



INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

G.729

Annex C+
(02/00)

SERIES G: TRANSMISSION SYSTEMS AND MEDIA,
DIGITAL SYSTEMS AND NETWORKS

Digital transmission systems – Terminal equipments –
Coding of analogue signals by methods other than PCM

Coding of speech at 8 kbit/s using Conjugate
Structure Algebraic Code-Excited Linear Prediction
(CS-ACELP)

**Annex C+: Reference floating-point
implementation for integrating G.729 CS-ACELP
speech coding main body with Annexes B, D
and E**

ITU-T Recommendation G.729 – Annex C+

(Previously CCITT Recommendation)

ITU-T G-SERIES RECOMMENDATIONS
TRANSMISSION SYSTEMS AND MEDIA, DIGITAL SYSTEMS AND NETWORKS

INTERNATIONAL TELEPHONE CONNECTIONS AND CIRCUITS	G.100–G.199
INTERNATIONAL ANALOGUE CARRIER SYSTEM	
GENERAL CHARACTERISTICS COMMON TO ALL ANALOGUE CARRIER-TRANSMISSION SYSTEMS	G.200–G.299
INDIVIDUAL CHARACTERISTICS OF INTERNATIONAL CARRIER TELEPHONE SYSTEMS ON METALLIC LINES	G.300–G.399
GENERAL CHARACTERISTICS OF INTERNATIONAL CARRIER TELEPHONE SYSTEMS ON RADIO-RELAY OR SATELLITE LINKS AND INTERCONNECTION WITH METALLIC LINES	G.400–G.449
COORDINATION OF RADIOTELEPHONY AND LINE TELEPHONY	G.450–G.499
TESTING EQUIPMENTS	
TRANSMISSION MEDIA CHARACTERISTICS	
DIGITAL TRANSMISSION SYSTEMS	
TERMINAL EQUIPMENTS	G.700–G.799
General	G.700–G.709
Coding of analogue signals by pulse code modulation	G.710–G.719
Coding of analogue signals by methods other than PCM	G.720–G.729
Principal characteristics of primary multiplex equipment	G.730–G.739
Principal characteristics of second order multiplex equipment	G.740–G.749
Principal characteristics of higher order multiplex equipment	G.750–G.759
Principal characteristics of transcoder and digital multiplication equipment	G.760–G.769
Operations, administration and maintenance features of transmission equipment	G.770–G.779
Principal characteristics of multiplexing equipment for the synchronous digital hierarchy	G.780–G.789
Other terminal equipment	G.790–G.799
DIGITAL NETWORKS	G.800–G.899
DIGITAL SECTIONS AND DIGITAL LINE SYSTEM	G.900–G.999

For further details, please refer to ITU-T List of Recommendations.

ITU-T RECOMMENDATION G.729

CODING OF SPEECH AT 8 kbit/s USING CONJUGATE STRUCTURE ALGEBRAIC CODE-EXCITED LINEAR PREDICTION (CS-ACELP)

ANNEX C+

Reference floating-point implementation for integrating G.729 CS-ACELP speech coding main body with Annexes B, D and E

Summary

This annex, dubbed "Annex C+", extends the former Annex C (09/98).

Former Annex C contained G.729 main body and G.729 A and B floating point implementation. Annex C+ defines the integration of G.729 main body with Annexes B, D and E in floating point arithmetic.

This annex includes an electronic attachment containing version 2.1 of reference C code for floating point implementation of CS-ACELP at 6.4 kbit/s, 8 kbit/s and 11.8 kbit/s with DTX functionality.

Source

Annex C+ to ITU-T Recommendation G.729 was prepared by ITU-T Study Group 16 (1997-2000) and was approved under the WTSC Resolution No. 1 procedure on 17 February 2000.

FOREWORD

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

The World Telecommunication Standardization Conference (WTSC), which meets every four years, establishes the topics for study by the ITU-T Study Groups which, in their turn, produce Recommendations on these topics.

The approval of Recommendations by the Members of the ITU-T is covered by the procedure laid down in WTSC Resolution No. 1.

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

NOTE

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

INTELLECTUAL PROPERTY RIGHTS

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

As of the date of approval of this Recommendation, the ITU had received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementors are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database.

