

Backend analysis and implementation of LMS adaptive filter using VLSI technology

N.J.R.Muniraj

Karpagam Innovation Centre, Karpagam College of Engineering, Coimbatore

ARTICLE INFO

Article history:

Received: 15 February 2011;

Received in revised form:

20 February 2011;

Accepted: 28 February 2011;

Keywords

HDL,
TSMC,
LMS,
FIR,
SPICE,
RTL,
ASTRO,
GDS11,
DRC.

ABSTRACT

The role of electronic equipments in the industry has increased tremendously in recent past. With new technologies and techniques being considered in other domains, such as automotive, multimedia communications, mobile applications bring down the cost of the electronic gadgets. As the cost factor controls the reliability and volume issues, there is a need for design and development of low cost, reliable technology for industrial applications. The proposed techniques have been modeled using Verilog HDL and the models have been verified using test benches with a functional coverage of 95%. The results obtained have been compared with MATLAB results, which are considered to be a benchmark. The HDL (Hardware Description Language) code is synthesized using Synopsys Design Compiler targeting 130-nanometer TSMC (Taiwan Semiconductor Manufacturing Company) library and target technology. The synthesized netlist obtained for all the adaptive filtering techniques proposed in this research work is taken through physical design flow consisting of Floorplanning, Placement and Routing steps. The results obtained at each step are simulated for the functionality. The final GDSII (Graphical Design Standard II) file is generated for the proposed techniques.. The floor planning, placement and routing of the netlist ensures that the overall size for the entire chip does not exceed by 2.15 square millimeters. The results obtained for adaptive filtering techniques have proven that the complexities in the industrial applications can be met if the design is implemented on ASIC.

© 2011 Elixir All rights reserved.

Introduction

The Least Mean Square (LMS) algorithm was first developed by Widrow and Hoff in 1959. It has become one of the most widely used algorithms in adaptive filtering (Zaknich 2003). The LMS algorithm is a type of adaptive filter known as stochastic gradient-based algorithms as it utilizes the gradient vector of the filter tap weights to converge on the optimal wiener solution. It is well known and widely used due to its computational simplicity. It is this simplicity that has made it is the benchmark against all other adaptive filtering algorithms (Long 1996). With each iteration of the LMS algorithm, the filter tap weights of the adaptive filter are updated according to the following formula

$$w(n+1) = w(n) + 2\mu e(n)x(n) \quad (1.1)$$

Here $x(n)$ is the input vector of time delayed input values, $x(n) = [x(n) \ x(n-1) \ x(n-2) \ \dots \ x(n-N+1)]^T$. The vector $w(n) = [w_0(n) \ w_1(n) \ w_2(n) \ \dots \ w_{N-1}(n)]^T$.

$w_{N-1}(n)$ represents the coefficients of the adaptive FIR filter tap weight vector at time n . The parameter μ is known as the step size parameter and is a small positive constant. This step size parameter controls the influence of the updating factor. Selection of a suitable value for μ is imperative to the performance of the LMS algorithm, if the value is too small the time the adaptive filter takes to converge on the optimal solution will be too long; if μ is too large the adaptive filter becomes unstable and its output diverges (Marque 2005). The input signal is sampled at 1 KHz, with 16 bit number using IEEE754 floating-point format which is considered as a test vector for

benchmarking the proposed technique. The signals (reference and actual) are fed to the architecture at a data rate of 16Kbits per second. The adaptive algorithm designed for 8stage (8th order) is used to filter the signals to produce the error with latency of 8 clocks and throughput of 1 clock cycle. The equivalent design architecture for the LMS is shown in Figure 1.1 where data1 refers $d(n)$, data2 refers $x(n)$, fir_op refers $y(n)$, error $e(n)$ and $c_0 - c_7$ refers $w(n)$. ECG

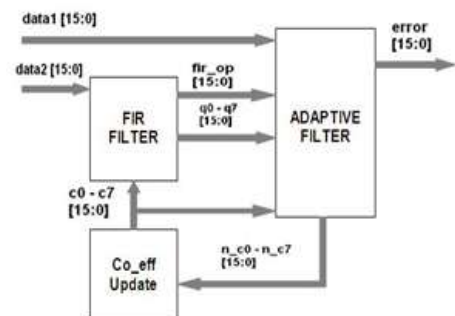


Figure 1 Designed architecture of Least Mean Square (LMS)

Each iteration of the LMS algorithm requires 3 distinct steps in this order (Haykin 1992).

