(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
11 April 2024 (11.04.2024)

WIPO | PCT

(10) International Publication Number
WO 2024/076828 A1

(51) International Patent Classification:
G10L 19/008 (2013.01) *G10L 19/16* (2013.01)

(21) International Application Number:
PCT/US2023/074310

(22) International Filing Date:
15 September 2023 (15.09.2023)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
63/378,497 05 October 2022 (05.10.2022) US
63/578,534 24 August 2023 (24.08.2023) US

(71) Applicants: **DOLBY LABORATORIES LICENSING CORPORATION** [US/US]; 1275 Market Street, San Francisco, California 94103 (US). **DOLBY INTERNATIONAL AB** [SE/IE]; 77 Sir John Rogerson's Quay, Block C, Grand Canal Docklands, Dublin, D02 VK60 (IE).

(72) Inventors: **KJOERLING, Kristofer**; c/o Dolby Laboratories, Inc., 1275 Market Street, San Francisco, California 94103 (US). **PURNHAGEN, Heiko**; c/o Dolby Laboratories, Inc., 1275 Market Street, San Francisco, California 94103 (US). **SEEFELDT, Alan J.**; c/o Dolby Laboratories, Inc., 1275 Market Street, San Francisco, California 94103 (US). **LANDO, Joshua B.**; c/o Dolby Laboratories, Inc., 1275 Market Street, San Francisco, California 94103 (US).

(74) Agent: **ANDERSEN, Robert L.** et al.; Intellectual Property Group, Dolby Laboratories, Inc., 1275 Market Street, San Francisco, California 94103 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CV, CZ, DE, DJ, DK, DM,

(54) Title: METHOD, APPARATUS, AND MEDIUM FOR ENCODING AND DECODING OF AUDIO BITSTREAMS WITH PARAMETRIC FLEXIBLE RENDERING CONFIGURATION DATA

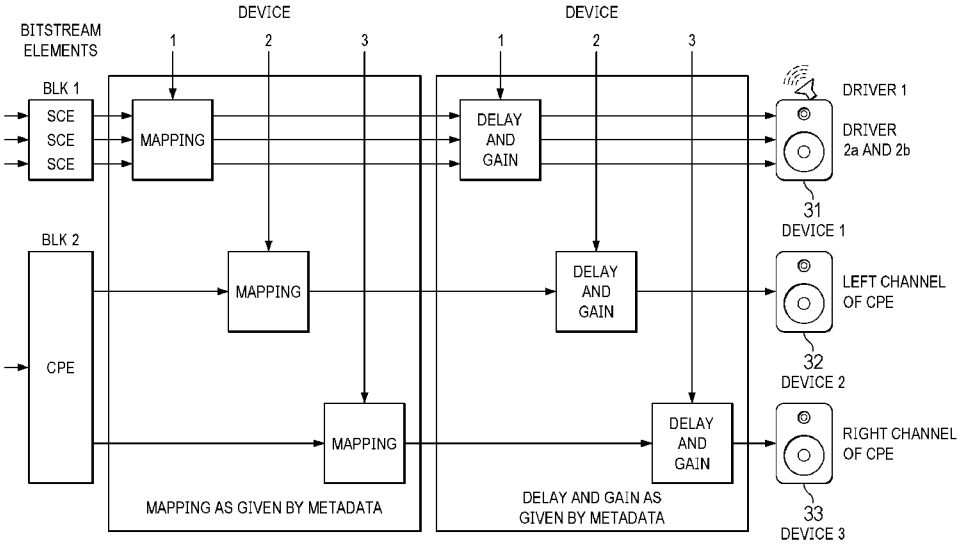

FIG. 7

(57) Abstract: A method for generating an encoded bitstream from an audio program comprising a plurality of audio signals, the method comprising receiving, for each of the plurality of audio signals, information indicating a playback device with which the respective audio signal is associated, receiving, for each playback device, information indicating at least one of a delay, a gain, and an equalization curve associated with the respective playback device, determining, from the plurality of audio signals, a group of two or more related audio signals, applying one or more joint-coding tools to the two or more related audio signals of the group to obtain jointly-coded audio signals, and combining the jointly-coded audio signals, an indication of the playback devices with which the jointly-coded audio signals are associated, and indications of the delay and the gain associated with the respective playback devices with which the jointly-coded audio signals are associated, into an independent block of an encoded bitstream.

WO 2024/076828 A1

[Continued on next page]

DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, MG, MK, MN, MU, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, ZA, ZM, ZW.

(84) **Designated States** *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, CV, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SC, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, ME, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

METHOD, APPARATUS, AND MEDIUM FOR ENCODING AND DECODING OF AUDIO BITSTREAMS WITH PARAMETRIC FLEXIBLE RENDERING CONFIGURATION DATA

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application claims the benefit of priority from U.S. Provisional Application Serial No. 63/378,497 filed on 5 October 2022 and U.S. Provisional Application Serial No. 63/578,534 filed on 24 August 2023, both of which are incorporated by reference herein in their entirety.

TECHNICAL FIELD

[0002] This disclosure relates generally to audio signal processing, and more specifically to audio source coding and decoding for low latency interchange of audio signals of immersive audio programs between devices.

BACKGROUND

[0003] Streaming of audio is common in today's society. The audio streaming is becoming more and more demanding with the users' expectations rising on quality but also the user's setup becoming more complex with a number of speakers but also types of speakers. The streaming is normally done at some part at least over a wireless link which then puts some requirement on the wireless link to have good quality and as many probably have experience, this is not always the case.

[0004] Therefore, there is a need to define an interchange format for use-cases where a certain format is streamed from the cloud/server and subsequently on a device transcoded to a more suitable low latency format for distribution over wireless (or, in some cases wired) links. Exemplary use-cases are in-home connectivity, as well as phone-to-automotive connectivity, although the format may be beneficial in any scenario where low latency distribution of audio signals from a single device to a one or more connected devices is desired.

[0005] In addition to the audio information being sent wirelessly and streamed there could also be other types of information being incorporated into the stream. Such other type of information will then also be affected by the quality of the wireless link and might have similar drawbacks as for the audio.

[0006] It would therefore be advantageous to overcome problems associated with wireless streaming for different types of streamed audio combined with other types of information or signals.

BRIEF SUMMARY

[0007] An object of the present disclosure is to overcome the above problem at least partly with wireless streaming of audio and other types of information.

[0008] According to a first aspect of the disclosure is a method for generating an encoded bitstream from an audio program comprising a plurality of audio signals, the method comprising, receiving, for each of the plurality of audio signals, information indicating a playback device with which the respective audio signal is associated, receiving, for each playback device, information indicating at least one of a delay, a gain, and an equalization curve associated with the respective playback device, determining, from the plurality of audio signals, a group of two or more related audio signals, applying one or more joint-coding tools to the two or more related audio signals of the group to obtain jointly-coded audio signals, and combining the jointly-coded audio signals, an indication of the playback devices with which the jointly-coded audio signals are associated, and the information indicating at least one of a delay, a gain, and an equalization curve associated with the respective playback devices with which the jointly-coded audio signals are associated, into an independent block of an encoded bitstream.

[0009] According to a second aspect of the disclosure is a method for decoding one or more audio signals associated with a playback device from a frame of an encoded bitstream, wherein the frame comprises one or more independent blocks of encoded data, the method comprising, identifying, from the encoded bitstream, an independent block of encoded data corresponding to the one or more audio signals associated with the playback device, extracting, from the encoded bitstream, the identified independent block of encoded data, determining that the extracted independent block of encoded data includes two or more jointly-coded audio signals, applying one or more joint-decoding tools to the two or more jointly-coded audio signals to obtain the one or more audio signals associated with the playback device, determining, from the extracted independent block of encoded data, at least one of a delay, a gain, and an equalization curve associated with the playback device, applying the delay, gain, and/or equalization curve associated with the playback device to the one or more audio signals associated with the playback device.

[0010] According to a third aspect of the disclosure is an apparatus configured to perform a method according the first and/or second aspect.

[0011] According to a fourth aspect of the disclosure is a non-transitory computer readable storage medium comprising a sequence of instructions which, when executed, cause one or more devices to perform the method of any one of a method according the first and/or second aspect.

[0012] Further examples of the disclosure are defined in the dependent claims.

[0013] In some examples, a delay, gain, and/or equalization curve associated with a respective playback device depend on a location of the respective playback device relative to a location of a listener, and/or the respective playback device relative to locations of other playback devices. In some examples, a delay, gain, and/or equalization curve are dynamically variable.

[0014] In some examples, a delay, gain, and/or equalization curve are adjusted in response to a change in the location of the listener, to a change in the location of the playback device, and/or to a change in the location of one or more of other playback devices.

[0015] In some examples is a method comprising determining, from a plurality of audio signals, an audio signal which is not part of a group of two or more related audio signals.

[0016] In some examples is a method comprising, for an audio signal which is not part of a group of two or more related audio signals, applying a delay, gain, and/or equalization curve associated with a playback device with which the audio signal is associated.

[0017] In some examples is a method comprising independently coding an audio signal which is not part of a group of two or more related audio signals, and combining the independently coded audio signal and an indication of a playback device with which the independently coded audio signal is associated into a separate independently decodable subset of an encoded bitstream.

[0018] In the present disclosure, a frame represents a time slice of the entirety of all signals. A block stream represents a collection of signals for the duration of a session. A block represents one frame of a block stream. For digital audio with a given sampling frequency, frame size is equivalent to the number of audio samples in a frame for any audio signal. The frame size usually stays constant for the duration of a session.

[0019] A wake word may comprise one word, or a phrase comprising two or more words in a fixed order.

[0020] Throughout this disclosure including in the claims, the expression "system" is used in a broad sense to denote a device, system, or subsystem. For example, a subsystem that implements a decoder may be referred to as a decoder system, and a system including such a subsystem (e.g., a system that generates X output signals in response to multiple inputs, in which the subsystem generates M of the inputs and the other X – M inputs are received from an external source) may also be referred to as a decoder system.

BRIEF DESCRIPTION OF THE FIGURES

[0021] Examples of the present disclosure will be described in more detail with reference to the appended drawings. In the following drawings like reference numbers are used to refer to like elements. Although the following figures depict various examples, one or more implementations are not limited to the examples depicted in the figures.

[0022] FIG. 1 illustrates an example of in-home connectivity low latency transcoding

[0023] FIG. 2 illustrates an example of automotive connectivity audio streaming

[0024] FIG. 3 illustrates an example of augmented TV audio streaming by use of wireless devices

[0025] FIG. 4 illustrates an example of audio streaming with a simple wireless speaker

[0026] FIG. 5 illustrates an example of metadata mapping from bitstream elements to a device

[0027] FIG. 6 illustrates an example of how to deploy a simple flexible rendering of audio streaming

[0028] FIG. 7 illustrates an example of mapping of flexible rendering data to bitstream elements

[0029] FIG. 8 illustrates an example of signaling of echo-references when playing audio and listening to commands with multiple devices

[0030] FIG. 9 illustrates an example of how frames, blocks, and packets are related to each other

[0031] FIG. 10 illustrates an example of further codec information in the current MPEG-4 structure

[0032] FIG. 11 illustrates an example of even further codec information in the current MPEG-4 structure

[0033] FIG. 12 illustrates an example of integrating listening capabilities and voice recognition

[0034] FIG. 13 illustrates an example of a frame comprising multiple blocks

[0035] FIG. 14 illustrates an example of a bitstream comprising multiple blocks

[0036] FIG. 15 illustrates an example of blocks having different priority

[0037] FIG. 16 illustrates an example of a bitstream comprising multiple blocks with different priority

[0038] FIG. 17 illustrates an example of a frame having different priorities

[0039] FIG. 18 illustrates an example of a bitstream comprising frames having different priorities

DETAILED DESCRIPTION

[0040] The principles of the present invention will now be described with reference to various examples illustrated in the drawings. It should be appreciated that the depiction of these examples is only to enable those skilled in the art to better understand and further implement the present invention; it is not intended for limiting the scope of the present invention in any manner.

[0041] In Figure 1, an immersive audio stream is streamed from the cloud or server 10, and decoded on the TV or Hub device 20. The immersive audio stream may be coded in any existing format, including, for example, Dolby Digital Plus, AC-4, etc. The output is subsequently transcoded into a low latency interchange format for further transmission to connected devices 30, preferably connected over a local wireless connection, e.g., WiFi soft access point, or a Bluetooth connection. Low latency usually depends on various factors such as for example frame size, sampling rate, hardware and/or software computational resources, etc. but low latency would normally be less than 40ms, 20ms or 10ms.

[0042] In another typical use-case a phone fetches the immersive audio stream from the cloud or a server and transcodes to the interchange format and subsequently transmits to a connected car. In the illustrated example of Figure 2, a mobile device (e.g., a phone or tablet) 20 connects to a server 10 and receives an immersive audio stream, performs a low latency transcode to a low latency interchange format, and transmits the transcoded signal to a car 30 which supports immersive audio playback. An example of an immersive audio stream is a stream which includes audio in the Dolby Atmos format, and an example of a car 30 which supports immersive audio playback is a car configured to play back Dolby Atmos immersive formats.

[0043] In general, the interchange format preferably has, low latency, low encode and decode complexity, an ability to scale to high quality, and reasonable coding efficiency. The format preferably also supports configurable latency, so that latency can be traded against efficiency, and also error resilience, to be operable under varying connectivity conditions.

HUB OR AUGMENTED DISPLAY WITH WIRELESS SPEAKERS WITH OR WITHOUT LISTEN CAPABILITY

[0044] Illustrated in figure 3 is an example of a Hub 20 driving a set of wireless speakers 30, or a display (e.g., a television, or TV) 20, possibly with built-in speakers 30 is augmented with several wireless speakers 30. The augmentation suggests that, in examples where the display 20 includes speakers 30, the display 20 is part of the audio reproduction as well. The wireless speakers/devices 30 may be receiving the same complete signal (Broadcast Mode, illustrated to the left in Figure 3) or an individual stream tailored for a specific device (Unicast-multipoint Mode, illustrated to the right in Figure 3).

[0045] Each speaker may include multiple drivers, covering different frequency ranges of the same channel, or corresponding to different channels in a canonical mapping. For example, a speaker may have two drivers, one of which may be an upwards firing driver which outputs a signal corresponding to a height channel to emulate an elevated speaker. The wireless devices may have listening-capability, (e.g., "smart speakers"), and accordingly may require echo-management, which may require the speaker to receive one or more echo-references. The echo-references may be local (e.g., for the same speaker/device), or may instead represent relevant signals from other speakers/devices in the vicinity.

[0046] The speakers may be placed in arbitrary locations, in which case a so-called "flexible rendering" may be performed, whereby the rendering takes the actual position of the speakers into account (e.g., as opposed to a canonical assumption, whereby the loudspeakers are assumed

to be located at fixed, predefined locations). The flexible rendering may take place in the Hub or TV, where subsequently the rendered signals are sent to the speakers/devices, either in a broadcast mode, where each of the rendered signals is sent to each device, and each respective device extracts and output the appropriate signal, or as individual streams to individual devices. Alternatively, the flexible rendering may take place locally on each device, whereby each device receives a representation of the complete immersive program, e.g., a 7.1.4 channel based immersive representation, and from that representation renders an output signal appropriate for the respective device.

[0047] The wireless devices may be connected to the Hub or TV through Soft Access Point provided by the device, or through a local access point in the home. This may impose different requirements on bitrate and latency.

MOBILE DEVICE PROJECTION IN AUTOMOTIVE APPLICATIONS

[0048] In another use-case, a mobile device (e.g., a phone or tablet) 20 fetches the immersive audio stream from the cloud or a server 10, transcodes the immersive audio stream to the interchange format, and subsequently transmits the transcoded stream to a connected car 30. In the example of Figure 2, a Dolby Atmos-enabled phone 20 connects to a server 10 and receives a Dolby Atmos stream, for low latency transcoding and transmission to a Dolby Atmos-enabled car 3. In this use-case, higher bitrates may be available than in living room use-cases, and also there may be different characteristics of the wireless channel. Possible reasons for the higher bitrate in the car use-case is due to less noisy wireless environment since the car acts as a kind of a faraday cage and shields the inside car environment from outside wireless disturbance, compared to the living room where all wireless devices such as from neighbors, other rooms and so on are adding noise to the wireless environment. At the same time there are normally very few wireless devices in the car that are competing for the available wireless bandwidth, compared to in the living-room use case where normally all the wireless devices in the household are competing for the available bandwidth, including the living room wireless devices.

[0049] For this use-case it is envisioned that the signal to be transcoded by the mobile device and transmitted to the car may be a channel based immersive representation, an object-based representation, a scene-based representation (e.g., and Ambisonics representation), or even a combination of different representations. For this example, the different rendering architectures and broadcast vs. multipoint may not be relevant, as generally the complete presentation will be transferred from the mobile device to a single end-point (e.g., the car).

DESCRIPTION OF THE INTERCHANGE FORMAT

[0050] The immersive interchange format is built on a modified discrete cosine transform (MDCT) with perceptually motivated quantization and coding. It has configurable latency, e.g., support for different transform sizes at a given sampling rate. Exemplary frame sizes are 128, 256, 512, 1024 and 120, 240, 480, 960 and 192, 384, 768 samples at sampling rates of 48kHz and 44.1kHz.

