ELSEVIER

Contents lists available at ScienceDirect

Microelectronics Journal



journal homepage: www.elsevier.com/locate/mejo

Low-energy GALS NoC with FIFO—Monitoring dynamic voltage scaling

Abbas Rahimi^a, Mostafa E. Salehi^{b,*}, Siamak Mohammadi^c, Sied Mehdi Fakhraie^c

^a CSE Department, University of California, San Diego, La Jolla, CA 92093-0404, USA

^b Islamic Azad University, Qazvin Branch, Qazvin 34185-1416, IR Iran

^c Dependable Systems Design Laboratory, School of ECE, University of Tehran, Tehran 14395-515, Iran

ARTICLE INFO

ABSTRACT

Article history: Received 19 September 2010 Received in revised form 31 January 2011 Accepted 29 March 2011 Available online 21 April 2011 Keywords:

Low energy GALS NoC Dynamic voltage scaling

1. Introduction

Technology scaling and the increasing device integration levels make power dissipation and on-chip communication as two major factors in the high-performance multiprocessor systemson-chip (SoC). On-chip communication is becoming increasingly important when SoCs grow in complexity and size [1]. Furthermore, power dissipation has emerged as the main design constraint in todays complex SoCs, limiting performance, battery life, and reliability. Networks-on-chip (NoCs) [2] constitute a new design paradigm for scalable, high-throughput on-chip communication in SoCs with billions of transistors, and offer a perfect platform for power management. SPIN, a micro-network, attempts to solve the bandwidth bottleneck in SoCs interconnecting a large number of IP cores via NoCs, [3,4]. A large number of researches on the synchronous NoC architectures have been performed such as AETHEREAL [5], XPIPES [6], and NOSTRUM [7].

Dynamic voltage and frequency scaling (DVFS) techniques, one of the most successful run-time techniques for improving power efficiency, are widely used for optimizing power in synchronous domains [8–10]. Most of these dynamic and static power saving techniques are related to scaling the voltage supply level, which affects power consumption quadratically.

The dynamic power can be efficiently controlled by clock gating at both register transfer (RT) and architecture levels. On the other hand, the asynchronous logic scheme offers both RTL and architectural clock gating inherently without the need of any

© 2011 Elsevier Ltd. All rights reserved. extra software [11]. Asynchronous circuits automatically switch to standby state when they are inactive, and have shown their interesting dynamic power savings, due to their unclocked nature [12]. As an alternative solution for NoC design, the MANGO clockless NoC [13] is one of the first asynchronous NoCs. Asynchronous scalable packet-switching integrated network (ASPIN) [14] is another asynchronous micro-network, which is the asynchronous implementation of (scalable distributed packet-switching integrated network (DSPIN)[15]. These two implementations are systematically compared in [16] and the results show that ASPIN implementation surpasses DSPIN by having 2.5 times smaller average packet latency. The other proposed asynchronous

In this paper we propose two dynamic voltage scaling (DVS) policies for a GALS NoC, which is designed based

on fully asynchronous switch architectures. The first one is a history-based DVS policy, which exploits the link

utilization and adjusts the voltages of different parts of the router among a few voltage levels. The second one

is a FIFO-adaptive DVS, which uses two FIFO threshold levels for decision making. It judiciously adjusts supply

voltage of each switch among only three voltage levels. The introduced architecture is simulated in 90 nm

CMOS technology with accurate Spice simulations. Experimental results show that the FIFO-adaptive DVS not

only lowers the implementation cost, but also in a 86% saturated network achieves 36% energy-delay

product (ED) saving compared to the DVS policy based on link utilization.

NoCs are QoS [17] and ANOC [18]. Globally asynchronous locally synchronous (GALS) [19] paradigm merges the benefits of both synchronous and asynchronous designs; it is being widely investigated as a viable alternative to purely synchronous designs [20,21]. Better power efficiency is achieved in the GALS system, as it offers a natural way to operate each domain at different frequencies and voltages, which facilitates the application of DVFS independently to different parts of circuit [22,23]. To enable GALS systems with multiple clock domains, including DVFS scaling per each synchronous module, the network should be implemented as an asynchronous circuit [24,25].

The rest of the paper is organized as follows. In Section 2 the most relevant recent researches are reviewed. In Section 3 the traffic model is described for a network containing a homogeneous 5×5 set of clusters. The history-based DVS policy based on link utilization is described in Section 4. In Section 5 A FIFO-adaptive DVS policy is presented in detail, including exploration of the threshold levels of FIFO, the three recommended voltage levels, its comparison with link-utilization-based DVS, and also its scalability. Finally, Section 6 concludes the paper.

^{*} Corresponding author. E-mail addresses: abrahimi@cs.ucsd.edu (A. Rahimi),

m.e.salehi@qiau.ac.ir (M.E. Salehi), smohammadi@ece.ut.ac.ir (S. Mohammadi),

m.e.saieni@qiau.ac.ir (M.E. Saieni), smonammadi@ece.ut.ac.ir (S. Monammadi), fakhraie@ut.ac.ir (S.M. Fakhraie).

