Mixed Signal System Design Verification Accelerated With Detector-Based Diagnostic Method

D. Dammers
Dolphin Integration
Bismarckstr. 142 a
47057 Duisburg,
Germany
+49 203 3062250
mems@dolphin-integration.com

C. Domingues
Dolphin Integration
39, avenue du Granier
38242 MEYLAN
France
+33 476 411096
mems@dolphin-integration.com

D. Schollän
Dolphin Integration
Bismarckstr. 142 a
47057 Duisburg
Germany
+49 203 3062250
mems@dolphin-integration.com

L. M. Voßkämper
Dolphin Integration
Bismarckstr. 142 a
47057 Duisburg,
Germany
+49 203 3062250
mems@dolphin-integration.com

ABSTRACT

Mixed-signal simulation of complete systems, i.e. pure analog/digital systems or electronics with its attached peripherals, such as sensor and actuator systems, already have a firm place in today's design process. While the digital part verification has gained a speed increase through the use of Accellera's Property Specification Language (PSL), the analog part suffers from not being supported by this language. To speed up the verification of the analog part innovation is needed. Based on assertion concepts, this paper presents an innovative methodology for the verification of mixed-signal circuits. Electrical validation of a sensor amplifier design serves as demonstrator for this methodology.

Keywords

mixed-signal design rule check, verification rule check, detectors, specification, assertion

1. INTRODUCTION

Digital hardware design verification using the standardized Property Specification Language (PSL, IEEE 1850) is widely spread nowadays in HDL development design flows. Here, the PSL language is included as comments in VHDL and Verilog model source code, or as verification units. With these instructions, dedicated specification rule checks can be performed during simulation. The PSL language is notably restricted to event driven aspects.

However, the opportunities to integrate more and more functionality into a (micro) System-on-Chip (SoC) heavily call for advanced mechanisms in the verification of complete system simulations, i.e. to additionally take into account analog and mixed-signal aspects. This paper describes a methodology to automatically check complete mixed-signal designs using a library of detectors implemented in VHDL-AMS.

The methodology is described in chapter "The Detector Methodology". In chapter "The Demonstrator", the application of the methodology is presented. "Conclusion" summarizes the advantages of the proposed methodology for mixed signal circuit design. And in chapter "Outlook", we present in which fields the detector methodology could bring benefits to mixed-signal circuit designers and system integrators.

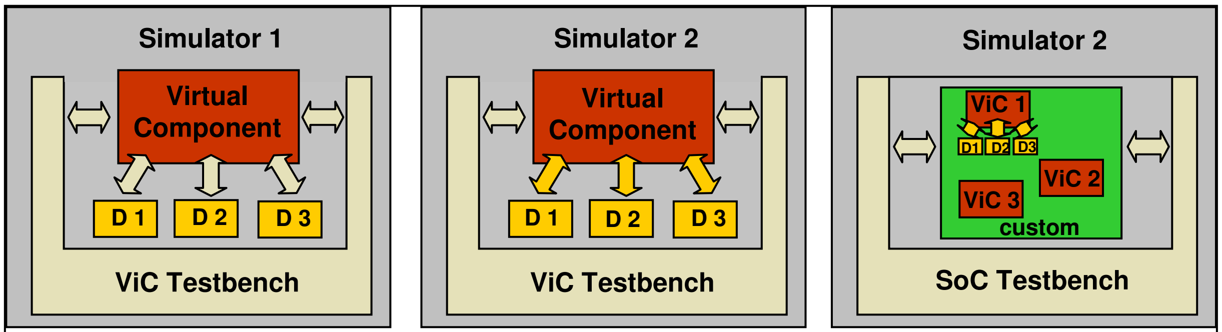
2. THE DETECTOR METHODOLOGY

The vision that led to the detector library implementation was to improve the design reliability and security through automated design verifications without manual waveform analysis.

