**GTX: The MARCO GSRC Technology Extrapolation System**

A. Caldwell, Y. Cao, A. B. Kahng, F. Koushanfar,  
H. Lu, I. Markov, M. Oliver,  
D. Stroobandt and D. Sylvester

http://vlsicad.cs.ucla.edu/GSRC/GTX/

Supported by Cadence, Synopsys and the MARCO Gigascale Silicon Research Center

---

**Outline**

- Introduction
- Previous efforts
- Goals for an “ideal” system
- GTX structure
- Fundamental features of GTX
- Example studies
  - Sensitivity analyses of cycle-time models
  - Evaluating new device models
  - Delay uncertainty study
Introduction: Technology Extrapolation

- Evaluates impact of
  - design technology
  - process technology
- Evaluates impact on
  - achievable design
  - associated design problems
- Questions to be addressed
  - Sets new requirements for CAD tools and methodologies
  - Roadmaps: familiar and influential example

What is the most power-efficient noise management strategy?

How and when do L, SOI, SER, etc. matter?

Will layout tools need to perform process simulation to efficiently model cross-die and cross-wafer manufacturing variation?

Sample Study 1: Optimization

- Most commonly cited optimal buffer sizing expression (Bakoglu)
- New study:
  - Sweep repeater size for single stage in the chain
  - Examine both delay and energy-delay product

[Graph showing Critical Path Delay vs. Repeater Size, with normalized energy-delay product as well.]
Sample Study 2: New Models

- Five different interconnect models
  - Bakoglu’s model (RC)
  - [Alpert, Devgan and Kashyap, ISPD 2000] (RC)
  - [Ismail, Friedman and Neves, TCAD 19(1), 2000] (RLC)
  - [Kahng and Muddu, TCAD 1997] (RLC)
  - Extension of [Alpert, Devgan and Kashyap, ISPD 2000] (RLC)

![Graph showing Wire Delay vs Wire Length and Wire Width](image)

What Do We Need?

- Reuse of existing models, effort
- Framework for adding new models to encompass new aspects of technology, new axes of achievable design
- Ability to evaluate models (sanity, consistency checks)
- Easy model substitution to compare between models
- Sweeping ability to assess the impact of modeling choices
- Constraints to allow elimination of infeasible solutions
## What is Available?

- Previous and ongoing efforts
  - ITRS Roadmaps
  - Tools: SUSPENS, GENESYS, RIPE, BACPAC, ...
  - Numerous tools in industry

- Observations
  - Predict “same” parameters but different assumptions, inputs
  - Lack of documentation and visibility of internal calculations
  - Single inference chain for a given output (hard-coded)
  - Inflexible: user cannot define studies of related parameters
  - Near-total duplication of effort
  - Missing: models of CAD tools and optimizations (what is really “achievable”?)
  - Missing: scope, comprehensive coverage

## Goals of A New Technology Extrapolation System

- Flexibility
  - Edit or define new parameters and relations between them
  - Perform specific studies (but different studies at different times)

- Quality
  - Continuous improvements
  - World-wide participation of experts

- Transparency
  - Open-source mechanism
  - Models are visible to the user

- Prevention of redundant effort
  - Permanent repository of first choice
  - Adoptability and maintainability
GTX: GSRC Technology Extrapolation System

- GTX is set up as a framework for technology extrapolation
- Openness in grammar, parameters and rules
  - Easy sharing of data in research environment
  - Contributions from other groups

Knowledge Representation

- Human-readable ASCII grammar

```plaintext
#parameter dl_chip
#type double
#units [m]
#default 1e-2
#description chip side length
#reference
#endparameter

#rule BACPAC_dl_chip
#description
#output
double [m] dl_chip;
#inputs
double [m^2] dA_chip;
#body sqrt(dA_chip)
#reference
#endrule
```
## Knowledge Representation

