Clock Distribution in RNS-based VLSI Systems

DANIEL GONZÁLEZ [1], ANTONIO GARCÍA [1], GRAHAM A. JULLIEN [2],
JAVIER RAMÍREZ [1], LUIS PARRILLA [1] AND ANTONIO LLORIS [1]

[1] Dpto. Electrónica y Tecnología de Computadores
Universidad de Granada
18071 Granada
SPAIN

[2] ATIPS Laboratory, Dept. of Electrical and Computer Engineering
University of Calgary
Calgary, Alberta T2N 1N4
CANADA

Abstract: Clock distribution networks synchronize the flow of data in digital systems, and the features of the clock signal affect system performance and reliability. The great advances in integration levels and speed requirements in actual digital systems lead to enormous complexity in synchronization. Differences in delay of clock signals along clock paths, due to different lengths or active elements such buffers, cause loss of performance and poor reliability. However, a perfect synchronization leads to a simultaneously triggering of great amount of devices and, thus, large current demands. This paper presents a new scheme for clock distribution in VLSI systems that avoids both skew and large current demand related problems. Different skewed clock signals synchronize different independent channels in a RNS-based system. This new strategy has been simulated over a three-stage decimation CIC filter, resulting in a diminution of current spikes and current change rate with no significant increase of power consumption.

Key-Words: Clock, Clock Distribution, RNS, Synchronization.

1 Introduction

The last advances in integrated circuit fabrication have lead to increasing integration levels and operating speeds. These factors make extraordinarily difficult the proper synchronization of integrated systems because the length of clock distribution lines increases along with the number of devices the clock signal has to supply, thus leading to substantial delays that limit system speed. For TSPC (True Single Phase Clock) one clock line must be distributed all over the chip, as well as within each operating block. More complex clocking schemes may even require two or four non-overlapping clock signals [1], thus increasing the resources required for circuit synchronization. Moreover, for clock frequencies over 500 MHz, phase differences between the clock signal at different locations of the chip (skew) start presenting serious problems [2]. Several techniques have been developed for overriding clock skew, being the most common RC-tree analysis and H-tree distribution. The first represents the circuit as a tree, modelling every line through a resistor and a capacitor and every block as a terminal capacitance [3]. In this way, distribution line delay is evaluated and elements to compensate clock skew can be subsequently added. The second option distributes symmetrically the different clock signals forming and "H", thus avoiding the appearance of differences in the delay of each path, as illustrated in Fig. 1. However, minimizing skew has negative sides, since the simultaneous triggering of many devices leads to short but large current demands. Because of this, a meticulous design of power supply lines and device sizes is required, with this large current demand resulting in area penalties. If this is not the case, parts of the chip may not receive as much energy as required. This approximation to the problems related to fully synchronous circuits and clock skew has been previously discussed [4]. This paper will present an alternative for efficiently synchronizing RNS-based circuits while keeping current demand to a minimum. The underlying idea is to generate non-overlapping clock signals, each one controlling an RNS channel.

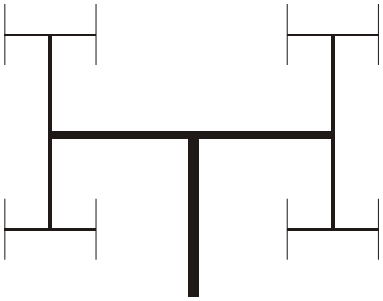

Fig. 1. H-tree distribution

This takes advantage of the non-communicating channel structure that characterizes RNS architectures in order to reduce the clock synchronization requirements for high-performance digital signal processing systems. A three-stage CIC (Cascade Integrator Comb) decimation filter [5-6] was used for evaluation of the presented strategy.

2 Skew

When clock propagates across different paths, it suffers different delays due to different path lengths and active elements. The difference in the value of the clock in two nodes is known as clock skew. Clock skew produces system performance losses when compared to that obtainable from individual blocks, since it is necessary to guarantee the proper function of the chip with a reduced speed clock. Skew will cause clock distribution problems if the following inequality holds [7]:

$$\frac{D}{v} > \frac{k}{f_{app}} \qquad (1)$$

where k<0.20 (typical value) is a constant, D is the typical size of the system, v is the propagation speed for the clock signal and f_{app} is the applied clock frequency. Existing solutions for clock skew provide two different approaches to the problem; the first one consists in equalizing the length of clock paths to processing elements with buffers and delay elements, or through H-tree, mesh or X-tree topologies [3, 8-10]. The other option eliminates or minimizes the skew caused by variations during chip fabrication [11-12] processes.

