



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2014년03월07일
(11) 등록번호 10-1369745
(24) 등록일자 2014년02월26일

(51) 국제특허분류(Int. Cl.)
H04N 21/236 (2011.01) H04N 21/434 (2011.01)
(21) 출원번호 10-2007-0035729
(22) 출원일자 2007년04월11일
심사청구일자 2012년04월10일
(65) 공개번호 10-2008-0092183
(43) 공개일자 2008년10월15일
(56) 선행기술조사문헌
US20030110286 A1
KR1019990068140 A

(73) 특허권자
삼성전자주식회사
경기도 수원시 영통구 삼성로 129 (매탄동)
(72) 발명자
이강은
강원도 강릉시 임영로179번길 5-1 (용강동)
주기현
서울 광진구 독성로41길 20, 105동 1202호 (자양동, 우성1차아파트)
(74) 대리인
리앤목특허법인

전체 청구항 수 : 총 24 항

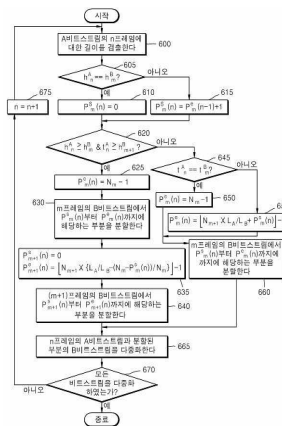
심사관 : 이성현

(54) 발명의 명칭 비동기화된 비트스트림들의 다중화 및 역다중화 방법 및장치

(57) 요약

본 발명은 비동기화 시간 정렬 방식으로 프레임의 길이가 서로 다르게 부호화된 비트스트림들을 다중화하는 방법 및 장치에 관한 것으로서, 서로 다른 프레임의 길이로 부호화하는 복수의 부호화기들에서 부호화된 비트스트림들 가운데 기준 비트스트림의 각 프레임에 대한 길이를 기준으로 나머지 비트스트림들을 분할하여 다중화한다.

대표도 - 도6



(72) 발명자

김중희

서울특별시 강서구 곰달래로 186, 4층 (화곡동)

오은미

경기도 성남시 분당구 중앙공원로 54, 시범단지
223-502 (서현동, 우성아파트)

특허청구의 범위

청구항 1

서로 다른 프레임의 길이로 부호화하는 복수의 부호화기들에서 부호화된 비트스트림들 가운데 기준이 되는 비트스트림으로 설정된 기준 비트스트림의 각 프레임에 대한 길이를 검출하는 단계;

상기 검출된 각 프레임에 대한 길이를 기준으로 하여 상기 기준 비트스트림을 제외한 나머지 비트스트림들을 분할하는 단계; 및

상기 각 프레임의 기준 비트스트림들과 상기 분할된 비트스트림들을 다중화하는 단계를 포함하는 것을 특징으로 하는 비동기화된 비트스트림들의 다중화 방법.

청구항 2

제1항에 있어서, 상기 분할하는 단계는

상기 검출된 각 프레임에 대한 길이를 이용하여 상기 나머지 비트스트림들에서 분할할 지점을 계산하는 단계; 및

상기 계산된 지점에서 상기 나머지 비트스트림들을 분할하는 단계를 포함하는 것을 특징으로 하는 비동기화된 비트스트림들의 다중화 방법.

청구항 3

제1항에 있어서, 상기 다중화하는 단계는

상기 기준 비트스트림의 각 프레임에 대응하는 상기 분할된 비트스트림을 상기 각 프레임의 기준 비트스트림의 전단 또는 후단에 마련되도록 상기 비트스트림들을 정렬하는 단계; 및

상기 정렬된 비트스트림들을 다중화하는 단계를 포함하는 것을 특징으로 하는 비동기화된 비트스트림들의 다중화 방법.

청구항 4

제1항에 있어서, 상기 다중화하는 단계는

상기 분할된 비트스트림들의 데이터의 크기와 관련된 정보를 포함하지 않고 다중화하는 것을 특징으로 하는 비동기화된 비트스트림들의 다중화 방법.

청구항 5

제1항에 있어서, 상기 기준 비트스트림은

비트 할당량이 많은 비트스트림이 설정되는 것을 특징으로 하는 비동기화된 비트스트림들의 다중화 방법.

청구항 6

제1항에 있어서, 상기 기준 비트스트림은

프레임의 길이가 짧은 비트스트림이 설정되는 것을 특징으로 하는 비동기화된 비트스트림들의 다중화 방법.

청구항 7

서로 다른 프레임의 길이로 부호화하는 복수의 부호화기들에서 부호화된 비트스트림들이 부호화단에서 다중화된 스트림을 역다중화하는 단계;

상기 역다중화된 비트스트림들 가운데 기준이 되는 비트스트림으로 설정된 기준 비트스트림의 각 프레임에 대한 길이를 검출하는 단계; 및

상기 검출된 각 프레임에 대한 길이를 이용하여 상기 기준 비트스트림을 제외한 나머지 분할된 비트스트림들을 추출하는 단계를 포함하는 것을 특징으로 하는 비동기화된 비트스트림들의 역다중화 방법.

청구항 8

제7항에 있어서, 상기 추출하는 단계는

상기 검출된 각 프레임에 대한 길이를 이용하여 상기 나머지 분할된 비트스트림들의 데이터 크기를 계산하는 단계; 및

상기 계산된 데이터 크기에 해당하는 나머지 분할된 비트스트림들을 추출하는 단계를 포함하는 것을 특징으로 하는 비동기화된 비트스트림들의 역다중화 방법.

청구항 9

제7항에 있어서, 상기 스트림은

상기 나머지 비트스트림들의 데이터 크기와 관련된 정보를 포함하지 않고 다중화하는 것을 특징으로 하는 비동기화된 비트스트림들의 역다중화 방법.

청구항 10

제7항에 있어서, 상기 기준 비트스트림은

비트 할당량이 많은 비트스트림이 설정되는 것을 특징으로 하는 비동기화된 비트스트림들의 역다중화 방법.

청구항 11

제7항에 있어서, 상기 기준 비트스트림은

프레임의 길이가 짧은 비트스트림이 설정되는 것을 특징으로 하는 비동기화된 비트스트림들의 역다중화 방법.

청구항 12

서로 다른 프레임의 길이로 부호화하는 복수의 부호화기들에서 부호화된 비트스트림들 가운데 기준이 되는 비트스트림으로 설정된 기준 비트스트림의 각 프레임에 대한 길이를 검출하는 단계;

상기 검출된 각 프레임에 대한 길이를 기준으로 하여 상기 기준 비트스트림을 제외한 나머지 비트스트림들을 분할하는 단계; 및

상기 각 프레임의 기준 비트스트림들과 상기 분할된 비트스트림들을 다중화하는 단계를 컴퓨터에서 실행시키기 위한 프로그램을 기록한 컴퓨터로 읽을 수 있는 기록 매체.

청구항 13

서로 다른 프레임의 길이로 부호화하는 복수의 부호화기들에서 부호화된 비트스트림들이 부호화단에서 다중화된 스트림을 역다중화하는 단계;

상기 역다중화된 비트스트림들 가운데 기준이 되는 비트스트림으로 설정된 기준 비트스트림의 각 프레임에 대한 길이를 검출하는 단계; 및

상기 검출된 각 프레임에 대한 길이를 이용하여 상기 기준 비트스트림을 제외한 나머지 분할된 비트스트림들을 추출하는 단계를 컴퓨터에서 실행시키기 위한 프로그램을 기록한 컴퓨터로 읽을 수 있는 기록 매체.

청구항 14

서로 다른 프레임의 길이로 부호화하는 복수의 부호화기들에서 부호화된 비트스트림들 가운데 기준이 되는 비트스트림으로 설정된 기준 비트스트림의 각 프레임에 대한 길이를 검출하는 프레임길이 검출부;

상기 검출된 각 프레임에 대한 길이를 기준으로 하여 상기 기준 비트스트림을 제외한 나머지 비트스트림들을 분할하는 분할부; 및

상기 각 프레임의 기준 비트스트림들과 상기 분할된 비트스트림들을 다중화하는 다중화부를 포함하는 것을 특징으로 하는 비동기화된 비트스트림들의 다중화 장치.

