T_i-states: Processor Power Management in the Temperature Inversion Region

Yazhou Zu[†] Wei Huang[*] Indrani Paul[*] Vijay Janapa Reddi[†]

[†]*The University of Texas at Austin*
yazhou.zu@utexas.edu, vj@ece.utexas.edu

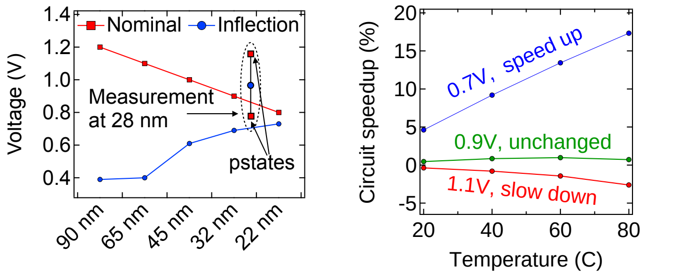
[*]*Advanced Micro Devices, Inc.*
{wein.huang, indrani.paul}@amd.com

Abstract*—Temperature inversion is a transistor-level effect that can improve performance when temperature increases. It has largely been ignored in the past because it does not occur in the typical operating region of a processor, but temperature inversion is becoming increasing important in current and future technologies. In this paper, we study temperature inversion's implications on architecture design, and power and performance management. We present the first public comprehensive measurement-based analysis on the effects of temperature inversion on a real processor, using the AMD A10-8700P processor as our system under test. We show that the extra timing margin introduced by temperature inversion can provide more than 5% V_{dd} reduction benefit, and this improvement increases to more than 8% when operating in the near-threshold, low-voltage region. To harness this opportunity, we present T_i-states, a power management technique that sets the processor's voltage based on real-time silicon temperature to improve power efficiency. T_i-states lead to 6% to 12% measured power saving across a range of different temperatures compared to a fixed margin. As technology scales to FD-SOI and FinFET, we show there is an ideal operating temperature for various workloads to maximize the benefits of temperature inversion. The key is to counterbalance leakage power increase at higher temperatures with dynamic power reduction by the T_i-states. The projected optimal temperature is typically around 60°C and yields 8% to 9% chip power saving. The optimal high-temperature can be exploited to reduce design cost and runtime operating power for overall cooling. Our findings are important for power and thermal management in future chips and process technologies.

***Keywords*-timing margin; temperature inversion; power management; reliability; technology scaling**

I. Introduction

Temperature inversion refers to the phenomenon that for certain voltage regions transistors speed up and operate faster at a higher temperature. When the temperature increases, transistor performance is affected by two fundamental factors: carrier mobility decrease and threshold voltage reduction. Carrier mobility decrease causes devices to slow down while threshold voltage reduction causes the devices to speedup. Temperature inversion happens in the region where the supply voltage is low enough to make the second factor (i.e., threshold voltage reduction) dominate. Otherwise, the devices slow down at the higher temperature, degrading performance.



(a) Temperature inversion's inflection voltage approaches nominal supply.

(b) Under low voltage, temperature inversion increases circuit performance.

Figure 1: Temperature inversion is having more impact on processor performance as technology scales.

In the past, temperature inversion has been safely discounted by processor designers because the nominal supply voltage at which this effect starts to occur is too low in prior technologies. At 250 nm, when temperature inversion was first discovered, the *inflection voltage* was more than 1.5 V lower than the nominal supply voltage [1], [2], [3]. With such a wide margin of separation, temperature inversion does not interfere with the processor's normal operating voltage region.

However, with technology scaling, today's processors are operating close to the temperature inversion's voltage region. Thus, the impact of this effect can no longer be safely discounted. Fig. 1a shows a detailed device analysis based on predictive technology models [4], [5]. As technology scales down from 90 nm to 22 nm, the inflection voltage increases with smaller feature sizes. At the 32 nm node, the inflection voltage is predicted to closer to the nominal supply voltage. Scaling into future FinFET and FD-SOI devices with smaller feature sizes, it is likely that temperature inversion will occur for all of a processor's operating voltage range [6], [7].

Silicon measurements performed on the AMD® A10-8700P processor confirm this behavior in practice. At the 28 nm node, the inflection voltage in Fig. 1a falls within the range of the processor's different P-states. The integrated GPU's highest P-state is only slightly above the inflection point. Fig. 1b shows the measured temperature inversion effect on circuit performance on the A10-8700P processor [8] with respect to a 0°C baseline. At 1.1 V, as temperature increases, circuit performance becomes slightly slower at 80°C, as expected from conventional wisdom. The measured inflection voltage for temperature inversion to take effect is 0.9 V, at which

978-1-5090-3508-3/16/$31.00 © 2016 IEEE

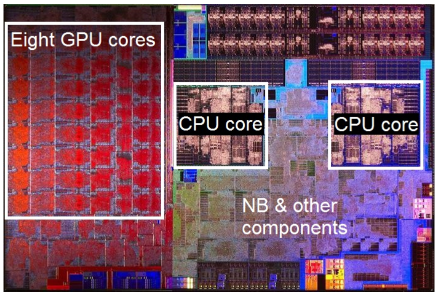


Figure 2: Die photo of the A10-8700P SoC.

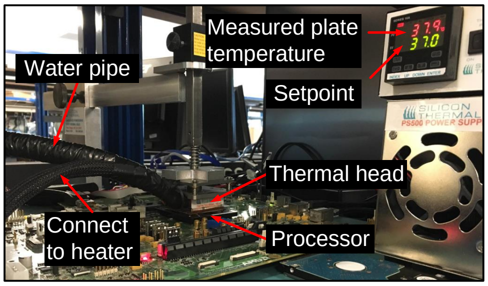


Figure 3: Temperature control setup.

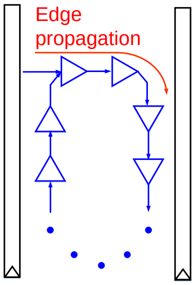


Figure 4: PSM logic.

exact point circuit speed remains almost constant at all product specified temperatures. At 0.7 V, however, circuit becomes faster by more than 15% at 80°C.

In this paper, we present the first public comprehensive characterization of temperature inversion's effects at the architecture level. For measurements, we use a 28 nm low-power A10-8700P processor with a CPU and GPU. We show that temperature inversion's performance benefit amplifies at lower voltages on both core types though we focus largely on the GPU because GPU's naturally lend themselves towards low-voltage, throughput-oriented computing. We also show how temperature inversion manifests under an idle system at different temperatures, as well as under a fully loaded system, operating with workloads that have different profiles.

In addition, we show how to exploit temperature inversion's benefit with an adaptive voltage margin [9]. Unlike the static design-time margin to tolerate worst case variations, an adaptive voltage margin provides just enough margin by constantly adjusting supply voltage based on runtime temperature information, and can thus save a considerable amount of power. To enable the adaptive margining, we introduce *T_i-states*—temperature inversion states that correspond to frequency and voltage pairs—that expose temperature inversion's speedup benefit. T_i-state exploits the extra timing margin gained as a result of temperature inversion to reduce voltage and save power. Alternatively, they could be used to boost performance. We present a systematic workflow to identify the T_i-state, as well an end-to-end full system architecture to make use of the T_i-state for adaptive margining.

Our work also has implications beyond the limits of our platform (28 nm bulk CMOS process at 0.7 V). We present scaling analysis for FinFET and FD-SOI technologies and demonstrate that there are larger, more workload-dependent gains to be achieved. In these technologies, dynamic power consumption strongly dominates static leakage power consumption. And because workloads can have different dynamic to static power consumption ratios, the "optimal" temperature at which to run the workload to maximize the benefits of temperature inversion will vary. We show how to leverage this unique behavior to reduce power consumption by as much as 12% with zero performance impact.

In summary, we make the following contributions:

- We quantify the effect of temperature inversion on processor performance using a comprehensive set of on-chip sensor measurements.
- We propose T_i-states, and adaptive margin control that can provide up to 6% V_{dd} reduction by exploiting the temperature inversion effect.
- We analyze the changes such an adaptive margin can bring about for temperature and power management in future FinFET and FD-SOI technologies.

The remainder of the paper is structured as follows: Sec. II explains the experimental setup. Sec. III characterizes the relationship between temperature and timing margin. Sec. IV proposes and evaluates temperature-aware adaptive margining using T_i-state. Sec. V analyzes T_i-state's gain on FinFET and FD-SOI technologies. Sec. VI discusses our work's implication on power management for future systems. Sec. VII addresses prior art and Sec. VIII concludes the paper.

II. Experimental Framework

In this section, we first introduce the AMD® processor under study (Sec. II-A). Since we study temperature inversion under different operating temperature conditions, we explain our temperature control setup (Sec. II-B). Finally, we describe the Power Supply Monitor (PSM) logic (Sec. II-C), which is an on-chip timing sensor that we use extensively to measure, characterize and quantify the temperature-inversion behavior.

A. AMD® A10-8700P Accelerated Processing Unit

The AMD® A10-8700P Accelerated Processing Unit (APU) is a state-of-the-art SoC manufactured in 28 nm HKMG planar bulk technology. It integrates two CPU core-pairs, eight GPU cores, and other components as shown in Fig. 2. Each CPU core-pair contains two out-of-order cores that share the front-end and floating point units. Each GPU core includes four 16-lane wide single instruction multiple data (SIMD) units.

