

MATLAB EXPO 2021

使用MATLAB进行混合信号系统的设计与验证

严小商, MathWorks



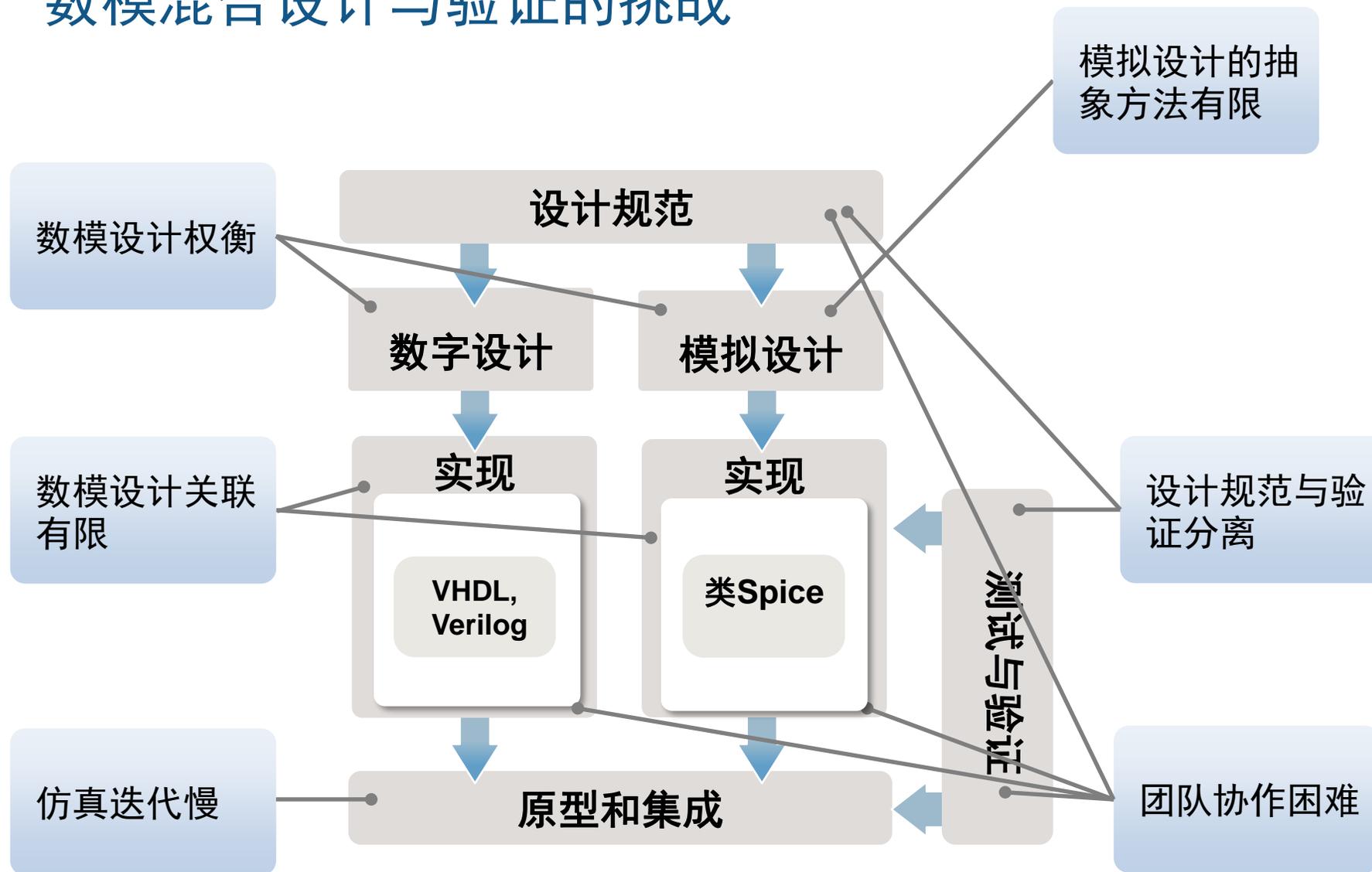
内容

- 数模混合设计与验证的挑战
- MATLAB和Simulink用于数模混合电路开发
 - 利用Mixed-Signal Blockset开发ADC/DAC及PLL
 - 用SerDes Toolbox开发SerDes系统并生成IBIS-AMI模型
- MATLAB和Simulink与其他EDA工具集成
- 总结及如何获取更多相关信息

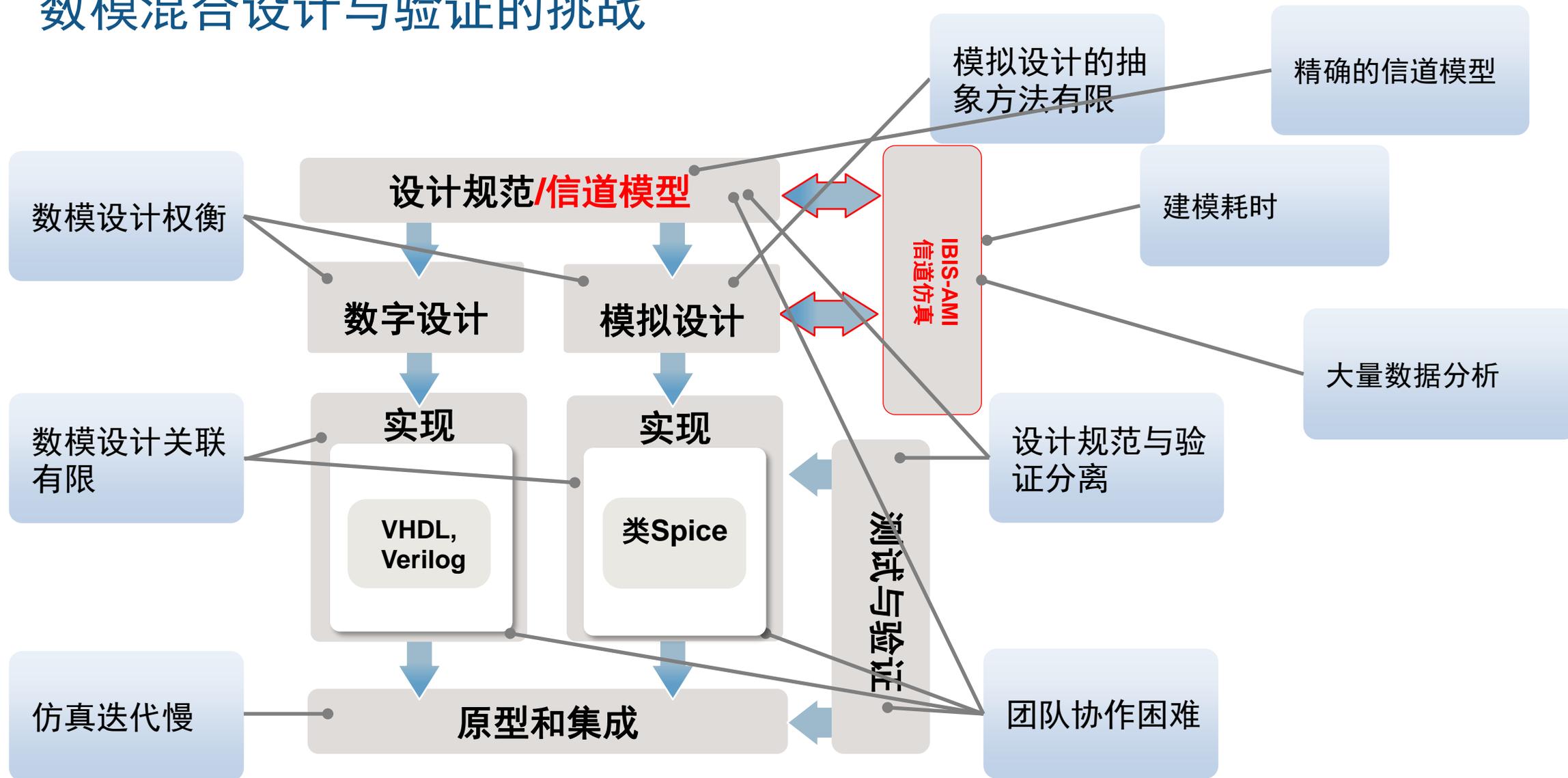
内容

- 数模混合设计与验证的挑战
- MATLAB和Simulink用于数模混合电路开发
 - 利用Mixed-Signal Blockset开发ADC/DAC及PLL
 - 用SerDes Toolbox开发SerDes系统并生成IBIS-AMI模型
- MATLAB和Simulink与其他EDA工具集成
- 总结及如何获取更多相关信息

数模混合设计与验证的挑战



数模混合设计与验证的挑战

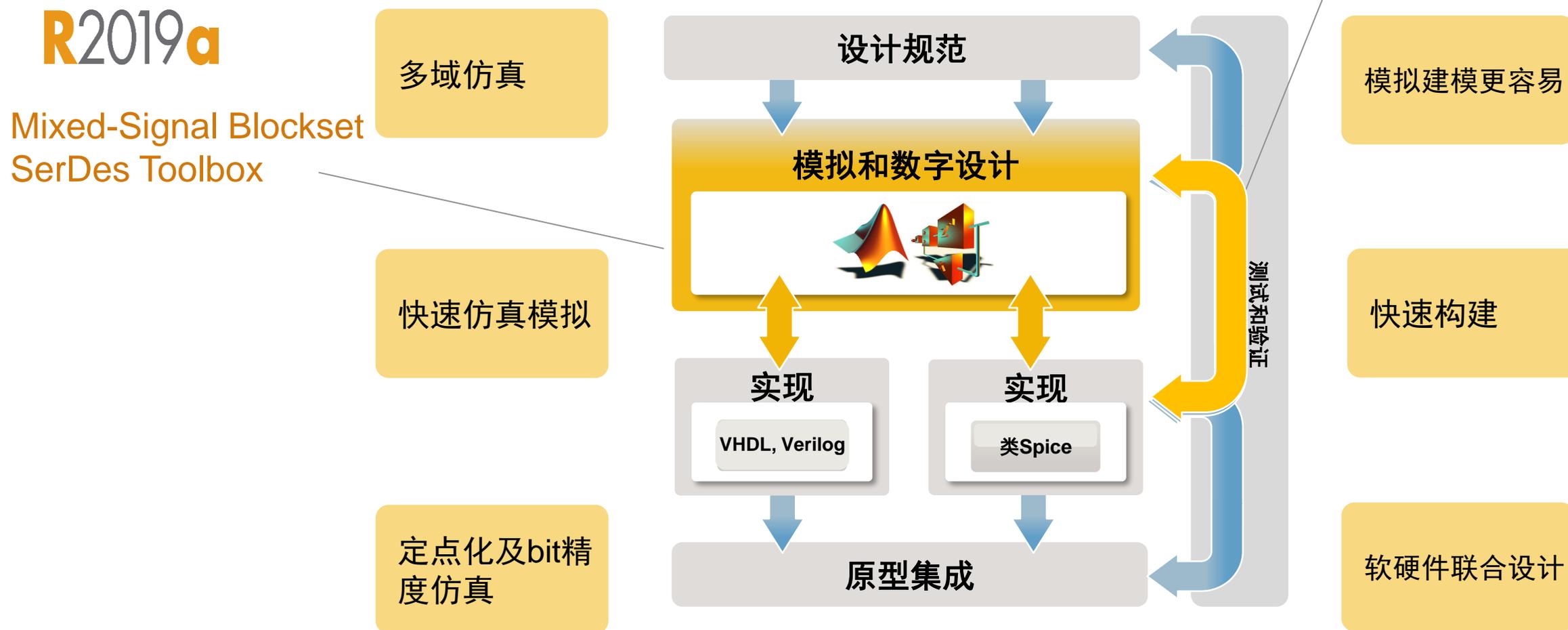


内容

- 数模混合设计与验证的挑战
- MATLAB和Simulink用于数模混合电路开发
 - 利用Mixed-Signal Blockset开发ADC/DAC及PLL
 - 用SerDes Toolbox开发SerDes系统并生成IBIS-AMI模型
- MATLAB和Simulink与其他EDA工具集成
- 总结及如何获取更多相关信息

