



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2021년03월04일
(11) 등록번호 10-2223694
(24) 등록일자 2021년02월26일

(51) 국제특허분류(Int. Cl.)
HO4N 19/187 (2014.01) *HO4N 19/30* (2014.01)
HO4N 19/423 (2014.01) *HO4N 19/597* (2014.01)
HO4N 19/70 (2014.01) *HO4N 21/2343* (2011.01)
HO4N 21/44 (2011.01) *HO4N 21/845* (2011.01)
 (52) CPC특허분류
HO4N 19/187 (2015.01)
HO4N 19/30 (2015.01)
 (21) 출원번호 10-2015-7031996
 (22) 출원일자(국제) 2014년04월04일
 심사청구일자 2019년03월28일
 (85) 번역문제출일자 2015년11월06일
 (65) 공개번호 10-2015-0140777
 (43) 공개일자 2015년12월16일
 (86) 국제출원번호 PCT/JP2014/001967
 (87) 국제공개번호 WO 2014/167817
 국제공개일자 2014년10월16일
 (30) 우선권주장
 13/858,076 2013년04월07일 미국(US)
 (뒷면에 계속)

(73) 특허권자
돌비 인터네셔널 에이비
 네덜란드 1101 씨엔 암스트레담 주이두스트 헤리
 커베르그벡 1-35 3이 아폴로 빌딩
 (72) 발명자
데시판드 사친 쥐.
 미국 워싱턴주 98607 카마스 에스이 포터세컨드
 씨클 19514
 (74) 대리인
김태홍, 김진희

(56) 선행기술조사문헌
 S. Deshpande. On Sub-picture Based CPB.
 JCT-VC of ITU-T and ISO/IEC. JCTVC-J0306
 Ver.2, Jul. 16, 2012, pp.1-8
 (뒷면에 계속)

전체 청구항 수 : 총 1 항

심사관 : 이상래

(54) 발명의 명칭 **출력 계층 세트들에서의 시그널링 변경**

(57) 요약

비디오 비트스트림을 디코딩하기 위한 시스템은 비디오 파라미터 세트 및 비디오 파라미터 세트 확장을 수신하는 것과 함께, 비트스트림 및 복수의 강화 비트스트림들을 수신하는 것을 포함한다. 시스템은 또한, 적어도 하나의 출력 계층 세트에서의 변경을 표시하는 정보를 포함하는 출력 계층 세트 변경 메시지를 수신한다.

대표도 - 도36

output_layer_sets_change(payloadSize) {	디스크립터
active_vps_id	u(4)
num_changed_output_layer_sets	ue(v)
for(i=0; i<num_changed_output_layer_sets; i++){	
changed_output_layer_set_idx_entry[i]	ue(v)
clsIdx = output_layer_set_idx[changed_output_layer_set_idx_entry[i]]	
for(j=0; j<=vps_max_layer_id; j++){	
if(layer_id_included_flag[clsIdx][j])	
output_layer_flag[clsIdx][j]	u(1)
}	
num_addl_output_layer_sets	
for(i=num_output_layer_sets; i<num_output_layer_sets+num_addl_output_layer_sets; i++){	
addlIdx = addl_output_layer_sets_idx[i]	ue(v)
for(j=0; j<=vps_max_layer_id; j++){	
if(layer_id_included_flag[addlIdx][j])	
output_layer_flag[addlIdx][j]	u(1)
}	
}	

(52) CPC특허분류

HO4N 19/423 (2015.01)
HO4N 19/597 (2015.01)
HO4N 19/70 (2015.01)
HO4N 21/234327 (2013.01)
HO4N 21/44004 (2013.01)
HO4N 21/8451 (2013.01)

(56) 선행기술조사문헌

S. Deshpande, et al. Sub-picture Based CPB Removal Timing. JCT-VC of ITU-T and ISO/IEC. JCTVC-J0569 Ver.3, Jul. 16, 2012, pp.1-2
Y-K. Wang, et al. AHG9: HeVC HRD cleanups. JCT-VC of ITU-T and ISO/IEC. JCTVC-L0044 Ver.6, Jan. 22, 2012, pp.1-4

KR1020150003235 A

KR1020150003236 A

(30) 우선권주장

61/844,272 2013년07월09일 미국(US)

61/845,309 2013년07월11일 미국(US)

61/856,575 2013년07월19일 미국(US)

명세서

청구범위

청구항 1

비디오 비트스트림을 디코딩하기 위한 방법에 있어서,

(a) 액세스 유닛 레벨로 코딩된 픽처 버퍼로부터 액세스 유닛을 제거하기 위해 코딩된 픽처 버퍼 제거 지연 파라미터를 포함하는 픽처 타이밍 파라미터 세트를 수신하는 단계;

(b) 코딩된 비디오 이미지를 나타내는 데이터를 수신하는 단계;

(c) 상기 코딩된 픽처 버퍼에 상기 데이터를 저장하는 단계;

(d) 상기 액세스 유닛이 상기 액세스 유닛 레벨로 디코딩을 위해 상기 코딩된 픽처 버퍼로부터 제거되어야 할 것인지, 또는 디코딩 유닛이 서브-픽처 레벨로 디코딩을 위해 상기 코딩된 픽처 버퍼로부터 제거되어야 할 것인지 결정하는 단계;

(e) 상기 디코딩 유닛이 상기 서브-픽처 레벨로 디코딩을 위해 상기 코딩된 픽처 버퍼로부터 제거되어야 할 때:

(i) 상기 비트스트림이 액세스 유닛 내의 모든 디코딩 유닛에 대한 공통 제거 지연을 나타내는 파라미터를 포함하는지 여부를 나타내는 플래그를 수신하는 단계,

(ii) 상기 플래그가 상기 모든 디코딩 유닛에 대한 공통 제거 지연을 나타내는 파라미터가 상기 비트스트림 내에 있다고 나타내는 경우, 상기 공통 제거 지연이, 상기 코딩된 픽처 버퍼로부터의 현재 디코딩 유닛의 제거 시간과 상기 코딩된 픽처 버퍼로부터의 바로 앞의 디코딩 유닛의 제거 시간 사이의 지속기간인 것인, 모든 디코딩 유닛에 대한 공통 제거 지연을 나타내는 파라미터를 수신하는 단계,

(iii) 상기 플래그가 상기 모든 디코딩 유닛에 대한 공통 제거 지연을 나타내는 파라미터가 상기 비트스트림 내에 있지 않다고 나타내는 경우, 상기 액세스 유닛 내의 연속 디코딩 유닛들 사이의 지연을 각각 나타내는 개별 디코딩 유닛 지연 파라미터를 수신하는 단계,

(iv) 상기 공통 제거 지연이나 상기 개별 디코딩 유닛 지연 파라미터 중의 어느 하나를 나타내는 상기 파라미터에 기초하여 상기 코딩된 픽처 버퍼로부터 상기 디코딩 유닛을 제거하는 단계, 및

(v) 상기 제거된 디코딩 유닛을 디코딩하는 단계; 및

(f) 상기 액세스 유닛이 상기 액세스 유닛 레벨로 디코딩을 위해 상기 코딩된 픽처 버퍼로부터 제거되어야 할 때:

(i) 상기 데이터 내의 상기 액세스 유닛에 대한 액세스 유닛 제거 지연을 결정하는 단계,

(ii) 상기 액세스 유닛 제거 지연에 기초하여 상기 코딩된 픽처 버퍼로부터 상기 액세스 유닛을 제거하는 단계, 및

(iii) 상기 제거된 액세스 유닛을 디코딩하는 단계

를 포함하고,

상기 디코딩 유닛은 상기 액세스 유닛의 서브세트이고,

상기 액세스 유닛 레벨로 상기 코딩된 픽처 버퍼로부터 상기 액세스 유닛을 제거하기 위한 상기 코딩된 픽처 버퍼 제거 지연 파라미터와 곱해진 클록 틱(clock tick)(t_c)의 값은, 상기 액세스 유닛 내의 모든 개별 디코딩 유닛 지연 파라미터의 합과 곱해진 서브-픽처 클록 틱의 값($t_{c,sub}$)과 동일하며, 상기 액세스 유닛 제거 지연은 클록 틱의 수로 나타나고, 상기 개별 디코딩 유닛 지연 파라미터는 서브-픽처 클록 틱의 수로 나타나는 것인, 비디오 비트스트림을 디코딩하기 위한 방법.

청구항 2

삭제

청구항 3

삭제

청구항 4

삭제

청구항 5

삭제

청구항 6

삭제

청구항 7

삭제

청구항 8

삭제

청구항 9

삭제

청구항 10

삭제

청구항 11

삭제

청구항 12

삭제

청구항 13

삭제

청구항 14

삭제

청구항 15

삭제

청구항 16

삭제

청구항 17

삭제

청구항 18

삭제

청구항 19

삭제

청구항 20

삭제

청구항 21

삭제

청구항 22

삭제

청구항 23

삭제

청구항 24

삭제

청구항 25

삭제

청구항 26

삭제

청구항 27

삭제

청구항 28

삭제

청구항 29

삭제

청구항 30

삭제

청구항 31

삭제

청구항 32

삭제

청구항 33

삭제

청구항 34

삭제

청구항 35

삭제

청구항 36

삭제

청구항 37

삭제

청구항 38

삭제

청구항 39

삭제

청구항 40

삭제

청구항 41

삭제

청구항 42

삭제

청구항 43

삭제

청구항 44

삭제

청구항 45

삭제

청구항 46

삭제

청구항 47

삭제

청구항 48

삭제

청구항 49

삭제

청구항 50

삭제

청구항 51

삭제

청구항 52

삭제

청구항 53

삭제

청구항 54

삭제

청구항 55

삭제

청구항 56

삭제

청구항 57

삭제

청구항 58

삭제

청구항 59

삭제

청구항 60

삭제

청구항 61

삭제

청구항 62

삭제

청구항 63

삭제

청구항 64

삭제

청구항 65

삭제

청구항 66

삭제

청구항 67

삭제

발명의 설명

기술 분야

[0001] 본 개시물은 일반적으로 전자 디바이스들에 관한 것이다. 더욱 구체적으로, 본 개시물은 서브-픽처 기반 가설 참조 디코더 파라미터들을 시그널링하기 위한 전자 디바이스들, 및 디코딩된 픽처 버퍼(decoded picture buffer; DPB)의 하이브리드 동작을 위한 시스템들 및 방법들에 관한 것이다.

배경 기술

[0002] 전자 디바이스들은 소비자의 필요들을 충족시키고 휴대성 및 편리성을 개선시키기 위하여 더 소형화되고 더욱 강력하게 되었다. 소비자들은 전자 디바이스들에 의존하게 되었으며 증가된 기능성을 기대하게 되었다. 전자 디바이스들의 일부의 예들은 데스크톱 컴퓨터들, 랩톱 컴퓨터들, 셀룰러 전화들, 스마트폰들, 미디어 플레이어 들, 집적 회로들 등을 포함한다.

[0003] 일부의 전자 디바이스들은 디지털 미디어를 프로세싱하고 디스플레이하기 위해 이용된다. 예를 들어, 휴대용 전자 디바이스들은 이제 디지털 미디어가 소비자가 있을 수도 있는 거의 임의의 장소에서 소비되도록 한다. 또한, 일부의 전자 디바이스들은 소비자의 이용 및 향유를 위해 디지털 미디어 콘텐츠의 다운로드 또는 스트리밍 을 제공할 수도 있다.

[0004] 디지털 미디어의 증가하는 인기는 몇몇 문제들을 제시하였다. 예를 들어, 저장, 송신 및 신속한 재생을 위하여 고품질 디지털 미디어를 효율적으로 표현하는 것은 몇몇 도전들을 제시한다. 이 논의로부터 관찰될 수 있는 바 와 같이, 개선된 성능으로 디지털 미디어를 효율적으로 표현하는 시스템들 및 방법들이 유익할 수도 있다.

[0005] 발명의 상기한 그리고 다른 목적들, 특징들, 및 장점들은 첨부한 도면들과 함께 취해진, 발명의 다음의 상세한 설명의 고려 시에 더욱 용이하게 이해될 것이다.

발명의 내용

[0006] 발명의 양태는 비디오 비트스트림을 디코딩하기 위한 방법을 제공하고, 상기 방법은, (a) 코딩된 비디오 시퀀스 를 나타내는 기본 비트스트림(base bitstream)을 수신하는 단계; (b) 상기 코딩된 비디오 시퀀스를 나타내는 복 수의 강화 비트스트림(enhancement bitstream)들을 수신하는 단계; (c) 상기 기본 비트스트림 및 상기 복수의 강화 비트스트림들에 적용되는 신택스 엘리먼트(syntax element)들을 포함하는 비디오 파라미터 세트를 수신하 는 단계로서, 상기 비디오 파라미터 세트는 비디오 파라미터 세트 확장을 시그널링하는 신택스 엘리먼트를 포함 하는, 상기 비디오 파라미터 세트를 수신하는 단계; (d) 상기 강화 비트스트림들 중의 적어도 하나에 관련된 신택스 엘리먼트들을 포함하는 상기 비디오 파라미터 세트 확장을 수신하는 단계; (e) 적어도 하나의 출력 계층 세트에서의 변경을 표시하는 정보를 포함하는 출력 계층 세트 변경 메시지를 수신하는 단계를 포함한다.

[0007] 발명의 양태는 비디오 비트스트림을 디코딩하기 위한 방법을 제공하고, 상기 방법은, (a) 코딩된 비디오 시퀀스 를 나타내는 기본 비트스트림을 수신하는 단계; (b) 상기 코딩된 비디오 시퀀스를 나타내는 복수의 강화 비트스 트림들을 수신하는 단계; (c) 상기 기본 비트스트림 및 상기 복수의 강화 비트스트림들에 적용되는 신택스 엘리먼트들을 포함하는 비디오 파라미터 세트를 수신하는 단계로서, 상기 비디오 파라미터 세트는 비디오 파라미터 세트 확장을 시그널링하는 신택스 엘리먼트를 포함하는, 상기 비디오 파라미터 세트를 수신하는 단계; (d) 상기 강화 비트스트림들 중의 적어도 하나에 대한 디코딩된 픽처 버퍼를 위한 디코딩된 픽처 버퍼 관련 파라미터들을 포함하는 신택스 엘리먼트들을 포함하는 상기 비디오 파라미터 세트 확장을 수신하는 단계를 포함한다.

[0008] 발명의 양태는 비디오 코딩을 위한 방법을 제공하고, 상기 방법은, 현재의 픽처의 최초 슬라이스 헤더를 파싱하 는 것을 시작하는 단계; 디코딩된 픽처 버퍼(DPB)에 의해 수행된 어느 단계들이 픽처 기반일 것인지와, 어느 단 계들이 액세스 유닛(access unit; AU) 기반일 것인지를 결정하는 단계; DPB로부터의 제거를 수행하는 단계; DPB 로부터의 픽처 출력을 수행하는 단계; DPB에서의 현재의 디코딩된 픽처의 디코딩 및 저장을 수행하는 단계; DPB 에서의 현재의 디코딩된 픽처를 표기(mark)하는 단계; 및 DPB로부터의 추가적인 픽처 출력을 수행하는 단계를

포함한다.

[0009]

발명의 양태는 비디오 코딩을 위해 구성된 전자 디바이스를 제공하고, 상기 전자 디바이스는 프로세서; 프로세서와 전자 통신하는 메모리를 포함하고, 메모리에 저장된 명령들은, 현재의 픽처의 최초 슬라이스 헤더를 파싱하는 것을 시작하고; 디코딩된 픽처 버퍼(DPB)에 의해 수행된 어느 단계들이 픽처 기반일 것인지와, 어느 단계들이 액세스 유닛(AU) 기반일 것인지를 결정하고; DPB로부터의 제거를 수행하고; DPB로부터의 픽처 출력을 수행하고; DPB에서의 현재의 디코딩된 픽처의 디코딩 및 저장을 수행하고; DPB에서의 현재의 디코딩된 픽처를 표기하고; 그리고 DPB로부터의 추가적인 픽처 출력을 수행하도록 실행가능하다.

도면의 간단한 설명

[0010]

도 1a는 메시지를 전송하고 비트스트림을 버퍼링하기 위한 시스템들 및 방법들이 구현될 수도 있는 하나 이상의 전자 디바이스들의 예를 예시하는 블록도이다.

도 1b는 메시지를 전송하고 비트스트림을 버퍼링하기 위한 시스템들 및 방법들이 구현될 수도 있는 하나 이상의 전자 디바이스들의 예를 예시하는 또 다른 블록도이다.

도 2는 메시지를 전송하기 위한 방법의 하나의 구성을 예시하는 흐름도이다.

도 3은 액세스 유닛에서의 디코딩 유닛들에 대한 하나 이상의 제거 지연들을 결정하기 위한 방법의 하나의 구성을 예시하는 흐름도이다.

도 4는 비트스트림을 버퍼링하기 위한 방법의 하나의 구성을 예시하는 흐름도이다.

도 5는 액세스 유닛에서의 디코딩 유닛들에 대한 하나 이상의 제거 지연들을 결정하기 위한 방법의 하나의 구성을 예시하는 흐름도이다.

도 6a는 전자 디바이스 상의 인코더(604)의 하나의 구성을 예시하는 블록도이다.

도 6b는 전자 디바이스 상의 인코더(604)의 하나의 구성을 예시하는 또 다른 블록도이다.

도 7a는 전자 디바이스 상의 디코더의 하나의 구성을 예시하는 블록도이다.

도 7b는 전자 디바이스 상의 디코더의 하나의 구성을 예시하는 또 다른 블록도이다.

도 8은 송신 전자 디바이스에서 사용될 수도 있는 다양한 컴포넌트(component)들을 예시한다.

도 9는 수신 전자 디바이스에서 사용될 수도 있는 다양한 컴포넌트들을 예시하는 블록도이다.

도 10은 메시지를 전송하기 위한 시스템들 및 방법들이 구현될 수도 있는 전자 디바이스의 하나의 구성을 예시하는 블록도이다.

도 11은 비트스트림을 버퍼링하기 위한 시스템들 및 방법들이 구현될 수도 있는 전자 디바이스의 하나의 구성을 예시하는 블록도이다.

도 12는 디코딩된 픽처 버퍼의 동작을 위한 방법의 하나의 구성을 예시하는 블록도이다.

도 13a는 상이한 NAL 유닛 헤더 선택스(NAL Unit header syntax)를 예시한다.

도 13b는 상이한 NAL 유닛 헤더 선택스를 예시한다.

도 13c는 상이한 NAL 유닛 헤더 선택스를 예시한다.

도 14는 일반적인 NAL 유닛 선택스를 예시한다.

도 15는 기존의 비디오 파라미터 세트를 예시한다.

도 16은 기존의 스케일러빌리티 타입(scalability type)들을 예시한다.

도 17은 예시적인 비디오 파라미터 세트를 예시한다.

도 18은 예시적인 스케일러빌리티 맵 선택스(scalability map syntax)를 예시한다.

도 19는 예시적인 비디오 파라미터 세트를 예시한다.

도 20은 기존의 비디오 파라미터 세트를 예시한다.

- 도 21은 기존의 차원 타입, 차원 id 선택스를 예시한다.
- 도 22는 예시적인 비디오 파라미터 세트를 예시한다.
- 도 23은 예시적인 스케일러빌리티 맵 선택스를 예시한다.
- 도 24는 예시적인 비디오 파라미터 세트를 예시한다.
- 도 25는 예시적인 비디오 파라미터 세트를 예시한다.
- 도 26은 예시적인 비디오 파라미터 세트를 예시한다.
- 도 27은 예시적인 스케일러빌리티 마스크 선택스(scalability mask syntax)를 예시한다.
- 도 28은 예시적인 비디오 파라미터 세트 확장 선택스를 예시한다.
- 도 29는 예시적인 비디오 파라미터 세트 확장 선택스를 예시한다.
- 도 30은 예시적인 비디오 파라미터 세트 확장 선택스를 예시한다.
- 도 31은 예시적인 비디오 파라미터 세트 확장 선택스를 예시한다.
- 도 32는 예시적인 비디오 파라미터 세트 확장 선택스를 예시한다.
- 도 33은 예시적인 비디오 파라미터 세트 확장 선택스를 예시한다.
- 도 34는 예시적인 비디오 파라미터 세트 선택스를 예시한다.
- 도 35는 예시적인 비디오 파라미터 세트 확장 선택스를 예시한다.
- 도 36은 예시적인 출력 계층 세트들 변경 선택스를 예시한다.
- 도 37은 또 다른 예시적인 출력 계층 세트들 변경 선택스를 예시한다.
- 도 38a는 예시적인 비디오 파라미터 확장 선택스를 예시한다.
- 도 38b는 예시적인 비디오 파라미터 확장 선택스를 예시한다.
- 도 39a는 예시적인 op_dpb_info_parameters(j) 선택스를 예시한다.
- 도 39b는 예시적인 op_dpb_info_parameters(j) 선택스를 예시한다.
- 도 40은 또 다른 예시적인 비디오 파라미터 확장 선택스를 예시한다.
- 도 41은 또 다른 예시적인 oop_dpb_info_parameters(j) 선택스를 예시한다.
- 도 42는 또 다른 예시적인 oop_dpb_info_parameters(j) 선택스를 예시한다.
- 도 43은 예시적인 num_dpb_info_parameters 선택스를 예시한다.
- 도 44는 또 다른 예시적인 oop_dpb_info_parameters(j) 선택스를 예시한다.
- 도 45는 또 다른 예시적인 num_dpb_info_parameters 선택스를 예시한다.
- 도 46은 또 다른 예시적인 num_dpb_info_parameters 선택스를 예시한다.
- 도 47은 또 다른 예시적인 비디오 파라미터 확장 선택스 및 layer_dpb_info(i)를 예시한다.
- 도 48은 예시적인 oop_dpb_info_parameters and layer_dpb_info(i) 선택스를 예시한다.
- 도 49a는 또 다른 예시적인 vps_extension()을 예시한다.
- 도 49b는 또 다른 예시적인 vps_extension()을 예시한다.
- 도 50은 예시적인 oop_dpb_maxbuffering_parameters(i)를 예시한다.
- 도 51은 예시적인 layer_dpb_info_parameters(i)를 예시한다.
- 도 52는 또 다른 예시적인 vps_extension()을 예시한다.
- 도 53은 또 다른 예시적인 vps_extension()을 예시한다.

- 도 54는 예시적인 oop_dpb_maxbuffering_parameters(i,k)를 예시한다.
- 도 55는 예시적인 oop_dpb_maxbuffering_parameters(i,k)를 예시한다.
- 도 56은 또 다른 예시적인 vps_extension()을 예시한다.
- 도 57은 예시적인 oop_dpb_maxbuffering_parameters(i,k)를 예시한다.
- 도 58은 예시적인 oop_dpb_maxbuffering_parameters(i,k)를 예시한다.
- 도 59는 예시적인 oop_dpb_maxbuffering_parameters(i,k)를 예시한다.
- 도 60은 예시적인 oop_dpb_maxbuffering_parameters(i,k)를 예시한다.
- 도 61은 예시적인 oop_dpb_maxbuffering_parameters(i,k)를 예시한다.
- 도 62는 예시적인 seq_parameter_set_rbsp()를 예시한다.
- 도 63은 다수의 전자 디바이스들 사이의 비디오 코딩을 예시하는 블록도이다.
- 도 64는 하이브리드 디코딩된 픽처 버퍼(DPB) 동작을 위한 방법의 흐름도이다.
- 도 65는 하이브리드 디코딩된 픽처 버퍼(DPB) 동작을 위한 또 다른 방법의 흐름도이다.
- 도 66은 디코더의 하나의 구성을 예시하는 블록도이다.
- 도 67a는 기본 계층(base layer) 및 강화 계층(enhancement layer)에 대한 별도의 디코딩된 픽처 버퍼(DPB)들 및 별도의 하이브리드 디코딩된 픽처 버퍼(DPB) 동작 모듈들에 의한 비디오 코딩을 위한 강화 계층 및 기본 계층 양자의 이용을 예시하는 블록도이다.
- 도 67b는 기본 계층 및 강화 계층에 대한 공유된 디코딩된 픽처 버퍼(DPB) 및 공유된 하이브리드 디코딩된 픽처 버퍼(DPB) 동작 모듈의 이용을 예시하는 블록도이다.
- 도 68은 하이브리드 디코딩된 픽처 버퍼(DPB) 동작을 예시하는 타이밍도이다.
- 도 69는 제 2 강화 계층(EL2)이 기본 계층(BL) 및 제 1 강화 계층(EL1)보다 더 낮은 픽처 레이트를 가질 때, 코딩된 픽처들 및 액세스 유닛(AU)들에 대한 계층들의 네트워크 추상화 계층(network abstraction layer; NAL) 유닛들에 대한 구조 및 타이밍을 예시하는 블록도이다.
- 도 70은 기본 계층(BL)이 제 1 강화 계층(EL1) 및 제 2 강화 계층(EL2)보다 더 낮은 픽처 레이트를 가질 때, 코딩된 픽처들 및 액세스 유닛(AU)들에 대한 계층들의 네트워크 추상화 계층(NAL) 유닛들에 대한 구조 및 타이밍을 예시하는 블록도이다.

발명을 실시하기 위한 구체적인 내용

예 1

[0011]

[0012]

메시지를 전송하기 위한 전자 디바이스가 설명된다. 전자 디바이스는 프로세서와, 프로세서와 전자 통신하는 메모리 내에 저장된 명령들을 포함한다. 전자 디바이스는, 코딩된 픽처 버퍼(CPB)가 서브-픽처 레벨 상에서의 동작을 지원할 때, 픽처 타이밍 보충 강화 정보(Supplemental Enhancement Information; SEI) 메시지 내에 공통 디코딩 유닛 CPB 제거 지연 파라미터를 포함할 것인지 여부를 결정한다. 전자 디바이스는 또한, 공통 디코딩 유닛 CPB 제거 지연 파라미터가 픽처 타이밍 SEI 메시지(또는 일부의 다른 SEI 메시지 또는 일부의 다른 파라미터 세트, 예컨대, 픽처 파라미터 세트 또는 시퀀스 파라미터 세트 또는 비디오 파라미터 세트 또는 적응 파라미터 세트(adaptation parameter set)) 내에 포함되어야 할 때, 공통 디코딩 유닛 CPB 제거 지연 파라미터를 생성하고, 여기서, 공통 디코딩 유닛 CPB 제거 지연 파라미터는 CPB로부터의 액세스 유닛에서의 모든 디코딩 유닛들에 적용가능하다. 전자 디바이스는 또한, 공통 디코딩 유닛 CPB 제거 지연 파라미터가 픽처 타이밍 SEI 메시지 내에 포함되지 않아야 할 때, 액세스 유닛에서의 각각의 디코딩 유닛에 대한 별도의 디코딩 유닛 CPB 제거 지연 파라미터를 생성한다. 전자 디바이스는 또한, 공통 디코딩 유닛 CPB 제거 지연 파라미터 또는 디코딩 유닛 CPB 제거 지연 파라미터들을 갖는 픽처 타이밍 SEI 메시지를 전송한다.

[0013]

공통 디코딩 유닛 CPB 제거 지연 파라미터는 CPB로부터 픽처 타이밍 SEI 메시지와 연관된 액세스 유닛에서의 현재의 디코딩 유닛을 제거하기 전에, 바로 선행하는 디코딩 유닛의 CPB로부터의 제거 후에 대기하기 위한 서브-

픽처 클록 틱(clock tick)들의 양을 특정할 수도 있다.

- [0014] 또한, 디코딩 유닛이 액세스 유닛에서의 최초 디코딩 유닛일 때, 공통 디코딩 유닛 CPB 제거 지연 파라미터는 CPB로부터 픽처 타이밍 SEI 메시지와 연관된 액세스 유닛에서의 최초 디코딩 유닛을 제거하기 전에, 선행하는 액세스 유닛에서의 가장 최근의 버퍼링 주기(buffering period) SEI 메시지와 연관된 액세스 유닛에서의 최후 디코딩 유닛의 CPB로부터의 제거 이후에 대기하기 위한 서브-픽처 클록 틱들의 양을 특정할 수도 있다.
- [0015] 대조적으로, 디코딩 유닛이 액세스 유닛에서의 최초가 아닌(non-first) 디코딩 유닛일 때, 공통 디코딩 유닛 CPB 제거 지연 파라미터는 CPB로부터 픽처 타이밍 SEI 메시지와 연관된 액세스 유닛에서의 현재의 디코딩 유닛을 제거하기 전에, 픽처 타이밍 SEI 메시지와 연관된 액세스 유닛에서의 선행하는 디코딩 유닛의 CPB로부터의 제거 후에 대기하기 위한 서브-픽처 클록 틱들의 양을 특정할 수도 있다.
- [0016] 디코딩 유닛 CPB 제거 지연 파라미터들은 CPB로부터 픽처 타이밍 SEI 메시지와 연관된 액세스 유닛에서의 i번째 디코딩 유닛을 제거하기 전에, 최후 디코딩 유닛의 CPB로부터의 제거 후에 대기하기 위한 서브-픽처 클록 틱들의 양을 특정할 수도 있다.
- [0017] 전자 디바이스는 모듈로(modulo) $2^{(cpb_removal_delay_length_minus1 + 1)}$ 카운터의 나머지(remainder)에 따라 디코딩 유닛 CPB 제거 지연 파라미터들을 계산할 수도 있고, 여기서, $cpb_removal_delay_length_minus1 + 1$ 은 공통 디코딩 유닛 CPB 제거 지연 파라미터의 길이이다.
- [0018] 전자 디바이스는 또한, CPB가 액세스 유닛 레벨 상에서의 동작을 지원할 때, CPB로부터 픽처 타이밍 SEI 메시지와 연관된 액세스 유닛 데이터를 제거하기 전에, 선행하는 액세스 유닛에서의 가장 최근의 버퍼링 주기 SEI 메시지와 연관된 액세스 유닛의 CPB로부터의 제거 후에 얼마나 많은 클록 틱들을 대기할 것인지를 특정하는 CPB 제거 지연 파라미터를 포함하는 픽처 타이밍 SEI 메시지를 생성할 수도 있다.
- [0019] 전자 디바이스는 또한, CPB가 서브-픽처 레벨 또는 액세스 유닛 레벨 상에서의 동작을 지원하는지 여부를 결정할 수도 있다. 이것은 코딩된 픽처 버퍼(CPB)가 픽처 타이밍 플래그의 값에 기반으로 하여 서브-픽처 레벨 상에서의 동작을 지원하는 파라미터들을 제공하는지 여부를 표시하는 픽처 타이밍 플래그를 결정하는 것을 포함할 수도 있다. 픽처 타이밍 플래그는 픽처 타이밍 SEI 메시지 내에 포함될 수도 있다.
- [0020] 공통 디코딩 유닛 CPB 제거 지연 파라미터를 포함할 것인지 여부를 결정하는 것은, 공통 디코딩 유닛 CPB 제거 지연 파라미터가 픽처 타이밍 SEI 메시지 내에 포함되어야 할 때, 공통 디코딩 유닛 CPB 제거 지연 플래그를 1로 설정하는 것을 포함할 수도 있다. 그것은 또한, 공통 디코딩 유닛 CPB 제거 지연 파라미터가 픽처 타이밍 SEI 메시지 내에 포함되지 않아야 할 때, 공통 디코딩 유닛 CPB 제거 지연 플래그를 0으로 설정하는 것을 포함할 수도 있다. 공통 디코딩 유닛 CPB 제거 지연 플래그는 픽처 타이밍 SEI 메시지 내에 포함될 수도 있다.
- [0021] 전자 디바이스는 또한, CPB가 서브-픽처 레벨 상에서의 동작을 지원할 때, 액세스 유닛에서의 각각의 디코딩 유닛에 대한 NAL 유닛들의, 1에 의해 오프셋된 양(amount)을 표시하는 별도의 네트워크 추상화 계층(NAL) 유닛들 관련 파라미터들을 생성할 수도 있다. 대안적으로 또는 추가적으로, 전자 디바이스는 액세스 유닛에서의 각각의 디코딩 유닛에 대해 공통인 NAL 유닛들의, 1에 의해 오프셋된 양을 표시하는 공통 NAL 파라미터를 생성할 수도 있다.
- [0022] 비트스트림을 버퍼링하기 위한 전자 디바이스가 또한 설명된다. 전자 디바이스는 프로세서와, 프로세서와 전자 통신하는 메모리 내에 저장된 명령들을 포함한다. 전자 디바이스는 CPB가 액세스 유닛에 대한 서브-픽처 레벨 상에서 파라미터들을 시그널링하는 것으로 결정한다. 전자 디바이스는 또한, 수신된 픽처 타이밍 보충 강화 정보(SEI) 메시지가 공통 디코딩 유닛 코딩된 픽처 버퍼(CPB) 제거 지연 플래그를 포함할 때, 액세스 유닛에서의 모든 디코딩 유닛들에 적용가능한 공통 디코딩 유닛 CPB 제거 지연 파라미터를 결정한다. 전자 디바이스는 또한, 픽처 타이밍 SEI 메시지가 공통 디코딩 유닛 CPB 제거 지연 플래그를 포함하지 않을 때, 액세스 유닛에서의 각각의 디코딩 유닛에 대한 별도의 디코딩 유닛 CPB 제거 지연 파라미터를 결정한다. 전자 디바이스는 또한, 공통 디코딩 유닛 CPB 제거 지연 파라미터 또는 별도의 디코딩 유닛 CPB 제거 지연 파라미터들을 이용하여 CPB로부터 디코딩 유닛들을 제거한다. 전자 디바이스는 또한, 액세스 유닛에서의 디코딩 유닛들을 디코딩한다.
- [0023] 하나의 구성에서, 전자 디바이스는 픽처 타이밍 플래그가 픽처 타이밍 SEI 메시지에서 설정되는 것으로 결정한다. 전자 디바이스는 또한, 하기에 따라 CPB 제거 지연 파라미터, $cpb_removal_delay$ 를 설정할 수도 있다.

수학식 1

$$\text{cpb_removal_delay} = \frac{\sum_{i=0}^{\text{num_decoding_units_minus1}} \text{du_cpb_removal_delay}[i] * t_{c,sub}}{t_c}$$

[0024]

[0025] 여기서, du_cpb_removal_delay[i]는 디코딩 유닛 CPB 제거 지연 파라미터들이고, t_c는 클록 틱이고, t_{c,sub}는 서브-픽처 클록 틱이고, num_decoding_units_minus1은 1에 의해 오프셋된 액세스 유닛에서의 디코딩 유닛들의 양이고, i는 인덱스이다.

[0026] 대안적으로, 전자 디바이스는 하기 수식을 만족시키도록 CPB 제거 지연 파라미터, cpb_removal_delay 및 du_cpb_removal_delay[num_decoding_units_minus1]를 설정할 수도 있다.

수학식 2

$$-1 \leq \text{cpb_removal_delay} * t_c - \left(\sum_{i=0}^{\text{num_decoding_units_minus1}} \text{du_cpb_removal_delay}[i] * t_{c,sub} \right) \leq 1$$

[0027]

[0028] 여기서, du_cpb_removal_delay[i]는 디코딩 유닛 CPB 제거 지연 파라미터들이고, t_c는 클록 틱이고, t_{c,sub}는 서브-픽처 클록 틱이고, num_decoding_units_minus1은 1에 의해 오프셋된 액세스 유닛에서의 디코딩 유닛들의 양이고, i는 인덱스이다.

[0029] 대안적으로, 전자 디바이스는 cpb_removal_delay * t_c = du_cpb_removal_delay[num_decoding_units_minus1] * t_{c,sub}에 따라 CPB 제거 지연 파라미터, cpb_removal_delay 및 du_cpb_removal_delay[num_decoding_units_minus1]을 설정할 수도 있고, 여기서, du_cpb_removal_delay[num_decoding_units_minus1]은 num_decoding_units_minus1번째 디코딩 유닛에 대한 디코딩 유닛 CPB 제거 지연 파라미터이고, t_c는 클록 틱이고, t_{c,sub}는 서브-픽처 클록 틱이고, num_decoding_units_minus1은 1에 의해 오프셋된 액세스 유닛에서의 디코딩 유닛들의 양이다.

[0030] 하나의 구성에서, 전자 디바이스는 픽처 타이밍 플래그가 픽처 타이밍 SEI 메시지에서 설정되는 것으로 결정한다. 전자 디바이스는 또한, 수식: -1 <= (cpb_removal_delay * t_c - du_cpb_removal_delay[num_decoding_units_minus1] * t_{c,sub}) <= 1을 만족시키도록 CPB 제거 지연 파라미터들, cpb_removal_delay 및 du_cpb_removal_delay[num_decoding_units_minus1]을 설정할 수도 있고, 여기서, du_cpb_removal_delay[num_decoding_units_minus1]은 num_decoding_units_minus1번째 디코딩 유닛에 대한 디코딩 유닛 CPB 제거 지연 파라미터이고, t_c는 클록 틱이고, t_{c,sub}는 서브-픽처 클록 틱이고, num_decoding_units_minus1은 1에 의해 오프셋된 액세스 유닛에서의 디코딩 유닛들의 양이다.

[0031] ClockDiff 변수는 ClockDiff = (num_units_in_tick - (num_units_in_sub_tick * (num_decoding_units_minus1 + 1)) / time_scale)로서 정의될 수도 있고, 여기서, num_units_in_tick은 클록 틱 카운터의 하나의 증분(increment)에 대응하는 주파수 time_scale Hz에서 동작하는 클록의 시간 단위들의 수이고, num_units_in_sub_tick은 서브-픽처 클록 틱 카운터의 하나의 증분에 대응하는 주파수 time_scale Hz에서 동작하는 클록의 시간 단위들의 수이고, num_decoding_units_minus1+1은 액세스 유닛에서의 디코딩 유닛들의 양이고, time_scale은 1 초 내에 통과하는 시간 단위들의 수이다.

[0032] 낮은 지연의 가상 참조 디코더(hypothetical reference decoder; HRD) 플래그(예컨대, low_delay_hrd_flag)가 1로 설정되고, t_{r,n}(m) < t_{af}(m)이고, 픽처 타이밍 플래그가 1로 설정되고, CPB가 서브-픽처 레벨에서 동작하고 있고, ClockDiff가 제로(zero)보다 더 클 때, 디코딩 유닛 m에 대한 제거 시간, t_r(m)은 t_r(m) = t_{r,n}(m) + t_{c,sub} * Ceil((t_{af}(m) - t_{r,n}(m)) / t_{c,sub}) + ClockDiff에 따라 결정되고, 여기서, t_{r,n}(m)은 디코

딩 유닛 m에 대한 명목 제거 시간(nominal removal time)이고, t_{c_sub} 는 서브-픽처 클록 틱이고, $Ceil()$ 은 천장 함수(ceiling function)이고, $t_{af}(m)$ 은 디코딩 유닛 m의 최종 도달 시간이다.

[0033] 낮은 지연의 가상 참조 디코더(HRD) 플래그(예컨대, `low_delay_hrd_flag`)가 1로 설정되고, $t_{r,n}(n) < t_{af}(n)$ 이고, 픽처 타이밍 플래그가 1로 설정되고, CPB가 액세스 유닛 레벨에서 동작하고 있고, `ClockDiff`가 제로(zero)보다 더 클 때, 액세스 유닛 n에 대한 제거 시간, $t_r(n)$ 은 $t_r(n) = t_{r,n}(n) + t_c * Ceil((t_{af}(n) - t_{r,n}(n)) / t_c) - \text{ClockDiff}$ 에 따라 결정되고, 여기서, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $Ceil()$ 은 천장 함수이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이다.

[0034] 낮은 지연의 가상 참조 디코더(HRD) 플래그(예컨대, `low_delay_hrd_flag`)가 1로 설정되고, $t_{r,n}(m) < t_{af}(m)$ 이고, 픽처 타이밍 플래그가 1로 설정되고, CPB가 서브-픽처 레벨에서 동작하고 있을 때, 액세스 유닛의 최후 디코딩 유닛 m에 대한 제거 시간 $t_r(m)$ 은 $t_r(m) = t_{r,n}(m) + \max((t_{c_sub} * Ceil((t_{af}(m) - t_{r,n}(m)) / t_{c_sub})), (t_c * Ceil((t_{af}(n) - t_{r,n}(n)) / t_c)))$ 에 따라 결정되고, 여기서, $t_{r,n}(m)$ 은 최후 디코딩 유닛 m의 명목 제거 시간이고, t_{c_sub} 는 서브-픽처 클록 틱이고, $Ceil()$ 은 천장 함수이고, $t_{af}(m)$ 은 최후 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이다.

[0035] 낮은 지연의 가상 참조 디코더(HRD) 플래그가 1로 설정되고, $t_{r,n}(n) < t_{af}(n)$ 이고, 픽처 타이밍 플래그가 1로 설정되고, CPB가 액세스 유닛 레벨에서 동작하고 있을 때, 액세스 유닛 n에 대한 제거 시간 $t_r(n)$ 은 $t_r(n) = t_{r,n}(n) + \max((t_{c_sub} * Ceil((t_{af}(m) - t_{r,n}(m)) / t_{c_sub})), (t_c * Ceil((t_{af}(n) - t_{r,n}(n)) / t_c)))$ 에 따라 결정되고, 여기서, $t_{r,n}(m)$ 은 최후 디코딩 유닛 n의 명목 제거 시간이고, t_{c_sub} 는 서브-픽처 클록 틱이고, $Ceil()$ 은 천장 함수이고, $t_{af}(m)$ 은 최후 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이다.

[0036] 낮은 지연의 가상 참조 디코더(HRD) 플래그(예컨대, `low_delay_hrd_flag`)가 1로 설정되고, $t_{r,n}(m) < t_{af}(m)$ 이고, 픽처 타이밍 플래그가 1로 설정되고, CPB가 서브-픽처 레벨에서 동작하고 있을 때, 액세스 유닛의 최후 디코딩 유닛 m에 대한 제거 시간 $t_r(m)$ 은 $t_r(m) = t_{r,n}(m) + \min((t_{c_sub} * Ceil((t_{af}(m) - t_{r,n}(m)) / t_{c_sub})), (t_c * Ceil((t_{af}(n) - t_{r,n}(n)) / t_c)))$ 에 따라 결정되고, 여기서, $t_{r,n}(m)$ 은 최후 디코딩 유닛 m의 명목 제거 시간이고, t_{c_sub} 는 서브-픽처 클록 틱이고, $Ceil()$ 은 천장 함수이고, $t_{af}(m)$ 은 최후 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이다.

[0037] 낮은 지연의 가상 참조 디코더(HRD) 플래그가 1로 설정되고, $t_{r,n}(n) < t_{af}(n)$ 이고, 픽처 타이밍 플래그가 1로 설정되고, CPB가 액세스 유닛 레벨에서 동작하고 있을 때, 액세스 유닛 n에 대한 제거 시간 $t_r(n)$ 은 $t_r(n) = t_{r,n}(n) + \min((t_{c_sub} * Ceil((t_{af}(m) - t_{r,n}(m)) / t_{c_sub})), (t_c * Ceil((t_{af}(n) - t_{r,n}(n)) / t_c)))$ 에 따라 결정되고, 여기서, $t_{r,n}(m)$ 은 최후 디코딩 유닛 n의 명목 제거 시간이고, t_{c_sub} 는 서브-픽처 클록 틱이고, $Ceil()$ 은 천장 함수이고, $t_{af}(m)$ 은 최후 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이다.

[0038] 낮은 지연의 가상 참조 디코더(HRD) 플래그(예컨대, `low_delay_hrd_flag`)가 1로 설정되고, $t_{r,n}(m) < t_{af}(m)$ 이고, 픽처 타이밍 플래그가 1로 설정되고, CPB가 서브-픽처 레벨에서 동작하고 있을 때, 액세스 유닛의 최후 디코딩 유닛 m에 대한 제거 시간 $t_r(m)$ 은 $t_r(m) = t_{r,n}(m) + (t_c * Ceil((t_{af}(n) - t_{r,n}(n)) / t_c))$ 에 따라 결정되고, 여기서, $t_{r,n}(m)$ 은 최후 디코딩 유닛 m의 명목 제거 시간이고, t_{c_sub} 는 서브-픽처 클록 틱이고, $Ceil()$ 은 천장 함수이고, $t_{af}(m)$ 은 최후 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이다.

- [0039] 낮은 지연의 가상 참조 디코더(HRD) 플래그가 1로 설정되고, $t_{r,n}(n) < t_{af}(n)$ 이고, 픽처 타이밍 플래그가 1로 설정되고, CPB가 액세스 유닛 레벨에서 동작하고 있을 때, 액세스 유닛 n에 대한 제거 시간 $t_r(n)$ 은 $t_r(n) = t_{r,n}(n) + (t_c * \text{Ceil}((t_{af}(n) - t_{r,n}(n)) / t_c))$ 에 따라 결정되고, 여기서, $t_{r,n}(m)$ 은 최후 디코딩 유닛 n의 명목 제거 시간이고, t_{c_sub} 는 서브-픽처 클록 틱이고, $\text{Ceil}()$ 은 천장 함수이고, $t_{af}(m)$ 은 최후 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이다.
- [0040] 낮은 지연의 가상 참조 디코더(HRD) 플래그(예컨대, `low_delay_hrd_flag`)가 1로 설정되고, $t_{r,n}(m) < t_{af}(m)$ 이고, 픽처 타이밍 플래그가 1로 설정되고, CPB가 서브-픽처 레벨에서 동작하고 있을 때, 최후 디코딩 유닛이 아닌 디코딩 유닛 m에 대한 제거 시간은 $t_r(m) = t_{af}(m)$ 로서 설정되고, 여기서, $t_{af}(m)$ 은 디코딩 유닛 m의 최종 도달 시간이다. 낮은 지연의 가상 참조 디코더(HRD) 플래그(예컨대, `low_delay_hrd_flag`)가 1로 설정되고, $t_{r,n}(m) < t_{af}(m)$ 이고, 픽처 타이밍 플래그가 1로 설정되고, CPB가 서브-픽처 레벨에서 동작하고 있을 때, 액세스 유닛의 최후 디코딩 유닛 m인 디코딩 유닛 m에 대한 제거 시간 $t_r(m)$ 은 $t_r(m) = t_{r,n}(m) + (t_{c_sub} * \text{Ceil}((t_{af}(m) - t_{r,n}(m)) / t_{c_sub}))$ 에 따라 결정되고, 여기서, $t_{r,n}(m)$ 은 최후 디코딩 유닛 m의 명목 제거 시간이고, t_{c_sub} 는 서브-픽처 클록 틱이고, $\text{Ceil}()$ 은 천장 함수이고, $t_{af}(m)$ 은 최후 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이고, $t_{af}(m)$ 은 액세스 유닛 n에서의 최후 디코딩 유닛 m의 최종 도달 시간이다.
- [0041] 낮은 지연의 가상 참조 디코더(HRD) 플래그(예컨대, `low_delay_hrd_flag`)가 1로 설정되고, $t_{r,n}(m) < t_{af}(m)$ 이고, 픽처 타이밍 플래그가 1로 설정되고, CPB가 서브-픽처 레벨에서 동작하고 있을 때, 최후 디코딩 유닛이 아닌 디코딩 유닛 m에 대한 제거 시간은 $t_r(m) = t_{af}(m)$ 로서 설정되고, 여기서, $t_{af}(m)$ 은 디코딩 유닛 m의 최종 도달 시간이다. 낮은 지연의 가상 참조 디코더(HRD) 플래그(예컨대, `low_delay_hrd_flag`)가 1로 설정되고, $t_{r,n}(m) < t_{af}(m)$ 이고, 픽처 타이밍 플래그가 1로 설정되고, CPB가 서브-픽처 레벨에서 동작하고 있을 때, 액세스 유닛의 최후 디코딩 유닛 m인 디코딩 유닛 m에 대한 제거 시간 $t_r(m)$ 은 $t_r(m) = t_{r,n}(m) + (t_c * \text{Ceil}((t_{af}(m) - t_{r,n}(m)) / t_c))$ 에 따라 결정되고, 여기서, $t_{r,n}(m)$ 은 최후 디코딩 유닛 m의 명목 제거 시간이고, t_{c_sub} 는 서브-픽처 클록 틱이고, $\text{Ceil}()$ 은 천장 함수이고, $t_{af}(m)$ 은 최후 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이고, $t_{af}(m)$ 은 액세스 유닛 n에서의 최후 디코딩 유닛 m의 최종 도달 시간이다.
- [0042] 낮은 지연의 가상 참조 디코더(HRD) 플래그(예컨대, `low_delay_hrd_flag`)가 1로 설정되고, $t_{r,n}(m) < t_{af}(m)$ 이고, 픽처 타이밍 플래그가 1로 설정되고, CPB가 서브-픽처 레벨에서 동작하고 있을 때, 디코딩 유닛 m에 대한 제거 시간은 $t_r(m) = t_{af}(m)$ 로서 설정되고, 여기서, $t_{r,n}(m)$ 은 디코딩 유닛 m의 명목 제거 시간이고, t_{c_sub} 는 서브-픽처 클록 틱이고, $\text{Ceil}()$ 은 천장 함수이고, $t_{af}(m)$ 은 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이고, $t_{af}(m)$ 은 액세스 유닛 n에서의 디코딩 유닛 m의 최종 도달 시간이다.
- [0043] 낮은 지연의 가상 참조 디코더(HRD) 플래그가 1로 설정되고, $t_{r,n}(n) < t_{af}(n)$ 이고, 픽처 타이밍 플래그가 1로 설정되고, CPB가 액세스 유닛 레벨에서 동작하고 있을 때, 액세스 유닛 n에 대한 제거 시간 $t_r(n)$ 은 $t_r(n) = t_{af}(n)$ 에 따라 결정되고, 여기서, $t_{r,n}(m)$ 은 최후 디코딩 유닛 n의 명목 제거 시간이고, t_{c_sub} 는 서브-픽처 클록 틱이고, $\text{Ceil}()$ 은 천장 함수이고, $t_{af}(m)$ 은 최후 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이다.
- [0044] 추가적으로, 일부의 경우에는, 상기 대안적인 수학적식들 중의 어느 것이 디코딩 유닛들의 제거 시간 및 액세스 유닛의 제거 시간을 판정하기 위하여 이용되는지를 시그널링하기 위하여, 플래그가 비트스트림의 일부로 전송될 수도 있다. 하나의 경우에는, 플래그가 `du_au_cpb_alignment_mode_flag`로 칭해질 수도 있다.

du_au_cpb_alignment_mode_flag가 1일 경우, 서브-픽처 기반 모드에서 동작하는 CPB의 동작을 액세스 유닛 모드에서 동작하는 CPB와 정렬하는 상기 수학식들이 이용된다. du_au_cpb_alignment_mode_flag가 0일 경우, 서브-픽처 기반 모드에서 동작하는 CPB의 동작을 액세스 유닛 모드에서 동작하는 CPB와 정렬하지 않는 상기 수학식들이 이용된다.

[0045] 하나의 경우에는, 플래그 du_au_cpb_alignment_mode_flag가 비디오 유용성 정보(video usability information: VUI)에서 시그널링될 수도 있다. 또 다른 경우에는, 플래그 du_au_cpb_alignment_mode_flag가 픽처 타이밍 SEI 메시지로 전송될 수도 있다. 또 다른 경우에는, 플래그 du_au_cpb_alignment_mode_flag가 비트스트림의 일부의 다른 규범 부분에서 전송될 수도 있다. 본원에서 개시된 시스템들 및 방법들에 따른 수정된 선택스 및 시맨틱(semantic)들의 하나의 예는 다음과 같이 표 (0)에서 주어진다.

표 0

pic_timing(payloadSize) {
if(CpbDpbDelaysPresentFlag) {
cpb_removal_delay
dpb_output_delay
if(sub_pic_cpb_params_present_flag) {
num_decoding_units_minus1
du_au_cpb_alignment_mode_flag
for(i = 0; i <= num_decoding_units_minus1; i++) {
num_nalus_in_du_minus1[i]
du_cpb_removal_delay[i]
}
}
}
}

표(0)

[0046]

[0047] 다양한 변수들에 대해 위에서 이용된 것들과는 상이한 심볼(symbol)들(명칭들)이 이용될 수도 있다는 것에 주목해야 한다. 예를 들어, 액세스 유닛 n의 $t_r(n)$ 은 CpbRemovalTime(n)으로 칭해질 수도 있고, 디코딩 유닛 n의 $t_r(m)$ 은 CpbRemovalTime(m)으로 칭해질 수도 있고, t_{c_sub} 는 ClockSubTick로 칭해질 수도 있고, t_c 는 ClockTick으로 칭해질 수도 있고, 액세스 유닛 m의 $t_{af}(n)$ 은 액세스 유닛 n의 FinalArrivalTime(n)으로 칭해질 수도 있고, 디코딩 유닛 m의 $t_{af}(m)$ 은 FinalArrivalTime(m)으로 칭해질 수도 있고, $t_{r,n}(n)$ 은 액세스 유닛 n의 NominalRemovalTime(n)으로 칭해질 수도 있고, $t_{r,n}(m)$ 은 디코딩 유닛 m의 NominalRemovalTime(m)으로 칭해질 수도 있다.

[0048] 전자 디바이스에 의해 메시지를 전송하기 위한 방법이 또한 설명된다. 방법은, 코딩된 픽처 버퍼(CPB)가 서브-픽처 레벨 상에서의 동작을 지원할 때, 픽처 타이밍 보충 강화 정보(Supplemental Enhancement Information; SEI) 메시지 내에 공통 디코딩 유닛 CPB 제거 지연 파라미터를 포함할 것인지 여부를 결정하는 것을 포함한다. 방법은 또한, 공통 디코딩 유닛 CPB 제거 지연 파라미터가 픽처 타이밍 SEI 메시지 내에 포함되어야 할 때, 공통 디코딩 유닛 CPB 제거 지연 파라미터를 생성하는 것을 포함하고, 여기서, 공통 디코딩 유닛 CPB 제거 지연 파라미터는 CPB로부터의 액세스 유닛에서의 모든 디코딩 유닛들에 적용가능하다. 방법은 또한, 공통 디코딩 유닛 CPB 제거 지연 파라미터가 픽처 타이밍 SEI 메시지 내에 포함되지 않아야 할 때, 액세스 유닛에서의 각각의 디코딩 유닛에 대한 별도의 디코딩 유닛 CPB 제거 지연 파라미터를 생성하는 것을 포함한다. 방법은 또한, 공통 디코딩 유닛 CPB 제거 지연 파라미터 또는 디코딩 유닛 CPB 제거 지연 파라미터들을 갖는 픽처 타이밍 SEI 메시지를 전송하는 것을 포함한다.

[0049] 전자 디바이스에 의해 비트스트림을 버퍼링하기 위한 방법이 또한 설명된다. 방법은 CPB가 액세스 유닛에 대한 서브-픽처 레벨 상에서 파라미터들을 시그널링하는 것으로 결정하는 것을 포함한다. 방법은 또한, 수신된 픽처 타이밍 보충 강화 정보(SEI) 메시지가 공통 디코딩 유닛 코딩된 픽처 버퍼(CPB) 제거 지연 플래그를 포함할 때, 액세스 유닛에서의 모든 디코딩 유닛들에 적용가능한 공통 디코딩 유닛 CPB 제거 지연 파라미터를 결정하는 것을 포함한다. 방법은 또한, 픽처 타이밍 SEI 메시지가 공통 디코딩 유닛 CPB 제거 지연 플래그를 포함하지 않을 때, 액세스 유닛에서의 각각의 디코딩 유닛에 대한 별도의 디코딩 유닛 CPB 제거 지연 파라미터를 결정하는

것을 포함한다. 방법은 또한, 공통 디코딩 유닛 CPB 제거 지연 파라미터 또는 별도의 디코딩 유닛 CPB 제거 지연 파라미터들을 이용하여 CPB로부터 디코딩 유닛들을 제거하는 것을 포함한다. 방법은 또한, 액세스 유닛에서의 디코딩 유닛들을 디코딩하는 것을 포함한다.

[0050] 본원에서 개시된 시스템들 및 방법들은 메시지를 전송하고 비트스트림을 버퍼링하기 위한 전자 디바이스들을 설명한다. 예를 들어, 본원에서 개시된 시스템들 및 방법들은 서브-픽처 파라미터들에서 시작하는 비트스트림들에 대한 버퍼링을 설명한다. 일부의 구성들에서는, 본원에서 개시된 시스템들 및 방법들이 서브-픽처 기반 가상 참조 디코더(HRD) 파라미터들을 시그널링하는 것을 설명할 수도 있다. 예를 들어, 본원에서 개시된 시스템들 및 방법들은 픽처 타이밍 보충 강화 정보(SEI) 메시지에 대한 수정을 설명한다. 본원에서 개시된 시스템들 및 방법들(예컨대, HRD 수정)은, 각각의 서브-픽처가 도달하고 규칙적인 간격들로 CPB로부터 제거될 때에 파라미터들의 더욱 간결한 시그널링으로 귀착될 수도 있다.

[0051] 또한, 서브-픽처 레벨 CPB 제거 지연 파라미터들이 존재할 때, 코딩된 픽처 버퍼(CPB)는 액세스 유닛 레벨 또는 서브-픽처 레벨에서 동작할 수도 있다. 본 시스템들 및 방법들은 또한, 서브-픽처 레벨 기반 CPB 동작 및 액세스 유닛 레벨 CPB 동작이 디코딩 유닛 제거의 동일한 타이밍으로 귀착되도록 비트스트림 제약(constraint)을 부과할 수도 있다. 구체적으로, 서브-픽처 모드에서 동작할 때의 액세스 유닛에서의 최후 디코딩 유닛의 제거의 타이밍과, 액세스 유닛 모드에서 동작할 때의 액세스 유닛의 제거의 타이밍은 동일할 것이다.

[0052] 용어 "가상"은 HRD와 관련하여 이용되지만, HRD는 물리적으로 구현될 수도 있다는 것에 주목해야 한다. 예를 들어, "HRD"는 실제적인 디코더의 구현예를 설명하기 위하여 이용될 수도 있다. 일부의 구성들에서, HRD는 비트스트림이 고효율 비디오 코딩 (High Efficiency Video Coding; HEVC) 사양들을 준수하는지 여부를 결정하기 위하여 구현될 수도 있다. 예를 들어, HRD는 타입 I 비트스트림들 및 타입 II 비트스트림들이 HEVC 사양들을 준수하는지 여부를 결정하기 위하여 이용될 수도 있다. 타입 I 비트스트림은 비디오 코딩 계층(Video Coding Layer; VCL) 네트워크 액세스 계층(Network Access Layer; NAL) 유닛들 및 필터 데이터 NAL 유닛들만을 포함할 수도 있다. 타입 II 비트스트림은 추가적인 다른 NAL 유닛들 및 신택스 엘리먼트들을 포함할 수도 있다.

[0053] 비디오 코딩에 관한 합동 협력 팀(Joint Collaborative Team on Video Coding; JCTVC) 문서 JCTVC-I0333은 서브-픽처 기반 HRD를 포함하고, 픽처 타이밍 SEI 메시지들을 지원한다. 이 기능성은, 그 전체적으로 참조를 위해 본원에 편입된 고효율 비디오 코딩(HEVC) 위원회 초안(JCTVC-I1003) 내로 편입되었다. B. Bros, W-J. Han, J-R. Ohm, G.J. Sullivan, Wang, 및 T-. Wiegand, "High efficiency video coding (HEVC) text specification draft 10 (for DFIS & Last Call)", JCTVC-J10003_v34, Geneva, January 2013은 이것에 의하여, 그 전체적으로 참조를 위해 본원에 편입된다. B. Bros, W-J. Han, J-R. Ohm, G.J. Sullivan, Wang, 및 T-. Wiegand, "High efficiency video coding (HEVC) text specification draft 10", JCTVC-L1003, Geneva, January 2013은 이것에 의하여, 그 전체적으로 참조를 위해 본원에 편입된다.

[0054] 본원에서 개시된 시스템들 및 방법들에 따른 수정된 신택스 및 시맨틱들의 하나의 예는 다음과 같이 표 (1)에서 주어진다.

표 1

pic_timing(payloadSize){
if(CpbDpbDelaysPresentFlag){
cpb_removal_delay
dpb_output_delay
if(sub_pic_cpb_params_present_flag){
num_decoding_units_minus1
common_du_cpb_removal_delay_flag
if(common_du_cpb_removal_delay_flag){
common_du_cpb_removal_delay
}
for(i=0; i<=num_decoding_units_minus1; i++){
num_nalus_in_du_minus1[i]
if(!common_du_cpb_removal_delay_flag)
du_cpb_removal_delay[i]
}
}
}
}

표 (1)

[0055]

- [0056] 본원에서 개시된 시스템들 및 방법들에 따른 버퍼링 주기 SEI 메시지 시맨틱들에 관한 예들은 다음과 같이 주어진다. 특히, 수정된 선택스 엘리먼트들의 시맨틱들에 관한 추가적인 세부사항은 다음과 같이 주어진다. NalHrdBpPresentFlag 또는 VclHrdBpPresentFlag가 1과 동일할 때, 버퍼링 주기 SEI 메시지는 비트스트림에서의 임의의 액세스 유닛과 연관될 수 있고, 버퍼링 주기 SEI 메시지는 각각의 IDR 액세스 유닛, 각각의 CRA 액세스 유닛, 및 복구 포인트 SEI 메시지와 연관된 각각의 액세스 유닛과 연관될 수도 있다. 일부의 애플리케이션들에 대해서는, 버퍼링 주기 SEI 메시지의 빈번한 존재가 바람직할 수도 있다. 버퍼링 주기는 디코딩 순서에서 버퍼링 주기 SEI 메시지의 2 개의 인스턴스(instance)들 사이의 액세스 유닛들의 세트로서 특정된다.
- [0057] 'seq_parameter_set_id'는 시퀀스 HRD 속성들을 포함하는 시퀀스 파라미터 세트를 특정한다. seq_parameter_set_id의 값은 버퍼링 주기 SEI 메시지와 연관된 1차 코딩된 픽처에 의해 참조된 픽처 파라미터 세트에서의 seq_parameter_set_id의 값과 동일할 수도 있다. seq_parameter_set_id의 값은 0 내지 31까지의 범위에 있을 수도 있다.
- [0058] 'initial_cpb_removal_delay'[SchedSelIdx]는 HRD 초기화 후의 최초 버퍼링 주기에 대하여, 버퍼링 주기 SEI 메시지와 연관된 액세스 유닛과 연관된 코딩된 데이터의 최초 비트의 CPB에서의 도달 시간과, 동일한 액세스 유닛과 연관된 코딩된 데이터의 CPB로부터의 제거 시간 사이의 SchedSelIdx번째 CPB에 대한 지연을 특정한다. 선택스 엘리먼트는 initial_cpb_removal_delay_length_minus1 + 1에 의해 주어진 비트들에서의 길이를 가진다. 그것은 90 kHz 클록의 단위이다. initial_cpb_removal_delay[SchedSelIdx]는 0과 동일하지 않을 수도 있고, 90 kHz 클록 단위들에서 CPB 사이즈(size)와 시간-동등한, $90000 * (\text{CpbSize}[\text{ SchedSelIdx }] / \text{BitRate}[\text{ SchedSelIdx }])$ 를 초과하지 않을 수도 있다.
- [0059] 'initial_cpb_removal_delay_offset'[SchedSelIdx]는, 코딩된 액세스 유닛들의 초기 전달 시간을 CPB에 특정하기 위하여 cpb_removal_delay와 조합하여 SchedSelIdx번째 CPB에 대해 이용된다. initial_cpb_removal_delay_offset[SchedSelIdx]는 90 kHz 클록의 단위이다. initial_cpb_removal_delay_offset[SchedSelIdx] 선택스 엘리먼트는, 비트들에서의 그 길이가 initial_cpb_removal_delay_length_minus1 + 1에 의해 주어지는 고정된 길이 코드이다. 이 선택스 엘리먼트는 디코더들에 의해 이용되지 않고, (예컨대, JCTVC-I1003의 부록 C에서 특정된 바와 같이) 전달 스케줄러(HSS)에 대해서만 필요하게 된다.
- [0060] 전체의 코딩된 비디오 시퀀스 상에서, initial_cpb_removal_delay[SchedSelIdx] 및 initial_cpb_removal_delay_offset[SchedSelIdx]의 합은 SchedSelIdx의 각각의 값에 대하여 일정할 수도 있다.
- [0061] 'initial_du_cpb_removal_delay'[SchedSelIdx]는 HRD 초기화 후의 최초 버퍼링 주기에 대하여, 버퍼링 주기 SEI 메시지와 연관된 액세스 유닛에서의 최초 디코딩 유닛과 연관된 코딩된 데이터의 최초 비트의 CPB에서의 도달 시간과, 동일한 디코딩 유닛과 연관된 코딩된 데이터의 CPB로부터의 제거 시간 사이의 SchedSelIdx번째 CPB에 대한 지연을 특정한다. 선택스 엘리먼트는 initial_cpb_removal_delay_length_minus1 + 1에 의해 주어진 비트들에서의 길이를 가진다. 그것은 90 kHz 클록의 단위이다. initial_du_cpb_removal_delay[SchedSelIdx]는 0과 동일하지 않을 수도 있고, 90 kHz 클록 단위들에서 CPB 사이즈(size)와 시간-동등한, $90000 * (\text{CpbSize}[\text{ SchedSelIdx }] / \text{BitRate}[\text{ SchedSelIdx }])$ 를 초과하지 않을 수도 있다.
- [0062] 'initial_du_cpb_removal_delay_offset'[SchedSelIdx]는, 디코딩 유닛들의 초기 전달 시간을 CPB에 특정하기 위하여 cpb_removal_delay와 조합하여 SchedSelIdx번째 CPB에 대해 이용된다.
- [0063] initial_cpb_removal_delay_offset[SchedSelIdx]는 90 kHz 클록의 단위이다. initial_du_cpb_removal_delay_offset[SchedSelIdx] 선택스 엘리먼트는, 비트들에서의 그 길이가 initial_cpb_removal_delay_length_minus1 + 1에 의해 주어지는 고정된 길이 코드이다. 이 선택스 엘리먼트는 디코더들에 의해 이용되지 않고, (예컨대, JCTVC-I1003의 부록 C에서 특정된 바와 같이) 전달 스케줄러(HSS)에 대해서만 필요하게 된다.
- [0064] 전체의 코딩된 비디오 시퀀스 상에서, initial_du_cpb_removal_delay[SchedSelIdx] 및 initial_du_cpb_removal_delay_offset[SchedSelIdx]의 합은 SchedSelIdx의 각각의 값에 대하여 일정할 수도 있다.

- [0065] 본원에서 개시된 시스템들 및 방법들에 따른 픽처 타이밍 SEI 메시지 시맨틱들에 관한 예들은 다음과 같이 주어진다. 특히, 수정된 선택스 엘리먼트들의 시맨틱들에 관한 추가적인 세부사항은 다음과 같이 주어진다.
- [0066] 픽처 타이밍 SEI 메시지의 선택스는 픽처 타이밍 SEI 메시지와 연관된 코딩된 픽처에 대해 활성화된 시퀀스 파라미터 세트의 콘텐츠(content)에 종속적이다. 그러나, 동일한 액세스 유닛 내의 버퍼링 주기 SEI 메시지가 순간적 디코딩 리프레시(Instantaneous Decoding Refresh; IDR) 액세스 유닛의 픽처 타이밍 SEI 메시지를 선행하지 않으면, 연관된 시퀀스 파라미터 세트의 활성화(그리고, 비트스트림에서 최초 픽처가 아닌 IDR 픽처들에 대하여, 코딩된 픽처가 IDR 픽처라는 결정)는 코딩된 픽처의 최초 코딩된 슬라이스 네트워크 추상화 계층(NAL) 유닛의 디코딩까지는 발생하지 않는다. 코딩된 픽처의 코딩된 슬라이스 NAL 유닛은 NAL 유닛 순서에서 픽처 타이밍 SEI 메시지를 후행하므로, 디코더가 코딩된 픽처에 대해 활성화된 시퀀스 파라미터의 파라미터들을 결정할 때까지 픽처 타이밍 SEI 메시지를 포함하는 원시 바이트 시퀀스 페이로드(raw byte sequence payload; RBSP)를 저장하고, 다음으로, 픽처 타이밍 SEI 메시지의 파싱(parsing)을 수행하는 것이 필요한 경우들이 있을 수도 있다.
- [0067] 비트스트림에서의 픽처 타이밍 SEI 메시지의 존재는 다음과 같이 특정된다. CpbDpbDelaysPresentFlag가 1과 동일할 경우, 하나의 픽처 타이밍 SEI 메시지는 코딩된 비디오 시퀀스의 매 액세스 유닛에서 존재할 수도 있다. 이와 다를 경우(CpbDpbDelaysPresentFlag가 0과 동일함), 픽처 타이밍 SEI 메시지들은 코딩된 비디오 시퀀스의 임의의 액세스 유닛에서 존재하지 않을 수도 있다.
- [0068] 'cpb_removal_delay'는, 픽처 타이밍 SEI 메시지와 연관된 액세스 유닛 데이터를 버퍼로부터 제거하기 전에, 선행하는 액세스 유닛에서의 가장 최근의 버퍼링 주기 SEI 메시지와 연관된 액세스 유닛의 CPB로부터의 제거 후에 얼마나 많은 클럭 틱들(JCTVC-I1003의 하위조항 E.2.1 참조)을 대기할 것인지를 특정한다. 이 값은 또한, JCTVC-I1003의 부록 C 에서 특정된 바와 같이, HSS에 대한 CPB로의 액세스 유닛 데이터의 도달의 가장 이른 가능한 시간을 계산하기 위하여 이용된다. 선택스 엘리먼트는 그 비트들에서의 길이가 $cpb_removal_delay_length_minus1 + 1$ 에 의해 주어지는 고정된 길이 코드이다. $cpb_removal_delay$ 는 모듈로 $2^{(cpb_removal_delay_length_minus1 + 1)}$ 카운터의 나머지이다.
- [0069] $cpb_removal_delay$ 는, 상이한 코딩된 비디오 시퀀스의 액세스 유닛일 수도 있는, 버퍼링 주기 SEI 메시지를 포함하는 선행하는 액세스 유닛의 제거 시간에 관련된 클럭 틱들의 수를 특정하지만, 선택스 엘리먼트 $cpb_removal_delay$ 의 (비트들에서의) 길이를 결정하는 $cpb_removal_delay_length_minus1$ 의 값은 픽처 타이밍 SEI 메시지와 연관된 1차 코딩된 픽처에 대해 활성화된 시퀀스 파라미터 세트에서 코딩된 $cpb_removal_delay_length_minus1$ 의 값이다.
- [0070] 'dpb_output_delay'는 픽처의 디코딩된 픽처 버퍼(DPB) 출력 시간을 연산하기 위하여 이용된다. 그것은 디코딩된 픽처가 DPB 로부터 출력되기 전에, CPB로부터의 액세스 유닛에서의 최후 디코딩 유닛의 제거 후에 얼마나 많은 클럭 틱들을 대기할 것인지를 특정한다(JCTVC-I1003의 하위조항 C.2 참조).
- [0071] DPB에 대하여, 픽처가 "단기 참조(short-term reference)를 위해 이용됨" 또는 "장기 참조(long-term reference)를 위해 이용됨"으로서 여전히 표기될 때, 픽처는 그 출력 시간에 DPB로부터 제거되지 않는다. 오직 하나의 dpb_output_delay 가 디코딩된 픽처에 대해 특정된다. 선택스 엘리먼트 dpb_output_delay 의 길이는 $dpb_output_delay_length_minus1 + 1$ 에 의해 비트들로 주어진다. $max_dec_pic_buffering[max_temporal_layers_minus1]$ 이 0과 동일할 때, dpb_output_delay 는 0과 동일할 수도 있다.
- [0072] JCTVC-I1003의 하위조항 C.2에서 특정된 바와 같이 출력 타이밍 준수 디코더로부터 출력되는 임의의 픽처의 dpb_output_delay 로부터 유도된 출력 시간은 디코딩 순서에서 임의의 후속 코딩된 비디오 시퀀스에서의 모든 픽처들의 dpb_output_delay 로부터 유도된 출력 시간을 선행할 수도 있다. 이 선택스 엘리먼트의 값들에 의해 확립된 픽처 출력 순서는 하위조항에 의해 특정된 바와 같이, $PicOrderCnt()$ 의 값들에 의해 확립된 것과 동일한 순서일 수도 있다. 하위조항의 "범핑(bumping)" 프로세스에 의해 출력되지 않은 픽처들에 대하여, 이들이 디코딩 순서에서, 1 과 동일하거나 1 과 동일한 것으로 추론된 $no_output_of_prior_pics_flag$ 를 갖는 IDR 픽처를 선행하기 때문에, dpb_output_delay 로부터 유도된 출력 시간들은 동일한 코딩된 비디오 시퀀스 내의 모든 픽처들에 관련된 $PicOrderCnt()$ 의 증가하는 값과 함께 증가하고 있을 수도 있다.
- [0073] 'num_decoding_units_minus1' 플러스(plus) 1은 픽처 타이밍 SEI 메시지가 연관되는 액세스 유닛에서의 디코딩 유닛들의 수를 특정한다. $num_decoding_units_minus1$ 의 값은 0 내지 $PicWidthInCtbs * PicHeightInCtbs - 1$ 까지의 범위에 있을 수도 있다.

- [0074] 1과 동일한 'common_du_cpb_removal_delay_flag'는 선택스 엘리먼트 common_du_cpb_removal_delay가 존재하는 것으로 특징한다.
- [0075] 0과 동일한 'common_du_cpb_removal_delay_flag'는 선택스 엘리먼트 common_du_cpb_removal_delay가 존재하지 않는 것으로 특징한다.
- [0076] 'common_du_cpb_removal_delay'는 다음과 같이 정보를 특징한다: 디코딩 유닛이 픽처 타이밍 SEI 메시지와 연관된 액세스 유닛에서의 최초 디코딩 유닛일 경우, common_du_cpb_removal_delay는 CPB로부터 픽처 타이밍 SEI 메시지와 연관된 액세스 유닛에서의 최초 디코딩 유닛을 제거하기 전에, 선행하는 액세스 유닛에서의 가장 최근의 버퍼링 주기 SEI 메시지와 연관된 액세스 유닛에서의 최후 디코딩 유닛의 CPB로부터의 제거 후에 얼마나 많은 서브-픽처 클록 틱들(JCTVC-I1003의 하위조항 E.2.1 참조)을 대기할 것인지를 특징한다.
- [0077] 이와 다르게, common_du_cpb_removal_delay는 CPB로부터 픽처 타이밍 SEI 메시지와 연관된 액세스 유닛에서의 현재의 디코딩 유닛을 제거하기 전에, 픽처 타이밍 SEI 메시지와 연관된 액세스 유닛에서의 선행하는 디코딩 유닛의 CPB로부터의 제거 후에 얼마나 많은 서브-픽처 클록 틱들(JCTVC-I1003의 하위조항 E.2.1 참조)을 대기할 것인지를 특징한다. 이 값은 또한, 부록 C에서 특정된 바와 같이, HSS에 대한 CPB로의 디코딩 유닛 데이터의 도달의 가장 이른 가능한 시간을 계산하기 위하여 이용된다. 선택스 엘리먼트는 그 비트들에서의 길이가 $cpb_removal_delay_length_minus1 + 1$ 에 의해 주어지는 고정된 길이 코드이다. common_du_cpb_removal_delay는 모듈로 $2^{(cpb_removal_delay_length_minus1 + 1)}$ 카운터의 나머지이다.
- [0078] 'common_du_cpb_removal_delay'를 특징하기 위한 대안적인 방법은 다음과 같다.
- [0079] common_du_cpb_removal_delay는 CPB로부터 픽처 타이밍 SEI 메시지와 연관된 액세스 유닛에서의 현재의 디코딩 유닛을 제거하기 전에, 최후 디코딩 유닛의 CPB로부터의 제거 후에 얼마나 많은 서브-픽처 클록 틱들(JCTVC-I1003의 하위조항 E.2.1 참조)을 대기할 것인지를 특징한다. 이 값은 또한, 부록 C에서 특정된 바와 같이, HSS에 대한 CPB로의 디코딩 유닛 데이터의 도달의 가장 이른 가능한 시간을 계산하기 위하여 이용된다. 선택스 엘리먼트는 그 비트들에서의 길이가 $cpb_removal_delay_length_minus1 + 1$ 에 의해 주어지는 고정된 길이 코드이다. common_du_cpb_removal_delay는 모듈로 $2^{(cpb_removal_delay_length_minus1 + 1)}$ 카운터의 나머지이다.
- [0080] common_du_cpb_removal_delay는, 상이한 코딩된 비디오 시퀀스의 액세스 유닛일 수도 있는, 버퍼링 주기 SEI 메시지를 포함하는 선행하는 액세스 유닛에서의 최초 디코딩 유닛의 제거 시간에 관련된 서브-픽처 클록 틱들의 수를 특징하지만, 선택스 엘리먼트 common_du_cpb_removal_delay의 (비트들에서의) 길이를 결정하는 $cpb_removal_delay_length_minus1$ 의 값은 픽처 타이밍 SEI 메시지와 연관된 코딩된 픽처에 대해 활성인 시퀀스 파라미터 세트에서 코딩된 $cpb_removal_delay_length_minus1$ 의 값이다.
- [0081] 'num_nalus_in_du_minus1[i]' 플러스 1은 픽처 타이밍 SEI 메시지가 연관되는 액세스 유닛의 i번째 디코딩 유닛에서의 NAL 유닛들의 수를 특징한다. num_nalus_in_du_minus1[i]의 값은 0 내지 $PicWidthInCtbs * PicHeightInCtbs - 1$ 까지의 범위에 있을 수도 있다.
- [0082] 액세스 유닛의 최초 디코딩 유닛은 액세스 유닛에서의 디코딩 순서에서 최초 num_nalus_in_du_minus1[0] + 1 연속적인 NAL 유닛들로 구성된다. 액세스 유닛의 i번째(i는 0 보다 더 큼) 디코딩 유닛은 디코딩 순서에서, 액세스 유닛의 이전의 디코딩 유닛에서의 최후 NAL 유닛을 바로 후행하는 num_nalus_in_du_minus1[i] + 1 연속적인 NAL 유닛들로 구성된다. 각각의 디코딩 유닛에서 적어도 하나의 VCL NAL 유닛이 있을 수도 있다. VCL NAL 유닛과 연관된 모든 비-VCL(non-VCL) NAL 유닛들은 동일한 디코딩 유닛 내에 포함될 수도 있다.
- [0083] 'du_cpb_removal_delay[i]'는 CPB로부터 픽처 타이밍 SEI 메시지와 연관된 액세스 유닛에서의 i번째 디코딩 유닛을 제거하기 전에, 선행하는 액세스 유닛에서의 가장 최근의 버퍼링 주기 SEI 메시지와 연관된 액세스 유닛에서의 최초 디코딩 유닛의 CPB로부터의 제거 후에 얼마나 많은 서브-픽처 클록 틱들(JCTVC-I1003의 하위조항 E.2.1 참조)을 대기할 것인지를 특징한다. 이 값은 또한, (예컨대, JCTVC-I1003의 부록 C에서 특정된 바와 같이) HSS에 대한 CPB로의 디코딩 유닛 데이터의 도달의 가장 이른 가능한 시간을 계산하기 위하여 이용된다. 선택스 엘리먼트는 그 비트들에서의 길이가 $cpb_removal_delay_length_minus1 + 1$ 에 의해 주어지는 고정된 길이 코드이다. du_cpb_removal_delay[i]는 모듈로 $2^{(cpb_removal_delay_length_minus1 + 1)}$ 카운터의 나머지이다.
- [0084] du_cpb_removal_delay[i]는, 상이한 코딩된 비디오 시퀀스의 액세스 유닛일 수도 있는, 버퍼링 주기 SEI 메시지를 포함하는 선행하는 액세스 유닛에서의 최초 디코딩 유닛의 제거 시간에 관련된 서브-픽처 클록 틱들의 수

를 특정하지만, 선택스 엘리먼트 $du_cpb_removal_delay[i]$ 의 (비트들에서의) 길이를 결정하는 $cpb_removal_delay_length_minus1$ 의 값은 픽처 타이밍 SEI 메시지와 연관된 코딩된 픽처에 대해 활성인 시퀀스 파라미터 세트에서 코딩된 $cpb_removal_delay_length_minus1$ 의 값이다.

