Frequency and Yield Optimization using Power Gates in Power-Constrained Designs

Nam Sung Kim[1], Jun Seomun[2], Abhishek Sinkar[1], Jungseob Lee[1], Tae Hee Han[3], Ken Choi[4], and Youngsoo Shin[2]

[1]University of Wisconsin, Madison, WI, U.S.A., [2]Korea Advanced Inst. of Sci. & Tech., Taejon, Korea,
[3]Sungkyunkwan University, Suwon, Korea, [4]Illinois Inst. of Tech., Chicago, IL, U.S.A.,

ABSTRACT

Manufactured dies exhibit a large spread of maximum frequency and leakage power due to process variations, which have been increasing with technology scaling. Reducing the spread is very important for maximizing the frequency and the yield of power-constrained designs, because otherwise many dies that do not satisfy frequency or power constraints would be discarded. In this paper, we propose two optimization methods to improve the maximum operating frequency and the yield using power gates that already exist in many power-constrained designs. In the first method, we consider the designs of multiple cores, where each of them can be independently power-gated. When each core shows different frequencies due to within-die variations, the strength of a power gate in each core is adjusted to make their maximum operating frequencies even. This allows faster cores to consume less active leakage power, reducing the total power consumption well below a power constraint in a globally-clocked design. We subsequently increase global supply voltage for higher overall frequency until the power constraint is satisfied. In our experiments assuming multicore processors with 2~16 cores, the maximum operating frequency was improved by 4~23%. In the second method, we take leaky-but-fast dies (which otherwise would be discarded) and adjust the strength of the power gates such that they can operate in an acceptable power and frequency region. The problem is extended to designs employing a frequency binning strategy, where we have an additional objective of maximizing the number of dies for higher frequency bins. In our experiments with ISCAS benchmark circuits, most discarded fast-but leaky dies were recovered using the second method.

Categories and Subject Descriptors

B.7.1 [Types and Design Styles]: VLSI (very large scale integration)

General Terms

Design, Performance

Keywords

power gate, frequency, yield, optimization

1. INTRODUCTION

As technology scales below 65nm, manufactured dies began to exhibit a substantial spread of device delay and leakage power both across dies and within each die due to process variations. This results in a large amount of *maximum operating frequency* (F_{max}) and *total power* (P_{tot}) variations. Process variations are often classified into two categories: *die-to-die* (D2D) and *within-die* (WID) [1].

D2D variations, traditionally modeled using process corners, are resulted from the lot-to-lot, wafer-to-wafer, or a portion of within-wafer variations of processing temperatures, equipment properties, wafer polishing, and wafer placement; they affect all the devices on a die equally. Meanwhile, WID variations, consisting of random and systematic components, are mainly caused by random dopant fluctuations, line-edge roughness, and aberrations in stepper lens; they induce different electrical characteristics of devices across a die. Furthermore, due to increasing proportion of WID variations, the conventional approach that assumes the same process parameters (e.g. L_{eff}) at one particular process corner becomes too pessimistic. For example, as each individual core in multicore processors is becoming very small, spatially correlated WID variations lead to *core-to-core* (C2C) F_{max} and leakage power (P_{leak}) variations.

Traditionally, F_{max} constraints have determined the yield of manufactured dies. Recently, leakage power also affects the yield in power-constrained designs [2]-[5] due to limited total power budget, which is imposed by limited power-supply and cooling-solution capacity. Thus, many leaky dies, satisfying F_{max} constraints sufficiently, may be discarded because they exceed power (and thermal) constraint; this becomes worse as technology scaling increases the spread of leakage power (e.g. 20× in 0.13um technology [2]) and its percentage in total power.

To mitigate yield loss due to process variations, *adaptive body biasing* (ABB) has been adopted as an effective technique since it can either reduce P_{leak} or improve F_{max} [4][6]. Several problems, however, have been identified in the use of ABB: *reverse body biasing* (RBB) increases the amount of threshold voltage variations [7]; ABB in dual-V_{th} design is very difficult due to different body effect coefficient of high- and low-V_{th} devices [8]; ABB to both NMOS and PMOS devices requires a triple-well process. *Adaptive voltage scaling* (AVS) after manufacturing testing is another option to mitigate yield loss [9]. However, employing multiple voltage domains on a chip to mitigate WID variations (such as core-to-core F_{max} and P_{leak} variations in multicore processors) is very challenging in practice due to the increased cost of 1) design, verification, and testing time [10] as well as 2) voltage regulators and decoupling capacitors [11].

A power-gating technique, using an on-off current switch located between power supply rails and a circuit, is commonly used to minimize standby leakage power [12]. When a circuit is not active, the switch is turned off, disconnecting power supply from the circuit. This reduces a substantial amount of P_{leak} in standby mode. The switch often consists of an array of fixed size transistors connected in parallel [12]. To prevent any timing errors due to more-than-expected voltage drop across the switch, it is sized large enough not to exceed a certain amount of voltage drop for the peak current consumption of the circuit [12]. A programmable (or variable-width) power gate to control the F_{max} and P_{leak} spread of dies utilizes current switches that can be configured to have different widths [13]. Thus, it can be considered as another option to tune F_{max} and P_{leak} of a design after manufacturing.

