DESIGN AND ANALYSIS OF ENERGY EFFICIENT HIGH-SPEED ADDERS AND MULTIPLIERS FOR DIGITAL SIGNAL PROCESSORS

THESIS

Submitted by

KRISHNA NAIK DUNGAVATH

in partial fulfillment for the award of the degree

of

DOCTOR OF PHILOSOPHY

in

ELECTRONICS AND COMMUNICATION ENGINEERING

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

PONDICHERRY ENGINEERING COLLEGE

PONDICHERRY UNIVERSITY

PONDICHERRY – 605 014

INDIA

JANUARY 2018

Dr. V.VIJAYALAKSHMI
Associate Professor
Department of Electronics and Communication Engineering
Pondicherry Engineering College
Pondicherry – 605 014.

CERTIFICATE

Certified that this thesis entitled **"DESIGN AND ANALYSIS OF ENERGY EFFICIENT HIGH-SPEED ADDERS AND MULTIPLIERS FOR DIGITAL SIGNAL PROCESSORS"** submitted for the award of the degree of **DOCTOR OF PHILOSOPHY** in **ELECTRONICS AND COMMUNICATION ENGINEERING** of the Pondicherry University, Puducherry, is a record of original research work done by **Mr. KRISHNA NAIK DUNGAVATH** during the period of study under my supervision and that the thesis has not previously formed the basis for the award to the candidate of any Degree, Diploma, Fellowship or other similar titles. This thesis represents independent work on the part of the candidate.

Supervisor

(Dr. V.VIJAYALAKSHMI)

Date :
Place : Puducherry

DECLARATION

I hereby declare that the thesis titled **"DESIGN AND ANALYSIS OF ENERGY EFFICIENT HIGH-SPEED ADDERS AND MULTIPLIERS FOR DIGITAL SIGNAL PROCESSORS"** submitted to the Pondicherry University in fulfillment of the requirements for the award of the degree of **DOCTOR OF PHILOSOPHY** in **ELECTRONICS AND COMMUNICATION ENGINEERING,** is a record of original research work done by me under the supervision of **Dr.V.VIJAYALAKSHMI,** Associate Professor, Department of Electronics and Communication Engineering, Pondicherry Engineering College, Puducherry, and that the work has not been submitted either in whole or in part for any other degree or at any other university.

Signature

(KRISHNA NAIK. D)

ACKNOWLEDGEMENT

I am duty bound to express my sincere indebtedness to my supervisor and mentor, **Dr. V. Vijayalakshmi,** Associate Professor, Department of Electronics and Communication Engineering, for her unflinching support, selfless motivation and contagious enthusiasm from the inception of the research to the culmination stage. It would have been insuperable to present this thesis without her unstinted support. I revere her for her prolific edification in the upbringing of this work. It has been a great zest and honor to work with her.

I would also like to express my profound gratitude and allegiance to my Doctoral Committee Members, **Dr. V. Saminadan**, Professor, Department of Electronics and Communication Engineering, Pondicherry Engineering College and **Dr. S. Kanmani**, Professor, Department of Information Technology, Pondicherry Engineering College Affiliated to Pondicherry University, for their magnanimous benignity and benevolence, which has encouraged me to work harder and achieve the goal. I express my ingenious, sincere requital for their immense help, unswerving motivation and lively deliberation in spite of their busy schedule.

I express my gratitude to **Dr. P. Dananjayan,** Principal, Pondicherry Engineering College and **Dr. S. Himavathi**, Dean (Research) for their wholehearted support and for permitting me to make use of the facilities in the college for this research work.

I convey my gratefulness to **Dr. M. Tamilarasi,** Professor and Head, Department of Electronics and Communication Engineering, Pondicherry Engineering College for her wholehearted support and encouragement during the course of the research work.

I am grateful to **Dr. R. Srinivas,** Scientist/Engineer-SF, (ISRO), and **Prof. J. L Narasimhan**, for their wholehearted support and valuable guidance throughout my research work.

I wish to give the deepest gratitude to my parents **Mr. D. ThirupalNaik** and **Mrs. D. Lakshmi Bai** for their permanent encouragement in this long-suffering journey.

My profound gratitude to my wife, **Mrs. D. Hemavathi,** my son **Master Abhinav Singh Rathod** and my daughter **Chandana Priya** for their patience, kindheartedness and moral support offered during the crucial periods of my research.

I wish to sincerely thank my brothers, Mr. **Dr. Ram Naik**and **Mr. D. Ramesh Naik** and other family members for their sustained motivation. I am grateful to my friends, **Mr. V. Kamalakannan**, **Mr. K. Ashok Kumar,** **Mr. K. Sreekath Reddy**, **Mr. G.V.S Nandan,Mr. R. Heera Singh,** **Mr. A. JanardhanRao** and **Mr. R. L. Krupakaran** and my brother-in-laws **Mr. V. Sudhakar** and **Mr. V. Nagaraj** for their support and encouragement throughout my research work.

Finally, I immensely thank the Faculty, Staff and students of the Department of Electronics and Communication Engineering Pondicherry Engineering College and P.V.K.K. Institute of Technology, Anantapur, Andhra Pradesh, for the moral support rendered during the period of my research work.

I bow my head and thank THE ALMIGHTY for His blessings throughout my research period.

(KRISHNA NAIK .D)

ABSTRACT

The main objective of this research is to investigate the implementation and performance analysis of efficient high-speed Adders and low-power Multipliers with compact design and minimal power requirements for Digital Signal Processors (DSP). A Carry Select Adder (CSLA) has less delay compared to other Adders and is one of the fastest Adder used in many digital data processors to perform fast arithmetic functions. Phase I contribution is to develop an efficient Adder architecture. The proposed work uses a simple and an efficient gate-level modification to reduce the transistor count. It has been implemented by replacing dual Ripple Carry Adders (RCA's) with the proposed 8-bit Binary to Excess Converter (BEC) in the regular 512-bit Square-Root Carry Select Adder (SQRT CSLA) structure. The proposed Modified 512-bit SQRT CSLA has drastically reduced the area and delay of the conventional 512-bit SQRT CSLA. Based on gate-level modification 16, 32, 64, 128, 256 and 512-bit SQRT CSLA architectures have been developed by using 8-bit Binary to Excess-1 Converter (BEC) and compared with the regular 512-bit SQRT CSLA architecture. The Performance Analysis of the proposed Modified 512-bit Square-Root Carry Select Adder in terms of Delay, Area, and Power has been carried out by Spartan 3E and synthesized with Xilinx ISE 9.1i software.

The second outstanding design is the area and power optimization for Parallel Prefix Adders. Existing Parallel Prefix Adders are Kogge-Stone Adder (KSA), Spanning-Tree Adder (STA), Sparse Kogge-Stone Adder (SKA) and Brent-Kung Adder (BKA). Phase II of the research work proposes a new Hybrid Parallel Prefix Adder (HPPA) which is a combination of Kogge-Stone and Brent-Kung Adders and exploits the advantages of both the Adders. The proposed Hybrid Parallel Prefix Adder is implemented using Verilog Hardware Description Language (VHDL) and synthesized using Xilinx ISE 13.2 Design Suite for its performance analysis. The performance analysis of the proposed HPPA is better in terms of Area and power optimization in comparison to conventional Adders.

A 14 Transistor Full Adder-Subtractor circuit is proposed in the third approach by considering 180nm channel length. A typical control signal is applied to operate the circuit either as an Adder or Subtractor. The proposed technique produces an optimum performance at a value of 0.6 volts at 180nm CMOS technology using Tanner EDA Tools.

Further Modified Gate Diffusion Input (Mod-GDI) logic design is proposed which is an efficient power optimization technique. The Mod-GDI technique aims to reduce Delay, Area and Power dissipation while maintaining the functionality of the logic design and has been implemented and analyzed for different logic gates. The performance comparison of proposed Mod-GDI technique which is more power-efficient than the conventional CMOS logic design along with the results has been carried out. This work also explores a novel structure of Carry-Select Adder (CSLA) which uses fundamental gate levels designed with proposed Mod-GDI technique. The proposed Mod-GDI technique can be used to reduce the number of transistors compared to the conventional CSLA design for better performance. All logic gates in regular CSLA based on RCA's are replaced by the Modified GDI logic Cells. Performance comparison of standard CSLA with the proposed Mod-GDI based CSLA is carried out. The proposed logic design is simulated by using Mentor Graphics tool with a supply voltage of 1.6V at 90nm Process Technology.

Phase V focuses on the design of Energy Efficient High-Speed and Low-Power Hybrid Multiplier which is a combination of Array Multiplier and Wallace tree Multiplier. Low power digital Multiplier design based on Bypassing technique is mainly used to reduce the switching power dissipation. The proposed prototype of the Hybrid Multiplier Architecture (HMA) has been carried out on Spartan 3E FPGA and performance analysis was done using the Xilinx ISE 9.1i tool suit.

Finally, the design and implementation of FIR filter using high-speed low-power Adder and Multiplier are carried out. The Performance of FIR filter depends on Multiplier and Adder circuits used in the filter. To improve the overall performance of the FIR filter, different Multiplier and Adder combinations are used to reduce the dynamic power consumption as well as semiconductor chip size. The

Low-Power Modified 16-bit Square Root Carry Select Adder (M-SQRT CSLA) is proposed in this work by replacing Half Adders instead of Full Adders. The proposed 16-bit M-SQRT CSLA has been designed to reduce dynamic power consumption. The 16-bit M-SQRT CSLA is applied to the Wallace tree Multiplier for the addition process after the partial product generation stage. Multiply and Accumulate (MAC) unit of the Digital FIR filter is designed by using Modified Wallace multipliers and M-SQRT CSLA. Further, the Group-2, Group-3, Group-4 and Group-5 structures of M-SQRT CSLA were constructed using Half Adders only. Comparison between the proposed 16-bit M-SQRT CSLA with conventional 16-bit SQRT CSLA in terms of Area, Power and Delay has been carried out. From the simulation results it is proved that the proposed M-SQRT CSLA consumes less area and power than all the other methods. Simulation is performed by ModelSim 6.3c and Synthesis process is done by Xilinx 10.1i. Simulation results show that Digital FIR filter with proposed 16-bit M-SQRT CSLA occupies less area and consumes low power.

To summarize, in this work energy efficient high-speed Adders and Multipliers for Digital Signal Processors using new approaches has been evaluated. The proposed methods are found to be area and energy efficient and also offer low-power and less delay in VLSI circuits.

TABLE OF CONTENTS

CHAPTER NO. TITLE PAGE NO.

CHAPTER NO. **TITLE** **PAGE NO.**

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ALU	-	Arithmetic Logic Unit
BC'S	-	Block Cells
BCG	-	Block Carry Generator
BEC	-	Binary to Excess-1 Converter
BKA	-	Brent Kung Adder
CLA	-	Carry Look-ahead Adder
CMOS	-	Complementary Metal Oxide Semiconductor
CSA	-	Carry Save Adder
CSLA	-	Carry Select Adder
DPFPA	-	Double Precision Floating-Point Adder
DSP	-	Digital Signal Processing
EDA	-	Electronic Design Automation
FA	-	Full Adder
FFT	-	Fast Fourier Transform
FIR	-	Finite Impulse Response
FPGA	-	Field Programmable Gate Array
GC'S	-	Gray Cells
GDI	-	Gate Diffusion Input
GUI	-	Graphical User Interface
HKSS	-	Hybrid Kogge-Stone Structure Carry Select based Adder
IC'S	-	Integrated Circuits
IIR	-	Infinite Impulse Response
IOB	-	Input-Output Blocks
ISE	-	Integrated Software Environment
KSA	-	Kogge Stone Adder
LSB	-	Least Significant Bit
LUT	-	Look-Up Table
MAC	-	Multiply and Accumulate

Mod-GDI	-	Modified-Gate Diffusion Input
MSB	-	Most Significant Bit
MUX	-	Multiplexer
NMOS	-	N-channel Metal Oxide Semiconductor
PMOS	-	P-channel Metal Oxide Semiconductor
PPA'S	-	Parallel Prefix Adders
PTL	-	Pass Transistor Logic
RCA	-	Ripple Carry Adder
RTL	-	Register Transfer Level
SKA	-	Sparse Kung Stone Adder
SoC	-	System-on Chip
SQRT	-	Square-root
STA	-	Spanning Tree Adder
TGL	-	Transmission Gate Logic
VHDL	-	Very speed integrated circuit Hardware Description Language
VLSI	-	Very Large Scale Integration

LIST OF SYMBOLS

DIFF	-	Difference
GND	-	Ground
A_i	-	Input 'A' of i^{th} bit to Adder
B_i	-	Input 'B' of i^{th} bit to Adder
C_{in}	-	Input Carry
G_i	-	i^{th} Generate signal
P_i	-	i^{th} Propagate signal
S_N	-	N-type low voltage logic block
T	-	Number of transistors
C_{out}	-	Output Carry
Ps	-	Pico Second
Pw	-	Pico Watt
S_p	-	P-type high voltage logic block
V_{dd}	-	Supply Voltage
V_{th}	-	Threshold Voltage

CHAPTER 1

INTRODUCTION

1.1 FOREWORD

Very Large Scale Integrated (VLSI) systems are ubiquitous in everyone's day-to-day activities. This is the result of advancement in process technology and progress in density coupled with the innovation of faster and energy efficient devices. The major driving force behind this technology revolution over the last few decades is an exponential growth of transistor density within a single chip as described by Moore's law [1]. Because of the singular characteristics of negligible standby power, Complementary Metal Oxide Semiconductor (CMOS) process has evolved as the prominent VLSI technology after Bipolar, since large numbers of transistors can be integrated and accommodated on a chip.

