IJMTST | Volume: 2 | Issue: 06 | June 2016 | ISSN: 2455-3778

Traffic and Power reduction Routing Algorithm for NOC Cores

Kalivarathan[1] | Naveen[2]

[1]PG Scholar, Department of Electronics and Communication Engineering, Gojan School of Business and technology, Chennai, TamilNadu, India.
[2]Assistant Professor, Department of Electronics and Communication Engineering, Gojan School of Business and technology, Chennai, TamilNadu, India.

ABSTRACT

With the progress of VLSI technology, the number of cores on a chip keeps increasing, Now a days we are increasing the processing level of the chip, NOC is a best method to interconnect the core with each other core on the chip, it reducing the overall chip power and Traffic level by sharing the work load with other cores on the chip. And Dynamic Voltage Frequency Scaling (DVFS) is the technique for monitoring the Frequency/Voltage level of each core of the chip and providing sufficient power to the cores, ATPT is a Table that having (low and high) Frequency level table of the Each core. ATPT has very high prediction accuracy system. Depends upon the data speed of the core the voltage/frequency will be given by DVFS. If the core is in ideal state for a while, that core is moved to low power mode. So the power of the each core will be reduced.

KEYWORDS: *Network on Chip, Dynamic Voltage Frequency Scaling, Application Traffic Prediction Table, Last Value Prediction, Pattern Oriented Predictor, Performance Monitoring Unit, Core Traffic*

I. INTRODUCTION

With the progress of VLSI technology, the number of cores on a chip keeps increasing, now days we are increasing the processing level of the chip, NOC is a best method to interconnect the core with each other core, Here reducing the overall chip power and Traffic level by sharing the work load with other cores on the chip. And Dynamic Voltage Frequency Scaling (DVFS) is the technique for monitoring the Frequency/Voltage level of each core on the chip and providing sufficient power to the cores, ATPT is a Table that having (low and high) Frequency level table of the Each core. ATPT has very high prediction accuracy system. Analyses also show that it incurs a low area overhead and is very feasible. Increasing the clock frequency to increase performance is no longer an option due to, amongst others, energy consumption, heat developments and the enormous costs for new technologies. Power and thermal distribution are two critical problems in current chip design. With the progress of VLSI technology, the number of cores inside one chip is still increasing, so from multi core to many core, the Network on Chip (NoC) is the most widely used solution to interconnect the cores. and explicit tile to tile communication is supported. When cores are interconnected by the NOC, additional problems need to be considered, such as power consumption and heat from switches the control of traffic and intercommunication timing between tasks. Apparently, the NoC plays an important role in the many core design; the increasing number of cores requires a communication system different from a conventional bus system, since a bus quickly becomes the bottleneck of the system. One approach is to employ a Network on Chip (NoC). With the ongoing trend to increase the number of cores on chips, the NoC becomes an essential part of the system. There are many NoC topologies such as meshes, trees, multistage interconnection networks (MINs), and many more. NoCs have several advantages such as scalability and modularity.

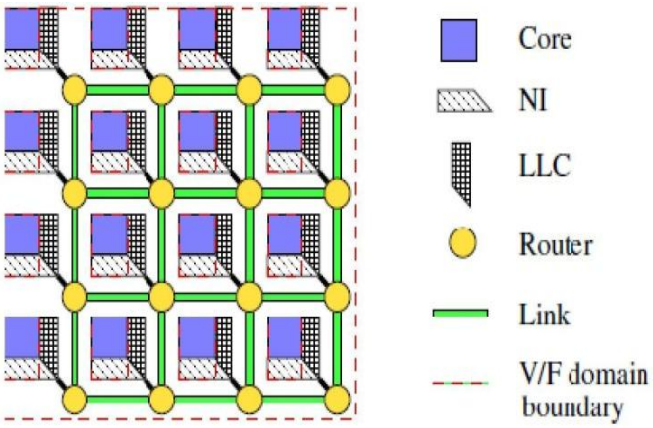

Figure 1: Architecture of NOC

According to this observation, in this paper we propose a novel application driven approach for predicting traffic in NoC and performing DVFS on communication links. We consider message passing many core architectures, in which cores communicate with each other directly through explicit message passing. The basic idea is to capture the communication patterns between parallel tasks, i.e., the endtoend traffic, by using a small table in the network interface (NI) of each core to record the outgoing messages from that core. The novel data structure is called the Application driven Traffic Pattern Table (ATPT). With the support of ATPTs, the amount of data injected into the NOC from each core can be predicted. Once the predictions are made, the utilization of each individual link can also be derived. The voltage/frequency (VF) level of the link can thus be adjusted proactively based on the predicted link utilization.

II. Background and Previous Work

A. Networks on Chip

The solution to scalably and efficiently connect onchip components is a packet switched on chip network. A NoC consists of a high speed router at each node, connected by links to its neighbors. Although the NoC may carry various kinds of traffic (e.g. interrupt requests), its most important purpose is to service cache miss requests. At the heart of this line of research are micro architectural questions such as router/link design or efficient topologies. However, as we discuss in this paper, some of the key problems in on chip networks are in fact networking problems, rather than architectural problems.