© ITU 2000

All rights reserved. No part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from the ITU.

CONTENTS

	Page
Annex C+ – Reference floating-point implementation for integrating G.729 CS-ACELP speech coding main body with Annexes B, D and E	1
C+.1 Scope	1
C+.2 Normative references	1
C+.3 Overview	1
C+.4 New functionality.....	2
C+.4.1 Annex B DTX operation with Annex D	2
C+.4.2 Annex B DTX operation with Annex E	2
C+.5 Algorithm description	2
C+.5.1 Music detection.....	2
C+.5.2 Update of state variables specific to Annex D during discontinued transmission	7
C+.5.3 Update of state variables specific to Annex E during discontinued transmission	7
C+.6 Description of C source code	7
C+.6.1 Use of the simulation software	7
C+.6.2 Organization of the simulation software	8
Electronic attachment:	
– reference C code implementation	

Recommendation G.729

CODING OF SPEECH AT 8 kbit/s USING CONJUGATE STRUCTURE ALGEBRAIC CODE-EXCITED LINEAR PREDICTION (CS-ACELP)

ANNEX C+

Reference floating-point implementation for integrating G.729 CS-ACELP speech coding main body with Annexes B, D and E¹

(Geneva, 2000)

C+.1 Scope

This annex provides a description of integrating the G.729 main body with Annexes B, D and E in floating point arithmetic. It presents a standard way of performing this integration and expansion of the functionality hereby guiding the industry and ensuring a standard speech quality and compatibility worldwide. The integration has been performed with focus on several constraints in order to satisfy the need of the industry:

- 1) Bit-exactness with the main body in floating point (Annex C).
- 2) Minimum additional program code, memory, and complexity usage.
- 3) Stringent quality requirements to new functionality inline with quality and application areas of the according standard annexes.

C+.2 Normative references

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; all users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published.

- [1] ITU-T Recommendation G.729 (1996), *Coding of speech at 8 kbit/s using conjugate structure algebraic code-excited linear prediction (CS-ACELP)*.
- [2] ITU-T Recommendation G.729 Annex B (1996), *A silence compression scheme for G.729 optimized for terminals conforming to Recommendation V.70*.
- [3] ITU-T Recommendation G.729 Annex C (1998), *Reference floating-point implementation for G.729 CS-ACELP 8 kbit/s speech coding*.
- [4] ITU-T Recommendation G.729 Annex D (1998), *6.4 kbit/s CS-ACELP speech coding algorithm*.
- [5] ITU-T Recommendation G.729 Annex E (1998), *11.8 kbit/s CS-ACELP speech coding algorithm*.

¹ This annex includes an electronic attachment containing version 2.1 of reference C code for floating point implementation of CS-ACELP at 6.4 kbit/s, 8 kbit/s and 11.8 kbit/s with DTX functionality.

C+.3 Overview

Recommendation G.729 main body [1] and Annexes B [2], D [4] and E [5] provide a bit-exact, fixed-point specification of a CS-ACELP coder at 8 kbit/s, with DTX functionality, lower and higher bit extension capability at 6.4 and 11.8 kbit/s. Exact details of these specifications are given in bit-exact, fixed-point C code in an electronic attachment to this annex. Annex C [3] describes and defines an alternative implementation of Recommendation G.729 main body. Annex C+ describes and defines the integration of the G.729 main body with Annexes B, D and E in floating point arithmetic. It can be considered as an extension of Annex C of G.729.

C+.4 New functionality

This subclause presents a brief overview of the modifications/additions to the algorithms in order to facilitate the integration of the main body and Annexes B, D and E. Also certain additions have been found necessary in order to accommodate the application area of the different modules.

C+.4.1 Annex B DTX operation with Annex D

Integrating Annexes B and D functionality in order to provide DTX operation with Annex D is straightforward. The VAD (Voice Activity Detection), SID coding (Silence Description), and CNG (Comfort Noise Generation) of Annex B are reused without any modifications. Care is taken to update the parameters for the phase dispersion for the postfilter in Annex D during discontinued transmission (see C+.5.2).

