

DYNAMIC CIRCUIT TECHNIQUE FOR LOW- POWER MICROPROCESSORS

Kuruva Hanumantha Rao¹ (M.tech)
hanumanthurao19@gmail.com¹

K.Prasad Babu² M.tech (Ph.d)
kprasadbabuece433@gmail.com²

¹PG scholar, VLSI, St.JOHNS COLLEGE OF ENGINEERING and TECHNOLOGY, YERRAKOTA, KURNOOL, ANDHRA PRADESH

²Assistant Professor, Dept of ECE, St.JOHNS COLLEGE OF ENGINEERING and TECHNOLOGY, YERRAKOTA, KURNOOL, ANDHRA PRADESH.

Abstract: Dynamic circuits are widely used in order to solve the problems occurred in the data path and the critical components of the microprocessor. The power consumption is significantly in dynamic circuits due to their switching activity. In order to obtain high performance dynamic circuits are used in microprocessors because of their special features such as speed and area. In this paper, we proposed flexible charge recycling design methodology and dynamic circuit selection algorithm in order to achieve high efficiency in the data path. According to the proposed methodology the simulation result of the design shows the consumption of the ALU (Arithmetic and Logic Unit) with proposed technique is reduced up to 50% to 60% than the conventional ALU. The proposed technology is carried out under 22nm technology.

Key words: charge recycling, low power, n-type dynamic circuit, p-type dynamic circuit.

I.INTRODUCTION

The number of transistors in a chip has extremely grown continuously with advanced technology, over a past decade As there is an increase in the transistor count it demands for less power consumption and less area, these are the main issues occur during the design of the circuit, with an increase in transistor count it leads to increase the transistor density, power consumption of the microprocessor, as a result these are major issues as to be carried out to ensure the high performance of the proposed system. In a modern microprocessor the data paths perform computing operations through a critical path. The operating frequency of the microprocessor can be determined by the operating speed of the data path. The data path consumes a significant amount of total power consumption and also it is a most active component. This results in a problem in those applications which as intensive computation on it, such as multimedia processor with

multiple core and digital signal microprocessor. Hence, in a modern micro processor, the low power data path can be achieved.

The dynamic circuit is widely used as a solution in critical data path and critical components in microprocessors due to their features such as speed and area. The n-type dynamic circuit is applied to on-chip memory and arithmetic and logic unit (ALU) of some microprocessors to minimize the latency. A high performance microprocessors use dynamic circuits, which can achieve a timing constraint than a static CMOS. The performance of dynamic circuits is increased due to low switching thresholds, reduced input capacitance and circuit implementation is achieved using high efficient and complex gates. The dynamic circuits are used in the implementation of a high frequency internal processor by using a logic blocks which have a timing requirement.

In the proposed paper, a novel p-type/n-type dynamic circuit selection (PNS) algorithm and flexible charge recycling (FCR) methodology is designed in modern microprocessors in order to obtain a low power data path.

This property makes synthesizing dynamic circuits with Computer Aided Design (CAD) tools more difficult than synthesizing static CMOS circuits. In addition, the varying characteristics of different types of dynamic circuits (n-type and p-type) increase the design complexity of a data path. Unfortunately, the existing solutions are not sufficient to solve these issues. In this paper, a novel p-type/n-type dynamic circuit selection (PNS) algorithm and a flexible charge recycling (FCR) design methodology are proposed, referred to here as PNS-FCR, which targets low power data paths in modern microprocessors.

