|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Fond-Rec_e | | **International Telecommunication Union** | | |
|  | |  | | |
| **ITU-T** | **G.722.2** | |
| TELECOMMUNICATION STANDARDIZATION SECTOR OF ITU | | **Annex C**  (12/2017) |
|  | SERIES G: TRANSMISSION SYSTEMS AND MEDIA, DIGITAL SYSTEMS AND NETWORKS  Digital terminal equipments – Coding of voice and audio signals | | | |
|  | Wideband coding of speech at around 16 kbit/s using Adaptive Multi-Rate Wideband (AMR-WB)  **Annex C: Fixed-point C-code** | | | |
|  | Recommendation ITU‑T G.722.2  – Annex C | | | |



ITU-T G-SERIES RECOMMENDATIONS

**TRANSMISSION SYSTEMS AND MEDIA, DIGITAL SYSTEMS AND NETWORKS**

|  |  |
| --- | --- |
|  |  |
| INTERNATIONAL TELEPHONE CONNECTIONS AND CIRCUITS | G.100–G.199 |
| GENERAL CHARACTERISTICS COMMON TO ALL ANALOGUE CARRIER-TRANSMISSION SYSTEMS | G.200–G.299 |
| INDIVIDUAL CHARACTERISTICS OF INTERNATIONAL CARRIER TELEPHONE SYSTEMS ON METALLIC LINES | G.300–G.399 |
| GENERAL CHARACTERISTICS OF INTERNATIONAL CARRIER TELEPHONE SYSTEMS ON RADIO-RELAY OR SATELLITE LINKS AND INTERCONNECTION WITH METALLIC LINES | G.400–G.449 |
| COORDINATION OF RADIOTELEPHONY AND LINE TELEPHONY | G.450–G.499 |
| TRANSMISSION MEDIA AND OPTICAL SYSTEMS CHARACTERISTICS | G.600–G.699 |
| DIGITAL TERMINAL EQUIPMENTS | G.700–G.799 |
| General | G.700–G.709 |
| **Coding of voice and audio signals** | **G.710–G.729** |
| Principal characteristics of primary multiplex equipment | G.730–G.739 |
| Principal characteristics of second order multiplex equipment | G.740–G.749 |
| Principal characteristics of higher order multiplex equipment | G.750–G.759 |
| Principal characteristics of transcoder and digital multiplication equipment | G.760–G.769 |
| Operations, administration and maintenance features of transmission equipment | G.770–G.779 |
| Principal characteristics of multiplexing equipment for the synchronous digital hierarchy | G.780–G.789 |
| Other terminal equipment | G.790–G.799 |
| DIGITAL NETWORKS | G.800–G.899 |
| DIGITAL SECTIONS AND DIGITAL LINE SYSTEM | G.900–G.999 |
| MULTIMEDIA QUALITY OF SERVICE AND PERFORMANCE – GENERIC AND USER-RELATED ASPECTS | G.1000–G.1999 |
| TRANSMISSION MEDIA CHARACTERISTICS | G.6000–G.6999 |
| DATA OVER TRANSPORT – GENERIC ASPECTS | G.7000–G.7999 |
| PACKET OVER TRANSPORT ASPECTS | G.8000–G.8999 |
| ACCESS NETWORKS | G.9000–G.9999 |
|  |  |

*For further details, please refer to the list of ITU-T Recommendations.*

|  |
| --- |
| Recommendation ITU-T G.722.2  Wideband coding of speech at around 16 kbit/s using Adaptive  Multi-Rate Wideband (AMR-WB)  Annex C  Fixed-point C-code |

|  |
| --- |
| Summary  Annex C to Recommendation ITU-T G.722.2 specifies the bit-exact ANSI C-code implementation of the AMR-WB algorithm specified in Recommendation ITU-T G.722.2, its Annexes A and B, and its Appendix I (non-normative).  This annex includes an electronic attachment containing the C-code of the G.722.2 AMR-WB speech transcoder. The C-code has been updated to harmonize with the AMR-WB coder in 3GPP specification TS 26.173 V14.0.0 (2017-04). |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| History   |  |  |  |  |  | | --- | --- | --- | --- | --- | | Edition | Recommendation | Approval | Study Group | Unique ID[[1]](#footnote-1)\* | | 1.0 | ITU-T G.722.2 | 2002-01-13 | 16 | [11.1002/1000/5650](http://handle.itu.int/11.1002/1000/5650) | | 1.1 | ITU-T G.722.2 Annex C | 2002-01-13 | 16 | [11.1002/1000/5662](http://handle.itu.int/11.1002/1000/5662) | | 1.3 | ITU-T G.722.2 Annex C | 2003-07-29 | 16 | [11.1002/1000/6865](http://handle.itu.int/11.1002/1000/6865) | | 1.4 | ITU-T G.722.2 Annex D | 2002-01-13 | 16 | [11.1002/1000/5663](http://handle.itu.int/11.1002/1000/5663) | | 1.4 | ITU-T G.722.2 Annex C | 2004-03-15 | 16 | [11.1002/1000/7206](http://handle.itu.int/11.1002/1000/7206) | | 2.0 | ITU-T G.722.2 | 2003-07-29 | 16 | [11.1002/1000/6506](http://handle.itu.int/11.1002/1000/6506) | | 2.1 | ITU-T G.722.2 Annex A | 2002-01-13 | 16 | [11.1002/1000/5660](http://handle.itu.int/11.1002/1000/5660) | | 2.2 | ITU-T G.722.2 Annex B | 2002-01-13 | 16 | [11.1002/1000/5661](http://handle.itu.int/11.1002/1000/5661) | | 2.5 | ITU-T G.722.2 Annex E | 2002-01-13 | 16 | [11.1002/1000/5664](http://handle.itu.int/11.1002/1000/5664) | | 2.6 | ITU-T G.722.2 Annex D | 2003-07-29 | 16 | [11.1002/1000/6871](http://handle.itu.int/11.1002/1000/6871) | | 2.6 | ITU-T G.722.2 Annex E (2002) Cor. 1 | 2003-07-29 | 16 | [11.1002/1000/6875](http://handle.itu.int/11.1002/1000/6875) | | 2.7 | ITU-T G.722.2 Annex F | 2002-11-29 | 16 | [11.1002/1000/6180](http://handle.itu.int/11.1002/1000/6180) | | 2.8 | ITU-T G.722.2 App. I | 2002-01-13 | 16 | [11.1002/1000/6096](http://handle.itu.int/11.1002/1000/6096) | | 2.9 | ITU-T G.722.2 App. I (2002) Amd. 1 | 2003-07-29 | 16 | [11.1002/1000/6877](http://handle.itu.int/11.1002/1000/6877) | | 2.9 | ITU-T G.722.2 (2003) Cor. 1 | 2005-09-13 | 16 | [11.1002/1000/8575](http://handle.itu.int/11.1002/1000/8575) | | 2.10 | ITU-T G.722.2 (2003) Cor. 2 | 2007-01-13 | 16 | [11.1002/1000/9019](http://handle.itu.int/11.1002/1000/9019) | | 2.11 | ITU-T G.722.2 Annex C | 2008-11-13 | 16 | [11.1002/1000/9635](http://handle.itu.int/11.1002/1000/9635) | | 2.12 | ITU-T G.722.2 Annex C | 2017-12-14 | 16 | [11.1002/1000/13429](http://handle.itu.int/11.1002/1000/13429) | | 2.13 | ITU-T G.722.2 Annex D | 2017-12-14 | 16 | [11.1002/1000/13430](http://handle.itu.int/11.1002/1000/13430) | |

