Article

Idleness-Aware Dynamic Power Mode Selection on the i.MX 7ULP IoT Edge Processor

Alfio Di Mauro [1],*[iD], **Hamed Fatemi** [2], **Jose Pineda de Gyvez** [3] **and Luca Benini** [1]

1   Integrated Systems Laboratory, ETHZ, 8092 Zurich, Switzerland; lbenini@iis.ee.ethz.ch
2   NXP Semiconductor, San Jose, CA 95134, USA; hamed.fatemi@nxp.com
3   NXP Semiconductor, 5656 Eindhoven, The Netherlands; jose.pineda.de.gyvez@nxp.com
*   Correspondence: adimauro@iis.ee.ethz.ch

check for
updates

Abstract: Power management is a crucial concern in micro-controller platforms for the Internet of Things (IoT) edge. Many applications present a variable and difficult to predict workload profile, usually driven by external inputs. The dynamic tuning of power consumption to the application requirements is indeed a viable approach to save energy. In this paper, we propose the implementation of a power management strategy for a novel low-cost low-power heterogeneous dual-core SoC for IoT edge fabricated in 28 nm FD-SOI technology. Ss with more complex power management policies implemented on high-end application processors, we propose a power management strategy where the power mode is dynamically selected to ensure user-specified target idleness. We demonstrate that the dynamic power mode selection introduced by our power manager allows achieving more than 43% power consumption reduction with respect to static worst-case power mode selection, without any significant penalty in the performance of a running application.

Keywords: edge devices; power management; energy efficiency

1. Introduction

Efficient energy management is very challenging in Internet of Things (IoT) edge devices [1]. On one hand there is the increasing demand of more near-sensor computing capabilities, on the other hand, strict constraints have to be set on the power consumption to maximize the lifetime of an IoT node, which is in many cases battery-supplied.

Researchers have responded to this challenge by proposing new SoC architectures, e.g., parallel-ultra-low-power multi-core computing platforms, and power management strategies [2]. Similar approaches are proposed also by industry, e.g., by adopting heterogeneous multi-core architectures where a low-power control core (e.g., Arm® Cortex®-M4 core (Arm and Cortex are registered trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere)) with modest computing capabilities is coupled with a more capable core (e.g., Arm Cortex-A7 core) for compute-intensive tasks.

In this scenario, well-known techniques such as clock-gating and power-gating are widely used to minimize the power consumption of inactive sub-modules of an SoC. However, restoring the functionality of SoC subsystems that are clock-gated or power-gated during energy saving states could require a non-negligible amount of time and energy. Additionally, in many event-driven applications, requiring fast time response or continuous event processing, duty-cycling computation phases, or even occasionally entering a sleep state can be unfeasible.

Shut-down-based power management can be complemented by more advanced energy saving techniques such as Dynamic Voltage and Frequency Scaling (DVFS). DVFS is the process of adapting, at run-time, the frequency and the supply voltage of a digital circuit; typically, this is obtained in a closed loop regulation, using as a feedback parameters such as core workload or desired core idleness.

The dynamic power consumption of digital circuits has linear dependency from the frequency, and it is in a quadratic relationship with the supply voltage. Therefore, such a technique is very effective to significantly reduce the energy consumption of digital circuits.

Unfortunately, the hardware and software infrastructure for DVFS available in heterogeneous edge devices is not yet as mature as the one integrated into many high-end and mobile multi-core application processors. Such development is not trivial since, in sharp contrast with high-end multi-cores, dual-core MCUs have most of the times heterogeneous ISAs, non-uniform memory hierarchy and neither cache coherency nor shared virtual memory support.

The NXP i.MX 7ULP is a low-power low-cost Arm Cortex-M4/Cortex-A7-based SoC which belongs to this device category. The chip has been fabricated in 28 nm Fully Depleted Silicon On Insulator (FD-SOI) technology, and it features an advanced power management infrastructure that enables dynamic power mode adaptation. The main contribution of this paper is the development and qualification of the first (to our knowledge) DVFS-based power management software infrastructure for this exemplary heterogeneous dual-core MCUs. Our power manager achieves up to 45% of power consumption reduction during the active state of the SoC, without any penalty on the application execution performance.

