Energy-Efficient Optimization of the Viterbi ACS Unit Architecture

Hoang Q. Dao, Bart R. Zeydel, Vojin G. Oklobdžija
Advanced Computer Systems Engineering Laboratory
Department of Electrical and Computer Engineering
University of California, Davis, CA 95616
{hqdao,zeydel,vojin}@acsel-lab.com

Abstract— **Different architectural approaches for saving energy are considered for the ACS unit of a Viterbi decoder. It was found that although providing less throughput improvement than parallelism, pipelining is more energy efficient. The optimal mix of these two architectures favors more pipelining at lower throughput requirements.**

I. INTRODUCTION

Architectural selection can greatly reduce the energy consumption of digital operations with a high level of parallelism. Chandrakasan analyzed two such architectural approaches: parallelism and pipelining [1]. His quantitative analysis was limited to the energy improvement associated with degree-2 parallelism and depth-2 pipelining for the same throughput using supply scaling. In addition, the general power formulations for parallelism and pipelining were too simple and did not account for the effects of loading difference caused by these architectures and circuit resizing. Therefore, they are not useful in application where a decision on architectural tradeoff needs to be made.

The add-compare-select (ACS) operations of a Viterbi decoder [2] inherit a great deal of parallelism that can be architecturally exploited to improve energy efficiency. The ACS circuit itself is one of the critical elements determining the performance of the decoder. Energy improvement must be achieved under a performance restriction to obtain an efficient implementation. Previous implementations of the ACS unit [3][4][5] have focused on the circuit structures and employed an ad-hoc approach for selecting the number of ACS circuits to use. In this paper, we provide a quantitative study of parallel and pipeline architectures for an ACS circuit in terms of energy and throughput improvement. We later expand the analysis to demonstrate the benefit of mixed architecture in the case study of the ACS unit of a Viterbi decoder.

A. ACS Basics

In a Viterbi decoder, the ACS unit is used to determine the shortest of possible trellis paths for each state and receiving symbol. The minimal path value, called path metric, is stored at each state.

The basic ACS operation is demonstrated in Fig. 1. For a radix-2 trellis, the computation of the shortest path to a state requires two additions and one comparison. The addition computes the value of each of the two possible paths from the current path metrics and the receiving symbol. The comparison selects the smaller of the two to be updated in the path metric. The decision path is forwarded to the decision memory that is used later to decode the receiving symbols. The decode process and architecture in the decision memory [3][6] are not the focus of this paper and are omitted.

Many implementations have been suggested for the ACS circuit. The most straight-forward is the direct concatenation of the adder and comparator circuits (Fig. 1) [3]. Another approach involves bitwise additions and comparisons [5][7]. Due to the long logic stages and high level of parallelism in ACS operations, these circuits can always be pipelined with no penalty associated with data dependency. A bit-pipelining implementation is given in [4]. These circuits can also be expanded to perform the ACS operations for every two consecutive receiving symbols [3][7].

B. Architectural Approaches

While simple ACS circuit structures promise efficient implementation (such as area and less routing), further energy saving can independently be achieved with architectural approaches. One architectural approach to improving energy-delay efficiency is parallelism, where the hardware is replicated N times and the output is multiplexed.

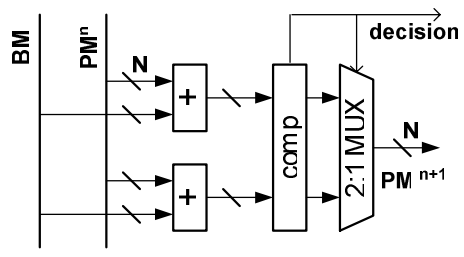

Figure 1. ACS block diagram

This work has been supported by SRC Research Grant No. 931.001, California MICRO 04-068, Fujitsu Ltd. and Intel Corp.

The advantage of parallelism is that the throughput can be improved up to N times. The disadvantages are the addition of an N:1 multiplexer at the output and an N-times increase in circuit area and output load.