[0051] The format may support mono, stereo, 5.1 and other channel configurations, including immersive channel configurations (e.g., including, but not limited to, 5.1.2, 5.1.4, 7.1.2, 7.1.4, 9.1.6, and 22.2, or any other channel configuration consisting of unique channels as specified in ISO/IEC 23091-3:2018, Table-2). The format may also support object-based audio and scene-based representations, such as Ambisonics (e.g., first order or higher-order). The format may also use a signaling scheme that is suitable for integration with existing formats (e.g., such as those described in ISO/IEC 14496-3, which may also be referred to as the MPEG-4 Audio Standard, ISO 14496-1 which may also be referred to as the MPEG-4 Systems Standard, ISO 14496-12 which may also be referred to as the ISO Base Media File Format Standard and/or ISO 14496-14 which may also be referred to as the MP4 File Format Standard).

[0052] Furthermore, the system may have support for the ability to skip parts of the syntax elements and only decode relevant parts for a given speaker, support for metadata controlled flex-rendering aspects such as delay alignment, level adjustment, and equalization, support for the use of smart-speakers (including the scenario where a set of signals is sent that allows to feed each of the driver/loudspeaker in a smart speaker independently) with listening capability, and associated support for echo-management by signaling of echo-references, support for quantization and coding in the MDCT domain with both low overlap windows as well as 50% overlap windows, and/or support for filtering along the frequency axis in the MDCT domain to do temporal shaping of quantization noise, e.g., TNS (Temporal Noise Shaping). Thus, in various examples the overlap windows are symmetric or asymmetric.

[0053] The disclosed interchange format may in some examples provide for improved joint channel coding, to take advantage of increased correlation between signals in the flexible rendering use-case, improved coding efficiency by shared scale factors across channel elements, inclusion of high frequency reconstruction and noise addition techniques in the MDCT domain, assorted improvements over legacy coding structures to allow for better efficiency and suitability to the use-cases at hand.

[0054] In some examples of controlling individual playback devices in broadcast mode, skippable blocks and metadata may be used. In a broadcast mode setting, each wireless device will need to play the relevant part of the complete audio for the specific drivers/channels that correspond to the specific device. This implies that the device needs to know what parts of the stream are relevant for that device, and extract those parts from the complete audio stream being broadcast. In order to enable low complexity decoding operations, it is preferable if the stream is constructed in a manner so that the decoder for the specific device can efficiently skip over elements that are not relevant for decoding to the elements that are relevant for the device and the given drivers on the device.

[0055] However, it should be noted that there may be benefits (from a compression perspective) to still allow for joint coding between signals that are destined for different devices (e.g., in the most simplistic scenario the left and right channel of a stereo presentation).

[0056] Hence, the format may: enable "skippable blocks" in the bitstream to enable efficient decoding of the parts only relevant for the specific device, include metadata that enables flexible mapping of one or more skippable blocks to a specific device, while retaining the ability to apply joint coding techniques between signals corresponding to different speakers/devices.

[0057] In Figure 4, an example of an arbitrary set-up is shown. There are three connected wireless devices 31,32,33, of which two are single channel speakers 32,33 (meaning one channel of audio) and one speaker 31 is a more advanced speaker 31 with three different drivers and thereby operating on three different signals. The first speaker 31 operates on 3 individual signals in this example, while the second and third speakers 32,33 operate on a stereo representation (e.g., Left and Right). As such, it may be beneficial to do joint coding of the signals of the stereo representation.

[0058] Given the above scenario, the format specifies a bitstream containing skippable blocks so that the first device can extract only the relevant parts of the stream for decoding the signals for that speaker, while decoder complexity vs. efficiency may be traded off for the stereo pair by constructing a signal where each speaker needs to decode two signals to output a single signal, with the upside that joint coding can be performed.

[0059] The format specifies a metadata format to enable a general and flexible representation which maps specific ones of a plurality of skippable blocks to one or more devices. This is illustrated in Figure 5, where the mapping may be represented as a matrix that associates each

device 31,32,33 to one or more bitstream elements, so that the decoder for a given device will know what bitstream elements to output and decode.

[0060] For instance, in the example of Figure 5, a first block or skip block Blk1 includes 3 single channel elements (one for each driver of device 1 31), while a second block or skip block Blk2 contains a channel pair element, which may include jointly-coded versions of the signals to be output by Devices 2 32 and 3 33. Device 1 31 extracts mapping metadata and determines that the signals it requires are in skip block 1 Blk1. It therefore extracts skip block 1 Blk1, decodes the three single channel elements therein, and provides them to drivers 1, 2a, and 2b, respectively.

[0061] Furthermore, device 1 31 ignores skip block 2, Blk2. Similarly, Device 2 32 extracts mapping metadata and determines that the signals it requires are in skip block 2, Blk2. Device 2 32 therefore skips skip block 1, Blk1 and extracts skip block 2, Blk2. Device 2 32 decodes the channel pair element and provides the left channel output of the CPE to its drivers. Similarly, Device 3 33 extracts mapping metadata and determines that the signals it requires are in skip block 2 Blk2, Device 3 33 therefore skips skip block 1 Blk1 and also extracts skip block 2 Blk2. Device 3 33 decodes the channel pair element and provides the right channel output of the CPE to its drivers.

[0062] In some examples, Device 2 32 may determine that it requires only a subset of the signals from skip block 2 Blk2. In such examples, when possible, Device 2 32 may perform only a subset of the operations required to fully decode the signals in skip block 2 Blk2. Specifically, in the example of Figure 5, Device 2 32 may only perform those processing operations required to extract the left channel of the CPE, thus enabling a reduction of computational complexity. Similarly, Device 3 33 may only perform those processing operations required to extract the right channel of the CPE.

[0063] For example, in the case of Figure 5, there may be times where the CPE is coded using joint-channel coding, and other times where it is coded using independent channel coding. During times when the CPE is coded using independent channel coding, Device 2 32 may extract only the first (e.g., left) channel of the CPE, while Device 3 33 may extract only the second (e.g., right) channel of the CPE.

[0064] In another example, the channels of the CPE may be coded using joint-channel coding, in which case devices 2 32 and 3 33 must extract two intermediate channels of the CPE. However, Device 2 32 may still be able to operate with reduced computational complexity by

only performing those operations required to extract the Left channel from the intermediate channels. Similarly, Device 3 33 may still be able to operate with reduced computational complexity by only performing those operations required to extract the Right channel from the intermediate decoded channels.

[0065] Depending on the specifics of the joint-channel coding, other optimizations may be possible. The identity of each of the different devices/decoders may be defined during a system initialization or set-up phase. Such a set-up is common and generally involves measuring the acoustics of a room, speaker distance to sweet spot, and so on.

DISTRIBUTION OF FLEXIBLE RENDERING (PRE-ENCODE AND POST DECODE) AND THE APPLICATION OF DELAY ETC. POST DECODER

[0066] For a use-case where flexible rendering is applied in a Hub/TV prior to sending the relevant signals to the specific devices/speakers, the rendering may create signals that, from a coding perspective, are more difficult to code, for example when considering joint coding of such signals. One reason is that flexible rendering may apply different delays, equalization, and/or gain adjustments for different devices (e.g., depending on placement of a speaker relative to the other speakers and the listener). It would also be possible to pre-set information for example the gain and delay at an initial set-up and flexibly only render the equalization. Other variants of pre-set information and flexible rendering information may also be possible in other examples. Note that, in this document, the term "gain" should be interpreted to mean any level adjustment (e.g., attenuation, amplification, or pass-through), rather than being restricted to only certain level adjustments (e.g., amplification).

[0067] In the example of Figure 6, the right channel 33 and left channel 32 speakers are given different latencies, reflecting different placements relative to the listener (e.g., since the speakers 32,33 may not be equidistant to the listener, different latencies may be applied to the signals output from the speakers 32,33 so that coherent sounds from different speakers 32,33 arrive at the listener at the same time). The introduction of such differing latencies to coherent signals intended for playback over different speakers 31,32,33 makes joint coding of such signals challenging.

[0068] To address this challenge, aspects of the flexible rendering process may be parameterized and applied in the endpoint device after decoding of the signal.

[0069] In the example of Figure 7, delay and gain values for each device are parameterized and included in the encoded signals sent to the respective speakers 31,32,33. The respective signals may be decoded by respective devices 31,32,33, which then may introduce the parameterized gain and delay values to respective decoded signals.

[0070] In an example, such as that shown in Figure 7, where coded signals for different devices 31,32,33 are sent in separable blocks (e.g., skippable blocks), the parameters (e.g., delay and gain) may also be sent in separable blocks, such that a device 31,32,33 may extract only a subset of the parameters required for that device 31,32,33 and ignore (and skip over) those parameters which are not required for that device 31,32,33. In such case, mapping metadata indicating which parameters are included in which blocks may be provided to each device.

[0071] Note also that, although Figure 7 only indicates delay and gain parameters, other parameters may also be included, such as equalization parameters. The equalization parameters may, for example, comprise a plurality of gains to be applied to different frequency regions, an indication of a predetermined equalization curve to be applied by the playback device 31,32,33, one or more sets of infinite impulse response (IIR) or finite impulse response (FIR) filter coefficients, a set of biquad filter coefficients, parameters specifying characteristics of a parametric equalizer, as well as other parameters for specifying equalization known to those skilled in the art.

[0072] Furthermore, the parameterization of flexible rendering aspects need not be static, and can be dynamic (for example, in case a listener moves during playback back of an audio program). As such, it may be preferable to allow for the parameters to change dynamically. In cases where one or more parameters change during an audio program, the device may interpolate between previous delay and/or gain parameters and updated delay and/or gain parameters to provide a smooth transition. This may be particularly useful in situations where the system dynamically tracks the location of the listener, and correspondingly updates the sweet-spot for dynamic rendering.

[0073] As has also been noted, and will be described subsequently, when flexible rendering is applied, increased level of correlation between channels may occur, and this can be exploited by a more flexible joint coding.

ECHO-REFERENCE CODING AND SIGNALING

[0074] For a use-case as outlined in Figure 8, where there are multiple device/speakers 30 operating together receiving the signal in either a broadcast Mode, illustrated to the left in Figure 8, or in a Unicast-multipoint mode, illustrated to the right of Figure 8, and at the same time there are microphones 40 on the device 30 to enable "listen" capability, the need for echo-management arises.

[0075] When performing echo-management with multiple speakers/devices, it may be beneficial to use echo-references from more than just the local speaker device. As an example, one device may be placed close to another device, and hence the signal from the device in close vicinity will impact echo-management of the device with the active microphone. In a broadcast mode use-case, each device receives the signals for all of the devices. When a device has signals for other devices, it may be beneficial to use those signals as echo-references. To do so, it is necessary to signal to a particular device which of the signals may be used as echo-references for which other devices.

[0076] In one example this may be done by providing metadata that not only maps the channels/signals to be played out (from the whole set) by specific speakers/devices, but also maps the channels/signals to be used as echo-references (from the whole set) for specific speakers/devices. Such metadata or signaling may be dynamic, enabling the indication of the preferred echo-reference to vary over time.

[0077] For a use-case where each device/speaker receives only the specific signals it is to play out, in order to provide appropriate echo-references, it may be necessary to transmit additional signals (e.g., echo-reference signals) to each device/speaker. Again, to do so, it is necessary to provide device-specific signaling so that each device can select the appropriate signal for playout and the appropriate signals for echo-management.

[0078] Since the signals for echo-management are only used by the device for performing echo management, and are not played out by the device for the listener, the echo-reference signals may be coded or represented differently than the signals intended for playback by the device. Specifically, because successful echo-management may be achieved with a signal coded at a lower rate than what would normally be used for playout for a listener, additional compression tools, such as a parametric representation of the signal, which may not be suitable for playout to a listener at all, but that captures the necessary features of the audio signal, may provide good echo-management at significantly reduced transmission cost.

ASSORTED SYNTAX ELEMENTS

[0079] In some examples, blocks are used for optimizing transport of audio. In the described format, each frame may be split up into blocks, as described above in relation to skippable blocks. A block may be identified by the frame number to which it belongs, a block ID which may be used to associate consecutive blocks from different frames with the same ID to a block stream, and a priority for retransmission. One example of the above is a stream with multiple frames N-2, N-1 and N, frame N comprises multiple blocks identified as ID1, ID2, ID3 which are illustrated in figure 13. An example of this stream but in a bitstream format is illustrated in figure 14. In some examples, for audio signals of an immersive audio program, the block ID of a block may indicate which set of signals of the entire immersive audio program is carried by that block.

[0080] A use case for the described format is the reliable transmission of audio over a wireless network such as Wifi at low latency. Wifi, for example, uses a packet-based network protocol. Packet sizes are usually limited. A typical maximum packet size in IP networks is 1500 bytes. The block-based architecture of the stream allows for flexibility when assembling packets for transmission. For instance, packets with smaller frames can be filled up with retransmitted blocks from other frames. Large frames can be split on block boundaries before packetizing them to reduce dependencies between packets on the network protocol layer.

[0081] Figure 9 shows the relationship between frames, blocks, and packets. A frame carries audio data, preferably all audio data, that represents a continuous segment of an audio signal with a start time, an end time, and a duration which is the difference of end time and start time. The continuous segment may comprise a time period according to ISO/IEC 14496-3, subpart 4, section 4.5.2.1.1. Section 4.5.2.1.1 describes the content of a raw_data_block(). A frame may also carry redundant representations of that segment e.g., encoded at a lower data rate. After encoding, that frame can be split into blocks. The blocks can be combined into packets for transmission over a packet-based network. Blocks from different frames can be combined in a single packet, and / or may be sent out of order.

[0082] In an example, blocks are used for addressing individual devices. Packets of data are received by individual devices or related groups of devices. The concept of skippable blocks may be used to address individual devices or related groups of devices. Even if the network operates in a broadcast mode when sending packets to different devices, the processing of audio (e.g., decoding, rendering, etc.) can be reduced to the blocks that are addressed to that device. All other blocks, even if received within the same packet, can simply be skipped over. In some examples, blocks may be brought in the right order, based on their decode or presentation time.

Retransmitted blocks with lower priority may be removed if the same block with higher priority has also been received. The stream of blocks may then be fed to the decoder.

[0083] In some examples, the configuration of the stream and the devices are sent out of band. The codec allows for setting up a connection where the audio streams are transmitted at comparably high rate but with low latency. The configuration of such a connection may stay stable over the duration of such a connection. In that case, instead of making the configuration part of the audio stream, it can be transmitted out of band. For such an out of band transmission, even different networks, or network protocols may be used. For instance, the audio stream may use User Datagram Profile (UDP) for low latency transmission while the configuration may use Transmission Control Protocol (TCP) to ensure a reliable transmission of the configuration.

ENABLING THE CODEC IN THE CONTEXT OF MPEG-4 AUDIO

[0084] One specific application of the present technology is use within MPEG-4 Audio. In MPEG-4 Audio, different AOTs (Audio Object Type) are defined for different codec technologies. For the format described in this document, a new AOT may be defined, which allows for specific signaling and data to that format. Furthermore, the configuration of a decoder in MPEG-4 is done in the DecoderSpecificInfo() payload that in turn carries the AudioSpecificConfig() payload. In the latter, certain general signaling agnostic of the specific format is defined, such as sampling rate and channel configurations, as well as specific info for the specific AOT. For conventional formats, where the whole stream is decoded by a single device, this may make sense. However, in a Broadcast Mode, where a single stream is transmitted to several decoders, with each respective decoder decoding only parts of the stream, up-front channel configuration signaling (as a means to set up the output functionality of the device) may be suboptimal.

[0085] Figure 10 illustrates the conventional MPEG-4 high level structure (in black), with modifications in grey. To support the Broadcast use-case, a codecSepcificConfig() (where "codec" may be a generic placeholder name) is defined, in which signaling is re-defined for specific use-cases, so that it is possible to map specific channel elements to specific devices, as well as to include other relevant static parameters. An MPEG-4 element channelConfiguration having a value of "0" is defined as channel configuration defined in codecSpecificConfig. Hence, this value may be used to enable a revision of the signaling of channel configurations inside the codec specific config.

[0086] Further, in the spirit of MPEG-4, raw payloads are specified for the specific decoder at hand, given that the codecSpecificConfig() is decodable. However, the format ensures that dynamic metadata are part of the raw payloads and ensure that length information is available for all of the raw payloads, so that the decoder easily can skip over elements that are not relevant for the specific device.

[0087] In Figure 11, part of an example raw_data_block as defined in MPEG-4 is given to the right. The raw data block contains channel elements (single channel elements (SCE) or channel pair elements (CPE)) in a given order. However, a decoder wishing to skip over parts of these channel elements which may not be relevant to the output device at hand would, in a conventional MPEG-4 Audio syntax have to parse (and to some extent also decode) all the channel elements to be able to extract the relevant parts. In the new raw_data_block illustrated to the left in Figure 11, the contents would be made up from skippable blocks so that the decoder can skip over the irrelevant parts and decode only the channel elements indicated by the metadata as being relevant for the device at hand. In an example, the skippable blocks comprise the raw_data_block, and related information.

