Analog Simulation Meets Digital Verification - A Formal Assertion Approach for Mixed-Signal Verification

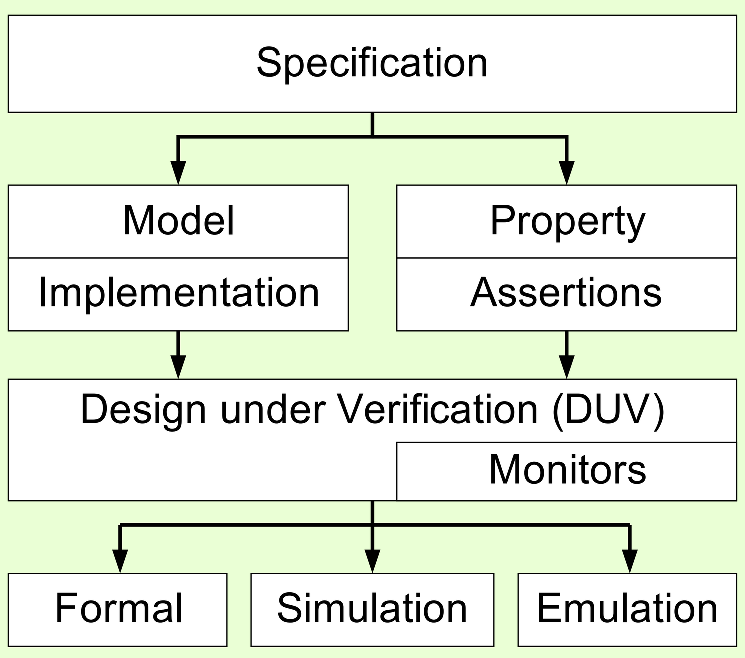
Alexander Jesser, Lars Hedrich
Department of Computer Science
University of Frankfurt, D-60325 Frankfurt a. M., Germany
{jesser,hedrich}@em.cs.uni-frankfurt.de

Stefan Lämmermann, Roland Weiss, Jürgen Ruf, Thomas Kropf, Wolfgang Rosenstiel
Department of Computer Engineering
University of Tübingen, D-72076 Tübingen, Germany
{laemmerm,weissr,ruf,kropf,rosenstiel}@informatik.uni-tuebingen.de

Alexander Pacholik, Wolfgang Fengler
Computer Architecture Group
Ilmenau Technical University, D-98693 Ilmenau, Germany
{alexander.pacholik,wolfgang.fengler}@tu-ilmenau.de

Abstract— Functional and formal verification are important methodologies for complex mixed-signal designs. But there exists a verification gap between the analog and digital blocks of a mixed-signal system. Our approach improves the verification process by creating mixed-signal assertions which are described by a combination of digital assertions and analog properties. The proposed method is a new assertion-based verification flow for designing mixed-signal circuits. The effectiveness of the approach is demonstrated on a Σ/Δ-converter.

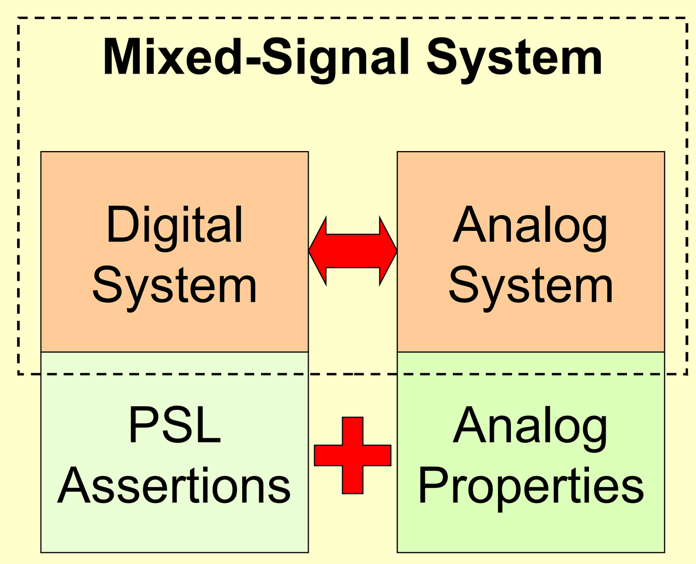
Digital Assertions



- Functional simulation with formal properties
- Assertions for simulation and formal verification

