12-11-1991

A Multiple Coupled Microstrip Transmission Line Model for High-Speed VLSI Interconnect Simulation

Lawrence Stevan Uzelac
Portland State University

Recommended Citation
Uzelac, Lawrence Stevan, "A Multiple Coupled Microstrip Transmission Line Model for High-Speed VLSI Interconnect Simulation" (1991). *Dissertations and Theses.* Paper 4526.
https://doi.org/10.15760/etd.6410

AN ABSTRACT OF THE THESIS OF Lawrence Stevan Uzelac for the

Master of Science in Electrical Engineering presented December 11th, 1991.

Title: A Multiple Coupled Microstrip Transmission Line Model for High-Speed VLSI Interconnect Simulation

APPROVED BY THE MEMBERS OF THE THESIS COMMITTEE:

[redacted]

W. Robert Daasch, Chair

[redacted]

Paul Van Halen

[redacted]

Andrew M. Fraser

A model is presented which incorporates the advantages of a mixed mode simulation to characterize transmission line behavior in multiple coupled Transmision line systems. The model is intended for use by digital circuit designers who wish to be able to obtain accurate transmission line behavior for complex digital systems for which continuous time simulation tools such as SPICE would time prohibitive.

The model uses a transverse electromagnetic wave approximation to obtain solutions to the basic transmission line equations. A modal analysis technique is used to solve for the attenuation and propagation constants for the transmission lines. Modal analysis done in the frequency domain after a Fast Fourier Transform of the time-domain input signals. Boundary conditions are obtained from the Thevinized transmission line input equivalent circuit and the transmission line output load impedance.

The model uses a unique solution queue system that allows n-line coupled transmission lines to be solved without resorting to large order matrix methods or the need to diagonals larger matrices using linear transformations. This solution queue system is based on the method of solution superposition. As a result, the CPU time required for the model is primarily a function of the number of transitions and not the number of lines modeled.

Incorporation of the model into event driven circuit simulators such as Network C is discussed. It will be shown that the solution queue methods used in this model make it ideally suited for incorporation into a event-driven simulation network.

The model presented in this thesis can be scaled to incorporate direct electromagnetic coupling between first, second, or third lines adjacent to the line transitioning. It is shown that modeling strictly adjacent line coupling is adequate for typical digital technologies.

It is shown that the model accurately reproduces the transmission line behavior of systems modeled by previous authors. Example transitions

on a 8-line system are reviewed. Finally, future model improvements are discussed.

A MULTIPLE COUPLED MICROSTRIP TRANSMISSION LINE MODEL FOR HIGH-SPEED VLSI INTERCONNECT SIMULATION

by

LAWRENCE STEVAN UZELAC

A thesis submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE
in
ELECTRICAL ENGINEERING

Portland State University
1992

TO THE OFFICE OF GRADUATE STUDIES:

The members of the Committee approve the thesis of Lawrence Stevan Uzelac presented December 11th, 1991.

[redacted]

W. Robert Daasch, Chair

[redacted]

Paul Van Halen

[redacted]

Andrew M. Fraser

APPROVED:

[redacted]

Rolf Schaumann, Chair, Department of Electrical Engineering

[redacted]

C. William Savery, Vice Provost for Graduate Studies and Research

ACKNOWLEDGEMENTS

The author would like to acknowledge the Portland Technology development group of Intel Corporation for use of the computer equipment on which much of the work presented in this thesis was accomplished.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

MICROSTRIP TRANSMISSION LINE MODEL

BACKGROUND AND MODEL DESCRIPTION

Device interconnect has become an increasingly important concern for high speed digital circuit design. As clock rates approach 100 Mhz, metal line pitch decreases to sub-micron dimensions, and die size increases to greater than 1 square centimeter, circuit interconnect will experience line delays and cross-talk of increasing relative proportions. Device reliability and circuit clock rate may be adversely affected by interconnect line delay and line to line cross talk. Modeling this cross talk and line delay based on interconnect topography aids the circuit designer in determining what levels of signal over-shoot and signal under-shoot the input and output drivers on a data bus must be able to tolerate for a given technology. In addition, the layout designer may be limited by the the magnitude of the cross talk allowed for a given circuit design, and must set the interconnect pitch, interconnect metal, or dielectric material accordingly to stay within the specified limits.

Several transmission line models exist that use various techniques to solve the transmission line equations for multiple coupled parallel microstrip lines. These methods include modal analysis in the frequency do-

main [1,2], modal analysis in the time domain [3,4,5], modal analysis combined with linear transformations of the transmission line matrices [6,7], convolution techniques [8,9,10], and finite difference time stepping techniques [11]. However, none of these modeling techniques has been adapted for mixed-mode (digital/analog) circuit simulation. This thesis introduces a transmission line model ideally suited for mixed mode digital/analog circuit simulators. The model incorporates modal analysis in the frequency domain as well as a Fast Fourier Transform technique to convert time domain signals to the frequency domain.

One of the unique features of this model is that it is designed to be incorporated into a mixed mode digital/analog simulation environment. This is accomplished by calculating the numerical solutions to the transmission line equations when triggered by a digital input node transition. Such transitions are typically referred to as "events". Circuit simulators that use both digital and analog signals are referred to as event driven simulators.

Solution superposition is used to allow simulation of multiple line transitions. In this manner the model is able to simulate n-line systems. The fact that the model is event driven and uses solution superposition allow for more efficient simulation of multiple coupled transmission lines than purely analog methods for large scale digital circuit simulation.

Figure 1 illustrates how digital circuit interconnect simulation applies to the process of VLSI digital chip design. Once the logic design is defined, the digital circuit design must take in to account timing limitations associated with the interconnect of digital functional blocks. The process intercon

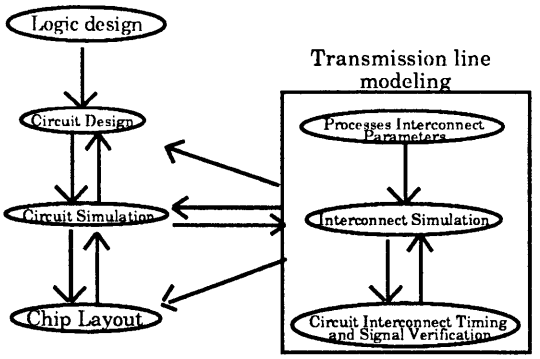

Figure 1. The Digital VLSI design process.

nect parasitic parameters are used during the modeling of the interconnect behavior. The results of the interconnect simulation may require modification of either the circuit design or the circuit layout design to achieve the desired timing goals.

The model presented has several features that make it desirable for event driven simulation. One feature is that the numerical algorithms determine the solution for 3-line subsystems of a larger n-line network. This allows for reduced CPU usage during the numerically intensive portions of the simulator. Solutions for lines not affected by the event are not calculated. Superposition is used to incorporate the three-line solution onto the solutions for a n-line system. In this manner a n-line coupled microstrip system (i.e. a 32 bit data bus) can be modeled for an arbitrary number of transitions. The total number of lines to be modeled and the total number of transitions can be easily scaled. CPU time is primarily a function of the number of transitions and not the number of lines modeled.

If the cross talk or "fan-out" of the electromagnetic coupling between lines affect more than the lines adjacent to the line transitioning, the model

can be scaled from a three line system to a five, seven, or nine line system as necessary to account for the coupling. This unique feature of the model allows the user to simultaneously solve for as many lines as are required for a single transition to completely model systems with more than adjacent line coupling. It will be shown later that three line solutions are adequate for modeling modern on-chip IC interconnect.

Another advantage of the model presented in this thesis is that it is event driven. Node voltage solution values for the input and output nodes of the n-line transmission line system are stored in individual arrays with array pointers controlled by the clock of the simulator. If no transition (or "event") occurs on the input nodes of the transmission line during any given portion of the simulation, no numerical solution is called by the model and the output node voltages are determined by the last steady-state voltage from previous transitions. The model's numerically intensive subroutines are called only during events that result in transition of the input node to the transmission lines. This type of simulation is commonly referred to as event driven simulation. A circuit simulator that takes advantage of mixed mode simulation is Network C [12]. Implementation of this model in Network C is discussed in Chapter III.

The model presented in this thesis also contains disadvantages when compared to other methods of transmission line simulation. All values of resistance, capacitance, and inductance are considered to be frequency and voltage independent. Therefore only linear systems can be modeled. In addition, this model has not taken advantage of all available numerical tech-

niques to reduce the solution calculations. Finally, this model has not taken advantage of programming techniques such as solution caching or dynamic memory allocation to improve CPU usage and memory management. However, there is nothing inherent in the general modeling technique presented that would prevent these limitations from being overcome.

This thesis will discuss the mathematical description of multiple coupled transmission lines, the numerical solution techniques used, and the programming algorithm used to calculate the solutions. CPU efficiency as a function of number of lines solved simultaneously is discussed. A comparison of the results of this model to the work of others is made. The model is then used to simulate a typical 8-line system for multiple transitions. Finally future model enhancements are discussed.

This thesis presents an alternative method for multiple coupled transmission line modeling that is particularly suited for mixed signal digital/analog or "event driven" simulation. The numerical methodology used to determine the solution to the transmission line equations for a given set of input node signals is discussed in detail in chapter II.

CHAPTER II

WAVE BEHAVIOR OF MICROSTRIP LINES

MATHEMATICAL DESCRIPTION

Transmission lines can be considered to be passive circuit elements that contain resistive, capacitive, and inductive components. Transmission line systems that have no net energy loss due to series resistance elements from the input to the output are termed lossless transmission lines. Transmission lines that do contain series resistance elements are termed lossy transmission lines. The model presented in this thesis can calculate solutions for both lossy and lossless transmission line systems.

Transmission line behavior has been characterized using the transverse electromagnetic wave approximation (TEM) [13,14]. The TEM approximation assumes that the transmission line exists in a homogeneous medium such that the signals propagate along the line as transverse electromagnetic waves. This assumption allows the use of Maxwell's equations to derive a mathematical description of the electromagnetic wave propagating along the transmission line. Using Maxwell's equations the following mathematical description of a transmission line is derived:

$$\frac{\delta v(x,t)}{\delta x} = - R(i\,(x,t)) - L\,\frac{\delta i(x,t)}{\delta t} \qquad (1)$$

$$\frac{\delta i(x,t)}{\delta x} = - G(v(x,t)) - C \frac{\delta v(x,t)}{\delta t} \qquad (2)$$

Both voltage and current are a function of position along the transmission line and time. R, L, G, and C represent the transmission line resistance, inductance, admittance and capacitance per unit length and are considered time invariant. Equations (1) and (2) will be used along with the input and output node voltage values to establish a boundary value problem from which the wave equations describing the signal propagation along the transmission line will be derived.

