ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Cardiac Design Using Multiple Constant Multipliers with Shared Input Methodology

Penubala Nagendra Babu
PG Scholar,
Dept of ECE,
Global College of Engineering & Technology,
Kadapa, YSR (Dt), AP, India.

D.C.Vinod Kumar
Assistant Professor,
Dept of ECE,
Global College of Engineering & Technology,
Kadapa, YSR (Dt), AP, India.

ABSTRACT:

CORDIC or CO-ordinate Rotation Digital Computer is a fast, simple, efficient and powerful algorithm used for diverse Digital Signal Processing applications. Primarily developed for real-time airborne computations, it uses a unique computing technique [7] which is especially suitable for solving the trigonometric relationships involved in plane co-ordinate rotation and conversion from rectangular to polar form.

It comprises a special serial arithmetic unit having three shift registers, three adders/subtractors, Look-Up table and special interconnections.Proposed methodology was compared with the actual value stotest for consistency and the percentage of accuracy was established.Power consumption and FPGA resourceutilizationwere observed. The results obtained were discussed.

I.INTRODUCTION:

The FPGA platform is much better choice for CORDIC algorithm implementation since it combines the flexibility of microprocessors as well as the speed and computational power of ASICs. The CORDIC (Coordinate Rotation Digital Computer) algorithm uses planar rotation and vectoring to compute elementary trigonometric functions when assigned with proper initial conditions. It is one of the iterative algorithm explored by Jack E. Volder [1] and later clearly refined by Walther [2] and others. CORDIC algorithm is very popular since it uses only shifts and adds to perform number of functions including certain trigonometric, vector rotations, hyperbolic, logarithmic functions. In communication applications it is widely used to implement universal modulator [3], demodulator [4]. CORDIC algorithm is used in various applications that includescalculators, mathematical coprocessor units, clock recovery circuits, waveform generators.

There are two different modes of CORDIC algorithm as rotation and vector mode. In the rotation mode, CORDIC is used for converting a vector from polar form to rectangular form and vector mode makes the reverse operation. The objective of this paper is to design, analyze and compare how an FPGA based unrolled CORDIC performs when multiplexers are used instead of shifters and adders with and without pipelining [5] [6] [7] [8].

BASICS OF CORDIC ALGORITHM:

The CORDIC algorithm is one of the iterative method toperform vector rotations for arbitrary angles using shifts and adds. Planar rotation for any vector A of (Xj, Yj) can be defined in matrix form as:

$$\begin{bmatrix} Xj \\ Yj \end{bmatrix} = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix} \begin{bmatrix} Xi \\ Yi \end{bmatrix} \qquad (1)$$

With a small rearrangement, the stepwise rotation is performed by

$$\begin{bmatrix} X_{n+1} \\ Y_{n+1} \end{bmatrix} = cos\theta_n \begin{bmatrix} 1 & -tan\theta_n \\ tan\theta_n & 1 \end{bmatrix} \begin{bmatrix} Xn \\ Yn \end{bmatrix} \qquad (2)$$

The angle parameter for each step will be

$$\theta_n = \arctan\left(\frac{1}{2^n}\right) \qquad (3)$$

II.RELATED WORK:

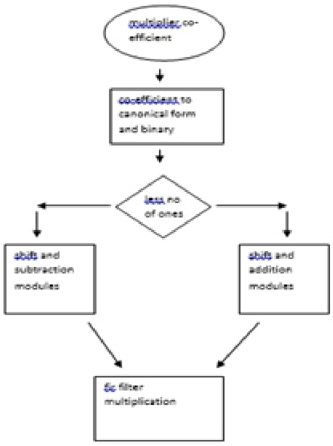
In the FPGA design, some special functions often need to be implemented, such as using FPGA to achieve some digital signal processing algorithms. If the algorithm uses a non-common (beyond) algebraic functions, we can use Taylor series to approximate this function, then the problem is simplified into a series of multiplication andaddition operations, but the program is complex and the consumption of resources is so huge that it is almost not feasible [1].

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

A more effective method is based on the coordinate rotation digital computation(Coordinate Rotational Digital Computing, CORDIC)[2]. CORDIC is presented by the J. D. Volder in 1959 and first used in navigation systems, making the vector rotation and the directional computing not need to do checking list of trigonometric functions, multiplication, square root and inverse trigonometric functions and other complex operations [3]. The basic idea of the algorithm is through a series of the continued deflection angle, fixed and associated with base number of calculate, to approximate the required rotation.

Because of its basic operation unit being only shift and add and subtract, which laid a good foundation for the CORDIC algorithm combines with the rapid development of ultra large scale integrated circuit(Very Large Scale Integrated circuits, VLSI)technology. A significant reduction in FPGA resources makes these algorithms easier to be achieved in hardware, and thus meet the requirements of designer [4] [5]. People pay more attention to its advantages, and it is widely used in computing real-time signal processing of high quality requirements, image processing and so on.

