Thermal Characterization of Next-Generation Workloads on Heterogeneous MPSoCs

Arman Iranfar*, Federico Terraneo†*, William Andrew Simon*, Leon Dragić‡, Igor Piljić‡,
Marina Zapater*, William Fornaciari†, Mario Kovač‡, David Atienza*

*Embedded Systems Laboratory (ESL), Swiss Federal Institute of Technology Lausanne (EPFL), Switzerland
{arman.iranfar, william.simon, marina.zapater, david.atienza}@epfl.ch
†Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Italy
{federico.terraneo, william.fornaciari}@polimi.it
‡Faculty of Electrical Engineering and Computing, University of Zagreb, Croatia
{leon.dragic, igor.piljic, mario.kovac}@fer.hr

Abstract—Next-generation High-Performance Computing (HPC) applications need to tackle outstanding computational complexity while meeting latency and Quality-of-Service constraints. Heterogeneous Multi-Processor Systems-on-Chip (MPSoCs), equipped with a mix of general-purpose cores and reconfigurable fabric for custom acceleration of computational blocks, are key in providing the flexibility to meet the requirements of next-generation HPC. However, heterogeneity brings new challenges to efficient chip thermal management. In this context, accurate and fast thermal simulators are becoming crucial to understand and exploit the trade-offs brought by heterogeneous MPSoCs. In this paper, we first thermally characterize a next-generation HPC workload, the online video transcoding application, using a highly-accurate Infra-Red (IR) microscope. Second, we extend the 3D-ICE thermal simulation tool with a new generic heat spreader model capable of accurately reproducing package surface temperature, with an average error of 6.8% for the hot spots of the chip. Our model is used to characterize the thermal behaviour of the online transcoding application when running on a heterogeneous MPSoC. Moreover, by using our detailed thermal system characterization we are able to explore different application mappings as well as the thermal limits of such heterogeneous platforms.

I. INTRODUCTION

For years, traditional mainstream High-Performance Computing (HPC) applications, have focused on providing maximum performance, understood as raw speed. However, next-generation HPC applications in the fields of medical imaging, video processing and big-data analytics are increasingly faced with Quality-of-Service (QoS) requirements, and their correctness depends not only on performance, but also on meeting latency constraints. Among them, online video services pose an interesting challenge due to their increasing traffic, memory requirements and computational demand. By 2021, global IP traffic is projected to reach 3.3 zettabytes/year [1], with video services representing 80 to 90% of the share [2], and streaming platforms such as Netflix and YouTube accounting for over 50% of downstream traffic [3].

Current streaming services do not provide video types able to match the great variety of devices accessing multimedia content under diverse network conditions, which often results in a waste of computational resources. Online video

transcoding addresses this challenge by serving each client with a video in the required format and resolution. Within this context, the usage of the High Efficiency Video Coding (HEVC) standard, which provides twice the compression of its predecessors [4] while maintaining the same video quality, is mandatory to satisfy large resolutions. As a result, however, HEVC exhibits a dramatic increase in the complexity, computational demand and power requirements [5]. The escalating quest for power/performance efficiency, exacerbated by the stagnation of Dennard's scaling, requires deep application-based customization of the underlying computing architecture, which can only be tackled via heterogeneous multiprocessor systems-on-chip (MPSoCs).

However, the use of heterogeneous MPSoCs with reconfigurable logic (such as the Zynq-7000 SoC, equipped with ARM cores and an FPGA) brings new challenges to efficient power and thermal management in next-generation HPC. In conventional MPSoCs, the knowledge of the chip floorplan allows to identify at design time the location of the hot spots. Thus, thermal sensors can be placed to provide overheat protection and adequate dynamic thermal control (e.g., via DVFS). However, this approach is no longer possible in heterogeneous MPSoCs equipped with reconfigurable fabric, as the thermal distribution, hot-spots and gradients also depend on the accelerators implemented, which are unknown during the design phase. As a result, energy efficiency comes at the cost of shifting thermal evaluation from the chip design phase to the application designer and the run-time management. Moreover, although current heterogeneous MPSoCs are lower power than conventional HPC processors, and thus have lower thermal dissipation requirements, to effectively utilize space in HPC infrastructures, a large number of such devices are placed close together on the same PCB and server, often with very small heatsinks (or with no heatsink at all), thus making thermal and cooling management a major concern. In this context, accurate, fast and flexible thermal simulators, such as 3D-ICE [6], can help understand the power dissipation requirements of applications, tailoring the heatsink size, airflow and cooling requirements to best utilize HPC infrastructures while keeping cooling costs at a minimum and enabling run-time management.

