Compliance driven Integrated circuit development based on ISO26262

Haridas Vilakathara

Manikantan panchapakesan

NXP Semiconductors, Bangalore

accellera SYSTEMS INITIATIVE

2014 DESIGN AND VERIFICATION DVCON CONFERENCE AND EXHIBITION INDIA

Outline

- Functional safety basic concepts (ISO26262 view)
  - IC failure and their causes
- Management of random faults
  - Safety architecture
  - Safety analysis
- Management of systematic faults
  - Product creation process
  - Compliance driven development process
  - Safety management after release to production

Functional safety basic concepts

- IEC 61508/ISO 26262 safety concept
  - It is impossible to develop a zero error (bug free) system
    - Specification, implementation or realization errors
    - Random hardware failure may occur due to
      - Reasonably foreseeable operational errors,
      - Reasonably foreseeable misuse.
  - It is is possible to build a system with acceptable failure rate
    - Acceptable failure rates vary per application
    - Classified by automotive safety integrity levels (ASIL) &
    - frequency and severity of functional failures
  - If a failure occurs, the system needs to be transferred into a safe state
    - Failure event should not lead to unacceptable risk
      - System must detect all faults that could lead to a hazardous event
      - The fault reaction time to achieve safe state must be short enough to avoid any hazardous event
    - Safe state can be defined as
      - Fail safe or fail operational
        - » Depends on application

IC failure and their causes

- Random failures
  - Due to
    - Random defects inherent to Semiconductor process
    - Environmental conditions
    - Usage & handling
- Systematic failures
  - Bugs introduced during, specification, design, development
  - n detected bugs during verification cycle
  - Due to not following best practices and methodologies for product development

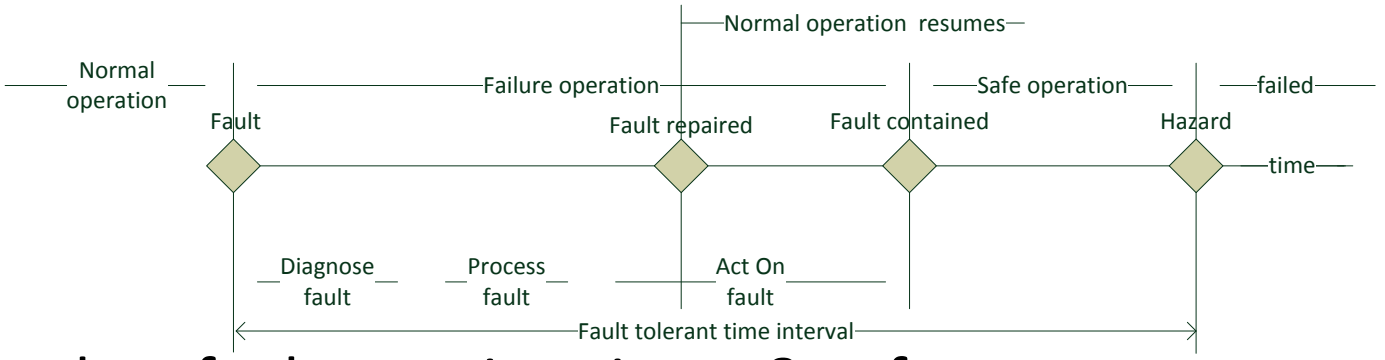
Safety architecture for detection and management of random failure (manage fault)

Compliance driven process development and continues improvement (avoid fault)

Outline

- Functional safety basic concepts (ISO26262 view)
  - IC failure and their causes
- Management of random faults
  - Safety architecture
  - Safety analysis
- Management of systematic faults
  - Product creation process
  - Compliance driven development process
  - Safety management after release to production

Functional safety architecture



- Based on fault reaction time & safety strategy
  - How much a system can tolerate the IC functional failure
  - Fail safe, fail operational etc.
    - Based on application profile
    - Type of IC (ASIC, MCU, SoC etc)
- Safety architecture to detect fault, diagnose fault and to act on fault within reasonable time

Application consideration

- IC Developed specifically for a customer (in context)
  - Clear specification from customer including safety requirements
    - Handle ISO26262 requirements through a development interface agreement (DIA)
- IC Developed for more than one applications (safety element out of context)
  - Assumptions are made on application profile
    - system level design, safety concept, and requirements based on the product application scenarios
  - Validate the assumptions through “proven in use” kind of arguments

Safety architecture (key requirements)

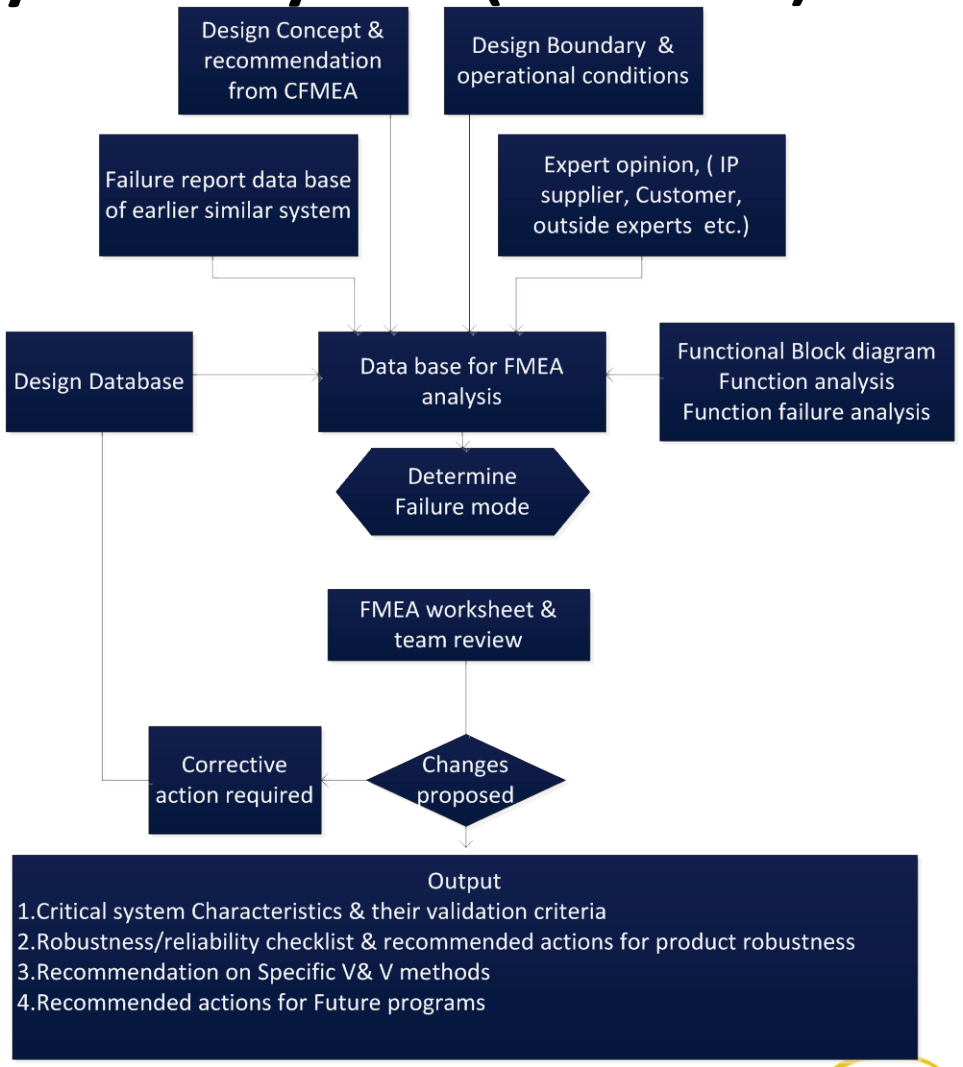
- To detect fault
  - Shall incorporate sufficient failure detection & diagnostic features
    - To detect environmental conditions
      - Temp sensor, voltage sensor etc.
    - To detect operational anomalies
      - Memory errors, watchdog features etc.
    - Safe island
      - Protected heavily to guarantee reliable operations under extreme conditions
        - » Provide diagnostic information back to system within FTTI
- Process fault
  - Logic to isolate fault and to assess the severity of fault
- Act on fault
  - Depends on application requirements
    - Fully operational (redundancy) Fail safe (safe mode of operation), fail operational (reduced functionality)
- Validate fault
  - Its hall be possible to fully verify & validate safety requirements
    - Fault injection testing