The device dimensions like junction depth, gate oxide thickness, channel length, etc., which are the significant factors of the device must be synchronized and scaled accordingly. However, it increases the dynamic power dissipation as well as the standby power exponentially. This exponential scaling in the power dissipation area and delay is also a desirable characteristic, and thus there is a need for design and development of novel low power circuit techniques to keep on with the progress of the device dimension scaling.

1.2 CHALLENGES IN LOW POWER VLSI

Nowadays, the major requirement for devices is in terms of low power, which is being achieved by the development of low power techniques. This tremendous increase in the requirement of low power consumption for devices is due to increase in battery operated devices and also reduction in the size of the devices.

Last decades, researches were related to the reduction of size and reducing the cost. As of now, power plays a major role in the development of IC chips due to exponential growth of low power electronic devices with high speed operation and increased complexity.

Generally the power consumed by the electronic devices varies with respect to the application involved. For example a battery operated mobile phone of low weight and small size should have long standby time similar to a battery operated laptop which is heavy and large in size. The power dissipation in these devices varies, therefore the methods used to reduce the power dissipation also varies. In the above specified battery operated devices the power dissipation to be reduced would be up to 50 percent of the total power dissipation.

Apart from low end battery operated electronic devices, power dissipation has to be reduced even in non-battery operated electronic devices. These devices are usually high-performance devices such as servers, workstations, etc. The primary goal of reducing the power should consider the manufacturing cost and also ensure long term reliability of the devices in terms of quality. Hence the recent focus is to reduce the power dissipation of these high performance devices by developing technical methods considering the above mentioned factors. Finally the challenge is to provide a solution for long term durability of the device implemented with reduced power dissipation method.

1.3 DESIGN RELATED ISSUES

In VLSI circuits, there are three different design levels which are considered in designing low power consumption circuits. The different design levels are the process technology, circuit layout and the architectural level. A considerable potential for power saving exists at the logic level implementation of combinational circuits which causes transition activity, short-circuit current, and switching capacitance. The chosen logic level design strongly influences each and every parameter related to power dissipation [2]. At the technology level, power consumption is going to scale down at the same rate with channel length technology shrinking day by day. As a result power saving can be achieved by enhancing the fabrication process which includes small feature size, very low voltages, interconnects and insulators with low dielectric components.

The system performance generally depends on the technique applied to design the circuit and implementation of the circuit for a particular application. From the existing literature investigations, it is found that the design of low power circuits is pointed towards designing of logic cells of arithmetic circuits such as adders and multipliers. In this thesis, the above observations and surveys have been considered and low power circuits are designed initially with basic logic gate and extending the concept to a much broader set of combinational arithmetic circuits [4]-[5]. The characteristics such as number of logic gates, execution time related to power dissipation of various conventional logic designs at different operating conditions are compared and analyzed in terms of quality and quantity. With the reduction of power at different design levels, the numbers of transistors are also reduced. The number of transistors in designing a circuit for minimizing the silicon area during fabrication paves the way for compactness of digital logic design.

1.4 NEED OF LOW POWER ADDERS AND MULTIPLIERS

The integrated circuit technology has lots of impact in the fabrication of devices with small size and low power with increased operating speed. Consumption of power in high performance devices increases with an increase in the features. Hence the major challenges to the developers are to design a low power device meeting all the design criteria.

The recent focus in the VLSI field design hierarchy is toward the implementation of low -power dissipation methodology at all the levels. Most of the techniques focus only on power dissipation whereas area and speed optimization are neglected. With the advancement in the semiconductor technology related to fabrication, the area optimization is possible by accommodating a larger number of transistors in a specified area of the chip. Due to tremendous increase in the usage of battery operated devices embedded in various applications, research is focused on the factors needed for reducing the power dissipation in such devices.

In the last two decades, techniques related to power reductions were applied at all levels of the design hierarchy. These techniques vary for different devices. But nowadays low power dissipation techniques are applied in deep submicron level CMOS technology for high performance devices. Power analysis and management has to be considered while implementing and testing high density Integrated Circuit chips. When tested the high density IC's switching activity consumes power more than the normal value. Even while using the parallel testing methods for higher density chips such as SOCs, power dissipation increases extensively. Thereby functional input vectors are applied successively in correlation because the successive test pattern's correlations are generally low.

In combinational circuits such as adders, there is always a limit on speed of operation. This limit is due to the time consumed for propagating the carry through the consecutive bits. The addition of single bit in the adder is due to the sequential propagation of the carry through the bit positions from the LSB to the MSB stage. In order to reduce the carry propagation time in the adder circuits, Carry Select Adder (CSLA) is used in complex Circuits. In the CSLA propagation time is reduced by generating multiple carry bits of each bit individually and generating the sum value. The disadvantage in CSLA is the consumption of large area as it consists of pair of dual Ripple Carry Adder (RCA) to generate overall sum and carry [6].

Nowadays, design of an adder consuming less area and also consuming less power for operation is the need of the digital world and hence research is oriented towards designing of an adder satisfying all the parameters. Hence conventional adders such as RCA are designed to satisfy parameters such as low power and low area [7]. This adder is compact in size but the execution time is more compared to standard adder. Therefore CLA is used for time critical applications. Hence CSLA is generally compromised between RCA and CLA. Thus adders with less area and low power consumption are used in multipliers and digital signal processors.

Multipliers are most commonly used in Digital Signal Processing to perform various functions and also applied in the FIR and IIR implementation. A few decades back in the VLSI Field, designers mainly concentrated on the area constraint. Optimization techniques were developed for reducing the area of the IC chip area. Last decade the main focus of the designers in the VLSI field was on the operating speed, which was a major constraint on the DSP design implementation. Presently, low-power consumption along with reduced area is considered for the fabrication of filters on SoCs, which can be achieved by reducing its dynamic power which is an important part of the total power dissipation.

In digital signal processors the most important elements are the digital filters. Generally, digital filters are of different structures such as Form-I, Form-II and Transposed. Thus, digital filters are selected appropriately based on the power dissipation requirement. These filters are optimized for low power and also for performances related to applications. Generally FIR filters are selected due to linear phase characteristics. This FIR filter also has low co-efficient sensitivity and stability.

The basic building blocks of filters are adders and multipliers. These combinational circuits are the ones consuming maximum power. In the filter MAC unit is the most essential part deciding the power dissipated and the time taken for operation. Therefore the focus is on multiplier. In digital filters such as FIR, consumption of dynamic switching power is reduced by optimization techniques. These techniques are applied to the adder and multiplier circuits to reduce unwanted power dissipation.

1.5 MOTIVATION OF THE RESEARCH

With the development of technology more circuits are being implemented in a single IC chip. Most of the circuits are related to signal processing applications with more capabilities and features, but these devices consume a lot of energy. Today's era, important constraints to be considered are area and speed, but power is now considered as a critical issue in the VLSI implementation. The requirement of low power devices arises mainly due to two reasons, firstly, the operating frequency increases the power dissipation in an IC chip, and secondly due to the increased use of battery operated devices, and hence there is a need for low power design methodologies to be implemented in embedded IC's. In CMOS technology, the transistors are continuously scaled down in terms of their size. This scaling in turn reduces the switching delay and power with respect to the area density. Reducing the size of the transistor in the VLSI field raises many challenges in the circuit design. As the size reduces in resource constrained devices, transistor leakage increases, which in turn degrades the system performance related to area, power and speed. Hence, the need of the hour is optimization of Power, Area, and Delay in VLSI circuit design, which leads to the motivation of this research work.

The main focus of this research is to reduce the appropriate choice for selecting the Adders and Multipliers topologies with the trade-off between Area, Delay and Power consumption. Many technical papers were surveyed to implement the novel approaches by using Logic synthesis and simulation results were carried out by using Xilinx ISE13.2, Tanner EDA tool, Spartan 3e FPGA kit, ModelSim and X-Power Estimator software.

1.6 OBJECTIVE OF THE RESEARCH

This research work mainly addresses on optimization of Adder and Multiplier circuits by different power and area optimization techniques. The goal of the research is to study and analyze the adder and multiplier operations for the filter applications.

In order to realize this goal, the present work has been carried out with the following objectives.

- To design and analyze the proposed Modified 512-bit SQRT CSLA by using an 8-bit Binary to Excess-1 Converter (BEC), so as to achieve high-performance and to compare with regular 512-bit SQRT CSLA. This work evaluates the performance of the proposed design in terms of Delay, Area and Power consumption.

- To propose and implement a Hybrid Parallel Prefix Adder (HPPA) by altering the logic blocks of the regular Parallel Prefix Adders and to analyze its performance.

- To design a 14 Transistor (14 T) Full Adder-Subtractor circuit considering 180nm channel length by using the 1-bit Full Adder logic and to analyze its power consumption in CMOS technology by using various supply voltages.

- To design an efficient and low power Modified Gate Diffusion Input (Mod-GDI) technique by adopting basic Gate Diffusion Input (GDI) architecture and implementing the same for 16 bit Carry-Select Adder (CSLA) architecture.

- To design and estimate the proposed high-speed and area efficient Hybrid Multiplier Architecture (HMA) using Bypassing Technique and to analyze its performance in optimizing area, power and delay.

- Finally, to design and implement a proposed FIR filter using Modified SQRT CSLA (M-SQRT CSLA) and high-speed Multiplier and to analyze its performance.

1.7 ORGANIZATION OF THE THESIS

The thesis is organized in such a way that each chapter deals with the major modules involved in this work.

Chapter 1 gives an Introduction to Very Large Scale Integrated (VLSI) circuit design, several challenges in low power VLSI and its design issues. It also discusses on the research motivation and objective of the research work. A brief organization of the thesis is given at the end of the chapter.

Chapter 2 elaborately discusses on the extensive literature survey carried out on the conventional VLSI systems which include the design and hardware realization of the high-speed, low-power Adders and Multipliers along with their performance metrics.

Chapter 3 .focuses on the proposed Modified 512-bit SQRT CSLA architecture. The proposed work is discussed and analyzed in terms of area, power and delay.

Chapter 4 proposes a Hybrid Parallel-Prefix Adder (HPPA) and its Performance analysis are discussed. This proposed HPPA architecture is compared with the conventional Parallel-Prefix Adders in terms of area, delay and power. This chapter also presents design of 14 Transistor Full Adder-Subtractor circuits and its performance analysis. The simulation results have been presented and summarized.

Chapter 5 describes the performance analysis of Modified Gate Diffusion Input technique (Mod-GDI) which is implemented in 16-bit Carry Select Adder (CSLA). The chapter also discusses the performance analysis of the proposed technique.

Chapter 6 discusses on proposed Hybrid Multiplier Architecture (HMA) and its performance analysis. This chapter also compares the performance analysis of the proposed HMA design with conventional techniques.

Chapter 7 presents high-speed low-power Adders and Multipliers implemented for FIR Filters so as to optimize the three most important parameters Area, delay, and Power. This chapter also presents the simulation results and a comparison of the important parameters of the proposed FIR Filter with conventional FIR filter architecture.

Chapter 8 concludes the thesis by emphasizing the major implications of the research work. A summary of the research contribution and scope for the future work is also furnished in this chapter.

conveniently using the threshold voltage at the circuit level. However, it leads to increase in the circuit delay, degrades the durability of the Adder cells and initiates threshold loss problem. These issues raised can be overcome through selection of an appropriate width and length ratio of the transistor involved in the design. Over the years, intensive researches have been made for high-performance Adder circuits with low-power application.

2.2 LOW-POWER ADDERS

A wide range of contemporary Adder architectures has been surveyed in literature over the past few decades. The Adder architectures can be classified into two broad domains, namely, static and dynamic. The dynamic Full Adders have many benefits for fast switching speed and with a smaller number of transistors, full dynamic range, and limited logic. A distributed logic concept has been applied to the adder design to ensure the number of transistors required for implementing the adder logic is reduced at the cost of reduced lustiness and increased power dissipation [7]. The number of transistors needed for static adder is 2n versus n+2 transistors needed for dynamic adder logic; if there is 'n' input logic functions. Generally there are three different logic concepts available. They are Compact circuit layout, Regular structure, and Fast logic evaluation. These concepts can be applied to static and dynamic Adder architectures in an efficient manner.

