

A Power-Constrained MPU Roadmap for the International Technology Roadmap for Semiconductors (ITRS)

Kwangok Jeong[†] and Andrew B. Kahng^{†‡}

[†]CSE and [‡]ECE Departments, UC San Diego, La Jolla, CA
kjeong@vlsicad.ucsd.edu, abk@cs.ucsd.edu

Abstract—Technology roadmaps help predict requirements for future technologies and guide ongoing technology research and development. The International Technology Roadmap for Semiconductors (ITRS) [1] has been published since 1999; it is fully revised in odd-numbered years and updated in even-numbered years. In this paper, we describe a modeling framework to predict future characteristics and implied technology requirements of the ITRS microprocessor (MPU) ‘system driver’, following Moore’s Law scaling under power constraints. The outcomes of this modeling effort are the basis of ITRS updates to be published in the 2009 ITRS edition.

I. INTRODUCTION

Semiconductor integrated-circuit products have seen rapid improvements in terms of integration level, cost, speed, power, etc. for more than four decades. These improvements are enabled by R&D investments whose growing size has motivated a number of industry R&D partnerships, consortia, and other cooperative ventures. To help guide and align these R&D programs, the Semiconductor Industry Association (SIA) in 1992, 1994 and 1997 published the National Technology Roadmap for Semiconductors (NTRS). In 1998, the SIA was joined by corresponding industry associations in Europe, Japan, Korea, and Taiwan to participate in a 1998 update of the Roadmap and to begin work toward the first International Technology Roadmap for Semiconductors (ITRS). The ITRS delivers an industry-wide, global consensus on the *best current estimate* of research and development needs out to a 15-year horizon. As such, the ITRS provides a guide to the efforts of companies, universities, governments, and other research providers or funders.

With respect to semiconductor industry evolution, the microprocessor (MPU) is regarded as a key ‘system driver’, since MPU products are often at the leading edge of process and device technology, integration level, design methodology, clock frequency, power consumption and other metrics. According to the ITRS [1], the MPU model reflects general-purpose instruction-set architectures (ISAs) that are found standalone in desktop and server systems, and embedded as cores in SOC applications.

As new technologies and architectures, along with increased levels of integration, have improved MPU performance, high power consumption has become an increasingly important topic of public discussion in recent years because of global CO₂ emissions. Power dissipation limits of packaging (despite being estimated to reach 200 W/cm² by the end of the 2007 ITRS timeframe) and the cost of cooling make high supply voltages and frequencies infeasible. Past clock frequency trends for the MPU system driver have been interpreted as future CMOS device performance (switching speed) requirements (e.g., a historic 17%/year improvement in the high-performance bulk CMOS device CV/I metric) that lead to large off-currents and extremely thin gate oxides. Given such devices, MPUs that simply continue existing circuit and architecture techniques would exceed package power limits by factors of nearly 4× by the end of 2020; alternatively, MPU logic content and/or logic activity would need to decrease to match package and platform power budget constraints. Portable and low-power embedded contexts have more stringent power limits, and encounter such market-driven obstacles earlier.

In this paper, we propose a modeling framework that captures characteristics and requirements of future MPU designs, following Moore’s Law scaling under power limits. Figure 1 summarizes our modeling approach with respect to MPU power consumption. All geometry information is derived based on MPU M1 half-pitch (F), which is the half-pitch of typical stagger-contacted metal1 (M1) bitlines. We define representative unit cell layouts and corresponding *A-factors* that enable the modeling of unit cell areas for SRAM and standard-cell logic circuit fabrics, in terms of F . We then

calculate transistor density as a function of F . With the physical technology parameters from ITRS Process Integration, Devices and Structures (PIDS) [3] [4] and Interconnect [5] [6] chapters, we model capacitance density, and finally derive MPU power model. Our closed-form MPU power model enables analysis of requirements for MPU design parameter scaling, such as MPU frequency scaling, in light of power constraints. Other ITRS requirements such as defect density are also derived from the MPU model.