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of tele­com­mu­ni­ca­tions, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU‑T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at <http://www.itu.int/ITU-T/ipr/>.

© ITU 2018

All rights reserved. No part of this publication may be reproduced, by any means whatsoever, without the prior written permission of ITU.

**Table of Contents**

Page

Annex C – Fixed-point C-code 1

C.1 C-code structure 1

C.2 Homing procedure 11

C.3 File formats 12

Electronic attachment: AMR-WB codec fixed-point C-code

Recommendation ITU-T G.722.2

Wideband coding of speech at around 16 kbit/s using Adaptive   
Multi-Rate Wideband (AMR-WB)

Annex C  
  
Fixed-point C-code

(This annex forms an integral part of this Recommendation.)

## C.1 C-code structure

This annex[[2]](#footnote-2) gives an overview of the structure of the bit-exact C-code for the correct implementation of the ITU-T G.722.2 main body, its Annex A (comfort noise aspects), Annex B (source controlled rate operation) and Appendix I (error concealment of erroneous or lost frames). It provides an overview of the contents and organization of the C-code attached to this annex. In case of discrepancy between the description given in the several parts of ITU-T G.722.2 (including its Annexes A, B and Appendix I) and the ANSI C-source code, the algorithm description of the ANSI C-code shall prevail.

The C-code has been verified on a number of systems

– Sun Microsystems workstations and GNU gcc compiler

– HP workstations and cc compiler

– IBM PC compatible computers with Windows NT4 operating system and GNU gcc compiler.

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

### C.1.1 Contents of the C source code

The C-code distribution has all files in the root level.

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 mostly in files with suffix "tab".

The C code distribution also contains one speech coder installation verification data file, "spch\_dos.inp". The reference encoder output file is named "spch\_dos.cod", the reference decoder input file is named "spch\_dos.dec" and the reference decoder output file is named "spch\_dos.out". These four files are formatted such that they are correct for an IBM PC/AT compatible computer. The same files with reversed byte order of the 16 bit words are named "spch\_unx.inp", "spch\_unx.cod", "spch\_unx.dec" and "spch\_unx.out", respectively.

Final verification of bit-exactness is to be performed using the adaptive multi-rate wideband test sequences described in Annex D of ITU-T G.722.2.

Makefiles are provided for the platforms in which the C-code has been verified (see above). Once the software is installed, this directory will have a compiled version of *encoder* and *decoder* (the bit‑exact C executables of the speech codec) and all the object files.

### C.1.2 Program execution

The adaptive multi-rate wideband codec is implemented in two programs:

– (encoder) speech encoder;

–(*decoder*) speech decoder.

The programs should be named as follows:

– encoder [encoder options] <speech input file> <parameter file>;

– decoder <parameter file> <speech output file>.

The speech files contain 16-bit linear encoded PCM speech samples and the parameter files contain encoded speech data and some additional flags.

The encoder and decoder options will be explained by running the applications without input arguments. See the readme.txt file for more information on how to run the *encoder* and *decoder* programs.

### C.1.3 Code hierarchy

Tables C.1 to C.3 are call graphs that show the functions used in the speech codec, including the functions of VAD, DTX and comfort noise generation.

Each column represents a call level and each cell a function. The functions contain calls to the functions in rightwards neighbouring cells. The time order in the call graphs is from the top downwards as the processing of a frame advances. All standard C functions: printf(), fwrite(), etc., have been omitted. Also, no basic operations (add(), L\_add(), mac(), etc.) or double precision extended operations (e.g., L\_Extract()) appear in the graphs. The initialization of the static RAM (i.e., calling the \_init functions) is also omitted.

The basic operations are not counted as extending the depth; therefore, the deepest level in this software is level 6.

The encoder call graph is broken down into two separate call graphs, Tables C.1 and C.2.

