Power-Simulation of Cell Based ASICs: Accuracy- and Performance Trade-Offs

Dirk Rabe[†], Gerd Jochens[‡], Lars Kruse[‡], Wolfgang Nebel[†‡]

[†]Carl von Ossietzky University Oldenburg
FB 10 – Department of Computer Science
D-26111 Oldenburg, Germany
<LASTNAME>@Informatik.Uni-Oldenburg.DE
http://EIS.Informatik.Uni-Oldenburg.DE/

[‡]OFFIS
Escherweg 2
D-26121 Oldenburg, Germany
<LASTNAME>@OFFIS.Uni-Oldenburg.DE
http://WWW.OFFIS.Uni-Oldenburg.DE/

Abstract

Within this paper the gate-level power-simulation tool GliPS (Glitch Power Simulator) is presented, which gives excellent accuracy (in the range of transistor-level simulators) at high performance. The high accuracy is achieved by putting emphasis on delay- and power-modelling. The impact of these modelling factors on accuracy and performance is demonstrated by comparing GliPS to other tools on circuit-level and a simple toggle count based power simulator TPS on gate level.

1. Motivation

Modern IC-production lines offer deep submicron technologies. With the ever shrinking structures new possibilities arise for a higher degree of functional integration, more complex applications in more handsome cases. For marketing, environmental and reliability reasons a low power consumption is gaining importance. The time of operation for portable battery applications is limited by its energy consumption. With the increasing power consumption of complete chips cooling problems arise which dramatically influence the packaging and its costs.

The power reduction of an application can be achieved by technology improvements, voltage scaling [1] and design decisions for low power [2]. Because of the high demands on lowering energy consumption all possibilities must be exploited.

Within design for low power the power consumption of a certain design solution needs to be evaluated. Therefore power estimation tools are available on different levels of abstraction: circuit-level, gate-level, RT-Level and currently some investigations also face higher levels. Power calculation is most accurate on the lower levels. However, this accuracy has to be paid with lower perform-

ance. Which level of abstraction fits best to calculate power clearly depends on the given constraints (accuracy, calculation time, information available of the design).

Besides the validation of certain design solutions, tools on lower levels of abstraction are needed for characterizing higher level modules. For RT-module characterization one has the choice to take tools on circuit- or gate-level. Even though this characterization has to be done only once for a module library, the use of SPICE-like tools is commonly not feasible due to the module's high complexity. On the other hand errors in accuracy during the characterization process will decrease the simulation accuracy on higher levels.

Within this paper we compared the following gate- and circuit-level tools with each other by applying them to a set of ISCAS'85- and datapath-module benchmark circuits:
- HSPICE (Version 93a) from META-Software,
- PowerMill Version 5.1 from EPIC
- Glitch Power Simulator GliPS from OFFIS,
- Toggle Power Simulator TPS from OFFIS.

GliPS is a stand-alone event-driven simulator, which puts high emphasis on accurately modelling signal waveforms including incomplete transitions (defined as glit-
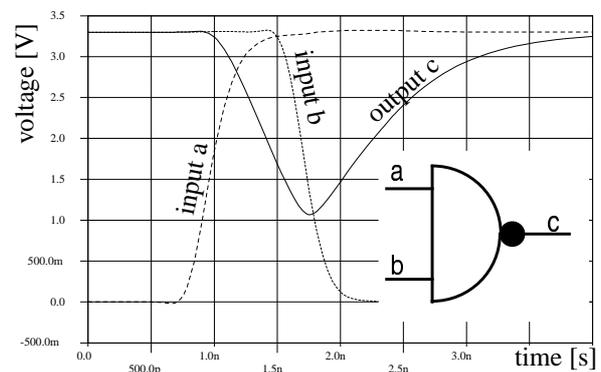

Figure 1: Example for a glitch at a NAND2-gate

ches, cf. fig. 1) [8]. TPS calculates power from simple toggle-count information, which is extracted from VERI-LOG-XL simulations [10]. TPS takes output loads and precharacterized gate internal power losses into account. Within the VERILOG-XL simulation an inertial delay model and a SDF (considering slope-effects) are used. A wide range of accuracy and simulator performance is covered by these four simulators.

Within this paper a glitch is defined as a pair of two or more colliding output waveforms which are so close together that the corresponding voltage waveform neither reaches V_{SS} nor V_{DD} in between (cf. fig. 1). The energy consumption of a glitch is usually less than that of the underlying complete transitions and hence must be calculated differently. The glitch model [8] which is used within GliPS includes the determination of essential data to consider glitches within the energy calculation.