- [0085] 하나의 구성에서, 디코딩 유닛 제거의 타이밍 및 디코딩 유닛들의 디코딩은 다음과 같이 구현될 수도 있다.
- [0086] $SubPicCpbFlag$ 가 0과 동일할 경우, 변수 $CpbRemovalDelay(m)$ 은 디코딩 유닛 m 인 액세스 유닛과 연관된 픽처 타이밍 SEI 메시지에서의 $cpb_removal_delay$ 의 값으로 설정되고, 변수 T_c 는 t_c 로 설정된다. 이와 다르게, $SubPicCpbFlag$ 가 1과 동일하고 $common_du_cpb_removal_delay_flag$ 가 0과 동일할 경우, 변수 $CpbRemovalDelay(m)$ 은 디코딩 유닛 m 을 포함하는 액세스 유닛과 연관된 픽처 타이밍 SEI 메시지에서의 디코딩 유닛 m (m 은 0으로부터 $num_decoding_units_minus1$ 까지의 범위임)에 대한 $du_cpb_removal_delay[i]$ 의 값으로 설정되고, 변수 T_c 는 t_{c_sub} 로 설정된다.
- [0087] 일부의 경우들에는, 이와 다르게, $SubPicCpbFlag$ 가 1과 동일하고 $common_du_cpb_removal_delay_flag$ 가 0과 동일할 경우, 변수 $CpbRemovalDelay(m)$ 은 디코딩 유닛 m 을 포함하는 액세스 유닛과 연관된 픽처 타이밍 SEI 메시지에서의 디코딩 유닛 m (m 은 0으로부터 $num_decoding_units_minus1$ 까지의 범위임)에 대한 $(m+1)*du_cpb_removal_delay[i]$ 의 값으로 설정되고, 변수 T_c 는 t_{c_sub} 로 설정된다.
- [0088] 이와 다르게, $SubPicCpbFlag$ 가 1과 동일하고 $common_du_cpb_removal_delay_flag$ 가 1일 경우, 변수 $CpbRemovalDelay(m)$ 은 디코딩 유닛 m 을 포함하는 액세스 유닛과 연관된 픽처 타이밍 SEI 메시지에서의 디코딩 유닛 m 에 대한 $common_du_cpb_removal_delay$ 의 값으로 설정되고, 변수 T_c 는 t_{c_sub} 로 설정된다.
- [0089] 디코딩 유닛 m 이 0과 동일한 n 을 갖는 디코딩 유닛(HRD를 초기화하는 액세스 유닛의 최초 디코딩 유닛)일 때, CPB로부터의 디코딩 유닛의 명목 제거 시간은 $t_{r,n}(0) = InitCpbRemovalDelay[SchedSelIdx] / 90000$ 에 의해 특정된다.
- [0090] 디코딩 유닛 m 이 HRD를 초기화하지 않는 버퍼링 주기의 최초 액세스 유닛의 최초 디코딩 유닛일 때, CPB로부터의 디코딩 유닛의 명목 제거 시간은 $t_{r,n}(m) = t_{r,n}(m_b) + T_c * CpbRemovalDelay(m)$ 에 의해 특정되고, 여기서, $t_{r,n}(m_b)$ 은 이전 버퍼링 주기의 최초 디코딩 유닛의 명목 제거 시간이다.
- [0091] 디코딩 유닛 m 이 버퍼링 주기의 최초 디코딩 유닛일 때, m_b 는 디코딩 유닛 m 의 제거 시간 $t_{r,n}(m)$ 에서 m 과 동일하게 설정된다. 버퍼링 주기의 최초 디코딩 유닛이 아닌 디코딩 유닛 m 의 명목 제거 시간 $t_{r,n}(m)$ 은 $t_{r,n}(m) = t_{r,n}(m_b) + T_c * CpbRemovalDelay(m)$ 에 의해 주어지고, 여기서, $t_{r,n}(m_b)$ 는 현재의 버퍼링 주기의 최초 디코딩 유닛의 명목 제거 시간이다.
- [0092] 디코딩 유닛 m 의 제거 시간은 다음과 같이 특정된다. $low_delay_hrd_flag$ 가 0과 동일하거나 $t_{r,n}(m) \geq t_{af}(m)$ 일 경우, 디코딩 유닛 m 의 제거 시간은 $t_r(m) = t_{r,n}(m)$ 에 의해 특정된다. 이와 다를 경우 ($low_delay_hrd_flag$ 가 1과 동일하고 $t_{r,n}(m) < t_{af}(m)$ 임), 디코딩 유닛 m 의 제거 시간은 $t_r(m) = t_{r,n}(m) + T_c * Ceil((t_{af}(m) - t_{r,n}(m)) / T_c)$ 에 의해 특정된다. 후자의 경우($low_delay_hrd_flag$ 가 1과 동일하고 $t_{r,n}(m) < t_{af}(m)$ 임)는, 디코딩 유닛 m 의 사이즈 $b(m)$ 이 너무 커서 그것이 명목 제거 시간에서의 제거를 방지함을 표시한다.
- [0093] 또 다른 경우에는, 디코딩 유닛 m 의 제거 시간이 다음과 같이 특정된다. $low_delay_hrd_flag$ 가 0과 동일하거나 $t_{r,n}(m) \geq t_{af}(m)$ 일 경우, 디코딩 유닛 m 의 제거 시간은 $t_r(m) = t_{r,n}(m)$ 에 의해 특정된다. 이와 다를 경우($low_delay_hrd_flag$ 가 1과 동일하고 $t_{r,n}(m) < t_{af}(m)$ 임), 액세스 유닛에서의 최후 디코딩 유닛이 아닌 디코딩 유닛 m 의 제거 시간은 $t_r(m) = t_{af}(m)$ 에 의해 특정되고, 액세스 유닛에서의 최후 디코딩 유닛인 디코딩 유닛 m 의 제거 시간은 $t_r(m) = t_{r,n}(m) + T_c * Ceil((t_{af}(m) - t_{r,n}(m)) / t_c)$ 에 의해 특정된다. 후자의 경우($low_delay_hrd_flag$ 가 1과 동일하고 $t_{r,n}(m) < t_{af}(m)$ 임)는, 디코딩 유닛 m 의 사이즈 $b(m)$ 이 너무 커서 그것이 명목 제거 시간에서의 제거를 방지함을 표시한다.
- [0094] 또 다른 경우에는, 디코딩 유닛 m 의 제거 시간이 다음과 같이 특정된다. $low_delay_hrd_flag$ 가 0과 동일하거나

$t_{r,n}(m) \geq t_{af}(m)$ 일 경우, 디코딩 유닛 m 의 제거 시간은 $t_r(m) = t_{r,n}(m)$ 에 의해 특정된다. 이와 다른 경우($low_delay_hrd_flag$ 가 1과 동일하고 $t_{r,n}(m) < t_{af}(m)$ 임), 액세스 유닛에서의 최후 디코딩 유닛이 아닌 디코딩 유닛 m 의 제거 시간은 $t_r(m) = t_{af}(m)$ 에 의해 특정되고, 액세스 유닛에서의 최후 디코딩 유닛인 디코딩 유닛 m 의 제거 시간은 $t_r(m) = t_{r,n}(m) + t_c * Ceil((t_{af}(m) - t_{r,n}(m)) / t_c)$ 에 의해 특정된다. 후자의 경우($low_delay_hrd_flag$ 가 1과 동일하고 $t_{r,n}(m) < t_{af}(m)$ 임)는, 디코딩 유닛 m 의 사이즈 $b(m)$ 이 너무 커서 그것이 명목 제거 시간에서의 제거를 방지함을 표시한다.

[0095] 또 다른 경우에는, 디코딩 유닛 m 의 제거 시간이 다음과 같이 특정된다. $low_delay_hrd_flag$ 가 0과 동일하거나 $t_{r,n}(m) \geq t_{af}(m)$ 일 경우, 디코딩 유닛 m 의 제거 시간은 $t_r(m) = t_{r,n}(m)$ 에 의해 특정된다. 이와 다른 경우($low_delay_hrd_flag$ 가 1과 동일하고 $t_{r,n}(m) < t_{af}(m)$ 임), 디코딩 유닛 m 의 제거 시간은 $t_r(m) = t_{af}(m)$ 에 의해 특정된다. 후자의 경우($low_delay_hrd_flag$ 가 1과 동일하고 $t_{r,n}(m) < t_{af}(m)$ 임)는, 디코딩 유닛 m 의 사이즈 $b(m)$ 이 너무 커서 그것이 명목 제거 시간에서의 제거를 방지함을 표시한다.

[0096] $SubPicCpbFlag$ 가 1과 동일할 때, 액세스 유닛 n 의 명목 CPB 제거 시간 $t_{r,n}(n)$ 은 액세스 유닛 n 에서의 최후 디코딩 유닛의 명목 CPB 제거 시간으로 설정되고, 액세스 유닛 n 의 CPB 제거 시간 $t_r(n)$ 은 액세스 유닛 n 에서의 최후 디코딩 유닛의 CPB 제거 시간으로 설정된다.

[0097] $SubPicCpbFlag$ 가 0과 동일할 때, 각각의 디코딩 유닛은 액세스 유닛이고, 이 때문에, 액세스 유닛 n 의 명목 CPB 제거 시간 및 CPB 제거 시간은 디코딩 유닛 n 의 명목 CPB 제거 시간 및 CPB 제거 시간이다.

[0098] 디코딩 유닛 m 의 CPB 제거 시간에서, 디코딩 유닛은 순간적으로 디코딩된다.

[0099] 본원에서 개시된 시스템들 및 방법들에 따른 픽처 타이밍 SEI 메시지에 대한 수정된 선택스 및 시맨틱들의 또 다른 예는 다음과 같이 표 (2)에서 주어진다. 본원에서 개시된 시스템들 및 방법들에 따른 수정들은 굵게 나타낸다.

표 2

<code>pic_timing(payloadSize) {</code>
<code> iff(CpbDpbDelaysPresentFlag) {</code>
<code> cpb_removal_delay</code>
<code> dpb_output_delay</code>
<code> iff(sub_pic_cpb_params_present_flag) {</code>
<code> num_decoding_units_minus1</code>
<code> common_du_cpb_removal_delay_flag</code>
<code> iff(common_du_cpb_removal_delay_flag) {</code>
<code> common_num_nalus_in_du_minus1</code>
<code> common_du_cpb_removal_delay</code>
<code> }</code>
<code> for(i=0; i <= num_decoding_units_minus1; i++) {</code>
<code> num_nalus_in_du_minus1[i]</code>
<code> iff(common_du_cpb_removal_delay_flag)</code>
<code> du_cpb_removal_delay[i]</code>
<code> }</code>
<code> }</code>
<code> }</code>
<code>}</code>

표 (2)

[0100]

[0101] 표 (2)에서 예시된 예는 선택스 엘리먼트 `common_num_nalus_in_du_minus1`를 표시하고, 이것은 디코딩 유닛을 제거할 때에 CPB로부터 얼마나 많은 데이터가 제거되어야 하는지를 결정하기 위하여 이용될 수도 있다. 'common_num_nalus_in_du_minus1' 플러스 1은 픽처 타이밍 SEI 메시지가 연관되는 액세스 유닛의 각각의 디코딩 유닛에서의 NAL 유닛들의 수를 특정한다. `common_num_nalus_in_du_minus1`의 값은 0 내지 `PicWidthInCtbs * PicHeightInCtbs - 1`까지의 범위에 있을 수도 있다.

[0102] 액세스 유닛의 최초 디코딩 유닛은 액세스 유닛에서의 디코딩 순서에서 최초 `common_num_nalus_in_du_minus1 + 1` 연속적인 NAL 유닛들로 구성된다. 액세스 유닛의 i 번째(i 는 0 보다 더 큼) 디코딩 유닛은 디코딩 순서에서,

액세스 유닛의 이전의 디코딩 유닛에서의 최후 NAL 유닛을 바로 후행하는 common_num_nalus_in_du_minus1 + 1 연속적인 NAL 유닛들로 구성된다. 각각의 디코딩 유닛에서 적어도 하나의 VCL NAL 유닛이 있을 수도 있다. VCL NAL 유닛과 연관된 모든 비-VCL NAL 유닛들은 동일한 디코딩 유닛 내에 포함될 수도 있다.

[0103] 본원에서 개시된 시스템들 및 방법들에 따른 픽처 타이밍 SEI 메시지에 대한 수정된 신택스 및 시맨틱들의 또 다른 예는 다음과 같이 표 (3)에서 주어진다. 본원에서 개시된 시스템들 및 방법들에 따른 수정들은 굵게 나타낸다.

표 3

pic_timing(payloadSize) {
if(CpbDpbDelaysPresentFlag) {
cpb_removal_delay
dpb_output_delay
if(sub_pic_cpb_params_present_flag) {
num_decoding_units_minus1
common_num_nalus_in_du_flag
if(common_num_nalus_in_du_flag) {
common_num_nalus_in_du_minus1
}
common_du_cpb_removal_delay_flag
if(common_du_cpb_removal_delay_flag) {
common_du_cpb_removal_delay
}
for(i = 0; i <= num_decoding_units_minus1; i++) {
if(common_num_nalus_in_du_flag)
num_nalus_in_du_minus1[i]
if(common_du_cpb_removal_delay_flag)
du_cpb_removal_delay[i]
}
}
}
}

표 (3)

[0104]

[0105] 표 (3)에서 예시된 예는, 1과 동일할 때, 신택스 엘리먼트 'common_num_nalus_in_du_minus1'가 존재하는 것으로 특징하는 신택스 엘리먼트 'common_num_nalus_in_du_flag'를 포함한다. 0과 동일한 'common_num_nalus_in_du_flag'는 신택스 엘리먼트 'common_num_nalus_in_du_minus1'가 존재하지 않는 것으로 특징한다.

[0106] 또 다른 실시형태에서, 플래그들 common_du_cpb_removal_delay_flag common_num_nalus_in_du_minus1은 전송되지 않을 수도 있다. 그 대신에, 신택스 엘리먼트들 common_num_nalus_in_du_minus1 및 common_du_cpb_removal_delay가 매 시간에 전송될 수 있다. 이 경우, 이 엘리먼트들이 시그널링되지 않음을 표시하기 위하여, 이 신택스 엘리먼트들에 대한 0(또는 일부의 다른)의 값이 이용될 수 있다.

[0107] 픽처 타이밍 SEI 메시지의 신택스 엘리먼트들 및 시맨틱들에 대한 수정들에 추가하여, 본 시스템을 및 방법들은 또한, 서브-픽처 기반 CPB 동작 및 액세스 유닛 레벨 CPB 동작이 디코딩 유닛 제거의 동일한 타이밍으로 귀착되도록 비트스트림 제약을 구현할 수도 있다.

[0108] sub_pic_cpb_params_present_flag가 1과 동일하여 서브-픽처 레벨 CPB 제거 지연 파라미터들이 존재할 때, CPB는 액세스 유닛 레벨 또는 서브-픽처 레벨에서 동작할 수도 있다. 0과 동일한 sub_pic_cpb_params_present_flag는 서브-픽처 레벨 CPB 제거 지연 파라미터들이 존재하지 않으며 CPB가 액세스 유닛 레벨에서 동작하는 것으로 특징한다. sub_pic_cpb_params_present_flag가 존재하지 않을 때, 그 값은 0과 동일한 것으로 추론된다.

[0109] 액세스 유닛 레벨 또는 서브-픽처 레벨의 양자에서의 동작을 지원하기 위해서는, 다음의 비트스트림 제약들이 이용될 수도 있다: sub_pic_cpb_params_present_flag가 1일 경우, 모든 i에 대한 cpb_removal_delay 및 du_cpb_removal_delay[i]에 대한 값들을 시그널링할 때에 다음의 제약이 준수된다는 것이 비트스트림 적합성 (bitstream conformance)의 요건이다:

수학식 3

$$cpb_removal_delay = \frac{\left(\sum_{i=0}^{num_decoding_units_minus1} du_cpb_removal_delay[i] \right) * t_{c,sub}}{t_c}$$

[0110]

[0111]

여기서, du_cpb_removal_delay[i]는 디코딩 유닛 CPB 제거 지연 파라미터들이고, t_c는 클록 틱이고, t_{c,sub}는 서브-픽처 클록 틱이고, num_decoding_units_minus1은 1에 의해 오프셋된 액세스 유닛에서의 디코딩 유닛들의 양이고, i는 인덱스이다. 일부의 실시형태들에서는, 상기 제약을 만족시키기 위하여 공차 파라미터(tolerance parameter)가 추가될 수 있다.

[0112]

액세스 유닛 레벨 또는 서브-픽처 레벨의 양자에서의 동작을 지원하기 위해서는, 다음과 같은 비트스트림 제약들이 이용될 수도 있다: 변수 T_{du}(k)는 다음과 같이 정의된다고 한다:

수학식 4

$$T_{du}(k) = T_{du}(k-1) + t_{c,sub} * \sum_{i=0}^{num_decoding_units_minus1_k} (du_cpb_removal_delay_minus1_k[i]+1)$$

[0113]

[0114]

여기서, du_cpb_removal_delay_minus1_k[i] 및 num_decoding_units_minus1_k은 k번째 액세스 유닛(HRD를 초기화 하였던 액세스 유닛에 대해 k=0이고, k<1에 대하여 T_{du}(k)=0)의 I번째 디코딩 유닛에 대한 파라미터들이고, 여기서, du_cpb_removal_delay_minus1_k[i]+1=du_cpb_removal_delay_minus1_k[i]는 k번째 액세스 유닛의 I번째 디코딩 유닛에 대한 디코딩 유닛 CPB 제거 지연 파라미터이고, num_decoding_units_minus1_k은 k번째 액세스 유닛에서의 디코딩 유닛들의 수이고, t_c는 클록 틱이고, t_{c,sub}는 서브-픽처 클록 틱이고, i 및 k는 인덱스들이다. 다음으로, 픽처 타이밍 플래그(예컨대, sub_pic_cpb_params_present_flag)가 1로 설정될 때, 다음의 제약은 참일 것이다: (au_cpb_removal_delay_minus1 + 1) * t_c == T_{du}(k), 여기서, (au_cpb_removal_delay_minus1 + 1)=cpb_removal_delay, CPB 제거 지연. 이에 따라, 이 경우, CPB 제거 지연(au_cpb_removal_delay_minus1 + 1)은 서브-픽처 기반 CPB 동작 및 액세스 유닛 기반 CPB 동작의 동작이 액세스 유닛 제거의 동일한 타이밍 및 액세스 유닛 제거의 최후 디코딩 유닛으로 귀착되도록 설정된다.

[0115]

액세스 유닛 레벨 또는 서브-픽처 레벨의 양자에서의 동작을 지원하기 위해서는, 다음의 비트스트림 제약들이 이용될 수도 있다: sub_pic_cpb_params_present_flag가 1일 경우, 모든 i에 대한 cpb_removal_delay 및 du_cpb_removal_delay[i]에 대한 값들을 시그널링할 때에 다음의 제약이 준수된다는 것이 비트스트림 적합성의 요건이다:

수학식 5

$$-1 \leq [cpb_removal_delay * t_c - \left(\sum_{i=0}^{num_decoding_units_minus1} du_cpb_removal_delay[i] \right) * t_{c,sub}] \leq 1$$

[0116]

[0117]

여기서, du_cpb_removal_delay[i]는 디코딩 유닛 CPB 제거 지연 파라미터들이고, t_c는 클록 틱이고, t_{c,sub}는 서브-픽처 클록 틱이고, num_decoding_units_minus1은 1에 의해 오프셋된 액세스 유닛에서의 디코딩 유닛들의 양이고, i는 인덱스이다.

[0118]

액세스 유닛 레벨 또는 서브-픽처 레벨의 양자에서의 동작을 지원하기 위해서는, 다음의 비트스트림 제약들이 이용될 수도 있다: sub_pic_cpb_params_present_flag가 1일 경우, 그것은 cpb_removal_delay 및

du_cpb_removal_delay[num_decoding_units_minus1]에 대한 값들을 시그널링할 때에 다음의 제약이 준수된다는 비트스트림 적합성의 요건이고: $cpb_removal_delay * t_c = du_cpb_removal_delay[num_decoding_units_minus1] * t_{c,sub}$ 이고, 여기서, du_cpb_removal_delay[num_decoding_units_minus1]은 num_decoding_units_minus1번째 디코딩 유닛에 대한 디코딩 유닛 CPB 제거 지연 파라미터이고, t_c 는 클록 틱이고, $t_{c,sub}$ 는 서브-픽처 클록 틱이고, num_decoding_units_minus1은 1에 의해 오프셋된 액세스 유닛에서의 디코딩 유닛들의 양이다. 일부의 실시형태들에서는, 상기 제약을 만족시키기 위하여 공차 파라미터가 추가될 수 있다.

[0119] 액세스 유닛 레벨 또는 서브-픽처 레벨의 양자에서의 동작을 지원하기 위해서는, 다음의 비트스트림 제약들이 이용될 수도 있다: sub_pic_cpb_params_present_flag가 1일 경우, 모든 i에 대한 cpb_removal_delay 및 du_cpb_removal_delay[i]에 대한 값들을 시그널링할 때에 다음의 제약이 준수된다는 것이 비트스트림 적합성의 요건이다: $-1 \leq (cpb_removal_delay * t_c - du_cpb_removal_delay[num_decoding_units_minus1] * t_{c,sub}) \leq 1$ 이고, 여기서, du_cpb_removal_delay[num_decoding_units_minus1]은 num_decoding_units_minus1번째 디코딩 유닛에 대한 디코딩 유닛 CPB 제거 지연 파라미터이고, t_c 는 클록 틱이고, $t_{c,sub}$ 는 서브-픽처 클록 틱이고, num_decoding_units_minus1은 1에 의해 오프셋된 액세스 유닛에서의 디코딩 유닛들의 양이다.

[0120] 추가적으로, 본 시스템들 및 방법들은 디코딩 유닛 제거의 타이밍을 수정할 수도 있다. 서브-픽처 레벨 CPB 제거 지연 파라미터들이 존재할 때, "큰 픽처들"에 대한 디코딩 유닛의 제거 시간($low_delay_hrd_flag$ 가 1이고 $t_{r,n}(m) < t_{af}(m)$ 일 때)은 클록 틱 카운터 및 서브-픽처 클록 틱 카운터로 인해 발생할 수 있는 차이를 보상하기 위하여 변경될 수도 있다.

[0121] sub_pic_cpb_params_present_flag가 1과 동일할 때에는, 서브-픽처 레벨 CPB 제거 지연 파라미터들이 존재하고, CPB는 액세스 유닛 레벨 또는 서브-픽처 레벨에서 동작할 수도 있다. 0과 동일한 sub_pic_cpb_params_present_flag는 서브-픽처 레벨 CPB 제거 지연 파라미터들이 존재하지 않으며 CPB가 액세스 유닛 레벨에서 동작하는 것으로 특정한다. sub_pic_cpb_params_present_flag가 존재하지 않을 때, 그 값은 0과 동일한 것으로 추론된다.

[0122] 구체적으로, 디코딩 유닛 제거의 타이밍 및 디코딩 유닛 구현예의 디코딩의 하나의 예는 다음과 같다. 변수 SubPicCpbPreferredFlag는 외부 수단에 의해 특정되거나, 또는 외부 수단에 의해 특정되지 않을 때에는 0으로 설정된다. 변수 SubPicCpbFlag는 다음과 같이 유도된다: $SubPicCpbFlag = SubPicCpbPreferredFlag \&\& sub_pic_cpb_params_present_flag$. SubPicCpbFlag가 0과 동일할 경우, CPB는 액세스 유닛 레벨에서 동작하고, 각각의 디코딩 유닛은 액세스 유닛이다. 이와 다를 경우, CPB는 서브-픽처 레벨에서 동작하고, 각각의 디코딩 유닛은 액세스 유닛의 서브세트이다.

[0123] SubPicCpbFlag가 0과 동일할 경우, 변수 CpbRemovalDelay(m)은 디코딩 유닛 m인 액세스 유닛과 연관된 픽처 타이밍 SEI 메시지에서의 cpb_removal_delay의 값으로 설정되고, 변수 T_c 는 t_c 로 설정된다. 이와 다를 경우, 변수 CpbRemovalDelay(m)은 디코딩 유닛 m을 포함하는 액세스 유닛과 연관된 픽처 타이밍 SEI 메시지에서의 디코딩 유닛 m에 대한 du_cpb_removal_delay[i]의 값으로 설정되고, 변수 T_c 는 $t_{c,sub}$ 로 설정된다.

[0124] 디코딩 유닛 m이 0과 동일한 n을 갖는 디코딩 유닛(HRD를 초기화하는 액세스 유닛의 최초 디코딩 유닛)일 때, CPB로부터의 디코딩 유닛의 명목 제거 시간은 $t_{r,n}(0) = InitCpbRemovalDelay[SchedSelIdx] / 90000$ 에 의해 특정된다.

[0125] 디코딩 유닛 m이 HRD를 초기화하지 않는 버퍼링 주기의 최초 액세스 유닛의 최초 디코딩 유닛일 때, CPB로부터의 디코딩 유닛의 명목 제거 시간은 $t_{r,n}(m) = t_{r,n}(m_b) + T_c * CpbRemovalDelay(m)$ 에 의해 특정되고, 여기서, $t_{r,n}(m_b)$ 은 이전 버퍼링 주기의 최초 디코딩 유닛의 명목 제거 시간이다.

[0126] 디코딩 유닛 m이 버퍼링 주기의 최초 디코딩 유닛일 때, m_b 는 디코딩 유닛 m의 제거 시간 $t_{r,n}(m)$ 에서 m과 동일하게 설정된다.

[0127] 버퍼링 주기의 최초 디코딩 유닛이 아닌 디코딩 유닛 m의 명목 제거 시간 $t_{r,n}(m)$ 은 $t_{r,n}(m) = t_{r,n}(m_b) + T_c * CpbRemovalDelay(m)$ 에 의해 주어지고, 여기서, $t_{r,n}(m_b)$ 는 현재의 버퍼링 주기의 최초 디코딩 유닛의 명목 제거 시간이다.

- [0128] 디코딩 유닛 m 의 제거 시간은 다음과 같이 특정된다. 변수 ClockDiff는 $ClockDiff = (num_units_in_tick - (num_units_in_sub_tick * (num_decoding_units_minus1 + 1))) / time_scale$ 로 정의된다. 일부의 경우에는, 파라미터들 num_units_in_tick, num_units_in_sub_tick, num_decoding_units_minus1은 다음의 수학적 식이 만족되도록 시그널링된다는 것이 비트스트림 적합성의 요건일 수도 있다. $(num_units_in_tick - (num_units_in_sub_tick * (num_decoding_units_minus1 + 1))) \geq 0$
- [0129] 일부의 다른 경우에는, 파라미터들 num_units_in_tick, num_units_in_sub_tick, num_decoding_units_minus1은 다음의 수학적 식이 만족되도록 시그널링될 수도 있다는 것이 비트스트림 적합성의 요건일 수도 있다. $(num_units_in_tick - (num_units_in_sub_tick * (num_decoding_units_minus1 + 1))) \leq 0$ low_delay_hrd_flag가 0과 동일하거나 $t_{r,n}(m) \geq t_{af}(m)$ 일 경우, 디코딩 유닛 m의 제거 시간은 $t_r(m) = t_{r,n}(m)$ 에 의해 특정된다.
- [0130] 이와 다른 경우(low_delay_hrd_flag가 1과 동일하고 $t_{r,n}(m) < t_{af}(m)$ 임), 그리고 sub_pic_cpb_params_present_flag가 1과 동일하고 CPB가 서브-픽처 레벨에서 동작하고 있을 때, 그리고 ClockDiff가 제로보다 더 클 경우에는, 그것이 액세스 유닛 n의 최후 디코딩 유닛일 때의 디코딩 유닛 m의 제거 시간은 $t_r(m) = t_{r,n}(m) + T_c * Ceil((t_{af}(m) - t_{r,n}(m)) / T_c) + ClockDiff$ 에 의해 특정된다.
- [0131] 이와 다른 경우(low_delay_hrd_flag가 1과 동일하고 $t_{r,n}(m) < t_{af}(m)$ 임), 그리고 sub_pic_cpb_params_present_flag가 1과 동일하고 CPB가 액세스 유닛 레벨에서 동작하고 있을 때, 그리고 ClockDiff가 제로보다 더 작을 경우, 액세스 유닛 n의 제거 시간은 $t_r(m) = t_{r,n}(m) + t_c * Ceil((t_{af}(m) - t_{r,n}(m)) / t_c) - ClockDiff$ 에 의해 특정된다.
- [0132] 이와 다른 경우(low_delay_hrd_flag가 1과 동일하고 $t_{r,n}(m) < t_{af}(m)$ 임), 디코딩 유닛 m의 제거 시간은 $t_r(m) = t_{r,n}(m) + T_c * Ceil((t_{af}(m) - t_{r,n}(m)) / T_c)$ 에 의해 특정된다. 후자의 경우(low_delay_hrd_flag가 1과 동일하고 $t_{r,n}(m) < t_{af}(m)$ 임)는, 디코딩 유닛 m의 사이즈 $b(m)$ 이 너무 커서 그것이 명목 제거 시간에서의 제거를 방지함을 표시한다.
- [0133] 이와 다른 경우(low_delay_hrd_flag가 1과 동일하고 $t_{r,n}(m) < t_{af}(m)$ 임), 그리고 픽처 타이밍 플래그가 1로 설정되고, CPB가 서브-픽처 레벨에서 동작하고 있을 때, 액세스 유닛의 최후 디코딩 유닛 m에 대한 제거 시간 $t_r(m)$ 은 $t_r(m) = t_{r,n}(m) + \min((t_{c_sub} * Ceil((t_{af}(m) - t_{r,n}(m)) / t_{c_sub})), (t_c * Ceil((t_{af}(n) - t_{r,n}(n)) / t_c)))$ 에 따라 결정되고, 여기서, $t_{r,n}(m)$ 은 최후 디코딩 유닛 m의 명목 제거 시간이고, t_{c_sub} 는 서브-픽처 클록 틱이고, Ceil()은 천장 함수이고, $t_{af}(m)$ 은 최후 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이다.
- [0134] 이와 다른 경우(low_delay_hrd_flag가 1과 동일하고 $t_{r,n}(n) < t_{af}(n)$ 임), 그리고 픽처 타이밍 플래그가 1로 설정되고, CPB가 액세스 유닛 레벨에서 동작하고 있을 때, 액세스 유닛 n에 대한 제거 시간 $t_r(n)$ 은 $t_r(n) = t_{r,n}(n) + \min((t_{c_sub} * Ceil((t_{af}(m) - t_{r,n}(m)) / t_{c_sub})), (t_c * Ceil((t_{af}(n) - t_{r,n}(n)) / t_c)))$ 에 따라 결정되고, 여기서, $t_{r,n}(m)$ 은 최후 디코딩 유닛 n의 명목 제거 시간이고, t_{c_sub} 는 서브-픽처 클록 틱이고, Ceil()은 천장 함수이고, $t_{af}(m)$ 은 최후 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이다.
- [0135] 이와 다른 경우(low_delay_hrd_flag가 1과 동일하고 $t_{r,n}(m) < t_{af}(m)$ 임), 그리고 픽처 타이밍 플래그가 1로 설정되고, CPB가 서브-픽처 레벨에서 동작하고 있을 때, 액세스 유닛의 최후 디코딩 유닛 m에 대한 제거 시간 $t_r(m)$ 은 $t_r(m) = t_{r,n}(m) + (t_c * Ceil((t_{af}(n) - t_{r,n}(n)) / t_c))$ 에 따라 결정되고, 여기서, $t_{r,n}(m)$ 은 최후 디코딩 유닛 m의 명목 제거 시간이고, t_{c_sub} 는 서브-픽처 클록 틱이고, Ceil()은 천장 함수이고, $t_{af}(m)$ 은 최후 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이다.
- [0136] 이와 다른 경우(low_delay_hrd_flag가 1과 동일하고 $t_{r,n}(n) < t_{af}(n)$ 임), 그리고 픽처 타이밍 플래그가 1로

설정되고, CPB가 액세스 유닛 레벨에서 동작하고 있을 때, 액세스 유닛 n에 대한 제거 시간 $t_r(n)$ 은 $t_r(n) = t_{r,n}(n) + (t_c * \text{Ceil}((t_{af}(n) - t_{r,n}(n)) / t_c))$ 에 따라 결정되고, 여기서, $t_{r,n}(m)$ 은 최후 디코딩 유닛 n의 명목 제거 시간이고, t_{c_sub} 는 서브-픽처 클록 틱이고, $\text{Ceil}()$ 은 천장 함수이고, $t_{af}(m)$ 은 최후 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이다.

[0137] 이와 다를 경우($\text{low_delay_hrd_flag}$ 가 1과 동일하고 $t_{r,n}(m) < t_{af}(m)$ 임), 그리고 픽처 타이밍 플래그가 1로 설정되고, CPB가 서브-픽처 레벨에서 동작하고 있을 때, 액세스 유닛의 최후 디코딩 유닛이 아닌 디코딩 유닛에 대한 제거 시간은 $t_r(m) = t_{af}(m)$ 로서 설정되고, 여기서, $t_{af}(m)$ 은 디코딩 유닛 m의 최종 도달 시간이다. 그리고, 액세스 유닛의 최후 디코딩 유닛 m의 제거 시간 $t_r(m)$ 은 $t_r(m) = t_{r,n}(m) + (t_{c_sub} * \text{Ceil}((t_{af}(m) - t_{r,n}(m)) / t_{c_sub}))$ 에 따라 결정되고, 여기서, $t_{r,n}(m)$ 은 최후 디코딩 유닛 m의 명목 제거 시간이고, t_{c_sub} 는 서브-픽처 클록 틱이고, $\text{Ceil}()$ 은 천장 함수이고, $t_{af}(m)$ 은 최후 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이고, $t_{af}(m)$ 은 액세스 유닛 n에서의 최후 디코딩 유닛 m의 최종 도달 시간이다.

[0138] 이와 다를 경우($\text{low_delay_hrd_flag}$ 가 1과 동일하고 $t_{r,n}(m) < t_{af}(m)$ 임), 그리고 픽처 타이밍 플래그가 1로 설정되고, CPB가 서브-픽처 레벨에서 동작하고 있을 때, 액세스 유닛의 최후 디코딩 유닛이 아닌 디코딩 유닛에 대한 제거 시간은 $t_r(m) = t_{af}(m)$ 로서 설정되고, 여기서, $t_{af}(m)$ 은 디코딩 유닛 m의 최종 도달 시간이다. 그리고, 액세스 유닛의 최후 디코딩 유닛 m의 제거 시간 $t_r(m)$ 은 $t_r(m) = t_{r,n}(m) + (t_c * \text{Ceil}((t_{af}(m) - t_{r,n}(m)) / t_c))$ 에 따라 결정되고, 여기서, $t_{r,n}(m)$ 은 최후 디코딩 유닛 m의 명목 제거 시간이고, t_{c_sub} 는 서브-픽처 클록 틱이고, $\text{Ceil}()$ 은 천장 함수이고, $t_{af}(m)$ 은 최후 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이고, $t_{af}(m)$ 은 액세스 유닛 n에서의 최후 디코딩 유닛 m의 최종 도달 시간이다.

[0139] 이와 다를 경우($\text{low_delay_hrd_flag}$ 가 1과 동일하고 $t_{r,n}(m) < t_{af}(m)$ 임), 그리고 픽처 타이밍 플래그가 1로 설정되고, CPB가 서브-픽처 레벨에서 동작하고 있을 때, 디코딩 유닛에 대한 제거 시간은 $t_r(m) = t_{af}(m)$ 로서 설정되고, 여기서, $t_{r,n}(m)$ 은 디코딩 유닛 m의 명목 제거 시간이고, t_{c_sub} 는 서브-픽처 클록 틱이고, $\text{Ceil}()$ 은 천장 함수이고, $t_{af}(m)$ 은 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이고, $t_{af}(m)$ 은 액세스 유닛 n에서의 디코딩 유닛 m의 최종 도달 시간이다.

[0140] 이와 다를 경우($\text{low_delay_hrd_flag}$ 가 1과 동일하고 $t_{r,n}(n) < t_{af}(n)$ 임), 그리고 픽처 타이밍 플래그가 1로 설정되고, CPB가 액세스 유닛 레벨에서 동작하고 있을 때, 액세스 유닛 n에 대한 제거 시간 $t_r(n)$ 은 $t_r(n) = t_{af}(n)$ 에 따라 결정되고, 여기서, $t_{r,n}(m)$ 은 최후 디코딩 유닛 n의 명목 제거 시간이고, t_{c_sub} 는 서브-픽처 클록 틱이고, $\text{Ceil}()$ 은 천장 함수이고, $t_{af}(m)$ 은 최후 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이다.

[0141] SubPicCpbFlag 가 1과 동일할 때, 액세스 유닛 n의 명목 CPB 제거 시간 $t_{r,n}(n)$ 은 액세스 유닛 n에서의 최후 디코딩 유닛의 명목 CPB 제거 시간으로 설정되고, 액세스 유닛 n의 CPB 제거 시간 $t_r(n)$ 은 액세스 유닛 n에서의 최후 디코딩 유닛의 CPB 제거 시간으로 설정된다.

[0142] SubPicCpbFlag 가 0과 동일할 때, 각각의 디코딩 유닛은 액세스 유닛이고, 이 때문에, 액세스 유닛 n의 명목 CPB 제거 시간 및 CPB 제거 시간은 디코딩 유닛 n의 명목 CPB 제거 시간 및 CPB 제거 시간이다. 디코딩 유닛 m의 CPB 제거 시간에서, 디코딩 유닛은 순간적으로 디코딩된다.

[0143] 상기한 것에 의해 예시된 바와 같이, 본원에서 개시된 시스템들 및 방법들은 서브-픽처 기반 파라미터들을 반송하는 픽처 타이밍 SEI 메시지 비트스트림들을 수정하는 신택스 및 시맨틱들을 제공한다. 일부의 구성들에서,

본원에서 개시된 시스템들 및 방법들은 HEVC 사양들에 적용될 수도 있다.

- [0144] 편의상, 몇몇 정의들은 다음과 같이 주어지고, 이것은 본원에서 개시된 시스템들 및 방법들에 적용될 수도 있다. 랜덤 액세스 포인트(random access point)는, 비트스트림의 디코딩이 현재의 픽처와, 출력 순서에서 상기 현재의 픽처에 후속하는 모든 픽처들을 디코딩하기 위하여 랜덤 액세스 포인트를 선행하는 비트스트림에서의 임의의 포인트에 대한 액세스를 요구하지 않는 데이터의 스트림(예컨대, 비트스트림)에서의 임의의 포인트일 수도 있다.
- [0145] 버퍼링 주기는 디코딩 순서에서 버퍼링 주기 SEI 메시지의 2 개의 인스턴스(instance)들 사이의 액세스 유닛들의 세트로서 특정될 수도 있다. 보충 강화 정보(SEI)는 VCL NAL 유닛들로부터의 코딩된 픽처들의 샘플들을 디코딩하기 위하여 필요하지 않은 정보를 포함할 수도 있다. SEI 메시지들은 디코딩, 디스플레이 또는 다른 목적들에 관련된 절차들을 보조할 수도 있다. 준수하는 디코더들은 HEVC 사양들에 대한 출력 순서 적합성을 위하여 이 정보를 프로세싱하도록 요구받지 않을 수도 있다(예를 들어, HEVC 사양들(JCTVC-I1003)의 부록 C는 준수를 위한 사양들을 포함함). 일부의 SEI 메시지 정보는 비트스트림 적합성을 검사하기 위하여, 그리고 출력 타이밍 디코더 적합성을 위해 이용될 수도 있다.
- [0146] 버퍼링 주기 SEI 메시지는 버퍼링 주기에 관련된 SEI 메시지일 수도 있다. 픽처 타이밍 SEI 메시지는 CPB 제거 타이밍에 관련된 SEI 메시지일 수도 있다. 이 메시지들은 비트스트림 도달 타이밍 및 코딩된 픽처 제거 타이밍을 정의하는 신택스 및 시맨틱들을 정의할 수도 있다.
- [0147] 코딩된 픽처 버퍼(CPB)는 가상 참조 디코더(HRD)에서 특정된 디코딩 순서에서 액세스 유닛들을 포함하는 선입선출(first-in first-out) 버퍼일 수도 있다. 액세스 유닛은 디코딩 순서에서 연속적이며 정확하게 하나의 코딩된 픽처를 포함하는 네트워크 액세스 계층(NAL) 유닛들의 세트일 수도 있다. 코딩된 픽처의 코딩된 슬라이스 NAL 유닛들에 추가하여, 액세스 유닛은 또한, 코딩된 픽처의 슬라이스들을 포함하지 않는 다른 NAL 유닛들을 포함할 수도 있다. 액세스 유닛의 디코딩은 항상 디코딩된 픽처로 귀착된다. NAL 유닛은 에뮬레이션 방지 바이트(emulation prevention byte)들로 필요에 따라 산재된 원시 바이트 시퀀스 페이로드의 형태로, 뒤따르는 데이터의 타입의 표시 및 그 데이터를 포함하는 바이트들을 포함하는 신택스 구조일 수도 있다.
- [0148] 본원에서 이용된 바와 같이, 용어 "공통"은 일반적으로 하나를 초과하는 것에 적용가능한 신택스 엘리먼트 또는 변수를 지칭한다. 예를 들어, 픽처 타이밍 SEI 메시지에서의 신택스 엘리먼트들의 문맥에서, 용어 "공통"은 신택스 엘리먼트(예컨대, common_du_cpb_removal_delay)가 픽처 타이밍 SEI 메시지와 연관된 액세스 유닛에서의 모든 디코딩 유닛들에 적용가능하다는 것을 의미할 수도 있다. 추가적으로, 데이터의 유닛들은 액세스 유닛들 및 디코딩 유닛들을 각각 일반적으로 지칭하는 "n" 및 "m"의 측면에서 설명된다.
- [0149] 다양한 구성들은 도면들을 참조하여 지금부터 설명되며, 여기서, 유사한 참조 번호들은 기능적으로 유사한 엘리먼트들을 표시할 수도 있다. 본원의 도면들에서 일반적으로 설명되고 예시된 바와 같은 시스템들 및 방법들은 매우 다양한 상이한 구성들로 배치되고 설계될 수 있다. 이에 따라, 도면들에서 나타난 바와 같이, 몇몇 구성들의 다음의 더욱 상세한 설명은 청구된 바와 같이 범위를 제한하도록 의도된 것이 아니라, 단지 시스템들 및 방법들을 대표한다.
- [0150] 도 1a는 메시지를 전송하고 비트스트림을 버퍼링하기 위한 시스템들 및 방법들이 구현될 수도 있는 하나 이상의 전자 디바이스들(102)의 예를 예시하는 블록도이다. 이 예에서는, 전자 디바이스 A(102a) 및 전자 디바이스 B(102b)가 예시된다. 그러나, 전자 디바이스 A(102a) 및 전자 디바이스 B(102b)와 관련하여 설명된 특징들 및 기능성의 하나 이상은 일부의 구성들에서 단일 전자 디바이스로 조합될 수도 있다는 것에 주목해야 한다.
- [0151] 전자 디바이스 A(102a)는 인코더(104)를 포함한다. 인코더(104)는 메시지 생성 모듈(108)을 포함한다. 전자 디바이스 A(102a) 내에 포함된 엘리먼트들(예컨대, 인코더(104) 및 메시지 생성 모듈(108))의 각각은 하드웨어, 소프트웨어 또는 양자의 조합으로 구현될 수도 있다.
- [0152] 전자 디바이스 A(102a)는 하나 이상의 입력 픽처들(106)을 획득할 수도 있다. 일부의 구성들에서, 입력 픽처(들)(106)는 이미지 센서를 이용하여 전자 디바이스 A(102a) 상에서 캡처될 수도 있고, 메모리로부터 취출될 수도 있고, 및/또는 또 다른 전자 디바이스로부터 수신될 수도 있다.
- [0153] 인코더(104)는 인코딩된 데이터를 생성하기 위하여 입력 픽처(들)(106)를 인코딩할 수도 있다. 예를 들어, 인코더(104)는 일련의 입력 픽처들(106)(예컨대, 비디오)을 인코딩할 수도 있다. 하나의 구성에서, 인코더(104)는 HEVC 인코더일 수도 있다. 인코딩된 데이터는 디지털 데이터(예컨대, 비트스트림(114)의 일부)일 수도

있다. 인코더(104)는 입력 신호에 기반으로 하여 오버헤드 시그널링(overhead signaling)을 생성할 수도 있다.

[0154] 메시지 생성 모듈(108)은 하나 이상의 메시지들을 생성할 수도 있다. 예를 들어, 메시지 생성 모듈(108)은 하나 이상의 SEI 메시지들 또는 다른 메시지들을 생성할 수도 있다. 서브-픽처 레벨 상에서의 동작을 지원하는 CPB에 대하여, 전자 디바이스(102)는 서브-픽처 파라미터들(예컨대, CPB 제거 지연 파라미터)을 전송할 수도 있다. 구체적으로, 전자 디바이스(102)(예컨대, 인코더(104))는 픽처 타이밍 SEI 메시지에서 공통 디코딩 유닛 CPB 제거 지연 파라미터를 포함할 것인지 여부를 결정할 수도 있다. 예를 들어, 인코더(104)가 픽처 타이밍 SEI 메시지에서 공통 디코딩 유닛 CPB 제거 지연 파라미터(예컨대, common_du_cpb_removal_delay)를 포함하고 있을 때, 전자 디바이스는 플래그(예컨대, common_du_cpb_removal_delay_flag)를 1로 설정할 수도 있다. 공통 디코딩 유닛 CPB 제거 지연 파라미터가 포함될 때, 전자 디바이스는 액세스 유닛에서의 모든 디코딩 유닛들에 적용가능한 공통 디코딩 유닛 CPB 제거 지연 파라미터를 생성할 수도 있다. 다시 말해서, 액세스 유닛에서의 각각의 디코딩 유닛에 대한 디코딩 유닛 CPB 제거 지연 파라미터를 포함하는 것이 아니라, 공통 파라미터는 픽처 타이밍 SEI 메시지가 연관되는 액세스 유닛에서의 모든 디코딩 유닛들에 적용할 수도 있다.

[0155] 대조적으로, 공통 디코딩 유닛 CPB 제거 지연 파라미터가 픽처 타이밍 SEI 메시지에서 포함되지 않아야 할 때, 전자 디바이스(102)는 픽처 타이밍 SEI 메시지가 연관되는 액세스 유닛에서의 각각의 디코딩 유닛에 대한 별도의 디코딩 유닛 CPB 제거 지연을 생성할 수도 있다. 메시지 생성 모듈(108)은 이하의 도 2 및 도 3과 관련하여 설명된 절차들 중의 하나 이상을 수행할 수도 있다.

[0156] 일부의 구성들에서, 전자 디바이스 A(102a)는 메시지를 비트스트림(114)의 일부로서 전자 디바이스 B(102b)로 전송할 수도 있다. 일부의 구성들에서, 전자 디바이스 A(102a)는 별도의 송신(110)에 의해 메시지를 전자 디바이스 B(102b)로 전송할 수도 있다. 예를 들어, 별도의 송신은 비트스트림(114)의 일부가 아닐 수도 있다. 예를 들어, 픽처 타이밍 SEI 메시지 또는 다른 메시지는 일부의 대역외(out-of-band) 메커니즘을 이용하여 전송될 수도 있다. 일부의 구성들에서, 다른 메시지는 위에서 설명된 픽처 타이밍 SEI 메시지의 특징들 중의 하나 이상을 포함할 수도 있다는 것에 주목해야 한다. 또한, 하나 이상의 양태들에서, 다른 메시지는 위에서 설명된 SEI 메시지와 유사하게 사용될 수도 있다.

[0157] 인코더(104)(및 예를 들어, 메시지 생성 모듈(108))는 비트스트림(114)을 생성할 수도 있다. 비트스트림(114)은 입력 픽처(들)(106)에 기반으로 하여 인코딩된 픽처 데이터를 포함할 수도 있다. 일부의 구성들에서, 비트스트림(114)은 또한, 픽처 타이밍 SEI 메시지 또는 다른 메시지, 슬라이스 헤더(들), PPS(들) 등과 같은 오버헤드 데이터를 포함할 수도 있다. 추가적인 입력 픽처(들)(106)이 인코딩되므로, 비트스트림(114)은 하나 이상의 인코딩된 픽처(들)을 포함할 수도 있다. 예를 들어, 비트스트림(114)은 대응하는 오버헤드 데이터(예컨대, 픽처 타이밍 SEI 메시지 또는 다른 메시지)를 갖는 하나 이상의 인코딩된 픽처(들)을 포함할 수도 있다.

[0158] 비트스트림(114)은 디코더(112)에 제공될 수도 있다. 하나의 예에서, 비트스트림(114)은 유선 또는 무선 링크를 이용하여 전자 디바이스 B(102b)로 송신될 수도 있다. 일부의 경우들에는, 이것이 인터넷 또는 로컬 영역 네트워크(Local Area Network; LAN)와 같은 네트워크를 통해 행해질 수도 있다. 도 1a에서 예시된 바와 같이, 디코더(112)는 전자 디바이스 A(102a) 상의 디코더(104)와는 별도로 전자 디바이스 B(102b) 상에서 구현될 수도 있다. 그러나, 인코더(104) 및 디코더(112)는 일부의 구성들에서 동일한 전자 디바이스 상에서 구현될 수도 있다는 것에 주목해야 한다. 인코더(104) 및 디코더(112)가 동일한 전자 디바이스 상에서 구현되는 구현예에서는, 예를 들어, 비트스트림(114)이 버스를 통해 디코더(112)에 제공될 수도 있거나, 디코더(112)에 의한 취출을 위해 메모리 내에 저장될 수도 있다.

[0159] 디코더 (112) 는 하드웨어, 소프트웨어, 또는 양자의 조합으로 구현될 수도 있다. 하나의 구성에서, 디코더 (112)는 HEVC 디코더일 수도 있다. 디코더(112)는 비트스트림(114)을 수신(예컨대, 획득)할 수도 있다. 디코더(112)는 비트스트림(114)에 기반으로 하여 하나 이상의 디코딩된 픽처(들)(118)을 생성할 수도 있다. 디코딩된 픽처(들)(118)는 디스플레이될 수도 있고, 재생될 수도 있고, 메모리 내에 저장될 수도 있고, 및/또는 또 다른 디바이스로 송신될 수도 있는 등등과 같다.

[0160] 디코더(112)는 CPB(120)를 포함할 수도 있다. CPB(120)는 인코딩된 픽처(들)을 일시적으로 저장할 수도 있다. CPB(120)는 언제 데이터를 제거할 것인지를 결정하기 위하여 픽처 타이밍 SEI 메시지에서 발견된 파라미터들을 이용할 수도 있다. CPB(120)가 서브-픽처 레벨 상에서의 동작을 지원할 때, 한번에 전체의 액세스 유닛들이 아니라, 개별적인 디코딩 유닛들이 제거될 수도 있다. 디코더(112)는 디코딩된 픽처 버퍼(DPB)(122)를 포함할 수도 있다. 각각의 디코딩된 픽처는 디코딩 프로세스에 의해 참조되는 것뿐만 아니라, 출력 및 크롭핑(cropping)을 위해 DPB(122)에서 배치된다. 디코딩된 픽처는 DPB 출력 시간, 또는 인터-예측 참조를 위해 이제 더 이상

필요하게 되지 않는 시간의 더 이후에 DPB로부터 제거된다.

- [0161] 디코더(112)는 메시지(예컨대, 픽처 타이밍 SEI 메시지 또는 다른 메시지)를 수신할 수도 있다. 디코더(112)는 또한, 수신된 메시지가 공통 디코딩 유닛 CPB 제거 지연 파라미터(예컨대, common_du_cpb_removal_delay)를 포함하는지 여부를 결정할 수도 있다. 이것은 공통 파라미터가 픽처 타이밍 SEI 메시지에서 존재할 때에 설정되는 플래그(예컨대, common_du_cpb_removal_delay_flag)를 식별하는 것을 포함할 수도 있다. 공통 파라미터가 존재할 경우, 디코더(112)는 액세스 유닛에서의 모든 디코딩 유닛들에 적용가능한 공통 디코딩 유닛 CPB 제거 지연 파라미터를 결정할 수도 있다. 공통 파라미터가 존재하지 않을 경우, 디코더(112)는 액세스 유닛에서의 각각의 디코딩 유닛에 대한 별도의 디코딩 유닛 CPB 제거 지연 파라미터를 결정할 수도 있다. 디코더(112)는 또한, 공통 디코딩 유닛 CPB 제거 지연 파라미터 또는 별도의 디코딩 유닛 CPB 제거 지연 파라미터들의 어느 하나를 이용하여 CPB(120)로부터 디코딩 유닛들을 제거할 수도 있다. CPB(120)는 이하의 도 4 및 도 5와 관련하여 설명된 절차들 중의 하나 이상을 수행할 수도 있다.
- [0162] 위에서 설명된 HRD는 도 1a에서 예시된 디코더(112)의 하나의 예일 수도 있다. 이에 따라, 전자 디바이스(102)는 일부의 구성들에서, 위에서 설명된 HRD와, CPB(120) 및 DPB(122)에 따라 동작할 수도 있다.
- [0163] 전자 디바이스(들)(102) 내에 포함된 엘리먼트들 또는 그 일부들 중의 하나 이상은 하드웨어로 구현될 수도 있다는 것에 주목해야 한다. 예를 들어, 3 개의 엘리먼트들 또는 그 일부들 중의 하나 이상은 칩, 회로부 또는 하드웨어 컴포넌트들 등으로서 구현될 수도 있다. 또한, 본원에서 설명된 기능들 또는 방법들 중의 하나 이상은 하드웨어를 이용하여 구현될 수도 있고, 및/또는 수행될 수도 있다는 것에 주목해야 한다. 예를 들어, 본원에서 설명된 방법들 중의 하나 이상은 칩셋(chipset), 애플리케이션 특정 집적 회로(Application-Specific Integrated Circuit; ASIC), 대규모 집적 회로(Large-Scale Integrated circuit; LSI) 또는 집적 회로 등을 이용하여 구현될 수도 있고, 및/또는 실현될 수도 있다.
- [0164] 도 1b는 인코더(1908) 및 디코더(1972)의 또 다른 예를 예시하는 블록도이다. 이 예에서는, 전자 디바이스 A(1902) 및 전자 디바이스 B(1970)가 예시된다. 그러나, 전자 디바이스 A(1902) 및 전자 디바이스 B(1870)와 관련하여 설명된 특징들 및 기능성은 일부의 구성들에서 단일 전자 디바이스로 조합될 수도 있다는 것에 주목해야 한다.
- [0165] 전자 디바이스 A(1902)는 인코더(1908)를 포함한다. 인코더(1908)는 기본 계층 인코더(1910) 및 강화 계층 인코더(1920)를 포함할 수도 있다. 비디오 인코더(1908)는 더 이후에 설명된 바와 같이, 스케일러블 비디오 코딩(scalable video coding) 및 멀티-뷰 비디오 코딩(multi-view video coding)을 위해 적당하다. 인코더(1908)는 하드웨어, 소프트웨어, 또는 양자의 조합으로 구현될 수도 있다. 하나의 구성에서, 인코더(1908)는 스케일러블 및/또는 멀티-뷰를 포함하는 고효율 비디오 코딩(HEVC) 코더일 수도 있다. 다른 코더들이 마찬가지로 이용될 수도 있다. 전자 디바이스 A(1902)는 소스(source)(1906)를 획득할 수도 있다. 일부의 구성들에서, 소스(1906)는 이미지 센서를 이용하여 전자 디바이스 A(1902) 상에서 캡처될 수도 있거나, 메모리로부터 취출될 수도 있거나, 또 다른 전자 디바이스로부터 수신될 수도 있다.
- [0166] 인코더(1908)는 기본 계층 비트스트림(1934) 및 강화 계층 비트스트림(1936)을 생성하기 위하여 소스(1906)를 코딩할 수도 있다. 예를 들어, 인코더(1908)는 소스(1906)에서의 일련의 픽처들(예컨대, 비디오)을 코딩할 수도 있다. 특히, 품질 스케일러빌리티(quality scalability)로서 또한 알려진 SNR 스케일러빌리티에 대한 스케일러블 비디오 인코딩을 위하여, 동일한 소스(1906)가 기본 계층 및 강화 계층 인코더에 제공될 수도 있다. 특히, 공간 스케일러빌리티에 대한 스케일러블 비디오 인코딩을 위하여, 다운로드 소스가 기본 계층 인코더를 위해 이용될 수도 있다. 특히, 멀티-뷰 인코딩을 위하여, 상이한 뷰 소스가 기본 계층 인코더 및 강화 계층 인코더를 위해 이용될 수도 있다. 인코더(1908)는 도 6b와 관련하여 더 이후에 설명된 인코더(1782)와 유사할 수도 있다.
- [0167] 비트스트림들(1934, 1936)은 소스(1906)에 기반으로 하여 코딩된 픽처 데이터를 포함할 수도 있다. 일부의 구성들에서, 비트스트림들(1934, 1936)은 또한, 슬라이스 헤더 정보, PPS 정보 등과 같은 오버헤드 데이터를 포함할 수도 있다. 소스(1906)에서의 추가적인 픽처들이 코딩되므로, 비트스트림들(1934, 1936)은 하나 이상의 코딩된 픽처들을 포함할 수도 있다.
- [0168] 비트스트림들(1934, 1936)은 디코더(1972)에 제공될 수도 있다. 디코더(1972)는 기본 계층 디코더(1980) 및 강화 계층 디코더(1990)를 포함할 수도 있다. 비디오 디코더(1972)는 스케일러블 비디오 디코딩 및 멀티-뷰 비디오 디코딩을 위해 적당하다. 하나의 예에서, 비트스트림들(1934, 1936)은 유선 또는 무선 링크를 이용하여 전

자 디바이스 B(1970)로 송신될 수도 있다. 일부의 경우들에는, 이것이 인터넷 또는 로컬 영역 네트워크(LAN)와 같은 네트워크를 통해 행해질 수도 있다. 도 1b에서 예시된 바와 같이, 디코더(1972)는 전자 디바이스 A(1902) 상의 인코더(1908)와는 별도로 전자 디바이스 B(1970) 상에서 구현될 수도 있다. 그러나, 인코더(1908) 및 디코더(1972)는 일부의 구성들에서 동일한 전자 디바이스 상에서 구현될 수도 있다는 것에 주목해야 한다. 인코더(1908) 및 디코더(1972)가 동일한 전자 디바이스 상에서 구현되는 구현예에서는, 예를 들어, 비트스트림들(1934, 1936)이 버스를 통해 디코더(1972)에 제공될 수도 있거나, 디코더(1972)에 의한 추출을 위해 메모리 내에 저장될 수도 있다. 디코더(1972)는 디코딩된 기본 계층(1992) 및 디코딩된 강화 계층 픽처(들)(1994)를 출력력으로서 제공할 수도 있다.

[0169] 디코더(1972)는 하드웨어, 소프트웨어, 또는 양자의 조합으로 구현될 수도 있다. 하나의 구성에서, 디코더(1972)는 스케일러블 및/또는 멀티-뷰를 포함하는 고효율 비디오 코딩(HEVC) 디코더일 수도 있다. 다른 디코더들이 마찬가지로 이용될 수도 있다. 디코더(1972)는 도 7b와 관련하여 더 이후에 설명된 디코더(1812)와 유사할 수도 있다. 또한, 기본 계층 인코더 및/또는 강화 계층 인코더는 각각, 도 1a와 관련하여 설명된 것과 같이, 메시지 생성 모듈을 포함할 수도 있다. 또한, 기본 계층 디코더 및/또는 강화 계층 디코더는 도 1a와 관련하여 설명된 것과 같이, 코딩된 픽처 버퍼 및/또는 디코딩된 픽처 버퍼를 포함할 수도 있다. 게다가, 도 1b의 전자 디바이스들은 적용가능한 바와 같이, 도 1a의 전자 디바이스들의 기능들에 따라 동작할 수도 있다.

[0170] 도 2는 메시지를 전송하기 위한 방법(200)의 하나의 구성을 예시하는 흐름도이다. 방법(200)은 인코더(104) 또는 그 하위-부분들 중의 하나(예컨대, 메시지 생성 모듈(108))에 의해 수행될 수도 있다. 인코더(104)는 CPB(120)가 서브-픽처 레벨 상에서의 동작을 지원하는지 여부를 표시하는 픽처 타이밍 플래그(예컨대, sub_pic_cpb_params_present_flag)를 결정할 수도 있다(202). 예를 들어, 픽처 타이밍 플래그가 1로 설정될 때, CPB(120)는 액세스 유닛 레벨 또는 서브-픽처 레벨 상에서 동작할 수도 있다. 픽처 타이밍 플래그가 1로 설정되더라도, 서브-픽처 레벨에서 실제적으로 동작할 것인지 여부에 대한 판단은 디코더(112) 자체에게 남겨진다.

[0171] 인코더(104)는 또한, 액세스 유닛에서의 디코딩 유닛들에 대한 하나 이상의 제거 지연들을 결정할 수도 있다(204). 예를 들어, 인코더(104)는 CPB(120)로부터의 액세스 유닛에서의 모든 디코딩 유닛들에 적용가능한 단일의 공통 디코딩 유닛 CPB 제거 지연 파라미터(예컨대, common_du_cpb_removal_delay)를 결정할 수도 있다. 대안적으로, 인코더(104)는 액세스 유닛에서의 각각의 디코딩 유닛에 대한 별도의 디코딩 유닛 CPB 제거 지연(예컨대, du_cpb_removal_delay[i])을 결정할 수도 있다.

[0172] 인코더(104)는 또한, 액세스 포인트에서 각각의 디코딩 유닛 내의 NAL 유닛들의, 1에 의해 오프셋된 양을 표시하는 하나 이상의 NAL 파라미터들을 결정할 수도 있다(206). 예를 들어, 인코더(104)는 CPB(120)로부터의 액세스 유닛에서의 모든 디코딩 유닛들에 적용가능한 단일의 공통 NAL 파라미터(예컨대, common_num_nalus_in_du_minus1)를 결정할 수도 있다. 대안적으로, 인코더(104)는 액세스 유닛에서의 각각의 디코딩 유닛에 대한 별도의 디코딩 유닛 CPB 제거 지연(예컨대, num_nalus_in_du_minus1[i])을 결정할 수도 있다.

[0173] 인코더(104)는 또한, 픽처 타이밍 플래그, 제거 지연 및 NAL 파라미터들을 포함하는 픽처 타이밍 SEI 메시지를 전송할 수도 있다. 픽처 타이밍 SEI 메시지는 또한, 다른 파라미터들(예컨대, cpb_removal_delay, dpb_output_delay 등)을 포함할 수도 있다. 예를 들어, 전자 디바이스(102)는 무선 송신, 유선 송신, 디바이스 버스, 네트워크 등 중의 하나 이상을 통해 메시지를 송신할 수도 있다. 예를 들어, 전자 디바이스 A(102a)는 메시지를 전자 디바이스 B(102b)로 송신할 수도 있다. 메시지는 예를 들어, 비트스트림(114)의 일부일 수도 있다. 일부의 구성들에서, 전자 디바이스 A(102a)는 (비트스트림(114)의 일부가 아닌) 별도의 송신(110)에서 메시지를 전자 디바이스 B(102b)로 전송할 수도 있다(208).

[0174] 예를 들어, 메시지는 일부의 대역외 메커니즘을 이용하여 전송될 수도 있다. 일부의 경우들에는, (204, 206)에서 표시된 정보가 픽처 타이밍 SEI 메시지와는 상이한 SEI 메시지에서 전송될 수도 있다. 또 다른 경우에는, (204, 206)에서 표시된 정보가 파라미터 세트 예컨대, 비디오 파라미터 세트 및/또는 시퀀스 파라미터 세트 및/또는 픽처 파라미터 세트 및/또는 적응 파라미터 세트 및/또는 슬라이스 헤더에서 전송될 수도 있다.

[0175] 도 3은 액세스 유닛에서의 디코딩 유닛들에 대한 하나 이상의 제거 지연들을 결정하기 위한 방법(300)의 하나의 구성을 예시하는 흐름도이다. 다시 말해서, 도 3에서 예시된 방법(300)은 도 2에서 예시된 방법(200)에서의 단계(204)를 추가로 예시할 수도 있다. 방법(300)은 인코더(104)에 의해 수행될 수도 있다. 인코더(104)는 공통 디코딩 유닛 CPB 제거 지연 파라미터(예컨대, common_du_cpb_removal_delay)를 포함할 것인지 여부를 결정할 수

도 있다(302).

- [0176] 이것은 공통 디코딩 유닛 CPB 제거 지연 플래그(예컨대, common_du_cpb_removal_delay_flag)가 설정되는지 여부를 결정하는 것을 포함할 수도 있다. 인코더(104)는 디코딩 유닛들이 규칙적인 간격으로 CPB로부터 제거되는 경우에 이 공통 파라미터를 전송할 수도 있다. 이것은 예를 들어, 각각의 디코딩 유닛이 픽처의 어떤 수의 행들에 대응하거나 일부의 다른 규칙적인 구조를 가질 때에 그러할 수도 있다.
- [0177] 예를 들어, 공통 디코딩 유닛 CPB 제거 지연 플래그는, 공통 디코딩 유닛 CPB 제거 지연 파라미터가 픽처 타이밍 SEI 메시지 내에 포함되어야 할 때에 1로, 그리고 그것이 포함되지 않아야 할 때에 0으로 설정될 수도 있다. 긍정일 경우(예컨대, 플래그가 1로 설정됨), 인코더(104)는 액세스 유닛에서의 모든 디코딩 유닛들에 적용가능한 공통 디코딩 유닛 CPB 제거 지연 파라미터(예컨대, common_du_cpb_removal_delay)를 결정할 수도 있다(304). 부정일 경우(예컨대, 플래그가 0으로 설정됨), 인코더(104)는 액세스 유닛에서의 각각의 디코딩 유닛에 대한 별도의 디코딩 유닛 CPB 제거 지연 파라미터들(예컨대, du_cpb_removal_delay[i])을 결정할 수도 있다(306).
- [0178] 공통 디코딩 유닛 CPB 제거 지연 파라미터가 픽처 타이밍 SEI 메시지 내에 존재할 경우, 그것은 CPB(120)로부터 픽처 타이밍 SEI 메시지와 연관된 액세스 유닛에서의 현재의 디코딩 유닛을 제거하기 전에, 바로 선행하는 디코딩 유닛의 CPB(120)로부터의 제거 후에 대기하기 위한 서브-픽처 클록 틱들의 양을 특정할 수도 있다.
- [0179] 예를 들어, 디코딩 유닛이 액세스 유닛에서의 최초 디코딩 유닛일 때, 공통 디코딩 유닛 CPB(120) 제거 지연 파라미터는 CPB(120)로부터 픽처 타이밍 SEI 메시지와 연관된 액세스 유닛에서의 최초 디코딩 유닛을 제거하기 전에, 선행하는 액세스 유닛에서의 가장 최근의 버퍼링 주기 SEI 메시지와 연관된 액세스 유닛에서의 최후 디코딩 유닛의 CPB(120)로부터의 제거 이후에 대기하기 위한 서브-픽처 클록 틱들의 양을 특정할 수도 있다.
- [0180] 디코딩 유닛이 액세스 유닛에서의 최초가 아닌 디코딩 유닛일 때, 공통 디코딩 유닛 CPB 제거 지연 파라미터는 CPB로부터 픽처 타이밍 SEI 메시지와 연관된 액세스 유닛에서의 현재의 디코딩 유닛을 제거하기 전에, 픽처 타이밍 SEI 메시지와 연관된 액세스 유닛에서의 선행하는 디코딩 유닛의 CPB(120)로부터의 제거 후에 대기하기 위한 서브-픽처 클록 틱들의 양을 특정할 수도 있다.
- [0181] 대조적으로, 공통 디코딩 유닛 CPB 제거 지연 파라미터(예컨대, common_du_cpb_removal_delay)가 픽처 타이밍 SEI 메시지에서 전송되지 않을 때, 별도의 디코딩 유닛 CPB 제거 지연 파라미터들(예컨대, du_cpb_removal_delay[i])은 액세스 유닛에서의 각각의 디코딩 유닛에 대한 픽처 타이밍 SEI 메시지 내에 포함될 수도 있다. 디코딩 유닛 CPB 제거 지연 파라미터들(예컨대, du_cpb_removal_delay[i])은 CPB(120)로부터 픽처 타이밍 SEI 메시지와 연관된 액세스 유닛에서의 i번째 디코딩 유닛을 제거하기 전에, 최후 디코딩 유닛의 CPB(120)로부터의 제거 후에 대기하기 위한 서브-픽처 클록 틱들의 양을 특정할 수도 있다. 디코딩 유닛 CPB 제거 지연 파라미터들은 모듈로 $2^{(\text{cpb_removal_delay_length_minus1} + 1)}$ 카운터의 나머지에 따라 계산될 수도 있고, 여기서, $\text{cpb_removal_delay_length_minus1} + 1$ 은 공통 디코딩 유닛 CPB 제거 지연 파라미터의 길이이다.
- [0182] 도 4는 비트스트림을 버퍼링하기 위한 방법(400)의 하나의 구성을 예시하는 흐름도이다. 방법(400)은 전자 디바이스(102)(예컨대, 전자 디바이스 B(102b))에서 디코더(112)에 의해 수행될 수도 있고, 이 전자 디바이스(102)는 메시지(예컨대, 픽처 타이밍 SEI 메시지 또는 다른 메시지)를 수신할 수도 있다(402). 예를 들어, 전자 디바이스(102)는 무선 송신, 유선 송신, 디바이스 버스, 네트워크 등 중의 하나 이상을 통해 메시지를 수신할 수도 있다(402). 예를 들어, 전자 디바이스 B(102b)는 전자 디바이스 A(102a)로부터 메시지를 수신할 수도 있다(402). 메시지는 예를 들어, 비트스트림(114)의 일부일 수도 있다. 또 다른 예에서, 전자 디바이스 B(102b)는 (예를 들어, 비트스트림(114)의 일부가 아닌) 별도의 송신(110)에서, 전자 디바이스 A(102a)로부터 메시지를 수신할 수도 있다. 예를 들어, 픽처 타이밍 SEI 메시지는 일부의 대역외 메커니즘을 이용하여 수신될 수도 있다. 일부의 구성들에서, 메시지는 픽처 타이밍 플래그, 액세스 유닛에서의 디코딩 유닛들에 대한 하나 이상의 제거 지연들 및 하나 이상의 NAL 파라미터들 중의 하나 이상을 포함할 수도 있다. 이에 따라, 메시지를 수신하는 것(402)은 픽처 타이밍 플래그, 액세스 유닛에서의 디코딩 유닛들에 대한 하나 이상의 제거 지연들 및 하나 이상의 NAL 파라미터들 중의 하나 이상을 수신하는 것을 포함할 수도 있다.
- [0183] 디코더(112)는 CPB(120)가 액세스 유닛 레벨 또는 서브-픽처 레벨 상에서 동작하는지 여부를 결정할 수도 있다(404). 예를 들어, 디코더(112)는 그것이 낮은 지연시간(latency)을 달성하기를 원할 경우에 서브-픽처에 기반으로 하여 동작하도록 판단할 수도 있다. 대안적으로, 판단은 디코더(112)가 서브-픽처 기반 동작을 지원하기 위하여 충분한 자원들을 가지는지 여부에 기반으로 할 수도 있다. CPB(120)가 서브-픽처 레벨 상에서 동작할

경우, 디코더는 액세스 유닛에서의 디코딩 유닛들에 대한 하나 이상의 제거 지연들을 결정할 수도 있다(406). 예를 들어, 디코더(112)는 액세스 유닛에서의 모든 디코딩 유닛들에 적용가능한 단일의 공통 디코딩 유닛 CPB 제거 지연 파라미터(예컨대, common_du_cpb_removal_delay)를 결정할 수도 있다. 대안적으로, 디코더(112)는 액세스 유닛에서의 각각의 디코딩 유닛에 대한 별도의 디코딩 유닛 CPB 제거 지연(예컨대, du_cpb_removal_delay[i])을 결정할 수도 있다. 다시 말해서, 픽처 타이밍 SEI 메시지는 액세스 유닛에서의 모든 디코딩 유닛들에 적용가능한 공통 파라미터, 매 디코딩 유닛에 대한 별도의 파라미터들을 포함할 수도 있다.

[0184] 디코더(112)는 또한, 디코딩 유닛들에 대한 제거 지연들에 기반으로 하여, 즉, 액세스 유닛에서의 모든 디코딩 유닛들에 적용가능한 공통 파라미터 또는 매 디코딩 유닛에 대한 별도의 파라미터들의 어느 하나를 이용하여 디코딩 유닛들을 제거할 수도 있다(408). 디코더(112)는 또한, 디코딩 유닛들을 디코딩할 수도 있다(410).

[0185] 디코더(112)는 다양한 시그널링된 파라미터들로부터 결정된 제거 시간을 결정할 때에 변수 ClockDiff를 이용할 수도 있다. 구체적으로, ClockDiff 변수는 $ClockDiff = (num_units_in_tick - (num_units_in_sub_tick * (num_decoding_units_minus1 + 1)) / time_scale)$ 에 따라 결정될 수도 있고, 여기서, num_units_in_tick은 클록 틱 카운터의 하나의 증분(increment)에 대응하는 주파수 time_scale Hz에서 동작하는 클록의 시간 단위들의 수이고, num_units_in_sub_tick은 서브-픽처 클록 틱 카운터의 하나의 증분에 대응하는 주파수 time_scale Hz에서 동작하는 클록의 시간 단위들의 수이고, num_decoding_units_minus1+1은 액세스 유닛에서의 디코딩 유닛들의 양이고, time_scale은 1 초 내에 통과하는 시간 단위들의 수이다.

[0186] 낮은 지연의 가상 참조 디코더(hypothetical reference decoder; HRD) 플래그(예컨대, low_delay_hrd_flag)가 1로 설정되고, $t_{r,n}(m) < t_{af}(m)$ 이고, 픽처 타이밍 플래그가 1로 설정되고, CPB가 서브-픽처 레벨에서 동작하고 있고, ClockDiff가 제로(zero)보다 더 클 때, 디코딩 유닛 m에 대한 제거 시간, $t_r(m)$ 은 $t_r(m) = t_{r,n}(m) + t_{c_sub} * Ceil((t_{af}(m) - t_{r,n}(m)) / t_{c_sub}) + ClockDiff$ 에 따라 결정되고, 여기서, $t_{r,n}(m)$ 은 디코딩 유닛 m에 대한 명목 제거 시간(nominal removal time)이고, t_{c_sub} 는 서브-픽처 클록 틱이고, Ceil()은 천장 함수(ceiling function)이고, $t_{af}(m)$ 은 디코딩 유닛 m의 최종 도달 시간이다.

[0187] 낮은 지연의 가상 참조 디코더(HRD) 플래그(예컨대, low_delay_hrd_flag)가 1로 설정되고, $t_{r,n}(n) < t_{af}(n)$ 이고, 픽처 타이밍 플래그가 1로 설정되고, CPB가 액세스 유닛 레벨에서 동작하고 있고, ClockDiff가 제로(zero)보다 더 클 때, 액세스 유닛 n에 대한 제거 시간, $t_r(n)$ 은 $t_r(n) = t_{r,n}(n) + t_c * Ceil((t_{af}(n) - t_{r,n}(n)) / t_c) - ClockDiff$ 에 따라 결정되고, 여기서, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, Ceil()은 천장 함수이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이다.

[0188] 낮은 지연의 가상 참조 디코더(HRD) 플래그(예컨대, low_delay_hrd_flag)가 1로 설정되고, $t_{r,n}(m) < t_{af}(m)$ 이고, 픽처 타이밍 플래그가 1로 설정되고, CPB가 서브-픽처 레벨에서 동작하고 있을 때, 액세스 유닛의 최후 디코딩 유닛 m에 대한 제거 시간 $t_r(m)$ 은 $t_r(m) = t_{r,n}(m) + \max((t_{c_sub} * Ceil((t_{af}(m) - t_{r,n}(m)) / t_{c_sub})), (t_c * Ceil((t_{af}(n) - t_{r,n}(n)) / t_c)))$ 에 따라 결정되고, 여기서, $t_{r,n}(m)$ 은 최후 디코딩 유닛 m의 명목 제거 시간이고, t_{c_sub} 는 서브-픽처 클록 틱이고, Ceil()은 천장 함수이고, $t_{af}(m)$ 은 최후 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이다.

[0189] 낮은 지연의 가상 참조 디코더(HRD) 플래그가 1로 설정되고, $t_{r,n}(n) < t_{af}(n)$ 이고, 픽처 타이밍 플래그가 1로 설정되고, CPB가 액세스 유닛 레벨에서 동작하고 있을 때, 액세스 유닛 n에 대한 제거 시간 $t_r(n)$ 은 $t_r(n) = t_{r,n}(n) + \max((t_{c_sub} * Ceil((t_{af}(m) - t_{r,n}(m)) / t_{c_sub})), (t_c * Ceil((t_{af}(n) - t_{r,n}(n)) / t_c)))$ 에 따라 결정되고, 여기서, $t_{r,n}(m)$ 은 최후 디코딩 유닛 n의 명목 제거 시간이고, t_{c_sub} 는 서브-픽처 클록 틱이고, Ceil()은 천장 함수이고, $t_{af}(m)$ 은 최후 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이다.

[0190] 낮은 지연의 가상 참조 디코더(HRD) 플래그(예컨대, low_delay_hrd_flag)가 1로 설정되고, $t_{r,n}(m) < t_{af}(m)$ 이고, 픽처 타이밍 플래그가 1로 설정되고, CPB가 서브-픽처 레벨에서 동작하고 있을 때, 액세스 유닛의 최후

디코딩 유닛 m에 대한 제거 시간 $t_r(m)$ 은 $t_r(m) = t_{r,n}(m) + \min((t_{c_sub} * \text{Ceil}((t_{af}(m) - t_{r,n}(m)) / t_{c_sub})), (t_c * \text{Ceil}((t_{af}(n) - t_{r,n}(n)) / t_c)))$ 에 따라 결정되고, 여기서, $t_{r,n}(m)$ 은 최후 디코딩 유닛 m의 명목 제거 시간이고, t_{c_sub} 는 서브-픽처 클록 틱이고, $\text{Ceil}()$ 은 천장 함수이고, $t_{af}(m)$ 은 최후 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이다.

[0191] 낮은 지연의 가상 참조 디코더(HRD) 플래그가 1로 설정되고, $t_{r,n}(n) < t_{af}(n)$ 이고, 픽처 타이밍 플래그가 1로 설정되고, CPB가 액세스 유닛 레벨에서 동작하고 있을 때, 액세스 유닛 n에 대한 제거 시간 $t_r(n)$ 은 $t_r(n) = t_{r,n}(n) + \min((t_{c_sub} * \text{Ceil}((t_{af}(m) - t_{r,n}(m)) / t_{c_sub})), (t_c * \text{Ceil}((t_{af}(n) - t_{r,n}(n)) / t_c)))$ 에 따라 결정되고, 여기서, $t_{r,n}(m)$ 은 최후 디코딩 유닛 n의 명목 제거 시간이고, t_{c_sub} 는 서브-픽처 클록 틱이고, $\text{Ceil}()$ 은 천장 함수이고, $t_{af}(m)$ 은 최후 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이다.

[0192] 낮은 지연의 가상 참조 디코더(HRD) 플래그(예컨대, `low_delay_hrd_flag`)가 1로 설정되고, $t_{r,n}(m) < t_{af}(m)$ 이고, 픽처 타이밍 플래그가 1로 설정되고, CPB가 서브-픽처 레벨에서 동작하고 있을 때, 액세스 유닛의 최후 디코딩 유닛 m에 대한 제거 시간 $t_r(m)$ 은 $t_r(m) = t_{r,n}(m) + (t_c * \text{Ceil}((t_{af}(n) - t_{r,n}(n)) / t_c))$ 에 따라 결정되고, 여기서, $t_{r,n}(m)$ 은 최후 디코딩 유닛 m의 명목 제거 시간이고, t_{c_sub} 는 서브-픽처 클록 틱이고, $\text{Ceil}()$ 은 천장 함수이고, $t_{af}(m)$ 은 최후 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이다.

[0193] 낮은 지연의 가상 참조 디코더(HRD) 플래그가 1로 설정되고, $t_{r,n}(n) < t_{af}(n)$ 이고, 픽처 타이밍 플래그가 1로 설정되고, CPB가 액세스 유닛 레벨에서 동작하고 있을 때, 액세스 유닛 n에 대한 제거 시간 $t_r(n)$ 은 $t_r(n) = t_{r,n}(n) + (t_c * \text{Ceil}((t_{af}(n) - t_{r,n}(n)) / t_c))$ 에 따라 결정되고, 여기서, $t_{r,n}(m)$ 은 최후 디코딩 유닛 n의 명목 제거 시간이고, t_{c_sub} 는 서브-픽처 클록 틱이고, $\text{Ceil}()$ 은 천장 함수이고, $t_{af}(m)$ 은 최후 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이다.

