UTILIZATION OF BACK PROPAGATION NEURAL NETWORK FOR CMOS VLSI CIRCUITS

Khushbu Kumari¹, Ankit,  
¹M. Tech, Department of Electronics and Communication, CBS Group of Institutions, Jhajjar (Maharshi Dayanand University, Rohtak)  
Haryana, India

Assistant Professor, Department of Electronics and Communication, CBS Group of Institutions, Jhajjar  
(Maharshi Dayanand University, Rohtak)  
Haryana, India

Abstract

Analog VLSI versions of artificial neural networks are often seen as efficient in a small space and low power consumption, yet highly programmable. Three circuits are presented as a first step towards implementation of neural analog VLSI backpack network. A connectivity matrix for a fully linked five input perceptron is one of these circuits. The second is a summer circuit which calculates the overall error. The third is a triggerable processor which optimizes the weight of particular synaptic mistake. The operations performed by these circuits in hardware are done in parallel and in real time. Higher-level design abstraction power estimation strategies. This lower both the time limit and the computing capital specifications. Artificial intelligence strategies like BPNN are appealing methods for solving complicated, non-linear, and time-constrained problems, and hence are well adapted for power estimation in CMOS VLSI circuits. 

Keywords: Back Propagation Neural Network, CMOS, VLSI, etc.

1. Introduction

VLSI innovations have recently advanced to the point that millions of transistors can be integrated on a single chip in VLSI circuits, power calculation at an early stage is critical since it has a direct effect on circuit reliability. Building power models for the composing modules is a popular higher-level power estimation technique (Subodh Gupta et al. 2000). The characterization technique is used to construct such models by running a series of simulations at a lower stage. As a result, the behavioral level power calculation can be done by calculating power consumption using the signal statistics derived from the practical simulation. Higher-level power estimation has a number of advantages over lower-level design abstraction power estimation strategies. This lower both the time limit and the computing capital specifications. Artificial intelligence strategies like BPNN are appealing methods for solving complicated, non-linear, and time-constrained problems, and hence are well adapted for power estimation in CMOS VLSI circuits.

2. VLSI Technology

The way to create an Integrated Circuit via the integration of thousands of transistors onto one chip is known as extremely broad integration (VLSI). In the 1970s, VLSI was born as a complicated semiconductor and networking technology. As a microprocessor, a VLSI computer is utilized. Since extremely large-scale integration designs are introduced, the number of Integrated Circuits (ICs) utilized in high-performance computing, controllers, telecommunications, image and video processing and consumer electronics has increased quickly. Before VLSI technologies were introduced, most ICs could perform only a limited number of functions. An electrical circuit has CPU< ROM,
RAM and other glue logic. VLSI enables IC programmers to integrate several of them onto a single chip. VLSI design and evaluation laboratory is composed of a series of a high performance work stations, integrated circuits testers and industrial computer aided design programs. The laboratory is used to develop design automation tools to diagnose, calibrate, model, estimate power and synthesize defects as well to design low power and highly testable integrated circuits. The demand for low power electronics was driven by the need for the portable connection and computer technology. Tools have recently been developed to estimate power dissipation in CMOS devices. The key benefits of VLSI technology are as follows:

- Increased cost-effectiveness for devices
- Reduced size for circuits.
- Requires less power than discrete components.
- Improved performance in terms of the operating speed of the circuits.
- Requires less space and promotes miniaturization.
- Higher device reliability

The fields research is conducted include VLSI circuits and computer assisted design where building blocks of emerging circuit technology, integrated circuit testing and defect detection, optical signal processing, computer-aided synthesis, field programmable gate arrays (FPGAs) and low-power circuit design.

Overall VLSI IC design incorporates two primary stages or parts:

**Back-End Design**: This consists of characterization and CMOS library design. Additionally, it involves fault stimulation and physical design.

The following are the Back-end design steps for hardware development:

- **Lithography**: This process (photolithography) includes masking with photo etching and photographic mask. Next, we apply a photoresist film on the wafer. A photo aligner then aligns the wafer to mask. Finally, we expose the wafer to ultraviolet light, thus highlighting the tracks through the mask.
- **Wafer Processing**: This step utilizes pure silicon melted in a pot at 1400 C. then, a small seed comprising the required crystal orientation is injected into liquified silicon and gradually pulled out, one mm per minute. We manufacture the silicon crystal as a cylindrical ingot and cut it into discs or wafers before polishing and crystal orientation.
- **Ion Implantation**: Here, we utilize a method to achieve a desired electrical characteristic in the semiconductor, i.e., a process of adding dopants. The process uses beam of high energy dopant ions to target precise area of wafers. The beams’ energy level determines the depth of wafer penetration.
- **Etching**: Here, we selectively remove material from the surface of the wafer to produce patterns. With an etching mask to protect the essential parts of the material, we use additional plasma or chemicals to remove the remaining photoresist.
- **Assembly and Packaging**: Everyone of the wafers contains hundreds of chips. Therefore, we use a diamond saw to cut the wafers into single chips. Afterwards, they receive electrical testing, and we discard the failures. In contrast, those that pass receive a thorough visual inspection utilizing a microscope. Finally, we package the chips that pass the visual inspection as well as recheck them.
- **Metallization**: In this step, we apply a thin layer of aluminum over the entire wafer.

