Mixed-Mode and Mixed-Domain Modelling and Verification of Radio Frequency Subsystems for SoC-Applications

[A demonstration of the top down approach]

Stefan Joeres
RWTH Aachen
Chair of Integrated Analog Circuits
Sommerfeldstrasse 24
52074 Aachen, Germany
joeres@ias.rwth-aachen.de

Dr. Stefan Heinen
RWTH Aachen
Chair of Integrated Analog Circuits
Sommerfeldstrasse 24
52074 Aachen, Germany
mailbox@ias.rwth-aachen.de

ABSTRACT

The functional verification of radio frequency circuits is the main target of this work. The use of baseband behavioral description models for Bluetooth^{TM} and WLAN receivers is demonstrated. Fundamental simulation comparisons for different implementation levels are made. This paper concludes with a suggestion for an extension of the Verilog-HDL-family to aid SoC designers in their desire to shorten the time to market.

1. INTRODUCTION

At present, the time consumption of mixed-mode circuit simulations for high frequency transceivers is mostly dependent on the carrier frequency, which determines the maximum time step for the simulator to be used. At carrier frequencies from 2.4 GHz and above for e.g. Bluetooth^{TM} and wireless LAN transceivers, the simulation times for large systems with a high number of components exhaust the available calculation power, even from state of the art computer systems. While simulation techniques like harmonic balance or periodic steady state analysis, show good performance in strict RF systems, they principally fail in mixed-mode simulations with digital parts as in common SoC's or single chip transceiver solutions. A simulation of a complete transmission burst at transistor level on the final tape out data base is therefore impossible. A verification of the tape out data base is an important goal to minimize the development costs. Redesigns due to simple but undetected errors could be avoided and reduce the development time to achieve a better time to market.

Therefore, a top down approach for RF-subsystems and its requirements on the EDA-software is proposed. The visibility of this approach is demonstrated for two Bluetooth^{TM} and wireless LAN transceivers. The benefits and demands on new language constructs for present and future behavioral description languages will be shown.

2. PROBLEM DESCRIPTION AND STATE OF THE ART

This paper focusses on two different aspects. One goal is to achieve a functional verification of the final tape out database, the other one to enable long time transient simulations for the calculation of e.g. bit error rates (BER) with an acceptable performance. At present, most system designers only think in terms of BER or even package error rates (PER) which need a huge number of received bits for accurate calculations. The RF designers, creating the schematics and the real Silicon focus mainly on noise figures and gain values of their building blocks. Therefore, the simulations used for system concept engineering and transistor level schematics differ a lot. This gap is usually imperfectly closed by spreadsheets which contain simulated requirements and additional specifications for each block. Another long term target of the presented work is to achieve common executable specifications in a matter that can be used by both RF designers and system engineers.

The long-time transient system simulations, which are required for BER or PER are typically done on abstract description levels. These are mainly languages and tools like Matlab^{TM}, which don't support a functional check neither on the transitor level schematics nor on the tape out database. In contrast to this, the RF designers focus on noise figure simulations and short transient simulations like periodic steady state or harmonic balance, which are simulated with tools like spectre^{TM}.

The RF carrier frequency, with it's short time period, dominates the simulation requirements. The size of the matrix, which is to be solved for each time step, is approximately proportional to the number of nodes used in the circuit. In analog simulations, each transistor requires at least about ten equations to be evaluated. Even a small digital part with only a few hundred transistors, leads to enormous computing demands compared to an analog frontend with typically far less transistors. What essentially is missing, is the functional

0-7803-9352-X/05/$20.00 © 2005 IEEE.

verification of couplings between these two worlds such as gain control loops or even more simple tasks like switching on or off the biasing of sub blocks.

The method to use baseband models of the high frequency components has been investigated for quite some time and the libraries are getting more complex and more complete for different architectures.

Verilog, as one of the major hardware description languages does not have support for complex structures or data types to be passed over single connect wires as desired for continous functional verification. The international standards, which are the base for the software implementations of the EDA vendors, still lack sufficient support in terms of new language constructs to ease the connectivity verification. Upcoming SystemVerilog promises good constructs, but misses analog extensions like in Verilog-AMS.

