Experimental Assessment and Biaffine Modeling of the Impact of Ambient Temperature on SoC Power Requirements

Kameswar Rao Vaddina[1], Florian Brandner[1], Gérard Memmi[1], and Pierre Jouvelot[1,2]

[1] LTCI - Télécom Paris - Institut polytechnique de Paris, Palaiseau, France
[2] Mines Paris, PSL University, Paris, France

Abstract. Based on fundamental physics-based considerations, we introduce the Biaffine Temperature-Voltage power model (BiTV) for SoC systems, which takes the influence of dynamic voltage, frequency, and ambient temperature conditions into account. Using an ARM-Cortex-based AM572x system operating in a temperature-controlled oven, we provide experimental evidence of the validity of the BiTV power model over a significant range of ambient temperatures (25 to 55 °C), voltages (0.98 to 1.23 V) and frequencies (100 to 1,500 MHz).
These experiments and the BiTV model provide quantitative elements to assess the impact of ambient temperature on systems' performance. Such insights could be of use to system designers and compiler writers, in particular when dealing with embedded systems operating in harsh conditions or under energy-critical constraints.

Keywords: Energy · temperature · energy profiling · monitoring infrastructure.

1 Introduction

The laws of thermodynamics impose that the warmer the ambient temperature is, the more difficult it will be to cool down silicon-based processing devices, and thus the more energy these devices will consume. Many research works investigate the subtle interplay between energy, power, and temperature at a rather large system scale (see, for instance, among the recent ones, [Guermouche and Orgerie(2022)]). The situation is, in fact, even more complex at a finer-grained level, where system-provided performance and environmental counters typically used in the above-mentioned lines of research might not provide the precision required. In particular, since many embedded systems are operated on energy-constrained batteries and in various mobile settings (cold remote outdoors, temperate indoors, hot engines, etc.) [Prakash et al.(2020)], a detailed understanding of how power requirements and energy consumption parameters evolve with respect to temperature is clearly needed to implement performance-increasing strategies.

If the system-on-chip (SoC) manufacturers provide tools such as Xilinx Power Estimator [Inc.(2018)] or Intel FPGA Power and Thermal Calculator [Intel(2020)] for power-related analysis, they are CPU-intensive and very complex to set up and use, thus being mostly dedicated to the design phase of SoCs. Our goal here is to come up with a simpler yet experimentally and physically validated model that could be used, say, within a compiler or an embedded runtime to take temperature-related power-management decisions. Such a model could be, for instance, used to decide when and how to increase the CPU frequency to meet real-time computational demands while taking into account the temperature conditions. This is a key element to ensure that the processor temperature is always maintained within its manufacturer-specified limits and thus that thermal runaways are avoided.

Unfortunately, setting up an experiment to inform such an issue is rather complex for a typical computer-oriented academic laboratory, which may explain the scarcity of related works in the domain (see Section 5), even though the application spectrum is rather vast, in particular when dealing with battery-equipped systems.

In this paper, we describe how we conducted such an experiment on one specific system, namely a Texas Instrument AM572x chip. Our contributions are (1) an experimental assessment of the impact of ambient temperature on the energy and power characteristics of a SoC system, and (2) the Biaffine Temperature-Voltage power model (BiTV), a physics-inspired and experimentally-validated analytical model of SoC system behavior with respect to both ambient temperature, dynamic voltage, and CPU clock frequency. Our experiments suggest that the BiTV power model provides a good approximation of the system behavior over a significant spectrum of temperatures, voltages, and frequencies.

These new results are a considerable improvement with regard to the previous work of [Skadron et al.(2003)][Chandrakasan et al.(1996)][Vaddina et al.(2017)], where all experiments were run at constant ambient temperature. Using a temperature-controlled oven, we report here on an extensive data-gathering run, from which a detailed analysis of how such a system behaves for various temperature points is performed. Even though we focus in this paper on one particular SoC system, we believe that the approach taken here and the corresponding results should be, thanks to their reliance on physics-informed concepts (see Section 2), similar for other SoC systems overall. Of course, future work will be needed to confirm this hypothesis.

The structure of the paper is as follows. In Section 2, we introduce the BiTV power model. In Section 3, we describe the experimental setup and protocol used to check the validity of BiTV. In Section 4, we present and analyze the data that support this model. Related work is presented in Section 5. We conclude and discuss future work in Section 6.

2 The BiTV power model

Even though transistor-level models exist to study how the various involved currents (gate leakage current, sub-threshold current, etc.) vary with respect to say temperature, voltage or technology-specific parameters such as channel length and width, scaling these models to the billions of transistors found in a typical chip is very difficult. Given the interplay between all these elements, power requirements for digital computing devices P_{cpu} are thus linked to many complex physical processes. A coarse definition for this power requirement is:

$$\begin{aligned} P_{cpu} &= P_{dynamic} + P_{static} \\ &= P_{dynamic} + P_{gate} + P_{subthreshold}, \end{aligned} \tag{1}$$

where the static power is considered to be the sum of two dominant components [Narendra and (eds.)(2006)]: (1) the sub-threshold (leakage) power, of key importance, since it doesn't contribute to the integrated circuit's (IC) function and constitutes a significant fraction of IC energy consumption [Lucian(2011)], and (2) the usually smaller gate-leakage power.[3] Each power component P_i corresponds to a current component I_i, such that $P_i = V_{dd} I_i$, where V_{dd} is the supply voltage at drain. An approximate assessment of a component's energy consumption while running a device, E_i, can then be obtained as $P_i t$, where t is the time during which the system is run.