[0088] It will also be possible to use blocks for retransmission. Each frame can be split up into blocks, as described in the skippable blocks section above. A block can be identified by the frame number to which it belongs, a block ID which may be used to associate consecutive blocks from different frames with the same ID to a block stream, and a priority for retransmission, as illustrated in Figures 15 and 16. For instance, a high priority for retransmission (e.g., indicated in Figures 15 and 16 with Priority 0) signals that a block shall be preferred at the receiver over another block with the same block ID and frame counter but lower priority for retransmission (e.g., indicated in Figures 15 and 16 with Priority 1). Note that, as in the example of Figures 15 and 16, priority may decrease as priority index increases (e.g., Priority 1 may be lower priority than Priority 0), while in other examples, priority may increase as priority index increases (e.g., Priority 0 is lower priority than Priority 1). Still other examples will be evident to the skilled person.

[0089] The syntax may support retransmission of audio elements. For retransmission, various quality levels may be supported. The retransmitted blocks may therefore carry a 'priority' flag to indicate which blocks with the same block ID should take priority for the decoder, because blocks received with the same frame counter and block ID are redundant, and hence for the decoder mutually exclusive, as illustrated in Figures 17 and 18.

[0090] The retransmission of blocks may be done at a reduced data rate. Such a reduced data rate may be achieved by reducing the signal to noise ratio of the audio signal, reducing the bandwidth of the audio signal, reducing the channel count of the audio signal (e.g., as described in U.S. Patent 11,289,103, which is incorporated by reference in its entirety), or any combination thereof. In order for the decoder to pick the audio block that provides the best possible quality, the block that provides the highest quality signals may have the highest decoding priority, the block with the second highest quality signals may have the second highest priority, and so on.

[0091] The blocks may also be retransmitted at the same quality level. In such a case, the priority may reflect the latency of the retransmitted block.

TOOLS TO IMPROVE CORE CODING EFFICIENCY

[0092] Further examples could allow for sharing MDCT Quantization Scale Factors over Channel Elements. In some cases, it may be possible to share MDCT scale factors over certain channels to reduce the associated side bit rate. Furthermore, scale factor sharing could be extended to different Channel Elements, for example as for 7.1.4 input. Usage of shared scale factors may be indicated by 2 signaling bits allowing for 3 active configurations. One possible configuration would be to share scale factors in left horizontal channels, left top channels, right horizontal channels, and right top channels. The specific syntax may be aligned to the skippable blocks concept, to ensure that scale factors are only shared within skip blocks.

[0093] It would also be possible to have joint coding of more than two channels. In the example illustrated in Figures 4, 5, 6, and 7 block Blk 1 carries three SCEs, one for each of the drivers in the smart speaker considered in this scenario. In such a situation, in can be beneficial to enable joint coding of not just two channels (as is provided by the CPE, which could be extended to include the stereo prediction, e.g, the MDCT-based complex prediction stereo tool as described in ISO/IEC 23003-3, referred to as MPEG-D USAC), but more than two channels, for example as described in the SAP (Stereo Audio Processing) tool introduced in ETSI TS 103 190.

[0094] Similarly, it is noted that in the case of flexible rendering, there may be a high amount of correlation between signals across devices. For such signals, it may be beneficial to construct skip blocks that cover multiple devices and allow for joint channel coding to be applied over more than two channels using the tools outlined above.

[0095] Another joint channel coding tool which may be used, for certain transform lengths, is channel coupling, in which a composite channel and scale factor information is transmitted for

mid and high frequencies. This may provide bitrate reduction for playback in the good quality range. For example, using a channel coupling tool for frames with a transform length of 256, corresponding to a frame length around 256 samples for low latency coding, may be beneficial.

[0096] This disclosure would also allow for efficient coding of bandlimited signals. In the scenario where separate signals feeding different drivers of a smart speaker are sent, some of these signals can be bandlimited, such as, for example, the case of a three-way driver configuration with woofer, mid-range, and tweeter. Hence, efficient encoding of such bandlimited signals is desirable, which can translate into specifically tuned psychoacoustic models and bit allocation strategies as well as potential modifications of the syntax to handle such scenarios with improved coding efficiency and/or reduced computational complexity (e.g., enabling the use of a bandlimited IMDCT for the woofer feed).

[0097] In some examples, combination of high frequency reconstruction and noise addition in the MDCT domain would be possible. Modern audio codecs are typically designed to support parametric coding techniques, and similarly here one can envision the inclusion of e.g., high frequency reconstruction methods in the MDCT domain to retain the low latency and perform noise addition schemes in the MDCT domain to manage tonal to noise ratio in the reconstructed high band. Such parametric coding techniques may be useful especially for lower operation points, and particularly in scenarios where re-transmission occurs as part of an FEC (Forward Error Correction) scheme, where typically the main signal is transmitted, and then same signal is re-transmitted at a lower bitrate in a delayed fashion.

[0098] These examples would further allow for peak complexity reduction in the encoder. This would be applicable if the buffer model restrictions for a constant bitrate transport channel are not in place. In constant bitrate with buffer mode, it could happen that after quantization and counting, the resulting number of bits to be used for the to be encoded frame is above the limit of allowed bits. At least one new, coarser quantization and bit counting step need to be conducted to fulfill the bit requirement. If this limitation is more relaxed, the encoder could keep the first quantization result and would be done at a slightly higher instantaneous bitrate. Still an encoder behavior similar to that of the constant bitrate with buffer model encoder can be achieved, by using the same bit reservoir control mechanism, together with a virtual buffer fullness that gets updated according to an encoder obeying the buffer requirements. This would save additional quantization and bit counting steps and the resulting audio quality would be the same or better compared to the constant bitrate case with the drawback of a slightly increased overall bitrate.

RETURN CHANNEL CONCEPT

[0099] For play and listen use-cases, smart speakers 30 may have one or more microphones 40 (e.g., a microphone or a microphone array), which are configured to capture a mono or a spatial soundfield (e.g., in mono, stereo, A-format, B-format, or any other isotropic or anisotropic channel format), and there is a need for the codec to be able to do efficient coding of such formats in a low latency manner, so that the same codec is used for broadcast/transmission to the smart speaker device 30 as is used for the return channel, illustrated in for example Figure 12.

[0100] It should also be noted, that while wake word detection is performed on the smart speaker device, the speech recognition typically is done in the cloud, with a suitable segment of recorded speech triggered by the wake word detection.

[0101] In this context it may be interesting to save system complexity overall by creating speech analytics systems on e.g., an intermediate format/representation within the codec. For the simplest use-case, where no human-to-human conversation is ongoing, a suitable representation can be defined specific to the speech recognition task, as it is not to be listened to by another human. This could be band energies, Mel-frequency Cepstral Coefficients (MFCCs) etc., or a low bitrate version of MDCT coded spectra.

[0102] In a use-case where there is human to human conversation ongoing, and there is a wake word and required speech recognition interleaved, one would either want to be able to extract the relevant parts of the stream for that. Such extraction could entail a layered coding structure, of simply additional data sent in parallel to the main speech signal. We also envision a defined transcode from the existing decoded MDCT spectra to the representation relevant, and in doing so simplifying the structure. Essentially a decoder would decode and output the human listenable audio until signaled that it should also decode (in parallel) the speech recognition representation. This representation can be "peeled off" a layered stream, simply an alternative decode of the same complete data, or simply the decode and output of an additional representation in the stream. In this use-case, the signaling is the enabling piece that indicates that the decoder on the receiving side should output a speech recognition relevant representation.

ENUMERATED EXAMPLES

[0103] In the following, seven sets of enumerated example examples (EEE-As, EEE-Bs, EEE-Cs, EEE-Ds, EEE-Es, EEE-Fs, and EEE-Gs), which are not claims, describe aspects of the examples disclosed herein.

[0104] EEE-A1. A method for decoding an audio signal, the method comprising:

receiving a bitstream comprising at least one frame, wherein each frame of the at least one frames comprises a plurality of blocks;

determining, from signaling data, information to identify portions of one or more blocks of the plurality of blocks to be skipped over when decoding, based on device information of an output device; and

decoding the bitstream while skipping over the identified portions of the one or more blocks.

[0105] EEE-A2. The method of EEE-A1, wherein the information to identify portions of the one or more blocks of the plurality of blocks to be skipped over when decoding comprises a matrix that associates each output device of a plurality of output devices to one or more bitstream elements.

[0106] EEE-A3. The method of EEE-A2, wherein the one or more bitstream elements are required for the decoding of the bitstream for the corresponding associated output device.

[0107] EEE-A4. The method of any of EEE-A1 to EEE-A3, wherein the output device may comprise at least one of a wireless device, a mobile device, a tablet, a single-channel speaker, and/or a multi-channel speaker.

[0108] EEE-A5. The method of any of EEE-A1 to EEE-A4, wherein the identified portion comprises at least one block.

[0109] EEE-A6. The method of any of EEE-A1 to EEE-A5, wherein the output device is a first output device, further comprising applying joint coding techniques between one or more signals of the bitstream to a second output device and a third output device.

[0110] EEE-A7. The method of any of EEE-A1 to EEE-A6, wherein an identity of each output device and/or decoder is defined during a system initialization phase.

[0111] EEE-A8. The method of any of EEE-A1 to EEE-A7, wherein the signaling data is determined from metadata of the bitstream.

[0112] EEE-A9. An apparatus configured to perform the method of any one of EEE-A1 to EEE-A8.

[0113] EEE-A10. A non-transitory computer readable storage medium comprising a sequence of instructions which, when executed, cause one or more devices to perform the method of any one of EEE-A1 to EEE-A8.

[0114] EEE-B1. A method for generating an encoded bitstream from an audio program comprising a plurality of audio signals, the method comprising:

receiving, for each of the plurality of audio signals, information indicating a playback device with which the respective audio signal is associated;

receiving, for each playback device, information indicating at least one of a delay, a gain, and an equalization curve associated with the respective playback device;

determining, from the plurality of audio signals, a group of two or more related audio signals;

applying one or more joint-coding tools to the two or more related audio signals of the group to obtain jointly-coded audio signals;

combining the jointly-coded audio signals, an indication of the playback devices with which the jointly-coded audio signals are associated, and indications of the delay and the gain associated with the respective playback devices with which the jointly-coded audio signals are associated, into an independent block of an encoded bitstream.

[0115] EEE-B2. The method of EEE-B1, wherein the delay, gain, and/or equalization curve associated with the respective playback device depend on a location of the respective playback device relative to a location of a listener.

[0116] EEE-B3. The method of EEE-B1 or EEE-B2, wherein the delay, gain, and/or equalization curve associated with the respective playback device depend on a location of the respective playback device relative to locations of other playback devices.

[0117] EEE-B4. The method of any one of EEE-B1 to EEE-B3, wherein the delay, gain, and/or equalization curve are dynamically variable.

[0118] EEE-B5. The method of EEE-B4, wherein the delay, gain, and/or equalization curve are adjusted in response to a change in the location of the listener.

[0119] EEE-B6. The method of EEE-B4 or EEE-B5, wherein the delay, gain, and/or equalization curve are adjusted in response to a change in the location of the playback device.

[0120] EEE-B7. The method of any one of EEE-B4 to EEE-B6, wherein the delay, gain, and/or equalization curve are adjusted in response to a change in the location of one or more of the other playback devices.

[0121] EEE-B8. The method of any one of EEE-B1 to EEE-B7, further comprising determining, from the plurality of audio signals, an audio signal which is not part of the group of two or more related audio signals.

[0122] EEE-B9. The method of EEE-B8, further comprising, for the audio signal which is not part of the group of two or more related audio signals, applying the delay, gain, and/or equalization curve associated with the playback device with which the audio signal is associated.

[0123] EEE-B10. The method of EEE-B9, further comprising independently coding the audio signal which is not part of the group of two or more related audio signals, and combining the independently coded audio signal and an indication of the playback device with which the independently coded audio signal is associated into a separate independently decodable subset of the encoded bitstream.

[0124] EEE-B11. The method of EEE-B8, further comprising independently coding the audio signal which is not part of the group of two or more related audio signals, and combining the independently coded audio signal, an indication of the playback device with which the independently coded audio signal is associated, and an indication of the delay, gain, and/or equalization curve associated with the playback device with which the independently coded audio signal is associated into a separate independently decodable subset of the encoded bitstream.

[0125] EEE-B12. A method for decoding one or more audio signals associated with a playback device from a frame of an encoded bitstream, wherein the frame comprises one or more independent blocks of encoded data, the method comprising:

identifying, from the encoded bitstream, an independent block of encoded data corresponding to the one or more audio signals associated with the playback device;

extracting, from the encoded bitstream, the identified independent block of encoded data;

determining that the extracted independent block of encoded data includes two or more jointly-coded audio signals;

applying one or more joint-decoding tools to the two or more jointly-coded audio signals to obtain the one or more audio signals associated with the playback device;

determining, from the extracted independent block of encoded data, at least one of a delay, a gain, and an equalization curve associated with the playback device;

applying the delay, gain, and/or equalization curve associated with the playback device to the one or more audio signals associated with the playback device.

[0126] EEE-B13. The method of EEE-B12, wherein the determined delay, gain, and/or equalization curve associated with the playback device depend on a location of the playback device relative to a location of a listener.

[0127] EEE-B14. The method of EEE-B12 or EEE-B13, wherein the determined delay, gain, and/or equalization curve associated with the playback device depend on a location of the playback device relative to other playback devices.

[0128] EEE-B15. The method of any one of EEE-B12 to EEE-B14, wherein the determined delay, gain, and/or equalization curve of the playback device are dynamically variable.

[0129] EEE-B16. The method of EEE-B15, wherein, when the determined delay, gain, and/or equalization curve associated with the playback device differ from a previously determined delay, gain, and/or equalization curve associated with the playback device, the method further comprises interpolating between the previously determined delay, gain, and/or equalization curve associated with the playback device and the determined delay, gain, and/or equalization curve associated with the playback device.

[0130] EEE-B17. The method of EEE-B16, wherein the determined delay, gain, and/or equalization curve differs from the previously determined delay, gain, and/or equalization curve due to a change in the location of a listener.

[0131] EEE-B18. The method of EEE-B16 or EEE-B17, wherein the determined delay, gain, and/or equalization curve differ from the previously determined delay, gain, and/or equalization curve due to a change in the location of the playback device.

[0132] EEE-B19. The method of any one of EEE-B16 to EEE-B18, wherein the determined delay, gain, and/or equalization curve differ from the previously determined delay, gain, or equalization curve due to a change in the location of one or more of the other playback devices.

[0133] EEE-B20. The method of any one of EEE-B12 to EEE-B19, wherein the frame of the encoded bitstream comprises two or more independent blocks of encoded data, the method further comprising:

determining that one or more of the independent blocks contain audio signals not associated with the playback device; and

ignoring the one or more independent blocks that contain audio signals not associated with the playback device.

[0134] EEE-B21. The method of any one of EEE-B12 to EEE-B20, wherein applying one or more joint-decoding tools comprises identifying a subset of the jointly-coded audio signals that are associated with the playback device, and reconstructing only that subset of the jointly-coded audio signals to obtain the one or more audio signals associated with the playback device.

[0135] EEE-B22. The method of any one of EEE-B12 to EEE-B20, wherein applying one or more joint-decoding tools comprises reconstructing each of the jointly-coded audio signals, identifying a subset of the reconstructed jointly-coded audio signals associated with the playback device, and obtaining the one or more audio signals associated with the playback device from the subset of the reconstructed jointly-coded audio signals associated with the playback device.

[0136] EEE-B23. An apparatus configured to perform the method of any one of EEE-B1 to EEE-B22.

[0137] EEE-B24. A non-transitory computer readable storage medium comprising a sequence of instructions which, when executed, cause one or more devices to perform the method of any one of EEE-B1 to EEE-B22.

[0138] EEE-C1. A method for generating a frame of an encoded bitstream of an audio program comprising a plurality of audio signals, wherein the frame comprises two or more independent blocks of encoded data, the method comprising:

receiving, for one or more of the plurality of audio signals, information indicating a playback device with which the one or more audio signals are associated;

receiving, for the indicated playback device, information indicating one or more additional associated playback devices;

receiving one or more audio signals associated with the indicated one or more additional associated playback devices;

encoding the one or more audio signals associated with the playback device;

encoding the one or more audio signals associated with the indicated one or more additional associated playback devices;

combining the one or more encoded audio signals associated with the playback device and signaling information indicating the one or more additional associated playback devices into a first independent block;

combining the one or more encoded audio signals associated with the one or more additional associated playback devices into one or more additional independent blocks; and

combining the first independent block and the one or more additional independent blocks into the frame of the encoded bitstream.

[0139] EEE-C2. The method of EEE-C1, wherein the plurality of audio signals comprises one or more groups of audio signals not associated with the playback device or the one or more additional associated playback devices, further comprising:

encoding each of the one or more groups of audio signals not associated with the playback device or the one or more additional associated playback devices into a respective independent block; and

combining the respective independent block for each of the one or more groups into the frame of the encoded bitstream.

[0140] EEE-C3. The method of EEE-C1 or EEE-C2, wherein the one or more audio signals associated with the indicated one or more additional associated playback devices are specifically intended for use as echo-references for performing echo-management for the playback device.

[0141] EEE-C4. The method of EEE-C3, wherein the one or more audio signals intended for use as echo-references are transmitted using less data than the one or more audio signals associated with the playback device.

[0142] EEE-C5. The method of EEE-C3 or EEE-C4, wherein the one or more audio signals intended for use as echo-references encoded using parametric coding tools.