[0194] 낮은 지연의 가상 참조 디코더(HRD) 플래그(예컨대, `low_delay_hrd_flag`)가 1로 설정되고, $t_{r,n}(m) < t_{af}(m)$ 이고, 픽처 타이밍 플래그가 1로 설정되고, CPB가 서브-픽처 레벨에서 동작하고 있을 때, 액세스 유닛의 최후 디코딩 유닛이 아닌 디코딩 유닛에 대한 제거 시간은 $t_r(m) = t_{af}(m)$ 로서 설정되고, 여기서, $t_{af}(m)$ 은 디코딩 유닛 m의 최종 도달 시간이다. 그리고, 액세스 유닛의 최후 디코딩 유닛 m에 대한 제거 시간 $t_r(m)$ 은 $t_r(m) = t_{r,n}(m) + (t_{c_sub} * \text{Ceil}((t_{af}(m) - t_{r,n}(m)) / t_{c_sub}))$ 에 따라 결정되고, 여기서, $t_{r,n}(m)$ 은 최후 디코딩 유닛 m의 명목 제거 시간이고, t_{c_sub} 는 서브-픽처 클록 틱이고, $\text{Ceil}()$ 은 천장 함수이고, $t_{af}(m)$ 은 최후 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이고, $t_{af}(m)$ 은 액세스 유닛 n에서의 최후 디코딩 유닛의 최종 도달 시간이다.

[0195] 낮은 지연의 가상 참조 디코더(HRD) 플래그(예컨대, `low_delay_hrd_flag`)가 1로 설정되고, $t_{r,n}(m) < t_{af}(m)$ 이고, 픽처 타이밍 플래그가 1로 설정되고, CPB가 서브-픽처 레벨에서 동작하고 있을 때, 액세스 유닛의 최후 디코딩 유닛이 아닌 디코딩 유닛에 대한 제거 시간은 $t_r(m) = t_{af}(m)$ 로서 설정되고, 여기서, $t_{af}(m)$ 은 디코딩 유닛 m의 최종 도달 시간이다. 그리고, 액세스 유닛의 최후 디코딩 유닛 m에 대한 제거 시간 $t_r(m)$ 은 $t_r(m) = t_{r,n}(m) + (t_c * \text{Ceil}((t_{af}(m) - t_{r,n}(m)) / t_c))$ 에 따라 결정되고, 여기서, $t_{r,n}(m)$ 은 최후 디코딩 유닛 m의 명목 제거 시간이고, t_{c_sub} 는 서브-픽처 클록 틱이고, $\text{Ceil}()$ 은 천장 함수이고, $t_{af}(m)$ 은 최후 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클록 틱이고, $t_{af}(n)$

)은 액세스 유닛 n의 최종 도달 시간이고, $t_{af}(m)$ 은 액세스 유닛 n에서의 최후 디코딩 유닛의 최종 도달 시간이다.

[0196] 낮은 지연의 가상 참조 디코더(HRD) 플래그(예컨대, `low_delay_hrd_flag`)가 1로 설정되고, $t_{r,n}(m) < t_{af}(m)$ 이고, 픽처 타이밍 플래그가 1로 설정되고, CPB가 서브-픽처 레벨에서 동작하고 있을 때, 디코딩 유닛에 대한 제거 시간은 $t_r(m) = t_{af}(m)$ 로 설정되고, 여기서, $t_{r,n}(m)$ 은 디코딩 유닛 m의 명목 제거 시간이고, $t_{c,sub}$ 는 서브-픽처 클럭 틱이고, `Ceil()`은 천장 함수이고, $t_{af}(m)$ 은 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클럭 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이고, $t_{af}(m)$ 은 액세스 유닛 n에서의 디코딩 유닛의 최종 도달 시간이다.

[0197] 낮은 지연의 가상 참조 디코더(HRD) 플래그가 1로 설정되고, $t_{r,n}(n) < t_{af}(n)$ 이고, 픽처 타이밍 플래그가 1로 설정되고, CPB가 액세스 유닛 레벨에서 동작하고 있을 때, 액세스 유닛 n에 대한 제거 시간 $t_r(n)$ 은 $t_r(n) = t_{af}(n)$ 에 따라 결정되고, 여기서, $t_{r,n}(m)$ 은 최후 디코딩 유닛 n의 명목 제거 시간이고, $t_{c,sub}$ 는 서브-픽처 클럭 틱이고, `Ceil()`은 천장 함수이고, $t_{af}(m)$ 은 최후 디코딩 유닛 m의 최종 도달 시간이고, $t_{r,n}(n)$ 은 액세스 유닛 n의 명목 제거 시간이고, t_c 는 클럭 틱이고, $t_{af}(n)$ 은 액세스 유닛 n의 최종 도달 시간이다.

[0198] CPB가 액세스 유닛 레벨 상에서 동작할 경우, 디코더(112)는 CPB 제거 지연 파라미터를 결정할 수도 있다(412). 이것은 수신된 픽처 타이밍 SEI 메시지(예컨대, `cpb_removal_delay`) 내에 포함될 수도 있다. 디코더(112)는 또한, CPB 제거 지연 파라미터에 기반으로 하여 액세스 유닛을 제거할 수도 있고(414), 액세스 유닛을 디코딩할 수도 있다(416). 다시 말해서, 디코더(112)는 액세스 유닛 내의 디코딩 유닛들이 아니라, 한번에 전체의 액세스 유닛들을 디코딩할 수도 있다.

[0199] 도 5는 액세스 유닛에서의 디코딩 유닛들에 대한 하나 이상의 제거 지연들을 결정하기 위한 방법(500)의 하나의 구성을 예시하는 흐름도이다. 다시 말해서, 도 5에서 예시된 방법(500)은 도 4에서 예시된 방법(400)에서의 단계(406)를 추가로 예시할 수도 있다. 방법(500)은 디코더(112)에 의해 수행될 수도 있다. 디코더(112)는 수신된 픽처 타이밍 SEI 메시지가 공통 디코딩 유닛 CPB 제거 지연 파라미터를 포함하는지 여부를 결정할 수도 있다(502). 이것은 공통 디코딩 유닛 CPB 제거 지연 플래그(예컨대, `common_du_cpb_removal_delay_flag`)가 설정되는지 여부를 결정하는 것을 포함할 수도 있다. 긍정일 경우, 디코더(112)는 액세스 유닛에서의 모든 디코딩 유닛들에 적용가능한 공통 디코딩 유닛 CPB 제거 지연 파라미터(예컨대, `common_du_cpb_removal_delay`)를 결정할 수도 있다(504). 부정일 경우, 디코더(112)는 액세스 유닛에서의 각각의 디코딩 유닛에 대한 별도의 디코딩 유닛 CPB 제거 지연 파라미터들(예컨대, `du_cpb_removal_delay[i]`)을 결정할 수도 있다(506).

[0200] 픽처 타이밍 SEI 메시지 시맨틱들을 수정하는 것에 추가하여, 본 시스템들 및 방법들은 또한, 서브-픽처 기반 CPB 동작 및 액세스 유닛 기반 CPB 동작이 디코딩 유닛 제거의 동일한 타이밍으로 귀착되도록 비트스트림 제약을 부과할 수도 있다. 구체적으로, 픽처 타이밍 플래그(예컨대, `sub_pic_cpb_params_present_flag`)가 1로 설정될 때, CPB 제거 지연은 하기에 따라 설정될 수도 있다.

수학식 6

$$cpb_removal_delay = \frac{\left(\sum_{i=0}^{num_decoding_units_minus1} du_cpb_removal_delay[i] \right) * t_{c,sub}}{t_c}$$

[0201]

[0202] 여기서, `du_cpb_removal_delay[i]`는 디코딩 유닛 CPB 제거 지연 파라미터들이고, t_c 는 클럭 틱이고, $t_{c,sub}$ 는 서브-픽처 클럭 틱이고, `num_decoding_units_minus1`은 1에 의해 오프셋된 액세스 유닛에서의 디코딩 유닛들의 양이고, i 는 인덱스이다.

[0203] 대안적으로, CPB 제거 지연은 다음으로 설명된 바와 같이 설정될 수도 있다: 변수 $T_{du}(k)$ 는 다음과 같이 정의된다고 한다:

수학식 7

$$T_{du}(k) = T_{du}(k-1) + t_{c,sub} * \sum_{j=0}^{num_decoding_units_minus1_k} (du_cpb_removal_delay_minus1_k[i] + 1)$$

[0204]

[0205]

여기서, $du_cpb_removal_delay_minus1_k[i]$ 및 $num_decoding_units_minus1_k$ 은 k 번째 액세스 유닛(HRD를 초기화하였던 액세스 유닛에 대해 $k=0$ 이고, $k < 1$ 에 대하여 $T_{du}(k)=0$)의 I 번째 디코딩 유닛에 대한 파라미터들이고, 여기서, $du_cpb_removal_delay_minus1_k[i] + 1 = du_cpb_removal_delay_minus1_k[i]$ 는 k 번째 액세스 유닛의 I 번째 디코딩 유닛에 대한 디코딩 유닛 CPB 제거 지연 파라미터이고, $num_decoding_units_minus1_k$ 은 k 번째 액세스 유닛에서의 디코딩 유닛들의 수이고, t_c 는 클록 턱이고, $t_{c,sub}$ 는 서브-픽처 클록 턱이고, i 및 k 는 인덱스들이다. 다음으로, 픽처 타이밍 플래그(예컨대, $sub_pic_cpb_params_present_flag$)가 1로 설정될 때, 다음의 조건은 참일 것이다:

[0206]

$(au_cpb_removal_delay_minus1 + 1) * t_c == T_{du}(k)$, 여기서, $(au_cpb_removal_delay_minus1 + 1) = cpb_removal_delay$, CPB 제거 지연. 이에 따라, 이 경우, CPB 제거 지연($au_cpb_removal_delay_minus1 + 1$)은 서브-픽처 기반 CPB 동작 및 액세스 유닛 기반 CPB 동작의 동작이 액세스 유닛 제거의 동일한 타이밍 및 액세스 유닛 제거의 최후 디코딩 유닛으로 귀착되도록 설정된다.

[0207]

대안적으로, CPB 제거 지연은 하기에 따라 설정될 수도 있다.

수학식 8

$$-1 \leq [cpb_removal_delay * t_c - (\sum_{i=0}^{num_decoding_units_minus1} du_cpb_removal_delay[i] * t_{c,sub})] \leq 1$$

[0208]

[0209]

여기서, $du_cpb_removal_delay[i]$ 는 디코딩 유닛 CPB 제거 지연 파라미터들이고, t_c 는 클록 턱이고, $t_{c,sub}$ 는 서브-픽처 클록 턱이고, $num_decoding_units_minus1$ 은 1에 의해 오프셋된 액세스 유닛에서의 디코딩 유닛들의 양이고, i 는 인덱스이다.

[0210]

대안적으로, $cpb_removal_delay$ 및 $du_cpb_removal_delay[num_decoding_units_minus1]$ 은 $cpb_removal_delay * t_c = du_cpb_removal_delay[num_decoding_units_minus1] * t_{c,sub}$ 에 따라 설정될 수도 있고, 여기서, $du_cpb_removal_delay[num_decoding_units_minus1]$ 은 $num_decoding_units_minus1$ 번째 디코딩 유닛에 대한 디코딩 유닛 CPB 제거 지연 파라미터이고, t_c 는 클록 턱이고, $t_{c,sub}$ 는 서브-픽처 클록 턱이고, $num_decoding_units_minus1$ 은 1에 의해 오프셋된 액세스 유닛에서의 디코딩 유닛들의 양이다.

[0211]

픽처 타이밍 SEI 메시지 시맨틱들을 수정하는 것에 추가하여, 본 시스템들 및 방법들은 또한, 서브-픽처 기반 CPB 동작 및 액세스 유닛 기반 CPB 동작이 디코딩 유닛 제거의 동일한 타이밍으로 귀착되도록 비트스트림 제약을 부과할 수도 있다. 구체적으로, 픽처 타이밍 플래그(예컨대, $sub_pic_cpb_params_present_flag$)가 1로 설정될 때, $cpb_removal_delay$ 및 $du_cpb_removal_delay[num_decoding_units_minus1]$ 에 대한 값들은 하기를 만족시키도록 설정될 수도 있다: $-1 \leq (cpb_removal_delay * t_c - du_cpb_removal_delay[num_decoding_units_minus1] * t_{c,sub}) \leq 1$ 이고, 여기서, $du_cpb_removal_delay[num_decoding_units_minus1]$ 은 $num_decoding_units_minus1$ 번째 디코딩 유닛에 대한 디코딩 유닛 CPB 제거 지연 파라미터이고, t_c 는 클록 턱이고, $t_{c,sub}$ 는 서브-픽처 클록 턱이고, $num_decoding_units_minus1$ 은 1에 의해 오프셋된 액세스 유닛에서의 디코딩 유닛들의 양이다.

[0212]

도 6a는 전자 디바이스(602) 상의 인코더(604)의 하나의 구성을 예시하는 블록도이다. 전자 디바이스(602) 내에 포함된 것으로서 예시된 엘리먼트들 중의 하나 이상은 하드웨어, 소프트웨어, 또는 양자의 조합으로 구현될 수도 있다는 것에 주목해야 한다. 예를 들어, 전자 디바이스(602)는 하드웨어, 소프트웨어, 또는 양자의 조합

으로 구현될 수도 있는 인코더(604)를 포함한다. 예를 들어, 인코더(604)는 회로, 집적 회로, 애플리케이션-특정 집적 회로(ASIC), 실행가능한 명령들을 갖는 메모리와 전자 통신하는 프로세서, 펌웨어, 필드-프로그래밍가능한 게이트 어레이(field-programmable gate array; FPGA) 등, 또는 그 조합으로서 구현될 수도 있다. 일부의 구성들에서, 인코더(604)는 HEVC 코더일 수도 있다.

- [0213] 전자 디바이스(602)는 소스(622)를 포함할 수도 있다. 소스(622)는 픽처 또는 이미지 데이터(예컨대, 비디오)를 하나 이상의 입력 픽처들(606)로서 인코더(604)에 제공할 수도 있다. 소스(622)의 예들은 이미지 센서들, 메모리, 통신 인터페이스들, 네트워크 인터페이스들, 무선 수신기들, 포트들 등을 포함할 수도 있다.
- [0214] 하나 이상의 입력 픽처들(606)은 인트라-프레임 예측 모듈 및 재구성 버퍼(624)에 제공될 수도 있다. 입력 픽처(606)는 또한, 모션 추정 및 모션 보상 모듈(646)과, 감산 모듈(628)에 제공될 수도 있다.
- [0215] 인트라-프레임 예측 모듈 및 재구성 버퍼(624)는 하나 이상의 입력 픽처들(606) 및 재구성된 데이터(660)에 기반으로 하여 인트라 모드(intra mode) 정보(640) 및 인트라 신호(intra signal)(626)를 생성할 수도 있다. 모션 추정 및 모션 보상 모듈(646)은 하나 이상의 입력 픽처들(606) 및 디코딩된 픽처 버퍼(676)로부터의 참조 픽처(678)에 기반으로 하여 인터 모드(inter mode) 정보(648) 및 인터 신호(inter signal)(644)를 생성할 수도 있다. 일부의 구성들에서, 디코딩된 픽처 버퍼(676)는 디코딩된 픽처 버퍼(676)에서의 하나 이상의 참조 픽처들로부터의 데이터를 포함할 수도 있다.
- [0216] 인코더(604)는 모드에 따라 인트라 신호(626) 및 인터 신호(644) 사이에서 선택할 수도 있다. 인트라 신호(626)는 인트라-코딩 모드에서 픽처 내의 공간적 특징들을 활용하기 위하여 이용될 수도 있다. 인트라 신호(644)는 인터 코딩 모드에서 픽처들 사이의 시간적 특징들을 활용하기 위하여 이용될 수도 있다. 인트라 코딩 모드에 있는 동안에는, 인트라 신호(626)가 감산 모듈(628)에 제공될 수도 있고, 인트라 모드 정보(640)는 엔트로피 코딩 모듈(642)에 제공될 수도 있다. 인터 코딩 모드에 있는 동안에는, 인터 신호(644)가 감산 모듈(628)에 제공될 수도 있고, 인터 모드 정보(648)는 엔트로피 코딩 모듈(642)에 제공될 수도 있다.
- [0217] 예측 잔차(prediction residual)(630)를 생성하기 위하여, (모드에 따라) 인트라 신호(626) 또는 인터 신호(644)의 어느 하나가 감산 모듈(628)에서의 입력 픽처(606)로부터 감산된다. 예측 잔차(630)는 변환 모듈(632)에 제공된다. 변환 모듈(632)은 양자화 모듈(636)에 제공되는 변환된 신호(634)를 생성하기 위하여 예측 잔차(630)를 압축할 수도 있다. 양자화 모듈(636)은 변환 및 양자화된 계수(transformed and quantized coefficient; TQC)들(638)을 생성하기 위하여 변환된 신호(634)를 양자화한다.
- [0218] TQC들(638)은 엔트로피 코딩 모듈(642) 및 역양자화 모듈(650)에 제공된다. 역양자화 모듈(650)은 역변환 모듈(654)에 제공되는 역양자화된 신호(652)를 생성하기 위하여 TQC들(638)에 대한 역양자화를 수행한다. 역변환 모듈(654)은 재구성 모듈(658)에 제공되는 압축해제된 신호(656)를 생성하기 위하여 역양자화된 신호(652)를 압축해제한다.
- [0219] 재구성 모듈(658)은 압축해제된 신호(656)에 기반으로 하여 재구성된 데이터(660)를 생성할 수도 있다. 예를 들어, 재구성 모듈(658)은 (수정된) 픽처들을 재구성할 수도 있다. 재구성된 데이터(660)는 디블록킹 필터(deblocking filter)(662), 그리고 인트라 예측 모듈 및 재구성 버퍼(624)에 제공될 수도 있다. 디블록킹 필터(662)는 재구성된 데이터(660)에 기반으로 하여 필터링된 신호(664)를 생성할 수도 있다.
- [0220] 필터링된 신호(664)는 샘플 적응적 오프셋(sample adaptive offset; SAO) 모듈(666)에 제공될 수도 있다. SAO 모듈(666)은 엔트로피 코딩 모듈(642)에 제공되는 SAO 정보(668)와, 적응적 루프 필터(adaptive loop filter; ALF)(672)에 제공되는 SAO 신호(670)를 생성할 수도 있다. ALF(672)는 디코딩된 픽처 버퍼(676)에 제공되는 ALF 신호(674)를 생성한다. ALF 신호(674)는 참조 픽처들로서 이용될 수도 있는 하나 이상의 픽처들로부터의 데이터를 포함할 수도 있다.
- [0221] 엔트로피 코딩 모듈(642)은 비트스트림 A(614a)(예컨대, 인코딩된 픽처 데이터)를 생성하기 위하여 TQC들(638)을 코딩할 수도 있다. 예를 들어, 엔트로피 코딩 모듈(642)은 컨텍스트-적응적 가변 길이 코딩(Context-Adaptive Variable Length Coding; CAVLC) 또는 컨텍스트-적응적 이진 산술 코딩(Context-Adaptive Binary Arithmetic Coding; CABAC)을 이용하여 TQC들(638)을 코딩할 수도 있다. 특히, 엔트로피 코딩 모듈(642)은 인트라 모드 정보(640), 인터 모드 정보(648) 및 SAO 정보(668) 중의 하나 이상에 기반으로 하여 TQC들(638)을 코딩할 수도 있다. 비트스트림 A(614a)(예컨대, 인코딩된 픽처 데이터)는 메시지 생성 모듈(608)에 제공될 수도 있다. 메시지 생성 모듈(608)은 도 1과 관련하여 설명된 메시지 생성 모듈(108)과 유사하게 구성될 수도 있다. 추가적으로 또는 대안적으로, 메시지 생성 모듈(608)은 도 2 및 도 3과 관련하여 설명된 절차들 중의 하나 이상

을 수행할 수도 있다.

- [0222] 예를 들어, 메시지 생성 모듈(608)은 서브-픽처 파라미터들을 포함하는 메시지(예컨대, 픽처 타이밍 SEI 메시지 또는 다른 메시지)를 생성할 수도 있다. 서브-픽처 파라미터들은 디코딩 유닛들에 대한 하나 이상의 제거 지연들(예컨대, common_du_cpb_removal_delay 또는 du_cpb_removal_delay[i]) 및 하나 이상의 NAL 파라미터들(예컨대, common_num_nalus_in_du_minus1 또는 num_nalus_in_du_minus1[i])을 포함할 수도 있다. 일부의 구성들에서는, 비트스트림 B(614b)를 생성하기 위하여 메시지가 비트스트림 A(614a) 내로 삽입될 수도 있다. 이에 따라, 예를 들어, 전체의 비트스트림 A(614a)가 생성된 후에(예컨대, 비트스트림 B(614b)의 대부분이 생성된 후에), 메시지가 생성될 수도 있다. 다른 구성들에서, 메시지는 비트스트림 A(614a) 내로 삽입되는 것이 아니라(이 경우, 비트스트림 B(614b)가 비트스트림 A(614a)와 동일할 수도 있음), 별도의 송신(610)에서 제공될 수도 있다.
- [0223] 일부의 구성들에서, 전자 디바이스(602)는 비트스트림(614)을 또 다른 전자 디바이스로 전송한다. 예를 들어, 비트스트림(614)은 통신 인터페이스, 네트워크 인터페이스, 무선 송신기, 포트 등에 제공될 수도 있다. 예를 들어, 비트스트림(614)은 LAN, 인터넷, 셀룰러 전화 기지국 등을 통해 또 다른 전자 디바이스로 송신될 수도 있다. 비트스트림(614)은 추가적으로 또는 대안적으로, 전자 디바이스(602) 상의 메모리 또는 다른 컴포넌트들 내에 저장될 수도 있다.
- [0224] 도 6b는 전자 디바이스(1702) 상의 비디오 인코더(1782)의 하나의 구성을 예시하는 블록도이다. 비디오 인코더(1782)는 강화 계층 인코더(1706), 기본 계층 인코더(1709), 해상도 업스케일링(resolution upscaling) 블록(1770), 및 출력 인터페이스(1780)를 포함할 수도 있다. 예를 들어, 도 6b의 비디오 인코더는 본원에서 설명된 바와 같이, 스케일러블 비디오 코딩 및 멀티-뷰 비디오 코딩을 위해 적당하다.
- [0225] 강화 계층 인코더(1706)는 입력 픽처(1704)를 수신하는 비디오 입력부(1781)를 포함할 수도 있다. 비디오 입력부(1781)의 출력은 예측 선택부(1750)의 출력을 수신하는 가산기/감산기(1783)에 제공될 수도 있다. 가산기/감산기(1783)의 출력은 변환 및 양자화 블록(1752)에 제공될 수도 있다. 변환 및 양자화 블록(1752)의 출력은 엔트로피 인코딩 블록(1748) 및 스케일링 및 역변환 블록(1772)에 제공될 수도 있다. 엔트로피 인코딩(1748)이 수행된 후, 엔트로피 인코딩 블록(1748)의 출력은 출력 인터페이스(1780)에 제공될 수도 있다. 출력 인터페이스(1780)는 인코딩된 기본 계층 비디오 비트스트림(1707) 및 인코딩된 강화 계층 비디오 비트스트림(1710)의 양자를 출력할 수도 있다.
- [0226] 스케일링 및 역변환 블록(1772)의 출력은 가산기(1779)에 제공될 수도 있다. 가산기(1779)는 또한 예측 선택부(1750)의 출력을 수신할 수도 있다. 가산기(1779)의 출력은 디블록킹 블록(1751)에 제공될 수도 있다. 디블록킹 블록(1751)의 출력은 참조 버퍼(1794)에 제공될 수도 있다. 참조 버퍼(1794)의 출력은 모션 보상 블록(1754)에 제공될 수도 있다. 모션 보상 블록(1754)의 출력은 예측 선택부(1750)에 제공될 수도 있다. 참조 버퍼(1794)의 출력은 또한, 인트라 예측기(1756)에 제공될 수도 있다. 인트라 예측기(1756)의 출력은 예측 선택부(1750)에 제공될 수도 있다. 예측 선택부(1750)는 또한, 해상도 업스케일링 블록(1770)의 출력을 수신할 수도 있다.
- [0227] 기본 계층 인코더(1709)는, 다운샘플링된 입력 픽처, 또는 또 다른 이미지와 조합하기 위해 적당한 다른 이미지 콘텐츠, 또는 대안적인 뷰 입력 픽처 또는 동일한 입력 픽처(1703)(즉, 강화 계층 인코더(1706)에 의해 수신된 입력 픽처(1704)와 동일함)를 수신하는 비디오 입력부(1762)를 포함할 수도 있다. 비디오 입력부(1762)의 출력은 인코딩 예측 루프(1764)에 제공될 수도 있다. 엔트로피 인코딩(1766)은 인코딩 예측 루프(1764)의 출력 상에 제공될 수도 있다. 인코딩 예측 루프(1764)의 출력은 또한, 참조 버퍼(1768)에 제공될 수도 있다. 참조 버퍼(1768)는 피드백을 인코딩 예측 루프(1764)에 제공할 수도 있다. 참조 버퍼(1768)의 출력은 또한, 해상도 업스케일링 블록(1770)에 제공될 수도 있다. 일단 엔트로피 인코딩(1766)이 수행되었으면, 출력은 출력 인터페이스(178)에 제공될 수도 있다. 인코딩된 기본 계층 비디오 비트스트림(1707) 및/또는 인코딩된 강화 계층 비디오 비트스트림(1710)은 희망하는 바와 같이, 하나 이상의 메시지 생성 모듈들에 제공될 수도 있다.
- [0228] 도 7a는 전자 디바이스(702) 상의 디코더(712)의 하나의 구성을 예시하는 블록도이다. 디코더(712)는 전자 디바이스(702) 내에 포함될 수도 있다. 예를 들어, 디코더(712)는 HEVC 디코더일 수도 있다. 디코더(712)와, 디코더(712) 내에 포함된 것으로서 예시된 엘리먼트들 중의 하나 이상은 하드웨어, 소프트웨어, 또는 양자의 조합으로 구현될 수도 있다. 디코더(712)는 디코딩을 위한 비트스트림(714)(예컨대, 비트스트림(714) 내에 포함된 하나 이상의 인코딩된 픽처들 및 오버헤드 데이터)을 수신할 수도 있다. 일부의 구성들에서, 수신된 비트스트림(714)은 메시지(예컨대, 픽처 타이밍 SEI 메시지 또는 다른 메시지), 슬라이스 헤더, PPS 등과 같은 수신된

오버헤드 데이터를 포함할 수도 있다. 일부의 구성들에서, 디코더(712)는 추가적으로 별도의 송신(710)을 수신할 수도 있다. 별도의 송신(710)은 메시지(예컨대, 픽처 타이밍 SEI 메시지 또는 다른 메시지)를 포함할 수도 있다. 예를 들어, 픽처 타이밍 SEI 메시지 또는 다른 메시지는 비트스트림(714) 대신에, 별도의 송신(710)에서 수신될 수도 있다. 그러나, 별도의 송신(710)은 선택적일 수도 있으며 일부의 구성들에서 사용되지 않을 수도 있다는 것에 주목해야 한다.

- [0229] 디코더(712)는 CPB(720)를 포함한다. CPB(720)는 상기 도 1과 관련하여 설명된 CPB(120)와 유사하게 구성될 수도 있다. 추가적으로 또는 대안적으로, 디코더(712)는 도 4 및 도 5와 관련하여 설명된 절차들 중의 하나 이상을 수행할 수도 있다. 예를 들어, 디코더(712)는 서브-픽처 파라미터들을 갖는 메시지(예컨대, 픽처 타이밍 SEI 메시지 또는 다른 메시지)를 수신할 수도 있고, 서브-픽처 파라미터들에 기반으로 하여 액세스 유닛에서의 디코딩 유닛들을 제거하고 디코딩할 수도 있다. 하나 이상의 액세스 유닛들은 비트스트림 내에 포함될 수도 있으며 인코딩된 픽처 데이터 및 오버헤드 데이터 중의 하나 이상을 포함할 수도 있다는 것에 주목해야 한다.
- [0230] 코딩된 픽처 버퍼(CPB)(720)는 인코딩된 픽처 데이터를 엔트로피 디코딩 모듈(701)에 제공할 수도 있다. 인코딩된 픽처 데이터는 엔트로피 디코딩 모듈(701)에 의해 엔트로피 디코딩될 수도 있음으로써, 모션 보상 신호(703) 및 양자화되고, 스케일링되고 및/또는 변환된 계수들(705)을 생성할 수도 있다.
- [0231] 모션 정보 신호(703)는, 인터-프레임 예측 신호(782)를 생성할 수도 있는 모션 보상 모듈(780)에서 디코딩된 픽처 버퍼(709)로부터의 참조 프레임 신호(798)의 부분과 조합될 수도 있다. 양자화되고, 디스케일링되고 (descaled), 및/또는 변환된 계수들(705)은 역 모듈(inverse module)(707)에 의해 역양자화될 수도 있고, 스케일링될 수도 있고, 역변환될 수도 있음으로써, 디코딩된 잔차 신호(784)를 생성할 수도 있다. 디코딩된 잔차 신호(784)는 조합된 신호(786)를 생성하기 위하여 예측 신호(792)에 추가될 수도 있다. 예측 신호(792)는 모션 보상 모듈(780)에 의해 생성된 인터-프레임 예측 신호(782), 또는 인트라-프레임 예측 모듈(788)에 의해 생성된 인트라-프레임 예측 신호(790)로부터 선택된 신호일 수도 있다. 일부의 구성들에서, 이 신호 선택은 비트스트림(714)에 기반(예컨대, 이에 의해 제어됨)할 수도 있다.
- [0232] 인트라-프레임 예측 신호(790)는 (예를 들어, 현재의 프레임에서의) 조합된 신호(786)로부터의 이전에 디코딩된 정보로부터 예측될 수도 있다. 조합된 신호(786)는 또한, 디블록킹 필터(794)에 의해 필터링될 수도 있다. 결과적인 필터링된 신호(796)는 디코딩된 픽처 버퍼(709)에 기록될 수도 있다. 결과적인 필터링된 신호(796)는 디코딩된 픽처를 포함할 수도 있다. 디코딩된 픽처 버퍼(709)는 출력(718)될 수도 있는 디코딩된 픽처를 제공할 수도 있다. 일부의 경우에는, 709가 프레임 메모리로서 간주될 수도 있다.
- [0233] 도 7b는 전자 디바이스(1802) 상의 비디오 디코더(1812)의 하나의 구성을 예시하는 블록도이다. 비디오 디코더(1812)는 강화 계층 디코더(1815) 및 기본 계층 디코더(1813)를 포함할 수도 있다. 비디오 디코더(812)는 또한, 인터페이스(1889) 및 해상도 업스케일링부(1870)를 포함할 수도 있다. 예를 들어, 도 7b의 비디오 디코더는 본원에서 설명된 바와 같이, 스케일러를 비디오 디코딩 및 멀티-뷰 비디오 디코딩을 위해 적당하다.
- [0234] 인터페이스(1889)는 인코딩된 비디오 스트림(1885)을 수신할 수도 있다. 인코딩된 비디오 스트림(1885)은 기본 계층 인코딩된 비디오 스트림 및 강화 계층 인코딩된 비디오 스트림으로 구성될 수도 있다. 이 2 개의 스트림들은 별도로 또는 함께 전송될 수도 있다. 인터페이스(1889)는 인코딩된 비디오 스트림(1885)의 일부 또는 전부를 기본 계층 디코더(1813)에서의 엔트로피 디코딩 블록(1886)에 제공할 수도 있다. 엔트로피 디코딩 블록(1886)의 출력은 디코딩 예측 루프(1887)에 제공될 수도 있다. 디코딩 예측 루프(1887)의 출력은 참조 버퍼(1888)에 제공될 수도 있다. 참조 버퍼는 피드백을 디코딩 예측 루프(1887)에 제공할 수도 있다. 참조 버퍼(1888)는 또한, 디코딩된 기본 계층 비디오 스트림(1884)을 출력할 수도 있다.
- [0235] 인터페이스(1889)는 또한, 인코딩된 비디오 스트림(1885)의 일부 또는 전부를 강화 계층 디코더(1815)에서의 엔트로피 디코딩 블록(1890)에 제공할 수도 있다. 엔트로피 디코딩 블록(1890)의 출력은 역양자화 블록(1891)에 제공될 수도 있다. 역양자화 블록(1891)의 출력은 가산기(1892)에 제공될 수도 있다. 가산기(1892)는 역양자화 블록(1891)의 출력 및 예측 선택 블록(1895)의 출력을 가산할 수도 있다. 가산기(1892)의 출력은 디블록킹 블록(1893)에 제공될 수도 있다. 디블록킹 블록(1893)의 출력은 참조 버퍼(1894)에 제공될 수도 있다. 참조 버퍼(1894)는 디코딩된 강화 계층 비디오 스트림(1882)을 출력할 수도 있다. 참조 버퍼(1894)의 출력은 또한, 인트라 예측기(1897)에 제공될 수도 있다. 강화 계층 디코더(1815)는 모션 보상부(1896)를 포함할 수도 있다. 모션 보상부(1896)는 해상도 업스케일링부(1870) 이후에 수행될 수도 있다. 예측 선택 블록(1895)은 인트라 예측기(1897)의 출력 및 모션 보상부(1896)의 출력을 수신할 수도 있다. 또한, 디코더는 예컨대, 인터페이스(1889)와 함께, 희망하는 바와 같이 하나 이상의 코딩된 픽처 버퍼들을 포함할 수도 있다.

- [0236] 도 8은 송신 전자 디바이스(802)에서 사용될 수도 있는 다양한 컴포넌트들을 예시한다. 본원에서 설명된 전자 디바이스들(102, 602, 702) 중의 하나 이상은 도 8에서 예시된 송신 전자 디바이스(802)에 따라 구현될 수도 있다.
- [0237] 송신 전자 디바이스(802)는 전자 디바이스(802)의 동작을 제어하는 프로세서(817)를 포함한다. 프로세서(817)는 또한 CPU로서 지칭될 수도 있다. 판독-전용 메모리(ROM), 랜덤 액세스 메모리(RAM), 또는 정보를 저장할 수도 있는 임의의 타입의 디바이스를 포함할 수도 있는 메모리(811)는 명령들(813a)(예컨대, 실행가능한 명령들) 및 데이터(815a)를 프로세서(817)에 제공한다. 메모리(811)의 부분은 또한, 비휘발성 랜덤 액세스 메모리(non-volatile random access memory; NVRAM)를 포함할 수도 있다. 메모리(811)는 프로세서(817)와 전자 통신할 수도 있다.
- [0238] 명령들(813b) 및 데이터(815b)는 또한, 프로세서(817)에서 상주할 수도 있다. 프로세서(817)로 로딩된 명령들(813b) 및/또는 데이터(815b)는 또한, 프로세서(817)에 의한 실행 또는 프로세싱을 위해 로딩되었던 명령들(813a) 및/또는 데이터(815a)를 포함할 수도 있다. 명령들(813b)은 본원에서 개시된 시스템들 및 방법들을 구현하기 위하여 프로세서(817)에 의해 실행될 수도 있다. 예를 들어, 명령들(813b)은 위에서 설명된 방법들(200, 300, 400, 500) 중의 하나 이상을 수행하도록 실행가능할 수도 있다.
- [0239] 송신 전자 디바이스(802)는 다른 전자 디바이스들(예컨대, 수신 전자 디바이스)과 통신하기 위한 하나 이상의 통신 인터페이스들(819)을 포함할 수도 있다. 통신 인터페이스들(819)은 유선 통신 기술, 무선 통신 기술, 또는 양자에 기반으로 할 수도 있다. 통신 인터페이스(819)의 예들은 직렬 포트, 병렬 포트, 범용 직렬 버스(Universal Serial Bus; USB), 이더넷 어댑터(Ethernet adapter), IEEE 1394 버스 인터페이스, 소형 컴퓨터 시스템 인터페이스(small computer system interface; SCSI) 버스 인터페이스, 적외선(infrared; IR) 통신 포트, 블루투스(Bluetooth) 무선 통신 어댑터, 3세대 파트너십 프로젝트(3rd Generation Partnership Project; 3GPP) 사양들에 따른 무선 트랜시버, 기타 등등을 포함한다.
- [0240] 송신 전자 디바이스(802)는 하나 이상의 출력 디바이스들(823) 및 하나 이상의 입력 디바이스들(821)을 포함할 수도 있다. 출력 디바이스들(823)의 예들은 스피커, 프린터 등을 포함한다. 전자 디바이스(802)내에 포함될 수도 있는 하나의 타입의 출력 디바이스는 디스플레이 디바이스(825)이다. 본원에서 개시된 구성들과 함께 이용된 디스플레이 디바이스들(825)은 음극선관(cathode ray tube; CRT), 액정 디스플레이(liquid crystal display; LCD), 발광 다이오드(light-emitting diode; LED), 가스 플라즈마, 전계발광(electroluminescence) 등과 같은 임의의 적당한 이미지 투영 기술을 사용할 수도 있다. 디스플레이 제어기(827)는 메모리(811) 내에 저장된 데이터를 디스플레이(825) 상에 나타낸 텍스트, 그래픽, 및/또는 동영상들로 (적절한 바와 같이) 변환하기 위하여 제공될 수도 있다. 입력 디바이스들(821)의 예들은 키보드, 마우스, 마이크로폰, 원격 제어 디바이스, 버튼(button), 조이스틱(joystick), 트랙볼(trackball), 터치패드(touchpad), 터치스크린(touchscreen), 라이트펜(lightpen) 등을 포함한다.
- [0241] 송신 전자 디바이스(802)의 다양한 컴포넌트들은, 데이터 버스에 추가하여, 전원 버스, 제어 신호 버스, 및 상태 신호 버스를 포함할 수도 있는 버스 시스템(829)에 의해 함께 결합된다. 그러나, 간결성을 위하여, 다양한 버스들은 도 8에서 버스 시스템(829)으로서 예시되어 있다. 도 8에서 예시된 송신 전자 디바이스(802)는 특정한 컴포넌트들의 리스트가 아니라 기능적 블록도이다.
- [0242] 도 9는 수신 전자 디바이스(902)에서 사용될 수도 있는 다양한 컴포넌트들을 예시하는 블록도이다. 본원에서 설명된 전자 디바이스들(102, 602, 702) 중의 하나 이상은 도 9에서 예시된 수신 전자 디바이스(902)에 따라 구현될 수도 있다.
- [0243] 수신 전자 디바이스(902)는 전자 디바이스(902)의 동작을 제어하는 프로세서(917)를 포함한다. 프로세서(917)는 또한 CPU로서 지칭될 수도 있다. 판독-전용 메모리(ROM), 랜덤 액세스 메모리(RAM), 또는 정보를 저장할 수도 있는 임의의 타입의 디바이스를 포함할 수도 있는 메모리(911)는 명령들(913a)(예컨대, 실행가능한 명령들) 및 데이터(915a)를 프로세서(917)에 제공한다. 메모리(911)의 부분은 또한, 비휘발성 랜덤 액세스 메모리(NVRAM)를 포함할 수도 있다. 메모리(911)는 프로세서(917)와 전자 통신할 수도 있다.
- [0244] 명령들(813b) 및 데이터(915b)는 또한, 프로세서(917)에서 상주할 수도 있다. 프로세서(917)로 로딩된 명령들(913b) 및/또는 데이터(915b)는 또한, 프로세서(917)에 의한 실행 또는 프로세싱을 위해 로딩되었던 명령들(913a) 및/또는 데이터(915a)를 포함할 수도 있다. 명령들(913b)은 본원에서 개시된 시스템들 및 방법들을 구현하기 위하여 프로세서(917)에 의해 실행될 수도 있다. 예를 들어, 명령들(913b)은 위에서 설명된 방법들

(200, 300, 400, 500) 중의 하나 이상을 수행하도록 실행가능할 수도 있다.

- [0245] 수신 전자 디바이스(902)는 다른 전자 디바이스들(예컨대, 송신 전자 디바이스)과 통신하기 위한 하나 이상의 통신 인터페이스들(919)을 포함할 수도 있다. 통신 인터페이스(919)는 유선 통신 기술, 무선 통신 기술, 또는 양자에 기반으로 할 수도 있다. 통신 인터페이스(919)의 예들은 직렬 포트, 병렬 포트, 범용 직렬 버스(USB), 이더넷 어댑터, IEEE 1394 버스 인터페이스, 소형 컴퓨터 시스템 인터페이스(SCSI) 버스 인터페이스, 적외선 (IR) 통신 포트, 블루투스 무선 통신 어댑터, 3세대 파트너십 프로젝트(3GPP) 사양들에 따른 무선 트랜시버, 기타 등등을 포함한다.
- [0246] 수신 전자 디바이스(902)는 하나 이상의 출력 디바이스들(923) 및 하나 이상의 입력 디바이스들(921)을 포함할 수도 있다. 출력 디바이스들(923)의 예들은 스피커, 프린터 등을 포함한다. 전자 디바이스(902)내에 포함될 수도 있는 하나의 타입의 출력 디바이스는 디스플레이 디바이스(925)이다. 본원에서 개시된 구성들과 함께 이용된 디스플레이 디바이스들(925)은 음극선관(CRT), 액정 디스플레이(LCD), 발광 다이오드(LED), 가스 플라즈마, 전계발광 등과 같은 임의의 적당한 이미지 투영 기술을 사용할 수도 있다. 디스플레이 제어기(927)는 메모리(911) 내에 저장된 데이터를 디스플레이(925) 상에 나타낸 텍스트, 그래픽, 및/또는 동영상들로 (적절한 바와 같이) 변환하기 위하여 제공될 수도 있다. 입력 디바이스들(921)의 예들은 키보드, 마우스, 마이크로폰, 원격 제어 디바이스, 버튼, 조이스틱, 트랙볼, 터치패드, 터치스크린, 라이트펜 등을 포함한다.
- [0247] 수신 전자 디바이스(902)의 다양한 컴포넌트들은, 데이터 버스에 추가하여, 전원 버스, 제어 신호 버스, 및 상태 신호 버스를 포함할 수도 있는 버스 시스템(929)에 의해 함께 결합된다. 그러나, 간결성을 위하여, 다양한 버스들은 도 9에서 버스 시스템(929)으로서 예시되어 있다. 도 9에서 예시된 수신 전자 디바이스(902)는 특정한 컴포넌트들의 리스트가 아니라 기능적 블록도이다.
- [0248] 도 10은 메시지를 전송하기 위한 시스템들 및 방법들이 구현될 수도 있는 전자 디바이스(1002)의 하나의 구성을 예시하는 블록도이다. 전자 디바이스(1002)는 인코딩 수단(1031) 및 송신 수단(1033)을 포함한다. 인코딩 수단(1031) 및 송신 수단(1033)은 상기 도 1, 도 2, 도 3, 도 6 및 도 8 중의 하나 이상과 관련하여 설명된 기능들 중의 하나 이상을 수행하도록 구성될 수도 있다. 예를 들어, 인코딩 수단(1031) 및 송신 수단(1033)은 비트 스트림(1014)을 생성할 수도 있다. 상기 도 8은 도 10의 구체적인 장치 구조의 하나의 예를 예시한다. 다른 다양한 구조들은 도 1, 도 2, 도 3, 도 6 및 도 8의 기능들 중의 하나 이상을 실현하도록 구현될 수도 있다. 예를 들어, DSP는 소프트웨어에 의해 실현될 수도 있다.
- [0249] 도 11은 비트스트림(1114)을 버퍼링하기 위한 시스템들 및 방법들이 구현될 수도 있는 전자 디바이스(1102)의 하나의 구성을 예시하는 블록도이다. 전자 디바이스(1102)는 수신 수단(1135) 및 디코딩 수단(1137)을 포함할 수도 있다. 수신 수단(1135) 및 디코딩 수단(1137)은 상기 도 1, 도 4, 도 5, 도 7 및 도 9 중의 하나 이상과 관련하여 설명된 기능들 중의 하나 이상을 수행하도록 구성될 수도 있다. 예를 들어, 수신 수단(1135) 및 디코딩 수단(1137)은 비트스트림(1114)을 수신할 수도 있다. 상기 도 9는 도 11의 구체적인 장치 구조의 하나의 예를 예시한다. 다른 다양한 구조들은 도 1, 도 4, 도 5, 도 7 및 도 9의 기능들 중의 하나 이상을 실현하도록 구현될 수도 있다. 예를 들어, DSP는 소프트웨어에 의해 실현될 수도 있다.
- [0250] 도 12는 디코딩된 픽처 버퍼(DPB)의 동작을 위한 방법(1200)의 하나의 구성을 예시하는 흐름도이다. 방법(1200)은 인코더(104) 또는 그 하위-부분들 중의 하나(예컨대, 디코딩된 픽처 버퍼 모듈(676))에 의해 수행될 수도 있다. 방법(1200)은 전자 디바이스(102)(예컨대, 전자 디바이스 B(102b))에서 디코더(112)에 의해 수행될 수도 있다. 추가적으로 또는 대안적으로, 방법(1200)은 디코더(712) 또는 그 하위-부분들 중의 하나(예컨대, 디코딩된 픽처 버퍼 모듈(709))에 의해 수행될 수도 있다. 디코더는 픽처(1202)의 최초 슬라이스 헤더를 파싱할 수도 있다. 현재의 픽처의 디코딩 전의(그러나, 현재의 픽처의 최초 슬라이스의 슬라이스 헤더를 파싱한 후의) DPB로부터의 픽처들의 출력 및 제거는, 현재의 픽처를 포함하는 액세스 유닛의 최초 디코딩 유닛이 CPB로부터 제거될 때에 순간적으로 발생하고 다음과 같이 진행된다.
- [0251] - 참조 픽처 세트(reference picture set; RPS)에 대한 디코딩 프로세스가 호출된다. 참조 픽처 세트는, 연관된 픽처 또는 디코딩 순서에서 연관된 픽처를 후행하는 임의의 픽처의 인터 예측을 위해 이용될 수도 있는, 디코딩 순서에서 연관된 픽처 이전에 있는 모든 참조 픽처들로 구성되는, 픽처와 연관된 참조 픽처들의 세트이다.
- [0252] - 비디오의 비트스트림은 네트워크 추상화 계층(NAL) 유닛들로서 일반적으로 지칭된 논리적 데이터 패킷들로 배치되는 신택스 구조를 포함할 수도 있다. 각각의 NAL 유닛은 연관된 데이터 페이로드의 목적을 식별하기 위하여, 2-바이트 NAL 유닛 헤더(예컨대, 16 비트들)와 같은 NAL 유닛 헤더를 포함한다. 예를 들어, 각각의 코딩된

슬라이스(및/또는 픽처)는 하나 이상의 슬라이스(및/또는 픽처) NAL 유닛들에서 코딩될 수도 있다. 다른 NAL 유닛들은 예를 들어, 보충 강화 정보, 시간적 서브-계층 액세스(temporal sub-layer access; TSA) 픽처의 코딩된 슬라이스, 스텝별 시간적 서브-계층 액세스(step-wise temporal sub-layer access; STSA) 픽처의 코딩된 슬라이스, 비-TSA의 코딩된 슬라이스, 비-STSA 후미 픽처(trailing picture), 파손 링크 액세스 픽처(broken link access picture)의 코딩된 슬라이스, 순간적 디코딩된 리프레시 픽처의 코딩된 슬라이스, 클린 랜덤 액세스 픽처(clean random access picture)의 코딩된 슬라이스, 디코딩가능한 선두 픽처(decodable leading picture)의 코딩된 슬라이스, 폐기 태그된 픽처(tagged for discard picture)의 코딩된 슬라이스, 비디오 파라미터 세트, 시퀀스 파라미터 세트, 픽처 파라미터 세트, 액세스 유닛 구분자(access unit delimiter), 시퀀스 종료, 비트스트림 종료, 필러 데이터(filler data), 및/또는 시퀀스 강화 정보 메시지와 같이, 데이터의 다른 카테고리들에 대해 포함될 수도 있다. 표 (4)는 NAL 유닛 코드들 및 NAL 유닛 타입 클래스들의 하나의 예를 예시한다. 다른 NAL 유닛 타입들은 희망하는 바와 같이 포함될 수도 있다. 또한, 표 (4)에서 도시된 NAL 유닛들에 대한 NAL 유닛 타입 값들은 다시 섞일 수도 있고 재배정될 수도 있다는 것을 이해해야 한다. 또한, 추가적인 NAL 유닛 타입들이 추가될 수도 있다. 또한, 일부의 NAL 유닛 타입들이 제거될 수도 있다.

[0253]

인트라 랜덤 액세스 포인트(intra random access point; IRAP) 픽처는, 각각의 비디오 코딩 계층 NAL 유닛이 표 (4)에서 도시된 바와 같이, BLA_W_LP 내지 RSV_IRAP_VCL23까지의 범위에서 nal_unit_type을 가지는 코딩된 픽처이다. IRAP 픽처는 인트라 코딩된 (I) 슬라이스들만을 포함한다. 순간적 디코딩 리프레시(instantaneous decoding refresh; IDR) 픽처는, 각각의 비디오 코딩 계층 NAL 유닛이 표 (4)에서 도시된 바와 같이, IDR_W_RADL 또는 IDR_N_LP와 동일한 nal_unit_type을 가지는 IRAP 픽처이다. 순간적 디코딩 리프레시(IDR) 픽처는 I 슬라이스들만을 포함하고, 디코딩 순서에서 비트스트림에서의 최초 픽처일 수도 있거나 비트스트림에서 더 이후에 나타날 수도 있다. 각각의 IDR 픽처는 디코딩 순서에서 코딩된 비디오 시퀀스(coded video sequence; CVS)의 최초 픽처이다. 파손 링크 액세스(broken link access; BLA) 픽처는, 각각의 비디오 코딩 계층 NAL 유닛이 표 (4)에서 도시된 바와 같이, BLA_W_LP, BLA_W_RADL, 또는 BLA_N_LP와 동일한 nal_unit_type을 가지는 IRAP 픽처이다. BLA 픽처는 I 슬라이스들만을 포함하고, 디코딩 순서에서 비트스트림에서의 최초 픽처일 수도 있거나 비트스트림에서 더 이후에 나타날 수도 있다. 각각의 BLA 픽처는 새로운 코딩된 비디오 시퀀스를 시작하고, 디코딩 프로세스에 대해 IDR 픽처와 동일한 효과를 가진다. 그러나, BLA 픽처는 비위치지 않은 참조 픽처 세트를 특징하는 신텍스 엘리먼트들을 포함한다.

표 4

nal_unit_type	nal_unit_type의 명칭	NAL 유닛 및 원시 바이트 시퀀스 페이로드(RBSP) 선택스 구조의 콘텐츠	NAL 유닛 타입 클래스
0 1	TRAIL_N TRAIL_R	비-TSA, 비-STSA 후미 픽처의 코딩된 슬라이스 세그먼트 slice_segment_laver_rbsp()	비디오 코딩 계층 (VCL)
2 3	TSA_N TSA_R	시간적 서브-계층 액세스(TSA) 픽처의 코딩된 슬라이스 세그먼트 slice_segment_laver_rbsp()	VCL
4 5	STSA_N STSA_R	스탬벌 시간적 서브-계층 액세스(STSA) 픽처의 코딩된 슬라이스 세그먼트 slice_segment_laver_rbsp()	VCL
6 7	RADL_N RADL_R	랜덤 액세스 디코딩가능한 선두(RADL) 픽처의 코딩된 슬라이스 세그먼트 slice_segment_laver_rbsp()	VCL
8 9	RASL_N RASL_R	랜덤 액세스 스킵된 선두(RASL) 픽처의 코딩된 슬라이스 세그먼트 slice_segment_laver_rbsp()	VCL
10 12 14	RSV_VCL_N10 RSV_VCL_N12 RSV_VCL_N14	예약된 비-IRAP 서브-계층 비-참조 VCL NAL 유닛 타입들	VCL
11 13 15	RSV_VCL_R11 RSV_VCL_R13 RSV_VCL_R15	예약된 비-IRAP 서브-계층 참조 VCL NAL 유닛 타입들	VCL
16 17 18	BLA_W_LP BLA_W_RADL BLA_N_LP	과손 링크 액세스(BLA) 픽처의 코딩된 슬라이스 세그먼트 slice_segment_laver_rbsp()	VCL
19 20	IDR_W_RADL IDR_N_LP	순간적 디코딩 리프레시(IDR) 픽처의 코딩된 슬라이스 세그먼트 slice_segment_laver_rbsp()	VCL
21	CRA_NUT	클린 랜덤 액세스(CRA) 픽처의 코딩된 슬라이스 세그먼트 slice_segment_laver_rbsp()	VCL
22 23	RSV_IRAP_VCL22 RSV_IRAP_VCL23	예약된 IRAP VCL NAL 유닛 타입들	VCL
24..31	RSV_VCL24.. RSV_VCL31	예약된 비-IRAP VCL NAL 유닛 타입들	VCL
32	VPS_NUT	비디오 파라미터 세트 video_parameter_set_rbsp()	비-비디오 코딩 계층 (비-VCL)
33	SPS_NUT	시퀀스 파라미터 세트 seq_parameter_set_rbsp()	비-VCL
34	PPS_NUT	픽처 파라미터 세트 pic_parameter_set_rbsp()	비-VCL
35	AUD_NUT	액세스 유닛 구분자 access_unit_delimiter_rbsp()	비-VCL
36	EOS_NUT	시퀀스 종료 end_of_seq_rbsp()	비-VCL
37	EOB_NUT	비트스트림 종료 end_of_bitstream_rbsp()	비-VCL
38	FD_NUT	필러 데이터 filler_data_rbsp()	비-VCL
39 40	PREFIX_SEI_NUT SUFFIX_SEI_NUT	보충 강화 정보 sei_rbsp()	비-VCL
41..47	RSV_NVCL41.. RSV_NVCL47	예약됨	비-VCL
48..63	UNSPEC48.. UNSPEC63	미특정	비-VCL

표 (4)

[0254]

[0255]

표 (5)를 참조하면, NAL 유닛 헤더 선택스는 데이터의 2 바이트들, 즉, 16 비트들을 포함할 수도 있다. 최초 비트는 NAL 유닛의 시작 시에 항상 제로(zero)로 설정되는 "forbidden_zero_bit"이다. 다음 6 비트들은, 표 (4)에서 도시된 바와 같이 NAL 유닛 내에 포함된 원시 바이트 시퀀스 페이로드들("RBSP") 데이터 구조의 타입을 특정하는 "nal_unit_type"이다. 다음 6 비트들은 계층의 식별자를 특정하는 "nuh_layer_id"이다. 일부의 경우 들에는, 이 6 비트들이 그 대신에 "nuh_reserved_zero_6bits"로서 특정될 수도 있다. nuh_reserved_zero_6bits는 표준의 기본 사양에서 0과 동일할 수도 있다. 스케일러블 비디오 코딩 및/또는 선택스 확장들에서, nuh_layer_id는 이 특별한 NAL 유닛이 이 6 비트들의 값에 의해 식별된 계층에 속하는 것으로 특정할 수도 있다. 다음의 선택스 엘리먼트는 "nuh_temporal_id_plus1"이다. nuh_temporal_id_plus1 마이너 스(minus) 1은 NAL 유닛에 대한 시간적 식별자를 특정할 수도 있다. 변수 시간적 식별자 TemporalId는 TemporalId = nuh_temporal_id_plus1 - 1로서 특정될 수도 있다. 시간적 식별자 TemporalId는 시간적 서브-계

층을 식별하기 위하여 이용된다. 변수 HighestTid는 디코딩되어야 할 최고 시간적 서브-계층을 식별한다.

표 5

nal_unit_header() {	디스크립터
forbidden_zero_bit	f(1)
nal_unit_type	u(6)
nuh_layer_id	u(6)
nuh_temporal_id_plus1	u(3)
}	

[0256]

표 (5)

[0257]

표 (6)은 예시적인 시퀀스 파라미터 세트(sequence parameter set; SPS) 선택스 구조를 도시한다.

[0258]

pic_width_in_luma_samples는 루마 샘플들의 단위로 각각의 디코딩된 픽처의 폭을 특정한다. pic_width_in_luma_samples는 0과 동일하지 않을 것이다.

[0259]

pic_height_in_luma_samples는 루마 샘플들의 단위로 각각의 디코딩된 픽처의 높이를 특정한다. pic_height_in_luma_samples는 0과 동일하지 않을 것이다.

[0260]

sps_max_sub_layers_minus1 플러스 1은 SPS를 참조하는 각각의 CVS에서 존재할 수도 있는 시간적 서브-계층들의 최대 수를 특정한다. sps_max_sub_layers_minus1의 값은 0 내지 6까지의 범위에 있을 것이다.

[0261]

1과 동일한 sps_sub_layer_ordering_info_present_flag 플래그는, sps_max_dec_pic_buffering_minus1[i], sps_max_num_reorder_pics[i], 및 sps_max_latency_increase_plus1[i] 선택스 엘리먼트들이 sps_max_sub_layers_minus1 + 1 서브-계층들에 대해 존재하는 것으로 특정한다. 0과 동일한 sps_sub_layer_ordering_info_present_flag는, sps_max_dec_pic_buffering_minus1[sps_max_sub_layers_minus1], sps_max_num_reorder_pics[sps_max_sub_layers_minus1], 및 sps_max_latency_increase_plus1[sps_max_sub_layers_minus1]의 값들이 모든 서브-계층들에 적용되는 것으로 특정한다.

[0262]

sps_max_dec_pic_buffering_minus1[i] 플러스 1은, HighestTid가 i와 동일할 때에 픽처 저장 버퍼들의 단위로 CVS에 대한 디코딩된 픽처 버퍼의 최대 요구된 사이즈를 특정한다. sps_max_dec_pic_buffering_minus1[i]의 값은 0 내지 MaxDpbSize - 1까지의 범위에 있을 것이고, 여기서, MaxDpbSize는 픽처 저장 버퍼들의 단위로 최대 디코딩된 픽처 버퍼 사이즈를 특정한다. i가 0보다 더 클 때, sps_max_dec_pic_buffering_minus1[i]는 sps_max_dec_pic_buffering_minus1[i - 1] 이상일 것이다. sps_max_dec_pic_buffering_minus1[i]가 0 내지 sps_max_sub_layers_minus1 - 1까지의 범위에서 i에 대해 존재하지 않을 때, sps_sub_layer_ordering_info_present_flag가 0과 동일한 것으로 인해, 그것은 sps_max_dec_pic_buffering_minus1[sps_max_sub_layers_minus1]과 동일한 것으로 추론된다.

[0263]

sps_max_num_reorder_pics[i]는, HighestTid가 i와 동일할 때, 디코딩 순서에서 CVS에서의 임의의 픽처를 선행할 수 있으며 출력 순서에서 그 픽처를 후행할 수 있는 픽처들의 최대 허용된 수를 표시한다. sps_max_num_reorder_pics[i]의 값은 0 내지 sps_max_dec_pic_buffering_minus1[i]까지의 범위일 것이다. i가 0보다 더 클 때, sps_max_num_reorder_pics[i]는 sps_max_num_reorder_pics[i - 1] 이상일 것이다. sps_max_num_reorder_pics[i]가 0 내지 sps_max_sub_layers_minus1 - 1까지의 범위에서 i에 대해 존재하지 않을 때, sps_sub_layer_ordering_info_present_flag가 0과 동일한 것으로 인해, 그것은 sps_max_num_reorder_pics[sps_max_sub_layers_minus1]과 동일한 것으로 추론된다.

[0264]

0과 동일하지 않은 sps_max_latency_increase_plus1[i]는, HighestTid가 i와 동일할 때, 출력 순서에서 CVS에서의 임의의 픽처를 선행할 수 있으며 디코딩 순서에서 그 픽처를 후행할 수 있는 픽처들의 최대 수를 특정하는 SpsMaxLatencyPictures[i]의 값을 연산하기 위하여 이용된다.

[0265]

sps_max_latency_increase_plus1[i]가 0과 동일하지 않을 때, SpsMaxLatencyPictures[i]의 값은 다음과 같이 특정된다:

[0266]

SpsMaxLatencyPictures[i] = sps_max_num_reorder_pics[i] + sps_max_latency_increase_plus1[i] - 1

[0267] sps_max_latency_increase_plus1[i]가 0과 동일할 때, 대응하는 한계가 표현되지 않는다.

[0268] sps_max_latency_increase_plus1[i]의 값은 0 내지 $2^{32} - 2$ 까지의 범위에 있을 것이다. sps_max_latency_increase_plus1[i]가 0 내지 sps_max_sub_layers_minus1 - 1까지의 범위에서 i에 대해 존재하지 않을 때, sps_sub_layer_ordering_info_present_flag가 0과 동일한 것으로 인해, 그것은 sps_max_latency_increase_plus1[sps_max_sub_layers_minus1]과 동일한 것으로 추론된다.

표 6

seq_parameter_set_rbsp() {
...
sps_max_sub_layers_minus1
...
pic_width_in_luma_samples
pic_height_in_luma_samples
...
for(i = (sps_sub_layer_ordering_info_present_flag ? 0 :
sps_max_sub_layers_minus1);
i <= sps_max_sub_layers_minus1; i++) {
sps_max_dec_pic_buffering_minus1[i]
sps_max_num_reorder_pics[i]
sps_max_latency_increase_plus1[i]
}
...
}

표 (6)

[0269]

[0270] 현재의 픽처가 IRAP 픽처일 때, 다음이 적용된다:

[0271] - 현재의 픽처가 IDR 픽처, BLA 픽처, 디코딩 순서에서 비트스트림에서의 최초 픽처, 또는 디코딩 순서에서 시퀀스 NAL 유닛의 종점을 후행하는 최초 픽처일 경우, 변수 NoRaslOutputFlag가 1과 동일하게 설정된다.

[0272] - 이와 다르게, 변수 HandleCraAsBlaFlag를 현재의 픽처에 대한 값으로 설정하기 위하여 일부의 외부 수단이 이용가능할 경우, 변수 HandleCraAsBlaFlag는 외부 수단에 의해 제공된 값과 동일하게 설정되고, 변수 NoRaslOutputFlag는 HandleCraAsBlaFlag과 동일하게 설정된다.

[0273] - 이와 다를 경우, 변수 HandleCraAsBlaFlag는 0과 동일하게 설정되고, 변수 NoRaslOutputFlag는 0과 동일하게 설정된다.

[0274] - 현재의 픽처가 픽처 0이 아닌, 1과 동일한 NoRaslOutputFlag를 갖는 IRAP 픽처일 경우, 다음의 순서화된 단계들이 적용된다:

[0275] 1. 변수 NoOutputOfPriorPicsFlag는 다음과 같이 테스트 중인 디코더에 대해 유도된다:

[0276] - 현재의 픽처가 CRA 픽처일 경우, NoOutputOfPriorPicsFlag는 (no_output_of_prior_pics_flag의 값에 관계없이) 1과 동일하게 설정된다.

[0277] - 이와 다르게, 활성 SPS로부터 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid]의 값이 선행하는 픽처에 대해 활성인 SPS로부터 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid]의 값과는 각각 상이할 경우, NoOutputOfPriorPicsFlag는 no_output_of_prior_pics_flag의 값에 관계없이, 테스트 중인 디코더에 의해 1로 설정될 수도 있다(그러나, 설정되지 않아야 함).

[0278] - 이와 다를 경우, NoOutputOfPriorPicsFlag는 no_output_of_prior_pics_flag와 동일하게 설정된다.

[0279] 2. 테스트 중인 디코더에 대해 유도된 NoOutputOfPriorPicsFlag의 값은 다음과 같이 HRD에 대해 적용된다:

[0280] - NoOutputOfPriorPicsFlag가 1과 동일할 경우, DPB에서의 모든 픽처 저장 버퍼들은 이들이 포함하는 픽처들의 출력 없이 비워지고, DPB 충만도(fullness)는 0과 동일하게 설정된다.

[0281] - 이와 다를 경우(NoOutputOfPriorPicsFlag가 0과 동일함), "출력을 위해 필요하지 않음" 및 "참조를 위해 이용

되지 않음"으로서 표기되는 픽처를 포함하는 모든 픽처 저장 버퍼들은 (출력 없이) 비워지고, DPB에서의 모든 비워지지 않은 픽처 저장 버퍼들은 "범핑" 프로세스(1204)를 반복적으로 호출함으로써 비워지고, DPB 충만도는 0과 동일하게 설정된다.

- [0282] - 이와 다를 경우(현재의 픽처가 1과 동일한 NoRaslOutputFlag를 갖는 IRAP 픽처가 아님), "출력을 위해 필요하지 않음" 및 "참조를 위해 이용되지 않음"으로서 표기되는 픽처를 포함하는 모든 픽처 저장 버퍼들이 (출력 없이) 비워진다. 비워지는 각각의 픽처 저장 버퍼에 대하여, DPB 충만도는 1씩 감소된다. 다음 조건들 중의 하나 이상이 참일 때에는, 다음 조건들의 어느 것도 참이 아닐 때까지, 비워지는 각각의 추가적인 픽처 저장 버퍼에 대하여 DPB 충만도를 1씩 추가로 감소시키면서, "범핑" 프로세스(1204)가 반복적으로 호출된다.
- [0283] 1. "출력을 위해 필요함"으로서 표기되는 DPB에서의 그 특별한 nuh_layer_id 값을 갖는 픽처들의 수는 (그 특별한 nuh_layer_id 값이 0과 동일할 때의) 활성 시퀀스 파라미터 세트로부터, 또는 그 특별한 nuh_layer_id 값에 대한 활성 계층 시퀀스 파라미터 세트로부터의 sps_max_num_reorder_pics[HighestTid]보다 더 크다.
- [0284] 2. (그 특별한 nuh_layer_id 값이 0과 동일할 때의) 활성 시퀀스 파라미터 세트로부터, 또는 그 특별한 nuh_layer_id 값에 대한 활성 계층 시퀀스 파라미터 세트로부터의 sps_max_latency_increase_plus1[HighestTid]가 0과 동일하지 않고, 연관된 변수 PicLatencyCount가 그 특별한 nuh_layer_id 값에 대한 SpsMaxLatencyPictures[HighestTid] 이상인, "출력을 위해 필요함"으로서 표기되는 DPB에서의 그 특별한 nuh_layer_id 값을 갖는 적어도 하나의 픽처가 있을 경우.
- [0285] 3. DPB에서의 그 특별한 nuh_layer_id 값을 갖는 픽처들의 수가 (그 특별한 nuh_layer_id 값이 0과 동일할 때의) 활성 시퀀스 파라미터 세트로부터, 또는 그 특별한 nuh_layer_id 값에 대한 활성 계층 시퀀스 파라미터 세트로부터의 sps_max_dec_pic_buffering[HighestTid] + 1 이상이다.
- [0286] 블록(1206)에서의 픽처 디코딩 프로세스(픽처 디코딩 및 표기)는 현재의 픽처를 포함하는 액세스 유닛의 최후 디코딩 유닛이 CPB로부터 제거될 때 순간적으로 발생한다.
- [0287] "출력을 위해 필요함"으로서 표기되는 DPB에서의 현재의 픽처의 nuh_layer_id 값과 동일한 nuh_layer_id 값을 갖는 각각의 픽처에 대하여, 연관된 변수 PicLatencyCount는 PicLatencyCount + 1과 동일하게 설정된다.
- [0288] 현재의 픽처는 픽처의 최후 디코딩 유닛이 디코딩된 후에 디코딩된 것으로 간주된다. 현재의 디코딩된 픽처는 DPB에서의 비워진 픽처 저장 버퍼에서 저장되고, 다음이 적용된다:
- [0289] - 현재의 디코딩된 픽처가 1과 동일한 PicOutputFlag를 가질 경우, 그것은 "출력을 위해 필요함"으로서 표기되고, 그 연관된 변수 PicLatencyCount는 0과 동일하게 설정된다.
- [0290] - 이와 다를 경우(현재의 디코딩된 픽처가 0과 동일한 PicOutputFlag를 가짐), 그것은 "출력을 위해 필요하지 않음"으로서 표기된다.
- [0291] 현재의 디코딩된 픽처는 "단기 참조를 위해 이용됨"으로서 표기된다.
- [0292] 다음의 조건들 중의 하나 이상이 참일 때에는, 다음의 조건들의 어느 것도 참이 아닐 때까지, 추가적인 "범핑" 프로세스(1208)가 반복적으로 호출된다:
- [0293] - "출력을 위해 필요함"으로서 표기되는 DPB에서의 현재의 픽처의 nuh_layer_id 값과 동일한 nuh_layer_id 값을 갖는 픽처들의 수는 (현재의 픽처의 nuh_layer_id 값이 0과 동일할 때의) 활성 시퀀스 파라미터 세트로부터, 또는 현재의 픽처의 nuh_layer_id 값에 대한 활성 계층 시퀀스 파라미터 세트로부터의 sps_max_num_reorder_pics[HighestTid]보다 더 크다.
- [0294] - (현재의 픽처의 nuh_layer_id 값이 0과 동일할 때의) 활성 시퀀스 파라미터 세트로부터, 또는 현재의 픽처의 nuh_layer_id 값에 대한 활성 계층 시퀀스 파라미터 세트로부터의 sps_max_latency_increase_plus1[HighestTid]가 0과 동일하지 않고, 연관된 변수 PicLatencyCount가 그 특별한 nuh_layer_id 값에 대한 SpsMaxLatencyPictures[HighestTid] 이상인, "출력을 위해 필요함"으로서 표기되는 DPB에서의 그 특별한 nuh_layer_id 값을 갖는 적어도 하나의 픽처가 있다.
- [0295] "범핑" 프로세스(1204) 및 추가적인 범핑 프로세스(1208)는 단계들의 측면에서 동일하고, 다음의 순서화된 단계들로 구성된다: 출력을 위해 최초인 픽처는 "출력을 위해 필요함"으로서 표기된 DPB에서의 모든 픽처들의 픽처 순서 카운트(PicOrderCntVal)의 최소 값을 가지는 것들로서 선택된다. 픽처 순서 카운트는, 각각의 픽처와 연관되고, CVS에서의 모든 픽처들 중에서 연관된 픽처를 고유하게 식별하고, 연관된 픽처가 디코딩된 픽처 버퍼로

부터 출력되어야 할 때, 디코딩된 픽처 버퍼로부터 출력되어야 하는 동일한 CVS에서의 다른 픽처들의 출력 순서 위치들에 관련하여 출력 순서에서 연관된 픽처의 위치를 표시하는 변수이다.

[0296] - 이 픽처들은 0과 동일한 nuh_layer_id를 갖는 픽처에 대한 활성화 시퀀스 파라미터 세트에서, 또는 픽처의 그것과 동일한 nuh_layer_id 값에 대한 활성화 계층 시퀀스 파라미터 세트에서 특정된 적합성 크롭핑 윈도우 (conformance cropping window)를 이용하여 크롭핑되고, 크롭핑된 픽처들은 nuh_layer_id의 오름차순으로 출력되고, 픽처들은 "출력을 위해 필요하지 않음"으로서 표기된다.

[0297] - "참조를 위해 이용되지 않음"으로서 표기된 픽처를 포함하며 크롭핑 및 출력되었던 픽처들 중의 하나를 포함하였던 각각의 픽처 저장 버퍼는 비워진다.

[0298] 도 13a를 참조하면, 이전에 설명된 바와 같이, NAL 유닛 헤더 선택스는 데이터의 2 바이트들, 즉, 16 비트들을 포함할 수도 있다. 최초 비트는 NAL 유닛의 시작 시에 항상 제로로 설정되는 "forbidden_zero_bit"이다. 다음 6 비트들은, NAL 유닛 내에 포함된 원시 바이트 시퀀스 페이로드들("RBSP") 데이터 구조의 타입을 특정하는 "nal_unit_type"이다. 다음 6 비트들은 "nuh_reserved_zero_6bits"이다. nuh_reserved_zero_6bits는 표준의 기본 사양에서 0과 동일할 수도 있다. nuh_reserved_zero_6bits의 다른 값들은 희망하는 바와 같이 특정될 수도 있다. 디코더들은 표준의 기본 사양에 기반으로 하여 스트림을 처리할 때, 0과 동일하지 않은 nuh_reserved_zero_6bits의 값들을 갖는 모든 NAL 유닛들을 무시(즉, 비트스트림으로부터 제거하여 폐기함)할 수도 있다. 스케일러블 또는 다른 확장에서, nuh_reserved_zero_6bits는 스케일러블 비디오 코딩 및/또는 선택스 확장들을 시그널링하기 위하여 다른 값들을 특정할 수도 있다. 일부의 경우에는, 선택스 엘리먼트 nuh_reserved_zero_6bits가 reserved_zero_6bits로 칭해질 수도 있다. 일부의 경우에는, 도 13b 및 도 13c에서 예시된 바와 같이, 선택스 엘리먼트 nuh_reserved_zero_6bits가 layer_id_plus1 또는 layer_id로서 칭해질 수도 있다. 이 경우, 엘리먼트 layer_id는 layer_id_plus1 마이너스 1일 것이다. 이 경우, 그것은 스케일러블 코딩된 비디오의 계층에 관련된 정보를 시그널링하기 위하여 이용될 수도 있다. 다음의 선택스 엘리먼트는 "nuh_temporal_id_plus1"이다. nuh_temporal_id_plus1 마이너스 1은 NAL 유닛에 대한 시간적 식별자를 특정할 수도 있다. 변수 시간적 식별자 TemporalId는 TemporalId = nuh_temporal_id_plus1 - 1로서 특정될 수도 있다.

[0299] 도 14를 참조하면, 일반적인 NAL 유닛 선택스 구조가 예시되어 있다. 도 13의 NAL 유닛 헤더 2 바이트 선택스는 도 14의 nal_unit_header()를 참조하여 포함된다. NAL 유닛 선택스의 나머지는 RBSP와 주로 관련 있다.

[0300] "nuh_reserved_zero_6bits"를 이용하기 위한 하나의 기존의 기법은, nuh_reserved_zero_6bits의 6 비트들을 별개의 비트 필드들, 즉, 각각이 스케일러블 코딩된 비디오의 상이한 계층의 식별을 지칭하는, 종속성 ID, 품질 ID, 뷰 ID, 및 심도 플래그 중의 하나 이상으로 파티셔닝함으로써 스케일러블 비디오 코딩 정보를 시그널링하는 것이다. 따라서, 6 비트들은 이 특별한 NAL 유닛이 스케일러블 인코딩 기법의 어느 계층에 속하는지를 표시한다. 다음으로, 도 15에서 예시된 바와 같이, 비디오 파라미터 세트("VPS") 확장 선택스("scalability_type")와 같은 데이터 페이로드에서는, 계층에 대한 정보가 정의된다. 도 15의 VPS 확장 선택스는, NAL 유닛 헤더에서 layer_id_plus1(또는 layer_id)을 통해 시그널링된 코딩된 비디오 시퀀스 및 차원들에서의 이용 주인 스케일러빌리티 타입들을 특정하는 스케일러빌리티 타입(선택스 엘리먼트 scalability_type)에 대한 4 비트들을 포함한다. 스케일러빌리티 타입이 0과 동일할 때, 코딩된 비디오 시퀀스는 기본 사양을 준수하고, 이에 따라, 모든 NAL 유닛들의 layer_id_plus1은 0과 동일하고, 강화 계층 또는 뷰에 속하는 NAL 유닛들이 없다. 스케일러빌리티 타입의 더 높은 값들은 도 16에서 예시된 바와 같이 해독된다.

[0301] layer_id_dim_len[i]는 i번째 스케일러빌리티 차원 ID의 비트들에서의 길이를 특정한다. 0 내지 7의 범위에서의 모든 i 값들에 대한 값들 layer_id_dim_len[i]의 합은 6 이하이다. vps_extension_byte_alignment_reserved_zero_bit는 제로이다. vps_layer_id[i]는 다음의 계층 종속성 정보가 적용되는 i번째 계층의 layer_id의 값을 특정한다. num_direct_ref_layers[i]는 i번째 계층이 직접적으로 종속되는 계층들의 수를 특정한다. ref_layer_id[i][j]는 i번째 계층이 직접적으로 종속되는 j번째 계층을 식별한다.

[0302] 이러한 방식으로, 기존의 기법은 도 16에서 열거된 스케일러빌리티 타입들 사이에서 비트들을 할당하기 위하여 NAL 유닛 및 비디오 파라미터 세트에서 스케일러빌리티 식별자들을 시그널링한다. 다음으로, 각각의 스케일러빌리티 타입에 대하여, 도 16은 얼마나 많은 차원들이 지원되는지를 정의한다. 예를 들어, 스케일러빌리티 타입 1은 2 개의 차원들(즉, 공간 및 품질)을 가진다. 차원들의 각각에 대하여, layer_id_dim_len[i]는 이 2개의 차원들의 각각에 할당된 비트들의 수를 정의하고, 여기서, layer_id_dim_len[i]의 모든 값들의 총 합은, NAL 유

닛 헤더의 nuh_reserved_zero_6bits에서의 비트들의 수인 6 이하이다. 이에 따라, 조합하여, 기법은 어느 타입들의 스케일러빌리티가 이용 중인지와, NAL 유닛 헤더의 6 비트들이 스케일러빌리티 사이에서 어떻게 할당되는지를 식별한다.

[0303] 도 16에서 예시된 바와 같이, 상이한 스케일러빌리티 차원들의 이러한 고정된 조합은 많은 애플리케이션들에 대해 적당하지만, 포함되지 않은 바람직한 조합들이 있다. 도 17을 참조하면, 수정된 비디오 파라미터 세트 확장 선택스는 nuh_reserved_zero_6bits 선택스 엘리먼트에서의 각각의 비트에 대한 스케일러빌리티 타입을 특정한다. vps_extension_byte_alignment_reserved_zero_bit는 0으로 설정된다. max_num_layers_minus1_bits는 layer_id_plus1 또는 nuh_reserved_zero_6bits로서 지칭된 도 13에서의 NAL 유닛 헤더의 최초 2 바이트들에서의 선택스 엘리먼트에 대해 이용된 비트들의 총 수를 표시한다. scalability_map[i]은 layer_id_plus1 선택스 엘리먼트에서의 각각의 비트에 대한 스케일러빌리티 타입을 특정한다. 일부의 경우에는, layer_id_plus1 선택스 엘리먼트가 그 대신에, nuh_reserved_zero_6bits 또는 rreserved_zero_6bits 선택스 엘리먼트로 칭해질 수도 있다. 선택스 엘리먼트 layer_id_plus1의 모든 비트들에 대한 스케일러빌리티 맵은 모두 함께 코딩된 비디오 시퀀스에서 이용 중인 스케일러빌리티를 특정한다. 스케일러빌리티 타입들의 각각에 대한 식별자의 실제적인 값은 NAL 유닛 헤더에서의 layer_id_plus1(nuh_reserved_zero_6bits) 필드에서 그 대응하는 비트들을 통해 시그널링된다. scalability_map[i]이 i의 모든 값들에 대해 0과 동일할 때, 코딩된 비디오 시퀀스는 기본 사양을 준수하고, 이에 따라, NAL 유닛들의 layer_id_plus1 값은 0과 동일하고, 강화 계층 또는 뷰에 속하는 NAL 유닛들이 없다. vps_layer_id[i]는 다음의 계층 종속성 정보가 적용되는 i번째 계층의 layer_id의 값을 특정한다. num_direct_ref_layers[i]는 i번째 계층이 직접적으로 종속되는 계층들의 수를 특정한다. ref_layer_id[i][j]는 i번째 계층이 직접적으로 종속되는 j번째 계층을 식별한다.

[0304] scalability_map[i]의 더 높은 값들은 도 18에서 도시된 바와 같이 해독된다. 스케일러빌리티 맵 [i]은 (0) 없음; (1) 공간; (2) 품질; (3) 심도; (4) 멀티뷰; (5) 미특정; (6) 예약됨; (7) 예약됨의 스케일러빌리티 차원들을 포함한다.

[0305] 그러므로, NAL 유닛 헤더에서의 각각의 비트는 스케일러빌리티 차원이 무엇인지(예컨대, 없음, 공간, 품질, 심도, 멀티뷰, 미특정, 예약됨)에 대한 비디오 파라미터 세트에서의 3 비트들에 기반으로 하여 해독된다. 예를 들어, layer_id_plus1에서의 모든 비트들이 공간적 스케일러빌리티에 대응하는 것을 시그널링하기 위하여, VPS에서의 scalability_map 값들은 NAL 유닛 헤더의 6 비트들에 대해 001 001 001 001 001 001로서 코딩될 수도 있다. 또한, 예를 들어, layer_id_plus1에서의 3 비트들이 공간적 스케일러빌리티에 대응하고 3 비트들이 품질 스케일러빌리티에 대응하는 것을 시그널링하기 위하여, VPS에서의 scalability_map 값들은 NAL 유닛 헤더의 6 비트들에 대해 001 001 001 010 010 010으로서 코딩될 수도 있다.

