Modeling Your Systems in ADS
Challenges for Aerospace and Defense Applications

• Custom signal formats required for design & testing
• Bring user’s IP in ADS
• Unique signal processing
• Evaluating and Modeling Hardware
Creating models for System Applications

Write your C code models using templates

Sub Net Models with Standard Model Set - Build With:

Signal Formats which are Variants of Wireless Formats - Modify:
- Design Libraries: UWB OFDM, WLAN, WCDMA, CDMA, DTV, TD-SCDMA, GSM, EDGE, 1XEV
- Design/Application Guides: Radar, UWB, cdma2000

Custom or Proprietary IP - Co-Simulate With:
- MATLAB®, System-C, HDL, Verilog-A, Circuit/RFIC

Custom or Proprietary IP - Use System call to
- Use’s software programs

“Difficult to Model” - Use Connected Solutions To:
- Capture a measured signal with a signal analyzer; read signal into simulation as a source
- Turn simulation signal into an RF/analog/digital test signal; input into HW for analysis
Growing Design Trend: ADS Co-Simulation
Save design time with Agilent Ptolemy

**IP Integration**

- Complex partitioning of RF, baseband and sub-system requirements

---

Custom Models

- C++
  - Connected Solutions

Algorithms

- Matlab
  - Circuit & System Designs

IP Development

- HDL

Test and Measurement

- Connected Solutions

Mixed signal IP & Designs

- Circuit & System Designs

---

SDF

Agilent Ptolemy

TSDF

✓ **Speeds up product development**
Building Block #1: Numeric and Timed Libraries

Possible Models with Standard Model Set - Build With:
- Numeric and Timed libraries: sources, signal processing, math, matrix, control, filters, modem...
- User Sub-network models based on libraries.
- Modified models from existing ADS models

Numeric Library Uses Synchronous DataFlow

Transmitted Chirp Signal = A \cdot \cos (wc*(time\_stop-t) + 0.5 \cdot u*(time\_stop-t)^2)

u = \frac{dw}{dt}
User sub-networks models for CFAR Systems

Possible Models with Standard Model Set—Build With:
- Numeric and Timed libraries: sources, signal processing, math, matrix, control, filters, modem...
- User Sub-network models based on libraries.
- Modified models from existing ADS models

- CFAR processing must be used for Radar system working in clutters plus noise environments

- Basic CFAR systems are considered as below
User sub-networks model for CA-CFAR Systems

- Detection Probability: \( P_d = P(1(S > T)) \)
- False Alarm Probability: \( P_{fa} = P\left(\frac{S}{Z} \geq T\right) \), where \( S \) is detect signal, \( Z \) is averaging value for cells, \( T \) is the threshold, Testing under assumption of clutter/noise only
User sub-networks model for CA-CFAR Systems

- Numeric models are used to create delay lines for cell averaging

- Cell Averaging model also can be used for frequency domain CFAR system for Pulse Doppler Radar
User sub-networks model for CAGO-CFAR Systems

- In homogeneous noise condition, CA-CFAR gives a good performance

- In non homogeneous clutter situations, such as in typical transition areas between noise and beginning clutter areas, the CAGO-CFAR need to be considered
User sub-networks model for CAGO-CFAR Systems

- Numeric models are used to create delay lines for cell averaging

- Cell Averaging model also can be used for frequency domain CFAR system for Pulse Doppler Radar
Building Block #2: Design Libraries and Design Guides

Signal Formats which are Variants of Wireless Formats - Modify:
- Design Libraries: UWB OFDM, WLAN, WCDMA, CDMA, DTV, TD-SCDMA, GSM, EDGE, 1XEV
- Design Guides: Radar, UWB, cdma2000

Design Libraries Primarily Use Synchronous DataFlow, except when Modulated on to an RF Carrier

Design Libraries have RF signal sources, RF measurements, propagation channels, BER/BLER/PER, and baseband functionality needed for coding and decoding
Signal Formats which are Variants of Wireless Formats - Modify:
- Design Libraries: UWB OFDM, WLAN, WCDMA, CDMA, DTV, TD-SCDMA, GSM, EDGE, 1XEV
- Design/Application Guides: Radar, UWB, cdma2000

Design Guides Use a Mix of Synchronous DataFlow and Timed Synchronous DataFlow

Design/Application Guides consist of sets of pre-configured schematics and data displays
Building Block #3: Co-Simulation with Custom/Proprietary IP

