International Journal of Engineering & Technology, 7 (2.21) (2018) 123-126

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Latency and throughput analysis of a pipelined GDI ripple carry adder

N. Shylashree[1*], D.S. Mahesh[2]

[1]*Associate Prof. Dept. of ECE, R.V.C.E. Bengaluru.*
[2]*Student. Dept. of ECE, R.V.C.E. Bengaluru*
**Corresponding author E-mail: shylashreen@rvce.edu.in*

Abstract

Latency and Throughput are deemed parameters of prime importance that determine the speed of an Adder Circuit. Ongoing research in the field of Digital Signal Processing involves optimizing an Adder regarding these parameters. This article picks up the study of a ripple carry adder and presents the use of two methods towards ameliorating the performance of an adder – viz., the use of GDI (Gate Diffusion Input) technology for reduced Latency, and implementation of a pipelined architecture towards increasing the throughput. In this paper, we have dileneated the function of a basic GDI cell, with which a 1-bit ripple carry full adder was designed, which in turn formed the basic building blocks of 8-bit and 32-bit ripple carry adders. These full adders were designed using GDI technology while employing the concept of pipelining resulting in a novel structure optimizing both latency and throughput. This paper also presents a comparison among CMOS and GDI RCAs of 8 and 32bits with and without pipelining.
On simulating 32-bit RCAs in Cadence virtuoso tool using gpdk 180nm technology ,those with pipelining had a 4.5 times increase in throughput with 42.8% increase in latency.

Keywords: *Latency, throughput, digital signal processing, ripple carry adder, GDI, pipeline.*

1. Introduction

Addition is a basic building block for almost all the operations done on modern computing systems [1],[10]-[11]. Prevailing computers of today use a maximum of 64-bit width bus. Whereas, modern cryptographic applications, e.g. Elliptic Curve Cryptography (ECC), and also quad-precision floating point arithmetic demand precisions well beyond 64 bits. A trivial system would use an arrangement of Ripple Carry Mechanisms. This warrants larger latencies and poor throughputs. This paperpresents the study of two possible solutions, viz., use of a different technology, as opposed to the intuitive idea of pipelining. A novel approach would be to use both, which also lies within the scope of this paper.
Proposed method for reducing latencies use minimum delay logic. Here, GDI is presented as a viable option. Three parameters are held as the basis for analyzing the performance of a gate level system, viz., Latency (or Delay), Power Consumption and the Number of Transistors used, along with the quality of waveforms output by each gate. CMOS is compared along with GDI and the various trade-offs with regards to the above parameters are tabulated.
Also, Pipelining is envisaged as a viable solution for increasing the throughput of a system. Pipelined adders are designed with both of the above gates at the core, and the feasibility in scaling a system to higher bus widths for each gate is studied. A detailed description of GDI cell is given in [1]-[3], including it's advantages.[10] has given a brief overview about RCAs.

2. Gate diffusion input

A GDI cell is shown in Fig. 1 [1]. It is somewhat similar to a CMOS inverter circuit, however, the bulks of both the transistors form a single input, while the supply railsanalogous to CMOS form two more inputs, viz, P and N,[2],[3],[7],[8].

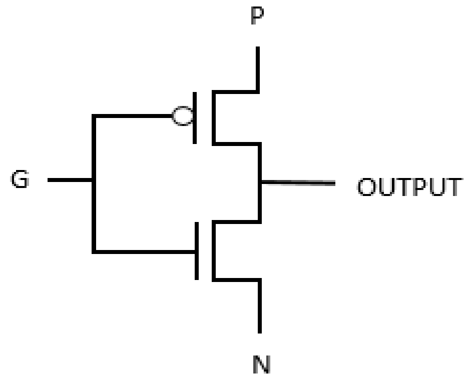

Fig.1: GDI Cell

The main motivation for GDI was to reduce the number of transistors required to realize logic gates and to optimize performance parameters like speed and power consumption[9]. GDI can be viewed as a form of Pass Transistor technology where multiple Boolean operations can be realized by varying the input configurations at the terminals. The catch, however is reduced voltage swings due to threshold losses [1]. Figure 2. Shows the schematic of a GDI cell, which is designed using Cadence Virtuoso design tool using gpdk 180 nm technology.