[0306] 도 19를 참조하면, 또 다른 실시형태는 num_scalability_dimensions_minus1을 이용하여 NAL 유닛 헤더의 6 비트들에서의 스케일러빌리티 차원들의 수를 시그널링하는 비디오 파라미터 세트를 포함한다. num_scalability_dimensions_minus1 플러스 1은 layer_id_plus1; nuh_reserved_zero_6bits; 및/또는 reserved_zero_6bits 선택스 엘리먼트들을 통해 시그널링된 스케일러빌리티 차원들의 수를 표시한다. scalability_map[i]은 도 17과 관련하여 위에서 설명된 것과 동일한 시맨틱들을 가진다. num_bits_for_scalability_map[i]은 I번째 스케일러빌리티 차원에 대한 비트들에서의 길이를 특정한다. i=0, ..num_scalability_dimensions_minus1에 대한 num_bits_for_scalability_map[i]의 전부의 합은 6과 동일하다 (또는 이와 다르게, layer_id_plus1; vps_reserved_zero_6bits; max_num_layers_minus1; reserved_zero_6bits; nuh_reserved_zero_6bits 선택스 엘리먼트들에 대해 이용된 비트들의 수와 동일함).

[0307] 도 17 및 도 19에 대하여, 희망할 경우, 다른 변형들이 이용될 수도 있다. 하나의 실시형태에서는, 예를 들어, scalability_map[i]이 u(4)(또는, n>3 또는 n<3인 u(n))로 시그널링될 수도 있다. 이 경우, scalability_map[i]의 더 높은 값들은 비디오 기법의 특별한 프로파일을 준수하는 비트스트림들에 대해 예약된 것으로서 특정될 수도 있다. 예를 들어, 스케일러빌리티 맵 값들 6..15는 u(4)로 scalability_map[i]을 시그널링할 때에 '예약됨'으로서 특정될 수도 있다. 또 다른 실시형태에서는, 예를 들어, scalability_map[i]이 ue(v) 또는 일부의 다른 코딩 방식으로 시그널링될 수도 있다. 또 다른 실시형태에서는, 예를 들어, scalability_map[i] 값들은 단조 비-감소(또는 비-증가) 순서로 배치되도록, 한정사항(restriction)이 특정될 수도 있다. 이것은 인접해 있는 NAL 유닛 헤더 내의 layer_id_plus1 필드에서의 다양한 스케일러빌리티 차원 필드들로 귀착된다.

[0308] "layer_id_plus1" 또는 "nuh_reserved_zero_6bits" 선택스 엘리먼트를 이용하여 스케일러블 비디오 코딩을 시

그널링하기 위한 또 다른 기존의 기법은 비디오 파라미터 세트에서 일반적인 룩업 테이블을 시그널링함으로써 NAL 유닛에서의 layer_id_plus1을 계층 식별에 맵핑하는 것이다. 도 20을 참조하면, 기존의 기법은 룩업 테이블의 i번째 계층에 대한 타입들 및 차원 식별들의 수를 특정하는 비디오 파라미터 세트를 포함한다. 특히, vps_extension_byte_alignment_reserved_zero_bit는 제로이다. num_dimensions_minus1[i] 플러스 1은 i번째 계층에 대한 차원 타입들(dimension_type[i][j]) 및 차원 식별자들(dimension_id[i][j])의 수를 특정한다. dimension_type[i][j]는, 도 31에서 특정된 바와 같이, i와 동일한 layer_id 또는 layer_id_plus1을 가지는 i번째 계층의 j번째 스케일러빌리티 차원 타입을 특정한다. 도 21에서 예시된 바와 같이, 식별되는 차원들은 (0) 뷰 순서 인덱스; (1) 심도 플래그; (2) 종속성 ID; (3) 품질 ID; (4)-(15) 예약됨을 포함한다. dimension_id[i][j]는, 존재하지 않을 때에 0인 것으로 추론되는 i번째 계층의 j번째 스케일러빌리티 차원 타입의 식별자를 특정한다. num_direct_ref_layers[i]는 i번째 계층이 직접적으로 종속되는 계층들의 수를 특정한다. ref_layer_id[i][j]는 i번째 계층이 직접적으로 종속되는 j번째 계층을 식별한다. 불운하게도, 도 20에서 예시된 제안된 실시형태는 다루기 힘들게 큰 룩업 테이블로 귀착된다.

[0309] 도 22를 참조하면, 수정된 비디오 파라미터 세트 확장은 스케일러빌리티 차원과 조합하여 이용되는 스케일러빌리티 마스크를 포함한다. scalability_mask는 도 23의 스케일러빌리티 맵 선택스에 의해 표시된 바와 같이 하나의 스케일러빌리티 차원에 대응하는 각각의 비트로 0 및 1의 패턴을 시그널링한다. 특별한 스케일러빌리티 차원에 대한 1의 값은 이 스케일러빌리티 차원이 이 계층(I번째 계층)에서 존재함을 표시한다. 특별한 스케일러빌리티 차원에 대한 0의 값은 이 스케일러빌리티 차원이 이 계층(I번째 계층)에서 존재하지 않음을 표시한다. 예를 들어, 00100000의 비트들의 세트는 품질 스케일러빌리티를 지칭한다. 존재하는 특별한 스케일러빌리티 차원의 실제적인 식별자 값은 시그널링된 scalability_id[j] 값에 의해 표시된다. num_scalability_types[i]의 값들은 1의 값을 가지는 scalability_mask에서의 비트들의 수의 합과 동일하다. 이에 따라,

수학식 9

$$\text{num_scalability_types}[i] = \sum_{k=0}^7 \text{scalability_mask}[i](k)$$

[0310]

[0311] scalability_id[j]는 scalability_mask 값에 의해 시그널링되는 스케일러빌리티 값들의 타입에 대한 j번째 스케일러빌리티 차원의 식별자 값을 표시한다.

[0312] 도 24를 참조하면, 도 22의 수정은 루프 외부에서 시그널링되는 스케일러빌리티 마스크를 포함한다. 이것은 각각의 계층 식별에 대한 하나의 공통 마스크로 귀착된다. 도 25를 참조하면, 이 수정에서, 대응하는 예시적인 비디오 파라미터 세트는 포함되지 않는 스케일러빌리티 마스크를 갖는 스케일러블 식별을 포함할 수도 있다. 이 경우, 선택스 엘리먼트 scalable_id[j]는 도 22에서의 선택스 엘리먼트 scalability_id[j]와 동일한 해독을 가진다.

[0313] 도 26을 참조하면, 도 22의 수정은 루프 외부에서 시그널링되는 스케일러빌리티 마스크(scalability_mask)를 포함한다. 이것은 각각의 계층 식별에 대한 하나의 공통 마스크로 귀착된다. scalability_mask는 도 27의 스케일러빌리티 맵 선택스에 의해 표시된 바와 같이 하나의 스케일러빌리티 차원에 대응하는 각각의 비트로 0 및 1의 패턴을 시그널링한다. 특별한 스케일러빌리티 차원에 대한 1의 값은 이 스케일러빌리티 차원이 이 계층(i번째 계층)에서 존재함을 표시한다. 특별한 스케일러빌리티 차원에 대한 0의 값은 이 스케일러빌리티 차원이 이 계층(i번째 계층)에서 존재하지 않음을 표시한다. 예를 들어, 00100000의 비트들의 세트는 품질 스케일러빌리티를 지칭한다. 존재하는 특별한 스케일러빌리티 차원의 실제적인 식별자 값은 시그널링된 scalability_id[j] 값에 의해 표시된다. num_scalability_types[i]의 값들은 1의 값을 가지는 scalability_mask에서의 비트들의 수의 합과 동일하다. 이에 따라,

수학식 10

$$\text{NumScalabilityTypes}[i] = \sum_{k=0}^{15} \text{scalability_mask}(k)$$

[0314]

[0315]

이 경우, scalability_id[j] 변수는 그 대신에, dimension_id[i][j] 변수로 칭해질 수도 있다. dimension_id[i][j]는 i번째 계층의 j번째 스케일러빌리티 차원의 스케일러빌리티 식별자를 특정한다. 다음으로, 변수 ScalabilityId[i][j]는 다음과 같이 유도된다.

표 7

for(i = 1; i <= vps_max_layers_minus1; i++) {
for(k=0, j=0; k<=15; k++) {
if(scalability_mask(k)==1)
ScalabilityId [i][k]=dimension_id[i][j++]
else
ScalabilityId [i][k]=0;
}
}

[0316]

[0317]

여기서, ScalabilityId [i][k]는 다음과 같이 대응하는 스케일러빌리티 타입에 대한 차원 ID를 시그널링한다.

표 8

k	ScalabilityId [i][k]
0	DependencyId[i][k]
1	QualityId[i][k]
2	depthFlag[i][k]
3	ViewId[i][k]
4-15	예약됨

[0318]

[0319]

여기서, DependencyId[i][1]는 i번째 계층에 대한 공간적 스케일러빌리티 차원에 대한 종속성 ID이고, QualityId[i][2]는 i번째 계층에 대한 품질 스케일러빌리티 차원에 대한 품질 ID이고, depthFlag[i][3]는 i번째 계층에 대한 심도 스케일러빌리티 차원에 대한 심도 플래그/심도 ID이고, ViewId[i][4]는 i번째 계층에 대한 멀티뷰 스케일러빌리티 차원에 대한 뷰 ID이다.

[0320]

또한, 도 26에서, 1과 동일한 avc_base_codec_flag는 기본 계층이 Rec를 준수하는 것으로 특정한다. ITU-T H.264 | ISO/IEC 14496-10, 및 1과 동일한 avc_base_codec_flag는 HEVC를 특정한다. vps_nuh_layer_id_presnet_flag는 NAL 유닛 헤더에서 layer_id의 값을 시그널링하는 layer_id_in_nuh[i] 변수가 시그널링되는지를 표시한다.

[0321]

또 다른 실시형태에서, 선택스 엘리먼트들 scalability_mask[i], scalability_mask, scalability_id[j] 중의 하나 이상은 u(8)과는 상이한 수의 비트들을 이용하여 시그널링될 수도 있다. 예를 들어, 이들은 u(16)(또는, n>8 또는 n<8인 u(n))으로 시그널링될 수 있다. 또 다른 실시형태에서는, 이 선택스 엘리먼트 중의 하나 이상이 ue(v)로 시그널링될 수 있다. 또 다른 실시형태에서, scalability_mask는 layer_id_plus1; vps_reserved_zero_6bits; max_num_layers_minus1; reserved_zero_6bits; 및/또는 nuh_reserved_zero_6bits

신택스 엘리먼트들 내의 NAL 유닛 헤더에서 시그널링될 수도 있다. 일부의 실시형태들에서, 시스템은 VPS NAL 유닛들에 대해서만, 또는 비-VPS NAL 유닛들에 대해서만, 또는 모든 NAL 유닛들에 대하여 이것을 행할 수도 있다. 또 다른 실시형태에서, scalability_mask는 비트스트림 내의 어딘가에서 픽처 당 시그널링될 수도 있다. 예를 들어, 그것은 슬라이스 헤더, 픽처 파라미터 세트, 비디오 파라미터 세트, 또는 임의의 다른 파라미터 세트 또는 비트스트림의 임의의 다른 규범 부분에서 시그널링될 수도 있다.

[0322] 도 13의 NAL 유닛 헤더 내의 신택스 엘리먼트 nuh_reserved_zero_6bits 또는 layer_id_plus1이 6 비트들을 가지므로, 도 13, 도 15, 도 18, 도 20, 도 21, 도 22, 도 23 및 대응하는 설명은 6비트들을 지칭하는 것에 주목해야 한다. 그러나, 그 신택스 엘리먼트가 6 비트들과는 상이한 수의 비트들을 이용하였을 경우에, 모든 상기 설명은 적당하게 수정될 수 있다. 예를 들어, 그 대신에, 그 신택스 엘리먼트(nuh_reserved_zero_6bits 또는 layer_id_plus1)가 9 비트들을 이용하였을 경우, 도 17에서는, max_num_layer_minus1 비트들의 값이 9일 것이고, scalability_map[i]은 6 비트들 대신에 9비트들의 각각에 대해 시그널링될 것이다.

[0323] 도 24를 참조하면, 도 22의 수정은 계층 종속성 정보를 시그널링하기 위한 신택스를 제공한다. 새로운 신택스 엘리먼트 layer_dependency_information_pattern이 정의된다.

[0324] layer_dependency_information_pattern은 vps_max_layers_minus1과 동일한 길이를 갖는 0 및 1 비트들의 패턴을 시그널링한다. i번째 비트에 대한 0의 값은 layer_id (i+1)를 갖는 계층이 독립적 계층임을 표시한다. i번째 비트에 대한 1의 값은 layer_id (i+1)를 갖는 계층이 다른 계층들 중의 하나 이상에 종속되는 종속적 계층임을 표시한다.

[0325] NumDepLayers의 값들은 1의 값을 가지는 layer_dependency_information_pattern에서의 비트들의 수의 합과 동일하다. 이에 따라,

수학식 11

$$\text{NumDepLayers} = \sum_{k=0}^{\text{vps_max_layer_minus1}-1} \text{layer_dependency_information_pattern}(k)$$

[0326]

[0327] 도 29를 참조하면, 도 26의 수정은 계층 종속성 정보를 시그널링하기 위한 신택스를 제공한다. 새로운 신택스 엘리먼트 layer_dependency_flag[i]가 정의된다. layer_dependency_flag[i]는 계층이 다른 계층들에 종속하는지를 시그널링한다. 플래그에 대한 0의 값은 layer_id i를 갖는 계층이 독립적 계층임을 표시한다. i번째 비트에 대한 1의 값은 layer_id i를 갖는 계층이 종속적 계층임을 표시한다.

[0328] 도 30을 참조하면, 도 26의 수정은 계층 종속성 정보를 시그널링하기 위한 신택스를 제공한다. 새로운 신택스 엘리먼트 layer_dependency_map[i]이 정의된다. layer_dependency_map[i]은 vps_max_layers_minus1과 동일한 길이를 갖는 0 및 1 비트들의 패턴을 시그널링한다. layer_dependency_map[i]의 k번째 비트에 대한 0의 값은 계층 i가 layer_id (k+1)을 갖는 계층에 종속되지 않음을 표시한다. layer_dependency_map[i]의 k번째 비트에 대한 1의 값은 계층 i가 layer_id (k+1)을 갖는 계층에 종속됨을 표시한다.

[0329] 도 31을 참조하면, 도 26의 수정은 계층 종속성 정보를 시그널링하기 위한 신택스를 제공한다. 새로운 신택스 엘리먼트 layer_dependency_information_pattern이 정의된다.

[0330] layer_dependency_information_pattern은 vps_max_layers_minus1과 동일한 길이를 갖는 0 및 1 비트들의 패턴을 시그널링한다. i번째 비트에 대한 0의 값은 layer_id (i+1)를 갖는 계층이 독립적 계층임을 표시한다. i번째 비트에 대한 1의 값은 layer_id (i+1)를 갖는 계층이 다른 계층들 중의 하나 이상에 종속되는 종속적 계층임을 표시한다. NumDepLayers의 값들은 1의 값을 가지는 layer_dependency_information_pattern에서의 비트들의 수의 합과 동일하다. 이에 따라,

수학식 12

$$\text{NumDepLayers} = \sum_{k=0}^{\text{vps_max_layer_minus1}-1} \text{layer_dependency_information_pattern}(k)$$

[0331]

[0332]

layer_dependency_map[i]은 vps_max_layers_minus1과 동일한 길이를 갖는 0 및 1 비트들의 패턴을 시그널링한다. layer_dependency_map[i]의 k번째 비트에 대한 0의 값은 계층 i가 layer_id (k+1)을 갖는 계층에 종속되지 않음을 표시한다. layer_dependency_map[i]의 k번째 비트에 대한 1의 값은 계층 i가 layer_id (k+1)을 갖는 계층에 종속됨을 표시한다.

[0333]

도 32를 참조하면, 도 26의 수정은 계층 종속성 정보를 시그널링하기 위한 선택스를 제공한다. 도 28은 도 27에서의 선택스에 기반한 변형 선택스이다. 새로운 선택스 엘리먼트 layer_dependency_information_pattern이 정의된다.

[0334]

layer_dependency_information_pattern은 vps_max_layers_minus1과 동일한 길이를 갖는 0 및 1 비트들의 패턴을 시그널링한다. i번째 비트에 대한 0의 값은 layer_id (i+1)를 갖는 계층이 독립적 계층임을 표시한다. i번째 비트에 대한 1의 값은 layer_id (i+1)를 갖는 계층이 다른 계층들 중의 하나 이상에 종속되는 종속적 계층임을 표시한다.

[0335]

NumDepLayers의 값들은 1의 값을 가지는 layer_dependency_information_pattern에서의 비트들의 수의 합과 동일하다. 이에 따라,

수학식 13

$$\text{NumDepLayers} = \sum_{k=0}^{\text{vps_max_layer_minus1}-1} \text{layer_dependency_information_pattern}(k)$$

[0336]

[0337]

선택스 엘리먼트들 num_direct_ref_layers[i] 및 ref_layer_id[i][j]는 layer_dependency_information_pattern(i)이 1의 값을 가질 때에만 시그널링된다. 여기서, layer_dependency_information_pattern(i)은 선택스 엘리먼트 layer_dependency_information_pattern의 i번째 비트이다.

[0338]

도 33을 참조하면, 도 26의 수정은 계층 종속성 정보를 시그널링하기 위한 선택스를 제공한다. 도 29는 도 31에서의 선택스에 기반한 변형 선택스이다. 새로운 선택스 엘리먼트 layer_dependency_information_pattern이 정의된다.

[0339]

layer_dependency_information_pattern은 vps_max_layers_minus1과 동일한 길이를 갖는 0 및 1 비트들의 패턴을 시그널링한다. i번째 비트에 대한 0의 값은 layer_id (i+1)를 갖는 계층이 독립적 계층임을 표시한다. i번째 비트에 대한 1의 값은 layer_id (i+1)를 갖는 계층이 다른 계층들 중의 하나 이상에 종속되는 종속적 계층임을 표시한다.

[0340]

NumDepLayers의 값들은 1의 값을 가지는 layer_dependency_information_pattern에서의 비트들의 수의 합과 동일하다. 이에 따라,

수학식 14

$$\text{NumDepLayers} = \sum_{k=0}^{\text{vps_max_layer_minus1}-1} \text{layer_dependency_information_pattern}(k)$$

[0341]

[0342]

layer_dependency_map[i]은 vps_max_layers_minus1과 동일한 길이를 갖는 0 및 1 비트들의 패턴을 시그널링한다. layer_dependency_map[i]의 k번째 비트에 대한 0의 값은 계층 i가 layer_id (k+1)을 갖는 계층에 종속되

지 않음을 표시한다. layer_dependency_map[i]의 k번째 비트에 대한 1의 값은 계층 i가 layer_id (k+1)을 갖는 계층에 종속됨을 표시한다. 선택스 엘리먼트들 layer_dependency_map[i]은 layer_dependency_information_pattern(i)이 1의 값을 가질 때에만 시그널링된다. 여기서, layer_dependency_information_pattern(i)은 선택스 엘리먼트 layer_dependency_pattern의 i번째 비트이다.

[0343] 또 다른 실시형태에서, layer_dependency_information_pattern 선택스 엘리먼트는 1 비트 플래그 값들의 세트로서 시그널링될 수도 있다. 이 경우, vps_max_layers_minus1 1 비트 값들의 전체는 하기로서 시그널링될 것이다:

표 9

for(i = 1; i <= vps_max_layers_minus1 ; i++)
{
layer_dependency_information_pattern_flags[i];
}

[0344]

[0345] 또 다른 실시형태에서, layer_dependency_map[i] 선택스 엘리먼트는 1 비트 플래그 값들의 세트로서 시그널링될 수도 있다. 이 경우, vps_max_layers_minus1 1 비트 값들의 전체는 하기로서 시그널링될 것이다:

표 10

for(j = 1; j <= vps_max_layers_minus1 ; j++)
{
layer_dependency_map_values[i][j];
}

[0346]

[0347] 또 다른 실시형태에서, 선택스 엘리먼트들 layer_dependency_information_pattern, layer_dependency_map 중의 하나 이상은 u(v) 대신에 알려진 고정된 수의 비트들을 이용하여 시그널링될 수도 있다. 예를 들어, 이들은 u(64)를 이용하여 코딩될 수 있다.

[0348] 또 다른 실시형태에서, 선택스 엘리먼트들 layer_dependency_information_pattern, layer_dependency_map 중의 하나 이상은 ue(v) 또는 일부의 다른 코딩 방식으로 시그널링될 수도 있다.

[0349] 또 다른 실시형태에서, 다양한 선택스 엘리먼트들 및 그 시맨틱들의 명칭들은 plus1 또는 plus2를 가산함으로써, 또는 설명된 선택스 및 시맨틱들에 비해 minus1 또는 minus2를 감산함으로써 변경될 수도 있다.

[0350] 또 다른 실시형태에서, layer_dependency_information_pattern, layer_dependency_map, layer_dependency_flag[i] 등과 같은 다양한 선택스 엘리먼트들은 비트스트림 내의 어딘가에서 픽처 당 시그널링될 수도 있다. 예를 들어, 그것은 슬라이스 헤더, pps/ sps/ vps/ aps, 또는 임의의 다른 파라미터 세트 또는 비트스트림의 임의의 다른 규범 부분에서 시그널링될 수도 있다.

[0351] 이전에 설명된 바와 같이, 스케일러블 비디오 코딩은 하나 이상의 서브세트 비트스트림들을 또한 포함하는 비디오 비트스트림을 인코딩하는 기법이다. 서브세트 비디오 비트스트림은 서브세트 비트스트림에 대해 요구된 대역폭을 감소시키기 위하여 더 큰 비디오로부터 패킷들을 누락시킴으로써 유도될 수도 있다. 서브세트 비트스트림은 더 낮은 공간적 해상도(더 작은 스크린), 더 낮은 시간적 해상도(더 낮은 프레임 레이트), 또는 더 낮은 품질의 비디오 신호를 나타낼 수도 있다. 예를 들어, 비디오 비트스트림은 5 개의 서브세트 비트스트림들을 포함할 수도 있고, 여기서, 서브세트 비트스트림들의 각각은 추가적인 콘텐츠를 기본 비트스트림에 추가한다. Hannuksela 등, "Test Model for Scalable Extensions of High Efficiency Video Coding (HEVC)" JCTVC-L0453, Shanghai, October 2012는 이것에 의하여, 그 전체적으로 참조를 위해 본원에 편입된다. Chen 등,

"SHVC Draft Text 1" JCTVC-L1008, Geneva, March, 2013은 이것에 의하여, 그 전체적으로 참조를 위해 본원에 편입된다. Wang 등, "AHG9: On VPS and SPS in HEVC 3DV and scalable extensions" JCTVC-M0268, Incheon, April 2013은 이것에 의하여, 그 전체적으로 참조를 위해 본원에 편입된다.

[0352] 이전에 설명된 바와 같이, 멀티-뷰 비디오 코딩은 대안적인 뷰들을 나타내는 하나 이상의 다른 비트스트림들을 또한 포함하는 비디오 비트스트림을 인코딩하는 기법이다. 예를 들어, 다수의 뷰들은 입체 비디오 (stereoscopic video)를 위한 한 쌍의 뷰들일 수도 있다. 예를 들어, 다수의 뷰들은 상이한 뷰포인트 (viewpoint)들로부터의 동일한 장면의 다수의 뷰들을 나타낼 수도 있다. 이미지들은 상이한 뷰포인트들로부터의 동일한 장면이므로, 다수의 뷰들은 일반적으로 인터-뷰 통계적 종속성들의 많은 양을 포함한다. 그러므로, 조합된 시간적 및 인터-뷰 예측은 효율적인 멀티-뷰 인코딩을 달성할 수도 있다. 예를 들어, 프레임은 일시적으로 관련된 프레임들로부터 뿐만 아니라, 이웃하는 뷰포인트들의 프레임들로부터도 효율적으로 예측될 수도 있다. Hannuksela 등, "Common specification text for scalable and multi-view extensions" JCTVC-L0452, Geneva, January 2013은 이것에 의하여, 그 전체적으로 참조를 위해 본원에 편입된다. Tech 등 "MV-HEVC Draft Text 3 (ISO/IEC 23008-2:201x/PDAM2)" JCT3V-C1004_d3, Geneva, January 2013은 이것에 의하여, 그 전체적으로 참조를 위해 본원에 편입된다.

[0353] Chen 등 "SHVC Draft Text 1" JCTVC-L1008, Geneva, January 2013; Hannuksela 등 "Test Model for Scalable Extensions of High Efficiency Video Coding (HEVC)" JCTVC-L0453-spec-text, Shanghai, October 2012; 및 Hannuksela "Draft Text for Multiview Extension of High Efficiency Video Coding (HEVC)," JCTVC-L0452-spec-text-r1, Shanghai, October 2012의 각각은 그 전체적으로 참조를 위해 본원에 편입된다. JCTVC-L0452 및 JCTVC-L0453은 각각 출력 동작 포인트들의 개념을 가진다. 이것은 JCTVC-L1008에서의 출력 계층 세트들의 개념으로 더 이후에 변경되었다. 신택스 엘리먼트들에 대하여, num_output_operation_points, output_op_point_index[], 및 output_layer_flag[][]는 JCTVC-L0453 및 JCTVC-L0452에 대한 출력 동작 포인트들에 대해 정의되고, JCTVC-L1008에서의 num_output_layer_sets, output_layer_set_idx[], 및 output_layer_flag[][]로 수정된다. 이러한 문서들의 각각에서는, 계층 종속성 변경 SEI 메시지가 정의된다. SEI 메시지는 SEI 메시지를 포함하는 현재의 액세스 유닛으로 시작하는 계층 종속성 정보 변경들을 시그널링하는 것을 허용한다.

[0354] Chen 등 "SHVC Draft Text 1" JCTVC-L1008, Geneva, January 2013; Hannuksela 등 "Test Model for Scalable Extensions of High Efficiency Video Coding (HEVC)" JCTVC-L0453-spec-text, Shanghai, October 2012; 및 Hannuksela "Draft Text for Multiview Extension of High Efficiency Video Coding (HEVC)" JCTVC-L0452-spec-text-r1, Shanghai, October 2012의 각각은 그 전체적으로 참조를 위해 본원에 편입되고, 각각은 DPB로부터의 픽처들 0의 출력 및 제거를 위하여, sps_max_num_reorder_pics[HighestTid], sps_max_latency_increase_plus1[HighestTid], 및 sps_max_dec_pic_buffering[HighestTid]를 이용하는 것에 기반으로 하여 동작하는 출력 순서 디코딩된 픽처 버퍼(DPB)를 가진다. 이 정보는, 만약 존재한다면, 강화 계층들을 포함하는 비디오 콘텐츠에 대한 버퍼링 정보를 제공하는, 기본 계층에 대한 비디오 파라미터 세트에서 시그널링된다.

[0355] 도 34를 참조하면, 비디오 파라미터 세트의 또 다른 예가 예시된다. 특히, 도 34의 비디오 파라미터 세트는 연 관된 비디오 파라미터 세트 확장(vps_extension)에 대한 참조를 포함한다. vps_extension은 vps_extension_flag, vps_extension2_flag, 및/또는 vps_extension_data_flag로 시그널링될 수도 있다.

[0356] 도 35를 참조하면, 비디오 파라미터 세트 확장의 또 다른 예가 예시된다. 특히, 도 35의 비디오 파라미터 세트 확장은, 디코더에 의해 뷰어(viewer)에게 출력될 수도 있는 비트스트림의 계층들의 세트의 그러한 계층들을 정의하는 num_output_layer_sets에 대한 참조를 포함한다. num_output_layer_sets는, 출력 계층들이 output_layer_set_index[i] 및 output_layer_flag[lIdx][j]로 특정되는 계층 세트들의 수를 특정한다. 존재하지 않을 때, num_output_layer_sets의 값은 0과 동일한 것으로 추론된다. 출력 계층들을 설명하는 계층 세트는 출력 계층 세트이다.

[0357] output_layer_set_idx[i]는, output_layer_flag[lIdx][j]가 존재하는 계층 세트의 인덱스 lIdx를 특정한다.

[0358] l과 동일한 output_layer_flag[lIdx][j]는 j와 동일한 nuh_layer_id를 갖는 계층이 lIdx번째 계층 세트의 타겟 출력 계층인 것으로 특정한다. 0과 동일한 output_layer_flag[lIdx][j]의 값은 j와 동일한 nuh_layer_id를 갖는 계층이 lIdx번째 계층 세트의 타겟 출력 계층이 아닌 것으로 특정한다.

- [0359] num_output_layer_sets가 설명된 방식으로 vps_extension에서 정의됨으로써, num_output_layer_sets는 대응하는 비디오 파라미터 세트 확장과 함께 새로운 비디오 파라미터 세트를 전송함으로써 업데이트될 수도 있다. 불운하게도, 대응하는 비디오 파라미터 세트 확장과 함께 새로운 비디오 파라미터 세트를 전송하는 것은 이용가능한 대역폭에서의 상당한 감소로 귀착된다. 추가적으로, 새로운 비디오 파라미터 세트는 오직 어떤 픽처 타입들에서, 예를 들어, 인트라 랜덤 액세스 포인트 픽처 타입들에서 활성화될 수 있다.
- [0360] 일 예로서, 비트스트림은 기본 계층 0 및 4 개의 강화 계층들, 즉, 강화 계층 1, 강화 계층 2, 강화 계층 3, 및 강화 계층 4를 포함할 수도 있다. 제 1 계층 세트는 기본 계층 0 및 강화 계층 1일 수도 있다. 제 2 계층 세트는 기본 계층 0, 강화 계층 1, 및 강화 계층 2일 수도 있다. 제 3 계층 세트는 기본 계층 0, 강화 계층 1, 및 강화 계층 3일 수도 있다. 제 4 계층 세트는 기본 계층 0 및 강화 계층 4일 수도 있다. 출력 계층 세트들은 출력으로서 제공될 수도 있는 계층 세트의 특별한 계층들을 정의한다. 예를 들어, 출력 계층 세트 1은, 계층 0이 출력이고 계층 1이 출력이 아닌 것일 수도 있다. 예를 들어, 출력 계층 세트 2는, 계층 0이 출력이고, 계층 1이 출력이고, 계층 2가 출력이 아닌 것일 수도 있다. 예를 들어, 출력 계층 세트 3은, 계층 0이 출력이 아니고, 계층 1이 출력이 아니고, 계층 3이 출력인 것일 수도 있다. 예를 들어, 출력 계층 세트 4는, 계층 0이 출력이고 계층 4가 출력이 아닌 것일 수도 있다. 출력 계층들의 제한된 수는 제한된 디코딩 용량들을 수용하기 위해 유용하다. 일부의 경우들에는, 다른 계층들을 디코딩하는 것 및/또는 디코더 용량들을 수용하기 위하여 상이한 계층들 사이의 일부의 종속성들을 디스에이블(disable)하는 것이 바람직하다. 일부의 계층 종속성들이 변경될 때, 다른 계층들을 출력 계층들로서 인에이블(enable), 디스에이블, 및/또는 추가하는 것이 마찬가지로 바람직할 수도 있다. 일 예로서, 계층 종속성 변경은 특별한 타겟 계층(예컨대, MV-HEVC에서의 뷰)을 디코딩하기 위하여 더 적은 계층들이 디코딩될 것을 요구할 수 있다. 이에 따라, 디코더는 그 하드웨어 제한으로 인해 그것이 이전에 할 수 없었던 계층 종속성이 변경된 후에 계층(예컨대, 그리고 추가적인 뷰)을 디코딩 및 출력할 수 있다. 현재에는, 새로운 비디오 파라미터 세트를 전송하지 않으면서, 출력 계층 세트들에서의 변경을 시그널링하는 것을 허용하기 위한 메커니즘이 존재하지 않는다.
- [0361] 도 36을 참조하면, SEI 메시지 선택스는 출력 계층 세트들의 변경들, 즉, output_layer_sets_change(payloadSize)를 가능하게 하기 위하여 이용될 수도 있다. 이 SEI 메시지는, 출력 계층 세트들 정보가 SEI 메시지를 포함하는 현재의 액세스 유닛으로 시작하여 변경되고 활성화 비디오 파라미터 세트에 대하여 해독됨을 표시한다. 하나를 초과하는 비디오 파라미터 세트가 비트스트림과 함께 포함될 수도 있고, 활성화 비디오 파라미터 세트는 희망하는 바와 같이 변경된다. 존재할 때, 출력 계층 세트들 변경 SEI 메시지는, 어느 것이든지 디코딩 순서에 있어서 더 빠른, 다음 출력 계층 세트들 변경 SEI 메시지 또는 코딩된 비디오 시퀀스의 종료까지, 현재의 액세스 유닛 및 디코딩 순서에서의 모든 후속 액세스 유닛들로 구성되는 타겟 액세스 유닛 세트에 적용된다. 출력 계층 세트들 변경 SEI 메시지들은 바람직하게는 누적 효과(cumulative effect)를 가지지 않는다.
- [0362] 'active_vps_id'는, 변경 또는 추가가 적용되는 출력 계층 세트들 정보를 포함하는 활성화 비디오 파라미터 세트를 식별한다. active_vps_id의 값은 SEI 메시지를 포함하는 액세스 유닛의 VCL NAL 유닛들에 대한 활성화 비디오 파라미터 세트의 video_parameter_set_id의 값과 동일하다.
- [0363] 'num_changed_output_layer_sets'는, 출력 계층들이 active_vps_id에 의해 식별된 활성화 비디오 파라미터 세트의 vps 확장 섹션에서 특정되는 변경된 출력 계층 세트들의 수를 특정한다. num_changed_output_layer_sets의 값은 0 내지 num_output_layer_sets까지의 범위에 있어야 한다. 존재하지 않을 때, num_changed_output_layer_sets의 값은 0과 동일한 것으로 추론된다.
- [0364] 'changed_output_layer_sets_idx_entry'[i]는, output_layer_flag[clsIdx][j] 변경이 적용되는 active_vps_id에 의해 식별된 vps 확장에서의 출력 계층 세트 엔트리들의 리스트에서 인덱스 엔트리를 식별한다. changed_output_layer_set_idx_entry[i]의 값은 0 내지 num_output_layer_sets까지의 범위에 있어야 한다. changed_output_layer_sets_idx_entry[i]는 i와 동일하지 않은 임의의 j에 대하여 changed_output_layer_sets_idx_entry[j]와 동일하지 않다.
- [0365] 1과 동일한 'output_layer_flag'[clsIdx][j]는, j와 동일한 nuh_layer_id를 갖는 계층이 clsIdx번째 계층 세트의 타겟 출력 계층인 것으로 특정한다. 0과 동일한 output_layer_flag[clsIdx][j]의 값은 j와 동일한 nuh_layer_id를 갖는 계층이 clsIdx번째 계층 세트의 타겟 출력 계층이 아닌 것으로 특정한다.
- [0366] output_layer_flag[clsIdx][j]가 0 내지 vps_num_layer_sets_minus1까지의 범위에서의 clsIdx에 대해, 그리고 0 내지 63까지의 범위에서의 j에 대해 존재하지 않을 때, output_layer_flag[clsIdx][j]는 (j = =

LayerSetLayerIdList[clsIdx][NumLayersInIdList[clsIdx] - 1])과 동일한 것으로 추론된다.

- [0367] 'num_addl_output_layer_sets'는, 출력 계층들이 addl_output_layer_set_idx[i] 및 output_layer_flag[addlIdx][j]로 지정되는 추가적인 계층 세트들의 수를 특정한다. 존재하지 않을 때, num_addl_output_layer_sets의 값은 0과 동일한 것으로 추론된다. 'addl_output_layer_sets_idx'[i]는 output_layer_flag[addlIdx][j]가 존재하는 계층 세트의 인덱스 addlIdx를 식별한다.
- [0368] 1과 동일한 'output_layer_flag'[addlIdx][j]는, j와 동일한 nuh_layer_id를 갖는 계층이 addlIdx번째 계층 세트의 타겟 출력 계층인 것으로 특정한다. 0과 동일한 output_layer_flag[addlIdx][j]의 값은 j와 동일한 nuh_layer_id를 갖는 계층이 addlIdx번째 계층 세트의 타겟 출력 계층이 아닌 것으로 특정한다.
- [0369] output_layer_flag[addlIdx][j]가 0 내지 vps_num_layer_sets_minus1까지의 범위에서의 addlIdx에 대해, 그리고 0 내지 63까지의 범위에서의 j에 대해 존재하지 않을 때, output_layer_flag[addlIdx][j]는 (j = = LayerSetLayerIdList[addlIdx][NumLayersInIdList[addlIdx] - 1])과 동일한 것으로 추론된다.
- [0370] output_layer_flag[addlIdx][j]가 0 내지 vps_num_layer_sets_minus1까지의 범위에서의 addlIdx에 대해, 그리고 0 내지 63까지의 j에 대해 존재하지 않을 때, 그리고 output_layer_flag[clsIdx][j]가 0 내지 vps_num_layer_sets_minus1까지의 범위에서의 clsIdx에 대해, 그리고 0 내지 63까지의 범위에서의 j에 대해 존재하지 않을 때, output_layer_flag[addlIdx][j]는 (j = = LayerSetLayerIdList[addlIdx][NumLayersInIdList[addlIdx] - 1])과 동일한 것으로 추론된다.
- [0371] output_layer_flag[addlIdx][j]가 0 내지 vps_num_layer_sets_minus1까지의 범위에서의 addlIdx에 대해, 그리고 0 내지 63까지의 j에 대해 존재하지 않을 때, 그리고 output_layer_flag[clsIdx][j]가 0 내지 vps_num_layer_sets_minus1까지의 범위에서의 clsIdx에 대해, 그리고 0 내지 63까지의 범위에서의 j에 대해 존재하지 않을 때, output_layer_flag[clsIdx][j]는 (j = = LayerSetLayerIdList[clsIdx][NumLayersInIdList[clsIdx] - 1])과 동일한 것으로 추론된다.
- [0372] 출력 계층들의 타겟 리스트는 희망하는 경우에 유도될 수도 있다. 선택된 출력 계층 세트에 대한 출력 계층들인 nuh_layer_id 값들의 증가하는 순서에서, nuh_layer_id 값들의 리스트를 특정하는 타겟 출력 계층 식별자 리스트, targetOpLayerIdList는 다음과 같이 유도될 수도 있다:
- [0373] 선택된 출력 계층 세트 oLsIdx에 대하여, 타겟 출력 계층 식별자 리스트 targetOpLayerIdList는 하기로서 유도된다:

표 11

for(k=0, numOutputLayers=0;k<=vps_max_layer_id;k++)
if(output_layer_flag[oLsIdx][k])
targetOpLayerIdList[numOutputLayers++]=layer_id_in_nuh[k]

- [0374]
- [0375] 타겟 출력 계층 세트는 연관된 타겟 출력 계층 식별자 리스트 targetOpLayerIdList에 의해 식별될 수도 있다.
- [0376] 선택된 출력 계층 세트 oLsIdx는, 출력 계층 세트들 변경 SEI 메시지들이 없을 경우에 0 및 num_output_layer_sets 사이이다.
- [0377] 선택된 출력 계층 세트 oLsIdx는 출력 계층 세트들 변경 SEI 메시지들이 존재할 경우에 0 및 num_output_layer_sets + num_addl_output_layer_sets 사이이다.
- [0378] 디코딩된 계층들의 타겟 리스트는 희망하는 경우에 유도될 수도 있다. 선택된 출력 계층 세트에 대해 디코딩될 필요가 있는 nuh_layer_id 값들의 리스트를 특정하는 타겟 디코딩된 계층 식별자 리스트, targetDLayerIdList는 다음과 같이 유도될 수도 있다:
- [0379] 선택된 출력 계층 세트 oLsIdx에 대하여, 타겟 출력 계층 식별자 리스트 targetOpLayerIdList는 하기로서 유도된다:

표 12

for(k=0, numOutputLayers=0;k<=vps_max_layer_id;k++)
if(output_layer_flag[oLsIdx][k])
targetOpLayerIdList [numOutputLayers++]=layer_id_in_nuh[k]

[0380]

[0381] 타겟 디코딩된 계층 식별자 리스트 targetDLayerIdList는 하기로써 유도될 수도 있다:

표 13

for(m=0, numDecodedLayers=0;m< numOutputLayers;m++) {
for(n=0;n<NumDirectRefLayers[LayerIdInVps[targetOpLayerIdList[m]]];n++) {
rLid=RefLayerId[LayerIdInVps[targetOpLayerIdList[m]]][n]
if(rLid not included in targetDLayerIdList[0,..., numDecodedLayers])
targetDLayerIdList[numDecodedLayers++]=rLid;
}
}

[0382]

[0383] 일부의 경우에는, 최종 targetDLayerIdList는 nuh_layer_id 값들의 증가하는 순서로 상기 targetDLayerIdList를 소팅(sorting)함으로써 획득될 수도 있다.

[0384] 타겟 디코딩된 계층 세트는 연관된 타겟 디코딩된 계층 식별자 리스트 targetDLayerIdList에 의해 식별된다. 일부의 경우에 있어서, targetDLayerIdList는, 디코딩되어야 할 NAL 유닛들의, nuh_layer_id 값들의 증가하는 순서에서 nuh_layer_id 값들의 리스트를 특정하는, JCTVC-L1008로부터의 계층 식별자 리스트 TargetDeclayerIdList와 동일할 수도 있다.

[0385] 선택된 출력 계층 세트 oLsIdx는, 출력 계층 세트들 변경 SEI 메시지들이 없을 경우에 0 및 num_output_layer_sets 사이이다.

[0386] 선택된 출력 계층 세트 oLsIdx는 출력 계층 세트들 변경 SEI 메시지들이 존재할 경우에 0 및 num_output_layer_sets + num_add1_output_layer_sets 사이이다.

[0387] 도 37을 참조하면, 또 다른 SEI 메시지 신택스는 출력 계층 세트들의 변경, 즉, output_layer_sets_change(payloadSize)를 가능하게 하기 위하여 이용될 수도 있다.

[0388] 'num_deleted_output_layer_sets'는, 삭제되어 더 이상 존재하지 않는 출력 계층 세트들의 수를 특정한다. num_deleted_output_layer_sets의 값은 0 내지 num_output_layer_sets까지의 범위에 있다. 존재하지 않을 때, num_deleted_output_layer_sets의 값은 0과 동일한 것으로 추론된다.

[0389] 'deleted_output_layer_sets_idx_entry'[i]는, 삭제되어 더 이상 존재하지 않는 것으로 표시되는 출력 계층 세트 엔트리들의 리스트에서 인덱스 엔트리를 식별한다. deleted_output_layer_set_idx_entry[i]의 값은 0 내지 num_output_layer_sets까지의 범위에 있어야 한다. deleted_output_layer_sets_idx_entry[i]는 i와 동일하지 않은 임의의 j에 대하여 deleted_output_layer_sets_idx_entry[j]와 동일하지 않다.

[0390] 'num_changed_output_layer_sets'는, 출력 계층들이 active_vps_id에 의해 식별된 활성 비디오 파라미터 세트의 vps 확장 섹션에서 특정되는 변경된 출력 계층 세트들의 수를 특정한다. num_changed_output_layer_sets의 값은 0 내지 num_output_layer_sets - num_deleted_output_layer_sets까지의 범위에 있어야 한다. 존재하지

않을 때, num_changed_output_layer_sets의 값은 0과 동일한 것으로 추론된다.

[0391] 출력 계층들의 타겟 리스트는 희망하는 경우에 유도될 수도 있다. 선택된 출력 계층 세트에 대한 출력 계층들인 nuh_layer_id 값들의 증가하는 순서에서, nuh_layer_id 값들의 리스트를 특정하는 타겟 출력 계층 식별자 리스트, targetOpLayerIdList는 다음과 같이 유도될 수도 있다:

[0392] 선택된 출력 계층 세트 oLsIdx에 대하여, 타겟 출력 계층 식별자 리스트 targetOpLayerIdList는 하기로서 유도된다:

표 14

for(k=0, numOutputLayers=0;k<=vps_max_layer_id;k++)
if(output_layer_flag[oLsIdx][k])
targetOpLayerIdList [numOutputLayers++]=layer_id_in_nuh[k]

[0393]

[0394] 타겟 출력 계층 세트는 연관된 타겟 출력 계층 식별자 리스트 targetOpLayerIdList에 의해 식별된다.

[0395] 선택된 출력 계층 세트 oLsIdx는, 출력 계층 세트들 변경 SEI 메시지들이 없을 경우에 0 및 num_output_layer_sets 사이이다.

[0396] 선택된 출력 계층 세트 oLsIdx는 출력 계층 세트들 변경 SEI 메시지들이 존재할 경우에 0 및 num_output_layer_sets - num_deleted_output_layer_sets + num_addl_output_layer_sets 사이이다.

[0397] 디코딩된 계층들의 타겟 리스트는 희망하는 경우에 유도될 수도 있다.

[0398] 선택된 출력 계층 세트에 대해 디코딩될 필요가 있는 nuh_layer_id 값들의 리스트를 특정하는 타겟 디코딩된 계층 식별자 리스트, TargetDLayerIdList는 다음과 같이 유도될 수도 있다:

[0399] 선택된 출력 계층 세트 oLsIdx에 대하여, 타겟 출력 계층 식별자 리스트 targetOpLayerIdList는 하기로서 유도될 수도 있다:

표 15

for(k=0, numOutputLayers=0;k<=vps_max_layer_id;k++)
if(output_layer_flag[oLsIdx][k])
targetOpLayerIdList [numOutputLayers++]=layer_id_in_nuh[k]

[0400]

[0401] 다음으로, 타겟 디코딩된 계층 식별자 리스트 targetDLayerIdList는 하기로서 유도될 수도 있다:

표 16

for(m=0, numDecodedLayers=0;m< numOutputLayers;m++) {
for(n=0;n<NumDirectRefLayers[LayerIdInVps[targetOpLayerIdList[m]]];n++) {
rLid=RefLayerId[LayerIdInVps[targetOpLayerIdList[m]]][n]
if(rLid not included in targetDLayerIdList[0,..., numDecodedLayers])
targetDLayerIdList[numDecodedLayers++]=rLid;
}
}

[0402]

[0403] 일부의 경우들에는, 최종 targetDLayerIdList는 nuh_layer_id 값들의 증가하는 순서로 상기 targetDLayerIdList를 소팅(sorting)함으로써 획득될 수도 있다.

[0404] 타겟 디코딩된 계층 세트는 연관된 타겟 디코딩된 계층 식별자 리스트 targetDLayerIdList에 의해 식별된다. 일부의 경우들에 있어서, targetDLayerIdList는, 디코딩되어야 할 NAL 유닛들의, nuh_layer_id 값들의 증가하는 순서에서 nuh_layer_id 값들의 리스트를 특정하는, JCTVC-L1008로부터의 계층 식별자 리스트 TargetDecLayerIdList와 동일할 수도 있다.

[0405] 선택된 출력 계층 세트 oLsIdx는, 출력 계층 세트들 변경 SEI 메시지들이 없을 경우에 0 및 num_output_layer_sets 사이이다.

[0406] 선택된 출력 계층 세트 oLsIdx는 출력 계층 세트들 변경 SEI 메시지들이 존재할 경우에 0 및 num_output_layer_sets - num_deleted_output_layer_sets + num_add1_output_layer_sets 사이이다.

[0407] 출력 계층 세트들 변경 SEI 메시지는 효과가 누적적이지 않도록 정의된다. 또 다른 실시형태에서, 이 메시지 선택스 및/또는 시맨틱들은 이들이 누적적이도록 정의될 수 있다. 이 경우, 디코더(및 인코더)는 (vps 확장으로부터) num_output_layer_sets 및/또는 num_deleted_output_layer_sets 및/또는 num_changed_output_layer_sets 및/또는 num_add1_output_layer_sets의 모든 이전 값들을 추적할 수 있고, 새로운 SEI 메시지가 시그널링될 때마다 이 변경들을 누적시킬 수 있다.

[0408] 또 다른 실시형태에서, 선택스 엘리먼트들 중의 하나 이상은 ue(v) 대신의 u(v) 대신으로 알려진 고정된 수의 비트들을 이용하여 시그널링될 수도 있다. 예를 들어, 이들은 u(8) 또는 u(16) 또는 u(32) 또는 u(64) 등을 이용하여 시그널링될 수 있다.

[0409] 또 다른 실시형태에서, 이 선택스 엘리먼트 중의 하나 이상은 u(v) 코딩과 같은 고정된 수의 비트들 대신으로, ue(v) 또는 일부의 다른 코딩 방식으로 시그널링될 수 있다.

[0410] 또 다른 실시형태에서, 다양한 선택스 엘리먼트들 및 그 시맨틱들의 명칭들은 plus1 또는 plus2를 가산함으로써, 또는 설명된 선택스 및 시맨틱들에 비해 minus1 또는 minus2를 감산함으로써 변경될 수도 있다.

[0411] 또 다른 실시형태에서, 출력 계층 세트들 SEI 메시지 내에 포함된 다양한 선택스 엘리먼트들은 비트스트림 내의 어딘가에서 픽처 당 또는 다른 주파수에서 시그널링될 수도 있다. 예를 들어, 이들은 슬라이스 세그먼트 헤더, pps/ sps/ vps/ 적응 파라미터 세트 또는 임의의 다른 파라미터 세트 또는 비트스트림의 다른 규범 부분에서 시그널링될 수도 있다.

[0412] 또 다른 실시형태에서, 다양한 선택스 엘리먼트들은 비트스트림 내의 어딘가에서 픽처 당 또는 다른 주파수에서 시그널링될 수도 있다. 예를 들어, 이들은 슬라이스 세그먼트 헤더, pps/ sps/ vps/ 적응 파라미터 세트 또는 임의의 다른 파라미터 세트 또는 비트스트림의 다른 규범 부분에서 시그널링될 수도 있다.

[0413] 또 다른 실시형태들에서, 출력 계층 세트들에 관련된 이 발명에서 정의된 모든 개념들은 JCTVC-L0452 및 JCTVC-L0453에서 정의된 바와 같은 출력 동작 포인트들, 및/또는 JCTVC-L1003에서 정의된 바와 같은 동작 포인트들에

적용될 수 있다.

[0414]

예 2

[0415]

본 발명의 또 다른 예는 이하에서 설명된다. 편의상, 제 1 예에서 예시된 각각의 부재들의 기능들과 동일한 기능들을 가지는 부재들에는 각각의 동일한 참조 번호들이 부여되고, 그 부재들의 설명은 여기서 생략되는 것에 주목해야 한다.

[0416]

DPB로부터의 픽처들의 출력 및 제거를 위한 `sps_max_num_reorder_pics[HighestTid]`, `sps_max_latency_increase_plus1[HighestTid]` 및 `sps_max_dec_pic_buffering[HighestTid]` 신택스 엘리먼트들을 이용하는 것에 기반으로 하여 출력 순서 디코딩된 픽처 버퍼(DPB)를 시그널링하는 것은, 콘텐츠가 사용자의 시청 선호도들에 기반으로 하여 인코딩된 후에 변동되는 경향이 있는 상이한 수들의 강화 계층들, 및 콘텐츠가 사용자의 시청 선호도들에 기반으로 하여 인코딩된 후에 변동되는 경향이 있는 멀티-뷰 강화 계층들이 이용될 때와 같은, 스케일러블 비디오 코딩으로부터 기인할 수도 있는 버퍼 특성들을 고려하지 않는 것으로 결정되었다. 또한, DPB로부터의 픽처들의 출력 및 제거를 위한 `sps_max_num_reorder_pics[HighestTid]`, `sps_max_latency_increase_plus1[HighestTid]`, 및 `sps_max_dec_pic_buffering[HighestTid]` 신택스 엘리먼트들을 이용하는 것에 기반으로 하여 출력 순서 디코딩된 픽처 버퍼(DPB)를 시그널링하는 것은, 디코더가 어떤 동작 포인트에서 동작하고 및/또는 선택된 출력 계층 세트를 출력하고 있을 때의 DPB의 메모리 용법의 측면에서 최적 이 아닐 수도 있는 것으로 결정된다. 시청 선호도들에서의 이러한 차이들을 수용하기 위하여, 출력 순서 디코딩된 픽처 버퍼(DPB)는 추가로 및/또는 대안적으로, 강화 계층들 중의 하나 이상에 대한 신택스 엘리먼트들을 제공하기 위하여 비디오 파라미터 세트 확장(VPS 확장)과 함께 포함되는 이러한 신택스 엘리먼트들에 기반으로 할 수도 있다. 이러한 방식으로, 신택스 엘리먼트들은, 사용자의 시청 선호도들에 대응하는 경향이 있는 특별한 동작 포인트 또는 출력 계층 세트에 대해 특히 적당한 것으로 선택될 수도 있다.

[0417]

DPB 버퍼링 관련 파라미터들, `vps_max_dec_pic_buffering_minus1`, `vps_max_num_reorder_pics`, `vps_max_latency_increase_plus1`은 하나 이상의 동작 포인트들에 대한 CVS에 대한 서브-계층들, 및/또는 VPS 확장에서의 출력 계층 세트들에 대해 시그널링될 수도 있다. 유사하게, 시스템은 VPS 확장으로부터의 상기 시그널링된 DPB 버퍼링 파라미터들이 테스트 중인 동작 포인트에 대하여, 또는 선택된 출력 계층 세트에 대하여 시그널링될 경우에, VPS로부터의 상기 시그널링된 DPB 버퍼링 파라미터들을 이용하기 위하여 출력 순서 DPB에 대한 동작 및 범핑 프로세스를 정의할 수도 있다. 이와 다를 경우, 현재의 계층의 `layer_id`에 따라, 활성 SPS (현재의 픽처의 `nuh_layer_id`에 대응하는 `currLayerId`가 0과 동일할 때)로부터, 또는 활성 계층 SPS로부터의 대응하는 SPS 레벨 파라미터들이 이용된다. 일부의 경우들에는, 파라미터들 `vps_max_dec_pic_buffering_minus1`, `vps_max_num_reorder_pics`, `vps_max_latency_increase_plus1`은 `max_vps_dec_pic_buffering_minus1`, `max_vps_num_reorder_pics`, `max_vps_latency_increase_plus1`로서 칭해질 수도 있다.

[0418]

또 다른 경우에는, DPB 버퍼링 관련 파라미터 `max_vps_dec_pic_buffering_minus1`은 VPS 확장에서의 하나 이상의 출력 계층 세트들에 대한 CVS에 대한 서브-계층들에 대해 시그널링된다. 출력 계층 세트들은 동작 포인트들에 대응할 수도 있다. DPB 버퍼링 관련 파라미터들 `max_vps_num_reorder_pics`, `max_vps_latency_increase_plus1`은 VPS 확장에서의 각각의 계층에 대한 CVS에 대한 서브-계층들에 대해 시그널링될 수도 있다. 유사하게, 시스템은 VPS 확장으로부터의 상기 시그널링된 DPB 버퍼링 파라미터들이 테스트 중인 동작 포인트에 대하여, 또는 선택된 출력 계층 세트에 대하여 시그널링될 경우에, VPS로부터의 상기 시그널링된 DPB 버퍼링 파라미터들을 이용하기 위하여 출력 순서 DPB에 대한 동작 및 범핑 프로세스를 정의할 수도 있다. 이와 다를 경우, 현재의 계층의 `layer_id`에 따라, 활성 SPS(현재의 픽처의 `nuh_layer_id`에 대응하는 `currLayerId`가 0과 동일할 때)로부터, 또는 활성 계층 SPS로부터의 대응하는 SPS 레벨 파라미터들이 이용된다.

[0419]

도 38a를 참조하면, 예시적인 수정된 `vps_extension`이 예시되어 있다. 수정된 `vps` 확장은 새로운 신택스, 즉, `num_op_dpb_info_parameters` 및 `operation_point_layer_set_idx[i]`를 포함한다. 이 수정된 `vps` 확장은 또 다른 비트스트림, 타겟 최고 `TemporalId`, 및 타겟 계층 식별자 리스트를 입력들로서 갖는 서브-비트스트림 추출 프로세스의 동작에 의해 또 다른 비트스트림으로부터 생성된 비트스트림인 동작 포인트의 측면에서 정의될 수도 있다.

[0420]

`num_output_layer_sets`는, 출력 계층들이 `output_layer_set_index[i]` 및 `output_layer_flag[lsIdx][j]`로 특정되는 계층 세트들의 수를 특정한다. 존재하지 않을 때, `num_output_layer_sets`의 값은 0과 동일한 것으로 추론된다. 출력 계층들을 설명하는 계층 세트는 출력 계층 세트이다.

- [0421] output_layer_set_idx[i]는, output_layer_flag[lIdx][j]가 존재하는 계층 세트의 인덱스 lIdx를 특정한다.
- [0422] 1과 동일한 output_layer_flag[lIdx][j]는 j와 동일한 nuh_layer_id를 갖는 계층이 lIdx번째 계층 세트의 타겟 출력 계층인 것으로 특정한다. 0과 동일한 output_layer_flag[lIdx][j]의 값은 j와 동일한 nuh_layer_id를 갖는 계층이 lIdx번째 계층 세트의 타겟 출력 계층이 아닌 것으로 특정한다.
- [0423] num_op_dpb_info_parameters는 동작 포인트의 측면에서 정의된, VPS 확장 RBSP에서 존재하는 op_dpb_parameters() 선택스 구조들의 수를 특정한다. num_op_dpb_info_parameters 디코더들은 0 내지 vps_num_layer_sets_minus1까지의 범위에 있다.
- [0424] operation_point_layer_set_idx[i]는 VPS 확장에서의 i번째 op_dpb_info_parameters() 선택스 구조가 적용되는 동작 포인트들에 의해 정의된 계층 세트들의 리스트로의 인덱스를 특정한다. operation_point_layer_set_idx[i]의 값은 0 내지 vps_num_layer_sets_minus1까지의 범위에 있을 수도 있다. 비트스트림 적합성을 위하여, operation_point_layer_set_idx[i]는 i와 동일하지 않은 임의의 j에 대하여 operation_point_layer_set_idx[j]와 동일하지 않다.
- [0425] 도 39a를 참조하면, op_dpb_info_parameters는 vps_max_sub_layers_minus1[j], vps_sub_layer_ordering_info_present_flag[j], vps_max_dec_pic_buffering_minus1[j][k], vps_max_num_reorder_pics[j][k], 및 vps_max_latency_increase_plus1[j][k]를 특정한다.
- [0426] vps_max_sub_layers_minus1[j] 플러스 1은 얼마나 많은 서브 계층들이 포함되는지를 표시한다. vps_max_sub_layers_minus1[j] 플러스 1은, j와 동일한 nuh_layer_id를 갖는 계층에 대한 CVS에서 존재할 수도 있는 시간적 서브-계층들의 최대 수를 특정한다. vps_max_sub_layers_minus1[j]의 값은 0 내지 6까지의 범위에 있다.
- [0427] vps_sub_layer_ordering_info_present_flag[j]는 선택스가 모든 계층들을 포함하는 하나의 세트에 대한 것인지 또는 각각의 개별적인 계층에 대한 것인지 여부를 표시한다. 1과 동일한 vps_sub_layer_ordering_info_present_flag[j]는, vps_max_dec_pic_buffering_minus1[j][k], vps_max_num_reorder_pics[j][k], 및 vps_max_latency_increase_plus1[j][k]이 vps_max_sub_layers_minus1[j] + 1 서브-계층들에 대한 j와 동일한 nuh_layer_id를 갖는 계층들에 대하여 존재하는 것으로 특정한다. 0과 동일한 vps_sub_layer_ordering_info_present_flag[j]는, vps_max_dec_pic_buffering_minus1[j][vps_max_sub_layers_minus1[j]], vps_max_num_reorder_pics[j][vps_max_sub_layers_minus1[j]], 및 vps_max_latency_increase_plus1[j][vps_max_sub_layers_minus1[j]]의 값들이 j와 동일한 nuh_layer_id를 갖는 계층에 대한 모든 서브-계층들에 대해 적용되는 것으로 특정한다.
- [0428] vps_max_dec_pic_buffering_minus1[j][k] 플러스 1은, HighestTid가 k와 동일할 때 픽처 저장 버퍼들의 단위로, j와 동일한 nuh_layer_id를 갖는 계층에 대한 CVS에 대한 디코딩된 픽처 버퍼의 최대 요구된 사이즈를 특정한다. vps_max_dec_pic_buffering_minus1[j][k]의 값은 0 내지 (하위조항 A.4에서 특정된 바와 같이) MaxDpbSize - 1까지의 범위에 있을 것이다. k가 0보다 더 클 때, vps_max_dec_pic_buffering_minus1[j][k]는 vps_max_dec_pic_buffering_minus1[j][k - 1] 이상일 것이다. vps_max_dec_pic_buffering_minus1[j][k]이 0 내지 vps_max_sub_layers_minus1[j] - 1까지의 범위에서의 k에 대해 존재하지 않을 때, vps_sub_layer_ordering_info_present_flag[j]가 0과 동일한 것으로 인해, 그것은 vps_max_dec_pic_buffering_minus1[j][vps_max_sub_layers_minus1[j]]와 동일한 것으로 추론된다.
- [0429] vps_max_num_reorder_pics[j][k]는, HighestTid가 k와 동일할 때, 디코딩 순서에서 j와 동일한 nuh_layer_id를 갖는 계층에 대한 CVS에서의 임의의 픽처를 선행할 수 있으며 출력 순서에서 그 픽처를 후행할 수 있는 픽처들의 최대 허용된 수를 표시한다. vps_max_num_reorder_pics[j][k]의 값은 0 내지 vps_max_dec_pic_buffering_minus1[j][k]까지의 범위에 있을 것이다. k가 0보다 더 클 때, vps_max_num_reorder_pics[j][k]는 vps_max_num_reorder_pics[j][k - 1] 이상이다. vps_max_num_reorder_pics[j][k]가 0 내지 vps_max_sub_layers_minus1[j] - 1까지의 범위에서의 k에 대해 존재하지 않을 때, vps_sub_layer_ordering_info_present_flag[j]가 0과 동일한 것으로 인해, 그것은 vps_max_num_reorder_pics[j][vps_max_sub_layers_minus1[j]]와 동일한 것으로 추론된다.
- [0430] 0과 동일하지 않은 vps_max_latency_increase_plus1[j][k]은, HighestTid가 k와 동일할 때, 출력 순서에서

j와 동일한 nuh_layer_id를 갖는 계층에 대한 CVS에서의 임의의 픽처를 선행할 수 있으며 디코딩 순서에서 그 픽처를 후행할 수 있는 픽처들의 최대 수를 특정하는 VpsMaxLatencyPictures[j][k]의 값을 연산하기 위하여 이용된다.

- [0431] vps_max_latency_increase_plus1[j][k]이 0과 동일하지 않을 때, VpsMaxLatencyPictures[j][k]의 값은 다음과 같이 특정될 수도 있다:
- [0432] $VpsMaxLatencyPictures[j][k] = vps_max_num_reorder_pics[j][k] + vps_max_latency_increase_plus1[j][k] - 1$
- [0433] vps_max_latency_increase_plus1[j][k]이 0과 동일할 때, 대응하는 한계가 표현되지 않는다.
- [0434] vps_max_latency_increase_plus1[j][k]의 값이 0 내지 $2^{32} - 2$ 까지의 범위에 있다. vps_max_latency_increase_plus1[j][k]가 0 내지 vps_max_sub_layers_minus1[j] - 1까지의 범위에서의 k에 대해 존재하지 않을 때, vps_sub_layer_ordering_info_present_flag[j]가 0과 동일한 것으로 인해, 그것은 vps_max_latency_increase_plus1[j][vps_max_sub_layers_minus1[j]]과 동일한 것으로 추론된다.
- [0435] 'vps_max_sub_layers_minus1'[id][j] 플러스 1은, 인덱스 id와 연관된 동작 포인트에 대하여 j와 동일한 nuh_layer_id를 갖는 계층에 대한 CVS에서 존재할 수도 있는 시간적 서브-계층들의 최대 수를 특정한다. vps_max_sub_layers_minus1[id][j]의 값은 0 내지 6까지의 범위에 있을 것이다.
- [0436] 1과 동일한 'vps_sub_layer_ordering_info_present_flag'[id][j]는, vps_max_dec_pic_buffering_minus1[id][j][k], vps_max_num_reorder_pics[id][j][k], 및 vps_max_latency_increase_plus1[id][j][k]가 vps_max_sub_layers_minus1[id][j] + 1 서브-계층들에 대한 인덱스 id와 연관된 동작 포인트에 대하여 j와 동일한 nuh_layer_id를 갖는 계층에 대하여 존재하는 것으로 특정한다.
- [0437] 0과 동일한 vps_sub_layer_ordering_info_present_flag[id][j]는, vps_max_dec_pic_buffering_minus1[id][j][vps_max_sub_layers_minus1[id][j]], vps_max_num_reorder_pics[id][j][vps_max_sub_layers_minus1[id][j]], 및 vps_max_latency_increase_plus1[id][j][vps_max_sub_layers_minus1[id][j]]의 값들이 인덱스 id와 연관된 동작 포인트에 대하여 j와 동일한 nuh_layer_id를 갖는 계층에 대한 모든 서브-계층들에 적용되는 것으로 특정한다.
- [0438] 'vps_max_dec_pic_buffering_minus1'[id][j][k] 플러스 1은, HighestTid가 k와 동일할 때, 픽처 저장 버퍼들의 단위로 인덱스 id와 연관된 동작 포인트에 대하여 j와 동일한 nuh_layer_id를 갖는 계층에 대한 CVS에 대한 디코딩된 픽처 버퍼의 최대 요구된 사이즈를 특정한다. vps_max_dec_pic_buffering_minus1[id][j][k]의 값은 0 내지 (하위조항 A.4에서 특정된 바와 같이) MaxDpbSize - 1까지의 범위에 있을 것이다. k가 0보다 더 클 때, vps_max_dec_pic_buffering_minus1[id][j][k]는 vps_max_dec_pic_buffering_minus1[id][j][k - 1] 이상일 것이다. vps_max_dec_pic_buffering_minus1[id][j][k]가 0 내지 vps_max_sub_layers_minus1[id][j] - 1까지의 범위에서 k에 대해 존재하지 않을 때, vps_sub_layer_ordering_info_present_flag[id][j]가 0과 동일함으로 인해, 그것은 vps_max_dec_pic_buffering_minus1[id][j][vps_max_sub_layers_minus1[id][j]]과 동일한 것으로 추론된다.
- [0439] 'vps_max_num_reorder_pics'[id][j][k]는, HighestTid가 k와 동일할 때, 디코딩 순서에서 인덱스 id와 연관된 동작 포인트에 대한 j와 동일한 nuh_layer_id를 갖는 계층에 대한 CVS에서의 임의의 픽처를 선행할 수 있으며 출력 순서에서 그 픽처를 후행할 수 있는 픽처의 최대 허용된 수를 표시한다. vps_max_num_reorder_pics[id][j][k]의 값은 0 내지 vps_max_dec_pic_buffering_minus1[id][j][k]까지의 범위에 있을 것이다. k가 0보다 더 클 때, vps_max_num_reorder_pics[id][j][k]는 vps_max_num_reorder_pics[id][j][k - 1] 이상일 것이다. vps_max_num_reorder_pics[id][j][k]가 0 내지 vps_max_sub_layers_minus1[id][j] - 1까지의 범위에서 k에 대해 존재하지 않을 때, vps_sub_layer_ordering_info_present_flag[id][j]가 0과 동일함으로 인해, 그것은 vps_max_num_reorder_pics[id][j][vps_max_sub_layers_minus1[id][j]]과 동일한 것으로 추론된다.
- [0440] 0과 동일하지 않은 'vps_max_latency_increase_plus1'[id][j][k]은, HighestTid가 k와 동일할 때, 출력 순서에서 인덱스 id와 연관된 동작 포인트에 대한 j와 동일한 nuh_layer_id를 갖는 계층에 대한 CVS에서의 임의의 픽처를 선행할 수 있으며 디코딩 순서에서 그 픽처를 후행할 수 있는 픽처들의 최대 수를 특정하는

VpsMaxLatencyPictures[id][j][k]의 값을 연산하기 위하여 이용된다.

- [0441] vps_max_latency_increase_plus1[id][j][k]이 0과 동일하지 않을 때, VpsMaxLatencyPictures[id][j][k]의 값은 다음과 같이 특정된다:
- [0442]
$$\text{VpsMaxLatencyPictures}[id][j][k] = \text{vps_max_num_reorder_pics}[id][j][k] + \text{vps_max_latency_increase_plus1}[id][j][k] - 1$$
- [0443] vps_max_latency_increase_plus1[id][j][k]이 0과 동일할 때, 대응하는 한계가 표현되지 않는다.
- [0444] vps_max_latency_increase_plus1[id][j][k]의 값은 0 내지 $2^{32} - 2$ 까지의 범위에 있을 것이다. vps_max_latency_increase_plus1[id][j][k]가 0 내지 vps_max_sub_layers_minus1[id][j] - 1까지의 범위에서 k에 대해 존재하지 않을 때, vps_sub_layer_ordering_info_present_flag[id][j]가 0과 동일함으로 인해, 그것은 vps_max_latency_increase_plus1[id][j][vps_max_sub_layers_minus1[id][j]]과 동일한 것으로 추론된다.
- [0445] 도 39b를 참조하면, op_dpb_info_parameters는 도시된 바와 같이 op_dpb_info_parameters(id,j)로 추가로 수정될 수도 있다. 이 경우, VPS 확장의 선택스는 도 38b에서 예시된 바와 같을 수도 있다. 가상 참조 디코더(HRD)는 비트스트림 및 디코더 적합성을 검사하기 위하여 이용된다. 비트스트림들 또는 비트스트림 서브세트들의 2 개의 타입들은 비디오 코딩에 관한 합동 협력 팀(Joint Collaborative Team on Video Coding; JCT-VC)을 위한 HRD 적합성 검사를 받는다. 타입 I 비트스트림으로 칭해진 제 1 타입은 비트스트림에서의 모든 액세스 유닛들에 대한 VCL NAL 유닛들과, FD-NUT 와 동일한 nal_unit_type 을 갖는 NAL 유닛들 (필러 데이터 NAL 유닛들) 만을 포함하는 NAL 유닛 스트림이다. 타입 II 비트스트림으로 칭해진 제 2 타입은 비트스트림에서의 모든 액세스 유닛들에 대한 VCL NAL 유닛들 및 필러 데이터 NAL 유닛들에 추가하여, (a) 필러 데이터 NAL 유닛들 이외의 추가적인 비-VCL NAL 유닛들; 및 (b) NAL 유닛 스트림으로부터 바이트 스트림을 형성하는 모든 leading_zero_8bits, zero_byte, start_code_prefix_one_3bytes, 및 trailing_zero_8bits 선택스 엘리먼트들 중의 적어도 하나를 포함한다.
- [0446] HRD에 대해 요구된 비-VCL NAL 유닛들의 선택스 엘리먼트들(또는 선택스 엘리먼트들의 일부에 대한 그 디폴트 값들)은 부록들 D 및 E, 조항 7의 시맨틱 하위조항들에서 특정된다.
- [0447] HRD 파라미터 세트들(NAL HRD 파라미터들 및 VCL HRD 파라미터들)의 2 개의 타입들이 이용된다. HRD 파라미터 세트들은, SPS 선택스 구조 또는 VPS 선택스 구조의 일부일 수도 있는 hrd_parameters() 선택스 구조를 통해 시그널링된다.
- [0448] 다수의 테스트들은 테스트 중인 비트스트림으로서 지칭되는 비트스트림의 적합성을 검사하기 위하여 필요하게 될 수도 있다. 각각의 테스트에 대하여, 다음의 단계들이 열거된 순서로 적용된다:
- [0449] (1) TargetOp로서 나타낸 테스트 중인 동작 포인트가 선택된다. TargetOp의 계층 식별자 리스트 OpLayerIdList는, 테스트 중인 비트스트림에서 존재하는 nuh_layer_id 값들의 서브세트인, TargetOp와 연관된 비트스트림 서브세트에서 존재하는 nuh_layer_id 값들의 증가하는 순서에서, nuh_layer_id 값들의 리스트로 구성된다. TargetOp의 OpTid는 TargetOp와 연관된 비트스트림 서브세트에서 존재하는 최고 TemporalId와 동일하다.
- [0450] (2) TargetDeclayerIdList는 TargetOp의 OpLayerIdList와 동일하게 설정되고, HighestTid는 TargetOp의 OpTid와 동일하게 설정되고, 조항 10에서 특정된 바와 같은 서브-비트스트림 추출 프로세스는 입력들로서 테스트 중인 비트스트림 HighestTid 및 TargetDeclayerIdList로 호출되고, 출력은 BitstreamToDecode에 배정된다.
- [0451] (3) TargetOp 에 적용가능한 hrd_parameters() 선택스 구조 및 sub_layer_hrd_parameters() 선택스 구조가 선택된다. TargetDeclayerIdList가 테스트 중인 비트스트림에서 존재하는 모든 nuh_layer_id 값들을 포함할 경우, 활성 SPS에서의(또는 이 사양에서 특정되지 않은 외부 수단을 통해 제공된) hrd_parameters() 선택스 구조가 선택된다. 이와 다를 경우, TargetOp에 적용되는 활성 VPS에서의(또는 이 사양에서 특정되지 않은 외부 수단을 통해 제공된) hrd_parameters() 선택스 구조가 선택된다. 선택된 hrd_parameters() 선택스 구조 내에서, BitstreamToDecode가 타입 I 비트스트림일 경우, 조건 "if(vcl_hrd_parameters_present_flag)" 을 바로 후행하는 sub_layer_hrd_parameters(HighestTid) 선택스 구조가 선택되고, 변수 NalHrdModeFlag는 0 과 동일하게 설정되고; 이와 다를 경우(BitstreamToDecode는 타입 II 비트스트림임), 조건 "if(vcl_hrd_parameters_present_flag)" (이 경우, 변수 NalHrdModeFlag는 0과 동일하게 설정됨) 또는 조건 "if(

nal_hrd_parameters_present_flag)" (이 경우, 변수 NalHrdModeFlag는 1과 동일하게 설정됨)의 어느 하나를 바로 후행하는 sub_layer_hrd_parameters(HighestTid) 신택스 구조가 선택된다. BitstreamToDecode가 타입 II 비트스트림이고 NalHrdModeFlag가 0과 동일할 때, 필터 데이터 NAL 유닛들을 제외한 모든 비-VCL NAL 유닛들과, 존재할 때에, (부록 B에서 특정된 바와 같은) NAL 유닛 스트림으로부터의 바이트 스트림을 형성하는 모든 leading_zero_8bits, zero_byte, start_code_prefix_one_3bytes, 및 trailing_zero_8bits 신택스 엘리먼트들은 BitstreamToDecode로부터 폐기되고, 나머지 비트스트림은 BitstreamToDecode에 배정된다.

[0452] 또 다른 경우에는, 다수의 테스트들이 테스트 중인 비트스트림으로서 지칭되는 비트스트림의 적합성을 검사하기 위하여 필요하게 될 수도 있다. 각각의 테스트에 대하여, 다음의 단계들이 열거된 순서로 적용된다:

[0453] (1) TargetOpLs로서 나타낸, 테스트 중인 출력 계층 세트가 선택된다. output_layer_set_idx[]에 의해 TargetOpLs에서 참조된 동작 포인트는 테스트 중인 동작 포인트를 식별한다. TargetOpLs의 출력 계층 식별자 리스트 OpLayerIdList는, 테스트 중인 비트스트림에서 존재하는 nuh_layer_id 값들의 서브세트인, TargetOp 및 TargetOpLs와 연관된 비트스트림 서브세트에서 존재하는 nuh_layer_id 값들의 증가하는 순서에서, nuh_layer_id 값들의 리스트로 구성된다. TargetOp의 OpTid는 TargetOp와 연관된 비트스트림 서브세트에서 존재하는 최고 TemporalId와 동일하다.

[0454] (2) TargetDeclayerIdList는 선택된 출력 계층 세트 TargetOpLs에 대한 타겟 디코딩된 계층 식별자 리스트 targetDLayerIdList와 동일하게 설정되고, HighestTid는 TargetOp의 OpTid와 동일하게 설정되고, 조항 10에서 특정된 바와 같은 서브-비트스트림 추출 프로세스는 입력들로서 테스트 중인 비트스트림 HighestTid 및 TargetDeclayerIdList로 호출되고, 출력은 BitstreamToDecode에 배정된다.

[0455] (3) TargetOp 에 적용가능한 hrd_parameters() 신택스 구조 및 sub_layer_hrd_parameters() 신택스 구조가 선택된다. TargetDeclayerIdList가 테스트 중인 비트스트림에서 존재하는 모든 nuh_layer_id 값들을 포함할 경우, 활성 SPS에서의(또는 이 사양에서 특정되지 않은 외부 수단을 통해 제공된) hrd_parameters() 신택스 구조가 선택된다. 이와 다를 경우, TargetOp에 적용되는 활성 VPS에서의(또는 이 사양에서 특정되지 않은 일부의 외부 수단을 통해 제공된) hrd_parameters() 신택스 구조가 선택된다. 선택된 hrd_parameters() 신택스 구조 내에서, BitstreamToDecode가 타입 I 비트스트림일 경우, 조건 "if(vcl_hrd_parameters_present_flag)" 을 바로 후행하는 sub_layer_hrd_parameters(HighestTid) 신택스 구조가 선택되고, 변수 NalHrdModeFlag는 0 과 동일하게 설정되고; 이와 다를 경우(BitstreamToDecode는 타입 II 비트스트림임), 조건 "if(vcl_hrd_parameters_present_flag)" (이 경우, 변수 NalHrdModeFlag는 0과 동일하게 설정됨) 또는 조건 "if(nal_hrd_parameters_present_flag)" (이 경우, 변수 NalHrdModeFlag는 1과 동일하게 설정됨)의 어느 하나를 바로 후행하는 sub_layer_hrd_parameters(HighestTid) 신택스 구조가 선택된다. BitstreamToDecode가 타입 II 비트스트림이고 NalHrdModeFlag가 0과 동일할 때, 필터 데이터 NAL 유닛들을 제외한 모든 비-VCL NAL 유닛들과, 존재할 때에, (부록 B에서 특정된 바와 같은) NAL 유닛 스트림으로부터의 바이트 스트림을 형성하는 모든 leading_zero_8bits, zero_byte, start_code_prefix_one_3bytes, 및 trailing_zero_8bits 신택스 엘리먼트들은 BitstreamToDecode로부터 폐기되고, 나머지 비트스트림은 BitstreamToDecode에 배정된다.

[0456] 준수하는 디코더는 이 하위조항에서 특정된 모든 요건들을 충족시킬 수도 있다.

[0457] (1) VCL NAL 유닛들에서 참조된 모든 VPS들, SPS들 및 PPS들과, 적절한 버퍼링 주기 및 픽처 타이밍 SEI 메시지들이 비트스트림에서 (비-VCL NAL 유닛들에 의해), 또는 이 사양에서 특정되지 않은 외부 수단에 의해, 시기적 절한 방식으로 디코더로 운반된다면, 특정 프로파일, 티어 및 레벨에 대한 적합성을 요구하는 디코더는 부록 A 에서 특정된 방식으로 하위조항 C.4에서 특정된 비트스트림 적합성 요건들을 준수하는 모든 비트스트림들을 성공적으로 디코딩할 수 있을 것이다.

[0458] (2) 비트스트림이 예약된 것으로 특정되는 값들을 가지는 신택스 엘리먼트들을 포함하며 디코더들이 예약된 값 들을 갖는 신택스 엘리먼트들을 포함하는 NAL 유닛들 또는 신택스 엘리먼트들의 값들을 무시할 것이라고 특정되 고, 비트스트림이 이와 다르게 이 사양을 준수하고 있을 때, 준수하는 디코더는, 그것이 준수하는 비트스트림을 디코딩하는 것과 동일한 방식으로 비트스트림을 디코딩할 것이고, 특정된 바와 같은 예약된 값들을 가지는 신택 스 엘리먼트들을 포함하는 NAL 유닛들 또는 신택스 엘리먼트들의 값들을 무시할 것이다.

[0459] 디코더의 적합성의 2 개의 타입들이 있다: 출력 타이밍 적합성 및 출력 순서 적합성.

[0460] 디코더의 적합성을 검사하기 위하여, 하위조항 C.4에서 특정된 바와 같이, 요구된 프로파일, 티어 및 레벨을 준 수하는 테스트 비트스트림들은 가상 스트림 스케줄러(HSS)에 의해 HRD 및 테스트 중인 디코더(DUT)의 양자로 전

달된다. HRD에 의해 출력된 모든 크롭핑된 디코딩된 픽처들은 또한 DUT에 의해 출력될 것이고, DUT에 의해 출력된 각각의 크롭핑된 디코딩된 픽처는 1과 동일한 PicOutputFlag를 갖는 픽처일 것이고, DUT에 의해 출력된 각각의 이러한 크롭핑된 디코딩된 픽처에 대하여, 출력되는 모든 샘플들의 값들은 특정된 디코딩 프로세스에 의해 생성된 샘플들의 값들과 동일할 것이다.