3. FUNCTIONAL VERIFICATION BASEBAND MODELS FOR TYPICAL RECEIVER STRUCTURES

Complex baseband signals are normaly upconverted in the transmitter to the carrier frequency with a 90 degree phase shift like shown in equation 1.

$$RF(t) = I(t) \cdot \cos \omega t - Q(t) \cdot \sin \omega t. \tag{1}$$

Another representation for this is the real part of the complex envelope with the modulated carrier

$$RF(t) = \Re\Big\{B(t) \cdot e^{j\omega t}\Big\}. \tag{2}$$

This can be written as baseband representation with

$$B(t) = I(t) + jQ(t) \tag{3}$$

$$j = \sqrt{-1} \tag{4}$$

For nearly every simulation, one is only interested in the distortion that this baseband information receives during its ride on the carrier, not on harmonic disturbances of the carrier itself. Most distortions and nonlinear effects occur as simple amplitude modulations, but depending on the signal encryption, phase modulation is also of interest.

To respect both AM-AM and AM-PM conversion effects, the signal is converted to it's representation in polar coordinates as

$$A(t) = \sqrt{I(t)^2 + Q(t)^2} \tag{5}$$

$$\Phi(t) = \arctan \frac{Q(t)}{I(t)} \tag{6}$$

.

Simple nonlinearities for amplifiers are explained in detail in [1].

The only contribution from terms of interest can be expressed as

$$A_{dist}(t) = a \cdot A(t) - \frac{3}{4} c \cdot A^3(t)) \tag{7}$$

where a represents the implementation value of the desired gain (G) in dB

$$a = 10^{\frac{G}{20}} \tag{8}$$

and

$$c = a \cdot \frac{\frac{4}{3}}{10^{\frac{ip3}{10}} \cdot 2 \cdot R_{in} \cdot 0.001}. \tag{9}$$

These expressions are used in the models to express the gain and the third order non-linearities. To simulate limiting effects, the output is clamped to a maximum value for extreme input signals.

The maximum acceptable input Voltage is typically

$$V_{in,max} = \sqrt{\frac{a}{3 \cdot c}} \tag{10}$$

which leads to the maximum output of

$$V_{out,max} = \frac{2 \cdot a}{3} \cdot V_{in,max}. \tag{11}$$

The baseband output is the complex modulation of this as

$$B(t) = A_{dist}(t) \cdot cos(\Phi t) + j\,(A_{dist}(t) \cdot sin(\Phi t)) \tag{12}$$

or expressed as seperate signals

$$I(t) = A_{dist}(t) \cdot cos(\Phi t) \tag{13}$$

$$Q(t) = A_{dist}(t) \cdot sin(\Phi t) \tag{14}$$

One of the main problems one may deal with while using baseband models is the correct signal power. The instant power of an sinusoidal RF signal with the amplitude A at a load of 50 Ohm is

$$P(t) = \frac{A^2 \cdot \cos^2(\omega t)}{50\Omega}. \tag{15}$$

The time average of this can be evaluated to

$$P = \frac{A^2}{50\Omega \cdot 2} \tag{16}$$

Since we omit the carrier frequency for the baseband signals, we miss the factor of 1/2. This must be admitted for in the models when calculating noise powers and gains.

3.1. Example Receiver

In Figure 1, one of the two examples of this paper is shown. The schematic shows the building blocks of a Bluetooth^{TM} low-IF receiver. For the demonstration, the GFSK-signal is generated in a Verilog block and transmitted over a fictional AWGN channel. The received signal is amplified and down-converted to the intermediate frequency. It passes an I/Q crossbar switch and a polyphase filter for channel selection. The filtered signal is again amplified, limited and demodulated using a polyphase structure. The final signal is then digitized and passed to the digital block (not shown).

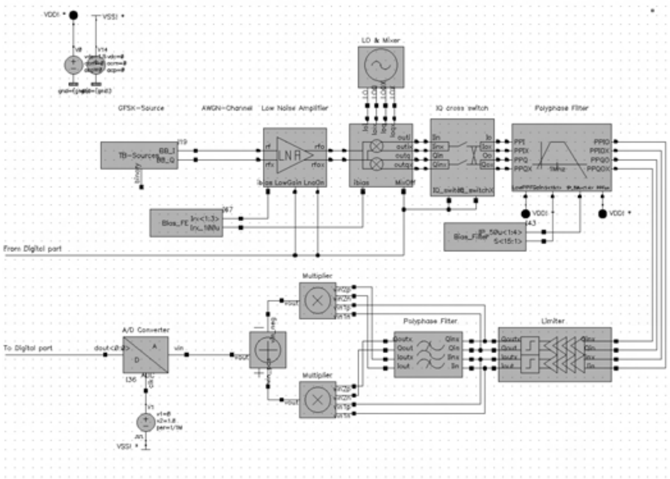


Figure. 1. Schematic View of the Receiver Example

3.2. Low Noise Amplifier

The low noise amplifier is modeled as a differential voltage amplifier. To minimize calculations during runtime, the conversion from typical engineering units (e.g. intercept point 3rd order, noise figure and gain in dB) to the necessary implementation values are done at the initial step of the simulation.