In this paper, we present two methods to optimize F_{max} and yield of power-constrained designs implemented with power gates. Our main contributions can be summarized as follows:

- We present a method to optimize F_{max} in power-constrained designs implemented with multiple power-gating domains, which are often used in multicore processors for a more efficient power management. For example, in multicore processors, some cores are faster and leakier than others while the F_{max} is often limited by the slowest core (unless frequency

ISLPED'09, August 19–21, 2009, San Francisco, California, USA.

islands are used) due to WID frequency and leakage variations. Hence, to reduce excessive P_{leak} of the faster cores in a die, we adjust the strength of their power gates such that their frequencies become equal to the F_{max} of the slowest core. Then the total power consumption may become well below the power constraint. This allows us to increase the global supply voltage for higher operating frequency until the power constraint is just satisfied again.

- We provide a method to optimize yield by recovering discarded dies due to excessive P_{leak} in power-constrained designs for two different cases: 1) ASIC-type fixed-P_{leak} and 2) microprocessor-type variable-P_{leak} constraints. Discarded fast-but-leaky dies can be recovered by adjusting (weakening) the strength of power gates to reduce the P_{leak} until each die can satisfy the P_{tot} constraint (but within the F_{max} constraint).

The remainder of this paper is organized as follows. In the next section, we will briefly review a programmable-width power-gating technique. A F_{max} optimization method for power-constrained designs having multiple power-gating domains is presented in Section 3. In Section 4, we address two yield optimization problems for power-constrained designs: one for fixed frequency target and the other for frequency binning. The experimental setup and methodology are presented in Section 5, and we draw conclusions in Section 6.

2. BACKGROUND

2.1 Programmable-Width Power Gate

Figure 1 illustrates a concept of a programmable-width power gate [13]. PMOS header switches (NMOS footer switches can be used instead) are connected to the SLEEP signal through NAND gates; the other input of each NAND gate is connected to a configuration bit. Hence, the header switches with a configuration bit set to "0" are always turned off while those with a configuration bit set to "1" can be turned on (when SLEEP is "0") and off (when SLEEP is "1"). As a result, a less number of switches will be turned on during active mode (when SLEEP is "1") when we have less configuration bits set to 1. This increases the series resistance and the voltage drop across the switches (decreases VV_{DD} linearly), which reduces the leakage power (exponentially) and the speed (close-to-linearly with a small amount of VV_{DD} drop) of the circuit. Therefore, we can control leakage power and speed (thus F_{max}) of a particular die depending on how we program the configuration bits after manufacturing.

2.2 Programming Configuration Bits

Configuration bits can be programmed by *one-time-programmable* (OTP) *e-fuses* after manufacturing characterization of F_{max} and P_{leak}. *Note that characterizing and programming can be done like any other variation compensation techniques (e.g., ABB or AVS).* To reduce the number of configuration bits while providing various sizes of header switches that designers can choose from, we arrange the header switches in exponentially increasing widths as shown in Figure 1. Note that each switch, in turn, may consist of several smaller switches connected in parallel. This allows the sum of header widths proportional to the binary value of configuration bits. This mechanism facilitates easier programming and requires a less number of configuration bits than the original scheme in [13]. For example, only 8-bit configuration bits are enough to provide 4mV granularity for adjusting VV_{DD} while 256 bits are needed for the same granularity if the original scheme in [13] is used. Note that the header switches and buffers within the dotted-line box represent the required resource to implement a conventional power gate. Therefore, the additional area by the NAND gates, which is an overhead of programmable power-gating, is negligible since the large power-gating switches are driven by existing buffers.

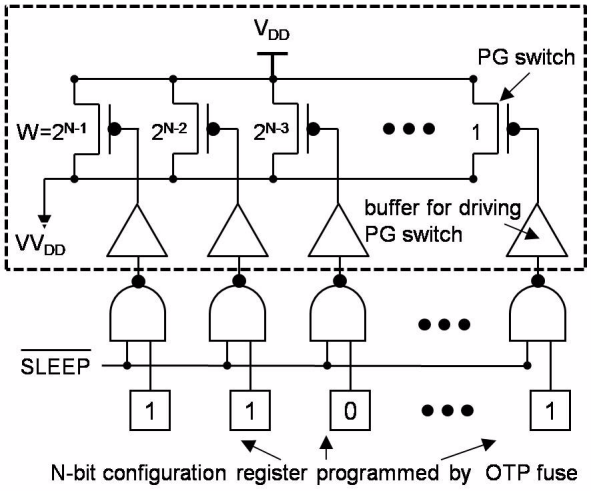


Figure 1. Programmable power gate.

