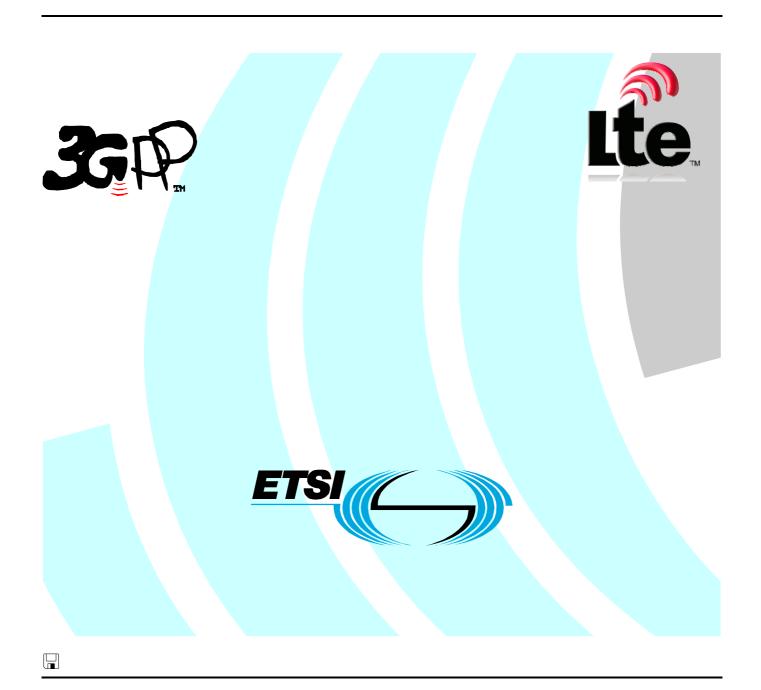
# ETSI TS 126 243 V9.0.0 (2010-01)

Technical Specification

Digital cellular telecommunications system (Phase 2+);
Universal Mobile Telecommunications System (UMTS);
LTE;
ANSI C code for the fixed-point distributed speech recognition
extended advanced front-end
(3GPP TS 26.243 version 9.0.0 Release 9)



# Reference RTS/TSGS-0426243v900 Keywords

GSM, LTE, UMTS

#### **ETSI**

650 Route des Lucioles F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C Association à but non lucratif enregistrée à la Sous-Préfecture de Grasse (06) N° 7803/88

## Important notice

Individual copies of the present document can be downloaded from: <u>http://www.etsi.org</u>

The present document may be made available in more than one electronic version or in print. In any case of existing or perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF). In case of dispute, the reference shall be the printing on ETSI printers of the PDF version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status.

Information on the current status of this and other ETSI documents is available at

<a href="http://portal.etsi.org/tb/status/status.asp">http://portal.etsi.org/tb/status/status.asp</a>

If you find errors in the present document, please send your comment to one of the following services: http://portal.etsi.org/chaircor/ETSI\_support.asp

## 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.

© European Telecommunications Standards Institute 2010. All rights reserved.

**DECT**<sup>TM</sup>, **PLUGTESTS**<sup>TM</sup>, **UMTS**<sup>TM</sup>, **TIPHON**<sup>TM</sup>, the TIPHON logo and the ETSI logo are Trade Marks of ETSI registered for the benefit of its Members.

**3GPP**<sup>™</sup> is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **LTE**<sup>™</sup> is a Trade Mark of ETSI currently being registered

for the benefit of its Members and of the 3GPP Organizational Partners.

 $\textbf{GSM} \\ \textbf{@} \text{ and the GSM logo are Trade Marks registered and owned by the GSM Association}.$ 

# Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: "Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards", which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://webapp.etsi.org/IPR/home.asp).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

## **Foreword**

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

The present document may refer to technical specifications or reports using their 3GPP identities, UMTS identities or GSM identities. These should be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between GSM, UMTS, 3GPP and ETSI identities can be found under <a href="http://webapp.etsi.org/key/queryform.asp">http://webapp.etsi.org/key/queryform.asp</a>.

# Contents

| Intell | lectual Property Rights                        | 2  |
|--------|--|----|
| Forev  | word   | 2  |
|        |  |    |
| rorev  | word   | 4  |
| 1      | Scope  | 5  |
| 2      | References                                     | 5  |
| 3      | Definitions and abbreviations                  | 5  |
| 3.1    | Definitions                                    |    |
| 3.2    | Abbreviations                                  | 5  |
| 4      | C code structure                               | 5  |
| 4.1    | Contents of the C source code                  | 5  |
| 4.2    | Program execution                              | 6  |
| 4.3    | Code hierarchy                                 |    |
| 4.5    | Variables, constants and tables                | 12 |
| 4.5.1  | Description of constants used in the C-code    | 13 |
| 4.5.2  | Description of fixed tables used in the C-code | 16 |
| 4.5.3  | Static variables used in the C-code            |    |
| 5      | File formats                                   | 21 |
| 5.1    | Speech file                                    |    |
| Anne   | ex A (informative): Change history             | 22 |
| Histo  | orv  |    |

# **Foreword**

This Technical Specification has been produced by the 3<sup>rd</sup> 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.

# 1 Scope

The present document contains an electronic copy of the ANSI-C code for DSR Extended Advanced Front-end. The ANSI-C code is necessary for a bit exact implementation of DSR Extended Advanced Front-end.

## 2 References

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

[1] ETSI ES 202 050: "Distributed Speech Recognition; Advanced Front-end Feature Extraction

Algorithm; Compression Algorithm", Oct 2002.

[2] ETSI ES 202 212 "Distributed Speech Recognition; Extended Advanced Front-end Feature

Extraction Algorithm; Compression Algorithm, Back-end Speech Reconstruction Algorithm",

Nov 2003.

[3] 3GPP TS 26.177: "Speech Enabled Services (SES); Distributed Speech Recognition (DSR)

extended advanced front-end test sequences".

# 3 Definitions and abbreviations

## 3.1 Definitions

Definition of terms used in the present document, can be found in [1], [2]

## 3.2 Abbreviations

For the purpose of the present document, the following abbreviations apply:

ANSI American National Standards Institute

I/O Input/Output

RAM Random Access Memory ROM Read Only Memory AFE Advanced Front-end

X-AFE eXtended Advanced Front-end DSR Distributed Speech Recognition

# 4 C code structure

This clause gives an overview of the structure of the bit-exact C code and provides an overview of the contents and organization of the C code attached to this document.

The C code has been verified on the following systems:

- Sun Microsystems workstations and GNU gcc compiler
- IBM PC compatible computers with Linux operating system and GNU gcc compiler.

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

## 4.1 Contents of the C source code

The distributed files with suffix "c" contain the source code and the files with suffix "h" are the header files.

Makefiles are provided for the platforms in which the C code has been verified (listed above).

# 4.2 Program execution

There are separate executables for the FrontEnd and Vector Quantization, with and without Extensions. The command line options are described below.

<> - indicates parameters for the given option for running the executable

() – indicates default parameter.

#### FrontEnd w/ Extension:

USAGE: bin/ExtAdvFrontEnd infile HTK\_outfile pitch\_outfile class\_outfile [options] OPTIONS:

-q Quiet Mode (FALSE)

-F format Input file format *<NIST,HTK,RAW>* (NIST)
-fs freq Sampling frequency in kHz *<8,16>* (8)
-swap Change input byte ordering (Native)
-noh No HTK header to output file (FALSE)

-noc0 No c0 coefficient to output feature vector (FALSE)
 -nologE No logE component to output feature vector (FALSE)
 -skip header bytes n - Skip header, first n bytes (Only for -F RAW)

-noh, -noc0, -nologE and -skip\_header\_bytes are not used and should not be changed.

#### FrontEnd w/o Extension:

USAGE: bin/AdvFrontEnd infile HTK\_outfile [options]

OPTIONS: - Same as FrontEnd w/ Extension

#### **Vector Quantization w/ Extension:**

Usage: extcoder htk file in pitch file in class file in bitstream file out pitch file out txt file out -freq x -

VAD/No\_VAD

pitch\_file\_out txt\_file\_out -freq x Output quantised pitch period file. Vector quantiser output in text format. Sampling frequency in kHz (8 or 16).

-VAD Use voice activity detector data. Voice activity input file must have same name as htk\_file, but

extension .vad

-No\_VAD Do not incorporate voice activity detector information in output bitstream.

### **Vector Quantization w/o Extension:**

Usage: coder htk\_file\_in bitstream\_file\_out txt\_file\_out -freq x -VAD/No\_VAD htk\_file\_in Input mel-frequency cepstral coefficient file in HTK MFCC format.

bit\_file\_out Binary output bitstream.

txt\_file\_out Vector quantiser output in text format.
-freq x Sampling frequency in kHz (8 or 16).

-VAD Use voice activity detector data. Voice activity input file must have same name as htk\_file, but

extension .vad

-No\_VAD Do not incorporate voice activity detector information in output bitstream.

## File extension descriptions as generated by the sample script:

.cep – Binary file containing cepstral features in HTK format. Output from the FrontEnd, input to the vector quantizer. .pitch – Binary file containing pitch information. Output from the FrontEnd, input to the vector quantizer. Only used for Extension.

.class – Ascii file containing class information. Output from the FrontEnd, input to the vector quantizer. Only used for Extension.

.bs – Binary file containing the bitstream. Output from the vector quantizer.

.log – Log files from the different executables.

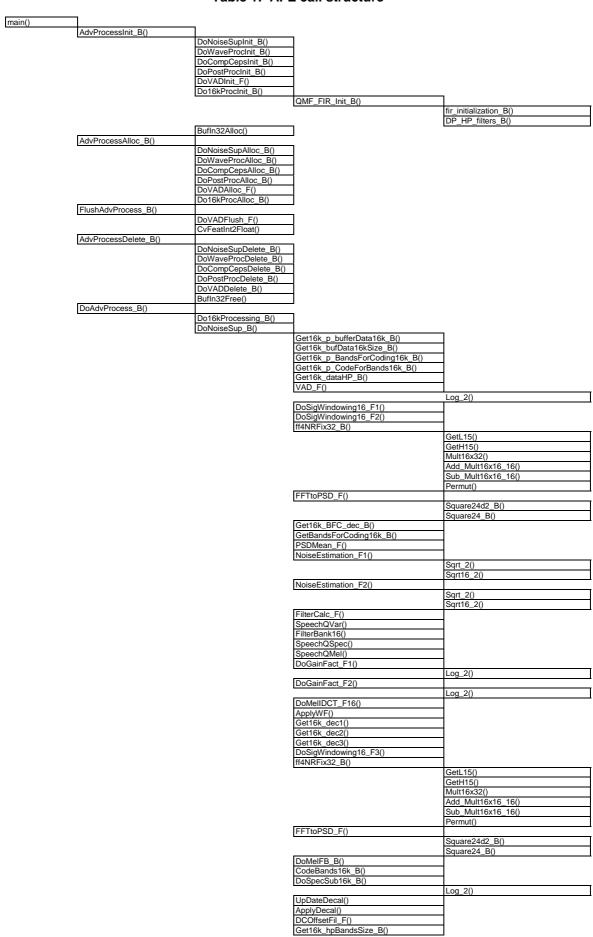
# 4.3 Code hierarchy

Tables 1 to 3 are call graphs that show the functions used for AFE (table 1), VQ (table 2), and Extension (table 3).

Each column represents a call level and each cell a function. The functions contain calls to the functions in rightwards neighboring 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 basic operations are not counted as extending the depth, therefore the deepest level in this software is level 7.

