

Work Load Scheduling For Multi Core Systems With Under-Provisioned Power Delivery

Divya Pathak
Drexel University
divya.pathak@drexel.edu

Houman Homayoun
George Mason University
hhomayoun@gmu.edu

Ioannis Savidis
Drexel University
isavidis@coe.drexel.edu

ABSTRACT

An energy efficient power delivery method for multi-core systems with under-provisioned on-chip voltage regulators has been proposed in literature. The power delivery network is reconfigurable at run-time to meet the varying current demands of the cores exceeding the maximum output current rating of the voltage regulators. In this paper, a real-time workload scheduling heuristic is developed that assigns the tasks to the cores such that the total load current consumption of the cores is always less than the total current capability of the under-provisioned on-chip voltage regulators. In addition, the energy-efficient scheduling of the tasks on to the cores ensures that the reconfiguration of the power delivery network is minimized. The heuristic includes DVFS management based on the unique constraints of the under provisioned voltage regulators. The work load scheduler is evaluated on homogeneous and heterogeneous multi-core platforms based on the Exynos 5410 big.LITTLE architecture. The proposed workload scheduler along with the run time voltage regulator clustering algorithm proposed in the literature provides a robust cross-layer power management technique for under-provisioned on-chip power delivery.

Keywords

low power scheduling, real-time scheduling, on-chip voltage regulation, under-provisioned power delivery, low-power design, power-aware systems.

1. INTRODUCTION

Energy efficiency has emerged as a critical design parameter in multi-core or chip multi-processor (CMP) systems. Apart from increased energy efficiency, power delivery through on-chip voltage regulators (OCVRs) offers several benefits such as reduced latency to apply DVFS, point of load power delivery with minimal power supply noise, and reduced I/O pin count devoted to power and ground signals [1, 2]. The peak power consumption and worst case power supply noise transient in a CMP determine the design of the power delivery system. Conventionally, the power rating and the design

topology of the OCVR is selected based on the maximum possible power consumption of the load circuit. The work done in [3, 4] demonstrates that the conventional method to design the power delivery network is over-provisioned by at least an order of magnitude of the maximum output current rating of the OCVRs. By under-provisioning the OCVRs to meet the typical or average need of the load circuits, the energy efficiency of the CMP system is increased by upto 44% [3]. A reconfigurable power delivery network with run-time clustering of the outputs of the OCVRs is proposed in [3] (refer to Fig. 1). The algorithm developed in [3] for run-time OCVR clustering is an example of supply side load management. The on-chip power management unit reconfigures the connections between the OCVRs and the cores to meet the changing load current demands of the cores. The run-time reconfiguration of the power delivery network operates under a power constraint. The total power demanded at any time instant by the cores is less than the total power delivery capability of the OCVRs. The power constraint is expressed mathematically in (1) for a CMP with N cores and N OCVRs. $I_{sense,x}$ and V_x are, respectively, the sensed load current and the operating voltage of each core x . I_{avg} and $V_{dd,m}$ are, respectively, the maximum output current and the maximum supported power supply voltage level of each OCVR.

$$\sum_{i=1}^N V_x \cdot I_{sense,x} < N \cdot V_{dd,m} \cdot I_{avg} \quad (1)$$

In this paper, an energy optimized work load scheduling technique is developed which relaxes the power constraint (1) on the run-time OCVR clustering algorithm developed in [3]. Low power workload scheduling on heterogeneous processors is a widely researched field [5, 6], albeit the workload schedulers in literature are oblivious of the power lost in the DC-DC converters delivering regulated power supply to the cores. The workload scheduler described in this paper is a demand side load management technique. Workload schedulers are classified into three categories [7]: best effort scheduling, with acceptance test, and robust scheduling. The proposed heuristic imposes an acceptance test on each incoming task in the system and schedules it on to one of the cores only if it meets the power constraint of the under-provisioned power delivery system. Workloads running on a CMP system are either controllable loads with soft deadlines or non-controllable loads with hard deadlines. The rescheduling of controllable tasks reduces the energy consumption of the CMP system for a given scheduling cycle. Real time applications fall under the category of non-controllable loads as they impose a hard or firm deadline. In the case of non-real time tasks with soft

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

GLSVLSI '17, May 10-12, 2017, Banff, AB, Canada.

© 2017 ACM. ISBN 978-1-4503-4972-7/17/05...\$15.00.