B. NoC Characteristics

NoCs feature a number of special characteristics: high performance demands, coupled with hardware implementation constraints, lead to a different trade off space for NoCs compared to most traditional off chip networks. NoCs run at higher utilization, and traffic patterns do not exhibit flow character (such as in the Internet), but are characterized by the self throttling nature of applications on the various cores. Aspects such as chip area/space, power consumption, and implementation complexity (e.g. the expense of arbitration and routing logic) are first-class considerations. These and other characteristics lend on chip networks an interesting and unique flavor, and have important ramifications on the resulting networking solutions.

The presented work is completely Depends upon the traffic prediction, There is no other activity for reducing the traffic, In NOC cores dynamic voltage frequency is the major role that can monitoring the data transmission from one core to another core and also it can increase/ reduce the power depends upon the data transmission rate(Bits/Sec). So the total chip power is reduced. By this system Application Driven Traffic Prediction Table (ATPT) is the table that having Low and High frequency value of each cores. ATPT having two types of table (i) Last Value Prediction (ii) Pattern Oriented Predictor, In that the core working in high frequency that is the core data transmission speed(Bits/Sec) is high, That Time DVFS provides sufficient power to the core which is acting high or low Data transmission rate, DVFS is the best method to monitor the transmission rate of the cores, The total power is reduced by operating the cores as the above method, The overall process is to monitoring the Traffic level and giving sufficient power to the cores, There is no other activity to reduce the traffic level of the system.

C. Limitations

- Predicting the traffic of each cores
- There is no other work to reduce the traffic level in the chip

III. System Module

NOC system is the best method for interconnecting the core with each other cores, The main contribution is power and traffic, In this proposed system we are reducing the overall traffic level and overall power consumption of the chip .Here the End to End data transmission has monitored by Dynamic Voltage Frequency Scaling (DVFS), Depends upon the data transmission rate the power has given to the cores, This is the method to manage the wastage power, so the total

chip power as reduced. If the data rate is high that core considering as a overload system, that time the data's are shared with other core on the same chip, By sharing the data's with other core, we can manage the traffic level. So there is no need to give more power to the core like the existing system.

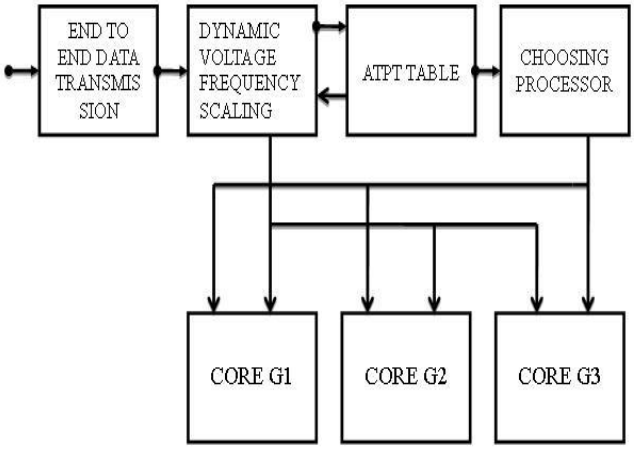

Figure 2: Block Diagram of Proposing Methodology

We are reducing the Traffic and power level of the chip, By using DVFS technique , And there are two types of (high and low) frequency table for predicting the Data rate of each cores, If the data rate is high that time the data has been shared with other core so the traffic level is reduced, To solve the power consumption problem in NoC, previous works have proposed techniques such as dynamic voltage/frequency scaling (DVFS) to adjust the power mode of the switches and links to match the traffic flows. The challenge is to predict the traffic flowing through the switches and links in the next time interval. For example, if a switch or a link can be predicted to be idle for a while, then it can be set to a low power mode by lowering its voltage or frequency. According to this observation, in this paper we propose a novel application driven approach for predicting traffic in NoC and performing DVFS on communication links.

We consider message passing many core architectures, in which cores communicate with each other directly through explicit message passing. The basic idea is to capture the communication patterns between parallel tasks, i.e., the endtoend traffic, by using a small table in the network interface (NI) of each core to record the outgoing messages from that core. The novel data structure is called the Application driven Traffic Pattern Table (ATPT). With the support of ATPTs, the amount of data injected into the NOC from each core can be predicted. Once the predictions are made, the utilization of each individual link can also be derived. The voltage/frequency (VF) level of the link can thus be adjusted proactively based on the predicted link utilization. In comparison to previous studies that make the DVFS decision based on the hardware status, our approach uses the data transmission behavior of the application as the VF scaling reference. The data transmission behavior is a better guide, because it is more predictable and the repetitive behavior exists in the execution phase.

Merits

- By sharing the overload data's with other core, we can manage the traffic level. So the overall chip traffic level is reduced
- Depends upon the data rate the power as given to the cores
- If the core has been idle for a while, then the core is set to be a low power mode

IV. Overview of Network on Chip

Today buses are the dominating technology for systems on chips (SoCs) However, buses have severe limitations that become evident, if the number of components in a system is large, The bus is a communication bottleneck, bandwidth is limited. Since they provide a much larger amount of communication resources and are scalable Network on chip is an emerging paradigm for communications within large VLSI systems implemented on a single silicon chip. The layered-stack approach to the design of the on chip inter-core communications the networkonchip (NOC) methodology. "In a NoC system, modules such as processor cores, memories and specialized IP blocks exchange data using a network as a "public transportation" subsystem for the information traffic. A NoC is constructed from multiple point-topoint data links interconnected by switches, such that messages can be relayed from any source module to any destination module over several links, by making routing decisions at the switches. A NoC is similar to a modern telecommunications network, using digital bit packet switching over multiplexed links.