In a first approach, every given functionality from a physically approved model was implemented. This included a high-/low gain switch, variation of the bias current and power down mode during runtime. Although simulations with a periodic steady state analysis are not the intention of functional baseband models, it was decided to test the models for hidden states. This lead to the conclusion, that e.g. the gain switch was later modeled as only switchable during initial start of the simulation. Other implementations without hidden states are possible, but do consume additional calculation time [1].

To feature the verification of additional functionalities like power down mode, power supply connections, checks for necessary voltages and their influences on the output and input, these signals are checked and failures or important changes are reported throughout the simulation.

Noise is modeled as a white noise voltage source and added to each input. Figure 2 shows the noise figure simulations for the transistor level schematic and for the baseband model. It is obvious, that the point of interest is at $2.401GHz$ for the transistor level schematic, but only at $1MHz$ for the baseband model, since the carrier frequency is eliminated. With respect to section 3 the values can be copied from the high frequency to the baseband implementation.

3.3. Mixer

For baseband simulations, the mixer can essentially be reduced to it's gain feature, some third order non-linearities and I/Q mismatch effects. To achieve this, each path is implemented with separate parameters. The gain and limiting effects are then modeled the same way as in the low noise amplifier.

While amplitude mismatch is easily modeled with different gains for the I and Q signal paths, phase mismatch requires additional weighted additions according to

$$B(t) = (I(t) + jQ(t)) \cdot e^{j\omega_{carrier}t} \cdot e^{-j\omega_{LO}t + \phi_{I,Q}} \tag{17}$$

$$= (I(t) + jQ(t)) \cdot e^{\phi_{I,Q}} \tag{18}$$

$$= I(t) \cdot cos\phi_I - Q(t) \cdot sin\phi_I + j\,(I(t) \cdot sin\phi_Q + Q(t) \cdot cos\phi_Q) \tag{19}$$

The real part of equation 19 equals the downconverted in phase component and the imaginary part the quadrature phase component. When dealing with phase changes during runtime, it is important to model the instantaneous phase as an integral part over time, to avoid phase jumps.

For baseband simulations, it would be typical to ommit the frequency conversion itself, since the fundamental frequency of the signal is used. To ease the addition of frequency conversion effects, this is added as optional complex mixer calculations, so that different low IF frequencies can easily be simulated.

Listing 1. Complex upconversion in the mixer

```
 [..]
// intended LO and RF Frequency don't match
if ( frequency_difference >0)
begin
 bound_step(0.125/ frequency_difference );
 tmp  = 2*‘M_PI*idt( frequency_difference ,0);
 tmpi = iout*cos(tmp) − qout*sin(tmp);
 tmpq = iout*sin(tmp) + qout*cos(tmp);
end
 [..]
```

3.4. Polyphase Filter

To further propose the top-down design flow, the polyphase filter was first implemented as a typical ladder-filter implementation with the historic values from Szverev [2].

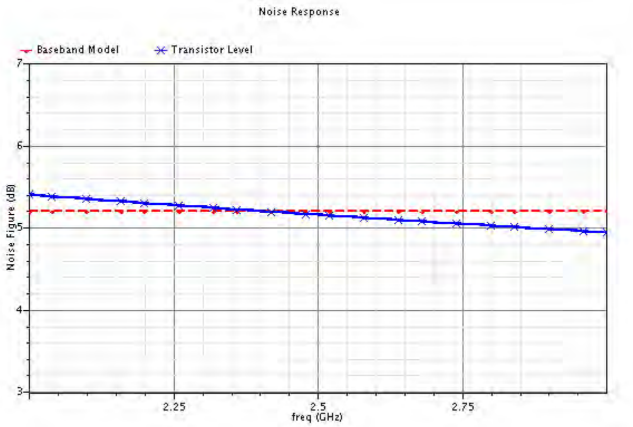


Figure. 2. Simulated Noise Figures of the LNA for Transistor and Baseband Level

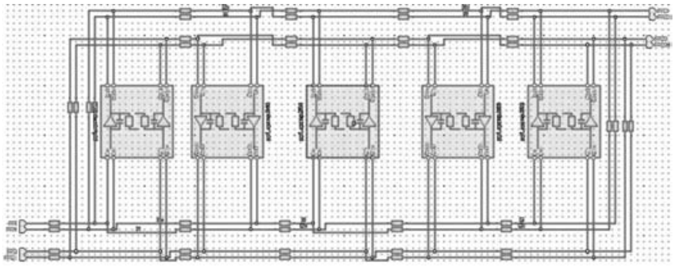


Figure. 3. Schematic of the polyphase filter

The filter structure used in this first design step is shown in Figure 3.