Several techniques have been used to determine the solution to the transmission line equations. These include finite difference methods in the time domain, time domain modal analysis, and frequency domain modal analysis. Frequency domain modal analysis has been chosen in this work due to the efficiency of this approach over time stepping finite difference methods [11]. A detailed description of how the program solves the modal analysis problem is given in Chapter III.

The equivalent circuit representing a single microstrip line is shown in figure 2. The transmission line is represented by a "T" equivalent circuit with distributed inductive, resistive, and capacitive elements. The transmission line is connected to a source and a load. The source end consists of a Thevinized voltage source and series source impedance. The load is represented by an equivalent load impedance. The variable x denotes the spatial distance along the length of the transmission line. For a line of length l, the input voltage value at x=0 and the output voltage value at x=l will be used

as boundary values to establish the solutions to equations (1) and (2).

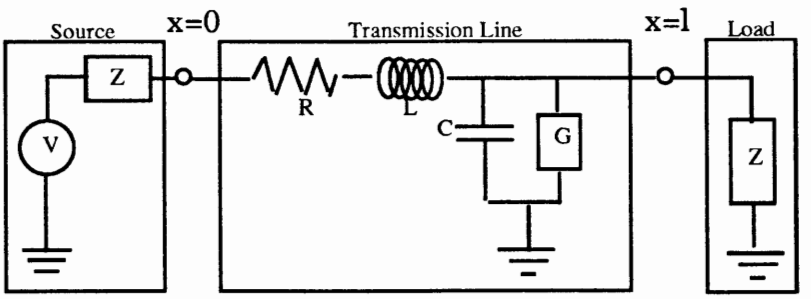

Figure 2. Transmission line equivalent circuit.

Multiple parallel microstrip lines are represented in Figure 3. These lines may be modeled in the same manner as a single microstrip line with the individual elements of the model now represented by a matrix of elements representing both self and mutual resistive, inductive, and capacitive values. This can be mathematically expressed in the form of the basic transmission line equation. The R, L, G, and C values of the transmission line equation are replaced with [R], [L], [G], and [C] matrices with r_{11}, r_{22}, r_{33}, ...r_{nn}, and l_{11}, l_{22}, l_{33}, ... l_{nn}, etc. representing the self resistance, inductance, conductance and capacitance values.

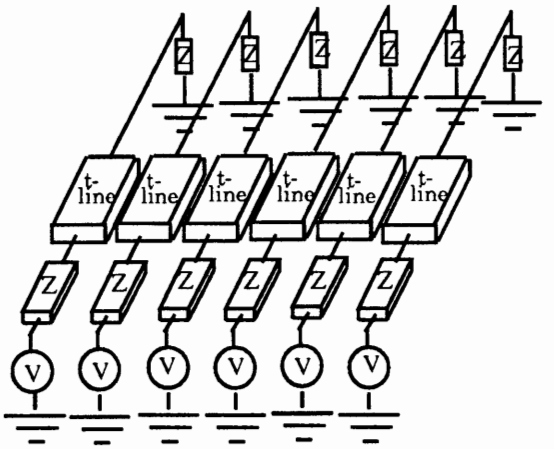

Figure 3. Multiple Mircrostrip Transmission Lines.

The off-diagonal elements r_{nm}, l_{nm}, etc. (n not equal to m) represent the mutual resistance, inductance, conductance, and capacitance values. As stated previously the behavior of the signal being modeled is assumed to be a transverse electromagnetic wave (TEM wave). As such, a exponential wave solution that is both a function of position and time will be used. If this solution is taken to be of the form $e^{j\omega t-\gamma x}$, the transmission line equation can be expressed as a eigenvalue/eigenvector problem.

$$[R] = \begin{bmatrix} r_{11} & r_{12} & r_{13} \cdots & r_{1n} \\ r_{21} & r_{22} & r_{23} \cdots & r_{2n} \\ r_{31} & r_{32} & r_{33} \cdots & r_{3n} \\ \vdots & \vdots & \vdots & \vdots \\ r_{n1} & r_{n2} & r_{n3} \cdots & r_{nn} \end{bmatrix} \qquad [L] = \begin{bmatrix} l_{11} & l_{12} & l_{13} \cdots & l_{1n} \\ l_{21} & l_{22} & l_{23} & l_{2n} \\ l_{31} & l_{32} & l_{33} \cdots & l_{3n} \\ \vdots & \vdots & \vdots & \vdots \\ l_{n1} & l_{n2} & l_{n3} \cdots & l_{nn} \end{bmatrix}$$

$$[G] = \begin{bmatrix} g_{11} & g_{12} & g_{13} \cdots & g_{1n} \\ g_{21} & g_{22} & g_{23} \cdots & g_{2n} \\ g_{31} & g_{32} & g_{33} \cdots & g_{3n} \\ \vdots & \vdots & \vdots & \vdots \\ g_{n1} & g_{n2} & g_{n3} \cdots & g_{nn} \end{bmatrix} \qquad [C] = \begin{bmatrix} c_{11} & c_{12} & c_{13} \cdots & c_{1n} \\ c_{21} & c_{22} & c_{23} \cdots & c_{2n} \\ c_{31} & c_{32} & c_{33} \cdots & c_{3n} \\ \vdots & \vdots & \vdots & \vdots \\ c_{n1} & c_{n2} & c_{n3} \cdots & c_{nn} \end{bmatrix}$$

In matrix form, equations (1) and (2) become:

$$\frac{\delta[v]}{\delta x} = -[R][i] - [L]\frac{\delta[i]}{\delta t} \tag{3}$$

$$\frac{\delta[i]}{\delta x} = -[G][v] - [C]\frac{\delta[v]}{\delta t} \tag{4}$$

Assuming [v(x,t)] and [i(x,t)] have the form:

$$[v(x,t)] = V_0 [e^{j\omega t - \gamma x}] \quad (5)$$

$$[i(x,t)] = I_0 [e^{j\omega t - \gamma x}] \quad (6)$$

with

$$\gamma = \alpha + j\beta$$

γ is a complex number with the real component of γ is termed the wave attenuation constant (α), and the imaginary component of γ represents the wave propagation constant (β). γ will be purely imaginary for lossless transmission lines, and will have both real and imaginary components for lossy lines. The propagation constant represents the wave velocity of the line and will ultimately determine the line delay for a given line length and load.

Next, substituting (5) into (3) and (6) into (4) one obtains:

$$-V_0 = \frac{I_0}{\gamma_n} \{ [R] + j\omega[L] \} \quad (7)$$

$$I_0 = \frac{V_0}{\gamma_n} \{ [G] - j\omega[C] \} \quad (8)$$

Using (7) and (8) to solve for γ:

$$\gamma^2 = \{[R] + j\omega[L]\}\{[G] - j\omega[C]\} \quad (9)$$

which can be translated to:

$$(\lambda[I] + [Y][Z])[S_I] = 0 \quad (10A)$$

$$(\lambda[I] + [Z][Y])[S_V] = 0 \quad (10B)$$

with

$$[I] = \text{identity matrix}$$

$$\lambda = \gamma^2$$

$$[Z] = [R] + j\omega[L]$$

$$[Y] = [G] - j\omega[C]$$

The solution to (10B) consists of a (nxn) voltage eigenvector matrix $[S_V]$ representing n modes of propagation. These "modes" result from the fact that for given a (nxn) matrix [A], there exists n eigenvalues and n eigenvectors that satisfy the basic eigenvalue problem:

$$[A][\varepsilon]_n = \lambda_n[\varepsilon]_n$$

or

$$(\lambda_n[I] - [A])[\varepsilon]_n = 0$$

with

$$[\varepsilon]_n = n^{th} \text{ eigenvector}$$

$$\lambda_n = n^{th} \quad \text{eigenvalue}$$

Physically the n modes can be considered to be describing characteristic transverse electromagnetic waves propagating through the transmission line that are determined by the impedance and admittance properties of the lines. The parameters that determine the modes of propagation for a transmission line system are the [R], [L], [G], and [C] matrices and frequency. The voltage and current eigenvectors are related as follows:

let

$$[\Gamma] = \text{diag}[\gamma_1, \gamma_2, \gamma_3, \ldots \gamma_n]$$

and define Z_c such that:

$$[v(x,t)] = [Z_C][i(x,t)]$$

Then the voltage and current eigenvectors are related by:

$$[Z_C] = [S_V][S_I]^{-1} \tag{11}$$

or

$$[S_I] = [Z_C]^{-1}[S_V][\Gamma] \tag{12}$$

The current eigenvector can be derived from (10A) directly, or from equation (11) once the voltage eigenvector is derived from (10B).

The boundary values for line voltages and currents are shown in equations (13) and (14) as being the summation of the forward traveling and reverse traveling wave solutions. The boundary value problem arises from the need to relate the end-of-line current and voltage values to those values represented by the TEM waves traveling the length of the transmission line. Boundary conditions (voltage and current) for the source end of the circuit are set by nodal analysis of the equivalent circuit network for the source illustrated in figure 3. The object of the boundary value problem is to determine the coefficients $W^+(0)$ and $W^-(0)$ in equations (13) and (14) for the forward traveling and reverse traveling waves at x=0 and x =l.

$$[V(x)] = [S_V]([W^+(0)e^{-\gamma x}] + [W^-(0)e^{+\gamma x}]) \tag{13}$$

$$[I(x)] = [Y_C][S_V]([W^+(0)e^{-\gamma x}] - [W^-(0)e^{+\gamma x}]) \quad (14)$$

with

$$\gamma = (\gamma_1, \gamma_2, \gamma_3, \ldots \gamma_n)$$

and

$$[Y_C] = [Z_C]^{-1}$$

The forward and reverse wave coefficients $W^+(0)$ and $W^-(0)$ are determined from the voltage and current values of the input and load nodes as follows: The input node voltage for the transmission line is the applied input voltage less the voltage drop due to the transmission line input impedance. Therefore the input node voltage on the transmission line is expressed:

$$[V(0)] = [E] - [Z_G][I(0)] \quad (15)$$

[E] is the source voltage matrix, $[Z_G]$ is the series source impedance matrix. The output node voltage of the transmission line is determined by the voltage drop across the load:

$$[V(l)] = [Z_L][I(l)] \quad (16)$$

$[Z_L]$ is the load impedance matrix.

