Asynchronous Gate-Diffusion-Input (GDI) Circuits

Arkadiy Morgenshtein, Michael Moreinis and Ran Ginosar
Electrical Engineering Department, Technion—Israel Institute of Technology, Haifa 32000, Israel
[ran@ee.technion.ac.il]

Abstract: Novel Gate-Diffusion Input (GDI) circuits are applied to asynchronous design. A variety of GDI implementations are compared with typical CMOS asynchronous circuits. Dynamic GDI state holding elements are 2× smaller than CMOS C-elements, 30% faster, and consume 85% less power, but certain CMOS elements are preferred when static storage is called for. A GDI bundled controller outperforms CMOS on all accounts, having 1/3 the delay and requiring less than half the area while consuming the same power. A combination CMOS-GDI circuit provides the optimal solution for qDI combinational logic, saving 1/3 the power, half the area and 10% in delay relative to a CMOS implementation. GDI circuits provide some measure of enhanced hazard tolerance, and are more suitable for low voltage operation.

Index Terms – GDI, asynchronous circuits, low-power design, delay, combinatorial logic, C-element.

I. Introduction

Asynchronous design has been established as a competitive alternative to synchronous circuits thanks to the potential for high-speed, low-power, reduced electromagnetic interference, and timing modularity [1]. However, these desirable characteristics usually come at a cost of either silicon area, or speed, or power, and cannot be achieved all at once. Furthermore, asynchronous circuits are typically more complicated than their synchronous counterparts. Many researchers have sought efficient asynchronous circuit implementations, e.g., aggressive pulsed circuits [2][3], fast FIFO stages [4] and dynamic structures [5].

In this paper we propose a novel application of GDI (Gate-Diffusion Input) circuits to asynchronous design. The GDI methodology has originally been introduced for the design of low-power combinational synchronous circuits [6][7][8].

GDI implementations of basic asynchronous circuits are presented and analyzed. SR-Latches and C-elements are compared with a variety of CMOS state holding circuits. A bundled-data controller and two qDI combinational logic circuits (a XOR gate and a full adder) demonstrate that systems employing GDI components outperform standard CMOS implementations in area, power and speed. Furthermore, they provide some enhanced hazard tolerance and are suitable for low supply voltage operation. All designs are validated and compared using SpectreS simulations.

II. GDI as Alternative Circuit Methodology for Asynchronous Design

The GDI method [6][7][8] is based on the use of a simple cell as shown in Fig. 1. At a first glance the basic cell resembles the standard CMOS inverter, but there are some important differences: GDI cell contains three inputs – G (the common gate input of the nMOS and pMOS transistors), P (input to the outer diffusion node of the pMOS transistor) and N (input to the outer diffusion node of the nMOS transistor). The Out node (the common diffusion of both transistors) may be used as input or output port, depending on the circuit structure.

Table 1 shows how a simple change of the input configuration of the simple GDI cell corresponds to very different Boolean functions. Most of these functions require a complex (6-12 transistors) gate in CMOS (as well as in standard PTL implementations), but are very simple (only two transistors per function) in the GDI design methodology. GDI enables simpler gates, lower transistor count, and lower power dissipation.

Multiple-input gates can be implemented by combining several GDI cells. The buffering constrains, due to possible VT drop are described in detail in [8], as well as the technological compatibility with CMOS (and with SOI).

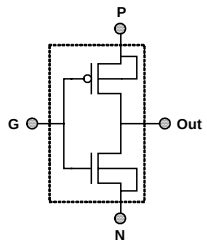


Fig. 1. GDI basic cell

N	P	G	Out	Function
'0'	B	A	$\overline{A}B$	F1
B	'1'	A	$\overline{A}+B$	F2
'1'	B	A	$A+B$	OR
B	'0'	A	AB	AND
C	B	A	$\overline{A}B+AC$	MUX
'0'	'1'	A	$\overline{A}$	NOT

Table 1. Some logic functions that can be implemented with a single GDI cell

III. GDI Implementation of the C-Element and SR-Latch

Combinational GDI circuits have been shown to be fast and low-power relative to CMOS and PTL implementations [6][7] [8]. In the following we analyze and compare the GDI C-element and SR-latch. For purpose of comparison we consider four C-element circuits from [9]: The dynamic, conventional [10], week feedback [11] and symmetric [12] circuits (Fig. 2), of which the latter was identified as the most energy-efficient and high-speed implementation [9]. In addition, we also consider the static (protected feedback) circuit [13] (Fig. 2d).