Detectors are built to observe specific system characteristics, e.g. to measure currents, voltages, frequencies, slopes, delays, jitters etc., on the condition not to influence the system behavior in the simulation. So the detectors are passive "observers" with respect to the circuit. But they are active with respect to the designer: detectors announce online, during simulation, when signals violate specification rules and write these events in report files for further analysis. Depending on the severity level set up for the alerts, the simulation can be aborted, paused or continued.

While implementing the detectors, special focus was set on being compatible with any possible applications, for instance through parameterization in order to be adaptable to different specifications.

Besides the analog ports for attaching the detectors to the analog/mixed signal circuit nets to be observed, every detector has a logic input port to attach an enable signal and a logic output port to supply a trigger signal to the system. So, the user is able to build more complex specification rule checkers with help of the basic detectors already defined in the library.



Circuit Development

Virtual Component Developer provides Virtual Component (ViC) + Detectors + Testbench reduced thanks to detectors

Circuit Transfer

The reduced Testbench must be adapted to the target simulator; the detectors remain untouched

SoC Integration

Customer integrates different ViCs together with custom circuitry into a SoC. The detectors observe the compliance of the specifications in the changed environment.

Figure 1: Detector featured Mixed Signal Design and Circuit Transfer.

The standardized hardware description language VHDL-AMS was chosen to build the library of detectors. This ensures that designers using diverse simulators supporting multiple languages (Verilog(AMS), VHDL(AMS), SPICE...) can integrate detector components assembled using this library. Figure 1 illustrates the application of the detectors in circuit design, transfer and SoC Integration. For Circuit design, they are helpful in speeding up the verification process through automatic specification rule checks. The detectors replace simulator specific testbench code with independent detectors embedded in the model. Thanks to the use of VHDL-AMS, the same detectors can operate in different simulator environments and testbenches so that they can be delivered to a customer to check if the component behaves as expected in the changed environment. For SoC Integration, detectors enable checking that component integration rules are respected, verifying that specifications are met and detecting unexpected component interactions when integrated in a complex SoC. The main benefit is that SoC Integrators do not have to perform manual, and error prone, checking of a huge amount of analog waveforms in order to verify that specifications are met and thereby get a reliable feedback. All the specification rules that are represented by the detectors are checked and documented automatically during the simulation.

The advantage of using a library of detectors, rather than of using a special language, is that already verified model libraries remain untouched. Detectors can easily be placed inside the schematics of the design to verify and of course compounded specification rule checkers can be reused independently from any device/circuit model.

3. DEMONSTRATOR

To demonstrate the advantages of this new methodology we apply voltage detectors on metal oxide semiconductor (MOS) devices in a sensor rail-to-rail preamplifier, which is used in voltage follower configuration in order to have a high impedance input, see Figure 2 and Figure 3 The preamplifier power supply voltage is 3.3V. The output differential voltage OUTDIFF, is obtained using two single preamplifiers, Figure 4. OUTN (1) and OUTP (2) represent respectively the voltage of the negative and positive output. The frequency of the targeted application is between DC and 1 kHz.

The MOS devices of the preamplifier input differential pair are 1.8V devices in order to lower noise and gate to source voltage (VGS) as input voltage is at 3V.

