EAI Endorsed Transactions
on Industrial Networks and Intelligent Systems Research Article **EAI.EU**

A Survey of System Level Power Management Schemes in the Dark-Silicon Era for Many-Core Architectures

Emmannuel Ofori-Attah[1], Xiaohang Wang[2], Michael Opoku Agyeman[1,*]

[1]Department of Computing, University of Northampton, United Kingdom
[2]South China University of Technology, 1121 Haibin Road, Nansha, Guangzhou

Abstract

Power consumption in Complementary Metal Oxide Semiconductor (CMOS) technology has escalated to a point that only a fractional part of many-core chips can be powered-on at a time. Fortunately, this fraction can be increased at the expense of performance through the dark-silicon solution. However, with many-core integration set to be heading towards its thousands, power consumption and temperature increases per time, meaning the number of active nodes must be reduced drastically. Therefore, optimized techniques are demanded for continuous advancement in technology. Existing efforts try to overcome this challenge by activating nodes from different parts of the chip at the expense of communication latency. Other efforts on the other hand employ run-time power management techniques to manage the power performance of the cores trading-off performance for power. We found out that, for a significant amount of power to saved and high temperature to be avoided, focus should be on reducing the power consumption of all the on-chip components. Especially, the memory hierarchy and the interconnect. Power consumption can be minimized by, reducing the size of high leakage power dissipating elements, turning-off idle resources and integrating power saving materials.

Received on 10 August 2018; accepted on 03 September 2018; published on 19 August 2018

Keywords: Dark-Silicon, Many-Core, Multi-Core, NoC, Low Power NoC Architectures, Power Budgeting

doi:10.4108/eai.19-9-2018.155569

1. Introduction

Aggressive transistor scaling with technology has fuelled an unprecedented growth in the number of Processing Elements (PE) available in modern Systems-on-Chip (SoCs). However, due to the excess thermal issues caused by the breakdown of Dennard's scaling, multi-core/many-core chips do not scale properly with die area. Continues scaling of the chip size aggravates the total power consumption and hence to meet the systems power budget, only a subset of nodes can be powered-on while the rest are powered-off (dark). To make things worse, researchers have already estimated that in the near future, 50% of a chip size at 8 nanometer (nm) technology will be powered-off. This implies that only half of the applications that are currently being executed in many-core chips will be executed in the future at a time. The industrial approach to this solution is the fabrication of processor chips, designed to work within a thermal design constraint to prevent possible overheating and permanent damage. Unfortunately, as a trade-off, this solution prevents peak frequency level operation and thus novel techniques are needed to maximise the chip's performance.

One possible solution for this challenge is application mapping, where specific nodes are selected for incoming applications. Unfortunately, prior works only focus on distributing application tasks in different regions on the chip without considering the performance of the applications. Another alternative solution is through Dynamic Thermal Management (DTM) techniques such as power-gating, Dynamic Voltage Frequency Scaling (DVFS) and Task migration. Unfortunately, this also

[*]Corresponding author: Michael.OpokuAgyeman@northampton.ac.uk

trades-off application performance to satisfy the temperature threshold by scaling the Voltage Frequency (V/F) levels of the cores. Nonetheless, there have been many works and techniques proposed through the dark-silicon solution, however only a few consider the power consumption of on-chip components such as the memory hierarchy and the interconnect. Recently, Multi-Level Caches (MCA) and the Network-on-Chip (NoC) paradigm have replaced single-level caches and buses respectively as the standard components for many-core future chip designs [1–4]. However, these components increase the power consumption and impacts heavily on the temperature of the chip. In fact, the Last-Level-Caches and NoC in a 16-core machine [5] constitutes to 33% of the total power consumption. Therefore, to avoid high temperature, a reduction of power consumption in these elements is very essential.

The remainder of this paper is organised as follows. Section II discuss the causes of power consumption. Section III introduce techniques for DCSCs while Section VI discusses the influence that the interconnect and memory sub-system have on the chip's total power. Section V summarises all the techniques presented and finally, the conclusion is drawn in Section VI.

2. Background

For decades, Moore and Dennardian theories were the embodiment of exa-scale technology. Dennard's scaling revealed that, by reducing the size of transistors, it can be utilised at lower power and voltage because, power density is equivalent to the square of applied voltage, therefore, it remains the same [6–8]. Consequently, by reducing the physical parameters of transistors, it has been possible to utilise them under lower power and voltage. Thus, enabling an advent in resource duplication for performance enhancement resulting in the multi-core/many-core technologies.

Figure 2 depicts the effects of transistor scaling. A chip at 8nm with all its node activated will cause the chip's temperature to be very high because of high of power consumption. To prevent this, the dark-silicon solution permits fractions of the chip to be powered-off. By switching parts of the chip off, the leakage power consumption reduces as well as the dynamic power consumption. Power consumption materialises as a subset of dynamic and leakage power in CMOS integrated circuits. Until the deep-submicron processes emerged, dynamic power consumption was held accountable for majority of the power consumption in CMOS technology. Unfortunately, transistor size reduction has caused a halt in voltage scaling down and resulted with an increase in the amount of sub-threshold leakage, as well the gate tunnelling leakage current caused by having thinner gate oxides. Therefore, it has been reported that leakage power contributes to a higher percentage of the chip's total power consumption at the deep-sub micron level [8–10]. Consequently, high power density generates excess heat and increases the temperature of the chip. The consequence of such peak temperature is overheating, permanent damage, transient faults and faster ageing [11, 12]. Therefore, to reduce power consumption at the transistor level, designers adjust the equations above. Dynamic power consumption can be reduced by scaling the V/F level and reducing the activities. Leakage power consumption on the hand is reduced by utilizing low power cells or reducing the number of active transistors. An example of such a technique is the dark-silicon. Where, every chip is allocated a Thermal Design Power (TDP) for the chip to operate with.

3. Dark-Silicon: The Future For Emerging SoC Designs

Unfortunately, the TDP provided by the industries only allow DSCSs to operate at a feasible power budget to keep the thermal profile of the chip down. Thus, limiting them from operating at high V/F levels. Nonetheless, Intel's Turbo Boost [13] and AMD's Turbo CORE [14] violate the TDP constraint during short intervals by boosting the system for higher performance. When the threshold is violated, DTM techniques are used to cool down the chip. The result of such an action is performance degradation. Therefore, such techniques have to be used appropriately. Particularly, Task migration and DVFS.

3.1. Task Migration

Task migration is an optimized technique used to migrate tasks between nodes that are dissipating high temperature. For example, if a task causes a node to generate excess heat thus raising the temperature, that task is then migrated to a node in exchange for another task with low temperature. This is done to avoid excess heat which can have a negative effect on neighbouring tasks. Unfortunately, this technique only applies when some nodes are executing heavy loaded tasks. If all the nodes are executing heavy loaded tasks, task migration will not have any effect. Figure 3 depicts an image of task migration. The temperature of application 1 has been reduced because the task in node 1 has been migrated to node 7 because it has a lower temperature. Unfortunately, in application 3, because all the nodes are executing heavy load tasks, there is no difference in the overall temperature of the chip when the task is migrated to a different node [15].

3.2. Dynamic Voltage Frequency Scaling

DVFS is used to dynamically vary the V/F levels of a node that exhibits high temperature. This technique is

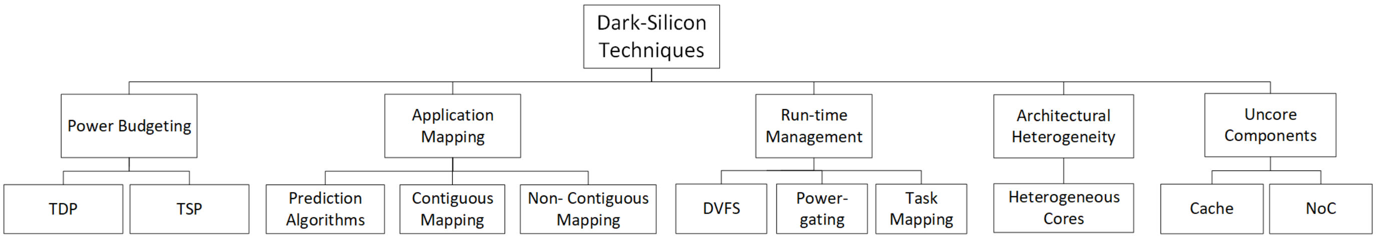


Figure 1. An Overview of Dark-Silicon Techniques

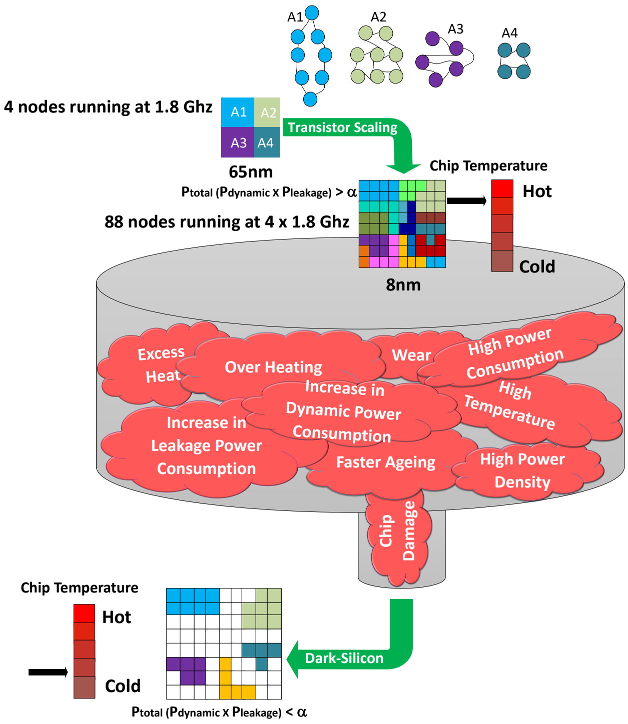


Figure 2. Effects of Transistor Scaling

best used in a Run-Time Management (RTM) system because accommodating newly arrived applications can at sometimes cause an overshoot. During this process, DVFS is used to vary the V/F levels until the application has been successfully mapped. Although, this helps to reduce the temperature of the chip, the tasks in question suffer from performance loss. With such technique, tasks are likely to be executed beyond the deadline time [15].

