15th International Conference
MIXED DESIGN
MIXDES 2008
Poznań, POLAND
19 - 21 June 2008

A NEW DIGITAL MULTIPLIER/DIVIDER ARCHITECTURE, VIA HYBRID ANALOG-DIGITAL PROCESSING

N. TAHERINEJAD, H.S. SHAH-HOSSEINI
IRAN UNIVERSITY OF SCIENCE & TECHNOLOGY, IRAN

KEYWORDS: Multiplying/dividing, Hybrid (Analog-digital mixed signal) processing, ADC, DAC

ABSTRACT: Digital multiplying and dividing are time consumer tasks if are done via software. To speed up, the processor special hardware may be developed. Digital multipliers/dividers designed by digital circuits are too huge and complex in a system point of view and each of them need own hardware and so they're occupying large amount of die. In this paper a digital multiplier/divider which uses hybrid processes' of analog and digital circuits will be introduced. This multiplier/divider is very simple and uses the same device for both operations, so it'll occupy the die, less. The idea will be developed in system level and some comparisons with similar devices will be done at this level.

INTRODUCTION

Not processed data is meaningless. So it's a long time which processors are studied and improved in many aspects. Analog processors were substituted by digital ones, cause of their weak points such as inflexibility to change the process procedure, inaccuracy and long design time [1]. Digital ones instead were programmable and accurate. They could also store a large amount of data which analog ones couldn't. More attention to digital processors caused their development. FPGA[1]s dedicated the reconfigurability to these processors and made them more desirable to use.
The benefits of digital processor developed the concept of processing analog signals digitally. ADCs, DACs, mixed-signal microcontrollers and DSPs were developed rapidly when neither types of signal can be avoided, such as interfacing analog sensors [2].
Analog processors were developed too, especially with FPAA[2]s appearance and its improvements [3], they added more flexibility to these circuits. Such development of analog processors and design speed up [4] led in new concepts in processors; analog processing of digital signals and Hybrid digital-analog (Co-) Processors [5].
In fact when processing equations get larger, digital processors become slow cause of they iterative structure and partial solutions, while analog or hybrid analog-digital ones can solve whole the equations or most part of them almost as a real time task [6]. Therefore in such tasks they can be very useful. This conditionally advantage, result in special purpose use of this concept rather than general purpose uses. The most uses of this concept, has been done in Image processing [7, 8], although it's not the lone usage field [9].
Similarly to solve some problem of digital processing a hybrid approach is intended in this paper.
It's obvious that Multiplying and Dividing are two important operations which can't be avoided in the most processes. But these operations are time/die consumer tasks which both means the more cost. To solve this problem, or at least make a trade-off between time or die consuming, an analog-digital hybrid multiplier/divider will be introduced which uses the same hardware for both operations.
In section 2 digital multiplication and division will be discussed. As our new hardware uses DAC and ADC, these two mixed-signal blocks will be introduced briefly in the same section. In section 3 the main idea and system structure of our multiplier/divider will be presented and discussed. Section 4 consists of realizeability of proposed system and some comparisons about improved characteristics. Finally conclusion and further researches are placed in section 5.

BASIC CONCEPTS

As mentioned before in this section multiplication/division and their advantages or disadvantages will be studied and then DAC and ADC as two basic block of new hybrid multiplier/divider will be introduced in brief.

Multiplication

To multiply binary digits two general solutions exist:
Software Based. In this method, processor has to use existing adder and shifter to multiply operand and no extra hardware exist for this operation. One of the simplest general solution is to shift one of operands (operand B in an operation of $M = A \times B$) right, if the carry out is one then A will be added to M. In second step A will be multiplied by two (i.e. left shift) and B will shift right again and if the carry out is one, new A (multiplied by 2) will be added to M and so on [10].
Simply it can be understood that this iterative solution consumes lots of time, as an example for an n-bit operands, $2n$ shifts and n adds should be done. So considering one clock time length for shifting and adding, Multiplication Time (M_T) duration is at least:

$$M_T = 2n \cdot T + n \cdot T = 3n \cdot T \quad (1)$$

where T is the clock time length of processor.

[1] Field Programmable Gate Array
[2] Field Programmable Analog Array

It's obvious that this duration is not adequate, especially when n increases, but this disadvantage accompanies important advantages; simplicity in hardware design and low die space required by hardware which means low cost.