3. OPTIMIZING F_{MAX} OF DESIGNS WITH MULTIPLE POWER-GATING DOMAINS

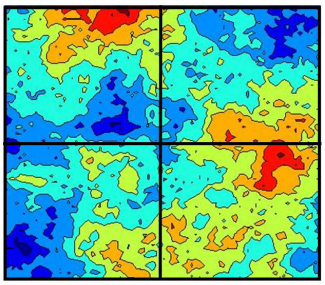
A large design such as SoCs or multicore processors consists of several IPs or cores where each of them has a its own power gate (or power-gating domain) to support efficient standby leakage power reduction. For example, we can disable three cores using the associated power gates to minimize standby leakage power when only one core can serve workload demand in a quadcore processor [14]. Meanwhile, if we perform frequency binning of this multicore processor, the bin where a particular die is placed is determined by the F_{max} of the slowest core (unless each core is clocked at its own frequency using frequency islands). Furthermore, the maximum power consumption or the *thermal design power* (P_{TDP}) of a multicore processor operating at $V_{DD,TDP}$ is often limited by power and thermal constraints when all the cores are running simultaneously at sustainable maximum performance. As a result, V_{DD} (thus F_{max}) cannot be increased beyond $V_{DD,TDP}$ due to the power and thermal constraints although there is available headroom to increase V_{DD} (thus F_{max}).

As F_{max} of each IP or core in an SoC or a multicore processor becomes noticeably different due to increasing WID process variations, the F_{max} of the die employing a global-clocking scheme is limited by the slowest IP or core. Note that many commercial SoCs and multicore processors use a global-clocking scheme to avoid clock-domain crossing that complicates design, verification, and test. In a multicore processor, for example, assume that we have a programmable-width power gate per core. Then we can set the power-gating configuration bits of each core such that the F_{max} of each core can be set as even as possible to that of the slowest core. This will lead to a significant amount of each core's P_{leak} reduction without impacting the F_{max} of the processor, which, in turn, lets the total power consumption (P_{tot}) of the processor become much smaller than the power constraint. Consequently, we can increase the global V_{DD} (thus the F_{max}) of the processor until power and other constraints such as *maximum junction temperature* (T_j), *maximum* V_{DD} ($V_{DD,max}$), etc. are not violated so that we can put the die in a higher frequency bin.

Let $VV_{DD,i}$ be represented as follows:

$$VV_{DD,i} = v_i(W_i) \times V_{DD} \quad (1)$$

where W_i is the effective channel width (the sum of header switches that are connected to configuration bits of "1") of a power gate in domain i; and v_i is a function that returns a VV_{DD} scaling factor of domain i, proportional to W_i. Note that VV_{DD} is a strong function of W_i and it is always lower than V_{DD}, as it must when we use a programmable-width power gate alone. However, if we scale global V_{DD} together with a local programmable-width power gate, VV_{DD} can become higher than initial V_{DD}. Note that modulating the strength of a power gate only affects the VV_{DD} of the corresponding domain while scaling global V_{DD} affects the VV_{DD} of all the domains. Then the main objective here is to maximize the F_{max} of the die as given by:



W	X
(1.00, 1.53)	(1.21, 1.62)
Y	Z
(1.28,1.88)	(1.05, 1.00)

(a) (b)

Figure 2. (a) Systematic V_{th} variation map for a quadcore processor. (b) The initial F_{max} and P_{leak} of each core, normalized to the F_{max} of the slowest core and the P_{leak} of the least leaky core, respectively.

$$F_{max} = min\{F_{max,1}(VV_{DD,1}), ..., F_{max,N}(VV_{DD,N})\} \quad (2)$$

where $F_{max,i}$ and $VV_{DD,i}$ are the F_{max} and the VV_{DD} of the circuitry in power-gating domain i, while the constraints imposed on total power consumption (P_{TDP}), junction temperature ($T_{j,max}$), and V_{DD} ($V_{DD,max}$) are satisfied as below:

$$P_{tot} = \sum_{i=1}^{N} P_{tot,i}(VV_{DD,i}, F_{max}) \le P_{TDP} \quad (3)$$

$$T_j \le T_{j,max}, V_{DD} \le V_{DD,max} \quad (4)$$

where N is the number of power-gating domains and $P_{tot,i}$ ($P_{dyn,i}$+$P_{leak,i}$) is the P_{tot} of the circuit in power-gating domain i.

Figure 2-(a) shows a V_{th} variation map for a quadcore processor (details of how to obtain the map are explained in Section 5), where each rectangle represents a core. A pair of numbers within each core shown in Figure 2-(b) corresponds to the F_{max} and the P_{leak} of each core; they are each normalized to the smallest values from all four cores. Since the F_{max} of a multicore processor employing a global-clocking scheme is limited by the slowest core — W, we take core X, Y, and Z and change the power-gating configuration bits of each of them (i.e. decrease their VV_{DD}) until their F_{max}'s become as close as possible to the F_{max} of core W. Since the VV_{DD} of faster cores is reduced, the P_{leak} of core X, Y, and Z become smaller as well, as shown in Figure 3-(a) where all the cores now have the same F_{max}. Note that the sum of P_{leak} becomes 4.71 while it was 6.03 in Figure 2-(b), which is about 22% P_{leak} reduction. As a result, we reduced the P_{tot} by 7% in Figure 3-(a) assuming that sum of P_{leak} before we program the power-gating configuration bits of each core is 40% of P_{TDP} at $V_{DD,TDP}$. Then we increase the global V_{DD} (thus the F_{max}) until the P_{tot} becomes the same as before (i.e. the P_{tot} of Figure 2-(b)) as long as $T_{j,max}$ and $V_{DD,max}$ constraints are not violated; Figure 3-(b) shows the result, where we can see F_{max} increase by 10.5%. T_j was checked using

