(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2007-0105999
(43) 공개일자 2007년10월31일

(51) Int. Cl.
*H04N 7/26* (2006.01) *H04N 7/24* (2006.01)
(21) 출원번호 10-2007-7018642
(22) 출원일자 2007년08월14일
심사청구일자 없음
번역문제출일자 2007년08월14일
(86) 국제출원번호 PCT/US2006/003520
국제출원일자 2006년02월01일
(87) 국제공개번호 WO 2006/088644
국제공개일자 2006년08월24일
(30) 우선권주장
60/653,538 2005년02월16일 미국(US)

(71) 출원인
**톰슨 라이센싱**
프랑스 세데 볼로뉴 께아 르 갈로 46
(72) 발명자
**데 루카, 마이클, 안토니**
미국, 매사추세츠 01803, 버링톤, 스파호크 디파이브 3
(74) 대리인
**김학수, 문경진**

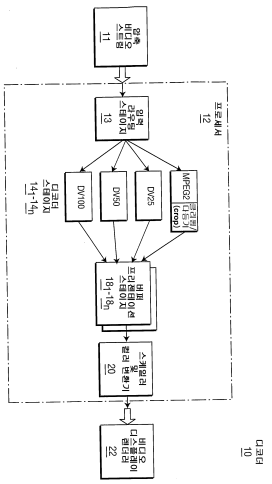
전체 청구항 수 : 총 16 항

## (54) 고속 디코더

### *(57) 요 약*

디코더 장치(10)는 다른 포맷의 복수의 스트림을 포함하는 복수의 스트림($11_1$-$11_n$)을 디코딩하기 위해 프로그래밍된 프로세서(12)을 구비한다. 기능성에 관해서, 이 디코더 장치는 각 스트림을 다른 디코더 스테이지($14_1$-$14_n$)으로 라우팅하는 라우팅 스테이지(13)을 구비하며, 각 디코더 스테이지는 특정 포맷의 스트림을 디코딩하여 출력에서 압축해제 스트림을 생성할 수 있다. 각 복수의 버퍼 스테이지($16_1$-$16_n$)는 관련된 디코더 스테이지에 의해 출력된 압축해제 스트림의 연속적인 프레임을 저장한다. 출력 스테이지는 버퍼 스테이지에 의해 저장된 프레임을 디스플레이 디바이스(22)에 대한 입력을 위한 공통 크기로 스케일링한다.

*대표도* - 도1



## *특허청구의 범위*

**청구항 1**

적어도 제 1 및 제 2 스트림을 디코딩하기 위한 디코더 장치로서,

상기 적어도 제 1 및 제 2 스트림의 각 하나를 각각 디코딩하여 적어도 제 1 및 제 2 압축해제 스트림을 생성하는 적어도 제 1 및 제 2 디코더 스테이지;

상기 적어도 제 1 및 제 2 스트림의 각 하나를 적어도 제 1 및 제 2 디코딩 스테이지로 각기 라우팅하는 라우팅 스테이지;

각기, 상기 제 1 및 제 2 압축해제 스트림의 각 하나의 프레임을 각각 저장하는 적어도 제 1 및 제 2 버퍼 스테이지; 및

상기 제 1 및 제 2 버퍼 스테이지로부터 공통 크기로 프레임을 스케일링하는 스케일러

를 포함하는, 디코딩하기 위한 디코더 장치.

**청구항 2**

제 1 항에 있어서,

상기 적어도 제 1 및 제 2 스트림 각각은 각기 인코딩된 제 1 및 제 2 포맷이고, 상기 적어도 제 1 및 제 2 디코더 스테이지의 각각은 제 1 및 제 2 디코딩 포맷을 가지며, 각각 포맷은 각기 상기 적어도 제 1 및 제 2 스트림의 각기 상기 제 1 및 제 2 포맷에 매칭하는, 디코딩하기 위한 디코더 장치.

**청구항 3**

제 2 항에 있어서,

상기 제 1 인코딩 포맷은 표준화질(SD:Standard Definition) 비디오를 위한 MPEG 2, DV25, DV50 및 고화질 비디오를 위한 MPEG 2, H.264-MPE4 AVC 및 DV100 중 하나를 포함하는, 디코딩하기 위한 디코더 장치.