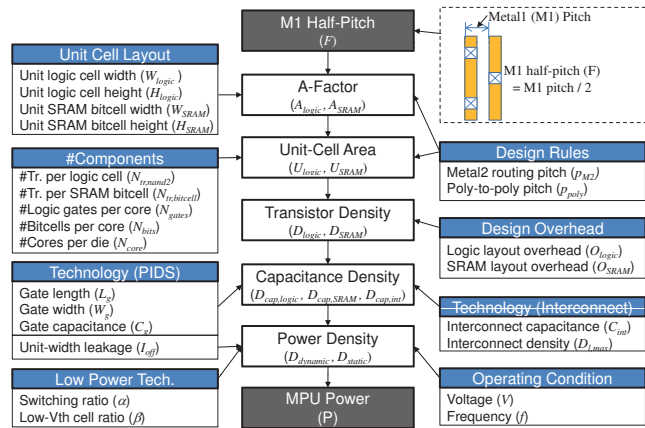


Fig. 1. Data flow for MPU power modeling.

The remainder of this paper is organized as follows. Section II presents the scaling trends of the required parameters for power estimation, and Section III presents the density models of MPU designs. Section IV gives our power models, and Section V discusses the power-constrained frequency roadmap for the ITRS MPU system driver. Section VI summarizes our conclusions.

II. TECHNOLOGY SCALING FOR THE 2009 ITRS

Basic physical technology parameters which serve as input parameters for power estimation are modeled in various chapters of the ITRS. Dynamic power estimation is based on capacitance of transistors and interconnects, along with power supply voltage. Static power estimation requires per unit-width device leakage current (I_{off}) values. Relevant transistor- and interconnect-related parameters are predicted in the ITRS PIDS and Interconnect chapters, respectively.¹ Table I summarizes ITRS predictions for key parameters, versus a timeline of MPU M1 half-pitch (minimum metal width or space of given technologies) and the physical gate length (L) which governs transistor electrical characteristics.

Given these physical parameters, we can model power of MPUs based on integration level, i.e., transistor count per die. Transistor count scaling of MPUs is determined following historical and current MPU design trends as follows.

- **Moore’s Law.** The number of transistors is doubled at every technology generations, following Moore’s Law. We define technology generation at the time when M1 half-pitch becomes $\sqrt{2}$; 2× more transistors can be packed into the same die area as with previous technology.

¹Transistor-related parameters are obtained by ITRS contributors using the Model for Assessment of CMOS Technologies And Roadmaps (MASTAR) [2].

TABLE I
TECHNOLOGY PARAMETER SCALING IN ITRS.

†: Preliminary values for ITRS 2009. Transistor types: HP = high-performance; LOP = low operating power; LSTP = low standby power.
‡: Values from ITRS 2008 updates [6], being updated for ITRS 2009.
†‡: Values from ITRS 2008 updates [4]. Anticipated ITRS 2009 Ioff values of HP and LSTP transistors are $0.1\mu A$ and $0.1nA$, respectively.

Technology year	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022	2023	2024
Dimensions																
MPU Half-Pitch (nm) [†]	54	45	38	32	27	24	21	18.9	16.9	15.0	13.4	11.9	10.6	9.5	8.4	7.5
Physical Lgate (HP) (nm) [†]	29	27	24	22	20	18	17	15	14.0	12.8	11.7	10.7	9.7	8.9	8.1	7.4
Supply Voltage Parameters																
VDD (HP) [†]	1.00	0.97	0.93	0.90	0.87	0.84	0.81	0.78	0.76	0.73	0.71	0.68	0.66	0.64	0.62	0.60
VDD (LOP) [†]	0.95	0.95	0.85	0.85	0.80	0.80	0.75	0.75	0.70	0.70	0.65	0.65	0.60	0.60	0.60	0.60
Interconnect-Related Parameters																
Capacitance (pF/cm) [‡]	1.75	1.75	1.75	1.65	1.65	1.65	1.55	1.55	1.55	1.40	1.40	1.40	1.20	1.20	-	-
Total length (Km/cm^2) [‡]	2.00	2.22	2.50	2.86	3.13	3.57	4.00	4.55	5.00	5.56	6.25	7.14	7.69	9.09	-	-
Transistor-Related Parameters																
Ioff (HP) ($\mu A/\mu m$) ^{†‡}	0.17	0.46	0.71	0.70	0.64	0.69	0.57	0.62	0.56	0.55	0.38	0.40	0.44	0.48	-	-
Ioff (LSTP) ($pA/\mu m$) ^{†‡}	30.5	30.7	30.2	30.2	30.9	31.7	30.2	31.0	32.7	33.8	26.2	23.9	33.8	28.9	-	-
Cg,total (HP) ($fF/\mu m$) [†]	1.00	0.98	0.93	0.96	0.97	0.75	0.76	0.71	0.67	0.64	0.58	0.55	0.53	0.50	0.48	0.46

