QCELP Implementation on TMS320C30 DSP Board

TMS320C30 DSP를 이용한 QCELP Codec의 실현

Kyong Ho Han*
한 경 호*

ABSTRACT

The implementation of the voice codec is implemented by using TMS320C30, which is the floating point DSP chip from Texas Instrument.

QCELP (Qualcomm Code Excited Linear Prediction) is used to encode and decode the voice. The QCELP code is implemented by the TMS320C30 C-code. The DSP board is controlled by the PC. The PC program tranfors the voice file from and to the DSP board, which is also implemented by C-code.

The voice is encoded by the DSP board and the encoded data is transferred to PC to be stored as a file. To hear the voice, the voice data file is sent to DSP board and decoded to synthesize audible voice. Two flags are used by both programs to notify the status of the operation. By checking the flags, DSP and PC decides when the voice data is transferred between them.

요 약

디지틀 이동통신에서 사용되는 음성 압축기술의 한가지인 QCELP를 TI사의 TMS320C30을 사용한 DSP 보드를 이용하여 구현하였다.

음성을 받아 QCELP 방식으로 압축하는 프로그램은 TI의 C 코드로 작성하여 DSP 보드의 RAM에 download하여 수행되도록 하였다.

PC에는 DSP 보드에서 생성된 voice 데이타를 받아 file로 저장하는 작업을 하게된다. 이것도 C 코드로 작성하였다.

외부 마이크로 입력된 음성신호는 A/D 변환을 거쳐 PCM 데이타가 된다.

PCM 데이타는 DSP에 입력되어 QCELP방식으로 압축된 음성 데이타 패킷이 된다. 이 패킷은 PC로 보내 file로 저장하게 된다.

음성을 듣고자 할 경우, 압축된 음성 데이타 패킷을 PC가 DSP보드로 보내어 QCELP방식으로 음성을 합성, 재생한다. 이것을 D/A변환을 거쳐 실지음성이 된다.

DSP보드와 PC는 각각의 프로그램 수행상태를 나타내는 software flag로서로 패킷 데이타를 주고 받을때 결정한다.

*Dept. of Electrical Engineering, Dan-kuk Univ.
접수일자 : 1995년 1월 9일

Ⅰ. Introduction

As the mobile cellular system is getting popular, the service capacity is required to be increased. The current analog mobile system is almost on the edge of the capacity limit and new digital mobile systems are developed to solve the service capacity problem. To meet the increasing service demands in the urban area, operating companies are moving from the analog cellular system to the digital cellular systems.

In the North America, Time Division Multiple Access(TDMA) and Code Division Multiple Access (CDMA) are developed and in Europe, Group Special Mobile (GSM) is developed. The digital cellular system can increase the capacity by 3 to 10 times.

To code the speech signal, the technology, they use various coding algorithms. TDMA uses 8K bps Vector Sum Excited Linear Prediction (VSELP), GSM uses 13K bps Regular Pulse Excited Long Term Prediction (RPE-LTP) and CDMA uses variable rate Qualcomm Code Excited Linear Prediction (QCELP).

The QCELP algoithm is one the CELP (Code Excited Linear Prediction) type coders and was introduced by the QUALCOMM [1].

In coding the speech by using the QCELP algorithm, most of the time is spent by the pitch search and codebook search procedures. So reducing the complexity of those two search techniques makes the biggest contribution to the complexity reduction of the QCELP algorithm. In this paper, we proposed a complexity reduction of the pitch and codebook search by using a recursive form in the energy term calculations.

To implement QCELP algorithm on the DSP, the modified QCELP C program is compiled and ran by the floating point DSP chip. The TMS 320C30 DSP board encodes the PCM data and then decoded to generate the new PCM data. This procedure emulates the QCELP codec by the DSP chip. We present DSP system configurations and the oper ation of the related software routines.

Ⅱ. QCELP Algorithm

The QCELP vocoder developed by QUALCOM is a variable rate speech coder applicable 8, 4, 2, or kbps, depending on the level op voice activity. The QCELP algorithm is based upon the general CELP structure. The overall block diagram of encoder is shown in Fig. 1.