We conducted temperature inversion studies on both the CPU and GPU. A separate power delivery network allows us to control the CPU and GPU voltage independently. But in this work, we present the results for the GPU only because the GPU's throughput-oriented architecture allows low-voltage region operation with meaningful and realistic performance. However, because the temperature inversion effect we study depends solely on the supply voltage, and not necessarily the underlying architecture, the analysis and benefits we present on the GPU naturally do extend to the CPU as well.

The GPU clock is set at 300 MHz in the voltage region we explore around 0.7 V. We pick 300MHz because its associated low voltage is within the temperature inversion region, and makes it possible to explore the potential impact of temperature inversion on future near-threshold technologies. The 300 MHz frequency corresponds to the GPU's lowest P-State, and in practice we have observed this P-State being exercised frequently during normal workload execution.

We use the ATITool [10] to set the GPU's voltage and frequency over a wide operating range. To measure power, we use a National Instrument's DAQ that reads the GPU's isolated supply voltage rail once every 10 ms.

B. Temperature Control

To characterize temperature inversion's effect on performance and power under different operating conditions, we have to carefully regulate the processor's on-die temperature. In our work, we generally sweep temperature range from 0°C to 80°C. This temperature range falls within the product's operating temperature range, and does not affect aging significantly.

Fig. 3 shows our temperature control setup. A thermal head is attached to the processor package. To stabilize the die temperature, which is measured via an on-chip thermal diode, at a user-specified target value, the thermal head's temperature is adjusted every 10 ms. Physically, the thermal head's temperature is controlled via a water pipe and a heater. The water pipe is connected to an external chiller to offer low temperatures while the heater increases temperature to reach the desired temperature setting. Under feedback control, we see a 2°C temperature variation on the diode in the worst-case. So, for instance, Fig. 3 shows the thermal head set its temperature to 37°C to let the die temperature stay at 40°C.

C. On-chip Power Supply Monitors (PSMs)

We use power supply monitors (PSMs) [11], [12] to accurately measure circuit speed changes in the chip under different temperature conditions. A PSM is a time-to-digital converter that reflects circuit time-delay or speed in numeric form. Originally designed as a voltage noise sensor, a PSM can sense minute circuit timing changes due to di/dt droops [11]. We use the PSM as a means to characterize circuit performance under temperature variation.

Fig. 4 shows the structure of a PSM. Its core component is a ring oscillator that counts the number of inverters an "edge" has traveled through in each clock cycle. When the circuit is faster (e.g., under smaller di/dt effects or stronger temperature inversion), an edge can pass more inverters and PSM will produce a higher count output. A supporting module logs ring oscillator's per-cycle output and accumulates the minimum, maximum, and average values over a time.

The A10-8700P processor has ten PSMs in each CPU core-pair and two PSMs in each GPU core, distributed across the cores to account for process variation and spatial differences in di/dt effect. Through measurements we determined that the changes in the different PSM readings under different temperatures are nearly identical, thus we only show the result of one representative PSM in GPU. The results are representative of using other or more than one PSM.

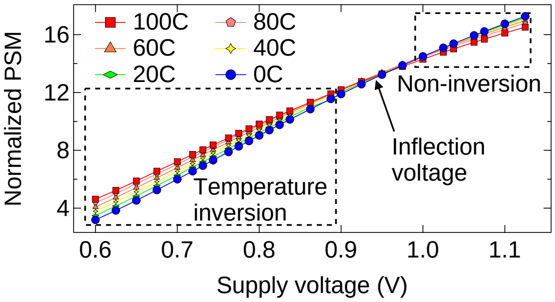


Figure 5: Temperature inversion happens below 0.9 V and is progressively stronger when voltage scales down.

For reasons that prevent us from showing absolute values, we normalize the PSM reading to a reference value measured under 0.7 V, 300 MHz, 0°C, and idle chip condition. We log the minimum, maximum, and average output of all the PSMs.

III. Timing Margin Analysis

In this section, we first view PSM as a normal logic path to understand circuit performance under different temperature environment (Sec. III-A). Then, we use PSM as a timing sensor to reveal workload timing margin's behavior under different silicon temperatures and supply voltages (Sec. III-B). We make two key observations about temperature inversion: its speedup effect on circuits becomes stronger with lower voltage, and the speedup turns into workload extra timing margin independent of other factors like di/dt effects.

A. Circuit Speed at Chip Idle

The PSM by itself is a digital circuit located between the pipeline latches with other normal logic paths [9], and therefore its speed characteristics are representative of a pipeline's overall performance. For this reason, we use the PSM's output to quantify circuit performance across a wide range of different steady-state temperatures.

We keep the chip idle (i.e., the clock is still running) and read the PSM's "average" value to exclude the di/dt effect caused by workload dynamics. Fig. 5 shows the circuit speed under different supply voltages and die temperatures. Speed is reflected by the PSM's normalized output – higher value implies a faster circuit. At a higher supply voltage, the circuit switches faster, and the PSM can travel more inverters in a cycle which produces a higher count. The voltage-to-PSM relationship conforms to similar analysis as in [13].

We find that temperature's impact on circuit performance depends on the supply voltage. In the high supply voltage region around 1.1 V, the PSM's reading becomes progressively smaller as the temperature rises from 0°C to 100°C. The circuit is operating slower at a higher temperature, which aligns with conventional belief [14]. The reason for this circuit performance degradation is that the transistor's carrier mobility decreases at a higher temperature, leading to smaller switch-on current (I_{on}) and longer switch time [4].

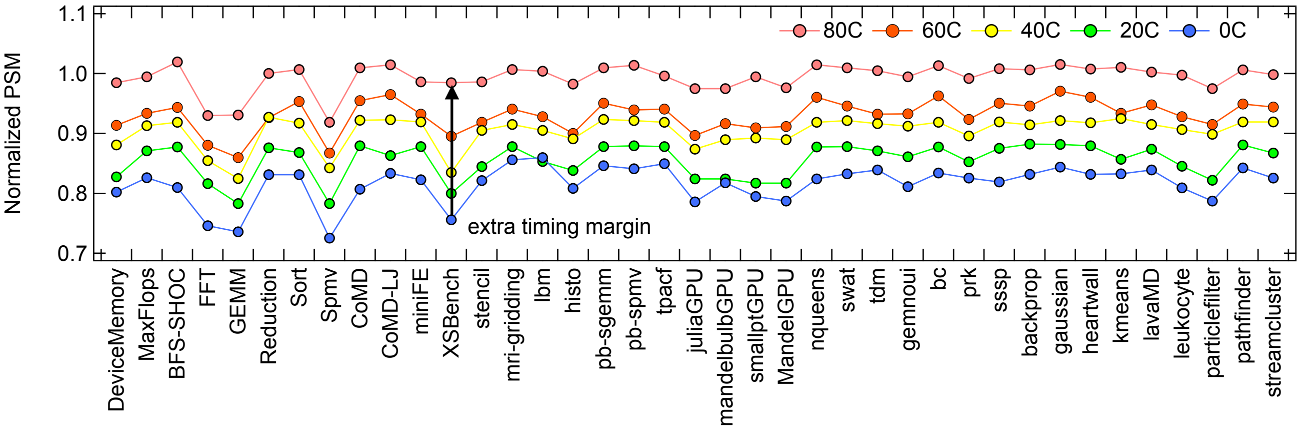


Figure 6: Temperature inversion's circuit speedup effect varies across a range of typical operating temperatures. Thus, it can provide the system with extra timing margin that can be exploited across a range of different workloads.

Under a lower supply voltage, the PSM's reading increase with higher temperature, which means the circuit switches faster (i.e., the temperature inversion phenomenon). During temperature inversion the transistor's threshold voltage (V_{th}) decreases linearly as temperature increases [4], [1], [3]. Thus, for the same supply voltage, a lower V_{th} provides more drive current (I_{on}) which makes the circuit switch faster. The speedup effect is more dominant when supply voltage is low because then the supply voltage is closer to V_{th}.

When the supply voltage is low enough, the speedup contribution from the reduced V_{th}, at some point, will balance out the carrier mobility slowdown. We call this voltage point the *inflection voltage*. The inflection voltage may change from chip to chip due to V_{th} variations, and it can be characterized during the binning process. In Fig. 5, we show that the tested processor's inflection voltage is between 0.9 V and 1 V. In this region, the temperature does not have a notable impact on circuit performance. Below the inflection voltage (0.95 V) is the temperature inversion region while above it is the non-inversion region. Half of the GPU's P-states, which range from 0.75 V to 1.1 V, operate in the temperature inversion region.

In Fig. 5's temperature inversion region, the speed change between any two temperatures increases when the supply voltage scales further away from the inflection point. As voltage scales into the lower voltage region around 0.6 V, the PSM reading varies by more than 40%, indicating the drastic speedup at a higher temperature. As voltage goes lower towards the near-threshold region, the overdrive voltage ($V_{dd} - V_{th}$) becomes small and it is very sensitive to small V_{th} changes. Thus, temperature inversion's V_{th} reduction has a more significant impact on device performance.

