LOW POWER DEDICATED DATAPATHS FOR LMS ADAPTIVE FILTER

Douglas Foster, Nathalia Ardizzone, Marcelo Fonseca, Sergio Almeida, Eduardo Costa
E-mail: douglas_foster@ig.com.br, nathalia_ardizzone@hotmail.com, marcelor.fonseca@gmail.com, smelo@ucpel.tche.br, ecosta@ucpel.tche.br

UCPel - Universidade Católica de Pelotas

ABSTRACT

This paper addresses the use of architectural transformations for the low power realization of LMS Adaptive Filter on dedicated datapath architectures. New low power arithmetic operators, which maintains the pure form of an array multiplier, and an efficient adder are used as basic modules in the LMS filter structure. We report power savings using the proposed methodology.

1. INTRODUCTION

The implementation of the Least Mean Square (LMS) Adaptive filter through dedicated architectures is discussed. The main goal is to reduce the power consumption by the use of transformation techniques. This type of filter is the most popular and widely used because of its simplicity and robustness [1], [2]. Since multiplier modules are common to this type of algorithm, one of the low power techniques used in this work is the use of low power multiplier architectures [3]. As observed in this paper, LMS Adaptive filter architectures that use the multiplier of [3] are more efficient than those that use the common Booth multiplier. We have tested the use of low power arithmetic modules in combinations of Fully-Sequential and Semi-Parallel architectures.

2. DEDICATED LMS ADAPTIVE FILTER DATAPATH IMPLEMENTATION

The LMS adaptive algorithm is a convenient method of adapting the coefficients of a finite impulse response (FIR) filter [1], [2]. Equation 1 describes this algorithm, where $\omega(n) = [\omega_0(n), \omega_1(n), .., \omega_{N-1}(n)]^T$ are coefficients and $x(n) = [x(n), x(n-1), ..., x(n-N+1)]^T$ are the input data samples currently in filter memory, $d(n)$ is the desired response, N is the filter length, and μ is the algorithm step size.

$$\omega(n+1) = \omega(n) + \mu.(d(n) - \omega^T(n)x(n)).x(n) \quad (1)$$

We have implemented two different 16-bit (8-order) dedicated architectures for the Equation 1. The first type is the Fully-Sequential architecture, as a manner to reduce hardware requirements for the LMS Adaptive filter algorithm, shown in Figure 1. In this implementation the basic idea is to re-use as much of the hardware as possible. In this architecture, (4 clock cycles are necessary for a full calculation (64 cycles for the FIR computation, 1 cycle for the error calculation, 1 cycle for the updating coefficient evaluation and 8 cycles for the new coefficients calculation).

In order to speed-up the Adaptive filter calculation, we have experimented a Semi-Parallel architecture. In this architecture, shown in Figure 2, hardware requirements in terms of arithmetic operators are duplicated with respect to the Fully-Sequential, allowing two samples to be processed simultaneously. Thus, the full filter operation can be performed using 42 clock cycles (32 cycles for the FIR calculation and 10 cycles for the other operations as in the Fully-Sequential circuit).