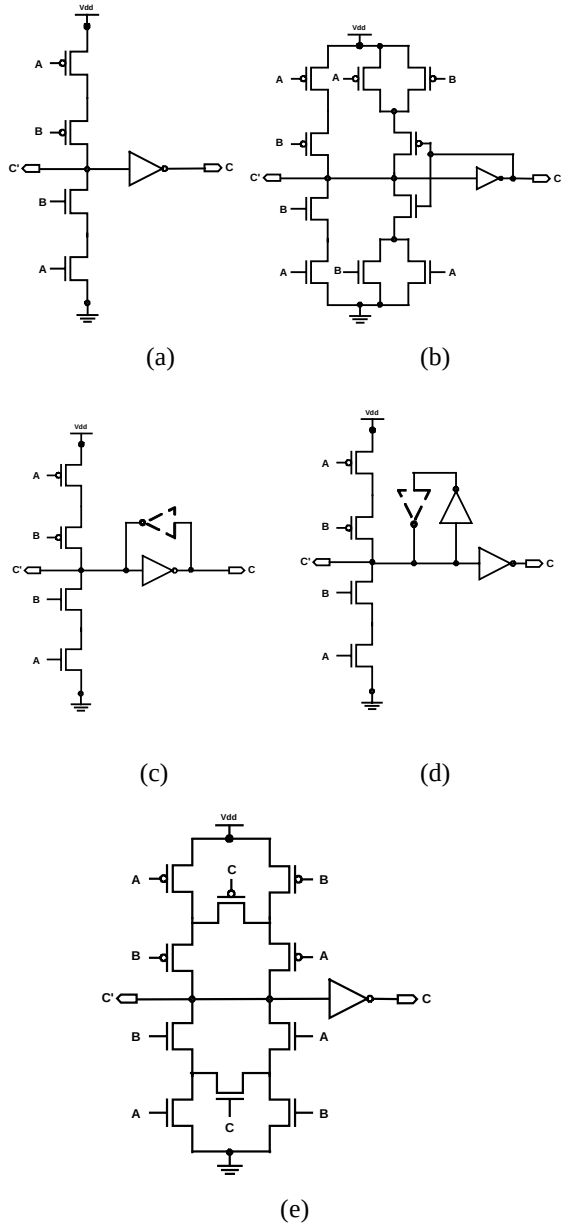


Fig. 2. CMOS C-element Circuits: (a) dynamic, (b) conventional, (c) weak feedback, (d) static, (e) symmetric.

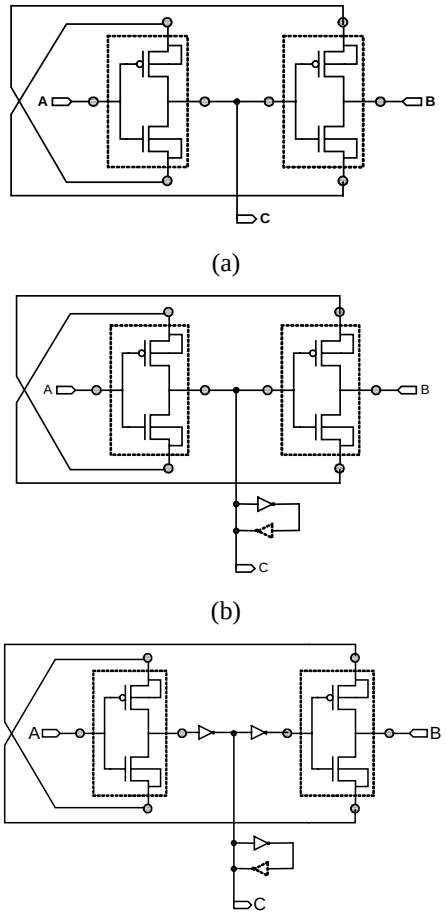


Fig. 3. GDI implementations of C-element: (a) dynamic, (b) static, (c) buffered static

The dynamic GDI C-element circuit is shown in Fig. 3a. It comprises two GDI cells with cross-connected diffusion areas. Note that the common diffusion node of the GDI cell is used both as input (B) and output (C). Likewise, the outer diffusion nodes of each GDI cell are used as bi-directional terminals. The dynamic GDI C-element employs only four transistors, as compared with six transistors in the CMOS dynamic circuit (Fig. 2a).

The weak feedback GDI C-element is shown in Fig. 3b. It requires eight transistors, rather than 10 in the CMOS circuit (Fig. 2d), because the output inverter is not needed.

The paths from input to output in either of the GDI circuits always pass through one nMOS and one pMOS transistors. In contrast, CMOS C-elements contain pull-up paths that traverse two pMOS transistors in series. This difference contributes to the lower delay of the dynamic GDI C-element. This advantage is particularly important for low supply voltages, typically employed for reduced power consumption.

While the A input in both GDI circuits drives transistor gates, the B input does not drive any gates of the GDI cells; rather, it is only gated to the output through pass transistors. The signal path to the output is double-controlled, by the other input (A) and by the output (C). This double-control reduces the probability of output hazards. This advantage is

extremely useful in asynchronous design, where the C-element is often assumed an atomic, hazard free building block [1]. However, due to transmission through two pass transistors, the B signal degrades by at least one V_T. In addition, that signal needs to drive not only the load, but also the feedback inverter. Consequently, the B→C path becomes critical in the C-element. Finally, the B signal presents an increased load on the previous stage (which sources B).

