ASYNCHRONOUS VLSI TECHNOLOGY AND DESIGN OF AN ASYNCHRONOUS CORE FOR DIGITAL WATERMARKING OF VIDEOS

Yash Doshi, Siddharth Rajagopal, Parneet Singh Soni
yashdoshi89@gmail.com, siddharthrajagopal@engineerl.com, soniparneet@gmail.com
Vivekanand Education Society’s Institute of Technology, Mumbai 400071.
Under the guidance of Hardik Shah (lecturer, department of Electrical Engineering, ESIT)

ABSTRACT
How would an opera at the Sydney opera house be like without the expertise of the conductor? With the violins, harps, flutes and trumpets playing according their own accord, there would be nothing but clamor. Now, in a microprocessor, if the ALU, memory, other units were to perform their activities without a grand orchestrator—the clock, we could have the result of $1 + 1$ as $3$. That pretty much sums up the role of the clock and its significance in the current concurrent world of logic circuits. While the clock acts as a conductor in the opera of the sequential logic, this clock is also a source of concern for the system. With its own demands of timing constraints, metastability, clock skew and highest power consumption compared to most other components on the chip, the clock seems a necessary evil.

But the long forgotten and newly revived domain of asynchronous VLSI systems is a field that despite having its own share of problems, promises to overcome most of the problems faced by synchronous systems today. This paper looks into the working of the clocked systems in section 1 and traces the history of asynchronous VLSI systems in section 2. The advantages and the challenges in the field and the current state of this technology are looked at in section 3. We discuss the design of an asynchronous core for digital watermarking of videos that we are currently working on in section 4. Section 5 discusses the future possibilities of integration of asynchronous and synchronous systems and section 6 gives an insight into the recent developments in this field.

1. WORKING OF CLOCKED SYSTEMS

1.1. The Payoffs.
An advantage of synchronous chips is that the order in which signals arrive doesn't matter. Signals can arrive at different times, but the register waits until the next clock tick before capturing them. As long as they all arrive before the next tick, the system can process them in the proper order [1]. While designing the clocked systems, the designer does not have to worry about the maximum time required to complete all the operations. Calculating the maximum time required only involves identifying the stage that consumes the maximum time and the number of clock cycles associated with it.

1.2. The downside.
On the flip side, a synchronous chip can only work as fast as its slowest component [2].

Hence, if a particular section in a pipeline takes more time than the others, the other stages are forced to remain idle till the time this stage completes its operation. This is a waste of computing time and also results in wastage of power in these idle circuits.

With exponentially increasing density of the chip, it is becoming increasingly difficult to develop schemes for distributing a global clock in the chip resulting in clock skew. Hierarchies of buses and circuits that adjust clock readings at various components delay the start of a clock tick so that it occurs when circuits are ready to pass and receive data thereby reducing the problem of clock skew [3]. However, physical design considerations enter the picture where the large capacitances and layout routes are realized [4]. A lot of research has gone into effectively routing the clocks to these various blocks in order to achieve minimum clock skew [5], [6], [7].

Finally, the clock itself is always on leading to maximum power consumption in the chip. The registers use energy to switch at every clock tick whether they have any input to process or not adding to power wastage in chips. This is an important consideration for applications like cellular phones and portable electronic gadgets where scarce battery power is invaluable.
2. The Asynchronous System

Consider a traffic light example: Imagine sensors for traffic lights that change the colors according to local conditions, enabling freedom from the central clock. A clocked system must wait until the tardiest signal in the whole bunch makes its transition; a clockless system allows for signals to switch without unnecessary waiting for others [10].

In asynchronous circuits internally generated events/signals take care of the synchronizing the system functions. Once a sender register has sampled the input data, it sends a request to the next stage and waits for the acknowledgement. On receiving the acknowledgement, sender register sends the data and it reaches the receiver register via the combinatorial logic. The handshake can be implemented using either "Delay Padding" or "Completion Detection" [13].

- a) Some sort of redundant data encoding is done so that the computation completion can be detected and this forms the request to the receiver stage. This is an additional area overhead.

- b) Completion detection can also be a part of the receiver stage.

- c) This implementation tolerates data skew

Delay padding is another implementation of asynchronous communication systems. Following are the features corresponding to this:

- a) This is somewhat similar to synchronous communication system except that the delay here can be different (corresponding to the delay of the logic involved) across different stages of a multi-stage pipeline.

- b) Additional overhead is involved because whenever the logic between the register stages changes, the delay should be calculated and re-padded.

- c) Time redundancy is involved (timing margins are required)

Thus asynchronous communication follows the handshake principle. Two stages mutually request and acknowledge for their operations. Thus in case of multistage pipelines, the delay involved in one stage is not going to hurt other stages. Here it is the "average case delay" that enables substantially higher throughput rate as against synchronous systems.