1. The output of the FIR filter, $y(n)$ is calculated using equation 1.2.

$$y(n) = \sum_{i=0}^{N-1} w_i(n).x(n-i) \quad (1.2)$$

2. The value of the error estimation is calculated using equation 1.3

$$e(n)=d(n)-y(n) \tag{1.3}$$

3. The tap weights of the FIR vector are updated in preparation for the next iteration, by equation 1.4.

$$w(n+1)=w(n)+2\mu e(n)x(n) \tag{1.4}$$

The main reason for the LMS algorithms popularity in adaptive filtering is its computational simplicity, making it easier to implement than all other commonly used adaptive algorithms(Emmanuel 2002). For each iteration the LMS algorithm requires 2N additions and 2N+1 multiplications (N for calculating the output, y(n), one for 2μe(n) and an additional N for the scalar by vector multiplication) (Keshab 1999).

The architectures of LMS is designed using Verilog HDL (RTL Level Coding).The simulation reports of these architectures are shown in Figure 1.1.

The obtained results at the 8th iteration output of LMS is 38403(unsigned 16 bit value).The LMS architecture is considered in this research and modeled using HDL language for hardware implementation (Mark Gordon1999). The Figure 1.1 shows ModelSim simulation results at 10th and 20th iteration.

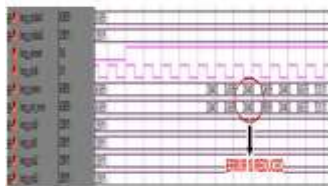


Figure 1.1 Simulation result of LMS architecture

Synthesis Design Flow

The next phase in the ASIC design flow is converting RTL code to the gate level netlist, called synthesis, targeted to the specific technology. TSMC 130nm technology is selected as a target library and Synopsys Design Compiler tool is used for synthesis (Basker 2004).

Synthesis is a three-phase process where it starts with translating the RTL code to the gate level netlist (Michael 2001). The netlist is optimized using the constraints given. Constraints are two types namely, environmental and optimization constraints. Optimization constraints include operating frequency (clock period), input and output delays at the IOs. Operating temperature, process variations, supply voltage and wire load models comes under Environmental constraints (Ahmed Elhossini, 2004). The Figure.2.1 shows the schematic, which was generated after synthesizing the RTL code

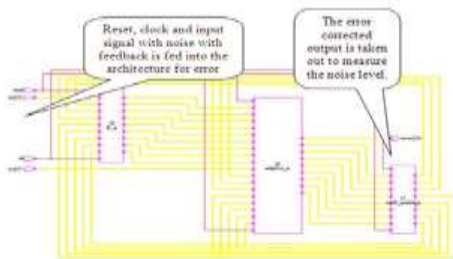


Figure 2 Synthesized schematic of LMS architecture

Static timing analysis:

Once the design is synthesized the next step is to verify the design for timing. STA is the process in which the delays of a circuit are calculated by adding the individual gate and net delays for each path, also the process in which the path delays in a circuit are compared against their required minimum (hold) and maximum (Setup) values. Static Timing Analysis uses

SPICE characterized data stored in a technology library to verify circuit's timing. Synopsys PrimeTime is used for the timing analysis. The PrimeTime is a sign-off static timing analysis tool targeted for complex, multimillion-gate designs.

Timing reports of LMS without and with pipelined architectures are shown in Figure 3

data required time	10.15
data required time	10.15
data arrival time	-10.11
slack (MET)	0.04

Figure 3 Timing report

The Figure 3.1 shows the histogram of the path slack. The design meets the timing requirements, but there are 118 paths, which have the equal slack.