청구항 15

제14항에 있어서, 상기 분할부는

상기 검출된 각 프레임에 대한 길이를 이용하여 상기 나머지 비트스트림들에서 분할할 지점을 계산하는 분할지점 계산부; 및

상기 계산된 지점에서 상기 나머지 비트스트림들을 분할하는 비트스트림 분할부를 포함하는 것을 특징으로 하는 비동기화된 비트스트림들의 다중화 장치.

청구항 16

제14항에 있어서, 상기 다중화부는

상기 기준 비트스트림의 각 프레임에 대응하는 상기 분할된 비트스트림을 상기 각 프레임의 기준 비트스트림의 전단 또는 후단에 마련되도록 상기 비트스트림들을 정렬하는 비트스트림 정렬부; 및

상기 정렬된 비트스트림들을 다중화하는 비트스트림 다중화부를 포함하는 것을 특징으로 하는 비동기화된 비트스트림들의 다중화 장치.

청구항 17

제14항에 있어서, 상기 다중화부는

상기 분할된 비트스트림들의 데이터의 크기와 관련된 정보를 포함하지 않고 다중화하는 것을 특징으로 하는 비동기화된 비트스트림들의 다중화 장치.

청구항 18

제14항에 있어서, 상기 기준 비트스트림은

비트 할당량이 많은 비트스트림이 설정되는 것을 특징으로 하는 비동기화된 비트스트림들의 다중화 장치.

청구항 19

제14항에 있어서, 상기 기준 비트스트림은

프레임의 길이가 짧은 비트스트림이 설정되는 것을 특징으로 하는 비동기화된 비트스트림들의 다중화 장치.

청구항 20

서로 다른 프레임의 길이로 부호화하는 복수의 부호화기들에서 부호화된 비트스트림들이 부호화단에서 다중화된 스트림을 역다중화하는 역다중화부;

상기 역다중화된 비트스트림들 가운데 기준이 되는 비트스트림으로 설정된 기준 비트스트림의 각 프레임에 대한 길이를 검출하는 길이 검출부; 및

상기 검출된 각 프레임에 대한 길이를 이용하여 상기 기준 비트스트림을 제외한 나머지 분할된 비트스트림들을 추출하는 추출부를 포함하는 것을 특징으로 하는 비동기화된 비트스트림들의 역다중화 장치.

청구항 21

제20항에 있어서, 상기 추출부는

상기 검출된 각 프레임에 대한 길이를 이용하여 상기 나머지 분할된 비트스트림들의 데이터 크기를 계산하는 데이터크기 계산부; 및

상기 계산된 데이터 크기에 해당하는 나머지 분할된 비트스트림들을 추출하는 비트스트림 추출부를 포함하는 것을 특징으로 하는 비동기화된 비트스트림들의 역다중화 장치.

청구항 22

제20항에 있어서, 상기 스트림은

상기 나머지 비트스트림들의 데이터 크기와 관련된 정보를 포함하지 않고 다중화하는 것을 특징으로 하는 비동

기화된 비트스트림들의 역다중화 장치.

청구항 23

제20항에 있어서, 상기 기준 비트스트림은

비트 할당량이 많은 비트스트림이 설정되는 것을 특징으로 하는 비동기화된 비트스트림들의 역다중화 장치.

청구항 24

제20항에 있어서, 상기 기준 비트스트림은

프레임의 길이가 짧은 비트스트림이 설정되는 것을 특징으로 하는 비동기화된 비트스트림들의 역다중화 장치.

명세서

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

- [0019] 본 발명은 복수의 비트스트림들을 다중화하는 방법 및 장치에 관한 것으로, 보다 상세하게는 비동기화 시간 정렬 방식으로 프레임의 길이가 서로 다르게 부호화된 비트스트림들을 다중화하는 방법 및 장치에 관한 것이다.
- [0020] 도 1에 도시된 바와 같이 각각 다른 방식에 의하여 부호화하는 부호화기에 해당하는 A 부호화기(100)와 B 부호화기(110)에서 부호화된 비트스트림들을 다중화부(120)에서 다중화하여 스트림을 출력할 수 있다. 예를 들어, A 부호화기(100)는 AAC(Advanced Audio Coding)에 의하여 오디오 신호를 부호화하고, B 부호화기(110)는 MPEG(Moving Picture Experts Group) Surround에 의해 멀티채널에 해당하는 오디오 신호를 부호화함으로써 각각 다른 방식에 의해 부호화된 비트스트림들을 다중화부(120)에서 다중화할 수 있다.
- [0021] 이와 같이 서로 다른 방식에 의하여 부호화할 경우 도 2에 도시한 바와 같이 A 부호화기(100)의 각 프레임에 대한 길이가 B 부호화기(110)의 각 프레임에 대한 길이와 일치하는 동기 시간 정렬(synchronous time align) 방식이 있으며, 도 3에 도시한 바와 같이 A 부호화기(100)의 각 프레임에 대한 길이가 B 부호화기(110)의 각 프레임에 대한 길이와 불일치하는 비동기 시간 정렬(asynchronous time align) 방식이 있다.
- [0022] 이 가운데 비동기 시간 정렬 방식에 의해 다중화부(120)에서 비트스트림들을 다중화함에 있어서, 종래에는 A 부호화기(100) 및 B 부호화기(110) 가운데 어느 하나의 부호화기를 선택하여 선택된 부호화기에서 부호화된 비트스트림은 프레임을 기준으로 분할되고, 선택되지 않은 부호화기에서 부호화된 비트스트림은 기 설정된 기준에 따라 분할된 후 다중화된다. 여기서, 선택되지 않은 부호화기에서 부호화된 비트스트림이 분할된 데이터의 크기와 관련된 정보를 복호화단에서 인지할 수 없으므로 다중화부(120)는 선택되지 않은 부호화기에서 부호화되어 각 분할된 비트스트림의 길이 정보를 삽입한다. 예를 들어, 도 3에서와 같이 서로 다른 프레임의 길이로 부호화된 A 부호화기(100)와 B 부호화기(110)에서 부호화된 비트스트림들을 다중화부(120)에서 다중화하여 스트림을 생성할 경우, A 부호화기(100)에서 부호화된 (m-1) 프레임의 비트스트림(300)을 제1 (m-1) 프레임의 비트스트림(303)과 제2 (m-1) 프레임의 비트스트림(306)으로 분할하고, 분할된 제1 (m-1) 프레임의 비트스트림(303)과 제2 (m-1) 프레임의 비트스트림(306)에 대한 길이를 복호화단에서 인식할 수 있도록 제1 (m-1) 프레임의 비트스트림(303)에 대한 길이 정보인 제(n-2) 길이 정보(330)와 제2 (m-1) 프레임의 비트스트림(306)에 대한 길이 정보인 제(n-1) 길이 정보(335)를 삽입한다.
- [0023] 그러나 이와 같이 부호화단에서 분할된 각 비트스트림의 길이 정보들을 삽입하여 다중화하면, 분할된 비트스트림의 개수만큼 분할된 각 비트스트림의 길이 정보를 생성하여 스트림에 포함시켜 복호화단으로 전송해야 하므로 부호화하는 과정이 더 복잡해질 뿐 만 아니라 부호화를 수행하는 시간이 더 소요되며, 부호화단에서 복호화단으로 전송해야 할 데이터의 크기가 증가하는 문제점을 갖는다.
- [0024] 또한, 도 5에 도시된 바와 같이 B 부호화기(110)의 프레임 길이에 대한 A 부호화기(100)의 프레임 길이가 차이가 현저히 날 경우, A 부호화기(100)에서 부호화된 단위 프레임의 비트스트림이 상당히 많은 개수로 분할된다. 이에 따라 삽입해야 할 분할된 비트스트림들에 대한 길이 정보의 개수와 데이터의 크기가 과도하게 증가하는 문제점이 발생한다.