Requirements for various important Adders relating to the area and time are shown in Table 2.1. The CRA stands for Carry Ripple Adder and the CSA for Carry Save Adder, Parallel-Prefix Carry Look-ahead Adder (PCLA) and CLA for Carry Look-ahead Adder. The complexity seen in the area and time of several types of Adders is listed with n number of stages in Table 2.1.

Table 2.1 Comparison of Area and Time requirements of various types of Adders

Types of Adders	Area	Time
Carry Save Adder	O(n)	O(√ n)
Carry Ripple Adder	O(n)	O(n)
Carry Look-ahead Adder	O(n log n)	O(log n)
Parallel-Prefix Carry Look-ahead Adder	O(2nlogn)	O(2logn)

2.3 LOW-POWER MULTIPLIERS

Multipliers are most commonly used in Digital Signal Processing to perform various functions and are also applied for FIR and IIR implementation. A few decades back in the VLSI Field, the designers mainly concentrated on the area constraint. Optimization techniques were developed for reducing the area of the IC chip. Last decade the main focus of the designers in the VLSI field was focused on the operating speed which was a major constraint in the DSP design implementation. Presently, low-power consumption along with reduced area is considered for the fabrication of filters on SoCs which can be achieved by reducing its dynamic power which is an important part of the total power dissipation [43].

In digital signal processors the most important elements are the digital filters. Generally, digital filters are of different structures such as Form-I, Form-II and Transposed. Thus, digital filters are selected appropriately based on the power dissipation requirement. These filters are optimized for low power and also for performances related to applications [44]. Generally FIR filters are selected due to its linear phase characteristics. This FIR filter also has low co-efficient sensitivity and stability.

Adder and Multiplier play an important role in Multipliers as well as in Digital FIR filters. Multipliers are the most power consuming components. In DSP systems, Multiply and Accumulate (MAC) is an outstanding unit, which influences the operating speed and the power dissipated. The use of MAC influences most of the DSP system's power consumption, therefore efficient multiplier has to be designed for DSP applications. Dynamic switching power is reduced in the adder, multiplier and FIR filter circuits by applying different power reduction techniques and data transmission techniques. This chapter presents an exhaustive literature survey on the design and implementation of hardware circuits of high-speed Adders, Multipliers and digital FIR Filters known for efficiency, which are implemented to enhance the performance of the arithmetic operations in the digital systems.

2.4 CARRY SELECT ADDER

In 1962, O.J.Bedrij et al., [1] proposed the extremely High-Speed Digital Adder, which generates the multiple-radix carry along with sum selection. The work involves a comparison of the standard quality of hardware and logarithmic delay for 100-bit CSLA with RCA. The carry forward problem seen in the proposed adder was overcome by generating multiple radix carry independently. The carry generated is applied together simultaneously to generate sums.

Ramkumar and Harish et al., [2] have proposed a method where the BEC-1 block replaced the Ripple carry block in the CSLA. Here BEC is an efficient gate-level modification method bringing down the area and power dissipation of SQRT CSLA. The use of BEC in the place of RCA decreases the area and power consumption. Here a 4-bit BEC is applied with a multiplexer.

Ms. S. Manjuiet et al., [6] has suggested and implemented an Area efficient SQRT CSLA technique based on First Zero Detection logic. The authors have proposed a new add-on scheme for the purpose of simplifying the layout and lowering the transistor count to enable further interconnection and reduction in logic area. This scheme neither employs any single inverter nor any buffers, but MUX circuit is not implemented for substituting exclusive NOR gates.

Shanigarapuet et al., [7] has, to their credit the efficient CSLA architectural design, where one RCA and D-Latch are used for providing partial sum and carry. D-Latch is known for consumption of low delay-time when compared to traditional CSLA with Dual RCAs.

Saxena et al., [8] have proposed a method to substitute the D-latch in CSLA by BEC in order to provide a partial sum and carry using a reduced delay, area and power, and also it offers smaller delay for the 128-bit addition. These architectures are used mostly in FIR filter for reduction in dynamic power consumption and meeting computational efficiency.

B. Ramkumar and H.M.Kittur, [9] have suggested a direct approach for improving the speed of addition. A 16, 32 and 64-bit Adder architecture was developed on the basis of this technique and comparison made with the conventional fast Adder architectures. In many of the Parallel Multipliers used for speeding up the final addition, CLA was outlined in the form of CSLA for improving the speed of the Adder. But due to the structure of CSLA a larger area is consumed, as a result of multiple pairs of RCA's used for the generation of the partial sum and carry bits through consideration of C_{in} as 0 and 1 respectively, resulting in the circuit complexity of the final Adder structure being high. Hence, the RCA of CLA with the use of C_{in} as 1 was replaced by BEC logic, which helps in reducing the maximum area but the delay is increased in the final Adder structure.

K. M. Chu et al., [10] has illustrated an Efficient High-Speed CSLA, which works based on the BEC technique. In this Adder type, the RCA block with the input carry of $C_{in}=1$ has been substituted by the BEC block. NMOS stage constructs the BEC logic circuit apart from acting as a pass gate, for n-bit RCA, and an n+1 bit of BEC is used. The process is done with the objective of reducing the area and power demand of the earlier CSLA.

Stone Adder and Brent Kung Adder. Their simulation results were verified using Modelsim 6.4b and synthesis part carried out with Xilinx ISE 10.1i.

A.N. Jayanthi et al., [20] have done a study of the performance of high-speed VLSI Adder circuits. They have implemented and studied adders of 16-bit, 32-bit and 64-bit length and comparison were obtained in terms of delay. From the analysis, it was found that the 16-bit Ling Adder reduced the delay by fifty percent when compared to RCA. However, the circuit is known to have limitations with respect to fan-in. This limitation was solved by the KS Adder which increased however the wiring connections which in turn consumed more power.

Mangesh B Kondalkar et al., [21] have done an investigation on improvement of the Fault Tolerant of SKS Adder. The proposed technique is used for error correction on the basis of inherent-redundancy in the Carry-tree and with the possibility of error detection.

2.6 BASIC GATE DIFFUSION INPUT (GDI) TECHNIQUE

The Gate Diffusion Input (GDI) logic technique has been developed by Morgenstern, et al [23]. GDI logic reduces power consumption, apart from reducing the size of the circuit. The GDI technique advantages lie in the implementation of complex logic functions using two-transistor logic, and restoration of cell swing within the conventional low-power design techniques.

M. VijayaLaxmi et al., [24] has proposed a high-speed asynchronous Hybrid Kogge-Stone Structure Carry-Select based Adder (HKSS-CSA) and has discussed its applications. The analysis of the adder indicates that large numbers of logic levels are used. The paramount propagation delay and different logic resources used by these Adders were analyzed.

A. M. Shams et al., [25] have designed an Adder with the first deals finder circuit operating at low power. This adder also occupies a smaller area when compared to the conventional CSLA circuit. It also makes three basic modifications

CHAPTER 3

DESIGN AND ANALYSIS OF THE PROPOSED MODIFIED 512-BIT SQRT CARRY SELECT ADDER

3.1 INTRODUCTION

In digital combinational circuit execution speed depend on the execution speed of the adder circuit and the simulation time is generally restricted by the time consumed for propagation of carry through the adder. The sum obtained for each bit of a stage is sequentially applied one after the other only when the previous bit has been summed and a carry has been propagated. To solve the problem of propagation delay in the adder circuits, an alternate adder circuit known as the Carry-Select Adder (CSLA) is applied for computational processes. Here multiple carry bits are generated independently and a carry is selected to generate the total sum. Generally Carry-Select Adders are of two types; one is specified as Linear Carry-Select Adder (L-SQRT) and the other is specified as Square-Root Carry-Select Adder (SQRT CSLA). The CSLA has a drawback i.e., it consumes more area due to the existence of multiple pairs of RCAs. These RCAs generates partial sum and carry if and only if the carry input is zero or one [1].The limitation in the adder circuit is related to that of the final sum and carry generated by the stages which are selected by the Multiplexers (MUX) circuit to overcome the above specified limitation. The basic methodology is to apply a 8-bit Binary to Excess-1 Converter (BEC) instead of RCA within the conventional 512-bit SQRT CSLA in order to attain a reduction in area and decrease in the power dissipated. The foremost advantage of applying Binary to Excess-1 Converter (BEC) logic is related to the reduction of total logic gates used when compared to n-bit Full Adder (FA) structure.

A standard CSLA circuit can be implemented in two different modes, firstly in the uniform block mode and later in the variable block mode. In the uniform block mode,the data bits are divided into groups which are of uniform (equal) size throughout the entire design. In the variable block mode, the data bits are divided into different groups of variable (not identical) sizes. Among the two modes of design, the variable mode is the highly recommended design because the propagation delay of carry in this mode is low when compared with the uniform design.

In this chapter a Modified 512-bit SQRT CSLA architecture is proposed and its gate level modification is performed by using an 8-bit Binary Excess-1 Converter (BEC) logic in order to optimize three critical parameters area, delay and power dissipation.

3.2 CARRY SELECT ADDER (CSLA)

The CSLA circuit is generally used in many applications such as computers and digital systems. This circuit reduces the problem which occures due to the propagation delay in the adder circuit. Here multiple carry is generated independently and then a carry selected to generate the sum. The efficiency of CSLA is not improved in terms of size or area as it uses a collective pair of Ripple Carry Adders (RCAs) to generate partial sum and carry. The partial sum and carry are generated by considering the input to carry as 0 or 1. In each adder circuit, MUX is used to generate the final sum and carry. The Carry-Select Adders are usually represented in two forms, firstly as Linear Carry-Select Adder (L-CSLA) and later as Square-Root Carry Select Adder (SQRT CSLA) [2].

3.2.1 Linear Carry Select Adder

The Linear Carry-Select Adder is usually designed by cascade structure constructed by binding equal length of adder stages. For example an n-bit Adder circuit would be implemented with the 'n' equal length of Carry-Select Adder. The regular CSLA is applied in many data computations because it minimizes the problem of delay which was generated due to carry propagation. It is designed by dividing the architecture into small groups comprising of Ripple Carry Adders and Multiplexers (MUX). Advanced computations are performed based on the input carry equal to 'zero' and 'one' for Ripple Carry Adders and the final sum and carry are selected by the MUX. The select line signal is obtained from the carry generated in the previous group. The speed of the regular Linear Carry-Select Adder is high when compared to Serial-bit Adder, but the drawback is in terms of area as it consumes a lot of space to implement.

3.2.2 Square-Root Carry Select Adder

The Square-Root Carry-Select Adder (SQRT CSLA) is usually designed by balancing the delay using signals of block multipliers and carry chains obtained from the previous stage, thereby improving the speed of the Square Root Carry-Select Adder (SQRT CSLA). In this proposed work for 512-bit SQRT CSLA architecture is designed by dividing the 512-bits into 31 groups with different sizes of Ripple Carry Adders. Even though speed is improved by using Square Root Carry-Select Adder, the total area is more when compared to n-bit RCA. In order to overcome the above drawback, SQRT CSLA with BEC is designed so that area is reduced drastically. Here RCA with the carry input of '1' is replaced by employing Binary to Excess-1 Converter (BEC) logic in order to achieve low area [3].

3.3 CONVENTIONAL 512-BIT SQRT CSLA

The conventional 512-bit SQRT CSLA architecture implementation is represented in Fig. 3.1. The design circuit is mostly divided into different groups of varying sizes, i.e., a traditional 512-bit SQRT CSLA consists of 31 groups (stages). Each group has a bit length of various sizes. These bit lengths are broadened using two RCAs and a MUX in each group. Only Group 0 has one Ripple Carry Adder and there is no necessity of a Multiplexer for selecting the data.

In this architecture, starting from Group-0, there is only one 2-bit RCA which adds up the data input bits furnished along with the input carry and generates the sum [1:0] and carry-out bit. The carry-out bit, which is generated by Group-0 will act as a selection line for the next higher group in the sequence, which is Group-1. Based on the selection line from Group-0, the Multiplexer (MUX) selects the corresponding upper RCA (C_{in}=0) or lower RCA (C_{in}=1). In this way, the pattern continues for the remaining groups depending on C_{out} [2] -[4].

