Volume 2, Issue 6
December 2013

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

Control Loop Feedback Mechanism for Generic Array Logic Chip Multiprocessor

[1]V. Karthikeyan and [2]V. J. Vijayalakshmi

[1]Department of Electronics and Communication Engineering, SVS College Of Engineering, Coimbatore, India

[2]Department of Electrical and Electronics Engineering, Sri Krishna College of Engineering & Technology, India

E-mail: [1]karthick77keyan@gmail.com , [2]vijik810@gmail.com

ABSTRACT

Control Loop Feedback Mechanism for Generic Array Logic Chip Multiprocessor is presented. The approach is based on control-loop feedback mechanism to maximize the efficiency on exploiting available resources such as CPU time, operating frequency, etc. Each Processing Element (PE) in the architecture is equipped with a frequency scaling module responsible for tuning the frequency of processors at run-time according to the application requirements. We show that generic array logic Chip Multiprocessors with large inter-processor. First in First Output buffers can inherently hide much of the Generic Array Logic performance penalty while executing applications that have been mapped with few communication loops. In fact, the penalty can be driven to zero with sufficiently large First in First Outs and the removal of multiple-loop communication links. We present an example mesh-connected Generic Array Logic chip multiprocessor. Furthermore, adaptive clock and voltage scaling for each processor provides an approximately 40% power savings without any performance reduction.

Keywords: First-Inputs First Outputs, Processing Element, Chip Multiprocessors, Multiple-Loop Communication Links

1. INTRODUCTION

Dynamic Frequency Scaling (DFS) [5] is widely used technique aimed at adjusting computational power to application needs. It is often associated with Dynamic Voltage Scaling (DVS) therefore enabling to achieve significant power reductions when computing demand is low; some cited benefits also comprises the reduction of thermal hot spots that participate in the accelerated aging of the circuits [4] due to the thermal stress. In core systems such as general purpose processors and high performance embedded processors, the operating system is responsible for dynamically adjusting the frequency of each processor to the current workload. This is facilitated by the presence of dedicated hardware monitors that the Operating System can rapidly access. In Linux based systems, two popular policies are used at kernel-level: on-demand and conservative. The on-demand governor switches to the highest available frequency whenever a load is detected whereas the conservative policy incrementally increases the frequency in a step-by-step fashion, yielding to better power savings at the expense of a lesser reactivity. Fast Fourier Transform (FFT) processor], the latency of a block of data will likely be lower with the source synchronous multi-word flow control method compared to the single transaction handshaking method due to its higher throughput. Dual-clock First in First Outs are well-suited to provide asynchronous boundary communication using the source synchronous multi-word flow control method. Single processor in the Generic Array Logic system Processors utilizes an individual programmable ring oscillator to configured a wide range of frequencies. Each processor also contains two dual-clocks First in First Outs, which write and read in independent clock domains and reliably transfer data across the asynchronous boundaries, GAL's systems need to bring together the circuits a mid clock domain to Consistently [4] transport statistics. Clock phase edge alignment time for unmatched clocks and

synchronization Circuitry introduces a synchronization delay. This delay normally results in a reduction of performance (throughout). In this section, we discuss in depth principles behind how Generic Array Logic clocking affects system throughput and find several key architectural features which can hide the Generic Array Logic effects. Fullyavoiding any Generic Array Logic performance penalties is possible for the described Generic Array Logic chip multiprocessor. [2-3]