In the next section some parameters for gate level power modelling are introduced. In section 3 delay modelling alternatives, which influence accuracy of activity estimation, are dealt with. In section 4 the impact of these modelling parameters will be evaluated by analysing benchmark simulations. In the final section conclusions are drawn.

2. Power calculation of CMOS gates

The average power consumption of a single CMOS gate can be divided into three parts:

$$P = P_{\text{leakage}} + P_{\text{short-circuit}} + P_{\text{Cap}}$$

The power consumption due to leakage currents is much smaller than the other two dynamic components and hence it is often neglected within power calculation. This is true except for very low voltage supplies and as a consequence for very low threshold voltages [2]. The impact of leakage currents is not further discussed in this paper.

During switching a conducting path through the pull-up- and pull-down network of a gate is present and as a consequence a short-circuit current is occurring. The third component is the capacitive component which takes into account the capacitive loading of switched capacitors. The short-circuit charge Q_{SC} is often considered together with loading of gate-internal capacitors (not including fanin- and fanout-capacitors) as internal charge. If the internal charge is used within characterization, the capacitive component only contains the fanout-capacitor (including fanin-capacitors of consecutive gates).

We first discuss single complete output-transitions which are caused by single input-transitions. This model will afterwards be extended to consider glitches (i.e. partial transitions).

The short-circuit charge respectively the internal charge consumption depends on voltages at gate-internal nodes, the transition causing input-port, the switching output-port(s), possibly some stable logic input-values, input-slopes and output-loads.

Within TPS and GliPS precharacterized internal charge consumption data is available for each cell. TPS uses a constant internal charge per gate-transition and as a consequence the above given effects can only be considered on average. Within GliPS linear interpolation is used to get the desired charge and delay values from precharacterized data, which is organized in lookup tables. Within these tables all of the above given effects can be modelled. However, the voltage at gate-internal nodes, which are not uniquely given by the in- and output logic assignments, are not considered. The characterization data is derived from circuit-level simulations of single gates.

If a partial transition occurs instead of a complete transition the capacitive energy consumption is:

$$E_{\text{GlitchCap}} = 1/2 \cdot V_{DD} \cdot |Q_{\text{Cap}}| \qquad (1)$$

$$= 1/2 \cdot V_{DD} \cdot C_L \cdot |\Delta V| = E_{\text{complete tr.}} \cdot \frac{|\Delta V|}{V_{DD}}$$

This equation also holds for complete transitions with $\Delta V = V_{DD}$. The capacitive power calculation is straight forward:

$$P_{Cap} = \frac{1}{2} \cdot V_{DD} \cdot C_L \cdot \lim_{T \to \infty} \frac{\sum_i |\Delta V_i|}{T} \qquad (2)$$

The sum $\sum \Delta V_i$ can be obtained by logic simulation over a *sufficient* time interval [7] using a gate-level glitch model [8]. Equation 2 holds for all kind of glitches. As an example a dynamic glitch which consists of three ramps is illustrated in figure 2.
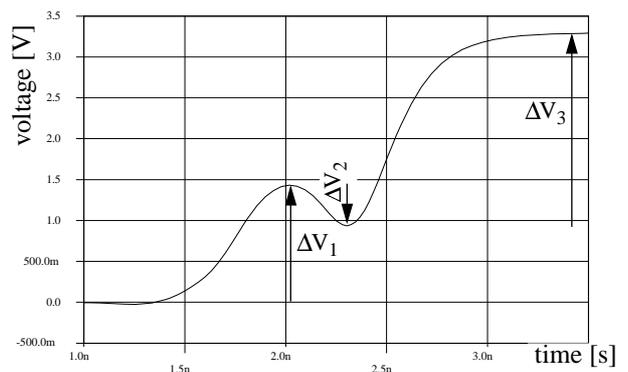

Figure 2: Example for a dynamic glitch

As a rough approximation the internal charge, which is consumed due to a complete output-transition, is also scaled by $\Delta V_i / V_{DD}$ within GliPS.

3.   Delay modelling of static CMOS

Within the last section it was emphasised that dynamic power-consumption of CMOS gates is dominant. Hence the activity calculation plays an important role within power simulation. During one computation cycle circuit nodes may multiply switch due to different path delays. This leads to unnecessary power consumption which can be as high as 65% for arithmetic units [6]. In order to consider this unnecessary switching the delay model plays an important role. Using a zero-delay model, only necessary power is calculated. Within the unit-delay model real delays are approximated in a very rough way, leading to path-delays which may have nothing to do with reality.