발명이 이루고자 하는 기술적 과제

- [0025] 본 발명이 이루고자 하는 기술적 과제는, 서로 다른 프레임의 길이로 부호화하는 복수의 부호화기들에서 부호화된 비트스트림들 가운데 기준 비트스트림의 각 프레임에 대한 길이를 기준으로 나머지 비트스트림들을 분할하여 다중화하는 방법 및 장치를 제공하는 것이다.
- [0026] 본 발명이 이루고자 하는 다른 기술적 과제는, 서로 다른 프레임의 길이로 부호화하는 복수의 부호화기들에서 부호화된 비트스트림들이 부호화단에서 다중화된 스트림을 역다중화하여 기준 비트스트림의 각 프레임에 대한 길이를 검출하고 각 프레임에 대한 길이를 이용하여 나머지 비트스트림들을 추출하는 방법 및 장치를 제공하는 것이다.

발명의 구성 및 작용

- [0027] 상기의 과제를 이루기 위한 본 발명에 의한 비동기화된 비트스트림들의 다중화 방법은, 서로 다른 프레임의 길이로 부호화하는 복수의 부호화기들에서 부호화된 비트스트림들 가운데 기준이 되는 비트스트림으로 설정된 기준 비트스트림의 각 프레임에 대한 길이를 검출하는 단계, 상기 검출된 각 프레임에 대한 길이를 기준으로 하여 상기 기준 비트스트림을 제외한 나머지 비트스트림들을 분할하는 단계 및 상기 각 프레임의 기준 비트스트림들과 상기 분할된 비트스트림들을 다중화하는 단계를 포함하는 것을 특징으로 한다.
- [0028] 상기의 과제를 이루기 위한 본 발명에 의한 비동기화된 비트스트림들의 역다중화 방법은, 서로 다른 프레임의 길이로 부호화하는 복수의 부호화기들에서 부호화된 비트스트림들이 부호화단에서 다중화된 스트림을 역다중화하는 단계, 상기 역다중화된 비트스트림들 가운데 기준이 되는 비트스트림으로 설정된 기준 비트스트림의 각 프레임에 대한 길이를 검출하는 단계 및 상기 검출된 각 프레임에 대한 길이를 이용하여 상기 기준 비트스트림을 제외한 나머지 분할된 비트스트림들을 추출하는 단계를 포함하는 것을 특징으로 한다.
- [0029] 상기의 과제를 이루기 위한 본 발명에 의한 기록 매체는, 서로 다른 프레임의 길이로 부호화하는 복수의 부호화기들에서 부호화된 비트스트림들 가운데 기준이 되는 비트스트림으로 설정된 기준 비트스트림의 각 프레임에 대한 길이를 검출하는 단계, 상기 검출된 각 프레임에 대한 길이를 기준으로 하여 상기 기준 비트스트림을 제외한 나머지 비트스트림들을 분할하는 단계 및 상기 각 프레임의 기준 비트스트림들과 상기 분할된 비트스트림들을 다중화하는 단계를 컴퓨터에서 실행시키기 위한 프로그램을 기록한 컴퓨터로 읽을 수 있다.
- [0030] 상기의 과제를 이루기 위한 본 발명에 의한 기록 매체는, 서로 다른 프레임의 길이로 부호화하는 복수의 부호화기들에서 부호화된 비트스트림들이 부호화단에서 다중화된 스트림을 역다중화하는 단계, 상기 역다중화된 비트스트림들 가운데 기준이 되는 비트스트림으로 설정된 기준 비트스트림의 각 프레임에 대한 길이를 검출하는 단계 및 상기 검출된 각 프레임에 대한 길이를 이용하여 상기 기준 비트스트림을 제외한 나머지 분할된 비트스트림들을 추출하는 단계를 컴퓨터에서 실행시키기 위한 프로그램을 기록한 컴퓨터로 읽을 수 있다.
- [0031] 상기의 과제를 이루기 위한 본 발명에 의한 비동기화된 비트스트림들의 다중화 장치는, 서로 다른 프레임의 길이로 부호화하는 복수의 부호화기들에서 부호화된 비트스트림들 가운데 기준이 되는 비트스트림으로 설정된 기준 비트스트림의 각 프레임에 대한 길이를 검출하는 프레임길이 검출부, 상기 검출된 각 프레임에 대한 길이를 기준으로 하여 상기 기준 비트스트림을 제외한 나머지 비트스트림들을 분할하는 분할부 및 상기 각 프레임의 기준 비트스트림들과 상기 분할된 비트스트림들을 다중화하는 다중화부를 포함하는 것을 특징으로 한다.
- [0032] 상기의 과제를 이루기 위한 본 발명에 의한 비동기화된 비트스트림들의 역다중화 장치는, 서로 다른 프레임의 길이로 부호화하는 복수의 부호화기들에서 부호화된 비트스트림들이 부호화단에서 다중화된 스트림을 역다중화하는 역다중화부, 상기 역다중화된 비트스트림들 가운데 기준이 되는 비트스트림으로 설정된 기준 비트스트림의 각 프레임에 대한 길이를 검출하는 길이 검출부 및 상기 검출된 각 프레임에 대한 길이를 이용하여 상기 기준 비트스트림을 제외한 나머지 분할된 비트스트림들을 추출하는 추출부를 포함하는 것을 특징으로 한다.
- [0033] 이하, 첨부된 도면들을 참조하여 본 발명에 따른 비동기화된 비트스트림들의 다중화 및 역다중화 방법 및 장치에 대해 상세히 설명한다.
- [0034] 도 6은 본 발명에 의한 비동기화된 비트스트림들의 다중화 방법에 대한 일 실시예를 흐름도로 도시한 것이다. 본 실시예에서는 서로 프레임의 길이를 다르게 하여 부호화하는 복수의 부호화기들이 아닌 서로 프레임의 길이를 다르게 하여 부호화하는 A 부호화기와 B 부호화기만을 기재함으로써 기준 비트스트림과 단수의 비트스트림만을 다중화하는 예로서 서술하였으나 이는 단순한 예에 지나지 않으며 이에 한정되지는 않는다. 그러므로 본 발

명이 속하는 기술분야에서 통상의 지식을 가진 자라면 본 실시예에 의하여 기준 비트스트림과 복수의 비트스트림들도 다중화하도록 실시할 수 있다.

