Design of Dynamic Asynchronous Flip-Flops and Counters based on Dynamic STG

Prof. Christian Piguet

CSEM, Neuchâtel, Switzerland
& LAP, EPFL, Lausanne, Switzerland

Abstract.
This paper presents a synthesis method for generating true single-clock dynamic CMOS flip-flops. Such dynamic circuits are used today for very high performance microprocessors and ASICs, however with some supplementary keeper transistors. Many references have presented dynamic single-phase clock latches and flip-flops, however without a corresponding synthesis method. The goal of this paper is to present “asynchronous” synthesis methods based on STG that are commonly used for asynchronous static circuits, while modifying these known methods for the generation of dynamic circuits. In this way, dynamic single-phase clock design and asynchronous STG-based methods are unified. The second goal of this paper is to be able to analyze and to synthesize hazard-free dynamic circuits. It is known that some versions of latches and flip-flops presented in literature may present some hazards, races or glitches. The presented design method, based on the properties of negative gates, can produce dynamic circuits that are hazard-free. Furthermore, it helps to analyze existing circuits to detect hazards and glitches, while providing other structures that can be more convenient to specific applications. The presented method is not limited to flip-flops, but can be used for latches and any dynamic sequential circuit described by a STG or a flow table.

Motivation
Very fast circuits are designed today to satisfy speed performances that increase dramatically, while trying to limit as much as possible the power consumption. Microprocessors up to 1.5 GHz are reported in the literature. As dynamic circuits are known to provide 20 to 30% speed improvements over static circuits, they are generally used in very high performance microprocessors, at least on the critical paths of most logic blocks.

The basic principle of dynamic circuits, such as flip-flops, latches or frequency dividers, is the following. Such cells have to store information. This is not done by cross-coupled gates, as is the case in static logic, but parasitic capacitances are used to dynamically store the information. Such a logic style reduces the number of transistors, resulting in very fast cells.

Such dynamic cells have been proposed by many authors, however without the description of the corresponding synthesis method. This paper presents an “asynchronous” method to design dynamic cells from Signal Transition Graphs or flow tables. It provides logical structures that minimize the number of logical gates. Races or critical races can also be avoided, while minimizing the number of switching variables between stable states of the dynamic cells. Furthermore, this method can be used to analyze the dynamic cells that have been proposed to detect races, critical races or glitches. Such a method has been presented in for static circuits.

The presented “asynchronous” method basically synthesizes a static circuit and then modifies the “static” STG to get a dynamic circuit. Obviously, one can think that it would be possible to start directly from a “dynamic” STG to get a dynamic circuit. Such a method has been used for latches but is not presented here due to lack of space.