The surface radiation panels measured in an anechoic chamber. The results give rise to radiation at the block ends thus creating a ripple interference pattern. In Figs. 20 and 21, the effect on the parameters. Field reinforcement near dry wall surfaces is likely to occur in practical buildings. Ray-tracing and other field prediction techniques may be theoretically and experimentally demonstrated for some typical constructions.

Discussion and conclusions: Surface wave propagation on walls has been theoretically and experimentally demonstrated for some typical building materials. The water content of the material has pronounced effect on the parameters. Field reinforcement near dry wall surfaces is likely to occur in practical WLAN scenarios, both around and within buildings. Ray-tracing and other field prediction techniques may require modification to account for the surface wave fields and the complex interactions taking place.

Acknowledgments: Helpful discussions with R. Craven (DSTL) and R. Fairweather (Defence Science Technology Laboratory (DSTL), Malvern, Worcs., WR14 3PS, United Kingdom) are gratefully acknowledged.

References

Modelling noise and delay in VLSI circuits
D. Pamlenuwa, S. Ellassaad and H. Tonkun

New models for estimating delay and noise in VLSI circuits, based on closed form expressions for the first and second moment of the impulse response in coupled RC trees are reported. The effect of crosstalk on delay and noise can be accurately estimated with a complexity only marginally higher than the Elmore delay.

Timing analysis in VLSI circuits has long used the simple model of an RC tree where all capacitors are connected to ground, which we shall call a simple tree. Signal delay through the tree is usually estimated by approximating the dominant time constant with the Elmore delay [1], the first moment of the impulse response. For waveforms which are poorly represented by a single exponential, a two pole, single zero model based on the first two moments and the sum of the open circuit time constants was proposed in [2]. Generic moment matching techniques were later derived in [3], which could be applied for any type of circuit with linear elements. Recently [4] and [5] have presented alternate second-order models for the transfer function, using the first three moments, and the first two moments with heuristics, respectively.

With the integrated implementation of nanometre processing, noise coupling has become very important, and the circuit model has to include coupling capacitors. In estimating the delay in such coupled trees, the usual practice is to modify the Elmore delay by an empirical coefficient [6] which however results in poor accuracy. General moment matching can be applied, but for initial analysis that requires many iterations, the formulation of the nodal matrices and subsequent costly matrix manipulations are best avoided, even for models which depend only on the first three or four moments [7].

In this Letter, we report second-order models for general arbitrarily-coupled trees with multiple drivers that depend only on the first two moments of the circuit, which moreover are explicitly matched to the element values by means of closed form equations. Our models can be thought of as an extension of the methodology presented in [2] to coupled trees, and represent the minimum complexity with which second-order approximations can be obtained without compromising generality. The response for each driver switching is obtained by grounding all other inputs. The superposition of all waveforms allows accurate delay and noise estimations.

The circuit topology consists of simple trees coupled to each other through series capacitors, and an example is shown in Fig. 1. The notation we employ is that $C_{g}$ is the capacitance to ground at node $k$ in the $p$th tree, $C_{n}$ is the capacitance between node $k$ and node $j$ in the $q$th tree where the first sub(super)script refers to the reference tree, $R_{s}$ is the resistance shared on the paths between the source and nodes $e$ and $k$, respectively, on tree $p$, and $V_{D}$ is the tree voltage.

Superscripts always refer to simple trees while subscripts always refer to nodes, except in the definition for moments, where the superscript refers to the order of the moment. Additionally, rail voltages are normalised to 0 and 1, and the expression for a positive step without loss of generality.

Acknowledgments: Helpful discussions with R. Craven (DSTL) and R. Fairweather (Defence Science Technology Laboratory (DSTL), Malvern, Worcs., WR14 3PS, United Kingdom) are gratefully acknowledged.

© IEE 2003 28 November 2002
Electronics Letters Online No: 20030122
DOI: 10.1049/el:20030122
J.R. James and L.L. Morrow (Laboratory of Electromagnetic Research, Department of Aerospace, Power and Sensors, Cranfield University, RMCS, Shrivenham, Swindon, Wilts. SN6 8LA, United Kingdom)
P. Cushnahan and A.S. Fairweather (Defence Science Technology Laboratory (DSTL), Malvern, Worcs., WR14 3PS, United Kingdom)
A coupled RC tree is characterised by a resistive path from the receiver node to the forcing (victim) driver, and series capacitive elements to other (aggressor) drivers. Hence when the victim driver switches the output will always change rails, whereas it will start and end at the same rail for an aggressor switching. Therefore the second-order transfer function for the former case will have a zero on the negative part of the real axis:

\[ H_v(s) = \frac{1 + s \tau_{v1}}{1 + s \tau_{v2}} \]  