- [0035] 먼저, A 부호화기에서 부호화된 A 비트스트림 및 B 부호화기에서 부호화된 B 비트스트림 가운데 기준이 되는 비트스트림으로 설정된 기준 비트스트림에 해당하는 A 비트스트림의 n 프레임에 대한 길이를 검출한다(제600단계). 여기서, 기준 비트스트림은 비트 할당량이 많은 비트스트림 또는 프레임의 길이가 짧은 비트스트림으로 설정하는 것이 바람직하다.
- [0036] 제600단계 후에, A 비트스트림에서 n 프레임의 시작점인 h^A_n 과 B 비트스트림에서 m 프레임의 시작점인 h^B_m 가 일치하는지 여부를 판단한다(제605단계).
- [0037] 만일 제605단계에서 h^A_n 과 h^B_m 가 일치한다고 판단되면, n 프레임의 A 비트스트림과 다중화할 m 프레임의 B 비트스트림에서 분할할 시작점을 나타내는 포인터인 $P^S_m(n)$ 에 '0' 값을 할당한다(제610단계).
- [0038] 만일 제605단계에서 h^A_n 과 h^B_m 가 일치하지 않는다고 판단되면, A모듈의 n-1번째 프레임의 비트스트림과 다중화된 B모듈의 m번째 프레임의 비트스트림에서 분할된 종료점을 나타내는 포인터인 $P^S_m(n-1)$ 의 다음 값을 $P^S_m(n)$ 에 할당한다(제615단계). 다시 말하면, 제615단계에서는 $P^S_m(n)$ 에 ' $P^S_m(n-1)+1$ ' 값을 할당한다.
- [0039] 제610단계 또는 제615단계 후에, h^A_n 가 h^B_m 보다 크거나 같고 A 비트스트림에서 n 프레임의 종료점인 t^A_n 가 B 비트스트림에서 (m+1) 프레임의 시작점인 h^B_{m+1} 보다 큰지 여부를 판단한다(제620단계).
- [0040] 만일 제620단계에서 h^A_n 가 h^B_m 보다 크거나 같고 t^A_n 가 h^B_{m+1} 보다 크다고 판단되면, n 프레임의 A 비트스트림과 다중화할 m 프레임의 B 비트스트림에서 분할한 종료점을 나타내는 포인터인 $P^S_m(n)$ 에 B모듈의 m번째 프레임의 비트스트림의 길이인 N_m 에서 1을 감산한 값을 할당한다(제625단계).
- [0041] 제625단계 후에, m 프레임의 B 비트스트림에서 $P^S_m(n)$ 부터 $P^E_m(n)$ 까지에 해당하는 부분을 분할한다(제630단계).
- [0042] 제630단계 후에, n 프레임의 A 비트스트림과 다중화할 (m+1) 프레임의 B 비트스트림에서 분할할 시작점을 나타내는 포인터인 $P^S_{m+1}(n)$ 에 값을 할당하고, n 프레임의 A 비트스트림과 다중화할 (m+1) 프레임의 B 비트스트림에서 분할할 종료점을 나타내는 포인터인 $P^E_{m+1}(n)$ 에 $\lfloor \lfloor N_{m+1} + \{L^A_n \cdot L^B_{m+1} - (N_m - P^S_m(n)) \cdot N_m\} \rfloor - 1 \rfloor$ 값을 할당한다(제635단계). 여기서, L^A_n 는 A 부호화기의 n 번째 프레임의 길이이고, L^B_{m+1} 는 B 부호화기의 (m+1) 번째 프레임의 길이이다.
- [0043] 제635단계 후에, (m+1) 프레임의 B 비트스트림에서 $P^S_{m+1}(n)$ 부터 $P^E_{m+1}(n)$ 까지에 해당하는 부분을 분할한다(제640단계).
- [0044] 제640단계 후에, n 프레임의 A 비트스트림과 제630단계 및 제640단계에서 분할된 B 비트스트림의 부분들을 다중화한다(제665단계). 제665단계에서 다중화함에 있어서, n 프레임의 A 비트스트림의 후단에 제630단계 및 제640단계에서 분할된 B 비트스트림이 마련되도록 정렬하여 다중화하는 것이 바람직하다. 그러나 n 프레임의 A 비트스트림의 전단에 제630단계 및 제640단계에서 분할된 B 비트스트림이 마련될 수도 있다.
- [0045] 만일 제620단계에서 h^A_n 가 h^B_m 보다 작거나 t^A_n 가 h^B_{m+1} 보다 작거나 같다고 판단되면, t^A_n 과 B 비트스트림에서 m 프레임의 종료점인 t^B_m 가 일치하는지 여부를 판단한다(제645단계).

- [0046] 제645단계에서 t^A_n 과 t^B_m 가 일치한다고 판단되면, $P^e_m(n)$ 에 N_m 에서 1은 감소한 값을 할당한다(제650단계).
- [0047] 제645단계에서 t^A_n 과 t^B_m 가 일치하지 않는다고 판단되면, $P^e_m(n)$ 에 $\lfloor \lfloor N_{m+1} + L^A_n \cdot L^B_{m+1} + P^s_m(n) \rfloor - 1 \rfloor$ 값을 할당한다(제655단계).
- [0048] 제655단계 후에, m 프레임의 B 비트스트림에서 $P^s_m(n)$ 부터 $P^e_m(n)$ 까지에 해당하는 부분을 분할한다(제660단계).
- [0049] 제660단계 후에, n 프레임의 A 비트스트림과 제660단계에서 분할된 B 비트스트림을 다중화한다(제665단계). 제665단계에서 다중화함에 있어서, n 프레임의 A 비트스트림의 후단에 제660단계에서 분할된 B 비트스트림이 마련 되도록 정렬하여 다중화하는 것이 바람직하다. 그러나 n 프레임의 A 비트스트림의 전단에 제660단계에서 분할된 B 비트스트림이 마련될 수도 있다.
- [0050] 제665단계 후에, 모든 비트스트림을 다중화하였는지 여부를 판단한다(제670단계).
- [0051] 제670단계에서 모든 비트스트림을 다중화하지 않았다고 판단되면, 'n'에 '1'을 가산하고(제675단계), 제600단계를 수행한다.
- [0052] 도 7은 본 발명에 의한 비동기화된 비트스트림들의 다중화 장치의 일 실시예를 블록도로 도시한 것으로서, 비동기화된 비트스트림들의 다중화 장치는 프레임길이 검출부(700), 분할지점 계산부(710), 비트스트림 분할부(720) 및 다중화부(730)를 포함하여 이루어진다. 본 실시예에서는 서로 프레임의 길이를 다르게 하여 부호화하는 복수의 부호화기들이 아닌 서로 프레임의 길이를 다르게 하여 부호화하는 A 부호화기와 B 부호화기만을 기재함으로써 기준 비트스트림과 단수의 비트스트림만을 다중화하는 예로서 서술하였으나 이는 단순한 예에 지나지 않으며 이에 한정되지는 않는다. 그러므로 본 발명이 속하는 기술분야에서 통상의 지식을 가진 자라면 본 실시예에 의하여 기준 비트스트림과 복수의 비트스트림들도 다중화하도록 실시할 수 있다.
- [0053] 프레임길이 검출부(700)는 A 부호화기에서 부호화된 A 비트스트림 및 B 부호화기에서 부호화된 B 비트스트림 가운데 기준이 되는 비트스트림으로 설정된 기준 비트스트림에 해당하는 A 부호화기의 n번째 프레임에 대한 길이를 검출한다. 여기서, 기준 비트스트림은 비트 할당량이 많은 비트스트림 또는 프레임의 길이가 짧은 비트스트림으로 설정하는 것이 바람직하다.
- [0054] 분할지점 계산부(710)는 프레임길이 검출부(700)에서 검출된 A 부호화기의 n 번째 프레임에 대한 길이를 이용하여 n 번째 프레임의 A 비트스트림과 다중화할 B 부호화기의 m 번째 프레임 또는 (m+1) 번째 프레임의 비트스트림에서 분할할 지점을 계산한다. 분할지점 계산부(710)에서 분할할 지점을 계산함에 있어서, A 비트스트림의 n 프레임과 B 비트스트림의 m 프레임 또는 (m+1) 프레임에 대한 A 비트스트림의 각 경우에 따라 이하에서 기술한 바와 같이 다르게 계산한다.
- [0055] 첫째, 도 10에 도시된 바와 같이 A 비트스트림에서 n 프레임의 시작점인 h^A_n 과 B 비트스트림에서 m 프레임의 시작점인 h^B_m 가 일치하고 t^A_n 가 h^B_m 와 t^B_m 사이에 위치하는 경우 수학적 1을 적용하여 m 프레임의 B 비트스트림에서 분할할 지점을 계산한다.
- [0056] [수학적 1]
- $$P^s_m(n) = 0$$
- $$P^e_m(n) = \lfloor \lfloor N_m \cdot L^A_n \cdot L^B_m \rfloor - 1 \rfloor$$
- [0057]
- [0058] 여기서, $P^s_m(n)$ 는 n 프레임의 A 비트스트림과 다중화할 m 프레임의 B 비트스트림에서 분할할 시작점을 나타내는 포인터이고, $P^e_m(n)$ 는 n 프레임의 A 비트스트림과 다중화할 m 프레임의 B 비트스트림에서 분할한 종료점을 나타내는 포인터이며, N_m 는 m 프레임의 B 비트스트림에 해당하는 길이이고, L^A_n 는 A 부호화기의 n 프레임에

대한 길이이며, L^B_m 는 B 부호화기의 m 프레임에 대한 길이이다.

[0059] 둘째, 도 11에 도시된 바와 같이 A 비트스트림에서 n 프레임의 종료점인 t^A_n 과 B 비트스트림에서 m 프레임의 종료점인 t^B_m 가 일치하고 h^A_n 가 h^B_m 와 t^B_m 사이에 위치하는 경우 수학적 2를 적용하여 m 프레임의 B 비트스트림에서 분할할 지점을 계산한다.