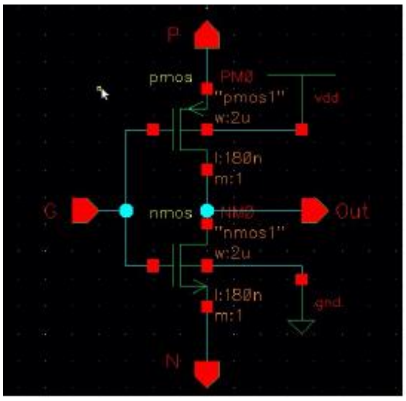

Fig.2: Schematic of a GDI cell

Table 1 displays the various functions realized using GDI based on different scenarios of input combinations at the terminals [2]-[3]. The general function of the GDI cell is G'P+GN Substituting the input signals in this function yields the output signal in terms of the inputs as shown in Table 1.Besides the multitude of functions realized, Table 1 also displays the no. of transistors required to realize these functions based solely on GDI.

Table 1: Various Boolean Functions Realized using GDI Along with Transistor Count

N	P	G	Output	Function	Transistor Count
0	1	A	A'	Inverter	2
0	B	A	A'B	F1	2
B	1	A	A'+B	F2	2
1	B	A	A+B	OR	2
B	0	A	AB	AND	2
C	B	A	A'B+AC	MUX	2
B'	B	A	A'B+B'A	XOR	4
B	B'	A	AB+A'B'	XNOR	4

Aside from the advantage of low complexity design, less transistor count and greater speed [3], GDI have a few drawbacks compared to CMOS. The reduced voltage swings can be detrimental to the performance of the circuit [1] and in also switching operations when connected in cascade. In case low supply rails, it may even lead to circuit malfunction.A full Adder implementation employing GDI technology is shown in Fig. 3.

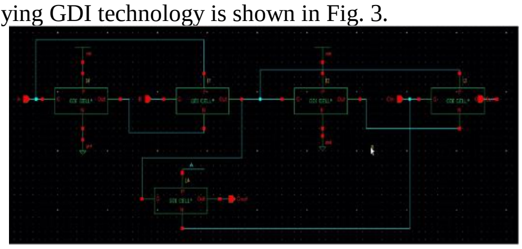

Fig.3: 1-bit full adder using GDI

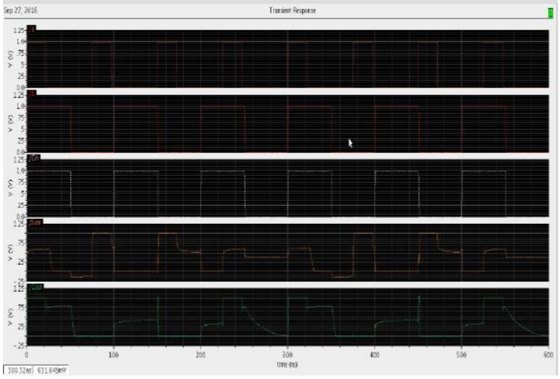

Fig.4: Output wave form of 1-bit GDI full adder without static inverter. The waveforms depict the 'Sum' followed 'Carry out' of inputs a, b and cin (first 3 waveforms).

Fig 4 displays the output waveform of a 1-bit GDI full adder without the use of a static inverter to improve voltage swings. The output is degraded due to threshold losses inherent in GDI.

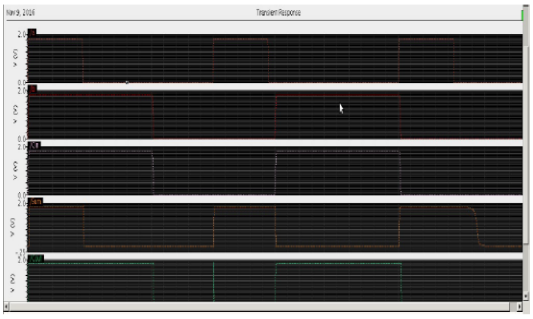

Fig.5: Output waveform of 1-bit GDI full adder using static inverterwhere the first 3 are input signals followed by sum and carry-out.

As shown in Figure 4, the output waveforms of Sum and Carry-out are undoubtedly distorted. This drawback can be overcome using swing restoration buffers employing static inverters as evidently seen in Figure 5.By using static inverters we are able to achieve optimum voltage swings, at the cost of increased transistor count.