There exist analytical and numerical models for each of these components (see, for instance, Chandrakasan et al. [Chandrakasan et al.(1996)] or Skadron et al. [Skadron et al.(2003)]). Even though these apply only to the smallest components of an actual SoC and thus cannot be used directly at the system level, one can still build upon them and Equation 1. In particular, these observations suggest the following constraints to follow when designing a parameterized temperature-aware system-level analytical power model.

- $I_{dynamic}$ is linked to the actual running of programs, and can be deemed proportional (by a factor ϵ) to the frequency f of the processor and to V_{dd} [De Vogeleer et al.(2014a)]:

  $$I_{dynamic} = \epsilon f V_{dd}.$$

- $I_{subthreshold}$ is proportional to T^2 and to an exponential of a function of the temperature T and V_{gs}, the gate-to-source voltage:

  $$I_{subthreshold} = \alpha T^2 e^{\beta (V_{gs} - V_{th})/T},$$

  where V_{th} is the threshold voltage [Liu et al.(2007)]. Since V_{gs} varies between 0 and V_{dd}, V_{th} being its minimum, we decide to approximate the voltage term $V_{gs} - V_{th}$ in the exponent by Taylor expansion around the minimum of the voltage range (which is narrow here, between 0.98 V and

[3] The usually smaller short-circuit power dissipation effects are here ignored.

1.23 V, see Section 3) to first order in V_{dd}. After rewriting αT^2 as $e^{\ln(\alpha T^2)}$ and adding the exponents, we further simplify the model, since the considered ambient-temperature range is here, when expressed in Kelvins, narrow (from 25 °C to 55 °C, see Section 3), by approximating β/T and $\ln(\alpha T^2)$ by Taylor expansion around 273.15 K (0 °C) to first order in T. The resulting approximate exponent is then of the form $a + bV_{dd} + cT + dV_{dd}T$, where T is now expressed in °C.
- I_{gate} depends in a complex manner upon T and V_{dd}. However, since this term is most of the time very small when compared to $I_{subthreshold}$, we assume it to be 0, to simplify the model [Liu and Kursun(2007)].

We suggest thus to build upon this high-level analysis of the physical foundations behind the modeling of the power requirement, P_{cpu}, for a single CPU to generalize it for a whole SoC board, at least as far as CPU-bound tasks are concerned. Abstracting over the formal mathematical formulae introduced above, we introduce thus the *Biaffine Temperature-Voltage Power Model* (BiTV)[4] to approximate the power requirements of SoCs. It is specified as follows.

$$P_{BiTV} = V_{dd}(\epsilon f V_{dd} + e^{a+bV_{dd}+cT+dV_{dd}T}). \tag{2}$$

The BiTV model parameters ϵ, a, b, c, and d are linked to the particularities of the system at hand such as the technology parameters or the actual code being run on the device (OS, user code, I/O, etc.). In the rest of this paper, we provide experimental evidence for the validity of the BiTV power model.

3 Experimental setup

In order to validate BiTV under varying conditions, including benchmark characteristics, temperature, processor frequency, and supply voltage, we performed a campaign measuring the instantaneous power requirements of a SoC system. The setup for these measurement experiments is shown in Figure 1.

3.1 Equipment

The central component of this setup is an oven that allows us to precisely control the ambient temperature inside the oven's chamber. The oven model is a UFP 400, produced by Memmert GmBH (Germany), which has a chamber volume of 53 l, a temperature range starting from 5 °C above room temperature (no cooling possible) up to 300 °C, and forced air convection. The temperature probes of the oven are located at the top of the chamber. We used the oven to heat the chamber to 25, 35, 45, and 55 °C. We refer to the temperature inside the oven's chamber as the *ambient temperature* for the experiments.

As a SoC system, we use an AM572x EVM development board manufactured by Texas Instruments. The board hosts a Sitara ARM SoC chip, 2 GB DDR3L

[4] BiTV is loosely called "biaffine" since it is affine in its exponent term with respect to each of its T and V_{dd} parameters when the other is held constant.

Fig. 1: Representative picture of the experimental setup. The AM572x board is inside the thermal oven. It is connected to the host system and NI cDAQ with its I/O modules NI 9215 and NI 9211. [Oven courtesy Centre des matériaux, Mines Paris, Evry]

memory, 4 GB of flash memory, a TPS659037 power management chip, and several connectors (audio, HDMI, Ethernet, USB, etc.). For our measurements, we are interested in monitoring the power of the microprocessors inside the Sitara SoC only. We thus modified the board to directly measure the power at the current-sense resistors of the microprocessor submodule, which are intended for high-precision power monitoring and thus allow measurements independent from other components on the board/SoC. The chip itself is fabricated using a 23-nm design process and contains two ARM Cortex A15 cores in the microprocessor submodule. During the experiments, only a single core was active, running the Linux operating system from Texas Instruments' Processor SDK (v.04).

We modified the setup of the Linux power management to allow us to manually control the microprocessor's clock frequency and voltage settings. The processor supports three voltage settings: `OPP_NOM` (0.98 V), `OPP_OD` (1.09 V), and `OPP_HIGH` (1.23 V).[5] The clock frequency can be controlled in steps of 100 MHz in the range from 100 to 1,500 MHz. However, the processor specification imposes a minimum voltage level for certain frequency settings, i.e., `OP_NOM`, up to 1,000 MHz, `OPP_OD`, up to 1,176 MHz, and `OPP_HIGH`, for higher frequencies. Note that it is possible to impose higher voltage levels; for our measurements, we nevertheless always use the lowest possible voltage level for a given frequency.

