A Simple Procedure for High Performance VLSI Registers Design

Duarte Lopes de Oliveira Sandro Shoiti Sato Osamu Saotome
duarte@ita.br shoiti@ita.br osaotome@ita.br
Divisão de Engenharia Eletrônica do Instituto Tecnológico de Aeronáutica – IEEA – ITA
Praça Marechal Eduardo Gomes, 50 – CEP 12228-900 – São José dos Campos– São Paulo – Brazil.

Abstract
Register is an important part in a digital system and it influences strongly the system clock rate. To synthesize K operations registers, is used either the traditional technique of iterative nets or the technique based on functional units. The iterative nets technique gets better results, but the synthesized cells present several delay levels, degrading the clock rate. In this work, a new technique to synthesize cells to K operations registers is proposed. This technique presents a better clock rate when compared with the classical one. This new technique makes use of a new flip-flop (FF), called functional FF, in cells synthesis. They allow joining the combinatory logic block with the store element.

Keywords: registers, iterative nets, non-conventional flip-flops and cell synthesis.

1. Introduction
The increasing number of digital systems applications, features such as high clock rates and smaller power consumption, are important recurrent themes [7]. In a digital system, registers take an important role. To synthesize registers of N bits with K operations is used either functional units technique[1] or interactive nets technique [7,12]. Functional units techniques has as a strong point its simplicity, but the synthesized circuits present low performance in terms of cycle time and the area occupation (in terms of number of transistors) is sub-optimized [6]. Interactive nets technique partitions the register of N bits with K operations in one or more bits cells. These cells are synthesized though classical combinatorial technique [12]. Iterative net implemented with classical combinatory technique synthesizes only the logic block of register cell (see Figure 1) [12]. The synthesis is performed extracting the truth table from register's table of operation, and obtaining the minimized excitation equations (design of combinatory block – see Figure 1). In spite of a better performance of resulting circuits, this technique presents a severe limitation, due to the generation of cells (flip-flop, plus block) with several delay levels, degrading the performance of cycle time (see equation 1).

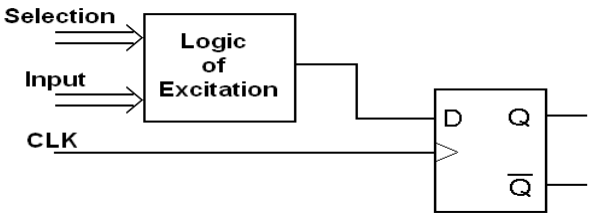

Figure 1 – Synchronous technical: basic cell

$$T_{Cycle} \geq T_{MAX\text{-}Propagation\text{-}Block} + T_{SETUP\text{-}FF} + T_{MAX\text{-}Propagation\text{-}FF} \quad (1)$$

In this article, we propose an **architecture to K operations registers cells**, and a **simple synthesis procedure**, based on iterative nets technique. Our architecture is a non-conventional flip-flop, based on two lathes structure (master-slave) [13]. Our technique allows cells generation with a better performance in terms of cycle time, when compared with the classical technique of iterative nets, but presents a small area penalty.
The noticeable advantage of our technique is that the designer can synthesize the non-conventional flip-flop, without knowledge of asynchronous synthesis [1,3,4].
Our method generates VLSI cells, implemented by basic ports (standard cells technology) or with full custon cells.

The remaining of this article is organized as follows. In section 2, functional Flip-Flop is presented; in section 3, the synthesis procedure of basic cell is presented; in section 4, examples of cell synthesis are shown; and in section 5, the performances of the proposed circuits are analyzed.

2. Basic cell architecture
Proposed architecture, called functional Flip-Flop (FF-MS F) is shown in Figure 2. FF-MS F is based on master-slave model [11,12]. Master F latch is triggered by high level of the clock signal and D latch by low level. The state transition graphs of D an F latches are shown in Figures 3a, b, respectively. Figure 4 shows full-custom D latch.

[1] It is also known as the technique that works with components MSI (medium scale integration) and additional logic.

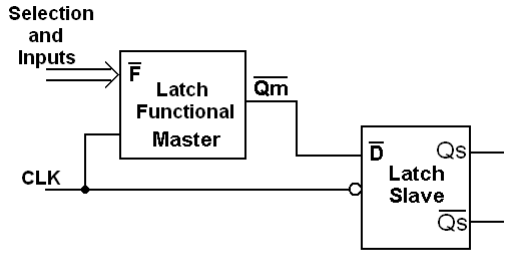

Figure 2 – Target architecture: master-slave functional Flip-Flop (FF-MS F).

