|  |  |
| --- | --- |
| 3GPP TS 26.251 V1.0.2 (2024-06) | |
| Technical Specification | |
| 3rd Generation Partnership Project;  Technical Specification Group Services and System Aspects;  Codec for Immersive Voice and Audio Services (IVAS);  C code (fixed-point)  (Release 18) | |
|  | |
|  |  |
|  | |
| The present document has been developed within the 3rd Generation Partnership Project (3GPP TM) and may be further elaborated for the purposes of 3GPP. The present document has not been subject to any approval process by the 3GPPOrganizational Partners and shall not be implemented. This Specification is provided for future development work within 3GPPonly. The Organizational Partners accept no liability for any use of this Specification. Specifications and Reports for implementation of the 3GPP TM system should be obtained via the 3GPP Organizational Partners' Publications Offices. | |

|  |
| --- |
|  |
| ***3GPP***  Postal address  3GPP support office address  650 Route des Lucioles - Sophia Antipolis  Valbonne - FRANCE  Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16  Internet  https://www.3gpp.org |
| ***Copyright Notification***  No part may be reproduced except as authorized by written permission. The copyright and the foregoing restriction extend to reproduction in all media.  © 2024, 3GPP Organizational Partners (ARIB, ATIS, CCSA, ETSI, TSDSI, TTA, TTC).  All rights reserved.  UMTS™ is a Trade Mark of ETSI registered for the benefit of its members  3GPP™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners LTE™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners  GSM® and the GSM logo are registered and owned by the GSM Association |

Contents

Foreword 4

1 Scope 6

2 References 6

3 Definitions of terms, symbols and abbreviations 6

3.1 Terms 6

3.2 Symbols 7

3.3 Abbreviations 7

4 C code structure 7

4.1 Contents of the C source code 7

4.2 Program execution 8

5 File Formats 8

5.1 Audio Input/output file format 8

5.2 Rate switching profile (encoder input) 8

5.3 Bandwidth switching profile (encoder input) 8

5.4 Channel-aware configuration file (encoder input and decoder output) 8

5.5 Object based audio metadata file (encoder/renderer input and decoder output) 8

5.6 Metadata-assisted spatial audio (MASA) metadata file (encoder/renderer input and decoder output) 9

5.7 Parameter bitstream file (encoder output / decoder input) 9

5.8 VoIP parameter bitstream file (decoder input) 9

5.9 JBM trace file (decoder output) 9

5.10 HRTF filter file (decoder/renderer input) 9

5.11 Head rotation trajectory file (decoder/renderer input) 9

5.12 Reference rotation/vector file (decoder/renderer input) 9

5.13 External orientation file (decoder/renderer input) 9

5.14 Renderer config file (decoder/renderer input) 9

5.15 Scene description file (renderer input) 9

Annex A (informative): Change history 10

# Foreword

This Technical Specification has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

x the first digit:

1 presented to TSG for information;

2 presented to TSG for approval;

3 or greater indicates TSG approved document under change control.

y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.

z the third digit is incremented when editorial only changes have been incorporated in the document.

In the present document, modal verbs have the following meanings:

**shall** indicates a mandatory requirement to do something

**shall not** indicates an interdiction (prohibition) to do something

The constructions "shall" and "shall not" are confined to the context of normative provisions, and do not appear in Technical Reports.

The constructions "must" and "must not" are not used as substitutes for "shall" and "shall not". Their use is avoided insofar as possible, and they are not used in a normative context except in a direct citation from an external, referenced, non-3GPP document, or so as to maintain continuity of style when extending or modifying the provisions of such a referenced document.

**should** indicates a recommendation to do something

**should not** indicates a recommendation not to do something

**may** indicates permission to do something

**need not** indicates permission not to do something

The construction "may not" is ambiguous and is not used in normative elements. The unambiguous constructions "might not" or "shall not" are used instead, depending upon the meaning intended.

**can** indicates that something is possible

**cannot** indicates that something is impossible

The constructions "can" and "cannot" are not substitutes for "may" and "need not".

**will** indicates that something is certain or expected to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document

**will not** indicates that something is certain or expected not to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document

**might** indicates a likelihood that something will happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

**might not** indicates a likelihood that something will not happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

In addition:

**is** (or any other verb in the indicative mood) indicates a statement of fact

**is not** (or any other negative verb in the indicative mood) indicates a statement of fact

The constructions "is" and "is not" do not indicate requirements.

