Design & Estimating the Performance for Asynchronous Pipelining using FIR filters on FPGA

G.Naveen
PG Scholar, VLSI & ES,
Dept of ECE,
Vidya Bharathi Institute of Technology,
Janagaon, Warangal, Telangana.

B.Kranthi Kumar, M.Tech
Associate Professor,
Dept of ECE,
Vidya Bharathi Institute of Technology,
Janagaon, Warangal, Telangana.

Abstract:

The asynchronous paradigm has interesting features due to the lack of the clock signal and it is another option for the project of digital systems. This paradigm has several design styles, where the micropipeline style is the most suitable one for FPGA platforms, due to the simplicity of its control. In this paper, we propose a pipeline archi-tecture to implement asynchronous digital systems, in bundled-data micropipeline style, considering FPGAs as target devices. Through a case study, we show that the proposed architecture presents a 29% decrease in latency time and a 13% increase in throughput, compared with the state of the art architecture MOUSETRAP.

I. INTRODUCTION:

FPGAs devices (Field Programmable Gate Array) have become a very popular way to develop and implement digital circuits due to their cost and design time. High per-formance FPGAs are implemented in Deep-Sub-Micron MOS technology (DSM-MOS). In this technology, the delay on the lines should be considered and may be higher than the delay of the logic gates [1], [2]. Digital systems are traditionally designed in synchronous paradigm, i.e. they use a global clock signal to synchronize their opera-tions. They are quite popular due to their simplicity of de-sign and availability of commercial CAD tools for auto-matic synthesis. In DSM-MOS technology, a clock signal requires attention due to its noise generation, electromag-netic interference and power consumption.

Besides these factors, the distribution of the clock signal along the chip is a task with increasing complexity because of the clock skew problem, which drops the system performance. The overhead caused by the clock signal can reach 130% in a VLSI (Very Large Scale Integration) implementation [3] and worsens when FPGAs are employed. Consider-ing the design style of asynchronous systems, it is easier to implement the micropipeline style in commercial FP-GAs due to the simplicity of its control [4], [5]. This is important because of the difficulty to achieve hazard-free Considering the design style of asynchronous systems, it is easier to implement the micropipeline style in commer-cial FPGAs due to the simplicity of its control [4], [5].

This is important because of the difficulty to achieve hazard-free [7], [8], [9], [10], [11], [12], [13]. However, most of these pipeline architectures are focused on VLSI implementations, employing full-custom control [8], [9], [10]. FPGA oriented pipelines have been proposed [12], [13], but one uses a complex delay mechanisms [12] and the other does not obey the fundamental-mode [13]. The pipeline architecture known as MOUSETRAP, pro-posed in [11], has a good performance, it is based on logic gates and can be implemented on FPGAs (see Fig. 1).In this paper, we propose a FPGAs oriented architecture for micropipeline design style (see Fig. 2). The proposed lin-ear pipeline architecture employs flip-flops as registers, due to the large availability of these elements in FPGAs. The control is based on logic gates and it is mapped into two LUTs (Look-Up Table).

The new pipeline architecture has a better performance, whencompared to the MOUSETRAP architecture im-plemented on FPGA, because it has a smaller latency and a greater throughput. A digital FIR filter of fourth order shows the efficiency of our architecture on a FPGA plat-form.
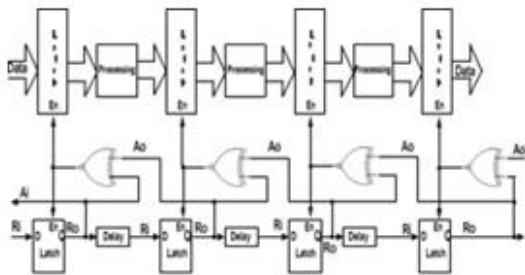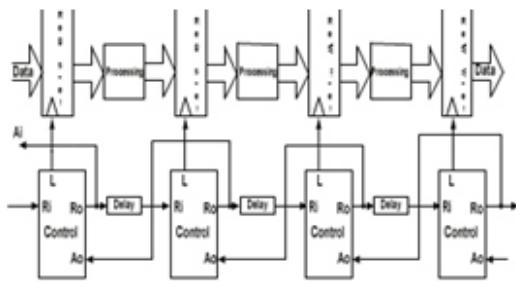

Fig. 1. MOUSETRAP linear micropipeline

Fig. 2. Proposed linear micropipeline architecture.

