iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 7 **Issue:** IV **Month of publication:** April 2019

DOI: https://doi.org/10.22214/ijraset.2019.4055

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 7 Issue IV, Apr 2019- Available at www.ijraset.com

Rounding Multiplier to improve the efficiency using Brent Kung Adder

Arul. R
Assistant Professor at Department Of ECE, Dhirajlal Gandhi College of Technology, Salem.

Abstract: ***In this paper, we propose an approximate multiplier to improve the efficiency over the traditional approximate multiplier, which is commonly based on rounding technique. In general, the approximate multipliers are used to achieve high performance with a huge trade-off in accuracy. The Brent-Kung adder, which is a pruned version of Kogge-Stone adder has a better accuracy trade-off. The proposed approximate multiplier, which uses the Brent-Kung adder shows better performances in terms of area, power consumption and speed than the Kogge-Stone based rounding approximate multiplier. The designed multipliers have been simulated using Cadence Encounter(R) RTL compiler with 180nm technology.The proposed multiplier based on gate pruning achieves power savings of 12% and speed to improved 74% compared to kogge stone based rounding multiplier. The proposed multiplier will be applied for real-time application in future.***
Index Terms: Approximate Multiplier, Kogge-stone adder, Brent-Kung Adder.

I. INTRODUCTION

Power Consumption and area reduction are the very important factor in Digital Systems. In some systems accuracy not important, it only depends on the performance of the operation, that system more favour for approximation logic. It is mainly used for Digital Signal Processing Applications (DSPs). It can use to improve the efficiency of this system such as reduction of delay, area, power and also improve the speed of the operation. In image processing applications, the human being can obtain valuable information from a little incorrect yield. The design of multiplier having low power consumption and low propagation delay results of great interest for the implementation of modern digital systems. In Applying the approximation to the arithmetic units can be performed at different design abstraction levels including circuit, logic, and architecture levels, as well as algorithm and software layers. The approximation may be performed using different techniques such as allowing some timing violations (e.g., the voltage over scaling or overclocking) and function approximation methods (e.g., modifying the Boolean function of a circuit) or a combination of them. In the category of function approximation methods, a number of approximating arithmetic building blocks, such as adders and multipliers, at different design levels have been suggested. In this paper, we focus on proposing a high-speed low power/energy yet approximate multiplier appropriate for error resilient DSP applications. The proposed approximate multiplier, which is also area efficient, is constructed by modifying the conventional multiplication approach at the algorithm level assuming rounded input values.This rounding-based approximate multiplier. The proposed multiplication approach is applicable to both signed and unsigned multiplications for which three optimized architectures are presented. The efficiencies of these structures are assessed by comparing the delays, power and energy consumptions, energy-delay products (EDPs), and areas with those of some approximate and accurate (exact) multipliers.

II. PROPOSED APPROXIMATE MULTIPLIER

A. Multiplication Algorithm of Rounding Multiplier

The proposed approximate multiplier is to make use of the ease of operation when the numbers are two to the power *n* (2^n). To elaborate on the operation of the approximate multiplier, first, let us denote the rounded numbers of the input of *A* and *B* by *Ar* and *Br* respectively.

The multiplication of *A* by *B* may be rewritten as

$$A \times B = (A_r - A) \times (B_r - B) + A_r \times B + B_r \times A - A_r \times B_r. \quad \rightarrow(1)$$

Where the key observation is that the multiplications of $A_r \times B_r$, $A_r \times B$, and $B_r \times A$ may be implemented just by the shift operation. In this operation $(A_r - A) \times (B_r - B)$ is omitted

And the remaining expressions has introduced some error but this error has acceptable. and hence it called approximation multiplier.

$$A \times B = A_r \times B + B_r \times A - A_r \times B_r \quad \rightarrow(2)$$

©IJRASET: All Rights are Reserved

Thus, one can perform the multiplication operation using three shift and two addition/subtraction operations. In this approach, the nearest values for *A* and *B* in the form of 2^n should be determined. When the value of *A* (or *B)* is equal to the $3 \times 2p-2$ (where *p* is an arbitrary positive integer larger than one), it has two nearest values in the form of 2*n* with equal absolute differences that are 2*p* and $2p-1$. While both values lead to the same effect on the accuracy of the proposed multiplier, selecting the larger one (except for the case of $p = 2$) leads to a smaller hardware implementation for determining the nearest rounded value, and hence, it is considered in this paper. It originates from the fact that the numbers in the form of $3 \times 2p-2$ are considered as do not care in both rounding up and down simplifying the process, and smaller logic expressions may be achieved if they are used
in the rounding up. The only exception is for three, which in this case, two is considered as its nearest value in the proposed approximate multiplier.