The AC simulation of this filter shows a good frequency response and can be used to determine the necessary order and bandwith of the filter (Figure 4).

As simulation of a possible IP-reuse, it was later replaced by the industrial schematic. To speed up simulations this was simplified by a reduced one with Verilog blocks for the biasing, switch checks and Verilog models for the operational amplifier circuits. Table 1 shows the comparison for the number of elements used in the three circuits. It is clearly visible, that the simplified models will speed up the simulation, not only due to the baseband modeling of the high frequency frontend, but especially by reducing the size of the matrix to be solved.

	Reference	Reduced	Verilog
nodes	1212	71	71
equations	1245	104	114
transistors	894	0	0
capacitors	3332	20	20
diodes	66	0	0
resistors	676	64	64

Table 1. Number of Simulation Components for Different Implementations of the Polyphase Filter

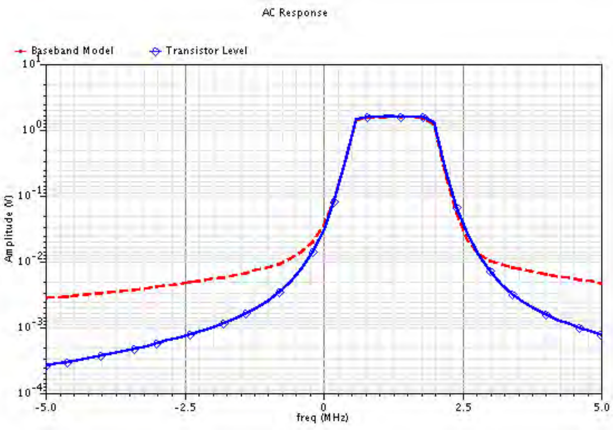


Figure. 4. Frequency response of the polyphase filter

4. WORK-AROUNDS AT CURRENT STATE OF THE ART

As stated before, one of the major problems is the functional verification of the final tape out database. While the baseband models provide enough performance to enable a full verification in an acceptable time frame, one major problem persists. The schematics and connections between baseband models and transistor level schematics must be identical to ensure correct connectivity throughout the design process. But, the number of connections to baseband models and RF models differ in general. While high frequency components just require one input signal (although often differential), even the simplest baseband models require the in phase and quadrature phase signals as separate components.

Differential circuits are commonly used in RFICs, so it was possible to avoid the use of additional wires by putting the I - signal on the positive net and the Q-signal on the negative net respectively. Since noisefigures, gain and intercept points are specified for the given carrier frequency, this is at the moment sufficient, but not really the desired solution.

The low frequency blocks can more easily be switched between the different implementations, since I and Q signals already exist after the first complex downconversion mixer.

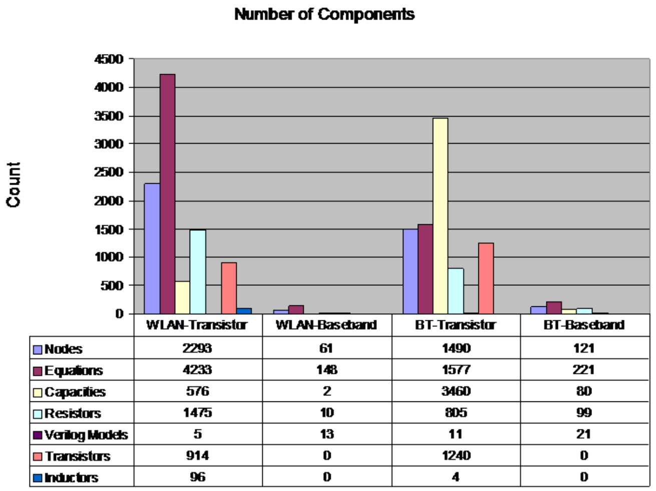
5. SIMULATION RESULTS

The Bluetooth and WLAN receiver structures were simulated using a state of the art multi processor computer system and the current Cadence^{TM} IC 5.10.41 USR 1. The simulated real time was around 10 ms for the baseband structures and far less for the transistor level schematics. For the Bluetooth receiver, the simulation took $3458s$ per μs for the full transistor level schematic compared to only $4.7s$ per μs for the baseband implementation.