This will increase the congestion (lack of routing resources) in the physical design flow and timing will get worse in critical paths.

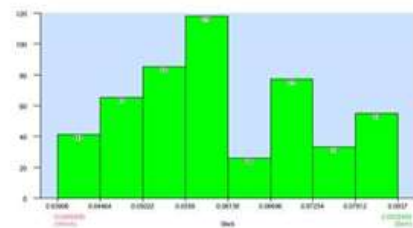


Figure 3.1 Timing histogram of LMS

Design compiler has in built capability for finding the area requirement. The results obtained are shown in Figure 3.2 the cell area is 6.4 square mm and the net is 11.217.

```

Design : VLSI
Version : V-2004.06-SP1-1
Date : Sun Mar 18 15:17:11 2007
Library(s) used:
ch11120_130m_top (File: /home/Muniraj/Info/0011120_130m_top.db)
Number of ports: 33
Number of data: 493
Number of cells: 3
Number of references: 3
Combinational area: 3079.793000
Noncombinational area: 3081.303000
Net interconnect area: 11217.710000
Total cell area: 6461.096000
Total area: 71821.903000
    
```

Figure 3.2 Area reports of LMS

The power report taken from the design compiler. The total dynamic power is 64mw for LMS. DC also can produce the power report for the design which is shown below.

The design is expected to drive a fanout out of 1pf of load capacitance at 1.2 volts. The power report clearly mentions that the dynamic power is dominating the leakage power. Use of low

Figure 3.3 power report of LMS

```

Report: power
Design: ecg_q
Version: V-2004.06-SP1-1
Date : Sun Mar 18 15:15:55 2007
Global Operating Voltage = 1.2
Power-specific unit information:
Voltage Units = 1V
Capacitance Units = 1.000000pf
Time Units = 1ns
Dynamic Power Units = 1mW (derived from V,C,T units)
Leakage Power Units = 1pW

Cell Internal Power = 49.2725mW (77%)
Net Switching Power = 14.5266mW (23%)

Total Dynamic Power = 63.7991mW (100%)

Cell Leakage Power = 164.1022uW
    
```

Power techniques such as clock gating; power gating, multi Vt cells and pipelining the power consumption can be consumed by 30%.

The synthesized netlist with the constraints file is taken into the back end design flow or physical design flow. During this phase, floorplanning, placement, clock tree synthesis and routing is done on the design to obtain the GDSII file, which can be sent for fabrication. The Figure.3.4 shows the power report taken from the design compiler. The total dynamic power is 63 mw.

Figure 3.4 shows the floor-planned view of the design. 130 I/O cells are placed on the perimeter, the cell utilization is considered to be 80% with flip chip and double back. The power supply for I/O cells and core area are separated, as both require different power supplies. Five metal layers are used for routing the entire design, power supply and ground connects are on the top layer. Floor planning is done using Jupiter XT the sign of tool from Synopsys.



Figure 3.4 Floor planned die of LMS

The floor-planned design is used for automatic placement. Placement is a process of placing the standard cells in suitable locations in the core area. The core area should be free from any obstacles like power routes, macros and hot spots. This is automatically done with the help of ASTO from Synopsys. Figure 3.5 shows the placed cells; since there are no red color displays, it implies that there are no violations.

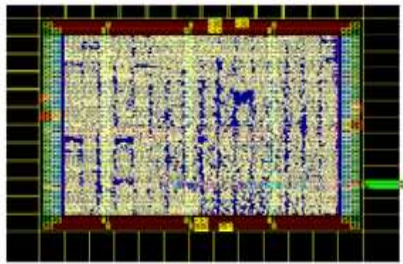


Figure 3.5 Placed design without any violations

The design when placed it fit all the cells in the core area, has to be connected with clock supply, since the die receives clock from one source or one input pad, this clock pad has to drive the flip flops placed in the entire core area. The clocks reaching all the flip-flops should have minimum latency and zero skew. In order to meet these requirement clock tree network is identified that can carry clock from the pad to all the flops. This process is called as clock tree synthesis, which is a performed using Astro tool.

