Minimizing Power across Multiple Technology and Design Levels

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

Abstract

Approaches to achieve low-power and high-speed VLSI’s are described with the emphasis on techniques across multiple technology and design 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, $V_{DD}$-hopping and $V_{TH}$-hopping are promising, which are cooperative approaches between circuit and software. Power consumed in interconnect system can be reduced by a cooperative approach between application and layout as in bus shuffling. Other low-power design approaches are also discussed.

1. Introduction

Power consumption of VLSI’s is ever increasing and various effective techniques to mitigate the power problem have been proposed at a level of system, algorithm, software, CAD, circuit, technology and assembly. There is, however, a new trend in low-power designs to exploit cooperation across multiple technology and design levels. To support the new paradigm, EDA tools are required. In Fig.1, some of the cooperative schemes are tabulated.

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-$V_{TH}$ MOSFET’s (BGMOS in Fig.2, [1]). The basic idea is the same as MTCMOS [19] but MTCMOS becomes slow when $V_{DD}$ gets less than 1V and stops operating when $V_{DD}$ gets less than 0.5V. In BGMOS, the non-leaking power switch is realized by a high-$V_{TH}$ (0.6V for example) MOSFET but the gate of the switch is driven up to higher voltage than $V_{DD}$ to ensure high drivability. The gate oxide thickness of the power switch should be thicker than normal transistors to cope with the higher gate voltage.

To realize the scheme, the technology side provides a thicker oxide transistor, while the design side thinks about using the different type of transistors and thus the scheme can be called cooperation between a technology level and a circuit level. Design tools are needed to handle various types of transistors for low power. MOSFET’s tuned for the higher voltage is also helpful in SRAM, I/O and analog designs as shown in Fig.2.

3. Cooperation between circuit and software: $V_{DD}$ hopping and $V_{TH}$ hopping

In an active mode, changing $V_{DD}$ and $V_{TH}$ in time in accordance with required performance is effective for power reduction. If $V_{DD}$ is lowered or $V_{TH}$ is increased, the power decreases but speed is degraded as shown in Fig.3. The difficulty is to find the timing to lower the speed. Only software knows when it is possible to decrease the processor performance without sacrificing the system performance. Hardware provides a method to change $V_{DD}$ and/or $V_{TH}$. The scheme is shown to be effective even for real-time multimedia applications.

In $V_{DD}$-hopping, $V_{DD}$ is changed according to software’s decision [14, 6] (see Fig.4). The scheme has been applied to a MPEG4 codec system and the power of the processor has been reduced to one fourth of the conventional fixed $V_{DD}$ processor in the measurement. The video codec system guarantees real-time operation for any data input but the highest performance is needed only for 6% of time.

The algorithm to adaptively change $V_{DD}$ 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. By chopping a real-time task into slices, and by monitoring current time and deadline for a slice, we can successfully control $V_{DD}$ to reduce power. There is a software feedback loop.

It is to be noted that $V_{DD}$ hopping algorithm works fine for every multimedia application we tried including MPEG2 and VSELP although the switching time between voltage levels requires 0.2ms which is considered to be extraordinary long in terms of processor clock period. This long transition time is due to the charging and discharging of huge capacitance on $V_{DD}$ nodes on the board and in the LSI. In a multimedia application, however, the real-time feature is for humans and human is slow. This is the reason why the $V_{DD}$ 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.5.

The $V_{DD}$ hopping scheme can also be applied to multitasking real-time operating system [20] (see Fig.6). Since OS has higher-level information on available time slot that can be assigned to an application, higher efficiency can be realized compared with 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 as shown in Fig.7, the same software control mechanism can be used in $V_{TH}$ hopping scheme where $V_{TH}$ is changed in time in accordance with the required performance [5] (see Figs.8-9). About 80% power reduction is possible for a multimedia real-time application.