[5] Note that these voltage numbers are specific to a particular board and thus may slightly vary between different boards.

Power monitoring is semi-automatized using National Instruments LabVIEW software running on a separate host machine (a MacBook Pro running Windows, in this case). The host machine is connected to a Compact DAQ data-acquisition module by National Instruments (NI cDAQ-9174) holding an I/O module NI 9215, in-turn connected to the aforementioned current-sense resistors on the microprocessor submodule[6] of the AM572x EVM development board. A second I/O module, an NI 9211, is connected to a temperature probe at the heatsink. The measured sensor data from both I/O modules are processed by the cDAQ module and sent to the host machine, where the data is annotated and recorded in a trace file. LabVIEW and the cDAQ equipment ensure a tight synchronization between the two measured data streams.

3.2 Protocol

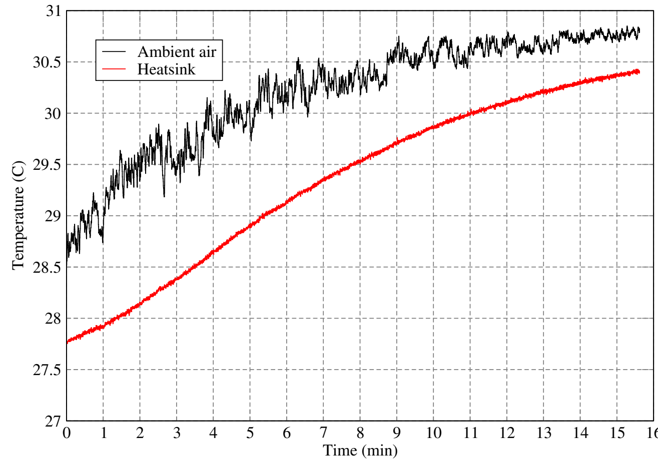


Fig. 2: Ambient air and heatsink temperatures when the system is shutdown.

The AM572x EVM development board is placed inside the oven at the 2nd of 4 levels of shelves in order to improve air circulation and minimize the time to reach the target temperature. Figure 2 provides an idea of the time needed to heat the oven's chamber with respect to a specified target temperature. In addition to the oven's temperature probes, we also placed, as mentioned above, a probe at the SoC's heatsink. This *heatsink temperature* is the best possible approximation of the actual temperature conditions of the transistors inside the chip that can be measured from the outside. More precisely, we compute this last so-called *junction temperature*, i.e., the temperature at the transistor level,

[6] This submodule hosts the 2 processor cores, the L1 and L2 caches, the boot ROM, the power management unit and the PLL.

from the thermal resistance of the heatsink [CTS Electronic Components(2006)] and the SoC chip [TI Inc.(2011),TI Inc.(2017),TI Inc.(2018)]. Note that the chip itself also provides integrated temperature probes, but they are not very precise and reliable, e.g., reading temperature values from these sensors interferes with the execution of the current program.

For measurements, we consider 3 benchmark kernels: `Goldrader` (a bit-reversal algorithm), `Blowfish` (a symmetric block cipher), and `SHA` (a hashing function). These kernels are compute-bound and thus mostly stress the microprocessors, but do so with a different mix of instructions. Thus running these kernels for a short period (a couple of seconds) allows us to optimally control the actual temperature conditions at the transistor level of the chip, while monitoring power of a processor core in isolation. Given that these running times are rather short, we assume the chip-level temperatures to remain quasi-constant.

As can be seen in Figure 2, it takes about 10 minutes to increase the ambient temperature from 28.5 °C to roughly 30.5 °C. It can also be seen that the heatsink temperature for a system that is shutdown follows with a slight lag, but eventually converges towards the ambient temperature. Heating times are considerable; the experiments were thus performed in batches with temperature strictly increasing within a batch. More precisely, a batch starts with a completely cooled oven (at room temperature); we then set the oven to a target temperature of 25 °C, while keeping the SoC system in an `IDLE` state. Once the oven and heatsink temperatures have both reached the target temperature, we run one of the short kernels as a benchmark at the lowest possible clock frequency (100 MHz) and put the system again into an `IDLE` state. Running the kernel might slightly increase the temperature inside the chip; we thus keep the system in an `IDLE` state for a while to allow any excess heat to dissipate before proceeding to the next measurement, i.e., by increasing the clock frequency by 100 Mhz, switching the benchmark kernel, or increasing the target temperature by 10 °C. We also made sure that all runs were using warm cache states.

We performed a series of experimental campaigns to gather physical data, namely power requirements for the three benchmark kernels running at different settings. The parameters under study were the following:

- benchmark B. One out of the three kernels `Goldrader`, `Blowfish`, or `SHA`;
- temperature T. Ambient temperatures of 25, 35, 45, or 55 °C in the oven;[7]
- frequency f. Clock frequency of the microprocessor module, varied in steps of 100 MHz from 100 to 1,500 MHz;
- voltage V_{dd}. Minimal voltage settings `OPP_NOM`, `OPP_OD`, or `OPP_HIGH`, depending on the requirements of the clock frequency.

[7] We limited the maximum ambient temperature to 55 °C, since going beyond this value would increase the on-chip temperature to more than 90 °C, at which point the CPU would experience a thermal emergency and shutdown. Cooling the board to less than 25 °C would require a refrigerating enclosure (see Section 6).

