Z Domain Delay Subcircuits and Compact Verilog-A Macromodels for Mixed-mode Sampled Data Circuit Simulation

M. E. Brinson and H. Nabijou, *Member, IEEE*

***Abstract*—Mixed-mode simulation is an important circuit design and system testing tool for established and emerging semiconductor sampled data technologies. This paper describes a number of functional, computationally efficient, Z domain delay models, outlining the role of current and charge equations in the construction of subcircuit and compact Verilog-A delay macromodels. To illustrate the properties of the proposed macromodels a number of Qucs (Quite universal circuit simulator) transient and frequency domain simulation examples are presented. Each of these stresses the use of test and data extraction techniques which are not easily undertaken with the SPICE 2g6 or 3f5 simulators.**

***Index Terms*—Mixed-mode sampled data circuit simulation; Functional delay subcircuits; Compact Verilog-A delay macromodels; Qucs (Quite universal circuit simulator)**

I. Introduction

In modern circuit design the term mixed-mode simulation has become synonymous with the analysis and design of integrated analog and digital electronic systems. Early simulators, and indeed some more recent releases of commercial and GNU Public License (GPL) open source packages, were primarily analog circuit analysis tools [1] with polynomial sources (SPICE 2g6 [2]) and non-linear controlled sources (SPICE 3f5 [3]) modeling non-linear components at a functional level. Today, most circuit simulators include a digital simulator that operates synchronously with an analog analysis engine [4], allowing the analysis and testing of complex mixed-mode circuit designs. In terms of sampled data technologies mixed-mode simulation is much more than simply the combination of analog and digital analysis software. Designers are faced with the need to simulate multi-domain systems which include an ever increasing range of electrical and non-electrical technologies. Moreover, amongst the current general purpose simulators it is not common for packages to include dedicated simulation engines for sampled data system analysis and testing [5]. Hence, accurate and computationally efficient signal delay, signal summing and signal multiplication models are required for use with general purpose circuit simulators. Ideally, such sampled data component models should also be optimized for minimum memory usage. This paper introduces a number of fundamental functional subcircuit and Verilog-A [6] delay models. The proposed models have been implemented and tested as equation defined device subcircuits [7] and compact Verilog-A macromodels [8] using the Qucs (Quite universal circuit simulator) GPL software [9]. The text stresses those model features which are not found in SPICE 2g6 or 3f5 and outlines how sampled data system models can be developed which are suitable for use with any general purpose circuit simulator that allows subcircuits with parameters or compiled Verilog-A hardware description language models.

M. E. Brinson is with the Faculty of Computing, London Metropolitan University, London, UK. (phone: 0044-607-2789; e-mail: mbrin72043@ yahoo.co.uk). He is a member of the Qucs development team.

H. Nabijou, is with the Faculty of Computing, London Metropolitan University, London, UK.(e-mail: h.nabijou@londonmet.ac.uk).

Manuscript received October XX, 2009

II. Fundamental properties of sampled delay components

A key component in the simulation of sampled data systems is a delay element with a delay of one sampled data period. In the small signal AC frequency domain an ideal delay is represented by (1).

$$Z^{-1} = e^{-j\omega T} = cos(\omega T) - j \cdot sin(\omega T) \tag{1}$$

The magnitude and phase angle of Z^{-1} are given by (2).

$$mag(Z^{-1}) = 1, angle(Z^{-1}) = -\omega T \tag{2}$$

Similarly, group delay (*GD*) and the first differential of the group delay (dGD/d ω) are given by (3).

$$GD = \frac{-d(angle(Z^{-1}))}{d\omega} = T, \frac{dGD}{d\omega} = 0 \tag{3}$$

Where ω is angular frequency in radians per second and T is the sampling period in seconds. These properties should be true for all frequencies over which the delay element is expected to operate. SPICE and other circuit simulators often employ ideal transmission lines as delay elements. Unfortunately, this approach is far from perfect because ideal transmission lines tend to consume large amounts of computer memory and often simulate rather slowly during transient analysis [10]. The speed factor is particularly true for circuit simulators developed from the SPICE 2g6 code when simulating circuits which include ideal transmission lines with short delays; this occurs because the 2g6 version of SPICE restricts the maximum transient analysis time step to half the smallest transmission line delay [11]. In a realistic transient run time both the available memory and simulation speed constraints limit the size of sampled data mixed-mode circuits which can be simulated to around five to ten ideal transmission line delays per circuit [12].

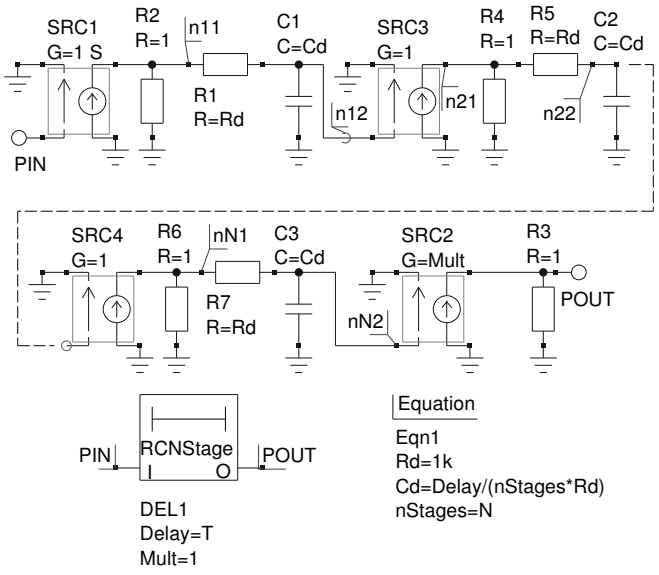

Fig. 1. Delay network subcircuit symbol and circuit consisting of N series cascaded RC stages separated by controlled source buffering stages.

