UNIVERSITI PUTRA MALAYSIA

A NEW FRAMEWORK FOR MULTI-PARTITION FINITE STATE MACHINE LOW POWER DESIGN BASED ON GENETIC ALGORITHM

SEYEDHOSSEIN MASOUMIDEZFOULI

FK 2019 160

A NEW FRAMEWORK FOR MULTI-PARTITION FINITE STATE MACHINE LOW POWER DESIGN BASED ON GENETIC ALGORITHM

By

SEYEDHOSSEIN MASOUMIDEZFOULI

Thesis Submitted to the School of Graduate Studies, Universiti Putra Malaysia, in Fulfilment of the Requirements for the Degree of Master of Science

February 2018

Abstract of thesis presented to the Senate of Universiti Putra Malaysia in fulfilment of the requirement for the degree of Master of Science

A NEW FRAMEWORK FOR MULTI-PARTITION FINITE STATE MACHINE LOW POWER DESIGN BASED ON GENETIC ALGORITHM

By

SEYEDHOSSEIN MASOUMIDEZFOULI

February 2018

Chairman : Fakhrul Zaman bin Rokhani, PhD
Faculty : Engineering

At present, the power consumption of digital systems receives substantial attention due to the demand for long-lasting batteries for portable devices, such as cameras, smartphones, and various electronic gadgets. These devices are control based and keep running throughout their lifetime. Given that most of these devices use digital controllers implemented as finite-state machines (FSMs), the reduction of an FSM's power consumption is essential. FSMs can control a simple transmission or control different components of an extensive system on chip design. High-level low-power design techniques are used to reduce the power consumption of digital circuits.

Many researchers use a variety of algorithms to optimize a solution. Among these methods is the genetic algorithm (GA)-based optimization, that technique shows a high performance to reduce FSM power consumption. Partitioning is one of the high-level low-power techniques which reduces FSM power consumption by dividing the logic of the FSM into smaller areas of active and inactive circuits. Therefore, the activation of smaller areas of logic circuits at a time reduces the total power consumption of the FSM.

Partitioning of two areas (bi-partitioning) used in conjunction with the GA methods reduces the power consumption of FSMs. Advancements in low-power implementation methods such as power-gating and clock-gating in semi-custom IC design guarantees the practicality of the methods used for partitioning. Therefore, by using such techniques, an innovative multi-partitioning solution can be achieved for a low-power FSM design using the GA algorithm.

This dissertation proposes a new framework to a multi-partitioning approach to implement FSMs, which aims to improve low power design. Multi-partitioning reduces the power consumption of an FSM by increasing the number of partitions. Therefore, the

logic for each partition becomes smaller and accordingly reduces the power consumption of the FSM. However, by increasing the number of partitions, the overhead of powering control circuitries increases for each new partition to the logic circuit.

Experimental results performed to a set of selected benchmarks shows the power consumption saving for each FSM topology tested. This thesis compares novel approaches to multipartition an FSM alongside standard methods using readily accepted benchmark tests. Standard monolithic and bi-partitioned forms are tested to validate the findings of previous studies. Then accordingly, this study makes a comparison between those results and multi-partitioned results.

Experimental results show that the multi-partitioning methods innovated in this study are effective at reducing the power consumption of FSMs. Furthermore, reports a threshold limit to the number of partitions practically made to certain types of circuit, this is due to the overhead power consumption of the control logic, whereby over-partitioning may exceed the total power consumption of bi-partitioned FSM and monolithic FSM.

In conclusion, this study successfully applies innovated FSM circuit topologies to selected FSM benchmark tests which report a more significant power consumption reduction compared with monolithic FSM power consumption and conventional bi-partitioning GA methods. This approach consistently improves the results of the GA process and shows reliable performance regarding power optimization.

Abstrak tesis yang dikemukakan kepada Senat Universiti Putra Malaysia sebagai memenuhi keperluan untuk ijazah Master Sains

RANGKA KERJA BARU UNTUK REKA BENTUK KUASA RENDAH MESIN KEADAAN TERHINGGA BERBILANG-PARTISI BERDASARKAN ALGORITMA GENETIK

Oleh

SEYEDHOSSEIN MASOUMIDEZFOULI

Februari 2018

Pengerusi : Fakhrul Zaman bin Rokhani, PhD
Fakulti : Kejuruteraan

Pada masa ini, penggunaan kuasa sistem digital mendapat perhatian yang ketara kerana permintaan untuk bateri tahan lama untuk peranti mudah alih, seperti kamera, telefon pintar, dan pelbagai alat elektronik. Peranti ini adalah berdasarkan kawalan dan terus berjalan sepanjang hayat mereka. Memandangkan kebanyakan peranti ini menggunakan pengawal digital yang dilaksanakan sebagai mesin keadaan terhingga (FSM), pengurangan penggunaan kuasa FSM adalah penting. FSMs boleh mengawal penghantaran mudah atau mengawal komponen yang berbeza dari sistem yang luas pada reka bentuk cip. Teknik rekabentuk kuasa rendah peringkat tinggi digunakan untuk mengurangkan penggunaan kuasa litar digital.

Ramai penyelidik menggunakan pelbagai algoritma untuk mengoptimumkan penyelesaian. Antara kaedah ini ialah pengoptimuman berasaskan algoritma genetik (GA), teknik itu menunjukkan prestasi yang tinggi untuk mengurangkan penggunaan kuasa FSM. Pemisahan adalah salah satu teknik berkuasa rendah peringkat tinggi yang mengurangkan penggunaan kuasa FSM dengan membahagikan logik FSM ke kawasan kecil litar yang aktif dan tidak aktif. Oleh itu, pengaktifan kawasan litar logik yang lebih kecil pada satu masa mengurangkan jumlah penggunaan kuasa FSM.

Pemisahan dua bidang (bi-partitioning) yang digunakan bersamaan dengan kaedah GA mengurangkan penggunaan kuasa FSM. Kemajuan dalam kaedah pelaksanaan kuasa rendah seperti kuasa gating dan clock-gating dalam reka bentuk IC separa adat menjamin kepraktisan kaedah yang digunakan untuk pembahagian. Oleh itu, dengan menggunakan teknik tersebut, penyelesaian berbilang-partisi yang inovatif boleh dicapai untuk reka bentuk FSM kuasa rendah menggunakan algoritma GA.