This problem may be solved by buffer insertion. The buffered GDI C-element is presented in Fig. 3c. Here, instead of adding a two-inverter buffer at the output, the inverters were distributed inside the circuit before and after the output C. This may make the circuit more efficient, charging both external and internal gates. In addition to their amplification role, the inverters perform a swing restoration, so that no V_T drop is observed at the output. The buffered GDI C-element, however, is less area efficient.

GDI and CMOS three-input C-elements are shown in Fig. 4. As explained above, the problem of a high pMOS stack in CMOS C-elements is somewhat mitigated in the GDI circuit.

In cases where the C-element inputs are mutually exclusive [1], it can be successfully represented by an SR-Latch, as shown in Fig. 5. GDI implementations of the SR-latch are presented in Fig. 6. The A input is inverted, as is typically useful in asynchronous applications (Fig. 5). The implementation is area-efficient: The SR-latch requires only two GDI cells (four transistors). Two configurations are presented, F1- and F2-based (the F1 and F2 functions are defined in Table 1).

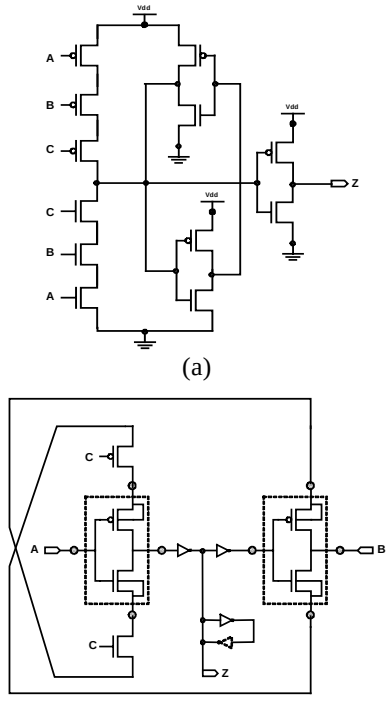


Fig. 4. Three-input C-elements in (a) CMOS and (b) GDI

C-elements in common applications such as in Muller pipelines require one inverted input. While in CMOS C-elements this is achieved by adding an inverter, in GDI the inversion can be performed by simply switching the interconnects of the diffusion nodes as shown Fig. 7. This eliminates the need for an additional inverter and reduces the delay of the Acknowledge signal in the Muller pipeline. In the case of GDI SR-latch, an inverter is removed from one of its inputs, making it an even smaller circuit.

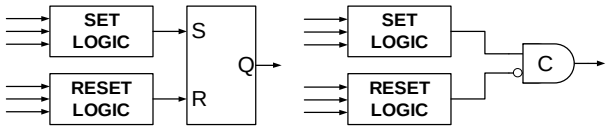


Fig. 5. Representation of C-element by SR-Latch

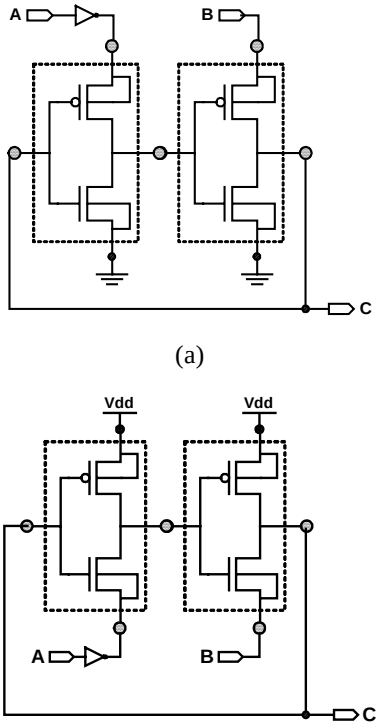


Fig. 6. GDI SR-Latch circuits using (a) F1, (b) F2 functions

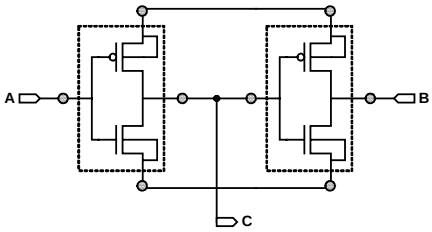


Fig. 7. GDI implementations of dynamic C-element with inverted input A.

IV. Comparison of GDI and CMOS C-Elements

All GDI and CMOS circuits were designed for a 0.35μm technology with 3.3V supply. The circuits were simulated with the SpectreS simulator, and comparisons were performed in terms of average power consumption, maximal delay and number of transistors (area) of the circuit. Fig. 8 illustrates the simulation environment. The C-element is driven by two inverters, which are driven by ideal sources, to imitate the real environment and signals. The inverters are also useful for measuring the current flow from VDD that is caused by transitions in the diffusion inputs in GDI which sink current from the previous logic stage. The C-element is driving a 100fF load capacitor.

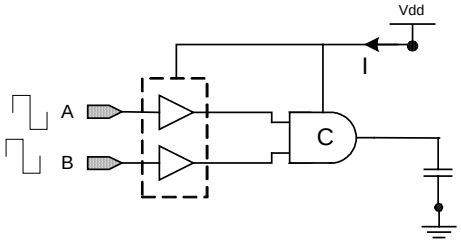


Fig. 8. Simulation environment of C-element.