III. Cascaded RC delay networks as delay elements

The circuit schematic illustrated in Fig. 1 is a Qucs subcircuit simulation model of a delay network consisting of N series cascaded RC stages separated by buffering voltage controlled current sources with resistive loads. Each RC stage provides a delay of *T/nStages* seconds, where T is the total delay of the network, and *nStages* is the number of cascaded stages. Qucs equation block Eqn1 determines, prior to the start of simulation, the values of components R_d and C_d from subcircuit parameter *Delay* and the value of *nStages*. Subcircuit parameter *Mult* allows the delay output signal amplitude to be scaled, the default value being 1. The RC delay network characteristics are given by (4), (5), (6), and (7) respectively.

$$mag(Z^{-1}) = \frac{Mult}{[1+A^2]^{\frac{nStages}{2}}} \tag{4}$$

$$angle(z^{-1}) = -nStages \cdot arctan\,(A) \tag{5}$$

$$GD = \frac{nStages}{\omega_p} \cdot \frac{1}{[1+A^2]} \tag{6}$$

$$\frac{dGD}{d\omega} = \frac{-2 \cdot \omega \cdot nStages}{\omega_p^3 \cdot [1+A^2]^2} \tag{7}$$

Where $\omega_p = 1/(R_d \cdot C_d) = nStages/T$, and $A = \omega/\omega_p$. Fig. 2 introduces a simple AC test circuit for obtaining the fundamental properties of an RC delay network under test. The Qucs equation blocks listed in Fig. 2 either set up the test conditions (Eqn2 plus the AC1 simulation parameters) or are post simulation data extraction scripts (Eqn6 and Eqn7) for processing output data and theoretical data calculated from (4), (5), (6), and (7). Qucs pre and post simulation data processing features allow equations to be entered in any order, and in one or more Eqn blocks, based on the popular MATLAB

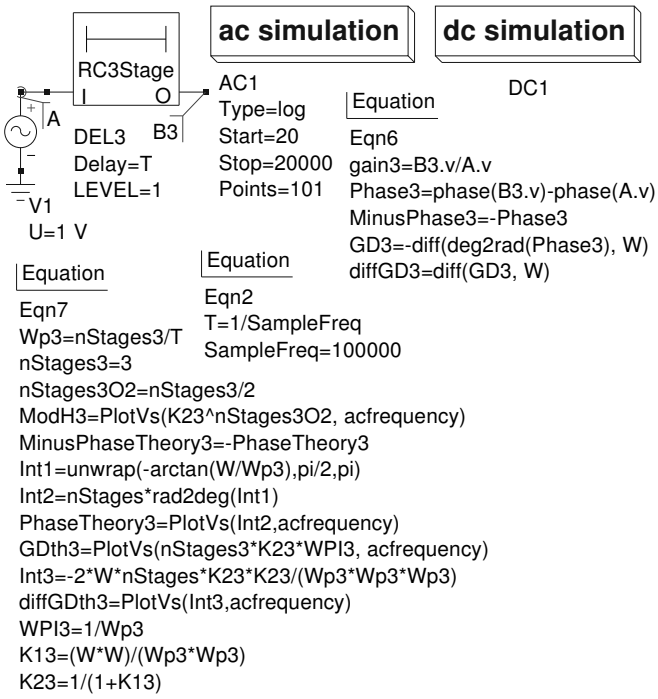

Fig. 2. Three stage RC delay subcircuit AC small signal test circuit and data extraction equation scripts: sampling frequency set at 100K Hz, analysis frequency range 20 Hz to 20 kHz.

®[13] / Octave [14] language syntax and the mathematical operators and functions defined in the Verilog-A hardware description language. A set of simulated data plots for the characteristics of one to three RC section delay networks are illustrated in Fig. 3. Table I lists an example Verilog-A analog module code routine for a three section RC delay network. Code line numbers are written in `'{ }'` brackets at the left hand side of each line. In Table I Qucs subcircuit parameters are given in lines `{6}` and `{7}`. Verilog-A code lines `{10}` to `{13}` determine the values of variables specified in line `{8}`. Model current contributions are listed in lines `{15}` to `{23}`. These entries correspond directly to the circuit structure and component types shown in Fig.1, making translation from Qucs subcircuit schematic to Verilog-A code a straightforward process.

IV. Cascaded First Order RC Padé all Pass Delay Networks

Although the RC delay networks outlined in section III provide the required low node and component count their delay performance at high frequencies is poor. In the test example shown in Fig. 2 the characteristics of the delay element under test are simulated over the audio band of frequencies. At frequencies above approximately 1 kHz the group delay becomes downgraded, resulting in observable error. Increasing the number of RC sections, or the sampling rate, would of course improve the delay performance but these approaches either increase the number of nodes and components or the simulation time which in turn reduces simulation performance, particularly during transient simulation. A better solution is to choose a network with improved delay characteristics per section. Consider the