1. The asynchronous system

3.1 The advantages

As already discussed clock skew, a function of load, network distribution, and device mismatch, as well as temperature and voltage gradients affects the performance of the system. As the clock has been done away with in asynchronous systems, the problem of clock skew no longer arises in the design of these systems.

Standard synchronous circuits have to toggle clock lines, and possibly pre-charge and discharge signals, in portions of a circuit unused in the current computation. For example, even though a floating point unit on a processor might not be used in a given instruction stream, the unit still must be operated by the clock. Although asynchronous circuits often require more transitions on the computation path than synchronous circuits, they generally have transitions only in areas involved in the current computation [11].

Synchronous circuits must wait until all possible computations have completed before latching the results, yielding worst-case performance. Many asynchronous systems sense when a computation has completed, allowing them to exhibit average-case
performance. For circuits such as ripple-carry adders where the worst-case delay is significantly worse
than the average-case delay, this can result in a substantial savings [11].

Clockless circuits using domino logic are activated only when the input changes thus they leave unused
circuits ready to respond quickly to other demands. This results in reduced power consumption in the
circuit. Also unlike in synchronous designs where the data moves on every clock edge, causing voltage
spikes, in clockless chips, data doesn’t change at the same time, which spreads out current flow, thereby
minimizing the strength and frequency of spikes and causing less EMI in the nearby circuits.

3.2 The Challenges

The biggest challenge that faces designers today is
the interface of clockless and clocked technologies. The technology that rules the day is that of clocked
circuits where everything is synchronized. But in the world of asynchronous technology there is no global
synchronization. The modules interact with each other whenever they need to and not at every clock
tick. This introduces problems when the two systems
need to be brought together into one unified network.

Within the asynchronous system, the various blocks must communicate effectively with each other. For this there must be adequate handshaking signals incorporated into the system. This adds an
overhead in the circuit design and delay in transmitting the data from one stage to another. For
complex design like a pipelined processor, this will reduce the throughput of the system.

Another challenge that stares the development of the asynchronous designs in the face is the non-
availability of adequate EDA tools. Though the circuits are being simulated and tested using
conventional synchronous CAD tools, the need for a full suite of tools specifically designed for
asynchronous designs is becoming increasingly felt as more asynchronous designs are hitting the labs.

There is no benchmark for the performance characterization of these asynchronous circuits. This
poses a problem when it comes to testing these circuits. Since the testing methods are heavily
committed to clock for applying stimuli and observing the outputs, the timing driven test is not
suitable for the even driven world of asynchronous circuits.

1. Design of an asynchronous core for
watermarking of videos

The need for watermarking is restricted only by the
uses it can be put to. Watermarking is applied to all
the domains where security is a concern. Digital
watermarking of videos finds applications in a variety
of arenas ranging from contenting authoring, to use
as legal evidence to security.

The advantages of the asynchronous VLSI
technology over the conventional synchronous VLSI
technology have motivated us to develop an
asynchronous core for digital watermarking of
videos. Herein we discuss the design of an
asynchronous core for the same.

![Asynchronous Core for Watermarking of Videos](image)

a) Architecture

The basic blocks in the core are the control unit,
the DCT generator, the quantizer and adder. The
heart of the design is the control, unit. The control
unit is responsible for synchronizing the operations of
DCT, quantization and addition via control signals
that it generates in response to the request signals that
it receives from the individual modules. The DCT
generator generates the 2D DCT of the incoming data
(watermark or the image) using Chen’s algorithm
which is a fast algorithm for the computation of 2D
DCT. The quantization is carried out only on the
DCT of the image. This operation discards the 2-D
DCT high frequency and small amplitude coefficients
in the transform of the image. This reduces the
memory requirement for storing and transmitting the
video frame. However, the process of quantization
also causes some irreversible loss of data in the
original video frame. Quantization can be performed
either in a fixed manner wherein the data is quantized
as per a predetermined rule, or in an adaptive manner to minimize the error between the original image and the quantized image. However adaptive quantization requires more computation and is more time consuming. Hence fixed quantization is preferred. The adder used in the design can be chosen from among the Nielsen adder, Chong adder or the dynamic full adder [14]. These adders were chosen for their high speed and low energy to minimize the power consumption in the entire circuit.

b) Operation

At power up, the control unit issues a read command to the watermark register. When the read operation is completed, the register issues a done command, which triggers the computation of the 2D DCT of the watermark. As the DCT operation is being completed in a pipelined manner, the control unit issues a read command to the image memory. After computation of the DCT of the watermark, the DCT is stored in RAM. When the control unit receives a done command from the image register, the image DCT is computed. On a request from the DCT generator, the coefficients of the DCT of the image are quantized to reduce the memory requirement. The control unit then sends out an add command to the adder, which adds the DCT of the image and that of the watermark to produce a composite image in the frequency domain.

