

# Clock Scheme for FPGA Implementation of Globally Asynchronous Locally Synchronous (GALS) Circuits

Gouri Wazurkar<sup>1\*</sup> and Sanjay L. Badjate<sup>2</sup>

<sup>1</sup>Shri Ramdeobaba College of Engineering and Management, Nagpur - 440013, Maharashtra, India; morankarg@rknc.edu

<sup>2</sup>S. B. Jain Institute of Technology, Management and Research, Nagpur - 441501, Maharashtra, India; s\_badjate@rediffmail.com

## Abstract

**Objective:** In this paper clock scheme for FPGA implementation of globally asynchronous locally synchronous circuits to achieve low power dissipation by reducing switching power consumption in a circuit is proposed. **Method:** Clock scheme for globally asynchronous locally synchronous (GALS) using clock divider and decoder module is evaluated. Clock divider and decoder module mainly divides a global clock into low switching rate control signals that simplifies in circuit clock management modules and reduce global clock rate. **Findings:** Asynchronous or clockless designs are considered aggressively for replacement of conventional synchronous digital system design. The major advantages of asynchronous are low power dissipation, higher speed and robustness. Traditional Quasi-Delay-Insensitive (QDI) circuits are nearly impossible to be mapped onto commercial FPGAs. The accurate functionality of a circuit using a bundled-data scheme depends on the assumption that the logic and routing delay of each block is predictable. Thus asynchronous circuit designers can use a delay-matching block to satisfy timing constraint. Clock divider and decoder modules does not incorporates delay matching block. Global clock is finely partition to the low rate control signals that results in low power dissipation with less complex circuitry and most importantly facilitates FPGA implementation. **Applications:** The proposed clock scheme using clock divider and decoder module may be incorporated in the GALS implementation of digital signal processor on FPGA for wireless sensor nodes.

**Keywords:** Clock Scheme, FPGA, GALS, Low Power, Switching Power

## 1. Introduction

Due to advancement in semiconductor technology and internet of things (IoT), many systems are increasingly connected with wireless sensor network. Most important issue in the design of wireless sensor nodes is power dissipation of each node and network as a whole. Low power operation is desirable in all wireless sensor node applications for long sensor node life. Most system-on-chip (SoC) such as processors, employs the global synchronous logic signaling scheme where synchronization is based on a global clock or its variants. As the systems becomes increasingly large and complex, this global clock signal may result in challenges such as clock skew, timing delay and clock

management. There is however a trend, as detailed in the International Technology Roadmap for Semiconductors (ITRS)<sup>1</sup>, for an increasing use of asynchronous logic from the present 15 % to 49 % in 2024. In some of these SoCs, asynchronous signaling scheme<sup>2,3</sup> were used for synchronization between the different fully synchronous modules using global clock signal. Asynchronous signaling schemes are used for both between modules (inter-modules) and within modules (intra-module). This hybrid inter-module asynchronous cum intra-module synchronous, termed Globally-Asynchronous-Locally-Synchronous (GALS) may be advantageously exploited to simplify some challenging design issues<sup>4,5</sup> so as to achieve minimum power dissipation.

\*Author for correspondence

Xilinx Spartan / vertex series FPGAs and Altera Cyclone series FPGAs are designed for synchronous (clocked) circuits and not for asynchronous circuits. Mapping an asynchronous circuit onto these FPGA is difficult due to its basic structure. The main difficulty comes from variable delays caused due slices, switches and wires. Routing is implemented using local switchboxes (LSBs) and global switch-boxes(GSBs)<sup>6</sup>. An output (except for the carry signals) of a slice cannot directly connect to an input of a slice nearby<sup>6</sup>. Longer and variable delay is estimated since the connections are routed via switch-boxes. Thus it is difficult to implement asynchronous elements on FPGA, especially for those with feedback signals.

