INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

G.191

(03/93)

GENERAL CHARACTERISTICS OF INTERNATIONAL TELEPHONE CONNECTIONS AND INTERNATIONAL TELEPHONE CIRCUITS

SOFTWARE TOOLS FOR SPEECH AND AUDIO CODING STANDARDIZATION

ITU-T Recommendation G.191

(Previously "CCITT Recommendation")

FOREWORD

The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of the International Telecommunication Union. 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, established the topics for study by the ITU-T Study Groups which, in their turn, produce Recommendations on these topics.

ITU-T Recommendation G.191 was prepared by the ITU-T Study Group XV (1988-1993) and was approved by the WTSC (Helsinki, March 1-12, 1993).

NOTES

1 As a consequence of a reform process within the International Telecommunication Union (ITU), the CCITT ceased to exist as of 28 February 1993. In its place, the ITU Telecommunication Standardization Sector (ITU-T) was created as of 1 March 1993. Similarly, in this reform process, the CCIR and the IFRB have been replaced by the Radiocommunication Sector.

In order not to delay publication of this Recommendation, no change has been made in the text to references containing the acronyms "CCITT, CCIR, or IFRB" or their associated entities such as Plenary Assembly, Secretariat, etc. Future editions of this Recommendation will contain the proper terminology related to the new ITU structure.

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

© ITU 1993

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
1	General	1
2	Software Tools	1
3	License and Copyright	1
Annex A – List of software tools available		2
Annex B – CCITT software tools general public license		5

SOFTWARE TOOLS FOR SPEECH AND AUDIO CODING STANDARDIZATION

(Helsinki, 1993)

1 General

In the process of generating speech and audio coding standards, the following situations often happen:

a) in many cases, experimental results generated with different software tools may not be directly compared;

b) software tools used by different organizations may not perfectly conform to related CCITT Recommendations, which may delay CCITT standardization processes;

c) CCITT Recommendations may leave scope for different implementations;

d) new speech and audio coding standards are increasing in complexity, leading to non-bitexact specifications; furthermore, appropriate testing procedures to assure interoperability of different implementations are needed.

The need for a common set of tools has been recognized in past CCITT standardization activities of speech algorithms. As a consequence, a library of portable, interworkable and reliable software routines has been established.

2 Software tools

To clarify the use of the set of software tools, arranged as a software tool library, the CCITT makes the following recommendations:

1) The software tools specified in the Annex A should be used as building modules of signal processing blocks to be used in the process of generation of CCITT Recommendations, particularly those concerned with speech and audio coding algorithms.

2) Some of the tools shall be used in procedures for the verification of interoperability of CCITT standards, mainly of speech and audio coding algorithms whose description is in terms of non-bitexact specifications.

3) The use of these modules should be made strictly in accordance with the technical instructions of their attached documentation, and should respect the following terms:

3 License and copyright

The modules in the CCITT Software Tool Library (STL) are free software; they can be redistributed and/or modified under the terms of the "CCITT software tools General Public License" of the Annex B, as published by the CCITT; this applies to any of the versions of the modules in the STL.

The STL has been carefully tested and it is believed that both the modules and the example programs on their usage conform to their description documents. Nevertheless, the CCITT STL is provided "as is", in the hope that it will be useful, but without any warranty.

The STL is intended to help the scientific community to achieve new standards in telecommunications more efficiently, and for such must not be sold, entirely or in parts. The original developers, except where otherwise noted, retain ownership of their copyright, and allow their use under the terms and conditions of the "CCITT software tools General Public License".

Annex A

List of software tools available

(This annex forms an integral part of this Recommendation)

This annex contains a list with a short description of the latest version released of the software tools available in the CCITT Software Tool Library. This is referred to in the associated documentation as the Software Tool Library release 1992, or STL92. All the routines in the modules are written in C. One application of the STL92 can be found in "Software tools for the qualification of a codec at 8 kbit/s".