[0143] EEE-C6. The method of EEE-C1 or EEE-C2, wherein the one or more audio signals associated with the one or more other playback devices are suitable for playback from the one or more other playback devices.

[0144] EEE-C7. A method for decoding one or more audio signals associated with a playback device from a frame of an encoded bitstream, wherein the frame comprises two or more independent blocks of encoded data, wherein the playback device comprises one or more microphones, the method comprising:

identifying, from the encoded bitstream, an independent block of encoded data corresponding to the one or more audio signals associated with the playback device;

extracting, from the encoded bitstream, the identified independent block of encoded data;

extracting from the identified independent block of encoded data the one or more audio signals associated with the playback device;

identifying, from the encoded bitstream, one or more other independent blocks of encoded data corresponding to one or more audio signals associated with one or more other playback devices;

extracting from the one or more other independent blocks of encoded data the one or more audio signals associated with the one or more other playback devices;

capturing one or more audio signals using the one or more microphones of the playback device; and

using the one or more extracted audio signals associated with the one or more other playback devices as echo-references for performing echo-management for the playback device in response to the one or more captured audio signals.

[0145] EEE-C8. The method of EEE-C7, the method further comprising:

determining that the encoded bitstream comprises one or more additional independent blocks of encoded data; and

ignoring the one or more additional independent blocks of encoded data.

[0146] EEE-C9. The method of EEE-C8, wherein ignoring the one or more additional independent blocks of encoded data comprises skipping over the one or more additional independent blocks of encoded data without extracting the additional one or more independent blocks of encoded data.

[0147] EEE-C10. The method of any one of EEE-C7 to EEE-C9, wherein the one or more audio signals associated with the one or more other playback devices are specifically intended for use as echo-references for performing echo-management for the playback device.

[0148] EEE-C11. The method of EEE-C10, wherein the one or more audio signals specifically intended for use as echo-references are transmitted using less data than the one or more audio signals associated with the playback device.

[0149] EEE-C12. The method of EEE-C10 or EEE-C11, wherein the one or more audio signals specifically intended for use as echo-references are reconstructed from parametric representations of the one or more audio signals.

[0150] EEE-C13. The method of any one of EEE-C7 to EEE-C9, wherein the one or more audio signals associated with the one or more other playback devices are suitable for playback from the one or more other playback devices.

[0151] EEE-C14. The method of EEE-C7, wherein the encoded signal includes signaling information indicating the one or more other playback devices to use as echo-references for the playback device.

[0152] EEE-C15. The method of EEE-C14, wherein the one or more other playback devices indicated by the signaling information for a current frame differ from the one or more other playback devices used as echo-references for a previous frame.

[0153] EEE-C16. An apparatus configured to perform the method of any one of EEE-C1 to EEE-C15.

[0154] EEE-C17. A non-transitory computer readable storage medium comprising a sequence of instructions which, when executed, cause one or more devices to perform the method of any one of EEE-C1 to EEE-C15.

[0155] EEE-D1. A method for transmitting an audio signal, the method comprising:

generating packets of data comprising portions of a bitstream, wherein the bitstream comprises a plurality of frames, wherein each frame of the plurality of frames comprises a plurality of blocks, wherein the generating comprises:

assembling a packet of data with one or more blocks of the plurality of blocks, wherein blocks from different frames are combined into a single packet and/or are transmitted out of order; and

transmitting the packets of data via a packet-based network.

[0156] EEE-D2. The method of EEE-D1, wherein each block of the plurality of blocks comprises identifying information.

[0157] EEE-D3. The method of EEE-D2, wherein the identify information comprises at least one of a block ID, a corresponding frame number associated with the block, and/or a priority for retransmission.

[0158] EEE-D4. The method of any of EEE-D1 to EEE-D3, wherein each frame of the plurality of frames carries all audio data that represents a continuous segment of an audio signal with a start time, an end time, and a duration.

[0159] EEE-D5. A method for decoding an audio signal, the method comprising:

receiving packets of data comprising portions of a bitstream, wherein the bitstream comprises a plurality of frames, wherein each frame of the plurality of frames comprises a plurality of blocks;

determining a set of blocks of the plurality of blocks addressed to a device; and

decoding the set of blocks addressed to the device and skipping decoding the blocks of the plurality of blocks not addressed to the device.

[0160] EEE-D6. A method for transmitting an audio stream, the method comprising:

transmitting the audio stream, wherein the audio stream comprises a plurality of frames, wherein each frame of the plurality of frames comprises a plurality of blocks, wherein the transmitting comprises transmitting configuration information for the audio stream out of band.

[0161] EEE-D7. The method of EEE-D6, wherein transmitting configuration information for the audio stream out of band comprises:

transmitting the audio stream via a first network and/or a first network protocol; and

transmitting the configuration information via a second network and/or a second network protocol.

[0162] EEE-D8. The method of EEE-D7, wherein the first network protocol is a User Datagram Protocol (UDP) and the second network protocol is a Transmission Control Protocol (TCP).

[0163] EEE-D9. A method for decoding an audio signal, the method comprising:

receiving a bitstream that comprises:

information corresponding to a signaling of static configuration aspects;

static metadata; and

mapping one or more channel elements to one or more devices based on the information and/or static metadata.

[0164] EEE-D10. The method of EEE-D9, wherein the bitstream is received by a plurality of decoders configured to decode the bitstream, wherein each decoder of the plurality of decoders is configured to decode a portion of the bitstream.

[0165] EEE-D11. The method of EEE-D9 or EEE-D10, wherein the bitstream further comprises dynamic metadata.

[0166] EEE-D12. The method of any of EEE-D9 to EEE-D11, wherein the bitstream comprises a plurality of blocks, wherein each block of the plurality of blocks comprises:

information that enables for a portion of the block to be skipped during decoding, wherein the portion is not needed for a device; and

dynamic metadata.

[0167] EEE-D13. A method for re-transmitting blocks of an audio signal, the method comprising:

transmitting one or more blocks of a bitstream, wherein the bitstream comprises a plurality of blocks, wherein each of the one or more blocks of the bitstream has been previously transmitted; and

wherein each of the one or more blocks comprises a decoding priority indicator.

[0168] EEE-D14. The method of EEE-D13, wherein the decoding priority indicator indicates to a decoder an order of priority for decoding the one or more blocks of the bitstream.

[0169] EEE-D15. The method of EEE-D13 or EEE-D14, wherein each block of the one or more blocks comprises a same block ID.

[0170] EEE-D16. The method of any of EEE-D13 to EEE-D15, wherein the transmission of the one or more blocks of the bitstream is transmitted by reducing a data rate in comparison to the previous transmission.

[0171] EEE-D17. The method of EEE-D16, wherein reducing the data rate comprises at least one of reducing a signal to noise ratio of the audio signal, reducing a bandwidth of the audio signal, and/or reducing a channel count of the audio signal.

[0172] EEE-E1. A method for generating a frame of an encoded bitstream of an audio program comprising a plurality of audio signals, wherein the frame comprises one or more independent blocks of encoded data, the method comprising:

receiving, for each of the plurality of audio signals, information indicating a playback device with which the respective audio signal is associated;

encoding one or more audio signals associated with a respective playback device to obtain one or more encoded audio signals;

combining the one or more encoded audio signals associated with the respective playback device into a first independent block of the frame;

encoding one or more other audio signals of the plurality of audio signals into one or more additional independent blocks; and

combining the first independent block and the one or more additional independent blocks into the frame of the encoded bitstream.

[0173] EEE-E2. The method of EEE-E1, wherein two or more audio signals are associated with the playback device, and each of the two or more audio signals is a bandlimited signal intended for playback by a respective driver of the playback device, and wherein different encoding techniques are used for each of the bandlimited signals.

[0174] EEE-E3. The method of EEE-E2, wherein a different psychoacoustic model and/or a different bit allocation technique is used for each of the bandlimited signals.

[0175] EEE-E4. The method of any one of EEE-E1 to EEE-E3, wherein an instantaneous frame rate of the encoded signal is variable, and is constrained by a buffer fullness model.

[0176] EEE-E5. The method of EEE-E1, wherein encoding one or more audio signals associated with the respective playback device comprises jointly-encoding the one or more audio signals associated with the respective playback device and one or more additional audio signals associated with one or more additional playback devices into the first independent block of the frame.

[0177] EEE-E6. The method of EEE-E5, wherein jointly-encoding the one or more audio signals and one or more additional audio signals comprises sharing one or more scale factors across two or more audio signals.

[0178] EEE-E7. The method of EEE-E6, wherein the two or more audio signals are spatially related.

[0179] EEE-E8. The method of EEE-E7, wherein the two or more spatially related audio signals comprise left horizontal channels, left top channels, right horizontal channels, or right top channels.

[0180] EEE-E9. The method of EEE-E5, wherein jointly-encoding the one or more audio signals and one or more additional audio signals comprises applying a coupling tool comprising:

combining two or more audio signals into a composite signal above a specified frequency; and

determining, for each of the two or more audio signals, scale factors relating an energy of the composite signal and an energy of each respective signal.

[0181] EEE-E10. The method of EEE-E5, wherein jointly-encoding the one or more audio signals and one or more additional audio signals comprises applying a joint-coding tool to more than two signals.

[0182] EEE-E11. A method for decoding one or more audio signals associated with a playback device from a frame of an encoded bitstream, wherein the frame comprises one or more independent blocks of encoded data, the method comprising:

identifying, from the encoded bitstream an independent block of encoded data corresponding to the one or more audio signals associated with the playback device;

extracting, from the encoded bitstream, the identified independent block of encoded data;

decoding the one or more audio signals associated with the playback device from the independent block of encoded data to obtain one or more decoded audio signals;

identifying, from the encoded bitstream, one or more additional independent blocks of encoded data corresponding to one or more additional audio signals; and

decoding or skipping the one or more additional independent blocks of encoded data.

[0183] EEE-E12. The method of EEE-E11, wherein two or more audio signals are associated with the playback device, and each of the two or more audio signals is a bandlimited signal intended for playback by a respective driver of the playback device, and wherein different decoding techniques are used to decode the two or more audio signals.

[0184] EEE-E13. The method of EEE-E12, wherein a different psychoacoustic model and/or a different bit allocation technique was used to encode each of the bandlimited signals.

[0185] EEE-E14. The method of EEE-E11 or EEE-E13, wherein an instantaneous frame rate of the encoded bitstream is variable, and is constrained by a buffer fullness model.

[0186] EEE-E15. The method of EEE-E11, wherein decoding the one or more audio signals associated with the playback device comprises jointly-decoding the one or more audio signals associated with the respective playback device and one or more additional audio signals associated with one or more additional playback devices from the independent block of encoded data.

[0187] EEE-E16. The method of EEE-E15, wherein jointly-decoding the one or more audio signals and one or more additional audio signals comprises extracting scale factors shared across two or more audio signals.

[0188] EEE-E17. The method of EEE-E16 where the two or more audio signals are spatially related.

[0189] EEE-E18. The method of EEE-E17, wherein the two or more spatially related audio signals comprise left horizontal channels, left top channels, right horizontal channels, or right top channels.

[0190] EEE-E19. The method of EEE-E15, wherein jointly-decoding the one or more audio signals and one or more additional audio signals comprises applying a decoupling tool.

[0191] EEE-E20. The method of EEE-E19, wherein the decoupling tool comprises:

extracting independently decoded signals below a specified frequency;

extracting a composite signal above the specified frequency;

determining respective decoupled signals above the specified frequency from the composite signal and scale factors relating an energy of the composite signal and energies of respective signals; and

combining each independently decoded signal with a respective decoupled signal to obtain the jointly-decoded signals.

[0192] EEE-E21. The method of EEE-E15, wherein jointly-decoding the one or more audio signals and one or more additional audio signals comprises applying a joint-decoding tool to extract more than two audio signals.

[0193] EEE-E22. The method of EEE-E11, wherein decoding the one or more audio signals associated with the playback device comprises applying bandwidth extension to the audio signals in the same domain as the audio signals were coded.

[0194] EEE-E23. The method of EEE-E22, wherein the domain is a modified discrete cosine transform (MDCT) domain.

[0195] EEE-E24. The method of EEE-E22 or EEE-E23, wherein the bandwidth extension comprises adaptive noise addition.

[0196] EEE-E25. An apparatus configured to perform the method of any one of EEE-E1 to EEE-E24.

[0197] EEE-E26. A non-transitory computer readable storage medium comprising a sequence of instructions which, when executed, cause one or more devices to perform the method of any one of EEE-E1 to EEE-E24.

[0198] EEE-F1. A method, performed by a device with one or more microphones, for generating an encoded bitstream, the method comprising:

capturing, by the one or more microphones, one or more audio signals;

analyzing the captured audio signals to determine presence of a wake word;

upon detecting presence of a wake word:

setting a flag to indicate a speech recognition task is to be performed on the captured audio signals:

encoding the captured audio signals;

assembling the encoded audio signals and the flag into the encoded bitstream.

[0199] EEE-F2. The method of EEE-F1, wherein the one or more microphones are configured to capture a mono or a spatial soundfield.

[0200] EEE-F3. The method of EEE-F2, wherein the spatial soundfield is in an A-Format or a B-format.

[0201] EEE-F4. The method of any one of EEE-F1 to EEE-F3, wherein the captured audio signals are intended for use only in performing the speech recognition task.

[0202] EEE-F5. The method of EEE-F4, wherein the captured audio signals are encoded such that when the captured audio signals are decoded, the quality of the decoded audio signals is sufficient for performing the speech recognition task but is not sufficient for human listening.

[0203] EEE-F6. The method of EEE-F4 or EEE-F5, wherein the captured audio signals are converted to a representation comprising one or more of band energies, Mel-frequency Cepstral Coefficients, or Modified Discrete Cosine Transform (MDCT) spectral coefficients prior to encoding the captured audio signals.

[0204] EEE-F7. The method of any one of EEE-F1 to EEE-F3, wherein the captured audio signals are intended for both human listening and for use in performing the speech recognition task.

[0205] EEE-F8. The method of EEE-F7, wherein the captured audio signals are encoded such that when the captured audio signals are decoded, the quality of the decoded audio signals is sufficient for human listening.

[0206] EEE-F9. The method of EEE-F7, wherein encoding the captured audio signals comprises generating a first encoded representation of the captured audio signals and a second encoded representation of the captured audio signals, wherein the first encoded representation is generated such that when the captured audio signals are decoded from the first encoded representation, the quality of the decoded audio signals is sufficient for human listening, and wherein the second encoded representation is generated such that when the captured audio signal signals are decoded from the second encoded representation, the quality of the decoded audio signals is sufficient for performing the speech recognition task, but is not sufficient for human listening.

[0207] EEE-F10. The method of EEE-F9, wherein generating the second encoded representation of the captured audio signals comprises converting the captured audio signals to one or more of a parametric representation, a coarse waveform representation, or a representation comprising one or more of band energies, Mel-frequency Cepstral Coefficients, or Modified Discrete Cosine Transform (MDCT) spectral coefficients, prior to encoding the captured audio signals.

[0208] EEE-F11. The method of EEE-F9 or EEE-F10, wherein assembling the encoded audio signals into a bitstream comprises inserting the first encoded representation into a first independent block of the encoded bitstream, and inserting the second encoded representation into a second independent block of the encoded bitstream.

[0209] EEE-F12. The method of EEE-F9 or EEE-F10, wherein the first encoded representation is included in a first layer of the encoded bitstream, the second encoded representation is included in a second layer of the encoded bitstream, and the first and second layers are included in a single block of the encoded bitstream.

[0210] EEE-F13. The method of any one of EEE-F1 to EEE-F12, further comprising, when presence of the wake word is not detected:

setting the flag to indicate a speech recognition task is not to be performed on the captured audio signals;

encoding the captured audio signals;

assembling the encoded audio signals and the flag into the encoded bitstream.

[0211] EEE-F14. A method for decoding audio signal, comprising:

receiving an encoded bitstream comprising encoded audio signals and a flag indicating whether a speech recognition task is to be performed;

decoding the encoded audio signals to obtain decoded audio signals; and

when the flag indicates that the speech recognition task is to be performed, performing the speech recognition task on the decoded audio signals.

[0212] EEE-F15. The method of EEE-F14, wherein the decoded audio signals are intended for use only in performing the speech recognition task.

[0213] EEE-F16. The method of EEE-F15, wherein the quality of the decoded audio signals is sufficient for performing the speech recognition task but is not sufficient for human listening.

[0214] EEE-F17. The method of claim EEE-F15 or EEE-F16, wherein the decoded audio signals are in a representation comprising one or more of band energies, Mel-frequency Cepstral Coefficients, or Modified Discrete Cosine Transform (MDCT) spectral coefficients prior to encoding the captured audio signals.

[0215] EEE-F18. The method of EEE-F14, wherein the captured audio signals are encoded such that when the captured audio signals are decoded, the quality of the decoded audio signals is sufficient for human listening.

[0216] EEE-F19. The method of EEE-F18, wherein the encoded audio signals comprise a first encoded representation of one or more audio signals and a second encoded representation of the one or more audio signals.

[0217] EEE-F20. The method of EEE-F18, wherein the quality of audio signals decoded from the first representation is sufficient for human listening, and wherein the quality of audio signals decoded from the second representation is sufficient for performing the speech recognition task, but is not sufficient for human listening.