In this paper we extend the 3D-ICE thermal simulator with

a heat spreader model to obtain accurate temperature maps on heterogeneous MPSoCs, and we use it to assess the hot spots and thermal gradients created by a next-generation HPC workload, the online transcoding application [4], when utilizing the heterogeneous resources of the system (i.e., ARM cores and FPGA) under different DVFS settings. Our model can be used to simulate conventional thermal dissipation schemes, where a heat sink is placed atop the chip, as well as heatsink-less conditions where the package is directly exposed to the airflow within a server.

Our contributions in this work are as follows:

- We extend the 3D-ICE thermal simulator tool with a generic heat spreader model that can accurately simulate plastic, ceramic, and metal packaged heterogeneous SoCs by employing a uniform grid, thus making it possible to accurately simulate thermal gradients across the entire surface.

- Using the heat spreader model, we perform a thermal characterization of the online transcoding application when utilizing the various resources of a heterogeneous platform (the Zynq-7000 SoC), and evaluate the trade-offs encountered in terms of hot spots, average temperature and gradients.

- We validate our 3D-ICE model by gathering real power measurements and obtaining temperature maps via a highly-accurate Infra-Red (IR) microscope, obtaining an average error of 6.8%.

The remainder of this paper is organized as follows. Section II reviews the related work. Section III presents our application case study, whereas Section IV describes the proposed model. Experimental setup and results are presented and discussed in Sections V and VI, respectively. Finally, the main conclusions are drawn in Section VII.

II. RELATED WORK

Heterogeneous MPSoCs pose an important challenge from a thermal perspective with respect to homogeneous MPSoCs because of their inherent imbalance in power and thermal profiles [7]. Thermal gradients across the chip deteriorate the system reliability and degrade its performance in several ways. In particular, power-saving techniques such as DFVS or core turn-off may eventually lead a reduction of the mean time to failure (MTTF) of the whole system due to thermal cycling [8]. For metallic structures, when a thermal cycle amplitude increases from 10^oC to 20^oC, the lifetime reliability may decrease up to 16 times [9]. In addition, performance is highly affected by the thermal profile. For instance, the speed of an inverter may drop by more than 35% if working at 110^oC rather than 60^oC [10].

Together, performance and reliability issues make joint power and thermal management crucial and inevitable for modern MPSoCs, leading to development of both design-time [11] and run-time [12] management strategies. Efficient thermal management, however, is not possible without sufficiently accurate knowledge about the thermal profile of the chip in both steady and transient states. To tackle this challenge some approaches make use of RC-based thermal simulators [13],

[6], thermal guns or infra-red thermography [14], performance counters [15], or thermal sensors [16].

Among these techniques, thermal simulation of integrated circuits is a well-studied research topic as it enables not only thermal characterization but also architectural exploration.

Several steady state thermal simulators have been proposed that produce a thermal map given its power dissipation [17], [18]. However, steady state simulators are unable to capture the thermal dynamics that occur in real MPSoCs with time-varying power consumption. To overcome that issue, transient thermal simulators were introduced. One of the most widely-used transient simulators is HotSpot [13].

HotSpot models MPSoCs coupled to a traditional heat dissipation stack composed of a thermal interface material, a heat spreader and a heatsink with an optional fan. The ability to perform transient simulations while receiving power data from an MPSoC simulator allows it to be used as a component in complete simulation flows [19]. HotSpot models the heat spreader and heatsink using a non-uniform grid that is more fine-grained at the center in order to match the chip layers. As a result, the heat spreader simulation is less precise due to the inability to capture thermal gradients in the spreader sides. Moreover, the heat dissipation stack is fixed and cannot be modified without performing deep modifications to its source code. For this reason, it is difficult to simulate heterogeneous MPSoCs directly exposed to forced heat convection without a heatsink.