C+.4.2 Annex B DTX operation with Annex E

Integrating Annexes B and E functionality in order to provide DTX operation with Annex E is slightly more involved. Since the DTX operation of Annex B is based on the 10th order LPC analysis, the VAD function of Annex B is performed after the 10th order forward adaptive LPC analysis and before the backward adaptive LPC analysis of Annex E. In case the VAD function detects "non-speech", the LPC mode of Annex E is forced to forward adaptive LPC and the backward adaptive LPC analysis is skipped. Furthermore, it has been found necessary to add a correctional module after the VAD in order to detect music and accommodate the somewhat expanded application area of Annex E – one of the purposes of Annex E is to provide transmission capability of music with a certain quality. Accordingly, during the development of Annex E there were strict requirements to the performance with music signals. On the other hand, for the main body and Annexes B and D there were no strict requirements to the performance with music signals. In order to guarantee the quality with music signals of Annex E during Annex B DTX operation, the music detection function forces the VAD to "speech" during music segments, hereby ensuring that the music segments are coded with the 11.8 kbit/s of Annex E. The SID coding and the CNG of Annex B are reused without any modifications. Furthermore, care is taken to appropriately update the parameters of the LPC mode selection algorithm of Annex E during discontinued transmission (see C+.5.3).

C+.5 Algorithm description

This subclause presents the algorithm description of the necessary additions to the algorithms of the individual annexes in order to facilitate the integration. All remaining modules originate from the main body, Annex B, D or E.

C+.5.1 Music detection

The music detection is a new function. It is performed immediately following the VAD and forces the VAD to "speech" during music segments. It is active only during Annex E operation, though its parameters are updated continuously independently of bit-rate mode during DTX operation of the integrated G.729.

The music detection algorithm corrects the decision from the Voice Activity Detection (VAD) in the presence of music signals. It is used in conjunction with Annex E during Annex B DTX operation, i.e. in Discontinuous Transmission mode. The music detection is based on the following parameters:

- *Vad_dec*: VAD decision of the current frame.
- *PVad_dec*: VAD decision of the previous frame.
- *Lpc_mod*: flag indicator of either forward or backward adaptive LPC of the previous frame.
- *Rc*: reflection coefficients from LPC analysis.
- *Lag_buf*: buffer of corrected open loop pitch lags of last 5 frames.
- *Pgain_buf*: buffer of closed loop pitch gain of last 5 subframes.
- *Energy*: first autocorrelation coefficient $R(0)$ from LPC analysis.
- *LEnergy*: normalized log energy from VAD module.
- *Frm_count*: counter of the number of processed signal frames.
- *Rate*: selection of speech coder.

The algorithm has two main parts:

- 1) Computation of relevant parameters.
- 2) Classification based on parameters.

C+.5.1.1 Computation of Relevant Parameters

This subclause describes the computation of the parameters used by the decision module.

Partial Normalized Residual Energy

$$LEnergy = 10 \log_{10} \left(\prod_{i=1}^4 \left(1 - Rc(i)^2 \right) \frac{Energy}{240} \right)$$

Spectral Difference and Running Mean of Partial Normalized Residual Energy of Background Noise

A spectral difference measure between the current frame reflection coefficients Rc and the running mean reflection coefficients of the background noise mRc is given by:

$$SD = \sum_{i=1}^{10} (Rc(i) - mRc(i))^2$$

The running means \overline{mrc} and $mLEnergy$ are updated as follows using the VAD decision Vad_deci that was generated by the VAD module.

```

if Vad_deci == NOISE {
     $\overline{mrc} = 0.9\overline{mrc} + 0.1rc$ 
     $mLEnergy = 0.9mLEnergy + 0.1LEnergy$ 
}

```

Open loop Pitch Lag Correction for Pitch Lag Buffer Update

The open loop pitch lag T_{op} is corrected to prevent pitch doubling or tripling as follows:

$$\begin{aligned} avg_lag &= \sum_{i=1}^4 \frac{Lag_buf(i)}{4} \\ \text{if} &\left(abs\left(\frac{T_{op}}{2} - avg_lag \right) \leq 2 \right) \\ Lag_buf(5) &= \frac{T_{op}}{2} \\ \text{else if} &\left(abs\left(\frac{T_{op}}{3} - avg_lag \right) \leq 2 \right) \\ Lag_buf(5) &= \frac{T_{op}}{3} \\ \text{else} & \\ Lag_buf(5) &= T_{op} \end{aligned}$$

It should be noted that the open loop pitch lag T_{op} is not modified and is the same as derived by the open loop analysis.