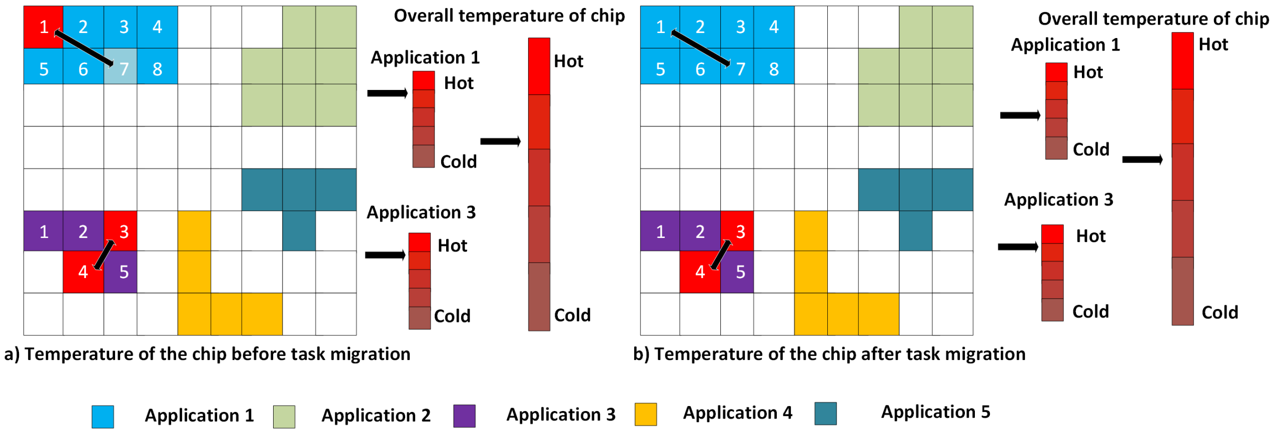
3.3. Thermal Design Power Techniques

For this purpose, Pagani et al. [16] proposed a power budgeting technique for DSCSs to operate at their highest power. Unlike TDP where all the cores are modelled with one power value, TSP allows different groups of cores to have different power constraints based on the incoming application. Therefore, for every floor-plan, a different power constraint is computed for the worst-case mappings. This is contrary to TDP architectures, where all the cores are judged to be functioning in their worst-case V/F. TSP worst case mapping is computed based on the number of active cores, their position, and the influence of neighbouring cores temperature. Therefore, TSP is considered as the most optimized thermal constraint because the amount of power the chip can operate on, is based on core alignment which is determined by application mapping [17, 18].

Similarly, Wang et al. [15] also introduced a new power budgeting technique for DSCSs. The proposed power budgeting technique advocates the number of cores that needs to be activated, as well as selecting the maximum power that every core can consume based on the current thermal profile of the chip. In addition to this, the proposed technique uses a model prediction method to generate a power budget for the chip for future mappings.

3.4. Run-Time Management Systems

The purpose of a RTM is to monitor the power budget, reserve idle cores and allocate them to applications. In case there is not enough power available for an incoming application, the application is halted until an executing application leaves the system. However, due to the dynamic nature of workloads, the number of core count available for applications may change, depending on the requirement of an application [19]. This can result in a change of layout, increase in power consumption and deadline time for executing tasks. A biased RTM will result in impecunious resource allocation limiting the maximum achievement of the system. In a such a system, more stress is placed on regions where applications can be executed faster. Therefore, a run-time management system which incorporates design factors such as the layout of the processor chip, heterogeneity, uncore components (NoC and cache), architecture (2D, 3D and Wireless), temperature of the system and TDP/TSP power budget



a) Temperature of the chip before task migration

b) Temperature of the chip after task migration

Figure 3. An Example of Task Migration

[16]. In addition to this, the RTM needs to consider which DTM techniques to use for specific applications.

Run-time Management System Algorithms. Rahmani et al. [20] propose a dynamic power management with a multi-objective approach for NoC based dark-silicon many-core platforms. The proposed management system utilises per-core power-gating and DVFS based on the following characteristics: workload, network congestion and the power performance of the cores. The management system incorporates the following to measure the characteristics of the system: Application Power Calculator, Application Processor Utilization Calculator, Application Buffer Utilization Calculator, Application Injection Rate Calculator, a TSP Lookup Table, and Proportional Integral Derivative Controller. There are four algorithms which can be activated in the system.

The first algorithm dynamically scales the V/F Level by monitoring various feedback from the system. The second algorithm downscale the applications with the lowest priority when there is an overshoot in the system. Among the lowest priority, the congested applications are chosen to be optimized since congested areas contributes to high power consumption. The third algorithm is used to scale up the applications when there is an undershoot. In particular, priority is given to applications that were previously downscaled, not congested and non-intensive. Since a newly added application can push the power consumption above the TSP/TDP constraint, algorithm four performs the following tasks: when a new application arrives, the algorithm checks the available power budget and determines if the new application can be mapped to nodes. After checking the new application, the algorithm predicts the power consumption of the system when the new application is executed.

If the application is likely to cause an overshoot, the currently running applications are scaled down to execute the application. However, if it does not exceed the TSP/TDP constraint, the new application is added without any scaling. The only disadvantage with the proposed RTM is how application tasks are mapped. The dark nodes which are used to cool down the chip are only used to separate applications. Although external heat generated from neighbouring application nodes are minimized, internal heat is ignored. The mapping algorithms could be further enhanced to contain dark nodes inside the region which has been selected. In this way, internal heat is minimized.

Salehi et al. [21] on the hand propose a power-constrained reliability Management System for dark-silicon chips (dsReliM) which considers the reliability of tasks. The model of the system has been categorised into four different parts. (Hardware Architecture, Application Model, Reliability Model, and Power Model). The hardware architecture model of the system consists of heterogeneous cores which can operate at different V/F levels through DVFS. A reliability compiler is utilised in the application model to compile multiple code versions for each application task with properties such as reliability and execution time. The purpose of dsRelim is to execute applications with minimum reliability loss while meeting deadlines. Firstly, the code version with the highest reliability is chosen along with the maximum V/F level. If the selected code exceeds the TDP constraint, the V/F level is gradually adjusted. If the execution task of the system is violated after adjusting the V/F, another version of the task which meets the deadline but with a lesser reliability loss is chosen. However, if there are not any code versions which meets the deadline time, the code with the minimum execution task is chosen with a performance trade-off. The V/F level of the selected code version is scale down to meet the TDP constraint.

Rahmani et al. resolves the reliability problem by introducing a novel power controller unit [22].

The power controller contains an operating mode selector which monitors the workload or intensity of the system and selects the following modes for the system to operate at: overboosting mode and reliability aware. The overboosting mode is selected when there are high intensive applications which requires full system operation without considering the reliability performance. The other mode is reliability-aware. This mode is selected for applications with low priority. Unlike [21], during this mode, the system operates at feasible V/F where thermal hotspots are considered as well as good performance.

Haghbayan et al. [23] also proposed a reliability-aware resource management for many-core systems which prioritises the younger cores than older cores. The proposed solution consists of two units (Reliability Analysis Unit (RAU) and Runtime Mapping Unit (RMU). The RAU monitors the ageing information/status of all the cores. The RMU on other hand, takes into the account the ageing status of the cores provided by the RAU and then the total power consumption of system provided by a power monitor before mapping applications to cores. MapPro [24] is used to locate the first node, however regions with busy cores are ignored during the application mapping stage. Furthermore, during the application mapping, a reliability factor metric is applied to prioritize the selection of younger cores for performance enhancement.

Similarly, Khan et al. [25] presents a hierarchical budget scheme which distributes resource and power budgets based on the system workload for clusters. Firstly, the scheme determines the number of cores required for an application to be executed successfully. For the inter clusters, several factors are used to determine which cluster is allocated more power. One factor that is used is the number of cores in a cluster. Another is the history of an application. For example, an application with a history of requiring high power consumption is allocated more power at the next epoch. For the intra clusters, since different types of threads require different amount of power for execution, in video applications, data tiles which consists of high motion content are allocated more power. Therefore, in the inter clusters, cores are allocated power individually based on their data tile.

Likewise, Yang et al. [26] proposed a run-time management system to handle a scalable hardware topology based on a Quad-core cluster. Quad-core cluster is a tile-based architecture which consists of heterogeneous cores ((High Performance (HP), General Purpose (GP), Power Saving (PA) and Low Energy (LE)) and a shared cache within each cluster. The purpose of having different cores is to utilise them based on the incoming application requirements. For example, the HP cores are used for high workloads and thus consumes the most power while the LE cores do the opposite. Only one core is activated in each cluster to keep the temperature at a minimum. Consequently, each core in the node has various V/F levels which can meet an application's demand. In addition to this, idle cores are turned-off to keep the temperature under the safe value. Furthermore, the active cores are physically decentralised to avoid possible heat dissipation.

3.5. Application Mapping

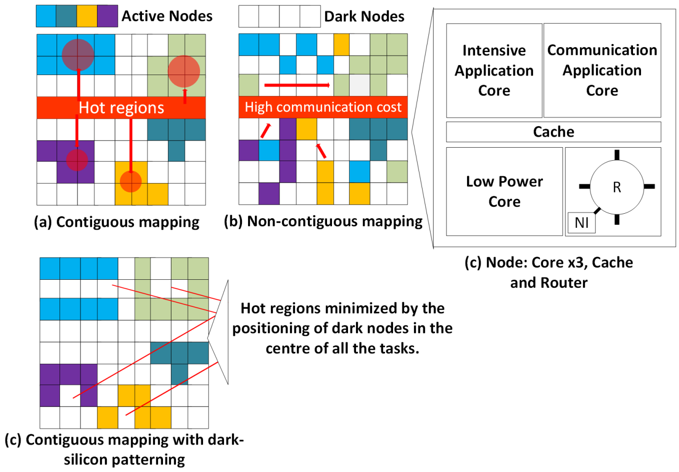
As previously stated, application mapping ensures that specific nodes are selected on the chip for mapping. This can be done in several ways [17, 23, 27–29]. Different mapping algorithms produces different results (temperature effects and power budget). The selection of the correct nodes enables more nodes to be activated to accommodate more applications and run tasks faster.