Hereon forward we use temperature inversion at 0.7 V as a case study to dive deeper and extra more insights at the architecture level, running workloads on a real system. Although we restrict ourselves to this single voltage, there is ample opportunity to demonstrate how temperature inversion may add new ingredients to overall system management. We anticipate that the opportunity and benefit we show will likely extend into future CMOS technologies when voltage scales towards near-threshold computing levels.

B. Workload Timing Margin

We extend our study to full workload analysis to understand how temperature inversion affects workloads with different execution characteristics. The study allows us to determine how much *timing slack* temperature inversion introduces at a low voltage and under different operating temperature conditions. In particular, we seek to understand how temperature inversion's speedup will affect the amount of timing margin, and whether this impact interferes with other factors such as di/dt droop as workloads are running. We select a set of workloads that shows a wide range of dynamic-to-leakage power ratio. As will be explained later, this ratio determines how much power savings we can leverage from the temperature inversion effect. We conduct our study at 0.7 V and we sweep a range of different temperatures using our temperature control experimental setup (Sec. II-B).

We use the PSM as a timing sensor to measure timing margin behavior under different temperature conditions and workload di/dt droop. Because circuit performance varies under temperature changes, di/dt and other system effects [15], [11], [9], chip vendors over-design pipeline cycle time with a margin to guarantee timing safety across all scenarios. Fig. 6 analyzes the workloads' timing margin under different operating temperatures. We measure the *worst-case PSM reading* during each workload run to account for the worst-case di/dt droop [13], [16]. The PSM readings are normalized to the condition when the chip is idle and at 0°C. Making measurements and observations relative to 0°C is a relatively standard industry-wide practice [17].

PSM's variations within each temperature setting in Fig. 6 reflect the di/dt droop caused by workload activity [18], [19], [14], [20]. The workload PSM reading at 0°C is less than 1.0 because di/dt causes circuit slowdown compared to a quiescent 0°C power supply network. We observe 4% to 8% voltage droop caused by worst-case di/dt droop, which is in line with previous measurement-based studies [13], [14], [16].

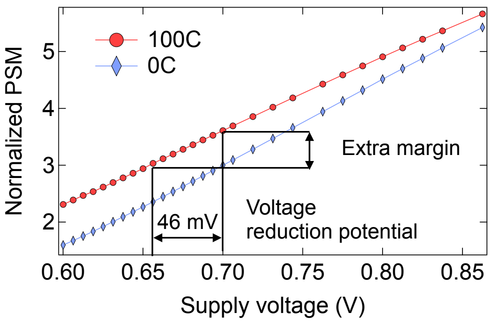


Figure 7: Estimating voltage reduction potential based on PSM characterization at different temperatures.

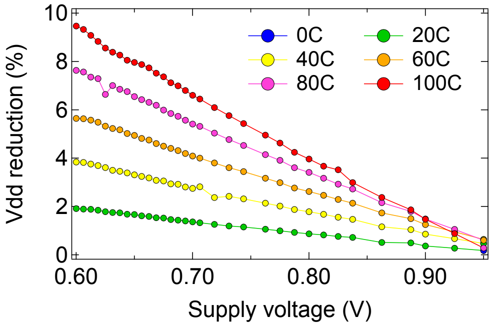


Figure 8: Voltage reduction potential is more pronounced in the near-threshold low voltage region.

Across different temperatures, the workload's di/dt effect is unaffected by the temperature. For instance, FFT, GEMM, Spmv, and XSBench have the strongest di/dt effect at all temperatures, whereas workloads such as Reduction, Sort, CoMD CoMD-LJ and Reduction have the weakest di/dt effect. The observation is intuitive because di/dt droop is the result of sudden current swings caused by microarchitecture activities [18], [19]. Changing temperature does not directly affect workload specific processor activities, whereas the leakage power change does not contribute to current variation.

We repeated the same experiment around the inflection voltage, where circuit speed is not affected by temperature variation. At this point, too, we found that the di/dt-induced circuit slowdown across different workloads does not change across different temperatures. Therefore, we come to the conclusion that temperature and di/dt impact on circuit speed are independent. Thus, temperature inversion can separately speedup circuit and offer extra timing margin as slack.

However, across the different temperatures, the workloads have more timing margin when the temperature progressively increases from 0°C to 80°C, as indicated by the higher PSM reading in Fig. 6. This behavior is caused by the extra timing margin provided by temperature inversion. At 80°C, the PSM almost reads the same as a quiet power delivery network at 0°C, meaning that at 80°C temperature inversion can completely offset the average 6% voltage loss from the di/dt effect, providing about 20% extra timing margin.

IV. Temperature Inversion States

We propose T_i-states for architecture-level power management to harness temperature inversion's performance speedup benefit. T_i-states reclaim the extra margin that is enabled by temperature inversion through voltage reduction. We first quantify the voltage reduction potential (Sec. IV-A). Next, we present a systematic method to accurately determine the T_i-state settings (Sec. IV-B), and then finally show that T_i-states can save a significant amount of power. T_i-states can save 6% power when the processor temperature is around 40°C and up to 12% when the die temperature is around 80°C (Sec. IV-C).

A. Voltage Reduction Potential

When undervolting, it is crucial that the system only reclaims the extra timing margin from temperature inversion and not the margin that is allocated for other effects, such as di/dt and loadline DC voltage loss [16], [14], [13]. Otherwise, pipeline timing may fail under some worst-case workloads, such as in the case of voltage stressmarks [21], [22].

We use the timing margin measured at 0.7 V and 0°C as the "golden" reference when reclaiming temperature inversion's extra margin. In other words, the voltage T_i-state sets as the nominal supply voltage should always make the timing margin measured by the PSM match the "golden" reference. Under this constraint, we can undervolt to maximize power saving.

We choose 0°C as the reference because under temperature inversion lower temperature degrades circuit performance. Even though 0°C rarely occurs in desktop, mobile, and datacenter applications, the timing margin still needs to be set to tolerate this worst-case condition. In the industry, 0°C or below is used as a standard circuit design guideline [17]. In certain scenarios, such as military use, an even more conservative reference of -25°C is considered [3].

To determine the correct amount of voltage reduction we can exploit because of the extra timing margin shown previously in Fig. 6, we first estimate the "approximate" potential using PSM characterization data in Fig. 5. Fig. 7 shows our estimation process. The x-axis zooms into the low voltage region between 0.6 V and 0.86 V in Fig. 5. The figure shows temperature inversion's performance benefit at 100°C over the 0°C baseline, and this benefit increases as the supply voltage decreases. For instance, in the illustrated example, the extra margin at 0.7 V translates to a 46 mV voltage reduction.

The PSM difference between the high-temperature 100°C line and the "golden reference" line at 0°C represents the extra timing margin in the units of inverter delays. In other words, it reflects how much faster the circuits can run at a higher temperature. To bring the faster circuit back to the original speed, supply voltage needs to reduce such that under a higher temperature the PSM will ideally read the same value.

We estimate the voltage reduction potential with linear extrapolation. Fig. 8 shows the estimated opportunity at different temperatures. As supply voltage scales down, the voltage reduction potential goes up almost linearly. Temperature inversion effect is stronger in the lower voltage regions, and hence the greater timing margin opportunity. At 0.6 V and 100°C, the extra timing margin provided

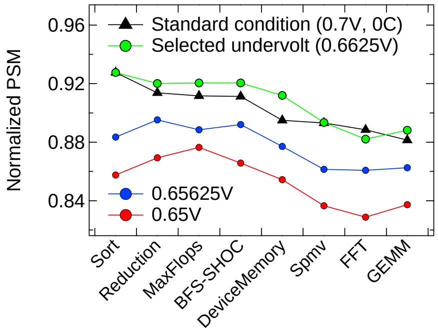


Figure 9: Exploring T$_i$-state at 80°C: measuring the "training" workloads' timing margin.

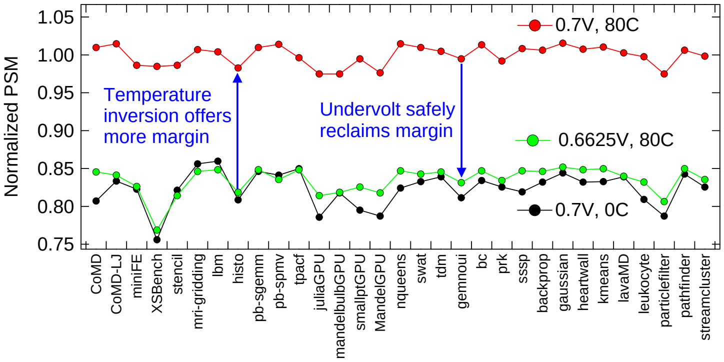


Figure 10: T$_i$-state undervolting decision at 80°C closely tracks the "golden" reference runs' timing margin, which is needed for reliability.

by temperature inversion can turn into almost 10% voltage reduction compared to 0°C. As a reference, 5% voltage reduction is considered significant in previous works [23]. At 0.7 V in our study, we can have 1.5% to 7% voltage reduction potential depending on the processor temperature.

B. T$_i$-state Table Construction

We require a systematic method to accurately establish the voltage to temperature relationship, as compared to Fig. 8 which provides us a way to estimate the potential for voltage reduction. We must determine which undervolting setting corresponds to a given temperature while ensuring that the processor has sufficient timing margin to tolerate the presence of other effects, such as workload-induced di/dt droop.