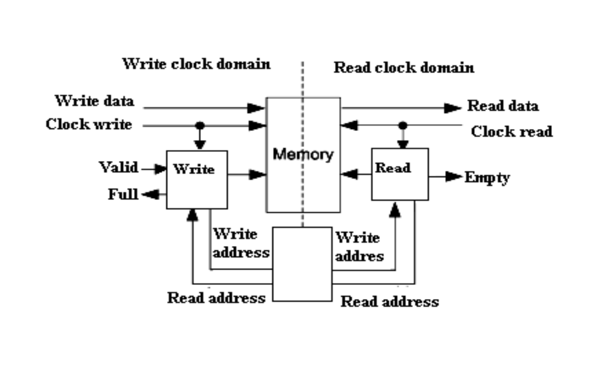


Fig. 1: Block Diagram of Clock domain

2. RELATED WORK

To simplify the discussion, in this section both Generic Array Logic and synchronous systems use the same clock frequencies. A. Related WorkSignificant previous research has studied the Generic Array Logic uniprocessor in which portions of each processor are located in separate clock domains. Results have shown Generic Array Logic Unipro-Caesars experience a nonnegligible performance reduction compared to a corresponding synchronous uniprocessor. This value is sent to the frequency scaling module which will be responsible for scaling up and down the frequency of the processor to cope with application requirements. The procedure is then repeated and the obtained throughput gradually gets closer to the desired throughput. The error value is reduced by assuming that the values of P, I and D have been correctly chosen. The system computes an error value which is obtained from the difference between the desired and obtained throughput. As the output of the PID controller a frequency value is indicated.

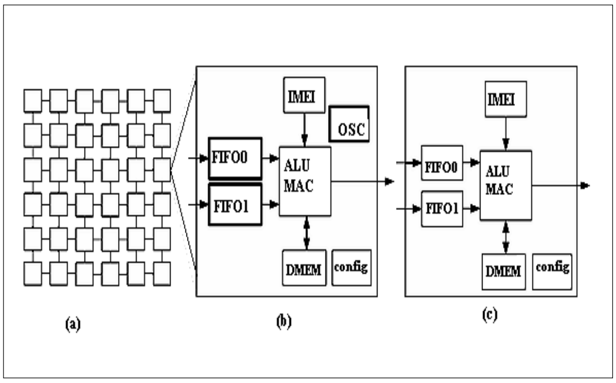


Fig. 2: Reducing and Eliminating Performance Penalties of Generic Array Logic

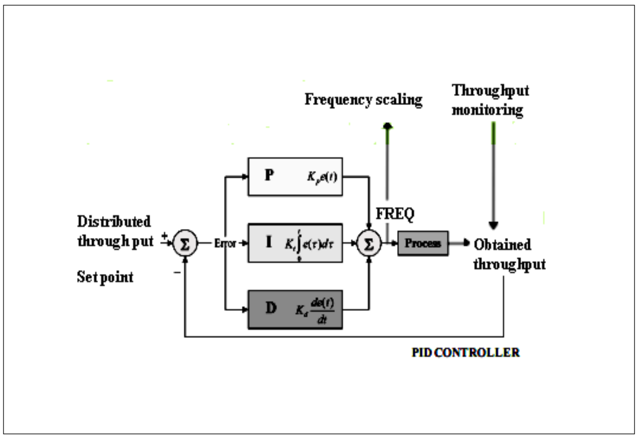


Fig. 3: Block of PID Controller

3. IMPLEMENTATION OF MULTIPROCESSOR

It presents the promising results regarding the adaptability of the system. We have demonstrated the efficiency of the proposed PID controller by presenting three different scenarios. For validating our approach we have implemented a multi-threaded version of the MJPEG decoder together with an Adaptive Pulse Code Modulation and FIR filter applications which exchanges their information's by means of message passing interface (MPI). The networking strategy between processors are strongly impacts the behavior of Generic Array Logic chip multiprocessor. Generic Array Logic chip multiprocessor with inter-processor and processor-memory communicates through a shared global bus. This scheme provides very flexible communication, but places heavy demands on the global bus; thus, the system performance is highly dependent on the level and intensity of the bus traffic.

4. RESULT ANALYSIS

The simple performance in reduction of the Generic Array Logic chip multiprocessor motivates us to understand the factors that affect the performance in Generic Array Logic's style processors. It is a complex relationship and several methods are available to hide the Generic Array Logic penalties. Increasing First in First Out Sizes: Increasing the First in First Out size will reduce First In First Out stalls as well as First in First Out stall loops, and hence increases system performance and reduce the Generic Array Logic performance penalty With a sufficient amount, The top and middle subplots performed with different First in First Out sizes for the synchronous and Generic Array Logic systems, respectively. We assume that all First in First Outs system results the same size and thus, their sizes are scaled together in this analysis.

