Design and Optimization of Low Voltage High Performance Dual Threshold CMOS Circuits *

Liqiong Wei, Zhanping Chen, Mark Johnson and Kaushik Roy
School of Electrical and Computer Engineering
Purdue University, W. Lafayette, IN 47907-1285

Vivek De
Microcomputer Research Labs, Intel Corp., Hillsboro, OR 97124-6461

Abstract

Reduction in leakage power has become an important concern in low voltage, low power and high performance applications. In this paper, we use dual threshold technique to reduce leakage power by assigning high threshold voltage to some transistors in non-critical paths, and using low-threshold transistors in critical paths. In order to achieve the best leakage power saving under target performance constraints, an algorithm is presented for selecting and assigning an optimal high threshold voltage. A general standby leakage current model which has been verified by HSPICE is used to estimate standby leakage power. Results show that dual threshold technique is good for power reduction during both standby and active modes. The standby leakage power savings for some ISCAS benchmarks can be more than 50%.

1 Introduction

With the growing use of portable and wireless electronic systems, reduction in power consumption has become more and more important in today's VLSI circuit and system designs [1], [2], [3].

In CMOS digital circuits, power dissipation consists of dynamic and static components. Since dynamic power is approximately proportional to the square of supply voltage V_{dd} and static power is proportional to V_{dd}, lowering supply voltage is the most effective way to reduce power consumption as long as dynamic power is dominant. With the lowering of supply voltage, transistor threshold voltage should also be scaled in order to satisfy the performance requirements. Unfortunately, such scaling can lead to a dramatic increase in leakage current, which becomes an important concern in low voltage high performance circuit designs.

Multiple thresholds can be used to deal with the leakage problem. This technique has commonly been used in DRAM chips by raising threshold voltages of the array devices with a fixed body bias [5]. For LSI circuits, Multithreshold-Voltage CMOS (MTCMOS) circuit technology was proposed to reduce the standby leakage current by inserting high threshold devices in series to normal circuitry [8]. However, the large inserted MOSFETs will increase the area and delay.

For a logic circuit, a higher threshold voltage can be assigned to some transistors on non-critical paths so as to reduce leakage current, while the performance is maintained due to the low threshold transistors in the critical path(s). Therefore, no additional transistors are required, and both high performance and low power can be achieved simultaneously. Recently, a dual-Vth MOSFET process was developed [6], which makes the implementation of dual-Vth logic circuits more feasible.

However, due to the complexity of a circuit, not all the transistors in non-critical paths can be assigned a high threshold voltage. In order to achieve the best leakage power saving under performance constraints, we present a heuristic algorithm for selecting and assigning an optimal high threshold voltage. A standby leakage model which has been verified by HSPICE is used to estimate the standby leakage power of a circuit. The power dissipations of single-Vth and dual-Vth circuits in active mode are also compared using HSPICE simulations.

2 Delay Model

2.1 Definitions

A combinational circuit can be represented as a directed acyclic graph $G(V, E)$. Each node (except for primary inputs and outputs) in the graph maps to a logic gate in the circuit while each edge maps to a path.

The propagation delay through node x, denoted as $t_p(x)$, defines how quickly the output responds to a change in the input. The propagation delay of a path π_j, denoted as $Pd(\pi_j)$, is the sum of the propagation delays $t_p(i)$ of each node i along this path. It can be expressed as

$$Pd(\pi_j) = \sum t_p(i) \qquad (1)$$

The arrival time ($T_a(x)$) is the propagation delay of each fan-in path of node x. Among all the fan-in paths, there exists a path (or paths) which has a maximum propagation delay value $T_{max}(x)$, where

$$T_{max}(x) = \max_{i \in all\ fanins} \{T_a(x)[i]\} \qquad (2)$$

The departure time ($T_l(x)$) of node x is defined as

$$T_l(x) = T_{max}(x) + t_p(x) \qquad (3)$$

*Acknowledgment: This research is supported in part by DARPA (F33615-95-C-1625), NSF CAREER award (9501869-MIP) & Intel.

