A Mesoporous Pipelining Scheme for High Performance Digital Systems using Asynchronous Cache

Sukanya. K, G. Laxminarayana

Abstract: To relate the increasing behavior of processor and main memory in economical manner, new cache designs and implementations are essential. Cache is liable for themain part of energy consumption. This paper presents an implementation of mesochronous pipelined scheme for high performance digital circuit using asynchronous cache. As a result of the real fact that design of cache memory is time consuming and error prone manner, configurable and synthesizable model generates a particular variety of caches in reproducible and speedy fashion. The mesochronous pipelined cache, implemented by C-Elements which act as a disseminated message passing system. The RTL cache model is implemented in 8×8 multiplier circuit in this paper contains large amount of data and instruction caches and it has a wide array of configurable parameters. Finally, the proposed model produces low delay, reduced area and low power consumption compared to the existing 8 bit multiplication process.

Keywords: Mesochronous pipelined, asynchronous cache, delay, area, power consumption, 8 bit multiplier, RTL model

I. INTRODUCTION

In digital market, popular of processors are related to embedded systems (Tennenhouse, 2000). The microcontrollers, FPGAs DSPs and groupings of SoCs offer solution for embedded system. Since FPGAs have revealed important advances in density, speed and storage capacity, they have been usually employed in general purpose embedded computing systems. Due to the different functionality of FPGAs, complication level rises thus enhancing the number of logic gates used in the circuit. This result showed improved energy dissipation in the digital system (Anderson and Najm, 2004). Another anxiety is the increasing the speed of processor and main memory which produces a bottleneck however retrieving data from memory. Though, the programmers need unlimited fast memory, it is too costly to achieve. Hence, the optimal resolution is to deliver memory hierarchy in which individual levels are faster and more costly per bit than instantaneous higher level (Patterson and Hennessy, 2003). The cache is the initial level in memory hierarchy. Modern multi-core processors use multiple levels of cache to increase the performance. In most such systems, caches are regularly shared by processors for synchronized applications. According to the principle of locality, instruction set showed temporal and spatial locality.

This principle creates the origin of cache memory hierarchy, leads to the performance of the processor and it is dependent on the speed of cache moderately than slow main memory. 45% energy consumption of a processor is donated by cache hierarchy (Segars, 2001).

Research reveals that the alteration in cache memory parameters can deliver decrease in energy consumption to 62% (Ross et al., 2005) and it leads to the performance improvement by 30% (Ross et al., 2004). The complex task of defining a suitable cache configuration for different application, analysis and simulation takes significant computational time period. To reduce area and energy dissipation without conceding the performance, designers want to create strategies for new design space exploration.

Pipelining denotes the execution of instructions concurrently with individual instruction set present at various processing stage. It improves the performance of the system by enhancing the number of instructions is processed in a specific machine cycle without disturbing clock frequency (Nowick, 2011). The number of concurrent instructions executed with the number of pipelining stages which improves throughput. From numerous classifications, pipelines are generally categorized into synchronous and asynchronous counter. A synchronous pipeline has all its modules, administrated by a single global clock pulse, but in an asynchronous pipelining, all modules are independent by different and shake protocols to communicate. Both counters have own set of merits and demerits. The upper sides of asynchronous pipelines are self-governing due to the clock skew and global timing issues (Gupta, Pandey, & Gupta, 2012).

II. LITERATURE REVIEW

While implementing embedded system, one has to compromise energy dissipation, behavior model and cost. The critical task is to choose the best cache model which comprises the group of total cache dimensions, cache line size, and amount of associatively and some other architectural options. These characteristics greatly create an impaction hit rate and energy consumption in cache access. The power consumption arises when (i) cache is accessed (ii) data are transferred to the next memory level throughout the cache miss and also data will be transformed to the idle processor when miss arises (Peeter, 1996). Associatively separates cache into number of ways, each and individual cache is looked up simultaneously during cache access. For some programs, the hit rate of cache is enhanced by increasing number of ways from two to four.
(Patterson and Hennessey, 2003), further than four, the enhancement is not significant. More ways denote more simultaneous look-ups per access leads to more energy per access: a direct mapped cache utilizes only 30% of energy per access as a four way set associated cache (Reinman and Jouppi, 1999).

Clocking is a vital component in digital system design. Switching actions in pipelined system arise in well-defined pattern and exact moments with reference to globally distributed clock signal. In digital system, data and control signals, clock signal will produce leading fan-out and the fastest switching rate. To attain greater clock frequencies, ultra thin super pipelines were used and as a result the load of clock distribution is increasing and it is become tremendously difficult to allocate GHz clock signal (Oklobdzija, 2002). In conventional pipeline schemes, larger currents were consumed by clock network and pipeline registers and it will increase the power consumption. The clock network consumes 50% of the total chip power consumption. These larger currents cause greater IR drops in power supply network, which is mandatory for RLC network. Also, slew rates (di/dt) of large current are linked with on-chip inductance. These power supply noise disturb the power supply integrity and further this is deteriorated due to reducing supply voltage levels (Duarte et al., 2002).