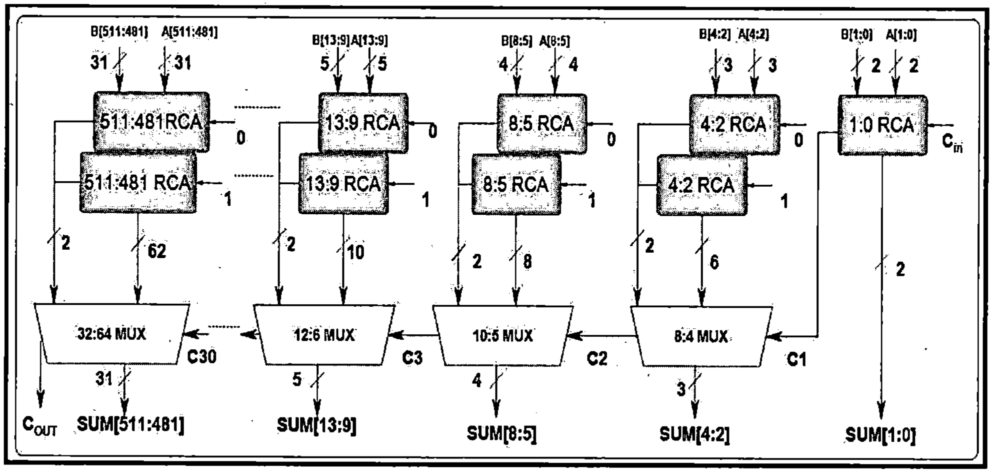


Fig. 3.1 Block Diagram of the Conventional 512-bit SQRT CSLA Architecture

3.3.1 Limitations of the Conventional 512-bit SQRT CSLA

The traditional 512-bit SQRT CSLA has the disadvantage of a larger area because of multiple pairs of RCA structures and a multiplexer which leads to increased area propagation delay and it also not power efficient.

3.4 PROPOSED 512-BIT SQRT CSLA

The proposed method for Modified 512-bit SQRT CSLA (Mod-SQRT CSLA) is implemented and is compared with the conventional design architecture.

The Modified design is obtained in replacing RCA with C_{in} as 1 by BEC logic. The most significant benefit of BEC logic is that it occupies a smaller area but performs similar addition operation compared to RCA with C_{in} as 1. Hence, the existing design architecture has been modified by replacing RCA with BEC.

3.4.1 Function of the Binary to Excess-1 Converter (BEC) Logic Structure

These section discuses the BEC logic design and its importance in the optimization of parameters like Delay, Area and Power consumption implemented in the proposed design. The main purpose of this implementation is to replace the RCA with C_{in} as 1 in the conventional 512-bit SQRT CSLA with BEC for achieving reduced area and increased speed of operation when compared with traditional designs. Fig. 3.2 shows the structure of an 8-bit BEC logic structure.

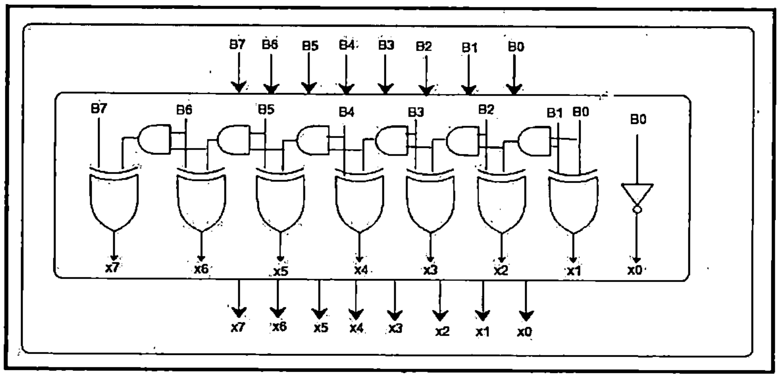


Fig. 3.2 The Structure of 8-bit BEC logic

Designing 12-bit Adder using the traditional RCA, requires 12 Full Adders (FA) which uses an area of 156 gates (13*12) and delay of 82 (12*6) gates. But only 112 gates in the area and 52 gate delays are required for the implementation of the same 12-bit Adder using the BEC. This BEC logic can be substituted in RCA by C_{in}=1 in the conventional designs so as to increase the speed of operation and also in order to occupy less area. Generally, n+1 bit BEC logic is required for substituting an n-bit RCA in the combinational design. Table 3.1 is the functional table of 8-bit BEC.

The logic expressions for the 8-bit BEC cell is given as follows

X0 = ~B0

X1 = B0^B1

X2 = B2^ (B0 & B1)

X3 = B3^ (B0 & B1 & B2)

X4 = B4^ (B0 & B1 & B2 & B3)

X5 = B5^ (B0 & B1 & B2 & B3 & B4)

X6 = B6^(B0 & B1 & B2 & B3 & B4 & B5)

X7 = B7 ^ (B0 & B1 & B2 & B3 & B4 & B5 & B6)

TABLE 3.1 Functional Table of the 8-Bit BEC

B[7:0]	X[7:0]
00000000	00000001
00000001	00000010
.	.
.	.
.	.
11111110	11111111
11111111	00000000

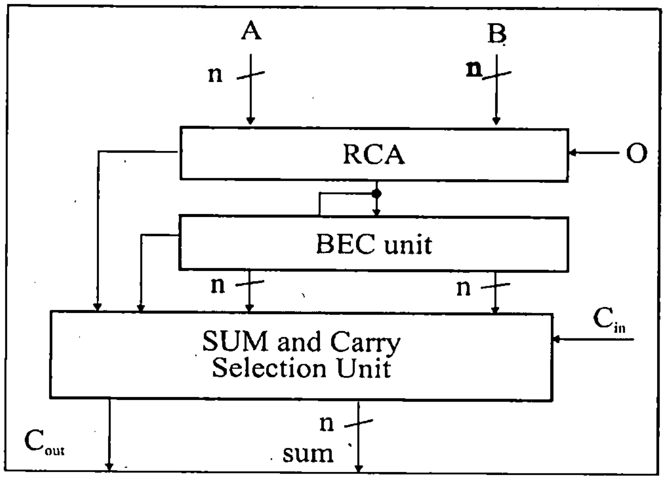


Fig. 3.3 Basic Structure of BEC based CSLA

The Boolean logic expressions shown above are implemented in gate level using the AND, INVERTER and XOR gates. The increase in Boolean logic expressions to any number of bits is possible with the corresponding RCA substituted with this new 8-bit BEC logic design as shown in the Fig. 3.3. The comparison of conventional 512-bit SQRT CSLA Adder with Mod-512-bit SQRT CSLA synthesis is carried out by using the Xilinx ISE 9.1i software simulation tool with Spartan 3E. The Area and Delay of the BEC logic design is reduced when compared with a conventional RCA design as a result of which the BEC logic design is more suitable for high-speed and low-power applications.

3.4.2 Methodology for the Evaluation of Delay and Area in the Basic Adder Blocks

Delay and Area are calculated considering each gate as having a Delay and Area equal to 1 unit respectively. The most extended path is derived from this design. It shows the maximum delay of implementation from the beginning of input bits to the closing of output bits. The total number of AOI (AND, OR and INVERTER) gates gives the gate count for Area calculation.

Table 3.2 shows the gates and area count obtained for different designs. The table shows an analysis of increase in the Area and Delay when the design includes more number of gates.

Table. 3.2 Evaluation of Area and Delay

Adder Blocks	Area (no. of gates)	Delay (no. of gates)
XOR	5	3
2:1 MUX	4	3
Half Adder	6	3
Full Adder	13	6

The evaluation of AND, OR, and Inverter (AOI) implementation of the XOR gate is shown in Fig. 3.4. The gates among the dotted lines indicate the performance of the operations in parallel, while the numeric representation of each gate denotes the delay resulting from that gate. The area and delay estimation technique for all gates is by using the AND, OR and Inverter (AOI) implementation.

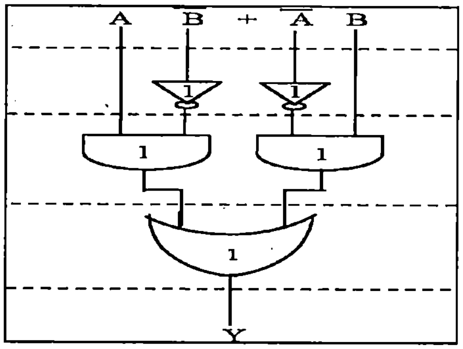


Fig. 3.4 Delay and Area of XOR gate

The conventional and the Mod-512-bit SQRT CSLA architecture are analyzed based on the power. The Fig. 3.11 represents the analysis pertaining to the total power dissipated. The inference from the simulation results shows that the proposed work provides an improvement of 11.1% in power (µW).

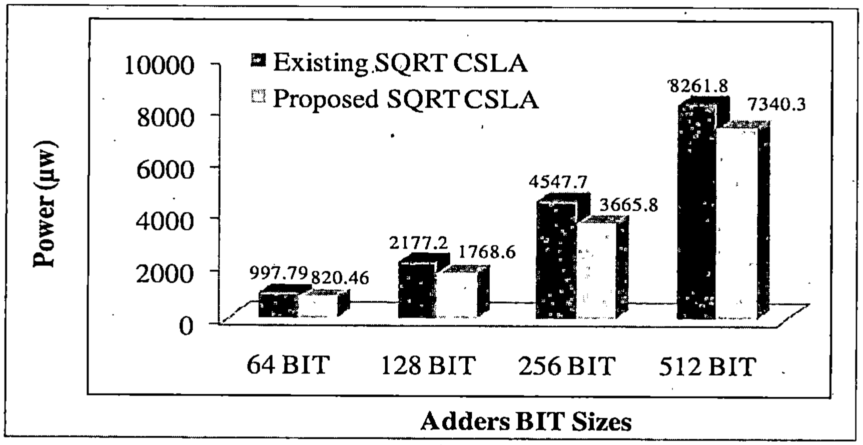


Fig. 3.11 Analysis of Conventional and Modified 512-bit SQRT CSLA in terms of Power (µW)

3.6 SUMMARY

In this chapter implementation of the proposed Mod-512-bit SQRT CSLA architecture has been carried out. In this work alteration of the logic blocks in the conventional SQRT CSLA is performed. The alteration with respect to the logic blocks of the conventional architecture provides more advantages in terms of the reduced area or number of gates consumed. The proposed Mod-512-bit SQRT CSLA has been implemented through use of an 8-bit BEC logic for gate level modification of adder architecture. The BEC logic is used instead of RCA, which offers reduced Delay, Area and low-power in the proposed work. The advantages of Delay, Area and Power of Adder circuits in this technique are 1149 (no. of gates), 112.38 (ns) and 7340.31 (µW) respectively. Hence, an improvement of 25.5% in Area, 16.5% of Delay, and 11.1% of Power has been achieved.

CHAPTER 4

PROPOSED HYBRID PARALLEL-PREFIX ADDER AND 14 TRANSISTOR FULL ADDER/SUBTRACTOR

4.1 INTRODUCTION

The adder circuit is the fundamental circuit applied in digital systems or digital processors. The performance of the adder has great influence in terms of high-speed and accurate computational operation. Improvement in the performance of the Digital Adder can help in speeding the binary operations inside the processes to a considerable extent. Analysis of operating speed, layout area and power dissipation determines the performance of a digital Adder block. In today's world Parallel Prefix Adder (PPA) has become an important due to its capability of implementation on VLSI chips for achieving fast mode of operation [15]. VLSI circuits generally depend on fast and dependable arithmetic computation. The first module in this chapter proposes the design of Hybrid Parallel-Prefix Adder (HPPA) and also its implementation while the second involves in the evaluation of 14 Transistor (14 T) Full Adder-Subtractor architecture using 1-bit Adder-Subtractor circuit design. Parallel Prefix Adders (PPA's) are of different types, few of them are Sparse Kogge-Stone Adder (SKSA), Kogge-Stone Adder (KSA), Spanning-Tree Adder (STA) and Kogge-Stone Adder (KSA). Investigation of Brent-Kung and Kogge-Stone Adders has been studied and analyzed in the first module so has to propose a Hybrid Parallel Prefix Adder with optimized performance metrics.

Recent trends indicate that low power dissipation methodology have become a significant design constraint. Apart from this the transistor count is also considered as a significant factor in designing of Full-Adder-Subtractor circuits. Design of transistor count can affect the design of complexity of an Arithmetic Logic and shift Unit (ALU) and even other functional units. The propagation delay has a lot to do with the design and synthesis of the VLSI circuits. The speed of the Adder design is limited, due to a large number of gates needed for implementation, on the size of the transistor, occurrence of the delay in the critical path and parasitic capacitance involved. The driving capability of a Full-Adder is a major concern for the designer as it is used mostly in a cascade structure where the output of one drives the input for another. Reducing the supply voltage is also a critical process involved in reducing power consumption. However, scaling of the supply voltage also escalates circuit delay. For the reasons, the achievement of low-power and high-speed design with reduced area has been taken up as the research objective of this module.

In this chapter the first module proposes the design and estimation of high-speed new Hybrid Parallel-Prefix Adder (HPPA) architecture. The proposed HPPA is a combination of area-efficient Carry tree Adders like KSA and BK Adders. The simulation results of the HPPA have been used for optimization of crucial factors like area, propagation delay and its power dissipation values have been compared with those of the conventional techniques.

The second module proposes the design and analysis of 14 T Full Adder/Subtractor circuits followed by the implementation with 1bit Full-Adder/Subtractor that generates less propagation delay and has low-power consumption.

