Automated correct-by-construct methodology for RTL design and analog mixed-signal test bench generation

Enables early design closure of mixed-signal SoC

Lakshmanan B, Murugesh Prashanth S, Atul R Lele, Ranjit K Dash
Texas Instruments (India) Pvt. Ltd.

accellera SYSTEMS INITIATIVE

Texas Instruments

2014 DESIGN AND VERIFICATION DVCON CONFERENCE AND EXHIBITION INDIA

Acknowledgment

- Keith Kunz, Neeraj Saxena & Padmini S of Texas Instruments (TI) for their continuous support and motivation
- Ravi C V & Harikrishna P of TI for their support in interception of these methods in several product executions
- Devendra S, Sonal R Sarthi, Rajesh Kedia & Cletan Sequeira of TI for their support in implementing the RTL generation aspects

Objective

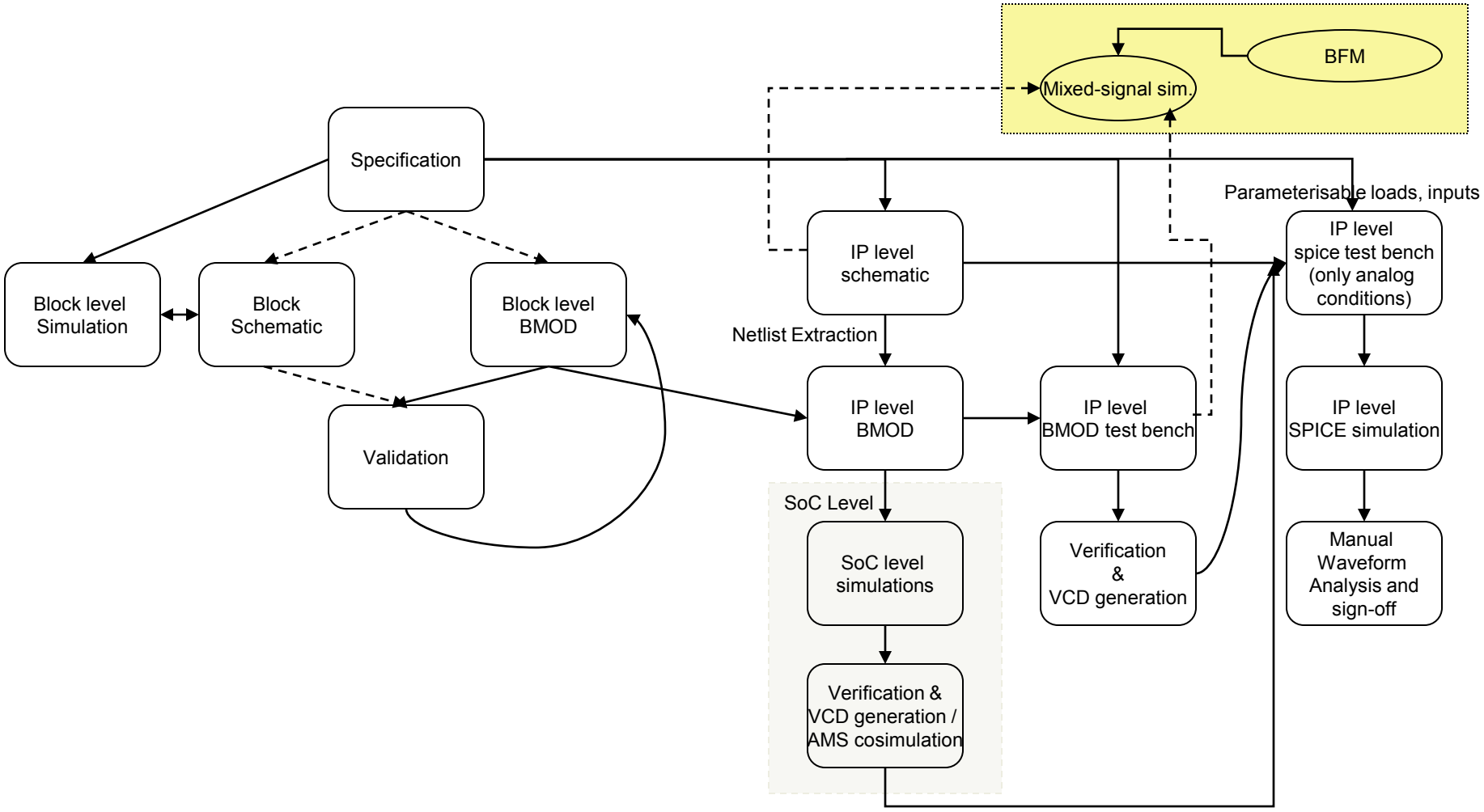
- Context
  - Analog IP development & verification
  - AMS SoC integration & verification
- Executable specification driven methods for
  - Automated analog test bench generation
  - RTL & ABMOD stub generation

