European Conference on
Speech Technology
Edinburgh, Scotland, UK
September 1987

HARDWARE IMPLEMENTATION OF A MULTI-RATE REAL-TIME SPEECH CODEC

Bruce Bukiet*, Roderick J. Ragland°, Dr. John Damoulakis°

ABSTRACT

This paper describes a multi-rate voice coder/decoder which is totally self contained and portable, requiring only an outlet to a standard North American 110 VAC power line. Virtually everything needed to perform speech coding experiments is contained within the instrument case.

This system employs the INTEL 8751 micro-controller and two digital signal processors: the Texas Instrument TMS320 for data acquisition and the AT&T DSP32 for speech coding. The speech coding repertoire includes the following algorithms:

- o 400 bps vector quantized LPC
- o 800 bps vector quantized LPC
- o 2800 bps scalar quantized LPC
- o 10500 bps residual excited LPC
- o 32000 bps ADPCM

Both the hardware and software configurations are described, along with measured execution times and program and data memory usages. A review of the coding algorithms is presented, with a brief summary of the system's operational parameters.

INTRODUCTION

During the past few years, single chip signal processors have undergone significant advancement. Several manufacturers provide a range of digital signal processing (DSP) chips which can perform both fixed and floating point numerical operations. The advent of such DSP chips has made it feasible to cost effectively implement complex DSP algorithms in real time. However, there are significant differences in the architecture, capabilities and programming strategy as compared to previously available programmable devices, such as microprocessors or multiplier-accumula-tors. The implementation of complex signal processing algorithms with DSP chips requires careful planning and consideration in both hardware and software architectures. Those considerations are important in exploiting the full capabilities of these new signal processors.

This paper summarizes the structure of a real time multi-rate speech coder/decoder developed at Gould, Ocean Systems Division. This multi-rate speech processor, from here on referred to as the Speech Compression System (SCS), is a portable unit which can perform several speech compression algorithms in real-time. The aspects that play a role in real time implementation of the algorithms is presently along with the high level hardware architecture, software programming strategy, memory management, and inter-processor synchronization. The paper concludes with a discussion of real-time performance and Diagnostic Rhyme Test (DRT) results.

SPEECH CODING ALGORITHMS

The SCS implements four compression techniques described below as well as their respective synthesis programs. Each has the capability of operating on input speech signals whose sampling rate is either 8 KHz or 10 KHz where each sample is stored as a 16 bit data word. Three of the four algorithms are LPC based. The number of poles used to describe the LPC filter is 12 for 8 KHz sampled input signals and 14 for 10 KHz sampling.

In addition to the analysis and synthesis programs which implement the four algorithms, the SCS supports a random bit error generator which allows the operator to inject bit errors into the compressed data stream to simulate transmission channel errors. The available error rates are 0, 1/1000000, 1/100000, 1/10000, 1/1000 and 1/100.

VECTOR QUANTIZED LPC (VQ-LPC)

The basic premise behind vector quantization (VQ) of speech is that a relatively small selection of LPC spectral template vectors can represent a much larger or infinite set of vectors within a specific degree of error. This representative set of vectors is stored in a codebook which the encoder searches for a best match to an input source vector. The index of the chosen codevector is transmitted along with necessary side information to the decoder which extracts the proper template from the codebook for synthesis. Details concerning VQ codebook design and efficient search techniques are discussed in reference 1.

The LPC side information which consists of pitch, gain and voicing, can be handled in a number of ways. In the present system, pitch and voicing are combined into a single parameter and quantized using a tree-search coding algorithm. The log of the gain signal is handled in the same manner.

The SCS is capable of accepting VQ codebooks containing 1024, 2048 or 4096 codevectors which yield 10-bit, 11-bit or 12-bit codewords per input source vector. The tree coder yields 2 bits per gain codeword and 2 bits per pitch/voicing codeword. As a result, the VQ-LPC algorithm compresses speech to 700, 750 or 800 bps based on a fixed frame rate of 50 Hz (50 frames/sec.).

A further reduction of transmission bit rate is achieved through a simple frame repeat strategy. The frame repeat function involves transmitting a single bit (i.e. a binary 1) if the current frame is spectrally similar to the previous frame; in this case the codevector chosen for the previous frame also represents the current frame. If the spectral similarity of two consecutive frames is beyond a predetermined threshold, the full codevector index is transmitted with a leading zero attached to indicate no frame repeat. Depending on how the threshold is set, the frame repeat option reduces the VQ-LPC average bit rate to as low as 400 bps.