Another approach to improving energy efficiency is to pipeline the circuit. Similar to parallelism, the throughput is improved by the degree approximately equal to the number of pipeline stages. No multiplexer is needed, nor does output load vary significantly. The overhead is the addition of extra Clock Storage Elements (CSEs) between the pipeline stages, which degrades performance and increases energy consumption.

II. Circuit Implementation and Sizing

For our analysis, the ACS circuit is implemented with static CMOS gates using the Kogge-Stone scheme for both the adders and the comparator [8] with a fixed 2μm input size. The output load is set according to the system under study to account for loading variation in different architectural approaches.

Both the circuit sizing and supply scaling aspects of the designs are analyzed. Circuit sizing is performed using the optimization methodology discussed in [9]. Energy and delay estimations are computed from gate characterizations for the 1.2V, 0.13μm CMOS technology. The supply scaling results are obtained by scaling down the supply voltage from the circuit-sizing point where its hardware intensity η (circuit sizing sensitivity) matches its voltage intensity θ (voltage scaling sensitivity) as explained in [10]. The general equations for these terms are shown in (1). For the 130nm technology used in the analysis, θ is approximately equal to 2.1 at the nominal supply voltage and decreases monotonically at lower voltages.

$$\eta = \frac{D}{E}\frac{\partial E}{\partial D}\bigg|_{\text{fix voltage, change size}} : \text{hardware intensity} \qquad (1)$$

$$\theta = \frac{D}{E}\frac{\partial E}{\partial D}\bigg|_{\text{fix size, change voltage}} : \text{voltage intensity}$$

III. Effects of Architectures

The quantitative study on the energy benefits of parallelism and pipelining in [1] is too simple for actual application. It is now redone, accounting for the effects of output load difference and circuit resizing. The circuit setup in Fig. 2 is used. The feedback bus is added to reflect the routing cost of different architectures. In particular, parallelism is significantly affected due to large area increase (for replicated circuits) while pipelining is not. For our analysis, the feedback bus is assumed 128μm long, approximately equal to the size of the reference ACS. It remains the same for the pipelined ACS and is multiplied by the degree for parallelism for the parallel ACS.

A. Effects of Individual Architectures

Fig. 3 shows the results for parallelism of degree N from 2 to 5 and for pipelining of depth N from 2 to 5, where Nx in the figure corresponds to the architectural degree or depth N. The circuit sizing results at the nominal supply voltage are points on the solid lines and supply scaling results on the dot-dash lines. The design points for η = 2.1 are represented by larger filled symbols.

At the nominal supply voltage the throughput for parallelism is improved by 0.87N compared with the reference, which uses no pipelining or parallelism. Both energy and throughput improvement are degraded at higher degrees of parallelism due to the increasing output load.

The throughput improvement for pipelining is 0.79N (slightly less than parallelism) at the nominal supply voltage. However, its energy does not vary much at the nominal supply voltage. This is possible because the CSE overhead is offset by sizing reduction in the ACS logic and no change in the output load.