Motivation: Analog mixed signal

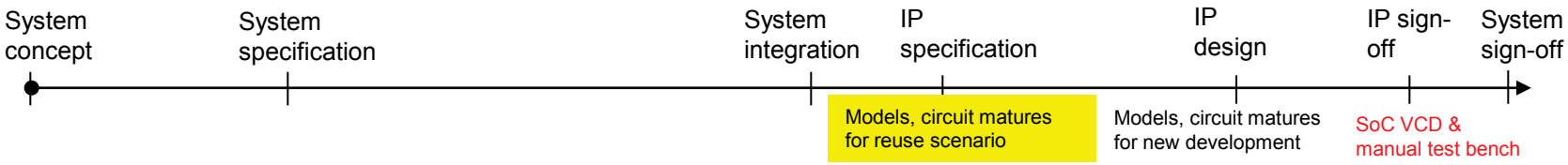
- Increasing analog integration
- Increasing system complexity, flexibility and richness of functionalities
- Time-to-market pressures & first pass silicon success
- Finding issues later in the product or design cycle is expensive

Quality product on time in the hands of end user

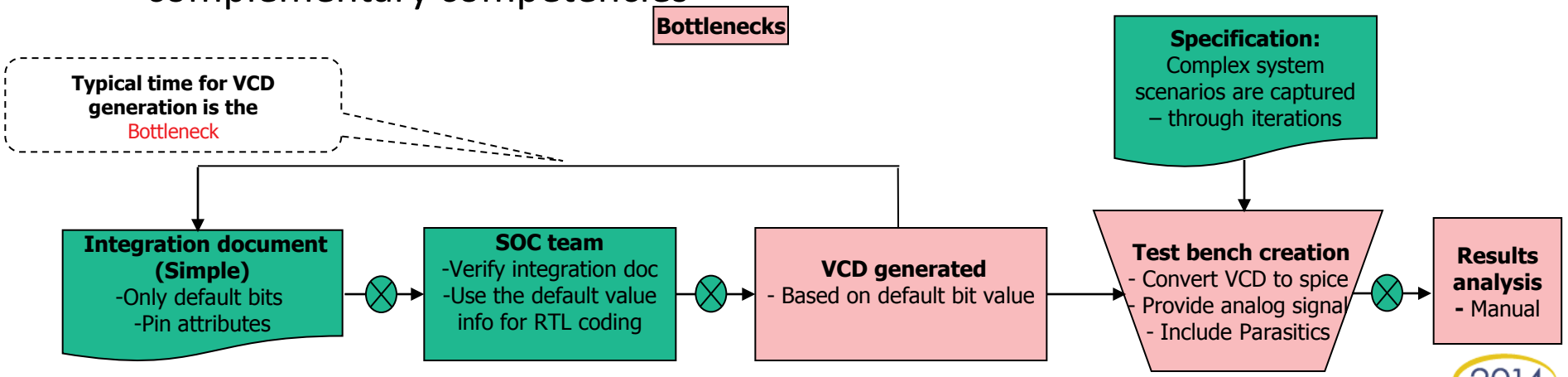
