International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

G.722.2

Annex C
Corrigendum 1

(08/2018)

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

Corrigendum 1: Corrections to Table C.5

Recommendation ITU-T G.722.2 Annex C (2017) – Corrigendum 1

International Telecommunication Union

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

Corrigendum 1

Corrections to Table C.5

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).

Corrigendum 1 details a number of minor corrections to the capitalizations of certain file and table names in the description in Table C.5 used in the C-code in Annex C of ITU-T G.722.2. These changes harmonize Annex C with the AMR-WB codec specification, TS 26.173 version 14.1.0 (Rel-14).

The changes are only relevant to the text of Annex C and not to the source ANSI-C code of its electronic attachment.

History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T G.722.2	2002-01-13	16	11.1002/1000/5650
1.1	ITU-T G.722.2 Annex C	2002-01-13	16	11.1002/1000/5662
1.3	ITU-T G.722.2 Annex C	2003-07-29	16	11.1002/1000/6865
1.4	ITU-T G.722.2 Annex D	2002-01-13	16	11.1002/1000/5663
1.4	ITU-T G.722.2 Annex C	2004-03-15	16	11.1002/1000/7206
2.0	ITU-T G.722.2	2003-07-29	16	11.1002/1000/6506
2.1	ITU-T G.722.2 Annex A	2002-01-13	16	11.1002/1000/5660
2.2	ITU-T G.722.2 Annex B	2002-01-13	16	11.1002/1000/5661
2.5	ITU-T G.722.2 Annex E	2002-01-13	16	11.1002/1000/5664
2.6	ITU-T G.722.2 Annex D	2003-07-29	16	11.1002/1000/6871
2.6	ITU-T G.722.2 Annex E (2002) Cor. 1	2003-07-29	16	11.1002/1000/6875
2.7	ITU-T G.722.2 Annex F	2002-11-29	16	11.1002/1000/6180
2.8	ITU-T G.722.2 App. I	2002-01-13	16	11.1002/1000/6096
2.9	ITU-T G.722.2 App. I (2002) Amd. 1	2003-07-29	16	11.1002/1000/6877
2.9	ITU-T G.722.2 (2003) Cor. 1	2005-09-13	16	11.1002/1000/8575
2.10	ITU-T G.722.2 (2003) Cor. 2	2007-01-13	16	11.1002/1000/9019
2.11	ITU-T G.722.2 Annex C	2008-11-13	16	11.1002/1000/9635
2.12	ITU-T G.722.2 Annex C	2017-12-14	16	11.1002/1000/13429
2.13	ITU-T G.722.2 Annex D	2017-12-14	16	11.1002/1000/13430
2.14	ITU-T G.722.2 Annex C (2017) Cor. 1	2018-08-29	16	11.1002/1000/13663

* 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.

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/.

Table of Contents

Electronic attachment: AMR-WB codec fixed-point C-code (available with the base text for Annex C (2017))

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

Corrigendum 1

Corrections to Table C.5

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

Editorial note: This is a complete-text publication. Modifications introduced by this corrigendum are shown in revision marks relative to Recommendation ITU-T G.722.2 Annex C (2017).

C.1 C-code structure

This annex[1] 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.

[1] This annex contains an electronic attachment with the AMR-WB codec fixed-point C-code. The electronic attachment is distributed with the base text of Annex C (2017).

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)

Table C.4 – Global constants

Constant	Value	Description
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
~~C~~c4t64fx.c	~~T~~tipos	36	Starting points of iterations
~~C~~cod_main.c	HP_gain	16	High band gain table for 23.85 kbit/s mode
~~C~~cod_main.c	~~I~~interpol_frac	4	LPC interpolation coefficients
~~C~~cod_main.c	~~I~~isp_init	16	Isp tables for initialization
~~C~~cod_main.c	~~I~~isf_init	16	Isf tables for initialization
~~D~~d_gain2.c	cdown_unusable	7	Attenuation factors for codebook gain in lost frames
~~D~~d_gain2.c	cdown_usable	7	Attenuation factors for codebook gain in bad frames
~~D~~d_gain2.c	pdown_unusable	7	Attenuation factors for adaptive codebook gain in lost frames
~~D~~d_gain2.c	pdown_usable	7	Attenuation factors for adaptive codebook gain in bad frames
~~D~~d_gain2.c	~~P~~pred	4	Algebraic code book gain MA predictor coefficients
~~D~~dec_main.c	HP_gain	16	High band gain table for 23.85 kbit/s mode
~~D~~dec_main.c	~~I~~interpol_frac	4	LPC interpolation coefficients
~~D~~dec_main.c	~~I~~isp_init	16	Isp tables for initialization
~~D~~dec_main.c	~~I~~isf_init	16	Isf tables for initialization
~~D~~decim54.c	fir_down	120	Downsample FIR filter coefficients
~~D~~decim54.c	fir_up	120	Upsample FIR filter coefficients
~~D~~dtx.c	en_adjust	9	Energy scaling factor for each mode during comfort noise
~~G~~grid100.tab	grid	101	~~This table~~Grid points of ~~to specific~~ Chebyshev polynomials
~~H~~ham_wind.tab	~~W~~window	384	LP analysis window
~~H~~hp400.c	~~A~~a	3	HP filter coefficients (denominator) in higher band energy estimation
~~H~~hp400.c	~~B~~b	3	HP filter coefficients (numerator) in higher band energy estimation