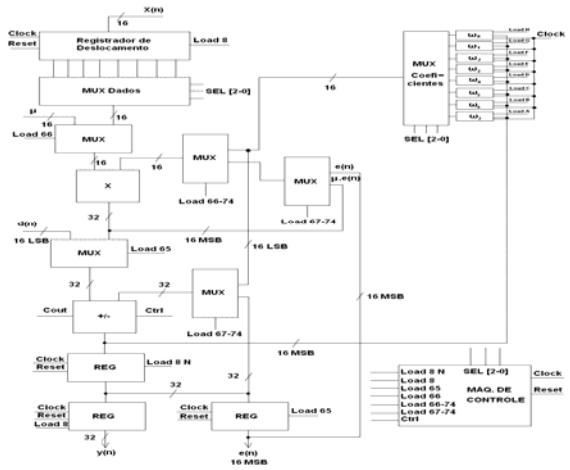

Fig 1. Datapath of Fully-Sequential LMS Adaptive Filter

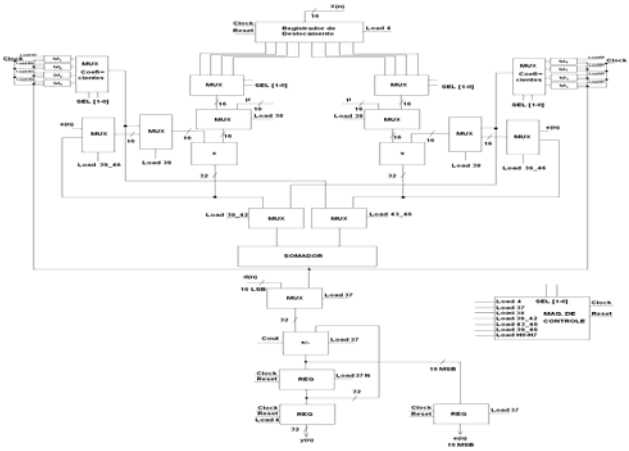

Fig 2. Datapath of Semi-Parallel LMS Adaptive Filter

3. RELATED WORK ON LMS ADAPTIVE FILTER REALIZATION

Various architectures have been used in LMS Adaptive filter realizations [4], [5]. The work proposed in this paper will be based on some transformation, specially the techniques that target the increase in performance and switching activity reduction. In our work, we experiment the use of low power arithmetic operators in the dedicated LMS Adaptive filter architectures. In a previous work, Carry Save and Wallace tree multipliers are experimented as arithmetic operators for non-adaptive and adaptive filter implementations [6]. However, when power consumption is the primary concern, Booth multiplier has been the primary choice [(]. In [8] Booth multiplier was used as a module of a LMS Adaptive filter architecture. In our work, we propose the use of a more efficient multiplier architecture in the structure of Fully-Sequential and Semi-Parallel LMS Adaptive filter implementations.

Table 1. Area, Maximum operating frequency and Power per sample results

Architectural Alternative	Area (LCs)		Difference (%) Array vs. Booth	Max. Freq. (MHz)		Difference (%) Array vs. Booth	Power (mW)		Diff. (%) Array vs. Booth
	Booth	Array		Booth	Array		Booth	Array	
Fully-Seq.	964	1290	+25.2	18.6	19.6	+5.1	16.4	16.3	-0.6
Semi-Par	1355	212	+36.2	18.0	14	-3.4	9.4	8.	-8.1

4. LOW POWER ARITHMETIC OPERATORS

For the operation of a radix-2^m multiplication, the operands are split into group of m bits. Each of these groups can be seen as representing a digit in a radix-2^m. The radix-2^m operation in 2´s complement representation by using the methodology of [3] is illustrated in Figure 3.

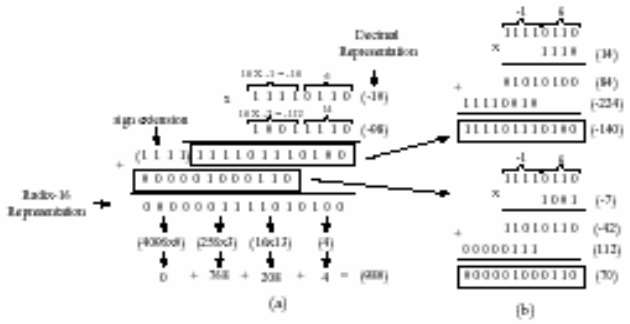

Fig 3. Example of a 2´s complement 8-bit wide radix-16 multiplication

The radix-4 Booth's algorithm (also called Modified Booth) has been presented in [ ]. In this architecture it is possible to reduce the number of partial products by encoding the two's complement multiplier. In the circuit the control signals (0,+X,+2X,-X and -2X) are generated from the multiplier operand for each group of 3-b.

