Asynchronous Digital Design for Sub-threshold Regime Operation

S. Koumousi, M. Pons Solé, M.-N. Morgan, J.-L. Nagel, S. Emery

Reducing the supply voltage down to the sub-threshold regime allows a significant reduction of power consumption. However, synchronous sub-threshold designs are very sensitive to even small variations in process, voltage and temperature, which can either lead to incorrect functionality or degrade the performance due to huge design margins. On the other hand, asynchronous designs based on NULL Convention Logic (NCL) are self-timed designs that are insensitive to variations and thus require no design margins.

Asynchronous circuits can be classified into different categories depending on their degree of asynchronicity, from locally-synchronous to delay-insensitive. The latter category is extremely interesting for sub-threshold designs, which are very sensitive to both global and local variations, and also greatly simplifies design automation compared to other asynchronous approaches. However, designing delay-insensitive circuits usually comes at a cost, as they involve a large logic overhead e.g. due to completion detection.

In this work, NULL Convention Logic (NCL) was chosen among existing delay-insensitive asynchronous methodologies. Transforming a synchronous RTL design into its clockless NCL counterpart was done using UNCLE [1], an academic set of scripts partly relying on conventional logic synthesizers to map synchronous RTL to a netlist of generic functions and, subsequently, into an NCL netlist. More information on the design flow can be found in [2].

The TSMC 65 nm LP process was chosen for these experiments. An NCL sub-threshold standard cell library of 23 hysteresis state-holding gates was developed and used with UNCLE to generate an NCL-mapped netlist, which was subsequently placed-and-routed using standard tools. This step of the NCL flow required particular optimization, as an asynchronous design intrinsically contains many combinational loops when seen from a static timing engine, and prevent the tool from applying optimizations such as buffering and gate sizing. The focus was set on automatizing this loop breaking at specific locations and guaranteeing that all paths would get optimized. Finally, the netlist extracted from layout could be simulated at transistor level.

A 3-stage pipeline design was coded in synchronous RTL and used as a test case: the first stage consists of a 5-tap FIR filter; the second stage of a multiplication of the FIR output with an external control signal; the last stage of the addition of the upper and lower bits of the second stage output. The same functional verification was applied both to the synchronous RTL and to the NCL asynchronous transistor netlist to verify correct functionality of the layout.

The same RTL was physically implemented as a synchronous version (using an existing sub-threshold standard cell library) in order to compare the speed, power and sensitivity to variations of both synchronous and asynchronous versions.

Table 1 shows that the synchronous design was able to reach an operating frequency f_{CLK} of 81.3 kHz at 0.3 V in typical corner (TT), while the asynchronous design reaches 3x this speed. The asynchronous speed degrades 30x in slow (SS) corner at -25°C, but improves 10x in fast corner (FF) at 75°C (the synchronous design is limited to 81.3 kHz for all corners in the absence of process sensing). However, the asynchronous NCL design typically has 1.7x the area of the synchronous version and, in nominal conditions (TT process corner, Vdd = 0.3 V, 25ºC), the asynchronous version also exhibits 3x more energy consumption than its synchronous counterpart. Note that the asynchronous design functional validity was simulated down to 0.2 V.

Table 1: NCL vs. synchronous design performance.

			Asynchronous		Synchronous	
Process corner	Temp	Voltage	Average freq.	Power	Frequency	Power
SS	-25 C	0.3 V	6.25-8 kHz	4.15-4.33 nW	81.3 kHz	3-13 nW
TT	25 C	0.3 V	230-250 kHz	185-190 nW	81.3 kHz	20-38 nW
FF	75 C	0.3 V	2.38 MHz	3.8 uW	81.3 kHz	1.27-1.41 uW
Area			11.16 um2		6.38 um2	

The asynchronous design automatically adapts to process, voltage and temperature variation and the computation delay is naturally data dependent. Table 2 summarizes the effect of operating condition variation on speed and power consumption with respect to nominal.

Table 2: Influence of operating conditions on speed and power.

	Process corner			Temperature			Voltage			Data
	SS	TT	FF	-25C	25C	75C	0.2V	0.3V	1V	
Frequency	-5x	1	5x	-5x	1	2.8x	-12x	1	100x	±18%
Power	-6x	1	6x	-4.9x	1	3.8x	-8.2x	1	1100x	

Monte Carlo simulations at the worst case corner (SS, -25°C and 0.3 V) and at the maximal frequency of operation of 81.3 kHz showed that the synchronous design sometimes failed to operate correctly, whereas the NCL did not produce erroneous outputs. This demonstrates that additional costly design margins should be definitely added to the synchronous version to cope for the local variations and that they are not required for NCL design.

In conclusion, though the selected delay insensitive asynchronous design flow definitely proved to be robust to operating condition variations and to local mismatch, the logic overhead compared to a synchronous design mostly cancels the benefit of reducing design margins in terms of power consumption. Nevertheless this approach might still be interesting for systems running on very low-voltage and intermittent supplies (e.g. small solar cells), where a complex power management unit could be advantageously replaced by a self-timed, clockless digital approach.

[1] R. B. Reese, *et al*., "Uncle - An RTL approach to asynchronous design", Proc. ASYNC The Int. Symp. on Asynchronous Circuits and Systems (2012).

[2] S. Koumousi, "Asynchronous digital design for sub-vth operation", MSc thesis.