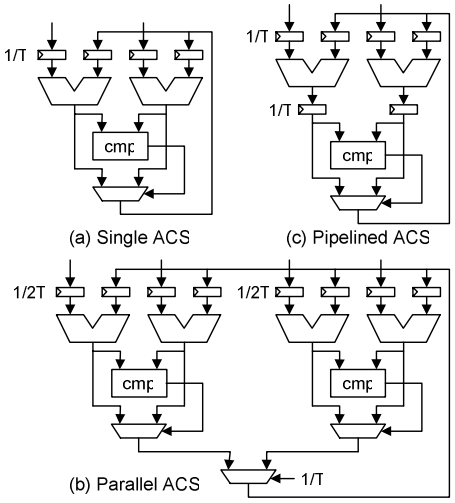

Figure 2. Architectural approaches for an ACS circuit

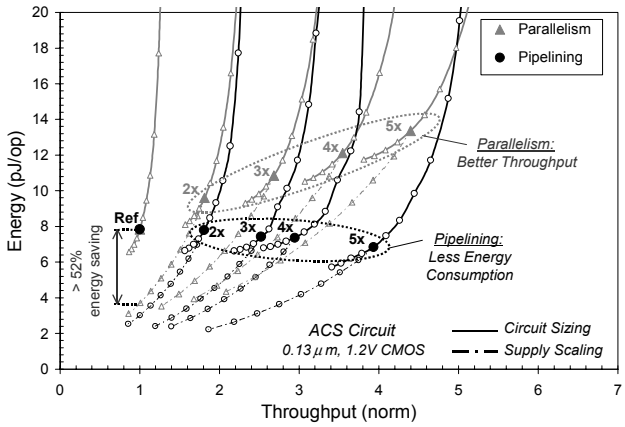

Figure 3. Energy-throughput relationship of architectures

Pipelining is generally more efficient than parallelism in terms of energy per operation. This indicates that the cost of adding CSEs in pipelining is consistently less than the cost of the increasing output load due to longer routing and MUX addition when using parallelism. Furthermore, pipelining is also the preferred approach because of the increasing impact of leakage energy in future technologies.

The throughput improvement using architectural approaches cannot not be efficiently traded for energy with circuit sizing only because the delay range is small over the energy sensitive region. It should rather be traded using supply scaling to achieve better energy efficiency. Supply scaling data in Fig. 3 shows more than 52% of energy can be saved compared to the reference design.

B. Mixed Architecture

Pipelining and parallelism can also be mixed to combine their effects and allows for flexible applications. Fig. 4 shows the energy throughput relationship between architectures. The mixed architecture data are estimated from those of parallelism and pipelining (at $\eta = 2.1$), using energy and throughput relationship of the latter to the reference. As expected, for the same depth•degree product, the mixed architecture designs deliver energy and throughput results between those using only parallelism and pipelining. This allows for more choices of architectures over the throughput region of interest and finer energy-delay difference between them.

IV. MIXED ARCHITECTURAL APPLICATION IN THE ACS UNIT OF A VITERBI DECODER

The effects of mixed architecture approach are studied on the ACS unit implementation of a Viterbi decoder. The reference implementation is a 64-state rate-½ Viterbi decoder with 96-symbol maximal traceback length, similar to [7] using a radix-2 ACS. The main circuit blocks are the ACS unit and the traceback (TB) memory. The ACS unit consists of 8 radix-2 8-bit ACS computing units (or, ACS units for short) and therefore requires 8 clock cycles to complete the update of all path metrics (PM) for 64 states and write one full column decision to TB memory. On the other hand, the TB memory read is performed every clock cycle. This results in a read-to-write rate of 8 for the TB memory, which assumingly balances the hardware constraints between the ACS unit and the memory. Since the read operation of the TB memory depends on the previous result, no pipelining can be applied to the TB operation. However, the ACS operations are independent on one another in each of the 8 clock cycles (as long as path metrics are not updated until completely used). Mixed architecture can be applied to reduce the number of ACS units.

Mixed architecture with pipelining of depths 2 and 4 are analyzed. The floor plan of the ACS unit is shown in Fig. 5. Note that significant routing length is reduced when switching from the non-pipelining to the pipelining of depths 2 and 4. This results from less ACS circuits and therefore