- **Constant die area.** Die area is assumed to be constant over the course of the roadmap, and is broken into logic, memory (SRAM), and integration overhead components. For given die area, the number of logic and memory transistors are decided, consistent with integration overhead area for analog and mixed-signal cores, I/O blocks, whitespace between blocks, etc., which is assumed to be around 30% of die area.
- **Multi-core organization.** Multiple processing units in the MPU product chip was modeled starting with the 2001 ITRS and the 130nm technology node. The MPU model assumes a multi-core organization with shared memory.

Initial numbers of cores, logic transistors per core, and SRAM cache size (bytes) per die in 2007 were assumed to be 4, 50Mtx,² and 16MBytes, respectively; this content fits in $310mm^2$ die area with reasonable integration overhead.

The ITRS MPU model grows number of transistors at the same pace as M1 half-pitch scaling. From year 2009 to 2013, M1 half-pitch shrinks by $2\times$ every two years, and from year 2014 to 2022 by $2\times$ in every 3.5 years. Based on multi-core design trends, we scale the number of cores and the number of transistors per core by $\sqrt{2}\times$ at every generation, respectively, so that the total number of transistors in a die is doubled at every generation. Table II shows the number of cores and transistors per die from 2009 to 2024.³

III. DENSITY MODEL

MPU design-specific parameter scaling was presented in Section II. In this section, we discuss the detailed modeling approaches for design-specific parameters, such as design area and density, in terms of M1 half-pitch. We model the area of logic and SRAM separately, then obtain the entire MPU die area with an integration overhead factor. From the given number of cores in MPU and the number of transistors per core, we calculate the area and density of transistors in MPU. To model the area, we first propose unit layouts of a logic cell and an SRAM bitcell. Then, total area can be calculated by multiplying the area of unit cells and their respective counts.

A. Logic Density Model

Unit logic cell. To determine the unit cell size for logic, we assume a layout of a unit logic cell, i.e., two-input NAND gate (NAND2), as shown in Figure 2.

We only consider simplest design rules. The height and width of the unit logic cell are defined by unit drawing pitches. Traditionally, drawing pitch for standard cells has been decided from M2 (intermediate) minimum routing pitch values. However, as the restricted design rule (RDR) is introduced, poly pitch can be used as the unit pitch for width direction, and M1 (or M2) minimum routing pitch can be used as the unit pitch for height direction.

For width calculation, we consider two types of NAND2 implementation. Figure 2(a) shows a traditional NAND2 gate drawn by M2

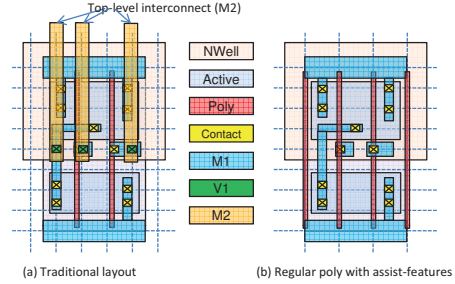


Fig. 2. Unit logic cell layout, two-input NAND gate. (a) Traditional layout (b) Regular-poly layout.

minimum routing pitch; Figure 2(b) shows the regular layout style for the NAND2 gate. With traditional layout style, NAND2 requires at least three M2 routing pitches in the width direction to connect input/output pins (A, B, Y) using M2, so the NAND2 cell can be drawn with three M2 pitches in width. With restricted one-direction routing for poly, around four M2 routing pitches are required to draw all intra-cell connections, e.g., source/drain connections. Layers under M2 do not need to be drawn on routing grid. With adoption of design for manufacturability (DFM) techniques, e.g. assist features, poly pitch would be proper to define width rather than M2 routing pitch. Since there exist two input poly lines and two assist-features at the boundaries in a NAND2 gate, four poly lines (three poly pitches) are required. Each product team or company can have different layout styles. Considering that we will see more restricted layout styles in future nodes, we use layout style (b) for the unit logic cell. The height of the unit cell can differ based on the application domain (high density, high speed, etc.). We assume eight M2 routing tracks for the cell height.