Outline

- Functional safety basic concepts (ISO26262 view)
  - IC failure and their causes
- Management of random faults
  - Safety architecture
  - Safety analysis
- Management of systematic faults
  - Product creation process
  - Compliance driven development process
  - Safety management after release to production

Qualitative safety analysis (FMEA)

- FMEA results
  - Potential IC failure modes and causes.
  - Critical characteristics of the IC
    - Help in design priority setting
    - Recommended actions for reducing severity,
    - Eliminate or reduce the causes of IC failure modes
    - Improve failure detection
  - Feedback of design changes to the design community

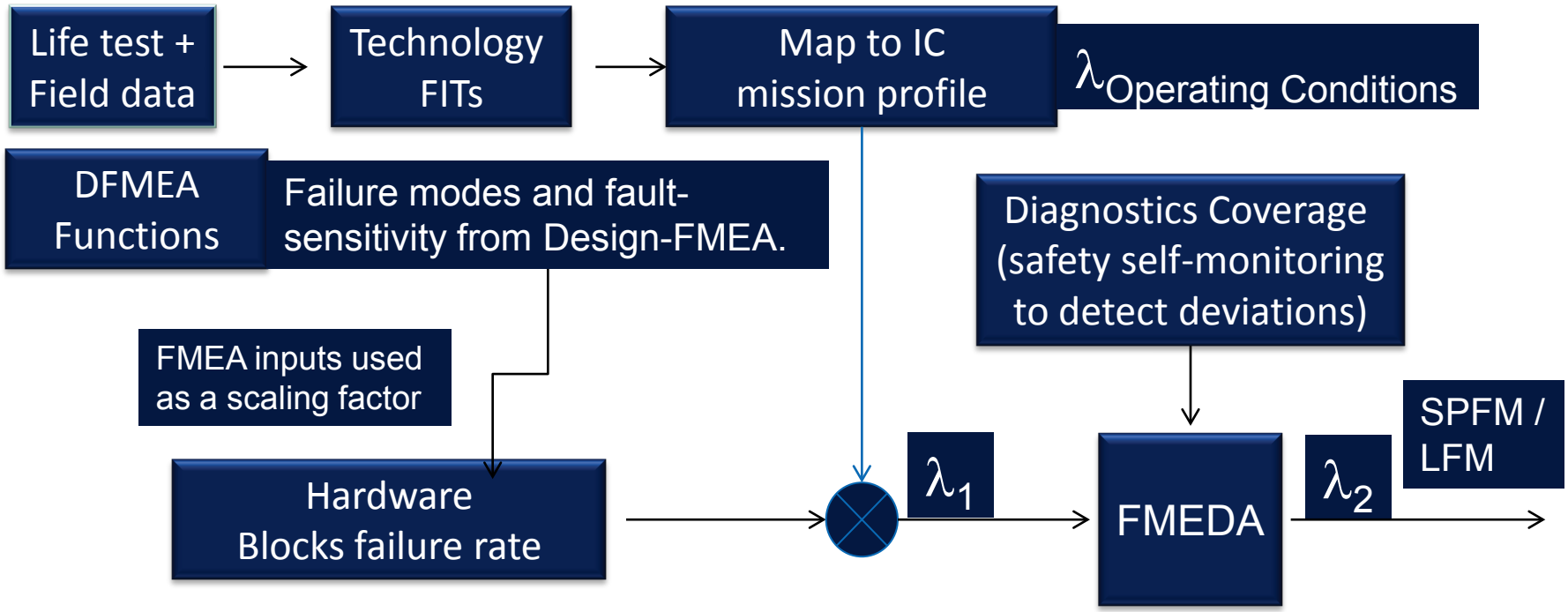


Quantitative safety analysis (FMEDA)

- Based on Diagnostics coverage
  - Ratio of detectable failures probability against all failure probability
  - Diagnostic or self checking elements modelled
    - Complete Failure Mode Coverage .
    - All failure modes of all components must be in the model
- Failure Mode Classifications in FMEDA
  - Safe or Dangerous: Failure modes are classified as Safe or Dangerous
  - Detectable : Failure modes are given the attribute DETECTABLE or UNDETECTABLE
  - Four attributes to Failure Modes: Safe Detected(SD), Safe Undetected(SU), Dangerous Detected(DD), Dangerous Undetected(DU)
- **Goal : Statistical Safety:** based on Safety Integrity Level (ASIL)

Combined Safety analysis

- D-FMEA inputs (function criticality) are used as scaling factor in FMEDA



Outline

- Functional safety basic concepts (ISO26262 view)
  - IC failure and their causes
- Management of random faults
  - Safety architecture
  - Safety analysis
- Management of systematic faults
  - Product creation process
  - Compliance driven development process
  - Safety management after release to production