Our approach is different from all these works since, we use dynamic frequency modulation of routers to manage congestion and improve throughput. Recently, Moscibroda. show that a buffer less routing approach leads to lower power consumption without significantly sacrificing performance in onchip networks.

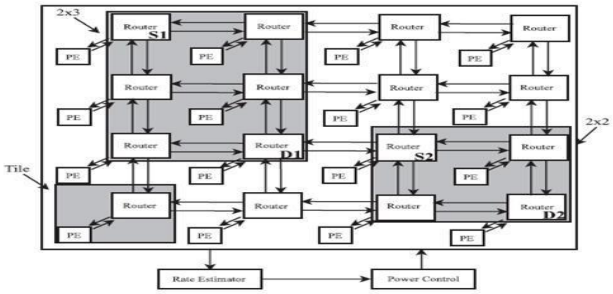

Figure 3: NOC Grid Topology

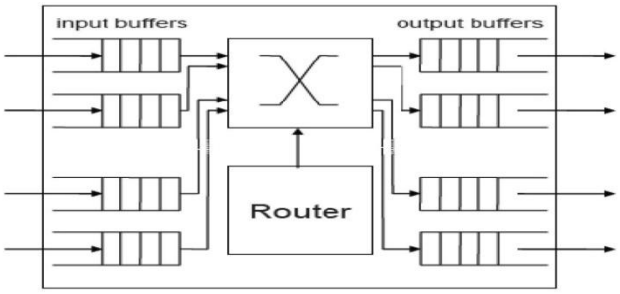

Figure 4: Architecture of Switch

An activity based power management scheme was recently implemented in the routers of the Intel 80core chip For this chip, power reduction was achieved by deactivating ports in the onchip router and putting individual queues in the ports to sleep or clockgating them based on activity inside the router. Ours approach is orthogonal to this idea and can be implemented on top of this design for performance and power management as well.

A. Link Delay and Power

To assess the impact of global wires, we have studied 65 nm NoC links in isolation from the NoC modules. Several factors have to be considered in link design, includingobviously length and desired clock frequency. Short or slowclocked links do not pose problems. However, as either length or target frequency is increased, an undesired rise of power consumption is also observed.

These results show that NoC links implemented using the LPHVT library are substantially more powereffective, but impose much tighter constraints on link feasibility.

B. NOC Components

According to the NoC approach, a core is connected to a switch via a network adapter. The connection among switches is implemented with links and determines the logical network topology. It is worth mentioning that nowadays fabricated chips are incorporating 8x8 cores (TILERA Tile64). The structure of a Mesh topology is shown in Fig .The core is the NoC component which sends and receives messages. The network adapter relieves the core from taking care for the communication process. It is the interface which connects the core to the NoC. The main task of the adapter is to handle the flow control, which refers to the allocation of network buffers and links among the data competing for network resources in order to be delivered at their destination. The switch determines the path over which the packets are delivered. The basic micro architecture of a switch is shown in Figure 4.

1. packet routing Link
2. usually bidirectional
3. Resource Network Interface (RNI, NI)
4. Packet sizes data
5. EndtoEnd flow control
6. Processing Element (PE), Core,
7. Resource could be a memory block

The number of inputs of a switch depends on the topology, considering the topology in the switch has at maximum four inputs from other switches. An additional input is needed to enable the receipt of data from the local core. Incoming data are stored in local buffers. Note, a switch can also be output buffered or have both input and output buffers. According to the routing strategy (Chapter 2.3.2) the router determines the output for incoming packets. Packets are switched to the selected output and are forwarded to the adjacent switch Routing on NoC Routing on NoC is quite similar to routing on any network. A routing algorithm determines how the data is routed from sender to receiver. Routing algorithms are divided into two groups, oblivious and adaptive algorithms. Oblivious algorithms are also divided into two subgroups: deterministic and stochastic algorithms.

The small size of Network on Chip circuits sets special requirements for all operations. The network technology of the Internet is very hard to straightly shrink to the NoC so the technologies should be specially adapted to the NoC. The routing algorithms presented in this report are difficult to be set in the order of superiority. Different applications need different routing algorithms. While some algorithm is suitable to one system, another algorithm works better in some other system. However, it can be generalized that in most of the cases a simple algorithm suits to simple systems while complex algorithms fit to more complex systems.

C. Characteristics of NOC'S

With an understanding of NoC and buffer less NoC design, an important question that remains is:

in what sense do on chip net works differ from other types of networks? These differences provide insight into what makes a NoC interesting from a networking research point of view, and helps to guide the design of our congestion control mechanism. We present key properties of both general and buffer less NoCs.