In DSCSs, application mapping is initiated in two stages. For clarification purposes, we refer to the first stage as region mapping and the second stage as task mapping. Region mapping is the process of finding a particular region on the chip with sufficient nodes available for task mapping. Task mapping on the hand refers to the process of identifying and assigning tasks to preferred nodes from the pool of nodes found using application mapping. The most applied method for region mapping is to find an optimal node and then map task to surrounding nodes to form a rectangular/square shape mapping. In practice, MapPro is used by many [17, 23] to automatically calculate and determine this approach.

Subsequently, existing application mapping algorithms can be categorized into groups. These are contiguous mapping and non-contiguous mapping. Figure 4 shows the impact of contiguous and non-contiguous mapping. Contiguous mapping is the process of activating nodes in one region for an application to be mapped to reduce communication overhead between tasks. Non-contiguous mapping on the hand, assign application tasks to any available node.

In practice, some techniques aim to monitor the temperature of nodes periodically in order to map incoming applications. These techniques predict and estimate the temperature and produce optimized mapping algorithms with minimum chip temperature [30].

J. Wang et al. [28] propose an Ant-Colony (ACO) based thermal-ware thread-to-core mapping. The ACO-based thread-to-core mapping algorithm releases an ant colony into the system. Each ant conducts the thread-to-core mapping individually. After conducting the thread-to-core mapping, the ant with the best minimizing Chip Multi Processors (CMP) peak temperature is chosen. The temperature of the results generated by the


Active Nodes
Dark Nodes
Hot regions
High communication cost
Intensive Application Core
Communication Application Core
Cache
Low Power Core
R
NI
Hot regions minimized by the positioning of dark nodes in the centre of all the tasks.

(a) Contiguous mapping (b) Non-contiguous mapping (c) Node: Core x3, Cache and Router

(c) Contiguous mapping with dark-silicon patterning

Figure 4. Contiguous Mapping and Non-Contiguous Mapping

best ants are then used to map incoming applications to nodes.

Similarly, Wang et al. [27] also proposed a thread-to-core mapping management system. However, this mapping systems uses two different types of virtual mapping algorithms to estimate the performance of applications when different number of dark cores are used. Upon the arrival of a new application, the virtual mapping process is used to estimate the performance of the application with different number of dark cores. In addition to this, the mapping system consists of two different modes: Computational and Communication. Applications which are affected by the task computation performances are mapped as far away from each other as possible. Applications whose performances are affected by their communication volume are mapped closer to each other. With this algorithm, it is possible to migrate a task from core to core to harness the best performance.

Li et al. [30] propose a Mixed Integer Linear Programming (MILP) thermal model which monitors all nodes and map applications while minimizing the temperature. The proposed algorithm works by predicting the temperature of chip when applications are mapped. This approach sorts out all applications into groups and execute them starting from the highest V/F to their lowest V/F. During this process, MILP is used find the best optimized mapping with the least minimized temperature. In addition, an efficient algorithm is proposed to release some applications from the list in case they violate the temperature threshold.

Application Mapping Techniques. Contiguous mapping algorithms are preferred choices for application mapping because non-contiguous mapping techniques do not consider the increase in communication latency between tasks which requires inter communication. Contiguous mapping on the hand ensures that tasks are mapped to cores located in the same region. However, due to the alignment of nodes directly next to each other, the dissipating heat generated by each node affects their neighbouring nodes gradually increasing the temperature of the system as applications are being executed. Furthermore, because contiguous mapping demands that an application is mapped in one region, an incoming application may be forced to wait when there are insufficient nodes available for it to be mapped in one region. As a matter of fact, in some cases, the application will be non-contiguously mapped to free nodes to satisfy the application deadline [31].

Nonetheless, to accommodate more applications on the chip, Ng et al. [31] propose an optimized technique called defragmentation. Defragmentation ensures that, all the free nodes which are dispersed on the chip are gathered into one region for an application to be mapped contiguously. Similarly, X. Wang et al. [32] introduced an application mapping algorithm which dynamically adjust and shift tasks onto different nodes for a contiguous mapping to take place. The algorithm proposed relocate tasks to different nodes to accommodate a new application in a square-shaped region.

Unfortunately, accommodating more applications means more hot regions on the chip as shown in Figure 4. For this purpose, Kanduri et al. [17] presented an optimized application mapping and patterning algorithm for DSCSs based on MapPro. The dense nodes from the region are activated as dark nodes to cool their active counterparts. During task mapping, the task with the highest communication volume is mapped to the first node. This process is repeated until the last task is assigned. After this process, one node in the square region is left un-occupied and used as a dark node to avoid hotspot.

Similarly, Aghaaliakbari et al. [33] propose a contiguous mapping algorithm which positions dark nodes in between application tasks to reduce heat. Rezaei et al. [34] also proposed a contiguous mapping algorithm contiguous mapping algorithm called Round Rotary mapping which targets a hybrid Wireless NoC virtually divided into regions. The proposed algorithm map applications in a round robin approach to evenly distribute applications all over the chip.

Moreover, by placing dark nodes in contiguous mapping algorithms, the hot regions could be reduced thus allowing more nodes to be activated. In addition to this, prioritising younger cores is also an essential technique because ageing cores dissipate more heat when they are stressed. Furthermore, nodes are able to perform at peak V/F levels when dark nodes are activated near it. Therefore, it will be beneficial for contiguous mapping algorithms to incorporate dark-silicon patterning approaches for a trade-off between communication cost and hot regions by efficiently positioning dark nodes in between tasks. Another alternative would be to incorporate heterogeneity in such a way that different resources are used to perform different computations. Every application has its own requirements for executing tasks. Computer-intensive applications require more power to execute applications while communication tasks require close connection with other tasks. These various applications could be executed using different nodes.

3.6. Architectural Heterogeneity

It has been proven [35] that incorporating heterogeneity through diverse materials which offers extra power savings in DSCSs, reduces the dynamic and leakage power consumption at a cost of a slight degrade in performance [36]. For this purpose, many techniques have been proposed in literature that combines different materials, sizes etc. to offer more power for actual computation. Shafique et al. [19] conducted a survey about the challenges in dark-silicon trends. In the survey, Shafique addresses the challenges of dark-silicon by presenting factors which demands high emphasis on when designing a system. Particularly, high emphasis is placed on the importance of incorporating heterogeneity.

Heterogeneous Cores. Zhang et al. [36] demonstrated that the employment of diverse materials to form processors can lead to less dark areas on the chip. Zhang proved that by integrating High-K (consists of big cores) and NEMS-CMOS (consists of small cores), the processor can operate more efficiently than the conventional CMPs formed with one material. In addition to this, because NEMS-CMOS consumes less power and generate less heat, the power density is smaller compared to CMPs formed with a single material.

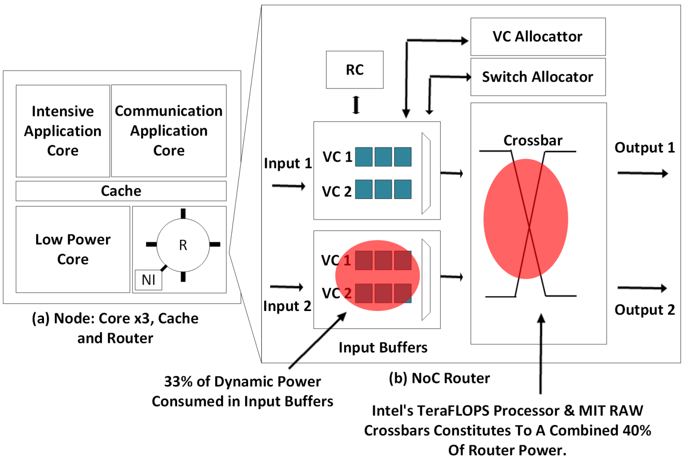
Yang et al. [26] approached the use of heterogeneity in a different design aspect by introducing a Quad-core Cluster Architecture which is not situated about the size of the core but rather the purpose of each core. The Quad-core Cluster Architecture consists of four different types of heterogeneous cores: High Performance (HP), General Purpose (GO), Power Saving (PA) and Low Energy (LE). The integration of these cores allows different types of applications to be executed on different cores depending on the workload. For example: In this architecture, the HP cores are used for intensive workloads which consumes the highest power consumption while the LE cores are used for workloads which consumes the lowest power consumption.

Power consumption is one major factor which constitutes to the heat dissipated by the on-chip resources. The amount of heat generated by the resources is proportional to the amount of power consumed by each resource. Incorporating components which consists of power hungry elements results in high power consumption which increases the amount of heat being generated by the resources. By incorporating heterogeneity, power saving materials can be used to form low power architectures.

Moreover, one common action that all these techniques that we have review share is that, to reduce power consumption or temperature, they ignore uncore components such as the Last Level Cache (LLC) and the routers in NoC. Ignoring these components result in an increase in heat since they contribute to the power consumption. Additionally, these components consume a significant of on-chip power and therefore impacts heavily on the power budget. To ensure that, that the power budget allocated for a specific chip is sufficient enough for high performance computation, we target at reducing the power consumption in the NoC interconnect and the memory sub-system without performance degradation.

4. The Dominance of Uncore components in Dark-Silicon Constraint Systems: The NoC Interconnect and Cache Architecture

The power dominance of uncore components (memory hierarchy (L2/L3 caches), memory controllers (MCs) and Interconnect) are often ignored in DSCSs, with majority of the power budgeting techniques (V/F scaling, power-gating, dynamic cache resizing and pipeline reconfigurations) that are found in literature, either targeted at the processor level or chip level. Therefore, for more power to be available for executing applications, the power consumption in on-chip components needs to be addressed.



(a) Node: Core x3, Cache and Router

(b) NoC Router

33% of Dynamic Power Consumed in Input Buffers

Intel's TeraFLOPS Processor & MIT RAW Crossbars Constitutes To A Combined 40% Of Router Power.