Permission to make digital/hard copy of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage, the copyright notice, the title of the publication and its date appear, and notice is given that copying is by permission of ACM, Inc. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

DAC 98, San Francisco, California
©1998 ACM 0-89791-964-5/98/06..$5.00

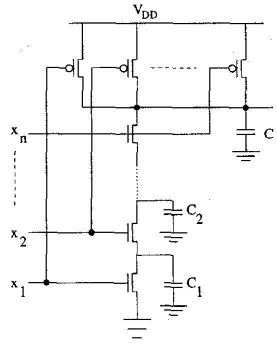

Figure 1: n-input NAND gate

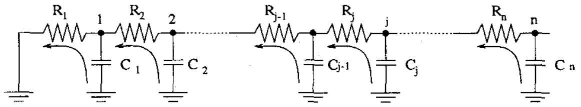

Figure 2: Equivalent pull-down network of the NAND gate

The path which determines the maximum speed of the circuit is called the critical path. There may be more than one critical path. Critical delay ($T_{critical}$) is the delay along the critical path.

2.2 Elmore delay model

Consider an n-input NAND gate (Figure 1). It can be analyzed using an equivalent RC circuit. Figure 2 shows the equivalent RC circuit of the pull-down network(PDN).

The worst case occurs when all C_j's are discharged simultaneously. Based on the Elmore delay model, the worst case delay (t_{PHL}) of the PDN is given by

$$t_{PHL} = 0.69 \sum_{j=1}^{n} (C_j \sum_{k=1}^{j} R_k) \qquad (4)$$

The capacitance of each internal node j ($j = 1 \ldots n-1$) in the n-input NAND gate is given as follows,

$$C_j = 2\, C_{dN} \qquad (5)$$

where C_{dN} is the diffusion capacitance of an NMOSFET. The capacitance of the gate at output is given by

$$C_n = F_O\,(C_{gP} + C_{gN}) + F_I\, C_{dP} + C_{dN} + F_O\, C_{int} \qquad (6)$$

where C_{dP} is the diffusion capacitance of a PMOSFET. C_{gP} and C_{gN} are the gate capacitances of PMOS and NMOS transistors, respectively. C_{int} represents the interconnect capacitance per fan-out. F_O is the number of fan-outs of the gate, while F_I represents the number of fan-ins. For an n-input NAND gate, we have $F_I = n$.

Assuming that each NMOSFET has the same on-resistance, the worst-case delay of the PDN can be simplified as follows

$$t_{PHL} = 0.69[R_N\, C_{dN}\, F_I\, (F_I - 1) + F_I\, R_N\, C_n] \qquad (7)$$

Although the on-resistance depends on the operation point and varies during the switching transient, we still can make a reasonable approximation by using a fixed value. This value is the average of the resistances at the end points of the transitions [7]. The on-resistance of an NMOSFET is given by

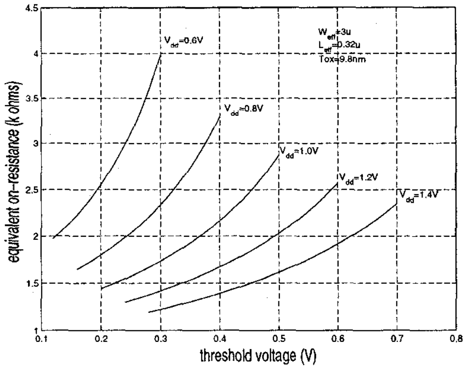

Figure 3: Relationship between R_N and Vth

$$\begin{aligned} R_N &= \frac{R_{NMOS}\,|_{V_{out}=V_{dd}} + R_{NMOS}\,|_{V_{out}=V_{dd}/2}}{2} \\ &= \frac{1}{2}(\frac{V_{DS}}{I_D}\,|_{V_{out}=V_{dd}} + \frac{V_{DS}}{I_D}\,|_{V_{out}=V_{dd}/2}) \\ &= \frac{V_{dd}}{k_N(V_{dd}-V_{TN})^{\alpha}} + \frac{V_{dd}}{k_N[2(V_{dd}-V_{TN})V_{dd} - \frac{V_{dd}^2}{2}]} \end{aligned} \qquad (8)$$

where V_{TN} is the threshold voltage of an NMOSFET and k_N is the gain factor. The constant α is 2 and 1.3 for long channel and short channel MOSFETs, respectively. The relationship between R_N and V_{TN} at different supply voltages is shown in Figure 3. For a PMOSFET, the on-resistance(R_P) can be evaluated similarly. For simplicity, we assume that $|V_{TN}| = |V_{TP}| = |V_{th}|$ and $R_N = R_P$.