The keepers had a minimal W/L=0.35/0.35μm size, main body transistors size was 1/1μm for nMOS and 4/1μm for pMOS transisotrs, and the weak inverters size was 1/4μm. Simulation results are presented in Fig. 9.

Simulations of C-elements (light bars in Fig. 9)

Average Power: Best results of average power were observed for dynamic GDI – 94% less than the static CMOS implementation, and 80% less than the dynamic CMOS circuit (which is the best CMOS implementation in terms of power). GDI SR-latch-based C-element have shown results close to the CMOS dynamic circuit, and better than any static CMOS implementation.

Maximal Delay: The Dynamic GDI C-element is the fastest circuit, showing up to 89% decrease compared to standard CMOS techniques, and a 63% improvement compared to the Symmetric C-element, which is the fastest technique among CMOS circuits.

Number of transistors: Dynamic and SR-based GDI circuits are the most area efficient (up to 33% less transistors than CMOS). Buffered GDI, on the other hand, has the highest number of transistors among the GDI circuits (12 transistors).

In summary, CMOS C-elements are preferred over GDI for some static circuits, but in other cases the dynamic GDI C-element or the GDI SR latch may offer a superior solution.

Simulations of C-elements with inverted input A

While the implementation of one inverted input requires an extra inverter in CMOS C-elements, GDI circuits either retain the same complexity or even get smaller (in the case of SR-based C-elements). This contributes to the superior performance of GDI, as shown by the dark bars in Fig. 9.

Average Power: GDI offers up to 85% improvement in power dissipation compared to CMOS. This is consistent with the size reduction in SR-based circuits by elimination of the input inverter.

Maximal Delay: SR-F1, SR-F2 and the dynamic GDI demonstrate the shortest delay among all circuits. In total the delay improvement in GDI is in the 22%-82% range compared to CMOS.

Note that the inverted input GDI C-element is slower than the non-inverted input one. This is due to the fact that while in the non-inverted GDI each path through the pass-transistors contains one nMOS and one pMOS transistors, in the inverted input GDI one of the paths goes through two pMOS transistors.

Number of transistors: As explained above, inverted-input CMOS circuits are bigger than non-inverted ones, and the opposite is true for the SR-based GDI circuits. Other GDI circuits have the same size in both cases.

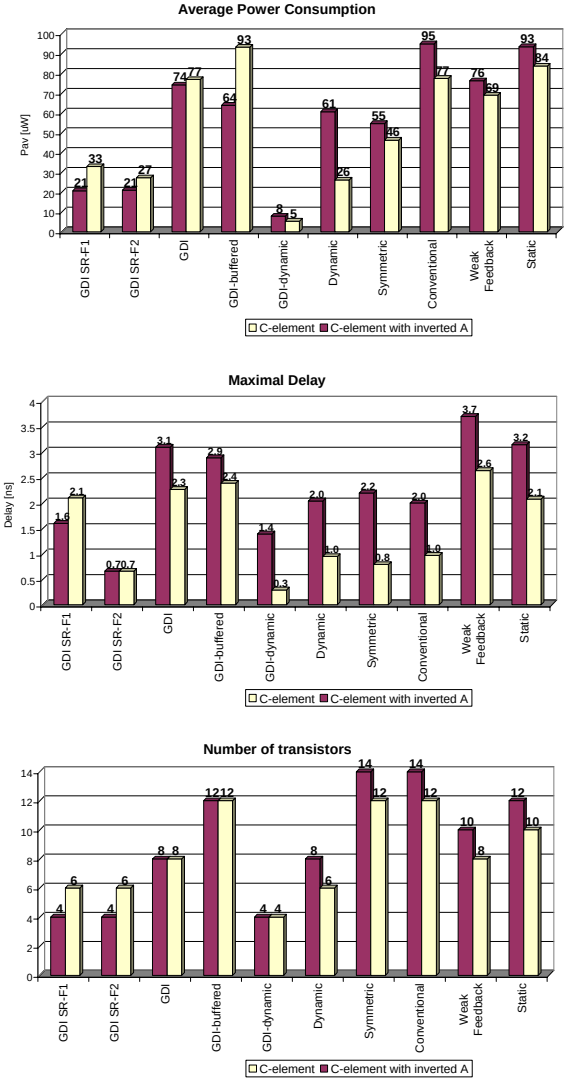


Fig. 9. Simulation results for the C-element implementations

Simulations of 3-input C-elements

The simulations of the three-input C-elements (Fig. 10) were performed over four supply voltage values from 3.3V down to 1.57V. GDI achieves a lower delay, showing 35% advantage over CMOS.

It can be explained by better conductivity of the nMOS-double-pMOS path of GDI over the triple-pMOS stack of the CMOS circuit. The buffered GDI circuit consumes more power than CMOS, due to its higher complexity. The advantage of GDI when the power supply voltage is decreased is expressed in the higher gradient of its power curve, which falls much faster than CMOS with decreasing supply voltage. It results in improved power-delay parameter of GDI, with a crossing point at 2V supply, below which GDI shows superior results. Both circuits cannot operate below 1.57V in this technology.

