Workshop on Duplicating, Decontructing and Debunking (WDDD), in conjunction with ISCA 39, June 2012, pp. 1-7.

Reevaluating Fast Dual-Voltage Power Rail Switching Circuitry

Ronald G. Dreslinski, Bharan Giridhar, Nathaniel Pinckney,
David Blaauw, Dennis Sylvester, and Trevor Mudge
EECS Department - University of Michigan, Ann Arbor, MI.

Abstract

Several recent papers have been published proposing the use of dual-voltage rails and fast switching circuitry to address bottlenecks or overcome process variation in near-threshold computing systems. The published results yield boosting transition times of 7-10ns, which, in some cases, is needed for the architectural contributions to be justified. However, the analysis of these circuits assumed incorrect core models, ideal off-chip power supplies, and non-worst case scenarios. When realistic bonding capacitance and inductance are included, proper core models are used, and worst-case simulation is performed these transitions times can be off by 3x, adversly impacting the potential gains in the system.

In this paper we analyze the previously proposed designs, and propose a new design in order to achieve the desired transition time. By using a third internal power rail and some additional on-chip capacitors the supply voltage noise can be isolated from the main external power supplies. Ultimately the new circuit achieves the desired 10ns transition time, allowing the architectural contributions of the previous studies to still be attainable.

1 Introduction

Power has become a first class design constraint, not only in embedded/mobile devices where battery life is critical, but also in warehouse scale server farms [4]. Recent work advocates Near-Threshold Computing (NTC)—optimizing circuits for an aggressively-scaled supply voltage just above the transistor threshold voltage—as an approach to significantly improve throughput and energy efficiency [1, 9]. NTC systems are designed to operate at substantially lower voltages than conventional designs (thereby achieving far greater energy efficiency) by explictly designing circuits to combat the increased leakage and variability challenges of low-voltage operation. NTC enables higher throughput by allowing many more cores within a fixed thermal design power (TDP) budget—the maximum power the chip's packaging is designed to dissipate—while achieving far greater energy-efficiency per core [1].

Recently several proposals have used the idea of dual-voltage rails to overcome variation and performance bottlenecks in NTC systems [6, 5, 2]. In these proposals two supply voltage rails are used and cores are quickly boosted to higher frequencies when needed. The techniques by Dreslinski et al. [2] rely on boosting transition times of around 10ns. Figure 1 shows a sensitivity study to boosting latency, indicating that latencies larger than 10-100ns mitigate the savings of the boosting technique significantly. Miller et al. [6, 5] do not include such a sensitivity study and are assumed to be similar to those presented by Dreslinski et al. However, in all these papers the circuit simulations are incorrect. Focusing on the work by Miller et al. [5], they incorrectly model the core, assume ideal voltage rails, and simulate a non-worst case scenario. If these mistakes are properly accounted for, then simulation shows that the achieved transition time is longer than 30ns (a 3x discrepancy). The architectural techniques still remain valid, provided an alternative circuit can be designed that achieves the desired transition time.

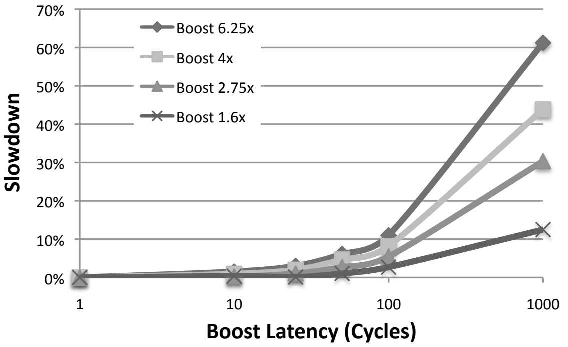


Figure 1: Sensitivity results from Dreslinski et al. [2]. The data shows that for latencies greater than 10s of nanoseconds the technique experiences significant slowdown.

