Voltage and Temperature Scalable Gate Delay and Slew Models Including Intra-Gate Variations

Bishnu Prasad Das[1]*, Janakiraman V. [2], Bharadwaj Amrutur[2], H.S. Jamadagni[1], N.V. Arvind[3]
1 C.E.D.T., Indian Institute of Science, Bangalore, India.
2 Dept of E.C.E., Indian Institute of Science, Bangalore, India.
3 Texas Instruments, Bangalore, India

Abstract

We investigate the feasibility of developing a comprehensive gate delay and slew models which incorporates output load, input edge slew, supply voltage, temperature, global process variations and local process variations all in the same model. We find that the standard polynomial models cannot handle such a large heterogeneous set of input variables. We instead use neural networks, which are well known for their ability to approximate any arbitrary continuous function. Our initial experiments with a small subset of standard cell gates of an industrial 65nm library show promising results with error in mean less than 1% , error in standard deviation less than 3% and maximum error less than 11% as compared to SPICE for models covering 0.9-1.1V of supply, -40^0C to 125^0C of temperature, load, slew and global and local process parameters. Enhancing the conventional libraries to be voltage and temperature scalable with similar accuracy requires on an average 4x more SPICE characterization runs.

1. Introduction

In today's complex industrial designs, both temperature and supply voltage have strong location dependency, i.e. they are non-uniform across the chip. The current practice is to do a worst case design using the extreme values of supply and temperature for timing and power analysis. The authors in [6] argue that this can lead to unnecessarily large margins being included in the design and hence advocate a voltage and temperature aware timing analysis. Such an analysis is now feasible due to the emergence of sophisticated power grid analysis tools, which can predict the supply voltage at any point in the power grid. Similarly, a chip's thermal profile can also be estimated as a function of the computational activity and the heat removal capability of the cooling system. Thus it is possible in principle to use a better estimate of the local supply and temperature [11] at any gate and hence provide more accurate bounds on the gate's and hence the chip's speed.

Due to technology scaling, a large number of process related effects force a wide spread in process parameters, in turn causing a large variation in the chip's delay and power [1] [10] [11]. The existing corner based models leads to over design of the chip. Some of the design margins can be recovered by using dynamic voltage and frequency management (DVFM) as demonstrated by the authors in [7]. Here the supply voltage to the chip is adjusted on a chip by chip basis, as well as for different performance targets over the course of the chip's operations. Thus the chip no longer runs at a given supply, but instead uses a dynamic range of supplies. Timing analysis for such applications provides another motivation for voltage aware gate delay models.

On the analysis side, Statistical Timing Analysis (SSTA) is being advocated for better insights into the chip's delay spread instead of worst case corner based analysis [2] [5]. Most of the existing literature on SSTA [2] [5] [10], only considers gate level variations, in effect having one random process parameter per gate. But the authors in [8] and [9] have shown that ignoring intra-gate variations can introduce substantial errors. Dealing with intra-gate variations involves considering the effect of process variations on each transistor in the gate. For a complex gate with N transistors and M parameters per transistor, we would need NM parameters as input to the delay model. The authors in [9] propose a sensitivity-based enhancement to the linear delay model.

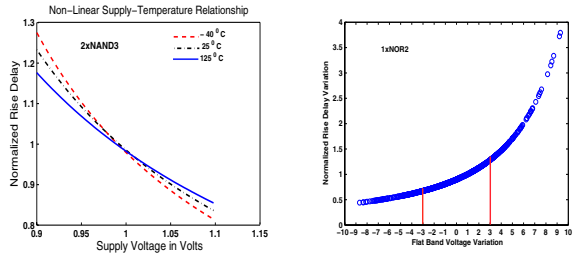
Existing delay models for both static and statistical timing analysis are table based. Usually one delay table per process, supply and temperature corner is created for every gate. The table is indexed by the gate's load and the input edge slew and contains the gate's delay in the corresponding table entry . Similarly, statistical delay models, which are usually linear [2] or quadratic [5], use such tables to store the delay sensitivity coefficients. Extending this approach to cover multiple supply and temperature points at finer granularity for voltage and temperature analysis will be quite expensive in terms of characterization effort (for

*Corresponding Author, Email: bpdas@cedt.iisc.ernet.in

1063-9667/08 $25.00 © 2008 IEEE
DOI 10.1109/VLSI.2008.92

e.g., [7] uses 5mV steps for dynamic supply control from 0.9V to 1.6V).