Table C.1 – Speech encoder call structure

| coder | Copy |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Decim\_12k8 | Down\_samp | Interpol (function) |  |  |
|  |  | Copy |  |  |  |
|  | Set\_zero |  |  |  |  |
|  | HP50\_12k8 |  |  |  |  |
|  | Scale\_sig |  |  |  |  |
|  | wb\_vad | Filter\_bank | Filter5 |  |  |
|  |  |  | Filter3 |  |  |
|  |  |  | Level\_calculation |  |  |
|  |  | vad\_decision | Ilog2 |  |  |
|  |  |  | Noise\_estimate\_update | update\_cntrl |  |
|  |  |  | hangover\_addition |  |  |
|  |  | Estimate\_Speech |  |  |  |
|  | tx\_dtx\_handler |  |  |  |  |
|  | Parm\_serial |  |  |  |  |
|  | Autocorr |  |  |  |  |
|  | Lag\_window |  |  |  |  |
|  | Levinson |  |  |  |  |
|  | Az\_isp | Chebps2 |  |  |  |
|  | Int\_isp | Isp\_Az | Get\_isp\_pol |  |  |
|  | Isp\_isf |  |  |  |  |
|  | Gp\_clip\_test\_isf |  |  |  |  |
|  | Weight\_a |  |  |  |  |
|  | Residu |  |  |  |  |
|  | Deemph2 |  |  |  |  |
|  | LP\_Decim2 |  |  |  |  |
|  | Scale\_mem\_Hp\_wsp |  |  |  |  |
|  | Pitch\_med\_ol | Hp\_wsp |  |  |  |
|  |  | Isqrt\_n |  |  |  |
|  | wb\_vad\_tone\_detection |  |  |  |  |
|  | Med\_olag | median5 |  |  |  |
|  | dtx\_buffer | Copy |  |  |  |
|  | dtx\_enc | Find\_frame\_indices |  |  |  |
|  |  | Aver\_isf\_history |  |  |  |
|  |  | Qisf\_ns | Sub\_VQ |  |  |
|  |  |  | Disf\_ns | Reorder\_isf |  |
|  |  | Parm\_serial |  |  |  |
|  |  | Pow2 |  |  |  |
|  |  | Random |  |  |  |
|  |  | Dot\_product12 |  |  |  |
|  |  | Isqrt\_n |  |  |  |
|  | Isf\_isp |  |  |  |  |
|  | Isp\_Az | Get\_isp\_pol |  |  |  |
|  | Synthesis | Copy |  |  |  |
|  |  | Syn\_filt\_32 |  |  |  |
|  |  | Deemph\_32 |  |  |  |
|  |  | HP50\_12k8 |  |  |  |
|  |  | Random |  |  |  |
|  |  | Scale\_sig |  |  |  |
|  |  | Dot\_product12 |  |  |  |
|  |  | Isqrt\_n |  |  |  |
|  |  | HP400\_12k8 |  |  |  |
|  |  | Weight\_a |  |  |  |
|  |  | Syn\_filt |  |  |  |
|  |  | Filt\_6k\_7k |  |  |  |
|  | Reset\_encoder | Set\_zero |  |  |  |
|  |  | Init\_gp\_clip |  |  |  |
|  |  | Init\_Phase\_dispersion | Set\_zero |  |  |
|  | Qpisf\_2s\_36b | VQ\_stage1 |  |  |  |
|  |  | Sub\_VQ |  |  |  |
|  |  | Dpisf\_2s\_36b | Reorder\_isf |  |  |
|  | Qpisf\_2s\_46b | VQ\_stage1 |  |  |  |
|  |  | Sub\_VQ |  |  |  |
|  |  | Dpisf\_2s\_46b | Reorder\_isf |  |  |
|  | Syn\_filt |  |  |  |  |
|  | Preemph2 |  |  |  |  |
|  | Pitch\_fr4 | Norm\_Corr | Convolve |  |  |
|  |  |  | Isqrt\_n |  |  |
|  |  | Interpol\_4 |  |  |  |
|  | Gp\_clip |  |  |  |  |
|  | Pred\_lt4 |  |  |  |  |
|  | Convolve |  |  |  |  |
|  | G\_pitch | Dot\_product12 |  |  |  |
|  | Updt\_tar |  |  |  |  |
|  | Preemph |  |  |  |  |
|  | Pit\_shrp |  |  |  |  |
|  | Cor\_h\_x |  |  |  |  |
|  | ACELP\_2t64\_fx | Dot\_product12 |  |  |  |
|  |  | Isqrt\_n |  |  |  |
|  | ACELP\_4t64\_fx | See Table 2 |  |  |  |
|  | Q\_gain2 | Dot\_product12 |  |  |  |
|  |  | Pow2 |  |  |  |
|  | Gp\_clip\_test\_gain\_pit |  |  |  |  |
|  | voice\_factor | Dot\_product12 |  |  |  |

Table C.2 – ACELP\_4t64\_fx call structure

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ACELP\_4t64\_fx | Dot\_product12 |  |  |  |  |
|  | Isqrt\_n |  |  |  |  |
|  | cor\_h\_vec |  |  |  |  |
|  | search\_ixiy |  |  |  |  |
|  | quant\_1p\_N1 |  |  |  |  |
|  | quant\_2p\_2N1 |  |  |  |  |
|  | quant\_3p\_3N1 | quant\_2p\_2N1 |  |  |  |
|  |  | quant\_1p\_N1 |  |  |  |
|  | quant\_4p\_4N | quant\_4p\_4N1 | Quant\_2p\_2N1 |  |  |
|  |  | quant\_1p\_N1 |  |  |  |
|  |  | quant\_3p\_3N1 | Quant\_2p\_2N1 |  |  |
|  |  |  | Quant\_1p\_N1 |  |  |
|  |  | quant\_2p\_2N1 |  |  |  |
|  | quant\_5p\_5N | quant\_3p\_3N1 | Quant\_2p\_2N1 |  |  |
|  |  |  | Quant\_1p\_N1 |  |  |
|  |  | quant\_2p\_2N1 |  |  |  |
|  | quant\_6p\_6N\_2 | quant\_5p\_5N | Quant\_3p\_3N1 | quant\_2p\_2N1 |  |
|  |  |  |  | Quant\_1p\_N1 |  |
|  |  |  | quant\_2p\_2N1 |  |  |
|  |  | quant\_1p\_N1 |  |  |  |
|  |  | quant\_4p\_4N | quant\_4p\_4N1 | quant\_2p\_2N1 |  |
|  |  |  | quant\_1p\_N1 |  |  |
|  |  |  | quant\_3p\_3N1 | quant\_2p\_2N1 |  |
|  |  |  |  | quant\_1p\_N1 |  |
|  |  |  | quant\_2p\_2N1 |  |  |
|  |  | quant\_2p\_2N1 |  |  |  |
|  |  | quant\_3p\_3N1 | quant\_2p\_2N1 |  |  |
|  |  |  | Quant\_1p\_N1 |  |  |