Problem

- Discrete time
- Signal triggered
- Boolean values

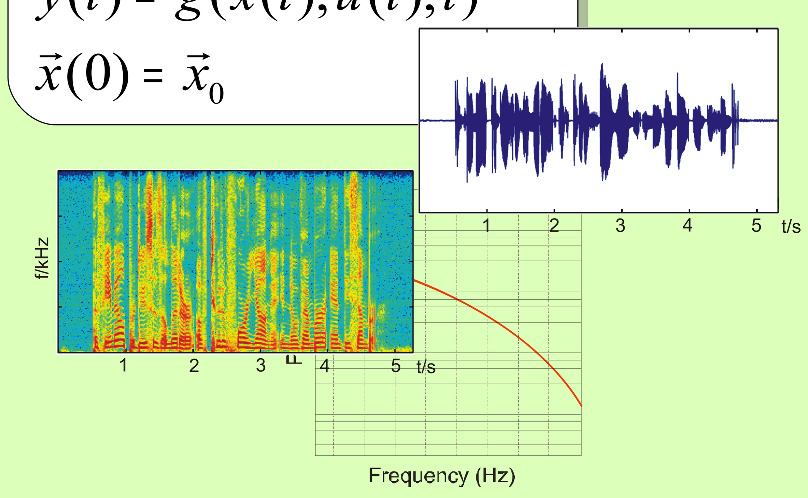


- Continuous time
- Time intervals
- Continuous values
- Ranges with inequalities

Formal property description languages for mixed-signal systems are still not sophisticated

Analog Properties

$$0 = f(\dot{\vec{x}}(t), \vec{x}(t), \vec{u}(t), t)$$
$$\vec{y}(t) = g(\vec{x}(t), \vec{u}(t), t)$$
$$\vec{x}(0) = \vec{x}_0$$



- Continuous signal monitoring
- Continuous time consideration
- Frequency analysis

Digital Condition

Analog Condition

Solution: combination of conditions

Characteristics (general)

- Including monitor points (digital/analog block)
- Specification language: PSL (property specification language)
- Using implication → operator for combining analog and digital conditions
- Interconnection verification

General definition of Mixed-Signal Assertions (MSA)

Precondition	Postcondition
$\{Analog\}$ $\mapsto$	$\{Digital\}$
$\{Digital\}$ $\mapsto$	$\{Analog\}$

Characteristics (analog)

- PSL extension for mixed-signal systems
- Discretize continuous time into, e.g., equidistant time points
- Quantify signal values into a Boolean representation
- Map time intervals to discrete time points

