

HALF-RATE GSM VOCODER IMPLEMENTATION ON A DUAL MAC DIGITAL SIGNAL PROCESSOR

Mohit K. Prasad Paul D'Arcy Arup Gupta Marc S. Diamondstein Hosahalli R. Srinivas

Wireless and Multimedia, Microelectronics Group, Lucent Technologies, Allentown, PA 18103, USA

ABSTRACT

The Global System for Mobile (GSM) communications uses a 13 Kbps vocoder which expands to 22.8 Kbps after channel coding. To increase the user capacity the half-rate channel has a gross transfer rate of 11.4 Kbps. The vocoder for the half-rate channels operates at 5.6 Kbps. The computational requirements of a half-rate vocoder and other necessary services required design of an entirely new digital signal processing architecture geared towards 1-D signal and speech processing. The architecture is characterized by Very Large Instruction Word (VLIW) and two multiply-accumulate (MAC) units. Other enhancements of the hardware allow an efficient implementation of the half-rate GSM vocoder. This paper describes the architecture and compares the vocoder performance with existing implementations.

1. INTRODUCTION

The GSM standard is a mobile telephony standard in Europe for cellular phones operating in the 900 MHz band [1][2]. GSM permits encoded speech and user data to be carried over a mixture of full and half-rate channels. By using half-rate channels exclusively, the number of users on the network can be doubled. Speech is encoded in the full-rate channel at 13 Kbps becoming 22.8 Kbps after channel coding. Similarly, the half-rate encoder operates at 5.6 Kbps becoming 11.4 Kbps after channel coding. Typically, the channel codec and the vocoder are implemented in software on a programmable processor.

When this project was initiated in 1994 the baseband processing requirements of the half-rate GSM (HRGSM) digital terminal provided a significant obstacle to implementation on existing digital signal processor (DSP). The problem required design of an efficient co-processor to off-load the computationally intensive speech coding functions and address the needs of future signal processing applications such as echo-cancellation, noise cancellation, channel coding and decoding, etc.. The Voice Coding Processor (VCP) was specifically directed towards meeting those needs. The HRGSM vocoder provided the platform for development and testing of the design.

2. ARCHITECTURE

The VCP is a 16-bit fixed point processor, i.e. the data is a 16-bit integer, usually represented as a 2's complement

number. It is a Very Large Instruction Word (VLIW) machine with minimal control functions. The primary design objective was to speed the execution of the signal processing function which would normally have run on a DSP from Lucent's 16xx family of 16-bit processors. The result is a data-path geared towards efficient signal processing. There is no instruction decode unit, no I/O unit, and its interrupt handling capability is minimal but efficient. The VCP operates as a co-processor in conjunction with a controller, which may be a C-programmable RISC engine or another DSP. The DSP takes care of all control and I/O functions. The interface to the DSP is also kept very simple. Data is shared through common memory. The VCP has a four stage pipe-line. Efficient design features ensure no pipeline hazards.

The VCP data path is shown in Figure 1. Some key features of the architecture are listed below.

2.1. Dual MAC Units

The VCP consists of two multipliers and two Arithmetic Logic Units (ALU's). The 16x16 2's complement multipliers are specially designed for low power [3]. They can also perform 15-bit unsigned multiplication. The two ALU's can perform arithmetic operations on three operands at a time and logical operations on two operands. In addition, there is barrel-shifting unit (BSU), shared by both sides, to perform shifting functions.

A distinguishing feature of the VCP data path is the close interaction of the two MACs as opposed to having two identical units operating in parallel. For example, the output of the multiplier from side A (or B) can be input to the ALU from either side A or B or both. Memory access from either side is available to any of the multipliers or the ALUs. Complex multiplications, double precision operations, and many vector operations can be performed on this architecture at their theoretical optimum limits.

2.2. High Memory Bandwidth

Like many DSP's based on the Harvard architecture the VCP has two memory banks to store data and coefficients. In the VCP there are two memory banks devoted solely to data. There is a third memory bank for instructions. The two data banks have a 16-bit linear addressing space of 16-bit words each. The addressing unit of the two banks are independent with six 16-bit pointers each. The two data banks can be accessed simultaneously. The access can either be a single word or a double word (32-bit) access. In

addition, the pointers have some unique addressing features. These features provide the VCP with the large memory bandwidth.

2.3. Very Large Instruction Word

The VCP datapath is controlled by a minimally encoded horizontal instruction word. It provides the VCP with orthogonal control of the different compute elements in the data path. Each instruction is 144 bits wide. The instruction allowed the programmers to operate the different compute units in parallel for maximal utilisation of the hardware.