Hardware based. In fact this method also uses the former concept to do multiplication but extra hardware is designed to accelerate this operation.

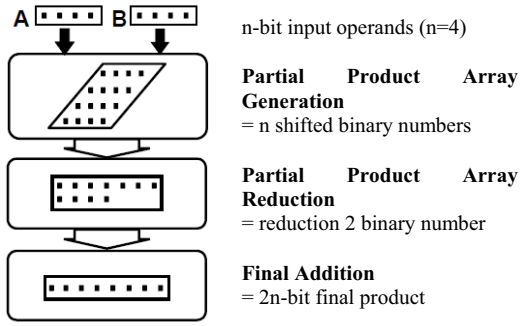

Fig. 1. Digital multiplication flow[11]

As shown in Fig. 1 the operation is divided in three parts. Main optimization may be done in these partitions, such as Wallace Tree in Partial Product Reduction and Booth Recoding in Partial Product Reduction/Generation and so on [11].

In this method CPU clock time length is not critical in determining the time length of multiplication, in fact number of stages used by structure and delay of each stage and basic elements determines the calculation time length.

Basic elements' delay is limited by technology but other delays may be reduced by better circuit design for each stage or better system design leading in fewer stages.

By the way, speed advantage of this method accompanies disadvantages; large space usage of this method which means cost increase and power consuming which is due to huge hardware structure.

Division

First of all it should be mentioned that division here stands for integer division without floating point. Second; analogous to multiplication, division has two general solutions:

Software Based. When the division had to be done by software a simple usual method is sequential subtracts; divisor is subtracted from dividend number and a counter will count up each subtraction, and the action keeps on while the dividend is greater than the divisor.

In this algorithm the division time length may change upon to operands but it may also keep on for n cycles (for n-bit operands). So considering one cycle for each subtraction, one for counting up and one cycle for comparison, the task may take long a 3n.T times

$$D_T = n \cdot T + n \cdot T + n \cdot T = 3n \cdot T \qquad (2)$$

So again like the software multiplication, advantage is compactness and disadvantage is time consuming.

One of the other methods which has hardware implementation too is mentioned in the next part. [12]

Hardware based. Basic principle for many applications again a sequence of iterations until desired precision, called 'sequential division' [13] and the most prevalent one; 'radix-r' that runs in constant number of cycles. The most efficient method to speed it up is named 'SRT division' which as the main drawback needs large look-up tables which grow quadratically with increasing radix. Next drawback is possible few bit lost occurrences during the development process of a large look-up table, which is popular in Pentium Flaw [14].

Another approach is data-dependent dividers that execute in variable time. The basic principle is to skip all redundant operations and carry out only shifts as long as there are leading zeros of the remainder or divisor depending on the algorithm used [15]. This method doesn't need look-up table but its speed is currently not adequate to be used in pipelined architectures.

This method generally can be implemented also by software but considerations and some hardware implementation speeds it up. For example; running each iteration of algorithm in one cycle. In [16] more description about shifting, details of algorithm and also their improvements may be found. By the way their offered structure is too complicated even in system level. Fig. 2 is inserted just as an overview illustration of system complexity, although details are too long to be mentioned in this paper.

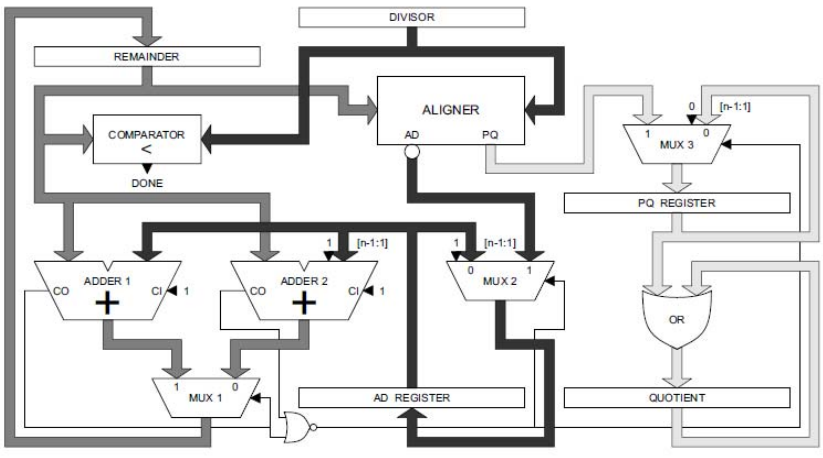

Fig. 2. A Hybrid-Aligner divider proposed by [16] to show system complexity

DAC

Digital to Analog Converters (DAC) as it's clear from their name, is a system which change a Digital discrete signal to an Analog continuous one.