We propose a workload-centric methodology that constructs a set of temperature-voltage states in the inversion region (T$_i$-states) at test-time. A platform's voltage regulator module (VRM) sets supply voltage with increments of a small voltage step [24]. T$_i$-state accounts for this by choosing the lowest quantized voltage that provides enough margin for reliability.

We use a subset of workloads as the "training" set to first get a tentative temperature-voltage mapping. Then we validate this mapping with another set of "test" workloads to establish the final T$_i$-state. During training the T$_i$-state is constructed in a manner that is agnostic to workload-specific settings, so therefore we can be sure our voltage selection will provide enough margin for any workload that is run on the processor.

For each of the training workloads, we first measure their "golden" reference margin at 0°C. Then, at the temperature being characterized, we select four candidate voltages. These candidates voltage are picked such that they are around the extrapolated voltage value from Fig. 8. The candidates voltages are chosen such that they are two VRM steps above and two VRM steps below the extrapolated value.

Once we have the set of candidate voltages, we step through each candidate voltage and record the training workloads' timing margin using the PSM at every temperature that is being characterized. The timing margin measured at the candidate voltage is compared against the reference margin. Finally, we select the candidate voltage that has the minimum PSM difference from the golden reference.

It is worthwhile to note that on our particular chip the data variation for the 16 PSMs on our GPU is under 2%, so it makes little difference to use worst-case versus average. However, under severe intra-chip variation, transistors undervolting potential can differ significantly. In that case, worst-case PSMs values need to be used for comparison.

Algorithm 1 T$_i$-state Construction Methodology

```
procedure GET REFERENCE MARGIN
    set voltage and temperature to reference
    for each training workload do
        workloadMargin ← PSM measurement
        push RefMarginArr, workloadMargin
    return RefMarginArr
procedure EXPLORE UNDERVOLT
    initVdd ← idle PSM extrapolation
    candidateVddArr ← voltage around initVdd
    minErr ← MaxInt
    set exploration temperature
    for each Vdd in candidateVddArr do
        set voltage to Vdd
        for each training workload do
            workloadMargin ← PSM measurement
            push TrainMarginArr, workloadMargin
        err ← diff(RefMarginArr,TrainMarginArr)
        if err < minErr then
            minErr ← err
            exploreVdd ← Vdd
    return exploreVdd
```

Algorithm 1 summarizes our methodology. Fig. 9 shows an example at 80°C. At this temperature, Fig. 8's extrapolated voltage is 0.65625 V. The candidate voltages are 0.6625 V, 0.65625 V, and 0.65 V. Our platform's smallest VRM step is 6.25mV. The original four candidate voltage is capped by a lower hard limit of 0.65 V, and so we cannot set the voltage any lower. Algorithm 1 chooses 0.6625 V as the T$_i$-state voltage for 80°C because it has the closest timing margin compared to "golden" reference. Other candidate voltages with less timing margin run the risk of hampering the timing safety under potentially worst-case workloads.

	20°C	40°C	60°C	80°C	100°C
693.75mV	3.7%	-	-	-	-
687.50mV	**2.2%**	-	-	-	-
681.25mV	8.4%	**2.3%**	-	-	-
675.00mV	13.9%	5.3%	4.9%	-	-
668.75mV	-	9.5%	**2.5%**	-	-
662.50mV	-	13.5%	6.5%	**1.9%**	-
656.25mV	-	-	12.2%	5.6%	9.9%
650.00mV	-	-	-	9.3%	**5.1%**

Table I: PSM error compared to the reference setting for different $< temperature, voltage >$ configurations.

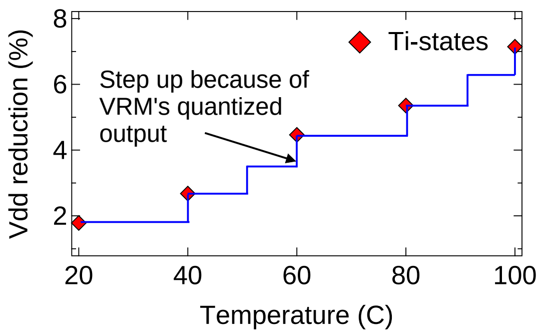


Figure 11: V_{dd} reduction due to T$_i$-states. The line corresponds to the VRM's quantized output values.

Fig. 10 verifies Algorithm 1's T$_i$-state selection at 80°C. At 0.7 V, going from 0°C to 80°C offers more than 15% extra timing margin. After voltage reduction, the workload timing margins closely track the golden reference with some workloads showing slightly higher margin.

Fig. 10 proves yet another important point. It shows that the voltage explored using a small set of training workloads can be safely applied to future unknown workloads. The reason that the approach we present works in practice is because the extra margin that arises from temperature inversion is mainly a device property and it is workload-independent.

Algorithm 1 will repeat the same process at different temperatures. Using results similar to Fig. 9 and Fig. 10, our methodology will eventually construct a temperate-voltage pairing table with all the proper T$_i$-states. Table I shows the measured results on our A10-8700P processor for 20°C, 40°C, 60°C and 80°C. For each temperature, there is one voltage that has the smallest deviation from the "golden" reference margin, as highlighted and bolded in the table. These points are selected as the final T$_i$-states for the processor to use.

T$_i$-state table construction would add little overhead to existing silicon test procedures. Per-bin or even per-part characterization is already an industry-standard practice, especially for the high-end server market sector. Therefore, we believe that T$_i$-state table construction is a practical approach.

As we will describe in greater detail later in Sec. V-D, a power management scheme can use runtime temperature sensor data to index into a T$_i$-state table and determine a suitable supply voltage, similar to prior schemes [9]. In our work and the restricted scope of this paper, T$_i$-states are constructed for the GPU clock frequency of 300 MHz. In practice, however, the T$_i$-state table can be constructed across different frequencies, and the power management unit can index into the right table by frequency during runtime.

C. Voltage and Power Reduction Benefits

We use a representative subset of all workloads to evaluate T$_i$-state's power reduction at different temperatures. We start with Fig. 11, which shows the V_{dd} reduction at various T$_i$-states. One temperature range corresponds to one voltage and this is because of the VRM's quantized output. To make the VRM reduce voltage by one step, the temperature has to be high enough to speed up the circuit beyond the current point. Between 20°C and 40°C, the VRM can reduce V_{dd} by exactly one step, yet from 40°C to 60°C there are two VRM steps in between. The results show that V_{dd} reduction is larger at a higher temperature because the extra timing margin offered by temperature inversion is larger than at a lower temperature.

In Fig. 12 we compare the average power savings of the various GPU workloads as a result of the V_{dd} reduction at different temperatures. We set the die temperature manually using our temperature control setup (Sec. II-B) to 40°C, 60°C, and 80°C to mimic the various temperature conditions that the processor typically faces. We manually set the temperature because the GPU on the A10-8700P does not heat up the chip often in the voltage region we study, which limits the temperature range we can use to thoroughly characterize. Therefore, rather than examine the workloads under a "free run," we interject with external temperature control. But on the more high-end and power-hungry server parts, the GPU would hit the higher temperatures we are characterizing.

An added benefit of temperature control is that it facilitates controlled and repeatable experiments. Our choice of temperatures is reasonable because, usually, for a high-end cooling system that has around 0.2°C/W ambient-silicon thermal resistance, a workload consuming 60 W will have a steady state temperature of 40°C. For a less capable 0.5°C/W cooling system the same workload will stabilize around 60°C [25], [26], [27]. So we cover different cooling options.

Fig. 12 shows that on average the T$_i$-states can save 6.2%, 9.5%, 12.2% power at 40°C, 60°C and 80°C, respectively. The power saving primarily comes from dynamic power reduction. Leakage power consumption also reduces at lower voltages, but only by a little. At each temperature, the relative power saving does not vary much between different workloads, but this is to be expected because T$_i$-state's voltage reduction is workload independent. Hence, the relative dynamic power saving for each workload should stay the same for each temperature. In practice, different workloads stabilize at different temperatures at runtime, and T$_i$-state will reduce the operating voltage accordingly. When the temperature varies under workload phase changes, a VRM can index into T$_i$-state table in real-time and adjust the supply voltage step by step [9].

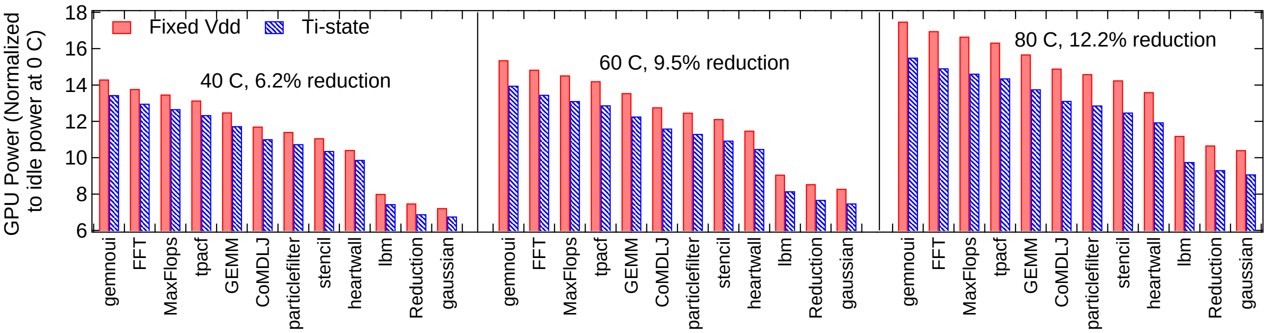


Figure 12: Power saving increases at higher temperatures. We mimic workload temperature by externally controlling die temperature to a 40°C – 80°C range. T_i-state's power reduction is independent of the workload activity.