[0461] 출력 타이밍 디코더 적합성을 위하여, HSS는 비트 레이트 및 CPB 사이즈가 특정된 프로파일, 티어 및 레벨에 대해 부록 A에서 특정된 바와 같이 한정되는 SchedSelIdx의 값들의 서브세트로부터만 선택된 전달 스케줄들로, 또는 비트 레이트 및 CPB 사이즈가 부록 A에서 특정된 바와 같이 한정되는 이하에서 특정된 바와 같은 "보간된 (interpolated)" 전달 스케줄들로, 위에서 설명된 바와 같이 동작한다. 동일한 전달 스케줄은 HRD 및 DUT 양자에 대해 이용된다.

[0462] HRD 파라미터들 및 버퍼링 주기 SEI 메시지들에 0 보다 더 큰 cpb_cnt_minus1[HighestTid]가 제공될 때, 디코더는 피크 비트 레이트 r, CPB size c(r), 및 초기 CPB 제거 지연을 갖는 것으로 특정된 "보간된" 전달 스케줄을 이용하여 HSS 동작으로부터 전달된 바와 같은 비트스트림을 디코딩할 수 있을 것이다.

수학식 15

$$(f(r) \div r)$$

[0463]

[0464] 다음과 같다:

표 17

$\alpha = (r - \text{BitRate}[\text{ SchedSelIdx} - 1]) \div (\text{BitRate}[\text{ SchedSelIdx}] - \text{BitRate}[\text{ SchedSelIdx} - 1]),$	(C-22)
$c(r) = \alpha * \text{CpbSize}[\text{ SchedSelIdx}] + (1 - \alpha) * \text{CpbSize}[\text{ SchedSelIdx} - 1],$	(C-23)
$f(r) = \alpha * \text{InitCpbRemovalDelay}[\text{ SchedSelIdx}] * \text{BitRate}[\text{ SchedSelIdx}] + (1 - \alpha) * \text{InitCpbRemovalDelay}[\text{ SchedSelIdx} - 1] * \text{BitRate}[\text{ SchedSelIdx} - 1]$	(C-24)

[0465]

[0466] 임의의 SchedSelIdx > 0 및 r에 대하여, BitRate[SchedSelIdx - 1] <= r <= BitRate[SchedSelIdx]가 되어, r 및 c(r)은 특정된 프로파일, 티어 및 레벨에 대한 최대 비트 레이트 및 버퍼 사이즈에 대하여 부록 A에서 특정된 바와 같은 한계들 내에 있다. InitCpbRemovalDelay[SchedSelIdx]는 하나의 버퍼링 주기로부터 또 다른 버퍼링 주기까지 상이할 수 있고, 재계산되어야 한다.

[0467] 출력 타이밍 디코더 적합성을 위하여, 위에서 설명된 바와 같은 HRD가 이용되고, 픽처 출력의 (최초 비트의 전달 시간에 관한) 타이밍은 HRD 및 DUT 양자에 대해 고정된 지연까지 동일하다.

[0468] 출력 순서 디코더 적합성을 위하여, 다음이 적용된다:

[0469] (1) HSS는 DUT로부터의 "요구에 의해 (by demand)" 비트스트림 BitstreamToDecode을 DUT로 전달하고, 이것은 DUT가 그 프로세싱을 진행하기 위하여 더 많은 비트들을 요구할 때에만 (디코딩 순서에서) 비트들을 전달한다는 것을 의미한다. 이것은 이 테스트를 위하여, DUT의 코딩된 픽처 버퍼가 최대 디코딩 유닛의 사이즈만큼 작을 수 있다는 것을 의미한다.

[0470] (2) 이하에서 설명된 바와 같은 수정된 HRD가 이용되고, HSS는 비트 레이트 및 CPB 사이즈가 부록 A에서 특정된 바와 같이 한정되도록 비트스트림 BitstreamToDecode에서 특정된 스케줄들 중의 하나에 의해 비트스트림을 HRD로 전달한다. 출력된 픽처들의 순서는 HRD 및 DUT 양자에 대해 동일할 것이다.

[0471] (3) HRD CPB 사이즈는 하위조항 E.2.3에서 특정된 바와 같이 CpbSize[SchedSelIdx]에 의해 주어지고, 여기서, SchedSelIdx 및 HRD 파라미터들은 하위조항 C.1에서 특정된 바와 같이 선택된다. DPB 사이즈는 활성 SPS로부터(현재의 디코딩된 픽처에 대한 nuh_layer_id가 0과 동일할 때), 또는 현재의 디코딩된 픽처의 nuh_layer_id의 값에 대한 활성 계층 SPS로부터의 sps_max_dec_pic_buffering_minus1[HighestTid] + 1에 의해 주어진다. 일부의 경우들에는, 동작 포인트 DPB 정보 파라미터들 op_dpb_info_parameters()가 선택된 출력

계층 세트에 대해 존재할 경우, DPB 사이즈는 currLayerId가 0과 동일할 때에 vps_max_dec_pic_buffering_minus1[HighestTid]에 의해 주어지거나, currLayerId가 0보다 더 클 때에 테스트 중인 동작 포인트에 대한 currLayerId에 대하여 vps_max_dec_pic_buffering_minus1[CurrLayerId][HighestTid]로 설정되고, 여기서, currLayerId는 현재의 디코딩된 픽처의 nuh_layer_id이다. 이와 다르게, 동작 포인트 DPB 정보 파라미터들 op_dpb_info_parameters()이 테스트 중인 동작 포인트에 대해 존재하지 않을 경우, DPB 사이즈는 활성 SPS로부터(현재의 디코딩된 픽처에 대한 nuh_layer_id가 0과 동일할 때), 또는 현재의 디코딩된 픽처의 nuh_layer_id의 값에 대한 활성 계층 SPS로부터의 sps_max_dec_pic_buffering_minus1[HighestTid] + 1에 의해 주어진다.

[0472] 일부의 경우들에는, 출력 계층 세트들 DPB 정보 파라미터들 oop_dpb_info_parameters()이 선택된 출력 계층 세트에 대해 존재할 경우, DPB 사이즈는 currLayerId가 0과 동일할 때에 vps_max_dec_pic_buffering_minus1[HighestTid]에 의해 주어지거나, 선택된 출력 계층 세트에 대한 currLayerId에 대하여 vps_max_dec_pic_buffering_minus1[CurrLayerId][HighestTid]로 설정되고, 여기서, currLayerId는 현재의 디코딩된 픽처의 nuh_layer_id이다. 이와 다르게, 출력 계층 DPB 정보 파라미터들 oop_dpb_info_parameters()이 선택된 출력 계층 세트에 대해 존재하지 않을 경우, DPB 사이즈는 활성 SPS로부터(현재의 디코딩된 픽처에 대한 nuh_layer_id가 0과 동일할 때), 또는 현재의 디코딩된 픽처의 nuh_layer_id의 값에 대한 활성 계층 SPS로부터의 sps_max_dec_pic_buffering_minus1[HighestTid] + 1에 의해 주어진다.

[0473] HRD 에 대한 CPB 로부터의 제거 시간은 최종 비트 도달 시간이고, 디코딩은 즉각적이다. 이 HRD 의 DPB 의 동작은 하위조항들 C.5.2 내지 C.5.2.3에서 설명된 바와 같다.

[0474] 디코딩된 픽처 버퍼는 픽처 저장 버퍼들을 포함한다. 0과 동일한 nuh_layer_id에 대한 픽처 저장 버퍼들의 수는 활성 SPS로부터 유도된다. 각각의 제로가 아닌 nuh_layer_id 값에 대한 픽처 저장 버퍼들의 수는 그 제로가 아닌 nuh_layer_id 값에 대한 활성 계층 SPS로부터 유도된다. 픽처 저장 버퍼들의 각각은 "참조를 위해 이용됨"으로서 표기되거나 미래의 출력을 위해 유지되는 디코딩된 픽처를 포함한다. 하위조항 F.13.5.2.2에서 특정된 바와 같은, DPB로부터의 픽처들의 출력 및 제거를 위한 프로세스가 호출되고, 하위조항 F.13.5.2.3에서 특정된 바와 같은 픽처 디코딩, 표기, 추가적인 범핑, 및 저장을 위한 프로세스의 호출이 뒤따른다. "범핑" 프로세스는 하위조항 F.13.5.2.4에서 특정되고, 하위조항들 F.13.5.2.2 및 F.13.5.2.3에서 특정된 바와 같이 호출된다.

[0475] 현재의 픽처의 디코딩 전의(그러나, 현재의 픽처의 최초 슬라이스의 슬라이스 헤더를 파싱한 후의) DPB로부터의 픽처들의 출력 및 제거는, 현재의 픽처를 포함하는 액세스 유닛의 최초 디코딩 유닛이 CPB로부터 제거될 때에 순간적으로 발생하고 다음과 같이 진행된다.

[0476] 하위조항 8.3.2에서 특정된 바와 같은 RPS에 대한 디코딩 프로세스가 호출된다.

[0477] (1) 현재의 픽처가, 1과 동일한 NoRas1OutputFlag를 가지며 픽처 0이 아닌 0과 동일한 nuh_layer_id를 갖는 IRAP 픽처일 경우, 다음의 순서화된 단계들이 적용된다:

[0478] (A) 변수 NoOutputOfPriorPicsFlag는 다음과 같이 테스트 중인 디코더에 대해 유도된다:

[0479] (i) 현재의 픽처가 CRA 픽처일 경우, NoOutputOfPriorPicsFlag는 (no_output_of_prior_pics_flag의 값에 관계없이) 1과 동일하게 설정된다.

[0480] (ii) 이와 다르게, 활성 SPS로부터 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid]의 값이 선행하는 픽처에 대해 활성인 SPS로부터 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid]의 값과는 각각 상이할 경우, NoOutputOfPriorPicsFlag는 no_output_of_prior_pics_flag의 값에 관계없이, 테스트 중인 디코더에 의해 1로 설정될 수도 있다(그러나, 설정되지 않아야 함). NoOutputOfPriorPicsFlag를 no_output_of_prior_pics_flag와 동일하게 설정하는 것이 이 조건들 하에서 선호되지만, 테스트 중인 디코더는 이 경우에 NoOutputOfPriorPicsFlag를 1로 설정하도록 허용된다.

[0481] (iii) 이와 다를 경우, NoOutputOfPriorPicsFlag는 no_output_of_prior_pics_flag와 동일하게 설정된다.

[0482] (B) 테스트 중인 디코더에 대해 유도된 NoOutputOfPriorPicsFlag의 값은 다음과 같이 HRD에 대해 적용된다:

[0483] (i) NoOutputOfPriorPicsFlag가 1과 동일할 경우, DPB에서의 모든 픽처 저장 버퍼들은 이들이 포함하는 픽처들

의 출력 없이 비워지고, DPB 충만도(fullness)는 0과 동일하게 설정된다.

- [0484] (ii) 이와 다를 경우(NoOutputOfPriorPicsFlag가 0과 동일함), "출력을 위해 필요하지 않음" 및 "참조를 위해 이용되지 않음"으로서 표기되는 픽처를 포함하는 모든 픽처 저장 버퍼들은 (출력 없이) 비워지고, DPB에서의 모든 비워지지 않은 픽처 저장 버퍼들은 하위조항 F.13.5.2.4에서 특정된 "범핑" 프로세스를 반복적으로 호출함으로써 비워지고, DPB 충만도는 0과 동일하게 설정된다.
- [0485] (iii) 이와 다를 경우(현재의 픽처가 1과 동일한 NoRaslOutputFlag 및 0과 동일한 nuh_layer_id를 갖는 IRAP 픽처가 아님), "출력을 위해 필요하지 않음" 및 "참조를 위해 이용되지 않음"으로서 표기되는 픽처를 포함하는 모든 픽처 저장 버퍼들이 (출력 없이) 비워진다. 비워지는 각각의 픽처 저장 버퍼에 대하여, DPB 충만도는 1씩 감소된다. 변수 currLayerId는 현재의 디코딩된 픽처의 nuh_layer_id와 동일하게 설정된다.
- [0486] 변수들 MaxNumReorderPics[TargetOp][currLayerId][HighestTid], MaxLatencyIncreasePlus1[TargetOp][currLayerId][HighestTid], MaxLatencyPictures[TargetOp][currLayerId][HighestTid], MaxDecPicBufferingMinus1[TargetOp][currLayerId][HighestTid]은 테스트 중인 현재의 동작 포인트에 기반으로 하여 다음과 같이 유도된다:
- [0487] (1) 동작 포인트 DPB 정보 파라미터들 op_dpb_info_parameters()이 테스트 중인 동작 포인트 TargetOp에 대해 존재할 경우, MaxNumReorderPics[TargetOp][currLayerId][HighestTid]는 currLayerId가 0과 동일할 때에 vps_max_num_reorder_pics[HighestTid]로 설정되거나, currLayerId가 0보다 더 클 때에 테스트 중인 동작 포인트에 대한 currLayerId에 대해 vps_max_num_reorder_pics[TargetOp][CurrLayerId][HighestTid]로 설정된다. 이와 다르게, 동작 포인트 DPB 정보 파라미터들 op_dpb_info_parameters()이 테스트 중인 동작 포인트에 대해 존재하지 않을 경우, MaxNumReorderPics[TargetOp][currLayerId][HighestTid]는 활성 SPS로부터 (currLayerId가 0과 동일할 때), 또는 currLayerId의 값에 대한 활성 계층 SPS로부터의 sps_max_num_reorder_pics[HighestTid]로 설정된다.
- [0488] (2) 동작 포인트 DPB 정보 파라미터들 op_dpb_info_parameters()이 테스트 중인 동작 포인트 TargetOp에 대해 존재할 경우, MaxLatencyIncreasePlus1[TargetOp][currLayerId][HighestTid]는 currLayerId가 0과 동일할 때에 vps_max_latency_increase_plus1[HighestTid]로 설정되거나, currLayerId가 0보다 더 클 때에 테스트 중인 동작 포인트에 대한 currLayerId에 대해 vps_max_latency_increase_plus1[TargetOp][CurrLayerId][HighestTid]로 설정된다. 동작 포인트 DPB 정보 파라미터들 op_dpb_info_parameters()이 테스트 중인 동작 포인트에 대해 존재할 경우, MaxLatencyPictures[TargetOp][currLayerId][HighestTid]는 currLayerId가 0과 동일할 때에 VpsMaxLatencyPictures[HighestTid]로 설정되거나, currLayerId가 0보다 더 클 때에 테스트 중인 동작 포인트에 대한 currLayerId에 대해 VpsMaxLatencyPictures[TargetOp][CurrLayerId][HighestTid]로 설정된다. 이와 다르게, 동작 포인트 DPB 정보 파라미터들 op_dpb_info_parameters()이 테스트 중인 동작 포인트에 대해 존재하지 않을 경우, MaxLatencyIncreasePlus1[TargetOp][currLayerId][HighestTid]는 활성 SPS(currLayerId가 0과 동일할 때), 또는 currLayerId의 값에 대한 활성 계층 SPS의 sps_max_latency_increase_plus1[HighestTid]로 설정되고, MaxLatencyPictures[TargetOp][currLayerId][HighestTid]는 활성 SPS로부터(currLayerId가 0과 동일할 때), 또는 currLayerId의 값에 대한 활성 계층 SPS로부터 유도된 SpsMaxLatencyPictures[HighestTid]로 설정된다.
- [0489] (3) 동작 포인트 DPB 정보 파라미터들 op_dpb_info_parameters()이 테스트 중인 선택된 동작 포인트 TargetOp에 대해 존재할 경우, MaxDecPicBufferingMinus1[TargetOp][currLayerId][HighestTid]는 currLayerId가 0과 동일할 때에 vps_max_dec_pic_buffering_minus1[HighestTid]로 설정되거나, currLayerId가 0보다 더 클 때에 테스트 중인 동작 포인트에 대한 currLayerId에 대해 vps_max_dec_pic_buffering_minus1[TargetOp][CurrLayerId][HighestTid]로 설정된다. 이와 다르게, 동작 포인트 DPB 정보 파라미터들 op_dpb_info_parameters()이 테스트 중인 동작 포인트에 대해 존재하지 않을 경우, MaxDecPicBufferingMinus1[TargetOp][currLayerId][HighestTid]는 활성 SPS로부터(currLayerId가 0과 동일할 때), 또는 currLayerId의 값에 대한 활성 계층 SPS로부터의 sps_max_dec_pic_buffering_minus1[HighestTid]로 설정된다.
- [0490] 다음 조건들 중의 하나 이상이 참일 때에는, 다음 조건들의 어느 것도 참이 아닐 때까지, 비워지는 각각의 추가적인 픽처 저장 버퍼에 대하여 DPB 충만도를 1씩 추가로 감소시키면서, 하위조항 F.13.5.2.4에서 특정된 "범핑" 프로세스가 반복적으로 호출된다.
- [0491] (1) "출력을 위해 필요함"으로서 표기되는 DPB에서의 currLayerId와 동일한 nuh_layer_id를 갖는 픽처들의 수는

MaxNumReorderPics[TargetOp] [CurrLayerId][HighestTid]보다 더 크다.

- [0492] (2) MaxLatencyIncreasePlus1[TargetOp] [CurrLayerId][HighestTid]가 0과 동일하지 않고, 연관된 변수 PicLatencyCount[currLayerId]가 MaxLatencyPictures[TargetOp] [CurrLayerId][HighestTid] 이상인, "출력을 위해 필요함"으로서 표기되는 DPB에서의 currLayerId와 동일한 nuh_layer_id를 갖는 적어도 하나의 픽처가 있을 경우.
- [0493] (3) DPB에서의 currLayerId와 동일한 nuh_layer_id를 갖는 픽처들의 수는 MaxDecPicBuffering[TargetOp] [CurrLayerId][HighestTid] 이상이다.
- [0494] 이 하위조항에서 특정된 프로세스들은 현재의 픽처를 포함하는 액세스 유닛 n의 최후 디코딩 유닛이 CPB로부터 제거될 때 순간적으로 발생한다.
- [0495] 변수 currLayerId는 현재의 디코딩된 픽처의 nuh_layer_id와 동일하게 설정된다.
- [0496] "출력을 위해 필요함"으로서 표기되며 currLayerId와 동일한 nuh_layer_id 값을 가지는 DPB에서의 각각의 픽처에 대하여, 연관된 변수 PicLatencyCount[currLayerId]는 PicLatencyCount[currLayerId] + 1와 동일하게 설정된다.
- [0497] 현재의 픽처는 픽처의 최후 디코딩 유닛이 디코딩된 후에 디코딩된 것으로 간주된다. 현재의 디코딩된 픽처는 DPB에서의 비워진 픽처 저장 버퍼에서 저장되고, 다음이 적용된다:
- [0498] (A) 현재의 디코딩된 픽처가 1과 동일한 PicOutputFlag를 가질 경우, 그것은 "출력을 위해 필요함"으로서 표기되고, 그 연관된 변수 PicLatencyCount[currLayerId]는 0과 동일하게 설정된다.
- [0499] (B) 이와 다른 경우(현재의 디코딩된 픽처가 0과 동일한 PicOutputFlag를 가짐), 그것은 "출력을 위해 필요하지 않음"으로서 표기된다.
- [0500] 현재의 디코딩된 픽처는 "단기 참조를 위해 이용됨"으로서 표기된다.
- [0501] 다음의 조건들 중의 하나 이상이 참일 때에는, 다음의 조건들의 어느 것도 참이 아닐 때까지, 하위조항 F.13.5.2.4에서 특정된 "범핑" 프로세스가 반복적으로 호출된다:
- [0502] (A) "출력을 위해 필요함"으로서 표기되는 DPB에서의 currLayerId와 동일한 nuh_layer_id를 갖는 픽처들의 수는 MaxNumReorderPics[TargetOp] [CurrLayerId][HighestTid]보다 더 크다.
- [0503] (B) MaxLatencyIncreasePlus1[TargetOp] [CurrLayerId][HighestTid]가 0과 동일하지 않고, 연관된 변수 PicLatencyCount[currLayerId]가 MaxLatencyPictures[TargetOp] [CurrLayerId][HighestTid] 이상인, "출력을 위해 필요함"으로서 표기되는 DPB에서의 currLayerId와 동일한 nuh_layer_id를 갖는 적어도 하나의 픽처가 있을 경우.
- [0504] 다른 경우에는, 변수들 MaxNumReorderPics[currLayerId][HighestTid], MaxLatencyIncreasePlus1[currLayerId][HighestTid], MaxLatencyPictures[currLayerId][HighestTid], MaxDecPicBufferingMinus1[currLayerId][HighestTid]은 다음과 같이 유도될 수도 있다:
- [0505] (1) 동작 포인트 DPB 정보 파라미터들 op_dpb_info_parameters()이 테스트 중인 동작 포인트에 대해 존재할 경우, MaxNumReorderPics[currLayerId][HighestTid]는 currLayerId가 0과 동일할 때에 vps_max_num_reorder_pics[HighestTid]로 설정되거나, currLayerId가 0보다 더 클 때에 테스트 중인 동작 포인트에 대한 currLayerId에 대해 vps_max_num_reorder_pics[CurrLayerId][HighestTid]로 설정된다. 이와 다르게, 동작 포인트 DPB 정보 파라미터들 op_dpb_info_parameters()이 테스트 중인 동작 포인트에 대해 존재하지 않을 경우, MaxNumReorderPics[currLayerId][HighestTid]는 활성 SPS로부터(currLayerId가 0과 동일할 때), 또는 currLayerId의 값에 대한 활성 계층 SPS로부터의 sps_max_num_reorder_pics[HighestTid]로 설정된다.
- [0506] (2) 동작 포인트 DPB 정보 파라미터들 op_dpb_info_parameters()이 테스트 중인 동작 포인트에 대해 존재할 경우, MaxLatencyIncreasePlus1[currLayerId][HighestTid]는 currLayerId가 0과 동일할 때에 vps_max_latency_increase_plus1[HighestTid]로 설정되거나, currLayerId가 0보다 더 클 때에 테스트 중인 동작 포인트에 대한 currLayerId에 대해 vps_max_latency_increase_plus1[CurrLayerId][HighestTid]로 설정된다. 동작 포인트 DPB 정보 파라미터들 op_dpb_info_parameters()이 테스트 중인 동작 포인트에 대해 존재

할 경우, MaxLatencyPictures[currLayerId][HighestTid]는 currLayerId가 0과 동일할 때에 VpsMaxLatencyPictures[HighestTid]로 설정되거나, currLayerId가 0보다 더 클 때에 테스트 중인 동작 포인트에 대한 currLayerId에 대해 VpsMaxLatencyPictures[CurrLayerId][HighestTid]로 설정된다. 이와 다르게, 동작 포인트 DPB 정보 파라미터들 op_dpb_info_parameters()이 테스트 중인 동작 포인트에 대해 존재하지 않을 경우, MaxLatencyIncreasePlus1[currLayerId][HighestTid]는 활성 SPS(currLayerId가 0과 동일할 때), 또는 currLayerId의 값에 대한 활성 계층 SPS의 sps_max_latency_increase_plus1[HighestTid]로 설정되고, MaxLatencyPictures[currLayerId][HighestTid]는 활성 SPS로부터(currLayerId가 0과 동일할 때), 또는 currLayerId의 값에 대한 활성 계층 SPS로부터 유도된 SpsMaxLatencyPictures[HighestTid]로 설정된다.

- [0507] (3) 동작 포인트 DPB 정보 파라미터들 op_dpb_info_parameters()이 테스트 중인 선택된 동작 포인트에 대해 존재할 경우, MaxDecPicBufferingMinus1[currLayerId][HighestTid]는 currLayerId가 0과 동일할 때에 vps_max_dec_pic_buffering_minus1[HighestTid]로 설정되거나, currLayerId가 0보다 더 클 때에 테스트 중인 동작 포인트에 대한 currLayerId에 대해 vps_max_dec_pic_buffering_minus1[CurrLayerId][HighestTid]로 설정된다. 이와 다르게, 동작 포인트 DPB 정보 파라미터들 op_dpb_info_parameters()이 테스트 중인 동작 포인트에 대해 존재하지 않을 경우, MaxDecPicBufferingMinus1[currLayerId][HighestTid]는 활성 SPS로부터 (currLayerId가 0과 동일할 때), 또는 currLayerId의 값에 대한 활성 계층 SPS로부터의 sps_max_dec_pic_buffering_minus1[HighestTid]로 설정된다.
- [0508] 다음 조건들 중의 하나 이상이 참일 때에는, 다음 조건들의 어느 것도 참이 아닐 때까지, 비워지는 각각의 추가적인 픽처 저장 버퍼에 대하여 DPB 층만도를 1씩 추가로 감소시키면서, 하위조항 F.13.5.2.4에서 특정된 "범핑" 프로세스가 반복적으로 호출된다.
- [0509] (1) "출력을 위해 필요함"으로서 표기되는 DPB에서의 currLayerId와 동일한 nuh_layer_id를 갖는 픽처들의 수는 MaxNumReorderPics[CurrLayerId][HighestTid]보다 더 크다.
- [0510] (2) MaxLatencyIncreasePlus1[CurrLayerId][HighestTid]가 0과 동일하지 않고, 연관된 변수 PicLatencyCount[currLayerId]가 MaxLatencyPictures[CurrLayerId][HighestTid] 이상인, "출력을 위해 필요함"으로서 표기되는 DPB에서의 currLayerId와 동일한 nuh_layer_id를 갖는 적어도 하나의 픽처가 있을 경우.
- [0511] (3) DPB에서의 currLayerId와 동일한 nuh_layer_id를 갖는 픽처들의 수는 MaxDecPicBuffering[CurrLayerId][HighestTid] 이상이다.
- [0512] 이 하위조항에서 특정된 프로세스들은 현재의 픽처를 포함하는 액세스 유닛 n의 최후 디코딩 유닛이 CPB로부터 제거될 때에 순간적으로 발생한다.
- [0513] 변수 currLayerId는 현재의 디코딩된 픽처의 nuh_layer_id와 동일하게 설정된다.
- [0514] "출력을 위해 필요함"으로서 표기되며 currLayerId와 동일한 nuh_layer_id 값을 가지는 DPB에서의 각각의 픽처에 대하여, 연관된 변수 PicLatencyCount[currLayerId]는 PicLatencyCount[currLayerId] + 1와 동일하게 설정된다.
- [0515] 현재의 픽처는 픽처의 최후 디코딩 유닛이 디코딩된 후에 디코딩된 것으로 간주된다. 현재의 디코딩된 픽처는 DPB에서의 비워진 픽처 저장 버퍼에서 저장되고, 다음이 적용된다:
- [0516] (A) 현재의 디코딩된 픽처가 1과 동일한 PicOutputFlag를 가질 경우, 그것은 "출력을 위해 필요함"으로서 표기되고, 그 연관된 변수 PicLatencyCount[currLayerId]는 0과 동일하게 설정된다.
- [0517] (B) 이와 다를 경우(현재의 디코딩된 픽처가 0과 동일한 PicOutputFlag를 가짐), 그것은 "출력을 위해 필요하지 않음"으로서 표기된다.
- [0518] 현재의 디코딩된 픽처는 "단기 참조를 위해 이용됨"으로서 표기된다.
- [0519] 다음의 조건들 중의 하나 이상이 참일 때에는, 다음의 조건들의 어느 것도 참이 아닐 때까지, 하위조항 F.13.5.2.4에서 특정된 "범핑" 프로세스가 반복적으로 호출된다:
- [0520] (A) "출력을 위해 필요함"으로서 표기되는 DPB에서의 currLayerId와 동일한 nuh_layer_id를 갖는 픽처들의 수는 MaxNumReorderPics[CurrLayerId][HighestTid]보다 더 크다.
- [0521] (B) MaxLatencyIncreasePlus1[CurrLayerId][HighestTid]가 0과 동일하지 않고, 연관된 변수 PicLatencyCount[

currLayerId]가 MaxLatencyPictures[CurrLayerId][HighestTid] 이상인, "출력을 위해 필요함"으로서 표기되는 DPB에서의 currLayerId와 동일한 nuh_layer_id를 갖는 적어도 하나의 픽처가 있을 경우.

- [0522] "범핑" 프로세스는 다음의 순서화된 단계들로 구성된다:
- [0523] (A) 출력을 위해 최초인 픽처들은 "출력을 위해 필요함"으로서 표기된 DPB에서의 모든 픽처들의 PicOrderCntVal의 최소 값을 가지는 것들로서 선택된다.
- [0524] (B) 이 픽처들은 0과 동일한 nuh_layer_id를 갖는 픽처에 대한 활성 SPS에서, 또는 픽처의 그것과 동일한 nuh_layer_id 값에 대한 활성 계층 SPS에서 특정된 적합성 크롭핑 윈도우를 이용하여 크롭핑되고, 크롭핑된 픽처들은 nuh_layer_id의 오름차순으로 출력되고, 픽처들은 "출력을 위해 필요하지 않음"으로서 표기된다.
- [0525] (C) "참조를 위해 이용되지 않음"으로서 표기된 픽처를 포함하며 크롭핑 및 출력되었던 픽처들 중의 하나를 포함하였던 각각의 픽처 저장 버퍼는 비워진다.
- [0526] VPS 확장은 희망할 경우에 추가적인 수정들을 가질 수도 있다.
- [0527] 도 40을 참조하면, 추가적인 수정은 동작 포인트들 대신으로, 출력 계층 세트들에 대한 VPS 확장에서 전송되는 DPB 파라미터들을 포함할 수도 있고, 여기서, oop_dpb_info_parameters(j)는 도 41에서 예시되어 있다.
- [0528] num_dpb_info_parameters는 VPS 확장 Rbsp에서 존재하는 oop_dpb_parameters() 선택스 구조들의 수를 특정한다. num_dpb_info_parameters 디코더들은 0 내지 num_output_layer_sets까지의 범위에 있을 것이다.
- [0529] output_point_layer_set_idx[i]는 VPS 확장에서의 i번째 oop_dpb_info_parameters() 선택스 구조가 적용되는 타겟 출력 계층 세트들의 리스트로의 인덱스를 특정한다.
- [0530] output_point_layer_set_idx[i]의 값은 0 내지 num_output_layer_sets까지의 범위에 있어야 한다. output_point_layer_set_idx [i]는 i와 동일하지 않은 임의의 j에 대한 output_point_layer_set_idx [j]와 동일하지 않을 것이라는 것은 비트스트림 적합성의 요건이다.
- [0531] 도 42를 참조하면, oop_dpb_info_parameters(c)가 추가로 수정될 수도 있고, 여기서, VPS 확장에서의 선택스는 도 43에서 예시된 바와 같을 수도 있다.
- [0532] 도 44를 참조하면, oop_dpb_info_parameters(c)가 추가로 수정될 수도 있고, 여기서, VPS 확장에서의 선택스는 도 45 또는 도 46에서 예시된 바와 같을 수도 있다.
- [0533] VPS 확장에서의 선택스에 대한 예시적인 대안은 다음과 같고,

표 18

for(j = 0; j <= vps_max_layer_id; j++)
oop_dpb_info_parameters(j)

- [0534]
- [0535] 이것은 하기로 변경될 수도 있다.

표 19

for(j = 0; j <= vps_max_layers_minus1; j++)
oop_dpb_info_parameters(j)

- [0536]
- [0537] vps_max_layer_id는 CVS에서의 모든 NAL 유닛들의 nuh_layer_id의 최대 허용된 값을 특정한다. vps_max_layers_minus1은 CVS에서 존재할 수도 있는 계층들의 최대 수를 특정하고, 여기서, 계층은 예컨대, 공간적 스케일러블 계층, 품질 스케일러블 계층, 텍스처 뷰 또는 심도 뷰일 수도 있다.
- [0538] VPS 확장에서의 선택스에 대한 또 다른 예시적인 대안은 다음과 같고,

표 20

[0539]

for(j = 0; j <= vps_max_layer_id; j++)
oop_dpb_info_parameters(j)

[0540] 이것은 하기로 변경될 수도 있다.

표 21

[0541]

for(j = 0; j < numOutputLayers; j++)
oop_dpb_info_parameters(j)

[0542] 여기서, 선택된 출력 계층 세트 인덱스 opIsIdx에 대한 numOutputLayers는 하기로서 유도된다:

표 22

[0543]

for(k=0, numOutputLayers=0;k<=vps_max_layer_id;k++)
if(output_layer_flag[opLsIdx][k])
targetOpLayerIdList [numOutputLayers++]=layer_id_in_nuh[k]

[0544] VPS 확장에서의 선택에 대한 또 다른 예시적인 대안은 다음과 같고,

표 23

[0545]

for(j = 0; j <= vps_max_layer_id; j++)
oop_dpb_info_parameters(j)

[0546] 이것은 하기로 변경될 수도 있다.

표 24

[0547]

for(j = 0; j < numDecodedLayers; j++)
oop_dpb_info_parameters(j)

[0548] 여기서, 선택된 opIsIdx에 대한 numOutputLayers는 하기로서 유도된다:

표 25

[0549]

for(k=0, numOutputLayers=0;k<=vps_max_layer_id;k++)
if(output_layer_flag[opLsIdx][k])
targetOpLayerIdList [numOutputLayers++]=layer_id_in_nuh[k]

[0550] 다음으로, 선택된 opIsIdx에 대한 타겟 디코딩된 세트 식별자 리스트 targetDLayerIdList 및 numDecodedLayers는 하기로써 유도된다:

표 26

for(m=0, numDecodedLayers=0;m< numOutputLayers;m++) {
for(n=0;n<NumDirectRefLayers[LayerIdInVps[targetOpLayerIdList[m]]];n++) {
rLid=RefLayerId[LayerIdInVps[targetOpLayerIdList[m]]][n]
if(rLid not included in targetDLayerIdList[0,..., numDecodedLayers])
targetDLayerIdList[numDecodedLayers++]=rLid;
}
}

[0551]

[0552] 하나의 실시형태에서, 추가적인 플래그는 oop_dpb_information_parameters가 다음과 같이 특별한 계층에 대해 시그널링되는지를 표시하기 위하여 시그널링될 수도 있다:

표 27

for(j = 0; j <= vps_max_layer_id; j++) {	
vps_layer_info_present_flag[j]	u(1)
if(vps_layer_info_present_flag)	
oop_dpb_info_parameters(j)	
}	

[0553]

[0554] 1과 동일한 vps_layer_info_present_flag[j]는, oop_dpb_info_parameters가 특별한 출력 계층 세트에 대한 j번째 계층에 대해 존재하는 것으로 특정한다. 0과 동일한 vps_layer_info_present_flag[j]는, oop_dpb_info_parameters가 특별한 출력 계층 세트에 대한 j번째 계층에 대해 존재하지 않는 것으로 특정한다.

[0555] 또 다른 실시형태에서, num_dpb_info_parameters 디코더들은 0 내지 1024까지의 범위에 있을 것이다. 또 다른 실시형태에서, 상이한 고정된 수가 1024 대신에 이용될 수 있다.

[0556] 대안적인 실시형태에서, output_point_layer_set_idx[i]는 0 내지 1023까지의 범위에 있다.

[0557] 도 47을 참조하면, 또 다른 수정된 VPS 확장 및 layer_dpb_info(i)는, DPB 파라미터들이 출력 계층 세트들 및 동작 포인트들에 대해 독립적으로 각각의 계층에 대한 VPS 확장에서 전송될 경우에 이용될 수도 있다.

[0558] 도 48을 참조하면, 수정된 layer_dpb_info(i)가 이용될 수도 있고, 여기서, VPS로부터 시그널링된 선택스 엘리먼트 vps_max_sub_layer_minus1은 모든 계층들에 대해 이용되고, oop_dpb_info_parameters(id) / oop_dpb_info_parameters(id)에서 별도로 시그널링되지 않는다.

[0559] 도 49a 및 도 49b를 참조하면, 예시적인 수정된 vps_extension이 예시되어 있다. 수정된 vps 확장은 새로운 선택스, 즉, max_sub_layers_vps_predict_flag[i], max_sub_layers_vps_minus1[i] num_dpb_info_parameters, output_point_layer_set_idx[i], oop_dpb_maxbuffering_parameters(i), 및 layer_dpb_info_parameters(i)를 포함한다. num_output_layer_sets는, 출력 계층들이 output_layer_set_index[i] 및 output_layer_flag[lsIdx][j]로 특정되는 계층 세트들의 수를 특정한다. 존재하지 않을 때, num_output_layer_sets의 값은 0

과 동일한 것으로 추론된다. 출력 계층들을 설명하는 계층 세트는 출력 계층 세트이다.

- [0560] 1과 동일한 'max_sub_layers_vps_predict_flag'[i]는, max_sub_layers_vps_minus1[i]이 max_sub_layers_vps_minus1[i - 1]과 동일한 것으로 추론되는 것으로 특정한다.
- [0561] 0과 동일한 max_sub_layers_vps_predict_flag[i]는, max_sub_layers_vps_minus1[i]이 명시적으로 시그널링 되는 것으로 특정한다. max_sub_layers_vps_predict_flag[0]의 값은 0과 동일한 것으로 추론된다.
- [0562] 'max_sub_layers_vps_minus1'[i] 플러스 1은, i와 동일한 nuh_layer_id를 갖는 계층에 대한 CVS에서 존재할 수도 있는 시간적 서브-계층들의 최대 수를 특정한다. max_vps_sub_layers_vps_minus1[i]의 값은 0 내지 6 까지의 범위에 있을 것이다. 일부의 경우에는, max_sub_layers_vps_minus1[i]이 SPS 신택스 엘리먼트 sps_max_sub_layers_minus1의 추론을 위해 이용된다. max_sub_layers_vps_predict_flag[i]가 1과 동일할 때, max_sub_layers_vps_minus1[i]은 max_sub_layers_vps_minus1[i - 1]과 동일한 것으로 추론된다. max_sub_layers_vps_minus1[0]의 값은 vps_max_sub_layers_minus1과 동일한 것으로 추론된다.
- [0563] 0 내지 num_dpb_info_parameters- 1까지의 범위에서의 setId에 대한 변수 MaxSubLayers[setId]는 다음과 같이 유도된다:

표 28

for(setId = 0; setId < num_dpb_info_parameters; setId++) {
lsIdx =
output_layer_set_idx_minus1[output_point_layer_set_idx[setId]] + 1
highestLayerId =
layerSetLayerIdList[lsIdx][numLayersInIdList[lsIdx]-1]
MaxSubLayers[setId] =
(max_sub_layers_vps_minus1[highestLayerId]+1)
}

- [0564]
- [0565] 'num_dpb_info_parameters'는 VPS 확장 RBSP에서 존재하는 oop_dpb_maxbuffering_parameters(i) 신택스 구조들의 수를 특정한다. num_dpb_info_parameters 디코더들은 0 내지 numOutputLayerSets까지의 범위에 있을 것이다.
- [0566] 'output_point_layer_set_idx'[i]는 VPS 확장에서의 i번째 oop_dpb_maxbuffering_parameters(i) 신택스 구조가 적용되는 출력 계층 세트들의 리스트로의 인덱스를 특정한다.
- [0567] output_point_layer_set_idx[i]의 값은 0 내지 numOutputLayerSets까지의 범위에 있어야 한다. output_point_layer_set_idx [i]는 i와 동일하지 않은 임의의 j에 대한 output_point_layer_set_idx [j]와 동일하지 않을 것이라는 것은 비트스트림 적합성의 요건이다.
- [0568] 도 50을 참조하면, oop_dpb_maxbuffering_parameters는 'sub_layer_vps_buf_info_present_flag'[i], 'max_vps_dec_pic_buffering_minus1'[i][j]을 특정한다.
- [0569] 1과 동일한 'sub_layer_vps_buf_info_present_flag'[i]는, max_vps_dec_pic_buffering_minus1[i][j]이 MaxSubLayers[i] 서브-계층들에 대해 존재하는 것으로 특정한다. 0과 동일한 sub_layer_vps_buf_info_present_flag[i]는, max_vps_dec_pic_buffering_minus1[i][MaxSubLayers[i] - 1]의 값들이 모든 서브-계층들에 적용되는 것으로 특정한다.
- [0570] 'max_vps_dec_pic_buffering_minus1'[i][j] 플러스 1은, HighestTid가 j와 동일할 때, 픽처 저장 버퍼들의 단위로 인덱스 i와 연관된 출력 계층 세트에서의 highestLayerId와 동일한 nuh_layer_id를 갖는 계층에 대한 CVS에 대한 디코딩된 픽처 버퍼의 최대 요구된 사이즈를 특정한다. max_vps_dec_pic_buffering_minus1[i][j]의 값은 (하위조항 A.4에서 특정된 바와 같이) 0 내지 MaxDpbSize - 1까지의 범위에 있을 것이다. j가 0보다 더 클 때, max_vps_dec_pic_buffering_minus1[i][j]는 max_vps_dec_pic_buffering_minus1[i][j - 1

] 이상일 것이다. 일부의 경우에는, $\text{max_vps_dec_pic_buffering_minus1}[i][j]$ 가 SPS 신택스 엘리먼트들 $\text{sps_max_dec_pic_buffering_minus1}[j]$ 의 값들의 추론을 위해 이용된다. $\text{max_vps_dec_pic_buffering_minus1}[i][j]$ 이 0 내지 $\text{MaxSubLayers}[i] - 2$ 까지의 범위에서 i 에 대해 존재하지 않을 때, $\text{sub_layer_vps_buf_info_present_flag}[i]$ 가 0과 동일한 것으로 인해, 그것은 $\text{max_vps_dec_pic_buffering_minus1}[i][\text{MaxSubLayers}[i] - 1]$ 과 동일한 것으로 추론된다.

[0571] j 의 각각의 값에 대한 $\text{max_vps_dec_pic_buffering_minus1}[0][j]$ 의 값은 $\text{vps_max_dec_pic_buffering_minus1}[j]$ 와 동일한 것으로 추론된다.

[0572] 도 51을 참조하면, $\text{layer_dpb_info_parameters}$ 는 'sub_layer_vps_ordering_info_present_flag'[i], $\text{max_vps_num_reorder_pics}[i][j]$, 'max_vps_latency_increase_plus1'[$i][j]$ 을 특정한다.

[0573] 1과 동일한 'sub_layer_vps_ordering_info_present_flag'[i]는, $\text{max_vps_num_reorder_pics}[i][j]$ 및 $\text{max_vps_latency_increase_plus1}[i][j]$ 이 $\text{max_sub_layers_vps_minus1} + 1$ 서브-계층들에 대해 존재하는 것으로 특정한다.

[0574] 0과 동일한 $\text{sub_layer_vps_ordering_info_present_flag}[i]$ 는, $\text{max_vps_num_reorder_pics}[i][\text{vps_max_sub_layers_minus1}]$ 및 $\text{max_vps_latency_increase_plus1}[i][\text{max_sub_layers_vps_minus1}]$ 의 값들이 모든 서브-계층들에 적용되는 것으로 특정한다.

[0575] 'max_vps_num_reorder_pics'[$i][j]$ 는, HighestTid가 j 와 동일할 때, 디코딩 순서에서 i 와 동일한 nuh_layer_id 를 갖는 계층에 대한 CVS에서의 임의의 픽처를 선행할 수 있으며 출력 순서에서 그 픽처를 후행할 수 있는 픽처들의 최대 허용된 수를 표시한다. $\text{max_vps_num_reorder_pics}[i][j]$ 의 값은 0 내지 $\text{max_vps_dec_pic_buffering_minus1}[i][j]$ 까지의 범위에 있을 것이다. j 가 0보다 더 클 때, $\text{max_vps_num_reorder_pics}[i][j]$ 는 $\text{max_vps_num_reorder_pics}[i][j-1]$ 이상일 것이다. 일부의 경우에는, $\text{max_vps_num_reorder_pics}[i][j]$ 가 SPS 신택스 엘리먼트들 $\text{sps_max_num_reorder_pics}[j]$ 의 값들의 추론을 위해 이용된다. $\text{max_vps_num_reorder_pics}[i][j]$ 이 0 내지 $\text{max_sub_layers_vps_minus1}[i] - 1$ 까지의 범위에서 i 에 대해 존재하지 않을 때, $\text{sub_layer_vps_ordering_info_present_flag}[i]$ 가 0과 동일한 것으로 인해, 그것은 $\text{max_vps_num_reorder_pics}[i][\text{max_sub_layers_vps_minus1}[i]]$ 과 동일한 것으로 추론된다.

[0576] 0과 동일하지 않은 $\text{max_vps_latency_increase_plus1}[i][j]$ 은, HighestTid가 j 와 동일할 때, 출력 순서에서 i 와 동일한 nuh_layer_id 를 갖는 계층에 대한 CVS에서의 임의의 픽처를 선행할 수 있으며 디코딩 순서에서 그 픽처를 후행할 수 있는 픽처들의 최대 수를 특정하는 $\text{VpsMaxLatencyPictures}[i][j]$ 의 값을 연산하기 위하여 이용된다.

[0577] $\text{max_vps_latency_increase_plus1}[i][j]$ 이 0과 동일하지 않을 때, $\text{VpsMaxLatencyPictures}[i][j]$ 의 값은 다음과 같이 특정된다:

[0578] $\text{VpsMaxLatencyPictures}[i][j] = \text{max_vps_num_reorder_pics}[i][j] + \text{max_vps_latency_increase_plus1}[i][j] - 1$

[0579] $\text{max_vps_latency_increase_plus1}[i][j]$ 이 0과 동일할 때, 대응하는 한계가 표현되지 않는다.

[0580] $\text{max_vps_latency_increase_plus1}[j][k]$ 의 값은 0 내지 $2^{32} - 2$ 까지의 범위에 있을 것이다. 일부의 경우에는, $\text{max_vps_latency_increase_plus1}[i][j]$ 이 SPS 신택스 엘리먼트들 $\text{sps_max_latency_increase_plus1}[j]$ 의 값들의 추론을 위해 이용된다. $\text{max_vps_latency_increase_plus1}[i][j]$ 가 0 내지 $\text{max_sub_layers_vps_minus1}[i] - 1$ 까지의 범위에서 i 에 대해 존재하지 않을 때, $\text{sub_layer_vps_ordering_info_present_flag}[i]$ 가 0과 동일한 것으로 인해, 그것은 $\text{max_vps_latency_increase_plus1}[i][\text{max_sub_layers_vps_minus1}[i]]$ 과 동일한 것으로 추론된다.

[0581] 도 52를 참조하면, 예시적인 수정된 vps_extension 이 예시되어 있다. 수정된 vps 확장은 새로운 신택스, 즉, 조건부로 시그널링되는 $\text{sub_layer_vps_buf_info_predict_flag}[i]$, $\text{sub_layer_vps_ordering_info_predict_flag}[i]$ 에 의한 도 49에서의 신택스의 추가의 수정을 포함한다. 1과 동일한 $\text{sub_layer_vps_buf_info_predict_flag}[i]$ 는, $\text{max_vps_dec_pic_buffering_minus1}[i][j]$ 이 j 의 각각의 값에 대한 $\text{max_vps_dec_pic_buffering_minus1}[i-1][j]$ 과 동일한 것으로 추론되는 것으로 특정한다.

- [0582] 0과 동일한 $sub_layer_vps_buf_info_predict_flag[i]$ 는, j 의 적어도 하나의 값에 대한 $max_vps_dec_pic_buffering_minus1[i][j]$ 이 명시적으로 시그널링되는 것으로 특정한다. $sub_layer_vps_buf_info_predict_flag[0]$ 는 0과 동일한 것으로 추론된다. 존재하지 않을 때, $sub_layer_vps_buf_info_predict_flag[i]$ 는 0과 동일한 것으로 추론된다.
- [0583] 1과 동일한 $sub_layer_vps_ordering_info_predict_flag[i]$ 는, 선택스 엘리먼트들 $sub_layer_vps_ordering_info_present_flag[i]$, $max_vps_num_reorder_pics[i][j]$, 및 $max_vps_latency_increase_plus1[i][j]$ 이 각각 $sub_layer_vps_ordering_info_present_flag[i-1]$, $max_vps_num_reorder_pics[i-1][j]$, 및 $max_vps_latency_increase_plus1[i-1][j]$ 과 동일한 것으로 추론되는 것으로 특정한다. 0과 동일한 $sub_layer_vps_ordering_info_predict_flag[i]$ 는, 선택스 엘리먼트들 $sub_layer_vps_ordering_info_present_flag[i]$, $max_vps_num_reorder_pics[i][j]$, 및 $max_vps_latency_increase_plus1[i][j]$ 이 명시적으로 시그널링됨을 표시한다. 존재하지 않을 때, $sub_layer_vps_ordering_info_predict_flag[i]$ 의 값은 0과 동일하게 설정된다.
- [0584] 도 52에 대한 다른 선택스 엘리먼트들 및 그 시맨틱 의미들은 도 49에 대한 것들과 동일하다.
- [0585] 도 53을 참조하면, 예시적인 수정된 $vps_extension$ 이 예시되어 있다. 수정된 vps 확장은 도 49에서의 선택스의 추가의 수정을 포함한다. 도 53에서, $oop_dpb_maxbuffering_parameters(i,j)$ 는, 출력 계층 세트 i 에서의 최고 계층 id 에 대한 $oop_dpb_maxbuffering_parameters(i)$ 파라미터들의 단일 세트를 시그널링하는 도 49와 비교하여, 특별한 출력 계층 세트 i 에 대한 각각의 계층 j 에 대해 시그널링된다.
- [0586] 도 54를 참조하면, $oop_dpb_maxbuffering_parameters$ 는 $sub_layer_vps_buf_info_present_flag[i][k]$, $max_vps_dec_pic_buffering_minus1[i][k][j]$ 을 특정한다.
- [0587] 1과 동일한 $sub_layer_vps_buf_info_present_flag[i][k]$ 는, $max_vps_dec_pic_buffering_minus1[i][k][j]$ 이 $max_sub_layers_vps_minus1[k]$ 서브-계층들에 대해 존재하는 것으로 특정한다. 0과 동일한 $sub_layer_vps_buf_info_present_flag[i]$ 는, $max_vps_dec_pic_buffering_minus1[i][k][max_sub_layers_vps_minus1[k]]$ 의 값들이 모든 서브-계층들에 적용되는 것으로 특정한다.
- [0588] $max_vps_dec_pic_buffering_minus1[i][k][j]$ 플러스 1은, HighestTid가 j 와 동일할 때, 픽처 저장 버퍼들의 단위로 인덱스 i 와 연관된 출력 계층 세트에 대한 k 와 동일한 nuh_layer_id 를 갖는 계층에 대한 CVS에 대한 디코딩된 픽처 버퍼의 최대 요구된 사이즈를 특정한다. $max_vps_dec_pic_buffering_minus1[i][k][j]$ 의 값은 (하위조항 A.4에서 특정된 바와 같이) 0 내지 $MaxDpbSize - 1$ 까지의 범위에 있을 것이다.
- [0589] j 가 0보다 더 클 때, $max_vps_dec_pic_buffering_minus1[i][k][j-1]$ 이상일 $max_vps_dec_pic_buffering_minus1[i][k][j]$ 은 SPS 선택스 엘리먼트들 $sps_max_dec_pic_buffering_minus1[j]$ 의 값들의 추론을 위해 이용된다.
- [0590] $max_vps_dec_pic_buffering_minus1[i][j]$ 이 0 내지 $max_sub_layers_vps_minus1[k] - 1$ 까지의 범위에서 i 에 대해 존재하지 않을 때, $sub_layer_vps_buf_info_present_flag[i][k]$ 가 0과 동일한 것으로 인해, 그것은 $max_vps_dec_pic_buffering_minus1[i][k][max_sub_layers_vps_minus1[k]]$ 과 동일한 것으로 추론된다.
- [0591] i 및 j 의 각각의 값에 대한 $max_vps_dec_pic_buffering_minus1[i][0][j]$ 의 값은 $vps_max_dec_pic_buffering_minus1[j]$ 와 동일한 것으로 추론된다.
- [0592] 도 55를 참조하면, $oop_dpb_maxbuffering_parameters$ 는 $sub_layer_vps_buf_info_present_flag[i][k]$, $max_vps_dec_pic_buffering_minus1[i][k][j]$ 을 특정한다. 도 55는 $oop_dpb_maxbuffering_parameters$ 에 대한 도 54에서의 선택스와 비교하여, $oop_dpb_maxbuffering_parameters$ 에 대한 변형 선택스이다.
- [0593] 0 내지 $num_dpb_info_parameters - 1$ 까지의 범위에서의 $setId$ 에 대한 변수 $MaxSubLayers[setId][k]$ 는 다음과 같이 유도된다:

표 29

for(setId = 0; setId < num_dpb_info_parameters; setId++) {
for(k = 1; k <= vps_max_layers_minus1; i++) {
MaxSubLayers[setId][k] =(max_sub_layers_vps_minus1[k]+1)
}
}

[0594]

[0595] 이 경우, oop_dpb_maxbuffering_parameters 파라미터들(i, k)은 도 55에서와 같이 정의될 것이다.

[0596] 도 56을 참조하면, 예시적인 수정된 vps_extension이 예시되어 있다. 수정된 vps 확장은 새로운 선택스에 의한 도 52에서의 선택스의 추가의 수정을 포함한다. 도 53에서, 이 변형 oop_dpb_maxbuffering_parameters(i,j)는, 출력 계층 세트 i에서의 최고 계층 id에 대한 oop_dpb_maxbuffering_parameters(i) 파라미터들의 단일 세트를 시그널링하는 도 52와 비교하여, 특별한 출력 계층 세트 i에 대한 각각의 계층 j에 대해 시그널링된다.

[0597] oop_dpb_maxbuffering_parameters(I,k)는 도 57에서 도시된 바와 같다.

[0598] 도 58을 참조하면, oop_dpb_maxbuffering_parameters는 sub_layer_vps_buf_info_present_flag[i][k], max_vps_dec_pic_buffering_minus1[i][k][j]을 특정한다. 도 58은 oop_dpb_maxbuffering_parameters에 대한 도 57에서의 선택스와 비교하여, oop_dpb_maxbuffering_parameters에 대한 변형 선택스이다.

[0599] 0 내지 num_dpb_info_parameters- 1까지의 범위에서의 setId에 대한 변수 MaxSubLayers[setId][k]는 다음과 같이 유도된다:

표 30

for(setId = 0; setId < num_dpb_info_parameters; setId++) {
for(k = 1; k <= vps_max_layers_minus1; i++) {
MaxSubLayers[setId][k] =(max_sub_layers_vps_minus1[k]+1)
}
}

[0600]

[0601] 이 경우, oop_dpb_maxbuffering_parameters 파라미터들(i, k)은 도 58에서와 같이 정의될 것이다.

[0602] 도 59를 참조하면, 예시적인 수정된 vps_extension이 예시되어 있다. 수정된 vps 확장은 새로운 선택스, 즉, 조건부로 시그널링되는 추가적인 플래그 layer_dpb_info_parameters_presence_flag에 의한 도 56에서의 선택스의 추가의 수정을 포함한다. layer_dpb_info_parameters_presence_flag는 VPS 확장에서의 layer_dpb_info_parameters(i)의 시그널링을 선택적으로 만든다.

[0603] 1과 동일한 layer_dpb_info_parameters_presence_flag[i]는, 선택스 엘리먼트들 sub_layer_vps_ordering_info_predict_flag[i] 및 layer_dpb_info_parameters(i)이 vps_max_num_layers_minus1 계층들에 대해 존재하는 것으로 특정한다. 0과 동일한 layer_dpb_info_parameters_presence_flag[i]는, 선택스 엘리먼트들 sub_layer_vps_ordering_info_predict_flag[i] 및 layer_dpb_info_parameters(i)이 vps_max_num_layers_minus1 계층들에 대해 존재하지 않는 것으로 특정한다.

[0604] oop_dpb_maxbuffering_parameters(i,k)는 도 60에서 도시된 바와 같다.

[0605] 도 61을 참조하면, oop_dpb_maxbuffering_parameters는 sub_layer_vps_buf_info_present_flag[i][k],

max_vps_dec_pic_buffering_minus1[i][k][j]을 특정한다. 도 61은 oop_dpb_maxbuffering_parameters에 대한 도 60에서의 선택스와 비교하여, oop_dpb_maxbuffering_parameters에 대한 변형 선택스이다.

[0606] 0 내지 num_dpb_info_parameters- 1까지의 범위에서의 setId에 대한 변수 MaxSubLayers[setId][k]는 다음과 같이 유도된다:

표 31

for(setId = 0; setId < num_dpb_info_parameters; setId++) {
for(k = 1; k <= vps_max_layers_minus1; i++) {
MaxSubLayers[setId][k] =(max_sub_layers_vps_minus1[k]+1)
}
}

[0607]

[0608] 이 경우, oop_dpb_maxbuffering_parameters 파라미터들 (i, k)은 도 61에서와 같이 정의될 것이다.

[0609] VPS 확장에서의 선택스에 대한 예시적인 대안은 다음과 같고,

표 32

for(k = 1; j <= vps_max_layers_minus1; k++)
oop_dpb_maxbuffering_parameters(k)

[0610]

[0611] 이것은 하기로 변경될 수도 있다.

표 33

for(k = 0; k <= vps_max_layers_minus1; k++)
oop_dpb_maxbuffering_parameters(k)

[0612]

[0613] 이에 따라, 인덱스 k는 1 대신으로 0에서 시작할 수 있다.

[0614] VPS 확장에서의 선택스에 대한 예시적인 대안은 다음과 같고,

표 34

for(k = 0; k <= vps_max_layers_minus1; k++)
oop_dpb_maxbuffering_parameters(k)

[0615]

[0616] 이것은 하기로 변경될 수도 있다.

표 35

for(k = 0; k <= vps_max_layer_id; k++)
oop_dpb_maxbuffering_parameters(k)

[0617]

[0618] vps_max_layer_id는 CVS에서의 모든 NAL 유닛들의 nuh_layer_id의 최대 허용된 값을 특정한다. vps_max_layers_minus1은 CVS에서 존재할 수도 있는 계층들의 최대 수를 특정하고, 여기서, 계층은 예컨대, 공간적 스케일러블 계층, 품질 스케일러블 계층, 텍스처 뷰 또는 심도 뷰일 수도 있다.

[0619] VPS 확장에서의 선택스에 대한 또 다른 예시적인 대안은 다음과 같고,

표 36

```
[0620] for( k = 1; k <= vps_max_layers_minus1; k++ )
        oop_dpb_maxbuffering_parameters(I,k)
```

[0621] 이것은 하기로 변경될 수도 있다.

표 37

```
[0622] for( k = 1; k < numOutputLayers; k++ )
        oop_dpb_maxbuffering_parameters(I,k)
```

[0623] 여기서, numOutputLayers는 하기와 같이 유도되거나,

표 38

```
[0624] for( setId = 0; setId < num_dpb_info_parameters; setId++ ) {
        lsIdx =
        output_layer_set_idx_minus1[ output_point_layer_set_idx[ setId ] ] + 1
        numOutputLayers = [ numLayersInIdList[ lsIdx ] - 1 ]
    }
```

[0625] 또는 하기로 변경될 수 있다.

표 39

```
[0626] for( k = 1; k <=
        numLayersInIdList[output_layer_set_idx_minus1[output_
        point_layer_set_idx[ i ] ]]-1; i++ ) {
        oop_dpb_maxbuffering_parameters( i, k )
```

[0627] 하나의 실시형태에서, 추가적인 플래그는 oop_dpb_information_parameters가 다음과 같이 특별한 계층에 대해 시그널링되는지를 표시하기 위하여 시그널링될 수도 있다:

표 40

for(k = 1; k <= vps_max_layers_minus1; i++) {	
vps_layer_info_present_flag[k]	u(1)
oop_dpb_maxbuffering_parameters(i, k)	
}	

[0628]

[0629]

1과 동일한 vps_layer_info_present_flag[k]는, oop_dpb_maxbuffering_parameters가 특별한 출력 계층 세트에 대한 k번째 계층에 대해 존재하는 것으로 특징한다. 0과 동일한 vps_layer_info_present_flag[k]는, oop_dpb_maxbuffering_parameters가 특별한 출력 계층 세트에 대한 k번째 계층에 대해 존재하지 않는 것으로 특징한다.

[0630]

HRD에 대해 요구된 비-VCL NAL 유닛들의 선택스 엘리먼트들(또는 선택스 엘리먼트들의 일부에 대한 그 디폴트 값들)은 부록들 D 및 E, 조항 7의 시맨틱 하위조항들에서 특징된다.

[0631]

HRD 파라미터 세트들(NAL HRD 파라미터들 및 VCL HRD 파라미터들)의 2 개의 타입들이 이용된다. HRD 파라미터 세트들은, SPS 선택스 구조 또는 VPS 선택스 구조의 일부일 수도 있는 hrd_parameters() 선택스 구조를 통해 시그널링된다.

[0632]

다수의 테스트들은 테스트 중인 비트스트림으로서 지칭되는 비트스트림의 적합성을 검사하기 위하여 필요하게 될 수도 있다. 각각의 테스트에 대하여, 다음의 단계들이 열거된 순서로 적용된다:

[0633]

(1) TargetOpLs로서 나타낸, 테스트 중인 출력 계층 세트가 선택된다. output_layer_set_idx[]에 의해 TargetOpLs에서 참조된 동작 포인트는 테스트 중인 동작 포인트를 식별한다. TargetOpLs의 출력 계층 식별자 리스트 OplayerIdList는, 테스트 중인 비트스트림에서 존재하는 nuh_layer_id 값들의 서브세트인, TargetOp 및 TargetOpLs와 연관된 비트스트림 서브세트에서 존재하는 nuh_layer_id 값들의 증가하는 순서에서, nuh_layer_id 값들의 리스트로 구성된다. TargetOp의 OpTid는 TargetOp와 연관된 비트스트림 서브세트에서 존재하는 최고 TemporalId와 동일하다.

[0634]

(2) TargetDeclayerIdList는 선택된 출력 계층 세트 TargetOpLs에 대한 타겟 디코딩된 계층 식별자 리스트 targetDLayerIdList와 동일하게 설정되고, HighestTid는 TargetOp의 OpTid와 동일하게 설정되고, 조항 10에서 특징된 바와 같은 서브-비트스트림 추출 프로세스는 입력들로서 테스트 중인 비트스트림 HighestTid 및 TargetDeclayerIdList로 호출되고, 출력은 BitstreamToDecode에 배정된다.

[0635]

(3) TargetOp에 적용가능한 hrd_parameters() 선택스 구조 및 sub_layer_hrd_parameters() 선택스 구조가 선택된다. TargetDeclayerIdList가 테스트 중인 비트스트림에서 존재하는 모든 nuh_layer_id 값들을 포함할 경우, 활성 SPS에서의(또는 이 사양에서 특징되지 않은 외부 수단을 통해 제공된) hrd_parameters() 선택스 구조가 선택된다. 이와 다른 경우, TargetOp에 적용되는 활성 VPS에서의(또는 이 사양에서 특징되지 않은 외부 수단을 통해 제공된) hrd_parameters() 선택스 구조가 선택된다. 선택된 hrd_parameters() 선택스 구조 내에서, BitstreamToDecode가 타입 I 비트스트림일 경우, 조건 "if(vcl_hrd_parameters_present_flag)"을 바로 후행하는 sub_layer_hrd_parameters(HighestTid) 선택스 구조가 선택되고, 변수 NalHrdModeFlag는 0과 동일하게 설정되고; 이와 다른 경우(BitstreamToDecode는 타입 II 비트스트림임), 조건 "if(vcl_hrd_parameters_present_flag)" (이 경우, 변수 NalHrdModeFlag는 0과 동일하게 설정됨) 또는 조건 "if(nal_hrd_parameters_present_flag)" (이 경우, 변수 NalHrdModeFlag는 1과 동일하게 설정됨)의 어느 하나를 바로 후행하는 sub_layer_hrd_parameters(HighestTid) 선택스 구조가 선택된다. BitstreamToDecode가 타입 II 비트스트림이고 NalHrdModeFlag가 0과 동일할 때, 필터 데이터 NAL 유닛들을 제외한 모든 비-VCL NAL 유닛들과, 존재할 때에, (부록 B에서 특징된 바와 같은) NAL 유닛 스트림으로부터의 바이트 스트림을 형성하는 모든 leading_zero_8bits, zero_byte, start_code_prefix_one_3bytes, 및 trailing_zero_8bits 선택스 엘리먼트들은 BitstreamToDecode로부터 폐기되고, 나머지 비트스트림은 BitstreamToDecode에 배정된다.

[0636]

준수하는 디코더는 이 하위조항에서 특징된 모든 요건들을 충족시킬 수도 있다.

- [0637] (1) VCL NAL 유닛들에서 참조된 모든 VPS들, SPS들 및 PPS들과, 적절한 버퍼링 주기 및 픽처 타이밍 SEI 메시지들이 비트스트림에서 (비-VCL NAL 유닛들에 의해), 또는 이 사양에서 특정되지 않은 외부 수단에 의해, 시기적절한 방식으로 디코더로 운반된다면, 특정 프로파일, 티어 및 레벨에 대한 적합성을 요구하는 디코더는 부록 A에서 특정된 방식으로 하위조항 C.4에서 특정된 비트스트림 적합성 조건들을 준수하는 모든 비트스트림들을 성공적으로 디코딩할 수 있을 것이다.
- [0638] (2) 비트스트림이 예약된 것으로 특정되는 값들을 가지는 신택스 엘리먼트들을 포함하며 디코더들이 예약된 값들을 갖는 신택스 엘리먼트들을 포함하는 NAL 유닛들 또는 신택스 엘리먼트들의 값들을 무시할 것이라고 특정되고, 비트스트림이 이와 다르게 이 사양을 준수하고 있을 때, 준수하는 디코더는, 그것이 준수하는 비트스트림을 디코딩하는 것과 동일한 방식으로 비트스트림을 디코딩할 것이고, 특정된 바와 같은 예약된 값들을 가지는 신택스 엘리먼트들을 포함하는 NAL 유닛들 또는 신택스 엘리먼트들의 값들을 무시할 것이다.
- [0639] 디코더의 적합성의 2 개의 타입들이 있다: 출력 타이밍 적합성 및 출력 순서 적합성.
- [0640] 디코더의 적합성을 검사하기 위하여, 하위조항 C.4에서 특정된 바와 같이, 요구된 프로파일, 티어 및 레벨을 준수하는 테스트 비트스트림들은 가상 스트림 스케줄러(HSS)에 의해 HRD 및 테스트 중인 디코더(DUT)의 양자로 전달된다. HRD에 의해 출력된 모든 크롭핑된 디코딩된 픽처들은 또한 DUT에 의해 출력될 것이고, DUT에 의해 출력된 각각의 크롭핑된 디코딩된 픽처는 1과 동일한 PicOutputFlag를 갖는 픽처일 것이고, DUT에 의해 출력된 각각의 이러한 크롭핑된 디코딩된 픽처에 대하여, 출력되는 모든 샘플들의 값들은 특정된 디코딩 프로세스에 의해 생성된 샘플들의 값들과 동일할 것이다.
- [0641] 출력 타이밍 디코더 적합성을 위하여, HSS는 비트 레이트 및 CPB 사이즈가 특정된 프로파일, 티어 및 레벨에 대해 부록 A에서 특정된 바와 같이 한정되는 SchedSelIdx의 값들의 서브세트로부터만 선택된 전달 스케줄들로, 또는 비트 레이트 및 CPB 사이즈가 부록 A에서 특정된 바와 같이 한정되는 이하에서 특정된 바와 같은 "보간된 (interpolated)" 전달 스케줄들로, 위에서 설명된 바와 같이 동작한다. 동일한 전달 스케줄은 HRD 및 DUT 양자에 대해 이용된다.
- [0642] HRD 파라미터들 및 버퍼링 주기 SEI 메시지에 0 보다 더 큰 $cpb_cnt_minus1[HighestTid]$ 가 제공될 때, 디코더는 피크 비트 레이트 r , CPB size $c(r)$, 및 초기 CPB 제거 지연을 갖는 것으로 특정된 "보간된" 전달 스케줄을 이용하여 HSS 동작으로부터 전달된 바와 같은 비트스트림을 디코딩할 수 있을 것이다.

수학식 16

$$(f(r) \div r)$$

[0643]

[0644] 다음과 같다:

표 41

$\alpha = (r - BitRate[SchedSelIdx - 1]) + (BitRate[SchedSelIdx] - BitRate[SchedSelIdx - 1]),$	(C-22)
$c(r) = \alpha * CpbSize[SchedSelIdx] + (1 - \alpha) * CpbSize[SchedSelIdx - 1],$	(C-23)
$f(r) = \alpha * InitCpbRemovalDelay[SchedSelIdx] * BitRate[SchedSelIdx] + (1 - \alpha) * InitCpbRemovalDelay[SchedSelIdx - 1] * BitRate[SchedSelIdx - 1]$	(C-24)

[0645]

[0646] 임의의 $SchedSelIdx > 0$ 및 r 에 대하여, $BitRate[SchedSelIdx - 1] \leq r \leq BitRate[SchedSelIdx]$ 가 되어, r 및 $c(r)$ 은 특정된 프로파일, 티어 및 레벨에 대한 최대 비트 레이트 및 버퍼 사이즈에 대하여 부록 A에서 특정된 바와 같은 한계들 내에 있다. $InitCpbRemovalDelay[SchedSelIdx]$ 는 하나의 버퍼링 주기로부터 또 다른 버퍼링 주기까지 상이할 수 있고, 재계산되어야 한다.

[0647] 출력 타이밍 디코더 적합성을 위하여, 위에서 설명된 바와 같은 HRD가 이용되고, 픽처 출력의 (최초 비트의 전달 시간에 관한) 타이밍은 HRD 및 DUT 양자에 대해 고정된 지연까지 동일하다.

- [0648] 출력 순서 디코더 적합성을 위하여, 다음이 적용된다:
- [0649] (1) HSS는 DUT로부터의 "요구에 의해(by demand)" 비트스트림 BitstreamToDecode을 DUT로 전달하고, 이것은 DUT가 그 프로세싱을 진행하기 위하여 더 많은 비트들을 요구할 때에만 (디코딩 순서에서) 비트들을 전달한다는 것을 의미한다. 이것은 이 테스트를 위하여, DUT의 코딩된 픽처 버퍼가 최대 디코딩 유닛의 사이즈만큼 작을 수 있다는 것을 의미한다.
- [0650] (2) 이하에서 설명된 바와 같은 수정된 HRD가 이용되고, HSS는 비트 레이트 및 CPB 사이즈가 부록 A에서 특정된 바와 같이 한정되도록 비트스트림 BitstreamToDecode에서 특정된 스케줄들 중의 하나에 의해 비트스트림을 HRD로 전달한다. 출력된 픽처들의 순서는 HRD 및 DUT 양자에 대해 동일할 것이다.
- [0651] (3) HRD CPB 사이즈는 하위조항 E.2.3에서 특정된 바와 같이 CpbSize[SchedSelIdx]에 의해 주어지고, 여기서, SchedSelIdx 및 HRD 파라미터들은 하위조항 C.1에서 특정된 바와 같이 선택된다. DPB 사이즈는 활성 SPS로부터(현재의 디코딩된 픽처에 대한 nuh_layer_id가 0과 동일할 때), 또는 현재의 디코딩된 픽처의 nuh_layer_id의 값에 대한 활성 계층 SPS로부터의 sps_max_dec_pic_buffering_minus1[HighestTid] + 1에 의해 주어진다.
- [0652] 일부의 경우들에는, 출력 계층 세트들 DPB 정보 파라미터들 oop_dpb_maxbuffering_parameters()이 선택된 출력 계층 세트에 대해 존재할 경우, DPB 사이즈는 max_vps_dec_pic_buffering_minus1[CurrLayerId][HighestTid]에 의해 주어지고, 여기서, currLayerId는 현재의 디코딩된 픽처의 nuh_layer_id이다. 이와 다르게, 출력 계층 DPB 정보 파라미터들 oop_dpb_maxbuffering_parameters ()이 선택된 출력 계층 세트에 대해 존재하지 않을 경우, DPB 사이즈는 활성 SPS로부터(현재의 디코딩된 픽처에 대한 nuh_layer_id가 0과 동일할 때), 또는 현재의 디코딩된 픽처의 nuh_layer_id의 값에 대한 활성 계층 SPS로부터의 sps_max_dec_pic_buffering_minus1[HighestTid] + 1에 의해 주어진다.
- [0653] HRD 에 대한 CPB 로부터의 제거 시간은 최종 비트 도달 시간이고, 디코딩은 즉각적이다. 이 HRD 의 DPB 의 동작은 하위조항들 C.5.2 내지 C.5.2.3에서 설명된 바와 같다.
- [0654] 디코딩된 픽처 버퍼는 픽처 저장 버퍼들을 포함한다. 0과 동일한 nuh_layer_id에 대한 픽처 저장 버퍼들의 수는 활성 SPS로부터 유도된다. 각각의 제로가 아닌 nuh_layer_id 값에 대한 픽처 저장 버퍼들의 수는 그 제로가 아닌 nuh_layer_id 값에 대한 활성 계층 SPS로부터 유도된다. 픽처 저장 버퍼들의 각각은 "참조를 위해 이용됨"으로서 표기되거나 미래의 출력을 위해 이용되는 디코딩된 픽처를 포함한다. 하위조항 F.13.5.2.2에서 특정된 바와 같은, DPB로부터의 픽처들의 출력 및 제거를 위한 프로세스가 호출되고, 하위조항 F.13.5.2.3에서 특정된 바와 같은 픽처 디코딩, 표기, 추가적인 범핑, 및 저장을 위한 프로세스의 호출이 뒤따른다. "범핑" 프로세스는 하위조항 F.13.5.2.4에서 특정되고, 하위조항들 F.13.5.2.2 및 F.13.5.2.3에서 특정된 바와 같이 호출된다.
- [0655] 현재의 픽처의 디코딩 전의(그러나, 현재의 픽처의 최초 슬라이스의 슬라이스 헤더를 파싱한 후의) DPB로부터의 픽처들의 출력 및 제거는, 현재의 픽처를 포함하는 액세스 유닛의 최초 디코딩 유닛이 CPB로부터 제거될 때에 순간적으로 발생하고 다음과 같이 진행된다.
- [0656] 하위조항 8.3.2에서 특정된 바와 같은 RPS에 대한 디코딩 프로세스가 호출된다.
- [0657] (1) 현재의 픽처가, 1과 동일한 NoRas1OutputFlag를 가지며 픽처 0이 아닌 0과 동일한 nuh_layer_id를 갖는 IRAP 픽처일 경우, 다음의 순서화된 단계들이 적용된다:
- [0658] (A) 변수 NoOutputOfPriorPicsFlag는 다음과 같이 테스트 중인 디코더에 대해 유도된다:
- [0659] (i) 현재의 픽처가 CRA 픽처일 경우, NoOutputOfPriorPicsFlag는 (no_output_of_prior_pics_flag의 값에 관계없이) 1과 동일하게 설정된다.
- [0660] (ii) 이와 다르게, 활성 SPS로부터 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid]의 값이 선행하는 픽처에 대해 활성인 SPS로부터 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid]의 값과는 각각 상이할 경우, NoOutputOfPriorPicsFlag는 no_output_of_prior_pics_flag의 값에 관계없이, 테스트 중인 디코더에 의해 1로 설정될 수도 있다(그러나, 설정되지 않아야 함). NoOutputOfPriorPicsFlag를 no_output_of_prior_pics_flag와 동일하게 설정하는 것이 이 조건들 하에서 선호되

지만, 테스트 중인 디코더는 이 경우에 NoOutputOfPriorPicsFlag를 1로 설정하도록 허용된다.

- [0661] (iii) 이와 다를 경우, NoOutputOfPriorPicsFlag는 no_output_of_prior_pics_flag와 동일하게 설정된다.
- [0662] (B) 테스트 중인 디코더에 대해 유도된 NoOutputOfPriorPicsFlag의 값은 다음과 같이 HRD에 대해 적용된다:
- [0663] (i) NoOutputOfPriorPicsFlag가 1과 동일할 경우, DPB에서의 모든 픽처 저장 버퍼들은 이들이 포함하는 픽처들의 출력 없이 비워지고, DPB 충만도(fullness)는 0과 동일하게 설정된다.
- [0664] (ii) 이와 다를 경우(NoOutputOfPriorPicsFlag가 0과 동일함), "출력을 위해 필요하지 않음" 및 "참조를 위해 이용되지 않음"으로서 표기되는 픽처를 포함하는 모든 픽처 저장 버퍼들은 (출력 없이) 비워지고, DPB에서의 모든 비워지지 않은 픽처 저장 버퍼들은 하위조항 F.13.5.2.4에서 특정된 "범핑" 프로세스를 반복적으로 호출함으로써 비워지고, DPB 충만도는 0과 동일하게 설정된다.
- [0665] (iii) 이와 다를 경우(현재의 픽처가 1과 동일한 NoRaslOutputFlag 및 0과 동일한 nuh_layer_id를 갖는 IRAP 픽처가 아님), "출력을 위해 필요하지 않음" 및 "참조를 위해 이용되지 않음"으로서 표기되는 픽처를 포함하는 모든 픽처 저장 버퍼들이 (출력 없이) 비워진다. 비워지는 각각의 픽처 저장 버퍼에 대하여, DPB 충만도는 1씩 감소된다. 변수 currLayerId는 현재의 디코딩된 픽처의 nuh_layer_id와 동일하게 설정된다.
- [0666] 변수들 MaxNumReorderPics[currLayerId][HighestTid], MaxLatencyIncreasePlus1[currLayerId][HighestTid], MaxLatencyPictures[currLayerId][HighestTid], MaxDecPicBufferingMinus1[currLayerId][HighestTid]은 다음과 같이 유도된다:
- [0667] 계층 DPB 정보 파라미터들 layer_dpb_info_parameters()이 VPS에서 존재할 경우, MaxNumReorderPics[currLayerId][HighestTid]는 currLayerId가 0과 동일할 때에 vps_max_num_reorder_pics[HighestTid]로 설정되거나, currLayerId가 0보다 더 클 때에 currLayerId에 대한 max_vps_num_reorder_pics[CurrLayerId][HighestTid]로 설정된다. 이와 다르게, 계층 DPB 정보 파라미터들 layer_dpb_info_parameters()이 존재하지 않을 경우, MaxNumReorderPics[currLayerId][HighestTid]는 활성 SPS로부터(currLayerId가 0과 동일할 때), 또는 currLayerId의 값에 대한 활성 계층 SPS로부터의 sps_max_num_reorder_pics[HighestTid]로 설정된다.
- [0668] 계층 DPB 정보 파라미터들 layer_dpb_info_parameters()이 VPS에서 존재할 경우, MaxLatencyIncreasePlus1[currLayerId][HighestTid]는 currLayerId가 0과 동일할 때에 vps_max_latency_increase_plus1[HighestTid]로 설정되거나, currLayerId가 0보다 더 클 때에 currLayerId에 대한 max_vps_latency_increase_plus1[CurrLayerId][HighestTid]로 설정된다. 계층 DPB 정보 파라미터들 layer_dpb_info_parameters()이 VPS에서 존재할 경우, MaxLatencyPictures[currLayerId][HighestTid]는 currLayerId가 0과 동일할 때에 SpsMaxLatencyPictures [HighestTid]로 설정되거나, currLayerId가 0보다 더 클 때에 currLayerId에 대한 VpsMaxLatencyPictures [CurrLayerId][HighestTid]로 설정된다. 이와 다르게, 계층 DPB 정보 파라미터들 layer_dpb_info_parameters()이 테스트 중인 동작 포인트에 대해 존재하지 않을 경우, MaxLatencyIncreasePlus1[currLayerId][HighestTid]는 활성 SPS(currLayerId가 0과 동일할 때), 또는 currLayerId의 값에 대한 활성 계층 SPS의 sps_max_latency_increase_plus1[HighestTid]로 설정되고, MaxLatencyPictures[currLayerId][HighestTid]는 활성 SPS로부터(currLayerId가 0과 동일할 때), 또는 currLayerId의 값에 대한 활성 계층 SPS로부터 유도된 SpsMaxLatencyPictures[currLayerId][HighestTid]로 설정된다.
- [0669] 동작 포인트 DPB 정보 파라미터들 oop_dpb_maxbuffering_parameters ()이 선택된 출력 계층 세트에 대해 존재할 경우, MaxDecPicBufferingMinus1[currLayerId][HighestTid]은 currLayerId가 0과 동일할 때에 vps_max_dec_pic_buffering_minus1[HighestTid]로 설정되거나, currLayerId가 0보다 더 클 때에 테스트 중인 동작 포인트에 대한 currLayerId에 대한 max_vps_dec_pic_buffering_minus1[CurrLayerId][HighestTid]로 설정된다. 이와 다르게, 동작 포인트 DPB 정보 파라미터들 oop_dpb_maxbuffering_parameters ()이 테스트 중인 동작 포인트에 대해 존재하지 않을 경우, MaxDecPicBufferingMinus1[currLayerId][HighestTid]는 활성 SPS로부터(currLayerId가 0과 동일할 때), 또는 currLayerId의 값에 대한 활성 계층 SPS로부터의 sps_max_dec_pic_buffering_minus1[HighestTid]로 설정된다.
- [0670] 다음 조건들 중의 하나 이상이 참일 때에는, 다음 조건들의 어느 것도 참이 아닐 때까지, 비워지는 각각의 추가적인 픽처 저장 버퍼에 대하여 DPB 충만도를 1씩 추가로 감소시키면서, 하위조항 F.13.5.2.4에서 특정된 "범핑"

프로세스가 반복적으로 호출된다.

