iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 2 **Issue:** VIII **Month of publication:** August 2014

DOI:

www.ijraset.com

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

ISSN: 2321-9653

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

Novel Approach to Crosstalk Avoidance in RLC Modelled Interconnects Using Bus Encoder

Sheetal Meena[1], Ghanshyam Jangid[2]

1(Department of Electronics and communication , Gyan Vihar University, Jaipur, Raj)
2(Assistant Professor, Department of Electronics and Communication ,Gyan Vihar University , Jaipur, Raj)

Abstract-In this paper, we have proposed efficient bus encoder for crosstalk avoidance in RLC modelled interconnects. In past years many encoding methods have been proposed which dealt with only RC modelled VLSI inter-connects.Due to fast clock speeds, longer length of on-chip interconnects and smaller signal rise times, the on-chip inductive effects have increased for the deep submicron technologies (DSM). All these factors raises the overall concern for crosstalk, propagation delay and power dissipation. Therefore, this research paper introduces an efficient Bus Encoder by the help of Bus Inverting (BI) technique. Proposed design reduces both crosstalk delay and area in the RLC modelled interconnects which makes the circuit design suitable for current high-speed low power VLSI interconnects. The proposed work shows an overall reduction of crosstalk delay by 17.54% and area efficiency by 28.57%.

Keywords—Inductance effects, Bus –invert , Crosstalk delay, Area efficiency .

I. INTRODUCTION

In deep sub-micron technology, the performance of high speed chip depends on interconnects connecting different micro-cells within a VLSI/ULSI chip [1]. The effects of interconnects can't be restricted to RC models due to the increased length of interconnects and on-chip clock frequency. Because of faster rise time, wider wires and introduction of new materials for low resistance interconnects, the importance of on-chip inductance is increasing continuously. Inductance increases the interconnect delay per unit length and causes crosstalk in the signal waveforms, which adversely affects signal integrity.

System engineers present several kinds of option to decrease area and propagation delay on buses. The first option is that a shielding line (VDD/Ground) can be inserted between two adjacent signal lines. And other options are repeater insertion, optimal spacing between signal line and the most effective method is Bus Encoding technique.

This research paper uses Bus- invert (BI) method for the avoidance of crosstalk delay and overall reduction in the chip-size of RLC modelled interconnects. Thus, the proposed technique transforms the bus signal to reduce and eliminates five undesirable types of crosstalk i.e. Type-0,Type-1,Type-2,Type-3 and Type-4. They reduce power and wire propagation delay by decreasing the switching and coupling activities.

The paper is organized in five sections. The present section I introduces to the critical research scenario in VLSI interconnects. Section II deals with the crosstalk and power dissipation expression , section III give brief idea about the previous work , section IV explain our proposed model and last section V draws conclusion and its future scope.

II.CROSSTALK AND POWER IN RLC MODELED INTERCONNECTS.

The parasitic capacitance model of an interconnect consists of three parts : Ground Capacitance (C_G), the fringe or side-wall capacitance to substrate (C_F), and coupling capacitance (C_C).When adjacent wires tends to switch from 0 to 1 or 1 to 0 , the coupling capacitance becomes dominant and results in crosstalk delay. This crosstalk results in noise on non-switching wires and increased delay on switching wires.

Suppose two lines A and B and their respective capacitance. The effective coupling capacitance (C_{eff}) can be determined with respect to the behaviour of neighbouring wires. Table 1 shows the dependency of effective capacitance of line A on line B, where line A is switching.

ISSN: 2321-9653

Line 'B'	ΔV	$C_{eff(A)}$	MCF
Switching with 'A'	0	C_G	0
Constant	V_{dd}	$C_c + C_G$	1
Switching oppositely with 'A'	$2V_{dd}$	$2C_c + C_G$	2

Table 1: Dependency of effective capacitance of line A on line B

The Miller Coupling Factor (MCF) is '0' which indicates that there is no coupling capacitance when both adjacent lines are switching in the same direction. If one line is switching and other is constant then the MCF is '1' which indicates that coupling capacitance is greater as compared to that of MCF='0'. Finally when two adjacent lines are switching in opposite direction then the coupling capacitance is highest i.e. MCF='2', due to this crosstalk effect becomes dominant.