Simulation / Formal Verification

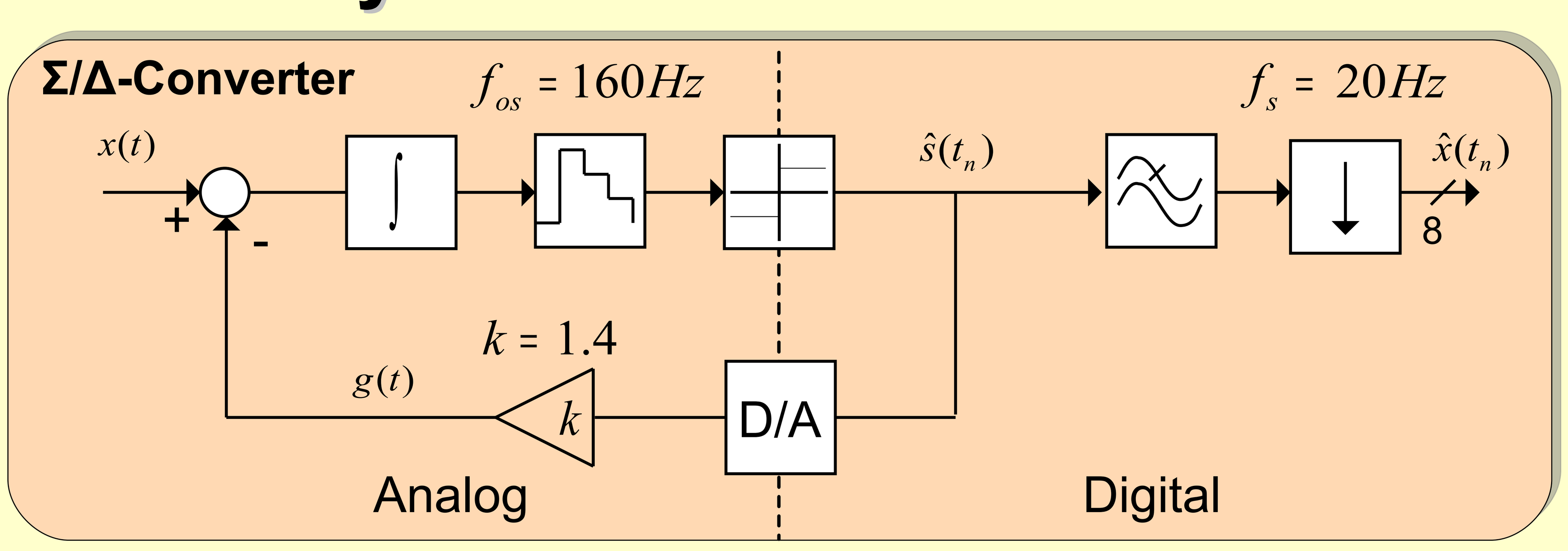
Structure of MSA

Analog Precondition — Digital Postcondition

$$\{x(t) < 0.7 \,\&\, x(t) > -0.7\} \mapsto \{next\_e[1:4] \quad (\hat{x}_0 \,\&\, \hat{x}_1 \,\&\, !\hat{x}_2)\}$$

Analog signal range — Temporal operator with interval — Digital signal

Case study



Mixed-Signal Assertions:

$1: \{x(t) < 0.7 \,\&\, x(t) > -0.7\} \mapsto \{next\_e[1:2]!((\hat{x}_0(t_n) \,\&\, \hat{x}_1(t_n) \,\&\, \hat{x}_2(t_n) \,\&\, \hat{x}_3(t_n) \,\&\, \hat{x}_4(t_n) \,\&\, \hat{x}_5(t_n) \,\&\, \hat{x}_6(t_n) \,\&\, \hat{x}_7(t_n)) \,||$
$(!\hat{x}_0(t_n) \,\&\, !\hat{x}_1(t_n) \,\&\, !\hat{x}_2(t_n) \,\&\, !\hat{x}_3(t_n) \,\&\, !\hat{x}_4(t_n) \,\&\, !\hat{x}_5(t_n) \,\&\, !\hat{x}_6(t_n) \,\&\, \hat{x}_7(t_n)))\}$

$2: \{next\_a[0.0:0.05](x(t) \geq 0.8)\} \mapsto \{next\_e[2](\hat{x}(t_n) = \hat{x}_0(t_n) \,\&\, \hat{x}_1(t_n) \,\&\, \hat{x}_2(t_n) \,\&\, \hat{x}_3(t_n) \,\&\, \hat{x}_4(t_n) \,\&\, \hat{x}_5(t_n) \,\&\, \hat{x}_6(t_n) \,\&\, !\hat{x}_7(t_n))\}$

$3: \{next\_a[0.0:0.05](x(t) \leq 0.8)\} \mapsto \{next\_e[2](\hat{x}(t_n) = !\hat{x}_0(t_n) \,\&\, !\hat{x}_1(t_n) \,\&\, !\hat{x}_2(t_n) \,\&\, !\hat{x}_3(t_n) \,\&\, !\hat{x}_4(t_n) \,\&\, !\hat{x}_5(t_n) \,\&\, !\hat{x}_6(t_n) \,\&\, \hat{x}_7(t_n))\}$