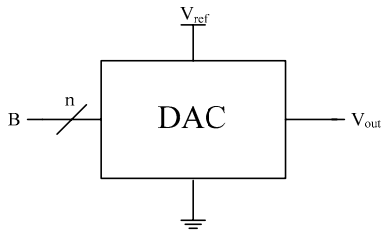

Fig. 3. An n-bit DAC with V_{ref} as the Reference Voltage

In fact when digital (processed or not processed) data are going to be used in an Analog system, they should be converted and DAC does this task.
A digital *n*-bit binary number (B) produces a voltage level at the output of DAC which obeys Eq. (3).

$$V_{out} = \frac{V_{ref}}{2^n} B \tag{3}$$

V_{ref} is the reference voltage which determines the steps of output voltage level.
As it's clearly understood from the Eq.(3) the output is quantized, if the B is not changing, then the output level is an Analog continuous signal relevant to input but if B is changing (for example Sinusoidal, ramp or etc.) then the output have to be continuous and not quantized. To satisfy this condition an Ideal Low Pass Filter should be located at the output. It's the bottleneck in DACs because realizing an ideal low pass filter circuit is not possible and getting closed to an Ideal LP filter costs a lot. [17]
It will be shown later, how this problem is not parted in multiplier/divider system presented in this paper.

ADC

Analog to Digital Converter (ADC) as it's obvious from its name, performs the inverse action of DAC.

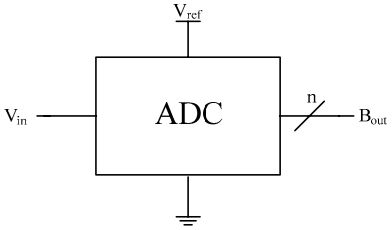

Fig. 4. An n-bit ADC with V_{ref} as the Reference Voltage

Interfacing analog sensors or when analog signals have to be processed by digital processors, this system translates the analog signals for digital parts. Eq. (4) shows the translation rule.

$$B_{out} = 2^n \frac{V_{in}}{V_{ref}} \tag{4}$$

In fact Analog signal of input is first sampled and hold, it means a voltage level, then it will be compare by the different levels which are determined by V_{ref}, and output digit will be produced.
The problems in ADC are more than DAC. First bottleneck in ADC is the input part, where if the sampling frequency is not adequate, results in aliasing problem. To decrease this problem, increasing the sampling frequency, up & down sampling and input LP filters may be used.
Second bottleneck is in conversion part, where the voltage levels have to be determined. In real world if a continuous analog signal is given to input node, holding the instantaneous voltage of input as the voltage level means missing the changes (i.e. some data) while next sampling. This problem also may decrease if the sampling frequency is increased although frequency limitations shouldn't be forgotten.
And the third bottleneck is when output is going to be produced. When the voltage levels are compared with quantization levels, it's a very little probability that these two levels completely matches (in probability theory of continues variables, it is zero). So the voltage will be rounded to one of quantized levels and it means again some data loss or some noise inserted. This problem will be decreased by adding quantization levels (more bits in the output to describe the input better and more exact) [17].
By the way, decreasing all these problems means in more complex design, more die space used and finally more production cost, but it will be shown later, how some of these problems won't be important for this special use and how others may be decreased with lower cost respect to general purpose ADCs.

PROPOSED SYSTEM

The main idea arises from the scaling capability that Reference Voltage (V_{ref}) prepares in both ADC and DAC.
To describe the idea, design of a multiplier with above capability of converters will be done step by step and at the end it'll be shown simply that the same device acts as a divider too.

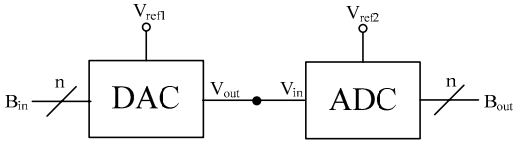

Fig. 5. B_{out} as an scaled version of B_{in}

First, taking a look to Fig. 5, Eq. (5) can be written out.

$$\left.\begin{aligned} V_{out} &= \frac{V_{ref1}}{2^n} B_{in} \\ B_{out} &= 2^n \frac{V_{in}}{V_{ref2}} \end{aligned}\right\} \Rightarrow B_{out} = \frac{V_{ref1}}{V_{ref2}} B_{in} \tag{5}$$

Now, having a multiplier, we should control the V_{ref1} Voltage by a digital number. It means using another DAC to supply V_{ref1} for DAC1, as illustrated in Fig. 6.