In this research, we investigate the feasibility of a comprehensive gate delay model which includes load, input slew, supply, temperature, global (inter-die) and local (intra-gate) process variations. However, the delay dependence on supply and temperature is quite non-linear as shown in Fig. 1a. Due to the contrary movements of the mobility and threshold voltage with temperature, the delay actually degrades for lower temperature at low supply voltages. Similarly delay is non-linear with some process parameters like the flat band voltage (Fig 1b). The figure shows the normalized rise delay (for cell 1xNOR2 at 65nm) at 0.9V and 25^0C versus the variations of the flat band voltage from the nominal (of 0), in units of 1σ . The + 3σ points are also highlighted. With increased variations in future technologies, the non-linearity within + 3σ region will increase.



(a) Non-linear relation of delay on temperature and supply (b) Non-linear dependence on flat band voltage

Figure 1.

Standard polynomial models don't work well for these requirements and we need to use a more sophisticated non-linear model. We have investigated neural networks as a modeling template in this work. In the next section, we provide a brief background of neural networks and explain our methodology of characterizing the library cells over large process and environmental ranges. We also present model complexity in this section. In Section 3 we provide some experimental results comparing the accuracy of neural network based models with SPICE. In Section 4 we motivate a few applications of such voltage and temperature models and finally conclude in Section 5.

2. Gate delay modeling using neural networks

Neural networks have been used for over a decade in pattern recognition applications [3] [4]. The ability of a neural network to model complex systems, which are dependent on a number of parameters, makes it a promising candidate for a comprehensive delay model. Thus any continuous function f(x), where x is an input vector, can be modeled very well by a neural network with a single hidden layer [3]. The exact equation for delay of a gate, though complicated to evaluate in closed form, is a smooth, continuous, but non linear, function of the underlying process parameters, load, input slew, supply voltage and temperature and hence it is expected that the delay of a gate can be modeled well by a neural network with a single hidden layer.

Mathematically, the neural network can be described as a real valued multi-variable function of input variables X_1, ..., X_n, as :

$$\hat{F}(X) = \sum_{j=1}^{M} \phi_j \left(\sum_{k=1}^{N} X_k W_{jk} + b_j \right) a_j + b_0 \quad (1)$$

Where, W_{jk} is the weight between the k^{th} input and j^{th} hidden node
a_j is the weight between the j^{th} hidden node and the output layer
X is the vector of inputs $X_1, \ldots, X_n$ to the network
b_j is the bias to the j^{th} hidden node
b_0 is the output layer bias
The activation function $\phi(t)$ used in our work is the tan sigmoid function given by

$$\phi(t) = \tanh(t) \quad (2)$$

The model is fit to sample delay data by adjusting the weights and bias values to give the minimum fitting error [3]. For timing analysis, one needs four delay parameters per gate: rise delay, fall delay, output rise slew and output fall slew. These are functions of the load, input slews, supply, temperature and process parameters. Hence we create four different neural networks, one for each of the above delay parameter.

$$D = F(G, L, Load, Slew, Supply, Temperature) \quad (3)$$

We consider only the global process variations (G) and local process variations (L) for the model. Global variations are also known as inter-die or Die to Die variations. Local variations are also called intra-gate or Mismatch variations. The other set of variations are the intra-die (or within die) variations and these are spatially correlated. These can be handled by partitioning the chip into grids and can be decorrelated using the technique in [2]. After decorrelation, they become like inter-die parameters and can be handled within our model. The number of global process parameters for our process is 8 and there are 2 local parameters per transistor in each gate.