**청구항 4**

제 2 항에 있어서,

상기 제 2 인코딩 포맷은 표준화질(SD:Standard Definition) 비디오를 위한 MPEG 2, DV25, DV50 및 고화질 비디오를 위한 MPEG 2, H.264-MPE4 AVC 및 DV100 중 하나를 포함하는, 디코딩하기 위한 디코더 장치.

**청구항 5**

제 1 항에 있어서,

상기 적어도 제 1 및 제 2 프리젠테이션 버퍼 스테이지는 다른 크기를 가지는, 디코딩하기 위한 디코더 장치.

**청구항 6**

제 5 항에 있어서,

상기 제 1 및 제 2 프리젠테이션 버퍼 스테이지는 720x480 픽셀, 720x576 픽셀, 1920x1080 픽셀 또는 1280x720 픽셀 중 하나의 크기를 가지는, 디코딩하기 위한 디코더 장치.

**청구항 7**

제 1 항에 있어서,

폐쇄 세마포어(lock semaphore)는 상기 제 1 및 제 2 프리젠테이션 버퍼 스테이지 중 적어도 하나의 각각이 다른 버퍼 스테이지를 침범하는 것을 방지하는, 디코딩하기 위한 디코더 장치.

**청구항 8**

제 1 항에 있어서,

상기 적어도 제 1 및 제 2 프리젠테이션 버퍼 스테이지의 각각은 각 디코더 스테이지가 시간 병렬성(temporal parallism)에서의 동작에 독립적으로 디코딩하는 것을 허용하기 위해 상기 적어도 제 1 및 제 2 디코더 스테이지 중 관련된 하나를 상기 스케일러로부터 연결해제하는, 디코딩하기 위한 디코더 장치.

**청구항 9**

제 1 항에 있어서,

상기 스케일러는 컬러 공간 변환을 수행하는, 디코딩하기 위한 디코더 장치.

**청구항 10**

적어도 제 1 및 제 2 스트림을 디코딩하기 위한 방법으로서,

디코딩하기 위하여 상기 적어도 제 1 및 제 2 스트림의 각 하나를 라우팅하는 단계;

적어도 제 1 및 제 2 압축해제 스트림 각각을 생성하도록 상기 적어도 제 1 및 제 2 스트림의 각 하나를 디코딩하는 단계;

상기 적어도 제 1 및 제 2 압축해제 스트림의 각 하나의 프레임을 저장하는 단계; 및

상기 제 1 및 제 2 압축해제 스트림의 상기 저장된 프레임을 공통 크기로 스케일링하는 단계

를 포함하는, 디코딩하기 위한 방법.

**청구항 11**

제 10 항에 있어서,

상기 적어도 제 1 및 제 2 스트림 각각은 각기 인코딩된 제 1 및 제 2 포맷이고, 상기 디코딩 단계는, 각기 제 1 및 제 2 디코딩 포맷을 사용하여, 상기 적어도 제 1 및 제 2 스트림을 디코딩하는 단계를 더 포함하는, 디코딩하기 위한 방법.

**청구항 12**

제 10 항에 있어서,

상기 제 1 인코딩 포맷은 표준화질(SD:Standard Definition) 비디오를 위한 MPEG 2, DV25, DV50 및 고화질 비디오를 위한 MPEG 2, H.264-MPE4 AVC 및 DV100 중 하나를 포함하는, 디코딩하기 위한 방법.

**청구항 13**

제 10 항에 있어서,

상기 제 2 인코딩 포맷은 표준화질(SD:Standard Definition) 비디오를 위한 MPEG 2, DV25, DV50 및 고화질 비디오를 위한 MPEG 2, H.264-MPE4 AVC 및 DV100 중 하나를 포함하는, 디코딩하기 위한 방법.

**청구항 14**

제 10 항에 있어서,

상기 저장된 스트림의 스케일링으로부터 상기 적어도 제 1 및 제 2 스트림의 인코딩을 연결해제하는 단계를 더 포함하는, 디코딩하기 위한 방법.

**청구항 15**

제 10 항에 있어서,

상기 저장된 프레임에 관해 컬러 변환을 실행하는 단계를 더 포함하는, 디코딩하기 위한 방법.

**청구항 16**

제 10 항에 있어서,

공간 병렬성으로 복수의 스트림을 디코딩하는 단계를 더 포함하는, 디코딩하기 위한 방법.