GALS approaches may be generally categorized by its clocking schemes, pausable clocking and the data-driven clocking. The major advantages of asynchronous circuits over synchronous circuits are low power dissipation, higher speed and robustness. The pausable clocking scheme involves a free running local clock that is only ceased to avoid meta stability during a data transfer. The data-driven clocking scheme activates the local clock only when there is a need for data transfer. From a broad perspective, the former scheme is advantageous for low latency while the latter advantageous for low power dissipation; the actual attributes may vary, depending on the specific operating conditions and circuit realizations<sup>2</sup>. Traditional Quasi-Delay-Insensitive (QDI) circuits are nearly impossible to be mapped onto commercial FPGAs. The accurate functionality of a circuit using a bundled-data scheme depends on the assumption that the logic and routing delay of each block is predictable. Thus asynchronous circuit designers can use a delay-matching block to satisfy timing constraint. Placement and routing of bundled-data scheme is most important and governing factor for successful implementation of bundled data scheme on FPGA. Large numbers of iterations are required to place and route delay matching blocks. Also it is difficult to implement asymmetrical delay-matching blocks in FPGAs, therefore 2-phase handshake latch controller<sup>2</sup> is used. To accelerate time of prototyping, 2-phase bundled data scheme may not be adopted since it requires delay matching blocks that need precise placement and routing.

In this paper, clock scheme for FPGA implementation of globally asynchronous locally synchronous circuits to achieve low power dissipation by reducing switching power consumption in a circuit is proposed. For fair benchmarking fully synchronous circuit and GALS pipe-

lined circuit was implemented using same FPGA and same logic cells. The paper is organized as follows section 1 introduces to GALS and FPGA implementation of asynchronous circuits, section 2 illustrates proposed clock divider and decoder module, results are discussed in section 3 and finally concluded in section 4.

## 2. Clock Divider and Decoder Module

### 2.1 Power Dissipation

In a CMOS circuit, power dissipation<sup>8</sup> can be summarised by

$$P_{avg} = P_{switching} + P_{short} + P_{Leakage} \quad (1)$$

$$P_{avg} = fC_L V_{dd}^2 + I_{sc} V_{dd} + I_{Leakage} V_{dd} \quad (2)$$

where, frequency of operation is  $f$ ,  $C_L$  is load capacitance,  $I_{sc}$  is short circuit current,  $I_{Leakage}$  is static leakage current. The first two components are the dynamic power dissipation caused by switching activity at the various nodes within the circuits, while the third component is caused by static leakage. Three main methods for reducing power dissipation in CMOS circuits are

- Reducing supply voltage  $V_{dd}$
- Reducing load capacitance  $C_L$
- Reducing switching or operating frequency  $f$

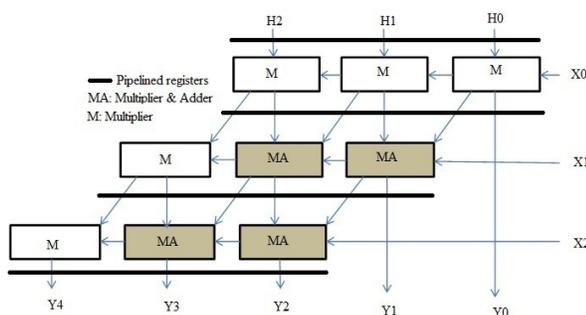
The switching activity that takes place within a circuit can be classified into two components: activity which is required to calculate the desired result, and unnecessary activity that occurs as a result of other activity within the circuit. Reducing unwanted activity in the circuit is very important. It can be reduced by adopting various methods such as number systems, signal encoding, algorithm design and using registers or latches between consecutive stages. Low power dissipation in FPGA circuits can be achieved by reducing switching rate  $f$  using GALS approach.