[0218] EEE-F21. The method of EEE-F19 or EEE-F20, wherein the first representation is in a first independent block of the encoded bitstream and the second representation is in a second independent block of the encoded bitstream.

[0219] EEE-F22. The method of EEE-F19 or EEE-F20, wherein the first representation is in a first layer of the encoded bitstream, the second encoded representation is included in a second

layer of the encoded bitstream, and the first and second layers are included in a single block of the encoded bitstream.

[0220] EEE-F23. The method of any one of EEE-F18 to EEE-F22, wherein decoding the encoded audio signals comprises decoding only the second representation, and ignoring the first representation.

[0221] EEE-F24. The method of any one of EEE-F18 to EEE-F23, wherein audio signals decoded from the second encoded representation are in a parametric representation, a waveform representation, or a representation comprising one or more of band energies, Mel-frequency Cepstral Coefficients, or Modified Discrete Cosine Transform (MDCT) spectral coefficients.

[0222] EEE-F25. An apparatus configured to perform the method of any one of EEE-F1 to EEE-F24.

[0223] EEE-F26. A non-transitory computer readable storage medium comprising a sequence of instructions which, when executed, cause one or more devices to perform the method of any one of EEE-F1 to EEE-F24.

[0224] EEE-G1. A method for encoding audio signals of an immersive audio program for low latency transmission to one or more playback devices, the method comprising:

receiving a plurality of time-domain audio signals of the immersive audio program;

selecting a frame size;

extracting a frame of the time-domain audio signals in response to the frame size, wherein the frame of the time-domain audio signals overlaps with a previous frame of time-domain audio signals;

segmenting the audio signals into overlapping frames;

transforming the frame of time-domain audio signals to frequency-domain signals;

coding the frequency-domain signals;

quantizing the coded frequency-domain signals using a perceptually motivated quantization tool;

assembling the quantized and coded frequency-domain signals into one or more independent blocks within the frame; and

assembling the one or more independent blocks into an encoded frame.

[0225] EEE-G2. The method of EEE-G1, wherein the plurality of audio signals comprise channel-based signals having a defined channel configuration.

[0226] EEE-G3. The method of EEE-G2, wherein the channel configuration is one of mono, stereo, 5.1, 5.1.2, 5.1.4, 7.1.2, 7.1.4, 9.1.6, or 22.2.

[0227] EEE-G4. The method of any one of EEE-G1 to EEE-G3, wherein the plurality of audio signals comprise one or more object-based signals.

[0228] EEE-G5. The method of any one of EEE-G1 to EEE-G4, wherein the plurality of audio signals comprise a scene-based representation of the immersive audio program.

[0229] EEE-G6. The method of any one of EEE-G1 to EEE-G5, wherein the selected frame size is one of 128, 256, 512, 1024, 120, 240, 480, or 960 samples.

[0230] EEE-G7. The method of any one of EEE-G1 to EEE-G6, wherein the overlap between the frame of time-domain audio signals and the previous frame of time-domain audios signal is 50% or lower.

[0231] EEE-G8. The method of any one of EEE-G1 to EEE-G7, wherein the transform is a modified discrete cosine transform (MDCT).

[0232] EEE-G9. The method of any one of EEE-G1 to EEE-G8, wherein two or more of the plurality of audio signals are jointly-coded.

[0233] EEE-G10. The method of any one of EEE-G1 to EEE-G9, wherein each independent block contains encoded signals for one or more playback devices.

[0234] EEE-G11. The method of any one of EEE-G1 to EEE-G10, wherein at least one independent block contains encoded signals for two or more playback devices, and the encoded signals comprise jointly-coded audio signals.

[0235] EEE-G12. The method of any one of EEE-G1 to EEE-G11, wherein at least one independent block contains a plurality of encoded signals covering different bandwidths intended for playback from different drivers of a playback device.

[0236] EEE-G13. The method of any one of EEE-G1 to EEE-G12, wherein at least one independent block contains an encoded echo-reference signal for use in echo-management performed by a playback device.

[0237] EEE-G14. The method of any one of EEE-G1 to EEE-G13, wherein coding the quantized frequency-domain signals comprises applying one or more of the following tools: temporal noise shaping (TNS), joint-channel coding, sharing of scale factors across signals, determining control parameters for high frequency reconstruction, and determining control parameters for noise substitution.

[0238] EEE-G15. The method of any one of EEE-G1 to EEE-G14, wherein one or more independent blocks include parameters for controlling one or more of delay, gain, and equalization of a playback device.

[0239] EEE-G16. A low latency method for decoding audio signals of an immersive audio program from an encoded signal, the method comprising:

receiving an encoded frame comprising one or more independent blocks;

extracting, from one or more independent blocks, quantized and coded frequency-domain signals;

dequantizing the quantized and coded frequency-domain signals;

decoding the dequantized frequency-domain signals;

inverse transforming the decoded frequency-domain signals to obtain time-domain signals; and

overlapping and adding the time-domain signals with time-domain signals from a previous frame to provide a plurality of audio signals of the immersive audio program.

[0240] EEE-G17. The method of EEE-G16, wherein the plurality of audio signals comprise channel-based signals having a defined channel configuration.

[0241] EEE-G18. The method of EEE-G17, wherein the channel configuration is one of mono, stereo, 5.1, 5.1.2, 5.1.4, 7.1.2, 7.1.4, 9.1.6, or 22.2.

[0242] EEE-G19. The method of any one of EEE-G16 to EEE-G18, wherein the plurality of audio signals comprise one or more object-based signals.

[0243] EEE-G20. The method of any one of EEE-G16 to EEE-G19, wherein the plurality of audio signals comprise a scene-based representation of the immersive audio program.

[0244] EEE-G21. The method of any one of EEE-G16 to EEE-G20, wherein a frame of time-domain samples comprises one of 128, 256, 512, 1024, 120, 240, 480, or 960 samples.

[0245] EEE-G22. The method of any one of EEE-G16 to EEE-G21, wherein the overlap with the previous frame is 50% or lower.

[0246] EEE-G23. The method of any one of EEE-G16 to EEE-G22, wherein the inverse transform is an inverse modified discrete cosine transform (IMDCT).

[0247] EEE-G24. The method of any one of EEE-G16 to EEE-G23, wherein each independent block contains quantized and coded frequency-domain signals for one or more playback devices.

[0248] EEE-G25. The method of any one of EEE-G16 to EEE-G24, wherein at least one independent block contains quantized and coded frequency-domain signals for two or more playback devices, and the quantized and coded frequency-domain signals are jointly-coded audio signals.

[0249] EEE-G26. The method of any one of EEE-G16 to EEE-G25, wherein at least one independent block contains a plurality of quantized and coded frequency-domain signals covering different bandwidths intended for playback from different drivers of a playback device.

[0250] EEE-G27. The method of any one of EEE-G16 to EEE-G26, wherein at least one independent block contains an encoded echo-reference signal for use in echo-management performed by a playback device.

[0251] EEE-G28. The method of any one of EEE-G16 to EEE-G27, wherein decoding the dequantized frequency-domain signals comprises applying one or more of the following decoding tools: temporal noise shaping (TNS), joint-channel decoding, sharing of scale factors across signals, high frequency reconstruction, and noise substitution.

[0252] EEE-G29. The method of any one of EEE-G16 to EEE-G28, wherein one or more independent blocks include parameters for controlling one or more of delay, gain, and equalization of a playback device.

[0253] EEE-G30. The method of any one of EEE-G16 to EEE-G29, wherein the method is performed by a playback device, and wherein extracting quantized and coded signals from one or more independent blocks comprises selecting only those blocks which contain quantized and coded frequency-domain signals for playback by the playback device, and ignoring independent blocks which contain quantized and coded frequency domain signals for playback by other playback devices.

[0254] EEE-G31. An apparatus configured to perform the method of any one of EEE-G1 – EEE-G30.

[0255] EEE-G32. A non-transitory computer readable storage medium comprising a sequence of instructions which, when executed, cause one or more devices to perform the method of any one of EEE-G1 to EEE-G30.

[0256] In the claims below and the description herein, any one of the terms comprising, comprised of, or which comprises is an open term that means including at least the elements/features that follow, but not excluding others. Thus, the term comprising, when used in the claims, should not be interpreted as being limitative to the means or elements, or steps listed thereafter. For example, the scope of the expression of a device comprising A and B should not be limited to devices consisting only of elements A and B. Any one of the terms including or which includes or that includes as used herein is also an open term that also means including at least the elements/features that follow the term, but not excluding others. Thus, including is synonymous with and means comprising.

[0257] It should be appreciated that in the above description of examples of the invention, various features are sometimes grouped together in a single example, figure, or description thereof for the purpose of streamlining the disclosure and aiding in the understanding of one or more of the various inventive aspects. This method of disclosure, however, is not to be interpreted as reflecting an intention that more features are required than are expressly recited in each claim. Rather, as the following claims reflect, inventive aspects lie in less than all features of a single foregoing disclosed example. Thus, the claims following the Detailed Description are hereby expressly incorporated into this Detailed Description, with each claim standing on its own as a separate example of this invention.

[0258] Furthermore, while some examples described herein include some, but not other features included in other examples, combinations of features of different examples are meant to be encompassed, and form different examples, as would be understood by those skilled in the art.

For example, in the following claims, any of the claimed examples can be used in any combination.

[0259] Furthermore, some of the examples are described herein as a method or combination of elements of a method that can be implemented by a processor of a computer system or by other means of carrying out the function. Thus, a processor with the necessary instructions for carrying out such a method or element of a method forms a means for carrying out the method or element of a method. Furthermore, an element described herein of an apparatus is an example of a means for carrying out the function performed by the element.

[0260] Furthermore, some of the examples described herein disclose connected solutions which should be interpreted as being possibly implemented in a distribution and/or transmission system such as a wired and/or wireless system. For example by use of any electrical, optical, and/or mobile system, such as 3G, 4G as well as 5G.

[0261] Thus, while there have been described specific examples of the invention, those skilled in the art will recognize that other and further modifications may be made, and it is intended to claim all such changes and modifications. For example, any formulas given above are merely representative of procedures that may be used. Functionality may be added or deleted from the block diagrams and operations may be interchanged among functional blocks. Steps may be added or deleted to the methods described.

[0262] Systems, devices, and methods disclosed hereinabove may be implemented as software, firmware, hardware, or a combination thereof. For example, aspects of the present application may be embodied, at least in part, in a device, a system that includes more than one device, a method, a computer program product, etc.

[0263] In a hardware implementation, the division of tasks between functional units referred to in the above description does not necessarily correspond to the division into physical units; to the contrary, one physical component may have multiple functionalities and one task may be carried out by several physical components in cooperation.

[0264] Certain components or all components may be implemented as software executed by a digital signal processor or microprocessor or be implemented as hardware or as an application-specific integrated circuit. Such software may be distributed on computer-readable media, which may comprise computer storage media (or non-transitory media) and communication media (or transitory media).

[0265] As is well known to a person skilled in the art, the term computer storage media includes both volatile and non-volatile, removable and non-removable media implemented in any method or technology for storage of information such as computer-readable instructions, data structures, program modules, or other data. Computer storage media includes but is not limited to, RAM, ROM, EEPROM, flash memory or other memory technology, CD-ROM, digital versatile disks (DVD) or other optical disk storage, magnetic cassettes, magnetic tape, magnetic disk storage or other magnetic storage devices, or any other medium which can be used to store the desired information, and which can be accessed by a computer.

[0266] Further, it is well known to the skilled person that communication media typically embodies computer-readable instructions, data structures, program modules, or other data in a modulated data signal such as a carrier wave or other transport mechanism and includes any information delivery media.

CLAIMS

1. A method for generating an encoded bitstream from an audio program comprising a plurality of audio signals, the method comprising:

receiving, for each of the plurality of audio signals, information indicating a playback device with which the respective audio signal is associated;

receiving, for each playback device, information indicating at least one of a delay, a gain, and an equalization curve associated with the respective playback device;

determining, from the plurality of audio signals, a group of two or more related audio signals;

applying one or more joint-coding tools to the two or more related audio signals of the group to obtain jointly-coded audio signals;

combining the jointly-coded audio signals, an indication of the playback devices with which the jointly-coded audio signals are associated, and the information indicating at least one of a delay, a gain, and an equalization curve associated with the respective playback devices with which the jointly-coded audio signals are associated, into an independent block of an encoded bitstream.

2. The method according to claim 1, wherein the information indicates the delay and gain.

3. The method according to claim 1, wherein the delay, gain, and/or equalization curve associated with the respective playback device depend on a location of the respective playback device relative to a location of a listener.

4. The method according to any one of claims 1 to 3, wherein the delay, gain, and/or equalization curve associated with the respective playback device depend on a location of the respective playback device relative to locations of other playback devices.

5. The method of any one of claims 1 to 4, wherein the delay, gain, and/or equalization curve are dynamically variable.

6. The method according to claim 5, wherein the delay, gain, and/or equalization curve are adjusted in response to a change in the location of the listener.

7. The method of claim 5 or claim 6, wherein the delay, gain, and/or equalization curve are adjusted in response to a change in the location of the playback device.

8. The method of any one of claims 5 to 7, wherein the delay, gain, and/or equalization curve are adjusted in response to a change in the location of one or more of the other playback devices.

9. The method of any one of claims 1 to 8, further comprising determining, from the plurality of audio signals, an audio signal which is not part of the group of two or more related audio signals.

10. The method according to claim 9, further comprising, for the audio signal which is not part of the group of two or more related audio signals, applying the delay, gain, and/or equalization curve associated with the playback device with which the audio signal is associated.

11. The method according to claim 10, further comprising independently coding the audio signal which is not part of the group of two or more related audio signals, and combining the independently coded audio signal and an indication of the playback device with which the independently coded audio signal is associated into a separate independently decodable subset of the encoded bitstream.

12. The method according to claim 9, further comprising independently coding the audio signal which is not part of the group of two or more related audio signals, and combining the independently coded audio signal, an indication of the playback device with which the independently coded audio signal is associated, and an indication of the delay, gain, and/or equalization curve associated with the playback device with which the independently coded audio signal is associated into a separate independently decodable subset of the encoded bitstream.

13. A method for decoding one or more audio signals associated with a playback device from a frame of an encoded bitstream, wherein the frame comprises one or more independent blocks of encoded data, the method comprising:

identifying, from the encoded bitstream, an independent block of encoded data corresponding to the one or more audio signals associated with the playback device;

extracting, from the encoded bitstream, the identified independent block of encoded data;

determining that the extracted independent block of encoded data includes two or more jointly-coded audio signals;

applying one or more joint-decoding tools to the two or more jointly-coded audio signals to obtain the one or more audio signals associated with the playback device;

determining, from the extracted independent block of encoded data, at least one of a delay, a gain, and an equalization curve associated with the playback device;

applying the delay, gain, and/or equalization curve associated with the playback device to the one or more audio signals associated with the playback device.

14. The method of claim 13, wherein the determined delay, gain, and/or equalization curve associated with the playback device depend on a location of the playback device relative to a location of a listener.

15. The method of claim 13 or claim 14, wherein the determined delay, gain, and/or equalization curve associated with the playback device depend on a location of the playback device relative to other playback devices.

16. The method of any one of claims 13 to 15, wherein the determined delay, gain, and/or equalization curve of the playback device are dynamically variable.

17. The method of claim 16, wherein, when the determined delay, gain, and/or equalization curve associated with the playback device differ from a previously determined delay, gain, and/or equalization curve associated with the playback device, the method further comprises interpolating between the previously determined delay, gain, and/or equalization curve associated with the playback device and the determined delay, gain, and/or equalization curve associated with the playback device.

18. The method of claim 17, wherein the determined delay, gain, and/or equalization curve differs from the previously determined delay, gain, and/or equalization curve due to a change in the location of a listener.

19. The method of claim 17 or claim 18, wherein the determined delay, gain, and/or equalization curve differ from the previously determined delay, gain, and/or equalization curve due to a change in the location of the playback device.

20. The method of any one of claims 17 to 19, wherein the determined delay, gain, and/or equalization curve differ from the previously determined delay, gain, or equalization curve due to a change in the location of one or more of the other playback devices.

21. The method of any one of claims 13 to 20, wherein the frame of the encoded bitstream comprises two or more independent blocks of encoded data, the method further comprising:

determining that one or more of the independent blocks contain audio signals not associated with the playback device; and

ignoring the one or more independent blocks that contain audio signals not associated with the playback device.

22. The method of any one of claims 13 to 21, wherein applying one or more joint-decoding tools comprises identifying a subset of the jointly-coded audio signals that are associated with the playback device, and reconstructing only that subset of the jointly-coded audio signals to obtain the one or more audio signals associated with the playback device.

23. The method of any one of claims 13 to 21, wherein applying one or more joint-decoding tools comprises reconstructing each of the jointly-coded audio signals, identifying a subset of the reconstructed jointly-coded audio signals associated with the playback device, and obtaining the one or more audio signals associated with the playback device from the subset of the reconstructed jointly-coded audio signals associated with the playback device.

24. An apparatus configured to perform the method of any one of claims 1 to 23.

25. A non-transitory computer readable storage medium comprising a sequence of instructions which, when executed, cause one or more devices to perform the method of any one of claims 1 to 23.