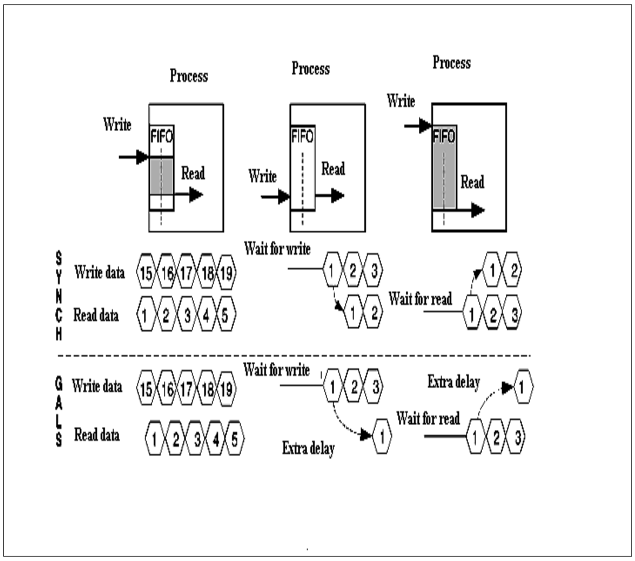
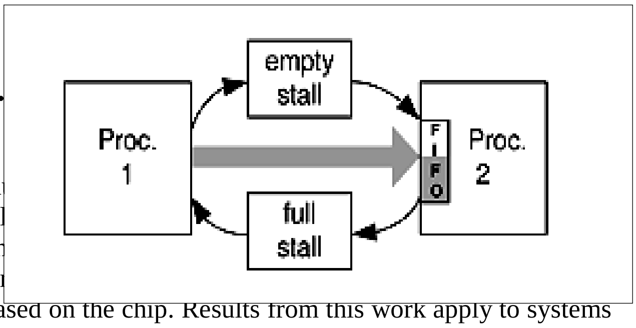
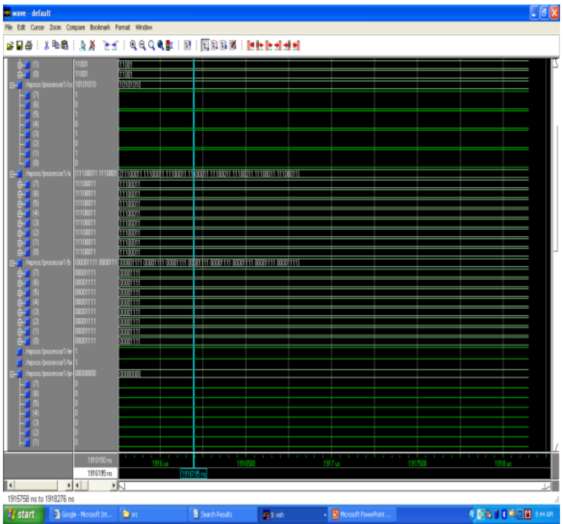


Fig. 4: result Synchronization circuit latency G6

5.



m
Al
on
Ar
based on the chip. Results from this work apply to systems with three key features as 1) Multi core processors (homogeneous and heterogeneous) operating in independent clock domains. 2) Source synchronous multi-word flow control for asynchronous boundary communication. 3) Distributed interconnect, such as mesh. While the results certainly vary over different applications and specific architectures, systems with these features should still exhibit the following benefits over many workloads: good scalability, small performance reductions asynchronous communication overhead, and large potential power reductions from clock frequency and supply voltage scaling. Results show the system’s capability of adapting to disturbing conditions. However, it also introduces the performance penalties due to the additional communication latency between clock domains



