및 68.9%의 연산 사이클을 가지고도 동일한 성능을 보임을 확인할 수 있었다. 또한, ADSP-21060 코어의 경우 동일한 연산을 9ms시간에 처리함에 비해, 본 발명은 150.88us에 처리하는 우수한 성능을 보였다.

- <71> 이상에서 설명한 바와 같이, 제안된 명령어와 하드웨어 구조는 기존의 연산 모듈들을 재사용하고 데이터 처리 회로 및 주소 생성 흐름 제어 등만 추가하면 되므로 설계비용 측면에서도 경제적이며, 연산 속도 측면에서도 매우 효율적으로 MPEG-2/4 AAC 알고리즘을 구현할 수 있다.
- <72> 이로 인해 종래 프로그래머블 프로세서에서 가지는 단점을 보완하고, 효율적으로 MPEG-2/4 AAC 알고리즘을 수행할 수 있다.

도면의 간단한 설명

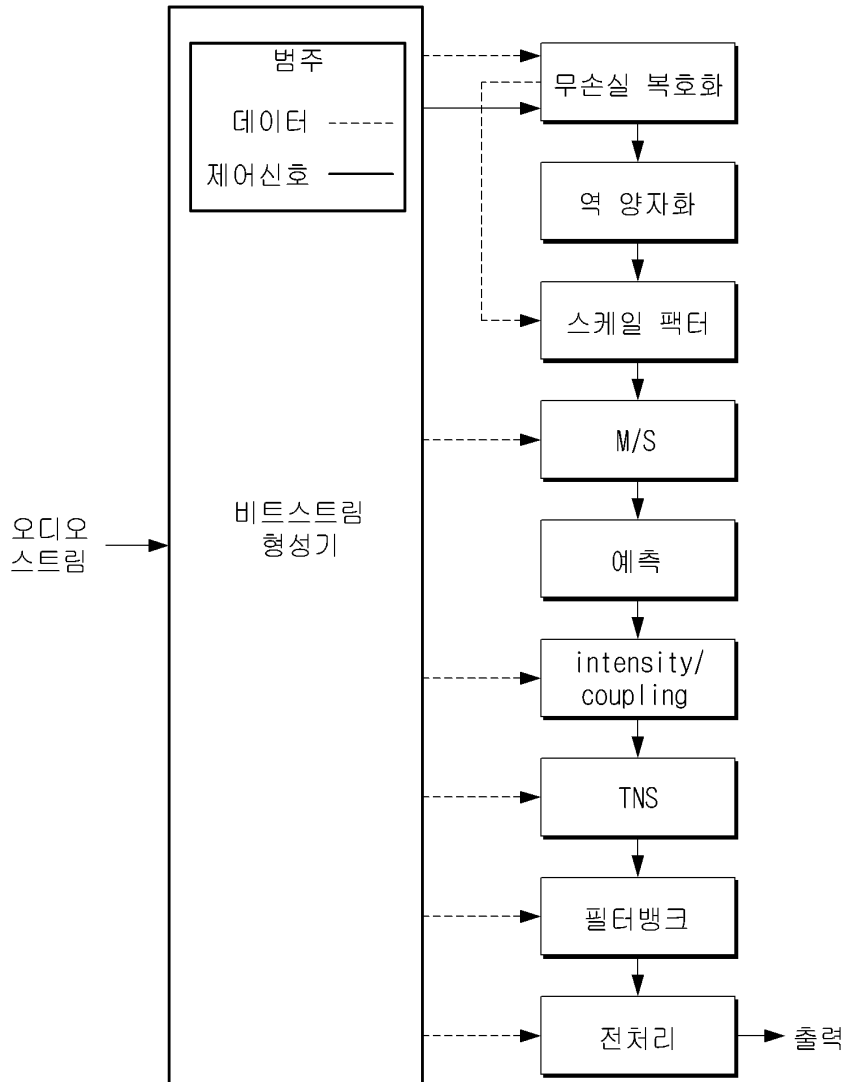
- <1> 도 1은 종래 기술에 따른 MPEG-2 AAC 오디오 복호화 알고리즘의 과정을 나타낸 블록도이다.
- <2> 도 2는 ISO/IEC에서 명시하는 MPEG-2 AAC LC 프로파일의 연산량을 나타낸 그래프이다.
- <3> 도 3은 일반적인 IMDCT 연산 과정을 단계별로 나타낸 블록도이다.
- <4> 도 4는 본 발명에 따른 프로그래머블 프로세서의 구성을 설명하기 위한 구성도이다.
- <5> 도 5는 본 발명에 따른 역주소 생성 과정을 설명하기 위한 구성도이다.
- <6> 도 6은 본 발명에 따른 주소 생성기의 구조를 설명하기 위한 구성도이다.
- <7> 도 7은 본 발명에 따른 역주소 연산 유닛의 구성을 설명하기 위한 구성도이다.
- <8> 도 8은 본 발명에 따른 역주소 연산 유닛의 제어신호 생성기의 구성을 설명하기 위한 구성도이다.
- <9> 도 9는 본 발명에 따른 산술 논리 연산기 내부에서 비트 추출 과정을 설명하기 위한 구성도이다.

