Who takes the driver seat for ISO 26262 and DO 254 verification?

Reconciling requirement based verification with coverage-driven verification

Avidan Efody, Mentor Graphics Corp.
Motivation

- ISO and DO users are often hesitant about advanced verification techniques
  - Results in lesser productivity
  - Is that grounded?
  - If yes, what can be done to help?
Agenda

• Intro
  – CD/CR
    • Context
    • Origins
  – RBV
    • Context
    • Origins

• Areas for consideration
  – reduced verification scope
  – Result stability
  – Coverage tracking gap

• Summary
CD/CR verification

• Context
  – Blocks getting too big to be verified by directed tests
  – Interactions between “independent” blocks not tested
  – Tailored for “RTL block verification” problems

• Added value
  – Open degrees of freedom to stimuli
  – Allow unexpected bugs to be found
  – Improved and more reliable metrics
RBV

• Context
  – Safety critical systems in mil-aero/medical/automotive
  – Need to make sure spec and docs 100% match deliverables
  – “RTL block verification” is just a tiny concern

• Added value
  – Full vertical/horizontal traceability
  – Requirement change cost can be estimated
  – Ownership, responsibility, liability clearly defined
  – Any value at RTL design and verification level?
RBV and CD/CR

- It's complicated...
  - Management wants control
  - Engineers are after efficiency
  - Where are the trade-offs?
Agenda

• Intro
  – CD/CR
    • Context
    • Origins
  – RBV
    • Context
    • Origins

• Areas for consideration
  – reduced verification scope
  – Result stability
  – Coverage tracking gap

• Summary
reduced verification scope

• Low level requirement scope is fairly limited
  – Determined by management/tracking convenience
  – Not by design/verification concerns

• Is that “micro-management” from a CD/CR testbench perspective?
  – Does CD/CR require a wider scope?
RBV vs. CD/CR process(1)

**CD/CR**
- Ideal process
  - Analyze spec holistically
  - Find commonalities, abstract
  - Implement
- Confidence derived from models overlapping

**RBV**
- Ideal process
  - Break spec into requirements
  - Implement
- Confidence derived from 100% requirements mapped
RBV vs. CD/CR process (2)

- Management tends to track number of requirements implemented
  - Hard to track CD/CR this way

![Diagram showing RBV vs. CD/CR process](image-url)
Example (1)

• A DMA controller with configurable AMBA interface
  – i.e. AXI3, AXI4

• CD/CR process:
  – Read AXI3/AXI4 spec carefully
  – Try to find all common points
  – Single generator, Single checker
  – <10 tests
  – Coverage using crosses

• Very similar to design
Example (2)

- Management asks difficult questions:
  - Why does the stimuli generation takes so long?
    - “Well, I’m reading the spec holistically…”
  - How many of the requirements did you already do?
    - “Well, can’t say, my test generates all stimuli together…”
  - A requirement has changed, what do we have to rerun?
    - “mmm…”
Solutions (1)

- “Cover” whatever you do
  - Even if initially it is all 0
- Advanced coverage databases answer:
  - % of requirements
  - Tests that need to be rerun
  - Regression efficiency, trends, etc
Solutions (2)

- This will only partially address the problem
  - Link from requirements to results will be there
  - But the results will be at 0 for relatively long

- Team experience and verbal skill often determine
  - Need to align management on expected results
  - Need to be able to explain why
Result stability

• ISO/DO demand that results are repeatable and reproducible
  – Allow easier auditing, configuration management, liability

• Is that problematic with a CD/CR verification environment?
  – Intuitively, some will say yes, is that intuition true?
CR result stability basics

• @ same code/seed/simulator results are stable
  – Very similar to non-random + seed
• Code modification can changes results
  – Can be true for non-random as well
• But, changes will be bigger with random
  – Invalidate list of tests/seeds to some extent
  – Reduce coverage
Introducing: random stability

• In a random stable environment code modifications impacts are:
  – Minimal
  – Local
  – Easily traced back to code modification
Result stability

Solutions (1)