- [0671] (1) "출력을 위해 필요함"으로서 표기되는 DPB에서의 currLayerId와 동일한 nuh_layer_id를 갖는 픽처들의 수는 MaxNumReorderPics[CurrLayerId][HighestTid]보다 더 크다.
- [0672] (2) MaxLatencyIncreasePlus1[CurrLayerId][HighestTid]가 0과 동일하지 않고, 연관된 변수 PicLatencyCount[currLayerId]가 MaxLatencyPictures[CurrLayerId][HighestTid] 이상인, "출력을 위해 필요함"으로서 표기되는 DPB에서의 currLayerId와 동일한 nuh_layer_id를 갖는 적어도 하나의 픽처가 있을 경우.
- [0673] (3) DPB에서의 currLayerId와 동일한 nuh_layer_id를 갖는 픽처들의 수는 MaxDecPicBuffering[CurrLayerId][HighestTid] 이상이다.
- [0674] 이 하위조항에서 특정된 프로세스들은 현재의 픽처를 포함하는 액세스 유닛 n의 최후 디코딩 유닛이 CPB로부터 제거될 때 순간적으로 발생한다.
- [0675] 변수 currLayerId는 현재의 디코딩된 픽처의 nuh_layer_id와 동일하게 설정된다.
- [0676] "출력을 위해 필요함"으로서 표기되며 currLayerId와 동일한 nuh_layer_id 값을 가지는 DPB에서의 각각의 픽처에 대하여, 연관된 변수 PicLatencyCount[currLayerId]는 PicLatencyCount[currLayerId] + 1와 동일하게 설정된다.
- [0677] 현재의 픽처는 픽처의 최후 디코딩 유닛이 디코딩된 후에 디코딩된 것으로 간주된다. 현재의 디코딩된 픽처는 DPB에서의 비워진 픽처 저장 버퍼에서 저장되고, 다음이 적용된다:
- [0678] (A) 현재의 디코딩된 픽처가 1과 동일한 PicOutputFlag를 가질 경우, 그것은 "출력을 위해 필요함"으로서 표기되고, 그 연관된 변수 PicLatencyCount[currLayerId]는 0과 동일하게 설정된다.
- [0679] (B) 이와 다를 경우(현재의 디코딩된 픽처가 0과 동일한 PicOutputFlag를 가짐), 그것은 "출력을 위해 필요하지 않음"으로서 표기된다.
- [0680] 현재의 디코딩된 픽처는 "단기 참조를 위해 이용됨"으로서 표기된다.
- [0681] 다음의 조건들 중의 하나 이상이 참일 때에는, 다음의 조건들의 어느 것도 참이 아닐 때까지, 하위조항 F.13.5.2.4에서 특정된 "범핑" 프로세스가 반복적으로 호출된다:
- [0682] (A) "출력을 위해 필요함"으로서 표기되는 DPB에서의 currLayerId와 동일한 nuh_layer_id를 갖는 픽처들의 수는 MaxNumReorderPics[CurrLayerId][HighestTid]보다 더 크다.
- [0683] (B) MaxLatencyIncreasePlus1[CurrLayerId][HighestTid]가 0과 동일하지 않고, 연관된 변수 PicLatencyCount[currLayerId]가 MaxLatencyPictures[CurrLayerId][HighestTid] 이상인, "출력을 위해 필요함"으로서 표기되는 DPB에서의 currLayerId와 동일한 nuh_layer_id를 갖는 적어도 하나의 픽처가 있을 경우.
- [0684] "범핑" 프로세스는 다음의 순서화된 단계들로 구성된다:
- [0685] (A) 출력을 위해 최초인 픽처들은 "출력을 위해 필요함"으로서 표기된 DPB에서의 모든 픽처들의 PicOrderCntVal의 최소 값을 가지는 것들로서 선택된다.
- [0686] (B) 이 픽처들은 0과 동일한 nuh_layer_id를 갖는 픽처에 대한 활성 SPS에서, 또는 픽처의 그것과 동일한 nuh_layer_id 값에 대한 활성 계층 SPS에서 특정된 적합성 크롭핑 윈도우를 이용하여 크롭핑되고, 크롭핑된 픽처들은 nuh_layer_id의 오름차순으로 출력되고, 픽처들은 "출력을 위해 필요하지 않음"으로서 표기된다.
- [0687] (C) "참조를 위해 이용되지 않음"으로서 표기된 픽처를 포함하며 크롭핑 및 출력되었던 픽처들 중의 하나를 포함하였던 각각의 픽처 저장 버퍼는 비워진다.
- [0688] 도 62를 참조하면, 예시적인 수정된 시퀀스 파라미터 세트(sps) 선택스 seq_parameter_set_rbsp가 예시되어 있다. 수정된 sps는 sps_dpb_params_present_flag를 포함한다. 이 플래그의 값에 기반하고 nuh_layer_id의 값에 기반한 하나의 실시형태에서, 선택스 엘리먼트들(예컨대, sps_sub_layer_ordering_info_present_flag, sps_max_dec_pic_buffering_minus1[i], sps_max_num_reorder_pics[i], sps_max_latency_increase_plus1[i])의 일부는 시그널링되지 않을 수도 있다.
- [0689] 0과 동일한 sps_dpb_params_present_flag는, nuh_layer_id >0인 SPS에 대하여, 선택스 엘리먼트들 sps_sub_layer_ordering_info_present_flag, sps_max_dec_pic_buffering_minus1[i],

sps_max_num_reorder_pics[i], sps_max_latency_increase_plus1[i]은 이 SPS에서 존재하지 않는 것으로 특정한다. 이 파라미터들의 값은 활성 VPS 확장에서 시그널링된 sub_layer_vps_buf_info_present_flag[i], max_vps_num_reorder_pics[i][j], 및 max_vps_latency_increase_plus1[i][j] 파라미터들의 값과 동일하게 설정된다. 1과 동일한 sps_dpb_params_present_flag는, nuh_layer_id >0인 SPS에 대하여, 신택스 엘리먼트들 sps_sub_layer_ordering_info_present_flag, sps_max_dec_pic_buffering_minus1[i], sps_max_num_reorder_pics[i], sps_max_latency_increase_plus1[i]은 이 SPS에서 존재하는 것으로 특정한다.

[0690] 또 다른 실시형태에서, 신택스 엘리먼트들 중의 하나 이상은 ue(v) 대신의 u(v) 대신으로 알려진 고정된 수의 비트들을 이용하여 시그널링될 수도 있다. 예를 들어, 이들은 u(8) 또는 u(16) 또는 u(32) 또는 u(64) 등을 이용하여 시그널링될 수 있다.

[0691] 또 다른 실시형태에서, 이 신택스 엘리먼트 중의 하나 이상은 u(v) 코딩과 같은 고정된 수의 비트들 대신으로, ue(v) 또는 일부의 다른 코딩 방식으로 시그널링될 수 있다.

[0692] 또 다른 실시형태에서, 다양한 신택스 엘리먼트들 및 그 시맨틱들의 명칭들은 plus1 또는 plus2를 가산함으로써, 또는 설명된 신택스 및 시맨틱들에 비해 minus1 또는 minus2를 감산함으로써 변경될 수도 있다.

[0693] 또 다른 실시형태에서, 다양한 신택스 엘리먼트들은 비트스트림 내의 어딘가에서 픽처 당 시그널링될 수도 있다. 예를 들어, 이들은 슬라이스 세그먼트 헤더, pps/ sps/ vps/ 또는 임의의 다른 파라미터 세트 또는 비트스트림의 다른 규범 부분에서 시그널링될 수도 있다.

[0694] 또 다른 실시형태들에서, 출력 계층 세트들에 관련된 이 발명에서 정의된 모든 개념들은 출력 동작 포인트들 [2,3], 및/또는 동작 포인트들[1]에 적용될 수 있다.

[0695] **예 3**

[0696] 비디오 코딩을 위한 방법이 개시되어 있다. 방법은 현재의 픽처의 최초 슬라이스 헤더를 파싱하기 시작하는 것을 포함한다. 디코딩된 픽처 버퍼(DPB)에 의해 수행된 어느 단계들이 픽처 기반일 것이고 어느 단계들이 액세스 유닛(AU) 기반일 것인지 결정된다. DPB로부터의 제거가 수행된다. DPB로부터의 픽처 출력이 수행된다. DPB에서의 현재의 디코딩된 픽처의 디코딩 및 저장이 수행된다. DPB에서의 현재의 디코딩된 픽처가 표기된다. DPB로부터의 추가적인 픽처 출력이 또한 수행된다.

[0697] 일부의 구성들에서, DPB로부터의 제거 및 출력은 AU 출력 플래그, AU rasl 출력 없음 플래그 및/또는 AU 이전 픽처들의 출력 없음 플래그와 같은 적어도 하나의 AU 출력 플래그에 기반으로 할 수도 있다. AU 출력 플래그는 비트스트림에서 시그널링된 신택스 엘리먼트들 및 다른 조건들에 기반으로 하여 유도될 수도 있다. AU 플래그들은 액세스 유닛(AU) 레벨에서 유도 및 적용된 플래그들을 나타낸다. 일부의 경우들에는, 이것들은 픽처 출력 플래그(예컨대, pic_output_flag), 픽처 rasl 출력 없음 플래그(예컨대, NoRaslOutputFlag), 및/또는 픽처 이전 픽처들의 출력 없음 플래그(예컨대, no_output_of_prior_pics flag)와 같이, 픽처 레벨에서 시그널링되거나 유도되는 유사한 플래그들과는 상이한 플래그들일 것이다.

[0698] 하나의 구성에서, 제 1 AU 출력 플래그는 DPB에 의해 유도 및 이용될 수도 있고, 제 2 AU 출력 플래그는 DPB에 의해 유도 및 이용된다. 또 다른 구성에서, 모든 픽처 출력 플래그 신택스 엘리먼트 값들은, 픽처 출력 플래그가 최초 슬라이스 헤더에서 존재할 때에 AU에서의 모든 코딩된 픽처들에 대한 동일한 값으로 제약될 수도 있다. 일부의 구성들에서, 모든 픽처 출력 플래그 신택스 엘리먼트 값들은, 픽처 출력 없음 플래그가 최초 슬라이스 헤더에서 존재할 때에 AU에서의 모든 코딩된 픽처들에 대한 동일한 값으로 제약될 수도 있다.

[0699] 하나의 구성에서, 제거는 픽처 기반일 수도 있고, 픽처 출력은 AU(액세스 유닛) 기반일 수도 있고, 저장 및 디코딩은 픽처 기반일 수도 있고, 표기는 픽처 기반일 수도 있고, 추가적인 픽처 출력은 AU 기반일 수도 있다. DPB로부터의 제거는 현재의 픽처의 디코딩 전에, DPB로부터 하나 이상의 픽처들을 제거할 수도 있다.

[0700] 또 다른 구성에서, 제거는 픽처 기반일 수도 있고, 픽처 출력, 디코딩 및 저장, 표기 및 추가적인 픽처 출력은 AU 기반일 수도 있다. 또 다른 구성에서, 제거, 픽처 출력, 디코딩 및 저장, 표기 및 추가적인 픽처 출력은 AU 기반일 수도 있다. 또 다른 구성에서, 제거, 픽처 출력, 디코딩 및 저장, 표기 및 추가적인 픽처 출력은 픽처 기반일 수도 있다.

[0701] DPB에서 현재의 디코딩된 픽처를 표기하는 것은 참조 표기 단계 및 출력 표기 단계를 포함할 수도 있다. 참조

표기 단계는 픽처 기반일 수도 있고, 출력 표기 단계는 AU 기반일 수도 있다. DPB 충만도는 디코딩 픽처가 비워진 저장 버퍼 내의 DPB에서 저장될 때에 1씩 증가될 수도 있다. DPB 충만도는 픽처가 DPB로부터 출력될 때에 1씩 감소될 수도 있다. DPB 충만도는 계층 당 추적될 수도 있다. DPB 충만도는 또한, 출력 계층 세트에 대해 추적될 수도 있다.

[0702] DPB는 상이한 해상도들, 상이한 비트-심도들 및 상이한 컬러 색도(chromaticity) 중의 하나 이상을 가지는 디코딩된 픽처들에 대한 별도로 식별되고 관리된 픽처 버퍼들을 포함할 수도 있다. DPB는 픽처 저장 버퍼들의 공통의 풀(pool)을 포함할 수도 있다. 디코딩된 픽처는 사이즈, 해상도 및 비트-심도 중의 적어도 하나에 기반으로 하여 픽처 저장 버퍼들에서 저장될 수도 있다. 하나의 구성에서, 디코딩된 픽처는 픽처 저장 버퍼들의 하나의 픽처 버퍼 슬롯 내에 저장될 수도 있다.

[0703] 방법은 스케일러블 고효율 비디오 코딩(scalable high efficiency video coding; SHVC) 표준을 준수하는 전자 디바이스 내의 디코더에 의해 수행될 수도 있다. 방법은 또한, 멀티-뷰 고효율 비디오 코딩(multi-view high efficiency video coding; MV-HEVC) 표준을 준수하는 전자 디바이스 내의 디코더에 의해 수행될 수도 있다.

[0704] 비디오 코딩을 위해 구성된 전자 디바이스가 또한 개시되어 있다. 전자 디바이스는 프로세서와, 프로세서와 전자 통신하는 메모리를 포함한다. 메모리에서의 명령들은 현재의 픽처의 최초 슬라이스 헤더를 파싱하기 시작하도록 실행가능하다. 메모리에서의 명령들은 또한, 디코딩된 픽처 버퍼(DPB)에 의해 수행된 어느 단계들이 픽처 기반일 것이고 어느 단계들이 액세스 유닛(AU) 기반일 것인지 결정하도록 실행가능하다. 메모리에서의 명령들은 DPB로부터의 제거를 수행하도록 추가로 실행가능하다. 메모리에서의 명령들은 또한, DPB로부터의 픽처 출력을 수행하도록 실행가능하다. 메모리에서의 명령들은 DPB에서의 현재의 디코딩된 픽처의 디코딩 및 저장을 수행하도록 추가로 실행가능하다. 메모리에서의 명령들은 또한, DPB에서의 현재의 디코딩된 픽처를 표기하도록 실행가능하다. 메모리에서의 명령들은 DPB로부터의 추가적인 픽처 출력을 수행하도록 추가로 실행가능하다.

[0705] 다양한 구성들은 도면들을 참조하여 지금부터 설명되며, 여기서, 유사한 참조 번호들은 기능적으로 유사한 엘리먼트들을 표시할 수도 있다. 본원의 도면들에서 일반적으로 설명되고 예시된 바와 같은 시스템들 및 방법들은 매우 다양한 상이한 구성들로 배치되고 설계될 수 있다. 따라서, 도면들에서 나타난 바와 같이, 몇몇 구성들의 다음의 더욱 상세한 설명은 청구된 바와 같이 범위를 제한하도록 의도된 것이 아니라, 단지 시스템들 및 방법들을 대표한다.

[0706] 도 63은 다수의 전자 디바이스들(2102a 내지 2102b) 사이의 비디오 코딩을 예시하는 블록도이다. 제 1 전자 디바이스(2102a) 및 제 2 전자 디바이스(2102b)가 예시되어 있다. 그러나, 제 1 전자 디바이스(2102a) 및 제 2 전자 디바이스(2102b)와 관련하여 설명된 특징들 및 기능성의 하나 이상은 일부의 구성들에서 단일 전자 디바이스(102)로 조합될 수도 있다는 것에 주목해야 한다. 각각의 전자 디바이스(102)는 비디오를 인코딩하고 및/또는 비디오를 디코딩하도록 구성될 수도 있다. 전자 디바이스들(102)은 하이브리드 디코딩된 픽처 버퍼(DPB) 동작을 이용하도록 구성될 수도 있다. 하이브리드 디코딩된 픽처 버퍼(DPB) 동작은 디코딩된 픽처 버퍼(DPB)(116) 상에서 수행된 제거, 출력(범핑), 저장, 표기 및 추가적인 출력(범핑)의 다양한 단계들이 픽처 기반 또는 액세스 유닛(AU) 기반의 어느 하나로 발생하는 시나리오들을 지칭한다. 이 단계들의 특정 조합들은 픽처 기반 또는 액세스 유닛(AU) 기반으로 수행되는 것으로 지칭되지만, 픽처 기반 또는 액세스 유닛(AU) 기반의 어느 하나로 이 단계들의 각각을 개별적으로 행하는 모든 가능한 조합들이 지원된다.

[0707] 본원에서 이용된 바와 같이, 액세스 유닛(AU)은, 특정된 분류 규칙에 따라 서로 연관되고, 디코딩 순서에서 연속적이고, 동일한 출력 시간 및 그 연관된 비-VCL NAL 유닛들과 연관된 모든 코딩된 픽처들의 비디오 코딩 계층(VCL) NAL 유닛들을 포함하는 네트워크 추상화 계층(NAL) 유닛들의 세트를 지칭한다. 기본 계층은 모든 VCL NAL 유닛들이 0과 동일한 nuh_layer_id를 가지는 계층이다. 코딩된 픽처는, nuh_layer_id의 특별한 값을 갖는 VCL NAL 유닛들을 포함하며 픽처의 모든 코딩 트리 유닛들을 포함하는 픽처의 코딩된 표현이다. 일부의 경우에는, 코딩된 픽처가 계층 컴포넌트로 칭해질 수도 있다. 픽처 기반 또는 액세스 유닛(AU) 기반인 단계들에 대한 추가적인 세부사항들은 이하의 도 69 및 도 70에 관련하여 주어진다.

[0708] 하나의 구성에서, 전자 디바이스들(102)의 각각은 고효율 비디오 코딩(HEVC) 표준, 스케일러블 고효율 비디오 코딩(SHVC) 표준 또는 멀티-뷰 고효율 비디오 코딩(MV-HEVC) 표준을 준수할 수도 있다. HEVC 표준은, H.264/MPEG-4 AVC(Advanced Video Coding; 진보된 비디오 코딩)에 대한 계승자로서 작동하며 개선된 비디오 품질 및 증가된 데이터 압축 비율들을 제공하는 비디오 압축 표준이다. 본원에서 이용된 바와 같이, 픽처는 단색 포맷인 루마 샘플들의 어레이, 또는 4:2:0, 4:2:2 및 4:4:4 컬러 포맷 또는 일부의 다른 컬러 포맷인 루마 샘플들의 어레이 및 크로마 샘플들의 2 개의 대응하는 어레이들이다. 가상 참조 디코더(HRD)의 동작 및 출력 순서

디코딩된 픽처 버퍼(DPB)(116)의 동작은 JCTVC-M1008, JCTVC-L1008, JCTVC-D1004, JCT3V-C1004, JCTVC-L0453 및 JCTVC-L0452에서의 SHVC 및 MV-HEVC에 대해 설명된다.

- [0709] 제 1 전자 디바이스(2102a)는 인코더(2108) 및 오버헤드 시그널링 모듈(2112)을 포함할 수도 있다. 제 1 전자 디바이스(2102a)는 입력 픽처(2106)를 획득할 수도 있다. 일부의 구성들에서, 입력 픽처(2106)는 이미지 센서를 이용하여 제 1 전자 디바이스(2102a) 상에서 캡처될 수도 있고, 메모리로부터 추출될 수도 있고, 및/또는 또 다른 전자 디바이스(102)로부터 수신될 수도 있다. 인코더(2108)는 인코딩된 데이터(2110)를 생성하기 위하여 입력 픽처(2106)를 인코딩할 수도 있다. 예를 들어, 인코더(2108)는 일련의 입력 픽처들(2106)(예컨대, 비디오)을 인코딩할 수도 있다. 인코딩된 데이터(2110)는 디지털 데이터(예컨대, 비트스트림)일 수도 있다.
- [0710] 오버헤드 시그널링 모듈(2112)은 인코딩된 데이터(2110)에 기반으로 하여 오버헤드 시그널링을 생성할 수도 있다. 예를 들어, 오버헤드 시그널링 모듈(2112)은 슬라이스 헤더 정보, 비디오 파라미터 세트(VPS) 정보, 시퀀스 파라미터 세트(SPS) 정보, 픽처 파라미터 세트(PPS) 정보, 픽처 순서 카운트(picture order count; POC), 참조 픽처 지정 등과 같이, 오버헤드 데이터를 인코딩된 데이터(2110)에 추가할 수도 있다. 일부의 구성들에서, 오버헤드 시그널링 모듈(2112)은 픽처들의 2 개의 세트들 사이의 천이(transition)를 표시하는 랩 표시자(wrap indicator)를 생성할 수도 있다.
- [0711] 인코더(2108)(및 예를 들어, 오버헤드 시그널링 모듈(2112))는 비트스트림(2114)을 생성할 수도 있다. 비트스트림(2114)은 입력 픽처(2106)에 기반으로 하여 인코딩된 픽처 데이터를 포함할 수도 있다. 일부의 구성들에서, 비트스트림들(2114)은 또한, 슬라이스 헤더 정보, VPS 정보, SPS 정보, PPS 정보 등과 같은 오버헤드 데이터를 포함할 수도 있다. 추가적인 입력 픽처들(2106)이 인코딩되므로, 비트스트림(2114)은 하나 이상의 인코딩된 픽처들을 포함할 수도 있다. 예를 들어, 비트스트림(2114)은 하나 이상의 인코딩된 참조 픽처들 및/또는 다른 픽처들을 포함할 수도 있다.
- [0712] 비트스트림(2114)은 디코더(2104)에 제공될 수도 있다. 하나의 예에서, 비트스트림(2114)은 유선 또는 무선 링크를 이용하여 전자 디바이스(2102b)로 송신될 수도 있다. 일부의 경우들에는, 이것이 인터넷 또는 로컬 영역 네트워크(LAN)와 같은 네트워크를 통해 행해질 수도 있다. 도 63에서 예시된 바와 같이, 디코더(2104)는 제 1 전자 디바이스(2102a) 상의 디코더(2108)와는 별도로 제 2 전자 디바이스(2102b) 상에서 구현될 수도 있다. 그러나, 인코더(2108) 및 디코더(2104)는 일부의 구성들에서 동일한 전자 디바이스(102) 상에서 구현될 수도 있다는 것에 주목해야 한다. 인코더(2108) 및 디코더(2104)가 동일한 전자 디바이스 상에서 구현될 때, 예를 들어, 비트스트림(2114)이 버스를 통해 디코더(2104)에 제공될 수도 있거나, 디코더(2104)에 의한 추출을 위해 메모리 내에 저장될 수도 있다.
- [0713] 디코더(2104)는 비트스트림(2114)을 수신(예컨대, 획득)할 수도 있다. 디코더(2104)는 비트스트림(2114)에 기반으로 하여 디코딩된 픽처(2118)(예컨대, 하나 이상의 디코딩된 픽처들(2118))을 생성할 수도 있다. 디코딩된 픽처(2118)는 디스플레이될 수도 있고, 재생될 수도 있고, 메모리 내에 저장될 수도 있고, 및/또는 또 다른 디바이스로 송신될 수도 있는 등등과 같다.
- [0714] 디코더(2104)는 디코딩된 픽처 버퍼(DPB)(116)를 포함할 수도 있다. 디코딩된 픽처 버퍼(DPB)(116)는 가상 참조 디코더(HRD)에 대해 특정된 참조, 출력 재순서화 또는 출력 지연을 위한 디코딩된 픽처들을 유지하는 버퍼일 수도 있다. 전자 디바이스(102) 상에서, 디코딩된 픽처 버퍼(DPB)(116)는 디코더(2104)에서 재구성된(예컨대, 디코딩된) 픽처들을 저장하기 위하여 이용될 수도 있다. 다음으로, 이 저장된 픽처들은 예를 들어, 인터-예측 메커니즘에서 이용될 수도 있다. 픽처들이 순서에 관계없이 디코딩될 때, 픽처들은 이들이 순서대로 더 이후에 디스플레이될 수 있도록 디코딩된 픽처 버퍼(DPB)(116)에서 저장될 수도 있다.
- [0715] H.264 또는 진보된 비디오 코딩(AVC) 표준에서는, 디코딩된 픽처 버퍼(DPB)(116) 관리(예컨대, 삭제, 픽처들의 추가, 픽처들의 재순서화 등)가 메모리 관리 제어 동작들(memory management control operations; MMCO)을 이용하여 수행된다. 많은 상이한 디코딩된 픽처 버퍼(DPB)(116) 관리 접근법들이 고려 중에 있다.
- [0716] 디코더(2104)는 하이브리드 디코딩된 픽처 버퍼(DPB) 동작 모듈(2120)을 포함할 수도 있다. 하이브리드 디코딩된 픽처 버퍼(DPB) 동작 모듈(2120)은 픽처 기반 단계들(2122)을 갖는 디코딩된 픽처 버퍼(DPB)(116) 관리 접근법들, 및/또는 액세스 유닛(AU) 기반 단계들(2124)을 갖는 디코딩된 픽처 버퍼(DPB)(116) 관리 접근법들을 허용할 수도 있다. 예를 들어, 제거, 저장 및 참조 표기를 위한 픽처 기반 단계들(2124)의 이용의 하나의 장점은, 최적의 디코딩된 픽처 버퍼(DPB)(116) 메모리가 다양한 계층들에 의해 이용될 것이라는 점이다. 이에 따라, 픽처 기반 단계들을 이용할 때의 전체적인 요구된 메모리는 더 낮을 수도 있다. 출력(출력, 출력 표기 및 추가적

인 출력을 포함함)을 위한 액세스 유닛(AU) 기반 단계들(2124)의 이용의 하나의 장점은 출력 프로세스가 간략화 될 수도 있다는 점이다.

[0717] 도 64는 하이브리드 디코딩된 픽처 버퍼(DPB)(116) 동작을 위한 방법(2200)의 흐름도이다. 방법(2200)은 전자 디바이스(102) 상의 디코더(2104)의 일부로서의 디코딩된 픽처 버퍼(DPB)(116)에 의해 수행될 수도 있다. 하나의 구성에서, 방법(2200)은 하이브리드 디코딩된 픽처 버퍼(DPB) 동작 모듈(2120)에 의해 수행될 수도 있다. 디코딩된 픽처 버퍼(DPB)(116)는 현재의 픽처의 최초 슬라이스 헤더를 파싱하기 시작할 수도 있다(2202). 디코딩된 픽처 버퍼(DPB)(116)는, 하이브리드 디코딩된 픽처 버퍼(DPB) 동작의 어느 단계들이 픽처 기반일 것이고 어느 단계들이 액세스 유닛(AU) 기반일 것인지 결정할 수도 있다(2204).

[0718] 디코딩된 픽처 버퍼(DPB)(116)는 디코딩된 픽처 버퍼(DPB)(116)로부터의 (출력 없는) 제거를 수행할 수도 있다(2206). 제거는 현재의 픽처의 디코딩 전에 디코딩된 픽처 버퍼(DPB)(116)로부터 픽처들을 제거할 수도 있다. 디코딩된 픽처 버퍼(DPB)(116)는 디코딩된 픽처 버퍼(DPB)(116)로부터의 픽처 출력(범핑)을 수행할 수도 있다(2208). 픽처 출력(범핑)은 코딩된 픽처 버퍼(CPB) 제거 시간에서의 디코딩된 픽처 버퍼(DPB)(116)로부터의 픽처들의 출력을 지칭할 수도 있다. 일부의 구성들에서, 용어 범핑은 디코딩된 픽처 버퍼(DPB)(116)로부터의 하나 이상의 픽처들의 출력을 표시하기 위하여 이용될 수도 있다. 이에 따라, 용어들 범핑 및 출력은 상호 교환 가능하게 이용될 수도 있다.

[0719] 디코딩된 픽처 버퍼(DPB)(116)는 디코딩된 픽처 버퍼(DPB)(116)에서 현재의 픽처를 디코딩 및 저장할 수도 있다(2210). 디코딩된 픽처 버퍼(DPB)(116)는 디코딩된 픽처 버퍼(DPB)(116)에서 저장된 현재의 디코딩된 픽처를 표기할 수도 있다(2212). 예를 들어, 디코딩된 픽처 버퍼(DPB)(116)는 현재의 디코딩된 픽처를 "참조를 위해 이용되지 않음", "참조를 위해 이용됨", "출력을 위해 필요함", 또는 "출력을 위해 필요하지 않음"으로서 표기할 수도 있다(2212). 디코딩된 픽처 버퍼(DPB)(116)는 또한, 디코딩된 픽처 버퍼(DPB)(116)로부터의 또 다른 픽처 출력(추가적인 범핑)을 수행할 수도 있다(2214). 일부의 구성들에서, 디코딩된 픽처 버퍼(DPB)(116)로부터의 픽처들의 반복된 제거/범핑은 어떤 조건들이 만족될 때까지 발생할 수도 있다.

[0720] 도 65는 하이브리드 디코딩된 픽처 버퍼(DPB)(116) 동작을 위한 또 다른 방법(2300)의 흐름도이다. 예를 들어, 도 65의 방법(2300)은 하이브리드 디코딩된 픽처 버퍼(DPB)(116) 동작을 위한 바람직한 방법일 수도 있다. 방법(2300)은 전자 디바이스(102) 상의 디코더(2104)의 일부로서의 디코딩된 픽처 버퍼(DPB)(116)에 의해 수행될 수도 있다. 하나의 구성에서, 방법(2300)은 하이브리드 디코딩된 픽처 버퍼(DPB) 동작 모듈(2120)에 의해 수행될 수도 있다. 디코딩된 픽처 버퍼(DPB)(116)는 현재의 픽처의 최초 슬라이스 헤더를 파싱하기 시작할 수도 있다(2302). 용어 하이브리드는, 디코딩된 픽처 버퍼(DPB)(116) 동작 단계들의 일부가 픽처 기반으로 수행되고 디코딩된 픽처 버퍼(DPB)(116) 동작 단계들의 일부는 액세스 유닛(AU) 기반으로 수행된다는 사실을 지칭한다. 디코딩된 픽처 버퍼(DPB)(116)는 디코딩된 픽처 버퍼(DPB)(116)로부터의 (출력 없는) 픽처 기반 제거를 수행할 수도 있다(2304). 디코딩된 픽처 버퍼(DPB)(116)는 디코딩된 픽처 버퍼(DPB)(116)로부터의 액세스 유닛(AU) 기반 픽처 출력(범핑)을 수행할 수도 있다(2306). 디코딩된 픽처 버퍼(DPB)(116)는 디코딩된 픽처 버퍼(DPB)에서의 현재의 픽처의 픽처 기반 디코딩 및 저장을 수행할 수도 있다(2308).

[0721] 디코딩된 픽처 버퍼(DPB)(116)는 디코딩된 픽처 버퍼(DPB)에서의 현재의 디코딩된 픽처의 픽처 기반 표기를 수행할 수도 있다(310). 디코딩된 픽처 버퍼(DPB)(116)에 의해 수행된 표기 단계는 참조 표기 단계 및 출력 표기 단계를 포함하기 위하여 추가로 재분할될 수도 있다. 본원에서 이용된 바와 같이, "참조를 위해 이용되지 않음" 또는 "참조를 위해 이용됨"으로서 픽처들을 표기하는 것은 참조 표기 단계로서 지칭된다. 디코딩된 픽처 버퍼(DPB)(116)에서의 디코딩된 픽처는 디코딩 프로세스 동작 동안의 임의의 주어진 순간에서 "참조를 위해 이용되지 않음", "단기 참조를 위해 이용됨", 또는 "장기 참조를 위해 이용됨" 중의 하나만으로서 표기될 수 있다. 이 표기들 중의 하나를 픽처에 배정하는 것은 픽처에 배정되는 표기들 중의 또 다른 것을 목적으로 제거한다. 픽처가 "참조를 위해 이용됨"으로서 표기되는 것으로 지칭될 때, 이것은 표기되고 있는 픽처를 "단기 참조를 위해 이용됨" 또는 "장기 참조를 위해 이용됨"의 어느 하나로서 집합적으로 지칭하지만, 양자 모두를 지칭하지는 않는다. 본원에서 이용된 바와 같이, "출력을 위해 필요함" 또는 "출력을 위해 필요하지 않음"으로서 픽처들을 표기하는 것은 출력 표기 단계로서 지칭된다.

[0722] 디코딩된 픽처 버퍼(DPB)(116)는, 참조 표기 단계 및 출력 표기 단계가 픽처 기반 또는 액세스 유닛(AU) 기반의 어느 하나로 발생할 수 있는 그러한 방식으로 동작할 수도 있다. 일반적으로, 픽처 기반으로 또는 액세스 유닛(AU) 기반으로 이 2 개의 표기 단계들을 개별적으로 행하는 것의 모든 가능한 조합들(전형적으로, 4 개의 조합들)이 지원된다. 그러나, 참조 표기 단계는 픽처 기반이고 출력 표기 단계는 액세스 유닛(AU) 기반인 것이 바

람직할 수도 있다.

- [0723] 디코딩된 픽처 버퍼(DPB)(116)는 또한, 현재의 디코딩된 픽처의 액세스 유닛(AU) 기반 출력 표기를 수행할 수도 있다(2312). 디코딩된 픽처 버퍼(DPB)(116)는 디코딩된 픽처 버퍼(DPB)(116)로부터의 또 다른 액세스 유닛(AU) 기반 픽처 출력(추가적인 범핑)을 수행할 수도 있다(2314). 일부의 구성들에서, 디코딩된 픽처 버퍼(DPB)(116)로부터의 픽처들의 반복된 제거/범핑은 어떤 조건들이 만족될 때까지 발생할 수도 있다.
- [0724] 일부의 접근법들에서, 비트스트림 적합성 제약들은 슬라이스 세그먼트 헤더에서의 플래그들에 대해 요구될 수도 있다. 일부의 경우들에는, 제약들이 동일한 액세스 유닛(AU)에 속하는 픽처들에 걸친 플래그들에 대해 적용될 수도 있다. 예를 들어, 플래그들 pic_output_flag 및/또는 no_output_of_prior_pics_flag는 비트스트림 적합성 제약들을 따르도록 요구될 수도 있다. 예를 들어, JCTVC-L1003, JCTVC-M1008 및 JCT3V-D1004는 pic_output_flag 및 no_output_of_prior_pics_flag를 이용한 슬라이스 세그먼트 헤더에서의 시그널링을 각각 설명한다. 또한, PicOutputFlag, NoRaslOutputFlag 및 NoOutputOfPriorPicsFlag와 같은 플래그들은 신택스 엘리먼트들 및 NAL 유닛 타입들에 기반으로 하여 유도가능하다.
- [0725] JCTVC-L1003, JCTVC-M1008, 및 JCT3V-D1004는 또한, HEVC, SHVC 및 MV-HEVC에 대한 DPB의 설명들을 포함한다. JCTVC-M1008 SHVC 초안 텍스트 1은 HEVC의 스케일러블 확장을 위한 텍스트 초안을 제공한다. JCT3V-D1004 MV-HEVC 초안 텍스트 4는 HEVC의 멀티-뷰 확장을 위한 텍스트 초안을 설명한다.
- [0726] JCTVC-L1003, JCTVC-M1008 및/또는 JCT3V-D1004에 따른 비디오 코딩된 비트스트림은, 네트워크 추상화 계층(NAL) 유닛들로서 일반적으로 지칭된 논리적 데이터 패킷들로 배치되는 신택스 구조를 포함할 수도 있다. 각각의 NAL 유닛은 연관된 데이터 페이로드의 목적을 식별하기 위하여, 2-바이트 NAL 유닛 헤더(예컨대, 16 비트들)와 같은 NAL 유닛 헤더를 포함한다. 예를 들어, 각각의 코딩된 슬라이스(및/또는 픽처)는 하나 이상의 슬라이스(및/또는 픽처) NAL 유닛들에서 코딩될 수도 있다. 다른 NAL 유닛들은 예를 들어, 보충 강화 정보, 시간적 서브-계층 액세스(TSA) 픽처의 코딩된 슬라이스, 스텝별 시간적 서브-계층 액세스(STSA) 픽처의 코딩된 슬라이스, 비-TSA의 코딩된 슬라이스, 비-STSA 후미 픽처, 파손 링크 액세스 픽처의 코딩된 슬라이스, 순간적 디코딩된 리프레시 픽처의 코딩된 슬라이스, 클린 랜덤 액세스 픽처의 코딩된 슬라이스, 디코딩가능한 선두 픽처의 코딩된 슬라이스, 폐기 태그된 픽처의 코딩된 슬라이스, 비디오 파라미터 세트, 시퀀스 파라미터 세트, 픽처 파라미터 세트, 액세스 유닛 구분자, 시퀀스 종료, 비트스트림 종료, 필러 데이터, 및/또는 시퀀스 강화 정보 메시지와 같이, 데이터의 다른 카테고리들에 대해 포함될 수도 있다. 이하의 표 (7)은 NAL 유닛 코드들 및 NAL 유닛 타입 클래스들의 하나의 예를 예시한다. 다른 NAL 유닛 타입들이 또한, 희망하는 바와 같이 포함될 수도 있다.
- [0727] 또한, 표 (7)에서 도시된 NAL 유닛들에 대한 NAL 유닛 타입 값들은 재배치될 수도 있고 재배정될 수도 있다는 것을 이해해야 한다. 또한, 추가적인 NAL 유닛 타입들이 추가될 수도 있거나 제거될 수도 있다.
- [0728] 인트라 랜덤 액세스 포인트(intra random access point; IRAP) 픽처는, 각각의 비디오 코딩 계층 NAL 유닛이 이하의 표 (7)에서 도시된 바와 같이, BLA_W_LP 내지 RSV_IRAP_VCL23까지의 범위에서 nal_unit_type을 가지는 코딩된 픽처이다. IRAP 픽처는 인트라 코딩된 (I) 슬라이스들만을 포함한다.
- [0729] 순간적 디코딩 리프레시(instantaneous decoding refresh; IDR) 픽처는, 각각의 비디오 코딩 계층 NAL 유닛이 표 (7)에서 도시된 바와 같이, IDR_W_RADL 또는 IDR_N_LP와 동일한 nal_unit_type을 가지는 IRAP 픽처이다. 순간적 디코딩 리프레시(IDR) 픽처는 I 슬라이스들만을 포함하고, 디코딩 순서에서 비트스트림에서의 최초 픽처일 수도 있거나 비트스트림에서 더 이후에 나타날 수도 있다.
- [0730] 각각의 IDR 픽처는 디코딩 순서에서 코딩된 비디오 시퀀스(CVS)의 최초 픽처이다. 파손 링크 액세스(BLA) 픽처는, 각각의 비디오 코딩 계층 NAL 유닛이 표 (7)에서 도시된 바와 같이, BLA_W_LP, BLA_W_RADL, 또는 BLA_N_LP와 동일한 nal_unit_type을 가지는 IRAP 픽처이다.
- [0731] BLA 픽처는 I 슬라이스들만을 포함하고, 디코딩 순서에서 비트스트림에서의 최초 픽처일 수도 있거나 비트스트림에서 더 이후에 나타날 수도 있다. 각각의 BLA 픽처는 새로운 코딩된 비디오 시퀀스를 시작하고, 디코딩 프로세스에 대해 IDR 픽처와 동일한 효과를 가진다. 그러나, BLA 픽처는 비워지지 않은 참조 픽처 세트를 특정하는 신택스 엘리먼트들을 포함한다.

표 42

nal_unit_type	nal_unit_type의 명칭	NAL 유닛 및 원시 바이트 시퀀스 페이로드(RBSP) 선택스 구조의 콘텐츠	NAL 유닛 타입 클래스
0 1	TRAIL_N TRAIL_R	비-TSA, 비-STSA 후미 픽처의 코딩된 슬라이스 세그먼트 slice_segment_laver_rbsp()	비디오 코딩 계층 (VCL)
2 3	TSA_N TSA_R	시간적 서브-계층 액세스(TSA) 픽처의 코딩된 슬라이스 세그먼트 slice_segment_laver_rbsp()	VCL
4 5	STSA_N STSA_R	스텝별 시간적 서브-계층 액세스(STSA) 픽처의 코딩된 슬라이스 세그먼트 slice_segment_laver_rbsp()	VCL
6 7	RADL_N RADL_R	랜덤 액세스 디코딩가능한 선두(RADL) 픽처의 코딩된 슬라이스 세그먼트 slice_segment_laver_rbsp()	VCL
8 9	RASL_N RASL_R	랜덤 액세스 스킵된 선두(RASL) 픽처의 코딩된 슬라이스 세그먼트 slice_segment_laver_rbsp()	VCL
10 12 14	RSV_VCL_N10 RSV_VCL_N12 RSV_VCL_N14	예약된 비-IRAP 서브-계층 비-참조 VCL NAL 유닛 타입들	VCL
11 13 15	RSV_VCL_R11 RSV_VCL_R13 RSV_VCL_R15	예약된 비-IRAP 서브-계층 참조 VCL NAL 유닛 타입들	VCL
16 17 18	BLA_W_LP BLA_W_RADL BLA_N_LP	파손 링크 액세스(BLA) 픽처의 코딩된 슬라이스 세그먼트 slice_segment_laver_rbsp()	VCL
19 20	IDR_W_RADL IDR_N_LP	순간적 디코딩 리프레시(IDR) 픽처의 코딩된 슬라이스 세그먼트 slice_segment_laver_rbsp()	VCL
21	CRA_NUT	클린 랜덤 액세스(CRA) 픽처의 코딩된 슬라이스 세그먼트 slice_segment_laver_rbsp()	VCL
22 23	RSV_IRAP_VCL22 RSV_IRAP_VCL23	예약된 IRAP VCL NAL 유닛 타입들	VCL
24..31	RSV_VCL24.. RSV_VCL31	예약된 비-IRAP VCL NAL 유닛 타입들	VCL
32	VPS_NUT	비디오 파라미터 세트 video_parameter_set_rbsp()	비-비디오 코딩 계층 (비-VCL)
33	SPS_NUT	시퀀스 파라미터 세트 seq_parameter_set_rbsp()	비-VCL
34	PPS_NUT	픽처 파라미터 세트 pic_parameter_set_rbsp()	비-VCL
35	AUD_NUT	액세스 유닛 구분자 access_unit_delimiter_rbsp()	비-VCL
36	EOS_NUT	시퀀스 종료 end_of_seq_rbsp()	비-VCL
37	EOB_NUT	비트스트림 종료 end_of_bitstream_rbsp()	비-VCL
38	FD_NUT	필러 데이터 filler_data_rbsp()	비-VCL
39 40	PREFIX_SEI_NUT SUFFIX_SEI_NUT	보충 강화 정보 sei_rbsp()	비-VCL
41..47	RSV_NVCL41.. RSV_NVCL47	예약됨	비-VCL
48..63	UNSPEC48.. UNSPEC63	미특정	비-VCL

표 (7)

[0732]

[0733]

이하의 표 (8)를 참조하면, NAL 유닛 헤더 선택스는 데이터의 2 바이트들, 즉, 16 비트들을 포함할 수도 있다. 최초 비트는 NAL 유닛의 시작 시에 항상 제로로 설정되는 "forbidden_zero_bit"일 수도 있다. 다음 6 비트들은, 상기 표 (7)에서 도시된 바와 같이 NAL 유닛 내에 포함된 원시 바이트 시퀀스 페이로드들("RBSP") 데이터 구조의 타입을 특정하는 "nal_unit_type"일 수도 있다. 다음 6 비트들은 계층의 식별자를 특정하는 "nuh_layer_id"일 수도 있다. 일부의 경우들에는, 이 6 비트들이 그 대신에 "nuh_reserved_zero_6bits"로서 특정될 수도 있다. nuh_reserved_zero_6bits는 표준의 기본 사양에서 0과 동일할 수도 있다. 스케일러블 비디오 코딩 및/또는 선택스 확장들에서, nuh_layer_id는 이 특별한 NAL 유닛이 이 6 비트들의 값에 의해 식별된 계층에 속하는 것으로 특정할 수도 있다.

[0734]

다음의 선택스 엘리먼트는 "nuh_temporal_id_plus1"일 수도 있다. nuh_temporal_id_plus1 마이너스 1은 NAL 유닛에 대한 시간적 식별자를 특정할 수도 있다. 변수 시간적 식별자 TemporalId는 TemporalId = nuh_temporal_id_plus1 - 1로서 특정될 수도 있다. 시간적 식별자 TemporalId는 시간적 서브-계층을 식별하기

위하여 이용된다. 변수 HighestTid는 디코딩되어야 할 최고 시간적 서브-계층을 식별한다.

표 43

nal_unit_header() {	디스크립터
forbidden_zero_bit	f(1)
nal_unit_type	u(6)
nuh_layer_id	u(6)
nuh_temporal_id_plus1	u(3)
}	

표 (8)

[0735]

[0736]

이하의 표 (9)는 예시적인 시퀀스 파라미터 세트(SPS) 선택스 구조를 도시한다. pic_width_in_luma_samples는 루마 샘플들의 단위로 각각의 디코딩된 픽처의 폭을 특징한다. pic_width_in_luma_samples는 0과 동일하지 않을 것이다. pic_height_in_luma_samples는 루마 샘플들의 단위로 각각의 디코딩된 픽처의 높이를 특징한다. pic_height_in_luma_samples는 0과 동일하지 않을 것이다.

[0737]

sps_max_sub_layers_minus1 플러스 1은 SPS를 참조하는 각각의 CVS에서 존재할 수도 있는 시간적 서브-계층들의 최대 수를 특징한다. sps_max_sub_layers_minus1의 값은 0 내지 6까지의 범위이다.

[0738]

1과 동일한 sps_sub_layer_ordering_info_present_flag 플래그는, sps_max_dec_pic_buffering_minus1[i], sps_max_num_reorder_pics[i], 및 sps_max_latency_increase_plus1[i] 선택스 엘리먼트들이 sps_max_sub_layers_minus1 + 1 서브-계층들에 대해 존재하는 것으로 특징한다. 0과 동일한 sps_sub_layer_ordering_info_present_flag는, sps_max_dec_pic_buffering_minus1[sps_max_sub_layers_minus1], sps_max_num_reorder_pics[sps_max_sub_layers_minus1], 및 sps_max_latency_increase_plus1[sps_max_sub_layers_minus1]의 값들이 모든 서브-계층들에 적용되는 것으로 특징한다.

[0739]

sps_max_dec_pic_buffering_minus1[i] 플러스 1은, HighestTid가 i와 동일할 때에 픽처 저장 버퍼들의 단위로 CVS에 대한 디코딩된 픽처 버퍼의 최대 요구된 사이즈를 특징한다. sps_max_dec_pic_buffering_minus1[i]의 값은 0 내지 MaxDpbSize - 1까지의 범위이고, 여기서, MaxDpbSize는 픽처 저장 버퍼들의 단위로 최대 디코딩된 픽처 버퍼 사이즈를 특징한다. i가 0보다 더 클 때, sps_max_dec_pic_buffering_minus1[i]는 sps_max_dec_pic_buffering_minus1[i - 1] 이상일 것이다. sps_sub_layer_ordering_info_present_flag가 0과 동일한 것으로 인해, sps_max_dec_pic_buffering_minus1[i]가 0 내지 sps_max_sub_layers_minus1 - 1까지의 범위에서 i에 대해 존재하지 않을 때, 그것은 sps_max_dec_pic_buffering_minus1[sps_max_sub_layers_minus1]과 동일한 것으로 추론된다.

[0740]

sps_max_num_reorder_pics[i]는, HighestTid가 i와 동일할 때, 디코딩 순서에서 CVS에서의 임의의 픽처를 선행할 수 있으며 출력 순서에서 그 픽처를 후행할 수 있는 픽처들의 최대 허용된 수를 표시한다. sps_max_num_reorder_pics[i]의 값은 0 내지 sps_max_dec_pic_buffering_minus1[i]까지의 범위이다. i가 0보다 더 클 때, sps_max_dec_pic_buffering_minus1[i]는 sps_max_dec_pic_buffering_minus1[i - 1] 이상일 수도 있다. sps_max_num_reorder_pics[i]가 0 내지 sps_max_sub_layers_minus1 - 1까지의 범위에서 i에 대해 존재하지 않을 때, sps_sub_layer_ordering_info_present_flag가 0과 동일한 것으로 인해, 그것은 sps_max_num_reorder_pics[sps_max_sub_layers_minus1]과 동일한 것으로 추론된다.

[0741]

0과 동일하지 않은 sps_max_latency_increase_plus1[i]는, HighestTid가 i와 동일할 때, 출력 순서에서 CVS에서의 임의의 픽처를 선행할 수 있으며 디코딩 순서에서 그 픽처를 후행할 수 있는 픽처들의 최대 수를 특징하는 SpsMaxLatencyPictures[i]의 값을 연산하기 위하여 이용될 수도 있다. sps_max_latency_increase_plus1[i]가 0과 동일하지 않을 때, SpsMaxLatencyPictures[i]의 값은 SpsMaxLatencyPictures[i] = sps_max_num_reorder_pics[i] + sps_max_latency_increase_plus1[i] - 1로서 특징된다. sps_max_latency_increase_plus1[i]가 0과 동일할 때, 대응하는 한계가 표현되지 않는다.

[0742]

sps_max_latency_increase_plus1[i]의 값은 0 내지 $2^{32} - 2$ 까지의 범위이다.

sps_max_latency_increase_plus1[i]가 0 내지 sps_max_sub_layers_minus1 - 1까지의 범위에서 i에 대해 존재하지 않을 때, sps_sub_layer_ordering_info_present_flag가 0과 동일한 것으로 인해, 그것은 sps_max_latency_increase_plus1[sps_max_sub_layers_minus1]과 동일한 것으로 추론된다.

표 44

seq_parameter_set_rbsp() {
...
sps_max_sub_layers_minus1
...
pic_width_in_luma_samples
pic_height_in_luma_samples
...
for(i = (sps_sub_layer_ordering_info_present_flag ? 0 :
sps_max_sub_layers_minus1);
I <= sps_max_sub_layers_minus1; i++) {
sps_max_dec_pic_buffering_minus1[i]
sps_max_num_reorder_pics[i]
sps_max_latency_increase_plus1[i]
}
...
}

표 (9)

[0743]

[0744]

또한, JCTVC-L1003은 HEVC 표준을 설명한다. 예를 들어, pic_out_flag 및 no_output_of_prior_pics_flag에 관한 세부사항은 이하의 표 (10)에서 제공된다:

표 45

일반적인 슬라이스 세그먼트 헤더 선택스
slice_segment_header() {
first_slice_segment_in_pic_flag
if(nal_unit_type >= BLA_W_LP && nal_unit_type <= RSV_IRAP_VCL23)
no_output_of_prior_pics_flag
slice_pic_parameter_set_id
if(!first_slice_segment_in_pic_flag) {
if(dependent_slice_segments_enabled_flag)
dependent_slice_segment_flag
slice_segment_address
}
if(!dependent_slice_segment_flag) {
for(i = 0; i < num_extra_slice_header_bits; i++)
slice_reserved_flag[i]
slice_type
if(output_flag_present_flag)
pic_output_flag
...
...
}

표 (10)

[0745]

[0746]

표 (10)에서, no_output_of_prior_pics_flag는 JCTVC-L1003의 부록 C에서 특정된 바와 같은 비트스트림에서의 최초 픽처가 아닌 IDR 또는 BLA 픽처의 디코딩 후에, 디코딩된 픽처 버퍼(DPB)에서의 이전-디코딩된 픽처들의 출력에 영향을 준다.

[0747]

1과 동일한 output_flag_present_flag는, pic_output_flag 선택스 엘리먼트가 연관된 슬라이스 헤더들에서 존재함을 표시한다. 0과 동일한 output_flag_present_flag는, pic_output_flag 선택스 엘리먼트가 연관된 슬라이스 헤더들에서 존재하지 않음을 표시한다. pic_output_flag는 JCTVC-L 1003의 부록 C에서 특정된 바와 같이 디코딩된 픽처 출력 및 제거 프로세스들에 영향을 준다. pic_output_flag가 존재하지 않을 때, 그것은 1과 동일한 것으로 추론된다.

- [0748] (JCTVC-L1003의 8.1에서 발견된 바와 같은) 일반적인 디코딩 프로세스에 대하여, PicOutputFlag는 다음과 같이 설정된다:
- [0749] - 현재의 픽처가 RASL 픽처이고 연관된 IRAP 픽처의 NoRaslOutputFlag가 1과 동일할 경우, PicOutputFlag는 0과 동일하게 설정될 수도 있다.
- [0750] - 이와 다를 경우, PicOutputFlag는 pic_output_flag와 동일하게 설정될 수도 있다.
- [0751] 추가적으로, (JCTVC-L1003의 섹션 8.3.3.1에서 발견된 바와 같은) 이용불가능한 참조 픽처들을 생성하기 위한 일반적인 디코딩 프로세스 동안, 생성된 픽처를 위한 PicOutputFlag의 값은 어떤 조건들 하에서 0과 동일하게 설정될 수도 있다.
- [0752] 현재의 픽처가 IRAP 픽처일 때, 다음이 적용된다:
- [0753] - 현재의 픽처가 IDR 픽처, BLA 픽처, 디코딩 순서에서 비트스트림에서의 최초 픽처, 또는 디코딩 순서에서 시퀀스 NAL 유닛의 종점을 후행하는 최초 픽처일 경우, 변수 NoRaslOutputFlag가 1과 동일하게 설정될 수도 있다.
- [0754] - 이와 다르게, 변수 HandleCraAsBlaFlag를 현재의 픽처에 대한 값으로 설정하기 위하여 JCTVC-L1003에서 특정되지 않은 일부의 외부 수단이 이용가능할 경우, 변수 HandleCraAsBlaFlag는 외부 수단에 의해 제공된 값과 동일하게 설정될 수도 있고, 변수 NoRaslOutputFlag는 HandleCraAsBlaFlag와 동일하게 설정될 수도 있다.
- [0755] - 이와 다를 경우, 변수 HandleCraAsBlaFlag는 0과 동일하게 설정될 수도 있고, 변수 NoRaslOutputFlag는 0과 동일하게 설정될 수도 있다.
- [0756] 위에서 설명된 바와 같이, 액세스 유닛(AU)은, 특정된 분류 규칙에 따라 서로 연관되고, 디코딩 순서에서 연속적이고, 동일한 출력 시간 및 그 연관된 비-VCL NAL 유닛들과 연관된 모든 코딩된 픽처들의 비디오 코딩 계층(VCL) NAL 유닛들을 포함하는 네트워크 추상화 계층(NAL) 유닛들의 세트를 지칭한다. 기본 계층은 모든 VCL NAL 유닛들이 0과 동일한 nuh_layer_id를 가지는 계층이다. 코딩된 픽처는, nuh_layer_id의 특별한 값을 갖는 VCL NAL 유닛들을 포함하며 픽처의 모든 코딩 트리 유닛들을 포함하는 픽처의 코딩된 표현이다. 일부의 경우에는, 코딩된 픽처가 계층 컴포넌트로 칭해질 수도 있다. 픽처 기반 또는 액세스 유닛(AU) 기반인 단계들에 대한 추가적인 세부사항들은 이하의 도 69 및 도 70에 관련하여 주어진다.
- [0757] 일부의 구성들에서, pic_output_flag 및/또는 no_output_of_prior_pics_flag에 대한 비트스트림 적합성 제약들은 액세스 유닛(AU)에서의 코딩된 픽처들에 대해 이용될 수도 있다. 게다가, 3 개의 새로운 액세스 유닛 출력 플래그들, AU 출력 플래그(예컨대, AuOutputFlag), AU RASL 출력 없음 플래그(예컨대, AuNoRaslOutputFlag) 및 AU 이전 픽처들의 출력 없음 플래그(예컨대, AuNoOutputOfPriorPicsFlag)는 AU에서의 코딩된 픽처들에 대한 다양한 선택스 엘리먼트들 및 NAL 유닛 타입들의 값에 기반으로 하여 AU에 대해 유도될 수도 있다. 일부의 구성들에서, 픽처들의 출력 및 제거는 SHVC 및 멀티-뷰 HEVC에 대한 이 3 개의 플래그들(예컨대, AuOutputFlag, AuNoRaslOutputFlag 및 AuNoOutputOfPriorPicsFlag)에 기반으로 할 수도 있다.
- [0758] 예를 들어, HEVC-확장들에 대한 비트스트림 적합성 제약들은 본원에서의 시스템들 및 방법들에서 설명된 바와 같이 다르게 될 수도 있다. 특히, 비트스트림 적합성 제약들은 SHVC 비트스트림에 적용될 수도 있다. 게다가, 비트스트림 적합성 제약들은 MV-HEVC 비트스트림에 적용될 수도 있다.
- [0759] 하나의 구성에서, 존재할 때, 슬라이스 세그먼트 헤더 선택스 엘리먼트들 pic_output_flag의 값은 액세스 유닛(AU)에서의 코딩된 픽처들의 모든 슬라이스 세그먼트 헤더들에서 동일하도록 요구될 수도 있다. 또 다른 구성에서, 존재할 때, 슬라이스 세그먼트 헤더 선택스 엘리먼트들 pic_output_flag의 값은, 코딩된 픽처들이 동일한 NAL 유닛 타입을 가질 때에 액세스 유닛(AU)에서의 코딩된 픽처들의 모든 슬라이스 세그먼트 헤더들에서 동일할 수도 있다.
- [0760] 하나의 구성에서, 존재할 때, 타겟 계층의 nuh_layer_id 값과 동일한 nuh_layer_id를 갖는 슬라이스 세그먼트들에 대한 슬라이스 세그먼트 헤더 선택스 엘리먼트들 pic_output_flag의 값은 액세스 유닛(AU)에서의 이러한 코딩된 픽처들의 모든 슬라이스 세그먼트 헤더들에서 동일할 수도 있다. 또 다른 구성에서, 존재할 때, 타겟 계층의 nuh_layer_id 값과 동일하지 않은 nuh_layer_id를 갖는 슬라이스 세그먼트들에 대한 슬라이스 세그먼트 헤더 선택스 엘리먼트들 pic_output_flag의 값은 액세스 유닛(AU)에서의 이러한 코딩된 픽처들의 모든 슬라이스 세그먼트 헤더들에서 0일 수도 있다.
- [0761] 하나의 구성에서, 타겟 계층은 JCTVC-L1003, JCTVC-M1008, 또는 JCT3V-D1004에서 정의된 바와 같이 계층 세트

또는 타겟 계층 세트 또는 출력 계층 세트에 속하는 계층일 수도 있다. 또 다른 구성에서, 타겟 계층은 디코딩 되도록 의도되는 계층일 수도 있다. 또 다른 구성에서, 타겟 계층은 디코딩 및 출력(디스플레이되거나, 또는 출력을 위해 전송됨)되도록 의도되는 계층일 수도 있다.

- [0762] 일부의 구성에서, 존재할 때, 슬라이스 세그먼트 헤더 신택스 엘리먼트들 no_output_of_prior_pics_flag의 값은 액세스 유닛(AU)에서의 코딩된 픽처들의 모든 슬라이스 세그먼트 헤더들에서 동일할 수도 있다. 다른 구성들에서, 존재할 때, 슬라이스 세그먼트 헤더 신택스 엘리먼트들 no_output_of_prior_pics_flag의 값은, 코딩된 픽처들이 동일한 NAL 유닛 타입을 가질 때에 액세스 유닛(AU)에서의 코딩된 픽처들의 모든 슬라이스 세그먼트 헤더들에서 동일할 수도 있다. 하나의 구성에서, 존재할 때, 타겟 계층의 nuh_layer_id 값과 동일한 nuh_layer_id를 갖는 슬라이스 세그먼트들에 대한 슬라이스 세그먼트 헤더 신택스 엘리먼트들 no_output_of_prior_pics_flag의 값은 액세스 유닛(AU)에서의 이러한 코딩된 픽처들의 모든 슬라이스 세그먼트 헤더들에서 동일할 수도 있다.
- [0763] 일부의 구성에서, 신택스 엘리먼트들 pic_output_flag 및/또는 no_output_of_prior_pics_flag는 nuh_layer_id > 0일 때에 시그널링되지 않을 수도 있다. 이 경우, nuh_layer_id > 0인 계층들에 대한 값들은 0과 동일한 nuh_layer_id에 대한 그 시그널링된 값들과 동일한 것으로 추론될 수도 있다.
- [0764] 일부의 구성들에서는, AuOutputFlag 및 AuNoRas1OutputFlag와 같은 추가적인 플래그들이 채용될 수도 있다. 플래그들 AuOutputFlag 및 AuNoRas1OutputFlag는 다수의 접근법들에 따라 유도될 수도 있다. 하나의 접근법 또는 구성에서, 2 개의 플래그들, AuOutputFlag 및 AuNoRas1OutputFlag는 DPB 동작을 위해 유도 및 이용될 수도 있다. PicOutputFlag가 AU에서의 모든 픽처들에 대해 1과 동일할 경우, AuOutputFlag는 1과 동일하게 설정될 수도 있다. 이와 다를 경우, AuOutputFlag는 0과 동일하게 설정될 수도 있다.
- [0765] 또 다른 접근법에서는, PicOutputFlag가 AU에서의 적어도 하나의 픽처에 대해 1과 동일할 경우, AuOutputFlag는 1과 동일하게 설정될 수도 있다. 이와 다를 경우, AuOutputFlag는 0과 동일하게 설정될 수도 있다. 이에 따라, 이 경우, PicOutputFlag가 AU에서의 모든 픽처들에 대해 0과 동일할 경우에는, AuOutputFlag가 0과 동일하게 설정될 수도 있다.
- [0766] 또 다른 접근법에서는, PicOutputFlag가 AU에서의 모든 타겟 출력 계층들에 속하는 픽처들에 대해 1과 동일할 경우, AuOutputFlag는 1과 동일하게 설정될 수도 있다. 이와 다를 경우, AuOutputFlag는 0과 동일하게 설정될 수도 있다.
- [0767] 더욱 또 다른 접근법에서는, PicOutputFlag가 AU에서의 적어도 하나의 타겟 출력 계층들에 속하는 픽처에 대해 1과 동일할 경우, AuOutputFlag는 1과 동일하게 설정될 수도 있다. 이와 다를 경우, AuOutputFlag는 0과 동일하게 설정될 수도 있다.
- [0768] 하나의 접근법에서, NoRas1OutputFlag가 AU에서의 모든 픽처들에 대해 1과 동일할 경우, AuNoRas1OutputFlag는 1과 동일하게 설정될 수도 있다. 이와 다를 경우, AuNoRas1OutputFlag는 0과 동일하게 설정될 수도 있다.
- [0769] 또 다른 접근법에서는, NoRas1OutputFlag가 AU에서의 적어도 하나의 픽처에 대해 1과 동일할 경우, AuNoRas1OutputFlag는 1과 동일하게 설정될 수도 있다. 이와 다를 경우, AuNoRas1OutputFlag는 0과 동일하게 설정될 수도 있다. 이에 따라, 이 경우, NoRas1OutputFlag가 AU에서의 모든 픽처들에 대해 0과 동일할 경우에는, AuNoRas1OutputFlag가 0과 동일하게 설정될 수도 있다.
- [0770] 또 다른 접근법에서는, NoRas1OutputFlag가 AU에서의 모든 타겟 출력 계층들에 속하는 픽처들에 대해 1과 동일할 경우, AuNoRas1OutputFlag는 1과 동일하게 설정될 수도 있다. 이와 다를 경우, AuNoRas1OutputFlag는 0과 동일하게 설정될 수도 있다.
- [0771] 더욱 또 다른 접근법에서는, NoRas1OutputFlag가 AU에서의 적어도 하나의 타겟 출력 계층들에 속하는 픽처에 대해 1과 동일할 경우, AuNoRas1OutputFlag는 1과 동일하게 설정될 수도 있다. 이와 다를 경우, AuNoRas1OutputFlag는 0과 동일하게 설정될 수도 있다.
- [0772] 상기 접근법들 및 구성들의 일부에서, DPB 동작은 PicOutputFlag 대신에 AuOutputFlag를 이용할 수도 있다. 추가적으로, DPB 동작은 NoRas1OutputFlag 대신에 AuNoRas1OutputFlag를 이용할 수도 있다.
- [0773] 본 시스템들 및 방법들에 따라, AuOutputFlag 및 AuNoRas1OutputFlag와 같은 AU 출력 플래그들의 이용을 보여주는 예들은 이하의 리스트 (1A) 및 리스트 (2A)에서 이하에서 제공된다. 또한, 리스트 (1), 리스트 (1A), 리스트 (2) 및 리스트 (2A)에서 이하에서 설명된 바와 같이, AU 출력 플래그 AuNoOutputOfPriorPicsFlag는 DPB

동작에 대해 유도 및 이용될 수도 있다.

- [0774] 본 시스템들 및 방법들은 표준들 문서들에 대한 변경들에 의해 구현될 수도 있다. 이하의 리스트 (1)은 본 시스템들 및 방법들을 수용하기 위하여 변경될 JCTVC-L1003의 섹션들을 제공한다.
- [0775] **리스트 1**
- [0776] C.3 디코딩된 픽처 버퍼(DPB)의 동작
- [0777] C.3.1 일반
- [0778] 이 하위조항에서의 사양들은 하위조항 C.1 에서 특정된 바와 같이 선택된 DPB 파라미터들의 각각의 세트에 독립적으로 적용된다. DPB는 각각의 계에 대해 별도로 또는 독립적으로 동작한다. 이에 따라, 다음의 단계들은 nuh_layer_id의 특별한 값을 갖는 각각의 디코딩된 픽처에 대해 독립적으로 발생한다.
- [0779] 디코딩된 픽처 버퍼는 픽처 저장 버퍼들을 포함한다. 각각의 계층은 픽처 저장 버퍼들의 그 자신의 세트로 구성된다. 이에 따라, 각각의 계층의 픽처 저장 버퍼들은 계층의 nuh_layer_id 값과 연관된다. 픽처 저장 버퍼들의 각각은 "참조를 위해 이용됨"으로서 표기되거나 미래의 출력을 위해 유지되는 디코딩된 픽처를 포함할 수도 있다. 하위조항들 C.3.2, C.3.3 및 C.3.4에서 특정된 프로세스들은 이하에서 특정된 바와 같이 순차적으로 적용된다.
- [0780] C.3.2 DPB로부터의 픽처들의 제거
- [0781] 현재의 픽처의 디코딩 전의 (그러나, 현재의 픽처의 최초 슬라이스의 슬라이스 헤더를 파싱한 후의) DPB 로부터의 픽처들의 제거는 (현재의 픽처를 포함하는) 액세스 유닛 n에 속하는 현재의 픽처의 최초 디코딩 유닛의 CPB 제거 시간에 순간적으로 발생하고 다음과 같이 진행된다:
- [0782] - 하위조항 8.3.2에서 특정된 바와 같은 RPS에 대한 디코딩 프로세스가 호출된다.
- [0783] - 현재의 픽처가 픽처 0이 아닌, 1과 동일한 NoRaslOutputFlag를 갖는 IRAP 픽처일 때, 다음의 순서화된 단계들이 적용된다: 1. 변수 NoOutputOfPriorPicsFlag는 다음과 같이 테스트 중인 디코더에 대해 유도된다:
- [0784] - - 현재의 픽처가 CRA 픽처일 경우, NoOutputOfPriorPicsFlag는 (no_output_of_prior_pics_flag의 값에 관계없이) 1과 동일하게 설정된다.
- [0785] - 이와 다르게, 현재의 픽처의 nuh_layer_id 값에 대응하는 활성 SPS로부터 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid]의 값이 현재의 픽처의 nuh_layer_id 값과 동일한 nuh_layer_id 값을 갖는 선행 픽처에 대해 활성인 SPS로부터 각각 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid]의 값과는 상이할 경우, NoOutputOfPriorPicsFlag는 no_output_of_prior_pics_flag의 값에 관계 없이, 테스트 중인 디코더에 의해 1로 설정될 수도 있다(그러나, 설정되지 않아야 함).
- [0786] 주목 - NoOutputOfPriorPicsFlag를 no_output_of_prior_pics_flag와 동일하게 설정하는 것이 이 조건들 하에서 선호되지만, 테스트 중인 디코더는 이 경우에 NoOutputOfPriorPicsFlag를 1로 설정하도록 허용된다.
- [0787] - 이와 다를 경우, NoOutputOfPriorPicsFlag는 no_output_of_prior_pics_flag와 동일하게 설정된다.
- [0788] 2. 테스트 중인 디코더에 대해 유도된 NoOutputOfPriorPicsFlag의 값은 HRD에 대해 적용되어, NoOutputOfPriorPicsFlag의 값이 1과 동일할 때, DPB에서의 현재의 픽처의 nuh_layer_id 값에 대응하는 모든 픽처 저장 버퍼들은 이들이 포함하는 픽처들의 출력 없이 비워지고, 현재의 픽처의 nuh_layer_id 값에 대한 DPB 층만도는 0과 동일하게 설정된다.
- [0789] 하나의 실시형태에서, 테스트 중인 디코더에 대해 유도된 NoOutputOfPriorPicsFlag의 값은 HRD에 대해 적용되어, NoOutputOfPriorPicsFlag의 값이 1과 동일할 때, DPB에서의 모든 nuh_layer_id 값들에 대응하는 모든 픽처 저장 버퍼들은 이들이 포함하는 픽처들의 출력 없이 비워지고, 모든 nuh_layer_id 값들에 대한 DPB 층만도는 0과 동일하게 설정된다.
- [0790] 하나의 실시형태에서, 테스트 중인 디코더에 대해 유도된 NoOutputOfPriorPicsFlag의 값은 HRD에 대해 적용되어, NoOutputOfPriorPicsFlag의 값이 1과 동일할 때, DPB에서의 현재의 픽처의 nuh_layer_id 값과 동일한 currLayerId를 갖는 모든 픽처 저장 버퍼들 PSB[currLayerId]는 이들이 포함하는 픽처들의 출력 없이 비워지

고, 현재의 픽처의 nuh_layer_id 값 currLayerId에 대한 DPB 충만도 DPBFullness[currLayerId]는 0과 동일하게 설정된다.

[0791] 하나의 실시형태에서, 테스트 중인 디코더에 대해 유도된 NoOutputOfPriorPicsFlag의 값은 HRD에 대해 적용되어, NoOutputOfPriorPicsFlag의 값이 1과 동일할 때, DPB에서의 모든 nuh_layer_id 값들에 대한 모든 픽처 저장 버퍼들 PSB[nuh_layer_id]은 이들이 포함하는 픽처들의 출력 없이 비워지고, 모든 nuh_layer_id 값들에 대한 DPB 충만도 DPBFullness[nuh_layer_id]는 0과 동일하게 설정된다.

[0792] - 다음의 조건들의 양자가 DPB에서의 현재의 픽처의 nuh_layer_id 값에 대응하는 픽처 저장 버퍼에서의 임의의 픽처들 k에 대해 참일 때, DPB에서의 모든 이러한 픽처들 k는 DPB로부터 제거된다:

[0793] - 픽처 k 는 "참조를 위해 이용되지 않음" 으로서 표기됨

[0794] - 픽처 k 는 0과 동일한 PicOutputFlag를 가지거나, 그 DPB 출력 시간은 현재의 픽처 n의 최초 디코딩 유닛(디코딩 유닛 m 으로서 나타냄)의 CPB 제거 시간 이하임; 즉, DpbOutputTime[k]는 CpbRemovalTime(m) 이하임

[0795] - DPB로부터 제거되는 각각의 픽처에 대하여, DPB 충만도는 1씩 감소됨.

[0796] - 하나의 실시형태에서, DPB로부터 제거되는 nuh_layer_id 값 nuhLayerIdk를 가지는 각각의 픽처 k에 대하여, DPB 충만도 DPBfulness[nuhLayerIdk]는 1씩 감소됨

[0797] C.3.3 픽처 출력

[0798] 이 하위조항에서 특정된 프로세스들은 액세스 유닛 n의 CPB 제거 시간, AuCpbRemovalTime[n]에서 순간적으로 발생한다.

[0799] 픽처 n이 1과 동일한 PicOutputFlag를 가질 때, 그 DPB 출력 시간 DpbOutputTime[n]은 다음과 같이 유도되고, 여기서, 변수 firstPicInBufferingPeriodFlag는 액세스 유닛 n이 버퍼링 주기의 최초 액세스 유닛일 경우에는 1, 그리고 이와 다를 경우에는 0과 동일하다:

```

if( !SubPicHrdFlag ) {
    DpbOutputTime[ n ] = AuCpbRemovalTime[ n ] + ClockTick * picDpbOutputDelay (C-16)
    if( firstPicInBufferingPeriodFlag )
        DpbOutputTime[ n ] -= ClockTick * DpbDelayOffset
} else
    DpbOutputTime[ n ] = AuCpbRemovalTime[ n ] + ClockSubTick *
picSptDpbOutputDuDelay
    
```

[0800]

[0801] 여기서, picDpbOutputDelay는 액세스 유닛 n과 연관된 픽처 타이밍 SEI 메시지에서의 pic_dpb_output_delay의 값이고, picSptDpbOutputDuDelay는, 존재한다면, 액세스 유닛 n과 연관된 디코딩 유닛 정보 SEI 메시지들에서의 pic_spt_dpb_output_du_delay의 값, 또는 액세스 유닛 n과 연관된 디코딩 유닛 정보 SEI 메시지가 없거나 액세스 유닛 n과 연관된 디코딩 유닛 정보 SEI 메시지가 존재하는 pic_spt_dpb_output_du_delay를 가지지 않을 때에는, 액세스 유닛 n과 연관된 픽처 타이밍 SEI 메시지에서의 pic_dpb_output_du_delay의 값이다.

[0802] 주목 - 신택스 엘리먼트 pic_spt_dpb_output_du_delay가 액세스 유닛 n과 연관된 임의의 디코딩 유닛 정보 SEI 메시지에서 존재하지 않을 때, 값은 액세스 유닛 n과 연관된 픽처 타이밍 SEI 메시지에서 pic_dpb_output_du_delay와 동일한 것으로 추론된다.

[0803] 그 nuh_layer_id가 TargetDecLayerIdList에서의 계층에 속할 경우의 현재의 픽처의 출력은 다음과 같이 특정된다:

[0804] -PicOutputFlag가 1과 동일하고 DpbOutputTime[n]이 AuCpbRemovalTime[n]과 동일할 경우, 현재의 픽처가 출력된다.

[0805] - 이와 다르게, PicOutputFlag가 0과 동일할 경우, 현재의 픽처는 출력되지 않지만, 하위조항 C.3.4에서 특정된 바와 같이, DPB에서의 현재의 픽처의 nuh_layer_id 값에 대응하는 픽처 저장 버퍼에서 저장될 것이다.

[0806] 하나의 실시형태에서: 이와 다르게, PicOutputFlag가 0과 동일할 경우, 현재의 픽처는 출력되지 않지만, 하위조항 C.3.4에서 특정된 바와 같이, DPB에서의 현재의 픽처의 nuh_layer_id 값 currLayerId에 대응하는 픽처 저장

버퍼 PSB[currLayerId]에서 저장될 것이다.

- [0807] - 이와 다르게(PicOutputFlag가 1과 동일하고 DpbOutputTime[n]이 AuCpbRemovalTime[n]보다 더 클 경우), 현재의 픽처는 더 이후에 출력되고 (하위조항 C.3.4에서 특정된 바와 같이) DPB에서 현재의 픽처의 nuh_layer_id 값에 대응하는 픽처 저장 버퍼에서 저장될 것이고, DpbOutputTime[n]를 선행하는 시간에서 1과 동일한 no_output_of_prior_pics_flag의 디코딩 또는 추론에 의해 출력되지 않아야 할 것으로 표시되지 않으면, 시간 DpbOutputTime[n]에서 출력된다.
- [0808] 또 다른 실시형태에서, 상기 단계들은 하기에 대해 특정된다:
- [0809] 그 nuh_layer_id가 (현재의) 동작 포인트에 대응하는 출력 계층 세트에 속하는 계층에 속할 경우의 현재의 픽처의 출력.
- [0810] 또 다른 실시형태에서, 상기 단계들은 하기에 대해 특정된다:
- [0811] 현재의 픽처의 출력(그것이 현재의 동작 포인트에 대한 TargetDecLayerIdList 또는 출력 계층 세트에 속하는지를 검사하지 않음).
- [0812] 출력될 때, 픽처는 픽처에 대한 활성 SPS에서 특정된 적합성 크롭핑 윈도우를 이용하여 크롭핑된다.
- [0813] 픽처 n이, 출력되며 출력되는 비트스트림의 최종 픽처가 아닌 픽처일 때, 변수 DpbOutputInterval[n]의 값은 다음과 같이 유도된다:
- [0814] $DpbOutputInterval[n] = DpbOutputTime[nextPicInOutOrder] - DpbOutputTime[n]$ (C-17)
- [0815] 여기서, nextPicInOutOrder는, 출력 순서에서 픽처 n을 수행하며 1과 동일한 PicOutputFlag를 가지는 픽처이다.
- [0816] C 3.4 현재의 디코딩된 픽처 표기 및 저장
- [0817] 이 하위조항에서 특정된 프로세스는 액세스 유닛 n의 CPB 제거 시간, CpbRemovalTime[n]에서 순간적으로 발생한다.
- [0818] 현재의 디코딩된 픽처는 현재의 픽처의 nuh_layer_id 값에 대응하는 비워진 픽처 저장 버퍼에서의 DPB 에서 저장되고, 현재의 픽처의 nuh_layer_id 값에 대한 DPB 충만도는 1씩 증가되고, 현재의 픽처는 "단기 참조를 위해 이용됨"으로서 표기된다.
- [0819] 하나의 실시형태에서:
- [0820] currLayerId와 동일한 nuh_layer_id를 갖는 현재의 디코딩된 픽처는 DPB에서의 현재의 픽처의 nuh_layer_id 값과 동일한 currLayerId를 갖는 비워진 픽처 저장 버퍼 PSB[currLayerId]에서의 DPB 에서 저장되고, 현재의 픽처의 nuh_layer_id 값에 대한 DPB 충만도 DPBFullness[currLayerId]는 1씩 증가되고, 현재의 픽처는 "단기 참조를 위해 이용됨"으로서 표기된다.
- [0821] C.4 비트스트림 적합성
- [0822] 하위조항 C.4에서의 사양들이 적용된다.
- [0823] C.5 디코더 적합성
- [0824] F.8.1.1 C.5.1 일반
- [0825] 하위조항 C.5.1에서의 사양들이 적용된다.
- [0826] 리스트 (1)
- [0827] 상기 리스트 (1)에서 이용된 바와 같이, PSB는 픽처 저장 버퍼를 지칭한다. DPBFullness는 디코딩된 픽처 버퍼 (DPB)(116)의 충만도를 설명하기 위하여 이용된 변수를 지칭한다.
- [0828] 이하의 리스트 (1A)는 본 시스템들 및 방법들을 수용하기 위한 것에 따른 리스트 (1)의 섹션 C.3.2에 대한 대안적인 접근법을 제공한다. 일부의 구성들에서, 리스트 (1A)는 JCTVC-L1003에서의 섹션 C.3.2에 대한 변경들을 나타내기만 할 수도 있다. 리스트 (1A)는 위에서 정의된 플래그들 AuNoOutputOfPriorPicsFlag 및 AuNoRaslOutputFlag를 이용할 수도 있다.