Architecture modifications can remove complex clock distribution to minimize power supply noise, power consumption and increase system performance. Alternative architectures like a synchronous pipelining, wave-pipelining and package wiring have been proposed. While asynchronous pipelining maybe interesting since it completely removes the distribution of clock pulse, it is complex and further linked to synchronous schemes (Friedman, 2001). Our Mesochronous pipeline (MPP) scheme alters pipeline architecture to report the power problems and it is also used to achieve higher performance (Tatapudi and Delgado-Frias, 2005).

In this paper, we present a high performance from MPP scheme with cache for 8 bit multiplier and compare it with the conventional pipeline scheme. The organization of this paper is as follows. In Section II, existing MPP and cache concept was discussed. In Section III, we discussion the implementation of an 8-bit multiplier unconventional and mesochronous pipeline with cache architecture. Performance analysis of the multiplier was presented in Section IV. Finally, concluding remarks were presented in Section V.

III. PROPOSED RESEARCH DESIGN

A. Mesochronous Pipeline Architecture

In a conventional pipeline (CPP) scheme, a digital system is separated into small sub-systems known as pipeline stages divided by pipeline registers. In pipelined system, at any time each stage works on only one data set. While the computation is completed in a particular stage, data is distributed to the next stage of the pipeline. Pipeline registers coordinate this data transfer from one stage to the next stage with the support of globally distributed clock signal. This clock signal must activate all the pipeline registers instantaneously. When new data enter into a stage only after data in a particular stage has been unoccupied. In pipelined system, pipeline stage with extended computation time orders clock-cycle time for the whole system (Gray et al., 1994).

The mesochronous pipeline scheme (MPP) alters CPP scheme to attain higher performance, by reducing the clock distribution it will reduce power consumption. Both MPP and CPP scheme, a digital system is partitioned into pipeline stages. Though, it is clocked such that a pipeline stage is working on more than one data set instantaneously. At any time, multiple data sets may exist in a stage and these data sets are divided based on physical properties of internal nodes. This removes the necessity of pipeline registers. The number of registers eliminated is always dependent on simultaneous data sets without synchronization. This concept is somewhat similar to wave-pipeline scheme (Burleson et al., 1998).

B. Proposed Mesochronous Pipeline for 8 Bit Multiplier using Asynchronous Cache

In this section we presented a multiplier simulated in MPP schemes and illustrated how MPP clocking technique influences on power and performance of mesochronous pipelined system. Carry-Save Adder (CSA) technique is a famous technique frequently used to attain fast multipliers. By using this technique, M-bit multiplier, M layers through 1-bit Full Adders (FA) decrease M-partial products to two partial products still the data flow from the one layer of adders to the next layer of adders. In the last stage of multiplier, two M-bit partial products have combined to generate final product. The adder used for the final integration includes propagation of carry signal. Fast adder implementations are similar to carry-look ahead and it is used to minimize the delay in last layer; though these structures raise the complication for large word sizes and create lessening returns.

Figure 1 Proposed Mesochronous Pipeline for 8 Bit Multiplier Using Asynchronous Cache

Instead of adding M-layers of 1-bit Half Adders, this method combines final partial products. This increases throughput, but there is upsurge in latency. To attain fast multiplier, CSA architecture is pipelined. In MPP scheme, minimum clock period is achieved by making 2M layers into a pipeline stages and it is divided by pipeline registers. Efficiently, MPP multiplier has 2M stages by 2M+1 pipelineregisters.

Published By:
Blue Eyes Intelligence Engineering & Sciences Publication Pvt. Ltd.

284
An 8×8-bit pipelined multiplier was designed with 16 pipeline stages and 17 sets of inter-stage registers. The logic enclosed among any two adjacent register stages maintains multiple data sets concurrently. In this implementation, there are 3 pipeline stages and 4 register stages. The replacement of the registers to cache memory was based on maximum delay difference and that can be controlled by target clock frequency. Figure 1 showed the mesochronous pipeline for 8 bit multiplier using asynchronous cache.

A fast multiplier can be designed if its basic cells have minimum propagation delay. The basic cells in the multiplier have flip-flop, two input AND gate, FA, HA, two input OR gate, and latches. The critical cells in multiplier circuit are FA and HA. A differential transmission-gate implementation has been employed to attain FA and HA (Rabaey et al., 2002). The registers in the multiplier have been recognized using asynchronous cache.

C. Design of Asynchronous Cache Interface

The asynchronous cache interface enables the data transfer from one component to another component using handshaking concept (Putnam et al., 2009). The process of Handshaking was explained as following steps:

Step 1: Primarily all four handshake signals are low.
Step 2: The block A transmits data and at the same time, it moves req_A to high.
Step 3: Subsequently req_B is low, ack_A will move to high. The positive edge of ack_A is used as a clock signal for data register. Thus, data_A is overloaded to the stage register.
Step 4: The ack_A signal is used as an input of subsequent C-Element. When ack_A is moves to high stage, it switches req_B to high.
Step 5: When req_A moves to low, ack_A will also turn into low.
Step 6: The second C-Element delays for ack_B signal from next stage and it goes to high. When ack_B is moved to high, it considers that next stage has read the output data from the register. This roots req_B to move low.