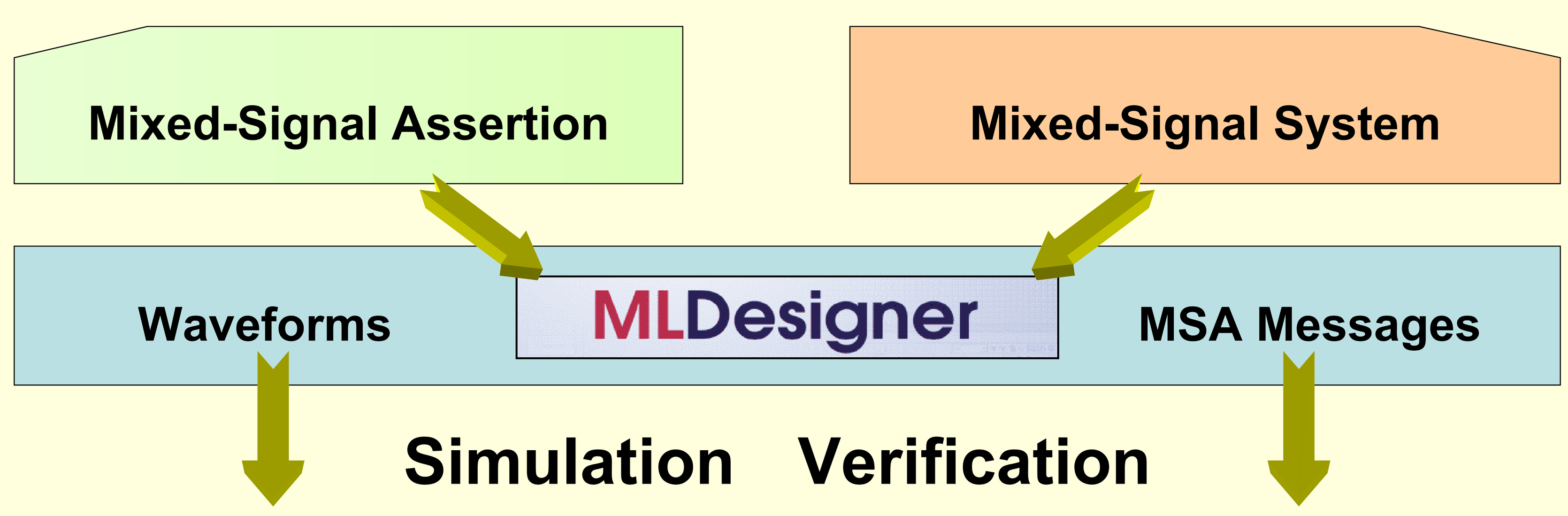
$4: \{x(t) = 0.891\} \mapsto \{next\_e[10](next\_a[4](\hat{s}(t_n)) \,\&\, next\_e[4:5](!\hat{s}(t_n))\}$

$5: \{x(t) = -0.891\} \mapsto \{next\_e[10](next\_a[4](!\hat{s}(t_n)) \,\&\, next\_e[4:5](\hat{s}(t_n))\}$