W	X
(1.00, 1.53)	(1.00, 1.12)
Y	Z
(1.00,1.17)	(1.00, 0.89)

W	X
(1.11, 2.16)	(1.11, 1.35)
Y	Z
(1.11,1.38)	(1.11, 1.13)

(a) (b)

Figure 3. Normalized F_{max} and P_{leak} of cores: a) after applying core-by-core programmable power-gating and (b) after increasing global V_{DD} until power constraint is satisfied again.

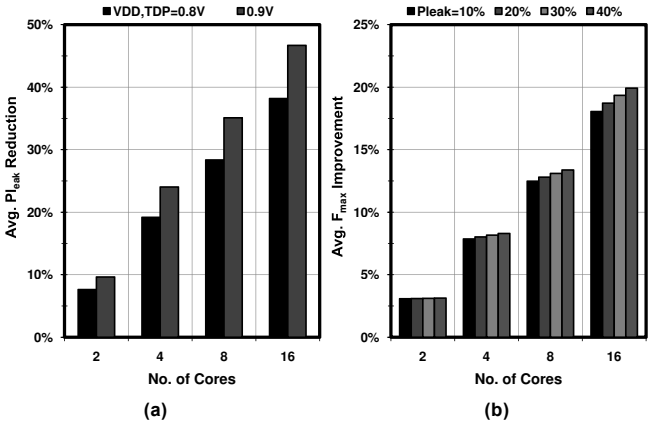


Figure 4. (a) P_{leak} reduction and (b) F_{max} improvement versus the number of cores per die. $V_{DD,TDP}$ is 0.8V in (b).

HotSpot 4.1 [15]. Note that the VV_{DD} of each core will be slightly different and voltage level shifters may be required.

Figure 4-(a) and (b) show the averages P_{leak} reduction and the F_{max} improvement versus the number of cores per die after applying the proposed optimization method to 100 die samples. As the number of cores per die increases, we have more relative P_{leak} and F_{max} spread among the cores as noted in [17], which provides more opportunities in reducing P_{leak} for faster cores and improving the overall F_{max} of a die as a result. The optimization method applied to the 100 samples show that P_{leak} is reduced by 8~38% for 2~16-core processors as shown in Figure 4-(a). Since P_{leak} scales more substantially at higher voltage, higher $V_{DD,TDP}$ can provide more P_{leak} reduction opportunities; $V_{DD,TDP}$ equal to 0.9V offers 2~9% more P_{leak} reduction in Figure 4-(a), potentially leading to more F_{max} improvement.

When $V_{DD,TDP}$ is 0.8V and P_{leak} is 40% of P_{TDP} at $V_{DD,TDP}$, we improve F_{max} by 3.1~19.9% on average for 2~16-core processors as shown in Figure 4-(b). The percentage of P_{leak} in P_{TDP} should also impacts the F_{max} improvements since P_{leak} can change more dramatically than P_{dyn} for adjusting programmable-width power gates and V_{DD}. However, as illustrated in Figure 4-(b), sweeping the P_{leak} percentage from 10% to 40% results in only 0.1%~1.9% difference in F_{max} improvement for 2~16 cores since P_{leak} scales at a similar rate to P_{dyn} when supply voltage is around the $V_{DD,TDP}$ region.

When the target P_{TDP} of a design is fixed, F_{max} improvement can be affected by $V_{DD,TDP}$ in two ways: 1) we can have better P_{leak} scaling at higher $V_{DD,TDP}$ as shown in Figure 4-(a), but 2) we have less power headroom to improve F_{max} because a design with lower $V_{DD,TDP}$ is assumed to have higher power than one with higher $V_{DD,TDP}$ at the same V_{DD}. Hence, the $V_{DD,TDP}$ difference should not affect the average F_{max} improvement significantly; $V_{DD,TDP}$ equal to 0.9V provides less than 1% difference in F_{max} improvement for 2~16-core processors even when P_{leak} is 40% of P_{TDP}.

4. OPTIMIZING YIELD IN POWER-CONSTRAINED DESIGNS

4.1 Designs with Frequency Target: Fixed P_{leak} Constraint

In typical ASIC designs, F_{max} and P_{tot} of each die have to satisfy *frequency target* (F) and *power target* (P), respectively, i.e.

$$F_{max}(VV_{DD}) \ge F \quad (5)$$

$$P_{tot} = P_{dyn}(VV_{DD}) + P_{leak}(VV_{DD}) \le P \quad (6)$$

where P_{dyn} is dynamic power and P_{leak} corresponds to leakage power, both during active mode. Note that the F_{max} here is not an operating frequency but a maximum frequency achievable from a