On the other hand, 3D-ICE [6] is an open-source transient thermal simulator primarily designed to simulate 3D chips, although it can also simulate conventional 2D chips. Moreover, given that it is designed to enable architectural exploration, it can simulate innovative heat dissipation strategies such as liquid cooling through microchannels etched directly in the back of the silicon die. In this work, we extend 3D-ICE with a generic heat spreader model, making it possible to accurately simulate thermal gradients across the entire surface. In this sense, we allow not only architectural exploration, but also cooling exploration for heterogeneous 2D and 3D MPSoCs, overcoming the limitations of other state-of-the-art simulators.

III. CASE STUDY: ONLINE VIDEO TRANSCODING

Efficient just-in-time online video transcoding is an extremely computation-intensive, highly-configurable, multi-stage process bound by strict timing requirements, and is thus an ideal case-study for the QoS-aware HPC.

In this paper online video transcoding application uses the novel High Efficiency Video Coding (HEVC/H.265) [4]. Efficient video transcoding requires significant work on modeling, mapping and optimizing parts of the algorithms to different underlying architectural elements. Due to its high configurability, a considerable number of trade-offs need to be considered to achieve the desired QoS or to reach expected performance. Software optimizations are required but not sufficient, and the use of the reconfigurable fabric in the shape of hardware accelerators for critical parts of the algorithm is mandatory [20] to enable efficient processing from the performance, power and QoS perspective.
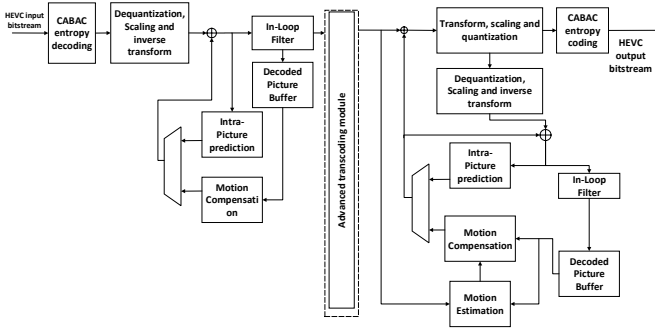
Fig. 1. Video transcoder block diagram

Fig. 2. Percentage of time used by the different phases of the transcoding application when encoding a frame

Figure 1 illustrates a simplified HEVC transcoder block diagram. Each block represents a stage in the encoding process, where most blocks can be configured using different parameter sets (i.e., configuration knobs) that impact main characteristics of the transcoding process such as computational complexity, coding efficiency and video quality. Balancing between these three characteristics in real-time presents a great challenge and is often considered as a critical consideration in video codec designs. Optimization of transcoding algorithms is necessary to efficiently utilize the processing power distributed across the various processing cores on a heterogeneous MPSoC. Data flow between the memory components and heterogeneous processing units also needs to be considered to ensure that frame processing time does not exceed the maximum period to ensure a smooth playback.

Identification of key kernels that would benefit the most from custom hardware implementation (i.e. from HW acceleration) is the first step towards exploiting heterogeneity. Profiling of the video transcoding application and obtaining its task call graph has been performed to understand which functions and kernels require higher computational effort or introduce bottlenecks in the application lifecycle. Some parts of the transcoding algorithm, such as motion compensation, motion estimation (ME) and transformation, and consequently kernels included in those algorithms, were identified as ideal candidates for hardware acceleration. Even though in this paper we tackle the Zynq-7000 SoC (as detailed in Section V), this exercise has been conducted in both the ARM cores of the application when encoding a frame in a heterogeneous Zynq-7000 SoC, and in an x86, obtaining similar results in terms of relative percentage of time used for each function.

A sample of our results are displayed in Figure 2, which summarizes the percentage of time spent in each function for a particular encoding configuration (Quantization Parameter of 37, usage of Inter-Picture prediction with a Search Area of 10) and video input (Old Town Cross video, 50fps, 1280x720) of the online transcoding application.