^{0026-2692/\$ -} see front matter \circledcirc 2011 Elsevier Ltd. All rights reserved. doi:10.1016/j.mejo.2011.03.016

2. Related work

Beigné et al. [26] propose a dynamic voltage and frequency scaling policy for IP units integrated within a GALS NoC, but their policy is applied only for all IPs within an SoC and ignores significant effects of the power dissipation of links and switches. Shang et al. [27] use a history-based dynamic voltage scaling policy for synchronous links, where the frequency and voltage of the links are dynamically adjusted to minimize power consumption. This work targets only the dynamic power optimization of the synchronous interconnection networks, which realizes 4.6 times power savings on average at the expense of 15.2 % increase in the average latency. Lee and Bagherzadeh [28] present a variable frequency link for a power-aware synchronous interconnection network, and apply a dynamic frequency scaling (DFS) policy that adjusts link frequency based on link utilization parameter.

Although recent researches introduce designs of fast on-chip regulators suitable for DVS techniques that permit perform voltage changes in nanoseconds [29,30], they do not fit the needs of current and future NoCs due to their energy inefficiency, costs overhead, and noises [31]. On the other hand, using discrete DVS techniques with a few number of voltage levels is promising [32]. For instance, a 167-processor computational platform supports DVFS using dual global external supply voltages [33]. Processors change their supply voltage by connecting their local power grid to one of the global supply voltages, which provides a simple and efficient approach with a switching delay of only a few clock cycles. Moreover a $V_{\rm DD}$ -hopping technique is proposed in [31], which uses local dynamic voltage scaling (LDVS) and efficiently hops between V_{low} and V_{high} for a GALS system. Furthermore, a dual power supply network in 0.18 µm technology has been proposed for Alpha 21264 processor in [34]. Their experiments show that the dual power supply network structure can switch between $V_{high} = 1.8$ Vand $V_{low} = 1.2$ V in 12 ns with 66 nJ energy dissipation during ramping.

In our previous work [35] the energy/throughput trade-off was analyzed on a GALS NoC which is based on a fully asynchronous NoC, ASPIN [14], using sync-to-async and async-to-sync interfaces [36] to connect synchronous IP cores to the asynchronous network. Our experimental results show that although DFS techniques can improve power consumption in synchronous circuits, interval scaling and consequently throughput scaling in not recommended for energy saving in the fully asynchronous NoCs and the best energy-delay product (ED) [37] saving is achieved in high throughput regions. On the other hand a DVS technique is able to save energy up to 40% at the expense of 13% throughput degradation, while a throughput scaling technique achieves only 0.6% energy saving for the same amount of throughput degradation. Thereby, these results as the first study in this region limit the range of throughput scaling and also limit voltage scaling between 1.0 and 0.75 V as the optimum ranges for a DVS scheme.

In this paper we propose two dynamic voltage scaling policies for GALS NoC architecture based on the previous optimum voltage scaling ranges. First, based on the link utilization parameter [28], a history-based [27] DVS policy is presented, which uses A few number of voltage levels. Second the FIFO-adaptive DVS policy is presented ,which uses only three voltage levels and achieves considerable amount of energy saving at the expense of negligible throughput degradation.

3. Traffic model

An important problem in DVS algorithms is how to predict upcoming workload with reasonable accuracy. An automatic traffic phase detection has been integrated in a DSPIN [15] prototyping environment, which is able to precisely emulate the behavior of the processors [38]. An interesting analogy to characterize the macroscopic behavior of NoCs has been proposed in [39], which predicts that the buffer occupancy follows a power law distribution: each NoC buffer and NoC packet injection rate are characterized by particular energy level and the temperature of a thermodynamic physical system, respectively. Moreover, a statistical physics inspired framework that can capture fractality and non-stationarity has been proposed in [40] to overcome the limitation of queuing theory and Markov chain approaches. Bogdan et al. in [41] propose a new statistical physics-inspired model for non-stationary analysis of NoC traffic. They provide evidence that the NoC traffic needs to be characterized using a multifractal approach.

To evaluate the GALS NoC performance, power, and saturation thresholds-the most important parameters [14]-we have focused on a network containing a homogeneous 5×5 set of clusters. Details of the asynchronous routers and GALS NoC architectures are provided in [35]. Synchronous IP cores transmit/receive data to/from the asynchronous router through sync-to-async/async-to-sync interfaces. The IPs that are connected to the local input port are used for generating traffic. Each IP consists of two parts, traffic generator (TG) and network analyzer (NA). The TG is connected to router's local input port and is used to model the uniform type of traffic [42], and inject packets to the network. Each TG generates a uniform traffic by producing 50 packets, each of them consisting of 8 Flits, in intervals relevant to the desired network load. To prevent TGs from sending the traffic out at the same time and to distribute over time, each TG starts its task after a random value of the interval. The simulation runs for different network loads from 40 GFlits s for each router up to the point the network saturates. The NA is connected to router's local output ports, consumes the generated traffic, and checks the delivery of packets. If too many IPs are generating traffic simultaneously, the network would be saturated. The saturation occurs when the traffic generated by each IP reaches a saturation threshold-that is, when the average packet latency rises exponentially to an infinite value. To account for network contention and to get a meaningful latency measurement, we have time-stamped the packets and posted them in FIFO buffers located in each TG.