4.2 PARALLEL PREFIX ADDERS

Parallel-Prefix Adders perform high-speed parallel addition, which is widely used in microprocessors, DSP-based systems, telecommunications and for high-speed applications. PPA's has the capability to reduce logic circuit complexity and delay; this has great consistence and improves the performance with parameters like area and power. Hence, the PPA's are highly required in high-speed arithmetic circuits.

PPA's, also referred to as Carry-tree Adders, and consists of Pre-compute, Propagate and Generate signals. These signals are aggregated using the major carry operator. The design of generation of carry for the Prefix Adders is possible in many contradictory ways based on the applications [16].

4.2.1 Parallel-Prefix Adder Structure

The conventional Parallel Prefix Adder structure is represented in Fig. 4.1. Here a tree structure form is used by PPA for increasing the speed of an arithmetic operation. PPA's are generally known for their increased computational speed for many digital systems.

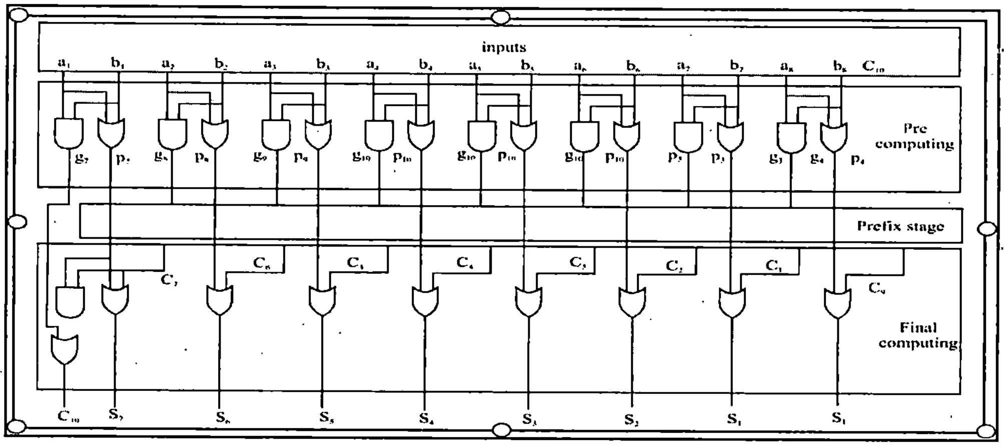

inputs
a_1 b_1 a_2 b_2 a_3 b_3 a_4 b_4 a_5 b_5 a_6 b_6 a_7 b_7 a_8 b_8 C_{in}
g_7 p_7 g_8 p_8 g_9 p_9 g_{10} p_{10} g_{10} p_{10} g_{10} p_{10} p_3 p_3 g_3 g_4 p_4
Pre computing
Prefix stage
C_7 C_6 C_5 C_4 C_3 C_2 C_1 C_0
Final computing
C_{10} S_7 S_6 S_5 S_4 S_3 S_2 S_1 S_1

Fig.4.1 Block diagram of the Conventional 16-bit Parallel-Prefix Adder Structure

The conventional PPA structure is implemented by all the 3 stages specified as shown in Fig. 4.1. The three stages are:

- Pre-processing stage
- Carry generation Network
- Post-processing stage

2

ORIGINAL ARCHIVAL COPY

ANALYSIS AND IMPLEMENTATION OF BINARY ADDITION IN NANOMETER CMOS TECHNOLOGY

BY

JOHANNES GRAD

Submitted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Electrical Engineering
in the Graduate College of the
Illinois Institute of Technology

Approved ______________________
Co-Adviser

Approved ______________________
Co-Adviser

Chicago, Illinois
May 2006

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

UMI Number: 3220877

INFORMATION TO USERS

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleed-through, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

PREVIEW

UMI®

UMI Microform 3220877

Copyright 2006 by ProQuest Information and Learning Company.
All rights reserved. This microform edition is protected against
unauthorized copying under Title 17, United States Code.

ProQuest Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

PREVIEW

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

ACKNOWLEDGMENT

I would like to thank my adviser, Dr. James Stine. He has been my guide, my mentor and my friend throughout my years at IIT. I would also like to thank my co-adviser, Dr. Jafar Saniie. His advice and support have been invaluable. I am very grateful to my defense committee members Dr. Dimitrios Velenis and Dr. Edward Reingold for agreeing to serve on my committee and reviewing this work. I am also thankful to my colleagues and friends Ivan Castellanos, Jun Chen, Vibhuti Dave, Nick Iliev, Nathan Jachimiec, John Moskal and Tanvir Sharif.

This work would not have been possible without the support, motivation and love from my family; my parents Johann and Rosina, as well as my sisters Susanne and Regina. Thank you for your endless patience and encouragement.

PREVIEW

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

TABLE OF CONTENTS

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

PREVIEW

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

LIST OF TABLES

Table Page

PREVIEW

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

LIST OF FIGURES

Figure Page

PREVIEW

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

PREVIEW

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

ABSTRACT

Binary adders see very high utilization in digital systems, making them hotspots of power consumption and thermal dissipation. This work presents new techniques and algorithms for fast addition at low power consumption. The first part presents a comparative analysis of a number of well known adder architectures as implemented in several CMOS technologies. The results are based on fully extracted mask layouts for each adder. It is found that parallel prefix adders have the lowest energy-delay product and that the area and energy consumption of an adder does not strongly depend on the underlying architecture.

In the second part a number of new prefix algorithms are introduced. The new Ling carry-select adder can be used to modify an existing sparse-tree adder into a Ling adder with less delay and equal area. The new single gate-delay carry propagation algorithm reduces the delay of carry propagation from that of an AND-OR-Invert gate to that of a single NAND gate. The algorithm is used in a novel Ling carry-increment adder, which is smaller and consumes less energy than a conventional carry-increment adder. New dual-mode and multi-mode adders are introduced in the final part. These adders provide multiple energy-delay operating points. This is achieved by turning off parts of the prefix tree. As a result, the energy consumption of the adder can be set dynamically depending on the overall system load.

PREVIEW

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

CHAPTER 1

INTRODUCTION

1.1 Motivation

The addition of two numbers is a fundamental operation in digital systems. Speaking in broad terms, any mathematical algorithm contains a certain number of additions and typically its execution time depends directly on the time it takes to add two numbers. It can, therefore, be expected that mathematical algorithms in general will be implemented in such a way as to maximize the number of additions and to minimize the number of more complex operations such as multiplication. As a direct result of this trend it can be expected that binary adders on a computing platform see very high utilization. This in turn provides the basis for a very high return on any investment into the optimization of binary adders, typically focused on three key areas: delay, power and area. A faster adder will typically result in the reduction of runtime of an algorithm or an increase in the throughput of data that can be processed in real time. By the same token, a reduction in power consumption will reduce the amount of energy required to perform an operation or reduce the average power required to sustain a certain throughput of real-time data. Finally, a reduction in area typically results in the reduction of production cost, both by reducing the area consumed on a Silicon wafer and by increasing yield due to a smaller area subject to manufacturing defects. One example would be the implementation of the FFT (Fast Fourier Transform), which is typically used in the compression of still images and streaming video content. If a faster adder is available the time to compress an image is reduced and the maximum resolution of streaming video is increased. In terms of power, the average energy required to compress one image is reduced and the average power required to sustain a certain resolution of streaming video is reduced as well, both of which would result in a longer battery life in a portable device. And if the area of the adder is reduced the overall cost to manufacture the device will be less as

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

well.

Although integrated circuits typically operate in a base-2 number system the fundamental problem of addition can be seen in the base-10 system. Consider the following example:

$$\begin{array}{rcccc} & 1 & 8 & 5 & 6 \\ + & 3 & 3 & 4 & 6 \\ \hline & 5 & 2 & 0 & 2 \end{array}$$

In its most basic form, this operation would be carried out from right to left. The final digit to be computed will be the "5" in column 4, since its value is not known until column 3 is completed, which in turn is not available until column "2" is evaluated, which in turn depends on column 1. The time to perform addition is, in this most basic case, a linear function of the operand-width, assuming the time to evaluate one column is constant. As a result, an addition with twice the operand-width will take twice as long. However, there are a number of algorithms available that can reduce that dependence from linear to as little as logarithmic. For each column the carry into the next column, as well as the column sum has to be computed. The carries are the most critical component, as each carry depends on all the lower carries, and each sum depends on the carry into its column. But no other value depends on the sum. As a result, much effort will be spent on accelerating the carries, since as they become available over the course of an addition, the sums can be computed in parallel to the remaining carries.

For example, consider column 3. Even without evaluating column 2 it is known that there will be a carry into column 4, since $8 + 3$ is greater than 9. At this point, the result in column 4 is independent of the lower columns and both sections can be evaluated in parallel. As a result, the average time to perform addition is reduced, but the worst-case time is still the same, since there are cases where no one column

PREVIEW

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

will add up to more than 9.

This algorithm can readily be improved. Looking at column 2 it can be seen that its sum is 9. It is safe to say that if there is a carry from column 1 into column 2, then there will also be a carry into column 3. Looking at column 4 on the other hand, it can be seen that there will never be a carry out of column 4, since its maximum value can not be greater than 5. As a result, one might perform addition by first computing for each column if it generates a carry (columns 1 and 3 in this example), propagates a carry (column 2), or removes a carry (column 4). The process of removing a carry is also referred to as "to kill a carry". The resulting algorithm is typically referred to as "carry-skip addition". It was first devised by Charles Babbage, who developed mechanical calculators in the 19th century [2, 43][1]. In his words, the process of determining the carry behavior of each column allows the adder to "foresee" incoming carries, rather than to wait for their successive computation:

> "Multitudes of contrivances were designed, and almost endless drawings made, for the purpose of economizing the time and simplifying the mechanism of carriage. In that portion of the Difference Engine in the Exhibition of 1862 the time of carriage has been reduced to about one-fourth part of what was at first required. At last having exhausted, during years of labour, the principle of successive carriages, it occurred to me that it might be possible to teach mechanism to accomplish another mental process, namely — to foresee."

Today, more than 140 years later, the development of fast adders is still being actively pursued and the common theme is still to accelerate carry propagation. New algorithms, addition circuits and trade-off analyses are being published on a regular basis. Part of the reason for this continual activity is the ever progressing refinement of CMOS (complementary metal oxide semiconductor) technology into the deep submicron and nanometer domain. Every generation within these technologies requires the circuit designer to adapt to new problems and limitations, but also provides new

[1] Corresponding to references in the Bibliography

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

possibilities for improvement and ideas. Most importantly, the high integration density of semiconductor devices has made practical the implementation of highly parallel addition algorithms, that would have been impossible or prohibitively expensive in mechanical or electro-mechanical adders. This allows engineers to continue the work first begun by Babbage: to derive new mathematical principles that allow for faster computation of the carry into each column, and to do so with a minimum of power consumption in as small a Silicon area as possible.

1.2 Significance of this Work

Binary addition is an important and fundamental problem within computer architecture and digital system design. It incorporates elements of both, algorithmic optimization and circuit implementation. Any adder design limited to either one domain is bound to fail, as most algorithmically optimal adders will perform poorly when implemented in Silicon. And likewise, any implementation purely focused on optimized circuit performance will occupy a low level of abstraction, precluding a designer an efficient method to search the design space. The design space for binary adders spans several dimensions, most importantly Silicon area, worst-case delay and average power consumption, as discussed earlier.

Adders in digital systems can be found in a spectrum of applications ranging from simple counters and incrementers, to highly sophisticated microprocessors. Each application has unique design constraints, ranging from very high speed, to ultra-low power and compact area. Well known applications are ALUs (algorithmic-logical units), as well as address computation units. Furthermore, subtractors, comparators, multipliers and dividers all rely on fast addition at the core of their operation. Floating point units also require adders as essential components. Moreover, digital signal processing makes extensive use of addition in the implementation of digital filters, either directly in hardware or in specialized DSPs (digital signal processors). In general-purpose microprocessors, integer addition is performed every clock cycle,

PREVIEW

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

resulting in 100% utilization. As a result, any decrease in delay will directly relate to an increase in throughput. However, the high utilization also results in a local maximum of switching activity on the Silicon die. Consequently, adders typically cause thermal hot spots, exacerbating the problem of chip cooling [40]. As these trends worsen with progressing miniaturization into the nanometer range, it is very important to develop new addition algorithms that provide high performance while reducing power consumption.

Although a large number of adders have been proposed, most have only been reported as implemented at the 250nm process node or above. The onset of a number of physical effects below 130nm makes it hard to judge the performance of existing adders on sub-130nm technologies. In addition, there has only been a limited number of surveys of adders implemented under comparable conditions on the same technology. An example is the survey by Oklobdzija et al. [49], based on the model of Logical Effort [58]. However, the simplicity of this model leaves a substantial margin of error, due to physical effects such as velocity saturation, short-channel effects and circuit-based anomalies. Only detailed SPICE-level simulation of netlists with extracted interconnect capacitances yields a level of accuracy necessary to compare high-speed adder architectures. SPICE and its derivatives allow detailed modeling and simulation of non-linear devices and circuits. SPICE was developed by Laurence Nagel in 1971 at the University of California Berkeley.