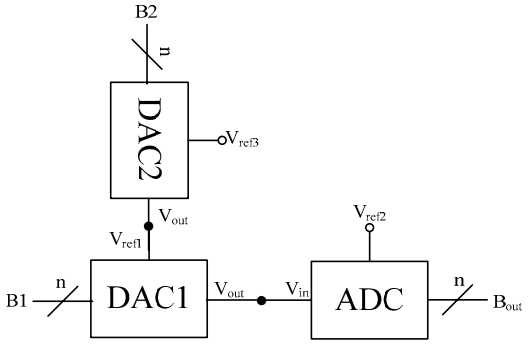

Fig. 6. $B_{out} = B_1 . B_2 / K$

Eq. (6) clarifies the relation between B_{out} , B_1 and B_2 .

$$\left.\begin{aligned} B_{out} &= \frac{V_{ref1}}{V_{ref2}} B_1 \\ V_{ref1} &= \frac{V_{ref3}}{2^n} B_2 \end{aligned}\right\} \Rightarrow B_{out} = \frac{V_{ref3}}{V_{ref2}} \frac{1}{2^n} B_1 . B_2 \qquad (6)$$

Even though the product of B_1 and B_2 is appeared in output, but there is a problem; it's the 2^{-n} coefficient which means great undesirable attenuation and having not true output. To compensate this attenuation it seems good to choose reference voltage as Eq. (7) says.

$$V_{ref3} = 2^n \times V_{ref2} \qquad (7)$$

It yield in unity coefficient for input products but two potential problems may occur. First; Integrated Circuits usually (especially about digital ones it could be said always), use single supply voltage (at least it's more preferred) which both V_{ref2} and V_{ref3} have to be made out of this voltage. The great difference between these two reference voltages makes it very difficult to have good reference voltages, especially for V_{ref2}. The problem gets harder; having in mind that logical circuits are bounded in 5 volts and this voltage is getting continuously lower.

The second problem is the circuit which wants to do this scaling of reference voltages. Usually these circuits have changes with temperature change or supply instabilities which may affect each reference voltage different to other one and in sequence the unity coefficient.

To solve both these problems, another DAC will be added to the system.

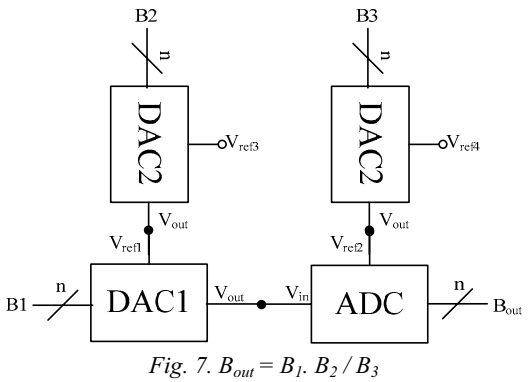

Fig. 7. $B_{out} = B_1 . B_2 / B_3$

Simple manipulations for new system result in Eq. (8).

$$V_{ref2} = \frac{V_{ref4}}{2^n} B_3 \Rightarrow B_{out} = \frac{V_{ref3}}{V_{ref4}} \left(\frac{B_1 . B_2}{B_3}\right) \qquad (8)$$

This addition solves these problems; V_{ref2} and V_{ref1} are supplied by the same device and so each offset may definitely lowered by symmetry. Next; to have unity coefficient V_{ref3} have to be equal to V_{ref4} and it means every possible change of reference voltages doesn't affect coefficient.

As mentioned before, we had focused on multiplier but Eq. (8) shows that simply we reached to a simultaneous multiplier/divider which brings us the capability of multiplication and division by the same device.

It should be mentioned it's not forgotten that the product of two n-bit numbers may lead in a *2n*-bit number, but if we consider the output as a *2n*-bit digital number and inputs as *n*-bit ones, Eq. (8) may be written as Eq. (9).

$$B_{out} = 2^n \frac{V_{ref3}}{V_{ref4}} \left(\frac{B_1 . B_2}{B_3}\right) \qquad (9)$$

Not equal reference voltages have been studied formerly, so to solving this problem B_3 had to be equal to 2^n and this means missing the capability of division.