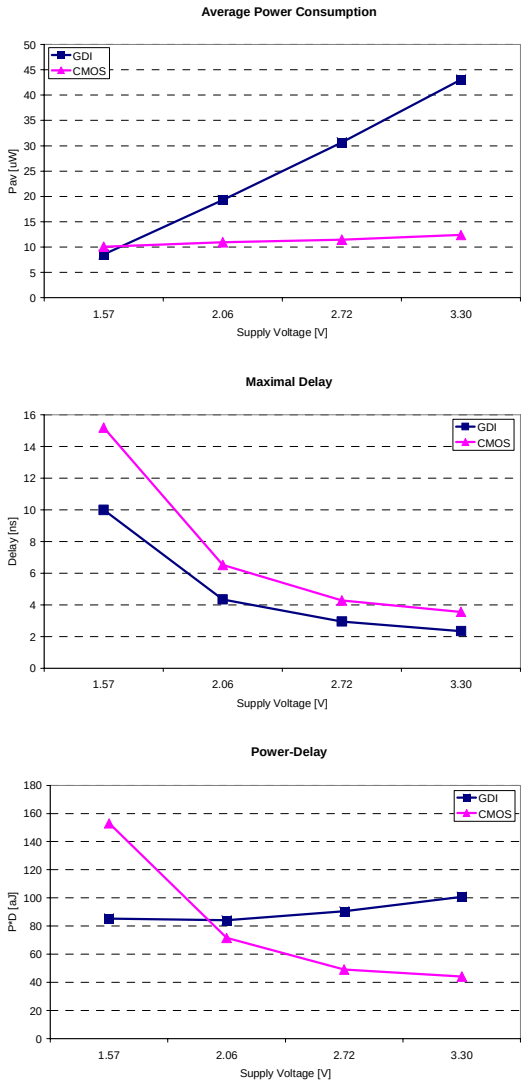


Fig. 10. Simulation results of operation at various supply voltages of 3-input C-element.

V. Comparison of Bundled-Data Controllers in CMOS and GDI

In order to perform a comparison of a robust asynchronous circuit implemented in CMOS and GDI, a Bundled-Data Filter Controller was selected. Fig. 11 shows the structure of the filter and the STG flow of its controller [10]. Petrify CMOS implementation of the controller is described in Fig. 12a. We use a CMOS Symmetric C-element in this comparison, to obtain a low-power circuit.

For the GDI circuit (Fig. 12b), we replaced the inverted-input-AND gates with GDI-OR gates (Table 1) and inverters. Using OR function resulted in a reduced number of transistors, and the inverters helped in swing-restoration. The inputs of the C-element are mutually exclusive, and hence it has been replaced by the smaller, faster and lower power GDI SR-latch.

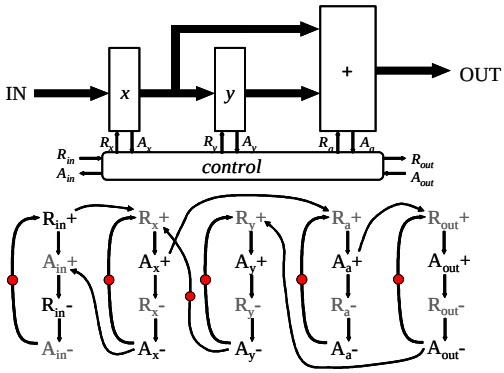
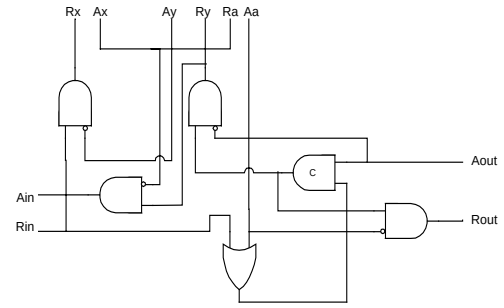
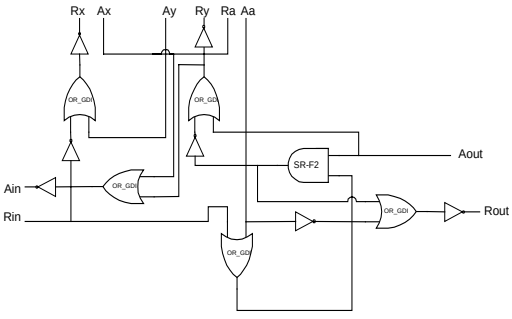


Fig. 11. Bundled-Data Filter Controller



(a)



(b)

Fig. 12. Bundled-Data Filter Controller: (a) CMOS-based, (b) GDI-OR-based.

RC delay units with time constant of 0.1ns were inserted between each Request and its corresponding Acknowledge signals, to emulate a reasonable environment.