The remainder of this paper is organized as follows. Section 2 gives an overview of the related work. Section 3 describes the i.MX 7ULP SoC. Section 4 describes the power manager. In Section 5 we describe the experimental setup used to characterize the power manager. Section 6 presents the results in terms of power consumption reduction, while Section 7 provides concluding remarks.

2. Related Work

Among several techniques to increase the energy efficiency, Dynamic Voltage and Frequency Scaling (DVFS) represents a well established, effective [3] and low-complexity strategy. Despite technological scaling, which reduces the margins for supply voltage regulation, DVFS still represents a viable energy saving techniques for modern platforms operating near threshold [4]. Those platforms reach high energy efficiencies thanks to a combination of voltage scaling and adaptive body biasing to avoid performance degradation [5,6]. In [7,8], authors present two approaches where DVFS is controlled by dedicated hardware units on embedded microprocessors. Similarly, in [9] DVFS is applied on high-end CPUs for cloud infrastructures.

More sophisticated techniques rely on gathering information on the status of the system to guarantee a desired quality of service. In [10], authors exploit a combination of DVFS and workload profiling in a closed loop configuration.In [11,12], to optimize the efficiency of Linux-based systems, DVFS is operated in combination with advanced power-aware scheduling algorithms that exploit a combination of system-level statistics. Finally, in [13] machine learning algorithms are used to gather higher level information from the applications running on the core [13].

In this paper, we propose a power management unit implementation that takes inspiration from the power management policies surveyed above and adopted on many Linux-capable application processors. The proposed power manager can perform DVFS on a new low-cost low-power ARM A7-M4 SoC architecture for IoT edge (e.g., the NXP i.MX 7ULP SoC). Compared to the available power management implementations [14], our approach targets lower-end devices where techniques such as DVFS are typically not applied because of the lack of hardware/software support. As with what happens in high-end CPUs based systems, the proposed power management gathers system level information, e.g., the idleness of the system, and exploit this information to optimize the use of available energy. The regulation of the power mode is operated by a software implemented power manager running in a low-power real-time domain of the SoC. Unlike application-specific DVFS techniques used in special-purpose accelerators, our approach relies on minimal knowledge of the system, since it exploits very high-level information such as the CPU idle time, making it suitable to be ported on different platforms with similar hardware features.

J. Low Power Electron. Appl. **2020**, *10*, 19

3 of 8

3. The i.MX 7ULP Platform

i.MX 7ULP is a new low-cost low-power Arm Cortex-M4/Cortex-A7-based SoC architecture for IoT edge. The chip has been developed by NXP and fabricated in 28 nm FD-SOI technology. The architecture of the microprocessor is divided into two processing domains hosted by two separate power domains. (i) The application domain is built around an Arm Cortex-A7 core. This processing domain can boot Linux-based embedded operating systems, offering high flexibility for application development. The main target of this processing domain are general purpose and computationally demanding workloads. (ii) The real-time domain is built around an Arm Cortex-M4 core. This domain targets low-power low-complexity sections of an application that presents strict requirements in terms of predictable tasks execution time, e.g., timing-critical tasks such as sensor sampling and IO event handling.

The SoC features flexible hardware support for power management. Each of the two power domains has a dedicated register set that can be used to store pre-defined parameters such as supply voltage, clock frequency divider and body-biasing. The switching between pre-configured power modes is governed by a dedicated System Mode Controller (SMC).