3. Pipelining

In order to improve instruction throughput and performance of processors, they make extensive use of a technique called pipelining. Instead of waiting for the results of a previous instructions to be written to register files or main memory, these pipelined processors fetch and execute the next instruction as soon as the previous instruction has been fetched and dispatched to the instruction register. As a result the processor starts fetching the next instruction before the previous operation is completed.

Pipelining as a concept is applicable to any system wherein the entire operation is broken down into different stages. This introduces granularity into the process and thereby, each stage has to deal with only its own specific sub task.

Pipelining involves the use of memory for local storage of data and hence tends to increase the transistor count. But it is to be noted that the aforementioned GDI technology can drastically reduce the transistor count and hence strike a balance with a conventional adder.

Pipelining applicable to an adder is shown in Fig. 6.Hence, the delay of the circuit becomes $4(T_{adder}+ T_{reg})$, an opposed to $4T_{adder}$, that of a Rippled Adder. But further intuition allows us to note that around four times more calculations can be performed in one such cycle, as opposed to a single one of the Rippled Adder, which gives a rise in the throughput of the system.

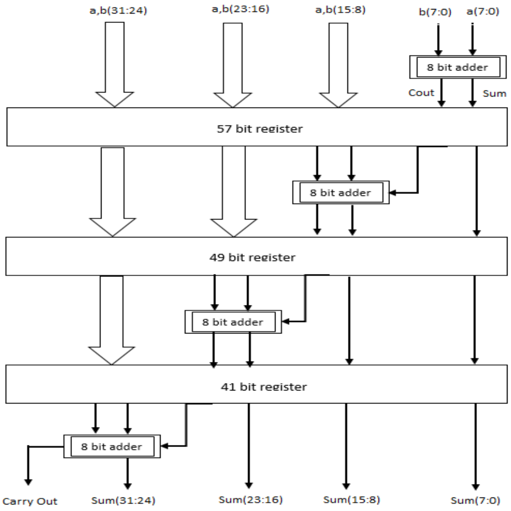

Fig.6: A pipelined adder

4. Proposed architecture

The adder architecture proposed in this paper calls for a union between both GDI technology as well as Pipelining, as mentioned previously. It is to be noted that the registers are to be clocked at rates not more than the latencies of each adder block. The basic gates like XOR, AND and OR that make a full adder is fabricated using a basic GDI cell. Depending on the nature of inputs at the terminals, GDI can behave as the intended logic gate. Fig3. Shows the design of the 1-bit full adder which is based on GDI .This 1-bit full adder will serve as a building block for the 8-bit ripple carry adder (RCA) as shown in Fig 7. This in turn would be a base for the 32-bit RCA. Using the 1-bit full adder, an 8-bit RCA can be constructed by connecting the carry-in and carry-out of successive full adders [10].

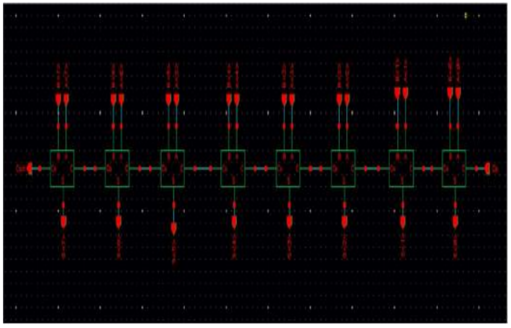

Fig.7: 8-bit RCA using GDI

Fig.8 and Fig.9 shows the architecture for the 32-bit RCA (without and with pipelining) using the 8-bit adders as the building blocks .The architecture is similar to that of the 8-bit RCA in the sense that the carry-outs and carry-ins of the successive adders are connected.