The procedure for gate delay model creation using neural network is described as follows. The first step is to generate samples from SPICE simulations of the gate to use for training and testing the gate's neural network model. For our examples, we have used the layout extracted library cells (using STAR-RCXT) for the SPICE simulations. All the sample data is normalized prior to use for model training or testing as follows. For any input parameter X, its normalized version X_{nom}, is scaled to lie between 0 and 1 as follows

$$X_{nom} = \frac{X - min(X)}{max(X) - min(X)} \quad (4)$$

The output of the network is scaled as follows:

$$D_{nom} = \frac{D}{max(D)} \quad (5)$$

Data from 1000 SPICE runs for the cell is used to train the network. Data from a further 900 SPICE runs is used to test the error between SPICE output and that predicted by the created model. If the error is unacceptable, then either the number of hidden nodes or the number of epochs or both is increased and retraining with the original 1000 SPICE samples is done [11]. In our experiments with 11 different standard library cells, we have found a successful model within 5 iterations, with the most complicated model having 26 hidden nodes (see Table I). We have used MATLAB's built in toolbox to train the network and no gate took more than 1000 seconds to train.

2.1 Sample selection for Modeling

Selecting the appropriate samples from the feasible space of input parameters for modeling is a crucial task. Unfortunately, finding an optimal training sample set is an open problem. Here, we have employed both uniform and random sampling. The range of input slew taken is 1ps-500ps. Parameters like load and input slew are sampled uniformly within their bounds to create about 36 sample points. These are combined with five sample points for supply (0.9, 0.95, 1.0, 1.05, 1.1) and four sample points for temperature (-40^0C,0^0C, 100^0C, 125^0C), to result in about 720 combinations for the load, slew, voltage and temperature. Another 280 samples are taken from uniform distribution between their bounds to obtain a total of 1000 samples needed for training. All the global and local process parameters are samples from a zero mean unit variance Gaussian distribution and are used to generate samples for Monte Carlo SPICE simulations of the standard cell gate. An additional set of 900 SPICE simulations have been performed to test the models. This is used during the model training phase. These include 9 corners for three temperatures (-40^0C, 25^0C and 125^0C) and three supply voltages (0.9v, 1.0v and 1.1v) constituting nine combinations. In each combination of 100 samples, load and slew are taken from uniform distribution from their bounds and process parameters are varied from Gaussian distribution with mean 0 and standard deviation 1. Hence a total of 1900 SPICE characterizations need to be done per gate, to create a NN model for that gate.

Our standard cell library contains cells with different drive strengths e.g. 1xNAND2, 2xNAND2, etc. If the implementation of the larger drive strength is done using more segmented transistors, it increases the number of local parameters for the gate. In our case, we have eight (4*2) local process parameters in 1xNAND2 and sixteen (8*2) for 2xNAND2.

We reduce the number of local process parameters by about half by only considering the transistors that get sensitized for a particular transition. For example, in the case of an inverter, we use local process parameters of PMOS for rise delay/slew modeling, and local process parameters of NMOS for fall delay/slew modeling.

Table 1. Number of hidden node used by the model for different gates

Gate Types	RD Hidden node	FD Hidden node	RS Hidden node	FS Hidden node
1xNAND2	19	18	19	18
2xNAND2	19	18	19	24
2xNAND3	24	26	19	25
1xNAND4	19	21	19	19
1xNOR2	19	19	19	20
2xNOR2	18	18	17	20
1xNOR3	19	19	19	20
2xNOR3	21	18	19	20
1xNOR4	20	21	18	19
1xXOR	22	20	22	18
1xINV	20	13	20	13

Table 1 shows the number of hidden nodes (a measure of the model complexity) for 11 gates from the std. cell library, for rise delay (RD), fall delay (FD), rise slew (RS) and fall slew (FS).