2.4. Three Input ALU's

The VCP can add or subtract three operands simultaneously and perform logical operations on two operands. The two ALU's operate independently. The output of the multipliers from sides A and B go to both ALU's. Three input addition also permits on-the-fly rounding and on-the-fly saturation. Because the HRGSM vocoder permits saturation arithmetic only, this leads to significant reduction in cycles in the vocoder implementation.

2.5. Accumulator Register File

To mitigate the effect of temporary storage and retrieval the VCP contains sixteen 40-bit accumulators. This register file is accessed by a set of six pointers with the ability to implement circular buffers. Direct as well as indirect access (using pointers) of the accumulators is possible.

2.6. Zero Overhead Looping, Function Calls, and Context Switch

The sequencing field of the VCP instruction is reserved for indicating the next address. Because, the VCP instruction requires no decoding the next instruction address is available at the same time as data path operations are performed. Hence, no cycles are spent exclusively for address computation. Loops, functions calls and context switches can therefore be performed with zero overhead.

2.7. Barrel Shifting Unit (BSU)

The VCP contains a 40-bit barrel shifter which performs various shifting functions. A novel feature is the ability to shift in a "0" or a "1", a feature used in implementing the Viterbi decoder effectively.

2.8. Loop Counters

The VCP contains four 16-bit loop counters. The loop counters have an auto-reload feature which eliminates taking an extra cycle to load the loop counter during nested loops.

2.9. Triangular Addressing Feature

In addition to implementing circular buffers, the VCP can also implement buffers whose size changes as per algorithm requirements. This feature called *triangular addressing* [4] is useful in implementing speech coding algorithms like the Levinson-Durbin's algorithm. This feature can also be used for implementing a moving data window.

2.10. Miscellaneous

The VCP also has other features, too numerous to list in this document, which make signal processing algorithms

easier to compute. The ALU for example, supports add-compare-select (ACS) operations, single cycle per quotient-bit divide operations, single cycle if-then-else operations, and single cycle butterfly operations.

The large instruction width allowed the designers to provide the various features which permitted parallel operations in an extremely powerful data path. The efficient use of the data path produced very compact code. As the discussion in the following section indicates, the memory required was no larger than that used in an existing application.

3. HRGSM FIRMWARE

The HRGSM voice coder is a Vector Sum Excited Linear Predictive (VSELP) analysis-by-synthesis vocoder [1][5][6][7][8]. Analog speech is sampled at 8 KHz, 13-bits/sample, into 20ms frames. Each frame is divided into four sub-frames of 40-bits. The VSELP algorithm compresses each frame into 112 bits at a ratio of nearly 20:1.

The speech coder models the human vocal tract by a Linear Prediction (LP) filter, specified by ten coefficients (LPC's). Given the inputs to this filter and the filter coefficients it is possible to reconstruct the human voice [6]. The speech encoder finds the best coefficients and vector quantizes. It also detects if the speech is voiced (quasi-periodic) or unvoiced. If the speech is voiced the encoder computes the period or lag. Based on the voiced-unvoiced decision the encoder finds the appropriate quantized input which will minimize the error between the reconstructed and the sampled speech. The encoder also finds and quantizes the energy content of each frame. The quantization process is very compute intensive because it requires exhaustive search of the vector-quantization tables, or code-books, some of which have as many as 512 entries. Only the indices of the code-book entries are transmitted. The quantized values of the coefficient, lag, and energy parameter are chosen in a similar manner. The encoder transmits these indices and the quantized LPC's. At the receiving end the decoder looks up the tables of coefficient, lag, inputs, and energy values and synthesizes human speech sub-frame by sub-frame. In the past, all these computations were performed off-line on a main-frame computer [8]. Today a \$25.00 DSP has to do the same in real-time!

To have uniformity between implementations ETSI has specified a set of input test vectors and reference outputs. The outputs of any vocoder implemented must match the reference results bit-per-bit [1]. It unifies the standard, but requires attention to implementation details which may not even be perceptible. Processing requirement is the added cost for this compromise.

At the top level, the three most compute intensive routines of the encoder are: A. LPC computation and quantization (routine Aflat - AF), B. Determining the lag for voiced signals and quantization of same (Open-loop lag search - routine OLLS), and C. Sub-frame analysis i.e. adjusting the LPC values, finding the quantized inputs to the LPC, and the short term lag value for each of the four sub-frames (routine SF). These three routines together account for 85% of the computational needs of the encoder. The encoder itself takes about 85% of the vocoder. The rest is taken by

Table 1. Comparing encoder performance with and without routines aflat, open-loop lag search, and sub-frame analysis off-loaded to the VCP.

