Welcome to our webcast

MQA: Automating Library Validation

Agilent EEsof EDA

Shuang Cai (CS)
Shuang_cai@agilent.com
### Agilent: The Leader In Device Modeling

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Market Share</td>
<td>No. 1</td>
</tr>
<tr>
<td>History</td>
<td>Longest</td>
</tr>
<tr>
<td>Complete</td>
<td>All device type (CMOS, III-V, BJT/HBT, Diode, L/R/C..)</td>
</tr>
<tr>
<td>Accurate</td>
<td>Measurement, extraction, qualification</td>
</tr>
<tr>
<td>Technology</td>
<td>Most advanced (20nm, 65GHz)</td>
</tr>
<tr>
<td>Team</td>
<td>Biggest (US, Japan, China, Europe)</td>
</tr>
<tr>
<td>Solution</td>
<td>Software, service, hardware</td>
</tr>
<tr>
<td>Brand</td>
<td>Most reliable measurement company</td>
</tr>
</tbody>
</table>
Device Modeling Overview

Premier Silicon and III-V Device Characterization and Modeling Solutions

Complete Flow

Measurement
- IC-CAP WaferPro
- Parametric Testers
- Automated Temperature Measurements
- Data analysis and selection
- $1/f$ Noise

Model Extraction
- MBP/ICCAP
- Ideal flow for Model Tuning
- Auto Library Read
- Target Modeling
- Statistical & Mismatch
- Reliability / Device Aging
- All model types

Model Qualification
- MQA
- Quality Checks
- Model Version Comparison
- Foundry Comparison
- Test vs. Simulated
- EDA environment compliance
- Automatic report & Documentation

Services
- Device modeling
- Measurement
- Knowledge Transfer
Device Modeling Overview

Database Integration

- WaferPro
  - Measurement
- MBP/ICCAP
  - Model Extraction
- MQA
  - Model Qualification

Integrated Database
Problems That You May Face When Dealing With SPICE Model Library

• Before you release PDK to your customer:
  – worry about the bugs in SPICE library that frustrate your customer
  – worry about the tedious model QA process that impact your delivery time
  – worry about young engineers making basic, but hard to catch mistakes during model extraction and model library assembly
  – feel bored with QA report generation and its formatting

• Or after you receive foundry’s PDK:
  – need to compare two foundries when second-sourcing
  – need to quickly evaluate the impact of a new PDK release on the current design
  – need to run benchmarking between different simulators or different versions of one simulator
  – need a handy, easy-to-use tool to generate various plots and tables when designers ask for technology information
MQA Model Qualification
The Golden Standard

SPICE Models

Meas. Data

MQA

Rules File

Simulation

100+ Customers 10+ Years in Production

Reports
MQA Model Qualification

Functions

Analyze
- Quality
- Operational Characteristics

Compare
- Models – Versions, Foundries
- Simulators, Meas. Data
- Process Nodes

Document
- Integrated Automated Documentation
MQA Model Qualification

Bring Libraries into Compliance

• Understand specific regions of operation
• Assess quality
• Compare versions of models
• Compare foundries
• Compare simulators
• Bring new libraries online quickly

Simulators
- ADS
- GoldenGate
- Hspice
- Eldo
- AFS
- Spectre
- SmartSpice
- FineSim
- ..

Foundries
- TSMC
- GlobalFoundries
- Samsung
- UMC
- SMIC
- IBM
- TowerJazz
- GSMC
- HLMC
- ..

Models
- BSIM
- PSP
- ..

Technologies
- 45nm
- 32nm
- 28nm
- 20nm
- ..

Library 1

Library 2

Library X-1

Library X
Proactively challenge your IC provider with MQA and MBP

- Challenge your foundry
- Investigate PDK changes
- Qualify your EDA simulators
MQA Value Proposition

• Quality
  – Comprehensive understanding of model operation
  – Avoid costly errors from undetected model imperfections
  – Confidently determine performance criteria

• Productivity
  – Quickly bring libraries into compliance
  – Rules-driven, Knowledge-based = IP reuse
  – Automated documentation functionality integrates into your reporting system
  - Single file drives all simulators

• Cost Savings
  – Make vs. buy decisions on internal fabs
  – Improve foundry supplier prices by qualifying second sources
MQA OVERVIEW

&

Demonstration
Device Modeling Overview

WaferPro  MBP/ICCAP  MQA

Measurement  Model Extraction  Model Qualification

Integrated Database
Model Qualification Challenges and Requirements

When a model library has been obtained from the foundry, it should be checked against:

- **Generic checks**
  - Physical behavior
  - Model behavior in terms of corners, statistical, etc.
- **Specific checks**
  - Design requirements (geometry, bias, temperature, etc.)
  - Special effect
  - All Kinds of Comparison

This means:

- Create a lot of netlist files
- Handling of huge amount of data
- Intelligent flagging of issues
- Detailed reporting & documentation

And, this require a

- Comprehensive, flexible and customizable platform
MQA – the industry standard qualification platform that assists modeling experts and model users to perform comprehensive model qualification with an knowledge-based, rules-driven approach.