### 2.2 Clock Divider and Decoder Module

In general, the Very Large Scale Integration (VLSI) systems SOCs in the nano scale may not operate under the control of a single clock and may require asynchronous techniques. In a synchronous system, all activity is gov-

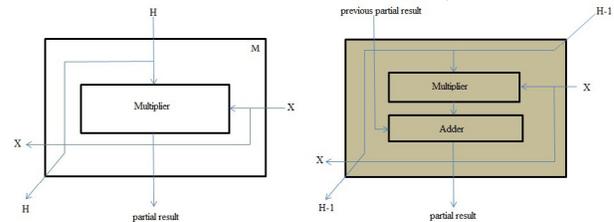
erned by the single global clock. Data is captured by latches in the pipeline at a particular point with the rising or falling edge of clock cycle<sup>9</sup>. The large parameter variations and power dissipation across a chip may make it very expensive to control delays in global clock signal. Therefore, it is necessary to introduce GALS system to reduce power dissipation, simplify global clock management module and reduce global clock rate. Particularly it is also essential to reduce power dissipation in the circuit when implemented on FPGA where switching rate  $f$  is the only controlling parameter.

Clock divider and decoder module consist of Moore machine. A finite state machine (FSM) whose output depends only on the present state and only input to the machine is a global clock signal. A state machine can be visualized by means of state-transition diagram and table. State diagram and table describes the control and execution flow. That clearly simplifies the process of controlling logic, systematizes data path and reduces chance of error. The number of states (s) can be determined by the number of execution steps required to complete a desired operation. Each state is represented by number of bits exactly equal to required number of control signals (n). Binary representation (n-bit) of each state is determined by the status (1 or 0) of each control signal. Thus a state diagram or table can be generated where each state indicates current status of all the control signals. Transition of states indicates change / no change in the status of every control signals. To justify the proposed method and its application to GALS, FPGA implementation of super systolic array for convolution is performed. Convolution is one of the fundamental operations in DSP processors used in wireless sensor nodes. It can determine the output of any linear time invariant systems given the input signal and impulse response. VLSI has made implementation of system hardware or even highly parallel array processors economically feasible and technically realizable. For fair



**Figure 1.** Super systolic array for convolution.

benchmarking fully synchronous pipelined super systolic array for convolution and GALS pipelined super systolic array for convolution is implemented using same FPGA and same logic cells. Figure 1 illustrates block diagram of super systolic array for convolution, length of the input signals is kept at three thus the length of the output signal is five and Figure 2 illustrates the corresponding internal structure of cells in super systolic array.



**Figure 2.** Internal structure of cells in super systolic array.

### 2.2.1 Fully Synchronous Pipelined Super Systolic Array for Convolution

Fully synchronous super systolic array<sup>10</sup> for convolution consist of multiplier, adder and pipelined registers<sup>11,12</sup>. In this circuit 8-bit Wallace unsigned partial multiplier and 16-bit carry look ahead adders are selected for implementation so as to perform operation of multiplication and addition respectively. The single global clock signal is applied to all the pipelined registers for synchronization and driving inputs and / or partial results and / or outputs to next stage. Since the global clock signal is applied to all pipelined registers, clock divider and decoder module is not required. As shown in Figure 1, the complete circuit is divided into four stages hence requires four pipelined registers to deliver final result of convolution operation. Due to pipelined operation, the global clock frequency  $f$  can be increased by multiplying with the number of pipelined stages that significantly increases speed of operation, switching rate and dynamic power dissipation according to equation (2).

### 2.2.2 GALS Pipelined Super Systolic Array for Convolution

GALS pipelined super systolic array for convolution additionally consists of clock divider and decoder module. The main purpose of this module is to decode clock signal and provide essential control signals to pipelined registers using proposed clock scheme. Decoder module decodes the clock signal and generates control signals

for pipelined registers that enables them to drive data to next stage. Since all pipelined registers receive distinct control signals, they are asynchronous in operation. It restricts the free flowing clock to clock divider module only. Therefore, the module receives single clock signal called as locally synchronous. The higher clock frequency signal is decoded and generates control signal at much lower frequency (clock frequency divided by number of pipelined stages). It does not affect the speed of operation but certainly reduces switching rate and dynamic power dissipation according to equation (2). As shown in Figure 1 four distinct control signals are generated to enable pipelined registers sequentially at the arrival of the rising (or falling) edge of the global clock signal. The number of states ( $s = 4$ ) to obtain the final result of operation, the number of control signals ( $n = 4$ ) since four pipelined stages therefore each state is represented with 4-bits. Table 1 shows the state transition table for the clock divider and decoder module. To design the Moore machine, flip flop can be selected and its corresponding inputs and simplified Boolean expression can be obtained using conventional digital techniques. Figure 3 shows the simulation output of clock divider and decoder module obtained after coding in VHDL.