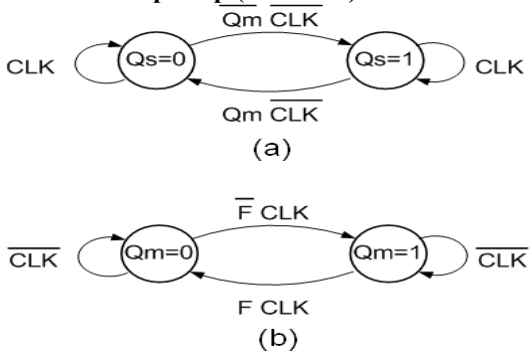

Figure 3 – STGs: a) D latch; b) F latch.

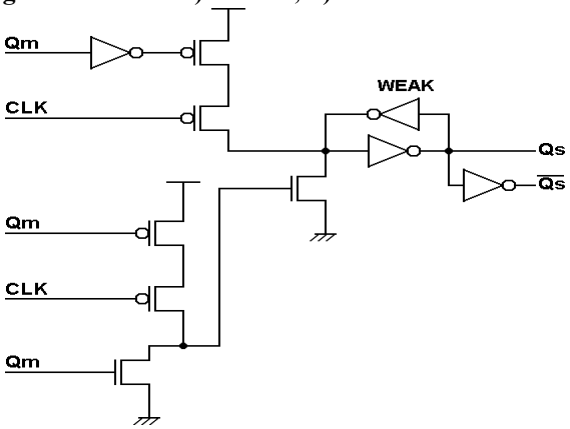

Figure 4 – Latch D: full custom circuit.

Figure 5 shows the logical circuit of F latch. Figure 6 shows a generalization of F latch, implemented with basic gates. In [4] shows the functional FF architecture that operates at both edges of clock signal (FF-DET F). This FF is used to design higher speed registers or when a clock rate reduction is desirable, to reduce power consumption [4,8]. To registers with operations, that obey fundamental mode, only F latch is used. Fundamental mode establishes that input signal change of the next state transition will occur only when the circuit is stabilized (without electric activity in lines and gates) [3].

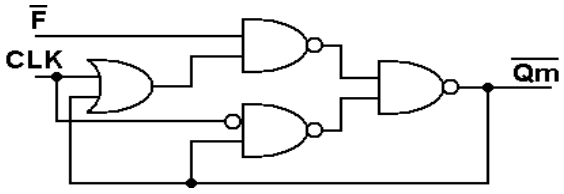

Figure 5 – Latch F: basic gates circuit.

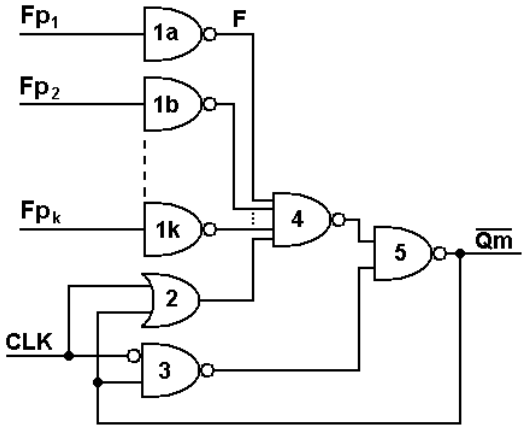

Figure 6 – Latch F: generalized basic gates circuit.

2.1 Timing analysis.

Setup and hold times are two important time parameters, for memory elements: setup time is the minimum time interval between stabilization of input signals and the clock signal transition. Hold time is the minimum time interval between clock signal transition and change of input signals for next state transition [2,10].

Figure 7 shows timing conditions for external environment and FF-MS F interaction, where we obtained the following time equations. The equations are obtained from FF-MS F shown in Figure 8, and T represents gate delay.