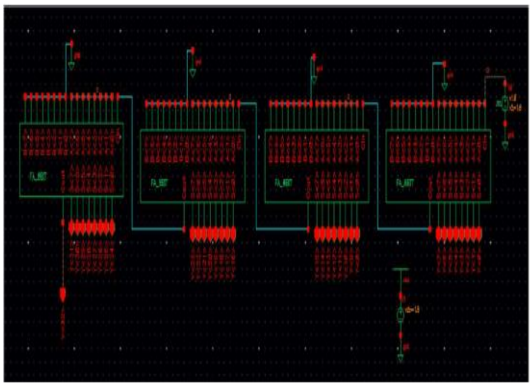

Fig.8: 32-bit RCA without pipelining

If Tadder is the delay of a single 8-bit RCA then its latency and throughput are 4Tadder and 1/4Tadder respectively. However by using registers for pipelining it's delay is increased to 4(Tadder+Tregister).But the registers inserted between successive levels incorporates fetching of 32-bit data for the next cycle as soon as the 32-bit addition is completed thus allowing the proposed 32-bit pipelined RCA to perform addition on the next set of values without waiting for the MSB in the previous cycle. By incorporating this model, despite the use of registers, consuming area ,with GDI reducing the transistor count ,this architecture strikes a balance between increased use of memory elements and improved throughput of 1/Tadder.

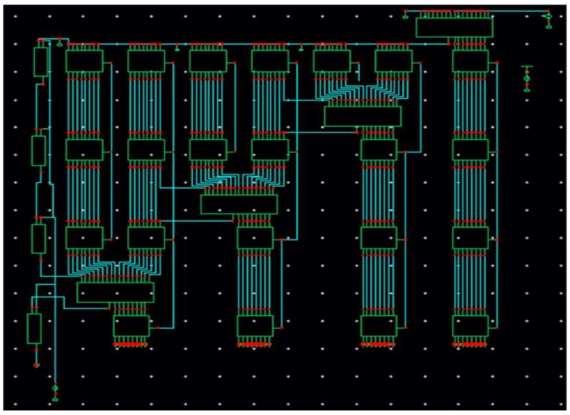

Fig.9: 32-bit RCA with pipelining

5. Results

A comparative analysis of performance parameters of 1-bit and 8-bit full adders using both CMOS and GDI technologies is delineated in table 2.

Table 2: Comparison of Performance Parameters of Full Adder

	No. of Transistors	Delay(ps)	Power(μW)
CMOS 1-bit	28	497	0.33
GDI 1-bit	10+8*	107	0.315
8-bit RCA using CMOS	224	4073	6.775
8-bit RCA using GDI	80+36*	622	6.422

Another set of values for performance parameters of 32-bit RCA using both CMOS and GDI technologies are tabulated in table 3.

Table 3: Comparison of Performance Parameters of 32-bit RCA

	No. of transistors	Delay(ns)	Power(μW)
32-bit CMOS RCA without pipelining	896	16.56	388
32-bit CMOS RCA with pipelining	1237	17.947	728
32-bit GDI RCA without pipelining	320+132*	3.424	378
32-bit GDI RCA with pipelining	805	4.89	662

* indicates the extra transistors needed for a full voltage gain as the GDI cell suffers from threshold losses. These transistors are used in static inverters to obtain a full voltage gain. Hence incorporating GDI technology significantly reduces the latency and number of transistors required without noticeable change in power consumption. Using pipelining places a significant demand on power and area when using CMOS.As a result, incorporating pipelining on adders using GDI seems to be a viable option. Finally, the latency and throughput analysis of the full adders using both GDI and CMOS technologies is elucidated in table 4. The latency is the computation time of the adders which is inferred from delay itself i.e. Latency = Overall delay of the adder. Also, the throughput is the sampling rate of the input bits and is given as

$$\text{Throughput} = 1/4\text{Tadder} \quad (1)$$

$$\text{Throughput}=1/\text{Tadder} \quad (2)$$

Where Tadder is the delay of 8-bit adder . Eqn(1) refers to throughput of 32-bit RCA without pipelining and (2) refers to that of 32-bit RCA with pipelining.