To achieve the desired transition time we propose a new circuit approach. The new approach switches cores across three power rails—V_{ddLow} (∼400 mV), V_{ddHigh} (∼600mV), and V_{boost} (an internal staging supply powered by on-chip capacitors). Each supply is distributed to the header of each core's local power grid, and power

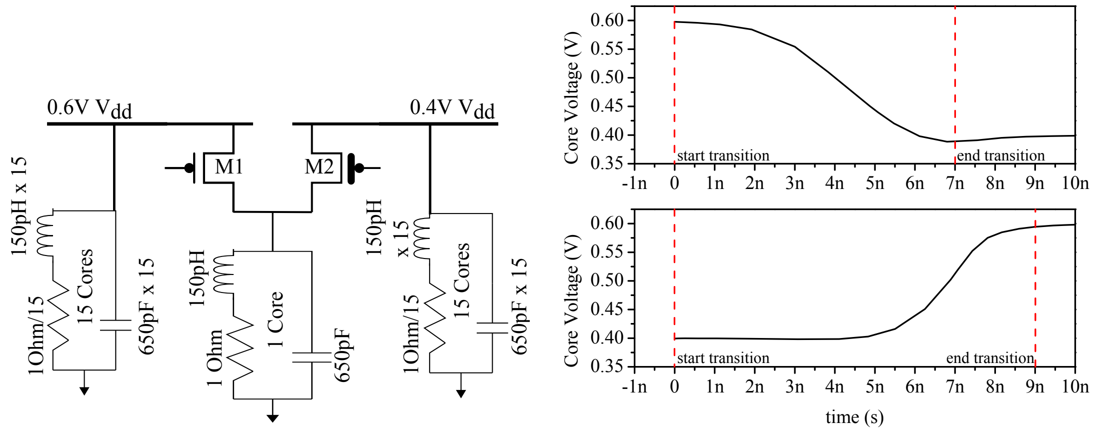


Figure 2: Circuit diagram and simulation results proposed by Miller et al. [5]. They show a transition time of ∼10ns to stabilize.

transistors are used to select the supply. By performing well-timed boost transitions in two steps, from V_{ddLow} to V_{boost} and then to V_{ddHigh}, the voltage instability that arises from the large switching current is isolated to the V_{boost} capacitor network, enabling extremely fast (∼9ns) transitions between grids. Ultimately the results in the previous papers are still achievable, but require a slightly more complicated circuit design.

The rest of the paper is organized as follows: Section 2 will re-create the original design, adding in proper circuit details. Section 3 will showcase one potential solution to achieve the desired transition times. The paper will conclude in Section 4.

2 Duplicating & Debunking

2.1 Original Circuit

For the purpose of this discussion we will use the circuit presented by Miller et al. [5] to illustrate the problems with the proposed dual-voltage boosting circuits. The circuit and simulation results presented by Miller et al. are presented in Figure 2. For the analyses in this paper we will assume the voltage to be stable when it settles within 10% of the voltage boosting differential (20mV for a boost from 400mV to 600mV).

2.2 Errors in Core Model and Supply Rails

The first problem with the model by Miller et al. is that they present an erroneous, lumped resistance-inductance-capacitance (RLC) model for a single Nehalem core, and use power and parasitic data from [3]. The Nehalem RLC core model includes five errors: (1) the model is of a single Nehalem core but the parasitic data is for an AMD 4-core Phenom processor [3]; (2) Miller et al. denote a core capacitance of 650 pF but simulations of their circuits reveal a capacitance of 650 nF was used in their analysis; (3) packaging parasitics and off-chip decoupling capacitance were incorrectly lumped with the core parasitics; (4) wire bond or C4 packaging models were not included in the simulation; (5) the core was modeled as a resistor instead of a voltage-controlled current source with quadratic dependence, although this error is debatable among experts.

By replicating the simulations of Miller et al. we found that the core capacitance used in their analysis needs to be 650 nF, not 650 pF, to match their reported transition times—a 1000x difference in capacitance value which may have been a typo in the circuit schematic. The source of the parasitic data, Leverich et al. [3], used for the Nehalem RLC model is actually from a AMD 4-core Phenom X4 9850 processor, not a Nehalem processor, and Leverich et al. do not imply that the parasitic data can be used directly to create an accurate lumped RLC model of a core. Our simulation of their model is presented in Figure 3, we show similar transition times but required a 1000x larger capacitance value.