2.2 Model evaluation complexity

The gate delay model is meant to be used for static and statistical timing analysis, during which the most fundamental operation is model evaluation for a given set of input parameters. With N inputs and H hidden nodes, the number of double precision multiplications, additions and tanh evaluations is (N+1)*H, (N+1)*H and H respectively. A single evaluation of tanh function is equivalent to 39 floating point operations. The two most complex models from the table above are the 2xNAND3 and 1xXOR. A single evaluation of each is equivalent to 1612 and 1276 floating point multiplications respectively. The additional cost of two double precision additions and divisions (Equations 4,5) needed to be carried out on the input and output parameters for normalization and renormalization can be amortized over a large number of model evaluations. While the NN model evaluation is at least three orders of magnitude faster than SPICE, further reduction of this computational cost is a topic of ongoing research.

3 EXPERIMENTAL RESULTS

The quadratic polynomial model is created as follows:

$$D = \sum_{i=1}^{N}\sum_{j=1}^{N} c(i,j) X_{nom}^i X_{nom}^j + \sum_{i=1}^{N} d(i) X_{nom}^i + e \quad (6)$$

The $c(i,j), d(i)$ and e are constants determined using the MATLAB's lsqcurvefit function. We find that for a given output load, input slew, supply and temperature, the quadratic model gives good accuracy for global and local process variations. However, the model fails to work well when the load, slew, voltage and temperature are incorporated, due to the inherently large non linearity. Fig.2 shows the maximum % error in the model by NN and the quadratic polynomial [5] for different gates in the library when compared to SPICE when tested over data from an additional 8000 sample points which are different from the 1900 used for the model creation. As expected the polynomial model shows a large error of about 100% for this application. The maximum error with NN model is less than 11% over the 11 gates shown in the Fig.2. Characterization of the other gates in the library is on going and we expect the neural network model to work as well for those based on the theory [4] and our experience with the gates shown in the figures.

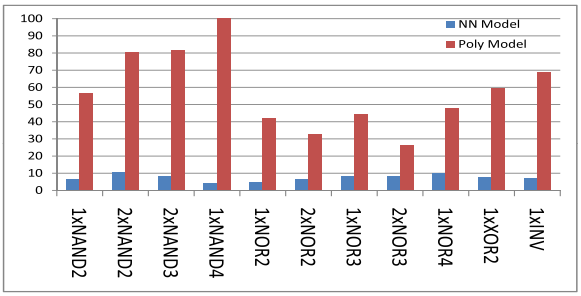


Figure 2. Comparison of maximum % error by NN and Polynomial model

Fig.3 shows the % error at extreme voltage and temperature corners for fall slew of 2xNOR2 gate. We have chosen nine corners with voltage at (0.9v, 1v and 1.1v) and temperature at (-40^0C, 25^0C, 125^0C). At each corner point, 1000 samples are generated with different output load and input slew with uniform distributions between their bounds, global and local process parameters are taken from Gaussian distribution. The figure shows that the maximum error is within 8% . Since the model works with acceptable error bounds at all the nine corners, we expect it to work well at any intermediate point based on the interpolation properties of the network [4].

Fig.4 shows the voltage scalable PDFs generated by SPICE, NN model and quadratic model at 0.9v and 1.1v for fall slew of 2xNAND3 cell. The figure clearly depicts the closeness of SPICE and NN model. The quadratic model is unable to generate accurate PDF. As expected, the spread in PDF is more at 0.9v as compared to spread at 1.1v due to the larger impact of process variation at reduced gate overdrive (supply voltage minus threshold voltage). Fig.5 shows the temperature scalable PDFs generated by SPICE, NN and the quadratic polynomial model for fall slew of 2xNOR2