Table 1: AFE call structure



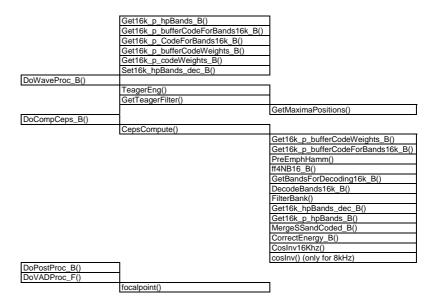
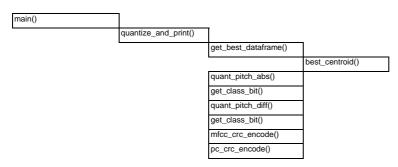
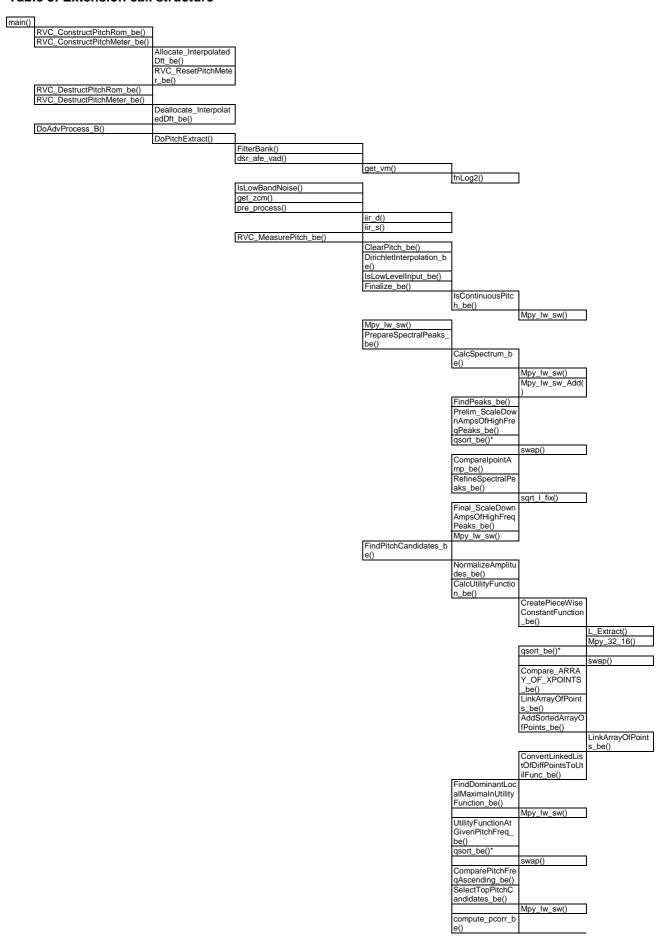
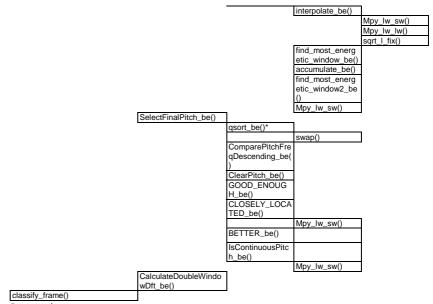


Table 2: VQ call structure



**Table 3: Extension call structure** 





\* qsort\_be() is a recursive function

# 4.5 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.

# 4.5.1 Description of constants used in the C-code

Table 5a: Global constants for AFE

| Constant                         | Value       | Description  |
|----------------------------------|-------------|--|
| NS_SPEC_ORDER_16K                | 64          | Noise suppression Array length   |
| NS_HANGOVER_16K                  | 15          | Noise suppression hangover count   |
| NS_MIN_SPEECH_FRAME_HANGOVER_16K | 4           | Noise suppression minmum speech frame hangover count                         |
| NS_ANALYSIS_WINDOW_16K           | 80          | Noise suppression analysis window  |
| PERC_CODED                       | 0.7         | lambda merge (empirically set constant)                                      |
| LAMBDA_NSE16k                    | 0.99        | Noise estimation Lambda  |
| NS_NB_FRAME_THRESHOLD_NSE        | 100         | Noise suppression number of frame threshold used for NSE                     |
| LENGTH_QMF                       | 118         | QMF filter length  |
| f24                              | 1           | multiplier for QMF filter coefficients                                       |
| SHFF_H                           | 8           | shift to get higher value  |
| L_H                              | 16          | shift to get lower value   |
| HP16k_MEL_USED                   | 3           | Higher frequnecy band Mel used   |
| NB_LP_BANDS_CODING               | 3           | Lower frequency band used in coding  |
| NE16k_FRAMES_THRESH              | 100         | Noise estimation frames threshold  |
| NB_TOPOSTPROC                    | 12          | Number of coefficients to postprocess  |
| CEP FRAME LENGTH                 | 200         | Frame length for cepstral coefficients                                       |
| CEP NB COEF                      | 13          | Number of cepstral coefficients (including c0)                               |
| CEP NB CHANNELS                  | 23          | Number of filters used for cepstral coefficients                             |
| CEP_FFT_LENGTH                   | 256         | FFT length for cepstral coefficients   |
| FRAME_BUF_SIZE                   | 241         | Denoised Output buffer size  |
| FRAME SHIFT                      | 80          | WaveProcessing input frame shift   |
| FRAME_LENGTH                     | 200         | WaveProcessing frame size  |
| NS SPEC ORDER                    | 65          | Noise suppression array length (8khz)  |
| NS_BUFFER_SIZE                   | 180         | Noise suppression past frame size  |
| NS_FRAME_SHIFT                   | 80          | Noise suppression input frame shift  |
| NS HALF FILTER LENGTH            | 8           | Noise suppression filter half size   |
| NS_NB_FRAME_THRESHOLD_LTE        | 10          | Noise suppression long term energy forgetting factor threshold (in frames)   |
| NS NB FRAME THRESHOLD NSE        | 100         | Noise suppression spectrum estimate forgetting factor threshold (in frames)  |
| NS_MIN_FRAME                     | 10          | Number of frame threshold to update average energy for Nosie suppression VAD |
| NS FFT LENGTH                    | 256         | FFT length for noise suppression   |
| WF_MEL_ORDER                     | 25          | Noise suppression Wiener filter order  |
| SHFT NOISE                       | 14          | shift applied to noise spectrum estimate                                     |
| SHFT FACT MUL                    | 14          | shift applied to gain coefficient (nosie suppression gain factoriization)    |
| IDCT_ORDER                       | 25          | Noise suppression idct order   |
| NS BETA                          | 0.98        | Noiseless signal suppression factor  |
| NS RSB MIN                       | 0.079432823 | Minimum a priori SNR   |
| NS LAMBDA NSE                    | 0.99        | Forgetting factor for noise spectrum estimate                                |
| NS LOG SPEC FLOOR                | -10.0       | average energy minimum threshold   |
| NS_SNR_THRESHOLD_VAD             | 15          | SNR threshold for noise suppression VAD                                      |
| NS_SNR_THRESHOLD_UPD_LTE         | 20          | Long term energy update threshold for noise suppression VAD                  |
| NS_ENERGY_FLOOR                  | 80          | Energy Minimum threshold for noise suppression VAD                           |
| MaxPos                           | 10          | Maximum number of maxima in waveprocessing                                   |
| WP EPS                           | 0.2         | weigthing value added or substracted for waveprocessing                      |

## Table 5b: Global constants for VQ

| Constant           | Value   | Description                            |
|--------------------|---------|--|
| MIN_PERIOD         | 1245184 | Minimum pitch period allowed           |
| MAX_PERIOD         | 9175040 | Maximum pitch period allowed           |
| NUM_MULTI_LEVELS_1 | 26      | number of levels in pitch quantization |
| NUM_MULTI_LEVELS_2 | 24      | number of levels in pitch quantization |
| UNVOICED_CODE      | 0       | init value for Qpindex                 |

## Table 5c: Global constants for Extension

| Constant                   | Value  | Description   |
|----------------------------|--------|---|
| HISTORY_LEN                | 100    | History length - past samples for pitch extraction      |
| DOWN_SAMP_FACTOR           | 4      | Down-sampling factor - used in computing correlation    |
| NO_OF_DFT_POINTS           | 128    | Number of DFT points                                    |
| BREAK_POINT                | 12     | Break point - marks the end of low frequency band       |
| LBN_HIST_WEIGHT            | 32440  | Low band noise history weight                           |
| LBN_CURR_WEIGHT            | 328    | Low band noise current weight (32768 - LBN_HIST_WEIGHT) |
| LBN_MAX_THR                | 124518 | Low band noise maximum threshold                        |
| LBN_LOW_ENR_LEVEL_MANT     | 32000  | Low band noise low energy level mantissa                |
| LBN_LOW_ENR_LEVEL_SHFT     | 22     | Low band noise low energy level shift                   |
| RVC_OK                     | 0      | Return code for success                                 |
| RVC_ERR                    | -1     | Return code for unspecified error                       |
| RVC_ERR_NOT_ENOUGH_MEMORY  | -2     | Return code for not enough memory                       |
| RVC_ERR_ILLEGAL_ARGUMENT   | -3     | Return code for an illegal input / output argument      |
| RVC_ERR_IO_FAILED          | -4     | Return code for failed input / output to a file         |
| RVC_ERR_BAD_FILE_FORMAT    | -5     | Return code for a bad file header                       |
| RVC_ERR_NOT_INITIALIZED    | -6     | Return code for failure due to improper initialization  |
| RVC_ERR_ILLEGAL_USAGE      | -7     | Return code for illegal usage of a function             |
| RVC_ERR_NOT_ENOUGH_SAMPLES | -8     | Return code for insufficient number of samples          |
| RVC_ERR_NOT_IMPLEMENTED    | -9     | Return code for an unimplemented function               |