Table C.3 – Speech decoder call structure

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| decoder | Rx\_dtx\_handler |  |  |  |  |
|  | Dtx\_dec | Copy |  |  |  |
|  |  | Disf\_ns | Reorder\_isf |  |  |
|  |  | Serial\_parm |  |  |  |
|  |  | Pow2 |  |  |  |
|  |  | Random |  |  |  |
|  |  | Dot\_product12 |  |  |  |
|  |  | Isqrt\_n |  |  |  |
|  | Serial\_parm |  |  |  |  |
|  | Isf\_isp |  |  |  |  |
|  | Isp\_Az | Get\_isp\_pol |  |  |  |
|  | Copy |  |  |  |  |
|  | Synthesis | Copy |  |  |  |
|  |  | Syn\_filt\_32 |  |  |  |
|  |  | Deemph\_32 |  |  |  |
|  |  | HP50\_12k8 |  |  |  |
|  |  | Oversamp\_16k | Copy |  |  |
|  |  |  | Up\_samp | Interpol |  |
|  |  | Random |  |  |  |
|  |  | Scale\_sig |  |  |  |
|  |  | Dot\_product12 |  |  |  |
|  |  | Isqrt\_n |  |  |  |
|  |  | HP400\_12k8 |  |  |  |
|  |  | Isf\_Extrapolation | Isf\_isp |  |  |
|  |  | Isp\_Az | Get\_isp\_pol |  |  |
|  |  | Weight\_a |  |  |  |
|  |  | Syn\_filt |  |  |  |
|  |  | Filt\_6k\_7k | Copy |  |  |
|  |  | Filt\_7k | Copy |  |  |
|  | Reset\_decoder | Set\_zero |  |  |  |
|  |  | Init\_Phase\_dispersion | Set\_zero |  |  |
|  | Dpisf\_2s\_36b | Reorder\_isf |  |  |  |
|  | Dpisf\_2s\_46b | Reorder\_isf |  |  |  |
|  | Int\_isp | Isp\_Az | Get\_isp\_pol |  |  |
|  | Lagconc | insertion\_sort | Insert |  |  |
|  |  | Random |  |  |  |
|  | Pred\_lt4 |  |  |  |  |
|  | Random |  |  |  |  |
|  | DEC\_ACELP\_2t64\_fx |  |  |  |  |
|  | DEC\_ACELP\_4t64\_fx | dec\_1p\_N1 |  |  |  |
|  |  | add\_pulses |  |  |  |
|  |  | dec\_2p\_2N1 |  |  |  |
|  |  | dec\_3p\_3N1 | Dec\_2p\_2N1 |  |  |
|  |  |  | dec\_1p\_N1 |  |  |
|  |  | dec\_4p\_4N | dec\_4p\_4N1 | dec\_2p\_2N1 |  |
|  |  |  | dec\_1p\_N1 |  |  |
|  |  |  | Dec\_3p\_3N1 | Dec\_2p\_2N1 |  |
|  |  |  |  | Dec\_1p\_N1 |  |
|  |  |  | Dec\_2p\_2N1 |  |  |
|  |  | dec\_5p\_5N | dec\_3p\_3N1 | Dec\_2p\_2N1 |  |
|  |  |  |  | Dec\_1p\_N1 |  |
|  |  |  | Dec\_2p\_2N1 |  |  |
|  |  | dec\_6p\_6N\_2 | Dec\_5p\_5N | dec\_3p\_3N1 | Dec\_2p\_2N1 |
|  |  |  |  |  | Dec\_1p\_N1 |
|  |  |  |  | dec\_2p\_2N1 |  |
|  |  |  | dec\_1p\_N1 |  |  |
|  |  |  | dec\_4p\_4N | dec\_4p\_4N1 | dec\_2p\_2N1 |
|  |  |  |  | dec\_1p\_N1 |  |
|  |  |  |  | Dec\_3p\_3N1 | Dec\_2p\_2N1 |
|  |  |  |  |  | Dec\_1p\_N1 |
|  |  |  |  | Dec\_2p\_2N1 |  |
|  |  |  | dec\_2p\_2N1 |  |  |
|  |  |  | dec\_3p\_3N1 | Dec\_2p\_2N1 |  |
|  |  |  |  | Dec\_1p\_N1 |  |
|  | Preemph |  |  |  |  |
|  | Pit\_shrp |  |  |  |  |
|  | D\_gain2 | Dot\_product12 |  |  |  |
|  |  | Isqrt\_n |  |  |  |
|  |  | Median5 |  |  |  |
|  |  | Pow2 |  |  |  |
|  | Scale\_sig |  |  |  |  |
|  | voice\_factor | Dot\_product12 |  |  |  |
|  | Phase\_dispersion | Set\_zero |  |  |  |
|  | Agc2 | Isqrt | Isqrt\_n |  |  |
|  | Set\_zero |  |  |  |  |
|  | Dtx\_dec\_activity\_update | Copy |  |  |  |

### C.1.4 Variables, constants and tables

The data types of variables and tables used in the fixed point implementation are signed integers in 2's complement representation, defined by:

– **Word16** 16-bit variable;

– **Word32** 32-bit variable.

#### C.1.4.1 Description of constants used in the C-code

This clause contains a listing of all global constants defined in cnst.h. See Table C.4.

| Table C.4 – Global constants | | |
| --- | --- | --- |
| Constant | Value | Description |
| L\_TOTAL | 384 | Total size of speech buffer |
| L\_WINDOW | 384 | Window size in LP analysis |
| L\_NEXT | 64 | Look-ahead size |
| L\_FRAME | 256 | Frame size in 12.8 kHz |
| L\_FRAME16k | 320 | Frame size in 16 kHz |
| L\_SUBFR | 64 | Subframe size in 12.8 kHz |
| L\_SUBFR16k | 80 | Subframe size in 16 kHz |
| NB\_SUBFR | 4 | Number of subframes |
| M16k | 20 | Order of LP filter in high-band synthesis in 6.60 mode |
| M | 16 | Order of LP filter |
| L\_FILT16k | 15 | Delay of down-sampling filter in 16 kHz |
| L\_FILT | 12 | Delay of down-sampling filter in 12.8 kHz |
| GP\_CLIP | 15565 | Pitch gain clipping |
| PIT\_SHARP | 27853 | Pitch sharpening factor |
| PIT\_MIN | 34 | Minimum pitch lag (all modes) |
| PIT\_FR2 | 128 | Minimum pitch lag with resolution ½ |
| PIT\_FR1\_9b | 160 | Minimum pitch lag with resolution for 9-bit quantization |
| PIT\_FR1\_8b | 92 | Minimum pitch lag with resolution for 8-bit quantization |
| PIT\_MAX | 231 | Maximum pitch lag |
| L\_INTERPOL | (16+1) | Length of filter for interpolation |
| OPL\_DECIM | 2 | Decimation in open-loop pitch analysis |
| PREEMPH\_FAC | 22282 | Pre-emphasis factor |
| GAMMA1 | 30147 | Weighting factor (numerator) |
| TILT\_FAC | 22282 | Tilt factor (denominator) |
| Q\_MAX | 8 | Scaling max. for signal |
| RANDOM\_INITSEED | 21845 | Random init value |
| L\_MEANBUF | 3 | Size of ISF buffer |
| ONE\_PER\_MEANBUF | 10923 | Inverse of L\_MEANBUF |

#### C.1.4.2 Description of fixed tables used in the C-code

This clause contains a listing of all fixed tables sorted by source file name and table name. All table data are declared as **Word16**. See Table C.5.