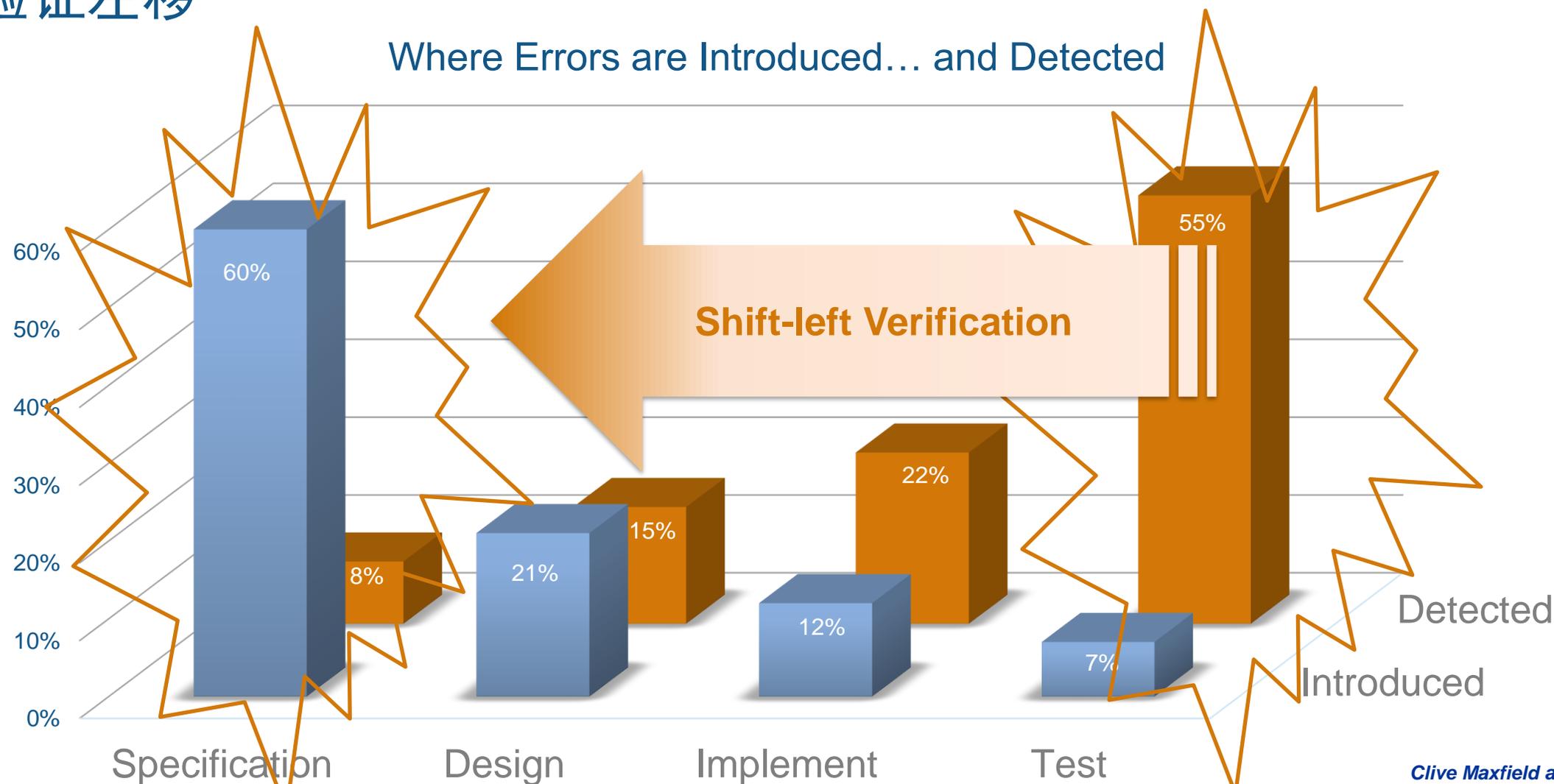
从系统级设计及分析到持续验证

与EDA工具及信道仿真器集成



验证左移

Where Errors are Introduced... and Detected

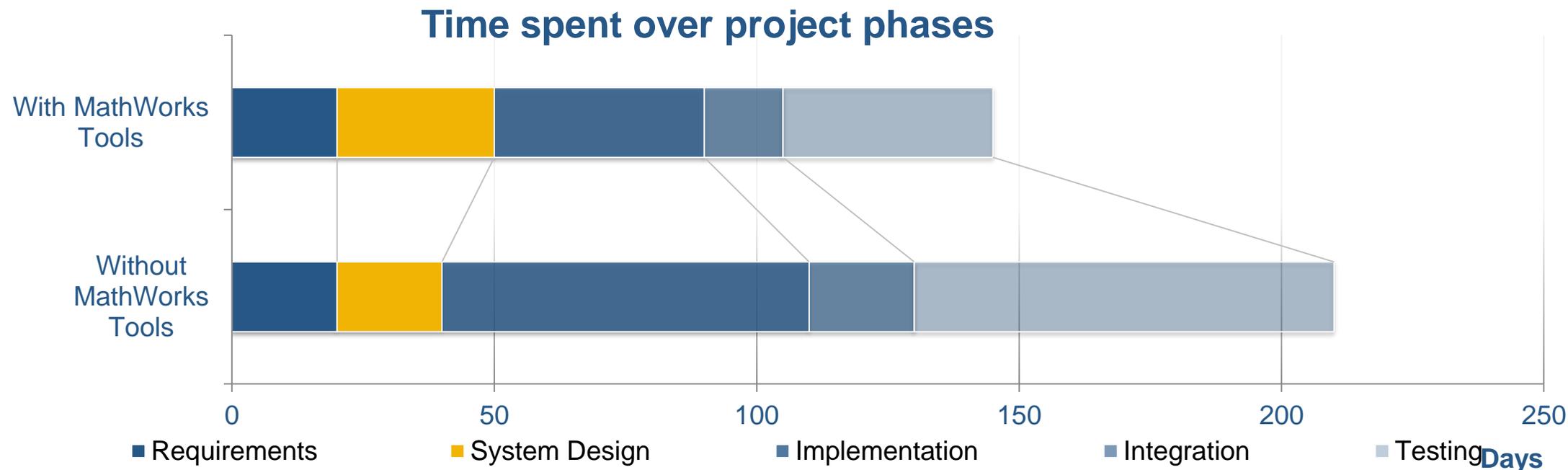


加强建模和仿真

传统验证方法

Clive Maxfield and Kuhoo Goyal
"EDA: Where Electronics Begins"

节省 30% 的开发时间 (提升质量, 减少返工)



EE Times - Top-down verification guides mixed-signal designs

[K. Kundert and H. Chang, Partners, Designer's Guide Consulting](#)

“为应对这些挑战,许多设计团队正在寻找或者已经实施了自顶向下的设计方法。在自顶向下的设计中,芯片的构架定义为框图,并使用MATLAB或Simulink这类系统仿真工具进行仿真和优化。从这种高层级仿真中,推导出各个电路模块的需求。”

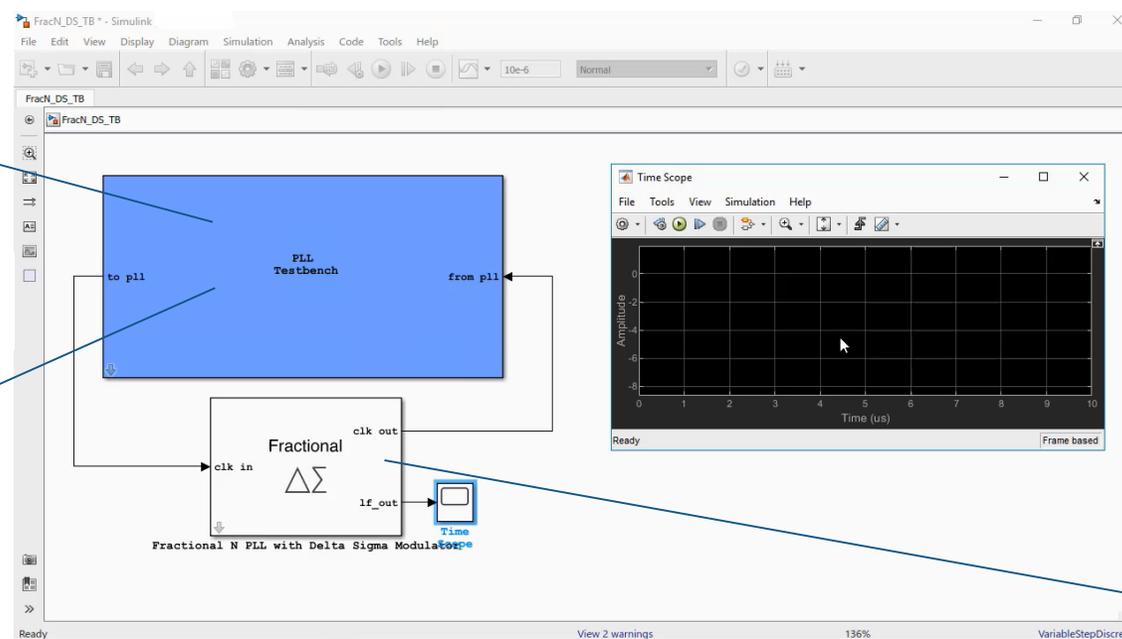
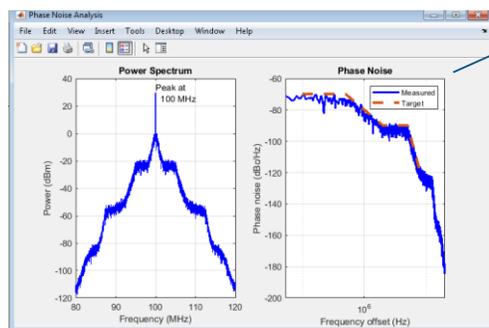
Mixed-Signal Blockset

设计和仿真模拟混合信号系统

- 连续和离散信号的时域仿真
- 行为模型的Simulink库：**PLL**、**ADC/DAC**
 - 可自定义、用于自顶向下设计典型构架
 - 典型组成模块，可注入模拟损伤
 - 测量模块及Testbench

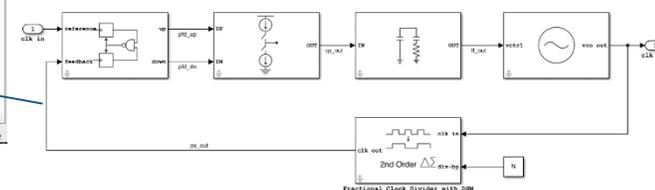
测量和Testbench

相噪分析



组成模块 (可注入模拟损伤)

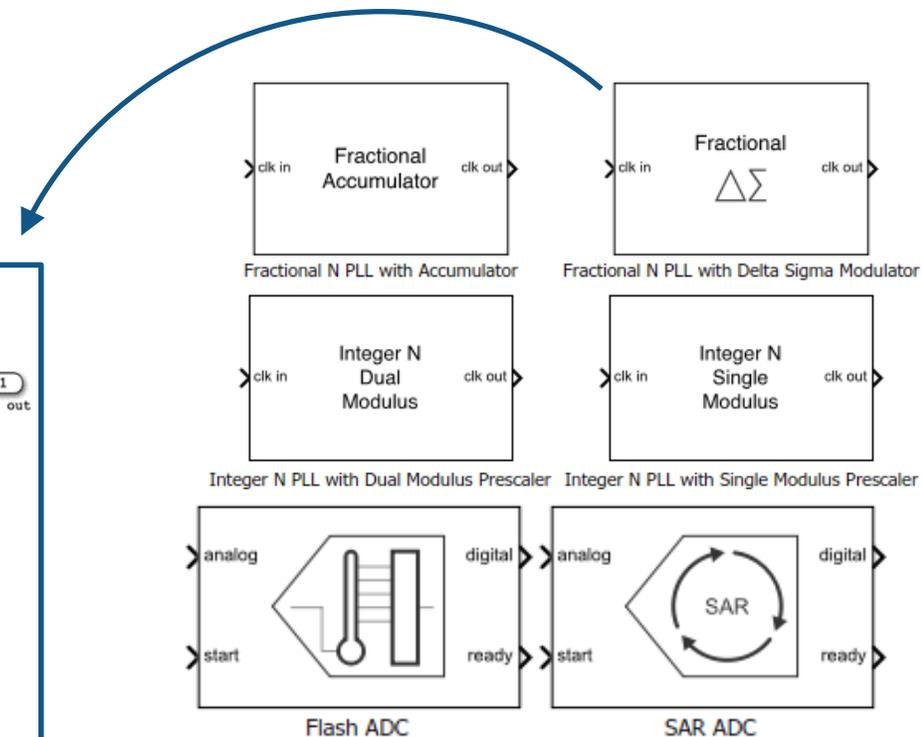
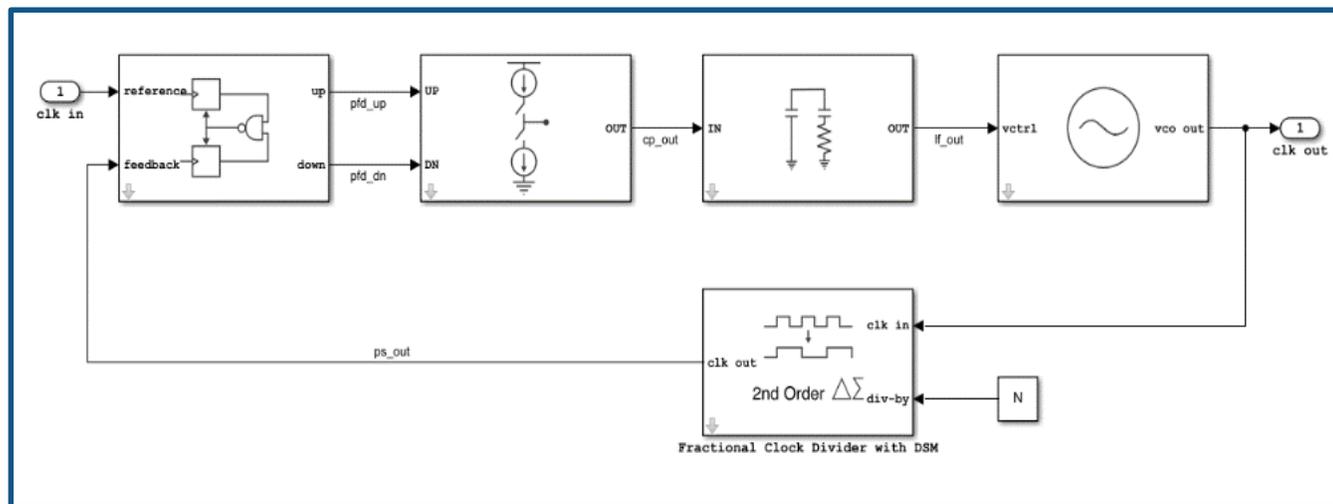
白盒化的构架模型



Mixed-Signal Blockset: 构架模型

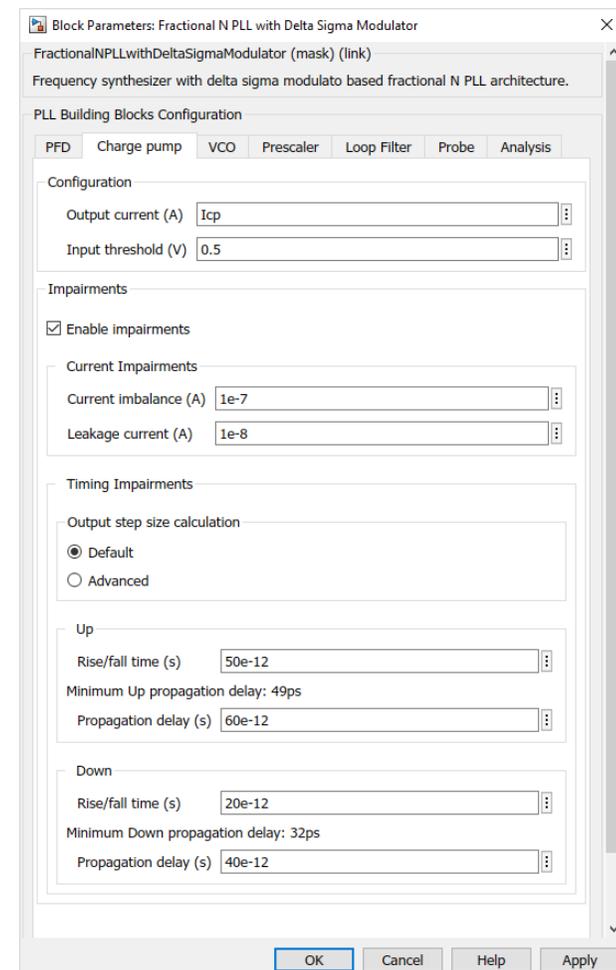
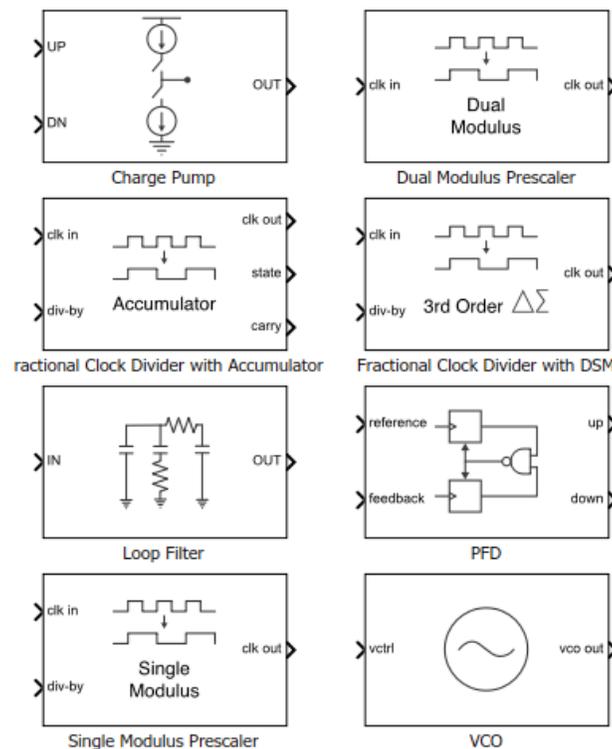
白盒构架模型:

- 带单/双前置分频器的整数分频PLL
- 带累加器/delta-sigma调制的小数分频 PLL
- Flash, SAR ADCs
- 二进制加权DAC



Mixed-Signal Blockset: 组成模块

- 包含损伤的组成模块
 - 上升和下降时间
 - 泄露、电流失衡
 - 相噪
 - 热噪
 - 量化效应
 - 孔径抖动
 - 失调、增益误差



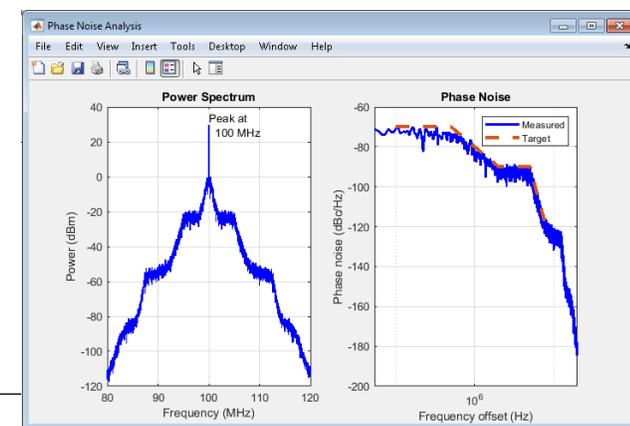
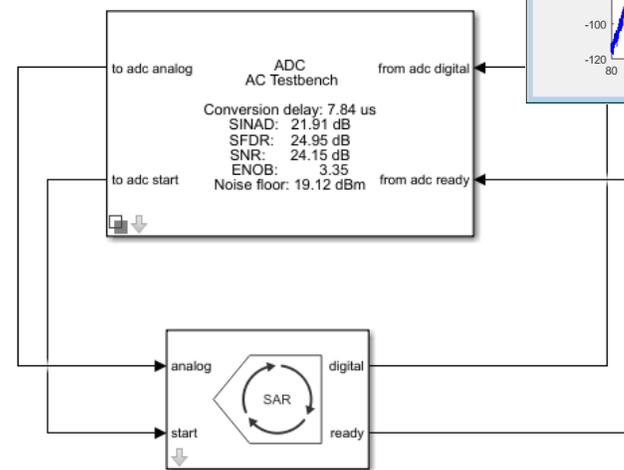
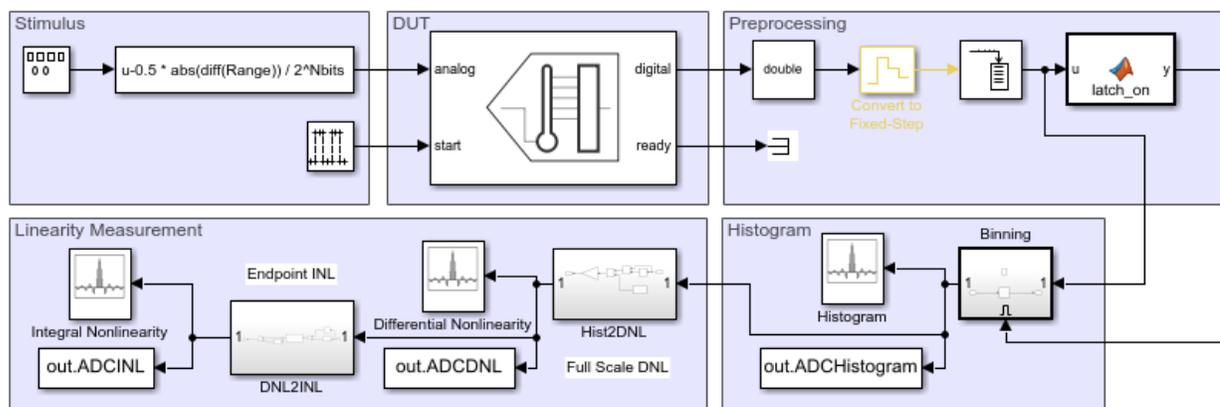
Mixed-Signal Blockset: 测量和Testbench

验证 ADC/DAC 和 PLL 的性能

- ADC / DAC 直流测量：偏移、增益误差、积分非线性、微分非线性...
- ADC / DAC 交流测量：信纳比、无杂散动态范围、信噪比、有效位数、噪底 ...
- 孔径抖动
- PLL 锁定时间, 频率, 相噪
- MATLAB function

Histogram Linearity Measurement Example

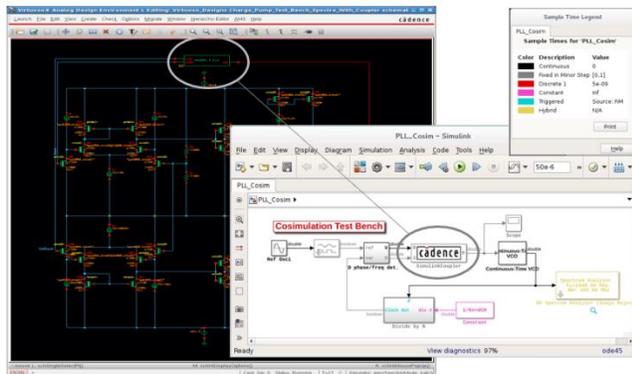
Copyright 2019 The MathWorks, Inc.



Mixed-Signal Blockset Models: 例子和教程

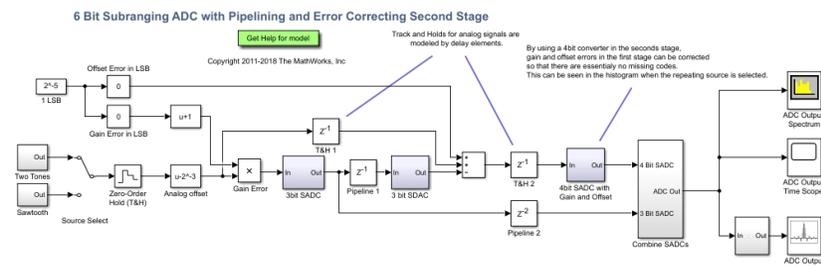
PLL

- PLL 设计教程
- PLL 行为模型 (带损伤参数)
- PLL 2.4GHz 及与 Cadence Virtuoso AMS Designer 联合仿真
- PLL 50x (含误差测量)
- PLL (带双前置分频)
- PLL (小数分频累加器及误差校正)



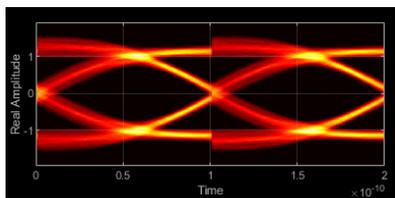
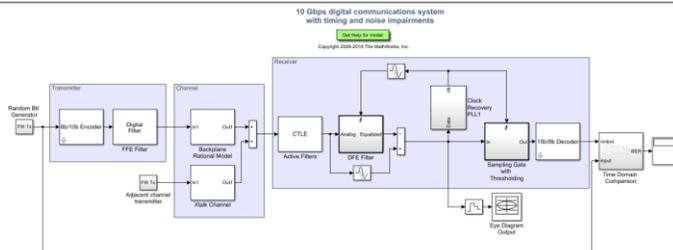
ADC

- ADC 设计教程 (含Cadence Incisive 联合仿真)
- Sigma-Delta ADC 的SystemVerilog 建模
- 三阶Sigma-Delta ADC (含电路级实现)
- 四阶Sigma-Delta ADC



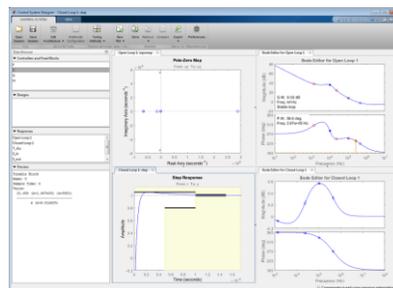
SerDes

- SerDes 设计教程
- 背板建模及App
- 64b/66b 编码
- 64b/67b 编码
- 8b/10b编码
- 可调均衡器及浴盆曲线生成
- 时钟恢复
- 10 Gbps SerDes设计
- 2 Gbps SerDes (带电路级CTLE)

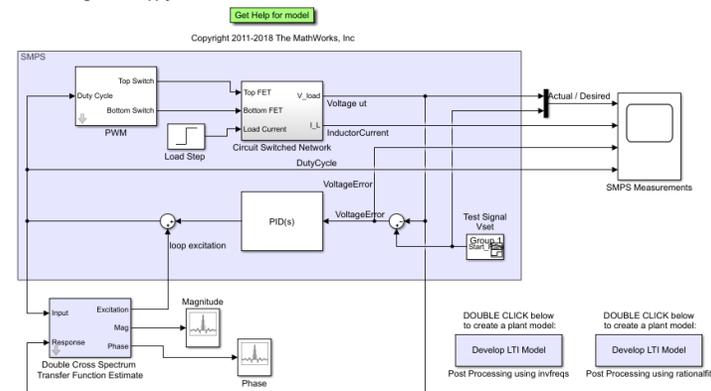


SMPS

- 开关电源设计教程
- Boost
- Buck
- Flyback
- SEPIC

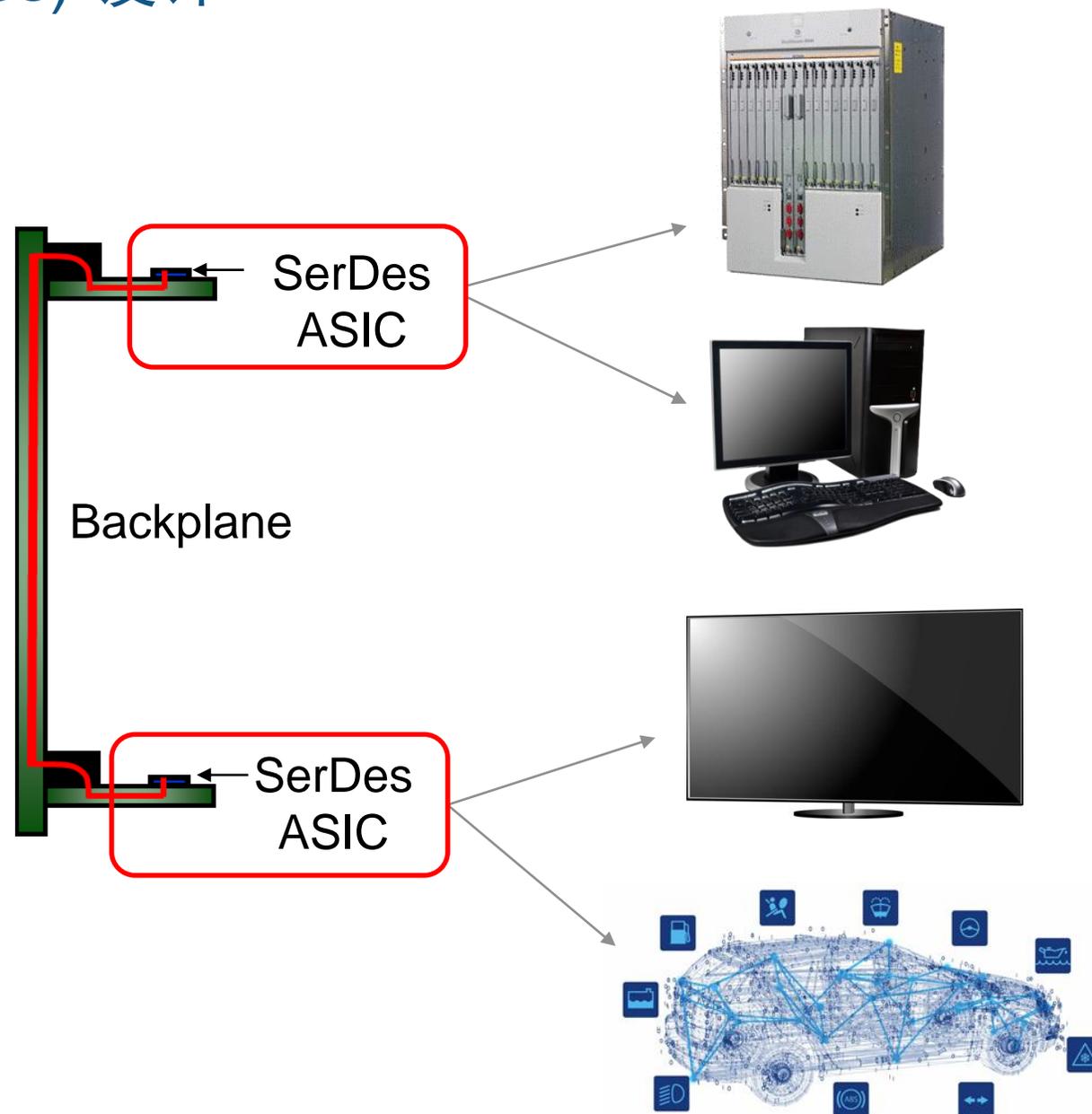


Switching Power Supply with PID Controller and Plant Transfer Function Estimation