Conventional IP verification flow



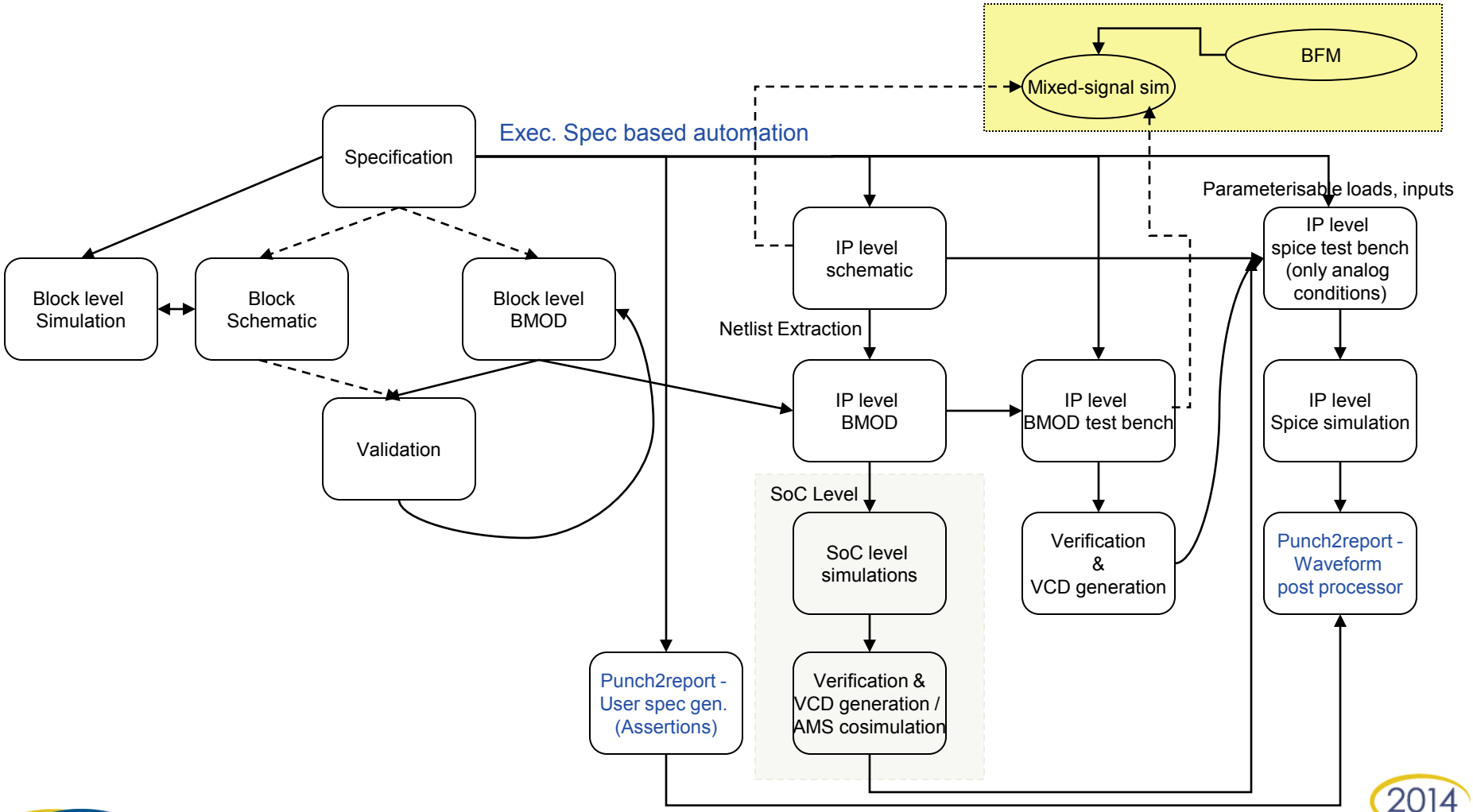
Verification bottlenecks



- IP sign-off & ABMOD validation test benches
  - Manual, duplicate and error prone
  - Review is much later in design cycle
- AMS at SoC level is very late in design cycle
- Communication bottleneck and complexity between different teams of complementary competencies