V(0) is the voltage at the transmission line input node, and V(l) is the voltage at the output node of the transmission line. Equations (13), (14), (15), and (16) lead to a set of linear equations from which $W^+(0)$ and $W^-(0)$ are determined.

Substituting the input node voltage from equation (15) into equation

(13) with x = 0 one obtains:

$$[E] - [Z_G][I(0)] = [S_V](W^+(0) + W^-(0))$$

or

$$[E] = [S_v](W^+(0) + W^-(0)) + [Z_G][I(0)] \quad (17)$$

Substituting equation (14) for the input node current at x=0 into equation (17) one obtains:

$$[E] = [S_V](W^+(0) + W^-(0)) - [Z_G][Y_C][S_V](W^+(0) - W^-(0)) \quad (18)$$

substituting equation (16) into equations (13) and (14) with x=l one obtains:

$$W^+(0)[e^{-\gamma l}](1-[Z_L][Y_C]) + W^-(0)[e^{+\gamma l}](1+[Z_L][Y_C]) = 0 \quad (19)$$

Equations (18) and (19) are a pair of linear equations in which $W^+(0)$ and $W^-(0)$ can be solved. Once these coefficients are determined, the input and output node voltages are solved using equations (13) and (14).

These voltages are obtained in the frequency domain at each frequency determined by a FFT of the time domain signal [15]. The original input signal that is contained in the [E] vector is transformed to the frequency domain using a FFT algorithm. In the frequency domain the values of the

source impedance, load impedance, characteristic admittance, and γ are constant for each frequency from the frequency transformed input vector [E]. The solutions determined from equations (13) and (14) in the frequency domain represent the frequency response of the transformed input signal to the transmission line system. A reverse FFT is done on the solutions once they are determined for all of the frequencies called for by the FFT to obtain the time domain response.

The methodology used to calculate the solution to the transmission line equation can be considered to be "brute force" . No linear transformations of the matrix equations are done to improve mathematical efficiency. The reason that this is acceptable in the model presented is that the maximum size of the matrix equations to be solved is (3 x 3). Other authors have used approaches which use linear transformations of the matrix equations to convert them to tri-diagonal Topolitz form [7] or to diagonal matrix form [10] to allow the eigenvalues and eigenvectors to be more easily obtained. Since the size of the matrices in the model presented in this thesis is only 3 x 3, no advantage is gained by doing such linear transformations.

The fact that the model uses superposition of solutions to limit the size of the matrices allows for simple programming algorithms to obtain the solutions directly. No approximations need to be made about the values of off-diagonal elements to calculate the solutions to reduce CPU time. This accuracy combined with the fact that the numerical solution algorithms are event driven allow the user to get exact solutions for transmission line systems with computational efficiency.

CHAPTER III

PROGRAM STRUCTURE

The transmission line model developed in chapter II is written in the C programming language. This was done to allow for compatibility with mixed mode circuit simulators such as Network C or other functional simulators that allow "foreign" models to be used. Network C is a mixed mode circuit simulator that allows for both continuous time calculation and discrete event simulation. Network C is a super-set of the C programming language. It is designed to allow easy incorporation of user defined models for circuit simulation. The hierarchical nature of circuit descriptions in Network C are such that models of various levels of abstraction can be used in the same simulation. Calls to models that perform continuous time calculations can be triggered by digital events. This is particularly useful for large scale digital circuit modeling in which the circuit complexity would prohibit a complete simultaneous SPICE simulation and yet the analog response of the system is desired. Functional blocks of the digital circuit can be modeled with simple functional models, while areas of interest such as digital functional block input/output and interconnect can be modeled for the continuous analog response. The transmission line model presented in this thesis is particularly suited to be used with the Network C circuit simulator due to it's ability to calculate the numerical solution when triggered by input node events.

The model consists of several functional blocks illustrated in figure (4). Block 1 contains the model to network interface. This block controls event driven model calling, and the time domain voltage and current values for the input and output nodes of the transmission line. The equivalent input and output circuits for the transmission line from Figure (3), representing for example digitally controlled line drivers, reside in the first functional block. Block 2 contains the queue for superimposing solutions of each of the lines, and determining which lines will be passed to block 3 for numerical analysis.

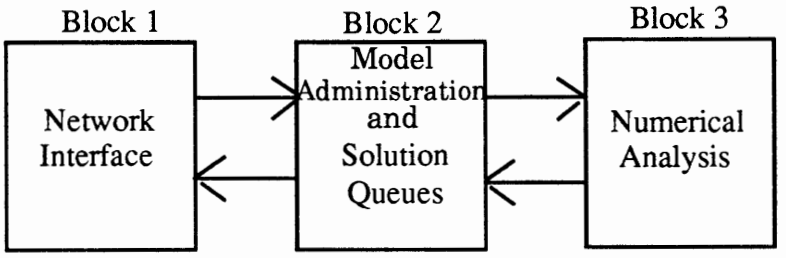

Figure 4. General Program Structure.

The third and final functional block determines the numerical solution for the transmission line system and returns the time-domain values to the 2nd "administrative" functional block.

Figure (5) is an illustration of the data path that the voltage signals follow in the transmission line model. The time-domain voltage signal data originates from n-input arrays that represent the time-domain input node voltage values. These signals originate as solutions to the equivalent circuit at the input node of the transmission line and stored in data arrays. The in

put node solutions are called when triggered by an event on the appropriate node. These signals may also contain voltage values that were the results of previous transitions on the line or lines adjacent to them. The signals for the "event" line and the two lines adjacent to it are passed to a subroutine that generates a mirror-symmetric signal in the second half of the three arrays. Due to the periodic nature of the FFT, signals passed to the FFT and back

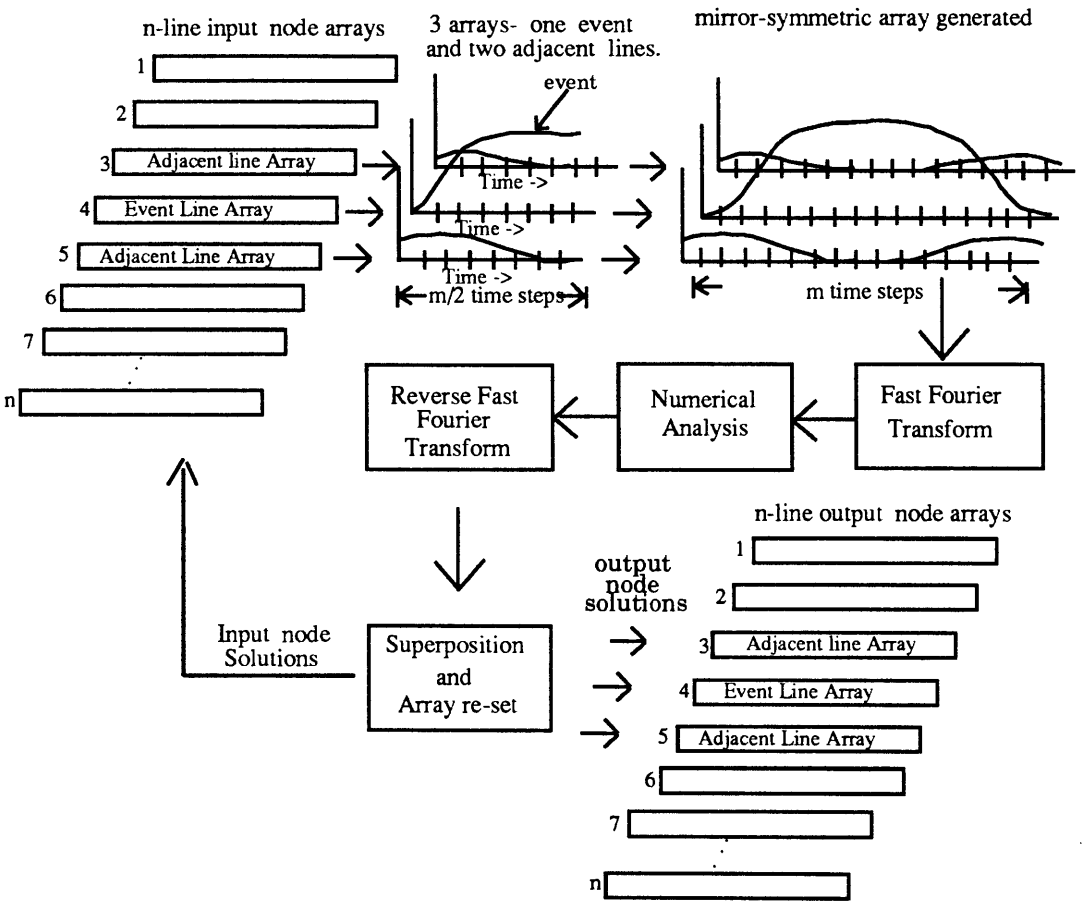

Figure 5. Data Path for Transmission Line Model.

may become distorted by the FFT algorithm's attempts to make the arbitrary signal periodic. By adding the mirror-symmetric signal to the array

this distortion is avoided. After the numerical solution is completed and the signals are transformed back to the time domain, only the first half of the solution array containing the physical signal is returned to the n-line input and output arrays. The second half of these arrays that contained the mirror symmetric values of the signal are discarded.