Serializer/Deserializer (SerDes) 设计

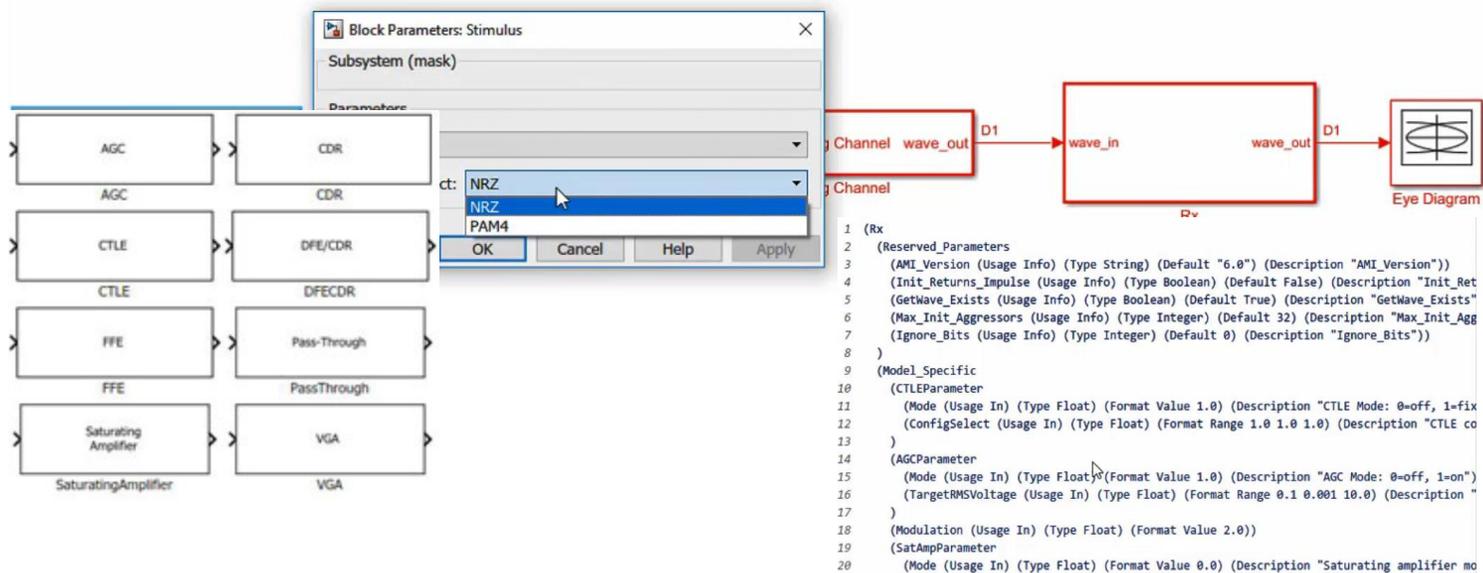
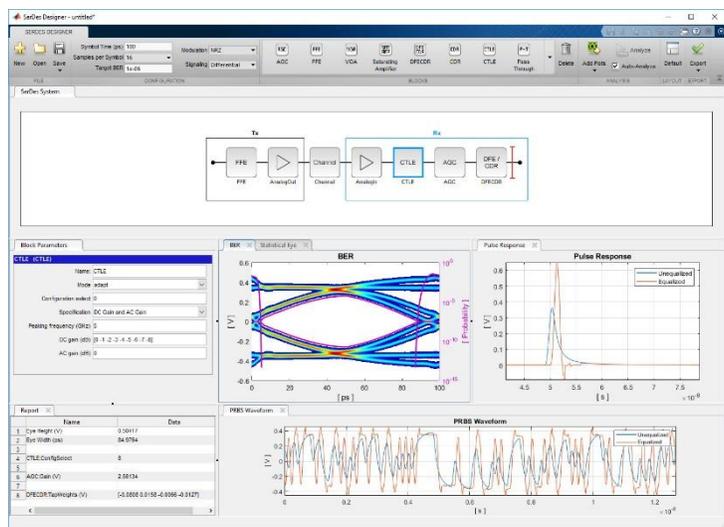
- 高速数据通信系统
 - 数据中心
 - 电脑
 - 消费电子
 - 汽车电子
- 协议形式: PCI-Express, Ethernet, USB.....
- 数字和模拟模块的复杂组合
 - PLL/CDR
 - CTLE
 - DFE/FFE



SerDes Toolbox

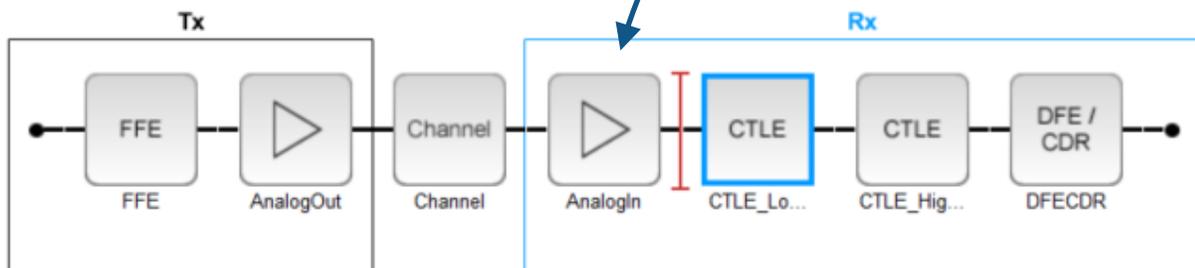
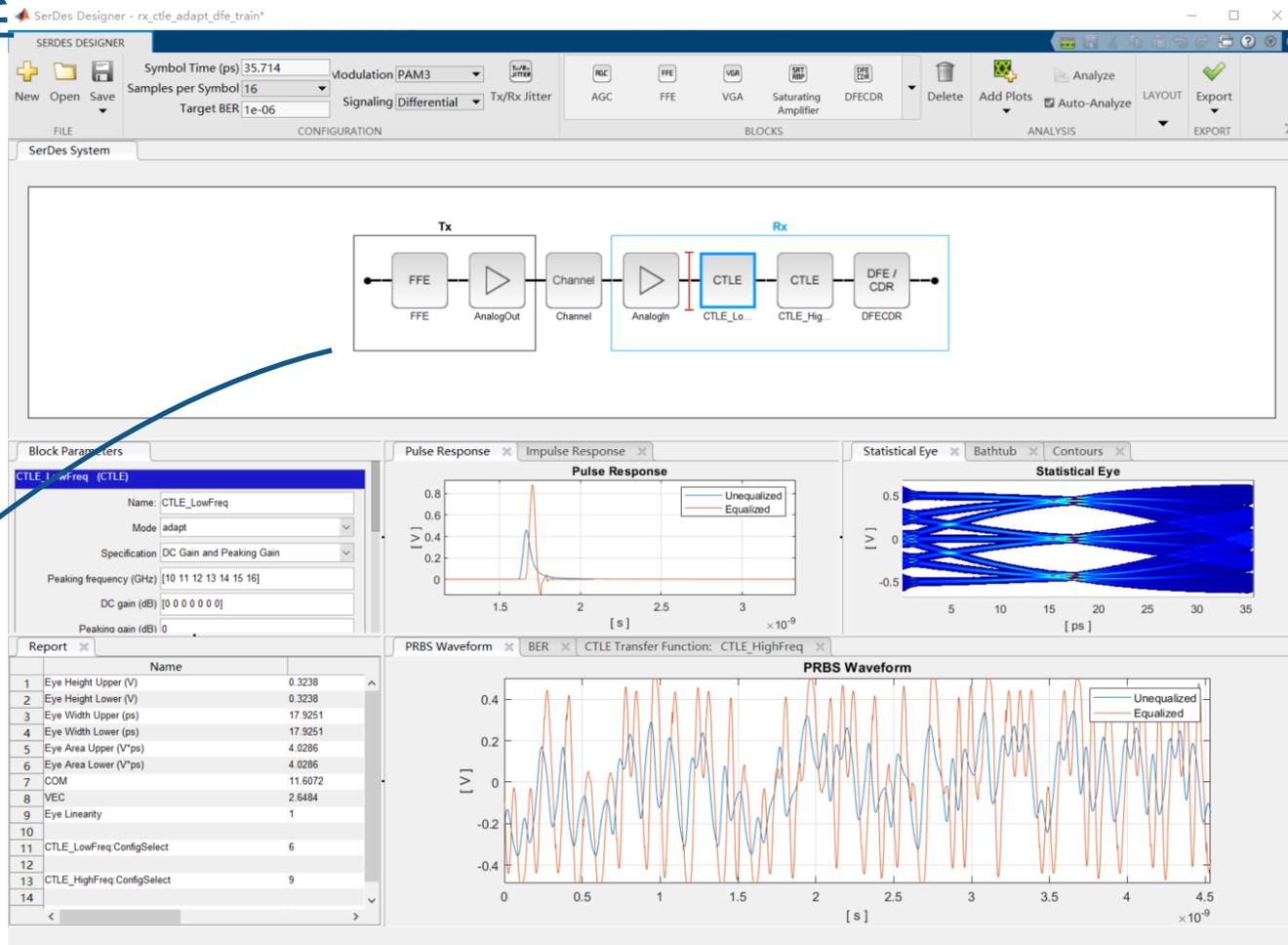
为高速互联接口设计SerDes系统并生成IBIS-AMI模型

- 用SerDes Designer App设计和分析SerDes收发系统
- 开发均衡算法：
 - FFE, DFE, AGC, CDR, CTLE...
- SerDes静态分析和时域仿真
- 生成单/双端IBIS-AMI模型用于三方信道仿真器
- 参考设计： Ethernet CEI-56G, DDR5, PCI-Gen4, USB3.1



SerDes Designer App: 快速上手

- 设计和分析SerDes系统
- 内置组件模块
- 静态分析: 眼图, BER, 浴盆曲线, 脉冲响应, Channel Operating Margin (COM)



SerDes Designer App: 快速上手

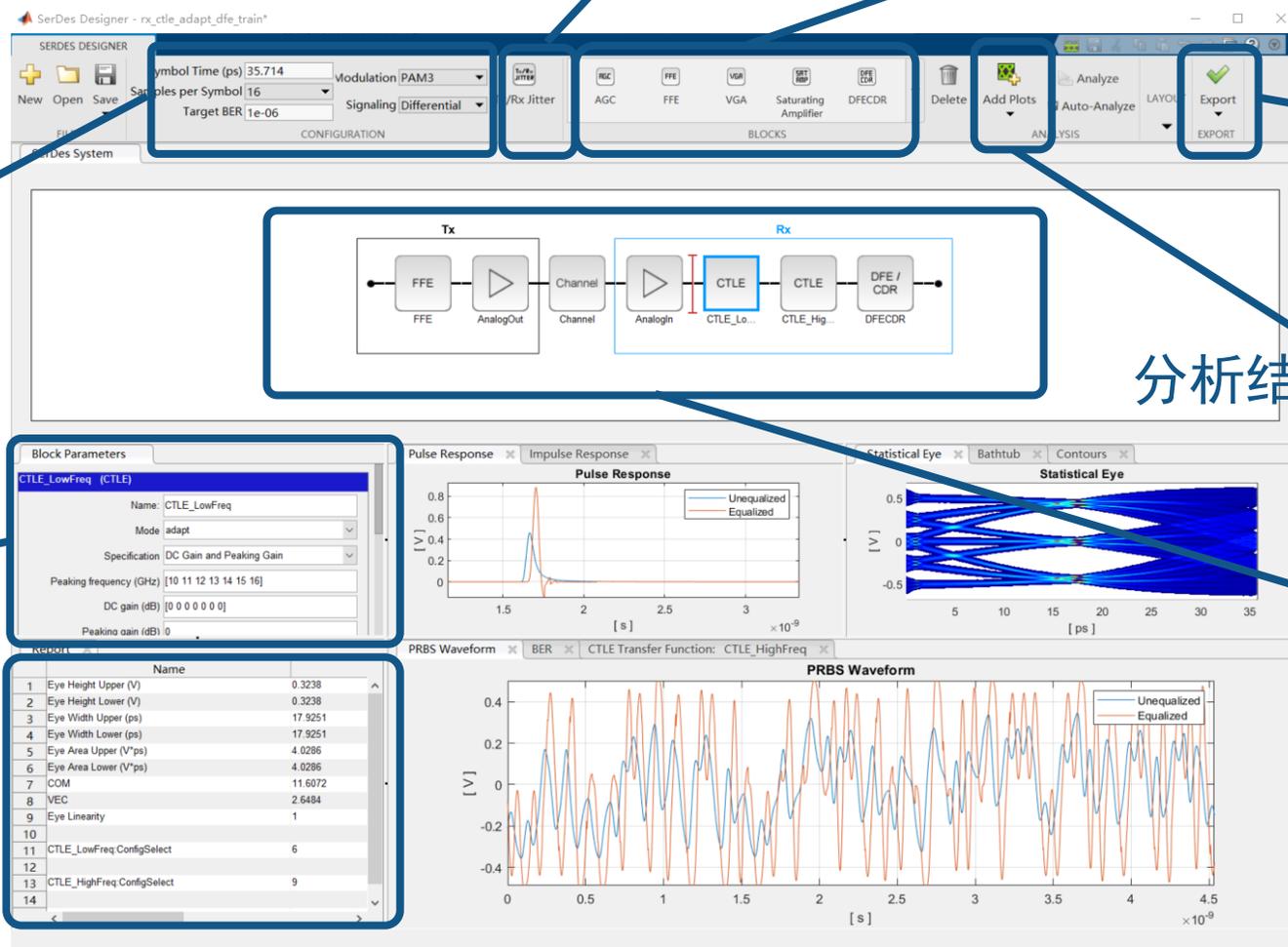
SerDes组件

抖动

导出:

- MATLAB
- Simulink
- IBIS-AMI

- 调制
- 符号速率
- 信号



分析结果画图

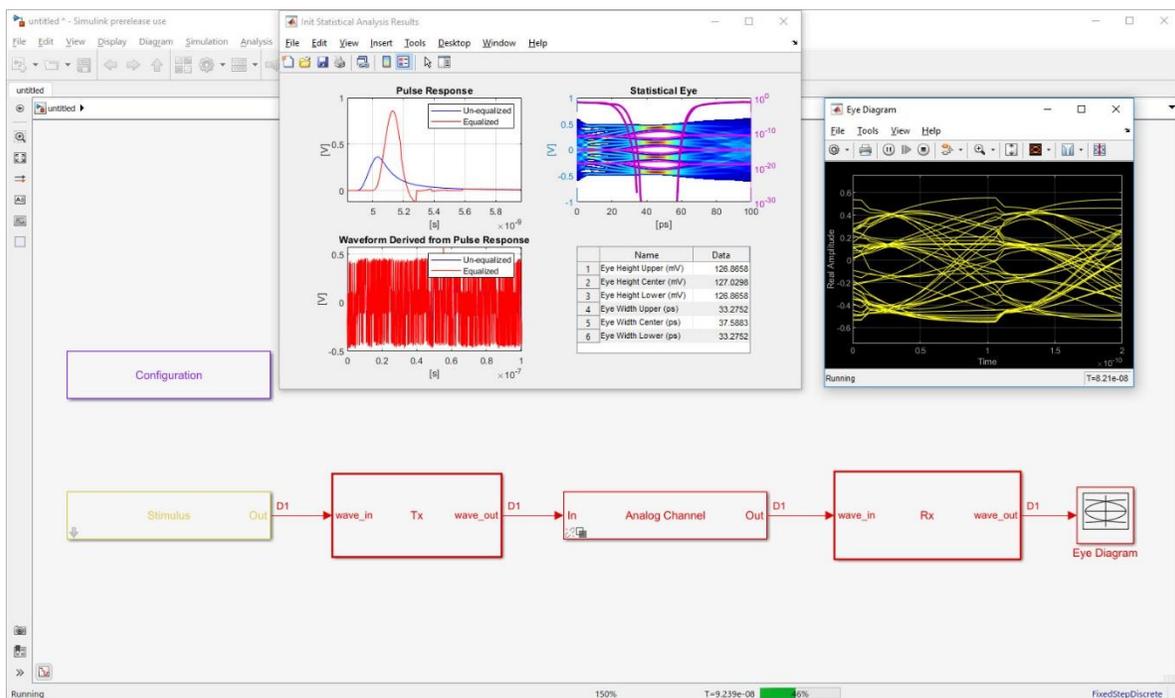
组件参数

SerDes链路

报表

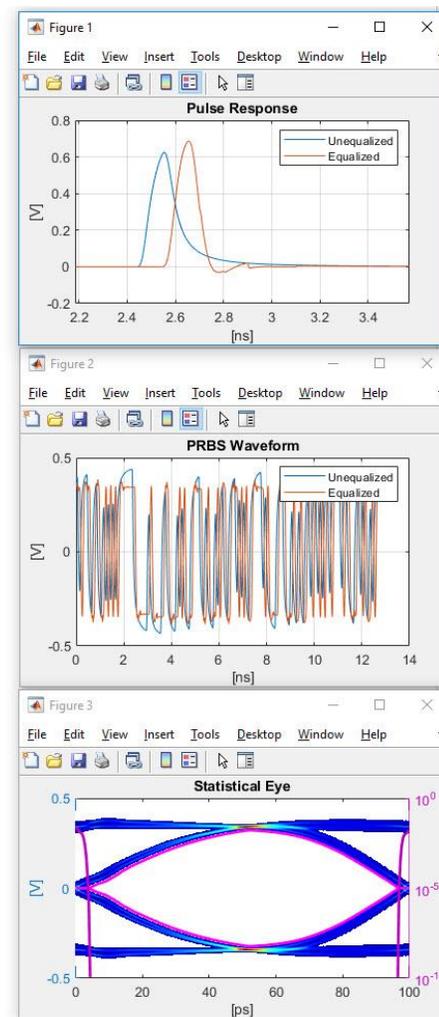
SerDes 设计流程

- 创建 MATLAB 脚本用于自动化设计和探索
- 导出Simulink模型用于时域仿真
- 创建 IBIS-AMI模型



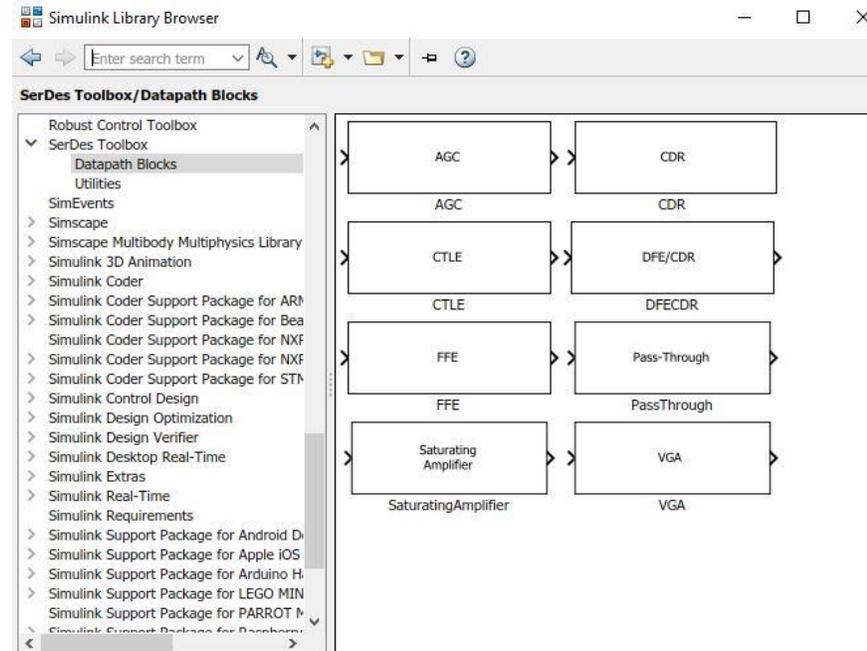
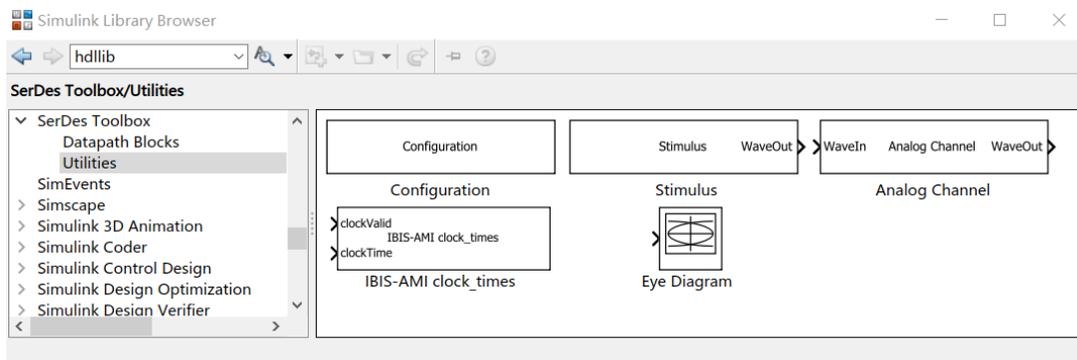
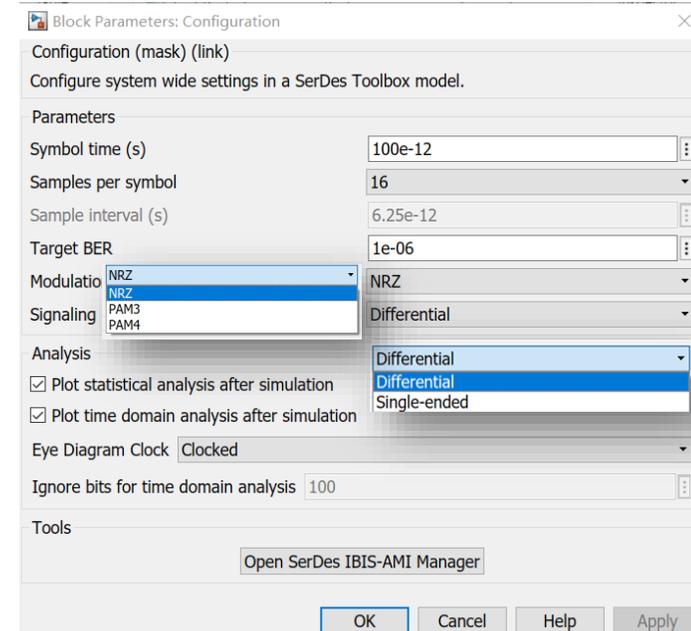
```

1
2
3
4
5 % Build cell array of Tx blocks:
6 txBlocks(1) = serdes.FFE;
7 txBlocks(1).BlockName = 'FFE';
8 txBlocks(1).Mode = 1;
9 txBlocks(1).TapWeights = [0 1 0 0 0];
10 txBlocks(1).Normalize = true;
11
12
13 % Build cell array of Rx blocks:
14 rxBlocks(1) = serdes.CTLE;
15 rxBlocks(1).BlockName = 'CTLE';
16 rxBlocks(1).Mode = 2;
17 rxBlocks(1).ConfigSelect = 0;
18 rxBlocks(1).Specification = 'DC Gain and Peaking Gain';
19 rxBlocks(1).PeakingFrequency = 5000000000;
20 rxBlocks(1).DCGain = [0 -1 -2 -3 -4 -5 -6 -7 -8];
21 rxBlocks(1).PeakingGain = [0 1 2 3 4 5 6 7 8];
22
23
24 rxBlocks(2) = serdes.AGC;
25 rxBlocks(2).BlockName = 'AGC';
26 rxBlocks(2).Mode = 1;
27 rxBlocks(2).TargetRMSVoltage = 0.3;
28
29
30 rxBlocks(3) = serdes.DFECDD;
31 rxBlocks(3).BlockName = 'DFECDD';
32 rxBlocks(3).Mode = 2;
33 rxBlocks(3).TapWeights = [0 0 0 0];
34 rxBlocks(3).MinimumTap = -1;
35 rxBlocks(3).MaximumTap = 1;
36
37 % Build txModel:
38 txAnalogModel = AnalogModel( ...
39 'R', 50, ...
40 'C', 1.000000e-13);
41 tx = Transmitter( ...
42 'Blocks', txBlocks, ...
43 'AnalogModel', txAnalogModel, ...
44 'RiseTime', 1.000000e-11, ...
45 'VoltageSwingIdeal', 1, ...
46 'Name', 'TX');
47
48 % Build rxModel:
49 rxAnalogModel = AnalogModel( ...
50 'R', 50, ...
51 'C', 2.000000e-13);
52 rx = Receiver( ...
53 'Blocks', rxBlocks, ...
54 'AnalogModel', rxAnalogModel, ...
55 'Name', 'RX');
56
57 % Build ChannelData:
58 channel = ChannelData( ...
59 'ChannelLossdB', 8, ...
60 'ChannelLossFreq', 5000000000, ...
61 'ChannelDifferentialImpedance', 100);
62
63 % Build SerDes System:
64 SymbolTime = 1e-10;
65 SamplesPerSymbol = 16;
66 ModulationLevels = 2;
67 BERtarget = 1e-06;
    
```



SerDes Toolbox: Simulink 模型

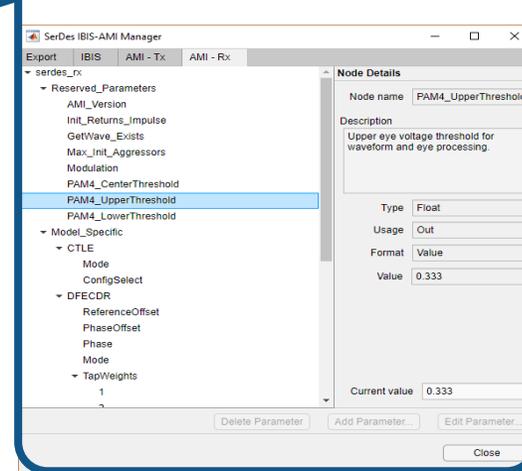
- 内置白盒模型，如 DFE, CTLE, AGC, CDR
- 参数化模型和算法
- 支持单端、差分信号
- PRBS或自定义激励信号
- 支持PAM4、NRZ、PAM3



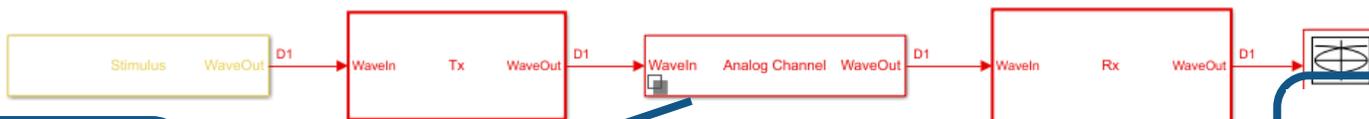
SerDes仿真和构架探索

- 自定义信道，可导入S参数
- 可使用其他Simulink模块
- 可配置Jitter和噪声
- 生成IBIS-AMI模型

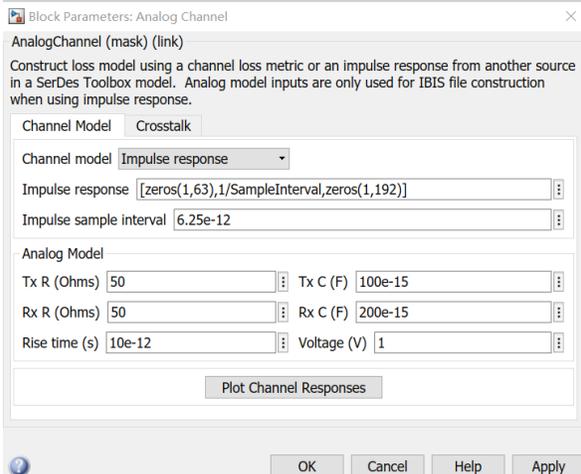
参数配置



Configuration



信道模型



白盒模型(可自定义)

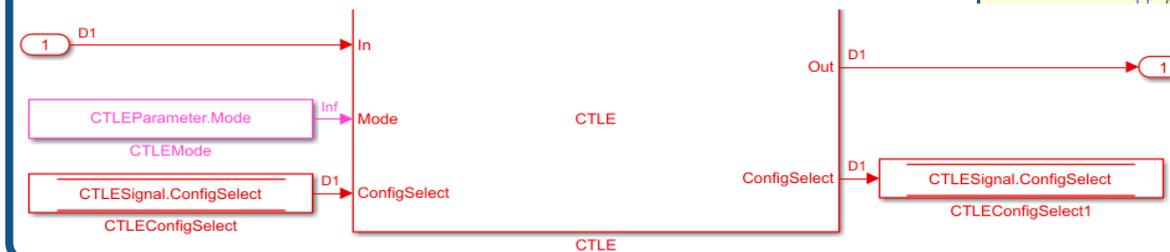
```
function [waveOut, Config] = stepimp1(obj,waveIn)

if isSample(obj) && (modeIsFixed(obj) || modeIsAdapt(obj))
    %Note that step or time domain adaptation of the CTLE must be
    %done in an exterior block.

if modeIsFixed(obj)
    Config = obj.ConfigSelect;
else
    if obj.privConfigInitialFlag
        obj.privConfigInitial = obj.ConfigSelect;
        obj.privConfigInitialFlag = false;
    else
        obj.privConfigInitial = obj.ConfigSelect;
    end
end

obj.FilterCoefficients.np(Config+1)+1;
obj.FilterCoefficients.nz(Config+1)+1;
```

仿真和自定义自适应均衡器



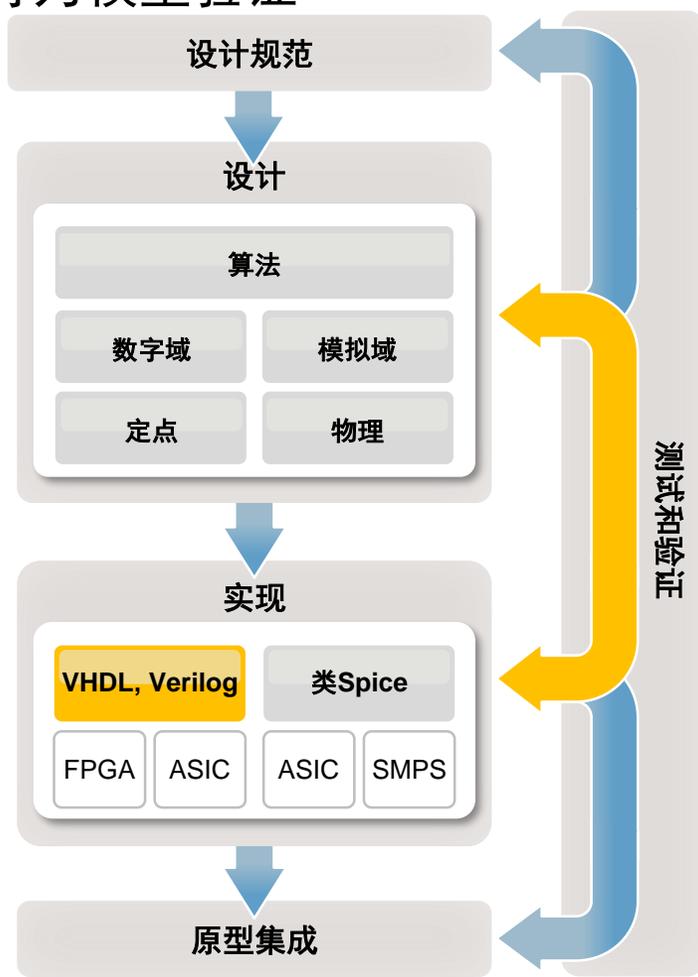
内容

- 数模混合设计与验证的挑战
- MATLAB和Simulink用于数模混合电路开发
 - 利用Mixed-Signal Blockset开发ADC/DAC及PLL
 - 用SerDes Toolbox开发SerDes系统并生成IBIS-AMI模型
- MATLAB和Simulink与其他EDA工具集成
- 总结及如何获取更多相关信息