Disertasi ini mencadangkan kerangka baru untuk pendekatan berbilang-partisi untuk melaksanakan FSM, yang bertujuan untuk meningkatkan reka bentuk kuasa rendah. berbilang-partisi mengurangkan penggunaan kuasa FSM dengan meningkatkan bilangan partition. Oleh itu, logik untuk setiap partition menjadi lebih kecil dan dengan itu mengurangkan penggunaan kuasa FSM. Bagaimanapun, dengan meningkatkan bilangan partition, overhead litar kawalan berkuasa meningkatkan setiap partition baru ke litar logik.

Keputusan eksperimen yang dilakukan pada satu set tanda aras terpilih menunjukkan penjimatan penggunaan kuasa bagi setiap topologi FSM yang diuji. Tesis ini membandingkan pendekatan novel untuk membahagi-bahagikan FSM bersama dengan kaedah piawai menggunakan ujian penanda aras yang mudah dikecualikan. Borang monolitik dan bi-partition piawai diuji untuk mengesahkan penemuan kajian terdahulu. Oleh itu, kajian ini menjadikan perbandingan antara hasil dan keputusan berbilang-partisi.

Keputusan eksperimen menunjukkan bahawa kaedah berbilang-partisi yang inovatif dalam kajian ini adalah berkesan untuk mengurangkan penggunaan kuasa FSM. Tambahan pula, melaporkan had ambang kepada bilangan partition yang praktikal dibuat kepada jenis litar tertentu, ini disebabkan oleh penggunaan kuasa overhed logik kawalan, di mana pembahagian lebih banyak mungkin melebihi jumlah penggunaan kuasa FSM yang dibahagikan dan FSM monolitik.

Kesimpulannya, kajian ini berjaya menerapkan topologi litar FSM yang inovatif kepada ujian penanda aras FSM terpilih yang melaporkan pengurangan penggunaan kuasa yang lebih besar berbanding penggunaan kuasa FSM monolitik dan kaedah bi-partition GA yang konvensional. Pendekatan ini secara konsisten meningkatkan hasil proses GA dan menunjukkan prestasi yang boleh dipercayai mengenai pengoptimuman kuasa.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my supervisor Dr. Fakhrul Zaman Rokhani for the continuous support for my master study and research, for his patience, motivation and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. I would like to extend my appreciation to Associate Prof. Dr. Syed Abdul Rahman Al-Haddad my co-supervisor for his support, motivation and patience for this study and research.

I thank my fellow labmates in Embedded and Communication System Lab in Universiti Putra Malaysia, Department of Computer and Communication engineering.

Last but not the least, I must express my very profound gratitude to my parents Seyed Ahmad Masoumi and Saeideh Mousavi for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thank you.

This thesis was submitted to the Senate of the Universiti Putra Malaysia and has been accepted as fulfilment of the requirement for the degree of Master of Science. The members of the Supervisory Committee were as follows:

Fakhrul Zaman bin Rokhani, PhD
Associate Professor Ir. Ts.
Faculty of Engineering
Universiti Putra Malaysia
(Chairman)

Syed Abdul Rahman Al-Haddad bin Syed Mohamed, PhD
Professor
Faculty of Engineering
Universiti Putra Malaysia
(Member)

ROBIAH BINTI YUNUS, PhD
Professor and Dean
School of Graduate Studies
Universiti Putra Malaysia

Date: 29 October 2019

Declaration by Members of Supervisory Committee

This is to confirm that:

- the research conducted and the writing of this thesis was under our supervision;
- supervision responsibilities as stated in the Universiti Putra Malaysia (Graduate Studies) Rules 2003 (Revision 2012-2013) are adhered to.

Signature:	
Name of Chairman of Supervisory Committee:	Associate Professor Dr. Ir. Ts. Fakhrul Zaman bin Rokhani
Signature:	
Name of Member of Supervisory Committee:	Professor Dr. Syed Abdul Rahman Al-Haddad bin Syed Mohamed

TABLE OF CONTENTS

		Page
ABSTRACT		i
ABSTRAK		iii
ACKNOWLEDGEMENTS		v
APPROVAL		vi
DECLARATION		viii
LIST OF TABLES		xii
LIST OF FIGURES		xiii
LIST OF ABBREVIATIONS		xvii
CHAPTER		
1	**INTRODUCTION**	1
	1.1 Background of the Study	1
	1.2 The Significance of the Genetic Algorithm (GA) based Power Optimization	3
	1.3 The Scope of the Study	3
	1.4 Problem Statement	4
	1.5 Objectives	4
	1.6 Thesis Organization	5
2	**LITERATURE REVIEW**	6
	2.1 Digital Circuit Power Dissipation	6
	2.2 Dynamic Power Component	7
	2.3 Static Power Dissipation	11
	2.4 Summary of Power Dissipation	12
	2.5 Digital System Components	12
	2.5.1 Combinational Circuit Design	13
	2.5.2 Sequential Circuit Design	14
	2.6 Finite-State Machine	15
	2.6.1 Benchmarks	16
	2.7 Finite-State Machine Power Dissipation	17
	2.8 Finite-State Machine Power Optimization	18
	2.8.1 State Assignment Algorithms	26
	2.8.2 Partitioning Methods	29
	2.8.3 Power Models	29
	2.8.4 Power Estimation Tools	30
	2.9 Summary and Conclusion	31
3	**METHODOLOGY**	32
	3.1 FSM Low-Power Design Flow	32
	3.1.1 Principal Components	33
	3.1.2 Evaluation of Fitness	33
	3.1.3 Stop Criteria	33
	3.1.4 GA Operators	34
	3.1.5 Crossover and Mutation	34