Pitch Lag Standard Deviation

$$std = \sqrt{\frac{Var}{4}}$$

where:

$$Var = \sum_{i=1}^{i=5} (Lag_buf(i) - \mu)^2 \text{ and } \mu = \sum_{i=1}^{i=5} \frac{Lag_buf(i)}{5}$$

Running Mean of Pitch Gain

$$mPgain = 0.8mPgain + 0.2\theta, \text{ where } \theta = \sum_{i=1}^{i=5} \frac{Pgain_buf(i)}{5}$$

The pitch gain buffer $Pgain_buf$ is updated after the subframe processing with a pitch gain value of 0.5 if $Vad_deci = NOISE$, and otherwise with the quantized pitch gain.

Pitch Lag Smoothness and Voicing Strength Indicator

A pitch lag smoothness and voicing strength indicator $Pflag$ is generated using the following logical steps:

First, two intermediary logical flags $Pflag1$ and $Pflag2$ are obtained as:

if ($std < 1.3$ and $mPgain > 0.45$) set $Pflag1 = 1$ else 0

if ($mPgain > Thres$) set $Pflag2 = 1$ else 0,

where $Thres = 0.73$ if $Rate = G729D$, otherwise $Thres = 0.63$

Finally, *Pflag* is determined from the following:

```
if ((PVad_dec == VOICE and (Pflag1 == 1 or Pflag2 == 1)) or (Pflag2 == 1))
set Pflag = 1 else 0
```

Stationarity counters

A set of counters are defined and updated as follows:

- a) *count_consc_rflag* tracks the number of consecutive frames where the 2nd reflection coefficient and the running mean of the pitch gain satisfy the following condition:

```
if (Rc(2) < 0.45 and Rc(2) > 0 and mPgain < 0.5)
    count_consc_rflag = count_consc_rflag + 1
else
    count_consc_rflag = 0
```

- b) *count_music* tracks the number of frames where the previous frame uses backward adaptive LPC and the current frame is "speech" (according to the VAD) within a window of 64 frames.

```
if (Lpc_mod == 1 and Vad_deci == VOICE)
    count_music = count_music + 1
```

Every 64 frames, a running mean of *count_music*, *mcount_music* is updated and reset to zero as described below:

```
if ((Frm_count mod 64) == 0){
    if (Frm_count == 64)
        mcount_music = count_music
    else
        mcount_music = 0.9mcount_music + 0.1count_music
}
```

- c) *count_consc* tracks the number of consecutive frames where the *count_music* remains zero:

```
if (count_music == 0)
    count_consc = count_consc + 1
else
    count_consc = 0
    if (count_consc > 500 or count_consc_rflag > 150) set mcount_music = 0
```

count_music in b) is reset to zero every 64 frames after the update of the relevant counters.

The logic in c) is used to reset the running mean *count_music*.

- d) *count_pflag* tracks the number of frames where *Pflag* = 1, within a window of 64 frames.

```
if (Pflag == 1)
    count_pflag = count_pflag + 1
```

Every 64 frames, a running mean of *count_pflag*, *mcount_pflag*, is updated and reset to zero as described below:

```

if ((Frm_count mod 64) == 0){
  if (Frm_count == 64)
    mcount_pflag = count_pflag
  else{
    if (count_pflag > 25)
      mcount_pflag = 0.98mcount_pflag + 0.02count_pflag
    else (count_pflag > 20)
      mcount_pflag = 0.95mcount_pflag + 0.05count_pflag
    else
      mcount_pflag = 0.9mcount_pflag + 0.1count_pflag
  }
}

```

e) *count_consc_pflag* tracks the number of consecutive frames satisfying the following condition.

```

if (count_pflag == 0)
  count_consc_pflag = count_consc_pflag + 1
else
  count_consc_pflag = 0

```

if (*count_consc_pflag* > 100 or *count_consc_rflag* > 150) set *mcount_pflag* = 0

count_pflag is reset to zero every 64 frames. The logic in e) is used to reset the running mean of *count_pflag*.