Figure 5. Router Architecture

Moreover, recent advances in technology have seen the NoC paradigm and MCA emerge as the standard interconnect for many-core systems, rapidly replacing the traditional buses and single-level-caches. NoC supplies high level of parallelism through multiple working routers and links, clustered with cores and caches together to form a node. With the introduction of these components in many-core systems in which they scale proportionate to, processing power is set to increase [37]. Figure 5 depicts a DSCS node comprised of 3 cores, a cache and a router. It is therefore important to address the power consumption of these uncore components as many-core systems dominate modern technology. Evaluation results conducted with McPAT [38] shows that uncore components are responsible for nearly half of the chip's total budget with the LLC and NoC interconnect being the largest consumers.

Caches suffer from high leakage in cache storage cells caused by the size reduction in emerging chip resources. NoC's power consumption on the other hand is down to its power-hungry elements. This problem becomes even worse when computational sprinting is applied on cores [39–41]. This affects the total power budget and makes it hard for more power to be used for actual computation. This part of the survey present techniques which can be applied to reduce the power consumption of these components but before we introduce a background information of each component.

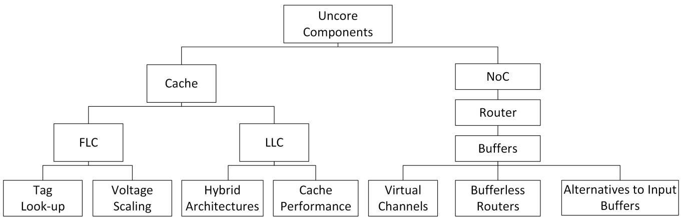


Figure 6. Uncore Components

4.1. NoC: Router Architecture

Conventional bus-based architectures inability to scale along with many-core systems, and supply performance proportionate to the number of nodes available in SoCs, have seen the NoC paradigm established itself as the standard interconnect for exa-scale computing [42–45]. NoC scales along with the size of the architecture and therefore amplifies the throughput equivalent to the system performance. A typical NoC architecture is comprised of routers and links. In the many-core system, NoC is used to form nodes as depicted in Figure 5. Routers communicate with each other through the links which establishes multiple access and communication channels between a source and a destination. However, the switching of activities of transistors during transmission causes an increase in dynamic power consumption and with leakage power already dominating power consumption, the overall chip temperature rises. [46–51]. Consequently, the router architecture has already been identified by many as the main component responsible for majority of the NoC's total power. However, with the continues shrinkage of technology, the power consumption in the links have also increased along with the workload. NoC routers are composed of power consuming elements such as the buffers, arbiters, crossbars, input and output ports. In as much as all these elements imbibe the power budget of the NoC, buffers and the crossbars are identified as the main culprits to exaggerate above the power constraints [47, 52]. With the increasing dark fractions in many-core systems, a significant portion of power can be saved through optimized algorithms and components. However, a lot of factors must be considered before proposing schemes relating to the buffers. One thing to consider is that, the absence of buffers provokes network congestion leading to high latency while on the other hand, excessive use of buffers aggravates the chips power consumption. Therefore, a balance is required.

Run-time Power Consumption Techniques For NoC Architectures. Modarressi et al. [53] propose a NoC Architecture in which packets can bypass the dark regions of the chip. The proposed algorithm takes the CTG of an input application and the number of active cores and establishes virtual long links among them. Furthermore, the router architecture of the dark nodes is optimized as follows: Firstly, the short-cut path of an input port that allows the pipeline stages of a router to be bypassed is selected. Secondly, incoming flits are then buffered using a register along a virtual long link which is established between two active nodes. Thirdly, the register indicates which output is part of the virtual link and which input port should be assigned to it. Bypassing the pipeline stages reduces the power consumption as power hungry elements such as buffers and virtual channels are avoided. In dynamic system workloads, the number of nodes available changes based on the arrival and departure of applications. In theory, active nodes can run at a higher frequency level if dark nodes are located near it for heat dissipation; this ultimately helps leverage the temperature of the system. Unfortunately, the downside of this is the communication latency between active cores. This proposed design allows the communication latency between two active cores to be minimized thus enhancing the performance of the system.

Bokhari et al. [54] propose the Malleable NoC for DSCS CMPs. In the proposed architecture, each node contains multiple heterogeneous routers designed for their frequencies and voltage to be altered. Depending on the behaviour/characteristics of an application, a router from each node is selected and formed into a low power NoC Fabric while idle routers are switched off.

Sharifi et al. [5] propose PEPON, a power budget distribution mechanism that shares the chip-wide power budget among the chip's resources (cores, caches and NoC) based on the workload for an optimized performance while respecting their allocated budget.

Reducing Power Consumption in The NoC Router Architecture: Buffers. Input buffers occupy majority of power consumption in the router architecture [52, 55]. Therefore, by reducing the power consumption of the input buffers, the power consumption of the chip will be reduced. The following techniques either seek to avoid the use of input buffers during run time or activate and deactivate them depending on the workload. An effective way to reduce power consumption is reducing the number of pipeline stages that packets traverse to reach their destination. By reducing the pipeline stages, dynamic power is reduced as well reducing the workload latency [56, 57].

Alternatively, virtual channels are employed in buffers to enable parallelism in one router. The Traffic-Based Virtual Channel Algorithm introduced in [58], enables the switch port of virtual channels to be organised into various cells. By grouping them, some of these cells can be powered-on or powered-off based on the network traffic and congestion.

Virtual channels are employed in buffers to enable simultaneous use of one physical channel. However, this consumes power. To reduce the power consumption of virtual channels, Zhan et. [59] propose an algorithm which categorises the virtual channels of a switch port into different levels. The architecture consists of a level (lower level) designed with SRAM and another with STT-RAM. The use of STT-RAM trade-off leakage power for dynamic power which can be tolerated. In addition to this, the algorithm allows the SRAM level to be powered-on, off or left in a drowsy state. In case of heavy traffic, the STT-RAM levels are activated. Nasirian et al. [60] on the other hand, employs a power-gating control unit to disable buffers when they are in-active for a number of cycles. However, power-gating can cause a performance penalty and therefore, system performance needs to be considered. This is because, constantly turning-on and off routers leads to non-negligible power overhead. Secondly, switched-off routers block all paths it intersects with and therefore, arriving packets have to wait until the router is powered-on first before traversing to the next router. Nonetheless, power punch is presented by [61] to send a signal three hops ahead to alert routers that are switched-off are about to switched off or routers which are switched-off to stay activated.

Another alternative to input buffers is the concept of bufferless routers. Bufferless Routers [62–65] have emerged as one possible solution to the leakage power consumption in routers. Unfortunately, due to the performance bottleneck that occur in bufferless router architectures, this technique is often disregarded. Buffers are used as temporal storage for packets which cannot be transmitted immediately. The absence of it causes packets to be deflected leading to livelock which also increases the power consumption. For this purpose, some techniques introduce heterogeneous architectures comprised of buffered and bufferless routers.

Fang et al. [66] propose a heterogeneous NoC architecture comprised of buffered and bufferless routers. Results show that the use of both these routers reduces the power consumption by 42%. Furthermore, this reduction in power allows more nodes to be activated compared to a generic buffered router.

Naik et al. [67] introduces a heterogeneous NoC comprised of circuit switched buffered and bufferless routers. The use of this heterogeneous approach reduces the power consumption by 26% and 32% in area.

Kodi et al. [68] on the other hand, introduced a dual-function links architecture called iDeal which unlike input buffers, does not consume a lot of power. iDeal uses a dynamic router buffer allocation to allocate incoming flits to any available buffer.

Similarly, DiTomaso et al. [52] propose a power-efficient architecture called QORE, which saves power consumption through the use of multi-function channel buffers and enhances the performance through reversible links. Li et al. [69] on the other hand, replaces the traditional SRAM with 3T_N eDRAM. The result of this is a reduction of 52% of power consumption and 43% of area.

4.2. Reducing Power Consumption in the Cache Architecture

On-chip cache memories account for a significant portion [70–77] of power consumption in embedded devices. Therefore, for mobile devices that run on batteries, efficient power optimization techniques are highly in demand as the sizes of transistors progressively decreases. The introduction of MCA trades-off performance for power expanding the fraction of chip area and on-chip power that caches account for [41, 70, 78]. Consequently, this increase in chip area and power can lead to thermal and reliability issues and therefore, reducing cache power consumption can avoid this and increase the power budget available for actual computation. While Last Level Caches (LLC) account for majority of leakage power due to their relatively large sizes, the First Level Cache (FLC) dominate dynamic power. Therefore, architectural techniques which seek to reduce the leakage power switches-off parts of the caches off and focus on minimizing transistor activity during cache accesses at the expense of performance. For lower memory caches, it is practical for sequential accesses of meta-data and data arrays to take place to save energy because very few access occur [79]. However, for FLC caches, there is a performance penalty. As a matter of fact, the lower memories are only accessed when there is a cache miss. Figure 7 depicts an image of a typical cache architecture in many-core CMPS. L1 is generally referred as the FLC and L2/L3 is referred as the LLC.

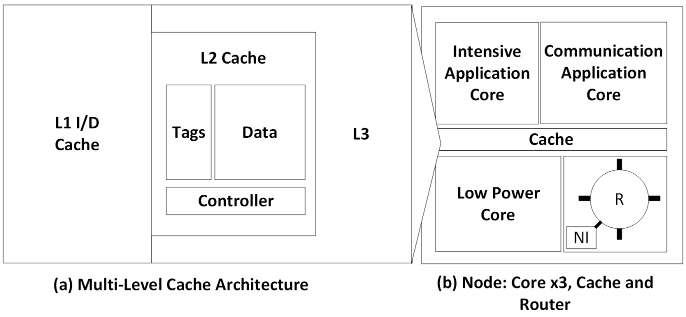

L1 I/D Cache
L2 Cache
Tags
Data
Controller
L3
Intensive Application Core
Communication Application Core
Cache
Low Power Core
R
NI
(a) Multi-Level Cache Architecture
(b) Node: Core x3, Cache and Router

Figure 7. Cache Architecture

Buffers and caches are similar because they are both used as temporal storage however, they consume a lot of power. Reducing the sizes of these two components can drastically affect the performance of the system. Chakraborty et al. [80] conducted a survey on caches and concluded that, turning-off cache banks trades-off performance for power. Results show that decreasing the cache banks from 16 to 8 caused a massive degrade in performance. However, when 12 banks are used, this degrade in performance is not as high. Although, power is reduced by shutting down cache banks, there is an increase in conflict misses. In this section of the paper, we present techniques for reducing power in both the FLC and LLC.