Using real-time delays two conventional ways of delay modelling are available: transport delay model and inertial delay model. Within the transport delay-model all input-events of a gate, causing a transition at the output-port(s) are propagated to the consecutive gates. This leads to an overestimation of activity. Within the inertial delay model output-pulses, which are shorter than a given threshold, are filtered.

The switching data, which is calculated using the transport or inertial delay model, does not include information about partial transitions. Hence each transition is associated with a V_{DD}-swing when power is calculated according to equation 2. In case of a partial transition, which does not start from V_{DD} respectively V_{SS}, the delay will be shorter than the characterized one for a full transition. I.e., the pulse is smaller in reality than the pulse derived from the inertial respectively transport delay model (cf. fig. 3). The reduction of pulse width is important for possible glitch filtering in consecutive gates.

These two phenomena are considered within the enhanced gate-level glitch model [8], which GliPS is based on. Different existing enhanced glitch modelling algorithms have been compared in [9]. The main important differences to conventional gate-level simulators are:
- a transition is modelled by linear ramps,
- ramps are derived from delay and slope information,
- ramps are scheduled into the event-queue of the simulator when the ramp *start*,
- glitches are presented by ramps crossing each other.

The ramp- and power calculation is done dynamically during simulation. This allows GliPS to consider different input-slopes at an instance's input-pin. Input ramps are generated by a driving gate's output. A static calculation of the driving gate's output-slope would require the averaging of all possible driver's input-slopes for all possible input-pins. As a consequence the accuracy would be decreased.
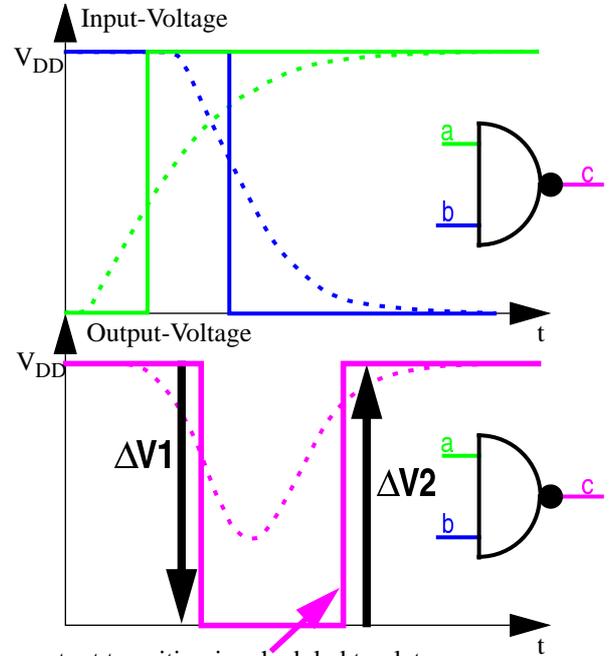
output transition is scheduled too late
$\Delta V1$ and $\Delta V2$ are modeled too large!

Figure 3: Modelling a glitch with the inertial or transport-delay model

4. Practical results

The simulators HSPICE (Vers. 93a), PowerMill (Version 5.1), GliPS and TPS were evaluated for the benchmark circuits given in Table 1. The first 3 benchmark circuits were generated from Synopsys' DesignWare and contain complex gates like Full-Adders. The ISCAS'85 benchmarks consist of basic gates (like AND, OR, NAND, NOR, EXOR) only. The designs are mapped on Atmel ES2's 1.0μm process and layout extracted data is available. Interconnects were modelled by single capacitors within HSpice and PowerMill.

For TPS delay calculation was done using the Cadence Delay Calculator (SDF enhanced wire delay model), which statically considers input-slope and output-load effects.

Within PowerMill the transistor-characterizations were run in advance (not included in performance data) and two alternatives were distinguished:
- accurate mode: the following options were applied:
  `set_sim_spd 0.2` and `set_powr_acc 1`,
- default mode: no user defined options were used.