## 명 세 서

### 기 술 분 야

<1> 본 출원은 35 U.S.C 119(e)하에서, 미국 가특허 출원번호 제60/653538호(2005.2.16일자 출원)에 대한 우선권을 주장하며, 이 출원의 교지는 본 출원에 병합된다.

<2> 본 출원은 압축된 스트림을 디코딩하는 디코더에 대한 것이다.

### 배 경 기 술

<3> 전통적으로, 비디오 디코더는 하드웨어-기반이었다. 달리 말하면, 일반적인 비디오 디코더는 이러한 목적을 위해 전용된 별도, 독립형 디바이스의 형태를 취한다. 하드웨어 디코더 구현으로, 다른 포맷의 디코딩 스트림은 보통 각 스트림 타입 및 비교적 긴 펌웨어 다운로드를 위한 다른 칩셋을 요한다. 다수의 칩셋은 매우 고비용이 될 수 있으며, 펌웨어 변경을 실시간으로 만드는 것은 비현실적이다.

<4> 따라서, 종래 기술의 단점을 극복하는 디코더 장치에 대한 필요가 존재한다.

### 발명의 상세한 설명

<5> 간단하게, 본 발명의 바람직한 실시예에 따라, 적어도 제 1 및 제 2 압축 스트림을 디코딩하기 위한 디코더가 제공된다. 이 디코더는 제 1 및 제 2 스테이지를 포함하며, 각각은 적어도 제 1 및 제 2 스트림 중 각기 하나를 디코딩하여 각각의 제 1 및 제 2 비압축 스트림을 생성할 수 있다. 라우팅 스테이지는 적어도 제 1 및 제 2 스트림을 각기 적어도 제 1 및 제 2 디코더 스테이지로 라우팅한다. 적어도 제 1 및 제 2 버퍼 스테이지는 제 1 및 제 2 디코더 스테이지에 의해 출력된 제 1 및 제 2 비압축 스트림 중 각기 하나를 각각 저장한다. 출력 스테이지는 적어도 제 1 및 제 2 버퍼 스테이지에 의해 저장된 비압축 스트림을 결합한다.

<6> 실제로, 본 발명의 디코더 장치는 소프트웨어로 프로그래밍된 프로세서를 포함하여 디코더 스테이지의 각각에 대한 기능을 수행한다. 소프트웨어형 디코더는 높은 정도의 고객 맞춤을 허용하며, 디코딩 과정에 대한 더 큰 유연성 및 제어를 제공한다. 비-선형 편집기(NLE: Non-Linear Editor)에서 사용된 소스 비디오 재료는 다른 소스로부터 올 수 있으며, 예를 들면, MPEG2, DV, JPG2K와 같은 다른 포맷으로의 압축을 겪을 수 있다. 또한, 재료는, 일례로 표준 화질 NTSC 대 고화질 1080i의 다른 프리젠테이션 크기를 가질 수 있다. 본 발명의 디코더 장치는 다른 포맷을 디코딩할 수 있는 디코더 스테이지를 이용함으로써, 원래의 압축 포맷 또는 이미지 크기에 관계없이 비디오 스트림을 디코딩할 수 있다. 본 발명의 디코더 장치는 다른 포맷의 프레임을 압축해제할 수 있고 비압축 프레임을 공통 "캔버스"로 조직할 수 있다. 추가로, 비교적 저렴한 멀티프로세서 시스템의 유용성으로도, 본 발명의 디코더 장치는 다수의 병렬 쓰레드를 실행하여 프로세싱 처리량은 숫자와 함께 스켈링 될 수 있으며(scale), 이용가능한 프로세서의 속도를 증가시킬 수 있다.

### 실 시 예

<8> 도 1은 본 발명의 바람직한 실시예에 따른 복수의 압축 비디오 스트림($11_1$-$11_n$, 여기서 n은 0보다 큰 정수임)을 압축해제하기 위한 디코더 장치(10)의 개략적인 블럭도를 묘사한다. 실제로, 비디오 스트림($11_1$-$11_n$) 중 하나 이상은 다른 것과 상이한 포맷을 가질 수 있다. 예를 들면, 스트림 중의 하나는 MPEG2 스트림의 형태를 취하는 반면에 다른 스트림은 DV25 또는 DV50스트림의 형태를 취할 수 있다. 스트림의 특정 포맷이 연역적으로 알려져 있는 한, 각 스트림의 특정 포맷은 중요하지 않은 채로 남아있다.