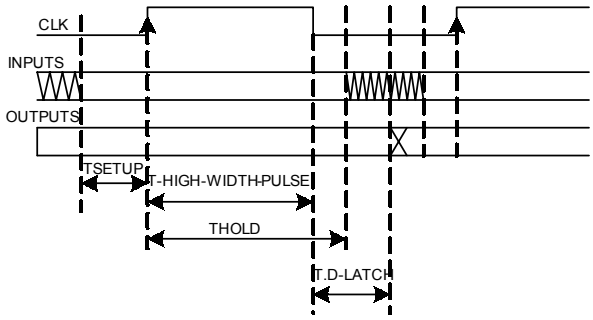

Figure 7 – Timing: FF-MS F

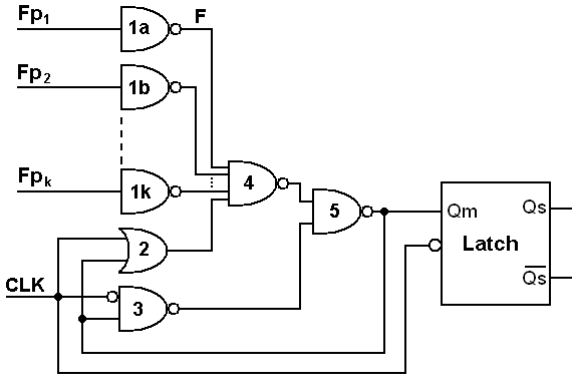

Figure 8 – FF-MS F: logic circuit

Minimum setup time of FF F is*:*

$$T_{SETUP} \geq T_{MAX\text{-}PRODUCT\text{-}1a} - T_{MAX\text{-}OR\text{-}2} \quad (2)$$

Minimum width of clock "high" is:

$$T_{MIN\text{-}HIGH\text{-}WIDTH\text{-}PULSE} \geq T_{MAX\text{-}NAND\text{-}4} + T_{MAX\text{-}NAND\text{-}5} \quad (3)$$

Minimum hold time of FF F is:

$$T_{HOLD} \geq T_{MIN\text{-}HIGH\text{-}WIDTH\text{-}PULSE} + T_{MAX\text{-}OR\text{-}2} - T_{MIN\text{-}PRODUCT\text{-}1A} \quad (4)$$

Maximum propagation time of D Latch is:

$$T_{MAX\text{-}LATCH\text{-}D} \quad (5)$$

Maximum propagation time of FF F is:

$$T_{MAX\text{-}PROPAGATION\_FF} \geq T_{HIGH\text{-}WIDTH\text{-}PULSE} + T_{MAX\text{-}LATCH\text{-}D} \quad (6)$$

Minimum width of clock "low" is:

$$T_{MIN\text{-}LOW\text{-}WIDTH\text{-}PULSE} \geq T_{MAX\text{-}LATCH\text{-}D} + T_{SETUP} \quad (7)$$

Minimum cycle time is:

$$T_{CYCLE} \geq T_{MIN\text{-}LOW\text{-}WIDTH\text{-}PULSE} + T_{MIN\text{-}HIGH\text{-}WIDTH\text{-}PULSE} \quad (8)$$

Maximum clock frequency is:

$$F_{MAX} \leq 1/T_{CYCLE} \quad (9)$$

3. Synthesis procedure

Interactive net makes use of one bi-dimensional cell to N bits, K operations registers. In case to cells to registers, five types of signals are expected, that are: 1) data inputs; 2) selection inputs; 3) stored data output; 4) status inputs; 5) status outputs.

Status inputs and outputs of a cell transport information from the previous cell to the next cell, respectively.

Our method is composed by four steps:

1. Defining the register's operational table (TO).
2. Defining selection codes to each register operation [14][2].
3. Using TO codified, to obtain minimized[3] function F of two levels [12].
4. Transforming (if adequate) two-level F function to a multi-level function and insert it in FF-MS F as full custom or basic gates.

Step 4 of our method allows two variations: 1) the operations satisfy hold time (fundamental mode operation), therefore, design using only F latch; 2) the register will operate in both edges of clock signal, therefore design using FF-DET F [4].

4. Examples

Our method will be applied to synthesize FF-MS F, for two examples [5,9].

A Shift Register

Steps 1 and 2 define the operation table with selection codes. Figures 9 and 10 shows operation table and the basic cell structure of the shift register. Steps 3 and 4 extracts two-level minimized function F (equation 10). Figure11 shows the logic circuit of F latch, with function F inserted, respectively.