As this figure shows, the *interpolation_C1dx0* function and its utility functions *clip*, *Sum of Absolute Differences (SAD)*, and *FetchBlockFromReferenceFrame* account for 93.2% of processing time taken to encode the frames in question. These functions are a part of the Motion Estimation block seen in Figure 1. For a frame with a resolution of 1920x1080 pixels and with the given encoding configuration, the interpolation function will have to be called 228000 times. Given the
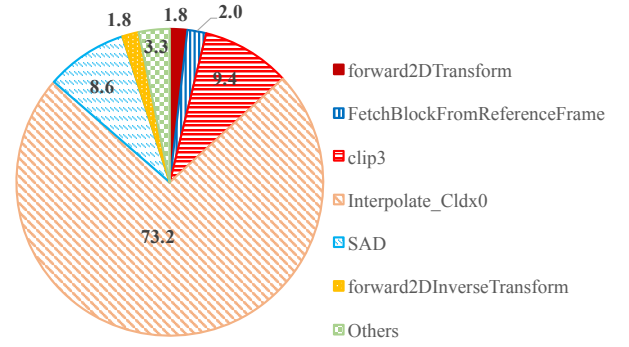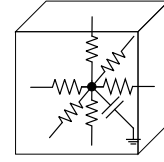

Fig. 3. A single volume as simulated by the 3D-ICE thermal simulator. Resistors account for heat conduction to neighbor volumes.

extreme computational power needed for ME, several parallelization strategies have been proposed in previous works, for example through a CPU plus GPU platform[21], and on ASIC[22]. However, the results of these papers do not include analysis of the temperature and power costs of the proposed solution. A hardware DMA/accelerator for inter-pixel ME is proposed in Section V. The detailed function and particular implementation of this accelerator is not in the scope of this paper and will be covered in more detail in future work.

IV. THERMAL MODEL

In order to model the thermal dynamics of heterogeneous MPSoCs, we extend the 3D-ICE [6] thermal simulator. 3D-ICE uses the finite difference method to approximate the differential equations governing heat transfer inside a chip. The chip structure is composed of the active silicon layer and the silicon bulk and is divided in a grid of volumes of uniform properties. Each volume of the grid, as shown in Figure 3, is modeled through a thermal capacitance, and up to six thermal resistances accounting for the heat transfer to adjacent volumes. This discretization produces a linear system that can be efficiently solved even for a large number of volumes thanks to its sparse nature.

3D-ICE was mainly developed to enable architectural and cooling exploration and simulate liquid cooling of 3D-ICs through microchannels etched between layers. For this reason, the chip structure is not fixed, and can be configured by defining a number of layers that are stacked atop of each other, therefore providing great flexibility to model existing MPSoCs. However, one limitation is that all layers must have the same dimensions, except for the height. This makes modeling a heat spreader as an ordinary layer in the stack difficult, as it is not possible to take into account its larger dimensions compared to the silicon die. This mismatch causes modeling errors. For
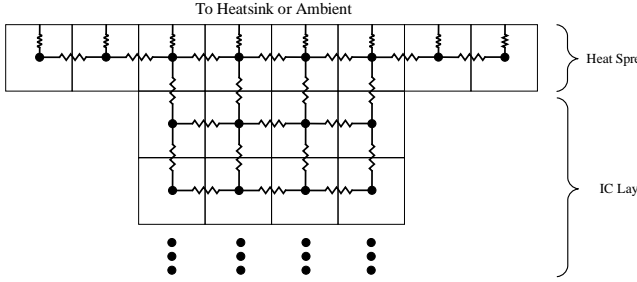
Fig. 4. The proposed generic heat spreader model seen from a transversal view. Each node has a thermal capacitance that is not drawn to improve readability.

the same reason, modeling convective heat dissipation through the top of the IC package is also challenging.

In this work, 3D-ICE has been extended with a generic heat spreader model, which is an additional layer at the top of the stack whose dimensions are not bound to the ones of the silicon die. Figure 4 shows the new heat spreader model connected to the layer stack. Contrary to the HotSpot [10] model, the generic heat spreader model in 3D-ICE employs a uniform grid, thus being able to accurately model lateral heat spreading and can both simulate the case where the IC has no heatsink and direct convective heat dissipation occurs between the top package surface and air, or be connected to a heatsink.

The presented model is validated in Section VI by comparing the thermal maps provided with 3D-ICE with real maps obtained using an Infra-Red microscope, using the setup explained in Section V.