This work investigates the implementation of binary adders in contemporary deep-submicron CMOS technology. It identifies prefix-tree topologies and proposes new carry propagation algorithms that performs well with modern CMOS technology. CMOS fabrication processes are commonly classified by the minimum possible length of a transistor gate in meter. A smaller gate length translates into increased switching speed of digital gates. A 90nm process is used for all implementations as it represents the current state of the art of commercially available technology. The

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

resulting conclusions and suggestions are helpful to both, designers of new adders and those planning future implementations on 90nm technology or below.

1.3 Organization

This work is organized as follows: Chapter 2 gives an overview of binary addition and its implementation in CMOS logic. Over the course of time a number of algorithms have been proposed, typically based on the implementation technique at the time, which has progressed from electro-mechanical relays, to tubes, discrete diodes and transistors, integrated circuits to ultra large scale integrated system-on-chips. Each algorithm is presented in its historic context and its relevance to modern CMOS technology. An in-depth analysis of existing addition algorithms is performed in Chapter 3. Each adder is implemented in three CMOS technologies, ranging from submicron to nanometer scale technologies. A number of novel algorithms for binary addition are presented in Chapter 4. They are presented in the form of hybrid adders to allow for a clear distinction between novel and well-known parts of each algorithm. Chapter 5 introduces two new classes of adders called dual-mode and multi-mode adders. These new algorithms apply to all sparse-tree adders and allow for multiple energy-delay operating points per adder. Depending on the overall system load these new adders can turn off part of their internal logic to trade propagation delay for energy savings. Finally, Chapter 6 concludes the work and summarizes its impact.

PREVIEW

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

CHAPTER 2

BACKGROUND AND LITERATURE REVIEW

2.1 Introduction

Binary addition is one of the fundamental problems in digital systems design. Adders are an important component of many digital building blocks. The problem of addition can be divided into two parts: the mathematical algorithm, expressed using Boolean arithmetic, and the circuit implementation, typically using logic gates or semiconductor devices like transistors.

This chapter introduces existing work on both the algorithms, as well as the circuit implementation of binary adders. Large scale digital systems are almost exclusively fabricated using CMOS (complementary metal oxide semiconductor) technology. This is due to the fact that CMOS allows for the highest level of integration with good production yield at feasible power consumption, as compared to other technologies. As a result, all circuits in this work are implemented using CMOS technology. The circuit design can either be performed at the transistor level, or at the gate level. Transistor level design is typically referred to as "full-custom" design, since the chip designer has full control over the properties of each individual transistor. As a result, full-custom designs have the highest performance but also require significant design effort, with a large portion of manual design tasks, making this design style only profitable for integrated circuits sold in large quantities, such as general purpose microprocessors. Designs targeted for medium or small quantities require a largely automated design flow, typically referred to as "semi-custom" design. In this approach digital logic is synthesized into a gate-level netlist using a description of the circuit in HDL (hardware description language). Synthesis tools include libraries of adders that can be optimized to match user-defined design constraints. One of the problems of binary addition is to devise a prefix tree to best match the design constraints. Synthesis tools can perform fine-grain optimization of the prefix tree,

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

whereas full-custom designers choose from a small set of well-known prefix trees, as they are performing manual circuit design on the transistor level, which can not be automated to the same degree as semi-custom designs.

The prefix tree is a regular structure of nodes and vertices that computes the carry for each bit position in the adder. Each carry depends on all the lower carries, hence the name prefix tree. A shallow tree uses few gates, but each gate has to drive many other gates, as each carry is an input for all the higher carries. On the other hand, a deep tree with many logic stages requires a larger amount of Silicon area and has more logic gates in the critical path. A number of prefix trees have been published in the literature and are introduced in this chapter.

2.2 CMOS Technology and Domino Logic

The prevailing way of building circuits in VLSI (very large scale integration) technology today is CMOS. This is partly due to economical reasons, with an unprecedented 30 year trend of scaling to ever smaller geometries. The other advantage of CMOS is its power efficiency, which ideally would result in zero static power consumption.

2.2.1 Power Consumption. Unlike bipolar transistors, which require a non-zero bias current, FETs (field effect transistors) as used in CMOS present a capacitive load that is charged and discharged to perform switching activity, but once charged essentially no current is required to sustain the conducting path through the device. As a result, the charging of the gate capacitance is the primary source of current flow in CMOS. The dynamic power consumption of CMOS circuits can be summarized as follows [71]:

$$P_{dynamic} = \alpha \cdot C \cdot V_{dd}^2 \cdot f \tag{2.1}$$

where the individual terms are given as

- α: The activity factor of the circuit, between 0 and 1

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

- C: The total capacitance that is being charged and discharged
- V_{dd}: The supply voltage of the circuit
- f: The clock frequency

It can be seen that the power consumption is determined by how often the capacitance of the circuit is charged to a value of V_{dd}. Conversely, power can be reduced by using a lower V_{dd}, a lower rate of switching, or by lowering the capacitance of the circuit that has to be charged and discharged.

As each wire also acts as a capacitor, the total capacitance C is the sum of all wire and transistor capacitances in the circuit. It takes a finite amount of time to charge and discharge a wire and a transistor, resulting in a certain, minimum time required to switch a logic gate built from transistors and wires. At nanometer scales, the capacitance of transistors grows smaller and smaller as compared to that of the wires. Therefore, is has become increasingly important to reduce the length and density of the chip interconnect, as the wire capacitance becomes a larger part of the overall switching delay and power consumption.

Fabricated devices exhibit non-ideal behavior and exhibit a small amount of current flow even when no switching activity takes place. This so called leakage current was small enough to be negligible in the past. But at nanometer scale geometries, leakage currents are increasing significantly. In addition, with millions of transistors integrated onto a chip even the smallest current through a single transistor adds up to a significant amount for the overall integrated circuit. In practice, leakage current can be mitigated by adjusting the threshold voltage V_t of each transistor. Using a higher value for V_t results in a slower transistor but also results in substantial reductions in leakage current, since it grows exponentially with decreasing V_t. By segmenting a circuit into critical and non-critical paths, high-V_t devices can be used in the non-critical sections. The threshold voltage is set during chip fabrication by controlling

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

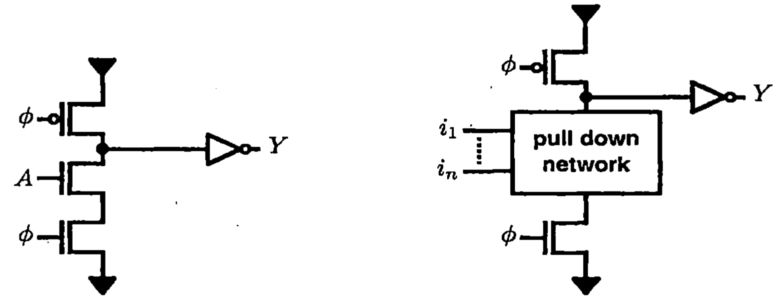


Figure 2.1. Example Domino Logic Gates

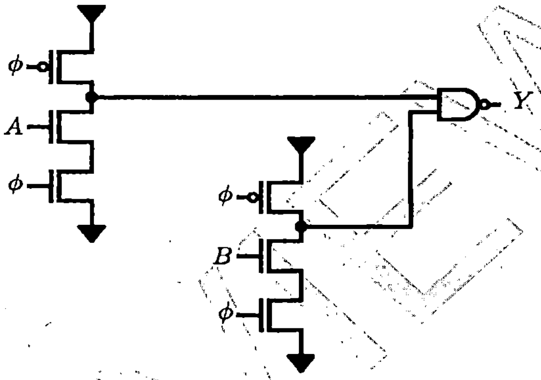


Figure 2.2. Example Compound Domino Logic Gate

PREVIEW

the amount of acceptors that are implanted into the substrate.

2.2.2 Domino Logic. There are several ways to build logic gates from CMOS transistors, typically referred to as logic families. The most commonly used types are static CMOS and domino logic. The advantage of static CMOS is that it results in very robust logic with good noise immunity. This has led to the development of design automation tools that can perform logic synthesis of textual descriptions into static CMOS gates. On the other hand, static CMOS requires a large number of transistors per gate, resulting in higher capacitive loads per gate and, therefore, in reduced switching speed. High-performance circuits on the other hand typically use domino logic, as it requires less transistors per gate. However, domino logic can only be used to construct non-inverting gates, precluding the use of logic synthesis tools.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Examples of domino gates are shown in Figure 2.1. Each gate is controlled by the clock signal ϕ and a number of signal inputs that form a pull-down network. In the simplest case, only one input A is present, resulting in the logic function $Y = A$. The pull-down network can be used to implement any inverting logic function. To guarantee the functionality of the gate, an inverting gate must be used at the output, resulting in an overall non-inverting logic function. The output of a domino gate is 0 by default, unless the inputs form a conducting path in the pull-down network, which results in an output of 1.

2.2.3 Compound Domino Logic. If an inverter is used as the inverting output gate in domino logic it acts as a buffer but does not perform any useful operation. As a result, domino logic can be extended by implementing logic into both the pull-down network, as well as the output logic, resulting in a logic family called compound domino logic. An example is shown in Figure 2.2, where two pull-down networks are joined by an inverting gate. This is faster than using an inverter for each gate and then joining both outputs in a third pull-down network. The logic function that is implemented by this gate is

$$Y = \overline{\overline{A} \cdot \overline{B}} = A + B \tag{2.2}$$

It can be seen that the output function is again non-inverting, as compound domino logic is also bound by the limitation that only non-inverting functions can be implemented. On the other hand, the two pull-down networks implement inverting functions. This makes it possible to use a compound domino logic gate as a container for two inverting functions that are joined by a third inverting function. Another benefit of compound domino logic is that two small pull-down networks are faster than one large pull-down network. Therefore, splitting a large domino gate into a compound domino gate is a way to improve performance. However, one drawback of compound domino logic is the potentially long routing of the dynamic node. Rather than being confined to a single gate, now the dynamic node is distributed over several

PREVIEW

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

gates. This makes it harder to analyse and combat the capacitive coupling onto the dynamic node, exacerbating noise sensitivity of the digital logic.

2.3 Basic Operations in Binary Adders

In this text, individual bits are shown in lower case (a_i) and binary words are ordered from most-significant bit (MSB) to least-significant bit (LSB), e.g $a_{32:1}$. The Boolean OR operation is shown as '+', while AND is shown as '·' or by proximity. The symbol '⊕' is used for the exclusive-or (XOR) operation.

Binary addition in a digital system is defined as the operation of adding two n-bit words $a_{n:1}$ and $b_{n:1}$ and a carry-in c_0, resulting in a n-bit sum word $s_{n:1}$ and a carry bit c_n. Adders are highly hierarchical and at their simplest can be modeled as a set of 1-bit adders. A half-adder (HA), typically referred to as the building block for VLSI adders, computes two bits, s_i and c_i, from two bits a_i and b_i:

$$s_i = a_i \oplus b_i \tag{2.3}$$

$$c_i = a_i \cdot b_i \tag{2.4}$$

A combination of two half adders results in a full-adder (FA), which computes the sum of three bits, where c_{i-1} is chosen as the third incoming bit to stress the fact that it commonly is the carry signal from the previous full adder:

$$s_i = a_i \oplus b_i \oplus c_{i-1} \tag{2.5}$$

$$c_i = a_i \cdot b_i + c_{i-1} \cdot (a_i + b_i) \tag{2.6}$$

A more compact implementation of s_i is possible by incorporating c_i:

$$s_i = a_i \cdot b_i \cdot c_{i-1} + \overline{c_i} \cdot (a_i + b_i + c_{i-1}) \tag{2.7}$$

While half- and full-adders can be used to model binary adders, they are less suitable for a high-speed implementation, which has to compute (2.5) for each bit position as quickly as possible. Since a_i and b_i are available immediately, the limiting

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

UMI

A Bell & Howell Information Company
300 North Zeeb Road, Ann Arbor MI 48106-1346 USA
313/761-4700 800/521-0600

PREVIEW

The Pennsylvania State University
The Graduate School
Department of Computer Science and Engineering

POWER, DELAY AND AREA TRADEOFFS IN CMOS ARITHMETIC MODULES

PREVIEW

A Thesis in
Computer Science
by
Chetana Nagendra

© 1996 Chetana Nagendra

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Doctor of Philosophy

May 1996

UMI Number: 9628150

UMI Microform 9628150
Copyright 1996, by UMI Company. All rights reserved.

This microform edition is protected against unauthorized
copying under Title 17, United States Code.

PREVIEW

UMI
300 North Zeeb Road
Ann Arbor, MI 48103

We approve the thesis of Chetana Nagendra.