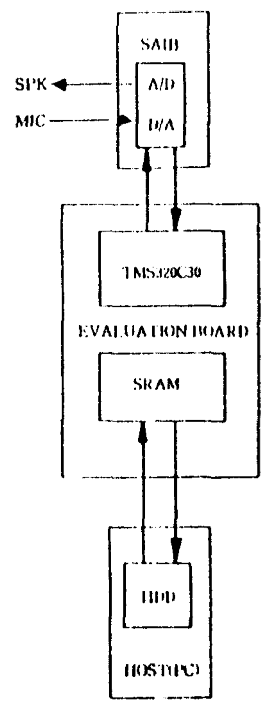

Fig. 1. Block Diagram of the System Configuration

The input speech is sampled at 8 kHz is broken down into 20 ms speech frames consisting of 160 samples. The linear predictive coding (LPC) filter coefficients with 10th order are calculated regardless of data rate selected. The LPC parameters are transformed into line spectral pair frequency (LSP) and quantized with the assigned bits according to the selected data rate. The LSPs are updated in each subframe through interpolation of LSPs in neighbor frames. A weighting filter $W(z)$ is used to reduce the loudness of the quantization noise. This filter is chosen to be

$$W(z) = \frac{A(z)}{A(z/\zeta)}$$

Where $A(z)$ is the formant prediction filter and ζ is given by 0.8. Within each frame, paramenters of pitch synthesized are updated a varying number of times, where the number of pitch parameters updates also depends on the selected data rate. Similarly, the codebook parameters are updated a varying number of times. The various parameters used for each rate is described in Table 1.

Table 1. Parameters Used for Each Rate

Parameter	Rate 1	Rate $\frac{1}{2}$
Framet for LPCm L_1	160	160
Bits per LPC Update	40	20
Pitch Subframe, L_p	40	80
Bits per Pitch Update	10	10
Codebook Subframe, L_c	20	40
Bits per Codebook Update	10	10
Parameter	**Rate $\frac{1}{4}$**	**Rate $\frac{1}{8}$**
Frame for LPC, L_1	160	160
Bits per LPC Update	40	20
Pitch Subframe, L_p	160	–
Bits per Pitch Update	10	–
Codebook Subframe, L_c	80	160
Bits per Codebook Update	10	6

Ⅲ. System Configuration

To implement QCELP codec by using the TMS 320C30 DSP chip [4], we used the C30 evaluation board, a SPIRIT30 DSP board from Sonitech Inc [5]. The DSP board has a TMS320C30 DSP operating at 60-ns execution time and 640K bytes of dual access fast SRAM banks and the board is installed in the 486-PC. The SRAM stores the QCELP executable codes downloaded form the host and the variables are also stored in the SRAM. The 486-PC is used as a host and the SRAM is accessed by both the PC and DSP board to pass the data between the PC and DSP board.

The DSP board receives and generates the PCM voice data and we used an audio interface board (SAIB) to generate the audible analog voice signal and to convert the analog voice signal to the PCM voice data. The SAIB contains the A-D converter and D-A converter. The voice signal from microphone is converted 16 bit PCM data and the PCM data is sent to DSP through the serial port. DSP generated PCM data comes out from the serial port and received by the SAIB, where the PCM data is converted into analog signal to output to speaker. PC is used as a host and stores the PCM data in the disk. PC and the DSP board communicates with each other by the SRAM on the DSP board.

Fig. 1 shows the block diagram of the system configuration.

Ⅳ. Software

Software is composed of a PC routine and a DSP routine. The PC routine is written in MicroSoft C and the DSP routine is written in TI C30 C code.

The PCM data is stored in the PC hard disk and PC routine moves the data between the PC hard disk and the DSP board. The DSP routine performs the QCELP codec operation. The QCELP program is written in TI C30 C code and the source code is compliled and optimized and downloaded into the DSP board.

8K word(16bit) buffer is allocated in the DSP SRAM and is used to move the data between the the PC and DSP. Two 16bit flags are used to indicate the status of PC an DSP.

We have PLAY task and RECORD task routine to heare the voice and to store the voice. The

PLAY task read the PCM data from the PC ahrd disk and send the data to SAIB and analog voice signal is generated from the speaker. The PLAY task is used to hear the PCM voice data in the PC hard disk.

The RECORD task receives the PCM data from the SAIB and send the data to PC to store in the hard disk. The RECORD task is used to record the voice in the PC hard disk.

The implementation software of the QCELP algorithm is the PCLOOPBACK task.

For the PCLOOPBACK task, QCELP executable code is downloaded into the DSP program memory so that the DSP can process the QCELP algorithm.

The QCELP executable code encodes the PCM data and then immediately decode the encoded result to create the new PCM data.

At the beginning of the routine, the PC reads data from the hard disk and check the status of the DSP. PC waits until the DSP is ready to accept the data from the PC. The status of the DSP is indicated by the DSP flag and the flag is set by the routine.

When the flag indicates that the DSP is ready to accept the data, PC places data into the buffer in the DSP SRAM and set the PC flag to note the DSP that the buffer is full.