3.1.6 GA for Low-Power FSM 36
3.2 Low-Power FSM Genome 36
3.2.1 FSM Gene Formation 36
3.2.2 FSM Chromosome Formation 37
3.2.3 Creating Population 39
3.3 Design Flow from an Input-Output Perspective 39
3.3.1 Inputs of the Optimization Flow 41
3.3.2 Preparing the Benchmarks for GA 42
3.3.3 Outputs of the Optimization Flow 44
3.3.4 Power Estimation and Fitness Evaluation 46
3.3.5 Power Estimation using Berkeley SIS Tool 47
3.3.6 Fitness Function and Power Evaluation 48
3.4 Power Model 50
3.5 Partitioning Models 53
3.6 Calculating Complexity of Heuristic and GA Algorithms 55
3.7 Measuring the Improvement of Power Consumption 61

4 RESULTS AND DISCUSSION 62
4.1 Consistent Improvement of the GA Optimization 62
4.2 Overhead of Partitioning 69
4.3 Power Reduction by Partitioning 76
4.4 Comparing Time Complexity of Heuristic and GA Algorithms 81

5 CONCLUSION AND FUTURE WORK 83
5.1 Future Work 84

REFERENCES 85
BIODATA OF STUDENT 94

LIST OF TABLES

Table		Page
1	ISCAS Benchmark Names and Primary Information	17
2	Selected Benchmarks from MCNC [63]	41
3	FSM Configuration Parameters	42
4	Description of Time Complexity Factors	81

LIST OF FIGURES

Figure		Page
1	Basic Configuration of a CMOS Inverter [24][25]	7
2	Charge Phase and Capacitive Load Current [24][25]	8
3	Discharge Phase and Capacitive Load Current [24][25]	8
4	Sources of Parasitic Capacitances in CMOS [24][25]	9
5	Short-Circuit Current [24][25]	10
6	Power Dissipation in CMOS Circuits	12
7	General Digital Circuit [22]	13
8	Karnaugh Logic Minimization Example [22]	14
9	Resulting NOR Gate Implementation[22]	14
10	State Transition Diagram [22]	15
11	State Transition Table [22]	15
12	Resulted Circuit for Parity Checker Example [22]	15
13	General FSM (Sequential Logic) Architecture	16
14	Levels of a Design Hierarchy of Digital Circuits	18
15	Fan-in Configuration Proposed in [32]	20
16	State Change and Self-Loop Determination	21
17	Decomposition Method in [10]	22
18	Kernel Extraction-Based Method [11]	23
19	Synchronous/Asynchronous Configuration [43]	24
20	GA and Partitioning Method [12]	25
21	HLFF, a Custom Designed FF for Low Power [22, 23]	25
22	Algorithm used for State Assignment	27

23	Average Improvements by Years, Works with Encoding Reference	28
24	Average Power Improvement by Years, Works without Encoding Reference	28
25	Cost Function by [29]	29
26	Tools that have been used for Low Power FSM Research	30
27	FSM Low-Power Factors	32
28	Genetic Algorithm General Flow	33
29	Crossover Parents Selection	34
30	Child Formation after Crossover	35
31	Mutated Child	35
32	Gene Formation for Low-Power Design	37
33	A gene for Each State	38
34	Chromosome Formation	38
35	Creating Population	39
36	Method Entire Flow	40
37	“mc.kiss” in form of STT	41
38	Database Preparation	43
39	Outputs of Optimization	44
40	Output Phase	45
41	In-Out Perspective Summary	46
42	GA and Power Estimation Steps	47
43	Berkeley SIS Power Estimation Flow	48
44	Fitness Evaluation Flow	49
45	Power Estimation of Fitness	50
46	Power Model for Gated Power	51

47	Transitional Power	52
48	Implementation Architecture	52
49	Multi-Partition Steady-State Power	54
50	Multi-Partition Transition Power	55
51	Main Loops of Power Optimization Algorithm [34]	56
52	A1 Algorithm Loops	57
53	Loops of A2 Algorithm	57
54	A3 Algorithm Loops	57
55	A4 Algorithm Loops	58
56	Constants and Variables used in Complexity Calculation	58
57	Main Steps of GA and their Sub-Routines	59
58	Create Population Loops	59
59	Complexity of GA Calculate Fitness Function	60
60	Crossover Mutation Complexity	60
61	Replace New Generation Complexity	60
62	Bbara Power Evolution in GA Process	62
63	Bbara Improvement among Generations	63
64	Dk512 Power Evolution in GA Process	63
65	Improvement among Generations	64
66	Donfile Power Evolution in GA Process	64
67	Improvement among Generations	65
68	Modulo12 Power Evolution in GA Process	65
69	Improvement among Generations	66
70	S1 Power Evolution in GA Process	66
71	Improvement among Generations	67

72	S208 Power Evolution in GA Process	67
73	Improvement among Generations	68
74	S386 Power Evolution in GA Process	68
75	Improvement among Generations	69
76	S386 Partitioning Overhead	70
77	S1 Partitioning Overhead	71
78	Modulo12 Partitioning Overhead	71
79	Donfile Partitioning Overhead	72
80	Dk512 Partitioning Overhead	72
81	Bbara Partitioning Overhead	73
82	Bbara Overhead Power Comparison	74
83	Dk512 Overhead Power Comparison	74
84	Donfile Overhead Power Comparison	74
85	Modulo12 Overhead Power Comparison	75
86	S1 Overhead Power Comparison	75
87	S386 Overhead Power Comparison	75
88	Bbara Power Consumption with Different Numbers of Partitions	76
89	Dk512 Power Consumption with Different Numbers of Partitions	77
90	Donfile Power Consumption for Different Numbers of Partitions	77
91	Modulo12 Power Consumption with Different Numbers of Partitions	78
92	S1 Power Consumption with Different Numbers of Partitions	78
93	S386 Power Consumption with Different Numbers of Partitions	79
94	Combinational Power Optimization Convergence	80
95	Comparison of Time Complexities	82

LIST OF ABBREVIATIONS

FSM	Finite State Machine
CMOS	Complementary metal–oxide–semiconductor
STT	State Transition Table
STG	State Transition Graph
KISS	Keep Internal States Simple
IC	Integrated Circuit
RTL	Register Transfer Level
GA	Genetic Algorithm
HDL	Hardware Description Language
VHDL	VHSIC Hardware Description Language
PLA	Programmable Logic Array
FPGA	Field Programmable Gate Array
ASIC	Application-specific Integrated Circuit
TTL	Transistor-Transistor Logic

CHAPTER 1

INTRODUCTION

1.1 Background of the Study

At present, computer and communication system architecture receives substantial attention due to the demands of longevity for portable battery powered devices, such as cameras, smartphones, and various electronic gadgets. Designers and engineers are driven to create circuits, which consume less power. Most of these devices use digital controllers implemented as finite-state machines (FSMs), the consumption of FSMs is crucial to the application [1][2][3][4]. Numerous studies concerned with the power consumption of electronic devices mainly focus on designing low-power FSMs. Accordingly, FSMs are at the heart of every controlled digital device. The size of an FSM can vary from a few gates to thousands of gates that form a processor or complex control logic that handles the control process of a broad application. FSMs often control a simple transmission or manage different components of an extensive system on chip design.[1]-[3], [5], [6].