First Level Cache Power Consumption. First-Level Caches (FLC) are generally optimised for performance enhancement with less emphasis on power efficiency due to the impact and importance of having high associativity. For this purpose, designers trade-off power consumption for high performance in FLCs. Consequently, to improve the performance of memory, Set Associates Caches (SAC) are employed to enable blocks to be stored anywhere in the cache. This reduces cache miss rates and improves the performance of the system. Unlike associative caches where, a tag array has to be compared with each block in the cache, data is accessed parallel with a lookup tag. Unfortunately, during this process, power is wasted reading meta-data and looking up all the sets when only one set will be accessed after the cycle. Consider a 16-way set associative cache. A significant amount of power is wasted accessing all sets when the required data resides in only one set.

To avoid these challenges, several techniques have been proposed to address the power loss in set-associative FLC's trading-off power for design complexity or an increase in latency. These techniques can however be can classified into two categories (Tag Look up and Voltage Scaling).

A. Tag Look up: Some of the techniques proposed perform tag lookup and data sequentially. Unfortunately, this increases the cache latency. Others on the hand, store parts of the data and retrieve way information before the FLC is accessed escaping the need for accessing all sets. Performing tag lookup and reading tags introduce extra cycles which consumes more power [81]. Therefore, optimized techniques are very essential because FLC performance plays an important role in processor efficiency. On a cache miss, the system suffers a performance penalty which further increases the power consumption.

For this purpose, Zhang et al. [82] propose an Early Tag Lookup (ETL) for FLC instruction caches. Unlike existing 2-phase methods, the proposed algorithm determines the matching way one cycle earlier than the actual cache access, eliminating non-matching way accesses without sacrificing performance. The technique proposed retains two instruction fetch addresses. One of these being the current fetch address stored in the program counter and the other in the next program counter. The matching way is determined by looking up the tag array using the next programming counter so when it is loaded up by the program counter, the matching way is already known. The program counter therefore accesses the matching way without accessing other ways.

Similarly, Dai et al. [83] proposed an early tag access cache technique which determines the location of most memory instructions before the FLC Data cache is accessed. The proposed technique operates by storing a part of the physical address in tag arrays while the conversion between the virtual address and physical address is performed by the Translation Look-aside Buffer (TLB). This data is used to locate the destination of a required memory instruction during the Load/Stage Queue before accessing the FLC data cache escaping the need for accessing all ways.

Valls et al. [84] on the proposed the tag filter cache which unlike the first two techniques can be applied to all levels of the cache hierarchy. The proposed architecture filters the number of tags and data blocks to be checked when accessing the cache hierarchy by using the least significant bits of the tag part of address to determine which ways to access. The proposed architecture reduces power consumption between 74% and 85.9%. Sembrant et al. [85] proposed an extended TLB which will provide the location of cache lines in the data-array by adding an extra way index information (way location and location of cache lines). This reduces extra data array reads and avoid tag comparisons.

In contrast, Bardizbanyan et al. [81] argues that accessing ways sequentially by predicting the location of memory instructions affect the performance by incurring extra cycles due to additional switching of the clock. For this purpose, they propose load data dependency detection, a technique which decides when to sequentially access the FLC data based on the data dependency of the load. Similarly, Dayalan et al. [86] propose a technique which dynamically selects the best associativity of the cache during execution. The proposed technique operates by employing shadow tags to monitor how the cache would have performed if it was operating in the other mode.

In conventional MCA, the FLC data cache and write buffer are accessed in parallel for the same data. During a write/read miss, both the FLC data cache and write buffer are updated. Lee et al. [70] proposed an architecture which functions opposite to this. In the proposed architecture, during write operations, only the write buffer is updated. The only time the FLC is updated is when data is retired or the write buffer is full.

B. Voltage Scaling: Alternatively, downscaling the supply voltage close to the transistors' threshold is a technique which can effectively reduce the power consumption in FLCs. However, as a result of operating below the safe margin, persistent faults occur caused by voltage and temperature variations. Therefore, techniques which employ near-threshold scaling utilizes Error Correcting Codes (ECC) to overcome this challenge. ECC encoder generates parity bits when a data line is updated. During the reading of the data line, the decoder regenerates the parity bits to check and correct any existing faults. This process requires extra cycles and consume power causing a performance penalty which FLCs cannot tolerate. This gets even worse, when the fault rate is very high which is normal in near threshold scaling [87, 88].

For this purpose, Reviriego et al. [89] proposed a Single Error Correcting - Multiple Adjacent ECC to correct faults in one cycle (SEC-MAECC). Similarly, Yalcin et al. [87] proposed an improved version of SEC-MAEC to correct the faults in half a cycle. The proposed architecture reduces the encoding and decoding latency up to 80%.

Hijaz et al. [88] proposed an FLC hybrid architecture which can operate in two different modes (Cache line disable and correction and disable). The proposed architecture switches in between mode to preserve the performance of the system. Cache Line disable techniques allow the FLC to function at near-threshold voltages. During this process, cache lines which suffer from error bit rates are shut down limiting the cache capacity. In case the cache capacity loss becomes too high, the FLC utilizes (SEC-MAECC) to correct the faulty cache lines and enable them for use.

Saito et al. [90] proposed a FLC architecture which operate under different speed through Dynamic Voltage Frequency Scaling (DVFS). The proposed architecture dynamically selects the right speed depending on the type of performance that is required. Yan et al. [91] proposed two techniques which permit voltage scaling in FLCs (data and instruction caches) without a performance penalty (access latency). The first technique, Fault-Free Window (FFW) reduces the effect of defective words by only permitting cache lines to only store the most likely accesses. The second technique prevent the core from accessing defective words. Das et al. proposed a replacement policy algorithm which prioritise remote blocks to remain in the FLC to avoid latency. The proposed technique reduces power consumption by 14.85%.

Last Level Cache Power Consumption.

As previously mentioned, LLCs have been reported to occupy and consume majority of leakage power because of its large size. To improve the power efficiency, several techniques have been proposed forward. These techniques can however be classified into two categories: hybrid architectures and cache performance.

A. Hybrid Architectures:

STT-RAM, has been widely tutored as the conventional material for LLC cache design. With similar like features such as high density, low power consumption and good performance, STT-RAM is able to mirror performances close to that of SRAM. However, STT-RAM suffers extensively from dynamic power consumption during write accesses. Consequently, STT-RAM read latency also becomes an issue when implemented in FLCs.

Komalan et al. [92] proposed a NVM FLC with a very wide buffer to mitigate the read latency. The proposed architecture offers more area with a reduction in power consumption. However, it is not quite sure how the proposed architecture will perform during heavy workloads as it was experimented under light workloads.

Similarly, Wang et al. [93] proposed a hybrid STT-RAM and SRAM FLC. The proposed design incorporates the MESI cache coherence protocol to effectively manage block relocation between the SRAM and STT-RAM partitions. However, because system performance is closely related to FLC, to the best of our knowledge, not many work have been done on designing FLCs out of SRAM. In terms architecting LLC caches out STT-RAM, the most considered optimized solution for MCA by many designers is employing both SRAM and STT-RAM [94–97].

Moreover, the following authors combine the benefits of both hardware technology to overcome the challenges that each technology brings. Li et al. [94], Kim et al. [98] and Safayenikoo [99] all proposed architectures which incorporates STT-RAM and SRAM technology. The proposed design proposed by Li, focuses on sharing private STT-RAM groups with neighbouring nodes to reduce latency and power consumption. KimâĂŹs architecture consists of algorithm which decides which region of the cache (STT-RAM and SRAM), data needs to be placed in. Safayenikoo's cache architecture moves data to the SRAM blocks when the energy writes in the STT-RAM increases.

Asad et al. [100] introduced a heterogeneous cache memory hierarchy for CMPs. Each cache level in the memory hierarchy has been designed with a different memory technology (Static Random Access Memory (SRAM), Embedded Dynamic Random Access Memory (eDRAM) and STT-RAM. Similarly, Onsori et al. [101] proposed a hybrid memory system for DSCSs comprised of NVM devices. The propose architecture consists of STT-RAM memory banks which have been incorporated with SRAM memory banks.

B. Cache Performance: Alternatively, power-gating techniques are employed to disable idle parts of the cache when under minimal workload. However, shutting down idle parts of a cache can incur performance penalties which can exacerbate the power being dissipated [102]. To ensure that power-gating techniques does not impose a significant threat on the performance, less likely used banks and powered-off and their requests forwarded to neighbouring requests [102]. Other techniques on the hand, power-off cache ways instead of banks. Azad et al. [103] on the other hand reduces power consumption by categorising cache blocks into different groups and apply ECC based on the level of protection that is required.

4.3. Summary

The breakdown of Dennardian Law has made it a challenge for systems to maintain the same power performance as the same time transistors quadrupled in many-core/multi-core systems. For this purpose, the dark-silicon phenomenon has become an interesting field because it allows only a subset of resources to be active and with the application of the techniques presented, this subset of resources can provide high level performances.

Table 1 presents a summary of all the techniques which target many-core systems in DSCSs. Unlike other work found in literature, we have targeted all of the on-chip components with considering the performance of the chip as well as the temperature. It can be deduced from the table that for a good temperature efficiency, the power budgeting technique must consider several factors. One of these checking the surrounding of the neighbouring codes before allocating power. Consequently, for a high-power efficiency, idle virtual channels can shut down while alternative buffers are employed. However, this can degrade the performance therefore, it is only wise to do so under minimal workload. Additionally, the implementation of STT-RAM and SRAM architecture offers a high-power efficiency resulting in the reduction of high temperature.

Unfortunately, with scaling set to go even deeper, dark-silicon may yet become a constraint rather than a solution. Deep scaling meant that fractional parts of the chip had to be shut down. Therefore, reducing the transistor size further will only increase the fractional part of the chip which are dark [104].