One of the major speed enhancement techniques used in modern digital circuits is the ability to add numbers with minimal carry propagation. In this work, we have used in the LMS Adaptive filter architectures, a more efficient multiplexer-based full adder (MBA)[9] rather than the conventional Ripple Carry adder (RCA).

5. RESULTS

In this section, we present the results for the LMS Adaptive filter architectures presented in Section 2. Area, delay and power results were obtained in the Quartus-II environment. LMS filters were synthesized to Stratix device from Altera. Area, presented in terms of logic cells (LCs), maximum operating frequency and power per sample results are shown. For the power simulation we have applied a random pattern signal with 100 input vectors.

5.1. Area results

Table 1 presents area results for the LMS Adaptive filter architectures using the array (m=2) and Modified Booth operators. As can be observed in this table, there is significant area difference between the architectures with these operators. The Fully-Sequential and Semi-Parallel architectures which use the array multiplier operators present more area. This due to the fact that the basic multiplier modules in the array multipliers require more area than the Booth circuits.

5.2 Maximum operating frequency results

Although LMS Adaptive filter architecture with the array operators present higher area, these architectures permit a higher operating frequency than the architectures with Booth operators, as shown in Table 1. In the Semi-Parallel architecture case, a reduction occurs due to where the higher number of MBA adders and the multiplier circuit present in the critical path contributes for a lower delay value in the filter circuit.

5.3 Power per sample results

As can be observed in Table 1, with the use of the array multiplier power per sample savings are achievable in the Fully-Sequential and Semi-Parallel LMS Adaptive filter architectures. This occurs because multiplier circuits are the main responsible for the power consumption in the filter architectures and the array multiplier consumes less power due to the simplest structure and smaller critical path and delay values. This power reduction is mainly due to the lower logic depth of the array multiplier structure which has a big impact on the reduction of the amount of glitching in the filter circuits. As observed in Table 1, the Semi-Parallel architecture presents more power reduction due to the higher number of multipliers in the filter circuit.

6. CONCLUSIONS

In this work, different dedicated architectures for LMS Adaptive filter were implemented. Power optimization techniques were tested including architectural exploration. Low power arithmetic operators were experimented in the filter architectures, by using the array (m=2) and Modified Booth operators. Results showed that, despite higher area shown by the architectures with the array operators, these architectures can present higher frequency and less and power consumption.

7. REFERENCES

[1] B. Widrow and S. D. Stearns. *Adaptive Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall.

[2] S. Haykin. *Adaptive Filter Theory*. 4. ed. 2002. Englewood Cliffs, NJ: Prentice-Hall.

[3] E. Costa, J. Monteiro, S. Bampi. "A New Architecture for Signed Radix-2^m Pure Array Multiplier". *ICCD02*, 2002.

[4] M. Goel, N. Shanbhag. "Low Power Adaptive Filter Architectures via Strenght Reduction". *ISLPED*, 1996.

[5] S. C. Douglas, Zhu, Q., K. F. Smith. "A Pipelined LMS Adaptive FIR Filter Architecture without Adaptation Delay". *IEEE Transactions on Signal processing,* 46: 5- 9, 1998.

[6] J. Park, et al. "Non-Adaptive and Adaptive Filter Implementation Based on Sharing Multiplication", *IEEE ICASSP*, June 2000.

[ ] I. Khater, A. Bellaouar and M. Elmasry. "Circuit Techniques for CMOS Low-Power High Performance Multipliers". *IEEE Journal of Solid-State Circuits*, 31:1535-1546, 1996.

[8] G. Stemmer, et al. "Implementações de Filtros Adaptativos para Cancelamento de Eco", VI *Iberchip*, p.551-559, 2000.

[9] A. Al-Sheraidah, et al. "A Novel Low Power Multiplexer-Based Full Adder", *ECCTD´01*, 2001.