* Diversified Technology, 511 W. Golf Road, Arlington Heights, IL 60005

° Gould, OSD, 6711 Baymeadow Drive, Glen Burnie, MD 21061

10.21437/ECST.1987-161

LINEAR PREDICTIVE CODING (LPC)

The SCS implements a standard LPC model which differs from the VQ-LPC algorithm described above only in the methods of parameter quantization. Both systems employ an autocorrelation with center clipping algorithm for pitch and voicing detection and both obtain the gain term as a consequence of the spectral determination.

The standard LPC algorithm performs scalar quantization of the gain and pitch/voicing parameters via look-up table where the VQ-LPC employs a differential coder. The LPC reflection coefficients are also scalar quantized using look-up tables rather than a vector quantizer. The per-frame bit allocations for the parameters are as follows: 6 bits for pitch/voicing, 5 bits for gain, 5 bits for each of the first two reflection coefficients, 4 bits each for the next 5, and 3 bits each for the remainder. As a result, the transmission bit rates based on a 50 Hz frame rate for the LPC algorithms are 2800 bps for a 8 KHz input sampling and 3100 bps for 10 Khz input sampling.

RESIDUAL EXCITED LINEAR PREDICTION (RELP)

The alternative to the pitch-excited LPC based systems described above is a residual excited LPC system. The two systems differ in the nature of the excitation function that serves as input to the LPC filter. The pitch-excited model employs either pitch pulses or white noise (depending on the voicing parameter) as the driving function for the spectral model while the RELP system substitutes the LPC residual error signal as the excitation function. Since the linear prediction process approximates a speech signal sample as a weighted summation of past signal samples without consideration of a separate excitation function, the residual error represents a better approximation of the true excitation source.

Since the LPC residual contains much more information than the gain and pitch/voicing signals used in pitch-excited LPC, significant reduction of bandwidth is essential. This goal is achieved by band-limiting the residual to 400 Hz and encoding the baseband using an Adaptive Delta Modulator (ADM). At the receiver, the baseband residual is decoded and the missing high frequency band is regenerated by passing the decoded baseband through a non-linear distortion device to generate the upper harmonics. Finally, the residual spectrum is flattend with an adaptive whitening filter. Further detail regarding RELP speech compression can be found in reference 2.

The parameters that comprise a frame of RELP coded speech consist of a gain term, M LPC reflection coefficients and N codewords resulting from the ADM of the baseband residual where M is the LPC filter order and N is the size of a speech frame (160 or 200 samples based on a frame rate of 50 Hz with an 8 KHz or 10 KHz sampling respectively). The bit allocation for the LPC coefficients and the associated gain term are the same as those listed above for the standard LPC program while the ADM yields 1 bit per sample. These allocations result in transmission bit rates of 10500 bps for 8 KHz sampling and 12800 bps for 10 KHz sampling.

ADAPTIVE DIFFERENTIAL PULSE CODE MODULATION (ADPCM)

Many schemes for reducing the transmission bit rate of Pulse Code Modulated (PCM) speech by a factor of two or more with little effect on signal-to-noise ratio have been developed and are collectively known as Adaptive Differential Pulse Code Modulation (ADPCM). The principle here is to quantize the difference between a current signal sample and its predecessor rather than the sample amplitude itself. Since speech signals are not likely to change by more than a fraction of the full quantizer scale during one sampling interval, fewer bits are necessary for each codeword. In order to accommodate both rapidly and slowly changing waveforms, the quantizer step size is made adaptive. Various ADPCM algorithms are described in reference 3.

The SCS implements an ADPCM program which allocates either 3 bits-per-sample or 4 bits-per-sample thus yielding bit rates of 24,000 bps or 32,000 bps for 8 KHz input sampling and 30,000 bps or 40,000 bps for 10 KHz sampling.

HARDWARE DESCRIPTION

The SCS operates by digitizing analog voice signals from either a microphone or tape recorder. After the analog-to-digital conversion process, the data is presented to the speech processor to perform any one of four compression techniques. The compressed data is then stored into static RAM. Upon command, the compressed speech will be read from RAM and sent to the speech processor for decoding and synthesis. The decoded speech is then converted to analog and played back through the speaker housed within the unit. The SCS is designed to be a totally self-contained portable unit, as indicated in Figure 1.

MULTI-PROCESSOR ARCHITECTURE

A block diagram of the SCS hardware is shown in Figure 2. The system is based upon a multi-processor architecture: the AT&T DSP32 signal processor, Texas Instruments TMS320C20, and INTEL 8751 micro-controller. The SCS multi processor architecture provides a high degree of parallel processing where each specific processor generally performs a class of functions which it is designed to perform best.