C+.5.1.2 Classification

Based on the estimation of the above parameters, the VAD decision *Vad_deci* from the VAD module is reverted if the following conditions are satisfied:

```

if (Rate = G729E){
  if (SD > 0.15 and (Lenergy - mLenergy) > 4 and LLenergy > 50)
    Vad_deci = VOICE
  else if ((SD > 0.38 or (Lenergy - mLenergy) > 4) and LLenergy > 50)
    Vad_deci = VOICE
  else if ((mcount_pflag >= 10 or mcount_music >= 5 or Frm_count < 64)
    and LLenergy > 7)
    Vad_deci = VOICE
}

```

Note that the music detection function is called all the time regardless of the operational coding mode in order to keep the memories current. However, the VAD decision *Vad_deci* is altered only if the integrated G.729 is operating at 11.8 kbit/s (Annex E). It should be noted that the music detection only has the capability to change the decision from "non-speech" to "speech" and not vice versa.

C+.5.2 Update of state variables specific to Annex D during discontinued transmission

The only state variables specific to Annex D are the state variables of the phase dispersion module (see D.6.2 of ITU-T G.729 – Annex D [4]) at the decoder. In case of inactive frames, the same update procedure as in case of nominal bit rate (8 kbit/s) is followed using as adaptive and ACELP gain estimations the gain values computed by the comfort noise excitation generator (see B.4.4 of ITU-T G.729 – Annex B [2]). Note also that the update for the higher rate is identical to the update for the nominal bit rate.

C+.5.3 Update of state variables specific to Annex E during discontinued transmission

C+.5.3.1 Update of encoder state variables specific to Annex E

At the encoder in case of inactive frames, the update of state variables is identical to the update performed in G.729 Annex E [5] in case of switch to the nominal bit rate 8 kbit/s. The update procedure is the following: the LP mode is set to 0, the global stationarity indicator is decreased and the high stationarity indicator is reset to 0 (see E.3.2.7.2 of ITU-T G.729 – Annex E [5]), the interpolation factor used to smoothly switch from LP forward filter to backward LP filter is reset to its maximum value (see E.3.2.7.1 of ITU-T G.729 – Annex E [5]). Note that this update is also performed in case of switch to the lower bit rate 6.4 kbit/s.

C+.5.3.2 Update of decoder state variables specific to Annex E during discontinued transmission

At the decoder in case of inactive frames, the update of state variables is almost identical to the update performed in G.729 Annex E [5] in case of switch to the forward mode only rates (8 kbit/s and 6.4 kbit/s) except that the pitch delay stationary indicator is reset to 0 instead of being computed by the pitch tracking procedure (see E.4.4.5 of ITU-T G.729 – Annex E [5]).

C+.6 Description of C source code

The Annex C+ integrating the G.729 main body, Annexes B, D and E functionality is simulated in floating point arithmetic ANSI-C code. As for G.729 Annex C, the typedef.h file contains a statement enabling the definition of all floating-point variables and constants as type either double or single. The ANSI-C code represents the normative specification of this annex. The algorithmic description given by the C code shall take precedence over the texts contained in the main body of Recommendation G.729 and in Annexes B, C, D, E and C+.

C+.6.1 Use of the simulation software

The C code consists of two main programs **codercp.c** and **decodercp.c**, which simulate encoder and decoder, respectively. The encoder is executed as follows:

codercp inputfile bitstreamfile dtx_option rate_option

The decoder is executed as follows:

decodercp bitstreamfile outputfile

The input file and output file are 8 kHz sampled data files containing 16-bit PCM signals. The bit stream file is a binary file containing the bit stream; the mapping table of the encoded bit stream is contained in the simulation software. The two options for the encoder are: **dtx_option** and **rate_option** where:

dtx_option = 1: DTX enabled 0: DTX disabled, the default is 0 (DTX disabled).

rate_option = 0 to select the lower rate (6.4 kbit/s); = 1 to select the main Recommendation G.729 (8 kbit/s); = 2 is to select the higher rate (11.8 kbit/s) or a **file_rate_name**: a binary file of 16-bit word containing either 0, 1, 2 to select the rate on a frame-by-frame basis; the default is 1 (8 kbit/s).