Consequently, this had led many researchers to now shift their focus to Near-Threshold Voltage Computing Constraint Systems (NTCCS) [105], [106]. In contrary to DSCSs where transistors are under-utilised, NTC allows all the transistors to operate in the near-threshold region thus providing a fluid balance between power and delay [107]. Since, the entire chip can be utilised at the same time, multiple applications can be executed, however this is at a cost of performance degradation and reliability loss. Another alternative would be a joint implementation of both dark-silicon and NTC in future technologies. A technique which has already been proven to provide better performances [108].

5. Conclusion

This paper introduced techniques which can be implemented in DSCSs to reduce power consumption whilst considering performance and avoiding high temperature. Particularly, efficient application mapping techniques and heterogeneous architectures are presented. Using the correct application mapping algorithm to distribute applications across the chip can effectively reduce thermal hotspots. In addition to this, we showed that by using resources which are made of power saving materials, high power consumption which increases the power density can be reduced. Thus, keeping the temperature of the working resources low enough for the chip to function beyond its supplied thermal design constraints.

In addition to this, we provided alternative thermal design constraints which can implemented to allow systems to function beyond their restricted threshold. Furthermore, we discussed novel techniques which can applied to the NoC interconnect and the cache architecture. Power consumption in the NoC interconnect can be reduced through the replacement of input buffers. Cache power consumption on the hand can reduced by implementing hybrid STT-RAM and SRAM architectures.

Table 1. A summary of DSCS techniques

Area	Techniques	Temperature Efficiency	Power Efficiency	Pros	Cons
Power Budgeting	- Maximum Power Budget	Medium	Medium	Performance enhancement	Ageing
Architectural Heterogeneity	- Heterogeneous Cores - Heterogeneous Memory	Medium	Low	- Increase in the number of executed applications - Hotspot reduced	Performance penalty
Run-Time Management	- DVFS - Task Migration - Power-gating	Medium	Medium	- Sudden overshoot avoided - Low V/F Operation - Different power metrics per-core	Performance penalty
NoC Interconnect	- Powering-off Virtual channels - Bufferless routers - Alternative buffers	Medium	High	- Low dynamic power - Low leakage power	- Latency penalty - Deadlock - Performance penalty
Cache Memory	- Tag Look Up - Voltage Scaling - Hybrid Architectures	Medium	High	- Low dynamic power - Low leakage power	Latency penalty

Based on these findings, it can be deduced that the thermal limitations in the dark-silicon era cannot be significantly reduce by application mapping alone. However, with the addition of architectural heterogeneity and consideration of uncore components, the thermal profile of the chip can be kept at a minimum whilst still maintaining the performance. With this in mind, optimization techniques for NoC and the memory subsystem will be the focal of our future work.

References

[1] M. O. Agyeman, A. Ahmadinia, and N. Bagherzadeh, "Performance and energy aware inhomogeneous 3d networks-on-chip architecture generation," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 6, pp. 1756–1769, 2016.

[2] M. O. Agyeman, K. Tong, and T. Mak, "Towards reliability and performance-aware wireless network-on-chip design," in *IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFTS)*, 2015, pp. 205–210.

[3] M. O. Agyeman, A. Ahmadinia, and A. Shahrabi, "Low power heterogeneous 3d networks-on-chip architectures," in *International Conference on High Performance Computing Simulation*, 2011, pp. 533–538.

[4] M. O. Agyeman and A. Ahmadinia, "Optimising heterogeneous 3d networks-on-chip," in *International Symposium on Parallel Computing in Electrical Engineering*, 2011, pp. 25–30.

[5] A. Sharifi, A. K. Mishra, S. Srikantaiah, M. Kandemir, and C. R. Das, "Pepon: Performance-aware hierarchical power budgeting for noc based multicores," in *International Conference on Parallel Architectures and Compilation Techniques (PACT)*, 2012, pp. 65–74.

[6] J. Henkel, H. Khdr, S. Pagani, and M. Shafique, "New trends in dark silicon," in *2015 52nd ACM/EDAC/IEEE Design Automation Conference (DAC)*, 2015, pp. 1–6.

[7] A. Pedram, S. Richardson, M. Horowitz, S. Galal, and S. Kvatinsky, "Dark memory and accelerator-rich system optimization in the dark silicon era," *IEEE Design Test*, vol. 34, no. 2, pp. 39–50, 2017.

[8] A. Kanduri, A. M. Rahmani, P. Liljeberg, A. Hemani, A. Jantsch, and H. Tenhunen, "A perspective on dark silicon," 01 2017.

[9] P. Saini and R. Mehra, "Article: Leakage power reduction in cmos vlsi circuits," *International Journal of Computer Applications*, vol. 55, no. 8, pp. 42–48, 2012.

[10] S. Borkar, "Getting gigascale chips: Challenges and opportunities in continuing moore's law," *Queue*, vol. 1, no. 7, pp. 26–33, 2003.

[11] M. Shafique, S. Garg, J. Henkel, and D. Marculescu, "The eda challenges in the dark silicon era: Temperature, reliability, and variability perspectives," in *Proceedings of the 51st Annual Design Automation Conference*, ser. DAC, 2014, pp. 185:1–185:6.

[12] D. Brooks, R. P. Dick, R. Joseph, and L. Shang, "Power, thermal, and reliability modeling in nanometer-scale microprocessors," *IEEE Micro*, vol. 27, no. 3, pp. 49–62, 2007.

[13] I. Corporation., "Dual-core intel xeon processor 5100 series datasheet," *revision 003*, August 2007.

[14] S. Nussbaum, "Amd trinity apu," *In Hot Chips*, 2012.

[15] H. Wang, M. Zhang, S. X. D. Tan, C. Zhang, Y. Yuan, K. Huang, and Z. Zhang, "New power budgeting and thermal management scheme for multi-core systems in dark silicon," in *IEEE International System-on-Chip Conference (SOCC)*, 2016, pp. 344–349.

[16] S. Pagani, H. Khdr, J. J. Chen, M. Shafique, M. Li, and J. Henkel, "Thermal safe power (tsp): Efficient power budgeting for heterogeneous manycore systems in dark silicon," *IEEE Transactions on Computers*, vol. 66, no. 1, pp. 147–162, 2017.

[17] A. Kanduri, M. H. Haghbayan, A. M. Rahmani, P. Liljeberg, A. Jantsch, and H. Tenhunen, "Dark silicon aware runtime mapping for many-core systems: A patterning approach," in *IEEE International Conference on Computer Design (ICCD)*, 2015, pp. 573–580.

[18] X. Wang, A. K. Singh, B. Li, Y. Yang, H. Li, and T. Mak, "Bubble budgeting: Throughput optimization for dynamic workloads by exploiting dark cores in

many core systems," *IEEE Transactions on Computers*, vol. 67, no. 2, pp. 178–192, 2018.

[19] M. Shafique and S. Garg, "Computing in the dark silicon era: Current trends and research challenges," *IEEE Design Test*, vol. 34, no. 2, pp. 8–23, 2017.

[20] A. M. Rahmani, M. H. Haghbayan, A. Kanduri, A. Y. Weldezion, P. Liljeberg, J. Plosila, A. Jantsch, and H. Tenhunen, "Dynamic power management for many-core platforms in the dark silicon era: A multi-objective control approach," in *IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED)*, 2015, pp. 219–224.

[21] M. Salehi, M. Shafique, F. Kriebel, S. Rehman, M. K. Tavana, A. Ejlali, and J. Henkel, "dsrelim: Power-constrained reliability management in dark-silicon many-core chips under process variations," in *International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS)*, 2015, pp. 75–82.

[22] A. M. Rahmani, M. H. Haghbayan, A. Miele, P. Liljeberg, A. Jantsch, and H. Tenhunen, "Reliability-aware runtime power management for many-core systems in the dark silicon era," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25, no. 2, pp. 427–440, 2017.

[23] M. H. Haghbayan, A. Miele, A. M. Rahmani, P. Liljeberg, and H. Tenhunen, "A lifetime-aware runtime mapping approach for many-core systems in the dark silicon era," in *Design, Automation Test in Europe Conference Exhibition (DATE)*, 2016, pp. 854–857.

[24] M.-H. Haghbayan, A. Kanduri, A.-M. Rahmani, P. Liljeberg, A. Jantsch, and H. Tenhunen, "Mappro: Proactive runtime mapping for dynamic workloads by quantifying ripple effect of applications on networks-on-chip," in *Proceedings of the 9th International Symposium on Networks-on-Chip*, ser. NOCS '15, 2015, pp. 26:1–26:8.

[25] T. S. Muthukaruppan, M. Pricopi, V. Venkataramani, T. Mitra, and S. Vishin, "Hierarchical power management for asymmetric multi-core in dark silicon era," in *50th ACM/EDAC/IEEE Design Automation Conference (DAC)*, 2013, pp. 1–9.

[26] L. Yang, W. Liu, N. Guan, M. Li, P. Chen, and E. H. M. Sha, "Dark silicon-aware hardware-software collaborated design for heterogeneous many-core systems," in *Asia and South Pacific Design Automation Conference (ASP-DAC)*, 2017, pp. 494–499.

[27] X. Wang, A. K. Singh, B. Li, Y. Yang, T. Mak, and H. Li, "Bubble budgeting: throughput optimization for dynamic workloads by exploiting dark cores in many core systems," in *IEEE/ACM International Symposium on Networks-on-Chip (NOCS)*, vol. PP, no. 99, 2017, pp. 1–1.

[28] J. Wang, Z. Chen, J. Guo, Y. Li, and Z. Lu, "Aco-based thermal-aware thread-to-core mapping for dark-silicon-constrained cmps," *IEEE Transactions on Electron Devices*, vol. PP, no. 99, pp. 1–8, 2017.

[29] X. Wang, T. Fei, B. Zhang, and T. S. T. Mak, "On runtime adaptive tile defragmentation for resource management in many-core systems," *Microprocessors and Microsystems - Embedded Hardware Design*, vol. 46, no. Part B, pp. 161 – 174, 2016.

[30] M. Li, J. Yi, W. Liu, W. Zhang, L. Yang, and E. H. M. Sha, "An efficient technique for chip temperature optimization of multiprocessor systems in the dark silicon era," in *IEEE International Conference on High Performance Computing and Communications, IEEE 7th International Symposium on Cyberspace Safety and Security, andIEEE 12th International Conference on Embedded Software and Systems*, 2015, pp. 688–693.