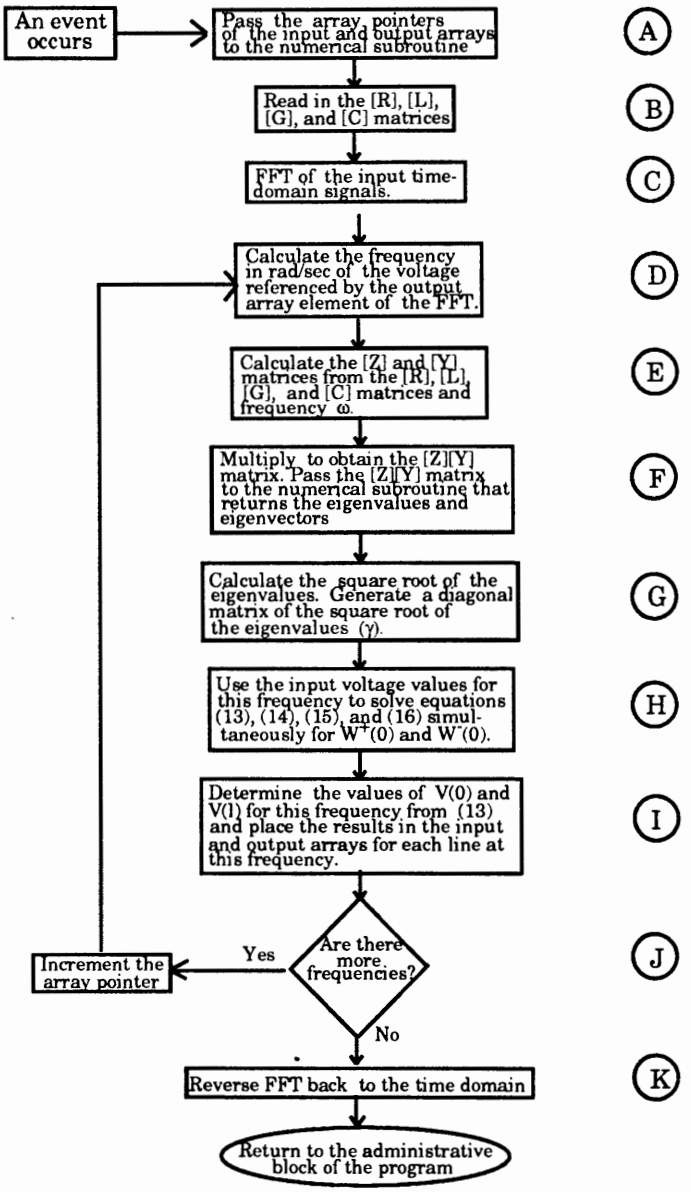

Figure 6. Numerical Calculation Flow Chart.

The numerical analysis is done on the input signals. Both the input node and output node solutions are placed back into the appropriate arrays in the n-line system. A detailed flow chart of the method in which the program calculates the numerical solution is illustrated in figure (6).

Figure (6) is a flow diagram of the algorithm used to solve for the input and output node voltage values for the transmission line in the frequency domain. Once the proper line signals are determined for the event line and two adjacent lines, the signals are passed to the numerical subroutine (A). The program then reads in the [R], [L], [G], and [C] matrices from external files (B). A FFT subroutine [15] is calculated for each of the three input line signals converting them from a voltage spectra as a function of time to a voltage spectra that is a function of frequency (C). The frequency of the the first voltage value in the transformed array is determined by the time step increment (dt) and the value of the array pointer. This is done in functional block (D). The user can set the value of dt to obtain the desired resolution of the signal. Once the frequency is determined, the [Z] and [Y] matrices can be calculated in step (E). The [Z] and [Y] matrices are multiplied and the product matrix is sent to the numerical subroutine that solves for the eigenvalues and eigenvectors of the [Z][Y] matrix [16]. The gama matrix from equation (11) is calculated from the eigenvalues in functional block (G). The voltage vector [E] is then used along with equations (13), (14), (15), and (16) to generate the forward and reverse wave coefficients $W^{+}(0)$ and $W^{-}(0)$ in functional block (H). The input and output node voltage values are solved for in functional block (I). The program repeats functional blocks (D) through (J) until all of the frequencies generated by

the FFT subroutine are processed. After this is done the complex conjugate of each of the solution voltage values is taken in functional block (K). A second FFT is done and the signals are converted back to the time domain. This is referred to as a reverse FFT. The input and output node solution arrays are then passed back to the administrative portion of the program for superposition back in to the appropriate locations.

Any time a transition occurs, the input node on the adjacent lines to the transitioning line are repositioned in the input signal storage arrays. The reposition procedure involves moving the remaining defined signal of the input wave forms to the front of the array to be passed for numerical analysis. The remaining elements of the first half of the array (those between the end of the defined waveform and the halfway point of the array) are filled with the last value of the defined input wave form. The last 1/2 of the array is filled with the mirror symmetric values of the input waveform after the reset and prior to the Fast Fourier Transform. The re-set process and mirror-symmetric signal generation are illustrated in figure (7).

After the numerical solution to a transition is completed and returned to the respective input and output arrays, all lines not involved in the transition are re-set as well. This is done so that the array pointer for any line in the n-line system is consistently at the same time step for both the lines that experienced a transition and the lines that did not. If no transition occurs, the model returns the values for the input and output node voltages that are stored in their respective arrays and are referenced by the array pointer which is stepped by the simulation environment's clock.

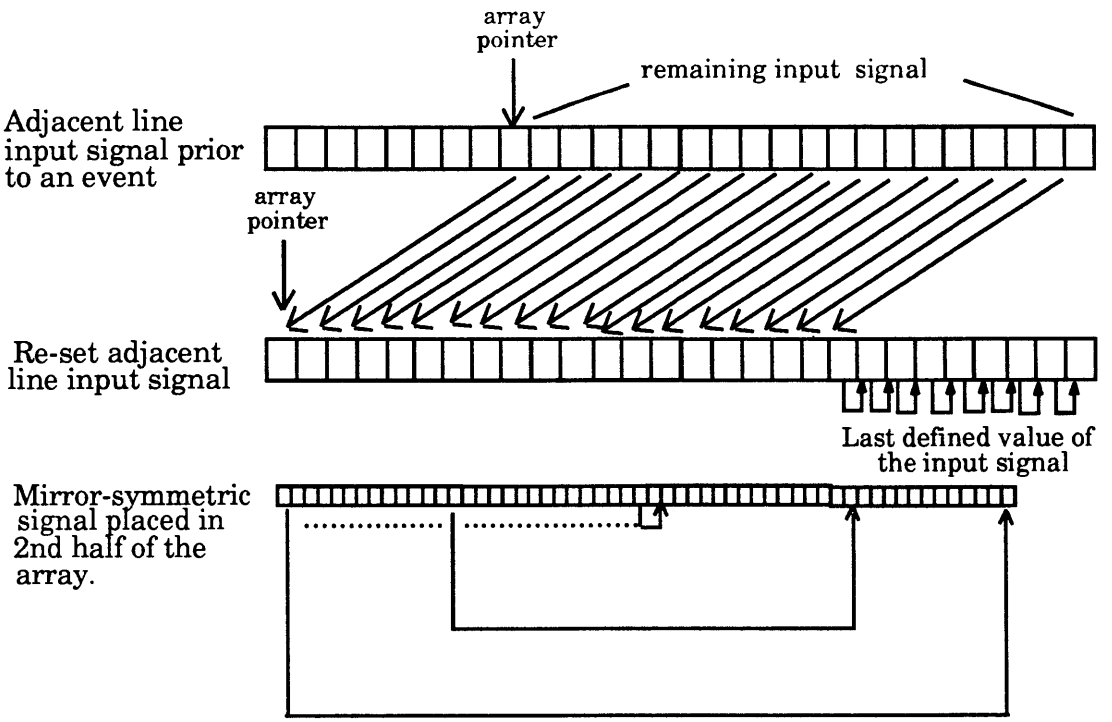

Figure 7. Adjacent line input array reset and mirror-symmetric signal generation.

The programming methods used to implement this model were conventional. No attempt was made to implement solution caching techniques, however this is something that the model can easily be upgraded to accommodate. In addition, if dynamic memory allocation for solution storage is used, the memory required to run the program will be reduced.

The programming structure presented lends itself to implementation in an event-driven simulation environment. This is due to the fact that the model's program structure allows for solutions to be organized and returned to the simulator's environment from data arrays that exist independent of the numerical portion of the program. The events control how the solution arrays are filled. If no events occur, then the solution arrays will be filled

with the last steady-state values of the transmission line input or output node voltages. It is this feature that allows this model to efficiently simulate the analog response of larger digital systems.

CHAPTER IV

NUMERICAL SIMULATION

NEAREST NEIGHBOR ANALYSIS

The order of the matrix equations to be solved, and therefore the CPU time required, is determined by the number of lines in the circuit being modeled for each individual transition. As will be shown later, computational time is reduced by limiting the number of lines and therefore the order of the matrix to 3. This is accomplished for larger order systems by passing only the signal generating line and it's two nearest neighbors on either side to the numerical portion of the model during an event.

This "nearest neighbor" approximation is justified by examining the response of a 5 line system representing a 0.8 micron space interconnect technology. The technology used for this discussion consists of a triple metal process with Plasma Enhanced Chemical Vapor Deposition (PECVD) SiO_2 dielectric. The metal lines modeled here represent the first layer of metal. Figure (8) shows the response of the output of a 5 line system with 1 cm interconnect length. Line 2 represents the line in which a signal is presented at the input, and lines 0,1,3,4 represent the "nearest neighbor" lines with the inputs held at ground. The output signals are plotted vs time for a typical low to high transition with a slew rate of 2×10^9 V/sec. The crosstalk

signal is apparent on the immediately adjacent lines (1 and 3), and to a much lesser extent on the next lines out (0 and 4). The maximum signal crosstalk present on the second line away from the signal line is 0.9% of the original signal. By ignoring signals on lines further than 2 away from the

