Modeling of Generic Transfer Functions in SystemVerilog

Elvis Shera, Dialog Semiconductor
Transfer Function Based Models in an Event-driven Simulator

- No assumption on small/large signal model. The time domain model is ok assuming that the F(s) is valid.
- Unlike a typical Z-Transform, we do not want to do sampling.
Background & Issues

\[ H(s) = \frac{Y(s)}{X(s)} = \frac{b_0 + b_1 s + \ldots + b_m s^m}{a_0 + a_1 s + \ldots + a_n s^n} \]

\[ y(t) = L^{-1}\{Y(s)\} = L^{-1}\{H(s)X(s)\} = h(t) * x(t) = \int_{-\infty}^{+\infty} h(t)x(t-\tau)d\tau \]

The tasks required can then be summarized into three main steps:

1. Inversion of \( H(s) \). \( \text{No Problem} \)
2. Inversion of \( X(s) \). \( \text{Issue} \)
3. Perform in the time domain, the convolution integral.

\[ x(t) = V_1 u(t-t_1) + (V_2 - V_1)u(t-t_2) + (V_3 - V_2)u(t-t_3) + \ldots \]
\[ = \sum_{i=t_0}^{T}(V_{ti} - V_{ti-1})u(t - t_i) = \sum_{i=t_0}^{T} A_{ti} u(t - t_i) \]

Digital representation of an analog signal. Discrete in time and continuous in amplitude.

The unknowns are:
1. \( V_i \)
2. \( t_i \)
From the Mathematics to the Simulation

\[
H(s) = \frac{Y(s)}{L\{\sum_{t=t_0}^{T} A_{t_i} u(t - t_i)\}} = \frac{Y(s)}{\sum_{t=t_0}^{T} A_{t_i} e^{-st_i}}
\]

\[
Y(s) = \sum_{i=t_0}^{T} \frac{A_{t_i} e^{-st_i}}{S} H(s) = \sum_{i=t_0}^{T} A_{t_i} e^{-st_i} \frac{H(s)}{s} = G(s) \sum_{i=t_0}^{T} A_{t_i} e^{-st_i}
\]

\[y(t) = L^{-1}\left\{G(s) \sum_{i=t_0}^{T} A_{t_i} e^{-st_i}\right\}\]

During simulation time is relative. If the input changes (a new step at time \(t_i\)) is like starting simulation from the beginning. This time with new initial condition. This allows us to remove from consideration the factor \(e^{-st_i}\) which is due to the time shift and consider only \(G(s)\) which needs to be inverted just one time.

@ interval [0-1] -> 10g(t)
@ interval [1-3] -> 6 + 16g(t).
@ interval [3-] -> 20 – 45g(t).
Low Pass Filter Example

\[ H(s) = \frac{V_{out}(s)}{V_{in}(s)} = \frac{1}{s + 1/RC} \]

\[ Y(s) = \frac{1}{s + 1/RC} \cdot \frac{A}{s} \]

\[ y(t) = A(1 - e^{-t/RC}) + y(t_0) \]

On a second input change, the model recalculates the new initial conditions and new input variation.
Caveats

Model recalculates the output only when there is a change in the input and this might not be acceptable for the downstream blocks.

A timer is required to insert additional calculation points to more accurately generate the output when the input does not change, so that downstream blocks 'see' a more representative signal.

To keep the simulation performance, a limitation on the number of additional points has been implemented.

Advantages over fixed sampling techniques are:

- it generates just enough points for the output, without compromising model accuracy.
- It stops the timer if $\text{abs}(\text{output} - \text{output}_{\text{infinite}}) < \text{Error}$. Error can be a model parameter.
Comparison - I

Calculated points using the bilinear Z-Transform

Calculated points using the proposed method

Increase sampling frequency to keep enough accuracy.

No more points generated
Comparison - II

Bilinear Transformation

Analog Filter

Lego Filter

Analytic model Overlaps perfectly with analog filter (mixed-signal simulation)

Analytic Model
Comparison - III

Difference between analog filter and the analytic one at the calculated point = 5.74 uV

Fewer points calculated with the proposed approach.

Analog solver would still calculate points at an average of 100ps which is still high in terms of number of events.
Conclusions

• We can model a generic transfer function with high accuracy in the event-driven simulator.

• To optimize model performances in simulation, just enough points are generated for the output in relation to the time constant of the circuit.

• In line with the modeling paradigm: Consider what it does and not how it does it.
Questions ?