For the pull-up network (PUN), the worst case occurs when only one PMOS transistor is "on". The worst case delay (t_{PLH}) can be expressed by

$$t_{PLH} = 0.69 R_P\, C_n \qquad (9)$$

The worst-case propagation delay of a CMOS gate is given by

$$t_p = (t_{PHL} + t_{PLH})/2 \qquad (10)$$

Following similar procedure, we can get the worst-case propagation delay of the other gates.

3 Standby Leakage Power Estimation

In standby mode, the power dissipation is produced by the standby leakage current through each transistor. The leakage current has two sources: reversed-biased diode junction leakage current and subthreshold leakage current. Diode junction leakage is small and can be ignored [7]. Subthreshold leakage exponentially increases with the reduction of threshold voltage, making it critical for low voltage circuits [4]. Therefore, in our simulation, we focus on subthreshold leakage power estimation.

In order to estimate standby leakage power accurately, a general transistor model [11], which considers sub-zero gate-to source voltage (V_{GS}) for NMOS and super-zero V_{GS} for PMOS (occurs when multiple series connected transistors are turned off), body effect and drain induced barrier lowering (DIBL), is used. The following analysis is done for NMOSFETs, but is equally applicable to PMOSFETs.

From BSIM2 MOS transistor model [12], the subthreshold current of a MOSFET can be modeled as

$$I_{sub} = A\, e^{\frac{q}{n'kT}(V_G - V_S - V_{TH_0} - \gamma'\, V_S + \eta\, V_{DS})}\, (1 - e^{\frac{-qV_{DS}}{kT}}) \quad (11)$$

where $A = \mu_0 C_{ox} \frac{W_{eff}}{L_{eff}} (\frac{kT}{q})^2 e^{1.8}$. C_{ox} is the gate oxide capacitance per unit area. μ_0 is the zero bias mobility. n' is the subthreshold swing coefficient of the transistor. V_{TH_0} is the zero bias threshold voltage. The body effect for small values of V_S is very nearly linear. It is represented by the term $\gamma' V_S$, where γ' is the linearized body effect coefficient. η is the DIBL coefficient.

The standby leakage power of a logic circuit can be expressed as follows [7],

$$P_{stdby} = (\sum_i I_{stdby_i})\, V_{dd} \quad (12)$$

where I_{stdby_i} is the standby leakage current through each node i. It depends on the gate topology as well as input signal levels.

Let's consider an NAND gate. Assume the transistors which are turned on are short circuits. If all the inputs are "1", the PDN is shorted and the standby leakage current is determined by the PUN. We can get the source and drain voltages of each transistor in the PUN easily. Using equation (11), the leakage contribution of each transistor in the PUN can be calculated separately and added together. If at least one input is "0", the PUN is shorted and standby leakage current is the leakage though the PDN. Suppose there are n transistors which are turned off in the PDN, the quiescent subthreshold leakage current through each of them must be identical. By equating the leakage current of the transistors in the stack, V_S and V_{DS} of each transistor can be obtained. If $n = 1$, $V_{DS1} = V_{dd}$ and $V_{S1} = 0$. Otherwise, the following equations can be used

$$V_{DS_2} = \frac{n'kT}{q(1 + 2\eta + \gamma')} ln(\frac{A_1}{A_2} e^{\frac{q\eta V_{dd}}{n'kT}} + 1) \quad (13)$$

$$V_{DS_i} = \frac{n'kT}{q(1 + \gamma')} ln(1 + \frac{A_{i-1}}{A_i}(1 - e^{-\frac{q}{kT} V_{DS_{i-1}}})) \quad (2 < i \leq n) \quad (14)$$

$$V_{S_i} = \sum_{j=i+1}^{n} V_{DS_j} \quad (1 \leq i \leq n) \quad (15)$$

$$V_{DS_1} = V_{dd} - V_{S_1} \quad (16)$$

where $i = 1$ represents the top transistor and $i = n$ represents the bottom transistor in the stack. A more detailed derivation of the above equations can be found in [11]. Now equation (11) can be used to calculate the quiescent leakage for any transistor in the stack, which is the leakage current of the PDN.