The parasitic data in from Leverich et al. includes core capacitance (C_{core}), internal decoupling capacitance (C_{dec_int}), external decoupling capacitance (C_{dec_ext}), and packaging inductance (L_{ext}). Miller et al. appeared to have summed all external and internal capacitances and included the total, along with the packaging inductance, in a lumped RLC model of the core. However, only a subset of this parasitic data should be included in the core model. The external decoupling capacitance (C_{dec_ext}) and packaging inductance (L_{ext}) are not on-chip, and thus should not be included in the core model. Instead L_{ext} should be included in a packaging model connected in series between the external voltage sources and on-chip power supply switches. By placing the L_{ext} in series with the core current, it inaccurately shields the core current

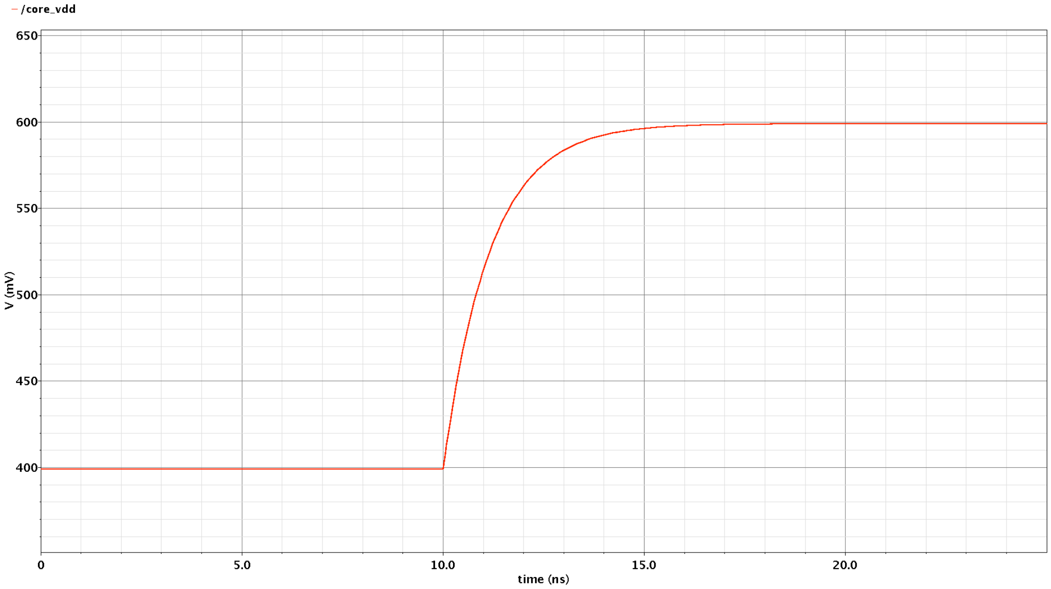


Figure 3: Our attempt to re-create the simulation presented by Miller et al. In order to achieve close to the same transition time a 1000x larger capacitance was used (likely a typo in their circuit schematic). The total transition, simulated via SPICE, takes only ∼3ns to stabilize.