B. Hardware Implementation of Rounding Multiplier

The rounding block extracts the nearest value for each absolute value in the form of 2^n. It should be noted that the bit width of the output of this block is *n* (the most significant
bit of the absolute value of an *n*-bit number in the two's complement format is zero).
To find the nearest value of input *A*, we use the following equation to determine each output bit of the rounding
block.

$$A_r[n-1] = \overline{A[n-1]}.A[n-2].A[n-3] + A[n-1].\overline{A[n-2]}$$

$$A_r[n-2] = \left(\overline{A[n-2]}.A[n-3].A[n-4] + A[n-2].\overline{A[n-3]}.\overline{A[n-1]}\right)$$

$$A_r[i] = \left(\overline{A[i]}.A[i-1].A[i-2] + A[i].\overline{A[i-1]}\right).\prod_{i=i+1}^{n-1}\overline{A[i]}$$

$$A_r[3] = \left(\overline{A[3]}.A[2].A[1] + A[3].\overline{A[2]}\right).\prod_{i=4}^{n-1}\overline{A[i]}$$

$$A_r[2] = A[2].\overline{A[1]}.\prod_{i=3}^{n-1}\overline{A[i]}$$

$$A_r[1] = A[1]\prod_{i=2}^{n-1}\overline{A[i]}$$

$$A_r[0] = A[0].\prod_{i=1}^{n-1}\overline{A[i]} \quad \rightarrow(3)$$

In the proposed equation, A_r [*i*] is one in two cases. In the first case, *A*[*i*] is one and all the bits on its left side are zero while *A*[$i-1$] is zero. In the second case, when *A*[*i*] and all its left-side bits are zero, *A*[$i-1$] and *A*[$i-2$] are both one. Having determined the rounding values, using three barrel shifter blocks, the products $A_r \times B_r$, $A_r \times B$, and $B_r \times A$ are calculated. Hence, the amount of shifting is determined based on $\log_2{}^{Ar} - 1$ (or $\log_2{}^{Br} - 1$*)* in the case of *A* (or *B*) operand. Here, the input bit width of the shifter blocks is *n*, while their outputs are 2*n*.

C. Kogge-Stone Adder Vs Brent Kung Adder

Kogge Stone Adder is one type of parallel prefix form carry look ahead adder. It generates carry in O (logn) time and is widely considered as the fastest adder and is widely used in the industry for high-performance arithmetic circuits. But Kogge stone adder considers more area hence delay are more.

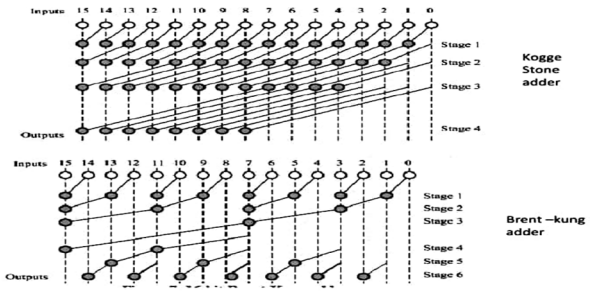


Fig 1. Kogge-Stone adder vs Brent Kung Adder

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 7 Issue IV, Apr 2019- Available at www.ijraset.com

So, we are using Brent Kung Adder. Brent-Kung has maximum logic depth and minimum area and power consumption are less. Kogge Stone based rounding multiplier used for high performance system such as DSP Applications. but DSP system main important factor are power consumption and Speed. In this Kogge Stone Adder used in rounding multiplier, it occupied more area and hence power consumption are more. So it can be replaced by Brent Kung Adder. It has less area and power consumption is less.