Table C.5 – Fixed tables

File	Table name	Length	Description
~~H~~hp50.c	~~A~~a	3	HP filter coefficients (denominator) in pre-filtering
~~H~~hp50.c	~~B~~b	3	HP filter coefficients (numerator) in pre-filtering
~~H~~hp6k.c	~~F~~fir_6k_7k	31	Bandpass FIR filter coefficients for higher band generation
~~H~~hp7k.c	~~F~~fir_7k	31	Bandpass FIR filter coefficients for higher band in 23.85 kbit/s mode
~~H~~hp_wsp.c	~~A~~a	3	HP filter coefficients (denominator) in open-loop lag gain computation
~~H~~hp_wsp.c	~~B~~b	3	HP filter coefficients (numerator) in open-loop lag gain computation
~~I~~isp_isf.tab	slope	128	Table to compute cos(x) in Isf_isp()
~~I~~isp_isf.tab	~~T~~table	129	Table to compute acos(x) in Isp_isf()
~~L~~lag_wind.tab	lag_h	16	High part of the lag window table
~~L~~lag_wind.tab	lag_l	16	Low part of the lag window table
~~L~~lp_dec2.c	h_fir	5	HP FIR filter coefficients in open-loop lag search
~~M~~math_op.c	table_isqrt	49	Table used in inverse square root computation
~~M~~math_op.c	table_pow2	33	Table used in power of two computation
~~P~~p_med_ol.tab	~~C~~corrweight	199	Weighting of the correlation function in open loop LTP search
~~P~~ph_disp.c	ph_imp_low	64	Phase dispersion impulse response
~~P~~ph_disp.c	ph_imp_mid	64	Phase dispersion impulse response
~~P~~pitch_fr4.c	inter4_1	32	Interpolation filter coefficients
~~P~~pred_lt4.c	inter4_2	128	Interpolation filter coefficients
~~Q~~q_gain2.c	pred	4	Algebraic code book gain MA predictor coefficients
~~Q~~q_gain2.tab	t_qua_gain6b	2*64	Gain quantization table for 6-bit gain quantization
~~Q~~q_gain2.tab	t_qua_gain7b	2*128	Gain quantization table for 7-bit gain quantization
~~Q~~qisf_ns.tab	dico1_isf_noise	2*64	1st ISF quantizer for comfort noise
~~Q~~qisf_ns.tab	dico2_isf_noise	3*64	2nd ISF quantizer for comfort noise
~~Q~~qisf_ns.tab	dico3_isf_noise	3*64	3rd ISF quantizer for comfort noise
~~Q~~qisf_ns.tab	dico4_isf_noise	4*32	4th ISF quantizer for comfort noise
~~Q~~qisf_ns.tab	dico5_isf_noise	4*32	5th ISF quantizer for comfort noise
~~Q~~qisf_ns.tab	mean_isf_noise	16	ISF mean for comfort noise
~~Q~~qpisf_2s.tab	dico1_isf	9*256	1st ISF quantizer of the 1st stage
~~Q~~qpisf_2s.tab	dico2_isf	7*256	2nd ISF quantizer of the 1st stage
~~Q~~qpisf_2s.tab	dico21_isf	3*64	1st ISF quantizer of the 2nd stage (not the 6.60 kbit/s mode)
~~Q~~qpisf_2s.tab	dico21_isf_36b	5*128	1st ISF quantizer of the 2nd stage (the 6.60 kbit/s mode)
~~Q~~qpisf_2s.tab	dico22_isf	3*128	2nd ISF quantizer of the 2nd stage (not the 6.60 kbit/s mode)

Table C.5 – Fixed tables

File	Table name	Length	Description
~~Q~~qpisf_2s.tab	dico22_isf_36b	4*128	2nd ISF quantizer of the 2nd stage (the 6.60 kbit/s mode)
~~Q~~qpisf_2s.tab	dico23_isf	3*128	3rd ISF quantizer of the 2nd stage (not the 6.60 kbit/s mode)
~~Q~~qpisf_2s.tab	dico23_isf_36b	7*64	3rd ISF quantizer of the 2nd stage (the 6.60 kbit/s mode)
~~Q~~qpisf_2s.tab	dico24_isf	3*32	4th ISF quantizer of the 2nd stage (not the 6.60 kbit/s mode)
~~Q~~qpisf_2s.tab	dico25_isf	4*32	5th ISF quantizer of the 2nd stage (not the 6.60 kbit/s mode)
~~Q~~qpisf_2s.tab	~~M~~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

Table C.6 – Speech encoder static variables

Struct name	Variable	Type[length]	Description
	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

Table C.6 – Speech encoder static variables

Struct name	Variable	Type[length]	Description
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

Table C.7 – Speech decoder static variables

Struct name	Variable	Type[length]	Description
	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 Bx 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

Printed in Switzerland
Geneva, 2019