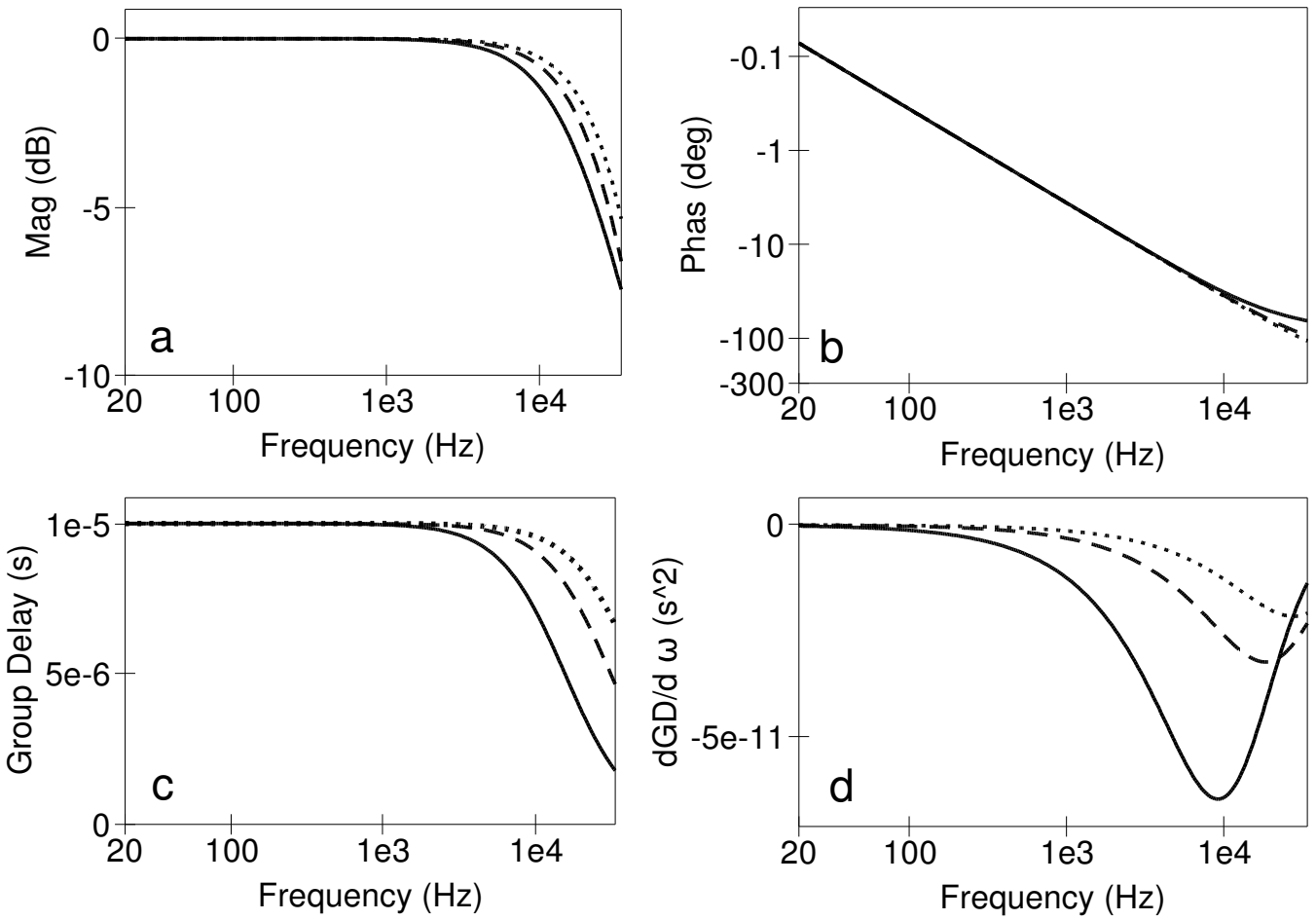

Fig. 3. Simulated RC delay subcircuit AC characteristics: graph a; $mag(Z^{-1})$ against frequency, graph b; $angle(Z^{-1})$ against frequency, graph c; GD against frequency, and graph d; $dGD/d\omega$ against frequency: where the solid lines are for a one stage RC delay network, the dashed lines are for a two stage RC delay network , and the dotted lines are for a three stage RC delay network.

the N section cascaded all pass first order Padé delay network shown in Fig. 4. The fundamental electrical characteristics of this delay network are given by (8), (9), (10), and (11) respectively, where $\omega_p = 1/(C_d \cdot R_d) = 2 \cdot nStages/T$, and $A = \omega/\omega_p$. Fig. 5 shows the simulated characteristics for one to three stage Padé all pass delay networks. These graphs clearly indicate the superior performance of the cascaded first order Padé network across the audio frequency band. For the three stage network this is achieved however, with 11 nodes and 20 components compared to 8 nodes and 14 components for the equivalent RC delay model. Table II lists an example Verilog-A analog module code routine for a three section Padé all pass delay network. The structure of the Verilog-A code routine presented in Table II follows that given in Table I.

$$mag(Z^{-1}) = \frac{Mult \cdot \left[(1 + 2 \cdot A^2 + 4 \cdot A^2\right]^{\frac{nStages}{2}}}{[1 + A^2]^{nStages}} \tag{8}$$

$$angle(z^{-1}) = -nStages \cdot arctan\left(\frac{2 \cdot \omega}{\omega_p \cdot (1 - A^2)}\right) \tag{9}$$

$$GD = \frac{2 \cdot nStages}{\omega_p} \cdot \frac{\left[1 + A^2\right]}{[1 + 2 \cdot A^2 + A^4]} \tag{10}$$

$$\frac{dGD}{d\omega} = \frac{-4 \cdot \omega \cdot nStages}{\omega_p^3 \cdot [1 + 2 \cdot A^2 + A^4]} \tag{11}$$