All possible switching configurations can be classified to Type-0,Type-1,Type-2,Type-3 and Type-4 depending on the value of MCF as shown in table 2.(↑: switching from 0 to 1, ↓: switching from 1 to 0, - : no transition)

Type-0	Type-1	Type-2	Type-3	Type-4
- - -	- - ↑	- ↑ -	- ↑ ↓	↑ ↓ ↑
↑ ↑ ↑	- ↑ ↑	↑ - ↑	- ↓ ↑	↓ ↑ ↓
↓ ↓ ↓	↑ - -	↑ - ↓	↑ ↓ -	
	↑ ↑ -	↑ ↑ ↓	↓ ↑ -	
	- - ↓	↑ ↓ ↓		
	- ↓ ↓	- ↓ -		
	↓ - -	↓ - ↓		
	↓ ↓ -	↓ - ↑		
		↓ ↓ ↑		
		↓ ↑ ↑		

Table 2: Classification of crosstalk

Now for reducing power dissipation in VLSI circuits one or more factors such as V_{dd}, f, C_L and α must be minimized.

Power dissipation is expressed as:

$$P=\alpha * V_{dd}^2 * f * C_L$$

Here, V_{dd} and f are supposed to be already optimized for low power. Thus dynamic power dissipation is proportional to the number of signal transition.

III. PREVIOUS WORK

The previous work deals only with the two types of crosstalk i.e. Type-0 and Type-1 for the 4 data bits.

In this the data bus is divided into different clusters where each cluster contains 4 data bits and one extra control bit. In bus-invert method, if the number of transmitting transitions are more than half of the bus width then the original data are inverted and the control line is set to 'High' otherwise, the original data are transmitted and the control line is set to 'low'. The control bit in bus invert method differentiates the transmission of original data and inverted data.

The block diagram of this proposed encoder is shown in the figure 1. the 5 bit bus encoder architecture consists of inverter, CNT0, CNT1_1, CNT1_2, 2 bit comparator, XOR stack and latch. CNT0 and CNT1 (CNT1_1 and CNT1_2) are crosstalk modules used to count Type0 and Type1 couplings respectively. CNT1_1 and CNT1_2 are used to counts the number of Type1 couplings with original data and inverted data respectively.

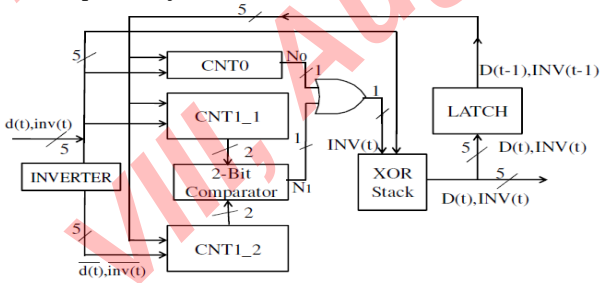


Figure 1: Block Diagram of 5-bit bus encoder

In this, the data to be transmitted i.e. d(t) and inv(t) through the bus encoder is inverted simultaneously into $\overline{d(t)}$ and $\overline{inv(t)}$. Initially it is assumed that the value of inv(t) in the data to be transmitted is at logic 'low'. Next, in CNT0 and CNT1_1 counter, the original data and the previous output i.e. D(t-1) and INV(t-1) are given as the inputs.The output of CNT0 is N_0 and that of CNT1_1 is K_1K_0. The data stored i.e. D(t-1)and INV(t-1)and the inverted input data $\overline{d(t)}$ and $\overline{inv(t)}$ are given as the input to counter CNT1_2 whose output is L_1L_0 . The output of two Type1 counters is compared in a 2-bit comparator. K_1K_0 and L_1L_0 are the inputs to the comparator and output from comparator is N_1which is of 1-bit.After the comparison of K_1K_0 and L_1L_0, the 2-bit comparator generates the output 'N_1=1' if the output of CNT1_1 is larger as compared to that of CNT1_2 otherwise 'N_1=0' in the converse condition. Then, N_0 and N_1 are given as the inputs to an OR gate and the output thus obtained from that OR gate is INV(t). This output of OR gate i.e. INV(t) along with the original

ISSN: 2321-9653

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

input data i.e. d(t) and inv(t) are given as the inputs to the XOR stack. The output of XOR stack depends on the value of INV(t). If the value of INV(t)=1 then the output of XOR stack is the inverted data otherwise the output is the original data given as the input.Finally the output obtained from XOR stack is in the encoded form (D(t) and INV(t)) which is then fed to the interconnects. For one clock cycle, the encoded output is stored in the latch (D(t-1), INV(t-1)) and after one clock cycle it is feedback for comparison with d(t) and inv(t). Finally, decoder extracts the original input data with the help of INV(t) line at the receiver end.