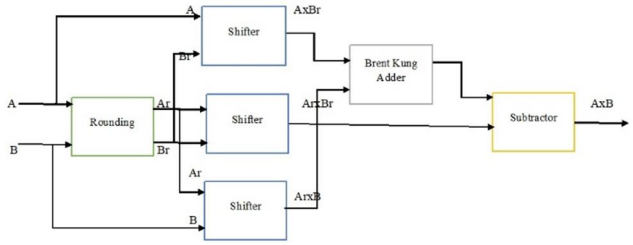


Fig.2 Block Diagram for Brent-Kung Adder based Rounding Multiplier (Proposed Multiplier)

D. Error Analysis of Rounding Multiplier

Rounding multiplier, which originate from omitting the term $(Ar - A) \times (Br - B)$ from the accurate multiplication of $A \times B$, are the same. Hence, the error is

$$error(A,B) = \frac{(A_r - A)(B_r - B)}{AB} \quad \rightarrow (4)$$

Assuming *Ar* and *Br* are equal to 2^n and 2^m, respectively, the maximum error occurs when *A* and *B* are equal to 3×2^n and 3×2^m, respectively.

In this case, both *Ar* and *Br* have the maximum arithmetic difference from their corresponding inputs.

E. Image Processing Applications

In most cases neighbouring image edges show Intensity discrepancies which are undesirable. These variations in intensity are present even when cross-correlation is almost perfect to the eye. In order to eliminate such effects, a blending algorithm is applied. Using the blending method, which applies changes in intensity, makes quantitative analysis of intensities invalid. The advantage of using blending method in improving visual quality of the composite image and making the edges invisible. In the overlapped area the image blending algorithm calculates the contribution of the new image and the composite image at every pixel. The blended image consists of pixels:

$N(x, y) = \alpha\, I(x, y) + (1-\alpha)C(x, y)$

Where, C(x, y) is the composite image pixel (before placing the new image), I(x, y) is the new image pixel and N(x, y) the new composite image pixel (with new image added). The blending algorithm minimises effects of intensity variations, removes the edges and improves the cross-correlation.

The proposed multiplier implement on image merging applications shown in fig[4], fig[5]. Fig[3],fig[6] is the block diagram of image merging application to implement on Simulink.Fig[7] is the color image to apply the image merging applications.

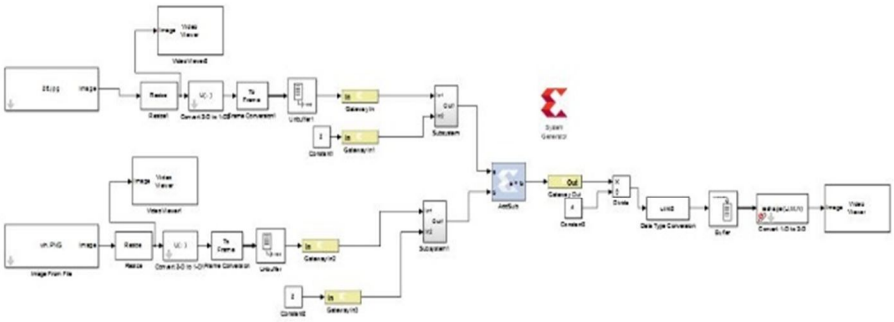


Fig 3. Brent Kung based Rounding Multiplier to implement in Simulink

©IJRASET: All Rights are Reserved

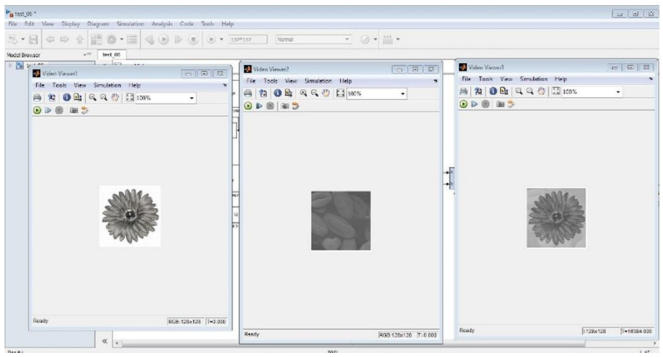


Fig 4.Brent Kung Adder based Roun;ding Multiplier for Image Merging Application. (Example : 1)

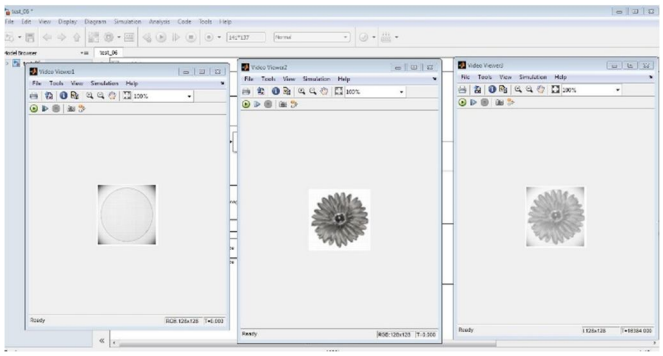


Fig 5. Brent Kung Adder based Rounding Multiplier for Image Merging Application. (Example: 2)