In table 2 the achieved accuracies of charge-consumption and in table 3 the simulator performances are reported with HSPICE as reference. It was not possible to simulate all pattern within one simulation run using HSPICE. As a consequence the simulations were split into

library	module name	function	no. of simul. random pattern	no. of primary inputs	no. of prim. outputs	circuit depth	no. of cells	layout-size [mm^2]
Design-Ware	ash	arithmetic shifter	1997	19	16	12	181	0.3
	mult	multiplier	1942	16	16	24	207	0.42
	sin	combinatorial sine	2000	8	8	25	196	0.36
ISCAS'85 bench-marks	c17	-	1056	5	2	4	6	0.008
	c499	ECAT	1000	41	22	13	202	0.38
	c1355	ECAT	1000	41	32	26	546	0.65
	c3540	ALU&contr.	500	50	22	48	1669	2.40
	c6288	16bit mult.	500	32	32	123	2406	3.21

Table 1: Used benchmark circuits for evaluation

module name	Hspice Q/Tr. [pC]	TPS ε_Q [%]	PowerMill accur. ε_Q [%]	PowerMill def. ε_Q [%]	GLiPS ε_Q [%]
ash	72.7	24.6	13.7	26.3	10.7
mult	184.1	15.3	3.85	13.7	-0.23
sin	77.7	21.2	9.77	21.8	11.9
$\varnothing=\Sigma|\varepsilon|$		20.4	9.11	20.6	7.71
c17	1.6	11.5	1.2	10.5	-6.7
c499	102	1.63	0.91	10.7	7.33
c1355	197	30.6	1.73	12.6	-0.97
c3540	1049	18.1	-6.89	0.66	2.79
c6288	3232	112	1.64	9.89	8.20
$\varnothing=\Sigma|\varepsilon|$		34.8	2.47	8.87	5.20

Table 2: Accuracy of charge consumption

module name	HSpice time / pattern [s]	TPS spd up	PowerMill accur. spd up	PowerMill def. spd up	GLiPS spd up
ash	148.78	45638	261	401	7051
mult	438.62	122863	332	533	13969
sin	116.08	42520	183	292	6036
c17	4.29	2258	223	263	3830
c499	170	31036	214	321	6540
c1355	283.2	32036	230	350	4240
c3450	3760	115984	743	1146	15732
c6288	9869	73108	360	535	3579
$\varnothing=\Sigma|\varepsilon|$		58180	318	480	7622

Table 3: Simulator Performance

several runs (each inluding initialization time). The TPS accuracies are better than 31% for all circuits except c6288, which has an untypical circuit depth of 123 gates. The speed-up of TPS is 58180 with HSPICE as reference. The PowerMill deviations are below 7% for the ISCAS'85 benchmarks (including c6288) in accurate mode. Using the default mode the accuracies for the ISCAS benchmarks are worse by a factor of 3.6. The inaccuracies of the DesignWare benchmarks raise up to 13.7% (26.3%) in accurate (default) mode. The speed-up is 310 (480) in average. The accuracies of GLiPS are in the same range as the PowerMill results. However the speed-up over PowerMill is more than one order of magnitude.

Two sources of error in accuracy can be distinguished: Errors in activity estimation and errors due to the power model.

In table 4 the activity accuracy of the default PowerMill mode, TPS and GLiPS is compared to the accurate PowerMill mode results. For TPS a high portion of charge-estimation error results from activity errors.