V. Ti-State Temperature Management in FinFET and FD-SOI Technologies

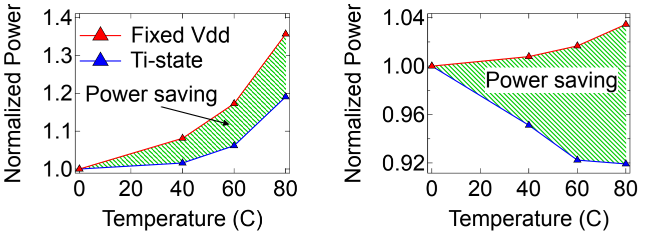
Thus far, we have shown the power savings from T_i-state as a result of voltage reduction with temperature statically set by the thermal head. High temperature increases leakage power exponentially, especially in planar bulk CMOS technology, which is against the dynamic power savings from T_i-state with voltage reduction at high temperature. These two opposite trends form a trade-off: an optimal temperature may exist where T_i-state's dynamic power reduction balances leakage power increase at higher temperatures and the overall processor power is minimized. Carefully evaluating this trade-off is crucial for T_i-state to be practical in runtime processor temperature and power management control.

In this section, we compare and contrast the benefits of T_i-state's power savings on planar bulk CMOS versus emerging FinFET and FD-SOI process technologies (Sec. V-A). FinFET is already present in latest processors [28], [29], and both technologies will be more broadly adopted in the coming years [30], [31], [32], [33]. Because we do not have access to a FinFET or FD-SOI processor to continue our measurement-based study, we scale our measurement results to these technologies. We explain our scaling approach for FinFET and FD-SOI (Sec. V-B), then we detail a careful analysis of T_i-states in these technologies to show that the trade-off described above exists and that it can be workload dependent (Sec. V-C). The effect for FD-SOI works similarly. Finally, we discuss a runtime power management control loop to minimize power consumption by leveraging the optimal temperature(Sec. V-D).

A. Planar Bulk CMOS versus FinFET and FD-SOI

Our measurement data in Fig. 12 shows that the total power is always greater at high temperatures despite T_i-state's power reduction capability within a given temperature. The normalized power at 40°C is less than 60°C, which in turn is less than the power consumption at 80°C. This is because in planar bulk CMOS technology the processor's leakage power increase is more significant than T_i-state's dynamic power saving as the temperature increases.

Fig. 13a examines the effect deeper using the benchmark GEMM. With a fixed nominal supply voltage, the GPU's power consumption increases by around 35% going from 0°C to 80°C because the leakage power increases exponentially. T_i-state lowers the power consumption by reducing the nominal supply voltage, and its power saving increases from 6% to 10% as we sweep the temperature from 40°C to 80°C (i.e., the gap between the two curves). However, T_i-state's absolute power consumption is still much higher because the amount of dynamic power reduction cannot offset the leakage power increase. That is, at 28 nm planar bulk CMOS technology, the optimal temperature that minimizes total power is still towards the low-temperature side of the curves.



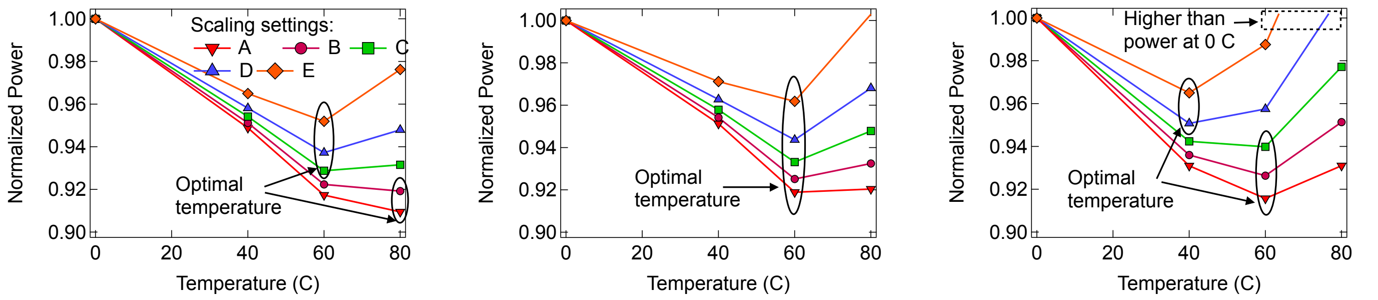
(a) Measured at 28 nm planar CMOS technology.

(b) Extrapolated to FinFET device technology.

Figure 13: Planar device's leakage power increases exponentially and overrides T_i-state's dynamic power saving, while FinFET's leakage is smaller, and therefore T_i-states consume less total power at higher temperatures.

The scenario in Fig. 13a will fundamentally change with the wide adoption of FinFET and FD-SOI technology. FinFET has better control at the transistor gate, and promises approximately 10× leakage power reduction while maintaining the same device speed [34], [35], [30], [31], [32]. FD-SOI effectively mitigates transistor's short channel effect with buried oxide, and can reduce leakage even more significantly [36], [37], [38], [39], [33].

Fig. 13b depicts T_i-state's power-temperature relationship with FinFET. FD-SOI processors follow the same trend. Under a fixed V_{dd} the total power only increases by 4% because of



(a) Benchmark FFT (b) Benchmark particlefilter (c) Benchmark Reduction

Figure 14: Power versus temperature under different scaling factors for different workloads. In FinFET and FD-SOI, T_i-state makes GPU power smaller at high temperature. The optimal temperature is different for the workloads and the different scaling settings, and this is because the ratio of static to dynamic power across the workloads varies.

FinFET's flat leakage-temperature profiles. In this case, T_i-state's dynamic power reduction can fully offset the leakage power increase at high temperatures. The optimal temperature that minimizes overall power shifts to high ranges, which means it is possible to dynamically adjust the cooling system to reach a higher temperature and optimize the total power.

B. *Scaling to FinFET and FD-SOI*

FinFET and FD-SOI technologies promise the opportunity of a temperature sweet spot for T_i-state-based power management. To understand and characterize this effect, we focus on analyzing T_i-state's power saving behavior under these technologies. Since we do not have access to a commercial FinFET or FD-SOI processors yet, in our analysis, we scale our measurements from a 28 nm planar bulk CMOS processor with different yet reasonable scaling options.

FinFET and FD-SOI have significantly different dynamic-to-leakage power ratios than traditional planar bulk CMOS. Here, we set up five reasonable scaling scenarios (ranging from aggressive to conservative leakage reductions) based on lessons from a 14 nm FinFET NTC prototype chip [40] as well as prior report [38]. Compared to 28 nm planar bulk CMOS, FinFET can reduce the off-current (I_{off}) by more than 10× under the same supply voltage for all device types, and FD-SOI can achieve even more leakage reduction. We mimic this scenario as setting *B* in Table II. Furthermore, the FinFET test chip runs at 650 MHz at 0.55 V [40], over 2× of the 300 MHz frequency we study at 0.7 V. In setting *A*, we scale dynamic power by 1.5 to simulate possible dynamic power changes.

Setting *C*, *D*, and *E* account for possible FinFET threshold voltage engineering by modestly scaling leakage power by 0.2. Setting *C* mimics a performance-centric scenario where lower threshold is utilized for higher frequency. We include setting *E* as a conservative scenario where dynamic power reduces with lower supply voltage. Overall, scaling setting *A* is an aggressive projection for FinFET, but it is a good example of FD-SOI's application scenario. Setting *B* reflects FinFET and FD-SOI's leakage power reduction capability, while settings *C* and *D* represent FinFET's more realistic usage cases.

Temperature inversion will exist in FinFET and FD-SOI. Prior work concludes FinFET processors will entirely work in temperature inversion range [6], [7], and its inflection voltage will be around the same as we measure in 28 nm [6]. Therefore, we assume the same T_i-state's voltage and power reduction capability within these technologies.

C. *T_i-state Power Analysis under FinFET and FD-SOI*

We examine power benefits for three different types of workloads that are representative of different typical dynamic-to-leakage power ratios. The workloads include FFT, particlefilter and Reduction, going from high to low dynamic power consumption. Fig. 14 shows T_i-state's GPU power under different scaling settings. Power is normalized to 0°C to show how power scales as temperature increases.

Fig. 14a shows that when the dynamic power is more dominant in settings *A* and *B* then FFT prefers to stay at 80°C. Under more conservative settings where leakage power is higher, the temperature sweet spot drops to 60°C. In these scaling settings, FinFET's leakage power increase beyond 60°C is more than T_i-state's dynamic power reduction.