Step 7: Finally, corresponding req_B moves to low, the block B moves ack_B signal to low.

IV. RESULTS AND DISCUSSION

This section validated the results of designed 8 bit multiplier with cache architecture. Once the design synthesis was performed and then the design is transformed and mapped by Xilinx ISE. The HDL coding for the executed 8 bit multiplier cache architecture is fully parameterized. This feature improves design flexibility and it can be easily used with different specification settings. Simulations have been achieved on multiplier layout in TSMC 180nm (drawn length 200nm, 1.8V supply voltage) CMOS technology. A number of simulations have been executed on the full adder to exactly characterize the performance.

A. Performance evaluation

In the design implementation, data cache and instruction cache are produced by 256 lines of 12-bits; four bits are utilized as tags. Therefore, total amounts are 3 kilobytes per cache. To validate the functionality of cache architecture, the gate-level simulations were performed.

RESET signal is used to bring the cache to initial state. It is asynchronous and no operations can be achieved while this signal is in higher level. Then the input is requested from the processor and the address will be in 16-bit from the processor and this address tag is used to carry the instruction. A low value designates a read operation whereas a high value designates a write operation. A and B are the two different input 8-bit data that is to be stored in the cache throughout write operation. Acknowledgement from the higher level memory indicates that it has examined the request from cache for refill process. The data from higher level memory was read then this signal will move to high and the memory request was controlled by memory interface. The data output bus contains 8-bit output data from a read operation for the processor and then the Acknowledgement out message is sent from the cache to the processor once the particular operation has been performed. This output process was shown in Figure 2.

![Figure 2 Gate level simulation of mesoporous pipelined 8 bit multiplier circuit with asynchronous cache](image-url)
The propagation delay for mesochronous pipelining was 280ps (dmin) and proposed mesoporous 8 bit multiplier with asynchronous cache was 200ps (dmax), resulting in a maximum delay variation of 80ps. The power consumption of mesochronous pipelining was 7.6 mW and proposed mesoporous 8 bit multiplier with asynchronous cache was 6.2 mW, resulting in a maximum power variation of 1.4 mW. The energy for mesochronous pipelining was 29.45 mJ and proposed mesoporous 8 bit multiplier with asynchronous cache was 26.63 mJ, resulting in maximum energy variation of 2.82 mJ. The area required for mesochronous pipelining was 0.07243 mm² and proposed mesoporous 8 bit multiplier with asynchronous cache was 0.05821 mm². Performance comparison of mesochronous pipeline scheme and 8 bit asynchronous cache method was shown in Table 1.

V. CONCLUSION

In this work, we have presented the high level implementation of mesochronous pipelined scheme for high performance digital circuits using asynchronous cache. Each of the units of cache architectures interfaced with 8 bit multiplier circuitry asynchronous interfaces. No doubt, the enriched pipelining reduces area and energy with increased control circuitry which is essential for controlling number of handshakes; we have saved significant area by eliminating clock generation, distribution and gating. The implemented cache model is configurable in terms of many parameters, e.g. area, power, energy and delay etc. This parameterization allows implemented model to adapt with any problem specification without altering the developed VHDL code. The results revealed that design effectively attains remarkable performance relatively but it has smaller design complexity while comparing with synchronous parts. Asynchronous design has advantage on automatic power down, our work focuses on designing asynchronous cache with improved simplicity from synchronization.

REFERENCES


K. Sukanya presently working as Associate professor in the department of Electronics and Communication Engineering at TKR College of Engineering & Technology, Medbowli, Meerpally, Saroor Nagar, Hyderabad, Telangana State, INDIA. She has 7 years of teaching experience. She is associated with ISTE as life member. She has obtained B. Tech. degree in Electronics and Communication Engineering from Jayamukhi Institute of Technological Sciences, Warangal, Jawaharlal Nehru Technological University Hyderabad, in 2006, M.Tech. degree in Embedded Systems from Ramappa Engineering College, Warangal, Jawaharlal Nehru Technological University Hyderabad, in 2011 and my area of Research interest is Embedded Systems, Ph.D (ECE) from Jawaharlal Nehru Technological University, Hyderabad and it is my part of Research work.

Dr. G. Laxminarayana presently working as Principal of Anurag College of Engineering. He has obtained BE from Osmania university, M.Tech from Indian Institute of Science, Bangalore and Ph.D from JNTUH under the guidance of Dr. K. Lakshshore (VC of JNTUA). He has 5 years of industrial experience and 30 years of teaching experience. He worked in Osmania University from 1979 to 1998. He worked as Head of the department, ECE at Sreenidhi Engineering College, VBIT and Aurora. He also worked as Director of Aurora Scientific Technological and Research Academy and Principal of Holy Mary Institute of Technology. He is an industrial consultant in instrumentation and worked in South Central Railways. He is associated with IETE for last 20 years and also member of IEEE, ISOR and ISTE. Presently he is an Executive Committee member in the present body of Hyderabad IETE chapter and R&D subcommittee chair at IETE Hyderabad. He is supervising 12 Ph.D students and published various papers in International journals and reputed National journals.