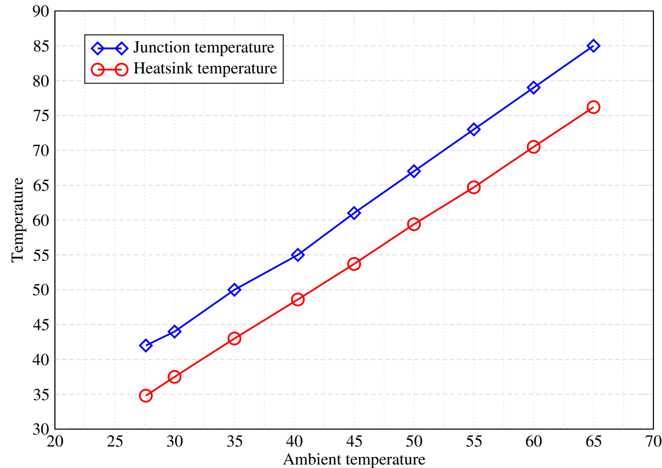


Fig. 3: Junction and heatsink temperatures with respect to ambient temperature (in °C). The system is running `IDLE`.

3.3 Temperature calibration

The introduction of an oven to induce higher ambient-temperature conditions introduces a new constraint on heat transfer via the chip's heatsink. There are thus many different temperatures involved in the current experiment. The model in Equation 2 relies on a temperature T that is, in fact, the junction temperature T_j, while what is controlled by the oven is the ambient temperature T_a. In addition, our temperature probe is mounted on the SoC chip's heatsink, thus providing us another temperature point T_h.

We compared these different temperatures as follows. As mentioned above, we derived T_j from T_h considering the thermal resistance parameters as specified for the heatsink [CTS Electronic Components(2006)] and the SoC chip [TI Inc.(2017)] using a standard manufacturer-provided formula [TI Inc.(2011)]. An on-chip microprocessor-domain thermal sensor measures the approximate junction temperature, and their values can be retrieved from the command line using the *sysfs* pseudo file system, a feature provided by the Linux kernel. After allowing the board to soak for 5 minutes, the junction temperature T_j corresponding to each ambient temperature T_a has thus been recorded. Using the junction temperature model and θ_i parameters provided in [TI Inc.(2017)], we also checked for the consistency of our measurements with the modeled ones.

Figure 3 relates the data points for T_j, T_h, and T_a for a system in an `IDLE` state and running at the lowest voltage/frequency setting. For every degree increase in T_a, as long as the system is running `IDLE`, T_j increases by a factor of 1.135. The obtained numbers are consistent with measurements by Texas Instruments Incorporated, the manufacturer of the SoC [TI Inc.(2018)].

When assessing the validity of the BiTV model, one has thus to be careful to use the proper temperature variant. Note however that, since the relationships

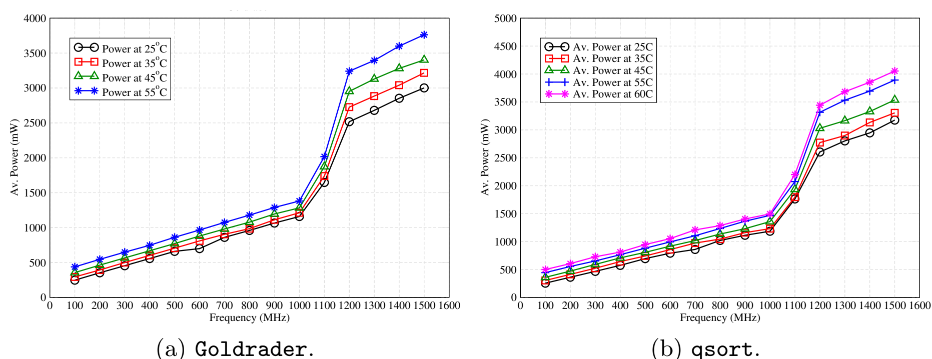
between temperatures are mostly linear when running `IDLE`, we surmised that using the biaffine model could still be valid using any of the above mentioned temperatures, the only impact being possible changes in the proportionality coefficients and parameters. This is, in fact, what we validate via the experiments described below. Unless otherwise mentioned, we consider thus from now on that T, the *ambient* temperature, is the parameter of interest here.

4 Results

Below we present the experimental results obtained using the previously described setup. Given the size of the gathered data, when discussing specific results, we select in this section the most interesting and/or illustrative subset of parameter values for B (benchmark index), T (°C), f (MHz), and V_{dd} (V). The other results are similar.

4.1 Measurements

In Figure 4a is plotted the measured average power when running $B =$ `Goldrader` for various T and f (i.e., also V_{dd}). As expected, the power needs increase with f. Three different regimes can be noticed, clearly linked to the frequency boundaries of the 3 different supply voltage settings (V_{dd}) required by the Texas Instruments chip (see Section 3). The significant and non-linear impact of T on power requirements is clearly visible, in particular at high-frequency settings. The non-linearity of power with respect to V_{dd} can indirectly be seen at the mid-voltage setting, since the points for frequencies 1, 1.1 and 1.2 GHz, corresponding each to a different value of V_{dd}, are not aligned.



(a) `Goldrader`. (b) `qsort`.

Fig. 4: Average power vs. frequency for `Goldrader` (4a) and `qsort` (4b).