We have measured the average packet latency as the time between the departure time in the source node and the arrival time in the destination node. The curve in Fig. 1 depicts average packet latencies, at voltage 1.0 V, versus the generated traffic by an IP. The network saturates for loads higher than 176 GFlits/s. In other words, if the IP flit injection rate exceeds this rate the flits will not be delivered. Similarly, at 0.75 V, the network saturates in loads higher than 144 GFlits/s. According to our results, in 1.0 V, the network is 100% saturated when the injection rate reaches 176 GFlits/s consequently the injection rates of 144, 152, and 160 GFlits/s saturate 82%, 86%, and 90% of the network, respectively, and are used for our simulation results.



Fig. 1. Packet latency versus different loads of the network at 1.0 V.

4. DVS based on link utilization parameter

The NoC switch and its corresponding 32-bit links are modeled in gate-level using VERILOG hardware description language. Fundamental asynchronous components such as C-gate, latch, AND, OR, NOT, MUTEX, etc. are simulated with a 90 nm Spice model using Spice simulations. To evaluate the energy dissipation, latency, and throughput of the network, physical parameters such as average power and propagation delay of gates are extracted from the Spice simulation results and back annotated to the gate-level HDL modules. Spice simulations are performed in a wide range of voltage values and energy dissipation and latency are measured at different operating voltages by gate-level simulation.

The power/energy saving and latency degradation for different voltage values are analyzed in [35]. The results indicate that when the voltage is scaled from 1.0 to 0.75 V, energy savings are higher than latency degradations; therefore, we introduce this region as the low energy region. In other words, for voltages below 0.75 V, little energy saving is gained versus large latency degradation.

4.1. Link utilization indicator

As results of the research in [35] show, the DVS algorithm can dynamically scale the operating voltage of the asynchronous switch (between 0.75 and 1.0 V) and improve the energy saving with negligible performance degradation. The DVS algorithms need to predict upcoming workload with reasonable accuracy. This requires knowing how many packets will traverse a link at any given time. To estimate the upcoming workload we use link utilization, which is an indicator of traffic through a link in each unit time. Lower link utilization indicates more idle states in a link caused by light workloads in the incoming port. Conversely, higher link utilization implies that higher voltages are required to pass the incoming flits to the destination port.

The link utilization that is proposed in [28] is measured by sampling a link at a given time during a pre-defined period (T). The direct link utilization is defined as.

$$U(T) = \sum_{t=0}^{T} u(t) \tag{1}$$

$$u(t) = \begin{cases} 1 & \text{if there is link flit in } [t-1,t] \\ 0 & \text{else} \end{cases}$$

The direct estimator, u(t), only indicates whether a link is occupied or not. It does not consider the number of flits traversing through a link during the given time. To reduce the complexity and hardware overhead of the link controller, direct link utilization can be realized with a counter that counts the number of input port requests during the pre-defined period (*T*). Therefore, U(T) accumulates the total number of flits traversed via link over a pre-defined period (T).

Network workload exhibits transient fluctuations and longterm transitions. In order to filter out transient fluctuations from link utilization and to predict future communication workload, a distributed history-based DVS policy was proposed in [27]. History-based link estimator uses exponential weighted average utilization to combine current U(T) and past $\Psi(n-1)$ link utilization history for smoothing and predicting future link utilization $\Psi(n)$ as follows:

$$\Psi(n) = \frac{\text{weight} \times U(T) + \Psi(n-1)}{\text{weight} + 1}$$
(2)

Weight is the contribution factor of the current link utilization level to the history-based link estimator. The hardware overhead is an important factor for the design of the estimator. Setting the weight equal to 3 the history-based estimator is realized with an adder and two shift registers, which reduces the additional hardware overhead [43]. The history-based link estimator is implemented on the input ports of each asynchronous switch.

4.2. Proposed voltage scaling regions

The limitations in implementing on-chip inductors can reduce the efficiency, accuracy, and the number of voltage levels generated by regulators [32]. Therefore, the voltage scaling algorithms must be efficient, even in the presence of a few voltage levels. The proposed techniques in the literature need many fine-grained voltage levels to produce energy-efficient results and their quality degrades significantly as the number of levels decreases [32]; on the other hand the additional gain of using an infinite number of voltage level is low [44]. We have presented a DVS algorithm based on link utilization parameter that is compatible with our asynchronous NoC and scales the operating voltage among a few voltage levels using the history-based link estimator. The system has multiple voltage levels represented by V_i . The link controller selects the operating voltage among supported voltages using the link utilization level. Since the link utilization $(\Psi(n))$ is used for DVS decisions, the effect of different voltage levels on $\Psi(n)$ for different loads has been shown in Fig. 2. As shown, we can divide the voltage range to regions in which the variations of the voltage does not highly affect the $\Psi(n)$ values. We select the border voltages of the adjacent regions as the recommended voltage levels. Based on this observation, our links support four different voltages level: 0.75, 0.8. 0.9. and 1.0 V.