To obtain a copy of the software tools listed below, as well as the associated "CCITT Software Tool Library manual" and the above referenced application example, contact the CCITT/ITU Secretariat:

ITU General Secretariat
Sales Service
Place des Nations
CH-1211 Geneve 20
Switzerland

a) *Example programs available*

Associated header file: ugstdemo.h

The following programs are examples of the use of the modules:

g711demo.c	on the use of the G.711 module.
g721demo.c	on the use of the G.721 module.
sv56demo.c	on the use of the speech voltmeter module, and also the gain/loss routine.
eiddemo.c	on the use of the error insertion device for bit error insertion and frame erasure.
hqdemo.c	on the use of the high-quality and IRS filters, associated with the rate change module.
pcmdemo.c	on the use of the G.712 (standard PCM) filters, associated with the rate change module.
mnrudemo.c	on the use of the modulated noise reference unity (P.81) module.
spdemo.c	on the use of the serialization and parallelization routines of the utility module.

b) *Rate change module with FIR (finite impulse response) routines*

Name: hqflt.c

Associated header file: hqflt.h

Functions included:

hq_kernel	FIR filtering function.
hq_down_2_to_1_init	initialize 2:1 down-sampling filter.
hq_down_3_to_1_init	initialize 3:1 down-sampling filter.
hq_up_1_to_2_init	initialize 1:2 up-sampling filter.
hq_up_1_to_3_init	initialize 1:3 up-sampling filter.
irs_8khz_init( )	initialize 8 kHz IRS weighting filter.
irs_16khz_init	initialize 16 kHz IRS weighting filter.
hq_reset	clear state variables.
hq_free	deallocate FIR-filter memory.

c) *Rate change module with IIR routines*

Name: pcmflt.c

Associated header file: pcmflt.h

Functions included:

stdpcm_kernel	standard PCM filter (kernel).
stdpcm_16khz_init	initialization of standard PCM-filter for input and output data at 16 kHz.
stdpcm_1_to_2_init	as "stdpcm_16khz_init( )", but needs input with sampling frequency of 8 kHz and returns data at 16 kHz.
stdpcm_2_to_1_init	as "stdpcm_16khz_init( )", but needs input with sampling frequency of 16 kHz and returns data at 8 kHz.
stdpcm_reset	clear state variables (needed only if another signal should be processed with the same filter).
stdpcm_free	deallocate filter memory.

d) *Error insertion module*

Name: eid.c

Associated header file: eid.h

Functions included:

open_eid	initializes the error pattern generator.
BER_generator	generates a bit error sequence with properties, defined by "open_eid".
BER_insertion	modifies the input data bits according to the error pattern, stored in a buffer.
FER_module	frame erasure module.
close_eid	frees memory allocated to the EID state variable buffer.

e) *G.711 module*

Name: g711.c

Associated header file: g711.h

Functions included:

alaw_compress	compands 1 vector of linear PCM samples to A-law; uses 13 Most Significant Bits (MSBs) from input and 8 Least Significant Bits (LSBs) on output.

alaw_expand	expands 1 vector of A-law samples to linear PCM; use 8 LSBs from input and 13 MSBs on output.
μlaw_compress	compands 1 vector of linear PCM samples to μ-law; uses 14 MSBs from input and 8 LSBs on output.
μlaw_expand	expands 1 vector of μ-law samples to linear PCM use 8 LSBs from input and 14 MSBs on output.

f) *G.721 module*

Name: g721.c

Associated header file: g721.h

Functions included:

G721_encode	G726 encoder at 32 kbit/s.
G721_decode	G726 decoder at 32 kbit/s.

g) *Modulated noise reference unit module*

Name: mnru.c

Associated header file: mnru.h

Functions included:

MNRU_process	Module for addition of modulated noise to a vector of samples, according to CCITT Recommendation P.81 (*Blue Book*), for the narrow-band model.

h) *Speech voltmeter module*

Name: sv-p56.c

Associated header file: sv-p56.h

Functions included:

init_speech_voltmeter	initializes a speech voltmeter state variable.
speech_voltmeter	measurement of the active speech level of data in a buffer according to *Blue Book* P.56.

i) *Module with UGST utilities*

Name: ugst-utl.c

Associated header file: ugst-utl.h

Functions included:

scale	gain/loss insertion algorithm.
sh2fl_16bit	conversion of two's complement, 16-bit integer to floating point.
sh2fl_15bit	conversion of two's complement, 15-bit integer to floating point.
sh2fl_14bit	conversion of two's complement, 14-bit integer to floating point.
sh2fl_13bit	conversion of two's complement, 13-bit integer to floating point.
sh2fl_12bit	conversion of two's complement, 12-bit integer to floating point.
sh2fl	generic function for conversion from integer to floating point.
fl2sh_16bit	conversion of floating point data to two's complement, 16-bit integer.
fl2sh_15bit	conversion of floating point data to two's complement, 15-bit integer.
fl2sh_14bit	conversion of floating point data to two's complement, 14-bit integer.

fl2sh_13bit	conversion of floating point data to two's complement, 13-bit integer.
fl2sh_12bit	conversion of floating point data to two's complement, 12-bit integer.
fl2sh	generic function for conversion from floating point to integer.
serialize_left_justified	serialization for left-justified data.
serialize_right_justified	serialization for right-justified data.
parallelize_left_justified	parallelization for left-justified data.
parallelize_right_justified	parallelization for right-justified data.