The three core benchmarks B_i are rather small program kernels. They amount to about a dozen of lines of code each and run rather fast (depending on frequency, between 8 and 120 s for `Goldrader`, 21 and 331 s for `Blowfish`, and 43

and 683 s for `SHA`). Thus, we also decided to run some more significant programs to further explore the impact of the ambient temperature on the thermal behavior under complex load. Using typical benchmark examples, namely `susan`, `bitcount`, `basicMath`, and `qsort` from the MiBench benchmark suite for embedded systems [Guthaus et al.(2001)], we obtained quite similar experimental results to those found for smaller benchmarks (see, for instance, Figure 4b), suggesting that a unique generic model of power requirements could probably be designed, which is what we did with BiTV.

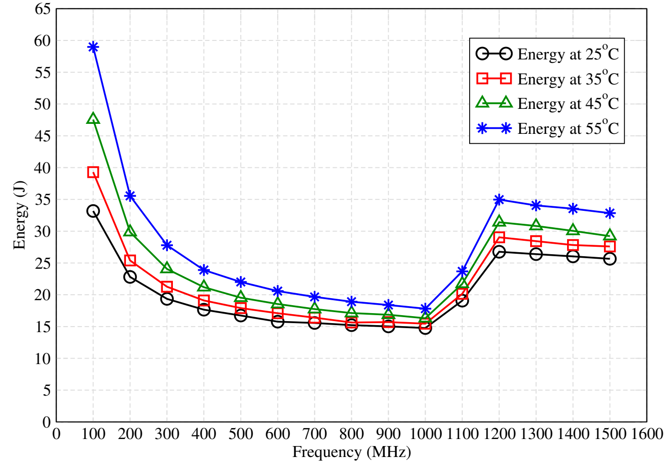


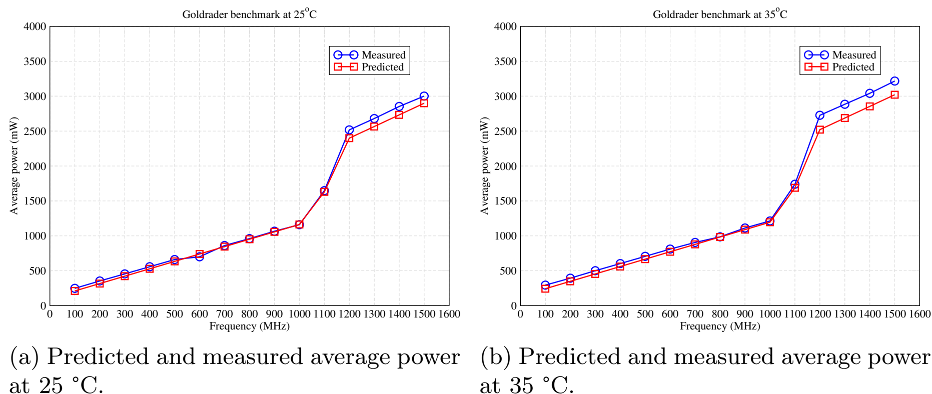
Fig. 5: Energy consumption vs. frequency and temperature (`Goldrader`).

Finally, when multiplying an average power value by the total running time of the corresponding program, one gets its energy consumption profile, e.g., for `Goldrader` in Figure 5. One can clearly see the typical convex energy/frequency curve discussed by De Vogeleer et al [De Vogeleer et al.(2014a)], although the characteristics of these convexity profiles vary for each of the three frequency intervals mentioned in Section 3. Notice that, in particular when running slowly, the energy consumption can vary by a quite significant factor with ambient temperature (almost 2, between 25 and 55 °C). Also, this graph suggests that the energy/frequency rule keeps being valid for each frequency interval when varying the ambient temperature, a property not experimentally checked until now.

4.2 BiTV model assessment

To assess the validity of BiTV, we performed a non-linear fitting of P_{BiTV} (Equation 2) using the experimental power data gathered from running the `Blowfish` benchmark kernel. We used the default solver for non-linear problems SWARM from LibreOffice. The model parameters we obtained for BiTV that minimize

the standard error for the `Blowfish` kernel are: $\epsilon = 1.10$, $a = -1.32$, $b = 4.44$, $c = 3.85 \times 10^{-2}$, and $d = -3.67 \times 10^{-4}$.



(a) Predicted and measured average power at 25 °C.

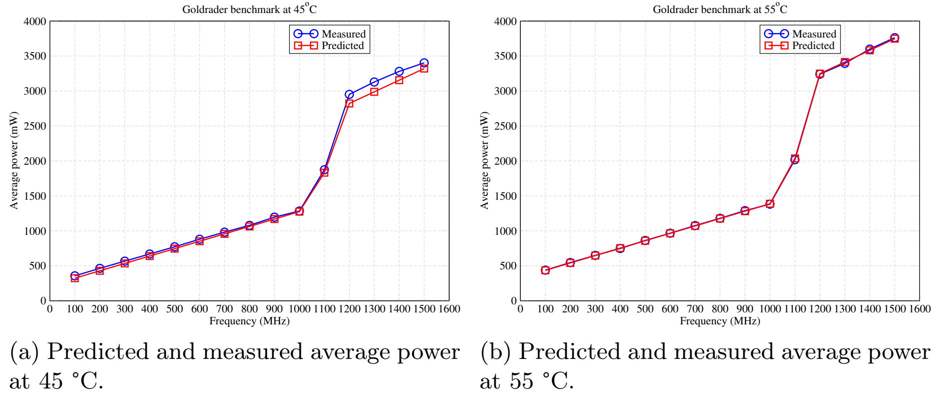
(b) Predicted and measured average power at 35 °C.