DOI: <http://dx.doi.org/10.1145/3060403.3060498>

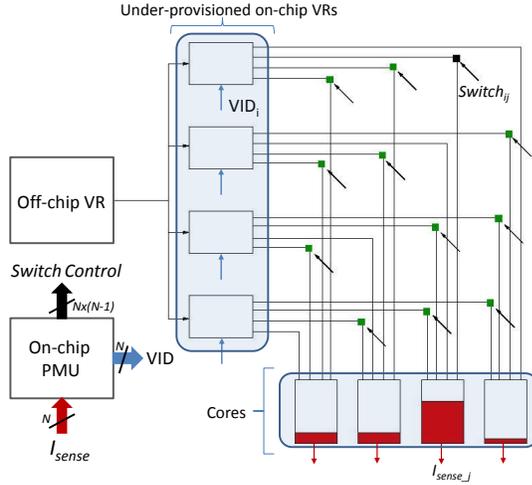


Figure 1: Proposed interconnected on-chip power delivery network in [3]. The on-chip voltage regulators are designed for a maximum output current equal to the average load current demand of workloads executed on the cores (I_{avg}). An algorithm for run-time voltage regulator clustering through a switching fabric is proposed in [3] to meet greater than I_{avg} load current demand.

deadlines and fixed priority, the tasks which violate the power constraint given by (1) are executed in the next scheduling cycle, leading to a performance penalty.

The energy consumed by a taskset on a processing element is a convex function of the computational capacity of the processing element and the task execution time. A convex energy optimization problem is solved to ensure the reliability of the proposed reconfigurable power delivery system with underprovisioned on-chip voltage regulators in [3]. The optimization problem is constrained by the total power budget of the CMP and is limited to the peak current rating of the OCVRs. The feasibility of the solution, determined by solving the optimization problem, is demonstrated through a real time workload scheduling heuristic. The scheduler is applicable to homogeneous and heterogeneous CMPs.

The rest of the paper is organized as follows: The models developed for the CMP system, OCVRs, real time periodic taskset and power consumption of the cores are described in Section 2. The convex energy optimization problem for workload scheduling is described in 3. An energy efficient workload scheduling heuristic is described in Section 4. The evaluation of the workload scheduling heuristic on homogeneous and heterogeneous CMP platforms is provided in Section 5. Concluding remarks are provided in Section 6.

2. SYSTEM MODEL AND NOTATIONS

The under-provisioned CMP system includes a set of processing elements or cores and per-core on-chip voltage regulators. The models constructed for the core architecture, CMP platform, voltage regulators, real time periodic tasks, and the power consumption of the cores are described in Subsections 2.1 through 2.4.

2.1 CMP models

CMP systems with homogeneous core configurations as well as heterogeneous configuration are developed to analyze the work load scheduler. The homogeneous CMP includes

Table 1: Frequency (MHz) and Voltage (V) pairs used by the DFVS procedure in *Algorithm 1*. The bold values listed in the table are the nominal voltages and frequencies.

big core (A15)	LITTLE core (A7)
1800/1.250	1200/1.225
1700/1.200	1100/1.125
1600/1.162	1000/1.100
1500/1.137	900/1.037
1400/1.100	800/0.987
1300/1.062	700/0.950
1200/1.025	600/0.950
1100/1.000	500/0.950
1000/0.962	400/0.950
900/0.925	300/0.950
800/0.900	200/0.950

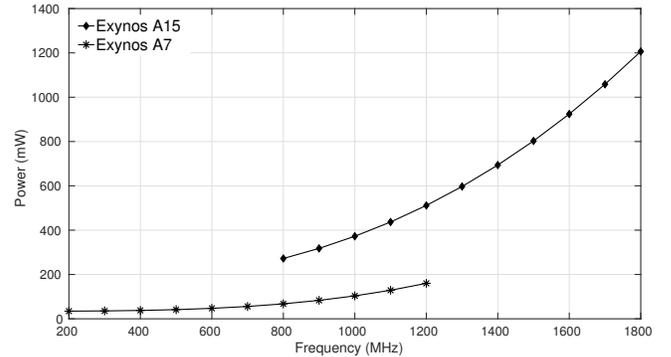


Figure 2: The power consumption of the Exynos big,LITTLE cores with frequency based on the model given by (2). The power model parameters are validated in [10].

processing elements based on the ARM A15 core integrated in the Samsung Exynos 5410 platform [8, 9]. The parameters used in constructing a 16 core homogeneous CMP platform are listed in Table 2. An eight core heterogeneous CMP with four ARM A15 and four A7 cores from the Exynos 5410 platform is also evaluated. The DVFS levels applied to the cores are listed in Table 1. The variation in the power consumption of the core with frequency, based on the power model given by (2) and validated in [10], is shown in Fig. 2.