The technique presented in [27] uses a single threshold value for a two-level voltage level in its synchronous NoC. On the other hand, we have observed that in our asynchronous NoC, using two single threshold values is not sufficient to track U(T). In the two-level voltage level scheme, when the voltage reaches a low value, our



Fig. 2. Effect of voltage scaling on changing $\Psi(n)$ for the three traffic models.



Fig. 3. Normalized U(T), $\Psi(n)$, and the voltage trace.

asynchronous elements become slower and chances to increase $\Psi(n)$ above the threshold value diminish. Therefore, multiple threshold values including at least three levels are required for an asynchronous NoC. Pre-defined multiple threshold values should be set according to each voltage level. Based on the observed regions in the link utilization, we propose four voltage levels. Finally the problem of power minimization in our DVS algorithm is to select a proper voltage value among different voltage levels for minimizing power consumption, while satisfying timing constraints.

4.3. Proposed DVS policy based on link utilization

Considering the link utilization of each input port, our DVS algorithm dynamically adapts voltages of different parts of the router to achieve power and energy savings with minimal impact on performance. Since the output ports must be fast enough to send out the receiving flits from input ports, all output ports of each switch have the same voltage level, which is equal to the value of maximum voltage among all input ports of the switch. The proposed DVS based on link utilization is described according to the following algorithm:

```
for(each 50ns)begin
```

```
for(i \in \{North, South, West, East, IP\}) begin

if(\Psi i(n) < Threshold1)

Vi = 0.75

else if (\Psi i(n) > = Threshold1 & \Psi i(n) < Threshold2)

Vi = 0.8

else if (\Psi i(n) > = Threshold2 & \Psi i(n) < Threshold3)

Vi = 0.9

else if (\Psi i(n) > = Threshold3)

Vi = 1

end

V_{out} = MAX(V_{North}, V_{South}, V_{West}, V_{East}, V_{IP})

end
```

Based on the levels of $\Psi(n)$, a proper voltage level is selected for each input port over each control period (*T*). The control period should be greater than the average packet latency of the network, because the DVS algorithm should observe at least one packet transmission to be able to select a fair V_{DD} and prevent oscillations of V_{DD} . On the other hand, very large control period slows down the workload tracking ability. As shown in Fig. 1 the average packet latency is 63 ns for a 22% saturated network (40 GFlits/s) at voltage 1.0 V, so by taking into account the voltage scaling, 50 ns can be a suitable control period for observing at least one packet transmission even for light workloads. This control period also provides enough time for changing the voltage of regulator [29,30] or hop between voltage levels [34].

Fig. 3 shows that the link utilization tracks well actual workload and therefore history-based DVS successfully adjusts link's voltage to track actual link utilization over time. The workload of network has been changed randomly in order to see the ability of DVS tracking: normalized U(T), $\Psi(n)$, and V_{DD} values of the north input port of the (3,3)th switch, in the middle of network, when the control period is 50 ns, are shown in Fig. 3. Link utilization estimator predicts future workload based on the history of workloads and the DVS policy dynamically adjusts voltage of the input and output ports according to the link utilization level. For instance, under a heavy traffic load, link utilization level as well as latency increases and applying DVS, the link controller increases operating voltage from Vi to Vj for reducing latency at the expense of more power consumption. When link utilization level goes low because of the reduced workload, link controller decreases voltage level and hence reduces power consumption. Fig. 3 shows that history-based DVS successfully adjusts voltage to track actual link utilization over time.

5. FIFO-adaptive DVS

In our second DVS policy the FIFO level is used to predict the upcoming workload instead of history-based and link utilization parameter. FIFO level is a good metric for knowing how many packets will traverse a switch and consequently set the voltage to the optimum value. To estimate the upcoming workload, we use a level of north, south, east, west, and local FIFOs, which is an indicator of traffic through a switch. Low FIFO occupancy level indicates low traffic intensity in a switch caused by light workloads in the incoming ports. Conversely, high FIFO occupancy level implies that higher voltages are required to pass the incoming flits to the destination ports.

To predict upcoming workloads in the previous DVS policy the link utilization is used, which is measured by sampling a link at a given time during a pre-defined period (*T*). Since this metric is evaluated based on fixed interval periods (50 ns) it requires a clock for synchronization, which is not perfectly matched with our fully asynchronous NoC. Furthermore, the link utilization requires a counter and other logics to count the number of input port requests. In addition to the link utilization component, the history-based method needs additional hardware to compute $\Psi(n)$. Therefore, in FIFO-adaptive DVS, the traffic intensity has been monitored by FIFO occupancy level to omit costs of hardware overhead and the clock signal distribution from our asynchronous circuits.