<9> 실제로, 본 발명의 디코더(10)는 이하에서 더 상세히 기술되는 바와 같이 스테이지에서 동작하는 마이크로프로세서, 마이크로컴퓨터 등과 같은 프로그래밍된 프로세서(12)를 포함한다. 소프트웨어형 디코더로서 동작하기 위해 프로그래밍되는 경우, 프로세서(12)는 압축 비디오 스트림($11_1$-$11_n$)을 인입 스트림의 압축 포맷에 의존하는 복수의 디코더 스테이지($14_1$-$14_n$) 중 적합한 하나에 라우팅하는 입력 라우팅 스테이지(13)를 구비한다. 각각의 디코더 스테이지($14_1$-$14_n$)는 특정 포맷의 인입를 압축해제하기 위해 동작한다. 일반적인 압축 포맷은 표준화질(SD:Standard Definition) 비디오를 위한 MPEG2, DV25, 및 DV50와, 고화질(HD:High Definition) 비디오를 위한

MPEG2, H.264-MPEG4 AVC 및 D100을 포함한다. 따라서, 압축 비디오 스트림($11_1$-$11_n$)의 구성에 따라, 디코더 스테이지($14_1$-$14_n$) 중 하나 이상은 MPEG2, DV25 및 DV50 SD 포맷 중의 하나 또는 MPEG2, H.264-MPE4 AVC 및 DV 100 HD 포맷 중 하나에서 비디오를 압축해제하는 능력을 가질 것이다.

<10> 특정 압축 포맷의 스트림($11_1$-$11_n$)중의 인입하는 하나를 수신하면, 바로 그 포맷과 관련된 디코더 스테이지는 이 스트림을 디코딩하여 연속적인 압축해제 프레임을 생성할 것이다. 디코더 스테이지에 의해 출력된 각 압축해제 프레임은, 압축해제 프레임을 수신하기 위해 일정 크기로 분할된, 각기 프리젠테이션 스테이지($18_1$-$18_n$) 중의 대응하는 하나에서 저장을 겪는다. 실제로, 각 프리젠테이션 스테이지($18_1$-$18_n$)은 관련된 디코딩 스테이지의 압축해제 포맷에 의존하는 특정 크기의 프레임을 유지한다. 일반적으로, 각 프리젠테이션 버퍼 스테이지($18_1$-$18_n$)는 다음의 표준 크기 중 하나를 가질 것이다:

<11> NTSC(720x480)

<12> PAL(720x576)

<13> 1080i(1920x1080)

<14> 720p(1280x720)

<15> 압축해제된 SD 프레임은 일반적으로 NTSC 또는 PAL 포맷에 따를 것이다. 크기에 의존하여, 압축해제된 HD 프레임은 1080i 또는 720p HD 프레임을 수용할 수 있는 버퍼 스테이지에서 저장을 겪을 것이다. 만일 디코딩 프레임 이미지가 프리젠테이션 버퍼 스테이지의 크기에 따르지 않으면, 일부 MPEG2 비디오 프레임으로 발생하는 바와 같이, 프레임은 클리핑(clipping) 또는 다듬기(cropping) 동작을 겪는다. 예를 들면, 720x512 MPEG2 프레임은 프리젠테이션 스테이지에 놓이기 앞서 32 라인을 가질 것이다.

<16> 스케일러(20)는 각 프리젠테이션 버퍼 스테이지($18_1$-$18_n$)로부터 읽어들인 프레임을 스케일(scale)하며, 또한 비디오 디스플레이 렌더러(22)와 관련된 필요한 컬러 공간 변환을 실행한다. 예를 들면, 만일 비디오 디스플레이 렌더러(22)가 1080i HD 프레임과 관련된 1920x1080 픽셀의 입력 프레임 크기를 가진다면, 스케일러(20)는 모든 프레임을 그 크기로 스케일링할 것이다. 1920x1080 픽셀보다 더 작은 임의의 프레임은 스케일러(20)에 의해 상향-변환을 겪을 것이다. 역으로, 720x480 픽셀의 입력 프레임 크기를 가지는 경우, 이후 스케일러(20)는 더 큰 크기 프레임으로 하향변환할 것이다.