- [0829] **리스트 1A**
- [0830] C.3.2 DPB로부터의 픽처들의 제거
- [0831] 현재의 픽처의 디코딩 전의 (그러나, 현재의 픽처의 최초 슬라이스의 슬라이스 헤더를 파싱한 후의) DPB 로부터의 픽처들의 제거는 (현재의 픽처를 포함하는) 액세스 유닛 n에 속하는 현재의 픽처의 최초 디코딩 유닛의 CPB 제거 시간에 순간적으로 발생하고 다음과 같이 진행된다:
- [0832] -하위조항 8.3.2에서 특정된 바와 같은 RPS에 대한 디코딩 프로세스가 호출된다.
- [0833] - 현재의 픽처가 픽처 0이 아닌, 1과 동일한 AuNoRaslOutputFlag를 갖는 IRAP 픽처일 때, 다음의 순서화된 단계들이 적용된다:
- [0834] - 또 다른 구성에서, 현재의 픽처가 픽처 0이 아닌, 1과 동일한 NoRaslOutputFlag를 갖는 IRAP 픽처일 때, 다음의 순서화된 단계들이 적용된다:
- [0835] 1. 변수 AuNoOutputOfPriorPicsFlag는 다음과 같이 테스트 중인 디코더에 대해 유도된다:
- [0836] - 현재의 픽처가 CRA 픽처일 경우, AuNoOutputOfPriorPicsFlag는 (현재의 픽처 또는 AU에서의 다른 픽처들에 대한 no_output_of_prior_pics_flag의 값에 관계없이) 1과 동일하게 설정된다.
- [0837] - 이와 다르게, 활성 SPS로부터 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid]의 값이 선행하는 픽처에 대해 활성인 SPS로부터 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid]의 값과는 각각 상이할 경우, AuNoOutputOfPriorPicsFlag는 no_output_of_prior_pics_flag의 값에 관계없이, 테스트 중인 디코더에 의해 1로 설정될 수도 있다(그러나, 설정되지 않아야 함).
- [0838] 주목 - AuNoOutputOfPriorPicsFlag를 no_output_of_prior_pics_flag와 동일하게 설정하는 것이 이 조건들 하에서 선호되지만, 테스트 중인 디코더는 이 경우에 AuNoOutputOfPriorPicsFlag를 1로 설정하도록 허용된다.
- [0839] - 이와 다를 경우, AuNoOutputOfPriorPicsFlag는 다음과 같이, 현재의 픽처 및 AU에서의 다른 픽처들에 대한 no_output_of_prior_pics_flag의 값에 기반으로 하여 설정된다:
- [0840] - no_output_of_prior_pics_flag가 AU에서의 적어도 하나의 픽처에 대해 1과 동일할 경우, AuNoOutputOfPriorPicsFlag는 1과 동일하게 설정된다. 이와 다를 경우, AuNoOutputOfPriorPicsFlag는 0과 동일하게 설정된다. 이에 따라, 이 경우, no_output_of_prior_pics_flag가 AU에서의 모든 픽처들에 대해 0과 동일할 경우에는, AuNoOutputOfPriorPicsFlag가 0과 동일하게 설정된다.
- [0841] - 또 다른 실시형태에서, no_output_of_prior_pics_flag가 현재의 픽처에 대해 1과 동일할 경우, AuNoOutputOfPriorPicsFlag는 1과 동일하게 설정된다. 이와 다를 경우에는, AuNoOutputOfPriorPicsFlag가 변경되지 않은 채로 남겨진다.
- [0842] - 또 다른 실시형태에서, no_output_of_prior_pics_flag가 AU에서의 모든 픽처들에 대해 1과 동일할 경우, AuNoOutputOfPriorPicsFlag는 1과 동일하게 설정된다. 이와 다를 경우, AuNoOutputOfPriorPicsFlag는 0과 동일하게 설정된다.
- [0843] - 또 다른 실시형태에서, no_output_of_prior_pics_flag가 AU에서의 타겟 출력 계층들에 속하는 모든 픽처들에 대해 1과 동일할 경우, AuNoOutputOfPriorPicsFlag는 1과 동일하게 설정된다. 이와 다를 경우, AuNoOutputOfPriorPicsFlag는 0과 동일하게 설정된다.
- [0844] - 또 다른 실시형태에서, no_output_of_prior_pics_flag가 AU에서의 타겟 출력 계층들에 속하는 적어도 하나의 픽처에 대해 1과 동일할 경우, AuNoOutputOfPriorPicsFlag는 1과 동일하게 설정된다. 이와 다를 경우, AuNoOutputOfPriorPicsFlag는 0과 동일하게 설정된다.
- [0845] 2. 테스트 중인 디코더에 대해 유도된 AuNoOutputOfPriorPicsFlag의 값은 HRD에 대해 적용되어, AuNoOutputOfPriorPicsFlag의 값이 1과 동일할 때, DPB에서의 모든 nuh_layer_id 값들에 대응하는 모든 픽처 저장 버퍼들은 이들이 포함하는 픽처들의 출력 없이 비워지고, 모든 nuh_layer_id 값들에 대한 DPB 충만도는 0과 동일하게 설정된다.
- [0846] 또 다른 실시형태에서, 테스트 중인 디코더에 대해 유도된 AuNoOutputOfPriorPicsFlag의 값은 HRD에 대해 적용

되어, AuNoOutputOfPriorPicsFlag의 값이 1과 동일할 때, DPB에서의 현재의 픽처의 nuh_layer_id 값에 대응하는 모든 픽처 저장 버퍼들은 이들이 포함하는 픽처들의 출력 없이 비워지고, 현재의 픽처의 nuh_layer_id 값에 대한 DPB 충만도는 0과 동일하게 설정된다.

- [0847] 리스트 (1A)
- [0848] 이하의 리스트 (2)는 본 시스템들 및 방법들을 수용하기 위하여 변경될 JCTVC-L1008의 섹션들을 제공한다.
- [0849] **리스트 2**
- [0850] F.13 가상 참조 디코더
- [0851] F.13.1 일반
- [0852] 하위조항 C.1에서의 사양들이 적용된다.
- [0853] F.13.2 코딩된 픽처 버퍼(CPB)의 동작
- [0854] 하위조항 C.2에서의 사양들이 적용된다.
- [0855] F.13.3 디코딩된 픽처 버퍼(DPB)의 동작
- [0856] 하위조항 C.3에서의 사양들은 nuh_layer_id의 특별한 값을 갖는 디코딩된 픽처들의 각각의 세트에 대해 별도로 적용된다.
- [0857] 타겟 출력 계층에서 포함되지 않는 픽처들에 대한 PicOutputFlag는 0과 동일하게 설정된다.
- [0858] 동일한 DPB 출력 시간 및 1과 동일한 PicOutputFlag를 갖는 디코딩된 픽처들은 이 디코딩된 픽처들의 nuh_layer_id 값들의 오름차순으로 출력된다.
- [0859] F.13.5 디코더 적합성
- [0860] F.13.5.1 일반
- [0861] 하위조항 C.5.1에서의 사양들이 적용된다.
- [0862] F.13.5.2. 출력 순서 DPB의 동작
- [0863] F.13.5.2.1 일반
- [0864] 디코딩된 픽처 버퍼는 픽처 저장 버퍼들을 포함한다. 각각의 계층은 픽처 저장 버퍼들의 그 자신의 세트로 구성된다. 이에 따라, 각각의 계층의 픽처 저장 버퍼들은 계층의 nuh_layer_id 값과 연관된다. 0과 동일한 nuh_layer_id에 대한 픽처 저장 버퍼들의 수는 0과 동일한 nuh_layer_id를 갖는 계층의 활성 SPS로부터 유도된다. 각각의 제로가 아닌 nuh_layer_id 값에 대한 픽처 저장 버퍼들의 수는 그 제로가 아닌 nuh_layer_id 값에 대한 활성 계층 SPS로부터 유도된다. 픽처 저장 버퍼들의 각각은 "참조를 위해 이용됨"으로서 표기되거나 미래의 출력을 위해 유지되는 디코딩된 픽처를 포함한다. 하위조항 F.13.5.2.2에서 특정된 바와 같은, DPB로부터의 픽처들의 출력 및 제거를 위한 프로세스가 호출되고, 하위조항 F.13.5.2.3에서 특정된 바와 같은 픽처 디코딩, 표기, 추가적인 범핑, 및 저장을 위한 프로세스의 호출이 뒤따른다. "범핑" 프로세스는 하위조항 F.13.5.2.4에서 특정되고, 하위조항들 F.13.5.2.2 및 F.13.5.2.3에서 특정된 바와 같이 호출된다.
- [0865] F.13.5.2.2 DPB로부터의 픽처들의 출력 및 제거
- [0866] 현재의 픽처의 디코딩 전의(그러나, 현재의 픽처의 최초 슬라이스의 슬라이스 헤더를 파싱한 후의) DPB로부터의 픽처들의 출력 및 제거는, 현재의 픽처를 포함하는 액세스 유닛에 속하는 현재의 픽처의 최초 디코딩 유닛이 CPB로부터 제거될 때에 순간적으로 발생하고 다음과 같이 진행된다.
- [0867] 하위조항 8.3.2에서 특정된 바와 같은 RPS에 대한 디코딩 프로세스가 호출된다.
- [0868] - 현재의 픽처가, 1과 동일한 NoRasIOutputFlag를 가지며 픽처 0이 아닌 0과 동일한 nuh_layer_id를 갖는 IRAP 픽처일 경우, 다음의 순서화된 단계들이 적용된다:
- [0869] 1. 변수 NoOutputOfPriorPicsFlag는 다음과 같이 테스트 중인 디코더에 대해 유도된다:
- [0870] - 현재의 픽처가 CRA 픽처일 경우, NoOutputOfPriorPicsFlag는 (no_output_of_prior_pics_flag의 값에 관계없

이) 1과 동일하게 설정된다.

- [0871] - 이와 다르게, 현재의 픽처의 nuh_layer_id 값에 대응하는 활성 SPS로부터 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid]의 값이 현재의 픽처의 nuh_layer_id 값과 동일한 nuh_layer_id 값을 갖는 선행 픽처에 대해 활성인 SPS로부터 각각 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid]의 값과는 상이할 경우, NoOutputOfPriorPicsFlag는 no_output_of_prior_pics_flag의 값에 관계 없이, 테스트 중인 디코더에 의해 1로 설정될 수도 있다(그러나, 설정되지 않아야 함).
- [0872] 주목 - NoOutputOfPriorPicsFlag를 no_output_of_prior_pics_flag와 동일하게 설정하는 것이 이 조건들 하에서 선호되지만, 테스트 중인 디코더는 이 경우에 NoOutputOfPriorPicsFlag를 1로 설정하도록 허용된다.
- [0873] - 이와 다를 경우, NoOutputOfPriorPicsFlag는 no_output_of_prior_pics_flag와 동일하게 설정된다.
- [0874] 2. 테스트 중인 디코더에 대해 유도된 NoOutputOfPriorPicsFlag의 값은 다음과 같이 HRD에 대해 적용된다:
- [0875] - NoOutputOfPriorPicsFlag가 1과 동일할 경우, DPB에서의 현재의 픽처의 nuh_layer_id 값에 대응하는 모든 픽처 저장 버퍼들은 이들이 포함하는 픽처들의 출력 없이 비워지고, 현재의 픽처의 nuh_layer_id 값에 대한 DPB 충만도는 0과 동일하게 설정된다.
- [0876] 하나의 실시형태에서, NoOutputOfPriorPicsFlag가 1과 동일할 경우, DPB에서의 모든 nuh_layer_id 값들에 대응하는 모든 픽처 저장 버퍼들은 이들이 포함하는 픽처들의 출력 없이 비워지고, 모든 nuh_layer_id 값들에 대한 DPB 충만도는 0과 동일하게 설정된다.
- [0877] 하나의 실시형태에서, NoOutputOfPriorPicsFlag가 1과 동일할 경우, DPB에서의 현재의 픽처의 nuh_layer_id 값 currLayerId에 대응하는 모든 픽처 저장 버퍼들 PSB[currLayerId]는 이들이 포함하는 픽처들의 출력 없이 비워지고, DPB 충만도 DPBFullness[currLayerId]는 0과 동일하게 설정된다.
- [0878] 하나의 실시형태에서, NoOutputOfPriorPicsFlag가 1과 동일할 경우, DPB에서의 모든 nuh_layer_id 값들에 대한 모든 픽처 저장 버퍼들 PSB[nuh_layer_id]는 이들이 포함하는 픽처들의 출력 없이 비워지고, 모든 nuh_layer_id 값들에 대한 DPB 충만도 DPBFullness[nuh_layer_id]는 0과 동일하게 설정된다.
- [0879] - 이와 다를 경우(NoOutputOfPriorPicsFlag가 0과 동일함), "출력을 위해 필요하지 않음" 및 "참조를 위해 이용되지 않음"으로서 표기되는 픽처를 포함하는 모든 픽처 저장 버퍼들은 (출력 없이) 비워지고, 현재의 픽처의 nuh_layer_id 값에 대응하는 DPB에서의 모든 비워지지 않은 픽처 저장 버퍼들은 하위조항 F.13.5.2.4에서 특정된 "범핑" 프로세스를 반복적으로 호출함으로써 비워지고, 현재의 픽처의 nuh_layer_id 값에 대한 DPB 충만도는 0과 동일하게 설정된다.
- [0880] 또 다른 실시형태에서, NoOutputOfPriorPicsFlag가 0과 동일할 때, "출력을 위해 필요하지 않음" 및 "참조를 위해 이용되지 않음"으로서 표기되는 픽처를 포함하는 모든 nuh_layer_id 값들에 대응하는 모든 픽처 저장 버퍼들이 (출력 없이) 비워지고, DPB에서의 모든 nuh_layer_id 값들에 대응하는 모든 비워지지 않은 픽처 저장 버퍼들은 하위조항 F.13.5.2.4에서 특정된 "범핑" 프로세스를 반복적으로 호출함으로써 비워지고, 모든 nuh_layer_id 값들에 대한 DPB 충만도는 0과 동일하게 설정된다.
- [0881] -또 다른 실시형태에서, 이와 다를 경우(NoOutputOfPriorPicsFlag가 0과 동일함), "출력을 위해 필요하지 않음" 및 "참조를 위해 이용되지 않음"으로서 표기되는 픽처를 포함하는 모든 픽처 저장 버퍼들은 (출력 없이) 비워지고, 현재의 픽처의 nuh_layer_id 값 currLayerId에 대응하는 DPB에서의 모든 비워지지 않은 픽처 저장 버퍼들 PSB[currLayerId]는 하위조항 F.13.5.2.4에서 특정된 "범핑" 프로세스를 반복적으로 호출함으로써 비워지고, 현재의 픽처의 nuh_layer_id 값에 대한 DPB 충만도 DPBFullness[currLayerId]는 0과 동일하게 설정된다.
- [0882] - 또 다른 실시형태에서, NoOutputOfPriorPicsFlag가 0과 동일할 때, "출력을 위해 필요하지 않음" 및 "참조를 위해 이용되지 않음"으로서 표기되는 픽처를 포함하는 DPB에서의 모든 nuh_layer_id 값들에 대한 모든 픽처 저장 버퍼들 PSB[nuh_layer_id]가 (출력 없이) 비워지고, DPB에서의 모든 nuh_layer_id 값들에 대응하는 모든 비워지지 않은 픽처 저장 버퍼들 PSB[nuh_layer_id]는 하위조항 F.13.5.2.4에서 특정된 "범핑" 프로세스를 반복적으로 호출함으로써 비워지고, 모든 nuh_layer_id 값들에 대한 DPB 충만도 DPBFullness[nuh_layer_id]는 0과 동일하게 설정된다.

- [0883] - 이와 다를 경우(현재의 픽처가 1과 동일한 NoRasIOutputFlag 및 0과 동일한 nuh_layer_id를 갖는 IRAP 픽처가 아님), "출력을 위해 필요하지 않음" 및 "참조를 위해 이용되지 않음"으로서 표기되는 픽처를 포함하는 현재의 픽처의 nuh_layer_id 값에 대응하는 모든 픽처 저장 버퍼들이 (출력 없이) 비워진다. 비워지는 각각의 픽처 저장 버퍼에 대하여, 현재의 디코딩된 픽처의 nuh_layer_id 값에 대응하는 DPB 충만도는 1씩 감소된다. 변수 currLayerId는 현재의 디코딩된 픽처의 nuh_layer_id와 동일하게 설정되고, 다음의 조건들 중의 하나 이상이 참일 때에는, 다음의 조건들의 어느 것도 참이 아닐 때까지, 비워지는 현재의 디코딩된 픽처의 nuh_layer_id 값에 대응하는 각각의 추가적인 픽처 저장 버퍼에 대해 DPB 충만도를 1씩 추가로 감소시키면서, 하위조항 F.13.5.2.4에서 특정된 "범핑" 프로세스는 반복적으로 호출된다.
- [0884] - "출력을 위해 필요함"으로서 표기되는 DPB에서의 currLayerId와 동일한 nuh_layer_id를 갖는 픽처들의 수는 (currLayerId가 0과 동일할 때의) 활성 SPS로부터, 또는 currLayerId의 값에 대한 활성 계층 SPS로부터의 sps_max_num_reorder_pics[HighestTid]보다 더 크다.
- [0885] - (currLayerId가 0과 동일할 때의) 활성 SPS, 또는 currLayerId의 값에 대한 활성 계층 SPS의 sps_max_latency_increase_plus1[HighestTid]은 0과 동일하지 않고, 연관된 변수 PicLatencyCount[currLayerId]가 (currLayerId가 0과 동일할 때의) 활성 SPS, 또는 currLayerId의 값에 대한 활성 계층 SPS로부터 유도된 SpsMaxLatencyPictures[HighestTid] 이상인, "출력을 위해 필요함"으로서 표기되는 DPB에서의 currLayerId와 동일한 nuh_layer_id를 갖는 적어도 하나의 픽처가 있다.
- [0886] - DPB에서의 currLayerId와 동일한 nuh_layer_id를 갖는 픽처들의 수는, (currLayerId가 0과 동일할 때의) 활성 SPS, 또는 currLayerId의 값에 대한 활성 계층 SPS로부터의 sps_max_dec_pic_buffering_minus1[HighestTid] + 1 이상이다.
- [0887] 또 다른 실시형태에서: 이와 다를 경우(현재의 픽처가 1과 동일한 NoRasIOutputFlag 및 0과 동일한 nuh_layer_id를 갖는 IRAP 픽처가 아님), "출력을 위해 필요하지 않음" 및 "참조를 위해 이용되지 않음"으로서 표기되는 픽처를 포함하는 모든 nuh_layer_id 값들에 대응하는 모든 픽처 저장 버퍼들이 (출력 없이) 비워진다. 비워지는 각각의 픽처 저장 버퍼에 대하여, 픽처의 nuh_layer_id 값에 대응하는 DPB 충만도는 1씩 감소된다.
- [0888] 또 다른 실시형태에서: 이와 다를 경우(현재의 픽처가 1과 동일한 NoRasIOutputFlag 및 0과 동일한 nuh_layer_id를 갖는 IRAP 픽처가 아님), "출력을 위해 필요하지 않음" 및 "참조를 위해 이용되지 않음"으로서 표기되는 픽처를 포함하는 현재의 픽처의 nuh_layer_id 값 currLayerId에 대응하는 DPB에서의 모든 픽처 저장 버퍼들 PSB[currLayerId]는 (출력 없이) 비워진다. 비워지는 각각의 픽처 저장 버퍼에 대하여, 현재의 디코딩된 픽처의 nuh_layer_id 값에 대응하는 DPB 충만도 DPBfulness[currLayerId]는 1씩 감소된다.
- [0889] 또 다른 실시형태에서: 이와 다를 경우(현재의 픽처가 1과 동일한 NoRasIOutputFlag 및 0과 동일한 nuh_layer_id를 갖는 IRAP 픽처가 아님), "출력을 위해 필요하지 않음" 및 "참조를 위해 이용되지 않음"으로서 표기되는 픽처를 포함하는 DPB에서의 모든 nuh_layer_id 값들에 대한 모든 픽처 저장 버퍼들 PSB[nuh_layer_id]는 (출력 없이) 비워진다. 비워지는 각각의 픽처 저장 버퍼에 대하여, 비워진 픽처의 nuh_layer_id 값에 대응하는 DPB 충만도 DPBfulness[nuh_layer_id]는 1씩 감소된다.
- [0890] F.13.5.2.3 픽처 디코딩, 표기, 추가적인 범핑, 및 저장
- [0891] 이 하위조항에서 특정된 프로세스들은 현재의 픽처를 포함하는 액세스 유닛 n의 최후 디코딩 유닛이 CPB로부터 제거될 때에 순간적으로 발생한다.
- [0892] 변수 currLayerId는 현재의 디코딩된 픽처의 nuh_layer_id와 동일하게 설정된다.
- [0893] "출력을 위해 필요함"으로서 표기되며 currLayerId와 동일한 nuh_layer_id 값을 가지는 DPB에서의 각각의 픽처에 대하여, 연관된 변수 PicLatencyCount[currLayerId]는 PicLatencyCount[currLayerId] + 1와 동일하게 설정된다.
- [0894] 현재의 픽처는 픽처의 최후 디코딩 유닛이 디코딩된 후에 디코딩된 것으로 간주된다. 현재의 디코딩된 픽처는 DPB에서의 currLayerId(현재의 픽처의 nuh_layer_id 값)에 대응하는 비워진 픽처 저장 버퍼에서 저장되고, 다음이 적용된다:
- [0895] - 현재의 디코딩된 픽처가 1과 동일한 PicOutputFlag를 가질 경우, 그것은 "출력을 위해 필요함"으로서 표기되고, 그 연관된 변수 PicLatencyCount[currLayerId]는 0과 동일하게 설정된다.

- [0896] - 이와 다를 경우(현재의 디코딩된 픽처가 0과 동일한 PicOutputFlag를 가짐), 그것은 "출력을 위해 필요하지 않음"으로서 표기된다.
- [0897] 하나의 실시형태에서, 현재의 디코딩된 픽처는 DPB에서의 currLayerId(현재의 픽처의 nuh_layer_id 값)에 대응하는 비워진 픽처 저장 버퍼에서 저장되고, 다음이 적용된다:
- [0898] - 현재의 디코딩된 픽처가 1과 동일한 PicOutputFlag를 가질 경우, 그것은 "출력을 위해 필요함"으로서 표기되고, 그 연관된 변수 PicLatencyCount[currLayerId]는 0과 동일하게 설정된다. 현재의 디코딩된 픽처와 동일한 픽처 순서 카운트 값(PicOrderCntVal)을 가지는 픽처를 포함하는 모든 픽처 저장 버퍼들은 "출력을 위해 필요함"으로서 표기된다.
- [0899] - 이와 다를 경우(현재의 디코딩된 픽처가 0과 동일한 PicOutputFlag를 가짐), 그것은 "출력을 위해 필요하지 않음"으로서 표기된다.
- [0900] 하나의 실시형태에서, 현재의 디코딩된 픽처는 DPB에서의 LayerId(현재의 픽처의 nuh_layer_id 값)에 대응하는 비워진 픽처 저장 버퍼 PSB[currLayerId]에서 저장되고, 현재의 픽처의 nuh_layer_id 값에 대한 DPB 충만도 DPBFullness[currLayerId]는 1씩 증가되고, 다음이 적용된다:
- [0901] 현재의 디코딩된 픽처는 "단기 참조를 위해 이용됨"으로서 표기된다.
- [0902] 다음의 조건들 중의 하나 이상이 참일 때에는, 다음의 조건들의 어느 것도 참이 아닐 때까지, 하위조항 F.13.5.2.4에서 특정된 "범핑" 프로세스가 반복적으로 호출된다.
- [0903] - "출력을 위해 필요함"으로서 표기되는 DPB에서의 currLayerId와 동일한 nuh_layer_id를 갖는 픽처들의 수는 (currLayerId가 0과 동일할 때의) 활성 SPS로부터, 또는 currLayerId의 값에 대한 활성 계층 SPS로부터의 sps_max_num_reorder_pics[HighestTid]보다 더 크다.
- [0904] - sps_max_latency_increase_plus1[HighestTid]이 0과 동일하지 않고, 연관된 변수 PicLatencyCount[currLayerId]가 (currLayerId가 0과 동일할 때의) 활성 SPS, 또는 currLayerId의 값에 대한 활성 계층 SPS로부터 유도된 SpsMaxLatencyPictures[HighestTid] 이상인, "출력을 위해 필요함"으로서 표기되는 DPB에서의 currLayerId와 동일한 nuh_layer_id를 갖는 적어도 하나의 픽처가 있다.
- [0905] F.13.5.2.4 "범핑" 프로세스
- [0906] "범핑" 프로세스는 다음의 순서화된 단계들로 구성된다:
- [0907] 1. 출력을 위해 최초인 픽처들은 "출력을 위해 필요함"으로서 표기된 DPB에서의 모든 픽처들의 PicOrderCntVal의 최소 값을 가지는 것들로서 선택된다.
- [0908] 2. 이 픽처들은 0과 동일한 nuh_layer_id를 갖는 픽처에 대한 활성 SPS에서, 또는 픽처의 그것과 동일한 nuh_layer_id 값에 대한 활성 계층 SPS에서 특정된 적합성 크롭핑 윈도우를 이용하여 크롭핑되고, 크롭핑된 픽처들은 nuh_layer_id의 오름차순으로 출력되고, 픽처들은 "출력을 위해 필요하지 않음"으로서 표기된다.
- [0909] 3. "참조를 위해 이용되지 않음"으로서 표기된 픽처를 포함하며 크롭핑 및 출력되었던 픽처들 중의 하나를 포함하였던 각각의 픽처 저장 버퍼는 비워진다.
- [0910] 리스트 (2)
- [0911] 이하의 리스트 (2A)는 본 시스템들 및 방법들을 수용하기 위한 것에 따른 리스트 (2)의 섹션 F.13.5.2.2에 대한 대안적인 접근법을 제공한다. 일부의 구성들에서, 리스트 (2A)는 JCTVC-L1008에서의 섹션 F.13.5.2.2에 대한 변경들을 나타내기만 할 수도 있다. 리스트 (2A)는 위에서 정의된 플래그들 AuNoOutputOfPriorPicsFlag 및 AuNoRasIOOutputFlag를 이용할 수도 있다.
- [0912] **리스트 2A**
- [0913] F.13.5.2.2 DPB로부터의 픽처들의 출력 및 제거
- [0914] 현재의 픽처의 디코딩 전의(그러나, 현재의 픽처의 최초 슬라이스의 슬라이스 헤더를 파싱한 후의) DPB로부터의 픽처들의 출력 및 제거는, 현재의 픽처를 포함하는 액세스 유닛의 최초 디코딩 유닛이 CPB로부터 제거될 때에 순간적으로 발생하고 다음과 같이 진행된다.

- [0915] 하위조항 8.3.2에서 특정된 바와 같은 RPS에 대한 디코딩 프로세스가 호출된다.
- [0916] - 현재의 픽처가, 1과 동일한 AuNoRas1OutputFlag를 가지며 픽처 0이 아닌 0과 동일한 nuh_layer_id를 갖는 IRAP 픽처일 경우, 다음의 순서화된 단계들이 적용된다:
- [0917] - 또 다른 구성에서, 현재의 픽처가, 1과 동일한 NoRas1OutputFlag를 가지며 픽처 0이 아닌 0과 동일한 nuh_layer_id를 갖는 IRAP 픽처일 경우, 다음의 순서화된 단계들이 적용된다:
- [0918] 1. 변수 AuNoOutputOfPriorPicsFlag는 다음과 같이 테스트 중인 디코더에 대해 유도된다.
- [0919] - 현재의 픽처가 0과 동일한 nuh_layer_id를 가질 경우, AuNoOutputOfPriorPicsFlag는 0으로 초기화된다. 다음으로:
- [0920] - 현재의 픽처가 CRA 픽처일 경우, AuNoOutputOfPriorPicsFlag는 (현재의 픽처 또는 AU에서의 다른 픽처들에 대한 no_output_of_prior_pics_flag의 값에 관계없이) 1과 동일하게 설정된다.
- [0921] - 이와 다르게, 활성 SPS로부터 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid]의 값이 선행하는 픽처에 대해 활성인 SPS로부터 유도된 pic_width_in_luma_samples, pic_height_in_luma_samples, 또는 sps_max_dec_pic_buffering_minus1[HighestTid]의 값과는 각각 상이할 경우, AuNoOutputOfPriorPicsFlag는 no_output_of_prior_pics_flag의 값에 관계없이, 테스트 중인 디코더에 의해 1로 설정될 수도 있다(그러나, 설정되지 않아야 함).
- [0922] 주목 - AuNoOutputOfPriorPicsFlag를 no_output_of_prior_pics_flag와 동일하게 설정하는 것이 이 조건들 하에서 선호되지만, 테스트 중인 디코더는 이 경우에 AuNoOutputOfPriorPicsFlag를 1로 설정하도록 허용된다.
- [0923] - 이와 다른 경우, AuNoOutputOfPriorPicsFlag는 다음과 같이, 현재의 픽처 및 AU에서의 다른 픽처들에 대한 no_output_of_prior_pics_flag의 값에 기반으로 하여 설정된다:
- [0924] - no_output_of_prior_pics_flag가 AU에서의 적어도 하나의 픽처에 대해 1과 동일할 경우, AuNoOutputOfPriorPicsFlag는 1과 동일하게 설정된다. 이와 다른 경우, AuNoOutputOfPriorPicsFlag는 0과 동일하게 설정된다. 이에 따라, 이 경우, no_output_of_prior_pics_flag가 AU에서의 모든 픽처들에 대해 0과 동일할 경우에는, AuNoOutputOfPriorPicsFlag가 0과 동일하게 설정된다.
- [0925] - 또 다른 실시형태에서, no_output_of_prior_pics_flag가 현재의 픽처에 대해 1과 동일할 경우, AuNoOutputOfPriorPicsFlag는 1과 동일하게 설정된다. 이와 다른 경우에는, AuNoOutputOfPriorPicsFlag가 변경되지 않은 채로 남겨진다.
- [0926] - 또 다른 실시형태에서, no_output_of_prior_pics_flag가 AU에서의 모든 픽처들에 대해 1과 동일할 경우, AuNoOutputOfPriorPicsFlag는 1과 동일하게 설정된다. 이와 다른 경우, AuNoOutputOfPriorPicsFlag는 0과 동일하게 설정된다.
- [0927] - 또 다른 실시형태에서, no_output_of_prior_pics_flag가 AU에서의 타겟 출력 계층들에 속하는 모든 픽처들에 대해 1과 동일할 경우, AuNoOutputOfPriorPicsFlag는 1과 동일하게 설정된다. 이와 다른 경우, AuNoOutputOfPriorPicsFlag는 0과 동일하게 설정된다.
- [0928] - 또 다른 실시형태에서, no_output_of_prior_pics_flag가 AU에서의 타겟 출력 계층에 속하는 적어도 하나의 픽처에 대해 1과 동일할 경우, AuNoOutputOfPriorPicsFlag는 1과 동일하게 설정된다. 이와 다른 경우, AuNoOutputOfPriorPicsFlag는 0과 동일하게 설정된다.
- [0929] 2. 테스트 중인 디코더에 대해 유도된 AuNoOutputOfPriorPicsFlag의 값은 다음과 같이 HRD에 대해 적용된다:
- [0930] - AuNoOutputOfPriorPicsFlag가 1과 동일할 경우, 모든 nuh_layer_id 값들에 대응하는 DPB에서의 모든 픽처 저장 버퍼들은 이들이 포함하는 픽처들의 출력 없이 비워지고, DPB 충만도는 모든 nuh_layer_id 값들에 대해 0과 동일하게 설정된다.
- [0931] - 또 다른 실시형태에서, AuNoOutputOfPriorPicsFlag가 1과 동일할 경우, 현재의 픽처의 모든 nuh_layer_id 값에 대응하는 DPB에서의 모든 픽처 저장 버퍼들은 이들이 포함하는 픽처들의 출력 없이 비워지고, DPB 충만도는 현재의 픽처의 nuh_layer_id 값에 대해 0과 동일하게 설정된다.
- [0932] 이와 다른 경우(AuNoOutputOfPriorPicsFlag가 0과 동일함), "출력을 위해 필요하지 않음" 및 "참조를 위해 이용

되지 않음"으로서 표기되는 픽처를 포함하는 모든 nuh_layer_id 값들에 대응하는 모든 픽처 저장 버퍼들이 (출력 없이) 비워지고, DPB에서의 모든 비워지지 않은 픽처 저장 버퍼들은 하위조항 F.13.5.2.4에서 특정된 "범핑" 프로세스를 반복적으로 호출함으로써 비워지고, 모든 nuh_layer_id 값들에 대한 DPB 충만도는 0과 동일하게 설정된다.

[0933] 또 다른 실시형태에서:

[0934] 이와 다를 경우(AuNoOutputOfPriorPicsFlag가 0과 동일함), "출력을 위해 필요하지 않음" 및 "참조를 위해 이용되지 않음"으로서 표기되는 픽처를 포함하는 현재의 픽처의 nuh_layer_id 값에 대응하는 모든 픽처 저장 버퍼들이 (출력 없이) 비워지고, DPB에서의 모든 비워지지 않은 픽처 저장 버퍼들은 하위조항 F.13.5.2.4에서 특정된 "범핑" 프로세스를 반복적으로 호출함으로써 비워지고, 현재의 픽처의 nuh_layer_id 값에 대한 DPB 충만도는 0과 동일하게 설정된다.

[0935] 리스트 (2A)

[0936] 하나의 구성에서는, 상기 리스트 (1), 리스트 (1A), 리스트 (2) 및 리스트 (2A)에서의 제안된 텍스트에 대하여, "현재의 픽처"의 모든 출현들은 "현재의 디코딩된 픽처"에 의해 대체될 수도 있다.

[0937] 이하의 리스트 (3)은 본 시스템들 및 방법들을 수용하기 위하여 디코딩하기 위한 추가적인 섹션을 제공한다. 리스트 (3)은 본 시스템들 및 방법들을 수용하기 위하여 변경될 JCTVC-L1003의 섹션들(즉, HEVC 사양 버전 34)을 제공한다.

[0938] **리스트 3**

[0939] 8.3.2. 참조 픽처 세트에 대한 디코딩 프로세스

[0940] 현재의 픽처가 1과 동일한 NoRasIOutputFlag를 갖는 IRAP 픽처일 때, (만약 있다면) 현재의 픽처의 nuh_layer_id 값에 대응하는 현재 DPB에서의 모든 참조 픽처들은 "참조를 위해 이용되지 않음"으로서 표기된다.

[0941] 또 다른 실시형태에서: 현재의 픽처가 1과 동일한 NoRasIOutputFlag를 갖는 IRAP 픽처일 때, (만약 있다면) 모든 nuh_layer_id 값들에 대응하는 현재 DPB에서의 모든 참조 픽처들이 "참조를 위해 이용되지 않음"으로서 표기된다.

[0942] 리스트 (3)

[0943] 다수의 상이한 변형들은 다양한 단계들이 픽처 기반 또는 액세스 유닛 기반인 하이브리드 디코딩된 픽처 버퍼 (DPB)(116) 동작들을 정의하기 위하여 이용될 수도 있다. 이하의 표들 (11) 내지 (15)는 단계들 및 대응하는 픽처 기반/액세스 유닛(AU) 기반 설정들의 다수의 변형들을 열거한다.

표 46

[0944]

단계	픽처 기반 / AU 기반
제거	픽처 기반
출력	AU 기반
저장	픽처 기반
표기	출력 표기에 대해 AU 기반, 참조 표기에 대해 픽처 기반
추가적인 출력	AU 기반

[0945] 표 (11)

표 47

[0946]

단계	픽처 기반 / AU 기반
제거	픽처 기반
출력	AU 기반
저장	AU 기반
표기	출력 표기에 대해 AU 기반, 참조 표기에 대해 AU 기반
추가적인 출력	AU 기반

[0947] 표 (12)

표 48

[0948] 단계	픽처 기반 / AU 기반
제거	AU 기반
출력	AU 기반
저장	AU 기반
표기	출력 표기에 대해 AU 기반, 참조 표기에 대해 AU 기반
추가적인 출력	AU 기반

[0949] 표 (13)

표 49

[0950] 단계	픽처 기반 / AU 기반
제거	픽처 기반
출력	AU 기반
저장	픽처 기반
표기	출력 표기에 대해 픽처 기반, 참조 표기에 대해 픽처 기반
추가적인 출력	AU 기반

[0951] 표 (14)

표 50

[0952] 단계	픽처 기반 / AU 기반
제거	픽처 기반
출력	픽처 기반
저장	픽처 기반
표기	출력 표기에 대해 픽처 기반, 참조 표기에 대해 픽처 기반
추가적인 출력	픽처 기반

[0953] 표 (15)

[0954] 도 66은 디코더(2404)의 하나의 구성을 예시하는 블록도이다. 디코더(2404)는 전자 디바이스(2402) 내에 포함될 수도 있다. 예를 들어, 디코더(2404)는 고효율 비디오 코딩(HEVC) 디코더일 수도 있다. 디코더(2404), 및/또는 디코더(2404) 내에 포함된 것으로서 예시된 엘리먼트들 중의 하나 이상은 하드웨어, 소프트웨어, 또는 양자의 조합으로 구현될 수도 있다. 디코더(2404)는 디코딩을 위한 비트스트림(2414)(예컨대, 비트스트림(2414) 내에 포함된 하나 이상의 인코딩된 픽처들)을 수신할 수도 있다. 일부의 구성들에서, 수신된 비트스트림(2414)은 수신된 슬라이스 헤더, 수신된 PPS, 수신된 버퍼 설명 정보 등과 같은 수신된 오버헤드 정보를 포함할 수도 있다. 비트스트림(2414) 내에 포함된 인코딩된 픽처들은 하나 이상의 인코딩된 참조 픽처들 및/또는 하나 이상의 다른 인코딩된 픽처들을 포함할 수도 있다.

[0955] (비트스트림(2414) 내에 포함된 하나 이상의 인코딩된 픽처들에서의) 수신된 심볼들은 엔트로피 디코딩 모듈(454)에 의해 엔트로피 디코딩될 수도 있음으로써, 모션 정보 신호(456) 및 양자화되고, 스케일링되고 및/또는 변환된 계수들(458)을 생성할 수도 있다.

[0956] 모션 정보 신호(456)는, 인터-프레임 예측 신호(468)를 생성할 수도 있는 모션 보상 모듈(460)에서 프레임 메모리(464)로부터의 참조 프레임 신호(484)의 부분과 조합될 수도 있다. 양자화되고, 디스케일링되고, 및/또는 변환된 계수들(458)은 역 모듈(462)에 의해 역양자화될 수도 있고, 스케일링될 수도 있고, 역변환될 수도 있음으

로써, 디코딩된 잔차 신호(470)를 생성할 수도 있다. 디코딩된 잔차 신호(470)는 조합된 신호(472)를 생성하기 위하여 예측 신호(478)에 추가될 수도 있다. 예측 신호(478)는 인터-프레임 예측 신호(782), 또는 인트라-프레임 예측 모듈(474)에 의해 생성된 인트라-프레임 예측 신호(476)로부터 선택된 신호일 수도 있다. 일부의 구성들에서, 이 신호 선택은 비트스트림(2414)에 기반(예컨대, 이에 의해 제어됨)할 수도 있다.

[0957] 인트라-프레임 예측 신호(476)는 (예를 들어, 현재의 프레임에서의) 조합된 신호(472)로부터의 이전에 디코딩된 정보로부터 예측될 수도 있다. 조합된 신호(472)는 또한, 디블록킹 필터(480)에 의해 필터링될 수도 있다. 결과적인 필터링된 신호(482)는 프레임 메모리(464)에 기록될 수도 있다. 결과적인 필터링된 신호(482)는 디코딩된 픽처를 포함할 수도 있다.

[0958] 프레임 메모리(464)는 본원에서 설명된 바와 같이 디코딩된 픽처 버퍼(DPB)(2416)를 포함할 수도 있다. 디코딩된 픽처 버퍼(DPB)(2416)는 하이브리드 디코딩된 픽처 버퍼(DPB)(116) 동작들을 할 수도 있다. 디코딩된 픽처 버퍼(DPB)(2416)는 단기 또는 장기 참조 프레임들로서 유지될 수도 있는 하나 이상의 디코딩된 픽처들을 포함할 수도 있다. 프레임 메모리(464)는 또한, 디코딩된 픽처들에 대응하는 오버헤드 정보를 포함할 수도 있다. 예를 들어, 프레임 메모리(464)는 슬라이스 헤더들, 비디오 파라미터 세트(VPS) 정보, 시퀀스 파라미터 세트(SPS) 정보, 픽처 파라미터 세트(PPS) 정보, 사이클 파라미터들, 버퍼 설명 정보 등을 포함할 수도 있다. 이 정보의 조각들 중의 하나 이상은 인코더(예컨대, 인코더(2108), 오버헤드 시그널링 모듈(2112))로부터 시그널링될 수도 있다.

[0959] 도 67a는 기본 계층 및 강화 계층에 대한 별도의 디코딩된 픽처 버퍼(DPB)들(516a 내지 516b) 및 별도의 하이브리드 디코딩된 픽처 버퍼(DPB) 동작 모듈들(520a 내지 520b)에 의한 비디오 코딩을 위한 강화 계층 및 기본 계층 양자의 이용을 예시하는 블록도이다. 제 1 전자 디바이스(502a) 및 제 2 전자 디바이스(502b)가 예시되어 있다. 제 1 전자 디바이스(502a)는 강화 계층 인코더(526) 및 기본 계층 인코더(528)를 포함하는 비디오 인코더(508)를 포함할 수도 있다. 제 1 전자 디바이스(502a) 내에 포함된 엘리먼트들(즉, 강화 계층 인코더(526) 및 기본 계층 인코더(528))의 각각은 하드웨어, 소프트웨어 또는 양자의 조합으로 구현될 수도 있다. 제 1 전자 디바이스(502a)는 입력 픽처(2506)를 획득할 수도 있다. 일부의 구성들에서, 입력 픽처(2506)는 이미지 센서를 이용하여 제 1 전자 디바이스(502a) 상에서 캡처될 수도 있거나, 메모리로부터 추출될 수도 있거나, 또 다른 전자 디바이스(502)로부터 수신될 수도 있다.

[0960] 강화 계층 인코더(526)는 인코딩된 데이터를 생성하기 위하여 입력 픽처(2506)를 인코딩할 수도 있다. 예를 들어, 강화 계층 인코더(526)는 일련의 입력 픽처들(2506)(예컨대, 비디오)을 인코딩할 수도 있다. 인코딩된 데이터는 인코딩된 강화 계층 비디오 비트스트림(530) 내에 포함될 수도 있다. 강화 계층 인코더(526)는 입력 픽처(2506)에 기반으로 하여 오버헤드 시그널링을 생성할 수도 있다.

[0961] 강화 계층 비디오 디코더(534)는 디코딩된 픽처 버퍼(DPB; 516a) 및 하이브리드 디코딩된 픽처 버퍼(DPB) 동작 모듈(520a)을 포함할 수도 있다. 마찬가지로, 기본 계층 비디오 디코더(536)는 디코딩된 픽처 버퍼(DPB; 516b) 및 하이브리드 디코딩된 픽처 버퍼(DPB) 동작 모듈(520b)을 포함할 수도 있다.

[0962] 기본 계층 인코더(528)는 또한, 입력 픽처(2506)를 인코딩할 수도 있다. 하나의 구성에서, 강화 계층 인코더(526)에 의해 이용된 동일한 입력 픽처(2506)는 또한, 기본 계층 인코더(528)에 의해 이용될 수도 있다. 또 다른 구성에서, 강화 계층 인코더(526)에 의해 이용된 입력 픽처와는 상이한(그러나 유사한) 입력 픽처는 기본 계층 인코더(528)에 의해 이용될 수도 있다. 예를 들어, (품질 스케일러빌리티로서 또한 지칭된) 신호-대-잡음 비율(signal-to-noise ratio; SNR) 스케일러빌리티를 위하여, 동일한 입력 픽처(2506)는 강화 계층 인코더(526) 및 기본 계층 인코더(528)의 양자에 의해 이용될 수도 있다. 또 다른 예로서, 공간적 스케일러빌리티를 위하여, 다운샘플링된 픽처가 기본 계층 인코더(528)에 의해 이용될 수도 있다. 또 다른 예로서, 멀티-뷰 스케일러빌리티를 위하여, 상이한 뷰 픽처가 기본 계층 인코더(528)에 의해 이용될 수도 있다. 기본 계층 인코더(528)는 인코딩된 기본 계층 비디오 비트스트림(532) 내에 포함된 인코딩된 데이터를 생성할 수도 있다.

[0963] 인코딩된 강화 계층 비디오 비트스트림(530) 및 인코딩된 기본 계층 비디오 비트스트림(532)은 입력 픽처(2506)에 기반으로 한 인코딩된 데이터를 각각 포함할 수도 있다. 하나의 예에서, 인코딩된 강화 계층 비디오 비트스트림(530) 및 인코딩된 기본 계층 비디오 비트스트림(532)은 인코딩된 픽처 데이터를 포함할 수도 있다. 일부의 구성들에서, 인코딩된 강화 계층 비디오 비트스트림(530) 및/또는 인코딩된 기본 계층 비디오 비트스트림(532)은 또한, 시퀀스 파라미터 세트(SPS) 정보, 픽처 파라미터 세트(PPS) 정보, 비디오 파라미터 세트(VPS) 정보, 슬라이스 헤더 정보 등과 같은 오버헤드 데이터를 포함할 수도 있다.

- [0964] 인코딩된 강화 계층 비디오 비트스트림(530)은 제 2 전자 디바이스(502b)에 제공될 수도 있다. 마찬가지로, 인코딩된 기본 계층 비디오 비트스트림(532)은 제 2 전자 디바이스(502b)에 제공될 수도 있다. 제 2 전자 디바이스(502b)는 비디오 디코더(2504)를 포함할 수도 있다. 비디오 디코더(2504)는 강화 계층 디코더(534) 및 기본 계층 디코더(536)를 포함할 수도 있다. 하나의 구성에서, 인코딩된 기본 계층 비디오 비트스트림(530)은 기본 계층 디코더(536)에 의해 디코딩되는 반면, 인코딩된 강화 계층 비디오 비트스트림(530)은 강화 계층 디코더(534)에 의해 디코딩된다.
- [0965] 하나의 예에서, 인코딩된 강화 계층 비디오 비트스트림(530) 및 인코딩된 기본 계층 비디오 비트스트림(532)은 유선 또는 무선 링크를 이용하여 제 2 전자 디바이스(502b)로 송신될 수도 있다. 일부의 경우들에는, 이것이 인터넷, 로컬 영역 네트워크(LAN), 또는 디바이스들 사이에서 통신하기 위한 다른 타입의 네트워크와 같은 네트워크를 통해 행해질 수도 있다. 일부의 구성들에서, 인코더들(즉, 강화 계층 인코더(526) 및 기본 계층 인코더(528)) 및 디코더(2504)(예컨대, 기본 계층 디코더(536) 및 강화 계층 디코더(534))는 동일한 전자 디바이스(502)(즉, 제 1 전자 디바이스(502a) 및 제 2 전자 디바이스(502b)는 단일 전자 디바이스(502)의 일부일 수도 있음) 상에서 구현될 수도 있다는 것에 주목해야 한다. 인코더들 및 디코더들이 예를 들어, 동일한 전자 디바이스(502) 상에서 구현되는 구현예에서, 인코딩된 강화 계층 비디오 비트스트림(530) 및 인코딩된 기본 계층 비디오 비트스트림(532)은 다양한 방법들로 비디오 디코더(2504)에 의해 이용가능하게 될 수도 있다. 예를 들어, 인코딩된 강화 계층 비디오 비트스트림(530) 및 인코딩된 기본 계층 비디오 비트스트림(532)은 버스를 통해 비디오 디코더(2504)에 제공될 수도 있거나, 비디오 디코더(2504)에 의해 취출을 위해 메모리 내에 저장될 수도 있다.
- [0966] 비디오 디코더(2504)는 인코딩된 강화 계층 비디오 비트스트림(530) 및 인코딩된 기본 계층 비디오 비트스트림(532)에 기반으로 하여 하나 이상의 디코딩된 픽처들을 생성할 수도 있다. (강화 계층 디코딩된 픽처(538) 및 기본 계층 디코딩된 픽처(540)를 포함할 수도 있는) 디코딩된 픽처(2118)는 디스플레이될 수도 있고, 재생될 수도 있고, 메모리 내에 저장될 수도 있고, 및/또는 또 다른 디바이스로 송신될 수도 있는 등등과 같다.
- [0967] 하나의 예에서, 디코딩된 픽처(2118)는 또 다른 디바이스로, 또는 제 1 전자 디바이스(502a)로 다시 송신될 수도 있다. 디코딩된 픽처(2118)는 또한, 제 2 전자 디바이스(502b) 상에서 저장될 수도 있거나 유지될 수도 있다. 또 다른 예에서, 제 2 전자 디바이스(502b)는 디코딩된 픽처(2118)를 디스플레이할 수도 있다. 다른 구성들에서, 디코딩된 픽처(2118)는 비트스트림(2114)에 대해 수행된 인코딩 및 다른 동작들에 기반으로 한 상이한 성질들을 갖는 입력 픽처(2506)의 엘리먼트들을 포함한다. 일부의 구성들에서, 디코딩된 픽처(2118)는 입력 픽처(2506)와는 상이한 해상도, 포맷, 사양들 또는 다른 속성을 갖는 픽처 스트림 내에 포함될 수도 있다.
- [0968] 도 67b는 기본 계층 및 강화 계층에 대한 공유된 디코딩된 픽처 버퍼(DPB; 516c) 및 공유된 하이브리드 디코딩된 픽처 버퍼(DPB) 동작 모듈(520c)의 이용을 예시하는 블록도이다. 도 67b는, 강화 계층 비디오 디코더(534) 및 기본 계층 디코더(536)가 디코딩된 픽처 버퍼(DPB; 516c) 및 하이브리드 디코딩된 픽처 버퍼(DPB) 동작 모듈(520c)의 양자를 공유한다는 것을 제외하고는, 도 67a의 컴포넌트와 동일한 컴포넌트들을 포함한다.
- [0969] 도 68은 하이브리드 디코딩된 픽처 버퍼(DPB)(116) 동작을 예시하는 타이밍도이다. 도 68의 하이브리드 디코딩된 픽처 버퍼(DPB)(116) 동작들은 바람직한 변형의 단계들을 도시하고, 여기서, 제거, 저장 및 참조 표기는 픽처 기반이고, 출력, 출력 표기 및 추가적인 출력은 액세스 유닛(AU) 기반이다. 이상적인 디코딩된 픽처 버퍼(DPB)(116) 및 가상 참조 디코더(HRD)는, 도시된 다양한 개별적인 단계들(예컨대, 제거(621a 내지 621b), 출력(623), 저장(625a 내지 625c), 표기(627a 내지 627c, 629) 및 추가적인 출력(631))은 모두 순간적으로 수행되도록 동작할 수도 있다. 이 단계들의 시퀀스 및 타이밍 오프셋들은 예시의 목적들을 위하여 개별적인 단계들 사이에서 예시된다. 출력 타이밍 디코더 적합성을 위하여, 픽처 출력의 (최초 비트의 전달 시간에 관련된) 타이밍은 가상 참조 디코더(HRD) 및 테스트 중인 디코더(DUT) 양자에 대해 고정된 지연까지 동일하다. 이와 같이, 예시된 타이밍 오프셋은 HRD와 비교하여 고정된 지연을 도입하는 DUT에 대해 발생할 수 있다.
- [0970] 제 1 강화 계층(EL1)(2642a)의 디코딩된 픽처 버퍼(DPB)(116), 제 2 강화 계층(EL2)(2642b)의 디코딩된 픽처 버퍼(DPB)(116), 및 기본 계층(BL)(2644)의 디코딩된 픽처 버퍼(DPB)(116)에 대한 단계들이 예시되어 있다. 코딩된 픽처 버퍼(CPB) 제거 시간(2646)이 예시되어 있다. 코딩된 픽처 버퍼(CPB) 제거 시간(2646) 후에, (출력 없는) 픽처 기반 제거(621a)는 제 2 강화 계층(EL2)(2642b)의 디코딩된 픽처 버퍼(DPB)(116)에 의해 수행될 수도 있고, (출력 없는) 픽처 기반 제거(621b)는 기본 계층(BL)(2644)의 디코딩된 픽처 버퍼(DPB)(116)에 의해 수행될 수도 있다. 타이밍 오프셋 후에, 액세스 유닛(AU) 기반 픽처 출력(범핑)(623)은 기본 계층(BL)(2644)의 디코딩된 픽처 버퍼(DPB)(116), 제 1 강화 계층(EL1)(2642a)의 디코딩된 픽처 버퍼(DPB)(116), 및 제 2 강화

계층(EL2)(2642b)의 디코딩된 픽처 버퍼(DPB)(116)에 의해 수행될 수도 있다.

- [0971] 또 다른 타이밍 오프셋 후에, 현재의 디코딩된 픽처 관련 프로세스(613)가 예시되어 있다. 픽처 기반 저장 단계(625a 내지 625c)는 기본 계층(BL)(2644)의 디코딩된 픽처 버퍼(DPB)(116), 제 1 강화 계층(EL1)(2642a)의 디코딩된 픽처 버퍼(DPB)(116), 및 제 2 강화 계층(EL2)(2642b)의 디코딩된 픽처 버퍼(DPB)(116)에 의해 수행될 수도 있다. 저장 단계는 기본 계층(BL)(2644), 제 1 강화 계층(EL1)(2642a) 및 제 2 강화 계층(EL2)(2642b)의 각각이 디코딩된 후에 수행될 수도 있다. 저장 단계(625)는 추가로 재분할될 수도 있다. 저장 단계들(625)에서, 디코딩된 픽처는 비워진 저장 버퍼 내의 디코딩된 픽처 버퍼(DPB)(116)에서 저장되고, 디코딩된 픽처 버퍼(DPB) 충만도는 1씩 증가된다. 또한, 픽처가 (출력 없이) 디코딩된 픽처 버퍼(DPB)(116)로부터 제거될 때, 디코딩된 픽처 버퍼(DPB) 충만도는 1씩 감소된다. 유사하게, 픽처가 (범핑 또는 추가적인 범핑 동안에) 디코딩된 픽처 버퍼(DPB)(116)로부터 범핑될 때, 디코딩된 픽처 버퍼(DPB) 충만도는 1씩 감소된다.
- [0972] 디코딩된 픽처 버퍼(DPB)(116)는 상이한 특성들을 가지는 디코딩된 픽처들에 대한 별도로 식별되고 관리된 픽처 버퍼들을 포함할 수도 있다. 예를 들어, 디코딩된 픽처 버퍼(DPB)(116)는 상이한 해상도들, 상이한 비트-심도들 및/또는 상이한 컬러 색도를 갖는 디코딩된 픽처들에 대한 별도로 식별되고 관리된 픽처 버퍼들을 포함할 수도 있다.
- [0973] 디코딩된 픽처는 그 대신에, 디코딩된 픽처 버퍼(DPB)(116) 내의 픽처 저장 버퍼들의 공통 풀 내에 저장될 수도 있다. 예를 들어, 2 개의 추가적인 하위 경우들은 범핑/제거 프로세스 및 레벨 정의들에 영향을 주는 디코딩된 픽처 버퍼(DPB)(116) 사이즈 제약들을 결정하기 위하여 이용될 수도 있다. 바이트 기반 디코딩된 픽처 버퍼(DPB)(116) 제약에서는, 디코딩된 픽처가 해상도 및/또는 비트-심도에 기반으로 한 사이즈에 대한 고려와 함께 저장될 수도 있다. 디코딩된 픽처 버퍼(DPB)(116) 사이즈 제약들은 각각의 디코딩된 픽처의 해상도 및 비트-심도를 고려하는 바이트 한계로서 정의될 수도 있다. 픽처 유닛 기반 디코딩된 픽처 버퍼(DPB)(116) 제약에서는, 디코딩된 픽처가 저장될 수도 있다(그리고 하나의 픽처 버퍼 슬롯을 취하는 것으로 고려됨). 다음으로, 디코딩된 픽처 버퍼(DPB)(116) 사이즈 제약들은 각각의 디코딩된 픽처의 해상도 및 비트-심도를 고려하지 않으면서 픽처 슬롯들의 수 한계로서 정의될 수도 있다.
- [0974] 하나의 구성에서, 디코딩된 픽처 버퍼(DPB) 충만도는 계층 당 추적될 수도 있다. 예를 들어, 계층 당, 디코딩된 픽처 버퍼(DPB)(116) 사이즈 제약들이 시그널링될 수도 있고, 범핑이 적용될 수도 있다. 각각의 계층이 그 자신의 픽처 저장 버퍼들을 포함할 경우, 변수 DPBFullness[nuh_layer_id]는 각각의 계층의 디코딩된 픽처 버퍼(DPB) 충만도까지 추적하기 위하여 이용될 수 있다. 픽처가 nuh_layer_id와 동일한 계층 ID 값을 갖는 계층으로부터 제거될 때, 변수 DPBFullness[nuh_layer_id]는 DPBFullness[nuh_layer_id]-1과 동일하게 설정될 수도 있다(즉, DPBFullness[nuh_layer_id]는 1씩 감소될 수도 있음). 이 경우, 픽처는 픽처 저장 버퍼 PSB[nuh_layer_id]로부터 제거되었다.
- [0975] 유사하게, nuh_layer_id와 동일한 계층 ID 값을 갖는 현재 디코딩된 픽처가 디코딩된 픽처 버퍼(DPB)(116)에서 저장될 때, 변수 DPBFullness[nuh_layer_id]는 DPBFullness[nuh_layer_id]+1과 동일하게 설정된다(즉, DPBFullness[nuh_layer_id]는 1씩 증가됨). 이 경우, 픽처는 픽처 저장 버퍼 PSB[nuh_layer_id] 내로 저장되었다.
- [0976] 디코딩된 픽처 버퍼(DPB) 충만도는 또한, 출력 계층 세트에 대해 추적될 수 있다. 다음으로, 출력 계층 세트에 대해 특정된 제약들에 기반으로 하여, 디코딩된 픽처 버퍼(DPB)(116) 사이즈 제약들이 시그널링될 수도 있고, 범핑이 적용될 수도 있다. DPBFullness 값은 테스트 중인 동작 포인트와 연관되는 출력 계층 세트에 대해 추적될 수 있다. 이에 따라, 픽처가 출력 계층 세트에 속하는 계층으로부터 제거될 때, 디코딩된 픽처 버퍼(DPB) 충만도의 값은 DPBFullness = DPBFullness-1로서 1씩 감소될 수도 있다. 마찬가지로, 현재 디코딩된 픽처가 디코딩된 픽처 버퍼(DPB)(116)에서 저장될 때, 디코딩된 픽처 버퍼(DPB) 충만도는 DPBFullness = DPBFullness+1로서 1씩 감소될 수도 있다.
- [0977] 현재의 코딩된 픽처 관련 프로세스(613) 내에서, 표기 프로세스(611)가 예시되어 있다. 표기 프로세스(611)는 참조 표기(627a 내지 627c) 및 출력 표기(629)를 포함할 수도 있다. 저장 단계들(625a 내지 625c)로부터의 타이밍 오프셋 후에, 픽처 기반 참조 표기 단계(627a 내지 627c)는 기본 계층(BL)(2644)의 디코딩된 픽처 버퍼(DPB)(116), 제 1 강화 계층(EL1)(2642a)의 디코딩된 픽처 버퍼(DPB)(116) 및 제 2 강화 계층(EL2)(2642b)의 디코딩된 픽처 버퍼(DPB)(116)에 의해 수행될 수도 있다. 또 다른 타이밍 오프셋 후에, 액세스 유닛(AU) 기반 출력 표기 단계(629)는 기본 계층(BL)(2644)의 디코딩된 픽처 버퍼(DPB)(116), 제 1 강화 계층(EL1)(2642a)의 디코딩된 픽처 버퍼(DPB)(116), 및 제 2 강화 계층(EL2)(2642b)의 디코딩된 픽처 버퍼(DPB)(116)에 의해 수행

될 수도 있다. 일단 현재의 디코딩된 픽처 관련 프로세스(613)가 완료되면, 액세스 유닛(AU) 기반 픽처 출력(추가적인 범핑) 단계(631)는 기본 계층(BL)(2644)의 디코딩된 픽처 버퍼(DPB)(116), 제 1 강화 계층(EL1)(2642a)의 디코딩된 픽처 버퍼(DPB)(116), 및 제 2 강화 계층(EL2)(2642b)의 디코딩된 픽처 버퍼(DPB)(116)에 의해 수행될 수도 있다.

[0978] 도 69는 제 2 강화 계층(EL2)(942b)이 기본 계층(BL)(944) 및 제 1 강화 계층(EL1)(942a)보다 더 낮은 픽처 레이트를 가질 때, 코딩된 픽처들 및 액세스 유닛(AU)들에 대한 계층들의 네트워크 추상화 계층(NAL) 유닛들에 대한 구조 및 타이밍을 예시하는 블록도이다. EL1 코딩된 픽처(953a)의 NAL 유닛들은 제 1 강화 계층(EL1)(942a)을 따라 예시되어 있다. EL2 코딩된 픽처(953b)의 NAL 유닛들은 제 2 강화 계층(EL2)(942b)을 따라 예시되어 있다. 기본 계층 코딩된 픽처(953c)의 NAL 유닛들은 기본 계층(BL)(944)을 따라 예시되어 있다.

[0979] 시간 t1에서, EL1 코딩된 픽처(953a)의 NAL 유닛들, EL2 코딩된 픽처(953b)의 NAL 유닛들, 및 기본 계층 코딩된 픽처(953c)의 NAL 유닛들은 액세스 유닛(AU)(955a)의 일부이다. 시간 t2에서, EL1 코딩된 픽처(953a)의 NAL 유닛들 및 기본 계층 코딩된 픽처(953c)의 NAL 유닛들은 액세스 유닛(AU)(955b)의 일부이다. 시간 t3에서, EL1 코딩된 픽처(953a)의 NAL 유닛들, EL2 코딩된 픽처(953b)의 NAL 유닛들, 및 기본 계층 코딩된 픽처(953c)의 NAL 유닛들은 액세스 유닛(AU)(955c)의 일부이다. 시간 t4에서, EL1 코딩된 픽처(953a)의 NAL 유닛들 및 기본 계층 코딩된 픽처(953c)의 NAL 유닛들은 액세스 유닛(AU)(955d)의 일부이다.

[0980] 도 70은 기본 계층(BL)(1044)이 제 1 강화 계층(EL1)(1042a) 및 제 2 강화 계층(EL2)(1042b)보다 더 낮은 픽처 레이트를 가질 때, 코딩된 픽처들 및 액세스 유닛(AU)들에 대한 계층들의 네트워크 추상화 계층(NAL) 유닛들에 대한 구조 및 타이밍을 예시하는 블록도이다. EL1 코딩된 픽처(1053a)의 NAL 유닛들은 제 1 강화 계층(EL1)(1042a)을 따라 예시되어 있다. EL2 코딩된 픽처(1053b)의 NAL 유닛들은 제 2 강화 계층(EL2)(1042b)을 따라 예시되어 있다. 기본 계층 코딩된 픽처(1053c)의 NAL 유닛들은 기본 계층(BL)(1044)을 따라 예시되어 있다.

[0981] 시간 t1에서, EL1 코딩된 픽처(1053a)의 NAL 유닛들, EL2 코딩된 픽처(1053b)의 NAL 유닛들, 및 기본 계층 코딩된 픽처(1053c)의 NAL 유닛들은 액세스 유닛(AU)(1055a)의 일부이다. 시간 t2에서, EL1 코딩된 픽처(1053a)의 NAL 유닛들 및 EL2 코딩된 픽처(1053b)의 NAL 유닛들은 액세스 유닛(AU)(1055b)의 일부이다. 시간 t3에서, EL1 코딩된 픽처(1053a)의 NAL 유닛들, EL2 코딩된 픽처(1053b)의 NAL 유닛들, 및 기본 계층 코딩된 픽처(1053c)의 NAL 유닛들은 액세스 유닛(AU)(1055c)의 일부이다. 시간 t4에서, EL1 코딩된 픽처(1053a)의 NAL 유닛들 및 EL2 코딩된 픽처(1053b)의 NAL 유닛들은 액세스 유닛(AU)(1055d)의 일부이다.

[0982] 용어 “컴퓨터-판독가능한 매체”는 컴퓨터 또는 프로세서에 의해 액세스될 수 있는 임의의 이용가능한 매체를 지칭한다. 본원에서 이용된 바와 같은 용어 “컴퓨터-판독가능한 매체”는 비-일시적(non-transitory)이고 유형(tangible)인 컴퓨터-판독가능한 및/또는 프로세서-판독가능한 매체를 나타낼 수도 있다. 제한이 아닌 예로서, 컴퓨터-판독가능한 또는 프로세서-판독가능한 매체들은 RAM, ROM, EEPROM, CD-ROM 또는 다른 광학 디스크 저장, 자기 디스크 저장, 또는 다른 자기 저장 디바이스들, 또는 명령들 또는 데이터 구조들의 형태로 희망하는 프로그램 코드를 반송 또는 저장하기 위해 이용될 수 있으며 컴퓨터 또는 프로세서에 의해 액세스될 수 있는 임의의 다른 매체를 포함할 수도 있다. 본원에서 이용된 바와 같은 디스크(disk) 및 디스크(disc)는 콤팩트 디스크(compact disc; CD), 레이저 디스크(laser disc), 광학 디스크(optical disc), 디지털 다기능 디스크(digital versatile disc; DVD), 플로피 디스크(floppy disk) 및 Blu-ray(등록 상표) 디스크(Blu-ray disc)를 포함하고, 여기서, 디스크(disk)들은 통상 데이터를 자기적으로 재생하는 반면, 디스크(disc)들은 데이터를 레이저들로 광학적으로 재생한다.

[0983] 본원에서 설명된 방법들 중의 하나 이상은 하드웨어를 이용하여 구현될 수도 있고, 및/또는 수행될 수도 있다는 것에 주목해야 한다. 예를 들어, 본원에서 설명된 방법들 또는 접근법들 중의 하나 이상은 칩셋(chipset), ASIC, LSI, 또는 집적 회로 등을 이용하여 구현될 수도 있고 및/또는 실현될 수도 있다.

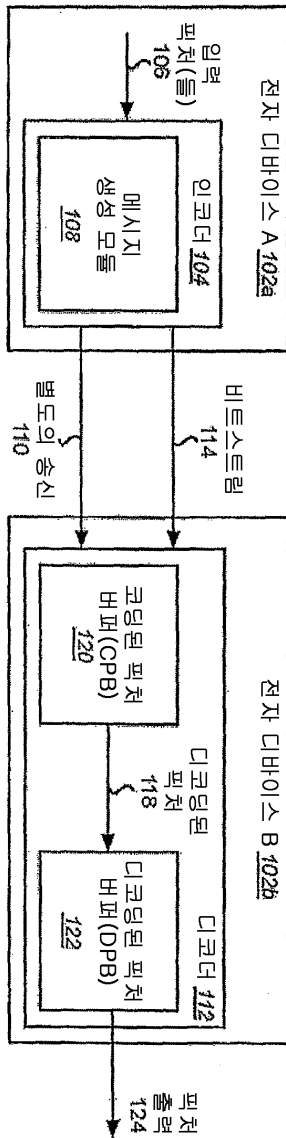
[0984] 본원에서 개시된 방법들의 각각은 설명된 방법을 달성하기 위한 하나 이상의 단계들 또는 액션들을 포함한다. 방법 단계들 및/또는 액션들은 청구의 범위로부터 이탈하지 않으면서 서로 교환될 수도 있고, 및/또는 단일 단계로 조합될 수도 있다. 다시 말해서, 설명되고 있는 방법의 적합한 동작을 위하여 단계들 또는 액션들의 특정한 순서가 요구되지 않는 한, 특정한 단계들 및/또는 액션들의 순서 및/또는 이용은 청구의 범위로부터 이탈하지 않으면서 수정될 수도 있다.

[0985] 청구항들은 위에서 예시된 정밀한 구성 및 컴포넌트들에 제한되지 않는 것으로 이해되어야 한다. 청구의 범위

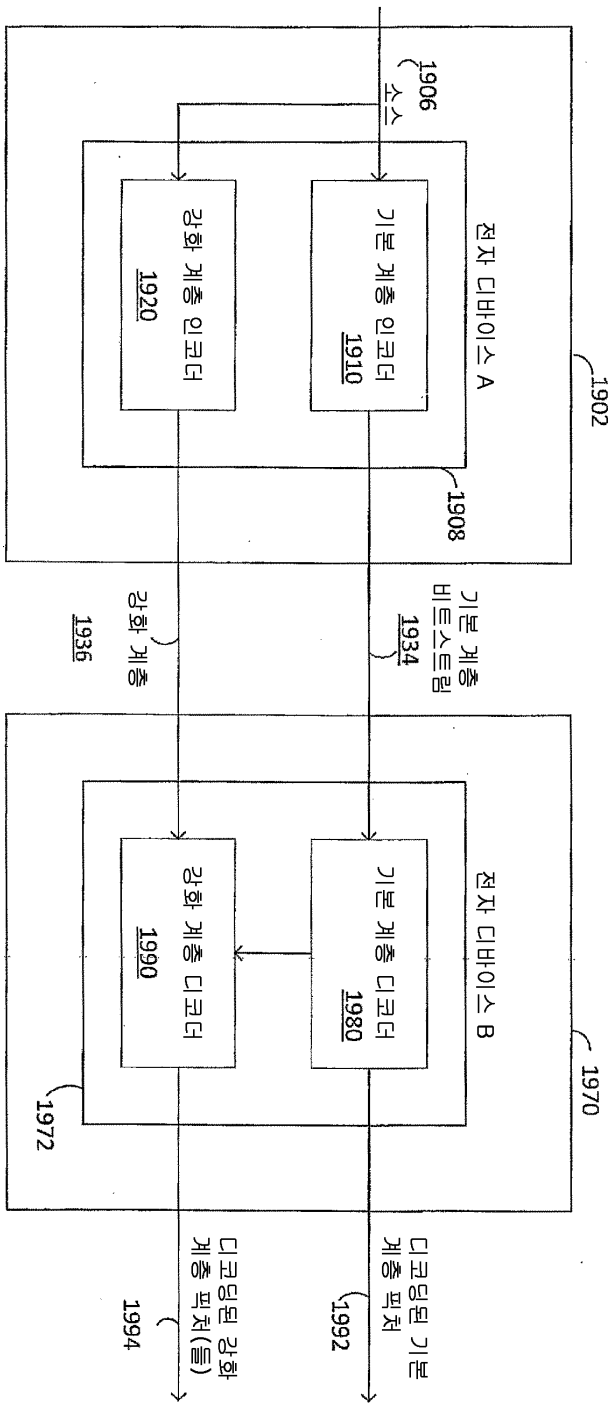
로부터 이탈하지 않고, 다양한 수정들, 변경들 및 변동들이 본원에서 설명되는 시스템들, 방법들, 및 장치의 배치, 동작 및 세부사항들에서 행해질 수도 있다.

