Easy G.728

Technical Document
Version 2.0－ Revision 2006-11-11

Table of Contents

Introduction

EasyG728 is an implementation of ITU G.728. EasyG728 support multiple channels concurrent. There is no limit in concurrent channels and it can up to thousands channels. EasyG728 is an 16kbps coder that encodes/decodes speech signal. The coder operates on speech frames of 10 ms, corresponding to 80 samples at a sampling rate of 8000 samples/sec. In addition to the 10 ms speech frame duration, there is also a look-ahead delay of 0.625 ms, resulting in a total initial algorithmic delay of 10.625 ms.

EasyG728 codec specifications	
Bit rate (kbps)	16
Speech sampling rate(Hz)	8000
Frame duration (ms)	10
Look-ahead delay (ms)	0.625
Samples in one Frame	80
Frame size before encode(bytes)	160
Frame size after encode(bytes)	20

EasyG728 has a binary release version on Windows and Linux. The source code of EasyG728 is written by C/C++, so you can easily port it to UNIX, PPC,DSP, Vxworks or other operation system that support C/C++.

PACKAGE CONTENTS

EasyG728.pdf	This document
EasyG728.lib	Win32 statically linkable library of G728 for Pentium and compatible processors.
libG728.a	Linux statically linkable library of G728 for Pentium and compatible processors.
EasyG728.h	API prototypes and constants declarations required by the sample programs.
test_encode directory	Microsoft VC6.0 sample application and Linux GCC sample application. Demonstrating encoder API calls to the codec for encoding a speech file.
test_decode directory	Microsoft VC6.0 sample application and Linux GCC sample application. Demonstrating decoder API calls to the codec for decoding a speech file.

The encoder requires raw 16-bit mono PCM speech data sampled at 8000 Hz as input, i.e., without any header information. For every speech frame, consisting of 80*16 bit (160 bytes) samples

CODEC COMPLEXITY

The codec complexity is represented as percentage of CPU usage, and is as follows when tested on an Intel 800 MHz Celeron-MMX:
Encoder less than 1% CPU time
Decoder less than 1% CPU time

ABOUT THE ENCODER/DECODER SAMPLE PROGRAMS

The sample programs under test_encode directory and test_decode directory are used to simulate the encoder and decoder, and demonstrate how to initialize and call the encoding and decoding process. The encoder and decoder are run as follows (where **infile** and **outfile** are raw 16 bit PCM files sampled at 8 kHz):

EasyG728_encoder infile bitstream
EasyG728_decoder bitstream outfile

To build the speech encoder (or decoder) sample programs on Windows, you can open TEST_ENCODE.dsw or TEST_DECODE.dsw with VC6.0 or later version. After compiler and link, it will create the execute program of test_encode.exe or test_decode.exe, you can test it with following command.

test_encode test.pcm test.cod
test_decode test.cod test.pcm

To build the speech encoder (or decoder) sample programs on Linux, you only need rum **make** command. After you successfully finished make command, you can run **make run** to test encoder and decoder.

EasyG728 API FUNCTIONS

EasyG728_init_encoder	
Description	Initializes the memory needed by the encoding process. This function must be called prior to opening or re-opening a channel.
Syntax	#include "EasyG728.h" CODER_HANDLE EasyG728_init_encoder( );
Arguments	none
Returned value	Return a handle that represent an encode channel, this value will used at EasyG728_encoder and EasyG728_release_encoder

EasyG728_encoder		
Description	Encode an 80 words speech frame into a 20 bytes packed bit stream.	
Syntax	#include "EasyG728.h" bool EasyG728_encoder(CODER_HANDLE hEncoder, short *speech, unsigned char *bitstream);	
Arguments	hEncoder	The coder handle returned by EasyG728_init_encoder
	speech	Input speech buffer containing one frame of 16-bit PCM speech data.

Bitstream   Output bit stream buffer containing packed bit stream.

Returned value   Return true if successful, return false if failed.

EasyG728_release_encoder

Description   release the memory allocated by the encoding process. This function must be called before you quit your program. If not, it will cause the memory leak.

Syntax   #include "EasyG728.h"

bool   EasyG728_release_encoder(CODER_HANDLE hEncoder);

Arguments   hEncoder   The coder handle returned by EasyG728_init_encoder

Returned value   Return true if successful, return false if failed.

EasyG728_init_decoder

Description   Initializes the memory needed by the decoding process. This function must be called prior to opening or re-opening a channel.

Syntax   #include "EasyG728.h"

CODER_HANDLE EasyG728_init_decoder( );

Arguments   None

Returned value   Return a handle that represent an decode channel, this value will used at EasyG728_decoder and EasyG728_release_decoder

---

EasyG728_decoder

Description   Decodes a 20 bytes packed bit stream into an 80 words speech frame.

Syntax   #include "EasyG728.h"

bool   EasyG728_decoder(CODER_HANDLE hDecoder, unsigned char *bitstream, short *synth_short);

Arguments   hDecoder   The decoder handle returned by EasyG728_init_decoder
bitstream   Input buffer containing packed bit-stream.
synth_short   Output buffer containing one frame of decoded 16 bits PCM.

Returned   Return true if successful, return false if failed.

value

EasyG728_release_decoder

Description	release the memory allocated by the decoding process. This function must be called before you quit your program. If not, it will cause the memory leak.

Syntax

#include "EasyG728.h"

bool   EasyG728_release_decoder(CODER_HANDLE hDecoder);

Arguments   hDecoder   The coder handle returned by EasyG728_init_decoder

Returned value   Return true if successful, return false if failed.

FAQs

Here are some frequently asked questions about the EasyG728.

Q — Is the implementation of G.728 interoperable with the other company's version?
A — The implementation of EasyG.728 is fully conform to ITU G.728, It can interoperate with other G.728 implementations.

Q — What type of speech input format is required?
A — Raw 16-bit mono PCM sampled at 8000Hz. Do not use .WAV files. They contain a header that will produce distortion at the start of a decoded audio sample because the encoder interprets the header as speech data.

Q — How can I convert my .WAV files to raw 16 bit mono PCM sampled at 8000 Hz?
A — Use an audio editing tool such as SoX - Sound eXchange. See home.sprynet.com/~cbagwell/sox.html for more information

Q — Can I get link on platforms other than Pentium or compatible?
A — The object code provided in this package is Microsoft Win32 and Linux x86 compatible. It is compiled for the Pentium family of processors. If you want to use EasyG728 on other platforms, you should buy the source code of EasyG728. Then you can compile and link.

Q — Is the EasyG728 codec able to handle multiple channels?
A — Yes, It can handle multiple channels. There is no limited.

Q — Is the EasyG728 codec free to use?
A — No, The version you get freely is a version only for test. If you want to use it in commercial, you must buy it from www.imtelephone.com. This version has the same function with the formal release version, but It can only run 60 hours continuously.

Q — How much does the EasyG728 codec cost?
A — The object code of Windows or Linux is $1000/year. The source code is $10000/year. You can buy it from www.imtelephone.com.