[0060] [수학적 2]

$$P^s_m(t) = P^e_{n-1} + 1$$

$$P^e_m(t) = N_m - 1$$

[0061]

[0062] 여기서, $P^s_m(t)$ 는 n 프레임의 A 비트스트림과 다중화할 m 프레임의 B 비트스트림에서 분할할 시작점을 나타내는 포인터이고, $P^e_m(t)$ 는 n 프레임의 A 비트스트림과 다중화할 m 프레임의 B 비트스트림에서 분할한 종료점을 나타내는 포인터이며, $P^s_m(t-1)$ 는 (n-1) 프레임의 A 비트스트림과 다중화할 m 프레임의 B 비트스트림에서 분할한 종료점을 나타내는 포인터이고, N_m 는 m 프레임의 B 비트스트림에 해당하는 길이이다.

[0063] 셋째, 도 12에 도시된 바와 같이 n 프레임의 A 비트스트림이 B 비트스트림의 m 프레임과 (m+1) 프레임 모두에 걸쳐있는 경우, 다시 말하면 h^A_n 가 h^B_m 와 t^B_m 사이에 위치하고 t^A_n 가 h^B_{m+1} 와 t^B_{m+1} 사이에 위치하는 경우, 수학적 3을 적용하여 m 프레임과 (m+1) 프레임의 B 비트스트림에서 분할할 지점을 계산한다.

[0064] [수학적 3]

$$P^s_m(t) = P^e_m(t-1) + 1$$

$$P^e_m(t) = N_m - 1$$

$$P^s_{m+1}(t) = 0$$

$$P^e_{m+1}(t) = \lfloor \frac{N_{m+1} \cdot (L^A_n \cdot L^B_m - (N_m - P^s_m(t)) \cdot N_m)}{N_m} \rfloor - 1$$

[0065]

[0066] 여기서, $P^s_m(t)$ 는 n 프레임의 A 비트스트림과 다중화할 m 프레임의 B 비트스트림에서 분할할 시작점을 나타내는 포인터이고, $P^e_m(t)$ 는 n 프레임의 A 비트스트림과 다중화할 m 프레임의 B 비트스트림에서 분할한 종료점을 나타내는 포인터이며, $P^s_{m+1}(t)$ 는 n 프레임의 A 비트스트림과 다중화할 (m+1) 프레임의 B 비트스트림에서 분할할 시작점을 나타내는 포인터이고, $P^e_{m+1}(t)$ 는 n 프레임의 A 비트스트림과 다중화할 (m+1) 프레임의 B 비트스트림에서 분할한 종료점을 나타내는 포인터이며, $P^s_m(t-1)$ 는 (n-1) 프레임의 A 비트스트림과 다중화할 m 프레임의 B 비트스트림에서 분할한 종료점을 나타내는 포인터이고, N_m 는 m 프레임의 B 비트스트림에 해당하는 길이이며, N_{m+1} 는 (m+1) 프레임의 B 비트스트림에 해당하는 길이이고, L^A_n 는 A 부호화기의 n 프레임에 대한 길이이며, L^B_m 는 B 부호화기의 m 프레임에 대한 길이이다.

[0067] 넷째, 도 13에 도시된 바와 같이 n 프레임의 A 비트스트림이 m 프레임의 B 비트스트림에 포함되는 경우, 다시 말하면 h^A_n 가 h^B_m 보다 뒤에 위치하고 t^A_n 가 t^B_m 보다 앞에 위치하는 경우, 수학적 4를 적용하여 m 프레임의 B 비트스트림에서 분할할 지점을 계산한다.

[0068] [수학식 4]

$$P_m^s(i) = P_m^e(i-1) + 1$$

$$P_m^e(i) = \left\lfloor \frac{N_m}{L_n} \cdot L_m^B + P_m^s(i) \right\rfloor - 1$$

[0069]

[0070] 여기서, $P_m^s(i)$ 는 n 프레임의 A 비트스트림과 다중화할 m 프레임의 B 비트스트림에서 분할할 시작점을 나타내는 포인터이고, $P_m^e(i)$ 는 n 프레임의 A 비트스트림과 다중화할 m 프레임의 B 비트스트림에서 분할한 종료점을 나타내는 포인터이며, $P_m^s(i-1)$ 는 (n-1) 프레임의 A 비트스트림과 다중화할 m 프레임의 B 비트스트림에서 분할한 종료점을 나타내는 포인터이고, N_m 는 m 프레임의 B 비트스트림에 해당하는 길이이며, L_n^A 는 A 부호화기의 n 프레임에 대한 길이이고, L_m^B 는 B 부호화기의 m 프레임에 대한 길이이다.

[0071] 분할지점 계산부(710)에서 전술한 수학식들에 의해 분할하는 지점을 계산하는 실시예를 도 14를 예를 들어 설명하면 다음과 같다.

[0072] 우선, a 지점은 시작점이므로 a에 '0'을 할당한다.

[0073] 다음으로 b 지점의 경우, (n-1) 프레임의 A 비트스트림이 (m-1) 프레임의 B 비트스트림에 포함되므로 분할지점 계산부(710)는 수학식 4를 적용하여 b 지점을 계산한다. 수학식 4에 A 비트스트림의 (n-1) 프레임에 대한 길이인 '640', B 비트스트림의 (m-1) 프레임에 대한 길이인 '2048', (m-1) 프레임의 B 비트스트림에 해당하는 길이

인 N_{m-1} 을 대입하면 b에 ' $\left\lfloor \frac{N_{m-1} \cdot 640}{2048} \right\rfloor - 1$ '가 할당된다.

[0074] 마찬가지로 c 지점의 경우, n 프레임의 A 비트스트림이 (m-1) 프레임의 B 비트스트림에 포함되므로 분할지점 계산부(710)는 수학식 4를 적용하여 c 지점을 계산한다. 수학식 4에 A 비트스트림의 n 프레임에 대한 길이인 '1280', B 비트스트림의 (m-1) 프레임에 대한 길이인 '2048', (m-1) 프레임의 B 비트스트림에 해당하는 길이인

N_{m-1} 을 대입하면 c에 ' $\left\lfloor \frac{N_{m-1} \cdot 1280}{2048} + b + 1 \right\rfloor - 1$ '가 할당된다.

[0075] 그리고 d 지점의 경우, (n+1) 프레임의 A 비트스트림이 B 비트스트림의 (m-1) 프레임과 m 프레임 모두에 걸쳐있으므로 분할지점 계산부(710)는 수학식 3을 적용하여 d 지점을 계산한다. 수학식 3에 A 비트스트림의 (n+1) 프레임에 대한 길이인 '1280', B 비트스트림의 (m-1) 프레임에 대한 길이인 '2048', (m-1) 프레임의 B 비트스트림에 해당하는 길이인 N_{m-1} 및 m 프레임의 B 비트스트림에 해당하는 길이인 N_m 을 대입하면 d에

$\left\lfloor \frac{N_m \cdot \left(\frac{1280}{2048} - \frac{N_{m-1} - c + 1}{N_{m-1}} \right)}{1} \right\rfloor - 1$ '가 할당된다.

[0076] 마지막으로 e 지점의 경우, (n+2) 프레임의 A 비트스트림이 B 비트스트림의 m 프레임과 (m+1) 프레임 모두에 걸쳐있으므로 분할지점 계산부(710)는 수학식 3을 적용하여 e 지점을 계산한다. 수학식 3에 A 비트스트림의 (n+2) 프레임에 대한 길이인 560 B 비트스트림의 m 프레임에 대한 길이인 '2048', m 프레임의 B 비트스트림에 해당하는 길이인 N_m 및 (m+1) 프레임의 B 비트스트림에 해당하는 길이인 N_{m+1} 을 대입하면 e에

$\left\lfloor \frac{N_{m+1} \cdot \left(\frac{2560}{2048} - \frac{N_m - d + 1}{N_m} \right)}{1} \right\rfloor - 1$ '가 할당된다.

[0077] 비트스트림 분할부(720)는 분할지점 계산부(710)에서 계산된 지점에서 B 비트스트림을 분할한다. 전술한 첫째, 둘째 및 넷째의 경우에는 B 비트스트림에서 m 프레임 가운데 해당하는 부분만을 분할하지만 셋째의 경우에는 B 비트스트림에서 m 프레임 및 (m+1) 프레임 양 프레임들에서 해당하는 부분들을 분할한다. 다시 말하면, 첫째,

둘째 및 넷째의 경우에는 B 비트스트림에서 $P^s_m(n)$ 부터 $P^e_m(n)$ 까지에 해당하는 비트스트림만을 분할하지만, 셋째의 경우에는 B 비트스트림에서 $P^s_m(n)$ 부터 $P^e_m(n)$ 까지에 해당하는 비트스트림과 $P^{sm}_m(n)$ 부터 $P^{em}_m(n)$ 까지에 해당하는 비트스트림을 분할한다.