Simulation results can be seen in Fig. 13. The GDI implementation requires only 20 transistors, as opposed to 50 in CMOS. The GDI controller is about 3× faster than the CMOS circuit, consuming about the same power. The reduced circuit complexity and the superior properties of the GDI SR-F2 (Section IV) are the main contributors for the advantages of the GDI controller.

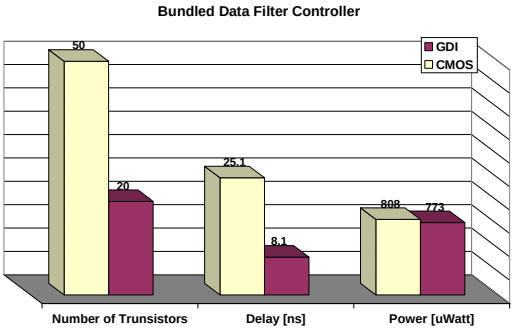


Fig. 13. Simulation results for Bundled-Data filter controller.

VI. GDI qDI Combinational Logic Circuits

In this section we investigate a qDI combinational logic circuit implemented in CMOS and GDI according to the DR-ST design methodology [15]. The *n*-input, *m*-output DR-ST circuit comprises four interconnected subnets (Fig. 14): ORN and CEN detect when all the inputs become *defined* or *undefined*. DRN is a monotonic implementation of the dual rail combinational functions, and OUTN enforces the strong conditions (all outputs remain *undefined* until all inputs become *defined*, and all outputs remain *defined* as long as not all inputs have become *undefined*). Other qDI implementations include Delay Insensitive Minterm Synthesis (DIMS) [16] and RSPCFB [5].

A simple XOR gate is used as an example. The CMOS and GDI implementations of the ORN and DRN subnets of the XOR DR-ST gate are presented in Fig. 15 and Fig. 16, respectively. Symmetric C-elements were used for the CMOS CEN and OUTN subnets, while the GDI implementation was based on the buffered GDI C-element.

We compared three different combinations of subnet implementations (Fig. 17): (a) *CMOS* – all four subnets are CMOS circuits. (b) *GDI* – all four subnets are GDI circuits. (c) *Hybrid* – ORN and DRN are GDI, CEN and OUTN are CMOS.

Simulation results are shown in Fig. 18. The GDI and Hybrid circuits are 38% smaller than the CMOS one. The GDI circuit is slower and consumes more power than the CMOS circuit; due to the use of buffered GDI C-elements, which are required in this case for their drive capability. But the Hybrid circuit, made of GDI gates and CMOS C-elements, consumes only half the power as CMOS while being just as fast. Still, we suggest that, when hazard immunity and low supply voltage tolerance are critical, such as in low noise, low power applications, an all-GDI circuit should be considered.