Therefore, FSMs are synchronous digital electronic circuits that often use Complementary Metal-Oxide-Semiconductor (CMOS) integrated circuit (IC) technology, since this technology has low power consumption characteristics. Furthermore, to realise a low-power circuit design, many different techniques can also be applied to the abstract layers whereby High-level low-power design techniques are extensively employed [7], [8]. Since lower level power estimation and power reduction can be time-consuming, it is desirable to apply the power reduction to the higher levels of abstraction such as RTL level, architecture level and system level. A description of an FSM is a high-level abstraction of the sequential circuit. The best power aware solutions for FSMs can consider the impact of micro-architectural power design choices [9]. Switching activity is an essential power factor determined in high-levels of the abstraction of an FSM [9], [10]. The state encoding, and state assignment are the features which impact the switching activity of an FSM. By controlling these two factors circuit switching activity is manageable [11], [12]. As an example, one-hot encoding used for increasing the reliability of the FSMs [13]. State assignment is a useful and effective method for reducing the power consumption of FSMs. State assignment reduces the power by inferring less toggling switching activity; it also improves the logic optimization, which is determined by the number of literals [14]. State assignment algorithms can vary from standard heuristic to genetic algorithms. Among these methods and algorithms, the genetic algorithm (GA)-based optimization showed higher performance compared with those of the heuristic and other algorithms, such as Kernighan–Lin [15].

Partitioning is another effective method for reducing the power consumption of FSMs. Partitioning retains two or more separate FSMs with identical functionalities as if the sub-FSMs can be combined to form a single-partition FSM. Partitioning methods split

the original FSM` into several smaller sub-FSMs and activate them one at a time. Many previous works employ partitioning and it is still applied to low-power FSM designs. Partitioning is applied in three ways: clock-gating, power-gating and input isolation [7], [12], [16]–[17]. Implementation of FSM partitioning method depends on the device and often differs for a Field Programmable Gate Array (FPGA) compared to an Application Specific Integrated Circuit (ASIC). Any implementation must be able to control the activity of the sub-FSMs. Activation and deactivation of each sub-FSM can be achieved using different methods such as clock-gating, power-gating and input deactivation [15], [18], [19] and [20].

Clock gating allows the deactivation of parts in an FSM by considering the output and the current logic state. Clock gating happens when a driver enables or disables the clock signal for a corresponding active or idle circuit. Clock gating reduces power consumption since the idle circuit is not changing state but consumes some power to hold on to the logic states from when it was active. Functional operation information such as the self-transition probability of each state contributes to the implementation of a clock-gating technique for low-power FSMs.

Input deactivation is a method to gate the inputs of the combinational part of an FSM or a sub-FSM, which prevents the toggling of the combinational logic to contribute to the next state or output logic. Input deactivation is an effective low-power implementation technique. The input deactivation method is more advantageous in terms of time, area and complexity compared to other gating implementation methods.

Power-gating however, can completely shut-down inactive sub-FSMs, and these combinational logic parts then cease to function losing the record of their previous logic states making them slow to reactivate. This technique has a significant advantage to reduce power leakage, but further discussion about the impact of power-gating on leakage power is out of the scope of this thesis [21].

The three mentioned procedures: state assignment, partitioning and Genetic Algorithm (GA) are the foundation of low-power design technique of low-power FSM design. The GA will search the state assignment and state encoding along with the possible partitions for any given FSM. The GA program employs any estimator tool whether academic or commercial to find the best possible low-power solution of the partitioned FSM.

To estimate the power consumption of the partitioned design, first, the power consumption of each partition of the FSM will be calculated using estimation tool. Then, an initiative developed power model based on selected low-power implementation will calculate the final power consumption of the partitioned FSM.

1.2 The Significance of the Genetic Algorithm (GA) based Power Optimization

GA-based optimization methods are evolutionary processes that gradually improve the optimization process. This method is inspired by natural evolutionary processes in the same manner that organisms are evaluated based on their present state of existence. The GA method mimics the biological evolution process by using operations like the natural genetic process models of selection, crossover, and mutation to impact on the generations. Using the GA method, the problem parameters map into arrays of data that will behave similarly to biological chromosomes. The GA code or program implements the following concepts by mimicking natural evolution:

- Gene: the principal factors of which impact our optimization subject (power) should be determined and formed inside the GA database
- Chromosome: after formation of the FSM genes, a chromosome should form. The chromosome contains information of an FSM; therefore, it will be a standalone unit of GA process that can estimate the power of an FSM. Each chromosome includes a combination of possible state assignment, state encoding, and partitioning of a given FSM.
- Population: is a sequence of chromosomes in which performs the power GA fitness assessment. GA operations apply to the population; hence, the best solutions are amongst the population members.

The considerable advantage of GA is that multiple parameters can be simultaneously considered to optimise the factor of interest. For example, the crucial factors that impact low-power FSM design (e.g., encoding and state assignment) are concurrently considered along with the partitioning parameter.

1.3 The Scope of the Study

This research is conducted to create a new framework for designing low power FSM's. This framework performs bi-partitioning and multi-partitioning along with encoding optimization to reduce the power consumption of FSMs. The experiment selects a set of MCNC benchmarks designed and employed in the research of logic and power optimization.

Due to the unavailability of commercial libraries for experiments, academic logic optimization and a power estimator tool are used to estimate power. However, if commercially created libraries were available, then the framework can use a commercial synthesis tool for power estimations.