V. Transient simulation of a Z domain integrator

Integrators are a basic building block in the synthesis of active filters and many other important classes of mixed-mode circuit. In this section the performance of the proposed delay macromodels are presented using the Z domain integrator as a test device. In the S domain the transfer function of an integrator is given by (12).

$$H(S) = \frac{1}{S} \tag{12}$$

A discrete approximation of equation (12) can be derived by applying Tustin's method [15] using (13) to give (14).

$$S = \frac{2}{T} \cdot \left[\frac{1 - Z^{-1}}{1 + Z^{-1}}\right] \tag{13}$$

$$H(Z) = \frac{Y(Z)}{X(Z)} = \frac{T}{2} \cdot \left[\frac{1 - Z^{-1}}{1 + Z^{-1}}\right] \tag{14}$$

TABLE I
VERILOG-A CODE FOR A THREE STAGE RC DELAY NETWORK

```
{1}  ‘include disciplines.vams
{2}  ‘include constants.vams
{3}  module PadeDelay3Stage (PIN, POUT);
{4}  inout PIN, POUT;
{5}  ‘define attr(txt) (*txt*)
{6}  parameter real Delay=1e-6 from [1e-20 : inf]
       ‘attr(info=Delay unit=s);
{7}  parameter real Mult=1 from [1e-20 : inf]
       ‘attr(info = Signal gain);
{8}  real Gd, Cd; // Variables
{9}  analog begin
{10} @(initial_model) // Variable initialisation code
{11}     begin
{12}       Gd=1e-3; Cd=Delay*Gd/3;
{13}     end
{14} // Current contributions
{15} I(n11) < + -V(PIN); I(n11) < + V(n11);
{16} I(n11,n12) < + V(n11,n12)*Gd;
{17} I(n12) < + ddt(Cd*V(n12)); I(n21) < + -V(n12);
{18} I(n21) < + V(n12);
{19} I(n21,n22) < + V(n21,n22)*Gd; I(n22) < + ddt(Cd*V(n22));
{20} I(n31) < + -V(n31);
{21} I(n31) < + V(n31); I(n31,n32) < + V(n31,n32)*Gd;
{22} I(n32) < + ddt(Cd*V(n32));
{23} I(POUT) < + -Mult*V(n32); I(POUT) < + V(POUT);
{24} end
{25} endmodule
```

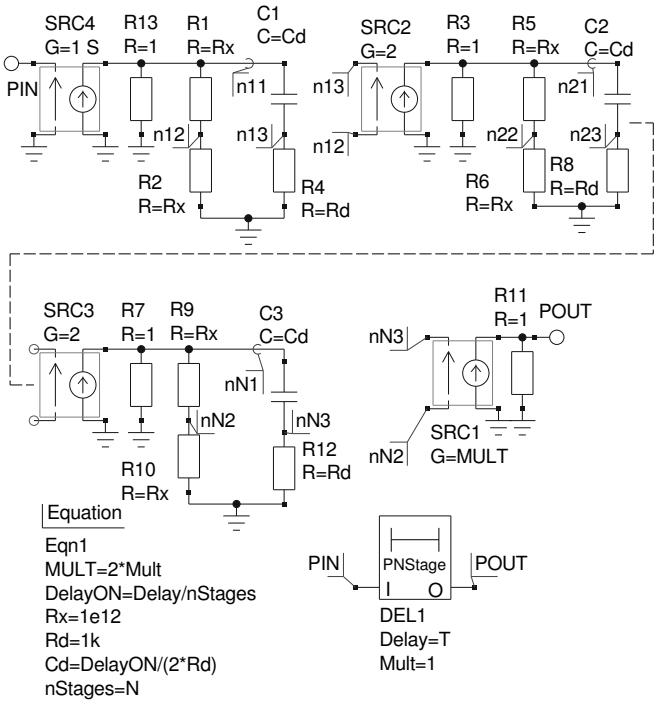

Fig. 4. Delay network subcircuit symbol and circuit consisting of N series cascaded first order Padé stages separated by controlled source buffering stages.

Expanding (14) results in (15).

$$Y(Z) = \frac{T}{2} \cdot X(Z) + \left[\frac{T}{2} \cdot X(Z) + Y(Z)\right] \cdot Z^{-1} \qquad (15)$$

Where X(Z) and Y(Z) are the Z domain input and output signals respectively. Illustrated in Fig. 6 is a large signal test circuit suitable for investigating the transient characteristics of the Z domain integrator: input signal A consists of the sum of