4. Power Manager

The goal of the power manager proposed in this paper is to minimize the power consumption when the system is in an active state, all the buses are clocked, and cores have complete access to all peripherals. In the rest of the paper, we will refer to this system configuration as *RUN* power state. The power manager operates an opportunistic voltage and frequency scaling on the application power domain to reduce the energy consumption when the average workload of the A7 core is lower than 100%. In this context, a statically selected power mode, tailored for a worst-case workload, is inefficient, and the same task could be executed operating the core at a lower voltage and frequency.

To limit the intrusiveness of the approach, and to have a deterministic time response to workload variations, we implemented the power manager on the real-time application domain. More specifically, the power manager is part of the firmware application running on the M4 core, which serves as a software Power Management Unit (PMU). Figure 1 illustrates the block diagram of the power manager and the main sub-modules.
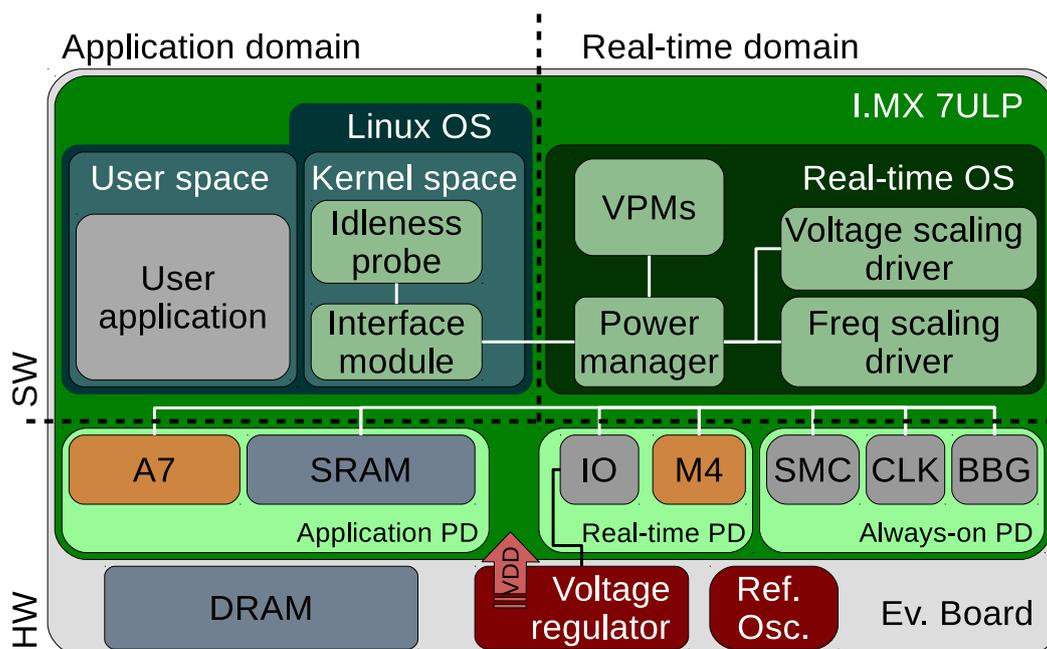

Figure 1. Power manager software architecture block diagram and Hardware/Software partitioning.

To operate the DVFS, we statically selected the RUN power mode. In this power state, the system uses the main PLL to generate a reference clock frequency in the range from 800 MHz to 300 MHz; the forward body-bias can be applied to the transistors, the RAM is powered on, and the system can tolerate DVFS. In this scenario, we introduced multiple *Virtual Power Modes (VPM)*. Each VPM is a set of values that the power manager uses to override configuration values for Voltage, clock Frequency and Body-bias activation of the *RUN* power mode. The number of VPMs can be arbitrarily configured by the user, their configuration values are stored in the M4 data memory.

The criterion on which the VPM of the system is switched is the idleness of the A7 core. This information is obtained directly from the operating system CPU statistics. The power manager compares the measured idleness with a user-specified target idleness, and determines how to adjust the performance of the system to match the target idleness. Algorithm 1 describes the performance adjustment procedure. The idleness observation time can vary in a range from 100 ms to 5 s depending on the application requirements.