CLK	Dir	Load	Q_i
⊓	0	0	Q_{i-1}
⊓	1	0	Q_{i+1}
⊓	X	1	D_i

Figure 9 – Shift register operation table

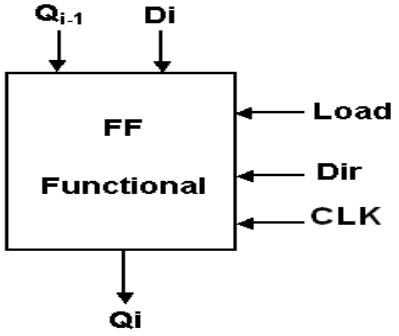

Figure 10 – Structure: basic cell

$$Fi = Load\ D_i + Load'\ D_{ir}'\ Q_{i-1} + Load'\ D_{ir}\ Q_{i+1} \quad (10)$$

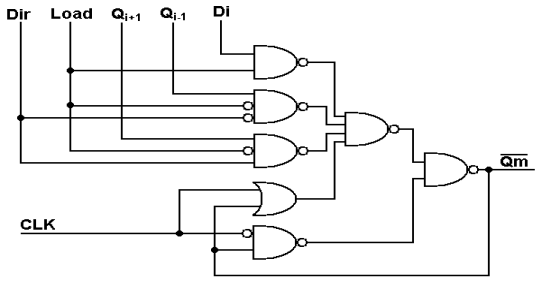

Figure 11 – Latch F: basic gates

B Accumulate Register

Steps 1 and 2 define operation table with selection code. Figures 12 and 13 shows operation table and the basic structure cell. Steps 3 and 4 extract minimized function F, translated to XOR gates (equation 11). Figure 14 shows the logic circuit of F latch with function F inserted, respectively.

CLK	S	Operation
⊓	0	$R_{ACC} \leftarrow R_{ACC} + Date$
⊓	1	$R_{ACC} \leftarrow Date$

Figure 12 – Shift register operation table

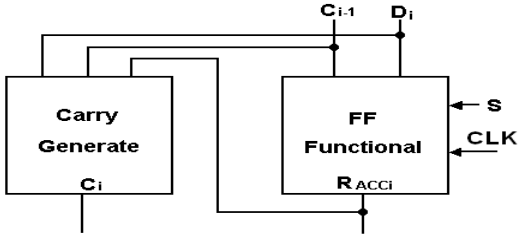

Figure 13 – Structure: basic cell.

[2] One step of our method is the selection variables coding. De Micheli [14] shows that the choice of selection variables coding contributes to Boolean equations simplification.

[3] Any minimization algorithm can be used, for example, Karnaugh map.

$$Fi=(D_i' \bullet R_{aci} \bullet C_{i-1}' + D_i \bullet R_{aci}' \bullet C_{i-1}' + D_i \bullet R_{aci} \bullet C_{i-1} + D_i' \bullet R_{aci}' \bullet C_{i-1}) \bullet S' + D_i \bullet S$$

$$Fi=(R_{aci} \bullet (D_i \oplus C_{i-1})' + R_{aci}' \bullet (Di \oplus C_{i-1})) \bullet S' + D_i S$$

$$Fi=(R_{aci} \oplus D_i \oplus C_{i-1}) \bullet S' + D_i \bullet S \qquad (11)$$

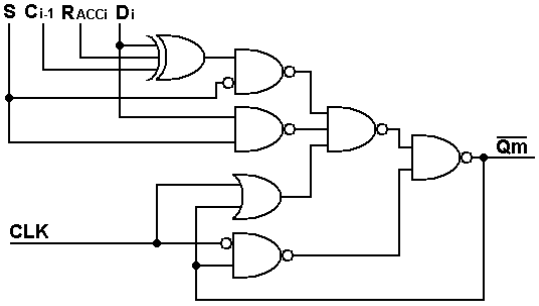

Figure 14 – Latch F: basic gates

5. Discussion and Results

In this section we compare our results using CMOS technology standard-cell library of the IMEC-96 of 0.7μm. The comparison is made in terms of number of transistors and cycle time. Table 1 shows the results of classical synthesis and our method, for 3 registers from literature. Our full custom D latch presented an estimated approximated propagation time of 1ns. Table 1 show that our method presented an averaged approximate reduction of 31% for cycle time and an averaged approximate penalty in transistor numbers, of 5%. For the example of counter register, we obtained the best results. Cycle time reduction attained was 33%. Figure 15 shows PSPICE simulation of our shift register cell that satisfies the specification for a set of stimuli.