| Table C.5 – Fixed tables | | | |
| --- | --- | --- | --- |
| File | Table name | Length | Description |
| C4t64fx.c | Tipos | 36 | Starting points of iterations |
| Cod\_main.c | HP\_gain | 16 | High band gain table for 23.85 kbit/s mode |
| Cod\_main.c | Interpol\_frac | 4 | LPC interpolation coefficients |
| Cod\_main.c | Isp\_init | 16 | Isp tables for initialization |
| Cod\_main.c | Isf\_init | 16 | Isf tables for initialization |
| D\_gain2.c | cdown\_unusable | 7 | Attenuation factors for codebook gain in lost frames |
| D\_gain2.c | cdown\_usable | 7 | Attenuation factors for codebook gain in bad frames |
| D\_gain2.c | pdown\_unusable | 7 | Attenuation factors for adaptive codebook gain in lost frames |
| D\_gain2.c | pdown\_usable | 7 | Attenuation factors for adaptive codebook gain in bad frames |
| D\_gain2.c | Pred | 4 | Algebraic code book gain MA predictor coefficients |
| Dec\_main.c | HP\_gain | 16 | High band gain table for 23.85 kbit/s mode |
| Dec\_main.c | Interpol\_frac | 4 | LPC interpolation coefficients |
| Dec\_main.c | Isp\_init | 16 | Isp tables for initialization |
| Dec\_main.c | Isf\_init | 16 | Isf tables for initialization |
| Decim54.c | fir\_down | 120 | Downsample FIR filter coefficients |
| Decim54.c | fir\_up | 120 | Upsample FIR filter coefficients |
| Dtx.c | en\_adjust | 9 | Energy scaling factor for each mode during comfort noise |
| Grid100.tab | grid | 101 | This table points to specific Chebyshev polynomials |
| Ham\_wind.tab | Window | 384 | LP analysis window |
| Hp400.c | A | 3 | HP filter coefficients (denominator) in higher band energy estimation |
| Hp400.c | B | 3 | HP filter coefficients (numerator) in higher band energy estimation |
| Hp50.c | A | 3 | HP filter coefficients (denominator) in pre-filtering |
| Hp50.c | B | 3 | HP filter coefficients (numerator) in pre-filtering |
| Hp6k.c | Fir\_6k\_7k | 31 | Bandpass FIR filter coefficients for higher band generation |
| Hp7k.c | Fir\_7k | 31 | Bandpass FIR filter coefficients for higher band in 23.85 kbit/s mode |
| Hp\_wsp.c | A | 3 | HP filter coefficients (denominator) in open-loop lag gain computation |
| Hp\_wsp.c | B | 3 | HP filter coefficients (numerator) in open-loop lag gain computation |
| Isp\_isf.tab | slope | 128 | Table to compute cos(x) in Isf\_isp() |
| Isp\_isf.tab | Table | 129 | Table to compute acos(x) in Isp\_isf() |
| Lag\_wind.tab | lag\_h | 16 | High part of the lag window table |
| Lag\_wind.tab | lag\_l | 16 | Low part of the lag window table |
| Lp\_dec2.c | h\_fir | 5 | HP FIR filter coefficients in open-loop lag search |
| Math\_op.c | table\_isqrt | 49 | Table used in inverse square root computation |
| Math\_op.c | table\_pow2 | 33 | Table used in power of two computation |
| P\_med\_ol.tab | Corrweight | 199 | Weighting of the correlation function in open loop LTP search |
| Ph\_disp.c | ph\_imp\_low | 64 | Phase dispersion impulse response |
| Ph\_disp.c | ph\_imp\_mid | 64 | Phase dispersion impulse response |
| Pitch\_fr4.c | inter4\_1 | 32 | Interpolation filter coefficients |
| Pred\_lt4.c | inter4\_2 | 128 | Interpolation filter coefficients |
| Q\_gain2.c | pred | 4 | Algebraic code book gain MA predictor coefficients |
| Q\_gain2.tab | t\_qua\_gain6b | 2\*64 | Gain quantization table for 6-bit gain quantization |
| Q\_gain2.tab | t\_qua\_gain7b | 2\*128 | Gain quantization table for 7-bit gain quantization |
| Qisf\_ns.tab | dico1\_isf\_noise | 2\*64 | 1st ISF quantizer for comfort noise |
| Qisf\_ns.tab | dico2\_isf\_noise | 3\*64 | 2nd ISF quantizer for comfort noise |
| Qisf\_ns.tab | dico3\_isf\_noise | 3\*64 | 3rd ISF quantizer for comfort noise |
| Qisf\_ns.tab | dico4\_isf\_noise | 4\*32 | 4th ISF quantizer for comfort noise |
| Qisf\_ns.tab | dico5\_isf\_noise | 4\*32 | 5th ISF quantizer for comfort noise |
| Qisf\_ns.tab | mean\_isf\_noise | 16 | ISF mean for comfort noise |
| Qpisf\_2s.tab | dico1\_isf | 9\*256 | 1st ISF quantizer of the 1st stage |
| Qpisf\_2s.tab | dico2\_isf | 7\*256 | 2nd ISF quantizer of the 1st stage |
| Qpisf\_2s.tab | dico21\_isf | 3\*64 | 1st ISF quantizer of the 2nd stage (not the 6.60 kbit/s mode) |
| Qpisf\_2s.tab | dico21\_isf\_36b | 5\*128 | 1st ISF quantizer of the 2nd stage (the 6.60 kbit/s mode) |
| Qpisf\_2s.tab | dico22\_isf | 3\*128 | 2nd ISF quantizer of the 2nd stage (not the 6.60 kbit/s mode) |
| Qpisf\_2s.tab | dico22\_isf\_36b | 4\*128 | 2nd ISF quantizer of the 2nd stage (the 6.60 kbit/s mode) |
| Qpisf\_2s.tab | dico23\_isf | 3\*128 | 3rd ISF quantizer of the 2nd stage (not the 6.60 kbit/s mode) |
| Qpisf\_2s.tab | dico23\_isf\_36b | 7\*64 | 3rd ISF quantizer of the 2nd stage (the 6.60 kbit/s mode) |
| Qpisf\_2s.tab | dico24\_isf | 3\*32 | 4th ISF quantizer of the 2nd stage (not the 6.60 kbit/s mode) |
| Qpisf\_2s.tab | dico25\_isf | 4\*32 | 5th ISF quantizer of the 2nd stage (not the 6.60 kbit/s mode) |
| Qpisf\_2s.tab | Mean\_isf | 16 | ISF mean |

#### C.1.4.3 Static variables used in the C-code

In this clause, two tables that specify the static variables for the speech encoder and decoder, respectively, are shown. All static variables are declared within a C **struct**.See Tables C.6 and C.7.