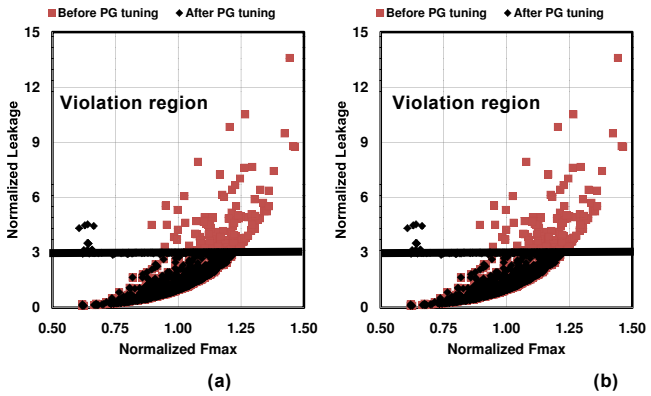


Figure 5. Normalized P_{leak} and F_{max} distribution before and after applying the proposed yield optimization under constant P_{leak} constraint: (a) C432 and (b) C3540.

particular die; as long as the F_{max} is not smaller than F, dies will operate at F. Since all the components (F_{max}, P_{dyn}, and P_{leak}) in (5) and (6) are functions of VV_{DD} as described in Section 2, some fast-but-leaky dies that satisfy (5) may be forced back to satisfy both (5) and (6) after adjusting VV_{DD} (or the strength of a programmable-width power gate) In addition, P_{dyn} can be safely considered to be constant since (i) operating frequency is fixed to F; (ii) VV_{DD} after adjusting the programmable-width power gate usually takes the values not too far from the nominal VV_{DD} (otherwise circuits will be very slow); and (iii) P_{dyn} is weakly dependent on process variations [6]. This lets us consider (6) as a leakage target:

$$P_{leak}(VV_{DD}) \le P' \tag{7}$$

where P' is the difference between P and P_{dyn}, i.e., leakage budget. Since the P_{dyn} becomes slightly smaller than the P_{dyn} at V_{DD} for non-zero voltage drop across the header switches, (7) is conservative. Hence, (6) will be satisfied as well if (7) is satisfied.

To assess how we can improve yield (percentage of dies that satisfy both (5) and (6)), we took C432 and C3540, both of which are the ISCAS benchmark circuits. The header switches are connected as shown in Figure 1 and the configuration bits are initially set to all "1." We assumed that process variations are applied to each example circuit 1,000 times emulating the same number of manufactured dies; the details of V_{th} and L_{eff} variations are explained in Section 5. Then the P_{leak} and the F_{max} of each die, obtained through SPICE simulation with a 45nm technology model, are shown as scatter plots in Figure 5 (square boxes). We arbitrarily assumed that P' in (7) is set to 3× of the nominal P_{leak} — $P_{leak,nom}$ (leakage power of a die without process variations); F is set to -3σ of the nominal F_{max}— $F_{max,nom}$ where σ is standard deviation of F_{max} in 1,000 dies (note that we cannot make dies run faster through changing configuration bits in this particular example, since all bits are initially set to 1 thereby causing the smallest voltage drop across header switches). The accepted dies, satisfying both (5) and (6) are shown within boxes; 116 and 118 dies are rejected in the example circuits, respectively, due to the leakage constraint, i.e. about 88% of yield for both examples.

For each of rejected dies due to excessive P_{leak}, we tried to change its configuration bits (by setting some of them to 0) so that it can fall within the box of accepted dies. The results are shown as another scatter plot (diamond shape) in Figure 5; only 12 and 16 dies are rejected now, i.e. we improved yield by recovering 99% and 98% of discarded fast-but-leaky dies, respectively, through the proposed optimization method.

Table 1 summarizes the yield loss due to violating P_{leak} constraints and the recovery before and after the optimization is applied, respectively. For the dies violating the P_{leak} constraints (exceeding 3× and 4× of the nominal P_{leak}), we adjust the strength of power-gating switches until we satisfy the P_{leak} constraints; meanwhile we must not violate the target frequency constraint F — -3σ frequency of the nominal F_{max} among 1000 samples per each

Table 1. Yield loss recovery for fixed P_{max} constraints

Circuit	P_{leak} Constraint	# of Violations		Yield Loss Recovery (%)
		Before PG Optimization	After PG Optimization	
C432	$3\times P_{leak,nom}$	116	12	90%
	$4\times P_{leak,nom}$	56	6	89%
C499	$3\times P_{leak,nom}$	118	13	89%
	$4\times P_{leak,nom}$	60	4	93%
C880	$3\times P_{leak,nom}$	101	12	88%
	$4\times P_{leak,nom}$	45	4	91%
C1355	$3\times P_{leak,nom}$	106	6	94%
	$4\times P_{leak,nom}$	43	2	95%
C1908	$3\times P_{leak,nom}$	121	12	90%
	$4\times P_{leak,nom}$	67	7	90%
C2670	$3\times P_{leak,nom}$	119	9	92%
	$4\times P_{leak,nom}$	60	2	97%
C3540	$3\times P_{leak,nom}$	118	18	85%
	$4\times P_{leak,nom}$	65	5	92%
Avg.	$3\times P_{leak,nom}$	114	12	90%
	$4\times P_{leak,nom}$	57	4	92%

circuit. We assume that the optimization process failed and recovery was unsuccessful, if the F_{max} of a die becomes slower than the target frequency constraint.