All analog-to-digital and digital-to-analog functions are controlled by the TMS320C20. This processor performs low-pass filtering of the input signal to alleviate aliasing effects during data acquisition, automatic gain control of the input signal, and low-pass filtering of the processed signal to eliminate spurious frequency components inherent to the algorithms of the system. All algorithm support functions are performed by the INTEL 8751. These support functions consists of interfacing and controlling the two signal processors, collecting and compacting compressed speech codes from the DSP32 and interfacing with all specially designed support hardware. The support hardware includes UV-EPROMS for storage of VQ-LPC codebooks and RAM used for mass storage of compressed speech. Front panel displays and control switches interfacing with the user are also controlled by the 8751. Communication between the 8751 controller and the data acquisition processor consists of an interrupt passed to the controller indicating completion of a sampled data frame. This strategy forms the basis of the timing for the entire system.

The DSP32 was incorporated to perform all direct algorithmic calculations because of its speed and 32 bit floating point precision. This processor has an extensive amount of processing power, on board memory and interfaces directly to the micro-controller. All communication between the AT&T DSP32 and the INTEL 8751 micro-controller is done through the signal processor's parallel port. After initialization, the DSP32 monitors its parallel data port waiting for the micro-controller to issue a command. If the user selects a compression algorithm for deployment, the controller issues the proper command and the DSP32 initializes the necessary parameters and performs the compression. Upon completion of each frame, the DSP32 writes compressed data into the parallel data port. Each time the parallel data port is written by the DSP32, the controller reads the compressed data codewords and stores them into compressed data RAM. This process continues until a stop command is issued by the micro-controller. The synthesis operation is handled in much the same manner.

MEMORY ORGANIZATION

Since the unit was designed to support a wide variety of coding algorithms, an ample amount of memory was provided to implement these algorithms. However, additional hardware and memory had to be developed to support VQ-LPC with its large memory requirements. To implement VQ-LPC with 4096 size codevector codebooks, at least 524,288 bytes of storage had to be provided. To complicate issues even further, this memory had to be accessible to the DSP32 operating at 250 nanosecond instruction cycles requiring memory access times of the order of 70 nanoseconds. Since the density and price of high speed memories preclude the use in this type of architecture, another alternative was taken. A provision was made to add slower and denser EPROMS in conjunction with high speed dual port memory to address the codebook memory problem. The EPROMS would contain the codebooks of individual speakers organized in a 2 level 64 x 64 tree, with vector lengths up to 16 coefficients. Specially designed hardware is designed to download (or page) a portion of the code vector codebook from slower EPROM to fast RAM which is directly interfaced to the DSP32 for processing. The 8751 micro-controller is responsible for directly initiating the download process and signaling its completion to the DSP32 so that it can complete its codebook searching.

Dual port RAM is also used for memory that is shared between the TMS320C20 and the DSP32. As mentioned earlier, the TMS320C20 ships blocks of data to the DSP32 for processing and receives data from the DSP32 for analog conversion. Dual-Ported memory is used as a medium for efficiently transporting large blocks of data between the two processors.

PERFORMANCE

Evaluation of the speech codec system loading needs to be done from the perspectives of both time and memory allocation. As described earlier, the speech codec contains a digital signal processor whose program and data space are common. The lower area of program memory contains all the speech coding and bit error generation software as well as the necessary look-up tables (not including the codebooks for VQ-LPC). The four algorithms implemented occupy approximately 13K bytes while the look-up tables occupy 10K bytes. Program memory usage is just under 23k bytes which is approximately 70% of the total program memory. It is hard to describe exactly how much of the memory usage is attributed to each algorithm since all share common subroutines.

The most computationally intensive algorithm implemented in the speech codec is the compression half of the VQ-LPC. The compression or synthesis of a single frame must be completed within 20 milliseconds. The VQ-LPC compression operation takes 10 milliseconds to complete which includes conversion of the input sampled data to floating point, ali necessary filtering, codevector download and search, gain coding, pitch detection and coding, bit error generation (if select-ed), and output of the compressed data codewords.

REAL-TIME PERFORMANCE

Throughout the design, a rather conservative approach was taken in regards to processor time loading. In particular, loading of the DSP32 processor for numerical calculations. Execution times for the DSP32 processor are listed in Table 1 for compression and Table 2 for synthesis. From the tables, the most computational intensive algorithms are VQ-LPC compression and RELP synthesis programs. In VQ-LPC compression, the vector quantization table search is tree-struc-tured to limit the number of codebook entries to compare. Though an optimal full search of the codebook would have been desirable, our simulation results has shown little or no perceptible difference between a tree-structured and full search method. For RELP synthesis, the most time consuming numerical calculations is to compute a spectral whitening filter from the decoded residual signal. The system was originally designed such that the average loading factor of any of the algorithms would not exceed 60%. As illustrated in Tables 1 and 2, the system is well under our design requirement.