The limitation of this design is that it only works on the Type-0 and Type-1 not on the other types of crosstalk.

IV. PROPOSED WORK

Our proposed design is the modification of the previous work of "Bus encoder for crosstalk avoidance in RLC modelled interconnects"[2]. Now in this paper we deal with the all five types of crosstalk i.e. Type-0 , Type-1 , Type -2, Type-3 and Type-4.

Type-0	Type-1	Type-2	Type-3	Type-4
↑↑↑↑	-↑↑↑	↑↑-↑	-↑↓↑	↑↓↑↓
↓↓↓↓	↑↑↑-	↑↑-↓	↑↓↑-	↓↑↓↑
	-↓↓↓	↑-↑↑	↑↓-↑	
	↓↓↓-	↑-↓↓	↑-↓↑	
		↑↓↓↓	↑↓-↓	
		↑↑↑↓	↑-↑↓	
		↑↑↓↓	-↓↑↓	
		-↑↓↓	↓↑↓-	
		-↑↑↓	↓↑-↓	
		↑↑↓-	↓-↑↓	
		↑↓↓-	↓↑-↑	
		↓↓-↓	↓-↓↑	
		↓↓-↑		
		↓-↓↓		
		↓-↑↑		
		↓↑↑↑		
		↓↓↓↑		
		↓↓↑↑		
		-↓↑↑		
		-↓↓↑		
		↓↓↑-		
		↓↑↑-		

Table 3: Cases of crosstalk which requires inversion of the original data

In this proposed technique , data bus is divided into different clusters and each cluster contains 4-data bits.This design reduces the overall crosstalk delay and area more effectively in comparison to the previous bus encoder.

The proposed design is consists of following blocks:- five types of crosstalk counters i.e. CNT0, CNT1, CNT2, CNT3, CNT4 , an OR gate, XOR stack and a latch.

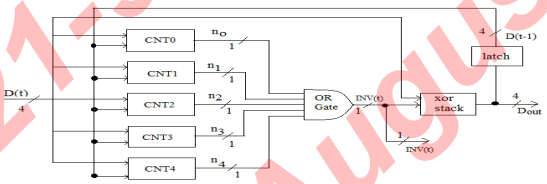


Figure 2: Block Diagram of 4-bit Encoder

The 4-bit data D(t) is inserted as the input to the different five types of crosstalk counters i.e. CNT0, CNT1, CNT2, CNT3 and CNT4 where these data is compared with the previously stored data in the latch i.e. D(t-1).

CNT0:- This block detects the Type0 coupling which occurs if all the four adjacent lines are having transitions in same direction (i.e. ↑↑↑↑ or ↓ ↓ ↓ ↓). The output 'n_0' of this block is 'high' when this coupling is detected.

CNT1:- It detects the Type1 coupling which occurs in the case when three adjacent lines have transition in the similar direction and the fourth line is kept unchanged. The respective output 'n1' of this block is at logic '1' when Type1 coupling is detected by this block.

CNT2:- This block is used to detect the Type2 coupling and the output of this block (i.e. n_2) is 'high' when three or four lines are changed and at least two adjacent lines of them are having transition in the similar direction.

CNT3:- It is used to detect the Type3 coupling and the output 'n_3' of this block is 'high' when it detects the case when three adjacent lines have transitions in the alternate direction and the fourth line is unchanged. Also, if two adjacent lines are

ISSN: 2321-9653

having transition then both the transitions should be in alternate direction, the third line is unchanged and the fourth line can have transition in either direction.

CNT4:- This block detects Type4 coupling and the output 'n_4' of this block is 'high' when all the four lines are changing and have transition in the same direction.

The original input data D(t) and the previously stored data in the latch i.e. D(t-1), is compared in the crosstalk counters. If coupling is detected by any one of the counter, then the output of that respective counter is 'high' and the output of all the other counters is 'low'. Next, n_0, n_1, n_2, n_3 and n_4 are fed as inputs to an OR gate whose output is INV(t) . This INV(t) and the original data D(t) are given as inputs to XOR stack. The output of XOR stack is inverted data, if INV(t) is '1' or the original data, if INV(t) is '0'. The output of the XOR stack is the encoded data $D_{out}(t)$ which is finally fed to interconnects. This encoded data is stored in latch for one clock cycle D(t-1), after which it is fed back for comparison with D(t). Finally, at the receiving side, decoder retrieves the original data with the help of INV(t) line.