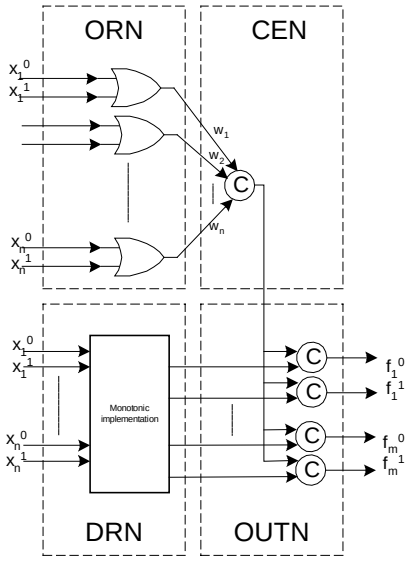


Fig. 14. General structure of DR-ST implementation

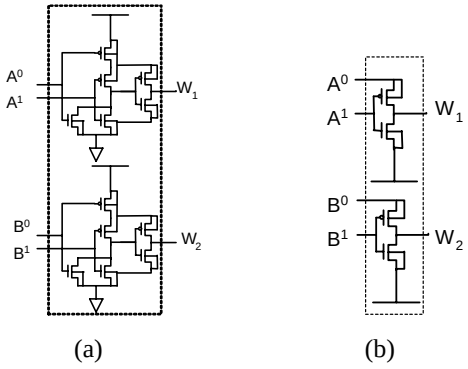


Fig. 15. ORN subnet (a) CMOS based (b) GDI based

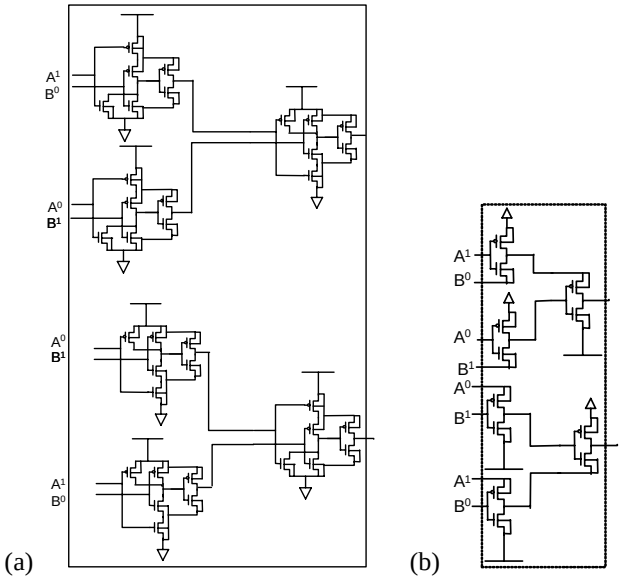
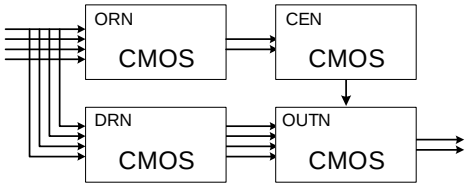
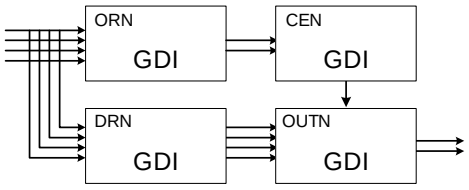


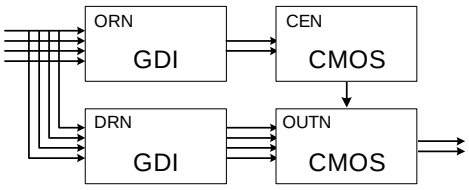
Fig. 16. DRN subnet of XOR: (a) CMOS XOR, (b)GDI XOR.



(a)



(b)



(c)

Fig. 17. Simulated Circuits: (a) CMOS (b) GDI (c) Hybrid

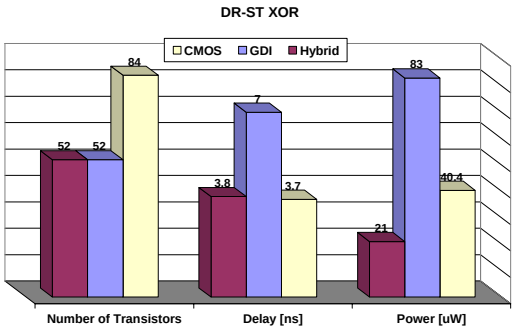
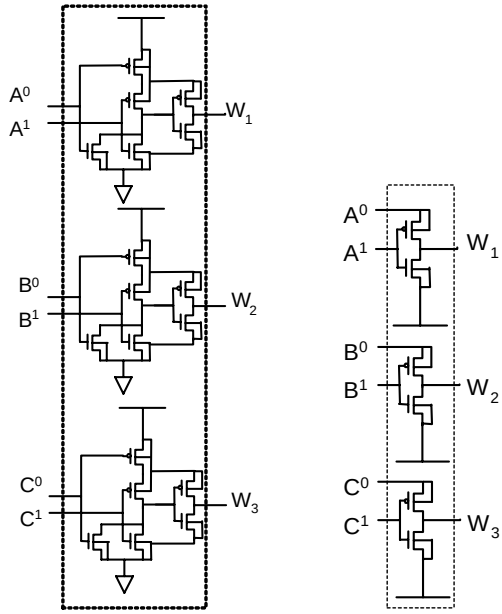


Fig. 18. Results comparison of DR-ST XOR.

VII. DR-ST Full Adder

We now turn to a more complex DR-ST combinational logic circuit. CMOS and Hybrid circuits of a full adder are designed and compared. The ORN and DRN subnets are presented in Fig. 19 and Fig. 20, respectively, and are either GDI or CMOS. The CEN subnet is based on a 3-input static CMOS C-element, while OUTN comprises 2-input symmetric CMOS C-elements.

Simulation results are shown in Fig. 21. In this bigger circuit the Hybrid implementation outperforms CMOS in all aspects. The Hybrid circuit is about half the size and consumes only about 2/3 the power, while being 10% faster than the CMOS one. It appears that qDI GDI circuits should be considered as a viable alternative to CMOS.



(a) (b)

Fig. 19. ORN subnet Full Adder (a) CMOS based (b) GDI based

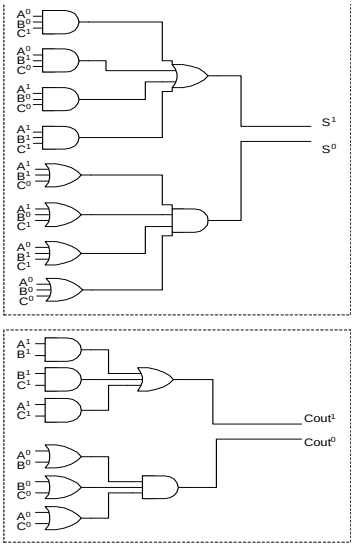


Fig. 20. DRN subnet Full Adder; each gate may be implemented with either CMOS or GDI

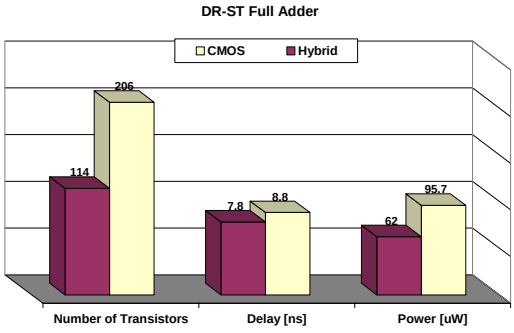


Fig. 21. DR-ST Full Adder performance results

VIII. Conclusions

A novel methodology for asynchronous circuits was presented. It is based on the two-transistor GDI (Gate Diffusion Input) cells [8]. A variety of GDI circuits were compared to conventional CMOS implementations.