| MR. PERION, PRIACE   59   Department of the price of th   | DVC EDD EAT ODEN EILE                 | -10         | Deturn code for failure to open a file                             |
|--|---------------------------------------|-------------|--|
| 2007 TOAL   100   2007   200   | RVC_ERR_FAIL_OPEN_FILE LIB_ENRG_FRAC: |             | Return code for failure to open a file  Upper hand energy fraction |
| SOME ON PAINT  | ZCM THLD                              |             |  |
| FRAME   First   Dis NY 2   25  | SQRT_ONE_HALF                         | 0x5A82      |  |
| HISTORY_LEL_GOS  |                                       | 50          | Frame length downsampled (200/4)                                   |
| WINDOW LENGTH  18 Window Length  18 So Press of author length total in companion controllation  Window Length  18 So Press of author length (15 - 4,05550)  Window Length   |                                       |             |  |
| INC. WINDOW, LENGTH  1620  162 |                                       |             |  |
| NIAD CHAN OL PARTO, MATISSA  2000 Monthly College Control of the Programmy bands MIT, SEC, ENTIRE, SHATTSA  2000 Monthly College Colle |                                       |             |  |
| Mill CHE LINES (AMTISSA  2000) Minimum channed energy markinsss Mill CHE LINES (SHITT)  38 Minimum channed energy markinss Mill CHE LINES (SHITT)  38 Minimum channed energy markinss Mill Sell CHE LINES (SHITT)  38 Minimum channed energy markinss Mill Sell CHE LINES (SHITT)  38 Minimum channed energy markinss Mill Sell CHE LINES (SHITT)  38 Minimum channed energy markinss Mill Sell CHE LINES (SHITT)  38 Minimum channed energy markinss Mill Sell CHE LINES (SHITT)  39 Minimum channed energy markinss Mill Sell CHE LINES (SHITT)  30 Cannel and sell sell check (Shitt)  30 Cannel and sell sell check (Shitt)  30 Cannel and sell sell sell sell sell sell sell sel  |                                       |             |  |
| Mile City Miles SHIFT  |                                       |             |  |
| NRT SIG ENRG MANTISSA  2018 B. Once signed energy professor  NRT SIG ENRG SISHET  8 7 THE SIG ENRG SISHET  8 7 THE SIG ENRG COOPE  1474 C. CO |                                       |             |  |
| No.   Section    | INIT SIG ENRG MANTISSA                |             |  |
| DE SIM FAC COMPL  14746  14746  14747  14747  14747  14747  15747 | INIT_SIG_ENRG_SHIFT                   | 8           | Initial signal energy shift  |
| CRISTON   COMPL   29451   Charmel noise energy emothing laters comprehens  | CE_SM_FAC                             | 18022       | Channel energy smoothing factor                                    |
| CRIESTIF FAC COMPL   29491   Channel noise enemy surrouting laster complement   D. CAMMA   2958   Deep german value   Deep g   |                                       |             |  |
| CG GAMMA COMPR.   29288   Low garmar value   Complement  |                                       |             | 0, 0   |
| G.G.MAMA.COMPL.  |                                       |             |  |
| H. GAMMA   |                                       |             |  |
| IEGANIAC COMPR   |                                       |             |  |
| 10 BETA  |                                       |             |  |
| Initial number of frames (considered to be noise frames)   SINE START CHAN   |                                       |             |  |
| Since Start Charter   4   Since start character (for sine wave electricis)   | HI_BETA                               | 32702       | High beta value  |
| PEAK_TO_AVE_THLD   | INIT_FRAMES                           | 10          | Initial number of frames (considered to be noise frames)           |
|  |                                       |             |  |
| HYSTER_CHT_HLD   |                                       |             |  |
| F. UPDATE CNT_THLD   |                                       |             |  |
| NON_SPECH_TH_D    32   |                                       |             |  |
| FIX. 14  |                                       |             |  |
| Fig. 18  |                                       |             |  |
| 1  |                                       |             |  |
| SWTHIRD. REF. BANDWIDTH         85         One third of the reference bandwidth           SWTWO.THIRDS. REF. BANDWIDTH         171         Two thirds of the reference bandwidth           MIN. ENERGY. SHET         18         Minimum energy shift           SWERS. SMER. EARLY QO         0x1F4D         Reference sampling rate in Qo format           SWERS. SMER. EARLY QO         0x1F4D         Reference sampling rate in Qo format           SWERS. SMER. EARLY QO         0x1F4D         Reference sampling rate in Qo format           SWERS. SWER. THUD QTS         0x0C         Coconsists           SWERS. SCHE THUD QTS         0x0C         0x0C           SWERS. THUD QTS         0x0C         0x0C           SWERS. THUD QTS         0x0C         0x0C           SWCANDORT THUD QTS         0x0C         0x0C           SWCANDORT THUD QTS         0x0C         0x0C         0x0C           SWCANDORT THUD QTS         0x0C         0x0C         0x0C           SWCANDORT THUD QTS         0x0C         0x0C         0x   | = :                                   |             |  |
| Minimum energy martissa   25000  | swTHIRD_REF_BANDWIDTH                 | 85          |  |
| MIN_ENRRY_SHIFT  |                                       |             |  |
| SWEFE SAMPLE RATE 00   |                                       |             |  |
| swCL DSCE_TACTOR_Q14         OvACCD         Closeness factor in Q14 format           swFD_SCORE_TH_D10_155         0x8507         Frequency domain score threshold 1 in Q15 format           swCORR_TH_D0_155         0x8517         Frequency domain score threshold 2 in Q15 format           swSUM_TH_D0_141         0x8617         Frequency domain score threshold 2 in Q15 format           swCANDCORE_TH_D10_155         0x0000170A         Sum threshold in Q14 format           swCANDCORE_TH_D2_015         0x0000170A         Pitch candidate correlation threshold 2 in Q15 format           swCANDCORE_TH_D2_015         0x00000170A         Pitch candidate correlation threshold 2 in Q15 format           swCANDCORE_TH_D3_015         0x060000000000000000000000000000000000  |                                       |             |  |
| swFD_SCORE_THILD2_015         Ox8307         Frequency domain score threshold 1 in 015 format           swCORD_THILD_015         0x6517         Correlation threshold in 015 format           swCORD_THILD_014         0x6667         Correlation threshold in 015 format           wCRITO_OFFSET_015         0x0000170A         Offset for finding a better pictic nacidate in 015 format           wCANDCORR_THILD2_015         0x0000170A         Offset for finding a better pictic nacidate in 015 format           swCANDCORR_THILD3_015         0x599A         Pitch candidate correlation threshold 1 in 015 format           swCANDCORR_THILD3_015         0x66CD         Disc candidate correlation threshold 2 in 015 format           swCANDCORR_THILD3_015         0x66CD         Pitch candidate correlation threshold 3 in 015 format           swCANDCORR_THILD3_015         0x66CD         Pitch candidate amplitude threshold 3 in 015 format           swCANDCARR_THILD3_015         0x66CD         Pitch candidate amplitude threshold 3 in 015 format           swCANDCARR_THILD3_015         0x66CD         Pitch candidate amplitude threshold 3 in 015 format           swCANDCARR_THILD3_015         0x66CD         Pitch candidate correlation threshold 3 in 015 format           swCANDCARR_THILD3_015         0x66CD         Pitch candidate correlation threshold 3 in 015 format           swCANDCARR_THILD3_015         0x66CD         Pitch candidate co  |                                       |             |  |
| swFD_SCORE_THLD2_015         Ox570A         Frequency domain score threshold 2 in Q15 format           swCORR_THLD_015         0x651F         Ox6567           swSUM_THLD_014         0x6567         Sum threshold in Q14 format           swCART_DEST_015         0x0000170A         Ox5667           swCANDCORR_THLD1_015         0x799A         Pitch candidate correlation threshold 1 in Q15 format           swCANDCORR_THLD3_015         0x599A         Pitch candidate correlation threshold 3 in Q15 format           swCANDADP_THLD3_015         0x505F         Pitch candidate correlation threshold 3 in Q15 format           swCANDADP_THLD3_015         0x505F         Pitch candidate samplitude threshold 3 in Q15 format           swCANDADP_THLD3_015         0x555F         Pitch candidate samplitude threshold 3 in Q15 format           swCANDADP_THLD3_015         0x555F         Pitch candidate samplitude threshold 3 in Q15 format           swCANDADP_THLD3_015         0x555F         Pitch candidate samplitude threshold 3 in Q15 format           swCANDADP_THLD3_015         0x555F         Samplitude state should 3 in Q15 format           swCENTER_CENTSPAN         0x555F         Samplitude state should 3 in Q15 format           swCENTER_LEARTSPAND         0x555F         Samplitude state should 3 in Q15 format           swTall partition of the state state should state state state state state state state sta  |                                       |             |  |
| SWOUNTHLD_Q14  Ox6667  WCRITD_Q15  Ox6667  WCRITD_OFFSET_Q15  Ox0000170A  Offset for finding a better pitch candidate in Q15 format  WCRITD_OFFSET_Q15  Ox0000170A  Offset for finding a better pitch candidate in Q15 format  WCRITD_OFFSET_Q15  Ox799A  Pitch candidate correlation threshold 2 in Q15 format  WCRITD_ORD 15  Ox66CD  Diffset ox66CD  Pitch candidate correlation threshold 2 in Q15 format  WCRITD_ORD 15  Ox66CD  Ox66CD  Diftch candidate correlation threshold 2 in Q15 format  WCRITD_ORD 15  Ox66CD  Ox66CD  Diftch candidate correlation threshold 2 in Q15 format  WCRITD_ORD 15  Ox66CD  Ox66CD  Diftch candidate correlation threshold 2 in Q15 format  WWCATHERG_COEFF  Ox66SF  Ox66SF  Diftch candidate angilitude threshold 3 in Q15 format  WWCATHERG_COEFF  Ox66SF  Ox66SF  Diftch candidate correlation threshold 3 in Q15 format  WWCATHERG_COEFF  Ox66SF  Ox66SF  Diftch candidate correlation threshold 3 in Q15 format  WWCATHERG_COEFF  Ox66SF  Ox66SF  Ox66SF  Diftch candidate correlation threshold 3 in Q15 format  WWCATHERG_COEFF  Ox66SF  Ox66SF  WWCATHERG_COEFF  Ox66SF  Ox66SF  Ox66SF  Diftch candidate correlation threshold 3 in Q15 format  WWCATHERG_COEFF  Ox66SF  Ox66SF  WWCATHERG_COEFF  Ox66SF  Ox66SF  Ox66SF  WWCATHERG_COEFF  Ox66SF  Ox66SF  Ox66SF  WWCATHERG_COEFF  Ox66SF  O |                                       |             |  |
| wsw.Unith.LD_014         Ox6667         Sum threshold in Q14 format           wscANDCORR_THLD1_Q15         0x0000170A         Ox599A         Pitch candidate correlation threshold 1 in Q15 format           wscANDCORR_THLD2_Q15         0x599A         Pitch candidate correlation threshold 1 in Q15 format           swcANDCORR_THLD3_Q15         0x60CD         Pitch candidate correlation threshold 3 in Q15 format           swcANDADAP_THLD3_Q15         0x60CD         Pitch candidate angliquide threshold 3 in Q15 format           swSTARTREQ_COEFF         0x553F         Start frequency coefficient (for candidate search)           swSTARTREQ_COEFF         0x553F         Start frequency coefficient (for candidate search)           DIRICHLET_KERNEL_SPAN         8         Direchlet kernal span (for interpolation)           REF_BANDWIDTH         4000         Reference sampling rate           REF_BANDWIDTH         873931333         One third of the reference bandwidth           wifVHIRD_REF_BANDWIDTH         174762667         Two thirds of the reference bandwidth           wifVHIRD_REF_BANDWIDTH         174762667         Two thirds of the reference bandwidth           wwWENTER_WEIGHT         0x1800         Side weight           swWIP_SCALE_DOWN2         0x399A         Amplitude seale down factor 2           swWIP_SCALE_DOWN2         0x399A         Amplitude seale down fact   |                                       |             |  |
| swCANDCORR, THLD1, Q15         0x799A         Pitch candidate correlation threshold 1 in Q15 format           swCANDCORR, THLD3, Q15         0x599A         Pitch candidate correlation threshold 3 in Q15 format           swCANDCORR, THLD3, Q15         0x68CD         Pitch candidate correlation threshold 3 in Q15 format           swSTARTREQ, COEFF         0x553F         Start frequency coefficient (for candidate search)           swSTARTREQ, COEFF         0x553F         Start frequency coefficient (for candidate search)           DIRICHLET, KERNEL, SPAN         8         Direchlet kemal span (for interpolation)           REF, SAMPLE, RATE         8000         Reference sampling rate           REF, SAMPLE, RATE         8000         Reference bandwidth           WiTHOR, DEF, SAMDWIDTH         4000         Reference bandwidth           WiTHOR, DEF, SAMDWIDTH         1747e2667         Two thirds of the reference bandwidth           WWTWO, THIRDS, REF, BANDWIDTH         1747e2667         Two thirds of the reference bandwidth           WWWENTER, WEIGHT         0x5000         Center weight           SwWAMP, SCALE, DOWN1         0x5333         Amplitude scale down factor 1           SwWAMP, SCALE, DOWN2         0x399A         Amplitude scale down factor 2           SwUDIST2         4460         Uillity function distance 2           SwUDIST3   | swSUM_THLD_Q14                        |             |  |
| swCANDCORR THLD2 Q15         0x599A         Pitch candidate correlation threshold 2 in Q15 format           swCANDCORR THLD3 Q15         0x6CDD         Pitch candidate correlation threshold 3 in Q15 format           swCANDCARP THLD3 Q15         0x68F6         Pitch candidate correlation threshold 3 in Q15 format           swCANDCARP COEFF         0x555F         Start frequency coefficient (for candidate search)           DIRICHLET KERLE SPAN         8         Direchlet kernel span (for interpolation)           REF SAMPLE RATE         8000         Reference sampling rate           REF BANDWIDTH         4000         Reference bandwidth           WITHIRD REF_BANDWIDTH         87381333         One third of the reference bandwidth           WITHIRD REF_BANDWIDTH         1747026677         Two thirds of the reference bandwidth           WITHIRD REF_BANDWIDTH         0x5000         Center weight           swCENTER_WEIGHT         0x5000         Center weight           swALD SCALE_DOWN1         0x5333         Amplitude scale down factor 1           swAMP_SCALE_DOWN2         0x3393         Amplitude scale down factor 2           swAMP_SCALE_DOWN2b         0x7333         Amplitude scale down factor 1           swUDIST1         -4460         Ultility function distance 1           swUJSTEP         -16384         Ultility function stable  | IwCRIT0_OFFSET_Q15                    | 0x0000170A  | Offset for finding a better pitch candidate in Q15 format          |
| swCANDCORR_THLD3_015  0x68F6 Pitch candidate correlation threshold 3 in Q15 format swSTARTREQ_COEFF  |                                       |             |  |
| swCANDAMP_THLD3_015 swSCARPEC_COEFF 0x565F 0x6666 End frequency coefficient (for candidate search) swENDFREQ_COEFF 0x4666 End frequency coefficient (for candidate search) swENDFREQ_COEFF 0x4666 End frequency coefficient (for candidate search) DIRICHLET_KERNEL_SPAN 8 Direchlet Kernel span for interpolation) REF_SAMPLE_RATE 8000 Reference sampling rate REF_BANDWIDTH 4000 Reference sampling rate WiTHIRD REF_BANDWIDTH 87391333 One third of the reference bandwidth WITHIRD REF_BANDWIDTH 174762667 Two thirds of the reference bandwidth WITHIRD REF_BANDWIDTH 17476267 Two thirds of the reference bandwidth SwCENTER_WEIGHT 0x5000 Center weight SwCENTER_WEIGHT 0x5000 Center weight SwCENTER_WEIGHT 0x5000 Amplitude scale down factor 1 swAMP_SCALE_DOWN12 0x399A Amplitude scale down factor 2 swAMP_SCALE_DOWN2 0x399A Amplitude scale down factor 2 swJDIST1 4160 Utility function distance 2 SwJDIST1 4160 Utility function distance 2 SwJDIST2 4600 Utility function distance 2 SwJDIST1 9x404P SCALE_DOWN2 0x399A MARGIN1 0x533 Anglitude scale down factor 2 SwJDIST1 4160 Utility function distance 2 SwJDIST2 4600 Utility function distance 2 SwJDIST2 476084 Utility function distance 2 SwJDIST3 476084 Amplitude margin 1 SwAMP_SCALE_PRAMES 6 Minimum number of stable frames MMA_TRANC_ROA_PERAMES 7 Amplitude margin 1 SwAMP_SCALE_PRAMES 8 Min.STABLE_FRAMES 9 Amplitude margin 1 SwAMP_SCALE_PRAMES 9 Amplitude margin 1 S |                                       |             |  |
| swSTARTREQ_COEFF   |                                       |             |  |
| swenDFREQ_COEFF   0x4666   End frequency coefficient (for candidate search)   DIRICHLET KERNEL_SPAN   8   Direchlet Kernel_Span (for interpolation)   REF_SANPLE_RATE   80.00   Reference sampling rate   REF_BANDWIDTH   40.00   Reference bandwidth   WTHIRD_REF_BANDWIDTH   87381333   One bird of the reference bandwidth   WTHIRD_REF_BANDWIDTH   174762667   Two thirds of the reference bandwidth   WTHIRD_REF_BANDWIDTH   0x5000   Center weight   SwSIDE_WEIGHT   0x5000   Center weight   SwSIDE_SCALE_DOWN2   0x399A   Amplitude scale down factor 1   SwSIDE_SCALE_DOWN2   0x399A   Amplitude scale down factor 2   SwSIDE_SCALE_DOWN2   0x399A   Amplitude margin 1   SwSIDE_SCALE |                                       |             |  |
| BIRCHLET_KERNEL_SPAN   8   |                                       |             |  |
| REF SAMPLE RATE  8000 Reference sampling rate REF BANDWIDTH  4000 Reference bandwidth  WTHIRD REF BANDWIDTH  87381333 One third of the reference bandwidth  WTWO THIRDS REF BANDWIDTH  174762667 Two thirds of the reference bandwidth  WWO THIRDS REF BANDWIDTH  174762667 Two thirds of the reference bandwidth  WWO THIRDS REF BANDWIDTH  174762667 Two thirds of the reference bandwidth  WWO THIRDS REF BANDWIDTH  174762667 Two thirds of the reference bandwidth  WWO THIRDS REF BANDWIDTH  174762667 Two thirds of the reference bandwidth  WWO THIRDS REF BANDWIDTH  WWO THIRDS REF BANDWIDTH  174762667 Two thirds of the reference bandwidth  WWO THIRDS REF BANDWIDTH  WWANDWIDTH  WWANDWIDTH  WWANDWIDTH  WWANDWIDTH  WWANDWIDTH  WWO THIRDS REF BANDWIDTH  WWO THIRDS REFERD BANDWIDTH  WWO THIRDS REF BANDWIDTH  WWO THIRDS REF BANDWIDTH |                                       |             |  |
| REF BANDWIDTH WTWO THIRDS REF BANDWIDTH 174762667 Two thirds of the reference bandwidth WTWO THIRDS. REF BANDWIDTH 174762667 Two thirds of the reference bandwidth 9740000 SecENTER WEIGHT 985000 SecHER WEIGHT 985000 Side weight 9850000 Side weight 98500000 Side weight 9850000000000 Side weight 985000000000000000000000000000000000000  |                                       |             |  |
| Immonth   Immonthh   Immonth   | REF_BANDWIDTH                         | 4000        | Reference bandwidth  |
| swCENTER_WEIGHT swSMP_SCALE_DOWN1 swAMP_SCALE_DOWN2 swAMP_SCALE_DOWN2 byAMP_SCALE_DOWN2 byAMP_SCALE_DO |                                       |             | One third of the reference bandwidth                               |
| SwSDE_VEIGHT   |                                       |             |  |
| SWAMP SCALE DOWN2 SWAMP SCALE DOWN2 SWAMP SCALE DOWN2b SWAMP SCALE DOW |                                       |             |  |