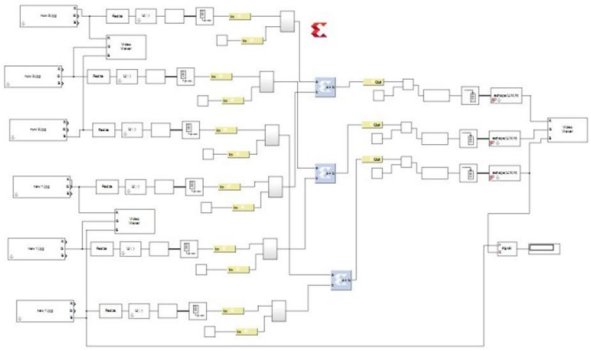


Fig 6. Brent Kung based Rounding Multiplier to implement in Simulink (Block Diagram for Color Image)

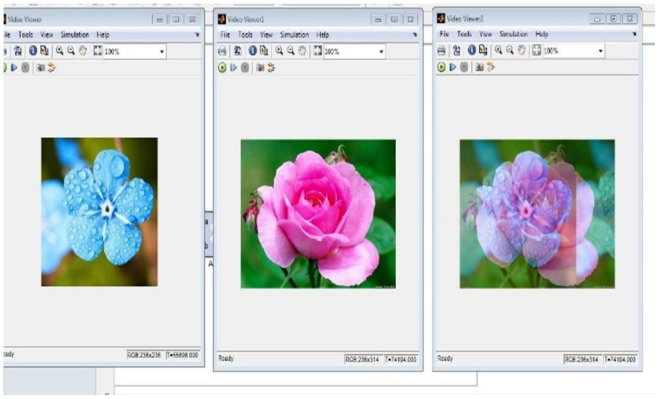


Fig 7.Brent Kung Adder based Roun;ding Multiplier for Image Merging Application. (Color Image)

©IJRASET: All Rights are Reserved

III. RESULTS AND DISCUSSION

The designed multipliers have been simulated using Cadence Encounter(R) RTL compiler with 180nm technology. To evaluate the efficiency of the proposed Brent Kung based rounding multiplier was compared with Kogge stone based rounding multiplier.

Table 1 .Performance Analysis of Rounding Mulitplier

Parameters	Kogge based Rounding Multiplier	Brent Kung based Rounding Multiplier	Efficiency
Report Power	1017070.1615nw	905446.659nw	12.32%
Report Area	29272(μm^2)	26312 (μm^2)	11.24%
Report Timing	317 ps	182 ps	74.17%