| Table C.6 – Speech encoder static variables | | | |
| --- | --- | --- | --- |
| Struct name | Variable | Type[length] | Description |
| Coder\_State | mem\_decim | Word16[30] | Decimation filter memory |
|  | mem\_sig\_in | Word16[6] | Pre-filter memory |
|  | mem\_preemph | Word16 | Pre-emphasis filter memory |
|  | old\_speech | Word16[128] | Speech buffer |
|  | old\_wsp | Word16[115] | Buffer holding spectral weighted speech |
|  | old\_exc | Word16[248] | Excitation vector |
|  | mem\_levinson | Word16[18] | Levinson memories |
|  | Ispold | Word16[16] | Old ISP vector |
|  | ispold\_q | Word16[16] | Old quantized ISP vector |
|  | past\_isfq | Word16[16] | Past quantized ISF prediction error |
|  | mem\_wsp | Word16 | Open-loop LTP deemphasis filter memory |
|  | mem\_decim2 | Word16[3] | Open-loop LTP decimation filter memory |
|  | mem\_w0 | Word16 | Weighting filter memory (applied to error signal) |
|  | mem\_syn | Word16[16] | Synthesis filter memory |
|  | tilt\_code | Word16 | Pre-emphasis filter memory |
|  | old\_wsp\_max | Word16 | Open-loop scaling factor |
|  | old\_wsp\_shift | Word16 | Maximum open loop scaling factor |
|  | Q\_old | Word16 | Old scaling factor |
|  | Q\_max | Word16[2] | Maximum scaling factor |
|  | gp\_clip | Word16[2] | Memory of pitch clipping |
|  | qua\_gain | Word16[4] | Gain quantization memory |
|  | old\_T0\_med | Word16 | Weighted open-loop pitch lag |
|  | ol\_gain | Word16 | Open-loop gain |
|  | ada\_w | Word16 | Weighting level depending on open-loop pitch gain |
|  | ol\_wght\_flg | Word16 | Switches lag weighting on and off |
|  | old\_ol\_lag | Word16[5] | Open-loop lag history |
|  | hp\_wsp\_mem | Word16[9] | Open-loop lag gain filter memory |
|  | old\_hp\_wsp | Word16[243] | Open-loop lag |
|  | vadSt | VadVars\* | See below in this table |
|  | dtx\_encSt | dtx\_encState\* | See below in this table |
|  | first\_frame | Word16 | First frame indicator |
|  | Isfold | Word16[16] | Old ISF vector |
|  | L\_gc\_thres | Word16 | Noise enhancer threshold |
|  | mem\_syn\_hi | Word16[16] | Synthesis filter memory (most significant word) |
|  | mem\_syn\_lo | Word16[16] | Synthesis filter memory (least significant word) |
|  | mem\_deemph | Word16 | De-emphasis filter memory |
|  | mem\_sig\_out | Word16[6] | HP filter memory in the synthesis |
|  | mem\_hp400 | Word16[6] | HP filter memory |
|  | mem\_oversamp | Word16[2\*12] | Oversampling filter memory |
|  | mem\_syn\_hf | Word16[16] | Higher band synthesis filter memory |
|  | mem\_hf | Word16[30] | Estimated BP filter memory (23.85 kbit/s mode) |
|  | mem\_hf2 | Word16[30] | Input BP filter memory (23.85 kbit/s mode) |
|  | mem\_hf3 | Word16[30] | Input LP filter memory (23.85 kbit/s mode) |
|  | seed2 | Word16 | Random generation seed |
|  | disp\_mem | Word16[8] | Phase dispersion memory |
|  | vad\_hist | Word16 | VAD history |
|  | Gain\_alpha | Word16 | Higher band gain weighting factor (23.85 kbit/s mode) |
| dtx\_encState | Isf\_hist | Word16[128] | ISP history (8 frames) |
|  | Log\_en\_hist | Word16[8] | Logarithmic frame energy history (8 frames) |
|  | Hist\_ptr | Word16 | Pointer to the cyclic history vectors |
|  | Log\_en\_index | Word16 | Index for logarithmic energy |
|  | Cng\_seed | Word16 | Comfort noise excitation seed |
|  | D | Word16[28] | ISF history distance matrix |
|  | sumD | Word16[8] | Sum of ISF history distances |
|  | dtxHangoverCount | Word16 | Is decreased in DTX hangover period |
|  | decAnaElapsedCount | Word16 | Counter for elapsed speech frames in DTX |
| vadState1 | bckr\_est | Word16[12] | Background noise estimate |
|  | ave\_level | Word16[12] | Averaged input components for stationary estimation |
|  | old\_level | Word16[12] | Input levels of the previous frame |
|  | sub\_level | Word16[12] | Input levels calculated at the end of a frame (lookahead) |
|  | a\_data5 | Word16[5][2] | Memory for the filter bank |
|  | a\_data3 | Word16[6] | Memory for the filter bank |
|  | burst\_count | Word16 | Counts length of a speech burst |
|  | Hang\_count | Word16 | Hangover counter |
|  | Stat\_count | Word16 | Stationary counter |
|  | Vadreg | Word16 | 15 flags for intermediate VAD decisions |
|  | Tone\_flag | Word16 | 15 flags for tone detection |
|  | sp\_est\_cnt | Word16 | Speech level estimation counter |
|  | Sp\_max | Word16 | Maximum signal level |
|  | sp\_max\_cnt | Word16 | Maximum level estimation counter |
|  | Speech\_level | Word16 | Speech level |
|  | prev\_pow\_sum | Word16 | Power of previous frame |