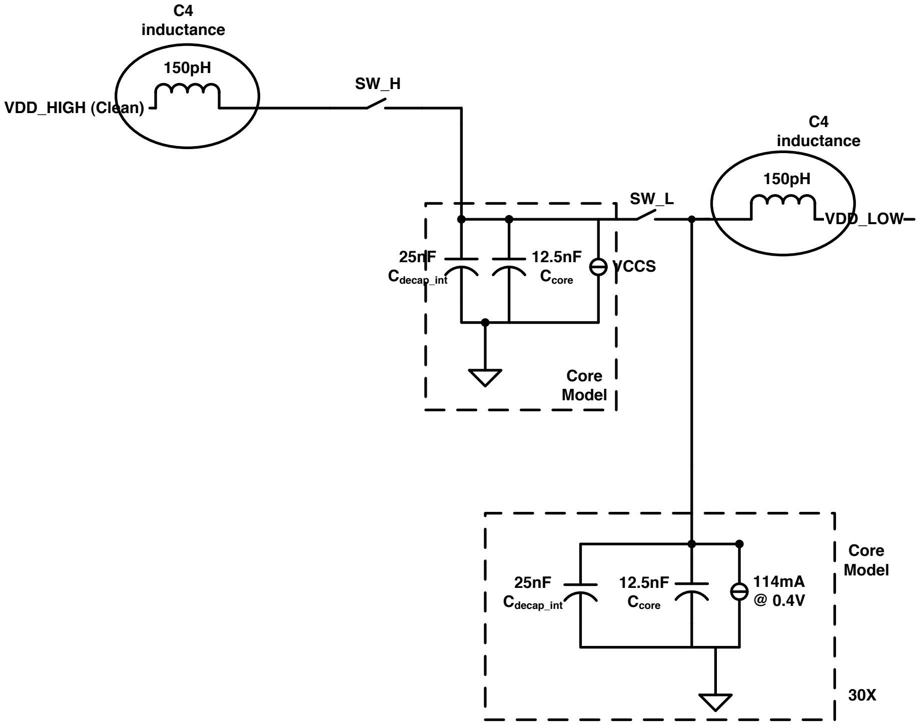


Figure 4: Correct version of the system with proper core models and power supply inductances. Worst case scenario presented where 30 cores are at low voltage, and one switched to high voltage.

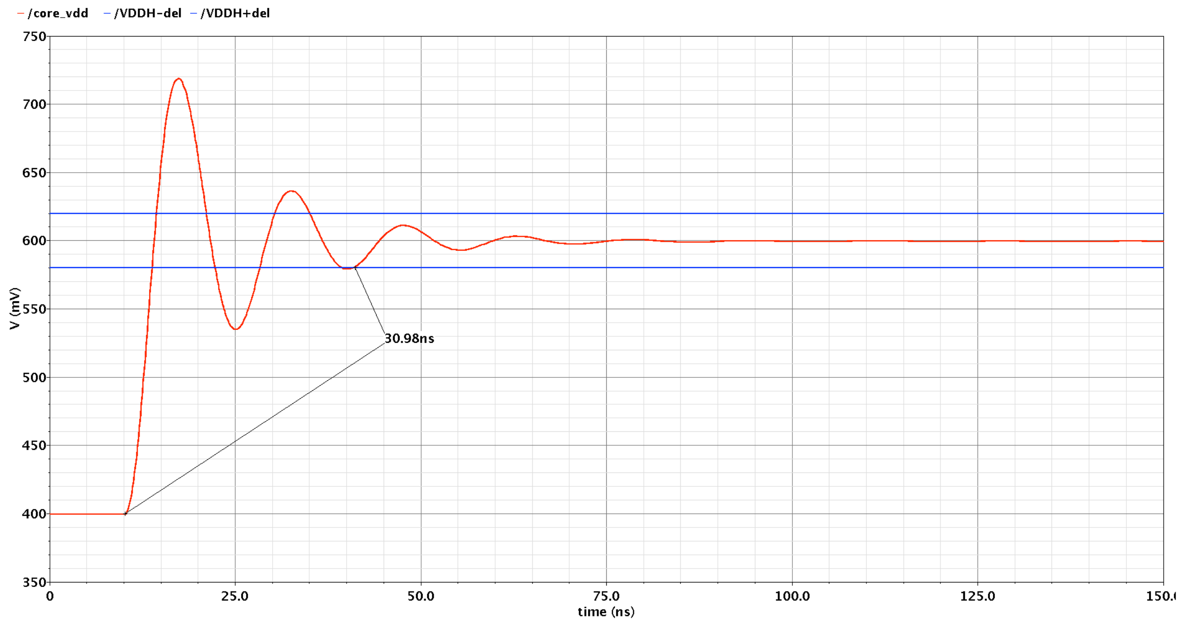


Figure 5: Simulation with worst case configuration, where all cores are at low voltage and one transitions to the high voltage. The total transition, simulated via SPICE, takes ∼30ns to stabilize.