Algorithm 1 This pseudo-code procedure illustrates the steps that are performed by the power manager at every iteration of the power mode regulation.

 1: **while** !new power mode request **do**
 2:　　wait for a new request
 3: **end while**
 4: **if** target idleness > curr. idleness **then**
 5:　　increase supply voltage
 6:　　**while** !Supply voltage change **do**
 7:　　　wait
 8:　　**end while**
 9:　　increase clock freq.
10:　　**while** !clock freq. change **do**
11:　　　wait
12:　　**end while**
13: **else**
14:　　decrease clock freq.
15:　　**while** !clock freq. change **do**
16:　　　wait
17:　　**end while**
18:　　decrease supply voltage
19:　　**while** !Supply voltage change **do**
20:　　　wait
21:　　**end while**
22: **end if**
23: power mode transition complete

4.1. Real-Time Domain

The kernel of the power manager is part of the firmware running on the real-time domain, and it is constituted by 4 main sub-modules that operate synchronously to perform (i) the idleness acquisition from the application domain, (ii) the target versus current idleness comparison, (iii) the power mode selection and the power mode configuration registers modification, (iv) the voltage and the frequency scaling.

4.2. Application Domain

The power manager uses the idleness of the core in a given time interval preceding the power mode regulation as a feedback signal. This information is made available by the operating system that collects statistics on the use of the core. To retrieve this information and transfer it to the real-time domain, we developed (i) a module to track the idleness and (ii) a Linux-driver module that acts as an interface with the power manager.

5. Experimental Setup

In this section, we describe the experimental setup for the measurements presented in Section 6. In our experiments, we focused on the dynamic reduction of the energy consumption of the application power domain only, since it represents more than 90% of the power consumption of the entire SoC. Operating frequency and voltage of the real-time power domain have been statically selected as the minimum values that allow accomplishing the power management task.

In our power management implementation, we introduced six virtual power modes available for dynamic power mode selection. Table 1 reports the configuration values related to the core supply voltage and clock frequency of every VPM. Supply voltages have been arbitrarily chosen according to a quadratic curve, operating frequencies have been selected as the maximum ones achievable by the core at the related supply voltages. To maximize performance, forward body-bias is active in all VPMs [15]. To verify that the system was able to operate at a given frequency, we executed several benchmarks on the core, selected among those provided by the *stress-ng* benchmark suite [16].

Table 1. Configuration values of Virtual Power Modes (VPM).

Power Mode ID	VPM0	VPM1	VPM2	VPM3	VPM4	VPM5
Voltage [mV]	1100	1000	900	860	835	800
Frequency [MHz]	800	640	490	420	370	300

The power mode regulation period has been chosen as 5 s. This value can be modified to adjust responsiveness of the controller to the idleness variation, the minimum regulation period is approximately of 100 ms; this value is bounded by the communication latency with the external voltage regulator hosted on the i.MX 7ULP evaluation board and does not constitute an intrinsic limitation of the i.MX 7ULP system.

To fully characterize the power manager and evaluate the benefits in terms of power saving we performed the following measurements on the i.MX 7ULP evaluation board:

(i) we fixed the VPM, and we measured its power consumption. We performed the measurements for each of the VPMs reported in Table 1, and, at each VPM, we swept the CPU usage in a range of 5% to 95%. The desired load of the core has been precisely changed by varying the number of tasks launched in a fixed time period of 5 s. The use information of the core has been obtained by accessing the core use and idleness statistic made available by the Linux kernel. In our experiment, we verified that by executing a small number of tasks the core use was below 5%, and we linearly increased the number of task launched on the core within the reference period until the core use reached 95%.

(ii) we evaluated the power consumption at different workload levels when the controller is adjusting the power mode. We analyzed the response of the power manager when the target idleness was in a range of 5% to 95%, and similarly to the previous case, the workload on the core was swept in a range of 5% to 95%.