Considering the fact that standby leakage current depends on input signal levels, the average leakage power of a circuit can be evaluated with random patterns applied to primary inputs.

4 Algorithm

Due to the exponential relationship between threshold voltage and drain current in the weak inversion region, a higher threshold voltage will significantly reduce leakage current, thereby reducing leakage power. However, Figure 3 indicates that a higher threshold voltage will increase the equivalent on-resistance of each transistor, which results in a higher propagation delay. Normally, threshold voltage is empirically defined to be around 20% of supply voltage to maintain the performance of a circuit [9]. For low supply voltage circuits, the threshold voltage could be very small, leading to a large leakage current.

This problem can be circumvented by using dual threshold voltages. A low threshold voltage is assigned to the transistors in critical path(s) in order to achieve high performance, while a high threshold may be assigned to some transistors in non-critical paths to reduce leakage power. The lower bound of low threshold voltage is determined by noise margin. The possible high threshold value should be in the range from low threshold to $0.5V_{dd}$. However, not all the transistors in non-critical paths can be assigned the high threshold voltage. Otherwise, some non-critical paths may become critical. Whether a node can be modified depends on the value of the high threshold. If it is too small, there is little difference of propagation delay between low-Vth and high-Vth transistors. Hence, more nodes can be assigned high-Vth without influencing the critical delay, but the leakage current improvement for each high-Vth transistor would be small. On the other hand, if the high threshold voltage is too large, the leakage current can be reduced by a large amount for each such transistor. However, fewer nodes can be modified. Hence, among the allowable values for high threshold voltage, there exist an optimal one. We developed a breadth-first search(BFS) algorithm to search for the optimal high-Vth.

The first step in our algorithm is to initialize a circuit with a single low threshold. After initialization, all necessary parameters associated with each node ($t_p(x)$, $T_{max}(x)$, and $T_l(x)$) are computed. By checking all the primary outputs and then backtracing, the critical delay and critical path(s) can be identified using a first-in-first-out (FIFO) queue Q. The pseudo-code for the initialization procedure is shown below. Note that, primary output (PO) does not map to a gate in a circuit, and each PO has only one fan-in gate ($fanin(PO)$).

```
Initialization () {
  Assign a level number to each node
  Calculate the propagation delay $t_p(x)$ of each node $x$
  Calculate $T_{max}(x)$ and $T_l(x)$ of each node $x$ level by level
  Identify $T_{critical}$ by checking the maximum $T_l(fanin(PO))$
  For each primary output $PO$ {
    If ($T_l(fanin(PO)) = T_{critical}$)
      Mark $fanin(PO)$ as a node in critical path
      Add node $fanin(PO)$ into a FIFO queue $Q$
    }
  While ($Q$ not empty) {
    Remove node $x$ from $Q$
    For each fan-in $y$ of node $x$ {
      If (($T_l(y) = T_{max}(x)$) && ($y$ is not a primary input))
        Mark $x$ as a node in critical path
        Add node $y$ into queue $Q$
    }
  }
}
```

During the initialization procedure, in order to obtain $T_{max}(x)$ and $T_l(x)$, the circuit has to be levelized. Essentially, levelization assigns a number to each node to indicate the depth of the node in the graph. The level of each primary input is defined to be 0. The level of any node x, denoted as $l(x)$, can be calculated as follows,

$$l(x) = 1 + \max_{j \in all\ fanins} \{l(j)\} \quad (17)$$

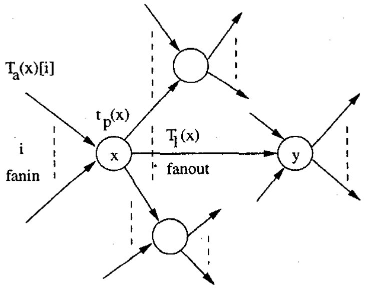

Figure 4: Diagram of a part of a logic circuit

where j varies for all fan-in nodes of node x.

For each primary input x, $t_p(x) = 0$, $T_a(x) = 0$, $T_l(x) = 0$ and $T_{max}(x) = 0$. For each node x in level 1, $T_a(x) = 0$, $T_{max}(x) = 0$ and $T_l = t_p(x)$. Therefore, the parameters $(t_p(x), T_a(x), T_l(x)$ and $T_{max}(x))$ associated with each node x can be computed by equations (2) and (3) level by level during the initialization procedure.