V. Module Classification

The Proposing Methodology having different classification of modules which are shown below. They are

1. End to End Data transmission
2. Dynamic Voltage Frequency Scaling
3. Application Traffic Prediction Table
4. Dimension Algorithm
5. Multicore Processor

A. End To End Data Transmission

End to End data transmission is the method of communication between one end to another end, Here the end to end data transmission is indicating the data transmission between one core to another core. For example every processor using for different functions, Inside the processor. There are so many cores for different functions, each core contains Register, Memory, ALU, MUX, and DMUX, etc .For this NOC method the core as interconnected with Each other core and that is for easy communication. By taking the core with four terminals that is named as E1, W1, N1 S1.End to end transmission is the core that is connected with other core.

A corresponding example for data communication using EndtoEnd data transmission is shown in Figure. One can see that the communication is much simpler in this case. Rather than checking the packet at each switch along the path, the packet is just forwarded to its final destination and a corresponding response is directly sent back to the source.

B. Dynamic Voltage Frequency Scaling

DVFS gives much more limited energy savings with relatively high performance overhead as compared to running workloads at high speed and then transitioning into low power state Power consumption has become an overriding concern in the design of computer systems today. Dynamic voltage frequency scaling (DVFS) is a power management technique, that dynamically scales the voltage and frequency (vf) settings of the cores so as to provide speed to process the system workload. Scaling down of voltage levels results in a quadratic reduction in cores dynamic power consumption. Many modern cores such as AMD Opteron and Intel Xeon are equipped with DVFS capability

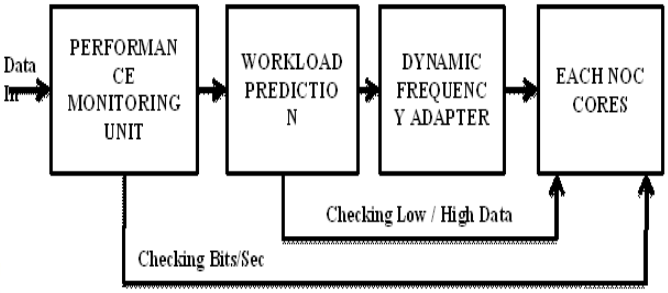

Figure 5: Dynamic Voltage Frequency Scaling Block Diagram

Here dynamic voltage frequency scaling is the method of monitoring and controlling the voltage/frequency value of the cores, the above method that is the end to end data transmission is monitored by DVFS. Several works have proposed DVFS in processors for power management. Recently, work done, it has shown the possibility of onchip voltage regulators. Prior works have proposed DVFS for links to manage power in off-chip networks.

Our proposed schemes are quite different from all related works discussed here in the sense that we propose an adaptive frequency router which adapts to load around its vicinity. We do distributed throttling of routers by frequency tuning in on chip networks to manage power and optimize performance. Moreover, our proposed schemes manage congestion in the network, and thereby, give additional throughput gain. To the best of our knowledge, no prior research has proposed any technique to tune a router's frequency dynamically and adapt to the load conditions.

C. Application Traffic Prediction Table

It's a table that having a High and low frequency range of each core of the system, Depends upon the ATPT table the DVFS system is giving the voltage to cores. There are two type of table. However, the POP uses more memory space than the LVP. The accuracy of the POP is related to how many patterns can be tracked in the 2ndlevel table. Currently, we have not limited the size of the 2nd-level table, and therefore all the patterns can be tracked and used to make predictions. But in a practical implementation, considering the transistor count overhead, the 2ndlevel table may be not large enough to hold all the patterns throughout application execution, so only a subset of the patterns could be recorded. Consequently,

the accuracy will be dramatically affected by the replacement policy. When a longer pattern history is needed, the number of total tracked patterns (the difference between the POP and the ATPTbased predictor) will be broadened as Figure 9 shows. To summarize, the ATPTbased predictor (triangle dotted line) adaptively selects an LVP in low-variance traffic, and a POP for highvariance traffic. Moreover, the experiments show that the ATPT uses fewer entries than the POP and is suitable for memory constrained design.

Pattern Learning and Misprediction Recovery: We give a simple example to demonstrate how the POP (as a part of the ATPTbased prediction) learns the communication pattern and recovers from mispredictionate runtime. shows two different time periods of the benchmark program execution, where the POP encountered the same communication patterns. The first time, the POP recognized the pattern of core1,and it made an inaccurate prediction (1 stands for there is a transmission in that time interval, and 0 stands for no transmission). The POP will change the prediction for that communication pattern in the 2ndlevel table when a misprediction is detected. Afterwards, when the same pattern appears, the POP can make a correct prediction.

- Last Value Predictor
- Pattern Oriented Predictor

The basic idea is to capture the communication patterns between parallel tasks, i.e., the end toend traffic, by using a small table in the network interface (NI) of each core to record the outgoing messages from that core. The novel data structure is called the Applicationdriven Traffic Pattern Table (ATPT).

Application Traffic Prediction Table: The ATPT can be applied in the following scenarios:

- Endtoend flow control for controlling the injection rate.
- Controlling the power mode of switches.
- Modeling the thermal distribution of a chip while taking both cores and switches in consideration simultaneously.
- Optimizing the intercommunications of the application at runtime.
- Three different strategies on DVFS in communication links are proposed that satisfy different optimization goals. Inside the processor each core having some working frequency range.