Fig. 6: Comparison of BiTV-predicted (red) and measured (blue) average power (in mW) at 25 and 35 °C while varying voltage and frequency (`Goldrader`).

We then applied the obtained model parameters to the other two benchmark kernels `Goldrader` and `SHA`. Figures 6 and 7 illustrate the good fit between the measured average power (in blue) and the BiTV-predicted average power (in red) for the `Goldrader` benchmark at various temperature (T), voltage (V_{dd}), and frequency (f) settings. The best fit occurs at $T = 55$ °C (Figure 7, right), while some divergence can be observed for all other settings – notably towards the high end of the frequency spectrum. Yet, the relative (with respect to the mean) standard errors of estimate (RSE) for these four temperature values are all below 2% for `Goldrader`.

Applying the model with the same parameter values (for ϵ, a, b, c, and d) to `SHA` yields similar results, which are not shown here. The RSE values vary slightly, but are still below 2% for all configurations. This suggests that the BiTV power model provides good accuracy even across the considered benchmark kernels and a wide range of temperatures.

As a final check, we compared the actual total energy consumed by `SHA` with a prediction based on the BiTV model, which is computed by multiplying the modeled power value (again using the same model-parameter values) for `SHA` by its actual running time. The results, for the temperatures (T) 45 and 55 °C and varying voltage and frequency settings (V_{dd} and f), are given in Figure 8. One can observe a good fit between the related curves, albeit with a clearly visible gap. This gap can be explained by the fact that the model was not fit using the `SHA` power data, consequently leading to a slight error (recall that the RSE was small, below 2%, but not zero). In addition, we can observe an



(a) Predicted and measured average power at 45 °C.

(b) Predicted and measured average power at 55 °C.

Fig. 7: Comparison of BiTV-predicted (red) and measured (blue) average power (in mW) at 45 and 55 °C while varying voltage and frequency (`Goldrader`).

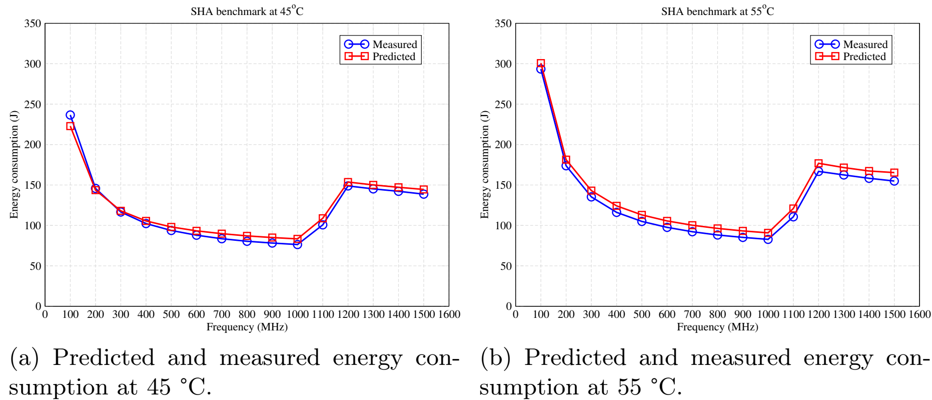
amplification of the absolute model error probably due to the multiplication with the benchmark's running time.

5 Related work

The impact of ambient temperature on CPU power requirements and/or energy consumption is extensively studied at the large-grain level of data centers or HPC farms (see, for instance, [El-Sayed et al.(2012)] [Gupta et al.(2021)]). Yet, research that focuses on more low-level devices, yet at a coarser level than Register Transfer Level (RTL), is rarer. A possible reason for this might be that experimental validation of analytical models is a complicated affair at such a small scale, requiring somewhat complex equipment and protocols.

Among significant related work, De Vogeleer et al. [De Vogeleer et al.(2014b)] introduce a temperature-aware power model for the Samsung Galaxy A7 and A15 processors. They experimentally confirm the exponential behavior of power w.r.t. temperature and equip an analytical model with parameters that are polynomial functions of temperature, frequency and number of cores. The use of an oven in our experiments allows for a much better controlled initial-temperature condition for the measurements. In particular, De Vogeleer et al. [De Vogeleer et al.(2014b)] manually forced heating, resulting in a less homogeneous environment for the device than in our experiments. Finally, the BiTV model is built on more physics-based foundations and is able to handle varying voltage.

Vaddina et al. [Vaddina et al.(2021)] proposed a workflow for energy and temperature profiling on high-performance systems running parallel applications. They carried out their experiments on Intel's X86-based multi-core processors, utilizing the NAS parallel benchmark suite. Their approach allows full and dynamic runtime control over the execution of applications, ensuring that the pro-



(a) Predicted and measured energy consumption at 45 °C.

(b) Predicted and measured energy consumption at 55 °C.

Fig. 8: Comparison of BiTV-predicted (red) and measured (blue) total energy consumption at 45 and 55 °C while varying (voltage and) frequency, in MHz (`SHA`). The fit is similar at 25 °C (and thus omitted here), while the experimental data are missing at 35 °C, due to technical issues during the experiment.

cessors' frequency remains within a specified range. They demonstrated that the energy response to frequency scaling is greatly influenced by the characteristics of the workload and forms a convex function around the optimal frequency point. Despite this previous work by the same lead author, the present paper, attempting to tackle similar issues, focuses on a totally different architecture, which, by itself, justifies this new work. In addition to introducing the new BiTV model, it also demonstrates the universality of the taken approach, methodology and theoretical framework, which are thus relevant to different architectures.