A-factor for logic. Let poly-to-poly pitch and M1 and M2 routing pitches be p_{poly} , p_{M1} , and p_{M2} , respectively. Width of the logic unit cell can be generalized as “(number of N (or P) transistors + 1) $\times p_{poly}$ ”. Here, p_{poly} is calculated by the summation of a minimum poly width, minimum contact width and two times poly-to-contact minimum space values. Each parameter value is obtained by averaging the design rules from leading-edge industry design manuals, and we find that p_{poly} and p_{M2} can be approximated as a multiple of p_{M1} or F .

$$p_{poly} \approx 1.50p_{M1} = 3F \quad (1)$$

$$p_{M2} \approx 1.25p_{M1} = 2.5F \quad (2)$$

Using the drawing pitch definitions and the number of drawing pitches required for the layout in Figure 2(b), we can calculate the area of the unit NAND2 (U_{nand2}) as

$$\begin{aligned} U_{logic} &= 3p_{poly} \times 8p_{M2} = 45(p_{M1})^2 \\ &= 180F^2 = A_{logic}F^2 \end{aligned} \quad (3)$$

We now have unit logic cell area in terms of M1 half-pitch (F) and a constant (A_{logic}) which is analogue to the A-factor in DRAM bitcell

²Mtx (Btx) stands for million (billion) transistors.

³For SRAM transistor count, we assume six transistors per bitcell, and nine bitcells per byte including an error correction bit.

TABLE II
MPU DESIGN PARAMETER SCALING IN ITRS.

Technology year	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022	2023	2024
#Logic cores per MPU	5.66	6.73	8.00	9.51	11.3	12.7	14.3	16.0	18.0	20.2	22.6	25.4	28.5	32.0	36.0	40.0
#Tr. per logic core (Mtx)	70.7	84.1	100	119	141	159	178	200	224	252	283	317	356	400	448	504
#Logic tr. per MPU (Btr.)	0.40	0.57	0.80	1.13	1.60	2.02	2.54	3.20	4.03	5.08	6.40	8.06	10.2	12.8	16.1	20.3
#SRAM tr. per MPU (Btr.)	1.81	2.56	3.62	5.12	7.25	9.13	11.5	14.5	18.3	23.0	29.0	36.5	46.0	58.0	73.0	92.0

area estimation which has been used in ITRS.⁴

Area and density model for logic. From the unit cell area and gate count for the die, we can calculate the total area of logic transistors (S_{logic}) as

$$S_{logic} = O_{logic}U_{logic}N_{core}N_{gate} \quad (4)$$

where N_{core} and N_{gate} are the number of cores in a die and the number of logic gates used in a core, respectively, and O_{logic} represents a design area overhead due to design integration, such as whitespace area for placement flexibility and power planning area, etc. We use 2.0 for O_{logic} in our model.

Transistor density of logic area is then calculated by dividing the number of transistors by the derived logic area area (S_{logic}).

$$D_{tr,logic} = N_{tr,logic}/S_{logic} = N_{tr,nand2}/(O_{logic}U_{logic}) \quad (5)$$

where the number transistors in a NAND2 gate ($N_{tr,nand2}$) is four, based on our unit cell definition.

B. SRAM Density Model

Unit SRAM bitcell. From the conventional 6T SRAM bitcell layout, we use the layout in Figure 3 as the unit SRAM bitcell.

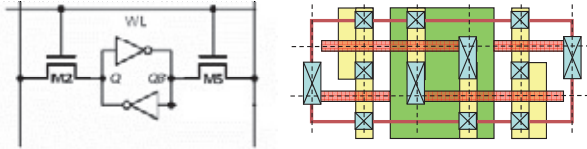


Fig. 3. SRAM 6-T bitcell schematic and layout.