from the grid and reduces voltage drop durring transition. On-chip decoupling capacitance (C_{dec_int}) should be included with the core. C_{dec_ext} may be dropped entirely since it is dependent on printer circuit board (PCB) design and is not visible to the core through the packaging model. Furthermore, the numbers given by Leverich et al. are of four cores, not a single core. The corrected core capacitance is (C_{core} + C_{dec_int})/4 = 38 nF.

A corrected core, packaging, and supply switch model is show in Figure 4. The core capacitance was reduced to 38 nF and a packaging model is included with L_{ext} data from Leverich et al. Lastly, at nominal voltage, the current consumption is quadratically dependent on voltage [8], not linearly, thus a resistor is insufficient for modeling. The core current source in the updated model uses a voltage-controlled current source with a quadratic dependence on power supply voltage.

2.3 Worst Case Simulation

Finally, Miller et al. also perform a simulation where there are 15 cores on both the high and low voltage supplies and a single core is transitioned from one rail to the other. In this case both rails only see a 1/16th change in current draw, and the impact of the transitioning core is minimal. The worst-case transition occurs when one rail contains all the cores and a single core is transitioned to a new rail. Keeping the core count constant with the study done by Miller et al. we re-run the updated core model/voltage rail simulation with 30 cores on the low voltage rail and transition a single core to the high rail. The results of the simulation are plotted in Figure 5. This incurs more ringing on the voltage rail and the transition stabilizes after 30ns, a 3x increase from the results reported by Miller et al.

2.4 Implications

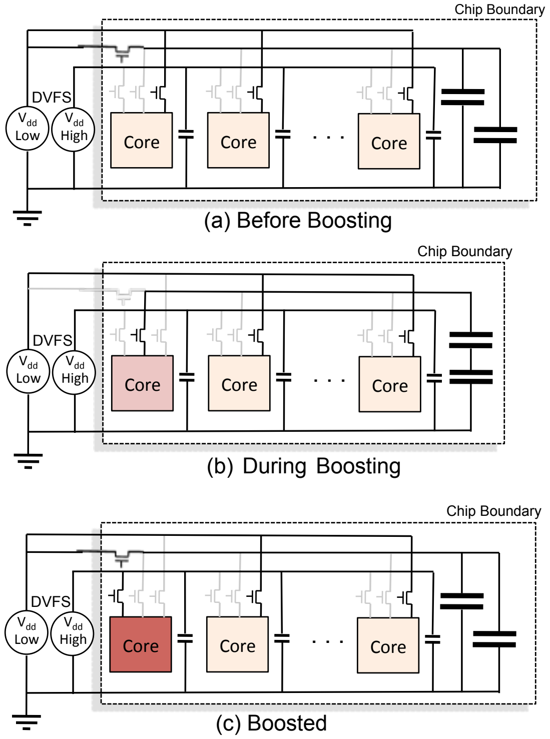
The proposed architectural contribution of Miller et al. may depend largely on the ability to perform rapid transitions between voltage rails (a sensitivity study was not presented by Miller et al., however Figure 1 reported by Dreslinski et al. indicates it may be). Their results assumed a transition time around 10ns. Given the 3x increase of the transition time that realistic simulations have demonstrated, the remaining conclusions of their paper are called into question. However, if a suitable alternative circuit can be designed that provided the desired transition time, then the rest of their paper's contributions will still be valid. In the next section we will present one such alternative that achieves the desired transition time.