Safe IC development process

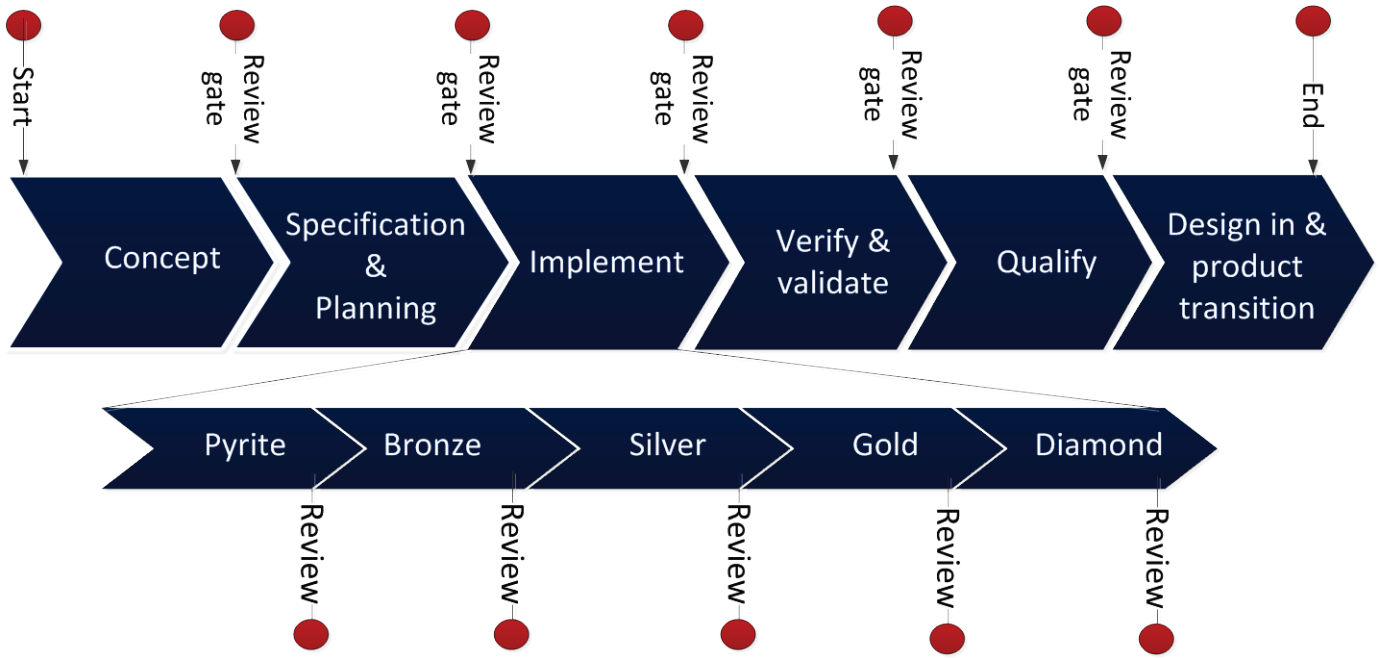
- Product creation process
  - Based on a gated product Creation and Management process
- Compliance driven development process
  - V model with safety extensions
- Safety analysis
  - Combined qualitative (FMEA) and quantitative (FMEDA)
  - fault injection testing for validation of diagnostic coverage
- Safety management after release to production
  - Managed through a separate department (test and product engineering)

Outline

- Functional safety basic concepts (ISO26262 view)
  - IC failure and their causes
- Management of random faults
  - Safety architecture
  - Safety analysis
- Management of systematic faults
  - Product creation process
  - Compliance driven development process
  - Safety management after release to production

Product creation and management

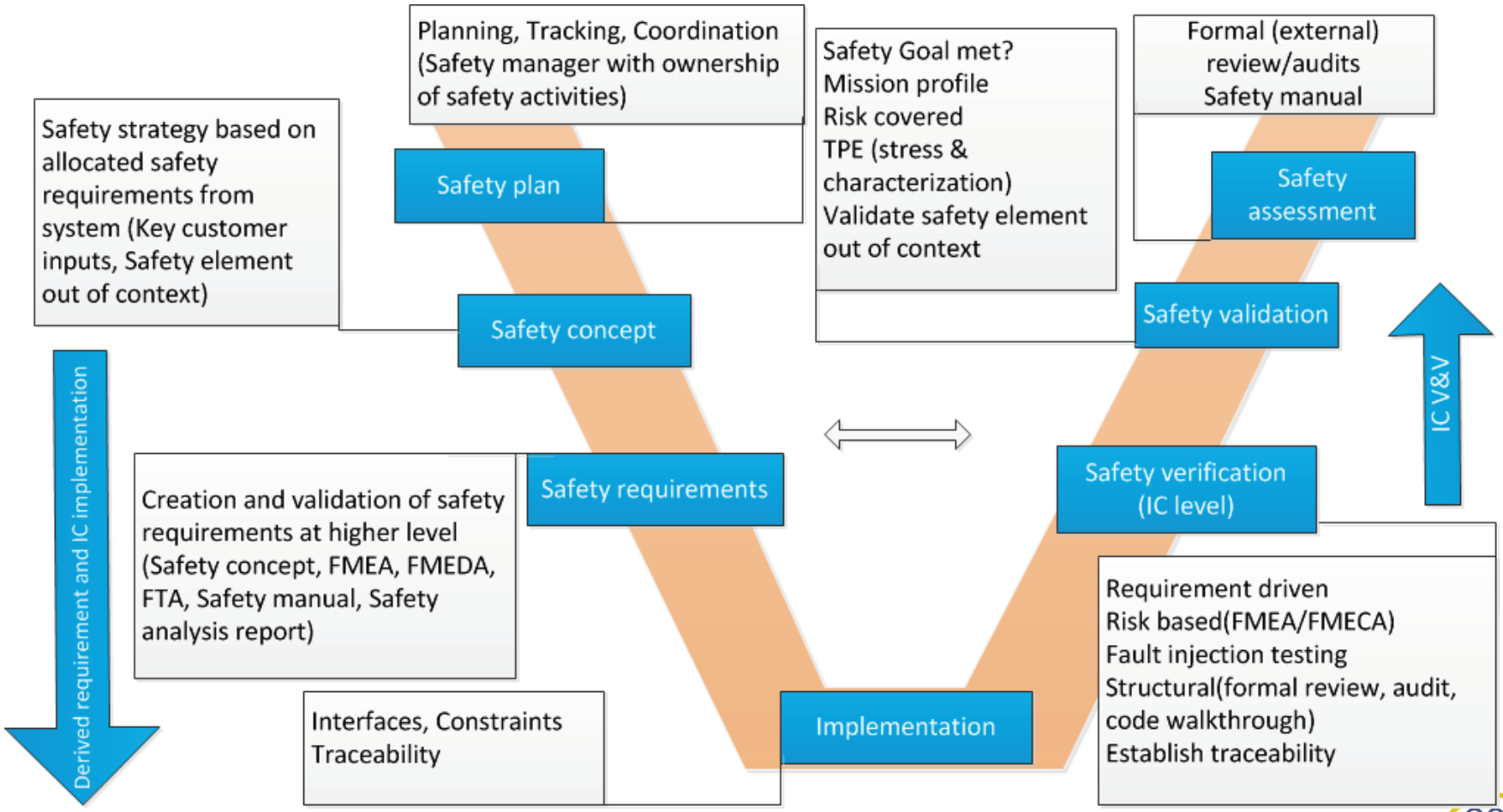
- Divide the project into manageable phases and sub phases
  - Thorough gate review process
    - Product and process compliance at every gate review/audit