Typically, synchronous systems consist of a chain of registers separated by combinational logic that performs data processing. The maximum clock frequency is derived from:

$$f_{\max}^{-1} \geq T_{PD} + T_{skew} \qquad (2)$$

where f_{max} is the maximum frequency obtainable, T_{PD} is the time between the arrival of the clock signal at the i-th register and stable processed data at the output of the (i+1)-th register and T_{skew} is the time between the arrival of the clock signal at the i-th register and the arrival of the same signal at the (i+1)-th register.

Clock skew can be considered as either positive or negative, although the sign criteria is not standardized. If positive, then from equation (2), the minimum system clock period is increased, while if negative, $T_{\min}$ decreases. This appears as an advantage, and can be used to improve the system features, but an excessive positive skew results in a race-related problems may arise if data processing time is lower that the skew.

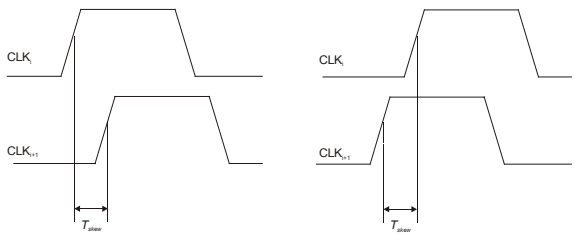

Fig. 2. Negative (left) and positive (right) skew.

3 New synchronization Strategy

The proposed strategy for synchronization of RNS-based systems introduces the generation of several signal clocks from the master clock. These clocks signals are generated slightly out-of-phase with non-overlapping edges. Each one of these clock signals synchronizes one of the RNS channels, while the master clock synchronizes global data (mainly at the global inputs and outputs of the system) and guarantees data coherency. Thus, each channel process data at different time instants, so the current demand is distributed over the whole clock cycle. This has the effect of reducing current spikes on the power supply lines approximately by a factor that is the number of generated clock signals. The phase difference between the generated clocks has to satisfy some specifications:

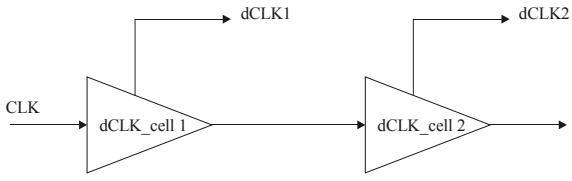

Fig. 3. `dCLK_cell` chain for clock generation

- the number of clock signals with overlapping active cycles has to be minimized, as well as the time two or more of these active cycles overlap,
- clock edges must not coincide,
- data coherency has to be respected at both the input and output of the system.

Clear advantages are obtained when these requisites are satisfied, since current spikes are reduced and power dissipation is distributed over the master clock cycle, rather than concentrated around the master clock edges. The temporal variation in current values is reduced also. Moreover, power supply lines may be scaled and clock distribution resources reduced, thus simplifying the chip design effort.

This strategy can be used in systems whose operation mode implicates several non-communicating modules or channels, as those based on the RNS. RNS [13-14], with non-communicating channels, perfectly suits this clocking scheme and a generated clock signal is applied to each independent channel, while the master clock signal used to generate these other clocks can also synchronize the global input and output. Moreover, the resources required for implementing this new strategy are minimum and a few transistors, basically three inverters, are required for each channel. Specifically, the master clock signal is routed through an inverter chain, thus being delayed. Meanwhile, the generated clock signals are extracted at adequate points of this chain and conditioned to be used as clock signal for a complete RNS channel. This scheme requires the inverter chain to alternate large and small input capacitances and low driving capabilities, so appropriate delays can be generated. This structure generates low-quality clock signals within the inverter chain, so additional buffers are required for obtaining adequate clock signals. Fig. 3 illustrates the hardware required to generate the proposed synchronization scheme, where `dCLK` stands for generated delayed clocks, while Fig. 4 shows the detailed scheme for the so-called `dCLK_cell` cell, which consists of three inverters. It can be deduced from Fig. 4 that three design parameters, L_d, W_b and L_b, are available in order to obtain the system specifications, while L_{min} represents the feature size of the fabrication process and W_{min} the minimum usual width for pMOS transistors. Connecting `CHout` pads to `CHin` pads, the inverter chain described above is built, while the master global clock is used as input to this chain. Fig. 4 illustrates how large capacitance inverters are alternated with minimum-size devices. Thus, the low driving capabilities of the latter allow modelling of the required delay using the L_d parameter. Meanwhile, the generated clock signal `dCLK` is regenerated by a third inverter that includes the parameters W_b and L_b. These allow matching of the timing specifications for a given system, also allowing the adaptation of the cell to the overall capacitance to be driven by the generated clock. However, these three design parameters L_d, W_b and L_b are not independent, and their relation needs a careful study of the final