TABLE II
VERILOG-A CODE FOR A THREE STAGE PADÉ ALL PASS DELAY NETWORK

```
{1}  ‘include disciplines.vams
{2}  ‘include constants.vams
{3}  module RCDelay3Stage (PIN, POUT);
{4}  inout PIN, POUT; electrical PIN, POUT;
{5}  electrical n11, n12, n13, n21;
{6}  electrical n22, n23, n31, n32, n33;
{7}  ‘define attr(txt) (*txt*)
{8}  parameter real Delay=1e-6 from [1e-20 : inf]
       ‘attr(info=Delay unit=s);
{9}  parameter real Mult=1 from [1e-20 : inf]
       ‘attr(info = Signal gain);
{10} real Gd, Cd; // Variables
{11} analog begin
{12} @(initial_model) // Variable initialisation code
{13}     begin
{14}       Gx= 1e-9; Gd=1e-3; // Rx=1e9 and Rd=1k;
           Cd=Delay*Gd/6;
{15}     end
{16} // Current contributions
{17} I(n11) < + -V(PIN); I(n11) < + V(n11);
{18} I(n11,n12) < + V(n11,n12)*Gx;
{19} I(n12) < + V(n12)*Gx; I(n11,n13) < + ddt(Cd*V(n11,n13));
{20} I(n13) < + V(n13)*Gd; I(n21) < + -2*V(n12,n13);
{21} I(n21) < + V(n21); I(n21,n22) < + V(n21,n22)*Gx;
{22} I(n22) < + V(n22)*Gx; I(n21,n23) < + ddt(Cd*V(n21,n23));
{23} I(n23) < + V(n23)*Gd;
{24} I(n31) < + -2*V(n22,n23); I(n31) < + V(n31);
(25) I(n31,n32) < + V(n31,n32)*Gx;
{26} I(n32) < + V(n32)*Gx; I(n31,n32) < + ddt(Cd*V(n31,n32));
(27) I(n33) < + V(n33)*Gd;
{28} I(POUT) < + -Mult*V(n32,n33); I(POUT) < + V(POUT);
{29} end
{30} endmodule
```

28 single tone sinusoidal signals each of one volt amplitude and differing fundamental, second and third harmonic frequencies. Signal B3S is the integrator output. The performance of the Z domain integrator can be charted by simulating the circuit over a period of one second. Summers Sum1 and Sum2 shown in Fig. 6 have properties represented by (16).

$$V(out) = GIN1 \cdot V(in1) + GIN1 \cdot V(in2) \qquad (16)$$

Where GIN1 and GIN2 are numerical constants or variables held in Qucs equation blocks. Fig. 7 presents plotted simulation data for the Z domain integrator. In order to show the detail in the input and output waveforms only the results for the first 0.5 seconds of the one second simulation period are displayed in Fig.7. Input and output data are analyzed using the fast Fourier transform technique controlled by the post simulation processing and data extraction script listed in Qucs equation block Eqn2, Fig. 6. Fig. 8 illustrates the large signal frequency domain gain $(V(B3S)/V(A))$ and phase $(phase(V(B3S)) - phase(V(A))$ function data extracted from the input and output signal amplitude spectra plotted in Fig. 7.

A. Relative timing and accuracy of the RC and Padé delay networks

Table III shows a set of relative simulation run times for the transient analysis of a series of single stage Z domain subcircuit integrators with identical circuits but differing delay models. All timings are relative to those recorded for a circuit