<10> *도면의 주요부분에 대한 부호의 설명*

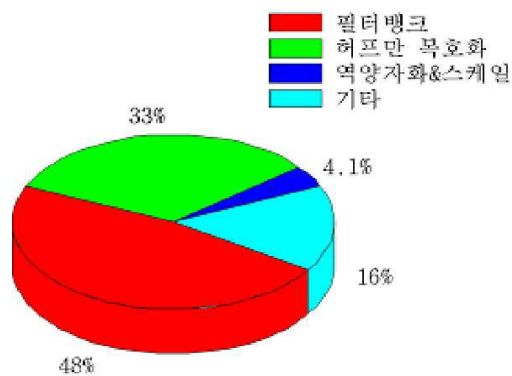
- <11> 110 : 프로그램 제어장치 120 : 주소생성기
- <12> 130 : 역주소 연산 유닛 140 : 데이터 처리 장치
- <13> 150 : 프로그램 메모리 160, 170 : 데이터 메모리
- <14> 180, 190 : 데이터 롬 201 : 제어신호 생성기
- <15> 202 ~ 215 : 인버터 216 ~ 229 : 2-입력 멀티플렉서
- <16> 301 : 8-입력 OR 게이트 302 ~ 308 : 2-입력 OR 게이트
- <17> 400 : 데이터 버스 스위치 401, 402 : 곱셈기
- <18> 403, 404 : 스몰 쉬프터(Small shifter)
- <19> 405, 406 : 덧셈기 407 : 데이터 처리 유닛 408 : 누적기
409 : 산술연산 처리장치 410 : 절대값/최소값/최대값(ABS/MIN/MAX) 연산기
- <20> 411 : 레지스터 파일

