A Symbolic Core Approach to the Formal Verification of Integrated Mixed-Mode Applications

Stefan Hendricx and Luc Claesen
Imec vzw/**K**atholieke **U**niversiteit **L**euven
Kapeldreef 75, B-3001 **H**ev**erlee** (**B**elgium)
e-mail: **hendricx@imec.be**

Abstract

In the past, formal verification – the promising alternative to simulation-based verification – has primarily been applied to digital system designs. *D*espite the ever-growing importance of integrated mixed analog/digital systems, hardly any formal approaches have been introduced to verify such designs. In this paper, a preliminary study of a symbolic modelling techni*q*ue is presented, which allows us to formally verify the functional correctness of integrated mixed-mode systems. *T*he usefulness of the approach has been demonstrated by verifying the Smart*P*en*TM*, a practical integrated mixed-mode application.

1. Introduction

***T*he increasing financial/economic conse*q*uences linked to the introduction of faulty electronic systems into their field of operation [5] have, together with their ever-growing complexity and impact in our modern society, resulted in the fact that a crucial part of the design process of any VLSI system is nowadays dedicated to proving its correctness. In case of industrial-commercial applications such correctness analyses are, up to now, mostly performed using traditional simulation-based verification techni*q*ues. However, simulation-based approaches have several well-documented disadvantages; for most applications, they are non-exhaustive, time-consuming and they typically call for a high level of user-tool interaction.**

Formal verification, the promising alternative to the simulation-based methods, is often credited for offering solutions to the disadvantages mentioned above [9]. Among others, the most important benefit of a formal approach is that it allows exhaustive design-verification within a more-or-less acceptable time-frame. Not surprisingly, a lot of research is currently focused on applying formal techni*q*ues for the verification of large circuits [10, 11]. *T*he systems under consideration are primarily digital in nature, whereas formal approaches for integrated mixed-mode systems are left nearly unexplored [6]. However, integrated mixed-mode system applications are becoming ever more important in our society; consider for instance pace-makers, electronic hearing-aids, biological implantable sensory systems and many others.

In order to demonstrate the functional correctness of an integrated mixed-signal system at an abstract level, we devised a straightforward symbolic modelling and verification techni*q*ue [2]. In this paper, a preliminary illustration of our symbolic approach is presented. Section 2 shows a real-life mixed-mode application – the Smart*P*en™ – to which our approach has been successfully applied. *T*he underlying concepts of the symbolic core approach are discussed in section 3. Subse*q*uently, sections 4 and 5 deal with a specific verification experiment: the formal verification of a mixed-mode ASIC of the Smart*P*en™ application. Conclusions are drawn in section 6, which also sketches possible future research activities.

2. *T*he Smart*P*en™, a mixed-mode application

***T*he Smart*P*en™ is a new computer input device, developed and currently being improved by the Integrated MicroSystems *D*esign Group of Imec, in association with LCI–Smartpen. Essentially, the smartpen device aims to improve existing human-computer interaction by offering users both the advantages of a conventional writing pen and those of electronic input devices such as a mouse or writing tablets. In addition to using the smartpen in the old-fashioned way – for instance to write some text in a journal or on a sheet of paper – the smartpen can be used to enter specific user-data in computer-applications. *T*he user-data entered with the pen may relate to cursor-positions, in which case the smartpen is used as a kind of**

ED&TC '97 on CD-ROM

electronic mouse, or the data may relate to the actual information (e.g. sentences, drawings, signatures…) written with the pen on paper. *T*he latter alternative, capturing a written image in 'electronic ink', is especially useful when a direct electronic record of the written information is re*q*uired. Biometric signature verification [3,4] is one practical example in which such electronic records can be used.

Inherently, the smartpen is an intricate microsystem that combines mixed analog/digital electronics (ASICS) with physical sensors (force, tilt and acceleration sensors) and wireless communication. A schematic representation of the different components of the smartpen prototype is given in figure 1, together with a photographic view of one of its mixed analog/digital ASICs. A more detailed account on the smartpen design and its applications is presented in [1].