Resources	DSP	DSP+VCP
RAM	4K words	5.5K words
ROM (program)	23K words	27.5K words
ROM (data)	8K words	7.6K words
DSP MIPs (Max/Avg)	25.6/21.6	4.1/4.0
VCP MIPs (Max/Avg)	-	10.8/8.0

Table 2. Comparing performance of key encoder modules on the DSP and the VCP. Maximum/average performance are compared.

Module	DSP	VCP	Speed
Aflat (AF)	8.0/7.5	3.1/2.9	2.6/2.6x
Lag search (OL)	5.5/2.8	2.5/1.1	2.2/2.5x
Sub-frame Anal. (SF)	8.5/7.3	5.2/4.0	1.6/1.8x
AF+OL+SF	22.0/17.6	10.8/8.0	2.0/2.2x

the decoder. So the three routines constitute about 75% of the entire vocoder.

The routines Aflat (AF), Open-loop lag search (OL), and sub-frame analysis (SF) were written and verified on a VCP simulator. The VCP was operated as a coprocessor for the Lucent's DSP1627 processor. The routines other than those listed above were run on the DSP1627. Table 1 compares the performance of the HRGSM vocoder running the three routines on the DSP1627 compared to the same three routines running on the VCP. In Table 2 the DSP and VCP performances are compared on a routine-by-routine basis. The overall VCP performance is more than 2.0x that of the DSP. Table 3 compares their program sizes. The VCP requires about 30% more ROM.

At lower levels the relative improvement on a VCP is far larger. Table 4 lists some of the lower-level functions and the number of cycles required for them on the DSP and the VCP. With further improvement of the hardware and software the VCP will be at least 15% more efficient. If the entire vocoder was implemented on the VCP today then one would require no more than 14 MIPs. In the future, this is likely to come down to 12.5 MIPs. Current program size is about 30% over that of the DSP. It is reasonable to expect that this difference will decrease to 15% with improvements in the VCP.

4. CONCLUSIONS

At Lucent, designers have broken away from the traditional single MAC complex instruction word processor architecture. The dual-MAC architecture, with close interaction between the two MAC's and independent control of the two,

Table 3. Program ROM usage of the key encoder modules. Each word is 16-bits wide.

Module	DSP	VCP
Aflat (AF)	4.0K	3.3K
Lag search (OL)	3.0K	5.2K
Sub-frame Anal. (SF)	5.4K	7.7K
AF+OL+SF	12.4K	16.2K

Table 4. DSP vs. VCP performance on some lower level modules.

Module	DSP	VCP	Speed
LPC computation	48602	10850	4.5x
LPC quantization	101291	49783	2.0x
Voice Activity detection	3997	1101	3.6x
IIR Filter	806	439	1.8x
Code vector construction	1004	390	2.6x
Code-book search	23200	7200	3.2x
Decorrelate code vectors	5400	1190	4.5x

with a large instruction word exploits parallelism inherent in many algorithms. In addition, by separating the control and signal processing functions the overall performance of a VCP-based signal processing system is improved. The overall increase in speed of the vocoder was by a factor of 2.0x with about 30% increase in the code size. Although called the Voice Coding Processor, the VCP can be used for different signal processing application with as much as 4x improvement in performance for specific algorithms. It is expected that the VCP performance can be improved further by modest changes in hardware and software.

5. ACKNOWLEDGEMENTS

The authors thank other members of the VCP development team, viz. S. J. Bachorski, R. Chatterjee, R. Kolagotla, H. S. Lau, W. V. Liu, C. R. Miller, S. Misra, B. Ng, L. Sankaranarayanan, P. Srivastava, M. E. Warner, and J. L. Winters, for their effort. They are also grateful to A. Fisher and J. Boddie for encouragement and support.