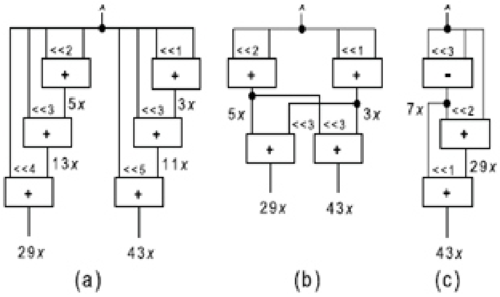
III.IMPLEMENTATION:


multiplier co-efficient
co-efficient to canonical form and binary
less no of ones
shift and subtraction modules
shift and addition modules
fir filter multiplication

For the shift-adds implementation of constant multiplications, an easy methodology, usually called digit primarily based secret writing, ab initio defines the constants in binary. Then, for every “1” within the binary illustration of the constant, consistent with its bit position, it shifts the variable and adds up the shifted variables to get the result. As a straightforward example, think about the constant multiplications 29x and 43x. Their decompositions in binary are listed as follows:

$$29x = (11101)_{\text{bin}}x = x \ll 4 + x \ll 3 + x \ll 2 + x$$
$$43x = (101011)_{\text{bin}}x = x \ll 5 + x \ll 3 + x \ll 1 + x$$

which requires six addition operations as illustrated



However, the digit-based secret writing technique doesn’t exploit the sharing of common partial product, that permits nice reductions within the range of operations and, consequently, in space and power dissipation of the MCM style at the gate level. Hence, the elemental optimisation drawback, known as the MCM drawback, is outlined as finding the minimum range of addition and subtraction operations that implement the constant multiplications. Note that, in bit-parallel style of constant multiplications, shifts is accomplished victimisation solely wires in hardware while not representing any space value. The algorithms designed for the MCM drawback is classified in 2 classes: common subexpression elimination (CSE) algorithms [7]–[9] and graph-based (GB) techniques [10]–[12]. The CSE algorithms at the start extract all attainable subexpressions from the representations of the constants once they ar outlined beneath binary, canonical signed digit (CSD) [7], or stripped signed digit (MSD) [8].

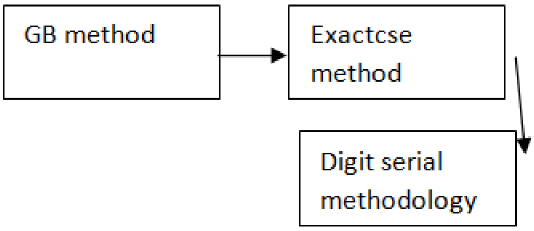
Then, they notice the “best” subexpression, usually the foremost common, to be shared among the constant multiplications. The GB ways aren’t restricted to any explicit range illustration and think about a larger variety of other implementations of a continuing, yielding higher solutions than the international intelligence agency algorithms, as shown in [11] and [12].Returning to our example in Fig. 2, the precise international intelligence agency algorithmic program of [9] provides an answer with four operations by finding the foremost common partial product 3x = (11)binx and 5x = (101)binx once constants area unit outlined beneath binary, as illustrated in Fig. 2(b). On the opposite hand, the precise GB algorithmic program [12] finds an answer with the minimum variety of operations by sharing the common partial product 7x in each multiplications, as shown in Fig. 2(c). Note that the partial product 7x = (111)binxcan not be extracted from the

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

binary illustration of 43x within the precise international intelligence agency algorithmic program [9].However, of these algorithms assume that the input file x is processed in parallel. On the opposite hand, in digit-serial arithmetic, the info words area unit divided into digit sets, consisting of d bits, that area unit processed one at a time [13]. Since digit serial operators occupy less space and area unit freelance of the info word length, digit-serial architectures supply different low quality styles in comparison to bit-parallel architectures. However, the shifts need the utilization of D flip-flops, as hostile the bit-parallel MCM style wherever they're free in terms of hardware. Hence, the high-level algorithms ought to take under consideration the sharing of shift operations similarly because the sharing of addition/subtraction operations in digit-serial MCM style. what is more, finding the minimum variety of operations realizing AN MCM operation doesn't invariably yield AN MCM style with best space at the gate level [14].