The decrease in accuracy is dramatic especially for the circuits with high circuit-depth (c6288, c3540, c1355).
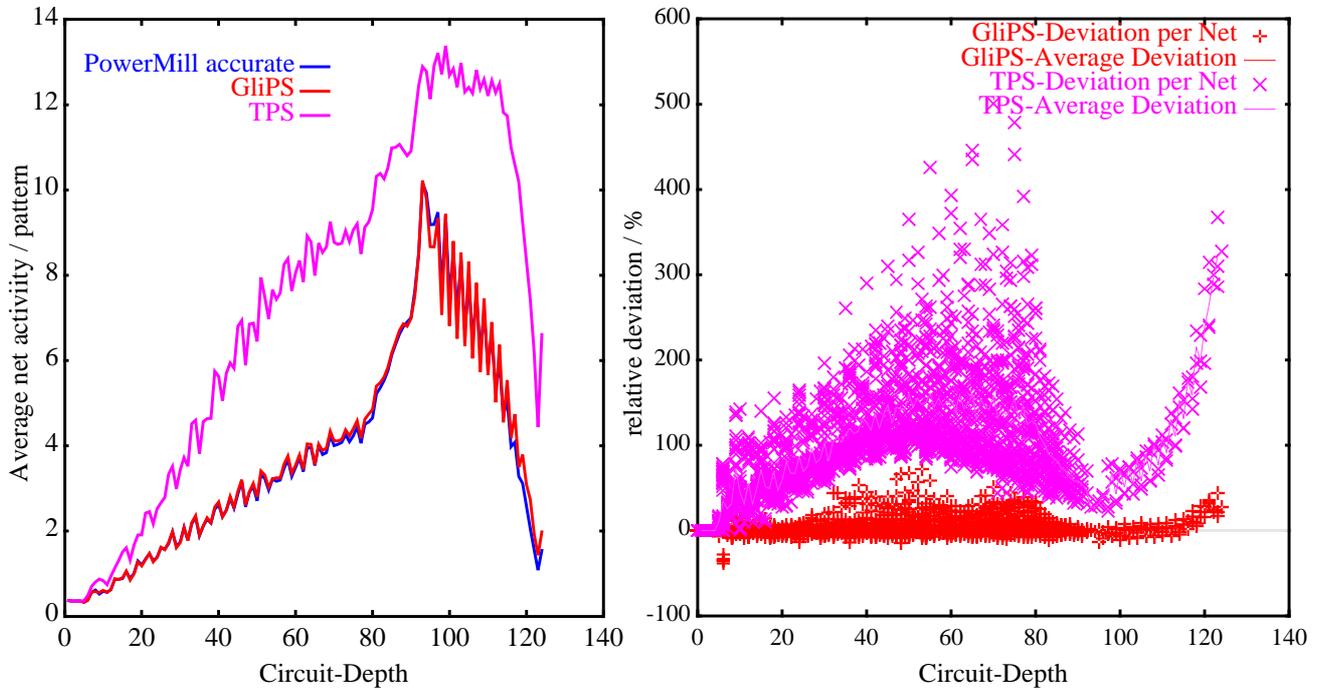
Figure 4: Net-activity (left) and relative accuracy of net-activity (right) as function of circuit depth position

module name	PowerMill		TPS	GliPS
	accurate mode	default mode		
	net-activity	ε_{act} [%]	ε_{act} [%]	ε_{act} [%]
ash	0.524	0.953	15.28	-1.38
mult	0.646	1.44	-1.25	3.48
sin	0.450	0.942	13.3	-2.41
$\varnothing=\Sigma|\varepsilon|$		1.11	9.94	2.43
c17	0.481	0	3.72	-0.72
c499	0.514	1.37	12.7	-4.33
c1355	0.526	0.7	19.7	-9.23
c3540	0.638	1.23	17.5	0.58
c6288	2.597	3.15	105	1.58
$\varnothing=\Sigma|\varepsilon|$		1.29	31.7	3.28

Table 4: Accuracy of activity estimation

Hence the main source of errors of TPS is the activity estimation inaccuracy. This observation is also documented in fig. 4 (the decrease of activity for circuit-depth positions above 100 is due to the circuit structure). The deeper a net is located within the circuit, the higher the activity estimation errors are. These results also indicate, that the accuracy of local charge estimation for single gates decreases with circuit depth.

module name	PowerMill		TPS	GliPS
	accurate	default		
ash	29	45	67	38
mult	41	34	69	65
sin	84	103	272	109
$\varnothing$	51	61	136	71
c17	39	84	178	51
c499	23	37	54	62
c1355	24	43	84	54
c3450	17	22	56	24
c6288	15	19	147	27
$\varnothing$	24	41	104	44