4. Cooperation between application and layout: Bus shuffling

Power consumed in interconnects is another issue. Recent interconnect consumes power by the coupling capacitance. Bus shuffling which reduces the power consumed by the coupling capacitance is an approach for low-power through cooperation between application level and layout. Bus layout is just shuffled without any encoder and decoder but the scheme achieves about 40% power reduction [2] (see Figs.10-11).

5. Other approaches

Another important low-power consideration is on I/O's. 3D integration using System in Package (SiP) will be effective in reducing the I/O power. In designing SiP, co-design between an LSI itself and an assembly structure will be needed. Voltage drop across power lines due to high current expected in low-$V_{DD}$ regime can also be mitigated by the use of the thicker metal layer on an interposer and area pads of an LSI (see Fig. 12-14). Design tools for the SiP are to be investigated.

In the future, when device is scaled further, power consumption of LSI's tend to increase due to the leakage increase including sub-threshold, gate tunneling, and junction leakage. One important way to mitigate active leakage problem is to adopt memory-rich architectures [7] (see Figs.15).

Some of the approaches for low-power LSI's in the active leakage dominant regime are summarized in Fig.16.

References


Fig. 1 Controlling \( V_{DD} \) and \( V_{TH} \) for low power

Fig. 2 Boosted Gate MOS (BG MOS) and low-voltage VLSI

Fig. 3 Power & Delay Dependence on \( V_{DD} \) & \( V_{TH} \)

Fig. 5 \( V_{DD} \) hopping reduces power in multimedia applications

Fig. 6 Power Conscious OS & Application Slicing

Fig. 7 Increasing leakage power
**Fig. 8** Schematic of VTH-hopping

**Fig. 9** Power comparison of VTH-hopping and others

- **On-chip bus coding**
  - Area, delay and power overhead by extra circuit

- **Bus shuffling**
  - Just shuffling and no overhead
  - Considering coupling among lines
  - Statistical characteristics of signal is required.

**Fig. 10** Bus shuffling concept

- **Result of heuristic**
  - 7 data address sets
  - 40% power saving compared to un-shuffled buses

**Fig. 11** Power reduction by bus shuffling

**Fig. 12** System in Package

**Fig. 13** 3D Integration

**Fig. 14** Reduction of I/O power by superconnect

**Fig. 15** Demand for memory-rich architecture

<table>
<thead>
<tr>
<th>Year</th>
<th>N_CHIP</th>
<th>N_LOGIC</th>
<th>N_MEMORY</th>
</tr>
</thead>
<tbody>
<tr>
<td>2011</td>
<td></td>
<td></td>
<td>3%</td>
</tr>
<tr>
<td>2012</td>
<td>15%</td>
<td></td>
<td>85%</td>
</tr>
<tr>
<td>2002</td>
<td></td>
<td></td>
<td>97%</td>
</tr>
</tbody>
</table>

**Fig. 16** Detailed table for low-power techniques

<table>
<thead>
<tr>
<th>Variable Vth</th>
<th>Active Dynamic</th>
<th>Active Leakage</th>
<th>Standby</th>
<th>Memory</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dual-VTH(15)</td>
<td>LBD(3)</td>
<td>MTCMOS(19)</td>
<td>Dual-VTH(12)</td>
<td></td>
</tr>
<tr>
<td>VTH hopping(5)</td>
<td>VTCMOS(16)</td>
<td>DLC(8)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dual-VDD(17)</td>
<td>BGmos(1)</td>
<td>Dual-V(6)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VDD hopping(14)</td>
<td>VDD hopping(14)</td>
<td>RRDV(9)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vos Rev. Bias</td>
<td>LBSF(4)</td>
<td>SCCMOS(13)</td>
<td>SSICMOS(16)</td>
<td></td>
</tr>
</tbody>
</table>

- **Bus Shuffling**
- LBD: Leakage-Biased Domino Circuits
- MTCMOS: Multi-Threshold CMOS
- VTCMOS: Variable Threshold CMOS
- BGmos: Boosted Gate MOS
- SCCMOS: Super-Cut-Off CMOS
- RRDV: Row by Row Dynamic Voltage Control
- SSICMOS: Switched-Source-Impedance CMOS