II. MICROPIPELINE PROJECT EMPLOY-ING FPGAS:

Programmable devices, such as FPGAs, are designed for synchronous designs [2]. Efforts for prototyping asyn-chronous design in commercial FPGAs [14], [15] and academic FPGAs [16], [17] been reported recently. The problems in implementing asynchronous systems in com-mercial FPGAs are related to the control. They are:a) Process of mapping risk-free boolean functions in logic blocks (macrocells). The commercial tools used for de-composition and mapping boolean functions in LUTs are not prepared to meet the requirements of logical hazard. The micropipeline style has as main feature, the simplifi-cation of the pipeline control, an important characteristic for asynchronous systems implemented in FPGAs.

The control is either distributed between stages or centralized, and it is responsible for the communication between the pipeline stages. In a FPGA platform, the control must be distributed in order to avoid hazard problems, as previ-ously mentioned. The pipeline communication employs the handshake protocol with Request and Acknowledge signals [4], [5]. The communication between the stages can be performed in twodifferent protocols: 4-phase or 2-phase. Fig. 3 shows the behavior of these protocols. This may cause a circuit malfunction, if manual interven-tion to fix the problem is not performed. The mapping function must satisfy the decomposition requirements proposed in Sigel et al. [18]. b) Internal routing process among macrocells can introduce significant delays. These delays can result in essential hazard and lead to the circuit malfunction [4], [5]. The circuit delay model defines how to solve the problem of essential hazard: the insertion of delay elements in the feedback lines or the employment of macrocells that satisfy the isochronic fork condition [4], [5]. The micropipeline architecture can be linear or nonlinear. [19]. In this paper we focus only on the linear architecture. This architecture doesnot need delayelements and timing analysis, moreover, it ismore robust. The proposed mi-cropipeline architecture operates in 2- phase handshake protocol and, therefore, the processing request occurs on both edges of this signal. Fig. 4 shows the general lay-out of a stage, which consists of a flip-flop based regis-ter and control. The control was specified in STG (signal transition graph) [20] (see Fig. 5). It is composed by the signals: Ri (request input), Ao (acknowledge output), Ai (acknowledge input), L (load) and Ro (output request). Fig. 6 shows the circuit of the logic control.
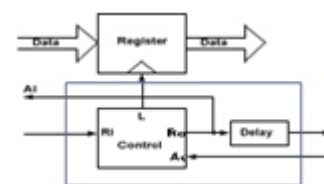

Fig. 3. Linear micropipeline target for FPGA

Generation of the pipeline data-path employing single-rail components and the pipeline synthesis according to [21]. This step generates the synchronous pipeline. (see Fig. 7). 3) Execution of the desynchronization process; detection of the critical path of each pipeline step and calculation ofthe respective delay element. 4) Change the clock sig-nal by the proposed asynchronous pipeline controller (see Fig. 8).5) Synthesize the asynchronous pipeline generated in step 4.
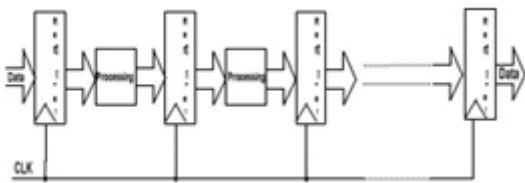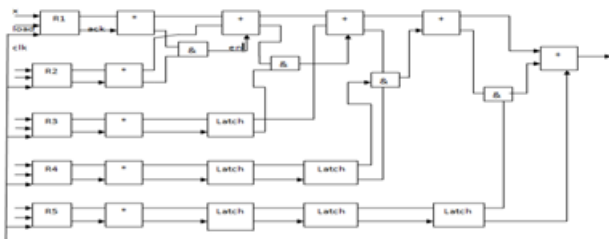

Fig. 4. Asynchronous pipeline architecture

**Fig.5.proposed figure for fir filter.**

VI. EXPERIMENTAL RESULTS:

A comparison was made with the MOUSETRAP pipeline architecture in order to evaluate the performance of the proposed architecture. The simulations were performed employing the Xilinx software, version 14.2 , considering an Spartan 6 as target de-vice. Fig. 11 and 12 show the simulations of the mousetrap control and the stages of the FIR filter. Fig. 13 and 14 show the simulations of the proposed pipeline: control and FIR stages.