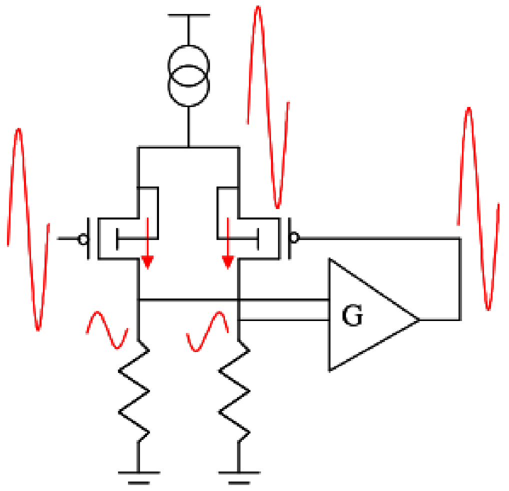
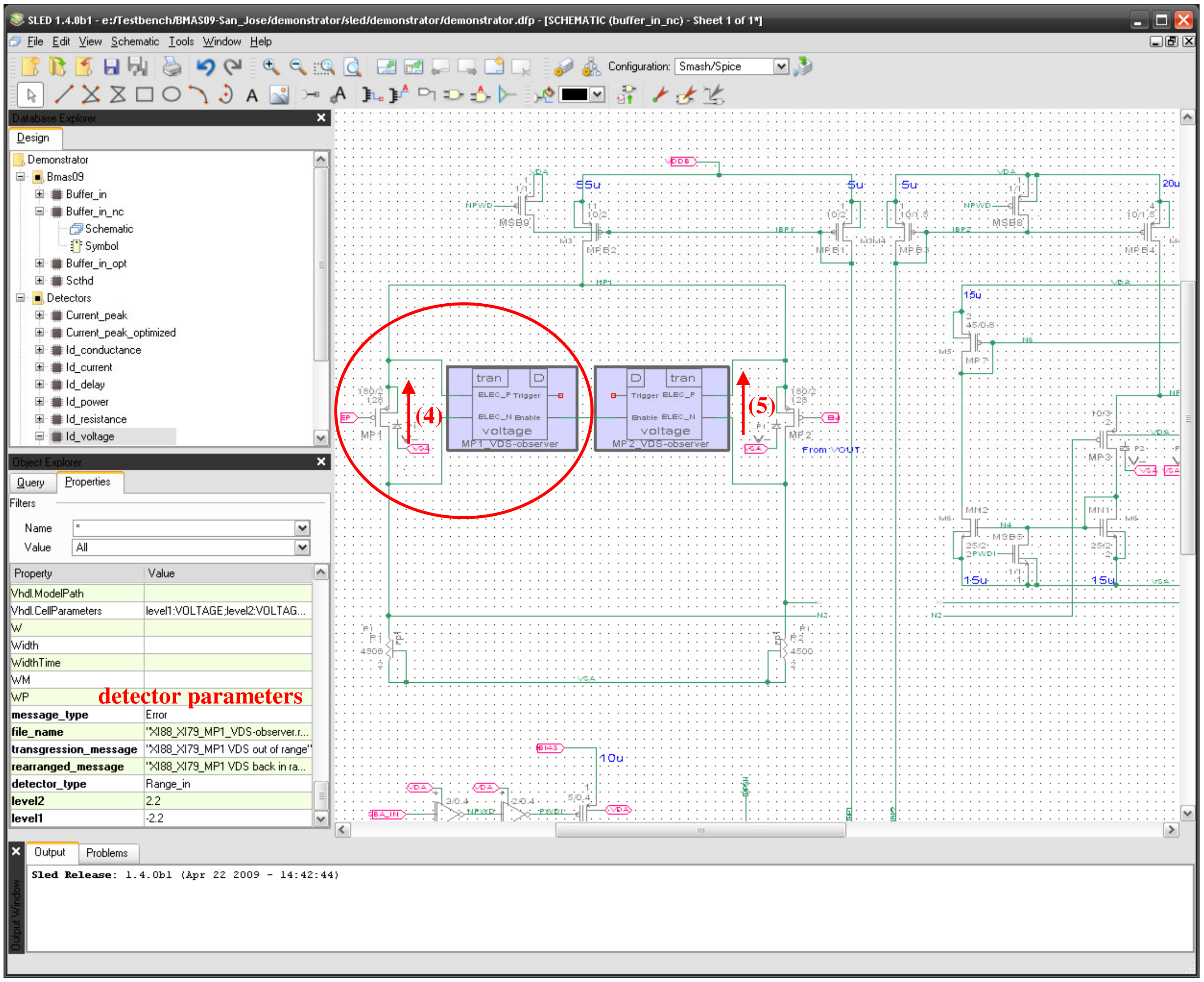


Figure 2: First stage of the pre-amplifier.