Outline

- Functional safety basic concepts (ISO26262 view)
  - IC failure and their causes
- Management of random faults
  - Safety architecture
  - Safety analysis
- Management of systematic faults
  - Product creation process
  - Compliance driven development process
  - Safety management after release to production

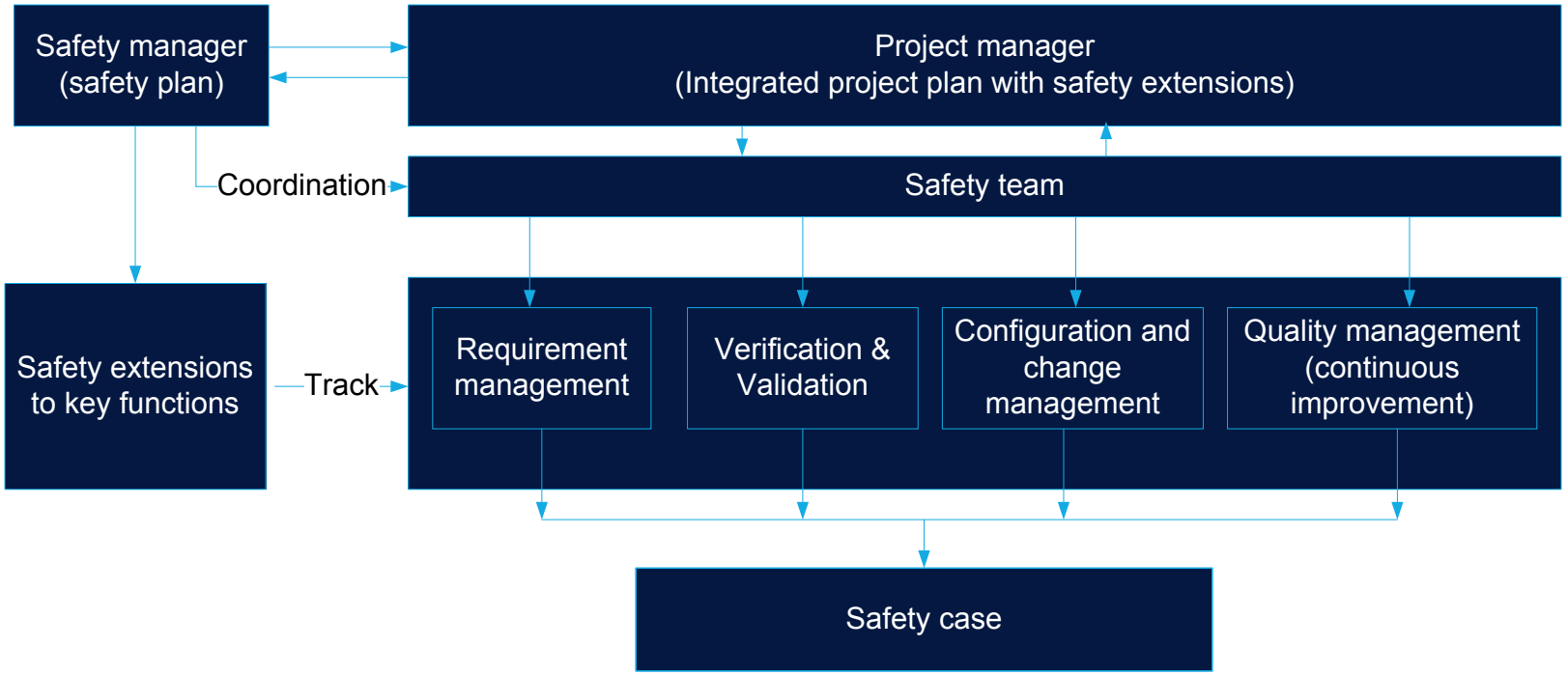
Safety lifecycle on top of “V” model



Safety life cycle extension

Project plan	Safety extension to project plan	Proof of safety
Project planning Project organization Development process	Safety life cycle activities (Safety org, Safety manager, safety plan, tracking & assessment) Safety team organization & management	Safety case and Safety arguments
Requirement management		
Requirement capture & requirement engineering, FMEA(qualitative)	Derive Safety goals, safety requirements, safety concept, and safety requirements and methods. FMEA, FMEDA & FTA (quantitative)	Safety goals, Safety concept Safety requirements with proof of traceability FMECA/FMEDA/FTA reports Safety manual
Verification & validation		
Product assessments V&V Reviews/audit	Safety V&V, Safety assessment, Tool qualification	Safety V&V reports Safety assessment reports Validation reports (proven in use argument)
Configuration management		
Work products Baseline management Change management	Safety case as CI Impact analysis on safety functions(for CR)	Baseline for safety case & safety case reports Traceability and impact analysis report liked to requirement management
QM & process assurance		
Quality manual Template/guidelines, Checklist/examples	Safety culture, Project level tailoring, Continuous improvements Reviewing and tracking compliance with plans. Identifying and documenting deviations from plans.	Tool confidence report, Evidence on safety culture deployment, Project tailoring report. Qualification reports, Risk assessment reports, Periodic confirmation review reports