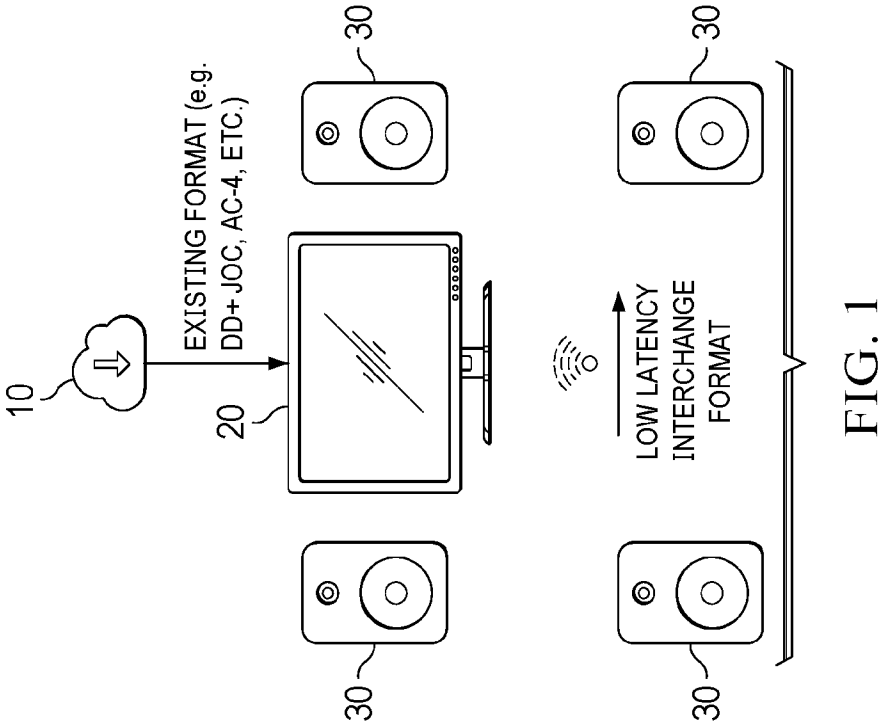
10
EXISTING FORMAT (e.g. DD+ JOC, AC-4, ETC.)
20
30
30
LOW LATENCY INTERCHANGE FORMAT
30
30

FIG. 1

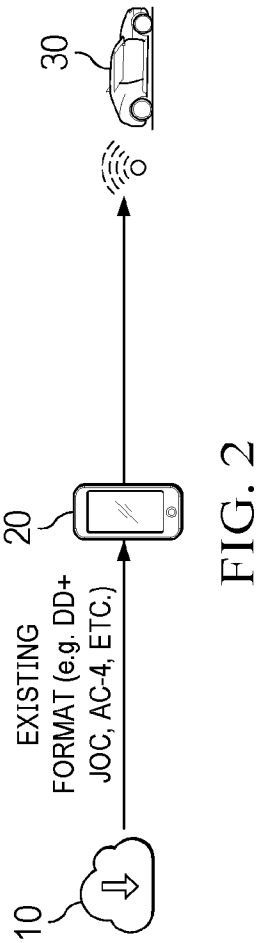
30
20
EXISTING FORMAT (e.g. DD+ JOC, AC-4, ETC.)
10

FIG. 2

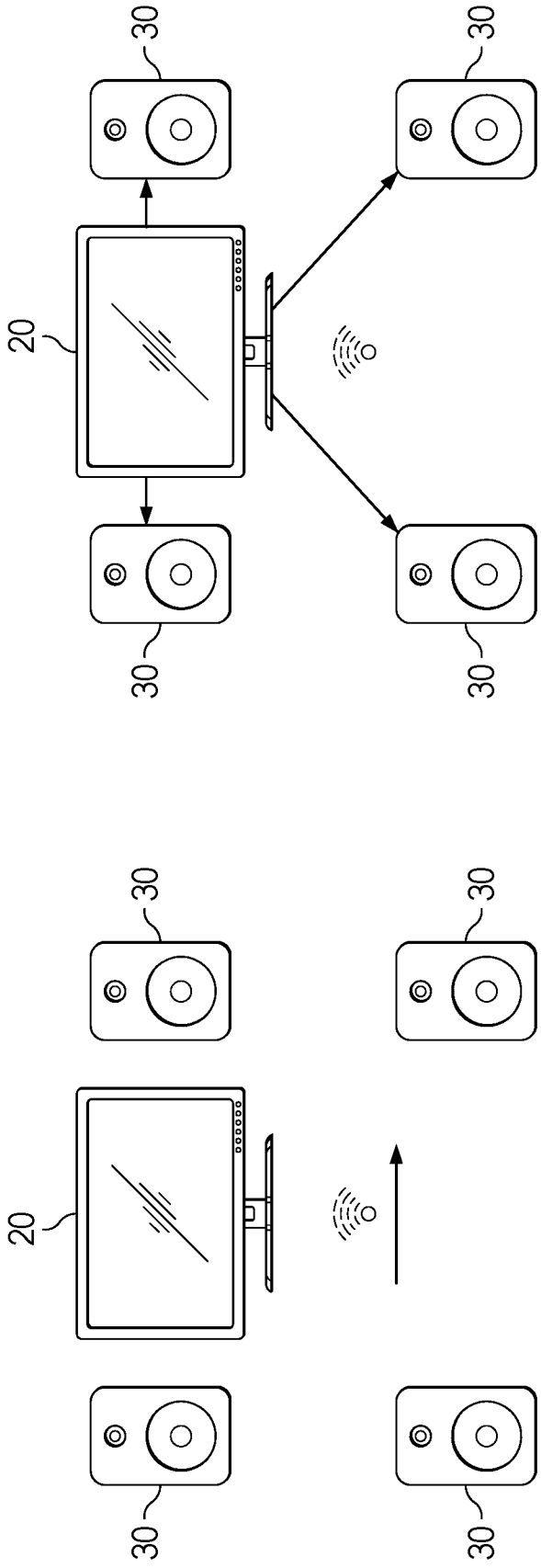

FIG. 3

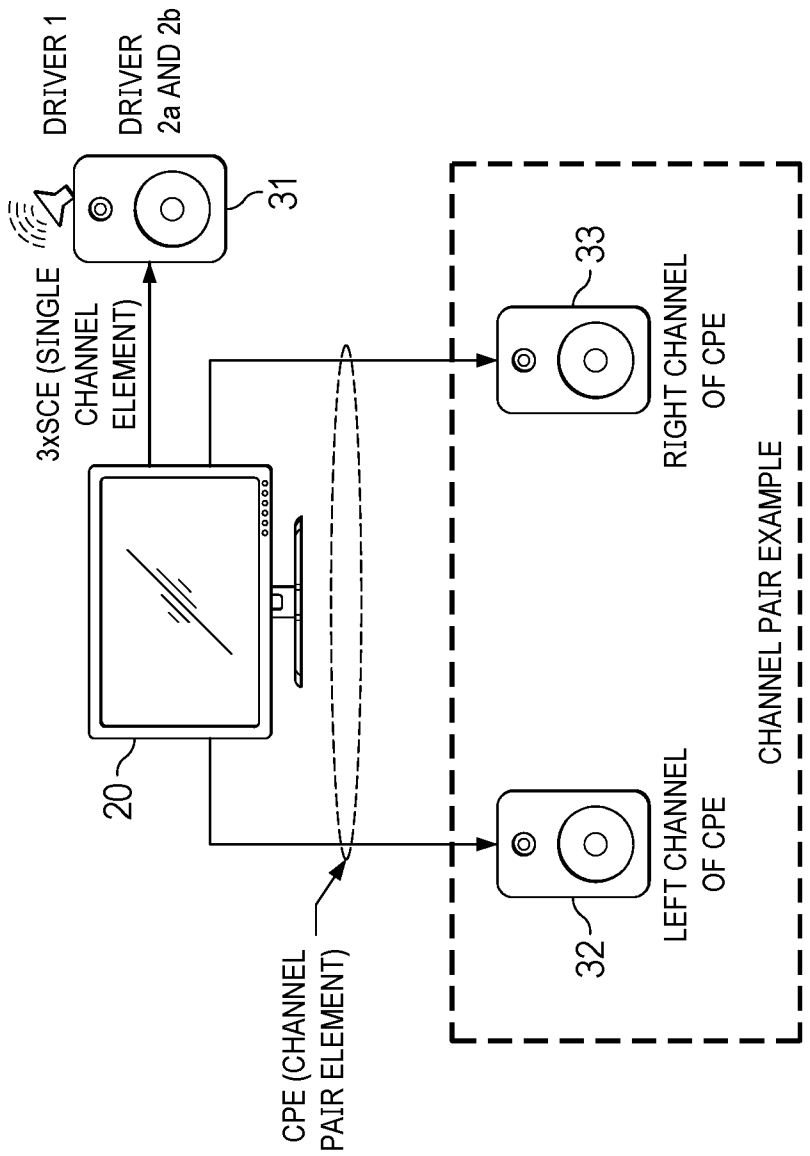
DRIVER 1
DRIVER
2a AND 2b
31
3xSCE (SINGLE
CHANNEL
ELEMENT)
20
CPE (CHANNEL
PAIR ELEMENT)
33
RIGHT CHANNEL
OF CPE
32
LEFT CHANNEL
OF CPE
CHANNEL PAIR EXAMPLE

FIG. 4

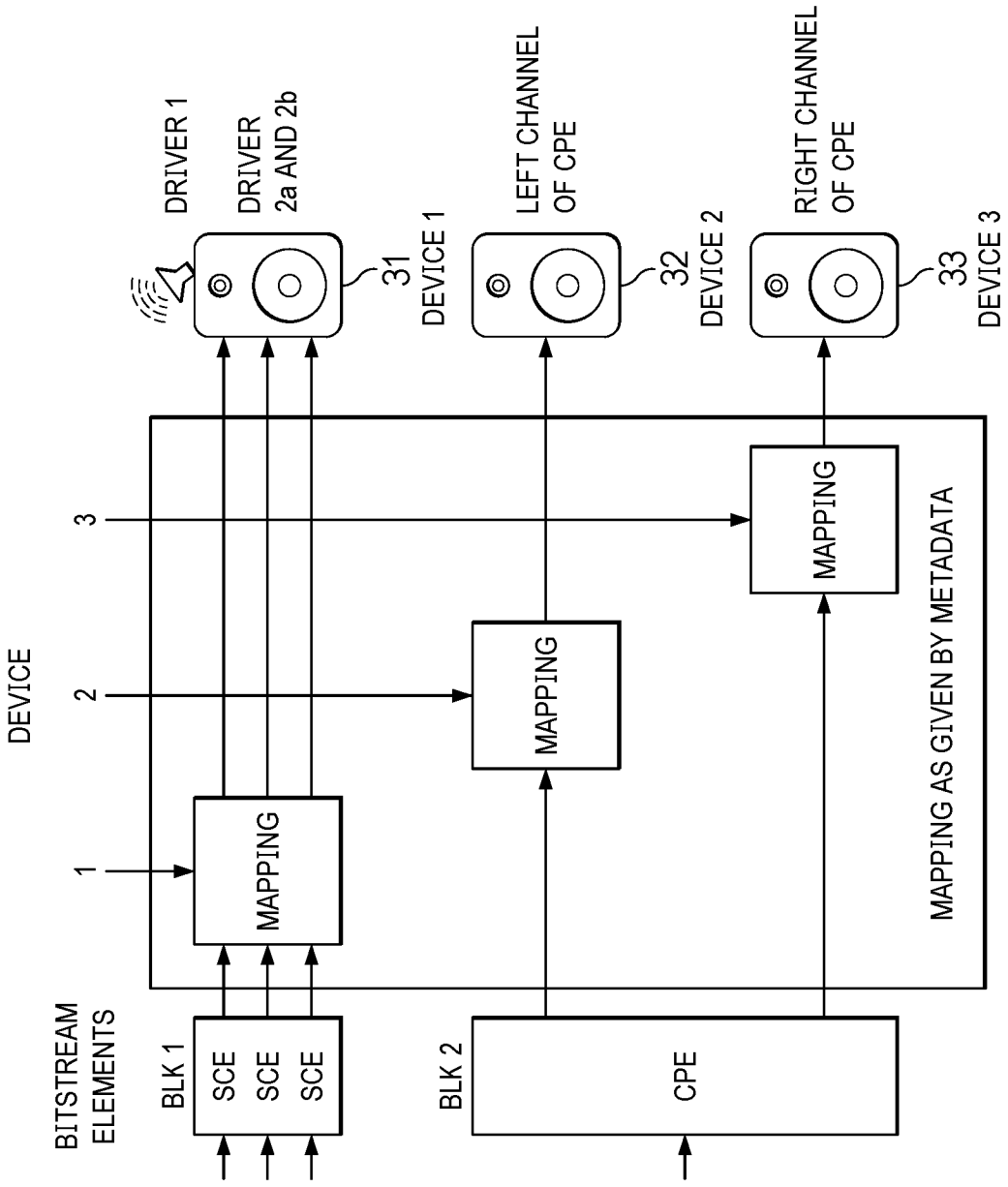
BITSTREAM ELEMENTS
BLK 1
SCE
SCE
SCE
BLK 2
CPE
DEVICE
1
2
3
MAPPING
MAPPING
MAPPING
MAPPING AS GIVEN BY METADATA
DRIVER 1
DRIVER 2a AND 2b
31
DEVICE 1
LEFT CHANNEL OF CPE
32
DEVICE 2
RIGHT CHANNEL OF CPE
33
DEVICE 3

FIG. 5

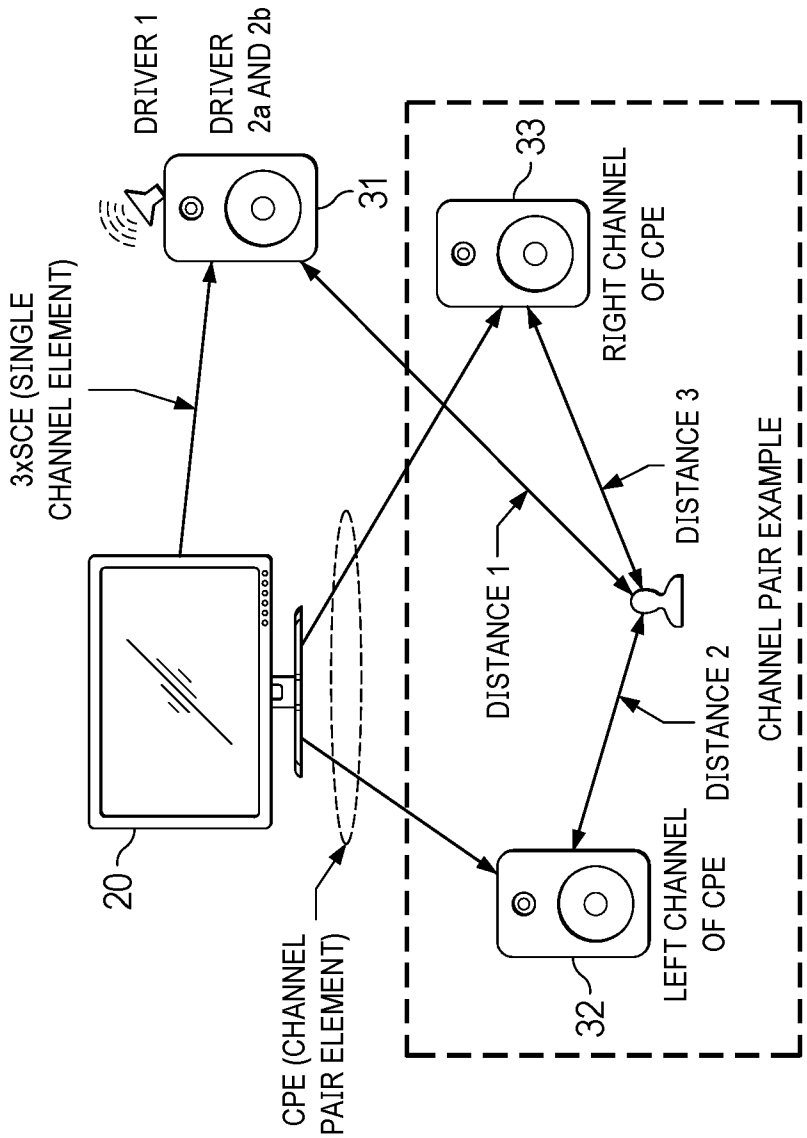
DRIVER 1
DRIVER 2a AND 2b
31
33
RIGHT CHANNEL OF CPE
3xSCE (SINGLE CHANNEL ELEMENT)
DISTANCE 3
DISTANCE 1
DISTANCE 2
CHANNEL PAIR EXAMPLE
20
CPE (CHANNEL PAIR ELEMENT)
32
LEFT CHANNEL OF CPE