For the desired determination of the signal to noise ratio at the input to achieve a BER of 10^{-3}, the simulations took 13 hours for the baseband models of the Bluetooth receiver and would have taken 400 days for the transistor level schematics. The WLAN receiver, as it's modeling was more complete (compare table 5) took only 8.3 minutes with its baseband model. The biggest block which was not completely replaced by a behavioral description was the polyphase filter, which was modeled as schematic with Verilog operational amplifiers. A complete transfer to a Verilog model would enhance the performance again dramatic.

For verification purposes it would be sufficient to ensure the reception of only the first few demodulated bits, which would take around 20 us real time to simulate. This would take only a couple of minutes, so the design flow would greatly benefit from this. Simulations of a complete burst could run over night to ensure full functionality.

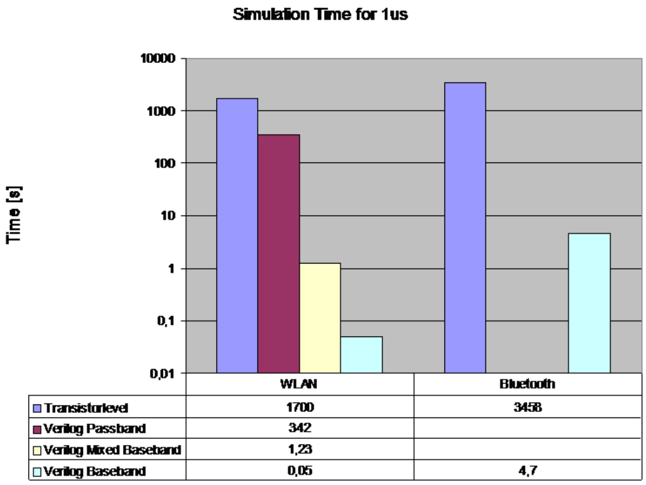
The big speedup from the WLAN receiver, compared between baseband and transistor level simulation shows, that the size of the matrix which is to be solved for each iteration process increases simulation performance by approximately a factor of five. Reducing the number of time steps to be used by eliminating the carrier frequency, additionally im-



	WLAN-Transistor	WLAN-Baseband	BT-Transistor	BT-Baseband
Nodes	2293	61	1490	121
Equations	4233	148	1577	221
Capacities	576	2	3460	80
Resistors	1475	10	805	99
Verilog Models	5	13	11	21
Transistors	914	0	1240	0
Inductors	96	0	4	0

Figure. 5. Number of Components

proves performance by a really large factor of six thousand. The complete table of simulation times is shown in figure 6.



	WLAN	Bluetooth
Transistorlevel	1700	3458
Verilog Passband	342	
Verilog Mixed Baseband	1,23	
Verilog Baseband	0,05	4,7

Figure. 6. Simulation Times

The simulated BER of the Bluetooth receiver was 10^{-3} at a signal to noise ratio of 16.7dB which is in good accordance to the measured silicon.

Figure 7 shows the simulated signals for the receiver. The small glitch at around 18.7 us is one of the decoding errors, but not detected as a bit error, because the digital sampling times do not fall within it's area. The noisy input signal is clearly visible as amplitude modulation at the frontend output.

6. CONCLUSION AND RECOMMENDATIONS FOR HDL

On the one hand, baseband modeling promises efficient ways of simulation speedup, but on the other hand, some information is lost. Conversion effects from different harmonics, which occur through nonlinearities and mixer effects, are not taken care of anymore.

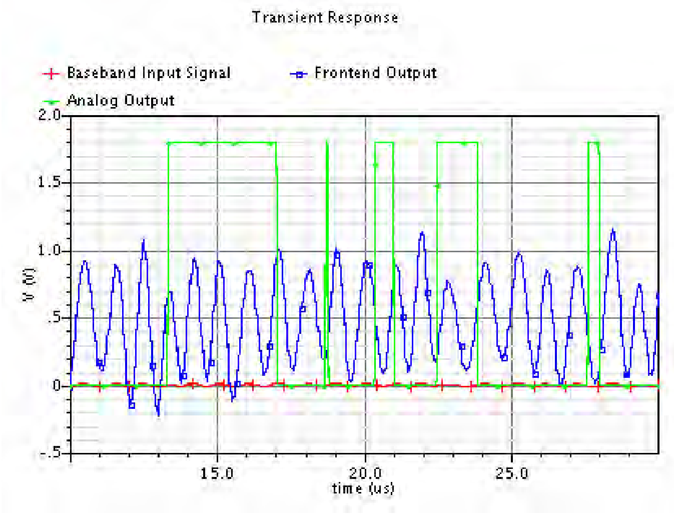


Figure. 7. Transient Simulation

To ease the implementation of such additional information, which are to be transferred between the different simulation blocks of a transceiver, it would be desirable to have complex structures passed over single connections. SystemVerilog as upcoming new language promises a good and effective way of modeling these structures, but misses analog extensions like in Verilog-AMS at the time being.