ARRAY MULTIPLIER
asynchronous pipeline
Area report

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slices	464	768	60%
Number of Slice Flip Flops	251	1536	16%
Number of 4 input LUTs	808	1536	52%
Number of bonded IOBs	103	124	83%
Number of GCLKs	3	8	37%

Delay report
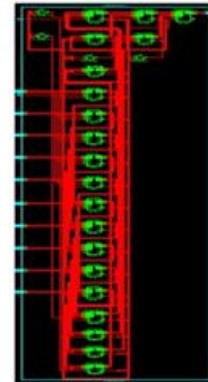
Block diagram for Array asynchronous pipeline

BOOTH MUL USING ASYNCHRONOUS PIPELINE
Area report

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slices	416	768	54%
Number of Slice Flip Flops	241	1536	15%
Number of 4 input LUTs	749	1536	48%
Number of bonded IOBs	99	124	79%
Number of GCLKs	3	8	37%

```
Timing Summary:
----------------
Speed Grade: -5

    Minimum period: 14.994ns (Maximum Frequency: 66.693MHz)
    Minimum input arrival time before clock: 18.741ns
    Maximum output required time after clock: 6.141ns
```
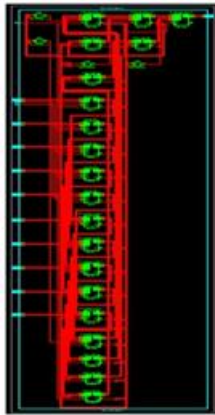
RTL SCHEMATIC

Fig.5. Implementation results for Existing and Proposed methods

VII. CONCLUSION:

The linear micro pipeline style of the asynchronous par-adigm is advantageous because the performance is in-creased and the control is simplified. In this paper, we proposed a new pipeline architecture oriented for FPGAs implementations. The control employs only two LUTs, and the registers used are based on flip-flops, allowing a better distribution of macrocells in a FPGA. The imple-mentation of the proposed FPGA control is free of essen-tial hazard. Through a case study, we show that our archi-tecture has a better performance than the MOUSETRAP architecture, considering the FPGA implementation.

REFERENCES:

[1]D. Goldhaber-Gordon, et al., "Overview of Nanoelec-tronic Devices," Proc. of the IEEE, vol. 85, No. 4, pp.521-540, April 1997.

[2]J. J. Rodriguez, et. Al., "Features, Design Tools, and Applications Domains of FPGAs", IEEE Trans. on Indus-trial Electronics, vol. 54, No. 4, pp.1810-1823, August 2007.

[3]J. Cortadella, et al., "Coping with the variability of combinational logic delays," ICCD, pages 505–508, 2004.

[4] C. J., Myers, "Asynchronous Circuit Design", Wiley & Sons, Inc., 2004, 2a edition.

[5] J. Sparsø e S. Furber, "Principles of Asynchronous Circuits Design", Kluwer Academic Publishers, 2001.

[6]I. E. Sutherland, "Micropipelines", Communication of the ACM, vol. 32, No.6, pp.720-738, June, 1989.

[7]S. B. Furber et al., "Four-Phase Micropipeline Latch Control Circuits," IEEE Trans. on VLSI Systems, vol.4, no. 2, pp.247-253, June, 1996.

[8]G. S. Taylor and G. M. Blair, "Reduced Complexity Two-Phase Micropipeline Latch Controller," IEEE Jour-nal of Solid-State Circuits, vol. 33, Nro. 10, pp.1590-1593, October, 1998.

[9]M. Singh and S. M. Nowick, "The Design of High-Performance Dynamic Asynchronous Pipelines: Looka-head Style," IEEE Trans. on VLSI Systems, vol.15, no. 11, pp.1256-1269, November, 2007.

[10] M. Singh and S. M. Nowick, "The Design of High-Performance Dynamic Asynchronous Pipelines High-Capacity Style", IEEE Trans. on VLSI Systems, vol.15, no.11, ´pp.1270-1283,November, 2007.