Low-power LSI
- Through cooperation among levels -

Takayasu Sakurai
Center for Collaborative Research, and Institute of Industrial Science, University of Tokyo
E-mail: tsakurai@iis.u-tokyo.ac.jp

Abstract
In this paper, methods to achieve low-power and high-speed VLSI's are described with the emphasis on cooperation between levels. To suppress the leakage current in a standby mode, Boosted Gate MOS (BGMOS) is effective, which is based on cooperation between technology level and circuit level. To reduce the power in an active mode, VDD-hopping and VTH-hopping are promising, which are cooperative approaches between circuit and software. Power consumed in interconnect system is another issue in low-voltage deep-submicron designs. A cooperative approach between VLSI and assembly to the interconnect power problem is also discussed.

1. Introduction
Power consumption of VLSI's is ever increasing as is shown in Fig.1 and could be a stumbling block in realizing giga-scale integration. The power consumption of CMOS VLSI's consists of a dynamic charging and discharging current component and a leakage current component shown in Fig.2. Various effective techniques to mitigate the power problem have been proposed at a level of system, algorithm, software, CAD, circuit, technology and assembly. There still remains effective ways to reduce the power consumption through cooperation between levels, which have not been pursued so rigorously because usually it is not easy to establish a cooperative mechanism among different levels of engineers. The treasured sword of CMOS scaling, however, may come to an end due to the power problem and every endeavor to decrease power consumption will be required in the future.

The supply voltage, VDD, is ever decreasing to ensure sufficient reliability of VLSI's with thin gate oxide used in deep submicron transistors. Low-power design in the low- VDD environments is a battle against the ever-increasing leakage current due to the use of low threshold voltage, VTH, to realize high-speed VLSI's. Since low-VDD and high-VTH are advantageous for low power consumption but disadvantageous for speed, there should be some controlling scheme among VDD, VTH and speed. Most of the controlling schemes can be categorized in a tabulated form as in Fig.3. In the table, multiple VDD and multiple VTH signify spatial assignment of VDD and VTH, while variable VDD and variable VTH correspond to temporal assignment of VDD and VTH.

Historically well-known techniques to control VDD and VTH are MTCMOS (Multi-Threshold CMOS) and VTCMOS (Variable Threshold CMOS) [1-3] but these schemes will not be able to stand in the future as shown in Fig.4. In this paper, Boosted Gate MOS (BGMOS) is described to cope with the standby current increase due to the use of low VTH, which is a cooperative approach between a technology level and a circuit level. As for the power reduction in an active mode, VDD-hopping and VTH-hopping are introduced, which are cooperative approaches between circuit and software. Lastly, the paper touches on an interconnect power problem by buffer insertion.

2. Cooperation between technology and circuit: BGMOS
In order to mitigate the leakage problem in a standby mode, it is effective to insert a non-leaking power switch in series to a leaky yet high-speed logic gate block made of low-VTH MOSFET's. The basic idea is the same as MTCMOS but MTCMOS becomes slow when VDD gets less than 1V and stops operating when VDD gets less than 0.5V. This is because the non-leaking power switch can be realized by a high-VTH (0.6V for example) MOSFET that is turned on in an active mode and turned off in a standby mode.

The problem of MTCMOS is solved if higher voltage around 1.5V-2.0V is applied to the gate of the
power switch MOSFET. The higher voltage achieves higher drivability and hence higher speed. This scheme is called boosted gate MOS scheme (BGMOS) as shown in Fig.6 [4]. The power switch should have higher oxide thickness to accommodate the higher voltage on the gate. A technology side provides a thicker oxide transistor, while designers think about using the different type of transistors and thus the scheme can be said cooperation between a technology level and a circuit level. MOSFET’s tuned for the higher voltage is also helpful in SRAM, I/O and analog designs as shown in Fig.8. The thicker oxide is also beneficial to decrease leakage current caused by direct tunneling through oxide of the power switch (see Fig.5).

It should be noted that there is an optimum thickness for the oxide of the power switch and thus there is optimal gate voltage which is 1.5V-2.0V as shown in Fig.7. This is due to the increase of channel length to suppress a short channel effect when the oxide gets thicker.

3. Cooperation between circuit and software: VDD hopping and VTH hopping

In an active mode, changing VDD and VTH in time in accordance with required performance at every moment is effective for power reduction. If VDD is lowered or VTH is increased, the power decreases but speed is degraded. The difficulty is to find the timing to lower the speed. Hardware cannot know the timing when decreasing the performance of a processor does not affect the system performance. Only software knows when it is possible to decrease the processor performance without sacrificing the system performance. Software should tell hardware when higher performance is needed.

The circuit side provides a processor whose operating frequency and VDD can be varied by software. The software side controls the frequency and VDD adaptively so that the frequency is lowered to a half when high-speed operation is not needed (see Fig.9). The scheme has been applied to a MPEG4 codec system and the processor power has been reduced to one fourth of the conventional fixed VDD processor [5, 6] (see Fig.11-15). The video codec system guarantees real-time operation for any data input but the highest performance is needed only for 6% of time (see Fig.14). The algorithm to adaptively change VDD depending on the workload is of importance. Since the workload depends strongly on data, the control should be dynamic in run-time, and should not be static in a compile-time. It is too late to notice that the past task is an easy task which can be done by using the lower voltage because once the task is completed by using the high voltage, energy has been already consumed. On the other hand, it is impossible to predict the workload of the task to be done in the future without error. To solve this problem, the algorithm introduces an application slicing and a software feedback loop. By chopping an application into slices, executing the first slices at the maximum frequency, and checking the current time and the time margin to execute the next slice, the optimum clock frequency and VDD adaptively selected by a software feedback loop (see Fig.10)

It is to be noted that VDD hopping algorithm works fine for every multimedia application we tried although the switching time between voltage levels requires 0.2ms which is considered to be extraordinary long in terms of processor clock period. In a multimedia application, however, the real-time feature is for humans and human is slow. This is the reason why the VDD hopping works fine in spite of the long transition time between voltage levels. The other point of interest is that the number of voltage levels can be as low as two as is shown in Fig.11.