The next step is to assign a high threshold to some transistors on non-critical paths under performance constraints. This is performed by checking the slack of each node using a BFS-based backtracing algorithm. Slack of a node $(T_\delta(x))$ denotes the amount by which the gate can be slowed down. For the nodes in critical path(s), slack is 0. For a PO,

$$T_\delta(PO) = T_{critical} - T_l(f_{anin}(PO)) \tag{18}$$

For any other node x (suppose x is traversed from node y during back-tracing (Figure 4)), $T_\delta(x)$ can be expressed as

$$T_\delta(x) = \min\{(T_\delta(y)+T_{max}(y)-T_l(x)), \min(T_{max}(z)-T_l(x))\}$$

$$\forall z = fanout(x) \neq y \tag{19}$$

where $f_{anin}(x)$ and $f_{anout}(x)$ are the fan-in nodes and fan-out nodes of node x, respectively. Consider Figure 4. The first term in equation (19) ensures that the propagation delay of the path(s): $\ldots \rightarrow fanin(x) \rightarrow x \rightarrow y \rightarrow \ldots$ is no greater than the critical delay. The second term guarantees that the modification of the propagation delay of node x cannot affect the propagation delay of all the other fanout paths of node x. To make sure that the critical delay is not affected, $T_\delta(x)$ is taken as the minimum value of the two terms as shown in equation (19).

The procedure for choosing the nodes with a high threshold voltage works as follows. From each PO, BFS is used to explore every node on the breadth-first tree of $G(V, E)$. If a node has not been visited yet, by checking its slack, we can decide whether its threshold voltage should be changed. Once the node is visited, it is marked to avoid repeating assignment. By definition, for each node in a single threshold circuit, its slack(T_δ) is no less than 0. Increasing the threshold voltage of a node can result in a higher propagation delay and departure time of this node. Therefore, the slack will decrease. Whether a node should be assigned to a high threshold depends on whether its slack is still positive if its threshold is changed to high threshold. If slack is still positive, this node will be assigned to the high threshold and the number of high threshold NMOS-PMOS pairs (N_c) is incremented. Since the slack of each node in critical path is 0, the threshold voltage of these transistors will not be changed, and hence, the performance is maintained. The pseudo-code for the above procedure is shown below:

```
High-Vth-Assignment($Vth_2$){
  For each primary output $PO$ {
    Explore each node $x$ using breadth-first search
      If $x$ has not been visited
        Calculate $t_p(x)$, $T_l(x)$ and $T_\delta(x)$ for high threshold $Vth_2$
        if $T_\delta(x) \geq 0$
          Assign $Vth_2$ to $x$
          Assign $t_p(x)$, $T_l(x)$ and $T_\delta(x)$ for $Vth_2$ to $x$
          $N_c$++
        else
          Keep $t_p(x)$, $T_l(x)$ and $T_\delta(x)$ for initial low Vth for $x$
        Mark $x$ visited
    }
}
```

Finally, we will search the optimal high threshold voltage (opt_Vth_2) corresponding to the best saving of standby leakage power. The high thresholds are sampled according to the different on-resistances (R). $R(0)$ is the on-resistance for original low Vth. Step size ΔR of $0.1R(0)$ is chosen for the simulation. The relationship between on-resistance and threshold voltage is given by equation (8). Standby leakage power can be evaluated using the method described in section 3. After updating the network for opt_Vth_2, the circuit can be transfered into SPICE netlist and simulated using HSPICE to verify the results. The procedure is outlined below:

```
Optimal-High-Vth (){
  i=1 and $R(i) = R(0) + \Delta R$
  Calculate $Vth_i$ corresponding to $R(i)$
  While ($Vth_i < 0.5V_{dd}$){
      Initialization
      High-Vth-Assignment($Vth_i$)
      Estimate standby leakage power $P_{stdby}$
      If standby leakage power is the least power so far
          $P_{stdby_{min}} = P_{stdby}$
          $opt\_Vth_2 = Vth_i$
      ++i and $R_i = R_0 + i \times \Delta R$
      Calculate $Vth_i$ corresponding to $R_i$
      }
  Update network with $opt\_Vth_2$
  Transfer the network into SPICE netlist
}
```

The above algorithm can be easily extended to solve the other problems, such as multiple supply voltage design & optimization [13].

