Low Power Design of an Acoustic Echo Canceller *Gmdfα* Algorithm on Dedicated VLSI Architectures

S. Gailhard, N. Julien, A. Baganne, E. Martin
LESTER-UBS Lab, France
{gailhard,julien,baganne,emartin}@iuplo.univ-ubs.fr

Abstract

The acoustic echo cancellation with adaptive filters is a computationally intensive problem that needs real time cost effective solutions for embedded systems. Low Power optimized signal processing architectures are likely to provide such solutions in the future. In this paper, we present different real-time optimized architectures of the popular Gmdfα algorithm, obtained by a HLS CAD tool providing trade-off between area and power dissipation.

1. Introduction

In Digital Signal Processing, we often have to deal with Real Time implementations problems with large amount of computational power. ASIC implementation provides lower cost for a large scale production than DSP and decreases the system power dissipation.

To get rapid prototyping of different ASIC architectures, it becomes essential to use high level Computer Aided Design (CAD) tools. At the first level of the design, architectural synthesis tools allow specification of applications at a behavioral level of abstraction. These tools automatically explore different parallel solutions in order to satisfy the algorithm execution timing constraints to optimize the whole design [cha95].

In this paper a methodology is described for real time implementation of the *GMDFα* (standing for Generalized Multi-Delay adaptive Filter with over-sampling factor α) acoustic echo cancellation algorithm using *Gaut_w* HLS CAD tools [gai98]. This algorithm is used under a 8 kHz throughput constraint. Previous works present *GMDFα* implementation on software, hardware or both [bag97], but the low power factor consideration represents a new work.

2. Overview on *GMDFα* algorithm and *Gaut_w* low-power HLS tool

The *GMDFα* algorithm is a frequency domain block algorithm where a block of speech input data is processed at one time, producing a block of outputs. The segmentation of the impulse response of size L in K blocks of smaller size N leads to short processing delay. Moreover, increased updating rate of the filter coefficients in the ratio α improves both the convergence speed and tracking capability of the algorithm. The algorithm complexity is approximately N(8K + 9log2(N) - 10) additions and N(10K + 3log2(N) - 3) multiplications [ ]bag97]

Gaut_w [gai98] is an architectural synthesis tool targeting real time and low power dedicated architectures. From the behavioral VHDL description of the application, this tool proposes different architectures satisfying the time constraint application with different area and power dissipation characteristics. The power estimation (memory, operator and clock tree) is processed at the architectural level. Three low power modules are implemented in the *Gaut_w* HLS tool [gai98]. **Module selection** shows the supply voltage variation impact and determines the best operator set from a formal library optimizing the design cost. **Assignment and Scheduling** optimize power dissipation by decreasing activity rate at operator entries using a power dissipation model depending of signal statistics. The **threshold value** concept is used in this tool. The *variable life times* (time intervals between the production and consumption for each variable) is calculated after the scheduling process. If this time is greater than the **threshold value** specified by the designer, the variable is stored in an external memory instead of a circuit register. Therefore, this value influences clock tree and memory power dissipation.

0-7695-0104-4/99 $10.00

3. Algorithm implementation

The library used is composed of 0.8 µm CMOS 16 bits fixed point format [gai98]. Several real time architectures can be obtained dealing with different power and area characteristics.

Threshold value impacts :

Table presents impacts of this value on the *Gmdfα* algorithm.

Threshold value (ns)	Mem (mW)	Clock (mW)	Circuit Power (mW)	Circuit Area (10^{-3} mm^2)
100	68	30	149	3495
200	67	86	184	5115
500	66	172	269	8111
1000	64	266	361	11665

Table 1 : Threshold value impact

Results show that in this case, it is power efficient to use as often as possible the memory in order to store data variables : using internal registers increases the register number but the power memory reducing is enable to compensate the clock power dissipation. Nevertheless, for applications with harder throughput constraint time, the ratio between one more register and one memory access is smaller and the minimum power is around 200 or 300 ns.

Selection module impacts :

Selection module results are presented in Table 1 and show variation of the circuit power dissipation and area when the supply voltage changes.

Supply Voltage	Power (mW)	Area (10^{-3} mm^2)	Power saving factor	Area saving factor
2.6	55	12105	63%	-245%
3.2	61	3495	59%	0%
5.0	149	3495	0%	0%

Table 1 : Module selection impact

The P power dissipation ($P = A.\ V_{dd}^2$ with A a constant) decreases with the supply voltage because of the quadratic dependence of V_{dd} on the power dissipation. But, when the supply voltage decreases, the design area is constant from 5.5 Volts to 3.2 Volts because the application throughput constraint is not very hard and allows to decrease operator performance. Nevertheless, under 3.2 Volts, more operators have to be allocated to satisfy the constraint. Therefore, this module leads to a trade-off : the designer can decrease the power dissipation up to 63% with increasing the area to 245% according to its needs. Nevertheless, using 3.2 Volts seems to be the best solution (59% power saving factor without area penalty).

Assignment and scheduling modules impacts

Statistics signals at an operator entry infer on its power dissipation [lan94]. We have done two synthesis with different parameters : ***Ass*** with only assignment module and ***Sch*** with only scheduling module. The ***Ref*** results have been carried out with no low power optimization module.

	Circuit Power (mW)	Circuit Area (10^{-3} mm^2)	Operator. power (mW)	Power saving factor	Area saving factor
Ref	149	3485	30	0%	0%
Ass	148	5705	29	1%	-64%
Sch	135	3639	24	9%	-4%

Table 2 : Assignment and scheduling

Results show that guiding the synthesis with statistic signals leads to decrease the power dissipation to 9% with only 4% of area penalty using scheduling method. The assignment module must not be used on this algorithm. Therefore, using a white noise model for input statistics leads to neglect a power optimization direction.

Another experience using all low power modules together (Table 3)shows that 67% power saving factor (49 mW) can be obtained while consenting only 22% area increasing (4265 10^{-3} mm^2).

Supply Voltage	Power (mW)	Area (10^{-3} mm^2)	Power saving factor	Area saving factor
3.2	49	4265	67%	-22%
5.0 ***(Ref)***	149	3485	0%	0%

Table 3 : Using all modules

4. Conclusion

In this paper, techniques for power optimization using *Gaut_w* CAD tool are presented. Results show that supply voltage variation has the greatest impact on power dissipation and may lead to 63% power reduction in regard to the common 5 Volts supply voltage. However, this low power method has to be correlated with the optimization of signal statistics and the choice of the *threshold value*.

References :

[bag97] A Baganne, J.L Philippe, E. Martin "A codesign methodology for telecommunication systems : A case study of an Acoustic Echo Canceller". Proceedings of the IEEE SIPS'97, Leicester, UK, Nov 97.

[cha95]A. P. Chandrakasan, M. Potkonjak, R. Mehra, J. Rabaey, R. W. Brodersen, *"Optimizing power using transformations"*, IEEE trans. on VLSI systems, Vol. 14, No 1, January 95, pp. 12-29

[gai98]S. Gailhard, N. Julien. J.-Ph. Diguet, E. Martin, "How to Transform an Architectural Synthesis Tool for Low Power VLSI Designs", pp. 426-431, 8-th IEEE GLS-VLSI 98, Louisiana, Feb. 1997