In height direction, we can observe that two parallel poly lines must be placed. Therefore, height can be modeled as “ $2p_{poly}$ ”. Bitcell width is determined by three distances; 1) Wordline contact to bitline contact, 2) N-active to the next N-active, and 3) transistor widths. The distance from the wordline contact to bitline contact can be defined by minimum metal design rules, and the distance from N-active to the next N-active can be determined by the maximum of distance calculated from metal design rules or distance calculated from diffusion design rules. In Figure 3, we can observe that five metal routing pitches are required in width direction.

A-factor for SRAM bitcell. Similar with the logic A-factor, after modeling the height and width, the unit area for an SRAM bitcell (U_{SRAM}) can be modeled using A-factor for SRAM (A_{SRAM}) and M1 half-pitch (F).

$$\begin{aligned} U_{SRAM} &= 2p_{poly} \times 5p_{M1} = 15(p_{M1})^2 \\ &= 60F^2 = A_{SRAM}F^2 \end{aligned} \quad (6)$$

Area and density model for SRAM. Using the unit cell area, we can calculate the total area occupied by SRAM transistors in the MPU:

$$S_{SRAM} = O_{SRAM}U_{SRAM}N_{core}N_{bits} \quad (7)$$

where N_{bit} is the number of bits per core, and O_{SRAM} represent a SRAM design overhead due to peripheral circuits, such as precharging, sensing and input/output muxing and buffering circuits. From industry SRAM layout data, we use 1.6 for O_{SRAM} .

Transistor density of the SRAM layout region area is

$$D_{tr,SRAM} = N_{tr,bitcell}/(O_{SRAM}U_{SRAM}) \quad (8)$$

where the number of transistors in a bitcell ($N_{tr,bitcell}$) is six.

⁴A-factor for logic is calculated as 180 in Eq. (3). However, after fitting with actual data collected from industry cell libraries, we decide to use 175 for the logic A-factor.

IV. MPU POWER MODEL

In this section, we propose a closed-form MPU power model using the basic parameters in Section II and the transistor area and density models in Section III.

A. Capacitance Density Model

For dynamic power estimation, we first model the capacitance density. From Tables PIDS2a and PIDS2b in the PIDS chapter, we can obtain unit width gate capacitance (C_g), which includes fringe and overlap capacitances. With the average width value (W_g) of transistors,⁵ capacitance of a single transistor (C_{tr}) and capacitance density are

$$C_{tr} = W_g C_g \quad (9)$$

$$D_{cap,tr} = D_{tr,logic} C_{tr} \quad (10)$$

However, not all the capacitances consume power. We define active capacitance density considering switching ratio (α), which is the ratio of actively charging or discharging nets (devices) to the total number of nets (devices) in design during operation.

$$D_{cap,active,tr} = \alpha D_{cap,tr} \quad (11)$$

From Tables INTC2a and INTC2b in the Interconnect chapter, we can obtain MPU estimated wirelength density ($D_{l,max}$).⁶ Considering routing congestion and power routing regions, we apply an additional derating factor to interconnect density $D_{l,max}$ and define effective wirelength $D_{l,eff}$ as $1/3 \times D_{l,max}$. We obtain per-unit length wire capacitance from the ITRS Interconnect Chapter (Tables INTC2a and INTC2b). Since $D_{l,max}$ assumes a stack of one M1 layer and five intermediate layers, we simply use the majority’s characteristic $C_{intermediate}$. Interconnect capacitance density ($D_{cap,int}$) and active interconnect capacitance density ($D_{cap,active,int}$) are defined as

$$D_{cap,int} = D_{l,eff} C_{intermediate} \quad (12)$$

$$D_{cap,active,int} = \alpha D_{cap,int} \quad (13)$$

Finally, total active capacitance density is calculated as

$$D_{cap,logic} = D_{cap,active,tr} + D_{cap,active,int} \quad (14)$$

B. Dynamic Power Density Model

For dynamic power estimation, we assume that MPU logic consists of low standby power transistors (LSTP) for non-timing critical paths and high performance transistors (HP) for timing critical paths. β denotes the fraction of HP transistors relative to total logic transistors in an MPU design.⁷ We also assume that the switching activity of LSTP transistors is $1/3 \times$ that of HP transistors. Dynamic power density in the logic area is

$$D_{dynamic,logic} = D_{cap,logic} f V_{HP}^2 \left(\beta + \frac{1}{3} (1 - \beta) \right) \quad (15)$$

Capacitance density of the SRAM region can be calculated similarly, and can be higher than that of logic. However, only a bitline and a wordline become active during operation. Therefore, the active capacitance density of SRAM blocks can be considered to be relatively smaller than that of logic. In our work, we simply assume that the dynamic power density of SRAM ($D_{dynamic,SRAM}$) is $1/10 \times$ that of logic.