Section 6 shows the results of the two experiments; the power consumption has been measured at the evaluation board 5 V power supply connector. The measurement conditions are the following:

- Operating temperature T = 25 °C
- Test benchmark for operating frequency validation and power estimation = stress-ng test suite
- Power measurement equipment = Keysight N6705B power analyzer (time resolution of 20 μs)

6. Results

In this section, we present the results in terms of power consumption reduction when the power manager is active.

Figure 2a shows the power consumption of the i.MX 7ULP evaluation board when a VPM is fixed. From the static analysis, we can conclude that the dynamic power mode adaptation allows saving up to 28% of power when the system is in RUN mode, and in absence of core activity.
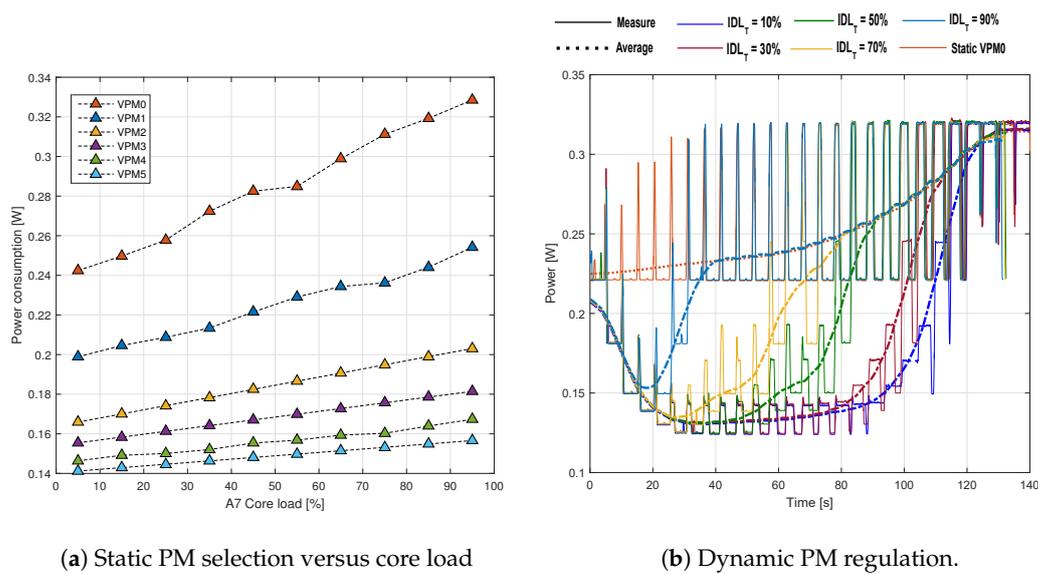
(**a**) Static PM selection versus core load　　　　　(**b**) Dynamic PM regulation.

Figure 2. Core power consumption.

The power consumption reported in the plot includes the power consumption of other components soldered on the board, e.g., the external regulator. The only contribution excluded from this analysis is the power consumption of the on-board external DRAM chip.

In the second experiment we measured the power consumption of the board when the power manager is regulating the VPM. Figure 2b reports the power consumption when the workload on the core increases linearly from from 5% (leftmost part of the plot) to 95% (rightmost part of the plot). The goal of this experiment is show, at run-time, how the PMU can promptly adapt the VPM to the workload scenario, without reducing the number of tasks completed in a reference time interval, and eventually settling on a desired average idleness level. Figure 2b shows that regardless of the selected target idleness, tasks can be released always with the same timing on the core. During the dynamic VPM adaptation, the actual core use depends on the power mode selected by the controller, which tries to restore the target idleness value. In this operating mode, the time to complete a tasks is not known in advance, because the CPU operates at a variable frequency. Please note that the quality of service, which in this context is represented by meeting the task deadlines is not degraded when the VPM is dynamically changed (i.e., the core computational bandwidth is never saturated). The Figure 2b reports that the average power consumption is always equal or lower than the case where a VPM is statically selected. Always in Figure 2b, it can be observed that the target idleness specified by the user affects the power manager behavior. Table 2 summarizes the peak power consumption reduction for several user-specified values of target idleness.