| SWAMP SCALE DOWN2b SWAMP SCALE DOWN2b SWAMP SCALE DOWN2b SWLOTST 4-4160 Uitility function distance 1 SWLOTST 4-4160 Uitility function distance 2 UItility function distance function distance function distance function dop lint for sind window dop function distance function dop lint for sind window dop f |                                       |             | ·  |
| SWADP SCALE_DOWN2b SWLDIST1 4-8160 Utility function distance 1 SWLDIST2 -6400 Utility function distance 2 SWLDISTEP -16384 Utility function distance 2 SWLDISTEP SWAMP_MARGIN1 -16384 Minimum print frequency margin 1 SWAMP_MARGIN2 -16384 Amplitude margin 1 SWAMP_MARGIN2 -16384 Minimum number of stable frames -16384 Minimum number of pack frames -16384 Minimum number of frames -16384 Minimum number of peaks (fraguency) -16384 Minimum number of  |                                       |             |  |
| SWUDIST1   |                                       |             |  |
| swUSTEP  |                                       |             |  |
| SWJSTEP SWFREQ_MARGIN1 SWAMP_MARGIN1 SWAMP_MARGIN2 SWAMP_MARGIN3 SWAMP_M | swUDIST2                              |             |  |
| swAMP_MARGIN1  |                                       |             |  |
| swAMP_MARGIN2         0x07AE         Amplitude margin 2           MIN_STABLE_FRAMES         6         Minimum number of stable frames           MAX_TRACK_GAP_FRAMES         2         Maximum pitch track gap frames           swSTABLE_FREQ_UPPER_MARGIN         0x4E14         Stable frequency upper margin           swSTABLE_FREQ_LOWER_MARGIN         0x68EB         Stable frequency lower margin           UNVOICED         0         Pitch frequency of an unvoiced frame           IWMAX_PITCH_FREQ         0x01440000L         Maximum pitch frequency           IWMN_PITCH_FREQ         0x034000L         Minimum pitch frequency           MAX_PITCH_FREQ         420         Maximum pitch frequency in Hz           MIN_PITCH_FREQ         420         Maximum pitch frequency in Hz           MIN_PITCH_FREQ         52         Minimum pitch frequency in Hz           MIN_PITCH_FREQ         52         Minimum pitch frequency in Hz           NO_OF_FRACS         77         Number of fractions in the frations table           WSHORT_WIN_START_FREQ         0x00C80000L         Short window start frequency           WSHORT_WIN_END_FREQ         0x00640000L         Single window start frequency           WSINGLE_WIN_END_FREQ         0x00340000         Double window start frequency           WSOUBLE_WIN_END_FREQ         0x   | swFREQ_MARGIN1                        |             |  |
| MIN_STABLE_FRAMES  AX_TRACK_GAP_FRAMES  2 Maximum number of stable frames  MAX_TRACK_GAP_FRAMES  2 Maximum number of peaks (preliminary)  Min_PITCH_FREQ  MAX_DITCH_FREQ  MAX_DEAKS_FOR_SORT  MOX_DITCH_FREQ  MAX_DITCH_FREQ  MAX_DITCH_FREQ  MAX_DEAKS_FINAL  MAX_DEAKS_FINAL  MAX_MAX_DEAKS_FINAL  MAX_MAX_DEAKS_FINCL_LOOP_LIM_SNG  MAX_DEAKAT_PICH_COOP_LIM_SNG  MAX_DEAKAT_PICH_COOP_LIM_SNG  MAX_DEAKAT_PICH_COOP_LIM_SNG  MAX_DEAKAT_PICCL_DOP_LIM_SNG  MAX_DEAKAT_PICCL_DOP_LIM_SNG  MAX_DEAKAT_PICCL_DOP_LIM_SNG  MAX_DEAKAT_PICCL_DOP_LIM_SNG  MAX_DEAKAT_PICCL_DOP_LIM_SNG  MAX_DEAKAT_PICCL_DOP_LIM_SNG  MAX_DECKATE_PICCCWISE_FUNC_LOOP_LIM_SNG  MAX_DEAKAT_PICCL_DOP_LIM_SNG  MAX_DEAKAT_PICCL_DITCH_SNG  MAX_DEAKAT_PICCL_DOP_LIM_SNG  MAX_DEAKAT_PICCL_DOP_LIM_SNG  MAX_DEAKAT_PICCL_DOP_LIM_SNG  MAX_DECKATE_PICCCWISE_FUNC_LOOP_LIM_SNG  MAX_DEAKAT_PICCL_DOP_LIM_SNG  MAX_DITCL_DITCL_DOP_LIM_SNG  MAX_DITCL_DITCL_DOP_LIM_SNG  MAX_DITCL_DITCL_DOP_LIM_SNG  MAX_DITCL_DITCL_DOP_LIM_SNG  MAX_DITCL_DITCL_DOP_LIM_SNG  MAX_DITCL_DITCL_DOP_LIM_SNG  MAX_DITCL_DITCL_DOP_LIM_SNG  MAX_DITCL_DITCL_DOP_LIM_SNG  MAX_DITCL_DITCL_DOP_LIM_SNG  MAX_DITCL_DITCL_DOP_LIM_SN |                                       |             |  |
| MAX_TRACK_GAP_FRAMES  2 Maximum pitch track gap frames  swSTABLE_FREQ_UPPER_MARGIN  0x4E14 Stable frequency upper margin  wSMSTABLE_FREQ_LOWER_MARGIN  0x68EB Stable frequency lower margin  UNVOICED  0 Pitch frequency of an unvoiced frame  iwMAX_PITCH_FREQ  0x01A40000L Maximum pitch frequency  MAX_PITCH_FREQ  420 Maximum pitch frequency  MAX_PITCH_FREQ  420 Maximum pitch frequency  MAX_PITCH_FREQ  MIN_PITCH_FREQ  420 Maximum pitch frequency in Hz  HIGHPASS_CUTOFF_FREQ  NO_OF_FRACS  77 Number of fractions in the frations table  wSHORT_WIN_START_FREQ  0x00C8000L Short window start frequency  iwSHORT_WIN_END_FREQ  0x001A40000 Short window start frequency  iwSINGLE_WIN_START_FREQ  0x00C8000L Single window and frequency  iwDOUBLE_WIN_END_FREQ  0x00340000 Double window start frequency  iwDOUBLE_WIN_END_FREQ  0x00340000 Double window start frequency  iwDOUBLE_WIN_END_FREQ  0x00340000 Double window start frequency  MAX_DCAL_MAXIMA_ON_SPECTRUM  70 Maximum number of local maxima on the spectrum  MAX_PEAKS_POR_SORT  30 Maximum number of peaks (preliminary)  MIN_PEAKS  7 Minimum number of peaks (preliminary)  MIN_PEAKS  MAX_PEAKS_FINAL  4 Maximum number of peaks (final)  MAX_PECEWISE_FUNC_LOOP_LIM_SNG  20 Create Piecewise function loop limit for short window  CREATE_PIECEWISE_FUNC_LOOP_LIM_SNG  20 Create Piecewise function loop limit for short window  CREATE_PIECEWISE_FUNC_LOOP_LIM_SNG  20 Create Piecewise function loop limit for short window  CREATE_PIECEWISE_FUNC_LOOP_LIM_SNG  20 Create Piecewise function loop limit for short window  |                                       |             |  |
| swSTABLE_FREQ_UPPER_MARGIN  swSTABLE_FREQ_LOWER_MARGIN  0x68EB  Stable frequency upper margin  0 Pitch frequency of an unvoiced frame  lwMax_PITCH_FREQ  0x01A40000L  Maximum pitch frequency  lwMIN_PITCH_FREQ  0x034000L  Minimum pitch frequency  MAX_PITCH_FREQ  420  Maximum pitch frequency in Hz  MIN_PITCH_FREQ  420  Maximum pitch frequency in Hz  MIN_PITCH_FREQ  420  Minimum pitch frequency in Hz  Minimum start frequency  Minimum number of local maxima on the spectrum  MAX_PEAKS_FOR_SORT  30  Maximum number of local maxima on the spectrum  MAX_PEAKS_FOR_SORT  30  Maximum number of peaks (final)  Minimum number of peaks (final)  MAX_PEAKS_FINAL  4  Maximum number of peaks (final)  MAX_PEAKS_FINAL  4  Maximum number of peaks (final)  MAX_PEAKS_FUNC_LOOP_LIM_SH  20  Create Piecewise function loop limit for single window  CREATE_PIECEWISE_FUNC_LOOP_LIM_SH  20  Create Piecewise function loop limit for single window   |                                       |             |  |
| SWSTABLE_FREQ_LOWER_MARGIN  0  |                                       |             |  |
| UNVOICED  0 Pitch frequency of an unvoiced frame   wMAX_PITCH_FREQ   |                                       |             |  |
| IwMAX_PITCH_FREQ     0x01A40000L     Maximum pitch frequency       IwMIN_PITCH_FREQ     0x00340000L     Minimum pitch frequency in Hz       MIN_PITCH_FREQ     420     Maximum pitch frequency in Hz       MIN_PITCH_FREQ     52     Minimum pitch frequency in Hz       HIGHPASS_CUTOFF_FREQ     300     Highpass cut-off frequency in Hz       NO_OF_FRACS     77     Number of fractions in the frations table       IwSHORT_WIN_START_FREQ     0x00C80000L     Short window start frequency       IwSHORT_WIN_END_FREQ     0x01A4000     Short window end frequency       IwSINGLE_WIN_START_FREQ     0x00640000L     Single window end frequency       IwSINGLE_WIN_START_FREQ     0x00040000     Single window end frequency       IwDOUBLE_WIN_START_FREQ     0x00340000     Double window start frequency       IwDOUBLE_WIN_START_FREQ     0x00340000     Double window end frequency       IwDOUBLE_WIN_START_FREQ     0x00780000L     Double window end frequency<  |                                       |             |  |
| IwMIN_PITCH_FREQ     0x00340000L     Minimum pitch frequency       MAX_PITCH_FREQ     420     Maximum pitch frequency in Hz       MIN_PITCH_FREQ     52     Minimum pitch frequency in Hz       HIGHPASS_CUTOFF_FREQ     300     Highpass cut-off frequency in Hz       NO_OF_FRACS     77     Number of fractions in the frations table       IwSHORT_WIN_END_FREQ     0x00C80000L     Short window start frequency       IwSHORT_WIN_END_FREQ     0x01A4000     Short window end frequency       IwSINGLE_WIN_START_FREQ     0x00640000L     Single window start frequency       IwSINGLE_WIN_START_FREQ     0x00020000L     Single window end frequency       IwDUBLE_WIN_START_FREQ     0x00340000     Double window start frequency       IwDOUBLE_WIN_START_FREQ     0x00340000     Double window start frequency       IwDOUBLE_WIN_START_FREQ     0x00340000     Double window end frequency       IwDOUBLE_WIN_START_FREQ     0x00780000L     Double window end frequency       IwDOUBLE_WIN_START_FREQ     0x00780000L     Double window end frequency       MAX_LOCAL_MAXIMA_ON_SPECTRUM     70     Maximum number of local maxima on the spectrum       MAX_PEAKS_FOR_SORT     30     Maximum number peaks (frosting)       MAX_PEAKS_FINAL     7     Maximum number of peaks (final)       MAX_PEAKS_FINAL     20     Maximum number of peaks (final)    <  |                                       | 0x01A40000L | Maximum pitch frequency  |
| MIN_PITCH_FREQ  52 Minimum pitch frequency in Hz HIGHPASS_CUTOFF_FREQ  300 Highpass cut-off frequency in Hz Number of fractions in the frations table NSHORT_WIN_START_FREQ NX00C80000L Short window start frequency NSHORT_WIN_END_FREQ NX01A40000 Short window start frequency NSHORT_WIN_END_FREQ NX01A40000 Short window end frequency NSHORT_WIN_END_FREQ NX00D20000L Single window start frequency NSHORLE_WIN_START_FREQ NX00D20000L Single window end frequency NSHORLE_WIN_START_FREQ NX00D20000L Single window end frequency NSHORLE_WIN_START_FREQ NX00D20000L Double window end frequency NDOUBLE_WIN_START_FREQ NX003A0000 Double window end frequency NDOUBLE_WIN_END_FREQ NX00780000L Double window end frequency NAX_PEAKS_FOR_SORT NOMAX_PEAKS_FOR_SORT NAX_PEAKS_PRELIM NAX_PEAKS_PRELIM NAX_PEAKS_FINAL NAX | IwMIN_PITCH_FREQ                      |             | Minimum pitch frequency  |
| HIGHPASS_CUTOFF_FREQ 300 Highpass cut-off frequency in Hz NO_OF_FRACS 77 Number of fractions in the frations table  WSHORT_WIN_START_FREQ 0x00C80000L WSHORT_WIN_END_FREQ 0x01A40000 Short window start frequency  WSINGLE_WIN_START_FREQ 0x00640000L Single window start frequency  WSINGLE_WIN_START_FREQ 0x00640000L Single window start frequency  WSINGLE_WIN_START_FREQ 0x000000L Single window end frequency  WSUNGLE_WIN_START_FREQ 0x00340000 Double window start frequency  WDOUBLE_WIN_START_FREQ 0x00780000L Double window end frequency  WAX_LOCAL_MAXIMA_ON_SPECTRUM 70 Maximum number of local maxima on the spectrum  MAX_PEAKS_FOR_SORT 30 Maximum number peaks for sorting  MAX_PEAKS_PRELIM 7 Maximum number of peaks (preliminary)  MIN_PEAKS 7 Minimum number of peaks  MAX_PEAKS_FINAL 20 Maximum number of peaks (final)  MAX_PELIM_CANDS 4 Maximum number of preliminary candidates (pitch)  CREATE_PIECEWISE_FUNC_LOOP_LIM_SNG 30 Create Piecewise function loop limit for single window  CREATE_PIECEWISE_FUNC_LOOP_LIM_SNG 30 Create Piecewise function loop limit for single window  | MAX_PITCH_FREQ                        |             |  |
| NO_OF_FRACS   Number of fractions in the frations table  |                                       |             |  |
| IwSHORT_WIN_START_FREQ     0x00C80000L     Short window start frequency       IwSHORT_WIN_END_FREQ     0x01A40000     Short window end frequency       IwSINGLE_WIN_START_FREQ     0x00640000L     Single window start frequency       IwSINGLE_WIN_END_FREQ     0x00D20000L     Single window start frequency       IwDOUBLE_WIN_START_FREQ     0x00340000     Double window start frequency       IwDOUBLE_WIN_END_FREQ     0x00780000L     Double window end frequency       MAX_INDA_ON_SPECTRUM     70     Maximum number of local maxima on the spectrum       MAX_PEAKS_FOR_SORT     30     Maximum number peaks for sorting       MAX_PEAKS_PRELIM     7     Maximum number of peaks (preliminary)       MIN_PEAKS     7     Minimum number of peaks (final)       MAX_PEAKS_FINAL     20     Maximum number of peaks (final)       MAX_PRELIM_CANDS     4     Maximum number of peaks (final)       MAX_PRELIM_CANDS     4     Maximum number of peaks (final)       CREATE_PIECEWISE_FUNC_LOOP_LIM_SH     20     Create Piecewise function loop limit for short window       CREATE_PIECEWISE_FUNC_LOOP_LIM_SNG     30     Create Piecewise function loop limit for single window  |                                       |             |  |
| IwSHORT_WIN_END_FREQ     0x01A40000     Short window end frequency       IwSINGLE_WIN_START_FREQ     0x00640000L     Single window start frequency       IwSINGLE_WIN_END_FREQ     0x00D20000L     Single window end frequency       IwDOUBLE_WIN_START_FREQ     0x00340000     Double window start frequency       IwDOUBLE_WIN_END_FREQ     0x00780000L     Double window end frequency       MAX_LOCAL_MAXIMA_ON_SPECTRUM     70     Maximum number of local maxima on the spectrum       MAX_PEAKS_FOR_SORT     30     Maximum number peaks for sorting       MAX_PEAKS_PRELIM     7     Maximum number of peaks (preliminary)       MIN_PEAKS     7     Minimum number of peaks (final)       MAX_PEAKS_FINAL     20     Maximum number of peaks (final)       MAX_PRELIM_CANDS     4     Maximum number of peaks (final)       CREATE_PIECEWISE_FUNC_LOOP_LIM_SH     20     Create Piecewise function loop limit for short window       CREATE_PIECEWISE_FUNC_LOOP_LIM_SNG     30     Create Piecewise function loop limit for single window   |                                       |             |  |
| IwSINGLE_WIN_START_FREQ     0x00640000L     Single window start frequency       IwSINGLE_WIN_END_FREQ     0x00D20000L     Single window end frequency       IwDOUBLE_WIN_START_FREQ     0x00340000     Double window start frequency       IwDOUBLE_WIN_END_FREQ     0x00780000L     Double window end frequency       MAX_LOCAL_MAXIMA_ON_SPECTRUM     70     Maximum number of local maxima on the spectrum       MAX_PEAKS_FOR_SORT     30     Maximum number peaks for sorting       MAX_PEAKS_PRELIM     7     Maximum number of peaks (preliminary)       MIN_PEAKS     7     Minimum number of peaks       MAX_PEAKS_FINAL     20     Maximum number of peaks (final)       MAX_PRELIM_CANDS     4     Maximum number of preliminary candidates (pitch)       CREATE_PIECEWISE_FUNC_LOOP_LIM_SH     20     Create Piecewise function loop limit for short window       CREATE_PIECEWISE_FUNC_LOOP_LIM_SNG     30     Create Piecewise function loop limit for single window   |                                       |             |  |
| IwSINGLE_WIN_END_FREQ     0x00D20000L     Single window end frequency       IwDOUBLE_WIN_START_FREQ     0x00340000     Double window start frequency       IwDOUBLE_WIN_END_FREQ     0x00780000L     Double window end frequency       MAX_LOCAL_MAXIMA_ON_SPECTRUM     70     Maximum number of local maxima on the spectrum       MAX_PEAKS_FOR_SORT     30     Maximum number of peaks for sorting       MAX_PEAKS_PRELIM     7     Maximum number of peaks (preliminary)       MIN_PEAKS     7     Minimum number of peaks       MAX_PEAKS_FINAL     20     Maximum number of peaks (final)       MAX_PRELIM_CANDS     4     Maximum number of preliminary candidates (pitch)       CREATE_PIECEWISE_FUNC_LOOP_LIM_SH     20     Create Piecewise function loop limit for short window       CREATE_PIECEWISE_FUNC_LOOP_LIM_SNG     30     Create Piecewise function loop limit for single window  |                                       |             |  |
| IwDOUBLE_WIN_START_FREQ     0x00340000     Double window start frequency       IwDOUBLE_WIN_END_FREQ     0x00780000L     Double window end frequency       MAX_LOCAL_MAXIMA_ON_SPECTRUM     70     Maximum number of local maxima on the spectrum       MAX_PEAKS_FOR_SORT     30     Maximum number peaks for sorting       MAX_PEAKS_PRELIM     7     Maximum number of peaks (preliminary)       MIN_PEAKS     7     Minimum number of peaks       MAX_PEAKS_FINAL     20     Maximum number of peaks (final)       MAX_PRELIM_CANDS     4     Maximum number of preliminary candidates (pitch)       CREATE_PIECEWISE_FUNC_LOOP_LIM_SH     20     Create Piecewise function loop limit for short window       CREATE_PIECEWISE_FUNC_LOOP_LIM_SNG     30     Create Piecewise function loop limit for single window   | IwSINGLE_WIN_END_FREQ                 |             |  |
| MAX_LOCAL_MAXIMA_ON_SPECTRUM 70 Maximum number of local maxima on the spectrum MAX_PEAKS_FOR_SORT 30 Maximum number peaks for sorting MAX_PEAKS_PRELIM 7 Maximum number of peaks (preliminary) MIN_PEAKS 8 7 Minimum number of peaks MAX_PEAKS_FINAL 9 Maximum number of peaks (final) MAY_PELIM_CANDS 4 Maximum number of preliminary candidates (pitch) CREATE_PIECEWISE_FUNC_LOOP_LIM_SH 20 Create Piecewise function loop limit for short window CREATE_PIECEWISE_FUNC_LOOP_LIM_SNG 30 Create Piecewise function loop limit for single window  | IwDOUBLE_WIN_START_FREQ               | 0x00340000  | Double window start frequency                                      |
| MAX_PEAKS_FOR_SORT  30 Maximum number peaks for sorting  MAX_PEAKS_PRELIM  7 Maximum number of peaks (preliminary)  MIN_PEAKS  7 Minimum number of peaks  MAX_PEAKS_FINAL  MAX_PEAKS_FINAL  MAX_PEAKS_FINAL  MAX_PEAKS_FINAL  MAX_PRELIM_CANDS  4 Maximum number of preliminary candidates (pitch)  CREATE_PIECEWISE_FUNC_LOOP_LIM_SH  CREATE_PIECEWISE_FUNC_LOOP_LIM_SNG  30 Create Piecewise function loop limit for short window  CREATE_PIECEWISE_FUNC_LOOP_LIM_SNG  30 Create Piecewise function loop limit for single window   | IwDOUBLE_WIN_END_FREQ                 |             | Double window end frequency  |
| MAX_PEAKS_PRELIM 7 Maximum number of peaks (preliminary) MIN_PEAKS 7 Minimum number of peaks MAX_PEAKS_FINAL Maximum number of peaks (final) MAX_PEAKS_FINAL 4 Maximum number of preliminary candidates (pitch) CREATE_PIECEWISE_FUNC_LOOP_LIM_SH 20 Create Piecewise function loop limit for short window CREATE_PIECEWISE_FUNC_LOOP_LIM_SNG 30 Create Piecewise function loop limit for single window  |                                       |             |  |
| MIN_PEAKS 7 Minimum number of peaks MAX_PEAKS_FINAL 20 Maximum number of peaks (final) MAX_PRELIM_CANDS 4 Maximum number of preliminary candidates (pitch) CREATE_PIECEWISE_FUNC_LOOP_LIM_SH 20 Create Piecewise function loop limit for short window CREATE_PIECEWISE_FUNC_LOOP_LIM_SNG 30 Create Piecewise function loop limit for single window   |                                       |             |  |
| MAX_PEAKS_FINAL  20 Maximum number of peaks (final)  MAX_PRELIM_CANDS  4 Maximum number of preliminary candidates (pitch)  CREATE_PIECEWISE_FUNC_LOOP_LIM_SH  CREATE_PIECEWISE_FUNC_LOOP_LIM_SNG  30 Create Piecewise function loop limit for single window  CREATE_PIECEWISE_FUNC_LOOP_LIM_SNG  |                                       |             |  |
| MAX_PRELIM_CANDS  4 Maximum number of preliminary candidates (pitch)  CREATE_PIECEWISE_FUNC_LOOP_LIM_SH  20 Create Piecewise function loop limit for short window  CREATE_PIECEWISE_FUNC_LOOP_LIM_SNG  30 Create Piecewise function loop limit for single window   |                                       |             |  |
| CREATE_PIECEWISE_FUNC_LOOP_LIM_SH 20 Create Piecewise function loop limit for short window CREATE_PIECEWISE_FUNC_LOOP_LIM_SNG 30 Create Piecewise function loop limit for single window  |                                       |             |  |
| CREATE_PIECEWISE_FUNC_LOOP_LIM_SNG 30 Create Piecewise function loop limit for single window   |                                       |             |  |
|  |                                       |             |  |
|  | CREATE PIECEWISE FUNC LOOP LIM SNG    |             |  |