- It qualifies device models/libraries of ever growing complexity.
- It compares models/libraries among various simulators, versions, foundries, process nodes, etc.
- It is an automated device model verification and documentation tool that can analyze curve trends of huge amount of simulation or/and measurement data.
What is MQA for?

**IDMs Foundries**

IDMs/Foundries use it to perform comprehensive model QA and to release verified model files/libraries in an automated fashion.

**Design House**

Fabless Design Houses use it to evaluate and accept the model libraries from Foundries.
MQA Model Qualification Architecture

- SPICE Models
- Meas. Data
- MQA
- Rules File
- Simulation
- Database/Reports
Apply MQA as ……

SPICE model qualification
• Knowledge based and rules driven
• Capability to handle huge amount of data

Design interface (foundry interface)
• Comparison
• Sharing the new technology characterization

Design documentation and QA result sharing
• Easy to generate report
• Bridge between foundry and design house
• Communication between different groups
MQA Demos

- Fab A vs. Fab B
- Simulator A vs. Simulator B
- More to compare…
MQA’s Basic Facts

Supported SPICE Simulators
- Commercial simulators: HSPICE, SPECTRE, Eldo, SPICE3, ADS, Golden Gate (2012.07), AFS, FineSim, SmartSpice
- In-house simulators

Support All SPICE models
- MOSFET: EKV, BSIM, SOI, PSP, HiSIM families
- BJTs: GP, VBIC, Mextram, HiCUM
- Passive devices (resistor, capacitor and inductor)
- Angelov, Angelov GaN, Agilent HBT (from MQA2012.07)
- Global model and binning model
- Compact model, macro model and Verilog-A model, TMI model
- Support any circuit block validation

Support different measurement formats
- IV, CV, S-par, Noise and ET
- Different file formats from various tools
- IC-CAP SQL database(from MQA2013.01)
# Built-in extensive Qualification Flow

## Checking & Analysis
- Measurement QA
- Accuracy check
- Single model QA
- Model library QA
- Statistical model
- Mismatch model
- Benchmark circuit

## Comparison
- Model versions
- Foundries
- Simulators
- Process nodes
- Model and data

## Reporting
- Report generation automation
- PDF, Word, Excel, HTML, PPT
Inside the rule file

A Rule file in MQA is a ASCII text file which defines how MQA performs certain checks on certain objects under certain conditions.

Example: Rule “Check Idsat vs. L”

[Label: 4001:title= Check Idsat vs. L] *for NMOS
[Condition: (devtype=1) and (application=1 or application=2 or application=3)]

[Loops
  : X=L(start=g_lmin,stop=g_lmax,perdec=10)
  : P=W (start=g_wmin,stop=g_wmax, num=3)
  : P1=Vgs(vgg)
  : P2=Vds(vdd)
  : P3=Vbs(0)
  : P4=T(tmin, tnom, tmax)]

[Target: y=Idsat]

[Check: 01:Check Trend: CheckTrend2D (p,x,y,"times=1","incAtFirst=-1"): error: Trend is not right]
[Compare: 11: Compare Trend: CompPlot(p,x,y): error: TBD]

A rule (ASCII, customizable) is typically organized like this:
1. **ID and title** for selection in the GUI.
2. **Conditions**, which determine when it is valid, i.e. when it will be executed.
3. **Loops**, which defines the variations like bias, W/L/T sweeping, etc.
4. **Target**, which defines target, normally it will be Y-axis on plot.
5. **Check functions**, to define which MQA checks to be applied, also to define the warning message if check failed.
Example of MQA Check functions

CheckTrend

This function plots a 2-D figure(x, y) and checks the curve trend.