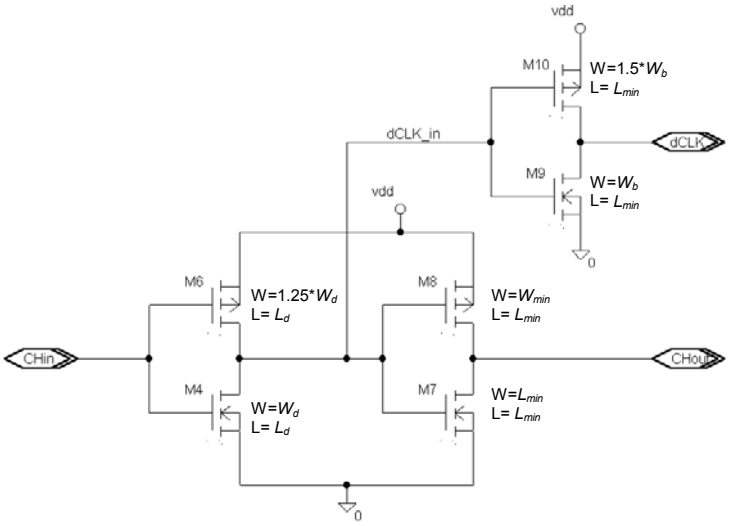

Fig. 4. `dCLK_cell` schematic.

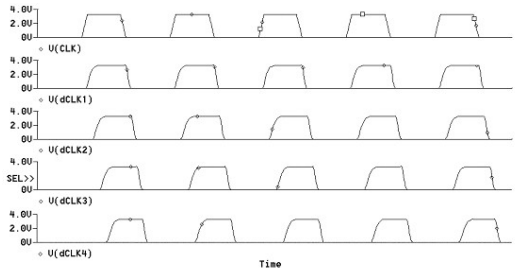

Fig. 5. Clock signals generated for the new strategy.

system to be synchronized in order to select their optimum values. Fig. 5 shows the resulting generated clocks in a simple design example for a 300 MHz master global clock and 0.1 pF loads for every `dCLK` signal. Note that the requirements enumerated above about non-overlapping edges and active cycles are matched, with every `dCLK` signal being to be used as clock for a given RNS channel.

4 Application to a Design Example and Simulation Results

A three-stage decimation RNS-based CIC filter [6] was considered for the evaluation of the proposed synchronization technique. CIC filters [5] have been shown to be a useful alternative for FIR implementation for a variety of communication systems, because no multipliers are required. Although the frequency response out of the pass band may be insufficient for certain applications, low-order conventional filters may correct this effect. An S stage CIC system is defined by the transfer function:

$$H(z) = \left(\frac{1 - z^{RD}}{1 - z^{-1}} \right)^S = \left(\sum_{k=0}^{RD-1} z^{-k} \right)^S \qquad (3)$$

Fig. 6 shows the structure of a three-stage CIC filter, illustrating the meaning of R and D parameters. The selected design example was a fully pipelined three-stage CIC decimation filter with R=32, D=2 and 26-bit dynamic range [6]. This RNS-enabled system requires four channels with moduli {256, 63, 61, 59}. This 26-bit dynamic range filter was designed at the transistor level and simulated using PSpice for both a single global clock and the proposed technique. For these

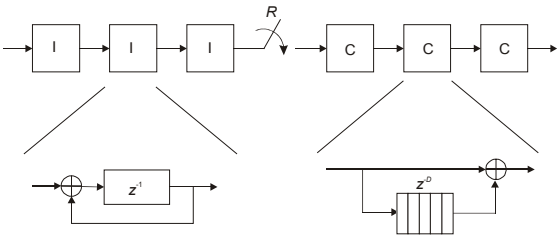

Fig. 6. Three-stage CIC decimation filter.

simulations, a public domain MOSIS CMOS 0.6 μm process [15] was used. This is a three-metal, one-poly, 3.3V CMOS process that is available to MOSIS costumers through Agilent Technologies.