Due to the stochastic nature of FSMs, it is nearly impossible to find a generalised power estimation rule. Therefore, each benchmark test is evaluated unilaterally to determine the improvement gained by that method. Thus, any estimation of the benchmark performance by considering the result of other FSMs is impossible.

1.4 Problem Statement

Nowadays high complex control-dominated systems have emerged with high-performance features. Therefore, the power consumption of such systems is significant. Finite state machines are used to implement such high-performance systems. Extensive design work is undergoing on FSM's to optimise the power control through state encoding and bi-partitioning. Such implementations use heuristic or GA algorithm to achieve the best solutions.

Because of the complexity of the system processes and unavailability of a proper framework, it is crucial to creating a framework that prepares for and enables changes to the process optimization parameters. For example, changing the number of partitions from two to any desired number further reduces the power consumption. In this work a new framework is proposed for power optimization of finite state machines, using the genetic algorithm and multi-partitioning. Experiments using this framework measure the effects of multi-partitioned designs on the total power consumption of the finite state machines. Also, our proposed framework can perform the conventional bi-partitioning method.

1.5 Objectives

The primary goal of this study is to create a new framework to perform power optimization for FSM design employing a multi-partitioning GA approach. The main aim of multi-partitioning GA optimization approach is to achieve considerably low power consumption for FSMs by reducing the size of their active combinational component. Hence the following objectives of this study are presented as follows:

- To create a new framework that can perform the preparation of the design conducts multi-partitioning, or bi-partitioning, or both, to the given FSM using GA optimization algorithm.
- To modify and extend the conventional bi-partitioning power model to multi-partitioned FSM power consumption.
- To develop the GA algorithm used by the framework to read an FSM in high-level representation and output the optimised FSM with optimised encoding and partitioning information in Verilog HDL format.

1.6 Thesis Organization

Chapter two provides the literature review with necessary background discussions on CMOS power consumption, FSM power consumption, different techniques and methods to reduce and optimise the power consumption of FSMs and give a review of various algorithms and tools used for low-power FDM design.

Chapter three provides the methodology of this study by explaining the modifications of the GA method, process of forming the framework of the experiments and tools, and detailed process of implementation.

Chapter four discusses the results of the experiments with the modified GA optimization algorithm.

Chapter five concludes this study and presents suggestions and directions for future research on the current topic.

REFERENCES

[1] P. Choudhury and S. N. Pradhan, "Power Aware Synthesis of Power Gated FSM," in *Proceedings of Global Engineering, Science and Technology Conference*, 2013, no. April, pp. 1–8.

[2] H. Mittal, D. Chandra, and A. Tiwari, "Designz of Low Power FSM Using Verilog in VLSI," pp. 377–386, 2013.

[3] S. N. Pradhan, D. Nath, P. Choudhury, and A. Nag, "Within-clock power gating architecture implimentation to reduce leakage," in *2012 5th International Conference on Computers and Devices for Communication (CODEC)*, 2012, vol. 3, pp. 1–4.

[4] M. Arora, *The Art of Hardware Architecture*. New York, NY: Springer New York, 2012.

[5] D. L. Oliveira, T. Curtinhas, D. Bompean, L. S. Ferreira, and L. Romano, "Synthesis of synchronous digital systems operating in double-edge of clock," *2012 IEEE 3rd Lat. Am. Symp. Circuits Syst.*, pp. 1–4, Feb. 2012.

[6] P. Choudhury and S. N. Pradhan, "Power Modeling of Power Gated FSM and Its Low Power Realization by Simultaneous Partitioning," pp. 19–29, 2012.

[7] N. Agarwal and N. Dimopoulos, "FSMD partitioning for low power using simulated annealing," *2008 IEEE Int. Symp. Circuits Syst.*, pp. 1244–1247, May 2008.

[8] L. Li and K. Choi, "Novel RT level methodology for low power by using wasting toggle rate based clock gating," *2009 Int. SoC Des. Conf.*, pp. 484–487, 2009.

[9] D. Hutchison and J. C. Mitchell, *Integrated Circuit and System Design Power and Timing Modeling* ,. 1973.

[10] L. Li, K. Choi, S. Park, and M. Chung, "Selective clock gating by using wasting toggle rate," *2009 IEEE Int. Conf. Electro/Information Technol.*, pp. 399–404, Jun. 2009.

[11] T. T. Hwang and a. C. H. Wu, "State assignment for power and area minimization," *Proc. 1994 IEEE Int. Conf. Comput. Des. VLSI Comput. Process.*, pp. 250–254, 1994.

[12] S. Chaudhury, K. T. Sistla, and S. Chattopadhyay, "Genetic algorithm-based FSM synthesis with area-power trade-offs," *Integr. VLSI J.*, vol. 42, no. 3, pp. 376–384, Jun. 2009.

[13] M. Cassel and F. L. Kastensmidt, "Evaluating One-Hot Encoding Finite State Machines for SEU Reliability in SRAM-based FPGAs," *12th IEEE Int. On-Line Test. Symp.*, pp. 139–144, 2006.

[14] L. Benini, S. Member, and G. De Micheli, "State Assignment for Low Power Dissipation," vol. 30, no. 3, pp. 258–268, 1995.

[15] G. Venkataraman, S. M. Reddy, and W. Lafayette, "GALLOP : Genetic Algorithm based Low Power FSM Synthesis by Simultaneous Partitioning and State Assignment," 2003.

[16] J. C. Monteiro and A. L. Oliveira, "FSM decomposition by direct circuit manipulation applied to low power design," *Proc. 2000. Des. Autom. Conf. (IEEE Cat. No.00CH37106)*, pp. 351–358, 2000.

[17] H. Mittal, D. Chandra, S. Kumar, J. S. S. A. Of, and T. Education, "Power Reduction Techniques Applied to FSM Architecture Using Different Technologies," pp. 457–465, 2011.

[18] R. S. S. H. Narayanan and M. P. Desai, "Orthogonal Partitioning and Gated Clock Architecture for Low Power Realization of FSMs Orthogonal Partitioning," pp. 266–270, 2000.