SLED 1.4.0b1 - e:/Testbench/BMAS09-San_Jose/demonstrator/sled/demonstrator/demonstrator.dfp - [SCHEMATIC (buffer_in_nc) - Sheet 1 of 1*]

Configuration: Smash/Spice

Filters

Name: *

Value: All

Property	Value
Vhdl.ModelPath	
Vhdl.CellParameters	level1:VOLTAGE;level2:VOLTAG...
W	
Width	
WidthTime	
WM	
WP	
message_type	Error
file_name	"XI88_XI79_MP1_VDS-observer.r...
transgression_message	"XI88_XI79_MP1 VDS out of range"
rearranged_message	"XI88_XI79_MP1 VDS back in ra...
detector_type	Range_in
level2	2.2
level1	-2.2

```
Sled Release: 1.4.0b1 (Apr 22 2009 - 14:42:44)
```

Figure 3: Buffer_in sub-circuit of the sensor amplifier design.

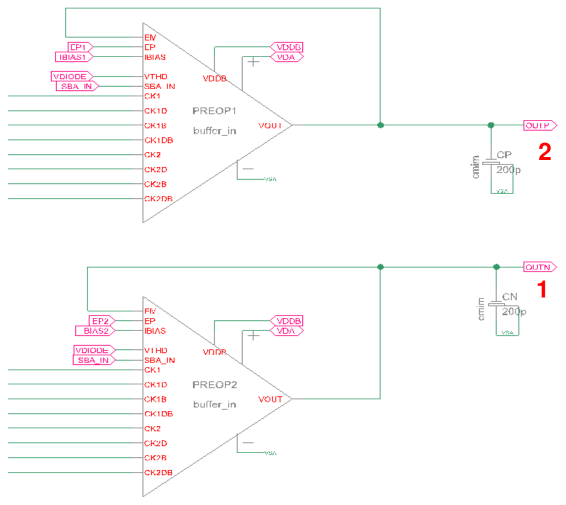
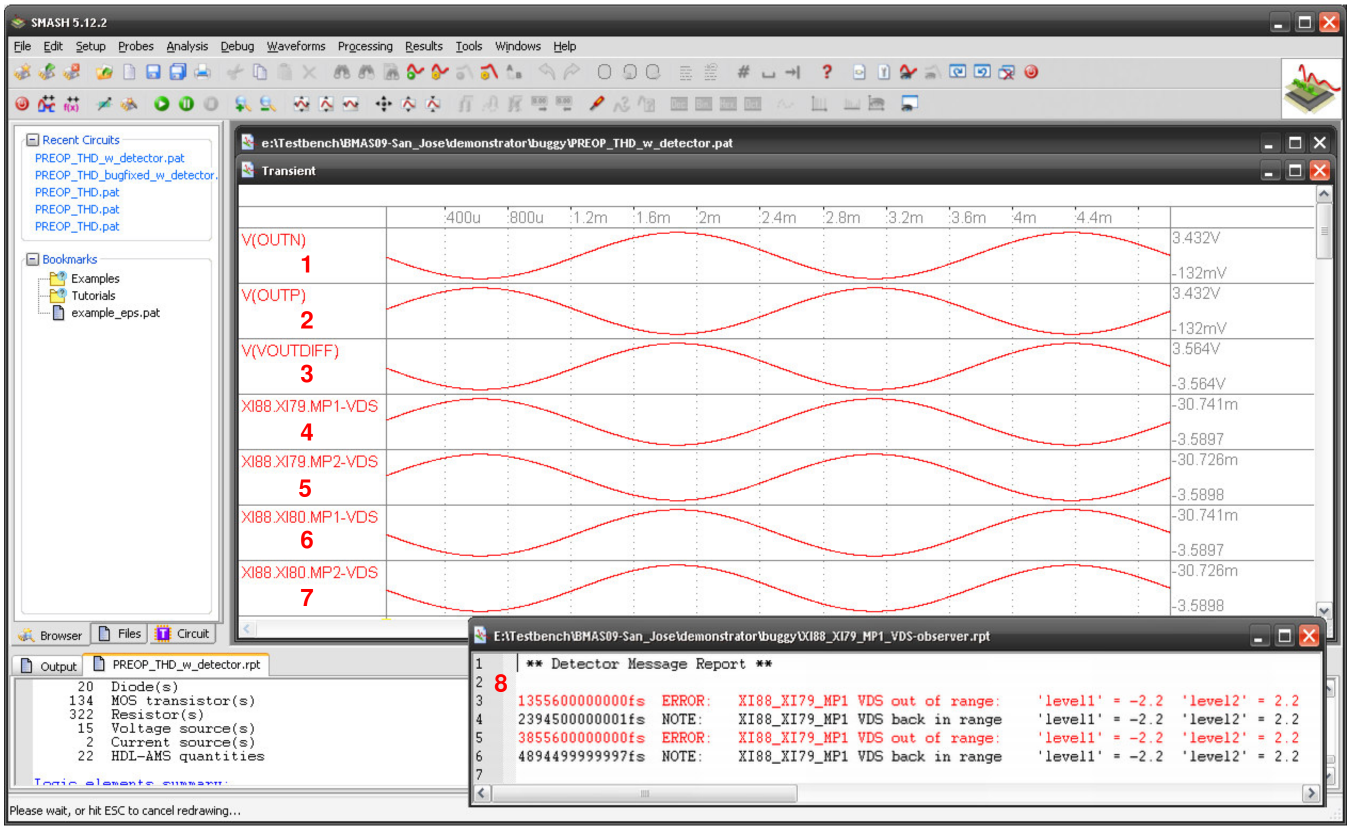