D. Dimension Algorithm

Purose of Routing: In many applications, there are separate nodes of some sort that wish to communicate with each other using communications channels of some sort. However, not all nodes are usually connected to each other, as connecting all nodes directly to each other involves lots of wires/cables and/or high-powered transceivers. Nodes in network must forward other nodes' transmissions to correct destination. The process of determining where to forward packets and actually doing so is called routing.

Adaptive Routing Algorithm: The issue of inter processor communication has become paramount due to the ever increasing speed of each unit and the amount of Processing Elements (PEs) now included in mainstream systems. Various architectures and routing algorithms have appeared in the last two decades in order to decrease the overhead to both the data rate as well as chip size. This document contains an overview of popular tactics devised for adaptive routing on multiprocessor microchips. A great variety of adaptive routing algorithms have been devised for networking in the more traditional sense. However, adaptive algorithms for routing in computer systems with multiple PEs onchip are more recent. They usually work by introducing flow control techniques that provide the adaptive behavior while precluding the possibility of deadlocks. Each approach may or may not be limited to a specific topology; both options are explored here.

Deadlocks: Deadlocks in fully adaptive routing can be a very difficult problem to solve since there are an infinite amount of possible traffic scenarios. Node buffer size as well as topology will dictate the probability of resource deadlock. All solutions discussed in this document include a form of flow restrictions, which is to say that only certain abstract "turns" along the topology are allowed.

Livelocks: A live lock is a type of starvation that can occur in adaptive routing where misrouting is permitted. Packets that were routed away from their destination may become locked in a loop sequence that persistently reroutes them away from their goal due to local congestion. The concept of fairness must be included in an algorithm in order to preclude this situation. Other solutions involve the same type of restrictions that are used for preventing deadlocks.

E. Multicore Processor

DVFS implementations on multicore processors are more complex than on single core processors and are often simplified by limiting the available voltage and/or frequency domains. Chipwide DVFS forces each core on a package to operate at the same frequency and voltage. This further

constrains the effectiveness of DVFS because workloads running on multiple cores must be analyzed as a whole in order to determine whether or not to scale frequency.

The most recent AMD Opteron used in our tests provides a single voltage domain, but independent frequency domains. Each core can operate at a different frequency, but the voltage must be no lower than that required by the core operating at the highest frequency.

VI. Power Analysis

A. Power Management

NOCs consist of a set of cores connected with the communication network. NOC we will be referring to throughout this paper. The NOC consists of four cores: MPEG audio, video, speech processing, and communication core. The cores communicate with each other via router using a networking protocol. The design of the networking protocol for NOCs is beyond the scope of this paper, but has been addressed in and. In this paper, we enhance each core's ability to control its power states by enabling closed-loop integrated node and network centric power management with DVS. In order to compute the power manager's (PM's) control, we need to develop a system model.We model NOC using Renewal theory as a queuing network with a number of service points representing cores. Management of energy consumption under QoS constraints is formulated as a closed-loop stochastic control problem. Control theory defines three different entities in a closed-loop control system: a system under control, an estimator, and a controller. PM, as shown in, contains the controller and the estimator. The estimator observes the requests coming into the core's queue, the state of the core and the incoming power management requests from the network. Based on the observations, it tracks any changes in the system parameters.

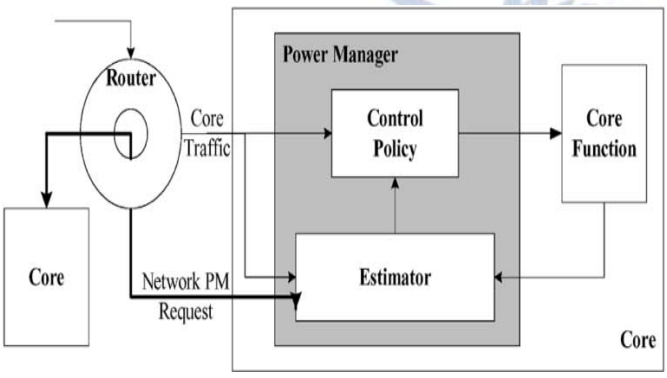

Figure 6: Cores Power Management

When a change is detected, it estimates the new parameter value, and recalculates the power management control using our fast-optimization method. The controller implements power management control calculated by the estimator. It gives commands to the core that determine its performance and energy characteristics (frequency and voltage) in the active state, and chooses when transition the core into one of the available low-power states when the core is idle. To illustrate our power management system operation, we will use the NOC shown in Fig. 1 with streaming MP3 audio example. Let us assume that initially all cores in the NOC are in the sleep state. The MP3 core's controller starts the transition from the sleep into the active state as soon as the user request arrives to it via core traffic input. Right at the beginning of initialization of MP3 decoder, the MP3 core PM also sends a network request via network PM request port to the communications core to start its wakeup process.

Thus, any performance penalty that might have been present without network centric power management is now masked. Next, the MP3 core receives encoded MP3 stream data from the communications core via core traffic input. MP3 estimator watches for the changes in the incoming data and decoding rates or a discrepancy between measured and desired MP3 QoS parameters at runtime. Upon detection, it recalculates the power management control using our system model and our fast optimization method and, thus,closes the optimization loop.