V. EXPERIMENTAL SETUP

A. Online transcoding on a heterogeneous MPSoC

In this work, we perform our experiments on a Parallella board[23] equipped with a Zynq7000 SoC, type Z-7010. The chip comprises of a dual core Cortex-A9 ARM processor with 32KB L1 data per core, and 512KB shared L2. The maximum recommended processor frequency is 667MHz to maintain the temperature at safe values without a heatsink, however, the cores can be clocked up to 1GHz. The chip also contains FPGA fabric consisting of 28K logic blocks and 3.8 Mb of BRAM. We clock the totality of the FPGA fabric at 100MHz. The board comes with a 1GB DDR3 RAM chip clocked at 533MHz, and a 16GB microSD card as primary storage. The board's Epiphany co-processor is unused in this setup.

The online transcoding application used is an in-house implementation written in C that we run on Parabuntu, an Ubuntu 15.04 special distribution for Parallella boards, as a testbed for our experiments. Code profiling is performed by using the statistical profiler OProfile[24] and the callgraph generation tool Valgrind[25] to locate the most critical, time consuming functions within the code, obtaining the sample results illustrated in Figure 2.

As a proof-of-concept for efficient thermal management in a heterogeneous platform, in this paper we focus on FPGA acceleration of the *interpolate* function, which is the main block of ME. This hardware block was designed and implemented

in Vivado 2016.1[26]. The accelerator communicates with the ARM processor and DDR3 memory over an AXI bus. The size limitations of the FPGA fabric on the Z-7010 chip mean that we are only able to implement one accelerator. One accelerator is sufficient for the goals of this paper. However, the design is easily scalable to larger FPGA boards.

To perform interpolation and ME, the calling C function passes pointers to the memory location of the relevant pixel data for the pixel block on which ME is to be performed, as well as the block size and frame limits of the block if it is located on the edge of the frame. The accelerator then reads in the block pixel data of the current and previous frame and performs 16 interpolations and ME calculation in parallel. The *clip* and *SAD* functions are inherently performed in the DMA. Our modified application supports multiple instances of code sharing the same accelerator module through the use of semaphores, allowing us to run multiple encoding processes simultaneously, which is interesting for online transcoding scenarios.

In order to benchmark the temperature and power consumption of our architecture, we run the encoding application using multiple configurations. On the OS level, we run the system at multiple core clock frequencies (from 333MHz to 1GHz), run one or two instances of the application on the dual core processor (1c vs. 2c), and run the application with and without FPGA acceleration (ACC). Power measurement is performed using precision multimeters to log voltage and current traces for the rails supplying power to the Zynq chip, according to the schematic datasheet [27].

B. Real measurement of temperature maps

Temperature benchmarking is performed in software via an internal die sensor and also through the use of an Infra-Red (IR) microscope, able to provide very accurate temperature measurements [28]. Since most of the contrast in an unprocessed infra-red image is due to emissivity (a material property that influences how efficiently the material emits photons) rather than temperature, the IR-microscope measures and compensates for emissivity of the material in order to generate an accurate, calibrated thermal map image.

The IR-microscope used in this work is able to capture thermal map images at a maximum frequency of 53.42 frames/sec and total number of 2000 frames which makes it suitable for transient analysis. We use this maximum sampling frequency in this paper. Moreover, the microscope resolves temperature differentials of 0.1^oC and provides spatial resolution down to 0.2 μm. Figure 5 shows the IR microscope used to gather temperature maps of the Zynq-7000 SoC.