Recent work by Texas Instruments [TI Inc.(2018)] as well as Intel/ARM [Singla et al.(2015),Bhat et al.(2018)] used a similar setup as ours, notably with a temperature-controlled oven.[8] The measurements by Texas Instruments are also based on an AM572x EVM board and are in line with our measurements (cf. Figure 3), considering an `IDLE` system [TI Inc.(2018), Figure 1]. Other results are, however, not comparable, since they measured the power requirements of the entire SoC system using the `Drystone` benchmark. Yet, our measurements still follow the same overall trends. The biggest difference, though, is that Texas Instruments only provides measured data and does not introduce a power model that may be used for predictions.

The work supported by Intel and ARM uses different SoC systems based on the Odroid platform (Samsung Exynos 5410 and Exynos 5422) and aim at designing a Dynamic Thermal and Power Management (DTPM) algorithm (see [Singh et al.(2020)] for a survey). In the initial work [Singla et al.(2015)], the objective is to predict the evolution of the SoC system's temperature in the future and apply dynamic voltage and frequency scaling to control the tempera-

[8] This research was performed independently of ours.

ture – even without cooling support by a fan. In more recent work, the algorithm was extended to incorporate a power model [Bhat et al.(2018)] whose parameters have been determined in a similar setup as ours. However, the power model itself is only compared to the measured data, but otherwise is not evaluated. Instead, the DTPM algorithm is evaluated; it performs runtime monitoring and, based on a mix of measured and predicted parameters, tries to find an optimal frequency setting when the SoC's temperature exceeds a certain threshold. Our work focuses on the power model itself. Notably, we introduce a simpler analytical power model, explain its link to physical foundations, and show its adequacy with experimental measurements across different benchmarks as well as temperature, frequency, and voltage settings.

6 Conclusion and future work

Based on physics-informed and experimental considerations, we modeled and quantified the influence of the ambient temperature on the power requirements of SoC systems at the microprocessor level. We believe these results must be considered in future system energy profiles, especially when running on batteries and dealing with energy-critical applications. We introduced a new ambient-temperature-, frequency- and voltage-aware power model, BiTV. Preliminary experiments on an ARM-based AM572x system suggest that it provides a very good fit with actual experimental data on a wide spectrum of temperature, voltage, and frequency settings even across benchmark kernels.

For future work, it would be interesting to experiment with and analyze the influence of the ambient temperature at levels lower (typically in the $[-30, 20]$ degree Celsius) than the ones used in the experiments presented in this paper, in order to cover the range of realistic outdoor temperatures and assess the generality of the BiTV model in this extended domain.

Another interesting venue for research would be to provide a more scientific grounding to the BiTV analytical model, in particular the values of its parameters, based on more fundamental physics taking into account aggregation effects. This is an important issue since, for now, the model parameters are learned from experimental data; the cost of such experiments can only be justified for products deployed in large numbers.

Finally, and even though we expect the physical grounding of BiTV to make it somewhat universal, extending the types of computer boards (and even studying the impact of the individual differences among boards of the same type) and benchmarks used to validate and/or extend it is needed. In particular, the test programs used in this paper are CPU-bound kernels. Studying how well BiTV can be adapted, via parameter changes or more fundamental generalizations, to programs that access different memory-cache levels or boards that use more recent SoCs or other silicon-manufacturing technologies is clearly warranted.

Acknowledgments

This work was supported in part by ANR under the France 2030 program, grant NF-NAI: ANR-22-PEFT-0003. We thank the Centre des matériaux (Mines Paris, PSL University) for allowing access to and providing assistance for their temperature-controlled oven and the reviewers for their useful comments.

References

[Bhat et al.(2018)] Ganapati Bhat, Gaurav Singla, Ali K. Unver, and Umit Y. Ogras. 2018. Algorithmic Optimization of Thermal and Power Management for Heterogeneous Mobile Platforms. *IEEE Trans. Very Large Scale Integr. Syst.* 26, 3 (mar 2018), 544–557. https://doi.org/10.1109/TVLSI.2017.2770163

[Chandrakasan et al.(1996)] Anantha Chandrakasan, Isabel Yang, Carlin Vieri, and Dimitri Antoniadis. 1996. Design Considerations and Tools for Low-Voltage Digital System Design. In *Proceedings of the 33rd Annual Design Automation Conference (DAC '96)*. Association for Computing Machinery, 113–118. https://doi.org/10.1145/240518.240540

[CTS Electronic Components(2006)] CTS Electronic Components 2006. *Cooling Critical Components*. CTS Electronic Components. Application Note – AN1010.

[De Vogeleer et al.(2014a)] Karel De Vogeleer, Gerard Memmi, Pierre Jouvelot, and Fabien Coelho. 2014a. The Energy/Frequency Convexity Rule: Modeling and Experimental Validation on Mobile Devices. In *Parallel Processing and Applied Mathematics*. Springer Berlin Heidelberg, Berlin, Heidelberg, 793–803.