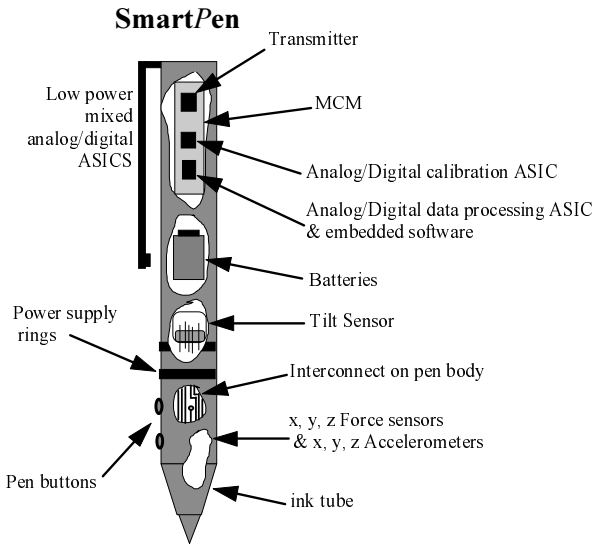


Figure 1a: Schematic representation of the Smart*P*en^{TM} prototype

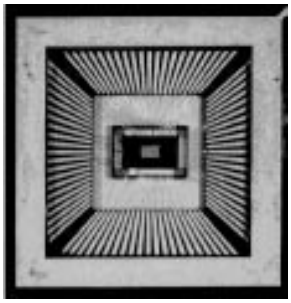


Figure 1b: an integrated mixed-mode ASIC of the smartpen

*T*he relationship between design complexity and an increased probability of introducing undesired errors or faults into a design is a well known one. Considering the complexity of the smartpen as an integrated microsystem, together with the delicate nature of its application field, ensuring the (functional) correctness of the final design is of the utmost importance. In order to demonstrate the correct functionality of the electronics of the smartpen, a symbolic modelling approach was developed for the symbolic verification of integrated mixed analog/digital systems [2]. In the subse*q*uent sections, the basic concepts of the verification approach are explained and a pragmatic verification example is presented.

3. Symbolic core approach to mixed-mode verification

In the past, the terminology "digital core approach to mixed-signal simulation" was often used to classify those simulation-based verification techni*q*ues which make use of a digital simulation kernel to verify – i.e. 'to simulate' in this context – mixed-mode designs [6]. For that purpose, the functional behaviour of the analog components had to be modelled in a digital manner.

Instead of using digital models for the analog components, the approach in this paper will model them using a symbolic formalism. A symbolic evaluation or simulation tool [7, 8] can then be employed to perform the actual formal proof. Accordingly, our verification approach can be labeled a "symbolic core approach to (formal) mixed-mode verification".

*T*he goal of our symbolic verification techni*q*ue is to prove the correctness of the abstract, weak observable (functional) input-output behaviour of a mixed-mode system. In our verification efforts, we initially do not consider the detailed dynamic behaviour – e.g. fre*q*uency response, timing aspects and others – of the systems to be verified. Neither will analog design issues as EM-interference, supply or ground coupling and transmission-line effects be the object of our approach. Even though each of these issues is intrinsically very important – in fact, they are what usually makes mixed-signal design so complicated – we are of the opinion that it is even more important to first demonstrate the functional correctness at a more abstract level. Without doubt, checking a given design at a detailed physical level is unavoidable, but time-consuming resources may well be wasted if the design itself is de facto incorrect with respect to its functional specification. *T*hus, both time and resources can be saved by first employing a simple and straightforward verification techni*q*ue to check if the abstract functionality of the design is free of faults. Once – probably after a number of iterative re-design and verification steps – the design proves to be correct at the abstract level, the verification process can proceed at a

more detailed level of description, using more dedicated tools as for instance SPICE. The process of successively lowering the level of abstraction during design validation is also quite common practice in traditional simulation-based verification approaches.

This section now deals with the specific concepts underlying our symbolic modelling and verification approach for integrated mixed analog/digital systems.

3.1 Symbolic representation of analog signals

A suitable symbolic representation of the analog signals involved is needed when symbolic models for the functional behaviour of the analog components are to be derived. Similar to the digital bit-vector representation for analog signals in digital core simulations, symbolic bit-vectors can be used for the symbolic modelling and verification approach discussed here.