Table 1. State transition table

| Present state |    |    |    | Next state |    |    |    |
|---------------|----|----|----|------------|----|----|----|
| C1            | C2 | C3 | C4 | C1         | C2 | C3 | C4 |
| 1             | 0  | 0  | 0  | 0          | 1  | 0  | 0  |
| 0             | 1  | 0  | 0  | 0          | 0  | 1  | 0  |
| 0             | 0  | 1  | 0  | 0          | 0  | 0  | 1  |
| 0             | 0  | 0  | 1  | 1          | 0  | 0  | 0  |

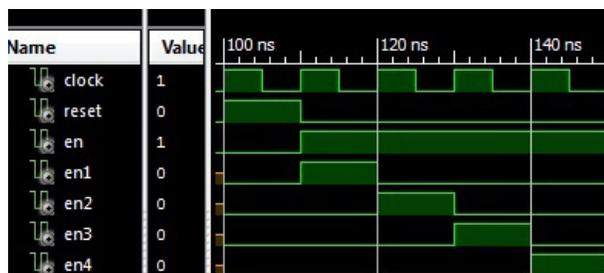


Figure 3. Simulation output of clock divider and decoder module.

### 3. Results and Discussion

The fully synchronous and GALS pipelined super systolic array convolution for input signals whose length ( $L = 3$ )

using 8-bit Wallace unsigned multiplier, 16-bit carry look ahead adder and pipelined register (D flip flop) was coded in VHDL and implemented in virtex 5 FPGA (xc5vlx20t-2ff323) device. The obtained results are also confirmed on other FPGA devices such as Spartan 5, vertex 6, vertex 6 (low power), Spartan 6 and Spartan 6 (low power). Virtex-5 FPGAs offer the best solution for addressing the needs of high-performance logic designers, high-performance DSP designers, and high-performance embedded systems designers with unprecedented logic, DSP, hard/soft microprocessor, and connectivity capabilities<sup>13</sup>. Built on a 65-nm state-of-the-art copper process technology, Virtex-5 FPGAs are a programmable alternative to custom ASIC technology<sup>13</sup>.

Table 2. Results

| FPGA Resources / Parameters | Synchronous | GALS       |
|-----------------------------|-------------|------------|
| Number of Slices            | 218         | 242        |
| Number of LUTs              | 1285        | 1413       |
| Number of FFs               | 1321        | 1511       |
| Delay                       | 7.384 ns    | 7.384 ns   |
| Maximum Frequency           | 135.43 MHz  | 135.43 MHz |
| Clock Frequency             | 500 MHz     | 500 MHz    |
| Total Power                 | 2.102 W     | 0.63 W     |
| Dynamic Power               | 1.662 W     | 0.069 W    |
| Leakage Power               | 0.44 W      | 0.561 W    |

The output of each block of circuit is verified using Xilinx ISE web pack 13.1 simulation and synthesis tool. Table 2 summarizes the result obtained after simulation and implementation of circuit. Results clearly indicate that fully synchronous circuit dissipates 3.33 times more power as compared to GALS circuit. But at the cost of increased area GALS circuit requires 1.099 times more number of slice LUT as compared to fully synchronous circuit. Power efficiency is mainly because of fine partitioning of global clock. It simplifies in circuit clock management module, since global clock is limited to single module. The control signals generated are at much lower frequency thus reduce global clock rate. The synthesis results also show that the total delay of the circuit is 7.384 ns, indicating maximum operating frequency is around 135.43 MHz. Since the total circuit is divided into four pipelined stages therefore the maximum operating frequency can be 541 MHz assuming all stages has

equal delay. If circuit is operated at this higher frequency in fully synchronous mode dissipates large amount of power. GALS circuit can also be operated at this frequency but clock divider and decoder modules generates control signals for pipelined stage at frequency of 135.43 MHz that reduces total power dissipation significantly. Table 2 shows the power dissipation for fully synchronous and GALS circuit at operating frequency of 500 MHz. The total power dissipation and operating speed can be further improved by implementing efficient multipliers and adders. The objective of the paper was only to demonstrate power efficiency using proposed clock scheme for FPGA implementation of GALS circuits.