Hence, the high-level algorithms ought to contemplate the implementation value of every digit-serial operation at the gate level.Since there area unit still instances that the precise international intelligence agency algorithmic program cannot handle, we tend to describe the approximate GB algorithmic program [16] that iteratively finds the "best" partial product that results in the best space in digit-serial MCM style at the gate level. This paper conjointly introduces a {computer-aided style|CAD|software|softwaresystem|softwarepackage|package} (CAD) tool referred to as SAFIR that generates the hardware descriptions of digit-serial MCM operations and FIR filters supported a design design and implements these circuits employing a business logic synthesis tool. In SAFIR, the digit-serial constant multiplications will be enforced beneath the shiftadds design, and can also be designed victimisation generic digitserial constant multipliers.

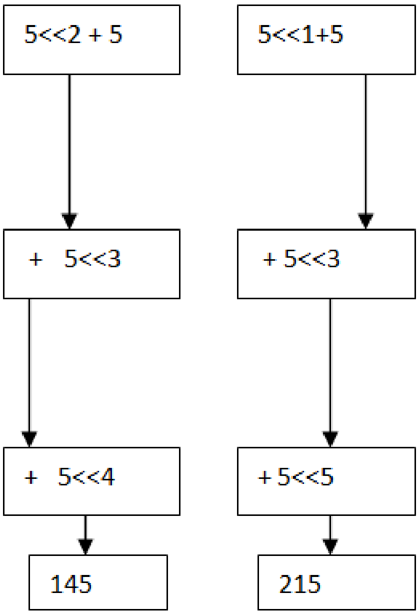
Design Flow:



Numerical Examples

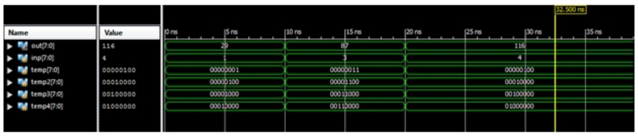
constant =29 (11101)
constant =43(101011)
input =5
theoretical
29*5=145
43*5=215
shift and add:
(11101)*5=
5<<4 + 5<<3 + 5<<2 + 5
=> 80 + 40 + 20 + 5
=> 145
(101011)*5
5<<5 + 5<<3 + 5<<1 + 5
=>160+40+10+5
=>215

Partial Product Sharing



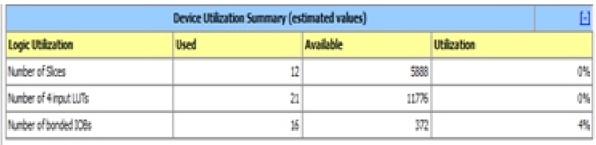
IV.RESULTS

Constant Rotator Result



```
Name          Value      0 ns   5 ns   10 ns   15 ns   20 ns   25 ns   30 ns   35 ns
                                                                  32.500 ns
out[7:0]      116          29            87                  116
inp[7:0]      4            1             3                   4
temp[7:0]     00000100     00000001      00000011            00000100
temp2[7:0]    00010000     00000100      00001100            00010000
temp3[7:0]    00100000     00001000      00011000            00100000
temp4[7:0]    01000000     00010000      00110000            01000000
```

Area



Device Utilization Summary (estimated values)

Logic Utilization	Used	Available	Utilization
Number of Slices	12	5888	0%
Number of 4 input LUTs	21	11776	0%
Number of bonded IOBs	16	372	4%

CONCLUSION:

It was shown that the conclusion of digit-serial FIR filters beneath the shift-adds design yields vital space reduction in comparison to the filter styles whose number blocks square measure enforced mistreatment digit-serial constant multipliers. it's discovered that a designer will notice the circuit that matches best in AN application by dynamic the digit size.

REFERENCES:

[1] L. Wanhammar, DSP Integrated Circuits. New York: Academic, 1999.

[2] C. Wallace, "A suggestion for a fast multiplier," IEEE Trans. Electron. Comput., vol. 13, no. 1, pp. 14–17, Feb. 1964.

[3] W. Gallagher and E. Swartzlander, "High radix booth multipliers using reduced area adder trees," in Proc. Asilomar Conf. Signals, Syst. Comput., vol. 1. Pacific Grove, CA, Oct.–Nov. 1994, pp. 545–549.

[4] J. McClellan, T. Parks, and L. Rabiner, "A computer program for designing optimum FIR linear phase digital filters," IEEE Trans. Audio Electroacoust., vol. 21, no. 6, pp. 506–526, Dec. 1973.

[5] H. Nguyen and A. Chatterjee, "Number-splitting with shift-and-add decomposition for power and hardware optimization in linear DSP synthesis," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 8, no. 4, pp. 419–424, Aug. 2000.

Author's Profile:

Penubala Nagendra Babu, Pursuing M.Tech (**VLSI**), Global College of Engineering & Technology, Kadapa, Andhra Pradesh, India.

D.C.Vinod Kumar, M.Tech, Assistant Professor at Global College of Engineering & Technology, Kadapa, Andhra Pradesh, India.