5 Implementation and Results

The method to reduce leakage power using dual-threshold-voltage transistors has been implemented in C under the Berkeley SIS environment. In order to simplify the analysis, technology-mapping was used to map the circuits to a library which contains NAND gates, NOR gates and Inverters. All the simulation results were based on a $0.5\mu m$ MOSIS process. The effective channel length was $0.32\mu m$ and the gate oxide thickness was 9.8nm. The effective channel widths for PMOSFETs and NMOSFETs were assumed to be $10.5\mu m$ and $3\mu m$, respectively.

Figure 5 gives an example circuit to show how our algorithm works. Figure 5(a) is the original single-Vth circuit, where the supply voltage is 1V and the threshold voltage is 0.2V. Figure 5(b), (c),and (d) show the dual-Vth circuits with the low Vth of $0.2V$ and the high-Vth of 0.25V, 0.396V,and 0.46V, respectively. Note that the critical paths and critical delay are maintained after the assignment.

Figure 6 shows the standby leakage power of the above example circuit with different high thresholds(Vth_2). The supply voltage is 1V, the low threshold voltage is 0.2V and the circuit temperature is 25°C. Vth_2 varies from 0.2V to

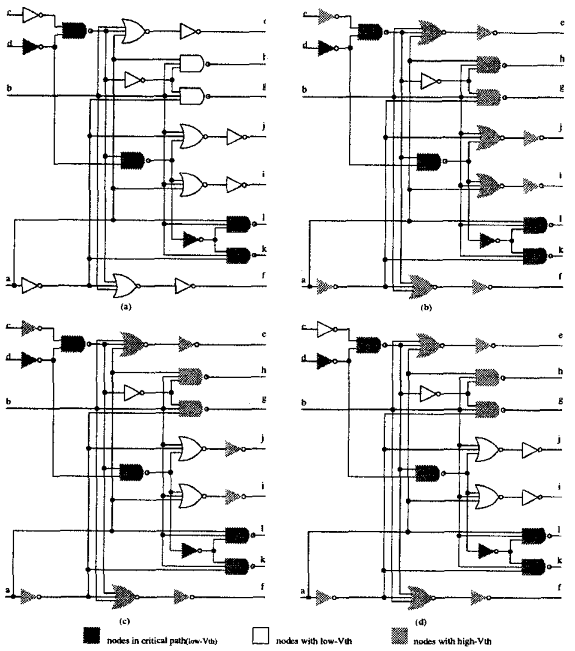

Figure 5: An example circuit (a) 1-Vth $V_{dd} = 1V$, $Vth = 0.2V$ (b)-(d) 2-Vth $V_{dd} = 1V$, $V_{th1} = 0.2V$ (b) $Vth_2 = 0.25V$ (c) $Vth_2 = 0.396V$ (d) $Vth_2 = 0.46V$

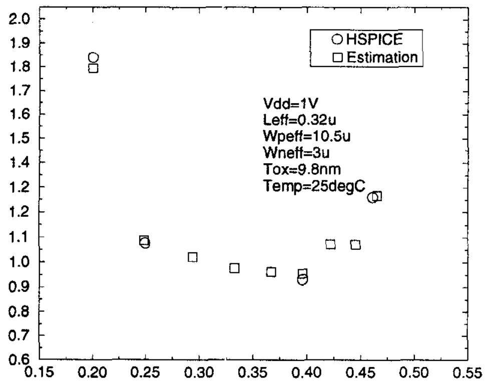

Figure 6: Standby Leakage power with different Vth_2

0.5V ($Vth_2 = 0.2V$ represents the single low threshold circuit). The squares represent the leakage power obtained by our estimation technique while the circles denote the leakage power obtained by HSPICE simulations. Clearly, the estimation results fit well with HSPICE simulation results. The convex point of the curve indicate that there exits an optimal high threshold voltage($0.396V$) which leads to a 50.67% savings in standby leakage power.