II. RELATED WORK

In earlier research the techniques used were n-p dynamic circuits, n-p dynamic circuits are categorized as n-dynamic and p-dynamic. In the n-dynamic circuits it adopts high-speed NMOS transistor to achieve high performance and in p-dynamic circuits it adopts slower PMOS transistor hence the speed is slower, but the power efficiency is enhanced due to suppressed gate and sub threshold leakage current of PMOS transistor. The n-p dynamic circuit has been proposed as a race-free dynamic CMOS technique for pipelined circuits. The n-p dynamic circuit has lower intrinsic delay and requires less silicon area due to the more compact logic than with the static CMOS logic. Binding algorithm is mainly based on framework for low leakage data path. This technique is effective only for some applications and there is a considerable speed loss and not suitable for any high performance application

A macro driven data path design methodology has developed and it generates different topologies for different macros [6]. Three more methods have developed for synthesizing dynamic circuits, but these only contain n-type dynamic circuits and failed to include p-type dynamic circuits. crosstalk-aware and speed-aware synthesis methodologies are presented, but neither of them considers power efficiency [7]-[8]. Finally, a dynamic data path is synthesized automatically, but requires a significant silicon area. The common feature of these techniques is that the potential for low power by combining different types of dynamic circuits is not effectively explored.

III. PROPOSED METHODOLOGY

The proposed PNS-FCR, exploring power saving opportunity for data path circuits, is presented in this section. The three-step PNS-FCR design methodology is depicted in fig1.

Design Flow for a Data Path Based on PNS-FCR: As shown in Fig. 2, the design flow for a data path based on PNS-FCR is as follows: 1) First, the gate library based on a p-type/n-type dynamic circuit is built. Two types of each gate occupy a similar layout area to avoid area penalty. 2) Based on the gate library, the appropriate type of gates is selected using PNS to

implement the data path or critical path, satisfying the performance requirements of different applications. 3) Next, the FCR is utilized to achieve high power efficiency in critical path by inserting the charge recycling paths between two independent gates or two neighboring gates. Note that the FCR is a tradeoff between power, performance, and silicon area. 4) Then, apply the proposed PNS-FCR to noncritical paths. The critical path is typically much longer than the uncritical path in the data path, and therefore, the gates in the uncritical path employ p-type for power efficiency. However, if an uncritical path formed by all p-type gates is even slower than the critical path, n-type gates would be inserted to meet the delay constraint based on PNS, and then the FCR is used to enhance the power efficiency. 5) Finally, the routing is completed manually or by CAD tools.

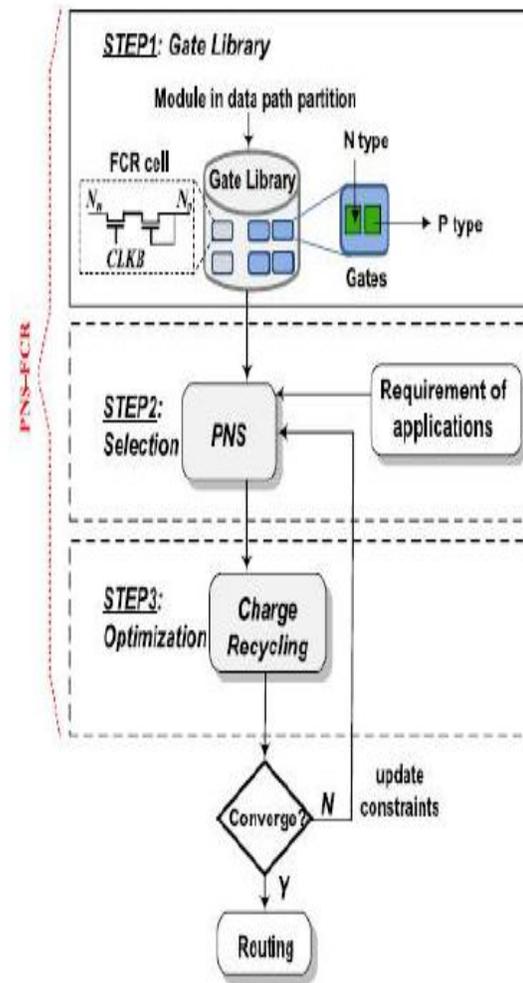


Fig.1. Proposed PNS-FCR methodology

PN Selection Algorithm Based on the gate library, the appropriate type of gate is selected to implement a data path, as shown in Fig. 1. To satisfy the performance requirements of different applications, a PNS algorithm is introduced based on the multidimensional multiple-choice 0-1 Knapsack problem (MMKP). The delay of the critical path determines the performance of the data path. The gates in the critical path are required to meet the performance constraint, while the gates in the noncritical paths, use a low power version to enhance power efficiency.