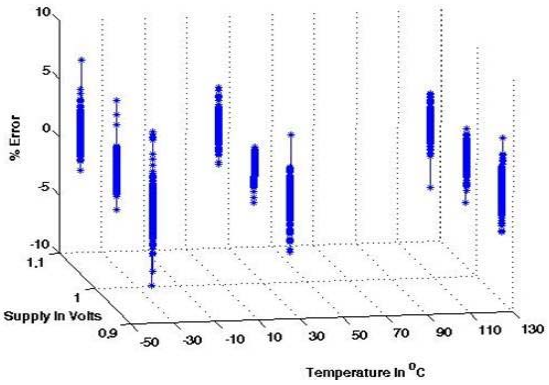


Figure 3. % error at nine corners of voltage and temperature for 2xNOR2 cell

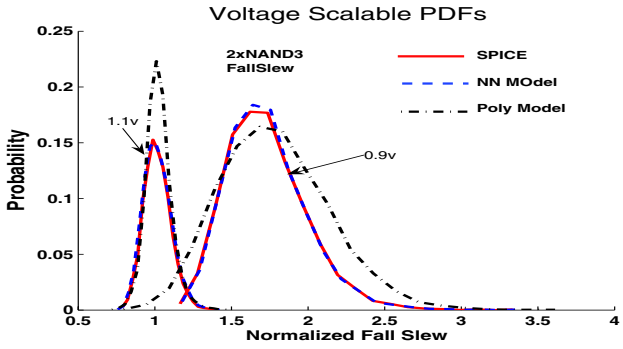


Figure 4. Comparison of voltage scalable PDFs

cell at supply voltage of 1.1v. The PDF from the NN model matches well with SPICE, unlike the quadratic model.

4. Applications

4.1 Voltage and temperature aware static timing analysis

The delay models can be used for voltage and temperature aware timing analysis [6] as well as analysis for dynamic voltage scaling applications. They can replace the corner based delay tables currently being used for static timing analysis. With such tables, interpolations need to be done to obtain values at intermediate parameter points. A small interpolation error necessitates more SPICE characterizations at finer input parameter granularities. Whereas the neural network naturally does good interpolation for arbitrary cell supply voltage and temperature. Such evaluations can done either real time during the analysis or alternatively, the model can be used to generate delay tables at arbitrary voltage and temperature points, thus saving costly SPICE characterization time.

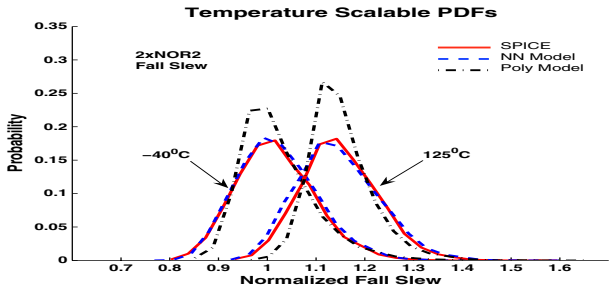


Figure 5. Comparison of temperature scalable PDFs

4.2 Voltage and temperature aware statistical timing analysis

We can extend the work in [6] to perform statistical timing analysis using the IR drop profile and/or the thermal profile in the chip. Fig.6 shows the Cumulative Distribution Function (CDF) of the rise delay for a path containing ten 1xNAND2 gates for three different voltages obtained via Monte Carlo evaluation of the NN model. The results match closely with the Monte Carlo evaluations done using SPICE. We can see that at the normalized rise delay of 2, one can get 100% yield at 1.1v, 90% yield at 1.05v and 40% yield at 1.0v. Thus one can do yield analysis for dynamic voltage scaling applications.