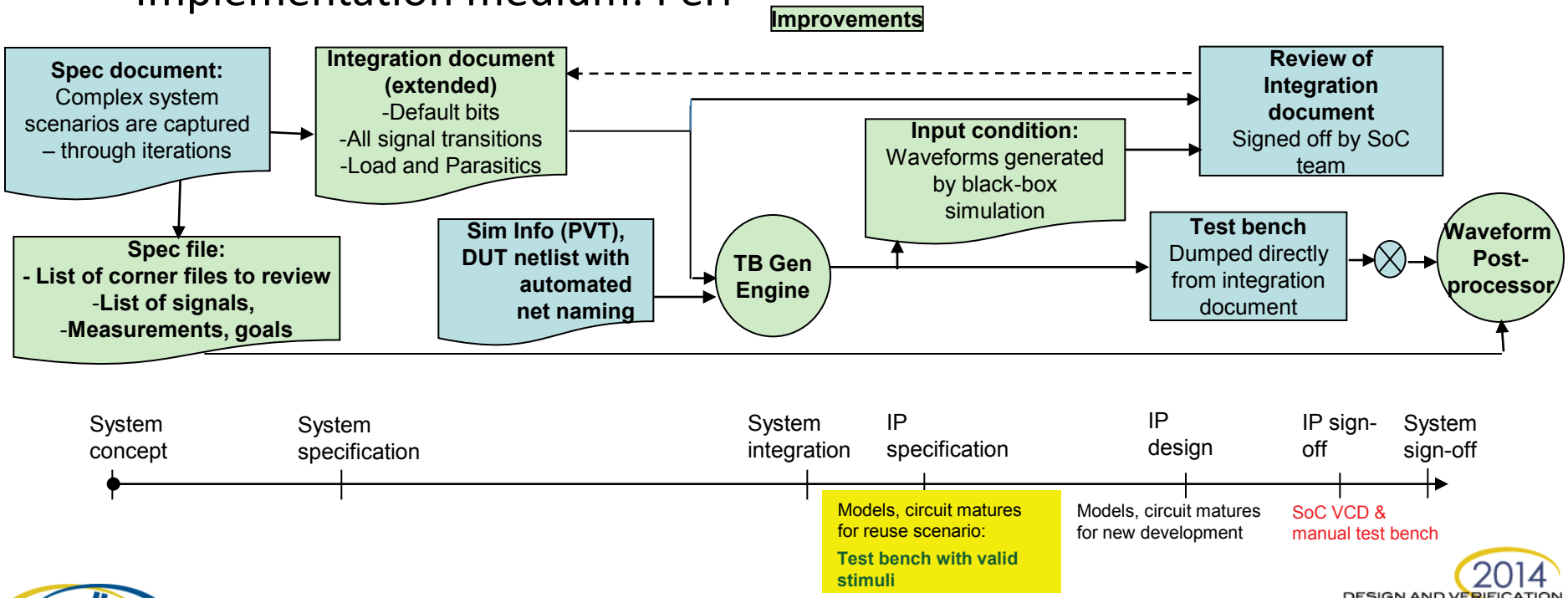
Improved IP verification flow proposal



Integration specification driven flow

Test bench generation

- Improved infrastructure for efficient and timely communication between different teams of complementary competencies
- Complements & supports existing input methods: A separate VCD & SPICE stimuli
- Implementation medium: Perl