Safety organization

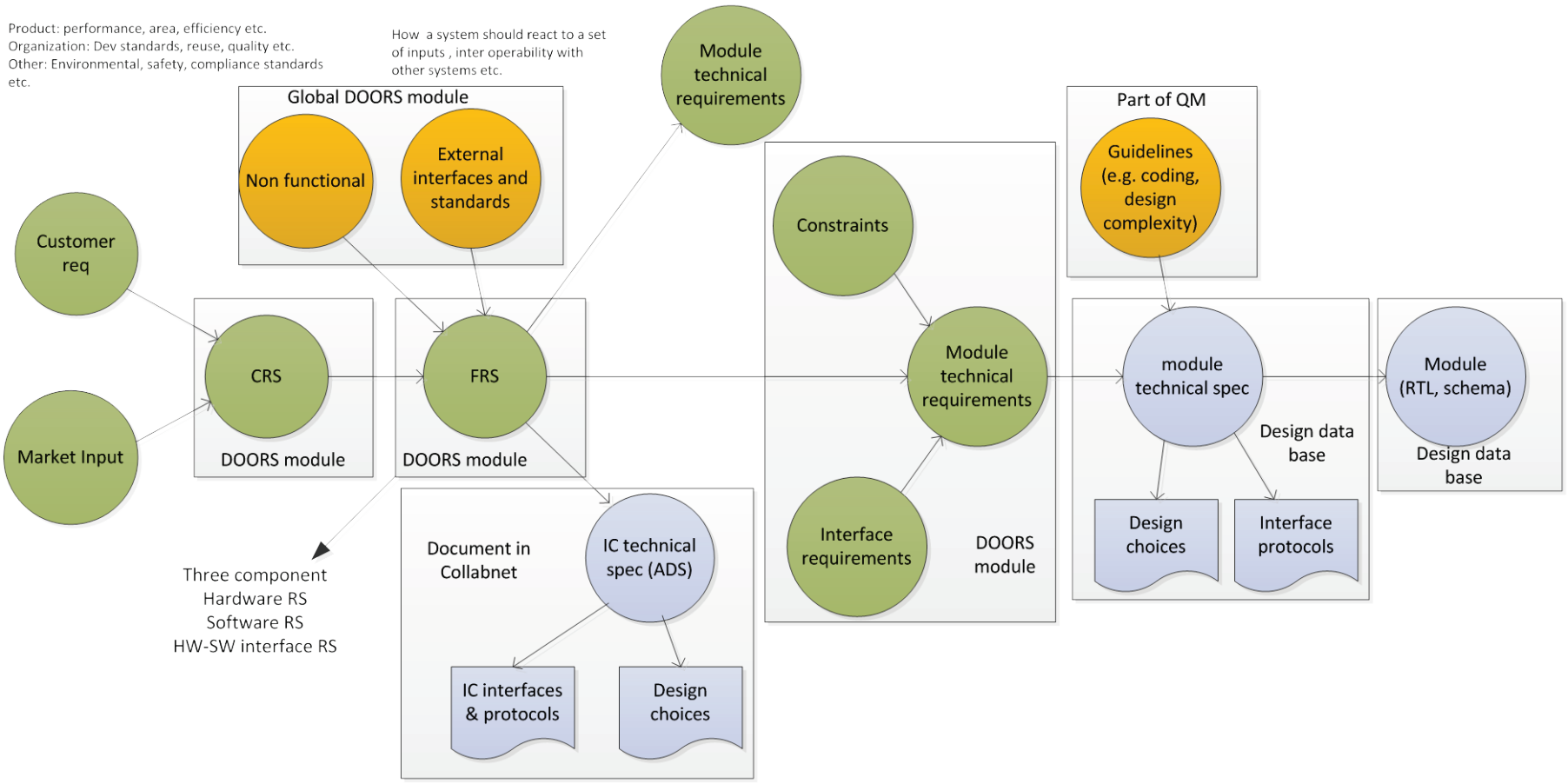


- Safety plan integrated into overall project plan
- Safety extension to key process

(Safe)Requirement management

- DOORS based formal requirement management with bi directional traceability
  - Formal process to track
    - Customer requirements
    - Allocated requirements from system
  - All requirements can be tracked to a design, verification and validation item
  - No design element in repository without an assigned requirement
  - Any PR raised can be tracked to a corresponding requirement ID
  - Impact on existing requirements can be identified for CR

Requirement management tree



Requirement Verification

- Requirements Verification (product right)
  - Checks consistency of the requirements specification artefacts and other development products (design, implementation, ...)
    - Req->design->RTL/schematics-> netlist-> Back End ->GDS2
    - Safety not compromised during design translation process
  - Each requirements are associated with a verification item

Verification is performed throughout the entire safety lifecycle, by each specific party involved, for each of the major work products.
Compliance driven verification is the key