| Table C.7 – Speech decoder static variables | | | |
| --- | --- | --- | --- |
| Struct name | Variable | Type[length] | Description |
| Decoder\_State | old\_exc | Word16[248] | Excitation vector |
|  | ispold | Word16[16] | Old ISP vector |
|  | isfold | Word16[16] | Old ISF vector |
|  | isf\_buf | Word16[48] | ISF vector history |
|  | past\_isfq | Word16[16] | Past quantized ISF prediction error |
|  | tilt\_code | Word16 | Pre-emphasis filter memory |
|  | Q\_old | Word16 | Old scaling factor |
|  | Qsubfr | Word16 | Scaling factor history |
|  | L\_gc\_thres | Word16 | Noise enhancer threshold |
|  | mem\_syn\_hi | Word16[16] | Synthesis filter memory (most significant word) |
|  | mem\_syn\_lo | Word16[16] | Synthesis filter memory (least significant word) |
|  | mem\_deemph | Word16 | De-emphasis filter memory |
|  | mem\_sig\_out | Word16[6] | HP filter memory in the synthesis |
|  | mem\_oversamp | Word16[24] | Oversampling filter memory |
|  | mem\_syn\_hf | Word16[20] | Higher band synthesis filter memory |
|  | mem\_hf | Word16[30] | Estimated BP filter memory (23.85 kbit/s mode) |
|  | mem\_hf2 | Word16[30] | Input BP filter memory (23.85 kbit/s mode) |
|  | mem\_hf3 | Word16[30] | Input LP filter memory (23.85 kbit/s mode) |
|  | seed | Word16 | Random code generation seed for bad frames |
|  | seed2 | Word16 | Random generation seed for higher band |
|  | old\_T0 | Word16 | Old LTP lag (integer part) |
|  | old\_T0\_frac | Word16 | Old LTP lag (fraction part) |
|  | lag\_hist | Word16[5] | LTP lag history |
|  | dec\_gain | Word16[23] | Gain decoding memory |
|  | seed3 | Word16 | Random LTP lag generation seed for bad frames |
|  | disp\_mem | Word16[8] | Phase dispersion memory |
|  | mem\_hp400 | Word16[6] | HP filter memory |
|  | prev\_bfi | Word16 | Previous BFI |
|  | state | Word16 | BGH state machine memory |
|  | first\_frame | Word16 | First frame indicator |
|  | dtx\_decSt | dtx\_decState\* | See below in this table |
|  | Vad\_hist | Word16 | VAD history |
| dtx\_decState | Since\_last\_sid | Word16 | Number of frames since last SID frame |
|  | true\_sid\_period\_inv | Word16 | Inverse of true SID update rate |
|  | log\_en | Word16 | Logarithmic frame energy |
|  | old\_log\_en | Word16 | Previous value of log\_en |
|  | isf | Word16[16] | ISF vector |
|  | Isf\_old | Word16[16] | Previous ISF vector |
|  | Cng\_seed | Word16 | Comfort noise excitation seed |
|  | Isf\_hist | Word16[128] | ISF vector history (8 frames) |
|  | Log\_en\_hist | Word16[8] | Logarithmic frame energy history |
|  | Hist\_ptr | Word16 | Index to beginning of ISF history |
|  | dtxHangoverCount | Word16 | Counts down in hangover period |
|  | DecAnaElapsedCount | Word16 | Counts elapsed speech frames after DTX |
|  | sid\_frame | Word16 | Flags SID frames |
|  | valid\_data | Word16 | Flags SID frames containing valid data |
|  | log\_en\_adjust | Word16 | Mode-dependent frame energy adjustment |
|  | dtxHangoverAdded | Word16 | Flags hangover period at end of speech |
|  | dtxGlobalState | Word16 | DTX state flags |
|  | data\_updated | Word16 | Flags CNI updates |

## C.2 Homing procedure

The principles of the homing procedures are described in the main body of this Recommendation. This clause only includes a detailed description of the nine decoder homing frames. For each AMR‑WB codec mode, the corresponding decoder homing frame has a fixed set of parameters. The parameters in serial format are packed into parameters in a 15-bit-long format where the first serial bit is inserted into most significant bit in the 15-bit-long format. These 15-bit-long parameters do not represent real speech parameters, but they decrease memory consumption compared to the speech parameters. Table C.8 shows the homing frame in a 15-bit-long format for different modes. In the decoder, the received speech parameters in serial format are first converted into a 15-bit-long format. Then the obtained parameters are compared against the homing frame table values. See Table C.8.

Table C.8 – Table values for the decoder homing frame  
in 15-bit-long format for different modes

|  |  |
| --- | --- |
| Mode | Value (MSB=b0) |
| 0 | 3168, 29954, 29213, 16121, 64, 13440, 30624, 16430, 19008 |
| 1 | 3168, 31665, 9943, 9123, 15599, 4358, 20248, 2048, 17040, 27787, 16816, 13888 |
| 2 | 3168, 31665, 9943, 9128, 3647, 8129, 30930, 27926, 18880, 12319, 496, 1042, 4061, 20446, 25629, 28069, 13948 |
| 3 | 3168, 31665, 9943, 9131, 24815, 655, 26616, 26764, 7238, 19136, 6144, 88, 4158, 25733, 30567, 30494, 221, 20321, 17823 |
| 4 | 3168, 31665, 9943, 9131, 24815, 700, 3824, 7271, 26400, 9528, 6594, 26112, 108, 2068, 12867, 16317, 23035, 24632, 7528, 1752, 6759, 24576 |
| 5 | 3168, 31665, 9943, 9135, 14787, 14423, 30477, 24927, 25345, 30154, 916, 5728, 18978, 2048, 528, 16449, 2436, 3581, 23527, 29479, 8237, 16810, 27091, 19052, 0 |
| 6 | 3168, 31665, 9943, 9129, 8637, 31807, 24646, 736, 28643, 2977, 2566, 25564, 12930, 13960, 2048, 834, 3270, 4100, 26920, 16237, 31227, 17667, 15059, 20589, 30249, 29123, 0 |
| 7 | 3168, 31665, 9943, 9132, 16748, 3202, 28179, 16317, 30590, 15857, 19960, 8818, 21711, 21538, 4260, 16690, 20224, 3666, 4194, 9497, 16320, 15388, 5755, 31551, 14080, 3574, 15932, 50, 23392, 26053, 31216 |
| 8 | 3168, 31665, 9943, 9134, 24776, 5857, 18475, 28535, 29662, 14321, 16725, 4396, 29353, 10003, 17068, 20504, 720, 0, 8465, 12581, 28863, 24774, 9709, 26043, 7941, 27649, 13965, 15236, 18026, 22047, 16681, 3968 |

## C.3 File formats

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

### C.3.1 Speech file (encoder input/decoder output)

Speech files read by the encoder and written by the decoder consist of 16-bit words where each word contains a 14-bit, left aligned speech sample. The byte order depends on the host architecture (e.g., MSByte first on SUN workstations, LSByte first on PCs, etc.). Both the encoder and the decoder program process complete frames (of 320 samples) only.

This means that the encoder will only process *n* frames if the length of the input file is *n\**320*+ k* words, while the files produced by the decoder will always have a length of *n\**320 words.

### C.3.2 Mode control file (encoder input)

The encoder program can optionally read in a mode control file which specifies the encoding mode for each frame of speech processed. The file is a text file containing one number per speech frame. Each line contains one of the mode numbers 0-8.

### C.3.3 Parameter bitstream file (encoder output/decoder input)

The files produced by the speech encoder/expected by the speech decoder contain an arbitrary number of frames in the following available formats.

