Pre-silicon digital verification sign-off at Infineon ATV MC

Tim Blackmore
24th November 2020

Context - Process

Product Verification Manager for Aurix3G family of microcontrollers
- Definition of verification processes
  - Executed by large number of people – need consistency
  - Aurix3G used in safety critical automotive applications including ADAS, drive-by-wire, airbag and powertrain
    - Sign-off must ensure compatibility with ISO26262 Standard

- In this context verification encompasses
  - Pre-silicon verification including digital, analogue, AMS, power-aware, timing, manufacturability, …
  - Post-silicon verification including validation, characterisation, reliability, test and safety analysis
- Focus on pre-silicon digital IP functional verification
  - Executed by around 100 teams based in seven sites across two continents

It's all in the planning ...

Product Verification Plan (PVP)

Our goals can only be reached through the vehicle of a plan, in which we must fervently believe, and upon which we must vigorously act. There is no other route to success — Pablo Picasso

Product Verification Plan (PVP) sets the Verification **Objectives**, describes the Verification **Methods** that can be used and stipulates the Method-dependent **Targets** that show that the Objectives have been achieved
Ensures *consistency* of execution and sign-off

Objectives vs. Targets
Objectives can be quite abstract and written without detailed knowledge of how they will be achieved
Targets are an expert interpretation of Objectives and must be determinable – unambiguous PASS or FAIL

Objectives

Methods

Targets

PASS   FAIL
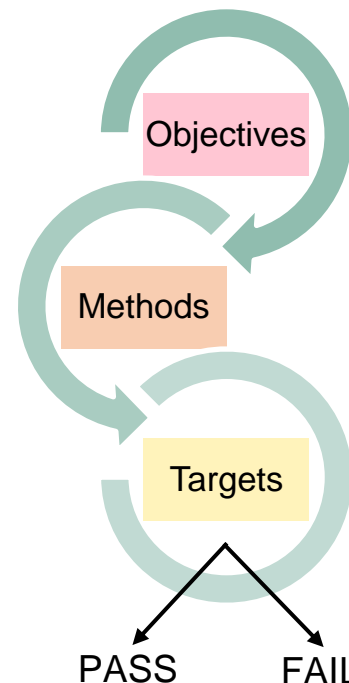
Showing that **Requirements** and **Features** are satisfied by the implementation
- **Requirements** are *verifiable statements* written by concept engineers derived from (not necessarily verifiable) customer requirements
- **Features** are *verifiable statements* written by verification engineers
  - Features are derived according to a table of methods listed in the PVP inspired by similar tables in the ISO26262 standard
    - Deriving methods including 'analysis of boundary values', 'knowledge or experience based error guessing', …
    - Input taken from other stakeholders

Showing that all RTL code has been exercised

Showing that the DUV has been extensively soak tested

Increasingly
Fine Safety Nets

Methods and Targets

Example Verification Targets for **simulation-based verification**
- **Test cases** determined as PASS across variety of checkers (reference model, assertions, self-check)
- **Functional Coverage** determined as PASS if hit by passing test cases
- **Branch** in RTL code determined as PASS if taken by passing test cases
- **Soak testing** determined as PASS if targeted number of generated test cases all passed