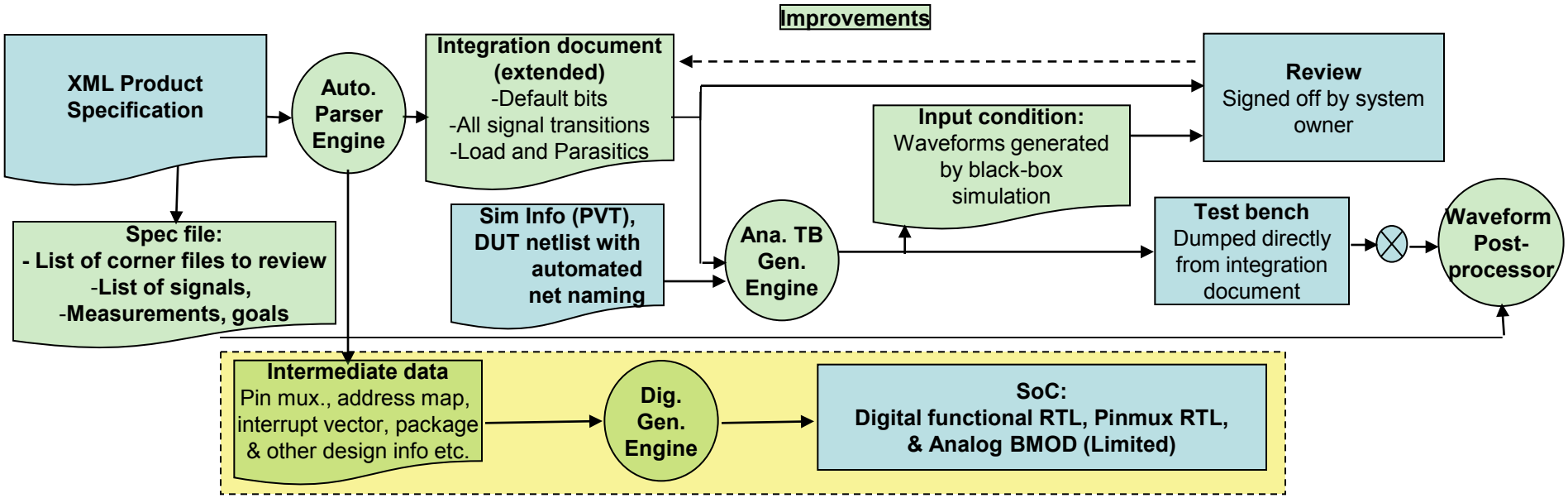


RTL & ABMOD stub generation

- RTL stub generation, system integration & verification
  - Pin multiplexing information for the interfaces & general purpose IOs
    - Truth tables
  - Translation of address mapping of registers
  - Interrupt & DMA trigger connections
- Verification
  - IFV based connectivity checks & assertions generation
  - Input boundary condition assertions for ABMOB
    - Reset/default states of the input control signals
    - Valid voltage/current levels
- Implementation medium: DocZone® XML, Xerces® XML parser, Magillum®, Atrenta® GenSys & Perl

Extension to RTL & ABMOD stub generation

- Specifications are developed in mark-up language (XML, JSON, etc.) ➔ XML parser



Flow: Inputs

- Integration document (csv format)
  - Generated from data sheet (XML or JSON, etc.) and augmented
- Top level spice netlist (spice netlist)
  - DUT instantiated
  - Name of NET connected to each pin should be same as the pin name with “fixed prefix”
- Simulation Information file (perl file)

Flow: ISD & its contents

Sub-Block	Ball	Port	Direction	Register	Bits	Description	Default Bits	Power Up

Efuse	Voltage domain	Min	Nom	Max	Associated Ground	Current	IR Drop	Location	Source	Test Scan (Any mode)	Test Scan (Test mode / IDDQ mode)	Input/ Output Capacitance (Please re of Relevent cells below)

Leakage on Power Ports (uA)	Analog/Digital/clock (A/D/C)	Power/Ground (P/G)	Sync/Async (S/A)	Pad/Port (P/S)	Control (H/L/S)	PER_TECH (WIRE/CMOS)	OUTPUT_CLASS _PER	MIN_ROUTE_SP ACING	MIN_ROUTE_WI DTH	Active Power (A)	LDO shutdown condition LDO outputs)

Block level interface		Remarks

Microsoft Excel Worksheet

Newly added fields

Feedback point	PBIT [dc voltage, value for 0 bit, value for 1 bit, 0-1 time delay(s), 0-1 rise time(s), 1-0 time delay(s), 1-0 fall time(s), bit transition time(s), delay time(s), periodic / non-periodic], bit value pattern(1/0)	Routing Capacitor	Routing resistor	Routing Inductor	Load Capacitor	Cap ESR	Load Inductor	Ind ESR	Load Resistor

Flow: Simulation Information file

```
$voltage_corner="typ";                    # typ, min or max
$temperature=27;
$process="nominal";                       #nominal, weak, strong, strong_lkg, skewnp or skewpn
$instance_name="I0";                      #NET name prefix used in spice file
$netlist_path="/sim/DUT/TESTBENCH_AUTO/tb_DUT.spi";
$model_path="/db/pdk/tech_node/current/models/model.paths.scs";
# input for transient simulation
$TSTEP="10e-9";
$TSTOP="5e-3";
$TPUNCH="100e-6,400e-6,600e-6,3.4e-3,3.7e-3,4.0e-3";
$int_doc_file="/sim/DUT/TESTBENCH_AUTO/dut_pin_report_1p0.csv";
$tb_file_name="atb_dut.spi";
# Optional (Default=No)
$auto_file_name="yes";
# Optional (Default=PWD)
$run_path="/sim/DUT/TESTBENCH_AUTO";
# Optional (Default=0)
$debug=1;
$debug_log="debug.log";
```

Flow: Waveform post processing

- Allows user to specify
  - List of waveform files
    - Preferably corner simulation results of the same design where the expected results are comparable or similar
  - List of signals to be processed in all the punch files
  - The processing and measurements to be done on the specified waveforms
  - Expected range of measurement for automatic check
  - Format of the table in which the report to be consolidated
- Generates
  - An HTML report in the user specified format
  - A consolidated waveform file with all the signals that the user processed

Flow: Waveform post processing

User Interface

Punch2report_pc.pl [-help] [<punch_list_file> <signal_list_file> <report_name>]