Figure 4: Buffer_in instances.

Due to their lower oxide thickness, it has to be verified that the drain to source voltage (VDS) and gate to source (VGS) of these 1.8V MOS devices is always below 2.2V. The voltage detectors observe VDS and report rule violations immediately. Figure 3 shows the “buffer_in” sub-circuit of the preamplifier design in the schematic editor SLED[1]. The two detectors (highlighted) are connected to MOS FETs to observe their drain source voltages. These simulated voltages are shown in trace (4) and (5) in Figure 5.

Also shown in Figure 3 are the parameters of the first detector, see lower left pane. Listing 1 shows the corresponding VHDL-AMS entity.

[1] **S**chematic **L**ink **ED**itor, EDA tool of Dolphin Integration http://www.dolphin-integration.com



```
 20  Diode(s)
134  MOS transistor(s)
322  Resistor(s)
 15  Voltage source(s)
  2  Current source(s)
 22  HDL-AMS quantities

Logic elements summary:
```

```
** Detector Message Report **

1355600000000fs  ERROR:  XI88_XI79_MP1 VDS out of range:   'level1' = -2.2  'level2' = 2.2
2394500000001fs  NOTE:   XI88_XI79_MP1 VDS back in range   'level1' = -2.2  'level2' = 2.2
3855600000000fs  ERROR:  XI88_XI79_MP1 VDS out of range:   'level1' = -2.2  'level2' = 2.2
4894499999997fs  NOTE:   XI88_XI79_MP1 VDS back in range   'level1' = -2.2  'level2' = 2.2
```

Figure 5: Preamplifier simulation with detector report.

The detectors parameters are the upper and lower voltage limits to detect if the observed voltage is below, above, in range or out of a specific range. Messages of the detectors can be set for the case that a specification violation occur and for the case that the observed voltage is back in specification The severity level affects can be set to note, error or fault, which then determines whether the simulation will be aborted on violation or not. The last parameter is the name of the report file.