Table 4: Comparison of latency and throughput

	Latency(ns)	Throughput(Mbps)
8-bit RCA using CMOS	4.703	212.63
8-bit RCA using GDI	0.622	1607.71
32-bit CMOS RCA without pipelining	16.56	60.38
32-bit CMOS RCA with pipelining	17.947	212.63
32-bit GDI RCA without pipelining	3.424	292.056
32-bit GDI RCA with pipelining	4.89	1607.71

6. Conclusion

GDI imparts less delay and consumes less no. of transistors compared to CMOS but at the cost of poorer voltages wings. The poor voltage swings can be improved by using static inverters but need more transistors, hence delay increases. Despite the above requirement GDI is still a good alternative compared to CMOS. Pipelined structures increase the latency of adders when compared to normal RC Adders in purview of the added registers. But, the throughput is increased as the number of calculation per a single amount of latency is quadrupled. Hence, in this paper we have quantitatively delineated the effectiveness of GDI RCA adders with pipelining in terms of latency and throughput. In case of 8-bit adders there is a 86.77% decrease in latency and a 6.56 times increase in throughput, thus demonstrating the effectiveness of GDI. In case of 32-bit GDI RCAs, those with pipelining boast a 4.5 times increase in throughput at the cost of 42.8% increase in latency.

Future scope

- Design of Hardware based Crypt, over software as the former can upscale the security of a system.
- Further gate level enhancements can be made by extending comparisons with other technologies. We envisage the use of GaNFETs (a class of HEMTs) for this, as they are an emerging branch of Microelectronics Devices.
- Fabrication of Security-On-Chips can be commenced with the fastest adder combination found from the above results. These Chips aim at offloading the burden on a system's main processor and function as a Cryptographic co-processor.

References

[1] Krishnendu D, "Design of a Low Power, High Speed, Energy Efficient Full Using Modified GDI and MVT Scheme in 45nm Technology", *International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)*, (2014).

[2] Jubal S & Shoaib K, "A Low Power Variable Sized CSLA Implementation Using GDI Logic In 45nm SOI Technology", *IEEE 1st International Conference on Next Generation Computing Technologies*, (2015).

[3] Biswarup M & Aniruddha G, "Design &Study of a Low Power High Speed Full Adder Using GDI Multiplexer", *IEEE 2nd International Conference on Recent Trends in Information Systems*, (2015).

[4] Shoba M & Nakkeeran R, "Performance Analysis of 1 bit Full Using GDI Logic", *IEEE ICICES2014 - S.A. Engineering College, Chennai, Tamil Nadu, India*, (2014).

[5] Lin CH & Wu AY, "Algorithm and Architecture for High-Performance Vector Rotational DSP Applications", *Regular IEEE Transactions: Circuits and Systems I*, Vol.52, (2005), pp.2385-2398.

[6] Vitoroulis K & Al-Khalili AJ, "Performance of Parallel Prefix Adders Implemented with FPGA technology", *IEEE Northeast Workshop on Circuits and Systems*, (2007), pp.498-501.

[7] Aniruddha G & Biswarup M, "Design &study of a low power high speed full adder using GDI multiplexer", *IEEE 2nd International Conference on Recent Trends in Information Systems (ReTIS)*, (2015), pp.465 – 470.

[8] Shoba M & Nakkeeran R, "Performance analysis of 1 bit full adder using GDI logic", *International Conference on Information Communication and Embedded Systems (ICICES),* (2014), pp.1 – 4.

[9] Anitesh S & Ravi T, "Low power 8-bit ALU design using full adder and multiplexer", *International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, (2016), pp.2160 – 2164.

[10] Rashmi DS, SadiyaRukhsar R, Shilpa HR, Vidyashree CR, Kunjan DS & Nithin HV, "Modeling of adders using CMOS and GDI logic for multiplier applications: A VLSI based approach", *International Conference on Circuit, Power and Computing Technologies(ICCPCT)*, (2016), pp.1 – 6.

[11] Sujatha H & Deepali K, "Low power full adder circuit using Gate Diffusion Input (GDI) MUX", *Fourth International Conference on Communication and Computing*, (2012), pp.53 – 56.

[12] Hoe DHK, Martinez C & Vundavalli J, "Design and Characterization of Parallel Prefix Adders using FPGAs", *IEEE 43rd Southeastern Symposium on System Theory*, (2011), pp.170- 174.

[13] Choi Y, "Parallel Prefix Adder Design", *Proc. 17th IEEE Symposium on Computer Arithmetic*, (2005), pp. 90-98.

[14] Chang L, Fei Q, Xinghua Y & Huazhong Y, "Hardware acceleration with pipelined adder for Support Vector Machine classifier", *Fourth International Conference on Digital Information and Communication Technology and it's Applications (DICTAP)*, (2014), pp.13 – 16.