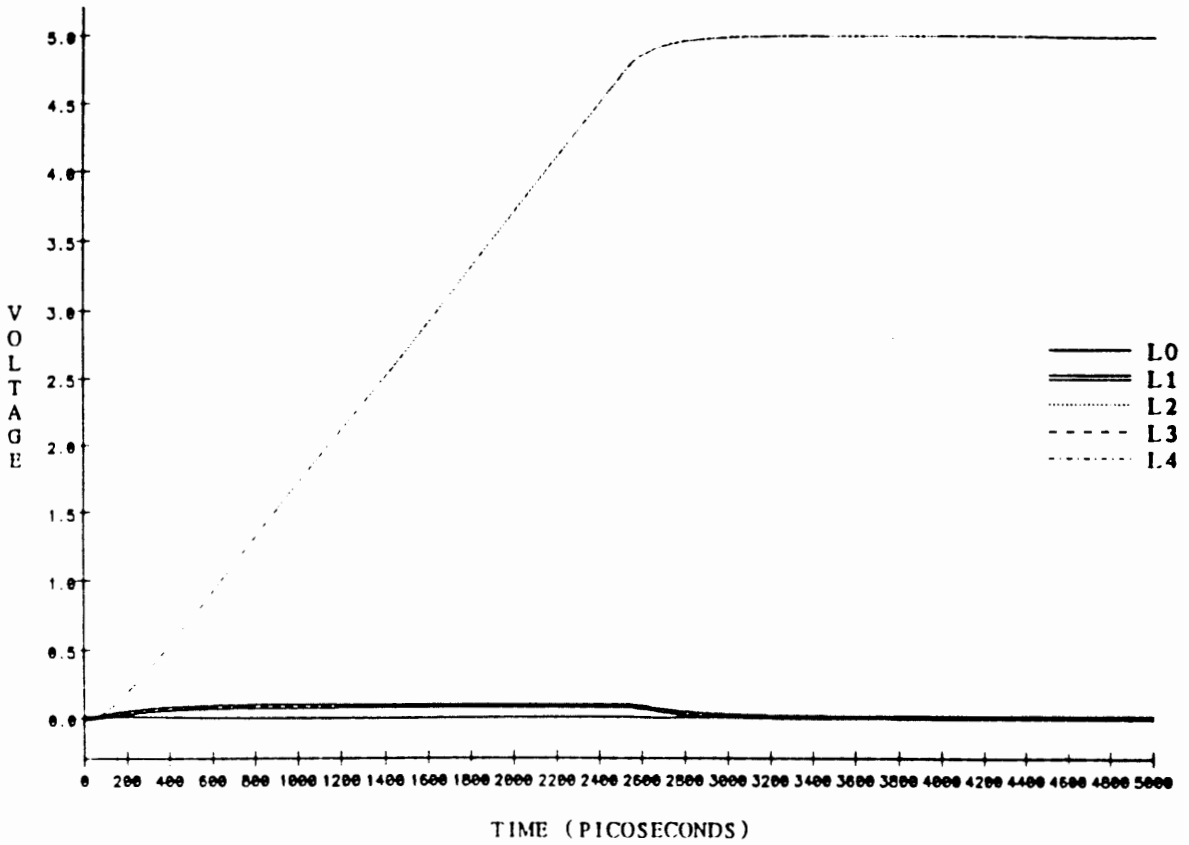

Figure 8. A 5 line system with a single transition on line 2.

event "driving" line, the order of the matrices and therefore CPU time can be reduced. In this manner larger systems such as 16 and 32 bit data buses can be analyzed without significantly increasing computational time.

The model allows the user to determine the size of the crosstalk fanout. The user must balance the desired signal size of interest vs the computational efficiency presented by the nearest-neighbor approximation. Figure (9) is a graph of CPU time vs number of lines simultaneously solved for 1 to 5 line systems.

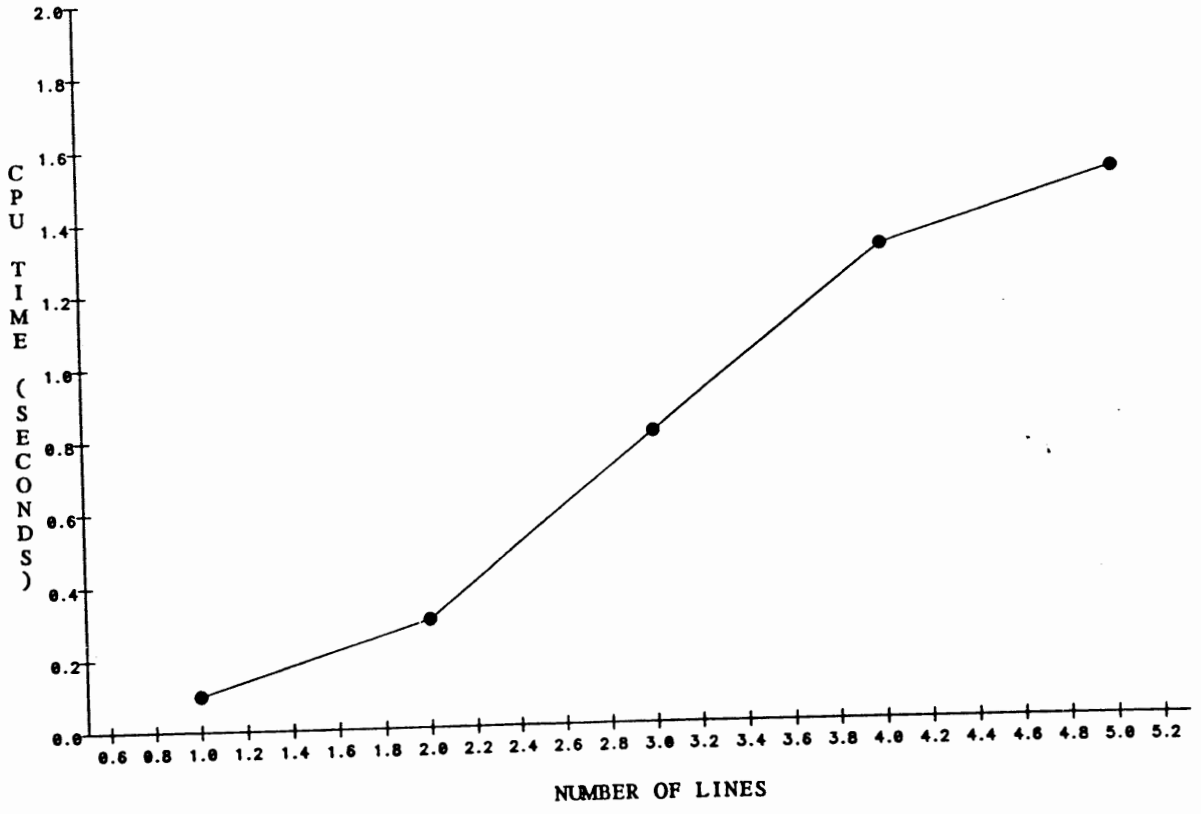

Figure 9. CPU time vs. number of lines for a single transition.

These simulations were run on a Digital Equipment Corporation System 5000 computer. CPU time goes up by roughly a factor of 2X between a

3-line solution and a 5-line solution. As demonstrated above, a 3-line nearest neighbor approximation adequately represents today's technology, however the model can accommodate a larger simultaneous solutions for higher resolution at the cost of higher CPU time per transition.

COMPARISON TO PREVIOUS WORK

To confirm that the numerical section of the transmission line model accurately calculates the transmission line behavior of coupled microstrip lines, a comparison of the model output for a single transition was made with results from A.R. Djordjevic et al. [11] The two line system is illustrated in figure (10):

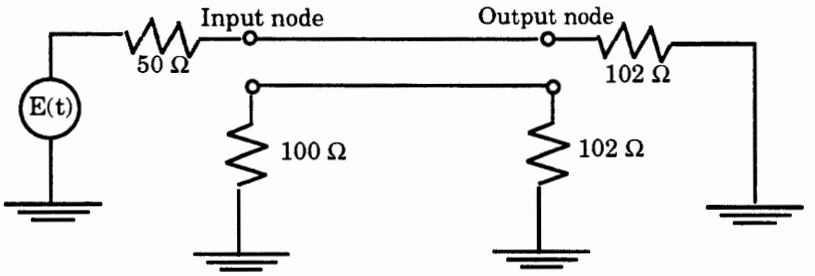

Figure 10. Two line system from Djordjevic et al.

The line length for the system is 304.8 mm long, the self inductance of each of the lines is 494.6 nH/m the mutual inductance is 63.3 nH/m, the self capacitance is 62.8 pF/m and the mutual capacitance is 4.94 pF/m. The voltage generator impedance is 50 ohms, and the line is considered lossless,

therefore the [R] matrix discussed in chapter II is 0. The load resistance of the driven line is 102 ohms, and the adjacent line has a input node impedance of 100 ohms and a output node impedance of 102 ohms.

Figure (11) and the dashed line in figure (12) shows the input node and output node signals on the driven line as a function of time from Djordjevic et al. Figure (13) shows the input node and output node signals generated from the model in this work.

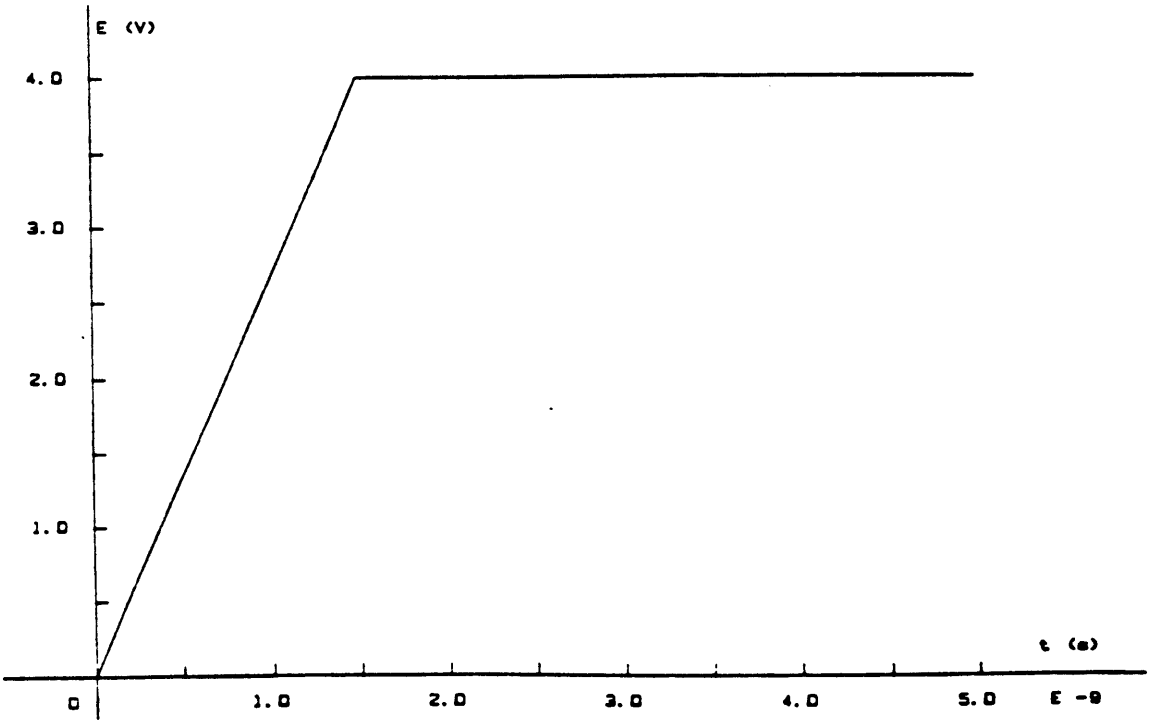

Figure 11. The input signal from Djordjevic [11].