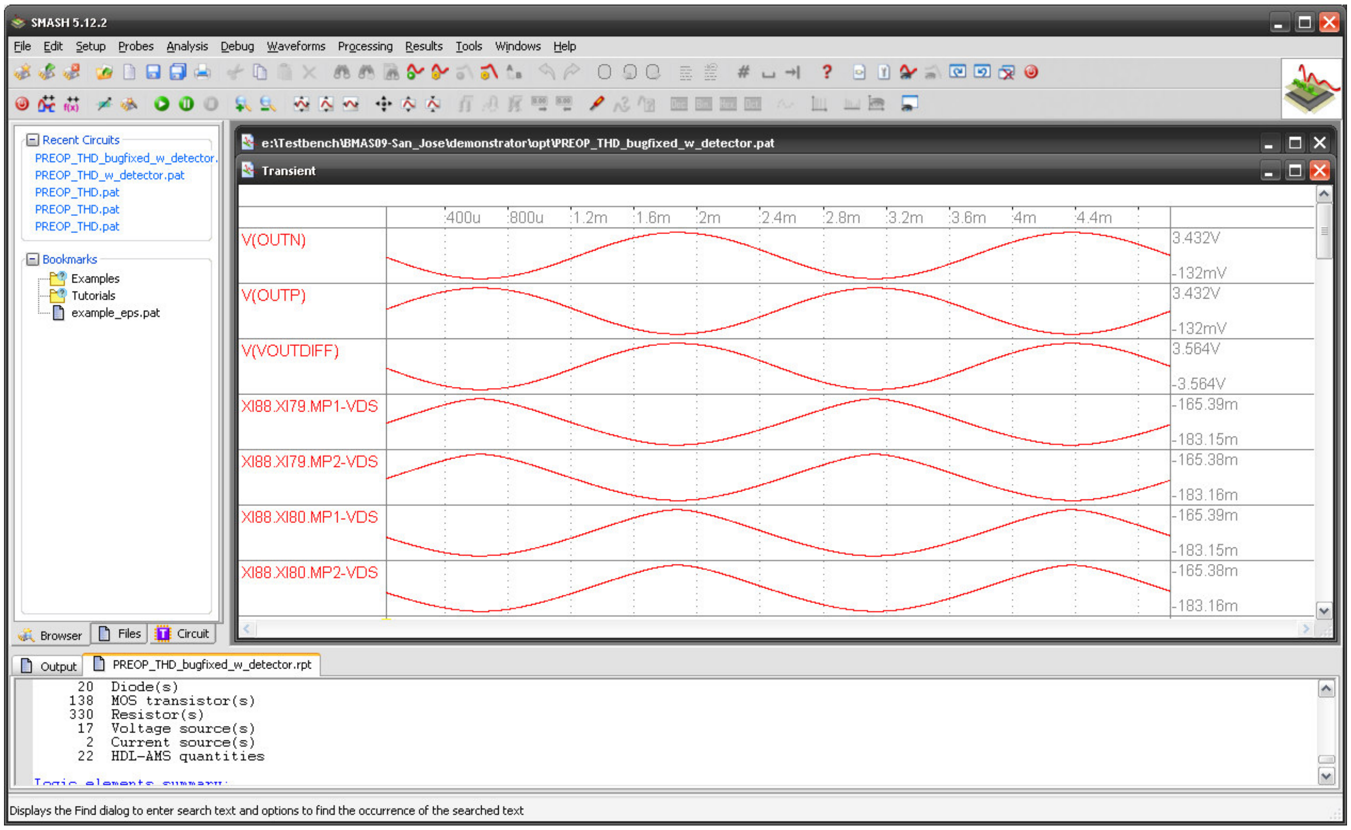
The simulation of the preamplifier testbench is shown in Figure 5 in the simulator SMASH[2]. Trace (1) shows the voltage OUTN of the preamplifiers, whereas trace (2) displays the voltage OUTP and the difference between OUTP and OUTN is shown in trace (3). Trace (4) shows the detectors "observing-voltage": The drain source voltage of the MOS-FET "XI88.XI79.MP1" shown in Figure 3, marked by the red circle. Also shown in Figure 5 is the detector observing report (8), which reports two electric rule violations. The traces (5) to (7) display detector "observing-voltages" for other MOS devices that also violate electrical rules. The report is stored in a separate file for later analysis and use, for instance for documentation.

Thanks to the information provided by the detector reports, it is clear that the design has to be adapted to avoid voltage amplitudes that will burn the MOS devices. Figure 7 shows the optimized design; it consists on adding cascodes[3] devices, which gate voltage is controlled to maintain the drain source voltage of the input differential pair constant. Figure 6 shows the simulation of the optimized preamplifier. One can see that rule violations no longer occur.