B. Core Parameter

Three main core parameters to estimate include: core frequency and voltage scaling characteristics, the time distribution for servicing incoming core requests, and the characteristics of the low-power states, such as the transition times to/from each state. Some core characteristics can be determined at design time as they depend on hardware parameters alone, such as the number of core frequency and voltage settings. Other parameters need to be tracked at run-time, for example the properties of service time distributions. Each NOC core has at least one main processor. Many of the today's processor have multiple active and low-power states. For example, each of the cores we characterize has one StrongARM processor, as shown in Fig. 1. The processor can be configured at runtime by a simple write to a hardware register to execute at one of eleven different frequencies.

$$P_{\text{uniform}} = \begin{cases} \frac{t - t_{\min}}{t_{\max} - t_{\min}}, & t_{\min} < t < t_{\max} \\ 0, & \text{else.} \end{cases}$$

In addition to the active state, each core can support multiple lower power states, such as: idle, sleep, and off. The core enters idle state as soon as all impeding requests are processed. The low-power state transitions are controlled by the PM. The transition time between the active and one of the low-power states can be best described using the uniform probability distribution shown in (6) where and are the minimum and the maximum transitions times.

C. *Power Estimator*

The main task of the estimator is to observe the system behavior and based on that to estimate the parameters needed for optimization and control. The quality of power management decisions strongly depends on estimator's ability to track changes of critical parameters at run-time. NOC power management requires estimation of workload characteristics, core parameters, and buffering behavior.

D. *Workload Characteristics*

Each core's workload includes request arrivals to the active state (buffer is not empty), the idle states (buffer is empty), and the network request arrivals. We distinguish between active and idle state arrivals, as they are characterized by two different distributions, and they also signify a different type of a decision on the part of the power manager. In the active state, PM decides only on the appropriate frequency and voltage setting, where as in the idle state the primary decision is which low-power state core, should transition to. The distribution of network requests only affects the decisions made by the PM in the idle and the low-power states.

$$P_{workload} = 1 - e^{-\lambda_{workloade} t}$$

E. *Data Coding In Noc Links*

The common characteristic of NoC architectures is that the functional IP blocks communicate with each other via intelligent switches. The data communication between IP's in a NoC takes place in the form of packets routed through a wormhole switching mechanism. The packets are broken down into fixed length flow control units or flits. The switch blocks need to store only a few flits. The header flits carry the relevant routing information. Consequently header decoding enables the establishment of a path that the subsequent payload flits simply follow in a pipelined fashion. There are a few joint crosstalk avoidance and single error correction codes (CAC/SEC) proposed by different research groups. Among these joint codes, the Dual Rail (DR) Code or Duplicate Add Parity (DAP), Boundary Shift Code (BSC) and Modified Dual Rail Code (MDR) reduce the switching capacitance associated with crosstalk from (1+41) CL to (1+21) CL, where 1 is the ratio of the coupling capacitance to the bulk capacitance and CL is the load capacitance, including the self-capacitance of the wire. However, due to intensive integration and device shrinkage in the UDSM era, single error correction will not be sufficient to protect against different transient malfunctions.

F. *Timing Characteristics*

The exchange of data among the constituent blocks in a SoC is becoming an increasingly difficult task because of growing system size and nonscalable global wire delay. To cope with these issues, designers must divide the end-toend communication medium into multiple pipelined stages, with the delay in each stage comparable to the clock-cycle budget. The inter-switch wire segments, along with the switch blocks, constitute a highly pipelined communication medium characterized by link pipelining, deeply pipelined switches, and latency-insensitive component design.

VII. ON-CHIP NETWORK PERFORMANCE ANALYSIS

Network-on-chip provides a structured communication platform for complex SoC integration. However, it aggravates the complexity of on-chip communication design. From the network perspective, there exists a huge design space to explore at the network, link and physical layers. In the network layer, we need to investigate topology, switching, routing and flow control. In the link layer, we can examine the impact of link capacity and link-level flow control schemes on performance. In the physical layer, we could inspect wiring, signaling, and robustness issues. Each of the design considerations (parameters) also has a number of options to consider. From the application perspective, the network should not only be customizable butalso be scalable.

To design an efficient and extensible on-chip network that suits a specific application or an application domain, performance analysis is a crucial and heavy task. The impact of the design parameters at the different layers and the performance-cost tradeoffs among these parameters must be well-understood. The customization on optimality and extensibility can

sometimes be in conflict with each other. For instance, a customized irregular topology may be optimal but not easy to scale. In addition, the analysis task is very much complicated because of the un-availability of domain-specific traffic models. Due to the separation between computation and communication, a communication platform may be designed in parallel with the design of computation. The concurrent developmentspeeds up time-to-market, but leaves the development of the communication platform without sufficiently relevant traffic knowledge. Therefore we must be able to evaluate network architectures and analyze their communication performance with various communication patterns extensively so as to make the right design decisions and trade-offs. Once a network is constructed in hardware, it is difficult, time-consuming, and expensive to make changes if performance problems are encountered.

Design decisions include both architecture-level decisions such as topology, switching, and routing algorithm, and application-level decisions such as task-tonode mapping, task scheduling and synchronization etc.