The FIFO occupancy level of each port is an indicator of the traffic on that port— the FIFO depth is 8. Therefore, to filter out transient fluctuations from the input ports, we use the sum of FIFO occupancy levels of all input ports as the traffic indicator of each switch. To reduce the overhead of DC–DC convertors and on-chip inductors the FIFO-adaptive DVS policy scales the operating voltage among the recommended voltage levels for all parts of the switch, including five input ports and five output ports. This leads to better decisions and also reduces the DC–DC convertors and other hardware components overhead, and hence facilitates its implementation.

5.1. Recommended threshold levels of the FIFO

The level of the proposed FIFO is monitored during a simulation with the load of 152 GFlits/s (i.e network saturated at 86%) versus different operating voltages and the results are summarized in Fig. 4. As the results show, lower operating voltages lead to higher FIFO levels and hence increase the probability of the network saturation. For example when V_{DD} is equal to 0.75 V during 30% of the simulation time the FIFO contains 10 Flits, while during 8% of the simulation time it contains the same number of Flits at V_{DD} =1.0 V. This figure also shows an upper bound for the threshold levels of the FIFO, because there is a very low probability that the number of Flits in the FIFO exceeds 24. So this range can be used for selecting appropriate voltage by the DVS policy.

To have optimum energy dissipation the FIFO-adaptive DVS algorithm should dynamically scale the operating voltage of the asynchronous switch between 0.75 and 1.0 V [35], and improve the energy saving with negligible performance degradation. To reduce the number of voltage levels generated by regulators, the switch operating voltage must be selected among the three recommended voltage levels called high voltage (V_h), medium voltage (V_m), and low voltage (V_l); due to the needs of asynchronous circuits described in Section 4.2. Since we have proposed the FIFO occupancy level as the traffic intensity indicator, we need two FIFO levels called low threshold (Th_l) and high threshold (Th_h) to decide when to switch between the voltage levels. The decision of the FIFO-adaptive DVS is based on three simple assumptions:

If (FIFO_level < Th_l) set V_{switch} to V_{l} If (Th_l ≤ FIFO_level < Th_h) set V_{switch} to V_{m} If (Th_h ≤ FIFO_level) set V_{switch} to V_{h}



Fig. 4. Observed FIFO level during simulation versus different voltages.

We have to find two suitable threshold levels for FIFO among the available range to have the best energy saving with the least performance degradation. In [35] we have shown that throughput degradation does not improve the energy saving in asynchronous circuits. Therefore, we try to achieve the highest throughput with the least required voltage. Higher throughput equals lower flit latency in each switch or in other words, low FIFO occupancy level. Therefore, we improve the throughput by minimizing the FIFO occupancy level and expect to have the best energy saving. The results will validate our assumption.

We have equaled $V_{\rm h}$ by the highest supported voltage (i.e 1.0 V), and $V_{\rm l}$ by the lowest supported voltage (i.e 0.75 V). For the sake of finding the threshold values, 0.85 V is selected for $V_{\rm m}$. Fig. 6 shows the FIFO occupancy level during simulation for different sets of threshold values.

These sets of threshold values are selected among the suitable range in Fig. 4. As results show, when (Th_l, Th_h) is equal to (14, 18) we have the lowest FIFO occupancy level and hence the highest throughput. Therefore, we propose 14 and 18 as the optimum values for Th_l and Th_h, respectively. Fig. 5 also shows the threshold values (14, 18) are highly effective in minimizing the FIFO occupancy level compared to (14, 24) and (10, 18).

To validate the claim that higher throughput yields lower energy we have calculated dynamic and total energy, and ED values versus different values for Th_l , and Th_h in Fig. 7 (a) and (b), respectively. As shown in the figures, (14, 18) leads to the best results for all of these parameters.

5.2. Three recommended voltage levels

We have used (14, 18) as the optimum FIFO threshold levels in the rest of the paper. These threshold values are used to set the operating voltage to the three recommended values (i.e V_h , V_m , V_l). V_h and V_l are set to 1.0 and 0.75 V respectively, based on our observations in [35]. With 0.75 V the ED is minimized as much as possible and 1.0 v leads to the lowest packet latency, and hence the highest throughput when required. The next step is to find the



Fig. 6. FIFO occupancy level during simulation versus different values for $(\mathrm{Th}_{l},\mathrm{Th}_{h}).$



Fig. 5. Threshold values (14,18) provide a lower occupancy FIFO relative to the threshold values.



Fig. 7. (a) Dynamic and total energy and (b) ED values versus different configurations of (Th_l, Th_h).



Fig. 8. Effects of voltage levels on (a) Total energyand (b) average packet latency.





suitable voltage value for $V_{\rm m}$. The optimum $V_{\rm m}$ would be the value that leads to the lowest energy dissipation with the least throughput degradation. To find the optimum value for $V_{\rm m}$, we have observed the effects of different voltage values for this parameter on total energy and average packet latency.