도면

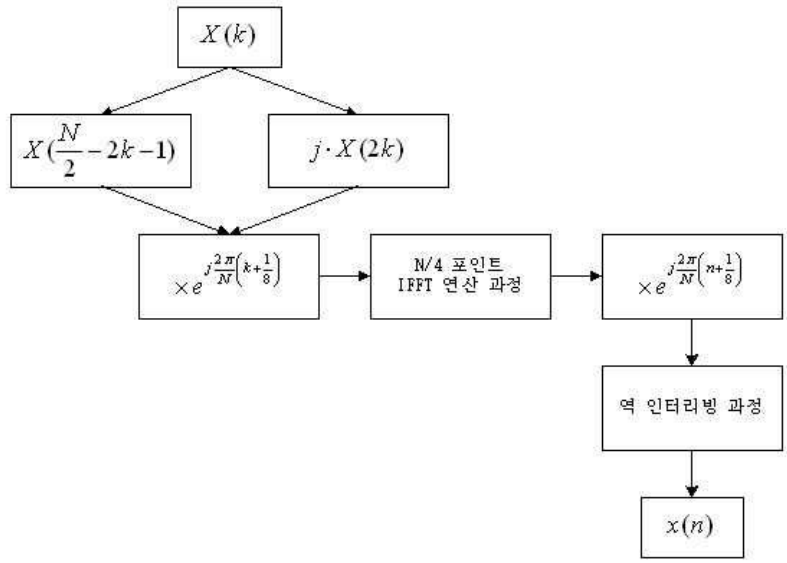
도면1



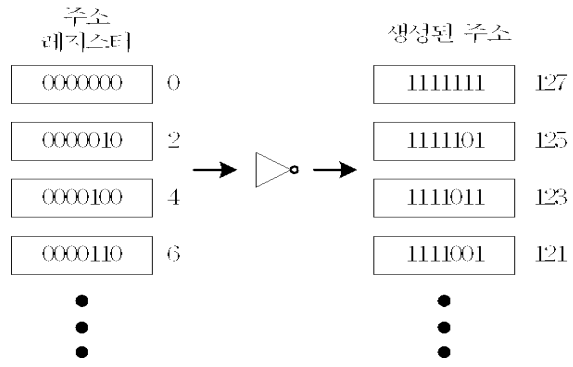
도면2



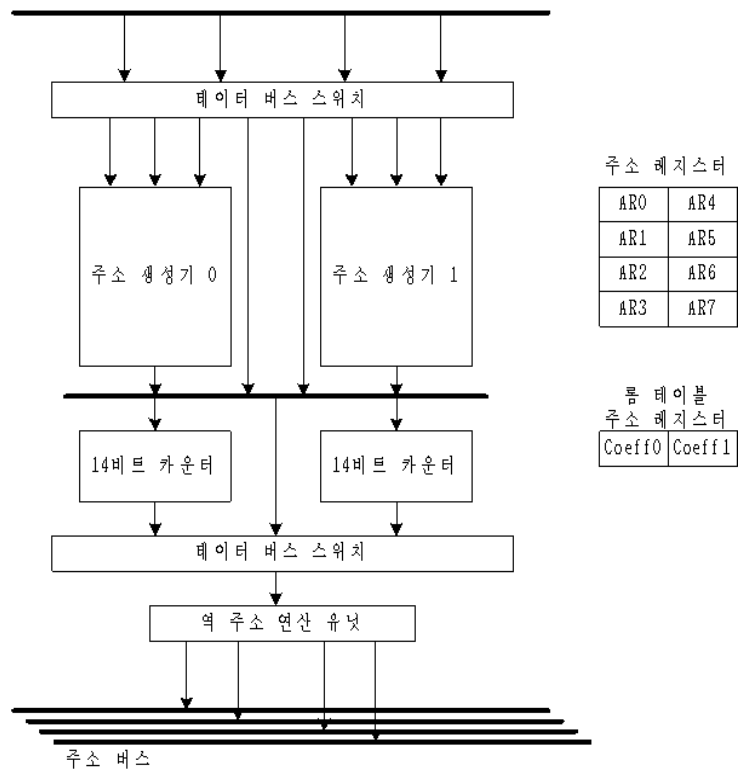
도면3