2.2 Power model

The power consumption of a processing element π_j is approximated as a function of frequency, similar to work done in [10]. The power consumed by any processing element is given by (2). The $\kappa * f^\alpha$ and β terms in (2) represent, respectively, the dynamic and static power consumption of the cores. The model parameters κ , α , and β for the Samsung Exynos A15 and A7 processors [10] are used to validate *Algorithm 1*. The power consumption with frequency using the estimated model parameters is shown in Fig. 2.

$$P(f) = \kappa * f^\alpha + \beta \quad (2)$$

2.3 Voltage regulator models

An on-chip power delivery network with per core voltage regulation is considered for the CMP platform. The on-chip

Table 2: Parameters of the CMP cores derived from the Samsung Exynos 5410 big.LITTLE architecture.

Parameter	big core (A15)	LITTLE core (A7)
Nominal voltage (V)	1.16	1.225
Nominal frequency (MHz)	1600	1200
OCVR maximum current rating (I_{avg} in mA)	800	110
Power model parameters [α_j , κ_j (mW/MHz^3), β_j (mW)]	[2.63, 2.91×10^{-6} , 146.49]	[3.28, 1.00×10^{-8} , 34.24]
Number of cores in homogeneous CMP system	16	0
Number of cores in heterogeneous CMP system	4	4

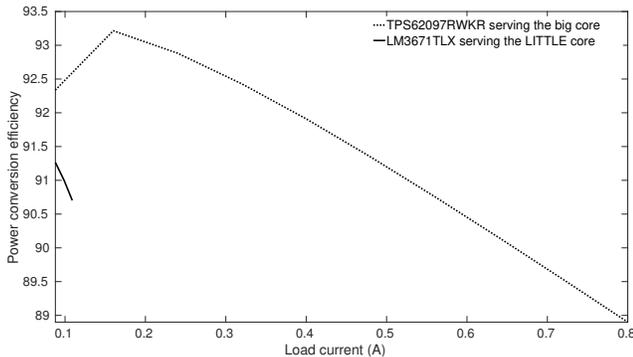


Figure 3: The power conversion efficiency of the voltage regulators serving the Exynos A15 (big) and Exynos A7 (LITTLE) cores [12].

voltage regulators (OCVRs) are modeled as DC-DC switching buck converters [11]. Buck converters with optimum power conversion efficiency based on the power consumption of the A15 and A7 cores (shown in Fig. 2) are developed using [12]. The important parameters of the buck converters for both the big and LITTLE cores are listed in Table 3. The variation in power conversion efficiency with load current is shown in Fig. 3.

2.4 Real-time periodic task model

The real-time workloads are modeled as a set of independent periodic tasks $\tau_i \in \mathcal{T}$ to be scheduled on a subset of cores of a many-core system $\pi_j \in \Pi$ [10]. Each task τ_i , has a hard deadline of D_i . Each core π_j supports distinct DVFS levels $V_x \in [V_{dd,1}, V_{dd,2}, \dots, V_{dd,m}]$ and $f_x \in [f_1, f_2, \dots, f_m]$. A task τ_i with a hard deadline D_i requires at most $C_{i,j}$ cycles to execute on a core π_j at the highest supported voltage $V_{dd,m}$ and frequency f_m . The context switching overhead and overhead due to resource sharing amongst tasks which remain unresolved after task partitioning is included in $C_{i,j}$. The computational capacity required by task τ_i on core π_j is defined as $u_{i,j} = \frac{C_{i,j}}{D_i}$. The subset of tasks T_j that are executed on core π_j therefore require a total computational capacity of $U_j = \sum_{\tau_i \in T_j} u_{i,j}$ cycles per second.