Table 1: Optimal High-Vth and Standby Leakage Power Saving (V_{dd}=1V, Temp=25°C, Single-Vth:$V_{th} = 0.2V_{dd}$, Dual-Vth: $Vth_1 = 0.2Vdd$, Vth_2=opt-Vth_2)

Circuit Chosen	Gate #	$P_{stdby}(\mu W)$ (1-Vth)	Vth_2 (mV)	$P_{stdby}(\mu W)$ (2-Vth)	Red. (%)
C432	206	27.97	333	11.60	58.51
C499	532	67.17	367	45.04	32.95
C880	353	46.29	333	17.64	61.90
C1355	517	64.86	333	37.14	42.73
C1908	615	77.68	333	37.58	51.63
C2670	807	106.96	333	54.98	48.60
C3540	1131	151.18	367	60.09	60.25
C5315	1778	228.58	367	102.69	55.08
C6288	2400	355.09	367	277.14	21.95
C7552	2803	378.95	367	201.01	46.96

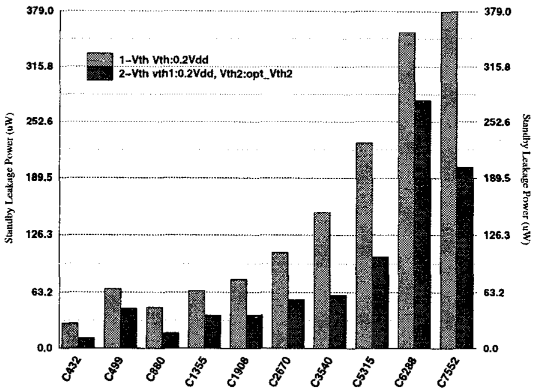

Figure 7: Standby leakage power savings for ISCAS benchmarks ($V_{dd} = 1V$)

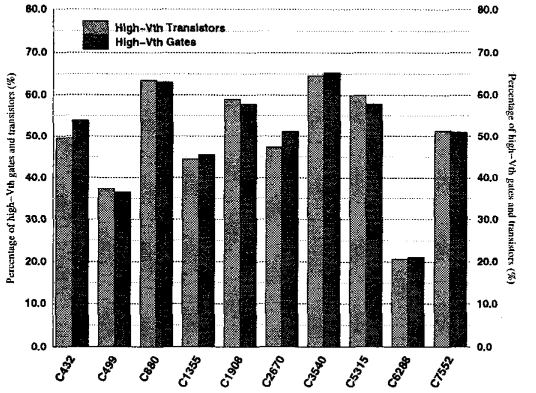

Figure 8: Percentage of high threshold gates and transistors for dual-Vth ISCAS Benchmarks($V_{dd} = 1V$)

Table 1 and Figure 7 show the optimal high threshold and standby leakage power savings for ISCAS benchmark circuits. The percentages of high-Vth transistors and gates over total transistors and gates for different dual-Vth benchmark circuits are illustrated in Figure 8. In this experiment, supply voltage was 1V and the circuit temperature was 25^oC. The low threshold voltage was assumed to be

0.2V and high threshold voltage was the optimal value obtained from our heuristic algorithm given in section 4. Results indicate that the percentage of high threshold voltage transistors can be more than 60% and standby leakage power can be reduced by around 50% for most of the circuits. Even though the optimal high threshold voltage varies for different circuits, for most of the circuits, it was between $0.3V_{dd}$ and $0.4V_{dd}$.

For a CMOS digital circuit, total power dissipation includes dynamic and static components in active mode. Ignoring power dissipation due to the short circuit current, total active power dissipation can be expressed as follows [10],

$$\begin{aligned} P_T &= P_{dyn} + P_{static} \\ &= \sum_i \alpha_i \, C_i \, V_i \, V_{dd} \, f_{clk} + I_{static} V_{dd} \qquad (20) \end{aligned}$$

where α_i is the switching activity (the probability of switching), C_i is the load and parasitic capacitances, f_{clk} is the operating frequency and V_i is the voltage swing which equals to V_{dd} at the output node and $V_{dd} - V_{th}$ at internal nodes. The summation is taken over all nodes of the circuit. I_{static} is the leakage current through the circuit.