- [0078] 다중화부(730)는 n 프레임의 A 비트스트림과 비트스트림 분할부(720)에서 분할된 m 프레임 또는 (m+1) 프레임의 B 비트스트림(들)을 다중화한다. 여기서, 다중화부(730)는 비트스트림 정렬부(733) 및 비트스트림 다중화부(736)를 포함하여 이루어진다.
- [0079] 비트스트림 정렬부(733)는 n 프레임의 A 비트스트림과 비트스트림 분할부(720)에서 분할된 m 프레임 또는 (m+1) 프레임의 B 비트스트림(들)을 정렬한다. 여기서, 비트스트림 정렬부(733)는 n 프레임의 A 비트스트림의 후단에 비트스트림 분할부(720)에서 분할된 m 프레임 또는 (m+1) 프레임의 B 비트스트림(들)이 마련되도록 정렬하는 것이 바람직하다. 그러나 n 프레임의 A 비트스트림의 전단에 비트스트림 분할부(720)에서 분할된 m 프레임 또는 (m+1) 프레임의 B 비트스트림이 마련될 수도 있다.
- [0080] 비트스트림 다중화부(736)는 비트스트림 정렬부(730)에서 정렬된 비트스트림들을 다중화하여 출력단자 OUT을 통해 출력한다.
- [0081] 도 8은 본 발명에 의한 비동기화된 비트스트림들의 역다중화 방법에 대한 일 실시예를 흐름도로 도시한 것이다.
- [0082] 먼저, 부호화단에서 서로 다른 프레임의 길이로 부호화하는 복수의 부호화기들에서 부호화된 비트스트림들이 다중화된 스트림을 입력받아 역다중화한다(제800단계). 제800단계에서 입력받은 스트림은 기준 비트스트림을 제외한 나머지 비트스트림들이 부호화단에서 적어도 하나 이상으로 분할된 각 비트스트림의 데이터에 대한 크기와 관련된 정보를 포함하지 않는다.
- [0083] 제800단계에서 역다중화된 비트스트림들 가운데 기준이 되는 비트스트림으로 설정된 기준 비트스트림의 각 프레임에 대한 길이를 검출한다(제810단계). 여기서, 기준 비트스트림은 비트 할당량이 많은 비트스트림 또는 프레임의 길이가 짧은 비트스트림인 것이 바람직하다.
- [0084] 제810단계에서 검출된 각 프레임에 대한 길이를 이용하여 기준 비트스트림을 제외한 나머지 분할된 각 비트스트림의 데이터 크기를 계산한다(제820단계).
- [0085] 제820단계에서 계산된 데이터 크기에 해당하는 나머지 분할된 비트스트림들을 제800단계에서 역다중화된 비트스트림들 가운데 추출한다(제830단계).
- [0086] 제830단계에서 추출된 비트스트림들을 합성한다(제840단계).
- [0087] 도 9는 본 발명에 의한 비동기화된 비트스트림들의 역다중화 장치의 일 실시예를 블록도로 도시한 것으로서, 비동기화된 비트스트림들의 역다중화 장치는 역다중화부(900), 길이 검출부(910), 데이터크기 계산부(920), 비트스트림 추출부(930) 및 비트스트림 합성부(940)를 포함하여 이루어진다.
- [0088] 역다중화부(900)는 부호화단에서 서로 다른 프레임의 길이로 부호화하는 복수의 부호화기들에서 부호화된 비트스트림들이 다중화된 스트림을 입력단자 IN을 통해 입력받아 역다중화한다. 역다중화부(900)에서 입력받은 스트림은 기준 비트스트림을 제외한 나머지 비트스트림들이 부호화단에서 적어도 하나 이상으로 분할된 각 비트스트림의 데이터에 대한 크기와 관련된 정보를 포함하지 않는다.
- [0089] 길이 검출부(910)는 역다중화부(900)에서 역다중화된 비트스트림들 가운데 기준이 되는 비트스트림으로 설정된 기준 비트스트림의 각 프레임에 대한 길이를 검출한다. 여기서, 기준 비트스트림은 비트 할당량이 많은 비트스트림 또는 프레임의 길이가 짧은 비트스트림인 것이 바람직하다.
- [0090] 데이터크기 계산부(920)는 길이 검출부(910)에서 검출된 각 프레임에 대한 길이를 이용하여 기준 비트스트림을 제외한 나머지 분할된 각 비트스트림의 데이터 크기를 계산한다.
- [0091] 비트스트림 추출부(930)는 데이터크기 계산부(920)에서 계산된 데이터 크기에 해당하는 나머지 분할된 비트스트림들을 역다중화부(900)에서 역다중화된 비트스트림들 가운데 추출한다.
- [0092] 비트스트림 합성부(940)는 비트스트림 추출부(930)에서 추출된 비트스트림들을 합성하여 출력단자 OUT을 통해 출력한다.
- [0093] 이러한 본 발명에 대한 이해를 돕기 위하여 도면에 도시된 실시예를 참고로 설명되었으나, 이는 예시적인 것에

불과하며, 당해 분야에서 통상적 지식을 가진 자라면 이로부터 다양한 변형 및 균등한 타 실시예가 가능하다는 점을 이해할 것이다. 따라서, 본 발명의 진정한 기술적 보호 범위는 첨부된 특허청구범위에 의해 정해져야 할 것이다.

[0094] 또한, 본 발명은 컴퓨터로 읽을 수 있는 기록 매체에 컴퓨터(정보 처리 기능을 갖는 장치를 모두 포함한다)가 읽을 수 있는 코드로서 구현하는 것이 가능하다. 컴퓨터가 읽을 수 있는 기록 매체는 컴퓨터 시스템에 의하여 읽혀질 수 있는 데이터가 저장되는 모든 종류의 기록 장치를 포함한다. 컴퓨터가 읽을 수 있는 기록 장치의 예로는 ROM, RAM, CD-ROM, 자기 테이프, 플로피 디스크, 광데이터 저장 장치 등이 있다.

발명의 효과

[0095] 본 발명에 의한 비동기화된 비트스트림들의 다중화 방법 및 장치에 의하면, 서로 다른 프레임의 길이로 부호화하는 복수의 부호화기들에서 부호화된 비트스트림들 가운데 기준 비트스트림의 각 프레임에 대한 길이를 기준으로 나머지 비트스트림들을 분할하여 다중화한다.

[0096] 이렇게 함으로써 기준 비트스트림을 제외한 나머지 비트스트림들이 분할된 각 비트스트림의 데이터 크기에 대한 정보를 생성하여 부호화단에서 전송하지 않아도 되므로 부호화하는 과정의 복잡도를 경감시킬 뿐만 아니라 부호화를 수행하는 시간을 절감할 수 있으며 부호화단에서 복호화단으로 전송하는 데이터의 크기를 줄일 수 있는 효과를 거둘 수 있다. 또한, 복호화단에서도 나머지 비트스트림들이 분할된 각 비트스트림의 데이터 크기에 대한 정보가 없어도 역다중화하여 복호화를 수행할 수 있는 효과를 거둘 수 있다. 이에 의하여 부호화단 및 복호화단에서 비트레이트(bitrate)에 대한 제어를 보다 정확하고 효율적으로 수행할 수 있다.

[0097] 그리고 나머지 비트스트림들을 기준 비트스트림에 대하여 부호화단에서 효율적으로 분할하여 복호화단으로 전송하므로 음질 또는 화질을 보다 향상시킬 수 있는 효과를 거둘 수 있다.