| swSUM_FRACTION          | 0x799A | Sum fraction   |
|-------------------------|--------|--|
| swAMP_FRACTION          | 0x33F8 | Amplitude fraction   |
| MAX_BEST_CANDS          | 2      | Maximum number of best candidates (pitch)  |
| N_OF_BEST_CANDS_SHORT   | 2      | Number of best candidates for short window   |
| N_OF_BEST_CANDS_SINGLE  | 2      | Number of best candidates for single window  |
| N_OF_BEST_CANDS_DOUBLE  | 2      | Number of best candidates for double window  |
| N_OF_BEST_CANDS         | 6      | Number of best candidates for all windows  |
| SIZE_SCRATCH_DOPITCH    | 1090   | Scratch memory size for DoPitch() function (This is the actual size required. The declared size in C simulation is 1632) |
| SIZE_SCRATCH_ADVPROCESS | 825    | Scratch memory size for DoAdvProcess() function (This is the actual size required.                                       |
|                         |        | The declared size in C simulation is 1100)   |
| RVC_PITCH_ROM_SIG       | 11031  | Signature for RVC_PITCH_ROM structure  |
| RVC_PITCH_METER_SIG     | 21053  | Signature for RVC_PITCH_METER structure  |

# 4.5.2 Description of fixed tables used in the C-code

This section contains a listing of all fixed tables sorted by source file name and table name. All table data is declared as **Word16**.