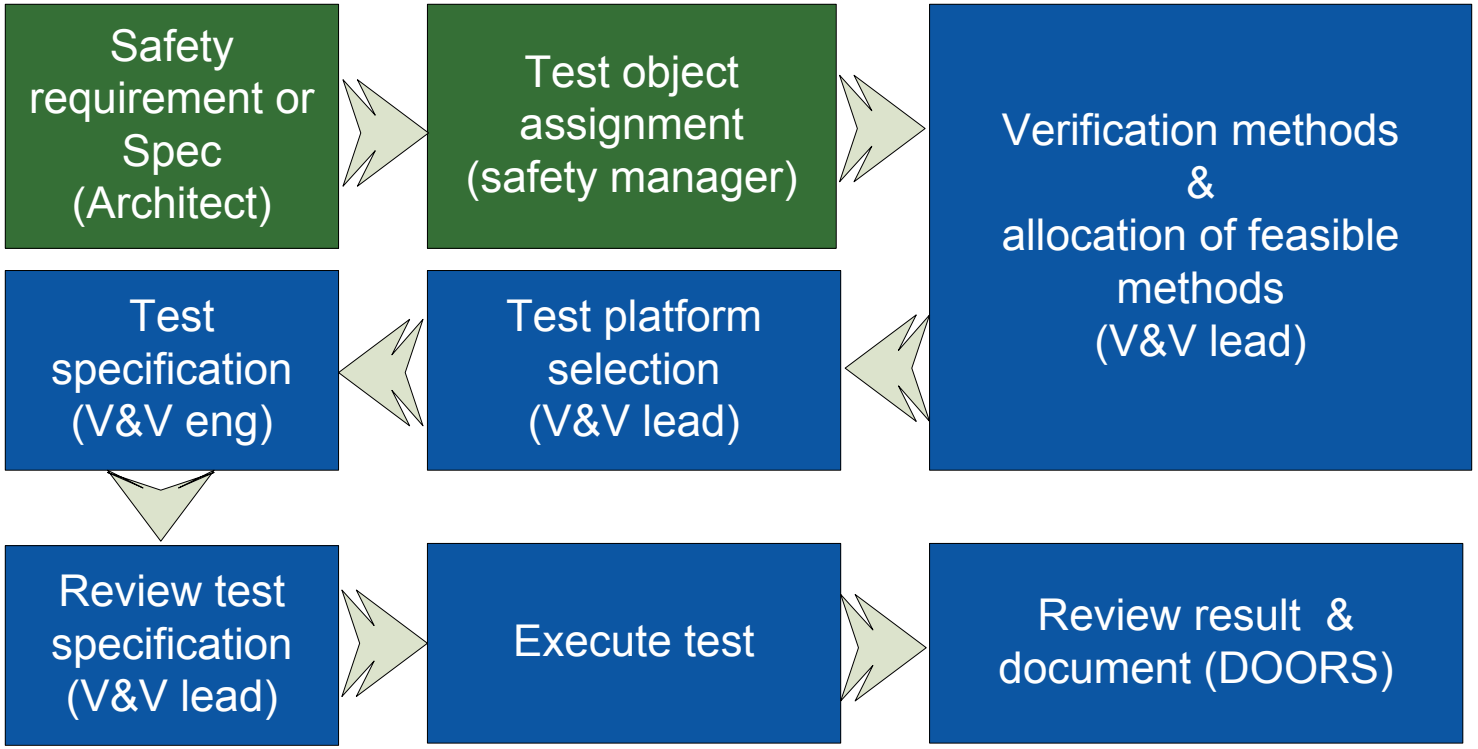
Requirement validation

- Requirements Validation (right product)
  - Check IC requirements specification against customers goals and requirements
  - Provides assurance that the hardware item derived requirements are correct and complete with respect to system requirements allocated to the hardware item
  - Ensure that the highest level safety requirement, the safety goal, has been met and that they are correct
    - The product is safe to use within the mission profile

Validation has a specific meaning against “safety element out of context”. Make sure that safety goals are met trough creation of “proven in use arguments”

V&V work flow (safety extension)

- Clear ownership on who does what within V&V organization



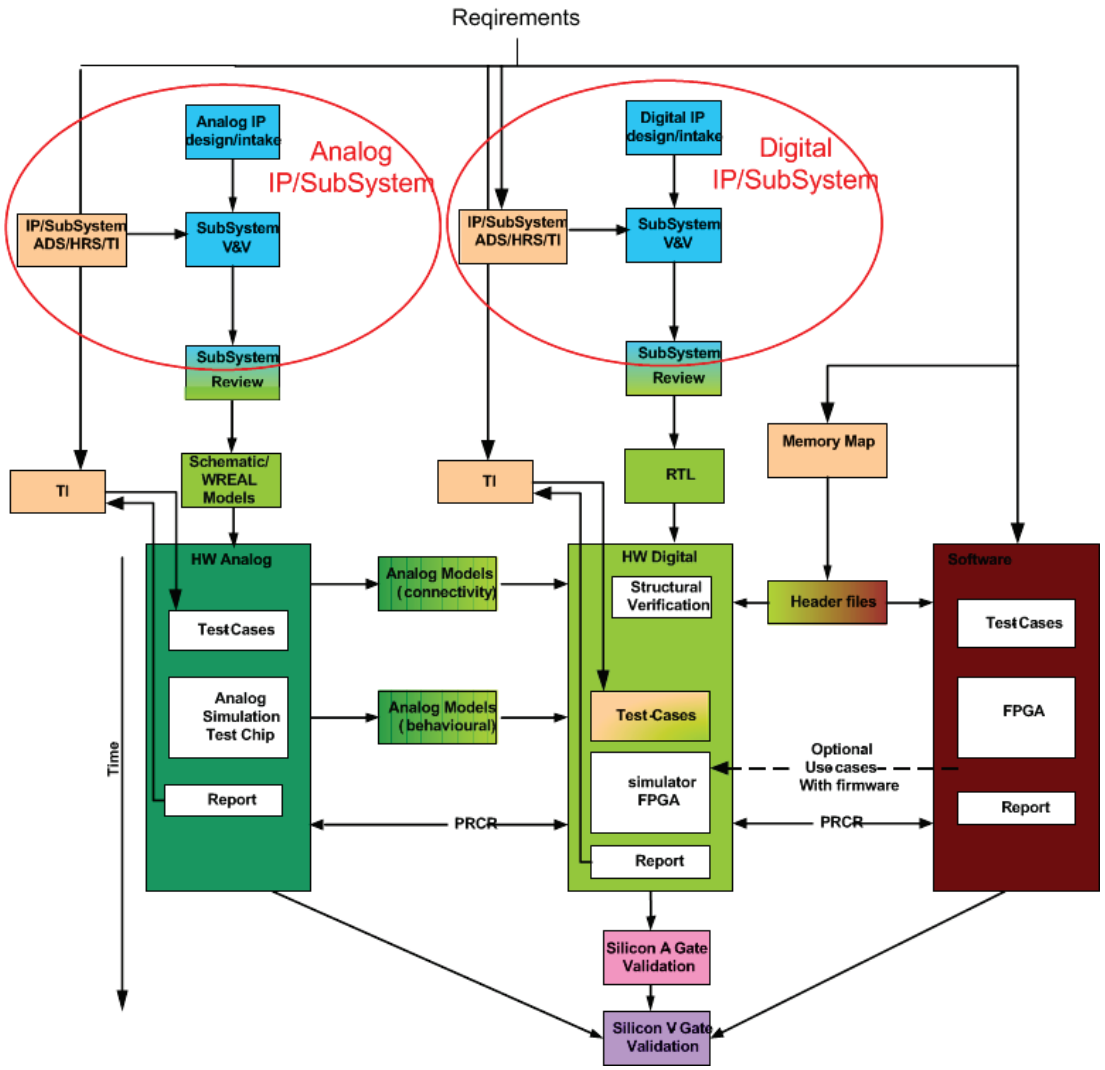
Compliance driven Verification scope

- Compliance driven verification
  - Coverage: Link coverage item to requirements with evidence(100% requirements coverage) In addition to other standard coverage metrics
  - Test plan, test item, test results linked to DOORS to establish traceability
  - Verification at every stage of data translation (req-design-RTL-netlist-post layout)
- Constraints driven
  - Interface constraints(limit, boundary value etc.), rules, design constraints such as Boundary value analysis, negative testing, control flow (protocols), data flow (protocol), overflow, underflow
- Design intent verification
  - Intended operational environment and rules
  - Temporal behavior to stimuli (protocol sequence, error recovery
- Risk oriented
  - Inputs from FMEA/FMECA
- Fault injection testing
  - Coverage for safety architecture components & diagnostic capabilities
- Structural
  - formal review, audit, code walkthrough