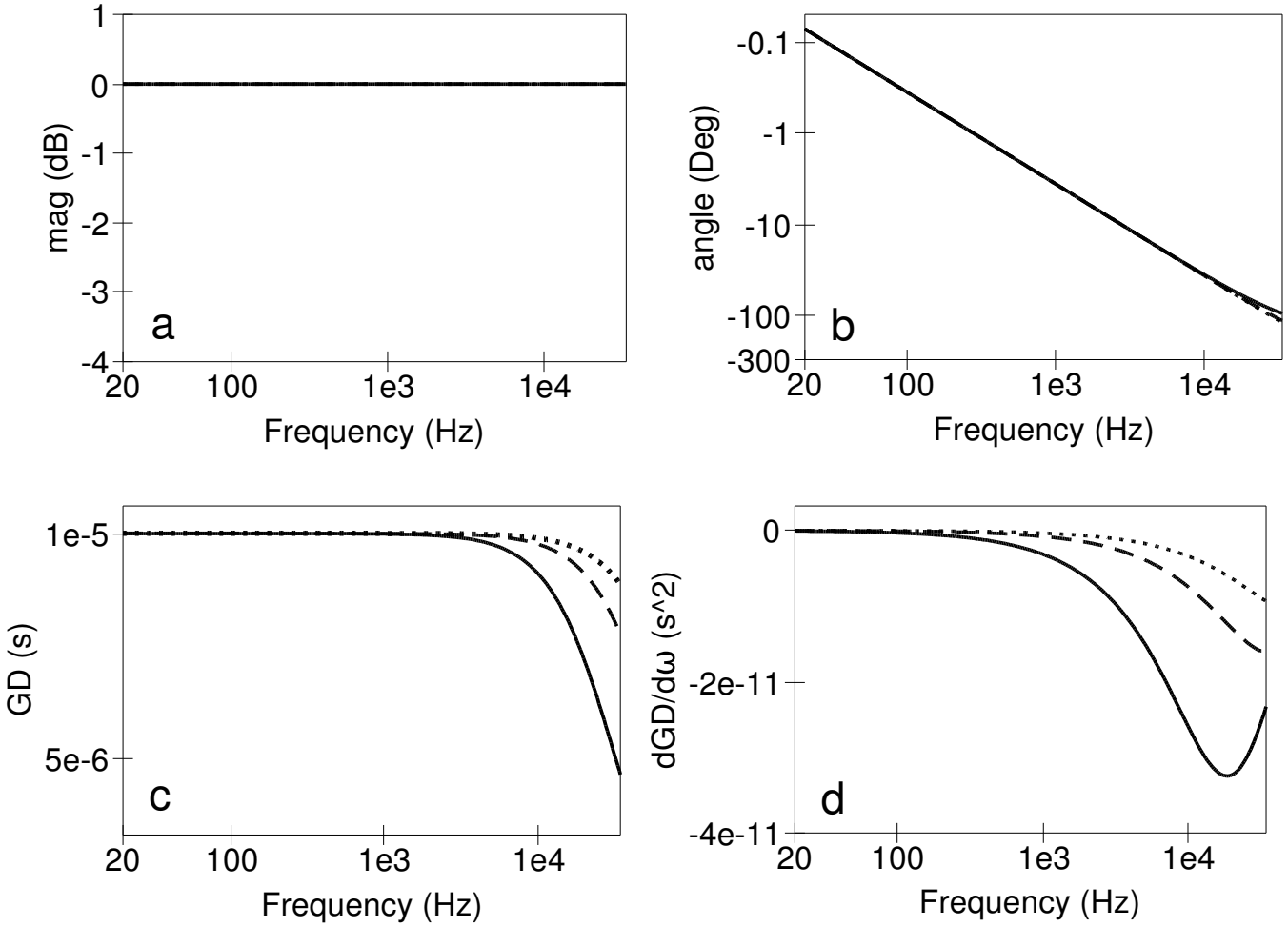

Fig. 5. Simulated all pass Padé delay subcircuit AC characteristics: graph a; $mag(Z^{-1})$ against frequency, graph b; $angle(Z^{-1})$ against frequency, graph c; GD against frequency, and graph d; $dGD/d\omega$ against frequency: where the solid lines are for a one stage Padé delay network, the dashed lines are for a two stage Padé delay network , and the dotted lines are for a three stage Padé network.

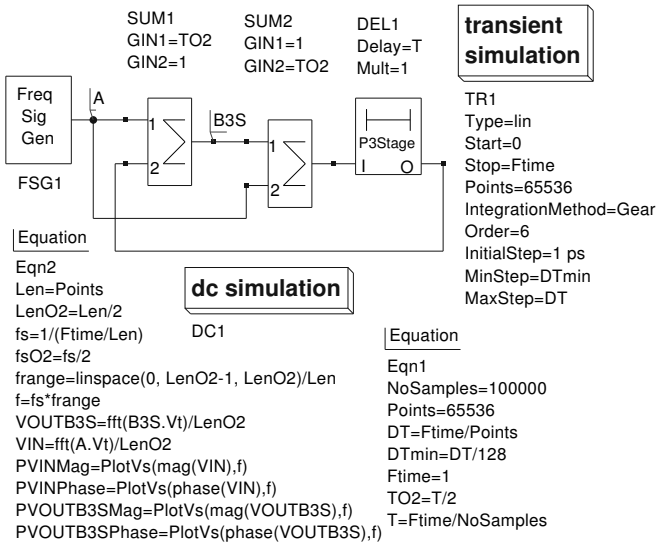

Fig. 6. Z domain integrator transient simulation test circuit: sampling rate is set at 100 kHz; Gear integration (order 6) with the maximum step size set as the inverse of the sampling rate.

with a single stage RC delay model simulated using the trapezoidal numerical integration rule. The simulation conditions were; sinusoidal input signal = 100 Hz at 1 V peak, finish time = 100s, number of samples = 100000 (sampling period = 10 μs), initial transient analysis step = 1ps, abstol = 1pA, vntol = 1uV, and reltol = 0.001. As expected these results confirm that the transient simulation time increases with the number of stages in a delay model. Selection of explicit or implicit numerical integration routine [16] also affects timing. The implicit Gear algorithm performing the best in the integrator test case. Hence, by carefully choosing the number of stages in an RC or Padé delay network a compromise between accuracy and simulation speed is possible. Fig. 9 illustrates the large signal AC transfer characteristics for a Z domain subcircuit integrator constructed around a single stage RC delay network employing an identical sampling rate to that used in the previous tests. Clearly at most frequencies, except for those at the top end of the frequency band, the results are similar to those given in Fig. 8, implying that the single RC delay model can be used without significant loss in accuracy at low frequencies while minimizing simulation run time. Timing tests for Z domain Verilog-A integrators constructed from