Date of Signature

Mary Jane Irwin — March 13, 1996

Mary Jane Irwin
Professor of Computer Science and Engineering
Thesis Co-Adviser
Chair of Committee

13 March 1996

Robert Michael Owens
Thesis Co-Adviser
Professor of Computer Science and Engineering

3/13/96

Ali R. Hurson
Associate Professor of Computer Science and Engineering

Claire Gu — 3/13/96

Claire Xiang-Guang Gu
Assistant Professor of Electrical Engineering

Thomas P. Kelliher — 3/13/96

Thomas P. Kelliher
Assistant Professor of Mathematics and Computer Science
Westminster College
Special Member

Joseph M Lambert — 3/13/96

Joseph M. Lambert
Associate Professor of Computer Science and Engineering
Head of the Department of Computer Science and Engineering

PREVIEW

Abstract

Advancements in VLSI technology have improved gate delay, lowered device area and made it possible to put bigger and more complex designs on a single chip. At the same time, low power design has become important for reliable operation and lower chip packaging costs. Low power design is also essential for battery-driven mobile computing and communication applications. However, our need for faster and faster computing resources requires that low power design be done without affecting performance.

This thesis is a study of the power, delay and area tradeoffs in integer adders and multipliers, which are the most widely used hardware arithmetic modules. Usually, a function can be performed in many different ways using circuits, which although functionally the same, may have vastly different power, delay and area characteristics. This is especially true of adders and multipliers, where research efforts to improve speed have produced a plethora of designs. In view of conserving power, a good design strategy is to use a circuit which consumes the least power while providing the required performance level. This choice is difficult because of the number of degrees of freedom available to a designer and a bad choice can result in a costly implementation.

The goal of this thesis is to facilitate this process by accurately characterizing the power, delay and power-delay product of different circuits. Some degrees of freedom explored in the area of adder design include different algorithms, number systems, voltage scaling and technology scaling. A novel blocked carry lookahead adder design, called ELM, which uses fewer interconnections than the conventional carry lookahead adder, was found to have the best power-delay product. The degrees of freedom studied in multipliers include recoding, pipeline granularity and clocking methodology. It was found that recoding does not always decrease delay, but is useful for reducing power. In heavily pipelined circuits, the clock circuitry was found to consume 50 to 75% of the total power of the system. A new multiplier design based on gated evaluation is proposed which can save power at a small increase in area and delay.

Table of Contents

PREVIEW

PREVIEW

List of Tables

PREVIEW

List of Figures

PREVIEW

PREVIEW

PREVIEW

Acknowledgments

I wish to express my gratitude to my thesis advisors, Mary Jane Irwin and Robert Owens, for their guidance, encouragement and financial support throughout my stay at Penn State. In addition to my thesis work, I learnt about computer architecture and building systems by working on the MGAP project under their leadership. I would also like to thank the other member of my committee, Ali Hurson, Thomas Kelliher and Claire Gu, for their valuable comments and suggestions on my thesis.

During my six years at Penn State, I made some very special friends. Paul Keltcher, Chanda Dharap, Raminder Bajwa and Manjit Borah have not only been invaluable as discussion partners on technical issues, but they have also been my companions for extra-curricular activities. I deeply appreciate the warmth and kindness of Flo Thomas, Vicki Keller, Charlene Hockman and all the secretaries in the department.

None of this would have been possible without the love and encouragement of my parents, Nagendra and Susheela, my aunt, Raja, my sister, Bhavana and my cousins, Jayanth and Suchitra Banavar. Few people go through graduate school without suffering from doubts and depressed moments at some point. My family provided the motivation and support necessary for me to persevere through such times and finish my doctorate. Special thanks goes to my dog, Lulu, who would sit on the window sill beside me, keeping me company when I had to study late at night.

PREVIEW

Chapter 1

Introduction

This thesis is a study of the power, delay and area tradeoffs in CMOS arithmetic modules. Understanding design tradeoffs is an essential part of meeting performance goals and acts as a stepping stone to building low power, high performance and compact data path designs.

1.1 Design Tradeoffs

Since the early days of computer design, speed, area, cost and reliability have been recognized as important design parameters. Much research effort in computer science and engineering has focused on improving the speed and capabilities of digital computing systems. High speed computing has become the accepted norm of computer users rather than the privilege of a few with access to mainframes. Earlier, power was not a concern unless some cooling limit was exceeded. In the 1970's, the increase in the number of devices per chip and clock rates was leading up to a potential power crisis. This was averted by the widespread conversion to CMOS (Complementary Metal Oxide Semiconductor) technology in the late 1970's and early 1980's. CMOS has been successful in other low power areas such as watches, handheld calculators [73], military and space hardware. Since the early 1980's, further power reduction has been possible as a result of lowering power supply voltage which is in part due to the scaling of MOSFET devices. However, the present emphasis on low power design goes beyond such straightforward evolution. It can be attributed to the emergence of new application areas, the primary one being mobile computing; that is, having access to computing resources (be it for computation or for entertainment) at any location without being physically connected to a network. The realization of such a mobile computing environment calls for high performance, compact, light-weight and low power consuming computers capable of computation and communication [23].

Since power, speed and area have been identified as important performance parameters, ideally, it is desirable to have the smallest, coolest, fastest circuit possible.

PREVIEW

This is depicted by the shaded region in Figure 1.1. Unfortunately, all three parameters cannot be optimized simultaneously. The following examples illustrate some of the conflicts.

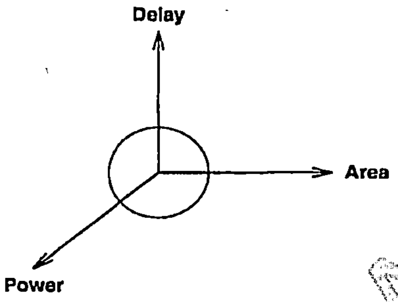


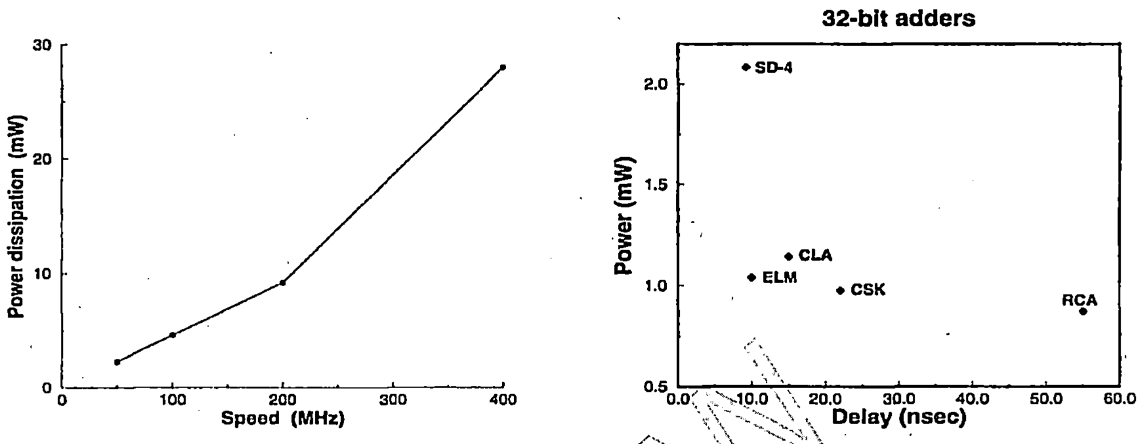
Fig. 1.1. Ideal Circuit with Smallest Area, Delay and Power.

PREVIEW

1. *Delay versus Power*: Figure 1.2 shows two examples. The first is a pipelined multiplier whose power is measured by running it at various frequencies ranging from 50–400 MHz. As speed increases, power dissipation increases as well. In a CMOS gate, the most significant part of the power consumption occurs during transitions when the load capacitances are charging or discharging. This power is proportional to $C_L \times V_{dd}^2 \times f$, where C_L is the load capacitance, V_{dd} is the supply voltage and f is the frequency of operation. In Figure 1.2(a), C_L and V_{dd} are fixed, while f is being varied.

   The second example in Figure 1.2(b) shows the power and delay of some 32-bits adders. These will be discussed in detail in Chapter 2; for now it suffices to notice that, on an average, the faster adders consume more power. In this case, f and V_{dd} are held constant, while C_L varies from adder to adder.

2. *Delay versus Area*: In general, faster algorithms require more complex circuitry to implement and hence require more area. This is true for the 32-bit adders shown in Figure 1.3(a). However, some algorithms, such as the Booth algorithm for multiplication [20], reduce both area and power.

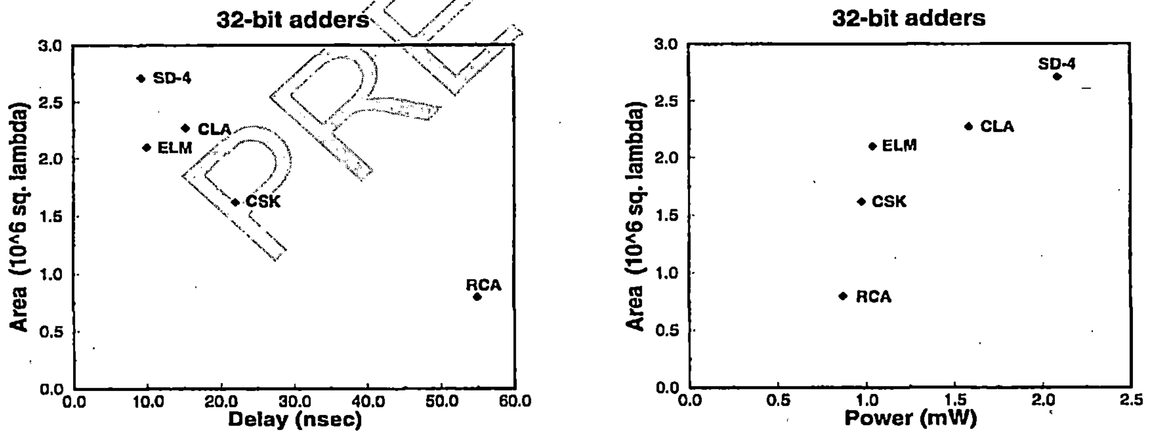


(a) Pipelined multiplier

(b) Adders.

Fig. 1.2. Faster Circuits Generally Consume more Power.

PREVIEW



(a) Faster Adders are Larger

(b) Larger Adders Consume more Power

Fig. 1.3. Relationship between Power, Delay and Area in Adders.

3. *Power versus Area:* Circuits with more transistors or interconnect or both occupy more area and it seems natural to expect them to consume more power. In most cases this is true as for example in most of the adders shown in Figure 1.3(b). However, it has been shown in [75] that there is a difference between technology mapping for power and for area. Tiwari et al found that while circuits optimized for power consume about 10.2% less power than circuits optimized for area, the former occupy 12.3% more area on an average. An example illustrating the difference for gates derived from the `lib2.genlib` library in the SIS distribution is shown in Figure 1.4.

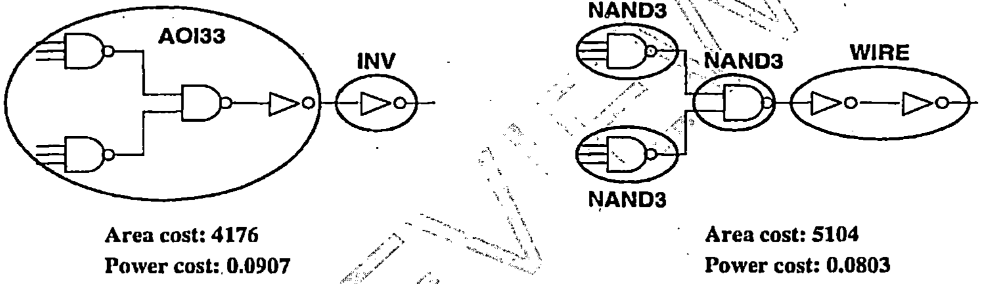


Fig. 1.4. Conflict between Technology Mapping for Area and Power.

PREVIEW

Thus it becomes clear that it is not easy to optimize power, area and delay in the same circuit since improving along one design dimension may increase the other two. Instead, the most important of these parameters has to be optimized, even if it is at the cost of the others. The relative importance of the various parameters depends on the application domain. In the computer/electronics industry, three main categories can be identified, each with differing needs.

1. In micropower battery-operated portable systems such as Personal Digital Assistants (PDA's), the design goal is minimum power/maximum battery life with the required level of performance. Low power design is the prime consideration since the only source of power is the battery. They should also be compact, but performance goals are not very high since the interface is directly with a human operator.

Table 1.1.
Three Design Goals.

Category	Area	Speed	Power
Personal Digital Assistants	Compact	Reasonable	Very low power
Portable Computers	Small	Fast	Low Power
Desktop Workstations and Mainframes	Large	Very fast	Reasonable

PDA's on the market come with a 25–33 MHz 486-generation microprocessor and weigh under 5 pounds.