<17> 도 1의 디코더 장치(10)는 시간적 병렬성(temporal parallelism)을 제공함으로써 효율성을 개선한다. 소트웨어-기반 디코더 장치로, 압축 비디오 스트림을 읽는 것은 디스크 액세스로 소비된 시간을 수반할 수 있다. 추가적으로, 비압축 출력 스트림의 프레임 쓰기는 일반적으로 디스플레이 하드웨어에 대한 버스 전송을 실행하는데 소비된 시간에 수반할 것이다. 발생할 이러한 입/출력(I/O) 태스크(task)를 완성하기 위한 대기는 프로세싱 사이클의 효율적인 사용을 만들어 내지 못한다. 본 발명의 디코더 장치(10)는 스케일링 및 컬러 공간 변환을 실행하는 스케일러(20)로부터 디코딩 스테이지($14_1$-$14_n$)을 결합해제하기 위해 프리젠테이션 버퍼($18_1$-$18_n$)를 사용한다. 프레임(N)이 프리젠테이션 버퍼에서의 디코딩 및 저장을 겪은 이후, 디코더 스테이지는 스케일러(20)가 출력을 위한 현재의 압축해제된 프레임(N)을 포맷함에따라, 다음 프레임(N+1)을 디코딩할 수 있다. 폐쇄 세마포어(lock semaphore)는 각 프리젠테이션 버퍼 스테이지를 보호하여 하나의 쓰레드가 다른 쓰레드를 침범하는 것을 방지한다.

<18> 위에서 기술한 바와 같이, 비록 반드시 직렬 방식일 필요는 없지만, 일반적으로 디코더 장치(10)는 복수의 스트림($11_1$-$11_n$)를 디코딩하기 위한 소프트웨어 기반 프로그램의 실행용 프로세서(12)를 이용한다. 비록 도 1에서 프로세서(12)가 디코딩 태스크에 전용되는 것으로 기술되어 있지만, 이 프로세서는 디코딩 이외에도 다른 동작을 수행할 수 있을 것이다. 비록 도 1에 명백하게 묘사되어 있지않지만, 대규모 시스템에 있는 다른 프로세서와 네트워킹된 경우, 프로세서(12)는 이용가능한 다른 프로세서의 갯수를 탐지할 수 있다. 만일 프로세서(12) 이외에 또 다른 프로세서가 이용가능하게 된다면, 공간 병렬성(spatial parallelism)이 자동적으로 가능하게 된다. 대부분의 비디오 압축 알고리즘은 매크로블럭이라 불리는 일련의 더 작은 부화상(sub-picture)를 이용하여 개별 프레임을 처리한다. 매크로블럭의 총갯수는 복수의 실행 쓰레드를 생성함으로써, 이용가능한 프로세서 사이에서 분할될 수 있다. 만일 각 프로세서용 블럭의 갯수가 디코딩을 요하는 블럭의 총갯수보다 적으면, 각 프레임을 디코딩하기 위해 요구되는 시간은 더 적게 된다. 이러한 방법을 사용하는 경우, 각 실행 쓰레드는 이미지 티어

링(image tearing)을 방지하기 위해 각 프레임의 끝에서 동기화된다.

<19> 소프트웨어 기반 디코딩 장치(10)는 독립 하드웨어형 디코더에 비교되는 여러 이점을 제공한다. 예를 들면, 디코더 장치(10)는 새로운 압축 표준을 지원하도록 쉽게 확장할 수 있는데, 필요한 모든 것은 소프트웨어 갱신이므로, 새로운 압축 표준이 이용가능해지기 때문이다. 추가로, 개별 제 3 당사자 코덱(CODEC) 구성소자는 더 빠르거나 더 좋은 대체품이 발견된다면 대체될 수 있다. 더욱이, 디코더 장치(10)는 일반적으로 NTSC율에서 매 33ms 및 PAL율에서 매 40ms마다 비디오 프레임 경계에 관한 압축 형태를 전환하여 모든 지원된 압축 포맷를 위한 끊임없는 디코딩 동작을 제공할 수 있다.

<20> 전기한 것은 다른 포맷의 비디오 스트림을 디코딩하기 위한 디코더 장치를 기술한다.