3. OPTIMAL WORKLOAD SCHEDULING

An optimization problem is defined to partition and schedule real time workloads on a many-core platform. A specific set of constraints unique to the proposed reconfigurable PDN are considered, which account for the use of underprovisioned on-chip voltage regulators. The objective of the optimization problem is to minimize the energy consumption of the many core platform, including the power consumed by the

OCVRs. The energy consumed by the system in a given scheduling period T_{epoch} is given by (3), where $P(U_j)$ is the power consumed by the core π_j with computational capacity U_j to execute the scheduled task set, and PCE_{U_j} is the combined power conversion efficiency of the OCVR(s) supplying current to the core π_j . The workload scheduling is constrained by the total computational capacity available to execute the taskset U_j on π_j , where the total capacity must exceed the computational demand of the taskset as given by (4). In addition, the operating frequency must fall within the supported frequency range of the cores as given by (5). The total power consumed by the cores at any time instant must be less than the combined maximum power supported by all OCVRs in the system as described by (6).

$$\min_{U_j} \sum_{\pi_j \in \Pi} \frac{P(U_j)}{PCE_{U_j}} \cdot T_{epoch} \quad (3)$$

$$\text{s.t.} \quad \sum_{\pi_j \in \Pi} U_j \geq \sum_{\tau_i \in \mathcal{T}} u_i \quad (4)$$

$$f_{1,j} \leq U_j \leq f_{m,j} \quad \forall \pi_j \in \Pi \quad (5)$$

$$\sum_{\pi_j \in \Pi} P(U_j) < N \cdot V_{dd,m} \cdot I_{avg} \quad (6)$$

4. WORKLOAD SCHEDULING HEURISTIC

A heuristic is described in this section, which performs the real time workload scheduling on the cores for the optimization problem developed in Section 3. The heuristic consists of three procedures: *PARTITION*, *DVFS*, and *SCHEDULE*. The *PARTITION* procedure is an evolution of the *Marginal Power Heuristic (M-PWR)* developed in [10]. Optimal workload partitioning is achieved by incrementing the load on each core such that the constraint given by (5) is not violated. The tasks $\tau_i \in \mathcal{T}$ are first sorted in decreasing order of the maximum computational demand $u_{i,j}$ on cores $\pi_j \in \Pi$. A task is assigned to a core if the scheduling of the task results in the least increase in the power consumption. The output from the procedure is a scheduled taskset Θ_j on each core.

The *DVFS* procedure reduces the operating frequency and the voltage of the cores until the constraint given by (6) is satisfied. The right hand side of (6) is a constant value equal to the total power P_{total} of the CMP. Expressing the total power consumed by the cores with the power model given by (2) in constraint (6) provides a limit to the operating frequency of the cores raised to the power α (refer to constraint (8)). The use of the *DVFS* procedure results in the optimal frequency of operation for each core by solving the bounded knapsack problem. The deadline of each task in the taskset Θ_j is analogous to the value of the item in the knapsack. The required computational demand at a given frequency f_j

Table 3: Operating values of the switching DC-DC buck converters [12] serving the ARM A15 and A7 cores.

Device	LM3671TLX	TPS62097RWKR
Input voltage (V)	2.5	2.5
Output voltage range (V)	0.9 to 1.3	0.9 to 1.3
Maximum output current (mA)	110	800
Efficiency at maximum output current	90.7	88.9
Peak to peak inductor ripple current (mA)	146.93	357.24
Switching frequency (MHz)	2	1.69
Duty cycle (%)	46.18	47.44
Peak-to-peak output ripple voltage (mV)	1.322	3.215
Total power dissipation (mW)	12.4	109.88
Footprint (mm^2)	37	93

Table 4: Parameters to generate real time periodic tasksets.

Parameter	Value
Number of tasks (N_i)	[32, 48, 64, 80]
Task utilization range	[0.1 to 0.9]
Task period range in seconds (T_i)	[10 to 100]
Taskset utilization factor (ρ)	[0.1 to 1]

Algorithm 1 Real time workload partitioning and scheduling on a many-core system with underprovisioned OCVRs.