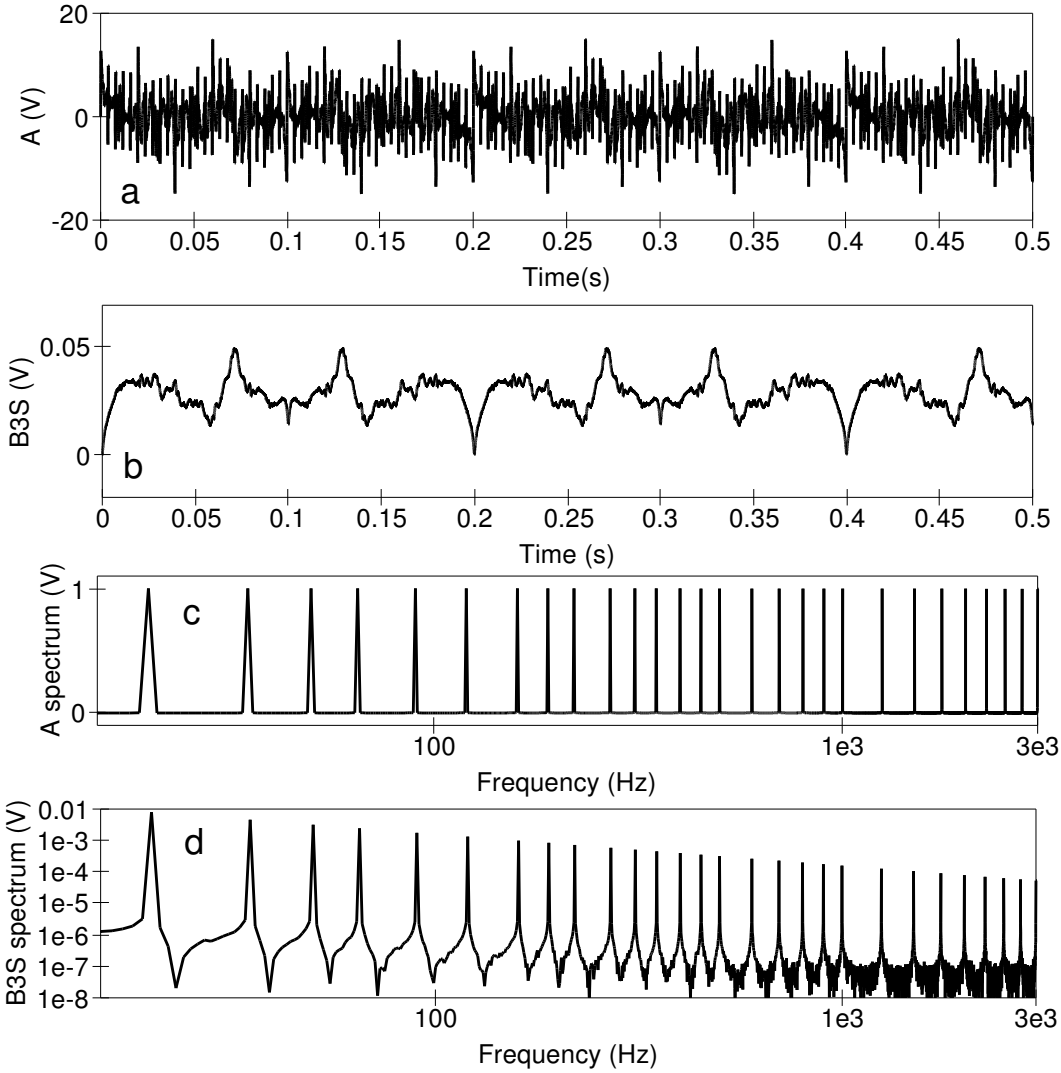

Fig. 7. Z domain integrator transient simulation characteristics: graph a; input signal A against time, graph b; output signal B3S against time, graph c; amplitude spectra for signal A against frequency, and graph d; amplitude spectra for signal B3S against frequency.

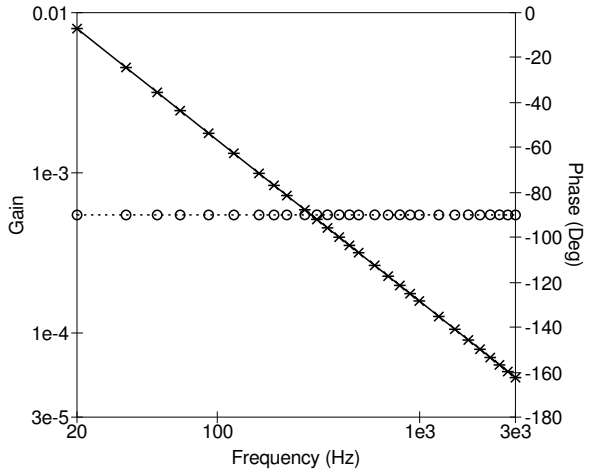

Fig. 8. Padé three stage Z domain integrator frequency domain characteristics: gain against frequency; solid line theory, crosses simulation, phase against frequency; dotted line theory, circles simulation.

TABLE III
RELATIVE TRANSIENT SIMULATION TIMINGS FOR SINGLE STAGE Z DOMAIN SUBCIRCUIT INTEGRATORS

Delay model type	Number of stages	Number of nodes	Number of components	Explicit Trapezoidal Integration	Implicit Gear Integration (Order 6)
RC	1	4	6	1	0.39
RC	2	6	10	1.31	0.67
RC	3	8	14	2.22	0.80
Padé	1	5	8	1.45	0.59
Padé	2	8	14	2.08	0,67
Padé	3	11	20	2.55	1.18

the code listed in Table I and Table II, using the ADMS (Analogue Device Model Synthesizer) [17] compiler, suggest that the relative performance of the Verilog-A models are similar to their equivalent subcircuits. As a general rule it has been found that highly non-linear Verilog-A device models tend to be computationally more efficient than their equivalent equation defined device subcircuits, resulting in measurable improvements in transient run times.

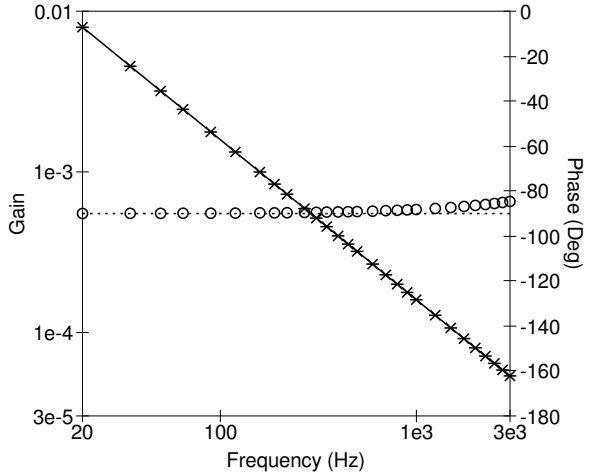

Fig. 9. RC one stage Z domain integrator: frequency domain characteristics; gain against frequency; solid line theory, crosses simulation, phase against frequency; dotted line theory, circles simulation.