- **Human-readable ASCII grammar**
- **Benefits:**
  - Easy creation/sharing of parameters/rules by multiple users
    - D. Sylvester and Y. Cao: device and power, SOI modules that “drop in” to GTX
    - P.K. Nag: Yield modeling
  - Extensible to models of arbitrary complexity (specialized prediction methods, technology data sets, optimization engines)
    - Avant! Apollo or Cadence SE P&R tool: just another wirelength estimator
  - Applies to any domain of work in semiconductors, VLSI CAD
    - Transistor sizing, single wire optimizations, system-level wiring predictions,…

## Parameters

- **Description of technology, circuit and design attributes**
- **Importance of consistent naming cannot be overstated**
  - **Naming conventions for parameters**
    
    \[
    \text{[preposition]} \_ \text{<principal>} \_ {\{<qualifier> \_ <place> \_ <qualifier> \_ <adverbial> \_ <index> \_ <unit>\}}
    \]

  - **Example:** `r_int_tot_lyr_pu_dl`
  - **Requirements:**
    - Relatively easy to understand parameter from its name
    - Distinguishable (no two parameters should have the same name)
      - `r_int` (interconnect resistance) = `r_int` (interconnect resistivity) ?
    - Unique (no two names for the same parameter)
      - `R_int` = `R_wire` ?
    - Sortable (important literals come first)
  - **Software to automatically check parameter naming**
Rules

- Methods to derive unknown from known parameters
- ASCII rules
  - Laws of physics, models of electrical behavior, statistical models
  - Include closed-form expressions, vector operations, tables
  - Storing of calibration data (e.g., “technology files”) for known process and design points in lookup tables
  - Constraints, used to limit range during “sweeping”
- “External executable” rules
  - Assume a callable executable (e.g., PERL script)
  - Use command-line interface and transfer through files
  - Allow complex semantics of a rule
- “Code” rules
  - Implemented in C++ and linked into the inference engine

Rule Chains

- “Rule chains” guide inference
  - Acyclic set of rules
  - Interactive specification and comparison of alternative modeling choices
- Studies
  - Input values + rules that make a rule chain
  - User-controlled and savable
  - “Sweeping” of a rule chain
    - Evaluation of all combinations of multi-valued inputs
    - Example: clock frequency for different Rent exponents and varying logic depth
GTX Engine

- Contains no domain-specific knowledge
- Evaluates rules in topological order
- Performs studies
- Multiple values through “sweeping”
- Runs on three platforms (Solaris, Windows and Linux)

URL: http://vlsicad.cs.ucla.edu/GSRC/GTX/

Graphical User Interface (GUI)

- Provides user interaction
- Visualization (plotting, printing, saving to file)

4 views:
  - Parameters
  - Rules
  - Rule chain
  - Values in chain
GTX Current Status

- Models implemented
  - Cycle-time models of SUSPENS (with extension by Takahashi), BACPAC (Sylvester, Berkeley), Fisher (ITRS)
  - Currently adding
    - GENESYS (with help from Georgia Inst. Tech.)
    - RIPE (with help from Rensselaer Univ.)
  - New device and power modules (Synopsys / Berkeley)
  - New SOI device model (Synopsys / Berkeley)
  - Inductance models (Silicon Graphics / Berkeley / Synopsys)
  - Yield model (CMU)

- Studies performed in GTX
  - Model analysis
  - Study of the impact of parameters
  - Design optimization studies

Outline

- Introduction
- Previous efforts
- Goals for an “ideal” system
- GTX structure
- Fundamental features
- Example studies
  - Sensitivity analyses of cycle-time models
  - Evaluating new device models
  - Delay uncertainty study
Sensitivity Analysis of Cycle-time Models: Parameter Sensitivity

- Change parameter values and observe resulting difference in outputs

Sensitivity Analysis of Cycle-time Models: Model Sensitivity

- Replace rule in a model's rule chain by another model's rule and observe the difference in outputs
New device models for bulk Si and Silicon-on-Insulator (SOI) devices
- Provided by D. Sylvester (Synopsys) and Y. Cao (UCB)
- SOI model assumes partially-depleted SOI (PD-SOI) technology and is based on popular BSIM3SOI models
- Both modules compared to BSIM3 HSPICE runs; results match within 10%