FIG. 6

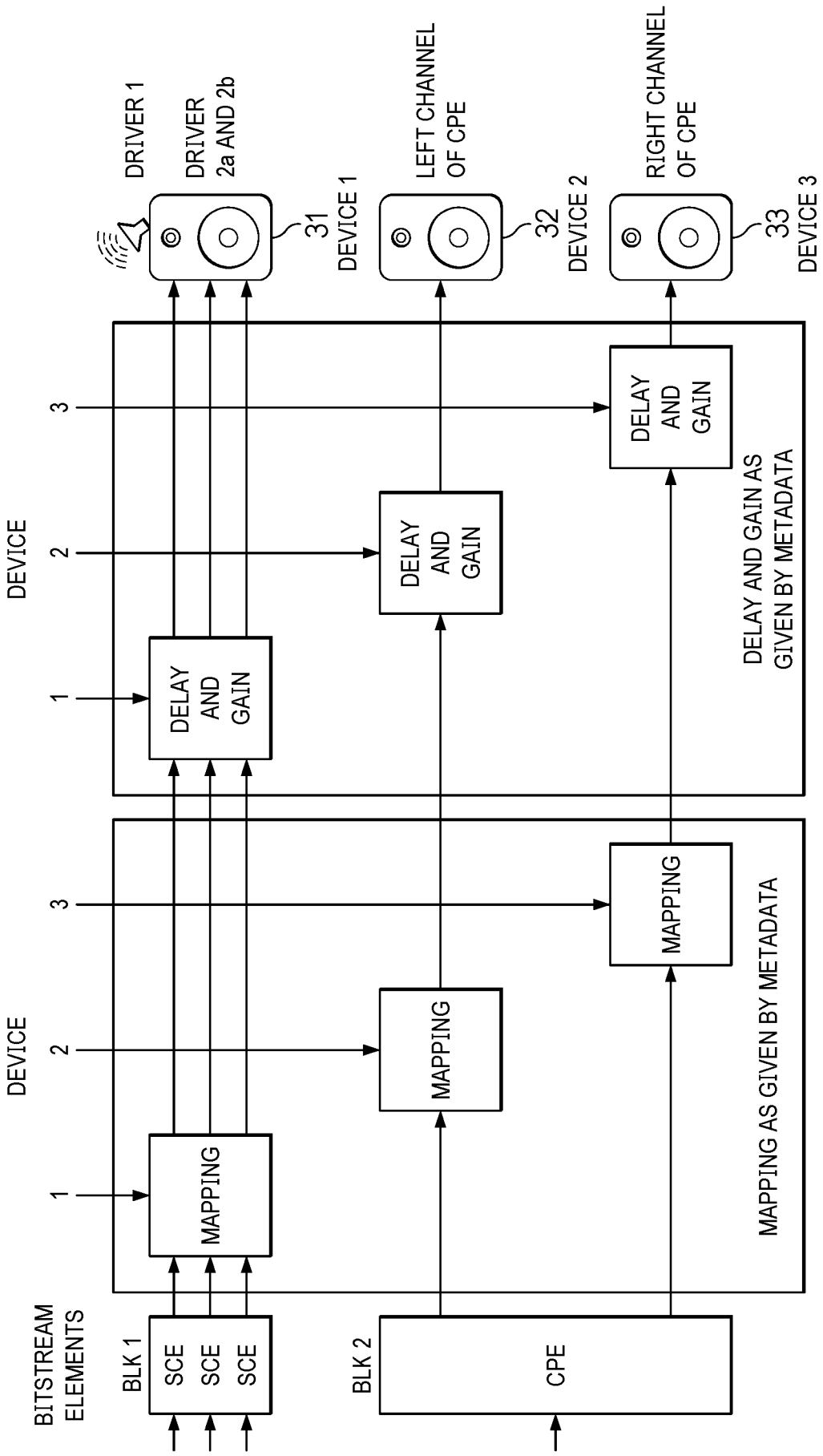
BITSTREAM ELEMENTS
BLK 1
SCE
SCE
SCE
BLK 2
CPE
DEVICE
1
2
3
MAPPING
MAPPING
MAPPING
MAPPING AS GIVEN BY METADATA
DELAY AND GAIN
DELAY AND GAIN
DELAY AND GAIN
DELAY AND GAIN AS GIVEN BY METADATA
DRIVER 1
DRIVER 2a AND 2b
31
DEVICE 1
LEFT CHANNEL OF CPE
32
DEVICE 2
RIGHT CHANNEL OF CPE
33
DEVICE 3

FIG. 7

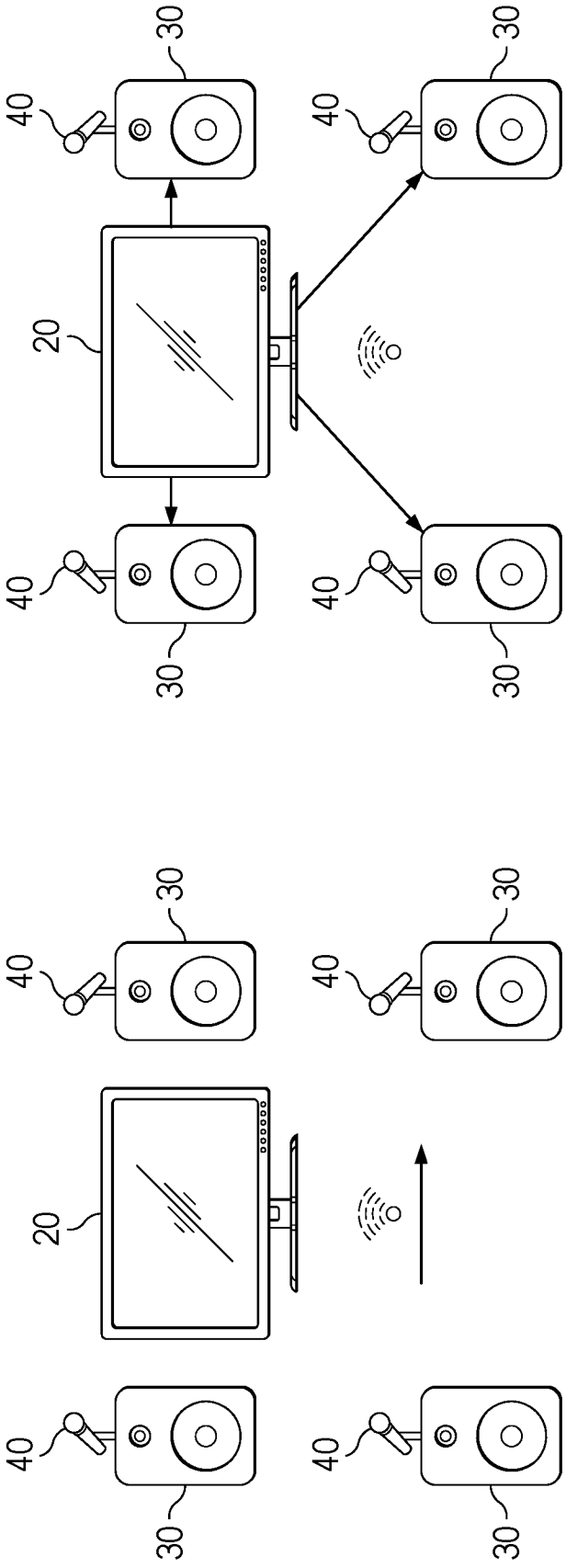

FIG. 8

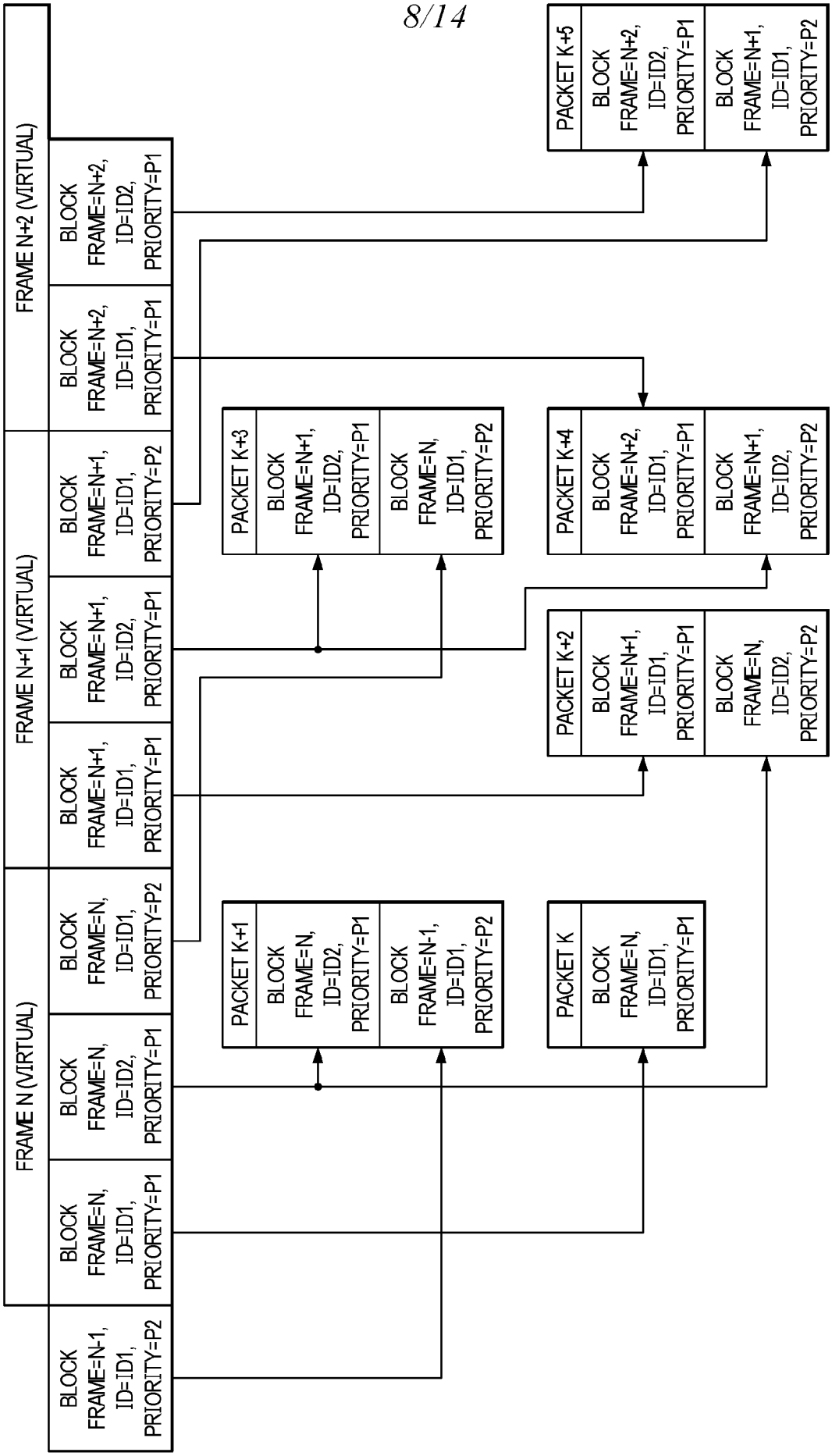
FRAME N (VIRTUAL)
FRAME N+1 (VIRTUAL)
FRAME N+2 (VIRTUAL)
BLOCK FRAME=N-1, ID=ID1, PRIORITY=P2
BLOCK FRAME=N, ID=ID1, PRIORITY=P1
BLOCK FRAME=N, ID=ID2, PRIORITY=P1
BLOCK FRAME=N, ID=ID1, PRIORITY=P2
BLOCK FRAME=N+1, ID=ID1, PRIORITY=P1
BLOCK FRAME=N+1, ID=ID2, PRIORITY=P1
BLOCK FRAME=N+1, ID=ID1, PRIORITY=P2
BLOCK FRAME=N+2, ID=ID1, PRIORITY=P1
BLOCK FRAME=N+2, ID=ID2, PRIORITY=P1
PACKET K
BLOCK FRAME=N, ID=ID1, PRIORITY=P1
PACKET K+1
BLOCK FRAME=N, ID=ID2, PRIORITY=P1
BLOCK FRAME=N-1, ID=ID1, PRIORITY=P2
PACKET K+2
BLOCK FRAME=N+1, ID=ID1, PRIORITY=P1
BLOCK FRAME=N, ID=ID2, PRIORITY=P2
PACKET K+3
BLOCK FRAME=N+1, ID=ID2, PRIORITY=P1
BLOCK FRAME=N, ID=ID1, PRIORITY=P2
PACKET K+4
BLOCK FRAME=N+2, ID=ID1, PRIORITY=P1
BLOCK FRAME=N+1, ID=ID2, PRIORITY=P2
PACKET K+5
BLOCK FRAME=N+2, ID=ID2, PRIORITY=P1
BLOCK FRAME=N+1, ID=ID1, PRIORITY=P2

FIG. 9

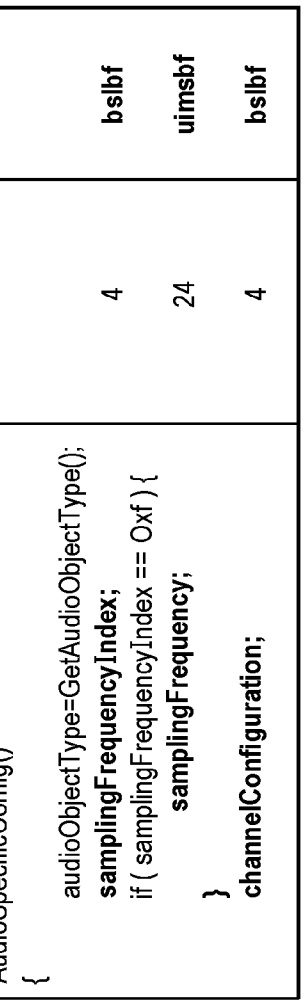
SYNTAX
NUMBER OF BITS
MNEMONIC
AudioSpecificConfig()
{
audioObjectType=GetAudioObjectType();
samplingFrequencyIndex;
4
bslbf
if (samplingFrequencyIndex == 0xf) {
samplingFrequency;
24
uimsbf
}
channelConfiguration;
4
bslbf
MPEG-4 PART 1 DecoderSpecificInfo()
MPEG-4 PART 3 AudioSpecificConfig()
• CHANNEL CONFIGURATION
• SAMPLE RATE
• ...
• AotSpecificConfig()
codecSpecificConfig()
codecSpecificConfig()
• SIGNALING OF STATIC CONFIGURATION ASPECTS (CHANNEL CONFIGURATIONS, MAPPING TO OUTPUT DEVICES, ETC.)
• STATIC METADATA
• ...

FIG. 10

raw_data_block_codec(configData)

- SIMILAR TO EXISTING RAW DATA BLOCKS IN MPEG-4
- WITH ADDITIONAL INFORMATION TO ENABLE SKIPPING OVER PART OF THE DATA BLOCK NOT NEEDED FOR A GIVEN DEVICE
- DYNAMIC METADATA

SYNTAX	NUMBER OF BITS	MNEMONIC

raw_data_block()
{
  while((id= **id_syn_ele**) != ID_END){
  switch (id) {
  case ID_SCE: single_channel_element();
  break;
  case ID_CPE: channel_pair_element();
  break;
  case ID_CCE: coupling_channel_element();
  break;
  case ID_LFE: life_channel_element();
  break;
  case ID_DSE: data_stream_element();
  break;
  case ID_PCE: program_config_element();
  break;
  case ID_FIL: fill_element();
	3	**uimsbf**

