Easy G.726

Technical Document
Version 2.0－ Revision 2006-11-11

Table of Contents

Introduction

EasyG726 is an implementation of ITU G.726. EasyG726 support multiple channels concurrent. There is no limit in concurrent channels and it can up to thousands channels. EasyG726 has 4 different bit rates, it has 16kbps/24kbps/32kbps/40kbps coder that encodes/decodes speech signal. The coder operates on speech frames is based on each frame.

EasyG726 codec specifications	
Bit rate (kbps)	16,24,32,40
Speech sampling rate(Hz)	8000

EasyG726 has a binary release version on Windows and Linux. The source code of EasyG726 is written by C, so you can easily port it to UNIX, PPC,DSP, Vxworks or other operation system that support C.

PACKAGE CONTENTS

EasyG726.pdf	This document
EasyG726.lib	Win32 statically linkable library of G726 for Pentium and compatible processors.
libG726.a	Linux statically linkable library of G726 for Pentium and compatible processors.
EasyG726.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, only include one sample(2 bytes)

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):

EasyG726_encoder `infile bitstream`

EasyG726_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.

EasyG726 API FUNCTIONS

EasyG726_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 “EasyG726.h”

CODER_HANDLE EasyG726_init_encoder( );

Arguments None

Returned value Return a handle that represent an encode channel, this value will used at EasyG726_encoder and EasyG726_release_encoder

EasyG726_encoder

Description Encode a sample(16bits) into a bit stream.

Syntax

#include “EasyG726.h”

int EasyG726_encoder(CODER_HANDLE hEncoder, short *speech, unsigned char *bitstream, int in_coding, int bitsPerSample);

Arguments

hEncoder The coder handle returned by EasyG726_init_encoder
speech Input speech buffer containing one sample of 16-bit PCM speech data.
bitstream Output bit stream buffer containing packed bit stream.
in_coding:
1 AUDIO_ENCODING_ULAW ISDN u-law
2 AUDIO_ENCODING_ALAW ISDN A-law
3 AUDIO_ENCODING_LINEAR PCM 2's-complement (0-center)
bitsPerSample
2 16kbps
3 24kbps
4 32kbps

5 40kbps

Returned value Return 0 if successful, return –1 if failed.

EasyG726_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 “EasyG726.h”

int EasyG726_release_encoder(CODER_HANDLE hEncoder);

Arguments hEncoder The coder handle returned by EasyG726_init_encoder

Returned value Return 0 if successful, return –1 if failed.

EasyG726_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 “EasyG726.h”

CODER_HANDLE EasyG726_init_decoder( );

Arguments None

Returned value Return a handle that represent an decode channel, this value will used at EasyG726_decoder and EasyG726_release_decoder

EasyG726_decoder

Description Decodes a packed bit stream into a sample(16bits).

Syntax #include “EasyG726.h”

int EasyG726_decoder(CODER_HANDLE hDecoder, unsigned char *bitstream, short *speech, int out_coding, int bitsPerSample);

Arguments

HDecoder The decoder handle returned by EasyG726_init_decoder
bitstream Input buffer containing packed bit-stream.
speech Output buffer containing one sample 16 bits PCM.
out_coding:
1 AUDIO_ENCODING_ULAW ISDN u-law

2	AUDIO_ENCODING_ALAW	ISDN A-law
3	AUDIO_ENCODING_LINEAR	PCM 2's-complement (0-center)

bitsPerSample

2	16kbps
3	24kbps
4	32kbps
5	40kbps

Returned value Return 0 if successful, return –1 if failed.

EasyG726_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 “EasyG726.h”

int   EasyG726_release_decoder(CODER_HANDLE hDecoder);
```

Arguments hDecoder The coder handle returned by EasyG726_init_decoder

Returned value Return 0 if successful, return -1 if failed.

FAQs

Here are some frequently asked questions about the EasyG726.

Q — Is the implementation of G.726 interoperable with the other company’s version?
A — The implementation of EasyG.726 is fully conform to ITU G.726, It can interoperate with other G.726 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 EasyG726 on other platforms, you should buy the source code of EasyG726. Then you can compile and link.

Q — Is the EasyG726 codec able to handle multiple channels?
A — Yes, It can handle multiple channels. There is no limited.

Q — Is the EasyG726 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 EasyG726 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.