3 Proposed Solution

To offer an alternative approach to overcome the problems shown in Section 2, we propose a new approach. Just like the original design, voltage boosting is done via dual external power supplies, as illustrated in Figure 7(a)-(c). In addition, there is an internal V_{boost} supply to aid in the transition of a core from the low to the high supply. Each core in the system is connected via multiple power gating transistors (shown as a single transistor in the diagram) to either the V_{ddHigh}, V_{ddLow}, or V_{boost} voltage rails. Decoupling capacitors (decaps) are placed between the high supply network and the ground node to reduce ripples on the node during transitions. In addition the V_{boost} supply has a set of reconfigurable decoupling capacitors to aid in transitioning the core quickly.

The operation of the proposed boosting scheme is as follows. In normal operation all the cores are initially connected to V_{ddLow}, as is the V_{boost} supply, shown in Figure 7(a). In addition, the decaps connected to the V_{boost} supply are in their parallel configuration and hence both charged to V_{ddLow}. To boost performance, a core is first switched over to the special V_{boost} supply while at the same time, the boosting network is disconnected from the V_{ddLow} supply and its decaps are changed to their series configuration, shown in Figure 7(b). By changing their configuration from parallel to series, the voltage of the V_{boost} supply is effectively doubled instantaneously (to *2× V_{ddLow})*, which causes it to rapidly charge up the voltage of the transitioning core. Once the core approaches the high supply voltage, the transitioning core is switched from the V_{boost} supply to the V_{ddHigh} supply completing the voltage transition, shown in Figure 7(c). After the transitioning core is disconnected from the boosting network, the V_{boost} supply is reconnected to the V_{ddLow} supply and the decaps are again placed in the parallel configuration. The decaps will re-charge drawing significant current from the V_{ddLow} supply network. However, the supply droop on this network is minimal because there are a large number of cores connected to the V_{ddLow} supply network providing large amounts of parasitic and explicit decap. Also, the recharging can be slowed down to further reduce the droop on the low power supply if necessary. After the boosting network decaps are recharged, the system is ready to transition the next core. This technique requires that no more than one core can be transitioning at any point in time.

The proposed boosting approach has several advantages. First, since the boosting decaps are on chip, they can act quickly and, through charge re-distribution, provide for a rapid transition. To analyze the speed of transition, we carry out SPICE simulations on the schematic shown in Figure 6 for a 31 core machine. The simulation results appear in Figure 8, which shows that transition from low to high can be accomplished in ~9ns, which corresponds to 9 clock cycles at 1GHz operation. Second, since the boosting decaps are shared by all the processors, their area overhead is amortized over all the cores. In addition, while the boosting network does require the distribution of a third supply rail (V_{boost}), this rail does not need to have a high level of signal integrity, meaning it can be more sparse. We find that the overall overhead of adding a boosting rail with reconfigurable decaps is 11%.



(a) Before Boosting

(b) During Boosting

(c) Boosted

Figure 7: Dual-V$_{dd}$ chip configurations. (a) shows the cores in normal operation, where all cores are connected to the low voltage network and the boosting cap network are placed in parallel. (b) shows the cores in boost transition. In this phase the core boosting is connected to the output of the boosting cap network and the boosting capacitors are connected in series. (c) shows the system once the transition stabilizes. Here the boosting cap network returns to parallel, and the boosted core runs off the external high voltage. DVFS can be used on external power supplies to adjust the degree of boosting over longer time frames. Area overhead of decap, power transistors, and extra supply rails is ~5-10%.

When considering advanced technologies, such as deep trench capacitors [7], this overhead can be reduced to less than 5%. Third, since the boosting network brings the voltage of the transitioning core to nearly V_{ddHigh}, the voltage droop on V_{ddHigh} is not nearly so large as when no boosting network is used. Extra decaps on the high supply further suppress the droop to a level that is acceptable.