VI. RESULTS

A. Heatsink validation

In order to validate the heat spreader model of 3D-ICE, we compare the temperature map at the heatsink provided with 3D-ICE with the one measured with the IR camera. To configure the input parameters of 3D-ICE we use a rough floorplan of the die derived from Vivado and the thermal map images obtained from the IR-microscope. In addition, we use the packaging and pinout datasheet[29] of the Zynq SoC to
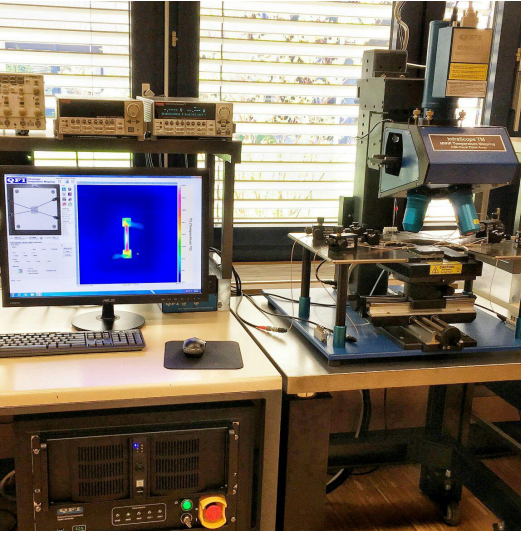
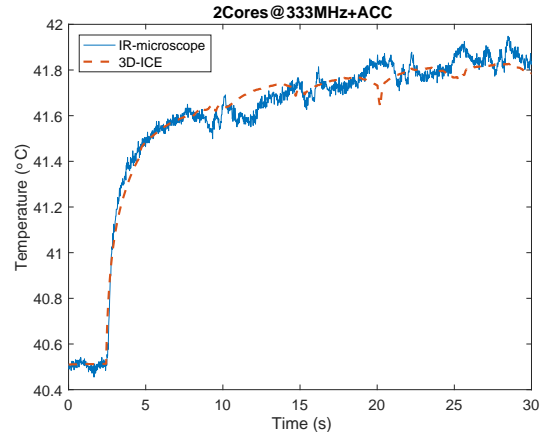Fig. 5. Infra-Red microscope used for obtaining MPSoC temperature maps

Fig. 6. Comparison between the simulated and measure temperature of the hot spot located at core #1, when running the application in 2 cores at 333MHz, and accelerators

apply the stack layer of the CLG400 packaging. The ambient temperature for all experiments is 24^oC.

Figure 6 shows the transient temperature over the heat spreader simulated by 3D-ICE and the one obtained from the IR-microscope, corresponding to the hot spot located on the first core, when the transcoding application is running on the Zynq SoC using both cores at 333MHz along with the accelerators. The average error compared to the real measurements is less than 0.04^oC, i.e., less than 2.7%. This error is thought to mainly come from the limited knowledge about the precise floorplan and the power distribution over the whole die. Figures 7 and 8 show the thermal map obtained from the IR microscope and 3D-ICE, respectively, indicating sufficiently accurate floorplanning as well as the reliable heat spreader model of 3D-ICE. Finally, Table I shows the maximum and average error achieved by 3D-ICE in comparison to real measurements for 8 different scenarios studied in this work. For instance, *"2c@333+ACC"* implies using 2 cores at 333MHz with accelerators. As it can be observed, the average error ranges from 2.7% to 10% for different test cases, which is a very small error for state-of-the-art tools using RC-based models.

B. Thermal exploration on heterogeneous MPSoCs

Thermal considerations are crucial when designing a reliable MPSoC. 3D-ICE, as a fast and accurate thermal simulator, facilitates chip design for manufacturability. In particular, the new heat spreader model, providing thermal maps, maximum temperature and chip gradients under varying workloads, can be used for design-time exploration on packaging and cooling techniques. This is especially important for heterogeneous MPSoCs where the imbalance in power consumption leads to uneven thermal maps, and exacerbated hot spots and gradients.

Moreover, this new feature of 3D-ICE, which allows to plug-in any external heatsink device, facilitates testing cases that cannot be evaluated in real life. In particular, many MPSoCs come with heatsinks that cannot be removed due to temperature-protection mechanisms, or have specific firmwares
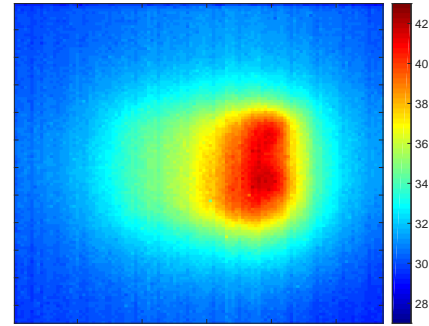