Table 5: Maximum deviation of charge consumption per pattern in %

The above given accuracy data refer to simulations of a large set of random pattern. The error of charge consumption for single changes of input pattern is typically much higher. The error *may* average out if a large pattern sequence is analyzed. In table 5 the maximum deviation of charge consumption is given. The maximum error of TPS for a single change of input pattern is above 100%. For the ISCAS'85 benchmarks the maximum error of GliPS is comparable to PowerMill (default mode). The more complex gates which are used for the DesignWare modules, were modelled as black box components within GliPS and TPS (even though GliPS is capable to model these gates more accurately). For this reason the maximum deviation of GliPS is higher than PowerMill for the DesignWare modules. Fig. 5 contains a plot with the number of pattern
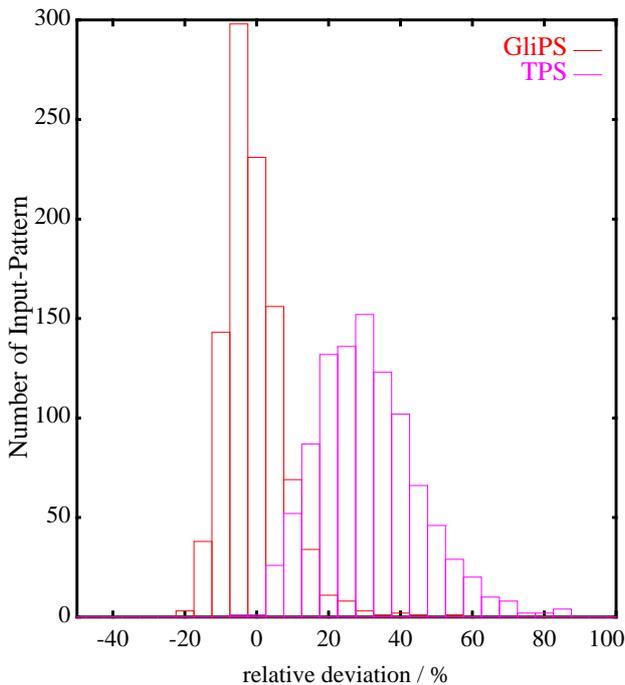

Figure 5: Power accuracy per pattern (c1355)

in a cirtain error intervall. The deviation in average and the variation of the data are higher for TPS than for GliPS.

RT-level power models may contain a large number of parameters, which need to be characterized using lower level simulators (i.e. gate- or transistor-level). Commonly only a subset of the complete set of input-pattern can be used to characterize a specific RT-level power model parameter, which is more error prone than the charge estimation of the whole pattern sequence.

5. Conclusions

Within this paper we have compared different power estimation methodologies at different levels of abstraction. The main important task is to find a good compromise between accuracy and simulation performance for the given constraints. As key point it was observed, that the activity estimation plays a major role. Simple toggle count based gate-level simulators (like TPS) deliver acceptable accuracy (in terms of power and activity) only for circuits with small logical depth. For large and moderate logical depth circuits the delay needs to be modelled more accurately. This is possible using the new gate-level power estimation tool GliPS. Within GliPS delays and power are carefully modelled. Its accuracy is comparable to transistor level simulators running more than one order of magnitude faster. GliPS can also be used for advanced timing analysis.

6. References

[1] B. Davari, R.H. Dennard, G.G. Shahidi. CMOS Scaling for High Performance and Low Power - The Next Ten Years. *Proceedings of the IEEE*, Vol. 83: 595-606, April 1995

[2] A.P. Chandrakasan, R.W. Brodersen. Minimizing Power Consumption in Digital CMOS Circuits. *Proceedings of the IEEE*, Vol. 83: 498-523, April 1995

[3] H.J.M. Veendrick. Short-Circuit Dissipation of Static CMOS Circuitry and its Impact on the Design of Buffer Circuits. *IEEE Journal of Solid State Circuits*, Vol. SC-19 NO. 4: 468-473, 1984

[4] J. Leijten, J. Meerbergen, J. Jess. Analysis and Reduction of Glitches in Synchronous Networks. *Proceedings of the European Design and Test Conference:* 398-403, 1995

[5] M. Favalli, L. Benini. Analysis of glitch power dissipation in CMOS ICs. *Proceedings of International Workshops on Low Power Design*: 123-128, 1995

[6] M.A. Ortega, J. Figueras. Extra Power Consumed in Static CMOS Circuits due to Unnecessary Logic Transitions. *Proceedings of IV International Design Automation Workshop, Russia:* 40-42, 1994

[7] R. Burch, F.N. Najm, P. Yang, T.N. Trick. A Monte Carlo Approach for Power Estimation. *IEEE Transactions on VLSI Systems*, Vol. 1: 63-71, March 1993

[8] D. Rabe, W. Nebel. A New Approach in Gate-Level Glitch Modelling. *Proceedings of EURO-DAC'96 with EURO-VHDL'96*, 66-71, 1996

[9] D. Rabe, B. Fiuczynski, L. Kruse, A. Welslau, W. Nebel. Comparison of Different Gate Level Glitch Models. *Proceedings of PATMOS (Power and Timing Modeling of Integrated Circuits)*, 167-176, 1996

[10] G. Jochens, L. Kruse, W. Nebel. Application of Toggle-Based Power Estimation to Module Characterization. *Proceedings of PATMOS (Power and Timing Modeling of Integrated Circuits)*, 161-170, 1997