⁵Average transistor width is assumed to be $5F$.

⁶Total interconnect length in $1cm^2$ die is calculated by assuming that only one of every three minimum pitch wiring tracks for metal1 (M1) and five intermediate levels of interconnects are populated. The wiring lengths for each level are then summed to calculate the total interconnect length per square centimeter of active area.

⁷The portions of HP and LSTP transistors are assumed as 10% and 90%, respectively.

C. Static Power Density Model

We use the same transistor constituent assumption (β) in calculating logic static power density.

$$D_{static,logic} = 0.5D_{tr,logic}V_{HP} \times (\beta I_{off,HP} + (1 - \beta)I_{off,LSTP}) \quad (16)$$

where the constant factor 0.5 represents that either NMOS or PMOS transistors in a logic cell are off-state at any given time.

For SRAM static power density, we again assume that multi-Vth technique is used, so that bitcell array uses LSTP transistors, and only β amount of transistors in the periphery are HP transistors.

$$D_{static,SRAM} = 0.5D_{tr,SRAM}V_{HP} \times \left\{ \frac{O_{SRAM} - 1}{O_{SRAM}} (\beta I_{off,HP} + (1 - \beta)I_{off,LSTP}) + \frac{1}{O_{SRAM}} I_{off,LSTP} \right\} \quad (17)$$

D. Total MPU Power Model

Finally, total MPU power is given by

$$P_{total} = (D_{dynamic,logic} + D_{static,logic})S_{logic} + (D_{dynamic,SRAM} + D_{static,SRAM})S_{SRAM} \quad (18)$$

V. POWER-CONSTRAINED MPU FREQUENCY MODEL

Based on the area, density and power models, other MPU design parameters, e.g., chip size, integration level and operating frequency, must be considered carefully to not exceed a permitted power limit.⁸

To keep up with Moore's Law, the transistor count scaling needs to follow the MPU design integration trends described in Section II. We evaluate integration overhead values, which is the ratio of the remaining area to the total die area after subtracting logic and memory transistors area, using the proposed new A-factor models and the transistor count scaling trends. With the old $310mm^2$ die area, the integration overhead grows to an unreasonable 41% over the course of the roadmap. To maintain reasonable integration overhead, we reduce die size to $260mm^2$, which matches recently reported MPU products⁹ and leads to 29.5% of integration overhead.

For MPU power roadmap with fixed die size and transistor count scaling, key levers at the design stage are frequency and switching activity factor. Figure 4 shows the dynamic and static power predictions based on our power model, using the PIDS device CV/I scaling factor (13% per year) as a frequency scaling factor. Due to aggressive standby power reduction techniques, such as multi-Vth, multi-Lgate, power-gating, etc., and development of low-leakage transistors, static power does not increase rapidly. However, dynamic power increases and exceeds 200W in 2015.

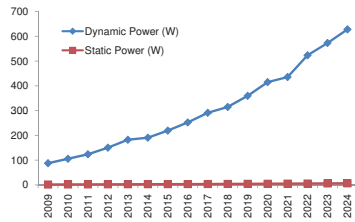


Fig. 4. Dynamic and static power prediction for MPUs. Frequency scaling factor is 13% per year and switching ration is fixed at 15%.

To mitigate this rapid dynamic power scaling, we introduces a design factor which is used to reduce switching activity, into dynamic power estimation. Initial switching activity in 2007 is assumed to be 0.15, and we introduce a 5% reduction of switching activity per year ($\alpha_N = 0.95\alpha_{N-1}$). This implies that low-power techniques to reduce the switching activity of a design must be developed to continue

⁸The power limit for a platform, e.g., desktop PC, is viewed as having a 130-150W limit due to market forces.