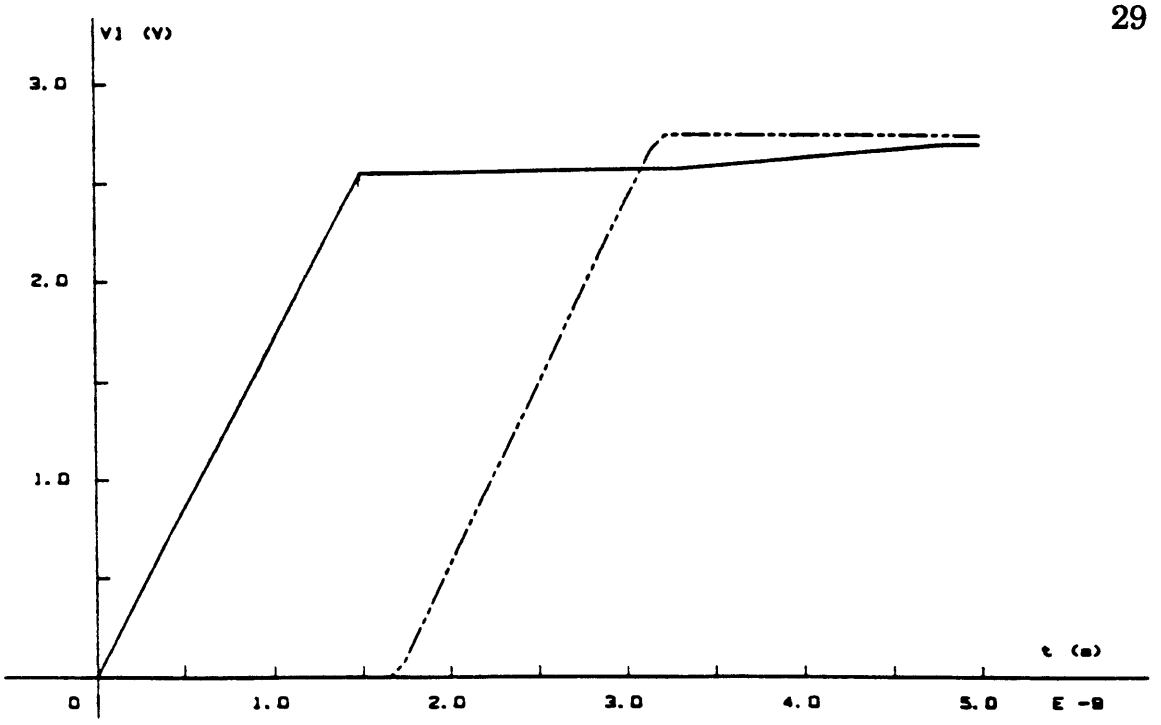

Figure 12. The output node signal from Djordjevic [11]

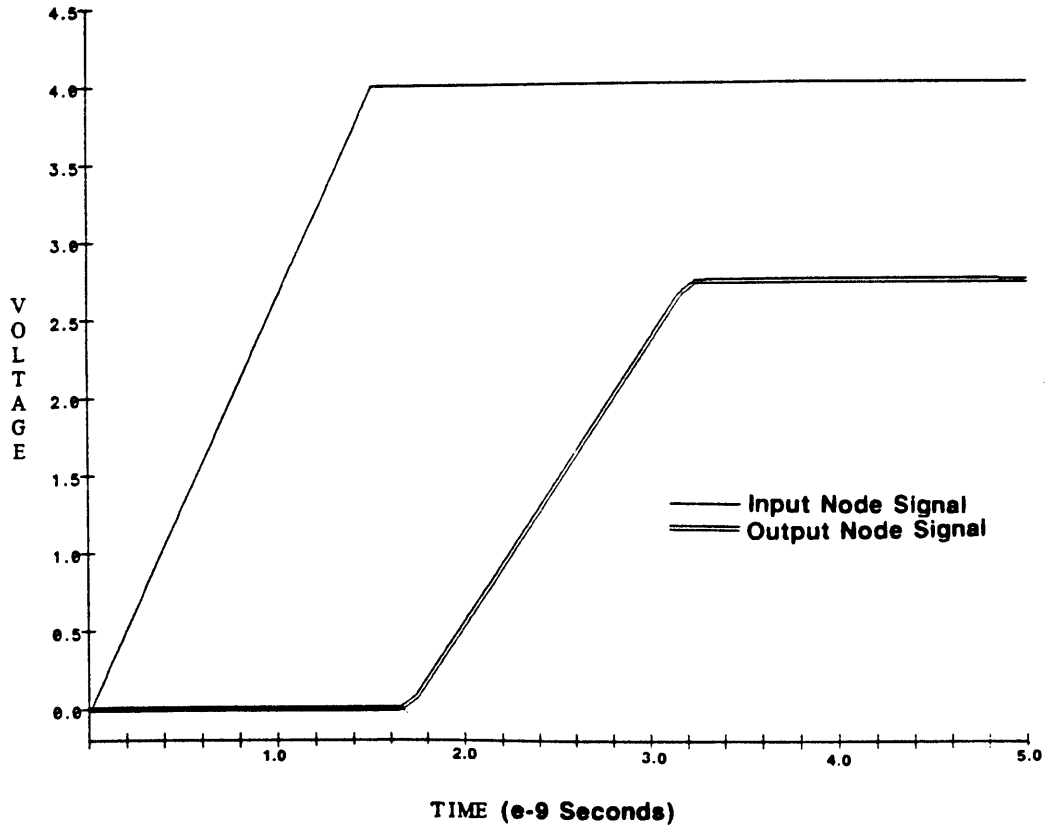

Figure 13. The input signal and output node signal from this work.

A comparison can be made between figures (11), (12), and (13). The input node signal shown in figures (11) and (12) rise at a slew rate of approximately 2.4 X 10^9 volts / second. The output node signal represented by the dotted line in figure (12) and the double line in figure (13) begins to rise after a delay of approximately 1.6 nanoseconds. It reaches it's high value of 2.5 volts at 3 nanoseconds. The output node signal generated from both works has the identical "rounding" effect at the beginning and ending of the transitions. From this comparison it is shown that the model presented in this work matches the results of others.

MODELING A 8-LINE SYSTEM WITH MULTIPLE TRANSITIONS

In order to obtain representative values for the [R], [L], [G], and [C] matrices for simulation, measurements were made of total resistance , self and mutual capacitance and inductance values on a metal interconnect test structure using a typical 0.8 um CMOS technology. The test pattern consisted of interdigitated comb-serpentine metal patterns illustrated in figure (14). Capacitance values were measured using a Hewlett-Packard 4275A LCR meter on a shielded probe station. Probe and pad capacitive and inductive values were measured independently and subtracted from the total measured values. The resistance, self capacitance, and self inductance values were measured with the substrate grounded and then divided by the length of the serpentine metal line to obtain the values per unit length. The mutual capacitive and inductive values were measured between the interdigitated comb structure and serpentine structure with the substrate floating. These values were also divide by the length of the structure

and then divided by 2 to obtain the mutual capacitance and inductance values represented by the off-diagonal elements in the matrix expressions. The values used for the [R], [L], [G] and [C] matrices as well as the input series impedance and load series impedance matrices are listed in the Appendix.

For the purpose of simulation a linear ramp was used to represent the input signal to the transmission line. Any signal can be used by the program however. The slew rate of the input signal was set at 2×10^9 V/sec which is typical for high speed digital circuits. Several combinations of 3-line transitions were simulated and illustrated in Figure (15). Table I lists the transitions.

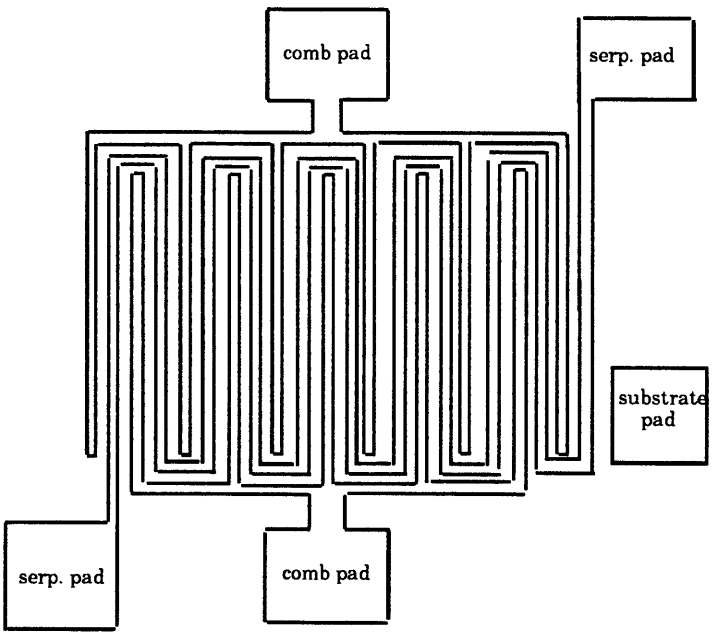

Figure 14. Interdigitated comb-serpentine metal pattern.

Figures (16) and (17) show the same transitions divided and overlayed on a 5 volt scale. Transition C represents a single line transition from low to high with the two adjacent lines to the primary transitioning line at a low value. This transition occurs on line 1 at the 100 picosecond point in figures (17) and (18). Lines 0 and 2 are immediately adjacent to line 1 and experience a sympathetic pulse of approximately 200 mV in magnitude for a duration of 2 nanoseconds. Transition D is similar to transition C with the exception that the two adjacent lines are already at high values. Transition D occurs at the 12.5 nanosecond point on line 2. Lines 1 and 3 are pulsed up to 5.2 volts from 5 volts for approximately 2 nanoseconds due to transition D. Worst-case transitions occur in transitions G and K at the 10.0 nanosecond point in figure 15 and 5.0 nanosecond point in figure (16) respectively. Here two lines are transitioning on either side of a line that is

TABLE I

TRANSITIONS SIMULATED IN FIGURES 15, 16, AND 17

Transition	line n-1	line n	line n+1
A	LOW	LOW->HIGH	HIGH
B	LOW	HIGH->LOW	HIGH
C	LOW	LOW->HIGH	LOW
D	HIGH	LOW->HIGH	HIGH
E	LOW	HIGH->LOW	LOW
F	HIGH	HIGH->LOW	HIGH
G	LOW->HIGH	LOW	LOW->HIGH
H	LOW->HIGH	HIGH	LOW->HIGH
I	HIGH->LOW	HIGH	LOW->HIGH
J	HIGH->LOW	LOW	HIGH->LOW
K	HIGH->LOW	HIGH	HIGH->LOW
L	LOW->HIGH	LOW	HIGH->LOW
M	LOW->HIGH	HIGH	HIGH->LOW