Fig. 7. Thermal map of the Zynq SoC measured by the IR-microscope, when running on 2 cores at 333MHz, and accelerators.
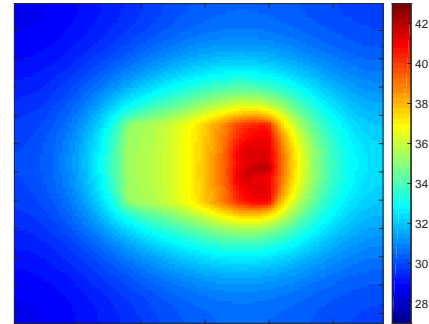

Fig. 8. Thermal map of the Zynq SoC achieved by 3D-ICE, when running on 2 cores at 333MHz, and accelerators.

that force the chip to be over-cooled, e.g., by shutting down the system once a predefined threshold non-configurable by the user, is reached. This is the case of our target system, the Zynq SoC, which comes with an on-chip temperature protection that forces a shutdown on the system when the die temperature sensor reaches 60° C. Without the heatsink, the die reaches around 57.4^o C when running both ARM cores at 666MHz. Even though 60° C is far from causing a reliability issue on the chip, the Zynq SoC is strictly over-cooled and does not allow running workloads at 800MHz or 1GHz without a heat-sink, thus preventing further thermal exploration.

Our validated heat spreader model for 3D-ICE, however,

TABLE I.   TEMPERATURE ERROR OBTAINED FOR DIFFERENT SCENARIOS

Error	2c@333+ACC	1c@333+ACC	2c@333	1c@333	2c@666+ACC	1c@666+ACC	2c@666	1c@666
Max	**10%**	**16%**	**15%**	**18%**	**7.2%**	**11%**	**11%**	**15%**
Average	**2.7%**	**7.5%**	**7.8%**	**6.6%**	**6.4%**	**7.0%**	**6.5%**	**10%**

TABLE II.   TEMPERATURE RESULTS (° C) SIMULATED WITH 3D-ICE FOR A FREQUENCY OF 1GHZ

	2c@1000+ACC	1c@1000+ACC	2c@1000	1c@1000
Package Max. Gradient	**17.5**	**16.2**	**17.2**	**16.1**
Package Max. Temperature	**55.5**	**52.4**	**54.7**	**51.9**
Die Max. Gradient	**17.2**	**15.9**	**16.8**	**16.0**
Die Max. Temperature	**62.9**	**59.2**	**61.8**	**58.9**

addresses all these issues and facilitates thermal-aware design. It allows us to further characterize the online video transcoding on simulation space with 3D-ICE by running application at 1GHz. Thus, we collect the power values of the board when running with its heatsink, and we plug the power traces on 3D-ICE, to obtain the temperature maps attained when running without heatsink. The results of these simulated test cases are shown in Table II. This table shows that the maximum temperature attained by the die is of $62.9°$ C. Although this temperature is enough for rebooting the system, still it is far from the critical temperature at which reliability failure mechanisms would start [30]. This arises doubts about using a heatsink or a heatsink and a fan atop for the studied application and the heterogeneous SoC. This is, however, on the contrary to the recommendations suggested by the Parallella board manual, where even using a fan is highly recommended which results in higher power consumption of the whole board.

VII.   CONCLUSION

In this paper we have extended the 3D-ICE thermal simulator with a generic heat spreader model to enable architectural and cooling exploration on heterogeneous MPSoCs. Our model has been validated with an Infra-Red microscope, achieving temperature errors below $0.15°$ C (less than 7%) for the hot spots of the chip. Thanks to our model we have been able to undertake an extensive characterization of a reconfigurable heterogeneous MPSoC (the Zynq7000 SoC), proving the reliable operation of the board under worst-case thermal conditions, and the limited benefits for less energy-efficient fan-based cooling. Moreover, our work provides a new methodology for the accurate exploration of both architectural and cooling designs in generic heterogeneous MPSoCs. We plan to extend our work by providing more generic interfaces to the heat spreader model developed, creating generic interfaces that will allow to simulate arbitrary cooling mechanisms, ranging from conventional heat-sinks to heat-sinks with fans, as well as pluggable liquid-cooling devices.