(1)

and at the origin for the latter:

\[ H_a(s) = \frac{s \tau_{a1}}{1 + s \tau_{a2}} \]  

(2)

Using Kirchoff's laws and integration by parts, the first moment of the impulse response at \( s \) for the victim driver switching is seen to be:

\[ Y_1^v = \sum_{k \in \text{victim}} R_k \left[ C_k^{\text{in}} + CC_k^{\text{in}} + \cdots \right] = t_{v2} \text{ say} \]  

(3)

Similarly, the second moment is:

\[ Y_2^v = 2 \sum_{k \in \text{victim}} R_k \left[ C_k^{\text{in}} t_{v2} + CC_k^{\text{in}} \left[ t_{v2} + \sum_{k \in \text{victim}} R_k CC_k^{\text{in}} \right] \right] + C_k^{\text{in}} t_{v2} + \sum_{k \in \text{victim}} R_k CC_k^{\text{in}} + \cdots \right] = 2(t_{v2})^2 \text{ say} \]  

(4)

The first moment at node \( e \) for aggressor \( a \) switching is:

\[ Y_1^a = - \sum_{k \in \text{aggressor}} R_k \left[ C_k^{\text{out}} + CC_k^{\text{out}} + \cdots \right] = -t_{a2} \text{ say} \]  

(5)

The second moment is:

\[ Y_2^a = -2 \sum_{k \in \text{aggressor}} R_k \left[ C_k^{\text{out}} + CC_k^{\text{out}} + \cdots \right] t_{a2}^2 \]  

(6)

The moments are matched to the characteristic time constants in the circuit by using the identity that the \( n \)th moment of the impulse response at \( s \) is \((-1)^n\) times \( n \)th derivative of the transfer function evaluated at \( s = 0 \). When applied to (1), (3) and (4), this results in (7) and (8):

\[ t_{v2} = t_1 + t_2 = t_v \]  

(7)

\[ (t_{v2})^2 = (t_1 + t_2)(t_1 + t_3) - t_1 t_2 \]  

(8)

Now \( t_1 \) and \( t_2 \) in (7) and (8) refer to the dominant poles for the event of the victim driver switching. To solve this system of three unknowns, a third equation is required. Since all aggressors are grounded, the metric that gives the best solution is the sum of the open circuit time constants with reference to the victim driver, which we shall call \( \tau_v \). This is a good approximation for the sum of the pole time constants, giving:

\[ \tau_v = \tau_1 + \tau_2 \]  

(9)

Now (7), (8) and (9) can be solved for the zero and two poles associated with the victim switching.

To solve for the poles and zeros associated with an aggressor switching, the above identity is used on (2), (5) and (6) to give:

\[ t_{a2}^2 = t_2 + 2t_3 \]  

(10)

\[ (t_{a2})^2 = t_1 + t_3 \]  

(11)

Now the zero is available immediately, and dividing (11) by (10) results in the pole sum:

\[ \frac{(t_{a2})^2}{t_{a2}} = \frac{t_1}{t_2} + 1 \]  

(12)

Again, the solution for the poles requires extra information, which can be obtained by using intuition gained from the physical interpretation of the moments. The first moment includes resistances of the switching line, and either all capacitances connected to it (when the victim switches) or capacitances connecting it to a particular line (when an aggressor switches). The second moment propagates outwards another level, and includes resistances and capacitances of immediately adjacent lines as well. Since the victim is the net of interest, combining the moments for the victim switching with those for a particular aggressor switching results in a solution which is biased towards the victim and that aggressor. Hence to obtain the solution to a switching aggressor, (7), (8) and (12) can be combined to generate the two poles of interest.

Fig. 1 Coupled RC tree (values repeated within simple trees)

Fig. 2 Response at output node \( e \) for switching of different drivers

a) Response of aggressor tree 4  
b) Response of aggressor tree 3  
c) Response of aggressor tree 6  
d) Response of aggressor tree 5  
e) Response of victim

The accuracy of the models for the transfer function were tested by comparing the step responses for a variety of circuits against a dynamic simulator Spectre, and found to be excellent for the vast majority. The waveforms shown in Fig. 2 correspond to the circuit of Fig. 1. For certain pathological cases where the significant poles for a switching aggressor are located very far away from those of the victim, this methodology can fail to generate real poles for the aggressor. Such an
occurrence is an indication that the receiver node is charged via a strong aggressor, i.e. through a relatively small time constant, and decays with a very long tail, dictated by the much larger time constant of the victim. One remedy would be to use the third moment of the impulse response to gather extra information, and use a technique that guarantees stability, similar to that described in [4] just for those corner cases. Another would be to consider simple circuit transformations, as any instability is an indication that the response is dictated by a very limited section of the overall circuit for that particular case. However for almost all the topologies tested by the authors, the methodology described gave stable and accurate results. The simplicity and accuracy of the models combined with their generality should make them useful in delay and noise estimations in complex systems, early in the design flow.