[19] S. Roy, "Partitioning Sequential Circuits for Low Power," in *VLSI Design, 1998. Proceedings., 1998 Eleventh International Conference on*, 1997, pp. 0–5.

[20] L. Mengibar, L. Entrena, and M. Garcı, "Partitioned state encoding for low power in FPGAs," vol. 41, no. 17, pp. 18–19, 2005.

[21] B. F. Tawadros and R. S. Guindi, "State Assignment for Low-Leakage Finite State-Machines," *3rd Int. IEEE-NEWCAS Conf. 2005.*, pp. 1–4, 2005.

[22] C. H. Roth and L. L. Kinney, *Fundamentals of Logic Design*, Seventh ed. 2014.

[23] S. Brown and Z. Vranesic, "of Digital Logic with VHDL Design."

[24] J. D. Alexander and V. Nelson, "Simulation Based Power Estimation For Digital CMOS Technologies," Auburn University, 2008.

[25] M. I. E. Allen, Jonathan, Muhammad S. Elrabaa, Issam S. Abu-Khater, *ADVANCED LOW-POWER DIGITAL CIRCUIT TECHNIQUES THE KLUWER INTERNATIONAL SERIES IN ENGINEERING AND COMPUTER SCIENCE VLSI , COMPUTER ARCHITECTURE AND Consulting Editor*. Springer Science Business Media, LLC, 1997.

[26] a. Rettberg and F. J. Rammig, "A new design partitioning approach for low power high-level synthesis," *Third IEEE Int. Work. Electron. Des. Test Appl.*, p. 6 pp.-148, 2006.

[27] T. Grze, “FINITE STATE MACHINES POWER DISSIPATION,” vol. 9, no. 1, pp. 31–44, 2012.

[28] C. Cao and B. Oelmann, “Low-power state encoding for partitioned FSMs with mixed synchronous/asynchronous state memory,” *Integr. VLSI J.*, vol. 41, no. 1, pp. 123–134, Jan. 2008.

[29] L. Benini, G. De Micheli, and F. Vermeulen, “Finite-state machine partitioning for low power,” *ISCAS ’98. Proc. 1998 IEEE Int. Symp. Circuits Syst. (Cat. No.98CH36187)*, vol. 2, no. c, pp. 5–8, 1998.

[30] P. Chandrakasan, S. Sheng, and R. W. Brodersen, “Lowpower CMOS digital design,” *IEEE Journal of Solid-State Circuits*, vol. 27, no. 4. pp. 473–484, 1992.

[31] K. Roy and S. C. Prasad, “Circuit Activity Based Logic Synthesis for Low Power Reliable Operations,” *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 1, no. 4, pp. 503–513, 1993.

[32] C.-Y. Tsui, M. Pedram, and A. M. Despain, “Technology decomposition and mapping targeting low power dissipation,” *30th ACM/IEEE Des. Autom. Conf.*, pp. 68–73, 1993.

[33] E. Olson and S. M. Kang, “Assignment for Low-Power FSM Synthesis Using Genetic Local Search E. Olson and,” *Cust. Integr. Circuits Conf. 1994., Proc. IEEE 1994*, pp. 140–143, 1994.

[34] S.-H. Chow, Y.-C. Ho, T. Hwang, and C. L. Liu, “Low power realization of finite state machines---a decomposition approach,” *ACM Trans. Des. Autom. Electron. Syst.*, vol. 1, no. 3, pp. 315–340, Jul. 1996.

[35] P. Surti, L. F. Chao, and a. Tyagi, “Low power FSM design using Huffman-style encoding,” *Proc. Eur. Des. Test Conf. ED TC 97*, pp. 521–525, 1997.

[36] J. C. Monteiro and a. L. Oliveria, “Finite state machine decomposition for low power,” *Proc. 1998 Des. Autom. Conf. 35th DAC. (Cat. No.98CH36175)*, pp. 758–763, 1998.

[37] L. Benini, G. De Micheli, A. Lioy, E. Macii, G. Odasso, and M. Poncino, “Computational Kernels and their Application to Sequen tial Power Optimization 1 Introduction 2 De nition of Computational Kernel 3 Heuristics for Kernel Extraction,” pp. 764–769, 1998.

[38] W. Lee and C. Tsui, “Finite state machine partitioning for low power,” *ISCAS’99. Proc. 1999 IEEE Int. Symp. Circuits Syst. VLSI (Cat. No.99CH36349)*, vol. 1, pp. 306–309, 1999.

[39] L. Benini and G. De Micheli, “Synthesis of low-power selectively-clocked systems from high-level specification,” *ACM Trans. Des. Autom. Electron. Syst.*, vol. 5, no. 3, pp. 311–321, Jul. 2000.

[40] L. Benini *et al.*, "Synthesis of Power-Managed Sequential Components Based on Computational Kernel Extraction," vol. 20, no. 9, pp. 1118–1131, 2001.

[41] J. C. Monteiro and a. L. Oliveira, "Implicit FSM decomposition applied to low-power design," *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 10, no. 5, pp. 560–565, Oct. 2002.

[42] J. P. Hayes, "ILP-based optimization of sequential circuits for low power," *Proc. 2003 Int. Symp. Low Power Electron. Des. 2003. ISLPED '03.*, pp. 140–145, 2003.

[43] C. Cao and B. Oelmann, "Mixed synchronous/asynchronous state memory for low power FSM design," *Euromicro Symp. Digit. Syst. Des. 2004. DSD 2004.*, pp. 363–370, 2004.

[44] W. Zhou, H. Huang, C. Sun, and X. Zhou, "Novel on-chip communication data channel architecture used in USB 2.0," *2006 8th Int. Conf. Solid-State Integr. Circuit Technol. Proc.*, pp. 1505–1507, 2006.

[45] Y. Xia, X. Ye, L. Wang, J. Tao, and A. E. A. Almaini, "A novel low power FSM partition approach and its implementation," pp. 5–8, 2004.

[46] M. T. Kumar, S. N. Pradhan, and S. Chattopadhyay, "Power-gated FSM synthesis integrating partitioning and state assignment," *TENCON 2008 - 2008 IEEE Reg. 10 Conf.*, pp. 1–6, Nov. 2008.