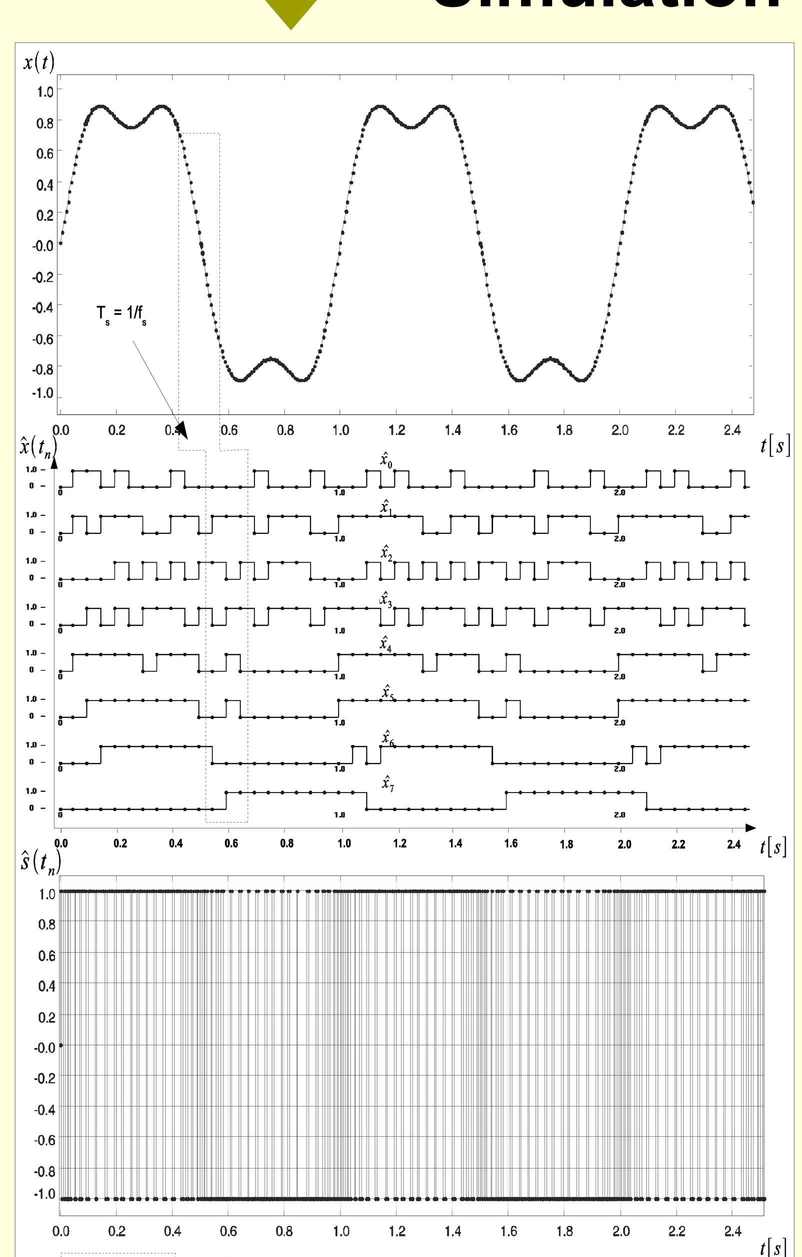
$6: \{x(t) = 0\} \mapsto \{next\_e[1:2](\hat{s}(t_n)) \,\&\, next\_e[1:2](!\hat{s}(t_n))\}$

$7: \{\hat{s}(t_n)\} \mapsto \{g(t) = 1.4\}$

Results



Simulation



Verification

MSA	Property Accept/Reject in t_{clk}				
	1	2	3	4	5
1 (f_s)	A,1	A,2	A,10	A,11	A,12
2 (f_s)	A,4	A,8	A,24	A,28	A,44
3 (f_s)	A,14	A,18	A,34	A,38	A,54
4 (f_{os})	A,26	A,31	A,69	A,186	A,191
5 (f_{os})	A,107	A,267	A,306	A,427	A,466
6 (f_{os})	A,81	A,161	A,241	A,321	A,401
7 (f_{os})	R,0	R,2	R,4	R,6	R,7

Counterexample

- MSA is based on the industry standard PSL
- MSA can be used for functional and formal verification
- Better monitoring/controlling of analog/digital interactions
- Check formal properties during simulation

14th Workshop on Synthesis And System Integration of Mixed Information technologies SASIMI 2007, Sapporo October 15-16, 2007
This work has been funded by BMBF and edacentrum e.V. within project FEST (No. 01M3072C)