NOTE ON DEFAULT 3GPP AND ITU BITSTREAM FORMATS – ITU stream format gives very limited possibilities to distinguish NO\_DATA and SID\_FIRST frame types at the beginning of a stream. In some very limited cases for which some instance between encoder and decoder cuts of the first hangover period frames (e.g., handovers, editing of the stream), the output of the decoder is different depending on the stream format, ITU or default 3GPP.

i) Default 3GPP format

This is the default format used in 3GPP. This format shall be used when the codec is tested against the test vectors.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| TYPE\_OF\_FRAME\_TYPE | FRAME\_TYPE | MODE | B1 | B2 | … | Bnn |

Each box corresponds to one Word16 value in the bitstream file, for a total of 3+nn words or 6+2nn bytes per frame, where nn is the number of encoded bits in the frame. Each encoded bit is represented as follows: Bit 0 = 0xff81, Bit 1 = 0x007f. The fields have the following meaning:

TYPE\_OF\_FRAME\_TYPE transmit frame type, which is one of

TX\_TYPE (0x6b21)  
RX\_TYPE (0x6b20)

If TYPE\_OF\_FRAME\_TYPE is TX\_TYPE,

FRAME\_TYPE transmit frame type, which is one of

TX\_SPEECH (0x0000)  
TX\_SID\_FIRST (0x0001)  
TX\_SID\_UPDATE (0x0002)  
TX\_NO\_DATA (0x0003)

If TYPE\_OF\_FRAME\_TYPE is RX\_TYPE,

FRAME\_TYPE transmit frame type, which is one of

RX\_SPEECH\_GOOD (0x0000)  
RX\_SPEECH\_PROBABLY\_DEGRADED (0x0001)  
RX\_SPEECH\_LOST (0x0002)  
RX\_SPEECH\_BAD (0x0003)  
RX\_SID\_FIRST (0x0004)  
RX\_SID\_UPDATE (0x0005)  
RX\_SID\_BAD (0x0006)  
RX\_NO\_DATA (0x0007)

B0…B2nn speech encoder parameter bits (i.e., the bitstream itself). Each B*x* either has the value 0x0081 (for bit 0) or 0x007F (for bit 1).

MODE\_INFO encoding mode information, which is one of

6.60 kbit/s mode (0x0000)  
8.85 kbit/s mode (0x0001)  
12.65 kbit/s mode (0x0002)  
14.25 kbit/s mode (0x0003)  
15.85 kbit/s mode (0x0004)  
18.25 kbit/s mode (0x0005)  
19.85 kbit/s mode (0x0006)  
23.05 kbit/s mode (0x0007)  
23.85 kbit/s mode (0x0008)

As indicated in clause C.3.1 above, the byte order depends on the host architecture.

ii) ITU format (activated with command line parameter -itu)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| SYNC\_WORD | DATA\_LENGTH | B1 | B2 | … | Bnn |

Each box corresponds to one Word16 value in the bitstream file, for a total of 2+nn words or 4+2nn bytes per frame, where nn is the number of encoded bits in the frame. Each encoded bit is represented as follows: Bit 0 = 0x007f, Bit 1 = 0x0081. The fields have the following meaning:

SYNC\_WORD Word to ensure correct frame synchronization between the encoder and the decoder. It is also used to indicate the occurrences of bad frames.

In the encoder output: (0x6b21)  
In the decoder input: Good frames (0x6b21)  
 Bad frames (0x6b20)

DATA\_LENGTH Length of the speech data. Codec mode and frame type is extracted in the decoder using this parameter:

| DATA \_LENGTH | PREVIOUS FRAME | CODEC MODE | FRAMETYPE |
| --- | --- | --- | --- |
| 0 | RX\_SPEECH\_GOOD/ RX\_SPEECH\_LOST | DTX | RX\_SID\_FIRST |
| 0 | OTHER THAN RX\_SPEECH\_GOOD/ RX\_SPEECH\_LOST | DTX | RX\_NO\_DATA |
| 35 | – | DTX | RX\_SID\_UPDATE |
| 132 | – | 6.60 kbit/s | RX\_SPEECH\_GOOD/ RX\_SPEECH\_LOST |
| 177 | – | 8.85 kbit/s | RX\_SPEECH\_GOOD/ RX\_SPEECH\_LOST |
| 253 | – | 12.65 kbit/s | RX\_SPEECH\_GOOD/ RX\_SPEECH\_LOST |
| 285 | – | 14.25 kbit/s | RX\_SPEECH\_GOOD/ RX\_SPEECH\_LOST |
| 317 | – | 15.85 kbit/s | RX\_SPEECH\_GOOD/ RX\_SPEECH\_LOST |
| 365 | – | 18.25 kbit/s | RX\_SPEECH\_GOOD/ RX\_SPEECH\_LOST |
| 397 | – | 19.85 kbit/s | RX\_SPEECH\_GOOD/ RX\_SPEECH\_LOST |
| 461 | – | 23.05 kbit/s | RX\_SPEECH\_GOOD/ RX\_SPEECH\_LOST |
| 477 | – | 23.85 kbit/s | RX\_SPEECH\_GOOD/ RX\_SPEECH\_LOST |

iii) MIME/file storage format (activated with command line parameter -mime)

Detailed description of the AMR-WB single channel MIME/file storage format can be found in clauses 5.1 and 5.3 of [IETF RFC 3267]. This format is used, e.g., by the Multimedia Messaging Service (MMS).

|  |  |
| --- | --- |
| **SERIES OF ITU-T RECOMMENDATIONS** | |
| Series A | Organization of the work of ITU-T |
| Series D | Tariff and accounting principles and international telecommunication/ICT economic and policy issues |
| Series E | Overall network operation, telephone service, service operation and human factors |
| Series F | Non-telephone telecommunication services |
| **Series G** | **Transmission systems and media, digital systems and networks** |
| Series H | Audiovisual and multimedia systems |
| Series I | Integrated services digital network |
| Series J | Cable networks and transmission of television, sound programme and other multimedia signals |
| Series K | Protection against interference |
| Series L | Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant |
| Series M | Telecommunication management, including TMN and network maintenance |
| Series N | Maintenance: international sound programme and television transmission circuits |
| Series O | Specifications of measuring equipment |
| Series P | Telephone transmission quality, telephone installations, local line networks |
| Series Q | Switching and signalling, and associated measurements and tests |
| Series R | Telegraph transmission |
| Series S | Telegraph services terminal equipment |
| Series T | Terminals for telematic services |
| Series U | Telegraph switching |
| Series V | Data communication over the telephone network |
| Series X | Data networks, open system communications and security |
| Series Y | Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities |
| Series Z | Languages and general software aspects for telecommunication systems |
|  |  |

1. \* To access the Recommendation, type the URL http://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID. For example, <http://handle.itu.int/11.1002/1000/11830-en>. [↑](#footnote-ref-1)
2. This annex contains an electronic attachment with the AMR-WB codec fixed-point C-code. [↑](#footnote-ref-2)