Representing analog signals by symbolic vectors of 8 bits long – different lengths can be used, depending on the desired accuracy of the representation – allows us to 'model' the complete analog voltage range by $x_7x_6x_5x_4x_3x_2x_1x_0$ (x_i is a Boolean variable, $i = 0..7$). Similarly, 00000a00 will denote both the voltages 0V and 0.078V, if 5V (0V) is encoded as 11111111 (00000000) and a is a Boolean variable.

3.2 Symbolic models for analog components

The symbolic models for the analog components describe the functional behaviour in terms of the symbolic representations of the signals involved, and not in terms of the physical entities themselves. Figure 2 illustrates the symbolic modelling of analog signals.

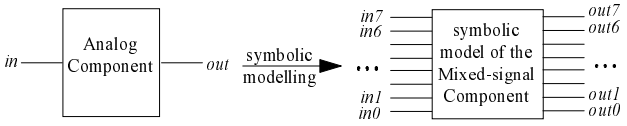
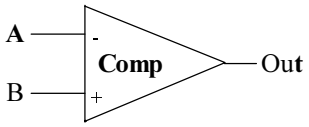


Figure 2: symbolic bit-vectors model the analog signals involved

The next subsection illustrates an example dealing with the construction of a symbolic model for one specific component – an analog comparator.

3.3 Example: an analog comparator

To illustrate the symbolic modelling approach, this subsection shows how the symbolic model for an analog comparator can be constructed. The functional behaviour of this component is quite simple: it acquires and 'compares' two analog voltages A and B and a digital output signal is generated, whose value indicates which of the 2 inputs is larger (see figure 3).



Out	condition
1	$A < B$
0	$A > B$
X	$A = B$

Figure 3: Functional behaviour of an analog comparator with analog inputs A and B and a digital output Out.

In order to model the above behaviour symbolically, the analog signals involved must first be represented in terms of the symbolic bit-vector representation. Suppose that the analog input voltages are described by symbolic bit-vectors $A_n(n:0)$ and $B_n(n:0)$, which can be written as:

$$A_n = \boxed{a_n \mid A_{n-1}(n-1:0)}$$

$$B_n = \boxed{b_n \mid B_{n-1}(n-1:0)}$$

Accordingly, it is possible to define the "smaller-than" operator in figure 3 recursively:

$$A_0 < B_0 \equiv \overline{a_0} \cdot b_0$$

$$A_n < B_n \equiv \overline{a_n} \cdot b_n + \left(a_n \cdot b_n + \overline{a_n} \cdot \overline{b_n}\right) \cdot \left(A_{n-1} < B_{n-1}\right)$$

Based on these recursive expressions – and based on similar expressions for the equality operator– a Boolean equation can be derived for the output Out. Writing down the resulting Boolean equation in terms of, for instance, the symbolic Lgc-formalism of the COSMOS environment [7, 8], gives us the symbolic model for the component.

3.4 Using the symbolic models in the SFG-Tracing methodology

The symbolic models describing the functional behaviour of analog components can be easily incorporated in existing formal verification methodologies for digital designs. As a result, verification techniques for digital systems can be used to verify integrated mixed-mode designs.

In our experiments, the Signal-Flow-Graph (SFG) Tracing, a methodology initially developed at Imec for the formal (symbolic) verification of digital VLSI design, was employed [12]. The practical value of the SFG-Tracing has already been demonstrated by formally proving the correctness of VLSI designs of up to 230.000 transistors [13]. Since a more elaborate discussion on the relationship between the symbolic models and the SFG-Tracing is outside the scope of this paper, the interested reader is referred to [2].

3.5 Analog macro building blocks

Without demeaning the general applicability of the verification approach, a far-reaching automation can be realized by only allowing symbolic models for so-called analog macro building blocks, i.e. predefined elements of an analog component library such as analog comparators, digital-to-analog converters, analog-to-digital converters, operational amplifiers and others. Restricting ourselves in this way can be justified since the majority of integrated mixed-mode systems is usually designed using only such pre-defined library components. A further justification is to be found in the fact that in this way the verification process can be brought closer to the practical experience of the designer himself.

It is important to observe that restricting ourselves to library-based designs is a self-imposed limitation: the approach may still be applied for integrated mixed analog/digital designs in which custom-designed analog subsystems are used, provided that the functional behaviour of the analog parts is accurately known and can be described in terms of the symbolic formalism available. The automation advantage, however, will be lost for such systems.