4 Conclusion

We showed that previous dual-voltage designs [2, 6, 5] inaccurately model boosting circuitry. These previous studies may rely on boosting transition times around 10ns for the architectural design to achieve the published results. However, we showed that the incorrect core models, the

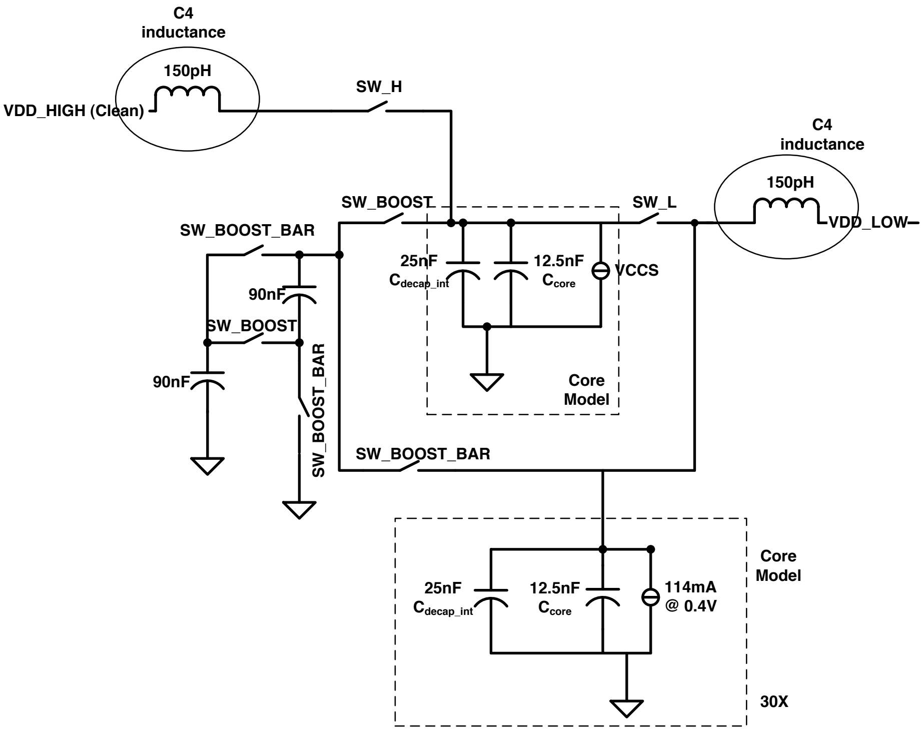


Figure 6: Circuit design for the proposed voltage boosting circuit.