• Make your environment random stable
  – Use UVM (version > 1.1d)
    • Provides a robust infrastructure
  – Follow specific coding guidelines
    • Most make sense anyhow
    • Can be found here
Solutions (2)

- Monitor coverage continuously
  - Minor changes can drop coverage numbers
  - Make sure you catch those early
  - Some coverage databases provide trend-analysis
Solutions (3)

- Consider advanced stimuli methods
  - Graph based methods combine
    - The stability of directed testing
    - The coverage space of random
    - And make it all more efficient
  - Standardization ongoing at Accellera PSWG
Functional coverage tracking gap

• ISO/DO require link to “verification results”
  – Could be functional coverage, except:
    • Bug in coverage?
    • Bug in tool?
  – Auditing might require something less processed
    • Wave files
    • Log files

• How can you go from functional coverage to raw results?
SV coverage types

- Cover groups
  - Good for data and crosses
  - But tied to single event

- Assertions
  - Good for event sequences
  - But data is very simple

- We want both!
  - Complex data & crosses
  - Sequences of events
  - Go from coverage to multiple points of time/data
Example

• Read request to secure zone gets OK response with garbage data
  – Two events: request & response
  – Data: address, request type, response, data

• Cover group implementation
  – Sampled at response time
  – Request time not captured
  – Response time capture?

• How do you go from cover group to wave?
Solutions(1)

• Home brewed
  – Print message to log with unique ID @ response, request
  – Print message to log before sampling covergroup
  – Match based on time/IDs
  – Lots of work with coverage report, log, wave
Functional coverage tracking gap

Solutions(2)

- UVM transaction recording
  - Allows linking from transaction to signals
  - Coverage->transaction
  - Transaction->wave/log
Solutions (3)

• Dynamic coverage
  – Post processing
  – Multiple events/data

• Requirements map to database queries
  – Get all data and events
  – No rerun @ requirement change
Summary

• ISO/DO teams should stay on top of advanced verification
  – Advanced verification is not a luxury
    • Matches design sizes and complexity
    • Can’t be competitive otherwise
  – Gaps are easily created
    • And are very hard to close afterwards
Summary

• RBV and CD/CR can work together
  – But require upfront planning
  – Awareness to specifics
  – rigorous emphasis on coverage

• With good planning
  – Productivity can go >> directed tests
  – And ISO/DO support can improve
Why Requirements Driven Verification?

- **Metric Driven Verification**
  - Allows us to define targets
  - And monitor progress
  The metrics can become the end rather than the means to the end

- **Coverage Driven Verification**
  - Most common metric driven verification approach
  - Code Coverage
  - Functional coverage
    - Might be related to features
  How often have you chased a coverage goal with limited ROI?

- **Feature Driven Verification**
  - Features MIGHT be related to spec
    - Is that relationship captured?
  - Are features related to requirements?

Shouldn’t everything we do be related to a requirement?
CD means CR

• CD flow:
  – Run tests
  – Look for coverage holes
  – Direct tests to those

• If we know what tests are doing, need for coverage is much smaller
  – And we can call that feature driven/requirement driven
Guidelines (1)

• Please keep the default font size for main lines at 28pt (or 26pt)
  – And use 24pt (or 22pt) font size for the sub bullets
• Use the default bullet style and color scheme supplied by this template
• Limited the number of bullets per page.
• Use keywords, not full sentences
• Please do not overlay Accellera or DVCon logo’s
• Check the page numbering
Guidelines (2)

• Your company name and/or logo are only allowed to appear on the title page.
• Minimize the use of product trademarks
• Page setup should follow on-screen-show (4:3)
• Do not use recurring text in headers and/or footers
• Do not use any sound effects
• Disable dynamic slide transitions
• Limit use of animations (not available in PDF export)
Guidelines (3)

• Use clip-art only if it helps to state the point more effectively (no generic clip-art)
• Use contrasting brightness levels, e.g., light-on-dark or dark-on-light. Keep the background color white
• Avoid red text or red lines
• Use the MS equation editor or MathType to embed formulas
• Embed pictures in vector format (e.g. Enhanced or Window Metafile format)
Questions

Finalize slide set with questions slide