## *산업상 이용 가능성*

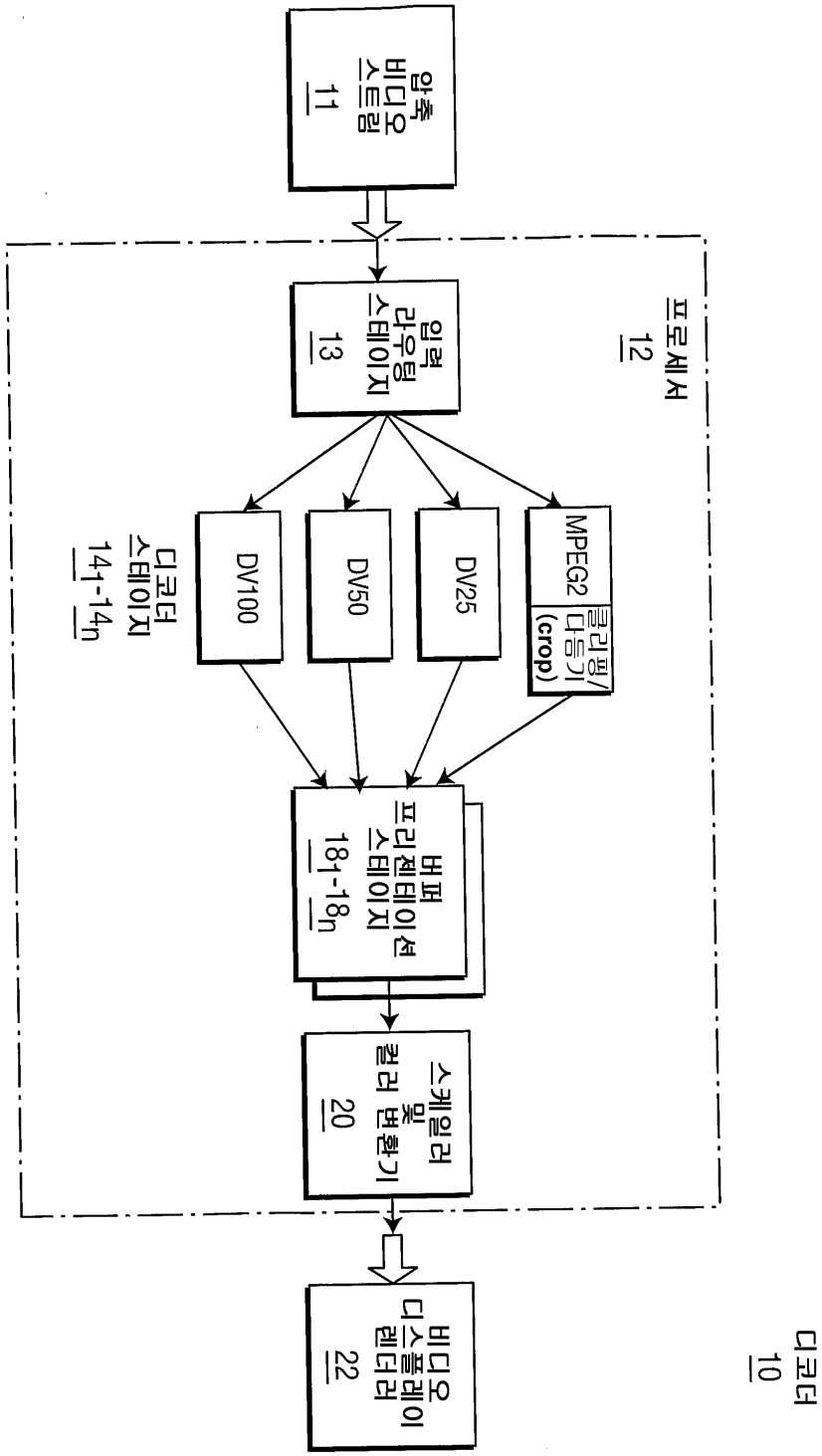
<21> 본 출원은 압축된 스트림을 디코딩하는 디코더에 이용가능하다.

## *도면의 간단한 설명*

<7> 도 1은 본 발명의 바람직한 실시예에 따른 디코더 장치의 개략적인 블럭도를 묘사한 도면.

도면

도면1


압축 비디오 스트림 11
프로세서 12
입력 라우팅 스테이지 13
디코더 스테이지 14₁-14ₙ
DV100
DV50
DV25
MPEG2
클리핑/다듬기 (crop)
버퍼 프리젠테이션 스테이지 18₁-18ₙ
스케일러 및 컬러 변환기 20
비디오 디스플레이 레더러 22
디코더 10