4. An application example: the Smart*P*en design

As mentioned before, the main incentive for the development of a new verification approach was the need to formally verify the functional correctness of integrated mixed-mode systems such as the smartpen. In this section, a concrete example – an integrated mixed analog/digital subsystem of the Smart*P*en prototype – of what has been verified using our symbolic approach is presented.

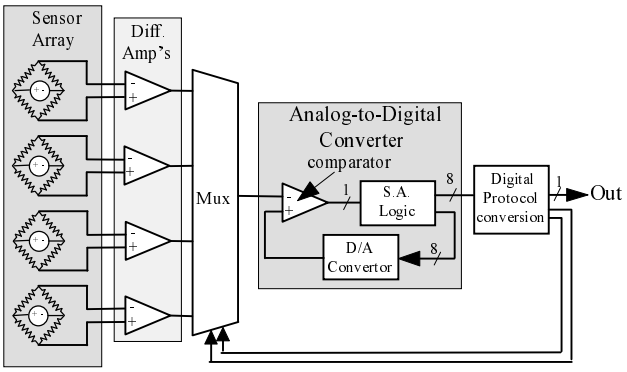


Figure 4: An integrated mixed-mode subsystem of the Smart*P*en^{TM} application

A simplified version of the mixed-mode system that was formally verified is illustrated in figure 4; a sensor array – a set of strain gauges in wheatstone bridge configurations – generates the analog inputs of the system. Alternately, the digital control unit selects one of four amplified (analog) difference voltages, which is subsequently converted into an 8-bit digital signal. The 8-bit digital signal is then used in a specific bit-sequential protocol. Finally, the bit-sequential information will be transmitted via a wireless communication link between the Smart*P*en and a personal computer, where the information can be stored or used by special purpose signature verification algorithms [3].

The actual verification process starts from a structural, hierarchical description of the system under consideration. In other words, the mixed-mode design is represented as a specific interconnection of analog components – e.g. comparators, multiplexers, A*D*C's and others – and several digital (hierarchical) blocks. The digital subsystems themselves are defined in terms of their gate-level schematics.

After the digital design hierarchy has been flattened, a symbolic model for the design can be constructed in a very straightforward way: each component (or gate) must simply be replaced by its corresponding model in a pre-defined symbolic library. Our symbolic library includes behavioural models for the most common gate-components found in digital design libraries (e.g. the MIE*T*EC 2.4µ CMOS technology library), as well as models for the most general analog macro building blocks. An outspoken advantage of these library-based symbolic models can be found in their simplicity; the functional correctness of the individual models can be exhaustively verified, which ultimately benefits the overall reliability of the verification approach itself.

Retaining the original interconnections, a symbolic model for the overall design can be generated starting from the individual models. The complete model can then be presented to symbolic evaluation or simulation tools such as COSMOS [7,8]. These tools enable us to formally check the correctness of the design with respect to a functional specification. Whereas the modelling of a (library-based) design can be fully automated, acquiring a formal specification of that design is, at present, still a manual task. Without doubt, further research into the matter is needed if we want to alleviate the verifier of this tedious task.

5. Errors uncovered during the symbolic verification

Applying the symbolic verification technique to the mixed-mode system described in the previous section, enabled us to identify or expose two undesired faults in the prototype design. A first design fault was detected in

the parity generation of the transmission protocol. An "odd-or-even"-pin allows us to generate an odd or even parity-bit for the transmitted information. Symbolic verification demonstrated that the actual parity-bit was in fact independent of the "odd-or-even" pin-value, contradicting the given specifications.

In addition to the 'parity' error, the symbolic verification of the smartpen also uncovered an unspecified initialization effect – the initialization of the pen appeared to be 1 clock cycle shorter than was formally specified. Closer examination however showed us that the initialization effect was, in fact, not really a design error. As it turned out, the formal specification used for the verification experiment proved to be more strict than was actually required for the correct working of the application. On the whole, it turned out that the symbolic modelling and verification approach explained above contributed to improving the overall design of the Smart*P*en prototype.

6. Conclusions and future work

In this paper, symbolic bit-vectors are used to represent analog voltages and the functional behaviour of mixed analog/digital building blocks is expressed in terms of models based on the symbolic bit-vector representations. We have shown that these symbolic models can be used to verify the (abstract) functional behaviour of an integrated mixed-mode system. *T*he symbolic modelling and verification approach has been successfully applied to verify the functional correctness of a mixed-mode subsystem of the Smart*P*en™, a realistic application. Some unspecified and undesired design aspects have been uncovered, resulting in an overall enhancement of the smartpen design.