Inputs:

Set of real time tasks: \mathcal{T}

Set of N cores in the many-core system: Π

Outputs: Schedulable taskset (Θ_j) on each core $\pi_j \in \Pi$ with assigned voltage $V_j \in [V_{dd,1}, V_{dd,2}, \dots, V_{dd,m}]$ and frequency $f_j \in [f_1, f_2, \dots, f_m]$

procedure PARTITION(\mathcal{T}, Π)

for each $\pi_j \in \Pi$ **do**

$\Theta_j \leftarrow \emptyset, U \leftarrow 0$

end for

$\mathcal{T}' \leftarrow \text{SORT}(\mathcal{T}$ by descending $max_j u_{i,j}$)

for each $\tau_i \in \mathcal{T}'$ **do**

$\Pi' \leftarrow j: U_j + u_{i,j} < f_{m,j} \triangleright$ Cores on which τ_i is schedulable

if $\Pi' = \emptyset$ **then return** Failed to schedule

end if

$k \leftarrow \arg \min_{j \in \Pi'} P_j(U_j + u_{i,j}) \triangleright$ core id on which τ_i consumes least power

$\Theta_k \leftarrow \Theta_k \cup \tau_i \triangleright \Theta_k$ is the schedulable set of tasks on π_k

$U_k \leftarrow U_k + u_{i,j}$

end for

return Θ

end procedure

procedure DVFS(Θ)

while $\sum_{\pi_j \in \Pi} f_j^\alpha > (P_{total} - N \cdot \beta) / \kappa$ **do**

for each $\pi_j \in \Pi$ **do**

while $\sum_{\tau_i \in \Theta_j} u_{i,f_j} \leq f_m$ **do**

$f_j \leftarrow (f_x \mid f_x \in F \text{ and } f_x \leq f_j) \triangleright$ lower the operating frequency to one of the supported DVFS levels $F = (f_1, f_2, \dots, f_m)$

end while

end for

end while

end procedure

procedure SCHEDULE(Θ_j, f_j)

$k \leftarrow \arg \min_{\tau_i \in \Theta_j} D_i$

$\pi_j \leftarrow \tau_k$

end procedure

on processor π_j is u_{i,f_j} . The weight added to the knapsack is analogous to u_{i,f_j} . The objective of the knapsack problem is to maximize the number of tasks executed on a core, without violating the task deadline. The procedure lowers the operating frequency of each task until constraints (7) and (8) are satisfied. Once the operating frequency of each task in Θ_j is determined, the *SCHEDULE* procedure schedules the tasksets on each core based on an earliest deadline first policy.

$$\sum_{\tau_i \in \Theta_j} u_{i,f_j} \leq f_m \quad (7)$$

$$\sum_{\pi_j \in \Pi} f_j^\alpha \leq (P_{total} - N \cdot \beta) / \kappa \quad (8)$$

5. SIMULATION RESULTS

Real time periodic tasks with implicit deadlines are considered to determine the efficacy of the proposed task scheduler. The task scheduling is performed for one hyperperiod of the taskset T_{epoch} , which is the least common multiple of the implicit deadlines of all tasks $\tau_i \in \mathcal{T}$. The tasks are generated with the parameters listed in Table 4. The computational capacity ($u_{i,j}$) of the tasks is selected as a random variable with a uniform distribution between 0.1x to 0.9x the maximum supported operating frequency of the cores in the CMP (maximum frequency f_m of 1800 MHz). The total computation time requested by the taskset in a hyperperiod is less than the available time on the processing elements to prevent system overload. This ensures that the taskset utilization factor or the system load is less than 1 ($\rho < 1$).

The resulting task schedule, from execution of *Algorithm 1* on a homogeneous CMP platform with 16 cores configured as Exynos 5410 A15s, is shown in Fig. 4. The task scheduling is constrained due to the limited power budget of the under-provisioned voltage regulators. For a maximum output current I_{avg} of 1 A, the percentage of tasks scheduled by

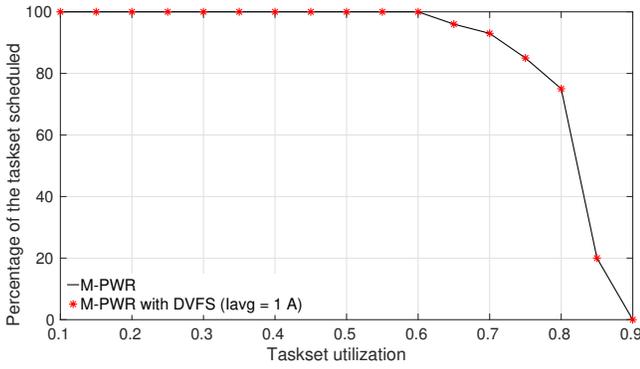


Figure 4: Percentage of tasks successfully partitioned by the M-PWR heuristic [10] and successfully scheduled by *Algorithm 1* in a 16 core homogeneous CMP. Task scheduling through *Algorithm 1* matches the M-PWR heuristic with the maximum output current rating of the voltage regulator set to 1 A.