Flexible Charge Recycling technique: A key design issue in low power data paths is exploring the choice of different power efficient n-type and p-type gates. Accordingly, the FCR is proposed to achieve high power efficiency, as shown in Fig. 2. Consider a critical data path exist between two cascaded gates. The initial gate is an n-type dynamic gate, while the latter gate is a p-type dynamic gate. During the pre charge stage (CLK = 0), the dynamic node of the n-type gate N_n is pre charged to V_{dd} through transistor P_{cn} , while the dynamic node of the p-type gate N_p is discharged to ground through the transistor N_{cp} .

During the evaluation stage, provided that the necessary input combination is applied, N_n is discharged to the ground and N_p is charged to V_{dd} . Otherwise, the high state of N_n and low state of N_p are maintained until the next pre charge stage. As the evaluation process completes, N_n discharges from high to low and N_p charges from low to high. In the following pre charge stage, N_n and N_p both consume dynamic power by charging N_n from V_{dd} and discharging N_p to ground. If a switch is inserted between the two dynamic gates, N_n is charged by N_p through a charge through a charge recycling path, thereby reducing the dynamic power.

The charge recycling path exists between two independent gates as well as two neighboring gates. If r p-type gates and q n-type gates are selected for a critical path, $\min(r, q)$ charge recycling paths can be inserted to reduce power. Accordingly, in the evaluation stage, the voltage waveforms of N_n and N_p without and with the FCR cell almost overlap.

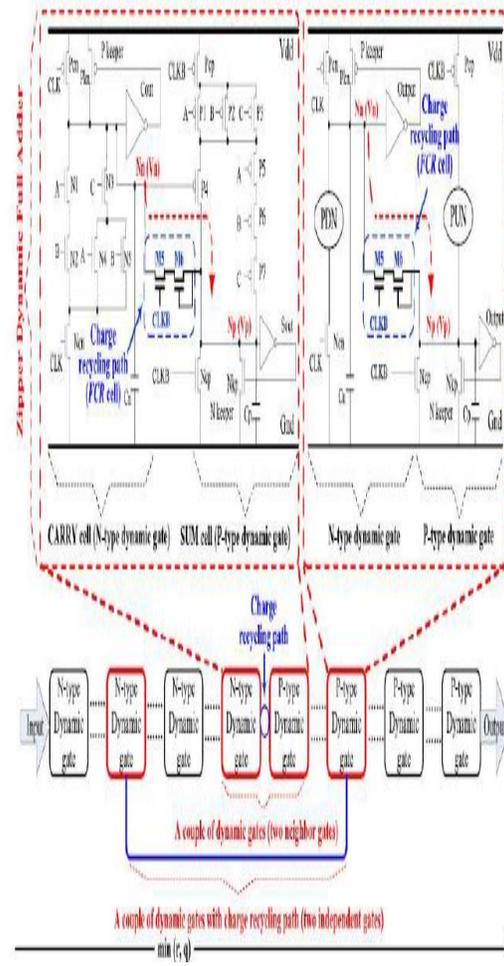


Fig.2. FCR technique

A dynamic full adder is taken as an example, as shown in Fig.2. When the input vectors of full adder are respectively (1, 1, 0), (1, 0, 1), and (0, 1, 1), at the end of an evaluation stage, N_n has been discharged to Gnd while N_p has been charged to V_{dd} , and the switch is turned on. And then, a desirable charge recycling path between N_n and N_p is built. With the FCR cell, in the precharge stage, the $CLKB$ makes the recycle path available. Consequently, two supplies V_{dd} and N_p charge N_n simultaneously, which makes the precharge speed much higher as compared with the conventional circuit with only single supply V_{dd} charging N_n . The additional capacitance C_r between dynamic nodes N_n and N_p due to an adding charge recycling path has a negligible effect on evaluation speed.