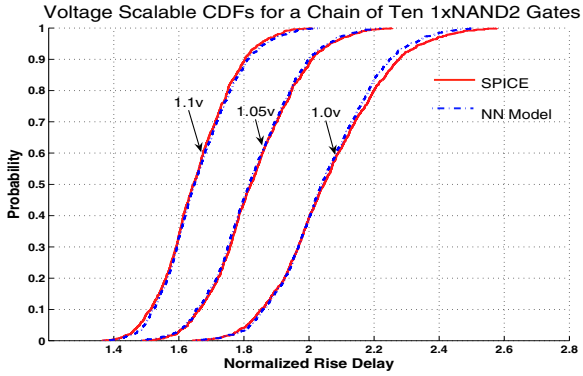


Figure 6. Voltage Scalable CDFs Comparison between NN and SPICE

One of the main arguments in [6] for voltage aware static timing analysis was that valuable excess margin put in by designers can be reduced. We will next show that doing voltage aware statistical analysis can help reduce excess margin further. We demonstrate this on a critical path of ISCAS c7552 consisting of 50 stages using the profile from [6] (Fig.7). In this profile, the voltage is assumed to be constant for five stages of gates and decreases in the steps of 0.05v till the middle of the critical path and increases in the steps of 0.05v till end of the path. We have done four different kinds of analyses on this critical path:

(1)Voltage is worst and process is worst (conventional worst case).

(2) Voltage is taken from profile and process is worst ([6]).

(3) Voltage is worst and process is statistical (conventional statistical).

(4) Voltage is taken from the profile and process is statistical (proposed).

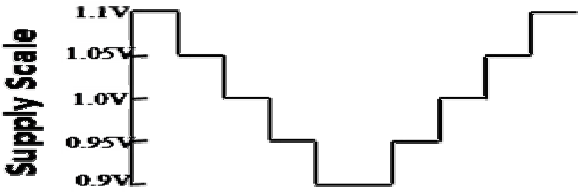


Figure 7. Voltage profile

Fig.8 shows normalized delay (in FO4 inverter delay) of the critical path of ISCAS c7552 benchmark for different types of analyses. Conventional worst case (case 1) has 47% excess margin compared to the proposed case 4. The approach in [6] (case 2) helps reduce the excess margin to 33% while the conventional statistical analysis (case 3) reduces the excess margin but still leaves a valuable 17% of margin on the table, compared to the more accurate voltage aware statistical analysis (case 4). We expect further savings in excess margins when accurate temperature profile is also considered. This clearly motivates the need for voltage and temperature scalable models to enable such statistical analysis

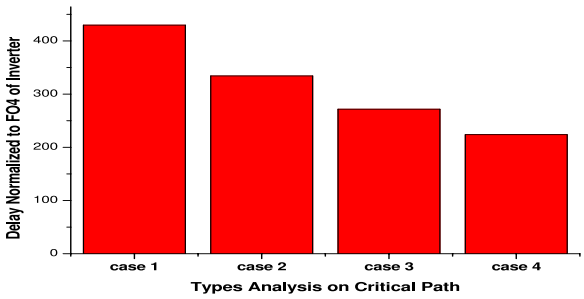


Figure 8. Different types of analysis on ISCAS c7552 critical path

A linear or quadratic sensitivity based delay model is quite accurate for a given supply, temperature, load and input slew [2] [5]. A table based approach for storing these sensitivities across a range of these conditions leads to large interpolations errors, unless fine gridding of these parameters are used. But this would in turn increase the SPICE characterization time to obtain these tables if one wanted to cover large ranges of voltage and temperature. Instead these sensitivities can be accurately generated by the neu-

ral network model using just one model evaluation per linear sensitivity calculation. Fig.9 shows delay sensitivity to flat band voltage generated by NN model and SPICE across a range of voltages at three different temperatures for the 1xNOR2 cell. The sensitivity values match quite well with SPICE. We can also see that the delay is more sensitive to temperature at lower supply voltage due to reduced gate overdrive.