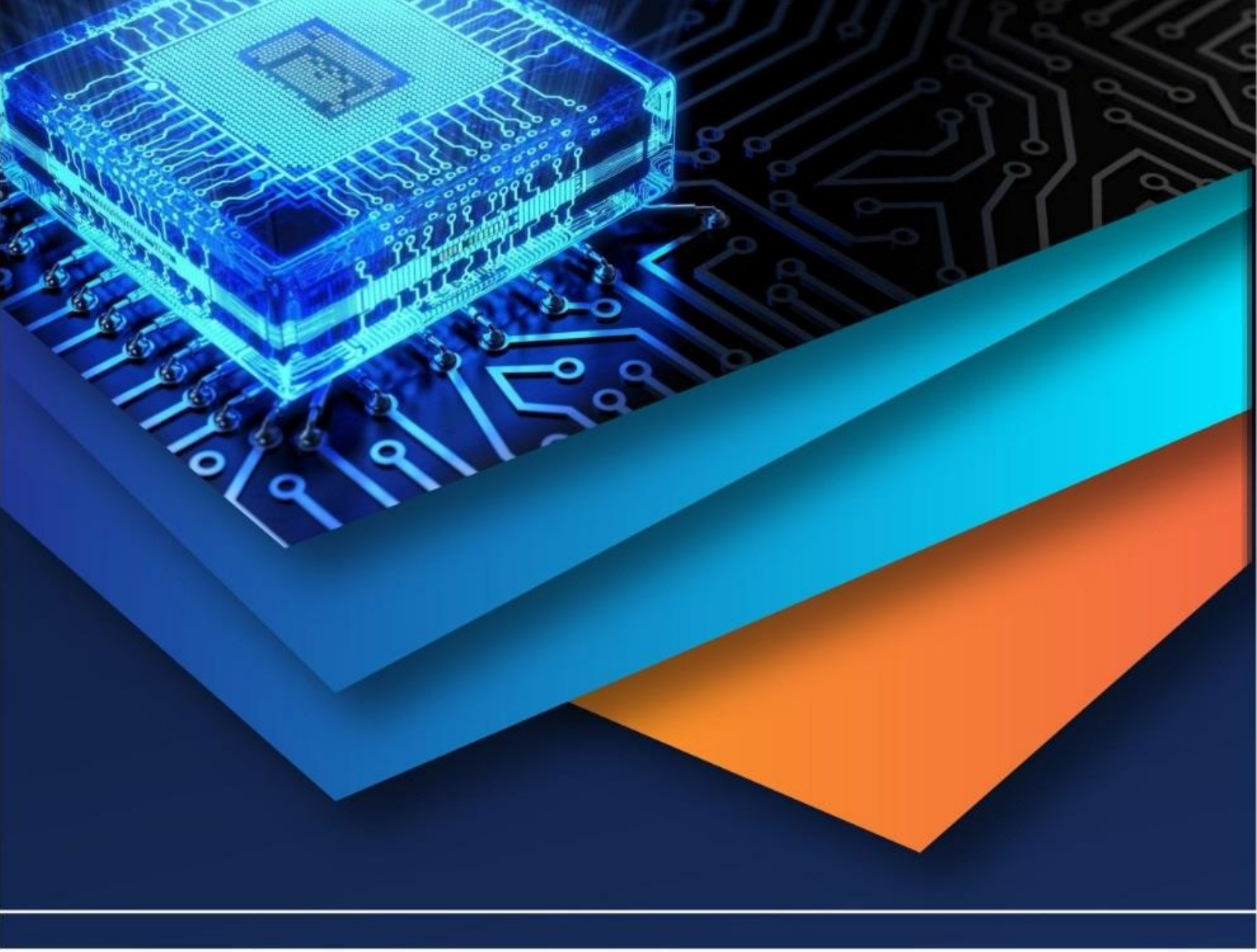
IV. CONCLUSION

In this paper, Rounding based on approximate multiplier using Brent-Kung Adder for the better performance. Kogge-Stone Adder Based Rounding multiplier has consumed more power and area but proposed approximate multiplier efficiency is improved in terms power, area and delay.

REFERENCES

[1] R.Zendegani, Mehdi Kamal, Milad Bahadori, Ali Afzali-Kusha, and Massoud Pedram. "RoBA Multiplier: A Rounding-Based Approximate Multiplier for High-Speed yet Energy-Efficient Digital Signal Processing" IEEE Trans. VLSI systems, Volume: 25, Issue: 2, Feb. 2017
[2] M. Alioto, "Ultra-low power VLSI circuit design demystified and explained: A tutorial," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 59, no. 1, pp. 3–29, Jan. 2012.
[3] V. Gupta, D. Mohapatra, A. Raghunathan, and K. Roy, "Low-power digital signal processing using approximate adders," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 32, no. 1, pp. 124–137, Jan. 2013.
[4] S. Hashemi, R. I. Bahar, and S. Reda, "DRUM: A dynamic range unbiased multiplier for approximate applications," in Proc. IEEE/ACM Int. Conf. Comput.-Aided Design (ICCAD), Austin, TX, USA, 2015, pp. 418–425.
[5] K. Y. Kyaw, W. L. Goh, and K. S. Yeo, "Low-power high-speed multiplier for error-tolerant application," in Proc. IEEE Int. Conf. Electron Devices Solid-State Circuits (EDSSC), Dec. 2010, pp. 1–4.
[6] A. Momeni, J. Han, P. Montuschi, and F. Lombardi, "Design and analysis of approximate compressors for multiplication," IEEE Trans. Comput., vol. 64, no. 4, pp. 984–994, Apr. 2015.
[7] K. Bhardwaj, P. S. Mane, and J. Henkel, "Power- and area-efficient approximate wallace tree multiplier for error-resilient systems," in Proc.15th Int. Symp. Quality Electron. Design (ISQED), 2014, pp. 263–269.
[8] https://en.wikipedia.org/wiki/Kogge–Stone_adder
[9] https://en.wikichip.org/wiki/Brent-Kung_adder

©IJRASET: All Rights are Reserved



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)