Validation (safety extensions)

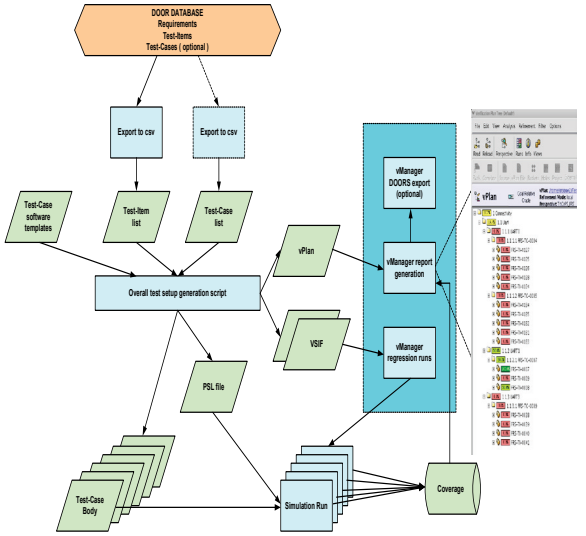
- Check whether safety goal is met
  - Customer driven or safety element out of context
  - Confirmation check on safety goals, safety architecture against functional safety of the items at intended use case level
- Use case validation
  - Question/review safety goals from a use case perspective
  - Compliance with governing standards and other regulatory requirements
    - Interface standards, EMI/EMC etc.
- Fault injection testing
  - Validate safety architecture and diagnostic features
- Evidences against all the above

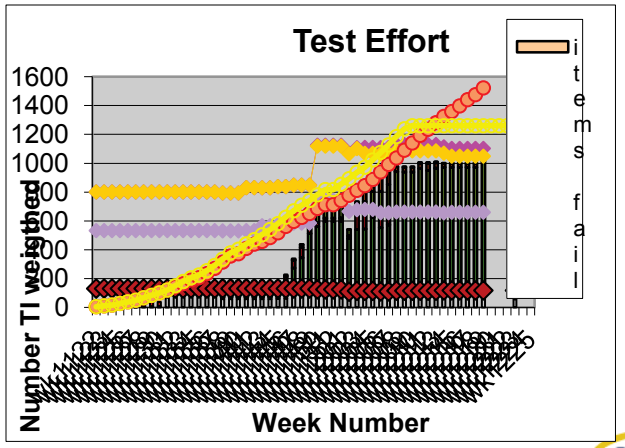
Verification & validation flow



Requirement traceability

- Both requirements and test items are in DOORS data base
- Impact on verification items due to requirement changes are immediately notified
- Easy to trace back a problem report back to the originating requirement
- Automated reporting on V&V progress

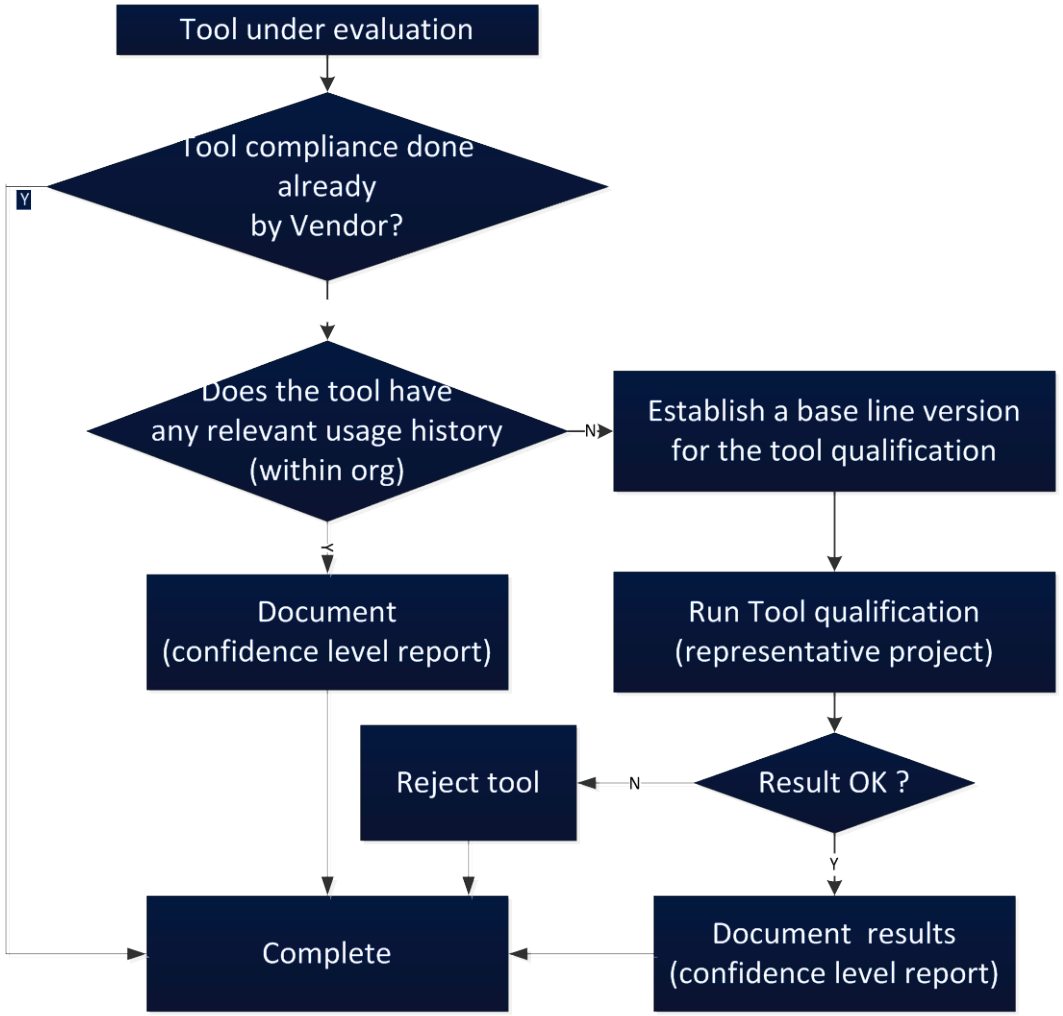




Tool qualification

- Why tool qualification
  - IC design involves many translation process, and tool in general has the capacity to introduce error during various translation process
    - e.g. RTL-> Synth netlist -> post layout
  - Verification tools may fail to detect errors in the hardware items
- Tool qualification makes sure that tool correctly functions (improve confidence in tool function)
- The library components and different views used during the design translation process also need to be of mature quality and qualified one