1. The future of asynchronous VLSI

In the near future, Handshake Solutions and ARM, a chip-design firm, plan to release a commercial asynchronous ARM core for use in devices such as smart cards, consumer electronics, and automotive applications. Sun Microsystems is already building a supercomputer with at least 100,000 processors, some using asynchronous circuits [1]. Though a completely asynchronous chip may not be a possibility in the near future, islands of clockless modules could be connected together, with the clock synchronizing only the data transfer between the various modules. Another approach would be to have “globally asynchronous and locally synchronous chips” wherein synchronous islands would communicate within the fabric of the asynchronous design.

2. Recent Developments

Previous asynchronous designs have had lackluster performance due to the use of only combinational logic to achieve the end result. This meant using use of larger and slower p-type transistors leading to larger chip sizes, and slower performance, particularly for clockless processors. But now there has been a shift in the way the design of asynchronous systems is being approached. With the use of domino logic and delay insensitive mode in processors, integrated pipelines have been possible making the designs faster. The use of domino logic enables evaluation of several lines in the same cycle, rather than a single line of data per cycle. Domino logic is also efficient because it acts only on data that has changed during processing, rather than acting on all data throughout the process. The delay-insensitive mode allows an arbitrary time delay for logic blocks. This enables registers to communicate at their fastest common speed. If one block is slow, the blocks that it communicates with slow down. This gives a system time to handle and validate data before passing it along, thereby reducing errors and increasing reliability of the system [1].

Companies have announced microprocessor prototypes and support circuitry that incorporate elements of clockless technology and are planning to gradually integrate an “island” of clockless logic into future-generation chips. Some companies are already marketing clockless chips that, for example, give pagers up to twice the battery life of their competitors’ products. Furthermore, designers are already investigating the use of clockless chips for mobile devices and smart cards. These innovations are based on the so-called null convention logic (NCL) as a way of letting clockless chips know when an operation is complete [2].

Sun’s UltraSPARC III processor for servers and workstations already features asynchronous circuits. Fulcrum Microsystems offers an asynchronous PivotPoint high-performance switch chip for multigigabit networking and storage devices. The company has also developed clockless cores for use with embedded systems [1]. With these developments paving the way for newer developments, asynchronous VLSI systems is like a genie out of a bottle offering lower power, greater robustness and greater reliability, the wishes of any designer.

CONCLUSION

Standing here at a point where the synchronous designs seem to be fast heading towards their maximum utilization, it would only be wise to acknowledge the untapped potential of the asynchronous VLSI domain and to embrace it with open arms. Further research towards the active development of EDA tools required for the design and synthesis of the asynchronous systems rather than using the current ones adapted to synchronous
systems would be more than welcome. Also as more
and more people are awakening to the domain of
asynchronous VLSI systems, the work on these
systems should be pursued more vigorously. A
successful integration of the long lost cousins,
asynchronous and synchronous VLSI systems can
open up possibilities that only seem a mirage in the
vast deserts of applications.

REFERENCES

1. David Geer’s article on clockless chip technology-
www.geercom.com/clockless chips.html

2. Guest Editors’ Introduction: Clockless VLSI
systems Volume 20, Number 6, November/December
2003 IEEE Design & Test of Computers by Soha
Hassoun, Yong-Bin Kim, Fabrizio Lombardi.

3. Clock distribution networks in synchronous
digital integrated circuits Friedman, E.G.
Proceedings of the IEEE Volume 89, Issue 5, May
2001 Page(s):665 – 692

4. Introduction to VLSI circuits and systems, John
P Uyemura, Wiley student edition 2002-03. Pg 598.

5. A Global Minimum Clock Distribution Network
Augmentation Algorithm for Guaranteed Clock Skew
Yield Bao Liu; Kahng, A.B.; Xu Xu; Jiang Hu;
Venkataraman, G. Design Automation Conference,
2007. ASP-DAC apos;07. Asia and South Pacific

6. A Buffer Distribution Algorithm for High-
Performance Clock Routing Jun Dong Cho;
Sarrafzadeh, M. Design Automation, 1993. 30th
Conference on Volume , Issue , 14-18 June 1993
Page(s): 537 – 543.

7. Design and analysis of a hierarchical clock
distribution system for synchronous standard
cell/macrocell VLSI Friedman, E.G.; Powell, S.
Solid-State Circuits, IEEE Journal of

8. Introduction to asynchronous VLSI systems and
a brief history of the work-
http://editorialpinch.com/async/?tag=asynchronous-
vlsi-system