도면의 간단한 설명

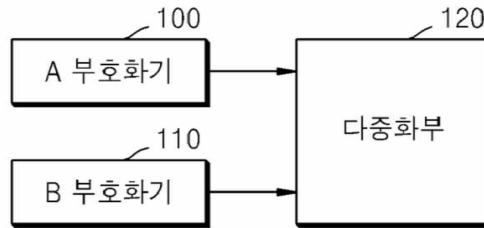
- [0001] 도 1은 종래의 비트스트림들을 다중화하는 장치를 설명하기 위한 블록도를 도시한 것이다.
- [0002] 도 2는 동기 시간 정렬 방식을 설명하기 위한 개념도를 도시한 것이다.
- [0003] 도 3은 비동기 시간 정렬 방식을 설명하기 위한 개념도를 도시한 것이다.
- [0004] 도 4는 종래의 비동기 시간 정렬 방식에서 다중화하는 방법 및 장치를 설명하기 위한 개념도를 도시한 것이다.
- [0005] 도 5는 종래의 비동기 시간 정렬 방식에서 다중화하는 방법 및 장치에 대한문제점을 설명하기 위한 개념도를 도시한 것이다.
- [0006] 도 6은 본 발명에 의한 비동기화된 비트스트림들의 다중화 방법에 대한 일 실시예를 흐름도로 도시한 것이다.
- [0007] 도 7은 본 발명에 의한 비동기화된 비트스트림들의 다중화 장치에 대한 일 실시예를 블록도로 도시한 것이다.
- [0008] 도 8은 본 발명에 의한 비동기화된 비트스트림들의 역다중화 방법에 대한 일 실시예를 흐름도로 도시한 것이다.
- [0009] 도 9는 본 발명에 의한 비동기화된 비트스트림들의 역다중화 장치에 대한 일 실시예를 블록도로 도시한 것이다.
- [0010] 도 10는 h^A_n 과 h^B_m 가 일치하고 t^A_n 가 h^B_m 와 t^B_m 사이에 위치하는 경우의 일 실시예를 개념도로 도시한 것이다.
- [0011] 도 11는 t^A_n 과 t^B_m 가 일치하고 h^A_n 가 h^B_m 와 t^B_m 사이에 위치하는 경우의 일 실시예를 개념도로 도시한 것이다.
- [0012] 도 12는 h^A_n 가 h^B_m 와 t^B_m 사이에 위치하고 t^A_n 가 h^B_{m+1} 와 t^B_{m+1} 사이에 위치하는 경우의 일 실시예를 개념도로 도시한 것이다.
- [0013] 도 13는 h^A_n 가 h^B_m 보다 뒤에 위치하고 t^A_n 가 t^B_m 보다 앞에 위치하는 경우의 일 실시예를 개념도로 도시한 것이다.
- [0014] 도 14는 본 발명에 의한 비동기화된 비트스트림들의 다중화 방법 및 장치에 의하여 기준 비트스트림을 제외한

나머지 비트스트림을 분할하는 데이터 크기를 계산하는 일 실시예를 개념도로 도시한 것이다.

- [0015] <도면의 주요 부호에 대한 간단한 설명>
- [0016] 700: 프레임길이 검출부 710: 분할지점 계산부
- [0017] 720: 비트스트림 분할부 730: 다중화부
- [0018] 733: 비트스트림 정렬부 736: 비트스트림 다중화부