Annex B

CCITT software tools General Public License

(This annex forms an integral part of this Recommendation)

Terms and conditions

B.1 This License Agreement applies to any module or other work related to the CCITT Software Tool Library, and developed by the User's Group on software tools. The term "Module", below, refers to any such module, tool or work, and a "work based on the Module" means either the Module or any work containing the Module or a portion of it, either verbatim or with modifications. Each licensee is addressed as "you".

B.2 You may copy and distribute verbatim copies of the Module's source code as you receive it, in any medium, provided that you:

- conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty;
- keep intact all the notices that refer to this General Public License and to the absence of any warranty; and
- give any other recipients of the Module a copy of this General Public License along with the Module.

You may charge a fee for the physical act of transferring a copy.

B.3 You may modify your copy or copies of the Module or any portion of it, and copy and distribute such modifications under the terms of B.1, provided that you also do the following:

- cause the modified files to carry prominent notices stating that you changed the files and the date of any change;
- when any work that you distribute or publish, that contains the Module or any part thereof, either with or without modifications, then do not charge a license fee to all third parties for the use of that Module under the terms of this General Public License (except that you may choose to grant warranty protection to some or all third parties, at your option).
- If the modified module normally reads commands interactively when run, you must cause it, when it starts to run interactive use, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the module under these conditions, and telling the user how to view a copy of this General Public License.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

Mere aggregation of another independent work with the Module (or its derivative) on a volume of a storage or distribution medium does not bring the other work under the scope of these terms.

B.4 You may copy and distribute the Module (or a portion or derivative of it, under B.2) in object code or executable form under the terms of B.1 and B.2 provided that you also do one of the following:

- accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of B.1 and B.2; or,

- accompany it with a written offer, valid for at least three years, to give any third party free (except for a nominal charge for the cost of distribution) a complete machine-readable copy of the corresponding source code, to be distributed under the terms of B.1 and B.2; or,

- accompany it with the information you received as to where the corresponding source code may be obtained. This alternative is allowed only for noncommercial distribution and only if you received the Module in object code or executable form alone.

Source code for a work means the preferred form of the work on which making modifications could be made. For an executable file, complete source code means all the source code for all modules it contains; but, as a special exception, it need not include source code for modules which are standard libraries that accompany the operating system on which the executable file runs, or for standard header files or definitions files that accompany that operating system.

B.5 You may not copy, modify, sublicense, distribute or transfer the Module except as expressly provided under this General Public License. Any attempt otherwise to copy, modify, sublicense, distribute or transfer the Module will automatically terminate your rights to use the Module under this License. However, parties who have received copies, or rights to use copies, from you under this General Public License will not have their licenses terminated so long as such parties remain in full compliance.

B.6 By copying, distributing or modifying the Module (or any work based on the Module) you indicate your acceptance of this license, and all its terms and conditions.

B.7 Each time you redistribute the Module (or any work based on the Module), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Module subject to these terms and conditions. You may not impose any further restrictions on the recipients' rights granted herein.

B.8 The CCITT may publish revised and/or new versions of this General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Module specifies a version number of the license which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the CCITT. If the Module does not specify a version number of the license, you may choose any version ever published by the CCITT.

B.9 If you wish to incorporate parts of the Module into other work whose distribution conditions are different, write to the author of that work to ask for permission. For work which is copyrighted by the CCITT, write to the CCITT Secretariat. In giving such permission the author of the work concerned must accept the conditions of this license and the two goals of preserving the free status of all derivatives of this free software and of promoting the sharing and reuse of software generally.

B.10 Because the Module is licensed free of charge, there is no warranty for the Module, to the extent permitted by applicable law. Except when otherwise stated in writing the copyright holders and/or other parties provide the Module "as is" without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The entire risk as to the quality and performance of the Module is with you. Should the Module prove defective, you assume the cost of all necessary servicing, repair or correction.

B.11 In no event unless required by applicable law or agreed to in writing will any copyright holder, or any other party who may redistribute the Module as permitted above, be liable to you for damages, including any general, special, incidental or consequential damages arising out of the use or inability to use the Module (including but not limited to loss of data or data being rendered inaccurate or losses sustained by you or third parties or a failure of the Module to operate with any other modules), even if such holder or other party has been advised of the possibility of such damages.