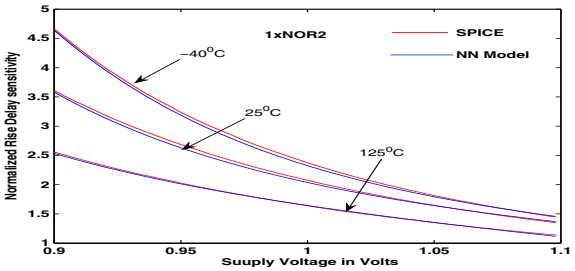


Figure 9. Delay Sensitivity comparison between NN and SPICE

Table 2 compares the number of SPICE runs needed to characterize the eleven cells used in this study for standard delay library format (.lib) and the NN model. NN model requires 1900 SPICE runs independent of types of gates. But .lib characterization needs different SPICE runs for different gates because of the intra-gate process parameters. We have taken a table of 6x6 for input slew and output load. One nominal simulation is needed at each slew and load point. There are 8 global process parameters and 2 local process parameters per transistor. There will be 9 corners due to three supply voltage points (0.9v, 1.0v, 1.1v) and three temperature points (-40^0C, 25^0C, 125^0C). In case of inverter, the number of SPICE runs required is (1+8+4)*6*6*9=4212. There is a minimum saving of 2.21x SPICE runs in case of inverter and maximum saving of 5.62x in case of 2xNAND3 cell.

5 Conclusion

A single delay model which is a function of supply voltage, temperature, inter-die and intra-gate process parameters will enable voltage and temperature aware static and statistical timing analysis. A neural network is a good model template for delay modeling. An initial experiments on a subset of 11 gates from a 65nm standard cell library showing maximum error of less than 11% compared to SPICE, over a large voltage, temperature, load and slew range. A quadratic polynomial model shows errors of up to 100% for the same range.

The neural network models can be created with less number of SPICE runs than that required for table lookup based models. Hence these can be used to efficiently generate table based models for delays as well sensitivities for the linear and quadratic statistical delay models for any supply, temperature, load and slew conditions. This will enable efficient voltage and temperature aware statistical timing analysis as well as yield analysis under dynamic voltage scaling framework.

Table 2. The SPICE run required for .lib creation vs NN model creation

Gate Types	SPICE Run for .lib creation	SPICE run for NN model creation	Saving in SPICE Run
Inverter	4212	1900	2.21
1xNAND2	5508	1900	2.89
2xNAND2	8100	1900	4.26
2xNAND3	10692	1900	5.62
1xNOR2	5508	1900	2.89
2xNOR2	8100	1900	4.26
1xNOR3	6804	1900	3.58
2xNOR3	10692	1900	5.62
1xNOR4	8100	1900	4.26
1xNAND4	8100	1900	4.26
1xXOR2	8100	1900	4.26

References

[1] S. Borkar et al. Parameter variations and impact on circuits and microarchitecture. In *DAC*, pages 338–342, 2003.

[2] H. Chang and S. Sapatnekar. Statistical timing analysis under spatial correlations. *IEEE Transcation CAD*, 24(9):1467–1482, Sept. 2005.

[3] S. Haykin. *Neural Network A comprehensive foundation*. PHI,, New Delhi, India, 1999.

[4] K. Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, (4):251257, 1991.

[5] V. Khandalwal and A. Srivastava. A general framework for accurate statistical timing analysis considering correlation. In *DAC*, pages 89–94, 2005.

[6] B. Lasbouygues, R. Wilson, N. Azemard, and P. Maurine. Temperature and voltage aware timing analysis. *IEEE Transcation CAD*, 26(4):801–815, April 2007.

[7] H. Mahmoodi, S. Mukhopadhyay, and K. Roy. Dynamic voltage and frequency management for a low-power embedded microprocessor. *IEEE JSSC*, 40(1):28–35, Jan 2005.

[8] H. Mahmoodi, S. Mukhopadhyay, and K. Roy. Estimation of delay variation due to random-dopant fluctuations in nanoscale cmos circuits. *IEEE JSSC*, 40(9):1787–1796, Sept. 2005.

[9] K. Okada, K.Yamaoka, and H.Onodera. A statistical gate-delay model considering intra-gate variability. In *ICCAD*, pages 908–913, 2003.

[10] A. Srivastava, D. Sylvester, and D. Blaauw. *Statistical analysis and optimization for VLSI: Timing and Power*. Springer,, USA, 2005.

[11] H. Su et al. Full chip leakage estimation considering power supply and temperature variations. In *ISLPED*, 2003.