**Front End Design**: Digital design comprises a hardware language description such as Verilog, system Verilog, and VHDL. This phase also includes design verification via simulation and other verification methods. The whole process also involves design, which begins with the gate and continues to be testable. The whole design process is progressive and the following are the front-end design steps:

- **Architecture Definition**: This includes fundamental specifications such as floating-point units and
which system to use, such as RISC or CISC and ALUs cache size.

- Problem specification: This is a high-level interpretation of a system. We address the key parameters, such as design techniques, functionality, performance, fabrication technology, and physical dimensions. The final specification includes the power, functionality, speed and size of the VLSI system.

- Logic Design: This step involves control flow, Boolean expression, word width, and register allocation.

- Functional design: This recognizes the vital functional units of a system and, thus, enables identification of each unit’s physical and electrical specifications and interconnect requirements.

- Physical Design: In this step, we create the layout by converting the netlist into a geometrical depiction. This step also follows some preconceived static rules, such as the lambda rules, which afford precise details of the ratio, spacing between components and size.

- Circuit Design: This step performs the realization of the circuit in the form of a netlist. Since this is a software step, it utilizes simulation to check the outcome.

To do the work right, state-of-the-art PCB design and analysis tools are necessary when designing the restricted margin of error in VLSI technology. Cadence’s Allegro is one such program that includes all the functionality and simulation resources you will need for anything from simple to complex circuit designs.

3. Circuit Optimization
Sizing circuit components (e.g., transistors and wire segments) has often been an important strategy for achieving desirable circuit output since the advent of Integrated Circuits nearly 40 years ago. The explanation behind this is because a circuit component’s resistance and capacitance are also featuring of its scale. Since the delay of a circuit component may be modeled as a product of the component’s resistance and the capacitance of the subcircuit powered by the component, sizing circuit components may reduce the delay of a circuit. With advancement of VLSI technology which allows for the integration of digital and analog circuits as a full device on a chip, the topic of optimal electronic circuit architecture, in comparison to manual design is becoming increasingly relevant. The architecture of analog circuit is usually done in two steps. The first stage entails selecting the circuit configuration so that it possesses the requisite capabilities to meet our circuit’s requirement. The optimization of circuit parameters, such as transistors size and element value so that output parameters, such as consuming power and gain, can be satisfied is the second step. In place of traditional optimization techniques, metaheuristics are used. They have the benefit of being easily adjusted and customized to meet the needs of individual problems.

4. Back Propagation Neural Network Design
Back Propagation is form of Artificial Neural Network (ANN) that consists of several interconnected layers of architecture. The Deepest Descent Technique is used in the BPNN learning algorithm. With a sufficient number hidden layers BPNN can mitigate error in nonlinear functions of high complexity. Back Propagation is a technique for training multilayer neural networks that uses supervised learning. Centered on the error correction learning law, its also known as the error back propagation algorithm. It consists of two passes, one forward and the other backward, that pass through the networks various layers. In the forward transfer, an operation pattern is added to networks input nods, and its influence is propagated layer by layer across the network. Finally, as the network’s actual answer a series of outputs is produced. The
BPNN is a feed forward back propagation neural network with four different parameter variations: learning intensity, momentum constant, activation mechanism, and training algorithm.

5. Utilization of Back Propagation Neural Network for CMOS VLSI Circuits

5.1 BPNN based method

The back propagation neural network is trained using eleven different network training functions namely Traingd, Traingda, Trainrp, Traingdx, Trainfgm, Traincfg, Traincgf, Traincgp, Traincbgb, Trainscg, Trainbfg and Trainoss. BPNN is trained by varying the epochs, learning rate and momentum constant.

5.2 Prediction Error

Prediction during testing and training is calculated by:

\[ Error\% = \frac{A - BX100}{A} \]

Where A is the expected value from the SPICE simulation program and B is the projected value from the network monitoring. The prediction error using Trainscg of BPNN, as seen in Table 4, indicates that BPNN with Trainscg under the conjugate gradient group is better fit for power estimation for sequential circuits with 9 inputs and 253 epochs. Trainscg is a network training function that uses the scaled conjugate gradient approach to change the weight and bias values. It functions best for networks with many inputs and is faster at solving function approximation problem.

5.3 ANFIS based Method

Table compares the power usage of BPNN and ANFIS, while Table indicates the error percentage of BPNN with ANFIS. The Trainscg function is the perfect BPNN algorithm for ISCAS ’89 circuits. Since ANFIS is well known for its ability to learn the training results, ANFIS with subtractive clustering and hybrid learning outperforms BPNN. ANFIS is a composite system that brings together the advantages of both FIS and ANN systems. Each iteration reduces the error, which is typically the major of squared variance between the real and expected results. The workspace is filled with input and testing results, which is then imported into ANFIS tool MATLAB. Exhaustive simulation is carried out by changing the number of parameters as seen in Table 2, and a new FIS is created as result. Initial FIS models are created using hybrid optimization and back propagation using constant and linear methods, with hybrid optimization with linear approach giving the lowest error. Tables 3.27 and 3.28 displays the result for different ANFIS structures with testing mistake using gaussian and sig member function. Figure 3.9 shows the graph of actual power with respect to power obtained by testing the network using ANFIS and BPNN. WE can infer that ANFIS is very close to the actual power. Figure 3.10 gives the information about error percentage comparison, ANFIS gives minimum error percentage when compared to BPNN.