- **-help** Provides usage information on command line
- **<signal_list_file>** The file that specifies the signals to be processed, type of processing / measurement, intended goals for comparison and highlighting in the report and an inherent report content formatting.
- Comment:
  *
- User format for table generation:
  { NEW_TABLE <argument_list>}
  {NEW_TABLE <title_as_quoted_string> <comments> <column_heading_1> … <column_heading_2>}
- Signal processing:
  <signal_name> [time] [{command [args]}]
  <signal_name> {<measurement_operation_1> [<arguments>] [<signal_name] [<goal>]} …
  [{<measurement_operation_n> [<arguments>] [<signal_name] [<goal>]}]
- *Waveform measurements & Commands supported*:
  GETX, GETY, AVG, RMS, FREQ, DELAYXX, DIP, PEAK, SET_THRES, SET_GP, SET_TOL, TRANS, TRANSITION_TIME, RISE_FALL_TIME, NEW_TABLE
- Macros:
  Combination of the above can be defined as a user defined macro

Example: Waveform post processing

** Defines a new report table by name "Power-up Sequence" with one column*

**{NEW_TABLE "Power-up Sequence
" "
" "Delay" }*

** Measuring delay between VI0_EN_1P8V & VXIO.GEN_1P8V*

**VXI0.GEN_1P8V {DELAYXX 1 1.6 VI0_EN_1P8V 1 1.3 >500e-6&<540e-6}*

** Defines a new report table by name "Power-up Measurements" with 7 columns*

**{NEW_TABLE "Power-up Measurements
" "
" "Value before PORZ
(V or A)" "Value after PORZ
(V or A)" "N_{TR}" "T_{TR}
(Secs)" "T_{R/F}
(Secs)" "Dip
(V or A)" "Peak
(V or A)"}*

VI0_EN {GETY 0 <10e-3} {GETY 4.5e-3 >=1.4}

VXI0.VDD_INT_1P2V {GETY 0 <10e-3} {GETY 4.5e-3 >0.9&<1.4} {dip 110e-6 5e-3 >0.8&<1.2} {peak 15e-6 5e-3 <1.3}

VI0_VDD_AD_1P8V {GETY 0 <10e-3} {GETY 4.5e-3 >=1.6}

VI0_VDD_CORE_1P2V {GETY 0 <10e-3} {GETY 4.5e-3 >=0.8} {dip 4.1e-3 5e-3 } {peak 4e-3 5e-3}

VI0_VDD_SRAM_1P2V {GETY 0 <10e-3} {GETY 4.5e-3 >=0.8} {dip 4.2e-3 5e-3 } {peak 4e-3 5e-3}

VI0_PORZ_CORE_1P2V {GETY 4.0e-3 <10e-3} {GETY 4.5e-3 >=0.8}

END -1

Summary

- Automated, error free test bench & design model generation mechanism
- Input format definition: An integration spec document format is defined to suit
  - Machine readabile & Human readable for ease of review
  - Inputs with pin attributes, simulation requirements as per specification
  - Stimuli format – embedded in the integration document
  - Enable efficient review by waveforms generated for each scenario
  - Improved infrastructure for efficient and timely communication between different teams of complementary competencies
  - Test bench components like load, parasitic information
  - Push button digital RTL and analog BMOD generation
- Engine:
  - Perl based engine to process the ISD and generate test bench, documentation (complex waveforms) for review
  - Waveform post processor: Signal measurement information & Goal checking
    - Generates HTML based report highlighting failure of goals
- Extension:
  - Generation of components for database hand-off process for SoC integration
  - Potential to extend it for a push-button sub-system level simulation and validation flow
- New process flow to improve quality, efficient and early verification

Future scope

- IPXACT compatibility & interoperability
- Analog test bench generation
  - Wider analog functional portfolio
  - Currently simple power management functions supported
- Verification of SoC memory map, connections such as DMA triggers and interrupts
- Supporting synthesis/STA timing constraints generation
- Analog assertions generation

Conclusions

- Bottlenecks exist in analog IP verification sign-off and closure
- Affects SoC maturity and time-to-market
- Automated, error free test bench & design model generation mechanism (input format & script based engine)
- Specification (XML, widely used format) to verification sign-off
- Potential to extend it for a push-button sub-system level simulation and validation flow

Questions

Thanks for your interest, attention, and time

accellera SYSTEMS INITIATIVE

2014 DESIGN AND VERIFICATION DVCON CONFERENCE AND EXHIBITION INDIA