Syntax
1. Format.
CheckTrend(x,y,'times'=1,'incAtFirst'='1')

<table>
<thead>
<tr>
<th>Argument</th>
<th>Possible value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Times</td>
<td>1</td>
<td>Stands for that the curve is monotonic.</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>Stands for that the curve has one peak.</td>
</tr>
<tr>
<td>incAtFirst</td>
<td>1</td>
<td>Curve increases at first, default.</td>
</tr>
<tr>
<td></td>
<td>-1</td>
<td>Curve decreases at first.</td>
</tr>
</tbody>
</table>

2. Return value.
Boolean: True or False

3. Application Scope
[Check function]

Schematic
Correct curve trend is similar to the schematics

Example
[Check:01:Check Trend:CheckTrend(x,y,'times'=1,'incAtFirst'='1')): error: Trend is not right]

Check the trend of Vth vs T curves. "incAtFirst'='1" specifies that Vth should decrease as T increase,
if the curve shows different trend, MQA will report an error: "Trend is not right".

CrossCheck

Check whether there is cross over for different P curves..

Syntax
1. Format.
CrossCheck(p, x, y, "Xstart=1", "Xstop=10", "Ystart=0", "YStop=100")
Xstart, Xstop can limit the check range of X axis, optional.
Ystart, Ystop can limit the check range of Y axis, optional.

2. Return value.
Boolean: True or False

3. Application Scope
[Check function]

Schematic
Correct Cross over

Example
[Check: 01:Check Cross: crosscheck(p,x,y); error: Trend is not right] if crossover occurs, MQA will report an error: "Trend is not right".
<table>
<thead>
<tr>
<th>Check Functions</th>
<th>Plot Functions</th>
<th>Calculation Functions</th>
<th>Complex Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>CheckTrend</td>
<td>Save2DData</td>
<td>DataAt2D</td>
<td>CalComplex</td>
</tr>
<tr>
<td>CheckTrend2D</td>
<td>SaveAnyDData</td>
<td>CalPage</td>
<td>Mod</td>
</tr>
<tr>
<td>CheckKink</td>
<td>SavePage</td>
<td>CalCurve</td>
<td>Mod2</td>
</tr>
<tr>
<td>CheckKink2D</td>
<td>Plot</td>
<td>D1</td>
<td>Real</td>
</tr>
<tr>
<td>checkValue2D</td>
<td>CompPlot</td>
<td>GetFt</td>
<td>Imag</td>
</tr>
<tr>
<td>checkPeak2D</td>
<td>CompareRMS</td>
<td>GetFmax</td>
<td></td>
</tr>
<tr>
<td>checkDelta2D</td>
<td>CompModel</td>
<td>GetPeriod</td>
<td></td>
</tr>
<tr>
<td>CrossCheck</td>
<td>CmpAbsDelta</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CompareTrend</td>
<td>CompPoint</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CompXPlot</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CompAnyD</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CompareYAvsYB</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PrintStdDv</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Benchmark circuit block can be implemented in rule and MQA will use user-defined circuit instead of auto-generated netlist.

A 11-stage RO example.
More MQA examples

- One model library:
  (Please read these demos before others to get familiar with basic operations.)
  - QA models
  - Corner QA
  - NP correlation with Monte Carlo

- Multiple model libraries:
  - Model comparison
  - Corner Correlation
  - Accuracy

- Statistics:
  - Histogram
  - Contour
  - Mismatch(Pelgrorn)

- RF:
  - Smith Chart
  - Polar
Key advantage of rule-driven QA process

- Default rule for different model type available
- Flexible and customizable rule files record customer’s IP
- Easy to reuse, accumulation and sharing

QA Coverage

MQA Usage

MQA default rules
Run
Add custom rule
Re-run
More custom rules

Agilent Technologies
It’s all about efficiency

- Easy QA rule setup
- Automate simulation & reporting in one run.
- Friendly simulator interface APIs.
- >10X productivity!
MQA’s Track Records

- The de-facto industry standard SPICE model signoff and acceptance platform
- Productivity improvement comparing with traditional manual/script solutions
- Customer Base Worldwide including the leading IDMs, foundries and design houses
- Comprehensive checking routines enablement to reassure yourself and your customers
MQA’s Competitive Advantages

- Integrated data flow base on Agilent’s device modeling platform
- One language (rule file) for all supported simulators
- Windows and Linux OS support
- GUI interface
- Batch mode and multiple-level parallelisms enablement
- One solution to integrate all the pieces of small in-house tools

Automated
Flexibility for Customization
NO Data Conversion is Needed
High Efficiency
Easy to Maintain
User Friendly
Thank you for your attention!

Q&A

You have the application...

... we have the solution

Agilent Technologies
For an evaluation, please go to:

eesof-usa_support@agilent.com
Shuang_cai@agilent.com

Link page to MQA and other Agilent Device Modeling tools:
http://www.agilent.com/find/eesof-device-modeling