After the DSP set the DSP flag to indicate that it ready to accept the new data, the DSP waits for the PC to finish downloading the PCM data into the DSP SRAM. When the flags indicate that the buffer is full, the DSP starts the QCELP codec processing with the downloaded PCM data. The DSP performs the QCELP algoithm by encoding the PCM data in the buffer and then decode the result to generate the new PCM data. The new PCM data is placed back in the same buffer in the DSP RAM. When the DSP finishes the QCELP algorithm, the DSP set the DSP flag to indicate that the DSP is ready to send the new PCM data to PC.

On the PC side, after sending the data to the DSP, the PC waits for the DSP until the DSP is ready to send the new PCM data. At the end of the QCELP routine on the DSP side, the PC detects that the DSP finished the QCELP processing and the new PCM data is placed in the buffer and then PC uploads the data from the DSP and stores into the hard disk. After PC finishes the uploading, it sets the PC flag to indicates that the PC finished the uploading.

On the DSP side, after finishing the QCELP processing, DSP waits for the PC until the PC finishes the uploading. When the DSP detects the end of data uploading, it sets DSP flag to indicate that the DSP is ready to accept next PCM data from the PC.

Then both the PC and DSP go to the start of each routine.

⟨Fig. 2⟩ and ⟨Fig. 3⟩ show the flow charts of the PC routines and the DSP routines.

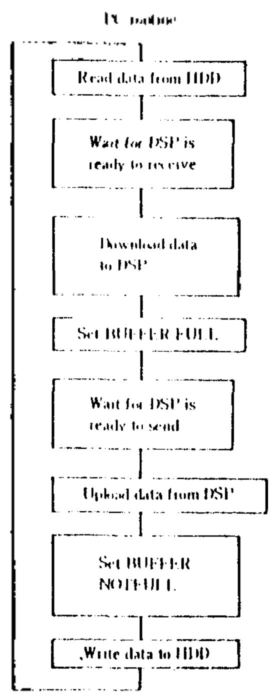

Fig. 2. Flow of the PCLOOPBACK routine on the PC

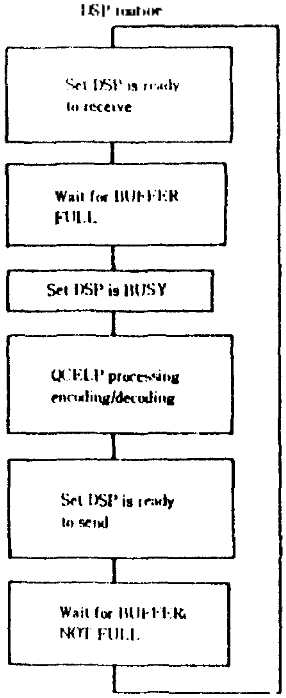

Fig. 3. Flow of the PCLOOPBACK routine on the DSP

V. Summary and Conclusions

In this paper, we handled the complexity reduction of pitch search and codebook search and the implementation of the QCELP codec on the TMS320C30 floating point DSP evaluation board. The PCM data is on the PC side and the QCELP program resides on the DSP board. The PCM data is moved form the PC to DSP and encoded and becoded to generate the new PCM data which is moved back to the PC.

For the further study, the QCELP program can be optimized in the assembly language level and time can be reduced more.

REFERENCES

1. QUALCOMM INC. *TR45 EIA/TIA interim standard* QUALCOMM Inc, 1993.
2. J.H. chen and A. Gersho, "Real-Time Vector APC speech Coding at 4800 BPS with Adaptive Postfiltering," *Proc. ICASSP*, pp. 2185-2188, Apr. 1987.
3. D.Lin, "Speech Coding Using Efficient Pseudo-Stochastic Block Codes," *Proc. ICASSP*, pp. 1354-1357, Apr. 1987.
4. Texas Instrument, *TMS320C30 user guide* Texas Instrument, 1990.
5. Sonitech Intl. Inc., *Spirit-30 System Technical Reference* Sonitech Intl. Incd., 1990.

▲한 경 호(Kyong Ho Han) 1959년 6월 25일생
1982년 2월 : 서울대학교 전자공학과 졸업(학사)
1984년 2월 : 서울대학교 대학원 전자공학과 졸업(석사)
1992년 8월 : 미 Texas A&M Univ. Dept. of E.E 졸업(Ph.D)
1984년~1985년 : 삼성 휴렛팩커드(연구원)
1985년~1997년 : 한국전기통신공사(전임연구원)
1992년~1993년 : 한국전자통신연구소 이동통신 개발단(선임연구원)
1993년 8월~현재 : 단국대학교 공과대학 전기공학과 전임강사