For medium dynamic power, Fig. 14b shows that particlefilter's temperature sweet spot is around 60°C for the scaling ratios. Particlefilter's dynamic power is not high enough to make T_i-state's power saving override leakage power at 80°C.

Scaling setting	Leakage power	Dynamic power	Dynamic-leakage Power scale ratio
A	0.1	1.5	15 (aggressive)
B	0.1	1	10 (test-chip [40])
C	0.2	1.5	7.5 (modest)
D	0.2	1	5 (modest)
E	0.2	0.6	3 (conservative)

Table II: FinFET and FD-SOI scaling settings: for completeness, we scale dynamic and leakage power with different factors to cover both aggressive and conservative scenarios.

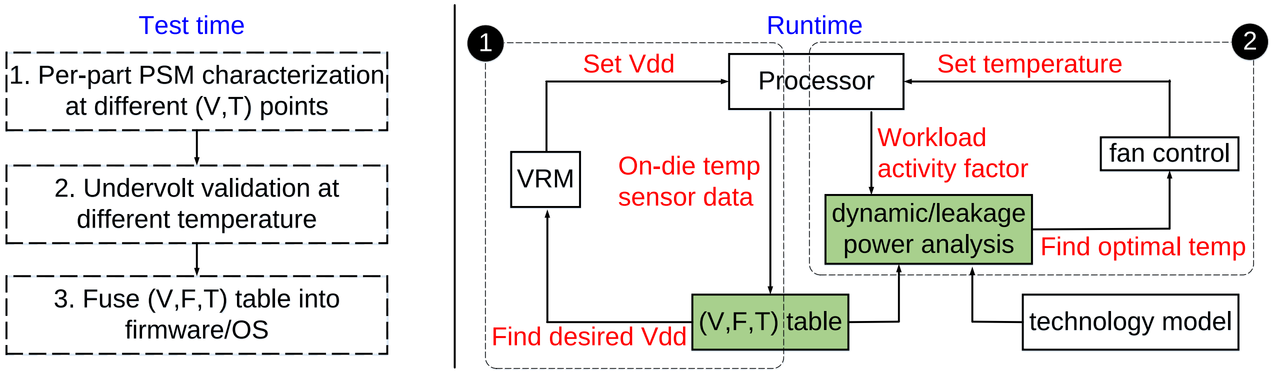

Test time
1. Per-part PSM characterization at different (V,T) points
2. Undervolt validation at different temperature
3. Fuse (V,F,T) table into firmware/OS
Runtime
1
2
Set Vdd
Processor
Set temperature
VRM
On-die temp sensor data
Workload activity factor
fan control
dynamic/leakage power analysis
Find optimal temp
Find desired Vdd
(V,F,T) table
technology model

Figure 15: T$_i$-state temperature and voltage control: two loops work in synergy to minimize power. Loop 1 is a fast control loop that uses T$_i$-state table to keep adjusting voltage in response to silicon temperature variation. Loop 2 is a slow control loop that sets the optimal temperature based on workload steady-state dynamic power profile.

In contrast to FFT and particlefilter, the workload Reduction does not consume much dynamic power. Fig. 14c shows that it prefers to stay at a lower temperature to minimize leakage power. Its dynamic power occupies a smaller portion of total power, therefore T$_i$-state's power reduction has a lesser effect. In the optimistic scaling settings *A* and *B*, Reduction's sweet spot temperature is 60°C, whereas in conservative settings *D* and *E*, the optimal temperature is at 40°C to avoid the exponential leakage power at a higher temperature.

In general, Fig. 14 shows that when leakage power is less prominent (i.e., leakage scaling is more aggressive in Table II), T$_i$-states have higher power saving and the optimal temperature is also higher. With smaller leakage, dynamic power occupies a larger portion of the total power, which is when T$_i$-state's improvement has a bigger power saving impact. In the extreme assumption where leakage power is completely agnostic of temperature, T$_i$-state would prefer to operate at the highest allowed temperature to maximize the magnitude of voltage reduction from temperature inversion.

We also find when the optimal temperature is higher, the corresponding optimal power tends to be lower as well. T$_i$-state's power saving capability increases with higher temperature. When a workload has a larger share of dynamic power and prefers to run under a higher temperature, T$_i$-state's higher power saving manifests as total power improvement.

Another observation that we can make from Fig. 14 is that high-power workloads typically have higher temperature sweet spots. For such workloads, the dynamic power is more dominant than the leakage power. Therefore, in such scenarios, for a given temperature, the percentage of dynamic power saving from T$_i$-state contributes more to the bottom-line.

D. Runtime Temperature Control

We notice that different temperature sweet spots under all workloads and scaling scenarios are essentially a result of processor's dynamic-to-leakage power ratio. To leverage this fact, we propose a set of temperature and voltage control algorithms in Fig. 15 to steer future FinFET and FD-SOI processors for maximum power efficiency. The solution consists of two stages: test-time and runtime.

At test time, the methodology described in Algorithm 1 establishes T$_i$-state's temperature-voltage tables. The process starts with characterizing the circuit speed behavior with on-chip timing sensors like the PSM, which are subsequently verified by workload timing margin measurements as we described earlier. The final temperature-voltage table can be fused into firmware for runtime lookup. For each chip, we envision less than 40 entries to be added in total. Constructing such tables is already in practice [9]. It only extends the existing test flow by a few steps, and adds minimal overhead.

At runtime, two loops work in synergy. Loop 1 is a fast loop that addresses quick yet small temperature variations from workload phase changes. It measures silicon temperature and index into T$_i$-state table in real time to get and set the desired voltage, similar to a typical DVFS table lookup. We envision this loop to occur at millisecond-level granularity, as in with other systems [41]. Loop 2 is a slow control loop that monitors the workload's average activity factor over a longer time period to estimate its dynamic-to-leakage power ratio. This ratio is used to find the optimal temperature in Fig. 14, and hence discovers the T$_i$-state's optimal long-term average voltage.

Suppose die temperature increases from 0°C to a hypothetical high temperature. T$_i$-state's total power change can be formulated as follows:

$$\begin{aligned}\Delta Power &= \Delta Dynamic + \Delta Leakage\\ &= Dynamic\_0°\text{C} * DynamicReductionRate\\ &\quad - Leakage\_0°\text{C} * LeakageIncreaseRate\end{aligned}$$

For each temperature, *DynamicReductionRate* can be modeled with V_{dd} reduction in Fig. 12. Chip vendor's technology model can provide leakage-related information. Therefore, a workload's dynamic-power intensity is what determines the output of the equation above. In practice, the dynamic power can be modeled accurately with on-chip power proxies based on performance counters [42]. The analysis module can search through all hypothetical temperatures to find the optimal setting, and instruct the cooling system to gradually coverage to the desired target temperature.

We envision that loop 2 will target the average power

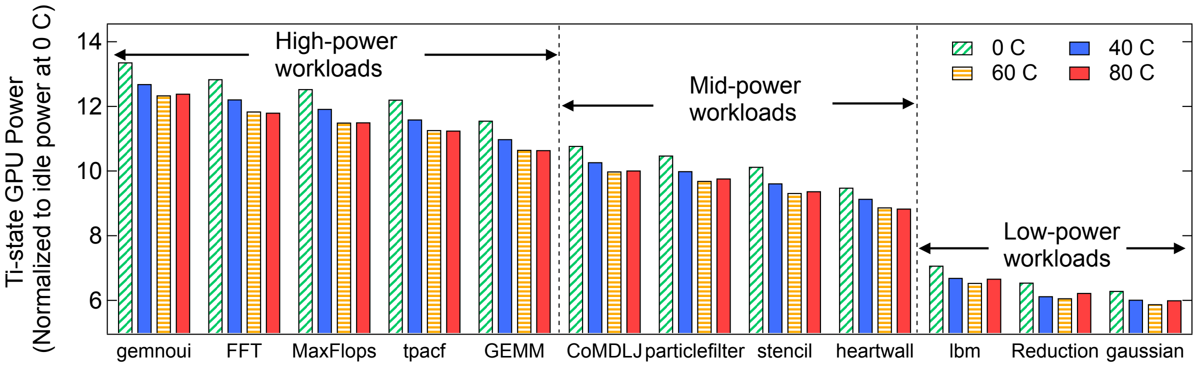


Figure 16: T_i-state's power at different temperature. Results are extrapolated using setting B, and normalized to A10-8700P's measured idle power at 0°C. High-power workloads on the left side have more power saving at higher temperature. Low-power workloads on the right side consume similar power across all temperature.

savings over a relatively long time (seconds or longer). This is because runtime temperature control by adjusting the cooling system is a relatively slow process. Many of today's workload have steady state behavior suitable for this behavior, such as scientific and deep learning applications, as well as web service workloads that have diurnal patterns [43]. Thus, it is feasible to enable power saving in this scenario.

As a case study, Fig. 16 evaluates T_i-state's power under scaling setting B, yet the insights we make apply to other possible scaling scenarios. Workloads are sorted based on power intensity. For the high power workloads on the left, 60°C and 80°C are the optimal temperatures that enables the T_i-state to have the lowest power. The 40°C case is not economic because the processor power is higher and cooling system needs to spend more power to cool down. For these workloads, control loop 2 in Fig. 15 will instruct the cooling system to stay at 60°C, or 80°C to reduce fan power.