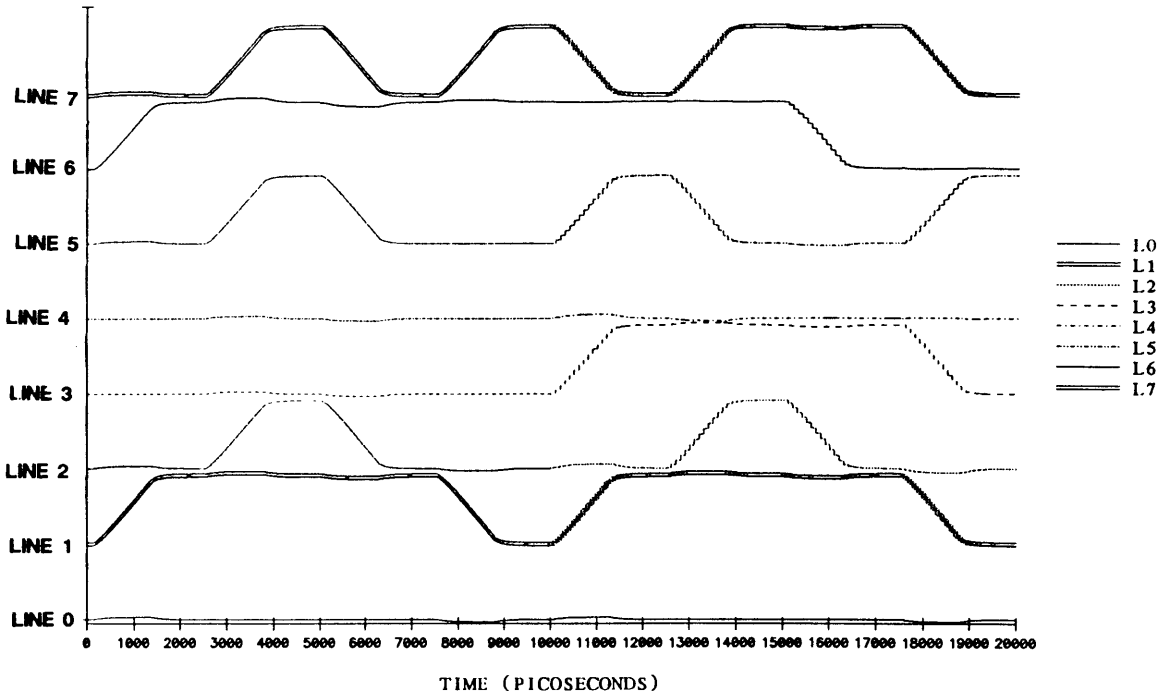

Figure 15. Modeled transitions on a 8-line system.

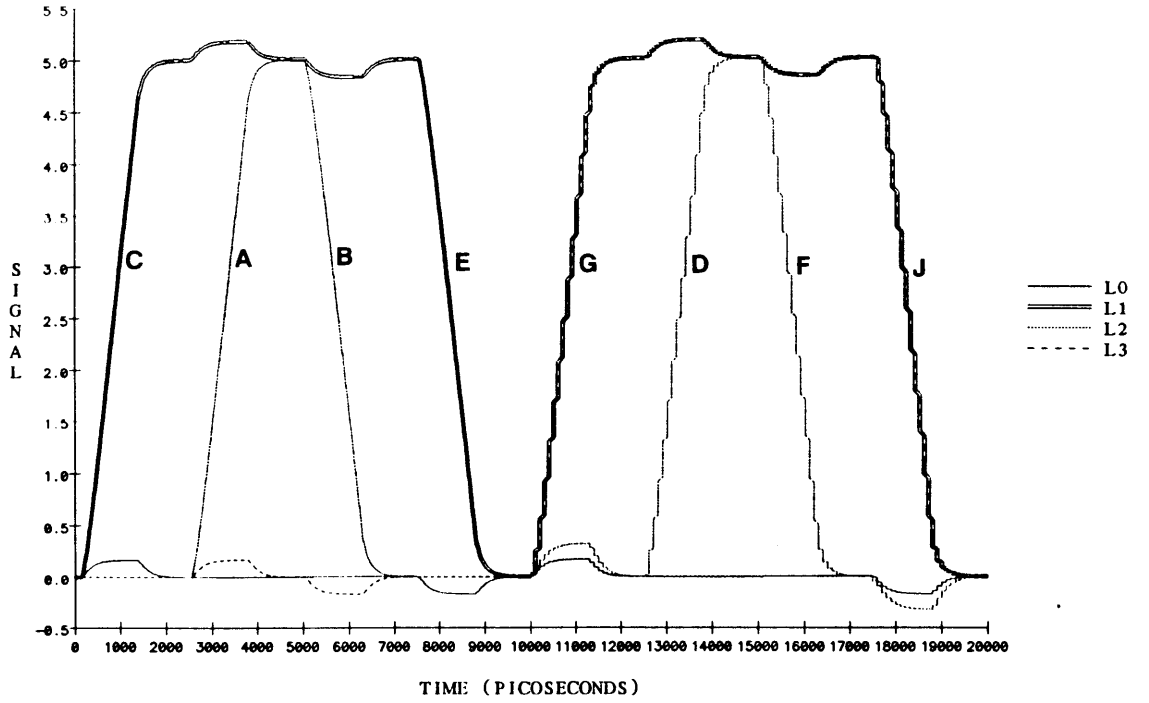

Figure 16. Lines 0 through 3 overlayed on a 5 volt scale.

held at a constant value. During transition G a low to high transition occurs on lines 1 and 3 with line 2 held at the low value. A sympathetic pulse occurs on line 2 that is double the magnitude of single transition pulses at 400 mV. The duration of the pulse is again about 2 nanoseconds. In the case of transition K, lines 5 and 7 transition from high to low with line 6 held at the high value. Line 6 momentarily drops to 4.6 volts for 2 nanoseconds due to the transitions of lines 5 and 7.

Transitions H and and J represent transitions that are worst-case for pumping adjacent lines outside of the 0 to 5 V range. Transition H occurs at 2.5 nanoseconds on lines 5 and 7 with both lines going from low to high simultaneously. Line 6 is initially high and is raised an additional 400 mV

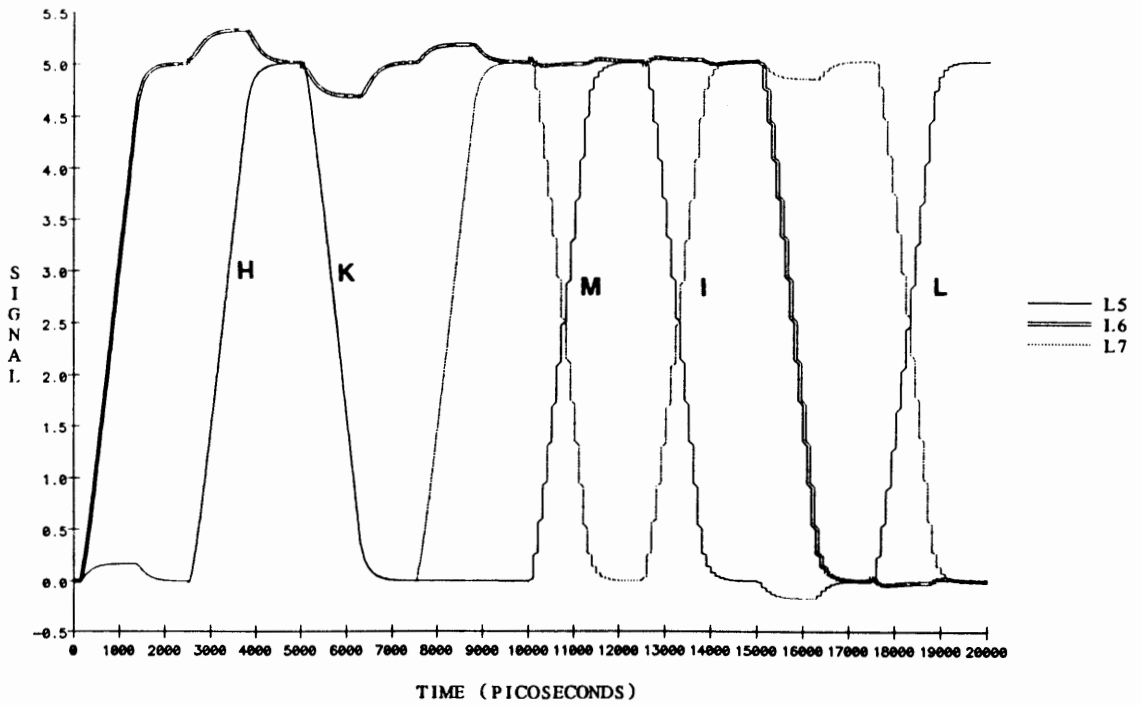

Figure 17. Lines 5 through 7 overlayed on a 5 volt scale.

above 5 volts due to the transitions on lines 5 and 7. Transition J occurs on lines 1 and 3 at 17.5 nanoseconds. Both lines 1 and 3 have transitions from high to low simultaneously with line 2 initially in a low state. The transitions on lines 1 and 3 cause line 2 to drop to -400 mV for 2 nanoseconds after the initiation of the pulse.

Mixed transitions are simulated with transitions L and M. Transition L occurs at 17.5 nanoseconds. on lines 5 and 7 , with line 5 transitioning between low to high and line 7 transitions between high to low. The net result is that the two adjacent transitions "cancel out" the sympathetic pulse that would have occurred on line 6 due to either the transitions on lines 5 of individually. Transition M occurs on lines 5 and 7 at 10.0 nanoseconds. In the case of transition M, line 5 transitions from low to high at the same time line line 7 is transitioning from high to low with line 6 held at the high level. The net result is once again a cancellation of the sympathetic pulse on line 6 due to the opposing nature of the pulses on lines 5 and 7.

The simulations run above used a time domain dt of 10 picoseconds per time step. Increasing dt has the effect of "roughening" the edges of the transitions. This is illustrated in figure (18). A single low to high transition is simulated with dt=1.0e-11, 1.0e-10, and 3.0e-10 seconds. It can be seen that the curve representing the dt of 3.0e-10 seconds has considerably more erratic at the beginning and end of the transition than the 1e-11 curve. The time step increment is a user defined value, and must be adjusted to match give the desired resolution.