Consider the example circuit(Figure 5). Figure 9 shows the HSPICE simulation results of the total active power dissipations of single-Vth and dual-Vth circuits at different frequencies. The circuits were simulated at 1V supply voltage and 110°C. The threshold voltage of single-Vth circuit was 0.2V. The low and high threshold voltages of dual-Vth circuit were 0.2V and 0.396V, respectively. At low frequency, the active power saving of dual-Vth circuit, which is mainly because of the static power reduction, is about 50%. As for high frequency circuits, the active power dissipation is dominated by dynamic consumption. In addition to leakage power saving, the dynamic power is reduced due to the reduction of internal node voltage swing for high threshold gates. In our example, the total power saving can be around 13% at 100MHz frequency.

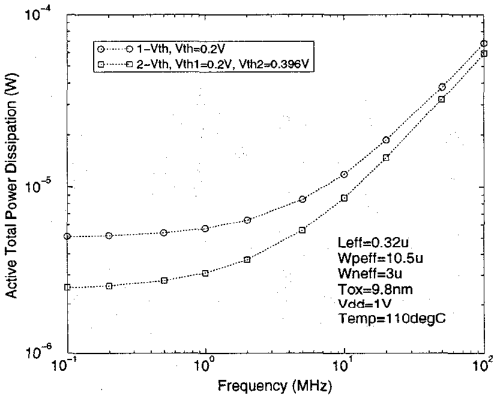

Figure 9: Active power dissipation at different frequencies

6 Conclusions

In this paper we present a method to design and optimize low voltage dual-Vth CMOS circuits. In order to reduce leakage power under performance constraints, starting with a single low Vth circuit, a heuristic algorithm for selecting and assigning an optimal high threshold voltage is proposed. For accurate leakage power estimation, a standby leakage current model which has been verified by HSPICE simulation is used. Results for ISCAS benchmark circuits show that the leakage power can be reduced by more than 50% under performance constraints. The optimal high threshold voltages are between $0.3V_{dd}$ and $0.4V_{dd}$, given that the low threshold voltage is $0.2V_{dd}$. The total active power dissipation can also be reduced using dual-Vth design technique. The total power saving can be about 13% for some circuit at 100MHz frequency.

References

[1] J. D. Meindl, "Low power Microelectronics: Retrospect and Prospect", *Proceedings of the IEEE,* Vol.83, No.4, pp.619, 1995.

[2] A. P.Chandrakasan, et al.,"Low-Power CMOS Digital Design", *IEEE Journal of Solid-State Circuits,* Vol.27, No.4, pp.473, 1992.

[3] P. Pant, V. De and A. Chstterjee, "Device-Circuit Optimization for Minimal Energy and Power Consumption in CMOS Random Logic Network", *DAC,* pp. 25.1.1-25.1.6, 1997.

[4] C. Hu, "Device and technology impact on low power electronics", in *Low Power Design Methodologies,* J.M.Rabaey and M.Pedram, Eds. Norwell, MA:Kluwer,pp21-36, 1996.

[5] B. Davari, et al., "CMOS Scaling for High Performance and Low Power-The Next Ten Years", *Proceedings of the IEEE,* Vol.83, No.4, pp595, 1995.

[6] Z. Chen, et al., "0.18um Dual Vt MOSFET Process and Energy-Delay Measurement", *IEDM Digest,* pp851, 1996.

[7] Jan M. Rabaey, "Digital Integrated Circuits", New Jersey: Prentice-Hall, 1996.

[8] S. Mutoh, et al., "1-V Power Supply High-Speed Digital Circuit Technology with Multithreshold-Voltage CMOS", *IEEE Journal of Solid-State Circuits,* Vol.30, No.8, pp. 847-853, 1993.

[9] H. Oyamatsu, et al., "Design Methodology of Deep Submicron CMOS Devices for 1V Operation",*IEICE Trans. Electron.,* VOL.E79-C, No.12, pp1720-1724, 1996

[10] A. Bellaouar and M. I. Elmasry, "Low-Power Digital VLSI Design", Kluwer Academic Publishers, 1995

[11] M.C. Johnson, K. Roy, and D. Somasekhar "A model for leakage control by transistor stacking", Technical Report TR-ECE 97-12, Purdue University, Dept. of ECE.

[12] J. Sheu, et. al., "BSIM: Berkeley Short-Channel IGFET Model for MOS Transistors", *IEEE J. Solid-State Circuits,* SC-22, 1987.

[13] M.C. Johnson and K. Roy, "Datapath scheduling with multiple supply voltages and level converters", ACM Transactions on Design Automation of Electronic Systems, July 1997.