# 1 Scope

Attached to this document is an electronic copy of the fixed-point C code for the Immersive Voice and Audio Services (IVAS) Codec. This C code is the unique alternative reference specification besides the floating-point C code for the IVAS Codec (3GPP TS 26.258) for a standard compliant implementation of the IVAS Codec (3GPP TS 26.253), Rendering (3GPP TS 26.254), Error Concealment of Lost Packets (3GPP TS 26.255) and Jitter Buffer Management (JBM) (3GPP TS 26.256).

The bit-exact fixed-point C code is the preferred implementation for all applications, but the floating-point codec as specified in 3GPP TS 26.258 may be used instead of the fixed-point codec when the implementation platform is better suited for a floating-point implementation.

Requirements for any implementation of the IVAS codec to be standard compliant are specified in 3GPP TS 26.252 (Test sequences).

# 2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non‑specific.

- For a specific reference, subsequent revisions do not apply.

- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

[2] 3GPP TS 26.253: "Codec for Immersive Voice and Audio Services - Detailed Algorithmic Description incl. RTP payload format and SDP parameter definitions".

[3] 3GPP TS 26.254: "Codec for Immersive Voice and Audio Services - Rendering".

[4] 3GPP TS 26.255: "Codec for Immersive Voice and Audio Services - Error concealment of lost packets".

[5] 3GPP TS 26.256: "Codec for Immersive Voice and Audio Services - Jitter Buffer Management".

[6] 3GPP TS 26.252: "Codec for Immersive Voice and Audio Services – Test Sequences".

[7] 3GPP TS 26.258: "Codec for Immersive Voice and Audio Services – C code (floating-point)".

# 3 Definitions of terms, symbols and abbreviations

## 3.1 Terms

For the purposes of the present document, the terms given in TR 21.905 [1] and the following apply. A term defined in the present document takes precedence over the definition of the same term, if any, in TR 21.905 [1].

## 3.2 Symbols

Void.

## 3.3 Abbreviations

For the purposes of the present document, the abbreviations given in TR 21.905 [1] and the following apply. An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in TR 21.905 [1].

HRTF Head Related Transfer Function

IVAS Immersive Voice and Audio Services

ISAR Immersive Audio for Split Rendering Scenarios

JBM Jitter Buffer Management

MASA Metadata-Assisted Spatial Audio

# 4 C code structure

This clause gives an overview of the structure of the floating-point C code and provides an overview of the contents and organization of the C code attached to the present document.

The C code has been verified on the following platforms:

- IBM PC compatible computers with Windows 10 operating systems and Microsoft Visual C++ 2017 compiler, 32-bit.

C was selected as the programming language because portability was desirable.

## 4.1 Contents of the C source code

The C code is organized as listed in Table 1:

Table 1: Source code directory structure

|  |  |
| --- | --- |
| Directory | Description |
| readme.txt | information on how to compile and use |
| Makefile | UNIX style encoder Makefile |
| Workspace\_msvc/ | Directory for the MSVC 2017 (or newer) project files |
| apps/ | Source code files used solely for the encoder/decoder/renderer applications; these applications make use of the libraries built from lib\_com, lib\_dec, lib\_enc, lib\_rend, and lib\_util |
| lib\_com/ | Source code files used both in encoder and decoder |
| lib\_dec/ | Source code files used solely in the decoder |
| lib\_enc/ | Source code files used solely in the encoder |
| lib\_isar/ | Source code files used solely for split rendering |
| lib\_lc3plus/ | Source code files used solely for split rendering |
| lib\_rend/ | Source code files used solely in the renderer |
| lib\_util/ | Source code files solely for utility functions used by the applications |

The distributed files with suffix "c" contain the source code and the files with suffix "h" are the header files. The ROM data is contained in files named "rom\_\*" and “ivas\_rom\_\*” with suffix "c".

Makefiles are provided for the platforms in which the C code has been verified (listed above). Once the software is installed, this directory will have a compiled version of the encoder (named IVAS\_cod), the decoder (named IVAS\_dec),the renderer (named IVAS\_rend) and the split rendering post-renderer (named ISAR\_post\_rend).

Note: As of the present version of this specification, IVAS fixed point code exists only with regards to the ISAR split rendering feature. This code is available as a software patch to the IVAS floating-point code; the software patch is provided as electronic attachment to this specification. Applying the patch allows running the ISAR split rendering feature in fixed-point code within the IVAS floating-point software framework. Future versions of this specification are expected to provide fixed-point code for the complete IVAS codec. In that case, the patch to the IVAS floating-point code will become obsolete.

## 4.2 Program execution

The codec for Immersive Voice and Audio Services is implemented in three programs:

*-* IVAS\_cod: encoder;

*-* IVAS\_dec: decoder;

*-* IVAS\_rend: renderer,

- ISAR\_post\_rend: split rendering post-renderer.