This proposed work eliminates the limitation of the previously proposed design.

V. RESULT AND COMPARISON

In this paper, performance of base paper and proposed Bus Encoder is evaluated and implemented. We synthesis this Bus Encoder using Xilinx ISE 9.2 and use Verilog as hardware description language. From the table 4, comparison can be done between previous and proposed design.

	Previous Design	Proposed Design
No.of slices	7	5
Delay (ns)	4.775	3.937

Table 4:Comparison between previous and proposed design.

We can see that the speed of proposed work is improved by 17.54% than the previous design and area of proposed work is also reduced by 28.57%. Comparing the simulation results of our proposed design with those of previous design it is proved that the proposed bus encoder performed best in terms of speed and area. The schematic of proposed bus encoder is shown in fig.3 and the simulation waveform is also shown in fig.4.

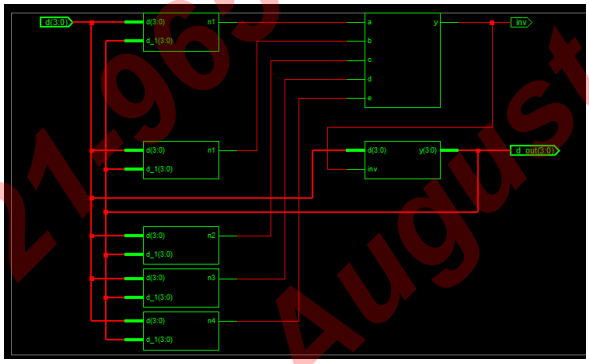


Fig.3: RTL Schematic of Proposed Bus Encoder

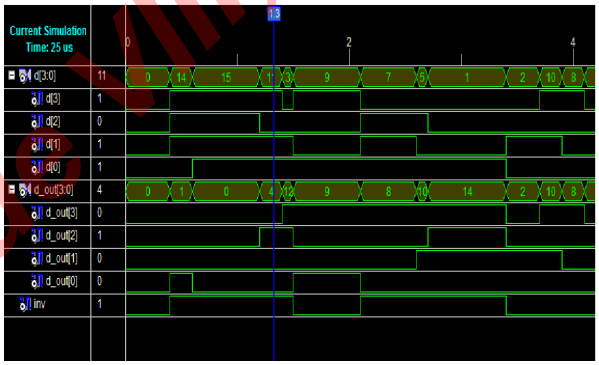


Fig.4: Simulation result

The schematic of proposed design is shown in fig.3and the simulation waveform is also shown in fig.4.