Table 6a: Fixed tables for AFE

| File                | Table Name         | Length | Description   |
|---------------------|--------------------|--------|---|
| 16kHzProcessing_B.c | table_pow2         | 33     | Table for square root   |
|                     | LambdaNSEx2        | 100    | Table used to compute first 100 LambdaNSE                       |
|                     | dp02_h             | 59     | MSB of QMF filter coefficients                                  |
|                     | dp02_l             | 43     | LSB of QMF filter coefficients                                  |
| PostProc_B.c        | targetLMS16        | 12     | Target for blind equalization                                   |
| ComCeps_B.c         | HalfHamming16      | 100    | Hamming window coefficients                                     |
|                     | CosMatrix16        | 144    | Inverse cosinus coefficients at 8Khz (not used at 16khz)        |
|                     | CosMatrix16_16khz  | 156    | Inverse cosinus coefficients at 16Khz                           |
|                     | pondMelFilter      | 309    | Mel bank coefficients   |
| ff4nrFix16_B.c      | tabSin             | 64     | Sine table  |
|                     | tabCos             | 64     | Cosine table  |
| MathFunc.c          | tbInt0             | 48     | Coefficients for computation of square root                     |
| ExtNoiseSup_B.c     | lambda_1divX       | 20     | Computation of 1/N  |
|                     | Hann_sh32_hi       | 100    | MSB of hanning window coefficients (32 bits)                    |
|                     | Hann_sh32_lo       | 100    | LSB of hanning window coefficients (32 bits)                    |
|                     | Hann_sh24_hi       | 100    | MSB of hanning window coefficients (24 bits)                    |
|                     | Hann_sh24_lo       | 100    | LSB of hanning window coefficients (24 bits)                    |
|                     | pondMelFilterNoise | 157    | Mel-frequency scale coefficients (applied to the Wiener filter) |
|                     | idctMel16          | 234    | Mel-warped inverse DCT coefficients                             |
|                     | pondMelFilter16k   | 134    | Filter bank coefficients at 16Khz                               |
|                     | M1_LamdaLTE        | 8      | Computation of 1/N  |
|                     | M1_LambdaNSEx2     | 100    | Computation of 2/N  |
|                     | M1_LamdaNSE        | 9      | Computation of 1/N  |
|                     | mInvLambda16       | 10     | Comutation od 2/N   |

Table 6b: Fixed tables for VQ

| File        | Table Name               | Length | Description                            |
|-------------|--------------------------|--------|--|
| coder_VAD.c | quantizer16kHz_0_1       | 128    | vq table                               |
| _           | quantizer16kHz_2_3       | 128    | vg table                               |
|             | quantizer16kHz_4_5       | 128    | vg table                               |
|             | quantizer16kHz_6_7       | 128    | vq table                               |
|             | quantizer16kHz_8_9       | 128    | vq table                               |
|             | quantizer16kHz_10_11     | 64     | vq table                               |
|             | quantizer16kHz_12_13     | 512    | vq table                               |
|             | quantizer8kHz_0_1        | 128    | vq table                               |
|             | quantizer8kHz_2_3        | 128    | vq table                               |
|             | quantizer8kHz_4_5        | 128    | vq table                               |
|             | quantizer8kHz_6_7        | 128    | vq table                               |
|             | quantizer8kHz_8_9        | 128    | vq table                               |
|             | quantizer8kHz_10_11      | 64     | vq table                               |
|             | quantizer8kHz_12_13      | 512    | vq table                               |
|             | weight16kHz_c0_shift     | 1      | vq weights                             |
|             | weight16kHz_c0_norm      | 1      | vq weights                             |
|             | weight16kHz_logE         | 1      | vq weights                             |
|             | weight8kHz_c0_shift      | 1      | vq weights                             |
|             | weight8kHz_c0_norm       | 1      | vq weights                             |
|             | weight8kHz_logE          | 1      | vq weights                             |
|             | plwQuantLevels[127]      | 127*2  | vq tables for pitch/class quantization |
|             | ppplwQuantSections[8][3] | 24*2   | vq tables for pitch/class quantization |
|             | plwQuantLevels[31]       | 31*2   | vq tables for pitch/class quantization |
|             | pplwQuantSections[4][3]  | 12*2   | vq tables for pitch/class quantization |
|             | pswRatioThId_1[4][6]     | 24     | vq tables for pitch/class quantization |
|             | piMultiLevelIndex[4]     | 4      | vq tables for pitch/class quantization |
|             | pswRatioThld_2[4][8]     | 32     | vq tables for pitch/class quantization |
|             | piMultiLevelIndex_2[4]   | 4      | vq tables for pitch/class quantization |
|             | swAlpha1                 | 1      | pitch/class constants                  |
|             | swAlpha2                 | 1      | pitch/class constants                  |

**Table 6c: Fixed Tables for Extension** 

| File               | Table name              | Length | Description   |
|--------------------|-------------------------|--------|---|
| ExtNoiseSup_B.c    | pswPePower              | 129    | Coefficients to compute the pre-emphasis power spectrum       |
| preProc_B.c        | pswHpfCoef              | 15     | High pass filter coefficients                                 |
| preProc_B.c        | pswLpfCoef              | 15     | Low pass filter coefficients                                  |
| preProc_B.c        | pswLfeCoef              | 3      | Low frequency emphasis filter coefficients                    |
| dsrAfeVad_B.c      | piBurstConst            | 20     | Burst length constants for different SNR's                    |
| dsrAfeVad_B.c      | piHangConst             | 20     | Hang length constants for different SNR's                     |
| dsrAfeVad_B.c      | piVADThld               | 20     | VAD voice metric thresholds for different SNR's               |
| dsrAfeVad_B.c      | piVMTable               | 90     | Voice metric table as a function of SNR index                 |
| dsrAfeVad_B.c      | piSigThld               | 20     | Signal threshold table as a function of SNR                   |
| dsrAfeVad_B.c      | piUpdateThld            | 20     | Update threshold table as a function of SNR                   |
| dsrAfeVad_B.c      | pswShapeTable           | 23     | Spectral shape correction table                               |
| fix_mathlib.c      | coeff_sqrt5_58          | 5      | Coefficients for computation of square root                   |
| fix_mathlib.c      | coeff_sqrt5_78          | 5      | Coefficients for computation of square root                   |
| rvc_pitch_init_B.h | ROM_astFrac             | 312    | Fractions table   |
| rvc_pitch_init_B.h | ROM_pstWindowshiftTable | 514    | Complex exponents table for time shifting in frequency domain |
| rvc_pitch_init_B.h | ROM_aswDirichletImag    | 8      | Imaginary part of the Dirichlet kernel                        |

## 4.5.3 Static variables used in the C-code

In this section two tables that specify the static variables for the AFE, VQ, and Extension respectively are shown.