도면

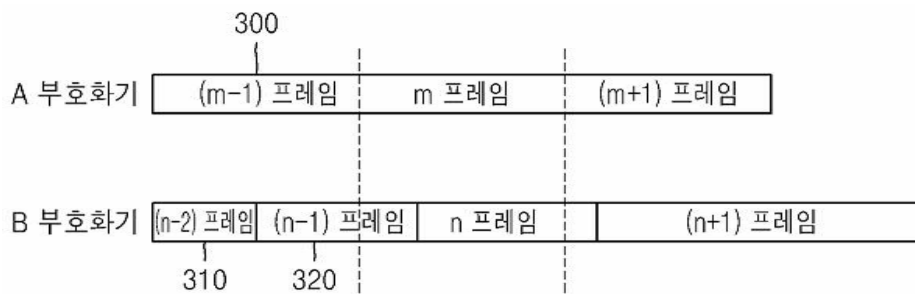
도면1



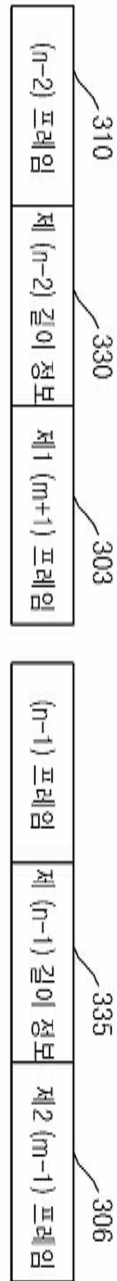
도면2



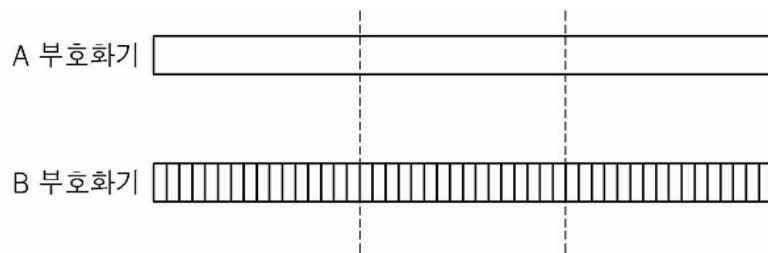
도면3



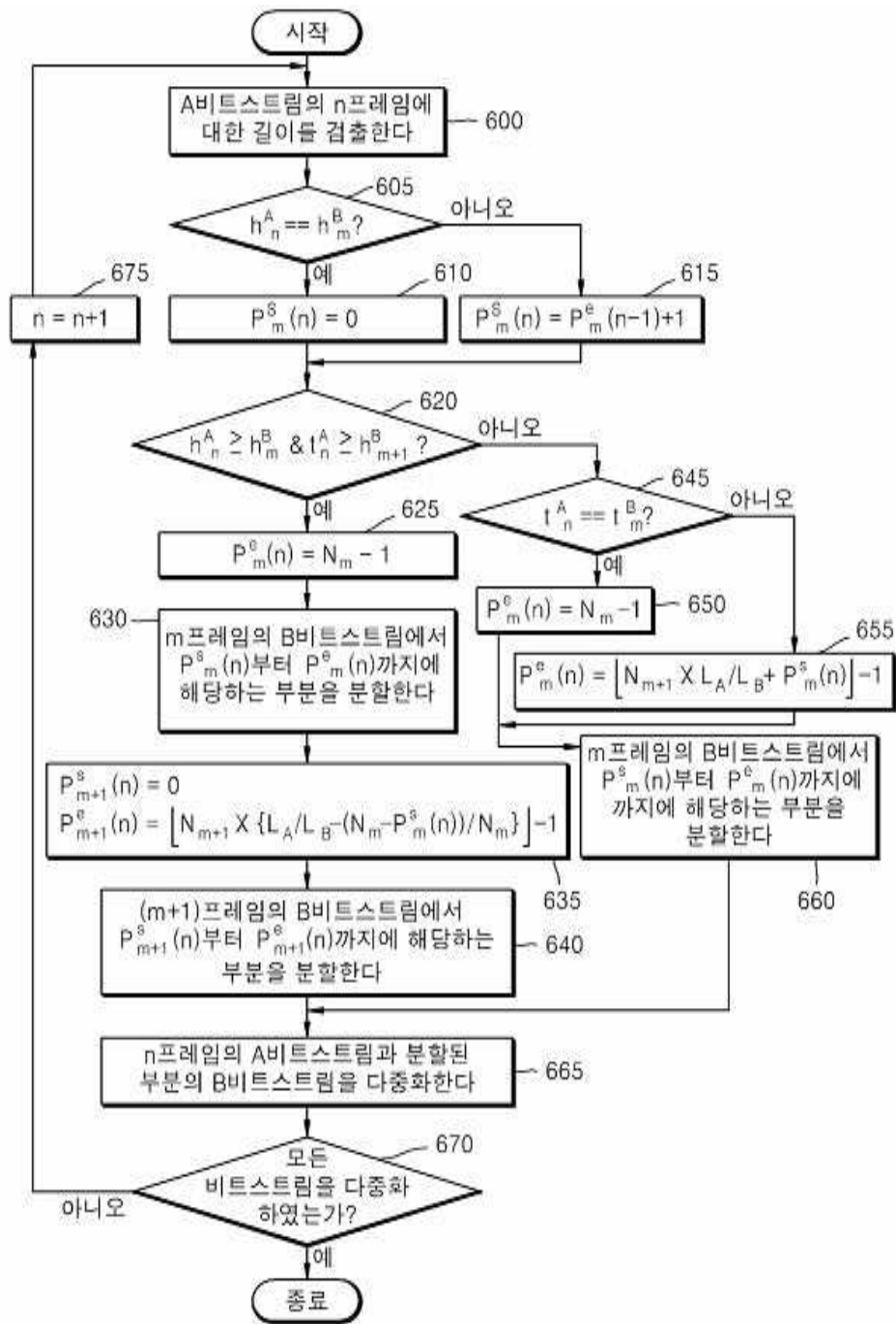
도면4



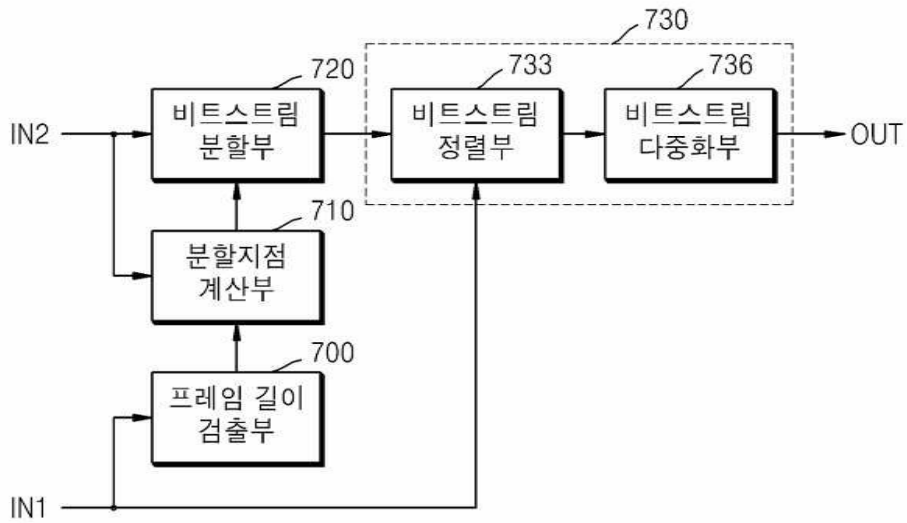
도면5



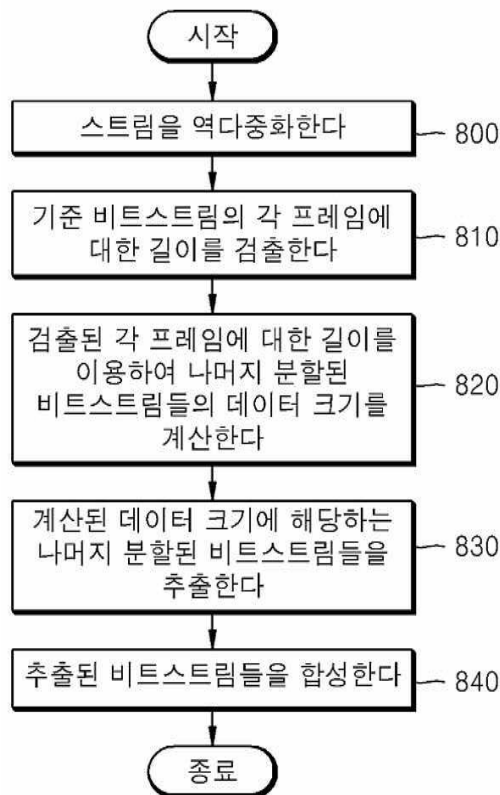
도면6



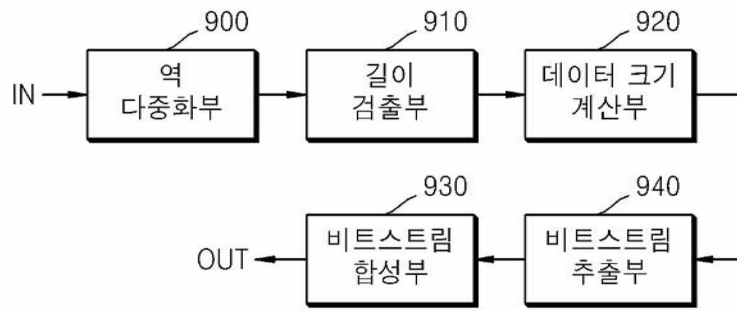
도면7



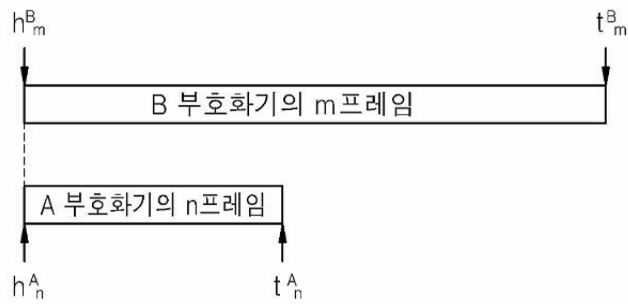
도면8



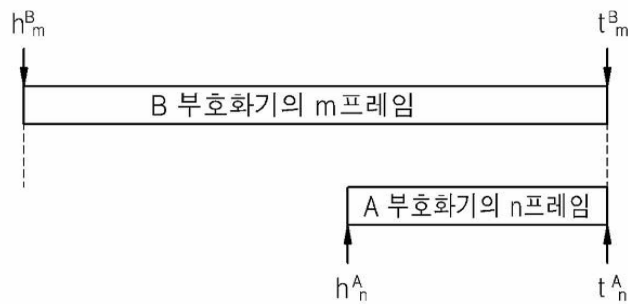
도면9



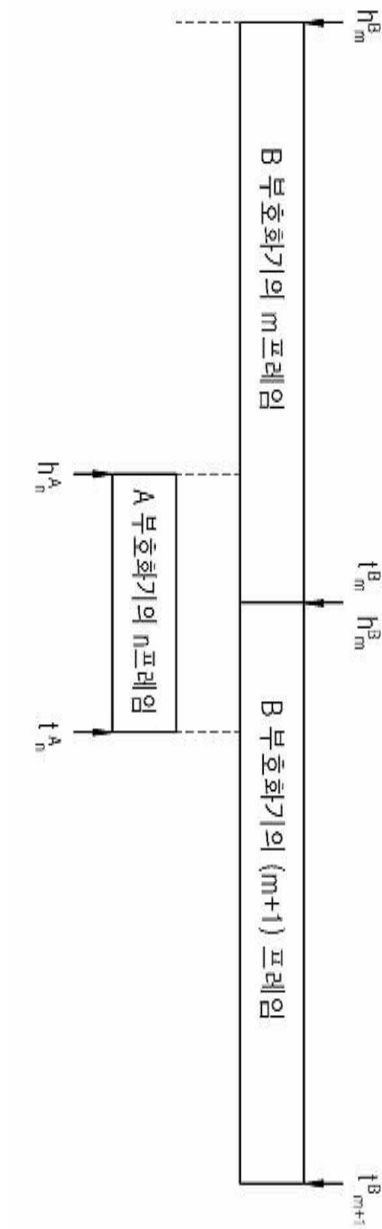
도면10



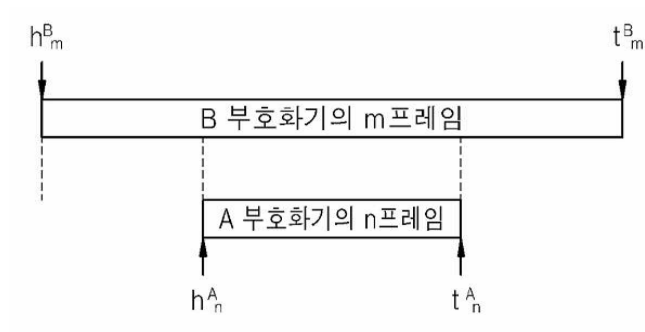
도면11



도면12



도면13



도면14