Another benefit of having the same number of bits in converters is having more compatibility. Therefore it would be better to design *2n*-bit converters to have *n*-bit multiplier and divider (or equally *n*-bit for n/2-bit operands).

REALIZATION & SOME COMPARISONS

Realizing ADCs and DACs are not new concepts. Different converters with various characteristics [18] may be realized.

To have an overview; a comparison between our system (using some realized converters) and some realized multipliers and dividers will be inserted in Table 1. To have comparable numbers some normalizations have been done, e.g. as it's clearly shown in the table, the technologies used by devices are different. To normalize the areas, knowing that area size is a product of technology size, the areas are dividend by technology size.

Table 1 declares that digital system is about 2.5 times larger in die area respect to new hybrid digital analog mixed system. It means good improvement but that's not all. Die areas mentioned in the table belong to realized ADC and DAC which are designed for general purpose use, but in the specific usage they can be simplified and so more compact.

DAC used in this system doesn't need the output LP filter, because we're not to create a continuous signal and just the output level is important which will be scaled by ADC.

TABLE 1. An approximated die area comparison between new overall system & digital multiplier, divider

	Area [mm^2]	Tech. [µm]	Norm. Area	Norm. Area	Tech. [µm]	Area [mm^2]	
ADC [19]	0.25	0.18	1.39	0.52	0.25	0.13	**Multiplier [21]**
DAC [20]	0.01	0.18	0.06	3.5	1.2	4.2	**Divider [22]**
Overall Hybrid System			1.57	4.02	**Overall Digital System**		

The ADC will simplify a lot too, input filter is not needed and over sampling and such problems won't occurred because the input is not continuous signal and if it's considered some frequency for the input changes (output of DAC) it's constant and known, so no aliasing may occur. Also the sample and hold part is not needed too, because the input is hold automatically by DAC.
And the final bottleneck of ADCs is not serious too, because the output of DAC is a quantized signal and quantization noise is not so meaningful.
A unique capability of this architecture is accepting 3 operands at once, i.e. both Multiplication and Division can be simultaneously done. This means a great speed up when these two operations are needed to be done on an operand.
About the speed of new system and digital systems; the digital systems consist of lots of staged from the input to the output and in fact the speed of system is limited by stages' delay and is not so improvable by circuit designs. But ADCs (which are too slower from DACs) can, and are improving in speed day by day. Especially if the sample and hold section eliminates, an important speed up is expected.
For example, ADC's have reached too 1.6 GSam./Sec [23] but digital multiplier achieved 256 MMul./sec [24].

CONCLUSION

In this paper a new digital multiplier/divider system was presented which uses hybrid analog-digital circuits.

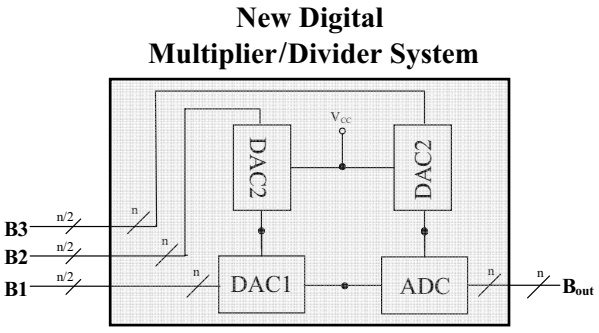

Fig. 8. New Proposed Multiplier/Diver Architecture

The system is too simple while digital ones are complex.
Both operations are done with the same device while digital ones haven't this capability.
Another unique property of this system is having three operands and performing both operations simultaneously.
The system potentially may lead in 2.5 times (or more) smaller size in die area.
Higher speed is potentially expected too.

FURTHER RESEARCHES

The system was not circuitry implemented, while special purpose devices may lead in better performance. So implementing the system circuitry may give out better criterions for comparisons.
Comparison between size, speed and power consumed by specialized circuit should be studied.
Development of system, performances and potential problems may be studied.
Other hybrid systems which use both benefits of Analog and Digital systems may be developed.