与EDA工具集成的两种方法

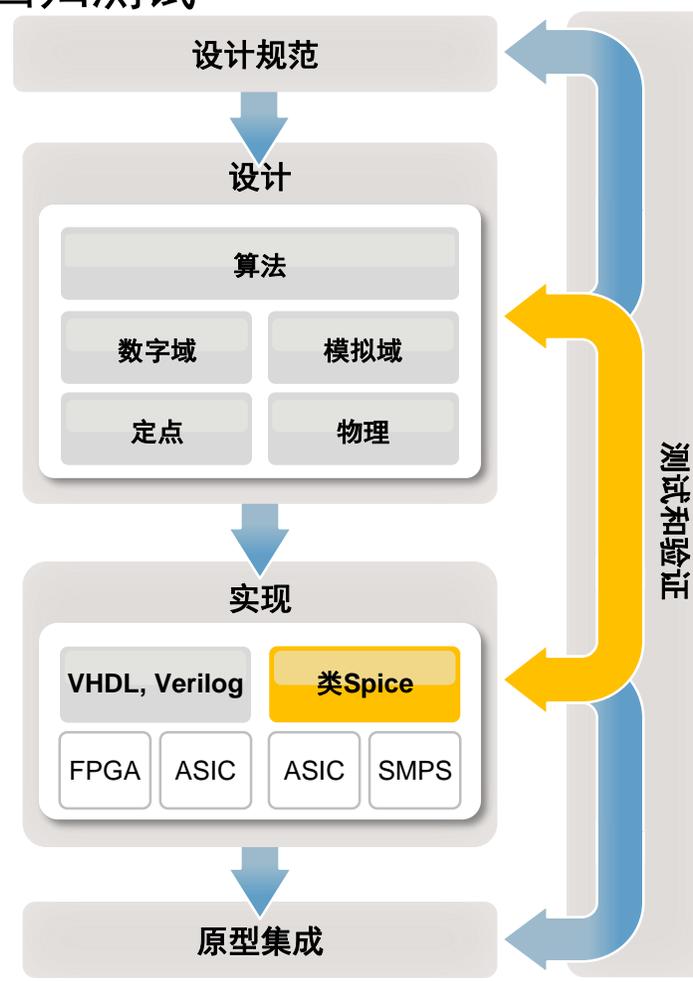
联合仿真

- 调试
- 行为模型验证

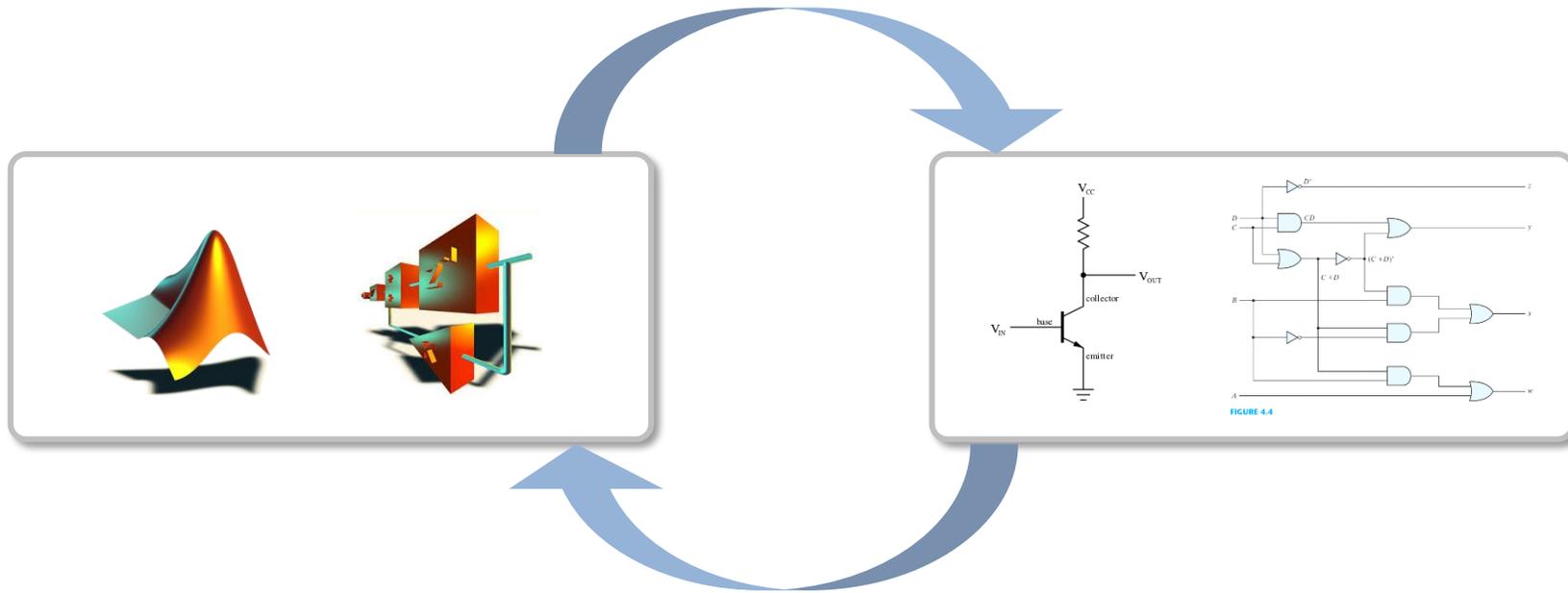


代码生成

- 生成Testbench
- 回归测试



方法一：联合仿真



两种联合仿真方法

数字域联合仿真

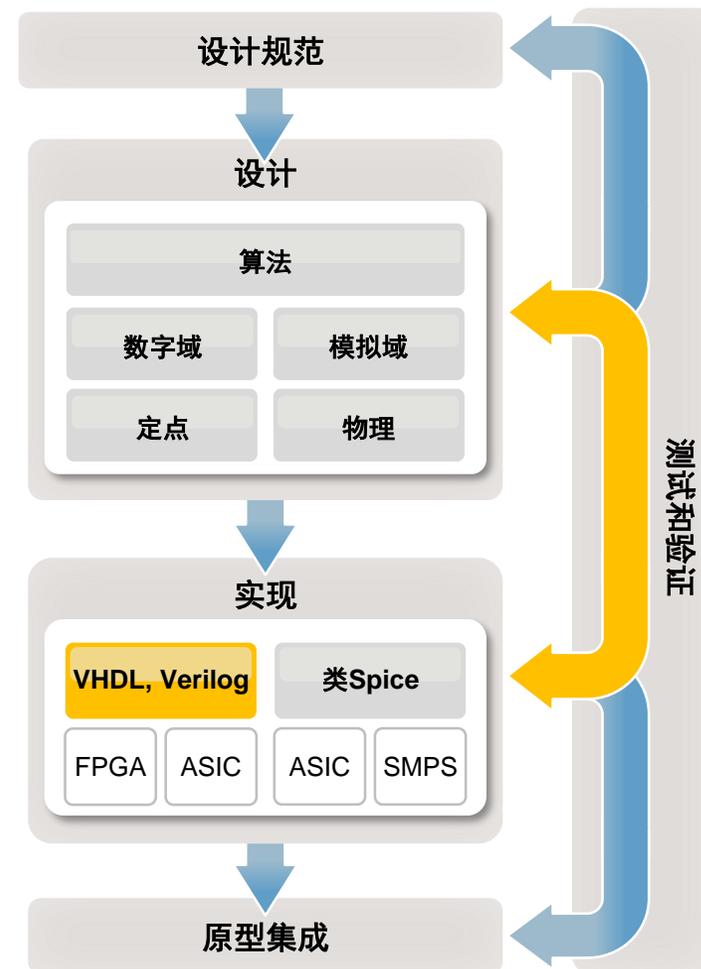
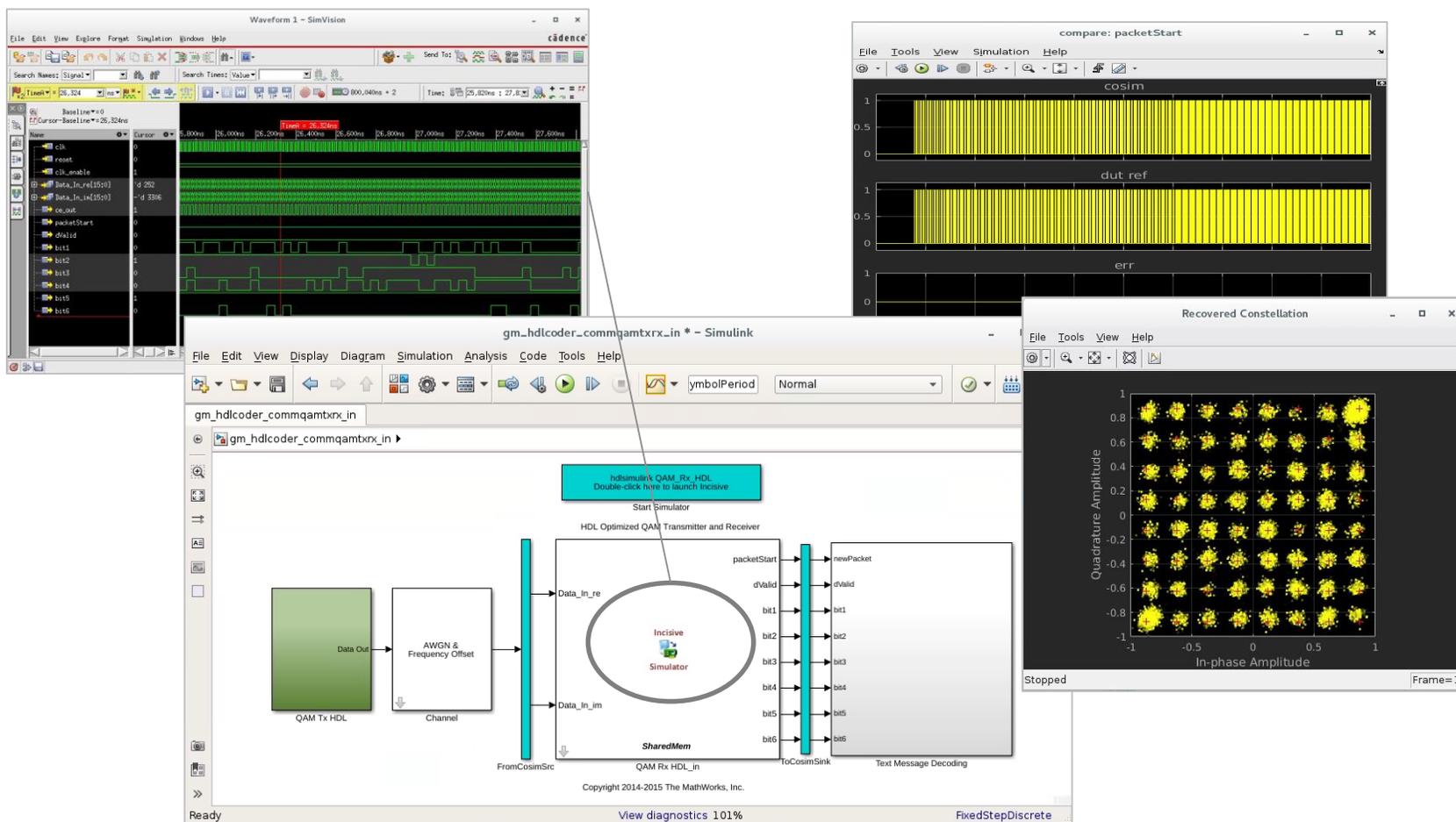
- 利用标准的 HDL API
- 基于离散时间信号
- HDL Verifier支持如下工具：
 - Cadence Incisive
 - Mentor Graphics QuestaSim
 - Synopsys VCS (Pilot support)
- 支持自动生成的HDL 代码及已有HDL代码
- MATLAB和Simulink环境都支持

模拟域联合仿真

- 没有标准的模拟 API
- 基于连续时间信号 (需要可变步长求解器)
- Cadence工具的支持：
 - Virtuoso AMS Designer
 - Allegro Pspice
- Simulink环境支持