A. *Flow Summary*

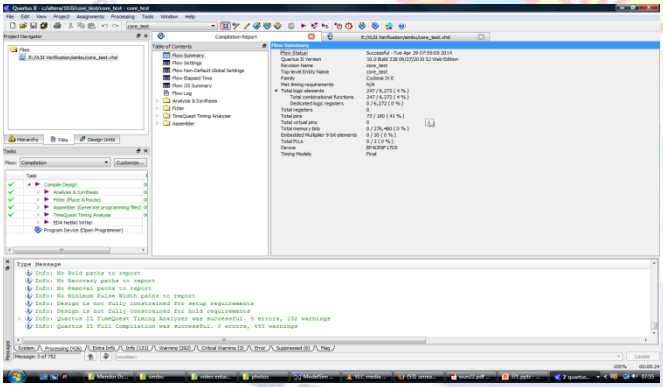

Figure 7: Flow Summary

Flow summary is nothing but usage of logic elements in the chip.here we are using 247/6,272 logic element and power consumed by the logic elements is 4% from the total power. Total number of register is 117/277. The switching strategy determines how a message traverses its route. There are two main switching strategies: circuit switching and packet switching. Circuit switching reserves a dedicated end-to-end path from the source to the destination before starting to transmit the data. The path can be a real or virtual circuit.

A packet typically consists of a header, payload and a tail. The header carries the routing and sequencing information. The payload is the actual data to be transmitted. The tail is the end of the packet and usually contains error-checking code. Packet-switching can be either connection-oriented or connection-less. Connection-oriented communication preserves resources while connection-less communication does not. Connection oriented communication can typically provide a certain degree of commitment for message delivery bounds. With connection-less communication, packets are routed individually in the network in a best-effort manner. The message delivery is subject to dynamic contention scenarios in the network, thus is difficult to provide bounds. However, the network resources can be better utilized.

B. *Network flow control*

The network flow control governs how packets are forwarded in the network, concerning shared resource allocation and contention resolution. The shared resources are buffers and links (physical channels). Essentially a flow control mechanism deals with the coordination of sending and receiving packets for the correct delivery of packets. Due to limited buffers and link bandwidth, packets may be blocked due to contention.

Whenever two or more packets attempt to use the same network resource (e.g., a link or buffer) at the same time, one of the packets could be stalled in placed, shunted into buffers, detoured to an unflavored link, or simply dropped. For packet-switched networks, there exist buffer less flow control and buffered flow control .Buffer less flow control is the simplest form of flow control. Since there is no buffering in switches, the resource to be allocated is link bandwidth. It relies on an arbitration to resolve contentions between contending packets. After the arbitration, the winning packet advances over the link. The other packets are either dropped or misrouted since there are no buffers.

C. *Power Analysis*

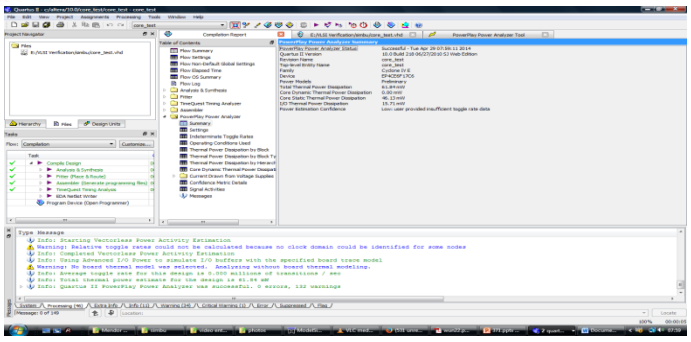

Figure 8: Power Analysis

- Total Thermal power : 61.54mV
- Dynamic power consumption of cores : 46.13mV
- Static thermal power consumption of core : 36.31mV
- I/O Thermal Power decipation : 15.71mV

Generally speaking, Quality-of-Service (QoS) defines the level of commitment for packet delivery. Such a commitment can be correctness of the result, completion of the transaction, and bounds on the performance. But, mostly, QoS has a direct association with bounds in bandwidth, delay and jitter, since correctness and completion are often the basic requirements for on-chip message delivery. Correctness is concerned with packet integrity (corrupt-less) and packet ordering. It can be achieved through different means at different levels. For example, error-correction at the link layer or re-transmission at the upper layers can be used to ensure packet integrity.