EXTENSION METHOD.

Clock gating technology can reduce the consumption of clock signals' switching power of flip-flops. The clock gate enable functions can be identified by Boolean analysis of the logic inputs for all adders. However, the enable functions of clock gate can be further simplified, and the average number of adders driven by enable functions can be improved. In this way, the circuit area can be reduced; therefore, the clock gating can be improved and power saving can be achieved.

IV. EXPERIMENTAL RESULTS

Results of proposed method

Verification of FCR: To verify the effectiveness of FCR, a full adder with clock-delay as designed, which is usually employed along the critical path in data path. A full adder with FCR includes one n-type Sum cell, one p-type Carry cell (dynamic full adder), and one FCR cell to enhance power efficiency (Fig. 2). The fig.3 and fig.4 shows the schematic design of the adder circuit without and with FCR respectively. The output waveforms shows the voltage and dynamic node variation with respect to clk signal.

SCHEMATIC:

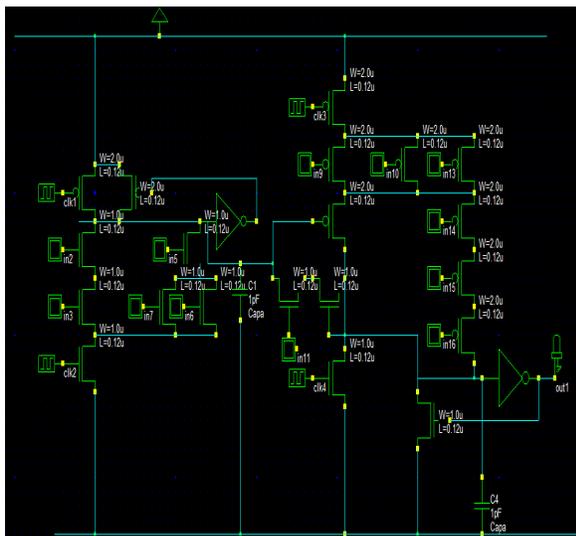


Fig.3. Schematic design of full adder circuit

In the above figure the adder circuit is designed and it is a dynamic full adder which consumes less power than the conventional adder. This adder circuit has 3 inputs and 2 outputs and the adder is called as ripple carry order where the carry

out of one adder is given to the next adder and the carry out is obtained at the last adder circuit. This adder is implemented in ALU in order to achieve high efficiency.

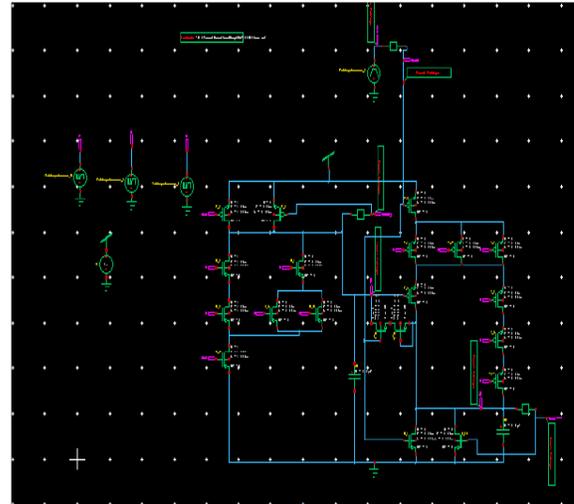


Fig.4. Schematic design of full adder with FCR circuit

Fig.4 shows the schematic view of the adder circuit with proposed technology. In this circuit the FCR circuit is implemented in order to have a charge recycling path between two dynamic circuits. This FCR circuits acts as a switch between dynamic circuits, this leads to low switching activity and high efficiency of the circuit.