© IEE 2003

5 November 2002

Electronics Letters Online No: 20030208
DOI: 10.1049/el:20030208

D. Pumnuwa and H. Tenhunen (Department of Microelectronics and Information Technology, Royal Institute of Technology (KTH), Electrum C229, SE-16440 Stockholm-Kista, Sweden)
E-mail: dineesh@imt.kth.se

S. Ellassaad (Cadence Berkeley Laboratories, Cadence Design Systems, 2001 Addison St., Berkeley, CA 94704, USA)

References


Watkins-Johnson converter completes tapped inductor converter matrix

D.A. Grant and Y. Darroman

The Watkins-Johnson converter has been identified as belonging to the tapped inductor converter family extending once more the matrix of DC-DC converter topologies. This converter is analysed in terms of the top positions and the switch duty cycle and its operation as a rail-to-tap buck converter is verified.

Introduction: Tapped inductor DC-DC converters are well-known [1, 2]. They are often used in industrial and domestic applications because they facilitate high or low output-to-input voltage ratios with good efficiency if there is no requirement for isolation. Moreover tapping the inductor has the benefits that the duty cycle of the converter at the operating point can be adjusted to a desirable value—typically a value at which device utilisation is improved. The tapping of the inductor also permits a different mix of voltage and current ratings for the various elements of the circuits.

Filling the void in tapped inductor matrix: Recent work on these converters has shown that tapped and untapped versions of the Watkins-Johnson converter formed a matrix of possibilities to which a new column was added with two new converter members [3, 4]. This categorisation left a void in the matrix at the position assigned to the buck converter of the rail-to-tap configuration. We now believe that the Watkins-Johnson converter [5-10] is the best candidate to fill this void. In [10] it is referred to as a buck converter with desirable properties—in particular, on switch-off, the isolation from the output of any energy stored in the inductor.

Fig. 1 Tapped inductor family of converters: circuit diagrams

Table 1: Tapped inductor family of converters: voltage transfer ratios

<table>
<thead>
<tr>
<th>Converter type</th>
<th>No tap</th>
<th>Switch-to-tap</th>
<th>Diode-to-tap</th>
<th>Rail-to-tap</th>
</tr>
</thead>
<tbody>
<tr>
<td>Buck</td>
<td>D(D(K-K))</td>
<td>(K+1(K-1))+1</td>
<td>D(K-K)</td>
<td>D(K-K)</td>
</tr>
<tr>
<td>Boost</td>
<td>1(1-β)</td>
<td>(K+1(K-1))/K</td>
<td>1(1-β)</td>
<td>(K-β)</td>
</tr>
<tr>
<td>Buck-boost</td>
<td>-β(1-β)</td>
<td>D(K+1)</td>
<td>-β(1-β)</td>
<td>(K-β)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Converter type</th>
<th>No tap</th>
<th>Switch-to-tap</th>
<th>Diode-to-tap</th>
<th>Rail-to-tap</th>
</tr>
</thead>
<tbody>
<tr>
<td>Buck</td>
<td>D(D(K-K))</td>
<td>(K+1(K-1))+1</td>
<td>D(K-K)</td>
<td>D(K-K)</td>
</tr>
<tr>
<td>Boost</td>
<td>1(1-β)</td>
<td>(K+1(K-1))/K</td>
<td>1(1-β)</td>
<td>(K-β)</td>
</tr>
<tr>
<td>Buck-boost</td>
<td>-β(1-β)</td>
<td>D(K+1)</td>
<td>-β(1-β)</td>
<td>(K-β)</td>
</tr>
</tbody>
</table>

The transfer ratio \( V_{out}/V_{in} \) (in continuous conduction mode) of the converters shown in Table 1 and the basic power circuits of each converter are shown in Fig. 1. The Watkins-Johnson converter (or mil-to-tap buck converter) is highlighted in these tables. The transfer ratio for the Watkins-Johnson converter indicates that it can buck without inversion of polarity. In this mode it can supply a passive load (positive output voltage and output current). It can buck and boost with polarity inversion although in this regime an active load is required since the output current must remain positive even though the output voltage is negative. Hence, with the classification scheme employed