验证电路设计: 数字域联合仿真

- 验证HDL实现与可执行设计规范的符合性

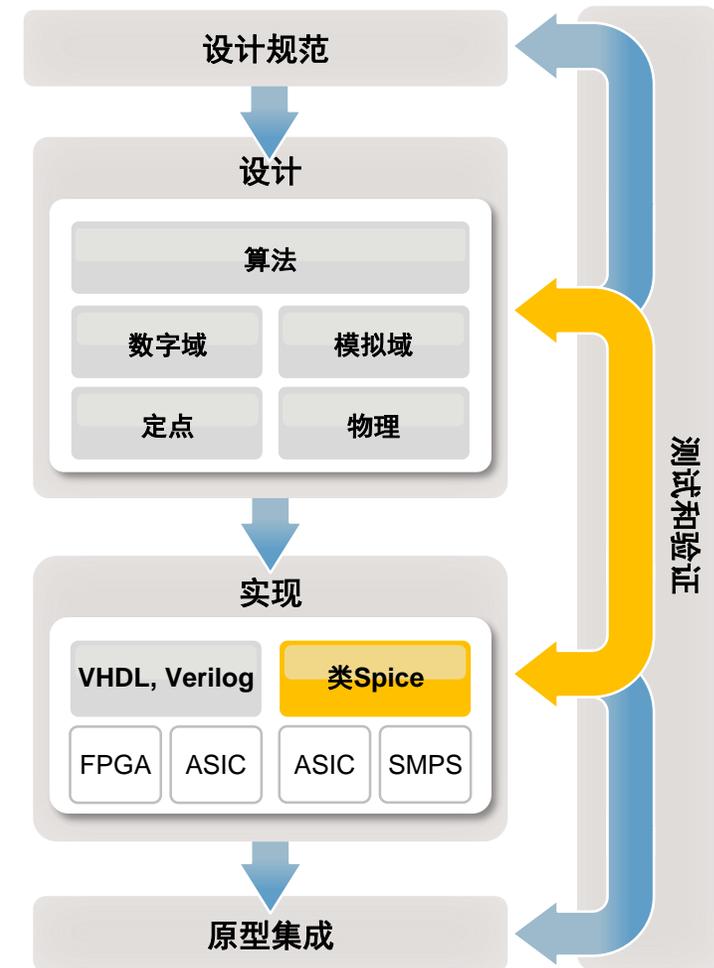


验证电路设计: 模拟域联合仿真

- 验证晶体管级实现与可执行设计规范的符合性

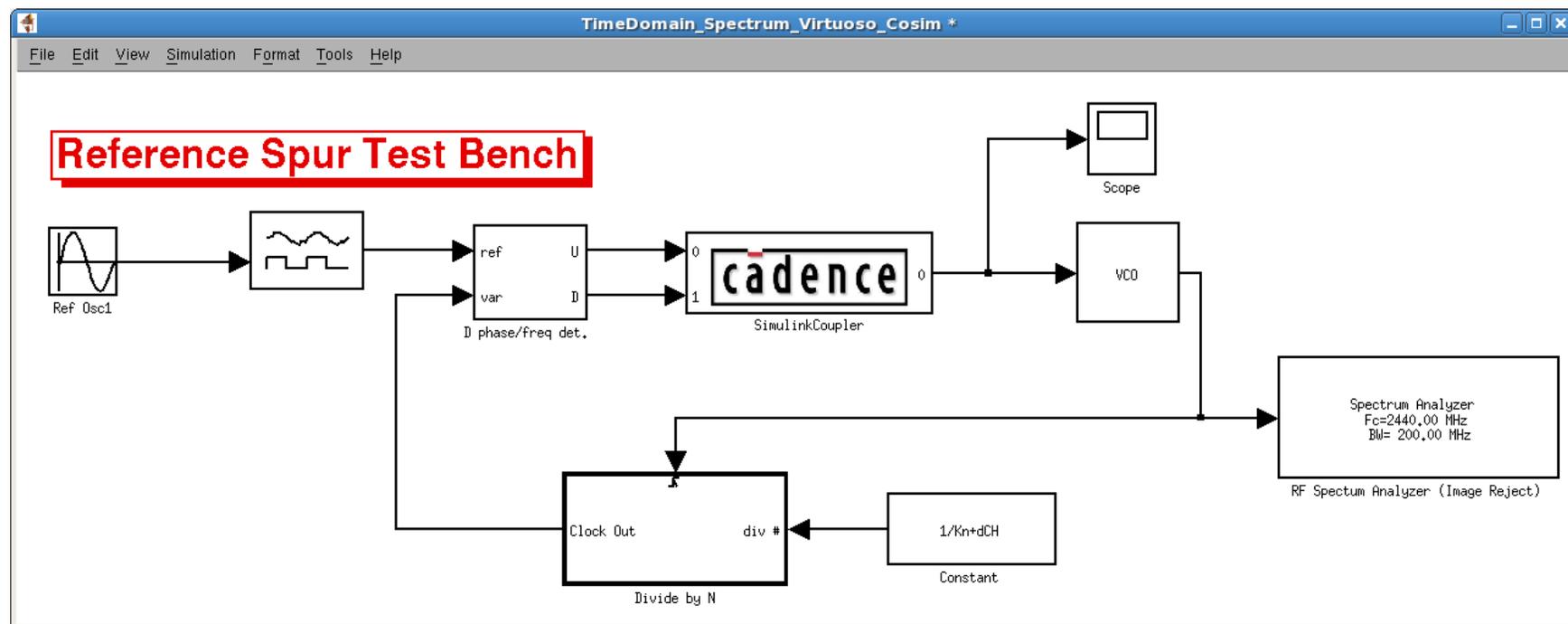
The image shows a screenshot of the Cadence Virtuoso Analog Design Environment. The main window displays a complex circuit schematic with various components like transistors, capacitors, and resistors. A specific component, a coupler, is highlighted with a red circle. Below the schematic, there is a Simulink window titled 'PLL_Cosim - Simulink'. This window shows a 'Cosimulation Test Bench' diagram. The diagram includes a 'Ref Osc1' block, a 'boolean' block, a 'double' block, a 'var' block, a 'cadence' block (representing the transistor-level model), a 'SimulinkCoupler' block, a 'Continuous-Time VCO' block, a 'Scope' block, a 'Spectrum Analyzer' block, and a 'Divide by N' block. A 'Sample Time Legend' window is also visible, showing sample times for 'PLL_Cosim' with columns for Color, Description, and Value.

Color	Description	Value
Black	Continuous	0
Grey	Fixed in Minor Step	[0,1]
Red	Discrete 1	5e-09
Magenta	Constant	Inf
Cyan	Triggered	Source: FIM
Yellow	Hybrid	N/A

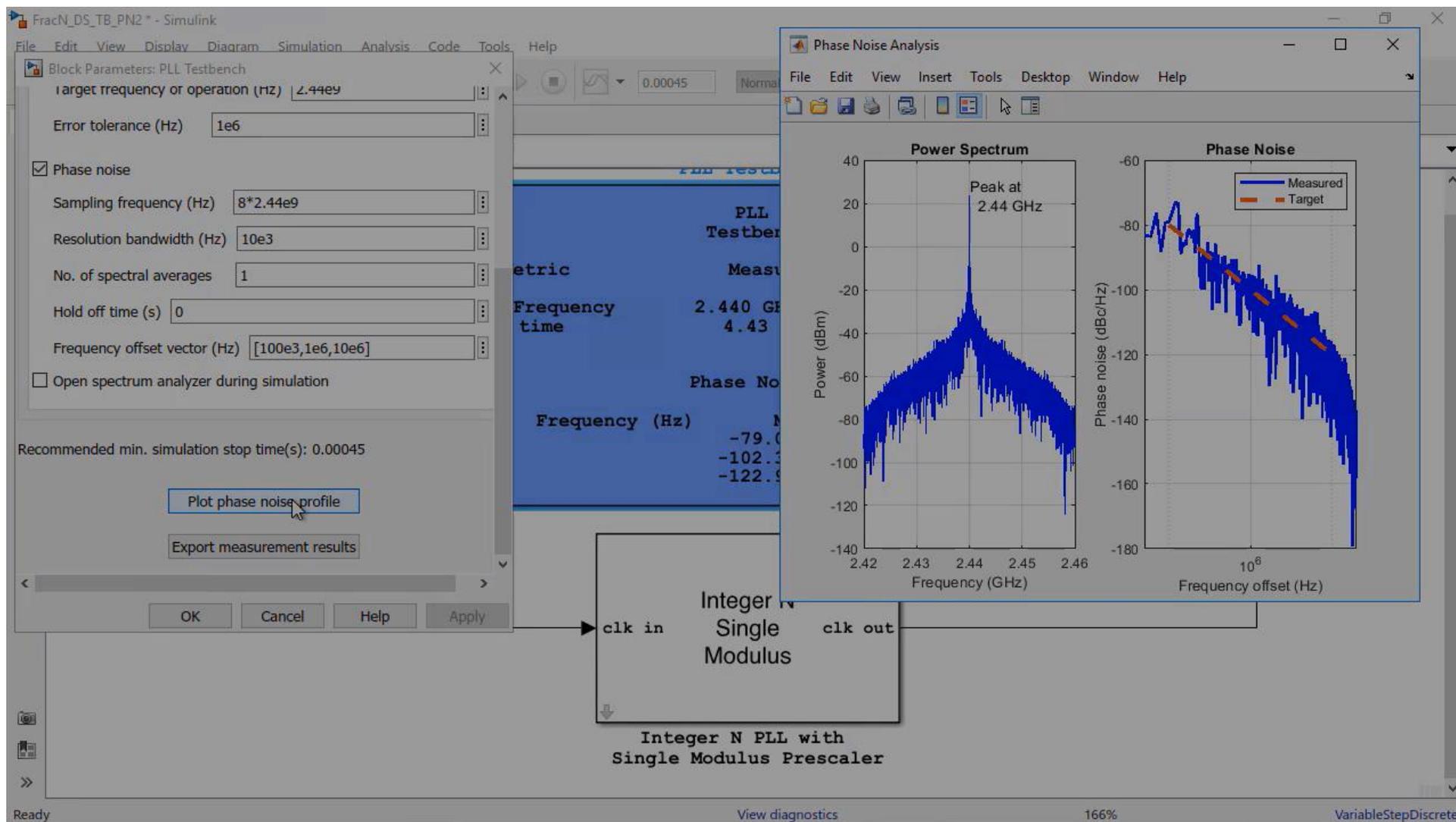


验证晶体管级/门级设计

- 在完整的系统仿真环境中测试 IP
- 利用Simulink和MATLAB的可视化和分析能力
- 独立测试每个模块
- 验证IP行为模型，加速系统级仿真

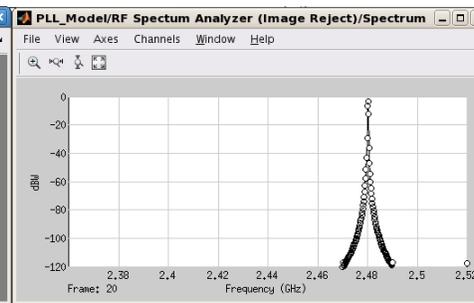
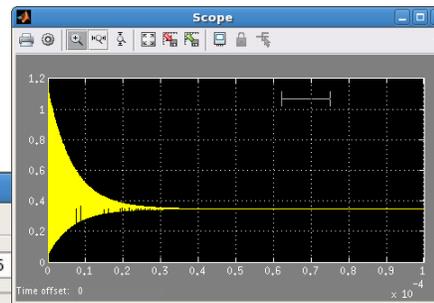
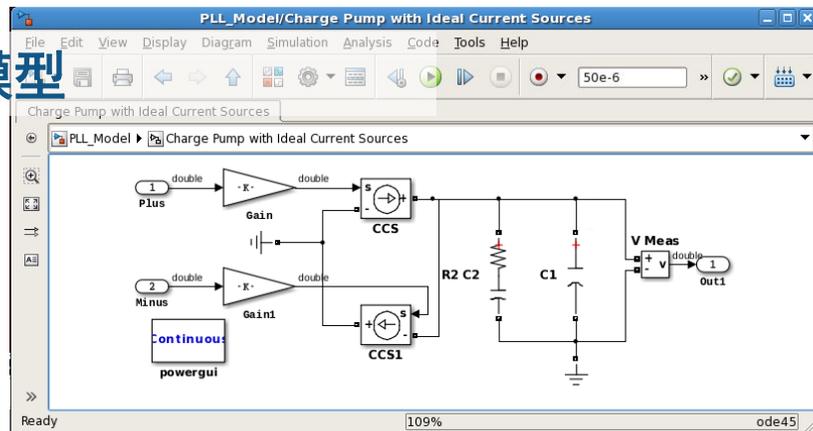


验证电路设计: 模拟域联合仿真

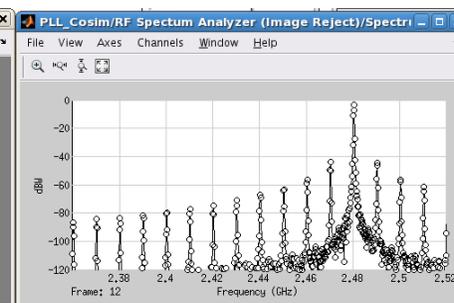
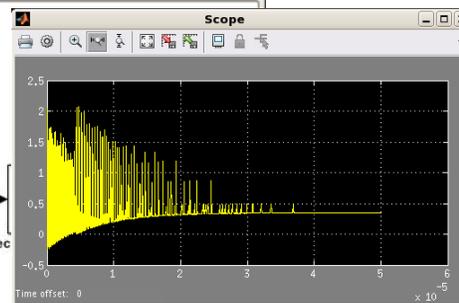
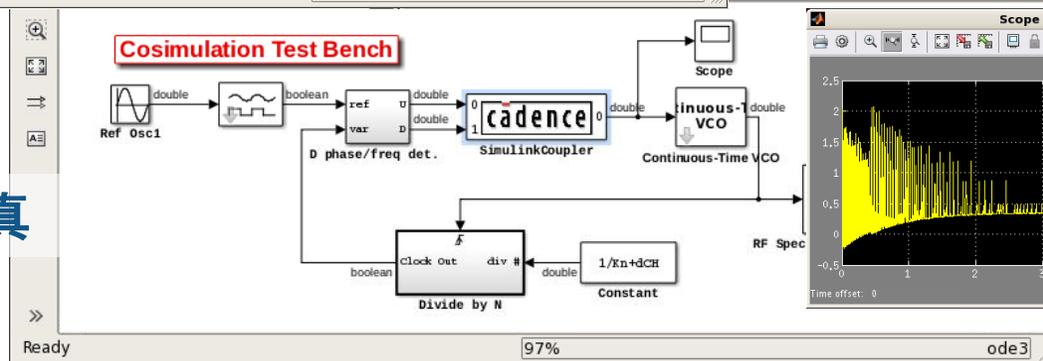


联合仿真模型验证流程

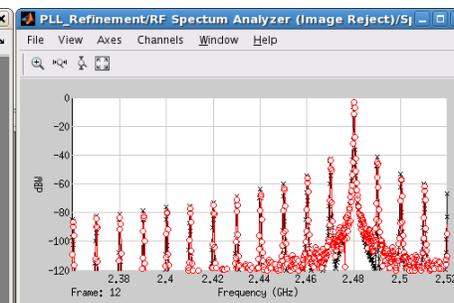
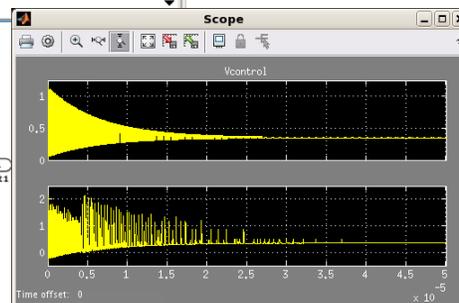
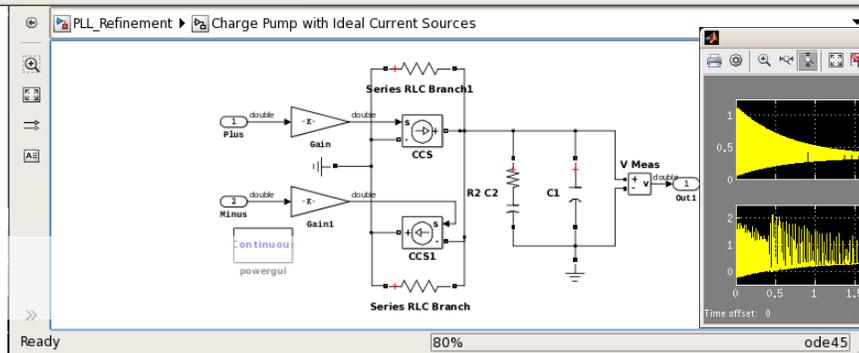
理想行为模型