Table 2. Peak power consumption reduction with respect to static VPM0 selection.

Target Idleness	90%	70%	50%	30%	10%
Peak power reduction	32%	33%	43%	44%	45%

From the results reported in Table 2, it is evident that when the target idleness is too high with respect to the actual core workload, the power manager cannot explore the entire set of available states. On the contrary, in all those cases where the power manager can operate the system at the lowest power modes (Target idleness = 50%, 30%, 10%), it is possible to achieve more than 43% of average power consumption reduction and a peak power consumption reduction of 45%. The power consumption improvement reported in our measurements refers to the case where VPM0 was statically selected, and the core executed the same workload.

In our experiment, we assumed that the number of operation performed at each task execution is constant. In Figure 2b, we observe that every 5 s, a new set of task is released on the core, originating a spike in the power trace caused by the core wake-up. At each experiment run, which was performed with a different target idleness, the core never saturated the computational bandwidth. Therefore, the number of task executed by the core was constant for all the experiments, while the power consumption significantly decreased, resulting in less energy consumed by the core.

7. Conclusions

In this paper, we presented the implementation of a power management unit capable of operating an idleness aware dynamic power mode adaptation. We validated the proposed approach on a novel low-cost low-power heterogeneous MCU for IoT edge fabricated in 28 nm SOI technology. The proposed power manager allows achieving a theoretical power consumption reduction higher than the 28% in absence of core activity. We confirmed the expectations during the operation of the system, reporting more than 43% of average power consumption reduction when the target idleness is selected according to the real idleness of the application, and a peak power consumption reduction of 45%.

Author Contributions: Conceptualization, L.B. and A.D.; methodology, H.F. and J.P.; software, A.D.; validation, A.D.; investigation, A.D.; data curation, A.D.; writing–original draft preparation, A.D.; writing–review and editing, A.D., H.F., J.P. and L.B.; supervision, H.F., J.P. and L.B.; project administration, H.F., J.P. and L.B. All authors have read and agree to the published version of the manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

IoT	Internet of Things
MCU	Micro Controller Unit
SoC	System on Chip
DVFS	Dynamic voltage and Frequency Scaling
ISA	Instruction Set Architecture
FD-SOI	Fully Depleted Silicon On Insulator
ULP	Ultra Low Power
CPU	Central Processing Unit
SMC	System Mode Controller
PMU	Power Management Unit
PLL	Phase Locked Loop
VPM	Virtual Power Mode

References

1. Conti, F.; Rossi, D.; Pullini, A.; Loi, I.; Benini, L. PULP: A Ultra-Low Power Parallel Accelerator for Energy-Efficient and Flexible Embedded Vision. *J. Signal Process. Syst.* **2015**, *84*, 339–354. [CrossRef]
2. Rossi, D.; Pullini, A.; Loi, I.; Gautschi, M.; Gurkaynak, F.K.; Teman, A.; Constantin, J.; Burg, A.; Miro-Panades, I.; Beignè, E.; et al. 193 mops/mw @ 162 mops, 0.32 v to 1.15 v voltage range multi-core accelerator for energy efficient parallel and sequential digital processing. In Proceedings of the 2016 IEEE Symposium in Low-Power and High-Speed Chips (COOL CHIPS XIX), Yokohama, Japan, 20–22 April 2016; pp. 1–3.
3. Qu, G. What is the limit of energy saving by dynamic voltage scaling? In Proceedings of the IEEE/ACM International Conference on Computer Aided Design, San Jose, CA, USA, 4–8 November 2001; pp. 560–563.
4. Rossi, D.; Pullini, A.; Loi, I.; Gautschi, M.; Gürkaynak, F.K.; Teman, A.; Constantin, J.; Burg, A.; Miro-Panades, I.; Beignè, E.; et al. Energy-efficient Near-Threshold Parallel Computing : The PULPv2 Cluster. *IEEE Micro* **2017**, *37*, 20–31. [CrossRef]
5. Di Mauro, A.; Rossi, D.; Pullini, A.; Flatresse, P.; Benini, L. Independent Body-Biasing of P-N Transistors in an 28nm UTBB FD-SOI ULP Near-Threshold Multi-Core Cluster. In Proceedings of the 2018 IEEE SOI-3D-Subthreshold Microelectronics Technology Unified Conference (S3S), Burlingame, CA, USA, 15–18 October 2018; pp. 1–3.