C+.6.2 Organization of the simulation software

Table C+.1 gives the list of the software files names with a brief description, and it is also indicated what annex the file has been derived from (identical or similar to G.729 Annex C file or fixed to floating point transcription of the files). Note that the fixed point files basic_op.c, oper_32b.c, dspfunc.c and basic_op.h, oper_32b.h are not needed for floating-point arithmetic. As for G.729 Annex C, a float to short conversion routine has been added to the file utilities file utilcp.c.

Table C+.1/G.729 – List of software files of integrated G.729 in floating point

File name	Description	Link
Gainpred.c	Gain predictor	C
Lpfuncpc.c	Miscellaneous routines related to LP filter	C+E
Cor_func.c	miscellaneous routines related to excitation computation	C
Pre_proc.c	Pre-processing (HP filtering and scaling)	C
P_parity.c	Compute pitch parity	C
Pwf.c	Computation of perceptual weighting coefficients (8 kbit/s)	C
Pred_lt3.c	Generation of adaptive codebook	C
Post_pro.c	Post processing (HP filtering and scaling)	C
Tab_ld8k.c	ROM tables	C
Ld8k.h	Function prototypes	C
Tab_ld8k.h	Extern ROM table declarations	C
Typedef.h	Data type definition (machine dependent)	C
Taming.c	Pitch instability control	C
Qsidgain.c	SID Gain Quantization	B
QsidLSF.c	SID-LSF Quantization	B
Tab_dtx.c	ROM tables	B
Sid.h	Prototype and Constants	B
Octet.h	Octet transmission mode definition	B
Tab_dtx.h	Extern ROM table declarations	B
Pwfe.c	Computation of perceptual weighting coefficients (11.8 kbit/s)	E
Vad.c	VAD	B
Dtx.c	DTX Decision	B
Vad.h	Prototype and Constants	B
Dtx.h	Prototype and Constants	B
Calcexc.c	CNG Excitation Calculation	B
Dec_sid.c	Decode SID Information	B
Utilcp.c	Utility functions	C+B
Phdisp.c	Phase dispersion	D
Bwfw.c	Backward/forward switch selection	E
Bwfwfunc.c	Miscellaneous routines related to backward/forward switch	E
Filtere.c	Filter functions	C+E
Lpccp.c	LP analysis	C+E
Lspcdece.c	LSP decoding routines	C+E

Table C+.1/G.729 – List of software files of integrated G.729 in floating point (concluded)

File name	Description	Link
Lspgetqe.c	LSP quantizer	C+E
Qua_lspe.c	LSP quantizer	C+E
Track_pi.c	Pitch tracking	E
Codercp.c	Main encoder routine	C+B+D+E
Codld8cp.c	Encoder routine	C+B+D+E
Decodcp.c	Main decoder routine	C+B+D+E
Decld8cp.c	Decoder routine	C+B+D+E
Acelp_cp.c	search ACELP fixed codebook (6.4, 8, 11.8 kbit/s)	C+D+E
Dacelpcp.c	Decode algebraic codebook (6.4, 8, 11.8 kbit/s)	C+D+E
Pitchcp.c	Pitch search	C+D+E
Declagcp.c	Decode adaptive-codebook index	C+D+E
Q_gaincp.c	Gain quantizer	C+D+E
Degaincp.c	Decode gain	C+D+E
Pstpcp.c	Postfilter routines	C+B+E
Bitscp.c	Bit manipulation routines	C+B+D+E
Tabld8cp.c	ROM tables for G.729 at 6.4 and 11.8 kbit/s	D+E
Tabld8cp.h	Extern ROM declarations for G.729 at 6.4 and 11.8 kbit/s	D+E
Ld8cp.h	Constant and Function prototypes for G.729 at 6.4 and 11.8 kbit/s	D+E
Mus_dtct.c	Music detection module	New

ITU-T RECOMMENDATIONS SERIES

Series A	Organization of the work of the ITU-T
Series B	Means of expression: definitions, symbols, classification
Series C	General telecommunication statistics
Series D	General tariff principles
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Construction, installation and protection of cables and other elements of outside plant
Series M	TMN and network maintenance: international transmission systems, telephone circuits, telegraphy, facsimile and leased circuits
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks and open system communications
Series Y	Global information infrastructure
Series Z	Languages and general software aspects for telecommunication systems