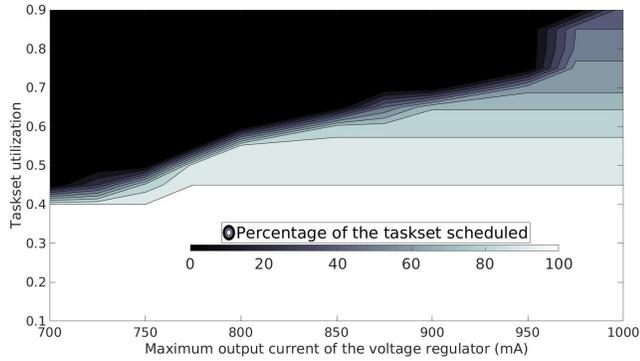


Figure 5: A contour plot of percentage of tasks successfully scheduled by *Algorithm 1* with varying taskset utilization and maximum output current of the voltage regulators in a homogeneous CMP with 16 cores.

Algorithm 1 is identical to the M-PWR heuristic in [10]. The execution of *Algorithm 1* is further characterized on a homogeneous platform with voltage regulators of varying maximum output current I_{avg} . The results are shown through the contour plot in Fig. 5. The percentage of tasks scheduled for a given taskset utilization factor decreases as the maximum output current of the voltage regulators is reduced. For a voltage regulator designed with a maximum output current I_{avg} of 0.7 A, the percentage of tasks scheduled matches the M-PWR heuristic [10] upto a taskset utilization factor ρ of 0.65.

The workload scheduler is also evaluated on a heterogeneous CMP platform with four Exynos A15 (big) and four Exynos A7 (LITTLE) cores. For a randomly chosen taskset hyperperiod, the task distribution and corresponding computational demand ($u_{i,j}$ of each task) is shown in Fig. 6. There are 11 tasks assigned to the big core cluster and five to the LITTLE core cluster. The maximum output current of the voltage regulators serving each of the big cores is set to 800 mA and the voltage regulators serving each of the LITTLE cores to 110 mA. The frequency assigned to each core to meet the constraint given by (8) is determined and shown

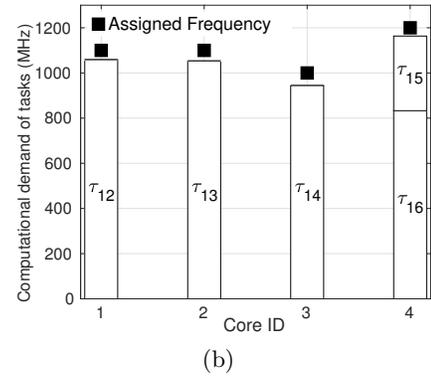
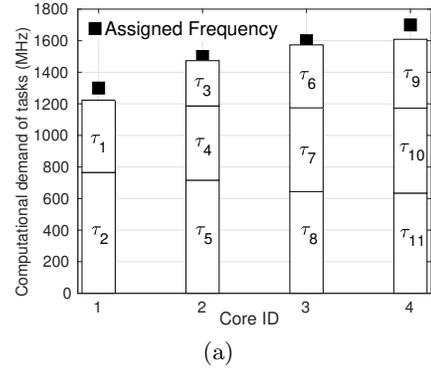


Figure 6: A snapshot of the task assignment on a heterogeneous CMP platform with (a) big cores modeled on A15 parameters, and (b) LITTLE cores modeled on A7 parameters. The maximum output currents of the voltage regulators serving each of the big cores and LITTLE cores are, respectively, 800 mA and 110 mA.

in the Fig. 6. Depending on the total computational demand of the tasks assigned to each core, the frequency is lowered from the maximum supported frequency of 1800 MHz for the big cores and 1200 MHz for the LITTLE cores. The task partitioning performed by the *PARTITION* procedure further improves power efficiency by preferentially assigning tasks to the LITTLE cores, which meet the task utilization constraint given by (4). Consequently, for an identical scaling factor of the peak output current of the OCVRs (I_{avg}/I_{peak}) serving the LITTLE core and the big core, the percentage of tasks scheduled through the *DVFS* procedure is lower for the LITTLE cores as compared to the big cores. As the LITTLE core cluster has a load current range of 100 mA, the scaling factor of the maximum output current of the voltage regulators serving the LITTLE cores is set to a larger value than that for the big cores to achieve a high task schedulability on the heterogeneous platform.