REFERENCES

[1] Cisco Systems, Inc., "The zettabyte era: Trends and analysis. cisco whitepaper." 2017.

[2] ——, "Cisco visual networking index: Forecast and methodology 2015-2020. cisco whitepaper." 2016.

[3] Sandvine, Inc., "Global internet phenomena report," 2013.

[4] V. Sze *et al.*, *High Efficiency Video Coding (HEVC): Algorithms and Architectures.* Springer Publishing Company, Incorporated, 2014.

[5] F. Bossen *et al.*, "Hevc complexity and implementation analysis," *IEEE TCSVT*, vol. 22, no. 12, pp. 1685–1696, 2012.

[6] A. Sridhar *et al.*, "3d-ice: Fast compact transient thermal modeling for 3d ics with inter-tier liquid cooling," IEEE Press, pp. 463–470, 2010.

[7] S. Sharifi *et al.*, "Hybrid dynamic energy and thermal management in heterogeneous embedded multiprocessor socs," IEEE Press, pp. 873–878, 2010.

[8] A. K. Coskun *et al.*, "Analysis and optimization of mpsoc reliability," *Journal of Low Power Electronics*, vol. 2, no. 1, pp. 56–69, 2006.

[9] ——, "Temperature management in multiprocessor socs using online learning," IEEE, pp. 890–893, 2008.

[10] W. Huang *et al.*, "Compact thermal modeling for temperature-aware design," ACM, pp. 878–883, 2004.

[11] T. Chantem *et al.*, "Temperature-aware scheduling and assignment for hard real-time applications on mpsocs," *IEEE TVLSI*, vol. 19, no. 10, pp. 1884–1897, 2011.

[12] G. Liu *et al.*, "Neighbor-aware dynamic thermal management for multi-core platform," EDA Consortium, pp. 187–192, 2012.

[13] W. Huang *et al.*, "Hotspot: a compact thermal modeling methodology for early-stage vlsi design," *IEEE TVLSI*, vol. 14, no. 5, pp. 501–513, May 2006.

[14] F. F. Farkhani and F. A. Mohammadi, "Temperature and power measurement of modern dual core processor by infrared thermography," IEEE, pp. 1603–1606, 2010.

[15] S. W. Chung and K. Skadron, "Using on-chip event counters for high-resolution, real-time temperature measurement," IEEE, pp. 114–120, 2006.

[16] Y. Zhang and A. Srivastava, "Accurate temperature estimation using noisy thermal sensors," ACM, pp. 472–477, 2009.

[17] P. Li *et al.*, "Efficient full-chip thermal modeling and analysis," Washington, DC, USA, pp. 319–326, 2004.

[18] H. Su *et al.*, "Full chip leakage-estimation considering power supply and temperature variations," pp. 78–83, Aug 2003.

[19] F. Terraneo *et al.*, "An accurate simulation framework for thermal explorations and optimizations," New York, NY, USA, pp. 5:1–5:6, 2015. [Online]. Available: http://doi.acm.org/10.1145/2693433.2693438

[20] M. Kovač and N. Ranganathan, "Vlsi circuit structure for implementing jpeg image compression standard," Aug. 1997, US Patent 5,659,362. [Online]. Available: http://www.google.com/patents/US5659362

[21] X.-w. Wang *et al.*, "Paralleling variable block size motion estimation of hevc on multi-core cpu plus gpu platform," IEEE, 2013.

[22] Z. Guo *et al.*, "An optimized mc interpolation architecture for hevc," IEEE, 2012.

[23] [Online]. Available: https://www.parallella.org/board/

[24] [Online]. Available: oprofile.sourceforge.net

[25] [Online]. Available: http://valgrind.org/

[26] [Online]. Available: www.xilinx.com/products/design-tools/vivado

[27] [Online]. Available: www.parallella.org/docs/parallella_schematic.pdf

[28] [Online]. Available: http://www.quantumfocus.com/

[29] [Online]. Available: https://www.xilinx.com/support/documentation/user/ug865-Zynq-7000-Pkg-Pinout.pdf

[30] J. R. Black, "Electromigration failure modes in aluminum metallization for semiconductor devices," *Proc. of the IEEE*, vol. 57, no. 9, pp. 1587–1594, 1969.