As shown the minimum total energy dissipation and maximum packet latency are obtained with $V_{\rm l}$, and the maximum total energy dissipation and minimum packet latency are provided with $V_{\rm h}$. Thereby, we have tried to find a suitable $V_{\rm m}$ somewhere between these two extremes, where energy delay product can be optimum, and therefore the middle of the curves should be a convergence point. As shown in Fig. 8(a), this point turns out to be around 0.86 V in favor of total energy dissipation curve while the average latency curve in Fig. 8(b) proposes 0.81 V for $V_{\rm m}$. Therefore, we select different $V_{\rm m}$ in this range and observe their effect in energy dissipation and packet latency.

Fig. 9 shows the effects of different $V_{\rm m}$ values on dynamic/ leakage/total energy, ED, and power consumption of the NoC. As shown, $V_{\rm m}$ =0.82 V leads to the lowest dynamic energy, leakage energy, total energy, power, and ED (normalized values). Thus, we have selected voltage levels V_1 =0.75 V, V_m =0.82 V and V_h =1.0 V.

5.3. Comparison of FIFO-adaptive and link-utilization-based DVS

So far we have specified two DVS algorithms: the historybased DVS policy, which uses the link utilization as the traffic intensity indicator, and the FIFO-adaptive DVS algorithm. To see the effectiveness of the proposed algorithms, we take into account the energy overhead of these techniques, which is less than 10% of total energy of a switch in FIFO-adaptive DVS algorithm. Fig. 10 shows the ED results of the proposed algorithms and a system without any DVS (voltage is fixed at 1.0 V) for different workloads. According to this figure the FIFO-adaptive DVS not only lowers implementation cost by removing the clock for synchronization of the fixed interval periods and utilizing the available one-hot address coding in FIFO [45] instead of using extra centers and hardware, but also surpasses the DVS based on link



Fig. 10. ED values in FIFO-adaptive DVS, DVS based on link utilization, and non-DVS for different loads.



Fig. 11. Normalized ED and leakage power for different mesh configurations.

utilization in ED saving for different loads. It achieves more than 36% and 43% ED savings compared to the DVS based on link utilization and non-DVS techniques in a 86% saturated networks, respectively.

We also examine oracle-driven voltage scaling to understand the limits of our FIFO-adaptive DVS. The oracle voltage is the best voltage, that an ideal DVS algorithm can predict; in other words, it is the minimum voltage that is required to pass U(T) flits in each control period. Energy dissipation has a direct relationship to V_{DD}^2 and therefore we can evaluate the effectiveness of the FIFOadaptive DVS policy by calculating the quadratic ratio of the predicted voltage to the ideal voltage proposed by oracle. In the worse case situation, the FIFO-adaptive DVS dissipates 1.4 times more energy than the ideal oracle.

5.4. Scalability of FIFO-adaptive DVS

Thanks to one-hot address coding, which has been implemented in low-power FIFO GALS [45], it is easy to find out the portion of the FIFO that is occupied, and thus the FIFO-adaptive DVS can decide the suitable voltage based on threshold levels using simple adders and comparators. This implementation not only lowers area overhead but also does not affect the critical path. To observe the scalability of FIFO-adaptive DVS, we have applied it to different mesh sizes ranging from 4×4 up to 8×8 . Fig. 11 shows the ED and leakage power results normalized to a non-DVS system at V_{DD} =1.0 V in a 86% saturated networks. Starting from 4×4 configuration the normalized ED is 0.56, and when the number of IPs is increased by a factor of 4 the normalized ED value increases less than 10% mainly because of leakage power issue, confirming scalability across a significant range of configurations.

6. Conclusion

In this paper we have exploited a fully asynchronous NoC architecture for GALS-based MPSoC architectures and proposed two DVS schemes for low-power and low-energy applications. To evaluate energy, power, and performance of the GALS NoC, we have introduced a traffic model and found the related saturation thresholds for different voltage levels. The link utilization indicator and the recommended voltage scaling regions have been then introduced and a history-based DVS algorithm based on link utilization has been proposed accordingly. We have also augmented the DVS algorithm with FIFO and explored the effective threshold levels for FIFO. Then a FIFO-adaptive DVS algorithm is proposed, which uses the FIFO level as the traffic intensity indicator and scales the operating voltage to three recommended optimum voltage levels. The FIFO-adaptive DVS not only has lower cost of implementation but also achieves better ED saving compared to the link-utilizationbased DVS in saturated networks.