Since the system is composed just of adders and registers, the well-known two-stage modulo adder [6] was used, while registers were implemented using negative edge triggered D flip-flops (nETDFF) based on TSPC logic [1], that simplifies the implementation of the proposed alternative. Because of the great locality of the connections for the study case, load driving is kept to a minimum and device sizes can be fixed to the process minimum for most of the transistor involved. Only transistors used for clock management will require larger sizes since they have to drive large loads. The systems under simulation include around 15.000 transistors.

In order to get illustrative results, the RNS-enabled CIC filter was simulated under three different clocking strategies: first of all, a single global clock was used to synchronize the whole circuit, through a train pulse voltage source. Because this is a "perfect" SPICE source, the second clocking alternative considered a buffer in the clock path while keeping a single clock for global synchronization. Finally, the proposed design example was synchronized using four `dCLK_cell` cells and four generated `dCLK` signals, each one synchronizing an RNS channel. The design parameters for the `dCLK_cell` cells, after careful selection, were selected as L_d=1 μm, W_d=2 μm and W_b=9 μm. These three alternatives were simulated for two different clock frequencies, 125 MHz and 300 MHz. A comparison between the proposed strategy and the buffered clock simulation will illustrate the low power penalty introduced by this new clocking strategy, which is an added feature.

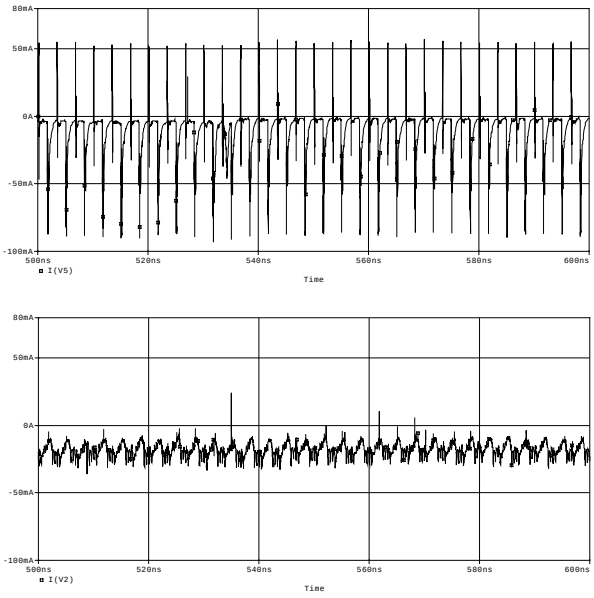

Fig. 7. Current from power supply line for a single clock (above) and the proposed alternative (below).

Fig. 7 shows the current on power supply lines for the CIC full system working with a 300 MHz clock for both a single global clock and the proposed synchronization strategy, while similar results were obtained for the 125 MHz case. The considerable decrease in the magnitude of the current spikes is clearly evident. In this way, current supply to the chip is distributed over time when the new strategy is considered, while for a global clock current spikes are around four times larger. This indicates that the expected benefits derived from the proposed synchronization scheme are confirmed through simulation. Table 1 summarizes the results obtained for the different simulations and both clock frequencies. Note that the maximum current spike is clearly reduced with this new clocking strategy, as well as the maximum value of the current change rate (d*i*/d*t*). This happens for the ideal clock as well as for the buffered one, thus indicating the validity of the proposed strategy. Finally, if power dissipation is considered, the comparison between the buffered clock and the proposed strategy shows that the latter does not introduce a significant increase in power.

5 Conclusions

This paper has presented a new alternative for synchronizing RNS-based systems and it has been shown how this method can reduce the current demand and its change rate. These advantages can be implemented with minimum resources, while power consumption is kept within reasonable values. The proposed strategy was tested using an RNS-enabled three-stage CIC decimation filter consisting of about 15.000 transistors. The application of this new synchronization scheme leads to reduced skew-related problems, as well as reduces chip area through the reduction of the power supply line size, caused by the decrease in current and current change rate requirements.