Performance Evaluation

Performance of ANFIS and BPNN can be measured calculating the root mean square error (RMSE) and coefficient of determination (R) which is given below in equation:

\[ RMSE = \frac{\sum_{i=1}^{N}(Y_i^0 - Y_i^C)^2}{N} \]

\[ R = \frac{\sum_{i=1}^{N}(Y_i^0 - Y^0)(Y_i^C - Y^C)}{\sqrt{\sum_{i=1}^{N}(Y_i^0 - Y^0)^2} \sqrt{\sum_{i=1}^{N}(Y_i^C - Y^C)^2}} \]
Table 1: Power estimation using ANFIS with gaussian function

Table 2: Power estimation using ANFIS with sig member function
Table 3: Comparison of power results of BPNN and ANFIS

<table>
<thead>
<tr>
<th>Benchmark Circuit</th>
<th>Actual Power</th>
<th>BPNN</th>
<th>ANFIS</th>
</tr>
</thead>
<tbody>
<tr>
<td>S641</td>
<td>0.03629</td>
<td>0.0246</td>
<td>0.0364</td>
</tr>
<tr>
<td>S400</td>
<td>0.01065</td>
<td>0.01087</td>
<td>0.0107</td>
</tr>
<tr>
<td>S386</td>
<td>0.01628</td>
<td>0.0182</td>
<td>0.0162</td>
</tr>
<tr>
<td>S382</td>
<td>0.01046</td>
<td>0.0176</td>
<td>0.0104</td>
</tr>
<tr>
<td>S344</td>
<td>0.010846</td>
<td>0.0193</td>
<td>0.0183</td>
</tr>
</tbody>
</table>

Table 4: Error calculation for BPNN and ANFIS

<table>
<thead>
<tr>
<th>Benchmark Circuit</th>
<th>BPNN</th>
<th>ANFIS</th>
</tr>
</thead>
<tbody>
<tr>
<td>S641</td>
<td>32.21</td>
<td>-0.303</td>
</tr>
<tr>
<td>S400</td>
<td>-2.06</td>
<td>-0.46</td>
</tr>
<tr>
<td>S386</td>
<td>-12.34</td>
<td>0</td>
</tr>
<tr>
<td>S382</td>
<td>-67.93</td>
<td>0.76</td>
</tr>
<tr>
<td>S344</td>
<td>-4.55</td>
<td>0.86</td>
</tr>
</tbody>
</table>

Table 5: Statistical analysis test on BPNN and ANFIS

<table>
<thead>
<tr>
<th>Parameter</th>
<th>BPNN</th>
<th>ANFIS</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td>0.84696</td>
<td>0.99961</td>
</tr>
<tr>
<td>RMSE</td>
<td>0.0004499</td>
<td>0.0002075</td>
</tr>
</tbody>
</table>

6. Conclusion

A power estimation technique based on BPNN is PROPOSED for combinational and sequential circuits. The neural network is trained with eleven different training algorithms. The BPNN is trained with five different attributes for combinational logic circuits and 629 attributes for sequential logic circuits. The experimental results show that results are closer to ideal values. Trainigd training function is most suitable for combinational circuit since it deviates from ideal power estimator only by 0.77% and 1.04% and MSE of 0.88751 and 1.08402 for NAND and NOR based circuits. Performance of trainscg deviates from ideal power consumption only by 0.01% with a MSE of 6.24 X 10 – 05 for sequential circuits (ISCAS’89). Training and testing for NAND circuit is 106 seconds and 5 seconds. Training and testing time for NOR circuit 105 seconds and 46 seconds respectively. Training time and testing time for ISCAS’89 circuit is 149 seconds and 48 seconds respectively. The results show that BPNN method estimates power precisely.

Based on this research it can be concluded that ANFIS is superior tool for power estimation in CMOS VLSI circuits. The ANFIS technique provides an alternative to traditional simulation strategies such as SPICE, which are built on the basis of pre-defined empirical equations with arbitrary parameters. The ANFIS findings are extremely reliable, and no knowledge of circuit structure or interconnections is needed as opposed to BPNN, the experimental finding shows that ANFIS with hybrid optimization using a linear approach yields improved results in terms of testing error, which ranges from 0% to 0.86%. ANFIS is capable of generating a FIS with linear relationship between input and output results. As a result, it appears that ANFIS with exclusive characteristic is a safer alternative power estimation in CMOS VLSI circuits. With a low RMSE of 0.0002075 and a strong coefficient of determination of 0.99961, it has ben shown that ANFIS is ideally adapted for power estimation utilizing statistical estimation.
such as RMSE and coefficient of determination (F).

References


17. Parallel Distributed Processing, Rumelhwt, D.E. and Mc Clelland,