Custom or Proprietary IP - Co-Simulate With:
- MATLAB®, C-code, System-C, HDL, Verilog-A

Co-Simulation allows 3rd party IP to be used

Matlab Co-Sim Uses Synchronous DataFlow
Matlab IP Import Wizard

New menu item for tool access
Available only for DSP schematics
Matlab IP Import Wizard

Import MATLAB Function Tool: 1

Function file name (.m) or function call
myAverage.m

Help
Include complete path for files not in the default search path, which is:
current project data directory
C:/Data/hpessol/adstolnys/src/matlabimport/matlab

Use MATLAB notation for any function call, e.g. [V,D] = eig(Z)
Copying files to the model directory maximizes model portability

Defaults to current project data directory

Next >>
Matlab IP Import Wizard

One unique function in imported script

“type” context specific

For input/output:
- Int
- Fix
- Real
- Complex

For parameter:
- Real: RealArray
- Int: IntArray
- Fix: FixArray
- Complex: ComplexArray
- String: StringArray
- Precision
- IntEnum
- FloatEnum
- Filename
User can choose to disable optimization and statistical distribution for specific parameters.
Matlab IP Import Wizard
Matlab IP Import Wizard
Matlab IP Import Wizard

Import MATLAB Function Tool: 1

Select Parameter

Edit Parameter
Value Type  Default Value
Filename

Optional
Parameter Description

Parameter Type
Unitless

File Extensions (m, mat)
m, txt, m

- Display parameter on schematic
- Not edited
- Optimizable
- Allow statistical distribution

Script header may include parameter descriptions:
File Extensions will filter file types when browsing.
Matlab IP Import Wizard

Import MATLAB Function Tool: 1

<table>
<thead>
<tr>
<th>I/O</th>
<th>type</th>
<th># rows</th>
<th># columns</th>
<th>transpose</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>b</td>
<td>output</td>
<td>1</td>
<td>1</td>
<td></td>
<td>b</td>
</tr>
<tr>
<td>gamma</td>
<td>output</td>
<td>1</td>
<td>1</td>
<td></td>
<td>gamma</td>
</tr>
<tr>
<td>a</td>
<td>input</td>
<td>1</td>
<td>1</td>
<td></td>
<td>a</td>
</tr>
</tbody>
</table>

Add/Edit ADS Parameters

Help

Type integer values to specify the matrix size of MATLAB input/output (can be 1).
Add and then select parameters to allow matrix size to be set on imported model.
Matrices are packed row-by-row, transpose for column-by-column packing.
Size information not necessary for variables specified as a matrix type.
The input/output description is optional.
Matlab IP Import Wizard

Import MATLAB Function Tool: 1

<table>
<thead>
<tr>
<th>I/O</th>
<th>type</th>
<th># rows</th>
<th># columns</th>
<th>transpose</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>b</td>
<td>output</td>
<td>1</td>
<td>1</td>
<td></td>
<td>b</td>
</tr>
<tr>
<td>gamma</td>
<td>output</td>
<td>1</td>
<td>1</td>
<td></td>
<td>gamma</td>
</tr>
<tr>
<td>a</td>
<td>input</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Edit ADS Parameters: 1

Add optional integer ADS parameters to allow specification of input/output matrix sizes on placed model.

Parameter Description: Array size
Parameter Type: Unitless
Display parameter on schematic: Not edited

Add
Cut

Help
OK
Cancel
Matlab IP Import Wizard

![Matlab IP Import Wizard Diagram]

**Help**

Type integer values to specify the matrix size of MATLAB input/output (can be 1).
Add and then select parameters to allow matrix size to be set on imported model.
Matrices are packed row-by-row; transpose for column-by-column packing.
Size information not necessary for variables specified as a matrix type.
The input/output description is optional.
Matlab IP Import Wizard

Import MATLAB Function Tool: 1

Specify additional MATLAB statements to execute

Setup

g=0;

Wrapup

g=0;

Help

Setup commands executed before simulation starts
Wrapup commands executed after simulation finishes
Warning: Syntax errors will not be indicated until simulation time.
Matlab IP Import Wizard

1. Predefined palette/library
2. Locations previously typed
3. Type in new location

Initial help text from Matlab standard header (if it exists)
Matlab IP Import Wizard

