Interconnect Modeling and Optimization in Deep Sub-Micron Technologies

BARKER
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
JUL 3 1 2002
LIBRARIES

by

Paul Peter P. Sotiriadis

Diploma in ECE,
National Technical University of Athens, Greece (1994)
MSEE, Stanford University (1996)

Submitted to the Department of Electrical Engineering and Computer Science in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 2002

© Massachusetts Institute of Technology 2002

Author . . . . . .
Department of Electrical Engineering and Computer Science
May 24, 2002

Certified by . . . . . .
Anantha P. Chandrakasan
Associate Professor
Thesis Supervisor

Accepted by . . . . . .
Arthur C. Smith
Chairman, Department Committee on Graduate Students

Interconnect Modeling and Optimization in Deep Sub-Micron Technologies

by

Paul Peter P. Sotiriadis

Submitted to the Department of Electrical Engineering and Computer Science
on May 24, 2002, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

Interconnect will be a major bottleneck for deep sub-micron technologies in the years to come. This dissertation addresses the communication aspect from a power consumption and transmission speed perspective.

A model for the energy consumption associated with data transmission through deep sub-micron technology buses is derived. The capacitive and inductive coupling between the bus lines as well as the distributed nature of the wires is taken into account. The model is used to estimate the power consumption of the bus as a function of the *Transition Activity Matrix*, a quantity generalizing the transition activity factors of the individual lines.

An information theoretic framework has been developed to study the relation between speed (number of operations per time unit) and energy consumption per operation in the case of synchronous digital systems. The theory provides us with the fundamental minimum energy per input information bit that is required to process or communicate information at a certain rate. The minimum energy is a function of the information rate, and it is, in theory, asymptotically achievable using coding. This energy-information theory combined with the bus energy model result in the derivation of the fundamental performance limits of coding for low power in deep sub-micron buses.

Although linear, block linear and differential coding schemes are favorable candidates for error correction, it is shown that they only increase power consumption in buses. Their resulting power consumption is related to structural properties of their generator matrices. In some cases the power is calculated exactly and in other cases bounds are derived. Both provide intuition about how to re-structure a given linear (block linear, etc.) code so that the energy is minimized within the set of all equivalent codes.

A large class of nonlinear coding schemes is examined that leads to significant power reduction. This class contains all encoding schemes that have the form of connected Finite State Machines. The deep sub-micron bus energy model is used to evaluate their power reduction properties. Mathematical analysis of this class of coding schemes has led to the derivation of two coding optimization algorithms. Both algorithms derive efficient coding schemes taking into account statistical properties of the data and the particular structure of the bus. This coding design approach is generally applicable to any discrete channel with transition costs.

For power reduction, a charge recycling technique appropriate for deep sub-micron buses is developed. A detailed mathematical analysis provides the theoretical limits of power reduction. It is shown that for large buses power can be reduced by a factor of two. An

efficient modular circuit implementation is presented that demonstrates the practicality of the technique and its significant net power reduction.

Coding for speed on the bus is introduced. This novel idea is based on the fact that coupling between the lines in a deep sub-micron bus implies that different transitions require different amounts of time to complete. By allowing only "fast" transitions to take place, we can increase the clock frequency of the bus. The combinatorial capacity of such a constrained bus is the theoretical maximum rate that data can be transmitted. While this rate is less than that of the original bus, we can clock the constrained bus faster. It turns out that the net data transmission speed can be significantly higher than that of the original bus, in some cases 120% higher. A methodology to estimate the amount of time each transition requires is introduced. The results are compared to HSPICE and MATLAB simulations and are shown to be conservative. Using these estimates the transitions are classified according to how fast they are completed. The fundamental theoretical limits of *Coding For Speed* are derived. The increase of the effective speed of the bus is estimated as a function of the number of lines and the coupling between them.

Finally, a class of practical coding schemes for speed, termed *Differential* $RLL(1,\infty)$ schemes, is presented. They have low complexity and they can increase the data transmission rate significantly.

Thesis Supervisor: Anantha P. Chandrakasan
Title: Associate Professor

Acknowledgments

I am most grateful to my advisor, Professor Anantha Chandrakasan, for his continuous encouragement and guidance. I consider myself extremely lucky for having worked with such a prominent leader in the field of circuits and systems. His teaching, leadership and guidance have been instrumental in my academic and professional development.

I am also very grateful to my collaborator and thesis reader Professor Vahid Tarokh, a great mathematician and a pioneer in the field of communications. Prof. Tarokh has been a tremendous source of inspiration and motivation. The collaboration with him has been essential in the understanding of the fundamental relations between energy and information.

I would like to express my deep gratitude and highest respect to my thesis reader Professor Sanjoy Mitter. Throughout my years at MIT, Professor Mitter has been an invaluable source of support and guidance as well as an inspiring teacher. I greatly admire him for his landmark and visionary work in mathematical systems and control theory.

For their financial support I would like to thank: The MARCO Focus Research Center on Interconnect supported by MARCO and DARPA; Alexander S. Onassis Public Benefit Foundation, the Greek Section of Scholarships and Research; Evgenides Foundation, Athens Greece, Bodossakis Foundation, Athens Greece; Gerondelis Foundation, MA, USA.

I greatly appreciate the many friendly discussions I have had with Prof. John Tsitsiklis throughout my years at MIT. His insightful suggestions have always been invaluable.

I am extremely grateful to Prof. Yannis Tsividis for his continuous encouragement and support and for being a superb teacher, mentor, and collaborator since my undergraduate years. His academic excellence, creativity, deep engineering intuition and character integrity have made him my preeminent role model.

I also thank my professors at the National Technical University of Athens, Greece, for the very high level of education that they provided. Obtaining a strong background early on has been an essential component of my subsequent carrier. I would also like to thank Prof. Protonotarios for his continuous support.

I would like to thank my wonderful Intel friends and collaborators Drs. Dan Bailey and Olivier Franza for sharing with me their professional experience and for providing constant encouragement.

Many thanks to the entire "digital integrated circuits and systems" group, including current students and graduates, and in particular to: Prof. Wendi Beth Heinzelman, Alice Wang, Manish Bhardwaj, Benton (Ben) Highsmith Calhoun, SeongHwan Cho, Raul Blazquez-Fernandez, Travis Furrer, Drs. Jim Goodman and Vadim Gutnik, Frank Honore, Dr. James Kao, Theodoros Konstantakopoulos, Rex Min, Dr. Jose Oscar Mur-Miran, Chee We Ng, Eugene Shih, Drs. Amit Sinha and Duke Xanthopoulos. The interaction with these brilliant people has been a most pleasant and fruitful experience. Their friendship is greatly appreciated. Alice, Ben and Theodoros have been wonderful collaborators in several research projects. Special thanks to Ben and Frank who put the extra effort to proofread my thesis. I also wish to thank Margaret Flaherty, our administrative assistant, for her skillful support.

Many thanks to Navid Sabbaghi, Louay Bazzi and Drs. Sekhar Tatikonda and Constantinos Boussios, for their friendship and the many intellectually stimulating discussions.

I am extremely grateful to my close friends Katerina Ioakeimidi, Toula Seremetis, Dr. Kali Stasi, entrepreneur John C. Dimitrakakis, and professors Peter Danias and Costis Maglaras. Throughout my years at Stanford and MIT they have enriched my life with their companion and friendship. They have always been my brothers and sisters and their support has been crucial for my achievements.

Most importantly, I am deeply indebted to my parents for everything I am and all I have accomplished. My mother Danae, the sweetest person I know, and my father Panagiotis, a most distinguished painter and architect, have always been my best friends, providing continuous encouragement, love, guidance and support. They were the first to teach me the beauty and value of life, art and science. They have always been and will be my inspiration and motivation for every high and noble achievement.

Paul Peter P. Sotiriadis

MIT, May 2002

To my most wonderful parents,

Danae and Panagiotis Sotiriadis

with all my love and respect

ΚΡΕΣΣΟΝΕΣ ΕΙΣΙ ΑΙ ΤΩΝ ΠΕΠΑΙΔΕΥΜΕΝΩΝ ΕΛΠΙΔΕΣ Η Ο ΤΩΝ ΑΜΑΘΩΝ ΠΛΟΥΤΟΣ

ΔΗΜΟΚΡΙΤΟΣ

"The visions of educated people are more important than the wealth of the uneducated"

Democritus

Contents

Chapter 3

Power Reduction Using Coding, The Fundamental Information Theoretic Limits

Chapter 4

Linear, Block Linear and Convolutional Coding For Power Reduction

Chapter 5

Transition Pattern Coding (TPC) For Power reduction

Chapter 6

A Charge Recycling Technique For Power Reduction In Deep Sub-Micron Buses

Chapter 7

Coding For Increasing The Throughput and Speed of Deep Sub-Micron Buses

Chapter 8

Coding For Speed: The Fundamental Limits and The Differential RLL(1,∞) Scheme

Chapter 9

Conclussions and Future Directions

Chapter 1

Introduction

1 Interconnect: A Major Bottleneck

We have experienced a tremendous advance in computer performance during the last three decades. Performance has doubled every eighteen months as G. Moore's law predicted [1].

To continue progressing at the same rate we will need to resolve some major technological problems. In the years to come transistors are expected to become even smaller and faster. The difficulty will be to inter-connect them efficiently. This is the opposite situation from a few years ago, when the performance of the circuit was completely determined by the quality of the transistors.

The set of wires in the circuit responsible for *communication*, *clock distribution* and *power distribution* is called the *interconnect.* Interconnect will be a major source of design problems for the next two decades [2, 3]. Increased speed of microprocessor's core implies that the communication rate of data and instructions must be increased as well. This is not trivial, especially in the *global* interconnect networks that are responsible for carrying information between distant locations on the circuit. The properties of the wires do not scale with technology in a favorable way [4]. Even in current technologies, signals require several clock cycles to travel across advanced microprocessor chips.

The speed of light becomes a limiting factor as the clock frequency and the size of chips are increasing. Clock distribution is an issue for the same reasons. Higher speeds mean less interconnect budget [5]. Finally, larger circuits and faster clocks set more demanding specifications for power distribution.

The interconnect problem is so severe and so general that Semiconductor Industry Association [6] has devoted a lot of resources to address it. To overcome the interconnect bottleneck, several technological advances will be needed.

This thesis concentrates on the communication aspect of interconnect. On-chip and inter-chip communications are realized by circuit devices called buses. A bus consists of parallel, aligned, similar wires that in most cases are on the same metal layer. Drivers and receivers are connected at the two ends of the wires, and in some cases there are intermediate repeaters as well. Transmitting data and instructions through long and wide buses requires a lot of energy and many clock cycles [3, 7, 8]. Also, in some widely used classes of circuits [9], the communication power may exceed half of the total power consumption.

In modern deep sub-micron (DSM) technologies, the communication problem has become severe and complex. A reason for this is the increased coupling between the lines of buses [8, 4, 10]. This is because of the smaller distances between the lines as well as the higher aspect ratio (height/width) necessary to maintain a linear resistance of reasonable size. Another complication is the distributed behavior of thin and long lines. Both issues are expected to become even more dominant in the future technologies [8, 5].

It it interesting that neither of these two problems, coupling nor distributed phenomena, occurred in earlier technologies. This means that early approaches to estimate power and delay in buses may not be appropriate for current DSM technologies. Furthermore, techniques invented to reduce power consumption do not perform well in modern technologies.

The present thesis intends to fill this gap by presenting new tools for estimating and analyzing the energy and delay in buses, and by introducing methodologies that can reduce the power consumption, increase the speed and reduce the latency of communication. Both, practical techniques and ultimate achievable performance limits are covered.

2 Contributions and Outline of the Thesis

The thesis addresses two major issues of on-chip and inter-chip parallel communication. Chapters 2 to 6 deal with energy consumption associated with information communication. Chapters 7 and 8 deal with the speed of communication.

Chapter 2 : In Chapter 2 we study the energy consumption associated with data transmission through DSM technology buses. The capacitive and inductive coupling between the bus lines as well as the distributed nature of the wires is taken into account. It is formally proven that under some reasonable assumptions on the timing and the drivers the energy depends only on the admittance matrix of the capacitive part of the bus [11]. The analysis results in an energy model. The model is used to reveal the differences in energy behavior between buses with strong inter-line coupling (modern DSM technologies) and that of buses without coupling (older technologies). Finally, the expected power consumption of the bus is evaluated as a function of the *Transition Activity Matrix*, a quantity generalizing the transition activity factors of the individual lines. The bus energy model is used in Chapters 3-6 to provide a basis for applications of coding and charge recycling techniques.

Chapter 3 : Information processing and communication are the purposes of all computers. It is desirable to perform these operations fast and with energy efficiency. It is also well known that speed and power consumption increase together [12].

In Chapter 3 we study the relation between speed, i.e. number of operations per second, and energy consumption per operation from an information theoretic point of view. We develop a theoretical framework that relates these two quantities in the case of synchronous digital circuits. Given such a circuit, the theory provides us with the fundamentally minimum energy per input information bit that is required to process information at a certain rate. The minimum energy is a function of the processing rate, and it is, in theory, asymptotically achievable using coding. The mathematical framework in Chapter 3 applies directly to combinational circuits. An immediate extension of the theory allows the treatment of finite state machines. Application to "non-circuit" devices is possible when they have discrete states and input, and an energy cost that at every time k depends on a fixed, finite number of previous states [13, 15].

The development of this mathematical framework was motivated by the energy consumption of DSM buses and the question of how much we can reduce it using coding. The energy model of Chapter 2 is used throughout the chapter to motivate the theory and to derive the limits of performance of coding for low power. A reader that would prefer a more practical perspective of the material, specifically for DSM buses, is referred to [14].

Chapter 4 : The theory in Chapter 3 provides us with the tools to calculate the minimum energy required to communicate through buses. It gives us the fundamental limits of the performance of coding schemes and tells us that these limits are asymptotically achievable. The question now is how to approach these limits with practical coding schemes. It is true that linear coding schemes are simply structured and very popular in many communication applications where the objective is the reduction of error probability. Chapter 4 deals with linear, block linear and differential coding schemes in DSM buses from an energy perspective. It is shown that these favorable candidates never reduce the power consumption. In Chapter 4 we relate the resulting power consumption using such coding schemes with the structure of their generator matrices. In some cases the energy is calculated exactly and in other cases we provide bounds. Both provide intuition about how to re-structure a given linear (block linear, etc.) code so that the energy is minimized within the set of all equivalent codes. Although these classes of codes do not reduce power consumption, they may be used for error correction in the bus. In this case we would like to achieve the best error correction performance with the minimum additional power consumption [16]. The results of Chapter 4 provide us with guidelines on how to do this.

Chapter 5 : In contrast to Chapter 4, where linear coding schemes are studied and proven inappropriate for power reduction, in Chapter 5 we examine a large class of nonlinear schemes. This class contains all encoding schemes that have the form of connected Finite State Machines. Their states are the vectors transmitted through the expanded bus, and the data is their input. We term this class *Transition Pattern Coding* (TPC) schemes since the design of the machines is determined by the energies required for the transitions of the bus [17]. This class of schemes does not introduce latency in the system, in the sense that the data vector can be transmitted during the cycle it arrives at the encoder.

Chapter 5 provides a detailed mathematical analysis of the energy consumption of the encoded bus and two optimization algorithms that derive the appropriate coding schemes given the transition energies of the bus. Furthermore, the case of bus partitioning is studied in detail as an approach to reduce the complexity of the encoding and decoding. We note that the TPC schemes and the TPC algorithm can be applied to other discrete channels and Finite State Machine optimizations respectively. The application to DSM buses is a special case. Finally, it is true that most of the schemes in the literature are particular examples of TPC schemes. The advantage of the TPC approach is that the design of the schemes takes into account the particular energy behavior of the bus and statistics of the data [19].

Chapter 6 : Another efficient and practical approach to reduce power consumption in buses is charge recycling. The idea is to use the charge of lines transitioning from V_{dd} to 0 in order to pre-charge lines transitioning from 0 to V_{dd}. Charge recycling can be thought as a special case of adiabatic circuit techniques. In Chapter 6 we discuss a charge recycling scheme that is appropriate for DSM buses [20]. A detailed mathematical analysis provides the theoretical limits of power reduction. It is shown that for large buses the power reduction can be up to 50%. Power is reduced further when charge recycling is combined with Bus Invert coding [14]. Finally, an efficient modular circuit implementation is also discussed that demonstrates the practicality of the technique and its significant net power reduction.

Chapter 7 : A recent work [21] has demonstrated how coding can be used to increase significantly the throughput of DSM buses. This new idea is based on the fact that coupling between the lines in the bus implies that different transitions require different amounts of time to complete. By allowing only "fast" transitions to take place, we can increase the clock frequency of the bus. A bus with restricted transitions is a constrained discrete channel when seen from a communication perspective. The combinatorial capacity of such a channel is the theoretical maximum rate that data can be transmitted. While this rate is less than the number of lines in the bus, we can clock the bus faster. It turns out that the net data transmission speed can be significantly higher than that of the uncoded bus, in some cases 120% higher. Chapter 7 presents a methodology to estimate the amount of time each transition requires. The results are compared to HSPICE and MATLAB simulations and

are proven to be conservative. Using these estimates we classify the transitions according to how fast they are completed. Finally, a simple coding example at the end of the chapter introduces the idea of *Coding For Speed.*

Chapter 8 : In Chapter 8 we present the theoretical limits of *Coding For Speed.* We estimate how much we can increase the throughput of buses as a function of their size and the coupling between their lines. For narrow buses, the limits are calculated numerically. For wide buses the limits are upper and lower bounded using an enumeration technique [22]. Finally, a class of practical coding schemes for speed, termed *Differential* $RLL(1, \infty)$ schemes, is presented in the chapter. They have very low complexity, since their encoder and decoder are combinational circuits, and can increase the data transmission rate significantly.

References

[1] G. Moore, *Progress in Digital Integrated Electronics*, IEDM, 1975.

[2] *International Technology Roadmap For Semiconductors, 2001 Edition ; Interconnect*, http://public.itrs.net/Files/2001ITRS/Home.htm

[3] H. Bakoglu, *Circuits, interconnections, and packaging for VLSI*, Addison-Wesley Pub. Co. Reading, Mass. 1990.

[4] C. Cheng, J. Lilis, S. Lin, N. Chang, *Interconnect Analysis and Synthesis*, Wiley Inter-Science 2000.

[5] S. Hall, G. Hall, J. McCall, *High-Speed Digital System Design*, Willey Interscience Publication, John Willey and Sons, 2000.

[6] *Semiconductor Industry Association*, http://www.semichips.org/pressroom.cfm

[7] V. Agarwal, M. Hrishikesh, S. Keckler, D. Burger, "Clock rate versus IPC: the end of the road for conventional microarchitectures", *Proceedings of the 27th Int. Symp. on Computer Architecture*, 2000, pp. 248-259.

[8] T. Sakurai, "Design Challenges for $0.1\mu m$ and Beyond", *Asia and South Pacific Design Automation Conference 2000, pp. 553-558.*

[9] E. Kush, J. Rabaey, "Low-Energy Embedded FPGA Structures", *IEEE/ACM International Symposium on Low Power Electronics and Design 1998, pp. 155-160.*

[10] J. Rabaey, *Digital Integrated circuits*, Prentice Hall 1996.

[11] P. Sotiriadis, A. Chandrakasan, "A Bus Energy Model for Deep Sub-micron Technology", *To appear in IEEE Transactions on VLSI Systems.*

[12] A. Chandrakasan, R. Brodersen, "Minimizing power consumption in digital CMOS circuits", *Proceedings of the IEEE*, Vol. 83 Issue 4 , April 1995 pp. 498 -523.

[13] P. Sotiriadis, V. Tarokh, A. Chandrakasan, "Energy Reduction in Deep Sub-micron Computation Modules An Information Theoretic Approach", *To appear in IEEE Transactions on Information Theory.*

[14] P. Sotiriadis, A. Chandrakasan, V. Tarokh, "Maximum Achievable Energy Reduction Using Coding with Applications to Deep Sub-Micron Buses", *IEEE International Symposium on Circuits and Systems*, Arizona, May 2002.

[15] P. Sotiriadis, V. Tarokh, A. Chandrakasan, "Information Theoretic Treatise of Energy Reduction in Deep Submicron Computation Modules", *IEEE International Symposium on Information Theory*, Lausanne, Switzerland, June 2002.

[16] P. Sotiriadis, V. Tarokh, A. Chandrakasan, "Linear Coding, Energy Reduction and Error Correction in deep sub-micron Buses", *in preparation.*

[17] P. Sotiriadis, A. Chandrakasan, "Bus Energy Minimization by Transition Pattern Coding (TPC) in Deep Sub-Micron Technologies", *Proceedings of the IEEE International Conference on Computer Aided Design*, San Jose, Nov. 2000.

[18] P. Sotiriadis, A. Chandrakasan, "Low Power Bus Coding Techniques Considering Inter-wire Capacitances", *Proceedings of the IEEE Custom Integrated Circuits Conference*, Orlando, May 2000.

[19] P. Sotiriadis, A. Chandrakasan, "Coding for Energy Reduction in Deep Sub-Micron Technologies, The Transition Pattern Coding Approach", *Submitted to IEEE Transactions on Circuits and Systems*, March 2001.

[20] P. Sotiriadis, T. Konstantakopoulos, A. Chandrakasan, "Analysis and Implementation of Charge Recycling for Deep Sub-micron Buses", *Proceedings of the IEEE/ACM International Symposium on Low Power Electronics and Design*, Long Beach, CA, 2001, pp. 109-114.

[21] P. Sotiriadis, A. Chandrakasan, "Reducing bus delay in sub-micron technology using coding", *IEEE Asia and South Pacific Design Automation Conference*, Yokohama, Japan 2001, pp. 322-327.

[22] P. Sotiriadis, "Increasing the on-chip Communication Speed; Fundamental Limits and Applications", In preparation.

Chapter 2

A Bus Energy Model For Deep Sub-Micron Technologies

1 Introduction

On-chip and inter-chip communication is responsible for a significant amount of power dissipated in modern digital circuits [1],[2]. Buses are the circuit units realizing this communication. Reliable estimation of the power that buses draw from the power supply is important. Peak and average power are crucial parameters of their design as well as the design of the supporting circuits. In addition, a model of their energy requirements is necessary for the efficient evaluation of the digital circuit as a whole and the evaluation of power reduction techniques; for example, charge recycling [3]-[7], coding for low power [8]-[11] and low-swing interconnect [12]-[14].

Until recently a very simple model for buses has been used for theoretical *energy estimation*. The bus lines were replaced by lumped grounded capacitors of equal sizes, and no inter-line coupling was present. This model is inadequate for deep sub-micron (DSM) technology buses. Technology scaling has introduced new elements that are not captured by this "traditional" energy model. In modern deep sub-micron technologies, the bus lines are *distributed* and heavily capacitively and inductive *coupled* to each other [15]-[23].

The coupled dynamics of the lines result in dependencies between the energy drawn from the power supply by the drivers. The coupling due to inter-line parasitics becomes stronger with technology scaling, and in most cases it is more significant than the coupling between individual lines and ground.

This chapter presents a *compact energy model for DSM buses* based on a detailed distributed circuit model of the bus. The model will be used extensively in the following chapters. In particular we will use it to estimate the efficiency of coding and other techniques for power reduction as well as to estimate the fundamental limits of power reduction.

2 Deep Sub-Micron Technology Bus Model

In general, a bus may consist of one or more sets of parallel lines with repeaters between them. Here we examine the simple case of one set of parallel lines driven by CMOS inverters

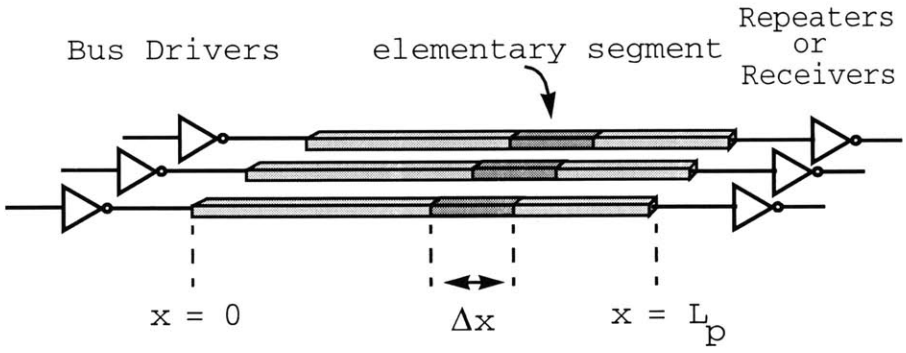

Figure 1: DSM bus

and loaded by the input capacitors of the next stage as in Figure 1. The energy for the general case is the sum of the energies of all stages.

2.1 The Lines

For the DSM bus of Figure 1 we will use the distributed line model of Figure 2. The lines here are assumed distributed, lossy, capacitively and inductively coupled. This is an established

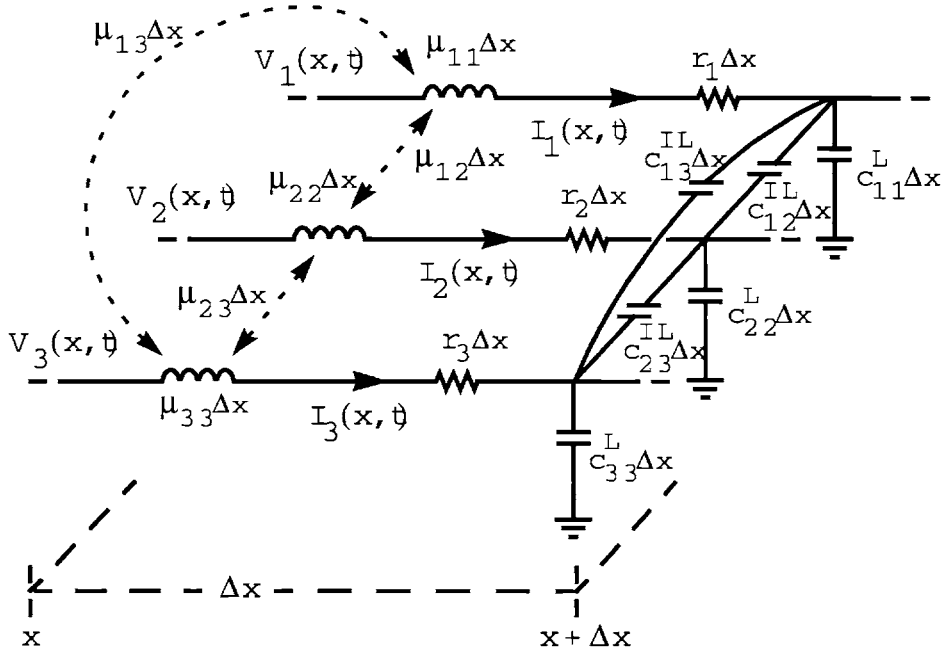

Figure 2: Elementary segments of DSM bus lines

model for DSM bus lines. It has been used in the past for delay estimation as well as signal integrity evaluation [26]-[29].

The lines are laid along the x axis and have physical length L_p. All the major parasitic elements between lines and ground (shielding or substrate) are included. The densities of the different quantities are, $r_i(x)$ for the serial resistance of the i^{th} line, $c_{i,i}(x)$ for the capacitance between the i^{th} line and ground, $c_{i,j}(x)$ for the capacitance between lines i and j. Also, $\mu_{i,i}(x)$ is the density of the self inductance of the i^{th} line and $\mu_{i,j}(x)$ is the density of the mutual inductance between lines i and j. The densities may depend on x. Finally, possible lumped parasitics can be included as limiting cases of distributed ones.

The current $I_i(x,t)$ is running through the i^{th} line at the point $x \in [0, L_p]$ and time $t \geq 0$. Let $I = [I_1, I_2, \cdots, I_n]^T$ be the current vector. Similarly, $V_i(x,t)$ is the voltage of that point with respect to ground and let $V = [V_1, V_2, \cdots, V_n]^T$ be the voltage vector. The line model of Figure 2 satisfies the system of partial differential equations:

$$-\frac{\partial I}{\partial x}(x,t) = C(x)\frac{\partial V}{\partial t}(x,t) \tag{1}$$

and

$$-\frac{\partial V}{\partial x}(x,t) = M(x)\frac{\partial I}{\partial t}(x,t) + R(x)I(x,t) \tag{2}$$

The three $n \times n$ matrices R, M, C correspond to the distributed resistance, inductance and capacitance respectively. They have the following forms:

$$R(x) = diag[r_1(x), r_2(x), \cdots, r_n(x)] \tag{3}$$

(diagonal matrix with elements : $r_1(x), r_2(x), \cdots, r_n(x)$).

$$M(x) = [\mu_{i,j}(x)]_{i,j=1}^{n} \tag{4}$$

and

$$C(x) = C^L(x) + \sum_{i<j} C_{i,j}^{IL}(x) \tag{5}$$

The diagonal matrix

$$C^L(x) = diag[c_{1,1}^L(x), \cdots, c_{n,n}^L(x)] \tag{6}$$

corresponds to the parasitic capacitances between the lines and the ground. The matrices $C_{i,j}^{IL}$, $i < j$ correspond to the inter-line capacitance densities $c_{i,j}^{IL}(x)$. The matrix $C_{i,j}^{IL}(x)$ may have nonzero elements only in the entries (i,i),(i,j),(j,i),(j,j) and it is of the form,

$$C_{i,j}^{IL}(x) = c_{i,j}^{IL}(x) \begin{bmatrix} 0 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & & \vdots & & \vdots & & \vdots \\ 0 & \cdots & 1 & \cdots & -1 & \cdots & 0 \\ \vdots & & \vdots & & \vdots & & \vdots \\ 0 & \cdots & -1 & \cdots & 1 & \cdots & 0 \\ \vdots & & \vdots & & \vdots & & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 0 \end{bmatrix} \tag{7}$$

As defined, $C(x)$ is the distributed capacitance conductance matrix of the lines.

So far, the electrical characterization of the bus lines has been given. More details can be found in the literature, for example in [15, 19, 22].

2.2 Timing Issues

Independently of the particular application, the data is expected to travel through the bus within some time period T. Without loss of generality, we can examine the bus during the time interval $[0, T]$. At $t = 0$ the bus is driven with the new data and at $t = T$ the data is sampled at the other end. It is reasonable to assume that by the time the data is sampled, the voltages along the lines have settled to their final values. In other words, if V_i^f is the data being transmitted through the i^{th} line (V_i^f is 0 or V_{dd}), then after time T we have that,

$$V_i(x, T) = V_i^f \tag{8}$$

for all $x \in [0, L_p]$ and $i = 1, 2, \cdots, n$ (superscript f stands for final). This assumption also implies that when the transition of the bus starts at time $t = 0$, the voltages along the lines correspond to their previous binary values, i.e.

$$V_i(x, 0) = V_i^i \tag{9}$$

for all $x \in [0, L_p]$ and $i = 1, 2, \cdots, n$ (superscript i stands for initial). Relations (9) and (8) are the initial and final conditions of the system of P.D.E.s given by (1) and (2). For convenience we define the vectors of initial and final voltages as, $V^i = [V_1^i, V_2^i, \cdots, V_n^i]$ and $V^f = \left[V_1^f, V_2^f, \cdots, V_n^f\right]$ respectively. Our assumption can be written as

$$V(x, 0) = V^i \tag{10}$$

and

$$V(x, T) = V^f \tag{11}$$

for all $x \in [0, L_p]$.

The dynamics of the bus lines along with their initial and final conditions are given by (1),(2),(10) and (11). We also need boundary conditions for the P.D.E.s. These will be provided by circuit models for the drivers and the loads of the lines.

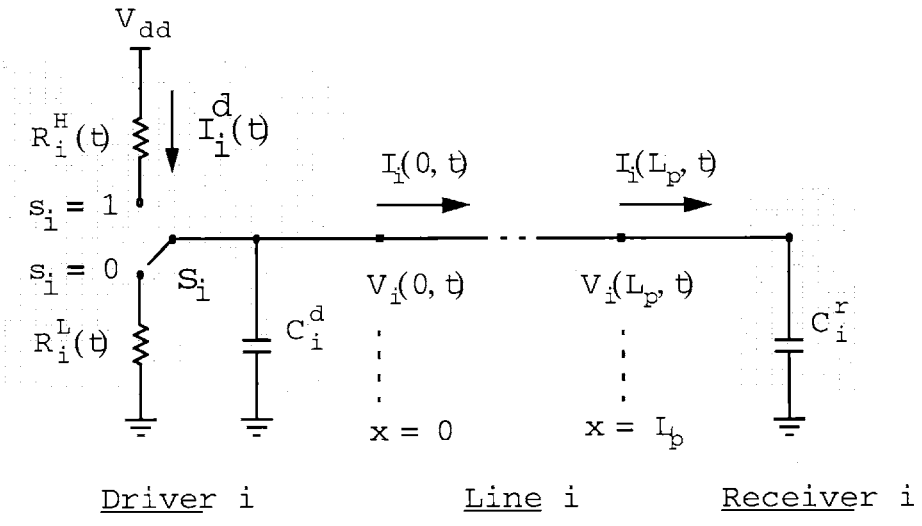

Figure 3: Driver-line-receiver

2.3 Drivers and Loads

In Figure 3 we see the i^{th} line connected to its driver and load. The driver (CMOS inverter) has been modelled as a switch connecting the line either to the power supply or the ground through resistors [25] [1]. The resistors $R_i^H(t)$ and $R_i^L(t)$ corresponding to the PMOS and NMOS of the inverter, are not necessarily the on resistors of the transistors in their linear regions. Their values can be *arbitrary* functions of time as long as the timing assumptions of Section 2.2 remain valid. The resistors simply express the current flow through the MOSFETs. The switch s_i connects to $R_i^H(t)$, $(s_i = 1)$ or $R_i^L(t)$, $(s_i = 0)$ depending on the binary value to be transmitted, i.e. depending on whether it is $V_i^f = V_{dd}$ or $V_i^f = 0$. Therefore it is $V_i^f = s_i V_{dd}$. The capacitance C_i^d is the parasitic capacitance at the output of the i^{th} driver. We define the diagonal matrix,

$$C^d = diag[C_1^d, C_2^d, \cdots, C_n^d] \tag{12}$$

[1] This model does not account for the short-circuit currents of the drivers which have to be evaluated independently using information of the waveforms in the inputs of the drivers.

The load at the end of the line (input of the receiver circuit) is represented by the capacitor C_i^r. We set,

$$C^r = diag[C_1^r, C_2^r, \cdots, C_n^r] \tag{13}$$

In Figure 3, $V_i(0,t)$ and $V_i(L_p,t)$ are the boundary voltages of the line. The currents $I_i(0,t)$ and $I_i(L_p,t)$ are the driving and loading currents of the line. The current drawn from V_{dd} is denoted by $I_i^d(t)$. This current is responsible for the power drawn from V_{dd} by the i^{th} driver. The current is zero when $s_i = 0$. Even more, since $V_i^f/V_{dd} = s_i$ we can write,

$$I_i^d(t) = \frac{V_i^f}{V_{dd}} \left\{ I_i(0,t) + C_i^d \frac{\partial V_i(0,t)}{\partial t} \right\} \tag{14}$$

for all $i = 1, 2, \cdots, n$.

3 Transition Energy

The energy drawn from V_{dd} by the i^{th} driver during the transition period $[0,T]$ is given by

$$E_i = \int_0^T V_{dd} I_i^d(t) dt \tag{15}$$

The time integral is evaluated in Appendix A and the energy E_i is given by the expression,

$$E_i = V_i^f e_i^T C^t (V^f - V^i) \tag{16}$$

The vector e_i has a one in the i^{th} position and zero everywhere else. The matrix C^t, which we call the *total capacitance conductance matrix*, is given by the sum,

$$C^t = C^d + C^r + \int_0^{L_p} C(x) dx \tag{17}$$

with C^d and C^r defined by (12) and (13) respectively and the matrix function $C(x)$ given by (5). Note that the inductance of the lines does not appear in the energy expression. This is because of our timing assumption; that is, the voltages along the lines have settled to their final values by the end of the clock period.

We define $\tilde{c}_{i,j}$, $i \neq j$, to be the total capacitance between lines i and j, ($\tilde{c}_{i,j} = \tilde{c}_{j,i}$), and $\tilde{c}_{i,i}$ to be the total capacitance between line i and ground (including the capacitances of the

driver and the receiver). It is,

$$\tilde{c}_{i,j} = \int_0^{L_p} c_{i,j}^{IL} dx \tag{18}$$

$$\tilde{c}_{i,i} = C_i^d + C_i^r + \int_0^{L_p} c_{i,i}^{L} dx \tag{19}$$

We conclude that $C^t = \left[c_{i,j}^t\right]_{i,j=1}^n$ is the conductance matrix of the *lumped* capacitive network

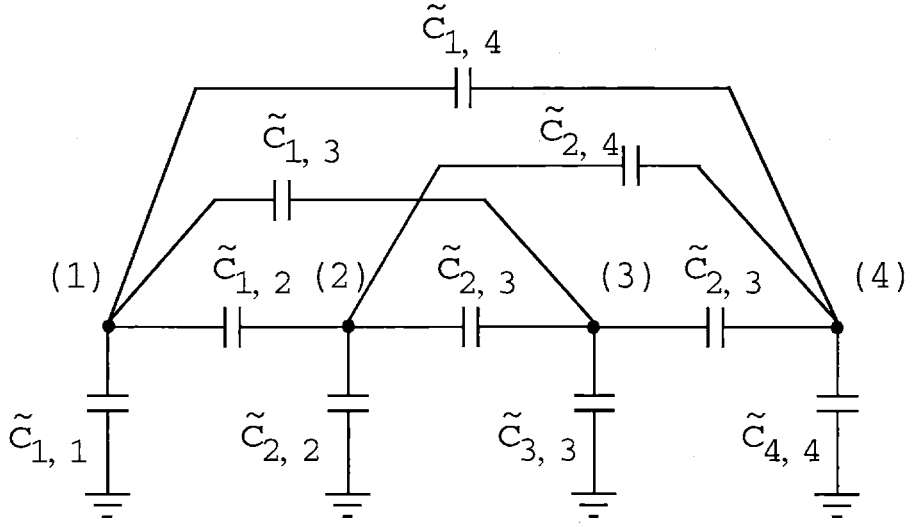

Figure 4: Equivalent capacitive network ($n = 4$)

in Figure 4 ($n = 4$ for simplicity). Even more, it is

$$c_{i,j}^t = \begin{cases} \sum_{k=1}^n \tilde{c}_{i,k} & \text{if } i = j \\ -\tilde{c}_{i,j} & \text{if } i \neq j \end{cases} \tag{20}$$

By expanding the right part of (16), we have that the energy drawn from V_{dd} during the transition is

$$\begin{aligned} E_i &= V_i^f e_i^T C^t (V^f - V^i) \\ &= V_i^f (V_i^f - V_i^i) c_{i,i}^t + \sum_{j,\, j \neq i} V_i^f (V_j^f - V_j^i) c_{i,j}^t \\ &= V_i^f (V_i^f - V_i^i) \sum_{j=1}^n \tilde{c}_{i,j} - \sum_{j,\, j \neq i} V_i^f (V_j^f - V_j^i) \tilde{c}_{i,j} \end{aligned} \tag{21}$$

We notice that this is exactly the energy that would be drawn from the power supply by the i^{th} driver during the transition if the lines of the bus and the parasitic capacitances of the drivers and the receivers were replaced by the network of Figure 4. The equivalent circuit

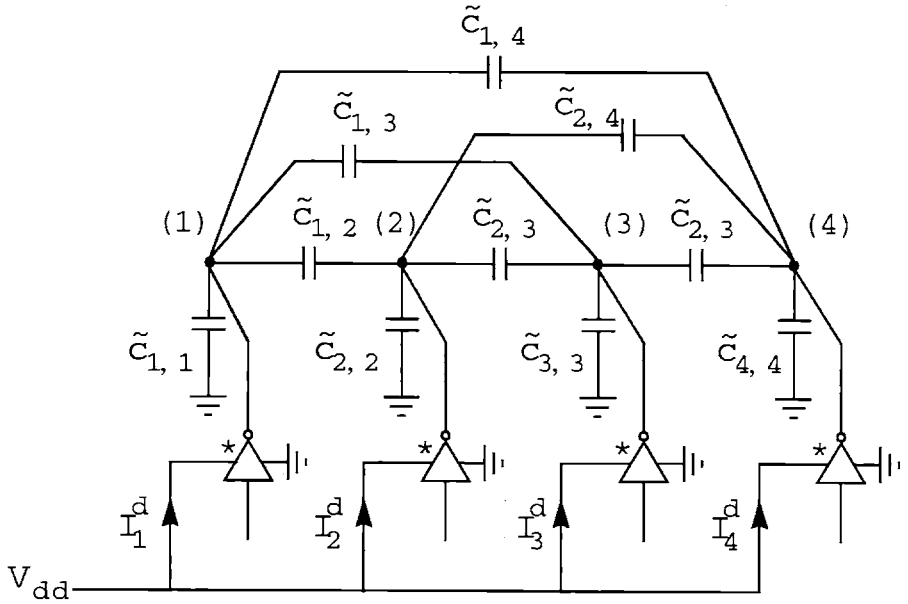

Figure 5: Energy-equivalent lumped DSM bus model (*Drivers without parasitic capacitances)

is shown in Figure 5 where the parasitic capacitances of the drivers have been removed. We call the capacitive network of Figure 5, the *lumped energy-equivalent DSM bus model.* **Notice that the energy E_i drawn from V_{dd} by the i^{th} driver may be negative!** For example suppose that $\tilde{c}_{1,2}$ is nonzero, that the 1^{st} line remains connected to V_{dd} during the transition i.e. $V_1^f = V_1^i = V_{dd}$ and that all other lines transit from zero to V_{dd} i.e. $V_2^i = V_3^i = \cdots = V_n^i = 0$ and $V_2^f = V_3^f = \cdots = V_n^f = V_{dd}$. Then, current will flow from the 1^{st} line back to the power supply (Figure 5).

Although individual drivers may return energy to the power supply, **the total energy drawn from it during the transition, $E = \sum_{i=1}^{n} E_i$ is always nonnegative** (Appendix B). Using Equation (16), the total energy E can be written in the form,

$$\boldsymbol{E = (V^f)^T C^t (V^f - V^i)} \tag{22}$$

Finally, the model of Figure 5 can be directly generalized to capture energy losses due to coupling between bus and *non-bus* lines (or lumped nodes). Suppose that the bus has n lines as before and some of them are capacitively (and possibly inductively) coupled to m other non-bus lines (or lumped nodes). We can expand the model of Figure 5 by adding m more nodes. Under assumptions on the transitions of the voltages of the additional lines (or nodes) similar to that of Section 2.2 we can define the expanded vectors of initial and final voltages V^i and V^f respectively. They both have $n+m$ coordinates the first n of which correspond to bus lines. The matrix C^t must be expanded appropriately. Then the energy drawn during the transition from the power supply by the bus drivers is

$$E = \sum_{i=1}^{n} V_i^f e_i^T C^t (V^f - V^i) \tag{23}$$

Note that here e_i has $n+m$ coordinates.

4 Simplified Energy-Equivalent Model

Traditionally, the bus lines are laid parallel and coplanar. In this case most of the electric field is trapped between the adjacent lines and the ground. This implies that the capacitance between non-adjacent lines is practically negligible relatively to the capacitance between adjacent lines or the capacitance between the lines and ground. Therefore an approximate energy-equivalent bus model can ignore the parasitics between non-adjacent lines [15, 19, 22, 25]. In this case the triplet, drivers-lines-receivers behaves energy-wise as the

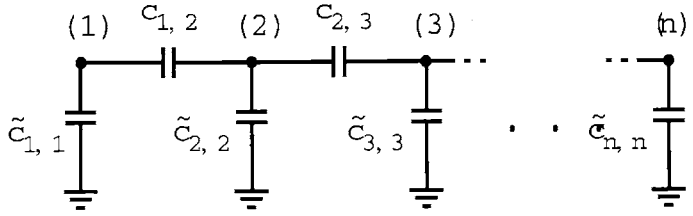

Figure 6: Energy-wise approximate DSM bus model

network of Figure 6, and the matrix C^t is reduced to the tri-diagonal matrix: $C^t =$

$$\begin{bmatrix} \tilde{c}_{1,1}+\tilde{c}_{1,2} & -\tilde{c}_{1,2} & 0 & \cdots & 0 & 0 \\ -\tilde{c}_{1,2} & \tilde{c}_{1,2}+\tilde{c}_{2,2}+\tilde{c}_{2,3} & -\tilde{c}_{2,3} & \cdots & 0 & 0 \\ 0 & -\tilde{c}_{2,3} & \tilde{c}_{2,3}+\tilde{c}_{3,3}+\tilde{c}_{3,4} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & \tilde{c}_{n-2,n-1}+\tilde{c}_{n-1,n-1}+\tilde{c}_{n-1,n} & -\tilde{c}_{n-1,n} \\ 0 & 0 & 0 & \cdots & -\tilde{c}_{n-1,n} & \tilde{c}_{n-1,n}+\tilde{c}_{n,n} \end{bmatrix}$$

This model can be further simplified if we assume that all grounded capacitors (except perhaps the boundary ones due to fringing effects) have the same values, and that all the inter-line capacitances are also of the same value. Say, $\tilde{c}_{2,2} = \tilde{c}_{3,3} = \cdots = \tilde{c}_{n-1,n-1} = C_L$ (subscript L for line capacitance) and $\tilde{c}_{1,2} = \tilde{c}_{2,3} = \cdots = \tilde{c}_{n-1,n} = C_I$ (subscript I for inter-line). Finally, $\tilde{c}_{1,1} = \tilde{c}_{n,n} = C_L + C_F$ (subscript F for fringing). (Note that by definition, C_L includes the capacitances of the driver and receiver). In this case the approximate model is

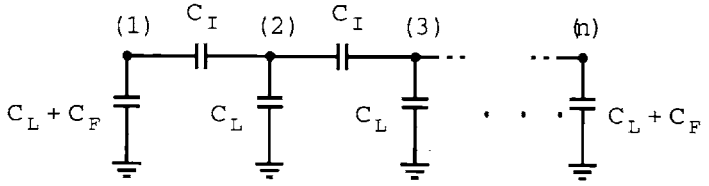

Figure 7: Simple energy-wise approximate DSM bus model

as in Figure 7 and C^t becomes C^{ta} (a for approximate) as in (24),

$$C^{ta} = \begin{bmatrix} 1+\lambda+\zeta & -\lambda & 0 & \cdots & 0 \\ -\lambda & 1+2\lambda & -\lambda & \cdots & 0 \\ 0 & -\lambda & 1+2\lambda & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1+\lambda+\zeta \end{bmatrix} C_L, \tag{24}$$

where we have set,

$$\lambda = \frac{C_I}{C_L}, \quad \zeta = \frac{C_F}{C_L} \tag{25}$$

The parameters λ and ζ depend on the technology as well as the specific geometry, metal layer and shielding of the bus. In general, ζ is between zero and λ. The parameter λ tends to increase with technology scaling. For standard 0.18μm technologies and minimum distance between the wires, λ is between 3 and 6 depending on the metal layer. It is expected to be even larger in 0.13μm. For wide buses the C_F (or 'ζ') terms can be ignored, since they do not contribute significantly to the total energy consumption.

5 Spice Simulations

$(V_1^i, V_2^i, V_3^i) \rightarrow (V_1^f, V_2^f, V_3^f)$

	000	001	010	011	100	101	110	111
000	0	6.02	11.3	6.97	6.02	12.0	6.97	3.14
	0	6.07	11.1	7.07	6.07	12.1	7.07	3.00
001	0	0	16.0	6.00	6.03	6.02	12.0	1.91
	0	0	16.2	6.07	6.07	6.07	12.1	2.00
010	0	11.1	0	0.96	11.1	21.4	0.96	2.10
	0	11.1	0	1.00	11.1	22.3	1.00	2.00
011	0	5.06	5.06	0	11.1	16.1	6.02	0.96
	0	5.07	5.07	0	11.1	16.2	6.07	1.00
100	0	6.02	16.1	12.0	0	6.02	6.00	1.92
	0	6.07	16.2	12.1	0	6.07	6.07	2.00
101	0	0	20.6	11.0	0	0	11.0	0.96
	0	0	21.3	11.1	0	0	11.1	1.00
110	0	11.1	5.06	6.03	5.06	16.1	0	0.96
	0	11.1	5.07	6.07	5.07	16.2	0	1.00
111	0	5.07	10.1	5.06	5.07	10.1	5.06	0
	0	5.07	10.1	5.07	5.07	10.1	5.07	0

Table 1: Normalized energy drawn from V_{dd} during the transitions of a 3-line DSM bus with $\lambda = 5.06$ (simulation: upper/ model: lower)

To verify the simplified model of Figure 7, a three line bus along with its drivers has been designed and simulated using HSPICE. In the design 0.18μm technology was used and the wires had minimal distance to each other and length of 2mm. The technology file gave $\lambda = 5.06$ (for metal layer 1). The normalized energies measured using HSPICE simulation, as

well as the theoretical ones calculated using the expression $E = V^f C^{ta}(V^f - V^i)$ (with $\zeta = 0$) are shown in Table 1. For every transition, the upper number is the HSPICE measurement and the lower one is the calculated value. The variations are within 5.2% and are mostly due to numerical errors in the energy integration.

6 Deep Sub-Micron vs. Standard Bus Energy Model

In this section the simple DSM bus model of Figure 7 is compared energy-wise to the standard bus model in Figure 8. The standard model has been used and is being used extensively for energy estimation. Although it is convenient for theoretical analysis, it gives misleading results for sub-micron technology buses.

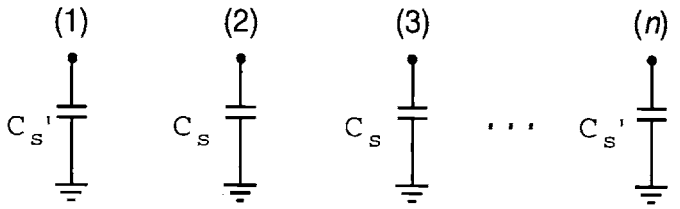

Figure 8: The standard model used for comparison

6.1 Energy Comparison

In Table 2 we see the behavioral differences between the DSM and the standard energy models. It shows the energy drawn from V_{dd} during the different transitions as it was estimated by the two models (Figures 7 and 8) for the case of a three line bus $n = 3$ with $\lambda = 3$. For a fair comparison of the two models, the capacitances of the standard bus were set to $C_s = C_L + 2C_I$ and $C_s' = C_L + C_I$. The supply voltage V_{dd} and the line-to-ground capacitance C_L have been normalized to one, and it was assumed that $\zeta = 0$. The upper numbers correspond to the DSM model and the lower ones to the standard model. There is difference not only in the average energy, but most importantly in the pattern of energy consumption. Some transitions are relatively more expensive according to one model than

$[V_1^i, V_2^i, V_3^i]$ \ $[V_1^f, V_2^f, V_3^f]$	000	001	010	011	100	101	110	111
000	0	4	7	5	4	8	5	3
	0	4	7	11	4	8	11	15
001	0	0	10	4	4	4	8	2
	0	0	7	7	4	4	11	11
010	0	7	0	1	7	14	1	2
	0	4	0	4	4	8	4	8
011	0	3	3	0	7	10	4	1
	0	0	0	0	4	4	4	4
100	0	4	10	8	0	4	4	2
	0	4	7	11	0	4	7	11
101	0	0	13	7	0	0	7	1
	0	0	7	7	0	0	7	7
110	0	7	3	4	3	10	0	1
	0	4	0	4	0	4	0	4
111	0	3	6	3	3	6	3	0
	0	0	0	0	0	0	0	0

Table 2: Normalized transition energy drawn from V_{dd} estimated using the DSM ($\lambda = 3$)(upper) and the standard (lower) model

the other one. These large differences are due to the interaction between the lines that is captured only by the DSM model. To illustrate this, let i be an intermediate line of the bus. Replacing C^t by C^{ta} in (16) we get:

$$\begin{aligned} E_i &= V_i^f e_i^T C^{ta}(V^f - V^i) \\ &= V_i^f[-\lambda, 1+2\lambda, -\lambda] \begin{bmatrix} V_{i-1}^f - V_{i-1}^i \\ V_i^f - V_i^i \\ V_{i+1}^f - V_{i+1}^i \end{bmatrix} C_L \end{aligned} \tag{26}$$

Depending on the values of , $V_{i-1}^i, V_{i-1}^f, V_i^i, V_i^f, V_{i+1}^i, V_{i+1}^f$, the energy E_i can take the values: -2λ, $-\lambda$, 0, λ, 2λ, 1, $1+\lambda$, $1+2\lambda$, $1+3\lambda$ or $1+4\lambda$ (times $V_{dd}^2 C_L$). On the contrary, for the standard model it is, $E_i = V_i^f C_s(V_i^f - V_i^i)$ which gives the possible values, 0 and $1+2\lambda$ (times $V_{dd}^2 C_L$). Figure 9 shows the histograms of the energies calculated with the standard and DSM models for the cases of $n = 3$ and $n = 8$ and with $\lambda = 3$.

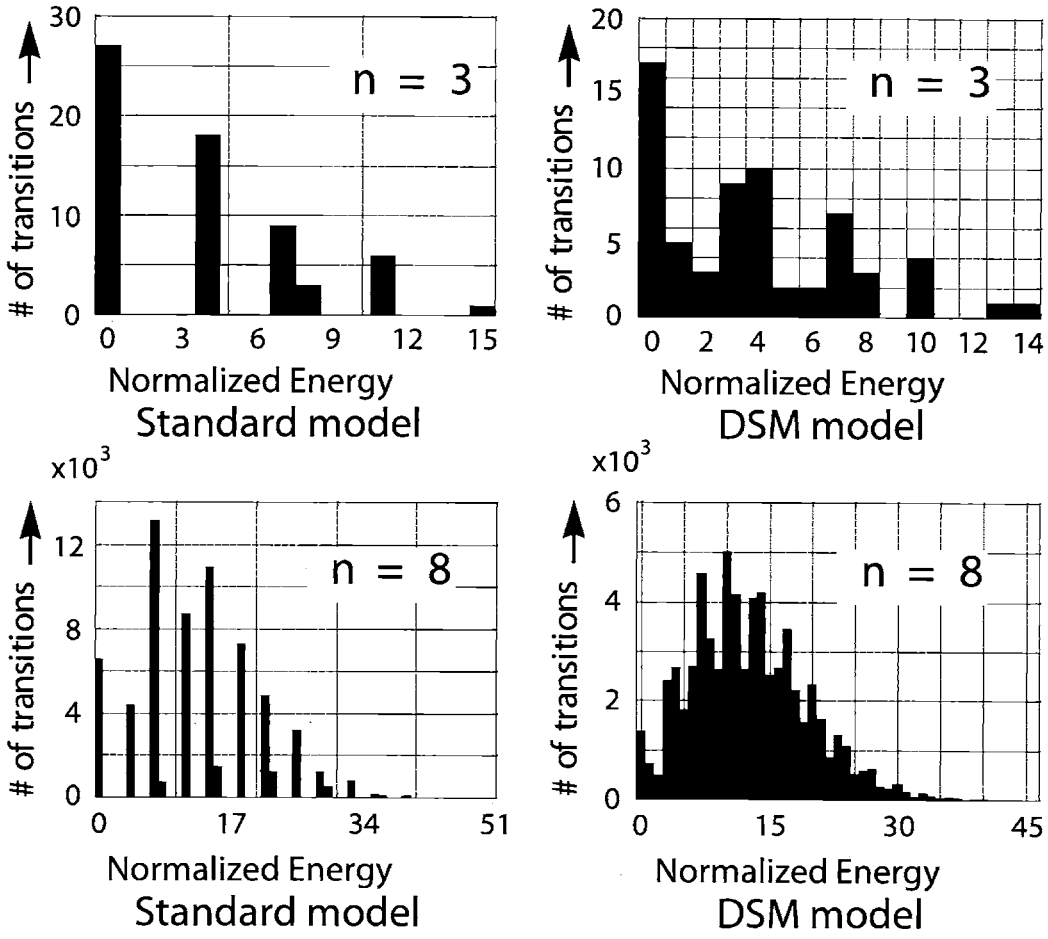

Figure 9: Energy histograms based on the standard and DSM ($\lambda = 3$)energy models

7 Energy Distribution and Statistical Energy Behavior of Deep Sub-Micron Buses

The purpose of this section is twofold. First the energy formula (22) is utilized to reveal how the energy drawn from V_{dd} is distributed in a bus, i.e. how much energy is (re)stored in the capacitors and how much is dissipated on the resistors. Second, the statistical average energy consumption of the bus is evaluated when the bus is driven by sequences of random data. This is done with the definition of the *transition activity matrix*, a generalization of the *transition activities* of individual lines.

7.1 Distribution of the Energy Drawn from Vdd

Let $V_1, V_2, \cdots, V_n$ be the voltages along the lines. The energy E_c stored in all the capacitances in the drivers, bus and receivers can be calculated using the lumped bus model of Figure 4. It is

$$E_c = \frac{1}{2}\sum_{i=1}^{n} \tilde{c}_{i,i} V_i^2 + \frac{1}{2}\sum_{i<j} \tilde{c}_{i,j}(V_i - V_j)^2$$

or in vector form (using Equation (20)),

$$E_c = \frac{1}{2} V^T C^t V \tag{27}$$

where $V = [V_1, V_2, \cdots, V_n]^T$.

In the beginning of the transition, $t = 0$, the lines have their initial voltages, $V^i = [V_1^i, V_2^i, \cdots, V_n^i]^T$ and so the stored energy is

$$E_c^i = \frac{1}{2}(V^i)^T C^t V^i \tag{28}$$

At the end of the transition period, $t = T$, the lines have their final voltages $V^f = [V_1^f, V_2^f, \cdots, V_n^f]^T$ and so the stored energy is

$$E_c^f = \frac{1}{2}(V^f)^T C^t V^f \tag{29}$$

The difference of the two values, $\Delta E_c = E_c^f - E_c^i$, is the energy transferred from the power supply to the capacitors. Through energy conservation

$$E = E_r + \Delta E_c \tag{30}$$

where E_r is the energy dissipated in the drivers and the distributed resistance of the lines. From (22),(28),(29) and (30) we have

$$\begin{aligned} E_r &= E - \Delta E_c = E - \left(E_c^f - E_c^i\right) \\ &= (V^f)^T C^t (V^f - V^i) - \frac{1}{2}(V^f)^T C^t V^f + \frac{1}{2}(V^i)^T C^t V^i \\ &= \frac{1}{2}(V^f)^T C^t V^f + \frac{1}{2}(V^i)^T C^t V^i - (V^f)^T C^t V^i \end{aligned} \tag{31}$$

The matrix C^t is symmetric and so the following identity holds,

$$(V^f)^T C^t V^i = \frac{1}{2}(V^f)^T C^t V^i + \frac{1}{2}(V^i)^T C^t V^f$$

Therefore from (31), the dissipated energy (transformed into heat) is,

$$\boldsymbol{E_r} = \frac{\mathbf{1}}{\mathbf{2}}(\boldsymbol{V^f} - \boldsymbol{V^i})^{\boldsymbol{T}}\boldsymbol{C^t}(\boldsymbol{V^f} - \boldsymbol{V^i}) \tag{32}$$

The right side of Equation (32) is a positive definite quadratic form since C^t is a positive definite matrix (assuming no $c^t_{i,i}$ is zero). It is also a generalization of the classic expression $E = cv^2/2$ and can replace (22) when long term average energy has to be estimated. To see this suppose the bus is driven by a finite sequence of data corresponding to the sequence of voltage vectors $V(0), V(1), \cdots, V(m)$. There are m transitions and by (22) the *average energy drawn from V_{dd} per transition* is:

$$\overline{E} = \frac{1}{m}\sum_{k=1}^{m} V^T(k)C^t\left(V(k) - V(k-1)\right) \tag{33}$$

From (32) the average energy dissipated on the resistors per transition is:

$$\overline{E}_r = \frac{1}{m}\sum_{k=1}^{m} \frac{\left(V(k) - V(k-1)\right)^T C^t \left(V(k) - V(k-1)\right)}{2} \tag{34}$$

By expanding (33) and (34) we get

$$\overline{E}_r = \overline{E} + \frac{V^T(0)C^tV(0) - V^T(m)C^tV(m)}{2m} \tag{35}$$

and so $\overline{E}$ and $\overline{E}_r$ become asymptotically equal as $m \to \infty$.

7.2 Expected Energy - The Transition Activity Matrix

The standard model has given rise to the transition activities of the lines. Let $l_i(1), l_i(2), \cdots$ be a sequence of bits transmitted through the i^{th} line. For the k^{th} transition we have, $V_i^i(k) = V_{dd}l_i(k)$ and $V_i^f(k) = V_{dd}l_i(k+1)$. Then, according to the standard model (Figure 7 with $C_I = C_F = 0$), the energy drawn from the power supply by the driver of the i^{th} line is

$$E_i(k) = l_i(k+1)\left[l_i(k+1) - l_i(k)\right]V_{dd}^2 C_L \tag{36}$$

If the random data sequence $\{l_i(k)\}_k$ is stationary with the wide sense, its autocorrelation function is defined as,

$$R_i(r) = \mathcal{E}\left[l_i(k)l_i(k+r)\right] \tag{37}$$

where $\mathcal{E}$ stands for expectation. Then, the *expected energy per transition drawn by the i^{th} driver* can be expressed as,

$$\mathcal{E}\left[E_i(k)\right] = \left[R_i(0) - R_i(1)\right] V_{dd}^2 C_L \tag{38}$$

The factor,

$$a_i = R_i(0) - R_i(1) \tag{39}$$

is called the *transition activity* of the i^{th} line. So,

$$\mathcal{E}\left[E_i(k)\right] = a_i V_{dd}^2 C_L \tag{40}$$

According to the standard model (Figure 7 with $C_I = C_F = 0$), the expected energy per cycle drawn by the whole bus is

$$\mathcal{E}\left[E_i(k)\right] = \left\{\sum_{i=1}^{n} a_i\right\} V_{dd}^2 C_L \tag{41}$$

In the standard model, the energy drawn by a line is independent of the data transmitted on the other lines. This is not true for the DSM bus model. Transition activity on a line influences the adjacent lines as well.

Again, let $\{l_i(k)\}_k$, be the random bit sequences on the lines $i = 1, 2, \cdots, n$ respectively. Here we assume the sequences are jointly wide sense stationary. Let

$$R_{i,j}(r) = \mathcal{E}\left[l_i(k) l_j(k+r)\right] \tag{42}$$

be the cross-correlation of the sequences i and j. We can define the sequence of random vectors $L(k) = [l_1(k), l_2(k), \cdots, l_n(k)]^T$ and its autocorrelation matrix,

$$R(r) = [R_{i,j}(r)]_{i,j=1}^{n} = \mathcal{E}\left[L(k) L^T(k+r)\right] \tag{43}$$

The voltages of the lines follow their binary values, $V(k) = V_{dd} L(k)$. For the k^{th} transition we have that,

$$\begin{aligned} E(k) &= V^T(k+1) C^t [V(k+1) - V(k)] \\ &= tr\left\{V^T(k+1) C^t [V(k+1) - V(k)]\right\} \\ &= tr\left\{C^t \left[V(k+1) V^T(k+1) - V(k) V^T(k+1)\right]\right\}, \end{aligned} \tag{44}$$

where $tr(X)$ is the trace of matrix X. In (44) we used the identity [31],

$$tr(XY) = tr(YX) \tag{45}$$

From Equations (43) and (44) we get,

$$\mathcal{E}\left[E(k)\right] = V_{dd}^2 \, tr\left\{C^t[R(0) - R(1)]\right\} \tag{46}$$

The similarity between (41) and (46) becomes clear if we rewrite (41) as

$$\mathcal{E}\left[E(k)\right] = V_{dd}^2 \, tr\left\{\mathcal{I} \cdot C^t[R(0) - R(1)]\right\}$$

where $\mathcal{I}$ is the identity matrix. Identity (45) along with the symmetry of C^t and the fact that $tr(A) = tr(A^T)$ imply that,

$$tr\left\{C^t R(1)\right\} = tr\left\{R^T(1)\left(C^t\right)^T\right\} = tr\left\{C^t R^T(1)\right\} \tag{47}$$

Equation (47) allows us to write (46) in the following symmetric form

$$\mathcal{E}\left[E(k)\right] = V_{dd}^2 \, tr\left\{C^t\left[R(0) - \frac{R(1) + R^T(1)}{2}\right]\right\} \tag{48}$$

We define the *transition activity matrix* $\mathcal{A}$ as

$$\boldsymbol{\mathcal{A}} = \boldsymbol{R(0)} - \frac{\boldsymbol{R(1)} + \boldsymbol{R^T(1)}}{\mathbf{2}} \tag{49}$$

the elements of which are

$$\begin{aligned} a_{i,j} &= R_{i,j}(0) - \left[R_{i,j}(1) + R_{j,i}(1)\right]/2 \\ &= \mathcal{E}\left[l_i(k) l_j(k)\right] - \frac{\mathcal{E}\left[l_i(k) l_j(k+1) + l_j(k) l_i(k+1)\right]}{2} \end{aligned} \tag{50}$$

The transition activity matrix $\mathcal{A}$ is symmetric by its definition. This was because both the cross-activities $a_{i,j}$ and $a_{j,i}$ express the interaction between the lines i and j. The transition activity matrix is also positive definite [31] (for non degenerate statistics). To see this, we use the stationarity of $\{L(k)\}_k$ to write $\mathcal{A}$ as

$$\mathcal{A} = \frac{1}{2}\,\mathcal{E}\left[\left(L(k+1) - L(k)\right)\left(L(k+1) - L(k)\right)^T\right]$$

The definition of the transition activity matrix is in agreement with the standard definition of transition activities of the lines. From (39) and (50) it is, $a_i = a_{i,i}$. Finally, expression (48) is written as

$$\boldsymbol{\mathcal{E}}\left[\boldsymbol{E(k)}\right] = \boldsymbol{V_{dd}^2}\, \boldsymbol{tr}\left\{\boldsymbol{C^t \mathcal{A}}\right\} \tag{51}$$

A simpler form for the expected energy can be found if we replace C^t in (51) by its approximation C^{ta}. In this case we have,

$$\begin{aligned}\mathcal{E}\left[E(k)\right] &= V_{dd}^2\, tr\left\{C^{ta}\mathcal{A}\right\} \\ &= \left\{(1+\lambda+\zeta)(a_{1,1}+a_{n,n}) + (1+2\lambda)\sum_{i=2}^{n-1} a_i - 2\lambda\sum_{i=1}^{n-1} a_{i,i+1}\right\} V_{dd}^2 C_L.\end{aligned}$$

The above expression can be further simplified if we make the approximation $\zeta = \lambda$,

$$\mathcal{E}\left[E(k)\right] = \left\{(1+2\lambda)\sum_{i=1}^{n} a_i - 2\lambda\sum_{i=1}^{n-1} a_{i,i+1}\right\} V_{dd}^2 C_L$$

7.3 Example: Uniformly Distributed, IID Data

A common assumption in energy estimation is that the data sequences are sequences of specially and temporally independent random bits with uniform distribution. In other words, that the random bits $l_1(1), l_2(1), \cdots, l_n(1), l_1(2), l_2(2), \cdots, l_n(2), \cdots$ are independent random variables s.t. $P_r(l_i(k) = 0) = P_r(l_i(k) = 1) = 1/2$ for all i and k. Then,

$$\mathcal{E}\left[l_i(k) l_j(k+r)\right] = \begin{cases} 1/2 \text{ if } i = j \text{ and } r = 0 \\ 1/4 \text{ otherwise} \end{cases}$$

The transition activity matrix is $\mathcal{A} = \frac{1}{4}\mathcal{I}$ and the expected energy is

$$\mathcal{E}\left[E(k)\right] = \left\{\frac{n}{4} + \frac{n-1}{2}\lambda + \frac{\zeta}{2}\right\} V_{dd}^2 C_L \tag{52}$$

8 Conclusions

A DSM bus energy model has been introduced and studied. Closed form expressions for the transition energy of this model have been derived. The deterministic and statistical energy behavior of the DSM model have been discussed and compared to that of the simplistic standard bus model. The transition activity matrix has been introduced as a generalization of the transition activities of the lines.

9 Appendix A

To prove equation (16) we start by replacing (14) into (15),

$$\begin{aligned} E_i &= V_i^f \int_0^T \left\{ I_i(0,t) + C_i^d \frac{\partial V_i(0,t)}{\partial t} \right\} dt \\ &= V_i^f \cdot \int_0^T I_i(0,t)dt + V_i^f C_i^d \int_0^T \frac{\partial V_i(0,t)}{\partial t} dt \\ &= V_i^f \int_0^T I_i(0,t)dt + V_i^f C_i^d \left(V_i^f - V_i^i\right) \end{aligned} \tag{53}$$

Now we evaluate the time integral of $I_i(0,t)$ using equation (1). Integration of both sides of (1) on x over $[0, L_p]$ gives:

$$I(0,t) = I(L_p,t) + \int_0^{L_p} C(x) \frac{\partial V(x,t)}{\partial t} dx \tag{54}$$

From Figure 3, $I_i(L_p,t) = C_i^r \frac{\partial V_i(L_p,t)}{\partial t}$, or written in vector form, $I(L_p,t) = C^r \frac{\partial V(L_p,t)}{\partial t}$. By replacing the last one into (54) and integrating over the time period we get

$$\begin{aligned} \int_0^T I(0,t) &= \int_0^T C^r \frac{\partial V(L_p,t)}{\partial t} dt + \int_0^T \left(\int_0^{L_p} C(x) \frac{\partial V(x,t)}{\partial t} dx \right) dt \\ &= C^r (V^f - V^i) + \int_0^T \left(\int_0^{L_p} C(x) \frac{\partial V(x,t)}{\partial t} dx \right) dt \end{aligned} \tag{55}$$

The matrix function $C(x)\frac{\partial V(x,t)}{\partial t}$ is Lebesgue integrable [30] within $[0, L_p] \times [0, T]$. This allows the use of Fubini's theorem [30] and the change of the integration order in the double integral of (55). It is:

$$\begin{aligned} \int_0^T \left(\int_0^{L_p} C(x) \frac{\partial V(x,t)}{\partial t} dx \right) dt &= \int_0^{L_p} \left(\int_0^T C(x) \frac{\partial V(x,t)}{\partial t} dt \right) dx \\ &= \int_0^{L_p} C(x) \left(\int_0^T \frac{\partial V(x,t)}{\partial t} dt \right) dx \\ &= \int_0^{L_p} C(x)(V^f - V^i) dx \\ &= \left(\int_0^{L_p} C(x) dx \right) (V^f - V^i) \end{aligned} \tag{56}$$

Expressions (55) and (56) give,

$$\int_0^T I(0,t)dt = \left[C^r + \int_0^{L_p} C(x)dx \right] (V^f - V^i) \tag{57}$$

Since it is $I_i(0,t) = e_i^T I(0,t)$, where e_i has a one in the i^{th} position and zero everywhere else, and since C^d is diagonal, Equations (53) and (57) imply that,

$$\begin{aligned} E_i &= V_i^f \int_0^T I_i(0,t)dt + V_i^f C^d (V_i^f - V_i^i) \\ &= V_i^f e_i^T \left[C^r + \int_0^{L_p} C(x)dx \right] (V^f - V^i) + V_i^f C^d (V_i^f - V_i^i) \\ &= V_i^f e_i^T \left[C^d + C^r + \int_0^{L_p} C(x)dx \right] (V^f - V^i) \end{aligned}$$

We have set $C^t = C^d + C^r + \int_0^{L_p} C(x)dx$ and therefore it is,

$$E_i = V_i^f e_i^T C^t (V^f - V^i) \tag{58}$$

10 Appendix B

Here we show that for any transition of the bus, the total energy $E = \sum_{i=1}^n E_i$ is nonnegative. Matrix C^t, whose elements are given by (20) can written as

$$C^t = \begin{bmatrix} \tilde{c}_{1,1} & & & \\ & \tilde{c}_{2,2} & 0 & \\ & 0 & \ddots & \\ & & & \tilde{c}_{n,n} \end{bmatrix} + \sum_{i<j} \left(e_i e_i^T - e_i e_j^T - e_j e_i^T + e_j e_j^T \right) \tilde{c}_{i,j}.$$

Starting from Equation (22) and using the above expression, the transition energy is written as:

$$\begin{aligned} E &= V^f C^t (V^f - V^i) \\ &= \sum_{i=1}^n V_i^f (V_i^f - V_i^i) \tilde{c}_{i,i} + \sum_{i<j} \left\{ V_i^f (V_i^f - V_i^i) + V_j^f (V_j^f - V_j^i) - V_i^f (V_j^f - V_j^i) - V_j^f (V_i^f - V_i^i) \right\} \tilde{c}_{i,j} \\ &= \sum_{i=1}^n V_i^f (V_i^f - V_i^i) \tilde{c}_{i,i} + \sum_{i<j} (V_j^f - V_i^f) \left[(V_j^f - V_i^f) - (V_j^i - V_i^i) \right] \tilde{c}_{i,j} \end{aligned}$$

It is easy to verify that $V_i^f (V_i^f - V_i^i) \geq 0$ and $(V_j^f - V_i^f) \left[(V_j^f - V_i^f) - (V_j^i - V_i^i) \right] \geq 0$ for all values of V^i and V^f. Therefore $E \geq 0$.

References

[1] S. Borkar, "Low power design challenges for the decade", *Asia and South Pacific Design Automation Conference 2001, pp. 293-296.*

[2] T. Sakurai, "Design Challenges for $0.1\mu m$ and Beyond", *Asia and South Pacific Design Automation Conference 2000, pp. 553-558.*

[3] B. Bishop, M. Irwin, "Databus charge recovery: Practical considerations", *IEEE/ACM International Symposium on Low Power Electronics and Design 1999, pp. 85-87.*

[4] H. Yamauchi, H. Akamatsu, T. Fujita, "An asymptotically zero power charge-recycling bus architecture for battery-operated ultrahigh data rate ULSIs", *IEEE Journal of Solid-State Circuits, pp. 423-431, April. 1995.*

[5] K. Khoo, A. Willson Jr., "Charge recovery on a databus", *IEEE/ACM International Symposium on Low Power Electronics and Design 1995, pp. 185-189.*

[6] E. Kush, J. Rabaey, "Low-Energy Embedded FPGA Structures", *IEEE/ACM International Symposium on Low Power Electronics and Design 1998, pp. 155-160.*

[7] P. Sotiriadis, T. Konstantakopoulos, A. Chandrakasan, "Analysis and Implementation of Charge Recycling for Deep Sub-micron Buses", *IEEE/ACM International Symposium on Low Power Electronics and Design 2001, pp. 364-369.*

[8] S. Ramprasad, N. Shanbhag, I. Hajj, "A coding framework for low-power address and data busses", *IEEE Trans. on VLSI Systems, Vol. 7, No. 2, June 1999, pp. 212-221.*

[9] M. Stan, W. Burleson, "Low-power encodings for global communication in cmos VLSI", *IEEE Transactions on VLSI Systems, pp. 49-58, Vol. 5, No. 4, Dec. 1997.*

[10] P. Sotiriadis, A. Chandrakasan, "Low power bus coding techniques considering inter-wire capacitances", *Custom Integrated Circuits Conference 2000, pp. 507-510. CICC.*

[11] P. Sotiriadis, A. Chandrakasan, "Bus Energy Minimization by Transition Pattern Coding (TPC) in Deep Sub-Micron Technologies", *IEEE/ACM International Conference on CAD 2000, pp. 322-327.*

[12] Y. Nakagome, K. Itoh, M. Isoda, K. Takeuchi, M. Aoki, "Sub 1V swing internal bus architecture for future low-power ULSIs", *IEEE Journal of Solid-State Circuits, pp. 414-419, April 1993.*

[13] M. Hiraki, H. Kojima, H. Misawa, T. Akazawa, Y. Hatano, "Data-dependent logic swing internal bus architecture for ultra low-power LSIs", *IEEE Journal of Solid-State Circuits, pp. 397-401, April. 1995.*

[14] H. Zhang, J. Rabaey, "Low swing interconnect interface circuits", *IEEE/ACM International Symposium on Low Power Electronics and Design, August 1998, pp. 161-166.*

[15] J. Davis, J. Meindl, "Compact Distributed RLC Interconnect Models Part II: Coupled Line Transient Expressions and Peak Crosstalk in Multilevel Networks", *IEEE transactions on electron devices, Vol. 47, No. 11, Nov. 2000.*

[16] K. Yamashita, S. Odanaka, "Interconnect Scaling Scenario Using a Chip Level Interconnect Model", *IEEE transactions on electron devices Vol. 47, No. 1, Jan. 2000.*

[17] N. Menezes, L. Pillegi, "*Analyzing on-chips Interconnect effects*", Design of High Performance Microprocessor Circuits, A. Chandrakasan, W.J. Bowhill, F. Fox, Eds., IEEE Editions 2001, Chapter 16.

[18] S. Morton, "*Techniques for Driving interconnect*", Design of High Performance Microprocessor Circuits, A. Chandrakasan, W.J. Bowhill, F. Fox, Eds., IEEE Editions 2001, Chapter 17.

[19] S. Das, W. Smith, C. Paul, "Modeling of data bus structures using numerical methods", *International Symposium on Electromagnetic Compatibility, 1993, pp. 409-414.*

[20] A. Ruehli, Ed. *Circuit analysis, simulation, and design*, Vol.1,2. Elsevier Science Pub. Co., 1986-1987.

[21] S. Wong, G-Y. Lee, D-J. Ma, "Modeling of Interconnect Capacitance, Delay, and Crosstalk in VLSI", *IEEE Transactions on Semiconductor Manufacturing, Vol.13, No. 1, Feb. 2000, pp. 108-111.*

[22] C. Cheng, J. Lillis, S. Lin, N. Chang, *Interconnect Analysis and Synthesis*, John Wiley and Sons, 2000.

[23] L. Pileggi, "Coping with RC(L) interconnect design headaches", *IEEE/ACM International Conference on CAD 1995, pp. 246-253.*

[24] Y. Ismail, E. Friedman, J. Neves, "Transient power in CMOS gates driving LC transmission lines", *IEEE International Conference on Electronics, Circuits and Systems, 1998, pp. 337-340.*

[25] J. Rabaey, *Digital Integrated circuits*, Prentice Hall 1996.

[26] T. Sakurai, "Closed-form expressions for interconnection delay, coupling, and crosstalk in VLSIs", *IEEE Transactions on Electron Devices, Vol. 40, Issue 1, Jan. 1993, pp. 118-124.*

[27] K. Tang, E. Friedman, "Peak noise prediction in loosely coupled interconnect [VLSI circuits]", *International Symposium on Circuit and Systems 1999, Vol.1, pp. 541-544.*

[28] T. Sakurai, S. Kobayashi, M. Noda, "Simple expressions for interconnection delay, coupling and crosstalk in VLSI", *International Symposium on Circuit and Systems 1991, Vol. 4, pp. 2375-2378.*

[29] M. Becer, I. Hajj, "An analytical model for delay and crosstalk estimation in interconnects under general switching conditions", *IEEE International on Electronics, Circuits and Systems, 2000, vol.2, pp. 831-834.*

[30] F. Riesz and B. Nagy, *Functional Analysis*, Dover 1990.

[31] R. Horn and C. Johnson, *Matrix Analysis*, Cambridge University Press 1994.

[32] A. Ruehli, "Equivalent Circuit Models for Three-Dimensional Multiconductor Systems", *IEEE transactions on Microwave Theory and Applications, MTT-22 (3), 216-221, March 1974.*

Chapter 3

Power Reduction Using Coding, The Fundamental Information Theoretic Limits

1 Introduction

We consider a computation device, $\mathcal{D}$, that can be modelled as a finite state machine with an internal state S_t, an input I_t and an output O_t at time $t = 1, 2, 3, \ldots$. For practical reasons we can assume that S_t, I_t and O_t are vectors of bits. Given a sequence of input vectors $I_1, I_2, \ldots, I_T$ and an initial state S_0, the device has some computation cost (such as time, energy, aging etc.) at each time $t = 1, 2, \ldots, T$. We assume that the cost of computation is known as a function of the input sequence and the initial state.

In the above example the input can be regarded as "*uncoded*", in the sense that all input sequences of vectors are possible. Suppose now that we can expand the device by adding *redundancy* in the input port, the internal states and the output port. This would result to a new device $\mathcal{D}'$ that may have higher operating cost. *The fundamental question is*: Can we *encode* the input and *decode* the output sequences of the new device $\mathcal{D}'$ so that it behaves like the original device $\mathcal{D}$ while operating at a cost lower than that of $\mathcal{D}$?

Naturally, the existence of such a coding scheme implies that the original input and output sequences can be mapped injectively into low cost sequences of the new device.

Note that this general framework of computation devices accounts for communication models as well. In this case the device reproduces the input to its output. In the classical information theory, the cost is always the probability of error, and coding is applied to reduce it. This scenario has a well established history.

On-chip (or inter-chip) communication, through channels called *buses*, is another communication paradigm where the major cost is the energy consumption. In this Chapter we use the energy model of the buses that have been studied extensively in Chapter 2 in order to illustrate our theory. This approach allows for a natural introduction of the concepts and a smooth development of the results. *The theory applies to any combinational circuit as well and can be expanded directly to include any computation device modelled as a finite state machine with cost.* The most general case though, requires an enormous amount of technicalities.

In the case of buses we also refer to the input vector at time t as the *state*. It is convenient to use both names *input* and *state* for one object, the vector that is transmitted. The reason is that although the bus does not have memory as a computation device (unlike a general finite state machine), it does have memory (and so state) when it comes to energy consumption.

We write S_t for input vector of the bus at time t, where $S_t = (s_t^1, s_t^2, s_t^3, \ldots, s_t^n)$ and $s_t^1, \ldots, s_t^n$ are the bits transmitted through lines $1, 2, \ldots, n$ respectively at time t. Although the bits take the values 0 or 1, they will in some cases be treated as *real* and not as *binary* numbers. The energy cost of the transition from state S_t to state S_{t+1} is given by (see Chapter 2 or Ref. [19]) :

$$\mathcal{E}(S_t \rightarrow S_{t+1}) = E_0 \, (S_{t+1} - S_t)\mathcal{B}(S_{t+1} - S_t)' \tag{1}$$

where V' is the *transpose* of vector V and

$$\mathcal{B} = \begin{pmatrix} 1+2\lambda & -\lambda & 0 & 0 & \cdots & 0 & 0 & 0 \\ -\lambda & 1+2\lambda & -\lambda & 0 & \cdots & 0 & 0 & 0 \\ 0 & -\lambda & 1+2\lambda & -\lambda & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \cdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & -\lambda & 1+2\lambda & -\lambda \\ 0 & 0 & 0 & 0 & \cdots & 0 & -\lambda & 1+2\lambda \end{pmatrix}. \tag{2}$$

The parameter λ is non-negative and depends on the physical properties of the lines such as geometry, size and distances between them as well as the type of technology used to manufacture the bus. The constant E_0 also depends on the technology and the physical design of the bus. For the obsolete *Long Channel* (LC) technologies, λ is practically zero, and $\mathcal{B}$ reduces to a scalar matrix. For modern *Deep Sub-micron technologies* (DSM) λ can be as high as 8 (for example in $0.13\mu m$ technologies). The total energy dissipation corresponding to a sequence $S_1, S_2, S_3, \ldots, S_T$ is given by $\sum_{t=1}^{T-1} \mathcal{E}(S_t \rightarrow S_{t+1})$. We can also consider the transition to S_1 from the initial state S_0 that has cost equal to $\mathcal{E}(S_0 \rightarrow S_1)$. Based on the above discussion, we can restate our coding problem for the case of buses.

The Coding Problem for Buses: Is it possible to reduce the expected energy per transmitted bit by adding more lines in the bus? If so, what are the achievable limits and the affiliated coding schemes?

The key element here is that we add extra lines in the bus while the data stream transmitted remains unchanged. By doing this we pay an extra area cost on the microchip, but we also get a communication channel of higher capacity. Therefore the question stated above can be reformed as : what is the relation between additional capacity on the channel and energy reduction ?

Note that redundancy of the channel's (bus) capacity can result not only by bus expansion but also by data rate reduction. From an application point of view, this is a completely different problem. From a theoretical point of view, the question is exactly the same as before: what is the best relation between the rate information being transmitted and the rate energy being consumed?

There are some places in microprocessor architecture, where redundancy exists because of highly "temporally" or/end "spatially" correlated data. A particular example is address buses where the amount of information transmitted each time is less than a bit while the size of the bus may be 8, 16 or 32. In such cases, from an information theoretic perspective, there is a tremendous amount of intrinsic redundancy that can be exploited to reduce energy consumption. How much energy reduction is possible?

Note that when $\lambda = 0$ the cost function of the general bus with n lines decomposes into the sum $\sum_{j=1}^{n}(s_{t+1}^{j} - s_t^j)^2$. This means that a bus with n lines imposes a relation between energy and bit rate identical to that of a bus with only one line. This relation has been studied in [12]. Related work involving theoretical and practical aspects of the problem of estimating energy consumption and coding design for energy reduction when $\lambda = 0$ has been presented in [2, 5, 11, 13, 16, 17, 18, 15, 6, 14, 20, 21, 22].

In the case of DSM technologies ($\lambda > 0$), unlike the case of LC technologies ($\lambda = 0$), the above cost function includes terms depending on the relation between bit values transmitted on different lines. This makes the treatment of the above coding problem in the DSM case a challenging task. The problem is addressed in the present chapter.

In Section 2, we consider the above coding problem. Using a differential coding scheme where the codewords are carefully chosen to have low Hamming weights with high probabilities, we compute a general upper-bound on the minimum possible average energy consumption in a DSM bus. When $\lambda = 0$, our bound is given by an explicit simple formula and coincides with that of [12]. In this section we also define the expected energy consumption per bit and the utilization of the bus by a stationary process.

In Section 3, we provide a non-constructive coding scheme based on the typical sequences of certain ergodic stationary Markov processes. These schemes lead to more powerful existence results and higher reductions than those of Section 2. In contrast, it is harder to compute these reductions numerically, except for small numbers of lines. These improvements motivate us to consider the category of stationary processes whose states correspond to the states of the bus.

We prove that for every stationary process there is a stationary ergodic Markov process of the same entropy rate (bus utilization) and less or equal energy per bit. As a consequence,

it is shown that the minimum possible energy per bit at a given utilization of the bus is asymptotically achievable using a code whose codewords are (finite) typical sequences of an ergodic Markov process.

In Section 4, we study the properties of the minimum energy per bit as a function of the bus utilization and establish its continuity. We formulate the computation of this function as a convex numerical optimization problem with a unique answer and give an explicit form for general cost functions using standard optimization techniques.

Conclusions and final remarks are given in Section 6.

2 The Entropy Bound on Achievable Energy Reductions

In this section, we consider the bus model described in the previous section. Recall that the energy cost $\mathcal{E}(S_t \rightarrow S_{t+1})$ of the transition $S_t \rightarrow S_{t+1}$ is given by expression (1) where the $n \times n$ matrix $\mathcal{B}$ is given by (2). The constant E_0 depends on the technology, and n is equal to the number of lines in the bus [19]. For convenience we set $E_0 = 1$ throughout this chapter. Finally, recall that the total energy required to transmit a sequence $\mathbf{c} = (S_1, S_2, S_3, \ldots, S_L)$ is:

$$\mathcal{E}(\mathbf{c}) = \sum_{t=0}^{L-1} \mathcal{E}(S_t \rightarrow S_{t+1}) \tag{3}$$

where S_t is the input vector (and state) of the bus at $t > 1$ and S_0 is the initial state of the bus at $t = 0$.

Let $\mathcal{Q} = \{0,1\}^n$ be the set of all binary vectors of length n. For all t, the state (and input) S_t of the bus at time t is in $\mathcal{Q}$. For any $S \in \mathcal{Q}$, let $w(S)$ denote the Hamming weight of S. We will also use the notation of $S \oplus U$ for the binary sum of any two vectors $S, U \in \mathcal{Q}$. Recall that in the calculation of the transition energy $\mathcal{E}(S_t \rightarrow S_{t+1})$ through the use of expression (1) we regard S_t and S_{t+1} as *real* and not binary vectors.

In order to establish achievable upper-bounds on minimum possible average energy consumed per transmitted bit, we first have to establish some technical results. We start with the following fundamental Definitions and Lemma.

Definition 2.1 *We define a code $\mathcal{C}$ of length L as a set of sequences $\mathbf{c}$ of L successive bus input vectors $\sigma_1, \sigma_2, \ldots, \sigma_L$ where $\sigma_i \in \mathcal{Q}$. The probability that the codeword $\mathbf{c} \in \mathcal{C}$ is transmitted is denoted by $\Pr(\mathbf{c})$.*

Definition 2.2 *The entropy per use of the bus of a code $\mathcal{C}$ is defined as,*

$$\mathcal{H}(\mathcal{C}) = -\frac{1}{L} \sum_{\mathbf{c} \in \mathcal{C}} \Pr(\mathbf{c}) \log \Pr(\mathbf{c}) \tag{4}$$

where log *is the binary logarithm. The definition extends to the uncoded case where $L = 1$ and $\mathcal{C}$ is the set of all vectors in $\mathcal{Q}$, each transmitted with probability $1/2^n$.*

Transmitting a codeword means that we transmit a certain sequence of L successive elements of $\{0,1\}^n$ through the bus, so we use the bus L times.

Definition 2.3 *The expected energy consumption per use of the bus, when applying the code $\mathcal{C}$, is defined as :*

$$\mathcal{E}_{av}(\mathcal{C}) = \frac{1}{L} \sum_{\mathbf{c} \in \mathcal{C}} \Pr(\mathbf{c}) \mathcal{E}(\mathbf{c}) + \frac{1}{L} \sum_{\mathbf{c}, \mathbf{c}' \in \mathcal{C}} \Pr(\mathbf{c}) \Pr(\mathbf{c}') \mathcal{E}(\sigma_L \rightarrow \sigma'_1) \tag{5}$$

where σ_L and σ'_1 are the last and first entries (vectors) of the codewords $\mathbf{c}$ and $\mathbf{c}'$ respectively.

Note that the second term is due to the energy loss on the transition between the last state of $\mathbf{c}$ and the first state of $\mathbf{c}'$. We also agree that the energy of codewords with only one entry is zero, that is, $\mathcal{E}(\sigma) = 0$ for every $\sigma \in \mathcal{Q}$. So for codes of length one, $L = 1$, the expected energy per use of the bus becomes, $\mathcal{E}_{av}(\mathcal{C}) = \sum_{\sigma, \omega \in \mathcal{C}} \Pr(\sigma) \Pr(\omega) \mathcal{E}(\sigma \rightarrow \omega)$.

In the analysis that follows we are mostly interested in the case where the length of the code becomes arbitrarily large. In this case the second term becomes zero and can be ignored. In particular, if we set $\mathcal{E}_{\max} = \max_{\sigma, \omega \in \mathcal{Q}} \mathcal{E}(\sigma \rightarrow \omega)$, there exists a constant $e(\mathcal{C})$ such that $e(\mathcal{C}) \leq \mathcal{E}_{\max}$ and

$$\mathcal{E}_{av}(\mathcal{C}) = \frac{1}{L} \sum_{\mathbf{c} \in \mathcal{C}} \Pr(\mathbf{c}) \mathcal{E}(\mathbf{c}) + \frac{e(\mathcal{C})}{L}. \tag{6}$$

Note also that on average, we transmit $\mathcal{H}(\mathcal{C})$ information bits per L bus uses. If no code was used, with L uses of the bus we can transmit nL information bits. A definition follows naturally.

Definition 2.4 *We define the utilization, α, of the bus by the code $\mathcal{C}$ to be the ratio of the expected number of information bits transmitted per use of the bus over the number of actual bits transmitted per use of the bus. It is :*

$$\alpha = \frac{\mathcal{H}(\mathcal{C})/L}{n} \tag{7}$$

Furthermore, we define the expected energy consumption per information bit transmitted through the bus, when using code $\mathcal{C}$, as

$$\mathcal{E}_b(\mathcal{C}) = \frac{\mathcal{E}_{av}(\mathcal{C})}{\mathcal{H}(\mathcal{C})/L}. \tag{8}$$

In general we use the term *utilization* of the bus to denote the ratio of a given information rate (expected number of bits transmitted per bus use) over the number of lines n of the bus.

The expression (8) equals the ratio of the expected energy cost per bus use over the expected number of information bits transmitted per bus use.

In the case of the uncoded bus, the input vectors are uniformly distributed in the set $\mathcal{Q} = \{0,1\}^n$ and so the individual bits transmitted are independent random variables with probability 1/2. It can be verified directly using expression (1), with $E_0 = 1$, that the expected energy per use of the bus (per transition) is:

$$\overline{\mathcal{E}(S_t \rightarrow S_{t+1})} = n\frac{(1+2\lambda)}{2} \tag{9}$$

Throughout the chapter, *overline* will denote expectation with respect to all random variables involved in the expression.

Definition 2.5 *The expected energy per information bit in the case of the uncoded bus will be denoted by $\mathcal{E}_u$, that is:*

$$\mathcal{E}_u = \frac{(1+2\lambda)}{2} \tag{10}$$

Lemma 2.1 *Let X denote a random vector in $\mathcal{Q} = \{0,1\}^n$ whose components are i.i.d uniformly distributed in $\{0,1\}$. Let S be a constant vector in $\mathcal{Q}$ and let $\overline{\mathcal{E}(X \rightarrow S \oplus X)}$ denote the expected value of the random variable $\mathcal{E}(X \rightarrow S \oplus X)$. Then*

$$\overline{\mathcal{E}(X \rightarrow S \oplus X)} = 2w(S)\mathcal{E}_u. \tag{11}$$

where $\oplus$ is the binary addition and $w(S)$ is the Hamming weight of vector S.

Proof: Let $S = (s_1, s_2, \ldots, s_n)$, $X = (x_1, x_2, \ldots, x_n)$ and $Y = S \oplus X = (y_1, y_2, \ldots, y_n)$. According to (1) we have to compute the expectation of $\mathcal{E}(X \rightarrow Y) = (Y - X)\mathcal{B}(Y - X)'$. Matrix $\mathcal{B}$ being given by (2) results in

$$\mathcal{E}(X \rightarrow Y) = (1 + 2\lambda) \sum_{i=1}^{n} (y_i - x_i)^2 - 2\lambda \sum_{i=1}^{n-1} (y_i - x_i)(y_{i+1} - x_{i+1}) \tag{12}$$

Since we deal with the components of $Y - X$ as real numbers, we have $(y_i - x_i) = (-1)^{x_i} s_i$, hence $(y_i - x_i)^2 = s_i$ and $(y_i - x_i)(y_{i+1} - x_{i+1}) = (-1)^{(x_i + x_{i+1})} s_i s_{i+1}$. Replacing them in (12) we have

$$\mathcal{E}(X \rightarrow S \oplus X) = 2\mathcal{E}_u \sum_{i=1}^{n} s_i - 2\lambda \sum_{i=1}^{n-1} (-1)^{(x_i + x_{i+1})} s_i s_{i+1}.$$

The result follows by taking expectations of both sides and using the identity $w(S) = \sum_{i=1}^{n} s_i$. □

Definition 2.6 *We define a pair (α, β), where α denotes the utilization of the bus and β the expected energy per bit, to be achievable if and only if there exists an infinite sequence of codes with strictly increasing lengths that utilize the bus arbitrarily close to a number $\alpha_1 \geq \alpha$ and have expected energy consumption per bit that gets arbitrarily close to a number $\beta_1 \leq \beta$.*

Definition 2.7 *We define the limiting expected energy consumption per bit, $\mathcal{E}_b$, at utilization, $\alpha \in [0, 1]$, of the bus, to be the function*

$$\mathcal{E}_b(\alpha) = \inf \{\beta \mid \text{The pair } (\alpha, \beta) \text{ is achievable}\}. \tag{13}$$

The same symbol, $\mathcal{E}_b$, has been used for both the limiting expected energy per bit as well as the expected energy per bit of a given code $\mathcal{E}_b(\mathcal{C})$. The argument will determine which quantity we refer to. Let $\mathcal{E}_{\max} = \max_{\sigma, \omega \in \mathcal{Q}} \mathcal{E}(\sigma \rightarrow \omega)$, then for any $\alpha \in [0, 1]$, the pair $(\alpha, \mathcal{E}_{\max})$ is always achievable, thus the quantity $\mathcal{E}_b(\alpha)$ is well defined.

According to its definition, $\mathcal{E}_b(\alpha)$ is (asymptotically) the minimum possible energy we have to spend per bit transmitted through the bus, when the information rate is $\alpha\, n$ bits per bus use (transition).

Our first result establishes an upper-bound on $\mathcal{E}_b(\alpha)$.

Theorem 2.1 *Let $h^{-1}(\cdot) : [0,1] \rightarrow [0,\frac{1}{2}]$ denote the inverse of the binary entropy function $h(\cdot)$ when restricted to the domain $[0,\frac{1}{2}]$. Then for any utilization α and for any bus of size n*

$$\mathcal{E}_b(\alpha) \leq \frac{2h^{-1}(\alpha)}{\alpha}\,\mathcal{E}_u. \tag{14}$$

Proof: Let U be an information source of independent vectors $U \in \mathcal{Q}$. Let the components of the vector $U = (u_1, u_2, \ldots, u_n)$ be i.i.d. with the probability p to be one. Let $\nu = p/(1-p)$ denote the ratio of the expected number of ones to the expected number of zeros, and $w(U)$ denote the weight $w(U) = \sum_{i=1}^{n} u_i$. Then the probability distribution of U is

$$p(U) = p^{w(U)}(1-p)^{n-w(U)} = \nu^{w(U)}/(1+\nu)^n$$

and so its entropy is:

$$\begin{aligned} \mathcal{H}(U) &= -\sum_{U\in\mathcal{Q}} \frac{\nu^{w(U)}}{(1+\nu)^n} \log \frac{\nu^{w(U)}}{(1+\nu)^n} && (15)\\ &= \frac{1}{(1+\nu)^n}\left[n\sum_{U\in\mathcal{Q}} \nu^{w(U)} \log(1+\nu) - \sum_{U\in\mathcal{Q}} w(U)\nu^{w(U)} \log\nu \right]. && (16) \end{aligned}$$

Since $p(U)$ is a probability distribution, we have

$$\sum_{U\in\mathcal{Q}} \nu^{w(U)} = (1+\nu)^n$$

and, deriving the above with respect to ν,

$$\sum_{U\in\mathcal{Q}} w(U)\nu^{w(U)} = n\nu(1+\nu)^{n-1} \tag{17}$$

so

$$\mathcal{H}(U) = n\left(\log(1+\nu) - \frac{\nu\log\nu}{1+\nu}\right).$$

Letting $p = h^{-1}(\alpha)$ results to entropy rate, $\mathcal{H}(U) = \alpha\, n$. Our coding scheme is described next.

First the initial state of the bus S_0 is assumed to be uniformly distributed in $\mathcal{Q}$ with i.i.d. components and independent of the outcomes of source U. Let $\epsilon > 0$ be given, then at each time t, as many as $\alpha\, n - \epsilon$ bits can be mapped in average into outcomes of U. Such

an encoding scheme exists by the Shannon source coding theorem. Subsequently, if U_t is the outcome of the source at time t and S_{t-1} is the state of bus at time $t-1$, then we set the state of bus at time t to be $S_t = S_{t-1} \oplus U_t$.

Because the original state of the bus is uniformly distributed, it can be seen that at each time t, the state of the bus is also uniformly distributed in $\mathcal{Q}$. This means that we can apply Lemma 2.1 and observe that the expected energy at time t assuming that input U_t is chosen is given by

$$\overline{\mathcal{E}_{S_t}(S_t \rightarrow S_t \oplus U_t)} = 2w(U_t)\mathcal{E}_u, \tag{18}$$

where the subscript indicates that the expectation is taken over with respect to S_t. Recall that $p(U_t) = \nu^{w(Ut)}/(1+\nu)^n$ and so, taking expectation over S_t and U in the above gives:

$$\begin{aligned} \overline{\mathcal{E}(S_t \rightarrow S_t \oplus U_t)} &= 2\mathcal{E}_u \sum_{u \in \mathcal{Q}} w(u)p(u) & (19) \\ &= 2n\mathcal{E}_u \frac{\nu}{1+\nu} = 2n\mathcal{E}_u h^{-1}(\alpha) & (20) \end{aligned}$$

where we have used expression (17). Thus the expected energy per bit of the bus, at utilization arbitrarily close to α, is arbitrarily close to $\frac{2h^{-1}(\alpha)}{\alpha}\mathcal{E}_u$. This means that the pair $\left(\alpha, \frac{2h^{-1}(\alpha)}{\alpha}\mathcal{E}_u\right)$ is achievable. We conclude that $\mathcal{E}_b(\alpha) \leq \frac{2h^{-1}(\alpha)}{\alpha}\mathcal{E}_u$. □

The bound of Theorem 2.1 is depicted in Figure 1. The ratio $\mathcal{E}_b(\alpha)/\mathcal{E}_u$ is plotted on the vertical axis. As we can see, we can obtain significant reduction in energy consumption if we transmit data at a rate less than the maximum.

We next show that when the number of lines n is large it is in fact possible to achieve the above limit using uniform input distribution and by using low Hamming weight codewords. To this end, we recall the following lemma of [8].

Lemma 2.2 *Let $0 \leq \nu \leq \frac{1}{2}$. Then for any n it is :*

$$\frac{2^{nh(\nu)}}{\sqrt{8n\nu(1-\nu)}} \leq \sum_{j=0}^{\lfloor \nu n \rfloor} \frac{n!}{j!(n-j)!} \leq 2^{nh(\nu)},$$

where $h(\cdot)$ is the binary entropy function.

Proof: We refer the reader to Corollary 9, Chapter 10, Section 11 of [8] (page 310). □

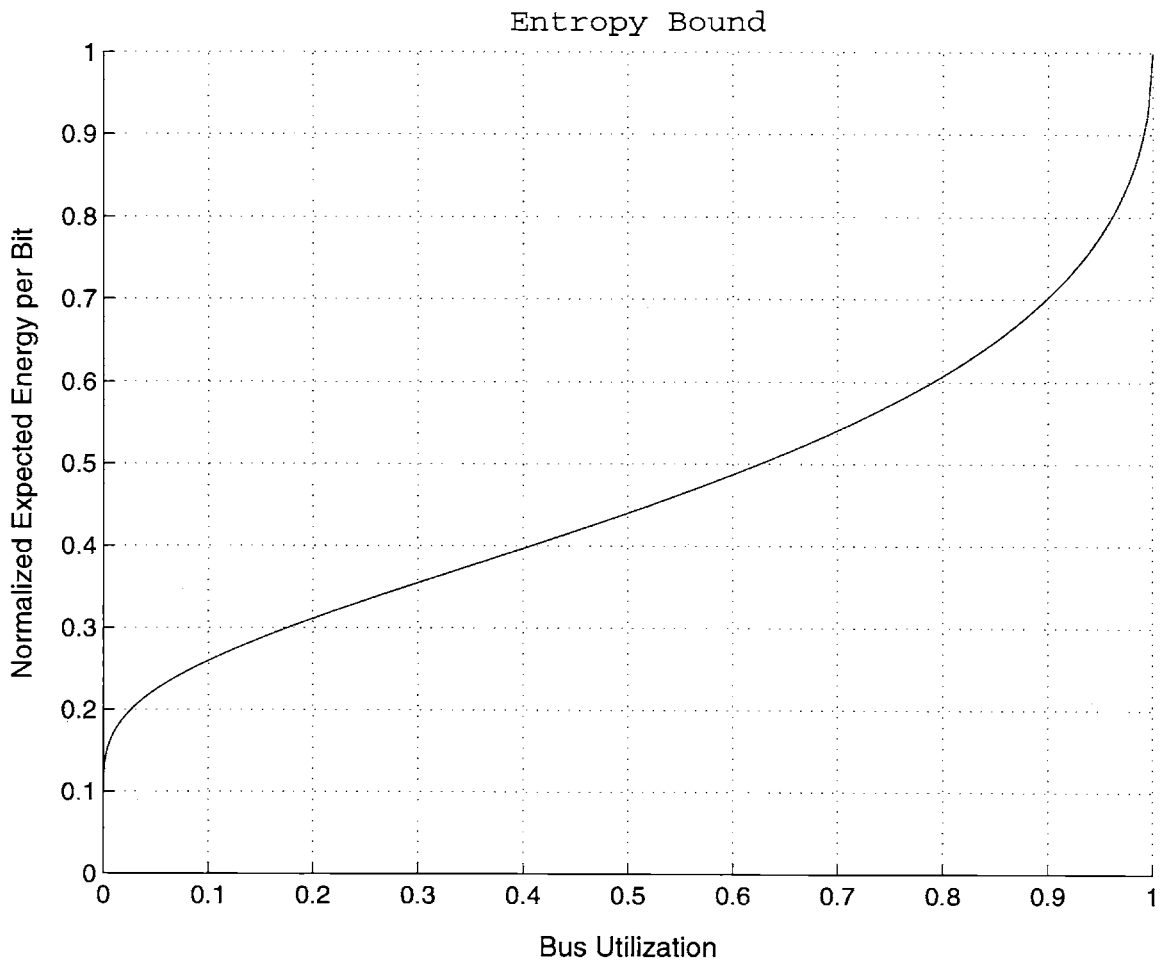

Figure 1: The Entropy Bound

Suppose that $0 \leq \alpha \leq 1$ is given. Now let $\nu = h^{-1}(\alpha)$ and consider the set $\mathcal{Q}_*$ of all elements in $\mathcal{Q} = \{0,1\}^n$ having Hamming weight less than or equal to $n\nu$. By the above Lemma, the cardinality $|\mathcal{Q}_*|$ of $\mathcal{Q}_*$ satisfies :

$$\frac{2^{nh(\nu)}}{\sqrt{8n\nu(1-\nu)}} \leq |\mathcal{Q}_*| \leq 2^{nh(\nu)}.$$

For large n, it can be observed from the above that $\log(|\mathcal{Q}_*|)/n \simeq h(\nu) = \alpha$. The approximation approaches the exact value as $n \to \infty$. We now state a second coding scheme.

Theorem 2.2 *The bound of Theorem 2.1 can be approached using a uniform input distribution and differential encoding.*

Proof: Let the initial state of the bus S_0 be uniformly distributed in $\mathcal{Q}$. Also, let U be an

information source producing i.i.d. outcomes that are uniformly distributed in the set $\mathcal{Q}_*$ that was defined above. The entropy rate of the source is of course $\log(|\mathcal{Q}_*|)$. Now suppose that the state of the bus at time $t-1$ is S_{t-1}. If U_t is the outcome of U at time t we set, $S_t = S_{t-1} \oplus U_t$. Since S_0 is uniformly distributed in $\mathcal{Q}$ it can be seen that at each time t the state of the bus is also uniformly distributed in $\mathcal{Q}$. This means that we can apply Lemma 2.1 and so the expected energy at time t is given by :

$$\overline{\mathcal{E}_{S_t}(S_t \rightarrow S_t \oplus U_t)} = 2w(U_t)\mathcal{E}_u, \tag{21}$$

where the subscript indicates that the expectation is taken over with respect to S_t. Taking expectation over S_t and U in the above gives:

$$\overline{\mathcal{E}(S_t \rightarrow S_t \oplus U_t)} \leq 2n\nu\mathcal{E}_u = 2nh^{-1}(\alpha)\mathcal{E}_u \tag{22}$$

Note that as n approaches infinity, and because of Lemma 2.2, the entropy rate $\log(|\mathcal{Q}_*|)$ of the source U asymptotically becomes equal to αn. Thus, as $n \rightarrow \infty$ the utilization approaches α while the expected energy per bit remains less than or equal to $\frac{2nh^{-1}(\alpha)\mathcal{E}_u}{\alpha n} = \frac{2h^{-1}(\alpha)}{\alpha}\mathcal{E}_u$ □

For $\lambda = 0$ (the LC case) the bound of Theorem 2.1 was also established by Ramprasad, Shanbhag and Hajj [13]. Our result though, accounts for $\lambda > 0$ as well, that is, for the case of modern technologies where there is energy coupling between different lines of the bus, the DSM case.

3 Coding Theorems From Stationary Ergodic Processes

The simple bound that was established in the previous section assumed a basic differential encoding. In that case the state of the bus at each time t depended only on the state at time $t-1$. This introduced the natural question : Is it possible to achieve more energy reduction using encoders with higher order memory ? We answer the question affirmatively by presenting an example. The example also motivates a systematic study of the problem. Moreover, *the method developed here applies to other computational models as well.*

To proceed it is important to simplify the notation. From now on we will identify the vector elements $(0,0,\ldots,0),(1,0,\ldots,0),\ldots,(1,1,\ldots,1)$ of $\mathcal{Q}$ (and states of the bus) with

the numbers $0, 1, \ldots, 2^n - 1$ respectively. In this notation, $\mathcal{E}(i \to j)$ is the energy required for the transition from state i to state j of the bus.

Theorem 3.1 *Let $\mathcal{M}$ be a Markov source with the 2^n states, $0, 1, \ldots, 2^n - 1$. Let $\nu > 0$ and suppose that the probability of transition from state i to state j is given by :*

$$\Pr(j \mid i) = \frac{\exp(-\nu \mathcal{E}(i \to j))}{\sum_{k=0}^{2^n-1} \exp(-\nu \mathcal{E}(i \to k))}. \tag{23}$$

Let $p(i), i = 0, 1, \ldots, 2^n - 1$ and $\mathcal{H}(\mathcal{M})$ denote the steady state distribution and entropy rate of $\mathcal{M}$ respectively. Then the limiting expected energy per bit, $\mathcal{E}_b$, at utilization $\mathcal{H}(\mathcal{M})/n$ of the bus, satisfies the inequality :

$$\mathcal{E}_b(\frac{\mathcal{H}(\mathcal{M})}{n}) \leq \frac{1}{\nu} \left[1 - \frac{\sum_{j=0}^{2^n-1} p(j) \log \left(\sum_{r=0}^{2^n-1} \exp(-\nu \mathcal{E}(j \to r)) \right)}{\mathcal{H}(\mathcal{M})} \right] \tag{24}$$

Proof: First note that by definition the Markov process $\mathcal{M}$ is irreducible and aperiodic, therefore, the stationary distribution $p(i)$ exists and is unique. Also note that the transition energy, expression (1), is symmetric with respect to the starting end ending states, that is, $\mathcal{E}(i \to j) = \mathcal{E}(j \to i)$ for all i, j. This allows us to write $p(i)$ explicitly as :

$$p(i) = \frac{\sum_{k=0}^{2^n-1} \exp(-\nu \mathcal{E}(i \to k))}{\sum_{j=1}^{2^n-1} \sum_{k=0}^{2^n-1} \exp(-\nu \mathcal{E}(j \to k)).} \tag{25}$$

In the cases where the energy cost function is not symmetric, $p(i)$ can be computed using standard methods. We endow the process $\mathcal{M}$ with the steady state distribution. The stationary Markov process $\mathcal{M}$ is irreducible, aperiodic and therefore ergodic. By Shannon-McMillan-Breiman Theorem, for every $\epsilon > 0$, there exists $L_0 > 0$ such that for every $L \geq L_0$, there exists a set

$$\mathcal{T}(L) = \left\{ (\sigma_1^{\mathbf{i}}, \ldots, \sigma_L^{\mathbf{i}}),\ \mathbf{i} = 1, 2, \ldots, |\mathcal{T}(L)| \right\} \tag{26}$$

of typical sequences, of bus states $\sigma_j^{\mathbf{i}}$, of length L such that :

$$2^{L(\mathcal{H}(\mathcal{M})-\epsilon)} \leq |\mathcal{T}(L)| \leq 2^{L(\mathcal{H}(\mathcal{M})+\epsilon)} \tag{27}$$

and

$$\mathcal{H}(\mathcal{M}) - \epsilon \leq \frac{-\log \Pr(\sigma_1^{\mathbf{i}}, \ldots, \sigma_L^{\mathbf{i}})}{L} \leq \mathcal{H}(\mathcal{M}) + \epsilon \tag{28}$$

for every $\mathbf{i} = 1, 2, \ldots, |\mathcal{T}(L)|$. We take the set of typical sequences $\mathcal{T}(L)$ as our bus code, that is $\mathcal{C}(L) = \mathcal{T}(L)$, and we choose the elements of $\mathcal{C}(L)$ with equal probability. Then the utilization of the bus using code $\mathcal{C}(L)$ is :

$$\frac{\mathcal{H}(\mathcal{M})}{n} - \frac{\epsilon}{n} \leq \alpha \leq \frac{\mathcal{H}(\mathcal{M})}{n} + \frac{\epsilon}{n}. \tag{29}$$

Now, writing $\Pr(\sigma_1^{\mathbf{i}}, \ldots, \sigma_L^{\mathbf{i}}) = p(\sigma_1^{\mathbf{i}}) \prod_{k=2}^{L} \Pr(\sigma_k^{\mathbf{i}} \mid \sigma_{k-1}^{\mathbf{i}})$ and replacing it into (28) we get that :

$$\mathcal{H}(\mathcal{M}) - \epsilon + \frac{\log p(\sigma_1^{\mathbf{i}})}{L} \leq -\frac{1}{L} \sum_{k=2}^{L} \log \Pr(\sigma_k^{\mathbf{i}} \mid \sigma_{k-1}^{\mathbf{i}}) \leq \mathcal{H}(\mathcal{M}) + \epsilon + \frac{\log p(\sigma_1^{\mathbf{i}})}{L} \tag{30}$$

Replacing expression (23) into the above inequalities we get :

$$\mathcal{H}(\mathcal{M}) - \epsilon + \frac{\log p(\sigma_1^{\mathbf{i}})}{L} \leq \frac{1}{L} \sum_{k=2}^{L} \nu \mathcal{E}(\sigma_{k-1}^{\mathbf{i}} \to \sigma_k^{\mathbf{i}}) + \sum_{j=0}^{2^n-1} \tilde{p}^{\mathbf{i}}(j) \log(\sum_{k=0}^{2^n-1} \exp(-\nu \mathcal{E}(j \to k))) \tag{31}$$

$$\leq \mathcal{H}(\mathcal{M}) + \epsilon + \frac{\log p(\sigma_1^{\mathbf{i}})}{L}$$

where we have set, $\tilde{p}^{\mathbf{i}}(j) = \frac{N^{\mathbf{i}}(j)}{L}$, and $N^{\mathbf{i}}(j)$ is the number of occurrences of state j in the sequence $\sigma_2^{\mathbf{i}}, \sigma_3^{\mathbf{i}}, \ldots, \sigma_L^{\mathbf{i}}$. By summing up the above inequalities over $\mathbf{i} = 1, 2, \ldots, |\mathcal{T}(L)|$ and dividing the sum by $|\mathcal{T}(L)|$ we obtain :

$$\mathcal{H}(\mathcal{M}) - \epsilon + \frac{1}{|\mathcal{T}(L)|} \sum_{\mathbf{i}=1}^{|\mathcal{T}(L)|} \frac{\log p(\sigma_1^{\mathbf{i}})}{L} \leq$$

$$\nu \frac{1}{L} \sum_{\mathbf{i}=1}^{|\mathcal{T}(L)|} \frac{1}{|\mathcal{T}(L)|} \left(\sum_{k=2}^{L} \mathcal{E}(\sigma_{k-1}^{\mathbf{i}} \to \sigma_k^{\mathbf{i}}) \right) + \frac{1}{|\mathcal{T}(L)|} \sum_{\mathbf{i}=1}^{|\mathcal{T}(L)|} \sum_{j=0}^{2^n-1} \tilde{p}^{\mathbf{i}}(j) \log(\sum_{k=0}^{2^n-1} \exp(-\nu \mathcal{E}(j \to k)))$$

$$\leq \mathcal{H}(\mathcal{M}) + \epsilon + \frac{1}{|\mathcal{T}(L)|} \sum_{\mathbf{i}=1}^{|\mathcal{T}(L)|} \frac{\log p(\sigma_1^{\mathbf{i}})}{L}$$

Using Equation (6), the first summand in the middle term of the above inequality can be written as

$$\frac{1}{L} \sum_{\mathbf{i}=1}^{|\mathcal{T}(L)|} \frac{1}{|\mathcal{T}(L)|} \left(\sum_{k=2}^{L} \mathcal{E}(\sigma_{k-1}^{\mathbf{i}} \to \sigma_k^{\mathbf{i}}) \right) = \mathcal{E}_{av}(\mathcal{C}(L)) - \frac{e(\mathcal{C})}{L}$$

where $0 \leq e(\mathcal{C}) \leq \mathcal{E}_{\max}$. Therefore we have :

$$\mathcal{H}(\mathcal{M}) - \epsilon + \frac{1}{|\mathcal{T}(L)|} \sum_{\mathbf{i}=1}^{|\mathcal{T}(L)|} \frac{\log p(\sigma_1^{\mathbf{i}})}{L} \leq$$

$$\nu\mathcal{E}_{av}(\mathcal{C}(L)) - \nu\frac{e(\mathcal{C})}{L} + \frac{1}{|\mathcal{T}(L)|}\sum_{\mathbf{i}=1}^{|\mathcal{T}(L)|}\sum_{j=0}^{2^n-1}\tilde{p}^{\mathbf{i}}(j)\log(\sum_{k=0}^{2^n-1}\exp(-\nu\mathcal{E}(j\rightarrow k)))$$

$$\leq \mathcal{H}(\mathcal{M}) + \epsilon + \frac{1}{|\mathcal{T}(L)|}\sum_{\mathbf{i}=1}^{|\mathcal{T}(L)|}\frac{\log p(\sigma_1^{\mathbf{i}})}{L}$$

The above result holds for every $L \geq L_0$. Also $\mathcal{E}(\mathcal{C}(L)) \leq \mathcal{E}_{\max} < \infty$. Letting $L \rightarrow \infty$ and by using strong typicality, [4], we have $\tilde{p}^{\mathbf{i}}(j) \rightarrow p(j)$ with probability one. Moreover $\nu\frac{e(\mathcal{C})}{L} \rightarrow 0$. Thus for sufficiently large L, we have

$$\mathcal{H}(\mathcal{M}) - 2\epsilon \leq \nu\mathcal{E}_{av}(\mathcal{C}(L)) + \sum_{j=0}^{2^n-1}p(j)\log(\sum_{k=0}^{2^n-1}\exp(-\nu\mathcal{E}(j\rightarrow k))) \leq \mathcal{H}(\mathcal{M}) + 2\epsilon \qquad (32)$$

with probability one. Since ϵ is arbitrary, we can see from the above that an expected energy consumption arbitrary close to $\frac{1}{\nu}[\mathcal{H}(\mathcal{M}) - \sum_{j=0}^{2^n-1}p(j)\log(\sum_{k=0}^{2^n-1}\exp(-\nu\mathcal{E}(j\rightarrow k)))]$ is achievable at utilizations arbitrary close to $\alpha = \mathcal{H}(\mathcal{M})/n$. Thus

$$\left(\frac{\mathcal{H}(\mathcal{M})}{n}, \frac{1}{\nu}\left[\mathcal{H}(\mathcal{M}) - \sum_{j=0}^{2^n-1}p(j)\log(\sum_{k=0}^{2^n-1}\exp(-\nu\mathcal{E}(j\rightarrow k)))\right]\right)$$

is an achievable pair for every $\nu \geq 0$ since elements of the code-book with arbitrarily small probability do not affect the energy consumption. □

The bound of Theorem 3.1 is referred to as the *exponential bound* throughout this chapter. We have plotted the results of Theorem 3.1 in Figures 2, 3 and 4 for small values of n and with the vertical axis labelled by $\mathcal{E}_b(\alpha)/\mathcal{E}_u$.

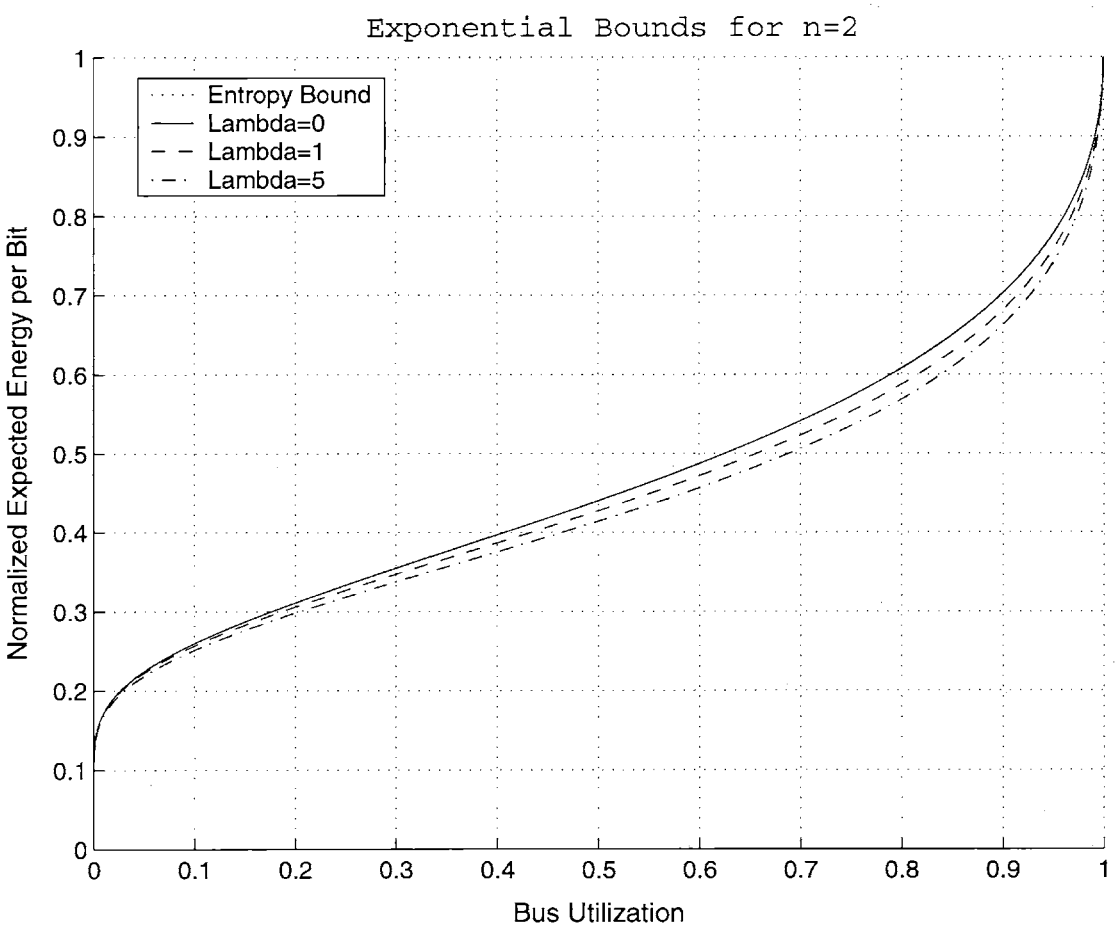

Figure 2: The Exponential Bound for $n = 2$ and $\lambda = 0, 1, 5$. For $\lambda = 0$ the exponential and entropy bounds coincide.

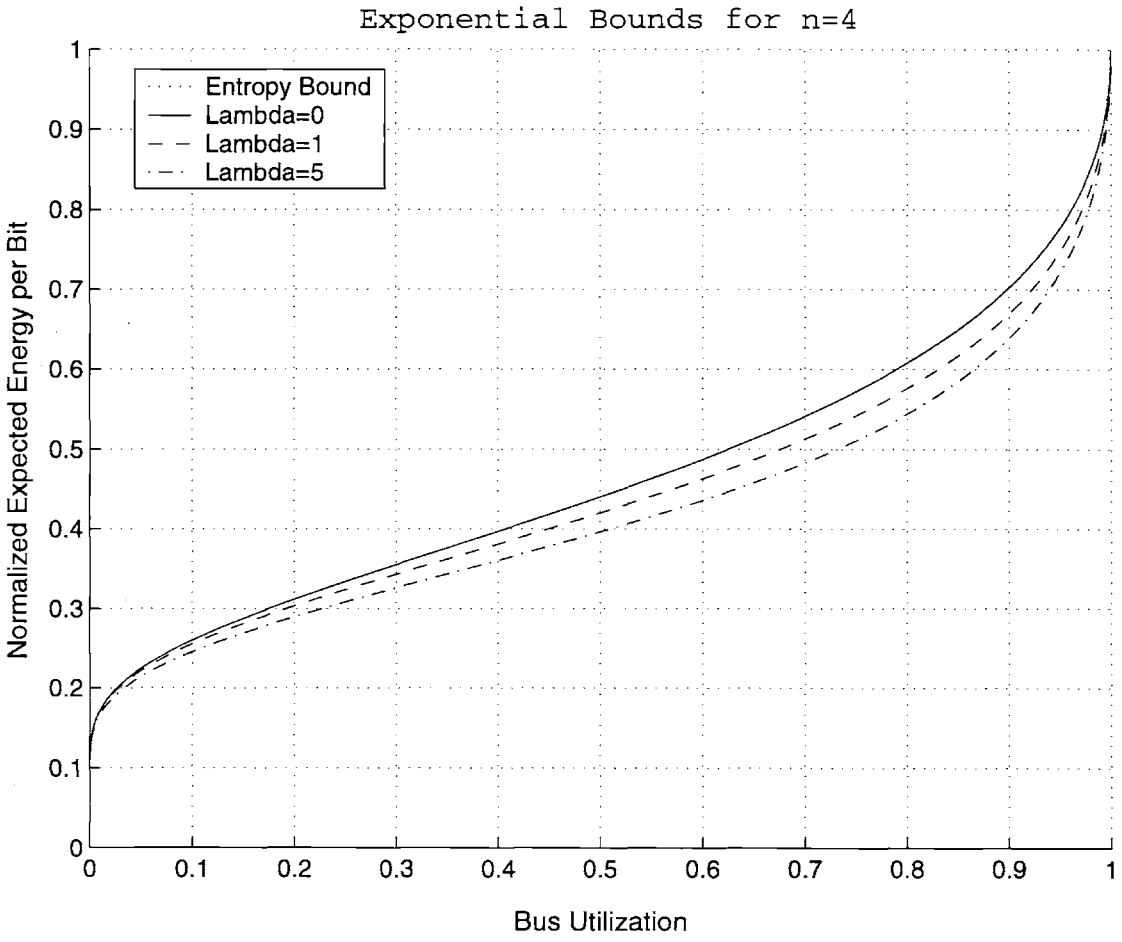

Figure 3: The Exponential Bound for $n = 4$ and $\lambda = 0, 1, 5$. For $\lambda = 0$ the exponential and entropy bounds coincide.

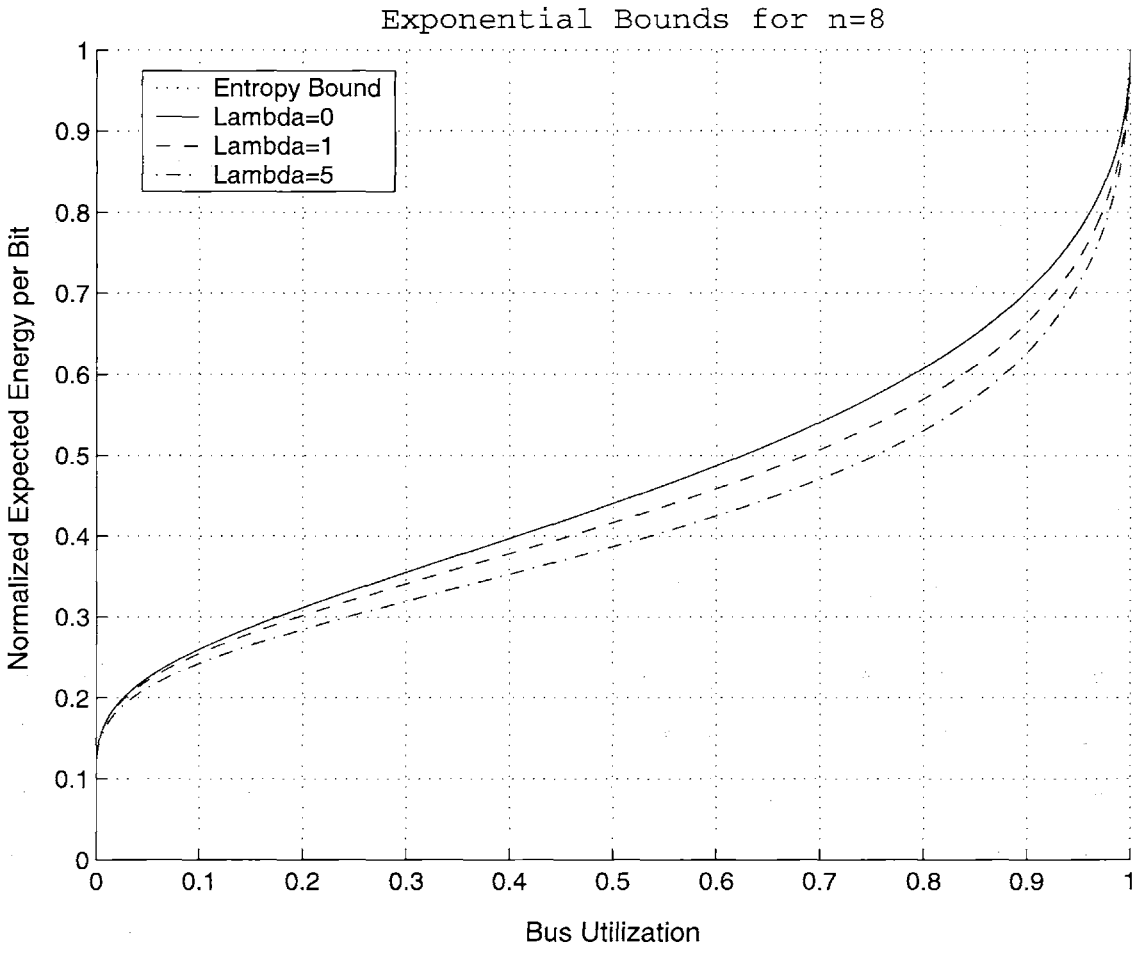

Figure 4: The Exponential Bound for $n = 8$ and $\lambda = 0, 1, 5$. For $\lambda = 0$ the exponential and entropy bounds coincide.

The bound is hard to compute for higher values of n as there are too many states of the Markov Chain $\mathcal{M}$. This makes the computation of the steady state distribution numerically difficult for $n > 8$. For comparison, we also plotted the bound of Theorem 2.1. We see in figures 2, 3 and 4 that the bound of Theorem 3.1 is always better than the bound of Theorem 2.1 for all the plotted values. Moreover, it is seen in the figures, and it is easy to prove, that for $\lambda = 0$, the exponential bound and the entropy bound coincide.

Motivated by the above result, it is natural to ask if we can use other stationary ergodic processes, invoke the Shannon-McMillan-Breiman (reference [4]) Theorem and obtain stronger results. This motivates our studies in the next Lemma.

Construction I: Let $\mathcal{X}$ denote a stationary ergodic stochastic process whose outcomes are elements of $\mathcal{Q} = \{0,1\}^n$. We consider the set $\mathcal{X}_L$ consisting of sequences $(\sigma_1, \sigma_2, \ldots, \sigma_{L-1}, \sigma_L)$ of L successive outcomes of the stationary ergodic stochastic process $\mathcal{X}$. It is of course $\mathcal{X}_L \subset \mathcal{Q}^L$. Given $\epsilon > 0$ we also consider the subset $\mathcal{T}_L(\mathcal{X})$ of $\mathcal{X}_L$ that contains all ϵ-typical sequences. That is, all $(\sigma_1, \sigma_2, \ldots, \sigma_{L-1}, \sigma_L)$ such that :

$$\mathcal{H}(\mathcal{X}) - \epsilon \leq -\frac{1}{L} \log \Pr(\sigma_1, \sigma_2, \ldots, \sigma_{L-1}, \sigma_L) \leq \mathcal{H}(\mathcal{X}) + \epsilon.$$

These two sets of sequences define respectively two codes $\mathcal{X}(L) = \mathcal{X}_L$ and $\mathcal{C}(L) = \mathcal{T}_L$ of length L. The codeword $(\sigma_1, \sigma_2, \ldots, \sigma_L)$ in each code is chosen with probability equal to $c(L) \Pr(\sigma_1, \sigma_2, \ldots, \sigma_L)$. The scalar $c(L)$ is 1 for $\mathcal{X}(L)$ and an appropriate normalization constant for $\mathcal{C}(L)$. As $L \to \infty$, we know from the Shannon-McMillan-Breiman Theorem that $\Pr(\mathcal{T}_L(\mathcal{X})) \to 1$ and $c(L) \to 1$. This implies the following Lemma.

Lemma 3.1 *Let a stationary ergodic process $\mathcal{X}$ be given. As $L \to \infty$, the expected energy per bit $\mathcal{E}_b(\mathcal{C}(L))$ of the code $\mathcal{C}(L)$ consisting of the ϵ-typical sequences $\mathcal{T}_L(\mathcal{X})$ becomes equal to the expected energy per bit $\mathcal{E}_b(\mathcal{X}(L))$ of the code $\mathcal{X}(L)$ containing all length L sequences $\sigma_1, \sigma_2, \ldots, \sigma_{L-1}, \sigma_L$ of successive outcomes of the stationary stochastic process $\mathcal{X}$. The same is true for the utilizations of the two codes.*

Proof: Let $\mathcal{E}_{\max} = \max_{\sigma,\omega \in \mathbb{Z}_2^n} \mathcal{E}(\sigma \to \omega)$. As $L \to \infty$ the set of all the codewords in $\mathcal{X}(L)$ that are not in $\mathcal{C}(L)$ has an arbitrary small probability δ. This means that they contribute at most $\delta \mathcal{E}_{\max}$ to the expected energy per bus transition. This contribution can be made as

small as desired. Since $c(L) \rightarrow 1$ as $L \rightarrow \infty$ the utilization afforded by $\mathcal{C}(L)$ and $\mathcal{X}(L)$ get arbitrary close to each other as $L \rightarrow \infty$. Combining these two observations yields the result. □

Definition 3.1 *Let $\mathcal{X}$ be a stationary (not necessarily ergodic) process in $\mathcal{Q}$. The utilization, expected energy consumption and the expected energy consumption per bit of process $\mathcal{X}$ are defined respectively as :*

$$\alpha = \mathcal{H}(\mathcal{X})/n \ , \tag{33}$$

$$\begin{aligned} \mathcal{E}_{av}(\mathcal{X}) &= \sum_{\sigma \in \mathcal{Q}} \sum_{\omega \in \mathcal{Q}} \Pr(X_1 = \sigma, X_2 = \omega)\, \mathcal{E}(\sigma \rightarrow \omega) \\ &= \overline{\mathcal{E}(\sigma \rightarrow \omega)} \end{aligned} \tag{34}$$

and

$$\mathcal{E}_b(\mathcal{X}) = \frac{\mathcal{E}_{av}(\mathcal{X})}{\mathcal{H}(\mathcal{X})} \ . \tag{35}$$

The following two definitions are analogous to that based on sequences of codes.

Definition 3.2 *We define a pair (α, β), where α denotes the utilization of the bus and β the expected energy per bit, to be achievable by a class $\mathbf{X}$ of stationary processes in $\mathcal{Q}$, if and only if there exists an infinite sequence of processes $\mathcal{X}_i$ in $\mathbf{X}$ (not necessarily distinct) such that, $\mathcal{H}(\mathcal{X}_i) \rightarrow \alpha_1 n$ for some $\alpha_1 \geq \alpha$ and $\mathcal{E}_b(\mathcal{X}_i) \rightarrow \beta_1$ for some $\beta_1 \leq \beta$.*

Definition 3.3 *We define the limiting expected energy consumption per bit, $\mathcal{E}_b$, at utilization $\alpha \in [0, 1]$ of a class $\mathbf{X}$ of stationary processes, to be the function :*

$$\mathcal{E}_b(\alpha, \mathbf{X}) = \inf \ \{\beta \mid \text{The pair } (\alpha, \beta) \text{ is achievable by the class of processes } \mathbf{X}\}. \tag{36}$$

We are now in the position to prove the following important Theorem.

Theorem 3.2 *Let $\mathcal{X}$ be a stationary (not necessarily ergodic) process in $\mathcal{Q}$, of utilization α and expected energy per bit, $\mathcal{E}_b(\mathcal{X})$. There exists an ergodic Markov process, the typical sequences of which can form a sequence of codes that achieves the pair $(\alpha, \mathcal{E}_b(\mathcal{X}))$.*

Proof: Let $\mathcal{X}$ be a stationary stochastic process with outcomes $X_1, X_2, X_3, \ldots$ at time $t = 1, 2, 3, \ldots$ respectively and with probability distribution $Pr(X_1, X_2, \ldots, X_k)$ for $k = 1, 2, \ldots$. We can assume that $\Pr(X_1 = \sigma) > 0$ for every $\sigma \in \mathcal{Q}$, otherwise we can remove σ from the set of states of $\mathcal{X}$. Let $\mathcal{M}$ be the stationary Markov process with stationary distribution $p(\sigma) = \Pr(X_1 = \sigma)$ and transition probabilities $\Pr(\omega \mid \sigma) = \Pr(X_2 = \omega \mid X_1 = \sigma)$. We first observe that

$$\mathcal{H}(\mathcal{X}) = \lim_{k \to \infty} \mathcal{H}(X_k \mid X_{k-1}, X_{k-2}, \ldots, X_1) \leq \mathcal{H}(X_2 \mid X_1) = \mathcal{H}(\mathcal{M}). \tag{37}$$

Suppose for the moment that $\mathcal{M}$ is an ergodic process. Let $\epsilon > 0$ and $\mathcal{M}(L)$ be the set of all ϵ-typical sequences of length L of the process $\mathcal{M}$. We can conclude from an application of Shannon-McMillan-Breiman that $\mathcal{M}(L)$ has at least $2^{L(\mathcal{H}(\mathcal{M})-\epsilon)}$ codewords (for L sufficiently large). Therefore, choosing ϵ sufficiently small, the utilization of the bus by the code $\mathcal{M}(L)$ can be as close to $\mathcal{H}(\mathcal{M})/n$ as desired. Note that $\mathcal{H}(\mathcal{M})/n \geq \mathcal{H}(\mathcal{X})/n$. Moreover, the ergodicity of $\mathcal{M}$ along with Lemma 3.1 imply that the expected energy consumption $\mathcal{E}_{av}(\mathcal{M}(L))$, of the code $\mathcal{M}(L)$, approaches the expected energy consumption $\mathcal{E}_{av}(\mathcal{M})$ of the process $\mathcal{M}$ as $L \to \infty$. By the definition of process $\mathcal{M}$ it is also true that $\mathcal{E}_{av}(\mathcal{M}) = \mathcal{E}_{av}(\mathcal{X})$ where

$$\mathcal{E}_{av}(\mathcal{X}) = \sum_{\sigma \in \mathcal{Q}} \sum_{\omega \in \mathcal{Q}} \Pr(X(0) = \sigma, X(1) = \omega) \mathcal{E}(\sigma \to \omega).$$

Thus, using typical sequences of the ergodic Markov process $\mathcal{M}$ we can construct a family of codes $\mathcal{M}(L)$ with $\liminf_{L \to \infty} \mathcal{H}(\mathcal{M}(L)) \geq \mathcal{H}(\mathcal{X})$ and $\lim_{L \to \infty} \mathcal{E}_{av}(\mathcal{M}(L)) = \mathcal{E}_{av}(\mathcal{X})$. This concludes the theorem in the case that process $\mathcal{M}$ is ergodic.

It remains to treat the case when the process $\mathcal{M}$ is not ergodic. Consider the matrix $\mathbf{P}$ with i, j-th element $\mathbf{P}_{ij} = \Pr(X(2) = j \mid X(1) = i)$, that is the transition probability matrix of the Markov process $\mathcal{M}$. Then we claim that some of the elements of $\mathbf{P}$ are zeroes. If not, $\mathbf{P}$ is an irreducible matrix and $\mathcal{M}$ is an aperiodic irreducible Markov chain and hence is ergodic. Consider $\mathbf{q}$ to be the row matrix whose i-th element is $\Pr(X(1) = i)$. Then, $\mathbf{qP} = \mathbf{P}$ and all the elements of $\mathbf{q}$ are positive. (This is true because we have assumed that process $\mathcal{X}$ visits all of its states and so process $\mathcal{M}$ is irreducible). For every $0 \leq \delta \leq 1$, we set $\mathbf{P}_\delta = (1 - \delta)\mathbf{P} + \delta \mathbf{1}^T \mathbf{q}$, where $\mathbf{1}$ is a row matrix with all of its elements being 1. We also have $\mathbf{qP}_\delta = \mathbf{q}$ for every $\delta \geq 0$.

We now consider the Markov source $\mathcal{M}_\delta$ whose stationary distribution is $\mathbf{q}$ and whose state transition matrix is $\mathbf{P}_\delta$. For every $\delta > 0$, the matrix $\mathbf{P}_\delta$ has only positive elements and thus $\mathcal{M}_\delta$ is a stationary irreducible ergodic Markov process. By the continuity of the entropy, the entropy of $\mathcal{M}_\delta$ can be made arbitrary close to the entropy of $\mathcal{M}$. Because $\mathbf{P}_\delta$ can get as close to $\mathbf{P}$ as desired the expected energy consumption for typical sequences of $\mathcal{M}_\delta$ approaches that of $\mathcal{X}$. Thus, as $\delta \to 0$, we can get arbitrary close to a superior or at least equivalent trade-off between energy and utilization than that of the stationary process $\mathcal{X}$. $\square$

The above result motivates us to ask the following question : Given a (utilization, expected energy consumption per bit) pair, (α, β), achievable by a sequence of codes (Definition 2.6), is it possible to construct a sequence of codes, using only typical sequences of ergodic Markov sources, that achieves (α, β) ?

Our goal is to answer this question affirmatively. To this end, we start by introducing the following construction that is in some way the inverse of that used before. Here we start with a given code and construct a stationary process of at least as high utilization and the same expected energy consumption with that of the code.

Construction II: Let $\mathcal{C}$ denote a code of length L and utilization α. We construct a stationary stochastic process $\mathcal{X}(\mathcal{C})$ from $\mathcal{C}$. To this end, we will first construct an interim stochastic process $\mathcal{Y} : Y_1, Y_2, Y_3, \ldots$ by describing the joint distribution of the indexed random variables $Y_1, Y_2, Y_3, \ldots$.

Let $\mathbf{Y}_1, \mathbf{Y}_2, \mathbf{Y}_3, \ldots$ denote the sequence of random vectors of length L with $\mathbf{Y}_k = (Y_{(k-1)L+1}, Y_{(k-1)L+2}, \ldots, Y_{kL})$ for $k = 1, 2, \ldots$. We define the random vectors $\mathbf{Y}_1, \mathbf{Y}_2, \mathbf{Y}_3, \ldots$ to be mutually independent and such that for every $\mathbf{c} \in \mathcal{Q}^L$ it is :

$$\Pr(\mathbf{Y}_k = \mathbf{c}) = \begin{cases} \Pr(\mathbf{c}) & \text{if } \mathbf{c} \in \mathcal{C} \\ 0 & \text{otherwise} \end{cases}$$

These define the interim stochastic process $\mathcal{Y}$ completely. Now for $l = 1, 2, \ldots$ we define a sequence of distributions (on the random variables $X_1, X_2, \ldots$)by letting

$$\Pr\left((X_1 = x_1, X_2 = x_2, \ldots, X_l = x_l)\right) = \frac{1}{L} \sum_{i=0}^{L-1} \Pr\left((Y_{1+i} = x_1, Y_{2+i} = x_2, \ldots, Y_{l+i} = x_l)\right).$$

Note that the RHS of the expression is a shift-and-average operation. This sequence of distributions is *consistent* in the sense that

$$\Pr\left((X_1 = x_1, X_2 = x_2, X_3 = x_3, \ldots, X_{l-1} = x_{l-1})\right) = \ldots$$
$$\ldots = \sum_{x_l \in \mathcal{Q}} \Pr\left((X_1 = x_1, X_2 = x_2, X_3 = x_3, \ldots, X_l = x_l)\right)$$

A Theorem of Kolmogorov (page 21 of [23]) implies the existence of a random process $\mathcal{X}(\mathcal{C}) : X_1, X_2, X_3, \ldots$ characterized by the family of the above distributions. Next we prove that the process $\mathcal{X}(\mathcal{C})$ is stationary. To this end, we first observe that the process $\mathcal{Y}$ is cyclo-stationary with period L in the sense that

$$\Pr\left(Y_1, Y_2, Y_3, \ldots, Y_l\right) = \Pr\left(Y_{1+mL}, Y_{2+mL}, Y_{3+mL}, \ldots, Y_{l+mL}\right)$$

for all $l = 1, 2, \ldots$ and $m = 0, 1, \ldots$. If $L = 1$ then $\mathbf{Y}_k = (Y_k)$ for all $k = 1, 2, \ldots$ and so processes $\mathcal{Y}$ and $\mathcal{X}$ are stationary processes. Suppose now that $L > 1$ and $j \geq 1$, $l \geq 1$ are given. Then :

$$\Pr(X_{1+j} = y_1, \ldots, X_{l+j} = y_l) = \sum_{x_1, x_2, \ldots, x_j} \Pr(X_1 = x_1, \ldots, X_j = x_j, X_{1+j} = y_1, \ldots, X_{l+j} = y_l)$$
$$= \sum_{x_1, x_2, \ldots, x_j} \frac{1}{L} \sum_{i=0}^{L-1} \Pr(Y_{1+i} = x_1, \ldots, Y_{j+i} = x_j, Y_{1+j+i} = y_1, Y_{2+j+i} = y_2, \ldots, Y_{l+j+i} = y_l)$$
$$= \frac{1}{L} \sum_{i=0}^{L-1} \sum_{x_1, x_2, \ldots, x_j} \Pr(Y_{1+i} = x_1, \ldots, Y_{j+i} = x_j, Y_{1+j+i} = y_1, Y_{2+j+i} = y_2, \ldots, Y_{l+j+i} = y_l)$$
$$= \frac{1}{L} \sum_{i=0}^{L-1} \Pr(Y_{1+j+i} = y_1, Y_{2+j+i} = y_2, \ldots, Y_{l+j+i} = y_l).$$

As i runs in the set $0, 1, 2 \ldots, L-1$, the value $i + j \mod(L)$ also runs in the same set. Combining this observation and the cyclo-stationarity of the process $\mathcal{Y}$, we conclude that

$$\frac{1}{L} \sum_{i=0}^{L-1} \Pr\left(Y_{1+j+i} = y_1, Y_{2+j+i} = y_2, \ldots, Y_{l+j+i} = y_l\right) = \frac{1}{L} \sum_{i=0}^{L-1} \Pr\left(Y_{1+i} = y_1, \ldots, Y_{l+i} = y_l\right)$$
$$= \Pr\left(X_1 = y_1, X_2 = y_2, X_3 = y_3, \ldots, X_l = y_l\right)$$

Thus the process $\mathcal{X}(\mathcal{C})$ is stationary.

Lemma 3.2 *The entropy rate $\mathcal{H}(\mathcal{X}(\mathcal{C}))$ of the random process $\mathcal{X}(\mathcal{C})$ is greater than or equal to the entropy $\mathcal{H}(\mathcal{C}) = -\frac{1}{L} \sum_{\mathbf{c} \in \mathcal{C}} \Pr(\mathbf{c}) \log \Pr(\mathbf{c})$ of the code $\mathcal{C}$.*

Proof: By the definition of the process $\mathcal{X}(\mathcal{C})$, it is :

$$\Pr(X_1, X_2, \ldots, X_l) = \sum_{i=0}^{L-1} \frac{1}{L} \Pr(Y_{1+i}, Y_{2+i}, \ldots, Y_{l+i}),$$

Since the entropy function is concave we can deduce that, for every $l = 1, 2, \ldots$, it is :

$$\mathcal{H}(X_1, X_2, \ldots, X_l) \geq \frac{1}{L} \sum_{i=0}^{L-1} \mathcal{H}(Y_{1+i}, Y_{2+i}, \ldots, Y_{l+i}).$$

Dividing by l and letting $l \to \infty$ we obtain $\mathcal{H}(\mathcal{X}) \geq \mathcal{H}(\mathcal{Y})$. But

$$\begin{aligned}
\mathcal{H}(\mathcal{Y}) &= \lim_{k\to\infty} \frac{\mathcal{H}(Y_1, Y_2, \ldots, Y_k)}{k} \\
&= \lim_{k\to\infty} \frac{\mathcal{H}(Y_1, Y_2, \ldots, Y_{Lk})}{kL} \\
&= \lim_{k\to\infty} \sum_{r=1}^{k} \frac{\mathcal{H}(Y_{(r-1)L+1}, Y_{(r-1)L+2}, \ldots, Y_{rL})}{kL} \\
&= \frac{1}{L} H(Y_1, Y_2 \ldots, Y_L)
\end{aligned}$$

where we have used the cyclo-stationarity of $\mathcal{Y}$ and the independence of the random vectors $\mathbf{Y}_1, \mathbf{Y}_2, \mathbf{Y}_3, \ldots$. By Definition 2.2 it is :

$$\frac{1}{L} H(Y_1, \ldots, Y_L) = \mathcal{H}(\mathcal{C}).$$

This concludes the proof. □

Lemma 3.3 *The expected energy consumption $\mathcal{E}_{av}(\mathcal{X})$ of the process $\mathcal{X}(\mathcal{C})$ equals the expected energy consumption $\mathcal{E}_{av}(\mathcal{C})$ of the code $\mathcal{C}$.*

Proof: The energy consumption of a sequence $\sigma_1, \sigma_2, \ldots, \sigma_l$ of elements in $\mathcal{Q}$ is, (Definition 3.1),

$$\mathcal{E}(\sigma_1, \sigma_2, \ldots, \sigma_l) = \sum_{i=1}^{l-1} \mathcal{E}(\sigma_i \to \sigma_{i+1}).$$

The expected energy consumption of the stationary process $\mathcal{X}(\mathcal{C})$ is given by :

$$\begin{aligned}
\mathcal{E}_{av}(\mathcal{X}) &= \overline{\mathcal{E}(\sigma \to \omega)} \\
&= \lim_{l\to\infty} \frac{\overline{\mathcal{E}(\sigma_1, \sigma_2, \ldots, \sigma_l)}}{l-1},
\end{aligned}$$

where the expectation $\overline{\mathcal{E}(\sigma_1, \sigma_2, \ldots, \sigma_l)}$ is taken with respect to the probability distribution $\Pr(X_1 = \sigma_1, \ldots, X_l = \sigma_l)$. However,

$$\Pr(X_1, X_2, \ldots, X_l) = \sum_{i=0}^{L-1} \frac{1}{L} \Pr(Y_{1+i}, Y_{2+i}, \ldots, Y_{l+i}),$$

thus

$$\overline{\mathcal{E}(\sigma_1, \sigma_2, \ldots, \sigma_l)} = \frac{1}{L} \sum_{i=0}^{L-1} \sum_{\sigma_1, \ldots, \sigma_l} \mathcal{E}(\sigma_1, \sigma_2, \ldots, \sigma_l) \Pr(Y_{1+i} = \sigma_1, \ldots, Y_{l+i} = \sigma_l).$$

Now, for $i = 0, 1, 2, \ldots, L-1$, we set $l_i = L - i + 1$ and $l^i = L\lfloor \frac{l}{L} \rfloor - i$. We can write :

$$\mathcal{E}(\sigma_1, \sigma_2, \ldots, \sigma_l) = \mathcal{E}(\sigma_1, \sigma_2, \ldots, \sigma_{l_i}) + \mathcal{E}(\sigma_{l_i}, \sigma_{l_i+1}, \ldots, \sigma_{l^i}) + \mathcal{E}(\sigma_{l^i}, \sigma_{l^i+1}, \ldots, \sigma_l)$$

where we agree that the expected energy of any trivial 1-element sequence is zero, i.e. $\mathcal{E}(\sigma) = 0$ for every $\sigma \in \mathcal{Q}$. Then we have

$$\mathcal{E}(\sigma_{l_i}, \sigma_{l_i+1}, \ldots \sigma_{l^i}) \leq \mathcal{E}(\sigma_1, \sigma_2, \ldots, \sigma_l) \leq \mathcal{E}(\sigma_{l_i}, \sigma_{l_i+1}, \ldots, \sigma_{l^i}) + 3L\mathcal{E}_{\max}. \tag{38}$$

where $\mathcal{E}_{\max} = \max_{\sigma, \omega \in \mathcal{Q}} \mathcal{E}(\sigma \to \omega)$. We conclude that :

$$\begin{aligned}
\overline{\mathcal{E}(\sigma_1, \sigma_2, \ldots, \sigma_l)} &= \frac{1}{L} \sum_{i=0}^{L-1} \sum_{\sigma_1, \ldots, \sigma_l} \mathcal{E}(\sigma_1, \sigma_2, \ldots, \sigma_l) \Pr(Y_{1+i} = \sigma_1, \ldots, Y_{l+i} = \sigma_l) \\
&\leq \frac{1}{L} \sum_{i=0}^{L-1} \sum_{\sigma_1, \ldots, \sigma_l} (\mathcal{E}(\sigma_{l_i}, \ldots, \sigma_{l^i}) + 3L\mathcal{E}_{\max}) \Pr(Y_{1+i} = \sigma_1, \ldots, Y_{l+i} = \sigma_l) \\
&\leq \left[\frac{1}{L} \sum_{i=0}^{L-1} \sum_{\sigma_{l_i}, \ldots, \sigma_{l^i}} \mathcal{E}(\sigma_{l_i}, \ldots, \sigma_{l^i}) \Pr(Y_{l_i+i} = \sigma_{l_i}, \ldots, Y_{l^i+i} = \sigma_{l^i}) \right] + 3L\mathcal{E}_{\max}. \qquad (39)
\end{aligned}$$

However,

$$\begin{aligned}
\Pr(Y_{l_i+i} = \sigma_{l_i}, \ldots, Y_{l^i+i} = \sigma_{l^i}) &= \Pr(Y_{L+1} = \sigma_{l_i}, \ldots, Y_{L\lfloor \frac{l}{L} \rfloor} = \sigma_{l^i}) \\
&= \Pr\left((\mathbf{Y}_2, \mathbf{Y}_3, \ldots, \mathbf{Y}_{\lfloor \frac{l}{L} \rfloor}) = (\sigma_{l_i}, \ldots, \sigma_{l^i}) \right)
\end{aligned}$$

Since the random vectors $\mathbf{Y}_2, \mathbf{Y}_3, \ldots, \mathbf{Y}_{\lfloor \frac{l}{L} \rfloor}$ are independent, we can write :

$$\Pr(Y_{l_i+i} = \sigma_{l_i}, \ldots, Y_{l^i+i} = \sigma_{l^i}) =$$

$$\Pr(\mathbf{Y}_2 = (\sigma_{l_i}, \ldots, \sigma_{l_i+L-1})) \cdot \Pr(\mathbf{Y}_3 = (\sigma_{l_i+L}, \ldots, \sigma_{l_i+2L-1})) \cdot \ldots \cdot \Pr(\mathbf{Y}_{\lfloor \frac{l}{L} \rfloor} = (\sigma_{l^i-L+1}, \ldots, \sigma_{l^i}))$$

In addition we can write :

$$\begin{aligned}\mathcal{E}(\sigma_{l_i},\ldots,\sigma_{l^i}) &= \mathcal{E}(\sigma_{l_i},\ldots,\sigma_{l_i+L-1})+\mathcal{E}(\sigma_{l_i+L-1}\rightarrow\sigma_{l_i+L})+\\ &+\mathcal{E}(\sigma_{l_i+L},\ldots,\sigma_{l_i+2L-1})+\mathcal{E}(\sigma_{l_i+2L-1}\rightarrow\sigma_{l_i+2L})+\\ &+\ldots+\mathcal{E}(\sigma_{l^i-L+1},\ldots,\sigma_{l^i})\end{aligned}$$

The expected energy of the code $\mathcal{C}$ is by its definition 2.3,

$$\begin{aligned}\mathcal{E}_{av}(\mathcal{C}) &= \frac{1}{L}\sum_{\sigma_{l_i},\ldots,\sigma_{l_i+L-1}}\mathcal{E}(\sigma_{l_i},\ldots,\sigma_{l_i+L-1})\Pr(\mathbf{Y}_2=(\sigma_{l_i},\ldots,\sigma_{l_i+L-1}))+\\ &+\frac{1}{L}\sum_{\sigma_{l_i},\ldots,\sigma_{l_i+2L-1}}Pr(\mathbf{Y}_2=(\sigma_{l_i},\ldots,\sigma_{l_i+L-1}))\cdot\\ &\cdot Pr(\mathbf{Y}_3=(\sigma_{l_i+L},\ldots,\sigma_{l_i+2L-1}))\,\mathcal{E}(\sigma_{l_i+L-1}\rightarrow\sigma_{l_i+L})\end{aligned}$$

Using the cyclo-stationarity of $\mathcal{Y}$ and the expressions above, we can arrive at :

$$\sum_{\sigma_{l_i},\ldots,\sigma_{l^i}}\mathcal{E}(\sigma_{l_i},\ldots,\sigma_{l^i})\Pr(Y_{l_i+i}=\sigma_{l_i},\ldots,Y_{l^i+i}=\sigma_{l^i})=L\left(\lfloor\frac{l}{L}\rfloor-1\right)\mathcal{E}_{av}(\mathcal{C})-\frac{e(\mathcal{C})}{L}.$$

where $e(\mathcal{C})$ is as in Equation (6). Finally, replacing the above into inequality (39), we get :

$$\overline{\mathcal{E}(\sigma_1,\sigma_2,\ldots,\sigma_l)}\leq l\cdot\mathcal{E}_{av}(\mathcal{C})+3L\mathcal{E}_{\max}. \tag{40}$$

Similarly we can show the inequality :

$$L\left(\lfloor\frac{l}{L}\rfloor-1\right)\mathcal{E}_{av}(\mathcal{C})-\mathcal{E}_{\max}\leq\overline{\mathcal{E}(\sigma_1,\sigma_2,\ldots,\sigma_l)} \tag{41}$$

Combining Inequalities (40) and (41) and dividing by l we have

$$\frac{L}{l}\left(\lfloor\frac{l}{L}\rfloor-1\right)\mathcal{E}_{av}(\mathcal{C})-\frac{\mathcal{E}_{\max}}{l}\leq\frac{\overline{\mathcal{E}(\sigma_1,\sigma_2,\ldots,\sigma_l)}}{l}\leq\mathcal{E}_{av}(\mathcal{C})+\frac{3L\mathcal{E}_{\max}}{l} \tag{42}$$

Letting $l\rightarrow\infty$, we arrive at

$$\mathcal{E}_{av}(\mathcal{C})=\mathcal{E}_{av}(\mathcal{X}) \tag{43}$$

□

Theorem 3.3 *Suppose that the pair (α,β), of utilization and expected energy per bit, is achievable according to definition 2.3. Then, there exists a family of codes, of strictly increasing lengths and constructed from typical sequences of Stationary Markov Ergodic processes, that achieves (α,β).*

Proof: By assumption, there exists a sequence of codes $\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3, \ldots$ of strictly increasing length which utilize the bus arbitrarily close to a number greater or equal to α and their expected energy consumptions per bit $\mathcal{E}_b(\mathcal{C}_i), i = 1, 2, \ldots$ gets arbitrary close to a number smaller or equal to β.

We apply Construction II and get a sequence of stationary processes $\mathcal{X}_i = \mathcal{X}(\mathcal{C}_i), i = 1, 2, \ldots$. By Lemma 3.2, the value $\mathcal{H}(\mathcal{X}_i)/n$ is greater or equal to the utilization of bus using the code $\mathcal{C}_i$. By Lemma 3.3, the expected energy consumption $\mathcal{E}_{av}(\mathcal{X}_i)$ equals $\mathcal{E}_{av}(\mathcal{C}_i)$. Thus as $i \to \infty$, $\mathcal{E}_b(\mathcal{X}_i)$ approaches a number smaller or equal to $\mathcal{E}_b(\mathcal{C}_i)$. By Theorem 3.2, there exist stationary ergodic Markov processes $\mathcal{M}_i, i = 1, 2, \ldots$ whose typical sequences utilize the bus at least as well as $\mathcal{H}(\mathcal{X}_i)/n$ and have expected energy consumption per bit less or equal to $\mathcal{E}_b(\mathcal{X}_i)$. Combining these observations we conclude the proof. □

Corollary 3.1 *For every $\alpha \in (0, 1]$, the limiting expected energy consumption per bit $\mathcal{E}_b(\alpha)$ equals $\mathcal{E}_b(\alpha, \mathcal{M}_\mathbf{e})$, where $\mathcal{M}_\mathbf{e}$ is the set of all stationary ergodic Markov processes in $\mathcal{Q}$.*

Proof: The proof follows from Theorems 3.3, Lemma 3.1 and the definition of $\mathcal{E}_b(\alpha)$. □

We can now prove the following important Theorem.

Theorem 3.4 *For every $\alpha \in (0, 1]$, the limiting expected energy consumption per bit $\mathcal{E}_b(\alpha)$ equals $\mathcal{E}_b(\alpha, \mathcal{M})$, where $\mathcal{M}$ is the set of all stationary Markov processes in $\mathcal{Q}$.*

Proof: The set of stationary Markov processes, $\mathcal{M}$, has the set of ergodic Markov stationary processes, $\mathcal{M}_\mathbf{e}$, as a subset and is a subset of the set of stationary processes.

Thus stationary Markov processes must do at least as well as stationary ergodic Markov processes and at most as well as stationary processes when considering the trade-off between utilization and expected energy consumption per bit.

However, we have already proven in Theorem 3.2 that stationary ergodic Markov sources achieve any trade-off achievable by stationary processes. This means that Stationary Markov processes provide the same trade-off between utilization and energy consumption as that of the stationary processes and stationary ergodic Markov processes. We now apply Corollary 3.1. □

4 Computation of The Function $\mathcal{E}_b(\alpha)$

Theorem 3.4 provides us with the computational arsenal to compute $\mathcal{E}_b(\alpha)$ by restricting our attention to the class of stationary ergodic Markov sources. To be able to use this arsenal, we will need to study the function $\mathcal{E}_b(\alpha)$ in more detail. We start with the following lemma.

Lemma 4.1 *For every $\alpha \in (0,1]$ there exists a sequence of codes $\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3, \ldots$ whose bus utilizations tend to α and whose expected energies per bit tend to $\mathcal{E}_b(\alpha)$.*

Proof: By definition of the limiting expected energy per bit, function $\mathcal{E}_b(\alpha)$, there exists a sequence of achievable pairs (α_i, β_i) such that $\liminf_{i\to\infty} \alpha_i \geq \alpha$ and $\beta_i \to \mathcal{E}_b(\alpha)$ as $i \to \infty$. Since α_i, $i = 1, 2, \ldots$ belong to a compact set, there exists a subsequence of achievable pairs $(\hat{\alpha}_i, \hat{\beta}_i)$ such that $\hat{\alpha}_i \to \hat{\alpha}$ and $\hat{\beta}_i \to \mathcal{E}_b(\alpha)$ for some $\hat{\alpha} \geq \alpha$. By an application of a Cantor's diagonalization argument and the definition of achievable pairs we conclude that there exists a sequence of codes $\hat{\mathcal{C}}_1, \hat{\mathcal{C}}_2, \hat{\mathcal{C}}_3, \ldots$ of strictly increasing lengths, whose bus utilizations tend to $\hat{\alpha}$ and whose expected energies per bit tend to $\mathcal{E}_b(\alpha)$.

Consider the following zero padding construction: Suppose that a code $\mathcal{C}$ of length L, utilization α and expected energy per bit $\mathcal{E}_b(\mathcal{C})$ is given. Then by expanding its codewords by l zero states of value zero, we obtain a new code $\mathcal{C}^l$ of length $L+l$, utilization $\alpha\frac{L}{L+l}$ and energy per bit $\mathcal{E}_b(\mathcal{C}^l)$ with $\frac{L+l}{L+l-1}\left[\mathcal{E}_b(\mathcal{C}) - \frac{\mathcal{E}_{max}}{\alpha L}\right] \leq \mathcal{E}_b(\mathcal{C}^l) \leq \frac{L+l}{L+l-1}\left[\mathcal{E}_b(\mathcal{C}) + \frac{\mathcal{E}_{max}}{\alpha L}\right]$.

By applying this zero padding technique to all the codes in the sequence $\hat{\mathcal{C}}_1, \hat{\mathcal{C}}_2, \hat{\mathcal{C}}_3, \ldots$, we can construct a new sequence of codes $\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3, \ldots$ whose utilization tends to α and their energy per bit tends to $\mathcal{E}_b(\alpha)$. □

Lemma 4.2 *The function $\mathcal{E}_b(\alpha)$ is continuous and non-decreasing for $0 < \alpha \leq 1$.*

Proof: The non-decreasing property of $\mathcal{E}_b(\alpha)$ follows directly its definition 2.7. We define the function :

$$\mathcal{E}_{av}(\alpha) = n\alpha\mathcal{E}_b(\alpha),$$

and claim that it is continuous. To prove the claim, let $\alpha_1, \alpha_2 \in (0,1]$ and $\nu \in (0,1)$ be given. By the definition of achievability and Lemma 4.1 there exist two sequences of

codes, $\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3, \ldots$ and $\mathcal{C}_1^*, \mathcal{C}_2^*, \mathcal{C}_3^*, \ldots$, of strictly increasing lengths, whose utilizations tend to α_1 and α_2 respectively and whose expected energy per bit tend to $\mathcal{E}_b(\alpha_1)$ and $\mathcal{E}_b(\alpha_2)$ respectively. This means that $\mathcal{E}_{av}(\mathcal{C}_i)$ tends to $n\alpha_1\mathcal{E}_b(\alpha_1) = \mathcal{E}_{av}(\alpha_1)$ and $\mathcal{E}_{av}(\mathcal{C}_i^*)$ tends to $n\alpha_2\mathcal{E}_b(\alpha_2) = \mathcal{E}_{av}(\alpha_2)$. We now use time sharing between $\mathcal{C}_i$ and $\mathcal{C}_i^*$ with shares of ν and $1-\nu$ by first using $\mathcal{C}_i$ and then $\mathcal{C}_i^*$ to obtain a code $\nu\mathcal{C}_i \oplus (1-\nu)\mathcal{C}_i^*$. The sequence of codes $\nu\mathcal{C}_i \oplus (1-\nu)\mathcal{C}_i^*$ have utilizations that tend to $\nu\alpha_1 + (1-\nu)\alpha_2$ and expected energies per bit $\mathcal{E}_b(\nu\mathcal{C}_i \oplus (1-\nu)\mathcal{C}_i^*)$ that tend to $\frac{\nu\alpha_1\mathcal{E}_b(\alpha_1)+(1-\nu)\alpha_2\mathcal{E}_b(\alpha_2)}{(\nu\alpha_1+(1-\nu)\alpha_2)}$. Thus

$$\mathcal{E}_b(\nu\alpha_1 + (1-\nu)\alpha_2) \leq \frac{\nu\alpha_1\mathcal{E}_b(\alpha_1) + (1-\nu)\alpha_2\mathcal{E}_b(\alpha_2)}{(\nu\alpha_1 + (1-\nu)\alpha_2)}.$$

This gives the convexity of $\mathcal{E}_{av}(\alpha) = n\alpha\mathcal{E}_b(\alpha)$. This convexity implies the continuity of $\mathcal{E}_{ave}(\alpha)$ and in turn the continuity of $\mathcal{E}_b(\alpha) = \frac{\mathcal{E}_{av}(\alpha)}{n\alpha}$. □

For $\alpha \in (0,1]$ let $\mathbf{M}^*(\alpha)$ denote the set of all stationary Markov processes $\mathcal{M}$ whose states are elements of $\mathcal{Q}$ and have entropy rate $\mathcal{H}(\mathcal{M}) \geq \alpha n$.

Lemma 4.3 *For every $\alpha \in (0,1]$ it is :*

$$\mathcal{E}_b(\alpha) = \inf_{\mathcal{M}\in\mathbf{M}^*(\alpha)} \mathcal{E}_b(\mathcal{M}). \tag{44}$$

Proof: Theorem 3.4 implies that

$$\mathcal{E}_b(\alpha) \leq \inf_{\mathcal{M}\in\mathbf{M}^*(\alpha)} \mathcal{E}_b(\mathcal{M}),$$

as the infimum is taken over a smaller set. Now let $\epsilon > 0$ be given. Applying Theorem 3.4 and the definition of achievability we observe that

$$\inf_{\mathcal{M}\in\mathbf{M}^*(\alpha)} \mathcal{E}_b(\mathcal{M}) \leq \mathcal{E}_b(\alpha+\epsilon).$$

This is true since all stationary Markov processes that are candidates for achieving the pair $(\alpha+\epsilon, \mathcal{E}_b(\alpha+\epsilon))$ must have entropy rate higher than αn. Since $\epsilon > 0$ is arbitrary and $\mathcal{E}_b(\cdot)$ is continuous, we conclude that

$$\inf_{\mathcal{M}\in\mathbf{M}^*(\alpha)} \mathcal{E}_b(\mathcal{M}) \leq \mathcal{E}_b(\alpha),$$

and the Lemma is proven. □

Every pair, (P, q), of a $2^n \times 2^n = |\mathcal{Q}| \times |\mathcal{Q}|$ transition matrix P and a 1×2^n state probability row vector q, satisfying $qP = q$, defines a stationary Markov process $\mathcal{M}$. Conversely, every stationary Markov process $\mathcal{M}$ defines, in the obvious way, a pair, (P, q), of a transition probability matrix P and a row probability vector q such that $qP = q$. The expected energy consumption per bit and the entropy rate of $\mathcal{M}$ can be computed using the elements of P and q.

From now on we will mix the notation by identifying the process $\mathcal{M}$ with the pair (P, q). Moreover, we introduce the notation $\mathcal{H}(P, q)$ for $\mathcal{H}(\mathcal{M})$, $\mathcal{E}_{av}(P, q)$ for $\mathcal{E}_{av}(\mathcal{M})$ and $\mathcal{E}_b(P, q)$ for $\mathcal{E}_b(\mathcal{M})$. Note that the function

$$\mathcal{E}_b(\mathcal{M}) = \mathcal{E}_b(P, q) = \frac{\mathcal{E}_{av}(P, q)}{\mathcal{H}(P, q)}$$

is continuous with respect to the elements of the pair (P, q).

Finally, we can interpret the set $\mathbf{M}^*(\alpha)$ as the set of all pairs (P, q) with $\mathcal{H}(P, q) \geq \alpha n$. This leads to the following result.

Lemma 4.4 *For every $\alpha \in (0, 1]$ we have :*

$$\mathcal{E}_b(\alpha) = \min_{(P,q) \in \mathbf{M}^*(\alpha)} \mathcal{E}_b(P, q). \tag{45}$$

Proof: The function $\mathcal{E}_b(P, q)$ is continuous in the pair (P, q) and the constraints of the problem define a compact set. We thus conclude that infimum is achieved. □

Now we can state and prove the following Theorem that will be important in the calculation of the limiting energy per bit.

Theorem 4.1 *For every $\alpha \in (0, 1]$ it is :*

$$\mathcal{E}_b(\alpha) = \min_{(P,q) \,:\, \mathcal{H}(P,q) = \alpha n} \frac{\mathcal{E}_{av}(P, q)}{\alpha n}. \tag{46}$$

Proof: To prove the Theorem we will construct a minimizing pair (P, q) for expression (45) in Lemma 4.4 such that $\mathcal{H}(P, q) = \alpha n$. Once this is established, the result follows.

To this end, suppose that (P_*, q_*) is a minimizing pair for (45) with $\mathcal{H}(P_*, q_*) > \alpha n$. For $\theta \in [0, 1]$ we define the pair

$$\begin{aligned} (P_\theta, q_*) &= \theta(P_*, q_*) + (1 - \theta)(I, q_*) \\ &= (\theta P_* + (1 - \theta) I \,,\, q_*) \end{aligned}$$

where I is the $2^n \times 2^n$ identity matrix. The new pair satisfies the equality $q_* P_\theta = q_*$. Also, we have :

$$\begin{aligned} \mathcal{E}_{av}(P_\theta, q_*) &= \mathcal{E}_{av}(\theta P_* + (1-\theta) I\ ,\ q_*) \\ &= \theta \mathcal{E}_{av}(P_*, q_*) + (1-\theta)\mathcal{E}_{av}(I, q_*) \\ &= \theta \mathcal{E}_{av}(P_*, q_*) + 0 \end{aligned}$$

The concavity of the function $\mathcal{H}(\cdot, q_*)$ implies that :

$$\begin{aligned} \mathcal{H}(P_\theta, q_*) &= \mathcal{H}(\theta P_* + (1-\theta) I\ ,\ q_*) \\ &\geq \theta \mathcal{H}(P_*, q_*) + (1-\theta)\mathcal{H}(I, q_*) \\ &= \theta \mathcal{H}(P_*, q_*) + 0. \end{aligned}$$

Finally,

$$\begin{aligned} \mathcal{E}_b(P_\theta, q_*) &= \frac{\mathcal{E}_{av}(P_\theta, q_*)}{\mathcal{H}(P_\theta, q_*)} \\ &\leq \frac{\theta \mathcal{E}_{av}(P_*, q_*)}{\theta \mathcal{H}(P_*, q_*)} = \mathcal{E}_b(P_*, q_*) \end{aligned}$$

Since $\mathcal{H}(P_0, q_*) = \mathcal{H}(I, q_*) = 0$, there exists some θ' such that $\mathcal{H}(P_{\theta'}, q_*) = \alpha n$. We conclude that $(P_{\theta'}, q_*)$ is also a minimizing pair of (45) with $\mathcal{H}(P_{\theta'}, q_*) = \alpha n$. $\square$

In light of Theorem 4.1, the computation of $\mathcal{E}_b(\alpha)$ has been reduced to a constraint minimization problem that can be solved using standard numerical methods such as the application of Lagrange multipliers.

In the following we will prove that for every $\alpha \in (0,1)$ there is a stationary Markov process that achieves the minimum in the aforementioned problems. We will derive formulas for the limiting expected energy as a function of the utilization.

Let (P, q) be a pair of a stochastic matrix and its probability eigenvector. The pair defines the matrix $\Pi = [\pi_{i,j}]_{i,j=0}^{2^n-1}$ with $\pi_{i,j} = q_i P_{i,j}$. For every i, j the entries $\pi_{i,j}$ satisfy the following relations,

$$\pi_{i,j} \geq 0 \tag{47}$$

$$\sum_{i,j} \pi_{i,j} = 1 \tag{48}$$

and

$$\sum_j \pi_{i,j} = \sum_k \pi_{k,i} \tag{49}$$

The last equality holds because $\sum_j \pi_{i,j} = \sum_j q_i P_{i,j} = q_i = \sum_k q_k P_{k,i} = \sum_k \pi_{k,i}$ and can be written also as $\sum_j (\pi_{i,j} - \pi_{j,i}) = 0$. The entropy of the matrix Π is defined as :

$$\mathcal{H}(\Pi) = -\sum_{i,j} \pi_{i,j} \log(\frac{\pi_{i,j}}{\sum_k \pi_{i,k}}) \tag{50}$$

and of course equals $\mathcal{H}(P,q)$. (We agree that $0 \log \frac{0}{0} = 0$ and a zero row of matrix Π contributes zero to the entropy). Now, independently of the pair (P,q), we define $\mathbf{\Pi}$ to be the set of all $2^n \times 2^n$ matrices Π satisfying the constraints (47), (48) and (49). We also define $\mathbf{\Pi}_+$ be the subset of $\mathbf{\Pi}$ consisting of only the positive matrices. We have the lemma :

Lemma 4.5 *The mapping* $\Pi \longmapsto P = \left[\frac{\pi_{i,j}}{\sum_k \pi_{i,k}}\right]_{i,j}$, *from* $\mathbf{\Pi}_+$ *to the subset of positive stochastic matrices, is bijective.*

Proof: Suppose that the positive stochastic matrix P is the image of an element $\Pi' = [\pi'_{i,j}]_{i,j}$ of $\mathbf{\Pi}_+$. Then, there exists a unique positive vector $q' = [q'_i]_i$ such that $\pi'_{i,j} = q'_i P_{i,j}$. It is straight forward to verify that $q'_i = \sum_k \pi'_{i,k}$. To show that the mapping $\Pi \longmapsto P$ is injective it suffices to show that there is only one positive vector q for which $[q_i P_{i,j}]_{i,j}$ belongs to $\mathbf{\Pi}_+$. For vector q, Property (49) implies that,

$$\sum_i q_i P_{i,j} = \sum_k q_j P_{j,k} = q_j \tag{51}$$

Therefore q must be a positive left eigenvector of P. The matrix P is positive and by Perron's theorem, q is unique up to a positive factor. The vector q is uniquely defined because of Property (48) which implies, $\sum_i q_i = \sum_{i,j} q_i P_{i,j} = 1$.

To show the mapping is onto, we start with a positive stochastic matrix P and set $\pi_{i,j} = v_i P_{i,j}$, where v is the unique (left) probability eigenvector of P. Then it is $\sum_{i,j} \pi_{i,j} = 1$ and $\sum_k \pi_{k,i} = \sum_k v_k P_{k,i} = v_i = \sum_j v_i P_{i,j} = \sum_j \pi_{i,j}$. Therefore Π belongs to $\mathbf{\Pi}_+$. □

In the following we will write $\Pi = (P,q)$ to denote that $\pi_{i,j} = q_i P_{i,j}$. Note that for a

matrix $\Pi \in \mathbf{\Pi}$ with a zero row, there is a set of pairs (P, q) such that $\Pi = (P, q)$. All of these pairs have the same vector q, the entries of which correspond to zero rows of Π, must be zero.

Lemma 4.6 *The set* $\mathbf{\Pi}$ *is convex. The entropy function* $\mathcal{H}$ *is concave in* $\mathbf{\Pi}$ *and strictly concave in the subset* $\mathbf{\Pi}_+$.

Proof: Conditions (47), (48) and (49) directly imply the convexity of the set $\mathbf{\Pi}$. Now let Π^1 and Π^2 be in $\mathbf{\Pi}$ and a be a constant such that $0 < a < 1$. Set $\Pi = a\Pi^1 + (1-a)\Pi^2$. The *log-sum* inequality states that :

$$a\,\pi^1_{i,j} \log\left(\frac{a\,\pi^1_{i,j}}{a\sum_k \pi^1_{i,k}}\right) + (1-a)\pi^2_{i,j} \log\left(\frac{(1-a)\pi^2_{i,j}}{(1-a)\sum_k \pi^2_{i,k}}\right) \geq$$
$$\geq \left(a\,\pi^1_{i,j} + (1-a)\pi^2_{i,j}\right) \log\left(\frac{a\,\pi^1_{i,j} + (1-a)\pi^2_{i,j}}{a\sum_k \pi^1_{i,k} + (1-a)\sum_k \pi^2_{i,k}}\right)$$

with equality if and only if

$$\frac{a\,\pi^1_{i,j}}{a\sum_k \pi^1_{i,k}} = \frac{(1-a)\pi^2_{i,j}}{(1-a)\sum_k \pi^2_{i,k}}$$

Therefore we have :

$$\begin{aligned}
\mathcal{H}(\Pi) &= -\sum_{i,j}\left(a\pi^1_{i,j} + (1-a)\pi^2_{i,j}\right) \log\left(\frac{a\pi^1_{i,j} + (1-a)\pi^2_{i,j}}{a\sum_k \pi^1_{i,k} + (1-a)\sum_k \pi^2_{i,k}}\right) \\
&\geq -a\sum_{i,j} \pi^1_{i,j} \log\left(\frac{\pi^1_{i,j}}{\sum_k \pi^1_{i,k}}\right) - (1-a)\sum_{i,j} \pi^2_{i,j} \log\left(\frac{\pi^2_{i,j}}{\sum_k \pi^2_{i,k}}\right) \\
&= a\mathcal{H}(\Pi^1) + (1-a)\mathcal{H}(\Pi^2)
\end{aligned}$$

and so $\mathcal{H}(\Pi) \geq a\mathcal{H}(\Pi^1) + (1-a)\mathcal{H}(\Pi^2)$. Suppose now that Π^1 and Π^2 have positive elements, and therefore belong to $\mathbf{\Pi}_+$. In this case, the equality above holds if and only if $P^1 = P^2$ where $P^r = \left[\frac{\pi^r_{i,j}}{\sum_k \pi^r_{i,k}}\right]_{i,j}$ and $r = 1, 2$. Applying Lemma 4.5 we conclude that $\mathcal{H}(\Pi) = a\mathcal{H}(\Pi^1) + (1-a)\mathcal{H}(\Pi^2)$ if and only if $\Pi^1 = \Pi^2$. □

The expected energy and expected energy per bit corresponding to the elements of the set $\mathbf{\Pi}$ are defined analogously to those of previous sections. This is done by letting, $\mathcal{E}_{av}(\Pi) = \sum_{i,j} \pi_{i,j}\mathcal{E}(i \to j)$ and $\mathcal{E}_b(\Pi) = \mathcal{E}_{av}(\Pi)/\mathcal{H}(\Pi)$. Note also that if $(P, q) = \Pi$ then $\mathcal{E}_{av}(P, q) = \mathcal{E}_{av}(\Pi)$ and $\mathcal{E}_b(P, q) = \mathcal{E}_b(\Pi)$. It is straight-forward to verify that

$$\mathcal{E}_b(\alpha) = \min_{\Pi \in \mathbf{\Pi}, \mathcal{H}(\Pi) \geq an} \mathcal{E}_b(\Pi) \tag{52}$$

If (P,q) is an optimal point for $\mathcal{E}_b(\alpha) = \min_{(P,q)\in \mathbf{M}^*(\alpha)} \mathcal{E}_b(P,q)$ then the minimizing Π in (52) is achieved by $\Pi = [q_i P_{i,j}]_{i,j}$. Conversely, if $\Pi \in \mathbf{\Pi}$ achieves the minimum in (52) and $(P,q) = \Pi$ then (P,q) is the minimizing of $\mathcal{E}_b(\alpha) = \min_{(P,q)\in \mathbf{M}^*(\alpha)} \mathcal{E}_b(P,q)$.

Lemma 4.7 *Every solution* Π *of* $\mathcal{E}_b(\alpha) = \min_{\Pi\in\mathbf{\Pi},\mathcal{H}(\Pi)\geq an} \mathcal{E}_b(\Pi)$ *belongs to* $\mathbf{\Pi}_+$.

Proof: Let Π be a solution of (52). We show first that if for some state k , it is $\sum_j \pi_{k,j} > 0$, then it must be $\pi_{k,j} > 0$ for all j. Suppose that this is not the case and let r be such that $\pi_{k,r} = 0$. For $\delta \in (0, \frac{1}{2})$ we set $\Pi_\delta = (1-2\delta)\Pi + \delta e_k e_r^T + \delta e_r e_k^T$ where e_i is the vector with one in the *i-th* coordinate and zeros everywhere else. It is easy to verify that , $\Pi_\delta \in \mathbf{\Pi}$ for every δ. If $\pi^\delta_{i,j}$ denotes the i,j element of Π_δ, then $\mathcal{H}(\Pi_\delta) = -\sum_{i,j} \pi^\delta_{i,j} \log\left(\frac{\pi^\delta_{i,j}}{\sum_v \pi^\delta_{i,v}}\right)$. For every i,j we have :

$$\pi^\delta_{i,j} \log\left(\frac{\pi^\delta_{i,j}}{\sum_v \pi^\delta_{i,v}}\right) = \ldots$$

$$\ldots = \left[(1-2\delta)\pi_{i,j} + \delta\left(\delta_{i,k}\delta_{j,r} + \delta_{i,r}\delta_{j,k}\right)\right] \cdot \log\left(\frac{(1-2\delta)\pi_{i,j} + \delta\left(\delta_{i,k}\delta_{j,r} + \delta_{i,r}\delta_{j,k}\right)}{(1-2\delta)\sum_v \pi_{i,v} + \delta\left(\delta_{i,k} + \delta_{i,r}\right)}\right)$$

where $\delta_{i,j}$ is 1 if $i = j$ and zero otherwise. From the above expression we can conclude that for every i,j it is :

$$\pi^\delta_{i,j} \log\left(\frac{\pi^\delta_{i,j}}{\sum_v \pi^\delta_{i,v}}\right) = \pi_{i,j} \log\left(\frac{\pi_{i,j}}{\sum_v \pi_{i,v}}\right) + c_{i,j}\,\delta\log\delta + \mathcal{O}(\delta)$$

with $c_{i,j}$ being a non-negative constant (it is $c_{i,j} = 0, 1$ or 2). Now, according to our assumptions it is $\pi_{k,r} = 0$ while $\sum_j \pi_{k,j} > 0$. This implies that :

$$\pi^\delta_{k,r} \log\left(\frac{\pi^\delta_{k,r}}{\sum_v \pi^\delta_{k,v}}\right) = 0 + c_{k,r}\,\delta\log\delta + \mathcal{O}(\delta)$$

with $c_{k,r} = 2$ or $c_{k,r} = 1$ depending on whether $k = r$ or not. We conclude that there exists a positive constant $c_1 \geq 1$ such that for sufficiently small δ it is,

$$\mathcal{H}(\Pi_\delta) \;=\; \mathcal{H}(\Pi) - c_1\delta\log\delta + \mathcal{O}(\delta)$$

Also, because $\mathcal{E}_{av}(\Pi)$ depends linearly on Π, there is a constant c_2 such that,

$$\mathcal{E}_{av}(\Pi_\delta) \;=\; \mathcal{E}_{av}(\Pi) + c_2\delta$$

Note that $\mathcal{H}(\Pi)$ is positive since $\mathcal{H}(\Pi) \geq an$. Also, $\mathcal{E}_{av}(\Pi)$ is positive since the transition energy $\mathcal{E}(\sigma \rightarrow \omega)$ is zero only when $\sigma = \omega$. Therefore, setting $c_3 = c_2/\mathcal{E}_{ave}(\Pi)$ and $c_4 = c_1/\mathcal{H}(\Pi) > 0$ we have,

$$\frac{\mathcal{E}_{av}(\Pi_\delta)}{\mathcal{H}(\Pi_\delta)} = \frac{\mathcal{E}_{av}(\Pi)}{\mathcal{H}(\Pi)} \cdot \frac{1 + c_3\delta}{1 - c_4\delta \log \delta + \mathcal{O}(\delta)}$$

with $c_4 > 0$. This means that there exists an arbitrary small δ for which $\mathcal{E}_b(\Pi_\delta) < \mathcal{E}_b(\Pi)$ and $\mathcal{H}(\Pi_\delta) \geq \mathcal{H}(\Pi) = an$. A contradiction ! Note that a state k with $\sum_j \pi_{k,j} > 0$ always exists. Hence there exists a row of Π with only positive entries. This means that all column-sums of Π are positive and because of property (49), all row-sums are positive as well. Using the argument above we conclude that Π is a positive matrix. □

The above lemmas lead to the following theorem.

Theorem 4.2 *Problem (52) has a unique solution* Π. *The solution is positive,* $\Pi \in \mathbf{\Pi}_+$ *and with* $\mathcal{H}(\Pi) = an$. *For no* $\alpha \in (0, 1)$ *the solution equals the matrix* $\Omega = \frac{1}{2^{2n}}[1]_{i,j}$.

Proof: From Lemma 4.7 we know that every solution of Problem (52) is positive. Linearity of the expected energy function $\mathcal{E}_{av}(\Pi)$ and strict concavity of $\mathcal{H}(\Pi)$ in $\mathbf{\Pi}_+$ imply the uniqueness of the solution. The same properties also imply that $\mathcal{H}(\Pi) = an$. This can be shown in a way similar to that of Theorem 4.1. Finally it is $\mathcal{H}(\Omega) = n$ which gives $\Pi \neq \Omega$ since $\alpha n < n$. □

The theorem has the following immediate consequences.

Corollary 4.1 *The minimizing pair* (P, q) *in (45) (exists and) is unique. Furthermore, the transition probability matrix* P *and the probability eigenvector* q *are positive.*

Corollary 4.2 *The Problem (53) below has a unique solution. Its solution is positive and identical to that of Problem (52).*

$$\mathcal{E}_b(\alpha) = \min_{\Pi \in \mathbf{\Pi}, \mathcal{H}(\Pi) = an} \frac{1}{n\alpha} \mathcal{E}_{av}(\Pi). \tag{53}$$

Lemma 4.8 *The (unique) solution of Problems (52) and (53) is a regular point of the set of (active) constraints of problem (53), which are :* $\sum_{i,j}\pi_{i,j}=1$, $\sum_j \pi_{i,j}=\sum_k \pi_{k,i}$ *for every* i *and* $\mathcal{H}(\Pi)=an$.

Proof: The gradients of the constraints are given in matrix form below,

$$B = \left[\frac{\partial\left(\sum_{i,j}\pi_{i,j}\right)}{\partial\pi_{k,r}}\right]_{k,r} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \cdots & \vdots \\ 1 & 1 & \cdots & 1 \end{bmatrix}$$

$$A_i = \left[\frac{\partial\left(\sum_j(\pi_{i,j}-\pi_{j,i})\right)}{\partial\pi_{k,r}}\right]_{k,r} = \begin{bmatrix} 0 & \cdots & 0 & -1 & 0 & \cdots & 0 \\ \vdots & & \vdots & \vdots & \vdots & & \vdots \\ 0 & \cdots & 0 & -1 & 0 & \cdots & 0 \\ 1 & \cdots & 1 & 0 & 1 & \cdots & 1 \\ 0 & \cdots & 0 & -1 & 0 & \cdots & 0 \\ \vdots & & \vdots & \vdots & \vdots & & \vdots \\ 0 & \cdots & 0 & -1 & 0 & \cdots & 0 \end{bmatrix}$$

where the nonzero elements of matrices A_i are in the *i-th* row and the *i-th* column. And finally,

$$C = \left[\frac{\partial\mathcal{H}(\Pi)}{\partial\pi_{k,r}}\right]_{k,r} = \left[\log\left(\frac{\pi_{k,r}}{\sum_v \pi_{k,v}}\right)\right]_{k,r}$$

The set $\{A_i\}_i$ is linearly independent and B is orthogonal to every A_i under the inner product $< X, Y >= tr(X\,Y')$. We conclude that any non trivial linear dependence between the matrices A_i, B and C can be written as $C = bB + \sum_k r_k A_k$. Element-wise this is equivalent to $c_{i,j} = b + r_i - r_j$ or $P_{i,j} = \exp(b + r_i - r_j)$, with $P = \left[\frac{\pi_{i,j}}{\sum_k \pi_{i,k}}\right]_{i,j}$. Since for every state i it is $\sum_j P_{i,j} = 1$ we conclude that $r_i = r_j$ for all i, j and therefore $P = \frac{1}{2^n}[1]_{i,j}$. This implies $\Pi = [\frac{1}{2^{2n}}]_{i,j} = \Omega$ as the only irregular point of the constraints. The proof follows from an application of Theorem 4.2. □

The fact that the optimal Π satisfies $\mathcal{H}(\Pi) = an$ and $\Pi \in \mathbf{\Pi}_+$ combined with the smoothness of the constraints within $\mathbf{\Pi}_+$ gives the following result:

Corollary 4.3 *The solution of Problem (53) is a stationary point of its Lagrangian :*

$$\mathcal{L} = v \sum_{i,j} \pi_{i,j} + \sum_{i,j} \lambda_i \left(\pi_{i,j} - \pi_{j,i}\right) - \mu \sum_{i,j} \pi_{i,j} \ln\left(\frac{\pi_{i,j}}{\sum_k \pi_{i,k}}\right) + \sum_{i,j} \pi_{i,j}\, \mathcal{E}(i \to j) \tag{54}$$

Proof: From Lemma 4.8, the solution Π of the problem is a regular point of its constraints. Therefore there exists a set of real numbers v, λ_i, μ for which $\frac{\partial \mathcal{L}}{\partial \pi_{i,j}} = 0$ for all i, j. (Proposition 3.1.1, page 255 of [3]). □

Lemma 4.9 *Every stationary point Π of the Lagrangian $\mathcal{L}$ in the subset $\mathbf{\Pi}_+$ is of the form,*

$$\Pi = \left[\frac{g_i g_j e^{-\gamma \mathcal{E}(i \to j)}}{\sum_{r,k} g_r g_k e^{-\gamma \mathcal{E}(r \to k)}}\right]_{i,j} \tag{55}$$

where $g = (g_0, g_1, \ldots, g_{2^n-1})'$ is the unique (up to a constant) positive eigenvector of the matrix $W(\gamma) = \left[e^{-\gamma \mathcal{E}(i \to j)}\right]_{i,j}$ and γ is a real number.

Proof: The partial derivative of the Lagrangian with respect to the variable $\pi_{i,j}$ is :

$$\frac{\partial \mathcal{L}}{\partial \pi_{i,j}} = v + \lambda_i - \lambda_j - \mu \ln\left(\frac{\pi_{i,j}}{\sum_k \pi_{i,k}}\right) + \mathcal{E}(i \to j) \tag{56}$$

We are interested in the solutions of the set of equations $\frac{\partial \mathcal{L}}{\partial \pi_{i,j}} = 0$, i.e. :

$$v + \lambda_i - \lambda_j - \mu \ln\left(\frac{\pi_{i,j}}{\sum_k \pi_{i,k}}\right) + \mathcal{E}(i \to j) = 0 \tag{57}$$

First we examine the case $\mu = 0$ and show that it is not feasible. Suppose that $\mu = 0$. Then Equations (57) imply that : $\mathcal{E}(i \to j) = -v - \lambda_i + \lambda_j$ for every i, j. Since the energy cost function, Equation (1) is symmetric, that is $\mathcal{E}(i \to j) = \mathcal{E}(j \to i)$, we must have $\lambda_i = \lambda_j$ for every i, j. Therefore, $\mathcal{E}(i \to j) = -v$, i.e. the cost function is constant. But this is impossible since from Equation (1) we have that $\mathcal{E}(i \to i) = 0$ and $\mathcal{E}(i \to j) \neq 0$ for every $i \neq j$ (matrix $\mathcal{B}$ given by expression (2) is positive definite). Therefore $\mu \neq 0$, and the system of Equations (57) imply that :

$$\frac{\pi_{i,j}}{\sum_k \pi_{i,k}} = e^{\frac{v + \lambda_i - \lambda_j + \mathcal{E}(i \to j)}{\mu}} \tag{58}$$

Since all the parameters v, λ_i and μ are real we can do the transformation : $f = e^{v/\mu}$, $g_i = e^{-\lambda_i/\mu}$ and $\gamma = -1/\mu$, and have $f > 0$ and $g_i > 0$ for every i. Then, expression (58) becomes

$$\frac{\pi_{i,j}}{\sum_k \pi_{i,k}} = f\frac{g_j}{g_i}e^{-\gamma\mathcal{E}(i\to j)} \tag{59}$$

Summing over j we get $g_i = f\sum_j e^{-\gamma\mathcal{E}(i\to j)}g_j$ or written in matrix form, $g = fWg$ with $g = (g_0, g_1, \ldots, g_{2^n-1})'$ and $W = [e^{-\gamma\mathcal{E}(i\to j)}]_{i,j}$. The matrix fW is always positive and therefore g must be its unique (up to a factor) positive eigenvector (Theorem 4.4, page 16 of [10]). To this end f must be the inverse of the maximal eigenvalue λ_{max} of W. Recall that the energy cost function is symmetric, i.e. $\mathcal{E}(i \to j) = \mathcal{E}(j \to i)$ for every i, j. Therefore we also have that $g' = f\,g'W$ or more explicitly :

$$g_j = f\sum_i g_i e^{-\gamma\mathcal{E}(i\to j)} \tag{60}$$

We define the probability vector, $q = (g_0^2, g_1^2, \ldots, g_{2^n-1}^2)/\|g\|^2$ and using Equation (60) we show that it is a left eigenvector of P,

$$\begin{aligned}\sum_i q_i P_{i,j} &= \sum_i \frac{g_i^2}{\|g\|^2} f\frac{g_j}{g_i}e^{-\gamma\mathcal{E}(i\to j)} \\ &= \frac{g_j}{\|g\|^2} f\sum_i g_i e^{-\gamma\mathcal{E}(i\to j)} \\ &= \frac{g_j^2}{\|g\|^2} = q_j\end{aligned}$$

Applying Lemma 4.5 we get that $\Pi = (P, q)$. This means that every stationary point Π of the Lagrangian must have entries $\pi_{i,j}$ of the form,

$$\pi_{i,j} = \frac{g_i\, g_j}{\|g\|^2} f e^{-\gamma\mathcal{E}(i\to j)}.$$

This concludes the proof of the theorem. □

Note that for every γ the matrix W is positive and so it has a unique positive maximal eigenvalue. The matrix W is also analytic in γ. Therefore its eigenvalue and the corresponding normalized eigenvector are both analytic functions of γ (Theorem 1, page 396 of [7]). Even more, the stationary point Π of the Lagrangian, parameterized on γ, approaches the identity matrix as γ approaches $+\infty$. This is because only transitions from every state to

itself have zero cost. Also, Π approaches Ω as γ approaches zero. We conclude that for every $\alpha \in (0,1)$ there exists a γ that solves the equation $\mathcal{H}(\Pi) = an$. It can be shown, although it is very involved, that for every $\alpha \in (0,1)$, there exists a unique γ for which the corresponding Π is the solution of the Problem (53). This γ is non-negative. In addition, $\mathcal{H}(\Pi)$ is strictly decreasing function of γ for $\gamma > 0$.

Theorem 4.3 *For every $\alpha \in (0,1)$ there exists a unique positive γ for which the matrix Π defined by Equation (55) satisfies $\mathcal{H}(\Pi) = an$ and is the unique solution of Problems (52) and (53). As a function of γ the entropy $\mathcal{H}(\Pi)$ is strictly decreasing for $\gamma > 0$.*

Theorem 4.3 is the tool to evaluate the limiting expected energy per transmitted bit, $\mathcal{E}_b(\alpha)$. Figures 5, 6, and 7 present the normalized value $\mathcal{E}_b(\alpha)/\mathcal{E}_u$, named the *Optimal Bound*, as a function of the bus utilization α, for $\lambda = 5$ and $n = 2, 4, 8$. For comparison, we have included the exponential and the entropy bounds.

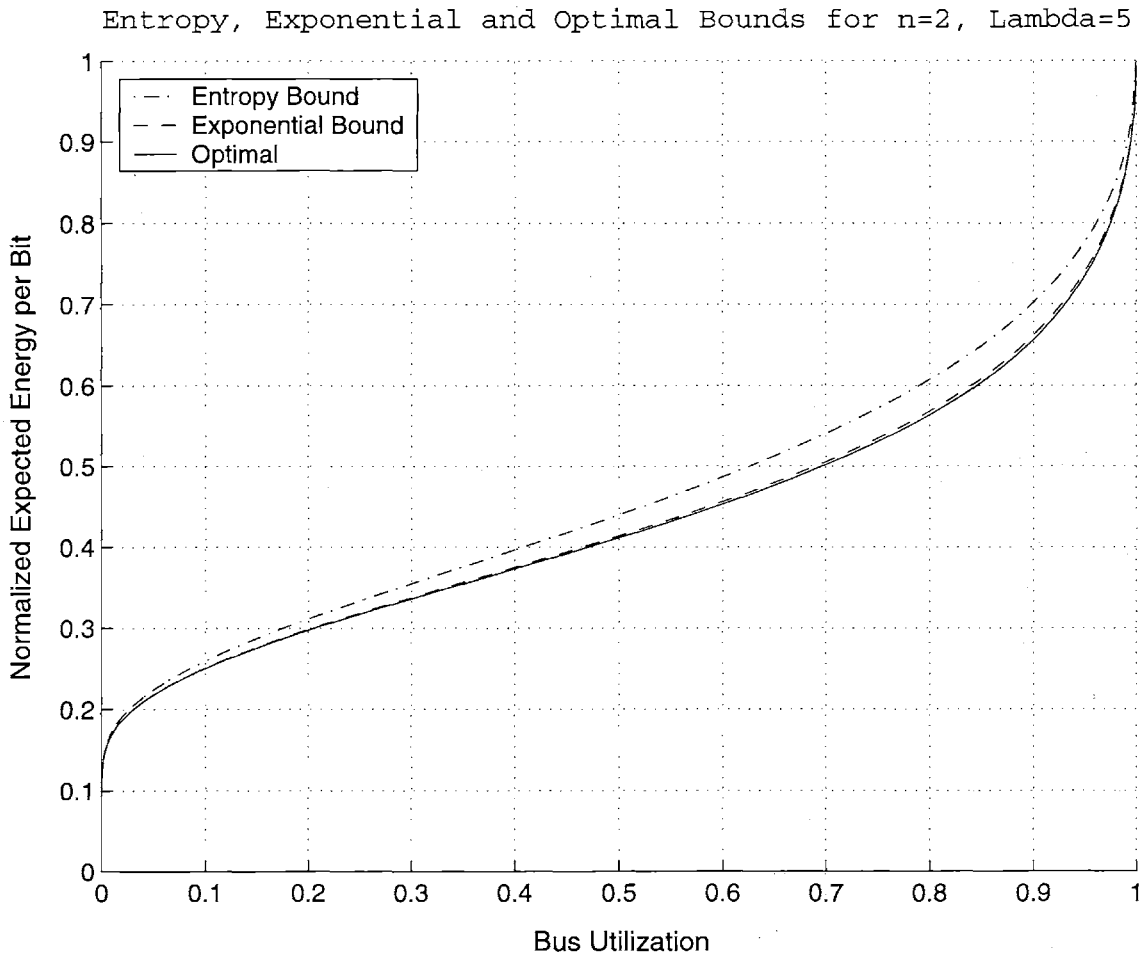

Figure 5: Comparison of the Entropy, Exponential and Optimal Bounds for $n = 2$ and $\lambda = 5$.

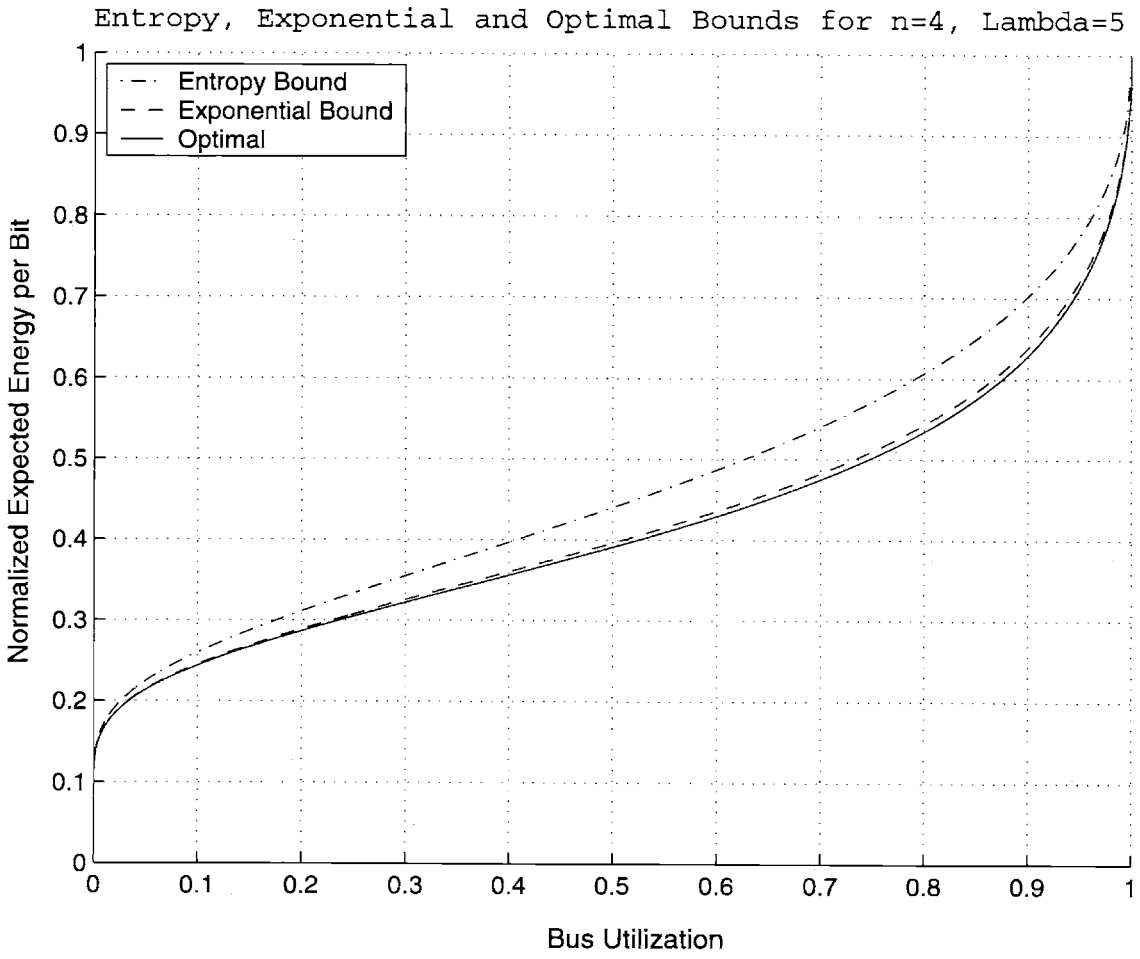

Figure 6: Comparison of the Entropy, Exponential and Optimal Bounds for $n = 4$ and $\lambda = 5$.

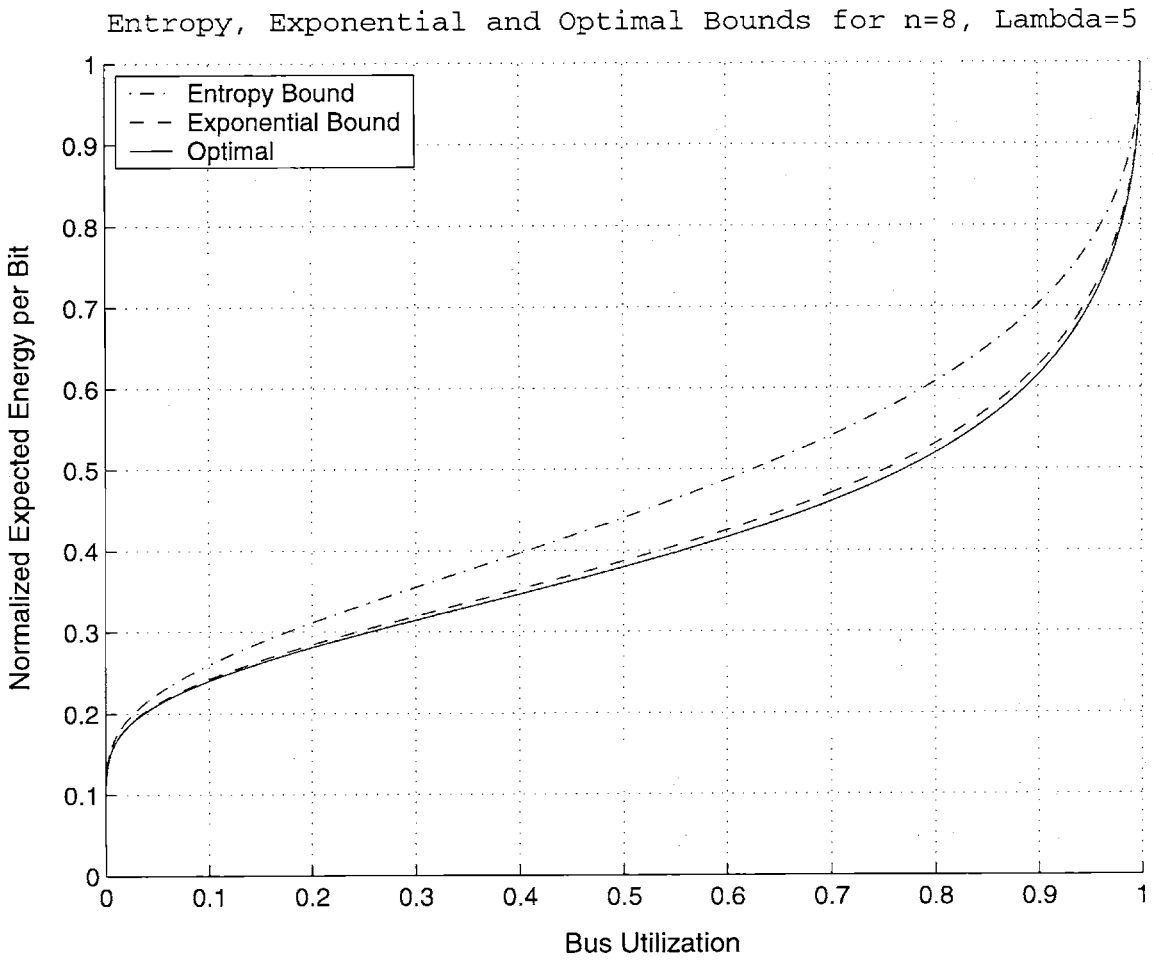

Figure 7: Comparison of the Entropy, Exponential and Optimal Bounds for $n = 8$ and $\lambda = 5$.

5 A Comment

Note that the presented analysis has been based on the fact that the structure of the bus is **given**. Given the structure, we asked what is the best way to operate it in order to communicate data at some rate with the minimum possible energy per bit. This is what the last three graphs presented. Now, choosing the *best bus structure*, within some limits, is a **different** problem. If for example we were allowed to change the structure of the bus, then we could remove some lines, say $n - k$ out of n and split the rest to decrease the interline coupling while keeping the total width fixed. This is illustrated in Example 3 of Chapter 8. Assuming i.i.d. and uniformly distributed data and using the energy expression (52) of Chapter 2, we get the following graph. (*This is a specific example using special data distribution and based on rough and rather optimistic approximations. It should not be used for general comparisons*). The stars "$*$" correspond to buses with k lines, maximum utilization and therefore relative utilization equal to k/n.

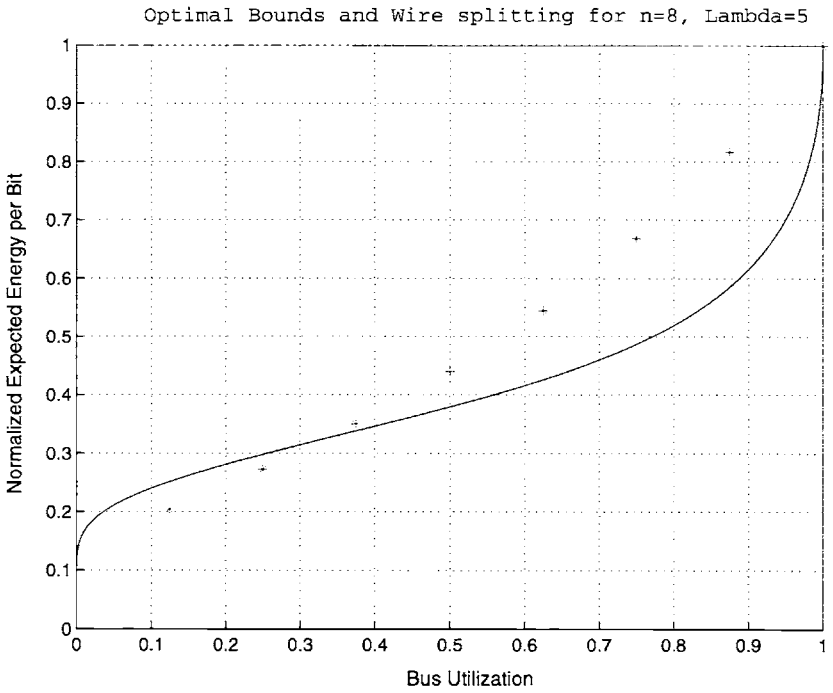

Figure 8: Optimal Bounds and Wire Splitting for $n = 8$ and $\lambda = 5$.

We conclude that the right amount of redundancy is needed for maximum energy reduction in buses when the bus width is fixed.

6 Conclusions and Final Remarks

We have considered the problem of reduction of energy consumption (or other computation cost functions) in a computation module as a coding and information theory problem. It has been shown that redundancy in the capacity of the computation module can be exploited, using coding techniques, to achieve significant reductions in the energy consumption (or other computation cost). Redundancy can be introduced by adding extra ports in the module and by coding the input and output sequences. Using tools of information theory, we have derived the maximum possible cost reductions in an explicit analytical format.

Although, for the ease of presentation, we presented the results using a specific cost function, it is noted that the methods we established here apply to any non-negative cost function for which the cost of a sequence of computations is the sum of the costs of individual steps.

References

[1] V. Agarwal, M. Hrishikesh, S. Keckler, D. Burger, "Clock rate versus IPC: the end of the road for conventional microarchitectures ", *Proceedings of the 27th Int. Symp. on Computer Architecture*, 2000, pp. 248-259.

[2] L. Benini, G. De Micheli, E. Macii, D. Sciuto, C. Silvano, "Asymptotic zero-transition activity encoding for address busses in low-power microprocessor-based systems", *Proceedings of Great Lakes VLSI Symp* , 1997, pp. 77-82.

[3] D. Bertsekas, *Nonlinear Programming*, Athena Scientific, Massachusetts 1995.

[4] T. Cover and J. Thomas, *Elements of Information Theory*, John Wiley and Sons, 1991.

[5] R. Fletcher, *Integrated circuit having outputs configured for reduced state changes*, U.S. Patent 4,667,337, May 1987.

[6] K. Kim, K. Baek, N. Shanbhag, C. Liu, S. Kang, "Coupling-Driven Signal Encoding Scheme for Low-Power Interface Design", *Proceedings of ICCAD*, Nov. 2000.

[7] P. Lancaster and M. Tismenetsky, *The Theory of Matrices, Second Edition*, Academic Press INC, 1988.

[8] F. MacWilliams and N. Sloane, *The Theory of Error Correcting Codes*, Amsterdam, The Netherlands: North-Holland, 1996.

[9] D. Marculescu, R. Marculescu, M. Pedram, "Information theoretic measures for power analysis," *IEEE Trans. Computer-Aided Design*, vol. 15, pp. 599-610, June 1996.

[10] H. Minc, *Nonnegative Matrices*, Wiley interscience series in discrete mathematics and optimization, 1988.

[11] F. Najm, "Transition Density: A New Measure of Activity in Digital Circuits", *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 12, No. 2, Feb 1993.

[12] S. Ramprasad, N. Shanbhag, I. Hajj, "Information-Theoretic Bounds on Average Signal Transition Activity", *IEEE Trans. on VLSI*, Vol. 7, No. 3, Sept. 1999.

[13] S. Ramprasad, N. Shanbhag, I. Hajj, "A coding framework for low-power address and data busses", *IEEE Trans. on VLSI*, vol. 7, June 1999.

[14] S. Savari, R. Gallager, "Arithmetic coding for finite-state noiseless channels", *IEEE Transactions on Information Theory*, Vol. 40, Issue 1, pp. 100-107, Jan. 1994.

[15] N. Shanbhag, "A mathematical basis for power reduction in digital VLSI systems", *IEEE Trans. Circuits Syst. II*, Vol. 44, pp. 935-951, Nov. 1997.

[16] M. Stan, W. Burleson, "Limited weight codes for low-power I/0", *Proceedings of IWLPD*, pp. 209-215, 1994.

[17] M. Stan, W. Burleson, "Bus-invert coding for low-power I/0", *IEEE Trans. on VLSI*, pp. 49-58, Vol. 3, No. 5, March 1995.

[18] M. Stan, W. Burleson, "Low-power encodings for global communication in CMOS VLSI", *IEEE Transactions on VLSI Systems*, pp. 444-455, Vol. 5, No. 4, Dec. 1997.

[19] P. Sotiriadis and A. Chandrakasan, "A Bus Energy Model for Deep Sub-micron Technology", *To be published in IEEE Transactions on VLSI Systems.*

[20] P. Sotiriadis, A. Chandrakasan, "Low Power Bus Coding Techniques Considering Inter-wire Capacitances", *Proceedings of CICC*, May 2000.

[21] P. Sotiriadis, A. Wang, A. Chandrakasan, "Transition Pattern Coding: An approach to reduce Energy in Interconnect", *Proceedings of ESSCIRC*, Stockholm, Sweden, Sept. 2000.

[22] P. Sotiriadis, A. Chandrakasan, "Bus Energy Minimization by Transition Pattern Coding (TPC) in Deep Sub-Micron Technologies", *Proceedings of ICCAD*, Nov. 2000.

[23] S. Srinivasan, K. Mehta, *Stochastic Processes*, Tata McGraw-Hill Publishing Company Limited, New Delhi 1988.

Chapter 4

Linear, Block Linear and Convolutional Coding For Power Reduction

1 Introduction

In very recent deep-sub-micron technologies where lines are very close together and strong, undesirable interference takes place, the reliability issue in microprocessors has become critical [1]. Buses are particularly sensitive to interference because of their infrastructure. Error correction mechanisms have started to appear in bus designs [3], most of them employing very simple schemes.

The theory of error-correction codes has been well developed during the last fifty years, (see for example [4]). From a theoretical perspective, error-correction coding for buses is not a new problem. What is new and very interesting is the connection between error-correction and power-reduction coding as well as their co-design. This is a challenging and important problem in an age where performance and reliability are the ultimate goals.

In this chapter we study in detail the energy "behavior" of differential, convolutional, linear and block linear codes applied to the deep sub-micron bus energy model introduced in Chapter 2. It is shown that these codes can only increase the power consumption of buses. Even though power reduction is not possible, power minimization is very important since these codes are very useful for error correction. In the following paragraphs we relate the resulting power consumption using such coding schemes with the structure of their generator matrices. In some cases the energy is calculated exactly, and in other cases we provide bounds. Both provide intuition about how to re-structure a given linear (block linear etc) code so that the energy is minimized within the set of all equivalent codes.

2 Definitions and Preliminary Results

For the bus, we assume the energy model of Figure 7 in Chapter 2 where the fringing capacitors C_F are taken equal to the interline capacitors C_I. Recall from expression (32) of Chapter 2 that the energy dissipated during the transition of the bus from logical state $S_t = (s_t^1, s_t^2, s_t^3, \ldots, s_t^n)$ in $\{0,1\}^n$ at time t to logical state $S_{t+1} = (s_{t+1}^1, s_{t+1}^2, s_{t+1}^3, \ldots, s_{t+1}^n)$ in $\{0,1\}^n$ at time $t+1$ is give by :

$$\mathcal{E}(S_t \rightarrow S_{t+1}) = E_0 (S_{t+1} - S_t)\, \mathcal{B}\, (S_{t+1} - S_t)^H \tag{1}$$

where

$$\mathcal{B} = \begin{pmatrix} 1+2\lambda & -\lambda & 0 & 0 & \cdots & 0 & 0 & 0 \\ -\lambda & 1+2\lambda & -\lambda & 0 & \cdots & 0 & 0 & 0 \\ 0 & -\lambda & 1+2\lambda & -\lambda & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \cdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & -\lambda & 1+2\lambda & -\lambda \\ 0 & 0 & 0 & 0 & 0 & \cdots & -\lambda & 1+2\lambda \end{pmatrix},$$

$E_0 = C_L V_{dd}^2/2$ and $\lambda = C_I/C_L$ exactly as in Equation (25) of Chapter 2. The ratio λ

is nonnegative, and n is equal to the number of lines in the bus. Since E_0 is a normalization factor, for notational simplicity we assume that $E_0 = 1$ throughout this chapter. Finally, the total energy dissipation corresponding to a sequence $\mathbf{c} = (S_1, S_2, S_3, \cdots, S_T)$ is given by $\mathcal{E}(\mathbf{c}) = \sum_{t=1}^{T-1} \mathcal{E}(S_t \to S_{t+1})$.

Throughout the chapter we mix the algebra of the fields, $\mathbb{Z}_2$ with addition $\oplus$, and, $\mathbb{R}$ with addition $+$. Priority is given to the binary addition $\oplus$ over the real addition $+$. In this sense we have that for binary constants a, a_1, a_2 and b, b_1, b_2 it is:

$$a - b = (-1)^b \, a \oplus b \tag{2}$$

and

$$\begin{aligned} (a_1 - b_1)(a_2 - b_2) &= (-1)^{b_1 + b_2} \, (a_1 \oplus b_1)(a_2 \oplus b_2) \\ &= (-1)^{b_1 \oplus b_2} \, (a_1 \oplus b_1)(a_2 \oplus b_2) \end{aligned} \tag{3}$$

Let $f(\alpha, \beta, \cdots)$ be a function of the random variables $\alpha, \beta, \cdots$. We denote by $\overline{f(\alpha, \beta, \cdots)}^{\beta}$ the expected value of $f(\alpha, \beta, \cdots)$ with respect only to random variable β and by $\overline{f(\alpha, \beta, \cdots)}$ the expected value of the expression over all random variables involved. Note that if a, b are independent random variables, uniformly distributed in $\mathbb{Z}_2$ then $a \oplus b$ is also uniformly distributed. Even more, $a, a \oplus b$ and $a \oplus b, b$ are pairs of independent variables. In this sense (2) implies that:

$$\overline{a - b} = \overline{(-1)^b} \cdot \overline{a \oplus b} = 0 \tag{4}$$

and

$$\overline{(a - b)^2} = \overline{a \oplus b} = 1/2 \tag{5}$$

Moreover, if the variables a_1, a_2, b_1, b_2 are independent and uniformly distributed in $\mathbb{Z}_2$ then from equation (3) and we get:

$$\overline{(a_1 - b_1)(a_2 - b_2)} = \overline{(-1)^{b_1 \oplus b_2}} \cdot \overline{(a_1 \oplus b_1)} \cdot \overline{(a_2 \oplus b_2)} = 0 \cdot \frac{1}{2} \cdot \frac{1}{2} = 0 \tag{6}$$

Lemma 2.1 *Let x and z be two independent random variables uniformly distributed in $\mathbb{Z}_2^n$. The expected energy for the transition $x \rightarrow z$ is:*

$$\overline{\mathcal{E}(x \rightarrow z)} = n\,(1 + 2\lambda)/2 \tag{7}$$

Proof: From the expression for the transition energy (1), equation (5) and equation (6) we have that:

$$\begin{aligned}
\overline{\mathcal{E}(x \rightarrow z)} &= \overline{(1+2\lambda)\sum_{i=1}^{n}(z_i - x_i)^2 - 2\lambda\sum_{i=1}^{n-1}(z_i - x_i)(z_{i+1} - x_{i+1})} \\
&= (1+2\lambda)\sum_{i=1}^{n}\overline{(z_i - x_i)^2} - 2\lambda\sum_{i=1}^{n-1}\overline{(z_i - x_i)(z_{i+1} - x_{i+1})} \\
&= (1+2\lambda)\sum_{i=1}^{n}\frac{1}{2} - 2\lambda\sum_{i=1}^{n-1}0 \\
&= \frac{1+2\lambda}{2}\,n \qquad \square
\end{aligned}$$

Definition 2.1 *We define $\mathcal{E}_u$ to be the expected energy per transmitted bit when a sequence of i.i.d. and uniformly distributed in $\mathbb{Z}_2^n$ bit-vectors is transmitted through the bus.*

The previous Lemma implies that:

$$\mathcal{E}_u = \frac{1+2\lambda}{2}$$

We stress that $\mathcal{E}_u$ was calculated with the assumption that the data was *i.i.d.* and uniformly distributed and therefore the bit rate through the bus was the maximum possible. Therefore $\mathcal{E}_u$ can be used as a reference value for evaluating the efficiency of energy reduction coding schemes.

For a vector $x = (x_1, x_2, \cdots, x_n)$ in $\mathbb{Z}_2^n$ we denote its Hamming weight by $w(x)$. The following Lemma was proven in Chapter 2 (Lemma 2.1). For convenience we repeat its proof.

Lemma 2.2 *Let x be a random vector uniformly distributed in $\mathbb{Z}_2^n$. Then for every constant (or random but independent of x) vector s it is:*

$$\overline{\mathcal{E}(x \to x \oplus s)}^{\,x} = 2\,w(s)\,\mathcal{E}_u \tag{8}$$

Proof: Let $x = (x_1, x_2, \cdots, x_n)$, $s = (s_1, s_2, \cdots, s_n)$. Using the expression of the transition energy (1) and equations (2) and (3) we have:

$$\begin{aligned}\mathcal{E}(x \to x \oplus s) &= (1+2\lambda)\sum_{i=1}^{n}(x_i \oplus s_i - x_i)^2 - 2\lambda\sum_{i=1}^{n-1}(x_i \oplus s_i - x_i)(x_{i+1} \oplus s_{i+1} - x_{i+1}) \\ &= (1+2\lambda)\sum_{i=1}^{n} s_i - 2\lambda\sum_{i=1}^{n-1}(-1)^{x_i \oplus x_{i+1}}\, s_i\, s_{i+1}\end{aligned}$$

and taking the expectation over x,

$$\overline{\mathcal{E}(x \to x \oplus s)} = (1+2\lambda)\sum_{i=1}^{n} s_i - 2\lambda\sum_{i=1}^{n-1} 0 = 2\,w(s)\,\mathcal{E}_u \qquad \square$$

Now, let $\mathcal{C}$ be a $[n, k, d]$ linear code (see for example [5] or [4] for the definition of the triple $[n, k, d]$) on $\mathbb{Z}_2$ and let the full rank $k \times n$ matrix G be a generator of it, that is:

$$\mathcal{C} = \left\{vG : v \in \mathbb{Z}_2^k\right\} \tag{9}$$

Note that the mapping $v \mapsto vG$ from $\mathbb{Z}_2^k$ to $\mathcal{C}$ is bijective. So, if v is a random variable uniformly distributed in $\mathbb{Z}_2^k$ then the random variable $w = vG$ is uniformly distributed in $\mathcal{C}$ as well. In this case, the coordinates of w have some straight forward, although very useful properties that relate to the columns of the matrix $G = [g_1, g_2, \cdots, g_n]$. They are summarized in the following Lemma.

Lemma 2.3 *If v is a random variable uniformly distributed in $\mathbb{Z}_2^k$ then the random variable $w = vG$ is uniformly distributed in $\mathcal{C}$ and its coordinates, $w = (w_1, w_2, \cdots, w_n)$ have the properties:*

1. *w_i is identically zero if $g_i = 0$ and uniformly distributed in $\mathbb{Z}_2$ if $g_i \neq 0$.*

2. *w_i and w_j coincide if and only if $g_i = g_j$.*

3. *w_i and w_j are independent if and only if $g_i \neq g_j$.*

Now let $\{i_1, i_2, \cdots, i_p\}$ and $\{j_1, j_2, \cdots, j_q\}$ be two subsets of $\{1, 2, \cdots, n\}$. We define the two random variables $a = w_{i_1} \oplus w_{i_2} \oplus \cdots \oplus w_{i_p}$ and $b = w_{j_1} \oplus w_{j_2} \oplus \cdots \oplus w_{j_q}$ and the two column vectors $g_a = g_{i_1} \oplus g_{i_2} \oplus \cdots \oplus g_{i_p}$ and $g_b = g_{j_1} \oplus g_{j_2} \oplus \cdots \oplus g_{j_p}$. It is of course

$$\begin{aligned} a &= w_{i_1} \oplus w_{i_2} \oplus \ldots \oplus w_{i_p} \\ &= (vg_{i_1}) \oplus (vg_{i_2}) \oplus \ldots \oplus (vg_{i_p}) \\ &= vg_a \end{aligned}$$

where vg_{i_r} is the inner product of vectors v and g_{i_r} in $\mathbb{Z}_2^k$. Similarly we have that $b = vg_b$. Lemma 2.3 has the immediate generalization:

Lemma 2.4 *With the setting of Lemma 2.3 and the previous definitions we have for the random variable (r.v.) a and b that:*

1. *The r.v. a is identically zero if $g_a = 0$ and uniformly distributed in $\mathbb{Z}_2$ if $g_a \neq 0$.*
2. *The r.v.s a and b coincide if and only if $g_a = g_b$.*
3. *The r.v.s a and b are independent if and only if $g_a \neq g_b$.*

The Lemma has the following consequences:

$$\overline{a} = \begin{cases} 1/2 \text{ if } g_a \neq 0 \\ 0 \text{ otherwise} \end{cases} \tag{10}$$

and

$$\overline{a\,b} = \begin{cases} 1/2 \text{ if } g_a \neq 0\,,\ g_b \neq 0\,,\ g_a = g_b \\ 1/4 \text{ if } g_a \neq 0\,,\ g_b \neq 0\,,\ g_a \neq g_b \\ 0 \text{ otherwise} \end{cases} \tag{11}$$

Even more,

$$\overline{(-1)^{a\oplus b}} = \begin{cases} 1 \text{ if } g_a = g_b \\ 0 \text{ otherwise} \end{cases} \tag{12}$$

The random vector w has been defined as $w = u\,G$ where u is uniformly distributed in $\mathbb{Z}_2^k$. In this case we will say that that matrix G **generates** w. Let F be an invertible $k \times k$ matrix. Since mapping $u \to uG$ is bijective in $\mathbb{Z}_2^k$, the random vector $\hat{u} = uF^{-1}$ is also uniformly distributed in $\mathbb{Z}_2^k$. Therefore the matrix FG is another generator of w.

We will call a random variable w_i **active** if it is not zero with probability one. With the setting of Lemma 2.3 and Lemma 2.4 we have the following.

Lemma 2.5 *The active random variables $w_{i_1}, w_{i_2}, \cdots, w_{i_q}$ are independent if and only if the columns $g_{i_1}, g_{i_2}, \cdots, g_{i_q}$ of matrix G are linearly independent. If the columns are linearly independent then the vector $(w_{i_1}, w_{i_2}, \cdots, w_{i_q})$ is uniformly distributed in $\mathbb{Z}_2^k$.*

Proof: Set $G_q = [g_{i_1}, g_{i_2}, \cdots, g_{i_q}]$ and let $q' = rank(G)$. The proof comes directly from the fact (see for example [2]) that there exist an invertible $k \times k$ matrix F and a permutation $q \times q$ matrix P such that :

$$F\,G\,P = \left[\begin{array}{c|c} \mathcal{I} & * \\ \hline 0 & 0 \end{array}\right]$$

where $\mathcal{I}$ is the $q' \times q'$ identity matrix and $*$ is a $q' \times (q - q')$ matrix. $\square$

3 Linear Differential Coding

In this section we study the efficiency of linear differential coding for energy reduction in DSM buses. The structure of the encoder is shown below. The input vectors $u(r)$ are

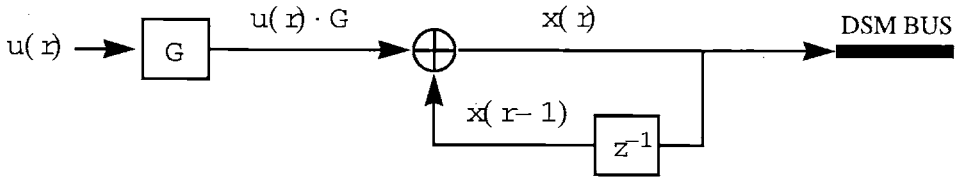

Figure 1: Differential Coding

assumed to be uniformly distributed in $\mathbb{Z}_2^k$ and independent to each other. The vectors $x(r)$ that are transmitted through the bus belong to $\mathbb{Z}_2^n$. The coding schemes has rate k/n and the variables satisfy the relation:

$$x(r) = x(r-1) \oplus u(r)\, G\,, \quad x(0) \in ST_0 \tag{13}$$

Where ST_0 is the set containing the initial state of the bus. The expected energy per transmitted bit is given by:

$$\mathcal{E}_b = \frac{1}{k}\, \overline{\mathcal{E}\left(x(r-1) \to x(r-1) \oplus v(r)\right)} \tag{14}$$

Throughout the chapter we assume that the matrix G is of full rank. Even more, for convenience we set $v(r) = u(r)\, G$, and we note that the random vector $v(r)$ is uniformly distributed within the code

$$\mathcal{C} = \{u\, G,\, u \in \mathbb{Z}_2^k\} \tag{15}$$

and independent of $x(r-1)$. So, the expected energy per transmitted bit can be calculated by taking the expectations over $x(r-1)$ and $v(r)$ independently of each other, i.e.:

$$\mathcal{E}_b = \frac{1}{k}\, \overline{\overline{\mathcal{E}\left(x(r-1) \to x(r-1) \oplus v\right)}^{\,x(r-1)}}^{\,v(r)} \tag{16}$$

Finally, the value of $\mathcal{E}_b$ depends not only on the particular generator matrix G but on the initial condition as well. To motivate the analysis of the coding scheme we examine first the simplest case.

3.1 Linear Differential Coding with Initial State Uniformly Distributed in $\mathbb{Z}_2^k$

In this section we analyze the case where $x(0)$ *is uniformly distributed in* $\mathbb{Z}_2^n$ *and independent of the input sequence* $u(1), u(2), \cdots$. The expected energy per bit of the scheme is given by the following Theorem.

Theorem 3.1 *The expected energy per transmitted bit using the linear differential coding scheme with initial state* $x(0)$ *uniformly distributed in* $\mathbb{Z}_2^n$ *and independent of the input sequence is :*

$$\mathcal{E}_b = \frac{\mu}{k}\,\mathcal{E}_u \tag{17}$$

with μ *being the number of non-zero columns of matrix* G. *Since* $\mu \geq k$ *it is* $\mathcal{E}_b \geq \mathcal{E}_u$.

Proof: The recursive relation (13) implies that for every r, the vector $x(r-1)$ is also uniformly distributed in $\mathbb{Z}_2^n$ and independent of $u(r)$. This allows as to use the result of Lemma 2.2 and write

$$\overline{\mathcal{E}\left(x(r-1) \rightarrow x(r-1) \oplus v(r)\right)}^{\,x(r-1)} = 2\,w(v(r))\,\mathcal{E}_u. \tag{18}$$

Therefore, the expected energy per bit, Equation (14), takes the form:

$$\mathcal{E}_b = \frac{1}{k}\,\overline{\overline{\mathcal{E}\left(x(r-1) \rightarrow x(r-1) \oplus v\right)}^{\,x(r-1)}}^{\,v(r)} = \frac{2\,\mathcal{E}_u}{k}\,\overline{w(v(r))} \tag{19}$$

The issue here is to calculate the expected weight of of the codewords in $\mathcal{C} = \{u\,G,\, u \in \mathbb{Z}_2^k\}$. This can be done easily using the McWilliams identity [4] which states that for every $\theta \neq 1$ it is

$$\sum_{i=1}^{n} A_i\,\theta^i = \frac{1}{2^{n-k}} \sum_{i=1}^{n} B_i (1+\theta)^{n-i}(1-\theta)^i \tag{20}$$

where A_i and B_i are the numbers of codewords in $\mathcal{C}$ and $\mathcal{C}^{\perp}$ respectively that have weight i. Taking the derivative of both sides in (20) we get

$$\sum_{i=1}^{n} i\, A_i\, \theta^{i-1} = \frac{1}{2^{n-k}} \sum_{i=1}^{n} \left[(n-i)\, B_i\, (1+\theta)^{n-i-1}(1-\theta)^i - i\, B_i\, (1+\theta)^{n-i}(1-\theta)^{i-1}\right]$$

Finally taking the limit as $\theta \longrightarrow 1$ we have

$$\sum_{i=1}^{n} i\, A_i = 2^{k-1}\, (n\, B_0 - B_1) \tag{21}$$

Note that for every code $B_0 = 1$ and that B_1 is the number of codewords in $\mathcal{C}^{\perp}$ of weight 1. This means that exactly B_1 columns of G are zero and so G has exactly $\tau = n - B_1$ non-zero columns. So

$$\frac{1}{2^k} \sum_{i=1}^{n} i\, a_i = \frac{\tau}{2} \tag{22}$$

The proof is concluded by observing that

$$\overline{w(v(r))} = \frac{1}{2^k} \sum_{i=1}^{n} i\, A_i \qquad \square$$

The interesting point here is that the expected energy per bit is determined only by the active lines, i.e. the number of non-zero columns of G and no other structural characteristics of it.

Although this kind of coding does not reduce the power consumption on the bus, the relation (17) allows us to estimate the power overhead when the above coding scheme is used for error correction.

3.2 Linear Differential Coding with Initial State Distributed in $\mathcal{C}$

In this section, the initial state $x(0)$ *is assumed arbitrarily distributed within the code* $\mathcal{C}$. In this case the expected energy per bit is given by the following Theorem.

Theorem 3.2 *The expected energy per transmitted bit $\mathcal{E}_b$, using the linear differential coding scheme with initial state $x(0)$, that is arbitrarily distributed within the code $\mathcal{C}$ and independent of the input sequence, can be expressed as:*

$$\mathcal{E}_b = \frac{\mu - \tilde{\mu}}{k}\, \mathcal{E}_u + \frac{\tilde{\mu}}{2k} \tag{23}$$

where μ is the number of non-zero columns of matrix G and $\tilde{\mu}$ is the number of indices $i = 1, 2, \cdots, n-1$ for which $g_i = g_{i+1} \neq 0$. It is $\mu - \tilde{\mu} \geq k$ and $\tilde{\mu} \geq 0$, therefore it is always $\mathcal{E}_b \geq \mathcal{E}_u$.

The proof of the Theorem will be a consequence of the following Lemma.

Lemma 3.1 *Let x and z be two independent random vectors uniformly distributed in $\mathcal{C} = \{u\,G,\, u \in \mathbb{Z}_2^k\}$. Then, with μ and $\tilde{\mu}$ defined as in Theorem 3.2 above , it is:*

$$\overline{\mathcal{E}(x \to z)}^{\,x,z} = (\mu - \tilde{\mu})\,\mathcal{E}_u + \tilde{\mu}/2 \tag{24}$$

Proof: Let $x = (x_1, x_2, \cdots, x_n)$ and $z = (z_1, z_2, \cdots, z_n)$. From expression (1) of the transition energy and equation (3) we have:

$$\begin{aligned}
\mathcal{E}(x \to z) &= (1+2\lambda)\sum_{i=1}^{n}(z_i - x_i)^2 - 2\lambda\sum_{i=1}^{n-1}(z_i - x_i)(z_{i+1} - x_{i+1}) \\
&= (1+2\lambda)\sum_{i=1}^{n} z_i \oplus x_i - 2\lambda\sum_{i=1}^{n-1}(-1)^{x_i + x_{i+1}}\,(z_i \oplus x_i)(z_{i+1} \oplus x_{i+1}) \\
&= (1+2\lambda)\sum_{i=1}^{n} w_i - 2\lambda\sum_{i=1}^{n-1}(-1)^{x_i \oplus x_{i+1}}\,w_i\,w_{i+1}
\end{aligned}$$

where we have set $w_i = x_i \oplus z_i$. Since w_i and x_j are statistically independent for every i, j, we have :

$$\overline{\mathcal{E}(x \to z)}^{\,x,z} = (1+2\lambda)\sum_{i=1}^{n}\overline{w_i}^{\,w} - 2\lambda\sum_{i=1}^{n-1}\overline{(-1)^{x_i \oplus x_{i+1}}}^{\,x} \cdot \overline{w_i\,w_{i+1}}^{\,w} \tag{25}$$

It is convenient to define the following "characteristic indices" of the columns $g_1, g_2, \cdots, g_n$ of G :

$$d_i = \begin{cases} 1 \text{ if } g_i \neq 0 \\ 0 \text{ if } g_i = 0 \end{cases} \quad \text{and} \quad \tilde{d}_i = \begin{cases} 1 \text{ if } g_i = g_{i+1} \\ 0 \text{ if } g_i \neq g_{i+1} \end{cases} \tag{26}$$

then using expressions (5), (12) and (11) we can write

$$\overline{w_i}^{\,w} = \frac{d_i}{2} \quad , \quad \overline{(-1)^{x_i \oplus x_{i+1}}}^{\,x} = \tilde{d}_i \quad , \quad \overline{w_i\,w_{i+1}}^{\,w} = \frac{(\tilde{d}_i + 1)}{4}\,d_i\,d_{i+1} \tag{27}$$

Equations (27) imply that

$$\overline{(-1)^{x_i \oplus x_{i+1}}}^{\,x} \cdot \overline{w_i\, w_{i+1}}^{\,w} = \frac{d_i\, \tilde{d}_i}{2} \tag{28}$$

Finally note that by their definition, μ and $\tilde{\mu}$ can be expressed as:

$$\mu = \sum_{i=1}^{n} d_i \quad \text{and} \quad \tilde{\mu} = \sum_{i=1}^{n-1} d_i\, \tilde{d}_i \tag{29}$$

Therefore we have:

$$\begin{aligned} \overline{\mathcal{E}(x \to z)}^{\,x,z} &= (1+2\lambda) \sum_{i=1}^{n} \frac{d_i}{2} - \lambda \sum_{i=1}^{n-1} d_i\, \tilde{d}_i \\ &= \mathcal{E}_u\, \mu - (\mathcal{E}_u - 1/2)\, \tilde{\mu} \\ &= (\mu - \tilde{\mu})\, \mathcal{E}_u + \tilde{\mu}/2 \end{aligned} \tag{30}$$

This concludes the proof of the Lemma. □

Proof of Theorem 3.2: We only need to note that the initial state $x(0)$ takes values in $\mathcal{C}$ and it is independent of the input sequence, therefore, the relation $x(r) = x(r-1) \oplus u(r)\, G$ implies that the vectors $x(r-1)$ and $x(r)$ are independent and uniformly distributed in $\mathcal{C}$ for every r. From the above Lemma we have $\mathcal{E}_b = \frac{\mu - \tilde{\mu}}{k}\, \mathcal{E}_u + \frac{\tilde{\mu}}{2k}$. Moreover since the mapping $u \mapsto u\, G$ is injective, the matrix G must have at least k distinct columns. Hence it must be $\mu - \tilde{\mu} \geq k$ and so $\mathcal{E}_b \geq \mathcal{E}_u$. □

Again, there is no energy reduction by applying the coding scheme. We observe though that the expected energy per bit here is less than or equal to the expected energy per bit when $x(0)$ is uniformly distributed in $\mathbb{Z}_2^n$ (expression (17)). This motivates us to get a better look at how the relation of adjacent columns of G influences the expected energy.

Let $I = \{1 \leq i \leq n : g_i \neq 0\}$. Since μ is the number of non-zero columns in the generator matrix G, the set I has exactly μ elements. Now, there exist some non-negative integers : q, $i_1, i_2, \cdots, i_q$ and $r_1, r_2, \cdots, r_q$ so that the elements of I written in increasing order are:

$$i_1, i_1+1, \cdots, i_1+r_1 \;<\; i_2, i_2+1, \cdots, i_2+r_2 \;<\; \cdots \;<\; i_q, i_q+1, \cdots, i_q+r_q$$

and also

$$
\begin{aligned}
& & g_{i_1} = g_{i_1+1} = \cdots = g_{i_1+r_1} & \neq g_{i_1+r_1+1} \\
g_{i_2-1} \neq & & g_{i_2} = g_{i_2+1} = \cdots = g_{i_2+r_2} & \neq g_{i_2+r_2+1} \\
g_{i_3-1} \neq & & g_{i_3} = g_{i_3+1} = \cdots = g_{i_3+r_3} & \neq g_{i_3+r_3+1} \\
& & \vdots & \\
g_{i_q-1} \neq & & g_{i_q} = g_{i_q+1} = \cdots = g_{i_q+r_q} &
\end{aligned}
$$

By definition of d_i's and d_i''s we have:

$$\mu = \sum_{i=1}^{n} d_i = (1+r_1) + (1+r_2) + \cdots + (1+r_q) \qquad \text{and} \qquad \tilde{\mu} = \sum_{i=1}^{n-1} d_i\, d_i' = r_1 + r_2 + \cdots + r_q$$

Suppose now that we replace the code $\mathcal{C}$ with an equivalent code $\mathcal{C}_P$ having generating matrix $G_P = G\,P$, where P is a permutation matrix. Although the two codes have exactly the same error correction properties, they may have different expected energy per bit. This is because the energy depends on the adjacency properties of the non-zero columns of matrix GP. We have the following Lemma.

Lemma 3.2 *The code $\mathcal{C}$ has the minimum expected energy per bit among all equivalent codes $\mathcal{C}_P$ if and only if for the set of indices of non-zero columns of G,*

$$
\begin{aligned}
I &= \{1 \le i \le n : g_i \neq 0\} \\
&= \{\, i_1, i_1+1, \cdots, i_1+r_1 \;<\; i_2, i_2+1, \cdots, i_2+r_2 \;<\; \cdots \;<\; i_q, i_q+1, \cdots, i_q+r_q \,\}
\end{aligned}
$$

with,

$$g_{i_1} = g_{i_1+1} = \cdots = g_{i_1+r_1} \;,\; g_{i_2} = g_{i_2+1} = \cdots = g_{i_2+r_2} \;,\; \cdots \;,\; g_{i_q} = g_{i_q+1} = \cdots = g_{i_q+r_q}$$

the columns $g_{i_1}, g_{i_2}, \cdots g_{i_q}$ are distinct (and $\neq 0$). In this case, the minimum energy per bit $\mathcal{E}_b^*$ is :

$$\mathcal{E}_b^* = \frac{q}{k}\,\mathcal{E}_u + \frac{\tilde{\mu}}{2k} \tag{31}$$

and q equals the number of distinct, non-zero columns of matrix G.

Proof: A permutation of the columns of G does not change μ but it may change $\tilde{\mu}$. Because of expression (30), we have $\overline{\mathcal{E}(x \to z)}^{x,z} = \mathcal{E}_u\,\mu - \lambda\,\tilde{\mu}$. We see that the energy is minimized when $\tilde{\mu}$ is maximized. This is achieved if and only if the conditions of the Lemma hold i.e., when identical columns are consecutive. $\square$

3.3 Linear Differential Coding with Arbitrary Initial State

The most general case of the initial state is examined here. The only assumption is that $x(0)$ *is independent of the input sequence* $u(1), u(2), \cdots$. We have the Theorem:

Theorem 3.3 *If the initial state $x(0)$ is arbitrarily distributed in $\mathbb{Z}_2^n$ and independent of the input sequence, then the expected energy per transmitted bit $\mathcal{E}_b$, using the linear differential coding scheme, is :*

$$\mathcal{E}_b = \frac{\mu - \tilde{\mu}}{k}\,\mathcal{E}_u + \frac{\tilde{\mu}}{2k} + \frac{\lambda}{k}\sum_{i=1}^{n-1}\left(1 + \overline{(-1)^{\,x_i(0)+x_{i+1}(0)}}\right) d_i\,d_i' \tag{32}$$

where μ, $\tilde{\mu}$, d_i and $\tilde{d}_i$ are as defined in Theorem 3.2.

The proof is similar to that of Theorem 3.2. It is based on the following Lemma that is a modification of Lemma 3.1.

Lemma 3.3 *Let x and z be two independent random vectors uniformly distributed in $\mathcal{C} = \{u\,G,\, u \in \mathbb{Z}_2^k\}$ and $\sigma = (\sigma_1, \sigma_2, \cdots, \sigma_n)$ is a given vector in $\mathbb{Z}_2^n$. Then, with μ, $\tilde{\mu}$, d_i and $\tilde{d}_i$ defined as in Lemma 3.1 we have:*

$$\overline{\mathcal{E}(x \oplus \sigma \to z \oplus \sigma)}^{x,z} = (\mu - \tilde{\mu})\,\mathcal{E}_u + \tilde{\mu}/2 + \lambda\sum_{i=1}^{n-1}\left[1 - (-1)^{\,\sigma_i+\sigma_{i+1}}\right] d_i\,d_i' \tag{33}$$

Proof: Setting again $w_i = x_i \oplus z_i$, $i = 1, 2, \cdots, n$, from expression (1) of the transition energy and equation (3) we have:

$$\begin{aligned}\mathcal{E}(x \oplus \sigma \to z \oplus \sigma) \;&=\; (1+2\lambda)\sum_{i=1}^{n}(z_i \oplus \sigma_i - x_i \oplus \sigma_i)^2 - \\ &\qquad 2\lambda\sum_{i=1}^{n-1}(z_i \oplus \sigma_i - x_i \oplus \sigma_i)(z_{i+1} \oplus \sigma_{i+1} - x_{i+1} \oplus \sigma_{i+1})\end{aligned}$$

$$
\begin{aligned}
&= (1+2\lambda)\sum_{i=1}^{n} z_i \oplus x_i - 2\lambda \sum_{i=1}^{n-1} (-1)^{x_i \oplus \sigma_i + x_{i+1} \oplus \sigma_{i+1}} \; (z_i \oplus x_i)(z_{i+1} \oplus x_{i+1}) \\
&= (1+2\lambda)\sum_{i=1}^{n} w_i - 2\lambda \sum_{i=1}^{n-1} (-1)^{x_i + \sigma_i + x_{i+1} + \sigma_{i+1}} \; w_i \, w_{i+1} \\
&= (1+2\lambda)\sum_{i=1}^{n} w_i - 2\lambda \sum_{i=1}^{n-1} (-1)^{\sigma_i + \sigma_{i+1}} \cdot (-1)^{x_i + x_{i+1}} \; w_i \, w_{i+1}
\end{aligned}
$$

Taking the expectation over x and w and using equations (27) and (28) we have:

$$
\begin{aligned}
\overline{\mathcal{E}(x \oplus \sigma \to z \oplus \sigma)}^{\,x,z} &= (1+2\lambda)\sum_{i=1}^{n} \overline{w_i} - 2\lambda \sum_{i=1}^{n-1} (-1)^{\sigma_i + \sigma_{i+1}} \cdot \overline{(-1)^{x_i \oplus x_{i+1}}}^{\,x} \cdot \overline{w_i \, w_{i+1}}^{\,w} \\
&= (1+2\lambda)\sum_{i=1}^{n} \frac{d_i}{2} - \lambda \sum_{i=1}^{n-1} (-1)^{\sigma_i + \sigma_{i+1}} \, d_i \, d'_i \\
&= \frac{1+2\lambda}{2} \sum_{i=1}^{n} d_i - \lambda \sum_{i=1}^{n-1} d_i \, d'_i + \lambda \sum_{i=1}^{n-1} d_i \, d'_i - \lambda \sum_{i=1}^{n-1} (-1)^{\sigma_i + \sigma_{i+1}} \, d_i \, d'_i \\
&= \mathcal{E}_u \, \mu - (\mathcal{E}_u - \frac{1}{2}) \, \tilde{\mu} + \lambda \sum_{i=1}^{n-1} \left[1 - (-1)^{\,\sigma_i + \sigma_{i+1}} \right] d_i \, d'_i \\
&= (\mu - \tilde{\mu}) \, \mathcal{E}_u + \tilde{\mu}/2 + \lambda \sum_{i=1}^{n-1} \left[1 - (-1)^{\,\sigma_i + \sigma_{i+1}} \right] d_i \, d'_i \qquad \square
\end{aligned}
$$

Proof of Theorem 3.3: The recursion $x(r) = x(r-1) \oplus u(r)G$ implies that $x(r) = x(0) \oplus [u(1) \oplus u(2) \oplus \cdots \oplus u(r)]\, G$. Therefore $x(r)$ can be written as $x(r) = x(0) \oplus x^r$ and $x(r-1)$ can be written as $x(r) = x(0) \oplus x^{r-1}$ with x^r and x^{r-1} being independent and uniformly distributed in $\mathcal{C}$. Since the expected energy per bit is :

$$
\mathcal{E}_b = \overline{\mathcal{E}(x(r-1) \to x(r))}^{\,x(r-1),x(r)} = \overline{\mathcal{E}(x^{r-1} \oplus x(0) \to x^r \oplus x(0))}^{\,x^{r-1},x^r}
$$

an application of Lemma 3.3 completes the proof. $\square$

4 Further Definitions and Properties of Transitional Energy

In order to derive the energy properties of convolutional and block linear coding in the follow sections, we need some more properties of the transition energy.

Definition 4.1 *Let G and H be two $k \times n$ matrices with columns $g_1, g_2, \cdots, g_n$ and $h_1, h_2, \cdots, h_n$ respectively. We define the relative index $I(G,H)$ of the two matrices to be the number of distinct, non-common and non-zero columns between them. Formally it is :*

$$I(G,H) = \frac{|\{0, g_1, g_2, \cdots, g_n\} - \{0, h_1, h_2, \cdots, h_n\}| + |\{0, h_1, h_2, \cdots, h_n\} - \{0, g_1, g_2, \cdots, g_n\}|}{2}$$

The results of this section are given by the following Theorems.

Theorem 4.1 *Let u be a random variable, uniformly distributed in $\mathbb{Z}_2^k$ and let $x = uG$, $z = uH$ for some binary $k \times n$ matrices G and H. Then :*

$$\overline{\mathcal{E}(x \to z)}^{x,z} \geq \mathcal{E}_u \cdot I(G,H) \tag{34}$$

Note that $I(G,H) = I(H,G)$ and so the same lower bound is given by the Theorem for both $\overline{\mathcal{E}(x \to z)}^{x,z}$ and $\overline{\mathcal{E}(z \to x)}^{x,z}$. Also note that $I(G,H)$ is a lower bound of the number, call it ξ, of indices $i = 1, 2, \cdots, n$ for which $g_i \neq h_i$. The bound given by the Theorem is intuitively justified since in a sense we expect that $\overline{\mathcal{E}(x \to z)}^{x,z} \geq \mathcal{E}_u \xi$. In the following sections we may allow some violation of notation and write $I(x,z)$ instead of $I(G,H)$ when it is clear that x and z satisfy $x = uG$ and $z = uH$ respectively.

Definition 4.2 *Let G be a $k \times n$ matrix with columns $g_1, g_2, \cdots, g_n$ and H be a $m \times n$ matrix with columns $h_1, h_2, \cdots, h_n$. We define the relative index $\tilde{I}(G,H)$ of the two matrices to be the one half of the number of their non-zero columns, that is :*

$$\tilde{I}(G,H) = \frac{|\{g_1, g_2, \cdots, g_n\} - \{0\}| + |\{h_1, h_2, \cdots, h_n\} - \{0\}|}{2}$$

Theorem 4.2 *Let u and v be independent random variables, uniformly distributed in $\mathbb{Z}_2^k$ and $\mathbb{Z}_2^m$ respectively. Let $x = uG$ and $z = uH$ for some binary $k \times n$ and $m \times n$ matrices G and H. Then:*

$$\overline{\mathcal{E}(x \to z)}^{x,z} \geq \mathcal{E}_u \cdot \tilde{I}(G, H) \tag{35}$$

Proof of Theorem 4.2: We can define the concatenated random vector $\widetilde{u} = [u, v]$ that is uniformly distributed in $\mathbb{Z}_2^{k+m}$ and the matrices

$$\tilde{G} = \begin{bmatrix} G \\ 0 \end{bmatrix} \quad \text{and} \quad \tilde{H} = \begin{bmatrix} 0 \\ H \end{bmatrix} \tag{36}$$

Then we have $x = \tilde{u}\tilde{G}$ and $z = \tilde{u}\tilde{H}$ and we can apply Theorem 4.1 directly. Observe that, $I(\tilde{G}, \tilde{H}) = \tilde{I}(G, H)$ $\square$

In order to prove Theorem 4.1 we need a sequence of Lemmas examining in detail the "interaction" between adjacent lines in the bus. We start by examining the product $(y_i - x_i)(y_{i+1} - x_{i+1})$ that is involved in the energy required for the transition from x to y.

Lemma 4.1 *Let u be a random variable, uniformly distributed in $\mathbb{Z}_2^k$ and $x = uG$, $y = uH$ for some binary $k \times n$ matrices G and H with columns $g_1, g_2, \cdots, g_n$ and $h_1, h_2, \cdots, h_n$ respectively. Then for every $i = 1, 2, \cdots, n-1$ the expression*

$$\Delta_i = \overline{(y_i - x_i)(y_{i+1} - x_{i+1})}^{x,y} \tag{37}$$

takes the values:

$$\Delta_i = -\frac{1}{4} \text{ if } \left\{ \begin{array}{ccc} y_i & & y_{i+1} \\ \nparallel & & \nparallel \\ x_i & \neq & x_{i+1} \end{array} \right\} \text{ and } \left\{ \begin{array}{ccc} x_i & \neq & y_{i+1} \\ x_{i+1} & = & y_i \end{array} \right\} \tag{38}$$

$$\Delta_i = -\frac{1}{4} \text{ if } \left\{ \begin{array}{ccc} y_i & & y_{i+1} \\ \nparallel & & \nparallel \\ x_i & \neq & x_{i+1} \end{array} \right\} \text{ and } \left\{ \begin{array}{ccc} x_i & = & y_{i+1} \\ x_{i+1} & \neq & y_i \end{array} \right\} \tag{39}$$

$$\Delta_i = -\frac{1}{2} \text{ if } \left\{ \begin{array}{ccc} y_i & & y_{i+1} \\ \nparallel & & \nparallel \\ x_i & \neq & x_{i+1} \end{array} \right\} \text{ and } \left\{ \begin{array}{ccc} x_i & = & y_{i+1} \\ x_{i+1} & = & y_i \end{array} \right\} \tag{40}$$

$$\Delta_i = \frac{1}{4} \text{ if } \left\{ \begin{array}{ccc} y_i & \neq & y_{i+1} \\ \nparallel & & \nparallel \\ x_i & = & x_{i+1} \end{array} \right\} \quad - \tag{41}$$

$$\Delta_i = \frac{1}{2} \text{ if } \left\{ \begin{array}{ccc} y_i & = & y_{i+1} \\ \nparallel & & \\ x_i & = & x_{i+1} \end{array} \right\} \text{ and } \left\{ \begin{array}{ccc} x_i & \neq & 0 \\ x_{i+1} & \neq & 0 \end{array} \right\} \tag{42}$$

$$\Delta_i = 0 \quad \text{otherwise} \tag{43}$$

where we write $x_i = x_j$ *if and only* $g_i = g_j$*, similarly for the rest of the relations between* x_i*,* x_{i+1}*,* y_i*,* y_{i+1} *and* 0*.*

Proof: We can write (Equation (3)), that:

$$(y_i - x_i)(y_{i+1} - x_{i+1}) = (y_i \oplus x_i)(y_{i+1} \oplus x_{i+1})(-1)^{x_i \oplus x_{i+1}}. \tag{44}$$

We know from Lemma 2.4 that two linear combinations (in $\mathbb{Z}_2$) of the random variables $x_1, x_2, \cdots, x_n$, $y_1, y_2, \cdots, y_n$ are either statistically independent or identical.

We observe from (44) that if the random variable $x_i \oplus x_{i+1}$ is not (identically) zero and it is independent of $x_i \oplus y_i$ and $x_{i+1} \oplus y_{i+1}$ then the expected value of the right part of (44) is zero since $\overline{(-1)^{x_i \oplus x_{i+1}}} = 0$.

In the following first three cases we examine Δ_i when $x_i \oplus x_{i+1}$ is non-zero and dependent on $x_i \oplus y_i$ or $x_{i+1} \oplus y_{i+1}$. In the fourth and fifth cases we examine Δ_i when $x_i \oplus x_{i+1} = 0$.

Case 1 : $x_i \oplus x_{i+1} = x_i \oplus y_i$ and $x_i \oplus x_{i+1} \neq x_{i+1} \oplus y_{i+1}$. Taking the expectation on Equation (44) we have that:

$$\Delta_i = \overline{(x_i \oplus x_{i+1})(-1)^{x_i \oplus x_{i+1}}} \cdot \overline{y_{i+1} \oplus x_{i+1}} \tag{45}$$

Therefore we have $\Delta_i = -1/4$ if $x_i \oplus x_{i+1} \neq 0$ and $y_{i+1} \oplus x_{i+1} \neq 0$, which is the case of Equation (38), and $\Delta_i = 0$ otherwise.

Case 2 : $x_i \oplus x_{i+1} \neq x_i \oplus y_i$ and $x_i \oplus x_{i+1} = x_{i+1} \oplus y_{i+1}$. Now Equation (44) implies that:

$$\Delta_i = \overline{y_i \oplus x_i} \cdot \overline{(x_i \oplus x_{i+1})(-1)^{x_i \oplus x_{i+1}}} \tag{46}$$

Again, we have $\Delta_i = -1/4$ if $x_i \oplus x_{i+1} \neq 0$ and $y_i \oplus x_i \neq 0$, which is the case of Equation (39) and $\Delta_i = 0$ otherwise.

Case 3 : $x_i \oplus x_{i+1} = x_i \oplus y_i$ and $x_i \oplus x_{i+1} = x_{i+1} \oplus y_{i+1}$. Then Equation (44) gives:

$$\Delta_i = \overline{(x_i \oplus x_{i+1})(-1)^{x_i \oplus x_{i+1}}} \tag{47}$$

and so we have $\Delta_i = -1/2$ if $x_i \oplus x_{i+1} \neq 0$, that is the case of Equation (40), and $\Delta_i = 0$ otherwise.

Now we calculate the expected value of (44) when $x_i \oplus x_{i+1} = 0$. We have to distinguish between the following two cases:

Case 4 : $x_i \oplus x_{i+1} = 0$ and $y_i \oplus x_i \neq y_{i+1} \oplus x_{i+1}$. Then from (44) we get:

$$\Delta_i = \overline{(y_i \oplus x_i)} \cdot \overline{(y_{i+1} \oplus x_{i+1})} \tag{48}$$

hence it is $\Delta_i = 1/4$ if $y_i \oplus x_i \neq 0$ and $y_{i+1} \oplus x_{i+1} \neq 0$, i.e. the case of Equation (41), and $\Delta_i = 0$ otherwise.

Case 5 : $x_i \oplus x_{i+1} = 0$ and $y_i \oplus x_i = y_{i+1} \oplus x_{i+1}$. Then from (44) we have:

$$\Delta_i = \overline{(y_i \oplus x_i)} \tag{49}$$

and so, $\Delta_i = 1/2$ if $y_i \oplus x_i \neq 0$, which is the case of Equation (42), and $\Delta_i = 0$ otherwise. This completes the proof of the Lemma. $\square$

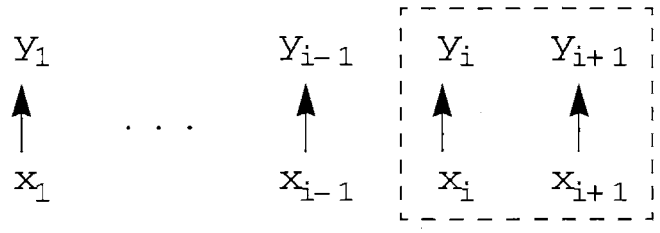

Figure 2: Energy accumulation with i

Now suppose that the bus has only the lines $1, 2, \cdots, i$ and we add line $i+1$ as it is illustrated below:

Then the expected energy during the transition $x \rightarrow y$ increases by the amount of Θ_i defined by the expression:

$$\Theta_i \equiv (1+2\lambda)\,\overline{(y_{i+1}-x_{i+1})^2} - 2\lambda\overline{(y_i-x_i)(y_{i+1}-x_{i+1})} \tag{50}$$

Note that $(1+2\lambda)\,(y_{i+1}-x_{i+1})^2 - 2\lambda(y_i-x_i)(y_{i+1}-x_{i+1}) \geq 0$ for all possible values of x_i,x_{i+1},y_i,y_{i+1} and so it is $\Theta_i \geq 0$. The following Lemma provides a better lower bound for Θ_i.

Lemma 4.2 *With the setting of Lemma 4.1 we have that for every $i = 1, 2, \cdots, n-1$ the expected additional energy for the $i+1$ line is bounded as:*

$$\Theta_i \geq \mathcal{E}_u \frac{I_i^x + I_i^y}{2} \tag{51}$$

where

$$I_i^x = \begin{cases} 1 & \text{if } x_{i+1} \notin \{0, x_i, y_i, y_{i+1}\} \\ 0 & \text{otherwise} \end{cases} \tag{52}$$

and

$$I_i^y = \begin{cases} 1 & \text{if } y_{i+1} \notin \{0, x_i, y_i, x_{i+1}\} \\ 0 & \text{otherwise} \end{cases} \tag{53}$$

Proof: If $I_i^x = I_i^y = 0$ then relation (51) is true because $\Theta_i \geq 0$. Now suppose that $I_i^x = 0$ and $I_i^y = 1$. The relation $I_i^y = 1$ implies first that $y_{i+1} \neq x_{i+1}$ and so $\overline{(y_{i+1}-x_{i+1})^2} = 1/2$

and second that $y_{i+1} \neq y_i$ which gives $\overline{(y_i - x_i)(y_{i+1} - x_{i+1})} \leq 1/4$ because of Lemma 4.1. Therefore we have that $\Theta_i \geq \frac{1+2\lambda}{4} = \frac{\mathcal{E}_u}{2}$ as desired. The same argument holds for the case $I_i^x = 1$ and $I_i^y = 0$. Suppose now that $I_i^x = I_i^y = 1$. Then it must be $y_{i+1} \neq x_{i+1}$ which implies $\overline{(y_{i+1} - x_{i+1})^2} = 1/2$ and also $x_{i+1} \neq x_i$ and $y_{i+1} \neq y_i$ which imply that $\overline{(y_i - x_i)(y_{i+1} - x_{i+1})} \leq 0$ by Lemma 4.1. Combining the two results we have $\Theta_i \geq \frac{1+2\lambda}{2} = \mathcal{E}_u$ and the Lemma has been proved. $\square$

Proof of Theorem 4.1: For $i = 1, 2, \cdots, n$ we define the partial vectors $x^i = (x_1, x_2, \cdots, x_i)$ and $y^i = (y_1, y_2, \cdots, y_i)$ as well as the partial relative index :

$$I^i = I\left([g_1, g_2, \cdots, g_i], [h_1, h_2, \cdots, h_i]\right)$$

The energy function $\mathcal{E}(x^i \rightarrow y^i)$ is defined to be the energy $\mathcal{E}([x^i, 0, \cdots, 0] \rightarrow [y^i, 0, \cdots, 0])$ where $n - i$ zeros have been added at the end of each of the vectors x^i and y^i. Therefore it is

$$\mathcal{E}(x^i \rightarrow y^i) = (1 + 2\lambda) \sum_{r=1}^{i} (y_r - x_r)^2 - 2\lambda \sum_{r=1}^{i-1} (y_r - x_r)(y_{r+1} - x_{r+1}) \tag{54}$$

The recursive relation follows immediately,

$$\begin{aligned} \overline{\mathcal{E}(x^{i+1} \rightarrow y^{i+1})} &= \overline{\mathcal{E}(x^i \rightarrow y^i)} + (1 + 2\lambda)\,\overline{(y_{i+1} - x_{i+1})^2} - 2\lambda \overline{(y_i - x_i)(y_{i+1} - x_{i+1})} \\ &= \overline{\mathcal{E}(x^i \rightarrow y^i)} + \Theta_i \end{aligned} \tag{55}$$

As we have mentioned above, we don't distinguish between a random variable x_i and the vector in matrix G or H generating it. Now we apply induction. For $i = 1$ we have,

$$\begin{aligned} \overline{\mathcal{E}(x^1 \rightarrow y^1)} &= (1 + 2\lambda)\,\overline{(y_1 - x_1)^2} \\ &\geq (1 + 2\lambda) \cdot \frac{|\{0, x_1\} - \{0, y_1\}| + |\{0, y_1\} - \{0, x_1\}|}{4} \\ &= \frac{1 + 2\lambda}{2} \cdot \frac{|\{0, x_1\} - \{0, y_1\}| + |\{0, y_1\} - \{0, x_1\}|}{2} \\ &= \mathcal{E}_u \cdot I^1 \end{aligned}$$

Now, suppose that for some $i < n$, it is, $\overline{\mathcal{E}(x^i \rightarrow y^i)} \geq \mathcal{E}_u \cdot I^i$. Using Lemma 4.2 and Equation (55) we have that

$$\begin{aligned} \overline{\mathcal{E}(x^{i+1} \rightarrow y^{i+1})} &= \overline{\mathcal{E}(x^i \rightarrow y^i)} + \Theta_i \\ &\geq \mathcal{E}_u \cdot \left[I^i + \frac{I_i^x + I_i^y}{2} \right] \end{aligned}$$

It is straight forward to verify that $I^i + (I_i^x + I_i^y)/2 \geq I^{i+1}$. This completes the induction step. Since it is, $I^n = I(G, H)$, the Lemma is proved. $\square$

Now we have the necessary tools to proceed in the analysis of more complex coding schemes for energy reduction.

5 Linear Convolutional Coding

In this section we study the energy properties of linear convolutional coding applied in DSM buses. The structure of the encoder is shown below. As before, the input vectors $u(r)$

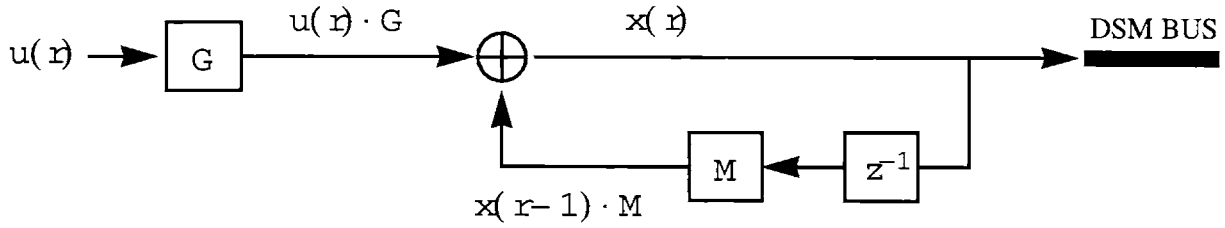

Figure 3: Convolutional Coding

are assumed to be uniformly distributed in $\mathbb{Z}_2^k$ and independent of each other. The vectors $x(r)$ that are transmitted through the bus belong to $\mathbb{Z}_2^n$. The coding schemes has rate k/n, equivalently, matrix G is of full rank. The variables satisfy the relation:

$$x(r) = x(r-1)\,M \oplus u(r)\,G \tag{56}$$

For the initial state $x(0)$ of the bus we make the assumption that it is either zero or uniformly distributed in $\mathbb{Z}_2^n$. The expected energy per transmitted bit is defined as usual, $\mathcal{E}_b = \overline{\mathcal{E}\,(x(r-1) \to x(r))}/k$ and the main result of this section is given by the Theorem:

Theorem 5.1 *The expected energy per transmitted bit $\mathcal{E}_b$, using the linear convolutional coding scheme presented above, is equal to or greater than $\mathcal{E}_u$.*

Proof: Starting from the recursive relation (56), for every $r > 1$ we have that:

$$x(r) = u(r)G \oplus u(r-1)GM \oplus \cdots \oplus u(1)GM^{r-1} \oplus x(0)M^r \tag{57}$$

$$x(r-1) = u(r-1)G \oplus u(r-2)GM \oplus \cdots \oplus u(1)GM^{r-2} \oplus x(0)M^{r-1} \tag{58}$$

If we define the composite random vector $v = [u(r), u(r-1), \cdots, u(1), x(0)]$ and the two matrices,

$$A = \begin{bmatrix} G \\ GM \\ GM^2 \\ \vdots \\ GM^{r-1} \\ M^r \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 0 \\ G \\ GM \\ \vdots \\ GM^{r-2} \\ M^{r-1} \end{bmatrix} \tag{59}$$

then $x(r)$ and $x(r-1)$ can be written as $x(r) = v\,A$ and $x(r-1) = v\,B$ respectively. Theorem 4.1 applies directly and gives:

$$\overline{\mathcal{E}\left(x(r-1) \rightarrow x(r)\right)} \geq \mathcal{E}_u \cdot I(A, B) \tag{60}$$

Note that matrix G is of full rank and so it has at least k non-zero distinct columns. Therefore matrices A and B differ in at least k non-zero columns, i.e. $I(A, B) \geq k$. This along with equation (60) proves the Theorem. □

6 Linear Coding

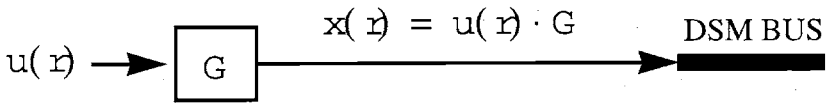

Figure 4: Linear Coding

Linear coding with $x(r) = u(r)G$ is obviously a special case of the convolutional coding with $M = 0$ or the differential coding with $x(0) \in \{u\,G,\, u \in \mathbb{Z}_2^k\}$. Therefore we can apply Theorem 3.2 directly and have that:

$$\mathcal{E}_b = \frac{\mu - \tilde{\mu}}{k}\,\mathcal{E}_u + \frac{\tilde{\mu}}{2k} \tag{61}$$

where μ is the number of non-zero columns of matrix G and $\tilde{\mu}$ is the number of indices $i = 1, 2, \cdots, n-1$ for which $g_i = g_{i+1} \neq 0$. Since $\mu - \tilde{\mu} \geq k$ and $\tilde{\mu} \geq 0$ it is always $\mathcal{E}_b \geq \mathcal{E}_u$.

The result of Lemma 3.2, on the optimality of a code $\mathcal{C}$, among its class of equivalent codes, applies directly to the case of linear coding. This means that given a generator matrix G, we can find a permutation matrix P so that the code $\{uGP,\, u \in \mathbb{Z}_2^k\}$ is optimal in the above sense and the expected energy per bit is given by expression (31).

7 Block Linear Coding

Here we study the energy performance of block linear coding in DSM buses. The structure of the encoder is shown below with $q = 0, 1, \ldots$. The bus has n parallel lines as before and the

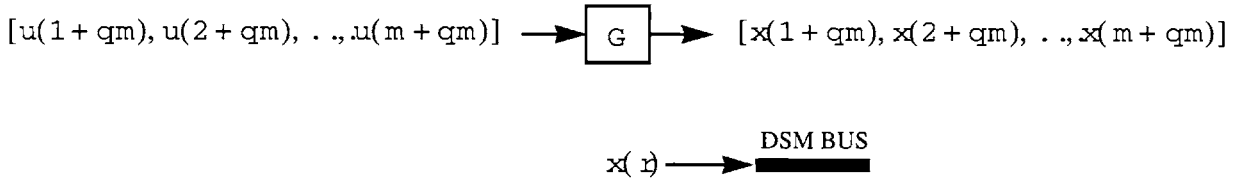

Figure 5: Linear Coding

code is of rate k/n. The data is an i.i.d. sequence $\{u(r)\}_r$ of vectors uniformly distributed in $\mathbb{Z}_2^k$. Blocks $[u(1), u(2), \cdots, u(m)]$ of m successive input vectors are encoded into blocks of m bus vectors, $[x(1), x(2), \cdots, x(m)]$, with $x(r) \in \mathbb{Z}_2^n$ for every r, in a linear fashion:

$$[x(1), x(2), \cdots, x(m)] = [u(1), u(2), \cdots, u(m)]\, \mathbf{G} \tag{62}$$

The matrix $\mathbf{G}$ has dimensions $mk \times mn$. The expected cost per transmitted bit using this block coding is:

$$\mathcal{E}_b^{LB} = \frac{1}{mk} \sum_{r=1}^{m} \overline{\mathcal{E}(x(r) \rightarrow x(r+1))} \tag{63}$$

We have the following Theorem :

Theorem 7.1 *The expected energy per transmitted bit using linear block coding is equal to or greater than that of the uncoded bus, i.e.:* $\mathcal{E}_b^{LB} \geq \mathcal{E}_u$.

So, as in the previous coding schemes, the linear block coding cannot reduce the power dissipation in DSM buses. For the proof of the Theorem we need to relate the expected energy to the rank of matrix $\mathbf{G}$. This is done using Theorems 4.1 and 4.2 from Section 4.

Proof of Theorem 7.1 : The estimation of the expected energy using the block coding scheme

$$[x(1), x(2), \cdots, x(m)] = [u(1), u(2), \cdots, u(m)]\,\mathbf{G}$$

requires us to examine the columns of matrix $\mathbf{G}$. We decompose the generator matrix as $\mathbf{G} = [G^1, G^2, \cdots, G^m]$ where each sub-matrix G^i is of size $km \times n$. Then if we set $v = [u(1), u(2), \cdots, u(m)]$ we can write $x(1) = vG^1$, $x(2) = vG^2, \cdots, x(m) = vG^m$. Note that v is uniformly distributed in $\mathbb{Z}_2^{km}$ and so Theorem 4.1 can be applied to give for $i = 1, 2, \cdots, n-1$,

$$\overline{\mathcal{E}(x(i) \rightarrow x(i+1))} \geq \mathcal{E}_u \cdot I(G^i, G^{i+1}) \tag{64}$$

Using similar reasoning and Theorem 4.2 we can conclude also that:

$$\overline{\mathcal{E}(x(m) \rightarrow x(m+1))} \geq \mathcal{E}_u \cdot \tilde{I}(G^m, G^1) \tag{65}$$

We can write $G^i = [g_1^i, g_2^i, \cdots, g_n^i]$, with g_j^i being a column vector and for every $i = 1, 2, \cdots, m$. We define the set of non-zero columns of G^i,

$$S^i = \{g_1^i, g_2^i, \cdots, g_n^i\} - \{0\}$$

Note that by definition, for every $i = 1, 2, \cdots, n-1$, the index $I(G^i, G^{i+1})$, which equals the number of distinct, non-common and non-zero columns of G^i and G^{i+1}, is:

$$I(G^i, G^{i+1}) = \frac{|S^{i+1} - S^i| + |S^i - S^{i+1}|}{2}.$$

From inequality (64) the expected cost of the sequence of transitions $x(1) \rightarrow x(2) \rightarrow \cdots \rightarrow x(m)$ is given by:

$$\begin{aligned} \overline{\mathcal{E}(x(1) \rightarrow x(2) \rightarrow \cdots \rightarrow x(m))} &= \sum_{i=1}^{m-1} \overline{\mathcal{E}(x(i) \rightarrow x(i+1))} \\ &\geq \mathcal{E}_u \cdot \sum_{i=1}^{m-1} \frac{|S^{i+1} - S^i| + |S^i - S^{i+1}|}{2} \end{aligned} \tag{66}$$

Regarding the transition $x(m) \rightarrow x(m+1)$ we observe that by definition it is:

$$\tilde{I}(G^m, G^1) = \frac{|S^1| + |S^m|}{2}$$

and from (65) we have:

$$\overline{\mathcal{E}(x(m) \rightarrow x(m+1))} \geq \mathcal{E}_u \cdot \frac{|S^m| + |S^1|}{2} \tag{67}$$

Combining inequalities (66) and (67) we conclude that:

$$\begin{aligned} \overline{\mathcal{E}(x(1) \rightarrow x(2) \rightarrow \cdots \rightarrow x(m+1))} &\geq \frac{\mathcal{E}_u}{2} \cdot \left[|S^1| + \sum_{i=1}^{m-1} |S^{i+1} - S^i| + |S^m| + \sum_{i=1}^{m-1} |S^i - S^{i+1}| \right] \\ &\geq \mathcal{E}_u \cdot |S^1 \cup S^2 \cup \cdots \cup S^m| \end{aligned}$$

Using definition (63) we have that

$$\begin{aligned} \mathcal{E}_b^{LB} &= \frac{1}{mk} \sum_{i=1}^{m} \overline{\mathcal{E}(x(i) \rightarrow x(i+1))} \\ &\geq \mathcal{E}_u \cdot \frac{|S^1 \cup S^2 \cup \cdots \cup S^m|}{mk} \end{aligned}$$

Now, note that $|S^1 \cup S^2 \cup \cdots \cup S^m|$ is the number of distinct non-zero columns of the generating matrix $\mathbf{G}$. Since $\mathbf{G}$ has full row rank, equal to mk, we conclude that $|S^1 \cup S^2 \cup \cdots \cup S^m| \geq mk$ and so $\mathcal{E}_b^{LB} \geq \mathcal{E}_u$ which proves the Theorem. $\square$

8 Conclusions

The presented analysis suggests that differential, convolutional, linear and block linear coding are not appropriate for power reduction. If we need to use a differential or linear coding for error correction then Lemma 3.2 provides the conditions for the code to be optimal among its class of equivalent codes. A variation of 3.2 can provide the optimality conditions of the case of block linear codes.

References

[1] V. Agarwal, M. Hrishikesh, S. Keckler, D. Burger, "Clock rate versus IPC: the end of the road for conventional microarchitectures", *Proceedings of the 27th Int. Symp. on Computer Architecture*, 2000, pp. 248-259.

[2] M. Artin, *Algebra*, Prentice Hall, 1991.

[3] E. Fujiwara, T. Rao, *Error-control coding for computer systems*, Prentice Hall, 1989.

[4] F. MacWilliams and N. Sloane, *The theory of error correcting codes*, Elsevier/North-Holland, 1978.

[5] L. Vermani, *Elements of algebraic coding theory*, London, New York : Chapman & Hall, 1996.

Chapter 5

Transition Pattern Coding (TPC) For Power reduction

1. Introduction

In Chapter 2 we introduced an energy model for buses that captures the parasitic elements appearing in modern deep sub-micron technologies. In Chapter 3 we developed theoretical tools that revealed the fundamental information theoretic limits of the maximum possible energy reduction achievable using coding. Combining the energy bus model and the theoretical tools we were able to calculate exactly the limits of coding in the case of buses.

The results of Chapter 3 (those for example in Figures 5,6 and 7) demonstrate significant energy reduction for relatively small reduction of the information rate. In establishing these limits we did not assume any kind of constraints in the encoder and decoder with respect to memory, complexity, self energy consumption or latency that they may introduce. All these issues are important for a practical system. The encoder and decoder must have the minimum possible complexity, minimum memory (except in some special sub-circuits that are tolerant to latency) and occupy the minimum possible area in the chip.

From a traditional communication perspective, the first choice for simple coding would be the obvious one; try linear or differential coding. We analyzed this in Chapter 4 and showed that no energy reduction is possible with such schemes. Therefore, the coding schemes needed should be non-linear and, for most practical purposes, they should have memory of order one. This class of coding schemes is the topic of the present Chapter 5. We have termed this class of schemes, the *Transition Pattern Coding schemes* (*TPC*). The name is motivated by the fact that what is transmitted through the bus at time t depends only on what was transmitted at time $t-1$, and the new data vector arrived at the encoder at time t. In other words, the *TPC* schemes are *Finite State Machines* (*FSM*s) whose purpose is to minimize the energy that is lost on each particular transition. The constraint that no future values of the data vectors are available in the encoder implies that these schemes cannot achieve the fundamental limits developed in Chapter 3.

To illustrate the gap between the energy reduction of the class of TPC schemes and the class of all coding schemes, we state without proof that the best energy reduction achievable by TPC schemes is given by the Entropy Bound of Figure 1 in Chapter 3, while the best of general schemes is given by Theorem 4.3 of the same Chapter (for some special cases, by the Figures 5,6 and 7).

From a practical point of view, the class of TPC schemes contains almost all coding schemes that have been proposed in the literature, for example those in [8]-[17] as well as more recent schemes appropriate for deep sub-micron buses like that in [21].

In the present chapter we present a complete mathematical treatment of TPC schemes. We start by providing a formal definition and parametrization of them. Then we derive closed formulas for the energy reduction they achieve. A novel algorithm, called the TPC algorithm, is presented that approximates the energy-wise optimal TPC scheme given the input statistics and bus energy behavior. In other words, given the statistical properties of the input data and the bus structure, we can derive the most appropriate TPC scheme. A simplified version of the algorithm is also provided that achieves very good results for buses with seven or more lines. Finally, in the last section of the Chapter, the issue of complexity is addressed by partitioning of the bus into blocks and encoding them independently. A detailed analysis of the energy lost in the boundaries of the individual blocks, because of the coupling between their boundary lines, is presented.

It is interesting to mention that there is no practical limit on energy reduction achieved using TPC schemes. This depends directly on the data statistics and bus structure. For example, in Section 5.3 of the present Chapter, possible energy reduction of up to 80% is demonstrated. This is the great advantage of the approach here compared to other known "fixed" coding schemes! The design methodology presented here is data-distribution and bus-structure driven. Moreover the coding scheme design, using the TPC algorithm, also demonstrates an interesting behavior with respect to the coupling between the lines, captured by parameter λ. For stronger capacitive coupling between neighboring lines, i.e. higher values of parameter λ, we get schemes with better energy reduction performance! This is in contrast to other proposed coding schemes whose performance deteriorates with λ. A comparison is presented in Section 4.3.

Finally, it is stressed that the TPC theory is independent of any energy model. The TPC methodology is general and can be used as an energy reduction approach to any communication channel that has the following properties: it is memory-less regarding the data transmission and it has memory of order one regarding the energy consumption.

2. Notation and Comments

Although Transition Pattern Coding has been evolved in an independent methodology, its birth was triggered by the need to design energy reduction schemes for the new deep sub-micron technology, buses. As it was discussed in detail in Chapter 2, the problem with the modern technologies is the strong coupling between neighboring lines in the bus.

We use the bus of Figure 7 in Chapter 1 as our guide to demonstrate the theory and the philosophy behind the TPC approach. The bus has n lines, and we assume that the fringing capacitance C_F is negligible and allow for different values of λ. As an energy measure we use the energy drawn from the power supply that is given by expression (22) in Chapter 2 and shown below for convenience.

$$E = (\boldsymbol{V}^{new})^T \cdot \boldsymbol{C}^{ta} \cdot (\boldsymbol{V}^{new} - \boldsymbol{V}^{old}) \tag{1}$$

The vectors V^{new}, V^{old} are the *old* and the *new* vectors of the values of the lines in the bus:

$$V^{old} = \begin{bmatrix} V_1^{old} \\ V_2^{old} \\ \vdots \\ V_n^{old} \end{bmatrix}, \quad V^{new} = \begin{bmatrix} V_1^{new} \\ V_2^{new} \\ \vdots \\ V_n^{new} \end{bmatrix}$$

and take the values 0 or V_{dd}. The matrix C^{ta} is given by:

$$C^{ta} = \begin{bmatrix} 1+\lambda & -\lambda & 0 & \dots & 0 & 0 \\ -\lambda & 1+2\lambda & -\lambda & \dots & 0 & 0 \\ 0 & -\lambda & 1+2\lambda & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & . & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1+2\lambda & -\lambda \\ 0 & 0 & 0 & \dots & -\lambda & 1+\lambda \end{bmatrix} \cdot C_L \qquad (2)$$

where C_L is the line-to-ground capacitance. Recall that λ is non-negative and for $0.13\mu m$ technologies it can be as high as eight. As an example application of formula (1), Table 1 shows the energy drawn from V_{dd} during the transition $[V_1^i, V_2^i] \rightarrow [V_1^f, V_2^f]$ for a two line bus.

Trans. Energy $C_L \cdot V_{dd}^2$		$[V_1^f, V_2^f]$			
		00	01	10	11
$[V_1^i, V_2^i]$	00	0	$1+\lambda$	$1+\lambda$	2
	01	0	0	$1+2\lambda$	1
	10	0	$1+2\lambda$	0	1
	11	0	λ	λ	0

Table 1: Energy drawn from Vdd for the example bus with two lines.

3. Transition Pattern Coding Schemes

In this section we describe a general class of bus coding schemes. Although many of the techniques proposed in the literature belong to this class, our unifying approach is new and results in exact closed form answers for the energy dissipation and power savings with respect to uncoded data transmission.

In Figure 1 we see the class of Transition Pattern Coding Schemes. They are time invariant and have the property that the data, input vector $\boldsymbol{D}(k) = [d_1(k), d_2(k), \ldots, d_m(k)]^T$ is, encoded, transmitted, and decoded "instantaneously", in the sense that the data is available at the output of the receiver during the same clock cycle k it is fed into the encoder.

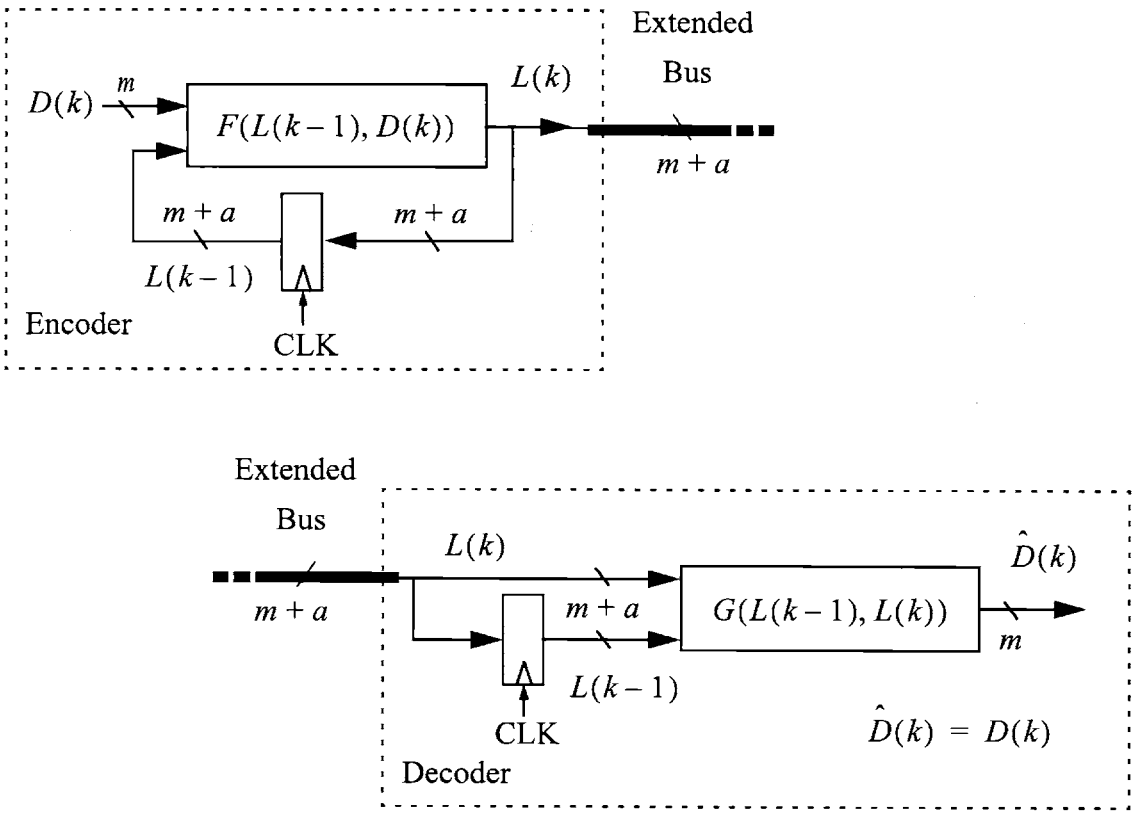

Figure 1: The Transition Pattern Coding Schemes

Note that the "original" data bus with m lines has been expanded to a bus of $m+a$ lines. The output of the decoder, *state vector of the bus,* $\boldsymbol{L}(k) = [l_1(k), l_2(k), \ldots, l_{m+a}(k)]^T$ contains the *logical values* of these $m+a$ lines, i.e. $\boldsymbol{L}(k) \in \{0, 1\}^{m+a}$. The following relation is true for every k:

$$\boldsymbol{L}(k) = F(\boldsymbol{L}(k-1), \boldsymbol{D}(k)) \tag{3}$$

The vector $\boldsymbol{L}(k)$ depends on both the input vector as well as the previous state $\boldsymbol{L}(k-1)$ of the bus. Note that the information of the previous state is important since the energy dissipation in the bus is due to the transition $\boldsymbol{L}(k-1) \rightarrow \boldsymbol{L}(k)$. The name "Transition Pattern Coding" comes from the fact that the encoder must minimize the expected cost per transition. Because of this form of energy cost, the state $\boldsymbol{L}(k)$ need not depend on earlier states $\boldsymbol{L}(k-2)$, $\boldsymbol{L}(k-3)$, ... Relation (3) implies that the vector $\boldsymbol{L}(k)$ may be restricted in a subset $W = \{w_1, w_2, \ldots, w_M\}$ of $\{0, 1\}^{m+a}$. It may be the case that $W = \{0, 1\}^{m+a}$. The M elements of W are the *codewords* of the coding scheme.

Now, at the other end of the bus the decoder uses the value of the current and the previous states of the bus to reconstruct the data vector. This information is sufficient, and no other earlier states are required. The decoder implies the relation:

$$\hat{\boldsymbol{D}}(k) = G(\boldsymbol{L}(k), \boldsymbol{L}(k-1)) \tag{4}$$

For every k it must be $\hat{\boldsymbol{D}}(k) = \boldsymbol{D}(k)$. Combining the operations of the encoder and the decoder the previous expression translates into the requirement that: for every $\boldsymbol{w} \in W$ and every $\boldsymbol{d} \in \{0, 1\}^m$ it is:

$$G(\boldsymbol{w}, F(\boldsymbol{w}, \boldsymbol{d})) = \boldsymbol{d}. \tag{5}$$

Therefore, for every fixed $\boldsymbol{w} \in W$ the mapping, $\boldsymbol{d} \rightarrow F(\boldsymbol{w}, \boldsymbol{d})$ must be injective. Moreover, for every $\boldsymbol{w}$ the set

$$X_{\boldsymbol{w}} = \{F(\boldsymbol{w}, \boldsymbol{d}) : \mathbf{d} \in \{0, 1\}^m\} \tag{6}$$

containing all the possible values of $\boldsymbol{L}(k)$ when the last state was $\boldsymbol{L}(k-1) = \boldsymbol{w}$, must have

exactly 2^m elements. This is because m bits of information must be transmitted each time. The function F gives rise to the *transition matrix* $\boldsymbol{T} = [t_{i,j}]_{i,j=1}^{M}$ with elements:

$$t_{i,j} = \begin{cases} 1 & if \ \boldsymbol{w}_j \in X_{\boldsymbol{w}_i} \\ 0 & otherwise \end{cases} \tag{7}$$

The matrix $\boldsymbol{T}$ has exactly 2^m 1's in every one of its rows. It defines the possible transitions in the bus, i.e. given a state $\boldsymbol{L}(k-1)$ it gives the states $\boldsymbol{L}(k)$ that can follow. It ignores though the way the data is mapped into these transitions. Finally, we define the *transition graph* G_T which carries the same information as the transition matrix,

$$G_T = \{(\boldsymbol{w}, F(\boldsymbol{w}, \boldsymbol{d})) : \mathrm{w} \in W, \ \mathbf{d} \in \{0,1\}^m\} \tag{8}$$

Relation (5) defines (the restriction of) the function G on the vertices of the transition graph. The values of G in the set $W \times W - G_T$ are immaterial and can be chosen in a convenient way to simplify the hardware implementation of the function. If the transition graph G_T has more than one strongly connected component ([34]) then the coding scheme is degenerate in the sense that some of the codewords are not utilized. To avoid this degeneracy from now on we assume that G_T is *strongly connected*. Strong connectivity of G_T is equivalent to the *irreducibility* of the transition matrix T ([34]), a property that will be used later.

3.1 A Motivational Example

A simple *TPC* scheme is presented where $m = 2$ bits are transmitted each time, the bus has been expanded to $m + a = 3$ lines and the set of codewords contains $M = 6$ elements $W = \{\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_6\}$ where,

$\boldsymbol{w}_1 = 000$	$\boldsymbol{w}_2 = 001$	$\boldsymbol{w}_3 = 010$
$\boldsymbol{w}_4 = 101$	$\boldsymbol{w}_5 = 110$	$\boldsymbol{w}_6 = 111$

The coding function F is given by the following diagram:

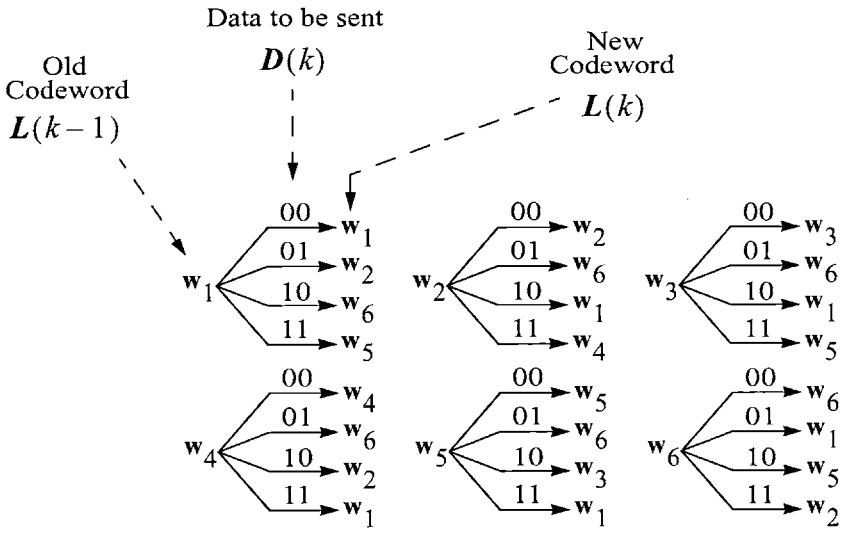

Figure 2: Graph representation of F

For example, if $\boldsymbol{L}(k-1) = 001 \quad (\boldsymbol{w}_2)$ and $\boldsymbol{D}(k) = 11$ then $\boldsymbol{L}(k) = 101 \quad (\boldsymbol{w}_4)$. The transition matrix resulting from F with columns and rows number according to states $\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_6$ is:

$$T = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 \end{bmatrix}$$

To indicate the correspondence between the input data and the 2^m ones of each row of T we may use the following symbolic matrix that contains all the information of the function F:

$$\tilde{T} = \begin{bmatrix} (00) & (01) & 0 & 0 & (11) & (10) \\ (10) & (00) & 0 & (11) & 0 & (01) \\ (10) & 0 & (00) & 0 & (11) & (01) \\ (11) & (10) & 0 & (00) & 0 & (01) \\ (11) & 0 & (10) & 0 & (00) & (01) \\ (01) & (11) & 0 & 0 & (10) & (00) \end{bmatrix}.$$

In this example, *TPC favors transition patterns in which the voltages of neighboring bus lines change values in the same direction.* This reduces the *effective capacitance* between adjacent wires. For a typical $0.18\mu m$ technology and minimum distance between the wires, $\lambda \cong 5$. In this case, the coding schemes results in a theoretical energy reduction of about 22%, *even though the number of the bus lines has been increased.* The savings occur because the scheme encodes the information $\boldsymbol{D}(k)$ into transitions among codewords that have an average energy cost less than that of the transitions in the original data.

In the following sections we address the three theoretical important issues of *TPC*: i) Exact calculation of the energy consumption. ii) Design of coding schemes. iii) Complexity reduction.

3.2 Statistical Measures of Energy

A transition of the bus from state $\boldsymbol{L}(k-1)$ to state $\boldsymbol{L}(k)$ requires a certain amount of energy $E(\boldsymbol{L}(k-1), \boldsymbol{L}(k))$ to be drawn from the power supply. Using formula (1) we can express it as:

$$\boldsymbol{E}(\boldsymbol{L}(\boldsymbol{k}-\boldsymbol{1}), \boldsymbol{L}(\boldsymbol{k})) = \boldsymbol{L}(\boldsymbol{k})^{\boldsymbol{T}} \cdot \boldsymbol{C}^{\boldsymbol{la}} \cdot [\boldsymbol{L}(\boldsymbol{k}) - \boldsymbol{L}(\boldsymbol{k}-\boldsymbol{1})]^{\boldsymbol{T}} \tag{9}$$

(The bits of the vectors $\boldsymbol{L}(k)$ are regarded to be zero or one in the field of real numbers and not in the Boolean algebra). If $\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_M$ are the codewords of the coding scheme, we define the *energy cost matrix* of the scheme as:

$$\boldsymbol{E} = [E(\boldsymbol{w}_i, \boldsymbol{w}_j)]_{i,j=1}^{M} \tag{10}$$

To quantify the energy behavior of the coding schemes we need to introduce a statistical energy measure that is both physically meaningful and can lead to relatively simple algebraic expressions. It seems appropriate to define the *time average expected energy* per transition of the bus, E_a as the limit of the average expected energy consumption over N consecutive transitions when N goes to infinity:

$$E_a = \lim_{N \to \infty} \frac{1}{N} \cdot \sum_{k=1}^{N} \overline{E(\boldsymbol{L}(k-1), \boldsymbol{L}(k))} \tag{11}$$

Throughout the Chapter, the over line means statistical expectation with respect to all random variables involved in the expression; here is it with respect to $\boldsymbol{L}(k-1)$ and $\boldsymbol{L}(k)$. In (11), the quantity:

$$\frac{1}{N} \cdot \sum_{k=1}^{N} \overline{E(\boldsymbol{L}(k-1), \boldsymbol{L}(k))}$$

is the average expected energy drawn from V_{dd} during the first N transitions. The limit as $N \to \infty$ provides the time averaging.

3.3 Energy Consumption Using *TPC*

Estimation of the energy properties of *TPC* schemes requires some assumptions on the statistics of the input data $\{\boldsymbol{D}(k) = (\boldsymbol{d}_1(k), \boldsymbol{d}_2(k), \ldots, \boldsymbol{d}_m(k))^T\}_k$. The analysis will be presented for the case where the data forms an *i.i.d and uniformly distributed sequence in* $\{0, 1\}^m$. *The result for the case that the data sequence is not uniformly distributed* is similar and will be stated without proof.

Suppose for now that $\{\boldsymbol{D}(k)\}_k$ *is i.i.d. and uniformly distributed.* Then the stochastic process $\boldsymbol{L}(k)$ (Figure 1) is first order homogeneous Markov [36] and the conditional probability for the transition $\boldsymbol{w}_i \to \boldsymbol{w}_j$ is given by

$$P_r(\boldsymbol{L}(k) = \boldsymbol{w}_j \mid \boldsymbol{L}(k-1) = \boldsymbol{w}_i) = \begin{cases} \dfrac{1}{2^m} & \text{if } \boldsymbol{w}_j \in X_{\boldsymbol{w}_i} \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

The *transition probability matrix* $\boldsymbol{P}$ of the states of the coding scheme is defined as:

$$\boldsymbol{P} = [P_{i,j}]_{i,j=1}^{M} = [P_r(\boldsymbol{L}(k) = \boldsymbol{w}_i \mid \boldsymbol{L}(k-1) = \boldsymbol{w}_j)]_{i,j=1}^{M} \tag{13}$$

From equations (12) we conclude that

$$\boldsymbol{P} = \boldsymbol{T}/2^m \tag{14}$$

The *probability* (row) *vector* of the state $\boldsymbol{L}(k)$ is:

$$\boldsymbol{p}(k) = \left[P_r(\boldsymbol{L}(k)= \boldsymbol{w}_1), \ \ldots, P_r(\boldsymbol{L}(k)= \boldsymbol{w}_M)\right] \tag{15}$$

If $\boldsymbol{p}(0)$ is the probability distribution of the initial state $\boldsymbol{L}(0)$ of the bus, then using the *Chapman-Kolmogorov* formula [36], that is: $\boldsymbol{p}(k) = \boldsymbol{p}(r) \cdot \boldsymbol{P}^{k-r}, \quad k \geq r$, we have

$$\boldsymbol{p}(k) = \boldsymbol{p}(0) \cdot \boldsymbol{P}^k \tag{16}$$

We use expression (16) to evaluate the time average expected energy of the general coding scheme of Figure 1. We start by calculating the expected value of $E(\boldsymbol{L}(k-1), \boldsymbol{L}(k))$. Let $p_i(k)$ be the *i-th* entrance of the probability vector $p(k)$ and $P_{i,j}$ be the (i,j) element of the transition probability matrix $\boldsymbol{P}$. Then

$$\begin{aligned}
\overline{E(\boldsymbol{L}(k-1), \boldsymbol{L}(k))} &= \sum_{i,j=1}^{M} P_r(\boldsymbol{L}(k) = \boldsymbol{w}_j, \boldsymbol{L}(k-1) = \boldsymbol{w}_i) \cdot E(\boldsymbol{w}_i, \boldsymbol{w}_j) \\
&= \sum_{i,j=1}^{M} P_r(\boldsymbol{L}(k) = \boldsymbol{w}_j \mid \boldsymbol{L}(k-1) = \boldsymbol{w}_i) \cdot (P_r(\boldsymbol{L}(k-1)) = \boldsymbol{w}_i) \cdot E(\boldsymbol{w}_i, \boldsymbol{w}_j) \\
&= \sum_{i,j=1}^{M} P_{i,j} \cdot p_i(k-1) \cdot E(\boldsymbol{w}_i, \boldsymbol{w}_j)
\end{aligned}$$

Let $\boldsymbol{A} \bullet \boldsymbol{B} = [a_{i,j} \cdot b_{i,j}]_{i,j}$ be the *Hadamard product* [37] of two matrices $\boldsymbol{A}, \boldsymbol{B}$ of the same dimensions. and let $\underline{\mathbf{1}}$ be the (column) vector with all its coordinates equal to one. Then we can write:

$$\overline{E(\boldsymbol{L}(k-1), \boldsymbol{L}(k))} = p(k-1) \cdot (\boldsymbol{P} \bullet \boldsymbol{E}) \cdot \underline{\mathbf{1}} \tag{17}$$

and using (16),

$$\overline{E(\boldsymbol{L}(k-1), \boldsymbol{L}(k))} = p(0) \cdot \boldsymbol{P}^{k-1} \cdot (\boldsymbol{P} \bullet \boldsymbol{E}) \cdot \underline{\mathbf{1}} \tag{18}$$

To continue on the evaluation of the *time average expected energy* it is necessary to recall our assumption on the *strong connectivity* of the transition graph G_T. As mentioned before the strong

connectivity of G_T is equivalent to the *irreducibility* of the transition matrix $\boldsymbol{T}$ and because of (14), to the *irreducibility* of the transition probability matrix $\boldsymbol{P}$, [37]. Moreover, the matrix $\boldsymbol{P}$ is row stochastic by its definition. All these allow us to use the following modified version of the *Perron-Frobineous* theorem. (corollary 8.4.6 in [37]).

Theorem 1 (Perron-Frobineous)

An irreducible row stochastic matrix $\boldsymbol{P}$ is similar to a matrix of the following block diagonal form:

$$\boldsymbol{\Delta} = \begin{bmatrix} \boldsymbol{\Delta}_1 & 0 \\ 0 & \boldsymbol{Y} \end{bmatrix} \tag{19}$$

where $\boldsymbol{Y}$ is in Jordan form with eigenvalues of modulus less than one and $\boldsymbol{\Delta}_1$ is the following diagonal matrix:

$$\boldsymbol{\Delta}_1 = diag\left(1, e^{\frac{2\pi i}{q}}, e^{\frac{4\pi i}{q}}, \ldots, e^{\frac{2(q-1)\pi i}{q}}\right) \tag{20}$$

The constant q is the number of eigenvalues of matrix $\boldsymbol{P}$ of modulus one. The number q is always equal or greater than one. •

Therefore, there exists a non singular matrix $\boldsymbol{A}$ and a matrix $\boldsymbol{\Delta}$ as in the theorem such that:

$$\boldsymbol{P} = \boldsymbol{A} \cdot \boldsymbol{\Delta} \cdot \boldsymbol{A}^{-1} \tag{21}$$

Moreover, from expression (18) it is

$$\overline{E(\boldsymbol{L}(k-1), \boldsymbol{L}(k))} = \boldsymbol{p}(0) \cdot \boldsymbol{A} \cdot \boldsymbol{\Delta}^{k-1} \cdot \boldsymbol{A}^{-1} \cdot (\boldsymbol{P} \bullet \boldsymbol{E}) \cdot \underline{\mathbf{1}} \tag{22}$$

and hence:

$$\frac{1}{N} \cdot \sum_{k=1}^{N} \overline{E(\boldsymbol{L}(k-1), \boldsymbol{L}(k))} = \boldsymbol{p}(0) \cdot \boldsymbol{A} \cdot \frac{1}{N} \cdot \left(\sum_{k=1}^{N} \begin{bmatrix} \boldsymbol{\Delta}_1^{k-1} & 0 \\ 0 & \boldsymbol{Y}^{k-1} \end{bmatrix} \right) \cdot \boldsymbol{A}^{-1} \cdot (\boldsymbol{P} \bullet \boldsymbol{E}) \cdot \underline{\mathbf{1}} \tag{23}$$

Since the spectral radius of $\boldsymbol{Y}$ is less than one, we have $\boldsymbol{Y}^r \to 0$ as $r \to \infty$. Also, for every integer r it holds that ([38]):

$$\lim_{N\to\infty} \frac{1}{N} \cdot \sum_{k=1}^{N} e^{\frac{r\pi i}{q}} = \begin{cases} 1 \text{ if } r \text{ is a multiple of } q \\ 0 \quad \text{otherwise} \end{cases} \tag{24}$$

We can conclude from (23) that:

$$\lim_{N\to\infty} \frac{1}{N} \cdot \sum_{k=1}^{N} \overline{E(\boldsymbol{L}(k-1), \boldsymbol{L}(k))} = \boldsymbol{p}(0) \cdot \boldsymbol{A} \cdot \boldsymbol{e}_1^T \cdot \boldsymbol{e}_1 \cdot \boldsymbol{A}^{-1} \cdot (\boldsymbol{P} \bullet \boldsymbol{E}) \cdot \underline{1} \tag{25}$$

Since $\boldsymbol{P}$ is a row stochastic matrix, its right eigenvector corresponding to the eigenvalue one is $\underline{1}$. Therefore we can write that, $\boldsymbol{A} \cdot \boldsymbol{e}_1^T \cdot \boldsymbol{e}_1 \cdot \boldsymbol{B} = \underline{1} \cdot \boldsymbol{b}_0^T$. Where $\boldsymbol{b}_0$ is the left eigenvector of matrix $\boldsymbol{P}$ corresponding to eigenvalue one, that is $\boldsymbol{b}_0^T \cdot \boldsymbol{P} = \boldsymbol{b}_0^T$, and satisfies $\boldsymbol{b}_0^T \cdot \underline{1} = 1$. Therefore equation (25) becomes,

$$\begin{aligned} \lim_{N\to\infty} \frac{1}{N} \cdot \sum_{k=1}^{N} \overline{E(\boldsymbol{L}(k-1), \boldsymbol{L}(k))} &= \boldsymbol{p}(0) \cdot \boldsymbol{A} \cdot \boldsymbol{e}_1^T \cdot \boldsymbol{e}_1 \cdot \boldsymbol{A}^{-1} \cdot (\boldsymbol{P} \bullet \boldsymbol{E}) \cdot \underline{1} \\ &= \boldsymbol{p}(0) \cdot \underline{1} \cdot \boldsymbol{b}_0^T \cdot (\boldsymbol{P} \bullet \boldsymbol{E}) \cdot \underline{1} \\ &= \boldsymbol{b}_0^T \cdot (\boldsymbol{P} \bullet \boldsymbol{E}) \cdot \underline{1} \end{aligned} \tag{26}$$

where we have used, $\boldsymbol{p}(0) \cdot \underline{1} = 1$. So, for the case of i.i.d. and uniform input sequence the time average expected energy is,

$$E_a = \boldsymbol{b}_0^T \cdot (\boldsymbol{P} \bullet \boldsymbol{E}) \cdot \underline{1} = \frac{1}{2^m} \cdot \boldsymbol{b}_0^T \cdot (\boldsymbol{T} \bullet \boldsymbol{E}) \cdot \underline{1} \tag{27}$$

Now, recall that the transition matrix $\boldsymbol{T}$ has a one in the (i, j) position if and only if the transition $\boldsymbol{w}_i \to \boldsymbol{w}_j$ is allowed. There are exactly 2^m ones in every row of the matrix $\boldsymbol{T}$. In the case of uniformly distributed input data the time average expected energy is not related to the particular way that the data is mapped into the transitions of the bus.

Now suppose that the data is an *i.i.d.* but *non-uniform* sequence. Let p_i^D be the probability that $\boldsymbol{D}(k) = i$, $i = 1, 2, \ldots, 2^m$. *(the obvious identification is used throughout the chapter between the binary vectors* $(0, \ldots, 0, 0)$, $(0, \ldots, 0, 1), \ldots, (1, \ldots, 1, 1)$ *and the integers* $1, 2, \ldots, 2^m$. *This is needed in order to use the matrix algebra).* Then, every row of the transition matrix $\boldsymbol{P}$ contains zeros and the numbers $p_1^D, p_2^D, \ldots, p_{2^m}^D$. The analysis presented above is still valid if in the place of $\boldsymbol{T}$ we use $2^m \cdot \boldsymbol{P}$. We have:

$$E_a = \boldsymbol{b}_0^T \cdot (\boldsymbol{P} \bullet \boldsymbol{E}) \cdot \underline{\mathbf{1}} \tag{28}$$

3.4 Energy Estimation; Examples

Here we apply the derived formulas on the example of Section 3.1. Assume again a two-line and a three-line bus with lambda equal to five. For simplicity we set $C_L = 1$ and $V_{dd} = 1$. Using formula (1) we calculate the transition energies for the 2-line bus:

		New State			
		00	01	10	11
Old State	00	0	6	6	2
	01	0	0	11	1
	10	0	11	0	1
	11	0	5	5	0

Table 2: Transition Energy of the 2-lines example bus.

In the example of Section 3.1 two bits of information are transmitted each time. Without using coding and assuming the data is i.i.d. and uniform the expected energy per transmission would be:

$$E_a = [0.25, 0.25, 0.25, 0.25] \cdot \begin{bmatrix} 0 & 6 & 6 & 2 \\ 0 & 0 & 11 & 1 \\ 0 & 11 & 0 & 1 \\ 0 & 5 & 5 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0.25 \\ 0.25 \\ 0.25 \\ 0.25 \end{bmatrix} = 3 \tag{29}$$

Suppose now we use the coding scheme presented in Figure 2. For the 3-line bus the energy of the transitions is given below in Table 3:

		New State							
		000	001	010	011	100	101	110	111
Old State	000	0	6	11	7	6	12	7	3
	001	0	0	16	6	6	6	12	2
	010	0	11	0	1	11	22	1	2
	011	0	5	5	0	11	16	6	1
	100	0	6	16	12	0	6	6	2
	101	0	0	21	11	0	0	11	1
	110	0	11	5	6	5	16	0	1
	111	0	5	10	5	5	10	5	0

Table 3: Transition Energy of the 3-lines example bus.

The codewords used in the example where $w_1 = 000$, $w_2 = 001$, $w_3 = 010$, $w_4 = 101$, $w_5 = 110$, $w_6 = 111$. The *energy cost matrix* of the scheme, defined by equation (10), is:

$$\boldsymbol{E} = \begin{bmatrix} 0 & 6 & 11 & 12 & 7 & 3 \\ 0 & 0 & 16 & 6 & 12 & 2 \\ 0 & 11 & 0 & 22 & 1 & 2 \\ 0 & 0 & 21 & 0 & 11 & 1 \\ 0 & 11 & 5 & 16 & 0 & 1 \\ 0 & 5 & 10 & 10 & 5 & 0 \end{bmatrix} \tag{30}$$

The probability transition matrix is:

$$\boldsymbol{P} = \begin{bmatrix} 0.25 & 0.25 & 0 & 0 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0 & 0.25 & 0 & 0.25 \\ 0.25 & 0 & 0.25 & 0 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0 & 0.25 & 0 & 0.25 \\ 0.25 & 0 & 0.25 & 0 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0 & 0 & 0.25 & 0.25 \end{bmatrix} \tag{31}$$

The left eigenvector $\boldsymbol{b}_0^T$ of $\boldsymbol{P}$ corresponding to the eigenvalue one and satisfying the equation $\boldsymbol{b}_0^T \cdot \underline{\mathbf{1}} = 1$ is:

$$\boldsymbol{b}_0^T = [0.250, 0.187, 0.062, 0.062, 0.187, 0.250] \tag{32}$$

So, the time average expected energy of the coding scheme, given by expression (27) is:

$$E_a = \boldsymbol{b}_0^T \cdot (\boldsymbol{P} \bullet \boldsymbol{E}) \cdot \underline{\mathbf{1}} = 2.3437$$

Moreover, the energy reduction is: $100 \cdot \left(\frac{3 - 2.343}{3}\right)\% = 21.88\ \%$

Now suppose that the data is not uniformly distributed, but for every time k the random vector $\boldsymbol{D}(k)$ has the probability distribution:

$\boldsymbol{u}$:	00	01	10	11
$P_r(\boldsymbol{D}(k) = \boldsymbol{u})$:	0.1	0.3	0.4	0.2

Table 4: Distribution of the Input vectors

Then the transition probability matrix $\boldsymbol{P}$ of the coding scheme becomes (see Figure 2):

$$\boldsymbol{P} = \begin{bmatrix} 0.1 & 0.3 & 0 & 0 & 0.2 & 0.4 \\ 0.4 & 0.1 & 0 & 0.2 & 0 & 0.3 \\ 0.4 & 0 & 0.1 & 0 & 0.2 & 0.3 \\ 0.2 & 0.4 & 0 & 0.1 & 0 & 0.3 \\ 0.2 & 0 & 0.4 & 0 & 0.1 & 0.3 \\ 0.3 & 0.2 & 0 & 0 & 0.4 & 0.1 \end{bmatrix} \tag{33}$$

The left eigenvector $\boldsymbol{b}_0^T$ of $\boldsymbol{P}$ corresponding to eigenvalue one and satisfying the equation $\boldsymbol{b}_0^T \cdot \underline{\mathbf{1}} = 1$ is $\boldsymbol{b}_0^T = [0.251, 0.160, 0.087, 0.036, 0.196, 0.271]$. The time average expected energy of the coding scheme, given by expression (28) is: $E_a = \boldsymbol{b}_0^T \cdot (\boldsymbol{P} \bullet \boldsymbol{E}) \cdot \underline{\mathbf{1}} = 2.736$. If the data was transmitted uncoded through a 2-line bus the time average expected energy would be:

$$E_a = [0.1, 0.3, 0.4, 0.2] \cdot \begin{bmatrix} 0 & 6 & 6 & 2 \\ 0 & 0 & 11 & 1 \\ 0 & 11 & 0 & 1 \\ 0 & 5 & 5 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0.1 \\ 0.3 \\ 0.4 \\ 0.2 \end{bmatrix} = 3.94 \tag{34}$$

and so the energy reduction is: $100 \cdot \left(\frac{3.94 - 2.736}{3.94}\right)\% = 30.6\ \%$

4. Two Algorithms for Deriving Efficient Coding Schemes

Suppose the original bus has m lines and that at each time moment, k, the input $\boldsymbol{D}(k)$ takes a value $(0\ldots00)$, $(0\ldots01)$, $(0\ldots10)$,..., $(1\ldots11)$ with probability $p_1^d, p_2^d, \ldots, p_{2^m}^d$ respectively. Suppose that we expand the bus by a additional lines, so we have $m+a$ bits to encode the data. Throughout this section we assume that all vectors in $\{0,1\}^{m+a}$ are possible states of the expanded bus, in other words it is, $M = 2^{m+a}$ and $W = \{0,1\}^{m+a}$. Without loss of generality we can set $\boldsymbol{w}_1 = (0\ldots00)$, $\boldsymbol{w}_2 = (0\ldots01)$,..., $\boldsymbol{w}_{2^{m+a}} = (1\ldots11)$. The transition energy cost matrix $\boldsymbol{E}$ is defined as in equation (10).

Definition.

Given the numbers m, a and the distribution $\boldsymbol{p}^d = [p_1^d, p_2^d, \ldots, p_{2^m}^d]$ we define the set $\Pi(m, a, \boldsymbol{p}^d)$ of all $2^{m+a} \times 2^{m+a}$ stochastic matrices $\boldsymbol{P}$ such that: Each row of $\boldsymbol{P}$ has 2^m entries with (all) the numbers $p_1^d, p_2^d, \ldots, p_{2^m}^d$ and the rest $2^{m+a} - 2^m$ of the entries are zeros.

Example: Let $m = 2$, $a = 1$ and $\boldsymbol{p}^d = [2, 1, 2, 5]/10$. The following three matrices are members of the set $\Pi(m, a, \boldsymbol{p}^d)$.

$$\boldsymbol{P}_1 = \frac{1}{10}\begin{bmatrix} 5&2&0&0&1&0&0&2 \\ 5&2&0&1&0&0&0&2 \\ 5&0&2&2&0&0&1&0 \\ 5&1&0&2&0&0&0&2 \\ 5&1&0&0&2&0&0&2 \\ 5&2&0&0&2&1&0&0 \\ 5&0&1&0&0&0&2&2 \\ 5&2&0&1&0&0&0&2 \end{bmatrix}, \quad \boldsymbol{P}_2 = \frac{1}{10}\begin{bmatrix} 5&2&0&0&1&0&0&2 \\ 2&5&0&0&0&1&0&2 \\ 1&0&5&2&0&0&2&0 \\ 2&0&1&5&0&0&0&2 \\ 2&0&0&0&5&1&0&2 \\ 1&2&0&0&2&5&0&0 \\ 2&0&1&0&0&0&5&2 \\ 2&0&0&2&0&0&1&5 \end{bmatrix}, \quad \boldsymbol{P}_3 = \frac{1}{10}\begin{bmatrix} 5&2&0&0&1&0&0&2 \\ 2&5&0&0&0&1&0&2 \\ 1&0&5&2&0&0&2&0 \\ 2&0&1&5&0&0&0&2 \\ 2&0&0&0&5&1&0&2 \\ 0&2&0&0&2&5&0&1 \\ 2&0&1&0&0&0&5&2 \\ 2&0&0&2&0&0&1&5 \end{bmatrix} \tag{35}$$

Following the discussion in the previous sections, the design of an efficient *TPC* scheme given the parameters $m, a, \boldsymbol{p}^d$ is equivalent to choosing a "good" matrix P from the set $\Pi(m, a, \boldsymbol{p}^d)$. Note that $\boldsymbol{P}$ leads directly to the calculation of the time average expected energy using expression (28), that is $E_a = \boldsymbol{b}_0^T \cdot (\boldsymbol{P} \bullet \boldsymbol{E}) \cdot \underline{1}$ with $\boldsymbol{b}_0$ the left probability eigenvector of $\boldsymbol{P}$.

4.1 Approximately Optimal Coding (AOC)

Motivated by expression (28) where the matrix $\boldsymbol{P}$ is involved in the point-wise product with the energy cost matrix $\boldsymbol{E}$, we may attempt to achieve a low value of E_a by simply choosing $\boldsymbol{P}$ so that each entry of the matrix $\boldsymbol{P} \bullet \boldsymbol{E}$ is minimal. In other words, minimize the quantity $\underline{1}^T \cdot (\boldsymbol{P} \bullet \boldsymbol{E}) \cdot \underline{1}$ instead of $\boldsymbol{b}_0^T \cdot (\boldsymbol{P} \bullet \boldsymbol{E}) \cdot \underline{1}$.

Example: Lets apply this idea to derive a scheme that encodes a 2-bit sequence with probability distribution $\boldsymbol{p}^d = [2, 1, 2, 5]/10$ into a 3-line bus. Lets assume a bus with $\lambda = 5$. Using expression (1) and definition (10) it is:

$$\boldsymbol{E} = \begin{bmatrix} 0 & 6 & 11 & 7 & 6 & 12 & 7 & 3 \\ 0 & 0 & 16 & 6 & 6 & 6 & 12 & 2 \\ 0 & 11 & 0 & 1 & 11 & 22 & 1 & 2 \\ 0 & 5 & 5 & 0 & 11 & 16 & 6 & 1 \\ 0 & 6 & 16 & 12 & 0 & 6 & 6 & 2 \\ 0 & 0 & 21 & 11 & 0 & 0 & 11 & 1 \\ 0 & 11 & 5 & 6 & 5 & 16 & 0 & 1 \\ 0 & 5 & 10 & 5 & 5 & 10 & 5 & 0 \end{bmatrix}$$

We examine the first row, $[0, 6, 11, 7, 6, 12, 7, 3]$ of the matrix $\boldsymbol{E}$. An optimal choice for the first row of the stochastic matrix $\boldsymbol{P}$ would be $[5, 1, 0, 0, 2, 0, 0, 2]/10$. Note that the maximal entry of the probability vector, $\boldsymbol{p}^d = [2, 1, 2, 5]/10$ was matched with the minimal element of the first row of $\boldsymbol{E}$. The second maximal element of $\boldsymbol{p}^d$ was matched with the second minimal element of the first row of $\boldsymbol{E}$ etc. In general this procedure requires only *sorting* the elements of the rows and it guarantees the minimality of the of the matrix $\boldsymbol{P} \bullet \boldsymbol{E}$ and moreover the minimality of the vector $(\boldsymbol{P} \bullet \boldsymbol{E}) \cdot \underline{1}$. In our example, a stochastic matrix in $\Pi(m, a, \boldsymbol{p}^d)$ that minimizes the quantity $(\boldsymbol{P} \bullet \boldsymbol{E}) \cdot \underline{1}$ is:

$$\boldsymbol{P}^* = \frac{1}{10}\begin{bmatrix} \mathbf{5} & \mathbf{1} & 0 & 0 & \mathbf{2} & 0 & 0 & \mathbf{2} \\ 5 & 2 & 0 & 1 & 0 & 0 & 0 & 2 \\ 5 & 0 & \mathbf{2} & 2 & 0 & 0 & 1 & 0 \\ 5 & 1 & 0 & 2 & 0 & 0 & 0 & 2 \\ 5 & 1 & 0 & 0 & \mathbf{2} & 0 & 0 & 2 \\ 5 & 2 & 0 & 0 & 2 & \mathbf{1} & 0 & 0 \\ 5 & 0 & 1 & 0 & 0 & 0 & \mathbf{2} & 2 \\ 5 & 1 & 0 & 2 & 0 & 0 & 0 & 2 \end{bmatrix} \tag{36}$$

The product $(\boldsymbol{P} \bullet \boldsymbol{E}) \cdot \underline{\mathbf{1}}$ becomes:

$$(\boldsymbol{P}^* \bullet \boldsymbol{E}) \cdot \underline{\mathbf{1}} = \frac{1}{10}\begin{bmatrix} 24 \\ 10 \\ 3 \\ 7 \\ 10 \\ 0 \\ 7 \\ 15 \end{bmatrix} \tag{37}$$

Using matrix $\boldsymbol{P}^*$ we can calculate the time average expected energy of the coding scheme. The left probability eigenvector of $\boldsymbol{P}^*$ is $(\boldsymbol{b}_o^*)^T = [0.5, 0.111, 0, 0.064, 0.125, 0, 0, 0.2]$ and the time average expected energy is: $E_a = \boldsymbol{b}^{*T}_0 \cdot (\boldsymbol{P}^* \bullet \boldsymbol{E}) \cdot \underline{\mathbf{1}} = 1.781$. Note the fact that vector $\boldsymbol{b}_o$ has only five non-zero elements. This means that the stochastic matrix $\boldsymbol{P}^*$ we chose corresponds to *TPC* schemes with only the five codewords: (000), (001), (011), (100) and (111). Note also that since the input vectors (00) and (10) have both probability 0.2, the assignment of the inputs to the transitions is not unique. For example, the transitions from state 1 to states 5, 8 can be assigned to inputs (00), (10) respectively or to (10), (00). Below we see a transition diagram based on $\boldsymbol{P}^*$ with a specific input assignment.

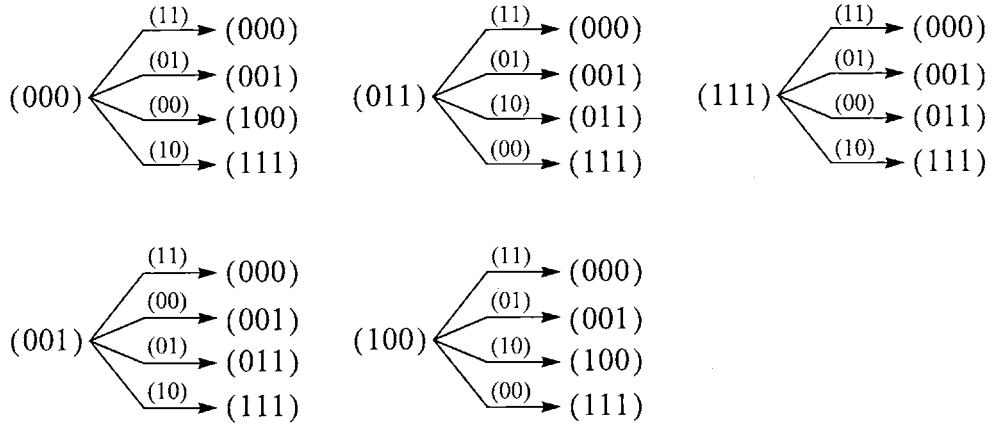

Figure 3: *Example transition diagram*

We know from the example of Section 3. that without coding, the expected energy consumption when transmitting the data though a 2-line bus is as in equation (34):

$$E_a = [0.1, 0.3, 0.4, 0.2] \cdot \begin{bmatrix} 0 & 6 & 6 & 2 \\ 0 & 0 & 11 & 1 \\ 0 & 11 & 0 & 1 \\ 0 & 5 & 5 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0.1 \\ 0.3 \\ 0.4 \\ 0.2 \end{bmatrix} = 1.9 \tag{38}$$

The energy reduction using this simple minimization is:

$$100 \cdot \left(\frac{1.9 - 1.781}{1.9} \right)\% = 6.26\ \%$$

which is insignificant. This very low energy reduction (it might be negative is other cases) is because we ignored the steady state probability vector $\boldsymbol{b}_0$ during the optimization process. Using the *TPC* algorithm that is presented in the next section the energy reduction for the particular example is 26.5%.

Although Approximately Optimal Coding may not be efficient for small m, it gives very good results for $m \geq 7$. In addition, it is a low complexity procedure.

The efficiency of approximate optimal coding for $m \geq 7$ will be demonstrated by examples in the following section. The efficiency is due to the tendency of the eigenvector $\boldsymbol{b}_0$ of $\boldsymbol{P}$ to become (more) uniform for large m and $m + a$.

4.2 Transition Pattern Coding (TPC) Algorithm

The objective of the *TPC* algorithm is to find a *TPC* scheme with as low time average expected energy as possible. In other words, to find a good approximation of the solution of the minimization problem:

$$E_a = \min_{\boldsymbol{P} \in \Pi(m, a, \boldsymbol{p}^d)} \boldsymbol{b}_0^T \cdot (\boldsymbol{P} \bullet \boldsymbol{E}) \cdot \underline{1}. \tag{39}$$

We always assume implicitly that $\boldsymbol{b}_0$ is the left probability eigenvector of $\boldsymbol{P}$. In the case that $\boldsymbol{P}$ is not strongly connected we keep the largest component of it and reduce the set of codewords appropriately. The *TPC* algorithm is presented below.

$k := 0, \xi := \underline{0}, \boldsymbol{P}^* := [0]$

Repeat

$\boldsymbol{P}^*_{old} := \boldsymbol{P}^*$

$k := k + 1$

$\boldsymbol{P}^* := \operatorname{argmin}_{\boldsymbol{P} \in \Pi(m, a, \boldsymbol{p}^d)} (\boldsymbol{E} + \underline{1} \cdot \xi^T) \bullet \boldsymbol{P}$

$\xi := [\, (\boldsymbol{E} + \underline{1} \cdot \xi^T) \bullet \boldsymbol{P}^*] \cdot \underline{1}$

Until $\boldsymbol{P}^* = \boldsymbol{P}^*_{old}$

Figure 4: The Transition Pattern Coding algorithm

The algorithm operates "backwards" and the variable k counts the backward steps. At each step the value of the vector $\boldsymbol{\xi} = (\xi_1, \xi_2, \ldots, \xi_M)^T$ and the stochastic matrix $\boldsymbol{P}^*$, that belongs to $\Pi(m, a, \boldsymbol{p}^d)$, are renewed. To keep track of the steps we think of $\boldsymbol{\xi}$ as $\boldsymbol{\xi}(k)$ and $\boldsymbol{P}^*$ as $\boldsymbol{P}^*(k)$. For each k, the value of $\xi_i(k)$ is the expected energy cost of a (forward) random path of length k starting from state i and evolving with transition probability matrices $\boldsymbol{P}^*(k)$, $\boldsymbol{P}^*(k-1),\ldots,\boldsymbol{P}^*(1)$.

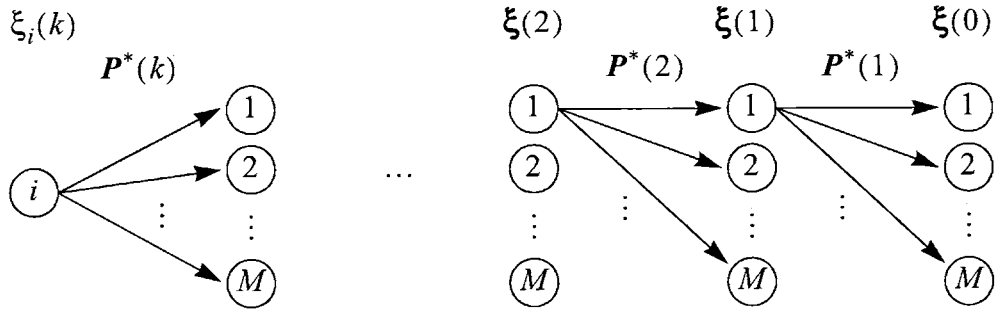

Figure 5: Expected costs of forward paths using the stochastic matrices derived by the optimization process.

To see this we start with $k = 0$. Since zero steps have been encountered starting from state i, it is of course $\xi_i(0) = 0$. For $k = 1$ the stochastic matrix $\boldsymbol{P}^*(1)$ is chosen within $\Pi(m, a, \boldsymbol{p}^d)$ so that starting from state i the expected cost of one step path starting from any state is minimal. Since $\boldsymbol{\xi}(0) = 0$, the choice of $\boldsymbol{P}^*(1)$ minimizes the quantity $\boldsymbol{E} \bullet \boldsymbol{P}$. Also, the expected costs of one step starting from different states are given by the vector $\boldsymbol{\xi}(1) = (\boldsymbol{E} \bullet \boldsymbol{P}) \cdot \underline{\mathbf{1}}$. On the *k-th* iteration of the algorithm the matrix $\boldsymbol{P}^*(k)$ is chosen to minimize $(\boldsymbol{E} + \underline{\mathbf{1}} \cdot \boldsymbol{\xi}^T) \bullet \boldsymbol{P}$ and moreover to minimize $[(\boldsymbol{E} + \underline{\mathbf{1}} \cdot \boldsymbol{\xi}^T) \bullet \boldsymbol{P}^*] \cdot \underline{\mathbf{1}}$. The minimizations of the M rows of $(\boldsymbol{E} + \underline{\mathbf{1}} \cdot \boldsymbol{\xi}^T) \bullet \boldsymbol{P}$ are independent to each other. The *i-th* row:

$$
\begin{aligned}
\boldsymbol{e}_i^T \cdot [\ (\boldsymbol{E} + \underline{\mathbf{1}} \cdot \boldsymbol{\xi}^T) \bullet \boldsymbol{P}^*(k)] \cdot \underline{\mathbf{1}} &= \boldsymbol{e}_i^T \cdot (\boldsymbol{E} \bullet \boldsymbol{P}^*(k)) \cdot \underline{\mathbf{1}} + \boldsymbol{e}_i^T \cdot [\ (\underline{\mathbf{1}} \cdot \boldsymbol{\xi}^T(k-1)) \bullet \boldsymbol{P}^*(k)] \cdot \underline{\mathbf{1}} \\
&= \boldsymbol{e}_i^T \cdot (\ \boldsymbol{E} \bullet \boldsymbol{P}^*(k)) \cdot \underline{\mathbf{1}} + \boldsymbol{e}_i^T \cdot \boldsymbol{P}^*(k) \cdot \boldsymbol{\xi}^T(k-1) \\
&= \sum_{j=1}^{M} p_{i,j}^*(k) \cdot (\boldsymbol{E}(i,j) + \xi_j(k-1))
\end{aligned}
\tag{40}
$$

is the sum of the expected costs of the steps going from the *k-th* to *(k-1)-th* state, plus the expected cost of a path of length $k-1$ starting from the *(k-1)-th* state. The algorithm stops the first time the calculated transition probability matrix equals the previous one.

Example: Let $m = 2$, $a = 1$ and $\boldsymbol{p}^d = [2, 1, 2, 5]/10$ as in the example in the beginning of Section 4. Again we assume a bus with $\lambda = 5$. The *TPC* algorithm terminates in three steps. For $k = 1, 2, 3$ it gives the stochastic matrices $\boldsymbol{P}_1^*, \boldsymbol{P}_2^*, \boldsymbol{P}_3^*$ shown in (35). These stochastic matrices result to time average expected energies: 1.7854, 1.3981, 1.3962 respectively. For $k = 4$ it is $\boldsymbol{P}_4^* = \boldsymbol{P}_3^*$.

Example: Now assume that $m = 8$, $a = 1$ and a bus with $\lambda = 5$. The algorithm is run for the following three distributions on the left of Figure 6. That is: *uniform*, *triangular* and *truncated gaussian*. The energy savings resulting using the stochastic matrices $\boldsymbol{P}^*(k)$, $k = 1, 2, \ldots$ of the algorithm are shown on the right. The first step, $k = 1$ corresponds to *AOC* and as we see gives a performance very close to the that of the final stochastic matrix. As it was mentioned before, this is the generally the case for $m \geq 7$. For the three input distributions, the algorithm requires 7, 10 and 11 iterations to stop.

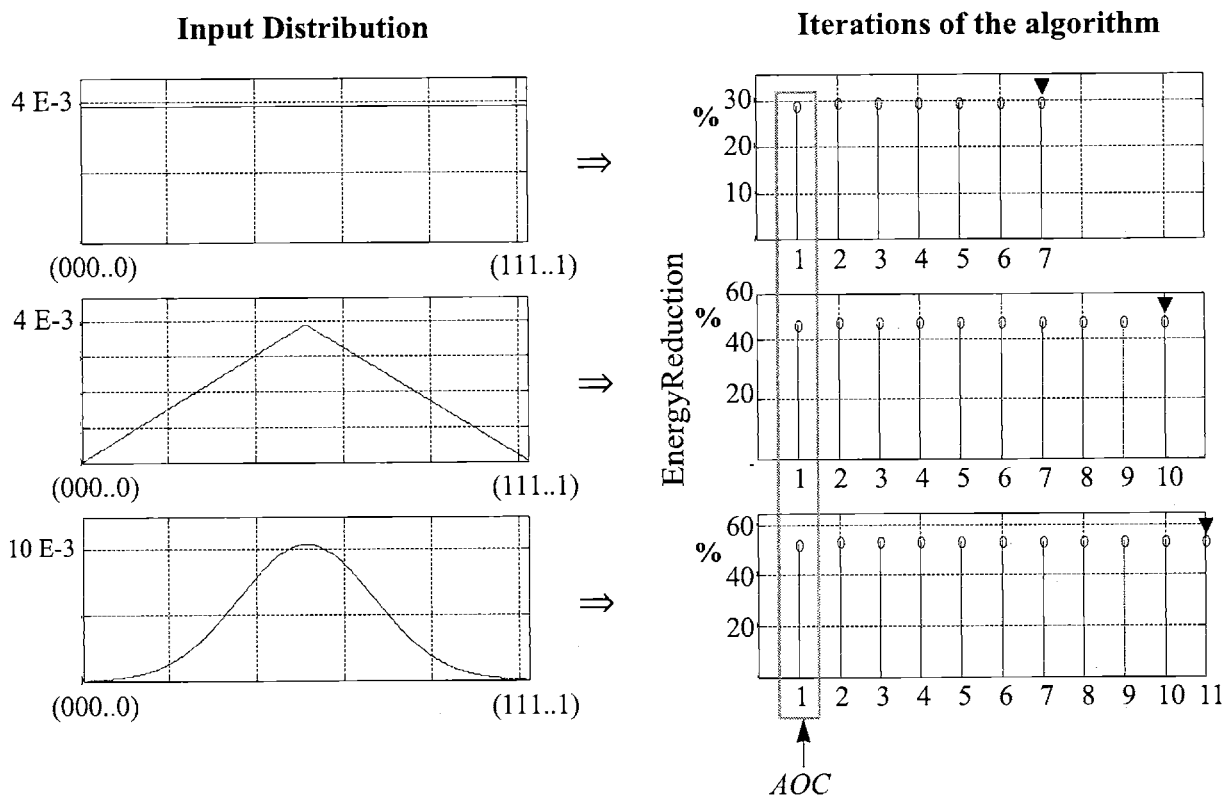

Figure 6: Energy savings corresponding to the stochastic matrix derived in each iteration of the *TPC* algorithm for different input distributions.

4.3 Results of the TPC Algorithm

The transition pattern coding algorithm has been used to derive coding schemes for many combinations of values of the quadruple $(m, a, \lambda, \textit{Input distribution})$ assuming the bus model as in Section 2. In Figure 7 we see the energy reductions (with respect to the uncoded bus) of coding schemes derived when $m = 2, 4, 8$, $a = 1$, $\lambda \in [0, 10]$ and with three input distributions: *uniform*, *triangular* and *truncated Gaussian* of normalized variance: $\sigma/\overline{D} = 0.5$. Each star "*" in the graphs depicts the energy reduction of a coding scheme with a particular quadruple $(m, a, \lambda, \textit{Input distribution})$. *The energy reduction can be arbitrarily high* and depends strongly on the distribution of the data. For example the graph on the right upper corner shows reduction up to 80%.

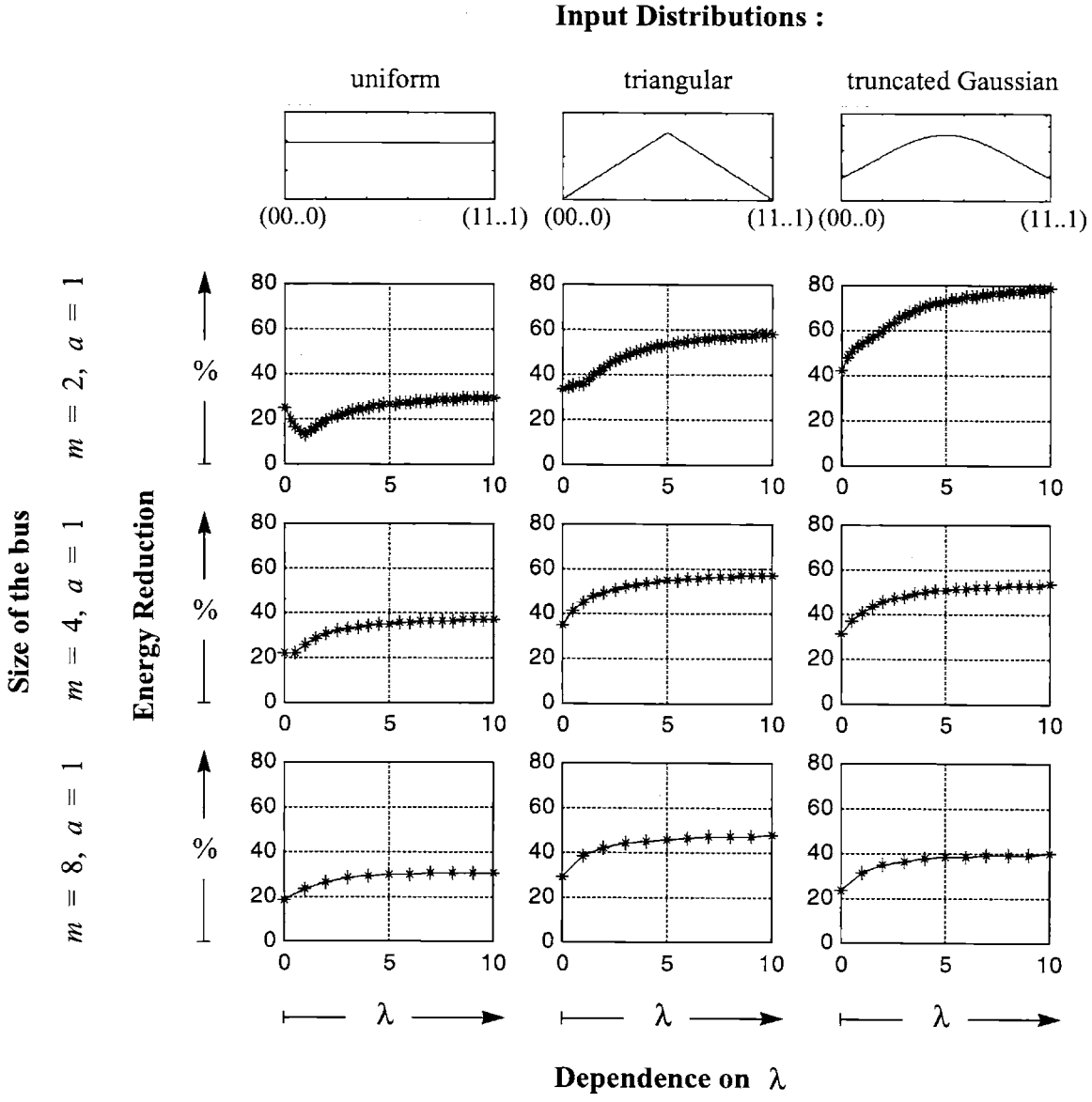

Figure 7: Energy reduction for $m = 2, 4, 8$, $a = 1$, $\lambda = 0, \ldots, 10$ and different input distributions

In the previous figure we saw results attributed to coding schemes with $a = 1$, i.e. with only one additional line. It is true that in most of the cases more additional lines result to higher energy reduction. This of course is not for free. Larger a means more additional chip area for the bus.

In the following figures we see the energy reductions of schemes resulting from the *TPC* algorithm when $m = 4$, $a = 1, 2, 3$, $\lambda \in [0, 10]$ and uniform input distribution. The savings are compared with that of Bus Invert [8].

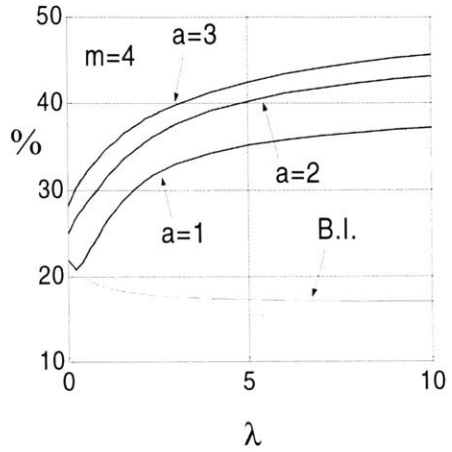

Figure 8: Energy Savings of *TPC* and Bus Invert for uniformly distributed input data

4.4 Copying With Complexity

A way to reduce complexity is to split the input data into groups and then encode each group independently of the others. The approach is shown below.

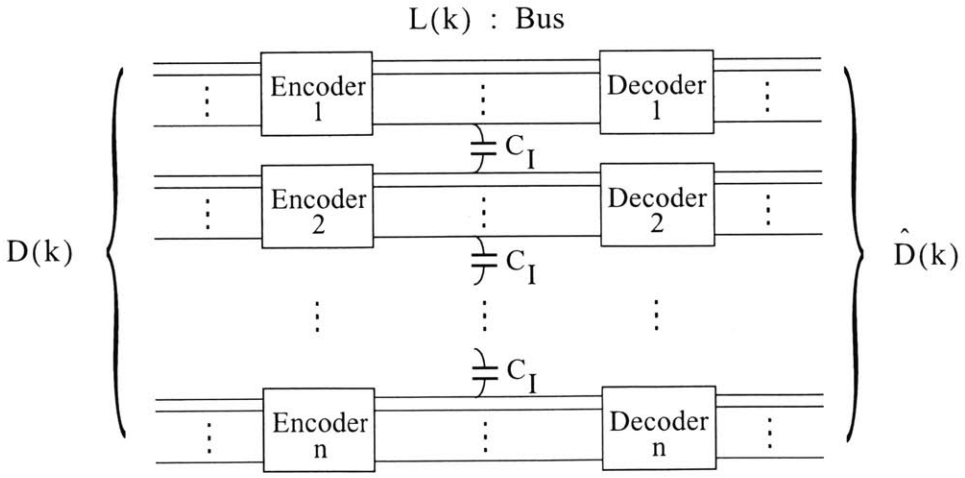

Figure 9: Partitioned Coding Scheme: A juxtaposition of simpler blocks

The energy dissipation in the partitioned coding scheme equals the sum of the energy dissipations of the individual blocks *plus* the energy losses due to the interactions at the boundaries of adjacent blocks. These interactions take place between the last line of the 1st block and the first line of the 2nd block, the last line of the 2nd block and the first line of the 3rd etc. The calculation of the expected energy dissipation caused by the interactions is presented in the next section.

4.5 The Interaction Energy

The energy loss caused by the interaction of two consecutive blocks corresponds to the λ's of Table 1 when V_1, V_2 are the values of the touching boundary lines of the two blocks. For simplicity, all coding blocks are assumed similar with parameters m,a,W,M and functions F and G as defined in Section 3. Let $\boldsymbol{L}(k)$ and $\boldsymbol{L}'(k)$ be the codewords of two consecutive coding blocks B and B' at time k. Suppose B is above B' (for example, in Figure 9 B and B' could the *1-st.* and *2-nd* block respectively). Let $l(k)$ be the *last* (bottom) bit of $\boldsymbol{L}(k)$ and $l'(k)$ be the *first* (top) bit of $\boldsymbol{L}'(k)$. So $l(k)$ and $l'(k)$ correspond to adjacent lines in the bus. From Table 1 we extract Table 5 that presents the *interaction energy* at time k caused by the inter-line capacitance. Call this (normalized) energy cost $J(k)$.

$J(k)$		$[l(k+1), l'(k+1)]$			
		00	01	10	11
$[l(k), l'(k)]$	00	0	λ	λ	0
	01	0	0	2λ	0
	10	0	2λ	0	0
	11	0	λ	λ	0

Table 5: Energy drawn from the power supply because of the interaction of adjacent lines $l(k)$ and $l'(k)$.

Since the purpose of partitioning the bus is to reduce the complexity of the encoders/decoders as well as the calculation of the total energy of the scheme, it is necessary to make the following simplification. We will assume that the *data streams* fed into the individual encoders of the parti-

tioned scheme are statistically identical and independent to each other, not necessarily i.i.d. sequences though. This approximation is not far from reality. Many data sources produce sequences of data where bits of similar significance tend to have high statistical correlation. On the contrary, bits that have very different significance tend to be more independent.

Following our assumption, the random variables $l(k)$ and $l'(k)$ are independent (because the data streams to different blocks are independent) and the expected value of $J(k)$ is given by Table 5,

$$\begin{aligned}
&\overline{J(k)} = \\
&\lambda \cdot P_r(l(k)=0, l'(k)=0, l(k+1)=0, l'(k+1)=1) + \\
&\lambda \cdot P_r(l(k)=0, l'(k)=0, l(k+1)=1, l'(k+1)=0) + \\
&2\lambda \cdot P_r(l(k)=0, l'(k)=1, l(k+1)=1, l'(k+1)=0) + \\
&2\lambda \cdot P_r(l(k)=1, l'(k)=0, l(k+1)=0, l'(k+1)=1) + \\
&\lambda \cdot P_r(l(k)=1, l'(k)=1, l(k+1)=0, l'(k+1)=1) + \\
&\lambda \cdot P_r(l(k)=1, l'(k)=1, l(k+1)=1, l'(k+1)=0)
\end{aligned} \tag{41}$$

By grouping the half of the *4rh* term with each of the *1st* and the *5th* terms and the half of the *3rd* with each of the *2nd* and the *6th* terms and taking into account that the two bit sequences $\{l(k)\}_k$ and $\{l'(k)\}_k$ are independent (to each other), (41) can be rewritten as:

$$\begin{aligned}
&\overline{J(k)} = \\
&\lambda \cdot P_r(l(k+1)=0) \cdot P_r(l'(k)=0, l'(k+1)=1) + \\
&\lambda \cdot P_r(l'(k+1)=1) \cdot P_r(l(k)=1, l(k+1)=0) + \\
&\lambda \cdot P_r(l'(k+1)=0) \cdot P_r(l(k)=0, l(k+1)=1) + \\
&\lambda \cdot P_r(l(k+1)=1) \cdot P_r(l'(k)=1, l'(k+1)=0)
\end{aligned} \tag{42}$$

To evaluate the different probabilities it is necessary to define the following sets of codewords W. Let W_{0*} and W_{1*} be the sets of the codewords whose *first* bit is 0 and 1 respectively. Similarly let W_{*0} and W_{*1} be the sets of the codewords whose *last* bit is 0 and 1 respectively. Then, for any α, β in $\{0, 1\}$ it is:

$$P_r(l(k) = \alpha,\, l(k+1) = \beta) = \sum_{v \in W_{*\alpha},\, w \in W_{*\beta}} P_r(\boldsymbol{L}(k) = \boldsymbol{v}\,,\, \boldsymbol{L}(k+1) = \boldsymbol{w})$$
$$= \sum_{v \in W_{*\alpha},\, w \in W_{*\beta}} P_r(\boldsymbol{L}(k+1) = \boldsymbol{w} \mid \boldsymbol{L}(k) = \boldsymbol{v}) \cdot P_r(L(k) = \boldsymbol{v}) \tag{43}$$

Similarly it is,

$$P_r(l'(k) = \alpha,\, l'(k+1) = \beta) = \sum_{v \in W_{\alpha*},\, w \in W_{\beta*}} P_r(\boldsymbol{L}'(k+1) = \boldsymbol{w} \mid \boldsymbol{L}'(k) = \boldsymbol{v}) \cdot P_r(\boldsymbol{L}'(k) = \boldsymbol{v}) \tag{44}$$

To get a compact expression for $\overline{J(k)}$ we need a few more definitions. For $\alpha = 0, 1$ and $i = 1, \ldots, M$ let

$$h^i_{*\alpha} = \left\{ \begin{matrix} 1 \text{ if } \mathrm{w}_i \in W_{*\alpha} \\ 0 \text{ if } \mathrm{w}_i \notin W_{*\alpha} \end{matrix} \right\}, \qquad h^i_{\alpha*} = \left\{ \begin{matrix} 1 \text{ if } \mathrm{w}_i \in W_{\alpha*} \\ 0 \text{ if } \mathrm{w}_i \notin W_{\alpha*} \end{matrix} \right\}$$

Then (43) and (44) can be written as

$$P_r(l(k) = \alpha,\, l(k+1) = \beta) = \sum_{i,j=1}^{M} P_{i,j} \cdot p_i(k) \cdot h^i_{*\alpha} \cdot h^j_{*\beta} \tag{45}$$

and

$$P_r(l'(k) = \alpha,\, l'(k+1) = \beta) = \sum_{i,j=1}^{M} P_{i,j} \cdot p_i(k) \cdot h^i_{\alpha*} \cdot h^j_{\beta*} \tag{46}$$

respectively. We also have that

$$P_r(l(k+1) = \alpha) = \sum_{i=1}^{M} p_i(k+1) \cdot h^i_{*\alpha} \tag{47}$$

and

$$P_r(l'(k+1) = \alpha) = \sum_{i=1}^{M} p_i(k+1) \cdot h^i_{\alpha*} \tag{48}$$

Finally we define the four diagonal matrices

$$\boldsymbol{H}_{*\alpha} = diag(h^1_{*\alpha}, h^2_{*\alpha}, \ldots, h^M_{*\alpha})$$

$$\boldsymbol{H}_{\alpha*} = diag(h^1_{\alpha*}, h^2_{\alpha*}, \ldots, h^M_{\alpha*})$$

Example: For the *TPC* scheme presented in Figure 2 we have the six codewords $\boldsymbol{w}_1 = 000$, $\boldsymbol{w}_2 = 001$, $\boldsymbol{w}_3 = 010$, $\boldsymbol{w}_4 = 101$, $\boldsymbol{w}_5 = 110$ and $\boldsymbol{w}_6 = 111$. Therefore it is:

$$W_{*0} = \{\boldsymbol{w}_1, \boldsymbol{w}_3, \boldsymbol{w}_5\} \qquad W_{0*} = \{\boldsymbol{w}_1, \boldsymbol{w}_2, \boldsymbol{w}_3\}$$
$$W_{*1} = \{\boldsymbol{w}_2, \boldsymbol{w}_3, \boldsymbol{w}_6\} \qquad W_{1*} = \{\boldsymbol{w}_4, \boldsymbol{w}_5, \boldsymbol{w}_6\}$$

and so,

$$\boldsymbol{H}_{*0} = \begin{bmatrix} 1 & & & & & 0 \\ & 0 & & & & \\ & & 1 & & & \\ & & & 0 & & \\ & & & & 1 & \\ 0 & & & & & 0 \end{bmatrix} \qquad \boldsymbol{H}_{*1} = \begin{bmatrix} 0 & & & & & 0 \\ & 1 & & & & \\ & & 0 & & & \\ & & & 1 & & \\ & & & & 0 & \\ 0 & & & & & 1 \end{bmatrix}$$

$$\boldsymbol{H}_{0*} = \begin{bmatrix} 1 & & & & & 0 \\ & 1 & & & & \\ & & 1 & & & \\ & & & 0 & & \\ & & & & 0 & \\ 0 & & & & & 0 \end{bmatrix} \qquad \boldsymbol{H}_{1*} = \begin{bmatrix} 0 & & & & & 0 \\ & 0 & & & & \\ & & 0 & & & \\ & & & 1 & & \\ & & & & 1 & \\ 0 & & & & & 1 \end{bmatrix} \tag{49}$$

From the definitions of $\boldsymbol{H}_{*\alpha}$, $\boldsymbol{H}_{\alpha*}$ and expressions (45) - (48) we get:

$$P_r(l(k) = \alpha, l(k+1) = \beta) = \boldsymbol{p}(k) \cdot \boldsymbol{H}_{*\alpha} \cdot \boldsymbol{P} \cdot \boldsymbol{H}_{*\beta} \cdot \underline{\mathbf{1}} \tag{50}$$

$$P_r(l'(k) = \alpha, l'(k+1) = \beta) = \boldsymbol{p}(k) \cdot \boldsymbol{H}_{\alpha*} \cdot \boldsymbol{P} \cdot \boldsymbol{H}_{\beta*} \cdot \underline{\mathbf{1}} \tag{51}$$

$$P_r(l(k+1) = \alpha) = \boldsymbol{p}(k+1) \cdot \boldsymbol{H}_{*\alpha} \cdot \underline{\mathbf{1}} \tag{52}$$

$$P_r(l'(k+1) = \alpha) = \boldsymbol{p}(k+1) \cdot \boldsymbol{H}_{\alpha*} \cdot \underline{\mathbf{1}} \tag{53}$$

Combining equations (50)-(53) and using (16) (the Chapman- Kolmogorov formula [36]) we get expression (54) for the expected value of the interaction energy between the adjacent lines l and

l' at the *k-th* transition.

$$\begin{aligned}\overline{J(k)} = \; & \lambda \cdot [\boldsymbol{p}(0) \cdot \boldsymbol{P}^{k+1} \cdot \boldsymbol{H}_{*0} \cdot \underline{1}] \cdot [\boldsymbol{p}(0) \cdot \boldsymbol{P}^{k} \cdot \boldsymbol{H}_{0*} \cdot \boldsymbol{P} \cdot \boldsymbol{H}_{1*} \cdot \underline{1}] \\ & + \lambda \cdot [\boldsymbol{p}(0) \cdot \boldsymbol{P}^{k+1} \cdot \boldsymbol{H}_{1*} \cdot \underline{1}] \cdot [\boldsymbol{p}(0) \cdot \boldsymbol{P}^{k} \cdot \boldsymbol{H}_{*1} \cdot \boldsymbol{P} \cdot \boldsymbol{H}_{*0} \cdot \underline{1}] \\ & + \lambda \cdot [\boldsymbol{p}(0) \cdot \boldsymbol{P}^{k+1} \cdot \boldsymbol{H}_{0*} \cdot \underline{1}] \cdot [\boldsymbol{p}(0) \cdot \boldsymbol{P}^{k} \cdot \boldsymbol{H}_{*0} \cdot \boldsymbol{P} \cdot \boldsymbol{H}_{*1} \cdot \underline{1}] \\ & + \lambda \cdot [\boldsymbol{p}(0) \cdot \boldsymbol{P}^{k+1} \cdot \boldsymbol{H}_{*1} \cdot \underline{1}] \cdot [\boldsymbol{p}(0) \cdot \boldsymbol{P}^{k} \cdot \boldsymbol{H}_{1*} \cdot \boldsymbol{P} \cdot \boldsymbol{H}_{0*} \cdot \underline{1}]\end{aligned} \tag{54}$$

Since we are interested in estimating the interaction energy over long periods of time it is necessary to introduce a new energy measure. We define the *time average expected interaction energy (TAEI)* as:

$$E_{ai} = \lim_{N \to \infty} \frac{1}{N} \cdot \sum_{k=1}^{N} \overline{J(k)} \tag{55}$$

To calculate the *time average expected interaction energy* between consecutive blocks of the partitioned coding scheme of Figure 9 the decomposition of the transition probability matrix $\boldsymbol{P} = \boldsymbol{A} \cdot \boldsymbol{\Delta} \cdot \boldsymbol{B}$ from theorem 1 is used. Again, $\boldsymbol{A} = [\underline{1}, \boldsymbol{a}_1, \ldots, \boldsymbol{a}_{M-1}]$ consists of the (right) eigenvectors of $\boldsymbol{P}$ (appropriately ordered) and $\boldsymbol{B} = [\boldsymbol{b}_0, \boldsymbol{b}_1, \ldots, \boldsymbol{b}_{M-1}]^T$ is the inverse of $\boldsymbol{A}$. Then,

$$\boldsymbol{P}^k = \underline{1} \cdot \boldsymbol{b}_0^T + \sum_{r=1}^{q-1} e^{\frac{2\pi i}{q} rk} \cdot \boldsymbol{a}_r \cdot \boldsymbol{b}_r^T + \boldsymbol{A} \cdot \begin{bmatrix} 0 & 0 \\ 0 & \boldsymbol{Y}^k \end{bmatrix} \cdot \boldsymbol{B} \tag{56}$$

From theorem 1 we have that $\boldsymbol{Y}^k \to 0$ as $k \to \infty$ and so the *1st* term of (54) becomes,

$$\boldsymbol{p}(0) \cdot \boldsymbol{P}^{k+1} \cdot \boldsymbol{H}_{*0} \cdot \underline{1} = \boldsymbol{b}_0^T \cdot \boldsymbol{H}_{*0} \cdot \underline{1} + \sum_{r=1}^{q-1} e^{\frac{2\pi i}{q} rk} \cdot \boldsymbol{p}(0) \cdot \boldsymbol{a}_r \cdot \boldsymbol{b}_r^T \cdot \boldsymbol{H}_{*0} \cdot \underline{1} + \varepsilon_1(k) \tag{57}$$

and

$$\begin{aligned}& \boldsymbol{p}(0) \cdot \boldsymbol{P}^{k} \cdot \boldsymbol{H}_{0*} \cdot \boldsymbol{P} \cdot \boldsymbol{H}_{1*} \cdot \underline{1} = \\ & = \boldsymbol{b}_0^T \cdot \boldsymbol{H}_{0*} \cdot \boldsymbol{P} \cdot \boldsymbol{H}_{1*} \cdot \underline{1} + \sum_{r=1}^{q-1} e^{\frac{2\pi i}{q} rk} \cdot \boldsymbol{p}(0) \cdot \boldsymbol{a}_r \cdot \boldsymbol{b}_r^T \cdot \boldsymbol{H}_{0*} \cdot \boldsymbol{P} \cdot \boldsymbol{H}_{1*} \cdot \underline{1} + \varepsilon_2(k)\end{aligned} \tag{58}$$

where $\varepsilon_1, \varepsilon_2 \to 0$ as $k \to \infty$. The rest of the terms in (55) can be expressed similarly. After lengthy algebraic manipulations it turns out that,

$$E_{ai} = \lambda \cdot \boldsymbol{b}_0^T \cdot [\boldsymbol{H}_{1*} \cdot \boldsymbol{P} \cdot \boldsymbol{H}_{0*} + \boldsymbol{H}_{*1} \cdot \boldsymbol{P} \cdot \boldsymbol{H}_{*0}] \cdot \underline{\mathbf{1}}$$
$$- 4 \cdot \sum_{r=1}^{q-1} \sin^2\left(\frac{\pi \cdot r}{q}\right) \cdot |\boldsymbol{p}(0) \cdot \boldsymbol{a}_r|^2 \cdot (\boldsymbol{b}_r^T \cdot \boldsymbol{H}_{*0} \cdot \underline{\mathbf{1}}) \cdot (\bar{\boldsymbol{b}}_r^T \cdot \boldsymbol{H}_{0*} \cdot \underline{\mathbf{1}}) \tag{59}$$

Where $\bar{\boldsymbol{b}}_r$ is the complex conjugate of the eigenvector $\boldsymbol{b}_r$ and q is the number of eigenvalues $\boldsymbol{P}$ of modulus one (Theorem 1). As it has been mentioned before, the case of $q = 1$ corresponds to *TPC*'s with connected transition graphs. This is always desirable in practice and so from now on we assume that $q = 1$. Then, formula (59) simplifies to the very compact one:

$$E_{ai} = \lambda \cdot \boldsymbol{b}_0^T \cdot [\boldsymbol{H}_{1*} \cdot \boldsymbol{P} \cdot \boldsymbol{H}_{0*} + \boldsymbol{H}_{*1} \cdot \boldsymbol{P} \cdot \boldsymbol{H}_{*0}] \cdot \underline{\mathbf{1}} \tag{60}$$

Example: Following our example of the *TPC* scheme in Figure 2 and assuming uniformly distributed data we have the probability transition matrix $\boldsymbol{P}$ of expression (31) with the maximal eigenvector $\boldsymbol{b}_0$ given by (32). Then for $\lambda = 5$, if we replace (49) into expression (60) we get:

$$E_{ai} = \lambda \cdot \boldsymbol{b}_0^T \cdot [\boldsymbol{H}_{1*} \cdot \boldsymbol{P} \cdot \boldsymbol{H}_{0*} + \boldsymbol{H}_{*1} \cdot \boldsymbol{P} \cdot \boldsymbol{H}_{*0}] \cdot \underline{\mathbf{1}}$$
$$= 5 \cdot \begin{bmatrix} 0.250 \\ 0.187 \\ 0.062 \\ 0.062 \\ 0.187 \\ 0.250 \end{bmatrix}^T \cdot \left\{ \begin{bmatrix} 0 & & & & & 0 \\ & 0 & & & & \\ & & 0 & & & \\ & & & 1 & & \\ & & & & 1 & \\ 0 & & & & & 1 \end{bmatrix} \cdot \frac{1}{4} \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & & & & & 0 \\ & 1 & & & & \\ & & 1 & & & \\ & & & 0 & & \\ & & & & 0 & \\ 0 & & & & & 0 \end{bmatrix} + \right.$$
$$\left. + \begin{bmatrix} 0 & & & & & 0 \\ & 1 & & & & \\ & & 0 & & & \\ & & & 1 & & \\ & & & & 0 & \\ 0 & & & & & 1 \end{bmatrix} \cdot \frac{1}{4} \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & & & & & 0 \\ & 0 & & & & \\ & & 1 & & & \\ & & & 0 & & \\ & & & & 1 & \\ 0 & & & & & 0 \end{bmatrix} \right\} \cdot \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = 2.1875 \tag{61}$$

4.6 Total Energy Consumption of the Partitioned Coding Scheme

Having formulas for the *time average expected energy* of the individual coding blocks and also formulas the *time average expected interaction energy* between the blocks, the *total time average expected energy consumption (TTAEE)* (defined similarly) of the whole coding scheme in Figure 9 is given by (62).

$$E_T = n \cdot E_a + (n-1) \cdot E_{ai} \tag{62}$$

where n is the number of blocks. And for the case $q = 1$ we have,

$$E_T = n \cdot \boldsymbol{b}_0{}^T \cdot (\boldsymbol{T} \bullet \boldsymbol{P}) \cdot \underline{\mathbf{1}} + \lambda \cdot (n-1) \cdot \boldsymbol{b}_0{}^T \cdot [\boldsymbol{H}_{1*} \cdot \boldsymbol{P} \cdot \boldsymbol{H}_{0*} + \boldsymbol{H}_{*1} \cdot \boldsymbol{P} \cdot \boldsymbol{H}_{*0}] \cdot \underline{\mathbf{1}} \tag{63}$$

Note that this energy is normalized with respect to V_{dd} and C_L, so the actual energy consumed is $V_{dd}^2 \times C_L \times E_T$. Figure 10 presents the energy saving of the *TPC* given by the algorithm for some combinations of the parameters, *number of blocks n*, *data lines m*, *bus lines m+a*, *lambda* λ.

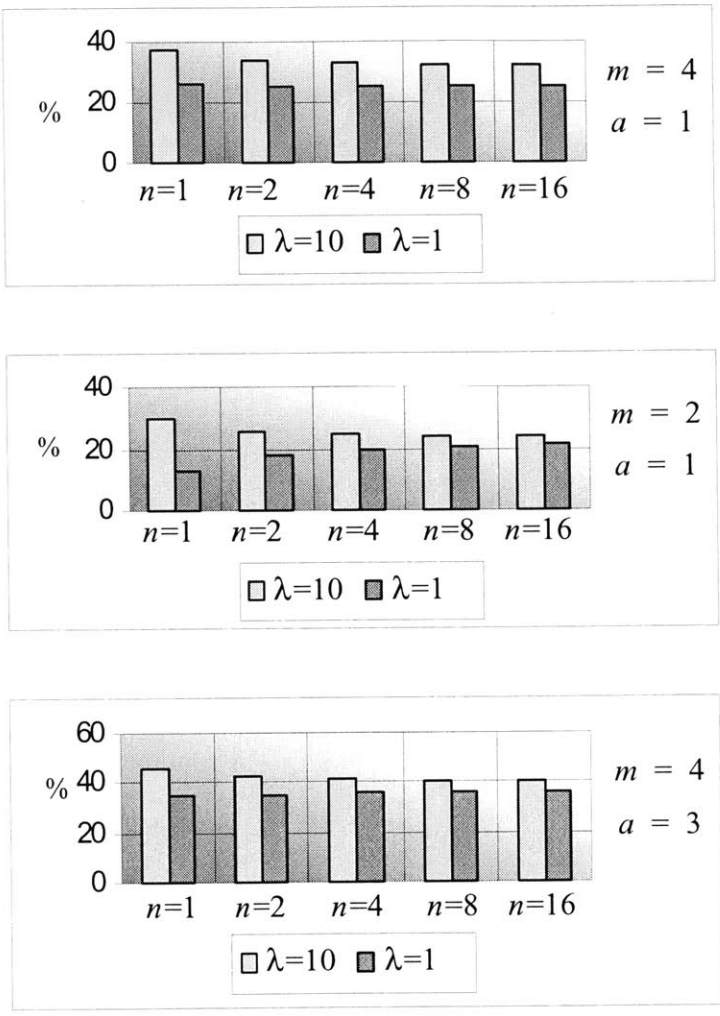

Figure 10: Percentage of Energy Saving
using Partitioned Coding Schemes

5. Conclusions

The purpose of this work is to provide a mathematical framework for designing coding schemes for low power. Given a specific bus structure and estimating or measuring the distribution of the data, the two proposed algorithms provide coding schemes that in theory can result in significant energy reduction. For practical application of a specific scheme, or a set of *TPC* schemes, further work will be required to estimate the complexity and the energy overhead of the encoder and decoder. Both complexity and overhead depend strongly on the particular technology as well as the specific circuit implementation that will be used. Another issue that has to be addressed is that coding schemes are non-catastrophic, or the encoder/decoder have a periodic resetting mechanism.

References

[1] H. Zhang, J. Rabaey, "Low swing interconnect interface circuits", *IEEE/ACM International Symposium on Low Power Electronics and Design, pp. 161-166, August 1998.*

[2] Y. Nakagome, K. Itoh, M. Isoda, K. Takeuchi, M. Aoki, "Sub-1-V swing internal bus architecture for future low-power ULSI's", *IEEE Journal of Solid-State Circuits, pp. 414-419, April 1993.*

[3] M. Hiraki, H. Kojima, H. Misawa, T. Akazawa, Y. Hatano, "Data-dependent logic swing internal bus architecture for ultralow-power LSI's", *IEEE Journal of Solid-State Circuits, pp. 397-401, April. 1995.*

[4] E. Kush, J. Rabaey, "Low-Energy Embedded FPGA Structures", *ISLPED '98. August '98 Monterey USA.*

[5] K. Khoo, A. Willson Jr., "Charge recovery on a databus", *IEEE/ACM International Symposium on Low Power Electronics and Design, pp. 185-189, 1995.*

[6] B. Bishop, M. Irwin, "Databus charge recovery: Practical consideration", *International Symposium on Low Power Electronics and Design, pp. 85-87, August 1999.*

[7] H. Yamauchi, H. Akamatsu, T. Fujita, "An asymptotically zero power charge-recycling bus architecture for battery-operated ultrahigh data rate ULSI's", *IEEE Journal of Solid-State Circuits, pp. 423-431, April. 1995.*

[8] M. Stan, W. Burleson, "Low-power encodings for global communication in cmos VLSI", *IEEE Transactions on VLSI Systems, pp. 49-58, Vol. 5, No. 4, Dec. 1997.*

[9] M. Stan, W. Burleson, "Two-dimensional codes for low-power", *Proc. of the International Symposium on Low Power Electronics Design, pp. 335-340, Aug. 1996.*

[10] P. Panda, N. Dull, "Reducing address bus transitions for low power memory mapping", *Proc. European Design and Test ConI Mar. 1996, pp. 63-37.*

[11] E. Musoll, T. Lang, J. Cortadella, "Working-zone encoding for reducing the energy in micro-processor address buses", *IEEE Trans. on VLSI Systems, 6(4):568-572, Dec. 1998.*

[12] W. Fomaciari, D. Sciuto, C. Silvano, "Power estimation for architectural exploration of HW/ SW communication on system-Ievel buses", *Int. Workshop on Hardware/Software Codesign 1999, pp. 152- 156.*

[13] S. Ramprasad, N. Shanbhag, I. Hajj, "A coding framework for low-power address and data busses", *IEEE Transactions on VLSI Systems, pp. 212-221, Vol. 7, No. 2, June 1999.*

[14] Y. Shin, S. Chae, K. Choi, "Reduction of bus transitions with partial bus-invert coding", *Electronics letters Vol.34, No.7, Apr. 1998.*

[15] Y. Zhang, W. Ye, M. Irwin, "An alternative architecture for on- chip global interconnect: segmented bus power modeling", *Asilomar Conf on Signals, Systems and Computers, 1998, pp. 1062-1065.*

[16] L. Benini, G. De Micheli, E. Macii, D. Sciuto, C. Silvano, "Asymptotic zero-transition activity encoding for address busses in low-power microprocessor-based systems", *Proc. of the Great Lakes Symp. VLSI 1997, pp. 77-82.*

[17] L. Benini, A. Macii, E. Macii, M. Poncino, R. Scarsi, "Synthesis of low-overhead interfaces for power-efficient communication over wide buses", *In Proc. AcMnEEE Design Automation Conf, pages 128-133, 1999.*

[18] S. Ramprasad, N. Shanbhag, I. Hajj, "Information-Theoretic Bounds on Average Signal Transition Activity", *IEEE Trans. on VLSI, Vol. 7, No. 3, Sept. 1999.*

[19] T. Sakurai, "Design Challenges for 0.1μm and Beyond", *Asia and South Pacific Design Automation Conference 2001, pp. 293-296.*

[20] P. Sotiriadis, A. Chandrakasan, "A Bus Energy Model For Deep Sub-Micron Technology", *To appear in the transactions on VLSI.*

[21] P. Sotiriadis, A. Chandrakasan, "Low Power Bus Coding Techniques Considering Inter-wire Capacitances", *Proceedings of CICC, May 2000.*

[22] P. Sotiriadis, A. Wang, A. Chandrakasan, "Transition Pattern Coding: An approach to reduce Energy in Interconnect", *ESSCIRC' 2000, Stockholm, Sweden, Sept. 2000.*

[23] K. Kim, K. Baek, N. Shanbhag, C. Liu, S. Kang, "Coupling-Driven Signal Encoding Scheme for Low-Power Interface Design", *Proceedings of ICCAD, Nov. 2000*

[24] Y. Shin, T. Sakurai, "Coupling-Driven Bus Design for Low-Power Application-Specific Systems", *Design Automation Conference 2001, pp.750-753.*

[25] J. Henkel, H. Lekatsas, "A^2BC: Adaptive Address Bus Coding for Low Power Deep Sub-Micron Design", *Design Automation Conference 2001, pp. 744-749.*

[26] C. Taylor, S. Dey, Y. Zhao, "Modeling and Minimization of Interconnect Energy Dissipation in Nanometer Technologies", *Design Automation Conference 2001, pp.*

[27] P. Sotiriadis, A. Chandrakasan, "Bus Energy Minimization by Transition Pattern Coding (*TPC*) in Deep Sub-Micron Technologies", *Proceedings of ICCAD, Nov. 2000.*

[28] J. Davis, J. Meindl, "Compact Distributed RLC Interconnect Models Part II: Coupled Line Transient Expressions and Peak Crosstalk in Multilevel Networks", *IEEE transactions on electron devices, Vol. 47, No. 11, Nov. 2000.*

[29] L. Pileggi, "Coping with RC(L) interconnect design headaches", *Proceedings of ICCAD, 1995.*

[30] S. Das, W. Smith, C. Paul, "Modeling of data bus structures using numerical methods", *International Symposium on Electromagnetic Compatibility, 1993, pp. 409-414.*

[31] C. Cheng, J. Lillis, S. Lin, N. Chang, *Interconnect Analysis and Synthesis*, John Wiley and Sons, 2000.

[32] H. Bakoglou, *Circuits, Interconnections, and Packaging in VLSI*, Addison-Wesley Pub. Co., 1990

[33] J. Rabaey, *Digital Integrated circuits*, Prentice Hall 1996.

[34] F. Harary, *Structural models: an introduction to the theory of directed graphs*, New York, Wiley 1965.

[35] F. Riesz and B.Nagy, *Functional Analysis*, Dover 1990.

[36] R. Gallager, *Discrete Stochastic Processes*, Kluwer Academic Publishers 1998.

[37] R. Horn and C. Johnson, *Matrix Analysis*, Cambridge University Press 1994.

[38] W. Rudin, *Principles of Mathematical Analysis*, McGraw-Hill, 1964

[39] A. Chandrakasan, R. Brodersen, *Low power CMOS Design*, IEEE Press, 1998.

Chapter 6

A Charge Recycling Technique For Power Reduction In Deep Sub-Micron Buses

1. Introduction

In this Chapter we study charge recycling as a technique for power reduction in deep sub-micron buses. The basic idea of charge recycling is illustrated through the following example.

Suppose we have a power supply of voltage V_{dd} and two capacitors of the same size C, both of which have one of their terminals connected to ground. Suppose also that, initially, capacitor #1 is charged to V_{dd} and capacitor #2 is completely discharged. We call this combination state A. Now assume we want to end up with state B that is when the charges have been reversed, i.e. when capacitor #2 is charged to V_{dd} and capacitor #1 is completely discharged. There are at least two ways of going from state A to state B. One is, by brute force, to discharge capacitor #1 and then charge capacitor #2 using the power supply. This implies an energy loss of CV_{dd}^2. This is because we loose $CV_{dd}^2/2$ by discharging #1 and $CV_{dd}^2/2$ more by charging #2 directly from the source.

An alternative way of changing state is first to connect the two capacitors together. This implies an energy loss of $CV_{dd}^2/4$ and that the two capacitors are finally charged at $V_{dd}/2$. Then, we discharge #1 completely, and so loose $CV_{dd}^2/8$, and also charge #2 from the source, which results to an additional $CV_{dd}^2/8$ of energy loss. The total energy loss for the second strategy is only $CV_{dd}^2/2$, that is, the half of the energy loss when we change the state by brute force.

This simple example is the basic principle of charge recycling and adiabatic charging. The distinction between the terms in practice is that adiabatic charging is mostly associated with the use of inductors while charge recycling is not.

Charge recycling strategies have been proposed as approaches to reduce the power dissipation in buses [3]-[5]. All of them have used a bus model where there is no coupling between the lines. As it was discussed in Chapter 2, the issue with the modern deep sub-micron technologies is that they introduce strong coupling between neighboring lines. This is especially true in buses. Therefore a charge recycling approach is needed that is appropriate for deep sub-micron technologies. The purpose of the present chapter is to discuss such a Charge Recycling Technique (CRT), present the mathematical limits of the possible energy reduction and suggest an efficient circuit design for the necessary control network.

In theory this technique can result in up to 50% power reduction. It is even more powerful when combined with Bus Invert coding [6]. The specific circuit, the CRT driver, that implements the CRT, is a modular design that can be parallelized in order to operate with buses of arbitrary size. For the HSPICE simulations the circuit was used on a 4-line and an 8-line bus resulting in a net energy saving (taking into account the energy consumption of the circuit itself) of up to 32%. Application of the technique on large buses of 32 or 64 lines or capacitively heavy interconnect networks in Field Programmable Gate Arrays (FPGAs) are expected to result to even higher net energy savings [11].

2. Bus and Drivers Models

We use the bus model of Figure 7 in Chapter 1 to demonstrate the Charge Recycling Technique (CRT). We assume it has n lines, and the fringing capacitance C_F is negligible. For convenience it is shown below along with the drivers of the lines.

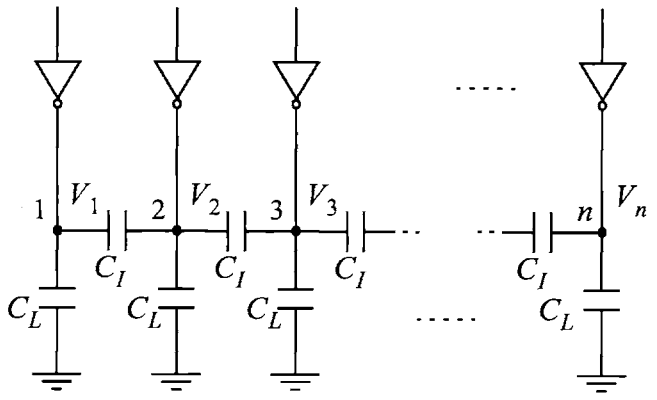

Figure 1: Deep sub-micron bus energy model with drivers

C_L is the capacitance between each line and the ground, and C_I is the capacitance between adjacent lines. Again we use the technology parameter lambda, $\lambda = C_I/C_L$. Recall that for .18μ technology, λ can be up to 5 and for .13μ technology it can be up to 9. Also recall from Chapter 2 that under some mild assumptions, the model of Figure 1 has the same energy behavior as its distributed version. To simplify the theoretical analysis we set $V_{dd} = 1$. All energies calculated under this assumption must be multiplied by the factor V_{dd}^2 to give the real energy value. The drivers are modeled as in Figure 2, [8].

The resistors $R_i^P(t)$ and $R_i^N(t)$ correspond to the PMOS and NMOS transistors of the drivers. Their values can be almost arbitrary functions of time. The switches s_i and $\bar{s}_i$ are complementary and their status corresponds to the desirable values of the lines. The parasitic capacitances of the drivers outputs can be lumped into C_L.

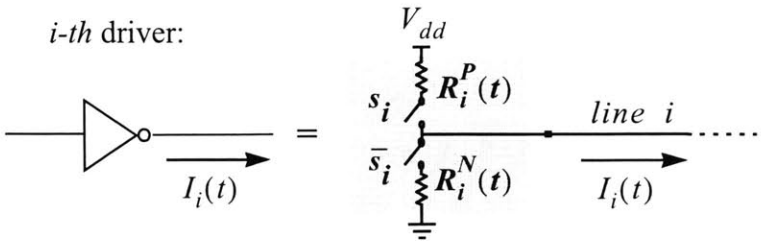

Figure 2: Bus Driver Model

3. The Charge Recycling Technique (CRT)

In this section we present the two steps of the charge Recycling Technique (CRT). Let T be the clock cycle period of the bus. New data is transmitted through every T seconds. We divide the time interval $[0, T]$ into the sub-intervals Int_1 and Int_2, the two steps of CRT. The two steps are related in time as in Figure 3.

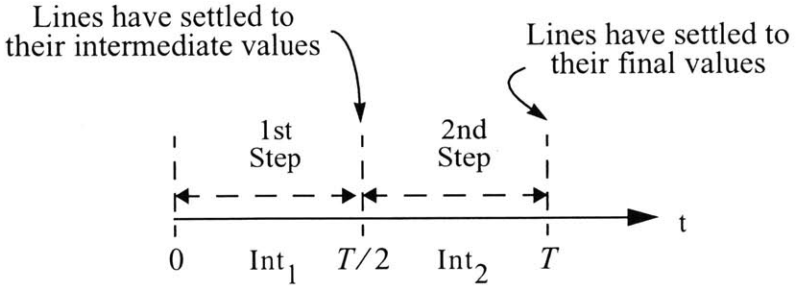

Figure 3: Timing of CRT

Suppose now that during the clock cycle the bus transitions from its current values, $x = [x_1, x_2, \ldots, x_n]^T$, to its new values, $y = [y_1, y_2, \ldots, y_n]^T$ (the values x_i, y_i correspond to line i). Since we have done the normalization, $V_{dd} = 1$, all entries x_i and y_i belong to $\{0, 1\}$. The voltages of the lines as functions of time $t \in [0, T]$ are denoted by $V = [V_1, V_2, \ldots, V_n]^T$.

At $t = 0$ and $t = T$ it is $V(0) = x$ and $V(T) = y$ respectively. The CRT is presented in Figure 4 with the modified driving circuit. We agree that switch w_i takes the value 0 if the node i is connected to the output of driver i, and the value 1, if the node i is connected to the common node q. During Int_1 the lines that change logical values (during the transition $x \rightarrow y$) are connected to common node q and not to their drivers. The lines retaining their logical values *remain connected to their drivers*. (This is a difference to the strategy in [5]. If there is coupling between the lines, then, keeping the ones retaining their values connected to their drivers during the recycling phase or not results in different energy patterns.). During Int_2 all lines are connected to their drivers.

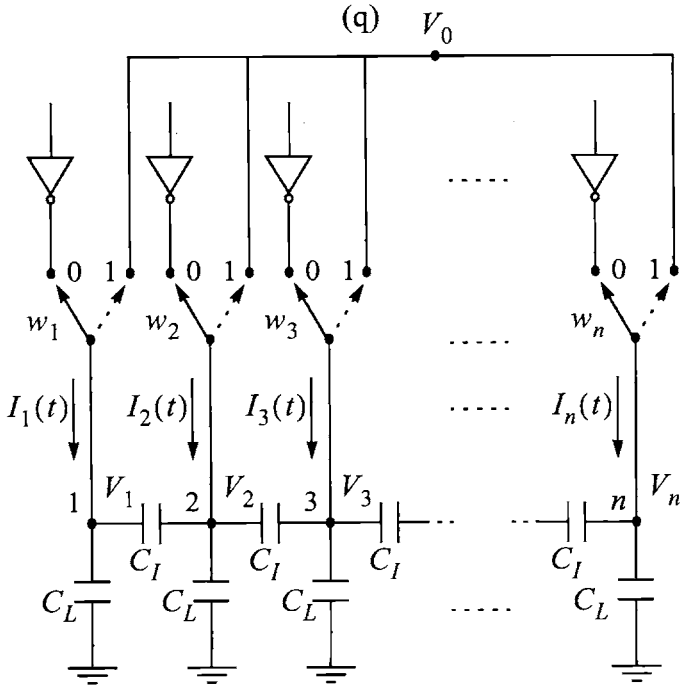

Figure 4: CRT - Network connections

3.1 First Step (Int 1)

For every line $i = 1, \ldots, n$ we set, $d_i = x_i \oplus y_i$. We also define the vector $d = x \oplus y$ where $d = [d_1, d_2, \ldots, d_n]^T$ and the diagonal matrix:

$$D = diag(d_1, d_2, \ldots, d_n) \tag{1}$$

During the transition, line *i* changes value if and only if $d_i = 1$. According to the CRT, during the time interval $Int_1 = (0, T/2]$ the lines with changing values are connected to node *q* and the rest remain connected to their drivers. Therefore the *i-th* switch must have the value:

$$w_i = d_i = x_i \oplus y_i$$

and the network must be configured appropriately. For example lets assume that $n = 4$, $x = [1, 0, 0, 1]^T$ and $y = [1, 1, 0, 0]^T$. Then $d = [0, 1, 0, 1]^T$ and during Int_1 the network is configured as in Figure 5.

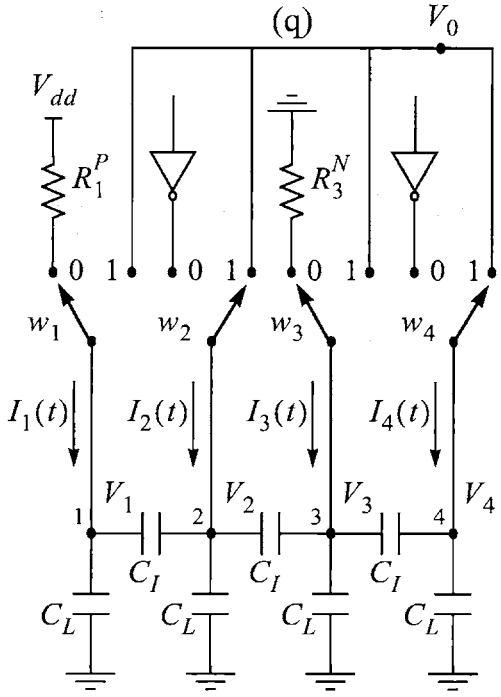

Figure 5: Example (*n*=4)

During Int_1 the dynamics of the network satisfies the set of differential equations (see Figure 4),

$$\begin{aligned} I_1 &= C_L \cdot \dot{V}_1 + C_I \cdot (\dot{V}_1 - \dot{V}_2) \\ I_k &= C_I \cdot (\dot{V}_k - \dot{V}_{k-1}) + C_L \cdot \dot{V}_1 + C_I \cdot (\dot{V}_k - \dot{V}_{k+1}), \quad 1 < k < n \\ I_n &= C_L \cdot \dot{V}_n + C_I \cdot (\dot{V}_n - \dot{V}_{n-1}) \end{aligned} \tag{2}$$

If we define the currents vector $I = [I_1, I_2, \ldots, I_n]^T$ and the $n \times n$ capacitance conductance matrix of the network shown in Figure 4, that is:

$$C_T = \begin{bmatrix} 1+\lambda & -\lambda & 0 & \ldots & 0 \\ -\lambda & 1+2\lambda & -\lambda & : & 0 \\ 0 & -\lambda & & : & : \\ : & : & : & 1+2\lambda & -\lambda \\ 0 & 0 & \ldots & -\lambda & 1+\lambda \end{bmatrix} \cdot C_L \tag{3}$$

then, equations (2) can be written in the compact form:

$$C_T \cdot \dot{V} = I \tag{4}$$

Now note that for $i = 1, \ldots, n$ the current $I_i(t)$ is drawn from the driver if $d_i = 0$ and from the node q if $d_i = 1$. The charge conservation at node q implies that $\sum_{i\,:\,d_i = 1} I_i(t) = 0$ or in vector form:

$$d^T \cdot I = 0 \tag{5}$$

Replacing (4) into (5) we the identity, $d^T \cdot C_T \cdot \dot{V} = 0$. Integrating it over the time interval $Int_1 = (0, T/2\,]$ of step 1 gives:

$$d^T \cdot C_T \cdot (V(\tfrac{T}{2}) - V(0)) = 0 \tag{6}$$

where $V(0) = x = [x_1, x_2, \ldots, x_n]^T$ are the initial conditions of the lines and $V(\frac{T}{2}) = [V_1(\frac{T}{2}), V_2(\frac{T}{2}), \ldots, V_n(\frac{T}{2})]^T$ are the intermediate ones.

Here we make the assumption that the time length $T/2$ is sufficient for the voltages of the network to settle. This assumption is reasonable for the current technology and is always used in the analysis of charge redistribution (and adiabatic) techniques. So for $i = 1, \ldots, n$ the voltage $V_i(\frac{T}{2})$ is either x_i if $d_i = 0$ or $z \equiv V_q(\frac{T}{2})$ if $d_i = 1$. The value z is of course the same for all lines that change logical value. Algebraically we have that $V_i(\frac{T}{2}) = (1 - d_i) \cdot x_i + d_i \cdot z$, or in vector form that:

$$V(\tfrac{T}{2}) = (I - D) \cdot x + z \cdot d \tag{7}$$

where I is the $n \times n$ identity matrix. The matrix D and the vector d are as defined before. From (6), (7) and $V(0) = x$ we get:

$$(d^T \cdot C_T \cdot d) \cdot z = d^T \cdot C_T \cdot D \cdot x \tag{8}$$

Now note that matrix C_T is positive definite. This implies that the quantity $(d^T \cdot C_T \cdot d)$ is positive if and only if $(d \neq 0)$ or equivalently, if and only if at least one line changes value during the transition $x \rightarrow y$. If there is no change during the transition then the energy dissipated during the transition is zero. For now we assume that at least one line changes value. Then from (8) we get,

$$z = V_o(\tfrac{T}{2}) = \frac{d^T \cdot C_T \cdot D \cdot x}{d^T \cdot C_T \cdot d} \tag{9}$$

3.2 Energy Dissipation on Step 1

Here we evaluate the energy that is **drawn from V_{dd}** on the first step of CRT. The current $I_{V_{dd}}(t)$ drawn from V_{dd} during (Int_1) is the sum of the currents drawn by the lines that do *not* change logical value during the transition *and* remain connected to V_{dd} through their drivers, i.e the lines $i = 1, \ldots, n$ for which $x_i = y_i = 1$. So,

$$I_{V_{dd}}(t) = \sum_{\substack{i\,:\,x_i = 1 \\ \text{and } y_i = 1}} I_i(t)$$

$$= \sum_{i=1}^{n} x_i \cdot (1 - d_i) \cdot I_i(t)$$

which can also be written in matrix form as:

$$I_{V_{dd}}(t) = x^T \cdot (I\text{–}D) \cdot I(t) \tag{10}$$

(Symbol I is used for both the current vector and the identity matrix. Is should be clear what I represents each time). Using equation (4) and (10) we have that,

$$I_{V_{dd}}(t) = x^T \cdot (I - D) \cdot C_T \cdot \dot{V} \tag{11}$$

Because of the normalization $V_{dd} = 1$ the energy drawn from V_{dd} during step 1 is:

$$E_1 = \int_0^{T/2} I_{V_{dd}}(t)dt$$

By replacing (11) in the integral we have $E_1 = x^T \cdot (I - D) \cdot C_T \cdot (V(\frac{T}{2}) - V(0))$. Finally we use (7) to get:

$$E_1 = x^T \cdot (I - D) \cdot C_T \cdot (z \cdot d - D \cdot x) \tag{12}$$

And by replacing z from (9) into (12) we have

$$E_1(x, y) = x^T \cdot (I - D) \cdot C_T \cdot \left\{ \left(\frac{d^T \cdot C_T \cdot D \cdot x}{d^T \cdot C_T \cdot d} \right) \cdot d - D \cdot x \right\} \tag{13}$$

if $d \neq 0$ and $E_1 = 0$ if $d = 0$.

3.3 Second Step (Int 2)

During the second step of the CRT, the time interval $Int_2 = (T/2, T]$, every line is connected to its driver (with the new value y_i). So for all $i = 1, \ldots, n$ it is $w_i = 0$. For the example with $n = 4$, $x = [1, 0, 0, 1]^T$ and $y = [1, 1, 0, 0]^T$, the network is configured as in Figure 6. Equation (4) holds during the second step as well.

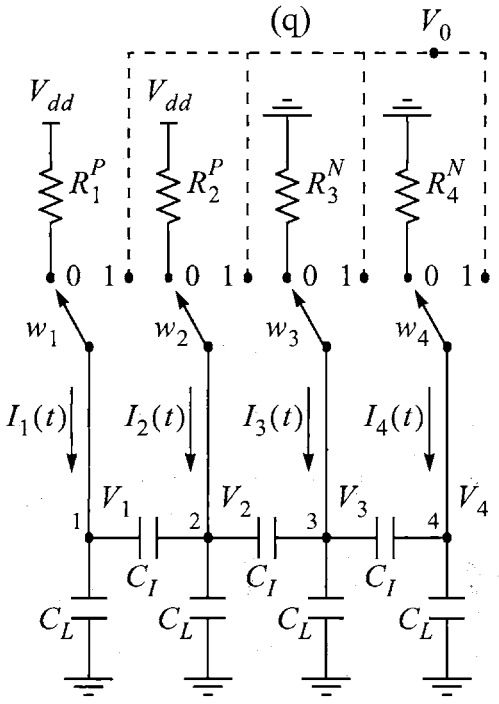

Figure 6: Step 2: Example (n=4)

3.4 Energy Dissipation on Step 2

During Int_2 the current drawn from V_{dd} equals the sum of the currents $I_i(t)$ of the lines connected to V_{dd} (through their drivers). So,

$$I_{V_{dd}}(t) = \sum_{i\,:\,y_i = 1} I_i(t) = \sum_{i=1}^{n} y_i \cdot I_i(t)$$

or in vector form,

$$I_{V_{dd}}(t) = y^T \cdot I(t) \tag{14}$$

Replacing (4) into (14) we get,

$$I_{V_{dd}}(t) = y^T \cdot C_T \cdot \dot{V} \tag{15}$$

The energy drawn from V_{dd} on step 2 is $E_2 = \int_{T/2}^{T} I_{V_{dd}}(t)dt$. Replacement of (15) into the integral gives,

$$E_2 = y^T \cdot C_T \cdot (V(T) - V(\tfrac{T}{2})) \tag{16}$$

Finally, $V(T) = y$, (7) and (9) imply,

$$E_2(x,y) = y^T \cdot C_T \cdot \left\{ y - (I - D) \cdot x - \left(\frac{d^T \cdot C_T \cdot D \cdot x}{d^T \cdot C_T \cdot d} \right) \cdot d \right\}. \tag{17}$$

4. Energy Properties of CRT

The total energy $E(x, y)$ drawn from V_{dd} during the transition $x \rightarrow y$ is of course $E(x, y) = E_1(x, y) + E_2(x, y)$. Using the identity $(I - D) \cdot x = (I - D) \cdot y$ and expressions (13) and (17) we get,

$$E(x, y) = y^T \cdot C_T \cdot (y - x) + y^T \cdot D \cdot C_T \cdot (D \cdot x - z \cdot d) \tag{18}$$

where z is given by (9). The first term of the right part of (18) equals the energy drawn from V_{dd} by the bus during the transition $x \rightarrow y$ when no charge recycling is applied [7]. The other terms correspond to the energy difference (savings) due to CRT.

□ with CRT ▨ without CRT

(x_1, x_2, x_3)	(y_1, y_2, y_3)	000	001	010	011	100	101	110	111
	000	0	6	11	7	6	12	7	3
		0	6	11	7	6	12	7	3
	001	0	0	10	6	3	6	6.3	2
		0	0	16	6	6	6	12	2
	010	0	5.1	0	1	5.1	11	1	2
		0	11	0	1	11	22	1	2
	011	0	5	5	0	5.3	10	3	1
		0	5	5	0	11	16	6	1
	100	0	3	10	6.3	0	6	6	2
		0	6	16	12	0	6	6	2
	101	0	0	10	5.1	0	0	5.1	1
		0	0	21	11	0	0	11	1
	110	0	5.3	5	3	5	10	0	1
		0	11	5	6	5	16	0	1
	111	0	5	10	5	5	10	5	0
		0	5	10	5	5	10	5	0

Table 1: Transition energies with and without the CRT

For a better intuition on how CRT influences the bus energy transition patterns, table 1 presents the case of a three line bus $n = 3$ when $\lambda = 5$. Five is a representative value of λ for the case of .18μ technologies (with minimal distance between the wires). For simplicity we set $C_L = V_{dd} = 1$.

For each transition $(x_1, x_2, x_3) \rightarrow (y_1, y_2, y_3)$ the shadowed value (below) is the energy cost *without* CRT, equal to $y^T \cdot C_T \cdot (y - x)$. The numbers on the white background (above) are the energies *with* CRT, i.e. to the values given by (18). The energy *with* CRT is always smaller. Also, the highest percentage of energy reduction occurs in the most expensive transitions $010 \rightarrow 101$ and $101 \rightarrow 010$ where adjacent lines transit in the opposite direction and the interline capacitances are charged by $2 \times V_{dd}$.

5. Energy Reduction

The result for the transition energy, formula (18), allows us to estimate numerically the expected energy drawn by the bus when the CRT is used. We do this for the case of uniformly distributed i.i.d. data.

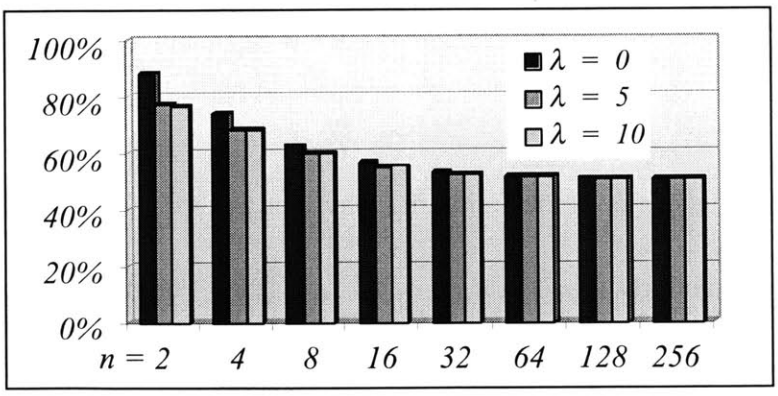

Figure 7: Energy with CRT / Energy without CRT

In Figure 7 we see the expected energy using CRT as a percentage of the expected energy without CRT for the cases of $n = 2,4,8,16,32,64,128,256$ and $\lambda = 0, 5, 10$. The figure suggests that for the number of lines $n = 32, 64, 128, 256$ the energy drawn from V_{dd} can be reduced to one half using CRT. Also, the results are independent of the capacitance to ground C_L, and they slightly improve when λ increases. In general λ tends to increase with technology scaling.

6. CRT and Bus-Invert

In the previous sections we showed how CRT reduces energy consumption. In Figure 8 we present an architecture where CRT is combined with Bus-Invert coding [6].

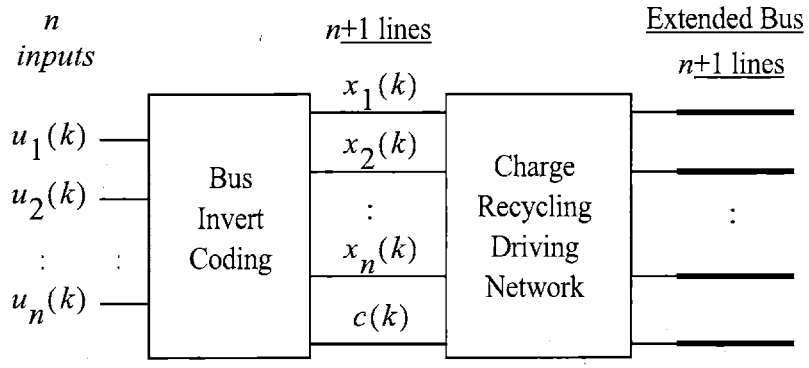

Figure 8: Combination of CRT with Bus Invert coding

The Bus Invert coding works in the following way. Let $u(k) = [u_1(k), u_2(k), \ldots, u_n(k)]^T$ be the new input vector and $x(k) = [x_1(k), x_2(k), \ldots, x_n(k)]^T$ be the new vector of the values of the lines. If the vector $u(k) \oplus x(k-1)$ contains more than $n/2$ ones then we set $x(k) = \overline{u(k)}$ and $c(k) = 1$, otherwise we set $x(k) = u(k)$ and $c(k) = 0$. The combined performance of CRT and Bus Invert is shown in Figure 9. We see a small improvement compared to the results of Figure 7. For buses with 16 lines or more the energy saving is more than *50%*.

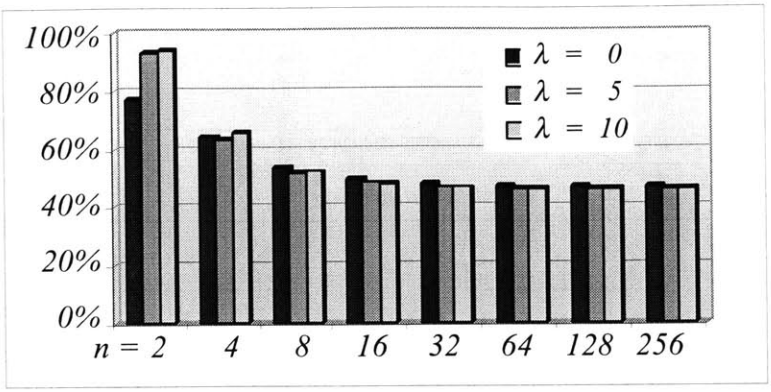

Figure 9: Energy with CRT and Bus Invert / Energy without them

7. A Circuit for CRT Drivers

To verify CRT we designed a circuit that implements the conceptual network of Figure 4. Our circuit implementation consisted of the bus and the CRT drivers of the lines. The CRT driver detects the transition of the line and connects it either to the common node (q) or to its regular driver (chain of inverters). The proposed CRT driver was designed and laid out in .18μ technology and its schematic is shown in Figure 10. Using this driver we tested CRT for a 4-line and an 8-line bus. The CRT driver operates as follows. The switches w_1, w_2, ... in Figure 4 are realized here by the pair of transmission gates. The charge recycling phase begins when CLK becomes 1. A negative spike appears at the output of the XNOR gate if the input x_i changes value. This sets the latch and connects the line to the common node q through the transmission gate. The charge recycling phase ends when CLK becomes 0. This resets the latch, isolates the line from the common node (q) and connects it to the buffer chain. If the input x_i does not make a transition, the latch remains reset during the whole clock cycle and the line remains connected to the buffer chain. The same circuit can be used unchanged for buses with arbitrary number of lines. For further details on the circuit implementation and layout the reader is referred to [10].

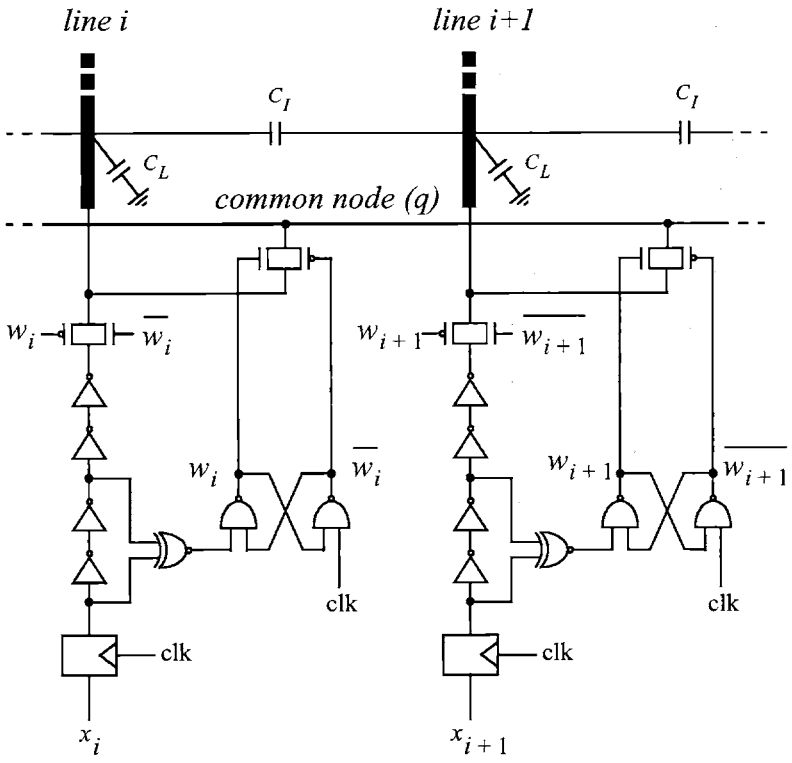

Figure 10: Efficient CRT - Driver

8. Simulation and Results

CRT drivers of Figure 10 were used to drive the lines of a four and an eight line bus, $n = 4, n = 8$. A netlist was extracted from the layout of the drivers for the simulation with HSPICE. The lines were modeled as in Figure 1 and for the capacitor C_L we used the values $50fF, 100fF, 150fF$ and $200fF$. Note that these values could represent not only the line capacitors but all the loads as well. This is particularly the case of reconfigurable interconnect networks (e.g. in FPGAs) where long buses are loaded by the parasitic capacitances of several mosfets resulting to total capacitive loads of the size of a few picofarads [11]. The clock frequency in the simulations was set to $100Mhz$ and the buses were fed with uniformly distributed i.i.d. sequences of data. In Figure 11 we see the average energy per cycle of the four line bus (left) and the eight line bus (right). The curves in the graphs showing higher energy consumption correspond to the standard buses. The curves showing lower consumption correspond to buses with CRT drivers.

In Figure 12 we see the average energy using CRT as a percentage of the average energy without CRT for the 4-line and 8-line buses. Again, the ratios are parametrized to C_L. The flat lines correspond to the minimum possible ratios resulting from the theoretical analysis and shown in Figure 7.

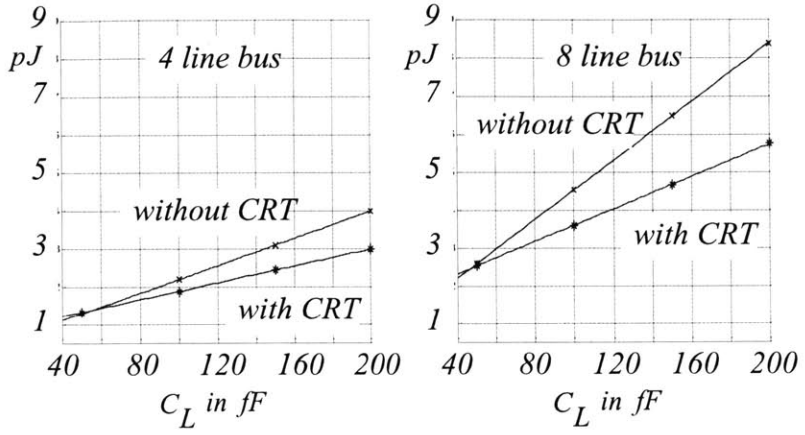

Figure 11: Average energy per cycle of a 4 and 8 line buses with and without CRT

As it should be expected, for higher capacitive loads we get higher percentages of energy saving. This is because the average energy per cycle of the additional circuitry of the drivers is relatively independent of the loads. For larger loads this additional energy becomes less significant.

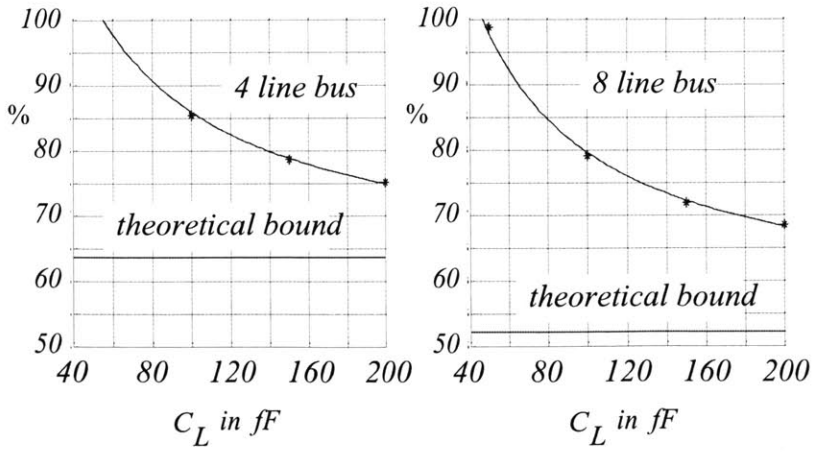

Figure 12: Normalized energy using CRT (HSPICE simulation)

Of course, as with any other charge recycling scheme, the CRT drivers should also replace any intermediate buffer stage of the bus. Finally, it is interesting to look at the waveforms of the individual lines during the two steps of the CRT. Figure 13 shows the waveforms of the line voltages of the 4-line bus. In this particular case, one line experiences a $0 \rightarrow 1$ transition and the rest three lines make a $0 \rightarrow 1$ transition. Since all lines transit they are all connected first to the common node q. The final voltage at node q during the charge redistribution period is of course z (equation (9)) and correspond to the converging point of the waveforms at time $T/2 = 5ns$. It is interesting to note that for an individual transition the maximum energy saving with CRT occurs when all lines transition and adjacent lines transition in opposite directions. This generalizes to the fact that CRT preforms very well when the sequences of the transitions of adjacent lines are negatively correlated.

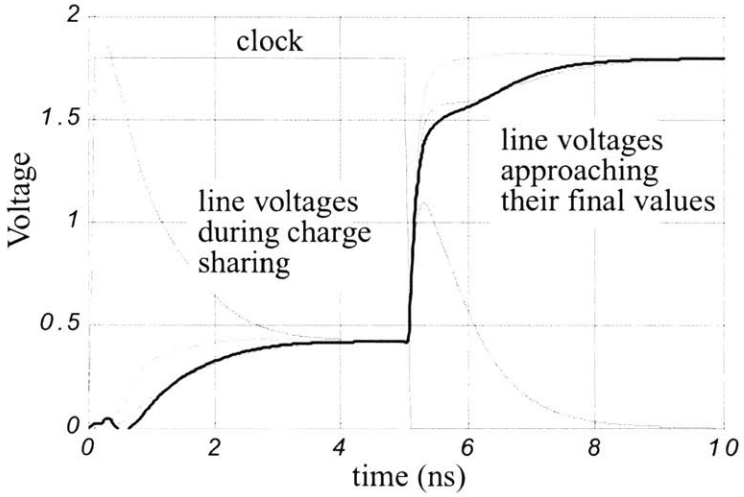

Figure 13: Line voltage waveforms during the two steps of CRT

9. Conclusions

The Charge Recycling Technique (CRT), proposed and analyzed, is appropriate for deep submicron buses. The theoretical evaluation of its performance showed up to 50% possible energy reduction. When combined with Bus Invert coding it becomes even more efficient. The line driver designed to implement CRT was used with an 8-line bus demonstrating net energy savings of up to 32%. Larger buses of 32 or 64 lines are expected to have higher energy savings.

10. References

[1] H. Zhang, J. Rabaey, "Low swing interconnect interface circuits", *IEEE/ACM International Symposium on Low Power Electronics and Design, pp. 161-166, August 1998.*

[2] Y. Nakagome, K. Itoh, M. Isoda, K. Takeuchi, M. Aoki, "Sub-1-V swing internal bus architecture for future low-power ULSI's", *IEEE Journal of Solid-State Circuits, pp. 414-419, April 1993.*

[3] H. Yamauchi, H. Akamatsu, T. Fujita,"An asymptotically zero power charge-recycling bus architecture for battery-operated ultrahigh data rate ULSI's", *Journal of Solid-State Circuits, IEEE, Vol. 30 Issue: 4, pp. 423 -431, April 1995.*

[4] K. Khoo, A. Willson Jr., "Charge recovery on a databus", *IEEE/ACM International Symposium on Low Power Electronics and Design, pp. 185-189, 1995.*

[5] B. Bishop, M. Irwin, "Databus charge recovery: Practical consideration", *International Symposium on Low Power Electronics and Design, pp. 85-87, August 1999.*

[6] M. Stan, W. Burleson, "Low-power encodings for global communication in cmos VLSI", *IEEE Transactions on VLSI Systems, pp. 49-58, Vol. 5, No. 4, Dec. 1997.*

[7] P. Sotiriadis, A. Wang, A. Chandrakasan, "Transition Pattern Coding: An approach to reduce Energy in Interconnect", *ESSCIRC 2000.*

[8] J. Rabaey, *Digital Integrated circuits*, Prentice Hall 1996.

[9] H. Bakoglou, *Circuits, Interconnections, and Packaging in VLSI*, Addison-Wesley Pub. Co., 1990

[10] T. Konstantakopoulos, *Implementation of Delay and Power Reduction in Deep Sub-Micron Buses Using Coding*, Master Thesis in EE, MIT, 2002.

[11] E. Kusse, J. Rabaey, "Low-Energy Embedded FPGA Structures", *ISLPED '98.*

Chapter 7

Coding For Increasing The Throughput and Speed of Deep Sub-Micron Buses

1. INTRODUCTION

Coding for speed is a new idea introduced in a recent paper [1]. The purpose of it, is to increase the data transmission rate through deep sub-micron buses. It is stressed that coding for speed is not related to data compression, which is widely used to shorten the data size. On the contrary, coding for speed expands data by *adding* redundancy! Nothing prevents of course from combining the two techniques. Why in buses? Why now? It is not surprising that coding for speed was introduced so recently. It was triggered by the modeling of modern technology buses and a new angle for studying their transient behavior.

Buses are formed by parallel lines that synchronously carry bits of data. As it was discussed in detail in Chapter 2, in older, obsolete technologies, these lines where modeled as grounded capacitors with no coupling between them. The drivers of the lines were responsible for charging and discharging the parasitic capacitors between zero and V_{dd} (the voltage of the power supply), in a way that corresponds to the transmitted data. The situation is shown below:

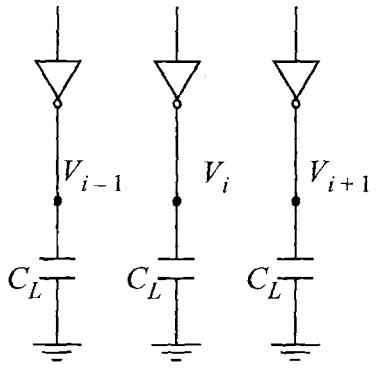

Figure 1: Older buses

In most designs the drivers are symmetric, in the sense that they have the same strength when charging and when discharging the capacitors. Since there is no inter-line coupling, all transitions $[\ldots, V_{i-1}^{old}, V_i^{old}, V_{i+1}^{old}, \ldots] \rightarrow [\ldots, V_{i-1}^{new}, V_i^{new}, V_{i+1}^{new}, \ldots]$ of the lines require the same amount of time to get completed (except when there is no transition at all!). In the modern deep sub-micron technologies the lines are neither lumped nor decoupled. But lets ignore for the moment their distributed behavior and concentrate on their coupling. The situation looks like that of Figure 2.

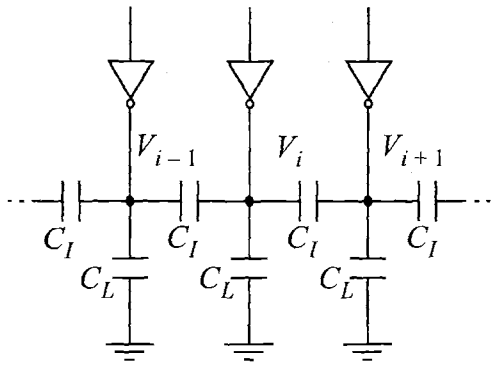

Figure 2: Deep sub-micron Buses (lumped approximation)

It is clear that the capacitive coupling influences directly the time needed for a transition of the lines to get completed. Moreover, what is important here that was not in the case of Figure 1, is that relative transitions of the lines strongly determine the completion time. For example we can think of the case $[\ldots, 0, 0, 0, \ldots] \rightarrow [\ldots, 1, 1, 1, \ldots]$ vs. $[\ldots, 1, 0, 1, \ldots] \rightarrow [\ldots, 0, 1, 0, \ldots]$.

In the first, the driver of the *i-th* line charges only C_L, while in the second transition, it charges $C_L + 4C_I$ because of the Miller effect.

In the work in [1], each line was studied individually, and the time required for the particular line to complete its transition was estimated as a function of the transitions of all lines in the bus. In this way, a transition time function was assigned to each of the lines. The transition time function of the whole bus was defined by taking the maximum of the values of the time functions of individual lines.

Different transitions have different completion times. The variation may be large. In the previous example, the time needed to charge $C_L + 4C_I$ is approximately 21 times larger than that needed to charge C_L when $C_I/C_L = 5$ (see the next sections for details). Although this is an extreme case, the message is clear: "Not all transitions are equally fast". This motivated us to first classify the transitions according to their delays. Then to remove some of the "slow" transitions. What we get in this case, is a *constrained channel*! (there is rich literature in information theory community on this topic).

Of course, the question that is raised is what is the advantage of keeping only the "fast" transitions. By throwing out some of the "slow" ones we directly decrease the number of bits we can transmit every time we use the bus. This is true, but at the same time we can run the bus faster. We can reduce the period of its synchronizing clock. Obviously, there are two opposite forces here. It turns out that the *net* result can be very significant. Net result in speed increase by a factor of more than 2 is possible (it will be shown in Chapter 8).

In this Chapter we discuss the details of the estimation of the transition completion times, and the classification of the transitions into their *delay* classes. We also provide a simple motivating coding example to establish the idea of *Coding For Speed* [1]. In Chapter 8, we continue with fundamental bounds and practical coding schemes.

2. DELAY ESTIMATION FOR BUSSES

In this section we present an electrical model for deep sub-micron busses. Based on this model we approximate the delay function loosely introduced above.

2.1 Coupled Bus Lines and Drivers

Figure 3 shows a model for the drivers and busses in sub-micron technologies. The lines are assumed capacitively coupled (distributed R-C, it can be shown that inductance does not influence the *1-st.* order Pade approximation that we use here). c_L is the *parasitic capacitance per unit length between each line and ground*, and c_I is the *interwire capacitance per unit length between adjacent lines*. r is the *distributed series resistance of the lines per unit length*. The drivers are modeled as voltage sources u_i with series resistances r_d^i.

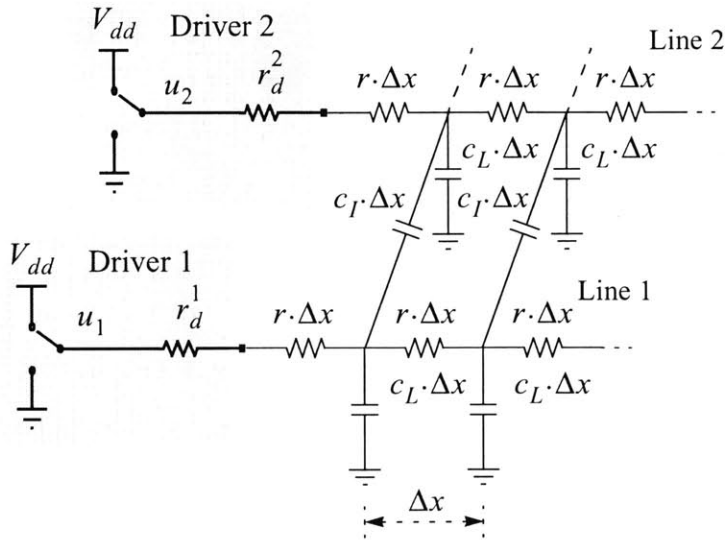

Figure 3: Lines and Drivers

2.2 Elmore Delay of Coupled Lines and Multiple sources

A commonly used approximate measure of the propagation delay of a step excitation through a linear system is the Elmore delay [10]. For a system $H(s)$ driven by a step input $u(t)$ and producing an output $y(t)$ the delay T is formally defined as the solution of the equation,

$$\int_0^\infty (t - T) \cdot \frac{dy}{dt}\, dt = 0 \tag{1}$$

For the definition to be physically meaningful $y(t)$ has to be increasing and its limit for $t \to \infty$ must exist. Elmore delay is used as a delay metric even in cases where the monotonicity condition doesn't hold. Assuming that $y(\infty)$ exists and it is different than $y(0)$, equation (1) implies that,

$$T = \frac{1}{y(\infty) - y(0)} \cdot \int_0^\infty t \cdot \frac{dy}{dt}\, dt \tag{2}$$

or in the Laplace domain,

$$T = \frac{-1}{y(\infty) - y(0)} \cdot \frac{d}{ds}[sY(s)]\Big|_{s=0} \tag{3}$$

Normalizing the supply voltage, $V_{dd} = 1$, equation (3) becomes,

$$T = -[y(\infty) - y(0)] \cdot \frac{d}{ds}[sY(s)]\Big|_{s=0} \tag{4}$$

In the case of the data bus there is more than one source exciting the network simultaneously. Moreover, since the data is random, the pattern of the driving voltages can be arbitrary. This introduces the need for a more generalized definition of the delay. To simplify the situation we assume that the source voltages in Figure 3 are steps of the form,

$$u_k(t) = \begin{cases} u_k^o & \text{for } t < 0 \\ u_k^N & \text{for } t \geq 0 \end{cases} \tag{5}$$

where $u_k^o, u_k^N \in \{0, 1\}$ for all $k = 1, \ldots, m$. Now let $V(x, t) = [V_1(x, t), \ldots, V_m(x, t)]^T$ be the voltages of the lines at time t and distance $x \in [0, L]$ from the drivers. L is the physical length of

the lines. Then for the voltages at the ends of the lines we have that, $V_k(L, 0) = u_k^o$ and in the limit as $t \to \infty$, $V_k(L, \infty) = u_k^N$.

Since the lines of the bus are electrically coupled, the delay at every line is a function of the initial and the final conditions $u^o = [u_1^o, \ldots, u_m^o]^T$ and $u^N = [u_1^N, \ldots, u_m^N]^T$ respectively. We define the delay function of the *k-th* line as:

$$T_k : \{0, 1\}^m \times \{0, 1\}^m \to [0, \infty) \tag{6}$$

such that

$$T_k(u^o, u^N) = -(u_k^N - u_k^o) \cdot \frac{d}{ds}[sV_k(L, s)]\Big|_{s=0} \tag{7}$$

2.3 Calculation of the Delay Functions

As mentioned before, the lines of the bus are assumed distributed with uniformly distributed parasitic series resistance per unit length r, capacitance to ground per unit length c_L and interwire parasitic capacitance per unit length c_I.

Now let $I(x, t) = [I_1(x, t), \ldots, I_m(x, t)]^T$ be the currents running through the lines at time t and at distance $x \in [0, L]$ from the drivers. The equivalent network of the bus in Figure 3 satisfies the following system of partial differential equations:

$$\left.\begin{aligned} \frac{\partial}{\partial x}V(x, t) + R \cdot I(x, t) &= 0 \\ \frac{\partial}{\partial x}I(x, t) + C \cdot \frac{\partial}{\partial t}V(x, t) &= 0 \end{aligned}\right\} \tag{8}$$

for all $t \geq 0$ and $x \in [0, L]$. The capacitance matrix C corresponding to the capacitive coupling in the network (Figure 3) is:

$$C = \begin{bmatrix} 1+\lambda & -\lambda & 0 & \dots & 0 \\ -\lambda & 1+2\lambda & -\lambda & : & 0 \\ 0 & -\lambda & & : & : \\ : & : & : & 1+2\lambda & -\lambda \\ 0 & 0 & \dots & -\lambda & 1+\lambda \end{bmatrix} \cdot c_L \tag{9}$$

where $\lambda = c_I / c_L$. The $m \times m$ resistance matrix R is $R = diag(r, r, \ldots, r)$. The network of the lines satisfies the *Initial Condition*:

$$V(x, 0^+) = u^o \tag{10}$$

for all $x \in (0, L)$, with $u^o = [u_1^o, \ldots, u_m^o]^T$. It also satisfies the *Boundary Conditions*:

$$V(0, t) + R_d \cdot I(0, t) = u^N \tag{11}$$

and

$$I(L, t) = 0 \tag{12}$$

for all $t > 0$. The $m \times m$ resistance matrix R_d is $R_d = diag(r_d^1, r_d^2, \ldots, r_d^m)$, where r_d^i is the output resistance of the *i-th* driver.

Now let $\tilde{V}(x, s)$ and $\tilde{I}(x, s)$ be the Laplace transforms of $V(x, t)$ and $I(x, t)$ with respect to the time variable. Then, from (8) we have,

$$\frac{\partial}{\partial x}\begin{bmatrix} \tilde{V} \\ \tilde{I} \end{bmatrix} = \begin{bmatrix} 0 & -R \\ -Cs & 0 \end{bmatrix} \cdot \begin{bmatrix} \tilde{V} \\ \tilde{I} \end{bmatrix} + \begin{bmatrix} 0 \\ C \cdot V(x, 0^+) \end{bmatrix} \tag{13}$$

We set

$$H(s) = \begin{bmatrix} 0 & -R \\ -Cs & 0 \end{bmatrix} \tag{14}$$

and use (13), (10) and (14) to get,

$$\frac{\partial}{\partial x}\begin{bmatrix} \tilde{V} \\ \tilde{I} \end{bmatrix} = H(s) \cdot \begin{bmatrix} \tilde{V} \\ \tilde{I} \end{bmatrix} + \begin{bmatrix} 0 \\ C \cdot u^o \end{bmatrix} \tag{15}$$

In the Laplace domain the boundary conditions (11) and (12) become:

$$\tilde{V}(0,s) + R_d \cdot \tilde{I}(0,s) = \frac{1}{s} \cdot u^N \tag{16}$$

and

$$\tilde{I}(L,s) = 0 \tag{17}$$

Let $e^{H(s)\cdot L}$ be the $2m \times 2m$ matrix exponential associated with equation (15). From the definition of the exponential we have,

$$\begin{aligned} e^{H(s)\cdot L} &= \sum_{k=0}^{\infty} \frac{L^k}{k!} \cdot H^k(s) \\ &= \sum_{k=0}^{\infty} \frac{L^{2k}}{(2k)!} \cdot \begin{bmatrix} RC & 0 \\ 0 & CR \end{bmatrix}^k \cdot s^k + \\ &+ \sum_{k=0}^{\infty} \frac{L^{2k+1}}{(2k+1)!} \cdot \begin{bmatrix} 0 & -R \\ -Cs & 0 \end{bmatrix} \cdot \begin{bmatrix} RC & 0 \\ 0 & CR \end{bmatrix}^k \cdot s^k \end{aligned} \tag{18}$$

We can decompose the exponential into $m \times m$ blocks as,

$$e^{H(s)\cdot L} = \begin{bmatrix} \alpha(s) & \beta(s) \\ \gamma(s) & \delta(s) \end{bmatrix} \tag{19}$$

where α, β, γ, δ are analytic matrix functions of the form,

$$\left.\begin{aligned} \alpha(s) &= I + \frac{L^2}{2} RCs + O(s^2) \\ \beta(s) &= -LR - \frac{L^3}{3!} RCRs + O(s^2) \\ \gamma(s) &= -LCs + O(s^2) \\ \delta(s) &= I + \frac{L^2}{2} CRs + O(s^2) \end{aligned}\right\} \tag{20}$$

Equations (15) form a system of first order linear differential equations with constant coefficients, so their solution ([2]) evaluated at $x = L$ is given by,

$$\begin{bmatrix}\tilde{V}(L,s)\\ \tilde{I}(L,s)\end{bmatrix} = e^{H(s)\cdot L}\left\{\begin{bmatrix}\tilde{V}(0,s)\\ \tilde{I}(0,s)\end{bmatrix} + H(s)^{-1}\begin{bmatrix}0\\ C\cdot u^{o}\end{bmatrix}\right\} - H(s)^{-1}\begin{bmatrix}0\\ C\cdot u^{o}\end{bmatrix} \tag{21}$$

Note that for $s \neq 0$ the matrix $H(s)$ is invertible and we have that,

$$(H(s))^{-1} = \begin{bmatrix}0 & -s^{-1}\cdot C^{-1}\\ -R^{-1} & 0\end{bmatrix} \tag{22}$$

Now using (14) and (19), equation (21) simplifies to,

$$\begin{bmatrix}\tilde{V}(L,s)\\ \tilde{I}(L,s)\end{bmatrix} = \begin{bmatrix}\alpha & \beta\\ \gamma & \delta\end{bmatrix}\cdot\left\{\begin{bmatrix}\tilde{V}(0,s)\\ \tilde{I}(0,s)\end{bmatrix} - \begin{bmatrix}\frac{1}{s}u^{o}\\ 0\end{bmatrix}\right\} + \begin{bmatrix}\frac{1}{s}u^{o}\\ 0\end{bmatrix} \tag{23}$$

From (16), (17) and (23) we have equation (24)

$$\begin{bmatrix}I & R_d\\ \gamma(s) & \delta(s)\end{bmatrix}\cdot\begin{bmatrix}\tilde{V}(0,s)\\ \tilde{I}(0,s)\end{bmatrix} = \frac{1}{s}\begin{bmatrix}u^{N}\\ \gamma(s)\cdot u^{o}\end{bmatrix} \tag{24}$$

which implies that,

$$\begin{bmatrix}\tilde{V}(0,s)\\ \tilde{I}(0,s)\end{bmatrix} = \frac{1}{s}\cdot\begin{bmatrix}u^{N}\\ 0\end{bmatrix} + \frac{1}{s}\cdot\begin{bmatrix}R_d(\delta-\gamma R_d)^{-1}\gamma\\ -(\delta-\gamma R_d)^{-1}\gamma\end{bmatrix}\cdot(u^{N}-u^{o}) \tag{25}$$

Replacing (25) into (23) we get,

$$\tilde{V}(L,s) = \frac{1}{s}\{u^{o} + [\alpha + (\alpha R_d - \beta)(\delta-\gamma R_d)^{-1}\gamma](u^{N}-u^{o})\} \tag{26}$$

which through (20) implies,

$$\frac{d}{ds}[s\tilde{V}(L,s)]\Big|_{s=0} = -\left(R_d + \frac{L}{2}\cdot R\right)LC(u^{N}-u^{o}) \tag{27}$$

We define the total resistance and capacitance *matrices* of the circuit as,

$$R_T \equiv R_d + \frac{L}{2} \cdot R$$
$$C_T \equiv L \cdot C \tag{28}$$

Then from (7), (27) and (28), it is,

$$T_k(u^o, u^N) = (u_k^N - u_k^o) \cdot e_k^T \cdot R_T \cdot C_T \cdot (u^N - u^o) \tag{29}$$

where, e_k^T is the row vector with *1* in the *i-th* coordinate and 0 everywhere else. Equation (29) can also be written in the vector form,

$$T(u^o, u^N) = diag(u^N - u^o) \cdot R_T \cdot C_T \cdot (u^N - u^o) \tag{30}$$

If we make the assumption that the output resistances of all the drivers are the same and independent of their logical outputs, i.e. $r_d^i = r_d$ for every $i = 1, \ldots, m$, then (30) is simplified to,

$$T(u^o, u^N) = diag(u^N - u^o) \cdot C_T \cdot (u^N - u^o) \cdot r_T \tag{31}$$

where $r_T = r_d + \frac{L}{2} \cdot r$. Finally, note that in this case, the sum of the delays in the bus is given by the quadratic form:

$$\sum_{k=1}^{m} T_k(u^o, u^N) = (u^N - u^o)^T \cdot C_T \cdot (u^N - u^o) \cdot r_T. \tag{32}$$

Remark:

Recall that the function (29), of the delay of the *k-th* line, is the result of a *1-st.* order Pade approximation (Elmore delay measure). It is straight forward to use Equation (26) and get higher order and more accurate approximations of the delay. It can be seen that a second order approximation of the delay would involve the transitions of four (and not two) near by lines. For the purpose of introducing the notion of coding for speed, the Elmore approximation is sufficient and provides us with a relatively accurate classification of the transitions. If we want to exploit all the power of coding for speed, the best would be to measure the performance of the bus we are going to encode.

3. A RELATION BETWEEN DELAY AND ENERGY

From Equation (32) of Chapter 2 we know that the energy dissipated into heat during the transition $u^o \rightarrow u^N$ is:

$$E_R(u^o, u^N) = \frac{1}{2} \cdot (u^N - u^o)^T \cdot C_T \cdot (u^N - u^o) \tag{33}$$

Comparing (32) with (33), we get the following relation between the dissipated energy and the sum of the delays,

$$\sum_{k=1}^{m} T_k(u^o, u^N) = 2 \cdot r_T \cdot E_R(u^o, u^N) \tag{34}$$

Equation (34) can be written as,

$$\frac{1}{m} \cdot \sum_{k=1}^{m} T_k(u^o, u^N) = 2 \cdot r_T \cdot \frac{E_R(u^o, u^N)}{m} \tag{35}$$

and so,

Average Delay per Line =
$2 \cdot r_T \cdot$ Average Dissipated Energy per Line

4. PROPERTIES OF THE DELAY FUNCTIONS

Following the assumption about the resistances of the drivers, equation (31) can be written more explicitly as,

$$\frac{T_k(u^o, u^N)}{r_T \cdot C_L} = \begin{cases} (1+\lambda)\Delta_1^2 - \lambda\Delta_1\Delta_2 & , \quad k = 1 \\ (1+2\lambda)\Delta_k^2 - \lambda\Delta_k(\Delta_{k-1} + \Delta_{k+1}) & , \quad 1 < k < m \\ (1+\lambda)\Delta_m^2 - \lambda\Delta_m\Delta_{m-1} & , \quad k = m \end{cases} \tag{36}$$

where Δ_k is the change of the binary voltage of the *k-th* line, i.e.:

$$\Delta_k = u_k^N - u_k^o \tag{37}$$

and $C_L = L \cdot c_L$ is the total capacitance between a line and the ground. Since $u_k^N, u_k^o \in \{0, 1\}$, it is of course $\Delta_k \in \{-1, 0, 1\}$.

Table 1 presents the delay of the *k-th* line as a function of Δ_{k-1}, Δ_k and Δ_{k+1}, where $k = 2, 3, \ldots, m-1$. We use the upward arrow $\uparrow$ to denote that $\Delta_i = 1$, the downward arrow $\downarrow$ to denote that $\Delta_i = -1$, and "-" when $\Delta_i = 0$. The possible normalized delay values of an intermediate line are, 0, 1, $1+\lambda$, $1+2\lambda$, $1+3\lambda$ and $1+4\lambda$. Each of these values corresponds to a set of transitions $[u_{k-1}^o, u_k^o, u_{k+1}^o] \rightarrow [u_{k-1}^N, u_k^N, u_{k+1}^N]$.

For the boundary lines 1 and m that have only one neighboring line we have Table 2. (lines 1,2 can be replaced by m, m-1 respectively). Note that Table 2 does not hold when the boundary lines have additional coupling to ground because of fringing effects or shielding of the bus. Here the possible values of the normalized delays are, 0, 1, $1+\lambda$, and $1+2\lambda$. Each of these values corresponds to a set of transitions $[u_1^o, u_2^o] \rightarrow [u_1^N, u_2^N]$, (or $[u_m^o, u_{m-1}^o] \rightarrow [u_m^N, u_{m-1}^N]$).

$\frac{T_k(u^o, u^N)}{r_T \cdot C_L}$			
Lines & Transitions			Delay of line k
$k-1$	k	$k+1$	
Δ_{k-1}	Δ_k	Δ_{k+1}	$\frac{T_k}{r_T C_L}$
-	-	-	0
-	-	↑	0
-	-	↓	0
↑	-	-	0
↓	-	-	0
↓	-	↑	0
↑	-	↓	0
↑	-	↑	0
↓	-	↓	0
↑	↑	↑	1
↓	↓	↓	1
-	↑	↑	$1+\lambda$
-	↓	↓	$1+\lambda$
↑	↑	-	$1+\lambda$
↓	↓	-	$1+\lambda$
-	↑	-	$1+2\lambda$
-	↓	-	$1+2\lambda$
↑	↑	↓	$1+2\lambda$
↓	↑	↑	$1+2\lambda$
↑	↓	↓	$1+2\lambda$
↓	↓	↑	$1+2\lambda$
-	↑	↓	$1+3\lambda$
-	↓	↑	$1+3\lambda$
↑	↓	-	$1+3\lambda$
↓	↑	-	$1+3\lambda$
↑	↓	↑	$\mathbf{1+4\lambda}$
↓	↑	↓	$\mathbf{1+4\lambda}$

Table 1: The Delay Function of Intermediate Lines

$\frac{T_1(u^o, u^N)}{r_T \cdot C_L}$		
Lines		Delay of line 1
1	2	
Δ_1	Δ_2	$\frac{T_1}{r_T C_L}$
-	-	0
-	↑	0
-	↓	0
↑	↑	1
↓	↓	1
↑	-	$1+\lambda$
↓	-	$1+\lambda$
↑	↓	$1+2\lambda$
↓	↑	$1+2\lambda$

Table 2: The Delay Function of the Boundary Lines

We define the *delay function* T_d on the set of the transitions of the bus as follows:

$$T_d(u^o, u^N) = \max_{k=1,2,\ldots,m} T_k(u^o, u^N) \tag{38}$$

We also define the partition of the transitions set, D_{00}, D_0, D_1, D_2, D_3 and D_4, called the *delay classes*, to be the fibres of the delay function:

$$\begin{aligned} D_{00} &= \{all\ (u^o, u^N)\ such\ that\ T_d(u^o, u^N) = 0\} \\ D_r &= \{all\ (u^o, u^N)\ such\ that\ T_d(u^o, u^N) = 1 + r\lambda\},\ r = 0, 1, 2, 3, 4 \end{aligned} \tag{39}$$

For example, if the bus has only 3 lines, i.e. $u^o, u^N \in \{0, 1\}^3$ we have the delay pattern of Table 3.

Delay Class of $u^o \rightarrow u^N$		u^N							
		000	001	010	011	100	101	110	111
u^o	000	D_{00}	D_1	D_2	D_1	D_1	D_1	D_1	D_0
	001	D_1	D_{00}	D_3	D_2	D_1	D_1	D_2	D_1
	010	D_2	D_3	D_{00}	D_1	D_3	$\boldsymbol{D_4}$	D_1	D_1
	011	D_1	D_2	D_1	D_{00}	D_2	D_3	D_1	D_1
	100	D_1	D_1	D_3	D_2	D_{00}	D_1	D_2	D_1
	101	D_1	D_1	$\boldsymbol{D_4}$	D_3	D_1	D_{00}	D_3	D_2
	110	D_1	D_2	D_1	D_1	D_2	D_3	D_{00}	D_1
	111	D_0	D_1	D_1	D_1	D_1	D_2	D_1	D_{00}

Table 3: Delay Classes in $\{0, 1\}^3$

5. A SAFE DELAY ESTIMATE

There is an issue with Elmore approximation that needs to be discussed. In Table 1 we see that whenever a line does not transition, its delay is zero. This might not always be true in reality. Consider a bus with *16* lines and suppose that λ is so large that the coupling to ground is relatively negligible. Also, suppose that all lines but line *8* change from *0* to *1*, while line *8* maintains a logical value *0*. It is clear that because of the coupling, the voltage of line *8* will exceed 50% V_{dd} if the drivers have a very steep input. So, the delay of line *8* cannot be assumed zero. It will take some time until its voltage drops below 50% V_{dd}. The situation where lines *8* and *9* maintain a logical *0* is similar. A safe approach to the above issue is to modify expression (36) as:

$$\frac{T_k(u^o, u^N)}{r_T \cdot C_L} \leq (|\Delta_k| + \lambda|2\Delta_k - \Delta_{k-1} - \Delta_{k+1}|) \tag{40}$$

with $\Delta_0 = \Delta_{m+1} = 0$. The modified delays are shown in Table 4.

$\lvert\Delta_k\rvert + \lambda\lvert 2\Delta_k - \Delta_{k-1} - \Delta_{k+1}\rvert$			
Lines & Transitions			Normalized, *Safe* Upper Bound of the Delay of line k
$k-1$	k	$k+1$	
Δ_{k-1}	Δ_k	Δ_{k+1}	
-	-	-	0
-	-	↑	λ
-	-	↓	λ
↑	-	-	λ
↓	-	-	λ
↓	-	↑	0
↑	-	↓	0
↑	-	↑	**2λ**
↓	-	↓	**2λ**
↑	↑	↑	1
↓	↓	↓	1
-	↑	↑	$1+\lambda$
-	↓	↓	$1+\lambda$
↑	↑	-	$1+\lambda$
↓	↓	-	$1+\lambda$
-	↑	-	$1+2\lambda$
-	↓	-	$1+2\lambda$
↑	↑	↓	$1+2\lambda$
↓	↑	↑	$1+2\lambda$
↑	↓	↓	$1+2\lambda$
↓	↓	↑	$1+2\lambda$
-	↑	↓	$1+3\lambda$
-	↓	↑	$1+3\lambda$
↑	↓	-	$1+3\lambda$
↓	↑	-	$1+3\lambda$
↑	↓	↑	**$1+4\lambda$**
↓	↑	↓	**$1+4\lambda$**

Table 4: Normalized, *Safe* delay estimate of line k (Intermediate Lines)

Note that the *safe* upper bound of the delay of individual lines, given by Equation (40) and shown in Table 4, may change the estimated delay of the whole transition only from $1+\lambda$ to 2λ and only when a transition pattern ... $\uparrow\ \uparrow$ - $\uparrow\ \uparrow$ ... or ... $\downarrow\ \downarrow$ - $\downarrow\ \downarrow$ ... appears.

Finally, we define the *safe delay function* T_d^s on the set of the transitions of the bus using expression (40) instead of (36), that is:

$$T_d^s(u^o, u^N) = \max_{k=1,2,\ldots,m} (|\Delta_k| + \lambda|2\Delta_k - \Delta_{k-1} - \Delta_{k+1}|) \cdot r_T \cdot C_L \tag{41}$$

We can use the safe delay function (41) instead of (38) to define the delay classes D_{00}, D_0, D_1, D_2, D_3 and D_4 as:

$$\begin{aligned} D_{00} &= \{all\ (u^o, u^N)\ such\ that\ \ T_d^s(u^o, u^N) = 0\} \\ D_r &= \left\{all\ (u^o, u^N)\ such\ that\ \frac{T_d^s(u^o, u^N)}{r_T \cdot C_L} = 1 + r\lambda\right\},\ r = 0, 1, 2, 3, 4 \end{aligned} \tag{42}$$

6. COMPARISON WITH SPICE AND MATLAB SIMULATIONS

In order to verify the results of Table 1 we used a bus with 8 lines and CMOS inverters for drivers in $.18\mu m$ technology. The signals to the drivers were steps. To compensate for larger buses the boundary lines (1 and 8) had total capacitance to ground equal to $C_L + C_I$. Finally, the parameter $\lambda = C_I/C_L$ was equal to seven. For the simulation we picked a few of the slowest representatives of each class. We also measured the delay of class D_0 separately, which we call d_0^e. In the Figure 4, the stars correspond to the ratios of d^e/d_0^e, where d^e are the delays of the different transitions simulated. The integers 0,1,2,3,4 on the x-axis are the numbers of the classes to which the transitions belong. The point $(0, 1)$ corresponds to class D_0.

In Figure 4 the solid line connects the point $(0, 1)$, (corresponding to D_0) to the point $(4, 22.5)$ corresponding to the worst case delay among all transitions. The important issue here is that ***all points lie below the solid line***. This means that if λ_{exp} is the slope of the solid line, then we have that

$$\tilde{T}_{class\ r} \leq 1 + r \cdot \lambda_{exp}\ , \tag{43}$$

while

$$\tilde{T}_{class\ 4} = 1 + 4 \cdot \lambda_{exp}\ . \tag{44}$$

Where $\tilde{T}_{class\ r}$ is the maximum experimental delay within class r. The importance of this observation will become clear in the next section where we discuss the coding for speed. Briefly, the idea is that ***the higher the ratio***:

$$\tilde{T}_{class\ 4} / \tilde{T}_{class\ r},\ r = 1, 2, 3 \tag{45}$$

is, the better! This ratio will determine the increase in the bus speed.

The value of λ_{exp} is 5.4 which is less than the design parameter $\lambda = 7$ (the dashed line in the graph has slope λ). This is expected because of the non-linear behavior of the real (CMOS) drivers. This difference should not cause a problem since as we saw, we can easily measure λ_{exp} and replace it directly into the affine formula of the delay. Finally, the same relations are valid when the input to the drivers is saturated ramp. Again, λ_{exp} has to be re-estimated.

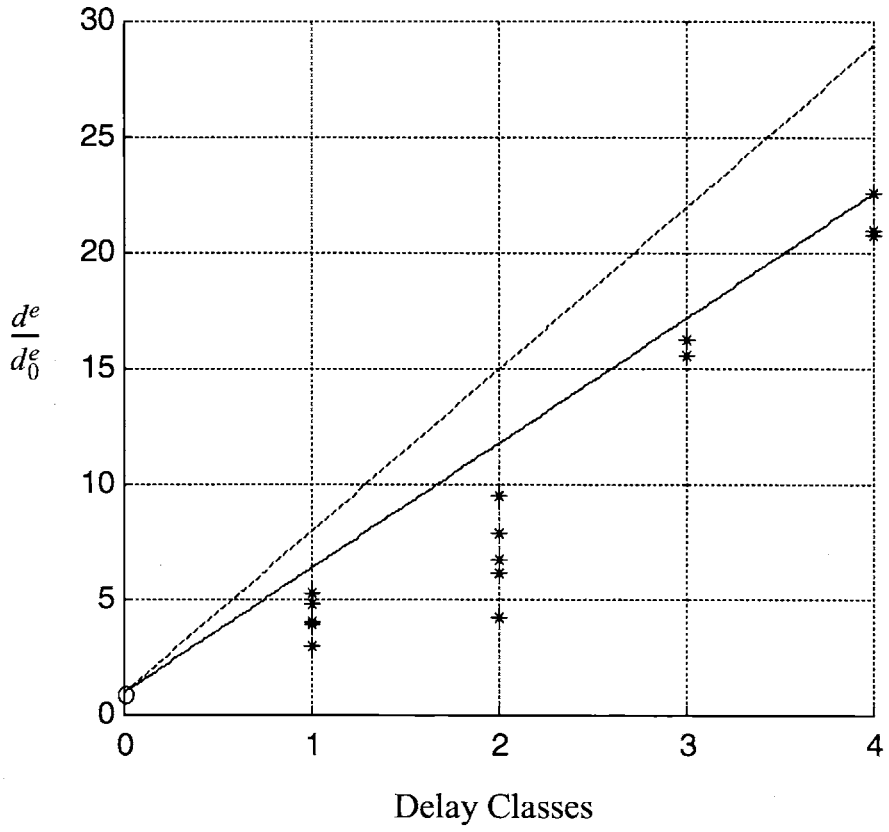

Figure 4: Measurements of the ratios d^e / d_0^e (HSPICE)

Figure 5 is similar to Figure 4, but this time we used MATLAB to simulate the waveforms of all 2^{16} transitions of the 8-line bus. In addition, to classify the transitions of the bus we used the *safe* rule of Table 4. In this case the experimentally measure λ_e was 5.5, instead of 5.4 found using HSPICE.

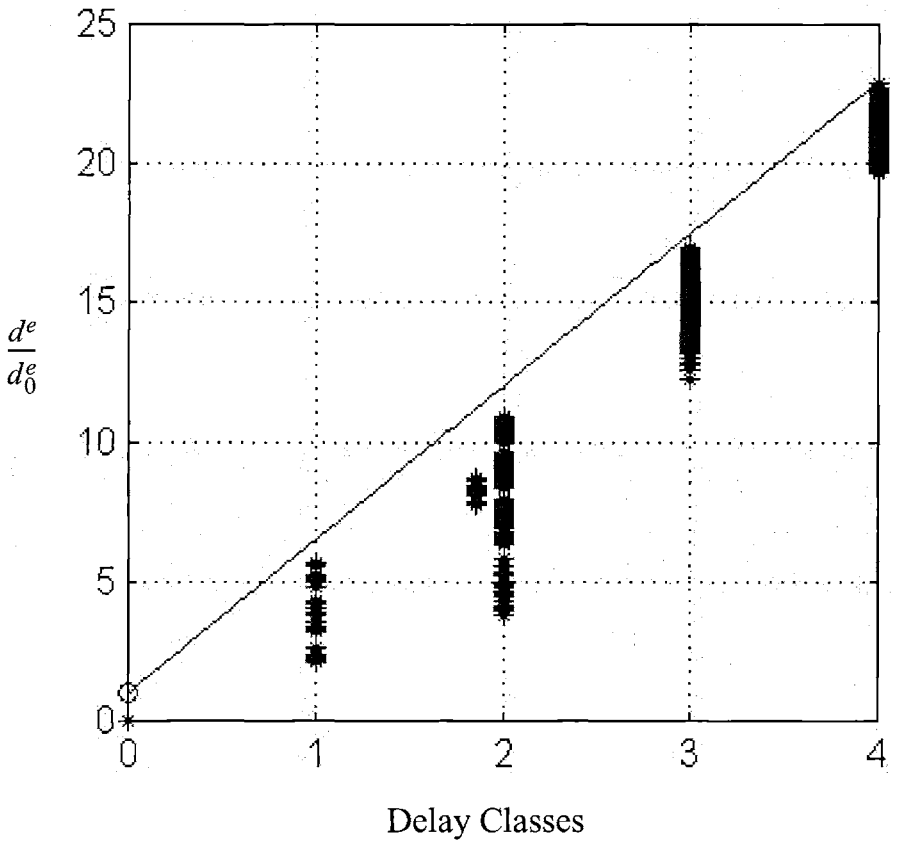

Figure 5: Measurements of the ratios d^e/d_0^e (Matlab)

7. CODING FOR SPEED

In the traditional operation of data busses, the clock period T_c is sufficiently large so that all the transitions in the bus have enough time to be completed. In other words it must be that,

$$T_c \geq \eta \cdot (1 + 4\lambda) \tag{46}$$

where η is a technology parameter. The analysis above suggests that we could use a smaller T_c, i.e. speed up the bus, if we could avoid time-expensive transitions. For example if only transitions of the classes D_{00}, D_0, D_1, D_2 were allowed, then inequality (46) could be replaced by inequality (47).

$$T_c \geq \eta \cdot (1 + 2\lambda) \tag{47}$$

This means that for large values of λ the speed of the bus can almost double. Of course, by not allowing some transitions we automatically reduce the rate of information through the bus.

Here is a concrete example. Suppose for simplicity that the bus has $m = 4$ lines and let $TR2$ be the set of all transitions that have (normalized) delay 0, 1, $1 + \lambda$ or $1 + 2\lambda$. By definition, it is $TR2 = D_{00} \cup D_0 \cup D_1 \cup D_2$. $TR2$ is shown in Table 5 where the dots stand for the allowed transitions and the **x**'s for the forbidden ones.

Although $TR2$ does not have any regular pattern, all the possible transitions are allowed among the states 0, 1, 3, 6, 7, 8, 9, 12, 14 and 15. If the set of states is reduced to $\{0, 1, 3, 6, 7, 8, 9, 12, 14, 15\}$ then the worst case delay is only $1 + 2\lambda$. In this case, the number of bits that can be transmitted each time is decreased from four to $log_2(10)$ which is about 3.3 bits. Also the speed has been increased by a factor of $\frac{1 + 4\lambda}{1 + 2\lambda}$. This ratio is about 1.85 for $\lambda = 3$. On the other hand, the number of bits per transition has been decreased by a factor of $\frac{4}{3.3} \approx 1.21$. So the net result is about $1.85/1.21 = 1.53$ times the initial data rate. The encoder and decoder needed for this example are trivial static maps.

Now, if the set of states is further reduced to $\{0, 1, 6, 7, 8, 9, 14, 15\}$, then exactly 3 bits per transition (an integer number of bits) are possible. This makes the encoding - decoding scheme trivial and the net result is about 1.4 times the initial data rate. That is, a four line bus behaves as a 5 (5.6) line bus!

		u^N															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
u^o	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
	1	.	.	X	.	.	.	.	.	.	.	X	.	.	.	.	.
	2	.	X	.	.	X	X	.	.	.	X	.	.	X	X	.	.
	3	.	.	.	.	X	X	.	.	.	.	.	.	.	X	.	.
	4	.	.	X	X	.	.	.	.	X	X	X	X	.	.	.	.
	5	.	.	X	X	.	.	X	.	X	X	X	X	.	.	X	.
	6	.	.	.	.	.	X	.	.	.	.	X	X	.	X	.	.
	7	.	.	.	.	.	.	.	.	.	.	X	X	.	.	.	.
	8	.	.	.	.	X	X	.	.	.	.	.	.	.	.	.	.
	9	.	.	X	.	X	X	.	.	.	.	X	.	.	.	.	.
	A	.	X	.	.	X	X	X	X	.	X	.	.	X	X	.	.
	B	.	.	.	.	X	X	X	X	.	.	.	.	X	X	.	.
	C	.	.	X	.	.	.	.	.	.	.	X	X	.	.	.	.
	D	.	.	X	X	.	.	X	.	.	.	X	X	.	.	X	.
	E	.	.	.	.	.	X	.	.	.	.	.	.	.	X	.	.
	F	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.

Table 5: *TR2* for 4-bit bus

8. CONCLUSIONS

The functions of the delays of the signals in the lines of general busses were estimated. The properties of these functions were studied and the interaction patterns among the lines causing delay were revealed. This allowed a classification of the transitions according to the time they need to get completed. Finally, the idea of coding for speed was introduced.

9. REFERENCES

[1] P. Sotiriadis, A. Chandrakasan, "Reducing Bus Delay in Submicron Technology Using Coding", *IEEE Asia and South Pacific Design Automation Conf., 2001, pp. 109-114.*

[2] G. Birkhoff and G. Rota, *Ordinary Differential Equations*, New York, Wiley 1978.

[3] P. Sotiriadis, A. Wang, A. Chandrakasan, "Transition Pattern Coding: An approach to reduce Energy in Interconnect", *ESSCIRC'2000, Stockholm, Sweden.*

[4] C. Lin-Hendel, "Accurate Interconnect Modeling for High Frequency LSI/VLSI Circuits and Systems", *Computer Design: VLSI in Computers and Processors, 1990. ICCD '90.*

[5] S. McCormick, J. Allen, "Waveform Moment Methods for Improved Interconnection Analysis", *27th ACM/IEEE Design Automation Conference, 1990.*

[6] A. Kahng, K. Masuko, S. Muddu, "Delay Models for MCM Interconnects When Response in Non-monotone", *Multi-Chip Module Conference, 1997. MCMC '97., 1997 IEEE.*

[7] A. Kahng, S. Muddu, "An Analytical Delay Model for RLC Interconnects", *IEEE Trans. on Computer-aided design of integrated circuits and systems, Vol. 16 NO.12 Dec. 1997.*

[8] T. Sakurai, "Closed-Form Expressions for Interconnect Delay, Coupling and Crosstalk in VLSI's", *IEEE Trans. on Electron Devices, Vol.40, No. 1, Jan. 1993.*

[9] T. Sakurai, S. Kobayashi and M. Noda, "Simple Expressions for Interconnection Delay", *IEEE International Symposium on Circuits and Systems, 1991.*

[10]W. Elmore, "The transient Response of Damped Linear Network with Particular Regard to Wide-band Amplifier", *Journal of Applied Physics, Vol. 19, pp. 55-63, 1948*

[11] G. Efthivoulidis, *personal communication.*

[12] M. Ghausi and J. Kelly, *Introduction to Distributed-Parameter Networks*, N.York, Holt, Rinehart and Winston, 1968

Chapter 8

Coding For Speed: The Fundamental Limits and The Differential RLL(1,∞) Scheme

1. INTRODUCTION

In Chapter 7 we studied the timing properties of deep sub-micron buses. We saw that different transitions of the bus require different amount of time to get completed. With a simple example we demonstrated how we can exploit this property to accelerate the transmission of data.

In the present Chapter we expand the idea of coding for speed into both theoretical and practical directions. We observe that by allowing only fast transitions in the bus, we transform it into a *constrained discrete channel* in the information theoretic sense. Shannon was one of the first to discuss the capacity of constrained noiseless channels in his landmark paper [1]. He introduced the method of the transfer matrix and showed that channel's capacity is equal to the logarithm of the spectral radius of the transfer matrix.

Here we use the transfer matrix method to study the information theoretic limits of the possible increase in the data transmission rate of the bus. The analysis is based on the delay estimation method of Chapter 7. We present examples of calculations of the capacity of small buses and employ mathematical tricks to estimate the capacities of larger buses. These techniques are non constructive and they do not provide any specific coding strategy.

Constrained channels are standard canonical models for magnetic and optical recording systems [2]-[5]. There is a rich literature of proposed coding schemes that approach the capacity of these channels, for example [2]-[5] and the well established, *state splitting algorithm* [7]. We used the state splitting algorithm to develop coding schemes for buses. The results were too complex to be practical. This is true more generally, the coding schemes developed for magnetic and optical systems are not good candidates for buses because of the following two reasons. First, the existing theory treats mostly serial binary channels. Moreover, the constraints in magnetic recording are usually of very short length resulting in a relatively small number of states. This is not the case of constrained buses, here, the number of states is generally large. Second and most important is that the coding in buses must take place within a few, if not less than one clock cycle of the microprocessor and the circuit implementing it must have the minimum possible size and consume very limited power. This is definitely not the case for optical and magnetic recording where large specialized circuits do the job. Therefore there is a gap of orders of magnitudes in the available resources.

For practical applications we introduce, what we call the *Differential Run Length Limited* codes. These codes are in some sense two dimensional. Their "spatial" dimension lies on the parallel transmission of bits through the lines of the bus. Run length limited codewords are used in the spatial dimension and simple binary differentiation takes place temporally. The result is a low complexity coding that can significantly increase the transmission speed.

2. BUSES, CONSTRAINTS AND CAPACITY

Lets consider a bus with n lines as that in Chapter 7 and let $S_n = \{0, 1\}^n$ be the set of all n-bit binary vectors. As we saw in Chapter 7, the time needed for a vector $y = (y_1, y_2, \ldots, y_n)$ to propagate through the bus depends on y as well as on the vector $x = (x_1, x_2, \ldots, x_n)$ transmitted just before. The propagation time of y is a function of the transition (x, y) and can be approximated by Equations (38) or (41) of Chapter 7. Furthermore, we saw that the set $S_n \times S_n$ of the 2^{2n} transitions can be partitioned into the six *delay classes*, D_{00}, D_0, D_1, D_2, D_3 and D_4. Where the delay classes are defined according to the expressions (39) or (42) of Chapter 7. Recall also from Chapter 7 that we can find two positive parameters Λ and T_*, using either analytical or experimental methods, for which we have that: the required completion time of the transitions in D_{00} is zero, and the completion time of the transitions in D_r, $r = 0, 1, \ldots, 4$ is always less than or equal to $(1 + r\Lambda) \cdot T_*$. Moreover, the highest completion time among the transitions in D_4 is equal to $(1 + 4\Lambda) \cdot T_*$.

Now lets define the subsets, M_1, M_2 and M_3, of the set of all transitions $S_n \times S_n$ of the bus, as:

$$\begin{aligned} M_1 &= D_{00} \cup D_0 \cup D_1 \\ M_2 &= D_{00} \cup D_0 \cup D_1 \cup D_2 \\ M_3 &= D_{00} \cup D_0 \cup D_1 \cup D_2 \cup D_3 \end{aligned} \tag{1}$$

Clearly, if we constrain the bus, in the sense that every transition must belong to one of the sets, then we can decrease its clock period by the *acceleration* factors:

$$\theta_1 = \frac{1 + 4\Lambda}{1 + \Lambda} \qquad \theta_2 = \frac{1 + 4\Lambda}{1 + 2\Lambda} \qquad \theta_3 = \frac{1 + 4\Lambda}{1 + 3\Lambda} \tag{2}$$

As in the coding example of Chapter 7, the question now is: *How many bits per use of the bus can be transmitted when only transitions of one of the sets M_1, M_2 or M_4 are allowed?*

Obviously, if we can transmit φ_r bits on average, per use of the bus, when using set M_r, then we can, at least in theory, accelerate the data transmission by a factor of $\theta_r\varphi_r/n$. We call the ratio ϕ_r/n, the *utilization* a_r of the bus using the transitions set M_r. The formal definition of the "average" number of bits per use of the bus φ_r is:

$$\varphi_r = \limsup_{k\to\infty} \frac{1}{k} \log_2 \left\{ \begin{array}{c} \text{\# of sequences of lenght } k \\ \text{whose transitions belong to } M_r \end{array} \right\} \tag{3}$$

The quantity φ_r is the combinatorial capacity [1] of the constrained bus viewed as constrained channel. The *utilization* of the bus is:

$$a_r = \frac{1}{n} \cdot \limsup_{k\to\infty} \frac{1}{k} \log_2 \left\{ \begin{array}{c} \text{\# of sequences of lenght } k \\ \text{whose transitions belong to } M_r \end{array} \right\} \tag{4}$$

3. COUNTING SEQUENCES, THE TRANSFER MATRIX

In this section we discuss the calculation of the utilizations a_1, a_2, a_3 of the bus. We use the *transfer matrix* of the constrained bus (channel), a device introduced by Shannon in [1]. For notational convenience we identify the vectors $(0, 0, \ldots, 0), (1, 0, \ldots, 0), (0, 1, \ldots, 0), \ldots, (1, 1, \ldots, 1)$ with the corresponding numbers $0, 1, \ldots, 2^n - 1$.

The *transfer matrix* $\boldsymbol{M}_r$ of the bus, when the transitions are constrained in the set M_r, is the 0-1, $2^n \times 2^n$ matrix with entries:

$$(\boldsymbol{M}_r)_{x,y} = \begin{cases} 1 \text{ if } (x, y) \in M_r \\ 0 \quad \text{otherwise} \end{cases} \tag{5}$$

Note that $\boldsymbol{M}_r$ is the adjacency matrix of the directed graph [8] corresponding to the set of transitions M_r. Therefore, the number of k-long sequences whose transitions belong to M_r is given by the sum of all entries of $(\boldsymbol{M}_r)^k$ (see for example [9]). So we have:

$$a_r = \frac{1}{n} \cdot \underset{k \to \infty}{\text{limsup}} \frac{1}{k} \log_2 \left\{ \underline{1}^T \cdot (\boldsymbol{M}_r)^k \cdot \underline{1} \right\}$$

where $\underline{1} = (1, 1, \ldots, 1)^T$. Since the matrix $\boldsymbol{M}_r$ is non negative, it is very easy to verify [10] that:

$$\underset{k \to \infty}{\text{limsup}} \frac{1}{k} \log_2 \left\{ \underline{1}^T \cdot (\boldsymbol{M}_r)^k \cdot \underline{1} \right\} = \log_2 \{ \rho(\boldsymbol{M}_r) \} \tag{6}$$

where $\rho(\boldsymbol{M}_r)$ is the maximal positive eigenvalue (and so the spectral radius) of matrix $\boldsymbol{M}_r$. This gives the compact expression for the utilizations:

$$a_r = \frac{\log_2 \{ \rho(\boldsymbol{M}_r) \}}{n} \tag{7}$$

Formula (7) is practically useful only for small n. Using MATLAB on a PC it was possible to calculate a_r, $r = 1, 2, 3$ for n up to 12. In Section 4. we discuss how to approximate a_r by deriving bounds of the spectral radius of matrix $\boldsymbol{M}_r$.

3.1 Example 1

Consider a bus as that in Figure 1, where the lines are assumed distributed but depicted as lumped for convenience. Lets assume that $\Lambda = \lambda = 5$.

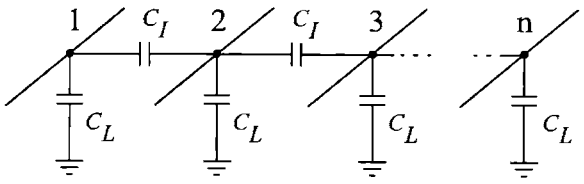

Figure 1: Deep sub-micron bus.

Using the *safe* classification of the transitions, Equations (41) & (42) or Table 4 of Chapter 7, for $n = 4$, we get the transfer matrices (dots stand for ones and holes stand for zeros):

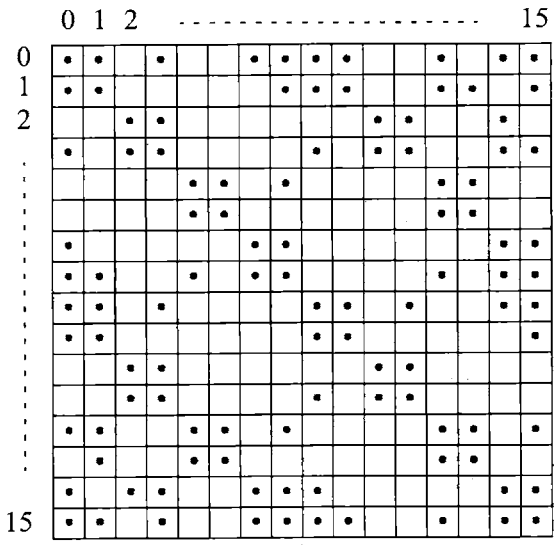

Figure 2: Transfer matrix $\boldsymbol{M}_1$ for $n = 4$ and *safe* classification of the transitions

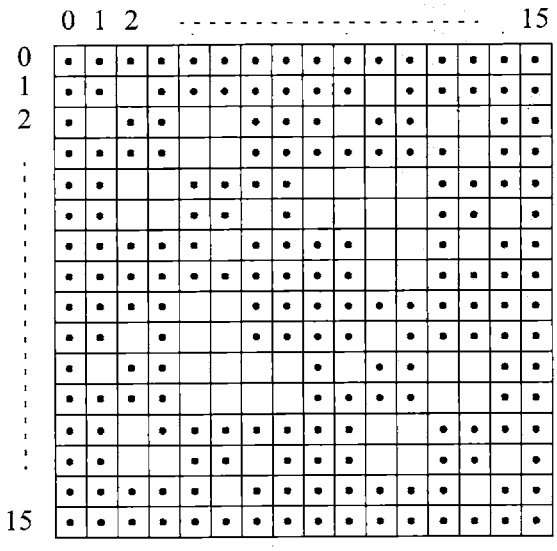

Figure 3: Transfer matrix $\boldsymbol{M}_2$ for $n = 4$ and *safe* classification of the transitions

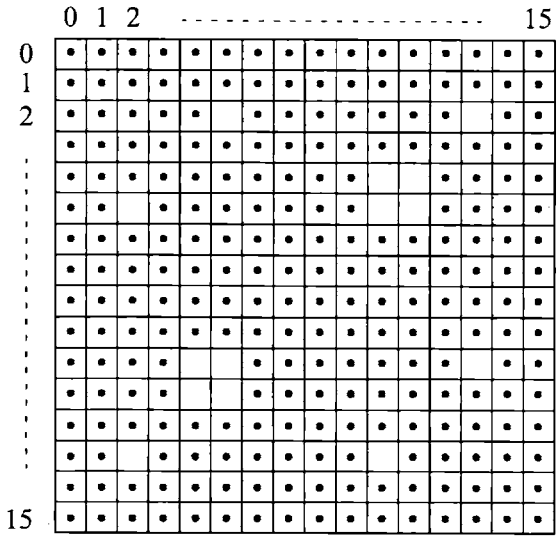

Figure 4: Transfer matrix $\boldsymbol{M}_3$ for $n = 4$ and *safe* classification of the transitions

The spectral radius of the matrices $\boldsymbol{M}_1$, $\boldsymbol{M}_2$ and $\boldsymbol{M}_3$ presented in the figures are: 7.5205, 12.6063 and 15.2062 respectively. Therefore, the corresponding utilizations are:

$$a_1 = 0.7277 \qquad a_2 = 0.9140 \qquad a_3 = 0.9816 \tag{8}$$

For $\Lambda = 5$ the acceleration factors given by Equations (2) are:

$$\theta_1 = 3.5 \qquad \theta_2 = 1.909 \qquad \theta_3 = 1.3125 \tag{9}$$

And so, the theoretical possible increases in the speed of the bus are:

$$\theta_1 a_1 = 2.547 \qquad \theta_2 a_2 = 1.7448 \qquad \theta_3 a_3 = 1.2884 \tag{10}$$

Therefore, at least from a theoretical perspective, increase of the data rate through the bus by a factor of more than 2.5 is possible! Moreover, recall that the classification of the transitions is relatively rough. Looking carefully in Figure 5 of Chapter 7, we see that some of the transitions in D_2 should belong to D_1 allowing for even higher transmission rates.

Now, is it interesting to see how the utilization factors of M_1, M_2 and M_3 change when we introduce boundary coupling to the bus. The situation is shown in Figure 5.

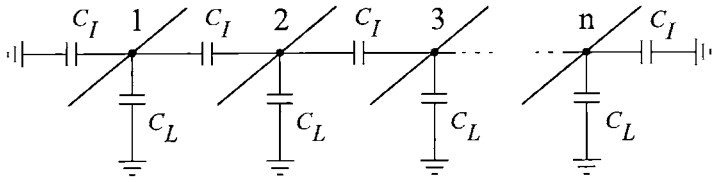

Figure 5: Isolated deep sub-micron bus.

Modern design methodology suggests the insertion of ground or power supply lines in the bus in order to isolate the bus lines from the environment and reduce the injected noise. For the bus of Figure 5 and $n = 4$, the corresponding transfer matrices are shown in the figures below.

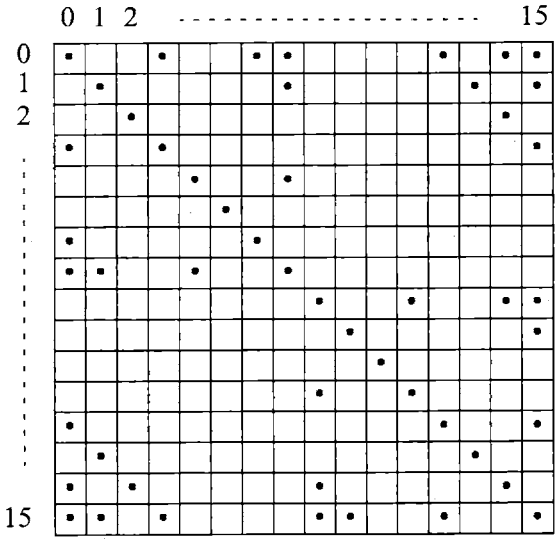

Figure 6: Transfer matrix $\boldsymbol{M}_1$ for $n = 4$, *safe* classification of the transitions and isolated bus

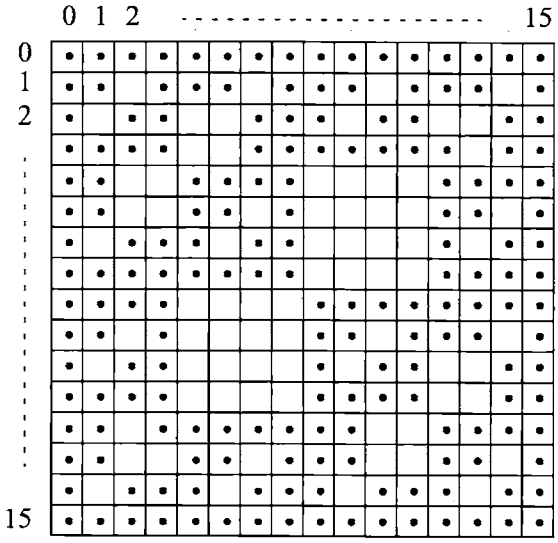

Figure 7: Transfer matrix M_2 for $n = 4$, *safe* classification of the transitions and isolated bus

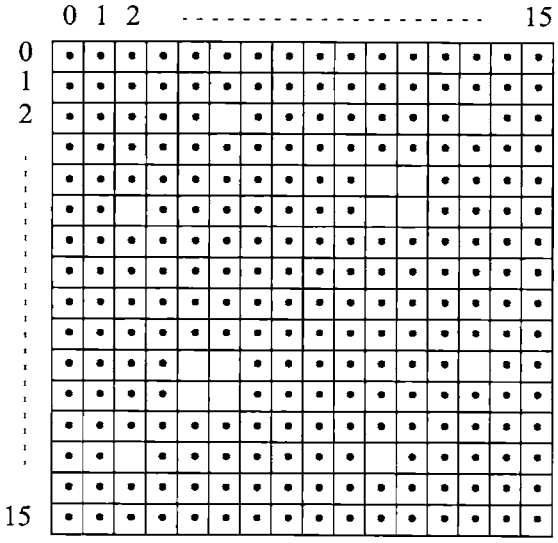

Figure 8: Transfer matrix M_3 for $n = 4$, *safe* classification of the transitions and isolated bus

The spectral radius of the matrices $\boldsymbol{M}_1$, $\boldsymbol{M}_2$ and $\boldsymbol{M}_3$ are: 4.4106, 11.7037 and 15.2062 respectivelly. Therefore the corresponding utilizations are:

$$a_1 = 0.5352 \qquad a_2 = 0.8872 \qquad a_3 = 0.9816$$

The acceleration factors are as in Equation (9). The speed of the data transmission can be increased by the factors:

$$\theta_1 a_1 = 1.8732 \qquad \theta_2 a_2 = 1.6937 \qquad \theta_3 a_3 = 1.2884$$

3.2 Example 2

For an 8-line bus of the Figure 1 and $\Lambda = 5$, the corresponding utilizations are:

$$a_1 = 0.6403 \qquad a_2 = 0.8882 \qquad a_3 = 0.9743$$

The acceleration factors are as in Equation (9). Therefore, in this case, the theoretically possible increases in the speed of the bus are:

$$\theta_1 a_1 = 2.2410 \qquad \theta_2 a_2 = 1.6956 \qquad \theta_3 a_3 = 1.2788$$

If the 8-line bus has boundary lines as in Figure 5, the utilizations and speed increases are changed accordingly to:

$$a_1 = 0.5408 \qquad a_2 = 0.8751 \qquad a_3 = 0.9743$$

and

$$\theta_1 a_1 = 1.8928 \qquad \theta_2 a_2 = 1.6706 \qquad \theta_3 a_3 = 1.2788.$$

Figure 9 presents the transfer matrix $\boldsymbol{M}_2$ for this case. It has an impressive fractal like structure.

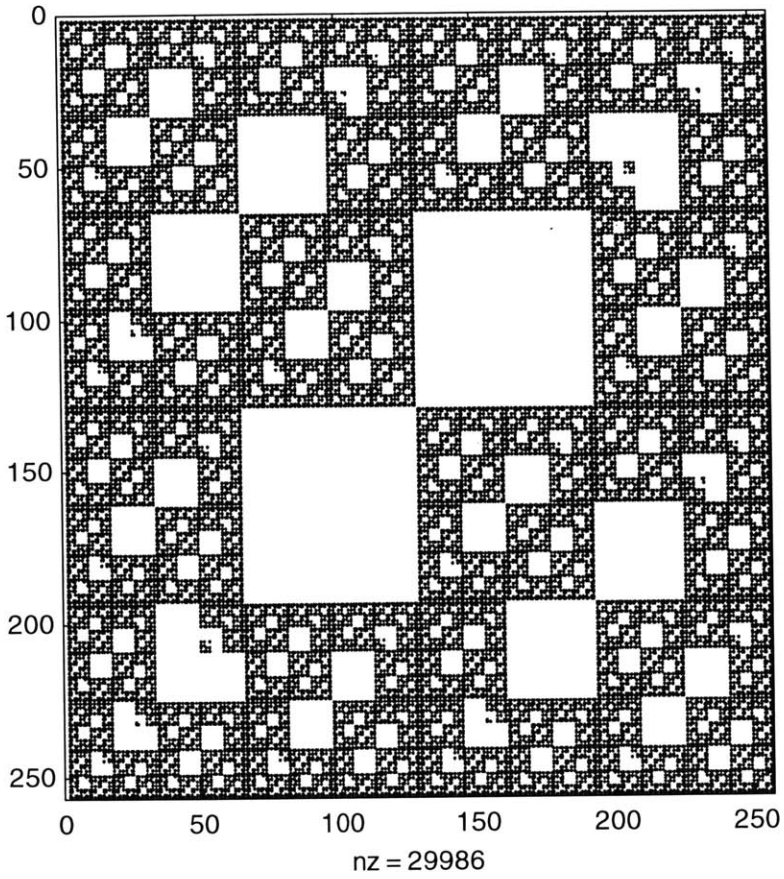

Figure 9: Transfer matrix $\boldsymbol{M}_2$ for $n = 8$, *safe* classification of the transitions and isolated bus

3.3 Utilization factors of small buses

For small buses of sizes $n = 2, 3, \ldots, 8$ the utilization factors can be calculated directly from the spectral radius of the transfer matrices $\boldsymbol{M}_1$, $\boldsymbol{M}_2$ and $\boldsymbol{M}_3$. In Table 1 and Table 2 below we see the values of the utilization factors for non-isolated and isolated buses. The *safe* classification of the transitions was used (Equations (41) & (42) or Table 4 of Chapter 7).

	Size of the bus:						
Utilizations	$n = 2$	$n = 3$	$n = 4$	$n = 5$	$n = 6$	$n = 7$	$n = 8$
a_1	0.9163	0.7820	0.7277	0.6909	0.6684	0.6523	0.6403
a_2	1.0000	0.9300	0.9140	0.9035	0.8965	0.8918	0.8882
a_3	1.0000	0.9861	0.9816	0.9787	0.9768	0.9754	0.9743

Table 1: Utilization factors of buses with $n = 2, 3, \ldots, 8$ lines

	Size of the isolated bus:						
Utilizations	$n = 2$	$n = 3$	$n = 4$	$n = 5$	$n = 6$	$n = 7$	$n = 8$
a_1	0.5000	0.5283	0.5352	0.5323	0.5344	0.5381	0.5408
a_2	0.9163	0.8941	0.8872	0.8821	0.8790	0.8767	0.8751
a_3	1.0000	0.9861	0.9816	0.9787	0.9768	0.9754	0.9743

Table 2: Utilization factors of isolated buses with $n = 2, 3, \ldots, 8$ lines

3.4 Example 3

Assuming $\Lambda = 5$, the acceleration factors, Equation (9), are: $\theta_1 = 3.5$, $\theta_2 = 1.909$, $\theta_3 = 1.3125$

The speed increase for buses with $n = 2, 3, \ldots, 8$ lines is given below using Table 2.

	Size of the isolated bus:						
Speed Incr.	$n = 2$	$n = 3$	$n = 4$	$n = 5$	$n = 6$	$n = 7$	$n = 8$
$\theta_1 a_1$	1.7500	1.8490	1.8732	1.8630	1.8704	1.8834	1.8928
$\theta_2 a_2$	1.7492	1.7068	1.6937	1.6839	1.6780	1.6736	1.6706
$\theta_3 a_3$	1.3120	1.2938	1.2879	1.2841	1.2816	1.2797	1.2783

Table 3: Speed Increase factors of isolated buses with $n = 2, 3, \ldots, 8$ lines

Now we compare the results of coding with another approach that can speed up the communication. We reduce the lines of the bus and split the remaining ones uniformly. For simplicity we assume the width of the wires is the same with the distance between them. The situation is shown in Figure 10 (left).

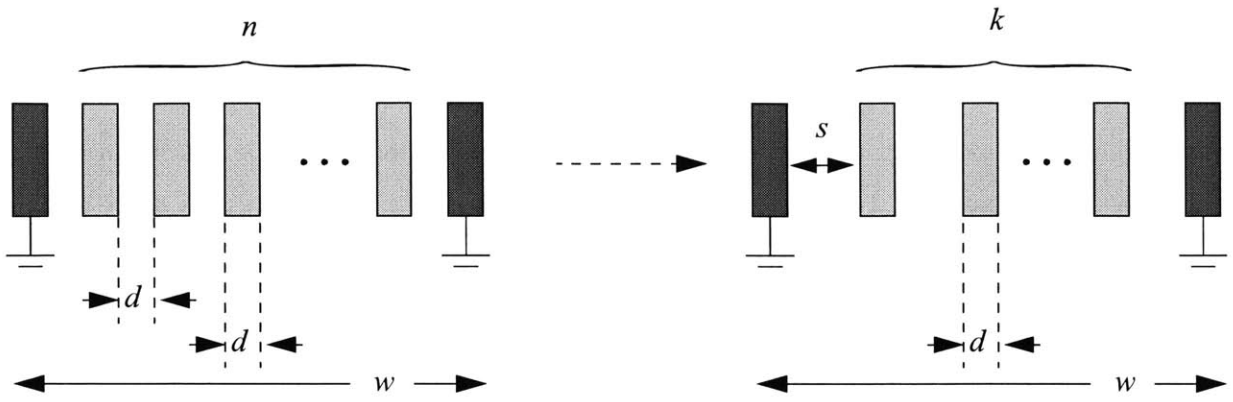

Figure 10: Keeping k out of the n lines and splitting them uniformly

If we keep k out of the n lines while maintaining the same total width w of the bus, then the distance between lines becomes $s = \frac{(2n+1-k)}{k+1} \cdot d$ and the new lambda can be approximated by:

$$\Lambda_k = \Lambda \cdot \frac{d}{s} = \Lambda \cdot \frac{k+1}{2n+1-k} \tag{11}$$

Then the relative speed increase (factor) is given by the approximate expression:

$$S_k(n) = \frac{k}{n} \cdot \frac{1+4\Lambda}{1+4\Lambda_k} \tag{12}$$

For each n, it is $k = 1, 2, \ldots, n-1$ with a particular value of k maximizing S_k. We define,

$$S(n) = \max_k S_k(n) \tag{13}$$

For $\Lambda = 5$, the above formula implies Table 4:

	Size of the isolated bus:						
Speed Incr.	$n = 2$	$n = 3$	$n = 4$	$n = 5$	$n = 6$	$n = 7$	$n = 8$
$S(n)$	1.0000	1.0769	1.0988	1.1455	1.1667	1.1892	1.2080

Table 4: Speed Increase factors of isolated buses with $n = 2, 3, \ldots, 8$ lines using line splitting

Comparing Table 3 with Table 4 we see that coding is a lot more effective. Also note that modern bus designs usually have partitioned structures like that of Figure 11 with n small, for example $n = 2, 3, 4$ (Isolating lines may connect to V_{dd} instead of Ground). For these cases, coding is extremely more effective. Finally, non-uniform distribution of the lines can improve slightly both the encoded and split bus (because of boundary effects).

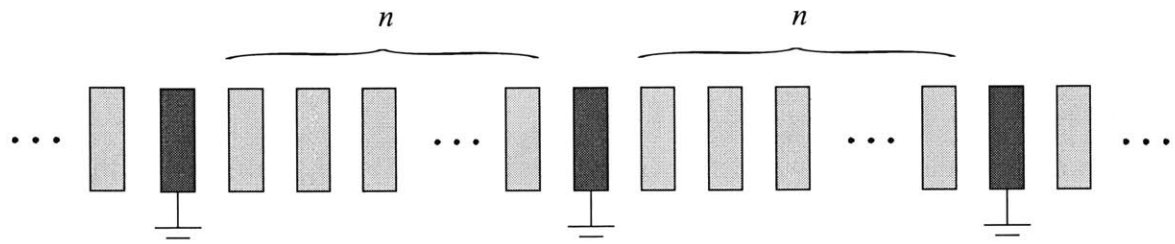

Figure 11: Partitioned Bus

4. UTILIZATION FACTORS OF LARGE BUSES

Calculating the spectral radius of a $2^{64} \times 2^{64}$ matrix, that does not have any convenient structure, is not trivial if not impossible. In order to calculate the utilizations of large buses we need to know the spectral radius of $\boldsymbol{M}_r$. If this is not an option we need to derive some bounds on it. Here, this is done by employing a simple trick: we can establish some rough upper and lower bounds of $\lambda_{max}(\boldsymbol{M}_r)$ if we know how many ones the matrix $\boldsymbol{M}_r$ has. The following two lemmas are in order.

Lemma 1

Let $A = [a_{i,j}]$ *be an* $M \times M$ *symmetric matrix. Then, its maximal eigenvalue* $\lambda_{max}(A)$ *satisfies the inequality:*

$$\lambda_{max}(A) \geq \frac{1}{M} \sum_{i,j=1}^{m} a_{i,j} \tag{14}$$

Proof: For every vector x with $\|x\| = 1$ it is $\lambda_{max}(A) \geq x^T A x$. We can choose x equal to $(1, 1, \ldots, 1)^T/(\sqrt{M})$. □

Lemma 2

Let $A = [a_{i,j}]$ *be an* $M \times M$ *symmetric 0-1 matrix with zeros in its diagonal. If* A *has* N *entries equal to one, then for its maximal eigenvalue* $\lambda_{max}(A)$ *we have:*

$$\lambda_{max}(A) \leq \sqrt{\frac{2N(M-1)}{M}}. \tag{15}$$

Proof: Let $\lambda_2, \lambda_3, \ldots, \lambda_M$ be the rest of the eigenvalues of matrix A. We have the equalities $tr(A) = 0$ and $tr(A^2) = N$ which along with the identities $tr(A) = \lambda_{max} + \lambda_2 + \ldots + \lambda_M$ $tr(A^2) = \lambda_{max}^2 + \lambda_2^2 + \ldots + \lambda_M^2$ give, $\lambda_{max} + \lambda_2 + \ldots + \lambda_M = 0$ and $\lambda_{max}^2 + \lambda_2^2 + \ldots + \lambda_M^2 = N$.

We need to find the maximum λ_{max} under the previous two equality constraints and the inequalities $\lambda_{max} \geq \lambda_2, \lambda_3, \ldots, \lambda_M \geq -\lambda_{max}$. The maximum value λ_{max} can take is the RHS of (15). □

Combining (14) and (15) we have that for every $M \times M$, 0-1 matrix A *with only ones in its diagonal* and a total number of ones, N, its maximal eigenvalue is bounded as:

$$N/M \leq \lambda_{max}(A) \leq 1 + \sqrt{2N} \tag{16}$$

Equation (16) applies directly to the transfer matrices M_1, M_2 and M_3. What we need now is to compute the number of ones they have.

4.1 Counting the Transitions

Lets consider a bus of n lines as that of Figure 1. We want to count how many of the transition pairs (x, y) in $S_n \times S_n$, $S_n = \{0, 1\}^n$ belong to M_1, M_2 and M_3 respectively. Instead of writing $(x, y) \in M_r$ it will be more convenient to think of the transition "vertically" as:

$$\begin{matrix} (y_1, y_2, \ldots, y_n) \\ \uparrow \\ (x_1, x_2, \ldots, x_n) \end{matrix}$$

$$\begin{pmatrix} y_1, y_2, \ldots, y_n \\ x_1, x_2, \ldots, x_n \end{pmatrix} \in M_r \tag{17}$$

Now note that (17) is true if and only if:

$$\begin{pmatrix} y_1, y_2, y_3 \\ x_1, x_2, x_3 \end{pmatrix}, \begin{pmatrix} y_2, y_3, y_4 \\ x_2, x_3, x_4 \end{pmatrix}, \ldots, \begin{pmatrix} y_{n-2}, y_{n-1}, y_n \\ x_{n-2}, x_{n-1}, x_n \end{pmatrix} \in M_r(3) \tag{18}$$

Where we use the notation $M_r(3)$ for the set M_r when $n = 3$ while keeping the symbol M_r for arbitrary size n. The equivalence between (17) and (18) is true because the classification of the transition was based on the delay functions of Equations (36) or (41) of Chapter 7 which "examine" the transition of every line along with the transitions of only its successor and predecessor line. Note that (18) needs to be modified for the case of isolated buses as that of Figure 5.

Now we define the subsets H_1, H_2 and H_3 of $\{0, 1\}^{2\times 2} \times \{0, 1\}^{2\times 2}$ as:

$$H_r = \left\{ \left[\begin{pmatrix} y_1, y_2 \\ x_1, x_2 \end{pmatrix}, \begin{pmatrix} y_2, x_3 \\ x_2, x_3 \end{pmatrix} \right] \text{ if and only if } \begin{pmatrix} y_1, y_2, y_3 \\ x_1, x_2, x_3 \end{pmatrix} \in M_r(3) \right\} \tag{19}$$

Therefore, relation (18), and so relation (17), is true if and only if,

$$\left[\begin{pmatrix} y_j, y_{j+1} \\ x_j, x_{j+1} \end{pmatrix}, \begin{pmatrix} y_{j+1}, x_{j+2} \\ x_{j+1}, x_{j+2} \end{pmatrix} \right] \in H_r \tag{20}$$

for all $j = 1, 2, \ldots, n-2$. We can think of (20) as the outcome of a constrained source [1], [7].

$$\begin{pmatrix} y_1, y_2 \\ x_1, x_2 \end{pmatrix} \rightarrow \begin{pmatrix} y_2, x_3 \\ x_2, x_3 \end{pmatrix} \rightarrow \begin{pmatrix} y_3, x_4 \\ x_3, x_4 \end{pmatrix} \rightarrow \ldots \rightarrow \begin{pmatrix} y_{n-1}, x_n \\ x_{n-1}, x_n \end{pmatrix} \tag{21}$$

Similarly to expression (5) we can define the 16×16 transfer matrices associated to the sets H_r and the propagation (21) as:

$$(\boldsymbol{H}_r)_{Z,\,W} = \begin{cases} 1 \text{ if } [Z, W] \in H_r \\ 0 \quad \text{otherwise} \end{cases} \tag{22}$$

where $Z = \begin{pmatrix} y_1, y_2 \\ x_1, x_2 \end{pmatrix}$ and $W = \begin{pmatrix} y_2, x_3 \\ x_2, x_3 \end{pmatrix}$ belong to $\{0, 1\}^{2\times 2}$. Then, as in Section 3. [9], the number of possible sequences (21) for $n > 2$ is:

$$h_r(n) = \underline{1}^T \cdot (\boldsymbol{H}_r)^{n-2} \cdot \underline{1}. \tag{23}$$

where $\underline{1} = (1, 1, \ldots, 1)^T$ is the all ones vector. With the convention, $Z = x_1 + 2x_2 + 4y_1 + 8y_2$ and $W = x_2 + 2x_3 + 4y_2 + 8y_3$, we see the matrices $\boldsymbol{H}_1$, $\boldsymbol{H}_2$ and $\boldsymbol{H}_3$ in Figure 12, Figure 13 and Figure 14 respectively. The dots stand for ones and the holes stand for zeros. For all three of them we used the *safe* classification of the transitions (Equations (41) & (42) or Table 4 of Chapter 7).

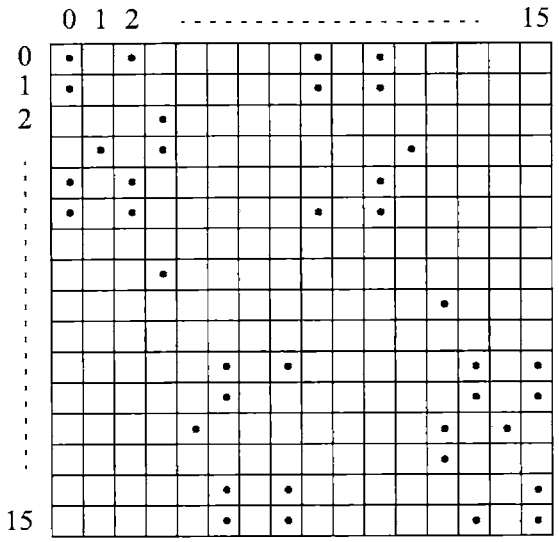

Figure 12: Transfer matrix $\boldsymbol{H}_1$ based on *safe* classification of the transitions

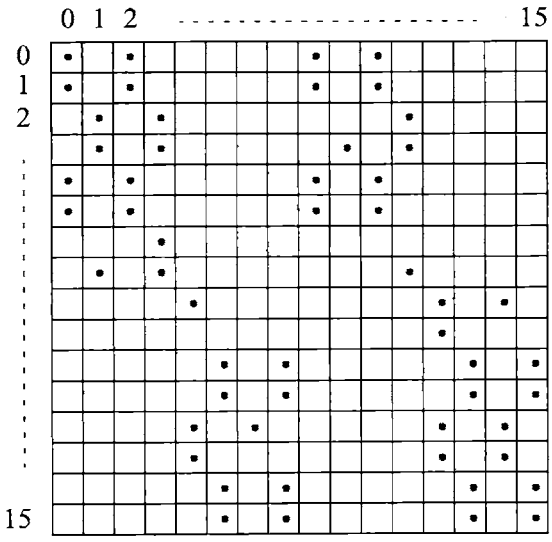

Figure 13: Transfer matrix $\boldsymbol{H}_2$ based on *safe* classification of the transitions

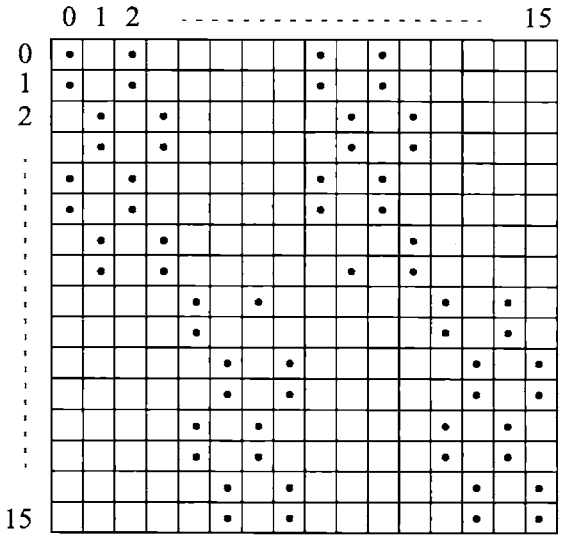

Figure 14: Transfer matrix $\boldsymbol{H}_3$ based on *safe* classification of the transitions

Equations (23) give:

$$\begin{aligned} h_1(n) &= 13.991 \cdot 2.7321^{n-2} + e_1(n) \\ h_2(n) &= 14.975 \cdot 3.594^{n-2} + 0.472 \cdot 1.104^{n-2} + e_2(n) \\ h_3(n) &= 15.907 \cdot 3.901^{n-2} + e_3(n) \end{aligned} \tag{24}$$

with all errors $|e_r(n)| < 1$. In Equation (16) we replace N by $h_r(n)$ and M by 2^n. This results to the bounds:

$$h_r(n)/2^n \le \lambda_{max}(\boldsymbol{M}_r) \le 1 + \sqrt{2h_r(n)} \tag{25}$$

By replacing (25) into the definition of utilization, Equation (7) where $\lambda_{max}(\boldsymbol{M}_r) = \rho(\boldsymbol{M}_r)$, we finally get:

$$\frac{1}{n}\log_2\{h_r(n)\} - 1 \le a_r \le \frac{1}{n}\log_2\{1 + \sqrt{2h_r(n)}\} \tag{26}$$

Asymptotically, as n becomes large, we have:

$$\begin{aligned} 0.450 \le a_1 \le 0.725 \\ 0.846 \le a_2 \le 0.923 \\ 0.964 \le a_3 \le 0.982 \end{aligned} \tag{27}$$

The graphs of the bounds in (26) are shown below in Figure 15.

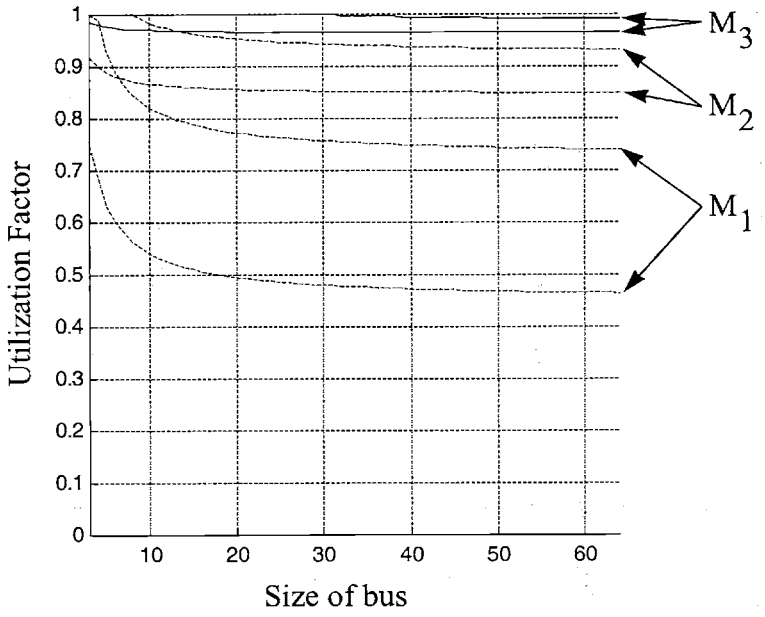

Figure 15: Bounds on the utilization factors of M_1, M_2 and M_3 based on *safe* classification of the transitions and non-isolated buses.

5. DIFFERENTIAL RLL(1,∞) CODING FOR SPEED

The theory developed in the previous sections provided important estimates and bounds on the theoretical limits of speed increase of the bus but it didn't give directions for deriving practical coding schemes. Here we introduce a class of schemes, which we named *differential RLL(1,∞)*, schemes, that have low complexity and provide significant increase in the transmission rate of the data. Their main advantage is that the encoder and decoder are *essentially* combinatorial circuits. Moreover, it can be shown that they can reduce the power consumption of the bus as well. The structure of the schemes is shown in Figure 16.

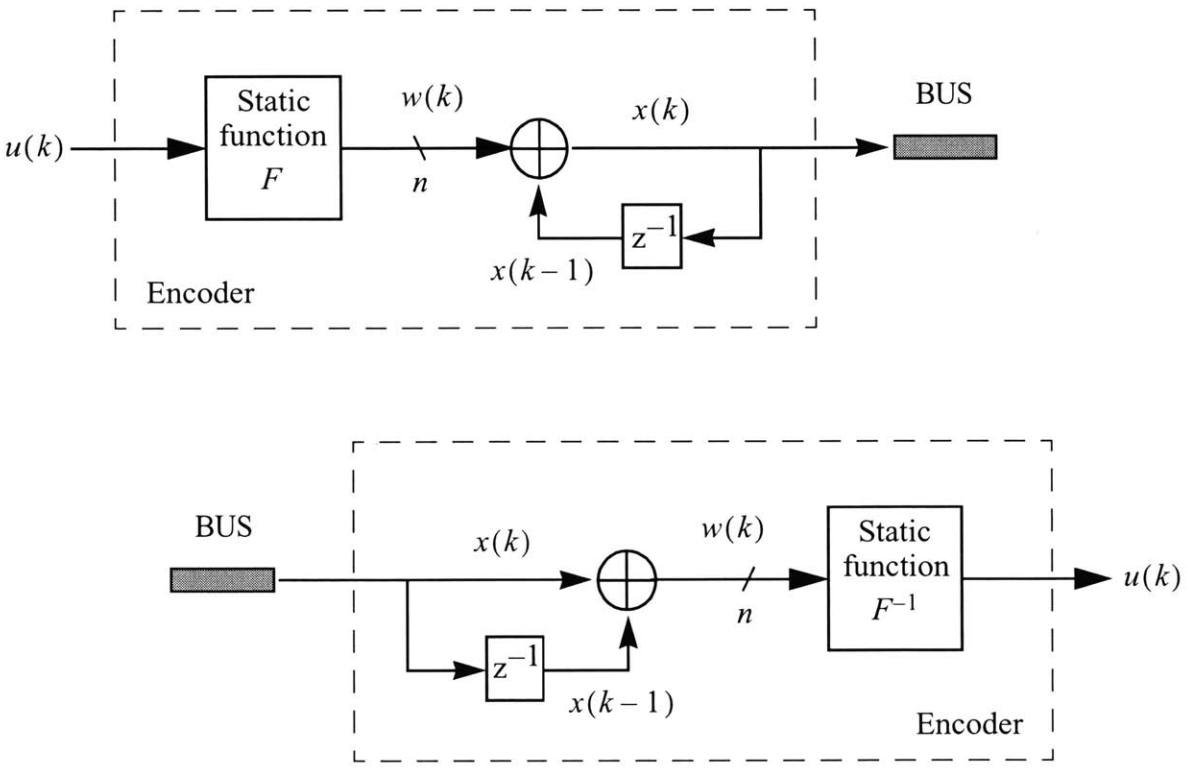

Figure 16: Differential $RLL(1, \infty)$ coding schemes for bus speed increase

At time $k = 1, 2, \ldots$, the n-bit vector $x(k)$ is transmitted through the n-line bus. Both isolated and not isolated buses (Figure 1 and Figure 5) are appropriate for differential $RLL(1, \infty)$ coding. The vector $x(k)$ satisfies the relation:

$$x(k) = w(k) \oplus x(k-1) \tag{28}$$

where $\oplus$ stands for vector *exclusive or*. The vector $w(k)$ equals $F(u(k))$. The function $F(.)$ maps the input data $u(k)$ at time k, to the *codeword* $w(k)$. The factor by which we can accelerate the bus, Equations (2), depends on the properties of the transitions $(x(k-1), x(k))$ that result by the encoding.

Here we make a simple observation. If the vector $w(k) = (w_1(k), w_2(k), \ldots, w_n(k))$ *does not have any two successive ones*, that is, $w_j(k) \cdot w_{j+1}(k) = 0$ for every $j = 1, 2, \ldots, n-1$, then because of Table 1 or Table 4 of Chapter 7, we see that no line will experience normalized delay of more than $(1 + 2\lambda)$ during the transition $(x(k-1), x(k))$. Therefore, the transition $(x(k-1), x(k))$ belongs to M_2. But, the rule, "no successive ones" defines the Run Length Limited codes $(1, \infty)$ [11]. Lets call C, the set of the $RLL(1, \infty)$ codewords in $\{0, 1\}^n$, we have $C \subset M_2$.

We know that if the transitions belong to M_2, then we can accelerate the bus by the factor θ_2, Equation (2). The question now is: *what is the resulting utilization of the bus when* $w(k) \in C$ *for every* k*?*. It is well known [9] (and very easy to show) that:

$$|C| = F_{n+1} \tag{29}$$

where F_m is the *m-th* term of the Fibonacci sequence,

$$F_m = \frac{1}{\sqrt{5}}\left\{\left(\frac{1+\sqrt{5}}{2}\right)^{m+1} - \left(\frac{1-\sqrt{5}}{2}\right)^{m+1}\right\} \tag{30}$$

Therefore, the utilization of the bus by the differential $RLL(1, \infty)$ coding scheme is given by the ratio:

$$a_{RLL} = \frac{1}{n}\log_2(F_{n+1}) \tag{31}$$

Asymptotically, as n becomes large, the utilization approaches $\log_2\left(\frac{1+\sqrt{5}}{2}\right) \cong 0.6942$. This means that for $\Lambda = 5$, and so acceleration factor $\theta_2 = 1.91$, and large buses, the net result is 1.326, i.e. 32% increase of the speed. In Figure 17.

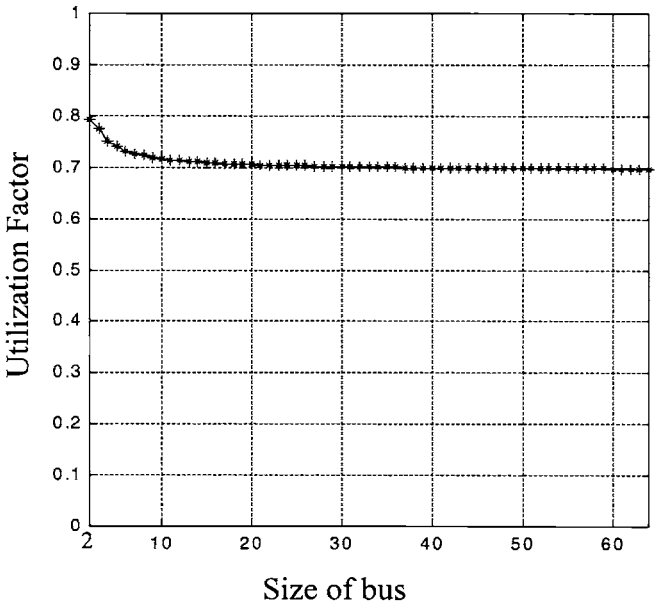

Figure 17: Utilization factor of the differential $RLL(1, \infty)$ coding scheme.

6. CONCLUSIONS

The idea of coding for increasing the communication speed through deep sub-micron buses was studied in the chapter from a more theoretical point view. The analysis was based on the classification of the transitions that was introduced in Chapter 7. The results were very promising and it was really surprising that the speed can be increased in some cases by more than 100% simply by

removing some slow transitions. It is true that a more accurate classification of the transitions based on a more detailed delay estimation or measurement would result in even better possible performances. Also, the bounds that were developed in Section 4.1 can be tighten if one takes into account the particular structure of matrices M_1, M_2 and M_3. Throughout the Chapters 7 and 8 we studied only "synchronous" buses, in the sense that the clocking of the bus is fixed. It is possible to achieve much higher speeds if we allow for "asynchronous" buses, that is, if we give to every transition the amount of time it needs to get completed and no more. (a "ready" signal would be needed). This case is modeled exactly as a Shannon's constrained noiseless channel [1]. The theory in [6] directly provides answers for the asynchronous problem.

Finally, the coding example using differential $RLL(1, \infty)$ coding is promising for practical applications, especially in modern bus designs were the bus is partitioned within small blocks of 2-4 lines (fabrics).

References

[1] C. Shannon, "A Mathematical Theory of Communication", *Bell Syst. Tech. J., Vol. 27, pp. 379-423, Jul. 1948.*

[2] P. Siegel, "Recording codes for digital magnetic storage", *IEEE Trans. Magnetics, vol. MAG-18, No. 6, Nov. 1984, pp. 1250-1252.*

[3] A. Palel, "Zero modulation encoding in magnetic recording", *IBM J. Res. Develop., vol. 19, No. 4, July 1975, pp. 366-378.*

[4] T. Weigandt, *Magneto-optic recording using a (2,18,2) run-length-limited code*, S.M. Thesis, Mass. Inst. Technol., Cambridge, MA, 1991.

[5] F. Dolivo, "Signal processing for high density digital magnetic recording", *in Proc. COMPEURO 89, Hamburg, Germany, May 1989.*

[6] P. Sotiriadis, A. Chandrakasan, "Reducing Bus Delay in Submicron Technology Using Coding", *ASP-DAC 2001.*

[7] B. Marcus, P. Siegel and J. Wolf, "Finite-State Modulation Codes for Data Storage", *IEEE Journal on selected areas in communications, Vol. 10, No. 1, Jan. 1992.*

[8] F. Harary, *Structural models: an introduction to the theory of directed graphs*, New York, Wiley 1965.

[9] J. Van Lint and R. Wilson, *A course in combinatorics*, Cambridge University Press 1999.

[10] H. Minc, *Nonnegative Matrices*, John Wiley & Sons 1987.

[11] K. Schouhamer Immink, "Runlength-limited sequences", *Proc. IEEE, Vol. 78, No. 11, Nov. 1990, pp. 1745-1759.*

Chapter 9

Conclusions and Future Directions

1 Conclusions

This thesis addresses two major issues of parallel communication in digital circuits, the power consumption and the communication speed.

The energy consumption associated with data transmission through DSM technology buses has been studied taking into account the capacitive and inductive coupling between the bus lines as well as the distributed nature of the wires. This has resulted in a new bus energy model as well as expressions for the expected power consumption. The concept of the *Transition Activity Matrix* has been introduced.

A theoretical framework has been developed to study the ultimate relation between processing or communication rate and energy per information bit in the case of synchronous digital systems. Given such a system, the theory provides us with the fundamentally minimum energy, per input information bit, that is required to process or communicate information at a certain rate. The minimum energy was proven to be asymptotically achievable using coding. The framework applies directly to combinational circuits. An immediate extension of the theory allows the treatment of finite state machines as well. Combining these results with the bus energy model introduced in the thesis, we derive the ultimate performance limits of coding for power reduction in DSM buses.

The application of linear, block linear and differential coding schemes on DSM buses has been studied from an energy perspective. The resulting power consumption has been evaluated exactly in some cases and bounded in some others. It has been shown that these schemes never reduce power consumption. Also, conditions have been found for linear codes to be energy-optimal within the set of all equivalent codes. These conditions can be applied for optimally re-structuring existing linear coding schemes used for error-correction. The results provide guidelines for energy-optimizing the other two coding classes as well.

The class of Finite State Machine encoders has been studied as an approach for power reduction in buses. The deep sub-micron bus energy model has been used to evaluate their power reduction properties. Two algorithms have been developed that derive very efficient coding schemes, called Transition Pattern Coding schemes. Both algorithms take into account statistics of the data and the particular structure of the bus. This coding design approach is generally applicable to any discrete channel with transition costs.

A charge recycling scheme appropriate for DSM buses has been mathematically analyzed and the theoretical limits of power reduction have been derived. It is shown that for large buses the power reduction can be up to 50%. Power is reduced further when charge recycling is combined with Bus Invert coding. An efficient modular circuit implementation is also discussed that demonstrates the practicality of the technique and its significant net power reduction.

The new concept of *Coding For Speed* in DSM buses has been introduced. It is based on a new approach for estimating the transitions delays. Keeping only "fast" transitions and clocking the bus faster can result in net speed increase of more than 120%. The theoretical limits of *Coding For Speed* have been derived for narrow buses and upper and lower bounded for wide buses. A class of practical coding schemes for speed, termed *Differential* $RLL(1,\infty)$ schemes, has been introduced. These schemes have low complexity while achieving a significant increase in communication throughput.

2 Future Directions

Applications : The present work provides many opportunities for practical implementations. Chapter 3 provides the limits of performance of coding for power reduction. It is desirable to build circuits that approach these limits. Towards this direction we can use the TPC algorithm of Chapter 5. It is important to design the encoder and decoder so that they consume as little power as possible. Although the theory provides us the guidelines and performance limits, the final result will depend strongly on the efficiency of the hardware implementation. *Coding for speed* is much more flexible in terms of circuit implementation. The complexity of the encoder/decoder is not critical in general. The results in net speed increase can be remarkable even with simple schemes like the *Differential* $RLL(1,\infty)$. A 35% increase can be easily achievable. Higher performances will most probably require a more complicated coding scheme that has the form of a finite state machine. Another practical challenge in *coding for speed* is the comparison between the original and the encoded buses in terms of clock speed. How do we measure how much faster the encoded one can be clocked? In some sense we need to "compare" the distortion in the signals in the buses. Finally, the theoretical bounds on power reduction using charge recycling show that there is a lot of room for circuit improvements. New techniques can be introduced as well.

Theory : There are many theoretical directions as well. One of the most challenging one is to establish a measure of the complexity of circuit implementations of encoders/decoders and incorporate it into a total performance criterion. This is more meaningful in coding for power reduction. Chapter 3 provides the basis towards an Energy-Information theory. It would be interesting to expand this framework in the case of networks of computation and/or communication devices. Another direction could be the treatment of devices introducing random errors. Regarding the coding for speed, it would be important to derive new classes of efficient coding schemes with low complexity. As it was presented in Chapter 8, there is a lot of room for advances between the practically useful *Differential* $RLL(1,\infty)$ schemes and the ultimate limits.