[De Vogeleer et al.(2014b)] Karel De Vogeleer, Gerard Memmi, Pierre Jouvelot, and Fabien Coelho. 2014b. Modeling the temperature bias of power consumption for nanometer-scale CPUs in application processors. In *2014 International Conference on Embedded Computer Systems: Architectures, Modeling, and Simulation (SAMOS XIV)*. 172–180. https://doi.org/10.1109/SAMOS.2014.6893209

[El-Sayed et al.(2012)] Nosayba El-Sayed, Ioan A. Stefanovici, George Amvrosiadis, Andy A. Hwang, and Bianca Schroeder. 2012. Temperature Management in Data Centers: Why Some (Might) like It Hot. In *Proceedings of the 12th ACM SIGMETRICS/PERFORMANCE Joint International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS '12)*. Association for Computing Machinery, New York, NY, USA, 163–174. https://doi.org/10.1145/2254756.2254778

[Guermouche and Orgerie(2022)] Amina Guermouche and Anne-Cécile Orgerie. 2022. Thermal design power and vectorized instructions behavior. *Concurr. Comput. Pract. Exp.* 34, 2 (2022). https://doi.org/10.1002/cpe.6261

[Gupta et al.(2021)] Rohit Gupta, Sahar Asgari, Hosein Moazamigoodarzi, Douglas G. Down, and Ishwar K. Puri. 2021. Energy, exergy and computing efficiency based data center workload and cooling management. *Applied Energy* 299 (2021), 117050. https://doi.org/10.1016/j.apenergy.2021.117050

[Guthaus et al.(2001)] M.R. Guthaus, J.S. Ringenberg, D. Ernst, T.M. Austin, T. Mudge, and R.B. Brown. 2001. MiBench: A free, commercially representative embedded benchmark suite. In *Proceedings of the Fourth Annual IEEE International Workshop on Workload Characterization. WWC-4 (Cat. No.01EX538)*. 3–14. https://doi.org/10.1109/WWC.2001.990739

[Inc.(2018)] Xilinx Inc. 2018. Xilinx Power Estimator User Guide.

[Intel(2020)] Intel. 2020. Intel® FPGA Power and Thermal Calculator User Guide.

[Liu et al.(2007)] Yongpan Liu, Robert P. Dick, Li Shang, and Huazhong Yang. 2007. Accurate Temperature-Dependent Integrated Circuit Leakage Power Estimation is Easy. In *Proceedings of the Conference on Design, Automation and Test in Europe (DATE '07)*. EDA Consortium, 1526–1531.

[Liu and Kursun(2007)] Zhiyu Liu and Volkan Kursun. 2007. PMOS-Only Sleep Switch Dual-Threshold Voltage Domino Logic in Sub-65-Nm CMOS Technologies. *IEEE Trans. Very Large Scale Integr. Syst.* 15, 12 (dec 2007), 1311–1319. https://doi.org/10.1109/TVLSI.2007.903947

[Lucian(2011)] Shirfen Lucian. 2011. Leakage Power - it's worse than you think. (2011). https://www.eetimes.com/leakage-power-its-worse-than-you-think/

[Narendra and (eds.)(2006)] S. G. Narendra and A. P. Chandrakasan (eds.). 2006. *Leakage in nanometer CMOS technologies*. Springer Science & Business Media.

[Prakash et al.(2020)] Alok Prakash, Siqi Wang, and Tulika Mitra. 2020. Mobile Application Processors: Techniques for Software Power-Performance Optimization. *IEEE Consumer Electronics Magazine* 9, 4 (2020), 67–76. https://doi.org/10.1109/MCE.2020.2969171

[Singh et al.(2020)] Amit Kumar Singh, Somdip Dey, Klaus McDonald-Maier, Karunakar Reddy Basireddy, Geoff V. Merrett, and Bashir M. Al-Hashimi. 2020. Dynamic Energy and Thermal Management of Multi-core Mobile Platforms: A Survey. *IEEE Design & Test* 37, 5 (2020), 25–33. https://doi.org/10.1109/MDAT.2020.2982629

[Singla et al.(2015)] Gaurav Singla, Gurinderjit Kaur, Ali K. Unver, and Umit Y. Ogras. 2015. Predictive Dynamic Thermal and Power Management for Heterogeneous Mobile Platforms. In *Proceedings of the 2015 Design, Automation & Test in Europe Conference & Exhibition (DATE '15)*. EDA Consortium, 960–965.

[Skadron et al.(2003)] Kevin Skadron, Mircea R. Stan, Wei Huang, Sivakumar Velusamy, Karthik Sankaranarayanan, and David Tarjan. 2003. Temperature-Aware Microarchitecture. In *Proceedings of the 30th Annual International Symposium on Computer Architecture (ISCA '03)*. Association for Computing Machinery, 2–13. https://doi.org/10.1145/859618.859620

[TI Inc.(2011)] TI Inc. 2011. *Understanding Thermal Dissipation and Design of a Heatsink*. TI Inc. Application Report – SLVA462.

[TI Inc.(2017)] TI Inc. 2017. *Thermal Design Guide for DSP and ARM Application Processors*. TI Inc. Application Report – SPRABI3B.

[TI Inc.(2018)] TI Inc. 2018. *AM572x Thermal Considerations*. TI Inc. Application Report – SPRAC53A.

[Vaddina et al.(2017)] Kameswar Rao Vaddina, Florian Brandner, Gerard Memmi, and Pierre Jouvelot. 2017. Experimental energy profiling of energy-critical embedded applications. In *2017 25th International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*. 1–6.

[Vaddina et al.(2021)] Kameswar Rao Vaddina, Laurent Lefevre, and Anne-Cécile Orgerie. 2021. Experimental Workflow for Energy and Temperature Profiling on HPC Systems. In *2021 IEEE Symposium on Computers and Communications (ISCC)*. IEEE, 1–7.