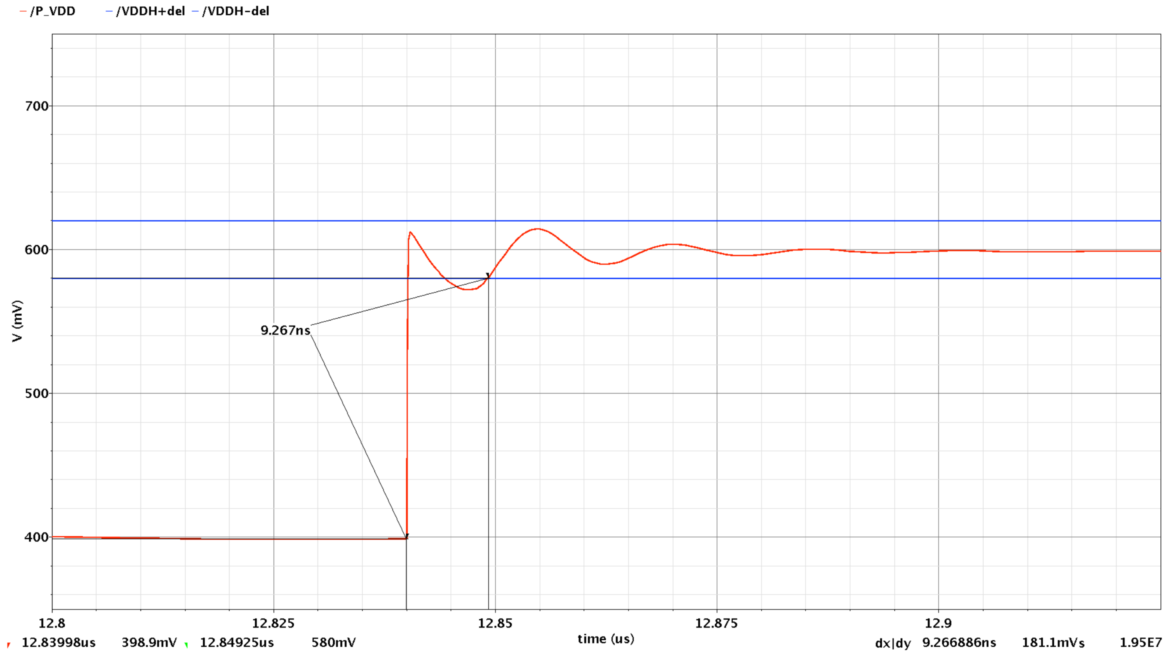


Figure 8: Boost transition. When boosting occurs, the boosting capacitors are arranged in series, increasing the output voltage of the boosting cap network. The core being transitioned jumps first to the boosting cap network supply and, once stable, finally transitions to the high supply voltage. The total transition, simulated via SPICE, takes ∼9ns to stabilize.

lack of supply rail inductance, and the non-worst case simulation in these previous studies led to boosting times that were up to 3x incorrect. When modeling all these components properly the transition times were in excess of 30ns. We then proposed a new dual-rail design that added a third internal power rail and additional capacitors to isolate the supply-rails durring boost transitions. The new design achieved transition times of around 9ns. Ultimately the original architectural contributions of the previous studies are still attainable, but require a slightly more complicated boosting circuit.

References

[1] R. Dreslinski, M. Wieckowski, D. Blaauw, D. Sylvester, and T. Mudge. Near-threshold computing: Reclaiming moore's law through energy efficient integrated circuits. *Proceedings of the IEEE*, 98(2):253 –266, February 2010.

[2] R. G. Dreslinski. Near threshold computing: From single core to many-core energy efficient architectures. *PhD thesis, The University of Michigan*, 2011.

[3] J. Leverich, M. Monchiero, V. Talwar, P. Ranganathan, and C. Kozyrakis. Power management of datacenter workloads using per-core power gating. *IEEE Comput. Archit. Lett.*, 8(2):48–51, July 2009.

[4] Luiz André Barroso and Urs Hölzle. *ThSe Datacenter as a Computer*. Morgan Claypool, 2009.

[5] T. Miller, X. Pan, R. Thomas, N. Sedaghati, and R. Teodorescu. Booster: Reactive core acceleration for mitigating the effects of process variation and application imbalance in low-voltage chips. In *High Performance Computer Architecture (HPCA), 2012 IEEE 18th International Symposium on*, pages 1 –12, feb. 2012.

[6] T. Miller, R. Thomas, and R. Teodorescu. Mitigating the effects of process variation in ultra-low voltage chip multiprocessors using dual supply voltages and half-speed units. *Computer Architecture Letters*, PP(99):1, 2011.

[7] G. Wang, D. Anand, N. Butt, A. Cestero, M. Chudzik, J. Ervin, S. Fang, G. Freeman, H. Ho, B. Khan, B. Kim, W. Kong, R. Krishnan, S. Krishnan, O. Kwon, J. Liu, K. McStay, E. Nelson, K. Nummy, P. Parries, J. Sim, R. Takalkar, A. Tessier, R. Todi, R. Malik, S. Stiffler, and S. Iyer. Scaling deep trench based edram on soi to 32nm and beyond. In *Electron Devices Meeting (IEDM), 2009 IEEE International*, pages 1 –4, dec. 2009.

[8] N. Weste and D. Harris. *CMOS VLSI Design: A Circuits and Systems Perspective*. Addison Wesley, 2010.

[9] B. Zhai, R. G. Dreslinski, D. Blaauw, T. Mudge, and D. Sylvester. Energy efficient near-threshold chip multi-processing. In *ISLPED '07: Proceedings of the 2007 international symposium on Low power electronics and design*, pages 32–37, New York, NY, USA, 2007. ACM.