Import MATLAB Function Tool: 1

- Auto-generate symbol
- Auto-generate bitmap

Symbol name: SYM_DSN_reflect.dsn
Bitmap name: BMP_reflect.bmp

Help:

"Auto-generate" will generate a symbol and bitmap based on the specified model. Specify the complete path when using existing symbol or bitmap. If no path is specified, the local adsptolemy/symbols directory is assumed.
Matlab IP Import Wizard

Automatic instance placement only possible if library can be loaded dynamically
Matlab IP Import Wizard

Push into sub network
Matlab Cosimulation example

Generating K-Destribution Rdar Clutter
Co-Simulating a Circuit Design with a Radar System Design

Top Level Radar System Design Using Agilent Ptolemy

Subsystem Transmitter Design Using RF Budget and Circuit Envelope

Circuit Design Using Circuit Envelope
Building Block #4: Use System call in ADS

Custom or Proprietary IP- Use System call to
- User software programs

- Customers in defense industry always have their own software programs to provide special functionality

- Use system call to customer’s software can integrate custom functions in ADS as models and then use it with ADS models for whole system simulation and verification purpose.
As an example: System call to SystemVue

- Select a design in user program to be exported to ADS
- Determine what data need to be exported
- Using file write to write data for exporting
Use System call in ADS to User software

Top level design – using sequencer

- Invoke SystemVue
- Processing and measuring
- VSA measurement

Radar Test Data Generation
Use System call in ADS to User software

Step 1: Make system call using ReadFilePreProc.

- Using a template Perl file to create a Radar_Sysvu.pl in example_prj/data

- In the Radar_Sysvu.pl invoke the user software in batch mode by calling : Radar_Sysvu.bat that is a batch file with command line commands.
Use System call in ADS to User software

Step 2: Processing and Measuring Data.

- Using File Read to read in exported data from User software
- Processing and Measuring
Use System call in ADS to User software

Step 3: Processing and Measuring Data.
• Using File Read to read in exported data from User software
• Processing and then Measuring using Agilent VSA
• Agilent ESG/PSG/MXG also can be connected to down load data for hardware measurements

Measure exported data using VSA
Building Block #3: Co-Simulation with Custom/Proprietary IP

Custom or Proprietary IP: Use System call to
- Use’s programs

Diagram with two graphs showing data over time.
“Difficult to Model”- **Use Connected Solutions To:**
- Capture a measured signal with a signal analyzer; read signal into simulation as a source
- Turn simulation signal into an RF/analog/digital test signal; input into HW for analysis

---

**Agilent Technologies LTE Connected Solutions:**
- LTE Uplink EVM & Raw BER Measurement

**Modeling DUT in Time Domain**
Model Oscillators or Interferers..

Benefits:
• Models are easily created
• Very complete performance characterization
• Model is compact and portable

This extracted model contains:
- Amplitude distortions
- Time distortions
- Noise
- Spurs
- Wide BW - up to 80 MHz with VSA

Build model using VSA or PSA

Receiver Design Example

Place VSA “source” in schematic to verify system
Connected Solutions Test Setup and Characterization Parameters

**PNA Measurement Parameters**
- Center Frequency = 1.95 GHz
- Frequency Span = 40 MHz
- 201 Frequency Points
- RF Power Swept from –15 dBm to + 7 dBm in 1 dB Steps
- Port 1 Power Set to -5 dBm for Calibration
- E8358A PNA (300 kHz- 9 GHz) Used

Modeling DUT in Frequency Domain
Evaluate Hardware Re-Use with Measurement-Based Modeling

Modeling DUT in Frequency Domain

Measurement-Based Model

Model file

Amplifier P2D
AMP1
Freq=1.95 GHz
P2DFile="meas_mga27543.p2d"

Title of Presentation
Agilent Restricted

Date
Restricted

Agilent Technologies
Summary

• Custom signal formats can easily be modeled in ADS using one of the “Four Building Blocks”

• ADS co-simulation allows Matlab® and circuit designs to be simulated together to help minimize integration risk

• ADS can integrate customer’s program as ADS model for simulation

• Connected Solution is a useful way not only for testing/verification but also for modeling your systems
Special Thanks To:

David Leiss, Greg Jue, Frank Ditore, Stephen Kratzet, Igor Koyfman of Agilent Technologies for their Key Contributions to this Work