Example Verification Targets for **formal verification**
- **Properties** determined to PASS if they hold non-vacuosly or if they have a bounded hold for a sufficient number of cycles
- **Statement** in RTL code determined to PASS if tool says it has been covered by property set
- **Exhaustive verification** is a PASS if property set is complete and all properties PASS
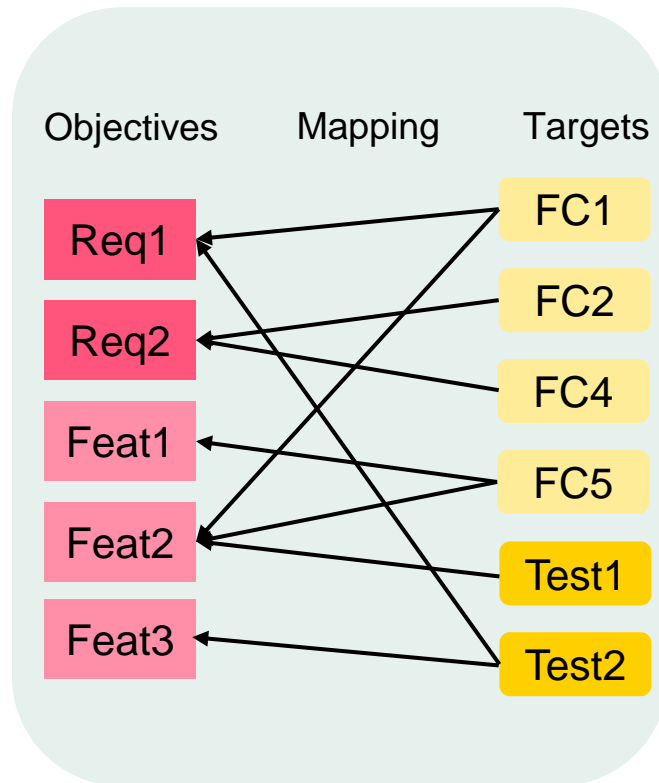
… and in the specifying …

Verification Specification

Plans are good but it's important to be specific
Each verification team writes a Verification Specification
- One PVP, 100+ Verification Specifications
- Requires Domain expertise

Each Verification Specification
- Includes specific Objectives for that team e.g. all relevant requirements and derived features
- Includes specific Targets for that team e.g. list of directed tests, functional coverage model
- Includes the mapping of Targets onto Objectives to ensure that all Objectives fully verified

Objectives   Mapping   Targets

Req1   Req2   Feat1   Feat2   Feat3

FC1   FC2   FC4   FC5   Test1   Test2

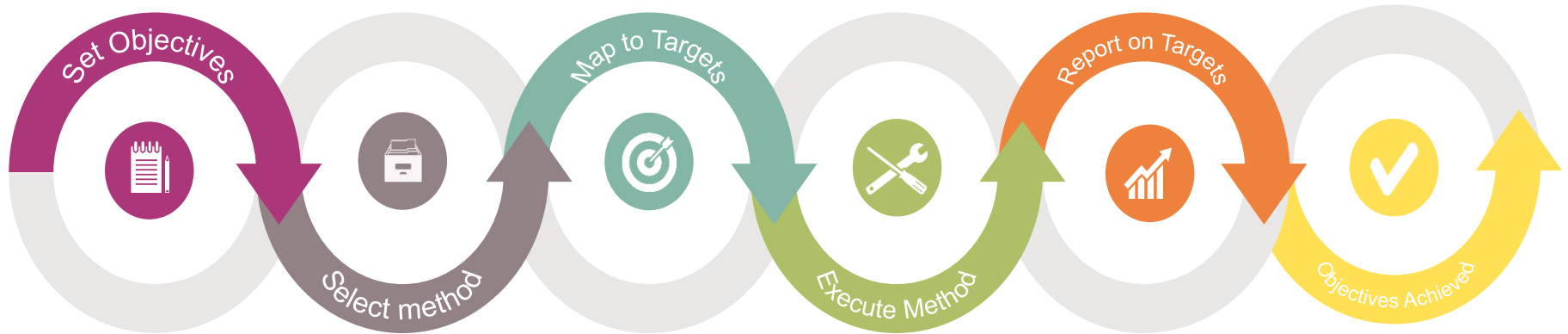.. and the rest is easy

Execute

Report all Targets PASS

Verification Environment Qualification

Qualification of verification environment important part of sign-off
- Based on
  - Review
  - Ability to detect mutations inserted in design

Summary

Questions?

Part of your life. Part of tomorrow.

Relating Objectives and Targets

- Verification of Requirements and Features satisfied by test cases, functional coverage and properties
- Exercising RTL code satsified by branch or statement coverage (and ...)
- Soak testing satisifed by targeted soak testing

- Targets change with Method but Objectives are the same
- Targets are tool dependent e.g. can only show that RTL code has been exercised if there is a tool to do it
- Objectives can be defined by non-experts but sufficiency of Verification Targets in satisfying Objectives is based on experience