The programs should be called like:

- IVAS\_cod [encoder options] <input file> <bitstream file>;

- IVAS\_dec [decoder options] <bitstream file> <output file>;

- IVAS\_rend [renderer options] -i <input file> -if <input format> -o <output file> -of <output format>,

- ISAR\_post\_rend [post-renderer options] -i < bitstream file or input file> -if <input format> -o <output file>.

The input and output files contain 16-bit linear encoded PCM samples (headerless or in WAVE format) and the bitstream file contains encoded data.

The encoder, decoder, and renderer options will be explained by running the programs without any input arguments. See the file readme.txt for more information on how to run the *IVAS\_cod*, *IVAS\_dec*, *IVAS\_rend* and *ISAR\_post\_rend* programs.

# 5 File Formats

This clause describes the file formats used by the encoder and decoder programs. The test sequences defined in [6] also use the file formats described here.

## 5.1 Audio Input/output file format

Audio input/output file format is in accordance with TS 21.258 [7] clause 5.1.

## 5.2 Rate switching profile (encoder input)

Rate switching profile (encoder input) is in accordance with TS 21.258 [7] clause 5.2.

## 5.3 Bandwidth switching profile (encoder input)

Bandwidth switching profile (encoder input) is in accordance with TS 21.258 [7] clause 5.3.

## 5.4 Channel-aware configuration file (encoder input and decoder output)

Channel-aware configuration file (encoder input and decoder output) is in accordance with TS 21.258 [7] clause 5.4.

## 5.5 Object based audio metadata file (encoder/renderer input and decoder output)

Object based audio metadata file (encoder/renderer input and decoder output) is in accordance with TS 21.258 [7] clause 5.5.

## 5.6 Metadata-assisted spatial audio (MASA) metadata file (encoder/renderer input and decoder output)

Metadata-assisted spatial audio (MASA) metadata file (encoder/renderer input and decoder output) is in accordance with TS 21.258 [7] clause 5.6.

## 5.7 Parameter bitstream file (encoder output / decoder input)

Parameter bitstream file (encoder output / decoder input) is in accordance with TS 21.258 [7] clause 5.7.

## 5.8 VoIP parameter bitstream file (decoder input)

VoIP parameter bitstream file (decoder input) is in accordance with TS 21.258 [7] clause 5.8.

## 5.9 JBM trace file (decoder output)

JBM trace file (decoder output) is in accordance with TS 21.258 [7] clause 5.9.

## 5.10 HRTF filter file (decoder/renderer input)

HRTF filter file (decoder/renderer input) is in accordance with TS 21.258 [7] clause 5.10.

## 5.11 Head rotation trajectory file (decoder/renderer input)

Head rotation trajectory file (decoder/renderer input) is in accordance with TS 21.258 [7] clause 5.11.

## 5.12 Reference rotation/vector file (decoder/renderer input)

Reference rotation/vector file (decoder/renderer input) is in accordance with TS 21.258 [7] clause 5.12.

## 5.13 External orientation file (decoder/renderer input)

External orientation file (decoder/renderer input) is in accordance with TS 21.258 [7] clause 5.13.

## 5.14 Renderer config file (decoder/renderer input)

Renderer config file (decoder/renderer input) is in accordance with TS 21.258 [7] clause 5.14.

## 5.15 Scene description file (renderer input)

Scene description file (renderer input) is in accordance with TS 21.258 [7] clause 5.15.

## 5.16 Split rendering pose correction file (decoder/renderer output, post-renderer input)

Split rendering pose correction file (decoder/renderer output, post-renderer input) is in accordance with TS 21.258 [7] clause 5.16.

## 5.17 Split rendering bitstream file (decoder/renderer output, post-renderer input)

Split rendering bitstream file (decoder/renderer output, post-renderer input) is in accordance with TS 21.258 [7] clause 5.17.

Annex A (informative):  
Change history

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Change history | | | | | | | |
| Date | Meeting | TDoc | CR | Rev | Cat | Subject/Comment | New version |
| 01-2024 | SA4#127 | SA4-24xxxx |  |  |  | Presented to Audio SWG for information | 0.0.1 |
| 03-2024 | SA#103 | SP-240023 |  |  |  | Version 1.0.0 created by MCC | 1.0.0 |
| 05-2024 | SA4#128 | SA4-241067 |  |  |  | Integrated changes wrt ISAR integration; presented to Audio SWG | 1.0.1 |
| 06-2024 | SA#104 |  |  |  |  | Change of spec title as approved by TSG SA in SP-240917 | 1.0.2 |