The VDD hopping scheme can also be applied to multi-tasking real-time operating system [7] (see Fig.16). Since OS knows higher-level information on available time slot assignable to an application, higher efficiency can be realized than application-only case as shown in Fig.16. One example we tried is modified power-conscious µ-ITRON OS running FFT and MPEG4 at the same time and the observed power reduction was 75% while the power saving for FFT alone was only 50%.

When subthreshold leakage becomes dominant in the future, the same software control mechanism can be used in VTH hopping scheme where VTH is changed in time in accordance with the required performance [8] (see Fig.17-19). About 80% power reduction is possible for a multimedia real-time application. It is found that time-domain assignment of VTH is more effective than spatial assignment of VTH for multimedia video application as shown in Fig.20.
4. Cooperation between VLSI and assembly: Superconnect

In deep submicron designs, RC delay of interconnect hinders a high-speed operation of VLSI's. A widely used remedy for the interconnect RC delay is a buffer insertion technique. The basic optimization theory is summarized in Fig.21. The drawback of the buffer insertion, however, is that the delay optimization by buffer insertion increases power consumption by 73% as shown in Fig.22. With the use of thicker layer of metals provided by superconnect [9,10], it is possible to decrease interconnect delay without increasing power consumption because buffer insertion is not needed with low resistance interconnects. Co-design between an LSI itself and an assembly structure such as an interposer and a package will be needed. High current expected in low-VDD regime shown in Fig.1 can also be mitigated by the use of the thicker metal layer in an assembly body and small area pads on an LSI.

5. Summary

Power consumption of LSI's tend to increase due to the scaling law and due to the leakage increase including sub-threshold, gate tunneling, and junction leakage.

New trend for low-power LSI's is to pursue cooperative approaches among levels: BGMOS to cut-off standby leakage, $V_{DD}$ / $V_{TH}$ hopping to reduce operating power, and super-connect to reduce I/O power are some of the examples.

One of the biggest barriers to the scaling is the leakage power increase and solutions are yet to be discovered, though several good trials have been made.

In deep submicron designs, RC delay of interconnect

References


Fig.1 $V_{DD}$, Power and Current Trend

Fig.2 Power & Delay Dependence on $V_{DD}$ & $V_{TH}$
Low power → Low $V_{DD}$ → Low speed → Low $V_{TH}$ → High leakage → $V_{DD}, V_{TH}$ control

<table>
<thead>
<tr>
<th>Multiple $V_{TH}$</th>
<th>Dual-$V_{TH}$</th>
<th>MTCMOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable $V_{TH}$</td>
<td>$V_{TH}$ hopping</td>
<td>VT-CMOS</td>
</tr>
<tr>
<td>Multiple $V_{DD}$</td>
<td>Dual-$V_{DD}$</td>
<td>Boosted gate MOS</td>
</tr>
<tr>
<td>Variable $V_{DD}$</td>
<td>$V_{DD}$ hopping</td>
<td></td>
</tr>
</tbody>
</table>

Technology-circuit cooperation

1) MTCMOS: Multi-Threshold CMOS
2) VT-CMOS: Variable Threshold CMOS
- Multiple: spatial assignment
- Variable: temporal assignment

Fig.3 Controlling $V_{DD}$ and $V_{TH}$ for low power

MT-CMOS [1]
- Tunneling leakage can not be cut-off.
- Area penalty increases when $V_{DD} < 1V$.

VT-CMOS [2]
- Junction leakage increases due to band-to-band tunneling.
- Tunneling leakage is not suppressed.


Fig.4 Previous circuit schemes

Fig.5 Transistors go leaky

$V_{DD}$

CMOS circuits
- low $V_{TH}$
- ultra thin $T_{OX}$

Virtual $V_{SS}$

Leak cut-off Switch (LS)
- high $V_{TH}$
- thick $T_{OX}$

Fig.6 Boosted gate MOS scheme

Fig.7 Power switch optimization

Fig.8 GSI’s in deep-submicron era

Energy consumption is proportional to the square of $V_{DD}$.

$V_{DD}$ should be lowered to the minimum level which ensures the real-time operation.

Fig.9 If you don’t need to hassle, $V_{DD}$ should be as low as possible

Fig.10 Application slicing and software feedback loop in $V_{DD}$ hopping
**Fig. 11** Run-time VDD hopping reduces power to less than 1/10

**Fig. 12** Block diagram

**Fig. 13** Measured voltage waveforms

**Fig. 14** Measured power characteristics

**Fig. 15** Simulation results

**Fig. 16** Power Conscious OS & Application Slicing

**Fig. 17** VTH-hopping

**Fig. 18** Schematic of VTH-hopping
0.6µm process
Overhead of VTH-hopping ≈ 14%
RISC core = 2.1mm x 2.0mm
VBS selector = 0.2mm x 0.6mm

Fig.19  Microphotograph of RISC processor

0.6
µm process

Fig.20  Power comparison


Fig.21  Buffer insertion


Fig.22  Power delay optimization

Fig.23  RC delay of global interconnections

Fig.24  3D Integration

Fig.25  3D Integration

Fig.26  Superconnect

Fig.27  Superconnect technology