THE AUTHORS

Nima Taherinejad & Hadi Shahriar Shah-hosseini
Electrical Engineering Department, Iran University of Science & Technology (IUST), IRAN
ntaherinejad@ee.iust.ac.ir
hshsh@iust.ac.ir

REFRENCES

[1] Yu-Tsun Chien and colleagues, "SPEED: Synthesis of High performance Large Scale Analog/Mixed Signal circuits", 2005.
[2] Kate L. Kraver and colleagues, "A mixed-signal sensor interface micro instrument", Sensor and Actuators A, p266-277, 2001.
[3] Yasunari, J. & Inoue, T. & Tsuneda, A., "A CMOS continuous-time FPAA analog core using automatically-tuned MOS resistors", 47th Midwest Symposium on Circuits and Systems, 2004.
[4] G.G.E Gielen, R.A. Rutenbar, "Computer-aided design of analog and mixed-signal integrated circuits", Proc. IEEE, vol. 88, issue 12, pp.1825-1854, December 2000.
[5] A.R.S. Romariz and colleagues, "Design of a Hybrid Digital-Analog Neural Co-Processor fo Signal Processing", IEEE Proceedings of Euro Micro-22, p513-519, 1996.
[6] Gamze Erten and Fathi M. Salam, "Programmable Hybrid Co-Processor for Real-Time Image understanding", IEEE Proceedings of ASILOMAR-29, p 885-888, 1996.
[7] David A. Martin & Hae-Seung Lee & Ichiro Masaki, "A Mixed-Signal Array Processor with Early Vision Applications", IEEE JOURNAL OF SOLID-STATE CIRCUITS, VOL. 33, NO. 3, MARCH 1998.
[8] Rafal Karakiewicz & Roman Genov & Adeel Abbas & Gert Cauwenberghs, "175 GMACS/mW Charge-Mode Adiabatic Mixed-Signal Array Processor", IEEE Symposium on VLSI Circuits Digest of Technical Papers, 2006.

[9] Shi Z.M. and colleagues, "A 2.7 V mixed signal processor for CDMA/AMPS cellular phones", IEEE Radio Frequency Integrated Circuits (RFIC) Symposium, 1999.
[10] Patel M.R. & Benette K.H., "Analysis of speed of a binary multiplier using a variable number of shifts per cycle", The Computer Journal, Vol. 19, No3, pp 254-257, 1976.
[11] Pascal Constantin Hens Meier, "Analysis and Design of Low Power Digital Multipliers", PhD thesis, Carnegie Mellon University, Pennsylvania, August 1999
[12] Barry. B. Brey, "Z80 microprocessor",1988.
[13] Mi Lu, "Arithmetic and Logic in Computer Systems", Wiley-Interscience, Hoboken, New Jersey, 2004.
[14] H. P. Sharangpani and M. L. Barton, Statistical Analysis of Floating Point Flaw in the Pentium Processor, Technical Report, 1994.
[15] Michael D. Ciletti, Advanced Digital Design with the Verilog HDL, Prentice Hall, Upper Saddle River, New Jersey, 2003.
[16] Rainer Trummer & Peter Zinterhof & Roman Trobec, "A High-Performance Data-Dependent Hardware Divider", Salzburg University, 2005.
[17] Oppenheim, "Discrete Time Signal Processing", 1999.
[18] Randall L. Geiger & Phillip E. Allen & Noel R. Strader, "VLSI Design Techniques for analog and digital circuits", McGraw Hill Pub, 1990.
[19] Cheng Chen & Junyan Ren, "An 8-bit 200-MSample's Folding and Interpolating ADC in 0.25 mm^2", Analog Integrated Circuits and Signal Processing, 2006.
[20] Greenley B.R. and colleagues, " 1.4V 10b CMOS DC DAC in 0.01mm/sup 2/", IEEE SOC Conference, 2003.
[21] Alexander Goldovsky and colleagues, "Design and implementation of a 16 by 16 Low-Power Two's Complement Multiplier", 2000.
[22] Hongge Ren and colleagues, "Design of a 16-bit CMOS divider/square-root circuit", The Twenty-Seventh Asilomar Conference on Signals, Systems and Computers, 1993.
[23] Robert Taft and colleagues, "Unique 1.6 GSPS CMOS 8-bit 1.8V ADC Delivers 7.26 ENOB Past Nyquist", Euro DesignCon, 2004.
[24] Amir R. Attarha & Mehrdad Nourani & M.Zakeri, "High Performance Low-Power Signed Multiplier", 1998.