References

- W.J. Dally, B. Towles, Route packets, not wires, in: Proceedings of the design automation conference on-chip interconnection networks, 2001, pp.684–689.
- [2] L. Benini, G. De Micheli, Networks on chips: a new SoC paradigm, IEEE Computer magzine. 35 (1) (2002) 70–78.
- [3] P. Guerrier, A. Greiner A generic architecture for on chip packet-switched interconnections, in: Proceedings of DATE conference, 2000, pp. 250–256.
- [4] A. Adriahantenaina, A. Greiner, Micro-network for SoC: implementation of a 32-port SPIN network in: Proceeding of DATE conference, 2003, pp. 11128.
- [5] J. Dielissen, A. Rădulescu, K. Goossens, E. Rijpkema, Concepts and implementation of the philips network-on-chip, in:proceedings of IP-SOC 2003 conference.
- [6] M. Dall'Osso, G. Biccari, L. Giovannini, D. Bertozzi, L. Benini, Xpipes: a latency insensitive parameterized network-on-chip architecture for multi-processor SoCs, in: Proceedings of the 21st ICCD conference, 2003, pp. 536–539.
- [7] M. Millberg, E. Nilsson, R. Thid, A. Jantsch, Guaranteed bandwidth using looped containers in temporally disjoint networks within the nostrum network on chip, in: Proceedings of DATE conference 2004, pp. 890–895.
- [8] C. Xian, Y.H. Lu, Z. Li, Dynamic voltage scaling for multitasking real-time systems with uncertain execution time, IEEE Transactions on Computeraided Design of Integrated Circuits and Systems 27 (8) (2008) 1467–1488.
- [9] U.Y. Ogras, R. Marculescu, D. Marculescu, E.G. Jung, Design and management of voltage-frequency island partitioned networks-on-chip, IEEE Transactions. on Very Large Scale Integration Systems 17 (3) (2009) 330–341.
- [10] M. Elgebaly, M. Sachdev, Variation-aware adaptvie voltage scaling system, IEEE Transactions on Very Large Scale Integration (VLSI) Systems 15 (2007) 560-571.
- [11] M. Es Salhiene, L. Fesquet, M. Renaudin, Dynamic voltage scheduling for real time asynchronous systems, in: Proceedings of PATMOS'2002, 2002, pp 155–171.
- [12] H. Van Gageldonk, K. Van Berkel, A. Peeters, D. Baumann, D. Gloor, G. Stegmann, An asynchronous low-power 80C51 microcontroller, in: Proceedings of ASYNC'98, 1998, pp. 96–107.
- [13] T. Bjerregaard, J. Sparsø, A router architecture for connection-oriented service guarantees in the MANGO clockless Network-on-Chip, in: Proceedings of DATE conference 2005, pp. 1226–1231.
- [14] A. Sheibanyrad, A. Greiner, I. Miro-Panades, Multisynchronous and fully asynchronous NoCs for GALS architectures, IEEE Design and Test 25 (6) (2008) 572–580.
- [15] I. Miro Panades, A. Greiner, A. Sheibanyrad, A Low Cost Networkon-chip with guaranteed service well suited to the GALS approach, in: Proceedings of Nano-Net 2006.
- [16] A. Sheibanyrad, I. Miro-Panades, and A. Greiner, Systematic comparison between the asynchronous and the multi-synchronous implementations of a network on chip architecture, in: Proceedings of DATE conference 2007, pp. 1090–1095.
- [17] T. Felicijan, S. B. Furber, An asynchronous on-chip network router with qualityof-service (QoS) support, in: Proceedings of SOCC 2004, pp. 274–277.
- [18] E. Beigne, F. Clermidy, P. Vivet, A. Clouard, M. Renaudin, An asynchronous NoC architecture providing low latency service and its multi-level design framework, in: Proceedings of the 11th ASYNC, 2005, pp. 54–63.
- [19] D. M. Chapiro, Globally asynchronous locally synchronous systems, PhD Thesis, Stanford University, 1984.
- [20] A. Iyer and D. Marculescu, Power and performance evaluation of globally asynchronous locally synchronous processors, in: Proceedings of ISCA, 2002, pp. 652–661.
- [21] G. P. Semeraro et al., Hiding synchronization delays in GALS processor microarchitecture, in: Proceedings of ASYNC, 2004, pp. 159–169.
- [22] G. Magklis, P. Chaparro, J. Gonzalez, A. Gonzalez, Independent front-end and back-end dynamic voltage scaling for a GALS microarchitecture, in: Proceedings of ISLPED '06, October 2006, pp. 49–54.