Table 7a: AFE static variables

| Struct Name        | Variable   | Type[Length]   | Description   |
|--------------------|--|--|---|
| QMF FIR            | Variable   | Type[Length]   | Description   |
| QIVII _I IIV       | lengthQMF  | Word32   | QMF Filter length   |
|                    | *dp_l  | Word16   | QMF filter low frequency Coeff  |
|                    | *dp_h  | Word16   | QMF filter high frequency Coeff   |
|                    | *T   | Word16   | Temporary QMF filter buffer   |
| DataFor16kProc_B   | T_dec  | Word16   | Multiplier for T  |
| Datal OFTOKPTOC_B  | FrameLength  | Word32   | Input Frame length  |
|                    | FrameShift   | Word32   | Shift value for the frame   |
|                    | numFramesInBuffer  | Word32   | Number of frames in buffer  |
|                    | SamplingFrequency  | Word32   | Sampling frequency (8/16)   |
|                    | Do16kHzProc  | BOOLEAN  | Flag to enable 16kHz processing   |
|                    | *hpBands_B<br>hpBandsSize  | Word32<br>Word32   | Buffer for HP bands<br>hpBands B buffer size  |
|                    | CodeForBands16k B  | Word32[9]  | HP coding buffer  |
|                    | bufferCodeForBands16k_B  | Word32[27]   | buffer used for HP coding   |
|                    | codeWeights_B  | Word16[3]  | code Weights buffer   |
|                    | bufferCodeWeights_B  | Word16[9]  | buffer used for code Weights  |
|                    | * pQMF_Fir   | QMF_FIR  | Pointer to QMF_FIR structure  |
|                    | *bufferData16k_B<br>bufData16kSize   | Word32<br>Word32   | temporary buffer to carry QMF LP data 16k data buffer size  |
|                    | *FirstWindow16k  | MeIFB_Window   | pointer to MeIFB_Window structure   |
|                    | noiseSE16k B   | Word32[3]  | noise spectrul energy variable  |
|                    | noise_dec  | Word16   | Multiplier for noiseSE16k_B   |
|                    | BandsForCoding16k_B  | Word32[9]  | buffer for storing Bands for Coding   |
|                    | vadCounter16k  | Word32   | vad flag counter  |
|                    | vad16k   | Word32   | vad flag  |
|                    | nbSpeechFrames16k<br>hangOver16k   | Word32<br>Word32   | number of speech frames counter<br>hang over used for VAD   |
|                    | meanEn16k  | Word32   | mean Energy variable  |
|                    | nb_frame_threshold_nse   | Word32   | threshold NSE for frame   |
|                    | lambda_nse   | Word16   | lambda NSE variable   |
|                    | *dataHP_B  | Word32   | buffer stores QMF HP value  |
|                    | dec_16k  | Word16[5]  | Multiplier for dataHP_B buffer  |
|                    | BFC_dec<br>fb16k dec   | Word16[1]<br>Word16[3]   | Multiplier for computing bands for coding  Buffer is used to store multiplier for current and pervious two frames   |
| PostProcStructX    | IDTOK_dec  | Word ro[3]   | buller is used to store multiplier for current and pervious two frames  |
| 7 0011 100011 0011 | weightLMS  | Word32[12]   | Current LMS weight  |
| CompCepsStructX    |  |  | · ·   |
|                    | FFTLength  | Word32   | FFT size  |
|                    | Do16khzProc  | Word16   | Flag to enable 16kHz processing   |
| WaveProcStructX    | *pData16k  | Word32   | Pointer to data for 16Khz processing  |
| WaveFlocSiluciA    | *TeagerFilter16  | Word32   | Pointer to teager filter  |
|                    | *TeagerWindow32  | Word32   | Pointer to teager window  |
|                    | TeagerOnset  | Word32   | Unused  |
|                    | FrameLength  | Word32   | Input frame length  |
| ns_var_F           |  | 14.0   | 0 1 (2/42)  |
|                    | SampFreq<br>Do16khzProc  | Word16<br>Word16   | Sampling frequency (8/16) Flag to enable 16kHz processing   |
|                    | buffers.nbFramesInFirstStage   | Word32   | number of frames in first stage   |
|                    | buffers.nbFramesInFirstStage   | Word32   | number of frames in second stage  |
|                    | buffers. nbFramesOutSecondStage  | Word32   | number of frames out og second stage  |
|                    | buffers. FirstStageIn16Buffer  | Word16[180]  | First stage buffer  |
|                    | buffers.SecondStageInBuffer32  | Word32[180]  | Second stage buffer   |
| -                  | buffers. SecondDecalSig<br>prevSamples32.lastSampleIn32  | Word16[4]<br>Word32  | Shift factor for each sub-frame of second stage buffer Last input sample of DC offset compensation  |
|                    | prevSamples32.lastSamplem32<br>prevSamples32.lastDCOut32   | Word32   | last output sample of DC offset compensation  |
|                    | prevSamples32. oldShift  | Word16   | Iprevious window shift factor of DC offset compensation   |
|                    | spectrum.indexBuffer1  | Word16   | Where to enter new PSD for first stage, alternatively 0 and 1   |
|                    | spectrum.indexBuffer2  | Word16   | Where to enter new PSD for second stage, alternatively 0 and 1  |
|                    | spectrum.noiseSE1_32   | Word32[65]   | Noise spectrum estimate for first stage   |
|                    | spectrum.noiseSE1_dec  | Word16[65]   | Shift factor for Noise spectrum estimate (first sage)   |
|                    | 1.   |  | Noise apportrum, actimate for second store  |
|                    | spectrum.noiseSE2_32   | Word32[65]   | Noise spectrum estimate for second stage Shift factor for Noise spectrum estimate (second sage)   |
|                    | spectrum.noiseSE2_32<br>spectrum.noiseSE2_dec  | Word32[65]<br>Word16[65]   | Shift factor for Noise spectrum estimate (second sage)  |
|                    | spectrum.noiseSE2_32   | Word32[65]   |   |
|                    | spectrum.noiseSE2_32 spectrum.noiseSE2_dec spectrum.PSDMeanAntBuffer1 spectrum.nSigSE1Ant_dec spectrum.PSDMeanAntBuffer2   | Word32[65]<br>Word16[65]<br>Word32[65]<br>Word16[65]<br>Word32[65]   | Shift factor for Noise spectrum estimate (second sage)  1st stage PSD Mean buffer for precedent frame Shift factor for PSD Mean buffer for precedent frame (1rst stage) 2nd stage PSD Mean bufferfor precedent frame  |
|                    | spectrum.noiseSE2_32 spectrum.noiseSE2_dec spectrum.PSDMeanAntBuffer1 spectrum.nSigSE1Ant_dec spectrum.PSDMeanAntBuffer2 spectrum.nSigSE2Ant_dec   | Word32[65] Word16[65] Word32[65] Word16[65] Word32[65] Word16[65]  | Shift factor for Noise spectrum estimate (second sage)  1st stage PSD Mean buffer for precedent frame Shift factor for PSD Mean buffer for precedent frame (1rst stage) 2nd stage PSD Mean bufferfor precedent frame Shift factor for PSD Mean buffer for precedent frame (2nd stage)   |
|                    | spectrum.noiseSE2_32 spectrum.noiseSE2_dec spectrum.PSDMeanAntBuffer1 spectrum.PSDMeanAntBuffer2 spectrum.PSDMeanAntBuffer2 spectrum.nSigSE2Ant_dec spectrum.denSigSE21_32   | Word32[65]<br>Word16[65]<br>Word32[65]<br>Word32[65]<br>Word3[65]<br>Word16[65]<br>Word3[65]   | Shift factor for Noise spectrum estimate (second sage)  1st stage PSD Mean buffer for precedent frame  Shift factor for PSD Mean buffer for precedent frame (1rst stage)  2nd stage PSD Mean bufferfor precedent frame  Shift factor for PSD Mean buffer for precedent frame (2nd stage)  1st stage PSD Mean buffer   |
|                    | spectrum.noiseSE2_32 spectrum.noiseSE2_dec spectrum.PSDMeanAntBuffer1 spectrum.nSigSE1Ant_dec spectrum.PSDMeanAntBuffer2 spectrum.nSigSE2Ant_dec spectrum.denSigSE1_32 spectrum.nSigSE1Cur_dec   | Word32[65] Word16[65] Word32[65] Word32[65] Word32[65] Word16[65] Word16[65] Word16[65] Word16[65]   | Shift factor for Noise spectrum estimate (second sage)  1st stage PSD Mean buffer for precedent frame  Shift factor for PSD Mean buffer for precedent frame (1rst stage)  2nd stage PSD Mean bufferfor precedent frame  Shift factor for PSD Mean buffer for precedent frame (2nd stage)  1st stage PSD Mean buffer  Shift factor for PSD Mean buffer (1rst stage)  |
|                    | spectrum.noiseSE2_32 spectrum.noiseSE2_dec spectrum.PSDMeanAntBuffer1 spectrum.nSigSE1Ant_dec spectrum.PSDMeanAntBuffer2 spectrum.nSigSE2Ant_dec spectrum.denSigSE1_32 spectrum.nSigSE1Cur_dec spectrum.denSigSE1_32   | Word32[65] Word16[65] Word32[65] Word16[65] Word32[65] Word16[65] Word32[65] Word32[65] Word32[65] Word32[65]  | Shift factor for Noise spectrum estimate (second sage)  1st stage PSD Mean buffer for precedent frame Shift factor for PSD Mean buffer for precedent frame (1rst stage)  2nd stage PSD Mean bufferfor precedent frame Shift factor for PSD Mean buffer for precedent frame Shift factor for PSD Mean buffer Shift factor for PSD Mean buffer Shift factor for PSD Mean buffer (1rst stage) 2nd stage PSD Mean buffer  |
|                    | spectrum.noiseSE2_32 spectrum.noiseSE2_dec spectrum.PSDMeanAntBuffer1 spectrum.nSigSE1Ant_dec spectrum.PSDMeanAntBuffer2 spectrum.nSigSE2Ant_dec spectrum.denSigSE1_32 spectrum.nSigSE1Cur_dec   | Word32[65] Word16[65] Word32[65] Word32[65] Word32[65] Word16[65] Word16[65] Word16[65] Word16[65]   | Shift factor for Noise spectrum estimate (second sage)  1st stage PSD Mean buffer for precedent frame  Shift factor for PSD Mean buffer for precedent frame (1rst stage)  2nd stage PSD Mean bufferfor precedent frame  Shift factor for PSD Mean buffer for precedent frame (2nd stage)  1st stage PSD Mean buffer  Shift factor for PSD Mean buffer (1rst stage)  |
|                    | spectrum.noiseSE2_32 spectrum.noiseSE2_dec spectrum.PSDMeanAntBuffer1 spectrum.PSDMeanAntBuffer2 spectrum.PSDMeanAntBuffer2 spectrum.nSigSE2Ant_dec spectrum.denSigSE1_32 spectrum.nSigSE1Cur_dec spectrum. denSigSE2_32 spectrum. nSigSE2Cur_dec vad_data_ns_F. nbFrame vad_data_ns_F. flagVAD  | Word32[65] Word16[65] Word16[65] Word16[65] Word32[65] Word16[65] Word32[65] Word32[65] Word32[65] Word16[65] Word16[65] Word16[65] Word16[65] Word16[65] Word16[65] Word16[65]  | Shift factor for Noise spectrum estimate (second sage)  1st stage PSD Mean buffer for precedent frame Shift factor for PSD Mean buffer for precedent frame (1rst stage) 2nd stage PSD Mean bufferfor precedent frame Shift factor for PSD Mean buffer for precedent frame (2nd stage) 1st stage PSD Mean buffer Shift factor for PSD Mean buffer (1rst stage) 2nd stage PSD Mean buffer Shift factor for PSD Mean buffer Shift factor for PSD Mean buffer (2 <sup>nd</sup> stage)   |
|                    | spectrum.noiseSE2_32 spectrum.noiseSE2_dec spectrum.PSDMeanAntBuffer1 spectrum.PSDMeanAntBuffer2 spectrum.PSDMeanAntBuffer2 spectrum.nSigSE2Ant_dec spectrum.denSigSE1_32 spectrum.nSigSE1Cur_dec spectrum.denSigSE2_32 spectrum.nSigSE2Cur_dec vad_data_ns_F.nbFrame vad_data_ns_F.flagVAD vad_data_ns_F.nagOver  | Word32[65] Word16[65] Word32[65] Word32[65] Word32[65] Word32[65] Word32[65] Word16[65] Word16[65] Word16[65] Word16[65] Word16[65] Word16[65] Word16  | Shift factor for Noise spectrum estimate (second sage)  1st stage PSD Mean buffer for precedent frame  Shift factor for PSD Mean buffer for precedent frame (1rst stage)  2nd stage PSD Mean bufferfor precedent frame  Shift factor for PSD Mean buffer for precedent frame (2nd stage)  1st stage PSD Mean buffer  Shift factor for PSD Mean buffer (1rst stage)  2nd stage PSD Mean buffer  Shift factor for PSD Mean buffer (2 <sup>nd</sup> stage)  Nubmer of frames (for the 2 stages)  Vad Flag (1 = SPEECH, 0 = NON SPEECH)  hangover   |
|                    | spectrum.noiseSE2_32 spectrum.noiseSE2_dec spectrum.PSDMeanAntBuffer1 spectrum.PSDMeanAntBuffer2 spectrum.PSDMeanAntBuffer2 spectrum.nSigSE1Ant_dec spectrum.denSigSE2Ant_dec spectrum.denSigSE1_32 spectrum.nSigSE1Cur_dec spectrum.nSigSE1Cur_dec spectrum.nSigSE2Cur_dec vad_data_ns_F.nbFrame vad_data_ns_F.flagVAD vad_data_ns_F.nbSpeechFrames   | Word32[65] Word16[65] Word32[65] Word32[65] Word32[65] Word32[65] Word32[65] Word16[65] Word32[65] Word16[65] Word16[65] Word16[65] Word16[65] Word16[65] Word16[02] Word16  | Shift factor for Noise spectrum estimate (second sage)  1st stage PSD Mean buffer for precedent frame  Shift factor for PSD Mean buffer for precedent frame (1rst stage)  2nd stage PSD Mean buffer for precedent frame  Shift factor for PSD Mean buffer for precedent frame (2nd stage)  1st stage PSD Mean buffer  Shift factor for PSD Mean buffer (1rst stage)  2nd stage PSD Mean buffer  Shift factor for PSD Mean buffer (3nd stage)  Number of frames (for the 2 stages)  Vad Flag (1 = SPEECH, 0 = NON SPEECH)  hangover  Number of speech frames (used to set hangover)  |
|                    | spectrum.noiseSE2_32 spectrum.noiseSE2_dec spectrum.PSDMeanAntBuffer1 spectrum.PSDMeanAntBuffer2 spectrum.nSigSE1Ant_dec spectrum.denSigSE2Ant_dec spectrum.denSigSE1_32 spectrum.nSigSE1Cur_dec spectrum. denSigSE2_32 spectrum.nSigSE2Cur_dec vad_data_ns_F.nbFrame vad_data_ns_F.nbFrame vad_data_ns_F.nbSpeechFrames vad_data_ns_F.nbSpeechFrames vad_data_ns_F.meanEn32   | Word32[65] Word16[65] Word16[65] Word32[65] Word32[65] Word32[65] Word32[65] Word16[65] Word32[65] Word16[65] Word16[65] Word16[65] Word16[65] Word16[65] Word16[65] Word16[65] Word16[65] Word16 Word16 Word16 Word16 | Shift factor for Noise spectrum estimate (second sage)  1st stage PSD Mean buffer for precedent frame Shift factor for PSD Mean buffer for precedent frame (1rst stage) 2nd stage PSD Mean bufferfor precedent frame Shift factor for PSD Mean buffer for precedent frame (2nd stage) 1st stage PSD Mean buffer Shift factor for PSD Mean buffer (1rst stage) 2nd stage PSD Mean buffer Shift factor for PSD Mean buffer (2nd stage) Nubmer of frames (for the 2 stages) Vad Flag (1 = SPEECH, 0 = NON SPEECH) hangover Number of speech frames (used to set hangover) Mean energy for VAD  |
|                    | spectrum.noiseSE2_32 spectrum.noiseSE2_dec spectrum.PSDMeanAntBuffer1 spectrum.PSDMeanAntBuffer2 spectrum.PSDMeanAntBuffer2 spectrum.nSigSE2Ant_dec spectrum.denSigSE1_32 spectrum.nSigSE1Cur_dec spectrum.denSigSE2_32 spectrum. denSigSE2_0r_dec vad_data_ns_F.nbFrame vad_data_ns_F.nbFrame vad_data_ns_F.nbSpeechFrames vad_data_ns_F.nbSpeechFrames vad_data_ns_F.msDFopechFrames vad_data_ns_F.meanEn32 vad_data_ca.flagVAD  | Word32[65] Word16[65] Word16[65] Word32[65] Word32[65] Word32[65] Word32[65] Word32[65] Word32[65] Word32[65] Word16[65] Word16[65] Word16[65] Word16[65] Word16 Word16 Word16 Word16 Word16 Word16 Word16 Word16      | Shift factor for Noise spectrum estimate (second sage)  1st stage PSD Mean buffer for precedent frame Shift factor for PSD Mean buffer for precedent frame (1rst stage) 2nd stage PSD Mean bufferfor precedent frame Shift factor for PSD Mean buffer for precedent frame Shift factor for PSD Mean buffer for precedent frame (2nd stage) 1st stage PSD Mean buffer Shift factor for PSD Mean buffer (1rst stage) 2nd stage PSD Mean buffer Shift factor for PSD Mean buffer (2 <sup>nd</sup> stage) Nubmer of frames (for the 2 stages) Vad Flag (1 = SPEECH, 0 = NON SPEECH) hangover Number of speech frames (used to set hangover) Mean energy for VAD Vad Flag (1 = SPEECH, 0 = NON SPEECH)                   |
|                    | spectrum.noiseSE2_32 spectrum.noiseSE2_dec spectrum.PSDMeanAntBuffer1 spectrum.PSDMeanAntBuffer2 spectrum.nSigSE1Ant_dec spectrum.denSigSE2Ant_dec spectrum.denSigSE1_32 spectrum.nSigSE1Cur_dec spectrum. denSigSE2_32 spectrum.nSigSE2Cur_dec vad_data_ns_F.nbFrame vad_data_ns_F.nbFrame vad_data_ns_F.nbSpeechFrames vad_data_ns_F.nbSpeechFrames vad_data_ns_F.meanEn32   | Word32[65] Word16[65] Word16[65] Word32[65] Word32[65] Word32[65] Word32[65] Word16[65] Word32[65] Word16[65] Word16[65] Word16[65] Word16[65] Word16[65] Word16[65] Word16[65] Word16[65] Word16 Word16 Word16 Word16 | Shift factor for Noise spectrum estimate (second sage)  1st stage PSD Mean buffer for precedent frame Shift factor for PSD Mean buffer for precedent frame (1rst stage) 2nd stage PSD Mean bufferfor precedent frame Shift factor for PSD Mean buffer for precedent frame (2nd stage) 1st stage PSD Mean buffer Shift factor for PSD Mean buffer (1rst stage) 2nd stage PSD Mean buffer Shift factor for PSD Mean buffer (2nd stage) Nubmer of frames (for the 2 stages) Vad Flag (1 = SPEECH, 0 = NON SPEECH) hangover Number of speech frames (used to set hangover) Mean energy for VAD  |
|                    | spectrum.noiseSE2_32 spectrum.noiseSE2_dec spectrum.PSDMeanAntBuffer1 spectrum.PSDMeanAntBuffer2 spectrum.PSDMeanAntBuffer2 spectrum.denSigSE2Ant_dec spectrum.denSigSE1_32 spectrum.nSigSE1Cur_dec spectrum.denSigSE2_32 spectrum.nSigSE2Cur_dec vad_data_ns_F.nbFrame vad_data_ns_F.nbFrame vad_data_ns_F.nangOver vad_data_ns_F.nangOver vad_data_ns_F.nangOver vad_data_ca.flagVAD vad_data_ca.flagVAD vad_data_ca.flangOver vad_data_ca.flangOver vad_data_ca.flangOver vad_data_ca.flangOver vad_data_ca.flangOver vad_data_ca.nbSpeechFrames vad_data_ca.nbSpeechFrames | Word32[65] Word16[65] Word16[65] Word16[65] Word32[65] Word32[65] Word32[65] Word32[65] Word32[65] Word16[65] Word16[65] Word16[65] Word16[65] Word16[65] Word16 Word16 Word16 Word16 Word16 Word16 Word16 Word16      | Shift factor for Noise spectrum estimate (second sage)  1st stage PSD Mean buffer for precedent frame Shift factor for PSD Mean buffer for precedent frame (1rst stage) 2nd stage PSD Mean bufferfor precedent frame Shift factor for PSD Mean buffer for precedent frame (2nd stage) 1st stage PSD Mean buffer (2nd stage) 1st stage PSD Mean buffer Shift factor for PSD Mean buffer (1rst stage) 2nd stage PSD Mean buffer Shift factor for PSD Mean buffer (2nd stage) Nubmer of frames (for the 2 stages) Vad Flag (1 = SPEECH, 0 = NON SPEECH) hangover Number of speech frames (used to set hangover) Mean energy for VAD Vad Flag (1 = SPEECH, 0 = NON SPEECH) hangover                                     |
|                    | spectrum.noiseSE2_32 spectrum.noiseSE2_dec spectrum.PSDMeanAntBuffer1 spectrum.nSigSE1Ant_dec spectrum.PSDMeanAntBuffer2 spectrum.nSigSE2Ant_dec spectrum.nSigSE2Ant_dec spectrum.nSigSE21_32 spectrum.nSigSE1Cur_dec spectrum.nSigSE1Cur_dec spectrum.nSigSE2Cur_dec vad_data_ns_F.nbFrame vad_data_ns_F.nbFrame vad_data_ns_F.nbSpeechFrames vad_data_ns_F.nbSpeechFrames vad_data_ca.flagVAD vad_data_ca.flagVAD vad_data_ca.nbSpeechFrames   | Word32[65] Word16[65] Word32[65] Word32[65] Word32[65] Word3[65] Word3[65] Word3[65] Word16[65] Word16[65] Word16[65] Word16[65] Word16[65] Word16[2] Word16 Word16 Word16 Word16 Word16 Word16 Word16 Word16 Word16   | Shift factor for Noise spectrum estimate (second sage)  1st stage PSD Mean buffer for precedent frame  Shift factor for PSD Mean buffer for precedent frame (1rst stage)  2nd stage PSD Mean buffer for precedent frame (2nd stage)  Shift factor for PSD Mean buffer for precedent frame (2nd stage)  1st stage PSD Mean buffer  Shift factor for PSD Mean buffer (1rst stage)  2nd stage PSD Mean buffer  Shift factor for PSD Mean buffer (2nd stage)  Nubmer of Fames (for the 2 stages)  Vad Flag (1 = SPEECH, 0 = NON SPEECH)  hangover  Number of speech frames (used to set hangover)  Mean energy for VAD  Vad Flag (1 = SPEECH, 0 = NON SPEECH)  hangover  Number of speech frames (used to set hangover) |