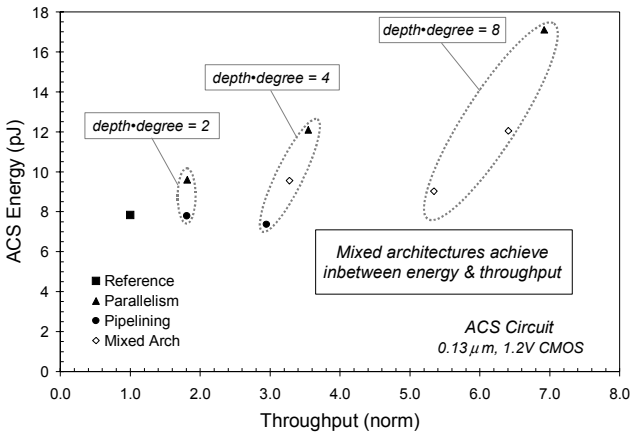

Figure 4. Estimated results for mixed architectures

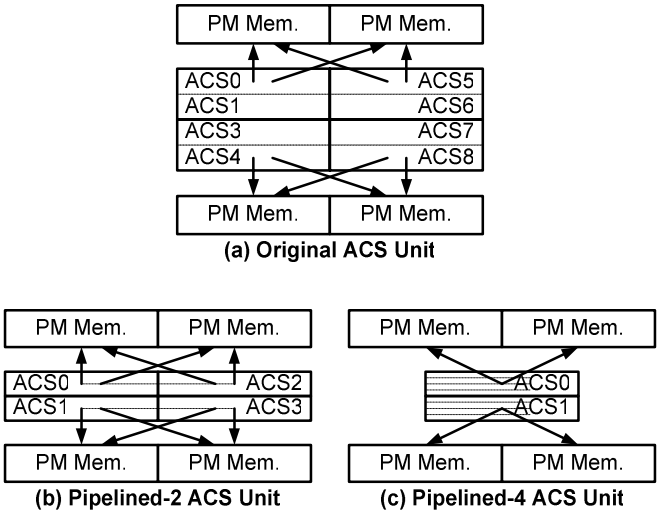

Figure 5. Floor plan of ACS units

smaller area. On the other hand, the number of PM elements loading the ACS units increases proportionally to the pipeline depth.

Table I summarizes the energy and delay estimations for the above 3 implementations of the ACS unit in 1.2V, 0.13μm CMOS technology. The ACS topology in section III is used. All circuits are optimized for the hardware intensity η of 2.1 at the nominal supply and room temperature.

TABLE I. ENERGY-THROUGHPUT COMPARISON OF ACS UNITS

# Pipeline Stages	**1**	**2**	**4**
# ACS	8	4	2
Pipeline Cycle (FO4)	20.4	11.0	6.4
Clock Cycle (FO4)	20.4	22.1	25.6
Throughput (norm)	1.00	0.92	0.80
Per ACS Energy (pJ)	17.1	18.3	18.7
Total ACS Energy (pJ)	136.6	73.1	37.3
Energy (norm)	*1.00*	*0.54*	*0.27*
EDP (norm)	*1.00*	*0.58*	*0.34*

The degradation of clock cycle when pipelining is introduced indicates that the increased number of PM elements loading the output of the ACS circuits and the addition of extra logic stages for CSEs still outweigh the routing reduction. However, the energy consumption per ACS unit remains relatively the same, mostly due to the delay relaxation in the pipelined implementations. Consequently, the total energy consumption in the whole ACS block is significantly reduced in pipelined designs due to the reduced number of ACS circuits used. The result is 46% and 73% energy saving for pipelining of depth 2 and 4 respectively. When the effects on clock cycle and energy are combined in terms of the energy-delay product (EDP), pipelining the ACS units allows for significant saving, 42% and 66% for depths of 2 and 4 respectively.

The advantages of pipelining the ACS unit can be observed more clearly in the energy-delay space, shown in Fig. 6. The solid lines represent circuit sizing points at nominal supply. Supply scaling results are shown in dot-dash lines with unfilled symbols. The results are consistent with earlier estimation on mixed architecture (section III-B). The reference 8-ACS parallel implementation can deliver the highest throughput but is very inefficient in energy. Pipelining the ACS unit results in significant energy reduction. The pipelined-2 4-ACS unit can deliver most of the throughput of the reference at lower energy cost. At the low throughput requirement, more than 37% of the energy can be saved. The pipelined-4 2-ACS implementation uses the least energy. At a very low throughput requirement, it can save more than 63% energy compared to the reference.

Beside the given three implementations of the ACS unit (running in 8 clock cycles), there can be other architectural implementations. For example, the ACS unit may have 16 parallel ACS circuits running in 4 clock cycles or 4 parallel ACS circuits running in 16 clock cycles – with any architectural mix of the same degree•depth product are acceptable for each case. In order to maintain similar throughput as the reference, the ACS performance requirement needs to be half for the first case and be double for the second one. However, it can be seen from Fig. 6 that, at the nominal supply voltage, the range of ACS performance is so small (less than 2) that it cannot accommodate the above new implementations. Therefore, the new implementations are not compatible in performance with the reference and do not need to be considered. However, they provide alternative implementations when the throughput requirement exceeds the performance range of the reference.