SIMULATION RESULT:

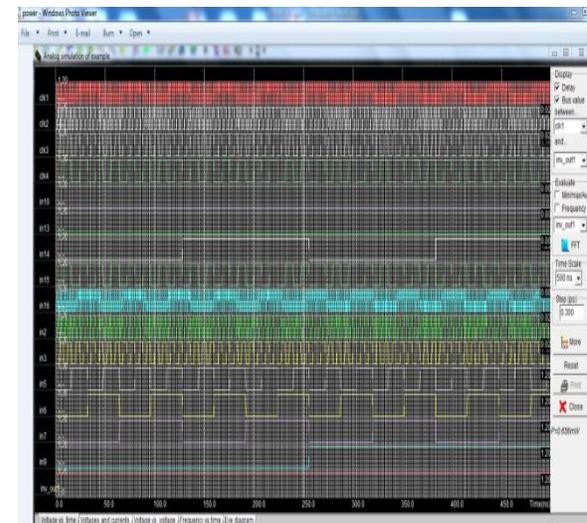


Fig.5. Waveforms of CLK and dynamic nodes of FCR cell

Fig 5 shows the simulation results of FCR cell. As there is a variation in the clock signal the nodes of the dynamic circuit also varies according to the inputs of the adder.

Layout Design:

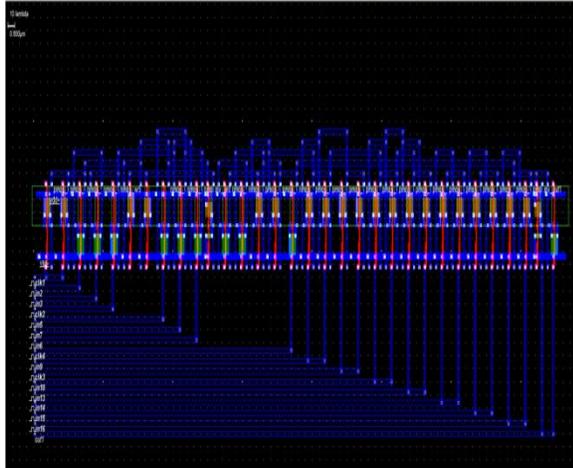
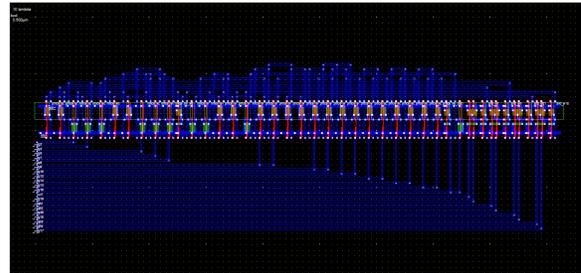
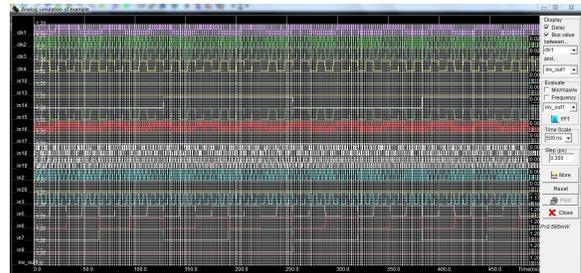


Fig.5. Layout of full adder with FCR circuit

Layout Design:



SIMULATION RESULT



```

Verilog, Hierarchy and Netlist
Verilog | Hierarchy | Netlist | Critical path
Information
Module name (8 char. max)
fig3_1
[ ] Add gate delay info
[ ] Append simulation informations
[ ] Add labels as comments
The Verilog file has 70 lines
The design includes 43 symbols
The circuit has 31 nodes
OK
// DSC# 2.7E
// 14-Oct-17 3:46:27 PM
// C:\Users\saandee\Desktop\PNS\fig3_1.sch
module fig3_1( in16,in2,in3,in15,in5,in6,in7,in14,
in9,in10,in11,clk4,in13,clk1,clk2,clk3,
out1);
input in16,in2,in3,in15,in5,in6,in7,in14;
input in9,in10,in11,clk4,in13,clk1,clk2,clk3;
output out1;
pmos # (31) pmos(w2,vdd,clk1); // 2.0u 0.12u
nmos # (31) nmos(w2,w3,in2); // 1.0u 0.12u
nmos # (10) nmos(w3,w5,in3); // 1.0u 0.12u
nmos # (24) nmos(w5,vss,clk2); // 1.0u 0.12u
pmos # (24) pmos(vdd,w2,w8); // 2.0u 0.12u
nmos # (31) nmos(w2,w9,in5); // 1.0u 0.12u
nmos # (17) nmos(w9,w5,in7); // 1.0u 0.12u
nmos # (17) nmos(w9,w5,in6); // 1.0u 0.12u
not # (10) inv(w8,w2);
nmos # (17) nmos(w13,w14,w15); // 1.0u 0.12u
nmos # (10) nmos(w14,w16,w16); // 1.0u 0.12u
pmos # (52) pmos(w16,w17,w13); // 2.0u 0.12u
nmos # (52) nmos(w16,vss,clk4); // 1.0u 0.12u
pmos # (31) pmos(w17,w19,in9); // 2.0u 0.12u
pmos # (24) pmos(w19,vdd,clk3); // 2.0u 0.12u
pmos # (31) pmos(w17,w19,in10); // 2.0u 0.12u
pmos # (31) pmos(w17,w19,in13); // 2.0u 0.12u
nmos # (10) nmos(w25,w17,in14); // 2.0u 0.12u

```

Fig.6. Verilog, Hierarchy and Netlist of full adder with FCR circuit

Verilog, Netlist of full adder with FCR circuit

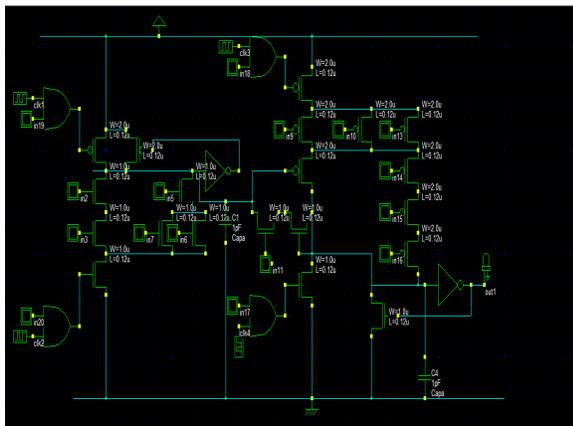
```

Verilog, Hierarchy and Netlist
Verilog | Hierarchy | Netlist | Critical path
Information
Module name (8 char. max)
ext_fig3_1
[ ] Add gate delay info
[ ] Append simulation informations
[ ] Add labels as comments
The Verilog file has 83 lines
The design includes 51 symbols
The circuit has 40 nodes
OK
// DSC# 2.7E
// 14-Oct-17 3:51:24 PM
// C:\Users\saandee\Desktop\PNS\ext_fig3_1.sch
module ext_fig3_1( in16,in2,in3,in15,in5,in6,in7,in14,
in9,in10,in11,clk4,in13,clk1,clk2,clk3,
in20,in17,in18,in19,out1);
input in16,in2,in3,in15,in5,in6,in7,in14;
input in9,in10,in11,clk4,in13,clk1,clk2,clk3;
input in20,in17,in18,in19;
output out1;
pmos # (31) pmos(w2,vdd,w1); // 2.0u 0.12u
nmos # (31) nmos(w2,w3,in2); // 1.0u 0.12u
nmos # (10) nmos(w3,w5,in3); // 1.0u 0.12u
pmos # (24) nmos(w5,vss,w7); // 1.0u 0.12u
pmos # (24) pmos(vdd,w2,w8); // 2.0u 0.12u
nmos # (31) nmos(w2,w9,in5); // 1.0u 0.12u
nmos # (17) nmos(w9,w5,in7); // 1.0u 0.12u
nmos # (17) nmos(w9,w5,in6); // 1.0u 0.12u
not # (10) inv(w8,w2);
nmos # (17) nmos(w13,w14,w15); // 1.0u 0.12u
nmos # (10) nmos(w14,w16,w16); // 1.0u 0.12u
pmos # (31) pmos(w16,w17,w13); // 2.0u 0.12u
nmos # (31) nmos(w16,vss,w18); // 1.0u 0.12u
pmos # (31) pmos(w17,w19,in9); // 2.0u 0.12u
pmos # (24) pmos(w19,vdd,w21); // 2.0u 0.12u
pmos # (31) pmos(w17,w19,in10); // 2.0u 0.12u
pmos # (31) pmos(w17,w19,in13); // 2.0u 0.12u

