Automatic Analog Circuit Synthesis using WiCkeD

Xiaoying Wang
J.-W. Goethe University of Frankfurt, Germany

MunEDA User Group Meeting, Munich, Sept. 2007

Outline

• Introduction
• Automatic Analog Circuit Synthesis
  – Topology generation
  – Topology selection
  – Circuit sizing using WiCkeD
• Summary
Introduce

Institute of Computer Science, University of Frankfurt, Germany
Electronic Design Methodology Group with Prof. Hedrich (founded since 2004)

EDA for analog circuits

<table>
<thead>
<tr>
<th>Project</th>
<th>Theme</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAMS</td>
<td>Struktur Synthesis of Analog Circuits</td>
<td>Finished</td>
</tr>
<tr>
<td>FEST</td>
<td>Model-checking of Nonlinear Circuits</td>
<td>Finished</td>
</tr>
<tr>
<td>VeronA</td>
<td>Formal Verification of Analog Circuits</td>
<td>Started</td>
</tr>
<tr>
<td>Honey</td>
<td>Yield and Reliability Oriented Design</td>
<td>Starting late 2007</td>
</tr>
</tbody>
</table>

Design flow

- Specification
  - gain, f_t, slew rate, etc.
- Topology generation
  - Hierarchical synthesis
- Topology selection
  - Symbolic analysis
- Sizing
  - WiCkeD sizing tool

No. of circuits target
Topology synthesis: state of the art

- Topology library
  - e.g. OASYS(89')
- Topology generation
  - with single transistor as building block
    [Klumperink] (01')
  - with genetic algorithms
    DARWIN(95'), [Dastidar] (05')

Hierarchical topology synthesis

- A set of well defined blocks with specialized signal information of terminals
- Synthesis rules for combination between blocks
- Block-chains/nets can represent the topology of circuits
- Generation instead of library: new circuits
Design example

- Block net:

- Schematic

Topology Selection – Symbolic Analysis

- Goal of topology selection
  - Reduce high number of synthesized circuits
  - Short run time

- Symbolic analysis
  - Fast performance estimation
  - Performance ⇔ parameter dependencies
  - Simple design equations ⇒ Initial sizing
Symbolic Analysis – Work Flow

gain, \( f_{ct} \), \( R_{out} \)

specification

topology generation

linear symbolic analysis

set-up

simplify

solve

performance evaluation

comparison to spec.

Performance =

\[ f(W_1, \ldots, W_n; I_{1}, \ldots, I_{n}; I_{\text{bias1}}, \ldots, I_{\text{biasn}}) \]

with \( \forall i \ W_i = W_{\text{nom}} \),

\( L_i = L_{\text{nom}} \),

\( I_{\text{biasi}} = I_{\text{biasnom}} \)

Circuit sizing: WiCkeD

- Sizing tool: WiCkeD
- Interface: GUI or CLI?
  - GUI: graphical user interface
    - Interactive
    - User-friendly
    - Manual command input
    \[ \rightarrow \text{few circuits} \]
  - CLI: command line interface
    - Non-interactive
    - Embedded into TCL
    - Efficiency and productivity
    \[ \rightarrow \text{more circuits} \]
CLI instead of GUI -- how?

- Sizing flow in GUI
  1. Circuit preparation
  2. Parameter setup
  3. Optimization / sizing

- CLI with command: `wicked -s xxx.tcl xxx.def`

Files about:
- circuit structure;
- simulation info;
- test bench;

Files about:
- constrains;
- specifications;

Files about:
- sizing process;

Circuit sizing: design flow

- topology generation
- symbolic analysis

specification
- gain, $f_c$,
- CMRR,
- power,
- etc..

Circuit structures
- Testbench

Feasibility Optimization
- Nominal Optimization

Sized circuits
- $m < n$
Synthesis example – comparator (I)

- Synthesize comparator:
  - op + output stage

Specifications:
- High Slew-Rate > 5V/μs
- Load capacity = 1nF

- Result:
  - 4424 => 446 => 40 circuits

Synthesis example – comparator (II)

- Block chain:
  - Diff.-pair
  - Current mirror
  - Output stage

- Auto. gen. schematic:

- Performance:
  - Area: 70 μm²
  - Gain: 67 dB
  - OutpVoltRange: 2.0 V
  - Ft: 9.9 MHz
  - Power: 7.4 mW
  - SlewF: 5.13 V/μs
  - SlewR: 5.36 V/μs
  - PSRR: 54.5 dB
  - CMRR: 41.8 dB
Summary

• Generation and selection of analog circuits step by step
  – Generation with hierarchical blocks
  – Fast symbolic analysis
  – Automatic sizing using WiCkeD CLI for a large number of circuits

• Outlook
  – More circuit classes

Thanks!