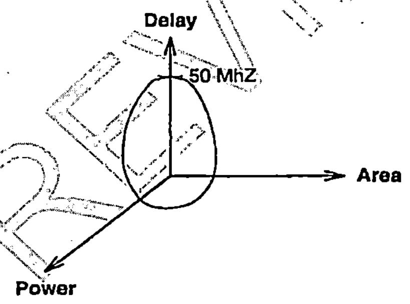
2. In portable PC's (laptops and notebooks), the goal is to achieve performance close to those of desktop PC's such that the overall system power requirement is very low. Like PDA's, they also come with a battery pack, but they can also be plugged into an AC electrical connection in a hotel room, for example, or to a home or office computer. Therefore, their power levels can be higher than those of PDA's. The system is usually designed for maximum performance at a desired power level. The notebooks currently available in the market come with a 75–120 MHz 486 or Pentium generation microprocessor; in comparison desktop PC's run at 100–133 MHz [29]. Also, notebooks should be compact (A4 size) and light-weight. Today's notebooks weigh between 4 and 10 pounds.

   In the late 1980's and early 1990's, the graphics and I/O subsystems consumed the majority of the power in a portable computer. The trend is reversing and future trends show that the CPU is the next main area to concentrate on [68]. Thus, one of the main goals in this category is to reduce CPU power.

3. In nonbattery powered systems such as mainframes, servers, workstations, PC's, etc. the goal is to maximize performance. Till recently, area and power consumption were usually considered secondary issues and high performance microprocessors consume excessive amounts of power. For example, the 300 MHz DEC Alpha consumes 50W of power. With the advent of deep submicron technology, power dissipation is becoming the new limiting factor for the number of devices that can be packed on one chip. The US Environmental Protection Agency (EPA) estimates

that over 80% of electricity consumption by office equipment can be attributed to computers, printers and related equipment. There is a drive towards lowering this percentage by using power managed computers and power conscious designs [1].

Table 1.1 summarizes the three design goals. Since the goal to be achieved along any one of the three design dimensions (namely, area, power, and delay) is dependent on which of the three categories the design falls under, there is scope for tradeoffs without sacrificing the desired level of performance. Take, for example, a design which is required to have a maximum performance of 50 MHz (a future generation PDA, for example). In this case, it is not necessary to build the fastest possible circuit. Instead, it suffices to make the design run fast enough to achieve the required performance of 50 MHz as shown in Figure 1.5. If it is possible to design the components of the system in many different ways, with different speeds and complexities, then a good design strategy is to pick that design which achieves this goal, and has the smallest power and area.



PREVIEW

Fig. 1.5. Trading off Speed for Area and Power.

Another example will make the idea of design tradeoffs clear. Figure 1.6 compares a linear time manchester carry chain adder (MCC) with a logarithmic time carry lookahead adder (CLA). An 8-bit MCC is almost as fast as an 8-bit CLA, but consumes 38% less power. On the other hand, a 32-bit CLA is more than twice as fast as a 32-bit MCC, but consumes 35% more power on an average. Thus, for wordlengths of up to 16 bits, the simple linear adder presents a lower power-delay product, in other words, a better tradeoff between power and delay when compared to a more complex carry lookahead scheme. However, for longer word lengths the latter presents a lower power-delay product.

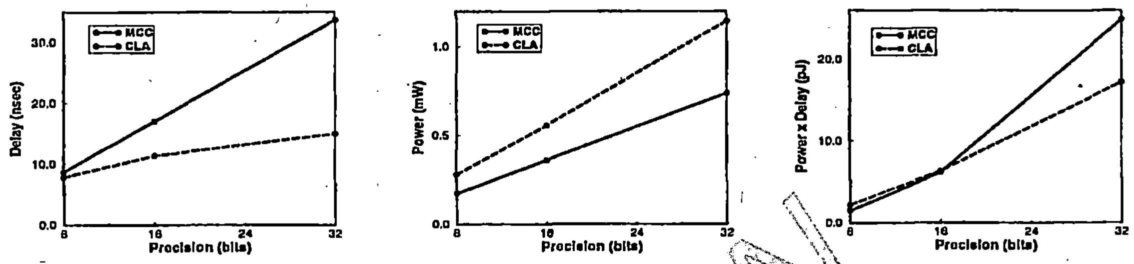


Fig. 1.6. Variation of Delay and Power of MCC and CLA with Precision.

A given function can usually be performed in many different ways which, although functionally the same, may have vastly different power, delay and area characteristics. This is especially true of arithmetic circuits where the wrong algorithm or circuit style can result in an expensive design. The task of choosing the appropriate design is governed by the performance guidelines that have to be met. This is where design tradeoffs play an important role. This thesis is a study of various power, delay and area tradeoffs in CMOS arithmetic modules. Research in computer arithmetic has fostered a burst of activity intended at improving the speed of adders and multipliers as evinced by the large number of available designs [35, 78]. The experiments conducted here are aimed at characterizing and gaining an insight into the effect of various design decisions on the power, delay and area of the module. Such a characterization would help in developing more accurate architectural-level power estimators, which are essential for a high-level synthesis tool for low-power, high-performance architectures.

PREVIEW

1.2 Sources of Power Dissipation in CMOS

CMOS has gained popularity over bipolar and NMOS technologies because of its low power dissipation, high packing density, and high yield [79]. Hence, this technology was chosen for the work presented here. Before looking at ways to minimize power dissipation, the sources of power dissipation in CMOS must be identified. There are three major sources of power dissipation in CMOS circuits as summarized by the following equation:

$$\begin{aligned} P_{\text{total}} &= \text{switching power} + \text{short circuit power} + \text{leakage power} \\ &= C_L \cdot V_{dd}^2 \cdot f \cdot p_f + I_{\text{sc}} \cdot V_{dd} \cdot f \cdot p_f + I_{\text{leakage}} \cdot V_{dd} \end{aligned}$$

The first term represents the switching or the dynamic component of power which is due to the charging and discharging of load capacitances, C_L, where V_{dd} is the supply voltage, f is the clock frequency, and p_f is the activity factor of the circuit, The second term is due to the short circuit current I_{sc}, which arises when both the n- and the p-transistors are on for a short period of time during 1 → 0 or 0 → 1 transitions. The short circuit power dissipation can be reduced by proper circuit design. When the inputs are steady, either the p- or the n-transistors (but not both) are on. However, there is some small reverse bias leakage current $I_{leakage}$ between the diffusion regions and the substrate. This static dissipation is negligible (1-2 nano-watts in an inverter operating at 5V) and is normally ignored. The dominant term in a well designed circuit is the switching component which can be lowered by reducing any one or more of p_f, C_L, V_{dd} and f, while retaining the required speed and functionality. The choice of the algorithm chosen to implement a function significantly affects the activity factor p_f and the load capacitance C_L, both of which influence the power consumption. For example, in Figure 1.6, the MCC has a lower p_f and hence lower power consumption compared to the CLA. However, speed considerations may prevent us from always choosing the algorithm which consumes the least power.

1.3 Related Work

Research directed towards improving area, delay and power spans many fields – from manufacturing technology to architectural and software design. Low area directly translates to low cost while high speed requires no motivation other than our need for faster and faster computing resources. In contrast to area and delay, low power has emerged as an important parameter in more recent times. The motivation for low power design comes from various quarters.

- Shrinking feature sizes combined with larger chip areas allows the packing of an increasing number of devices per chip. Higher levels of integration and speeds have resulted in increased power consumption and heat dissipation which leads to reliability issues and higher packaging costs. Thus, low power design is essential for maintaining reliability (to avoid electron migration and hot electron effect due to higher on-chip electric fields [5]) and reducing manufacturing costs.

- In battery driven mobile computing and communication applications, low power design directly translates to longer battery life and lower cost.

- The EPA is urging the lowering of power usage by computers and related equipment.

The relevant work in various fields, aimed at studying the pros and cons of using one design over another, are summarized below.

1.3.1 Manufacturing Technology

PREVIEW

Miniaturization of MOSFETS is beneficial to both area and delay. Shrinking dimensions coupled with supply and threshold voltage scaling can greatly reduce power dissipation as well [41, 24, 21]. However, lower voltages adversely affect speed [17]. Better VLSI technology is also enabling the building of larger and more complex chips, increasing power dissipation density. This directly affects packaging costs, hence there is a strong market force for lowering power dissipation. Liu and Svensson [41] have shown that power reductions of about 40× are possible without loss of speed by scaling supply voltage down to the 0.48 V–0.13 V range in sub-micron technology. By optimizing threshold voltage alone, they found an 8× power savings without loss of speed. Davari et al [21] from the IBM Research Division have illustrated the importance of optimizing the choice of supply voltage by showing two different scenarios of scaling – one for high speed and the other for low power. They conclude that about 7× speed improvement and more than two orders of magnitude improvement in power-delay product are possible by scaling bulk CMOS down from the present day 0.6 micron at 5 V to the sub-0.1 micron regime. However, they expect power density to rise 4× for the high speed scenario, that is, where supply voltage is not scaled in proportion to the feature size. They speculate that implementing scaled CMOS on silicon-on-insulator (SOI) [66] can lead to reduction in power dissipation due to a reduction of parasitic capacitances and body effect. However, they also mention that this improvement is only on the order of 1.5× – 2.5× compared to bulk CMOS. Thus, once the limits of scaled CMOS are reached (in the not too distant future), circuit design, architecture and software will play an important role in lowering power consumption.

Adiabatic computing is another field of research which involves energy recovering techniques to lower the dynamic dissipation of CMOS [82, 22, 2]. This is achieved at a tremendous increase in gate delay, so this may be of value only in applications which do not require high speed operation. Research in this area is still in its infancy and much work is needed before it can be used to build real machines.

It is evident that technological advancements provide scope for improving speed and power dissipation. However, it calls for industry involvement and changes in the manufacturing process. The other fields investigate approaches to low power design applicable across all feature sizes and supply voltages.

1.3.2 Circuit Design Style

Even though CMOS has been widely accepted because of its low power consumption features, it still provides the choice of many different logic styles and clocking schemes. Some of the different approaches that can be taken for implementing a function are (i) conventional fully complementary static, (ii) dynamic and (iii) pass-gate. Figure 1.7 shows the basic structure of some popular logic styles. A detailed discussion of static and dynamic circuits may be found in [79]. While static implementations are easier to design, they suffer from glitches or spurious transitions which increase switching activity within gates and correspondingly the power drawn by the circuit. Dynamic circuits like domino, on the other hand, undergo exactly one transition and thus avoid glitching. However, their power consumption may not be less than that of static CMOS because dynamic logic involves the overhead of precharging. Dynamic circuits come with a host of associated problems like charge sharing, and capacitance coupling. Thus, even though dynamic gates tend to be smaller (fewer transistors) than static CMOS gates, they require more careful design.

Conventional static CMOS gates consist of complementary n- and p-blocks. Clocked CMOS logic (C^{2}MOS) [73] gates include clocking transistors in series and are useful in forming clocked gates which incorporate latches. The gates have the same input capacitance as conventional static gates but have larger rise and fall times due to the series clocking transistors. By using a single static CMOS buffer with each gate, the domino logic [36] requires a single clock for precharging and evaluating a cascaded set of such blocks. A significant advantage is that the entire p-block is replaced by a single precharge transistor, thus saving area and power in complex gates. Another advantage is that NOR structures become very fast due to the elimination of the slow series p-pull up chain. The main limitation of domino logic is that only non-inverting structures are possible. This is overcome in arithmetic logic circuits which use XOR gates extensively by implementing the necessary XOR gate using static CMOS. Lee et al [38] compared XOR gate designs in various CMOS digital families and found the static CMOS design to have the best power-delay product. Pass-gate logic is attractive because XOR's, multiplexers and registers can be designed using very few transistors. However, a high voltage degrades when it passes through an n-transistor. To avoid this, transmission gates have both an n- and a p-transistor.

The main difficulty arises because the same style is not the best for all kinds of applications. Researchers have used variations of the basic logic styles outlined above. For example, cascode voltage switch logic (CVSL) [25] is a differential style of logic requiring both true and complement signals (dual rail) to be routed to gates. CVSL gates may be thought of as two domino gates with a minimized logic tree. Unlike domino logic, it is a complete logic family at the expense of extra routing, area and complexity. Yano et al [81] have reported that complementary pass logic (CPL) has lower power dissipation than static CMOS. A CMOS logic family based on additional transistors characterized by a modified threshold voltage, resulting in fast circuits that consume less power, is presented in [43].

Traditionally, transistor sizing and reordering have been used to improve the speed of a gate [33]. Their effects on the power consumption of a CMOS gate are evaluated in [74, 10] and it is shown that as much as 30% savings may be obtained, with no increase in delay, by sizing judiciously.

Asynchronous or self-timed logic can be achieved by carefully matching delays between components or by encoding completion within data signals [80]. While the former is very difficult to design, the latter involves the overhead of generating completion signals. Self-timed logic avoids spurious transitions since parts of a circuit are activated only when their inputs become available.