6. EXPERIMENTAL RESULTS

Fig. 6: Multiprocessor Module

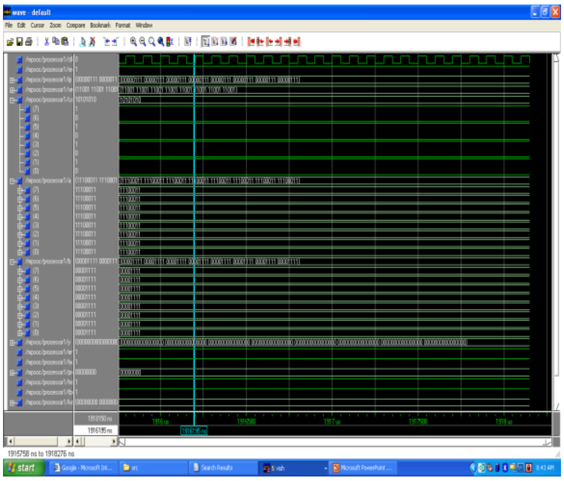


Fig. 7: Generic Array Logic Module

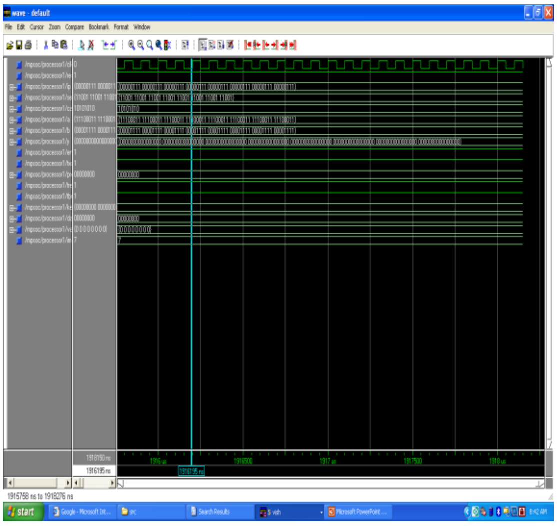


Fig. 8: Multiprocessor with Generic Array Logic Module

REFERENCES

[1] A. Acquaviva, A. Alimonda, S. Carta, and M. Pittau. "Assessing task migration impact on embedded soft real-time streaming multimedia applications" EURASIP J. Embedded System pp.1–15, 2008

[2] A. Alimonda and S. Carta "Impact of task migration on streaming multimedia for embedded multiprocessors: A quantitative evaluation" In ESTI media, pp 59–64. IEEE, 2007.

[3] S. Bertozzi and Supporting task migration in multi-processor systems-on-chip: A feasibility study. In '06. Proceedings, volume 1, pp 1–6, 2006

[4] S. Borkar, T. Kainik, S. Narendra, J. Tschanz, "Parameter variations and impact on circuits and microarchitec-ture," in Proc. IEEE Int. Conf. Des.Autom pp. 338–342, Jun. 2003

[5] F. Clermidy. "Dynamic and Distributed Frequency Assignment for Energy and Latency Constrained MP-SoC." On DATE'09, pp. 1564–1567, Nice, France, 04 2009

AUTHOR PROFILES

Prof V. Karthikeyan has received his Bachelor's Degree in Electronics and Communication Engineering from PGP college of Engineering and Technology in 2003, Namakkal, India, He received a Masters Degree in Applied Electronics from KSR college of Technology and Erode in 2006 He is currently working as Assistant Professor in SVS College of Engineering and Technology, Coimbatore. He has about 8 Years of Teaching Experience

Prof. V.J. Vijayalakshmi has completed her Bachelor's Degree in Electrical & Electronics Engineering from Sri Ramakrishna Engineering College, Coimbatore, India. She Finished her Masters Degree in Power Systems Engineering from Anna University of Technology, Coimbatore, She is currently working as Assistant Professor in Sri Krishna College of Engineering and Technology, Coimbatore She has about 5 years of teaching Experience.