- [23] G. Semeraro et al., Energy-efficient processor design using multiple clock domains with dynamic voltage and frequency scaling, in: Proceedings of ISHPC, 2002, pp. 29–40.
- [24] A. Lines, Nexus: an asynchronous crossbar interconnect for synchronous system-on-chip designs, in: Proceedings of the 11th Symposium on High Performance Interconnects, 2003, pp. 2–9.
- [25] R. Dobkin, V. Vishnyakov, E. Friedman, R. Ginosar, An asynchronous router for multiple service levels networks on chip, in: Proceedings of ASYNC, 2005, pp. 44–53.
- [26] E. Beigné, F. Clermidy, S. Miermont, P. Vivet, Dynamic voltage and frequency scaling architecture for units integration within a GALS NoC, in: Proceedings of the Second ACM/IEEE International Symposium on Networks-on-Chip, 2008, pp. 129–138.
- [27] L. Shang, L.-S. Peh, N.K. Jha, Dynamic voltage scaling with links for power optimization of interconnection networks, in: Proceedings of the ninth International Symposium on High-Performance Computer Architecture, 2003, pp. 91–102.
- [28] S.E. Lee, N. Bagherzadeh, A variable frequency link for a power-aware network-on-chip (NoC), Integration, The VLSI Journal 42 (4) (2009) 479–485.
- [29] W. Kim, M. Gupta, G.-Y. Wei, and D. Brooks, System level analysis of fast, percore DVFS using on-chip switching regulators, in: Proceedings of the International Symposium on High-Performance Computer Architecture, February 2008, pp. 123–134.
- [30] P. Hazucha, T. Karnik, B.A. Bloechel, C. Parsons, D. Finan, S. Borkar, Areaefficient linear regulator with ultra-fast load regulation, IEEE Journal of Solid-State Circuits 40 (4) (2005) 933–940.
- [31] S. Miermont, P. Vivet, and M. Renaudin, A power supply selector for energyand area-efficient local dynamic voltage scaling, in: Proceedings of PAT-MOS'2007, Göteborg, Sweden, 2007, pp.556–565.
- [32] B. Gorjiara, N. Bagherzadeh, P. Chou, An efficient voltage scaling algorithm for complex SoCs with few number of voltage modes, in: Proceedings of ISLPED, 2004, pp. 381–386.
- [33] D.N. Truong, W.H. Cheng, T. Mohsenin, Z. Yu, A.T. Jacobson, G Landge, M.J. Meeuwsen, C Watnik, A.T. Tran, Z. Xiao, E.W Work, J.W. Webb, P.V. Mejia, B.M. Baas, A 167-processor computational platform in 65 nm CMOS, IEEE Journal of Solid-State Circuits 44 (4) (2009) 1130–1144.
- [34] H. Li, C.-Y. Cher, T. Vijaykumar, and K. Roy, VSV: L2-miss-driven variable supply-voltage scaling for low power, in: Proceedings of the 36th Annual IEEE/ACM International Symposium on Microarchitecture, 2003, pp. 19–28.

- [35] A. Rahimi, M.E. Salehi, S. Mohammadi, S.M. Fakhraie, A. Azarpeyvand, Energy/ throughput trade-off in a fully asynchronous NoC for GALS-based MPSoC architectures, in: Proceedings of fifth International Conference on Design & Technology of Integrated Systems in Nanoscale era (DTIS), 2010, pp. 1–6.
- [36] A. Sheibanyrad and A. Greiner, Two efficient synchronous ↔ asynchronous converters well-suited for network on chip in GALS architectures, in: Proceedings of the Power and Timing Modeling, Optimization and Simulation (PATMOS 06), conference on Integrated Circuit and System Design LNCS 4148, Springer Berlin, 2006, pp. 191–202.
- [37] M. Pedram, J.M. Rabaey, Power Aware Design Methodologies, Kluwer: Academic, 2002.
- [38] Antoine Scherrer, Antoine Fraboulet, and Tanguy Risset, Automatic phase detection for stochastic on-chip traffic generation. in: Proceedings of the fourth International conference on Hardware/software Codesign and System Synthesis (CODES+ISSS '06), 2006, pp. 88–93.
- [39] Paul Bogdan and Radu Marculescu, Quantum-like effects in network-on-chip buffers behavior, in: Proceedings of the 44th Annual Design Automation Conference (DAC), 2007, pp. 266–267.
- [40] P. Bogdan and R. Marculescu, Statistical physics approaches for network-onchip traffic characterization, in: Proceedings of the seventh IEEE/ACM International Conference on Hardware/Software Codesign and System Synthesis, 2009, pp. 461–469.
- [41] P. Bogdan, M. Kas, R. Marculescu, O. Mutlu, QuaLe: A quantum-leap inspired model for non-stationary analysis of NoC traffic in chip multi-processors, in: Proceedings of the 2010 Fourth ACM/IEEE International Symposium on Networks-on-Chip, 2010, pp. 241–248.
- [42] S. Koohi, M. Mirza-Aghatabar, S. Hessabi, M. Pedram, High-level modeling approach for analyzing the effects of traffic models on power and throughput in mesh-based NoCs, in: Proceedings of the 21st International Conference on VLSI Design, 2008, pp. 415–420.
- [43] V. Soteriou, N. Eisley, L.-S. Peh, Software-directed power-aware interconnection networks, ACM Transactions on Architecture and Code Optimization (TACO) 4 (5) (2007) 1–40.
- [44] S. Lee, T. Sakurai, Run-time voltage hopping for low-power real-time systems, in: Proceedings of the Design Automation Conference (DAC), 2000, pp. 806–809.
- [45] M. Fattah, A. Manian, A. Rahimi, S. Mohammadi, A high throughput low power FIFO used for GALS NoC buffers, in: Proceedings of the IEEE Annual Symposium on VLSI (ISVLSI), 2010, pp. 333–338.