## 4. Conclusion

The fully synchronous and GALS pipelined super systolic array convolution for input signal whose length ( $L = 3$ ) using 8-bit Wallace unsigned multiplier, 16-bit carry look ahead adder and pipelined register (D flip flop) was coded in VHDL and implemented in virtex 5 FPGA (xc5vlx20t-2ff323) device. The primary objective of the paper was to demonstrate power efficiency by reducing switching power consumption in a circuit using proposed clock scheme for FPGA implementation of GALS circuits. For fair benchmarking fully synchronous circuit and GALS pipelined circuit was implemented using same FPGA and same logic cells. Fully synchronous circuit dissipates 3.33 times more power as compared to GALS circuit, but at the cost of increased area GALS circuit requires 1.099 times more number of slice LUT as compared to fully synchronous circuit. Power efficiency is mainly because of fine partitioning of global clock. It simplifies in circuit clock management module, since global clock is limited to single module. The proposed clock scheme may be used in the implementation of GALS circuits for digital signal processing applications.

## 5. References

1. Semiconductor industry association. International Technology Roadmap for Semiconductors [Internet]. [cited 2015 Jun 05]. Available from: <http://www.itrs.net>.
2. Plana LA, Furber SB, Temple S, Khan M, Shi Y, Wu J, Yang S. A GALS infrastructure for a massively parallel multiprocessor. *IEEE Design and Test of Computer*. 2007 Sep–Oct; 24(5):454–63.
3. Dasgupta S, Yakovlev A. Comparative analysis of GALS clocking schemes. *IET Computer and Digital Technology*. 2007 Mar; 1(2):59–69.
4. Chong K-S, Chang K-L, Gwee B-H, Chang JS. Synchronous-Logic and Globally-Asynchronous-Locally-Synchronous (GALS) acoustic digital signal processors. *IEEE Journal of Solid-State Circuits*. 2012 Mar; 47(3):769–80.
5. Dobkin R, Ginosar R, Sotiriou CP. Data synchronization issues in GALS SoCs. *Proceedings of International Symposium on Asynchronous Circuit and Systems, Greece*; 2004. p. 170–9.
6. Maxfield CM. *The design warrior's guide to FPGAs*. Publication Company: Elsevier; 2006.
7. Yun KY, Beerel PA, Arceo J. High performance asynchronous pipeline circuits. *IEEE International Symposium on Advance Research in Asynchronous Circuits and Systems, Japan*; 1996. p. 17–28.
8. Weste N, Eschragian K. *Principles of CMOS VLSI design: A systems perspective*. Publication Company: Addison-Wesley; 1988.
9. Kaur N, Kumar R. Hybrid topology control based on clock synchronization in wireless sensor network. *Indian Journal of Science and Technology*. 2016 Aug; 9(31). DOI: 10.17485/ijst/2016/v9i31/95334.
10. Lee J-J, Song G-Y. Implementation of the super-systolic array for convolution. *IEEE Proceedings of Design Automation Conference, Japan*; 2003. p. 491–4.
11. Sarada V, Vigneswaran T. Low power 64 point FFT processor. *Indian Journal of Science and Technology*. 2016 Jan; 9(4). DOI: 10.17485/ijst/2016/v9i4/83322.
12. Chitra E, Vigneswaran T. An efficient low power and high speed distributed arithmetic design for FIR filter. *Indian Journal of Science and Technology*. 2016 Jan; 9(4). DOI: 10.17485/ijst/2016/v9i4/79055.
13. Xilinx. *Virtex-5 Family Overview* [Internet]. 2015. [cited 2014 Apr 30]. Available from: <http://www.xilinx.com>.