[31] J. Ng, X. Wang, A. K. Singh, and T. Mak, "Defragmentation for efficient runtime resource management in noc-based many-core systems," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 24, no. 11, pp. 3359–3372, 2016.

[32] X. Wang, T. Fei, B. Zhang, and T. Mak, "On runtime adaptive tile defragmentation for resource management in many-core systems," 2016.

[33] F. Aghaaliakbari, M. Hoveida, M. Arjomand, M. Jalili, and H. Sarbazi-Azad, "Efficient processor allocation in a reconfigurable cmp architecture for dark silicon era," in *IEEE International Conference on Computer Design (ICCD)*, 2016, pp. 336–343.

[34] A. Rezaei, D. Zhao, M. Daneshtalab, and H. Zhou, "Multi-objective task mapping approach for wireless noc in dark silicon age," in *Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP)*, 2017, pp. 589–592.

[35] H. F. Dadgour and K. Banerjee, "Design and analysis of hybrid nems-cmos circuits for ultra low-power applications," in *ACM/IEEE Design Automation Conference*, 2007, pp. 306–311.

[36] Y. Zhang, L. Peng, X. Fu, and Y. Hu, "Lighting the dark silicon by exploiting heterogeneity on future processors," in *ACM/EDAC/IEEE Design Automation Conference (DAC)*, 2013, pp. 1–7.

[37] K. Fischer, H. K. Chang, D. Ingerly, I. Jin, H. Kilambi, J. Longun, R. Patel, C. Pelto, C. Petersburg, P. Plekhanov, C. Puls, L. Rockford, I. Tsameret, M. Uncuer, and P. Yashar, "Performance enhancement for 14nm high volume manufacturing microprocessor and system on a chip processes," in *In IEEE International Interconnect Technology Conference / Advanced Metallization Conference (IITC/AMC)*, 2016, pp. 5–7.

[38] S. Li, J. H. Ahn, R. D. Strong, J. B. Brockman, D. M. Tullsen, and N. P. Jouppi, "Mcpat: An integrated power, area, and timing modeling framework for multicore and manycore architectures," in *Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*, 2009, pp. 469–480.

[39] C. Sun, C. H. O. Chen, G. Kurian, L. Wei, J. Miller, A. Agarwal, L. S. Peh, and V. Stojanovic, "Dsent - a tool connecting emerging photonics with electronics for opto-electronic networks-on-chip modeling," in *Networks on Chip (NoCS), IEEE/ACM International Symposium on*, 2012, pp. 201–210.

[40] H. Y. Cheng, M. Poremba, N. Shahidi, I. Stalev, M. J. Irwin, M. Kandemir, J. Sampson, and Y. Xie, "Eecache: Exploiting design choices in energy-efficient last-level caches for chip multiprocessors," pp. 303–306, 2014.

[41] D. Wendel, R. Kalla, R. Cargoni, J. Clables, J. Friedrich, R. Frech, J. Kahle, B. Sinharoy, W. Starke, S. Taylor, S. Weitzel, S. G. Chu, S. Islam, and V. Zyuban, "The

implementation of power7tm: A highly parallel and scalable multi-core high-end server processor," pp. 102–103, 2010.

[42] M. O. Agyeman, A. Ahmadinia, and N. Bagherzadeh, "Performance and energy aware inhomogeneous 3d networks-on-chip architecture generation," *IEEE Trans. Parallel Distrib. Syst.*, 2016.

[43] M. O. Agyeman, W. Zong, J. Wan, A. Yakovlev, K. Tong, and T. S. T. Mak, "Novel hybrid wired-wireless network-on-chip architectures: Transducer and communication fabric design," in *Proceedings of the 9th International Symposium on Networks-on-Chip, NOCS*, 2015, pp. 32.1–32.2.

[44] M. O. Agyeman and W. Zong, "An efficient 2d router architecture for extending the performance of inhomogeneous 3d noc-based multi-core architectures," in *International Symposium on Computer Architecture and High Performance Computing Workshops (SBAC-PADW)*, 2016, pp. 79–84.

[45] M. O. Agyeman, J. Wan, Q. Vien, W. Zong, A. Yakovlev, K. Tong, and T. Mak, "On the design of reliable hybrid wired-wireless network-on-chip architectures," in *IEEE International Symposium on Embedded Multicore/Many-core Systems-on-Chip*, 2015, pp. 251–258.

[46] Y. Hoskote, S. Vangal, A. Singh, N. Borkar, and S. Borkar, "A 5-ghz mesh interconnect for a teraflops processor," *IEEE Micro*, vol. 27, no. 5, pp. 51–61, 2007.

[47] H. Zheng and A. Louri, "Ez-pass: An energy amp; performance-efficient power-gating router architecture for scalable nocs," *IEEE Computer Architecture Letters*, vol. 17, no. 1, pp. 88–91, 2018.

[48] H. Farrokhbakht, H. M. Kamali, and S. Hessabi, "Smart: A scalable mapping and routing technique for power-gating in noc routers," in *2017 Eleventh IEEE/ACM International Symposium on Networks-on-Chip (NOCS)*, 2017, pp. 1–8.

[49] T. G. Mattson, M. Riepen, T. Lehnig, P. Brett, W. Haas, P. Kennedy, J. Howard, S. Vangal, N. Borkar, G. Ruhl, and S. Dighe, "The 48-core scc processor: the programmer's view," pp. 1–11, 2010.

[50] J. Zhan, Y. Xie, and G. Sun, "Noc-sprinting: Interconnect for fine-grained sprinting in the dark silicon era," pp. 1–6, 2014.

[51] J. Howard, S. Dighe, S. R. Vangal, G. Ruhl, N. Borkar, S. Jain, V. Erraguntla, M. Konow, M. Riepen, M. Gries, G. Droege, T. Lund-Larsen, S. Steibl, S. Borkar, V. K. De, and R. V. D. Wijngaart, "A 48-core ia-32 processor in 45 nm cmos using on-die message-passing and dvfs for performance and power scaling," *IEEE Journal of Solid-State Circuits*, vol. 46, no. 1, pp. 173–183, 2011.

[52] D. DiTomaso, A. K. Kodi, A. Louri, and R. Bunescu, "Resilient and power-efficient multi-function channel buffers in network-on-chip architectures," *IEEE Transactions on Computers*, vol. 64, no. 12, pp. 3555–3568, 2015.

[53] M. Modarressi and H. Sarbazi-Azad, "A reconfigurable network-on-chip architecture for heterogeneous cmps in the dark-silicon era," in *IEEE International Conference on Application-Specific Systems, Architectures and Processors*, 2014, pp. 76–77.

[54] H. Bokhari, H. Javaid, M. Shafique, J. Henkel, and S. Parameswaran, "Malleable noc: Dark silicon inspired adaptable network-on-chip," in *Design, Automation Test in Europe Conference Exhibition (DATE)*, 2015, pp. 1245–1248.

[55] P. Kundu, "On-die interconnects for next generation cmps,âĂİ in workshop on on- and off-chip interconnection networks for multicore systems," 2006.

[56] J. Postman, T. Krishna, C. Edmonds, L. S. Peh, and P. Chiang, "Swift: A low-power network-on-chip implementing the token flow control router architecture with swing-reduced interconnects," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 21, no. 8, pp. 1432–1446, 2013.

[57] S. Shenbagavalli and S. Karthikeyan, "An efficient low power noc router architecture design," in *Online International Conference on Green Engineering and Technologies (IC-GET)*, 2015, pp. 1–8.

[58] S. T. Muhammad, M. A. El-Moursy, A. A. El-Moursy, and A. M. Refaat, "Optimization for traffic-based virtual channel activation low-power noc," in *International Conference on Energy Aware Computing Systems Applications*, 2015, pp. 1–4.

[59] J. Zhan, J. Ouyang, F. Ge, J. Zhao, and Y. Xie, "Hybrid drowsy sram and stt-ram buffer designs for dark-silicon-aware noc," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 24, no. 10, pp. 3041–3054, 2016.

[60] N. Nasirian and M. Bayoumi, "Low-latency power-efficient adaptive router design for network-on-chip," in *IEEE International System-on-Chip Conference (SOCC)*, 2015, pp. 287–291.

[61] L. Chen, D. Zhu, M. Pedram, and T. M. Pinkston, "Power punch: Towards non-blocking power-gating of noc routers," in *IEEE International Symposium on High Performance Computer Architecture (HPCA)*, 2015, pp. 378–389.

[62] C. Fallin, C. Craik, and O. Mutlu, "Chipper: A low-complexity bufferless deflection router," in *IEEE International Symposium on High Performance Computer Architecture*, 2011, pp. 144–155.

[63] B. K. Daya, L. S. Peh, and A. P. Chandrakasan, "Towards high-performance bufferless nocs with scepter," *IEEE Computer Architecture Letters*, vol. 15, no. 1, pp. 62–65, 2016.

[64] C. Feng, Z. Liao, Z. Lu, A. Jantsch, and Z. Zhao, "Performance analysis of on-chip bufferless router with multi-ejection ports," in *IEEE International Conference on ASIC (ASICON)*, 2015, pp. 1–4.

[65] H. Kim, Y. Kim, and J. Kim, "Clumsy flow control for high-throughput bufferless on-chip networks," *IEEE Computer Architecture Letters*, vol. 12, no. 2, pp. 47–50, 2013.

[66] J. Fang, M. Cai, Z. Leng, and S. Liu, "A perspective from exploiting heterogeneity on networks-on-chip based on dark silicon mitigation," in *International Conference on Computational Science and Computational Intelligence (CSCI)*, 2016, pp. 590–595.

[67] A. Naik and T. K. Ramesh, "Efficient network on chip (noc) using heterogeneous circuit switched routers," in *International Conference on VLSI Systems, Architectures,*

Technology and Applications (VLSI-SATA), 2016, pp. 1–6.

[68] A. K. Kodi, A. Sarathy, A. Louri, and J. Wang, "Adaptive inter-router links for low-power, area-efficient and reliable network-on-chip (noc) architectures," in *Asia and South Pacific Design Automation Conference*, 2009, pp. 1–6.