Five GDI C-elements were developed and compared with five CMOS circuits. GDI dynamic and SR-latch circuits outperformed CMOS in area, power and speed, but certain CMOS circuits are preferred when static C-elements are needed. GDI performs better under reduced supply voltage, and provides an enhanced tolerance to hazards.

A Bundled-Data controller circuit showed that, under certain circumstances, an all-GDI circuit required less area, ran faster and consumed less power than the CMOS equivalent. A study of two qDI combinational logic circuits (XOR gate and a full adder) revealed, however, that a GDI-CMOS hybrid combination provides the optimal circuit. But in cases of low power or low noise an all-GDI circuit should be considered.

Future research in GDI circuits should consider applications of GDI, synthesis algorithms and the introduction of GDI circuits into existing asynchronous CAD tools, as well as verification of optimal design configurations considering mixed GDI-CMOS circuits and cell libraries.

Acknowledgments

The authors would like to thank Dr. Israel Wagner and Mr. Alexander Fish for constructive comments on the presented implementations.

References

[1] J. Sparsø and S. Furber (eds.), *Principles of asynchronous circuit design - A systems perspective*, Kluwer Academic Publishers, 2001.

[2] K. Stevens, R. Ginosar, and S. Rotem, "Relative timing," in *Proc. International Symposium on Advanced Research in Asynchronous Circuits and Systems*, pp. 208–218, April 1999.

[3] K.S. Stevens, S. Rotem, R. Ginosar, P.A. Beerel, C.J. Myers, K.Y. Yun, R. Kol, C. Dike, M. Roncken, "An asynchronous instruction length decoder," in *IEEE JSSC*, **36**, issue: 2, pp. 217-228, February 2001.

[4] I. Sutherland and S. Fairbanks, "GasP: A minimal FIFO control," in *Proc. International Symposium on Advanced Research in Asynchronous Circuits and Systems*, IEEE Computer Society Press, pp. 46-53, March 2001.

[5] R. O. Ozdag and P. A. Beerel, "High-speed QDI asynchronous pipelines," in *Proc. International Symposium on Advanced Research in Asynchronous Circuits and Systems*, pp. 13–22, April 2002.

[6] A. Morgenshtein, A. Fish, I. A. Wagner, "Gate-Diffusion Input (GDI) – A Novel Power Efficient Method for Digital Circuits: A Detailed Methodology," *14th IEEE International ASIC/SOC Conference*, USA, September 2001.

[7] A. Morgenshtein, A. Fish, I. A. Wagner, "Gate-Diffusion Input (GDI) – A Technique for Low Power Design of Digital Circuits: Analysis and Characterization," *ISCAS'02*, USA, May 2002.

[8] A. Morgenshtein, A. Fish, I. A. Wagner, "Gate-Diffusion Input (GDI) – A Power Efficient Method for Digital Combinatorial Circuits," to be published, *IEEE Trans. on VLSI*.

[9] M. Shams, J.C. Ebergen, and M.I. Elmasry, "Modeling and Comparing CMOS Implementations of the C-Element," *IEEE Trans. VLSI*, **6**, no. 4, December 1998.

[10] I.E. Sutherland, "Micropiplines," *Comm. ACM*, **32**, pp. 720-738, June 1989.

[11] A.J. Martin, "Formal program transformations for VLSI circuit synthesis," in E.W. Dijkstra (ed), *Formal Development of Programs and Proofs*, UT Year of Programming Series, pp. 59-80. Addison-Wesley, 1989.

[12] A. Peeters and K. van Berkel, "Single-rail handshake circuits," *Asynchronous Design Methodologies*, IEEE Computer Society Press, pp. 53-62, May 1995.

[13] D.E. Muller and W.S. Bartky, "A Theory of Asynchronous Circuits," *Proc. Int. Symp. on the Theory of Switching*, Cambridge, MA: Harvard University Press, pp. 204-243, April 1959.

[14] J. Cortadella, M. Kishinevsky, A. Kondratyev and L. Lavagno, "Introduction to asynchronous circuit design: specification and synthesis," *Tutorial, Async. Conference*, 2000.

[15] I. David, R. Ginosar, and M. Yoeli, "An Efficient Implementation of Boolean Functions as Self-Timed Circuits," *IEEE Trans. Computers*, pp. 2-11, January 1992.

[16] J. Sparsø and J. Straunstrup, "Delay insensitive multi-ring structures," *Integration, the VLSI journal*, **15**(3), 313-340, October 1993.