On average, we recovered 90% and 92% of discarded fast-but-leaky dies when the P_{leak} constraints are $3\times P_{leak,nom}$ and $4\times P_{leak,nom}$, respectively. Relaxing the P_{leak} constraint gives fewer violations before applying the optimization, but it also provides more opportunity to recover the discarded dies from the violations, resulting in a similar or higher percentage of yield improvement within a certain range of P_{leak} constraints.

4.2 Designs with Frequency Binning: Variable P_{leak} Constraint

In Section 4.1, we considered a fixed frequency target, which is typical for ASIC designs. In high-performance microprocessor designs, on the other hand, we have a list of frequency targets, $F_1 < F_2 < \ldots < F_N$; the F_{max} of a die is compared to these frequency targets and then it is put into an appropriate bin, i.e.

$$f(F_{max}) = \begin{cases} F_i, \text{ if } F_i \le F_{max} < F_{i+1}, i = 1, 2, \ldots, N-1 \\ F_N, \text{ otherwise} \end{cases} \tag{8}$$

where f is a function that assigns an operating frequency; this process is called frequency binning. As a result, dies have different leakage power constraints depending on the bins where they are placed, since P_{dyn}, which is proportional to operating frequency, is different for different bins. Meanwhile the sum of P_{dyn} and P_{leak} has to be no greater than a fixed power constraint.

We can continue to use the programmable-width power gate to improve yield as we did in Section 4.1, except that we have varying P_{leak} constraints. Since the bins of higher frequencies are preferred, our optimization objective is to maximize the operating frequency of each die, i.e.

$$maximize(f(F_{max}(VV_{DD}))) \tag{9}$$

such that power constraint is satisfied,

$$P_{tot} = P_{dyn}(f(F_{max}), VV_{DD}) + P_{leak}(VV_{DD}) \le P \tag{10}$$

Since P_{dyn} is weakly dependent on VV_{DD} as we described in Section 4.1, we assume that it is a function of F_{max} alone. Let the nominal VV_{DD} be the VV_{DD} before we change configuration bits of

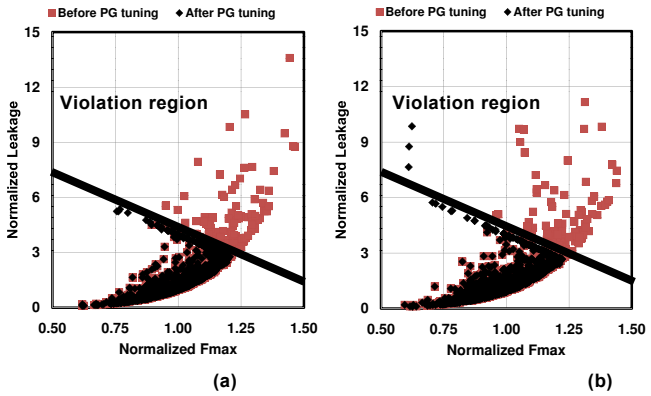


Figure 6. Normalized P_{leak} and F_{max} distribution before and after applying the proposed yield optimization under variable P_{leak} constraint: (a) C432 and (b) C3540.

the programmable-width power-gating. Then (10) can be simplified to:

$$P_{tot,i} \approx g_i(VV_{DD}) \cdot P_{dyn,nom} + h_i(VV_{DD}) \cdot P_{leak,nom} \quad (11)$$

where $g_i(VV_{DD})$ is a function that returns the F_{max} of die i at particular VV_{DD}, which is normalized to the nominal F_{max} (i.e. F_{max} at the nominal VV_{DD}); $P_{dyn,nom}$ is the P_{dyn} at the nominal F_{max}; $h_i(VV_{DD})$ is a function that returns P_{leak} of die i at particular VV_{DD}, which is normalized to the nominal P_{leak} (i.e. P_{leak} at the nominal VV_{DD}); $P_{leak,nom}$ is the P_{leak} at the nominal VV_{DD}.

We repeat the same experiment as presented in Section 4.1, but using the optimization process described in this section. In this experiment, we assume that power constraint P is equal to $P_{dyn,nom}$+ $4\times P_{leak,nom}$, where $P_{dyn,nom}$ takes about 60% of P. Before we begin the optimization, the dies above the diagonal line are discarded because their P_{tot} as described by (11) exceeds P. As explained earlier, the dies of higher F_{max} have less P_{leak} budget.

We tried to change the configuration bits of each of rejected dies so that it can fall within accepted region. The results are shown as diamond shapes in Figure 6. Initially 106 and 117 dies were rejected from C432 and C3540, respectively. However, after optimization, only 1 and 3 dies were rejected, recovering almost all the dies in these particular examples.