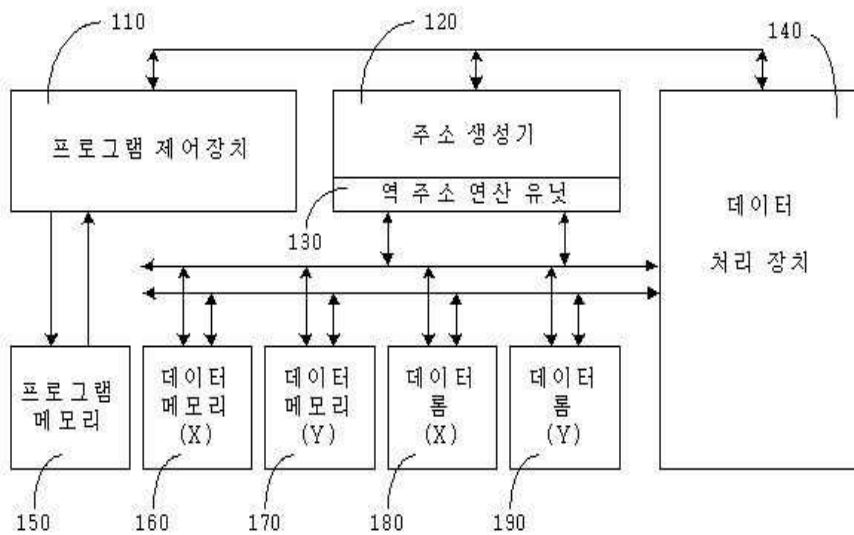
도면4



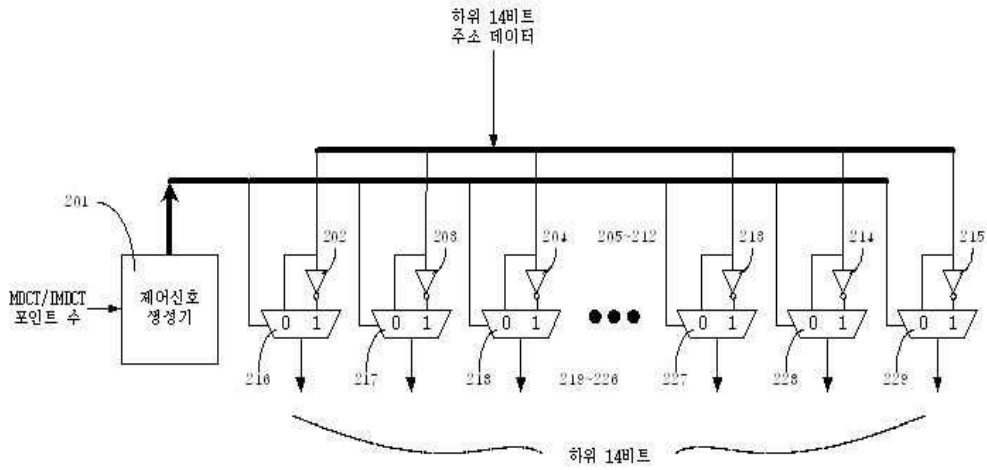
도면5



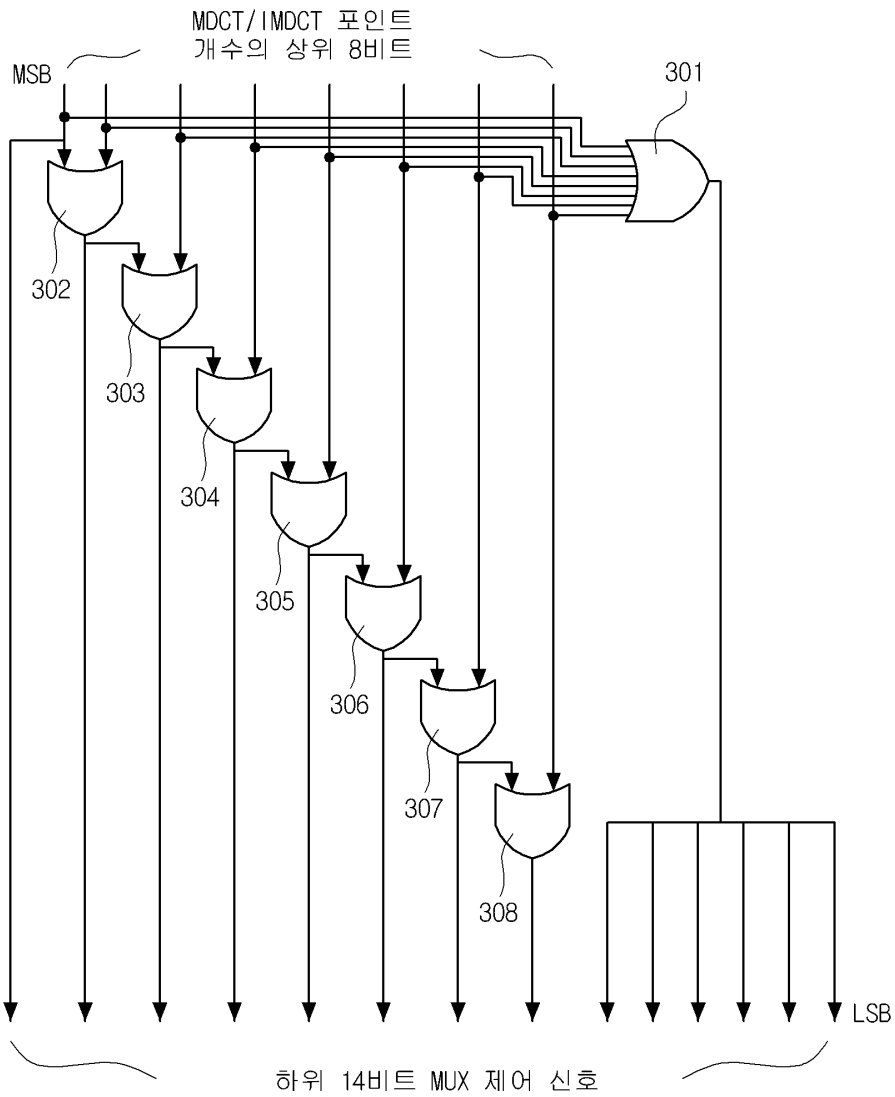
도면6



도면7



도면8



도면9