VI. Conclusions

Although ideal transmission delay lines are often used to model Z^{-1} functions their characteristics in terms of memory usage and simulation speed are far from ideal for mixed-mode simulation of sampled data systems. The delay models introduced in this paper demonstrate that it is possible to construct alternative Z^{-1} functions from standard electrical components. By extending conventional subcircuit technology to include parameters and non-linear equation defined components it becomes possible to construct delay models based on current and charge equations which translate easily into the Verilog-A hardware description language. An important advantage of this approach is that it allows single or multi-stage delay functions to be selected which meet the accuracy needed to simulate a specific circuit design whilst maintaining minimal simulation run time.

Acknowledgment

The material reported in this paper forms part of the work done in response to the MOS Modeling and Parameter Extraction Working Group (MOS-AK) Verilog-A compact modeling standardization initiative [18].

References

[1] D.O. Pederson. "A Historical Review of Circuit Simulation". *IEEE Trans. Circuits and Systems*, vol. CAS-31, 103-111, 1984.

[2] A.R. Newton, D.O. Pederson, and A. Sangiovanni-Vincentelli, *SPICE Version 2g User's Guide*. Department of Electrical Engineering and Computer Science, University of California, Berkeley, CA, 1981.

[3] B. Johnson, T. Quarles, A.R., D.O. Pederson, and A. Sangiovanni-Vincentelli, *SPICE3 Version 3f User's Manual*. Department of Electrical Engineering and Computer Science, University of California, Berkeley, CA, 1992.

[4] Tedj Tuma and Árpá Bũrmen. *Circuit simulation with SPICE OPUS.* Boston: Birkhuser, 2009

[5] D. Giannopoulos, S. Wong, and A. Lish. "Functional-level simulation of switched-capacitor circuits with nonideal switches and operational amplifiers". in *Proc. of the IEEE Custom Integrated Conference*, San Diago, 1989. p. 21.2/1-21.2/4.

[6] Verilog-AMS Language Reference Manual. Acellera, Version 2.2, 2004. Available http://www.accellera.org.

[7] S. Jahn, and M.E. Brinson "Interactive compact modeling using Qucs equation defined devices". *Int. J. Numer. Model.*, vol. 21, 2008, pp. 335-49.

[8] M.E. Brinson, and S. Jahn, "Compact macromodelling of operational amplifiers with equation defined devices". *Int. J. Electronics,* vol. 96, 2009, pp. 109-22.

[9] M. Margraf, S. Jahn, J. Flucke, R. Jacob, T. Habchi T, M.E. Brinson, et. al. (2003), Qucs (Quite universal circuit simulator). Available http://qucs.sourceforge.net/index.html.

[10] R.J. Faehnrich, (1994), SPICE does digital filters, EDN Access. Available http://www.edn.com/archives/1994/092994/20di1.htm.

[11] See reference [2] section 6.4, p. 8.

[12] Simulating digital filters and systems. Intusoft Newsletter Issue 41, 1995. Available http://www.intusoft.com/nlhtm/nl41.htm.

[13] MATLAB. Natick, MA: The MathWorks Inc., 1986.

[14] GNU Octave. 1992. Available http://www.gnu.org/software/octave/.

[15] A.P Page, and S.L. Smith, "Real-time digital simulation for control systems", in *Proc. of the IEEE*, vol. 54, pp. 1802-12, 1996.

[16] D.A. Calahn, *Computer aided network design.* New York. McGraw-Hill, 1972.

[17] L. Lemaitre, and B. Gu, "ADMS-a fully customizable Verilog-AMS compiler approach," presented at MOS-AK meeting Montreux, 2006. Available http://www.mos-ak.org/montreux/posters/17_Lemaitre_MOS-AK06.pdf.

[18] S. Jahn, M.E. Brinson, H. Parruitte, B. Adouin, P. Nenzi P, and L. Lemaitre, "GNU simulators supporting Verilog-A compact model standardization," presented at MOS-AK meeting Premstaetten, 2007. Available http://www.mos-ak.org/premstaetten/papers/MOS-AK _QUCS_ngspice_ADMS.pdf.

M.E. Brinson received a first class honours BSc degree in the Physics and Technology of Electronics from the United Kingdom Council for Academic Awards in 1965, and a PhD in Solid State Physics from the University of London in 1968. Since 1968 Dr Brinson has held academic posts in Electronics and Computer Science. From 1997 to 2000 he was a visiting Professor of Analogue Microelectronics at Hochschule, Bremen, Germany. Currently he is a visiting Professor in the Faculty of Computing at London Metropolitan University, UK. He is a Chartered Engineer (CEng) and a Fellow of the Institution of Engineering and Technology (FIET), a Chartered Physicist (CPhys) and a Member of the Institute of Physics (MinstP). Dr Brinson joined the Qucs project development team in 2006, specializing in device and circuit modeling, testing and document preparation.

H. Nabijou received his BSc in Electronic Communications, his MSc in Digital Systems and Instrumentation and his PhD in Statistical Signal Processing from UK universities. His research interests include modeling of non-linear stochastic systems. He is currently an academic member of staff in the Faculty of Computing, and a member of the Center for Communication Technology research, London Metropolitan University, UK. He is a member of the IEEE and a fellow of the Royal Statistical Society. Dr Nabijou has recently joined to the Qucs Development Project, as an associate, working on the modeling of signal processing components and algorithms.