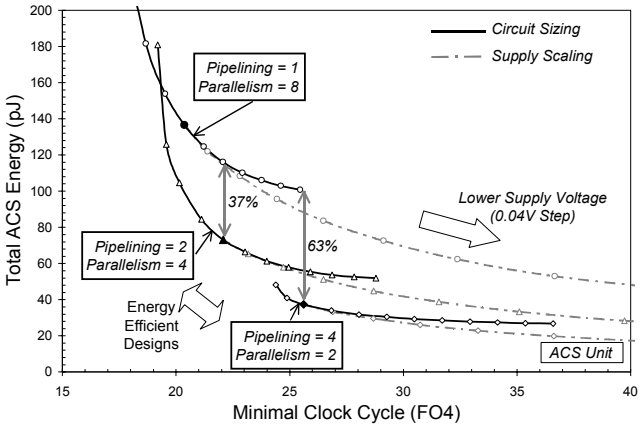

Figure 6. Energy-performance comparison of ACS units

V. CONCLUSION

We have provided a quantitative analysis of the effects of architectures on the energy and performance of the ACS unit. For a single ACS circuit, the throughput improvement is 87% and 79% of the ideal throughput for parallelism and pipelining, respectively. In addition, we observe that pipelining is more energy efficient than parallelism while delivering similar performance. Consequently, a mix of parallelism and pipelining in the implementation of a practical ACS unit allows for 37-63% energy saving versus a fully parallel implementation. The optimal depth of pipelining is dependent on the actual throughput requirement, where higher depth should be used for lower throughput.

REFERENCES

[1] A. Chandrakasan, "Low-Power Digital CMOS Design," PhD thesis, University of California at Berkeley, UCB/ERL Memorandum No. M94/65, August 1994.

[2] G. D. Forney, Jr., "The Viterbi Algorithm," Proc. IEEE, Vol. 61, no. 3, pp. 268-278, March 1973.

[3] P. Black, T. Meng, "A 140 MB/s 32-state radix-4 Viterbi decoder", IEEE Journal of Solid-State Circuits, vol. 27, no. 12, pp. 1877-1885, December 1992.

[4] A. Yeung, J. Rabaey, "A 210 MB/s Radix-4 Bit-Level Pipelined Viterbi Decoder," IEEE International Solid-State Circuits Conference, Digest of Technical Papers, pp. 88-89, 1995.

[5] Y.-N. Chang, H. Suzuki, K. K. Parhi, "A 2-Mb/s 256-State 10-mW Rate-1/3 Viterbi Decoder," IEEE Journal of Solid-State Circuits, vol. 35, no. 6, pp. 826-834, June 2000.

[6] G. Feygin, P. G. Gulak, "Architectural Tradeoffs for Survivor Sequency Memory Management in Viterbi Decoders," Trans. Communications, vol. 41, no. 3, pp. 425-429, March 1993.

[7] M. Anders, S. Mathew, R. Krishnamurthy, S. Borkar, "A 64-State 2GHz 500Mbps 40mW Viterbi Accelelrator in 90nm CMOS," Proceedings of the 2004 Symposium on VLSI Circuits, Honolulu, HI, June 17-19, 2004.

[8] P.M. Kogge, H.S. Stone, "A Parallel Algorithm for the Efficient Solution of General Class of Recurrence Equations," IEEE Trans. Computer, vol. C-22, no. 8, pp. 786-793, Aug 1973.

[9] H. Q. Dao, B. R. Zeydel, V. G. Oklobdžija, "Energy Optimization of Digital Pipelined Systems Using Circuit Sizing and Supply Scaling," IEEE Transaction on VLSI Systems, submitted for publication.

[10] V. Zyuban, P. Strenski, "Unified Methodology for Resolving Power-Performance Tradeoffs at the Microarchitectural and Circuit Levels," Proc. Int. Symp. on Low Power Electronics and Design, Aug. 2002, pp. 166-17.