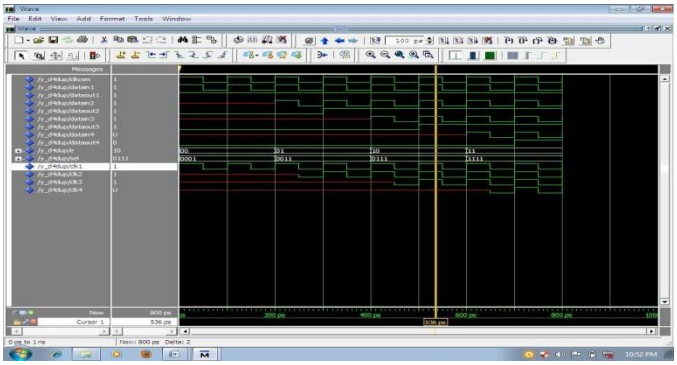

Figure 9: Multiclock output

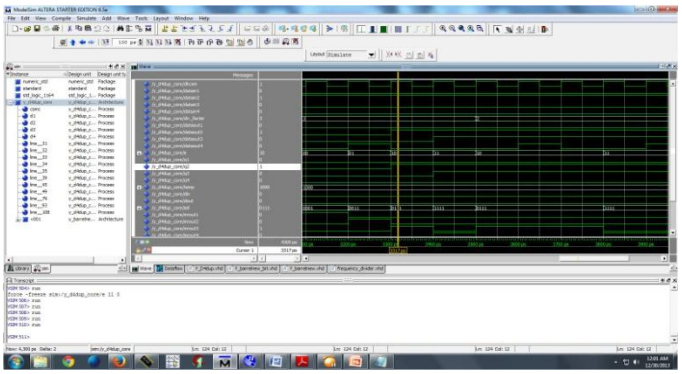

Figure 10: Core Output

VIII. Conclusion

By solving the power consumption and increasing the speed of the chip is difficult .here dynamic voltage/frequency scaling (DVFS) to adjust the power mode of the switches and links to match the traffic flows. The challenge is to predict the traffic flowing through the switches and links in the next time interval. switch or a link can be predicted to be idle for a while, then it can be set to a low power mode or sleep mode by lowering its voltage or frequency. If the data transmission is high the DVFS detect the data flow by using ATPT, Then the data's are shared to other core of the chip. By operating this type of functions we can reduce the power consumption of the chip and increasing the speed of the system.

Appendix

Appendixes, if needed, appear before the acknowledgment.

Acknowledgment

The preferred spelling of the word "acknowledgment" in American English is without an "e" after the "g." Use the singular heading even if you have many acknowledgments. Avoid expressions such as "One of us (S.B.A.) would like to thank ... ." Instead, write "F. A. Author thanks ... ." **Sponsor and financial support acknowledgments are placed in the unnumbered footnote on the first page.**

References

[1] M. Forsell and S. Kumar, Virtual Distributed Shared Memory for Network on Chip, Proc. of the 19th IEEENORCHIP Conference, Nov. 1213, 2015, Kista

[2] YiRan Sun, "Simulation and Performance Evaluation for Network on Chip", MSc thesis, Dept. of Microelectronics and Information Technology, Royal Institute of Technology, Stockholm. May 2014

[3] S. Bell, B. Edwards, J. Amann, R. Conlin, K. Joyce, V. Leung,. Tile64 processor: A 64core soc with mesh interconnect. In SolidState Circuits Conference, 2008.ISSCC 2008. Digest ofTechnical Papers. IEEE International, pages 88–598, Feb. 2010

[4] L. Shang, L.S.Peh, and N. K. Jha. Dynamic voltage scaling with links for power optimization of interconnection networks. In Proc. Ninth International Symposium on HighPerformance Computer Architecture HPCA9 2003, pages 91–102, Feb. 8–12, 2008

[5] Penolazzi S, Jantsch A. A high level power model for the nostrum noc. In: DSD'06: Proceedings of the 9th EUROMICRO conference on digital system design; 2006. p. 673

[6] Seo D, Ali A, Lim WT, Rafique N, Thottethodi M. Nearoptimal worstcase throughput routing for two-dimensional mesh networks. In: ISCA'05: Proceedings of the 32nd annual international

symposium on computer architecture; 2005. p. 432–43

[7] Seo D, Ali A, Lim WT, Rafique N, Thottethodi M. Nearoptimal worstcase throughput routing for two-dimensional mesh networks. In: ISCA'0 Proceedings of the 32nd annual international symposium on computer architecture; 2005. p. 432–43.

[8] Chen G, Chen H, Haurylau M, Nelson N, Albonesi D, Fauchet PM, et al. Electrical and optical onchip interconnects in scaled microprocessors. In: Proceedings of the international symposium on circuits and systems, vol. 3, 2005. p. 2514–7

[10] Ahmed Hemani, Axel Jantsch, Shashi Kumar, AdamPostula, Johnny Oberg, Mikael Millberg, and Dan Lindqvist. Network on chip: architecture for billion transistor era. In Proceeding of the IEEE NorChip Conference, November 2003

[11] Kurt Keutzer, Sharad Malik, Richard Newton, Jan Rabaey, and Alberto Sangiovanni-Vincentelli.Systemlevel design: Orthogonalization of concerns and platformbased design. IEEE Trasnactions on ComputerAided Design of Integrated Circuits and Systems, 19(12):1523{1543, Decmber 2000

[12] P. Meloni, S. Carta, R. Argiolas, L. Raffo, F. Angiolini, Area and power modeling methodologies for networksonchip, in: Proceedings of 1st International Conference on NanoNetworks and Workshops, 2000 (NanoNet'06), IEEE Press, New York, September 2000

[13] D. Bertozzi, A. Jalabert, S. Murali, R. Tamhankar, S. Stergiou, L. Benini, G. De Micheli, Noc synthesis flow for customized domain specific multiprocessor systems onchip, IEEE Trans. Parallel Distrib. Syst. 16 (2) (2005) 113–129