Evidently, the research discussed in this manuscript is still somewhat on the preliminary side and it should be clear that more work is required to further examine the full potential of the approach. In the near future, the objective is therefore to extend the existing library of symbolic models, and to cover all viable pre-defined analog blocks. Using the extended symbolic library, the practical feasibility and limitations of our verification approach will be investigated by verifying other integrated mixed-mode applications developed in the Integrated MicroSystems *D*esign group of Imec

7. Acknowledgments

*T*he authors would like to thank Randal E. Bryant and the members of the COSMOS team for making the COSMOS system available to them. *T*he research presented in this paper was supported by a scholarship from the Flemish Institute for the promotion of Scientific-*T*echnological Research in Industry.

8. References

[1] L. Claesen, *D*. Beullens, R. Martens, et al., "Smart*P*en – An Application of Integrated Microsystems and Embedded Hardware/Software Co*D*esign", In: IEEE Users Forum of E*D*&*T*C, *P*aris 1996, IEEE Computer Society, pp. 201-205

[2] Stefan Hendricx, Luc Claesen, "A Symbolic Modelling Approach for the Formal Verification of Integrated Mixed-Mode Systems", In: *P*roc. 3rd *D*esigning Correct Circuits Workshop, 2-4 September 1996, Båstad (Sweden)

[3] R. Martens, L. Claesen, "On-Line Signature Verification by *D*ynamic *T*ime-Warping", In: *P*roc. 13th International Conference on *P*attern Recognition, Vol. III, pp. 38-42, IEEE Computer Society, August 25-29, 1996, Vienna

[4] F. Leclerc, R. *P*lamondon, "Automatic Signature Verification: *T*he State Of *T*he Art – 1989-1993", In: Int. Journal on *P*attern Recognition. Artificial Intelligence, vol. 8, pp.643-660, 1994

[5] *T*. Coe, *T*. Mathisen, C. Moler, V. *P*ratt. "Computational Aspects of the *P*entium Affair", In: IEEE Computational Science & Engineering, spring 1995, pp. 18-31

[6] *D*. Conner. "Mixed Analog-*D*igital Simulation", In: E*D*N 1994 (September), pp. 39-44

[7] Randal E. Bryant, *D*erek Beatty, et al. "COSMOS – a compiled simulator for MOS circuits." In: 24th *D*esign Automation Conference. 1987

[8] *D*. Beatty, K. Brace, R. E. Bryant, K. Cho, L. Huang. "User's Guide to COSMOS: Compiled Simulator for MOS *D*igital Circuits"

[9] *P*. Camurati and *P*. *P*rinetto. "Formal Verification of Hardware Correctness: Introduction and Survey of Current Research", In: IEEE Computer 1988 (July), 8-19

[10] S. Hazelhurst, C-J. H. Seger. "An Integrated Approach to Verifying Large Circuits : A Case Study", In: *P*roc. 3rd *D*esigning Correct Circuits Workshop, 2-4 September 1996, Båstad (Sweden)

[11] *D*avid *P*. Appenzeller, Andreas Keuhlmann. "Formal Verification of a *P*ower*PC*™ Microprocessor", In: *P*roc. IEEE International Conference on Computer *D*esign: VLSI in Computers and *P*rocessors, Austin (*T*exas) 1995. IEEE Computer Society *P*ress, 1995

[12] L. Claesen, F. *P*roesmans, E. Verlind, H. *D*e Man. "SFG-*T*racing: a Methodology for the Automatic Verification of MOS *T*ransistor Level Implementations from High Level Behavioural Specifications", In: *P*roceedings ACM-SIG*D*A International Workshop on Formal Methods in VLSI *D*esign (ed. *P*. A. Subrahmanyam), 1991

[13] M. Genoe, L. Claesen, E. *P*roesmans, E. Verlind, H. *D*e Man. "Illustration of the SFG-*T*racing Multi-Level Behavioural Verification Methodology, by the Correctness *P*roof of a High to Low Level Synthesis Application in CA*T*HE*D*RAL-II", In: *P*roc. ICC*D*-91