For this special case, the simulation without detectors lasts 65 seconds. With detectors, the simulation time increases to 74 seconds which means that it lasts 9 seconds or 13.5% longer. It is difficult to estimate the time used by the designer to observe all signals, but certainly it will cost more than the simulation overhead. However, significantly more important than the saved time, the detectors, in opposite to a tired designer, unfailingly detect the fault!

[2] Single kernel, mixed-signal and multi language simulator by Dolphin Integration

[3] The MOS cascodes are used as dynamical voltage shifters.



```
     20   Diode(s)
    138   MOS transistor(s)
    330   Resistor(s)
     17   Voltage source(s)
      2   Current source(s)
     22   HDL-AMS quantities

  Logic elements summary:
```

Displays the Find dialog to enter search text and options to find the occurrence of the searched text

Figure 6: Simulation of the optimized pre-amplifier

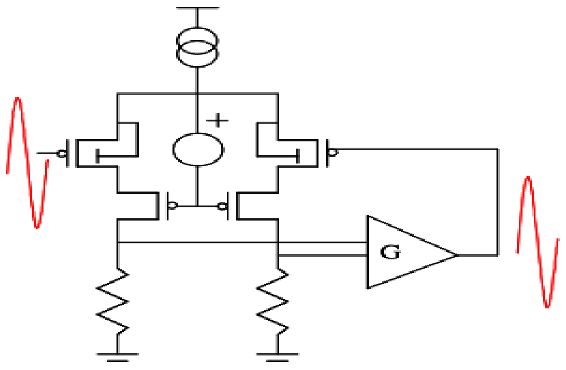


Figure 7: Optimized design.

The configurable detector messages allow easy and reliable location of specification rule violations at circuit elements. For further analysis, or in order to document the simulation runs, the designer has the possibility to enable the detector log-file, where it is also possible to configure the log-text. The log-files are simple text files in order to be accessible from any other application. This means that detectors facilitate the observation of important signals in the circuit, so that design productivity/security is noticeably increased.

4. IMPLEMENTATION

The shown voltage detector is one piece of a VHDL-AMS model library. The advantage of this library, in opposite to using assertion checks in "device-models", is the reusability. Since the detector models possess an enable input for activation and a trigger output they can be assembled to build more complex detectors. Currently the library comprises detectors to observe voltage, current, power, resistance, conductance, voltage at a specific time, current at a specific time, delay, frequency, frequency jitter, ratio, slew rate, rise time, fall time and generic edge. These manifold detectors make use of assertion statements to announce their alerts and report them. Special care was set to not influence the systems behavior while implementing the detectors as demonstrated in Listing 1, where a part of the implementation of the voltage detector in VHDL-AMS is shown. One can see that the model has two electrical terminals, "elec_p" and "elec_n", where only the across quantity i.e. the voltage v_test, is used to detect if v_test is above a limit, see "PROCESS (v_test'ABOVE(level1), v_test'ABOVE(level2),Enable) IS". PROCESS(transgression) generates the assertion message, as well as the report file. Depending on the severity level, the simulation will be continued, paused or aborted, see line “message_type : SEVERITY_LEVEL := ERROR”.