REFERENCES

- [1] M. Mouly and M-B. Pautet, *The GSM System for Mobile Communications*, M. Mouly et Marie-B Pautet, Papiseau, France, 1992.
- [2] *European Digital Cellular Communications System: Half-Rate Speech (GSM 6.20)*, ETSI 06921 Sophia Antipolis Cedex, France, 1995.
- [3] Ravi Kolagotla, Hosahalli R. Srinivas, and Jeffrey Burns, *VLSI Implementation of a 200-MHz Left-to-Right Carry-Free Multiplier in 0.35 μ m CMOS Technology for Next-Generation DSP's*, Paper submitted to CICC, Santa Clara, CA, May 5-8, 1997.
- [4] Mohit K. Prasad, *Triangular Addressing Scheme*, patent application, 1995.
- [5] Raymond Steele, *Mobile Radio Communications*, Pentech Press, London, UK, 1992.
- [6] J. D. Markel and A. H. Gray, Jr., *Linear Prediction of Speech*, Springer-Verlag, New York, NY, 1976.
- [7] L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals*, Prentice-Hall, Englewood Cliffs, NJ, 1978.
- [8] B. S. Atal, V. Cuperman, A. Gersho, eds., *Speech and Audio Coding for Wireless Network Applications*, Kluwer Academic Publishers, 1993.

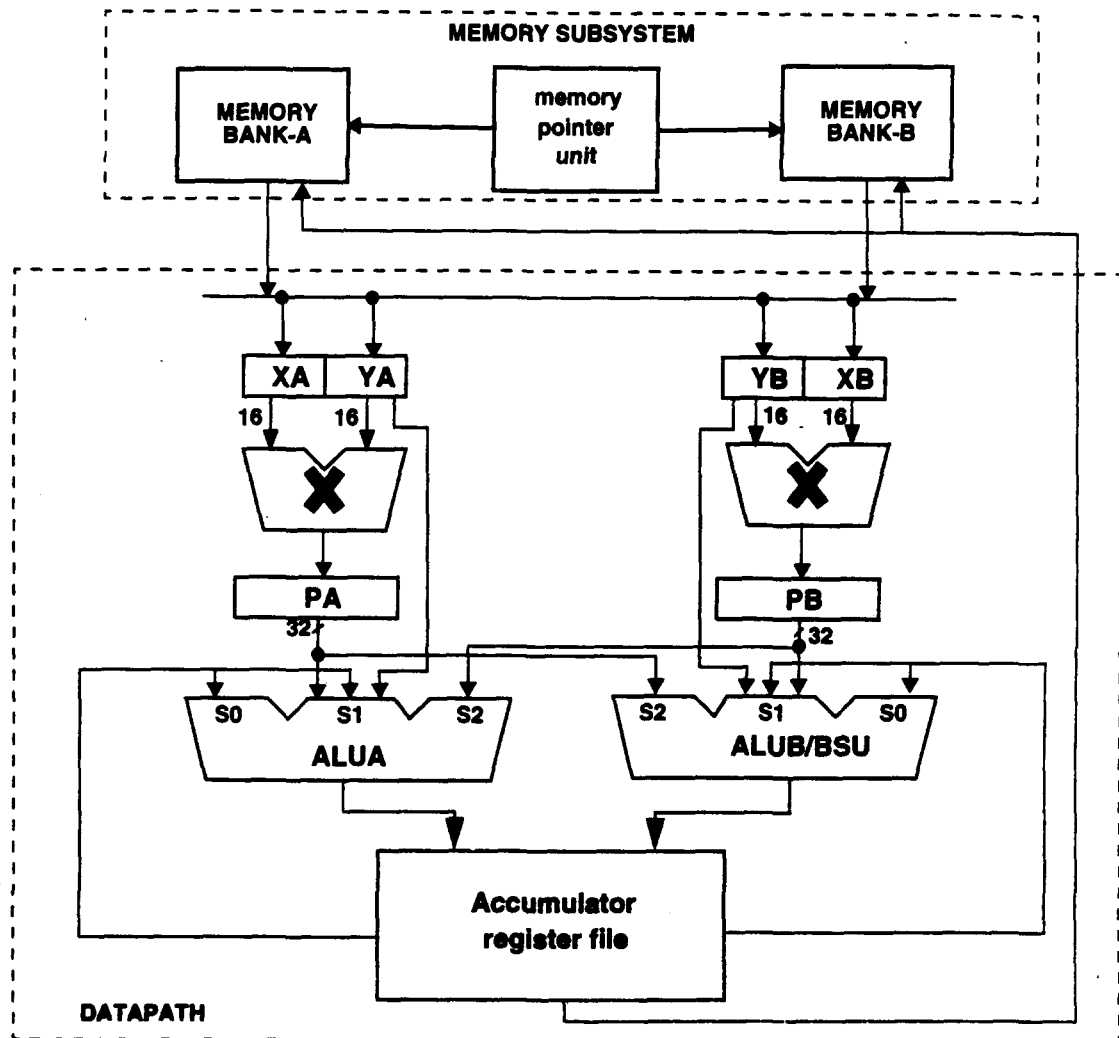


Figure 1. Architecture of the VCP datapath.