6.   Di Mauro, A.; Rossi, D.; Pullini, A.; Flatresse, P.; Benini, L. Temperature and process-aware performance monitoring and compensation for an ULP multi-core cluster in 28 nm UTBB FD-SOI technology. In Proceedings of the 2017 27th International Symposium on Power and Timing Modeling, Optimization and Simulation (PATMOS), Thessaloniki, Greece, 25–27 September 2017; pp. 1–8.

7.   Choi, K.; Soma, R.; Pedram, M. Fine-grained dynamic voltage and frequency scaling for precise energy and performance tradeoff based on the ratio of off-chip access to on-chip computation times. *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **2005**, *24*, 18–28.

8.   Seki, T.; Akui, S.; Seno, K.; Nakai, M.; Meguro, T.; Kondo, T.; Hashiguchi, A.; Kawahara, H.; Kumano, K.; Shimura, M. Dynamic voltage and frequency management for a low-power embedded microprocessor. *IEICE Trans. Electron.* **2005**, *88*, 520–527. [CrossRef]

9.   Krishnaswamy, V.; Brooks, J.; Konstadinidis, G.; McAllister, C.; Pham, H.; Turullols, S.; Shin, J.L.; YangGong, Y.; Zhang, H. Fine-Grained Adaptive Power Management of the SPARC M7 processor. In Proceedings of the 2015 IEEE International Solid-State Circuits Conference (ISSCC), San Francisco, CA, USA, 22–26 February 2015; pp. 74–76.

10. Cheng, W.H.; Baas, B.M. Dynamic voltage and frequency scaling circuits with two supply voltages. In Proceedings of the IEEE International Symposium on Circuits and Systems, Seattle, WA, USA, 18–21 May 2008; pp. 1236–1239.

11. Scordino, C.; Lipari, G. A resource reservation algorithm for power-aware scheduling of periodic and aperiodic real-time tasks. *IEEE Trans. Comput.* **2006**, *55*, 1509–1522. [CrossRef]

12. Spiliopoulos, V.; Kaxiras, S.; Keramidas, G. Green governors: A framework for continuously adaptive dvfs. In Proceedings of the 2011 International Green Computing Conference and Workshops, Orlando, FL, USA, 25–28 July 2011; pp. 1–8.

13. Juan, D.C.; Garg, S.; Park, J.; Marculescu, D. Learning the optimal operating point for many-core systems with extended range voltage/frequency scaling. In Proceedings of the Ninth IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS '13), Montreal, QC, Canada, 29 September–4 October 2013; IEEE Press: Piscataway, NJ, USA, 2013; pp. 8:1–8:10.

14. Pallipadi, V.; Starikovskiy, A. The Ondemand Governor. *Linux Symp.* **2006**, *2*, 223–238.

15. Rossi, D.; Pullini, A.; Loi, I.; Gautschi, M.; Gürkaynak, F.K.; Bartolini, A.; Flatresse, P.; Benini, L. A 60 GOPS/W, -1.8 v to 0.9 v body bias ULP cluster in 28 nm UTBB FD-SOI technology. *Solid-State Electron.* **2016**, *117*, 170–184. [CrossRef]

16. King, C.I. Stress-ng. Available online: https://github.com/ColinIanKing/stress-ng (accessed on 10 August 2018).