Listing 1: Implementation of the voltage-detector.

```
ENTITY id_voltage IS
  GENERIC
   (level1 : VOLTAGE := 0.0;     level2 : VOLTAGE := 0.0;
    detector_type : DETECTION_TYPE := Range_in;
    message_type : SEVERITY_LEVEL := ERROR,
    transgression_message: STRING := " V out of range";
    rearranged_message : STRING  := "V back in range";
    file_name : STRING  := "V-observer.rpt ");
  PORT (TERMINAL elec_p, elec_n:  ELECTRICAL;
    SIGNAL   Enable:  IN BOOLEAN;
    SIGNAL   Trigger: OUT BOOLEAN);
END id_voltage;
ARCHITECTURE voltage OF id_voltage IS
  QUANTITY v_test ACROSS elec_p TO elec_n;
  SIGNAL   transgression   : BOOLEAN := FALSE;
  SIGNAL   msg_type        : STRING(1 TO 10)  := " WARNING: ";
BEGIN
PROCESS(v_test'ABOVE(level1),v_test'ABOVE(level2),Enable) IS
  BEGIN
  IF Enable THEN
    IF level1<level2 THEN
      IF v_test>level2 THEN
        CASE detector_type IS
          WHEN ABOVE_1  => transgression <= TRUE;
          WHEN BELOW_1  => transgression <= FALSE;
          WHEN RANGE_IN => transgression <= FALSE;
          WHEN OTHERS   => transgression <= TRUE;
…
END PROCESS;
PROCESS(transgression) IS
  BEGIN
    IF transgression THEN
      IF detector_type = ABOVE_1 OR detector_type = BELOW_1 THEN
        write_detector_messages(file_name => file_name, message   =>
time'image(NOW) & "fs" & character'VAL(9) & msg_type& transgres-
sion_message & character'VAL(9) & " 'level1' = " & real'image(level1) &
character'VAL(9) & character'VAL(9) & ".   ");
        ASSERT FALSE REPORT time'image(NOW) & "fs" & charac-
ter'VAL(9) & msg_type & transgression_message & character'VAL(9) & "
'level1' = " & real'image(level1) & character'VAL(9) & character'VAL(9) &
".   " SEVERITY message_type;
…
```

5. CONCLUSION

It has been shown that with the use of the proposed methodology, the designer is able to create and compose specification rule checkers by using the parameterized basic detectors. Critical design parts can be observed continuously. During simulation, the detectors reliably check whether the design operates in its specifications and raises exceptions otherwise. Consequently, the verification phase can be automated which avoids error prone manual analysis of signal traces.

The compliance of the design specification, and therefore the overall functionality of a circuit, can be totally observed with dedicated detectors. This increases the comprehension of the design and of the influence of certain design parameters on the functionality of the circuit, thereby increasing design robustness.

All this increases designer's productivity and ensures design security through an accelerated automatic checking and reporting of important events. Since the detectors are implemented in a standardized hardware design language (HDL), they guarantee the compatibility of separate application schematics and different simulators, as well as minimize efforts in creating and embedding specification rule checks independently of the overall testbench.

6. OUTLOOK

The detectors offer diagnostic support in several fields. For mixed-signal circuit design, detectors can be placed at internal nets to observe specification violations. (Multi Level) model calibration is supported by the detectors through using a sequencing method to adapt model parameters to converge to measurements or simulations on lower levels. Circuit optimization is improved while using the detectors with a sequencing method to optimize circuit parameters. Design robustness tests can be enhanced through the use of detectors with a sequencing method to ensure that the IP stays in its specification. Circuit transfer efforts can be reduced through providing virtual sockets with detectors on IP’s ports to observe specifications of IP independently from HDL/simulator. Equivalence checking provides the possibility to check the equivalence between two models of the same or different level of abstraction and two models of the same or different language.

7. REFERENCES

[1] S. Max, "Fast Accurate and Complete ADC Testing", Proc. of the IEEE ITC, 1989, pp. 11 1-1 18.

[2] K. Arabi and B. Kaminska, "Parametric and Catastrophic Fault Coverage of Analog Circuits Using Oscillation-Test Methodology," IEEE VLSI Test Symp., 1997, Monterey, pp. 166-171.

[3] Arabi and B. Kaminska, "Oscillation Built-In Self Test (OBIST) Scheme for Functional and Structural Testing of Analog and Mixed-Signal Integrated Circuits“, Proc. of the IEEE ITC, 1997, pp. 786-795

[4] The detector library was developed and partly co-financed in the frame of the EU supported, regional funded project EMSIG (Development and transfer platform for the industrialization of mixed-signal circuits, FKZ 005-0604-0020). The reports and documentation of this project is part of the used literature