[47] S. N. Pradhan, M. T. Kumar, and S. Chattopadhyay, "Integrated Power-Gating and State Assignment for Low Power FSM Synthesis," *2008 IEEE Comput. Soc. Annu. Symp. VLSI*, pp. 269–274, 2008.

[48] S. Nath Pradhan, M. Tilak Kumar, and S. Chattopadhyay, "Low power finite state machine synthesis using power-gating," *Integr. VLSI J.*, vol. 44, no. 3, pp. 175–184, Jun. 2011.

[49] P. Choudhury, A. Nag, D. Nath, and S. N. Pradhan, "CG-in-PG architecture implementation for power reduction in FSMs," *Int. J. Electron. Lett.*, vol. 00, no. 00, pp. 1–6, Mar. 2014.

[50] S. Chaudhury, S. Chattopadhyay, and K. T. S, "Synthesis of Finite State Machines for Low Static and Dynamic Power thsitio prosaiti," pp. 437–440, 2007.

[51] A. Saldanha, H. Savoj, P. R. Stephan, and R. K. B. A. Sangiovanni-vincentelli, "SIS : A System for Sequential Circuit Synthesis," 1992.

[52] S. T. B, "Pyverilog : A Python-Based Hardware Design Processing Toolkit for Verilog HDL," pp. 451–460, 2015.

[53] A. N. Boray S. Deepaksubramanyan, “Analysis of subthreshold leakage reduction in CMOS digital circuits,” *Circuits Syst. 2007. MWSCAS 2007. 50th Midwest Symp.*, 2007.

[54] A. Yildiz, “Speeding Up Logic Locking via Fault Emulation and Dynamic Multiple Fault Injection,” pp. 525–536, 2015.

[55] P. M. Mattheakis and C. P. Sotiriou, “Polynomial Complexity Asynchronous Control Circuit Synthesis of Concurrent Specifications Based on Burst-Mode FSM Decomposition,” *2013 26th Int. Conf. VLSI Des. 2013 12th Int. Conf. Embed. Syst.*, pp. 251–256, Jan. 2013.

[56] Ł. Stefanowicz, M. Adamski, and R. Wisniewski, “Application of an Exact Transversal Hypergraph in Selection of SM-Components,” pp. 250–257, 2013.

[57] A. V. Reddy, “IMPLEMENTATION OF USB TRANSCEIVER MACROCELL INTERFACE,” pp. 289–296, 2012.

[58] S.-H. Kang, H. Yang, L. Schor, I. Bacivarov, S. Ha, and L. Thiele, “Multi-objective mapping optimization via problem decomposition for many-core systems,” *2012 IEEE 10th Symp. Embed. Syst. Real-time Multimed.*, vol. 1, no. c, pp. 28–37, Oct. 2012.

[59] Q. W. Yanyun Taoa, Yuzhen Zhang, “Fuzzy c-mean clustering-based decomposition with GA optimizer for FSM synthesis targeting to low power,” *Eng. Appl. Artif. Intell.*, vol. 68, 2018.

[60] I. Vamshi Veeramachaneni Iowa State University, Ames, I. Akhilesh Tyagi Iowa State University, Ames, and C. Suresh Rajgopal Intel Corp., Santa Clara, “Re-encoding for low power state assignment of FSMs,” in *ISLPED ’95 Proceedings of the 1995 international symposium on Low power design*, 1995.

[61] A. M. D. Chi-Ying Tsui, M. Pedram, “Low-power state assignment targeting two- and multilevel logic implementations,” *IEEE Trans. Comput. Des. Integr. Circuits Syst.*, vol. 17, no. 12, pp. 1281–1291, 1998.

[62] A. A. Pedram, M (Department of Electrical Engineering, University of Southern California, Los Angeles, USA), “Low-power RT-level synthesis techniques: a tutorial,” in *IEE Proceedings - Computers and Digital Techniques*, 2005.

[63] S. Yang, , *Logic synthesis of optimization benchmarks—user guide version 3.0, MCNC Technical Report*. Microelectronics Center of North Carolina, P. O. Box 12889, Research Triangle Park, NC 27709, 1991.

[64] L. Benini, G. De Micheli, E. Macii, G. Odasso, and M. Poncino, “Kernel-based power optimization of RTL components: exact and approximate extraction algorithms,” *Proc. 1999 Des. Autom. Conf. (Cat. No. 99CH36361)*, pp. 247–252.

[65] “Downloaded From WWW.books4career.blogspot.com.”

[66] J. H. Anderson and F. N. Najm, "Low-power programmable routing circuitry for FPGAs," *IEEE/ACM Int. Conf. Comput. Aided Des. 2004. ICCAD-2004.*, pp. 602–609.

[67] X. Zhang, K. Roy, and W. Lafayette, "Design and Synthesis of Low Power Weighted Random Pattern Generator Considering Peak Power Reduction."

[68] V. Zyuban and D. Meltzer, "Clocking Strategies and Scannable Latches for Low Power Applications," vol. 10598, pp. 346–351.

[69] M. Pomarlan, "Evaluating the usefulness of USB for real- time robotics applications."

[70] V.- Dean, C. Tech, T. P. Blessington, B. B. Murthy, G. V Ganesh, and I. Introduction, "IMPLEMENTATION OF SOC ' S AUDIO VIDEO COPROCESSOR."

[71] I. Engineering and H. Xinxiang, "Video Collection System Based on Embedded System and USB Communication Xu Lijun1," pp. 112–114.

[72] S. Brown and J. Rose, "Architecture of FPGAs and CPLDs : A Tutorial."

[73] Y. S. Goud and B. K. Madhavi, "Testing Strategies : Current State of the Art," vol. 13, no. 7, 2013.

[74] D. Ferrari, *Quality , Reliability , Security and Robustness in Heterogeneous Networks*, no. January. 2013.

[75] D. Sheldon, R. Kumart, F. Vahid, and D. Tullsent, "Application-Specific Customization of Parameterized FPGA Soft-Core Processors," pp. 261–268.

[76] J. Ou and K. F. Viktor, "Rapid Energy Estimation of Computations on FPGA based Soft Processors," pp. 0–3.

[77] J. G. Tong, I. D. L. Anderson, and M. A. S. Khalid, "Soft-Core Processors for Embedded Systems," pp. 170–173.

[78] D. Sheldon, R. Kumart, F. Vahid, and D. Tullsent, "Conjoining Soft-Core FPGA Processors," pp. 694–701.

[79] F. Plavec, B. Fort, Z. G. Vranesic, and S. D. Brown, "Experiences with Soft-Core Processor Design," *19th IEEE Int. Parallel Distrib. Process. Symp.*, p. 167b–167b.

[80] S. H. Masoumi, F. Zaman, B. Rokhani, and S. A. R. Al-haddad, "New Tool for Converting High-Level Representations of Finite State Machines to Verilog HDL," pp. 1–6.

[81] M. C. Hansen, H. Yalcin, and J. P. Hayes, "Unveiling the ISCAS-85 benchmarks: a case study in reverse engineering," *IEEE Des. Test Comput.*, vol. 16, no. 3, pp. 72–80, 1999.

[82] D. Mills and C. Cummings, "RTL Coding Styles That Yield Simulation and Synthesis Mismatches," in *SNUG (Synopsys Users Group) 1999 ...*, 1999, pp. 1–15.

[83] C. H. Intel, L. Microsoft, and N. E. C. Philips, "Universal Serial Bus Specification," 2000.

[84] A. Note, "Application Note A Minimal PowerPC [a] Boot Sequence for Executing Compiled C Programs Part II PowerPC Processor Initialization," 2000.

[85] R. B. Brown, S. Member, R. J. Lomax, L. S. Member, G. Carichner, and A. J. Drake, "A Microprocessor Design Project in an Introductory VLSI Course," vol. 43, no. 3, pp. 353–361, 2000.

[86] L. Benini, a. Bogliolo, and G. De Micheli, "A survey of design techniques for system-level dynamic power management," *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 8, no. 3, pp. 299–316, Jun. 2000.

[87] M. A. Perkowski, "Symbolic two-dimensional minimization of strongly unspeci ® ed ® nite state machines," vol. 47, 2001.

[88] M. Pedram, "Memory bus encoding for low power: a tutorial," *Proc. IEEE 2001. 2nd Int. Symp. Qual. Electron. Des.*, pp. 199–204, 2001.

[89] C. E. Cummings and S. Design, "The Fundamentals of Efficient Synthesizable Finite State Machine Design using NC-Verilog and BuildGates September 16-18 , 2002 San Jose , California," 2002.

[90] V. Saxena, F. N. Najm, and I. N. Hajj, "Estimation of state line statistics in sequential circuits," *ACM Trans. Des. Autom. Electron. Syst.*, vol. 7, no. 3, pp. 455–473, Jul. 2002.

[91] E. Fomina, A. Keevallik, M. Kruus, and A. Sudnitson, "for Register-Transfer Level Power Management," 2002.

[92] E. Fomina, A. Keevallik, A. Sudnitson, I. I. Fsm, and D. E. B. Asics, "Low Power Synthesis Based on Information Theoretic Measures," vol. 2, no. Miel, pp. 12–15, 2002.

[93] K. Roy, S. Mukhopadhyay, and S. Member, "Leakage Current Mechanisms and Leakage Reduction Techniques in Deep-Submicrometer CMOS Circuits," vol. 91, no. 2, 2003.

[94] J. H. Anderson and F. N. Najm, "IEEE 2004 CUSTOM INTEGRATED CIRCUITS CONFERENCE A Novel Low-Power FPGA Routing Switch," pp. 719–722, 2004.

[95] a. Raghunathan, G. Lakshminarayana, and N. K. Jha, "Input space adaptive design: a high-level methodology for optimizing energy and performance," *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 12, no. 6, pp. 590–602, Jun. 2004.

[96] A. Sudnitson, "Register Transfer Low Power Design Based on Controller Decomposition," vol. 2, no. MlEL, pp. 735–738, 2004.

[97] A. T. Abdel-Hamid, M. Zaki, and S. Tahar, "A tool converting finite state machine to VHDL," *Can. Conf. Electr. Comput. Eng.*, vol. 4, pp. 1907–1910, 2004.

[98] W. Kim, "Dynamic voltage scaling algorithm for fixed-priority real-time systems using work-demand analysis," *Proc. 2003 Int. Symp. Low Power Electron. Des. 2003. ISLPED '03.*, pp. 396–401, 2003.

[99] J. Cong, "Low-power high-level synthesis for FPGA architectures," *Proc. 2003 Int. Symp. Low Power Electron. Des. 2003. ISLPED '03.*, pp. 134–139, 2003.

[100] M. P. Desai, H. Narayanan, and S. B. Patkar, "The realization of ÿnite state machines by decomposition and the principal lattice of partitions of a submodular function," vol. 131, pp. 299–310, 2003.

[101] A. Bukowiec and L. Gomes, "Partitioning of Mealy Finite State Machines," 2003.

[102] I. Ahmad, "HARD: A hypercube embedding algorithm for state assignment of finite state machines," *Comput. Electr. Eng.*, vol. 29, no. 2, pp. 327–356, Mar. 2003.

[103] C. E. Cummings, "Synthesizable Finite State Machine Design Techniques Using the New SystemVerilog 3 . 0 Enhancements," 2003.

[104] B. Xiaoping, "Study on System Verification of USB2.0 Interface Protocol Control Chip Hardware Design," *2007 Int. Conf. Electr. Eng.*, pp. 1–5, Apr. 2007.

[105] A. Corporation, "White Paper Stratix III FPGAs vs . Xilinx Virtex-5 Devices : Architecture and Performance Comparison," no. October, pp. 1–16, 2007.

[106] H. Quinn, K. Morgan, P. Graham, J. Krone, and M. Caffrey, "A review of Xilinx FPGA architectural reliability concerns from Virtex to Virtex-5," *2007 9th Eur. Conf. Radiat. Its Eff. Components Syst.*, pp. 1–8, Sep. 2007.

[107] H. D. Languages, D. Compiler, R. T. L. Synthesis, D. C. Gui, D. Vision, and S. D. Compiler, “Design Compiler Tutorial Introduction Initialization,” pp. 1–27, 2007.