For low power workloads on the right, 40°C, 60°C, and 80°C have similar power saving. The workload's dynamic power is not high enough to make T_i-state's dynamic power saving offset leakage increase. For these workloads, the cooling system may target a higher temperature for the same processor power while reducing the processor fan power.

For most of the workloads under study, we find 60°C to be the best temperature across all of the five scaling scenarios. At 60°C the T_i-states have reasonably large voltage reduction because the leakage power is not too high. Under the 0.7 V 300 MHz operating point we study, the GPU typically reaches 40°C. When the CPU is co-assigned with workloads, the GPU typically heats up to 60°C due to the on-die spatial temperature gradient. Thus, the temperature sweet-spots we study are within the typical operating conditions for a chip.

Compared to a fixed V_{dd} operating at 0°C, T_i-state's temperature setting can, on average, save 8.5%, 7.5%, 6.7%, 5.4% and 3.6% power for scenarios A through E that were shown previously in Table II. 0°C is ideal, yet an impractical situation in practice. When compared to a more realistic 40°C setting, T_i-states can save, on average, 9.2%, 8.6%, 8.25%, 7.5%, 6.8% processor power compared to a fixed V_{dd} system.

VI. Discussion

In this paper we examine temperature inversion's speedup effect using a 28 nm processor, and show that it is practical to turn the speedup benefit into power reduction through undervolting. In addition to our measurement results, we believe there are a few points that are worth further discussion:

a) Near-threshold voltage: Fig. 7 and Fig. 8 show that temperature inversion is stronger at lower voltages, reaching up to 10% V_{dd} reduction at 0.6 V. In our measurements, we are limited to the voltage ranges shown in the figure because of A10-8700P's intrinsic pipeline design. But we expect that at even lower voltage, such as in the near-threshold voltage regime, temperature inversion's undervolting potential will be much more stronger since the supply voltage is closer to the threshold voltage. Therefore, we expect T_i-states to have more influence in systems that rely on near-threshold computing.

b) FD-SOI: FD-SOI is expected to be used in many low-power scenarios like Internet-of-Things. In these cases FD-SOI will be engineered to reduce leakage, potentially with body-biasing. As shown in Fig. 14, T_i-states favor high temperature under low leakage technologies. For FD-SOI, ultra-low leakage could make T_i-state's optimal temperature higher than device's reliable operating limit (e.g., 100°C). For this reason, it is important to characterize FD-SOI's temperature inversion effect, particularly under body biasing, and understand T_i-state's optimal temperature for FD-SOI, which may involve a trade-off between power and reliability.

c) Cooling power reduction: In Sec. V, we show T_i-state mitigates and sometimes reverses high temperature's negative effect on processor power for FinFET and FD-SOI. T_i-state's preference for high temperature promises an extra trade-off between chip and cooling power at the system level [44]. By allowing chips to operate under higher temperature, T_i-state can enable savings on cooling system power in addition to chip power. We defer a full system-level study to future work.

d) Overclocking benefit: Other than undervolting, temperature inversion's speedup benefits can be exploited by overclocking. Choosing whether to undervolt or to overclock is a trade-off between power, performance, and efficiency. Internally, we have explored overclocking's opportunity following the same steps as Sec. IV and Algorithm 1. We find temperature inversion can increase A10-8700P GPU's frequency by 10% to 20% at 0.7 V. The frequency benefit is projected to be up to 40% at 0.6 V, and even higher in near-threshold range. Meanwhile, we also observe overclocking incurs larger di/dt droop, which requires extra timing margin beyond temperature inversion's offering. Therefore, robustly overclocking in the temperature inversion region requires tight cooperation between T_i-states and an adaptive clocking system that protects against di/dt effects [41], [11], [45].

VII. Related Work

Temperature inversion has been reported for CMOS devices long before [1], [2], [3], [4]. These works address the reason for this phenomenon, largely at the device level. Recent works study temperature inversion in FinFETs [6], [7]. Our work, however, is the first to systematically measure and characterize temperature inversion under 28 nm process and discuss its implications to the architecture and its power management.

Adaptive voltage setting for temperature variation has been recently proposed [9]. T_i-states work in a similar way to the lookup table that the authors propose. However, our work focuses on the temperature's effect in the inversion region and provides an in-depth analysis, while the solution in [9] mixes process and temperature variation together. Moreover, prior work does not address the implications of temperature control in future technologies, as we do with our FinFET analysis.

Active timing guardband management using on-chip sensors has been recently proposed [41], [13]. These prior works focus mostly on transient di/dt droop and its effect on the timing margin. In contrast, we use PSMs to characterize temperature inversion and its effect on the timing margin. We also study temperature inversion's effect in an integrated manner with di/dt droop and discuss the relationship between the two.

Many papers have addressed architecture-level temperature management [25], [26], [27], [46]. These works try to avoid excess high temperature. But we demonstrate experimentally how temperature inversion can make high temperature a friendly environment for runtime power management.

VIII. Conclusion

Temperature inversion offers us a new avenue for reducing processor power consumption. In pressing times, when power efficiency is key, temperature inversion can be leveraged to cut down the processor power by 6% to 12%, which is a non-trivial reduction in processor power consumption. Based on detailed chip measurements, we present the first public comprehensive analysis for exploiting temperature inversion. Through the introduction of T_i-states at the architecture level, we show how the extra timing margin that becomes available in the inversion region can be harnessed by a feedback-directed power management unit, and how this unit can change the way power management is performed today. Applying such optimizations in the future will likely only become more crucial as technology scaling continues and we progress into near-threshold computing, where low voltages will dominate and temperature inversion's effect are much stronger.

Acknowledgment

We thank Sriram Sundaram, Stephen Kosonocky, Aaron Grenat, and Ravinder Rachala for their help. We are also grateful to the reviewers for their constructive feedback. This work was done while Yazhou was a co-op at AMD research. In addition, the work was supported by National Science Foundation (NSF) grant CCF-1528045 and Semiconductor Research Corporation (SRC) grant 2013-HJ-2408. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding agencies (i.e., NSF or SRC).

References

[1] C. Park, J. P. John, K. Klein, J. Teplik, J. Caravella, J. Whitfield, K. Papworth, and S. Cheng, "Reversal of temperature dependence of integrated circuits operating at very low voltages," in *International Electron Devices Meeting (IEDM)*, 1995.

[2] A. Bellaouar, A. Fridi, M. Elmasry, and K. Itoh, "Supply voltage scaling for temperature insensitive cmos circuit operation," *IEEE Transactions on Circuits and Systems II: Analog and Digital signal processing*, 1998.

[3] A. Dasdan and I. Hom, "Handling inverted temperature dependence in static timing analysis," *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, 2006.

[4] D. Wolpert and P. Ampadu, "Temperature effects in semiconductors," in *Managing Temperature Effects in Nanoscale Adaptive Systems*. Springer, 2012.

[5] W. Zhao and Y. Cao, "New generation of predictive technology model for sub-45 nm early design exploration," *IEEE Transactions on Electron Devices*, 2006.

[6] W. Lee, Y. Wang, T. Cui, S. Nazarian, and M. Pedram, "Dynamic thermal management for finfet-based circuits exploiting the temperature effect inversion phenomenon," in *International Symposium on Low Power Electronics and Design (ISLPED)*, 2014.

[7] E. Cai and D. Marculescu, "Tei-turbo: temperature effect inversion-aware turbo boost for finfet-based multi-core systems," in *International Conference on Computer-Aided Design (ICCAD)*, 2015.

[8] B. Munger, D. Akeson, S. Arekapudi, T. Burd, H. R. Fair III, J. Farrell, D. Johnson, G. Krishnan, H. McIntyre, E. McLellan *et al.*, "Carrizo: A high performance, energy efficient 28 nm apu," *Journal of Solid-State Circuits (JSSC)*, 2016.

[9] S. Sundaram, S. Samabmurthy, M. Austin, A. Grenat, M. Golden, S. Kosonocky, and S. Naffziger, "Adaptive voltage frequency scaling using critical path accumulator implemented in 28nm cpu," in *Proceedings of the International Conference on VLSI Design (VLSID)*, 2016.

[10] "Ati tool." [Online]. Available: http://www.techpowerup.com/atitool/

[11] A. Grenat, S. Pant, R. Rachala, and S. Naffziger, "Adaptive clocking system for improved power efficiency in a 28nm x86-64 microprocessor," in *International Solid-State Circuits Conference (ISSCC)*, 2014.

[12] K. Gillespie, H. R. Fair, C. Henrion, R. Jotwani, S. Kosonocky, R. S. Orefice, D. A. Priore, J. White, and K. Wilcox, "Steamroller: An x86-64 core implemented in 28nm bulk cmos," in *International Solid-State Circuits Conference (ISSCC)*, 2014.

[13] Y. Zu, C. R. Lefurgy, J. Leng, M. Halpern, M. S. Floyd, and V. J. Reddi, "Adaptive guardband scheduling to improve system-level efficiency of the power7+," in *Proceedings of the International Symposium on Microarchitecture (MICRO)*, 2015.

[14] J. Leng, A. Buyuktosunoglu, R. Bertran, P. Bose, and V. J. Reddi, "Safe limits on voltage reduction efficiency in gpus: a direct measurement approach," in *Proceedings of the International Symposium on Microarchitecture (MICRO)*, 2015.