[69] C. Li and P. Ampadu, "A compact low-power edram-based noc buffer," in *Low Power Electronics and Design (ISLPED), IEEE/ACM International Symposium on*, 2015, pp. 116–121.

[70] J. Lee and S. Kim, "Write buffer-oriented energy reduction in the l1 data cache for embedded systems," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 24, no. 3, pp. 871–883, 2016.

[71] M. A. Awan and S. M. Petters, "Enhanced race-to-halt: A leakage-aware energy management approach for dynamic priority systems," in *23rd Euromicro Conference on Real-Time Systems*, 2011, pp. 92–101.

[72] J. A.Artes, J.Ayala and F.Catthoor, "Survey of low-energy techniques for instruction memory organisations in embedded systems," in *Journal of Signal Processing Systems 70(1):1-19 January*, vol. 70, 2013, pp. 1–19.

[73] B. Maric, J. Abella, and M. Valero, "Analyzing the efficiency of l1 caches for reliable hybrid-voltage operation using edc codes," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 22, no. 10, pp. 2211–2215, 2014.

[74] E. Ofori-Attah, W. Bhebhe, and M. O. Agyeman, "Architectural techniques for improving the power consumption of noc-based cmps: A case study of cache and network layer," *Journal of Low Power Electronics and Applications*, vol. 7, 2017.

[75] E. Ofori-Attah, X. Wang, and M. O. Agyeman, "A survey of low power design techniques for last level caches," in *Applied Reconfigurable Computing. Architectures, Tools, and Applications - 14th International Symposium, Santorini, Greece, May 2-4„ Proceedings*, 2018, pp. 217–228.

[76] E. Ofori-Attah and M. O. Agyeman, "A survey of low power noc design techniques," in *Proceedings of the 2Nd International Workshop on Advanced Interconnect Solutions and Technologies for Emerging Computing Systems*, ser. AISTECS, 2017, pp. 22–27.

[77] ——, "A survey of recent contributions on low power noc architectures," in *2017 Computing Conference*, 2017.

[78] J. Choi and G. H. Park, "Nvm way allocation scheme to reduce nvm writes for hybrid cache architecture in chip-multiprocessors," *IEEE Transactions on Parallel and Distributed Systems*, vol. 28, no. 10, pp. 2896–2910, 2017.

[79] D. Williamson, "Low power applications, arm." 2017.

[80] S. Chakraborty, D. Deb, D. Buragohain, and H. K. Kapoor, "Cache capacity and its effects on power consumption for tiled chip multi-processors," in *International Conference on Electronics and Communication Systems (ICECS)*, 2014, pp. 1–6.

[81] A. Bardizbanyan, M. SjÃďlander, D. Whalley, and P. Larsson-Edefors, "Reducing set-associative l1 data cache energy by early load data dependence detection (eld3)," in *Design, Automation Test in Europe Conference Exhibition (DATE)*, vol. 24, no. 3, 2016, pp. 871–883.

[82] W. Zhang, H. Zhang, and J. Lach, "Reducing dynamic energy of set-associative l1 instruction cache by early tag lookup," in *IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED)*, 2015, pp. 49–54.

[83] J. Dai, M. Guan, and L. Wang, "Exploiting early tag access for reducing l1 data cache energy in embedded processors," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 22, no. 2, pp. 396–407, 2014.

[84] J. J. Valls, J. Sahuquillo, A. Ros, and M. E. GÃşmez, "The tag filter cache: An energy-efficient approach," in *23rd Euromicro International Conference on Parallel, Distributed, and Network-Based Processing*, 2015, pp. 182–189.

[85] A. Sembrant, E. Hagersten, and D. Black-Shaffer, "Tlc: A tag-less cache for reducing dynamic first level cache energy," in *Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*, 2013, pp. 49–61.

[86] K. Dayalan, M. Ozsoy, and D. Ponomarev, "Dynamic associative caches: Reducing dynamic energy of first level caches," in *IEEE 32nd International Conference on Computer Design (ICCD)*, 2014, pp. 118–124.

[87] G. Yalcin, E. Islek, O. Tozlu, P. Reviriego, A. Cristal, O. S. Unsal, and O. Ergin, "Exploiting a fast and simple ecc for scaling supply voltage in level-1 caches," in *IEEE 20th International On-Line Testing Symposium (IOLTS)*, 2014, pp. 1–6.

[88] F. Hijaz, Q. Shi, and O. Khan, "A private level-1 cache architecture to exploit the latency and capacity tradeoffs in multicores operating at near-threshold voltages," in *IEEE 31st International Conference on Computer Design (ICCD)*, 2013, pp. 85–92.

[89] P. Reviriego, S. Pontarelli, J. A. Maestro, and M. Ottavi, "Low-cost single error correction multiple adjacent error correction codes," *Electronics Letters*, vol. 48, no. 23, pp. 1470–1472, 2012.

[90] K. Saito, R. Kobayashi, and H. Shimada, "Reduction of cache energy by switching between l1 high speed and low speed cache under application of dvfs," in *International Conference On Advanced Informatics: Concepts, Theory And Application (ICAICTA)*, 2016, pp. 1–6.

[91] C. Yan and R. Joseph, "Enabling deep voltage scaling in delay sensitive l1 caches," in *46th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, 2016, pp. 192–202.

[92] M. P. Komalan, C. Tenllado, J. I. G. PÃľrez, F. T. FernÃąndez, and F. Catthoor, "System level exploration of a stt-mram based level 1 data-cache," in *Design, Automation Test in Europe Conference Exhibition (DATE)*, 2015, pp. 1311–1316.

[93] J. Wang, Y. Tim, W. F. Wong, Z. L. Ong, Z. Sun, and H. H. Li, "A coherent hybrid sram and stt-ram l1 cache architecture for shared memory multicores," in *Asia and South Pacific Design Automation Conference (ASP-DAC)*, 2014, pp. 610–615.

[94] J. Li, C. J. Xue, and Y. Xu, "Stt-ram based energy-efficiency hybrid cache for cmps," in *IEEE/IFIP International Conference on VLSI and System-on-Chip*,

2011, pp. 31–36.

[95] F. Shen, Y. He, J. Zhang, N. Jiang, Q. Li, and J. Li, "Feedback learning based dead write termination for energy efficient stt-ram caches," vol. 26, pp. 460–467, 05 2017.

[96] S. Agarwal and H. K. Kapoor, "Restrictingwrites for energy-efficient hybrid cache in multi-core architectures," in *IFIP/IEEE International Conference on Very Large Scale Integration (VLSI-SoC)*, 2016, pp. 1–6.

[97] R. K. Aluru and S. Ghosh, "Droop mitigating last level cache architecture for sttram," in *Design, Automation Test in Europe Conference Exhibition (DATE)*, 2017, pp. 262–265.

[98] N. Kim, J. Ahn, W. Seo, and K. Choi, "Energy-efficient exclusive last-level hybrid caches consisting of sram and stt-ram," in *IFIP/IEEE International Conference on Very Large Scale Integration (VLSI-SoC)*, 2015, pp. 183–188.

[99] P. Safayenikoo, A. Asad, M. Fathy, and F. Mohammadi, "Exploiting non-uniformity of write accesses for designing a high-endurance hybrid last level cache in 3d cmps," in *IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)*, 2017, pp. 1–5.

[100] A. Asad, O. Ozturk, M. Fathy, and M. R. Jahed-Motlagh, "Exploiting heterogeneity in cache hierarchy in dark-silicon 3d chip multi-processors," in *Euromicro Conference on Digital System Design*, 2015, pp. 314–321.

[101] S. Onsori, A. Asad, K. Raahemifar, and M. Fathy, "Optmem: Dark-silicon aware low latency hybrid memory design," in *International Conference on VLSI Systems, Architectures, Technology and Applications (VLSI-SATA)*, 2016, pp. 1–5.

[102] S. Chakraborty and H. K. Kapoor, "Static energy reduction by performance linked dynamic cache resizing," in *IFIP/IEEE International Conference on Very Large Scale Integration (VLSI-SoC)*, 2016, pp. 1–6.

[103] Z. Azad, H. Farbeh, A. M. H. Monazzah, and S. G. Miremadi, "An efficient protection technique for last level stt-ram caches in multi-core processors," *IEEE Transactions on Parallel and Distributed Systems*, vol. 28, no. 6, pp. 1564–1577, 2017.

[104] H. Esmaeilzadeh, E. Blem, R. St. Amant, K. Sankaralingam, and D. Burger, "Power limitations and dark silicon challenge the future of multicore," *ACM Trans. Comput. Syst.*, vol. 30, no. 3, pp. 11:1–11:27, 2012.

[105] R. G. Dreslinski, M. Wieckowski, D. Blaauw, D. Sylvester, and T. Mudge, "Near-threshold computing: Reclaiming moore's law through energy efficient integrated circuits," *Proceedings of the IEEE*, vol. 98, no. 2, pp. 253–266, 2010.

[106] S. Jain, S. Khare, S. Yada, V. Ambili, P. Salihundam, S. Ramani, S. Muthukumar, M. Srinivasan, A. Kumar, S. K. Gb, R. Ramanarayanan, V. Erraguntla, J. Howard, S. Vangal, S. Dighe, G. Ruhl, P. Aseron, H. Wilson, N. Borkar, V. De, and S. Borkar, "A 280mv-to-1.2v wide-operating-range ia-32 processor in 32nm cmos," in *IEEE International Solid-State Circuits Conference*, 2012, pp. 66–68.

[107] U. R. Karpuzcu, A. Sinkar, N. S. Kim, and J. Torrellas, "Energysmart: Toward energy-efficient manycores for near-threshold computing," in *IEEE International Symposium on High Performance Computer Architecture (HPCA)*, 2013, pp. 542–553.

[108] J. Wang, x. Fu, W. Zhang, Z. Junwei, K. Qiu, and T. Li, "On the implication of ntc vs. dark silicon on emerging scale-out workloads: The multi-core architecture perspective," *IEEE Transactions on Parallel and Distributed Systems*, vol. 28, no. 8, pp. 2314–2327, 2017.