联合仿真

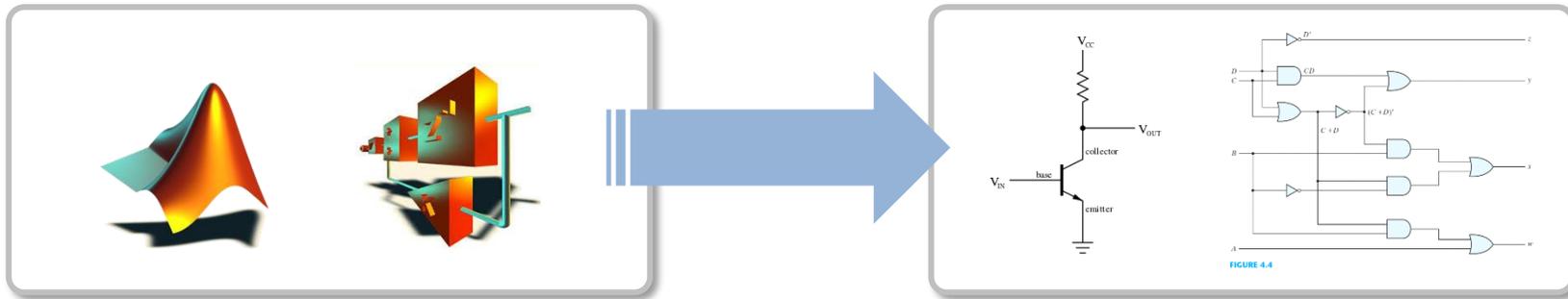


细化模型





方法二: 代码生成



代码生成的两种方法

HDL 代码

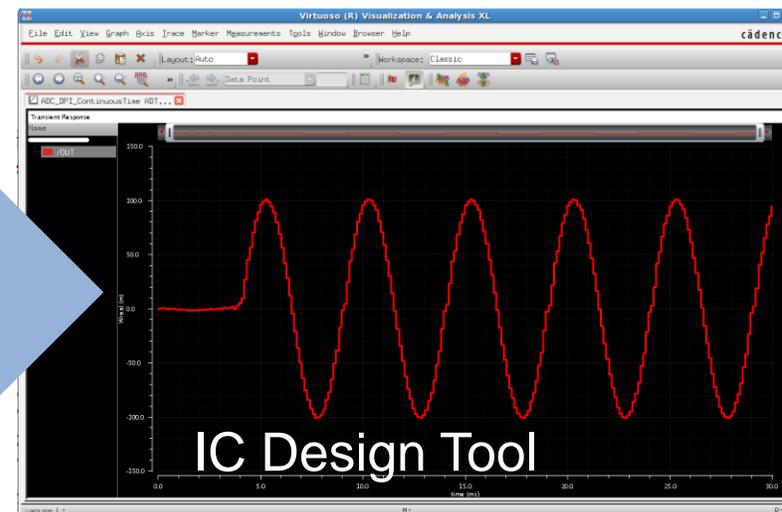
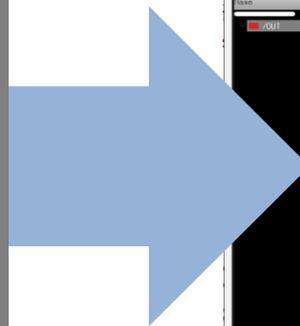
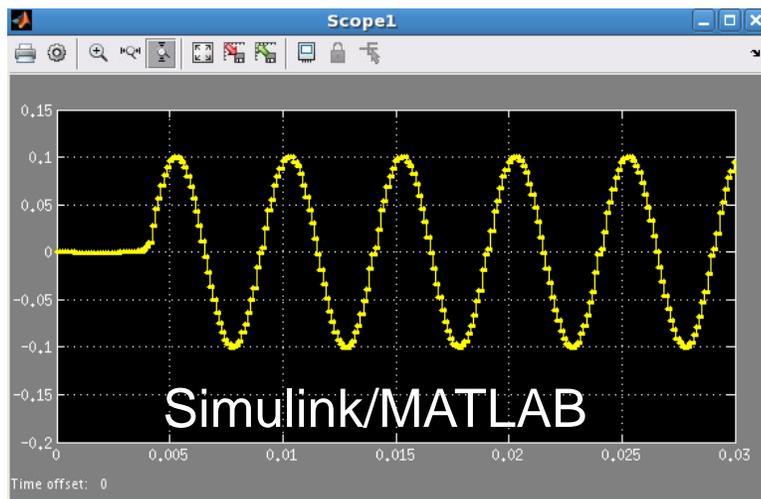
- 可综合的HDL代码 (Verilog, VHDL)
- Bit 精确的模型
- HDL Coder支持
- MATLAB 和 Simulink 环境都支持
- 适合原型设计和产品设计

行为代码 (SystemVerilog)

- C 代码进行SystemVerilog接口封装
- 实数模型
- HDL Verifier支持
- MATLAB 和 Simulink环境都支持
- 固定步长仿真

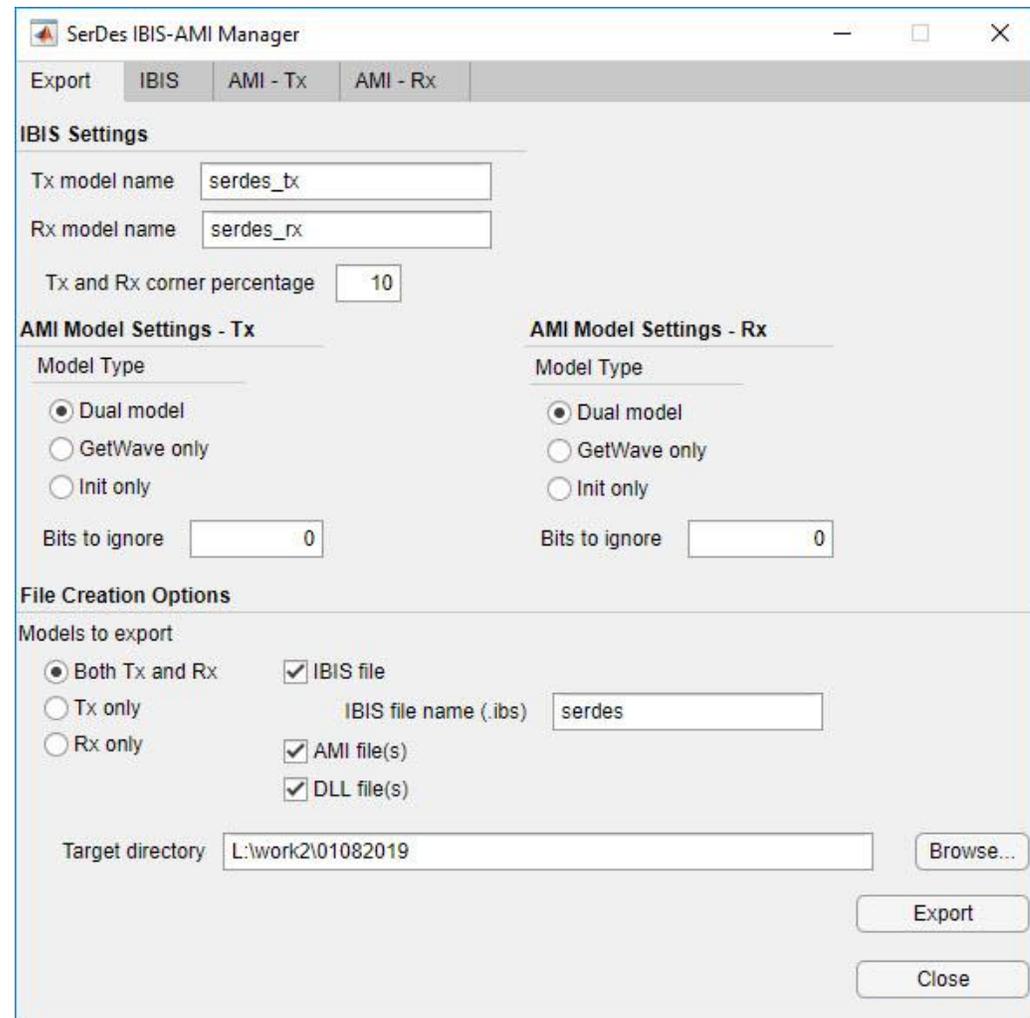
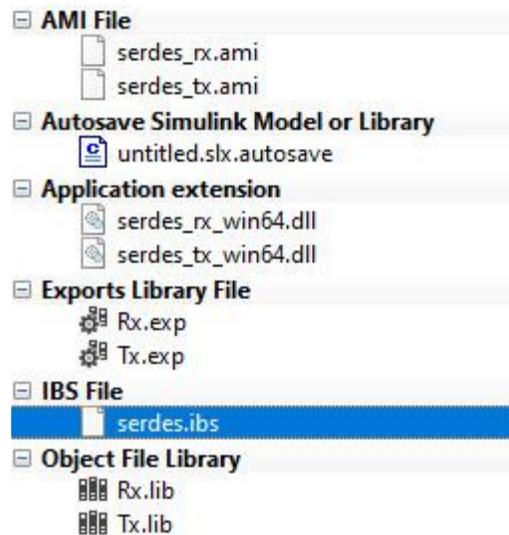
C 代码生成和 DPI-C 导出

- SystemVerilog API 实现快速仿真
- 不依赖于特定的 IC 设计工具
- 成熟的 C 代码生成技术可以提供实数模型
- 非常适合 Testbench 生成和 IC 验证
- 支持离散和连续时间信号
- 可从 MATLAB 和 Simulink 生成代码

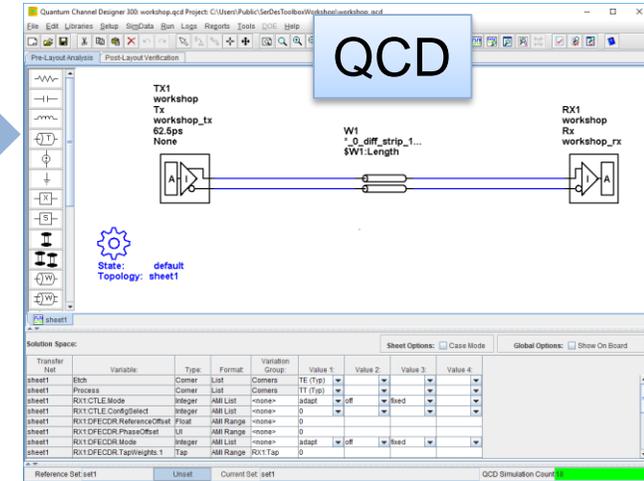
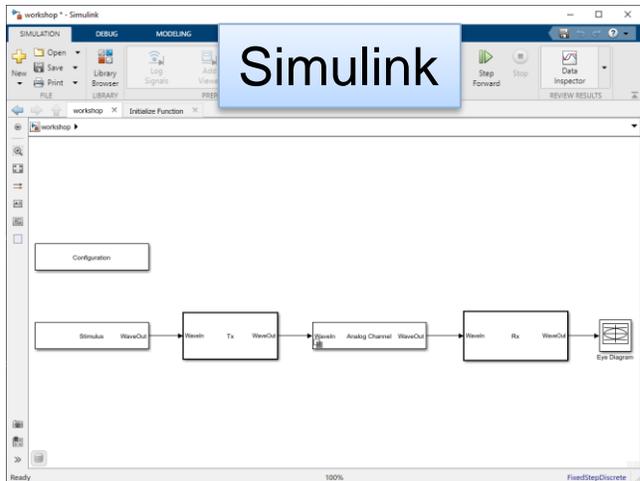
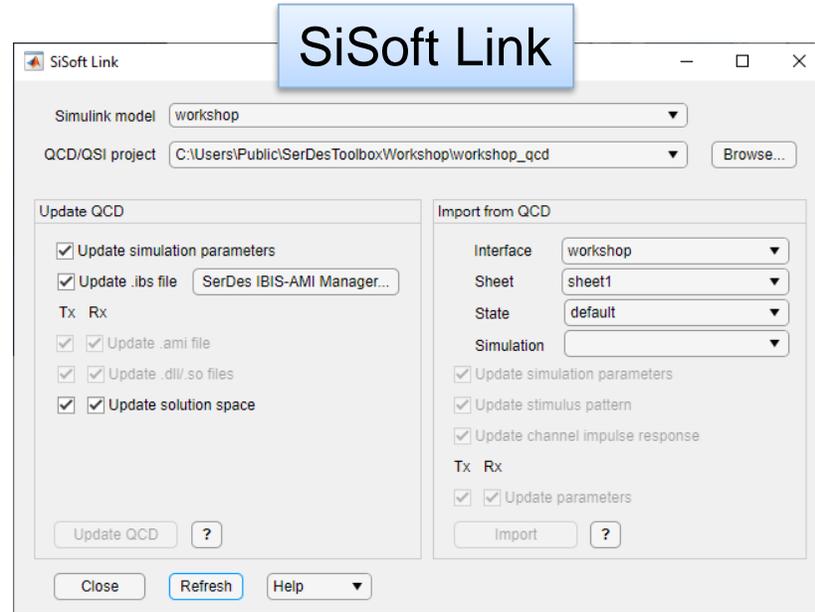


SerDes设计信道仿真器的集成：IBIS-AMI模型

- 生成Init 和 GetWave IBIS-AMI 模型
- 生成模拟 IBIS 模型
- 与信道仿真器集成：QCD/QSI (SiSoft Link)



SerDes设计信道仿真器的集成：IBIS-AMI模型



IBIS-AMI model files
UI, Samples/bit, Target BER

Stimulus, Channel impulse response
UI, Samples/bit, Target BER

内容

- 数模混合设计与验证的挑战
- MATLAB和Simulink用于数模混合电路开发
 - 利用Mixed-Signal Blockset开发ADC/DAC及PLL
 - 用SerDes Toolbox开发SerDes系统并生成IBIS-AMI模型
- MATLAB和Simulink与其他EDA工具集成
- 总结及如何获取更多相关信息

用MATLAB、Simulink、Mixed-Signal Blockset、SerDes Toolbox进行混合信号系统的设计与验证

- 使用MATLAB、Simulink提供的函数、模块开发算法，进行系统仿真
- 与EDA工具链接，建模并注入损伤，并重用Testbench
 - 联合仿真，HDL/SV 代码生成，数据后处理
- 与三方信道仿真器集成，用于SerDes验证
 - 生成 IBIS-AMI 模型
- 向客户发布 IBIS-AMI 模型和 DPI-C模型
 - 和客户交换模型，保护知识产权

MathWorks 混合信号解决方案

- <https://www.mathworks.com/products/mixed-signal.html>
- <https://www.mathworks.com/products/serdes.html>
- <https://www.mathworks.com/solutions/mixed-signal-systems.html>

Mixed-Signal Blockset
Design and simulate analog and mixed-signal systems

SerDes Toolbox
Design SerDes systems and generate IBIS-AMI models for high-speed digital interconnects

MATLAB and Simulink for Mixed-Signal Systems
Analyze, design, and verify analog and mixed-signal systems

MATLAB EXPO 2021

感谢参会！