General study
- Floating body effect: changes in $V_{th}$ and $I_{dsat}$
  - Calculate range of possible $I_{dsat}$ values
  - Model ignores the impact of capacitive coupling on body voltage
- Dynamic delay (due to coupling capacitances between same-layer interconnects)

Influence of device technology on clock frequency and power
- Best case: largest $I_{dsat}$ (realizable due to floating body effect, only for SOI) and no effective coupling capacitance: $f$ from 1.03 GHz (bulk) to 1.31 GHz (SOI)
- Worst case: smallest $I_{dsat}$ and switching factor of 2: 867 MHz and 1.05 GHz

Power results

<table>
<thead>
<tr>
<th></th>
<th>Bulk Si</th>
<th>SOI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logic + local wires</td>
<td>26.20 W</td>
<td>28.99 W</td>
</tr>
<tr>
<td>%</td>
<td>46.18%</td>
<td>43.91%</td>
</tr>
<tr>
<td>Global interconnects</td>
<td>2.20 W</td>
<td>2.60 W</td>
</tr>
<tr>
<td>%</td>
<td>3.88%</td>
<td>3.93%</td>
</tr>
<tr>
<td>I/O drivers + pads</td>
<td>11.7 W</td>
<td>13.35 W</td>
</tr>
<tr>
<td>%</td>
<td>20.65%</td>
<td>20.22%</td>
</tr>
<tr>
<td>Clock distribution</td>
<td>7.93 W</td>
<td>9.65 W</td>
</tr>
<tr>
<td>%</td>
<td>13.98%</td>
<td>14.62%</td>
</tr>
<tr>
<td>Memory</td>
<td>0.94 W</td>
<td>0.86 W</td>
</tr>
<tr>
<td>%</td>
<td>1.66%</td>
<td>1.31%</td>
</tr>
<tr>
<td>Short circuit</td>
<td>7.68 W</td>
<td>10.21 W</td>
</tr>
<tr>
<td>%</td>
<td>13.54%</td>
<td>15.47%</td>
</tr>
<tr>
<td>Leakage</td>
<td>0.07 W</td>
<td>0.38 W</td>
</tr>
<tr>
<td>%</td>
<td>0.12%</td>
<td>0.54%</td>
</tr>
<tr>
<td>Total power</td>
<td>56.74 W</td>
<td>66.03 W</td>
</tr>
<tr>
<td>%</td>
<td>100.00%</td>
<td>100.00%</td>
</tr>
</tbody>
</table>

- SOI: 16% increase in power versus Bulk but 24% increase in frequency
Bulk Si Versus SOI Device Models (Cont.)

- Parameter sensitivity of both models
  - Several technology related parameters are varied by +/- 10%
  - SOI slightly less sensitive to input parameter changes
  - Process spread (between best-case and worst-case) larger for SOI

Delay Uncertainty Study

- Staggered repeaters
  - First introduced in [Kahng et al, VLSI Design 99] to reduce delay and noise
### Conclusion

- GTX: a new framework for technology extrapolation
- Flexible and extensible
- Enables easy reuse of models
- Provides a common parameter base between all models
- Provides user interaction
- Relies on open-source and contributions by expert users
- “Living Roadmap”
- Technology extrapolation becomes easier
- More principled understanding of requirements for CAD tools

### GTX Project Information

- Design: A. Caldwell, A. B. Kahng, I. Markov, M. Oliver and D. Stroobandt
- Implementation: M. Oliver
- Knowledge gathering and study implementation: A. B. Kahng, F. Koushanfar, H. Lu and D. Stroobandt
- Model extensions and new studies: Y. Cao, X. Huang, S. Muddu, P.K. Nag and D. Sylvester
- To contact the developers, ask questions, send comments, or to contribute models to GTX, please send E-mail to GTX@cs.ucla.edu