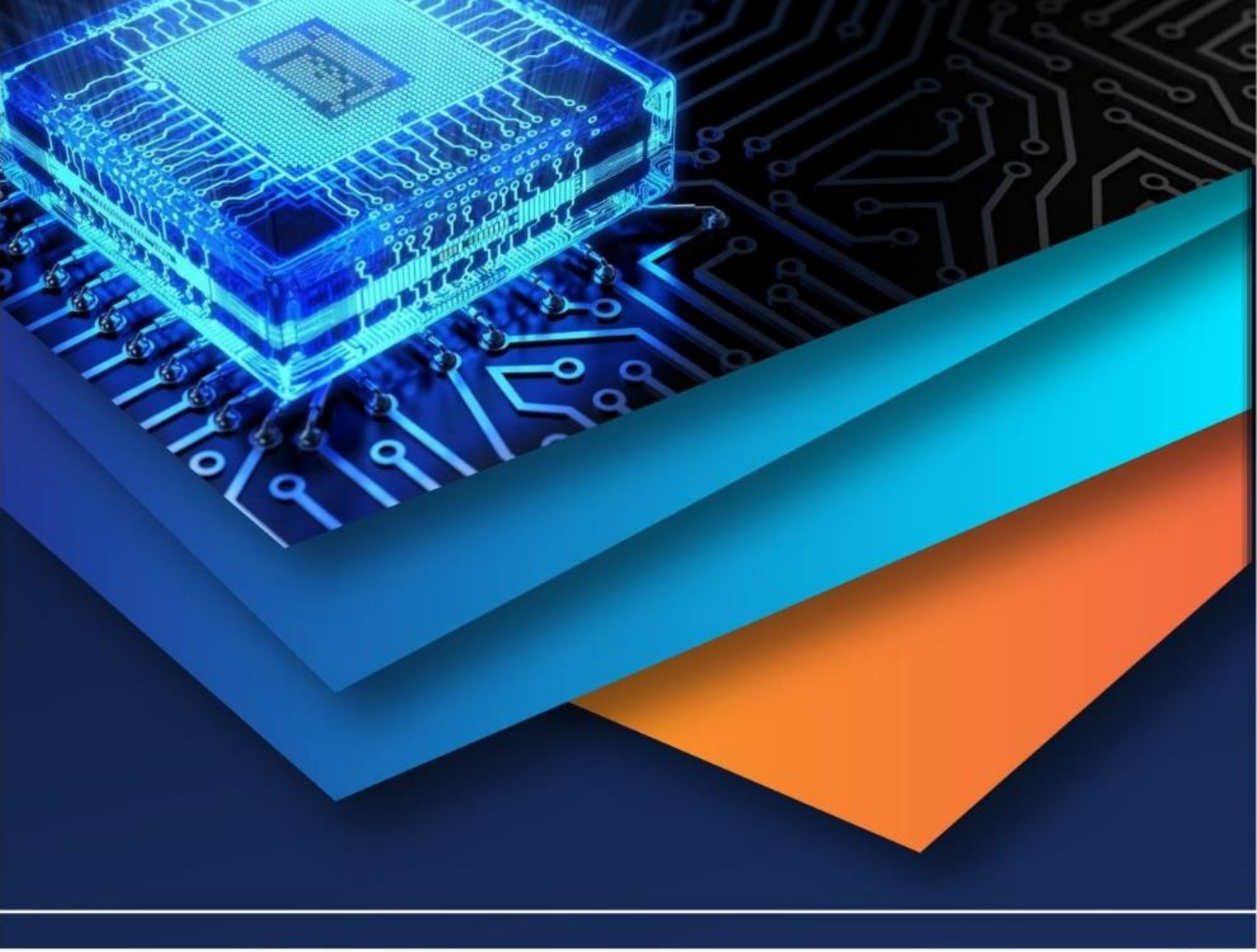
VI. CONCLUSION

In this paper, the previous and modified bus encoder are designed using Verilog. The speed of modified bus encoder is more as compared to previous work. We dealt with all five types of crosstalk in this paper. It increases the speed of signals by eliminating the crosstalk. The proposed bus encoder provides improved speed and area.

In future, this design can be used for higher no. of input bits such as 8,16,32 etc. Also, the delay and power can be improved using pipeline concept, which in turn enhances the speed and performance.

REFERENCES

[1] L.Trevillyan, D. Kung, R. Puri, L.N. Reddy, and M.A. Kazda, "An integration environment for technology closure of deep-submicron IC designs," IEEE Des. Test. Comput. , vol. 21, no. 1, pp. 14- 22, 2004.
[2] International Technology Roadmap for Semiconductors 2007.
[3] M.A.Elgamel, and M.A. Bayoumi, "Interconnect noise analysis and optimization in deep submicron technology," IEEE Circuits Syst. Mag., vol. 3, no. 4, pp. 6-17, 2003.
[4] L. He, and K.M. Lepak, "Simultaneous Shield Insertion and Net ordering for Capacitive and Inductive Coupling Minimization, " Int. Symp. Physical Design , pp. 55-60, 2000.
[5] T. Shang-Wie, C. Yao-Wen, and J. Jing-Yang, " RLC Coupling-Aware Simulation and On-Chip Bus International Journal of VLSI design & Communication Systems (VLSICS) Vol.3, No.1, February 2012 190 Encoding for Delay Reduction," IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems, vol. 25, no. 10, 2006.
[6] K. Hirose, and H. Yasuura, "A bus delay reduction technique considering crosstalk," In Proc. Design Automation and Test Eur. (DATE), Paris, France, pp. 441-445, 2000.
[7] C. Peng Fan, and C. Hao Fang, "Efficient RC low-power bus encoding methods for crosstalk reduction," Integration, the VLSI Journal, Elsevier, vol. 44, no. 1, pp. 75-86, 2011.
[8] B. Victor, and K. Keutzer, "Bus encoding to prevent crosstalk delay," in Proc. Int. Conf. on Computer-Aided Design, 2001, pp. 57-63.
[9] M. R. Stan, and W. P. Burleson, "Bus-Invert Coding for Low-Power I/O," IEEE Trans. VLSI Syst., vol. 3, pp. 49-58, 2005.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY