INFORMATION TO USERS

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

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

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

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

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

Bell & Howell Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

On-Chip Inductance in High Speed Integrated Circuits

by

Yehea I. Ismail

Submitted in Partial Fulfillment
of the
Requirements for the Degree
Doctor of Philosophy

Supervised by

Professor Eby G. Friedman

Department of Electrical and Computer Engineering
The College
School of Engineering and Applied Sciences

University of Rochester
Rochester, New York
2000

"*One remains knowledgeable until he thinks he knows enough.*"

Dedication

This work is dedicated to my parents: Professor Dr. Ismail and Professor Dr. Tahani, to my wife Heba, and to my daughter Mariam.

Curriculum Vitae

The author was born in Giza, Egypt on November 11, 1971. He attended the school of engineering, Department of Electronics and Communications at Cairo University from 1988 to 1993 where he received his B.Sc. degree in Electronics and Communications Engineering with distinction and honors. During his study, he was a recipient of the outstanding student academic award for four consecutive years (1989-1993) from Cairo University. As the top of his class, he was appointed as a teacher assistant at the Department of Electrical and Computer Engineering, Cairo University on August 1993. He received his first Masters degree in Electronics from Cairo University (distinction), Egypt on June 1996. He came to the University of Rochester on September 1996 where he received his second Masters in Electrical Engineering from the University of Rochester in 1998. He is currently working towards his Ph.D. degree in the area of high performance VLSI circuit design.

He was with IBM Cairo Scientific Center (CSC) from 1993 to 1996 on a part time basis and worked with IBM Microelectronics at East Fishkill, NY during the summers of 1997, 1998, and 1999. His primary research interests include interconnect, noise, innovative circuit simulation, and related circuit level issues in high performance VLSI circuits.

Acknowledgements

First of all, I would like to express my deep and sincere gratitude towards Professor Eby G. Friedman, my academic supervisor, whose kindness, fairness, and enthusiasm have brought out the best of me. His wisdom, professionalism, commitment, and experience are some of his attributes that make him, in my opinion, one of the best supervisors in the country. Professor Friedman has spent significant amount of time and effort directing and teaching me as he does with everyone of his students. I look forward to a lifetime relationship with him and I hope I will find someone as wonderful as he is wherever my next job will be. I intend to follow his example in many respects.

I would like to express my appreciation to the University of Rochester and the Department of Electrical and Computer Engineering which have a very unique environment that encourages high quality research and supports and respects graduate students. I would like to thank Professor Tom Hsiang, Professor Thomas B. Jones, Professor Mark Bocko, and Professor Judy Pipher for acting as members of my Ph.D. committee. I would also like to thank Professor David Albonesi for his support. I would like to thank every professor in the Department of Electrical and Computer Engineering since each of them has contributed to me in some way or another. I

would also like to thank all of the department secretaries and the network and computer staff for their active and valuable support.

The IBM Electronic Design Automation (EDA) group at East Fishkill deserves credit for their active support of my research both technically and financially. IBM has provided me with summer jobs for all of the summers of my Ph.D. where I received access to top of the line technologies and tools and was exposed to highly practical problems. IBM EDA has also funded my research by grants to our lab. More specifically, I would like to thank Dr. Jose L. Neves for his valuable support and advice. I would also like to thank my managers at IBM: Bob Maier, Ted Will, John Sayah, John Darringer, Jerry Kaminsky, and Jennifer Howland for their support.

To my parents, thank you for supporting me in every stage of my life and my debt to you I can never pay back. To my wife, thank you for bearing with me through hard years and your support was crucial for me during my Ph.D.

This research was made possible in part by support from the National Science Foundation under Grant No. MIP-9610108, the Semiconductor Research Corporation under contract No. 99-TJ-687, a grant from the New York State Science and Technology Foundation to the Center for Advanced Technology - Electronic Imaging Systems, and by grants from the Xerox Corporation, IBM Corporation, Intel Corporation, Lucent Technologies Corporation, and Eastman Kodak Company.

Abstract

It has become well accepted that interconnect delay dominates gate delay in current deep submicrometer VLSI circuits. With the continuous scaling of technology and increased die area, this situation is becoming worse. In order to properly design complex circuits, more accurate interconnect models and signal propagation characterization are required. Initially, interconnect has been modeled as a single lumped capacitance in the analysis of the performance of on-chip interconnects. With the scaling of VLSI technologies, narrower and longer wires with significant resistance have become commonplace. Currently, *RC* models are used to analyze high resistance nets and capacitive models are used for less resistive interconnect. However, inductance is becoming more important with faster on-chip rise times and longer wire lengths. Wide wires are frequently encountered in clock distribution networks and in upper metal layers. These wires are low resistance wires that can exhibit significant inductive effects. Furthermore, performance requirements are pushing the introduction of new materials for low resistance interconnect. One important example of these low resistance conductors is copper interconnect which has less resistivity as compared to the traditionally used aluminum interconnect. Copper interconnect is already used in many commercial CMOS technologies.

Inductance is therefore becoming an integral element in VLSI design and analysis methodologies. On-chip inductance is the focus of this dissertation in terms of anticipating related effects in VLSI circuits as well as developing appropriate design and analysis tools and methodologies that consider inductance in integrated circuits. Analytical expressions supported by simulation are provided to evaluate the importance of including inductance. These expressions characterize the magnitude of the error exhibited in the signal waveform and propagation delay when neglecting inductance and an *RC* model is used to model the interconnect. The effects of neglecting inductance on current design methodologies such as repeater insertion are also characterized to determine the penalty incurred if inductance is neglected. Fast and accurate tools for analyzing transmission lines and *RLC* have been developed. An equivalent Elmore delay is introduced for application to optimization and design methodologies. A repeater insertion algorithm targeting copper-based interconnect *RLC* trees is also introduced and used to optimize the delay across several industrial trees based on the equivalent Elmore delay model. It is shown that by neglecting inductance a significant cost in terms of delay, area, and power is incurred. Many industrial and experimental results are also presented throughout the dissertation in support of the presented theory.

Contents

List of Tables

List of Figures

Chapter 1 Introduction

Of the many technologies available for designing and building integrated circuits (IC), the complementary metal oxide semiconductor (CMOS) technology has the dominant share of the IC market [1]-[3]. The word *complementary* refers to the existence of two different types of MOS transistors that complement each other which are typically connected to eliminate any static power consumption. The two types of transistors are the N-channel (or NMOS) transistor and the P-channel (or PMOS) transistor. These two types of transistors are shown in Figure 1.1. The transistors are termed metal oxide semiconductor (MOS) due to the physical structure of the transistors. The gate is composed of a conducting material, typically polycrystalline silicon (polysilicon), although historically metal was originally used [1]-[3]. The gate is separated from the semiconductor silicon channel by a silicon dioxide (SiO_2) insulating layer. The channel connects the drain to the source of the transistor. For an NMOS transistor, a high voltage (usually V_{DD} in a digital circuit) applied at the gate creates a path (or channel), permitting electrons to flow between the drain and the source, thereby turning the transistor on. A low voltage (usually ground in a digital circuit) eliminates the path (or channel), turning the transistor off. These conditions of operation are reversed for a PMOS transistor. The existence of an insulating layer of SiO_2 between the gate, which acts as the input or control point of the transistor, and

the channel eliminates any current from passing into the channel or substrate from the input gate. This feature significantly reduces the power consumption of a MOS transistor based circuit as compared to the power consumption of circuits based on bipolar or MESFET GaAs transistors as well as provide isolation between transistors. The extremely high (effectively infinite) impedance seen at the input of a CMOS gate permits a full voltage swing, resulting in improved noise margins in digital circuits.

Efficient digital logic families have been developed that eliminate static power consumption in CMOS logic gates [4]-[5]. The most well known CMOS circuit structure is the CMOS inverter. A circuit schematic of a CMOS inverter is shown in Figure 1.2. When a high voltage is applied at the input of a CMOS inverter, the NMOS transistor turns on while the PMOS transistor turns off, forcing the output low. When a low voltage is applied at the input, the PMOS transistor turns on while the NMOS transistor turns off, forcing the output high. In both cases, the path from the supply to ground is eliminated by one of the two transistors that is off. Thus, the only power consumed by a CMOS gate is the transient switching power [6]-[8] and any leakage current through the off transistors. CMOS circuits, therefore, dissipate much less power than bipolar and MESFET based logic families. This power advantage permits much higher integration densities with CMOS circuits as compared to bipolar and MESFET circuits since the heat that must be removed is much less in CMOS circuits. Bipolar and MESFET technologies cannot achieve such high integration densities since the high level of power dissipated by circuits composed of many transistors manufactured in these technologies cannot be efficiently removed. On the speed side, CMOS has historically been slower than bipolar and MESFET

circuits. However, with the continuous scaling of the minimum feature size (primarily the channel length) of CMOS technologies, the speed of CMOS has steadily improved, reaching the same speed characteristics as most bipolar and MESFET circuits [9].

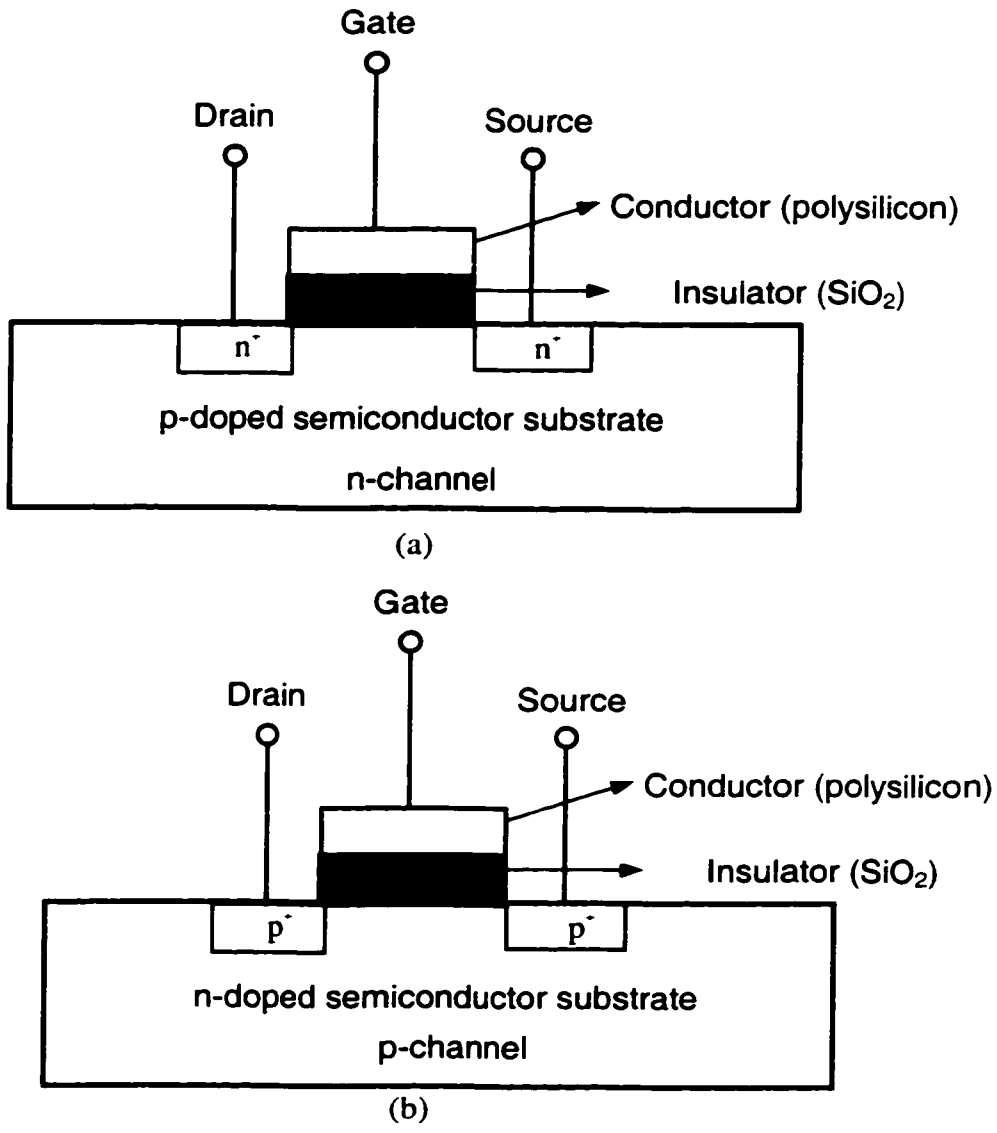


Figure 1.1 MOS transistors. a) NMOS transistor. b) PMOS transistor.

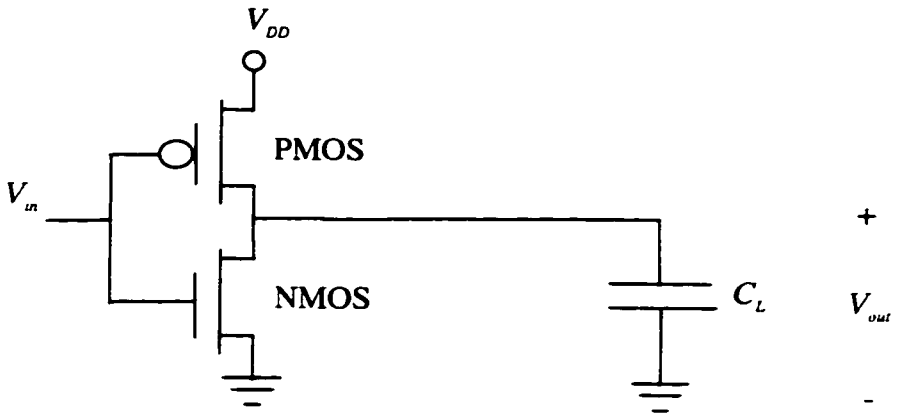


Figure 1.2 A CMOS inverter with a capacitive load

A CMOS VLSI circuit consists of active transistor devices (PMOS and NMOS transistors) and passive interconnects which connect the transistor devices. The active devices have capacitive parasitic impedances due to the gate-to-channel, source-to-channel, drain-to-channel, gate-to-drain, and gate-to-source capacitances as shown in Figure 1.3 [1]-[3]. The parasitic capacitances of the driving and driven CMOS gates are charged and discharged by the driving gates during the switching process. Another source of parasitic impedances is the interconnect. The interconnect has capacitances to the substrate, capacitances to the neighboring interconnect on the same layer, and capacitances to the neighboring interconnect in other layers, as shown in Figure 1.4. Moreover, each interconnect line also has an associated self resistance and self inductance.

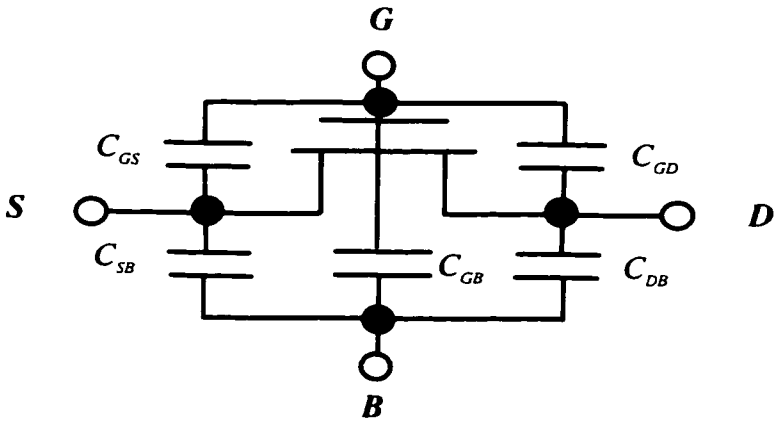


Figure 1.3 Parasitic capacitances associated with an MOS transistor.

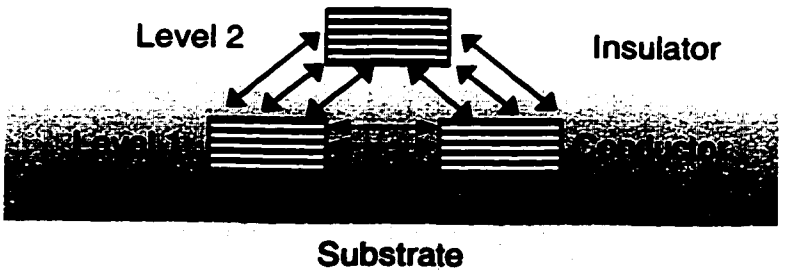


Figure 1.4 Electric field lines associated with VLSI interconnect

Historically, the gate parasitic impedances have been much larger than the interconnect parasitic impedances since the gate geometries (the width and length) were so large (about 5 μm in 1980). Thus, interconnect parasitic impedances have historically been neglected and the interconnect was modeled as a short circuit. With

the scaling of the minimum gate feature size, interconnect capacitances have become comparable to the gate capacitance, requiring the interconnect to be modeled as a single lumped capacitance [10]-[12] that is added to the gate capacitance as shown in Figure 1.5(a). With this interconnect model, new design techniques emerged to drive large capacitive loads associated with long global interconnects and large interconnect trees with high fanout. Cascaded tapered buffers are used to minimize the propagation delay of CMOS gates driving these large capacitive loads [13]-[20]. With increasing device densities per unit area, the interconnect density has also correspondingly increased. Thus, the crosssectional area of interconnects has been reduced to provide more interconnect per unit area [9]. Also, the improved yield of CMOS fabrication processes permits the manufacture of larger chips with higher reliability. Thus, the global wires connecting modules across a chip have increased in length. Both the decreased crosssectional area and the increased wire length have caused the global wire resistances to dramatically increase [21], [22]. The interconnect model now includes the resistance of the interconnect as shown in Figure 1.5(b). Including resistance in the interconnect model dramatically affects the behavior of the interconnect and hence the design techniques used to optimize the performance of the interconnect [24]-[32]. With a short-circuit or a capacitive interconnect model, the interconnect could be treated as a single node. However, by including series resistance, the interconnect now is composed of multiple nodes, each node having a different voltage waveform. This characteristic has greatly complicated the analysis of circuits with resistive interconnect [33]-[35].

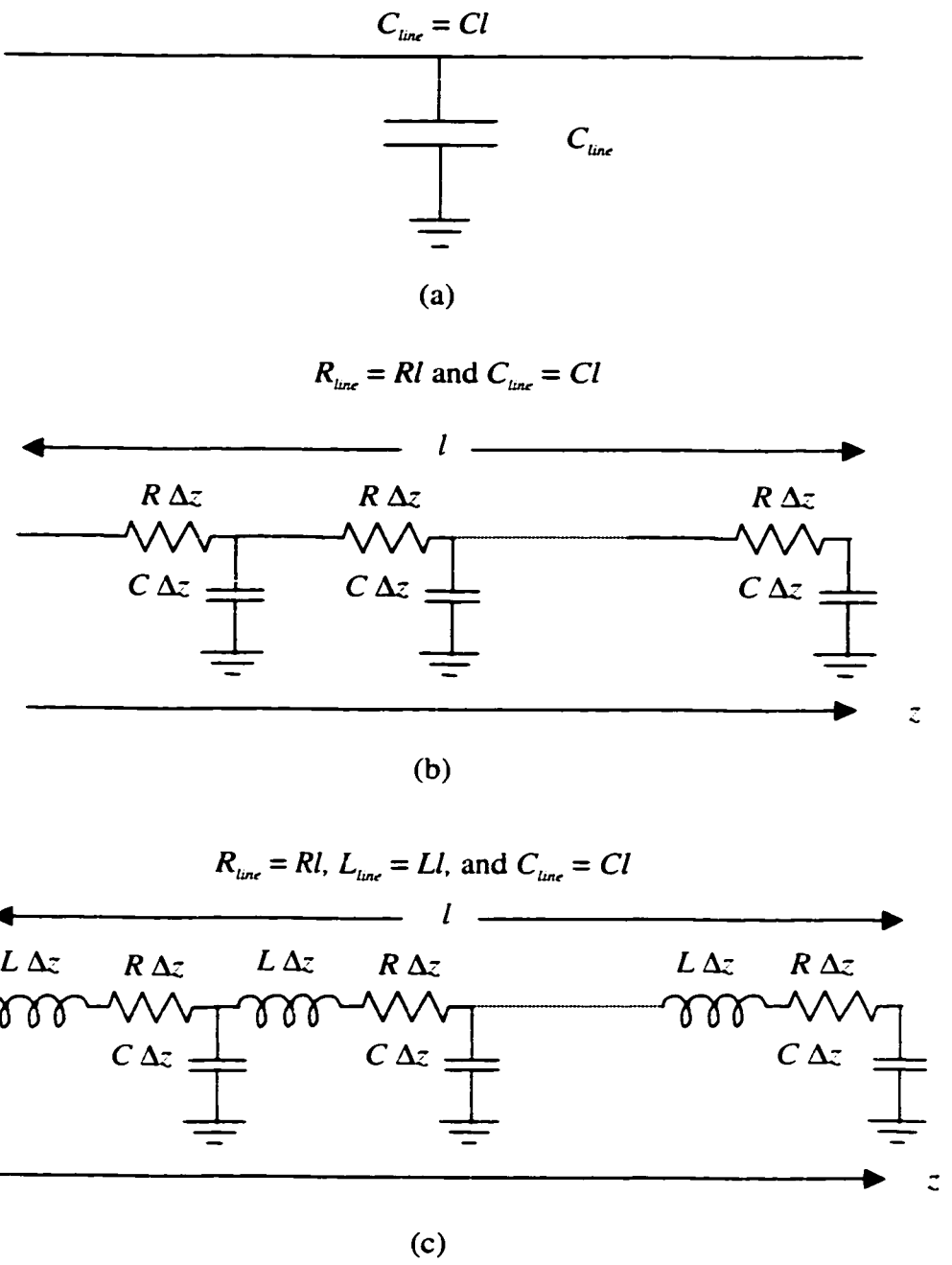


Figure 1.5 Evolution of interconnect models. a) Capacitive model. b) *RC* model. c) *RLC* model.

Another difference in the behavior of the interconnect when resistance is non-negligible is the dependence of the signal delay and rise time on the line length. The delay and rise time of a CMOS gate driving another CMOS gate with a connecting resistive wire (as shown in Figure 1.6) can be viewed as the superposition of three terms [36]. The first term is the delay component caused by the driving gate charging the local drain/source parasitic capacitances as well as the parasitic capacitances of the driven gate. This term is independent of the length of the interconnect and is described as the gate delay. The second term is due to the driving gate charging the interconnect capacitance and is linear with the length of the interconnect. The third term is due to the *RC* time constant of the interconnect and is quadratic with the length of the interconnect. A qualitative plot describing this behavior is shown in Figure 1.7. This quadratic increase of interconnect delay with interconnect length significantly degrades the performance of circuits with long wires. In order to reduce this quadratic increase in delay with interconnect length, repeaters (or a series of CMOS inverters separated by interconnect) are inserted along these long interconnect lines to partition the lines into shorter sections, thereby reducing the total delay of the interconnect path [37]-[44].

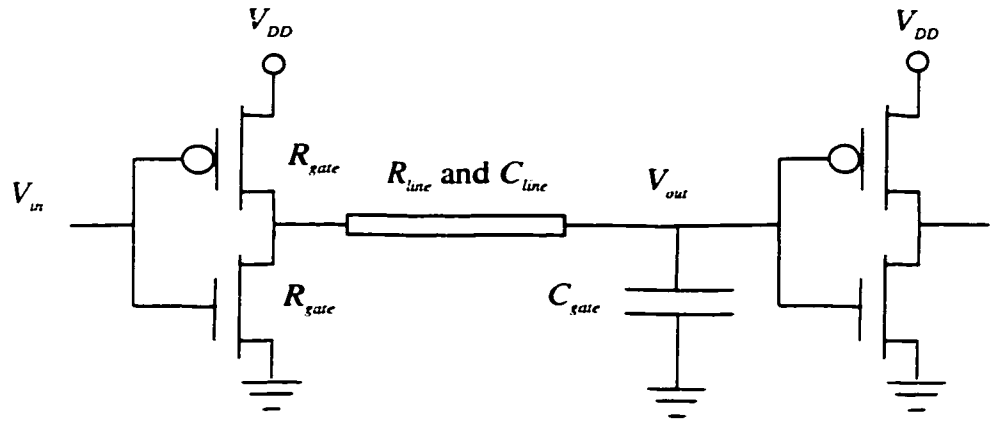


Figure 1.6 A CMOS gate driving another CMOS gate with a resistive-capacitive interconnect wire connecting the two inverters.

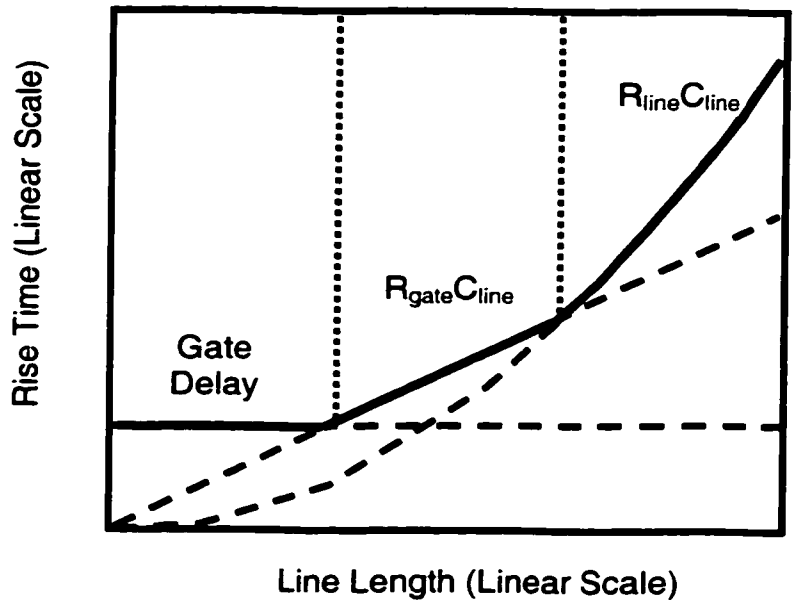


Figure 1.7 Dependence of the delay of the circuit shown in Figure 1.6 on the length of the interconnect.

On-chip inductance has currently become more important with faster on-chip rise times and wider wires [45]-[68]. Wide wires are frequently encountered in clock distribution networks and in upper metal layers [49]-[65]. These wires are low resistance lines that can exhibit significant inductive effects. Furthermore, performance requirements are pushing the introduction of new materials such as copper interconnect for low resistance interconnect [57], [61] and new dielectrics to reduce interconnect capacitance [57], [61]. These technological advances increase the importance of inductance as is described in this dissertation. In the limiting case, the advent of high critical-temperature superconductors has created the possibility of realizing high density, extremely high speed interconnects for VLSI circuits [69]. These superconducting interconnects have zero DC resistance and are highly sensitive to inductance. These interconnects are best modeled as lossless transmission lines.

A primary focus of this dissertation is to evaluate the importance of inductance in current technologies as well as to anticipate future trends. This dissertation also deals with the change in interconnect behavior with increasing inductance effects and the impact of this behavioral change on current design methodologies for VLSI interconnects. Methods for analyzing *RLC* interconnect such as closed form expressions for characterizing the delay of *RLC* lines and trees are also introduced. The important problem of inductance extraction is not considered in this dissertation, but rather design methodologies given a value of the interconnect inductance is the primary focus.

The dissertation is organized as follows. Basic transmission line theory is briefly reviewed in Chapter 2. The theory of transmission lines is discussed for three types of transmission lines: lossless *LC* transmission lines, lossy *RLC* transmission lines, and *RC* transmission lines [70]-[72]. Understanding transmission lines behavior is essential to the design and analysis of interconnect in high speed integrated circuits since a transmission line model is the most accurate on-chip interconnect model which maintains the distributed nature of on-chip interconnect impedances.

The underlying concepts of the most commonly used methods to evaluate the transient response of linear networks in VLSI circuits are reviewed in Chapter 3. Simple methods to calculate the delay of signals in an *RC* tree such as the Elmore delay [73] and Wyatt's approximation [74] are introduced in section 3.1 since extensions of these methods for *RLC* trees are presented in subsequent chapters. The calculation of more accurate transient responses for *RLC* interconnects via moment matching is discussed in section 3.2 [75]-[107].

The basic operational theory and characterization of MOS transistors are briefly reviewed in Chapter 4. Accurate characterization of the behavior of a MOSFET transistor is crucial for VLSI circuit analysis and design. The basic long channel behavior of a MOSFET device is first discussed. This long channel transistor model is followed by a discussion of the alpha power law model that characterizes the current-voltage behavior of a MOSFET transistor for deep submicrometer technologies [108].

Simple figures of merit to characterize the importance of on-chip inductance are introduced in Chapter 5 based on a single lossy transmission line analysis [66]-[68]. A closed form solution for the output signal of a CMOS inverter driving an *RLC* transmission line is presented. This solution is based on the alpha power law model for deep submicrometer technologies. Two figures of merit are presented that are useful for determining if a section of interconnect should be modeled as either an *RLC* or an *RC* impedance. The damping factor of a lumped *RLC* circuit is shown to be a useful criterion. The second useful figure of merit considered in this chapter is the ratio of the rise time of the input signal at the driver of an interconnect line to the time of flight of the signals across the line. One primary result of this study is evidence demonstrating that a range for the length of the interconnect exists for which inductance effects are prominent. Furthermore, it is shown that under certain conditions, inductance effects are negligible for any length of the section of interconnect.

In Chapter 6, a closed form solution for the propagation delay of a CMOS gate driving a lossy transmission line with a terminating CMOS gate is introduced [109]-[112]. It is shown that the error in the propagation delay if inductance is neglected and the interconnect is treated as a distributed *RC* line can be over 35% for current on-chip interconnect. It is also shown that the traditional quadratic dependence of the propagation delay on the length of the interconnect for *RC* lines approaches a linear dependence as inductance effects increase. This closed form solution is used to derive expressions for optimum repeater insertion within an *RLC*

interconnect line for minimum propagation delay. The repeater insertion expressions for *RLC* lines are compared to traditional expressions for *RC* lines to characterize the importance of including inductance in current design methodologies. It is shown that significant penalties are incurred in terms of the delay, area, and power when inductance is neglected and an *RC* circuit is used to model the interconnect.

Waveform characterization of signals at different nodes of an *RLC* tree is discussed in Chapter 7 [113]-[115]. A second order approximate transfer function is presented for a node of an *RLC* tree which can be used with an arbitrary input to determine the signal waveform at any node of an *RLC* tree. For the specific case of a step input, simple closed form solutions for the propagation delay, rise time, overshoots, and settling time are presented. These solutions have the same accuracy characteristics of the Elmore delay model for *RC* trees and preserves the simplicity and recursive characteristics of the Elmore delay model. Specifically, the complexity of calculating the time domain responses at all of the nodes of an *RLC* tree is linearly proportional to the number of branches in the tree where the solutions are always stable. The closed form expressions introduced here consider all damping conditions of an *RLC* circuit including the underdamped response, which is not considered by the Elmore delay model due to the non-monotone nature of the response. The continuous analytical nature of the solutions makes these expressions suitable for design methodologies and optimization techniques. Also, the solutions have significantly improved accuracy as compared to the Elmore delay for an overdamped response. The solutions introduced in Chapter 7 for *RLC* trees can be practically used for the same purposes that the Elmore delay model is used for *RC* trees.

In Chapter 8, the second order approximation discussed in Chapter 7 is used to determine figures of merit to characterize the importance of including inductance in the circuit model of a tree structured interconnect [116]. It is shown that the figures of merit based on a single line analysis in Chapter 5 can be misleading if applied to individual lines (or branches) of a tree.

The theoretical results of Chapter 6 for inserting repeaters in *RLC* lines are verified in Chapter 9 for a set of copper-based interconnect trees using a CAD tool for repeater insertion [117], [118] based on the equivalent Elmore delay model introduced in Chapter 7. The algorithm has a complexity proportional to the square of the number of possible repeater positions and determines a repeater solution that is close to the global minimum. The repeater insertion algorithm is used to insert repeaters within a variety of copper-based interconnect trees based on a 0.25 μm CMOS technology to minimize the maximum path delay based on both an *RC* model and an *RLC* model. It is shown that as inductance effects increase, the area and power consumed by the repeaters inserted to minimize the path delays of an *RLC* tree decreases. These results are in agreement with the theoretical results presented in Chapter 6.

The dynamic and short-circuit power consumption of a CMOS gate driving an *LC* transmission line as a limiting case of an *RLC* transmission line is described in Chapter 10 [8], [119]-[121]. Accurate closed form solutions for the output voltage and short-circuit power of a CMOS gate driving an *LC* transmission line are presented. The ratio of the short-circuit to dynamic power is shown to be less than 7%

for CMOS gates driving *LC* transmission lines where the line is matched or underdriven. The total power consumption is expected to decrease as inductance effects become more significant as compared to an *RC* dominated interconnect line. These results can be used to determine a limit on the effects of inductance on the power consumption of a CMOS gate since a lossless transmission line represents a limiting case for inductance effects.

In Chapter 11, it is shown that inductance has positive effects on the performance of VLSI circuits. It is shown both theoretically and experimentally that increasing inductance effects result in faster signal rise times, lower power consumption, and less active device area. Design methodologies can be developed to exploit these useful effects of on-chip inductance while maintaining noise at acceptable levels so as to guarantee the reliable performance of an integrated circuit. An industrial example of a high speed clock distribution network is presented to illustrate this behavior and design methodologies exploiting on-chip inductance are suggested.

A novel method for simulating *RLC* circuits is introduced in Chapter 12 that is both accurate and efficient [122]. The DTT (Direct Truncation of the Transfer function) method is introduced as an alternative to moment matching techniques to evaluate time domain signals within *RLC* trees with arbitrary accuracy in response to any input signal. This method depends on finding a low frequency reduced order transfer function by direct truncation of the exact transfer function at different nodes of an *RLC* tree. The method is numerically accurate for any order of approximation, which permits solutions with a large number of poles appropriate for approximating

RLC trees with underdamped responses. The method is computationally efficient with a complexity linearly proportional to the number of branches in an *RLC* tree. A common set of poles are determined that characterize the responses at all of the nodes of an *RLC* tree which further enhances the computational efficiency. Stability is guaranteed by the DTT method for low order approximations with less than five poles. Such low order approximations are useful for evaluating monotone responses exhibited by *RC* circuits.

Comments on the extraction and sensitivity of on-chip inductance are presented in Chapter 13 [123]. The efficient and accurate extraction of inductance is one of the primary bottlenecks that hinder incorporating on-chip inductance within integrated circuit design tools. Fortunately, as is shown in Chapter 13, on-chip inductance has two useful characteristics which enable simple methods to be used to extract the inductance. The first characteristic is that the sensitivity of a signal waveform to errors in the inductance values is low, particularly the propagation delay and the rise time. The second characteristic is that the magnitude of the on-chip inductance is a slow varying function of the width of a wire and the geometry of the surrounding wires. These two characteristics can be exploited by using simplified techniques that permit approximate and sufficiently accurate values of the on-chip inductance to be extracted with high computational efficiency.

Conclusions are provided in Chapter 14 and future work is described in Chapter 15. In Appendix A, practical industrial numbers to characterize the importance of inductance in current VLSI circuits are presented. A mathematical proof of the expressions for optimum repeater insertion in an *RLC* line is provided in

Appendix B. The algorithmic complexity of the proposed delay model for *RLC* trees described in Chapter 7 is described in Appendix C. The matching conditions of a CMOS gate driving a lossless transmission line are given in Appendix D. The algorithmic implementation of the simulation method presented in Chapter 12 is provided in Appendix E. Finally, the DTT method discussed in Chapter 12 is compared to simulation techniques based on moment matching in Appendix F.

Chapter 2 Basic Transmission Line Theory

In this chapter, the theory of transmission lines is briefly discussed. Three types of transmission lines are outlined: lossless *LC* transmission lines, lossy *RLC* transmission lines, and *RC* transmission lines [70]-[72]. Proper understanding and interpretation of transmission line behavior is essential to the design and analysis of interconnect in high speed integrated circuits. No closed form solution exists for the transient response of an *RC* transmission line as shown in section 2.1, therefore, approximate models for *RC* interconnect are presented in section 2.2 [22], [23]. Since the effects of inductance on the repeater insertion process is introduced later in this dissertation, some background behind the process of inserting repeaters into *RC* interconnect is introduced in section 2.3 [37]-[44].

2.1 Transmission Lines

A transmission line model can be used to accurately characterize on-chip interconnect. Transmission lines maintain the distributed nature of the interconnect impedances, including the effects of the inductance of the interconnect. A transmission line is a structure that supports the signal propagation of Transverse Electro-Magnetic Waves (TEM) [70]-[72]. The function of a transmission line, as the name implies, is to *transmit* electrical signals from one point to another point. Several

types of common transmission lines are shown in Figure 2.1. On-chip interconnect can most closely be modeled as a micro-strip or as a parallel plate transmission line [46]-[48]. The efficient transmission of signals from one point to another point requires that losses across the transmission line be low, that the attenuation and velocity of the waves across the transmission line be frequency independent to avoid distortion, and that appropriate matching of the source and load impedances with the characteristic impedance of the transmission line be maintained to avoid reflections [70]-[72].

A transmission line is composed of a series resistance, a series inductance, a shunt capacitance, and a shunt conductance such as the section of transmission line pictured in Figure 2.2. The per unit length resistance, inductance, capacitance, and conductance are denoted by R, L, C, and G, respectively. The resistance is the sum of the resistance of the two conductors of the transmission line (or in general, the wire resistance and the return path resistance). The inductance is the self-inductance of the transmission line. The capacitance is the capacitance between the two plates of the transmission line. The conductance is due to the current path caused by the losses within the dielectric. These parasitic impedances make the behavior of a transmission line deviate from ideal transmission characteristics. The behavior of a transmission line is discussed in greater detail in subsequent sections.

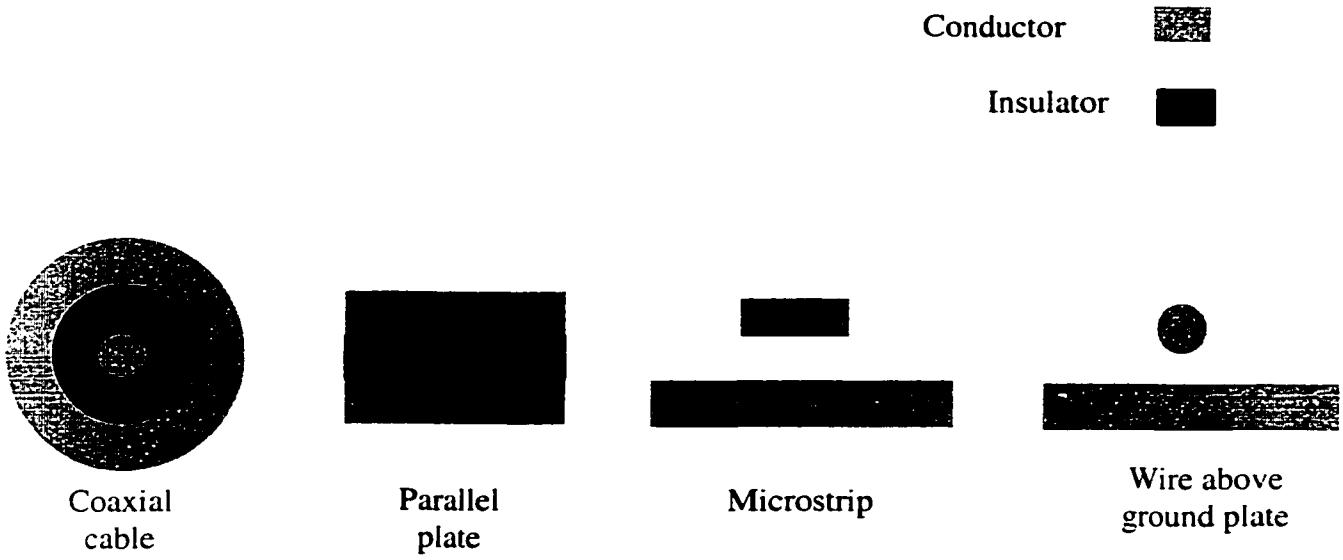


Figure 2.1 Crosssection of the most common types of transmission lines.

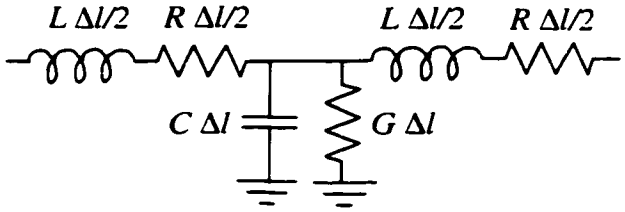


Figure 2.2 A section of an *RLCG* transmission line.

2.1.1 Lossless (Ideal) Transmission Lines

A *lossless* transmission line is a transmission line where the lossy elements (*R* and *G*) are negligible which is equivalent to a transmission line with perfect

conductors and a perfect dielectric. A discussion of lossless transmission lines is important since the relative simplicity of these lossless lines allows certain basic concepts underlying the behavior of general transmission lines to be better understood. Lossy transmission lines are described later in this chapter in light of the concepts discussed for lossless transmission lines.

A lossless transmission line is shown in Figure 2.3. Note that at low frequencies the inductance can be approximated by a short-circuit while the capacitance can be approximated by an open circuit. Thus, at low frequencies, a lossless transmission line behaves like a short-circuit which connects the two end points between which the electrical signal is being transmitted. Therefore, a low frequency lossless transmission line satisfies the requirements for ideal signal transmission. However, at high frequencies, the behavior of a lossless transmission line is quite different from a short-circuit. The delay of the waves propagating along the transmission line and signal reflections are two of the most important differences as compared to a shorted wire. The behavior of transmission lines at high frequencies is discussed in subsection 2.1.

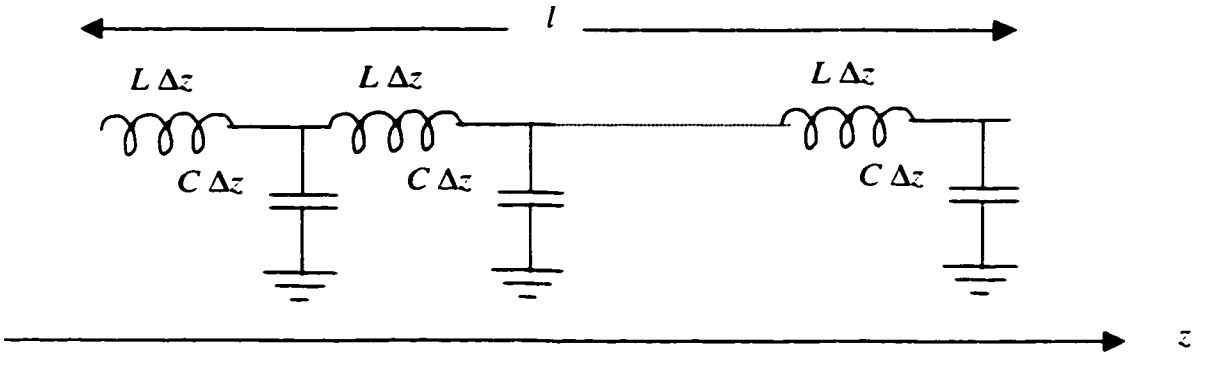


Figure 2.3 A lossless transmission line.

A. Time and Space Dependence of Signals in a Lossless Transmission Line

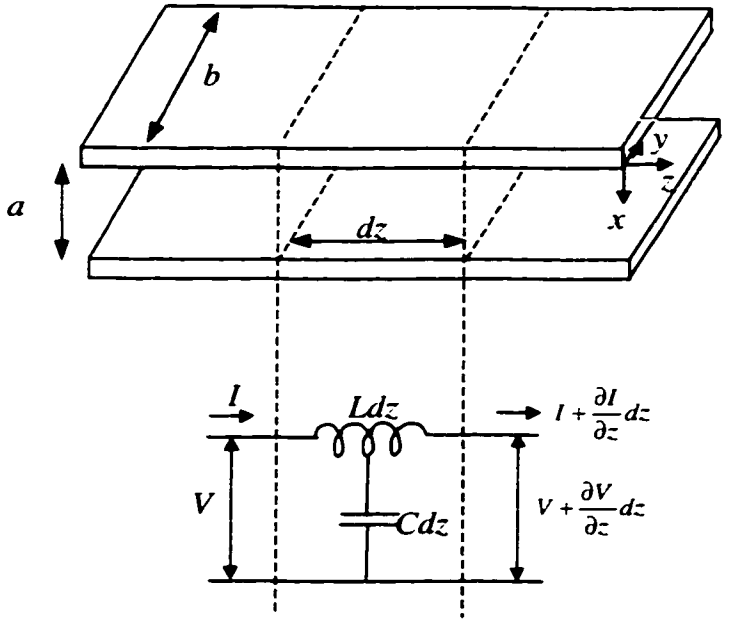


Figure 2.4 A section of a lossless parallel plate transmission line.

A circuit model for a differential section of a lossless transmission line is shown in Figure 2.4 [72]. Since the inductance and capacitance per unit length of the transmission line are L and C, respectively, a length of dz has an inductance of Ldz and a capacitance of Cdz. The change in voltage across this length dz is equal to the product of the inductance with the time rate of change of the current across the inductance and is

$$\text{Voltage change} = \partial V = -(Ldz)\frac{dI}{dt}. \tag{2.1}$$

Similarly, the change in current along the line at any specific time is merely the current that is shunted across the distributed capacitance (Cdz) and is

$$\text{Current change} = \partial I = -(Cdz)\frac{dV}{dt}. \tag{2.2}$$

Equations (2.1) and (2.2) can be rewritten as

$$\frac{\partial V}{\partial z} = -L\frac{\partial I}{\partial t}, \tag{2.3}$$

$$\frac{\partial I}{\partial z} = -C\frac{\partial V}{\partial t}. \tag{2.4}$$

Equations (2.3) and (2.4) are the fundamental differential equations characterizing a lossless transmission line. Partially differentiating (2.3) with respect to distance and (2.4) with respect to time, the resulting equations are

$$\frac{\partial^2 V}{\partial z^2} = -L\frac{\partial^2 I}{\partial z \partial t}, \tag{2.5}$$

$$\frac{\partial^2 I}{\partial z \partial t} = -C\frac{\partial^2 V}{\partial t^2}. \tag{2.6}$$

These two equations can be combined to eliminate the current component and determine a single second order differential equation that characterizes the voltage across a lossless transmission line and is

$$\frac{\partial^2 V}{\partial z^2} = LC\frac{\partial^2 V}{\partial t^2} = \frac{1}{v^2}\frac{\partial^2 V}{\partial t^2}, \tag{2.7}$$

where

$$v = \frac{1}{\sqrt{LC}}. \tag{2.8}$$

Equations (2.3) and (2.4) are known as the classical telegraph equations and (2.7) is the well-known one-dimensional wave equation [70]-[72]. A similar equation

describing the current within a lossless transmission line can be determined in the same manner and is

$$\frac{\partial^2 I}{\partial z^2} = \frac{1}{v^2}\frac{\partial^2 I}{\partial t^2}. \tag{2.9}$$

The solution of the wave equation in (2.7) is the sum of two waves, one wave propagating in the positive z direction with a speed v and the other wave propagating in the negative z direction with a speed v and is expressed mathematically as

$$V(z,t) = F_1(t - \frac{z}{v}) + F_2(t + \frac{z}{v}), \tag{2.10}$$

where F_1 and F_2 are arbitrary functions that are determined from the boundary conditions (the driver and load as well as the initial conditions). Equation (2.10) satisfies (2.7) by direct substitution. A constant value of $F_1(t\text{-}z/v)$ would be seen by an observer moving in the z direction with a velocity v. Similarly, $F_2(t+z/v)$ represents a wave moving in the negative z direction with velocity v.

To determine the current on the line in terms of the functions, F_1 and F_2, the expression of the voltage in (2.10) can be substituted into the transmission line expression (2.3),

$$-L\frac{\partial I}{\partial t} = -\frac{1}{v}F_1'(t - \frac{z}{v}) + \frac{1}{v}F_2'(t + \frac{z}{v}). \tag{2.11}$$

The result of integrating this expression with respect to t is

$$I = \frac{1}{Lv}\left[F_1(t - \frac{z}{v}) - F_2(t + \frac{z}{v})\right] + f(z). \tag{2.12}$$

If this solution is substituted into the second transmission line equation (2.4), it can be noted that the function of integration, f(z), could only be a constant. This constant is a

superposed DC solution which is not of particular interest in evaluating the wave solution and is therefore ignored. Equation (2.12) can therefore be written as

$$I = \frac{1}{Z_0}\left[F_1 \left(t - \frac{z}{v}\right) - F_2 \left(t + \frac{z}{v}\right) \right], \tag{2.13}$$

where

$$Z_0 = Lv = \sqrt{\frac{L}{C}}\ \Omega. \tag{2.14}$$

The constant Z_0 is called the characteristic impedance of the transmission line and is real (resistive) in the case of a lossless transmission line. Thus, the voltage across a lossless transmission line is the superposition of the waves traveling in the positive z direction and the waves traveling in the negative z direction. The speed of propagation of the waves across the transmission line is given by v in (2.8). For each voltage wave a companion current wave exists where the ratio between the voltage wave and the companion current wave is given by Z_0 (see Figure 2.5). The negative sign for the negatively traveling waves occurs since the wave propagates to the left (towards the source) while the convention in (2.13) for the current is positive if the wave propagates to the right (towards the load in terms of the circuit diagram shown in Figure 2.4).

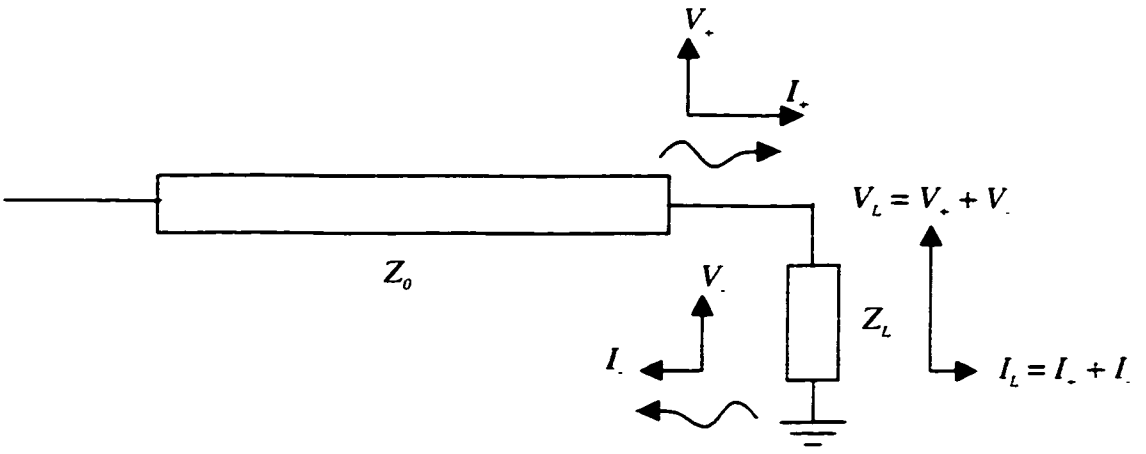


Figure 2.5 Reflections at an impedance discontinuity.

B. Relation of Electromagnetic Theory to Circuit Theory

Although distributed circuit theory has been used in the previous subsection to derive the basic equations that characterize the behavior of a lossless transmission line, it is shown here that these equations bear a strong relation to electromagnetic theory. To illustrate this relationship with electromagnetic theory, consider the parallel plate transmission line shown in Figure 2.4 with the conducting planes wide enough in the y direction so that fringing at the edges is negligible. A uniform plane wave with an electric field E_x polarized in the x direction and a magnetic field H_y polarized in the y direction propagates in the z direction between the transmission line plates. For such a transmission line, the inductance and capacitance per unit length are

$$L = \mu \frac{a}{b}, \tag{2.15}$$

$$C = \varepsilon \frac{b}{a}, \tag{2.16}$$

respectively, where μ is the magnetic permeability and ε is the electric permittivity of the dielectric medium between the transmission line plates, both of which are constant for a specific dielectric. The width of the plates is denoted *b* and the distance between the two plates is denoted *a*. Substituting (2.15) and (2.16) into (2.8), the velocity at which the waves travel along a transmission line is

$$v = \frac{1}{\sqrt{LC}} = \frac{1}{\sqrt{\mu\varepsilon}}. \tag{2.17}$$

The quantity in (2.17) is described in electromagnetic theory as the speed of light (for the electromagnetic waves) in a medium with magnetic permeability μ and electric permittivity ε. Thus, the wave travels in a lossless transmission line at a speed that equals the speed of light in the medium between the plates. Although (2.17) is based on a parallel plate transmission line model, the inductance and capacitance of any general transmission line satisfy relation (2.17).

Another interesting relationship between circuit theory and electromagnetic theory can be noted by examining Z_0 in (2.14). Substituting (2.15) and (2.16) into (2.14), the characteristic impedance is

$$Z_0 = \frac{V(z,t)}{I(z,t)} = \sqrt{\frac{L}{C}} = \frac{a}{b}\sqrt{\frac{\mu}{\varepsilon}} \ \Omega. \tag{2.18}$$

The voltage and current in the parallel plate transmission line shown in Figure 2.4 are given, respectively, by

$$V(z,t) = aE_x(z,t), \tag{2.19}$$

$$I(z,t) = bH_y(z,t). \tag{2.20}$$

Substituting for the voltage and current with the electric and magnetic fields, the ratio of the electrical to magnetic filed is

$$\frac{E_x(z,t)}{H_y(z,t)} = \sqrt{\frac{\mu}{\varepsilon}}\ \Omega. \tag{2.21}$$

The quantity in (2.21) is described in electromagnetic field theory as the wave impedance in a medium with magnetic permeability μ and electric permittivity ε. The wave impedance characterizes the ratio of the electrical field to the magnetic field of a uniform plane wave. Thus, the concepts presented in the previous subsection using distributed circuit theory are consistent with a physical perspective based on electromagnetic theory.

C. Reflection and Transmission at an Impedance Discontinuity

A voltage wave propagating across a transmission line has an associated current wave. The ratio between the voltage and the associated current is Z_0. When a voltage wave propagating across a transmission line reaches an impedance discontinuity such as a load impedance Z_L which is not equal to the characteristic impedance Z_0, a reflected wave is initiated such that the sum of the voltages and currents of the incident and reflected waves satisfies Kirchhoff's law at the load impedance. Such a situation is depicted in Figure 2.5.

If the incident voltage wave at the load is denoted V_+ and the reflected wave from the load is denoted V_-, the sum of the two voltage waves satisfy

$$V_+ + V_- = V_L, \tag{2.22}$$

where V_L is the voltage at the load impedance, Z_L. Also, the sum of the currents associated with the incident and reflected waves, I_+ and I_-, respectively, satisfy

$$I_+ + I_- = I_L, \tag{2.23}$$

where I_L is the current through the load impedance Z_L. The current of the incident wave is V_+ / Z_0 and the current of the reflected wave is $-V_- / Z_0$, as discussed in subsection *A*. The current of the load is given by $I_L = V_L / Z_L = (V_+ + V_-) / Z_L$. Substituting these relations into (2.23), a relation between V_+ and V_- in terms of Z_0 and Z_L can be found from

$$\frac{V_+}{Z_0} - \frac{V_-}{Z_0} = \frac{V_+ + V_-}{Z_L}. \tag{2.24}$$

The reflection coefficient at the load is defined as the ratio of the reflected voltage wave at the load to the incident voltage wave at the load. From the above relation the reflection coefficient is

$$\Gamma_L \equiv \frac{V_-}{V_+} = \frac{Z_L - Z_0}{Z_L + Z_0}. \tag{2.25}$$

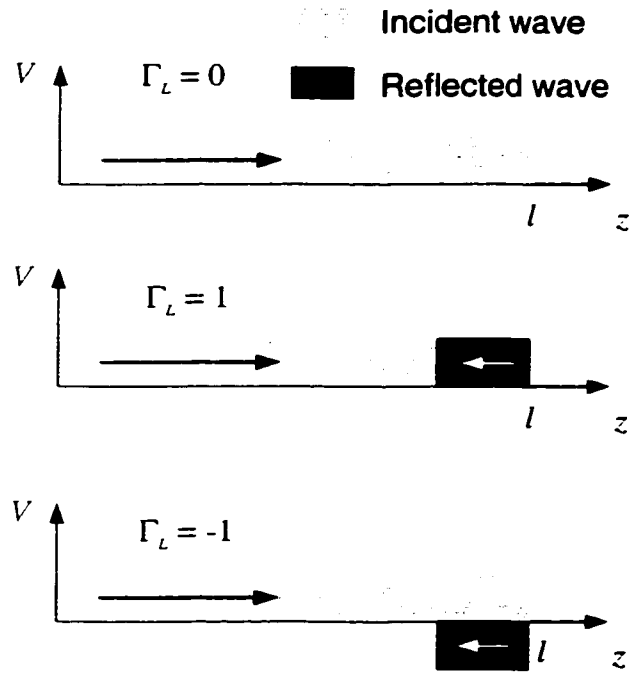
The reflection coefficient is bound between one and negative one. Note that no reflected wave exists (F_2 = 0) if the load impedance is equal to the characteristic impedance of the line since the incident wave encounters no impedance discontinuity. The power that is incident on the load is given by $W_+ = I_+ V_+ = V_+^2 / Z_0$. Similarly, the power reflected from the load is $W_- = V_-^2 / Z_0$. Thus, the portion of the incident power that is reflected from the load is given by

$$\frac{W_-}{W_+} = \left(\frac{V_-}{V_+}\right)^2 = \Gamma_L^2. \tag{2.26}$$

The portion of the incident power that is transmitted to the load is

$$\frac{W_L}{W_+} = 1 - \Gamma_L^2. \tag{2.27}$$

Note that the total incident power is transferred to the load when the load impedance is equal to the characteristic impedance of the transmission line since Γ_L is equal to zero. In this case, the load impedance is matched to the transmission line characteristic impedance. In the case of an open-circuit load, the reflection coefficient is equal to one ($\Gamma_L = 1$) and all of the power incident on the load is reflected back towards the source since an open-circuit can not consume power ($I_L = 0$). Similarly, in the case of a short-circuit load, the reflection coefficient is equal to negative one ($\Gamma_L = -1$) and no power is transmitted to the load since $V_L = 0$. The three different load cases discussed above are graphically depicted in Figure 2.6.



(a) Matched termination. No reflected wave.

(b) Open-circuit termination. Incident voltage wave totally reflected and has the same sign as the incident voltage wave.

(c) Short-circuit termination. Incident voltage wave totally reflected and has the opposite sign to the incident voltage wave.

Figure 2.6 Behavior of various transmission line terminations

The waves traveling in the negative z direction towards the source behaves in the same manner as discussed above if the source impedance Z_s is not matched to the characteristic impedance of the line. A source reflection coefficient is defined as the ratio of the reflected voltage wave from the source to the incident voltage wave on the source and is

$$\Gamma_s = \frac{Z_s - Z_0}{Z_s + Z_0}, \tag{2.28}$$

where Z_s is the impedance of the voltage source or current source driving the transmission line.

D. Transient Response of Lossless Transmission Lines

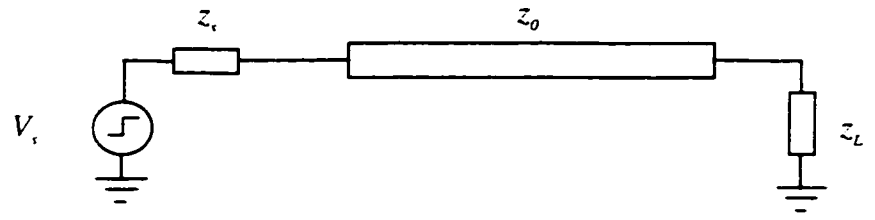


Figure 2.7 A transmission line driven by a voltage source with a source impedance Z_s and terminated by a load impedance Z_L.

Consider the transmission line shown in Figure 2.7. The voltage source driving the transmission line is a step input signal changing from zero to V_s volts at time zero and has a source resistance equal to Z_s. The transmission line has a characteristic impedance Z_0 and is terminated by a load impedance Z_L. At the beginning of the transient response (time zero), the transmission line appears as an impedance Z_0. Thus, a portion of the input signal V_s is injected into the transmission line and is

$$V_{source} = \frac{Z_0}{Z_s + Z_0} V_S. \tag{2.29}$$

This wave propagates with a speed v towards the load and reaches the load after time τ where τ is the time of flight of the signals across the transmission line and is given by l / v for a transmission line of length l. At the load a reflected signal is initiated that propagates back towards the source with speed v. The magnitude of the reflected signal depends upon the amount of mismatch between the load impedance and the characteristic impedance of the line and is

$$V_R = \frac{Z_L - Z_0}{Z_L + Z_0} V_{Source} = \Gamma_L V_{Source}. \tag{2.30}$$

The reflected wave is superimposed on the initial signal injected at the source. Thus, at that instant, there are two waves in the transmission line, one travelling in the positive z direction and the other wave traveling in the negative z direction. At time 2τ, the reflected wave reaches the source and another wave is reflected from the source impedance and travels back towards the load. The magnitude of this wave is given by

$$V_{Source1} = \frac{Z_s - Z_0}{Z_s + Z_0} V_R = \Gamma_S V_R. \tag{2.31}$$

This wave is superimposed on the two previous waves. When the wave reaches the load, another wave is initiated that travels back towards the source. This process of reflections continues forever except for the special cases where the source impedance or the load impedance is exactly equal to Z_0 (the matched condition). The magnitude of each reflection is smaller than the incident wave that caused the previous reflection because the reflection coefficient has a magnitude that is always less than one (except for the open circuit and short circuit termination conditions). Thus, the magnitude of the reflections is attenuated with time until the reflections become negligible and a steady state voltage is reached. The time required to reach the steady state condition depends upon the relative magnitude of the source and load impedances as compared to the characteristic impedance. For example, if the load impedance is equal to Z_0, the transient response terminates after time τ since the reflection at the load in this case is zero. If the load impedance is not equal to Z_0 and the source impedance is equal to Z_0,

the transient response ends after time 2τ. Alternatively, if the load impedance is an ideal open-circuit and the source impedance is an ideal short-circuit, the voltage across the transmission line oscillates forever since the reflection coefficient at the load is one and the reflection coefficient at the source is negative one.

The dynamics of the transient response is best illustrated with an example [1]. The input voltage V_s is equal to 5 volts in the circuit shown in Figure 2.7 and the load impedance is an open-circuit. Thus, the voltage reflection coefficient at the load is one. The time of flight of the signals across the transmission line τ is equal to 1.5 ns. Three different cases are considered when $Z_s = 5Z_o$, $Z_s = Z_o$, and $Z_s = Z_o/5$.

1. Large source impedance (*e.g.*, $Z_s = 5Z_o$)

According to the previous discussion, an initial voltage wave is injected into the transmission line at time zero and has the value 0.83 volts according to (2.29). When this voltage reaches the load, the wave is totally reflected at the open-circuited load and the voltage at the load reaches 1.66 volts, which is the sum of the incident and reflected waves. The reflected wave (with magnitude 0.83 volts) reaches the source after time 2τ and a new reflected wave travels back towards the load with a magnitude of 2/3*0.83 since the source voltage reflection coefficient is 2/3 according to (2.31). A graphical method to evaluate the reflections is called a *lattice diagram* [70]-[72]. The lattice diagram for this example is shown in Figure 2.8. The time response for this example is shown in Figure 2.9(a). This case is called the

underdriven, overdamped, or sluggish response and the signal requires several reflections of small magnitude to reach a steady-state response.

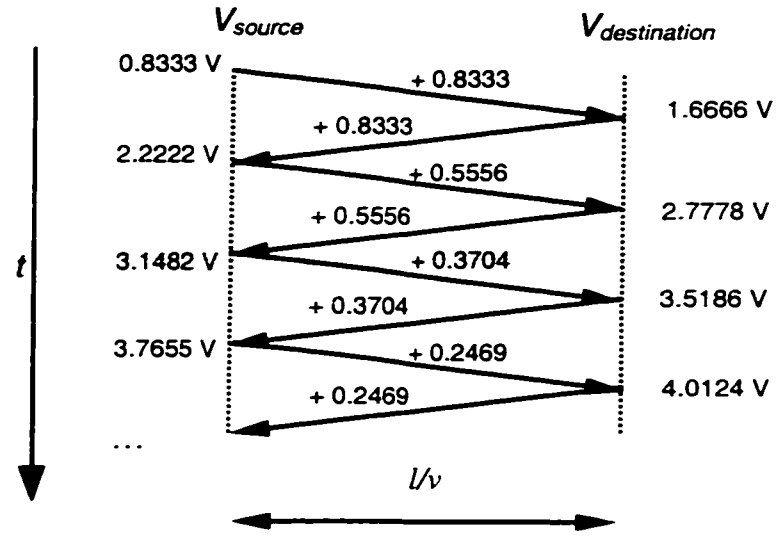


Figure 2.8 Lattice diagram for $Z_s = 5Z_0$ and $Z_L = \infty$. The input voltage is 5 volts.

2. Matched source impedance ($Z_S = Z_0$)

Half the input signal is injected at the source. The reflection at the load end doubles the signal, so the final value is reached immediately at the load. The reflected signal of half the input voltage reaches the source end after time 2τ and adds to the original signal and the transient response terminates since there is no reflection at the source. This system is the most efficient load condition to transmit signals across a transmission line in the case of an open-circuit load and is called the series matched case since the matching is at the source. Matching at the load is termed shunt matching. The transient response for this matched source impedance case is shown in Figure 2.9(b).

3. Small source impedance (*e.g.*, $Z_s = Z_o/5$)

A large portion of the input voltage (4.166 volts) is injected into the transmission line in this case of a small source impedance. The value of the injected voltage is doubled at the load which causes severe overshoots. The reflection coefficient at the source is negative (-2/3) and the signal changes phase when the signal reaches the source impedance. Thus, the signal oscillates and several reflections occur before a steady state voltage is reached. This case is called the overdriven or the underdamped case. The transient response for this case is shown in Figure 2.9(c).

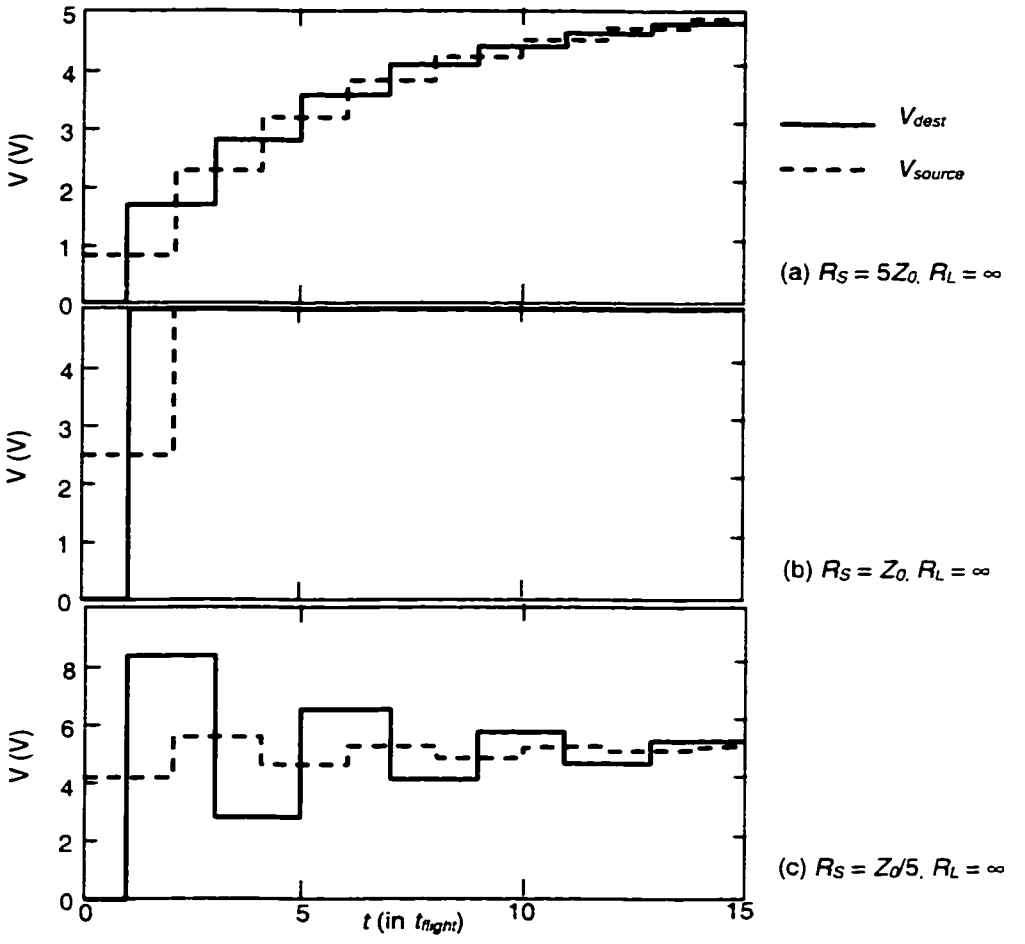


Figure 2.9 Transient response of a lossless transmission line with an open-circuit load and a step input. a) Large source impedance as compared to the characteristic impedance. b) Matched source impedance. c) Small source impedance as compared to the characteristic impedance.

2.1.2 Lossy *RLC* Transmission Lines

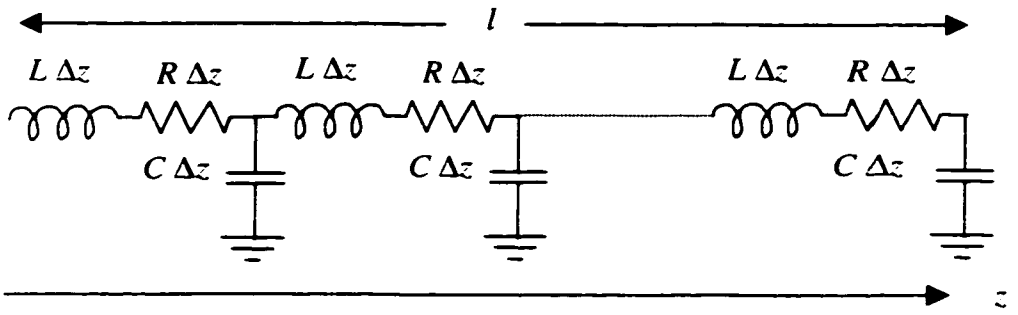


Figure 2.10 *RLC* transmission line.

A lossy *RLC* transmission line is shown in Figure 2.10. For an *RLC* transmission line driven by a sinusoidal input $Re\{e^{j\omega t}\}$, the voltage and current across the transmission line are [70]-[72]

$$V(z,t) = \mathrm{Re}\{V_1 e^{(j\omega t - \gamma z)} + V_2 e^{(j\omega t + \gamma z)}\}, \tag{2.32}$$

$$I(z,t) = \mathrm{Re}\{\frac{V_1}{Z_0} e^{(j\omega t - \gamma z)} - \frac{V_2}{Z_0} e^{(j\omega t + \gamma z)}\}. \tag{2.33}$$

The solution of $V(z,t)$ is the sum of two waves, one wave traveling in the positive z direction and the other wave traveling in the negative z direction. V_1 is the summation of the original voltage wave and all of the reflected voltage waves at the source traveling in the positive z direction. V_2 is the summation of all of the reflected voltage waves traveling in the negative z direction. The propagation constant of the transmission line γ describes the characteristics of the wave propagation across the line. For an *RLC* transmission line, the propagation constant is complex and is

$$\gamma = \alpha + j\beta, \tag{2.34}$$

where the real part α is the attenuation constant of the waves as the waves propagate across the line, and the imaginary part β is the phase constant which determines the speed of propagation of the waves across the line. Substituting (2.34) into (2.32), the real part of the voltage is given by

$$V(z,t) = V_1 e^{-\alpha z}\cos(\omega t - \beta z) + V_2 e^{\alpha z}\cos(\omega t + \beta z). \tag{2.35}$$

The attenuation of a traveling wave is exponentially dependent on the distance traveled by the wave. The speed of propagation of the wave across the line is

$$v = \frac{\omega}{\beta}. \tag{2.36}$$

The propagation constant of an *RLC* line is

$$\gamma = j\omega\sqrt{LC}\sqrt{1+\frac{R}{j\omega L}}. \tag{2.37}$$

The attenuation constant and phase constant are the real and imaginary parts of γ, respectively. Note that if the resistance of the line is negligible ($R = 0$), γ is imaginary. Thus, the attenuation constant is equal to zero and the signals do not attenuate as the signals travel across the transmission line. Also, the speed of the signals is frequency independent and the signals travel at a velocity described by (2.8). This case is that of a lossless transmission line. However, in the case of a lossy transmission line, two major differences occur. The first difference is that the waves attenuate as the signals travel along the line and the attenuation is frequency dependent. The second is that the speed of the waves is frequency dependent. The frequency dependence causes the frequency components of a signal to see different conditions across the transmission line and travel at different speeds. Thus, the shape of a signal degrades as the signal

travels across a lossy transmission line and the line is called a *dispersive* transmission line. Identical square pulse signals that are injected at the inputs of infinitely long lossless and lossy transmission lines are shown in Figure 2.11. The signals are observed after the signals travel some distance across the transmission lines. In the case of a lossless transmission line, the signal maintains the original shape and magnitude as the signals propagate across the line. In the case of a lossy transmission line, the signals attenuate and become wider due to the loss of the high frequency components at the edges of the waveform. Thus, the resistance of an *RLC* transmission line degrades the rise and fall times of the input signal as the signal propagates along the lossy line [70]-[72].

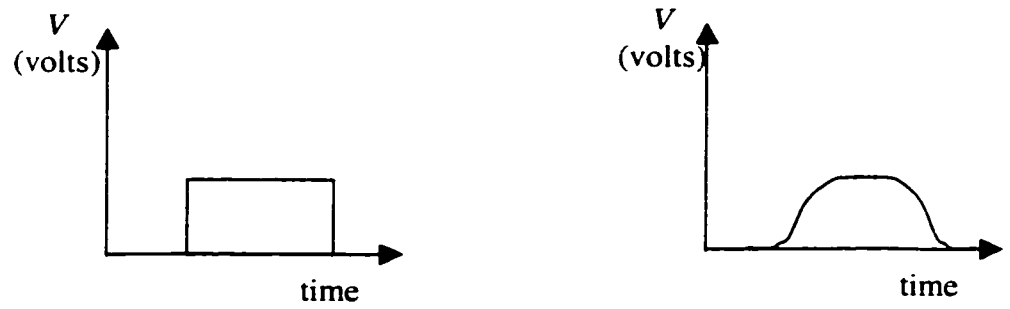


Figure 2.11 Signal dispersion of a square wave signal in lossy transmission lines. a) Pulse shape after traveling along a lossless transmission line. b) Pulse shape after traveling along a lossy transmission line.

The characteristic impedance of a lossy transmission line is also complex and is given by

$$Z_0 = \sqrt{\frac{L}{C}}\sqrt{1+\frac{R}{j\omega L}}. \tag{2.38}$$

Note that the characteristic impedance of a lossy transmission line is the same as the characteristic impedance of a lossless transmission line when $R = 0$. The transfer function at a distance d from the load of a lossless transmission line with a source impedance Z_s, a load impedance Z_L, and length l can be derived from the basic concepts discussed in this section and is

$$\frac{V_{out}(s,d)}{V_{in}(s,d)} = \frac{\left[\left(1-\frac{z_0}{z_L}\right)e^{\gamma l}+\left(1+\frac{z_0}{z_L}\right)e^{-\gamma l}\right]}{\left(1+\frac{z_s}{z_0}\right)\left(1+\frac{z_0}{z_L}\right)e^{\gamma l}+\left(1-\frac{z_s}{z_0}\right)\left(1-\frac{z_0}{z_L}\right)e^{-\gamma l}}, \tag{2.39}$$

where Z_0, Z_S, Z_L, and γ are functions of s and $s = j\omega$.

2.1.3 *RC* Transmission Lines

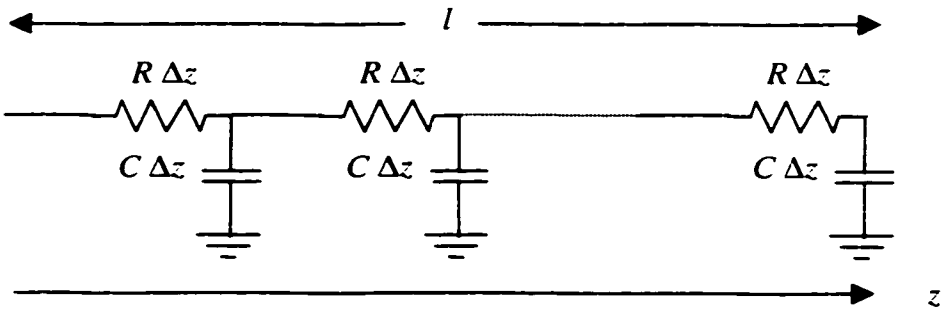


Figure 2.12 An *RC* transmission line.

The *RC* transmission line shown in Figure 2.12 has a voltage that varies with the position z and time t and is characterized by the differential equation,

$$RC\frac{\partial V}{\partial t}=\frac{\partial^2 V}{\partial z^2}. \tag{2.40}$$

This equation is the well-known diffusion equation [1], [40], [70]-[72]. The signals travel across an *RC* transmission line purely by diffusion. For a lossless transmission line, it has been shown that the signals travel with constant speed across the line. In that case, the signals are said to travel in the propagation mode. Signals in an *RLC* transmission line travel partially in the propagation mode and partially in the diffusion mode. The degree to which the signal behavior is affected by each mode depends upon the amount of inductance effects present in the line.

The diffusion equation given by (2.40) has no closed-form solution [1], [40], [70]-[72]. However, it is common to approximate the behavior of an *RC* transmission line by a one dominant time constant system given by

$$\tau(V_{out})=\frac{RCl^2}{2}. \tag{2.41}$$

Note the quadratic dependence of the dominant time constant on the line length l which means that the propagation delay and the rise time are also quadratically dependent on the line length. The propagation delay and rise time of a lossless transmission line are linearly dependent on the line length since the propagation delay and rise time are multiples of the time of flight across the transmission line which is linearly dependent on the line length.

2.2 Approximate Models for *RC* Interconnect

In general, there are two broad categories of simulation tools that can be used to analyze the timing behavior of a VLSI circuit. The first category is static circuit simulators [124]-[126], which uses analytical models for the devices and interconnect to reach an approximate solution for the delays and transition times of the signals propagating within a VLSI circuit. The advantage of such simulators is high computational speed which is necessary when analyzing high complexity VLSI circuits in a reasonable amount of time. Also, static simulators are numerically reliable and have low memory requirements. However, the loss of accuracy and the loss of waveform details such as spikes and overshoots are major disadvantages of static simulators.

The other category of simulators is dynamic circuit simulators, *e.g.*, [127], [128], which use a nodal matrix description of the input circuit starting from a set of given initial conditions and solve the differential equations describing the circuit to determine accurate waveforms for the signals within a VLSI circuit. Dynamic simulators use discrete numerical methods to solve the differential equations by dividing the simulation time into individual time steps. At each time step, numerical matrix methods are employed to determine the timing solution. Thus, dynamic simulators are much slower and require significantly more memory than static simulators but are also significantly more accurate than static simulators. Dynamic simulators are usually used to analyze the worst case paths of a VLSI circuit in order to accurately characterize the behavior of these critical paths.

The circuits simulated by static or dynamic simulators include interconnect, therefore, accurate interconnect models are required. However, the interconnect model used in each type of simulator is quite different. Analytical models for the propagation delay and rise time are used in static simulators while differential equations are used in dynamic simulators. In the case of static simulators, *RC* interconnect cannot be represented exactly since, as previously mentioned, no closed form solution for the diffusion equation that describes an *RC* transmission exists. Therefore, approximate analytical solutions are necessary. For dynamic simulators, one of two cases exists. If the simulator does not internally support a distributed *RC* model (such as in SPICE [127]), approximate lumped circuit elements (resistances and capacitors) are used to replace exact transmission line models. For other dynamic simulators that internally include models for distributed *RC* transmission lines such as AS/X [128], an exact representation of the *RC* interconnect is possible. However, even if exact transmission line models are supported by a dynamic simulator, the use of approximate models for the interconnect with discrete circuit elements may still be desirable since the use of these models permit the simulator to operate faster and more reliably. Only when high accuracy is required would the use of exact transmission line models be justified from a computational efficiency perspective.

For a dynamic simulator, the simplest model to represent an *RC* interconnect is a lumped *RC* circuit with one resistor equal to the total resistance of the line R_t and one capacitor equal to the total capacitance of the line C_t as shown in Figure 2.13(a). This model suffers high error since the lumped *RC* model does not accurately model the distributed nature of the interconnect. A lumped *RC* network to model a

distributed *RC* net can cause up to 50% error. For example, in the distributed *RC* line shown in Figure 2.12, only a portion of the capacitance of the line charges (or discharges) through the entire resistance of the line. Capacitors at the beginning of the line charge through a small resistance, capacitors at the middle of the line charge through half the resistance of the line, while capacitors at the end charge through all of the resistance of the line. On average, the capacitance of the line sees about half the resistance of the line. Thus, more accurate models to represent the line are shown in Figure 2.13(b) and Figure 2.13(c) [22], [23]. The interconnect models shown in Figure 2.13(b) and Figure 2.13 (c) are called π and T models, respectively, due to the shape of the *RC* impedance network [23]. These models are much more accurate than a lumped model. For even greater accuracy, multiple π and T models can be used by subdividing the line into smaller sections and representing each section by a π or a T model as shown in Figure 2.13(d) through Figure 2.13(h) [22], [23]. Another option is to model the interconnect by several ladder sections as shown in Figure 2.12. However, multiple π and T models are much more accurate than a ladder model for the same number of discrete circuit elements (or the same number of circuit nodes), directly affecting the computational performance of the dynamic simulator. For example, replacing an *RC* transmission line by a three section π model causes a discrepancy of less than 3% in the waveform shape as compared to an exact transmission line model [23]. Tens of sections would be required for a ladder model to achieve the same level of accuracy [22].

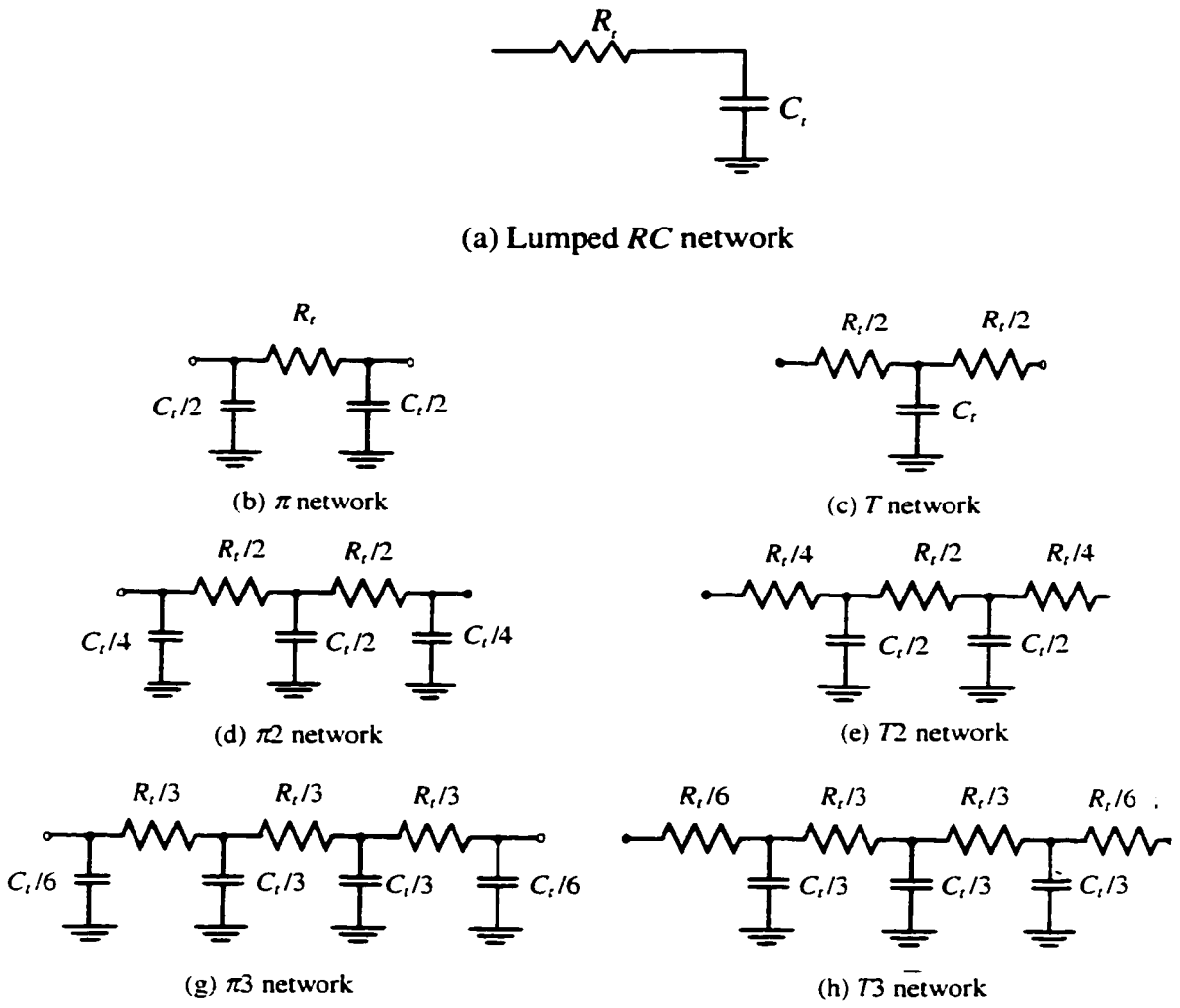


Figure 2.13 Discrete element circuit representation of an *RC* interconnect.

Analytical solutions for the propagation delay and rise time of *RC* interconnects interacting with gates are necessary for static simulators as well as for a variety of VLSI design methodologies. One such technique is the process of inserting repeaters to subdivide a long *RC* line into shorter sections in order to minimize the overall propagation delay as is discussed later in this chapter. A useful configuration to consider is that of a CMOS gate driving another CMOS gate with an interconnect

line between the two gates as shown in Figure 2.14 since any path in a VLSI circuit can be subdivided into several such configurations (where extra fanout is treated as a lumped capacitance).

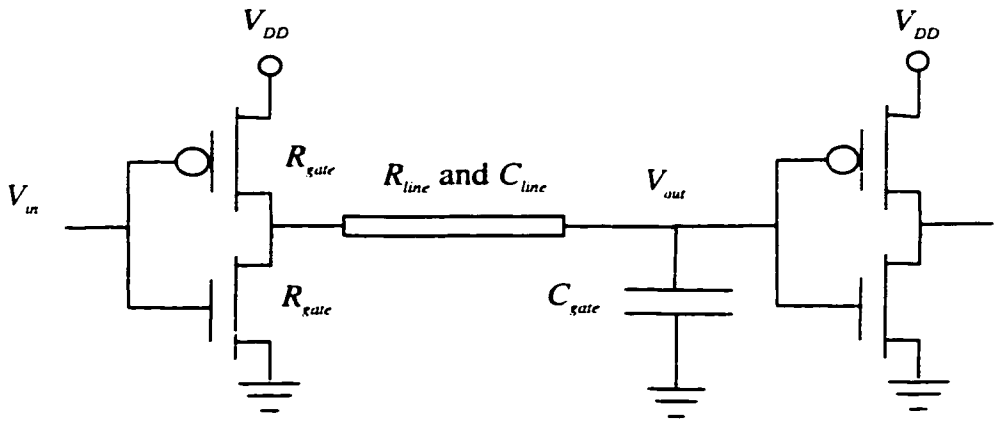


Figure 2.14 A CMOS gate driving another CMOS gate with a resistive-capacitive interconnect wire connecting the two inverters.

The driving gate is customarily approximated by an equivalent resistor, R_{tr}. The driven gate is modeled by an input capacitance, C_L. The interconnect is replaced by one of the approximate models shown in Figure 2.13. The simplest model to represent the interconnect is a lumped model as shown in Figure 2.13(a). The resulting equivalent circuit for the driving gate, the driven gate, and the lumped interconnect model is shown in Figure 2.15(a). For this simple circuit, the output response for a unit step input ($v_{in}(t) = 0$ for $t < 0$ and $v_{in}(t) = 1$ for $t \geq 0$) is

$$V_{out}(t) = 1 - \exp(-t/\tau_{RC}), \tag{2.42}$$

where τ_{RC} is the time constant of the circuit and is

$$\tau_{RC} = (R_{tr} + R_t)(C_t + C_L)\,. \tag{2.43}$$

The 50% propagation delay t_{pd} and the 10%-to-90% rise time t_r of such a signal are, respectively,

$$t_{pd} = 0.693\tau_{RC} = 0.693(R_{tr} + R_t)(C_t + C_L)\,, \tag{2.44}$$

$$t_r = 2.2\tau_{RC} = 2.2(R_{tr} + R_t)(C_t + C_L)\,. \tag{2.45}$$

These expressions, however, suffer large error due to the inaccuracies of the lumped interconnect model, as discussed previously. Another approach has been introduced by Sakurai in [23] which shows that an *RC* transmission line with a source resistor and a load capacitance at the terminations has a transfer function with an infinite number of poles. However, a single dominant pole can be used to approximate the transfer function with high accuracy. This single pole is given by $1/\tau'_{RC}$, where

$$\tau'_{RC} = R_{tr}(C_t + C_L) + R_tC_L + 0.5R_tC_t\,. \tag{2.46}$$

This single pole system results in a unit step response of

$$V_{out}(t) = 1 - \exp(-t/\tau'_{RC})\,, \tag{2.47}$$

which is the same expression as the response in (2.42) except for the different time constant. Thus, the propagation delay and rise time of this response are [23], [40]

$$t_{pd} = 0.693\tau'_{RC} = 0.693R_{tr}(C_t + C_L) + 0.693R_tC_L + 0.35R_tC_t\,, \tag{2.48}$$

$$t_r = 2.2\tau'_{RC} = 2.2R_{tr}(C_t + C_L) + 2.2R_tC_L + 1.1R_tC_t\,. \tag{2.49}$$

These closed form expressions are quite accurate and are within 4% of SPICE [23]. The time constant in (2.46) can be interpreted by referring to Figure 2.15(b) where the interconnect is replaced by a more accurate π model as compared to a simple lumped model. The time constant τ'_{RC} has four terms, each term of which has a one-to-one

correspondence to Figure 2.15(b). The first two terms in (2.46) have a factor of one since the entire interconnect and load capacitances see the transistor resistance. The third term has a factor of one since the load capacitance sees the entire interconnect resistance. The fourth term has a factor of 0.5 since on average, the interconnect capacitance sees half the interconnect resistance.

Another interesting concept introduced by Sakurai in [23] is that not every interconnect line needs to be represented by the most accurate model since the more accurate model requires additional simulation time. Only those nets that significantly affect the accuracy of the simulation should be replaced by a more accurate impedance model. A criterion for determining which interconnects require more accurate multiple section π or T models depends upon the ratio of the device parasitic impedances (R_{tr} and C_L) to the interconnect parasitic impedances (R_t and C_t). If the device parasitic impedances are much larger than the interconnect parasitic impedances, the response is dominated by the drive and load gates, which renders the interconnect model unimportant. In this case, the simplest interconnect model should be used (a short-circuit). Alternatively, for cases where the interconnect parasitic impedances are large as compared to the gate parasitic impedances, highly accurate multiple section π or T models should be used. Two variables have been introduced in [23] to characterize the importance of the accurate interconnect models. These variables are

$$R_T = \frac{R_{tr}}{R_t}, \tag{2.50}$$

$$C_T = \frac{C_L}{C_t}. \tag{2.51}$$

When R_T is high, the gate resistance dominates the interconnect resistance. When C_T is high, the load capacitance dominates the interconnect capacitance. The appropriate interconnect model is therefore a function of R_T and C_T. The worst case error exhibited by using an inaccurate interconnect model occurs when R_T and C_T are equal to zero. A lumped *RC* model as compared to a distributed *RC* model is shown in Figure 2.16 where R_T and C_T are equal to zero.

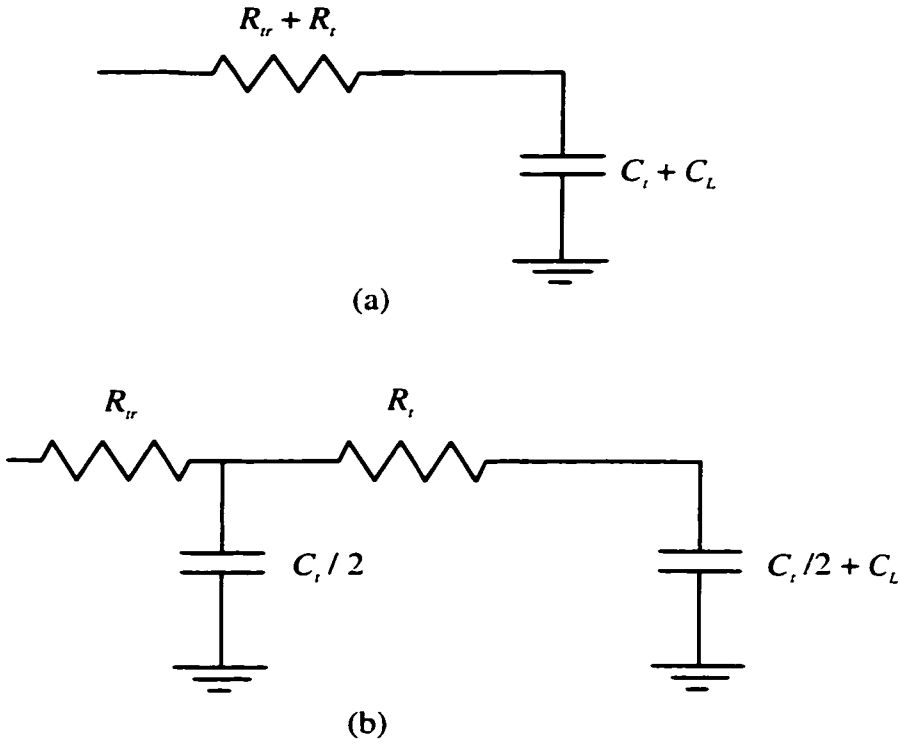


Figure 2.15 Approximate discrete element linear circuits of a CMOS gate driving another CMOS gate with an interconnect line. a) The interconnect is replaced by a lumped model. b) The interconnect is replaced by a single π section.

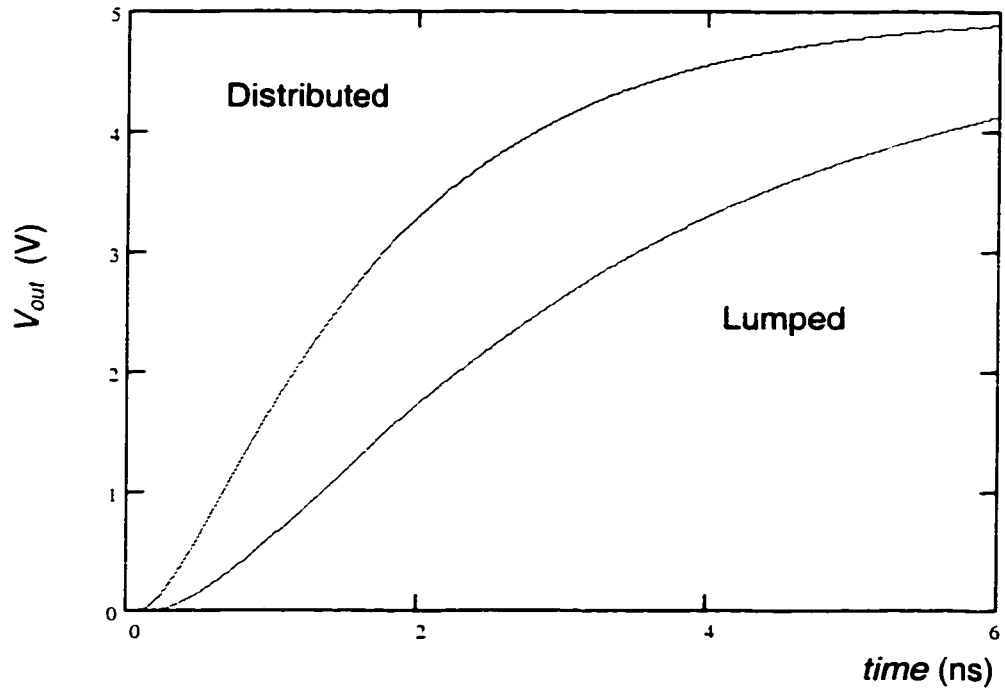


Figure 2.16 Simulated step response of lumped and distributed *RC* interconnect when $R_T = 0$ and $C_T = 0$.

2.3 Repeater Insertion in *RC* Lines

The dependence of the propagation delay on the length of an *RC* line is quadratic as discussed in section 2.2. This behavior of *RC* lines means that as the line gets longer, the propagation delay increases dramatically. Thus, it is advantageous to partition the line into shorter sections by inserting buffers within the line to reduce the total propagation delay from the beginning of the line to the end of the line [37]-[44]. The shorter the sections of the line, the greater the interconnect delay is reduced. However, inserted buffers increase the gate delay. Thus, there is an optimum number

of repeaters that should be inserted into an *RC* line to minimize the overall propagation delay. This trend is qualitatively described in Figure 2.17.

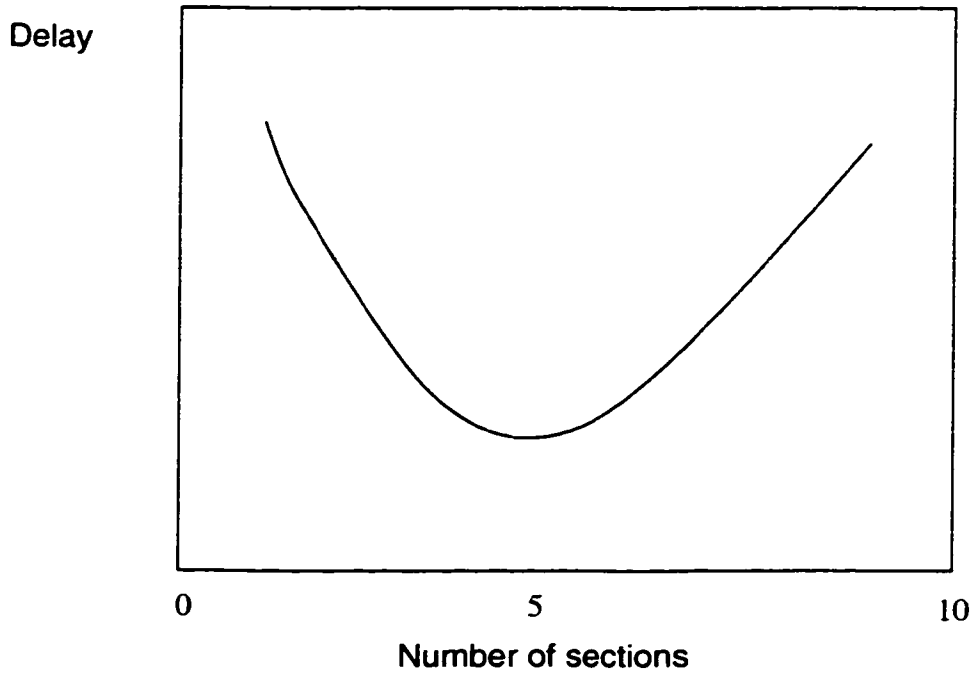


Figure 2.17 Relation between the number of sections an *RC* line is subdivided into and the total propagation delay.

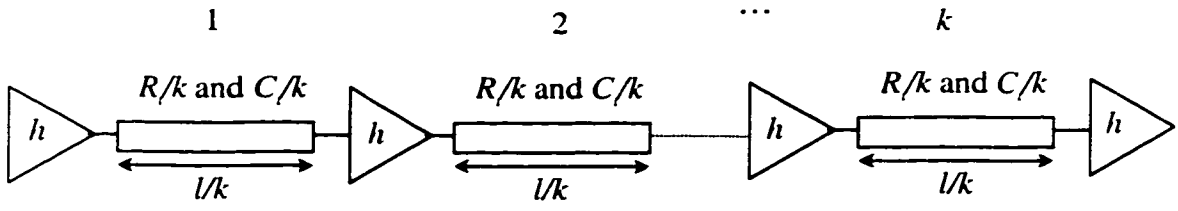


Figure 2.18 Repeaters inserted along an *RC* line.

Referring to Figure 2.18, an *RC* line can be subdivided into k sections by inserting k buffers within the line [39], [40]. Each buffer is h times wider than the

minimum size buffer permitted by the technology. The output resistance of each buffer is therefore R_0/h and the input capacitance is C_0h, where R_0 and C_0 are the output resistance and the input capacitance of a minimum size buffer, respectively. Each buffer drives an interconnect section with a total resistance of R_t/k and a total interconnect capacitance of C_t/k. The total propagation delay of the repeater system (the interconnect line and the repeaters) is the summation of the delays of each of the k sections of the interconnect. Since each of the sections are similar, the total propagation delay of the repeater system is k times the propagation delay of a single section. Using the formula in (2.48) for the propagation delay of a single section, the total propagation delay of a k section repeater system, as introduced by Bakoglu in [39], [40], is

$$t_{pdtotal} = k\left[0.693\frac{R_0}{h}\left(\frac{C_t}{k} + hC_0\right) + 0.693\frac{R_t}{k}hC_0 + 0.35\frac{R_t}{k}\frac{C_t}{k}\right]. \tag{2.52}$$

If the partial derivatives of the above expression with respect to h and k are equated to zero, the optimum values of h and k to minimize $t_{pdtotal}$ are

$$h = \sqrt{\frac{R_0C_t}{R_tC_0}}, \tag{2.53}$$

$$k = \sqrt{\frac{R_tC_t}{2R_0C_0}}. \tag{2.54}$$

Both of these equations can be interpreted intuitively. The optimum number of sections k depends upon the ratio of the interconnect delay R_tC_t to the gate delay R_0C_0. The larger the interconnect delay with respect to the gate delay, the more buffers that should be inserted to reduce the interconnect delay since the increase in propagation delay due to inserting more buffers is less significant. The optimum size of the

buffers h is chosen such that the output resistance of the buffers is close to the interconnect resistance of each section. This characteristic can be observed by dividing the buffer resistance (R_0/h) by the resistance of a single section (R/k). The ratio of the buffer resistance to the resistance of one section is 0.707 when using the values of h and k in (2.53) and (2.54), respectively. This behavior is understandable since if the output resistance of the buffers is much larger than the resistance of a single interconnect section, the gate delay dominates the interconnect delay and the delay can be further reduced by decreasing the channel resistance. Alternatively, if the output resistance of the buffers is much smaller than the resistance of a single interconnect section, the interconnect delay dominates the gate delay. It is therefore appropriate to further subdivide the line into more sections in order to reduce the interconnect delay, thereby reducing the interconnect resistance of each section. Thus, the optimum condition occurs at the point where the output resistance of each gate is comparable to the resistance of the interconnect section being driven.

Chapter 3 Evaluating the Transient Response of Linear Networks

The underlying concepts of the most commonly used methods to evaluate the transient response of linear networks in *VLSI* circuits are briefly reviewed in this chapter. Simple methods to calculate the delay of signals in an *RC* tree such as the Elmore delay [73] and Wyatt's approximation [74] are introduced in section 3.1 since extensions to these methods for *RLC* trees are presented in subsequent chapters. The calculation of more accurate transient responses of *RLC* interconnects via moment matching is discussed in section 3.2 [75]-[107].

3.1 Elmore Delay and Wyatt Approximation

Methods for calculating the propagation delay of a multi-pole system for a step input given the transfer function of the system are introduced in this section. These methods do not require the time domain signals of the given system transfer function to be known in order to numerically solve for the 50% point of the time domain signal. Furthermore, these approximations do not require complete knowledge of the transfer function but require only the first moment of the transfer function which, as is shown in this section, can be efficiently calculated for the most common *RC* interconnect structures (single lines and trees). Closed form solutions of the propagation delay of signals at different nodes of an *RC* tree are also discussed in this section.

3.1.1 Elmore Delay

In 1948, Elmore [73] introduced a general approach for calculating the propagation delay of a linear system given the transfer function of the system. If the transfer function of the system is $H(s)$, the normalized transfer function $H_n(s)$ is $H(s)/H(0)$, which can be generally described as

$$H_n(s) = \frac{1 + a_1 s + a_2 s^2 + \ldots + a_n s^n}{1 + b_1 s + b_2 s^2 + \ldots + b_m s^m}, \tag{3.1}$$

where a_i and b_i are real and $m > n$. For a monotone response, all of the poles of $H_n(s)$ should be real and for a stable system all of the poles should lie on the negative real axis. The unit step response of the normalized transfer function is $(1/s)\text{-}H_n(s)$. In the time domain, the transient unit step response $e(t)$ has a final value of one and is monotonically increasing as shown in Figure 3.1.

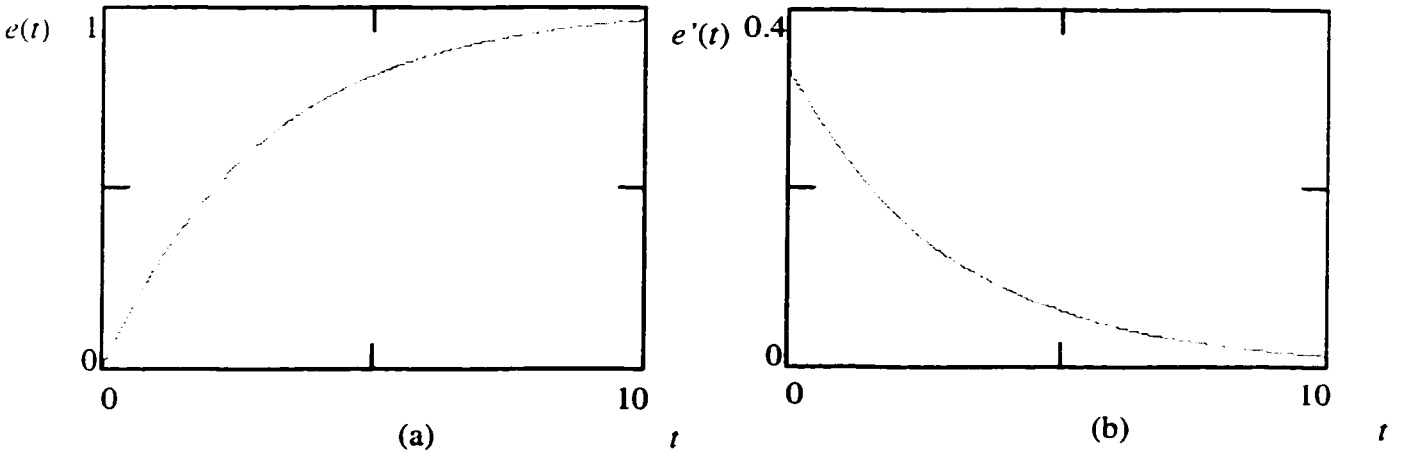


Figure 3.1 Step and impulse responses of a normalized monotone transfer function. (a) Step response. (b) Impulse response (which equals the time derivative of the step response).

Elmore proceeded from the observation that the time domain unit step response has the characteristics of the integral of a probability function since the response has a final value of one and is monotonically increasing which makes the area under $e'(t)$ equal to one and $e'(t)$ always positive. Thus, Elmore defined the 50% propagation delay (the time where $e(t)$ is equal to 0.5) as

$$T_D = \int_0^\infty te'(t)dt, \tag{3.2}$$

which is the centroid of the area under $e'(t)$. By noting that $e'(t)$ for a step input is simply $h_n(t)$, the transfer function $H_n(s)$ can be expressed as

$$H_n(s) = \int_0^\infty e'(t)e^{-st}dt = 1 - s\int_0^\infty te'(t)dt + \frac{s^2}{2!}\int_0^\infty t^2e'(t)dt - \ldots . \tag{3.3}$$

Thus, if the normalized transfer function is expanded in the powers of s, the 50% delay can be determined directly as the coefficient of s. From (3.1), the propagation delay is $T_D = b_1 - a_1$, which is the definition of the Elmore delay. The Elmore delay as defined by (3.3) is also described as the first moment of the transfer function [73].

3.1.2 Wyatt Approximation

In 1987, Wyatt [74] used the relationships that b_1 and a_1 are given by

$$b_1 = \sum_{i=1}^{m} \frac{1}{p_i}, \tag{3.4}$$

$$a_1 = \sum_{i=1}^{n} \frac{1}{z_i}, \tag{3.5}$$

respectively, where p_i and z_i are the poles and zeros of the transfer function, respectively. Thus, Wyatt treated $T_D = b_1 - a_1$ as the reciprocal of the dominant pole (the pole that has the smallest absolute value) of the system. This approximation is accurate for systems that can be modeled by a single dominant pole and with no low frequency zeros near the dominant pole. Applying this approximation, the unit step response of the system is

$$e(t) = 1 - \exp(-\frac{t}{T_D}), \tag{3.6}$$

which indicates a 50% propagation delay equal to $0.693T_D$ rather than T_D as anticipated by Elmore. For example, the simple *RC* circuit shown in Figure 3.2 has the transfer function,

$$H(s) = \frac{1}{sRC + 1}. \tag{3.7}$$

Thus, according to Elmore the propagation delay is *RC* and according to Wyatt the propagation delay is $0.693RC$. Note that Wyatt's solution is exact for this simple circuit and a step input signal. In general, Wyatt's solution is more accurate than Elmore's solution. However, the Wyatt approximation is often incorrectly referred to as the Elmore delay. Note also that the Elmore delay overestimates the propagation delay and is therefore used as a pessimistic (or a conservative) delay estimate.

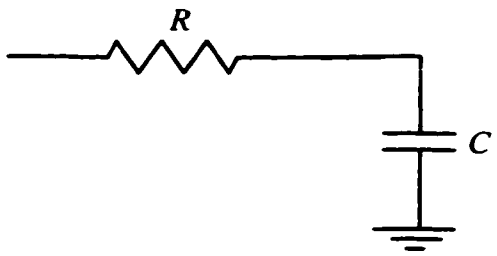


Figure 3.2. Simple *RC* circuit.

3.1.3 Calculating the Elmore Constant for *RC* Trees

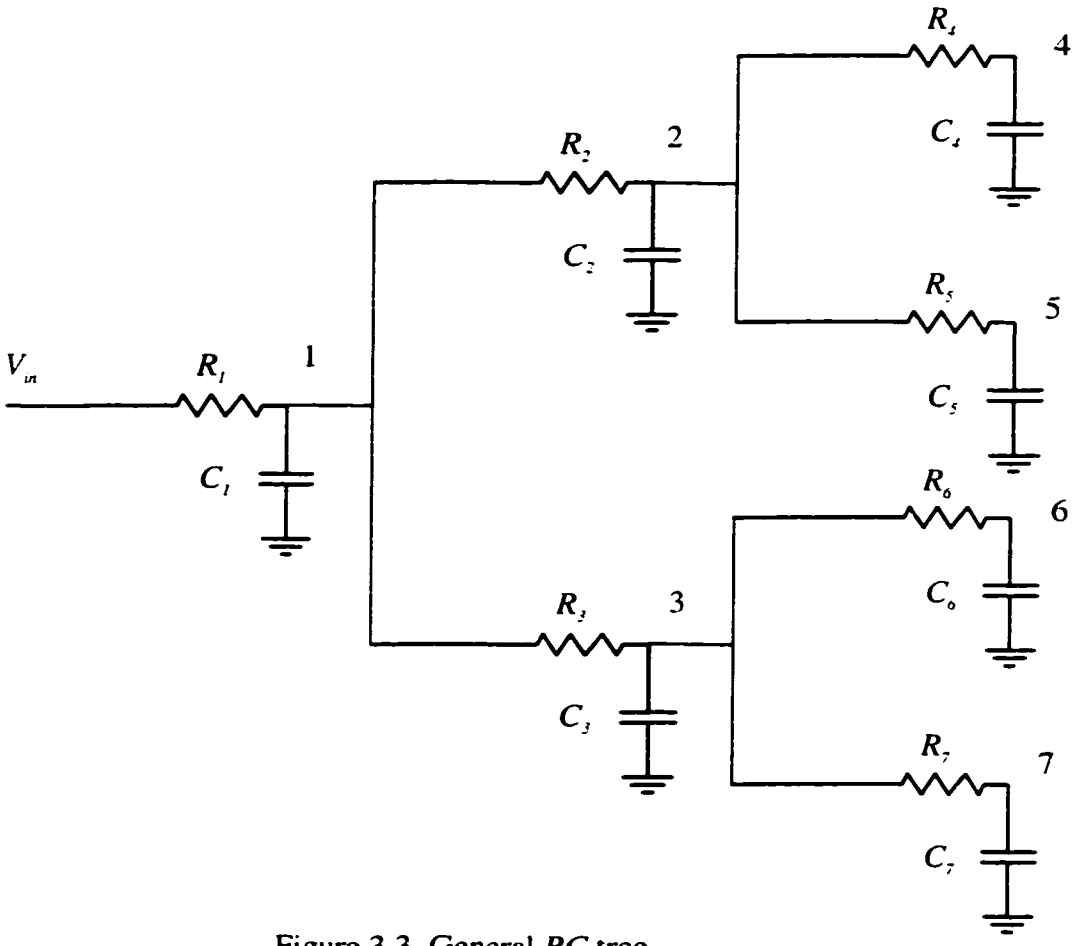


Figure 3.3. General *RC* tree.

For the general *RC* tree shown in Figure 3.3, if a unit step input is applied to the tree, the voltage drop at any node *i*, $v_i(t)$, as compared to the input voltage is [35]

$$1 - v_i(t) = \sum_k C_k R_{ik} \frac{dv_k(t)}{dt}, \tag{3.8}$$

where *k* is an index that covers every capacitor in the circuit and R_{ik} is the common resistance from the input to the nodes *i* and *k*. For example, for the *RC* tree shown in Figure 3.3, $R_{77} = R_1+R_3+R_7$, $R_{73} = R_{76} = R_1+R_3$, and $R_{71} = R_{72} = R_{74} = R_{75} = R_1$. $v_k(t)$ is the voltage across the capacitor at node *k* as a function of time. This expression can be further understood by noting that the current drained by the capacitor at node *k* is given by $C_k dv_k(t)/dt$ and that this current flows through all of the resistors in the path from the input to node *k*. Thus, a capacitor at node *k* contributes a voltage drop at node *i* as compared to the input voltage equal to the current drained by the capacitor multiplied by the common resistance from the input to the nodes *i* and *k*. This voltage drop is given by $R_{ik}(C_k dv_k(t)/dt)$. The total voltage drop at node *i* relative to the input voltage is found as the superposition of the voltage drops due to all of the capacitors in the tree as given by (3.8). Note also that the voltage at node *i* relative to the input voltage is 1 - $v_i(t)$ for $t > 0$ since the input voltage is a unit step.

If the impulse response at node *i* of a tree is $h_i(t)$, the Elmore delay (or time constant as treated by Wyatt) is shown in section 3.1.1 to be

$$T_D = \int_0^\infty t h_i(t) dt, \tag{3.9}$$

which is also described as the first moment of the transfer function. Note that if $h_i(t)$ is the normalized unit impulse response at node i, the unit step response of this normalized impulse response $v_i(t)$ is given by

$$v_i(t) = \int_0^t h_i(t)dt \,. \tag{3.10}$$

This relation can be alternatively described as $h_i(t) = v_i'(t)$. Note also that $v_i(t)$ has a final value of one when $t \rightarrow \infty$ due to the normalization of the transfer function. Equation (3.8) is integrated from 0 to t and the resulting expression is

$$\int_0^t [1 - v_i(t')]dt' = \sum_k C_k R_{ik} v_k(t) \,. \tag{3.11}$$

Integrating the left hand side above by parts with $t = \infty$,

$$\int_0^\infty [1 - v_i(t)]\,dt = t[1 - v_i(t)]\Big|_0^\infty - \int_0^\infty t v_i'(t)dt \,. \tag{3.12}$$

Using the steady state value of $v_i(\infty) = 1$, the first term in (3.12) is equal to zero after substituting this limit. Thus, (3.12) simplifies to

$$\int_0^\infty [1 - v_i(t)]dt = \int_0^\infty t v_i'(t)dt = \int_0^\infty t h_i(t)dt = T_{Di} \,, \tag{3.13}$$

since $h_i(t) = v_i'(t)$. Substituting $t = \infty$ into the right hand side of (3.11) and using the relation $v_k(\infty) = 1$, the Elmore delay at node i of an RC tree is given by [35]

$$T_{Di} = \sum_k C_k R_{ik} \,. \tag{3.14}$$

For example, the Elmore delay at node 7 in Figure 3.3 is given by

$$T_{D7} = (C_1 + C_2 + C_4 + C_5)R_1 + (C_3 + C_6)(R_1 + R_3) + C_7(R_1 + R_3 + R_7) \,.$$

The Elmore delay at node i of an RC tree can also be described in a different format as

$$T_{Di} = \sum_{k} C_{Tk} R_k \ , \tag{3.15}$$

where the summation index k operates over all of the resistors that belong to the path from the input to node i. C_{Tk} is the total capacitance seen by the resistor k. For example, in Figure 3.3, $T_{D7} = R_1(C_1 + C_2 + \ldots + C_7) + R_6(C_3 + C_6 + C_7) + R_7C_7$. This form of expressing the summation is convenient since it has recursive properties [37]. For example, if the delays at all of the nodes of the *RC* tree shown in Figure 3.3 are to be determined, the delays can be calculated incrementally. The delay at node 1 is calculated as $T_{D1} = R_1(C_1 + C_2 + \ldots + C_7)$. The delays at node 2 and 3 can be calculated as $T_{D2} = T_{D1} + R_2(C_2 + C_4 + C_5)$ and $T_{D3} = T_{D1} + R_3(C_3 + C_6 + C_7)$. Finally, the delays at the sinks (nodes 4, 5, 6, and 7) can be calculated as $T_{D4} = T_{D2} + R_4C_4$, $T_{D5} = T_{D2} + R_5C_5$, $T_{D6} = T_{D3} + R_6C_6$, and $T_{D7} = T_{D3} + R_7C_7$. Recursive algorithms based on this method are called path-tracking methods and are highly efficient when evaluating the delays of large numbers of nodes within a complex VLSI circuit [37]. Calculating the delays at different nodes within large *RC* trees is accomplished by static timers that provide fast approximate delays within a complex VLSI circuit [124]-[126]. The devices are typically approximated by equivalent resistors (where the assumption of an operating point in a digital circuit can create substantial error) and the parasitic capacitors and the interconnect are modeled as an *RC* circuit. Thus, when simulating the temporal properties of a VLSI circuit, the system is typically reduced into an equivalent large *RC* tree where the delays are calculated using methods similar to the path-tracking method presented above. Other more accurate methods such as moment matching techniques [75]-[96] can be used if greater accuracy is required at the expense of

longer processing time and problematic numerical issues associated with such methods. These moment matching methods are discussed in the following section [75]-[96].

3.2 Higher Order Transient Response Approximations Using Moment Matching Techniques

Moment matching techniques [75]-[96] have gained significant popularity recently for evaluating the transient response of linear circuits. Software packages based on moment matching techniques such as Rapid Interconnect Circuit Evaluation (RICE) [79] have now become available. Other common model order reduction techniques are Pade via Lanczos (PVL) [97], Matrix Pade via Lanczos (MPVL) [99], Arnoldi Algorithm [100], Block Arnoldi Algorithm [101], and Passive Reduced-Order Interconnect Macromodeling Algorithm (PRIMA) [106],[107]. These methods seek a reduced order state-space representation of the interconnect portion of a VLSI circuit which contain nonlinear active elements (transistors) connected with linear passive interconnect. The reduced order state space representation of the interconnect is included in the model of the VLSI circuit instead of the original interconnect, permitting the resulting circuit to be simulated with a dynamic circuit simulator such as SPICE. With this approach, the simulation time of large VLSI circuits is significantly reduced since typically an interconnect circuit with thousands of state variables can be reduced into a smaller state space representation with only tens of state variables. Only those methods for evaluating the transient response of a linear circuit are discussed in this section.

3.2.1 Finding a Reduced Order Transfer Function of a System Using Moments

Wyatt [74] used the first moment of a transfer function (referred to as T_D by Elmore) to calculate the dominant pole of the transfer function. Wyatt used this dominant pole to determine a first order transfer function which approximates the transient response of a high order transfer function by

$$\hat{H}(s) = \frac{1}{sT_D + 1} = \frac{1}{sm_1 + 1}, \tag{3.16}$$

where m_1 is the first moment of the original transfer function. However, not all systems can be accurately approximated by a single pole. Some systems have many poles with relatively close values (similar frequencies) and these poles cannot be neglected without creating serious error. Moreover, many systems have low frequency dominant zeros that play a significant role in the transient response. The Wyatt approximation assumes no low frequency zeros. Thus, for such systems, a first order approximation is inappropriate and higher order approximations are necessary.

Wyatt's approach of using the first moment of the transfer function to calculate the dominant pole can be extended to calculate more than one pole by using higher order moments of the transfer function [75]-[96]. A normalized transfer function of a given linear system can be expanded into powers of s such as

$$H(s) = \frac{1 + a_1 s + a_2 s^2 + \ldots + a_n s^m}{1 + b_1 s + b_2 s^2 + \ldots + b_m s^n} = 1 + m_1 s + m_2 s^2 + m_3 s^3 + \ldots, \tag{3.17}$$

where m_i is the i^{th} moment of the transfer function. This transfer function is of order m and has n poles and m zeros and for a real system $m < n$. A reduced order transfer

function can be used to approximate the transient response of the original transfer function,

$$\hat{H}(s) = \frac{1 + \hat{a}_1 s + \hat{a}_2 s^2 + \ldots + \hat{a}_r s^r}{1 + \hat{b}_1 s + \hat{b}_2 s^2 + \ldots + \hat{b}_q s^q} = 1 + \hat{m}_1 s + \hat{m}_2 s^2 + \hat{m}_3 s^3 + \ldots . \tag{3.18}$$

The reduced order transfer function is of order q and has q poles and r zeros where $q < m$ and $r < q$. This reduced order transfer function has $q + r$ variables ($\hat{a}_1 - \hat{a}_r$ and $\hat{b}_1 - \hat{b}_q$) which can be chosen to match the first $q + r$ moments of the reduced order transfer function to the first $q + r$ moments of the original transfer function (*i.e.*, $m_1 = \hat{m}_1, m_2 = \hat{m}_2, \ldots, m_{q+r} = \hat{m}_{q+r}$). For example, in Wyatt's case the transfer function has $r = 0$ and $q = 1$ and thus only the first moment can be matched. The choice of $\hat{b}_1 = m_1$ as given by (3.16) matches the first moment of the original transfer function as can be seen if (3.16) is expanded in terms of the powers of s.

For a reduced order system that matches only the first moment of the original transfer function to be an accurate approximation, it is necessary that s be sufficiently small so that the term $m_2 s^2$ and the subsequent terms are negligible with respect to the terms $1 + m_1 s$ in the original system. If the frequency is sufficiently low (s is small) so that this condition is satisfied, all of the terms with higher powers of s are sufficiently small in both the original system and the reduced order system, making the matching of higher order moments unimportant. This behavior demonstrates that using only the first moment represents a low frequency approximation of the original transfer function. The more moments that are matched, the higher the values of s (or higher frequencies) for which the reduced order system can approximate the original system.

For *RC* circuits, the response is monotone since the poles and zeros of an *RC* circuit are on the negative real axis [130]-[132]. The monotone response of an *RC* circuit is shown in Figure 3.4. Typically, monotone responses contain low frequency components and can therefore be accurately approximated by matching only a few moments. Alternatively, for underdamped *RLC* circuits, reflections occur which cause higher frequency components to be non-negligible in the transient response of the circuit as illustrated in the response shown in Figure 3.4. Thus, more moments should be matched to account for these higher frequency components in the case of an *RLC* circuit as compared to an *RC* circuit.

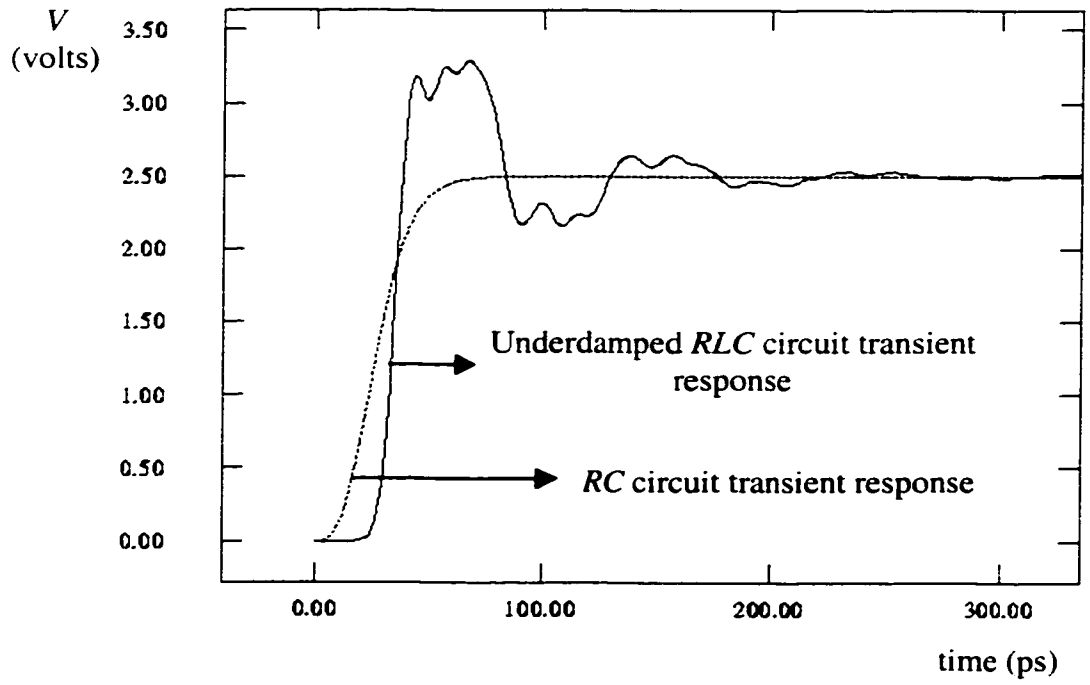


Figure 3.4. Transient responses of an *RC* circuit and an underdamped *RLC* circuit.

A standard method to evaluate a reduced order transfer function that approximates the transient behavior of a higher order transfer function is the Asymptotic Waveform Evaluation (AWE) method [77]-[82]. The AWE method approximates a transfer function by a q^{th} order transfer function of the form,

$$\hat{H}(s) = \frac{k_1}{s - p_1} + \frac{k_2}{s - p_2} + ... + \frac{k_q}{s - p_q}, \tag{3.19}$$

where $p_1, p_2, \ldots, p_q$ are the q poles of the reduced order transfer function and $k_1, k_2, \ldots, k_q$ are the corresponding residues, respectively. The reduced order transfer function has $2q$ variables (q poles and q residues) and can match $2q$ moments of the original transfer function. Given the first $2q$ moments of the original transfer function, $m_0, m_1, \ldots, m_{2q-1}$, the values of the q poles and q residues of the reduced order transfer function are determined so as to match these moments. The moments of the reduced order transfer function can be calculated from the time domain as

$$\hat{H}(s) = \int_0^\infty \hat{h}(t)e^{-st}dt = 1 - s\int_0^\infty t\hat{h}(t)dt + \frac{s^2}{2!}\int_0^\infty t^2\hat{h}(t)dt - ..... \tag{3.20}$$

Thus, the moments of the reduced order transfer function can be calculated from

$$m_i = \frac{(-1)^i}{i!}\int_0^\infty t^i\hat{h}(t)dt. \tag{3.21}$$

The reduced order transfer function in the time domain can be calculated by taking the inverse Laplace transform of (3.19) and is

$$\hat{h}(t) = k_1e^{p_1t} + k_2e^{p_2t} + ... + k_qe^{p_qt}. \tag{3.22}$$

Thus, the moments of the reduced order transfer function can be calculated from (3.21) and equated to the moments of the original transfer function, m_0, m_1, ..., m_{2q-1}, leading to the set of equations,

$$\begin{aligned}
-\left(\frac{k_1}{p_1}+\frac{k_2}{p_2}+\ldots.+\frac{k_q}{p_q}\right) &= m_0,\\
-\left(\frac{k_1}{p_1^{\,2}}+\frac{k_2}{p_2^{\,2}}+\ldots.+\frac{k_q}{p_q^{\,2}}\right) &= m_1,\\
&\vdots\\
-\left(\frac{k_1}{p_1^{\,2q}}+\frac{k_2}{p_2^{\,2q}}+\ldots.+\frac{k_q}{p_q^{\,2q}}\right) &= m_{2q-1},
\end{aligned} \tag{3.23}$$

where $m_0 = 1$ for a normalized transfer function. Note that the same result can be reached by simply expanding (3.19) into powers of s. Solving these $2q$ nonlinear equations in $2q$ variables, the poles and residues of the reduced order transfer function can be determined. However, the direct method of solving these nonlinear equations is not used to determine the poles and zeros. Alternative methods have been introduced in [77]-[80] to calculate the poles of the reduced order transfer function by determining the zeros of a polynomial function, the coefficients of which can be found by solving q linear equations. Specifically, the required reduced order system is of the form,

$$\hat{H}(s)=\frac{1+\hat{a}_1 s+\hat{a}_2 s^2+\ldots+\hat{a}_{q-1}s^{q-1}}{1+\hat{b}_1 s+\hat{b}_2 s^2+\ldots+\hat{b}_q s^q}=m_0+m_1 s+m_2 s^2+m_3 s^3+\ldots+m_{2q-1}s^{2q-1}. \tag{3.24}$$

Multiplying the denominator of the left hand side by the right hand side, the following expression results

$$1+\hat{a}_1 s+\hat{a}_2 s^2+\ldots+\hat{a}_{q-1}s^{q-1}=\left(m_0+m_1 s+m_2 s^2+m_3 s^3+\ldots+m_{2q-1}s^{2q-1}\right)\cdot \left(1+\hat{b}_1 s+\hat{b}_2 s^2+\ldots+\hat{b}_q s^q\right) \tag{3.25}$$

By comparing the coefficients of the same powers of s in the right and left hand sides of (3.25), $2q$ equations result. The set of equations that result from comparing the coefficients of $s^q - s^{2q-1}$ are

$$\begin{aligned}
s^q &\rightarrow \quad 0=m_0 b_q+m_1 b_{q-1}+m_2 b_{q-2}+\ldots+m_{q-1}b_1+m_q \\
s^{q-1} &\rightarrow \quad 0=m_1 b_q+m_2 b_{q-1}+m_3 b_{q-2}+\ldots+m_q b_1+m_{q-1} \\
&\vdots \\
s^{2q-1} &\rightarrow \quad 0=m_{q-1} b_q+m_q b_{q-1}+m_{q+1} b_{q-2}+\ldots+m_{2q-2}b_1+m_{2q-1}.
\end{aligned} \tag{3.26}$$

Thus, the poles of a q^{th} order approximation are the zeros of the polynomial whose coefficients are the solution of the following system of linear equations,

$$\begin{bmatrix} m_0 & m_1 & \cdots & m_{q-1} \\ m_1 & m_2 & \cdots & m_q \\ \vdots & \vdots & & \vdots \\ m_{q-1} & m_q & \cdots & m_{2q-2} \end{bmatrix} \begin{bmatrix} b_q \\ b_{q-1} \\ \vdots \\ b_1 \end{bmatrix} = -\begin{bmatrix} m_q \\ m_{q+1} \\ \vdots \\ m_{2q-1} \end{bmatrix}. \tag{3.27}$$

After the poles are found, the residues of the q poles can be determined by solving the first q linear equations of (3.23) which are given in matrix form by

$$\begin{bmatrix} \frac{1}{p_1} & \frac{1}{p_2} & \cdots & \frac{1}{p_q} \\ \frac{1}{p_1^2} & \frac{1}{p_2^2} & \cdots & \frac{1}{p_q^2} \\ \vdots & \vdots & & \vdots \\ \frac{1}{p_1^q} & \frac{1}{p_2^q} & \cdots & \frac{1}{p_q^q} \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \\ \vdots \\ k_q \end{bmatrix} = -\begin{bmatrix} m_0 \\ m_1 \\ \vdots \\ m_{q-1} \end{bmatrix}. \tag{3.28}$$

Once a reduced order transfer function is found in the form given by (3.19), calculating the output of the original system for an arbitrary input is straightforward. For example, for a unit step input, the output response can be calculated as

$$\hat{Y}(s) = \frac{1}{s}\hat{H}(s) = \frac{1}{s}\frac{k_1}{s-p_1} + \frac{1}{s}\frac{k_2}{s-p_2} + ... + \frac{1}{s}\frac{k_q}{s-p_q}, \tag{3.29}$$

where $\hat{Y}(s)$ is the approximate frequency domain output of the system calculated using the reduced order transfer function. The time domain response of a unit step input can be determined by considering the inverse Laplace transform of (3.29) and is

$$\hat{y}(t) = -(\frac{k_1}{p_1} + \frac{k_2}{p_2} + ... + \frac{k_q}{p_q}) + \frac{k_1}{p_1}e^{p_1 t} + \frac{k_2}{p_2}e^{p_2 t} + ... + \frac{k_q}{p_q}e^{p_q t}. \tag{3.30}$$

The same procedure can be followed for a ramp input, an exponential input, or any other type of input signal. Arbitrary accuracy can be achieved by matching additional moments although there are practical and numerical limitations as is discussed later in this chapter.

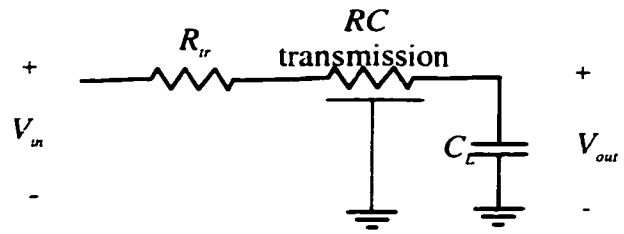


Figure 3.5. An *RC* transmission line with a source resistance and a load capacitance.

As an example, consider the circuit shown in Figure 3.5. The circuit is composed of a distributed *RC* transmission line driven by a lumped resistance R_{tr} (which represents the output impedance of the driving gate) and a load capacitance C_L (which represents the input capacitance of the driven gate). The line has a total resistance of R_t and a total capacitance of C_t. The values of the circuit elements shown in Figure 3.5 are R_t = 50 Ω, C_t = 1 pF, R_{tr} = 25 Ω, and C_L = 0.05 pF. With these parameters, the first four moments of the transfer function are

$$\begin{bmatrix} m_0 \\ m_1 \\ m_2 \\ m_3 \end{bmatrix} = \begin{bmatrix} 1 \\ -53.437 \\ 2499.589 \\ -115386.246 \end{bmatrix}. \tag{3.31}$$

The polynomial representing the denominator of a second order approximation can be determined by solving the system of linear equations in (3.27) with $q = 2$ and the moments given in (3.31). This polynomial is given by

$$1 + 51.086 \cdot s + 230.343 \cdot s^2 . \tag{3.32}$$

The zeros of this polynomial are the poles of the approximate transfer function and are

$$\begin{bmatrix} p_1 \\ p_2 \end{bmatrix} = \begin{bmatrix} -0.021697 \\ -0.200087 \end{bmatrix}. \tag{3.33}$$

The residues of the transfer function corresponding to these poles can be determined by solving the following system of linear equations according to (3.28),

$$\begin{bmatrix} -46.008 & -4.997 \\ 2124.162 & 24.978 \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} = -\begin{bmatrix} 1 \\ -53.437 \end{bmatrix}. \tag{3.34}$$

These residues are

$$\begin{bmatrix} k_1 \\ k_2 \end{bmatrix} = \begin{bmatrix} 0.025578 \\ -0.035784 \end{bmatrix}. \tag{3.35}$$

Finally, the output response of the circuit shown in Figure 3.5 to a step input signal is calculated using (3.30). The resulting time domain signal is given by

$$\hat{y}(t) = 1 - 1.178845e^{-0.216t} + 0.178845e^{-0.2t}. \tag{3.36}$$

This time domain signal is compared to SPICE in Figure 3.6. Note that despite the fact that the circuit shown in Figure 3.5 has infinite number of poles (form the distributed *RC* transmission line), only two dominant poles can produce an almost exact response as compared to SPICE. In general, more poles than the two poles used in this specific example may be needed to produce a highly accurate response.

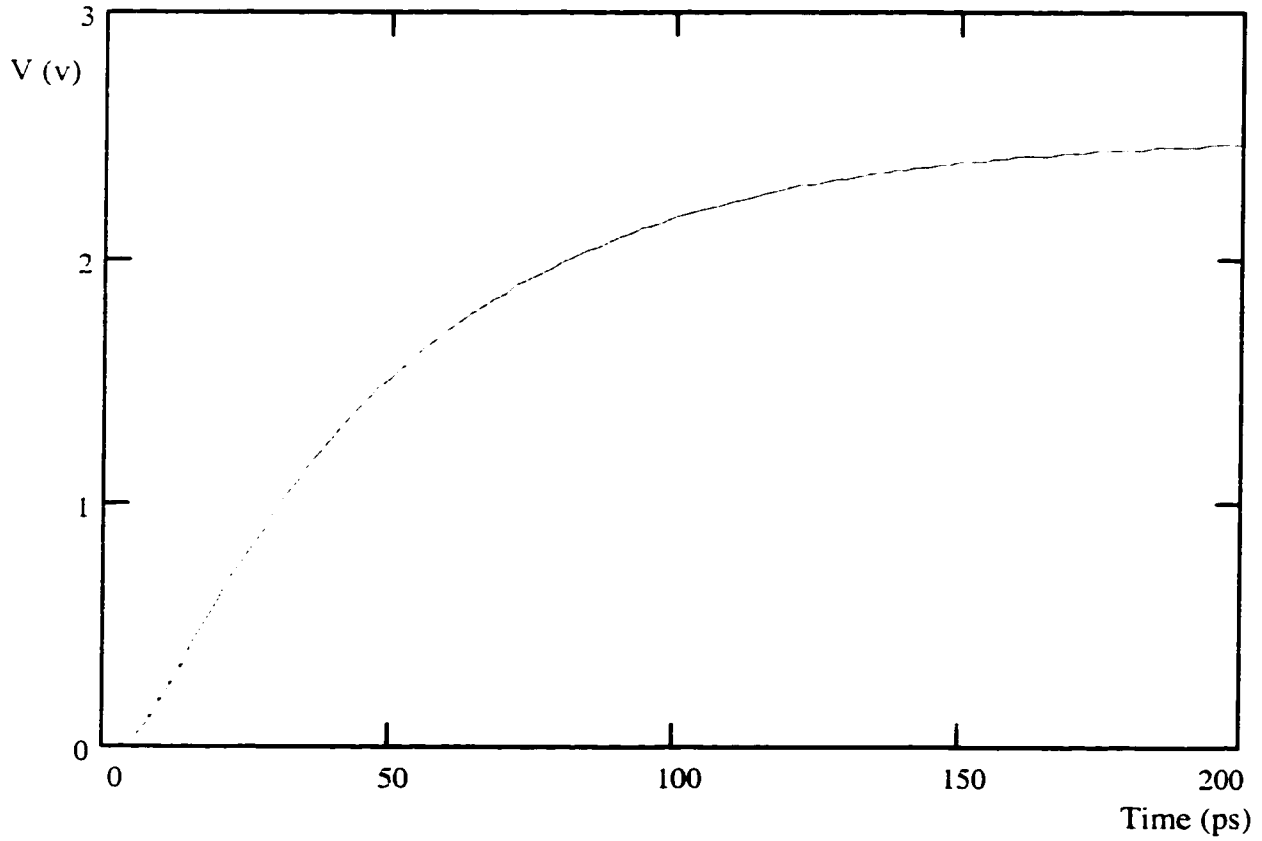


Figure 3.6. Comparison of a second order AWE approximation to SPICE for the output node of the circuit shown in Figure 3.5. The SPICE simulation is represented by a solid line while AWE is represented by a dotted line.

3.2.2 Calculating the Moments of an *RLC* Tree

In the previous subsection, the poles and residues of a reduced order transfer function are determined in order to approximate the transient response of a higher order system. In the analysis, it is assumed that the moments of the higher order

system that is approximated by a reduced order system are known. The process for calculating the moments of an *RLC* tree is illustrated in this subsection since most typical VLSI interconnects are tree structured [77].

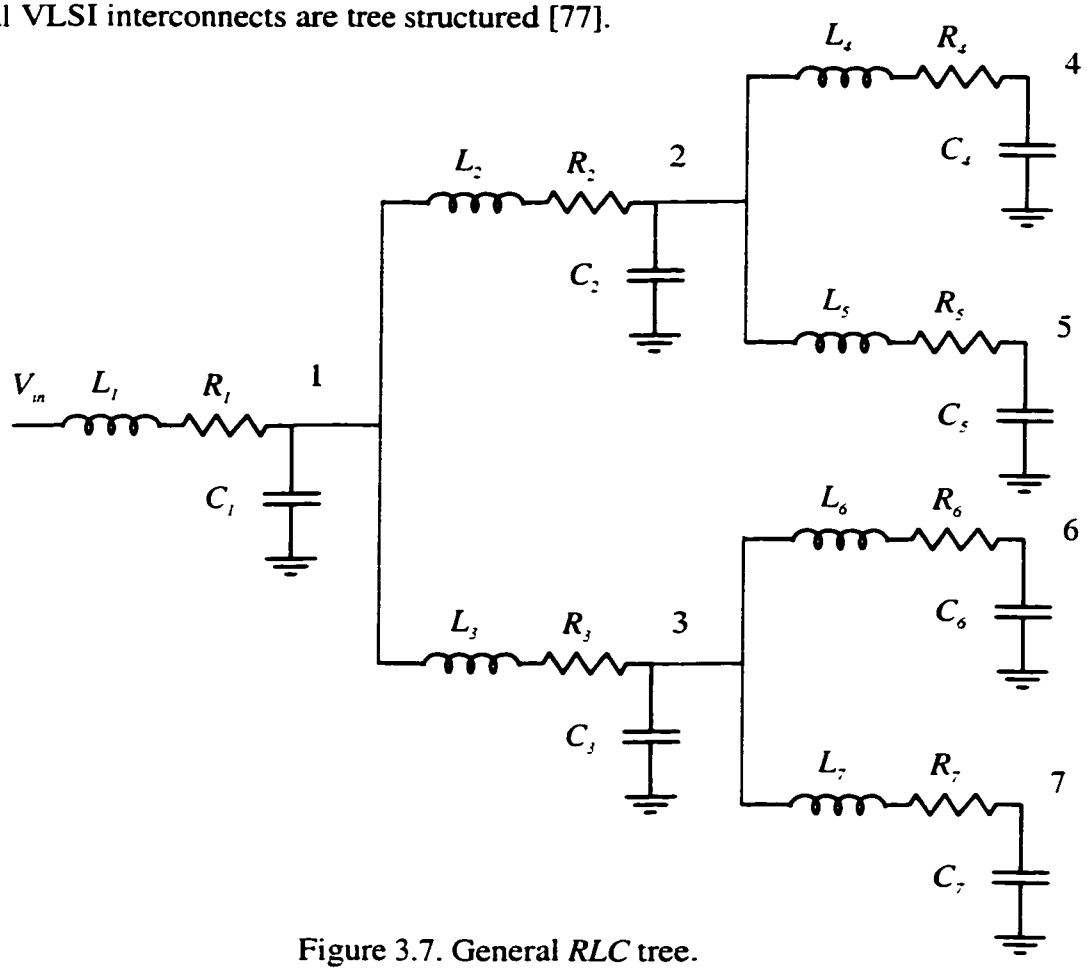


Figure 3.7. General *RLC* tree.

Consider the tree shown in Figure 3.7. Using the same method as described in section 3.1.3, the voltage drop at any node i as compared to the input voltage can be found in the frequency domain and is given by

$$V_{in}(s) - V_i(s) = \sum_k C_k V_k(s) s[R_{ki} + L_{ki} s], \tag{3.37}$$

where L_{ik} is the common inductance from the input to nodes i and k. If the input is a unit impulse, $V_{in}(s)$ is equal to 1.0 and the voltages at the nodes of the tree are the unit

impulse responses of these nodes (*i.e.*, the transfer functions at these nodes). Thus, the transfer function $H_i(s)$ at node i is given by $V_i(s)$ and is

$$H_i(s) = 1 - \sum_k C_k H_k(s) s[R_{ki} + L_{ki} s], \tag{3.38}$$

where the transfer functions at any node k is described by the voltage at this node with a unit impulse input signal, *i.e.*, $H_k(s) = V_k(s)$. Since a different transfer function exists at each node of an *RLC* tree, the moments at each node are different. At any node i of an *RLC* tree, the moments can be derived from the expansion of the transfer function at node i and are

$$H_i(s) = m_0^i + m_1^i s + m_2^i s^2 + \ldots., \tag{3.39}$$

where m_j^i is the j^{th} moment of the transfer function at node i. Substituting the transfer function expansion in (3.39) into (3.38), a relation between the moments can be found as

$$m_0^i + m_1^i s + m_2^i s^2 + \ldots. = 1 - \sum_k [C_k R_{ki} s + C_k L_{ki} s^2][m_0^k + m_1^k s + m_2^k s^2 + \ldots.], \tag{3.40}$$

which can be simplified in terms of the powers of s to

$$m_0^i + m_1^i s + m_2^i s^2 + \ldots. = 1 - \left[\sum_k (C_k m_0^k) R_{ik}\right] s - \left[\sum_k (C_k m_1^k) R_{ik} + \sum_k (C_k m_0^k) L_{ik}\right] s^2 - \ldots. \tag{3.41}$$

Thus, by comparing the powers of s in the left and right hand sides of (3.40) and (3.41), the moments of an *RLC* tree can be calculated at any node i and are

$$\begin{aligned} m_0^i &= 1 \\ m_1^i &= -\sum_k (C_k m_0^k) R_{ik} \\ m_2^i &= -\sum_k (C_k m_1^k) R_{ik} + \sum_k (C_k m_0^k) L_{ik}. \\ &\vdots \end{aligned} \tag{3.42}$$

From the observation that m_0 is equal to 1 at all of the nodes of an *RLC* tree, general recursive equations for calculating the moments at node i of an *RLC* tree are given by

$$\begin{aligned} m_0^i &= 1 \\ m_1^i &= -\sum_k C_k R_{ik} \\ m_j^i &= -\sum_k (C_k m_{j-1}^k) R_{ik} + \sum_k (C_k m_{j-2}^k) L_{ik} \qquad j = 2, 3, \ldots\ldots \end{aligned} \tag{3.43}$$

The first moment is the Elmore delay which can be efficiently calculated at all of the nodes of an *RLC* tree using the path-tracking method introduced in section 3.1.3. The second moment can be calculated by multiplying each capacitor by the first moment of the transfer function of the voltage across the capacitor followed by application of the path-tracking method [37], [93] to calculate the first term with a new value of the *equivalent* capacitance. The second term of the second moment can be calculated in the same way but with the inductance rather than the resistance and m_0 rather than m_1. The third moment can be calculated in the same manner using the first and second moments and so on. This recursive method is highly efficient in calculating the moments in all of the nodes of an *RLC* tree. Once the moments of the transfer functions at different nodes are known, the method discussed in the previous subsection 3.2.1 can be used to determine the transient response at any node for an arbitrary input. For more general interconnect structures such as networks that include meshes or coupling capacitances, more complex strategies can be used to calculate the moments at different nodes of the circuit using a state space representation of the interconnect [78]-[80].

3.2.3 Numerical and Computational Issues

The AWE method has several numerical and computational issues which are discussed in this subsection. These topics include pole instability, numerical precision, and computational efficiency.

A. Unstable Poles

The AWE method introduced above uses a Pade type approximation [83]-[87] to determine the poles and zeros of a reduced order transfer function given the moments of a higher order system. Pade type approximations can generate unstable poles in the reduced order transfer function of a stable system [83]-[87]. These unstable (or bad) poles must be compensated for in order to avoid a divergent transient response for a stable system. Methods as simple as just discarding the bad poles have been used [77], [78]. However, such simple methods deteriorate the accuracy of the approximated transient response. More complex methods that use constrained optimization for *RC* networks are introduced in [88]. These methods transform the variables p_i in (3.23) into another set of variables x_i such that

$$\frac{1}{p_i} = -e^{-x_i} . \tag{3.44}$$

Equation (3.23) is solved for the variables x_i using a Newton-Raphson iteration, permitting the use of (3.39) to determine the poles p_i. Such a transformation guarantees that the poles p_i are real and negative for any value of the variables x_i. The poles of any passive *RC* network are real and negative and hence this method can be used with *RC* networks. This method results in better accuracy for a transient

response approximation as compared to an unconstrained approximation. However, the Newton-Raphson method does not always converge and the solution depends upon the initial values chosen for the variables x_i. An estimate of the poles is initially determined by solving (3.23) without applying the transformation described by (3.39) and changing the sign of the unstable poles. Thus, numerical issues arise that must be considered when using the method described in [88]. The stability problem is worse for an *RLC* network than in the case of an *RC* network. A more efficient technique using a *moment-shifting* approach was introduced by Anastasakis *et al.* in [89] that improves the stability of an AWE approximation for *RLC* networks. The moment shifting technique is briefly discussed in subsection C.

As an example of encountering unstable poles, consider the *RC* tree shown in Figure 3.8. The first four moments at node O_4 shown in Figure 3.8 can be calculated as described in section 3.2.2 and are

$$\begin{bmatrix} m_0 \\ m_1 \\ m_2 \\ m_3 \end{bmatrix} = \begin{bmatrix} 1 \\ -11.91 \\ 118.776 \\ -1146.416 \end{bmatrix}. \tag{3.45}$$

The poles of a second order AWE approximation are calculated as described in section 3.2.1 using these moment values and are

$$\begin{bmatrix} p_1 \\ p_2 \end{bmatrix} = \begin{bmatrix} 0.00154 \\ -0.10032 \end{bmatrix}. \tag{3.46}$$

Note that the first pole has a positive value which would result in a divergent exponential creating, an unstable system. This positive pole is caused by AWE since

a passive *RC* circuit can only produce negative real poles. If this pole is discarded and only the single negative pole p_2 is used in the transient response, the signal waveform is illustrated in Figure 3.9 results. Note the large error (24% in the propagation delay) between AWE and SPICE which is a direct consequence of the unstable poles in the AWE approximations. These unstable poles can actually appear for low order approximations and *RC* circuits. This instability can therefore severely limit the utility of low order approximations and force the calculation of unnecessarily higher order approximations which require additional computational time. These instability issues become even more severe with *RLC* circuits. An alternative method is introduced in Chapter 12 which has guaranteed stability for approximation orders with less than five poles. This characteristic is quite useful for analyzing the temporal properties of *RC* circuits which exhibit monotone behavior. These *RC* systems can typically be accurately simulated with a few poles. The new method is compared to AWE in Appendix F.

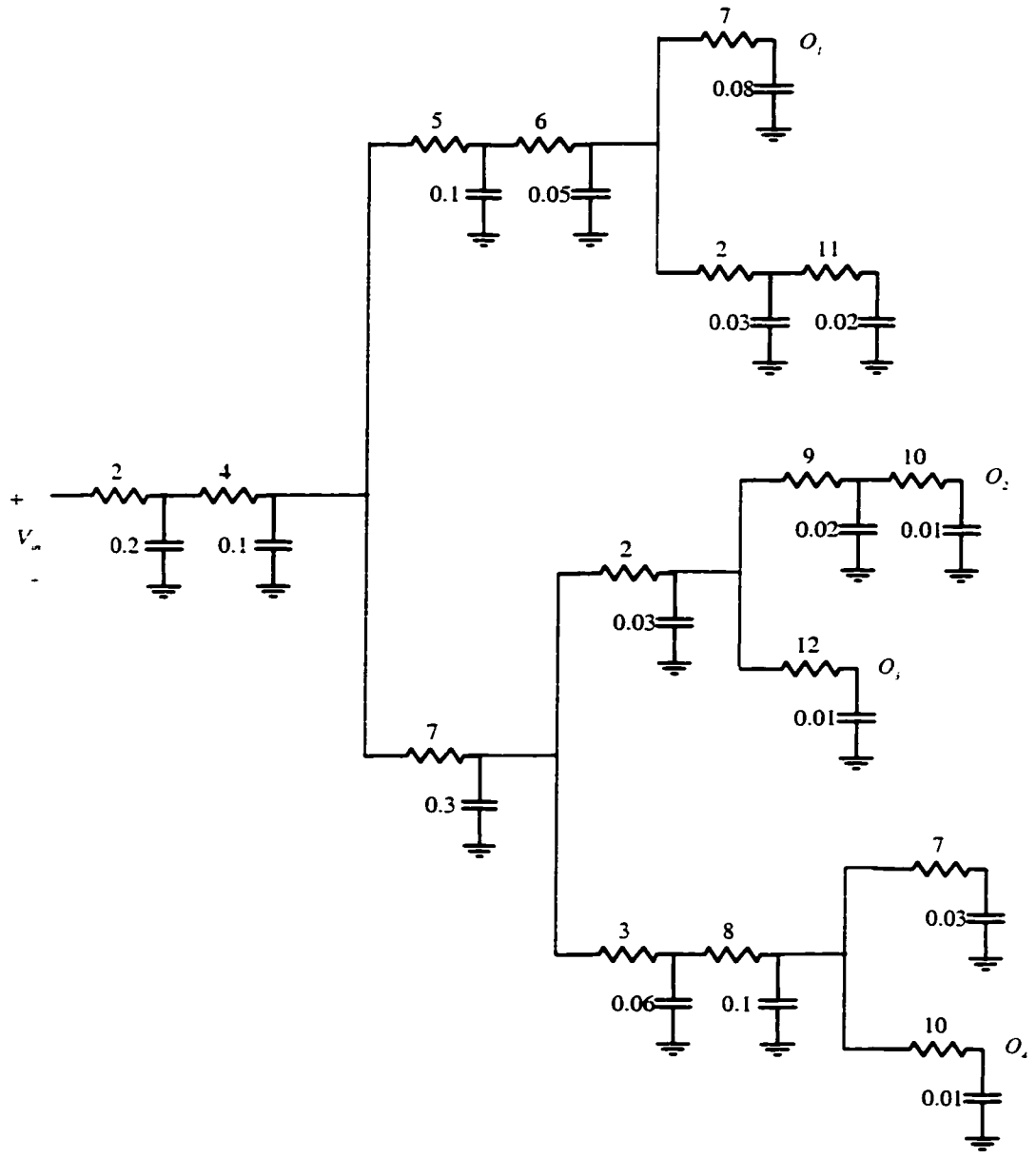


Figure 3.8. A general *RC* tree. The resistance values shown are in ohms and capacitance values are in pF.

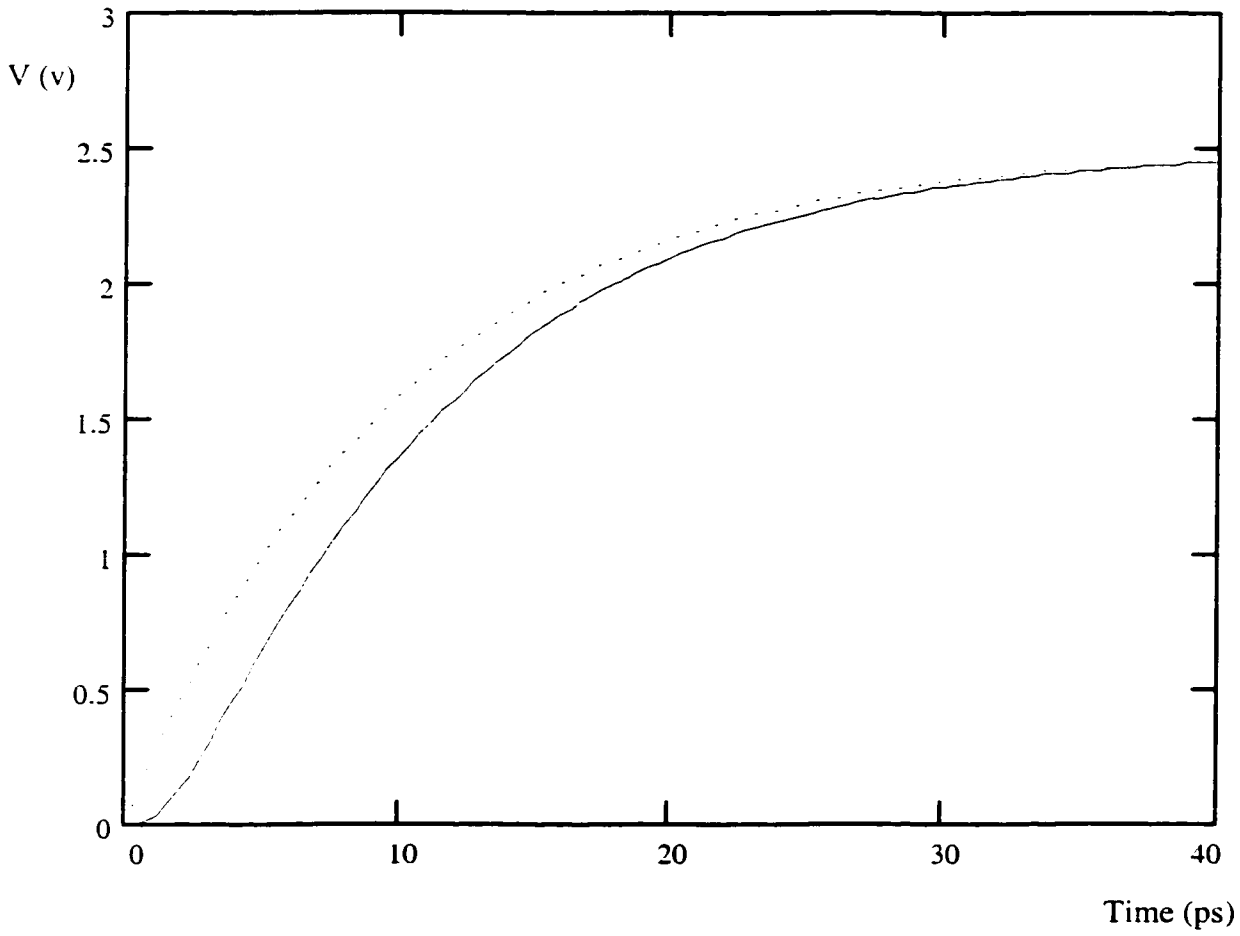


Figure 3.9. Comparison of a second order AWE approximation to SPICE of output node O_4 for the *RLC* circuit shown in Figure 3.8. The SPICE simulation is represented by the solid line while AWE is represented by the dotted line.

B. Numerical Precision

In addition to the stability problem discussed in the previous subsection, AWE has several numerical problems which limit the capability of determining higher order approximations. One numerical issue can be observed by noting that the moments of the original system described by (3.17) are given by

$$\begin{aligned} m_0 &= -(\frac{k_1}{p_1} + \frac{k_2}{p_2} + ... + \frac{k_n}{p_n}), \\ m_1 &= -(\frac{k_1}{p_1^2} + \frac{k_2}{p_2^2} + ... + \frac{k_n}{p_n^2}), \\ &\vdots \\ m_{2n-1} &= -(\frac{k_1}{p_1^{2n}} + \frac{k_2}{p_2^{2n}} + ... + \frac{k_n}{p_n^{2n}}). \end{aligned} \tag{3.47}$$

Note that this system of equations is different from the system of equations described by (3.23) since the moments described in (3.47) are calculated based on all of the n exact poles of the original system rather than using only the approximate q poles as in (3.23). The moments described in (3.47) represent the exact moments calculated for the original system by the method discussed in section 3.2.2 or by state space methods. Equation (3.23) is the set of equations used to match a subset of these moments (the first $2q$ moments) with the moments of a reduced order system with q poles and q residues. Equation (3.47) can be derived from (3.17) in the same manner (3.23) is derived from (3.19) and (3.24).

The system of equations described by (3.47) involves high powers of the poles. A q pole approximation requires calculating the first $2q$ moments which have powers up to p_i^{2q-1}. For an eight pole approximation, the fifteenth moment must be

calculated which involves the fifteenth power of all of the poles. If the values of the poles are normalized such that the smallest (most dominant) pole is one and there is a ten times larger pole in the approximation, the fifteenth power of the reciprocal of such a pole is 10^{-15}. This number is much smaller than the reciprocal of the most dominant pole (one). This discrepancy increases with q. With a limited number of significant digits on a computer machine, the addition of large numbers to much smaller numbers causes the truncation of the smaller numbers. For a machine with r significant digits in a word, the j^{th} moment will not carry any information characterizing poles with $(p_i / p_1)^j$ greater than 10^r, where p_1 is assumed to be the pole with the smallest magnitude. Hence, adding more moments with moment matching techniques might not lead to any improvement in the accuracy of the approximation since the higher moments do not include any extra information. As an example, the first four exact poles of the *RC* circuit shown in Figure 3.5 of section 3.2.1 are given by

$$\begin{bmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \end{bmatrix} = \begin{bmatrix} -0.021692 \\ -0.247302 \\ -0.805239 \\ -1.725512 \end{bmatrix}. \tag{3.48}$$

The ratios of the second, third, and fourth poles to the first pole are

$$\begin{bmatrix} \frac{p_2}{p_1} \\ \frac{p_3}{p_1} \\ \frac{p_4}{p_1} \end{bmatrix} = \begin{bmatrix} 11.449 \\ 37.278 \\ 79.884 \end{bmatrix}. \tag{3.49}$$

For a machine with sixteen decimal significant digits, the fourth pole will not appear after the eighth moment, the third pole after the tenth moment, and the second pole after the fifteenth moment. Hence, AWE will simply fail to determine any approximation beyond a fourth order approximation for this specific circuit since a five pole system requires the calculation of the tenth moment which has no useful information about poles beyond the third pole. A sixth order AWE is compared to SPICE in Figure 3.10. Note that a six pole approximation is highly inaccurate due to these numerical errors.

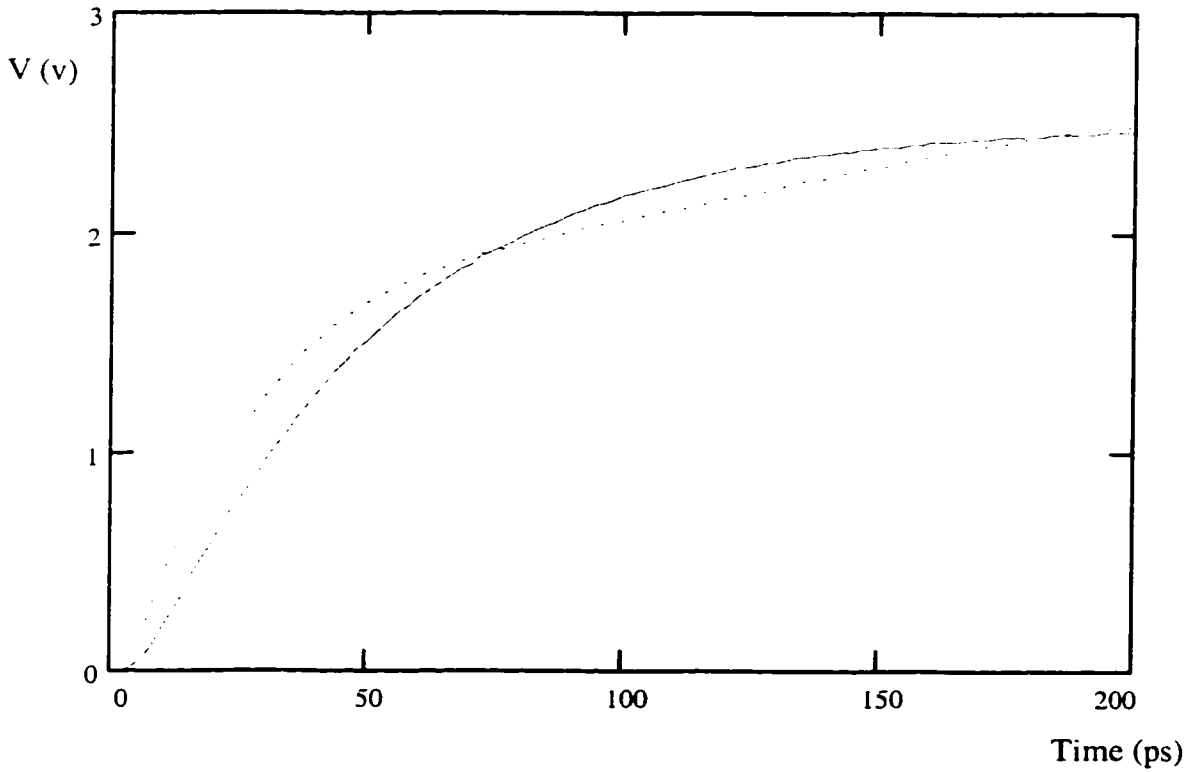


Figure 3.10. Comparison of a sixth order AWE approximation to SPICE for the output node of the circuit shown in Figure 3.5 of section 3.2.1. The SPICE simulation is represented by a solid line while AWE is represented by a dotted line.

It is interesting to note that the same property that makes AWE work is that which makes AWE fail when determining higher order approximations. This property is the disparity in pole values. An example is presented here to illustrate the concepts discussed in this section since there is no need to calculate high order approximations for this specific *RC* circuit when a second order approximation is sufficiently accurate as is shown in 3.2.1. Another example is discussed at the end of this section that demonstrates how AWE can fail in determining an accurate approximation of the response of an *RLC* line.

Another numerical problem is encountered when calculating the residues using the system of linear equation given by

$$\begin{bmatrix} \frac{1}{p_1} & \frac{1}{p_2} & \cdots & \frac{1}{p_q} \\ \frac{1}{p_1^2} & \frac{1}{p_2^2} & \cdots & \frac{1}{p_q^2} \\ \vdots & \vdots & & \vdots \\ \frac{1}{p_1^q} & \frac{1}{p_2^q} & \cdots & \frac{1}{p_q^q} \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \\ \vdots \\ k_q \end{bmatrix} = -\begin{bmatrix} m_0 \\ m_1 \\ \vdots \\ m_{q-1} \end{bmatrix}. \tag{3.50}$$

The poles are raised to the power q in the matrix describing a system of linear equations. As previously discussed, a discrepancy in the values of the poles is amplified by the large powers of the poles when q is relatively large. For example, calculating the residues of a fourth order approximation of the *RC* circuit shown in Figure 3.5 requires inverting the following matrix,

$$\begin{bmatrix} -0.918 & -1.151 & -4.043 & -46.099 \\ 0.843 & 1.326 & 16.353 & 2125 \\ -0.774 & -1.528 & -66.132 & -97969 \\ 0.711 & 1.760 & 267.437 & 4516340 \end{bmatrix}. \tag{3.51}$$

There is a ratio of about 10^7 between the largest and smallest elements. This ratio becomes 10^{14} for an eight pole approximation. A matrix with such a large difference between its elements is called *ill conditioned.* The inversion of such a matrix is prone to large numerical errors. On a machine with a reasonably large word width and with appropriate numerical techniques, the AWE method is limited to approximations with less than eight poles due to these numerical problems [77], [79]. In addition, certain poles calculated using AWE may be unstable as discussed in the previous subsection and therefore are discarded which further limits the ability of AWE to accurately model highly complicated signals.

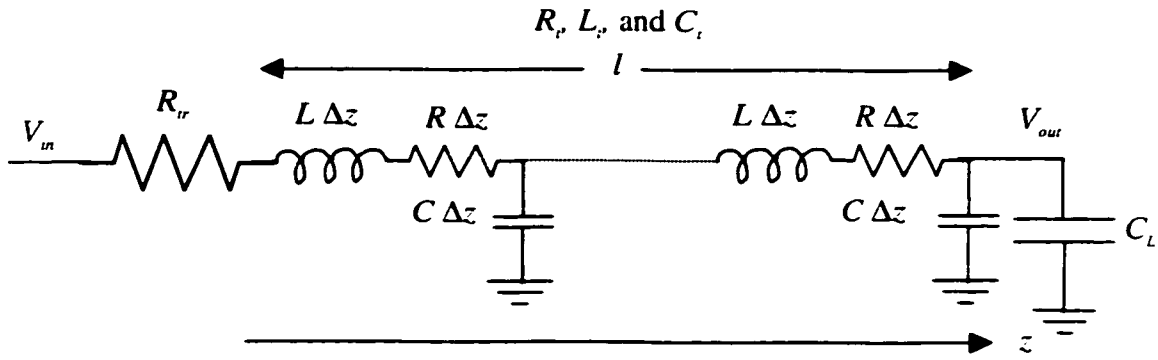
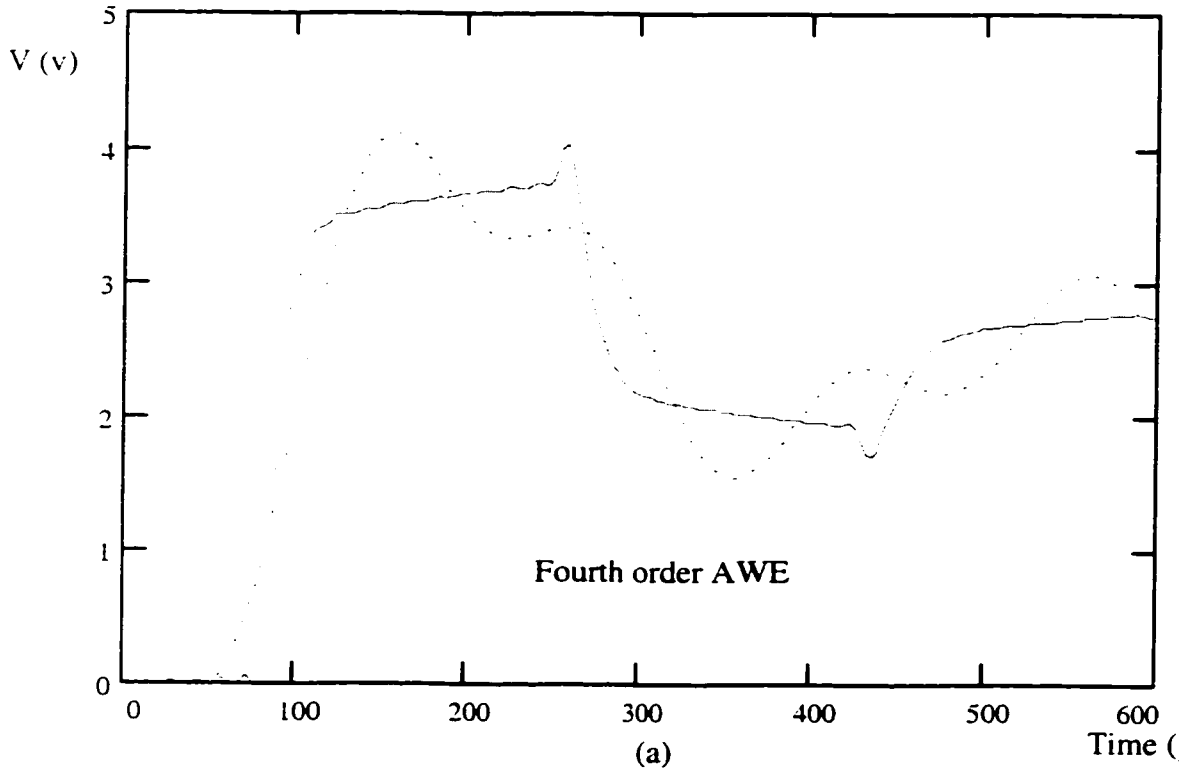


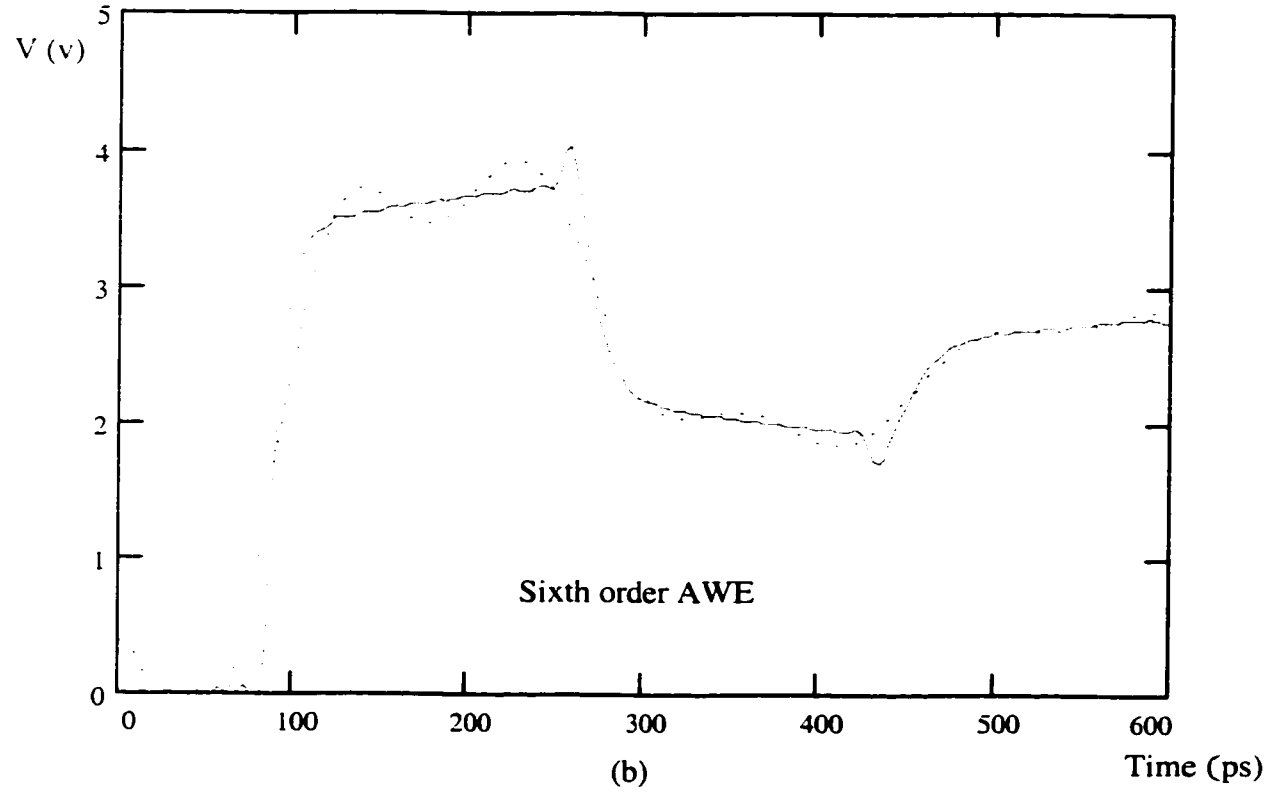
Figure 3.11. An *RLC* transmission line with a source resistance and a load capacitance.

To demonstrate the limitations of AWE when simulating complicated signals which require high approximation orders, consider the transmission line circuit with a source resistance R_{tr} and a load capacitance C_L shown in Figure 3.11. The values of the circuit elements are $R_t = 40\ \Omega$, $L_t = 7$ nH, $C_t = 1$ pF, $R_{tr} = 10\ \Omega$, and $C_L = 0.1$ pF.

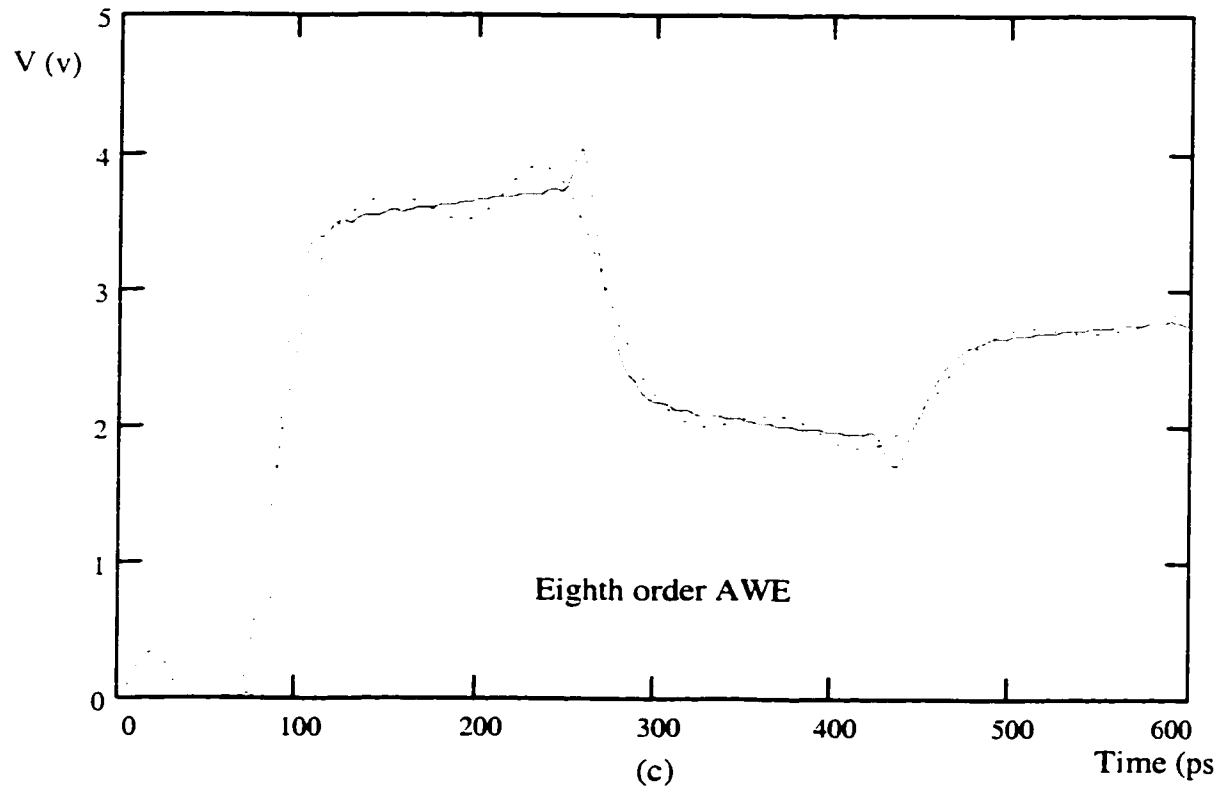
The transient response of this circuit is calculated based on fourth, sixth, eighth, tenth, and twelfth order AWE approximations. These approximations are compared to SPICE in Figure 3.12. The poles determined by the AWE approximations are compared in Table 3.1 to the exact poles of the circuit shown in Figure 3.11. Also, the rise times determined by the AWE approximations are compared to the exact rise time simulated by SPICE in Table 3.2. Note that the accuracy of the AWE approximations improves as the order increases up to the eight pole approximation. However, as the approximation order increases beyond the eighth order approximation, the numerical issues discussed in this section cause the accuracy of the approximation to deteriorate. Thus, for this example, the most accurate AWE order is eight, which represents the best achievable results by AWE. Even for the best case of eight poles, AWE suffers an error of 167% in the rise time as compared to SPICE. The area of the rise time signal shown in Figure 3.12 is magnified in Figure 3.13 to illustrate this large error. Such a significant error in the best approximation demonstrates that AWE is incapable of accurately simulating this circuit due to numerical errors with high order approximations. Another method for simulating complicated waveforms is presented in Chapter 12 which is numerically accurate for any approximation order and can accurately and efficiently simulate any complicated signal. The new method is compared to AWE in Appendix F.


V (v)
5
4
3
2
1
0
0
100
200
300
400
500
600
Time (ps)
Fourth order AWE

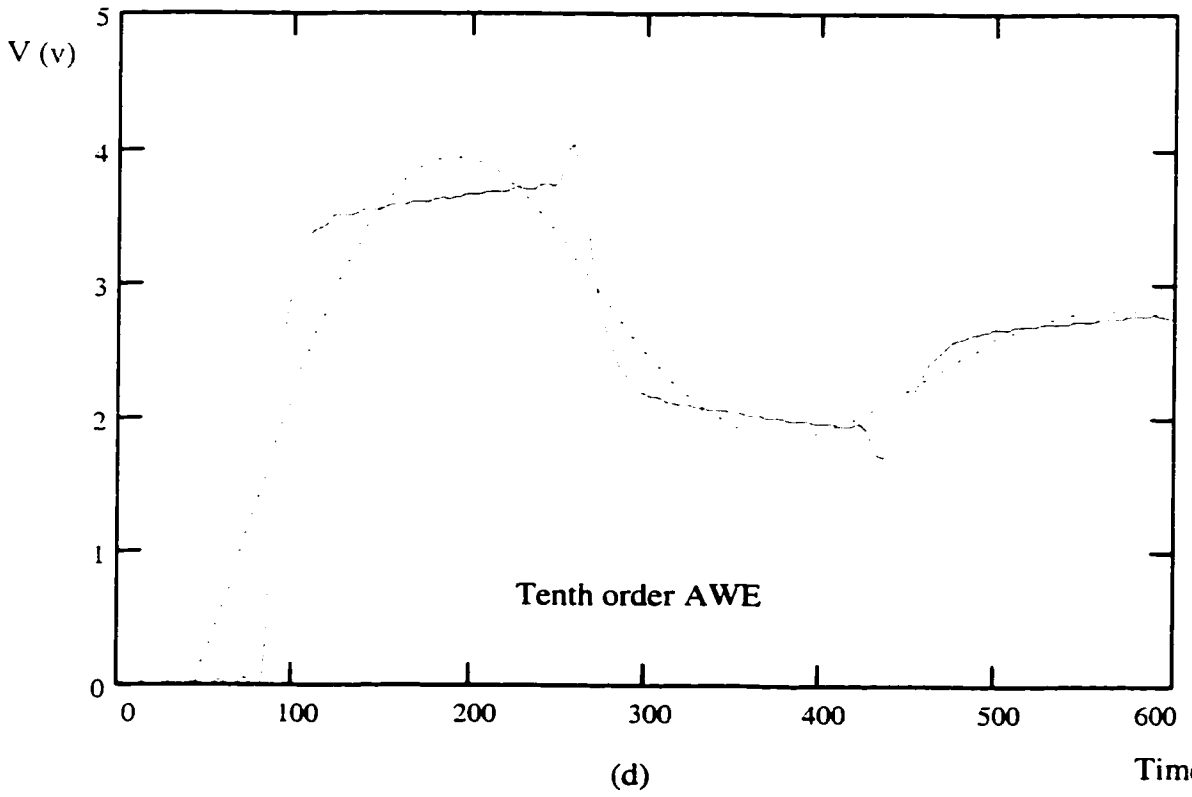
(a)


V (v)
5
4
3
2
1
0
0
100
200
300
400
500
600
Time (ps)
Sixth order AWE

(b)


V (v)
5
4
3
2
1
0
0
100
200
300
400
500
600
Time (ps)
Eighth order AWE

(c)


V (v)
5
4
3
2
1
0
0
100
200
300
400
500
600
Time (ps)
Tenth order AWE

(d)

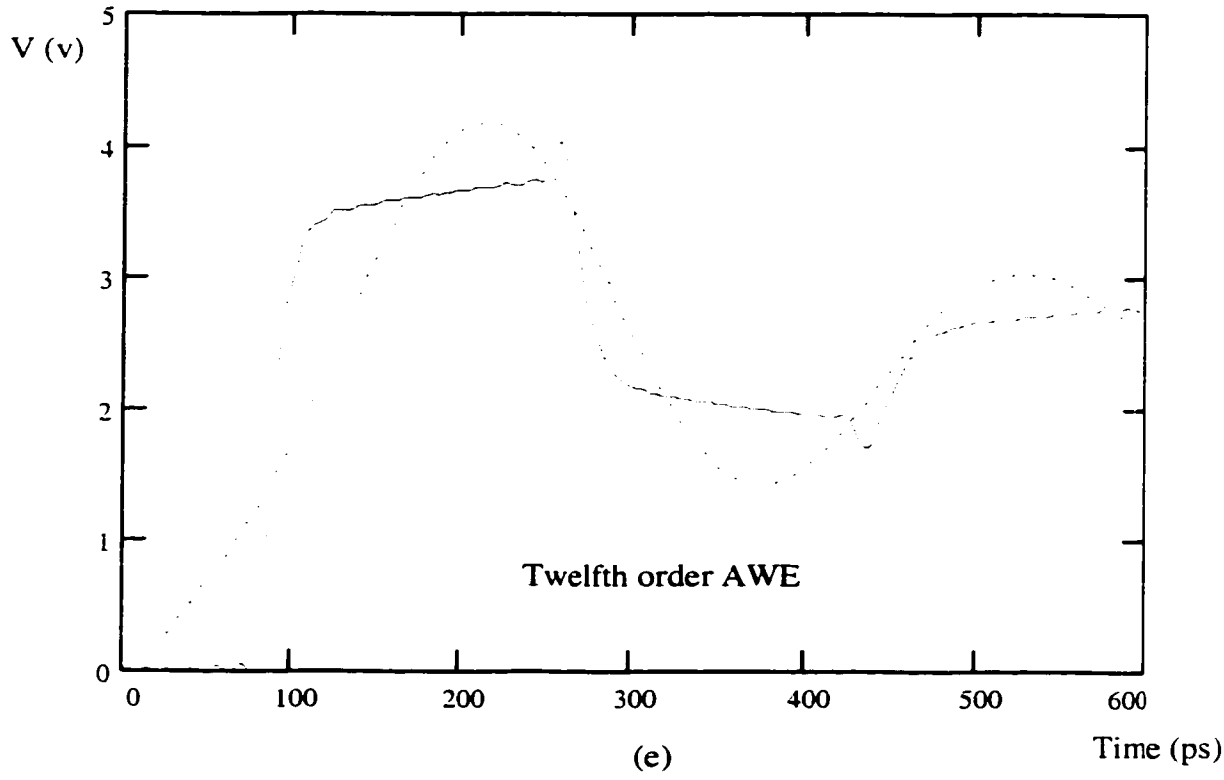


Figure 3.12. Comparison between the AWE approximations and SPICE for the output node of the circuit shown in Figure 3.11. The AWE simulations are based on approximation orders of (a) four, (b) six, (c) eight, (d) ten, (e) twelve. The SPICE simulations are represented by a solid line while AWE is represented by a dotted line.

Table 3.1 The poles determined by the AWE approximations used in Figure 3.12 as compared to the exact poles of the circuit shown in Figure 3.11.

Order	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8
Fourth	-0.004186 +0.01661i	-0.004186 -0.01661i	-0.001684 +0.04508i	-0.001684 -0.04508i	-	-	-	-
Sixth	-0.004181 +0.01661i	-0.004181 -0.01661i	-0.004623 +0.05161i	-0.004623 -0.05161i	0.000004 +0.0717i	0.000004 -0.0717i	-	-
Eighth	-0.004181 +0.01660i	-0.004181 -0.01660i	-0.004177 +0.05129i	-0.004177 -0.05129i	-0.009441 +0.08742i	-0.009441 -0.08742i	0.006055 +0.09525i	0.006055 -0.09525i
Tenth	-0.004180 +0.01661i	-0.004180 -0.01661i	0.001275 +0.03849i	0.001275 -0.03849i	-0.026727 +0.03786i	-0.026727 -0.03786i	0.038994 +0.03453i	0.038994 -0.03453i
Twelfth	-0.004462	0.01281	-0.004369 +0.01665i	-0.004369 -0.01665i	-0.002391 +0.02429i	-0.002391 -0.02429i	0.01574 +0.02058i	0.01574 -0.02058i
Exact	-0.004181 +0.01660i	-0.004181 -0.01660i	-0.004181 +0.05129i	-0.004181 +0.05129i	-0.004201 +0.08626i	-0.004201 -0.08626i	-0.004219 +0.01217i	-0.004219 -0.01217i

Table 3.2 Comparison between the rise times determined by the AWE approximations used in Figure 3.12 to the exact rise time of the circuit shown in Figure 3.11 simulated with SPICE.

Order	SPICE Rise Time	AWE Rise Time	Percent Error
Fourth	9	50	455%
Sixth	9	28	211%
Eighth	9	24	167%
Tenth	9	75	733%
Twelfth	9	120	1233%

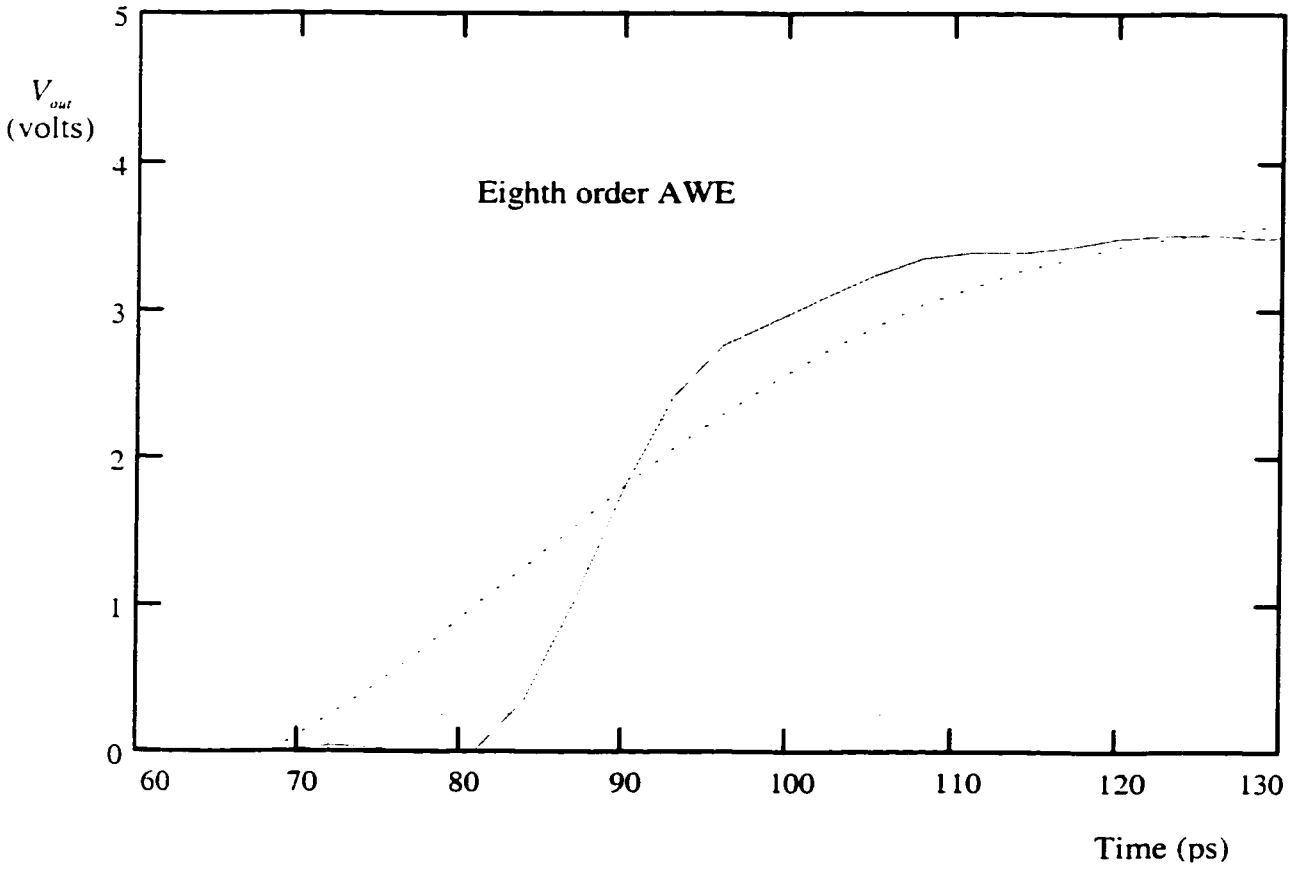


Figure 3.13. Comparison between an AWE eighth order approximation and the SPICE simulation for the output node of the circuit shown in Figure 3.11. The SPICE simulation is represented by a solid line while AWE is represented by a dotted line.

C. Computational Time

Computationally, the AWE method calculates a different set of poles for each node in a connected *RLC* network. It is well known that the transfer functions at different nodes of a connected *RLC* network have a common denominator, *i.e.*, the

RLC circuit has one characteristic equation [135]. This aspect implies that a common set of poles can be used when calculating the transient response at every node of an *RLC* network. The AWE method, however, does not exploit this feature. Such inefficiency can contribute significantly to the overall computational run time of programs using AWE if the transient response is required at many different nodes within the same *RLC* circuit (which is a common requirement in large VLSI circuits). The moment shifting approach introduced in [89] permits the use of a common set of poles for all of the nodes of an *RLC* network. This approach simply uses a higher order set of $2q$ moments rather than using the first $2q$ moments as given by (3.23) since any $2q$ equations can be used to determine an approximation of order q. Thus, the set of q equations used by moment shifting are given by

$$\begin{aligned}
-\left(\frac{k_1}{p_1^{r+1}}+\frac{k_2}{p_2^{r+1}}+\ldots.+\frac{k_q}{p_q^{r+1}}\right) &= m_r,\\
-\left(\frac{k_1}{p_1^{r+2}}+\frac{k_2}{p_2^{r+2}}+\ldots.+\frac{k_q}{p_q^{r+2}}\right) &= m_{r+1},\\
&\vdots\\
-\left(\frac{k_1}{p_1^{r+2q}}+\frac{k_2}{p_2^{r+2q}}+\ldots.+\frac{k_q}{p_q^{r+2q}}\right) &= m_{r+2q-1},
\end{aligned} \tag{3.52}$$

where r determines the amount of shifting of the moments with $r = 0$ representing no moment shifting as in (3.23). Moment shifting improves the accuracy of the dominant poles determined by AWE. To better understand the reason for this improved accuracy, note that the effect of the dominant poles is amplified in higher order moments since higher pole powers diminishes the effect of poles with larger magnitudes. Inaccuracies in dominant poles are usually due to the effect of

undesirable larger poles which affect the values of the individual moments. For example, assume that only one pole is required to approximate a circuit. Assume also that all of the other poles of this circuit are at least twice the value of this dominant pole. If moment shifting is not applied, the value of this pole can be calculated as m_0/m_1 according to (3.27). If moment shifting is used with $r = 10$, the value of this pole is given by m_{10}/m_{11}. Note that the values of the larger poles have been diminished by at least a factor of 1024 as compared to the dominant pole. Thus, the tenth and eleventh moments are approximately given by

$$m_{10} \approx -\frac{k_1}{p_1^{10}},$$
$$m_{11} \approx -\frac{k_1}{p_1^{11}}. \tag{3.53}$$

The value of m_{10}/m_{11} represents the dominant pole with much higher accuracy than m_0/m_1 since the poles with larger magnitude contribute significantly to the values of m_0 and m_1. For example, the *RC* circuit shown in Figure 3.5 of section 3.2.1 has a first dominant pole with the value 0.021692. The ratio m_0/m_1 is equal to 0.018713, m_2/m_3 is equal to 0.021662, and m_{10}/m_{11} is equal to 0.21692. Thus, moment shifting can be used to find several dominant poles with high accuracy. The higher accuracy usually improves the stability of the approximations since the exact poles of any passive circuit are guaranteed stable. To determine a common set of poles for an *RLC* circuit, the moment shifting technique is applied to the node closest to the input since the signal at this node is rich in harmonics which guarantees the calculation of the largest number of poles with high accuracy [89]. Alternatively, the residues of the poles at

this node do not decrease rapidly with larger magnitudes of poles guaranteeing that the effect of these poles remains significant with higher number of moments. The residues still must be calculated at each node where the transient response is required since the residues are different for different nodes.

Despite the increased accuracy of the first few dominant poles determined by the moment shifting technique, moment shifting has several disadvantages. The same characteristic of higher moments that improves the accuracy of the dominant poles by diminishing the effect of larger poles implies that an even smaller number of poles can be found when moment shifting is used. In addition, using the node closest to the input is not always a reliable technique to determine the common set of poles for an entire *RLC* circuit. According to this discussion, moment shifting is primarily useful with *RC* circuits (that exhibit monotone responses) since these circuits require few dominant poles with high accuracy. For more complicated signals such as may occur in *RLC* circuits, moment shifting actually degrades the already poor accuracy of AWE due to numerical instabilities. The algorithm and method presented in Chapter 12 determines the *exact* common denominator of an *RLC* circuit, thereby significantly reducing the computational complexity as compared to AWE. Also, the AWE method has to solve a set of q linear equations twice at each output node to determine the characteristic equation and the residues, thereby exhibiting a computational complexity proportional to q^3. The algorithm presented in Chapter 12 determines the characteristic equation directly and determines the residues by direct substitution into polynomial expressions. This new method is compared to AWE in Appendix F.

Chapter 4 MOSFET Current-Voltage Characteristics

A CMOS VLSI logic circuit is primarily composed of MOSFET devices connected by interconnect. Thus, an understanding of the operation of MOSFET transistors is important. Also, accurate characterization of the behavior of a MOSFET transistor is crucial for VLSI circuit analysis and design. In this chapter, the behavior of MOSFET transistors is briefly reviewed. The basic long channel behavior of a MOSFET device is first discussed. This long channel behavior is followed by a discussion of the alpha power law model that characterizes the current-voltage characteristics of a MOSFET transistor in deep submicrometer technologies [108].

4.1 Basic Theory of Operation of a MOSFET

The basic structure of an N-channel metal oxide semiconductor field effect transistor (MOSFET) is shown in Figure 4.1 [1]-[3]. The four terminal device consists of a p-type substrate, in which two n^+ diffusion regions, the drain and source, are formed. The surface of the substrate region between the drain and source is covered with a thin silicon dioxide (SiO_2) layer, which acts as an insulator. The metal gate (typically polysilicon acting as a conductor) is deposited on top of this thin gate dielectric. Note that the device is completely symmetric with respect to the drain and

source regions. The different roles played by the two regions are defined by the applied terminal voltages. The current that flows between the drain and source passes through the substrate area between the two n^+ regions. This path is called the channel. The channel has a length L in the direction of the current flow and a width W in the transverse direction to the current flow as shown in Figure 4.1.

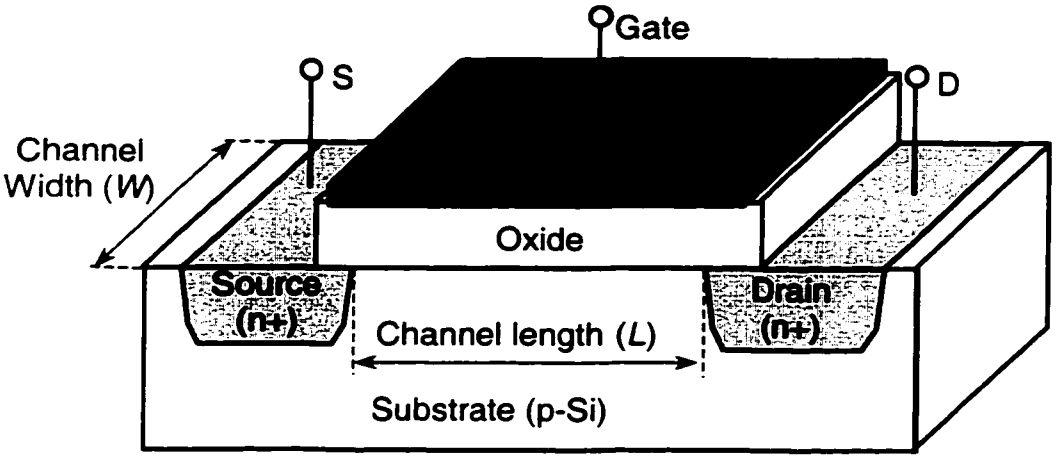


Figure 4.1. The physical structure of an N-channel enhancement-type MOSFET device.

To function correctly, the substrate of an N-channel device is connected to the lowest voltage in the circuit. The substrate is the fourth terminal of a MOSFET device and is usually called the body. The body in an N-channel device is commonly connected to the lowest potential in a digital CMOS circuit, typically ground. If a positive voltage is applied to the drain while the source is connected to ground, the direction of the current flow is from the drain to the source since the current flows from a positive voltage to a negative voltage. The magnitude of the current that flows from the drain to the source is determined by the gate voltage. The configuration of

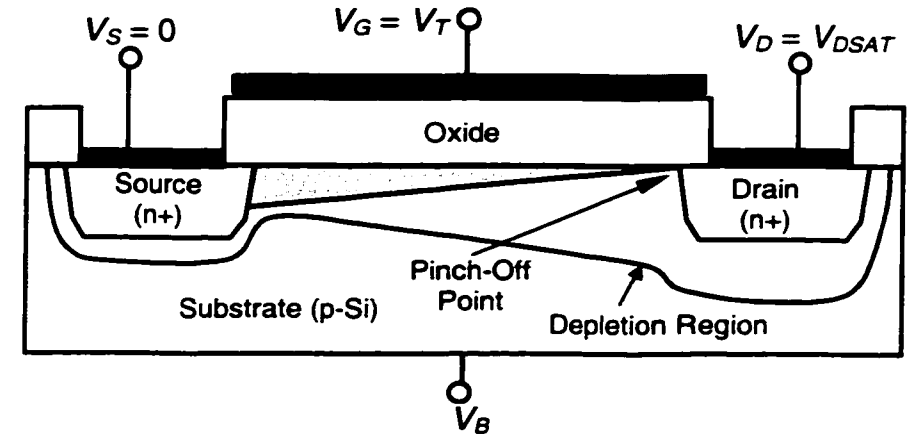
the gate and the substrate with an insulator in between forms a capacitor. If zero potential is applied to the gate, the channel contains positive charges since the substrate is p-type silicon. If a negative voltage is applied at the gate, the channel will attract even more positive charges induced by the electric field. If a small positive voltage is applied to the gate, some negative charges will be induced by the electric field. However, the net charge in the channel remains positive since there are more positive charges within the p-type substrate than the induced negative charges. In all of these cases, no current can flow from the drain to the source since the P-N junction at the drain is reversed biased. This reverse bias is due to the p-type substrate being connected to the most negative voltage in the circuit. Thus, for these gate voltages the transistor is said to be off. If a large positive voltage is applied to the gate of an N-channel transistor such that the negative charges induced by the electric field are greater than the positive charges of the p-type substrate, an N-type channel is formed between the drain and the source. The gate voltage required to cause *channel inversion* is termed the threshold voltage V_{Tn} and is always positive for an enhancement mode N-type MOSFET. The magnitude of the threshold voltage depends upon several parameters such as the gate oxide thickness, the substrate doping, and the body bias. Once an N-channel is formed between the two n^+ regions, current can flow between the drain and source and the transistor is said to be on. A MOSFET transistor is described as a majority carrier device since the current is carried by the majority charge type.

The amount of current that flows between the drain and source depends on several factors. The greater the gate voltage as compared to the threshold voltage, the

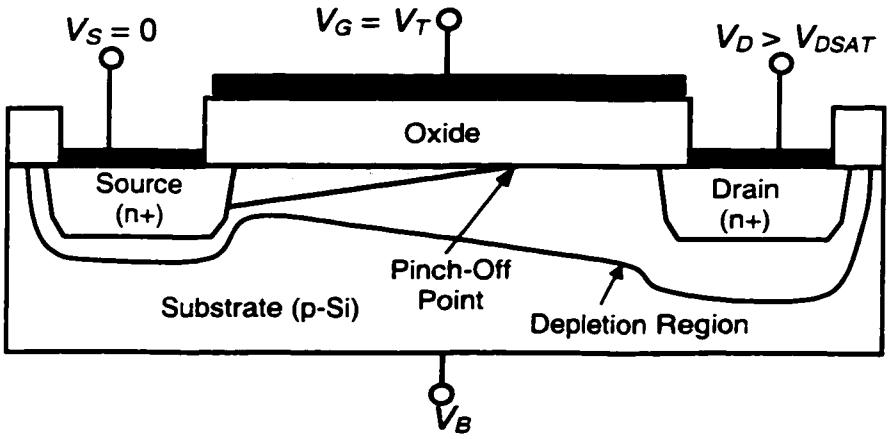
higher the current that will flow in the transistor channel due to the increased negative charges induced in the channel that can pass the current between the drain and the source. Also, as the drain-to-source voltage increases, the amount of current that flows between the drain and the source increases due to the higher electric field in the direction of the current flow. The increase in current is predominantly linear with increasing drain-to-source voltage V_{DS} and thus the transistor is said to operate in the *linear region*. However, if the drain voltage is increased further with respect to the gate voltage, the induced negative charge in the channel becomes zero in the vicinity of the drain due to the reversal of the direction of the electric field. The point at which the negative charge in the channel near the drain becomes zero is called the *pinch-off* point and is shown in Figure 4.2 (a). The drain voltage at which the pinch-off point occurs is termed V_{DSAT}, the magnitude of which depends upon the applied gate voltage. Beyond the pinch-off point where the drain voltage becomes more positive with respect to the gate voltage (or $V_D > V_{DSAT}$), a depleted surface region (called the depletion region) forms adjacent to the drain and grows towards the source with increasing drain voltages as shown in Figure 4.2 (b). This operational mode of a MOSFET transistor is called the *saturation mode* and the region of operation is the *saturation region*. For a MOSFET operating in the saturation region, the effective channel length is reduced while the voltage at the channel end close to the drain remains essentially constant and equal to V_{DSAT}. Note that the pinch-off (depleted) section of the channel absorbs most of the excess voltage drop, $V_D - V_{DSAT}$, and a high electric field forms between the end of the channel and the drain boundary. Electrons moving from the source to the channel end close to the drain are injected into the

depletion region and accelerated by this high electric field, usually reaching the limit of the drift velocity. The current flowing between the source and drain in the saturation region is almost independent of the drain-to-source voltage and is almost equal to the current in the case where $V_D = V_{DSAT}$. The gate voltage is usually measured with reference to the source voltage when characterizing the behavior of a MOSFET transistor and is termed the gate-to-source voltage, V_{GS}. A plot of the dependence of the drain-to-source current I_{DS} on the gate-to-source voltage V_{GS} is shown in Figure 4.3. A plot of the dependence of the drain-to-source current I_{DS} on the drain-to-source voltage V_{DS} is shown in Figure 4.4 for different values of V_{GS}. The dotted line that appears in Figure 4.4 separates the linear region from the saturation region in a MOSFET transistor.

The channel width and length also affect the magnitude of the current flowing through a MOSFET transistor. As the length of the channel increases, the current between the drain and source decreases since the electric field caused by V_{DS} across the channel length decreases. As the width of the channel increases, the amount of current flowing between the drain and source increases since a wider channel produces more carriers to conduct current. An alternative perspective to better understanding the dependence of a MOSFET transistor on the channel width and length is to view the channel as a resistance. The lower the resistance of the channel, the more current that can flow in a MOSFET transistor.



(a)



(b)

Figure 4.2. A MOSFET transistor operating in the saturation region a) at the pinch-off point. b) beyond the pinch-off point.

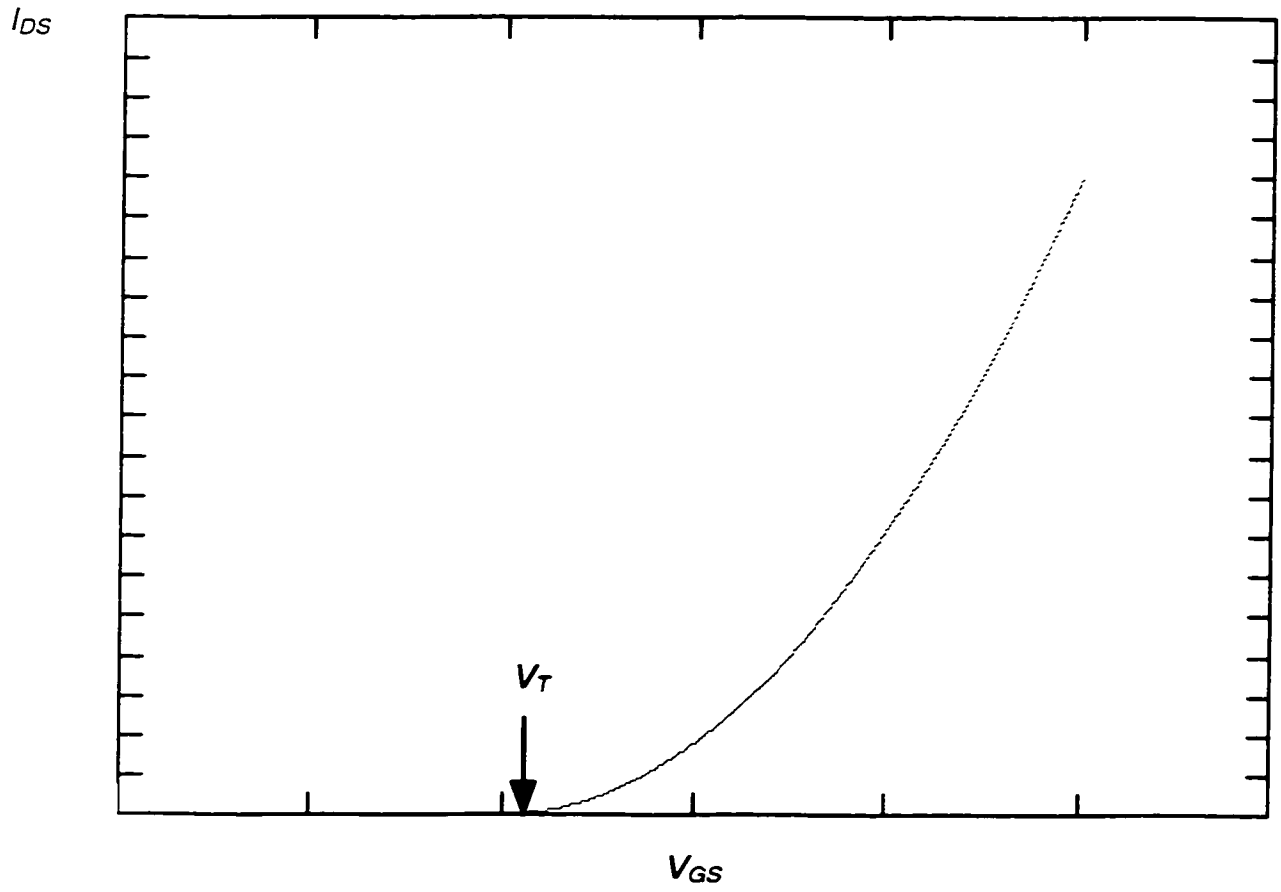


Figure 4.3. I_{DS} versus V_{GS} in an enhancement N-type MOSFET.

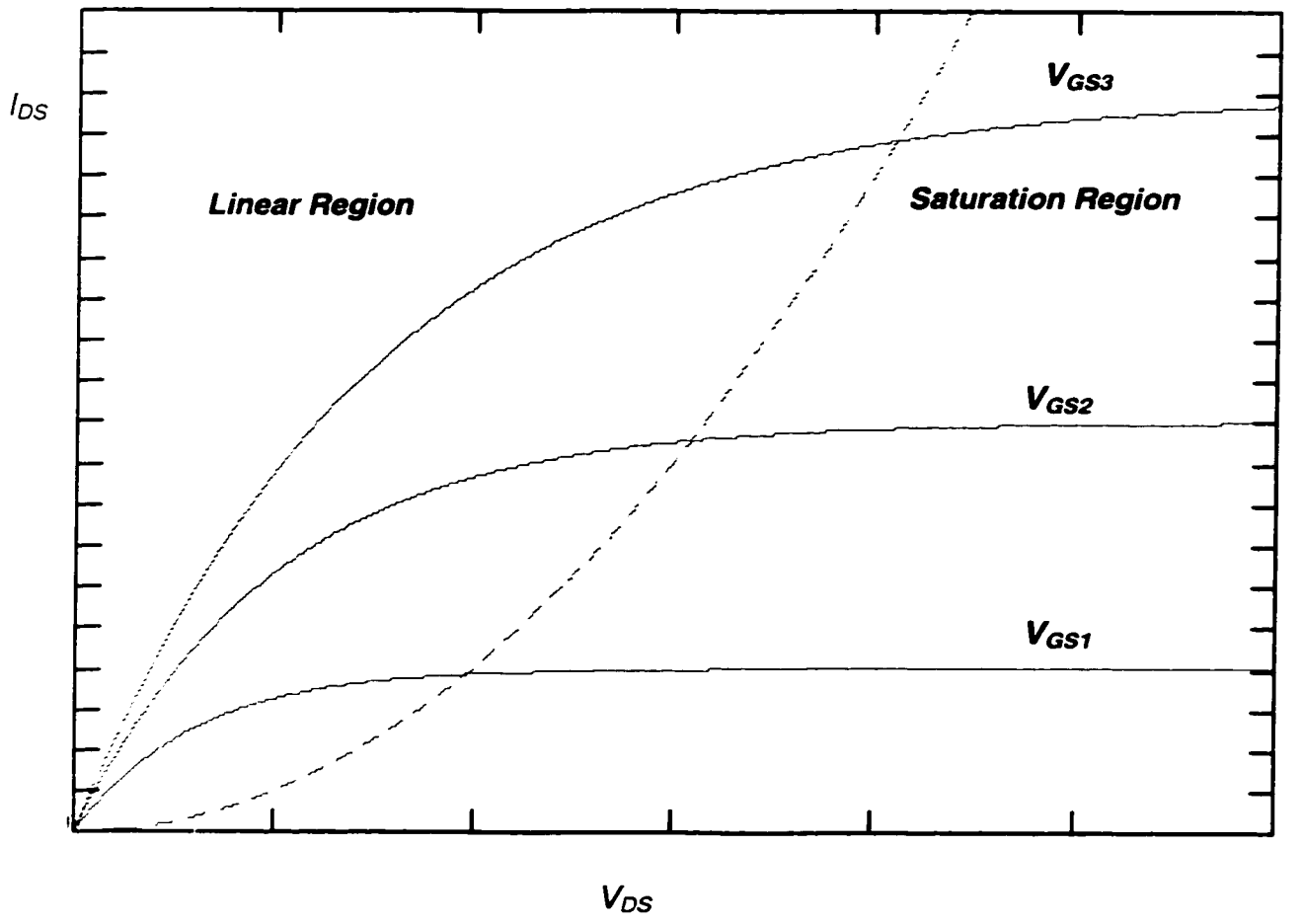


Figure 4.4. I_{DS} versus V_{DS} for several values of V_{GS} in an enhancement N-type MOSFET.

A P-type enhancement MOSFET transistor is similar in structure to an N-type MOSFET but with the body composed of an N-type silicon and the drain and source regions made of p^+ regions. The operational principles of a P-type enhancement MOSFET transistor is precisely the same as the principles previously discussed for an

N-type MOSFET transistor except that the voltage biases are inverted to conduct current flowing from the source to the drain. Also, the body of the P-channel device is connected to the most positive voltage in the circuit for correct operation (*i.e.*, V_{DD}) and the threshold voltage of a P-channel enhancement transistor is negative. The current is carried by positively charged holes in a P-type transistor which has a lower mobility as compared to electrons in an N-type transistor. Thus, a P-type MOSFET produces about one and half to two times less current than an identical N-type transistor under the same bias conditions.

4.2 Alpha Power Law Model for Short Channel Devices

One of the popular analytical models to characterize the behavior of a short channel MOSFET transistor has been introduced by Sakurai in [36]. According to the alpha power law model [36], the current in the saturation region is given by

$$I_{DS} = P_C \frac{W_d}{L_d}(V_{GS} - V_T)^{\alpha}. \tag{4.1}$$

and in the linear region, the current is

$$I_{DS} = \frac{P_C}{P_V}\frac{W_d}{L_d}(V_{GS} - V_T)^{\frac{\alpha}{2}} V_{DS} = \frac{V_{DS}}{R_{tr}}. \tag{4.2}$$

where

$$R_{tr} = \frac{1}{\frac{P_C}{P_V}\frac{W_d}{L_d}(V_{GS} - V_T)^{\frac{\alpha}{2}}}. \tag{4.3}$$

The switching point from the saturation region to the linear region of operation can be found by equating (4.1) and (4.2) and occurs when

$$V_{DS} \leq P_V (V_{GS} - V_T)^{\frac{\alpha}{2}}. \tag{4.4}$$

P_C and P_V are technology dependent constants that characterize the drive current of the transistor in the saturation and linear regions. W_d and L_d are the geometric width and length, respectively, of the device and V_T is the threshold voltage of the device. α is a constant between one and two. α is equal to two for long channel devices and decreases in magnitude to one due to velocity saturation as the channel length becomes shorter [36]. This model is accurate for deep submicrometer devices and is used in subsequent sections to characterize the MOSFET transistors.

Chapter 5 Figures of Merit to Characterize the Importance of On-Chip Inductance in Single Lines

It has become well accepted that interconnect delay dominates gate delay in current deep submicrometer VLSI circuits [1]-[3]. With the continuous scaling of technology and increased die area, this situation is expected to become worse. In order to properly design complex circuits, more accurate interconnect models and signal propagation characterization are required. Historically, interconnect has been modeled as a single lumped capacitance in the analysis of the performance of on-chip interconnects. With the scaling of technology and increased chip sizes, the cross-sectional area of wires has been scaled down while interconnect length has increased. The resistance of the interconnect has therefore become significant, requiring the use of more accurate *RC* delay models. At first interconnect was modeled as a lumped *RC* circuit. To further improve accuracy, the interconnect has been modeled as a distributed *RC* circuit (multiple T or Π sections) for those nets requiring more accurate delay models. A well known method used to determine which nets require more accurate delay models is to compare the driver resistance R_{tr} and the load capacitance C_L to the total resistance and capacitance of the interconnect line, R_t and

C_t [23], [33]. Typically, those nets that require more accurate *RC* models are longer, more highly resistive nets.

Currently, inductance is becoming more important with faster on-chip rise times and longer wire lengths. Wide wires are frequently encountered in clock distribution networks and in upper metal layers. These wires are low resistive lines that can exhibit significant inductive effects. Furthermore, performance requirements are pushing the introduction of new materials for low resistance interconnect [57], [61]. In the limiting case, high temperature superconductors may become commercially available [69]. With these trends it is becoming crucial to be able to determine which nets within a high speed VLSI circuit exhibit prominent inductive effects.

In this chapter a closed form solution for the output signal of a CMOS inverter driving an *RLC* transmission line is presented. This solution is based on the alpha power law for deep submicrometer technologies. Two figures of merit are presented that are useful for determining if a section of interconnect should be modeled as either an *RLC* or an *RC* impedance. The damping factor of a lumped *RLC* circuit is shown to be a useful criterion. The second useful figure of merit considered in this chapter is the ratio of the rise time of the input signal at the driver of an interconnect line to the time of flight of the signals across the line. AS/X circuit simulations of an *RLC* transmission line and a five section *RC* Π circuit based on a 0.25 μm IBM CMOS technology are used to quantify and determine the relative accuracy of an *RC* model. One primary result of this study is evidence demonstrating that a range for the length

of the interconnect exists for which inductance effects are prominent. Furthermore, it is shown that under certain conditions, inductance effects are negligible despite the length of the section of interconnect

This chapter is organized as follows. The equations describing the signal behavior of an *RLC* transmission line are provided in section 5.1, along with a closed form solution for the output signal of a CMOS inverter driving an *RLC* transmission line based on the alpha power law [108] for deep submicrometer (DSM) technologies. The damping factor of a lumped *RLC* circuit and the rise time of the input signal at the driver of the interconnect are used to derive two figures of merit that describe the relative significance of inductance of a local interconnect line. These figures of merit are presented in section 5.1. In section 5.2, the two figures of merit described in section 5.1 are combined to define a range of the length of interconnect at which inductance become important. AS/X[1] circuit simulations [128] calibrated for an advanced 0.25 μm CMOS technology are also compared in this section to the analytical results presented in the previous section. Finally, some conclusions are offered in section 5.3.

[1] AS/X is a dynamic circuit simulator developed and used by IBM. AS/X is similar to SPICE but has a specific emphasis on transmission line networks and uses the ASTAP language for the input files.

5.1 Theoretical Analysis of Inductance Effects in RLC Interconnect

The behavior of waves traveling across an *RLC* transmission line is explained in this section. The attenuation that a wave exhibits as it travels along a line is compared to the damping factor of a lumped *RLC* circuit representation of the same line. It is shown in subsection 5.1.1 that the damping factor of a lumped *RLC* circuit representation of a line can be a useful criterion to determine the relative importance of inductance. A closed form solution for the output voltage of a CMOS gate driving an *RLC* transmission line is presented in subsection 5.1.2. A closed form solution for a CMOS gate driving a single *RC* T section representation of the line is also presented. Both solutions are compared for different values of attenuation to further investigate the damping factor as a useful figure of merit. The two solutions are also compared with different input transition times which leads to the second figure of merit in this chapter. It is shown that the ratio of the transition time of the signal at the input of a CMOS gate driving an interconnect line to the time of flight of a wave across the interconnect is a second useful figure of merit to determine the relative importance of inductance to a specific line.

5.1.1 Damping Factor

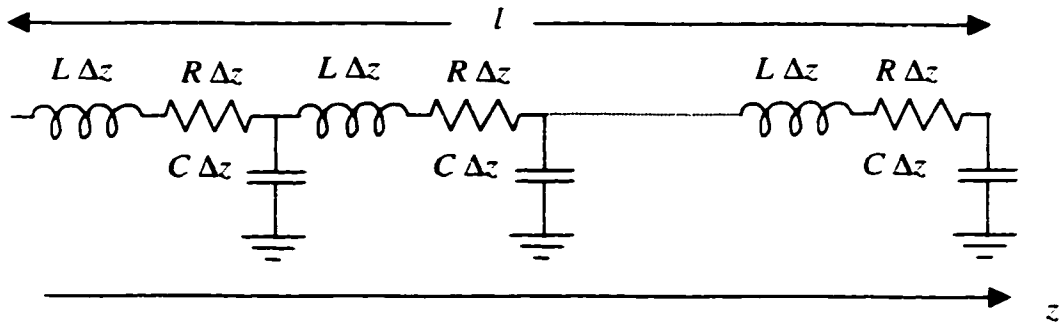


Figure 5.1. *RLC* transmission line model of an interconnect line.

As discussed in Chapter 2. A single interconnect line can be modeled as an *RLC* transmission line as shown in Figure 5.1, where R, L, and C are the resistance, inductance, and capacitance per unit length, respectively, and Δz is an incremental length segment of the line. For an *RLC* transmission line driven by a sinusoidal input $Re\{e^{j\omega t}\}$, the voltage across the transmission line [70]-[72] is

$$V(z,t) = \mathrm{Re}\{V_1 e^{(j\omega t - \gamma z)} + V_2 e^{(j\omega t + \gamma z)}\}. \tag{5.1}$$

The solution of $V(z,t)$ is the sum of two traveling waves, one in the positive z direction and the other in the negative z direction. V_1 is the summation of the original voltage wave and all the reflected voltage waves in the positive z direction. V_2 is the summation of all the reflected voltage waves traveling in the negative z direction. The propagation constant of the transmission line, γ, describes the characteristics of the wave propagation across the line. For an *RLC* transmission line, the propagation constant is complex [70]-[72] and is

$$\gamma = \alpha + j\beta, \tag{5.2}$$

where the real part α is the attenuation constant of the waves as the waves propagate across the line, and the imaginary part β is the phase constant which determines the speed of propagation of the waves across the line. Substituting (5.2) into (5.1), the real part of the voltage is given by

$$V(z,t) = V_1 e^{-\alpha z}\cos(\omega t - \beta z) + V_2 e^{\alpha z}\cos(\omega t + \beta z). \tag{5.3}$$

The attenuation of a traveling wave is exponentially dependent on the distance traveled by the wave, and both the attenuation and speed of the wave are frequency dependent.

The speed of propagation of the wave across the line [70]-[72] is

$$v = \frac{\omega}{\beta}. \tag{5.4}$$

The sequence of events that constitutes a transient response for an input wave begins with a portion of the wave launched into the line from the source end. This wave propagates across the line towards the load with a speed determined by (5.4). The wave attenuates as it travels across the lossy line. If a mismatch exists between the characteristic impedance of the transmission line and the load impedance, a reflected wave is generated and propagates towards the source to compensate for the mismatch. This reflected wave is further attenuated as it moves towards the source. The reflection process is repeated infinitely, but practically, the signal can be considered to be at steady state when the reflections become negligible. As the rate of attenuation increases, the reflections become negligible faster. This behavior can be explained by noting that the waves are multiplied by a factor of $e^{-2\alpha}$ for a round trip across the line,

where l is the length of the line. This aspect means that as the line becomes longer, the effect of the reflections becomes less and the line behaves as an *RC* line. This same behavior occurs if the resistance of the line increases, increasing the attenuation constant. The attenuation constant α of an *RLC* transmission line can be derived from the basic equations and is

$$\alpha = \omega\sqrt{LC}\sqrt{\frac{1}{2}\left(\sqrt{\left(1+\left(\frac{R}{\omega L}\right)^2\right)}-1\right)}. \tag{5.5}$$

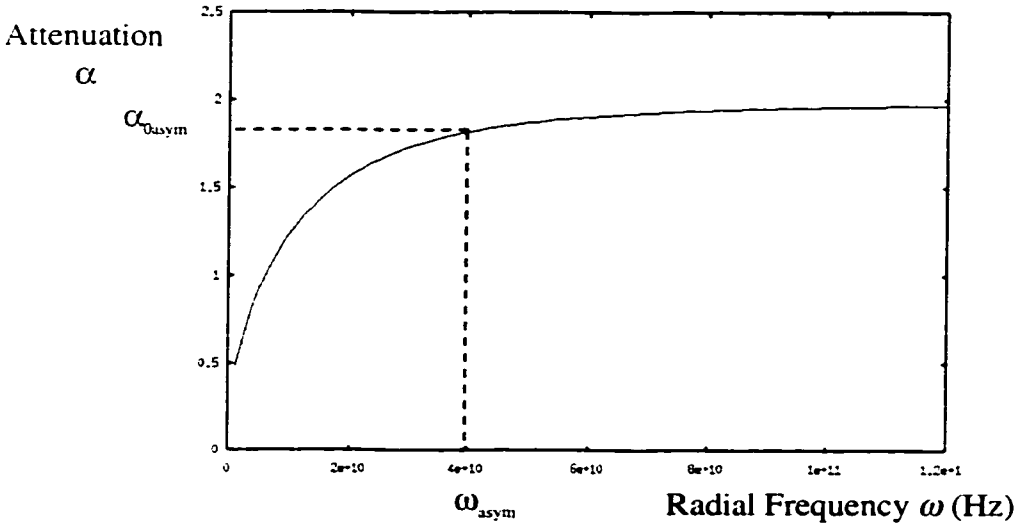


Figure 5.2. The attenuation constant α versus the radial frequency.

The attenuation constant as a function of frequency is plotted in Figure 5.2 with $L=10^{-8}$ *H/cm*, $R=400$ *Ω/cm*, *and* $C=10^{-12}$ *F/cm* [48]. The attenuation constant is shown to saturate with increasing frequency to an asymptotic value given by

$$\alpha_{asym} = \frac{R}{2}\sqrt{\frac{C}{L}}, \tag{5.6}$$

and the radial frequency at which this saturation begins is given by

$$\omega_{asym} \approx \frac{R}{L}. \tag{5.7}$$

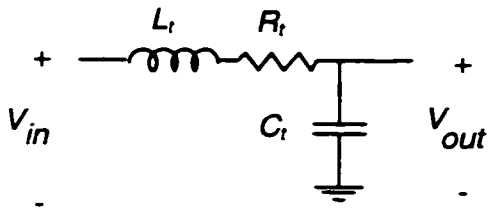


Figure 5.3. Simple lumped *RLC* circuit model of an interconnect line.

This analysis of an *RLC* transmission line is compared to the analysis of a lumped *RLC* circuit (see Figure 5.3 for a lumped *RLC* circuit). The interconnect is modeled as a single section *RLC* circuit with $R_t=Rl$, $L_t=Ll$, and $C_t=Cl$. The poles of this circuit are

$$p_{1,2} = \omega_0[-\xi \pm \sqrt{(\xi^2 - 1)}], \tag{5.8}$$

and the damping factor ξ is

$$\xi = \frac{Rl}{2}\sqrt{\frac{C}{L}} = l\alpha_{asym}. \tag{5.9}$$

As (5.8) implies, if ξ is greater than one, the poles are real and the effect of the inductance on the circuit is small. The greater the value of ξ, the more accurate the *RC* model becomes. On the other hand, as ξ becomes less than one, the poles become complex and oscillations occur. In that case, the inductance cannot be neglected. The strong analogy between the lumped *RLC* circuit and the *RLC* transmission line is illustrated by (5.9). This relationship is physically intuitive, since ξ represents the degree of attenuation the wave suffers as it propagates a distance equal to the length

of the line. As this attenuation increases, the effects of the reflections decrease and the *RC* model becomes more accurate. Therefore ξ is a useful figure of merit that anticipates the importance of considering inductance in a particular interconnect line. This criterion is the same result as described in [49]-[51] but is derived in a different way. Note that if ξ in (5.9) is squared, this figure of merit becomes a comparison between the time constant *L/R* and the time constant *RC*, which is the same result as described in [46] and [52].

5.1.2 Input Transition Time

The characteristic impedance of an *RLC* transmission line is complex with a portion that is negative and imaginary. Therefore, the characteristic impedance looks like a resistance in series with a capacitance. Thus, the characteristic impedance can be expressed as

$$Z_0 = R_0 - j\frac{1}{\omega C_0}. \tag{5.10}$$

where R_0 and C_0 are given by

$$R_0 = \sqrt{\frac{L}{C}}\sqrt{\frac{1}{2}\left(\sqrt{\left(1+\left(\frac{R}{\omega L}\right)^2\right)}+1\right)}, \tag{5.11}$$

$$C_0 = \frac{1}{\omega\sqrt{\frac{L}{C}}\sqrt{\frac{1}{2}\left(\sqrt{\left(1+\left(\frac{R}{\omega L}\right)^2\right)}-1\right)}}. \tag{5.12}$$

Plots of R_0 and C_0 versus frequency are shown in Figs. 4 and 5, respectively, with $L=10^{-8}$ *H/cm*, $R=400$ *Ω/cm*, *and* $C=10^{-12}$ *F/cm*.

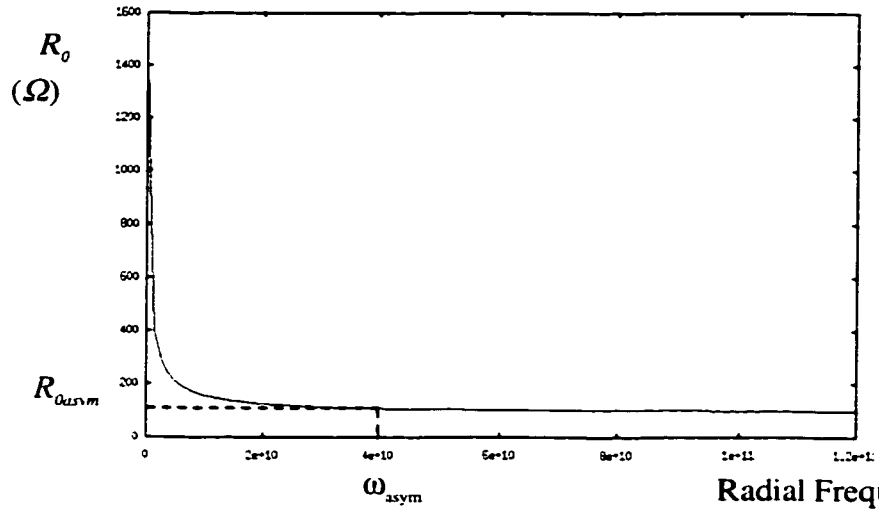


Figure 5.4. Real part of the characteristic impedance of an *RLC* transmission line.

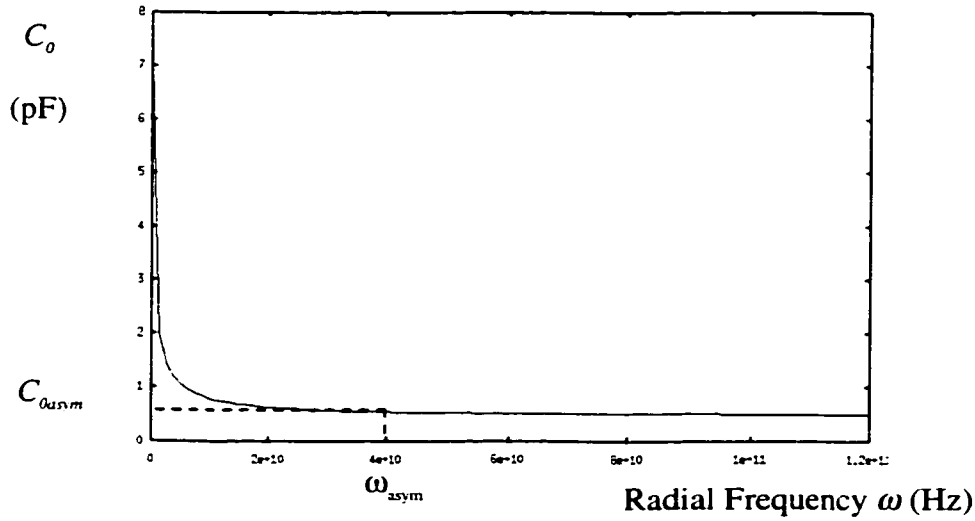


Figure 5.5. Equivalent capacitance of the characteristic impedance of an *RLC* transmission line.

Both R_0 and C_0 saturate to the asymptotic values given by

$$R_{0asym} = \sqrt{\frac{L}{C}}. \tag{5.13}$$

$$C_{0asym} = \frac{2\sqrt{LC}}{R}. \tag{5.14}$$

where the saturation frequency is given by (5.7).

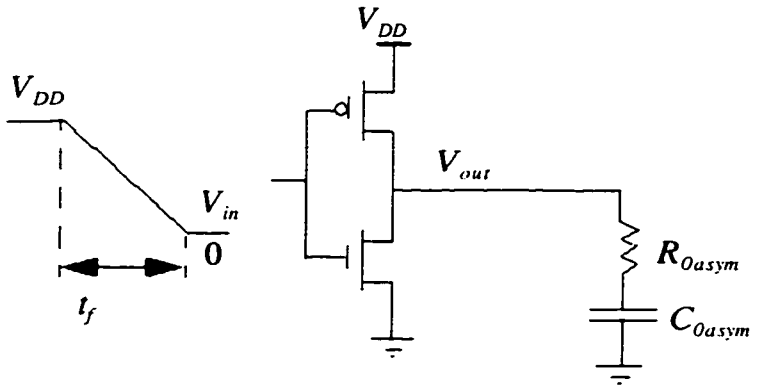


Figure 5.6. A CMOS inverter driving the equivalent characteristic impedance of an *RLC* transmission line.

An *RLC* transmission line driven by a CMOS inverter can be approximated as shown in Figure 5.6, for the period of time, $0<t<2T_0$, where T_0 is the time required for the waves to travel a distance equal to the length of the transmission line. This term is frequently described as the time of flight of the transmission line. The input is assumed to be a ramp with a fall time t_f. Asymptotic values for the characteristic impedance and the attenuation are assumed in the following analysis. The technology used in this analysis is an IBM 0.25 μm CMOS technology with a 2.5 volt power supply. The alpha power law is used to characterize the devices [108]. A pulse is generated at V_{out} for the period of time $0<t<2T_0$, where the time reference is chosen when the input signal reaches $V_{DD}+V_{Tp}$ where V_{Tp} is the threshold voltage of the P-channel devices and for an enhancement mode device is negative. Under the

aforementioned conditions and the assumption that the PMOS transistor is saturated and neglecting the effect of the NMOS transistor, V_{out} is

$$V_{out}(t) = P_{Cp}\frac{W_p}{L_p}V'_{SGp}(t)^{\alpha p}\left[\sqrt{\frac{L}{C}} + \frac{Rt}{2\sqrt{LC}}\right]u(t) = P(t), \tag{5.15}$$

for $0<t<2T_0$, where $u(t)$ is the unit step function and $V'_{SGp}(t) = V_{DD} + V_{Tp} - V_{in}(t)$. P_C is a constant that characterizes the drive current of the transistor in saturation, W and L are the geometric width and length, respectively, of the transistor, and α is a constant between one (strong velocity saturation) and two (weak velocity saturation) [108]. p indicates the PMOS transistor.

The pulse propagates across the transmission line. At the load, the signal is completely reflected assuming an open circuit (or a small load capacitor) at the end of the line. This reflected signal propagates back towards the driver and reaches the driver at a time $t = 2T_0$. After this round trip is completed, the pulse that reaches the source is attenuated by a factor of $e^{-2\alpha l}$ and can be described mathematically by $P(t-2T_0)\,e^{-2\alpha l}$. As long as the transistor is in saturation, the transistor maintains a relatively constant current. Thus, the current reflection coefficient is -1 and consequently the voltage reflection coefficient is 1. Therefore the pulse is multiplied by two. This cycle repeats as long as the transistor is saturated. The complete solution for the period when the transistor remains saturated is

$$V_{out}(t) = P(t) + \sum_{i=1}^{n}\left[2P(t-2nT_0)e^{-2\alpha nl}\right], \tag{5.16}$$

for $2nT_0<t<2(n+1)T_0$. This solution is compared to an RC representation of the line, as shown in Figure 5.7.

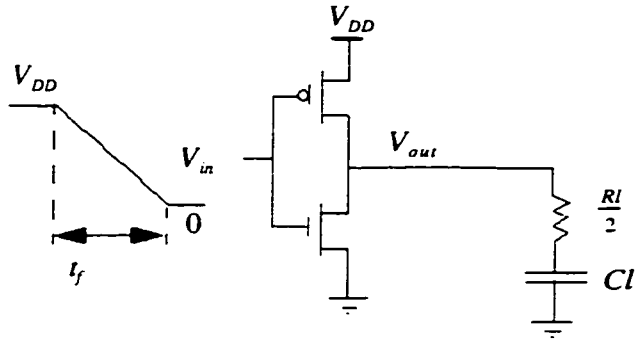


Figure 5.7. A CMOS inverter driving an *RC* approximation of an interconnect line.

The solution for V_{out} based on this model when the PMOS transistor is in saturation is

$$V_{out} = P_{Cp} \frac{W_p}{L_p} V'_{SGp}(t)^{\alpha p} \left[\frac{Rl}{2} + \frac{t}{Cl} \right]. \tag{5.17}$$

The analytical solution in (5.16) is compared with AS/X simulation [128] results for the *RLC* transmission line characterized in Figs. 8 and 9, with $L=10^{-7}$ *H/cm* and $C=10^{-12}$ *F/cm*. The analytical solution agrees with the simulations of an *RLC* transmission line for a wide variety of interconnect resistances and input fall times. As implied by the analytical solution, the output signal follows the changing input signal. The period when the input signal is falling represents the fast rising parts of the response that depend on the transition time of the input signal. Once the input signal is settled, the current provided by the transistor is constant and the output signal changes slowly due to the charging of the equivalent capacitor of the transmission line. This period of time represents the slow rising part of the response that depends upon the value of the equivalent capacitance of the transmission line. Note in Figure 5.8 that as *R* increases,

the slope of those portions of the response increases since the value of the equivalent capacitor decreases, as given by (5.14). It can also be seen that as the resistance of the line increases, the attenuation of the reflections increases as given by (5.9), which makes the *RC* response approach the *RLC* transmission line response.

The output response of the *RLC* transmission line tracks the output response of the *RC* circuit. The points of intersection with the *RC* response can be calculated by equating (5.16) and (5.17), and are given by

$$t_n = \frac{\left[T_0(2K_n - 1) + 4T_0\xi \cdot \frac{dK_n}{d(2\alpha l)} - \frac{RCl^2}{2}\right]}{[1 - \xi(2K_n - 1)]}. \tag{5.18}$$

where K_0=1 and

$$K_n = \frac{e^{-2\alpha l(n+1)} - 1}{e^{-2\alpha l} - 1} \qquad n=1,2... \tag{5.19}$$

The interesting point to note is that those times at which the *RC* response intersects the *RLC* transmission line response are not dependent on the transition time of the input signal. This characteristic can be noticed in Figure 5.9. Thus, as the transition time of the input signal increases, the slope of the fast changing portions of the response decreases, which reduces the width of the slowly varying parts of the response. This behavior makes the response of the *RLC* transmission line appear more continuous. Since the times at which the *RLC* response intersect with the *RC* response are constant, the *RLC* transmission line response approaches the *RC* circuit response.

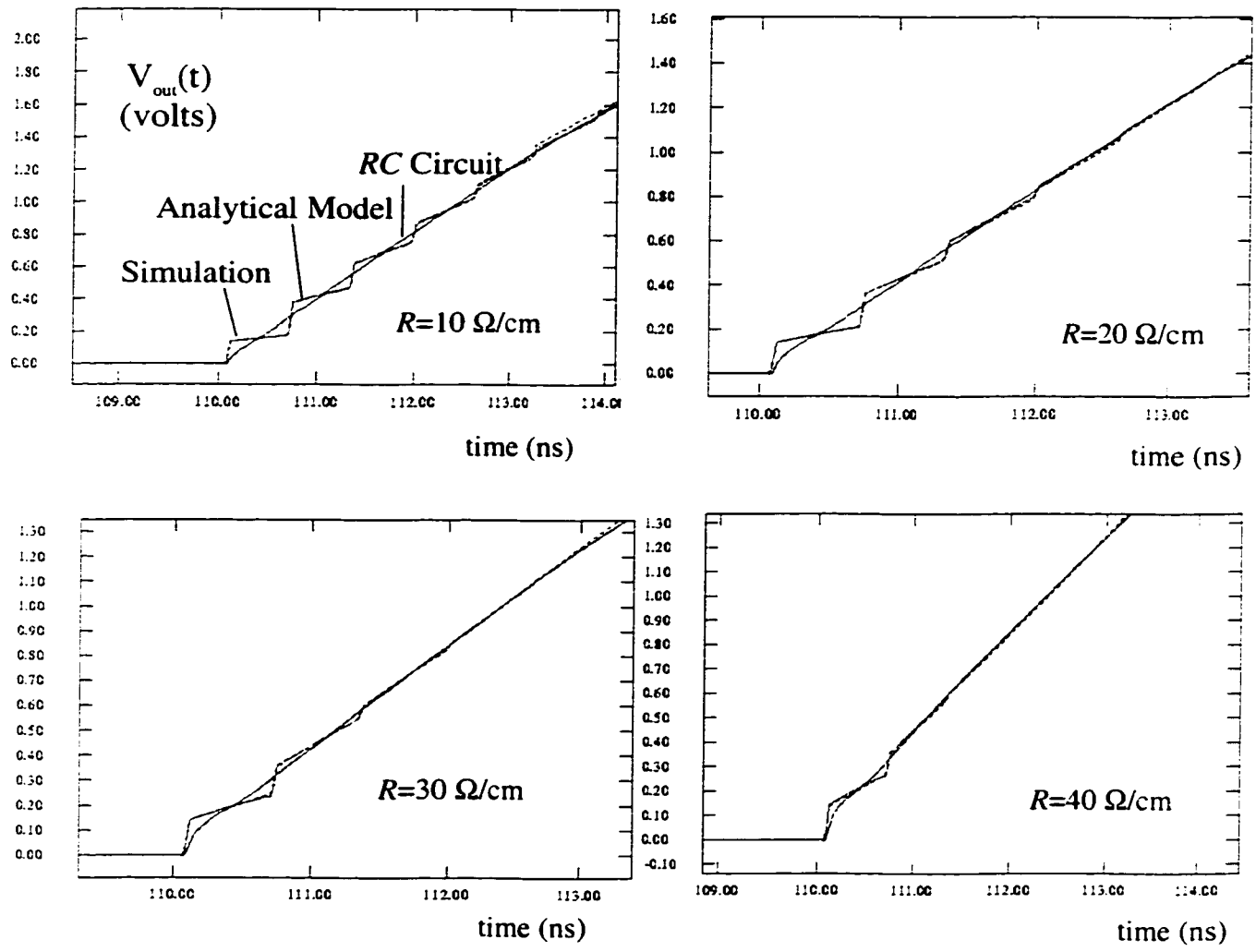


Figure 5.8. Analytical solution in (5.16) compared to AS/X simulations and a five section *RC* circuit. The fall time of the input signal is held constant at 60 ps, while *R* is varied.

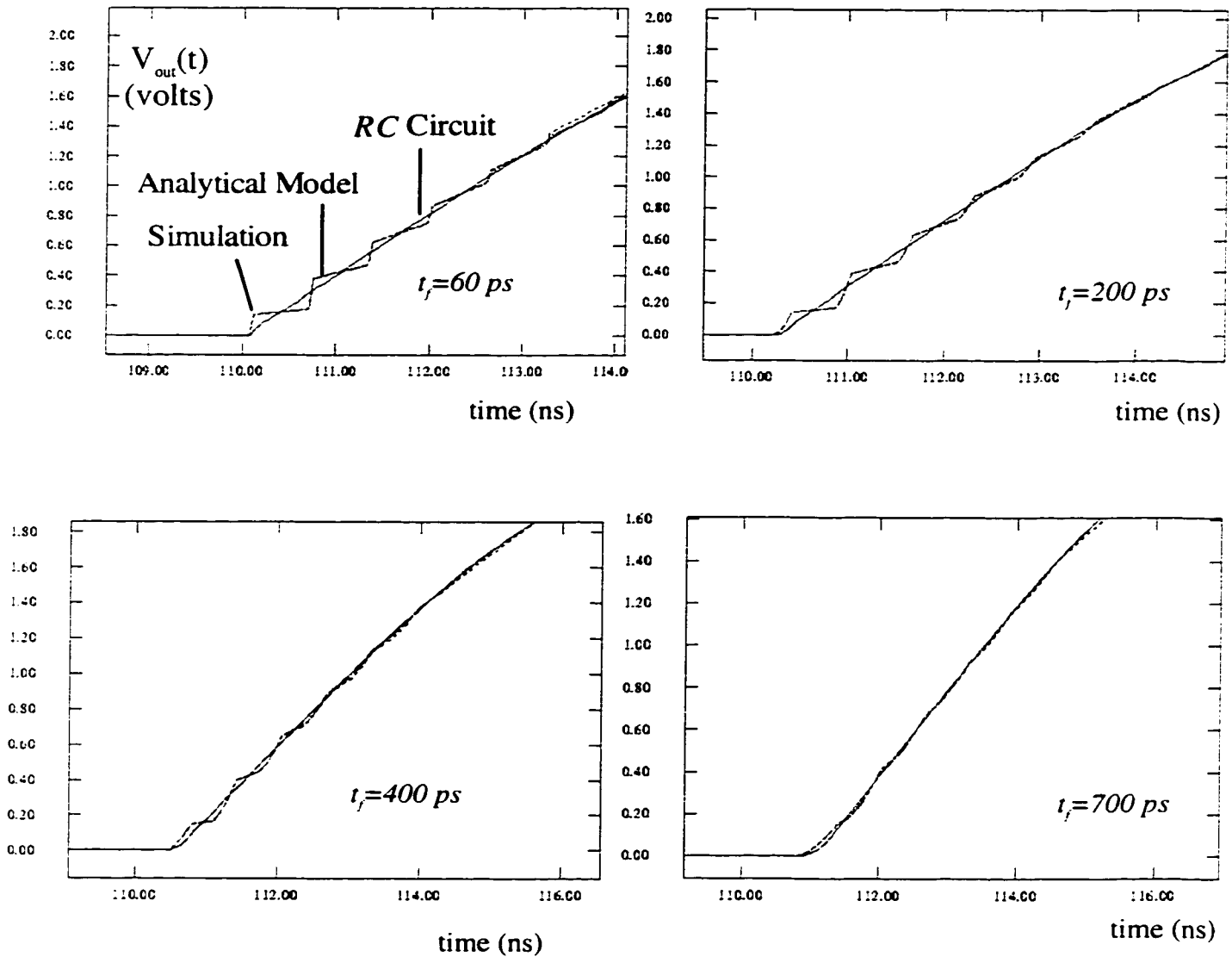


Figure 5.9. Analytical solution in (5.16) compared to AS/X simulations and a five section *RC* circuit. *R* is held constant at 10 cm, while the fall time of the input signal is varied.

As the transition time of the input signal becomes equal to or greater than $2T_0$, the slowly varying portions of the *RLC* transmission line response disappear, and the response coincides with the *RC* approximation. This behavior is evident from Figure 5.9 which leads to the second figure of merit given by

$$t_r > 2l\sqrt{LC}\,. \tag{5.20}$$

where the asymptotic value of T_0 is $l\sqrt{LC}$. When this inequality is satisfied, inductance becomes unimportant. Note that the criterion in (5.20) is accurate only if the line is matched (the width of the transistor driving the line is adjusted to match the transistor output impedance with the load impedance of the line to avoid reflections) or underdriven (the width of the transistor driving the line is less than is necessary to match the transistor impedance to the load impedance). However, this condition does not affect the validity of the results since in most practical cases it is undesirable to overdrive the line (using a transistor wider than the matched size). If the line is overdriven, overshoots occur which degrade performance. Also, to overdrive the line, wider transistors are needed which places a larger capacitive load on the previous stage.

5.2 Range of Interconnect for Significant Inductance Effects

The two figures of merit in (5.9) and (5.20) can be combined into a two sided inequality that determines the range of the length of interconnect in which inductance effects are significant. This condition is given by

$$\frac{t_r}{2\sqrt{LC}} < l < \frac{2}{R}\sqrt{\frac{L}{C}}. \tag{5.21}$$

This range depends upon the parasitic impedances of the interconnect per unit length as well as on the rise time of the signal at the input of the CMOS circuit driving the interconnect. In certain cases, this range can be non-existent if the following condition is satisfied,

$$t_r > 4\frac{L}{R}. \tag{5.22}$$

In this case, inductance is not important for any length of interconnect. For short lines, the time of flight across the line is too small compared to the transition time of the input signal. As the line becomes longer, the attenuation becomes large enough to make the inductance effects negligible. If the effect of the attenuation comes into play before the effect of the rise time vanishes, the inductance is not important for any length of interconnect

To demonstrate this behavior, AS/X simulations are shown in Figure 5.10 for $L = 10^{-7}$ H/cm, $R = 400$ Ω/cm, $C = 10^{-12}$ F/cm, and $t_r = 0.25$ ns. With these values, (5.21) reduces to 0.3259 cm $< l <$ 1.58 cm. This region defines the range of l for which an *RC* model is no longer accurate and the interconnect impedance model must include inductance. The response of a 5-section *RC* Π circuit compared to the response of an *RLC* transmission line is shown in Figure 5.10. The *RC* model is inaccurate in the range indicated by (5.21). The limits are not sharp and the further l is out of the range defined by (5.21), the more accurate the *RC* model becomes. The simulation results for $L = 10^{-8}$ H/cm, $R = 400$ Ω/cm, $C = 10^{-12}$ F/cm, and $t_r = 0.25$ ns are shown in Figure 5.11. In this case, (5.21) reduces to 1.25 cm $< l <$ 0.5 cm, which

demonstrates that no possible value of l exists for which the inductance effects are significant. The results depicted in Figure 5.11 show that the response of an RC circuit model is accurate for all l for this set of device and interconnect parameters.

The region where inductance becomes important in terms of the transition time and the length of interconnect is depicted in Figure 5.12. Note that as the inductance L increases, the upper bounding line shifts up and the slope of the lower bounding line decreases, thereby increasing the region where inductance is important. The effect of increasing the resistance is to shift the upper bounding line down, thereby decreasing the region where inductance is important. Increasing the capacitance shifts the upper bounding line down and decreases the slope of the lower bounding line. The transition time at which the two lines intersect remains constant at

$$t_r = 4\frac{L}{R}. \tag{5.23}$$

Thus, as the capacitance increases, the area where inductance is important is reduced. Also note that this area may be non-existent if (5.22) is satisfied.

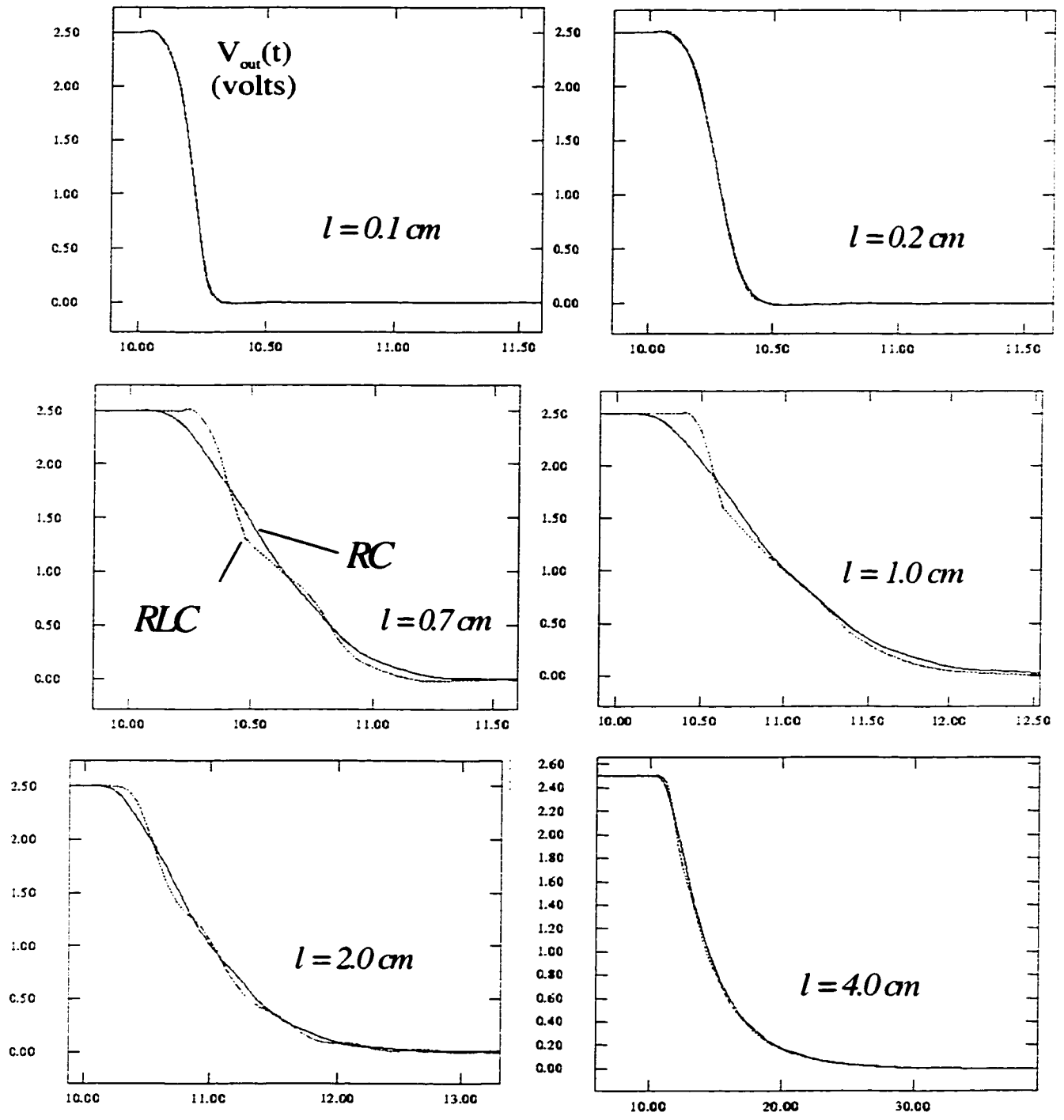


Figure 5.10. AS/X simulations of the response of a 5-section *RC* model compared to the response of an *RLC* transmission line for different values of *l*. $L = 10^{-7}$ H/cm, $R = 400$ Ω/cm, $C = 10^{-12}$ F/cm, and $t_r = 0.25$ ns. The results of the circuit simulation demonstrate that inductance has a significant effect on the response of a signal propagating across an interconnect line for the range of length defined by (5.21). Note that the RC circuit model becomes more accurate for small l or large *l*.

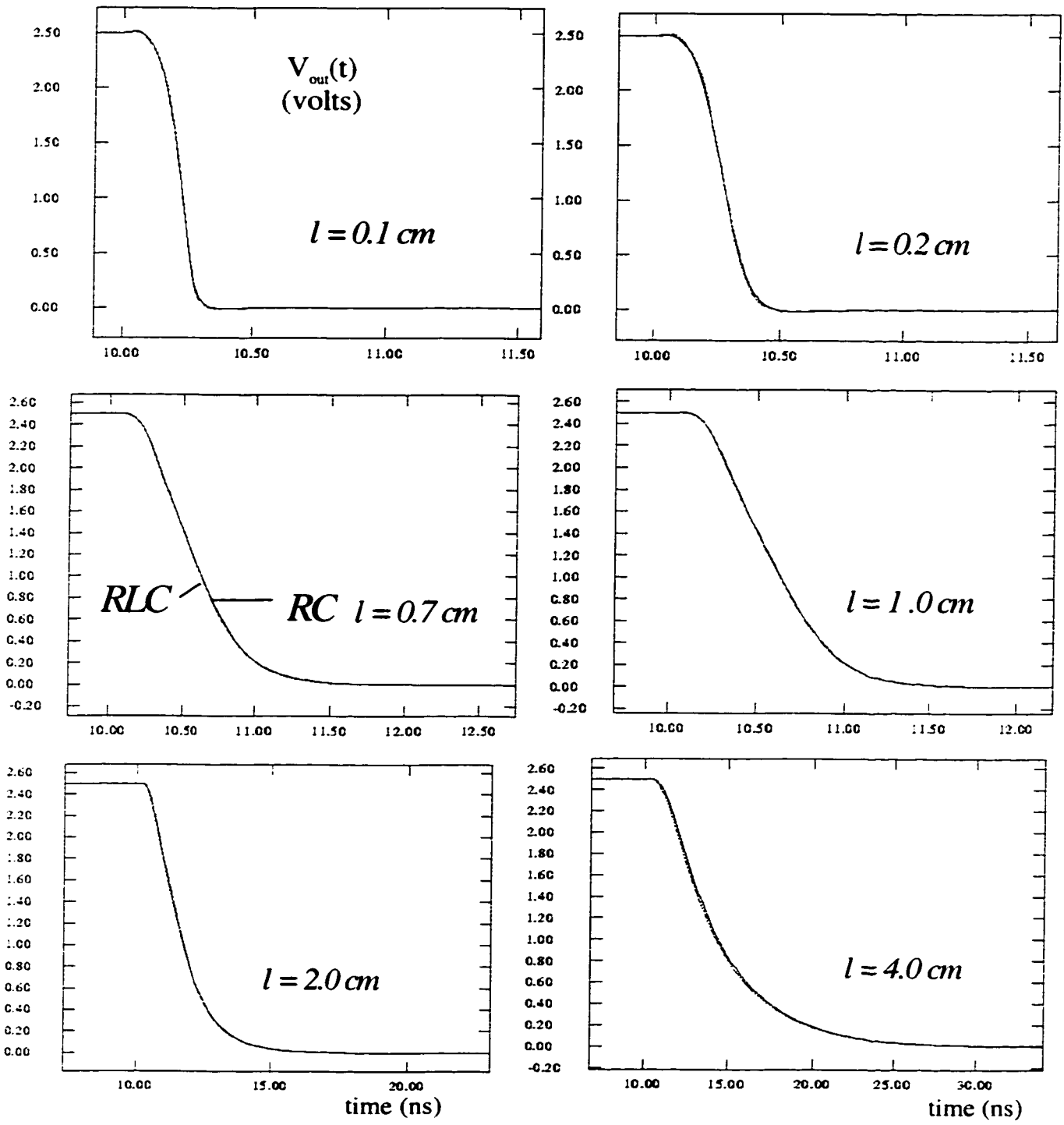


Figure 5.11. AS/X simulations of the response of a 5-section *RC* model compared to the response of an *RLC* transmission line for different values of *l*. $L = 10^{-8}$ H/cm, $R = 400\ \Omega$/cm, $C = 10^{-12}$ F/cm, and $t_r = 0.25$ ns. The results of the circuit simulation demonstrate that inductance has a minimal effect on the response of a signal propagating across an interconnect line despite the length of interconnect as given by (5.21) for the values of *R*, *L*, *C*, and t_r cited above.

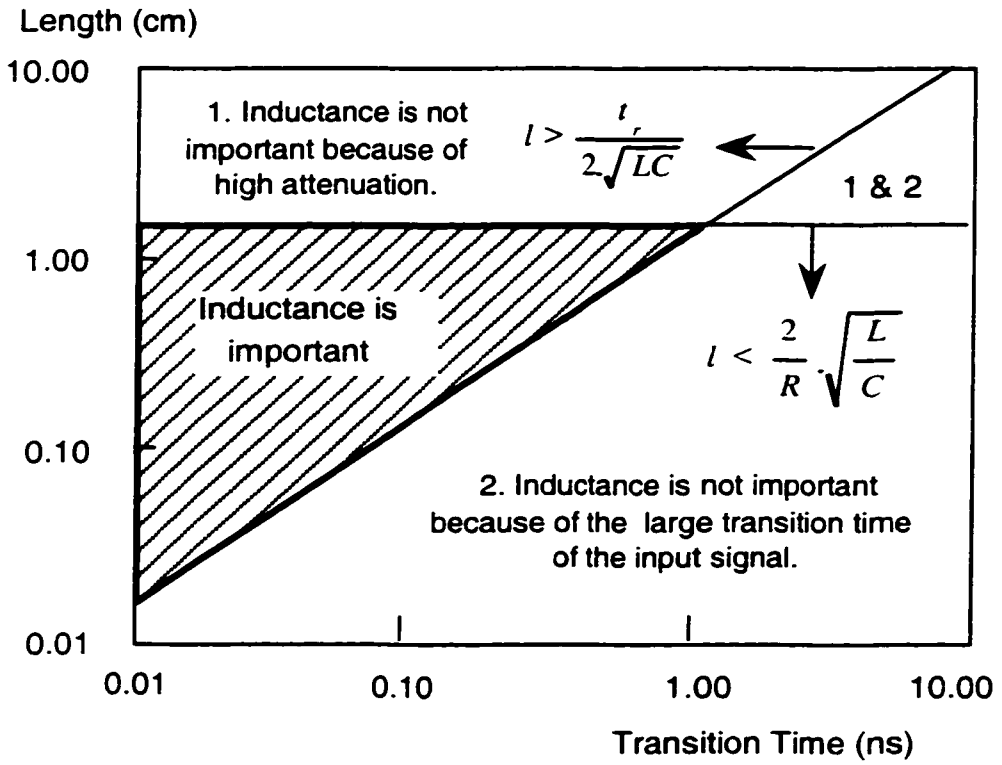


Figure 5.12. Transition time (t_r) versus the length of the interconnect line (l). The crosshatched area denotes the region where inductance is important. $L = 10^{-8}$ H/cm, $R = 400$ Ω/cm, and $C = 10^{-12}$ F/cm.

5.3 Conclusions

A closed form solution of the output response of a CMOS inverter driving an *RLC* transmission line is presented using the alpha power law for deep submicrometer technologies. Simple to use figures of merit have been developed that determine the relative accuracy of an *RC* impedance to model on-chip interconnect. The range of length of interconnect where a more accurate transmission line model becomes

necessary is shown to be based on the parasitic impedances of the line (*R, L, and C*) and the rise time of the input signal at the gate driving the line. AS/X simulations with a 0.25 μm IBM CMOS technology exhibit good agreement with these figures of merit. These figures of merit can be used in CAD tools to determine which nets needs to be modeled more accurately by including the effects of on-chip inductance. These figures of merit can also be used to properly size the interconnect and buffers along a line during the initial design phase of a high frequency circuit.

Chapter 6 Effects of Inductance on the Propagation Delay and Repeater Insertion process in *RLC* Lines

The focus of this chapter is to provide an accurate estimate of the propagation delay of a gate driving a *distributed RLC* line as well as to develop design expressions for optimum repeater insertion to minimize the delay of a signal propagating along a distributed *RLC* line. Repeaters are often used to minimize the delay required to propagate a signal through those interconnect lines that are best modeled as an *RC* impedance [37]-[44]. Thus, the objective of this chapter is to highlight the significance of increasing inductance effects in current VLSI circuits with respect to on-chip interconnect and repeater insertion in *RLC* lines.

A closed form expression for the propagation delay of a gate driving a distributed *RLC* line is introduced that is within 5% of dynamic circuit simulations for a wide range of *RLC* loads. It is shown that the error in the propagation delay if inductance is neglected and the interconnect is treated as a distributed *RC* line can be over 35% for current on-chip interconnect. It is also shown that the traditional quadratic dependence of the propagation delay on the length of the interconnect for *RC* lines approaches a linear dependence as inductance effects increase. On-chip

inductance is therefore expected to have a profound effect on traditional high performance IC design methodologies.

The closed form delay model is applied to the problem of repeater insertion in *RLC* interconnect. Closed form solutions are presented for inserting repeaters into *RLC* lines that are highly accurate with respect to numerical solutions. *RC* models can create errors of up to 30% in the total propagation delay of a repeater system as compared to the optimal delay if inductance is considered. The error between the *RC* and *RLC* models increases as the gate parasitic impedances decrease with technology scaling. Thus, the importance of inductance in high performance VLSI design methodologies will increase as technologies scale.

This chapter is organized as follows. In section 6.1, simple yet accurate propagation delay formula describing a gate driving a distributed *RLC* load is presented. In section 6.2, the propagation delay formula is used to develop design expressions for optimum repeater insertion to minimize the propagation delay of a distributed *RLC* line. Some conclusions are offered in section 6.3.

6.1 Propagation Delay of a Gate Driving an *RLC* Load

A simple yet accurate formula characterizing the propagation delay of a gate driving an *RLC* transmission line is presented in subsection 6.1.1. The closed form solution for the propagation delay is shown to be within 5% error of AS/X [128] simulations for a wide range of *RLC* lines. In subsection 6.1.2, the closed form solution for the propagation delay is shown to accurately describe the special case of an *RC* line as $L \rightarrow 0$. The solution for the propagation delay including inductance is

compared to the case where inductance is neglected and the line is treated as an *RC* line, permitting the error due to neglecting inductance to be quantified. In subsection 6.1.3, the dependence of the propagation delay on the length of an interconnect line is investigated. It is shown that the traditional quadratic dependence of the propagation delay on the length of the interconnect for an *RC* line tends to a linear relation as inductance effects increase.

6.1.1 Propagation Delay Formula

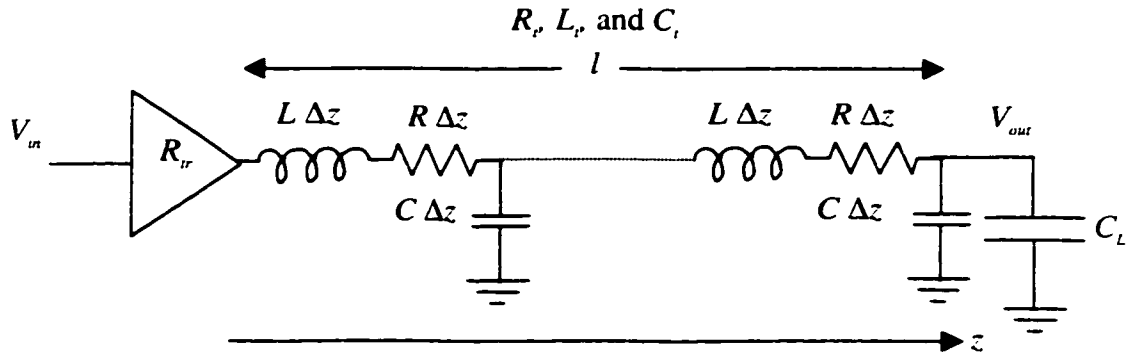


Figure 6.1. A gate driving an *RLC* transmission line.

A gate driving an *RLC* transmission line representation of an interconnect line is shown in Figure 6.1. R_t, L_t, and C_t are the total resistance, inductance, and capacitance of the line, respectively. The line parameters R_t, L_t, and C_t are given by $R_t = Rl$, $L_t = Ll$, and $C_t = Cl$, respectively, where R, L, and C are the resistance, inductance, and capacitance per unit length of the interconnect and l is the length of the line. The conductance of the line G is neglected since with current operating frequencies the capacitive impedance dominates the parallel semiconductor

conductance. R_{tr} is the equivalent output resistance of the gate driving the interconnect. C_L is the input capacitance of the following gate at the end of the interconnect section. A minimum size buffer has an output resistance R_0 and an input capacitance C_0. The input voltage V_{in} is a fast rising signal that can be approximated by a step signal. V_{out} is the far output voltage at the end of the interconnect section.

From the basic principles of a transmission lines (see section 2.1.2), the transfer function of a lossy transmission line with a source impedance z_s and a load impedance z_L, $V_{out}(s)/V_{in}(s)$ is given by

$$\frac{V_{out}(s)}{V_{in}(s)} = \frac{2}{\left(\frac{z_s}{z_0}+1\right)\left(\frac{z_0}{z_L}+1\right)e^{\gamma l} + \left(\frac{z_s}{z_0}-1\right)\left(\frac{z_0}{z_L}-1\right)e^{-\gamma l}}, \tag{6.1}$$

where γ and z_0 are the propagation constant and the characteristic impedance of the line and are given by

$$z_0 = \sqrt{\frac{L_t}{C_t}}\sqrt{1+\frac{R_t}{sL_t}}. \tag{6.2}$$

$$\gamma l = s\sqrt{L_t C_t}\sqrt{1+\frac{R_t}{sL_t}}. \tag{6.3}$$

For a CMOS gate driving another CMOS gate at the end of the line, $z_s = R_{tr}$ and $z_L = 1/sC_L$. A time scaling is applied by substituting t' / ω_n for each t where

$$\omega_n = \frac{1}{\sqrt{L_t(C_t + C_L)}}. \tag{6.4}$$

From the characteristics of the Laplace transform, the complex frequency s is substituted by $\omega_n s'$. With this time scaling, the variables γ, z_0, and z_L are transformed to γ', z'_0, and z'_L, respectively, which can be evaluated by substituting $\omega_n s'$ for each s and are

$$\gamma' l = \frac{s'}{\sqrt{1+C_T}}\sqrt{1+\frac{2\zeta_{line}}{s'}\sqrt{1+C_T}}, \tag{6.5}$$

$$z_0' = \sqrt{\frac{L_t}{C_t}}\sqrt{1+\frac{2\zeta_{line}}{s'}\sqrt{1+C_T}}, \tag{6.6}$$

$$z_L' = \sqrt{\frac{L_t}{C_t}}\frac{1}{s'}\frac{\sqrt{1+C_T}}{C_T}, \tag{6.7}$$

where

$$C_T = \frac{C_L}{C_t}, \tag{6.8}$$

$$\zeta_{line} = \frac{R_t}{2}\sqrt{\frac{C_t}{L_t}}. \tag{6.9}$$

Using the above expressions, the impedance ratios describing the transfer function in (6.1) become

$$\frac{z_0'}{z_L'} = \frac{1}{s'}\sqrt{1+\frac{2\zeta_{line}}{s'}\sqrt{1+C_T}}\frac{\sqrt{1+C_T}}{C_T}, \tag{6.10}$$

$$\frac{z_s'}{z_0'} = \frac{2R_T\zeta_{line}}{\sqrt{1+\frac{2\zeta_{line}}{s'}\sqrt{1+C_T}}}, \tag{6.11}$$

where

$$R_T = \frac{R_{tr}}{R_t}. \tag{6.12}$$

Referring to the transfer function in (6.1), (6.5), (6.10), and (6.11), the scaled transfer function in terms of s' is a function of only three variables: ζ_{line}, R_T, and C_T. The canonical number of variables to characterize the scaled transfer function in terms of s' is three. There are numerous ways to select the three variables that characterize the scaled transfer function. Three variables are chosen to simplify the process for determining the 50% delay point, which is the target of this analysis.

Thus, the three variables, ζ, R_T, and C_T, are chosen to describe the transformed transfer function, where

$$\zeta = \frac{R_t}{2}\sqrt{\frac{C_t}{L_t}} \cdot \frac{R_T + C_T + R_T C_T + 0.5}{\sqrt{(1+C_T)}} = \zeta_{line} \frac{R_T + C_T + R_T C_T + 0.5}{\sqrt{(1+C_T)}}. \tag{6.13}$$

The variables, R_T and C_T, characterize the relative significance of the gate parasitic impedances with respect to the parasitic interconnect impedances. Increasing R_T and C_T demonstrate that the gate parasitic impedances further affect the propagation delay. To clarify the process for selecting the third variable ζ, the transfer function is expressed as a series in the powers of s'. The exponential functions in the transfer function in (6.1) are replaced by a series expansion, resulting in

$$\frac{V_{out}(s')}{V_{in}(s')} = \frac{1}{\left(1+\frac{z_s}{z'_L}\right)\left(1+\frac{(\gamma' l)^2}{2!}+.....\right)+\left(\frac{z_s}{z'_0}+\frac{z'_0}{z'_L}\right)\left((\gamma' l)+\frac{(\gamma' l)^3}{3!}+.....\right)}. \tag{6.14}$$

The first few terms of the series expansion in powers of s' are

$$\frac{V_{out}(s')}{V_{in}(s')} = \frac{1}{1+2\zeta s'+\left(\frac{0.5+C_T}{1+C_T}+16\zeta^2\left(1-\frac{R_T^2+C_T^2+(R_T C_T)^2}{(R_T+C_T+R_T C_T+0.5)^2}\right)\right)s'^2+\cdots}. \tag{6.15}$$

The third variable ζ is the coefficient of s^1 in the denominator of the transfer function. ζ is chosen as the third variable since the 50% delay is primarily dependent on the coefficients of s^1 in the denominator and the numerator [73]. This characteristic is used to reduce the number of variables that affect the propagation delay from three to one (ζ). Note that the three variables, R_T, C_T, and ζ, are not independent since ζ is a function of R_T and C_T. Note also that (6.14) and (6.15) show the first terms of the series expansion of the transfer function in powers of s' and do not represent any truncation in the transfer function. The coefficients of powers of s' are functions of

only the three variables, R_T, C_T, and ζ, for any power as described by (6.1), (6.5), (6.10), and (6.11).

For a unit step input function, the output voltage waveform $V_{out}(t') = \mathcal{L}^{-1}\{(1/s)* V_{out}(s')/V_{in}(s')\}$ is also a function of the three variables, ζ, R_T, and C_T. The scaled 50% propagation delay t'_{pd} can be calculated by solving $V_{out}(t'_{pd}, \zeta, R_T, C_T) = 0.5$ which means that t'_{pd} is only a function of ζ, R_T, and C_T. Thus, the propagation delay of an *RLC* line with a source resistance R_{tr} and a load capacitance C_L has the form,

$$t_{pd} = \frac{t'_{pd}(\zeta, R_T, C_T)}{\omega_n}. \tag{6.16}$$

The scaled propagation delay t'_{pd} is dimensionless since ω_n has the units of 1/time. Note that this solution is a characteristic of an *RLC* line and that no approximations have been made in deriving this result.

Equation (6.16) states that the same value of the scaled 50% delay t'_{pd} results for many different transmission line configurations driven by a step input supply with a source resistance and a load capacitance. The value of t'_{pd} remains constant as long as R_t, L_t, C_t, R_{tr}, and C_L scale such that ζ, R_T, and C_T are constant. Thus, simulations are used to characterize t'_{pd} as a function of ζ, R_T, and C_T based on certain parameters, R_t, L_t, C_t, R_{tr}, and C_L. The resulting expression for t'_{pd} is guaranteed to correctly characterize any combination of the parameters, R_t, L_t, C_t, R_{tr}, and C_L.

AS/X [128] simulations of the time scaled 50% propagation delay t'_{pd} of a gate driving an *RLC* transmission line as a function of ζ, R_T, and C_T are shown in Figure 6.2. The simulations depicted in Figure 6.2 for the curve with $R_T = 0$ and $C_T = 0$ are

performed with R_t = 50 Ω, C_t = 1 pF, R_{tr} = 0, and C_L =0, and L_t is varied to vary ζ. AS/X is used to determine the 50% delay t_{pd} for each value of L_t. The result is divided by ω_n in (6.4) to determine t'_{pd}. For the curve with R_T = 1 and C_T = 1, the same procedure is used but with R_{tr} = 50 Ω and C_L = 1 pF. For the curve with R_T = 5 and C_T = 5, R_{tr} = 250 Ω and C_L = 5 pF. The specific values of the parameters (R_t, L_t, C_t, R_{tr}, and C_L) used in the simulations shown in Figure 6.2 are not important as long as the required ranges of ζ, R_T, and C_T are satisfied. For the cases where the output response crosses the 50% point several times due to severe ringing, the propagation delay is calculated based on the final crossing which represents the worst case delay. Note in Figure 6.2 that the propagation delay is primarily a function of ζ. The dependence on R_T and C_T is fairly weak. This characteristic does not imply that the transistor driving the interconnect and the load capacitance has a weak effect on the propagation delay since ζ includes the effects of R_T and C_T as given by (6.13). Only the extra effect of R_T and C_T that is not included in ζ is neglected. Note also that this effect is particularly weak in the range where R_T and C_T are between zero and one. This range is most important for global interconnect and long wires in current deep submicrometer technologies. Thus, the propagation delay is primarily a function of ζ, which collects the five parameters that affect the propagation delay, R_t, L_t, C_t, R_{tr}, and C_L, into a single parameter. The time scaled propagation delay t'_{pd} is considered as a function of only ζ in the range where R_T and C_T are between zero and one and the propagation delay is given by

$$t_{pd} \approx \frac{t'_{pd}(\zeta)}{\omega_n}. \tag{6.17}$$

Approximating the time scaled propagation delay t'_{pd} as a function of only one variable allows simple one-dimensional curve fitting methods to be applied to determine an expression describing the 50% delay. A curve fitting method is used to minimize the error when R_T and C_T are between zero and one as shown in Figure 6.2, resulting in the following expression for the 50% propagation delay,

$$t_{pd} = (e^{-2.9\zeta^{1.35}} + 1.48\zeta)/\omega_n. \tag{6.18}$$

AS/X [128] simulations of the propagation delay of an *RLC* transmission line as compared to t_{pd} in (6.18) are shown in Table 6.1. Note that the solution exhibits high accuracy (the maximum error is 4.6% and the average error is 1.65%) for a wide range of interconnect (R_t, L_t, and C_t) and gate impedances (R_{tr} and C_L). Values of ζ are calculated and listed in Table 6.1 for the simulated cases, which varies from 3.36 to 0.20. Thus, the simulation data listed in Table 6.1 include those cases with high inductive effects where the response is underdamped and overshoots occur (small ζ), and those cases with low inductive effects where the response is overdamped (large ζ). Equation (6.18) characterizes the propagation delay accurately for any set of parameters R_t, L_t, C_t, R_{tr}, and C_L for which R_T and C_T are in the range between zero and one and any value of ζ.

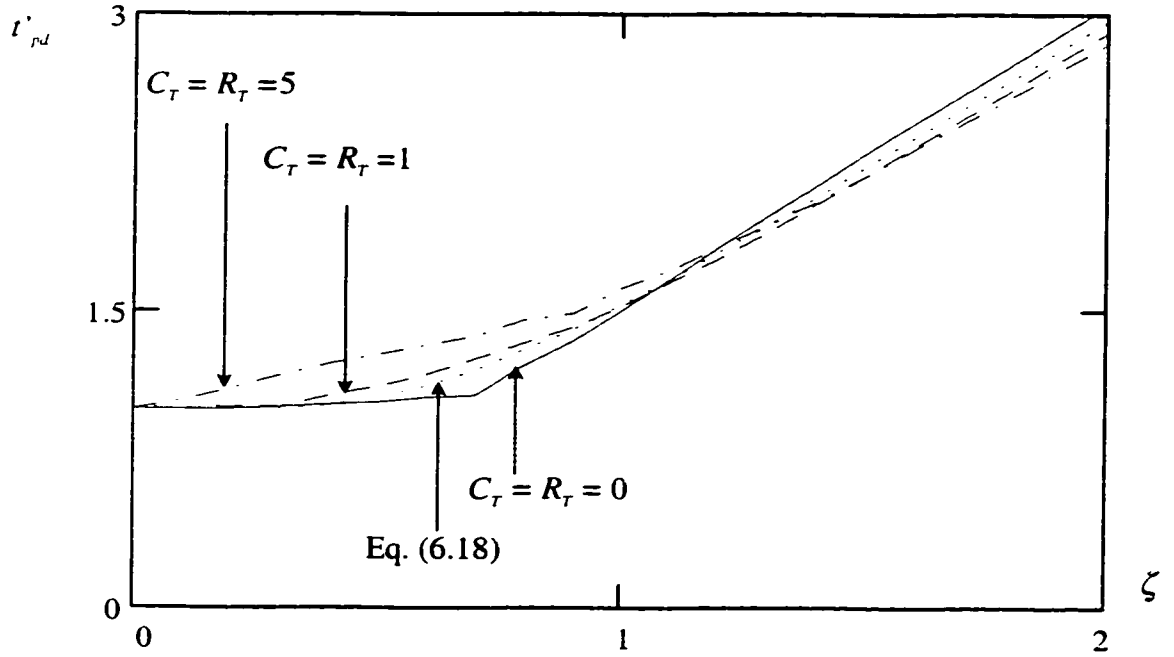


Figure 6.2. Comparison of the accuracy of (6.18) to AS/X [128] simulations of the time scaled 50% propagation delay t'_{pd} of an *RLC* transmission line with a source resistance R_{tr} and a load capacitance C_L. The propagation delay is plotted versus ζ for different values of R_T and C_T.

Table 6.1. Comparison of t_{pd} in (6.18) to AS/X simulations characterizing the propagation delay of a gate driving an *RLC* transmission line. C_t = 1 pF and R_{tr} = 25 Ω. The shaded rows represent the simulated cases shown in Figure 6.3.

R_t	L_t (nH)	$C_T = 0.1$				$C_T = 0.5$				$C_T = 1.0$			
		ζ (6.13)	(6.18) (ps)	AS/X (ps)	Error	ζ (6.13)	(6.18) (ps)	AS/X (ps)	Error	ζ (6.13)	(6.18) (ps)	AS/X (ps)	Error
0.1	2	1.89	131	134	2.2%	**2.62**	**213**	**214**	**0.5%**	3.36	314	311	1.0%
	5	1.19	133	135	1.5%	1.66	213	216	1.4%	2.12	314	313	0.3%
	8	**0.94**	**138**	**137**	**0.7%**	**1.31**	**214**	**218**	**1.8%**	**1.68**	**315**	**316**	**0.3%**
	10	0.84	142	138	2.9%	1.17	216	219	1.4%	1.503	315	320	1.6%
0.5	2	0.61	53	51	3.9%	0.80	71	71	0.0%	0.99	96	98	2.0%
	5	0.34	76	77	1.3%	0.50	92	95	3.1%	0.62	114	117	2.5%
	8	0.31	95	96	1.0%	0.40	112	115	2.6%	0.49	134	140	4.2%
	10	0.27	106	107	0.9%	0.36	124	126	1.6%	0.44	146	152	3.9%
1.0	2	0.45	49	49	0.0%	0.57	60	61	1.6%	0.69	75	78	3.8%
	5	0.29	75	76	1.3%	0.36	88	88	0.0%	**0.44**	**103**	**108**	**4.6%**
	8	0.23	95	95	0.0%	0.28	110	110	0.0%	0.34	128	131	2.3%
	10	**0.20**	**106**	**106**	**0.0%**	0.25	124	121	2.4%	0.31	143	144	0.7%

The parameter ζ can be used to characterize inductance effects more accurately and comprehensively than the figures of merit developed in [49]-[51] and [66]-[68] (see Chapter 5). To better explain this point, note that ζ can be rewritten as

$$\zeta = \frac{1}{2\sqrt{(1+C_T)}}\left[\frac{R_t}{2Z_{0l}} + \frac{R_{tr}}{Z_{0l}} + \frac{\tau_{C_L}}{\tau_f}\right], \tag{6.19}$$

where $Z_{0l} = \sqrt{L_t / C_t}$ is the characteristic impedance of a lossless transmission line, τ_{CL} $= C_L(R_t + R_{tr})$ is the time constant for charging the load capacitance C_L through the gate and wire resistances, and $\tau_f = \sqrt{L_t C_t}$ is the time of flight of the signals propagating across the transmission line. Thus, (6.19) characterizes three different factors that determine inductance effects in *RLC* lines. The first factor is the total line resistance R_t as compared to the lossless characteristic impedance of the line Z_{0l}. If the ratio of the total resistance of the line to the lossless characteristic impedance increases, inductance effects can be neglected. The second factor is the ratio between the driver resistance R_{tr} and the lossless characteristic impedance of the line. If this ratio increases, inductance effects can be neglected. The last factor is the ratio between the time required to charge C_L through the gate and wire resistances to the time of flight of the signals propagating across the line. If this ratio increases, inductance effects can be neglected. The three factors are collected in the single metric ζ which is sufficient to characterize inductance effects exhibited by an *RLC* line and includes the effects of the driver output resistance and the load capacitance. The same three factors are characterized in [51] by three separate equations each

characterizing when the inductance becomes negligible due to each factor[2]. The difficulty with this approach is that certain cases exist where each of these factors separately tested for inductance effects would predict that the line would suffer inductance effects while actually the line would suffer no inductance effects due to the *combined* effect of the three factors. The single metric ζ introduced here accurately models the combined effect of these three factors, which is represented by the addition in (6.19). Simulations comparing an *RLC* to an *RC* interconnect model for the shaded cells in Table 6.1 are depicted in Figure 6.3. Note that the error due to neglecting inductance is insignificant for $\zeta > 1.5$. Note also that the effect of the rise time of the input signal on the significance of inductance is not considered here but is characterized in [68].

[2] The load capacitor metric in [51] is different from the metric introduced here.

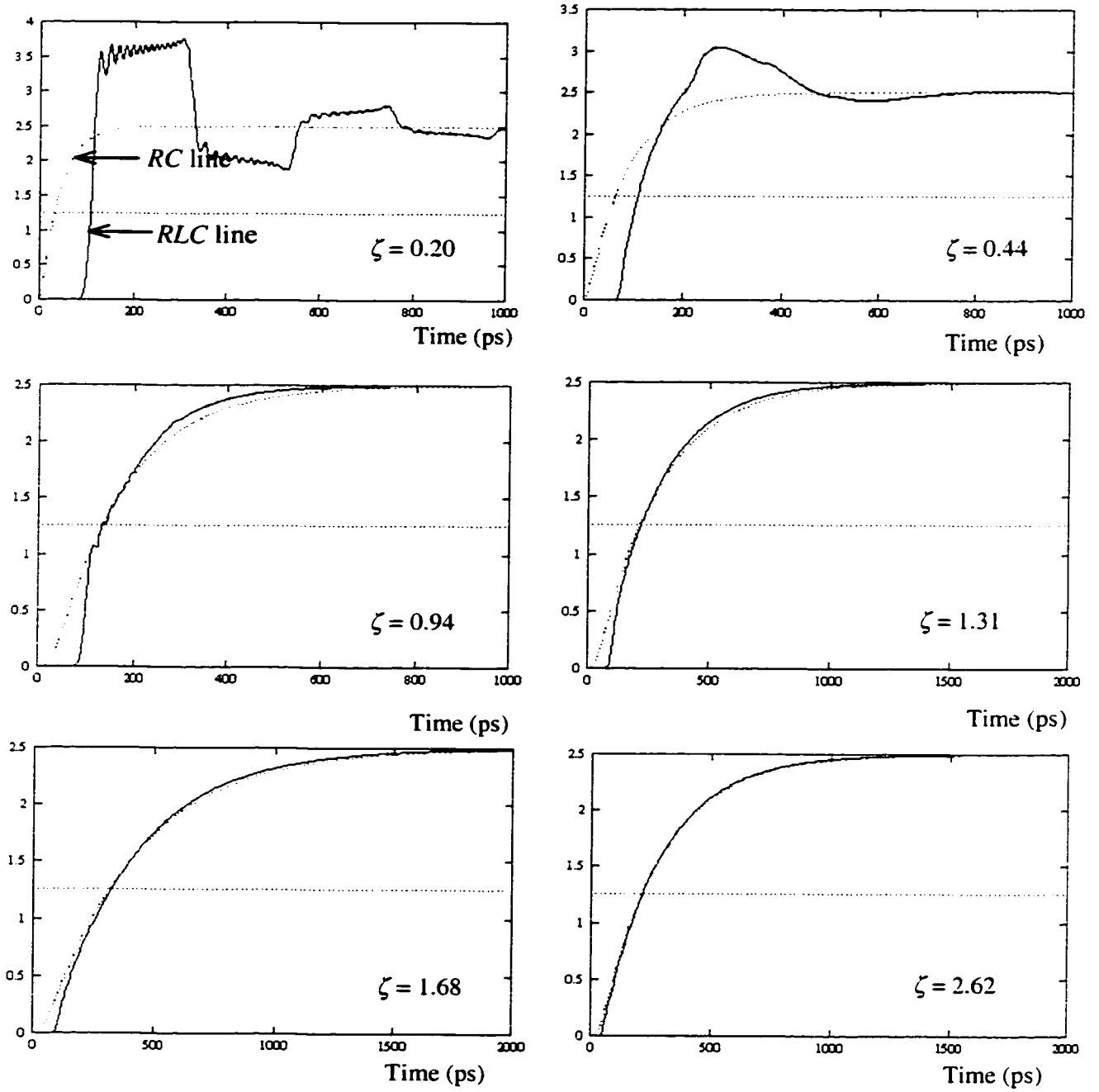


Figure 6.3. Circuit simulations comparing an *RLC* interconnect model to an *RC* interconnect model for the shaded cells in Table 6.1. The metric ζ in (6.13) is shown on each individual graph.

6.1.2 Comparison to an *RC* Model

The propagation delay t_{pd} in (6.18) can be rewritten as

$$t_{pd} = \frac{e^{-2.9\zeta^{1.35}}}{\omega_n} + 0.74R_tC_t\left(R_T + C_T + R_TC_T + 0.5\right). \tag{6.20}$$

To examine how accurately the closed form solution of the propagation delay of an *RLC* transmission line in (6.20) characterizes the special case of a distributed *RC* line, (6.20) is evaluated as L_t approaches zero (note that inductance is not equal to zero). Assuming that L_t approaches zero in the mathematical expressions means that inductance effects are negligible because the value of ζ is high. As given by (6.4) and (6.13), $\omega_n \rightarrow \infty$ and $\zeta \rightarrow \infty$ as $L_t \rightarrow 0$ and thus,

$$t_{pd}(RC) = 0.74R_tC_t\left(R_T + C_T + R_TC_T + 0.5\right), \tag{6.21}$$

which can be rearranged into

$$t_{pd}(RC) = 0.37R_tC_t + 0.74\left(R_tC_L + R_{tr}C_t + R_{tr}C_L\right). \tag{6.22}$$

Note the similarity of this expression to the expressions for the propagation delay of a distributed *RC* line in [23] and [40] (see section 2.2). Thus, the general expression for the propagation delay of a CMOS gate driving an *RLC* interconnect described by (6.18) also includes the special case of an *RC* interconnect. Note also that the term $1.48\zeta / \omega_n$ in (6.18) is $t_{pd}(RC)$. Thus, (6.18) can be viewed as the traditional *RC* delay plus a correction term representing the effects of inductance.

The error encountered when neglecting the inductance of an interconnect line and treating the line as an *RC* line is quantified by the expression ($t_{pd}(RLC)$ –

$t_{pd}(RC))/t_{pd}(RLC)$. $t_{pd}(RLC)$ is given by (6.18) and $t_{pd}(RC)$ is given by $1.48\zeta/\omega_n$. The per cent error with these expressions is

$$\%Error = \frac{100e^{-2.9\zeta^{1.35}}}{e^{-2.9\zeta^{1.35}} + 1.48\zeta}. \tag{6.23}$$

Note that the error is only a function of ζ. Eq. (6.23) and AS/X simulations are plotted in Figure 6.4. The closed form solution in (6.23) accurately anticipates the error in the propagation delay due to neglecting inductance and can be treated as a useful metric to determine when inductance should be included in an interconnect model. Note also that the error is less than 1% for $\zeta > 1.5$, permitting the *RC* model to be applicable with minimal error for $\zeta > 1.5$. However, for small ζ ($\zeta < 1$), the error rapidly increases (the error is 30% for $\zeta = 0.5$). Inductance should be included within the interconnect model to maintain sufficient accuracy for small ζ. Low resistance, wide wires (and thus low ζ) are frequently encountered in clock distribution networks and certain critical global interconnect (such as data busses). More accurate *RLC* models are required for these global interconnect lines particularly since accuracy is of great importance for these nets. Typical values of line parameters for a 0.25 μm CMOS technology are given in Appendix A for different line widths and lengths. Note that lines of widths 2.4 μm and 7.5 μm have a value of ζ significantly less than 1.5 for almost all wire lengths. These widths are common widths of global wires which can therefore exhibit significant inductance effects. This characteristic demonstrates that large errors can be encountered in current VLSI circuits if inductance is neglected. AS/X simulations of CMOS gates driving copper

interconnect lines from a 0.25 μm CMOS technology are shown in Figure 6.5. The simulations in Figure 6.5 compare the two cases of modeling an interconnect line as an *RLC* transmission line and as an *RC* transmission line for several driver widths and line dimensions. The error in the propagation delay due to neglecting inductance can be as high as 58% for wide drivers and wide wires. What makes these errors even more serious is that neglecting inductance and using an *RC* model rather than an *RLC* model always results in underestimating the propagation delay. Thus, VLSI circuits designed using an *RC* interconnect model may not satisfy the assigned frequency targets despite a worst case analysis being applied in the circuit design process and maintaining safety factors.

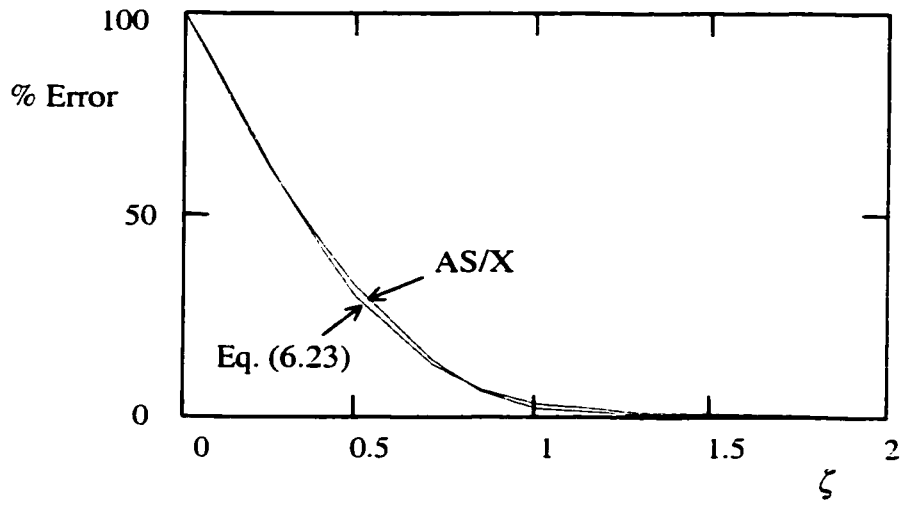
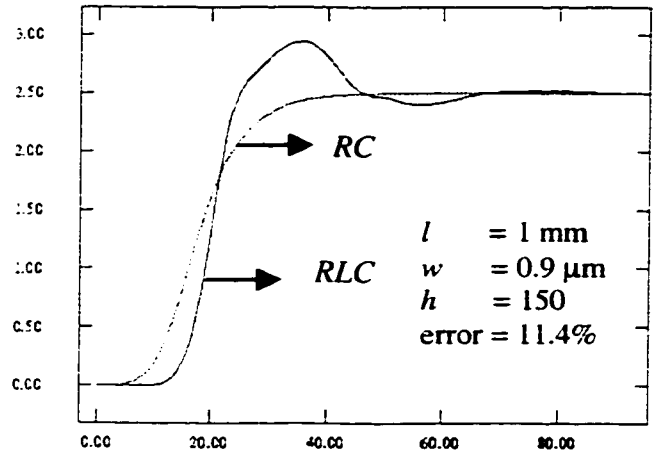
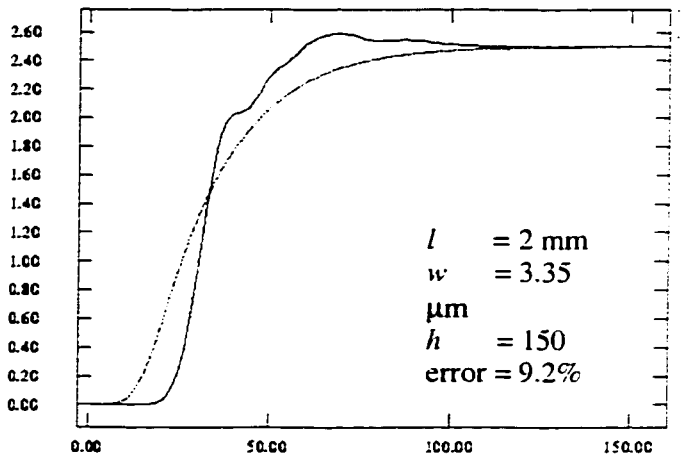


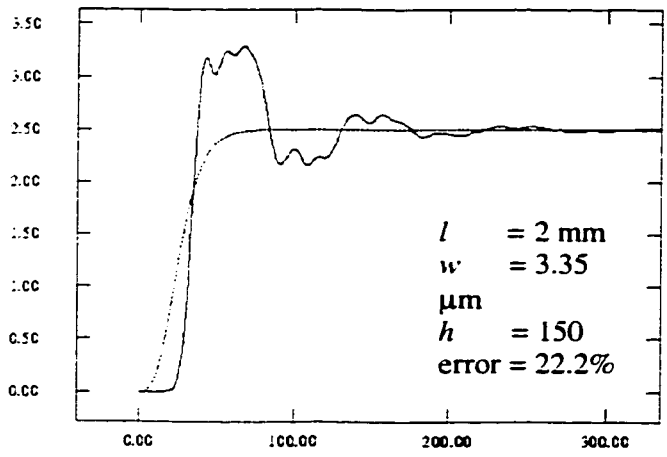
Figure 6.4. Eq. (6.23) as compared to AS/X simulations describing the error between an *RLC* transmission line model and an *RC* transmission line model. $R_t = 30\ \Omega$, $C_t = 1$ pF, $R_T = C_T = 0.5$, and L_t is varied to vary ζ.


RC
RLC
l = 1 mm
w = 0.9 µm
h = 150
error = 11.4%
Time (ps)

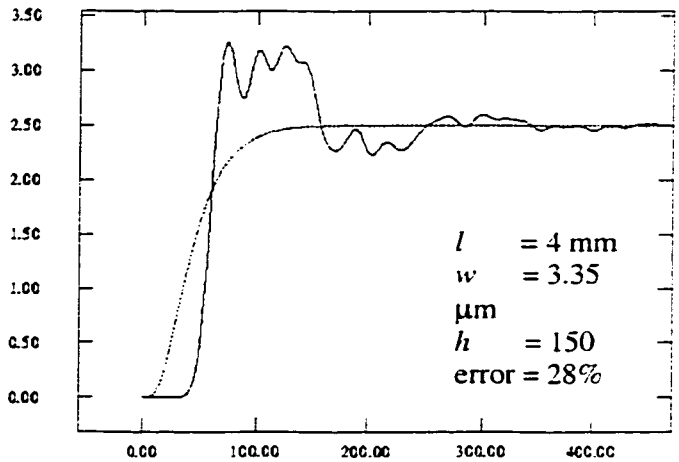
(a)


l = 2 mm
w = 3.35
µm
h = 150
error = 9.2%
Time (ps)

(b)


l = 2 mm
w = 3.35
µm
h = 150
error = 22.2%
Time (ps)

(c)


l = 4 mm
w = 3.35
µm
h = 150
error = 28%
Time (ps)

(d)

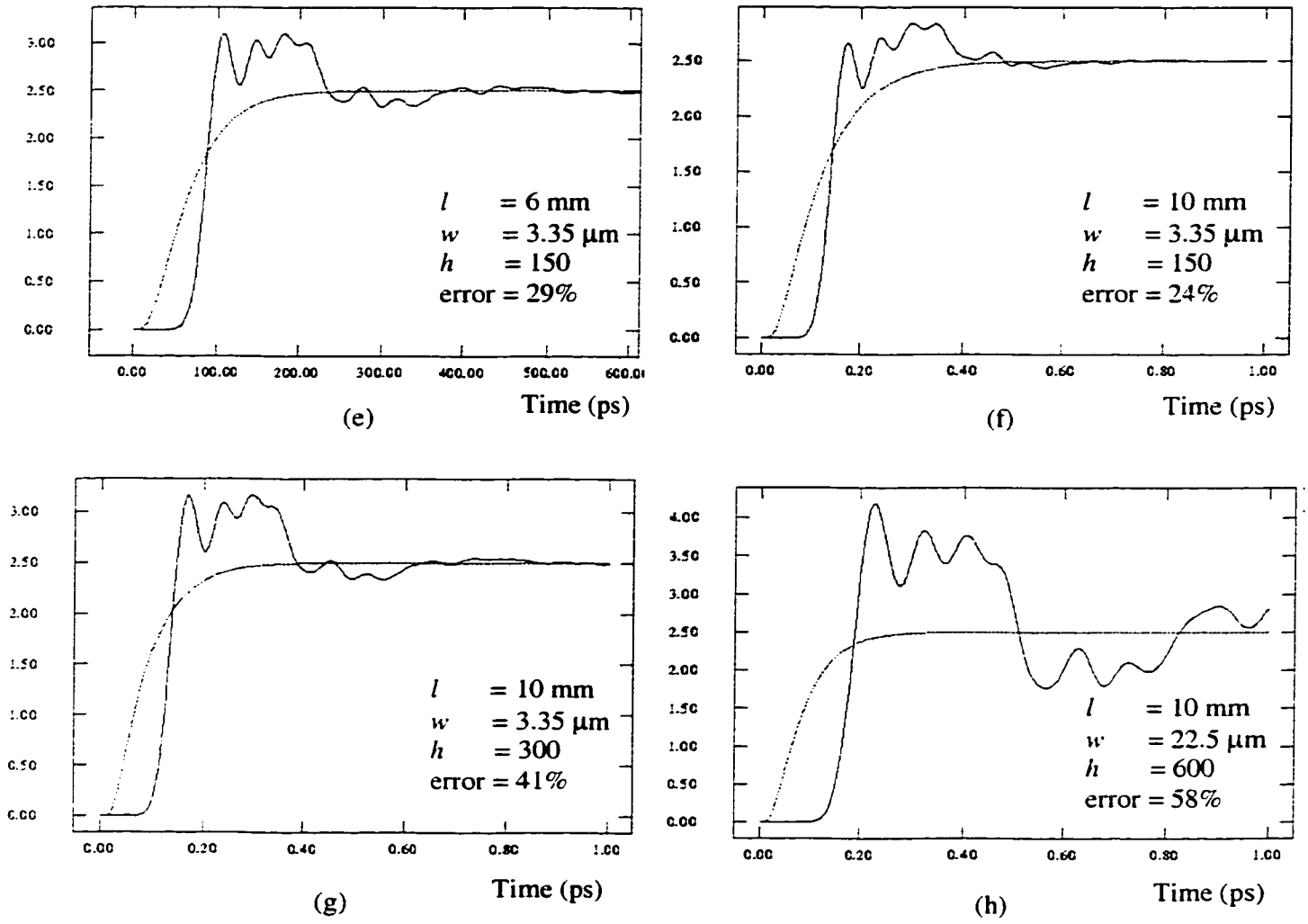


Figure 6.5. AS/X simulations of a CMOS gate driving a copper interconnect line based on a 0.25 μm CMOS technology. The lines are modeled as *RC* lines and as *RLC* lines and the two models are compared to characterize the effect of neglecting inductance. The wire length *l*, width *w*, and the size of the driving CMOS inverter as compared to a minimum size inverter *h* are shown in Figs. (a) to (h). The per cent error at the 50% delay point between the two models is also shown.

6.1.3 Dependence of Delay on Interconnect Length

An interesting special case occurs when the gate parasitics (C_L and R_{tr}) are neglected. This case is particularly important since it describes the propagation delay characteristics of a distributed *RLC* line without the distortion of the gate impedances. In this case, the propagation delay in (6.18) can be expressed as

$$t_{pd} = \sqrt{LC}(e^{-2.9(\alpha_{asym}l)^{1.35}} l + 0.74\alpha_{asym} l^2), \tag{6.24}$$

where

$$\alpha_{asym} = \frac{R}{2}\sqrt{\frac{C}{L}}. \tag{6.25}$$

α_{asym} is the asymptotic value at high frequencies of the attenuation per unit length of the signals as the signals propagate across a lossy transmission line. This expression is given in [68] and has the dimensions of nepers/cm [70].

For the limiting case where $L \rightarrow 0$, (6.24) reduces to $0.37RCl^2$. This expression is the same formula for the propagation delay of a distributed *RC* line as described in [1], [23], and [40]. Also note the well-known square dependence on the length of the wire. For the other limiting case where $R \rightarrow 0$, the propagation delay is given by $l\sqrt{LC}$. Note the linear dependence on the length of the line. The solution for the limiting case where $R \rightarrow 0$ is explained by noting that a distributed *RLC* line with zero resistance is simply a lossless transmission line. For a lossless transmission line, the speed at which a signal propagates is (see section 2.1.1)

$$v = \frac{1}{\sqrt{LC}}. \tag{6.26}$$

The time of flight of the signals across a lossless transmission line is $l/v = l\sqrt{LC}$ [70]. Thus, for a lossless transmission line, the propagation delay (in the case of $R_{tr} = 0$) is $l\sqrt{LC}$, which is the physically-based minimum limit for the propagation delay of an *RLC* line. This agreement between the general delay model in (6.18) and an *LC* transmission line demonstrates that the limiting case of an *LC* line can also be accurately described by (6.18).

The traditional quadratic dependence of the propagation delay on the length of an *RC* line approaches a linear dependence as inductance becomes more significant. According to (6.24), the parameter that describes this dependence on the interconnect length is α_{asym}. As described in [49], [52], and [133], signals propagate across a transmission line in two primary modes. The first mode is the propagation mode in which the signals travel at a constant velocity across the line and the delay is linear with the length of the interconnect. The second mode is the diffusion mode in which the signals diffuse through the line and the propagation delay is quadratic with the length of the interconnect. When there is no attenuation ($\alpha_{asym} = 0$), the signals propagate purely in the propagation mode as in the case of a lossless transmission line and therefore, $t_{pd} \propto l$. When the attenuation is large ($\alpha_{asym} > 1$), the signals propagate primarily in the diffusion mode as in the case of an *RC* transmission line and therefore, $t_{pd} \propto l^2$. Thus, α_{asym} describes the dependence of the propagation delay on the interconnect length. This behavior is illustrated in Figure 6.6. Note that for $\alpha_{asym} > 1$, the dependence on l is quadratic for all practical purposes. For $\alpha_{asym} < 1$, the square dependence is far from accurate which can have a profound effect on determining an

optimum strategy for driving an interconnect line such as repeater insertion and transistor sizing [37]-[44].

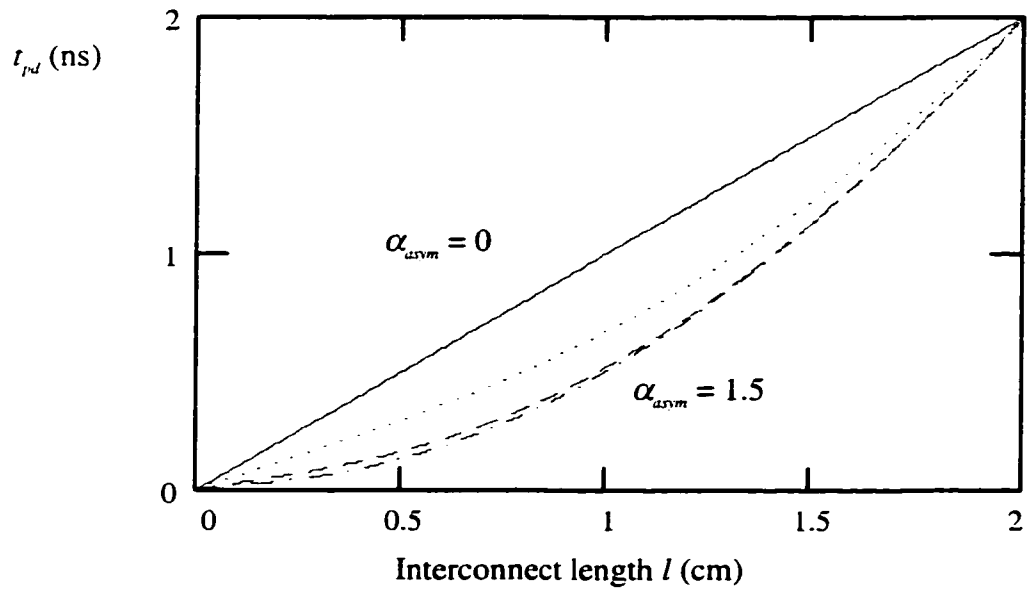


Figure 6.6. Dependence of the propagation delay on the length of the interconnect l ignoring the effects of the gate impedances. The curves represents $\alpha_{asym} = 0$, 0.5, 1.0, and 1.5 starting from the top curve.

6.2 Repeater Insertion for an *RLC* Interconnect

Traditionally, repeaters are inserted into *RC* lines to partition an interconnect line into shorter sections [37]-[44], thereby reducing the total propagation delay (see section 2.3). Applying the same idea to the general case of an *RLC* line, repeaters are used to divide the interconnect line into k sections as shown in Figure 6.7. The buffers are each uniformly the same size and h times larger than a minimum size buffer. The

buffer output impedance R_{tr} is R_0/h and the input capacitance of the buffer C_L is hC_0. The total propagation delay of the repeater system is the sum of the individual propagation delays of the k sections and is a function of h and k for a given interconnect line. The values of h and k at which the total delay $t_{pdtotal}$ is a minimum is determined by simultaneously solving the following two differential equations,

$$\frac{\partial t_{pdtotal}(h,k)}{\partial h} = 0, \tag{6.27}$$

$$\frac{\partial t_{pdtotal}(h,k)}{\partial k} = 0. \tag{6.28}$$

For the special case of an RC line ($L_t \rightarrow 0$), the solution for these equations is

$$h_{opt}(RC) = \sqrt{\frac{R_0 C_t}{R_t C_0}}, \tag{6.29}$$

$$k_{opt}(RC) = \sqrt{\frac{R_t C_t}{2R_0 C_0}}. \tag{6.30}$$

These equations are the same as described by Bakoglu in [40] (see section 2.3)..

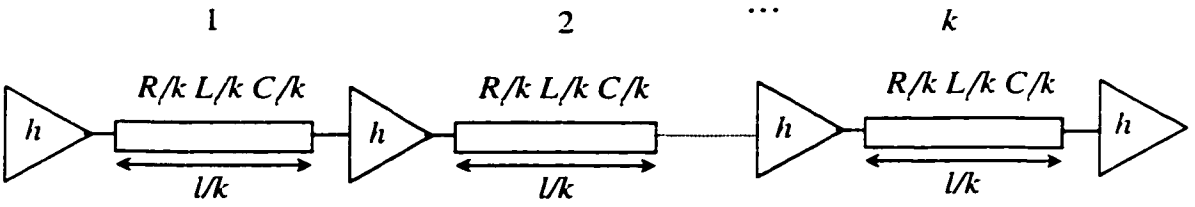


Figure 6.7. Repeaters inserted in an RLC line to minimize the propagation delay.

Solving (6.27) and (6.28) for the general case of an RLC line is analytically intractable. However, as described in Appendix B, h_{opt} and k_{opt} for an RLC line have the form,

$$h_{opt} = \sqrt{\frac{R_0 C_t}{R_t C_0}} \bullet h'(T_{L/R}) \ , \tag{6.31}$$

$$k_{opt} = \sqrt{\frac{R_t C_t}{2R_0 C_0}} \bullet k'(T_{L/R}), \tag{6.32}$$

where $h'(T_{L/R})$ and $k'(T_{L/R})$ are error factors that account for the effect of the inductance and $T_{L/R}$ is

$$T_{L/R} = \sqrt{\frac{L_t / R_t}{R_0 C_0}} \ . \tag{6.33}$$

The closed form solution for the propagation delay in (6.18) is used to characterize the delay of the repeater system shown in Figure 6.7 as described in Appendix B [(B.1)-(B.5)]. The resulting expression is partially differentiated with respect to h and k and the two derivatives are equated to zero. The resulting two equations are solved numerically for the optimum values of h and k (h_{opt} and k_{opt}). The values of $h'(T_{L/R})$ and $k'(T_{L/R})$ are found using (6.31) and (6.32) as

$$h'(T_{L/R}) = \frac{h_{opt}}{\sqrt{\frac{R_0 C_t}{R_t C_0}}} \ , \tag{6.34}$$

$$k'(T_{L/R}) = \frac{k_{opt}}{\sqrt{\frac{R_t C_t}{2R_0 C_0}}} \ . \tag{6.35}$$

h' and k' as functions of $T_{L/R}$ are plotted in Figure 6.8. The line and technology parameters used to generate Figure 6.8 are R_t = 100 Ω, C_t = 1 pF, R_0 = 1500 Ω, and C_0 = 2fF, and L_t is varied to vary $T_{L/R}$. Once h' and k' are characterized as functions of $T_{L/R}$ based on any line and technology parameters, h' and k' can be used in (6.31) and (6.32) with any other line and technology parameters (R_t, C_t, L_t, R_0, and C_0).

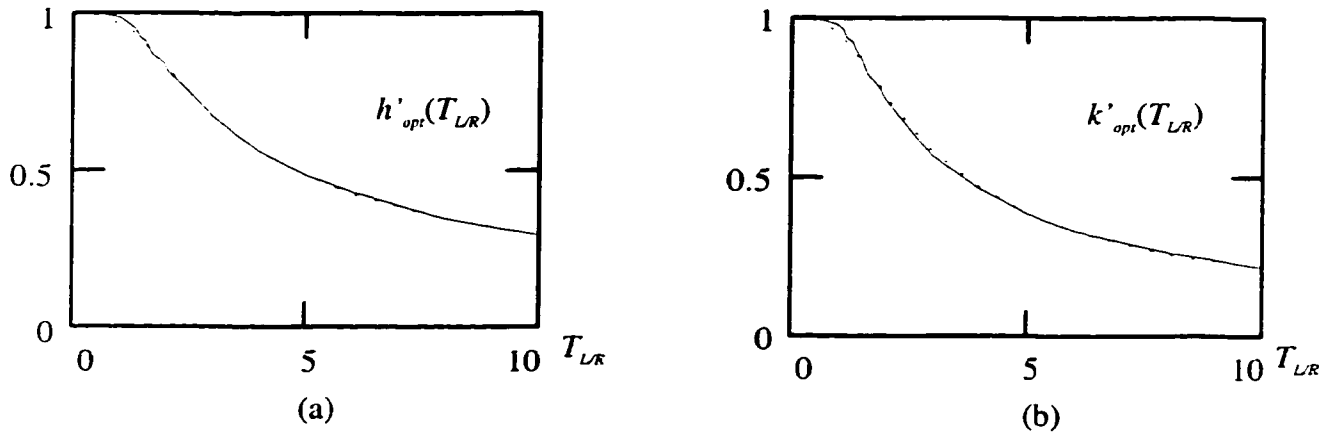


Figure 6.8. Numerical solutions of (6.27) and (6.28) and eqs. (6.36) and (6.37) for a) h_{opt} and b) k_{opt}, respectively. Numerical solutions are shown by the solid line while (6.36) and (6.37) are shown by the dashed line.

Curve fitting is employed to determine a function that accurately characterizes h_{opt} and k_{opt}. These functions are

$$h_{opt} = \sqrt{\frac{R_0 C_t}{R_t C_0}} \frac{1}{\left[1 + 0.16(T_{L/R})^3\right]^{0.24}}, \tag{6.36}$$

and

$$k_{opt} = \sqrt{\frac{R_t C_t}{2R_0 C_0}} \frac{1}{\left[1 + 0.18(T_{L/R})^3\right]^{0.3}}. \tag{6.37}$$

These closed form solutions are highly accurate with an error in the total propagation delay of the repeater system of less than 0.05% as compared to numerical analysis. These formulae can therefore be considered exact for all practical purposes.

Upon examination of (6.36) and (6.37), h_{opt} and k_{opt} are equal to $h_{opt}(RC)$ and $k_{opt}(RC)$ in (6.29) and (6.30) for the special case of an RC impedance when $L_t \rightarrow 0$ (or

$T_{L/R} \rightarrow 0$). A plot of k_{opt} based on both an *RC* model and an *RLC* model versus $T_{L/R}$ is shown in Figure 6.9. Note that the error between the two cases increases as $T_{L/R}$ increases. This behavior is understandable since inductance effects are more significant as $T_{L/R}$ increases (which increases the error of neglecting L_t). Also note that as $T_{L/R}$ increases (or the inductance effects increase), the number of sections k_{opt} decreases. This behavior is intuitively understandable by referring to the results of Figure 6.6 and noting that $T_{L/R}$ can be expressed as

$$T_{L/R} = \frac{1}{2\alpha_{asym}} \sqrt{\frac{RC}{R_0 C_0}}. \tag{6.38}$$

Note that as α_{asym} decreases, $T_{L/R}$ increases. As shown in Figure 6.6, the dependence of the propagation delay of an *RLC* line on the length of the interconnect is linear when $\alpha_{asym} = 0$ (*i.e.*, very high inductive effects) and quadratic when $\alpha_{asym} \rightarrow \infty$ (*i.e.*, no inductive effects). In general, the dependence of the propagation delay of an *RLC* line on the length of the interconnect is bounded between a linear and quadratic relationship depending on the value of α_{asym}. The improvement achieved by partitioning the line into shorter sections in the *RC* case is primarily due to this quadratic dependence of the propagation delay on l. In the other extreme case where $\alpha_{asym} = 0$, the propagation delay is linear with l and therefore no speed improvement is achieved by dividing the line into shorter subsections. Actually, adding repeaters in this case would only increase the total propagation delay because of the additional gate delay of the repeaters. Thus, as inductance effects increase, the optimum number of repeaters inserted to minimize the total interconnect delay decreases.

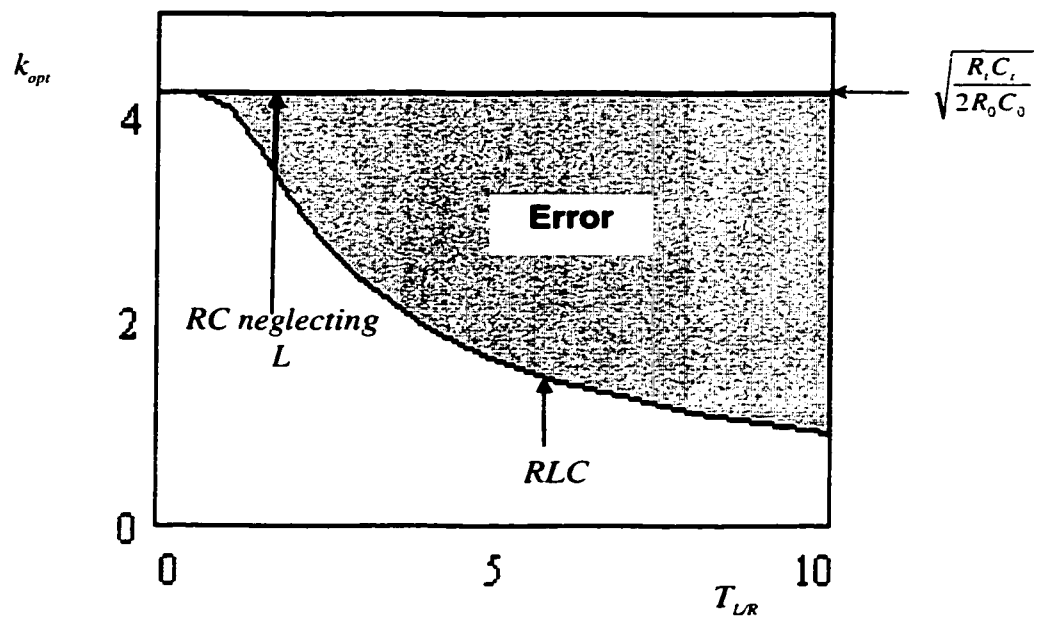


Figure 6.9. The number of sections k_{opt} that minimizes the propagation delay of an *RLC* line as a function of $T_{L/R}$. The cases where the inductance is neglected and where the inductance is included are considered. Note that the error between the two cases increases as $T_{L/R}$ increases.

The per cent increase in $t_{pdtotal}$ caused by neglecting inductance and treating an *RLC* line as an *RC* line as compared to including inductance based on (6.36) and (6.37) for h_{opt} and k_{opt}, respectively, is

$$\%\text{Increase} = \frac{100 * \left[(t_{pdtotal})_{RC} - (t_{pdtotal})_{RLC} \right]}{(t_{pdtotal})_{RLC}}. \qquad (6.39)$$

$(t_{pdtotal})_{RC}$ is calculated by substituting the solution for $h_{opt}(RC)$ and $k_{opt}(RC)$ in (6.29) and (6.30) into $t_{pdtotal}$. $(t_{pdtotal})_{RLC}$ is calculated by substituting the solution for h_{opt} and k_{opt} in (6.36) and (6.37), respectively, into $t_{pdtotal}$. The resulting solution is a function of $T_{L/R}$ only and can be accurately approximated by

$$\%\text{Increase} = \frac{30}{\sqrt{\left(1 + \frac{0.5}{T_{L/R}} + 23e^{-0.8T_{L/R}} + 10^{4}e^{-4T_{L/R}}\right)}}. \tag{6.40}$$

The per cent increase in $t_{pdtotal}$ over the *RLC* case is plotted in Figure 6.10. Note that $(t_{pdtotal})_{RC}$ is larger compared to $(t_{pdtotal})_{RLC}$ as $T_{L/R}$ increases. For $T_{L/R} = 3$, $t_{pdtotal}$ increases by 10%, for $T_{L/R} = 5$, $t_{pdtotal}$ increases by 20%, and for $T_{L/R} = 10$, $t_{pdtotal}$ increases by 30%.

The total area of the buffers in the repeater system is given by $A_{RLC} = h_{opt}*k_{opt}*A_{min}$ and $A_{RC} = h_{opt}(RC)*k_{opt}(RC)\ *A_{min}$ for the *RLC* and the *RC* case, respectively. A_{min} is the area of a minimum size buffer. The per cent area increase *%AI* is characterized by $100*(A_{RC}-A_{RLC})/A_{RLC}$ and is

$$\%AI = 100*\left\{\left[1+0.18(T_{L/R})^3\right]^{0.3} * \left[1+0.16(T_{L/R})^3\right]^{0.24} - 1\right\}. \tag{6.41}$$

The per cent area increase for $T_{L/R} = 3$ is 154% and for $T_{L/R} = 5$ is 435%. Thus, neglecting inductance not only increases the total delay of the repeater system but significantly increases the buffer area as well. This trend is expected since treating the interconnect as an *RC* line and neglecting inductance requires more repeaters. These extra repeaters add to the total delay and buffer area without reducing the line delay because inductance makes the dependence of the delay on the length of the interconnect sub-quadratic. Although the effect of inductance on the power dissipated by the repeater system has not been quantitatively characterized in this chapter, it is expected that considering inductance in the interconnect model would result in a repeater system that consumes less power due to the decreased buffer capacitance and width.

As described in Appendix A, $T_{L/R} > 3$ is common for a wide range of on-chip interconnect and $T_{L/R}$ approaches 10 for wider interconnects commonly seen in a typical 0.25 μm CMOS technology. Thus, the propagation delay of a repeater system can increase in a standard 0.25 μm CMOS technology by up to 30% and the buffer area by up to 15 times if inductance is neglected. Note also that $T_{L/R}$ increases as R_0C_0 decreases. This relation means that as the gate delay decreases, inductance becomes more important. Thus, the effects of inductance in next generation design methodologies will become fundamentally important as technologies scale.

This trend can be explained intuitively by examining the special case of a line with large inductance effects. As discussed before, the minimum total propagation delay can be achieved for such a line by not inserting any buffers independent of the intrinsic speed of the technology. If inductance is ignored and an *RC* model is used for such a line, the number of buffers that are inserted will increase as the buffers become faster since there is less of a penalty for inserting more buffers. Thus, the discrepancy between the buffer solutions based on an *RC* and an *RLC* model (zero buffer area for dominant inductance effects) increases as faster buffers are used. In general, the buffer area required to minimize the total propagation delay based on an *RC* model increases more rapidly when the devices become faster as compared to an *RLC* model.

Finally, in estimating the effect of inductance on repeater insertion, an equivalent linear resistor is used to model the nonlinear CMOS transistors. This linearization of the transistors results in an overestimation of inductance effects. This

behavior can be understood by noting that a transistor in a CMOS gate operates partially in the linear region and partially in the saturation region during switching. In the linear region, the transistor can be accurately approximated by a resistor. However, in the saturation region, the transistor is more accurately modeled as a current source with a parallel high resistance. The Thevinen equivalent of this circuit is a voltage source with a high resistance in series. This high resistance in series with an interconnect line overrides the series resistance and inductance of the line. Thus, the interconnect appears predominantly capacitive when the transistor operates in the saturation region and the effect of inductance (and resistance) is negligible. If the transistor operates in the saturation region during the entire switching time, there is very small error due to neglecting inductance (and resistance). Since the transistor operates partially in the linear region and partially in the saturation region, the inductance effects is limited by the characterizing equations presented in this chapter.

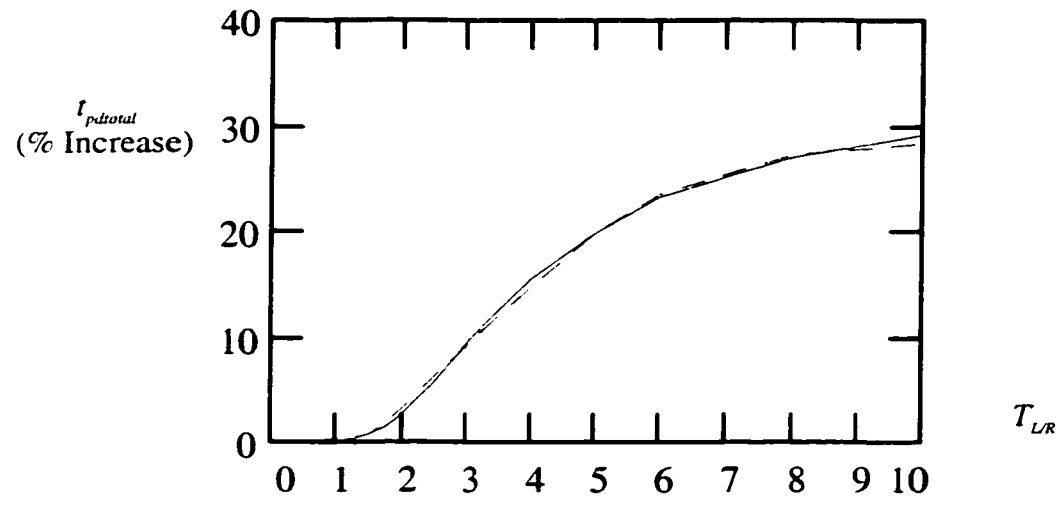


Figure 6.10. The increase in $t_{pdtotal}$ if inductance is neglected as a function of $T_{L/R}$. Numerical solutions are designated by the solid line while (6.40) is designated by the dashed line.

6.3 Conclusions

Closed form solutions for the propagation delay of a gate driving a distributed *RLC* load are presented that are within 5% of AS/X simulations. It is shown that neglecting inductance can cause large errors (over 35%) in the propagation delay for current on-chip interconnect. It is also shown that the traditional quadratic dependence of the propagation delay on the length of the interconnect for *RC* lines tends to a linear dependence as inductance effects increase. This behavior is expected to have a profound effect on future high speed CMOS technologies.

Closed form solutions are presented for inserting repeaters into *RLC* lines that are highly accurate with respect to numerical solutions. The process of inserting repeaters into *RLC* lines increases the propagation delay by up to 30% if inductance is neglected as compared to applying a distributed *RLC* impedance model of the interconnect. Thus, incorporating inductance into the impedance model of the interconnect is of crucial importance for estimating and minimizing the propagation delay of on-chip interconnect. This importance is expected to increase as the gate parasitic impedances decrease and as technologies increase in speed. Future work includes using more accurate gate models, determining delay formulae for *RLC* trees, and characterizing the effects of inductance on repeater insertion in tree structured on-chip interconnect.

Chapter 7 Equivalent Elmore Delay for *RLC* Trees

An interconnect line in a VLSI circuit is in general structured like a tree or a zigzag line rather than like a single continuous line as shown in Figure 7.1. Thus, the process of characterizing signal waveforms in tree and zigzag structured interconnect is of primary importance. One of the more popular delay models used within industry for analyzing the temporal properties of *RC* trees is the Elmore delay model [73], [74]. Despite not being highly accurate, the Elmore delay is widely used by industry for fast delay estimation. With IC's composed of tens of millions of gates it is often impractical to use highly accurate, time consuming methods to evaluate the delay at each node in the circuit. The Elmore delay model is therefore used to quickly estimate the relative delays of different paths in the circuit, permitting more exhaustive timing simulations to be performed for only the critical paths. Also, the Elmore delay is widely used as a delay model for the synthesis of VLSI circuits such as buffer insertion in *RC* trees and wire sizing [24]-[32]. The wide use of the Elmore delay as a basis for design methodologies is primarily because the Elmore delay has a high degree of *fidelity* [31]: an optimal or near-optimal solution achieved by a design methodology based on the Elmore delay is also near-optimal based on a more accurate (*e.g.*, SPICE-computed [127]) delay for routing constructions [31] and wire

sizing optimization [30]. Simulations [32] have shown that the clock skew derived under the Elmore delay model has a high correlation with SPICE-derived skew data.

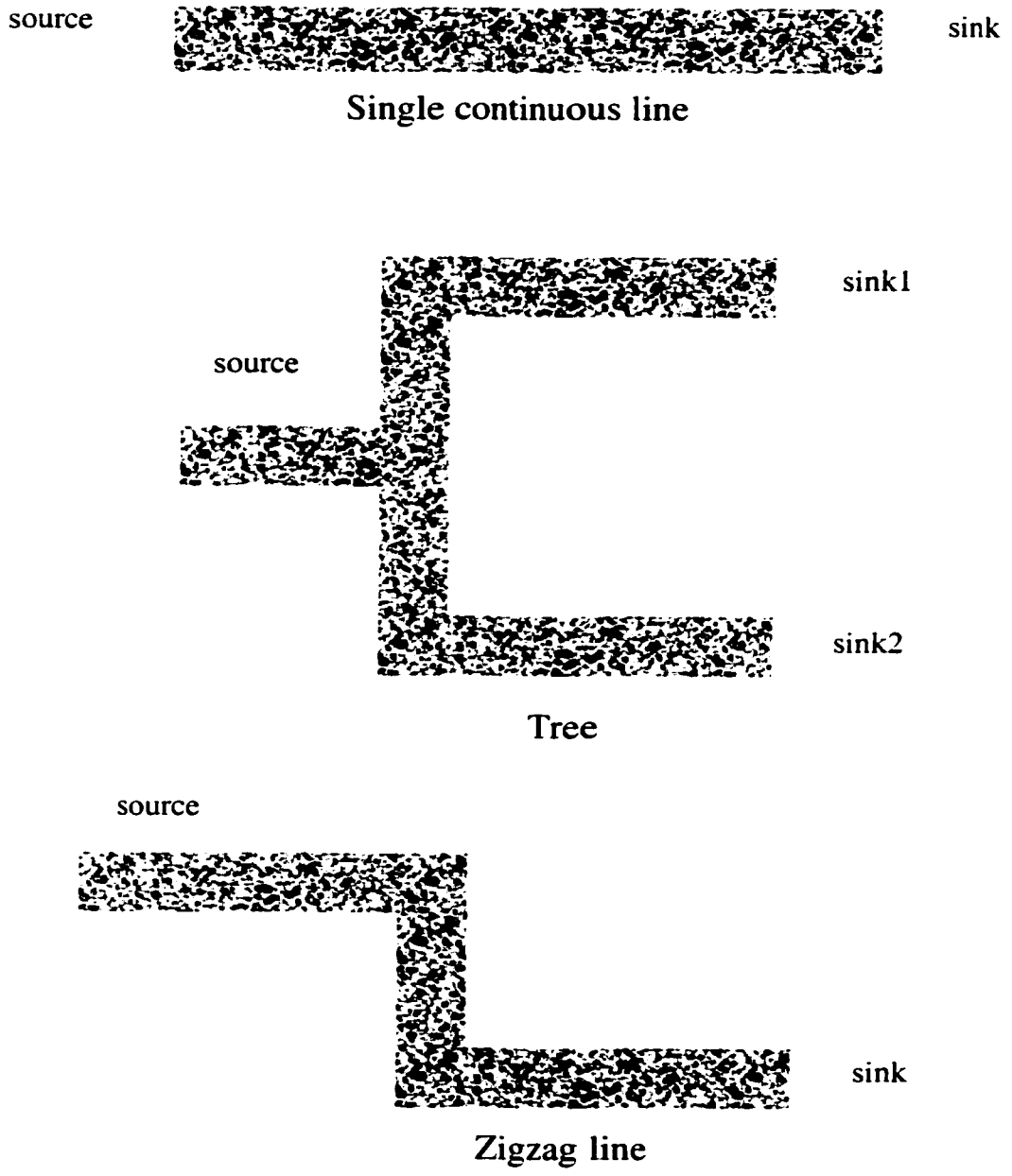


Figure 7.1. Common interconnect structures in an integrated circuit

The popularity of the Elmore delay is mainly due to the existence of a simple tractable formula for the delay [35] that has recursive properties [37], making the calculation of the circuit delays highly efficient even in large circuits. No formula for delay calculation has been determined for *RLC* trees that maintains all the characteristics of the Elmore delay. The absence of an equivalent delay model for *RLC* trees is primarily due to the fact that the Elmore delay does not cover non-monotone responses [73] which can occur in *RLC* circuits. The work described in [134] uses the first and second moments to characterize the response of *RLC* trees. However, the solutions in [134] are composed of three different formulae for the cases of real, complex, and multiple poles and there are no closed form solutions for the moments of a tree that can be directly incorporated into the delay model. Furthermore, the solutions in [134] only characterize a step input response and do not characterize the overshoots and settling time of an underdamped response.

In this chapter, closed form solutions for the 50% delay, rise time, overshoots, and settling time of signals in an *RLC* tree are presented. These solutions have the same accuracy characteristics of the Elmore delay for *RC* trees and preserves the simplicity and recursive characteristics of the Elmore delay. Specifically, the complexity of calculating the time domain responses at all the nodes of an *RLC* tree is linearly proportional to the number of branches in the tree (see Appendix C) and the solutions are always stable. The closed form expressions introduced here consider all damping conditions of an *RLC* circuit including the underdamped response, which is not considered by the Elmore delay due to the non-monotone nature of the response. The continuous analytical nature of the solutions makes these expressions suitable for

design methodologies and optimization techniques. Also, the solutions have significantly improved accuracy as compared to the Elmore delay for an overdamped response. The solutions introduced here for *RLC* trees can be practically used for the same purposes that the Elmore delay is used for *RC* trees.

This chapter is organized as follows. In section 7.1, an equivalent second order approximation of an *RLC* tree is developed. Closed form solutions for the 50% delay, rise time, overshoots, and settling time of the signals within an *RLC* tree are introduced in section 7.2. Accuracy characterization of the proposed delay model is presented in section 7.3. Finally, some conclusions are offered in section 7.4.

7.1 Second Order Approximation for *RLC* Trees

As mentioned in Chapter 3, the Elmore (Wyatt) delay does not properly characterize *RLC* networks due to the possibility of a non-monotone response of an *RLC* network. To illustrate this point, consider the simple single section *RLC* circuit depicted in Figure 7.2. This circuit has a second order transfer function that can be characterized by

$$g(s) = \frac{1}{s^2 LC + sRC + 1}. \tag{7.1}$$

Note that the coefficient of s^1 is RC, which does not include the inductance L. This coefficient of the Elmore time constant (and thus the Wyatt approximation) does not depend on the inductance. However, inductance can have a significant effect on the response of the circuit. To better observe the effect of inductance, the transfer function of the circuit can be reconfigured as

$$g(s) = \frac{\omega_n^2}{s^2 + s2\zeta\omega_n + \omega_n^2}, \tag{7.2}$$

where

$$\zeta = \frac{1}{2}\frac{RC}{\sqrt{LC}}, \tag{7.3}$$

$$\omega_n = \frac{1}{\sqrt{LC}}. \tag{7.4}$$

The poles of the transfer function are

$$P_{1,2} = \omega_n[-\zeta \pm \sqrt{\zeta^2 - 1}]. \tag{7.5}$$

Note that if ζ is less than one, the poles are complex and oscillations occur in the response which violates the monotone response condition of the Elmore delay. In this case, the response is underdamped and overshoots occur. If ζ is greater than one, the poles are real and the response is an overdamped response. If ζ is equal to one, the response is a critically damped response. ζ is called the damping factor of the system. From (7.3), as the inductance increases, ζ decreases which violates the assumption of a monotonic response.

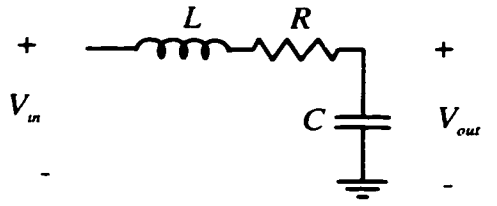


Figure 7.2. Simple *RLC* circuit.

At least a second order approximation is required to characterize a non-monotone response, because a non-monotone response involves complex poles which appear in conjugate pairs in a real system. Thus, a second order system such as (7.2) can be used to approximate a system with a non-monotone response. It is therefore necessary to determine ζ and ω_n in order to make the second order approximation as accurate as possible as compared to the exact transfer function. The transfer function in (7.2) can be expanded in powers of s where the first two moments of the transfer function are equated to the first two moments of the system, m_1 and m_2. The expansion of the transfer function in (7.2) is

$$g(s) = 1 - s\left(\frac{2\zeta}{\omega_n}\right) + s^2\left(\frac{-1+(2\zeta)^2}{\omega_n^2}\right) - \ldots = 1 + m_1 s + m_2 s^2 + \ldots\ldots \tag{7.6}$$

The parameters that characterize the second order approximation of a non-monotonic system, ζ and ω_n, can be calculated in terms of the moments of the non-monotonic system and are

$$\zeta = \frac{-m_1}{2}\frac{1}{\sqrt{m_1^2 - m_2}}, \tag{7.7}$$

$$\omega_n = \frac{1}{\sqrt{m_1^2 - m_2}}. \tag{7.8}$$

Hence, for a system with a non-monotonic response a second order approximation can be found if the first and second moments of the system are known.

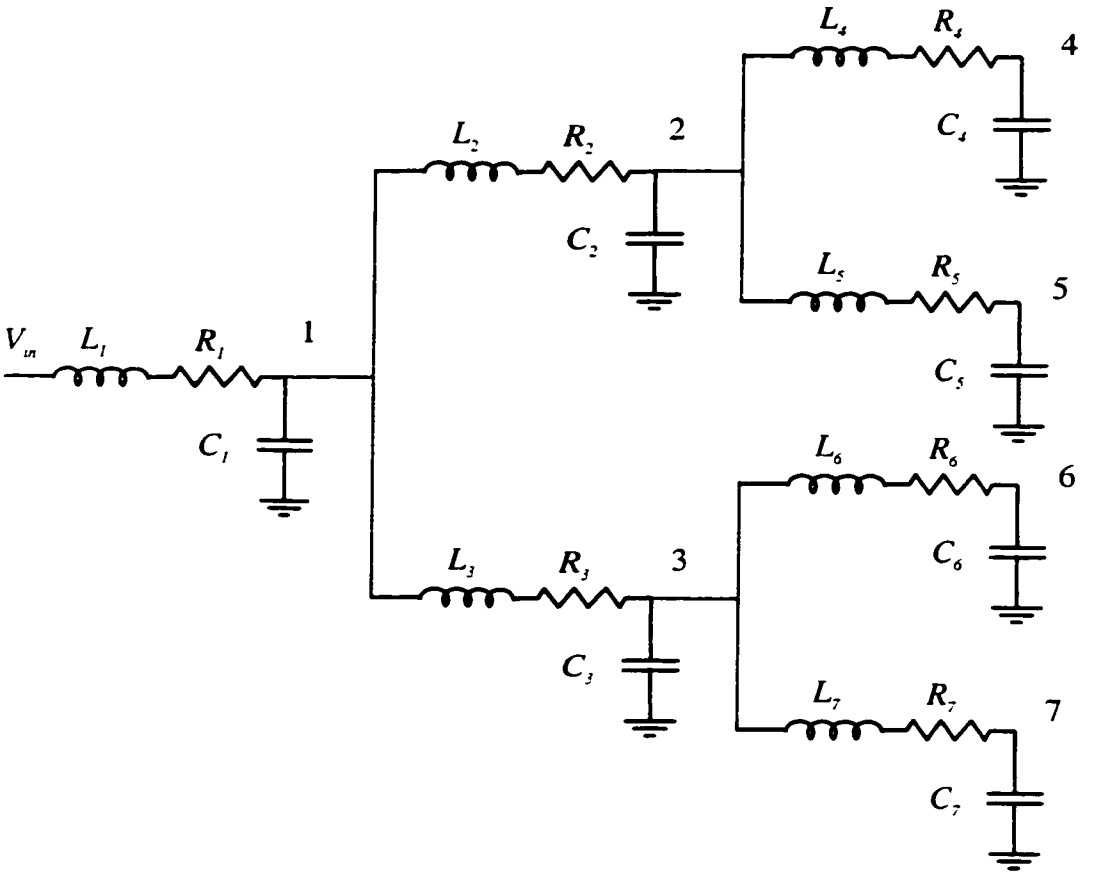


Figure 7.3. General *RLC* tree.

For the general *RLC* tree shown in Figure 7.3, the voltage drop at any node i as compared to the input voltage is

$$V_{in}(s) - V_i(s) = \sum_k C_k V_k(s) s[R_{ki} + L_{ki} s]. \tag{7.9}$$

If the input is a unit impulse, $V_{in}(s)$ is equal to 1.0 and the voltages at the nodes of the tree are the unit impulse responses of these nodes. Thus, the normalized transfer function $g_i(s)$ at node i is given by $V_i(s)$ and is

$$g_i(s) = 1 - \sum_k C_k V_k(s) s[R_{ki} + L_{ki} s] = 1 + m_1^i s + m_2^i s^2 + \ldots . \tag{7.10}$$

The first and second moments at node i can be derived from

$$m_1^i = \left. \frac{dg_i(s)}{ds} \right|_{s=0} , \tag{7.11}$$

$$m_2^i = \left. \frac{1}{2!} \frac{d^2 g_i(s)}{ds^2} \right|_{s=0} . \tag{7.12}$$

Differentiating (7.10) with respect to s and substituting $s = 0$,

$$m_1^i = -\sum_k C_k R_{ik} V_k(s)\big|_{s=0} , \tag{7.13}$$

$$m_2^i = -\sum_k C_k R_{ik} \left. \frac{dV_k(s)}{ds} \right|_{s=0} - \sum_k C_k L_{ik} V_k(s)\big|_{s=0} . \tag{7.14}$$

Note that $V_i(s)\,|_{s=0}=1$ and $dV_i(s)/ds\,|_{s=0}=m_1^k$ since $V_k(s) = g_k(s) = 1 + m_1^k s + m_2^k s^2 + \ldots .$

Thus, the first and second moments of a general *RLC* tree at node i are

$$m_1^i = -\sum_k C_k R_{ik} , \tag{7.15}$$

$$m_2^i = \sum_k \sum_j C_k R_{ik} C_j R_{kj} - \sum_k C_k L_{ik} . \tag{7.16}$$

Referring to section 3.1, Elmore (Wyatt) model approximates the first term in m_2^i by $\left(\sum_k C_k R_{ik} \right)^2$. A similar approximation is used here. Thus, the second moment is approximated by

$$m_2^i = \left(\sum_k C_k R_{ik} \right)^2 - \sum_k C_k L_{ik} . \tag{7.17}$$

Substituting the first and second moments of a general *RLC* tree into (7.7), ζ_i and ω_{ni} that characterize a second order approximation of the transfer function at node *i* are

$$\zeta_i = \frac{1}{2}\frac{\sum_k C_k R_{ik}}{\sqrt{\sum_k C_k L_{ik}}}, \tag{7.18}$$

$$\omega_{ni} = \frac{1}{\sqrt{\sum_k C_k L_{ik}}}. \tag{7.19}$$

Note the analogy with ζ and ω_n for a single *RLC* section in (7.3) and (7.4). The time constants *RC* and $\sqrt{LC}$ are replaced by the summations of the equivalent time constants in the tree. Note also that (7.18) and (7.19) becomes (7.3) and (7.4), respectively, for a single section. This second order approximation has the same accuracy characteristics as that of the Elmore (Wyatt) approximation for an *RC* tree. The accuracy characteristics of this second order approximation is discussed in section 7.3.

7.2 Signal Characterization in *RLC* Trees for a Step Input

The second order approximation of the transfer function of an *RLC* tree at node *i* described by (7.2), (7.18) and (7.19) can be used to determine the time domain signal at node *i* for an arbitrary input. The Laplace transform of the input is multiplied by the second order approximate transfer function. The inverse Laplace transform is calculated for the resulting expression to determine the time domain signal. After

determining an expression that describes the time domain signal at node i of an RLC tree, an iterative method is applied to calculate the primary parameters that characterize the time domain response such as the 50% propagation delay and the 90% rise time. However, for the special case of a step input, these parameters can be calculated directly without applying the aforementioned procedure due to the mathematical nature of the time domain signal.

For a step input and a supply voltage of V_{DD}, the time domain response at node i derived from the second order approximation is

$$S_i(t) = V_{DD} + \frac{V_{DD}}{2\sqrt{\zeta_i^2 - 1}}\left[\frac{\exp[\omega_{ni}t(-\zeta_i + \sqrt{\zeta_i^2 - 1})]}{-\zeta_i + \sqrt{\zeta_i^2 - 1}} - \frac{\exp[\omega_{ni}t(-\zeta_i - \sqrt{\zeta_i^2 - 1})]}{-\zeta_i - \sqrt{\zeta_i^2 - 1}}\right]. \quad (7.20)$$

The rise time is defined here as the time for the signal to rise from 10% to 90% of the final value. Also, the overshoots and the settling time for the case of an underdamped response are characterized. In the step response in (7.20), note that time is always multiplied by ω_{ni}. Thus, if the time is scaled by ω_{ni}, the step response at node i with a supply voltage of V_{DD} volts becomes a function of only one variable ζ_i and is

$$S_i'(t) = V_{DD} + \frac{V_{DD}}{2\sqrt{\zeta_i^2 - 1}}\left[\frac{\exp[t'(-\zeta_i + \sqrt{\zeta_i^2 - 1})]}{-\zeta_i + \sqrt{\zeta_i^2 - 1}} - \frac{\exp[t'(-\zeta_i - \sqrt{\zeta_i^2 - 1})]}{-\zeta_i - \sqrt{\zeta_i^2 - 1}}\right]. \quad (7.21)$$

where $S_i'(t)$ is the time scaled response at node i and t' is time scaled by ω_{ni}. The time scaled 50% delay and rise time can be calculated by equating $S_i'(t)$ to $0.5V_{DD}$, $0.1V_{DD}$, and $0.9V_{DD}$, respectively. The time scaled 50% delay at node i and the rise time are only functions of one variable ζ_i. The 50% delay and the rise time calculated for several values of ζ_i are plotted as functions of ζ_i in Figure 7.4. A curve fitting method

is applied to characterize the time scaled 50% delay and rise time as functions of ζ_i and these functions are

$$t'_{pdi} = 1.047e^{-\frac{\zeta_i}{0.85}} + 1.39\zeta_i, \tag{7.22}$$

$$t'_{ri} = 6.017e^{-\frac{\zeta_i^{1.35}}{0.4}} - 5e^{-\frac{\zeta_i^{1.25}}{0.64}} + 4.39\zeta_i, \tag{7.23}$$

where t'_{pdi} and t'_{ri} are the time scaled 50% delay and rise time at node i, respectively. The 50% delay and rise time at node i can be determined by dividing t'_{pdi} and t'_{ri} by ω_{ni} and are

$$t_{pdi} = (1.047e^{-\frac{\zeta_i}{0.85}} + 1.39\zeta_i)/\omega_{ni}, \tag{7.24}$$

$$t_{ri} = (6.017e^{-\frac{\zeta_i^{1.35}}{0.4}} - 5e^{-\frac{\zeta_i^{1.25}}{0.64}} + 4.39\zeta_i)/\omega_{ni}. \tag{7.25}$$

Note that the 50% delay and the rise time at node i can be described as

$$t_{pdi} = (1.047e^{-\frac{\zeta_i}{0.85}})/\omega_{ni} + 0.695\sum_k C_k R_{ik}, \tag{7.26}$$

$$t_{ri} = (6.017e^{-\frac{\zeta_i^{1.35}}{0.4}} - 5e^{-\frac{\zeta_i^{1.25}}{0.64}})/\omega_{ni} + 2.195\sum_k C_k R_{ik}. \tag{7.27}$$

For large ζ_i (low inductance effects), these solutions become the Elmore (Wyatt) approximation of the 50% delay and the rise time for an *RC* tree at node i. This relationship between (7.26) and (7.27) for large ζ_i and the Elmore (Wyatt) delay demonstrates that the general solutions for the 50% delay and the rise time introduced here include the Elmore (Wyatt) delay for the special case of an *RC* tree. Note also that the general solutions introduced here include all types of responses

(underdamped non-monotone, critically damped, and overdamped) in one continuous equation, which is useful in applications such as buffer insertion, wire sizing, and other VLSI-based design, synthesis, and analysis methodologies.

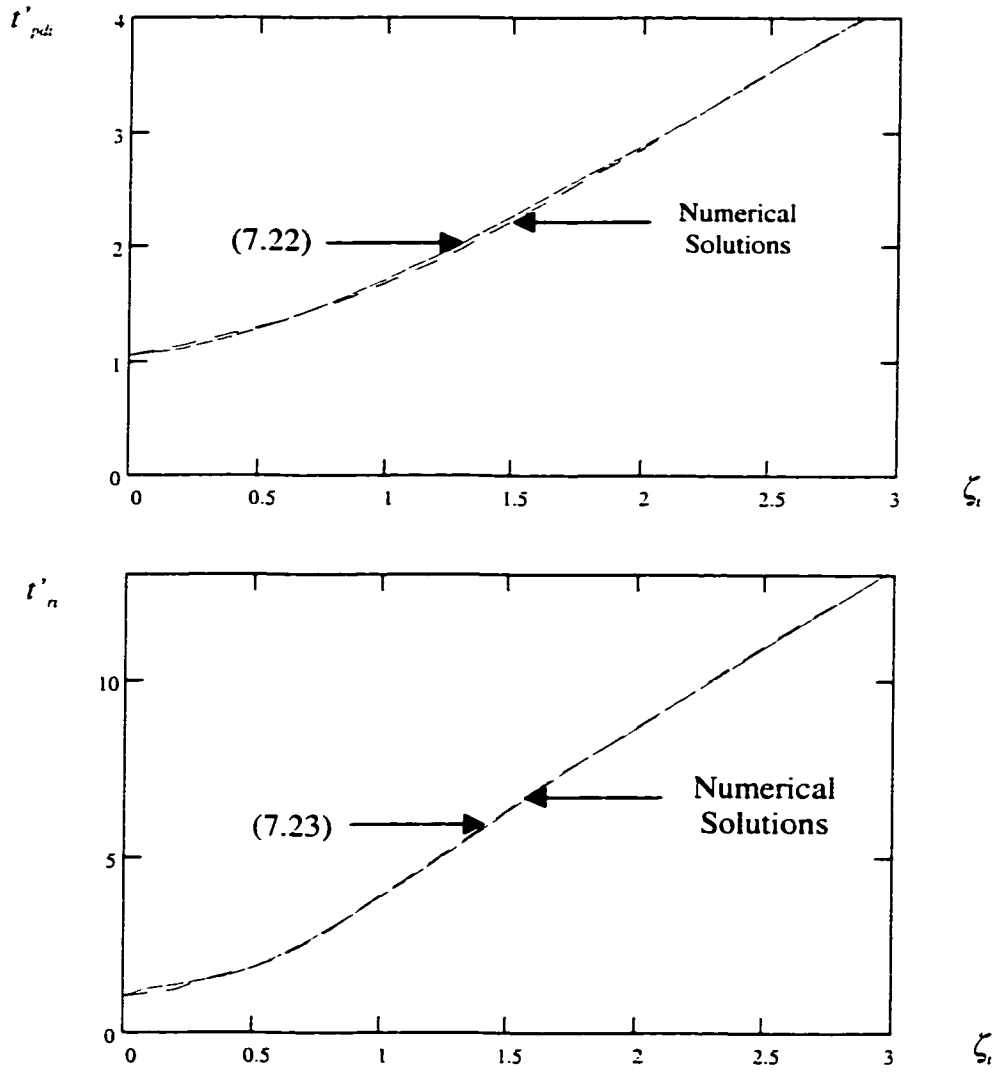


Figure 7.4. The time scaled 50% delay and rise time, t'_{pdi} and t'_n, versus ζ_i. (7.22) and (7.23) are also shown.

For the case of an underdamped non-monotone response when $\zeta_i < 1$ (see Figure 7.5), overshoots and undershoots occur which must also be characterized. Also, another parameter can be used to characterize non-monotone responses and is defined as the time when the oscillations about the steady state are smaller than x of the steady state value. This parameter is usually called the settling time and x is typically chosen to be 0.1 [135]. The value of the maximum or minimum oscillations can be found by differentiating (7.20) with respect to time and equating the result to zero. The values for the maximum or minimum oscillations at node i as a percentage of the final value are given by

$$\%O_i = (-1)^{n+1} \cdot 100 \exp\left(-\frac{n\pi\zeta_i}{\sqrt{1-\zeta_i^2}} \right) \qquad n = 1, 2, \ldots, \tag{7.28}$$

where $\%O_i$ represents the maximum overshoots for n odd and minimum undershoots for n even at node i. The time at which the n^{th} overshoot occurs at node i is

$$t_{Oi} = \frac{n\pi}{\omega_{ni}\sqrt{1-\zeta_i^2}}. \tag{7.29}$$

The settling time can be calculated by equating $\%O_i$ to x*100 to determine n which represents the first overshoot that is less than x times the steady state value. The time of this overshoot is the settling time and can be calculated by substituting n calculated from $\%O_i = x$*100 in (7.30). Thus, the settling time at node i is

$$t_{si} = \frac{-\ln(x)}{\zeta_i \omega_{ni}}. \tag{7.30}$$

For $x = 0.1$, t_{si} is

$$t_{si} = \frac{2.3}{\zeta_i \omega_{ni}}. \tag{7.31}$$

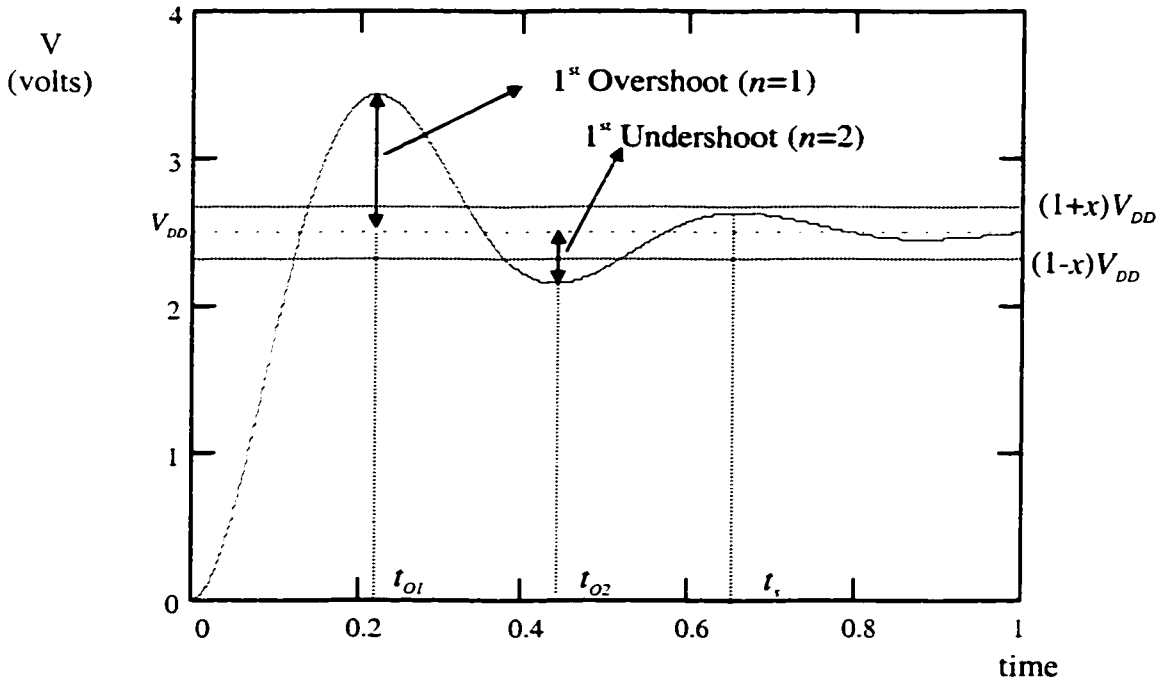


Figure 7.5. Characterization of an underdamped response. V_{DD} is the supply voltage. x is the ratio of the final value which bounds the oscillations for the response to be considered settled. The times t_{O1}, t_{O2}, ... are the times at which the overshoots and undershoots occur. t_s is the settling time.

7.3 Accuracy Characterization of the Second Order Approximation

The accuracy characteristics of the second order approximation introduced in section 7.1 are discussed and explained in this section. The effect of the signal applied

at the input of the tree on the accuracy of the second order approximation is discussed in subsection 7.3.1. The effects of the unbalance in impedances within the tree and the branching factor for balanced trees are discussed in subsections 7.3.2 and 7.3.3, respectively. The effect of the depth of the tree is discussed in subsection 7.3.4. The effect of the position with respect to the source of the node at which the response is evaluated is presented in subsection 7.3.5. Finally, the effect of higher order oscillations in the response is discussed in subsection 7.3.6. In general, the approximation introduced here for *RLC* trees has the same accuracy characteristics as that of the Elmore (Wyatt) delay for *RC* trees. Expression (7.24) in section 7.2 is used to calculate the propagation delay throughout this section.

7.3.1 Effect of the Input Waveform Shape

As mentioned in section 7.2, the second order approximation introduced in this chapter in (7.2), (7.18) and (7.19) can be used to calculate the time domain response of an arbitrary input signal. The error of the time domain response calculated using the second order approximation as compared to AS/X [128] simulations is dependent on the characteristic of the input signal. More specifically, the calculated time domain response becomes more accurate as the rise time of the input signal increases. To illustrate this behavior, an exponential input signal of the form,

$$V_{in}(t) = V_{DD}[1 - \exp(-t/\tau)]u(t), \tag{7.32}$$

is applied to the second order approximation where $u(t)$ is the unit step function, V_{DD} is the supply voltage, and the 90% rise time of the input signal is 2.3τ. τ is the time

constant of the exponential in (7.16). Note that an exponential signal more accurately characterizes the signals in VLSI circuits as compared to a ramp input signal. The time domain response at node *i* of an *RLC* tree for this exponential input is

$$e_{i_{RLC}}(t) = V_{DD}\left[1 - ke^{-\frac{t}{\tau}} + \frac{e^{-\zeta_i \omega_{ni} t}}{\sqrt{1-\zeta_i^2}}\left[\sin(\omega_{ni} t - \theta_1) - \sqrt{\frac{1}{k}}\sin(\omega_{ni} t - \theta_2)\right]\right], \tag{7.33}$$

where

$$\theta_1 = \tan^{-1}\left[\frac{\sqrt{1-\zeta_i^2}}{\zeta_i}\right], \tag{7.34}$$

$$\theta_2 = \tan^{-1}\left[\frac{\left(\tau/T_{LCi}\right)\sqrt{1-\zeta_i^2}}{\left(\tau/T_{LCi}\right)\zeta_i - 1}\right], \tag{7.35}$$

and

$$k = \frac{\left(\tau/T_{LCi}\right)^2}{\left(\tau/T_{LCi}\right)^2 - 2\zeta\left(\tau/T_{LCi}\right) + 1}. \tag{7.36}$$

T_{LCi} is

$$T_{LCi} = \sqrt{\sum_k C_k L_{ik}}\ . \tag{7.37}$$

This closed form time domain solution is evaluated for output O_2 of the *RLC* tree shown in Figure 7.6 and is compared to AS/X [128] simulations in Figure 7.7. Note in Figure 7.7 that as the rise time of the input signal increases as compared to T_{LCi}, the calculated time domain response becomes more accurate. This relationship is intuitive since the closed form solution accurately captures the characteristics of the input signal. As the input rise time increases as compared to the time constants of the

impedances within the *RLC* tree, the dependence of the output response on the input signal increases as compared to the dependence on the characteristics of the *RLC* tree. Hence, the output response becomes more accurate when the response is dominated by the input characteristics, which are accurately captured by the closed form solution. Thus, an argument can be made that the time domain response calculated using the second order approximation introduced here is largest for a step input (which has a zero rise time).

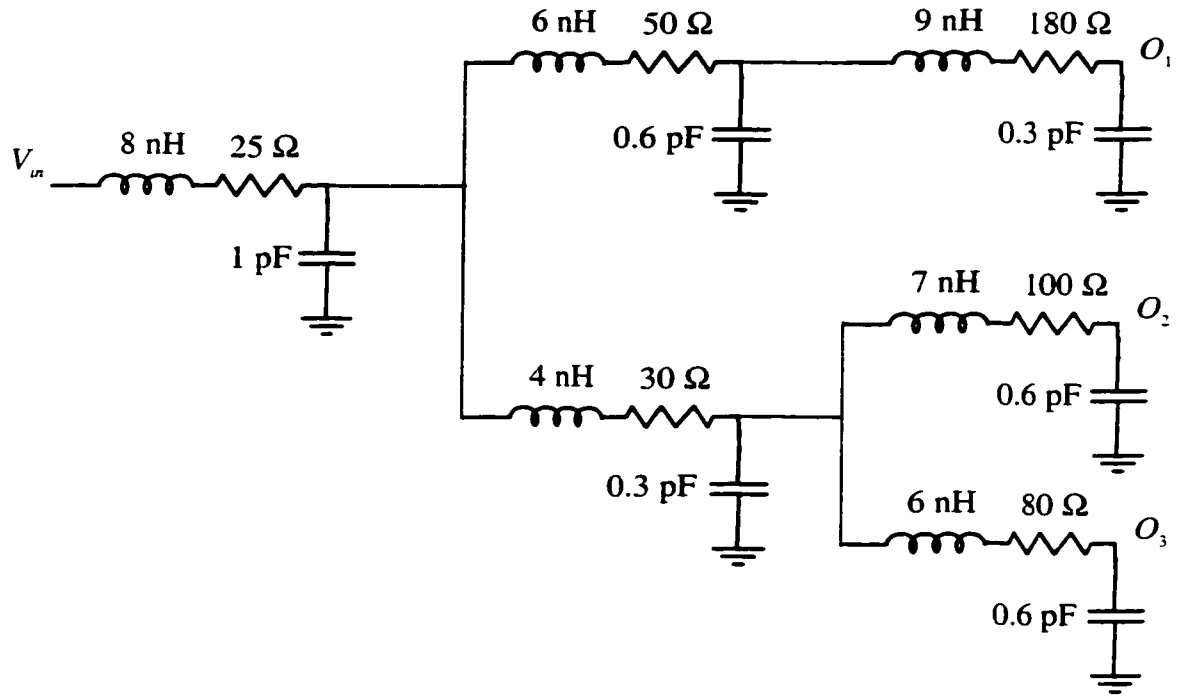


Figure 7.6. An example of an *RLC* tree

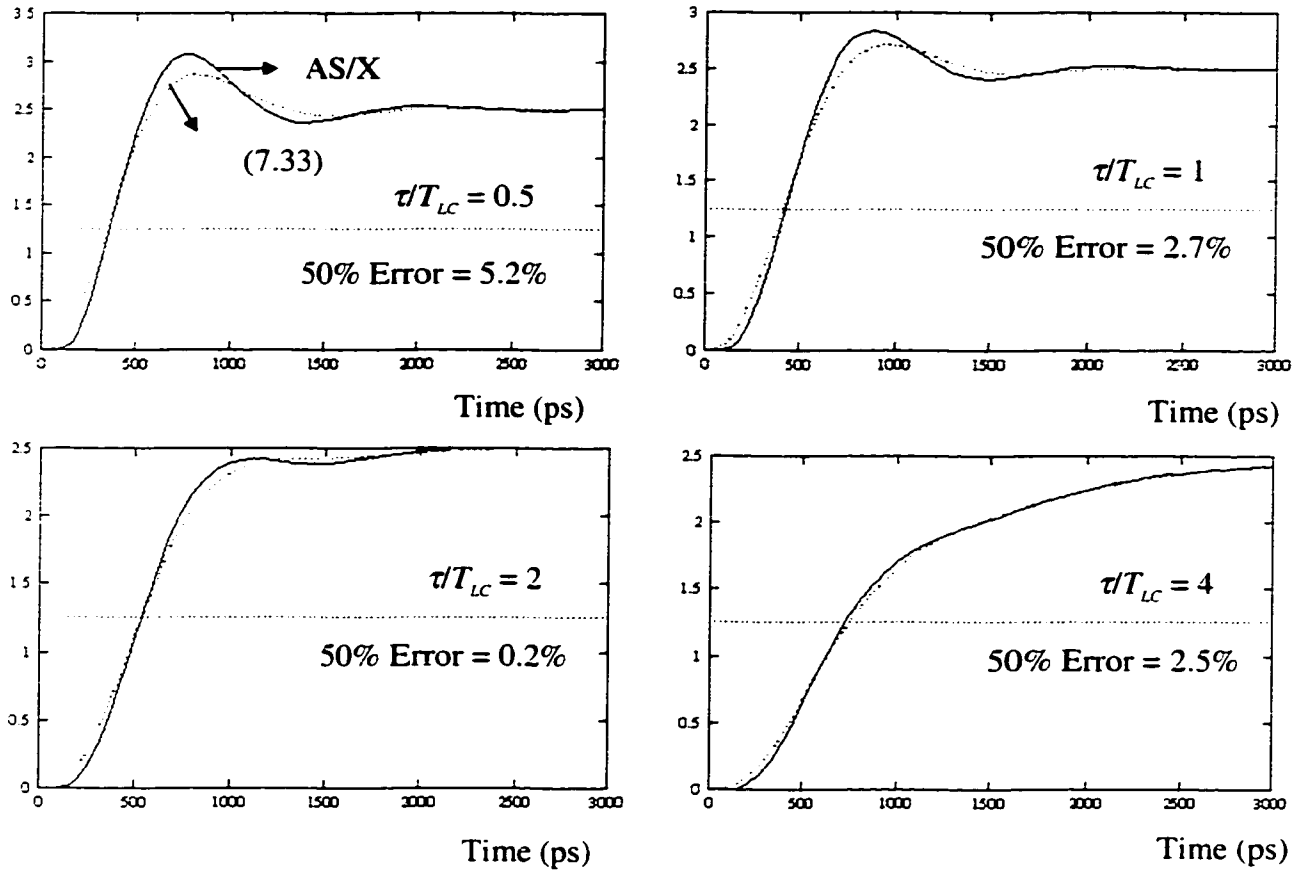


Figure 7.7. Simulations of the time domain response for output O_2 of the tree shown in Figure 7.6 as compared to the closed form solution in (7.32) for different input rise times.

7.3.2 Effect of Unbalanced Impedances within an *RLC* Tree

A balanced tree is a tree where the impedances of the *RLC* sections that constitute each level are equal, making the paths to all the sinks identical. For example, the tree shown in Figure 7.3 is balanced if the *RLC* sections, 2 and 3, which constitute the second level of the tree are identical and the *RLC* sections, 4, 5, 6, and 7, which constitute the third level are identical. If the tree in Figure 7.3 is not balanced, the transfer function at any of the sinks (nodes 4, 5, 6, or 7) is of order 14 since the tree has seven capacitors and seven inductors. The transfer function at any of the sinks has six of the 14 zeros (the total number of zeros is always equal to the total number of poles) at infinity since there are three shunt capacitors and three series inductors from the input to each sink. The remaining eight zeros are finite zeros making the order of the numerator eight. When the tree is balanced, an exact calculation of the transfer function illustrates that the eight finite zeros of the transfer function coincide with eight of the poles. These eight poles and zeros cancel, leaving the transfer function at the sinks only of order six with no finite zeros. To better interpret this behavior, note that nodes 2 and 3 can be shunted when the tree shown in Figure 7.3 is balanced due to symmetry without affecting the response at any node of the tree. Also, nodes 4, 5, 6, and 7 can be shunted due to symmetry. Thus, the *RLC* tree shown in Figure 7.3 is equivalent to the ladder circuit shown in Figure 7.8 after calculating the equivalent impedance of the parallel *RLC* sections. This ladder circuit has a transfer function of order six at the output with no finite zeros. Note that if the tree has a fourth level, the eight *RLC* sections of that level correspond to one *RLC*

section in the equivalent ladder circuit. In the fifth level, sixteen *RLC* sections correspond to one *RLC* section in the equivalent ladder circuit. Thus, the number of poles of the transfer function at the sinks of a balanced *RLC* tree increases linearly with the number of levels in the tree due to pole-zero cancellation. Note that no finite zeros are added by increasing the number of levels. For an unbalanced *RLC* tree with a binary branching factor, the number of poles and finite zeros at the sinks increases exponentially with the number of levels in the tree. The second order approximation used here has two poles and no finite zeros and more accurately approximates the transfer function of a balanced *RLC* tree than that of an unbalanced tree.

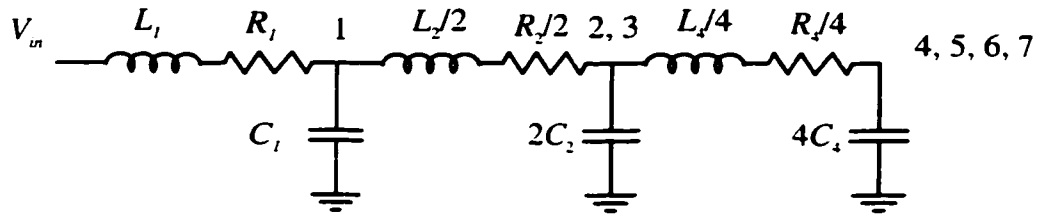


Figure 7.8. Equivalent ladder circuit of the *RLC* tree shown in Figure 7.3 when the tree is balanced

The closed form solution is compared to AS/X [128] simulations of the tree shown in Figure 7.3 at output node 7. The simulations are shown in Figure 7.9 for a balanced tree with several values of ζ_7 (the equivalent damping factor at node 7) and a step input which represents the highest error as discussed in subsection 7.3.1. The Elmore (Wyatt) solution is also shown for comparison. Note the high accuracy that the solution exhibits as compared to the AS/X simulations for the case of a balanced

tree. The error in the propagation delay is less than 4% for this balanced tree example. The accuracy of the solution introduced here deteriorates as the tree becomes more asymmetric. To quantify the error between the closed form solution introduced here and AS/X simulations, simulations and analytic solutions of several asymmetric trees are shown in Figure 7.10. The parameter *asym* is introduced to quantify the relative asymmetry of an *RLC* tree. For example, when *asym* is equal to two, the impedance of the left branch is always twice the impedance of the right branch. The higher *asym*, the higher the asymmetry of the tree. The error in the propagation delay can reach 20% for highly asymmetric trees. The error in the waveform shape is even higher as compared to AS/X simulations. These characteristics, however, are also typical for the Elmore (Wyatt) approximation for *RC* trees.

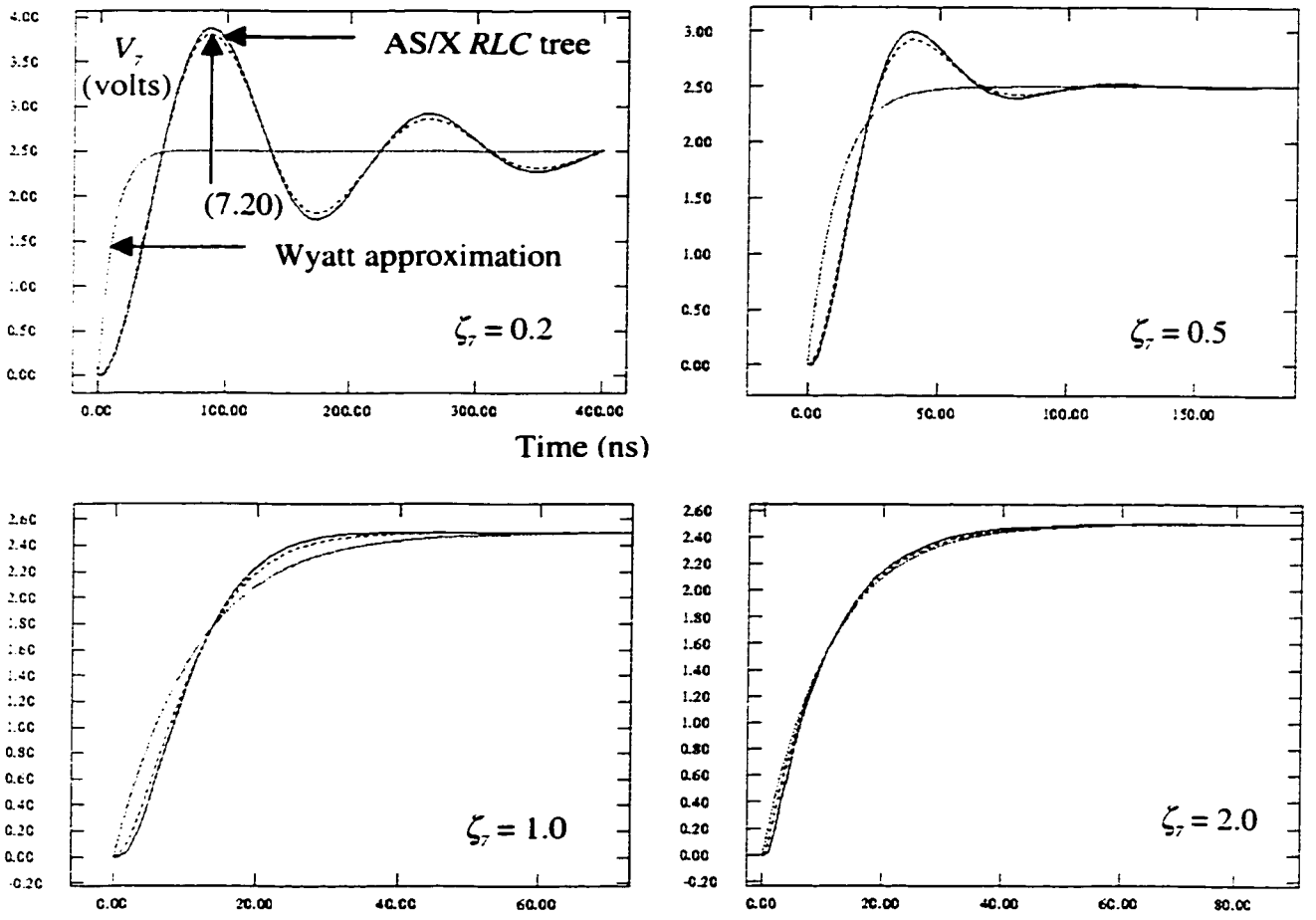


Figure 7.9. AS/X simulations as compared to (7.20) for several values of ζ. The Elmore (Wyatt) solution is also shown. Results are for node 7 shown in Figure 7.3.

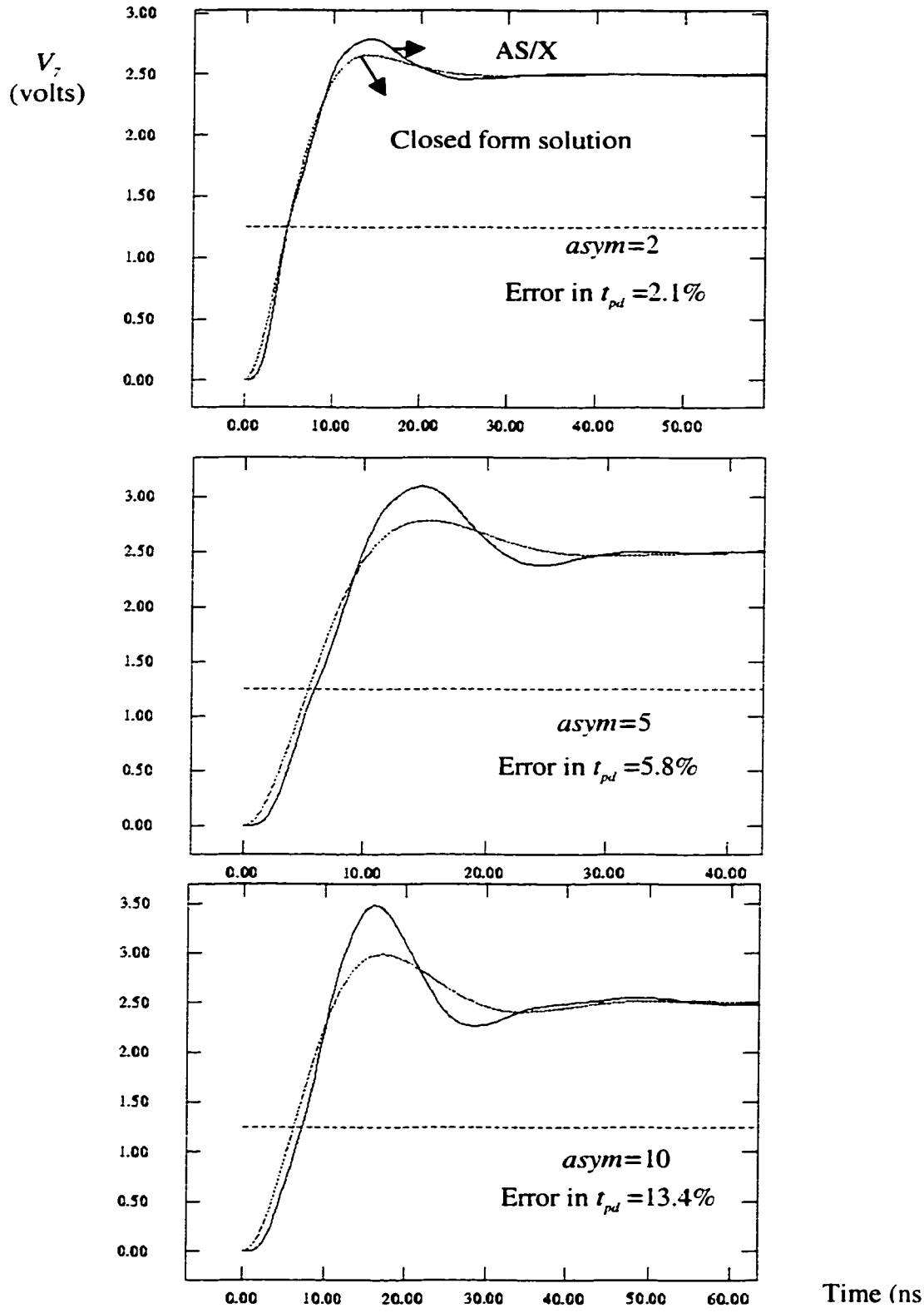


Figure 7.10 AS/X simulations as compared to (7.20) for several asymmetric trees. Results are for node 7 shown in Figure 7.3.

7.3.3 Effect of the Branching Factor for Balanced Trees

An *RLC* tree with a binary branching factor and n levels has $2^n - 1$ branches. As shown in the previous subsection, the tree is equivalent to a ladder circuit with n *RLC* sections if the tree is balanced due to pole-zero cancellation. The second order approximation is more accurate for balanced trees because of this exponential pole-zero cancellation. A tree with a general branching factor B and n levels has $(B^n-1)/(B-1)$ branches. However, if the tree is balanced, the tree is again equivalent to a ladder circuit with n *RLC* sections. Thus, a higher number of zeros are canceled by poles by increasing the branching factor of a balanced tree while keeping the number of sinks constant. For example, a balanced tree with a binary branching factor driving 16 sinks has five levels and is equivalent to a five section ladder circuit. If the same 16 sinks are driven by a balanced tree with a branching factor equal to 16, the tree has only two levels and is equivalent to a two section ladder circuit. Thus, the second order approximation more accurately describes an *RLC* tree with a branching factor equal to 16. AS/X simulations and the closed form solution from (7.20) with a step input for the response at the sinks of both trees are shown in Figure 7.11. In this example, all the *RLC* sections in the binary branching tree has R = 12.5 Ω, L = 5 nH, and C = 1 pF. All the *RLC* sections in the tree with a branching factor of 16 has R = 25 Ω, L = 5 nH, and C = 1 pF. Note that the second order approximation is less accurate in the case of a tree with a binary branching factor.

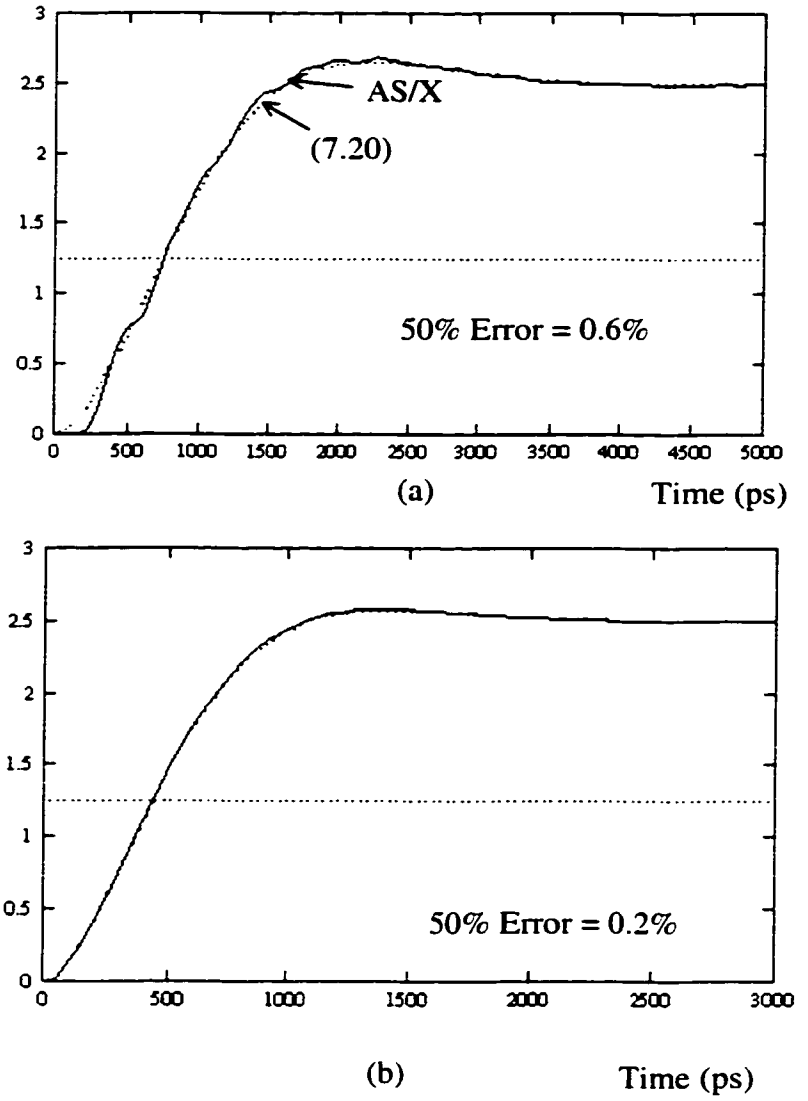


Figure 7.11. AS/X simulations as compared to (7.20) for the response at the 16 sinks of a balanced tree. a) The tree has a binary branching factor. b) The tree has a branching factor of 16.

7.3.4 Effect of the Depth of the Tree

The depth of a tree can be characterized by the number of levels n of the tree. The accuracy of the solution decreases as the number of levels in the tree increases since the order of the transfer function at the sinks increases. The increased error due to increasing the depth of the

tree can be best observed for a balanced tree since the error due to the unbalance overrides the error due to the depth in an unbalanced tree. AS/X simulations are compared to (7.20) in Figure 7.12 for balanced trees with a different number of levels. Note that the error between AS/X and the closed form solution increases as the number of levels of the tree increases. Note also that for a single line, the depth represents the number of sections of the line.

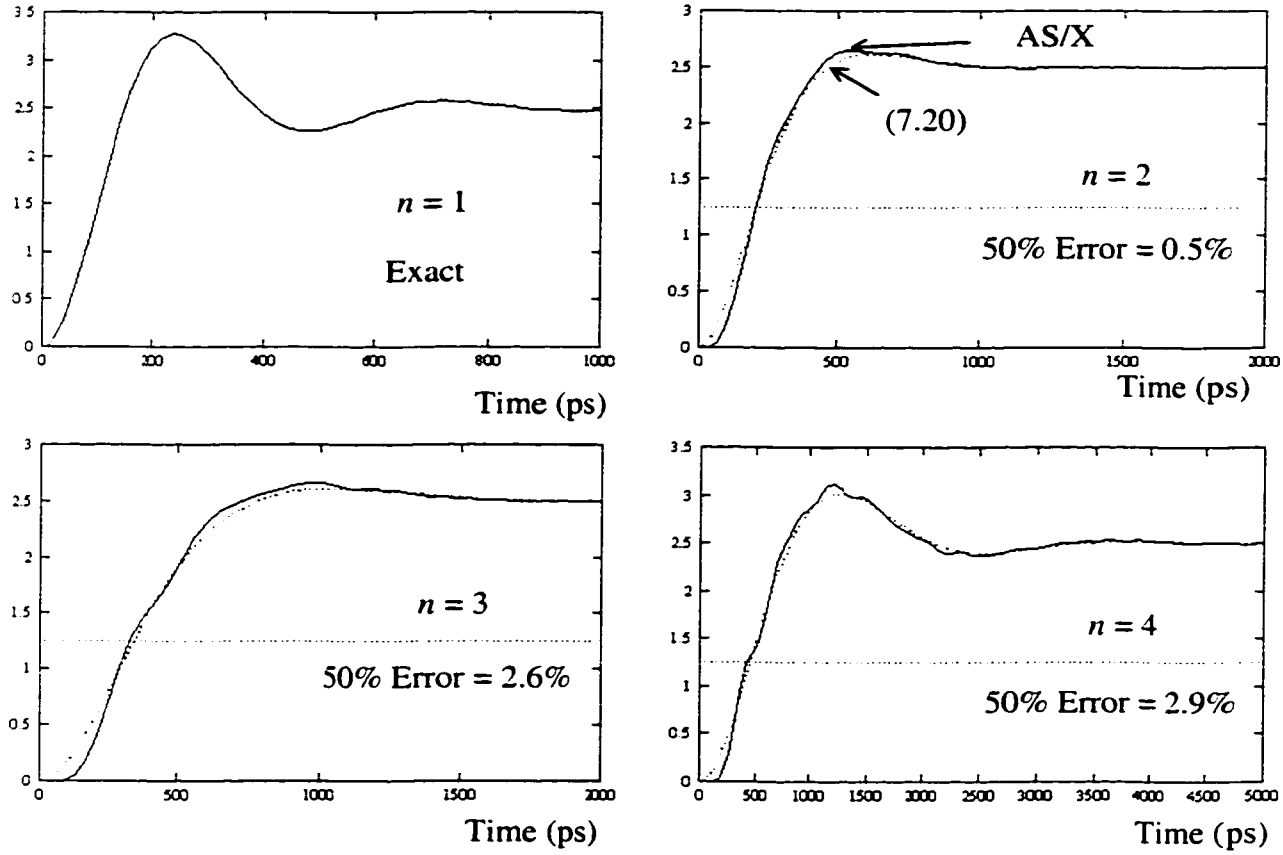


Figure 7.12. AS/X simulations as compared to (7.20) for several balanced trees with different depths. The horizontal dotted line characterizes the 50% threshold voltage.

7.3.5 Effect of the Node Position

The error exhibited by the second order approximation increases as the position of the node at which the response is evaluated moves from the sinks towards the source. This behavior is due to the extra finite zeros in the transfer function since there are less capacitors and inductors in the path from the input to the node at which the response is evaluated. Again, this effect is best observed for a balanced tree. AS/X simulations are compared to (7.20) in Figure 7.13 at several positions of the binary balanced tree with five levels described in subsection 7.3.3. Note that the error between AS/X and the closed form solution is least at the sinks which is typically the location of greatest interest.

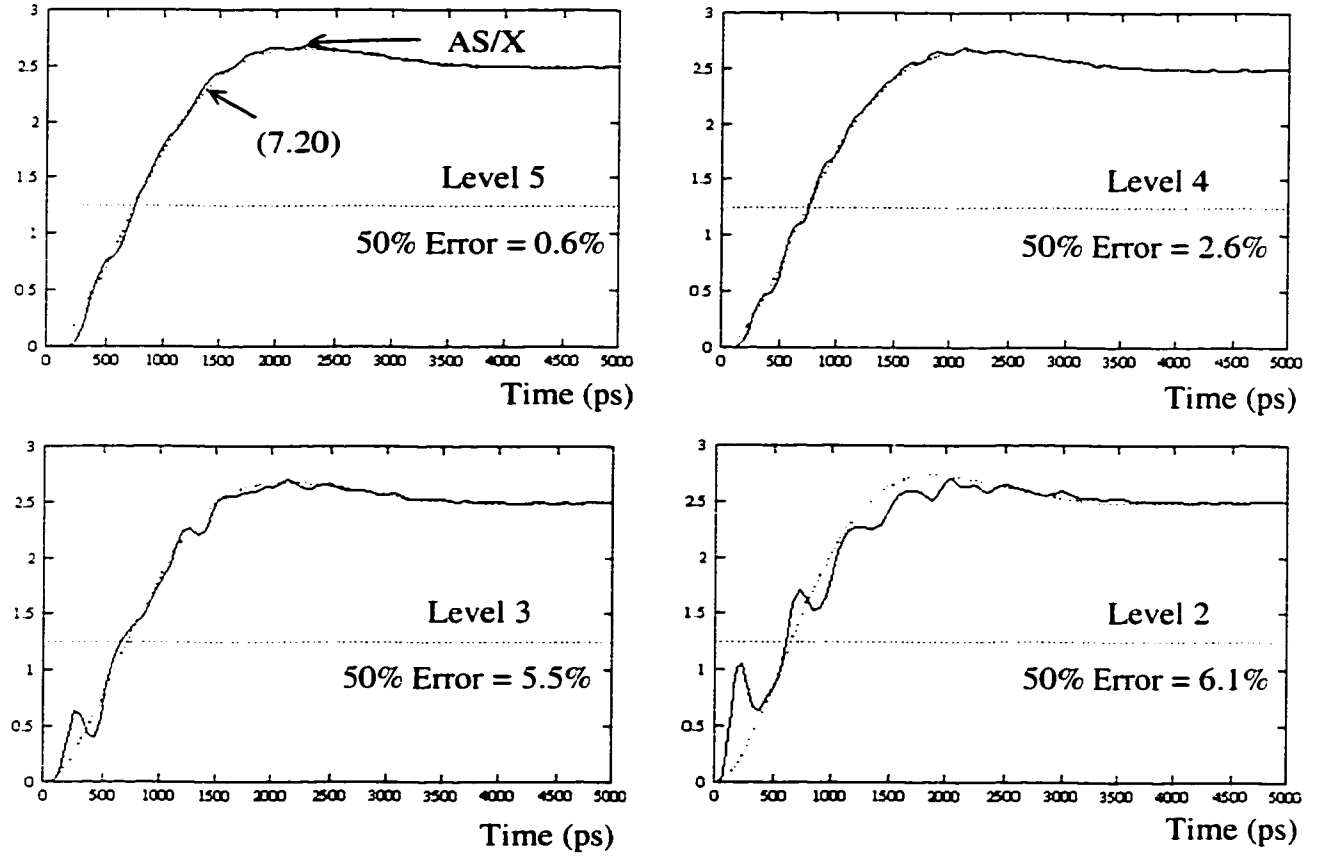


Figure 7.13. AS/X simulations as compared to (7.20) for a binary balanced tree for nodes at different levels within the tree. The horizontal dotted line characterizes the 50% voltage.

7.3.6 Effect of Second Order Oscillations

As an *RLC* tree becomes larger and as the number of levels increase, high frequency oscillations are superimposed over the primary response. For example, in Figure 7.14, the second order approximation (7.20) of the response for a large *RLC* tree is illustrated. Note the overshoots. AS/X simulations are also shown in Figure 7.14 and the actual signal oscillates around the second order approximation with a higher frequency as compared to the frequency of the primary oscillations. The oscillations around the low frequency response characterized by (7.20) are second order oscillations. The second order approximation introduced here cannot accurately model the higher frequency harmonics of the time domain response since the approximation only has two poles. However, the second order approximation can be used effectively to estimate the macro features of the response such as the propagation delay, the rise time, and the primary overshoots. If the fine details of the response are of interest, higher order delay models can be used such as AWE [75]-[79] at the expense of additional processing time, numerical issues, and stability issues. Note that the responses in the simulations presented in this section also exhibit second order oscillations. The second order approximation successfully characterizes the dominant low frequency response.

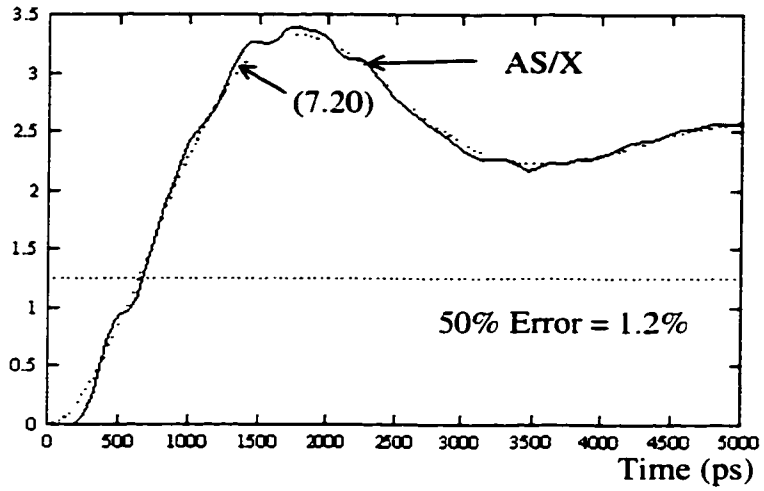


Figure 7.14. AS/X simulations as compared to (7.20) for a large *RLC* tree.

7.4 Conclusions

A general method to characterize the response of a linear non-monotone system that is equivalent to the Elmore delay is presented. The generated delay expressions for an *RLC* tree have the same accuracy characteristics as the Elmore (Wyatt) approximation for *RC* trees. Simple analytical expressions of signals in an *RLC* tree are provided for the 50% delay, the rise time, overshoots, and settling time. These expressions consider both monotone and non-monotone signal responses. The delay expressions are continuous and hence are useful for optimization and synthesis in VLSI-based design methodologies. The second order approximation introduced here is always stable and can be used with arbitrary inputs. Furthermore, the second order approximation is computationally efficient since the number of multiplication operations required to evaluate the approximation at all the nodes of an *RLC* tree is linearly proportional to the number of branches in the tree.

Chapter 8 Inductance Effects in *RLC* Trees

The importance of on-chip inductance for single lines has been characterized in [49]-[51] and [68] (see Chapter 5). However, nets in a VLSI circuit are often structured as a tree rather than as a single line. Also, the clock distribution network, which is common to all synchronous digital circuitry, is typically tree structured. The performance of a VLSI circuit therefore heavily depends upon the design of the clock distribution network where the most accurate interconnect models are required. It is shown in this chapter that a branch of a tree cannot be treated as a single line for the purpose of evaluating inductance effects. Rather, the entire tree should be examined for inductance effects as a single structure since a large interaction occurs among the different branches. It is therefore shown that applying a single line analysis to an *RLC* tree can cause misleading conclusions.

The focus of this chapter is the introduction of simple figures of merit that can be used as criteria to determine which nets (structured as trees) require more accurate *RLC* models. The second order approximation introduced in Chapter 7 is used together with Wyatt's approximation (see section 3.1.2) to derive figures of merit to characterize the effects of inductance at a specific node in an *RLC* tree. The *effective* damping factor of the signal at a specific node within an *RLC* tree is shown to be one useful figure of merit. It is also shown that as the effective damping factor of a signal increases, an *RC* model is sufficiently accurate to characterize the waveform. The rise time of the input signal driving an *RLC* tree is shown to be a second factor that affects

the relative significance of inductance. As the rise time of the input signal increases as compared to the effective *LC* time constant at a specific node within an *RLC* tree, the signal at this node will no longer exhibit the effects of inductance. It is demonstrated that a single line analysis to determine the importance of including inductance to characterize an interconnect line that is a part of a tree is invalid in many cases and can lead to erroneous conclusions. The error exhibited by a single line analysis is due to the large interaction among the various branches within the tree.

This chapter is organized as follows. The effective damping factor of a signal at a specific node of a tree and the rise time of the input signal are used to derive two figures of merit that describe the relative importance of inductance for the signal at this node. These figures of merit are presented in section 8.1. In section 8.2, examples of *RLC* trees are used to illustrate the error encountered in treating a branch of a tree as a single line. Finally, some conclusions are offered in section 8.3.

8.1 Effect of Damping Factor and Input Rise Time

A second order approximation of the signals in an *RLC* tree (see Chapter 7) is used in this section to determine if the signal at a certain node exhibits significant inductance effects. The comparison scheme used in this section applies the same input to the second order approximation and to Wyatt's approximation. Two expressions result in the *s* domain representing *RC* and *RLC* models of a tree structured interconnect and are

$$S_{RCi}(s) = \frac{1}{\left[\sum_k C_k R_{ik}\right] s + 1} V_{in}(s). \tag{8.1}$$

$$S_{RLCi}(s) = \frac{1}{\left[\sum_k C_k L_{ik}\right] s^2 + \left[\sum_k C_k R_{ik}\right] s + 1} V_{in}(s). \tag{8.2}$$

respectively. The time domain equivalent of these two expressions are compared to determine the conditions at which the two time domain signals exhibit negligible error with respect to each other. These conditions represent criteria for neglecting inductance and using an *RC* model without suffering significant errors.

In subsection 8.1.1, the effective damping factor ζ_i at node *i* of an *RLC* tree is used to characterize when an *RC* model is sufficiently accurate as compared to an *RLC* model, permitting inductance to be neglected. It is shown that as ζ_i increases (or as the equivalent *RC* time constant at node *i* increases as compared to the equivalent *LC* time constant), inductance effects decrease. In subsection 8.1.2, the effect of the input rise time on the importance of inductance is discussed. It is shown that as the input rise time increases as compared to the equivalent *LC* time constant at node *i*, the effect of inductance on the transient behavior of the signal at node *i* becomes less significant.

8.1.1 Damping Factor

A step signal is used as the input to the second order approximation of the transfer function at node *i* of an *RLC* tree to investigate the relationship between the effective damping factor ζ_i and the significance of inductance on the transient behavior of the signal at node *i*. A step input is used since it eliminates the effect of

the rise time and maximizes the significance of the inductance, permitting the effect of the damping factor to be investigated. For a step input and a supply voltage of V_{DD} volts, the signal at node i is

$$S_i(t) = V_{DD} + V_{DD}\left[\frac{\exp[\omega_{ni}t(-\zeta_i + \sqrt{\zeta_i^2 - 1})]}{-\zeta_i + \sqrt{\zeta_i^2 - 1}} - \frac{\exp[\omega_{ni}t(-\zeta_i - \sqrt{\zeta_i^2 - 1})]}{-\zeta_i - \sqrt{\zeta_i^2 - 1}}\right]. \quad (8.3)$$

As the damping factor increases, the importance of the inductance on the circuit decreases. Thus, the following approximation can be made assuming large ζ_i,

$$\sqrt{\zeta_i^2 - 1} = \zeta_i[1 - \frac{1}{2\zeta_i^2}], \quad \text{with a relative error} < \frac{1}{4\zeta_i^4}. \quad (8.4)$$

With $\zeta_i > 2.5$, the error due to this approximation is less than 0.7%. With this approximation, the signal at node i can be approximated by

$$S_i(t) = V_{DD} + \frac{V_{DD}}{2[\zeta_i - \frac{1}{2\zeta_i}]}\left[-2\zeta_i \exp[\omega_{ni}t(-1/2\zeta_i)] + \frac{\exp[\omega_{ni}t(-2\zeta_i - 1/2\zeta_i)]}{2\zeta_i}\right]. \quad (8.5)$$

For $\zeta_i > 2.5$, this expression can be further approximated by

$$S_i(t) \cong V_{DD} - V_{DD}\exp[\omega_{ni}t(-1/2\zeta_i)] = V_{DD} - V_{DD}\exp[-t/\sum_k C_k R_{ik}], \quad (8.6)$$

with an error less than 8%. The maximum error is realized when the signal initially switches since the exponential terms are still relatively large. Note that (8.6) is precisely Wyatt's approximation for a step response at node i of an *RC* tree [74]. This relation shows that for $\zeta_i > 2.5$, the inductance has a minimal effect on the transient response at node i which is similar to the response of an equivalent *RC* tree where inductance is neglected. Thus, the first figure of merit presented in this Chapter is

$$\zeta_i = \frac{1}{2}\frac{\sum_k C_k R_{ik}}{\sqrt{\sum_k C_k L_{ik}}} > 2.5. \tag{8.7}$$

If this inequality is satisfied, the effects of inductance at node *i* are negligible. A plot of AS/X [128] simulations for the *RLC* tree shown in Figure 7.3 at output node 7 as compared to an equivalent tree with all inductances equal to zero is shown in Figure 8.1 for several values of ζ_i. The closed form solution in (8.3) is also shown. Note that for $\zeta_i > 2.5$, the response of the *RLC* tree is almost identical to that of an equivalent *RC* tree in which inductance is neglected.

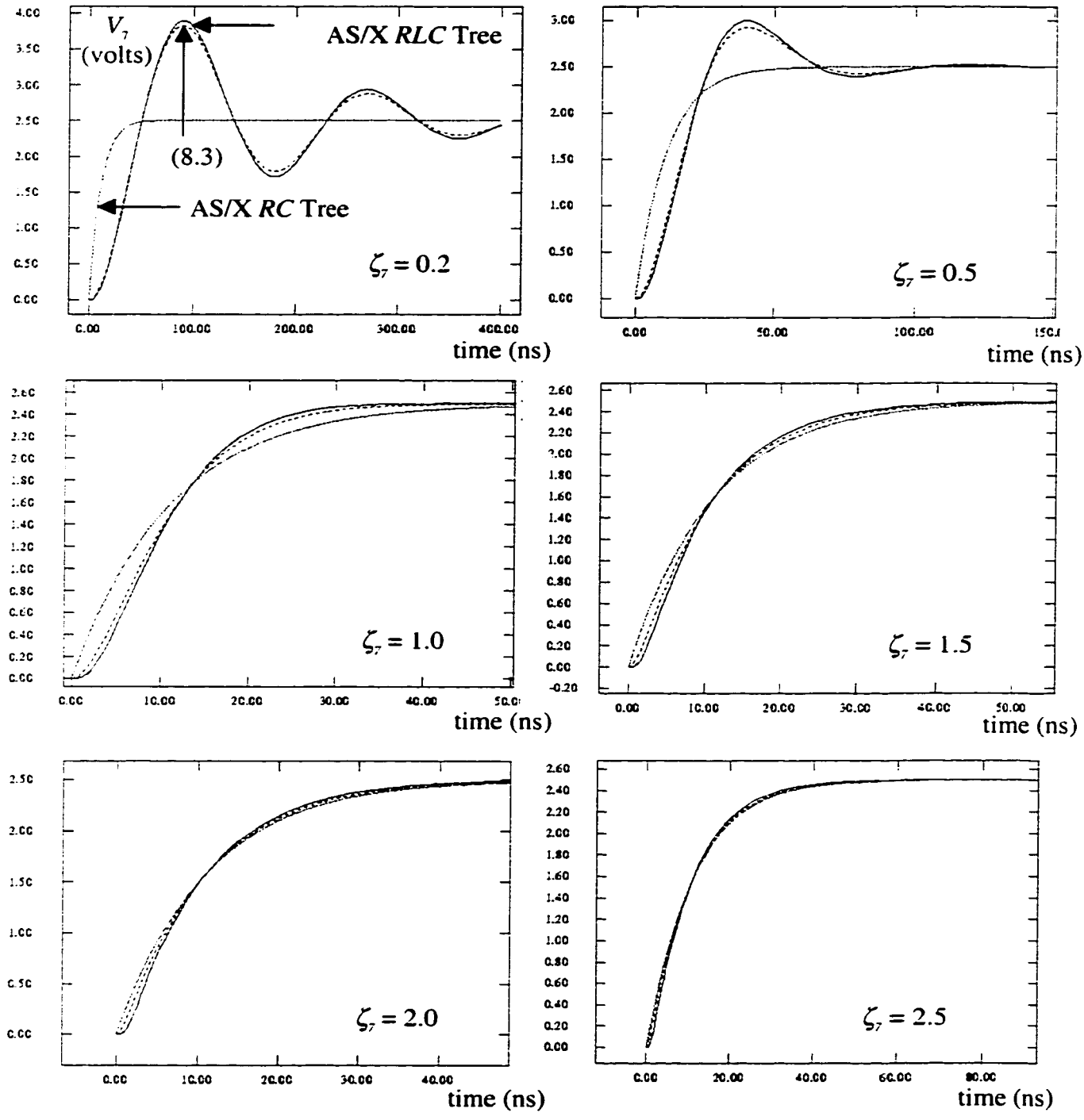


Figure 8.1. Effect of the equivalent damping factor on the accuracy of the *RLC* and *RC* models

8.1.2 Input Rise Time

An exponential signal of the form,

$$V_{in}(t) = V_{DD}[1 - \exp(-t/\tau)]u(t), \tag{8.8}$$

is used as the input to the second order approximation of the transfer function of an *RLC* tree to investigate the relationship between the input rise time and the effects of inductance on the transient behavior of the signal at node *i*. $u(t)$ is the unit step function, V_{DD} is the supply voltage, and the 90% rise time of the input signal is 2.3τ where τ is the time constant of the exponential in (8.8). With this input signal, the response at node *i* of an *RLC* tree is

$$e_{i_{RLC}}(t) = V_{DD}\left[1 - ke^{-\frac{t}{\tau}} + \frac{e^{-\zeta_i \omega_{ni} t}}{\sqrt{1-\zeta_i^2}}\left[\sin(\omega_{ni}t - \theta_1) - \sqrt{\frac{1}{k}}\sin(\omega_{ni}t - \theta_2)\right]\right], \tag{8.9}$$

where

$$\theta_1 = \tan^{-1}\left[\frac{\sqrt{1-\zeta_i^2}}{\zeta_i}\right], \tag{8.10}$$

$$\theta_2 = \tan^{-1}\left[\frac{\left(\tau/T_{LCi}\right)\sqrt{1-\zeta_i^2}}{\left(\tau/T_{LCi}\right)\zeta_i - 1}\right], \tag{8.11}$$

and

$$k = \frac{\left(\tau/T_{LCi}\right)^2}{\left(\tau/T_{LCi}\right)^2 - 2\zeta\left(\tau/T_{LCi}\right) + 1}. \tag{8.12}$$

T_{LCi} is

$$T_{LCi} = \sqrt{\sum_k C_k L_{ik}} \ . \tag{8.13}$$

According to Wyatt's approximation [74], if the same input is applied to an *RC* tree, the response at node *i* is

$$e_{t_{RC}}(t) = V_{DD}\left[1 - k_2 e^{-\frac{t}{\tau}} + e^{-t/T_{RCi}}[k_2 - 1]\right], \tag{8.14}$$

where

$$k_2 = \frac{\left(\tau / T_{RCi}\right)}{\left(\tau / T_{RCi}\right) - 1}, \tag{8.15}$$

$$T_{RCi} = \sum_k C_k R_{ik} \ . \tag{8.16}$$

When the rise time of the input signal increases, (8.9) approaches (8.14). This trend can be better understood by noting that if τ / T_{LCi} and τ / T_{RCi} are both much greater than one, k and k_2 tend to one and θ_2 tends to θ_1. Thus, if τ / T_{LCi} and τ / T_{RCi} are much greater than one, the response at node *i* of an *RLC* tree does not exhibit any effects caused by inductance and an *RC* tree model can be used to model the interconnect tree. These two conditions, τ / T_{LCi} and τ / T_{RCi}, are much greater than one and reduce to the first condition if the damping factor figure of merit described by (8.7) is considered. If ζ_i is greater than 2.5, the inductance effects are not significant because of the damping factor and there is no need to determine the rise time of the input signal. If ζ_i is less than 2.5, then $T_{RCi} < 5T_{LCi}$. Thus, $\tau / T_{LCi} < 5\tau / T_{RCi}$ is the range where the input rise time should be evaluated ($\zeta_i < 2.5$). Hence, if τ / T_{LCi} is much greater than one and $\zeta_i < 2.5$, then τ / T_{RCi} is also much greater than one.

The second figure of merit can be derived by assuming $\tau / T_{LCi} = 10$ and using the relation $t_{rn} = 2.3\tau$. Thus, the second figure of merit is

$$t_{rn} > 23\sqrt{\sum_k C_k L_{ik}} \,. \tag{8.17}$$

If this inequality is satisfied, the effects of inductance at node i can be neglected. A plot of AS/X [128] simulations of the *RLC* tree shown in Figure 7.3 at output node 7 as compared to an equivalent tree with no inductances is shown in Figure 8.2 for several values of t_{rn}. The closed form solution (8.9) is also shown. ζ_i is kept constant at 0.5 so that the inductance can not be ignored. Note that for $t_{rn} / T_{LCi} > 23$, the response of the *RLC* tree is the same as that of an equivalent *RC* tree in which inductance is neglected.

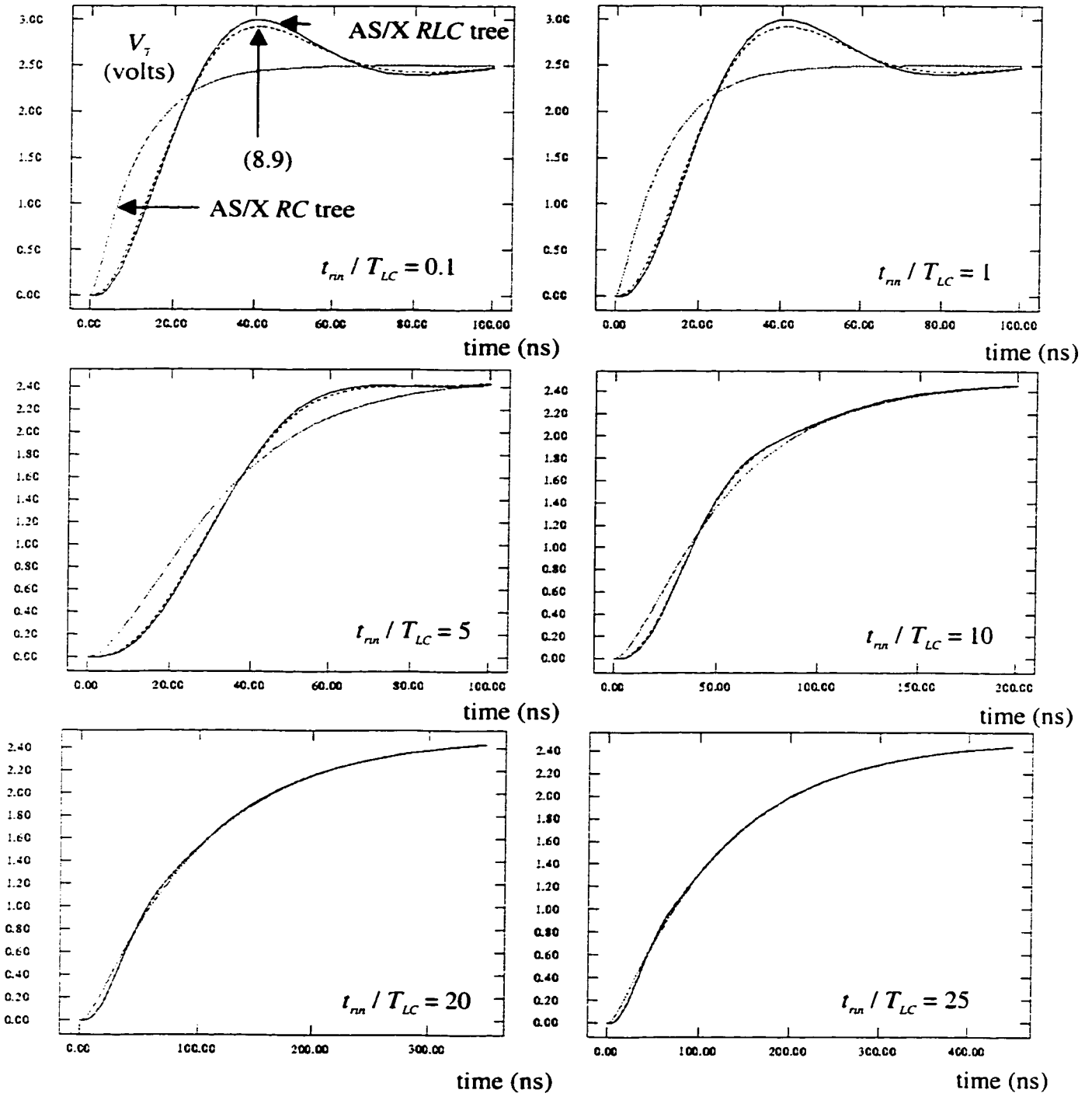


Figure 8.2. Effect of the rise time on the inductance effects in an *RLC* tree. t_{rn} / T_{LC} is varied from 0.1 to 25. AS/X simulations are shown for an *RC* tree and an *RLC* tree. (8.9) is also shown to illustrate the accuracy of the closed form solution introduced here. Note that as t_{rn} / T_{LC} increases, the *RC* model approaches the *RLC* model.

8.2 Results and Examples

Examples illustrating the importance of using a tree analysis for characterizing inductance effects as well as general traits of inductance effects in *RLC* trees are presented in this section. In subsection 8.2.1, a single line analysis to characterize the importance of inductance is compared to a tree analysis and an example is given that demonstrates that a single line analysis can lead to erroneous conclusions. The effect of the size of a tree on the significance of inductance is discussed in subsection 8.2.2. It is shown that there is a range of tree size for which inductance effects are prominent.

8.2.1 Tree Analysis Versus a Single Line Analysis

The analysis of single lines to characterize the importance of on-chip inductance has been previously evaluated [49]-[51] and [68]. However, analyzing single lines to characterize the importance of inductance in *RLC* trees can be invalid. To illustrate this point, values for the branch resistances, inductances, and capacitors for the *RLC* tree shown in Figure 7.3 are listed in Table 8.1. According to [49]-[51] and [68], if a single line analysis is used for each branch, the damping factor for branch i is

$$\zeta_i = \frac{1}{2}\frac{R_i C_i}{\sqrt{L_i C_i}}. \tag{8.18}$$

The damping factor of branch i affects the signal at node i. The single line analysis and the *RLC* tree analysis introduced here are compared in Table 8.2 for the tree shown in Figure 7.3. The branch impedance values listed in Table 8.1 are used. Note

the large difference in the values of the damping factors according to an *RLC* single line analysis as compared to an *RLC* tree analysis. For example, at node 7, the *RLC* single line analysis anticipates no significant inductance effects ($\zeta_7 = 1.58$) while an *RLC* tree analysis anticipates large inductance effects ($\zeta_7 = 0.529$). Simulations of the voltage signal at node 7 of the *RLC* tree shown in Figure 7.3 with the branch impedance values listed in Table 8.1 are shown in Figure 8.3. The voltage at node 7 exhibits high inductive effects as anticipated by the *RLC* tree analysis introduced here. This simple example demonstrates that an *RLC* single line analysis can lead in certain cases to erroneous conclusions. Note also that for node 1, the *RLC* single line analysis anticipates greater inductance effects ($\zeta_1 = 0.176$) as compared to the *RLC* tree analysis ($\zeta_1 = 0.306$).

The *RLC* single line analysis generates a significant difference between the maximum and minimum damping factors ($0.176 < \zeta < 1.58$) as compared to the difference between the maximum and minimum damping factors in the more accurate *RLC* tree analysis ($0.306 < \zeta < 0.529$). This behavior is due to analyzing each line individually while in reality all the branches in the tree interact significantly, distributing the effects of the branch inductances throughout the tree. Alternatively, the branches with higher inductive effects and the branches with lower inductance effects influence each other, making the effect of inductance less on those branches with higher inductance effects and more on those branches with lower inductance effects. This phenomenon is accurately captured by the *RLC* tree analysis introduced in this Chapter.

Table 8.1. Branch impedances for the *RLC* tree shown in Figure 7.3

Branch	R (Ω)	L (nH)	C (pF)
1	25	10	2
2	50	10	1
3	50	10	1
4	100	0.5	0.5
5	100	0.5	0.5
6	100	0.5	0.5
7	100	0.5	0.5

Table 8.2. Damping factors for the nodes of both the *RLC* single lines and the *RLC* tree shown in Figure 7.3

Node	ζ_i (*RLC* single line analysis)	ζ_i (*RLC* tree analysis)
1	0.176	0.306
2	0.25	0.441
3	0.25	0.441
4	1.58	0.529
5	1.58	0.529
6	1.58	0.529
7	1.58	0.529

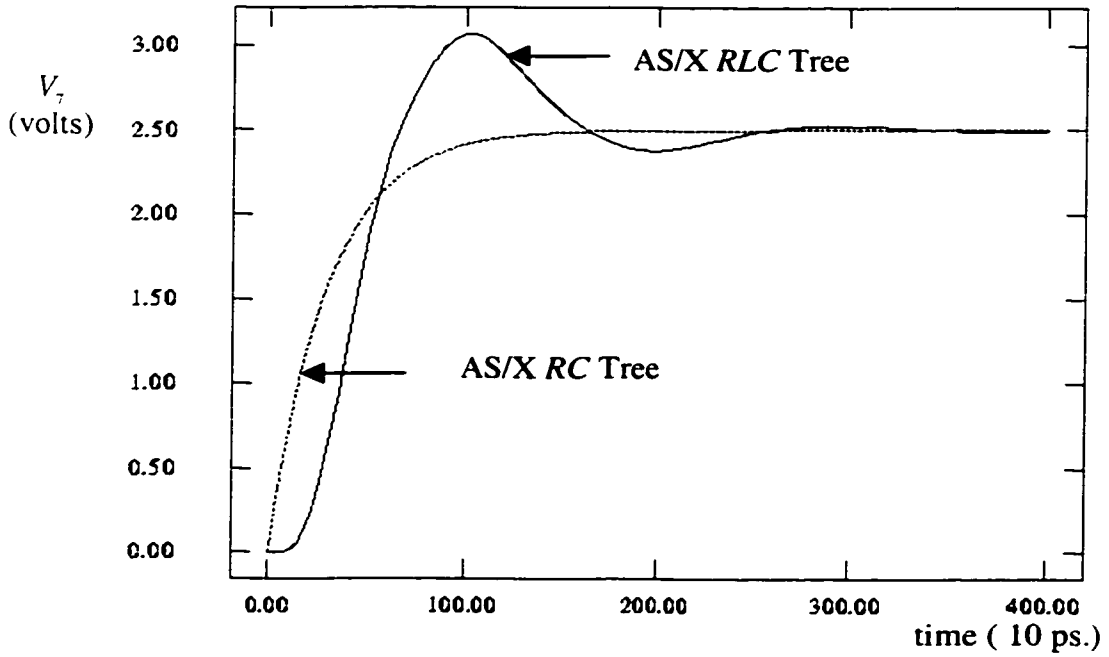


Figure 8.3. AS/X simulations of the output voltage at node 7 of the *RLC* tree shown in Figure 7.3 with the branch impedance values listed in Table 8.1 for the equivalent *RC* tree

8.2.2 Effect of Tree Size on the Significance of Inductance

The effect of increasing the size of the tree is to increase the damping factors at the nodes of the tree (and thus decrease the importance of the inductance). If the size of a tree increases, both of the summations $\sum_k C_k R_{ik}$ and $\sum_k C_k L_{ik}$ increase. As described by (7.18), ζ_i is half the first summation over the square root of the second summation. Thus, if the two summations increase at the same rate while increasing the size of the tree, the net result is an increase in ζ_i. For example, the damping factor at node 1 for the *RLC* tree shown in Figure 7.3 is

$$\zeta_1 = \frac{1}{2}\frac{R_1 C_T}{\sqrt{L_1 C_T}} = \frac{R_1}{2}\sqrt{\frac{C_T}{L_1}}, \tag{8.19}$$

where C_T is the total capacitance of the tree. If the size of the tree increases, C_T increases which increases the damping factor at node 1.

An important example of an *RLC* tree is a tree structured clock distribution network. A clock distribution network is often structured as a balanced tree with a wide trunk and narrowing branches [60]-[65]. If a tree has a branching factor of two (where each line is the parent of two other lines), for impedance matching purposes the parent will have double the width of its children [58], [59]. The size of the tree can be characterized by the number of levels n. A tree that has n levels has 2^n-1 branches. For example, the tree shown in Figure 7.3 has three levels and seven branches. The impedance of the branches in each level (r = 1, 2, ..., n) can be approximated by $2^{r-1}R_{root}$, L_{root}, and C_{root} / 2^{r-1}, where R_{root}, L_{root}, and C_{root} are the root resistance, inductance, and capacitance, respectively. The number of branches in level r is 2^{r-1}. Note that the inductance is assumed constant since it is a slowly varying

function with the width of the interconnect [48], [51]. The damping factor at an output node can be calculated as a function of the number of levels (representing the size of the tree) and the root impedance and is

$$\zeta_{out} = \frac{1}{4\sqrt{2}} \frac{R_{root} C_{root}}{\sqrt{L_{root} C_{root}}} \frac{n(n+1)}{\sqrt{2^{-n} + (n-1)}}. \tag{8.20}$$

Note that the output damping factor increases monotonically as n increases. For large n, ζ_{out} increases as $n^{1.5}$. A plot of ζ_{out} versus n is shown in Figure 8.4.

Alternatively, if the size of the tree is smaller, the rise time of the input signal can be much greater than T_{LCi} which, according to the second figure of merit in (8.17), eliminates the effects of inductance. Thus, there is a range of the size of an *RLC* tree where inductance effects are significant. For the special case of a single *RLC* line the size is simply represented by the length of the line which is consistent with the results described in [68], in which there is a range of interconnect line length where inductance effects are significant.

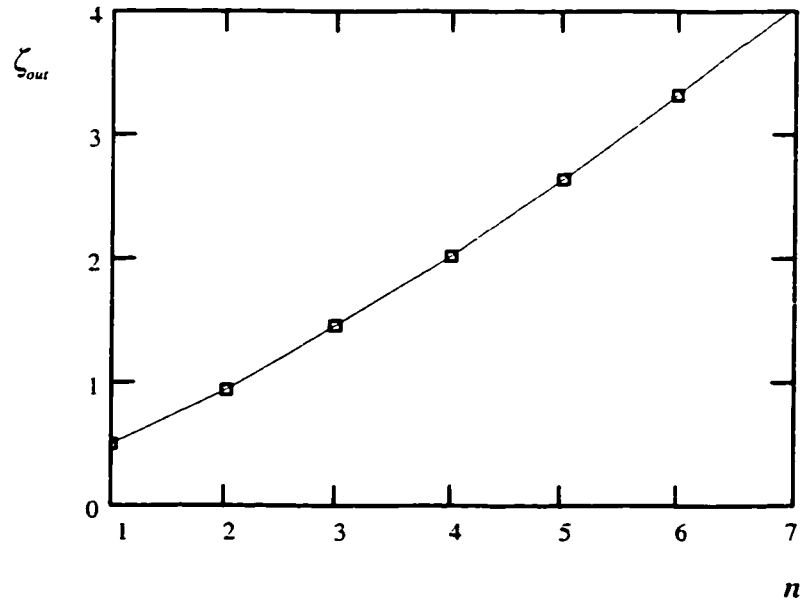


Figure 8.4. Effect of the number of levels n on the output damping factor ζ_{out} of a binary clock tree.

8.3 Conclusions

A second order approximation of an *RLC* tree with the same accuracy characteristics as the Wyatt approximation for an *RC* tree has been introduced. This second order approximation is used to derive two simple figures of merit to evaluate the significance of the inductance effects exhibited by an *RLC* tree. The first figure of merit is the damping factor of a signal at a specific node of a tree. It is shown that as the damping factor increases, inductance effects decrease. The second figure of merit is the rise time of the input signal as compared to the effective *LC* time constant of the tree at a specific node. It is also shown that as the input rise time increases as compared to the effective *LC* time constant, the importance of inductance decreases. Evidence is provided that using a single *RLC* line analysis for those branches within a tree can lead to incorrect conclusions. The error exhibited by a single line analysis is due to the large interaction among the branches of a tree. Finally, it is shown that there is a range of the size of an *RLC* tree where a tree can exhibit significant inductive effects.

Chapter 9 Repeater Insertion in Tree Structured Inductive Interconnect

As discussed in Chapter 6, repeater insertion has become an increasingly common design methodology for driving long resistive interconnect [37]-[44]. Closed form solutions for repeater insertion in *RLC* lines have been presented in Chapter 6. Most of the interconnects in a VLSI circuit are typically tree structured. To verify the theoretical results described in Chapter 6, an algorithm is introduced to insert and size repeaters within an *RLC* tree to optimize a variety of possible cost functions. Examples of cost functions are minimizing the maximum path delay, the skew between branches, or a combination of area, power, and delay. The algorithm has a complexity proportional to the square of the number of possible repeater positions and determines a repeater solution that is close to the global minimum. The repeater insertion algorithm is used to insert repeaters within a variety of 0.25 μm CMOS copper-based interconnect trees with the objective of minimizing the maximum path delay based on both an *RC* model and an *RLC* model. The two repeater solutions are compared using AS/X. It is shown that as inductance effects increase, the area and power consumed by the inserted repeaters to minimize the path delays of an *RLC* tree decreases. These results are in agreement with the theoretical results presented in Chapter 6. By including inductance in the repeater insertion methodology, the interconnect is modeled more accurately as compared to an *RC* model, permitting average savings in area, power, and delay of 40.8%, 15.6%, and 6.7%, respectively,

for the set of trees analyzed in this chapter. The average savings in area, power, and delay increases to 62.2%, 57.2%, and 9.4%, respectively, when using five times faster devices with the same interconnect trees.

The focus of this chapter is two fold: to describe a CAD system for repeater insertion in *RLC* trees in order to optimize a variety of cost functions and to characterize the effects of neglecting inductance on the repeater insertion process. Results from applying the repeater insertion tool to several industrial trees are also interpreted. The chapter is organized as follows. In section 9.1, the basic repeater insertion algorithm which can be used with any delay model for the interconnect and transistor devices is described. The specific delay models used in this chapter for the transistors and the interconnect are described in section 9.2. The results of applying the tool to insert repeaters in several practical copper-based interconnect trees are presented in section 9.3. Finally, a summary is given in section 9.4.

9.1 Algorithm for Repeater Insertion in *RLC* Trees

A generic algorithm to insert repeaters in a general *RLC* tree is presented in this section. The algorithm can be used with different delay models such as the Elmore delay, moment matching methods, and/or the effective capacitance model to evaluate the transient response of the buffered *RLC* tree. The algorithm has a quadratic complexity with the number of possible repeater positions in an *RLC* tree and achieves a repeater solution that is reasonably close to the global optimum repeater solution. In subsection 9.1.1, the repeater insertion problem is defined. The algorithm for repeater insertion used in this chapter is discussed in subsection 9.1.2. The complexity and optimality of the algorithm are discussed in subsection 9.1.3.

9.1.1 Problem Definition

The problem of inserting repeater in an *RLC* tree to minimize a given cost function is formulated and defined in this subsection. The terms and mathematical notations used in this chapter are also defined. An arbitrary tree is shown in Figure 9.1. The tree has n wires with the input source driving the root wire. Each wire w drives two wires, a left wire $left(w)$ and a right wire $right(w)$. If a left (right) wire does not exist then $left(w) = 0$ ($right(w) = 0$). A leaf is a wire that has $left(w) = 0$ and $right(w) = 0$. The tree has r leaf wires, each of which drives one of the sinks of the tree. A binary branching factor is used without loss of generality since any tree can be transformed into a binary tree by inserting zero impedance wires [37], [43]. At each sink $1 \leq i \leq r$, the propagation delay t_{di} is defined as the 50% delay of the output signal at sink i with respect to the input signal at the root of the tree. Within a tree, there are m pre-specified repeater positions where repeaters can be inserted to minimize a given cost function. The possible repeater positions are represented by the circles shown in Figure 9.1 and are placed at the beginning of each wire to allow for maximum capacitive decoupling of the critical paths [37], [43]. Each wire can be subdivided into several shorter wires to permit repeater insertion within long wires [43]. In some cases, no possible repeater positions can be assigned to some wires due to layout constraints. Those wires are labeled to indicate that no repeaters can be inserted along the wires.

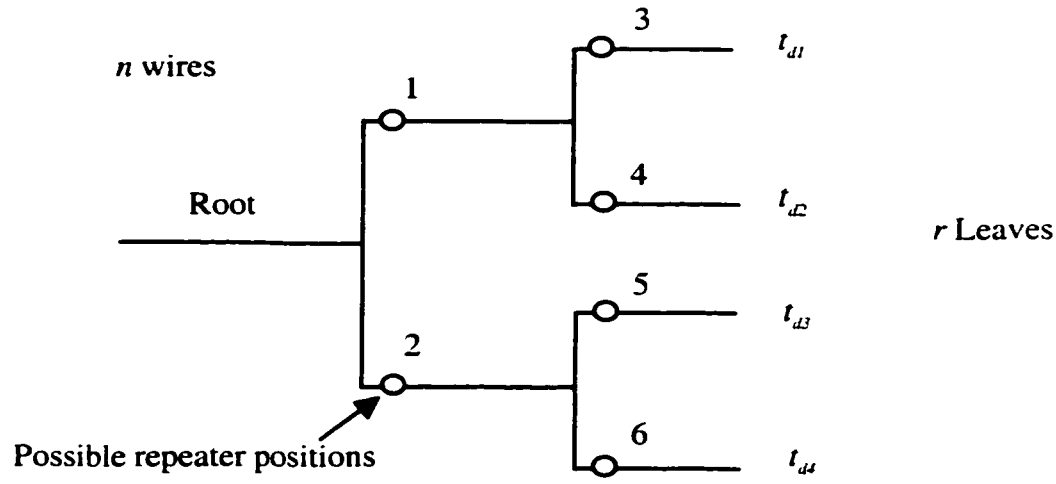


Figure 9.1. An arbitrary tree with n wires. The possible repeater positions are represented by circles.

The repeater insertion problem can be defined as: determine the set of repeater sizes h_j, $1 \le j \le m$, that minimizes a given cost function $C(h_1, h_2, \ldots, h_j, \ldots, h_m)$. The repeaters are considered to be symmetric inverters with widths h_j and a minimum sized channel length. The repeater sizes h_j are continuous numbers. The special repeater size $h_j = 0$ indicates that no repeater is inserted at node j. The sizes of the repeaters are to be found in the range $1 \le h_j \le h_{max}$ where h_{max} is the maximum allowable size of any repeater. A variety of cost functions can be used. Examples are: minimize($\max_i t_{di}$) which aims to minimize the maximum path delay, minimize($\max_{i,k}(t_{di}-t_{dk})$) where $1 \le i,k \le r$ which is equivalent to minimizing the skew between branches i and k, minimize(t_{dk}) where k is a critical output, or minimize ($f(t_{di})$ + $\sum_{j=1}^{m} h_j$) which considers the area of the repeaters. Other cost functions can include power and slew rate.

9.1.2 Repeater Insertion Algorithm

According to the problem definition described in the previous subsection, the sizes h_j that minimize the cost function $C(h_1, h_2, \ldots, h_j, \ldots, h_m)$ need to be calculated. The algorithm to calculate the optimum sizes of the repeaters to minimize the cost function is provided in Figure 9.2. Referring to Figure 9.1, the algorithm starts with the initial condition $h_j = 0 \ \forall \ j$ which corresponds to an unbuffered tree. The cost function $C(h_1, h_2, \ldots, h_j, \ldots, h_m)$ is evaluated for several sizes of the repeater at node 1, h_1, with all other repeater sizes $h_2, \ldots, h_m$ equal to zero (no repeaters). A binary search is applied which permits the value of h_1 that minimizes the cost function to be reached within a few steps where each step involves choosing a new value for h_1 and evaluating the cost function. The number of steps depends on h_{max} and is typically less than ten steps. If the case of no repeater at node 1 ($h_1 = 0$) provides the lowest cost, h_1 remains equal to zero. Thus, the algorithm can only improve the cost function at each step. Next, the size of the repeater at node 2, h_2, that minimizes the cost is determined in the same manner with h_1 set to the value calculated from the previous step and all other repeater sizes set to zero. The process is repeated for all m possible repeater positions. At each possible repeater position the size that minimizes the cost function is determined while all of the previous optimum repeater sizes remain constant. The process of covering all possible m repeater positions is defined as an iteration. Since in each step (determining the best repeater at node j) of an iteration the algorithm improves the cost function, the repeater solution at the end of an iteration generates a lower cost than at the beginning of an iteration. After the first iteration is completed, a second iteration starts by changing the sizes of the repeaters at the possible repeater

positions to determine the repeater sizes h_1, h_2, ..., h_m that minimize the cost function. However, in the second iteration, the initial repeater solution is the output of the previous iteration. Thus, at the second iteration (as compared to the first iteration), the capacitive load and driving resistance at the node at which the best repeater size is sought are closer to the values for minimum cost, enabling the optimum repeater sizes to be more accurately calculated. The iterations are repeated until there is no change in the size of any repeater as compared to the previous iteration. The algorithm typically converges within two or three iterations.

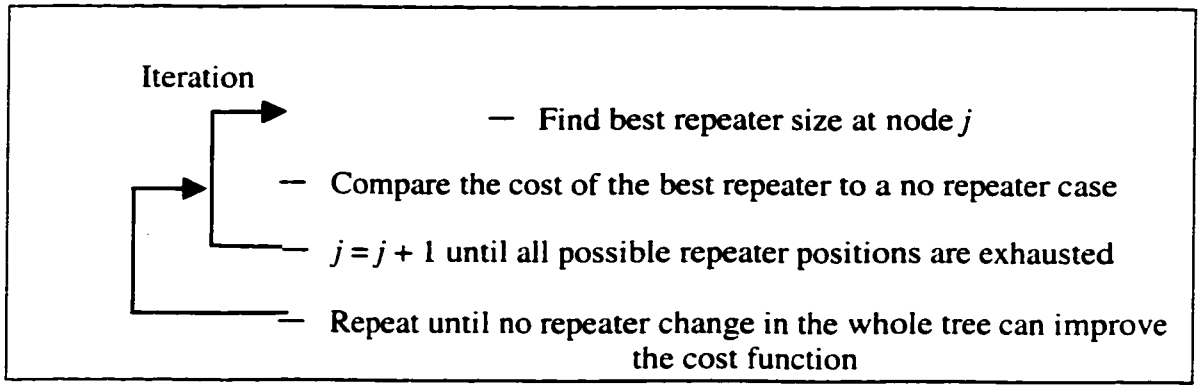


Figure 9.2. Proposed algorithm for inserting repeaters in an *RLC* tree.

9.1.3 Complexity and Optimality of Proposed Algorithm

The algorithm consists of several iterations. Each iteration scans the m possible repeater positions to determine the repeater sizes that minimize the objective cost. The number of necessary steps to find the repeater size at a possible repeater position which minimizes the cost is denoted B and is on average ten for the typical range of allowable repeater size ($1 \leq h_j \leq h_{max}$). The cost function is evaluated each

time the repeater size is changed at each of the B step. Thus, the complexity of an iteration is

$$\Theta(iteration) = O(m \cdot B) \cdot O(evaluating\ the\ cost\ function). \tag{9.1}$$

The complexity of evaluating the cost function depends upon the delay model used for the drivers and the interconnect. As shown in section 9.2, for the specific delay model used here, the cost function can be evaluated in a time proportional to the number of wires in the tree, n. Thus, the complexity of a single iteration is

$$\Theta(iteration) = O(m \cdot n \cdot B). \tag{9.2}$$

As mentioned previously, the number of iterations for convergence is typically two or three. The memory requirement of the algorithm is proportional to the number of wires, n.

The algorithm terminates when no change in the size of a single repeater can improve the cost function. This can be expressed mathematically as

$$\frac{dC(h_1, h_2, ..., h_j, ..., h_m)}{dh_j} = 0 \qquad \forall\ j. \tag{9.3}$$

This relation means that the algorithm reaches a minimum in the cost function. There is no guarantee, however, that this minimum is the global minimum. To improve the final repeater solution, the two repeaters at the left and right possible repeater positions of each wire are simultaneously changed. The process of determining two repeater sizes that minimize the cost simultaneously requires B^2 steps with the binary search algorithm used here. Since there are m / 2 possible repeater position pairs, the complexity of this modified algorithm is

$$\Theta(2^{nd}\ order\ alg) = O(m \cdot n \cdot \frac{B^2}{2}). \tag{9.4}$$

This modified algorithm does not reach the first minimum near the initial point. Rather, the modified algorithm searches for a minimum closer to the global minimum. The price is increased processing time. In general, a set of higher order algorithms can be achieved by simultaneously changing more repeaters. The complexities of these algorithms are

$$\Theta(alg) = O(m \cdot n \cdot B),\ O(m \cdot n \cdot \frac{B^2}{2}),\ O(m \cdot n \cdot \frac{B^3}{3}),\cdots\cdots,\ O(n \cdot B^m). \qquad (9.5)$$

The algorithm that changes m repeaters simultaneously is guaranteed to reach the global minimum. However, the processing time is exponential with the number of possible repeater positions and is prohibitively high even for relatively small trees. This set of algorithms has been examined for small trees (seven to eight possible repeater positions) and compared to the exhaustive algorithm that changes all m repeaters simultaneously. The results demonstrate that the second order algorithm consistently reaches the global or a near global minimum. The higher order algorithms introduced no or only a slight improvement in the final repeater solution as compared to the second order algorithm. The CPU run time of the second order algorithm is 20 sec on an S/490 IBM machine with one gigabyte of RAM for a large tree with 250 possible repeater positions. For typical trees with less than fifty possible repeater positions, the CPU time is less than one second. Hence, the second order algorithm is used in the examples discussed in this chapter.

9.2 Delay Model

As mentioned in the previous section, the repeater insertion algorithm can be used with any delay model. The specific delay model used in this chapter is discussed in this section.

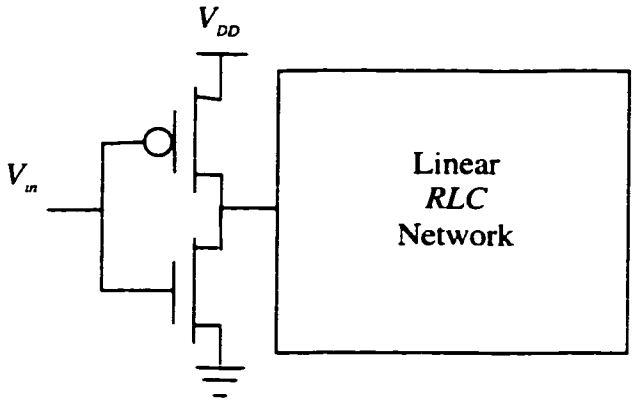


Figure 9.3. A symmetric CMOS inverter driving an *RLC* Network.

The problem of evaluating the delay at a sink of a buffered tree simplifies to adding the delay of several structures as shown in Figure 9.3 along the path from the input to the sink. The structure shown in Figure 9.3 is a symmetric inverter (repeater) driving an *RLC* tree (which is a sub-tree of the original *RLC* tree). Evaluating the delay of such a structure is complicated by a combination of linear and nonlinear elements constituting the circuit. It is common to replace the nonlinear transistors by equivalent linear resistors, *e.g.*, [23], [37], [39], [42], [43]. However, such an approximation strongly affects the final repeater solution, significantly increasing the final cost achieved by the repeater insertion algorithm. Thus, in this subsection, a method [109] is discussed that significantly improves the accuracy of the transistor

model as compared to a linear resistor approximation. The proposed method approximates the nonlinear transistor characteristic by a two piecewise linear curve as shown in Figure 9.4. Assuming a step input, the input signal is constant at the supply voltage V_{DD} for the entire switching time. Thus, the gate-to-source voltage of the NMOS transistor is V_{DD} and the PMOS transistor is off for the entire switching time. The curve shown in Figure 9.4 is the drain-to-source current I_{DS} versus the source-to-drain voltage of the NMOS transistor V_{DS} where V_{GS} is equal to V_{DD}.

The method used here calculates the delay of two linear networks, one assuming the transistor operates in the linear region for the entire switching time and the other assuming the transistor operates in the saturation region for the entire switching time. The two linear circuit models used for approximating the transistor in the linear and saturation regions are shown in Figure 9.5 (a) and (b), respectively. These linear and saturation transistor models are combined with the *RLC* tree driven by the repeater, resulting in two linear *RLC* networks. A delay value is determined for each *RLC* network using a linear network analysis method and are denoted t_{pdlin} and t_{pdsat} for the linear and saturation regions of operation, respectively. The parameters used to define the device model in the linear and saturation regions are C_{in}, R_{lin}, C_{out}, I_{sat}, and R_{sat}. I_{sat}, R_{sat}, and R_{lin}, respectively, and are shown in Figure 9.4. These parameters describe the saturation current of a transistor with V_{GS} equal to V_{DD} and the equivalent output resistance of a transistor in the saturation and linear regions, respectively. C_{in} and C_{out} are the input and output capacitances of the repeater. These parameters are calculated in terms of the corresponding parameters, C_{in0}, R_{lin0}, C_{out0}, I_{sat0}, and R_{sat0}, of a minimum size symmetric inverter. An inverter h times wider than a minimum size inverter has $C_{in}=C_{in0}h$, $R_{lin}=R_{lin0}/h$, $C_{out}=C_{out0}h$, $I_{sat}=I_{sat0}h$, and $R_{sat}=R_{sat0}/h$.

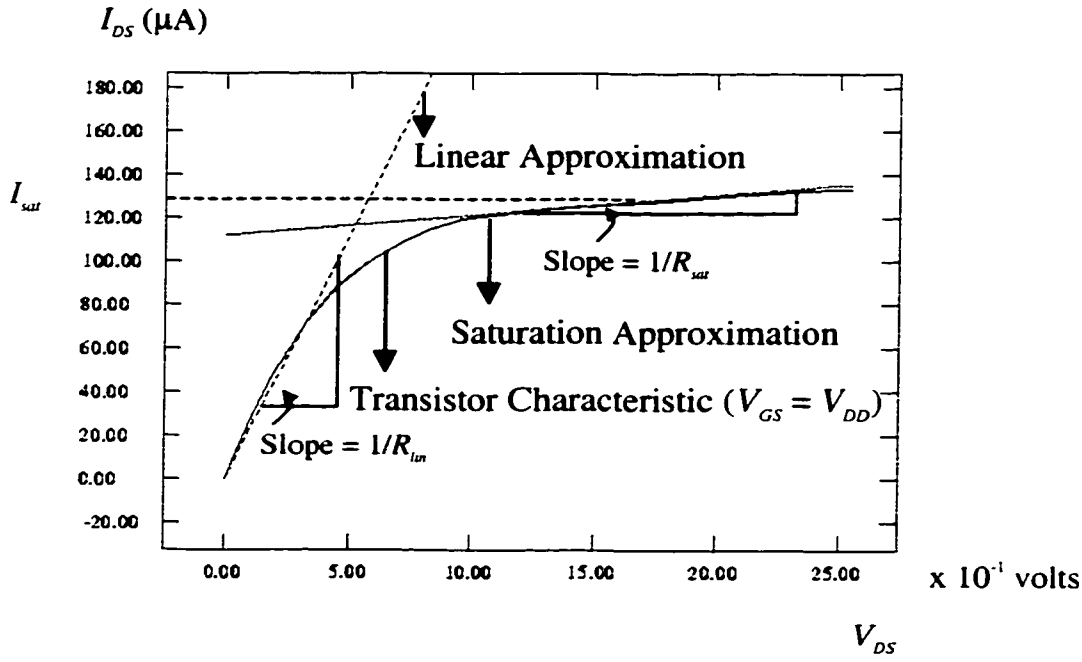


Figure 9.4. Piecewise linear approximation of an NMOS transistor for $V_{GS} = V_{DD}$.

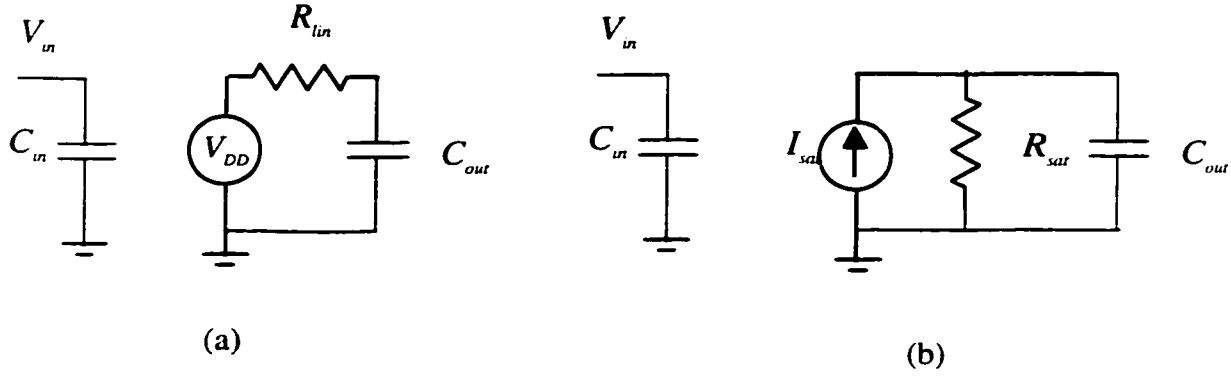


Figure 9.5. Equivalent circuit models of an NMOS transistor when operating (a) in the linear region and (b) in the saturation region for $V_{GS} = V_{DD}$.

The method presented here proceeds by observing the values of the two delays t_{pdsat} and t_{pdlin}, attempting to estimate a delay value that most closely approximates the delay of the network with nonlinear transistors based on t_{pdsat} and t_{pdlin}. The key

observation to achieve this goal is that if one of the delays is significantly larger than the other delay (say t_{pdsat} is significantly larger than t_{pdlin}), the larger delay is an accurate estimate of the delay of the nonlinear transistors. This trait can be explained by noting in Figure 9.4 that in region where the saturation approximation accurately describes the characteristic of the transistor, the linear region approximation predicts a higher source-to-drain current. Similarly, the saturation region approximation predicts a higher source-to-drain current where the linear region approximation is accurate. Thus, if the saturation delay t_{pdsat} is significantly larger than t_{pdlin}, the transistor has operated in the saturation region most of the time since a larger saturation delay indicates less current than predicted by the saturation region approximation. Alternatively, a significantly larger t_{pdlin} indicates that the transistor has operated primarily in the linear region. t_{pdlin} therefore accurately characterizes the delay of the nonlinear transistor. Inaccuracies arise when the magnitudes of t_{pdsat} and t_{pdlin} are relatively close, in which case neither t_{pdsat} nor t_{pdlin} can solely characterize the propagation delay of a nonlinear CMOS gate driving an *RLC* tree since the NMOS transistor operates partially in the saturation region and partially in the linear region. However, a combination of both t_{pdsat} and t_{pdlin} can be used to accurately characterize the total propagation delay [109].

Based on this discussion, the variable $\Delta = (t_{pdlin} - t_{pdsat})/t_{pdsat}$ is used as a criterion to determine how best to combine t_{pdsat} and t_{pdlin}. SPICE simulations of the propagation delay t_{pd} of a CMOS gate driving an *RLC* tree versus Δ are shown in Figure 9.6. Note that the parameters of the tree and the transistor sizes are varied such that t_{pdsat} remains constant. Thus, Δ changes linearly with t_{pdlin}. Referring to Figure 9.6, the delay is accurately characterized by t_{pdsat} for small Δ ($\Delta < -0.5$) and by t_{pdlin} for large Δ ($\Delta > 2$)

which agrees with the conclusions made above. Curve fitting is used to derive the function that best characterizes the delay as a function of Δ. This function is

$$t_{pd} = t_{pdsat}(1 + \Delta + e^{-1.1(1+\Delta)}). \tag{9.6}$$

Substituting for Δ, the resulting delay for the general case in terms of t_{pdsat} and t_{pdlin} is [109]

$$t_{pd} = t_{pdlin} + t_{pdsat} \exp(-1.1\frac{t_{pdlin}}{t_{pdsat}}). \tag{9.7}$$

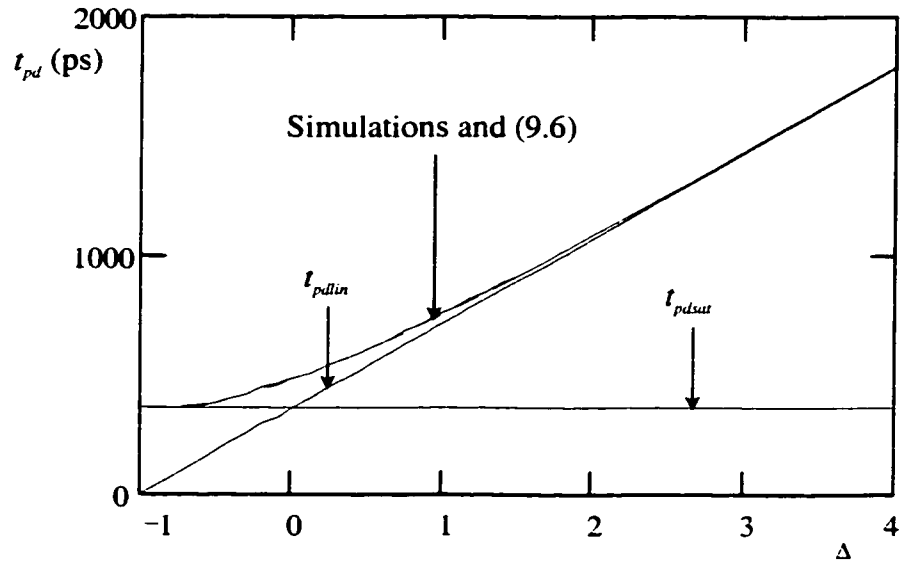


Figure 9.6. SPICE simulations of the propagation delay t_{pd} of a CMOS gate driving an *RLC* transmission line versus Δ. $C_L = 0$, $C_t = 1$ pF, $L_t = 10$ nH, and R_t is varied to change Δ. The interconnect is modeled as 32 *RLC* Π sections.

In general, this method is highly accurate (errors within 3%) for fast input signals. Additional error may result from the linear analysis method used to determine t_{pdsat} and t_{pdlin} of an *RLC* network.

9.3 Results and Discussion

The results of applying the CAD-based repeater insertion tool to several industrial copper-based interconnect trees are summarized and discussed in this section. The *RLC* trees described in this chapter are copper interconnect wires based on an IBM 0.25 μm CMOS technology. The depth of the trees (the maximum path length from the input to the sinks) is between 0.5 cm to 1.5 cm which represents a wide range of critical global signals typically encountered in VLSI circuits. Long wires within the trees are partitioned with a maximum segment length of 0.5 mm to permit repeaters to be inserted within these long wires in order to improved the performance of the circuit [43].

A repeater solution is determined to minimize the maximum path delay of each tree based on the *RLC* delay model discussed in the previous section. The total area of the repeaters inserted within each tree is described in terms of the area of a minimum size repeater. The tool also generates an AS/X [128] input file which is used to simulate the maximum path delay and the power consumption of the buffered *RLC* tree. The total inserted repeater area, the maximum path delay, and the power consumption of the buffered trees are depicted in Table 9.1. The tool is also used with AS/X to determine the total repeater area, the maximum path delay, and the power consumption of the buffered *RLC* trees when inductance is neglected and repeaters are inserted based on an *RC* model. The results based on the *RC* model are also listed in Table 9.1. Finally, AS/X simulations of the unbuffered *RLC* trees are used to determine the maximum path delay when repeater insertion is not employed. These results are listed in Table 9.1 as well.

Two important trends can be observed from the data listed in Table 9.1. The first trend is that inserting repeaters significantly reduces the maximum path delay as compared to the maximum path delay of an unbuffered tree. This behavior illustrates the importance of repeater insertion as an effective methodology to reduce interconnect delay. According to Table 9.2 and Table 9.3, the average saving in the maximum path delay when inserting repeaters based on an *RLC* model as compared to an unbuffered tree is about 40% where the maximum saving is 76% for TGL1 which is a large asymmetric tree. The second important trend apparent in the data listed in Table 9.1 is that inserting repeaters based on an *RLC* model as compared to an *RC* model consistently introduces savings in all of the three primary design criteria: area, power, and delay. This behavior demonstrates the importance of including inductance in a high speed repeater insertion methodology. According to Table 9.3, including inductance in the interconnect model saves an average 40.8% of the repeater area, 15.6% of the power dissipated by the buffered trees, and 6.7% of the maximum path delay as compared to using an *RC* model.

The reduced repeater area when including inductance in the interconnect model is due to the quadratic dependence of the delay on the length of an *RC* wire which tends to a linear dependence as inductance effects increase [110] (see Chapter 6). The 50% delay of an *RC* line is given by $0.35RCl^2$ [1], [23], [42] and by $l\sqrt{LC}$ [110] for an *LC* line when the line is driven by an ideal source with an open-circuit load. *R*, *L*, and *C* are the resistance, inductance, and capacitance per unit length of the line and *l* is the length of the line. These two cases of an *RC* line and an *LC* line are the limiting cases for inductance effects with the *RC* case representing no inductance effects and the *LC* case representing maximum inductance effects. In the *RC* case, the

square dependence on the interconnect length causes the delay to increase rapidly with wire length. It is therefore necessary to partition the line into multiple shorter sections by inserting repeaters, thereby reducing the total delay. However, for an *LC* line, the dependence is linear and no gain is achieved by breaking the line into shorter sections. Inserting repeaters in an *LC* line only degrades the delay due to the added gate delay. Thus, an *LC* line requires *zero* repeater area for minimum propagation delay.

In the general case of an *RLC* line, the repeater area for minimum propagation delay is between the maximum repeater area in the *RC* case and the zero repeater area in the *LC* case. The repeater area for minimum propagation delay of an *RLC* line decreases as inductance effects increase due to the sub-quadratic dependence of the propagation delay on the length of the interconnect [110]. Hence, inserting repeaters based on an *RC* model and neglecting inductance results in larger repeater area than necessary to achieve a minimum delay. The magnitude of the excess repeater area when using an *RC* model depends upon the relative magnitude of the inductance within the tree. For the specific copper-based interconnect *RLC* trees used here, almost half the repeater area can be saved by including inductance in the interconnect model. Note that a single line analysis can be used to interpret the behavior of a repeater insertion solution in a tree since in both cases repeaters are inserted to break the *RC* delay of long wires (paths and branches in the case of a tree).

Additionally, repeaters are inserted in a tree to decouple capacitance from the critical path. The effect of capacitance decoupling on improving the critical path delay is less significant when inductance effects increase. This trend is due to the *LC* time constant at node *i* of a tree ($\sqrt{\sum_k C_k L_{ik}}$) [113] (see Chapter 7), which has a square

root behavior as compared to the linear behavior of an *RC* time constant, $\sum_k C_k R_{ik}$.

Reducing the capacitance coupling has less effect on the *LC* time constant as compared to the *RC* time constant due to this square root behavior. As inductance effects increase, the square root behavior of the *LC* time constant dominates the behavior of the propagation delay. Thus, as inductance effects increase, the area of the inserted repeaters for capacitive decoupling also decreases.

A reduction in the power consumed by the buffered trees when including inductance in the interconnect model as compared to an *RC* model is a direct consequence of the reduced repeater area. The dynamic power consumption, which is linearly dependent on the total capacitance of the interconnect and the repeaters, decreases due to the reduced input and output capacitance of the repeaters. The short-circuit power consumption is significantly less for a smaller repeater since the short-circuit power consumed by a CMOS inverter is quadratically dependent on the width of the repeater [6]-[8]. The decreased delay achieved by including inductance is due to more accurate modeling of the interconnect; thereby enabling improved repeater insertion which eliminates the excess repeater area that would result when using an *RC* interconnect model. This excess repeater area increases the total delay due to the increased gate capacitance.

Another interesting aspect noted in Chapter 6 is that $T_{L/R}$ in (6.33) increases as the time constant R_0C_0 decreases, or alternatively, as faster repeaters are used. An increase in $T_{L/R}$ increases the discrepancy between an *RC* model and an *RLC* model as described by (6.36) and (6.37) even if the same interconnect trees are buffered to minimize the path delay. Thus, the analytical solutions in (6.33), (6.36), and (6.37) anticipate additional savings in repeater area by including inductance in the

interconnect model as compared to an *RC* model for technologies with faster devices. To verify this trend, five times faster devices than the 0.25 μm devices are used as repeaters to minimize the maximum path delays for the same set of trees listed in Table 9.1. The results corresponding to the data listed in Table 9.1 are listed in Table 9.4. Note that the savings in area, power, and delay increases when including inductance in the interconnect model rather than using an *RC* model with faster devices as compared to the 0.25 μm CMOS technology. The average savings increases from 40.8% to 62.2% for the repeater area, from 15.6% to 57.2% for the power consumption, and from 6.7% to 9.4% for the maximum path delay when using five times faster devices as compared to a 0.25 μm CMOS technology. Thus, with a faster technology, the penalty of ignoring inductance increases for all three primary design criteria: area, power, and delay. Therefore, with technology scaling, the issue of including inductance in the repeater insertion methodology will become of paramount importance.

This trend can be explained intuitively by examining the special case of a line with large inductance effects. As previously discussed, the minimum total propagation delay can be achieved for such a line by not inserting repeaters independent of the intrinsic speed of the technology. If inductance is ignored and an *RC* model is used for such a line, the number of repeaters that are inserted will increase as the repeaters become faster since there is less of a penalty for inserting more repeaters. Thus, the discrepancy between the repeater solutions based on an *RC* and an *RLC* model (zero repeater area for dominant inductance effects) increases as faster repeaters are used. In general, the area required by the repeaters to minimize the

total propagation delay based on an *RC* model as compared to an *RLC* model increases more rapidly as the devices become faster.

Table 9.1 Simulation results of unbuffered trees, buffered trees based on an *RLC* model, and buffered trees based on an *RC* model. The area, power, and maximum path delay are compared. The area is generated by the repeater insertion program while the power and maximum path delay are simulated using AS/X.

Tree Name	Area (minimum size inverters)			Power (pJ per Cycle)			Maximum Delay (ps)		
	Un-Buffered Tree	Buffered Tree *RLC* Model	Buffered Tree *RC* Model	Un-Buffered Tree	Buffered Tree *RLC* Model	Buffered Tree *RC* Model	Un-Buffered Tree	Buffered Tree *RLC* Model	Buffered Tree *RC* Model
TSs1	0	352	380	13.86	23.26	25	488	288	297
L1	0	102	250	8.15	11.19	13.76	342	267	272
TS2	0	0	659	25.67	25.67	37.90	193	193	193.5
L2	0	310	337	11.92	20.85	21.55	700	437	454
L3	0	0	422	22.8	22.8	30.3	213	213	237
TSm1	0	1246	1709	95	125	146	389	268	284
TSm2	0	1630	2751	135	211	221.5	343	278	296
TSL	0	1734	2471	147.5	196	227	431	292	304
TSL1	0	2999	4120	164	237	275	781	360	382
TGs1	0	649	842	38	51.2	57.8	262	231	256
TGs2	0	0	553	40.20	40.20	59.80	212	212	247
TGm1	0	1271	1854	89.1	120	139	460	306	344
TGL1	0	3823	7506	201	295	378	1740	442	495

Table 9.2 Percentage savings in area, power, and maximum path delay introduced by inserting repeaters based on an *RLC* model rather than an *RC* model. The percentage savings in delay when inserting repeaters as compared to an unbuffered tree are also listed.

Tree Name	Per cent savings in delay of a buffered tree based on an *RLC* model as compared to an unbuffered tree	Per cent savings in the area of repeaters inserted based on an *RLC* model as compared to using an *RC* model	Per cent savings in power dissipation when repeaters are inserted based on an *RLC* model as compared to using an *RC* model	Per cent savings in delay when repeaters are inserted based on an *RLC* model as compared to using an *RC* model
TSs1	40.9	7.3	6.9	3
L1	21.9	59.1	18.6	1.8
TS2	0	100	32.26	0.26
L2	37.6	8	3.2	3.7
L3	0	100	24.75	10.4
TSm1	31	27	14.3	5.9
TSm2	18.9	40.7	4.7	6.2
TSL	32.2	29.8	13.6	4
TSL1	51	27	13.8	5.9
TGs1	11.8	22.9	11.4	2.1
TGs2	0	100	32.77	14.5
TGm1	37.9	31.4	13.6	11.5
TGL1	76	49.5	21.9	11.9

Table 9.3 The total repeater area, total power, and total maximum path delay of all of the trees. The per cent savings shown here represent the average savings in area, power, and maximum path delay when using an *RLC* model for repeater insertion.

Totals					
	Un-Buffered	Savings in delay	Buffered *RLC* Model	Savings compared to *RC*	Buffered *RC* Model
Area (min inverters)	0	-	14116	40.8%	23854
Max delay (ps)	6554	42.2%	3787	6.7%	4061
Power (PJ/Cycle)	-	-	1379	15.6%	1632

Table 9.4 Simulation results of unbuffered trees, buffered trees based on an *RLC* model, and buffered trees based on an *RC* model with five times faster devices. The area, power, and maximum path delay are compared. The area is generated by the repeater insertion program while the power and maximum path delay are simulated using AS/X.

Tree Name	Area (minimum size inverters)			Power (pJ per Cycle)			Maximum Delay (ps)		
	Un-Buffered Tree	Buffered Tree *RLC* Model	Buffered Tree *RC* Model	Un-Buffered Tree	Buffered Tree *RLC* Model	Buffered Tree *RC* Model	Un-Buffered Tree	Buffered Tree *RLC* Model	Buffered Tree *RC* Model
TSs1	0	1349	1997	13.86	21.4	24.4	488	144	145
L1	0	569	1168	8.15	12.2	14.44	342	164	166
TS2	0	740	2738	25.67	34	62	193	154	165
L2	0	1137	1862	11.92	19.4	22.4	700	248	258
L3	0	534	1799	22.8	28	40	213	206	218
TSm1	0	5150	13468	95	177	348	389	222	240
TSm2	0	7107	21654	135	482	1516	343	238	262
TSL	0	12819	26674	147.5	382	832	431	220	240
TSL1	0	9358	35844	164	242	688	781	268	308
TGs1	0	2152	6392	38	60.8	115	262	198	224
TGs2	0	2402	4410	40.20	77.6	138.8	212	187	262
TGm1	0	5738	15184	89.1	141	302	460	212	232
TGL1	0	18905	37037	201	330	588	1740	346	378

Table 9.5 Percentage savings in area, power, and maximum path delay introduced by inserting repeaters based on an *RLC* model rather than an *RC* model. The devices used for the repeaters are from a five times faster technology as compared to the 0.25 μm CMOS technology used to generate the data listed in Table 9.2. The percentage savings in delay when inserting repeaters as compared to an unbuffered tree are also listed.

Tree Name	Per cent savings in delay of a buffered tree based on an *RLC* model as compared to an unbuffered tree	Per cent savings in the area of repeaters inserted based on an *RLC* model as compared to using an *RC* model	Per cent savings in power dissipation when repeaters are inserted based on an *RLC* model as compared to using an *RC* model	Per cent savings in delay when repeaters are inserted based on an *RLC* model as compared to using an *RC* model
TSs1	70.5	32.4	12.2	0.68
L1	52	51	15.5	1.2
TS2	20	72.9	45	6.6
L2	37.6	38.9	13.39	3.8
L3	3.2	70.31	28	5.5
TSm1	43	61.7	49.1	7.5
TSm2	30.6	67	68	9.2
TSL	49	52	54	8.3
TSL1	65.7	74	64.8	21.4
TGs1	24.4	66.3	47	11.6
TGs2	11.7	45.5	44.1	28.62
TGm1	53.9	62.2	53.3	8.6
TGL1	80	49	43	10.8

Table 9.6 The total repeater area, total power, and total maximum path delay of all of the trees using five times faster devices. The per cent savings shown here represent the average savings in area, power, and maximum path delay when using an *RLC* model for repeater insertion.

Totals					
	Un-Buffered	Savings in delay	Buffered *RLC* Model	Savings compared to *RC*	Buffered *RC* Model
Area (min inverters)	0	-	67960	62.2%	170227
Max delay (ps)	6554	57.17%	2807	9.4%	3098
Power (PJ/Cycle)	-	-	2007	57.2%	4691

9.4 Summary

The effect of inductance on inserting repeaters in *RLC* trees is investigated in this chapter. An algorithm is introduced to insert and size repeaters within an *RLC* tree to minimize a variety of possible cost functions. The algorithm has a polynomial complexity proportional to the square of the number of possible repeater positions and determines a repeater solution that is reasonably close to the global minimum. It is shown that as inductance effects increase, both the number of repeaters and the size of each repeater decrease. This trend means significantly less repeater area and power consumption due to decreased repeater capacitance. Also, less cost can be attained by including inductance in the design methodology rather than using an *RC* model since the interconnect is modeled more accurately. Hence, it is shown that including inductance in a repeater insertion design methodology as compared to using an *RC* model improves the overall repeater solution in terms of area, power, and delay. The average savings in area, power, and delay for the set of trees used in this Chapter are 40.8%, 15.6%, and 6.7%, respectively, when inserting repeaters based on an *RLC* delay model as compared to an *RC* delay model with repeaters from a 0.25 μm CMOS technology and copper interconnect. The average savings in area, power, and delay increases to 62.2%, 57.2%, and 9.4%, respectively, when using repeaters from a five times faster technology with the same set of interconnect trees.

Chapter 10 Dynamic and Short-Circuit Power of CMOS Gates Driving Lossless Transmission Lines

An *RC* model can be viewed as a limiting case of the *RLC* transmission line model where the inductance is considered to be negligible. This case has been thoroughly investigated and is well represented in the literature, *e.g.*, [21]-[44]. The other limiting case is an *LC* (or lossless) transmission line where the resistance is negligible. This case approximates the low loss lines encountered in Multi-Chip Modules (MCM) and Printed Circuit Boards (PCB). Although it is highly improbable that the resistance of an on-chip interconnect will become negligible within the near future, this *LC* analysis provides an upper limit for analyzing inductive effects in VLSI circuits. The behavior of an *RLC* transmission line case can therefore be bounded by analyzing the behavior of the *RC* and the *LC* cases.

The dynamic and short-circuit power consumption of a CMOS gate driving an *LC* transmission line as a limiting case of an *RLC* transmission line is investigated in this chapter. Closed form solutions for the output voltage and short-circuit power of a CMOS gate driving an *LC* transmission line are presented. These solutions agree with circuit simulations within 11% error for a wide range of transistor widths and line impedances for a 0.25 μm CMOS technology. The ratio of the short-circuit to dynamic power is shown to be less than 7% for CMOS gates driving *LC* transmission lines where the line is matched or underdriven. The total power consumption is

expected to decrease as inductance effects become more significant as compared to an *RC* dominated interconnect line.

This chapter is organized as follows. A closed form solution for the output voltage of a CMOS gate driving an *LC* transmission line is presented in section 10.1. This solution is based on the alpha power law for deep submicrometer (DSM) CMOS technologies [108]. This solution is compared to the output voltage of a CMOS gate driving a lumped capacitive representation of a line. It is also shown in section 10.1 that these two solutions become equivalent as the transition time of the signal at the input of the CMOS gate driving a transmission line becomes greater than twice the time of flight of the waves across the transmission line. The dynamic and short-circuit power consumption of a CMOS gate driving an *LC* transmission line is presented in section 10.2. The analysis in section 10.2 is performed for the case where the transition time of the signal at the input of the CMOS gate driving a transmission line is smaller than twice the time of flight of the waves propagating across the transmission line. Finally, some conclusions are offered in section 10.3.

10.1 Capacitive Approximation of a Lossless Transmission Line

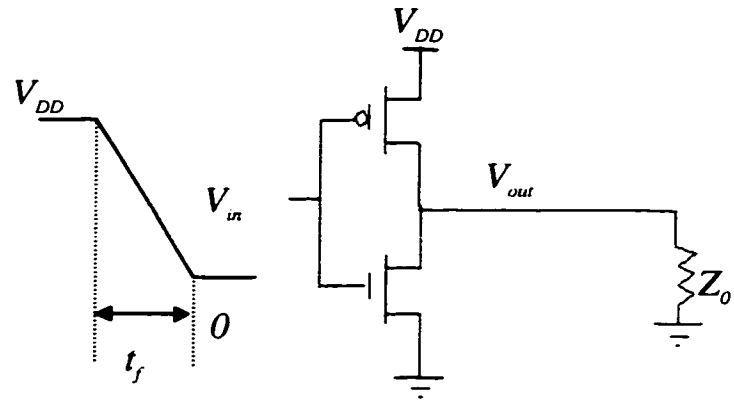


Figure 10.1. Equivalent circuit of a CMOS inverter driving a lossless transmission line for a period of time, $0<t<2T_0$

A transmission line can be replaced by its characteristic impedance Z_0 for the period of time $0 < t < 2T_0$, where T_0 is the time of flight of the signals across the line [70]-[72]. An equivalent circuit of a CMOS inverter driving a lossless transmission line with a characteristic impedance $Z_0 = \sqrt{L_t / C_t}$ is shown in Figure 10.1. For this equivalent circuit, the output voltage is given by the inverter output current multiplied by the characteristic impedance of the line. Ignoring the effect of the NMOS transistor for a falling input and assuming the PMOS transistor is saturated, the output current of the transistor does not depend on the output voltage (neglecting channel length modulation). Thus, the output voltage is

$$V_{out} = Z_0 I_{psat} = P(t), \tag{10.1}$$

for the period of time $0 < t < 2T_0$. I_{psat} is the saturation current of the PMOS transistor and is

$$I_{psat} = P_{Cp}\frac{W_p}{L_p}(VDD - V_{in} - |V_{Tp}|)^{\alpha_p}. \tag{10.2}$$

assuming the alpha power law is used to model the saturation current of a MOSFET [108], where P_C is a constant that characterizes the drive current of the transistor in saturation, W and L are the geometric width and length, respectively, of the transistor, and α is a constant between one (strong velocity saturation) and two (weak velocity saturation) [108]. V_{Tp} is the threshold voltage of the P-channel device and is negative for an enhancement mode device. p indicates the PMOS transistor and n indicates the NMOS transistor.

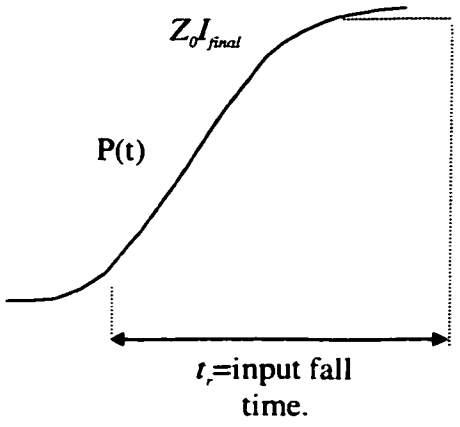


Figure 10.2. The initial output voltage pulse generated by the inverter for the period of time $0<t<2T_0$.

This current depends on the input voltage and has a final value of

$$I_{final} = P_{Cp}\frac{W_p}{L_p}(V_{DD} - |V_{Tp}|)^{\alpha_p}. \tag{10.3}$$

which is reached when the input voltage reaches its final value of zero volts. This initial output voltage pulse is shown in Figure 10.2. This pulse propagates across the line, reaches the open circuit (or small capacitor [36]), and reflects a signal of the

same magnitude and sign back towards the CMOS inverter. Since the PMOS transistor is assumed to be in saturation, the transistor maintains an almost constant current which requires a current reflection coefficient of negative one. Therefore, the voltage reflection coefficient is one. Thus, after a time $2T_o$, the original signal that is launched at time zero is multiplied by three (the initial signal, the signal reflected at the load, and the signal reflected at the transistor). Under the above conditions, the output voltage is

$$Vout = P(t) + \sum_{i=1}^{n} 2P(t - 2iT_0) \qquad for \;\; 2nT_o<t<2(n+1)T_o, \tag{10.4}$$

where $n = 1,2,...m$. m is the time at which the transistor is no longer saturated and enters the linear region. This behavior is illustrated in Figure 10.3. The horizontal parts of the curve have voltage values given by

$$V_{out} = (2n+1)Z_0 I_{final}. \tag{10.5}$$

At the times $(2n+1)T_o$, the output voltage has the values given by (10.5) since these times fall in the middle of the periods $2nT_o < t < 2(n+1)T_o$.

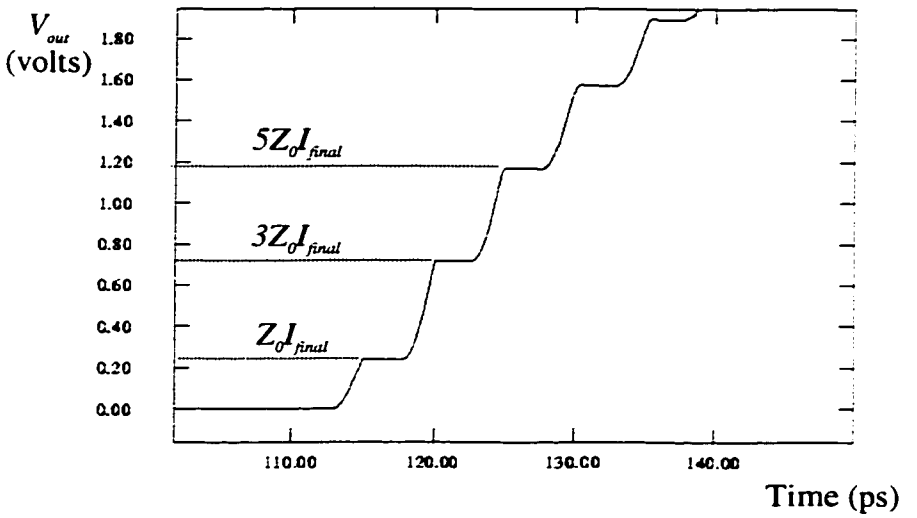


Figure 10.3. Voltage at the output of a saturated transistor connected to an open circuit transmission line.

A transistor driving a lossless transmission line in the matched or underdriven case has an output voltage that follows the response of a transistor driving a capacitance C_L, as shown in Figure 10.4. A CMOS gate is matched to a transmission line if the geometric widths of the transistors are finely adjusted to avoid reflections such that the output impedance of the CMOS gate is equal to the characteristic impedance of the line. The line is said to be underdriven if the widths of the transistors of the CMOS gate driving the transmission line are smaller than the transistor widths in the matched case such that the output impedance of the CMOS gate is greater than the characteristic impedance of the line. Since $T_0 = \sqrt{C_t L_t}$ and $Z_0 = \sqrt{L_t / C_t}$ (see Chapter 2), where L_t and C_t are the total inductance and capacitance of the line, respectively, the value T_0/Z_0 is equal to C_t. Thus, equating C_L to T_0/Z_0 or the total capacitance of the line, V_{out} becomes

$$V_{out} = I_{final} \frac{Z_0}{T_0} t. \tag{10.6}$$

after the input voltage reaches its final value. If V_{out} is sampled at times $(2n+1)T_0$, the resulting expression for V_{out} is

$$V_{out} = (2n+1) Z_0 I_{final}. \tag{10.7}$$

This expression is exactly the same as (10.5). The equivalence of (10.7) and (10.5) demonstrates that the output voltage of a CMOS gate driving a capacitor equal to the total capacitance of a transmission line intersects the output voltage of the CMOS gate driving the transmission line at times $(2n+1)T_0$. Results from the IBM developed circuit simulator AS/X [128] based on a 0.25 μm CMOS technology are shown in Figure 10.5. As the fall (rise) time of the signal at the input of the inverter increases, the slope of the rising (falling) portions of the transmission line response decreases while the intersection points determined by the capacitance approximation remain the same. Thus, while the rise (fall) time of the input signal increases with respect to T_0, the capacitive approximation more accurately matches the transmission line model as shown in Figure 10.5. Once t_r becomes greater than $2T_0$, the horizontal portions of the response are no longer apparent and the two responses coincide. It can be shown that the assumption made here (a lossless transmission line behaves as a single lumped capacitor when $t_r > 2T_0$) for a saturated transistor holds when the transistor also operates in the linear region. This behavior can be seen in Figure 10.5 at the exponential tail of the response at the end of the logical transition where the transistor enters the linear region. When the lumped capacitor response approaches the transmission line response when the transistor is saturated, the two responses also approach each other in the linear region. This behavior can be qualitatively

interpreted by noting that the inductive impedance is given by $j\omega L$ and the capacitive impedance is given by $1 / j\omega C$ where ω is the radial frequency. When the frequency decreases (the rise time increases), the capacitive impedance becomes large compared to the inductive impedance which make the inductance negligible permitting the line to be treated as a single lumped capacitor.

This behavior makes the study of a CMOS gate driving a lossless transmission line unnecessary when $t_r > 2T_0$ because the transmission line model can be simply replaced by a capacitor equal to the total capacitance of the line. Also, on-chip signal transition times are decreasing in next generation VLSI circuits while wires are becoming longer (which increases T_0). Therefore, the more important regime of interest when analyzing the effects of on-chip inductances occurs when $t_r < 2T_0$ (see also Chapter 5). Thus, it is assumed that $t_r < 2T_0$ in the remaining analysis presented in this chapter.

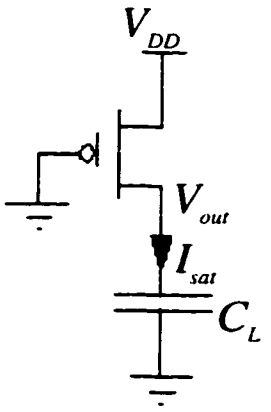


Figure 10.4. A PMOS transistor driving a capacitive approximation of a lossless transmission line.

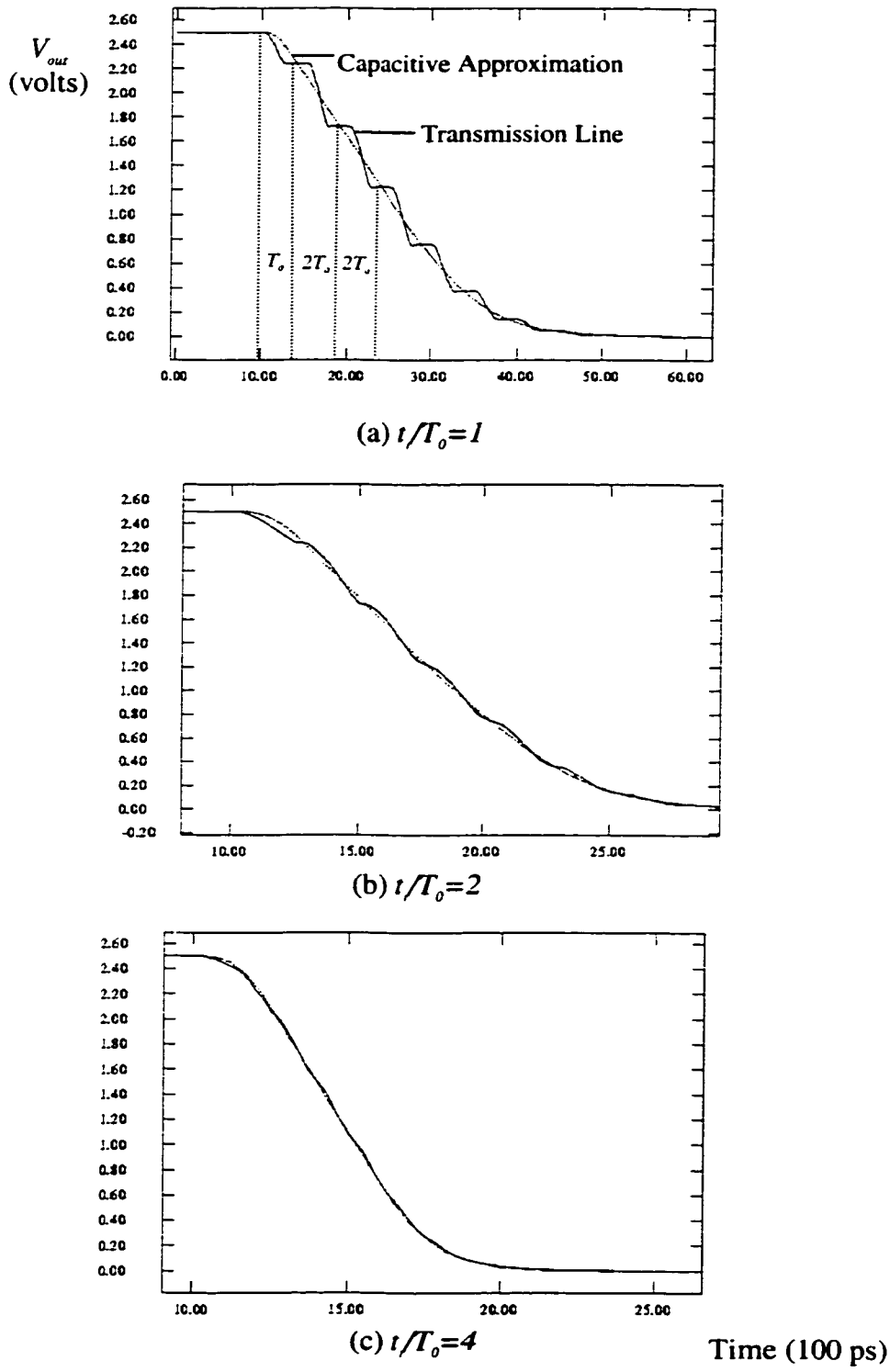


Figure 10.5. Effect of rise time on the capacitive approximation of a transmission line. Ratios of t/T_0 of 1, 2, and 4 are shown in figures (a), (b), and (c), respectively.

10.2 Dynamic and Short-Circuit Power

The dynamic power consumption can be derived from analyzing the behavior of a transmission line interacting with a CMOS gate. This topic is discussed in subsection 10.2.1. It is shown that the dynamic power consumption of a CMOS gate driving a transmission line is the same as the dynamic power consumption of a gate driving a capacitor equal to the total capacitance of the transmission line. In subsection 10.2.2, the short-circuit power consumption of a CMOS gate driving a lossless transmission line is investigated for the regime where $t_r < 2T_0$. The short-circuit to dynamic power ratio is examined in subsection 10.2.3 and compared to the same ratio when the load is a simple capacitor or an *RC* line.

10.2.1 Dynamic Power

A MOS transistor driving a lossless transmission line launches an initial voltage wave with a value $V_1 = I_{sat} Z_0$. This initial voltage signal propagates towards the load and reaches the load at time T_0. Assuming the load is open or is a small capacitor, a voltage wave propagates back towards the transistor with a magnitude V_1 and a current wave $-I_{sat}$. This wave returns to the MOS transistor at time $2T_0$. At that moment the current of the two waves cancels and the voltage adds to $2V_1$. For this period of time (from 0 to $2T_0$), a current of V_1/Z_0 is drawn from the power supply. At $t = 2T_0$, a new voltage wave is initiated with a value $V_2 = I(V_{DD}-2V_1-V_2)Z_0$, where $I(V_{DD}-2V_1-V_2)$ is the current of the transistor with $V_{out} = 2V_1+V_2$. The cycle is repeated and a current V_2/Z_0 is drawn from the supply for the period of time from $2T_0$ *to* $4T_0$. This cycle repeats until the voltage at the output of the transistor reaches V_{DD}. The last voltage wave is assumed to be V_n, where n is the number of traversals of the line

required to reach steady state. The number of iterations can in general be infinite or only one in the perfectly matched case. The output current I_{DS} of a PMOS transistor as a function of time pulling up the output voltage is shown in Figure 10.6.

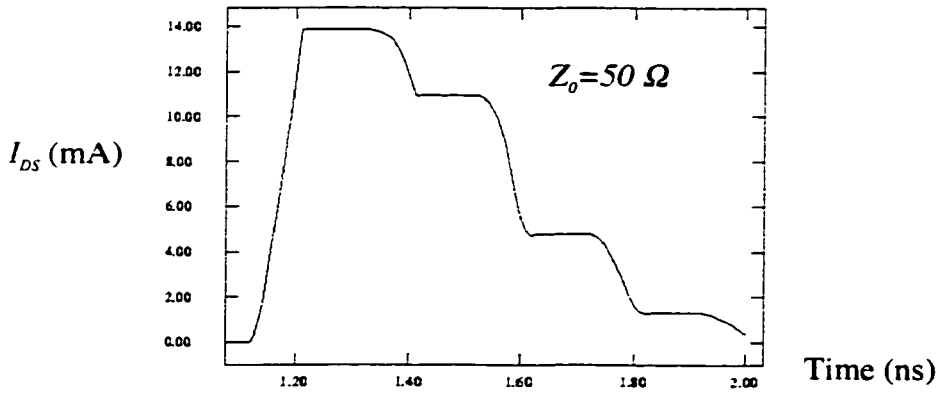


Figure 10.6. Source to drain current of a PMOS transistor driving a lossless transmission line.

Summing these voltages, the following condition is satisfied,

$$V_1 + V_2 + \ldots + V_n = \frac{V_{DD}}{2}, \tag{10.8}$$

since the final voltage is V_{DD} once the transient response reaches the steady state value. From this description, the energy taken from the power supply for a low to high transition is

$$E_{dyn}\,/\,Cycle = V_{DD} \cdot \frac{2T_o}{Z_0} \cdot (V_1 + V_2 + \ldots + V_n). \tag{10.9}$$

Using (10.8) and noting that T_o/Z_o is equal to C_t, the total capacitance of the line, (10.9) evaluates to

$$E_{dyn}\,/\,Cycle = C_t \cdot V_{DD}{}^2. \tag{10.10}$$

This energy is stored in the capacitance of the line and is passed to ground in the next high to low transition as the line is discharged. Thus, (10.10) represents the energy per cycle of the output. The dynamic power is therefore

$$P_{dyn} = C_t \cdot V_{DD}^2 f, \tag{10.11}$$

where f is the frequency of the signal at the output of the CMOS gate. This formula is the same as the dynamic power for a capacitor of the same value as the total capacitance of the line.

10.2.2 Short-Circuit Power

The other component of switching transient power in CMOS circuits is the short-circuit power that occurs when both the NMOS and the PMOS transistors conduct current at the same time. This power is consumed during the rise (fall) time of the input signal when the input signal is between V_{Tn} and $V_{DD}+V_{Tp}$ (where V_{Tn} is the threshold voltage of the N-channel device). The case considered here is when the rise time is less than twice the propagation delay across the line ($t_r < 2T_0$), as discussed in section 10.1. In this case, the reflections do not affect the short-circuit power because the signal returns to the driver after the input signal has reached its final value. Under this condition, the transmission line appears as a resistance with a value Z_0. The equivalent circuit of a CMOS inverter driving a lossless transmission line for the period of time 0 to T_0 is shown in Figure 10.7, where C_0 is the intrinsic drain capacitance of a CMOS inverter.

The differential equation describing the KCL for the output node is

$$(I_n - I_p) = C_0 \cdot \frac{d(V_{DD} - V_{out})}{dt} + \frac{(V_{DD} - V_{out})}{Z_0}, \tag{10.12}$$

where I_n and I_p are the currents of the NMOS transistor and the PMOS transistor, respectively. Note that the resistance Z_0 representing the transmission line is connected between V_{DD} and V_{out} because the line is assumed to be charged to V_{DD} and that $d(V_{DD}-V_{out})/dt = -d(V_{out})/dt$. The NMOS transistor is assumed to be in the saturation region during the rise time of the input signal. This assumption is justified because the output is initially V_{DD} which makes V_{DSn} large where V_{DSn} is the drain to source voltage of the NMOS transistor. Also, since the input is still rising, V_{GSn} is smaller than V_{DD}, where V_{GSn} is the gate to source voltage of the NMOS transistor. Thus, the saturation condition in [108], $V_{DSn} > P_{Vn}(V_{GSn}-V_{Tn})^{\alpha n/2}$, is satisfied during the rise time of the input signal. P_{Vn} is a constant that characterizes the current drive capability of the NMOS transistor when operating within the linear region [108]. This assumption is further accurate in deep submicrometer technologies because of the early saturation phenomenon [108]. This aspect can be understood by noting that α decreases for deep submicrometer devices and approaches one, which makes the aforementioned saturation condition be satisfied for a larger range of the output voltage. On the other hand, the PMOS transistor starts in the linear region and then enters the saturation region. The short-circuit current describes the current through the PMOS transistor during a rising input. The signal at the input of the CMOS driver is

$$V_{in} = kt = \frac{V_{DD}}{t_r} t \,. \tag{10.13}$$

before the input signal reaches V_{DD} (*i.e.*, $t < V_{DD}/k$).

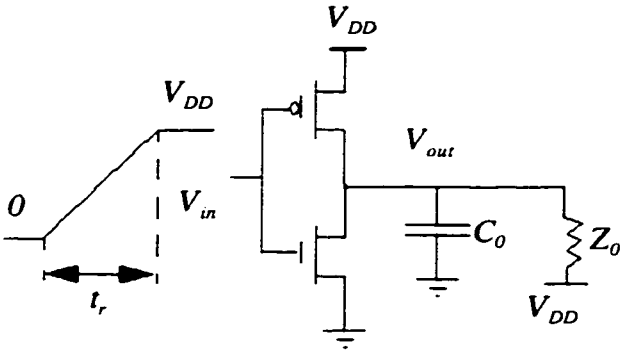


Figure 10.7. Equivalent circuit of a CMOS driver driving a lossless transmission line for $0 < t < 2T_0$.

Based on the alpha power law, when the PMOS transistor is in saturation the short-circuit current is

$$I_{SC} = P_{Cp} \cdot \frac{W_p}{L_p} \cdot (V_{DD} - kt - |V_{Tp}|)^{\alpha p}. \tag{10.14}$$

When the PMOS transistor operates in the linear region, the solution of (10.12) appears intractable because V_{DSp} (the drain to source voltage of the PMOS transistor) is a function of V_{out}. Therefore, the current from the output capacitor C_0 is assumed to be small compared to the current from the transmission line. This assumption is accurate in deep submicrometer technologies because the transistors have small parasitic capacitances and high current levels. Also Z_0 is typically in the range of 30 to 60 ohms, see *e.g.*, [48], [51], which results in large currents compared to the current sourced by the parasitic capacitance C_0. Under this assumption and using the alpha power law model, the output voltage is

$$V_{out} = V_{DD} - \frac{Z_0 \cdot P_{Cn} \cdot \frac{W_n}{L_n} \cdot (kt - V_{Tn})^{\alpha n}}{1 + \frac{P_{Cp}}{P_{Vp}} \cdot \frac{W_p}{L_p} \cdot \left(V_{DD} - kt - |V_{Tp}|\right)^{\frac{\alpha p}{2}} \cdot Z_o}. \tag{10.15}$$

for the time $0 < t < t_r$ where $t_r < 2T_0$ according to the assumption made in section 10.1. Noting that V_{DD}-V_{out} is V_{SD} of the PMOS transistor, the short circuit current of a CMOS inverter loaded by an LC transmission line is

$$I_{SC} = \frac{Z_0 \cdot P_{Cn} \cdot \frac{W_n}{L_n} \cdot (kt - V_{Tn})^{\alpha n}}{\frac{P_{Vp}}{P_{Cp}} \cdot \frac{L_p}{W_p} \cdot \left(V_{DD} - kt - |V_{Tp}|\right)^{-\frac{\alpha p}{2}} + Z_o}. \tag{10.16}$$

A CMOS gate driving a transmission line can be most efficiently characterized by the matching factor λ, where

$$\lambda = \frac{2S_p W_p Z_0}{V_{DD}}. \tag{10.17}$$

and S_p is a technology dependent constant. A detailed derivation of λ is given in Appendix D. When $\lambda = 1$, the transistor is optimally matched to the transmission line. If λ is less than one, the transmission line is underdriven and the response suffers from a slow output rise time. If λ is greater than one, the transmission line is overdriven and the response suffers from overshoots and undershoots that can cause reliability problems. If the overshoots and undershoots are large enough, logical errors can occur. Voltages higher than the supply voltage and lower than ground create large electric fields that can deteriorate the oxide and increase hot electron effects. Also, out of rail voltages can forward bias the junctions between the drain and source and the substrate, the n-well, or the p-well. When these junctions are forward biased, current flows directly into the wells or the substrate. This behavior is undesirable because it wastes current, dissipating extra power, and can induce other reliability problems within the substrate. Thus, the range defined by $0.3 < \lambda < 1.6$ is arbitrarily chosen to represent the range of interest characterizing practical matching

conditions. The analytical solutions of (10.14) and (10.16) are compared to AS/X simulations in Figure 10.8 assuming $t_r = 100\ ps$ and using λ as a design parameter. The characteristic impedance of the line Z_0 is kept constant while the PMOS transistor width W_p is varied to change λ. The analytical solution shows good agreement with the circuit simulations for a wide range of λ.

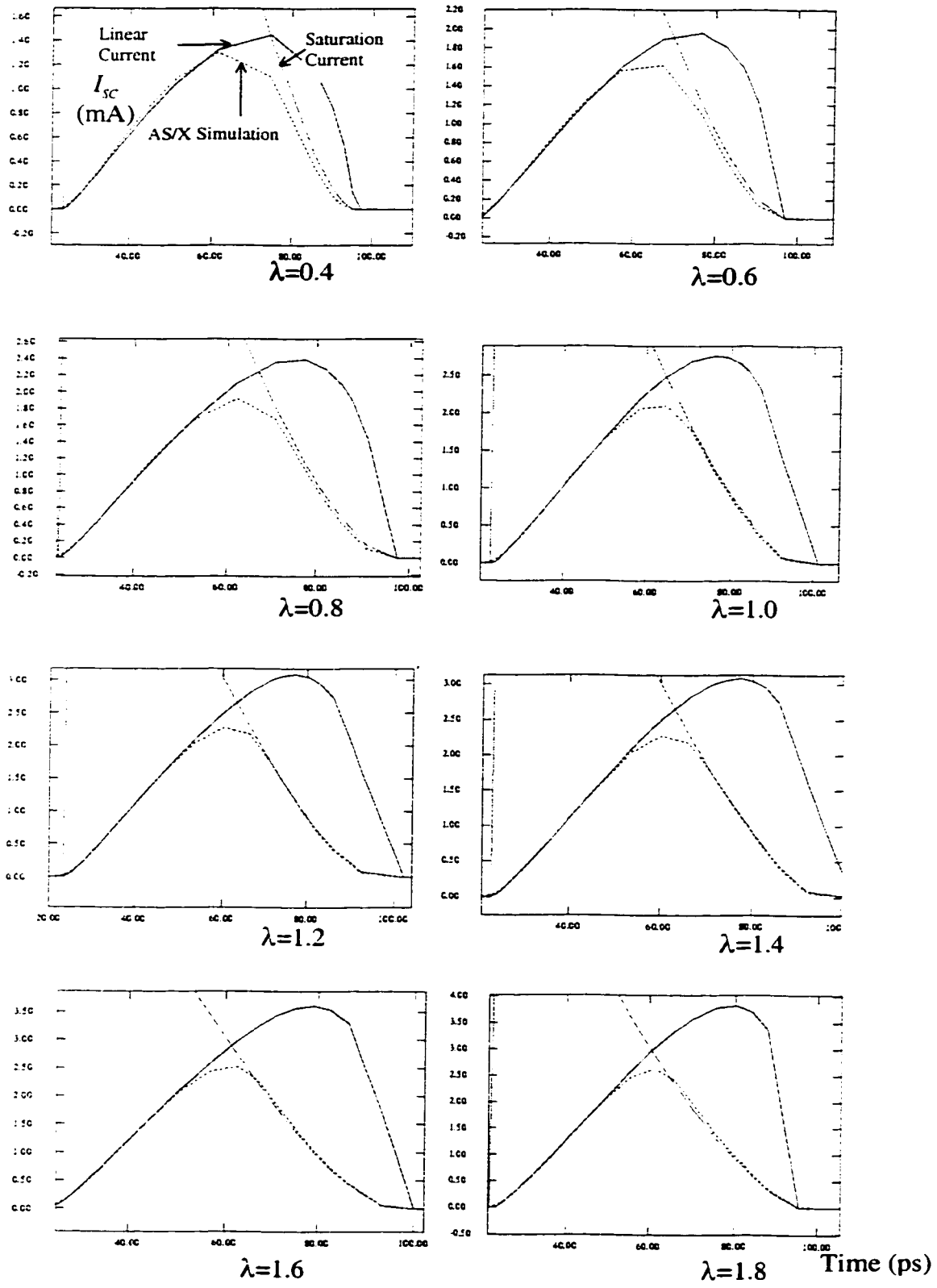


Figure 10.8. Analytical solutions of the short-circuit current in (10.14) and (10.16) compared to AS/X simulations for λ = 0.4, 0.6, 1.0, 1.2, 1.4, 1.6, and 1.8. The AS/X curve is denoted by dashes, (10.14) is denoted by dots, and (10.16) is solid.

The short-circuit energy per transition can be calculated from

$$E_{SC} / Transition = Area_{SC} \cdot V_{DD}. \tag{10.18}$$

where $Area_{SC}$ (in coulombs) is the area under the short-circuit current curve. This area can be approximated by a triangle [38] whose base is given by $(V_{DD}-V_{Tn}+V_{Tp})t/V_{DD}$ and height is given by I_{peak} (the maximum point on the short-circuit current curve). Thus, the short-circuit energy per transition is

$$E_{SC} / Transition = \frac{I_{peak}}{2}(V_{DD} - V_{Tn} - |V_{Tp}|)t_r K_c, \tag{10.19}$$

where K_c is a correction factor. Note in Figure 10.8 that the analytical solution deviates from the simulated results in a consistent way at the point of intersection of the saturated curve and the linear curve. This deviation is due to the non-monotonic nature of the alpha power law model used to characterize the devices. Due to this consistent error at the saturation/linear break point, a constant correction factor K_c is used and calibrated at $\lambda = 1$.

I_{peak} can be calculated by equating (10.14) and (10.16). Alternatively, I_{peak} can be calculated as

$$I_{peak} = K(\lambda)W_p, \tag{10.20}$$

where $K(\lambda)$ is determined from varying λ and calculating I_{peak}/W_p. $K(\lambda)$ for a specific 0.25 μm CMOS technology is plotted in Figure 10.9.

$K(\lambda)$ quantifies in I_{peak} the effect of the output waveform shape on the short-circuit current. Note that $K(\lambda)$ saturates to an asymptotic value. This behavior can be explained by observing how the shape of the output signal varies with λ in Figure 10.10. Note that the shape of the output voltage waveform depends heavily on λ for

small λ and this dependence saturates as λ increases. The short-circuit current depends upon the output voltage waveform because the drain to source voltage across the PMOS transistor is dependent on V_{out}. The gate to source voltage of the PMOS transistor depends only on the input signal of the transistor (since the source is connected to the power supply). The output voltage with respect to the input voltage is determined by λ, as shown in Figure 10.10. The peak current of the PMOS transistor is completely determined by λ and the geometric width of the PMOS transistor W_p, as given by (10.20). The transition time of the input signal results only in a shift of the time at which the peak current occurs without changing the magnitude of the peak current. The AS/X circuit simulator is used to quantify the accuracy of these analytic equations, such as the comparison of (10.19) with AS/X in Table 10.1. The analytical solution shows good agreement (less than 1% error for $\lambda \geq 1$) with the circuit simulations for a wide range of λ and exhibits a maximum error of 11% for small λ.

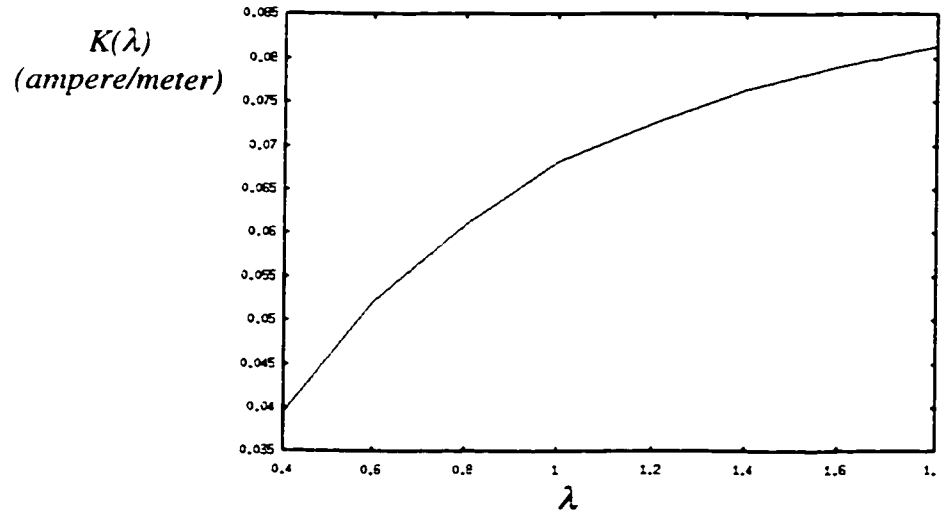


Figure 10.9. $K(\lambda)$ verses λ.

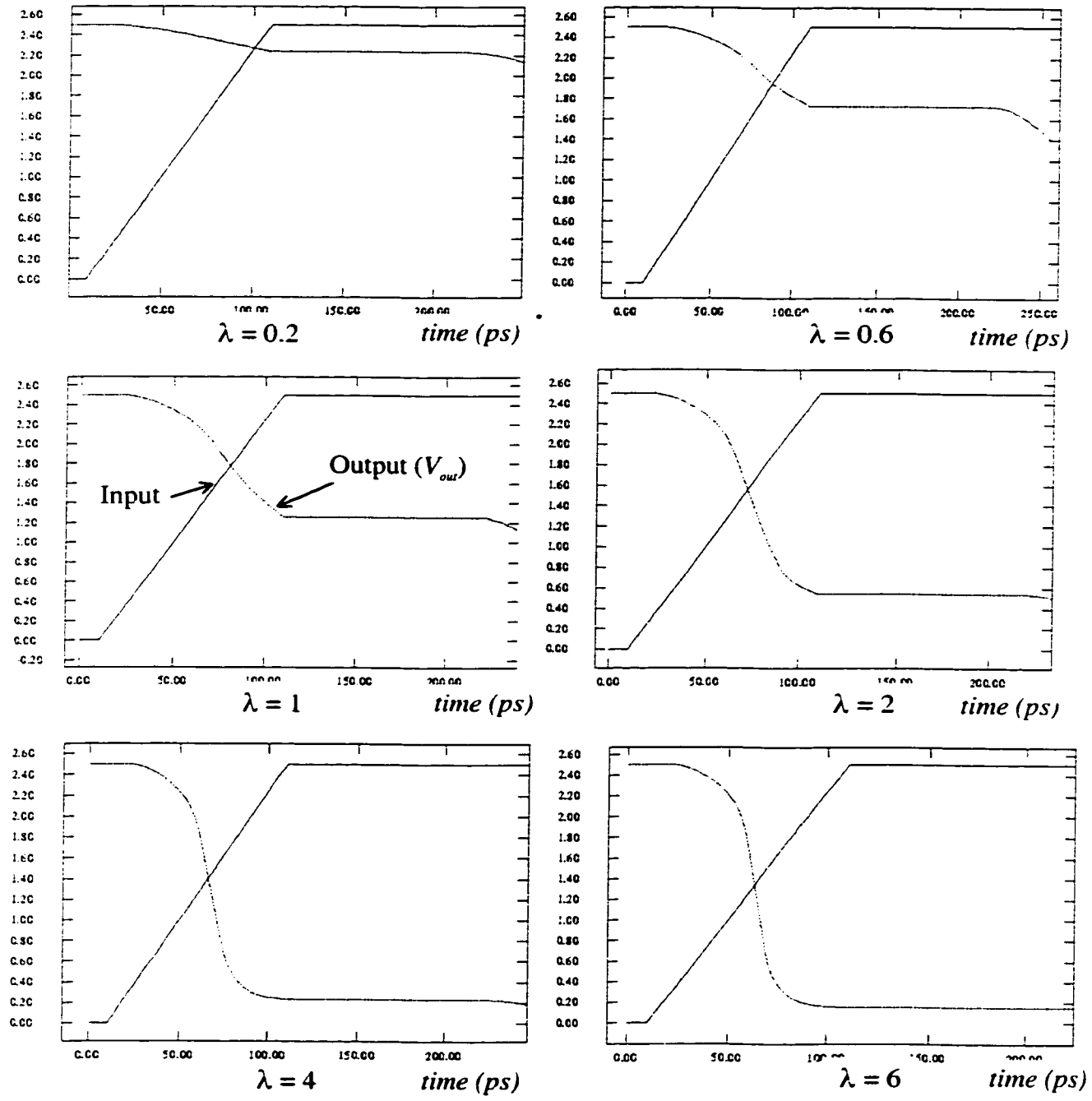


Figure 10.10. The dependence of the output voltage on λ for $\lambda = 0.2$, 0.6, 1.0, 2.0, 4.0, and 6.0.

Table 10.1 AS/X simulations compared to analytical solution for E_{sc}/transition (in joules).

λ	AS/X Simulation x10^{-15}	Analytical x10^{-15}	% Error
0.4	131.20	116.00	11.00
0.6	161.45	152.50	5.50
0.8	182.77	178.95	2.09
1.0	199.60	199.60	0.00
1.2	212.55	212.30	0.12
1.4	224.55	223.20	0.60
1.6	231.93	231.37	0.24
1.8	239.68	238.20	0.62

10.2.3 Short-Circuit to Dynamic Power Ratio

Assuming a symmetric CMOS gate, the short-circuit power is

$$P_{SC} = I_{peak}(V_{DD} - V_{Tn} - |V_{Tp}|)t_r f. \tag{10.21}$$

As described previously, the dynamic power is

$$P_{dyn} = \frac{T_0}{Z_0} \cdot V_{DD}{}^2 f. \tag{10.22}$$

Dividing (10.21) by (10.22), the magnitude of the dynamic power can be compared to the magnitude of the short-circuit power. The resulting ratio is

$$\frac{P_{SC}}{P_{dyn}} = K(\lambda)\lambda \cdot \frac{(V_{DD} - V_{Tn} - |V_{Tp}|)}{V_{DD}} \frac{t_r}{T_0} \cdot \frac{K_c}{2S_p}. \tag{10.23}$$

The ratio between the short circuit power and the dynamic power depends upon the matching condition λ of the transmission line impedance to the output impedance of the CMOS gate and the ratio between the rise time of the input signal to the time of flight of the waves across the transmission line (t/T_0). The dependence on the supply voltage is fairly weak. The dependence only exists if the supply voltage and the

threshold voltages scale differently. The dependence of the short-circuit to dynamic power ratio on λ is shown in Figure 10.11. As the matching condition moves from underdriven to matched to overdriven, the short-circuit to dynamic power ratio increases. This ratio is less than 7% for the matched (λ=1) and underdriven cases (λ<1). This low ratio is due to the small voltage step in the output voltage while the input signal is still changing, as can be seen from Figure 10.10. In the matched case, the input signal transitions approximately twice as fast as the output signal since the input signal transitions from *0* to V_{DD} at the same time as the output signal transitions from *0* to $V_{DD}/2$. This characteristic explains the low short-circuit to dynamic power ratio in a matched or underdriven circuit. Therefore, it is preferable to not overdrive the line (*i.e.*, make λ>1) in order to decrease the short-circuit power. The classical design criterion for driving a capacitive load is to maintain equal input and output transition times which gives rise to a short-circuit power of approximately *20%* of the dynamic power [6]. For *RC* loads, the P_{SC}/P_{Dyn} ratio is even greater because the voltage drop across the load resistance makes the source to drain voltage large once the transistor begins to switch.

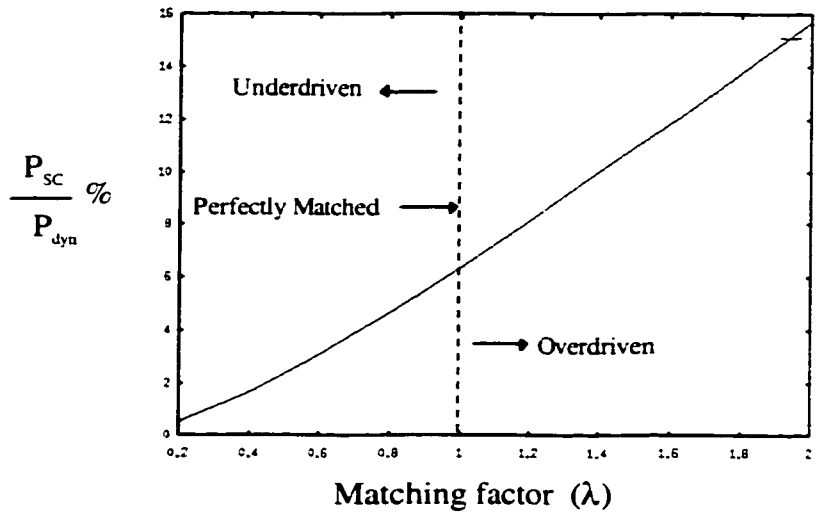


Figure 10.11. Dependence of the short circuit to dynamic power ratio on λ.

The maximum value of t/T_0 to exhibit nontrivial inductance effects is two. If t/T_0 is larger than two, the transmission line can be approximated by a capacitor, as is illustrated in section 10.1. As the rise time of the input signal t_r increases, the short-circuit power increases, since there is more time for the short-circuit current to flow. The rise time t_r does not affect the dynamic power and thus increasing t_r increases the ratio of the short-circuit power to the dynamic power. Dynamic power increases with increasing T_0 because the total capacitance of the line is T_0/Z_0. Thus, as T_0 increases, the capacitance of the line increases linearly which increases the dynamic power linearly. The short-circuit power is not affected by T_0 as long as T_0 is greater than half the input rise time, making the ratio of the short-circuit power to the dynamic power decrease as T_0 is increased.

10.3 Conclusions

A closed form solution for the output voltage of a CMOS gate driving a lossless transmission line is presented. Transmission line effects are shown to be

insignificant when the transition time of the input signal at the CMOS gate driving a transmission line is greater than twice the time of flight of the signals across the line. The dynamic and short-circuit power consumed by a CMOS gate driving a lossless transmission line is investigated. The dynamic power of a CMOS gate driving a lossless transmission line is shown to be the same as that of a CMOS gate driving a capacitor equal to the total capacitance of the line. A closed form solution for the short-circuit power is presented that agrees with circuit simulations within 11% error for a wide range of the matching factor λ. An expression for the short-circuit to dynamic power ratio is presented that shows that the short-circuit power is below 7% of the dynamic power for λ less than or equal to one. Thus, the short-circuit power for the case of an *LC* load is much less than that of the case of an *RC* load. In the case of an *RLC* load, the short-circuit power is in the middle, greater than the case of an *LC* load but less than the case of an *RC* load.

Chapter 11 Exploiting On-Chip Inductance in High Speed Clock Distribution Networks

As discussed in previous chapters, the importance of on-chip inductance is continuously increasing with faster on-chip rise times, wider wires, and the introduction of new materials for low resistance interconnect [45]-[61]. These increasing inductance effects are typically viewed as an additional problem which must be managed. This perspective is accurate in many respects. Dealing with inductance requires efficient inductance extraction methods [47]-[51], [55] and increases the computational processing time of CAD tools. Furthermore, underdamped responses can cause reliability issues and increase noise in integrated circuits. Thus, to date, much of the effort within industry has focused on limiting the effects of inductance, *e.g.*, [45], [52], and [44]. However, suppressing inductance effects is typically at the expense of deteriorating the performance of an integrated circuit in terms of speed, power consumption, and/or device area. An example of an industrial clock distribution network is presented towards the end of this chapter that illustrates this point. Design methodologies can be developed to exploit useful effects of on-chip inductance while maintaining noise at acceptable levels so as to guarantee the reliable performance of an integrated circuit.

The objective of this chapter is to describe certain beneficial effects of inductance on the performance of an integrated circuit. These effects are described in section 11.1. The design of an industrial clock distribution network is presented in

section 11.2 to illustrate how inductance effects can significantly improve circuit performance. Finally, some conclusions are provided in section 11.3.

11.1 Useful Inductance Effects

The effects of inductance on the rise time of signals within an integrated circuit is discussed in subsection 11.1.1. It is shown that increasing inductance effects result in faster signal rise times. In subsection 11.1.2, the effects of inductance on the area of repeaters inserted to reduce signal degradation along long interconnects is discussed. It is shown that the total repeater area decreases as inductance effects increase. The effects of inductance on the power dissipated by CMOS gates is discussed in subsection 11.1.3, particularly the dramatic decrease in the short-circuit power consumption of CMOS gates. Also, the decreased device area required to drive inductive lines results in less device capacitance, which further decreases the total power consumption.

11.1.1 Effects of Inductance on the Signal Rise Time

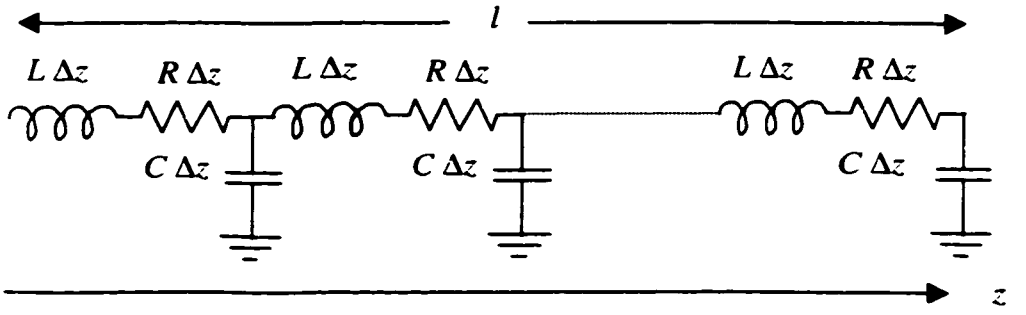


Figure 11.1. RLC transmission line.

The faster rise times of signals in a high speed integrated circuit as inductance effects increase can be best explained by examining the signal propagation characteristics in a lossy *RLC* transmission line. Signals propagating across an *RLC* transmission line travel with a frequency dependent velocity given by

$$v = \frac{1}{\sqrt{LC}\sqrt{\frac{1}{2}\left(\sqrt{\left(1+\left(\frac{R}{\omega L}\right)^2\right)}+1\right)}}, \tag{11.1}$$

where R, L, and C are the resistance, inductance, and capacitance per unit length of the line, respectively, and ω is the radial frequency. Furthermore, the signals attenuate as they travel across the line with an attenuation constant α given by [120]

$$\alpha = \omega\sqrt{LC}\sqrt{\frac{1}{2}\left(\sqrt{\left(1+\left(\frac{R}{\omega L}\right)^2\right)}-1\right)}. \tag{11.2}$$

The attenuation constant and the speed of propagation as functions of frequency are plotted in Figure 11.2. In the limiting case of a lossless line representing maximum inductance effects, the attenuation constant α is zero and the propagation speed becomes frequency independent and is

$$v = \frac{1}{\sqrt{LC}}. \tag{11.3}$$

Thus, as inductance effects increase, a pulse propagating across an *RLC* line maintains the high frequency components in the edges, improving the signal rise and fall times. This behavior is qualitatively illustrated in Figure 11.3.

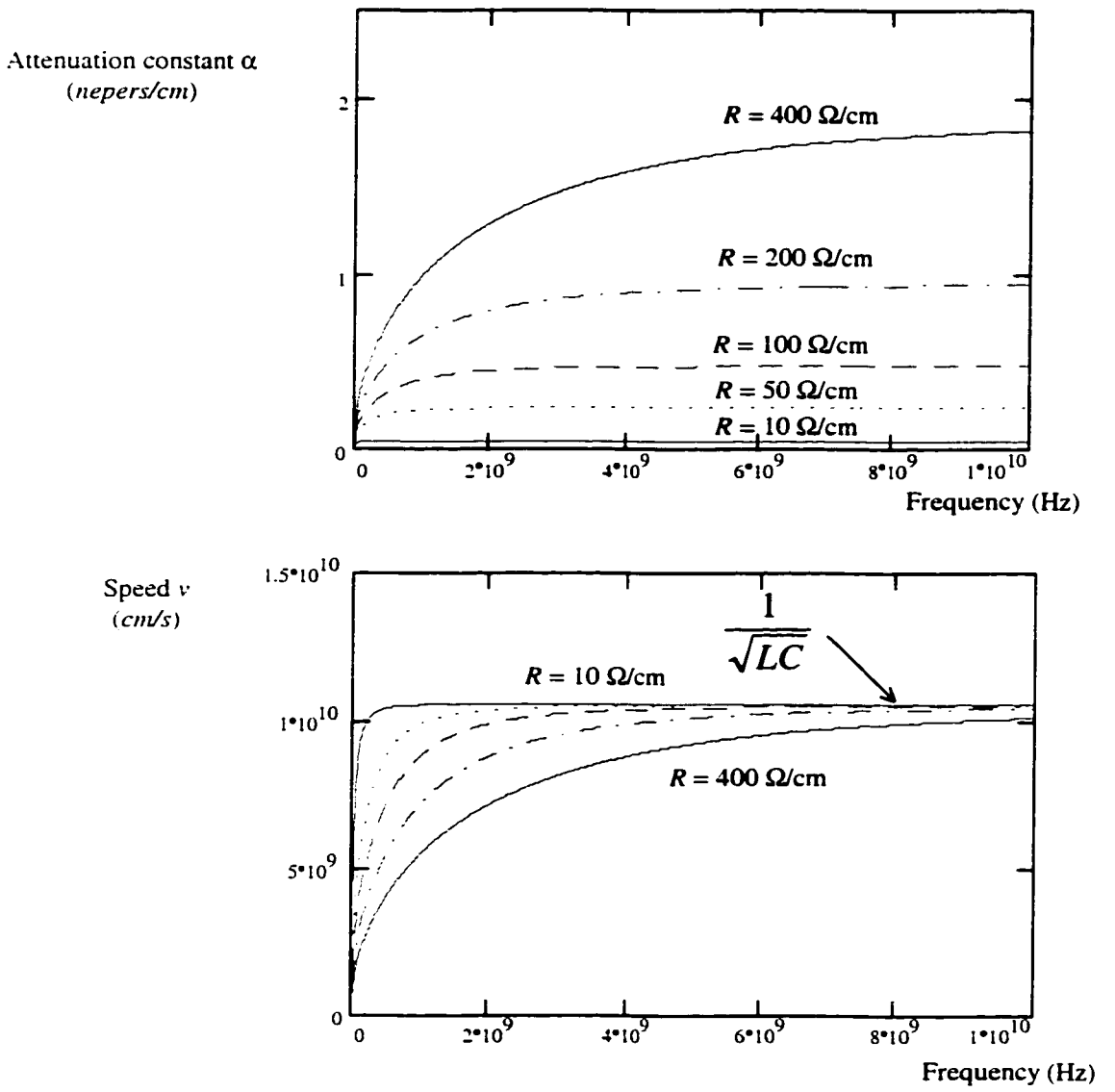


Figure 11.2. The attenuation constant and propagation speed versus frequency. $L = 10$ *nH/cm*, $C = 1$ *pF/cm*, and R is 10, 50, 100, 200, and 400 Ω/*cm*, respectively.

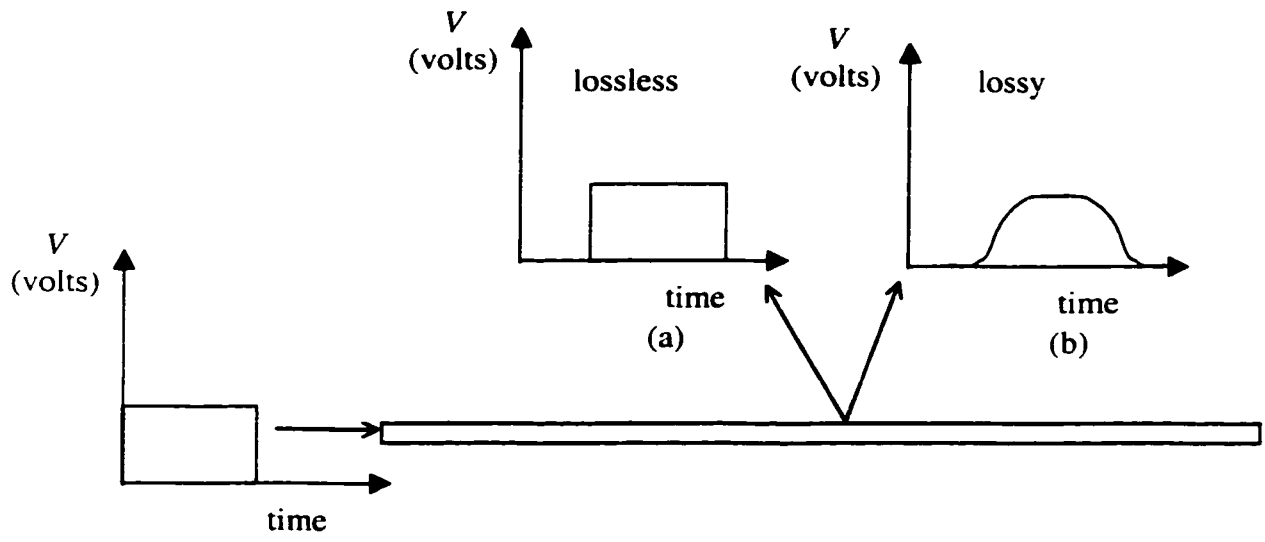


Figure 11.3. Signal dispersion of a square wave signal in a lossy transmission line. a) Pulse shape after traveling along a lossless transmission line. b) Pulse shape after traveling along a lossy transmission line.

11.1.2 Effects of Inductance on the Repeater Insertion Process

As discussed in Chapter 6 and Chapter 9, repeater insertion has become a common design methodology for driving long resistive interconnect [37]-[44]. Since the propagation delay has a square dependence on the length of an *RC* interconnect line, subdividing the line into shorter sections by inserting repeaters is an effective strategy for reducing the total propagation delay. Currently, typical high performance circuits have a significant number of repeaters inserted along the global interconnect lines. These repeaters are large gates and consume a significant portion of the total circuit power.

The repeater area for minimum propagation delay of an *RLC* interconnect decreases as inductance effects increase due to the sub-quadratic dependence of the

propagation delay on the length of the interconnect [110]. Hence, inserting repeaters based on an *RC* model and neglecting inductance results in a larger repeater area than necessary to achieve a minimum delay. The magnitude of the excess repeater area when using an *RC* model depends upon the relative magnitude of the inductance within the *RLC* tree. The reduced number of inserted repeaters also simplifies the layout and routing constraints. Finally, the reduced repeater area greatly reduces the power consumed by the repeaters in an integrated circuit.

11.1.3 Effects of Inductance on Power Dissipation

Power consumption is an increasingly important design parameter with mobile systems and high performance, high complexity circuits such as leading edge microprocessors. If the frequency of switching is f cycles per second, then the dynamic power consumption is described by the well-known formula,

$$P_{dyn} = C_t V_{DD}^2 f. \tag{11.4}$$

The dynamic power depends only on the total load capacitance, the supply voltage, and the operating frequency. As discussed in subsection 11.1.2, increasing inductance effects result in fewer number of repeaters as well as smaller repeater size. The smaller size and number of repeaters therefore significantly reduces the total capacitance of the repeaters and, consequently, reduces the total dynamic power consumption.

As described in Chapter 10, the short-circuit power [6]-[8] results from the NMOS and PMOS blocks of a CMOS gate being on simultaneously during the rise and fall times of the input signal, creating a current path between the power supply and ground. It has been shown in Chapter 10 that the short-circuit power consumption

of a CMOS gate decreases as the load driven by the gate exhibits more inductance effects. In addition to this trend, it has been shown in subsection 11.1.1 that the inductance reduces the rise time of the signals in an integrated circuit, which further reduces the short-circuit power. To quantify this effect, consider the circuit configuration shown in Figure 11.4. A fast input signal drives CMOS gate 1 which in turn drives an *RLC* transmission line. The output at the far end of the *RLC* transmission line is the input to the second gate V_{in2}. Gate 2 drives a capacitive load C_2. The short-circuit energy consumed by gate 2 per cycle is listed in Table 11.1 where the total inductance of the transmission line is varied while the resistance and capacitance are maintained constant. The data listed in Table 11.1 are also plotted in Figure 11.5. Note that as inductance effects increase, the short-circuit power consumption significantly decreases due to the faster input rise time.

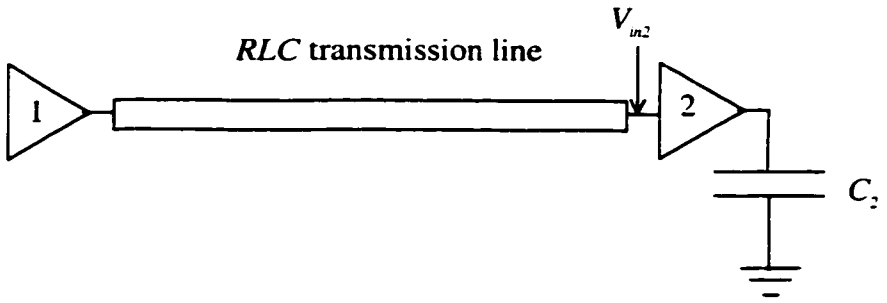


Figure 11.4. A CMOS gate driving another CMOS gate with an *RLC* transmission line connecting the two gates. The second gate drives a capacitive load.

Table 11.1 Simulations of the short-circuit energy consumed per cycle by gate 2 shown in Figure 11.4. The total inductance of the transmission line is varied while the resistance and capacitance are held constant at 100 Ω and 1 pF, respectively.

Total inductance of the line L_t (*nH*)	Short-circuit power at gate 2 (*pJ/cycle*)
1	2.20
2	2.05
4	1.66
6	1.27
8	1.00
10	0.83

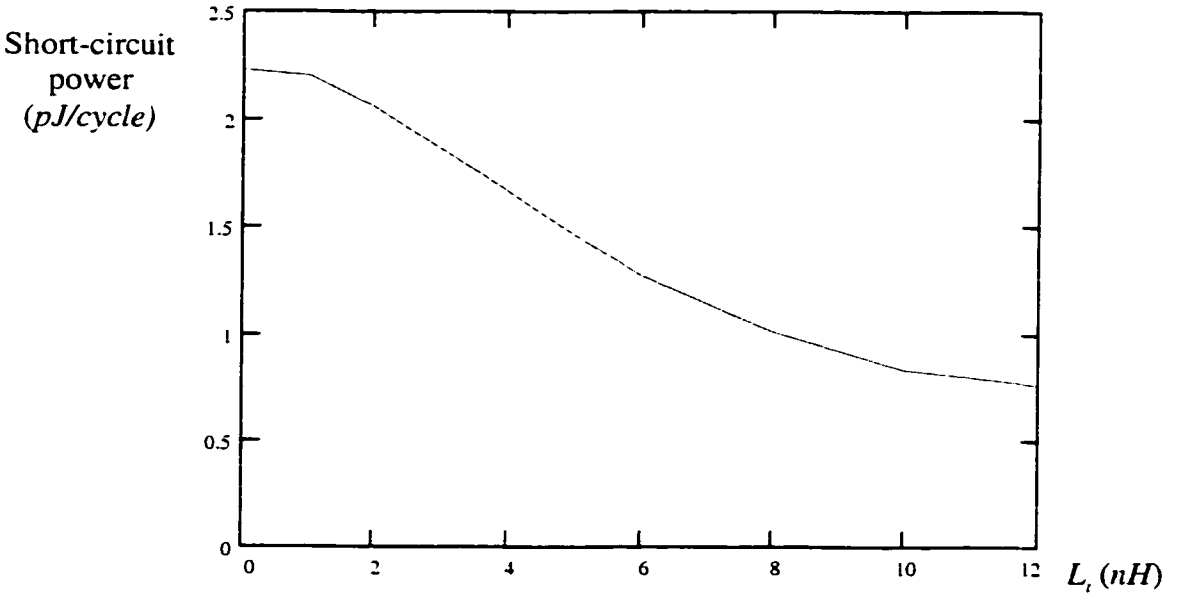


Figure 11.5. Simulations of the short-circuit energy consumed per cycle by gate 2 shown in Figure 11.4 versus the inductance of the transmission line. The total resistance and capacitance of the line are maintained constant at 100 Ω and 1 pF, respectively.

The effect of smaller repeater sizes on the short-circuit power consumption is significant. By decreasing the widths of the transistors, the short-circuit current decreases because the current drive of the NMOS and PMOS transistors is linearly

proportional to the transistor width. Also, by decreasing the width of the transistors, the output transition time becomes slower which decreases the source-to-drain voltage of the transistor passing the short-circuit current. Since the output current is proportional to the source-to-drain voltage of a MOS transistor operating in the linear region, the short-circuit current is smaller with decreasing transistor size. Thus, decreasing the transistor sizes has a two fold effect on the short-circuit power. In general, the short-circuit power has a super linear dependence on gate size. AS/X [128] simulations of the short-circuit energy/cycle of a CMOS gate driving a constant capacitance of 0.2 pF with a 100 ps input rise time versus gate size are depicted in Figure 11.6. The super linear behavior is evident in Figure 11.6. Thus, the smaller repeater size and number significantly reduces the overall short-circuit power.

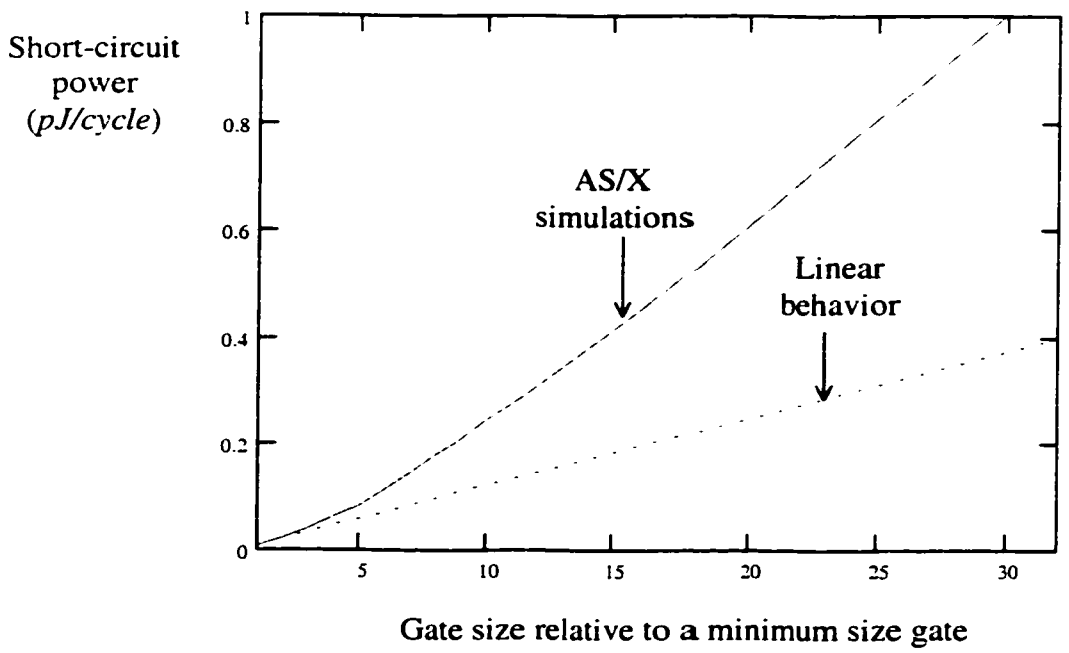


Figure 11.6. Simulations of the short-circuit energy consumed per cycle by a gate driving a load capacitance of 0.2 pF versus the gate width. The rise time of the input signal is 100 ps.

11.2 Clock Distribution Network Example

The clock distribution network significantly affects the performance of an integrated circuit and consumes a large portion of the total chip power (typically 30% to 40%) [62], [63]. Designing an efficient and reliable clock distribution network is of primary importance for a high performance integrated circuit. To illustrate the concepts discussed in section 11.1, the clock distribution network of an integrated circuit is investigated. The integrated circuit has been designed using a 0.18 μm IBM CMOS technology with copper interconnect. The supply voltage for this technology is 1.8 volts and the target frequency of the circuit is 250 MHz. The integrated circuit has four primary modules and several other smaller modules.

The clock distribution network at the top level is composed of a wide buffer that drives a four-node H-tree carrying the clock signal to the center of the four quadrants of the integrated circuit. At the center of each of the four quadrants, a local central wide buffer receives the clock signal and drives the local clock distribution network of each quadrant. Each local central buffer drives a clock tree connected to an average of 1350 sinks. At each sink, a final buffer (a CMOS inverter) receives the clock signal and drives the final group of flip-flops. Each of the final buffers drives a capacitive load of approximately 250 fF. The structure of the local clock distribution network of this module is schematically depicted in Figure 11.7.

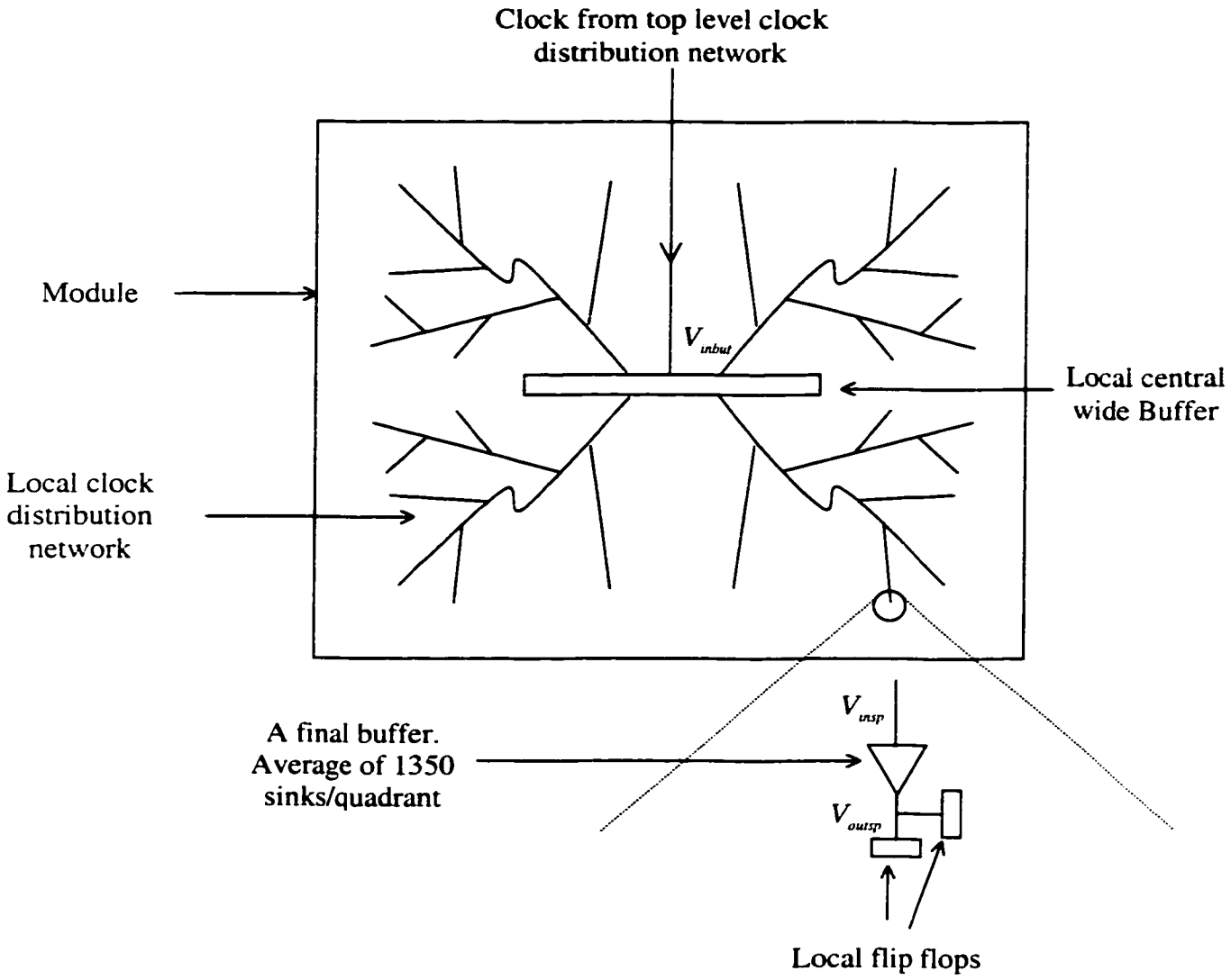


Figure 11.7. Local clock distribution network of a primary quadrant of a large integrated circuit.

The top level and local clock distribution networks have been initially simulated with wires sized to satisfy the design constraints of the clock tree. The slew should be within 5% of the clock period at the input of the latches and the clock delay (or the phase delay) must be less than the clock period. AS/X [128] simulated waveforms at the input of the central buffer V_{inbuf} and at the inputs of the final buffers V_{insp} are shown in Figure 11.8. The initial choice of wire sizes results in degraded

signal waveforms on the internal nodes of the clock distribution network. Note that the rise time of the signals illustrated in Figure 11.8 is greater than one ns. The final buffers restore the signal rise time to 200 ps at the input of the local flip flops V_{outsp}. This faster rise time is necessary to maintain stable operation of the flip flops. Thus, the performance of the clock distribution network satisfies a target cycle period of 4 ns. Note also that this clock distribution network suffers no inductance effects, therefore an *RC* model can be used to model the clock distribution network.

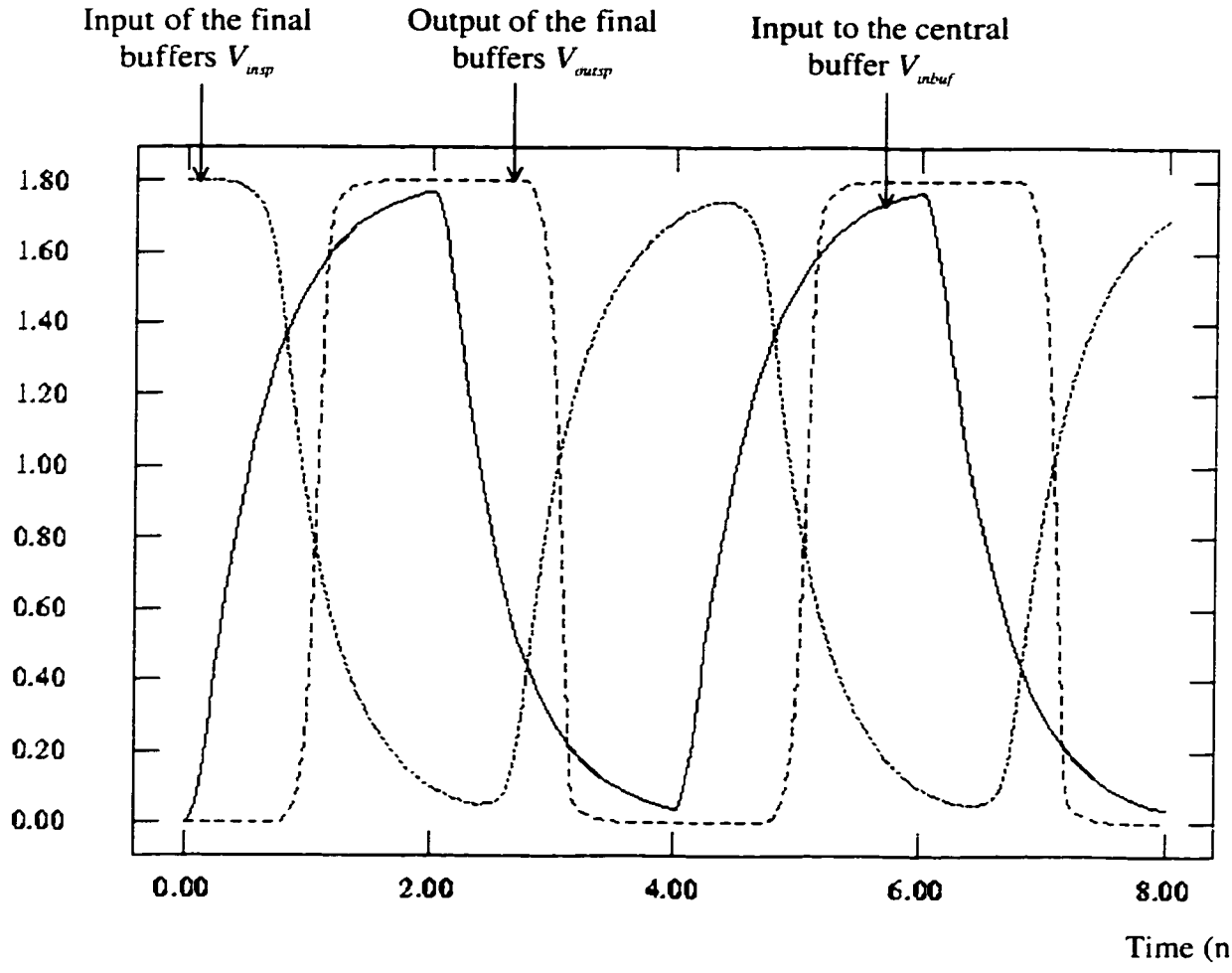


Figure 11.8. AS/X [128] simulations of the signals at the input of the central buffer V_{inbuf}, at the input of the final buffers V_{insp}, and at the output of the final buffers V_{outsp} for the local clock distribution network shown in Figure 11.7 with narrow wires.

The power consumption of the clock distribution network, however, is excessively high due to the slow signal rise times at the inputs of the central buffer and the final buffers. AS/X simulations of the dynamic and short-circuit power consumption of the central buffer are shown in Figure 11.9. The current depicted in Figure 11.9 is drawn from the supply voltage V_{DD} through the PMOS network. When the output is pulled down, this current represents the short-circuit current. When the output is pulled high, the current from the supply represents the sum of the short-circuit current (through the N-channel transistor) and the dynamic current charging the output capacitance. The energy shown in Figure 11.9 is the integration of the supply current multiplied by the supply voltage and represents the total energy consumed by the gate at any given time. Note in Figure 11.9 that the short-circuit power is much higher than the dynamic power consumption, constituting about 80% of the total power consumption of the central buffer. This large amount of short-circuit current directly contradicts the common conception that the short-circuit power contributes less than 20% of the total power consumption [6], [7]. This 20% figure is typically true when the input and output rise times are close to each other. However, for this clock distribution network example, the input rise time is extremely slow and the final buffers provide sufficient current to enable small rise times of 200 ps at the inputs of the flip flops. AS/X simulations of the dynamic and short-circuit power of one of the final buffers are shown in Figure 11.10. Note again that the short-circuit power dominates the dynamic power. The dynamic and short-circuit power consumption of the central buffer and the final buffers are listed separately in Table 11.2.

Table 11.2 Dynamic and short-circuit energy of the central buffer and the final buffers in the local clock distribution network depicted in Figure 11.7 with narrow wires

	Dynamic power (pJ/cycle)	Short-circuit power (pJ/cycle)	Total power (pJ/cycle)
Central buffer	68	133	201
Single final buffer	0.71	1.86	2.57
All final buffers	958	2511	3469
Local clock distribution network (total)	1026	2644	3670

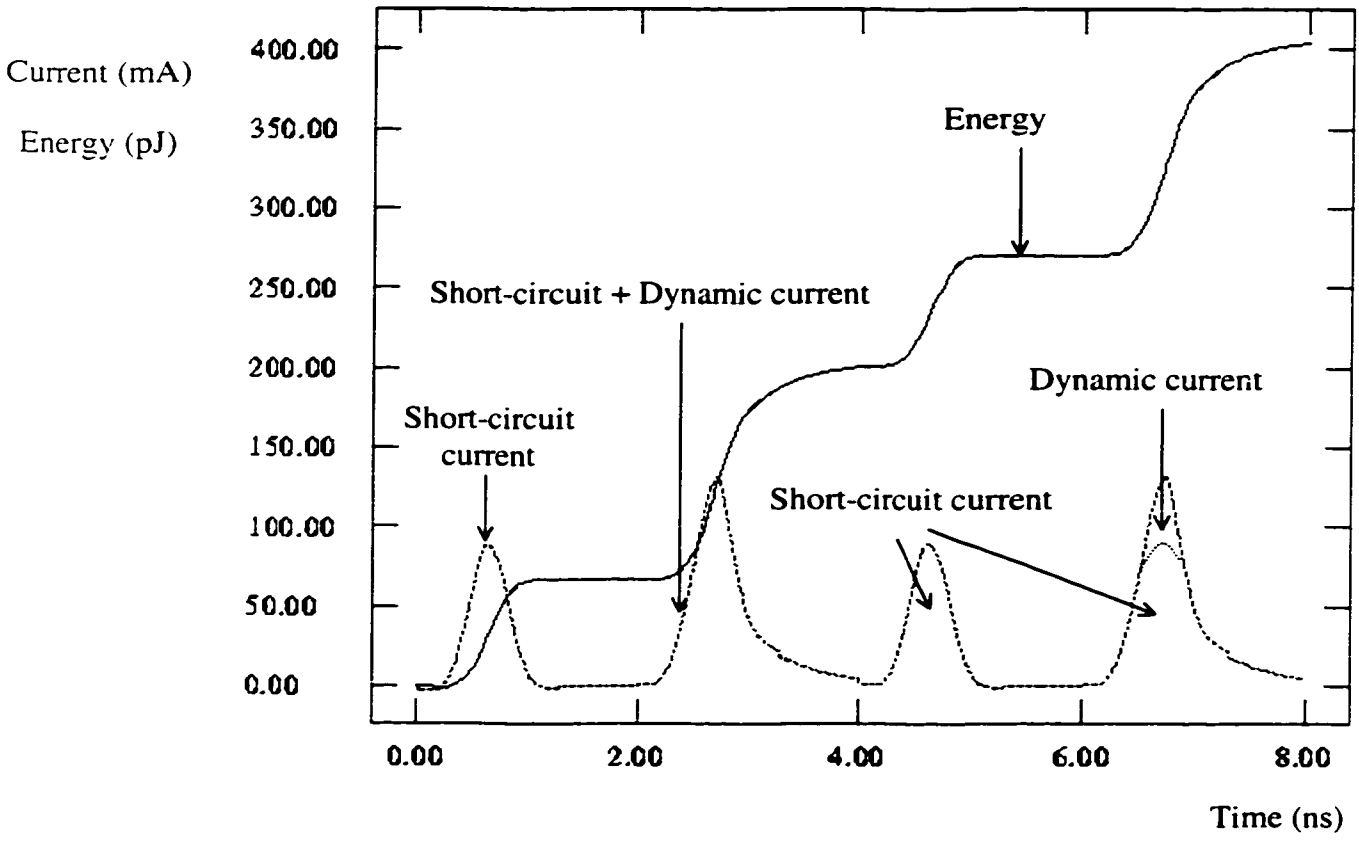


Figure 11.9. AS/X [128] simulations of the dynamic current, short-circuit current, and energy of the central buffer in the local clock distribution network depicted in Figure 11.7 with narrow wires.

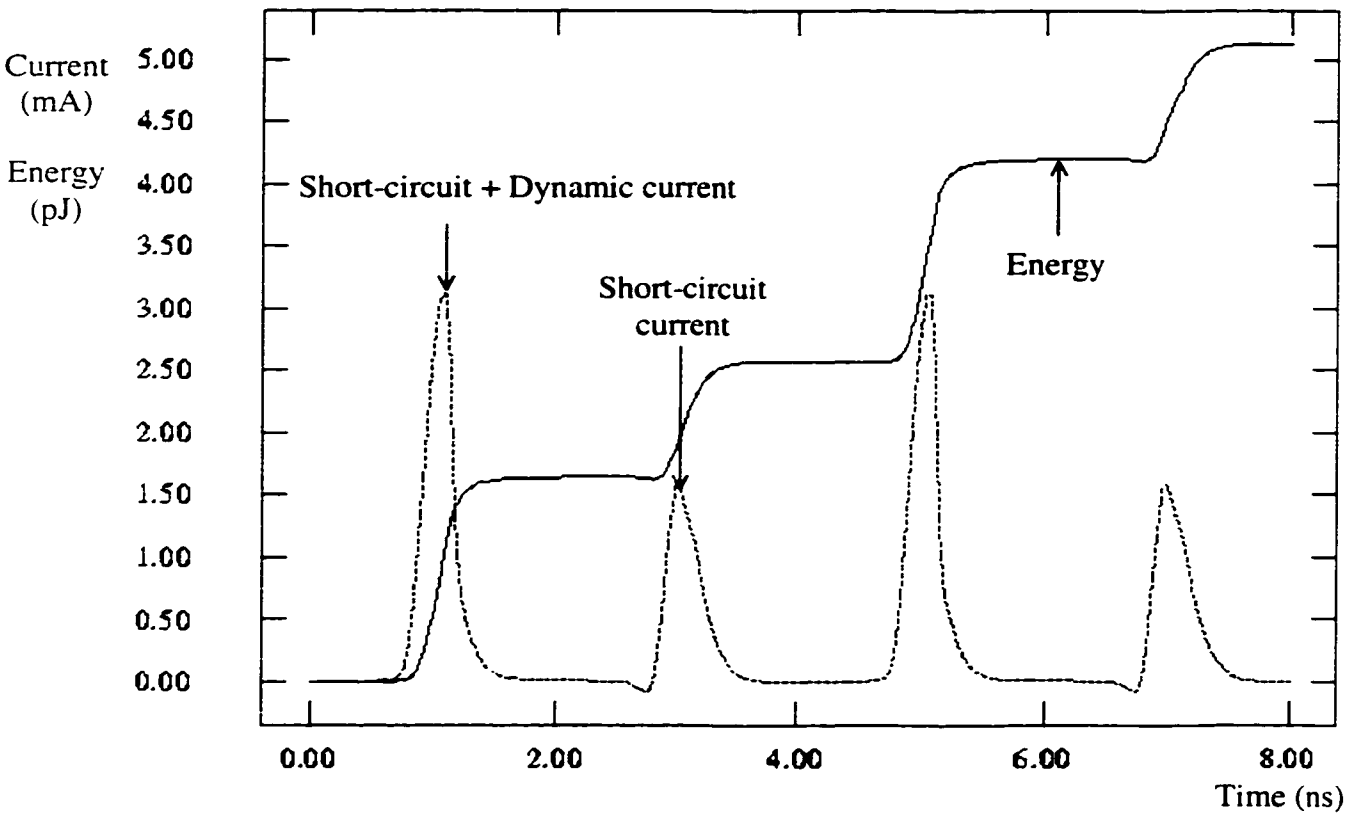


Figure 11.10. AS/X [128] simulations of the dynamic current, short-circuit current, and energy of one of the final buffers in the local clock distribution network depicted in Figure 11.7 with narrow wires.

To decrease the power dissipated by the final buffers, the clock distribution network is rerouted with wires twice as wide as the original wires. Simulations of the signals at the input of the central buffer and the final buffers are shown in Figure 11.11. The rise time of these signals is below 200 ps. The short-circuit and dynamic power of the central buffer and a single final buffer are shown in Figure 11.12 and Figure 11.13, respectively. Note that the dynamic power consumption of the central buffer has increased due to the increased capacitance of the wider wires driven by the central buffer. However, the faster input rise time has effectively eliminated the short-

circuit power, reducing the total power consumption of the central buffer. The short-circuit power consumed by the final buffers is also virtually eliminated while the dynamic power remains constant since the load of the final buffers has not changed. The power consumption of the redesigned clock distribution network is compared to the power consumption of the original clock distribution network design in Table 11.3.

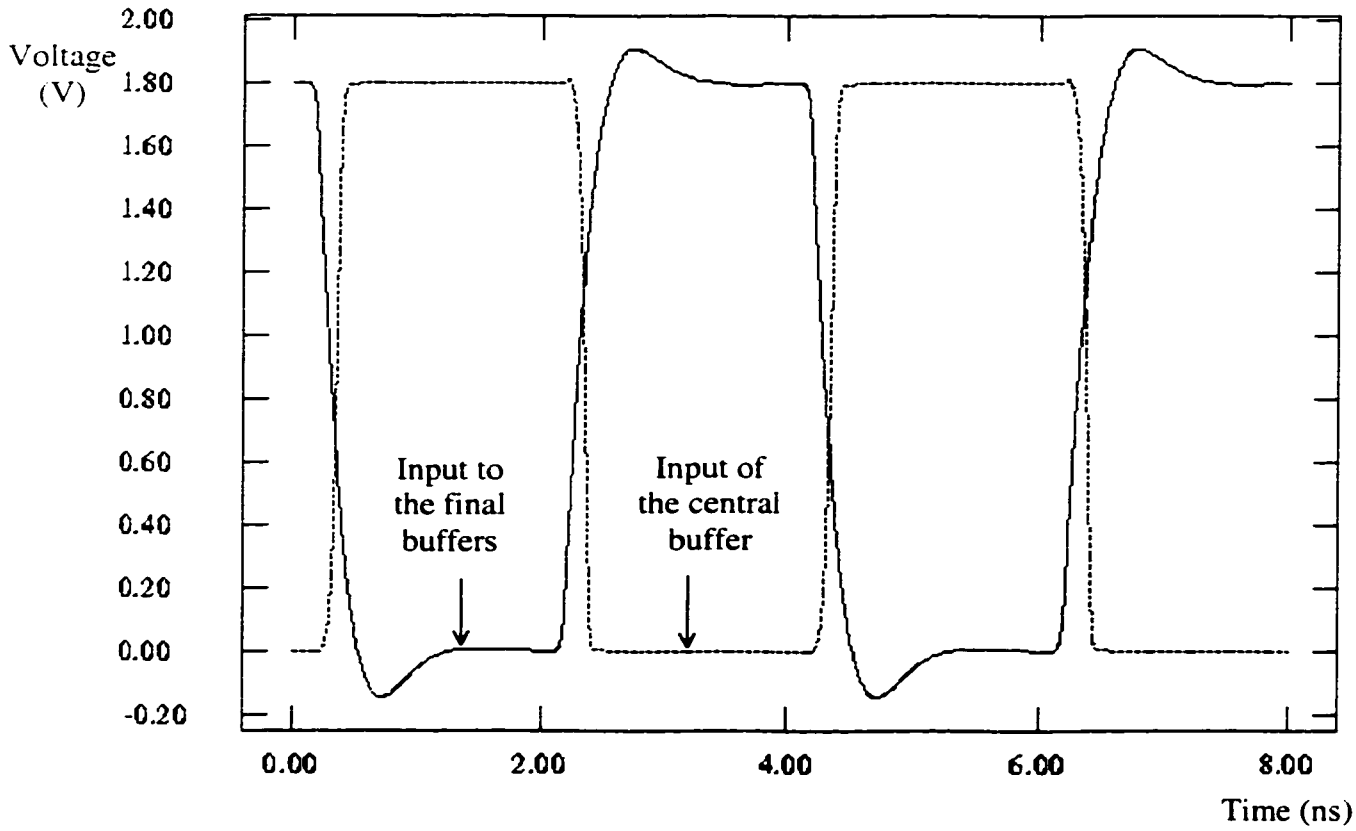


Figure 11.11. AS/X [128] simulations of the signals at the inputs of the central buffer and the final buffers in the local clock distribution network depicted in Figure 11.7 with wider wires.

Table 11.3 The power consumption of the central buffer, the final buffers, and the clock distribution network in Figure 11.7 when wider wires are used as compared to a narrow wire implementation.

Total power dissipation (pJ/cycle)	Old design (narrow wires)	New design (wider wires)	% power savings
Central buffer	201	137	31.8%
All final buffers	3469	1445	58.3%
Local clock distribution network	3670	1582	56.9%

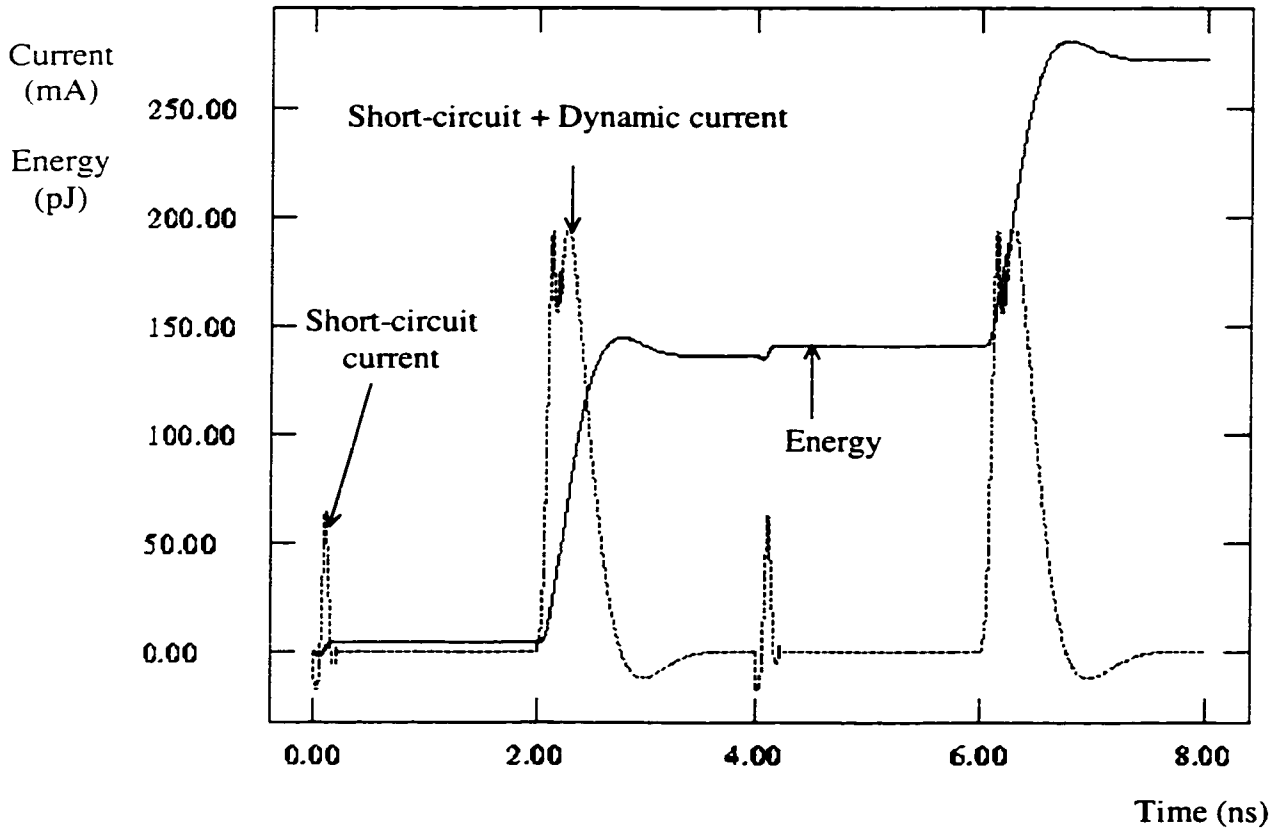


Figure 11.12. AS/X [128] simulations of the dynamic current, short-circuit current, and energy of the central buffer in the local clock distribution network depicted in Figure 11.7 with wider wires.

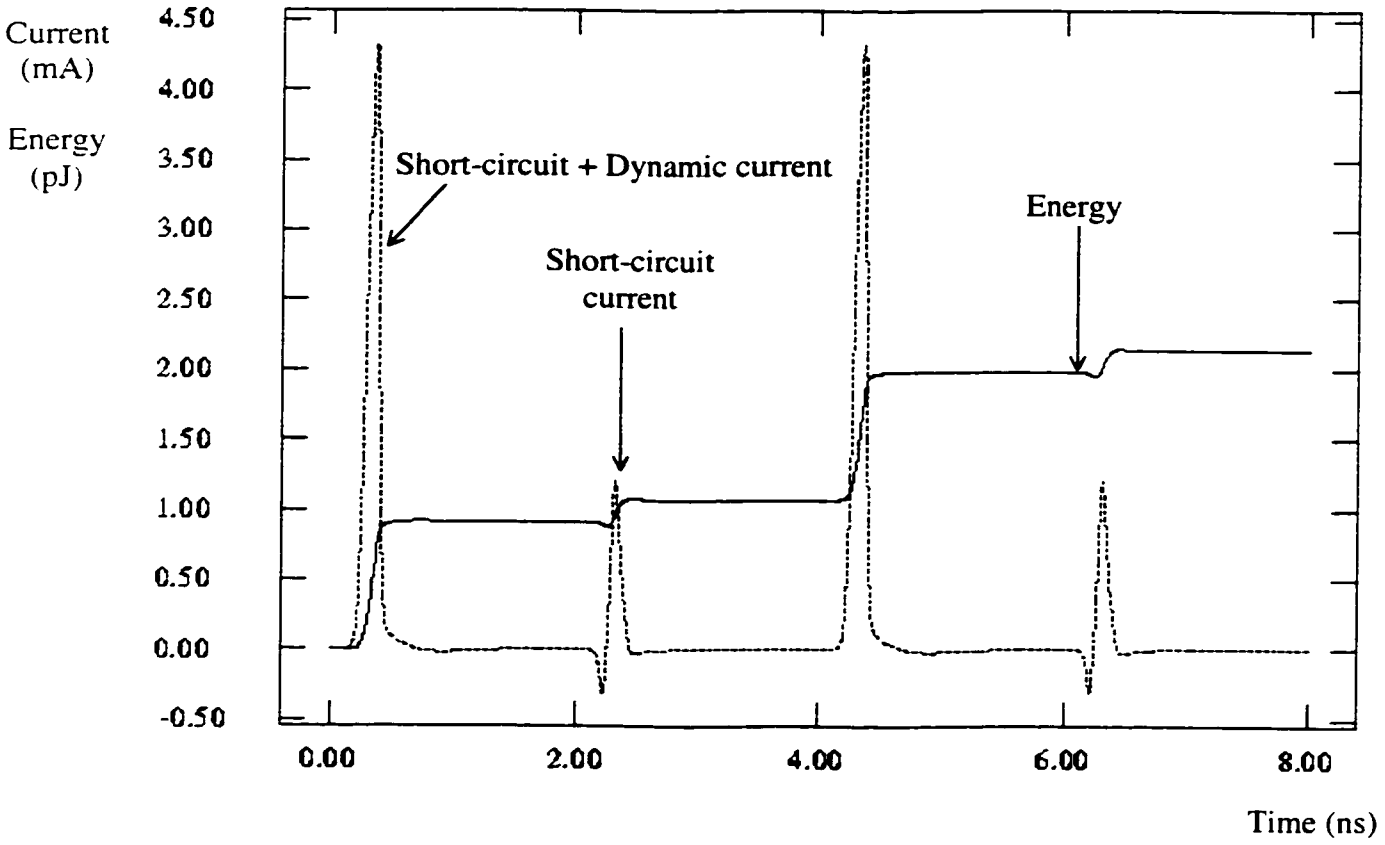


Figure 11.13. AS/X [128] simulations of the dynamic current, short-circuit current, and energy of one of the final buffers in the local clock distribution network depicted in Figure 11.7 with wider wires.

Note that the effects of inductance are now prominent with the use of wider wires in the clock distribution network, requiring that inductance be included in the interconnect model. This example illustrates that exposing inductance effects can improve the performance of an integrated circuit and that penalties in rise time, delay, and/or power consumption are incurred if these effects are eliminated. The overshoot that appears in the signal waveforms shown in Figure 11.13 does not cause any significant reliability problems. In certain cases, the increased power can be tolerated. However, if the original clock distribution network is intended to operate at 500

MHz, the signals at the inputs of the final buffers become problematic as shown in Figure 11.14. Note that the signals do not reach the required logic levels and the voltage swing is reduced, decreasing the noise margin. Such signals are unacceptable in a high performance integrated circuit. In this case, wider wires are not only necessary to reduce the power but to also maintain reliable operation of the integrated circuit. To achieve sufficiently fast signal rise times while maintaining reliable operation at high clock frequencies, wider drivers and wires should be used, resulting in greater inductance effects.

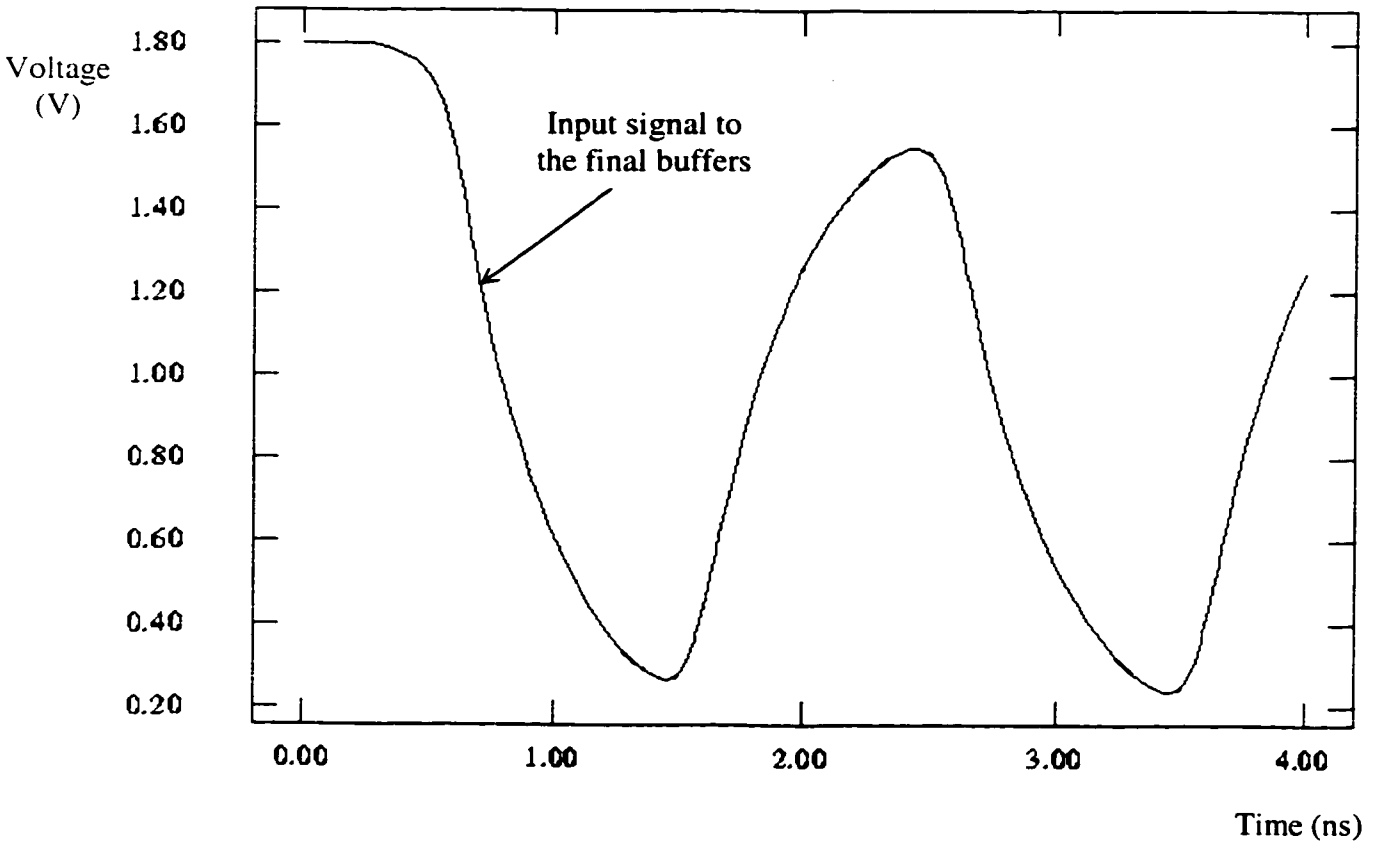


Figure 11.14. AS/X [128] simulations of the signal at the input of a final buffer in the local clock distribution network operating at 500 MHz as depicted in Figure 11.7 with narrow wires.

11.3 Summary

It is shown in this chapter that on-chip inductance can be exploited to improve the performance of high speed integrated circuits. Specifically, inductance improves the signal slew rate, dramatically reduces the short-circuit power consumption, and reduces the area of the active repeaters inserted to optimize the performance of long interconnects. These beneficial effects encourage design strategies that can exploit on-chip inductance. AS/X simulations of a clock distribution network have been presented to illustrate how inductance can be used to improve the performance of high speed integrated circuits. The power consumption of the clock distribution network decreases from 3670 pJ/cycle to 1582 pJ/cycle and the slew rate decreases from 1.2 ns to 200 ps on the internal nodes of the clock distribution network when wider, more inductive, wires are used.

Chapter 12 Accurate and Efficient Evaluation of the Transient Response in *RLC* Circuits: the DTT Method

Several popular methods for evaluating the transient response of VLSI circuits have been discussed in Chapter 3. The Elmore and Wyatt approximations for *RC* trees are discussed in 3.1. An equivalent Elmore delay model for *RLC* trees is presented in Chapter 7. These models are used as fast, low accuracy approximations of the delays within a VLSI circuit. The computational speed of these delay models makes these models appropriate for analyzing large VLSI circuits in a reasonable time in order to determine approximate delays within a circuit. The high fidelity of these delay models also makes these models appropriate for design methodologies where a solution is required. However, high accuracy characterization and simulation of the interconnect behavior and signal transients are required for analyzing performance critical modules and nets and to accurately anticipate possible hazards during switching. Also, increasing performance requirements has forced a reduction of the safety margins used in a worst case design, requiring more accurate interconnect delay characterization.

Dynamic simulators such as SPICE [127] or AS/X [128] can provide the required accuracy for the critical paths. However, these simulators are too slow for iterative design methodologies in current large VLSI circuits. Moment matching techniques such as AWE [75]-[80] have been discussed in section 3.2, having gained popularity as a more accurate delay model as compared to the Elmore delay model.

These moment matching techniques are also orders of magnitude faster than dynamic simulators [127], [128]. AWE is a well known algorithm that applies moment matching in which a set of low frequency dominant poles is determined that approximate the transient response at the nodes of an *RLC* tree. However, AWE suffers several problems and limitations as discussed in subsection 3.2.3. Specifically, AWE suffers two primary problems [77]-[79]. The first problem is that the AWE method can lead to an approximation with unstable poles even for low order approximations [77]-[79]. The second problem is that AWE becomes numerically unstable for higher order approximations which limits the order of the approximations determined using AWE to less than approximately eight poles (of which some poles may be unstable and are discarded) [77]-[79]. This limited number of poles is inappropriate for evaluating the transient responses of an underdamped *RLC* tree which requires a greater number of poles to accurately capture the transient response at all of the nodes. To overcome this limitation, a set of model order reduction algorithms have been developed to determine higher order approximations appropriate for *RLC* circuits based on the state space representation of an *RLC* network. Examples are Pade via Lanczos (PVL) [97], Matrix Pade via Lanczos (MPVL) [99], Arnoldi Algorithms [100], Block Arnoldi Algorithms [101], Passive Reduced-Order Interconnect Macromodeling Algorithm (PRIMA) [106], [107], and SyPVL Algorithm [136]. However, these model order reduction techniques have significantly higher computational complexity than AWE. The complexity of these techniques is super linear with n, where n is the order of the *RLC* tree and is equal to the total number of capacitors and inductors in the tree. This high complexity is due to these model order reduction techniques solving n linear equations in n variables

several times [97]-[136]. This complexity is much higher than the complexity of AWE which is linearly proportional to n for an *RLC* tree [77]-[79]. Note that n can be on the order of thousands for a typical large industrial *RLC* circuit.

The objective of this chapter is to introduce a new method [122] for evaluating the transient response at the nodes of a general *RLC* tree. This method is capable of determining high order approximations appropriate for underdamped *RLC* trees in a computationally efficient manner. The DTT (Direct Derivation of the Transfer function) method is introduced as an alternative to moment matching techniques to evaluate time domain signals within *RLC* trees with arbitrary accuracy in response to any input signal. This method depends on finding a low frequency reduced order transfer function by direct truncation of the exact transfer function at different nodes of an *RLC* tree. The method is numerically accurate for any order of approximation, permitting approximations to be determined with a large number of poles appropriate for approximating *RLC* trees with underdamped responses. The method is computationally efficient with a complexity linearly proportional to the number of branches in an *RLC* tree. A common set of poles is determined that characterize the responses at all of the nodes of an *RLC* tree which further enhances the computational efficiency. Stability is guaranteed by the DTT method for low order approximations with less than 5 poles. Such low order approximations are useful for evaluating monotone responses exhibited by *RC* circuits.

The rest of this chapter is organized as follows. A description of the DTT method is provided in section 12.1. In section 12.2, the complexity and stability characteristics of the DTT method are discussed. The transient responses based on the DTT method for several *RC* and *RLC* trees are compared to SPICE simulations in

section 12.3. Finally, some conclusions are offered in section 12.4. Pseudo-code describing the DTT method is provided in Appendix E. The DTT method is compared to AWE in Appendix F.

12.1 The DTT Method

The concepts used to develop the DTT method are explained in this section. The rules governing the poles and zeros in an *RLC* tree are defined in subsection 12.1.1. The method used to calculate the exact transfer functions at the nodes of an *RLC* tree is introduced in subsection 12.1.2. The use of transfer function truncation to determine a reduced order approximation is discussed in subsection 12.1.3. The process of determining the set of common poles describing the transient response of an *RLC* tree and the corresponding residues at each node of the tree is described in subsection 12.1.4.

12.1.1 Pole-Zero Behavior in *RLC* Trees

The poles and zeros of an *RLC* tree maintain specific relations to the poles and zeros of the subtrees forming the *RLC* tree. These rules are established in this subsection and are used in the following subsection to develop an algorithm to determine the poles and zeros of a general *RLC* tree by recursively subdividing the tree into smaller subtrees.

Rule 1: *The poles of an RLC circuit are zeros of the impedance seen at the input of the circuit.*

This rule can be understood by referring to Figure 12.1 and noting that the transfer functions describing the capacitor voltages and inductor currents have a common denominator (the characteristic equation of the tree) [129]-[132]. Thus, the transfer function at an arbitrary node *i* of an *RLC* tree and the input admittance of the tree are given by

$$\frac{V_i(s)}{V_{in}(s)} = \frac{N_i(s)}{D(s)}, \tag{12.1}$$

$$Y_{in}(s) = \frac{I_{in}(s)}{V_{in}(s)} = \frac{N_{I_{in}}(s)}{D(s)}, \tag{12.2}$$

respectively, where $N_i(s)$ and $N_{Iin}(s)$ are functions of s dependent on the circuit structure and $D(s)$ is the common denominator of the circuit. The input impedance is

$$Z_{in}(s) = \frac{V_{in}(s)}{I_{in}(s)} = \frac{D(s)}{N_{Iin}(s)}. \tag{12.3}$$

Thus, the common denominator of an *RLC* circuit is the numerator of the input impedance which proves rule 1.

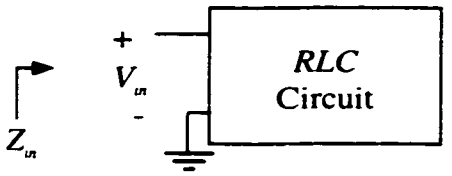


Figure 12.1. A general *RLC* circuit.

As an example, consider the single section *RLC* circuit shown in Figure 12.2. This circuit has a transfer function and an input impedance given by

$$\frac{V_{out}(s)}{V_{in}(s)} = \frac{1}{s^2LC + sRC + 1}, \tag{12.4}$$

$$\frac{V_{in}(s)}{I_{in}(s)} = sL + R + \frac{1}{sC} = \frac{s^2LC + sRC + 1}{sC}, \tag{12.5}$$

respectively. Note that the denominator of the transfer function is the numerator of the input impedance. Another way to interpret Rule 1 is that an *RLC* circuit has a short-circuit input impedance when s is equal to the poles of the circuit.

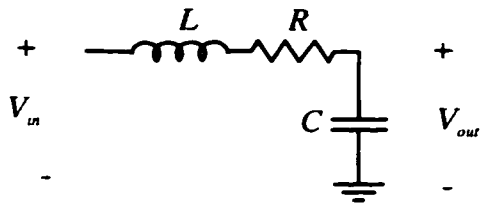


Figure 12.2. Simple *RLC* circuit.

Rule 2: *The poles of an RLC circuit driven at node x are zeros of the transfer function at node x.*

This rule can be explained by referring to Figure 12.3. Note that the *RLC* circuit 2 is driven by the *RLC* circuit 1 at node x. Applying rule 1, Z_{in2} is a short-circuit between node x and the ground at frequencies equal to the poles of circuit 2. Hence, $V_x(s)$ is equal to zero when s is equal to the poles of circuit 2, *i.e.*, the poles of circuit 2 are zeros of the transfer function at node x.

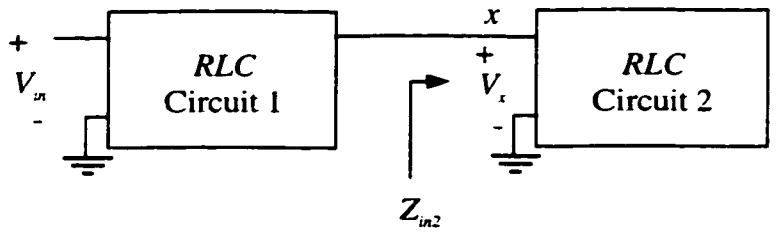


Figure 12.3. A general *RLC* circuit composed of two *RLC* subcircuits connected together.

As an example, consider the circuit shown in Figure 12.4. Note that the *RLC* subcircuit 2 is driven at node x and that if not connected, subcircuit 2 has a denominator given by $1 + R_2C_2s + L_2C_2s^2$. The transfer functions at node x and the output node are

$$\frac{V_x(s)}{V_{in}(s)} = \frac{1 + R_2C_2s + L_2C_2s^2}{1 + [R_1(C_1 + C_2) + R_2C_2]s + [L_1(C_1 + C_2) + L_2C_2 + R_1C_1R_2C_2]s^2 + [R_1C_1L_2C_2 + R_2C_2L_1C_1]s^3 + [L_1C_1L_2C_2]s^4},$$

$$\frac{V_{out}(s)}{V_{in}(s)} = \frac{1}{1 + [R_1(C_1 + C_2) + R_2C_2]s + [L_1(C_1 + C_2) + L_2C_2 + R_1C_1R_2C_2]s^2 + [R_1C_1L_2C_2 + R_2C_2L_1C_1]s^3 + [L_1C_1L_2C_2]s^4},$$

respectively. Note that the numerator at node x is the same as the denominator of the disconnected subcircuit 2 in accordance with rule 2.

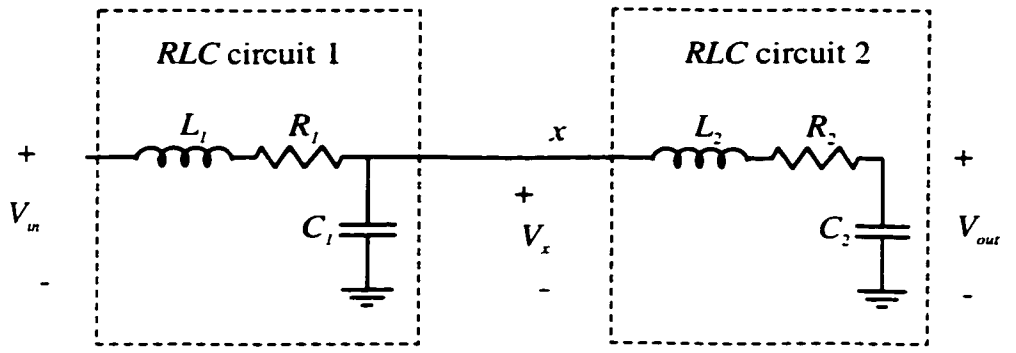


Figure 12.4. A ladder *RLC* circuit composed of two *RLC* sections in series.

Rule 3: *The poles of an RLC circuit driven at node x are zeros of the transfer functions at all of the nodes of parallel RLC circuits driven at the same node x.*

This rule can be explained by referring to Figure 12.5. The *RLC* subcircuits 2, 3, ..., k are driven by *RLC* subcircuit 1 at node x. Applying rule 1, Z_{in2} is a short-circuit at frequencies equal to the poles of circuit 2. Hence, $V_x(s)$ is equal to zero and

all of the current supplied by circuit 1 is sunk to ground by Z_{in2} when s is equal to the poles of circuit 2. Since $V_x(s)$ is equal to zero and no current is supplied to the subcircuits 3, ..., k when s is equal to the poles of circuit 2, the voltages at all of the nodes of subcircuits 3, ..., k are equal to zero. Alternatively, the poles of circuit 2 are zeros of the transfer functions at all of the nodes of the parallel subcircuits driven at node x. The same is true for the poles of subcircuits 3, ..., k which are zeros of the transfer functions at all of the nodes of the parallel subcircuits driven at node x.

As an example, consider the *RLC* tree shown in Figure 12.6. The *RLC* section 1 drives the two parallel *RLC* sections 2 and 3. The transfer functions at nodes x, 2, and 3 are given by

$$\frac{V_x(s)}{V_{in}(s)} = \frac{(1 + R_2C_2s + L_2C_2s^2)(1 + R_3C_3s + L_3C_3s^2)}{D}, \tag{12.6}$$

$$\frac{V_2(s)}{V_{in}(s)} = \frac{(1 + R_3C_3s + L_3C_3s^2)}{D}, \tag{12.7}$$

$$\frac{V_3(s)}{V_{in}(s)} = \frac{(1 + R_2C_2s + L_2C_2s^2)}{D}, \tag{12.8}$$

respectively, where D is the common denominator and is a polynomial in s of order six. The specific form of D is not of interest here. The denominators of subcircuits 2 and 3 are $1 + R_2C_2s + L_2C_2s^2$ and $1 + R_3C_3s + L_3C_3s^2$, respectively. Note that both denominators are multiplied in the numerator of the transfer function at node x showing that the poles of subcircuits 2 and 3 are zeros of the transfer function at the driving node x in accordance with rule 2. Note also that the poles of subcircuit 2 are zeros of the transfer function at node 3 and vice versa, which verifies rule 3.

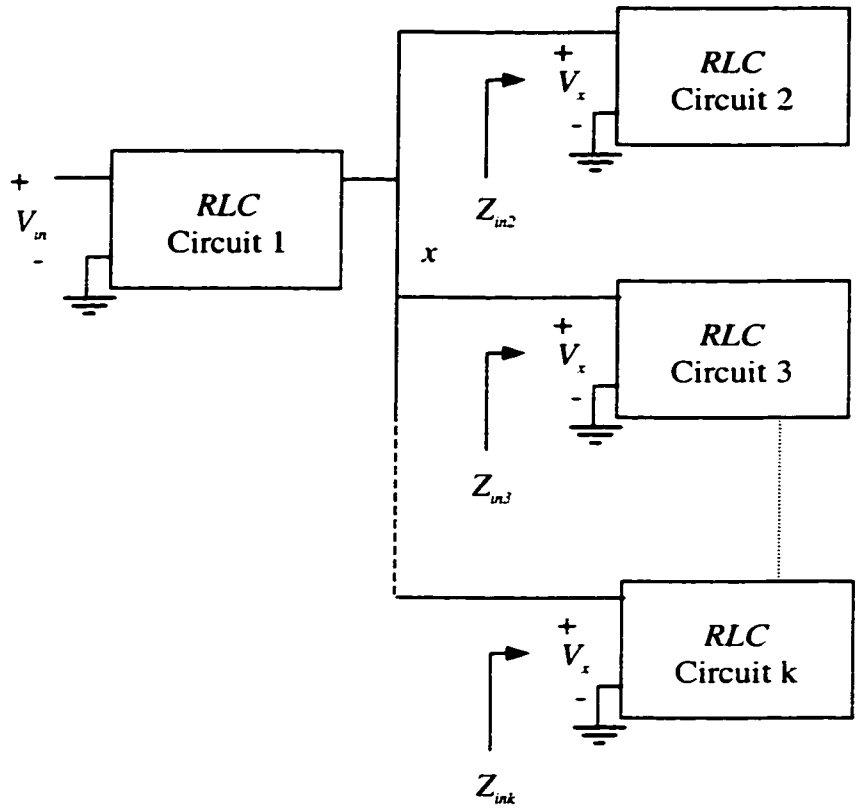


Figure 12.5. A general *RLC* circuit composed of an *RLC* subcircuit driving several subcircuits connected in parallel.

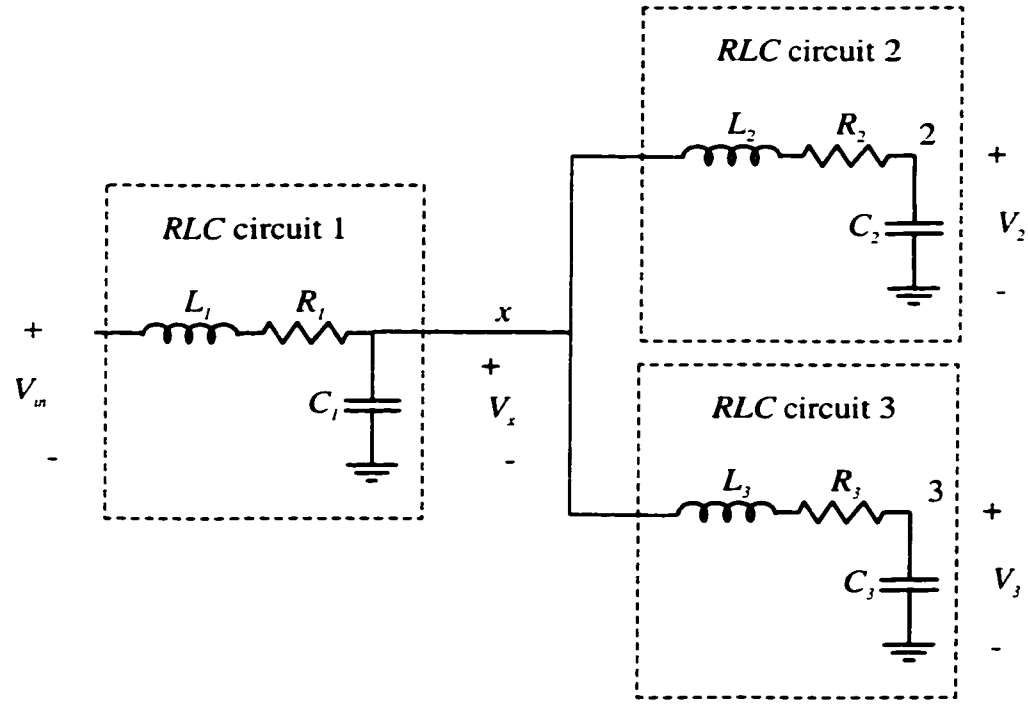


Figure 12.6. An *RLC* tree composed of three *RLC* sections.

12.1.2 Calculating the Transfer Functions at the Nodes of an *RLC* Tree

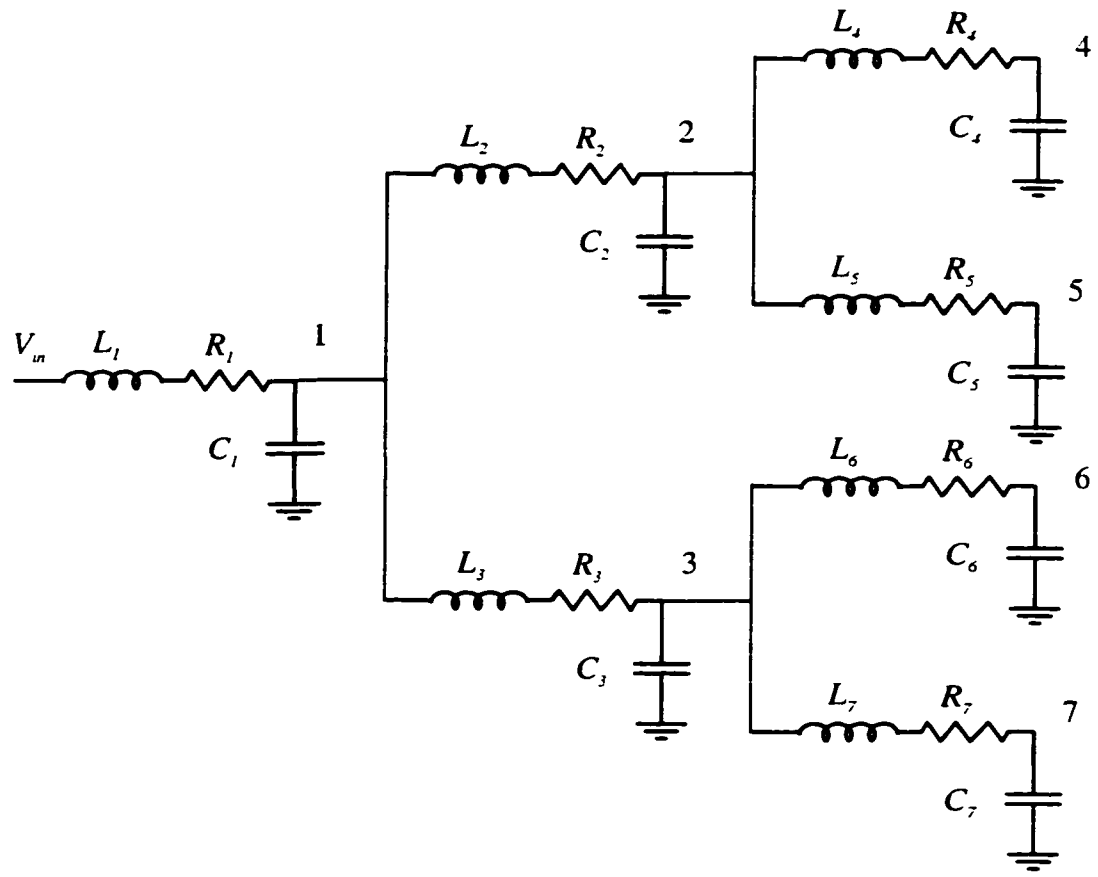


Figure 12.7. General *RLC* tree.

It is illustrated in this subsection how to recursively calculate the transfer functions at the nodes of an *RLC* tree using the concepts developed in the previous subsection. Consider the general *RLC* tree shown in Figure 12.7. The current sunk to ground by a capacitor k is given by $C_k dv_k(t)/dt$ where $v_k(t)$ is the voltage across C_k. Thus, the current passing through the resistance R_1 and the inductance L_1 is given by

$$i_1(t) = \sum_k C_k \frac{dv_k(t)}{dt}, \tag{12.9}$$

where the summation index k operates over all of the capacitors in the tree. The voltage drop across R_1 and L_1 is given by

$$v_{in}(t) - v_1(t) = R_1 i_1(t) + L_1 \frac{di_1(t)}{dt} = R_1 \sum_k C_k \frac{dv_k(t)}{dt} + L_1 \sum_k C_k \frac{d^2 v_k(t)}{dt^2}. \qquad (12.10)$$

In the frequency domain, this relation transforms to

$$V_{in}(s) - V_1(s) = (sR_1 + s^2 L_1) \sum_k C_k V_k(s). \qquad (12.11)$$

Dividing (12.11) by $V_{in}(s)$, the following relation results,

$$1 - T_1(s) = (sR_1 + s^2 L_1) \sum_k C_k T_k(s), \qquad (12.12)$$

where $T_i(s)$ is the transfer function at node 1 and $T_k(s)$ is the transfer function at node k. Note that determining the transfer function at node 1 is sufficient to determine the poles of the entire circuit since the transfer functions at all of the nodes of an *RLC* tree have a common denominator (as was mentioned previously).

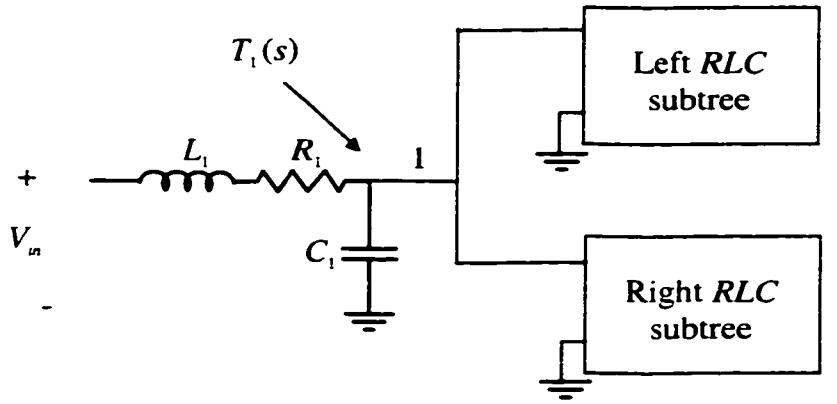


Figure 12.8. Building block of a general *RLC* tree.

Now consider the structure shown in Figure 12.8 which depicts an *RLC* section driving left and right subtrees. Without loss of generality, a binary branching factor is used here since a general tree with an arbitrary branching factor can be transformed into a binary tree by inserting zero impedance branches [37], [43]. The structure shown in Figure 12.8 can be used recursively to fully represent any *RLC* tree

since the left and right subtrees can in turn be represented by the same structure. The transfer function at node 1 of Figure 12.8 is given by (12.12), which can be reformulated by using the rational representations of the transfer functions, $T_1(s)=N_1(s)/D(s)$ and $T_k(s)=N_k(s)/D(s)$, and is

$$D(s) - N_1(s) = (sR_1 + s^2 L_1)\sum_k C_k N_k(s). \tag{12.13}$$

Assume that the transfer functions at all of the nodes of the left and right *RLC* subtrees (when the trees are disconnected) are known and are given by $T_{lk1}(s)=N_{lk1}(s)/D_l(s)$ at node k_1 of the left subtree and $T_{rk2}(s)=N_{rk2}(s)/D_r(s)$ at node k_2 of the right subtree. The numerator at node 1, $N_1(s)$ of Figure 12.8, can be directly calculated by applying rule 2 described in the previous subsection and is

$$N_1(s) = D_l(s) \bullet D_r(s). \tag{12.14}$$

The "•" operator above represents a polynomial multiplication. The denominator $D(s)$ can be determined from (12.13) as

$$D(s) = N_1(s) + (sR_1 + s^2 L_1)M_1, \tag{12.15}$$

where M_1 is defined as

$$M_1 = \sum_k C_k N_k(s), \tag{12.16}$$

and characterizes the summation of the numerators of the transfer functions across the capacitors in the tree multiplied by the corresponding capacitances. The summation in M_1 operates over all of the capacitors in the tree and can be divided into three components.

$$M_1 = C_1 N_1(s) + \sum_{k1} C_{k1} N_{k1}(s) + \sum_{k2} C_{k2} N_{k2}(s), \tag{12.17}$$

where k_1 covers the capacitors in the left subtree and k_2 covers the capacitors in the right subtree. By applying rule 3, the numerators in the left subtree can be described in terms of the parameters of the disconnected left and right subtrees as $N_{k1}(s)=N_{lk1}(s)\bullet D_r(s)$. Similarly, $N_{k2}(s)=N_{rk2}(s)\bullet D_l(s)$. Thus, (12.17) can be reconfigured as

$$M_1 = C_1 N_1(s) + \left(\sum_{k1} C_{k1} N_{lk1}(s) \right) \bullet D_r(s) + \left(\sum_{k2} C_{k2} N_{rk2}(s) \right) \bullet D_l(s). \tag{12.18}$$

Note that the two summations above are M_l and M_r of the disconnected left and right subtrees, respectively. Hence, M_1 can be fully calculated in terms of the disconnected left and right subtree parameters as

$$M_1 = C_1 N_1(s) + M_l(s) \bullet D_r(s) + M_r(s) \bullet D_l(s). \tag{12.19}$$

Thus, by knowing the parameters of the left and right subtrees, $M_l(s)$, $D_l(s)$, $M_r(s)$, and $D_r(s)$, (12.14), (12.19), and (12.15) can be used in that order to determine $N_1(s)$, $M_1(s)$, and $D(s)$, respectively. The parameters of the left and right subtrees, $M_l(s)$, $D_l(s)$, $M_r(s)$, and $D_r(s)$, can be determined in turn in terms of their left and right subtrees by using the structure shown in Figure 12.8 and (12.14), (12.19), and (12.15). This process is repeated recursively until the left and right subtrees are non-existent. If the left subtree does not exist, then $M_l(s) = 0$ and $D_l(s) = 1$. If the right subtree does not exist, then $M_r(s) = 0$ and $D_r(s) = 1$.

After this recursion process terminates, the denominator and numerator across each capacitance C_k in the tree represent the transfer function for the subtree rooted at the *RLC* section k. For example, for the tree shown in Figure 12.7, D(s) and N(s) at node 1 represent the transfer function at node 1 for the entire tree. However, D(s) and N(s) at node 2 represent the transfer function at node 2 for the subtree composed of

the *RLC* sections, 2, 4, and 5. Also, D(s) and N(s) at node 4 represent the transfer function at node 4 for the subtree composed of *RLC* section 4. Thus, after the recursion process terminates, the only relevant parameters for the entire *RLC* tree are D(s) and N(s) across the capacitor closest to the input (C_1 in the case of the tree shown in Figure 12.7). The denominators and numerators at all of the other nodes are incorrect. The denominators at these nodes need not be corrected since these denominators are the same as the denominator at the node closest to the input. However, the numerators differ at each node and need to be corrected. According to rule 3, all of the numerators in the left subtree have to be multiplied by $D_r(s)$ and all of the numerators in the right subtree have to be multiplied by $D_l(s)$. This process is repeated recursively starting at the root of the tree and advancing towards the sinks.

Thus, the process of determining the transfer function at all of the nodes of an *RLC* tree consists of two steps. The first step is to calculate the common denominator of the *RLC* tree and is accomplished by the function Cal_Denominator presented as pseudo-code in Appendix E which uses the recursive equations in (12.14), (12.19), and (12.15). The common denominator is the denominator at the node closest to the input of the *RLC* tree after the recursion terminates. The second step is to correct the numerators of the transfer functions at the nodes of the *RLC* tree. This task is achieved by the function Correct_Numerators which is also described as pseudo-code in Appendix E.

12.1.3 Transfer Function Truncation and Approximation Order

The process of calculating the exact transfer functions at all of the nodes of an *RLC* tree has been described in the previous subsection. However, calculating the

exact transfer function can be time consuming since n can be in the order of thousands for typical large industrial *RLC* trees. In practice, there is no need to calculate the thousands of poles characterizing an *RLC* tree since the transient behavior can be accurately characterized by a few number of low frequency dominant poles [75]-[80] (typically several tens of poles). Thus, a low frequency approximation is required that can correctly anticipate the set of dominant poles without calculating the exact high order transfer function.

Assume that the exact transfer function at a specific node of the *RLC* tree is given by

$$T(s) = \frac{1 + a_1 s + a_2 s^2 + \ldots + a_m s^m}{1 + b_1 s + b_2 s^2 + \ldots + b_n s^n}, \tag{12.20}$$

where $b_1 - b_n$ and $a_1 - a_m$ are positive real constants. The system order n is equal to the total number of capacitors and inductors in the tree. The order of the numerator polynomial m is less than n and is dependent on the node at which the transfer function is calculated. A q^{th} order approximate transfer function is found by direct truncation of the exact transfer function $T(s)$ in (12.20) and is given by

$$T_q(s) = \frac{1 + a_1 s + a_2 s^2 + \ldots + a_x s^x}{1 + b_1 s + b_2 s^2 + \ldots + b_q s^q}, \tag{12.21}$$

where $q < n$. The numerator order $x = m$ if $m \leq q$ - 1, otherwise $x = q$ - 1. The order of the numerator has to be less than the order of the denominator for a causal approximation. If s (or the frequency) is sufficiently small, the terms with higher power of s in the denominator and numerator polynomials ($b_{q+1}s^{q+1} - b_n s^n$, $a_{x+1}s^{x+1} - a_m s^m$) are negligible with respect to the lower power terms in $T_q(s)$. Thus, for low

frequencies, $T_q(s)$ is an accurate representation of $T(s)$. The range of frequencies for which $T_q(s)$ is accurate increases as q increases.

The calculation of a q^{th} order approximation for the transfer functions at all of the nodes of an *RLC* tree can be accomplished by an order limited polynomial multiplication. To better understand this concept, assume that A and B are two polynomials of orders n_a and n_b, respectively. The polynomial C given by $A \bullet B$ has an order of $n_c = n_a + n_b$. The polynomials A, B, and C are given by

$$A = \sum_{i=0}^{n_a} a_i s^i, \tag{12.22}$$

$$B = \sum_{i=0}^{n_b} b_i s^i, \tag{12.23}$$

$$C = \sum_{i=0}^{n_c} c_i s^i, \tag{12.24}$$

respectively, where the coefficients c_i are

$$c_i = \sum_{j=0}^{n_a} a_j b_{i-j}. \tag{12.25}$$

Note that b_{i-j} is equal to zero if i - j is out of the range of 0 to n_b. For a q limited polynomial multiplication, the highest desired power of s in C is q rather than n_c and the coefficients of higher powers of s do not need to be calculated. Also, A and B can be limited by q since higher powers than s^q in both polynomials cannot produce powers of s in C less than or equal to q. Hence, if a q^{th} order approximation is sought, all of the polynomial multiplications of the DTT method described in the previous subsection are q limited. These q limited polynomial multiplications are much less expensive than full polynomial multiplications since q is typically much less than n. The number of scalar multiplications required for a q limited polynomial

multiplication is at most $q(q+1)/2$ when the polynomial orders, n_a and n_b, are equal to q. As is explained in section 12.2, the actual number of scalar multiplications performed by the DTT method is much less than the number of multiplications anticipated using the $q(q+1)/2$ complexity of a polynomial multiplication.

12.1.4 Determining the Poles, Residues, and the Transient Response

Once the common denominator of order q, $D_q(s)$, is determined as described in the previous subsections, the first q dominant low frequency poles of the *RLC* tree can be calculated as the roots of the polynomial $D_q(s)$. A numerical method for evaluating the roots of a polynomial can be used to determine the *RLC* tree poles, p_1 – p_q, *e.g.*, [137], [138]. The residues corresponding to each pole at a specific node can be efficiently calculated by direct substitution of the poles into the numerator of the transfer function at this node. The residues corresponding to the pole p_i at node j of an *RLC* tree can be calculated as

$$k_i^j = \frac{N_j(s = p_i)}{DP_i}, \tag{12.26}$$

where

$$DP_i = b_q \prod_{\substack{r=1 \\ r \neq i}}^{q} (p_i - p_r), \tag{12.27}$$

where b_q is the coefficient of s^q in $D_q(s)$. Note that DP_i is independent of the node at which the residues are evaluated. Thus, DP_i can be evaluated once and used to calculate the residues at any number of nodes, which reduces the computational complexity when the transient response is required at many nodes.

The poles of the circuit and the corresponding residues at node j of an *RLC* tree can be used to characterize the transfer function at node j as

$$T_j(s) = \sum_{i=1}^{q} \frac{k_i^j}{(s - p_i)}. \tag{12.28}$$

This transfer function can be used to calculate the time domain response at node j for an arbitrary input by multiplying the Laplace transform of the input by $T_j(s)$ and calculating the inverse Laplace transform of the resulting expression. For example, for a unit step input, the output response at node j, $e_j(t)$ is

$$e_j(t) = 1 + \sum_{i=1}^{q} \left[\frac{k_i^j}{p_i} e^{p_i t} \right]. \tag{12.29}$$

For an exponential input of the form,

$$v_{in}(t) = 1 - e^{-t/\tau}, \tag{12.30}$$

the transient response at node j is given by

$$e_j(t) = 1 + e^{-t/\tau} \left[\sum_{i=1}^{q} \frac{k_i^j \tau}{p_i \tau + 1} \right] + \sum_{i=1}^{q} \left[\frac{k_i^j}{p_i} \frac{1}{p_i \tau + 1} e^{p_i t} \right], \tag{12.31}$$

where τ is the time constant of the input signal. Some of the poles determined using the DTT method can be unstable due to the truncation of the denominator polynomial as discussed in the following section. These unstable poles can be simply discarded from the summations in (12.29) and (12.31). However, all of the poles should be included when calculating the residues using (12.26) and (12.27).

12.2 Complexity and Stability of the DTT Method

The DTT method has a complexity linearly proportional to the order of the tree n, which is twice the number of *RLC* sections in the tree since each *RLC* section has one capacitor and one inductor. This linear complexity occurs because the DTT method traverses each section in the tree only once as illustrated in the previous section and in Appendix E. At each section of the *RLC* tree, polynomial multiplications are required to calculate the common denominator as given by (12.14), (12.19), and (12.15). Although polynomial multiplication has an apparent complexity proportional to q^2 for a q^{th} order approximation, the average number of scalar multiplications required per section is much lower than q^2 for any *RLC* tree. To better explain this argument, consider the following cases. A node of an *RLC* tree with the right subtree nonexistent has $M_r = 0$ and $D_r = 1$. Thus, (12.14), (12.19), and (12.15) become

$$N_1(s) = D_l(s), \tag{12.32}$$

$$M_1 = C_1 N_1(s) + M_l(s), \tag{12.33}$$

$$D(s) = N_1(s) + (sR_1 + s^2 L_1) M_1, \tag{12.34}$$

respectively. Note that the DTT method has no polynomial multiplication at a node of a tree driving only one branch. The DTT method is therefore particularly efficient for single lines and in those cases where branches of a tree can be subdivided into several series *RLC* sections to model the distributed nature of the interconnect impedance.

A binary tree (such as the tree illustrated in Figure 12.7) with a total of r branches has $r/2$ leaves. These $r/2$ leaves are driven by $r/4$ branches, which are in turn driven by $r/8$ branches and so on. Determining $N(s)$, $M(s)$, and $D(s)$ at the $r/2$ leaves

requires only two scalar multiplications independent of the desired approximation order since for leaf i, $N(s) = 1$, $M(s) = C_i$, and $D(s) = 1 + R_iC_is + L_iC_is^2$. Applying these values at the next level with $r/4$ branches, the number of scalar multiplications required to determine $N(s)$, $M(s)$, and $D(s)$ is ten multiplications for a fourth order approximation or higher. Thus, for a binary tree, the average number of scalar multiplications required by the DTT method is much less than q^2 multiplications per polynomial multiplication. For example, calculating a fourth order approximation at all of the nodes of a binary tree requires a total number of scalar multiplications, SM, given by

$$SM_4 = 2 \cdot \frac{r}{2} + 10 \cdot \frac{r}{4} + 25 \cdot \frac{r}{4} = 9.75r \,. \tag{12.35}$$

Thus, the average number of scalar multiplications per branch of the tree is 9.75. The number of scalar multiplications calculated based on the q^2 polynomial multiplication complexity is $62r$ which greatly overestimates the complexity. The overestimation is even worse for higher values of q. For $q = 60$, the actual number of scalar multiplications is $160r$ multiplications while the q^2 model would predict $11000r$ multiplications. As the branching factor of an *RLC* tree increases, the overestimation by the q^2 model increases. This trend occurs because the leaves of the tree (which require only two scalar multiplications) constitute a larger fraction of the total number of branches with higher branching factors. For example, a tree with a branching factor of ten has almost 9/10 of its branches as leaves. For a general tree with a random branching factor at each node, the average number of scalar multiplications per node is much less than the q^2 model.

The above analysis demonstrates that the complexity of calculating the transfer functions at all of the nodes of an *RLC* tree is almost linear with the desired

order of approximation, q. This feature greatly decreases the expense of calculating higher order approximations. Also, the method depends on simple polynomial multiplications, which are numerically accurate for very high orders of approximation [139]-[141].

An analysis of the stability of the approximations calculated using the DTT method shows that a DTT approximation with an order less than five is guaranteed to be stable. Assume that the exact common denominator of an *RLC* tree is given by

$$D(s) = 1 + b_1 s + b_2 s^2 + \ldots + b_n s^n . \tag{12.36}$$

The common denominator of a q^{th} order approximation is therefore given by

$$D_q(s) = 1 + b_1 s + b_2 s^2 + \ldots + b_q s^q . \tag{12.37}$$

For a second order approximation, the condition for stability is that b_1 and b_2 are positive [135]. Since b_1 and b_2 are the coefficients of s and s^2 in the exact common denominator $D(s)$, b_1 and b_2 are guaranteed to be positive. This behavior occurs because a passive *RLC* tree is guaranteed to be stable [129]-[132] and stability requires that all of the coefficients of s in the denominator are positive. Therefore, a second order approximation is always stable. For a third order approximation, the Routh-Hurwitz criterion for stability [135] requires that $b_1 b_2 > b_3$. The coefficients b_1, b_2, and b_3 are given by

$$b_1 = -\sum_{i=1}^{n} \frac{1}{p_i}, \tag{12.38}$$

$$b_2 = \sum_{j=1}^{n} \sum_{k=j+1}^{n} \frac{1}{p_j p_k}, \tag{12.39}$$

$$b_3 = -\sum_{i=1}^{n} \sum_{j=i+1}^{n} \sum_{k=j+1}^{n} \frac{1}{p_i p_j p_k}, \tag{12.40}$$

respectively, where $p_1, p_2, \ldots, p_n$ are the poles of the exact common denominator and have negative real parts due to the stability of a passive *RLC* circuit. Thus, the quantity $b_1b_2 - b_3$ is given by

$$b_1b_2 - b_3 = -\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=j+1}^{n}\frac{1}{p_i p_j p_k} + \sum_{i=1}^{n}\sum_{j=i+1}^{n}\sum_{k=j+1}^{n}\frac{1}{p_i p_j p_k} = -\sum_{i=1}^{n}\sum_{j=1}^{i}\sum_{k=j+1}^{n}\frac{1}{p_i p_j p_k}. \quad (12.41)$$

Note that the quantity $b_1b_2 - b_3$ is positive since $p_1, p_2, \ldots, p_n$ have negative real parts. Thus, a third order approximation is also guaranteed to be stable. The same procedure can be repeated for a fourth order system. It can be shown that stability is also guaranteed for a fourth order system. These low order approximations are useful for *RC* trees since the signals within an *RC* tree can typically be approximated with a few dominant poles due to the monotone nature of the response. Approximations of order five or higher are not guaranteed to be stable. However, since the DTT method is numerically stable for any order of approximation and since the computational complexity increases slowly with the approximation order, high order approximations can always be determined using the DTT method to correctly detect all of the poles in the frequency range of interest.

12.3 Experimental Results

The DTT method is applied in this section to calculate the transient response of several *RC* and *RLC* trees. The resulting transient responses are compared to SPICE simulations to evaluate the accuracy of the DTT method. The DTT method is applied first to evaluate the transient response of the *RC* circuit shown in Figure 12.9. The circuit is composed of a distributed *RC* transmission line driven by a lumped resistance R_{tr} (which represents the output impedance of the driving gate) and a load

capacitance C_L (which represents the input capacitance of the driven gate). The line has a total resistance of R_t and a total capacitance of C_t. The transient response based on the DTT method with approximation orders of two, three, and four are compared to SPICE in Figure 12.10. Note that a second order approximation has a negligible error in the transient response as compared to SPICE and that the third and fourth order approximations are practically exact.

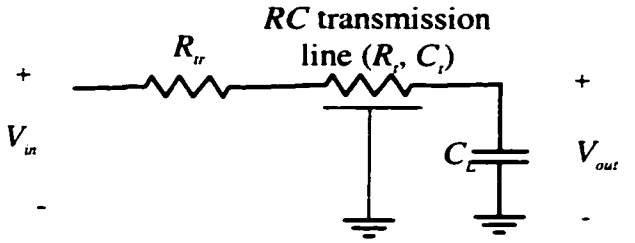
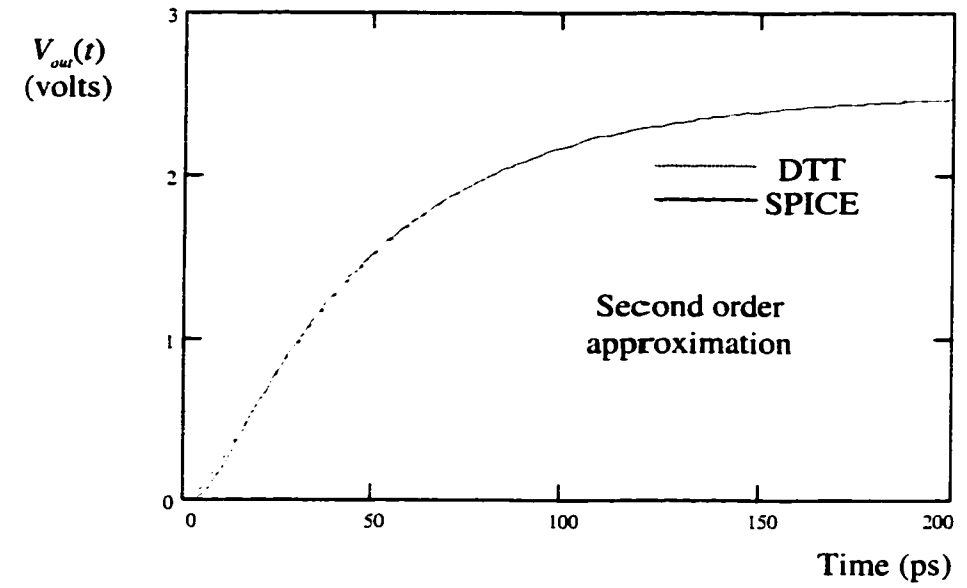
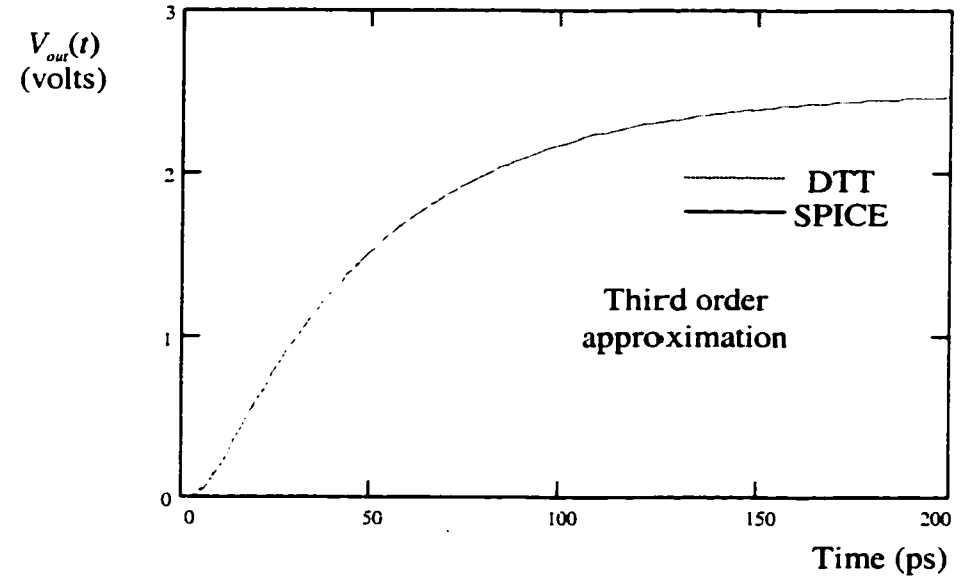


Figure 12.9. An *RC* transmission line with a source resistance and a load capacitance.


$V_{out}(t)$
(volts)
3
2
1
0
0
50
100
150
200
Time (ps)
DTT
SPICE
Second order
approximation


$V_{out}(t)$
(volts)
3
2
1
0
0
50
100
150
200
Time (ps)
DTT
SPICE
Third order
approximation

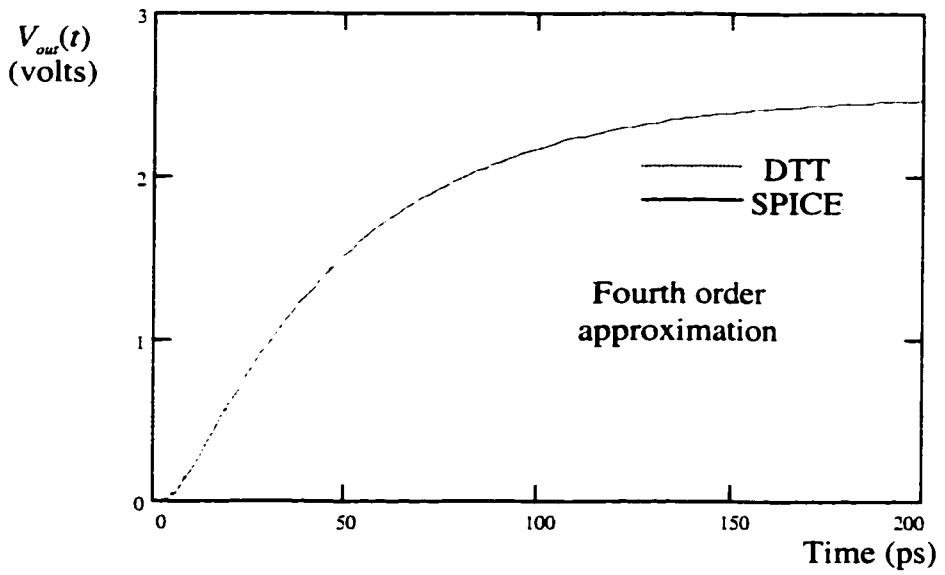


Figure 12.10. Transient response evaluated using the DTT method as compared to SPICE simulations for the circuit shown in Figure 12.9 using different approximation orders. SPICE simulations are represented by a solid line and the DTT simulations are represented by a dashed line. The circuit shown in Figure 12.9 is simulated with R_t = 50 Ω, C_t = 1 pF, R_{tr} = 25 Ω, and C_L = 0.05 pF.

The second circuit simulated using the DTT method is the *RC* tree shown in Figure 12.11. The transient response at several nodes of the tree are calculated based on the DTT method and compared to SPICE in Figure 12.12. A fourth order DTT approximation is used to calculate the transient responses shown in Figure 12.12. Note that a fourth order approximation is accurate as compared to SPICE simulations. In general, a fourth order approximation is sufficiently accurate for most *RC* trees. The guaranteed stability of a fourth order approximation is therefore a valuable

feature for *RC* circuits. Note that despite the fact that an *RC* circuit cannot produce complex poles [129]-[132], a reduced order approximation based on the DTT method can result in complex poles for an *RC* circuit. However, the resulting complex poles for *RC* circuits always produce accurate stable monotone responses.

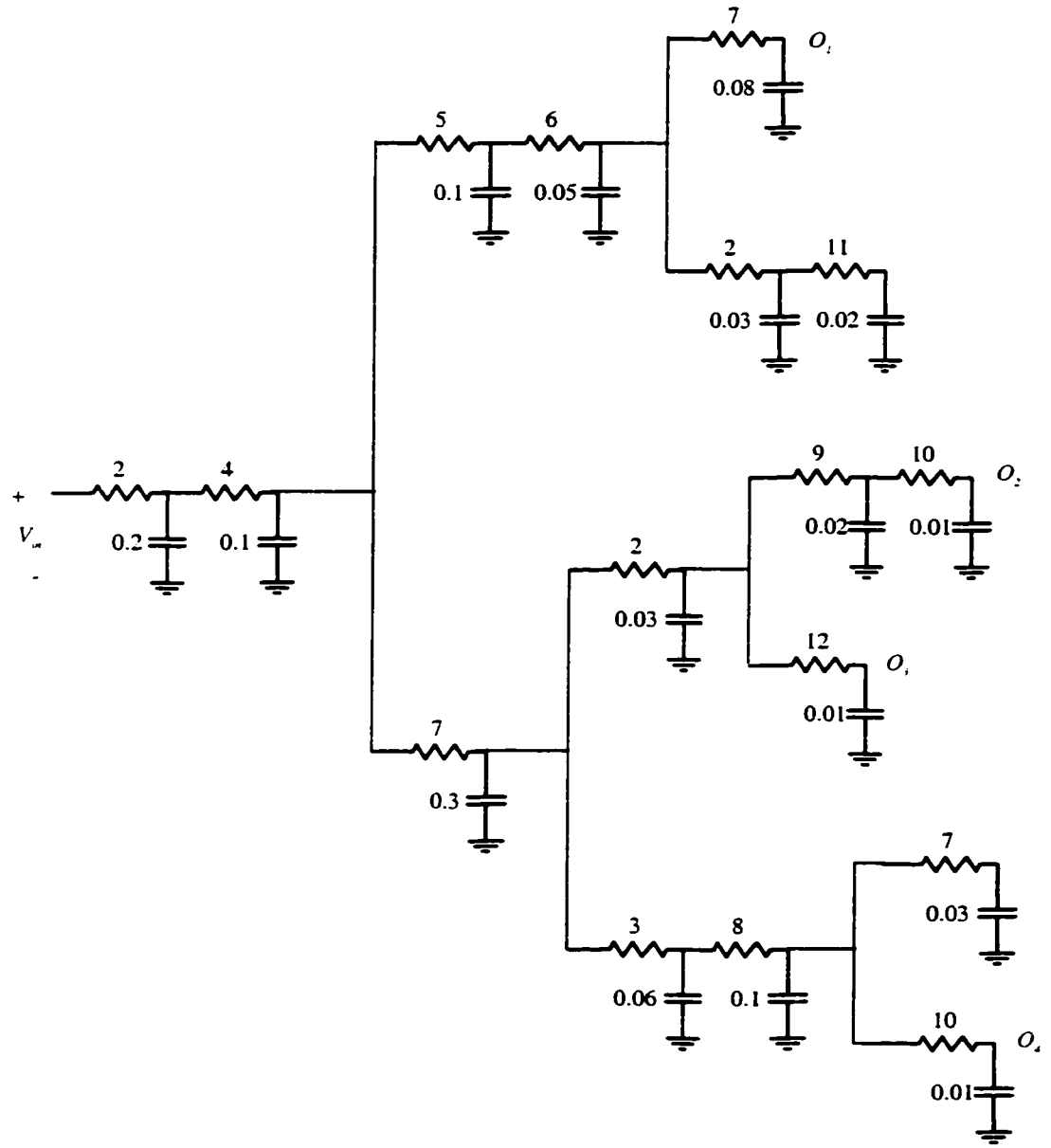
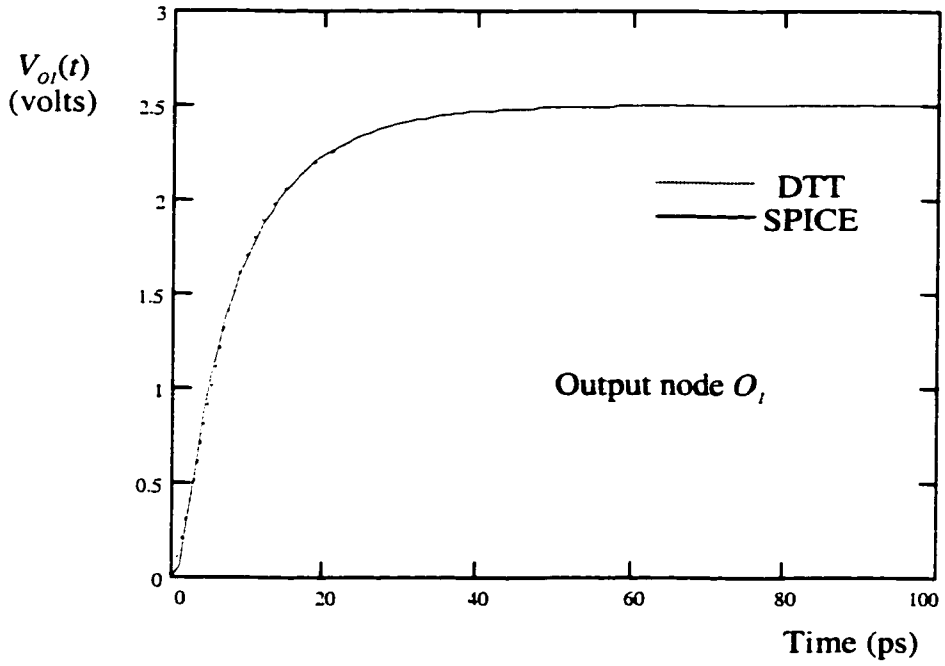
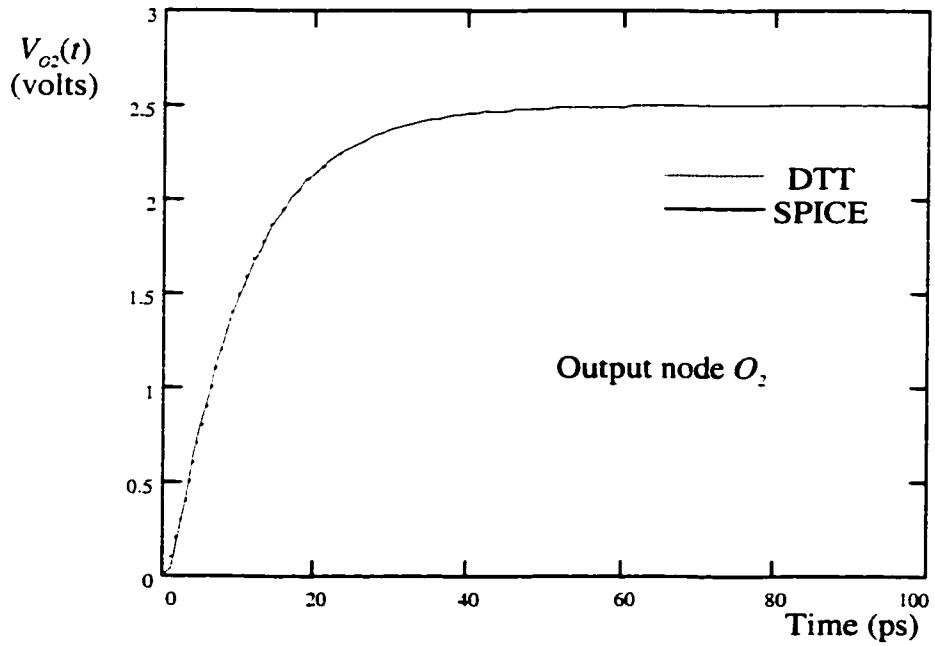


Figure 12.11. A general *RC* tree. The resistance values shown are in ohms, and capacitance values are in pF.


$V_{o1}(t)$
(volts)
3
2.5
2
1.5
1
0.5
0
0
20
40
60
80
100
Time (ps)
DTT
SPICE
Output node O_1


$V_{o2}(t)$
(volts)
3
2.5
2
1.5
1
0.5
0
0
20
40
60
80
100
Time (ps)
DTT
SPICE
Output node O_2

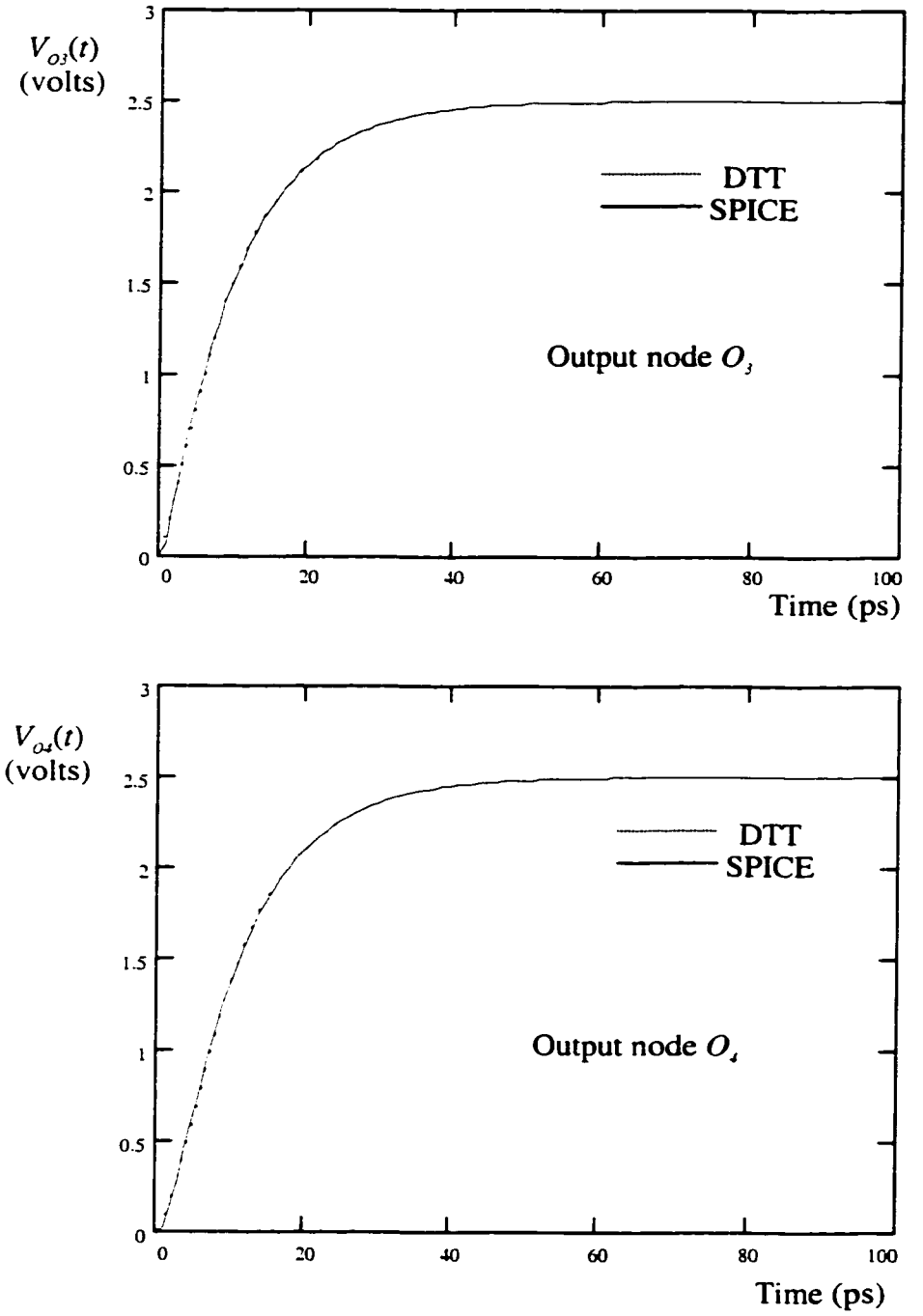


Figure 12.12. Transient response evaluated using the DTT method as compared to SPICE simulations at different nodes of the *RC* tree depicted in Figure 12.11. SPICE simulations are represented by a solid line and the DTT simulations are represented by a dashed line. A fourth order approximation is used.

The circuit shown in Figure 12.13 represents an *RLC* transmission line with a lumped source resistance and a load capacitance and is simulated using the DTT method. The transient response is calculated based on the DTT method with approximation orders of 4, 15, 25, and 35 and is compared to SPICE in Figure 12.14. Note that an approximation order between 25 and 35 is required for an underdamped response with second order oscillations to achieve a SPICE-like accuracy. Such high order approximations cannot be achieved by AWE [77]-[79] due to its numerical instability with high approximation orders. Other methods capable of calculating such high order approximations [97]-[136] have a much higher computational complexity as compared to the DTT method. The computational efficiency of the DTT method and its numerical accuracy for very high orders of approximation makes it suitable for accurately simulating *RLC* trees. Several simulations of the circuit shown in Figure 12.13 are shown in Figure 12.15 with different line parameters and source and load impedances. The DTT method accurately characterizes the waveform details as compared to SPICE.

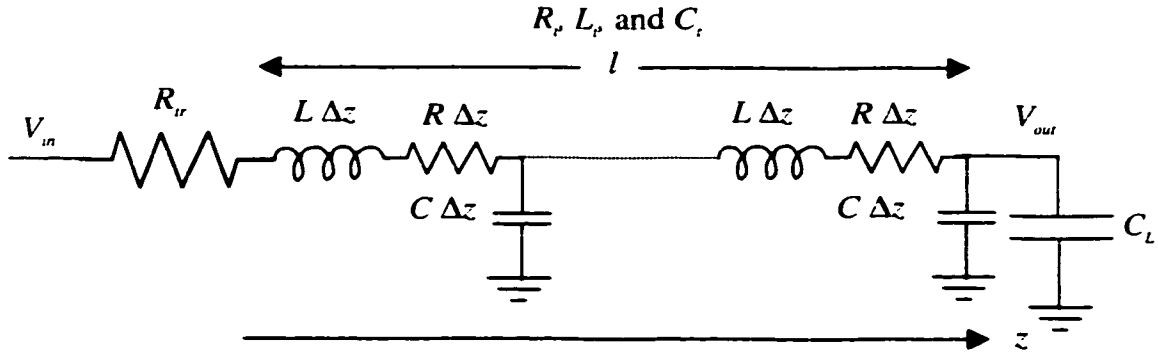
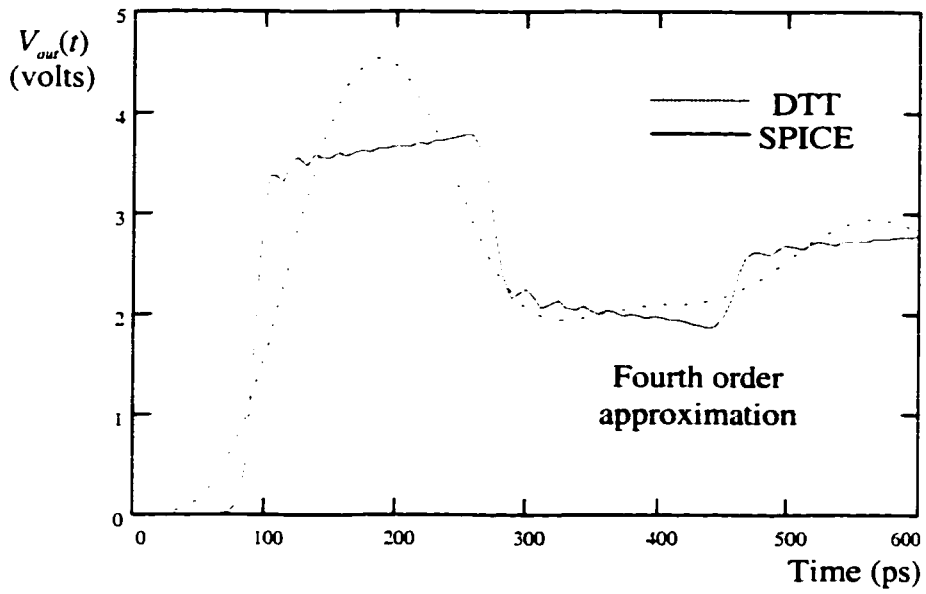
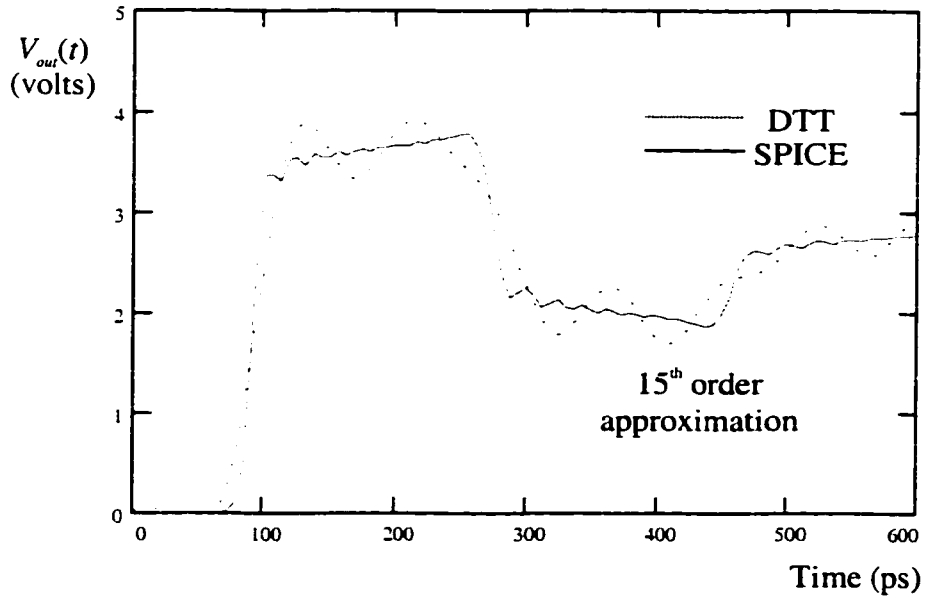


Figure 12.13. An *RLC* transmission line with a source resistance and a load capacitance.


$V_{out}(t)$
(volts)
DTT
SPICE
Fourth order
approximation
0
1
2
3
4
5
0
100
200
300
400
500
600
Time (ps)


$V_{out}(t)$
(volts)
DTT
SPICE
15th order
approximation
0
1
2
3
4
5
0
100
200
300
400
500
600
Time (ps)

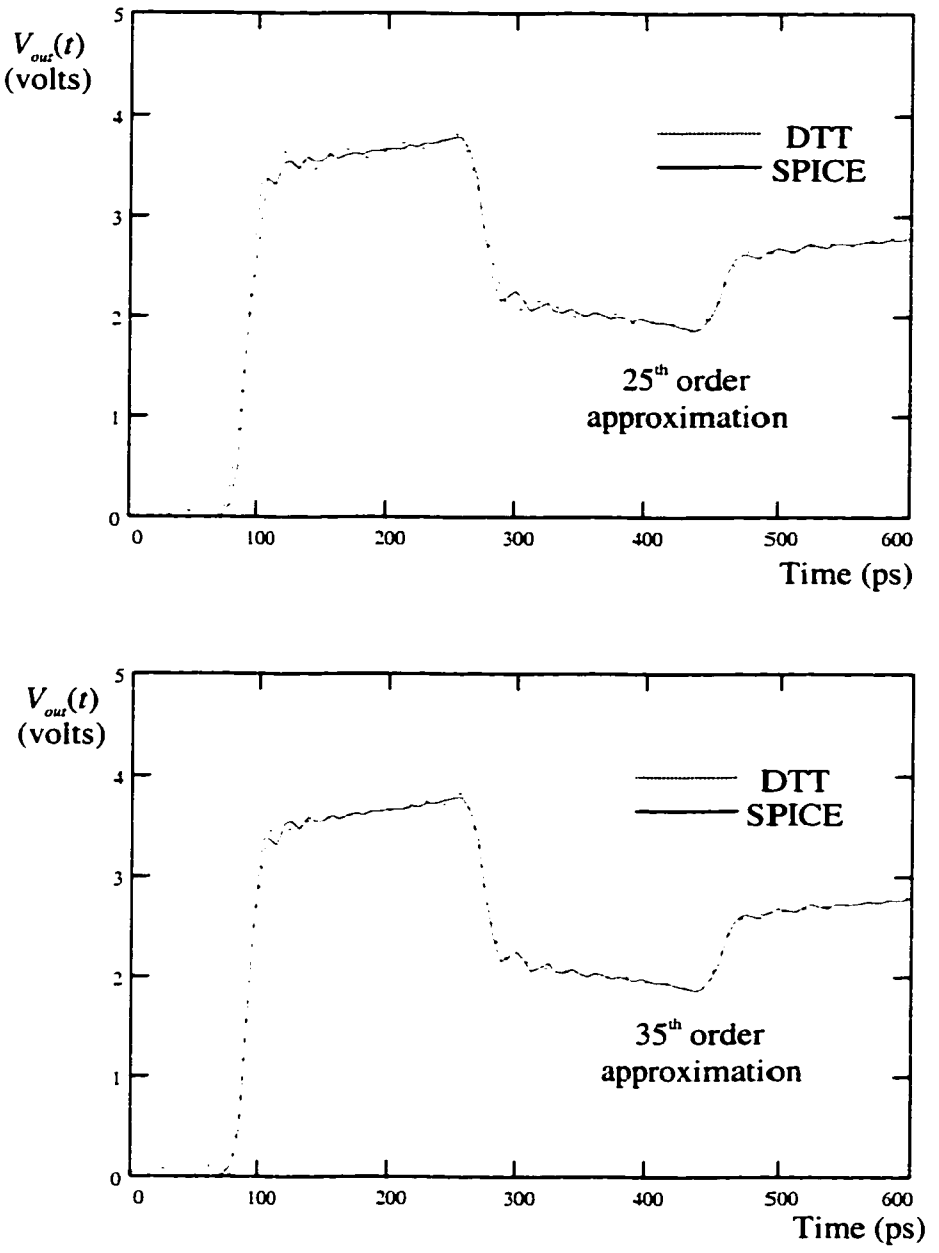


Figure 12.14. Transient response evaluated using the DTT method as compared to SPICE simulations for the circuit shown in Figure 12.13 using different orders of approximation. SPICE simulations are represented by a solid line and the DTT simulations are represented by a dashed line. The circuit shown in Figure 12.13 is simulated with $R_t = 40\ \Omega$, $L_t = 7$ nH, $C_t = 1$ pF, $R_{tr} = 10\ \Omega$, and $C_L = 0.1$ pF.

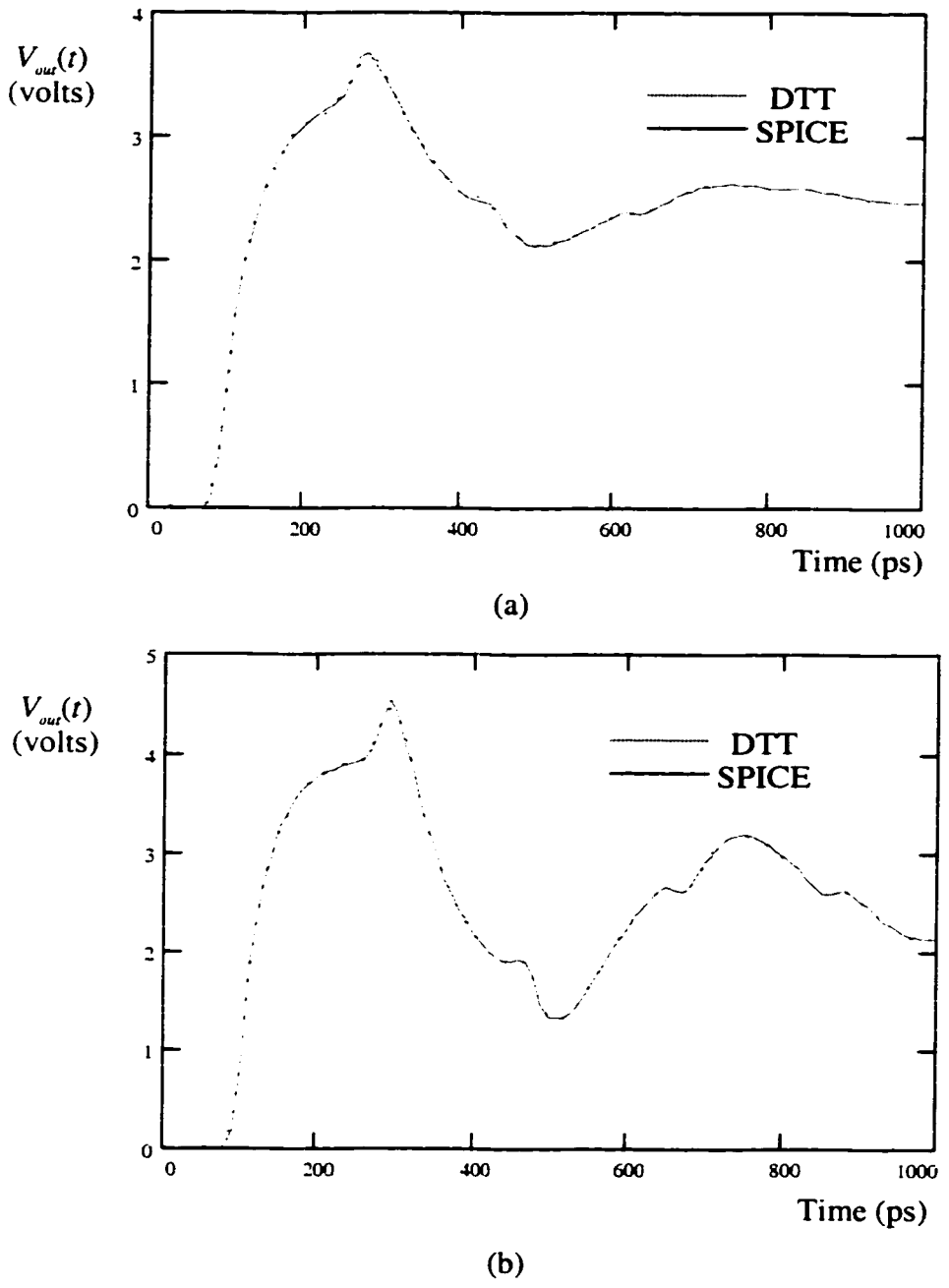
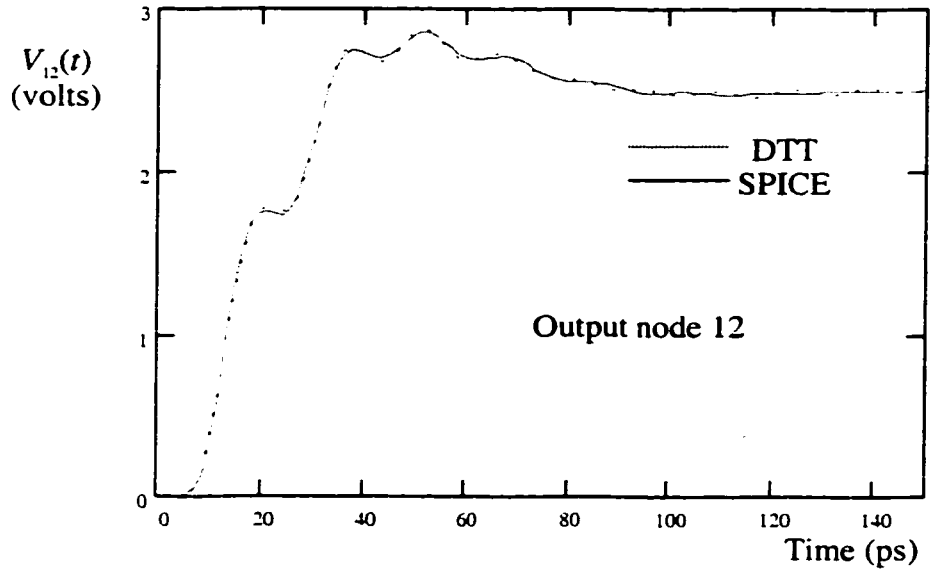


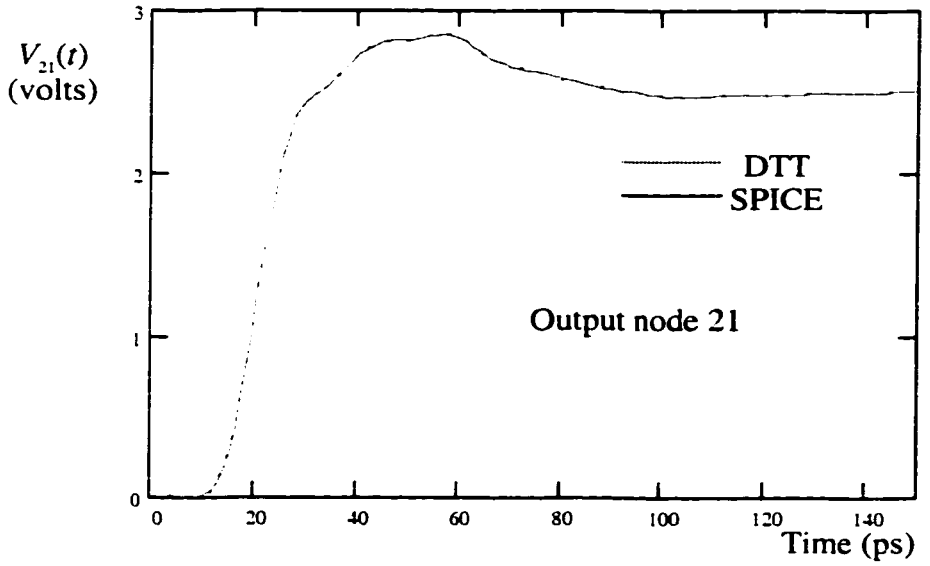
Figure 12.15. Transient response evaluated using the DTT method as compared to SPICE simulations for the circuit shown in Figure 12.13 using different line parameters. SPICE simulations are represented by a solid line and the DTT simulations are represented by a dashed line. (a) R_t = 30 Ω, L_t = 7 nH, C_t = 1 pF, R_{tr} = 20 Ω, C_L = 0.5 pF, and approximation order = 20. (b) R_t = 20 Ω, L_t = 8 nH, C_t = 1 pF, R_{tr} = 10 Ω, C_L = 0.4 pF, and approximation order = 25.

The transient response at several nodes of the *RLC* tree characterized in Table 1 are evaluated based on the DTT method and compared to SPICE in Figure 12.16. A 40^{th} order approximation is used and is highly accurate as compared to SPICE. A 45^{th} order approximation is used to evaluate the transient response of a large copper interconnect tree based on a 0.25 μm CMOS IBM technology. The tree has 673 capacitors and 673 inductors. The transient responses based on the DTT method and SPICE are compared in Figure 12.17. Note that the DTT method is capable of accurately characterizing the transient response of large industrial *RLC* trees with complicated non-monotone underdamped responses.

Table 12.1. A general *RLC* tree. The tree has several *RLC* sections, each section of which comprises a row of the table and has an ID number. The ID numbers of the left and right *RLC* sections driven by an *RLC* section are given in the fifth and sixth columns. A zero in these columns implies that the left or right sections do not exist.

RLC section number	R (Ω)	L (nH)	C (pF)	Left section number	Right section number
1	2	0.07	0.2	2	0
2	4	0.06	0.1	4	3
3	7	0.04	0.3	6	7
4	5	0.05	0.1	5	0
5	6	0.03	0.05	12	11
6	6	0.06	0.03	10	9
7	3	0.06	0.06	8	0
8	8	0.04	0.1	15	16
9	12	0.05	0.01	0	0
10	9	0.04	0.02	14	0
11	2	0.05	0.03	13	0
12	7	0.03	0.08	0	0
13	11	0.07	0.02	20	0
14	10	0.03	0.01	19	0
15	7	0.04	0.03	17	18
16	10	0.02	0.01	0	0
17	12	0.02	0.01	0	0
18	3	0.04	0.1	24	0
19	15	0.04	0.02	22	23
20	5	0.06	0.07	21	0
21	5	0.06	0.07	0	0
22	5	0.05	0.05	0	0
23	8	0.04	0.03	27	26
24	8	0.05	0.02	25	0
25	8	0.06	0.02	30	0
26	2	0.04	0.02	0	0
27	7	0.03	0.04	28	29
28	16	0.02	0.06	0	0
29	5	0.05	0.06	0	0
30	8	0.04	0.02	0	0


$V_{12}(t)$
(volts)
3
2
1
0
0
20
40
60
80
100
120
140
Time (ps)
DTT
SPICE
Output node 12


$V_{21}(t)$
(volts)
3
2
1
0
0
20
40
60
80
100
120
140
Time (ps)
DTT
SPICE
Output node 21

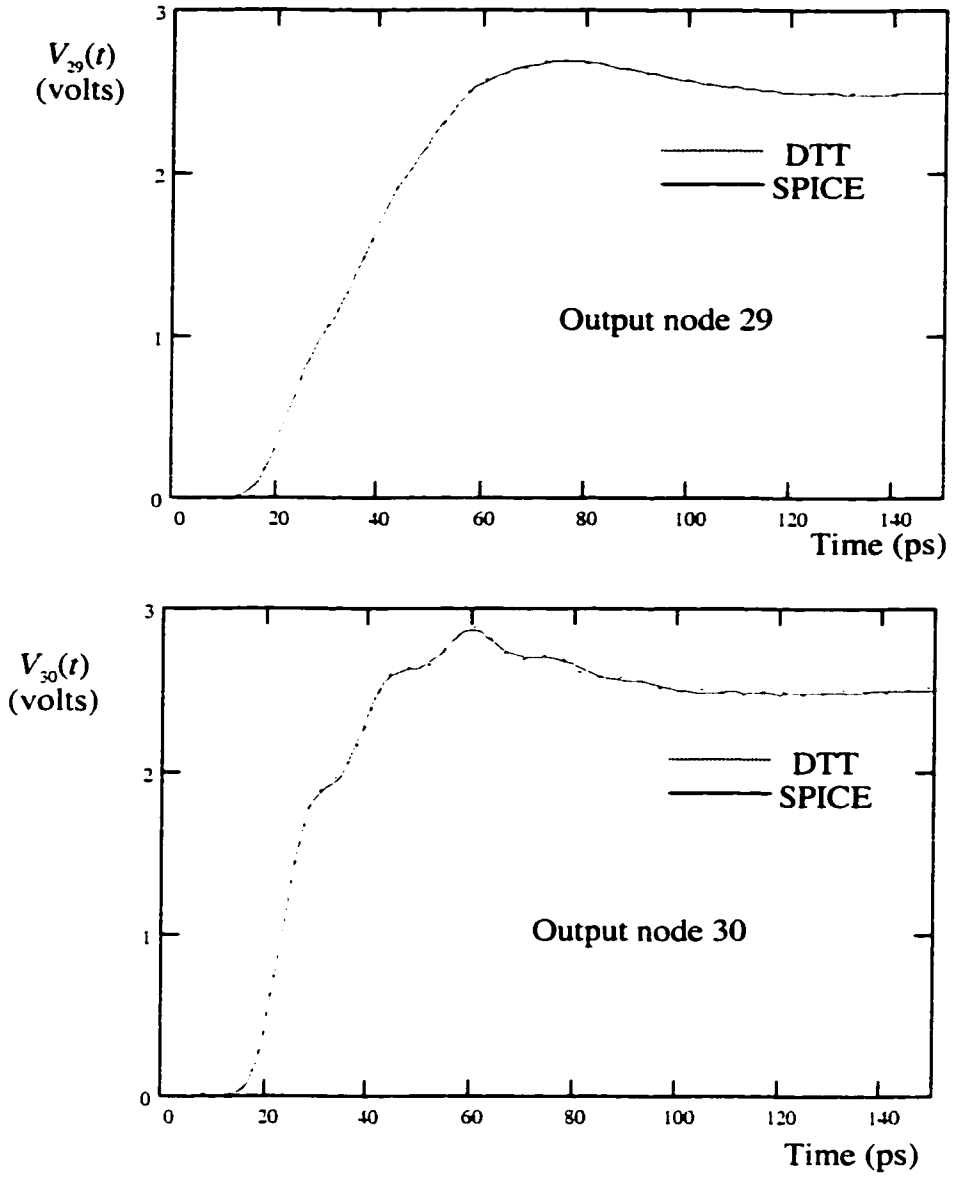


Figure 12.16. Transient response evaluated using the DTT method as compared to SPICE simulations at different nodes of the *RLC* tree characterized in Table 12.1. SPICE simulations are represented by a solid line and the DTT simulations are represented by a dashed line. A 40th approximation order is used.

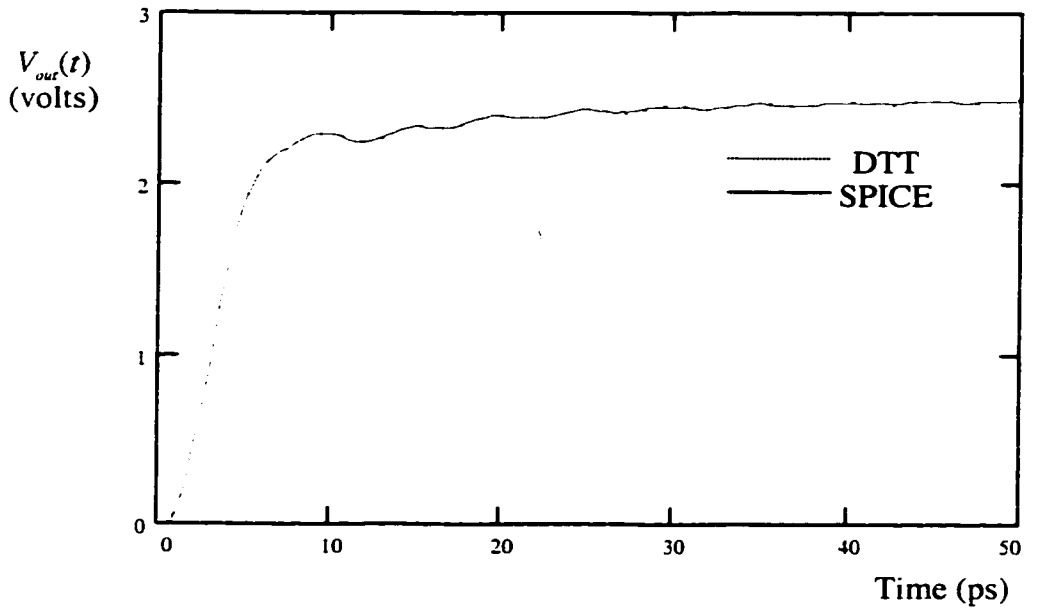


Figure 12.17. Transient response evaluated using the DTT method as compared to SPICE simulations at a particular leaf node of a large copper interconnect *RLC* tree based on an IBM 0.25 μm CMOS technology. SPICE simulations are represented by a solid line and the DTT simulations are represented by a dashed line. A 45th approximation order is used.

12.4 Conclusions

The DTT method has been introduced to evaluate the transient responses within *RLC* trees with arbitrary accuracy for any input signal. The DTT method is numerically accurate for any order of approximation, which permits approximations to be determined with a large number of poles appropriate for approximating *RLC* trees with underdamped responses. The DTT method is computationally efficient

with a complexity linearly proportional to the number of branches in the tree. A common set of poles is determined that characterizes the responses at all of the nodes of an *RLC* tree, which further enhances the computational efficiency of the proposed method. The stability is guaranteed by the DTT method for low order approximations with less than five poles, which is useful for efficiently analyzing *RC* circuits.

Chapter 13 On the Extraction of On-Chip Inductance

The efficient and accurate extraction of inductance is one of the primary bottlenecks that hinder incorporating on-chip inductance into integrated circuit design tools. To extract on-chip inductance, the return path of the current flowing in an interconnect line must be determined. Initial work has assumed the return path of the current to be within the substrate [45]-[48]. However, further investigation has provided evidence that the return path of the current is primarily within the power distribution network and other neighboring interconnect lines [49]-[56]. This characteristic of the return path severely complicates the process of accurately extracting the on-chip inductance since the value of the inductance of a wire not only depends on the wire characteristics but also on the characteristics of the other wires surrounding the line. This problem is further aggravated by the fact that the current return path can be distributed among many power and signal wires, some of which may be hundreds of micrometers away from the wire for which the inductance is being extracted [45], [49]-[56].

Some research on on-chip inductance extraction has been published which places an emphasis on accuracy and uses expensive 3-D numerical algorithms [45]-[53]. Fortunately, as shown in this chapter, on-chip inductance has two useful properties which enable simple methods to be used to extract the inductance. The first characteristic is that the sensitivity of a signal waveform to errors in the inductance

values is low, particularly the propagation delay and the rise time. It is quantitatively shown in this chapter that the error in the propagation delay and rise time is below 9.4% and 5.9%, respectively, assuming a 30% relative error in the extracted inductance values. If an *RC* model is used for the same example, the corresponding errors are 51% and 71%, respectively. The second characteristic is that the magnitude of the on-chip inductance is a slow varying function of the width of a wire and the geometry of the surrounding wires. These two characteristics can be exploited by using simplified techniques that permit approximate and sufficiently accurate values of the on-chip inductance to be generated with high computational efficiency. These two characteristics are described and analyzed in section 13.1. A summary is provided in section 13.2.

13.1 Characteristics of On-Chip Inductance which Simplify the Extraction Process

Two characteristics of on-chip inductance can be exploited to simplify the extraction process of on-chip inductance. These two characteristics are discussed in this section.

First characteristic: The sensitivity of a signal waveform to errors in the inductance values is low, particularly the propagation delay and rise time.

This characteristic occurs since inductance only appears under a square root function in a waveform or timing expression. The reason for this square root dependence is completely physical since an *LC* constant has the dimensions of time

squared, where L and C are any inductance and capacitance values in the circuit, respectively. The square root dependence can be compared to the linear dependence of the delay expressions on the resistance since any RC constant has the dimensions of time, where R is any resistance of the circuit. For example, according to the equivalent Elmore delay for RLC trees that was introduced in [113] (see Chapter 7), the 50% delay of the signal at node i of an RLC tree is

$$t_{pdi} = 1.047 \cdot \sqrt{\sum_k C_k L_{ik}} \cdot e^{\frac{\zeta_i}{0.85}} + 0.695 \cdot \sum_k C_k R_{ik} \,, \tag{13.1}$$

where ζ_i is the damping factor at node i and is

$$\zeta_i = \frac{1}{2} \frac{\sum_k C_k R_{ik}}{\sqrt{\sum_k C_k L_{ik}}}. \tag{13.2}$$

The square root dependence of the propagation delay on the inductance values in an RLC tree is evident in (13.1) and (13.2).

To quantify the error in the propagation delay due to errors in the extracted inductance, consider an extraction tool that generates a value of the extracted inductance with a maximum error E relative to the actual inductance value. Alternatively, the extracted inductance based on this extraction tool is in the range between L (1-E) and L (1+E) where L is the actual inductance value. The worst case error in the propagation delay occurs when all of the inductance values are overestimated by the maximum factor of (1+E) [or underestimated by the minimum factor of (1-E)]. In that case, the propagation delay in (13.1) becomes

$$t_{pdi\_E} = 1.047 \cdot \sqrt{1+E} \cdot \sqrt{\sum_k C_k L_{ik}} \cdot e^{\frac{\zeta_i}{0.85\sqrt{1+E}}} + 0.695 \cdot \sum_k C_k R_{ik} \,, \tag{13.3}$$

where L_{ik} represent the actual inductance values and (13.1) represents the actual propagation delay. Errors in the extracted inductance values result in a worst case relative error of the propagation delay as given by

$$Et_{pdi\_E} = \left| \frac{t_{pdi} - t_{pdi\_E}}{t_{pdi}} \right| = \left| \frac{1.047 \cdot \left[e^{\frac{\zeta_i}{0.85}} - \sqrt{1+E} \cdot e^{\frac{\zeta_i}{0.85\sqrt{1+E}}} \right]}{1.047 \cdot e^{\frac{\zeta_i}{0.85}} + 1.39 \cdot \zeta_i} \right| . \tag{13.4}$$

The worst case error in the propagation delay only depends on the damping factor and the worst case error in the extracted inductance values. Another interesting metric is the error in the propagation delay due to neglecting inductance altogether and using an *RC* interconnect model. The propagation delay in this case can be calculated by letting $L_{ik} \rightarrow 0$ in (13.1) and is

$$t_{pdi\_RC} = 0.695 \cdot \sum_k C_k R_{ik} , \tag{13.5}$$

which is simply the Elmore (Wyatt) approximation of the propagation delay [126], [127]. Thus, the relative error in the propagation delay when inductance is not extracted and an *RC* model is used is

$$Et_{pdi\_RC} = \left| \frac{t_{pdi} - t_{pdi\_RC}}{t_{pdi}} \right| = \left| \frac{1.047 \cdot e^{\frac{\zeta_i}{0.85}}}{1.047 \cdot e^{\frac{\zeta_i}{0.85}} + 1.39 \cdot \zeta_i} \right| . \tag{13.6}$$

The relative error in the propagation delay due to using an *RC* model is only a function of the damping factor.

The worst case error in the propagation delay due to errors in the extracted inductance values as given by (13.4) is plotted in Figure 13.1 versus ζ_i for several values of *E*. Equation (13.6) is also plotted in Figure 13.1. Note in Figure 13.1 that

including the extracted inductance when evaluating the propagation delay significantly improves the accuracy as compared to an *RC* model even with a 30% error in the extracted inductance. The error in the propagation delay decreases with increasing ζ_i since a higher damping factor means the inductance has less effect and most of the delay is due to the *RC* time constants in the circuit, which diminishes the relevance of the error caused by inexact inductance values. Note also that the improvement in accuracy by extracting approximate values of inductance as compared to using an *RC* model increases as the importance of the inductance increases (for small ζ_i, which is the range of primary interest). Numerical values of the error in the propagation delay are listed in Table 13.1 with different accuracy levels of the extracted inductance values and with no inductance extracted. At $\zeta_i = 0.4$ and with a relative error in the inductance values of 30%, the relative error in the propagation delay improves by a factor of five as compared to using an *RC* model.

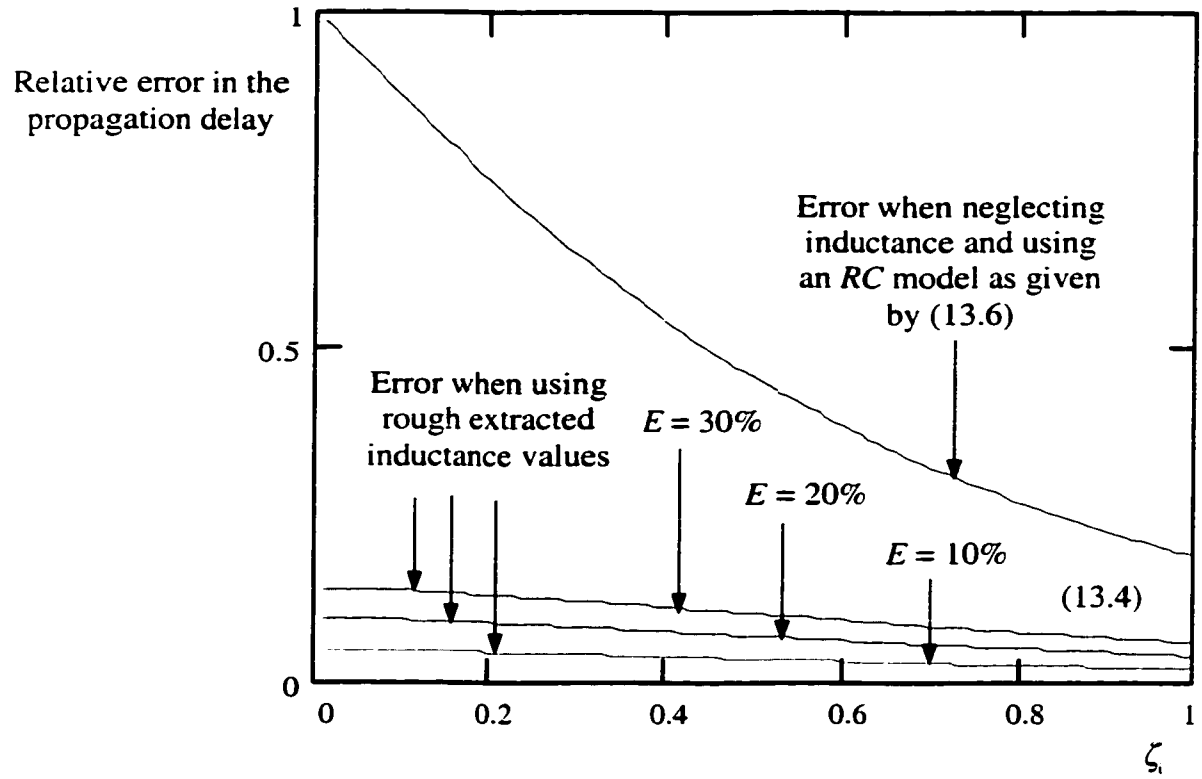


Figure 13.1. The relative error in (13.4) and (13.6) is plotted versus ζ_i. Several values of E in (13.4) are used as labeled in the figure.

Table 13.1 Relative error of the propagation delay when inductance is extracted and when an *RC* model is used. The relative errors for the extracted inductance values are 10%, 20%, and 30%.

ζ_i	Relative error of the propagation delay			
	Extraction error $E = 10\%$	Extraction error $E = 20\%$	Extraction error $E = 30\%$	*RC* model (no inductance)
0.0	4.9%	9.5%	14%	100%
0.2	4.5%	8.8%	13%	75%
0.4	3.9%	7.6%	11%	54%
0.6	3.2%	6.3%	9.3%	39%
0.8	2.6%	5.1%	7.5%	27%
1.0	2%	4%	6%	19%

As an example, consider the *RLC* tree shown in Figure 13.2. AS/X [128] simulations are performed for the *RLC* tree shown in Figure 13.2 with the inductance values shown in the figure, with no inductance (an *RC* model), and with all of the inductance values increased by 10%, 20%, and 30%. These simulations are depicted in Figure 13.3. Note in the simulations that using an approximate inductance estimation greatly improves the accuracy of the waveform as compared to using an *RC* model. The 50% delay and the 10% to 90% rise time are depicted in Table 13.2 for the circuit simulations shown in Figure 13.3. With a 30% error in the inductance values, the propagation delay differs by 9.4% from the actual value as compared to 51% if an *RC* model is used. The improvement in the rise time is even greater. The rise time differs from the actual value by 5.9% with a 30% error in the inductance values as compared to a 71% error when an *RC* model is used. Note that these are worst case errors since the circuits are simulated after overestimating all of the inductance values in the circuit by 30%. Practically, some of the inductance values are overestimated while others are underestimated. The error in the different directions partially cancels, thereby reducing the error in the total delay. The maximum error in the waveform shape occurs around the overshoots (see Figure 13.3). However, estimating the overshoot requires less accuracy since the overshoot is usually evaluated to decide if the overshoot is within an acceptable limit. This high tolerance of the delay expressions to errors in the extracted inductance encourages the use of simplified techniques with higher computational efficiency to extract the on-chip inductance.

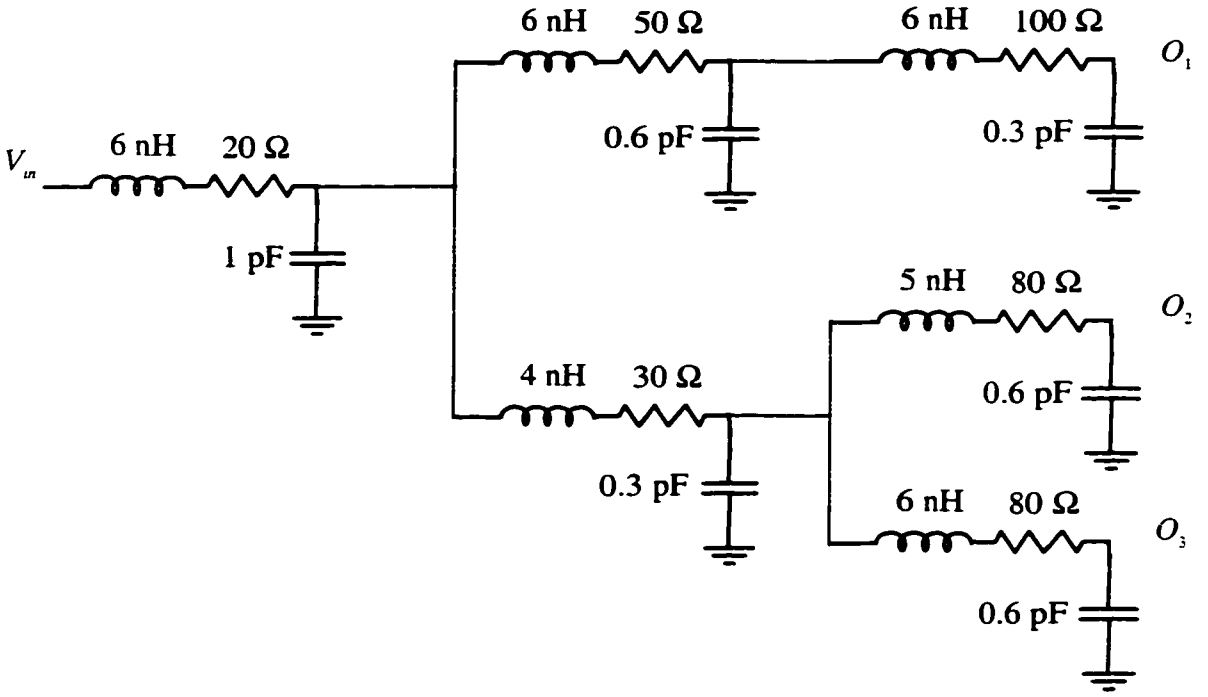


Figure 13.2. An example of an *RLC* tree

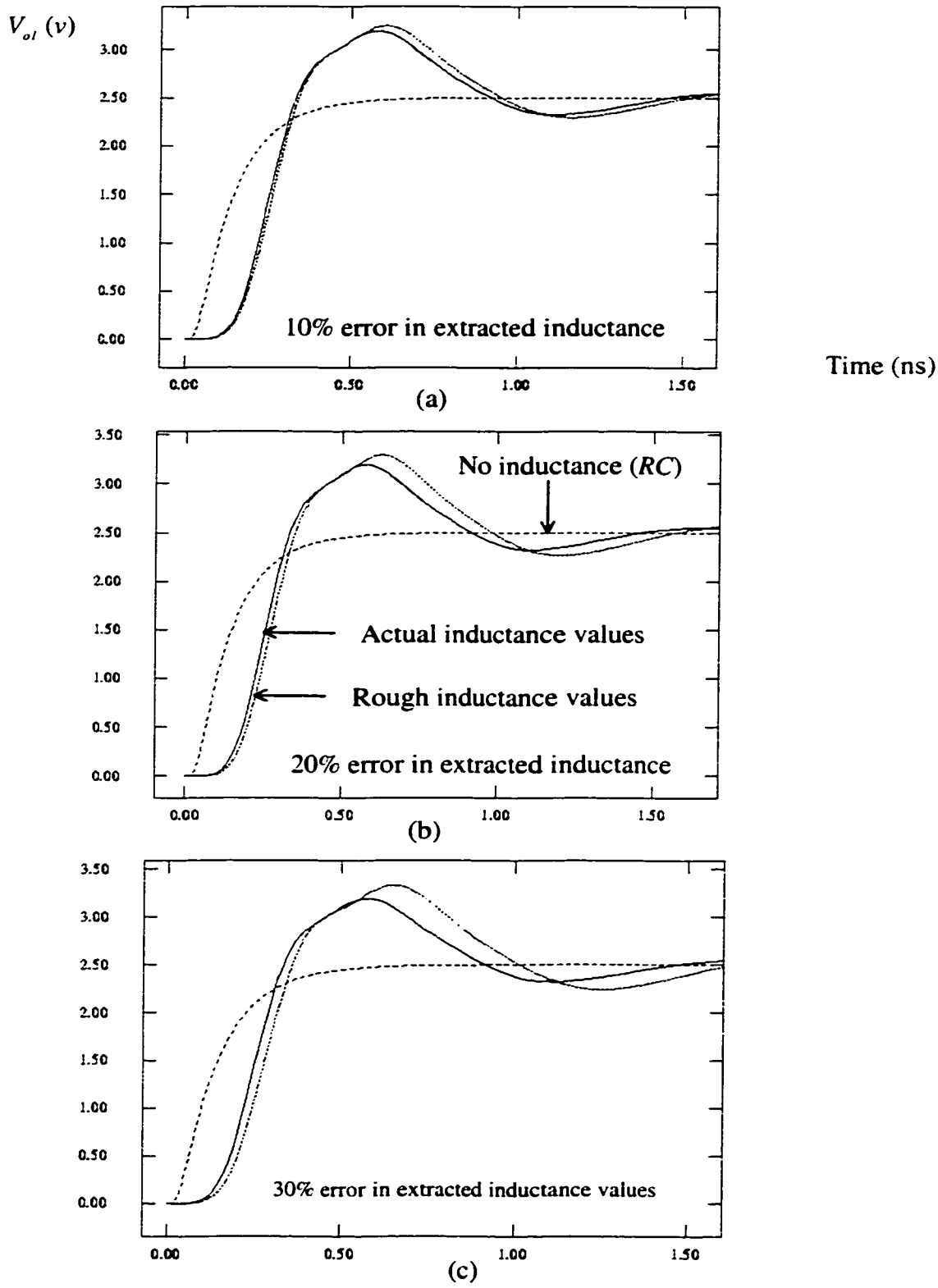


Figure 13.3. AS/X [128] simulations of the *RLC* tree shown in Figure 13.2 at output node O_1 with the actual inductance values, with no inductance (an *RC* model), and with all of the inductance values increased by a) 10%, b) 20%, and c) 30%.

Table 13.2 The 50% delay and the 10%-to-90% rise time from AS/X [128] simulations for the *RLC* tree shown in Figure 13.2 with the actual inductance values, with all of the inductance values increased by 10%, 20%, and 30%, and with no inductance (an *RC* model).

Relative error in inductance values		*RC* (no inductance)	Relative error ⟷	Actual inductance values	Relative error ⟷	Rough inductance values
10%	t_{pd} (ps)	116	51%	233	3.4%	241
	t_r (ps)	260	71%	152	2.0%	155
20%	t_{pd} (ps)	116	51%	233	6.9%	249
	t_r (ps)	260	71%	152	3.9%	158
30%	t_{pd} (ps)	116	51%	233	9.4%	255
	t_r (ps)	260	71%	152	5.9%	161

Second characteristic: The value of the on-chip inductance is a slow varying function of the width of the wire and the geometry of the surrounding wires.

Most of the analytical formulae approximating the on-chip inductance has a logarithmic dependence on the width of the interconnect [48]-[52], [55], which is a slow varying function. Also numerical three-dimensional extraction methods based on solving Maxwell's differential equations and experimental measurements have demonstrated this slow varying dependence of the on-chip inductance on the wire width and the surrounding wires geometries [49]-[53]. On-chip inductance values for a high performance VLSI circuit are typically between 4 nH/cm and 6 nH/cm for the range of wire widths used in [49]-[53].

This characteristic together with the low sensitivity of the delay expressions to errors in the extracted inductance values permit extraction techniques as simple as

using a constant inductance value per unit length of interconnect of 5 nH/cm. Using this constant average value technique results in a maximum overestimation of 25% and a maximum underestimation of 16.6% for the inductance values. According to this sensitivity analysis, these errors in the inductance values result in errors of less than 9% in the propagation delay and below 5% in the rise time for typical damping factors commonly seen in VLSI circuits (*i.e.*, $\zeta_i > 0.4$) [49]-[53]. As an example, all of the inductance values in the tree shown in Figure 13.2 are recalculated assuming a value of 5 nH/cm. AS/X [128] simulations are performed for the resulting tree and for the tree with actual inductance values. These simulations are shown in Figure 13.4. The actual values of the propagation delay and rise time are 233 ps and 152 ps, respectively. The estimated propagation delay improves from 116 ps when using an *RC* model to 215 ps when using a constant inductance value of 5 nH/cm, which is equivalent to an improvement in the relative error from 51% to 7.7%. The estimated rise time improves from 260 ps to 146 ps, which is equivalent to an improvement in the relative error from 71% to 3.9%. There is an insignificant overhead in including inductance using this method. However, a significant amount of information characterizing the signal waveform shape which is lost when using an *RC* model can be retrieved by using this simple technique. If a more accurate inductance estimation is required, other methods can be used such as simple curve fitting or look-up table methods to quickly estimate the inductance based on the wire width and certain characteristics of the power distribution network such as the metal pitch. Other techniques can be considered which utilize simple analytical formulae to provide an approximate estimate of the inductance based on the wire width and certain characteristics of the power distribution network [55].

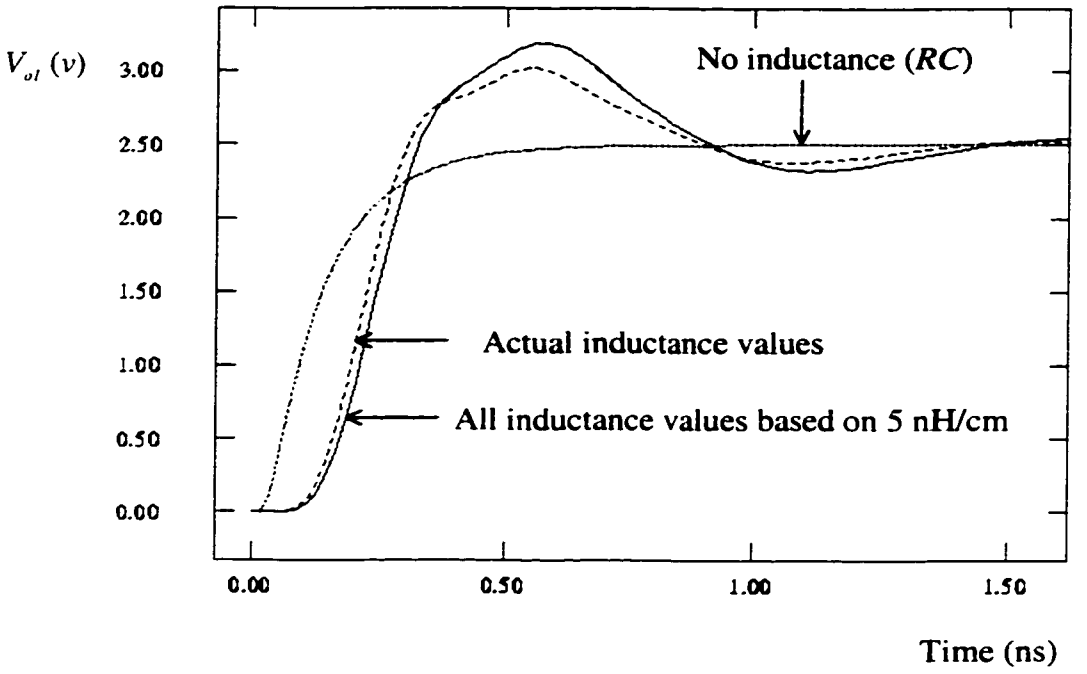


Figure 13.4. AS/X [128] simulations of the *RLC* tree shown in Figure 13.2 at output node O_1 with the actual inductance values, with no inductance (an *RC* model), and with all of the inductance values recalculated based on a value of 5 nH/cm inductance per unit length.

13.2 Summary

Two characteristics of on-chip inductances have been discussed in this chapter that can be exploited to significantly simplify the extraction of on-chip inductance. The first characteristic is that the sensitivity of a signal waveform to errors in the inductance values is low, particularly the propagation delay and the rise time. It is quantitatively shown in this chapter that the error in the propagation delay and rise time is below 9.4% and 5.9%, respectively, assuming a 30% relative error in the extracted inductance values. If an *RC* model is used for the same example, the

corresponding errors are 51% and 71%, respectively. The second characteristic is that the value of the on-chip inductance is a slow varying function of the width of the wire and the geometry of the surrounding wires. These two characteristics can be exploited by using simplified techniques to generate approximate inductance values with high computational efficiency. A trivial method such as using a constant value of average inductance per unit length has also been shown to significantly improve the accuracy of the propagation delay and the rise time as compared to simply using an *RC* model. Thus, it is recommended to tradeoff accuracy for computational efficiency when extracting on-chip inductance.

Chapter 14 Conclusions

Over the last two decades, the relative importance of interconnect to integrated circuits has increased dramatically with the interconnect contributing a significant portion of the overall delay. With this increasing importance of the interconnect, more accurate interconnect models are required to design correctly functioning high speed integrated circuits. To date, the VLSI industry has predominantly modeled the interconnect as one of three models: a short-circuit for short, low resistance, low capacitance wires with high output impedance drivers, a lumped capacitance for those wires with low total resistance relative to the driver output resistance but non-negligible capacitance, and an *RC* distributed impedance for longer, highly resistive wires.

It has been shown in this dissertation that inductance already has tangible effects in current high speed integrated circuits. For example, neglecting inductance and using an *RC* interconnect model in an IBM 0.25 μm CMOS production technology can cause large errors (over 35%) in the propagation delay for on-chip interconnect. Using an *RC* model always underestimates the propagation delay, consuming safety margins and decreasing the reliability of these integrated circuits which are designed based on an *RC* interconnect model. It has also been shown that including inductance in the repeater insertion design process as compared to using an *RC* model improves the overall repeater solution in terms of area, power, and delay. The average savings in area, power, and delay for the set of trees described in Chapter

9 are 40.8%, 15.6%, and 6.7%, respectively, when inserting repeaters based on an *RLC* delay model as compared to an *RC* delay model (based on a 0.25 μm CMOS technology with copper interconnect). The average savings in area, power, and delay increases to 62.2%, 57.2%, and 9.4%, respectively, when using repeaters from a CMOS technology that is five times faster with the same set of interconnect trees.

The increasing importance of inductance can dramatically change the methodologies used today to analyze and design high speed integrated circuits. It is shown in this dissertation that the traditional quadratic dependence of the propagation delay on the length of the interconnect for *RC* lines tends to a linear dependence as inductance effects increase. This behavior has profound effects on the repeater insertion process since repeaters are inserted within on-chip interconnect to reduce this square dependence of the propagation delay on the interconnect length. Computationally efficient delay models are required to estimate the delay and waveform characteristics within *RLC* lines, which can be used within optimization tools and related design methodologies. Also, accurate simulators capable of capturing the details of complicated non-monotone responses present in *RLC* interconnect are required to simulate the critical paths of an integrated circuit. Note that these complicated non-monotone responses do not exist with other interconnect models such as lumped capacitors or *RC* impedances. Moreover, different transistor and interconnect sizing techniques are required when inductance effects become significant. For example, buffers driving a highly inductive interconnect need to be sized correctly to match the characteristic impedances of the interconnect.

Inductance is shown to have useful effects on the performance of high speed integrated circuits. Specifically, inductance improves the signal slew rate,

dramatically reduces the short-circuit power consumption, reduces the area of the active repeaters inserted to optimize the performance of these long inductive interconnect lines. For example, the power consumption of a specific industrial clock distribution network (see Chapter 11) decreases from 3670 pJ/cycle to 1582 pJ/cycle and the slew rate decreases from 1.2 ns to 200 ps on the internal nodes of the clock distribution network when wider, more inductive wires are used. These beneficial effects encourage design strategies that can exploit on-chip inductance.

Several tools have been presented in this dissertation that can be used for the design and analysis of integrated circuits which consider inductance:

1. A general method to characterize the response of a linear non-monotone system that is equivalent to the Elmore delay. The generated delay expressions for an *RLC* tree have the same accuracy characteristics as the Elmore (Wyatt) approximation for *RC* trees. Simple analytical expressions of signals in an *RLC* tree are provided for the 50% delay, the rise time, overshoots, and settling time. These expressions consider both monotone and non-monotone signal responses. The delay expressions are continuous and hence are useful for optimization and synthesis in VLSI-based design methodologies. The second order approximation introduced here is always stable and can be used with arbitrary inputs.
2. Simple to use figures of merit have been developed that determine the relative importance of including inductance in the model of on-chip interconnect for a given integrated circuit (or a portion of an integrated circuit). These figures of merit are useful since including inductance requires the extraction of inductance and slows down design and analysis

tools due to the more complicated interconnect model. Hence inductance should only be included where intolerable errors are incurred if inductance is neglected.

3. An algorithm has been introduced to insert and size repeaters within an *RLC* tree to minimize a variety of possible cost functions. The algorithm has a polynomial complexity proportional to the square of the number of possible repeater positions and determines a repeater solution that is close to the global minimum. This algorithm is based on the equivalent Elmore delay model described in Chapter 7.
4. The DTT method has been introduced to evaluate the transient responses within *RLC* trees with arbitrary accuracy for any input signal. The DTT method is numerically accurate for any order of approximation, permitting solutions to be determined with a large number of poles appropriate for approximating *RLC* trees with underdamped responses. The DTT method is computationally efficient with a complexity linearly proportional to the number of branches in the tree. A common set of poles is determined that characterizes the responses at all of the nodes of an *RLC* tree, further enhancing the computational efficiency of this simulation algorithm. The stability is guaranteed by the DTT method for low order approximations with less than five poles, which is quite useful when analyzing *RC* circuits.
5. Closed form solutions to estimate the power consumption of CMOS gates driving inductive interconnect. Additional closed form solutions are also provided for evaluating the transient response of CMOS gates driving lossy transmission lines.

6. Strategies for fast, less accurate extraction methods for on-chip inductance since the sensitivity of a signal waveform to errors in the inductance values is low, particularly the propagation delay and the rise time. Hence, reasonable error in the extracted inductance has only a slight effect on the delay expressions and waveform shapes. Furthermore, the value of the on-chip inductance is a slow varying function of the width and geometry of the surrounding wires. This characteristic limits the overall error in the extracted inductance.

Certain VLSI trends will further the importance of inductance such as:

1. Lower resistivity metal alloys for interconnect, copper interconnect being a primary example.
2. Lower permitivity dielectrics to insulate the interconnect, thereby reducing the interconnect capacitance. Reducing the interconnect capacitance increases the effects of inductance.
3. Higher operating frequencies with faster signal transition times.
4. Faster devices with technology scaling and the increasing use of SOI devices with significantly higher speed and lower line-to-ground capacitance. Using faster devices increases the error caused by neglecting inductance in the repeater insertion process.
5. Tighter timing constraints in VLSI circuits to meet higher frequency targets which require more accurate delay models.

Therefore, it is imperative that inductance be included in the interconnect impedance model when analyzing and designing current and next generation high speed circuits.

Chapter 15 Future work

The significance of inductance in current VLSI technologies has been investigated in this dissertation. It has been shown that neglecting inductance can cause significant error in estimating the propagation delay of signals within VLSI circuits, resulting in inefficient and inaccurate design methodologies. Enhanced design methodologies have been presented that consider inductance in the repeater insertion process, for estimating the delays in *RLC* transmission lines and trees, and for estimating the power of *LC* systems. Trends in the integrated circuit industry such as higher operating frequencies, lower resistivity alloys for interconnect, faster devices, and the increasing use of wider wires at higher interconnect levels for global routing increase the importance of the on-chip inductance, requiring inductance to be included within the interconnect model. A more thorough understanding of the nature of on-chip inductance effects is therefore necessary. Also, design tools and methodologies need to be developed to cope with current and future industrial requirements. Future research in these directions includes:

1. Extending the DTT method to simulate any linear circuit.

   The DTT method has been discussed in Chapter 12. This algorithm simulates the signals within *RLC* trees with SPICE-like accuracy while remaining computationally efficient. However, *RLC* trees represent only a limited category of an *RLC* circuit. *RLC* circuits with multiple inputs,

inductive and capacitive coupling, and loops (or meshes) are examples of other circuit structures that exist in an integrated circuit. Also, arbitrary initial conditions can be useful in simulating certain VLSI structures or when using piecewise linear transistor models. This possible research task investigates the extension of the DTT method to efficiently simulate these circuit structures.

2. Innovative techniques to simulate circuits with both linear and non-linear elements.

   The DTT method is a frequency domain simulation method that depends upon evaluating the dominant poles and zeros of a transfer function. In general, frequency domain simulation techniques are not well suited to simulate circuits with nonlinear elements. Some event driven techniques exist that divide the characteristics of a nonlinear device into several piecewise linear regions, linearizing the device several times during a simulation run. However, the circuit has to be linearized and simulated several times depending upon the number of nonlinear devices and the number of piecewise linear regions. The number of times the circuit is linearized and simulated can be large for circuits with many nonlinear elements. This behavior significantly deteriorates the efficiency of frequency domain simulation techniques as compared to time marching techniques. This possible research task investigates innovative ways to efficiently simulate circuits with nonlinear elements in the frequency domain.

3. Investigating the dynamic and short-circuit power dissipated by CMOS gates driving *RLC* loads.

   The dynamic and short-circuit power of a CMOS gate driving a lossless *LC* transmission line as a limiting case of an *RLC* line has been investigated in Chapter 10. The goal of this research task is to develop simple analytical formulae for the dynamic and short-circuit power of a CMOS gate driving a lossy *RLC* transmission line, which would prove useful in evaluating power dissipation in high speed circuits.

4. Investigating the nature of inductive noise in VLSI circuits

   Inductive noise is significantly different from capacitive coupling noise since the inductive coupling physically reaches much further than capacitive coupling. Typically, an interconnect line capacitively couples to the adjacent lines in the same layer and neighboring layers. However, the same line can inductively couple to other lines within a distance of hundreds of micrometers. This feature of inductive coupling makes the nature of inductive noise significantly more complicated than capacitive noise. Furthermore, the frequency behavior of an inductive impedance is completely different from the frequency behavior of a capacitive impedance. This possible research task will focus on characterizing the inductive noise in high speed integrated circuits as well as provide a more thorough understanding of the nature of inductive noise.

5. Noise resistant circuit design techniques and circuit families.

   The type of logic family in an integrated circuit can have a profound effect on the magnitude of the inductive noise exhibited by the circuit. For example, some dynamic logic families precharge all of the gates simultaneously at the beginning of a precharge cycle and discharge some nodes at the beginning of an evaluation cycle. Such logic families result in currents flowing in the same direction in all of the gates at specific times. These unidirectional currents accumulate, resulting in large current changes in the power and ground lines, thereby increasing the *Ldi/dt* noise. Also, such high currents in one direction can create large inductive coupling in other lines, hence increasing inductive noise in these signal lines. This research task will focus on characterizing the effect of different logic families on noise as well as developing new low-noise circuit families.

6. Tools for efficiently evaluating noise in integrated circuits.

   Evaluating noise in integrated circuits is of crucial importance to guarantee that the noise is within acceptable bounds, thereby insuring the reliable operation of a circuit. Noise evaluation involves many factors. Noise is a probabilistic phenomenon that depends upon the direction of the signal transients. With inductive noise, the large number of lines that can couple into a victim line requires a statistical study of the behavior of the integrated circuits. After determining the directions of the wires switching for worst case and average noise, simulation methods are needed to evaluate the noise on a victim line. Elmore type techniques for fast, reasonably accurate noise

estimations can be a possible solution to the problem. Other more accurate simulation methods can possibly be investigated for simulating circuits with inductive and capacitive coupling.

7. Investigating the effects of process parameter variations and sensitivity analysis of VLSI circuits to inductance.

   The manufacturing process of integrated circuits cannot precisely realize the circuit parameters characterizing a technology. There are always errors between the nominal design parameters (such as line widths and lengths, channel widths and lengths, and threshold voltages) and the actual manufactured parameters. Designs that exhibit lower sensitivity to process parameter variations are more reliable and produce higher yield. The goal of this research task is to provide methods for evaluating the sensitivity of integrated circuits to process parameter variations as well as to develop design methodologies for VLSI circuits that are less sensitive to process parameter variations.

8. Investigating repeater insertion considering inductive noise

   Traditionally, the three design parameters to be optimized in a VLSI design methodology are area, power, and delay. Currently a fourth design parameter, noise, is becoming more important. Increasing noise in VLSI circuits due to the decreasing supply voltage, physically closer circuitry, and higher operating frequencies necessitates the development of design

methodologies that are noise conscious. This design for noise (DFN) prespective is the central goal of this research task.

9. Investigating the effects of on-chip inductance on the power distribution network in terms of the reliability and performance of CMOS circuits

   The power distribution network is composed of wide highly inductive wires which usually carry high levels of current which switch at very high frequencies. Such high speed changes in the large currents carried by these inductive wires causes large $L\ di/dt$ voltage drops within the power distribution network. Thus, the power supply voltage supplied to the CMOS circuitry is not constant and can exhibit large fluctuations when the circuit switches. Such fluctuations can result in reliability problems, affecting the performance of high speed VLSI circuits. Therefore, the goal of this research task is to characterize the changes of voltage within the power distribution network and to design CMOS circuitry that tolerate these fluctuations in the supply voltage.

10. Investigating simplified ways to avoid computationally expensive methods to for extracting on-chip interconnect inductance

    Exact inductance extraction is significantly more computationally expensive than resistance or capacitance extraction due to the non-localized nature of inductance. The resistance of a line depends primarily on the geometric characteristics of the line while the line capacitance depends on the geometries of the line and the adjacent neighboring lines. However, the

inductance of a line depends strongly on the current return path which can partially exist in the substrate and partially in the power and ground lines. The power and ground lines that constitute the current return path for a specific line can be hundreds of micrometers away. Therefore, inductance extraction requires two- or preferably three-dimensional information characterizing a large physical window around the line, making inductance extraction computationally expensive. This research task focuses on deriving effective heuristics to roughly determine the inductance of an interconnect by exploiting the behavior that inductance is a slowly varying function of the interconnect width (see Chapter 13).

Bibliography

[1] J. M. Rabaey, *Digital Integrated Circuits, A Design Perspective*, Prentice Hall, Inc., New Jersey, 1996.

[2] N. H. E. Weste and K. Eshraghian, *Principles of CMOS VLSI Design*, Addison-Wesley Publishing Co., New York, 1993.

[3] S. M. Kang and Y. Leblebici, *CMOS Digital Integrated Circuits Design and Analysis*, McGraw-Hill, Inc., New York, 1996.

[4] F. M. Wanlass and C.-T. Sah, "Nanowatt Logic Using Field-Effect Metal-Oxide Semiconductor Triodes," *Proceedings of the IEEE International Solid-State Circuit Conference*, pp. 32 - 33, February 1963.

[5] F. M. Wanlass "Low Stand-by Power Complementary Field Effect Circuitry," U.S. Patent 3,356,858, issued December 5, 1967.

[6] H. J. M. Veendrick, "Short-Circuit Dissipation of Static CMOS Circuitry and its Impact on the Design of Buffer Circuits," *IEEE Journal of Solid-State Circuits*, Vol. SC-19, No. 4, pp. 468 - 473, August 1984.

[7] S. R. Vemuru and N. Scheinberg "Short-Circuit Power Dissipation Estimation for CMOS Logic Gates," *IEEE Transactions on Circuits and Systems*, Vol. CAS-41, No. 11, pp. 762 - 765, November 1994.

[8] Y. I. Ismail, E. G. Friedman, and J. L. Neves," Dynamic and Short-Circuit Power of CMOS Gates Driving Lossless Transmission Lines," *Proceedings of the IEEE Great Lakes Symposium on VLSI*, pp. 39-44, February 1998.

[9] S. Bothra, B. Rogers, M. Kellam, and C. M. Osburn "Analysis of the Effects of Scaling on Interconnect Delay in ULSI Circuits," *IEEE Transactions on Electron Devices*, Vol. ED-40, No. 3, pp. 591 - 597, March 1993.

[10] T. Sakurai and K. Tamaru, "Simple Formulas for Two- and Three-Dimensional Capacitances," *IEEE Transactions on Electron Devices*, Vol. ED-30, No. 2, pp. 183 - 185, February 1983.

[11] E. Barke, "Line-to-Ground Capacitance Calculation for VLSI: A Comparison," *IEEE Transactions on Computer-Aided Design*, Vol. CAD-7, No. 2, pp. 295 - 298, February 1988.

[12] J. Chern, J. Huang, L. Arledge, P. Li, and P. Yang, "Multilevel Metal capacitance Models for CAD Design Synthesis Systems," *IEEE Electron Device Letters*, Vol. EDL-13, No. 1, pp. 32 - 34, January 1992.

[13] H. G. Lin and L. W. Linholm "An Optimized Output Stage for MOS Integrated Circuits," *IEEE Journal of Solid-State Circuits*, Vol. SC-10, No. 2, pp. 106 - 109, April 1975.

[14] R. C. Jaeger "Comments on 'An Optimized Output Stage for MOS Integrated Circuits'," *IEEE Journal of Solid-State Circuits*, Vol. SC-10, No. 2, pp. 185 - 186, June 1975.

[15] N. Hedenstierna and K. O. Jeppson "CMOS Circuit Speed and Buffer Optimization," *IEEE Transactions on Computer-Aided Design*, Vol. CAD-6, No. 2, pp. 270 - 281, March 1987.

[16] C. Prunty and L. Gal "Optimum Tapered Buffer," *IEEE Journal of Solid-State Circuits*, Vol. SC-27, No. 1, pp. 118 - 119, January 1992.

[17] N. Hedenstierna and K. O. Jeppson "Comments on the 'Optimum CMOS Tapered Buffer Problem'," *IEEE Journal of Solid-State Circuits*, Vol. SC-29, No. 2, pp. 155 - 158, February 1994.

[18] J. S. Choi and K. Lee "Design of CMOS Tapered Buffer for Minimum Power-Delay Product," *IEEE Journal of Solid-State Circuits*, Vol. SC-29, No. 9, pp. 1142 - 1145, September 1994.

[19] B. S. Cherkauer and E. G. Friedman "A Unified Design Methodology for CMOS Tapered Buffers," *IEEE Journal of Solid-State Circuits*, Vol. SC-30, No. 2, pp. 151 - 155, February 1995.

[20] B. S. Cherkauer and E. G. Friedman "Design of Tapered Buffers with Local Interconnect Capacitance," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Vol. 3, No. 1, pp. 99 - 111, March 1995.

[21] M. Horowitz and R. W. Dutton, "Resistance Extraction from Mask Layout Data," *IEEE Transactions on Computer-Aided Design*, Vol. CAD-2, No. 3, pp. 145 - 150, July 1983.

[22] R. J. Antinone and G, W. Brown, "The Modeling of Resistive Interconnects for Integrated Circuits," *IEEE Journal of Solid-State Circuits*, Vol. SC-18, No. 2, pp. 200 - 203, April 1983.

[23] T. Sakurai, "Approximation of Wiring Delay in MOSFET LSI," *IEEE Journal of Solid-State Circuits*, Vol. SC-18, No. 4, pp. 418 - 426, August 1983.

[24] J. Cong, L. He, C-K. Koh, and P. Madden, "Performance Optimization of VLSI Interconnect," *Integration, The VLSI Journal*, Vol. 21, pp. 1 - 94, November 1996.

[25] J. Cong and K. S. Leung, "Optimal Sire Sizing Under the Distributed Elmore

Delay Model," *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, pp. 634 - 639, November 1993.

[26] J. Cong and C-K. Koh, "Simultaneous Driver and Wire Sizing for Performance and Power Optimization," *IEEE Transactions on Very Large Scale Integration (VLSI)*, Vol. 2, No. 4, pp. 408 - 423, December 1994.

[27] K. D. Boese, A. B. Kahng, and G. Robins, "High Performance Routing Trees with Identified Critical Sinks," *Proceedings of the IEEE/ACM Design Automation Conference*, pp. 182 – 187, June 1993.

[28] K. D. Boese, A. B. Kahng, B. A. McCoy, and G. Robins, "Rectilinear Steiner Trees with Minimum Elmore Delay," *Proceedings of the IEEE/ACM Design Automation Conference*, pp. 381 – 386, June 1994.

[29] S. S. Sapatnekar, "RC Interconnect Optimization Under the Elmore Delay Model," *Proceedings of the IEEE/ACM Design Automation Conference*, pp. 387 – 391, June 1994.

[30] J. Cong and L. He, "Optimal Wire Sizing for Interconnects with Multiple Sources," *Proceedings of the IEEE International Conference on Computer-Aided Design*, pp. 586 – 574, November 1995

[31] K. D. Boese, A. B. Kahng, B. A. McCoy, and G. Robins, "Fidelity and Near-Optimality of Elmore-Based Routing Constructions," *Proceedings of the IEEE International Conference on Computer Design*, pp. 81 – 84, October 1993.

[32] J. Cong, A. B Kahng, C.-K. Koh and C.-W. A. Tsao, "Bounded-Skew Clock and Steiner Routing Under Elmore Delay," *Proceedings of the IEEE International Conference On Computer-Aided Design*, pp. 66 - 71, January 1995.

[33] G. Y. Yacoub, H. Pham, M. Ma, and E. G. Friedman, "A System for Critical Path Analysis Based on Back Annotation and Distributed Interconnect Impedance Models," *Microelectronic Journal*, Vol. 18, No. 3, pp. 21 - 30, June 1988.

[34] E. G. Friedman and J. H. Mulligan, Jr., "Ramp Input Response of *RC* Tree Networks," *Analog Integrated Circuits and Signal Processing*, Vol. 14, No. 1/2, pp. 53-58, September 1997.

[35] J. Rubinstein, P. Penfield, and M. Horowitz, "Signal Delay in *RC* Tree Networks," *IEEE Transactions on Computer-Aided Design*, Vol. CAD-2, No. 3, pp. 202 - 211, July 1983.

[36] T. A. Schreyer "The Effects of Interconnection Parasitics on VLSI Circuit Performance," Technical Report No. G815-3, Stanford University, Stanford, March 1989.

[37] L. V. Ginneken, "Buffer Placement in Distributed RC-tree Networks for Minimal Elmore Delay," *Proceedings of the IEEE International Symposium on Circuits*

and Systems, pp. 865 - 868, May 1990.

[38] V. Adler and E. G. Friedman, "Delay and Power Expressions for a CMOS Inverter Driving a Resistive-Capacitive Load," *Analog Integrated Circuits and Signal Processing*, Vol. 14, No. 1/2, pp. 29 - 39, September 1997.

[39] H. B. Bakoglu and J. D. Meindl, "Optimal Interconnection Circuits for VLSI," *IEEE Transactions on Electron Devices*, Vol. ED-32, No. 5, pp. 903 - 909, May 1985.

[40] H. B. Bakoglu, *Circuits, Interconnections, and Packaging for VLSI*, Addison-Wesley Publishing Company, 1990.

[41] V. Adler and E. G. Friedman, "Repeater Design to Reduce Delay and Power in Resistive Interconnect," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, Vol. CAS-45, No. 5, pp. 607 - 616, May 1998.

[42] V. Adler and E. G. Friedman, "Repeater Design to Reduce Delay and Power in Resistive Interconnect," *Proceedings of the IEEE International Symposium on Circuits and Systems*, pp. 2148 - 2151, June 1997.

[43] C. J. Alpert and A. Devgan, "Wire Segmenting for Improved Buffer Insertion," Proceedings of the IEEE/ACM Design Automation Conference, pp. 649-654, June 1997.

[44] S. Dhar and M. A. Franklin, "Optimum Buffer Circuits for Driving Long Uniform Lines," *IEEE Journal of Solid-State Circuits*, Vol. SC-26, No. 1, pp. 32 - 40, January 1991.

[45] D. A. Priore, "Inductance on Silicon for Sub-Micron CMOS VLSI," *Proceedings of the IEEE Symposium on VLSI Circuits*, pp. 17-18, May 1993.

[46] D. B. Jarvis, "The Effects of Interconnections on High-Speed Logic Circuits," *IEEE Transactions on Electronic Computers*, Vol. EC-10, No. 4, pp. 476 - 487, October 1963.

[47] M. P. May, A. Taflove, and J. Baron, "FD-TD Modeling of Digital Signal Propagation in 3-D Circuits with Passive and Active Loads," *IEEE Transactions on Microwave Theory and Techniques*, Vol. MTT-42, No. 8, pp. 1514 - 1523, August 1994.

[48] Y. Eo and W. R. Eisenstadt, "High-Speed VLSI Interconnect Modeling Based on S-Parameter Measurement," *IEEE Transactions on Components, Hybrids, and Manufacturing Technology*, Vol. CHMT-16, No. 5, pp. 555 - 562, August 1993.

[49] A. Deutsch, *et al.*, "High-Speed Signal Propagation on lossy transmission lines," *IBM Journal of Research and Development*, Vol. 34, No. 4, pp. 601 - 615, July 1990.

[50] A. Deutsch, *et al.*, "Modeling and Characterization of Long Interconnections for

High-Performance Microprocessors," *IBM Journal of Research and Development*, Vol. 39, No. 5, pp. 547 - 667, September 1995.

[51] A. Deutsch, *et al.*, "When are Transmission-Line Effects Important for On-Chip Interconnections?," *IEEE Transactions on Microwave Theory and Techniques*, Vol. 45, No. 10, pp. 1836 - 1846, October 1997.

[52] M. Shoji, *High-Speed Digital Circuits*, Addison Wesley, Massachusetts, 1996.

[53] A. Duetsch, A. Kopcsay, and G. V. Surovic, "Challenges Raised by Long On-Chip Wiring for CMOS Microprocessors," *Proceedings of the IEEE Topical Meeting on Electrical Performance of Electronic Packaging*, pp. 21 – 23, October 1995.

[54] Y. Massoud, S. Majors, T. Bustami, and J. White, "Layout Techniques for Minimizing On-Chip Interconnect Self Inductance," *Proceedings of the IEEE/ACM Design Automation Conference*, pp. 566 – 571, June 1998.

[55] B. Krauter and S. Mehrotra, "Layout Based Frequency Dependent Inductance and Resistance Extraction for On-Chip Interconnect Timing Analysis," *Proceedings of the IEEE/ACM Design Automation Conference*, pp. 303 – 308, June 1998.

[56] A. Duetsch, *et al.*, "Design Guidelines for Short, Medium, and Long On-Chip Interconnect," *Proceedings of the IEEE Topical Meeting on Electrical Performance of Electronic Packaging*, pp. 30 – 32, October 1996.

[57] J. Torres, "Advanced Copper Interconnections for Silicon CMOS Technologies," *Applied Surface Science*, Vol. 91, No. 1, pp. 112 - 123, October 1995.

[58] H. B. Bakoglu, J. T. Walker, and J. D. Meindl, "A Symmetric Clock-Distribution Tree and Optimized High Speed Interconnections for Reduced Clock Skew in ULSI and WSI Circuits," *Proceedings of the IEEE/ACM International Conference on Computer Design*, pp. 118 – 122, October 1986.

[59] M. Nekili, *et al.*, "Logic Based H-Trees for Large VLSI Processor Arrays: A Novel Skew Modeling and High-Speed Clocking Method," *Proceedings of the IEEE International Conference Microelectronics*, pp. 144 – 147, December 1993.

[60] C. F. Webb *et al.*, "A 400MHz S/390 Microprocessor," *Proceedings of the IEEE International Solid-State Circuits Conference*, pp. 448 – 449, February 1997.

[61] P. J. Restle and A. Duetsch, "Designing the Best Clock Distribution Network," *Proceedings of the IEEE VLSI Circuit Symposium*, pp. 2 – 5, June 1998.

[62] E. G. Friedman, *High Performance Clock Distribution Networks*, Kluwer Academic Publishers, Massachusetts, 1997.

[63] E. G. Friedman, *Clock Distribution Networks in VLSI Circuits and Systems*, IEEE Press, New Jersey, 1995.

[64] W. Bowhill, *et al.*, "Circuit Implementation of a 300-MHz 64-bit Second-

generation CMOS Alpha CPU," *Digital Technical Journal*, Vol. 7, No. 1, pp. 100 - 118, 1995.

[65] L. Sigal, *et al.*, "Circuit Design Techniques for the High-Performance CMOS IBM S/390 Parallel Enterprise Server G4 Microprocessor," *IBM Journal of Research and Development*, Vol. 41, No. 4/5, pp. 489 - 503, July/September 1997.

[66] Y. I. Ismail, E. G. Friedman, and J. L. Neves, "Performance Criteria for Evaluating the Importance of On-Chip Inductance," *Proceedings of the IEEE International Symposium on Circuits and Systems*, pp. 244-247 May 1998.

[67] Y. I. Ismail, E. G. Friedman, and J. L. Neves, "Figures of Merit to Characterize the Importance of On-Chip Inductance," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Vol. 7, No. 4, pp. 442 – 449, December 1999.

[68] Y. I. Ismail, E. G. Friedman, and J. L. Neves, "Figures of Merit to Characterize the Importance of On-Chip Inductance," *Proceedings of the IEEE/ACM Design Automation Conference*, pp. 560-565, June 1998.

[69] K. K. Likharev and V. K. Semenov, "RSFQ Logic/Memory Family: A New Josephson-Junction Technology for Sub-Terahertz-Clock Frequency Digital System," *IEEE Transactions on Applied Superconductivity*, Vol. AS-1, No. 1, pp. 3 - 28, March 1991.

[70] L. N. Dworsky, *Modern Transmission Line Theory and Applications*, John Wiley & Sons, Inc., New York, 1979.

[71] W. H. Hayt, *Engineering Electromagnetics*, McGraw-Hill, Inc., Japan, 1981.

[72] S. Ramo, J. R. Whinnery, and T. V. Duzer, *Fields and waves in Communication Electronics*, John Wiley & Sons, Inc., New Jersey, 1984.

[73] W. C. Elmore, "The Transient Response of Damped Linear Networks," *Journal of Applied Physics*, Vol. 19, pp. 55 - 63, January 1948.

[74] J. L. Wyatt, *Circuit Analysis, Simulation and Design*, Elsevier Science Publishers, North-Holland, 1987.

[75] L. T. Pillage and R. A. Rohrer, "Delay Evaluation with Lumped Linear RLC Interconnect Circuit Models," *Proceedings of the Caltech Conference on VLSI*, pp. 143-158, May 1989.

[76] M. A. Horowitz, "Timing Models for CMOS Circuits," Ph.D. Thesis, Stanford University, January 1984.

[77] L. T. Pillage, R. A. Rohrer, and C. Visweswariah, *Electronic Circuit and System Simulation Methods*, McGraw-Hill, Inc., USA, 1994.

[78] L. T. Pillage and R. A. Rohrer, "Asymptotic Waveform Evaluation for Timing Analysis," *IEEE Transactions on Computer-Aided Design*, Vol. CAD-9, No. 4,

pp. 352 - 366, April 1990.

[79] C. L. Ratzlaff, N. Gopal, and L. T. Pillage, "RICE: Rapid Interconnect Circuit Evaluator," *Proceedings of the IEEE/ACM Design Automation Conference*, pp. 555 – 560, June 1991.

[80] T. K. Tang and M. S. Nakhla, "Analysis of High-Speed VLSI Interconnects Using the Asymptotic Waveform Evaluation Techniques," *IEEE Transactions on Computer-Aided Design*, Vol. CAD-11, No. 3, pp. 341 - 352, March 1992.

[81] S. P. McCormick, *Modeling and Simulation of VLSI Interconnections with Moments*, Ph.D. thesis, Massachusetts Institute of Technology, June 1989.

[82] S. P. McCormick, *Asymptotic Waveform Evaluation for Timing Analysis*, Ph.D. thesis, Carnegie Mellon University, April 1989.

[83] G. A. Baker, Jr., *Essentials of Bade Approximants*, Academic Press, New York, 1975.

[84] G. A. Baker, Jr. and G.-M. P. *Encyclopedia of Mathematics and its Applications*, Addison-Wesley Publishing Co., New York, 1981.

[85] R. F. Brown, "Model Stability in Use of Moments to Estimate Pulse Transfer Functions," *IEEE Electronic Letters*, Vol. 7, pp. 352 - 366, April 1990.

[86] X. Huang, *Pade Approximation of Linear(ized) Circuit Responses*, Ph.D. thesis, November, Carnegie Mellon University, November 1990.

[87] Y. Shamash, "Stable Reduced-Order Models Using Pade Approximations," *IEEE Transactions on Automatic Control*, Vol. AC-19, No. 5, pp. 615 - 616, August 1974.

[88] N. Gopal and L. T. Pillage "Constrained Approximation of Dominant Time Constant(s) in RC Circuit Delay Models," Technical Report No. UT-CERC-TR-LTP91-01, University of Texas at Austin, Austin, Texas, January 1991.

[89] D. F. Anastasakis, N. Gopal, S. Y. Kim, and L. T. Pillage, "On the Stability of Approximations in Asymptotic Waveform Evaluation," *Proceedings of the IEEE/ACM Design Automation Conference*, pp. 207 – 212, June 1992.

[90] L. T. Pillage, "Coping with RC(L) Interconnect Design Headaches," *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, pp. 246 – 253, September 1995.

[91] A. B. Kahng, K. Masuko, and S. Muddu, "Analytical Delay Models for VLSI Interconnects Under Ramp Input," *Proceedings of the IEEE International Conference On Computer-Aided Design*, pp. 30 - 36, November 1996.

[92] C. J. Terman, *Simulation Tools for Digital LSI Design*, Ph.D. thesis, Massachusetts Institute of Technology, September 1983.

[93] C. L. Ratzlaff, *A Fast Algorithm for Computing the Time Moments of RLC Circuits*, Masters thesis, University of Texas at Austin, Austin, Texas, May 1991.

[94] X. Huang, V. Raghavan, and R. Rohrer, "AWEsim: A Program for the efficient analysis for linear(ized) circuits," *Proceedings of the IEEE/ACM International Conference on Computer Aided Design*, pp. 534 – 537, November 1990.

[95] A. Balivada, D. R. Holberg, and L. T. Pillage, "Calculation and Application of Time-Dominant Waveform Sensitivities in Asymptotic Waveform Evaluation," *Proceedings of the IEEE Custom Integrated Circuits Conference*, pp. 4 – 10, May 1991.

[96] N. Gopal, D. P. Neikirk, and L. T. Pillage, "Evaluation of RC-Interconnect Using Moment-Matching Approximations," *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, pp. 74 – 77, November 1991.

[97] P. Feldmann and R. W. Freund, "Efficient Linear Circuit Analysis by Pade Approximation via the Lancozos Process," *IEEE Transactions on Computer-Aided Design*, Vol. CAD-14, No. 5, pp. 639 - 649, May 1995.

[98] J. Kerns, I. L. Wemple, and A. T. Yang, "Stable and Efficient Reduction of Substrate Model Networks Using Congruence Transforms," *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, pp. 207 – 214, November 1995.

[99] P. Feldmann and R. W. Freund, "Reduced-Order Modeling of Large Linear Subcircuits via Block Lanczos Algorithm," *Proceedings of the IEEE/ACM Design Automation Conference*, pp. 474 – 479, June 1995.

[100] M. Silveira, M. Kamon, and J. White, "Efficient Reduced-Order Modeling of Frequency-Dependent Coupling Inductances Associated with 3-D Interconnect Structures," *Proceedings of the IEEE/ACM Design Automation Conference*, pp. 376 – 380, June 1995.

[101] D. L. Boley, "Krylov Space Methods on State-Space Control Models," *Journal of Circuits, Systems, and Signal Processing*, Vol. 13, No. 6, pp. 733 - 758, May 1994.

[102] S. Y. Kim, N. Gopal, and L. T. Pillage, "Time-Domain Macromodels for VLSI Interconnect Analysis," *IEEE Transactions on Computer-Aided Design*, Vol. CAD-13, No. 10, pp. 1257 - 1270, October 1994.

[103] L. M. Silveira, M. Kamon, I. Elfadel, and J. White, "A Coordinate Transformed Arnoldi Algorithm for Generating Guaranteed Stable Reduced-Order Models of Arbitrary RLC Circuits," *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, pp. 288 – 294, November 1996.

[104] I. M. Elfadel and D. D. Ling, "Zeros and Passivity of Arnoldi-Reduced-Order Models for Interconnect Networks," *Proceedings of the IEEE/ACM Design*

Automation Conference, pp. 28 – 33, June 1997.

[105] K. J. Kerns and A. T. Yang, "Preservation of Passivity During RLC Network Reduction via Split Congruence Transformations," *Proceedings of the IEEE/ACM Design Automation Conference*, pp. 34 – 39, June 1997.

[106] A. Odabasioglu, M. Celik, and L. T. Pillage, "PRIMA: Passive Reduced-Order Interconnect Macromodeling Algorithm," *IEEE Transactions on Computer-Aided Design*, Vol. CAD-17, No. 8, pp. 645 - 654, August 1998.

[107] A. Odabasioglu, M. Celik, and L. T. Pillage, "PRIMA: Passive Reduced-Order Interconnect Macromodeling Algorithm," *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, pp. 58 – 65, November 1997.

[108] T. Sakurai and A. R. Newton "Alpha-Power Law MOSFET Model and its Applications to CMOS Inverter Delay and Other Formulas," *IEEE Journal of Solid-State Circuits*, Vol. SC-25, No. 2, pp. 584 - 593, April 1990.

[109] Y. I. Ismail and E. G. Friedman, "Optimum Repeater Insertion Based on a CMOS Delay Model for On-Chip *RLC* Interconnect," *Proceedings of* the *IEEE ASIC Conference*, pp. 369-373, September 1998.

[110] Y. I. Ismail and E. G. Friedman, "Effects of Inductance on the Propagation Delay and Repeater Insertion in VLSI Circuits," *Proceedings of the ACM/IEEE Design Automation Conference*, pp. 721-724, June 1999.

[111] Y. I. Ismail and E. G. Friedman, "Effects of Inductance on the Propagation Delay and Repeater Insertion in VLSI Circuits," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* (in press).

[112] Y. I. Ismail and E. G. Friedman, "Repeater Insertion in *RLC* Lines for Minimum Propagation Delay," *Proceedings of the IEEE International Symposium on Circuits and Systems*, pp. 404-407, May 1999.

[113] Y. I. Ismail, E. G. Friedman, and Jose L. Neves, "Equivalent Elmore Delay for *RLC* Trees," *Proceedings of the ACM/IEEE Design Automation Conference*, pp. 715-720, June 1999.

[114] Y. I. Ismail, E. G. Friedman, and J. L. Neves, "Equivalent Elmore Delay for *RLC* Trees," *IEEE Transactions on Computer-Aided Design*, Vol. 19, No. 1, pp. 83 - 97 January 2000.

[115] Y. I. Ismail, E. G. Friedman, and J. L. Neves, "Signal Waveform Characterization in *RLC* Trees," *Proceedings of the IEEE International Symposium on Circuits and Systems*, pp. 190-193, May 1999.

[116] Y. I. Ismail, E. G. Friedman, and J. L. Neves, "Inductance Effects in *RLC* Trees," *Proceedings of the IEEE Great Lakes Symposium on VLSI*, pp. 56-59, March 1999

[117] Y. I. Ismail, E. G. Friedman, and J. L. Neves, "Repeater Insertion in Tree Structured Inductive Interconnect," *Proceedings of the ACM/IEEE International Conference on Computer-Aided Design*, pp. 420-424, November 1999.

[118] Y. I. Ismail, E. G. Friedman, and J. L. Neves, "Optimizing *RLC* Tree Delays by Employing Repeater Insertion," *Proceedings of the IEEE ASIC Conference*, pp. 14-18, September 1999.

[119] Y. I. Ismail, E. G. Friedman, and J. L. Neves, "Dynamic and Short-Circuit Power of CMOS Gates Driving Lossless Transmission Lines," *IEEE Transactions on Circuits and Systems 1: Fundamental Theory and Applications*, Vol. CAS-46, No. 8, pp. 950 - 961, August 1999.

[120] Y. I. Ismail, E. G. Friedman, and J. L. Neves, "Power Dissipated by CMOS Gates Driving Lossless Transmission Lines," *Proceedings of the IEEE International Symposium on Low-Power Electronics and Design*, pp. 139-141, August 1998.

[121] Y. I. Ismail, E. G. Friedman, and J. L. Neves, "Transient Power in CMOS Gates Driving *LC* Transmission Lines," *Proceedings of the IEEE International Conference on Electronics, Circuits and Systems*, pp. 377- 383, September 1998.

[122] Y. I. Ismail and E. G. Friedman, "Fast and Accurate Method for Simulating VLSI Interconnect," *Patent Application Pending*.

[123] Y. I. Ismail and E. G. Friedman "Sensitivity of Interconnect Delay to On-Chip Inductance," *Proceedings of the IEEE International Symposium on Circuits and Systems*, May 2000 (in press).

[124] J. J. Cherry, "Pearl: A CMOS Timing Analyzer," *Proceedings of the Design Automation Conference*, pp. 148-153, June 1988 (in press).

[125] T. G. Szymanski, "LEADOUT: A Static Timing Analyzer for MOS Circuits," *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, pp. 130 – 133, September 1986.

[126] J. P. Fishburn, "Optimization-Based Calibration of a Static Timing Analyzer to Path Delay Measurements," *Proceedings of the IEEE International Symposium on Circuits and Systems*, pp. 186-189, May 1999.

[127] L. W. Nagel, "SPICE2: A Computer Program to Simulate Semiconductor Circuits," *Technical Report ERL-M520*, University of California, Berkeley, May 1975

[128] *AS/X User's Guide*, IBM Corporation, New York, 1996.

[129] C.A. Desoer and E. S. Kuh, *Basic Circuit Theory*, McGraw-Hill, Inc., New York, 1969.

[130] B. J. Ley, S. G. Lutz, and C. F. Rehberg, *Linear Circuit Analysis*, McGraw-Hill,

Inc., New York, 1959.

[131] G. F. Paskusz and B. Bussell, *Linear Circuit Analysis*, Prentice Hall, Inc., New Jersey, 1963.

[132] R. E. Scott and W. Essigman, *Linear Circuits*, Addison-Wesley Publishing Co., Mass., 1960.

[133] E. Weber, *Linear Transient Analysis*, Vol. II, John Wiley & Sons, Inc., New York, 1956.

[134] A. B. Kahng and S. Muddu, "An Analytical Delay Model for *RLC* Interconnects," *IEEE Transactions on Computer-Aided Design*, Vol. CAD-16, No. 12, pp. 1507 - 1514, December 1997.

[135] B. C. Kuo, *Automatic Control Systems, A Design Perspective*, Prentice Hall of India, New Delhi, 1989.

[136] P. Feldmann and R. W. Freund, "Reduced-Order Modeling of Large Passive Linear Circuits by Means of the SyPVL Algorithm," *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, pp. 280 – 287, November 1996.

[137] P. Rabinowitz, *Numerical Methods for Non-Linear Algebraic Equations*, Gordon and Breach Science Publishers, London, England, 1970.

[138] M. Daehlen and A. Tveito, *Numerical Methods and Software Tools in Industrial Mathematics*, Birkhauser, Boston, 1997.

[139] S. Winograd, *Arithmetic Complexity of Computations*, J. W. Arrowsmith Ltd., Bristol, England, 1980.

[140] R. C. Agarwal and J. W. Cooley, "New Algorithms for Digital Convolution," *IEEE Transactions on Acoustics, Speech and Signal Processing*, Vol. 25, No. 5, pp. 392 - 410, May 1995.

[141] A. V. Aho, J. E. Hopcroft, and J. D. Ullman, *The Design and Analysis of Computer Algorithms*, Addison-Wesley Publishing Co., Mass., 1976.

Appendix A - Industrial Values for ζ and $T_{L/R}$

For a current 0.25 μm CMOS technology, experimentally measured interconnect parameters (R, L, and C) are provided in [51] for different line widths and are listed here. These line parameters are used in this dissertation to evaluate ζ and $T_{L/R}$ for different line geometries as shown in Table A.2. The data listed in Table A.2 also include the effect of the driver output impedance and the load capacitance on ζ and $T_{L/R}$. h represents the size of the driver and the load gates (assumed to be of equal size) and is with respect to a minimum size buffer. Thus, $R_{tr} = R_0/h$ and $C_L = hC_0$. Note that $T_{L/R}$ is independent of the length of the wire. Note also that the values of ζ are significantly less than one for common width wires, implying that significant error in the propagation delay will be incurred. The values indicated for $T_{L/R}$ demonstrate that large error can be encountered in the circuit design process if an *RC* model rather than an *RLC* model is used.

Table A.1. Interconnect parameters for different line widths [51].

Width (μm)	R (Ω/cm)	L (nH/cm)	C (pF/cm)
0.9	494	4.75	1.73
1.8	248	3.70	1.85
2.4	76	5.30	2.60
7.5	35	3.47	5.16

Table A.2. ζ and $T_{L/R}$ for different line widths and lengths in a current 0.25 μm CMOS technology. $R_0 = 2000\ \Omega$ and $C_0 = 4$ fF.

Line width (μm)	Buffer width h	ζ					$T_{L/R}$
		Line length (mm)					
		2	4	6	8	10	
0.9	40	1.327	1.770	2.235	2.702	3.171	1.096
	80	1.299	1.790	2.272	2.750	3.226	
	120	1.397	1.930	2.443	2.930	3.422	
	240	1.743	2.426	3.015	3.562	4.087	
1.8	40	1.101	1.337	1.600	1.870	2.143	1.366
	80	0.936	1.200	1.473	1.749	2.026	
	120	0.940	1.233	1.519	1.803	2.085	
	240	1.082	1.456	1.79	2.104	2.407	
2.4	40	0.752	0.800	0.871	0.949	1.029	2.952
	80	0.498	0.554	0.628	0.707	0.788	
	120	0.429	0.491	0.568	0.648	0.732	
	240	0.390	0.473	0.560	0.647	0.733	
7.5	40	1.118	1.151	1.206	1.268	1.332	3.525
	80	0.647	0.683	0.739	0.801	0.865	
	120	0.497	0.535	0.592	0.654	0.719	
	240	0.362	0.410	0.470	0.535	0.600	

Appendix B - Optimum Repeater Insertion in *RLC* Lines

As shown in section 6.1, the propagation delay of a gate driving a single section of interconnect with parameters of R_t, C_t, and L_t has the form given by (6.16). If repeaters are inserted to divide the line into k sections and each repeater is h times greater than a minimum size inverter, the total propagation delay of the system is the summation of the propagation delays of each of the sections. Since the delay of each section is equal, the total delay can be expressed as $t_{pdtotal} = kt_{pdsec}$, where t_{pdsec} is the propagation delay of a single section. Each section has interconnect parameters equal to R_t / k, C_t / k, and L_t / k. Since each repeater is h times larger than a minimum size buffer, each repeater has an output resistance $R_{tr} = R_0 / h$ and a load capacitance $C_L = C_0 h$. Thus, the total propagation delay of the repeater system is

$$t_{pdtotal} = k \cdot \frac{t_{pd}(\zeta_{\sec}, R_{T\sec}, C_{T\sec})}{\omega_{n\sec}}, \tag{B.1}$$

where R_{Tsec} and C_{Tsec} are

$$R_{T\sec} = \frac{k}{h}\frac{R_0}{R_t}, \tag{B.2}$$

$$C_{T\sec} = kh\frac{C_0}{C_t}. \tag{B.3}$$

ζ_{sec} and ω_{nsec} are

$$\zeta_{\sec} = \frac{R_t}{2k}\sqrt{\frac{C_t}{L_t}} \cdot \frac{R_{T\sec} + C_{T\sec} + R_{T\sec}C_{T\sec} + 0.5}{\sqrt{(1 + C_{T\sec})}}, \tag{B.4}$$

$$\omega_{n\sec} = \frac{k}{\sqrt{L_t C_t}\sqrt{1+C_{T\sec}}}. \tag{B.5}$$

The solution for the general case of an *RLC* interconnect is in the form of

$$h = \sqrt{\frac{R_0 C_t}{R_t C_0}} \bullet h', \tag{B.6}$$

$$k = \sqrt{\frac{R_t C_t}{2R_0 C_0}} \bullet k', \tag{B.7}$$

where h' and k' are error factors due to the existence of inductance and approach one as the inductance approaches zero. Substituting these values for h and k into (B.2), (B.3), (B.4), and (B.5), the variables $R_{T\sec}$, $C_{T\sec}$, $\zeta_{\sec}$, and $\omega_{n\sec}$ are

$$R_{T\sec} = \frac{k'}{h'\sqrt{2}}, \tag{B.8}$$

$$C_{T\sec} = \frac{h'k'}{\sqrt{2}}, \tag{B.9}$$

$$\zeta_{\sec} = \frac{1}{\sqrt{2}k'T_{L/R}} \cdot \frac{R_{T\sec} + C_{T\sec} + R_{T\sec}C_{T\sec} + 0.5}{\sqrt{(1+C_{T\sec})}}, \tag{B.10}$$

and

$$\frac{k}{\omega_{n\sec}} = \sqrt{L_t C_t}\sqrt{(1+C_{T\sec})}, \tag{B.11}$$

where $T_{L/R}$ is given by

$$T_{L/R} = \sqrt{\frac{L_t / R_t}{R_0 C_0}}. \tag{B.12}$$

Substituting (B.8)-(B.11) in (B.1), the total propagation delay has the form,

$$t_{pdtotal} = \sqrt{L_t C_t} \cdot f(h', k', T_{L/R}). \tag{B.13}$$

Determining the values of k' and h' that minimize the total propagation delay requires the simultaneous solution of the following two differential equations,

$$\frac{\partial f(h', k', T_{L/R})}{\partial h'} = 0, \tag{B.14}$$

$$\frac{\partial f(h', k', T_{L/R})}{\partial k'} = 0. \tag{B.15}$$

Thus, the optimum number of sections k_{opt} and the optimum repeater size h_{opt} to minimize the propagation delay of an *RLC* interconnect are only functions of $T_{L/R}$ and are

$$h_{opt} = \sqrt{\frac{R_0 C_t}{R_t C_0}} \bullet h'(T_{L/R}) \tag{B.16}$$

$$k_{opt} = \sqrt{\frac{R_t C_t}{2R_0 C_0}} \bullet k'(T_{L/R}). \tag{B.17}$$

Note that this solution is characteristic of an *RLC* line and that no approximations have been made in deriving this result.

Appendix C - Complexity o f the Equivalent Elmore Delay

Referring to (7.2), (7.18), and (7.19), the second order approximate transfer function at node *i* of an *RLC* tree is

$$g_i(s) = \frac{1}{\left[\sum_k C_k L_{ik}\right] s^2 + \left[\sum_k C_k R_{ik}\right] s + 1}. \tag{C.1}$$

Thus, evaluating this transfer function for all of the nodes of an *RLC* tree requires the calculation of the following two summations,

$$T_{RCi} = \sum_k C_k R_{ik}\,, \tag{C.2}$$

$$T_{LCi}^2 = \sum_k C_k L_{ik}\,, \tag{C.3}$$

for all of the nodes of the *RLC* tree. These two summations can be rewritten as

$$T_{RCi} = \sum_k C_{Tk} R_k\,, \tag{C.4}$$

$$T_{LCi}^2 = \sum_k C_{Tk} L_k\,, \tag{C.5}$$

where the summation index k operates over all of the *RLC* sections that belong to the path from the input to node i. R_k and L_k are the resistance and inductance of section k. C_{Tk} is the total load capacitance seen by R_k and L_k. For example, as shown in Figure 7.3, $T_{RC7} = R_1(C_1+C_2+\ldots+C_7) + R_6(C_3+C_6+C_7) + R_7C_7$. This form of expressing the summations is convenient since it has recursive properties [37], [93].

The summations in (C.4) and (C.5) of a tree rooted at section w_1 are calculated in two steps. The first step is calculating the total load capacitance seen at each section. Pseudo-code that performs this task is described in Figure C.1.

```
float Cal_Cap_Loads (section w)
{
        if(right(w)=0 and left(w)=0)    /* if w is a leaf */
                return w.C;

        if(right(w)≠0)
                C_TR=Cal_Cap_Loads(right(w));
        else
                C_TR=0;       /* No right branch is driven by w */

        if(left(w)≠0)
                C_TL=Cal_Cap_Loads(left(w));
        else
                C_TL=0;       /* No left branch is driven by w */

        w.C_T=C_TR+C_TL;

        return w.C_T;
}
```

Figure C.1. Pseudo-code for calculating the total load capacitance at each section

The function is initially called by Cal_Cap_Loads(w_1) and recursively calculates the capacitive load at each section. $w.C$ is the capacitance of the section w. The functions, left(w) and right(w), return the left and right sections driven by w, respectively. If no left (right) section is driven by w, left(w)=0 (right(w)=0). If w is a leaf, left(w)=0 and right(w)=0. The time required to calculate the total capacitive loads is proportional to the number of *RLC* sections in the tree, m, and requires no multiplication operations. Note that a binary branching factor is assumed without loss of generality since any

general tree can be transformed into a binary tree by inserting wires with zero impedances [37], [38].

The second step is to calculate and store the summations, (C.4) and (C.5), at the nodes of the tree. The function performing this task is described in Figure C.2. The function is initially called by Cal_Summations(w_1,0,0). $w.R$ and $w.L$ are the resistance and inductance of section w, respectively. The computational time required to calculate the summations is proportional to the number of *RLC* sections in the tree, m. The total number of multiplications required to evaluate the second order approximation at all of the nodes of an *RLC* tree is $2m$. Alternatively, the number of multiplications is equal to the order of the characteristic equation describing the *RLC* tree since the order of an *RLC* tree with m *RLC* sections is $2m$ (each *RLC* section has an inductor and a capacitor).

```
Cal_Summations(section w, float T_Rcprev, float T_LCprev)
{
        T_RC=T_Rcprev+w.R*w.C_T;
        T_LC=T_Rcprev+w.L*w.C_T;

        if(right(w)≠0)
                Cal_ Summations (right(w),T_RC,T_LC);

        if(left(w)≠0)
                Cal_ Summations (left(w),T_RC,T_LC);
}
```

Figure C.2. Pseudo-code for calculating the delays at the sinks of an *RLC* tree.

Appendix D - Matching Conditions of a CMOS Gate Driving a Lossless Transmission Line

To calculate the transistor widths that matches the output impedances of the transistors to the characteristic impedance of a lossless transmission line, an understanding of the nature of the signal propagation across a lossless transmission line is necessary. When the transistor is first turned on, the transmission line appears as a resistor with a value equal to the characteristic impedance of the line Z_0. This situation is shown in Figure D.1. The initial voltage wave that is launched into the transmission line ($V_{initial}$) can be determined by equating the current from the transistor with the current through Z_0. The gate-to-source voltage of the PMOS transistor is $-V_{DD}$ and the drain-to-source voltage is $V_{initial}$-V_{DD}. Thus, the current sourced by the transistor is a function of V_{DD} and $V_{initial}$. $V_{initial}$ can be calculated from

$$I_{ds}[V_{DD}, V_{initial}] = \frac{V_{initial}}{Z_0}, \tag{D.1}$$

where $I_{ds}[V_{DD}, V_{initial}]$ is the transistor output current and is a function of V_{DD} and $V_{initial}$.

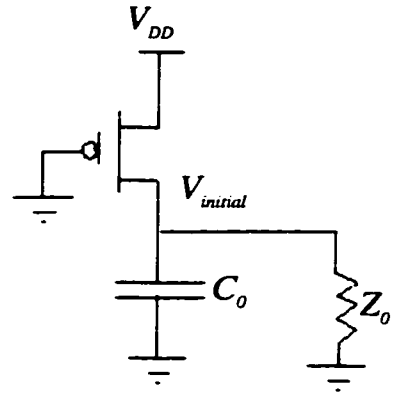


Figure D.1. A PMOS transistor driving a lossless transmission line. This circuit is the equivalent circuit of a CMOS inverter driving a lossless transmission line for the period of time $0 < t < 2T_0$.

The signal at the input of the transmission line does not reach the value $V_{initial}$ directly since the output capacitor in parallel with Z_0 requires some time to charge to $V_{initial}$. This time is dependent on the intrinsic delay of the technology and is typically smaller than twice the time of flight of the signals propagating across the transmission line, thereby permitting the output capacitance to be neglected. The signal propagates along the transmission line and takes a time T_0 to reach the load impedance. At the load, the signal reaches a steady state value of twice $V_{initial}$ due to the reflection at the load capacitance [36]. This voltage doubling occurs since the capacitor appears as an open circuit at steady state. Assuming that the incident wave $V_{initial}$ is a step input, the voltage across the load capacitor is [36]

$$V_c = 2V_{initial}\left[1 - e^{\left(\frac{-t}{Z_0 C_L}\right)}\right]. \qquad \text{(D.2)}$$

Thus, the time required to reach a steady state value of $2V_{initial}$ depends on the time constant Z_0C_L. If this time constant is much less than $2T_0$, the effect of the load

capacitance is negligible. Typically, this time constant is in the order of a few picoseconds and is sufficiently small to neglect the effects of the load capacitance (see Chapter 6).

The condition for matching is launching an initial voltage $V_{initial} = V_{DD}/2$. This voltage wave propagates across the line and is totally reflected at the load after a period of time T_0. The reflected wave of magnitude $V_{DD}/2$ propagates from the load back towards the transistor. As this wave propagates back towards the source, the signal adds to the initial voltage wave, charging the line to V_{DD}. When this reflected wave reaches the transistor at time $2T_0$, the drain voltage of the PMOS transistor reaches V_{DD} and V_{DSp} reaches zero volts. Thus, the transistor no longer conducts any current and the system becomes stable. Waveforms describing this condition is illustrated in Figure D.2 for a CMOS inverter driving a lossless transmission line with V_{DD} = 5 volts.

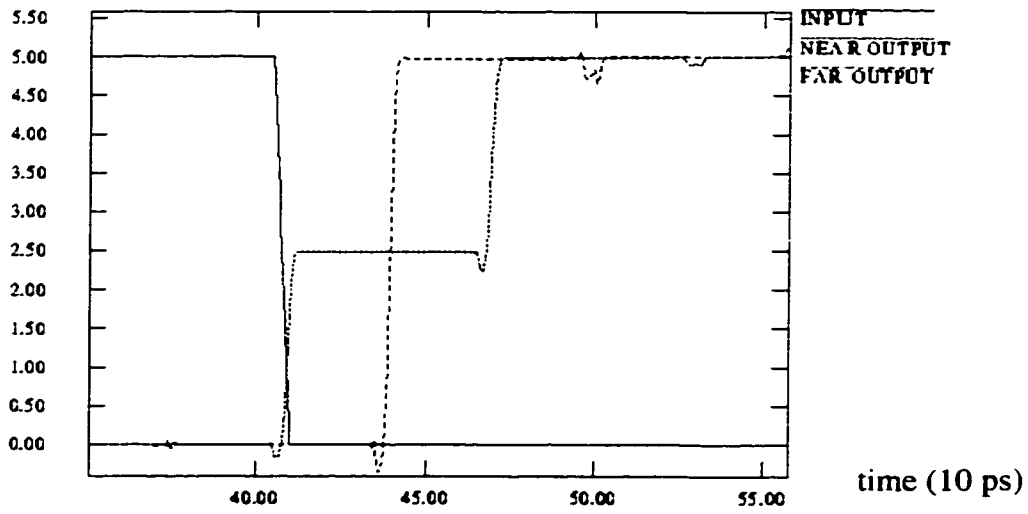


Figure D.2. AS/X simulations of a matched inverter driving an ideal transmission line.

To determine the widths of the transistors required to satisfy the matching condition, (D.1) is solved with $V_{initial} = V_{DD}/2$. Using the alpha power law model to characterize the MOS transistors in the saturation region, (D.1) evaluates to

$$P_{Cp}\frac{W_p}{L_p}\left(V_{DD}-\left|V_{Tp}\right|\right)^{\alpha_p}=\frac{V_{DD}}{2Z_0}. \tag{D.3}$$

The geometric widths of the transistors of a CMOS inverter that satisfy the matched condition are

$$W_p=\frac{V_{DD}}{2Z_0S_p}, \tag{D.4}$$

$$W_n=\frac{V_{DD}}{2Z_0S_n}, \tag{D.5}$$

where S_p and S_n are technology dependent parameters given by

$$S_p=\frac{P_{Cp}}{L_p}\left(V_{DD}-\left|V_{Tp}\right|\right)^{\alpha_p}, \tag{D.6}$$

$$S_n=\frac{P_{Cn}}{L_n}\left(V_{DD}-V_{Tn}\right)^{\alpha_n}. \tag{D.7}$$

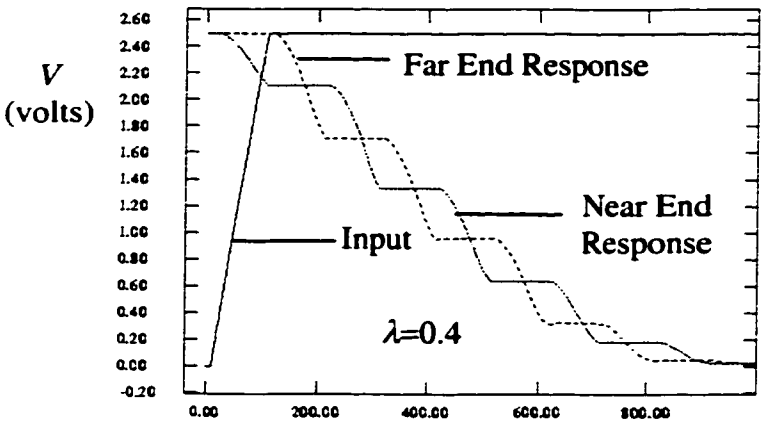
The assumption that the transistors operate in the saturation region is correct for deep submicrometer technologies because of the early saturation phenomenon [108]. The AS/X simulations in Figure D.2 demonstrate the accuracy of these equations and exhibit near perfect matching between the MOS transistors and the lossless (or low loss) transmission lines.

A useful parameter that characterizes the input/output relationship of a CMOS gate driving a lossless transmission line can be defined base on (D.4) for P-channel and N-channel transistors, respectively, as

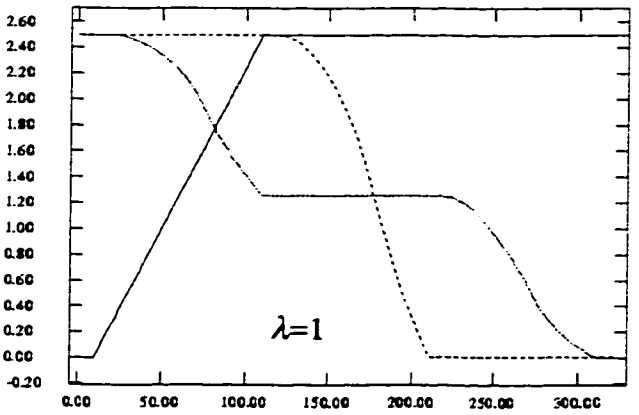
$$\lambda_p=\frac{2S_pW_pZ_0}{V_{DD}}, \tag{D.8}$$

$$\lambda_n = \frac{2S_n W_n Z_0}{V_{DD}}. \tag{D.9}$$

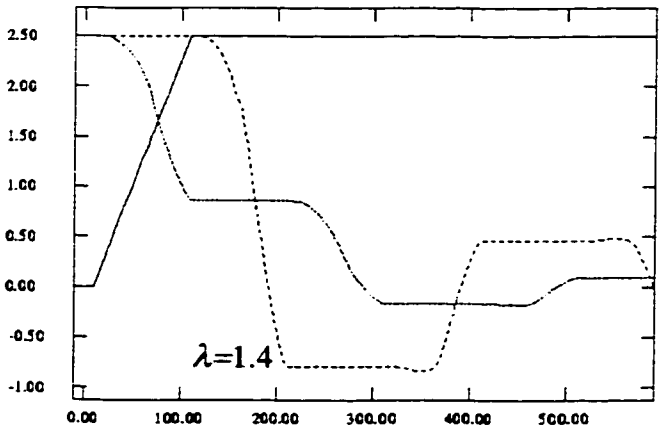
λ_p characterizes the response to a low-to-high output transition and λ_n characterizes the response to a high-to-low output transition. For a symmetric gate, $\lambda_p = \lambda_n$ and the CMOS inverter is simply characterized by one parameter λ. It can be seen that $\lambda = 1$ is the matched condition defined by (D.4). If λ is greater than one, wider transistors are used compared to the matched widths and the output response is overdriven, producing overshoots and undershoots. If λ is less than one, smaller transistor widths are used compared to the matched widths and the output response is underdriven or a sluggish response occurs. AS/X simulations are shown in Figure D.3 for several values of λ, depicting the underdriven, matched, and overdriven cases. The widths of the transistors are varied and λ is calculated according to (10.17). Note that λ accurately characterizes the output response as described above.



(a) Underdriven case: Transistor sizes are smaller than needed for matching



(b) Matched case: Fine tuning of the widths of the transistors is required to avoid reflections



(c) Overdriven case: Transistor sizes are larger than needed for matching

Figure D.3 AS/X simulations of a CMOS inverter driving an ideal transmission line for several values of λ, depicting the underdriven, matched, and overdriven cases.

Appendix E - The DTT Algorithm

A general *RLC* tree is composed of several connected *RLC* sections. Each *RLC* section has a series resistance, inductance, and capacitance with the capacitance grounded as shown in Figure 12.2. The objective is to calculate the transfer functions across all of the capacitors in the *RLC* tree. A function to calculate the common denominator of an *RLC* tree rooted at the *RLC* section w_i is Cal_Denominator which uses the DTT algorithm as explained in section 12.1. Pseudo-code that performs this task is described in Figure E.1.

```
Cal_Denominator (section* w)
{
        if(right(w)=0)                  /* there is no right section driven by w */
                {Dr=1; Mr=0;}
        else                            /* there is a right section driven by w */
                {Cal_Denominator(right(w)); Dr=right(w)->D; Mr=right(w)->M;}

        if(left(w)=0)                   /* there is no left section driven by w */
                {Dl=1; Ml=0;}
        else                            /* there is a left section driven by w */
                {Cal_Denominator(left(w)); Dl=left(w)->D; Ml=left(w)->M;}

        w->N = Dl•Dr;
        w->M = w->C*w->N + Ml•Dr + Mr•Dl;
        w->D = w->N + (w->M)•[(w->R)*s+(w->L)*s^2];
}
```

Figure E.1. Pseudo-code for calculating the common denominator of an *RLC* tree

The function is initially called by Cal_Denominator(w_1) and recursively calculates the common denominator. The structure "section" has the elements *R*, *L*, and *C*, which represent the resistance, inductance, and capacitance of an *RLC* section, respectively. The structure also has the arrays *N*, *M*, and *D*, which represent the polynomials of the numerator, *M* in (12.19), and the denominator of the transfer function across the capacitor of the *RLC* section, respectively. The operator "•" represents a polynomial multiplication. An efficient limited order polynomial multiplication function should be used as discussed in subsection 12.1.3. The functions, left(w) and right(w), return pointers to the left and right sections driven by w, respectively. If no left (right) section is driven by w, left(w)=0 (right(w)=0). The function uses (12.14), (12.19), and (12.15). The recursion termination conditions are described by the DTT method in subsection 12.1.2.

The second step is to correct the numerators of the transfer functions at the nodes of the *RLC* tree. The function performing this task is described in Figure E.2. The function is initially called by Correct_Numerators(w_1,1) and recursively corrects the numerators at all of the nodes of the *RLC* tree as described in subsection 12.1.2. Note that the Correct_Numerators function has to be called after the Cal_Denominator function has been called.

```
Correct_Numerators(section *w, Poly F_in)
{
        if(right(w)≠0)                    /* w drives a right section */
                {F_r = F_in•D_l; Correct_Numerator(right(w), F_r);}

        if(left(w)≠0)                     /* w drives a left section */
                {F_l = F_in•D_r; Correct_Numerator(left(w), F_l);}

        w->N = w->N•F_in;
}
```

Figure E.2. Pseudo-code for correcting the numerators of the transfer functions at all of the nodes of an *RLC* tree.

Appendix F - Comparison between DTT and AWE

Two methods have been discussed for accurately simulating the transient response of *RLC* circuits. The first method is AWE and is presented in section 3.2. The second method is DTT, which is discussed in Chapter 12. These two methods are compared in this appendix in terms of accuracy, stability, numerical stability, and computational speed. The primary similarities and differences between the two methods are listed in Table F.1.

Table F.1 Comparison between DTT and AWE

Comparison Criteria	DTT	AWE
1- Accuracy	SPICE like accuracy for any waveform	Cannot accurately simulate highly complicated signals due to lack of sufficient poles
2- Stability	Stable for approximations with less than five poles	Can be unstable for any approximation order
3- Numerical Stability	Numerically stable with any approximation order	Unstable with high approximation orders due to ill conditioned matrices
4- Maximum number of poles	Unlimited	Limited to eight or ten poles due to numerical errors and lack of extra information in higher order moments
5- Calculation of residues	Direct substitution	Requires the solution of an ill conditioned system of linear equations at each node
6- Common set of poles	Yes	Poles have to be recalculated at each node for better accuracy. Some forced methods exist such as moment shifting, but accuracy decreases
7- Complexity with number of elements in a tree	Linear	Linear
8- Complexity with approximation order	Almost linear (q)	Cubic (q^3)
9- Speed	Faster	Slower

Accuracy

A DTT approximation of order $2q$ has the same or better accuracy as compared to a q order AWE approximation. To explain this trait, note that $2q$ moments are needed to calculate an AWE approximation of order q. These $2q$ moments are matched to the moments of the reduced order system as described in section 3.2.1. A $2q$ order DTT approximation as given by (F.1) matches the first $2q$ moments of the original system since higher powers of s than $2q$ in the numerator and denominator cannot produce powers of s less than or equal to $2q$ in the series expansion.

$$T_{2q}(s) = \frac{1 + a_1 s + a_2 s^2 + \ldots + a_x s^{2q-1}}{1 + b_1 s + b_2 s^2 + \ldots + b_q s^{2q}} = 1 + (a_1 - b_1)s + \cdots . \tag{F.1}$$

By matching the first $2q$ moments, AWE uses information about the first $2q$ coefficients of s in the numerator and denominator embedded in the moments. Hence, it is appropriate to compare a q^{th} order AWE to a $2q^{th}$ order DTT in terms of accuracy. It is shown later in this appendix that DTT is significantly faster than AWE despite using double the approximation order. In addition, DTT has other important advantages over AWE as discussed later. To compare the accuracy of DTT and AWE, a figure of merit is introduced that sums the differences between equidistant samples of the exact signal and the approximate signal (determined by DTT or AWE) which is given by

$$\sigma = \sum_i \left| Sactual_i - Sapprox_i \right|, \tag{F.2}$$

where $Sactual_i$ represents the i^{th} sample of the exact signal, $Sapprox_i$ represents the i^{th} sample of the approximate signal, and the summation index i takes integer values in the range where the error is being evaluated. This range is selected to span the time

from the beginning of the transient response to the time when the signal reaches a steady state. The distance between adjacent samples is unimportant as long as sufficient precision is achieved and the same distance is used for both DTT and AWE. The accuracy of DTT and AWE is compared using several examples from section 12.3 listed in Table F.2. Note that a DTT approximation consistently provides higher accuracy as compared to an AWE approximation with half the order. This improved accuracy can be intuitively understood by noting that although both approximations match the same number of moments, the DTT solution contains more information about the exact form of the original transfer function since DTT matches the coefficients of the numerator and the denominator in addition to the moments. For example, by matching the numerator and the denominator of the original transfer function in addition to matching the moments, low DTT approximation orders are guaranteed to be stable. This characteristic is not true with AWE. Also, DTT has much better numerical stability with high order approximations which accounts for the significantly higher accuracy with high orders as listed in Table F.2.

Table F.2 Comparison of the accuracy of DTT and AWE using several examples from section 12.3. The figure of merit given by (F.2) is used for the comparison with the upper row of each example representing σ_{DTT} and the lower row representing σ_{AWE}.

q_{AWE}	2	4	6	8	10	12	14	16
q_{DTT}	4	8	12	16	20	24	28	32
Figure 12.10	0.018008	0.000215	0.000001	$7.7 \cdot 10^{-9}$	$2.2 \cdot 10^{-10}$	$1.5 \cdot 10^{-10}$	$1.4 \cdot 10^{-10}$	$1.3 \cdot 10^{-10}$
	0.095736	0.000223	11.69304	8.147014	4.268515	1.5098	8.32889	4.66800
Figure 12.11 Node O_4	0.89499	0.098930	0.00316	0	-	-	-	-
	11.89326	0.113261	0.00614	0.00142	-	-	-	-
Figure 12.14	63.30909	44.83124	26.42833	24.56688	18.66098	15.90539	11.58347	11.38745
	68.40276	63.24460	28.09890	24.75239	53.36206	80.35241	47.13582	58.66605
Figure 12.15 (a)	20.61302	10.60683	5.621289	3.174961	3.093296	2.096618	1.696990	1.746895
	25.58750	16.87190	12.22447	3.202715	38.42453	63.67413	26.99634	15.64751
Figure 12.15 (b)	30.29888	17.61748	8.911335	5.385857	5.046698	3.498040	2.791696	2.902061
	43.20270	29.42135	19.46530	6.815113	25.51021	53.90268	39.93335	29.99762

Stability

In terms of relative pole stability of the two approximation algorithms, as described in section 12.2, the DTT method is guaranteed to be stable for approximations with less than five poles. This characteristic is extremely useful with *RC* circuits which exhibit monotone responses and can be accurately simulated with few poles. For example, a second order AWE approximation of the *RC* circuit shown in Figure 3.8 is shown in section 3.2.3 to have one unstable pole. This instability causes significant accuracy loss as illustrated in Figure 3.9. A fourth order DTT approximation of the signal at output O_4 shown in Figure 3.8 is compared to a second order AWE and SPICE simulation in Figure F.1. Note that the guaranteed stability makes DTT significantly more accurate as compared to AWE for low approximation orders. According to the information listed in Table F.2, $\sigma_{DTT} = 0.89499$ and $\sigma_{AWE} = 11.89326$ for a fourth order DTT and a second order AWE, respectively.

Alternatevily, the added stability of DTT permits DTT to be 13 times more accurate than AWE.

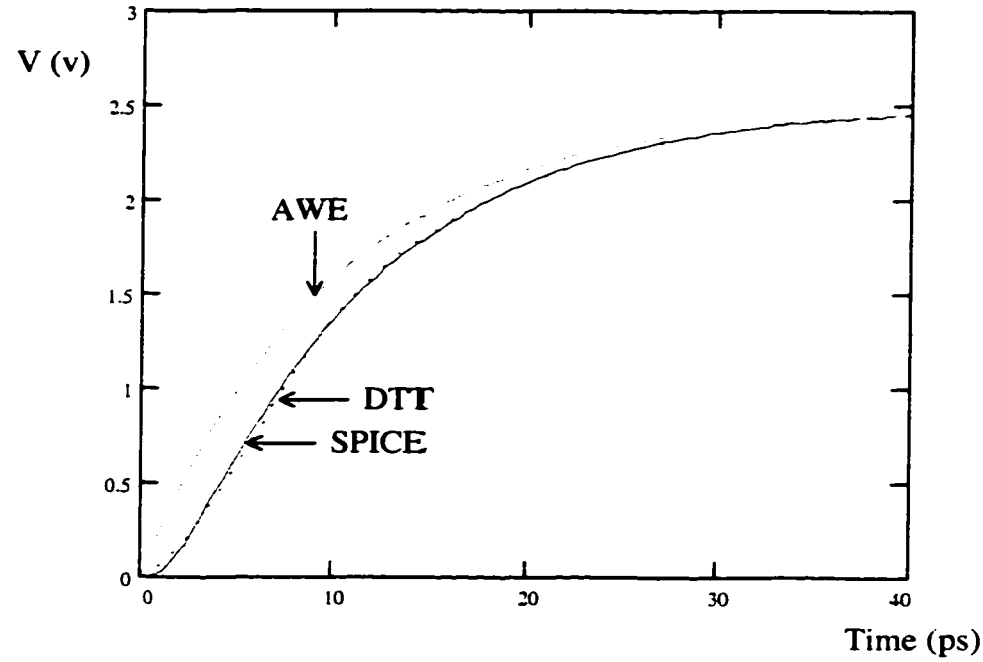


Figure F.1 Comparison between a fourth order DTT approximation, a second order AWE approximation, and SPICE of the signal at output O_4 of the *RC* circuit shown in Figure 3.8.

Numerical Stability

As discussed in section 3.2.3, AWE suffers from numerical inaccuracies with high order approximations because 1) the higher order moments do not contain additional information characterizing the poles and 2) due to the solution of ill-conditioned systems of linear equations when determining the poles and residues. The moments of the original system are related to the poles and residues of the system by

$$
\begin{aligned}
m_0 &= -(\frac{k_1}{p_1} + \frac{k_2}{p_2} + ... + \frac{k_n}{p_n}), \\
m_1 &= -(\frac{k_1}{p_1^2} + \frac{k_2}{p_2^2} + ... + \frac{k_n}{p_n^2}), \\
&\vdots \\
m_{2n-1} &= -(\frac{k_1}{p_1^{2n}} + \frac{k_2}{p_2^{2n}} + ... + \frac{k_n}{p_n^{2n}}).
\end{aligned}
\tag{F.3}
$$

where n is the order of the original system (see section 3.2.3). Thus, poles with larger magnitudes are truncated when added to dominant poles with smaller magnitudes in higher order moments due to the addition of poles raised to large powers. This behavior, in addition to the need to invert ill-conditioned matrices, renders AWE incapable of calculating higher order approximations to simulate complicated waveforms. In comparison, however, the DTT algorithm determines the poles by solving the polynomial given by

$$D_q(s) = 1 + b_1 s + b_2 s^2 + ... + b_q s^q , \tag{F.4}$$

where b_1 - b_q are the first exact q coefficients of the complete denominator given by

$$D(s) = 1 + b_1 s + b_2 s^2 + ... + b_n s^n . \tag{F.5}$$

These coefficients are related to the poles of the system as given by

$$
\begin{aligned}
b_1 &= -\sum_{i=1}^{n} \frac{1}{p_i} \\
b_2 &= \sum_{j=1}^{n} \sum_{k=j+1}^{n} \frac{1}{p_j p_k} \\
b_3 &= -\sum_{i=1}^{n} \sum_{j=i+1}^{n} \sum_{k=j+1}^{n} \frac{1}{p_i p_j p_k} \\
&\vdots
\end{aligned}
\tag{F.6}
$$

Note that lager magnitude poles are multiplied by smaller magnitude poles in all of the terms of the coefficients b_2 and higher. This relation between the denominator coefficients and the poles permits the poles to be determined with much larger magnitudes than the poles which AWE is capable of determining through the moments. Also, the DTT method determines the residues by direct substitution as described in section 12.1.4 and the denominator is directly determined through reliable polynomial multiplication operations. Thus, DTT does not require the solution of any system of linear equations. These characteristics make DTT capable of calculating much higher approximation orders than AWE. These higher order approximations are necessary for accurately simulating complicated non-monotone waveforms. For example, as shown in section 3.2.3 in the simulations illustrated in Figure 3.12, AWE is incapable of accurately calculating the transient response of the circuit deposited in Figure 3.11, producing a best case relative error in the rise time of 167%. This same example, re-simulated using DTT as shown in Figure 12.14 in section 12.3, exhibits high accuracy. Note also in Table F.2 that the accuracy of the AWE method consistently deteriorates beyond eight poles primarily due to the numerical issues discussed above. In comparison, the accuracy of the DTT method monotonically increases with approximation order.

Computational Speed

In terms of computational speed, DTT is significantly faster than AWE. The increased computational performance of DTT is due to several reasons. The DTT method determines the common denominator and the numerators of order $2q$ at different nodes faster than AWE finds the $2q$ moments. The reason for the increased speed of DTT in determining these polynomials is explained in section 12.2. The

second reason is that DTT determines a common denominator for the entire circuit while AWE determines a different denominator at each specific node where the transient response is required. To determine the poles at a node, AWE requires the solution of a system of q linear equations to find a polynomial of order q followed by the application of iterative methods to determine the zeros of this polynomial which represent the poles (see section 3.2.1). The complexity of this process is proportional to q^3. The third reason is that AWE solves a set of q linear equations to determine the residues at each node where the transient response is required which also has a complexity proportional to q^3. As discussed in section 12.1.4, the residues are determined in DTT by direct substitution into the numerator. Note also that the residues need only be determined in DTT for the stable accurate poles and that conjugate poles have conjugate residues, permitting the calculation of only one residue of each conjugate residue pair. Such selective calculation of the residues is not possible in the AWE method (see section 3.2.1). Thus, DTT performs computationally much more efficiently than AWE when a transient response is required at many nodes.

To quantitatively characterize the computational performance of DTT as compared to AWE, two circuits are investigated. The first circuit is that of a single distributed *RLC* line composed of 40 π sections. A transient response is required at only one node which is the output (or termination) of the line. The second circuit is that of a binary tree with ten levels (*i.e.*, 1024 nodes and 512 outputs). The comparison is between an AWE system of order q and a DTT system of order $2q$. This comparison guarantees a DTT approximation with the same or higher accuracy than AWE. The ratio of the computational time required by AWE to the

computational time required by DTT to simulate the transmission line circuit is plotted in Figure F.2 for different approximation orders. The numbers shown on the x-axis of Figure F.2 is the order of AWE. The order of DTT is twice that of AWE for each point (not shown in the figure). Note that DTT is significantly faster than AWE, particularly at higher approximation orders. Since the transient response is only required at one node, the performance advantage of DTT for the first circuit is primarily because the DTT method determines the common denominator and the numerators of order $2q$ at different nodes faster than AWE finds the $2q$ moments.

The ratio of the computational time required by AWE to the computational time required by DTT to simulate the *RLC* tree circuit is plotted in Figure F.3 for different approximation orders. Note that the performance advantage of DTT as compared to AWE in this second circuit is much greater as compared to the first case. The higher performance is achieved in the second circuit because a transient response is required at many nodes (half the total number of nodes), permitting the superior computational performance of DTT to be dramaticly demonstrated.

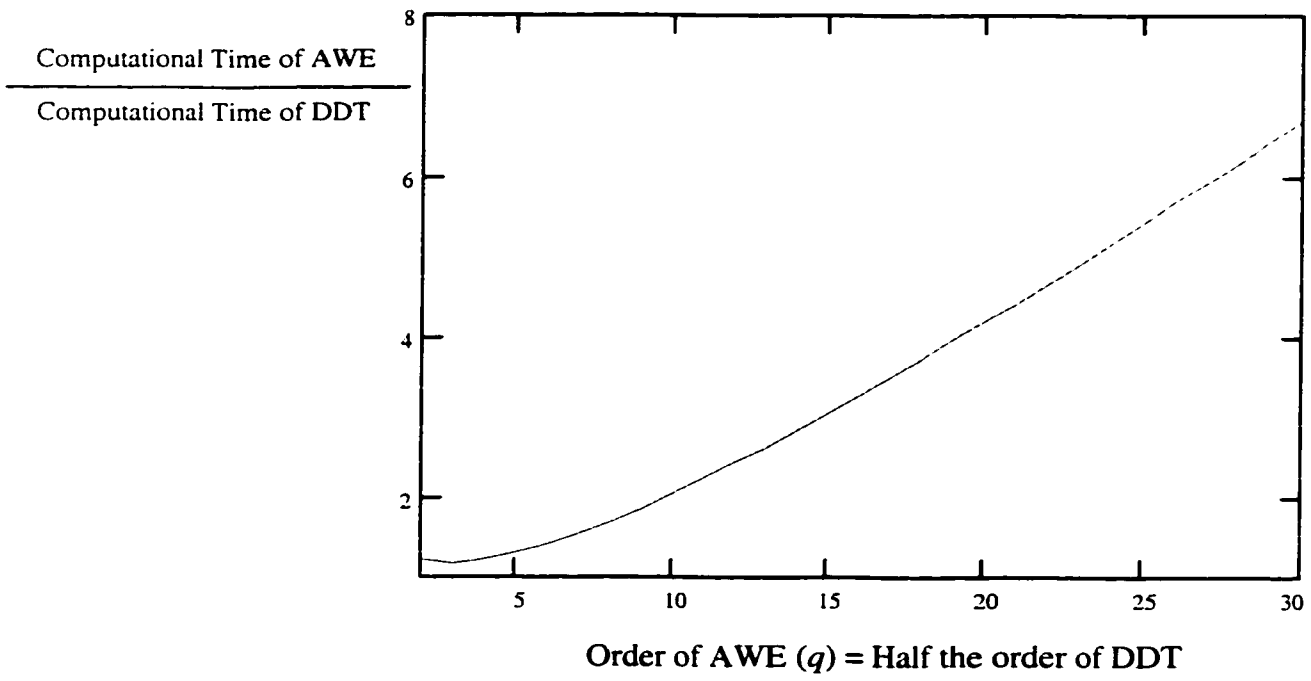


Figure F.2 The ratio of the computational time required by AWE to the computational time required by DTT to simulate the transient response at the output node of an *RLC* distributed line versus the approximation order. The numbers shown on the x-axis is the order of AWE. The order of DTT is twice the order of AWE at each point.

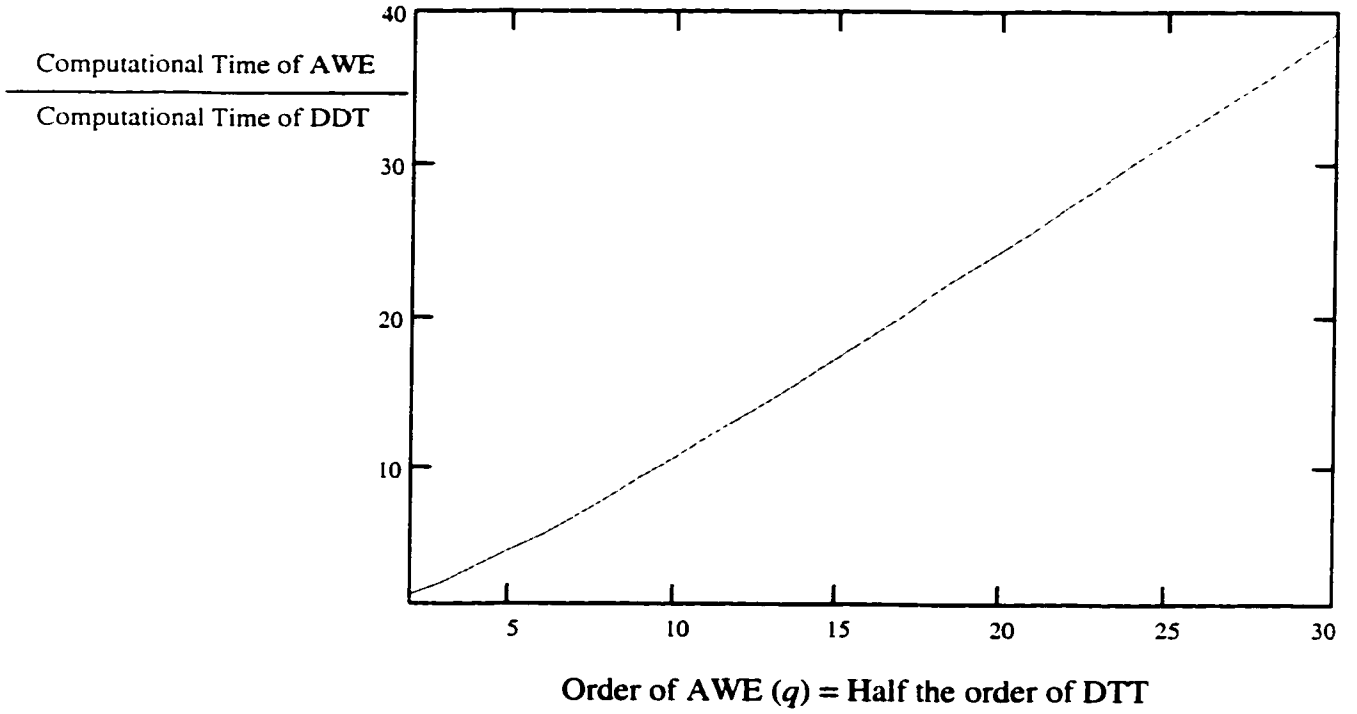


Figure F.3 The ratio of the computational time required by AWE to the computational time required by DTT to simulate the transient response at 512 output nodes of a binary balanced *RLC* tree versus the approximation order. The numbers shown on the x-axis is the order of AWE. The order of DTT is twice the order of AWE at each point.

Appendix G - Publications and Patents

Patent Disclosures

1. Y. I. Ismail, E. G. Friedman, and J. L. Neves "Driving Inductive Interconnect Using Cascaded Buffers" (patent pending).
2. Y. I. Ismail and E. G. Friedman, "DTT: Direct Truncation of Transfer Function. An Alternative to Moment Matching Techniques" (patent pending).

Refereed Journal Publications

1. Y. I. Ismail and E. G. Friedman, "Effects of Inductance on the Propagation Delay and Repeater Insertion in VLSI Circuits," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* (in press).
2. Y. I. Ismail, E. G. Friedman, and J. L. Neves, "Equivalent Elmore Delay for *RLC* Trees," *IEEE Transactions on Computer-Aided Design*, Vol. 19, No. 1, pp. 83 - 97 January 2000.
3. Y. I. Ismail, E. G. Friedman, and J. L. Neves, "Figures of Merit to Characterize the Importance of On-Chip Inductance," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Vol. 7, No. 4, pp. 442 – 449, December 1999.
4. Y. I. Ismail, E. G. Friedman, and J. L. Neves, "Dynamic and Short-Circuit Power of CMOS Gates Driving Lossless Transmission Lines," *IEEE Transactions on Circuits and Systems 1: Fundamental Theory and Applications*, Vol. CAS-46, No. 8, pp. 950 - 961, August 1999.

Submitted Journal Papers

5. Y. I. Ismail, E. G. Friedman, and J. L. Neves, "Inductance Effects in *RLC* Trees," *IEEE Transactions on Circuits and Systems 1: Fundamental Theory and Applications* (in submission).
6. Y. I. Ismail, E. G. Friedman, and J. L. Neves, "Repeater Insertion in Tree Structured Inductive Interconnect," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing* (in submission).
7. Y. I. Ismail and E. G. Friedman "On the Extraction of On-Chip Inductance," *IEEE Transactions on Computer-Aided Design* (in submission).
8. Y. I. Ismail, E. G. Friedman, and J. L. Neves, "Exploiting On-Chip Inductance in High Speed Clock Distribution Networks," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* (in submission).
9. Y. I. Ismail and E. G. Friedman "DTT: Direct Truncation of Transfer Function. An Alternative to Moment Matching for Tree Structured Interconnect," *IEEE Transactions on Computer-Aided Design* (in submission).

Refereed Conference Publications

10. Y. I. Ismail and E. G. Friedman "Sensitivity of Interconnect Delay to On-Chip Inductance," *Proceedings of the IEEE International Symposium on Circuits and Systems*, May 2000 (in press).
11. Y. I. Ismail, E. G. Friedman, and J. L. Neves, "Repeater Insertion in Tree Structured Inductive Interconnect," *Proceedings of the ACM/IEEE International Conference on Computer-Aided Design*, pp. 420-424, November 1999.
12. Y. I. Ismail, E. G. Friedman, and J. L. Neves, "Optimizing *RLC* Tree Delays by Employing Repeater Insertion," *Proceedings of the IEEE ASIC Conference*, pp. 14-18, September 1999.
13. Y. I. Ismail, E. G. Friedman, and Jose L. Neves, "Equivalent Elmore Delay for *RLC* Trees," *Proceedings of the ACM/IEEE Design Automation Conference*, pp. 715-720, June 1999.
14. Y. I. Ismail and E. G. Friedman, "Effects of Inductance on the Propagation Delay and Repeater Insertion in VLSI Circuits," *Proceedings of the ACM/IEEE Design Automation Conference*, pp. 721-724, June 1999.
15. Y. I. Ismail, E. G. Friedman, and J. L. Neves, "Signal Waveform Characterization in *RLC* Trees," *Proceedings of the IEEE International Symposium on Circuits and Systems*, pp. 190-193, May 1999.
16. Y. I. Ismail and E. G. Friedman, "Repeater Insertion in *RLC* Lines for Minimum Propagation Delay," *Proceedings of the IEEE International Symposium on Circuits and Systems*, pp. 404-407, May 1999.
17. Y. I. Ismail, E. G. Friedman, and J. L. Neves, "Inductance Effects in *RLC* Trees," *Proceedings of the IEEE Great Lakes Symposium on VLSI*, pp. 56-59, March 1999.
18. Y. I. Ismail, E. G. Friedman, and J. L. Neves, "Transient Power in CMOS Gates Driving *LC* Transmission Lines," *Proceedings of the IEEE International Conference on Electronics, Circuits, and Systems*, pp. 377- 383, September 1998.
19. Y. I. Ismail and E. G. Friedman, "Optimum Repeater Insertion Based on a CMOS Delay Model for On-Chip *RLC* Interconnect," *Proceedings of the IEEE ASIC Conference*, pp. 369-373, September 1998.
20. Y. I. Ismail, E. G. Friedman, and J. L. Neves, "Power dissipated by CMOS Gates Driving Lossless Transmission Lines," *Proceedings of the IEEE International Symposium on Low-Power Electronics and Design*, pp. 139-141, August 1998.
21. Y. I. Ismail, E. G. Friedman, and Jose L. Neves, "Figures of Merit to Characterize the Importance of On-Chip Inductance," *Proceedings of the IEEE/ACM Design Automation Conference*, pp. 560-565, June 1998.
22. Y. I. Ismail, E. G. Friedman, and J. L. Neves, "Performance Criteria for Evaluating the Importance of On-Chip Inductance," *Proceedings of the IEEE International Symposium on Circuits and Systems*, pp. 244-247, May 1998.
23. Y. I. Ismail, E. G. Friedman, and J. L. Neves, "Dynamic and Short-Circuit Power of CMOS Gates Driving Lossless Transmission Lines," *Proceedings of the IEEE Great Lakes Symposium on VLSI*, pp. 39-44, February 1998.