The final stage in the design process is routing all the cells in the core area and to the I/O cells. Routing is two step process, first global routing is carried out and then detailed routing is performed. This ensures that all the cells are interconnected as per the netlist obtained during synthesis. And at the same time the timing is also met, after detail routing, only a part of the core space is shown for its interconnections. This is performed using Synopsys Astro. The layout of the final chip is shown in Figure

3.6. The design does not have any DRC violations; it has met all the constraints as identified in the specifications. This is converted to GDSII file and sent for fabrication. The entire design is verified using sign off tools from Synopsys.

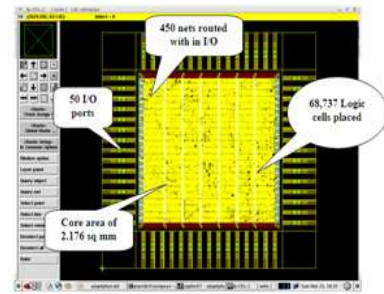


Figure 3.6 Final chip of designed LMS architecture.

Conclusion

Adaptive noise cancellation techniques such as LMS have been extensively used for noise cancellation techniques with good performances in this work. These techniques have been extended for use in industrial applications, wherein there is a need for accuracy, speed, reliability and cost. LMS algorithm has been realized on ASIC for comparison.

New architectures that incorporate pipelining is proposed and realized. The proposed architectures have been modeled and verified for its functionality successfully.

The models have been taken through the entire ASIC flow. Suitable results obtained at various stages of the ASIC flow using Synopsys clearly indicates that LMS is slow but optimizes area and power.

The input signal is sampled at 1K samples per second; has a data rate of 16Kbitsper second when fed through the proposed hardware produces output at 16Kbitsper second with latency of 8 clocks and throughput of 1 clock cycle. The proposed techniques have been modeled using Verilog HDL and compared with MATLAB results, which are then synthesized using Synopsys Design Compiler targeting 130-nanometer TSMC library and target technology.

The synthesized netlist obtained for all the adaptive filtering techniques proposed in this research work is taken through physical design flow consisting of Floor planning, Placement and Routing steps. The overall size for the entire chip does not exceed by 2.15 square millimeters.

References

1. Ahmed Elhossini, Shawki Areibi and Robert Dony.(2004), "An FPGA Implementation of the LMS Adaptive Filter for Audio Processing". In 16th International Conference on Microelectronics, Tunis, Tunisia, pp. 67–70.
2. Basker J. (2004) "Verilog HDL Primer", BS publication, Second Edition
3. Bernard Widrow and Samuel D.Stearns. (2002), "Adaptive signal Processing", Pearson Education.Second Edition
4. Emmanuel Ifeakor C. and Barrie Jervis W.(2002), "Digital Signal Processing –A practical approach", Pearson Education Asia.
5. Haykin S.(1992),"Adaptive Filter Theory". Englewood Cliffs, NJ: Prentice-Hall.
6. Keshab K. Parhi.(1999),"VLSI Digital Signal Processing Systems", John Wiley & Sons, 1st Edition..
7. Long G., Ling F. and Proakis J.G.(1996), "The ASIC design of an LMS-based decision feedback equalizer for TDMA digital cellular radio." IEEE Trans.Acoust., Speech, Signal Process., vol.37, pp.1397-1405.

8. Mark Gordon Arnold.(1999),”Verilog Digital Computer Design”, Prentice Hall PTR,1st Edition.
9. Marque C.,Bisch C.and Dantas R. (2005), “Adaptive filtering for ECG rejection from surface EMG recordings” Journal of Electromyography and Kinesiology 15 pp. 310–315.
10. Michael John Sebastian Smith.(2001), “Application-Specific Integrated Circuits”, Fifth Edition, Pearson Education Inc.
11. Zaknich A. (2003), ‘Principle of adaptive filters and self learning systems’, Springer Publishers.