|              | vad_data_fd.AccTest       | Word32       | SpeechQSpec (for frame dropping)                 |
|--------------|---------------------------|--------------|--|
|              | vad_data_fd.AccTest2      | Word32       |  |
|              | vad_data_fd.SpecMean      | Word32       | SpecMean (for frame dropping)                    |
|              | vad_data_fd.MelValues     | Word16[2]    | SpeechQMel (for frame dropping)                  |
|              | vad_data_fd.SpecValues    | Word32       | SpeechQSpec (for frame dropping)                 |
|              | vad_data_fd.SpeechInVADQ  | Word16       | Flag (for frame dropping)                        |
|              | vad_data_fd.SpeechInVADQ2 | Word16       | Flag (for frame dropping)                        |
|              | gainFact.logDenEn1_32     | Word32[3]    | Denoise frame energy for gain factorization      |
|              | gainFact.lowSNRtrack32    | Word32       | Low SNR level for gain factorization             |
|              | gainFact. alfaGF16        | Word16       | Wiener filter gain factorization coefficient     |
| VADStructX_F |                           |              |  |
|              | Focus                     | Word16       | Position of circular buffe                       |
|              | HangOver                  | Word16       | Hangover length                                  |
|              | FlushFocus                | Word16       | Position in circular buffer when emptying at end |
|              | H_CountDown               | Word16       | Main hangover countdown                          |
|              | V_CountDown               | Word16       | Short hangover countdown                         |
|              | **OutBuffer               | Word32       | outBuffer pointer pointer                        |
|              | *OutBuffer                | Word32[7]    | outBuffer pointer                                |
|              | OutBuffer                 | Word16[7x15] | outBuffer  |

## Table 7b: VQ static variables

| Struct Name | Variable        | Type [Length] | Description                              |
|-------------|-----------------|---------------|--|
| coder_VAD.c | four_frames[27] | Word16[27]    | Previous frames used to build multiframe |
|             | plwQPHistory[3] | Word32[3]     | History of Pitch                         |
|             | IReliableFlag   | Word16        | Pitch reliability flag                   |

**Table 7c: Extension static variables** 

| Struct Name    | Variable                 | Type[Length] | Description                                |  |  |
|----------------|--------------------------|--------------|--|--|--|
|                | iFirstFrameFlag          | Word16       | First frame flag                           |  |  |
|                | pswUBSpeech              | Word16[200]  | Upper band speech                          |  |  |
|                | pswDownSampledProcSpeech | Word16[75]   | Down-sampled processed speech              |  |  |
|                | IwCritMax                | Word32       | Maximum power ratio                        |  |  |
|                | iOldPitchPeriod          | Word16       | Old pitch period value                     |  |  |
|                | iOldFrameNo              | Word16       | Old frame number                           |  |  |
| PCORR_STATE_be | s_be                     |              |  |  |  |
|                | lwX1_X1                  | Word32       | X1*X1                                      |  |  |
|                | lwZ1_Z1                  | Word32       | Z1*Z1                                      |  |  |
|                | lwZ2_Z2                  | Word32       | Z2*Z2                                      |  |  |
|                | lwX1_Z1                  | Word32       | X1*Z1                                      |  |  |
|                | lwX1_Z2                  | Word32       | X1*Z2                                      |  |  |
|                | lwZ1_Z2                  | Word32       | Z1*Z2                                      |  |  |
|                | swX1_Sum                 | Word16       | Sum of X1                                  |  |  |
|                | swZ1_Sum                 | Word16       | Sum of Z1                                  |  |  |
|                | swZ2_Sum                 | Word16       | Sum of Z2                                  |  |  |
|                | iBurstConst              | Word16       | Burst constant                             |  |  |
|                | iBurstCount              | Word16       | Burst count                                |  |  |
|                | iHangConst               | Word16       | Hang constant                              |  |  |
|                | iHangCount               | Word16       | Hang count                                 |  |  |
|                | iVADThld                 | Word16       | VAD threshold                              |  |  |
|                | iFrameCount              | Word16       | Frame count                                |  |  |
|                | iFUpdateFlag             | Word16       | Forced update flag                         |  |  |
|                | iHysterCount             | Word16       | Hysteresis count                           |  |  |
|                | iLastUpdateCount         | Word16       | Last update count                          |  |  |
|                | iSigThId                 | Word16       | Signal threshold                           |  |  |
|                | iUpdateCount             | Word16       | Update count                               |  |  |
|                | iChanEnrgShift           | Word16       | Channel energy shift                       |  |  |
|                | iChanNoiseEnrgShift      | Word16       | Channel noise energy shift                 |  |  |
|                | pswChanEnrg              | Word16[23]   | Channel energy                             |  |  |
|                | pswChanNoiseEnrg         | Word16[23]   | Channel noise energy                       |  |  |
|                | swBeta                   | Word16       | Beta value                                 |  |  |
|                | swSnr                    | Word16       | SNR value                                  |  |  |
| NormSw         | pnsLogSpecEnrgLong       |              |  |  |  |
|                | swMantissa               | Word16[23]   | Mantissa                                   |  |  |
|                | iShift                   | Word16[23]   | Shift                                      |  |  |
|                | swC0                     | Word16       | C0 value                                   |  |  |
|                | swC1                     | Word16       | C1 value                                   |  |  |
|                | swC2                     | Word16       | C2 value                                   |  |  |
|                | pswHpfXState             | Word16[6]    | High pass filter input state               |  |  |
|                | pswHpfYState             | Word16[12]   | High pass filter output state              |  |  |
|                | pswLpfXState             | Word16[6]    | Low pass filter input state                |  |  |
|                | pswLpfYState             | Word16[12]   | Low pass filter output state               |  |  |
|                | pswLfeXState             | Word16       | Low frequency emphasis filter input state  |  |  |
|                | pswLfeYState             | Word16[2]    | Low frequency emphasis filter output state |  |  |

# 5 File formats

This section describes the file formats used by the AFE, VQ & Extension programs.

# 5.1 Speech file

Speech files read by the X-AFE and written by the Extension consist of 16-bit words. The byte order depends on the host architecture (e.g. MSByte first on SUN workstations, LSByte first on PCs etc)

# Annex A (informative): Change history

| Change history |      |           |     |     |   |       |       |  |
|----------------|------|-----------|-----|-----|---|-------|-------|--|
| Date           | TSG# | TSG Doc.  | CR  | Rev | Subject/Comment   | Old   | New   |  |
| 2004-06        | 24   | SP-040343 |     |     | Version 6.0.0 approved at 3GPP TSG SA#24  | 2.0.0 | 6.0.0 |  |
| 2004-12        | 26   | SP-040837 | 001 | 1   | Software bug correction: Removal of Basicops simulation of "C" shift operator             | 6.0.0 | 6.1.0 |  |
| 2004-12        | 26   | SP-040837 | 002 | 1   | Software bug correction: Initialization of the variables lwc and i2aScale                 | 6.0.0 | 6.1.0 |  |
| 2004-12        | 26   | SP-040837 | 003 | 1   | Software bug correction: Wrong assignment of the variables *piReliableFlag and *pcQPIndex | 6.0.0 | 6.1.0 |  |
| 2004-12        | 26   | SP-040837 | 004 | 2   | Software bug correction: Use of incorrect variable fRefPeriod instead of iRefPeriod       | 6.0.0 | 6.1.0 |  |
| 2004-12        | 26   | SP-040837 | 005 |     | Add reference to test sequences document  | 6.0.0 | 6.1.0 |  |
| 2007-06        | 26   |           |     |     | Version for Release 7   | 6.1.0 | 7.0.0 |  |
| 2008-12        | 42   |           |     |     | Version for Release 8   | 7.0.0 | 8.0.0 |  |
| 2009-12        | 46   |           |     |     | Version for Release 9   | 8.0.0 | 9.0.0 |  |

# History

| Document history |              |             |  |  |  |  |
|------------------|--------------|-------------|--|--|--|--|
| V9.0.0           | January 2010 | Publication |  |  |  |  |
|                  |              |             |  |  |  |  |
|                  |              |             |  |  |  |  |
|                  |              |             |  |  |  |  |
|                  |              |             |  |  |  |  |