Table 2 summarizes the yield loss improvement after the optimization is applied to the fast-leaky dies that violate the power constraint P. For the dies violating the target power constraint, we adjust the strength of power-gating switches until we satisfy the target power constraint, while maximizing F_{max} of each die. As F_{max} of a die becomes slower by adjusting the strength of power-gating switches, more P_{leak} will be allowed since F_{max} decrease reduces P_{dyn} of the die, which allows more power budget for P_{leak}. To set the target power constraint P, we assume that $P_{tot,max}$ is $P_{dyn,nom}$ + $4\times P_{leak,nom}$ at the nominal F_{max}, and that the ratios between $P_{dyn,nom}$ and $4\times P_{leak,nom}$ at the nominal F_{max} are 1) 0.6 and 0.4 and 2) 0.7 and 0.3 to see the sensitivity on the percentage of P_{leak}, in P, since the percentage of P_{leak} in most recent digital designs like microprocessors has been between 30~40% [18].

On average, we recovered 98% and 98% of the discarded dies, when the P_{leak} constraints at the $F_{max,nom}$ point are $0.3\times P$ and $4\times P$, respectively. Note that less P_{leak} budget (e.g., $P_{leak} = 0.3\times P$ at the nominal F_{max}) incurs more violations than more P_{leak} budget ($P_{leak} = 0.4\times P$ at the nominal F_{max}) before applying the optimization due to less P_{leak} headroom for P_{leak} variations at higher F_{max}.

5. METHODOLOGY

For a quadcore processor used in Section 3 we generated spatially correlated Vth and $Leff$ maps through a models shown in [17]. The die area was assumed to be 35mm^2; WID correlation distance coefficient ϕ (0.5), WID V_{th} variation $\sigma_{V_{th}}^{sys}$ (6.4%), and D2D variation $\sigma_{V_{th}}^{D2D}$ (5.0%) were used to model WID and D2D Vth and

Table 2. Yield loss recovery for variable P_{max} constraints

Circuit	P_{leak} at $F_{max,nom}$	# of Violations		Yield Loss Recovery
		Before PG Optimization	After PG Optimization	
C432	$0.3\times P$	204	2	99%
	$0.4\times P$	106	1	99%
C499	$0.3\times P$	208	4	98%
	$0.4\times P$	110	2	98%
C880	$0.3\times P$	190	0	100%
	$0.4\times P$	104	0	100%
C1355	$0.3\times P$	193	3	98%
	$0.4\times P$	94	1	99%
C1908	$0.3\times P$	214	2	99%
	$0.4\times P$	120	0	100%
C2670	$0.3\times P$	202	0	100%
	$0.4\times P$	108	0	100%
C3540	$0.3\times P$	203	5	98%
	$0.4\times P$	117	3	97%
Avg.	$0.3\times P$	202	2	98%
	$0.4\times P$	108	1	99%

$Leff$ variations of 100 dies with the same parameters presented in [17]. We broke each variation map into 80×80 grid points, and obtained a pair of Vth and $Leff$ values from each grid point. For each grid point modeled with a 24-stage FO4 inverter (INV) chain for F_{max} [18] and a large number of INV (50%), NAND (30%), and NOR (20%) gates for P_{leak}, we applied the corresponding pair of Vth and $Leff$ values to a 32nm technology model [16] to obtain F_{max} and P_{leak} (and F_{max} and P_{leak} scaling factors relative to the F_{max} and P_{leak} at $V_{DD,TDP}$) as functions of V_{DD} using SPICE and a curve fitting tool; each gate excluding INVs had a various number of inputs (2~4) and randomly selected input states were applied to measure P_{leak}. Note that, in power-gating domain i, the F_{max} scaling factor is decided by the slowest grid point [17] while the P_{leak} scaling factor is obtained by averaging all the P_{leak} scaling factors in all the grid points corresponding to each core region.

We assumed that P_{TDP} at $V_{DD,TDP}$ is 120W, which is typical for a server class multicore processor, and the P_{leak} percentage in P_{TDP} at $V_{DD,TDP}$ is between 10% and 40%. With the assumed P_{TDP} and the P_{leak} percentage in P_{TDP} at $V_{DD,TDP}$, we can estimate P_{dyn} and P_{leak} using the generated F_{max} and P_{leak} scaling factors at any given VV_{DD}. With the calculated $P_{tot,i}$, we compute the T_j of each domain in a die sample using HotSpot [15]. We used 0.3K/W for the convection resistance [19] and the given die size (35mm^2) assuming that the T_{jmax} is 100°C; 120W power consumption across a 35mm^2 die results in T_j=~100°C with the provided convection resistance. Note that we may underestimate the T_j if the core size becomes larger since we assume that an active core has a uniform power across the entire core area (due to lack of detailed power models associated with a floorplan).