A comprehensive SoC-verification will require both, a SystemVerilog as well as Verilog-AMS support for the RF components using complex data types in their interfaces. The first of these might be used in a pure digital simulation environment, to verify the RF together with a large digital part, verifying the whole SoC. The scond approach will be needed for mixed-mode simulations in order to design the transistor level of the baseband analog or mixed-signal part.

Listing 2 shows the intended structure for a single frequency wave. It includes the time varying I and Q components, the carrier frequency as a (typical) constant and additionally the phase of the carrier frequency. With these information, every single frequency effect can be modeled.

Listing 2. Typedefinition complex wave

```
typedef struct {
        real I,Q;
        real frequency;
        real phase;
        } complex_wave_t
```

Listing 3 shows the interface structure used to connect two building blocks with an arbitrarily number of complex waves as shown in listing 2. The number of distributed waves is implemented as a variable and can be evaluated by examining the demands of the building blocks through the netlisting procedure.

The waves itself are accessed via interface tasks which pro-

Listing 3. Interface definition

```
interface complex_struct #(parameter size = 0)
                                      (inout wire);
 complex_wave_t[size:0] complex_wave;
  real [0:1] wave;          // differential I/O
 task BB_in (I Q);          // read baseband signal
 endtask:BB_in
 task BB_out (I Q); // write baseband signal
 endtask:BB_out
 task PB_in (I);            // read passband signal
 endtask:PB_in
 task PB_out ( I ); // write passband signal
   // possible :
   // emulate passband output as upconverted
   // addition from the complex BB signals
 endtask:PB_out;
endinterface:complex_struct;
```

vide an easy access without the necessity of dealing with different demands in the building blocks itself. Nevertheless, this requires new ways of writing the models. The interface as it is shown here, has four tasks for accessing the information transmitted over the connection. Both baseband and passband models can be used and access the same information with additional upconversion if necessary.

A typical building block like an LNA can then be implemented in SystemVerilog as shown in listing 4.

Listing 4. SystemVerilog LNA

```
module LNA(complex_struct #(.size=0) LNA_in,
        complex_struct #(.size=0) LNA_out);
 // sample LNA work goes here
 LNA_in.complex_wave[0].read
        (I,Q,frequency,phase);
 LNA_out.complex_wave[0].write
        (I*gain,Q*gain,frequency,phase);
endmodule LNA;
```

The use of additional harmonics for example after a frequency conversion in the mixer could be modeled as in listing 6. As example, this mixer has a feedthrough of the original frequency damped with a certain factor as additional harmonic passed to the next block (e.g. a filter).

Listing 5. SystemVerilog Mixer

```
module mixer(complex_struct #(.size=0) mixer_in,
        complex_struct #(.size=1) mixer_out);
 // sample mixer work goes here
 mixer_in.complex_wave[0].read
        (I,Q,frequency,phase);
 mixer_out.complex_wave[0].write
        (I*gain,Q*gain,frequency- oscillator_frequency ,phase);
 mixer_out.complex_wave[1].write
        (I*damping,Q*damping,frequency,phase);
```

These constructs allow a simulation with baseband models, but lack connectivity to analog signals, used for cosimulations with transistor level models or alike. It would be desirable to have connectivity modules, compatible to VerilogA and based on electrical dimensions. The aim of this paper is to stimulate a discussion on the extension of the Verilog-AMS standard.

7. ADDITIONAL AUTHORS

Additional authors:
Andre Kruth, RWTH Aachen
Andreas Neyer, RWTH Aachen
Soeren Sappok, RWTH Aachen
Dr. Ralf Wunderlich, RWTH Aachen

8. REFERENCES

[1] K. Kundert. Hidden state in spectrerf. *www.Designers-Guide.org*, 1c, May 2003.

[2] A. Zverev. *Handbook of Filter Synthesis.* John Wiley and Sons, New York, 1967.