⁹The die size of 45nm CPUs, i.e., Intel®Core™i7 (Bloomfield) [7] and AMD Opteron™ (Shanghai) [8] are $263mm^2$ and $258mm^2$, respectively.

Moore's Law scaling of the MPU product class. Using 5% switching activity reduction factor, we can reduce dynamic power due to the active capacitance density as shown in Figure 5.

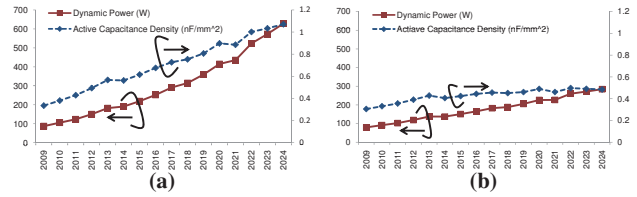


Fig. 5. (a) Power prediction with 13% per year CV/I scaling with fixed 0.15 switching ratio. (b) Power prediction with 13% per year CV/I scaling with 5% switching ratio reduction factor per year from an initial 0.15 switching ratio.

However, power still increases rapidly, so that power exceeds 200W in year 2019. This is mainly due to the fast 13% per year device CV/I scaling. To maintain a reasonable power value forced by markets, the frequency scaling needs to slow down. At 8% improvement per year of frequency scaling, the maximum MPU power is 132W throughout the roadmap years as shown in Figure 6.

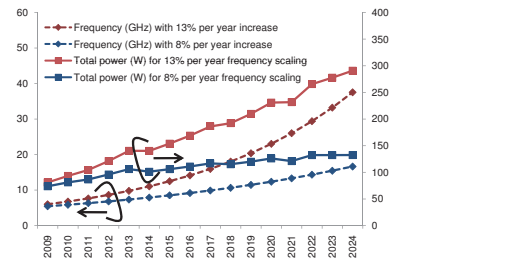


Fig. 6. Power-constrained frequency scaling. 8% per year scaling of frequency can meet 200W power limit.

The performance gap between the device and MPU can be utilized for further improvement in design size and power, since fast transistors will allow for use of smaller cell sizes during timing optimization. We note that the 8% frequency scaling trajectory needs to be met only by the large-sized MPU as we assumed. If we do not increase #functions, and shrink and improve current-generation designs, when power per die is smaller than the platform limit we can use the maximum device scaling factor (13%/year) as the frequency scaling factor.

VI. CONCLUSIONS

We describe a consistent, holistic modeling approach for MPU area, power and frequency roadmapping that forms the basis for planned updates in the 2009 edition of the ITRS. To maintain Moore's Law scaling under power constraints, we restrict MPU frequency scaling within the envelope of transistor performance scaling. In addition, we highlight the looming challenge of improved dynamic power reduction, which has recently received less attention than static power reduction.

REFERENCES

- [1] International Technology Roadmap for Semiconductors, <http://public.itrs.net/>.
- [2] Model for Assessment of CMOS Technology and Roadmaps (MASTAR), <http://www.itrs.net/Links/2007ITRS/LinkedFiles/PIDS/MASTAR5/MASTARDownload.htm>.
- [3] Process Integration, Devices and Structures (PIDS) Chapter in ITRS 2007 Edition, http://www.itrs.net/Links/2007ITRS/2007_Chapters/2007_PIDS.pdf.
- [4] Process Integration, Devices and Structures (PIDS) Chapter in ITRS 2008 Updates, <http://www.itrs.net/Links/2008ITRS/Home2008.htm>.
- [5] Interconnect Chapter in ITRS 2007 Edition, http://www.itrs.net/Links/2007ITRS/2007_Chapters/2007_Interconnect.pdf.
- [6] Interconnect Chapter in ITRS 2008 Updates, <http://www.itrs.net/Links/2008ITRS/Home2008.htm>.
- [7] Intel®Core™i7-920 Processor, Intel, <http://ark.intel.com/Product.aspx?id=37147>.
- [8] AMD Opteron 2384 Review, <http://www.techradar.com/reviews/pc-mac/pc-components/processors/amd-opteron-2384-shanghai-484339/specification>.