We also demonstrate the effectiveness of the proposed yield optimization methods presented in Section 4 by performing SPICE Monte-Carlo simulations with a 45nm technology model [16]. To model D2D and WID process variations, we applied (0.4V, 4σ) and (10nm, 3σ) for NMOS/PMOS V_{th} and L_{eff} variations, respectively. We chose a subset of ISCAS85 benchmark circuits (C432, C499, C880, C1355, C1908, C2670, and C3540) for the experiments. Initial power-gating switches at the nominal corner were sized such that the maximum voltage drop across the switches does not exceed 50mV for the peak current consumption ($I_{DD,max}$) of each circuit; $I_{DD,max}$ was estimated by applying 1,000 vectors at 100°C die temperature and selecting a pair of vectors causing the worst-case current consumption. For each die of a particular circuit (see, for example, Figure 5 and 6), the same 1,000 vectors were applied to

derive F_{max} and P_{leak} of the die. P_{leak}, which is active-mode leakage, was approximated by standby-mode leakage, i.e. steady-state (instead of transient) leakage was measure for P_{leak}.

6. CONCLUSION

We have proposed two optimization methods to improve maximum operating frequency and yield of power-constrained designs implemented with power gates. The first optimization method improved F_{max} of power-constrained designs implemented with multiple power-gating domains by adjusting the strength of power gates, domain by domain, which is followed by scaling global supply voltage for higher operating frequency. Our experimental results show that the proposed optimization method improved the overall die F_{max} by 3~21% for 2~16-core processors where each core has an independent programmable-width power gate.

The second optimization method recovers yield loss due to excessive active P_{leak}; a necessary amount of P_{leak} is reduced until each die can satisfy a power constraint within a frequency target. To demonstrate the effectiveness of the optimization method, we examined two different design styles: 1) ASIC-type fixed-P_{leak} and 2) microprocessor-type variable-P_{leak} constraints. Our experimental results show that the optimization method applied to various ISCAS benchmark circuits recovered 90% and 98% of discarded dies on average for the targeted fixed- and variable-P_{leak} constraints, respectively.

7. REFERENCES

[1] K. Bowman, S. Duvall, and J. Meindl, "Impact of die-to-die and within-die Parameter fluctuations on the maximum clock frequency distribution for gigascale integration," IEEE JSSC, Vol. 37, No. 2, pp. 183~190, Feb. 2002.

[2] S. Borkar, T. Karnik, and V. De, "Design and reliability challenges in nanometer technologies," in Proc. DAC, pp. 75, Jun. 2004.

[3] D. Blaauw and F. Najm, "Leakage power: trends, analysis and avoidance," in Proc. ASP-DAC, pp. 18~21, Jan. 2005.

[4] Z. Songqing V. Wason, and K. Banerjee, "A probabilistic framework to estimate full-chip subthreshold leakage power distribution considering within-die and die-to-die P-T-V variations," in Proc. ISLPED, pp. 156~161, Aug. 2004.

[5] K. Agarwal et al., "Parametric yield analysis and optimization in leakage dominated technologies," IEEE TVLSI, Vol. 15, No. 6, pp. 613~623, Jun. 2007.

[6] J. Tschanz et al., "Adaptive body bias for reducing impacts of die-to-die and within-die parameter variations on microprocessor frequency and leakage," IEEE JSSC, Vol. 37, No. 11, pp. 1396~1402, Nov. 2002

[7] S. Narendra et al., "Impact of using adaptive body bias to compensate die-to-die Vt variation on within-die Vt variation," in Proc. ISLPED, pp:229~232, Aug. 1999.

[8] Y. Yasuda et al., "System LSI multi-Vth transistors design methodology for maximizing efficiency of body-biasing control to reduce Vth variation and power consumption," in Proc. IEDM, pp. 68~71, Dec. 2005.

[9] T. Chen and S. Naffziger, "Comparison of adaptive body bias (ABB) and adaptive supply voltage (ASV) for improving delay and leakage under the presence of process variation", IEEE TVLSI, Vol. 11, No. 5, pp. 888~899, Oct. 2003.

[10] D. Lackey et al., "Managing power and performance for system-on-chip designs using voltage islands," in Proc. ICCAD, pp. 195~202, Nov. 2002.

[11] W. Kim et al., "System level analysis of fast, per-core DFVS using on-chip switching regulators," in Proc. HPCA, pp. 123~134, Feb. 2008.

[12] K. Shi and D. Howard, "Challenges in sleep transistor design and implementation in low power designs," in Proc. DAC, pp.113~116, Jun. 2006.

[13] H. Deogun et al, "Adaptive MTCMOS for dynamic leakage and frequency control using variable footer strength," in Proc. SOCC, pp. 147~150, Sep. 2005.

[14] "Power gating and turbo mode: Intel talks Nehalem at IDF," http://arstechnica.com/news.ars/post/20080820-power-gating-and-turbo-mode-intel-talks-nehalem-at-idf.html.

[15] HotSpot, http://lava.cs.virginia.edu/HotSpot/index.htm.

[16] Predictive technology model, http://www.eas.asu.edu/~ptm.

[17] S. Herbert and D. Marculescu, "Characterizing chip-multiprocessor variability-tolerance," in Proc. DAC, pp. 313~318, Jun. 2008.

[18] N. Kim et al., "Optimizing total power through pipelining and parallel processing under the presence of process variations," in Proc. ICCAD, pp. 534~539, Nov. 2005.

[19] R. Rao et al., "Throughput of multi-core processors under thermal constraints," in Proc. ISLPED, pp.201~206, Aug. 2007.