Acknowledgments

Daniel González, Antonio García, Javier Ramírez, Luis Parrilla and Antonio Lloris were supported by the *Comisión Interministerial de Ciencia y Tecnología* (CICYT, Spain) under project PB98-1354. Graham A. Jullien acknowledges financial support from Micronet R&D, iCORE and NSERC.

References:

[1] J. Yuan and C. Svensson, High-speed CMOS Circuit Technique, *IEEE Journal of Solid State Circuits*, Vol 24, No. 1, pp. 62-70, Jan. 1989.

[2] D. W. Bailey and B. J. Benchsneider, Clocking Design and Analysis for a 600-MHz Alpha Microprocessor, *IEEE Journal of Solid State Circuits,* Vol 33, pp.1627-1633, Dec 1998.

[3] P. Ramanathan, A. J. Dupont and K. G. Shin, Clock Distribution in General VLSI Circuits. *IEEE Transactions on Circuits and Systems I:*

	Single clock		**Single buffered clock**		**Proposed strategy**	
	125 MHz	**300 MHz**	**125 MHz**	**300 MHz**	**125 MHz**	**300 MHz**
Max pike	78.0 mA	96.5 mA	64.4 mA	93.8 mA	23.4 mA	32.8 mA
Max d*i*/d*t*	150 A/ns	254 A/ns	16.3 A/ns	36.6 A/ns	2.2 A/ns	11.9 A/ns
Power	17.0 mW	33.7 mW	19.5 mW	35.8 mW	23.5 mW	57.3 mW

Table 1. Summary of simulation results for an RNS-enabled three-stage CIC filter.

Fundamental Theory and Applications, Vol 41, No. 5, pp. 395-404, May 1994.

[4] J. Yoo, G. Gopalakrishnan and K. F. Smith, Timing Constraints for High-speed Counterflow-clocked Pipelining, *IEEE Transactions on VLSI Systems,* Vol. 7, No. 2, pp. 167-173, Jun. 1999.

[5] E. B. Hogenauer, An Economical Class of Digital Filters for Decimation and Interpolation, *IEEE Transactions on Acoustics, Speech and Signal Processing*, Vol. 29, No. 2, pp. 155-162, Feb. 1981.

[6] U. Meyer-Bäse, A. Garcia and F. J. Taylor, Implementation of a Communications Channelizer using FPGAs and RNS Arithmetic, *Journal of VLSI Signal Processing*, Vol. 28, No. 1/2, pp. 115-128, May 2001.

[7] W. D. Grover, A New Method for Clock Distribution. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, Vol 41, No. 2, pp. 149-160, Feb. 1994.

[8] M.A.B. Jackson, A. Srinivasan and E.S. Kuh, Clock Routing for High Performance IC's *27th ACM/IEEE Design Automation Conference*, 1990.

[9] D. F. Wann and N. A. Franklin, Asynchronous and Clocked Control Structures for VLSI Based Interconnect Networks, *IEEE Transactions on Computers*, Vol 32, No.5, pp. 284-293, May 1983.

[10] E. G. Friedman, Clock Distribution Networks in Synchronous Digital Integrated Circuits. *Proceedings of the IEEE,* Vol. 89, No.5, May 2001.

[11] E. G. Friedman and S. Powell, Design and Analysis of an Hierarchical Clock Distribution System for Synchronous Cell/macrocell VLSI, *IEEE Journal of Solid State Circuits*, Vol. 21, No. 2, pp., 240-246, Apr. 1986.

[12] M. Shoji, Elimination of Process-dependent Clock skew in CMOS VLSI. *IEEE Journal of Solid State Circuits,* Vol. 21, pp. 869-880, Oct. 1986.

[13] N. S. Szabo and R. I. Tanaka, *Residue Arithmetic and Its Applications to Computer Technology*, McGraw-Hill, NY, 1967.

[14] M. A. Soderstrand, W. K. Jenkins, G. A. Jullien and F. J. Taylor, *Residue Number System Arithmetic: Modern Applications in Digital Signal Processing*, IEEE Press, 1986.

[15] MOSIS Process Information, Hewlett Packard AMOS14TB, `http://www.mosis.org/technical/processes/proc-hp-amos14tb.html`