The task scheduling results on the homogeneous and heterogeneous CMP platform demonstrate that the proposed workload scheduler in tandem with the runtime on-chip voltage regulator clustering algorithm developed in [3], offer an efficient and robust cross layer energy optimization mechanism for CMPs with underprovisioned on-chip voltage regulators.

6. CONCLUSIONS

A real-time workload mapping heuristic is developed to minimize the reconfiguration of the power delivery network

with under-provisioned on-chip voltage regulators. The scheduled tasks are assigned optimum DVFS levels for each core. The heuristic is evaluated on homogeneous and heterogeneous CMP platforms with real time periodic tasks. The schedulability of the tasks with varying taskset utilization is compared against M-PWR, an energy efficient workload scheduler. The workload mapping heuristic in conjunction with the run-time reconfiguration of the power delivery network ensure reliable and energy efficient operation of the CMP with on-chip voltage regulators designed for only the typical or average load current demand.

7. REFERENCES

- [1] M. Tavana, D. Pathak, M. Hajkazemi, I. Savidis, and H. Homayoun, "Realizing Complexity-Effective On-Chip Power Delivery for Many-Core Platforms by Exploiting Optimized Mapping," *Proceedings of the IEEE International Conference on Computer Design*, pp. 581–588, October 2015.
- [2] W. Kim, M. S. Gupta, G. Wei, and D. Brooks, "System Level Analysis of Fast, Per-Core DVFS Using On-chip Switching Regulators," *Proceedings of the International Symposium on High Performance Computer Architecture*, pp. 123–134, February 2008.
- [3] D. Pathak, M. Hajkazemi, M. Tavana, H. Homayoun, and I. Savidis, "Load Balanced On-Chip Power Delivery for Average Current Demand," *Proceedings of the Great Lakes Symposium on VLSI*, pp. 439–444, May 2016.
- [4] D. Pathak, M. Hajkazemi, M. Tavana, H. Homayoun, and I. Savidis, "Energy Efficient On-Chip Power Delivery with Run-Time Voltage Regulator Clustering," *Proceedings of the IEEE International Symposium on Circuits and Systems*, pp. 1210–1213, May 2016.
- [5] T. Li, D. Baumberger, D. A. Koufaty, and S. Hahn, "Efficient Operating System Scheduling for Performance-Asymmetric Multi-Core Architectures," *Proceedings of the ACM/IEEE conference on Supercomputing*, , No. 53, November 2007.
- [6] S. Ghiasi, T. Keller, and F. Rawson, "Scheduling for Heterogeneous Processors in Server Systems," *Proceedings of the Conference on Computing Frontiers*, pp. 199–210, May 2005.
- [7] G. Buttazzo, *Hard Real-Time Computing Systems: Predictable Scheduling Algorithms and Applications*, Vol. 24, Springer Science & Business Media, 2011.
- [8] H. Chung, M. Kang, and H. Cho, "Heterogeneous Multi-Processing Solution of Exynos 5 Octa with ARM® big.LITTLE Technology," *Samsung White Paper*, 2012.
- [9] Y. Shin, H.J. Lee, K. Shin, P. Kenkae, R. Kashyap, D. Seo, B. Millar, Y. Kwon, R. Iyengar, M. Kim, *et al.*, "28nm high-K Metal Gate Heterogeneous Quad-Core CPUs for High-Performance and Energy-Efficient Mobile Application Processor," *Proceedings of the IEEE International SoC Design Conference*, pp. 198–201, November 2013.
- [10] A. Colin, A. Kandhalu, and R. Rajkumar, "Energy-Efficient Allocation of Real-Time Applications onto Heterogeneous Processors," *Proceedings of the IEEE International Conference on Embedded and Real-Time Computing Systems and Applications*, pp. 1–10, August 2014.
- [11] E. Salman and E. G. Friedman, *High Performance Integrated Circuit Design*, McGraw Hill, 2012.
- [12] Texas Instruments, "WEBENCH Design Center," <http://webench.ti.com>.