FIG. 11

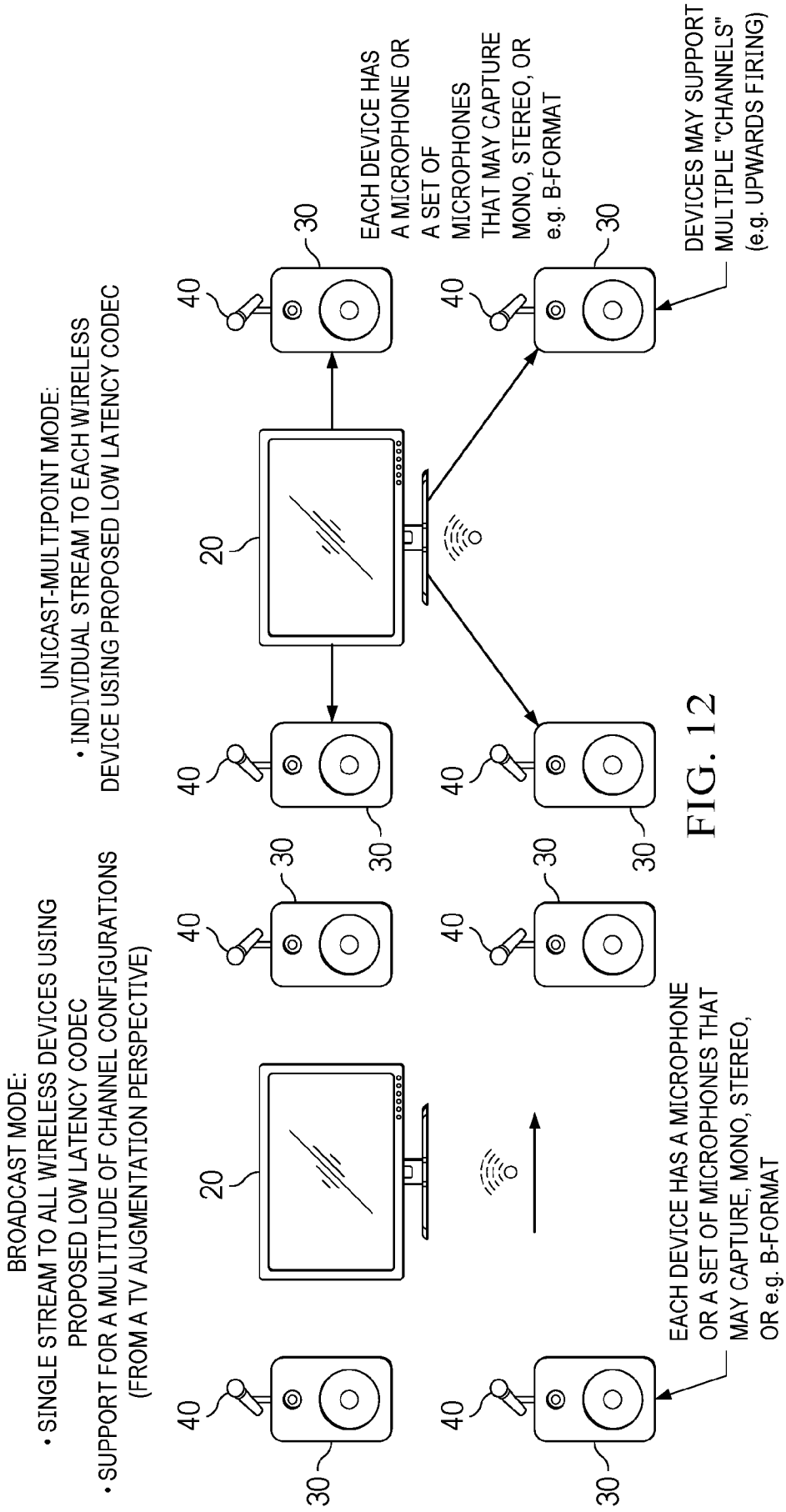
BROADCAST MODE:
• SINGLE STREAM TO ALL WIRELESS DEVICES USING PROPOSED LOW LATENCY CODEC
• SUPPORT FOR A MULTITUDE OF CHANNEL CONFIGURATIONS (FROM A TV AUGMENTATION PERSPECTIVE)
UNICAST-MULTIPOINT MODE:
• INDIVIDUAL STREAM TO EACH WIRELESS DEVICE USING PROPOSED LOW LATENCY CODEC
EACH DEVICE HAS A MICROPHONE OR A SET OF MICROPHONES THAT MAY CAPTURE MONO, STEREO, OR e.g. B-FORMAT
DEVICES MAY SUPPORT MULTIPLE "CHANNELS" (e.g. UPWARDS FIRING)
EACH DEVICE HAS A MICROPHONE OR A SET OF MICROPHONES THAT MAY CAPTURE, MONO, STEREO, OR e.g. B-FORMAT
20
30
40

FIG. 12

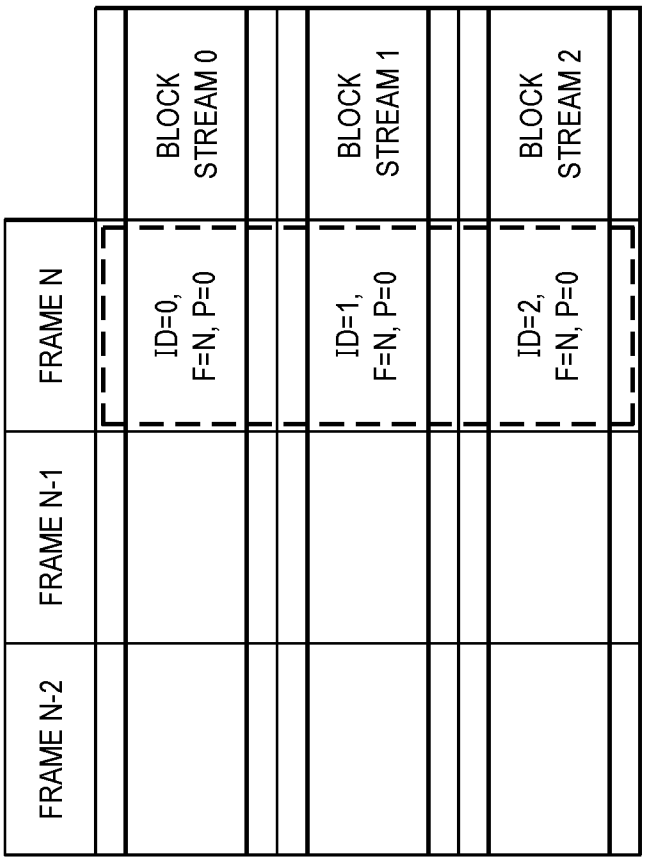
FRAME N-2
FRAME N-1
FRAME N
ID=0, F=N, P=0
ID=1, F=N, P=0
ID=2, F=N, P=0
BLOCK STREAM 0
BLOCK STREAM 1
BLOCK STREAM 2

FIG. 13

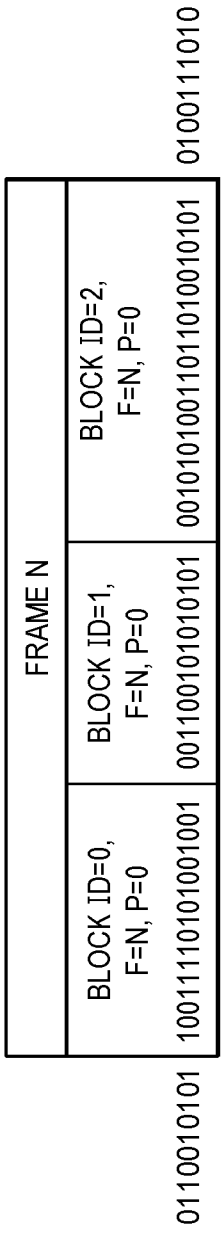
FRAME N
BLOCK ID=0, F=N, P=0
1001111010100100 1
BLOCK ID=1, F=N, P=0
0011001010101 01
BLOCK ID=2, F=N, P=0
0010101001101101001010 1
0110010101
0100111010

FIG. 14

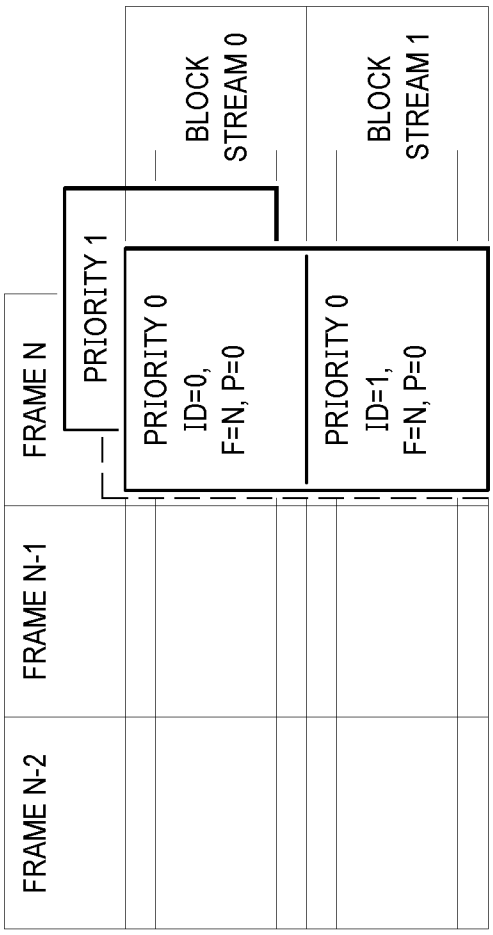
FRAME N-2
FRAME N-1
FRAME N
PRIORITY 1
PRIORITY 0
ID=0,
F=N, P=0
PRIORITY 0
ID=1,
F=N, P=0
BLOCK
STREAM 0
BLOCK
STREAM 1

FIG. 15

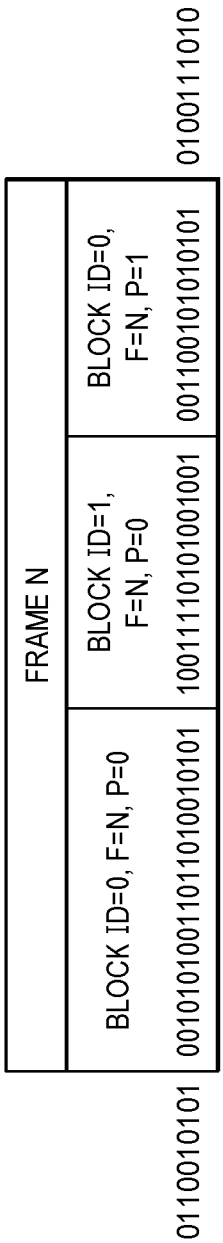
FRAME N
BLOCK ID=0, F=N, P=0
0010101001101101001010101
BLOCK ID=1,
F=N, P=0
1001111010100100 1
BLOCK ID=0,
F=N, P=1
0011001010101010
0110010101
0100111010

FIG. 16

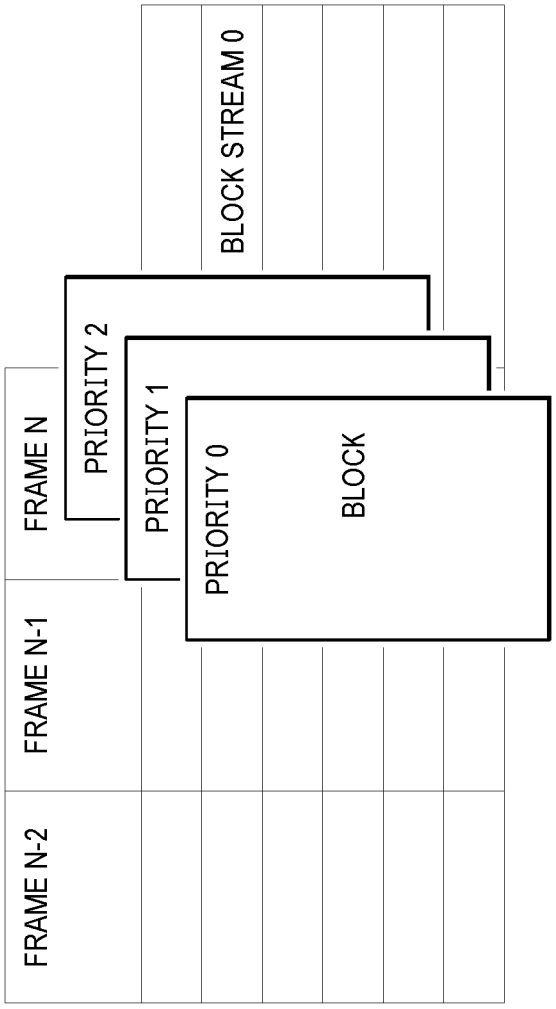
FRAME N-2
FRAME N-1
FRAME N
PRIORITY 2
PRIORITY 1
PRIORITY 0
BLOCK
BLOCK STREAM 0

FIG. 17

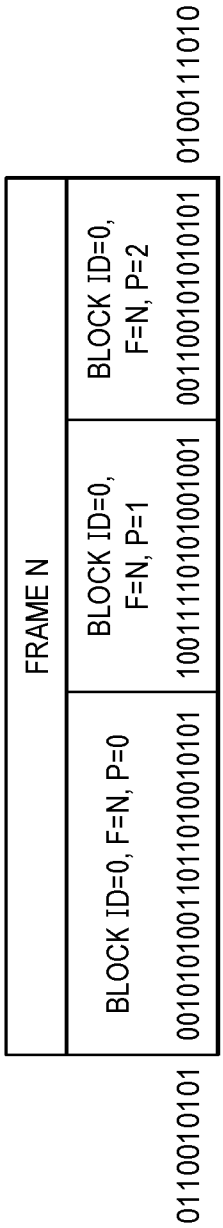
FRAME N
BLOCK ID=0, F=N, P=0
0010101001101101010010101
BLOCK ID=0, F=N, P=1
1001111010101001001
BLOCK ID=0, F=N, P=2
0011001010101010101
0110010101
0100111010

FIG. 18

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2023/074310

A. CLASSIFICATION OF SUBJECT MATTER
INV. G10L19/008 G10L19/16
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G10L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 2020/009841 A1 (QUALCOMM INC [US]) 9 January 2020 (2020-01-09) paragraph [0077] paragraph [0187] - paragraph [0188] paragraph [0167] - paragraph [0168] paragraph [0087]	1-25
A	EP 2 891 338 A1 (DOLBY LAB LICENSING CORP [US]) 8 July 2015 (2015-07-08) paragraph [0031] - paragraph [0033]	1-25
A	WO 2013/006338 A2 (DOLBY LAB LICENSING CORP [US]; ROBINSON CHARLES Q [US] ET AL.) 10 January 2013 (2013-01-10) paragraph [0109] figure 2	1-25

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance;; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance;; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
19 December 2023	05/01/2024

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Ziegler, Stefan

1

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/US2023/074310

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 2020009841 A1	09-01-2020	CN 112400204 A	23-02-2021
		EP 3818522 A1	12-05-2021
		TW 202007189 A	01-02-2020
		US 2020013426 A1	09-01-2020
		WO 2020009841 A1	09-01-2020
EP 2891338 A1	08-07-2015	CN 104604257 A	06-05-2015
		EP 2891338 A1	08-07-2015
		EP 3253079 A1	06-12-2017
		EP 4207817 A1	05-07-2023
		HK 1205845 A1	24-12-2015
		HK 1248046 A1	05-10-2018
		JP 6085029 B2	22-02-2017
		JP 2015530825 A	15-10-2015
		US 2015223002 A1	06-08-2015
		US 2018077511 A1	15-03-2018
		US 2019349701 A1	14-11-2019
		US 2020382892 A1	03-12-2020
		US 2022030373 A1	27-01-2022
		WO 2014036121 A1	06-03-2014
WO 2013006338 A2	10-01-2013	AR 086775 A1	22-01-2014
		AU 2012279357 B2	14-01-2016
		AU 2016202227 A1	05-05-2016
		AU 2018203734 A1	21-06-2018
		AU 2019204012 A1	11-07-2019
		AU 2020226984 A1	17-09-2020
		AU 2021258043 A1	25-11-2021
		AU 2023200502 A1	02-03-2023
		BR 112013033386 A2	24-01-2017
		BR 122020001361 B1	19-04-2022
		CA 2837893 A1	10-01-2013
		CA 2973703 A1	10-01-2013
		CA 3157717 A1	10-01-2013
		CN 103650539 A	19-03-2014
		CN 105792086 A	20-07-2016
		DK 2727383 T3	25-05-2021
		EP 2727383 A2	07-05-2014
		EP 3893521 A1	13-10-2021
		ES 2871224 T3	28-10-2021
		HK 1219604 A1	07-04-2017
		HU E054452 T2	28-09-2021
		IL 230046 A	30-06-2016
		IL 265741 A	30-06-2019
		IL 277736 A	30-11-2020
		IL 284585 A	31-08-2021
		IL 291043 A	01-05-2022
		IL 295733 A	01-10-2022
		IL 302167 A	01-06-2023
		JP 5912179 B2	27-04-2016
		JP 6174184 B2	02-08-2017
		JP 6486995 B2	20-03-2019
		JP 6523585 B1	05-06-2019
		JP 6637208 B2	29-01-2020
		JP 6759442 B2	23-09-2020
		JP 6821854 B2	27-01-2021
		JP 6882618 B2	02-06-2021
		JP 7009664 B2	25-01-2022

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/US2023/074310

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
		JP 7348320 B2	20-09-2023
		JP 2014522155 A	28-08-2014
		JP 2016165117 A	08-09-2016
		JP 2017215592 A	07-12-2017
		JP 2019095813 A	20-06-2019
		JP 2019144583 A	29-08-2019
		JP 2020057014 A	09-04-2020
		JP 2021005876 A	14-01-2021
		JP 2021073496 A	13-05-2021
		JP 2021131562 A	09-09-2021
		JP 2022058569 A	12-04-2022
		JP 2023164976 A	14-11-2023
		KR 20140017682 A	11-02-2014
		KR 20150013913 A	05-02-2015
		KR 20180035937 A	06-04-2018
		KR 20190014601 A	12-02-2019
		KR 20190086785 A	23-07-2019
		KR 20200058593 A	27-05-2020
		KR 20200137034 A	08-12-2020
		KR 20220081385 A	15-06-2022
		MY 165933 A	18-05-2018
		PL 2727383 T3	02-08-2021
		RU 2741738 C1	28-01-2021
		RU 2013158054 A	10-08-2015
		RU 2017112527 A	24-01-2019
		SG 10201604679U A	28-07-2016
		TW 201325269 A	16-06-2013
		TW 201642673 A	01-12-2016
		TW 201811070 A	16-03-2018
		TW 201909658 A	01-03-2019
		TW 202139720 A	16-10-2021
		UA 124570 C2	13-10-2021
		US 2014133683 A1	15-05-2014
		US 2016021476 A1	21-01-2016
		US 2016381483 A1	29-12-2016
		US 2017215020 A1	27-07-2017
		US 2018027352 A1	25-01-2018
		US 2018192230 A1	05-07-2018
		US 2018324543 A1	08-11-2018
		US 2019104376 A1	04-04-2019
		US 2019306652 A1	03-10-2019
		US 2020145779 A1	07-05-2020
		US 2021219091 A1	15-07-2021
		US 2023045090 A1	09-02-2023
		WO 2013006338 A2	10-01-2013