INTELLIGIBILITY TESTS

At the time of this writing, formal intelligibility testing of the real-time speech codec is in progress. Preliminary results indicate that VQ coding of speech at 800 bps (i.e. 4096 vector codebooks) yields Diagnostic Rhyme Test (DRT) scores consistent with those published elsewhere (see reference 4).

REFERENCES

1. R.M. Gray, "Vector Quantization", IEEE ASSP Magazine, PP. 4-29, April 1984

2. C.K. Lin and D.T. Magill, "The Residual-Excited Linear Predion Vocoder with Transmission Rate Below 9.6 Kbits/s", IEEE Trans. Comm., pp. 1466-1472, December 1975

3. N.S. Jayant and P. Noll, "Digital Coding of Wareforms", Prentice-Hall, Englewood Cliffs, NJ, 1984.

4. D.Y. Wong, B.H. Juang and A.H. Gray, Jr., "An 800 bit/s Vector Quantization LPC Vocoder", IEEE Trans. Accoust., Speech, Signal Processing, pp 770-780, October 1982.

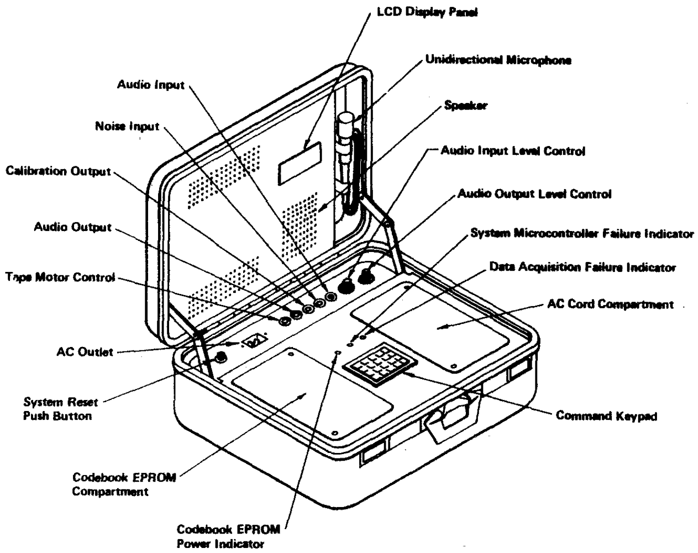

Figure 1. Picture of SCS Instrument Case

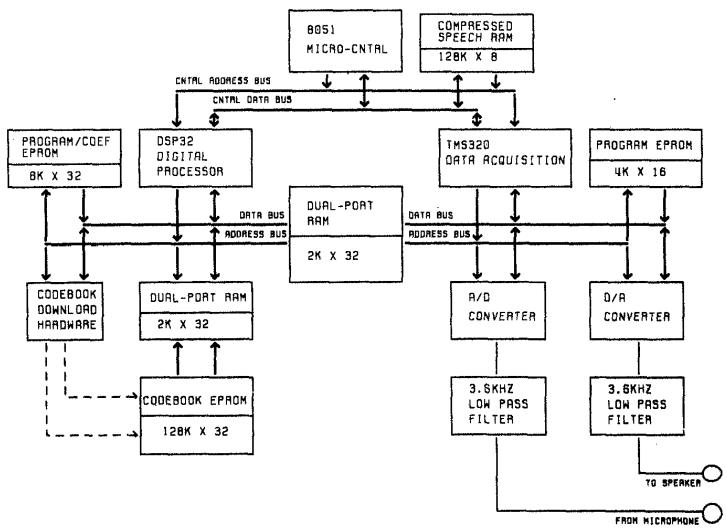

FIGURE 2. HARDWARE BLOCK LEVEL ARCHITECTURE

TABLE 1

Execution time of DSP compression routines.

Average execution times for each 20 ms frame.

Compression Algorithm	Execution Time (Milliseconds)	% Loading
VQ	10.291	51.45
RELP	3.86	19.3
LPC	6.122	30.61
ADPCM	3.85	19.25

TABLE 2

Execution time of DSP compression routines.

Average execution times for each 20 ms frame.

Synthesis Algorithm	Execution Time (Milliseconds)	% Loading
VQ	6.608	33.04
RELP	9.671	48.35
LPC	6.047	30.23
ADPCM	3.903	19.52