도면

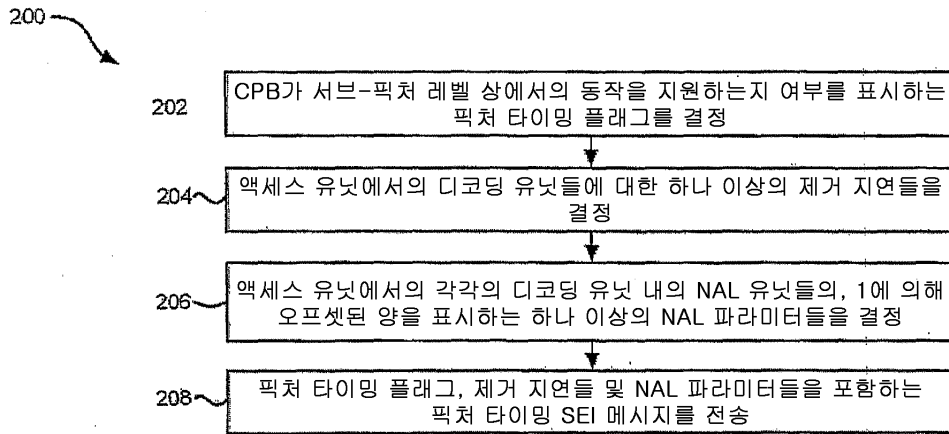
도면1a



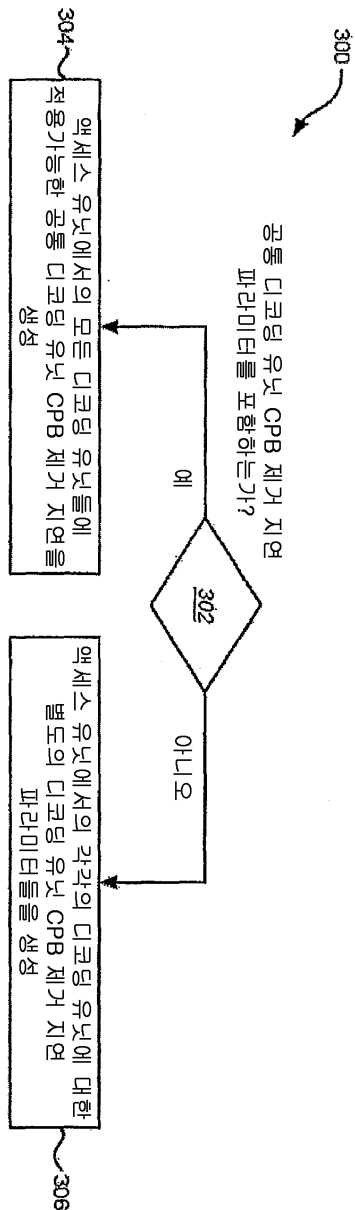
도면1b

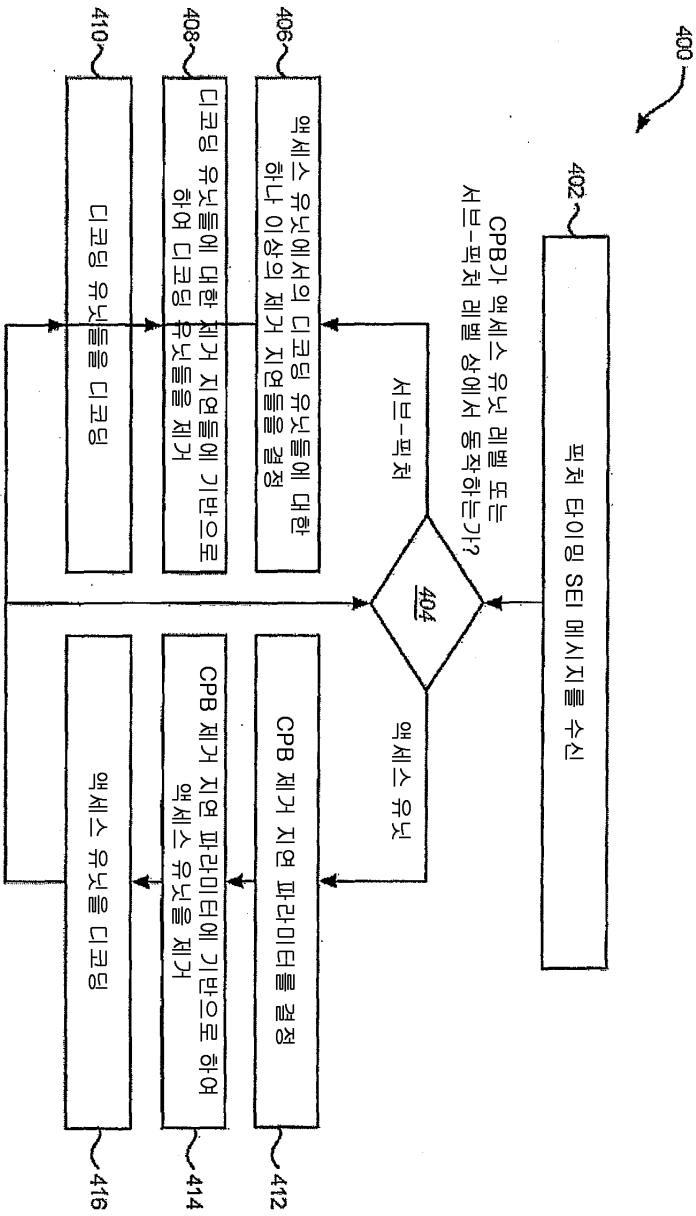


도면2



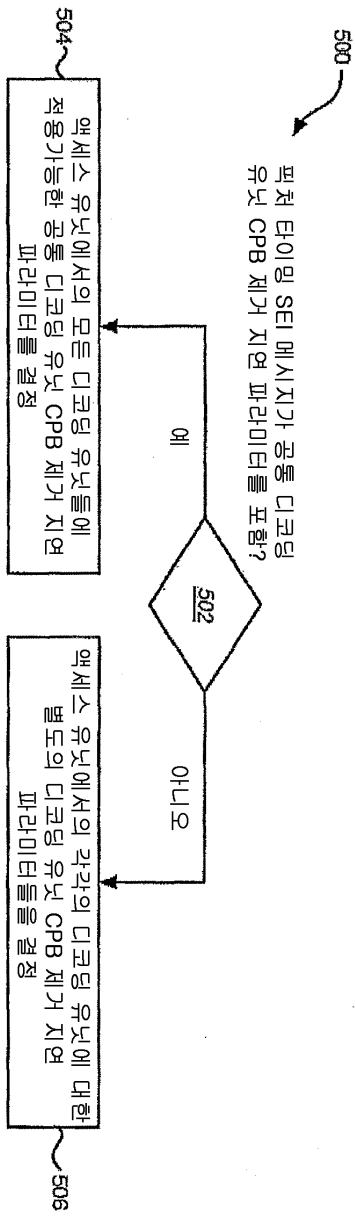
도면3



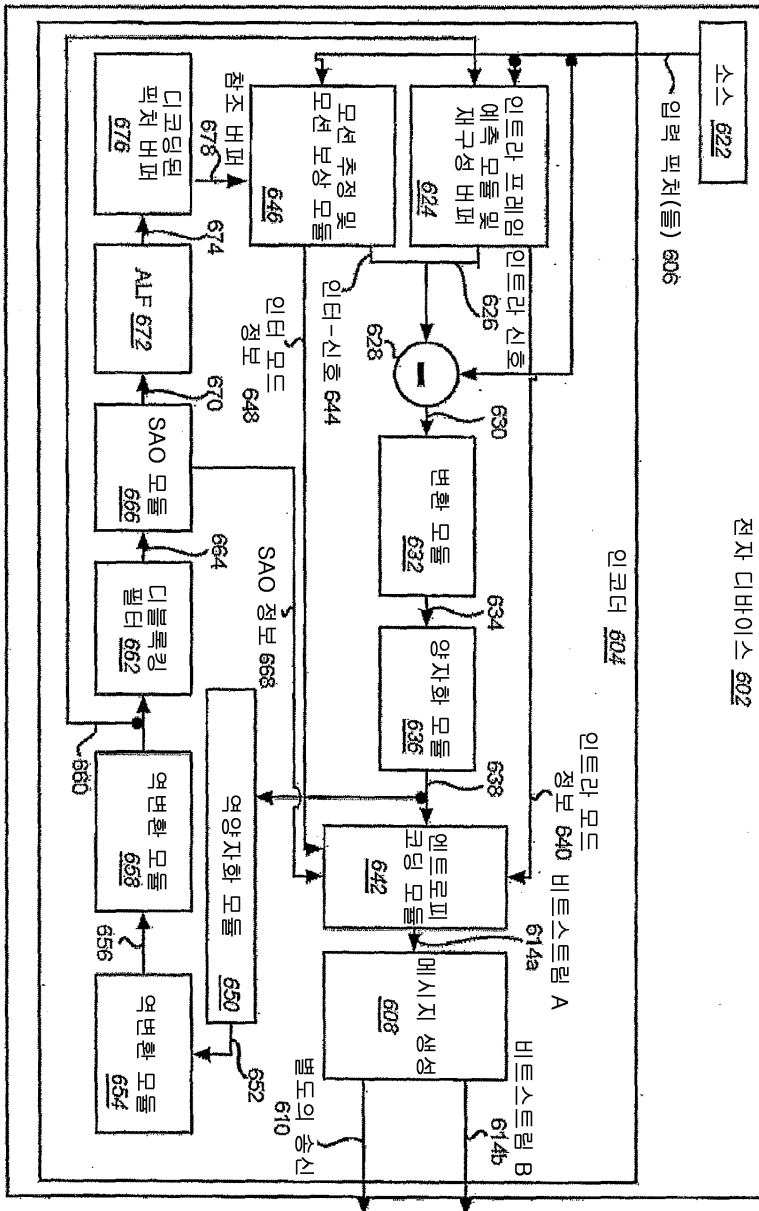


도면4

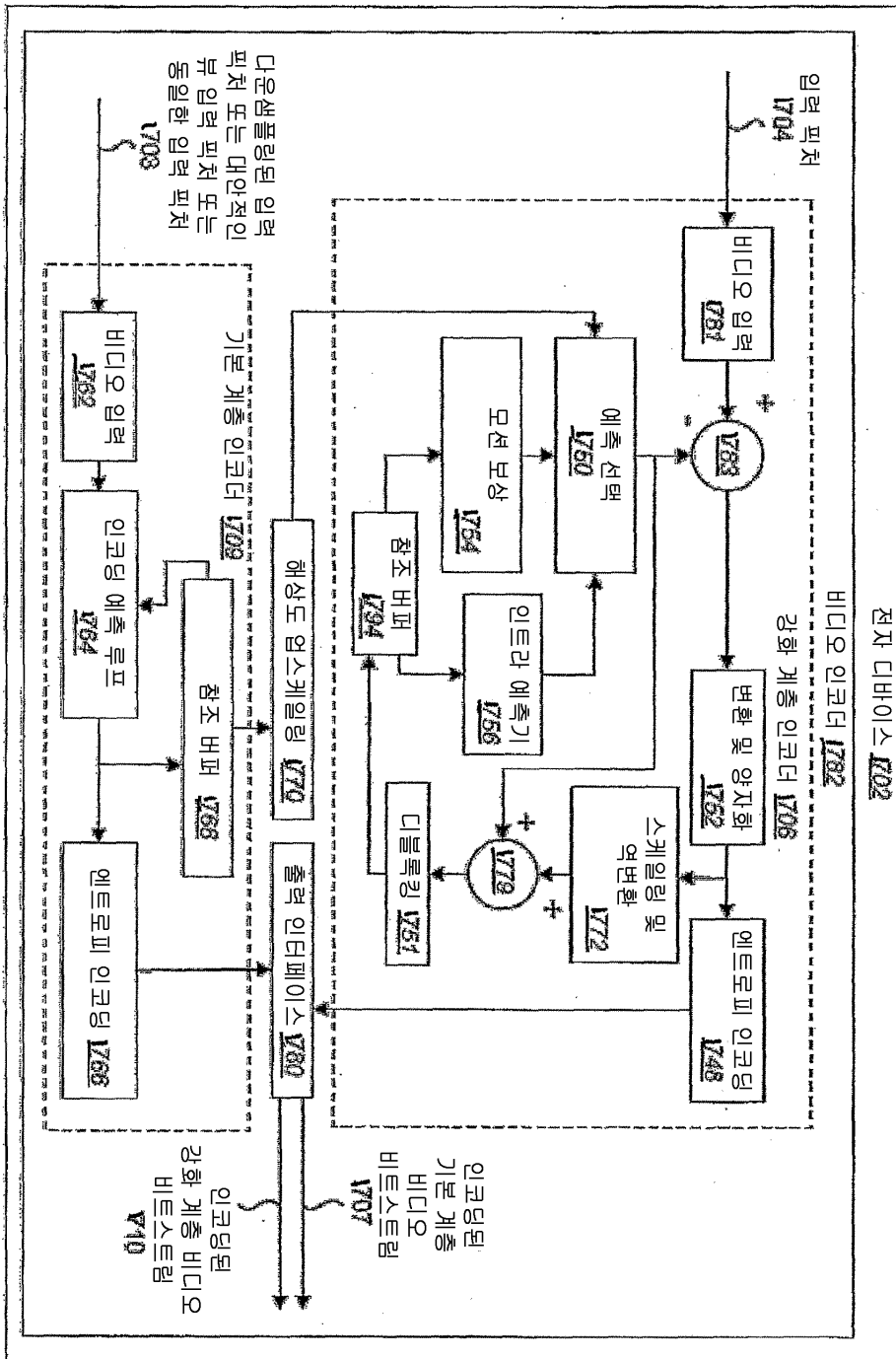
도면5



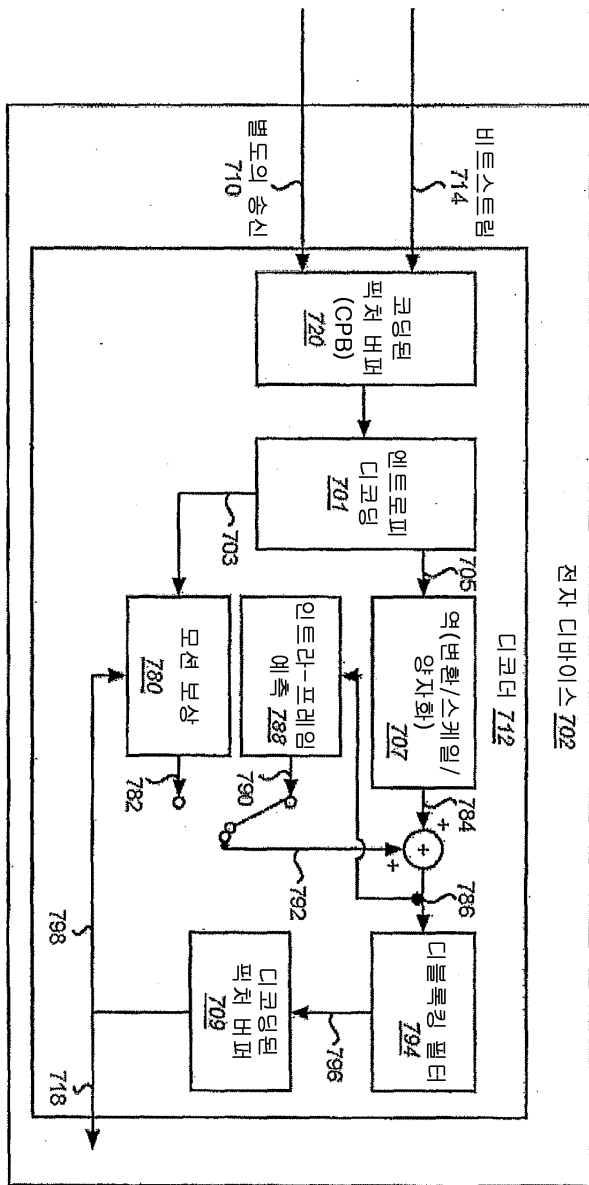
도면6a



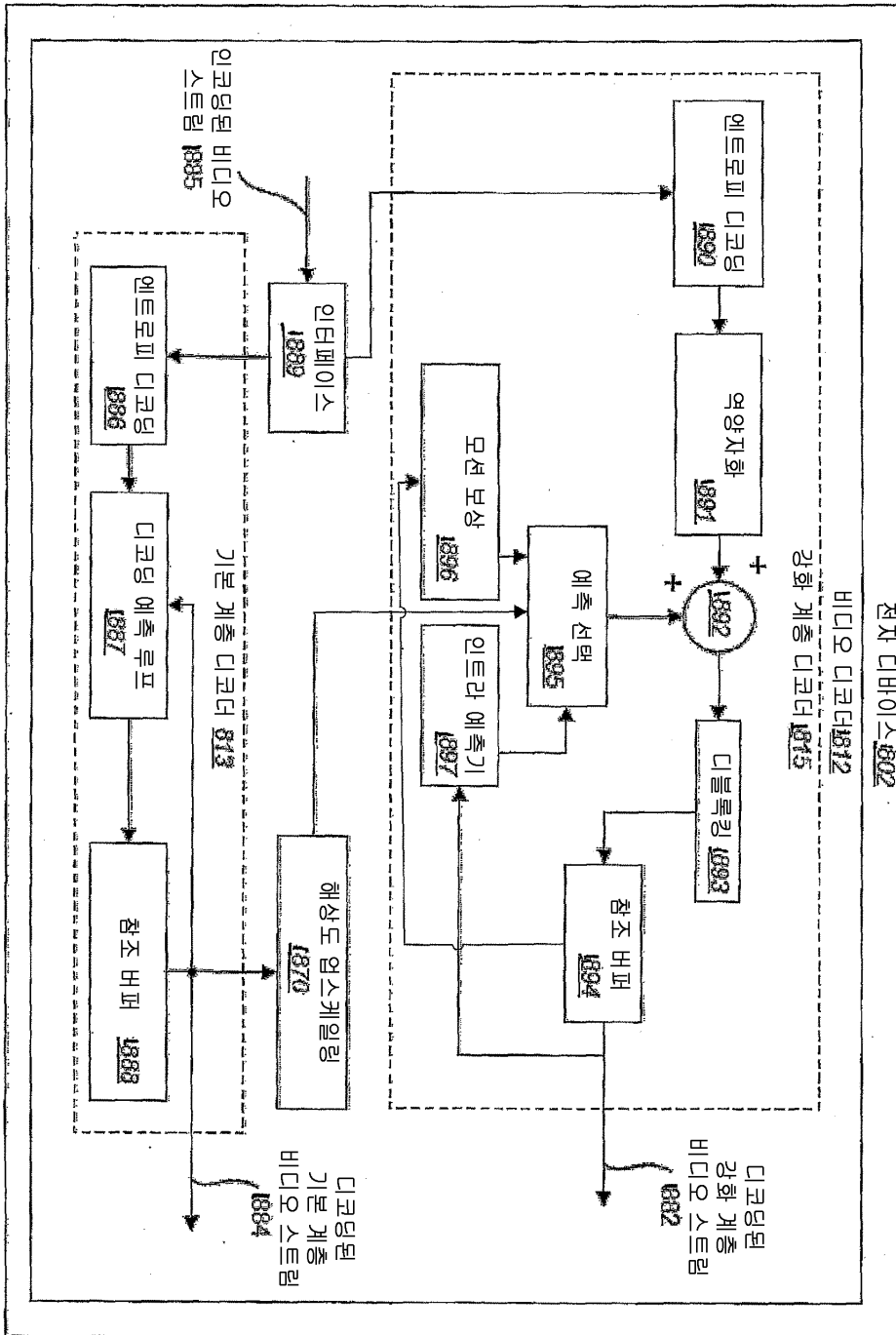
도면6b



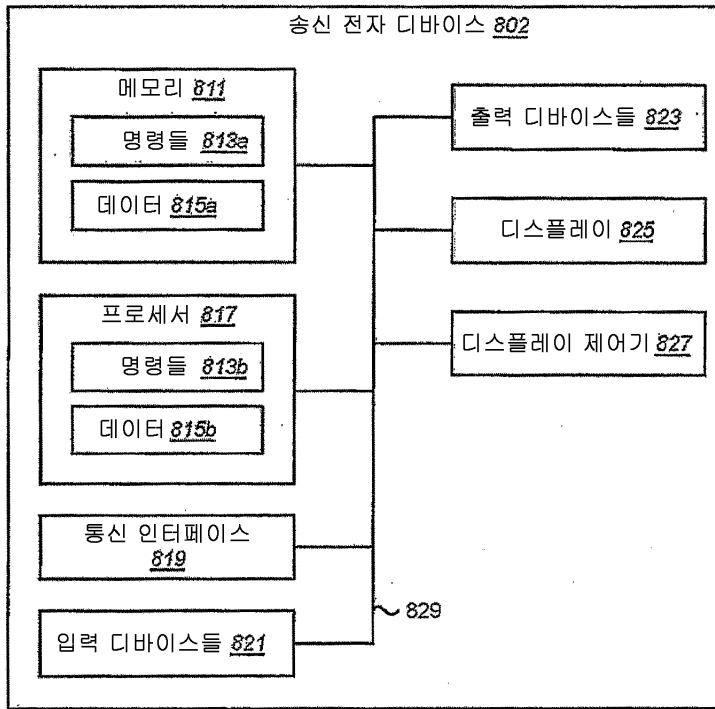
도면7a



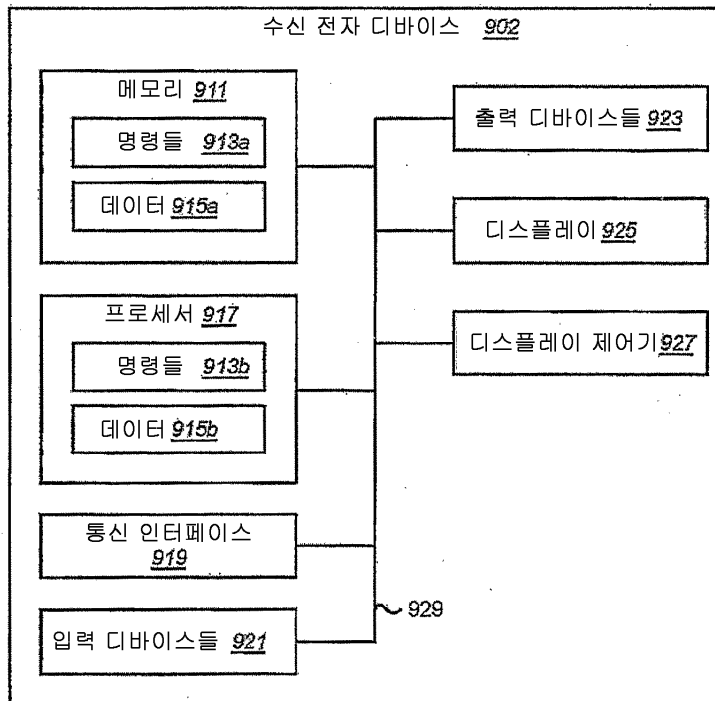
도면 7b



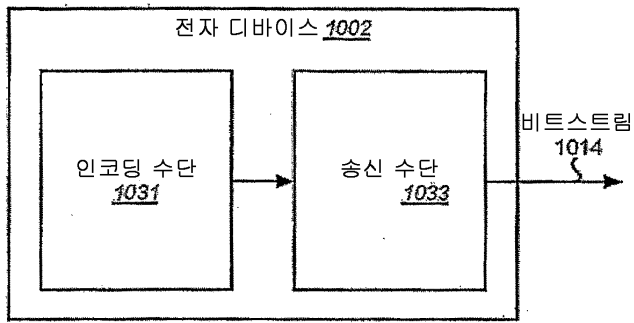
도면8



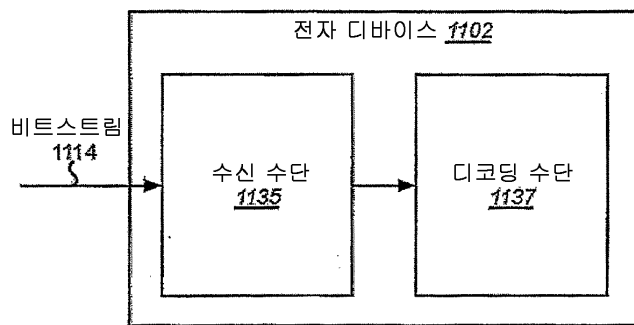
도면9



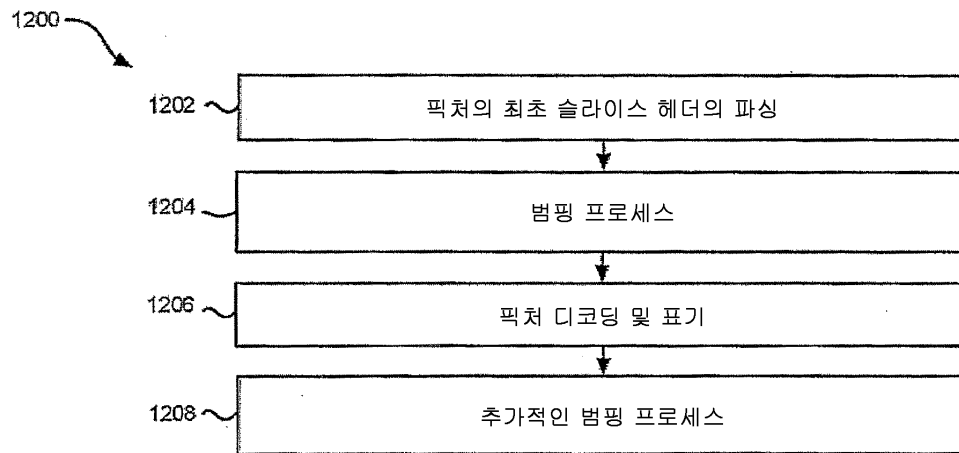
도면10



도면11



도면12



도면13a

nal_unit_header() {	디스크립터
forbidden_zero_bit	f(1)
nal_unit_type	u(6)
nuh_reserved_zero_6bits	u(6)
nuh_temporal_id_plus1	u(3)
}	

NAL 유닛 헤더 신택스

도면13b

nal_unit_header() {	디스크립터
forbidden_zero_bit	f(1)
nal_unit_type	u(6)
layer_id_plus1	u(6)
nuh_temporal_id_plus1	u(3)
}	

NAL 유닛 헤더 신택스

도면13c

nal_unit_header() {	디스크립터
forbidden_zero_bit	f(1)
nal_unit_type	u(6)
layer_id	u(6)
nuh_temporal_id_plus1	u(3)
}	

NAL 유닛 헤더 신택스

도면14

nal_unit(NumBytesInNALUnit) {	디스크립터
nal_unit_header() -	
NumBytesInRBSP == 0	
for(i = 2; i < NumBytesInNALUnit; i++) {	
if(i + 2 < NumBytesInNALUnit && next_bits(24) == 0x000003) {	
rbsp_byte[NumBytesInRBSP++]	b(8)
rbsp_byte[NumBytesInRBSP++]	b(8)
i += 2	
emulation_prevention_three_byte /* equal to 0x03 */	f(8)
} else	
rbsp_byte[NumBytesInRBSP++]	b(8)
}	
}	

일반적인 NAL 유닛 신택스

도면15

vps_extension() {	디스크립터
while(!byte_aligned())	
vps_extension_byte_alignment_reserved_zero_bit	u(1)
// 스케일러빌리티 타입 및 layer_id 파티셔닝 방법	
scalability_type	u(4)
for(i = 0; i < MaxDim(scalability_type); i++)	
layer_id_dim_len[i]	u(3)
// 계층 특징 정보	
for(i = 0; i <= max_num_layers_minus1; i++) {	
vps_layer_id[i]	u(6)
// 계층 종속성	
num_direct_ref_layers[i]	u(6)
for(j = 0; j < num_direct_ref_layers[i]; j++)	
ref_layer_id[i][j]	u(6)
}	
}	
}	

기존의 비디오 파라미터 세트 확장 신택스

도면16

scalability_type	MaxDim(scalability_type)	스케일러빌리티 차원들
0	1	없음 (기본 HEVC)
1	2	공간 및 품질
2	3	공간, 품질, 미특정
3	4	공간, 품질, 미특정, 미특정
4	2	멀티뷰 및 심도
5	3	멀티뷰, 심도, 미특정
6	4	멀티뷰, 심도, 미특정, 미특정
7	4	멀티뷰, 공간, 품질 및 심도
8	5	멀티뷰, 공간, 품질, 심도, 미특정
9	6	멀티뷰, 공간, 품질, 심도, 미특정, 미특정
10...15	예약됨	예약됨

기존의 스케일러빌리티 타입들

도면17

vps_extension() {	디스크립터
while(!byte_aligned())	
vps_extension_byte_alignment_reserved_zero_bit	u(1)
// 스케일러빌리티 맵 및 layer_id_plus1 맵핑 방법	
for (i=0; i<= max_num_layers_minus1_bits; i++) {	
scalability_map[i]	u(3)
}	
// 계층 특정 정보	
for(i=0; i <= max_num_layers_minus1; i++) {	
vps_layer_id[i]	u(6)
// 계층 종속성	
num_direct_ref_layers[i]	u(6)
for(j=0; j < num_direct_ref_layers[i]; j++)	
ref_layer_id[i][j]	u(6)
}	
}	

비디오 파라미터 세트 확장 신택스

도면18

Scalability_map[i]	스케일러빌리티 차원
0	없음 (기본 HEVC)
1	공간
2	품질
3	심도
4	멀티뷰
5	미특정
6, 7	예약됨

스케일러빌리티 맵 신택스

도면19

vps_extension() {	디스크립터
while(!byte_aligned())	
vps_extension_byte_alignment_reserved_zero_bit	u(1)
// 스케일러빌리티 맵 및 layer_id_plus1 맵핑 방법	
num_scalability_dimensions_minus1	u(3)
for(i=0; i<= num_scalability_dimensions_minus1; i++) {	
scalability_map[i]	u(3)
num_bits_for_scalability_map[i]	u(3)
}	
// 계층 특정 정보	
for(i=0; i <= max_num_layers_minus1; i++) {	
vps_layer_id[i]	u(6)
// 계층 종속성	
num_direct_ref_layers[i]	u(6)
for(j=0; j < num_direct_ref_layers[i]; j++)	
ref_layer_id[i][j]	u(6)
}	
}	

비디오 파라미터 세트 확장 선택스

도면20

vps_extension() {	디스크립터
while(!byte_aligned())	
vps_extension_byte_alignment_reserved_zero_bit	u(1)
// 계층 특정 정보	
for(i=1; i <= vps_max_layers_minus1; i++) {	
// 스케일러빌리티 확장 ID들의 계층 ID의 맵핑	
num_dimensions_minus1[i]	u(4)
for(j=0; j <= num_dimensions_minus1; j++) {	
dimension_type[i][j]	u(4)
dimension_id[i][j]	u(8)
}	
// 계층 종속성	
num_direct_ref_layers[i]	u(6)
for(j=0; j < num_direct_ref_layers[i]; j++)	
ref_layer_id[i][j]	u(6)
}	
}	

기존의 비디오 파라미터 세트 확장 선택스

도면21

dimension_type[i][j]	dimension_id[i][j]
0	뷰 순서 인덱스
1	심도 플래그
2	종속성 ID
3	품질 ID
4..15	예약됨

기존의 차원 타입, 차원 ID 선택스

도면22

vps_extension() {	디스크립터
while(!byte_aligned())	
vps_extension_byte_alignment_reserved_zero_bit	u(1)
// 계층 특정 정보	
for(i=1; i <= max_num_layers_minus1; i++) {	
// 이 계층 ID에 대해 존재하는 스케일러빌리티 타입들을 시그널링하는 마스크	
scalability_mask	u(8)
for(j=0; j <= num_scalability_types[i]; j++) {	
scalability_id[j]	u(8)
}	
// 계층 종속성	
num_direct_ref_layers[i]	u(6)
for(j=0; j < num_direct_ref_layers[i]; j++)	
ref_layer_id[i][j]	u(6)
}	
}	

비디오 파라미터 세트 확장 선택스

도면23

scalability_mask_bit[k]	스케일러빌리티 차원
0	공간
1	품질
2	심도
3	멀티뷰
4-15	예약됨

스케일러빌리티 맵 선택스

도면24

video_parameter_set_rbsp() {	디스크립터
video_parameter_set_id	u(4)
vps_temporal_id_nesting_flag	u(1)
reserved_zero_2bits	u(2)
max_num_layers_minus1 // reserved_zero_6bits	u(6)
vps_max_sub_layers_minus1	u(3)
profile_level(1, vps_max_sub_layers_minus1)	
next_essential_info_byte_offset // reserved_zero_12bits	u(12)
for(i = 0; i <= vps_max_sub_layers_minus1; i++) {	
vps_max_dec_pic_buffering[i]	ue(v)
vps_max_num_reorder_pics[i]	ue(v)
vps_max_latency_increase[i]	ue(v)
}	
num_hrd_parameters	ue(v)
for(i = 0; i < num_hrd_parameters; i++) {	
if(i > 0)	
op_point(i)	
hrd_parameters(i == 0, vps_max_sub_layers_minus1)	
}	
bit_equal_to_one	u(1)
scalability_mask	u(8)
for(i = 0; i <= max_num_layers_minus1; i++) {	
vps_extension_data()	
}	
vps_extension_flag	u(1)
if(vps_extension_flag)	
while(more_rbsp_data())	
vps_extension_data_flag	u(1)
rbsp_trailing_bits()	
}	

비디오 파라미터 세트 확장 선택스

도면25

vps_extension_data() {	디스크립터
vps_extension_type	u(2)
layer_id	u(6)
layer_max_sub_layers_minus1	u(3)
new_profile_level_flag	u(1)
if(new_profile_level_flag == 0) {	
profile_reference_layer_id	u(6)
}	
else {	
profile_level(1, layer_max_sub_layers_minus1)	
}	
for(j = 0; j <= num_scalability_types; j++) {	
scalable_id[j]	u(8)
}	
num_reference_layers	u(6)
for(n = 0; n < num_reference_layers; n++)	
direct_encoding_dependency_layer_id_plus1[n]	u(6)
}	

비디오 파라미터 세트 확장 선택스

도면26

vps_extension() {	디스크립터
while(!byte_aligned())	
vps_extension_byte_alignment_reserved_one_bit	u(1)
avc_base_codec_flag	u(1)
scalability_mask	u(16)
for(i = 0; i < NumScalabilityTypes; i++) {	
dimension_id_for_nuh_minus1[i]	u(3)
}	
vps_nuh_layer_id_present_flag	u(1)
// 계층 특정 정보	
for(i = 1; i <= vps_max_layers_minus1; i++) {	
// 스케일러빌리티 확장 ID들의 계층 ID의 맵핑	
if(vps_nuh_layer_id_present_flag)	
layer_id_in_nuh[i]	u(6)
for(j = 0; j < NumScalabilityTypes; j++)	
dimension_id[i][j]	u(v8)
}	
for(i = 1; i <= vps_max_layers_minus1; i++)	
profile_for_level(1, vps_max_sub_layers_minus1)	
for(i = 1; i <= vps_max_layers_minus1; i++) {	
// 계층 중속성	
num_direct_ref_layers[i]	u(4)
for(j = 0; j < num_direct_ref_layers[i]; j++)	
ref_layer_id[i][j]	u(6)
}	
}	
}	

비디오 파라미터 세트 확장 선택스

도면27

scalability_mask's k'th bit	스케일러빌리티 차원
0	공간
1	품질
2	심도
3	멀티뷰
4-15	예약됨

Scalability_mask 선택스

도면28

vps_extension() {	디스크립터
while(!byte_aligned())	
vps_extension_byte_alignment_reserved_one_bit	u(1)
avc_base_codec_flag	u(1)
scalability_mask	u(16)
for(i = 0; i < NumScalabilityTypes; i++) {	
dimension_id_len_minus1[i]	u(3)
}	
vps_nuh_layer_id_present_flag	u(1)
// 계층 특정 정보	
for(i = 1; i <= vps_max_layers_minus1; i++) {	
// 스케일러빌리티 확장 ID들의 계층 ID의 맵핑	
if(vps_nuh_layer_id_present_flag)	
layer_id_in_nuh[i]	u(6)
for(j = 0; j <= num_dimensions_minus1; j++)	
dimension_id[i][j]	u(v)
}	
for(i = 1; i <= vps_max_layers_minus1; i++)	
profile_tier_level(1, vps_max_sub_layers_minus1)	
layer_dependency_information_pattern	u(v)
for(i = 1; i <= NumDepLayers; i++) {	
// 계층 종속성 정보 시그널링	
num_direct_ref_layers[i]	u(6)
for(j = 0; j < num_direct_ref_layers[i]; j++)	
ref_layer_id[i][j]	u(6)
}	
}	
}	

비디오 파라미터 세트 확장 신택스

도면29

vps_extension() {	디스크립터
while(!byte_aligned())	
vps_extension_byte_alignment_reserved_one_bit	u(1)
avc_base_codec_flag	u(1)
scalability_mask	u(16)
for(i=0; i<NumScalabilityTypes; i++) {	
dimension_id_len_minus1[i]	u(3)
}	
vps_nuh_layer_id_present_flag	u(1)
// 계층 특정 정보	
for(i=1; i<=vps_max_layers_minus1; i++) {	
// 스케일러빌리티 확장 ID들의 계층 ID의 맵핑	
if(vps_nuh_layer_id_present_flag)	
layer_id_in_nuh[i]	u(6)
for(j=0; j<=num_dimensions_minus1; j++)	
dimension_id[i][j]	u(v)
}	
for(i=1; i<=vps_max_layers_minus1; i++)	
profile_tier_level(i, vps_max_sub_layers_minus1)	
for(i=1; i<=vps_max_layers_minus1; i++) {	
// 계층 종속성 정보 시그널링	
layer_dependency_flag[i]	u(1)
if(layer_dependency_flag[i]) {	
num_direct_ref_layers[i]	u(6)
for(j=0; j<num_direct_ref_layers[i]; j++)	
ref_layer_id[i][j]	u(6)
}	
}	
}	
}	

비디오 파라미터 세트 확장 선택스

도면30

vps_extension() {	디스크립터
while(lbyte_aligned())	
vps_extension_byte_alignment_reserved_one_bit	u(1)
avc_base_codec_flag	u(1)
scalability_mask	u(16)
for(i = 0; i < NumScalabilityTypes; i++) {	
dimension_id_len_minus1[i]	u(3)
}	
vps_nuh_layer_id_present_flag	u(1)
// 계층 특정 정보	
for(i = 1; i <= vps_max_layers_minus1; i++) {	
// 스케일러빌리티 확장 ID들의 계층 ID의 맵핑	
if(vps_nuh_layer_id_present_flag)	
layer_id_in_nuh[i]	u(6)
for(j = 0; j <= num_dimensions_minus1; j++)	
dimension_id[i][j]	u(v)
}	
for(i = 1; i <= vps_max_layers_minus1 ; i++)	
profile_tier_level(1, vps_max_sub_layers_minus1)	
for(i = 1; i <= vps_max_layers_minus1; i++) {	
// 계층 종속성 정보 시그널링	
layer_dependency_map[i]	u(v)
}	
}	

비디오 파라미터 세트 확장 선택스

도면31

vps_extension() {	디스크립터
while(lbyte_aligned())	
vps_extension_byte_aligument_reserved_one_bit	u(1)
avc_base_codec_flag	u(1)
scalability_mask	u(16)
for(i = 0; i < NumScalabilityTypes; i++) {	
dimension_id_len_minus1[i]	u(3)
}	
vps_nuh_layer_id_present_flag	u(1)
// 계층 특정 정보	
for(i = 1; i <= vps_max_layers_minus1; i++) {	
// 스케일러빌리티 확장 ID들의 계층 ID의 맵핑	
if(vps_nuh_layer_id_present_flag)	
layer_id_in_nuh[i]	u(6)
for(j = 0; j <= num_dimensions_minus1; j++)	
dimension_id[i][j]	u(v)
}	
for(i = 1; i <= vps_max_layers_minus1 ; i++)	
profile_tier_level(1, vps_max_sub_layers_minus1)	
layer_dependency_information_pattern	u(v)
for(i = 1; i <= NumDepLayers; i++)	
layer_dependency_map[i]	u(v)
}	

비디오 파라미터 세트 확장 선택스

도면32

vps_extension() {	디스크립터
while(!byte_aligned())	
vps_extension_byte_alignment_reserved_one_bit	u(1)
avc_base_codec_flag	u(1)
scalability_mask	u(16)
for(i = 0; i < NumScalabilityTypes; i++) {	
dimension_id_leu_minus1[i]	u(3)
}	
vps_nuh_layer_id_present_flag	u(1)
// 계층 특정 정보	
for(i = 1; i <= vps_max_layers_minus1; i++) {	
// 스케일러빌리티 확장 ID들의 계층 ID의 맵핑	
if(vps_nuh_layer_id_present_flag)	
layer_id_in_nuh[i]	u(6)
for(j = 0; j <= num_dimensions_minus1; j++)	
dimension_id[i][j]	u(v)
}	
for(i = 1; i <= vps_max_layers_minus1; i++)	
profile_tier_level(i, vps_max_sub_layers_minus1)	
layer_dependency_information_pattern	u(v)
for(i = 1; i <= vps_max_layers_minus1; i++) {	
if(layer_dependency_information_pattern(i)) {	
// 계층 종속성 정보 시그널링	
num_direct_ref_layers[i]	u(6)
for(j = 0; j < num_direct_ref_layers[i]; j++)	
ref_layer_id[i][j]	u(6)
}	
}	
}	
}	

비디오 파라미터 세트 확장 선택스

도면33

vps_extension() {	디스크립터
while(lbyte_aligned())	
vps_extension_byte_alignment_reserved_one_bit	u(1)
ave_base_codec_flag	u(1)
scalability_mask	u(16)
for(i=0; i < NumScalabilityTypes; i++) {	
dimension_id_len_minus1[i]	u(3)
}	
vps_nuh_layer_id_present_flag	u(1)
// 계층 특정 정보	
for(i=1; i <= vps_max_layers_minus1; i++) {	
// 스케일러빌리티 확장 ID들의 계층 ID의 맵핑	
if(vps_nuh_layer_id_present_flag)	
layer_id_in_nuh[i]	u(6)
for(j=0; j <= num_dimensions_minus1; j++)	
dimension_id[i][j]	u(v)
}	
for(i=1; i <= vps_max_layers_minus1; i++)	
profile_tier_level(1, vps_max_sub_layers_minus1)	
layer_dependency_information_pattern	u(v)
for(i=1; i <= vps_max_layers_minus1; i++) {	
if(layer_dependency_information_pattern(i)) {	
layer_dependency_map[i]	u(v)
}	
}	
}	

비디오 파라미터 세트 확장 선택스

도면34

video_parameter_set_rbsp() {	디스크립터
vps_video_parameter_set_id	u(4)
vps_reserved_three_2bits	u(2)
vps_max_layers_minus1	u(6)
vps_max_sub_layers_minus1	u(3)
vps_temporal_id_nesting_flag	u(1)
vps_extension_offset // vps_reserved_0xffff_16bits	u(16)
profile_tier_level(1, vps_max_sub_layers_minus1)	
vps_sub_layer_ordering_info_present_flag	u(1)
for(i = (vps_sub_layer_ordering_info_present_flag ? 0 : vps_max_sub_layers_minus1); i <= vps_max_sub_layers_minus1; i++) {	
vps_max_dec_pic_buffering_minus1[i]	ue(v)
vps_max_num_reorder_pics[i]	ue(v)
vps_max_latency_increase_plus1[i]	ue(v)
}	
vps_max_layer_id	u(6)
vps_num_layer_sets_minus1	ue(v)
for(i = 1; i <= vps_num_layer_sets_minus1; i++)	
for(j = 0; j <= vps_max_layer_id; j++)	
layer_id_included_flag[i][j]	u(1)
vps_timing_info_present_flag	u(1)
if(vps_timing_info_present_flag) {	
vps_num_units_in_tick	u(32)
vps_time_scale	u(32)
vps_poc_proportional_to_timing_flag	u(1)
if(vps_poc_proportional_to_timing_flag)	
vps_num_ticks_poc_diff_one_minus1	ue(v)
vps_num_hrd_parameters	ue(v)
for(i = 0; i < vps_num_hrd_parameters; i++) {	
hrd_layer_set_idx[i]	ue(v)
if(i > 0)	
cprms_present_flag[i]	u(1)
hrd_parameters(cprms_present_flag[i], vps_max_sub_layers_minus1)	
}	
}	
vps_extension_flag	u(1)
if(vps_extension_flag) {	
vps_extension()	
vps_extension2_flag	u(1)
if(vps_extension2_flag)	
while(more_rbsp_data())	
vps_extension_data_flag	u(1)
}	
rbsp_trailing_bits()	
}	

도면35

vps_extension() {	디스크립터
while(!byte_aligned())	
vps_extension_byte_alignment_reserved_one_bit	u(1)
avc_base_layer_flag	u(1)
splitting_flag	u(1)
for(i = 0, NumScalabilityTypes = 0; i < 16; i++) {	
scalability_mask[i]	u(1)
NumScalabilityTypes += scalability_mask[i]	
}	
for(j = 0; j < NumScalabilityTypes; j++)	
dimension_id_len_minus1[j]	u(3)
vps_nuh_layer_id_present_flag	u(1)
for(i = 1; i <= vps_max_layers_minus1; i++) {	
if(vps_nuh_layer_id_present_flag)	
layer_id_in_nuh[i]	u(6)
for(j = 0; j < NumScalabilityTypes; j++)	
dimension_id[i][j]	u(v)
}	
for(lsIdx = 1; lsIdx <= vps_num_layer_sets_minus1; lsIdx ++) {	
vps_profile_present_flag[lsIdx]	u(1)
if(!vps_profile_present_flag[lsIdx])	
profile_layer_set_ref_minus1[lsIdx]	ue(v)
profile_tier_level(vps_profile_present_flag[lsIdx], vps_max_sub_layers_minus1)	
}	
num_output_layer_sets	ue(v)
for(i = 0; i < num_output_layer_sets; i++) {	
output_layer_set_idx[i]	ue(v)
lsIdx = output_layer_set_idx[i]	
for(j = 0 ; j <= vps_max_layer_id; j++)	
if(layer_id_included_flag[lsIdx][j])	
output_layer_flag[lsIdx][j]	u(1)
}	
for(i = 1; i <= vps_max_layers_minus1; i++)	
for(j = 0; j < i; j++)	
direct_dependency_flag[i][j]	u(1)
}	

도면36

output_layer_sets_change(payloadSize) {	디스크립터
active_vps_id	u(4)
num_changed_output_layer_sets	ue(v)
for(i = 0; i < num_changed_output_layer_sets; i++) {	
changed_output_layer_set_idx_entry[i]	ue(v)
clsIdx = output_layer_set_idx[changed_output_layer_set_idx_entry[i]]	
for(j = 0; j <= vps_max_layer_id; j++)	
if(layer_id_included_flag[clsIdx][j])	
output_layer_flag[clsIdx][j]	u(1)
}	
num_addl_output_layer_sets	
for(i = num_output_layer_sets; i < num_output_layer_sets + num_addl_output_layer_sets; i++) {	
addlIdx = addl_output_layer_sets_idx[i]	ue(v)
for(j = 0; j <= vps_max_layer_id; j++)	
if(layer_id_included_flag[addlIdx][j])	
output_layer_flag[addlIdx][j]	u(1)
}	
}	

도면37

output_layer_sets_change(payloadSize) {	디스크립터
active_vps_id	u(4)
num_deleted_output_layer_sets	ue(v)
for(i = 0; i < num_deleted_output_layer_sets; i++)	
deleted_output_layer_sets_idx_entry[i]	ue(v)
num_changed_output_layer_sets	ue(v)
for(i = 0; i < num_changed_output_layer_sets; i++) {	
changed_output_layer_set_idx_entry[i]	ue(v)
clsIdx = output_layer_set_idx[changed_output_layer_set_idx_entry[i]]	
for(j = 0; j <= vps_max_layer_id; j++)	
if(layer_id_included_flag[clsIdx][j])	
output_layer_flag[clsIdx][j]	u(1)
}	
num_addl_output_layer_sets	
for(i = num_output_layer_sets - num_deleted_output_layer_sets; i < num_output_layer_sets - num_deleted_output_layer_sets + num_addl_output_layer_sets; i++) {	
addlIdx = addl_output_layer_sets_idx[i]	ue(v)
for(j = 0; j <= vps_max_layer_id; j++)	
if(layer_id_included_flag[addlIdx][j])	
output_layer_flag[addlIdx][j]	u(1)
}	
}	

도면38a

vps_extension() {	디스크립터
while(!byte_aligned())	
vps_extension_byte_alignment_reserved_one_bit	u(1)
avc_base_layer_flag	u(1)
splitting_flag	u(1)
for(i = 0, NumScalabilityTypes = 0; i < 16; i++) {	
scalability_mask[i]	u(1)
NumScalabilityTypes += scalability_mask[i]	
}	
for(j = 0; j < NumScalabilityTypes; j++)	
dimension_id_len_minus1[j]	u(3)
...	
num_output_layer_sets	ue(v)
for(i = 0; i < num_output_layer_sets; i++) {	
output_layer_set_idx[i]	ue(v)
lsIdx = output_layer_set_idx[i]	
for(j = 0; j <= vps_max_layer_id; j++)	
if(layer_id_included_flag[lsIdx][j])	
output_layer_flag[lsIdx][j]	u(1)
}	
num_op_dpb_info_parameters	ue(v)
for(i = 0; i < num_op_dpb_info_parameters; i++) {	
operation_point_layer_set_idx[i]	ue(v)
for(j = 0; j <= vps_max_layer_id; j++)	
op_dpb_info_parameters(j)	
}	
for(i = 1; i <= vps_max_layers_minus1; i++) {	
for(j = 0; j < i; j++)	
direct_dependency_flag[i][j]	u(1)
}	
}	

도면38b

vps_extension() {	디스크립터
while(1byte_aligned())	
vps_extension_byte_alignment_reserved_one_bit	u(1)
avc_base_layer_flag	u(1)
splitting_flag	u(1)
for(i = 0; NumScalabilityTypes = 0; i < 16; i++) {	
scalability_mask[i]	u(1)
NumScalabilityTypes += scalability_mask[i]	
}	
for(j = 0; j < NumScalabilityTypes; j++)	
dimension_id_len_minus1[j]	u(3)
...	
num_output_layer_sets	ue(v)
for(i = 0; i < num_output_layer_sets; i++) {	
output_layer_set_idx[i]	ue(v)
lsIdx = output_layer_set_idx[i]	
for(j = 0; j <= vps_max_layer_id; j++)	
if(layer_id_included_flag[lsIdx][j])	
output_layer_flag[lsIdx][j]	u(1)
}	
num_op_dpb_info_parameters	ue(v)
for(i = 0; i < num_op_dpb_info_parameters; i++) {	
operation_point_layer_set_idx[i]	ue(v)
for(j = 0; j <= vps_max_layer_id; j++)	
op_dpb_info_parameters(i, j)	
}	
for(i = 1; i <= vps_max_layers_minus1; i++) {	
for(j = 0; j < i; j++)	
direct_dependency_flag[i][j]	u(1)
}	
}	

도면39a

op_dpb_info_parameters(j) {	
vps_max_sub_layers_minus1[j]	u(3)
vps_sub_layer_ordering_info_present_flag[j]	u(1)
for(k = (vps_sub_layer_ordering_info_present_flag[j] ? 0 : vps_max_sub_layers_minus1[j]);	
k <= vps_max_sub_layers_minus1[j]; k++) {	
vps_max_dec_pic_buffering_minus1[j][k]	ue(v)
vps_max_num_reorder_pics[j][k]	ue(v)
vps_max_latency_increase_plus1[j][k]	ue(v)
}	
}	

도면39b

op_dpb_info_parameters(id, j) {	
vps_max_sub_layers_minus1[id][j]	u(3)
vps_sub_layer_ordering_info_present_flag[id][j]	u(1)
for(k = (vps_sub_layer_ordering_info_present_flag[id][j] ? 0 : vps_max_sub_layers_minus1[id][j]);	
k <= vps_max_sub_layers_minus1[id][j]; k++) {	
vps_max_dec_pic_buffering_minus1[id][j][k]	ue(v)
vps_max_num_reorder_pics[id][j][k]	ue(v)
vps_max_latency_increase_plus1[id][j][k]	ue(v)
}	
}	

도면40

vps_extension() {	디스크립터
while(!byte_aligned())	
vps_extension_byte_alignment_reserved_one_bit	u(1)
avc_base_layer_flag	u(1)
splitting_flag	u(1)
for(i = 0, NumScalabilityTypes = 0; i < 16; i++) {	
scalability_mask[i]	u(1)
NumScalabilityTypes += scalability_mask[i]	
}	
for(j = 0; j < NumScalabilityTypes; j++)	
dimension_id_len_minus1[j]	u(3)
...	
num_output_layer_sets	ue(v)
for(i = 0; i < num_output_layer_sets; i++) {	
output_layer_set_idx[i]	ue(v)
lsIdx = output_layer_set_idx[i]	
for(j = 0; j <= vps_max_layer_id; j++)	
if(layer_id_included_flag[lsIdx][j])	
output_layer_flag[lsIdx][j]	u(1)
}	
num_dpb_info_parameters	ue(v)
for(i = 0; i < num_dpb_info_parameters; i++) {	
output_point_layer_set_idx[i]	ue(v)
oplsIdx = output_point_layer_set_idx[i]	
for(j = 0; j <= vps_max_layer_id; j++)	
oop_dpb_info_parameters(j)	
}	
for(i = 1; i <= vps_max_layers_minus1; i++)	
for(j = 0; j < i; j++)	
direct_dependency_flag[i][j]	u(1)
}	

도면41

oop_dpb_info_parameters(j) {	
vps_max_sub_layers_minus1[j]	u(3)
vps_sub_layer_ordering_info_present_flag[j]	u(1)
for(k = (vps_sub_layer_ordering_info_present_flag[j] ? 0 : vps_max_sub_layers_minus1[j]); k <= vps_max_sub_layers_minus1[j]; k++) {	
vps_max_dec_pic_buffering_minus1[j][k]	ue(v)
vps_max_num_reorder_pics[j][k]	ue(v)
vps_max_latency_increase_plus1[j][k]	ue(v)
}	
}	

도면42

oop_dpb_info_parameters() {	
vps_max_sub_layers_minus1	u(3)
vps_sub_layer_ordering_info_present_flag	u(1)
for(k=(vps_sub_layer_ordering_info_present_flag?0: vps_max_sub_layers_minus1); k <= vps_max_sub_layers_minus1; k++) {	
vps_max_dec_pic_buffering_minus1[k]	ue(v)
vps_max_num_reorder_pics[k]	ue(v)
vps_max_latency_increase_plus1[k]	ue(v)
}	
}	

도면43

num_dpb_info_parameters	ue(v)
for(i=0; i<num_dpb_info_parameters; i++) {	
output_point_layer_set_idx[i]	ue(v)
oplsIdx=output_point_layer_set_idx[i]	
for(j=0; j<=vps_max_layer_id; j++)	
oop_dpb_info_parameters()	
}	

도면44

oop_dpb_info_parameters(id, j) {	
vps_max_sub_layers_minus1[id][j]	u(3)
vps_sub_layer_ordering_info_present_flag[id][j]	u(1)
for(k=(vps_sub_layer_ordering_info_present_flag[id][j]? 0: vps_max_sub_layers_minus1[id][j]); k <= vps_max_sub_layers_minus1[id][j]; k++) {	
vps_max_dec_pic_buffering_minus1[id][j][k]	ue(v)
vps_max_num_reorder_pics[id][j][k]	ue(v)
vps_max_latency_increase_plus1[id][j][k]	ue(v)
}	
}	

도면45

num_dpb_info_parameters	ue(v)
for(i=0; i<num_dpb_info_parameters; i++) {	
output_point_layer_set_idx[i]	ue(v)
oplsIdx=output_point_layer_set_idx[i]	
for(j=0; j<=vps_max_layer_id; j++)	
oop_dpb_info_parameters(oplsIdx, j)	
}	

도면46

num_dpb_info_parameters	ue(v)
for(i=0; i<num_dpb_info_parameters; i++) {	
output_point_layer_set_idx[i]	ue(v)
oplsIdx=output_point_layer_set_idx[i]	
for(j=0; j<=vps_max_layer_id; j++)	
oop_dpb_info_parameters(i, j)	
}	

도면47

vps_extension() {	디스크립터
while(!byte_aligned())	
vps_extension_byte_alignment_reserved_one_bit	u(1)
ave_base_layer_flag	u(1)
splitting_flag	u(1)
for(i = 0, NumScalabilityTypes = 0; i < 16; i++) {	
scalability_mask[i]	u(1)
NumScalabilityTypes += scalability_mask[i]	
}	
for(j = 0; j < NumScalabilityTypes; j++)	
dimension_id_len_minus1[j]	u(3)
...	
for(i = 1; i <= vps_max_layers_minus1; i++) {	
for(j = 0; j < i; j++)	
direct_dependency_flag[i][j]	u(1)
for(i = 1; i <= vps_max_layers_minus1; i++) {	
layer_dpb_info(i)	
}	
}	
}	

layer_dpb_info(i) {	
vps_max_sub_layers_minus1[i]	u(3)
vps_sub_layer_ordering_info_present_flag[i]	u(1)
for(k = (vps_sub_layer_ordering_info_present_flag[i] ? 0 : vps_max_sub_layers_minus1[i]); k <= vps_max_sub_layers_minus1[i]; k++) {	
vps_max_dec_pic_buffering_minus1[i][k]	ue(v)
vps_max_num_reorder_pics[i][k]	ue(v)
vps_max_latency_increase_plus1[i][k]	ue(v)
}	
}	

도면48

oop_dpb_info_parameters(id) / op_dpb_info_parameters(id) / layer_dpb_info(id) {	
vps_sub_layer_ordering_info_present_flag[id]	u(1)
for(i = (vps_sub_layer_ordering_info_present_flag ? 0 : vps_max_sub_layers_minus1); i <= vps_max_sub_layers_minus1; i++) {	
vps_max_dec_pic_buffering_minus1[id][i]	ue(v)
vps_max_num_reorder_pics[id][i]	ue(v)
vps_max_latency_increase_plus1[id][i]	ue(v)
}	
}	

도면49a

vps_extension() {	디스크립터
while(!byte_aligned())	
vps_extension_byte_alignment_reserved_one_bit	u(1)
avc_base_layer_flag	u(1)
splitting_flag	u(1)
for(i = 0, NumScalabilityTypes = 0; i < 16; i++) {	
scalability_mask[i]	u(1)
NumScalabilityTypes += scalability_mask[i]	
}	
for(j = 0; j < NumScalabilityTypes; j++)	
dimension_id_len_minus1[j]	u(3)
vps_nuh_layer_id_present_flag	u(1)
for(i = 1; i <= vps_max_layers_minus1; i++) {	
if(vps_nuh_layer_id_present_flag)	
layer_id_fm_nuh[i]	u(6)
for(j = 0; j < NumScalabilityTypes; j++)	
dimension_id[i][j]	u(v)
}	
for(i = 1; i <= vps_max_layers_minus1; i++) {	
max_sub_layers_vps_predict_flag[i]	u(1)
if(!max_sub_layers_vps_predict_flag[i])	
max_sub_layers_vps_minus1[i]	u(3)
}	
vps_number_layer_sets_minus1	u(10)
vps_num_profile_tier_level_minus1	u(6)
for(i = 1; i <= vps_num_profile_tier_level_minus1; i++) {	
vps_profile_present_flag[i]	u(1)
if(!vps_profile_present_flag[i])	
profile_ref_minus1[i]	u(6)
profile_tier_level(vps_profile_present_flag[i], vps_max_sub_layers_minus1)	
}	

도면49b

multiple_output_layer_sets_in_layer_setmore_output_layer_sets_than_default_flag	u(1)
if(!multiple_output_layer_sets_in_layer_setmore_output_layer_sets_than_default_flag)	
numOutputLayerSets = vps_number_layer_sets_minus1 + 1	
else {	
num_add_output_layer_sets	u(10)
numOutputLayerSets = num_add_output_layer_sets + vps_number_layer_sets_minus1 + 1	
}	
if(numOutputLayerSets > 1)	
default_one_target_output_layer_flag	u(1)
for(i = 1; i < numOutputLayerSets; i++) {	
if(i > vps_num_layer_sets_minus1) {	
output_layer_set_idx_minus1[i]	u(v)
lsIdx = output_layer_set_idx_minus1[i] + 1	
for(j = 0; j < NumLayersInIdList[lsIdx] - 1; j++)	
output_layer_flag[i][j]	u(1)
}	
profile_level_tier_idx[i]	u(v)
}	
num_dpb_info_parameters	ue(v)
for(i = 0; i < num_dpb_info_parameters; i++) {	
output_point_layer_set_idx[i]	ue(v)
oop_dpb_maxbuffering_parameters(i)	
}	
for(i = 1; i <= vps_max_layers_minus1; i++) {	
layer_dpb_info_parameters(i)	
}	
}	
for(i = 1; i <= vps_max_layers_minus1; i++)	
for(j = 0; j < i; j++)	
direct_dependency_flag[i][j]	u(1)
}	

도면50

oop_dpb_maxbuffering_parameters(i) {	
sub_layer_vps_buf_info_present_flag[i]	u(1)
for(j = (sub_layer_vps_buf_info_present_flag[i] ? 0 : MaxSubLayers[i] - 1); j <= MaxSubLayers[i] - 1; j++)	
max_vps_dec_pic_buffering_minus1[i][j]	ue(v)
}	

도면51

layer_dpb_info_parameters(i) {	
sub_layer_vps_ordering_info_present_flag[i]	u(1)
for(j = (sub_layer_vps_ordering_info_present_flag[i] ? 0 : max_sub_layers_vps_minus1[i]); j <= max_sub_layers_vps_minus1[i]; j++) {	
max_vps_num_reorder_pics [i][j]	ue(v)
max_vps_latency_increase_plus1[i][j]	ue(v)
}	
}	

도면52

vps_extension() {	디스크립터
while(!byte_aligned())	
vps_extension_byte_alignment_reserved_one_bit	u(1)
...	
for(j = 0; j < NumScalabilityTypes; j++)	
dimension_id[i][j]	u(v)
}	
for(i = 1; i <= vps_max_layers_minus1; i++) {	
max_sub_layers_vps_predict_flag[i]	u(1)
if(!max_sub_layers_vps_predict_flag[i])	
max_sub_layers_vps_minus1[i]	u(3)
}	
...	
profile_level_tier_idx[i]	u(v)
}	
num_dpb_info_parameters	ue(v)
for(i = 0; i < num_dpb_info_parameters; i++) {	
if(i>0 && (MaxSubLayers[i] == MaxSubLayers[i - 1]))	
sub_layer_vps_buf_info_predict_flag[i]	u(1)
if(!sub_layer_vps_buf_info_predict_flag[i]) {	
output_point_layer_set_idx[i]	ue(v)
oop_dpb_maxbuffering_parameters(i)	
}	
}	
for(i = 1; i <= vps_max_layers_minus1; i++) {	
if(max_sub_layers_vps_predict_flag[i])	
sub_layer_vps_ordering_info_predict_flag[i]	u(1)
if(!sub_layer_ordering_info_predict_flag[i])	
layer_dpb_info_paramcters(i)	
}	
}	
...	
}	

도면53

vps_extension() {	디스크립터
while(lbyte_aligned())	
vps_extension_byte_alignment_reserved_one_bit	u(1)
...	
for(j = 0; j < NumScalabilityTypes; j++)	
dimension_id[i][j]	u(v)
}	
for(i = 1; i <= vps_max_layers_minus1; i++) {	
max_sub_layers_vps_predict_flag[i]	u(1)
if(!max_sub_layers_vps_predict_flag[i])	
max_sub_layers_vps_minus1[i]	u(3)
}	
...	
profile_level_tier_idx[i]	u(v)
}	
num_dpb_info_parameters	ue(v)
for(i = 0; i < num_dpb_info_parameters; i++) {	
output_point_layer_set_idx[i]	ue(v)
for(k = 1; k <= vps_max_layers_minus1; i++) {	
oop_dpb_maxbuffering_parameters(i, k)	
}	
}	
for(i = 1; i <= vps_max_layers_minus1; i++) {	
layer_dpb_info_parameters(i)	
}	
}	
...	
}	

도면54

oop_dpb_maxbuffering_parameters (i, k) {	
sub_layer_vps_buf_info_present_flag[i][k]	u(1)
for(j = (sub_layer_vps_buf_info_present_flag[i][k] ? 0 : max_sub_layers_vps_minus1[k]); j <= max_sub_layers_vps_minus1[k]; j++; j++)	
max_vps_dec_pic_buffering_minus1[i][k][j]	ue(v)
}	

도면55

oop_dpb_maxbuffering_parameters (i, k) {	
sub_layer_vps_buf_info_present_flag[i][k]	u(1)
for(j = (sub_layer_vps_buf_info_present_flag[i] ? 0 : MaxSubLayers[i][k] - 1); j <= MaxSubLayers[i][k] - 1; j++)	
max_vps_dec_pic_buffering_minus1[i][k][j]	ue(v)
}	

도면56

vps_extension() {	디스크립터
while(!byte_aligned())	
vps_extension_byte_alignment_reserved_one_bit	u(1)
...	
for(j = 0; j < NumScalabilityTypes; j++)	
dimension_id[i][j]	u(v)
}	
for(i = 1; i <= vps_max_layers_minus1; i++) {	
max_sub_layers_vps_predict_flag[i]	u(1)
if(!max_sub_layers_vps_predict_flag[i])	
max_sub_layers_vps_minus1[i]	u(3)
}	
...	
profile_level_tier_idx[i]	u(v)
}	
num_dpb_info_parameters	ue(v)
for(i = 0; i < num_dpb_info_parameters; i++) {	
if(i>0 && (MaxSubLayers[i] == MaxSubLayers[i - 1]))	
sub_layer_vps_buf_info_predict_flag[i]	u(1)
if(!sub_layer_vps_buf_info_predict_flag[i]) {	
output_point_layer_set_idx[i]	ue(v)
for(k = 1; k <= vps_max_layers_minus1; i++) {	
oop_dpb_maxbuffering_parameters(i, k)	
}	
}	
}	
for(i = 1; i <= vps_max_layers_minus1; i++) {	
if(max_sub_layers_vps_predict_flag[i])	
sub_layer_vps_ordering_info_predict_flag[i]	u(1)
if(!sub_layer_ordering_info_predict_flag[i])	
layer_dpb_info_parameters(i)	
}	
}	
...	
}	

도면57

oop_dpb_maxbuffering_parameters (i, k) {	
sub_layer_vps_buf_info_present_flag[i][k]	u(1)
for(j = (sub_layer_vps_buf_info_present_flag[i][k] ? 0 : max_sub_layers_vps_minus1[k]); j <= max_sub_layers_vps_minus1[k]; j++); j++)	
max_vps_dec_pic_buffering_minus1[i][k][j]	ue(v)
}	

도면58

oop_dpb_maxbuffering_parameters (i, k) {	
sub_layer_vps_buf_info_present_flag[i][k]	u(1)
for(j = (sub_layer_vps_buf_info_present_flag[i] ? 0 : MaxSubLayers[i][k] - 1); j <= MaxSubLayers[i][k] - 1; j++)	
max_vps_dec_pic_buffering_minus1[i][k][j]	ue(v)
}	

도면59

vps_extension() {	디스크립터
while(!byte_aligned())	
vps_extension_byte_alignment_reserved_one_bit	u(1)
...	
for(j = 0; j < NumScalabilityTypes; j++)	
dimension_id[i][j]	u(v)
}	
for(i = 1; i <= vps_max_layers_minus1; i++) {	
max_sub_layers_vps_predict_flag[i]	u(1)
if(!max_sub_layers_vps_predict_flag[i])	
max_sub_layers_vps_minus1[i]	u(3)
}	
...	
profile_level_tier_idx[i]	u(v)
}	
num_dpb_info_parameters	ue(v)
for(i = 0; i < num_dpb_info_parameters; i++) {	
if(i>0 && (MaxSubLayers[i] == MaxSubLayers[i - 1]))	
sub_layer_vps_buf_info_predict_flag[i]	u(1)
if(!sub_layer_vps_buf_info_predict_flag[i]) {	
output_point_layer_set_idx[i]	ue(v)
for(k = 1; k <= vps_max_layers_minus1; i++) {	
oop_dpb_maxbuffering_parameters(i, k)	
}	
}	
}	
layer_dpb_info_parameters_presence_flag	u(1)
If(layer_dpb_info_parameters_presence_flag) {	
for(i = 1; i <= vps_max_layers_minus1; i++) {	
if(max_sub_layers_vps_predict_flag[i])	
sub_layer_vps_ordering_info_predict_flag[i]	u(1)
if(!sub_layer_ordering_info_predict_flag[i])	
layer_dpb_info_parameters(i)	
}	
}	
}	
}	
}	
...	
}	

도면60

oop_dpb_maxbuffering_parameters(i, k) {	
sub_layer_vps_maxbuffering_info_present_flag[i][k]	u(1)
for(j = (sub_layer_vps_maxbuffering_info_present_flag[i][k] ? 0 : max_sub_layers_vps_minus1[k]); j <= max_sub_layers_vps_minus1[k]; j++)	
max_vps_dec_pic_buffering_minus1[i][k][j]	ue(v)
}	

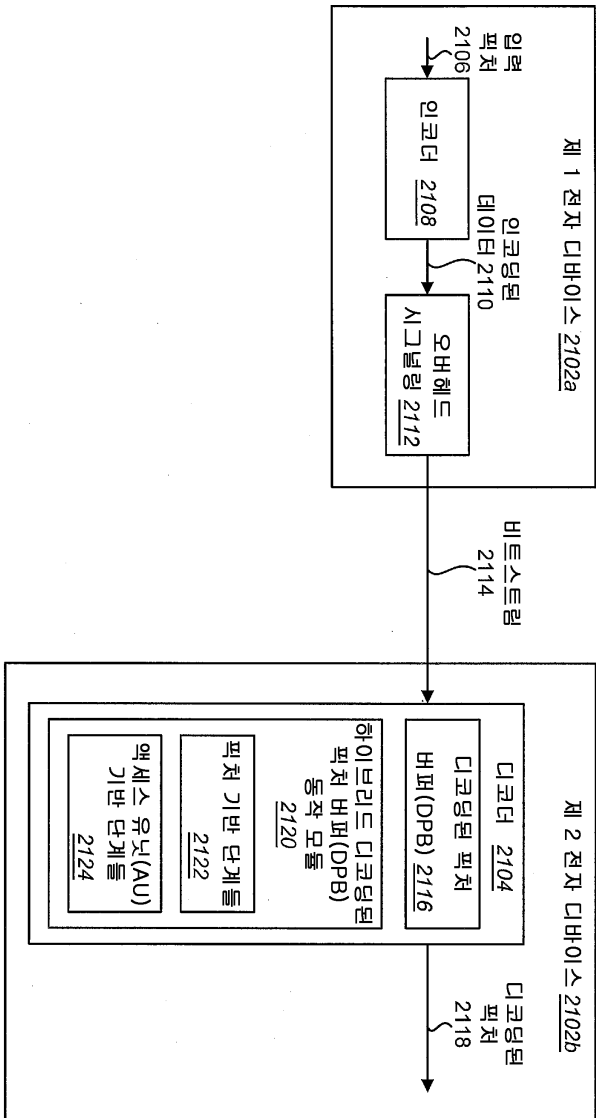
도면61

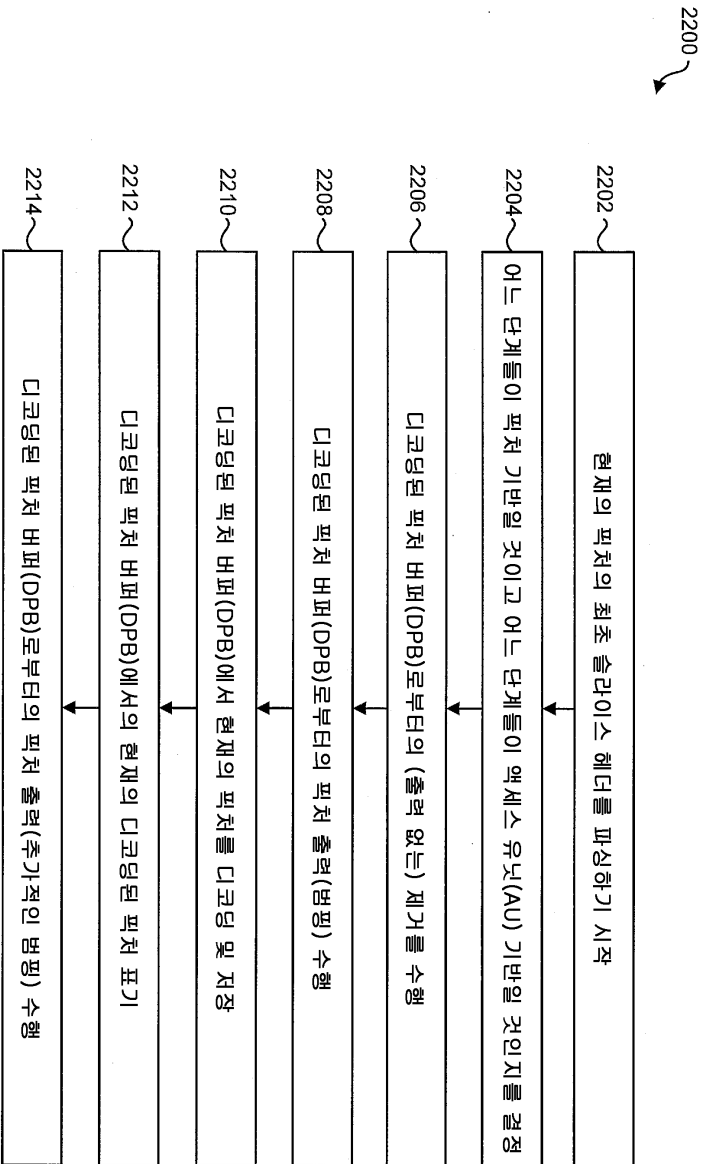
oop_dpb_maxbuffering_parameters(i, k) {	
sub_layer_vps_buf_info_present_flag[i][k]	u(1)
for(j = (sub_layer_vps_buf_info_present_flag[i][k] ? 0 : MaxSubLayers[i][k] - 1); j <= MaxSubLayers[i][k] - 1; j++)	
max_vps_dec_pic_buffering_minus1[i][k][j]	ue(v)
}	

도면62

seq_parameter_set_rbsp() {	디스크립터
sps_video_parameter_set_id	u(4)
sps_max_sub_layers_minus1	u(3)
sps_temporal_id_nesting_flag	u(1)
if(nuh_layer_id == 0)	
profile_tier_level(1, sps_max_sub_layers_minus1)	
sps_seq_parameter_set_id	ue(v)
...	
log2_max_pic_order_cnt_lsb_minus4	ue(v)
if(nuh_layer_id > 0)	
sps_dpb_params_present_flag	u(1)
if((nuh_layer_id == 0) ((nuh_layer_id > 0) && sps_dpb_params_present_flag)) {	
sps_sub_layer_ordering_info_present_flag	u(1)
for(i = (sps_sub_layer_ordering_info_present_flag ? 0 : sps_max_sub_layers_minus1);	
i <= sps_max_sub_layers_minus1; i++) {	
sps_max_dec_pic_buffering_minus1[i]	ue(v)
sps_max_num_reorder_pics[i]	ue(v)
sps_max_latency_increase_plus1[i]	ue(v)
}	
}	
log2_min_luma_coding_block_size_minus3	ue(v)
log2_diff_max_min_luma_coding_block_size	ue(v)
log2_min_transform_block_size_minus2	ue(v)
...	
sps_extension_flag	u(1)
if(sps_extension_flag) {	
sps_extension()	
sps_extension2_flag	u(1)
if(sps_extension2_flag)	
while(more_rbsp_data())	
sps_extension_data_flag	u(1)
}	
rbsp_trailing_bits()	
}	

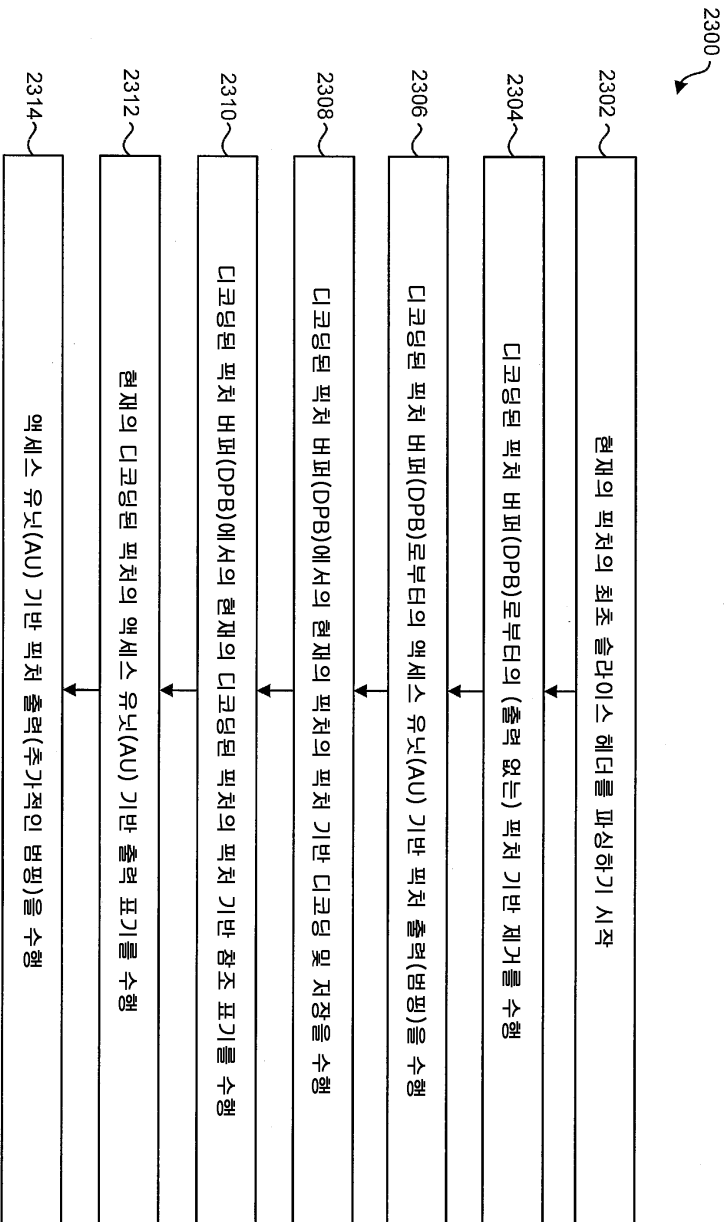
도면63



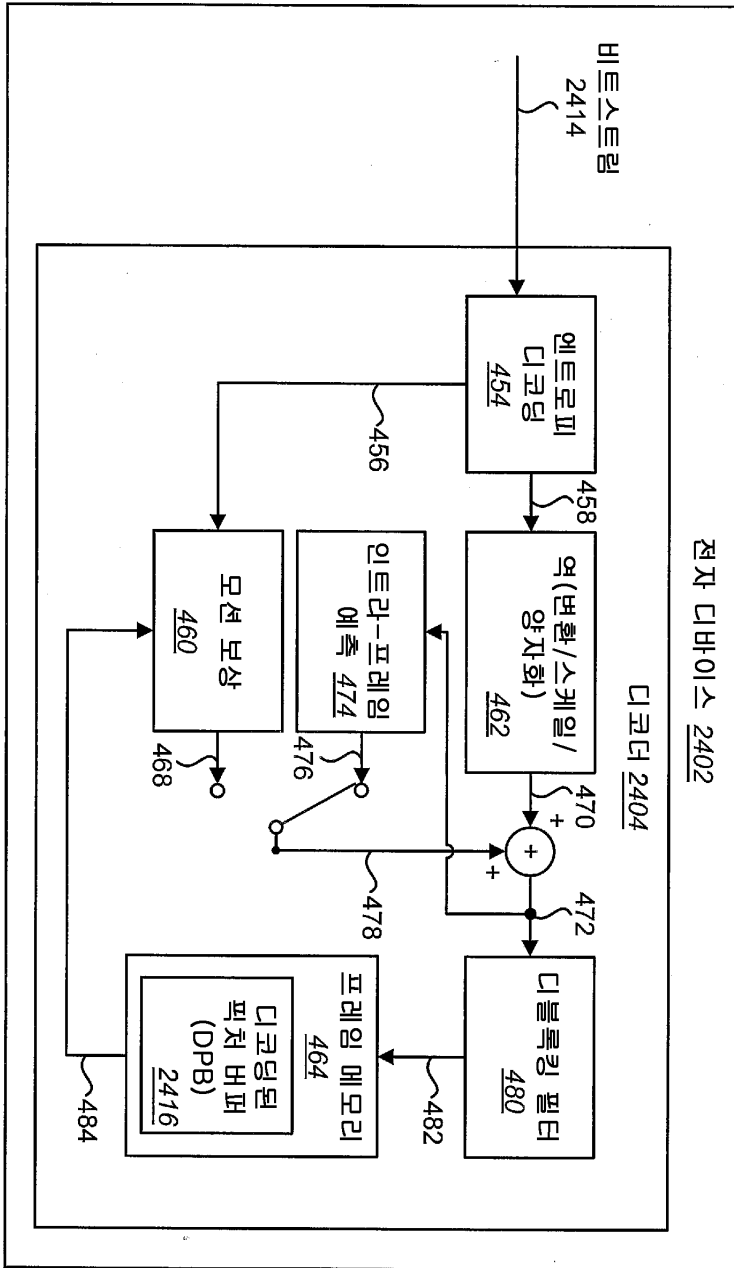


도면64

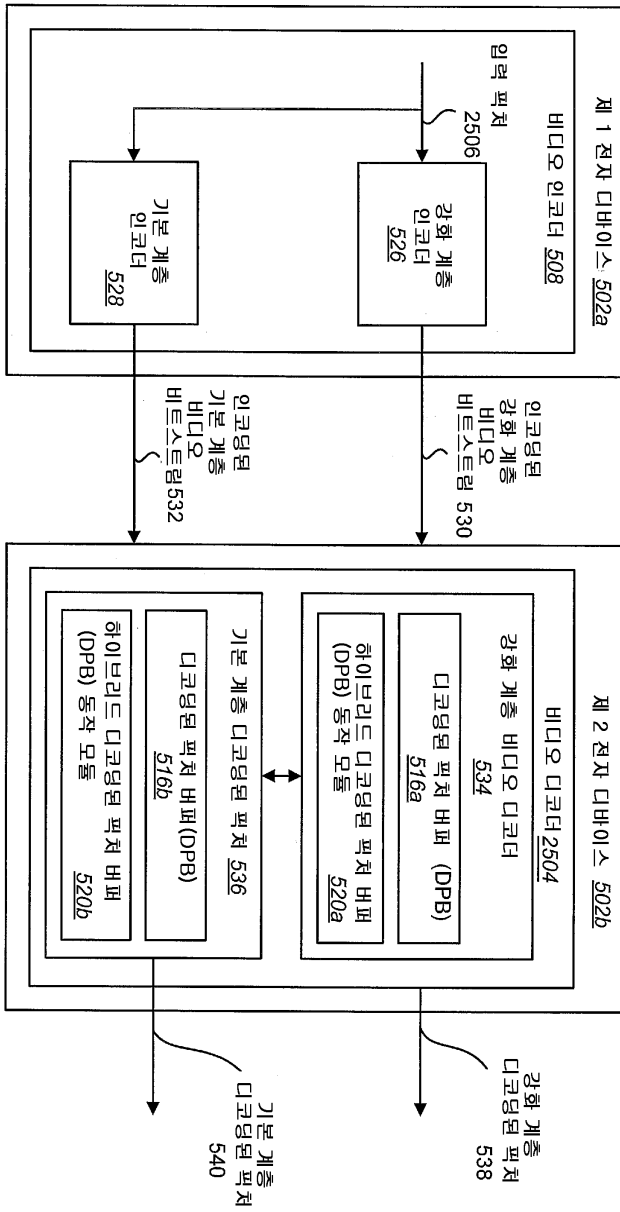
도면65



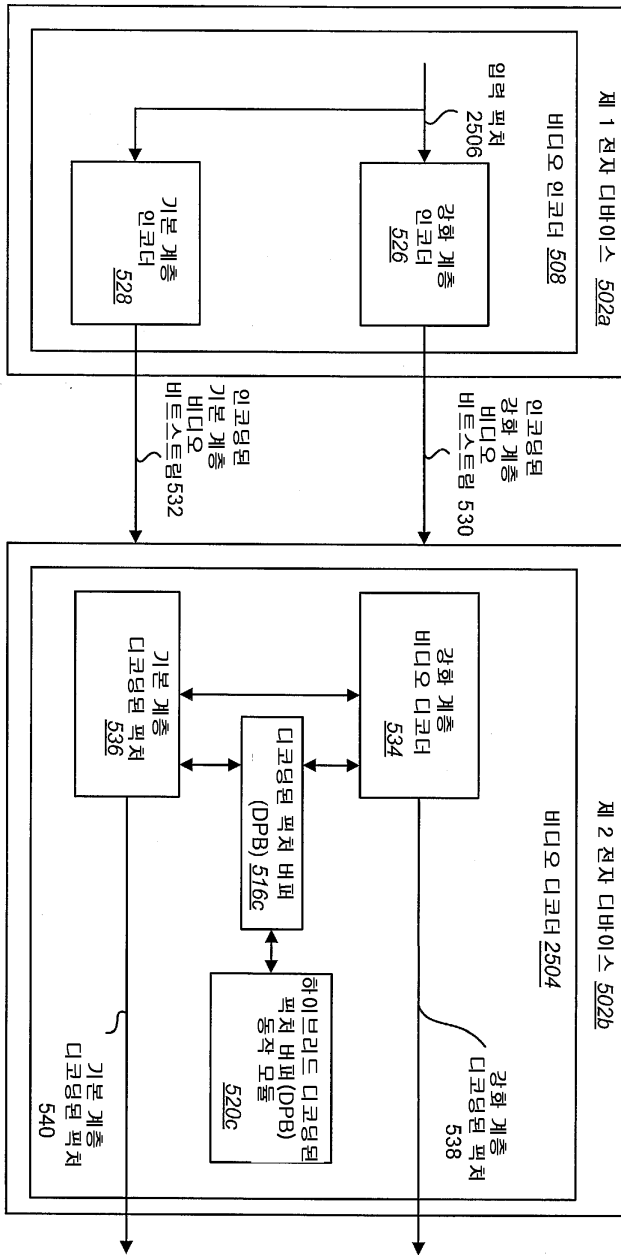
도면66



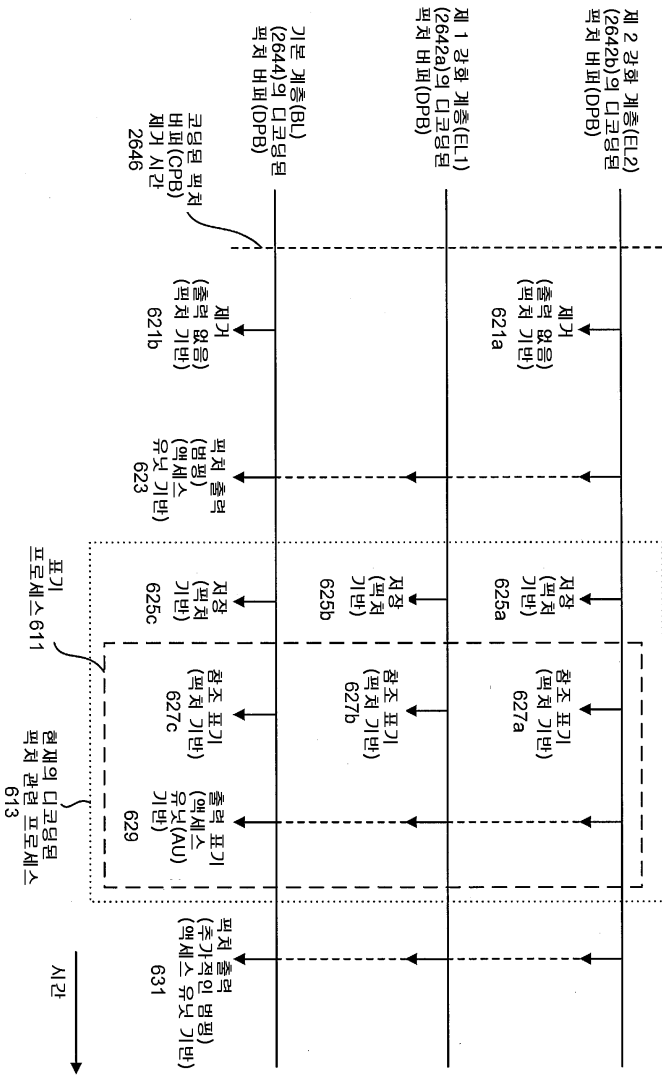
도면67a



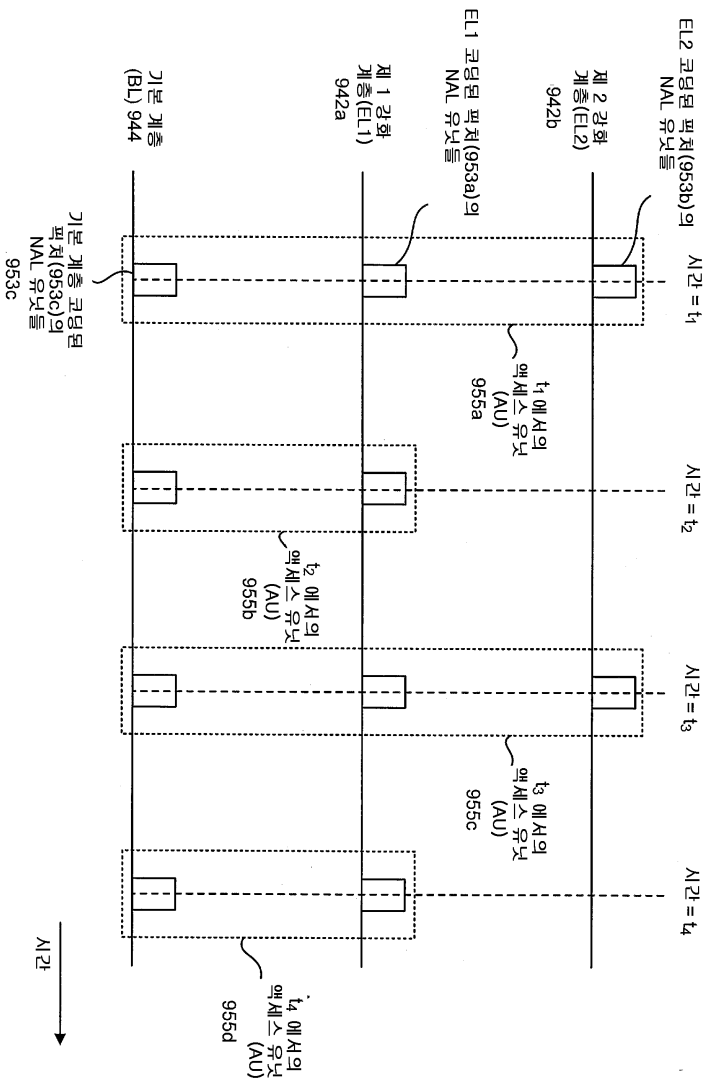
도면67b



도면 68



도면 69



도면70