[15] N. James, P. Restle, J. Friedrich, B. Huott, and B. McCredie, "Comparison of split-versus connected-core supplies in the power6 microprocessor," in *International Solid-State Circuits Conference (ISSCC)*, 2007.

[16] V. J. Reddi, S. Kanev, W. Kim, S. Campanoni, M. D. Smith, G.-y. Wei, and D. Brooks, "Voltage smoothing: Characterizing and mitigating voltage noise in production processors via software-guided thread scheduling," in *Proceedings of the International Symposium on Microarchitecture (MICRO)*, 2010.

[17] "Altera device model." [Online]. Available: https://www.altera.com/en_US/pdfs/literature/wp/wp-01139-timing-model.pdf

[18] M. S. Gupta, V. J. Reddi, G. Holloway, G.-Y. Wei, and D. M. Brooks, "An event-guided approach to reducing voltage noise in processors," in *Proceedings of the Conference on Design, Automation and Test in Europe (DATE)*, 2009.

[19] V. J. Reddi, M. S. Gupta, G. Holloway, G.-Y. Wei, M. D. Smith, and D. Brooks, "Voltage emergency prediction: Using signatures to reduce operating margins," in *Proceedings of the International Symposium on High Performance Computer Architecture (HPCA)*, 2009.

[20] J. Leng, Y. Zu, and V. J. Reddi, "Gpu voltage noise: Characterization and hierarchical smoothing of spatial and temporal voltage noise interference in gpu architectures," in *Proceedings of the International Symposium on High Performance Computer Architecture (HPCA)*, 2015.

[21] Y. Kim, L. K. John, S. Pant, S. Manne, M. Schulte, W. L. Bircher, and M. S. S. Govindan, "Audit: Stress testing the automatic way," in *International Symposium on Microarchitecture (MICRO)*, 2012.

[22] R. Bertran, A. Buyuktosunoglu, P. Bose, T. J. Slegel, G. Salem, S. Carey, R. F. Rizzolo, and T. Strach, "Voltage noise in multi-core processors: Empirical characterization and optimization opportunities," in *Proceedings of the International Symposium on Microarchitecture(MICRO)*, 2014.

[23] T. Webel, P. Lobo, R. Bertran, G. Salem, M. Allen-Ware, R. Rizzolo, S. Carey, T. Strach, A. Buyuktosunoglu, C. Lefurgy *et al.*, "Robust power management in the ibm z13," *IBM Journal of Research and Development*, 2015.

[24] "Intel vrm design guideline." [Online]. Available: http://www.intel.com/content/www/us/en/power-management/voltage-regulator-module-enterprise-voltage-regulator-down-11-1-guidelines.html

[25] K. Skadron, M. R. Stan, K. Sankaranarayanan, W. Huang, S. Velusamy, and D. Tarjan, "Temperature-aware microarchitecture: Modeling and implementation," *ACM Transactions on Architecture and Code Optimization (TACO)*, 2004.

[26] W. Huang, S. Ghosh, S. Velusamy, K. Sankaranarayanan, K. Skadron, and M. R. Stan, "Hotspot: A compact thermal modeling methodology for early-stage vlsi design," *IEEE Transactions on Very Large Scale Integration Systems (VLSI)*, 2006.

[27] S. Fan, S. Zahedi, and B. Lee, "The computational sprinting game," in *Proceedings of the International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, 2016.

[28] "Intel 22nm finfet." [Online]. Available: http://www.intel.com/content/www/us/en/silicon-innovations/intel-22nm-technology.html

[29] "Samsung 14nm finfet." [Online]. Available: http://www.samsung.com/semiconductor/foundry/process-technology/14nm/

[30] S.-Y. Wu, C. Y. Lin, M. Chiang, J. Liaw, J. Cheng, S. Yang, M. Liang, T. Miyashita, C. Tsai, B. Hsu *et al.*, "A 16nm finfet cmos technology for mobile soc and computing applications," in *International Electron Devices Meeting (IEDM)*, 2013.

[31] S. Natarajan, M. Agostinelli, S. Akbar, M. Bost, A. Bowonder, V. Chikarmane, S. Chouksey, A. Dasgupta, K. Fischer, Q. Fu *et al.*, "A 14nm logic technology featuring 2 nd-generation finfet, air-gapped interconnects, self-aligned double patterning and a 0.0588 μm 2 sram cell size," in *International Electron Devices Meeting (IEDM)*, 2014.

[32] C.-H. Lin, B. Greene, S. Narasimha, J. Cai, A. Bryant, C. Radens, V. Narayanan, B. Linder, H. Ho, A. Aiyar *et al.*, "High performance 14nm soi finfet cmos technology with 0.0174μm 2 embedded dram and 15 levels of cu metallization," in *International Electron Devices Meeting (IEDM)*, 2014.

[33] Q. Liu, B. DeSalvo, P. Morin, N. Loubet, S. Pilorget, F. Chafik, S. Maitrejean, E. Augendre, D. Chanemougame, S. Guillaumet *et al.*, "Fdsoi cmos devices featuring dual strained channel and thin box extendable to the 10nm node," in *International Electron Devices Meeting*, 2014.

[34] "Intel 22nm finfet details." [Online]. Available: http://download.intel.com/newsroom/kits/22nm/pdfs/22nm-details_presentation.pdf

[35] C. Auth, C. Allen, A. Blattner, D. Bergstrom, M. Brazier, M. Bost, M. Buehler, V. Chikarmane, T. Ghani, T. Glassman *et al.*, "A 22nm high performance and low-power cmos technology featuring fully-depleted tri-gate transistors, self-aligned contacts and high density mim capacitors," in *International Symposium on VLSI Technology (VLSIT)*, 2012.

[36] O. Faynot, F. Andrieu, O. Weber, C. Fenouillet-Béranger, P. Perreau, J. Mazurier, T. Benoist, O. Rozeau, T. Poiroux, M. Vinet *et al.*, "Planar fully depleted soi technology: A powerful architecture for the 20nm node and beyond," in *International Electron Devices Meeting (IEDM)*, 2010.

[37] N. Planes, O. Weber, V. Barral, S. Haendler, D. Noblet, D. Croain, M. Bocat, P.-O. Sassoulas, X. Federspiel, A. Cros *et al.*, "28nm fdsoi technology platform for high-speed low-voltage digital applications," in *International Symposium on VLSI Technology (VLSIT)*, 2012.

[38] B. Pelloux-Prayer, M. Blagojević, O. Thomas, A. Amara, A. Vladimirescu, B. Nikolić, G. Cesana, and P. Flatresse, "Planar fully depleted soi technology: The convergence of high performance and low power towards multimedia mobile applications," in *IEEE Faible Tension Faible Consommation (FTFC)*, 2012.

[39] Y. Solaro, P. Fonteneau, C.-A. Legrand, D. Marin-Cudraz, J. Passieux, P. Guyader, L.-R. Clement, C. Fenouillet-Beranger, P. Ferrari, and S. Cristoloveanu, "Innovative esd protections for utbb fd-soi technology," in *International Electron Devices Meeting*, 2013.

[40] R. Rachala, S. Kosonocky, K. Heloue, R. Bochkar, A. Tula, M. Rodriguez, and E. Ergin, "Design methodology for operating in near-threshold-computing (ntc) region," in *Design Automation Conference (DAC)*, 2016.

[41] C. R. Lefurgy, A. J. Drake, M. S. Floyd, M. S. Allen-Ware, B. Brock, J. A. Tierno, and J. B. Carter, "Active management of timing guardband to save energy in power7," in *Proceedings of the International Symposium on Microarchitecture (MICRO)*, 2011.

[42] W. Huang, C. Lefurgy, W. Kuk, A. Buyuktosunoglu, M. Floyd, K. Rajamani, M. Allen-Ware, and B. Brock, "Accurate fine-grained processor power proxies," in *Proceedings of the International Symposium on Microarchitecture (MICRO)*, 2012.

[43] D. Lo, L. Cheng, R. Govindaraju, L. A. Barroso, and C. Kozyrakis, "Towards energy proportionality for large-scale latency-critical workloads," in *Proceeding of the International Symposium on Computer Architecture (ISCA)*, 2014.

[44] W. Huang, M. Allen-Ware, J. B. Carter, E. Elnozahy, H. Hamann, T. Keller, A. Lefurgy, J. Li, K. Rajamani, and J. Rubio, "Tapo: Thermal-aware power optimization techniques for servers and data centers," in *International Green Computing Conference and Workshops (IGCC)*, 2011.

[45] K. Bowman, S. Raina, T. Bridges, D. Yingling, H. Nguyen, B. Appel, Y. Kolla, J. Jeong, F. Atallah, and D. Hansquine, "A 16nm auto-calibrating dynamically adaptive clock distribution for maximizing supply-voltage-droop tolerance across a wide operating range," in *International Solid-State Circuits Conference (ISSCC)*, 2015.

[46] A. Raghavan, Y. Luo, A. Chandawalla, M. Papaefthymiou, K. P. Pipe, T. F. Wenisch, and M. M. Martin, "Computational sprinting," in *International Symposium on High Performance Computer Architecture (HPCA)*, 2012.