```

Results of Extension method

SCHEMATIC:



V. CONCLUSION

A novel methodology is presented in this paper for designing dynamic circuits in the functional units of modern processors. The proposed PNS-FCR methodology achieves high power efficiency, while satisfying specific timing constraints. Simulation results shows that the power consumption of adder in different submicron technology. The proposed adder utilizes low power than the conventional adder and the high performance is achieved. This proposed technique is implemented in ALU which consumes low power and achieves a higher efficiency than the conventional ALU. This methodology can be extended to static CMOS, pass gate, transmission gate, tri state gate, and other logic families.

REFERENCES

- [1] H. McIntyre et al., "Design of the two-core x86-64 AMD 'Bulldozer' module in 32 nm SOI CMOS," IEEE J. Solid-State Circuits, vol. 47, no. 1, pp. 164-176, Jan. 2012.
- [2] M. Golden, S. Arekapudi, and J. Vinh, "40-entry unified out-of-order scheduler and integer execution unit for the AMD Bulldozer x86-64 core," in IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers, Feb. 2011, pp. 80-82.
- [3] R. Riedlinger et al., "A 32 nm, 3.1 billion transistor, 12 wide issue Itanium processor for mission-critical servers," IEEE J. Solid-State Circuits, vol. 47, no. 1, pp. 177-193, Jan. 2012.
- [4] K. Limnitis, Y. Tsiatouhas, T. Haniotakis, and A. Arapoyanni, "A design technique for energy reduction in NORA CMOS logic," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 53, no. 12, pp. 2647-2655, Dec. 2006.
- [5] C. Gopalakrishnan and S. Katkoori, "KnapBind: An area-efficient binding algorithm for low-leakage datapaths," in Proc. 21st Int. Conf. Comput. Design, Oct. 2003, pp. 430-435.
- [6] M. Nemani and V. Tiwari, "Macro-driven circuit design methodology for high-performance datapaths," in Proc. ACM/IEEE Design Autom. Conf., Jun. 2003, pp. 661-666.
- [7] Y.-Y. Liu and T. Hwang, "Crosstalk-aware domino-logic synthesis," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 26, no. 6, pp. 1115-1161, Jun. 2007.
- [8] T. J. Thorp, G. S. Yee, and C. M. Sechen, "Design and synthesis of dynamic circuits," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 11, no. 1, pp. 141-149, Feb. 2003.
- [9] H. Kellerer, U. Pferschy, and D. Pisinger, Knapsack Problems. Berlin, Germany: Springer-Verlag, 2004.
- [10] J. Wang, N. Gong, S. Geng, L. Hou, W. Wu, and L. Dong, "Low power and high performance Zipper domino circuits with charge recyclepath," in Proc. IEEE 9th Int. Conf. Solid-State Integr.-Circuit Technol., Oct. 2008, pp. 2172-2175.



Kuruva Hanumantha Rao is a PG Scholar, in VLSI, from St. Johns college of Engineering and Technology, Yerrakota, Kurnool, Andhra Pradesh.

BIOGRAPHIES:



K. Prasad Babu is an Assistant Professor in Department of ECE, in St. Johns college of Engineering and Technology, Yerrakota, Kurnool, Andhra Pradesh.