	Number of Operations	Classic technique		Our method	
		Number of Transistor	Time of Cycle (ns)	Number of Transistor	Time of Cycle (ns)
Shift Register	3	54	4,48	57	3,03
Accumulate	2	66	4,86	69	3,38
Program Counter	4	102	5,12	108	3,44
Total	——	222	14,46	234	9,85

Table 1 – Results: area and cycle time

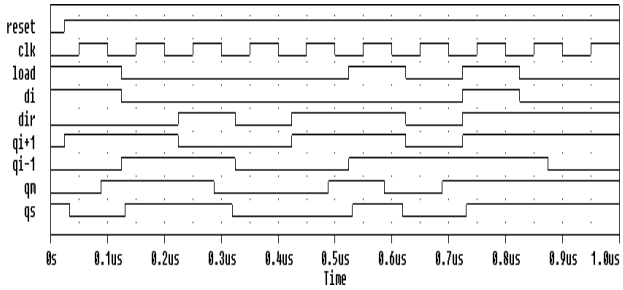

Figure 15 – Simulation

6. Conclusion

We presented in this article a new method to register synthesis. The two existing techniques from literature (functional units and classical combinatorial) lead to a low performance in clock rate. New approach is derived from a new flip-flop design, called F FF that generates registers with better performance for clock rates, under a small are a penalty. To future works, our method will be applied to low power consumption register synthesis, without loss in clock rate and area performances.

References

[1] S. H. Unger, "Double-Edge-Triggered Flip-Flops", IEEE Trans. on Computers, vol. C-30, No. 6, pp.447-451, June 1981.

[2] S. H. Unger, "Clocking Schemes for High-Speed Digital Systems", IEEE Trans. on Computers, vol. C-35, No. 10, pp.880-895, October, 1986.

[3] D. L. Oliveira, et al. "Automatic Synthesis of Non-Conventional Flip-Flops", XII Worshop Iberchip, (mídia eletrônica) San José, Costa Rica, 2006.

[4] D. L. Oliveira et al. "Synthesis of Low Power Synchronous Finite State Machine using Latches", Relatório Técnico – Instituto Tecnológico de Aeronáutica –ITA – Divisão de Eng. Eletrônica - 2006.

[5] H. Hsich, F. Balarin, L. Lavagno and A. Sangiovanni-Vincentelli, "Synchronous approach to the equivalence of embedded system implementations," IEEE on Trans. CAD of Integrated Circuits and Systems, vol.20, No.8, pp. 1016-1033, August 2001.

[6] J. O. Hambien and M. D. Furman, Rapid Prototyping of Digital Systems a Tutorial Approach, 2Ed., 2002.

[7] M. Pedram and ª Abdollahi, Low-power RT-level synthesis techniques: a tutorial. IEE Proc. Comp. Digit. Tech., Vol.152, No.3, pp.333-343, May 2005.

[8] Cao Cao and Bengt Oelmann, Mixed Synchronous/ Asynchronous State Memory for Low Power FSM Design, Proc. Euromicro Systems on Digital System Design, pp.363-370 2004.

[9] A. Hertwig and H.-J. Wunderlich, Fast Controllers for Data Dominated Applications. Proc. European Design and Test Conference, pp.84-89, 1997.

[10] F. Champernowne, et al., Latch-to-Lach Timing Rules. IEEE Trans. on Computers, vol.39, No.6, pp.798-808, June 1990.

[11] K. A. Sakallah, et al., Analysis and Design of Latch-Controlled Synchronous Digital Circuits. IEEE Trans. CAD, vol.11, No.3, pp.322-333, March 1992.

[12] R. H. Katz, Contemporary Logic Design, The Benjamin/ Cummings Publishing Company, Inc., 2ª edition 2003.

[13] C. E. Stroud, "Automated Bist for Sequential Logic Synthesis,"IEEE Design & Test of Computers, pp.22-32, December 1998.

[14] G. De Micheli, Synthesis and Optimization of Digital Circuits, McCraw-Hill Int. Editions, 1994.