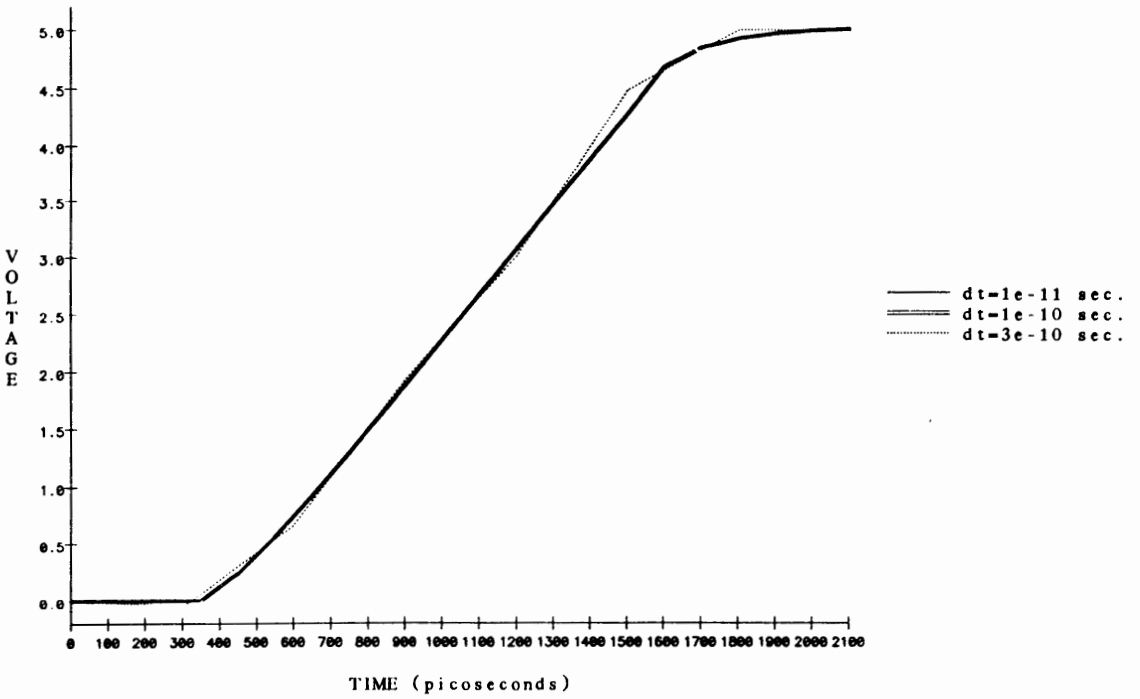

Figure 18. Low to high transitions with various time steps.

The example transitions reviewed in this chapter occurred simultaneously in many instances to illustrate worst-case transition conditions for sympathetic pulse generation. The numerical model does not require that the transitions occur simultaneously, and in fact are calculated in a serial fashion. The previous simulations were completed using 3-line solutions per transition. This could have just as easily been run with 5 or 7 lines per transition if the electromagnetic coupling to the second or third lines away from the transitioning line was significant. This of course would be at the cost of additional CPU time per transition.

In this chapter it was demonstrated that a 3-line "nearest neighbor" approximation adequately simulates line to line cross talk, and that the second and third line direct coupling is insignificant for a typical 0.8 micron CMOS technology. It was also demonstrated that the model presented in

this thesis is capable of replicating the simulation results of previous work.

The model has also been shown to be able to calculate the transmission line delay and mutual cross talk for multiple line systems. All possible 3-line combinations of transitions were simulated on a example 8-line network. The effects of the line to line cross talk were shown to be significant in worst-case transitions. Simultaneous adjacent line high to low and adjacent line low to high transitions were shown to cancel out the cross talk on the center line. The model has been shown to predict line overshoot and undershoot due to mutual line to line cross talk.

It should be noted that due to the current status of the code, the simulations conducted in this chapter were run using raw data dumps of voltage ramps to simulate input signals during an event. A "event list" was used to generate high to low or low to high transitions. When the model is fully incorporated into a event driven simulation environment such as Network C, the event list would be replaced by events generated by the primary event driven simulation program.

CHAPTER V

CONCLUSION

The model presented in this thesis has been demonstrated to accurately represent the behavior of multiple coupled microstrip lines. The model is unique in that it is designed to be easily incorporated into a mixed-mode digital and analog simulation environment. This model should be particularly useful to High-Speed VLSI Digital Circuit Designers. Circuit interconnect timing verification can now be easily taken into account for both the line delay and cross talk behavior of multiple coupled microstrip lines. Reliability Engineers may also use this model to analyze the duty cycle and magnitude of pulses generated by cross talk that push data lines above 5 volts. Over a period of time the above 5 volt signals may lead to premature device failure.

The programming methodology used to numerically solve the transmission line behavior in the frequency domain and then superimpose the resulting solutions in the time domain was discussed. It was shown that only three lines need to be simultaneously solved for a single transition. By limiting the numerically intensive portions of the program to only a 3-line system per transition, CPU time is reduced. The use of solution superposition allows this model to easily incorporate n-line systems at no great increase in CPU cost.

The model was confirmed to give accurate results when compared to previous work. By using measured values of microstrip line capacitance inductance and resistance, the model predicted the response of a .8 micron space 8-line microstrip system for various transition states. The resulting cross talk pushed adjacent line voltages well above and below their 5 volt or 0 volt nominal values. The magnitude of the cross-talk signals simulated demonstrates the need for Digital Circuit Designers to investigate the nature of the circuit interconnect behavior.

Future work on this model should focus on improving the CPU efficiency through improvement of the coding. Solution caching can also improve the CPU efficiency. In practice, solutions may be used more than once in a given simulation. If the "state" of the line is identical to a transition that was previously solved for, then the model would simply "look-up" the solution and superimpose it as appropriate. Currently the model calculates the solution of the three-line system at every transition.

A user-friendly interface for Network C applications should be developed so that this model can be easily used in mixed-signal applications. Currently this model requires that the interconnect parameters of capacitance, resistance, and inductance be known. Ideally the user of the model would need only to enter values for the interconnect resistance per unit length, line width, line spacing, and dielectric constants of the medium. The model would then generate the capacitive, inductive and resistive matrices.

REFERENCES

1. A.J. Groudis, C.S Chang "Coupled Lossy Transmission Line Characterization and Simulation" IBM Journal of Research and Development, Vol 25, No 1, January, 1981.

2. H. You, M. Soma, " Crosstalk Analysis of Interconnection Lines and Packages in High-Speed Integrated Circuits" IEEE Trans. on Circuits and Systems, Vol. 37, No 8, August 1990, pp 1019-1026

3. J. Chilo, T. Arnaud, " Coupling Effects in the Time Domain for an Interconnecting Bus in High-Speed GaAs Logic Circuits" IEEE Trans. on Electron Devices, Vol. ED-31, No 3. March 1984, pp 347-352

4. G. L. Matthaei, J. C.-H. Shu, S. I. Long, " Simplified Calculation of Wave-Coupling Between Lines in High-Speed Integrated Circuits" IEEE Trans. on Circuits and Systems, Vol. 37, No 10, October 1990, pp 1201-1208.

5. H. Hasegawa, S. Seki, "Analysis of Interconnect Delay on Very High-Speed LSI/VLSI Chips Using an MIS Microstrip Model" IEEE Trans. on Electron Devices, Vol. ED-31, No 12, December 1984, pp 1954-1960.

6. V. K. Tripathi, R. J. Bucolo, "Analysis and Modeling of Multilevel Parallel and Crossing Interconnection Lines" IEEE Trans. on Electron Devices, Vol. ED-34, No 3, March 1987, pp 650-658.

7. D. S. Gao, A. T. Yang, S. M. Kang, " Modeling and Simulation of Interconnection Delays and Crosstalks in High-Speed Integrated Circuits" IEEE Trans. on Circuits and Systems, Vol. 37, No 1, January 1990, pp 1-9.

8. K. Araki, Y. Naito, " Computer Aided Analysis of Coupled Lossy Transmision Lines", IEEE Proceedings of ISCAS 1985, pp 423-426.

9. M. Passlack, M. Uhle, H. Elschner " Analysis of Propagation Delays in High-Speed VLSI Circuits Using a Distributed Line Model" IEEE Tans. on Computer Aided Design, Vol. 9, No 8, August 1990, pp 821-826.

10. J. S. Roychowdhury, D. O. Pederson "Efficient Transient Simulation of Lossy Interconnect" Dept. of Electrical Engineering and Computer Sciences, University of California, Berkeley. (unpublished).

11. A. R. Djordjevic, T. K. Sarkar, R. F. Harrington " Time-Domain Resposne of Multiconductor Transmision Lines" Proceedings of the IEEE, Vol. 75, No 6, June 1987, pp 743-764.

12. W. Beckett "Modeling Electronic Circuits and Systems in Network C" Northwest Laboratory for Integrated Systems, Dept. of Computer Science, University of Washington, July 14th, 1988

13. M. N. O. Sadiku, L. C. Agba "A Simple Introduction to the Transmission-Line Modeling" IEEE Trans. on Circuits and Systems, Vol. 37, No. 8, August 1990, pp 991-999

14. C. H. Durney, C. C. Johnson: Introduction to Modern Electromagnetics McGraw-Hill Book Company, New York, 1969, pp 316-339.

15. E.O. Brigham: The Fast Fourier Transform, Prentice-Hall, Engelwood Cliffs N.J. [1974]. pp 164-165.

16. W.H. Press: Numerical Recipies in C: The Art of Scientific Computing, Cambridge University Press, New York, 1988

APPENDIX

Measured values for the [R], [L], [G], and [C] matrices used in Chapter 4:

$$[R] = \begin{bmatrix} 60.0e\text{-}3 & 0.0 & 0.0 \\ 0.0 & 60.0e\text{-}3 & 0.0 \\ 0.0 & 0.0 & 60.0e\text{-}3 \end{bmatrix}$$

$$[L] = \begin{bmatrix} 1.941e\text{-}9 & 0.329e\text{-}9 & 0.0 \\ 0.329e\text{-}9 & 1.941e\text{-}9 & 0.329e\text{-}9 \\ 0.0 & 0.329e\text{-}9 & 1.941e\text{-}9 \end{bmatrix}$$

$$[G] = \begin{bmatrix} 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 \end{bmatrix}$$

$$[C] = \begin{bmatrix} 1.400e\text{-}12 & -0.422e\text{-}12 & 0.0 \\ -0.422e\text{-}12 & 1.400e\text{-}12 & -0.422e\text{-}12 \\ 0.0 & -0.422e\text{-}12 & 1.400e\text{-}12 \end{bmatrix}$$