Tool qualification flow



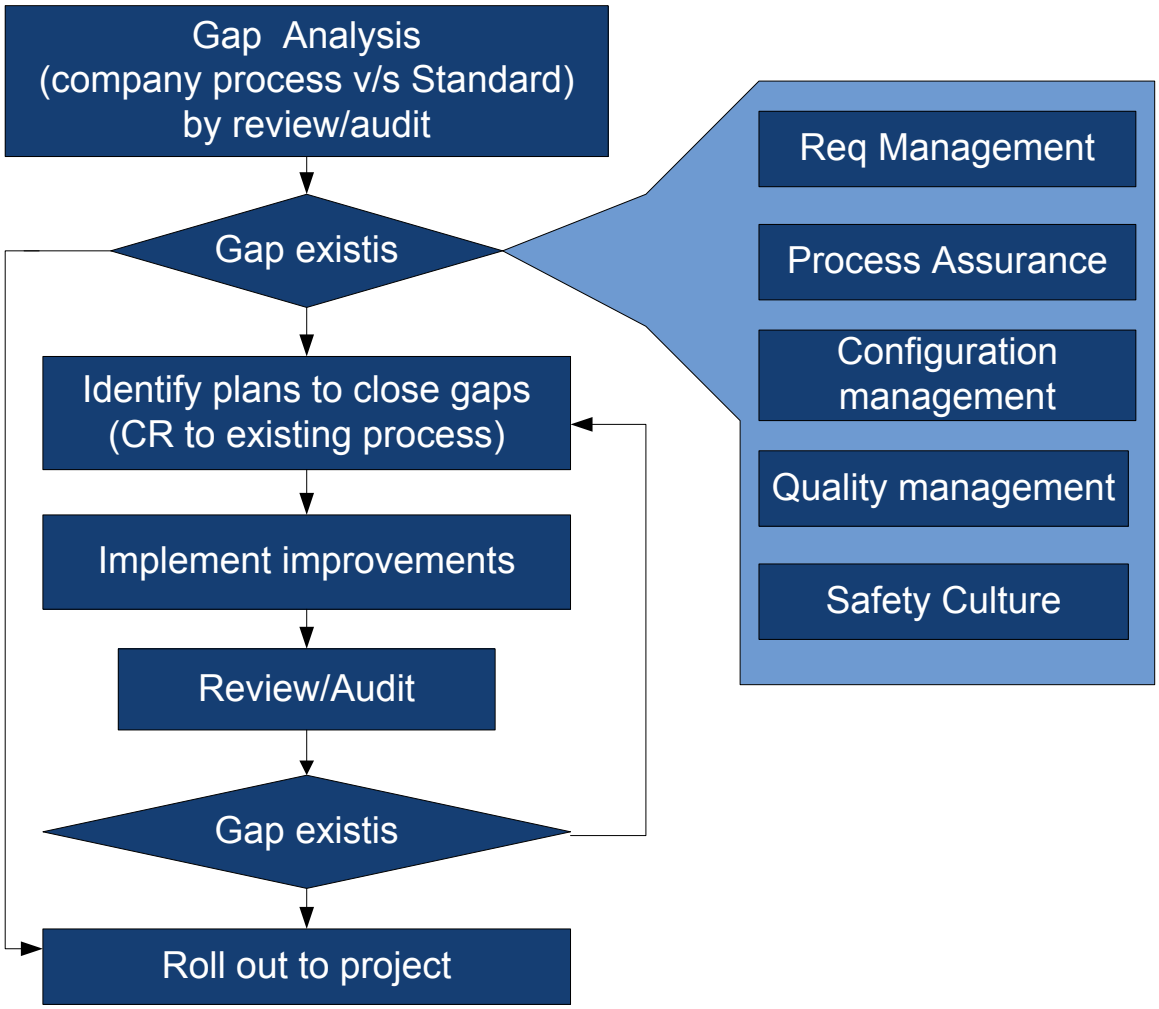
Configuration & Change management process

- Configuration management plan aligned with ISO TS 16949, ISO9001
  - Maintenance of safety case as a living argument to provide a mechanism to establish the interdependency between the elements of safely case
- Change management
  - Establish link between change management & problem reporting process and requirement management (through DOORS)
- Safety representative in CCB
  - Ascertain the impact on functional safety w.r.t change requests

Quality management and process assurance

- Process Gap analysis (safety)
- Safety culture within organisation & project team
- Process tailoring with respect to the project
- Process assurance
  - Design data base management
  - Design assurance

Process gap analysis



Gap analysis

Requirement engineering	Demonstrate process for - Requirement capture - To establish requirement traceability
Process assurance	• Reviewing and tracking compliance with plans. • Identifying and documenting deviations from plans. • Determining if transition criteria have been met between lifecycle stages.
Configuration management	Demonstrate supporting processes - Change Management - Management of Requirements - Configuration Management
Quality management	•Quality manual update for safety life cycle process & methodologies •Supporting process (check list, templates guide lines) •e.g. FMEA/FMEDA template & process
Safety culture	•Evidence of management of Safety Life Cycle activities •Organisation chart showing safety role •Competence management (training & qualification program) •Safety assessment

Safety culture

Normal development process	Safety culture
Safety measures are not planned. No Specific budget for safety	Necessary measures are planned according to safety plan (planning, tracking, coordination). Safety plan is integrated into project plan (receivables, deliverables, and resources)
Risk analysis & tracking is done, but may not structured (updated on regular basis)	Structured risk analysis is a must at beginning of project, and is continuously updated with evidence
Typically an FMEA cycle is followed (qualitative analysis)	Both qualitative (FMEA) and quantitative (FMECA/FMEDA,FTA) analysis is a followed at beginning of a project and is continuously updated with evidence
Change request are accepted at any stage (primarily look only into schedule impact)	Change requests are analyzed and evaluated against functional safety and accepted only through a strict formal change management process (four eye principle) and impact analysis
Safety assessment not a must	Safety assessment is a must and evidence preserved until end of life

ISO 26262 tailoring for IC (HW only) project (example)

ISO26262 tailoring for Dolphin	Applicability	Remark
2 –Management of functional safety	Applicable	Based on functional safety plan
3 –Concept phase	Driven by customer inputs	
4 –Product development at the system level	Mostly applicable	Primarily towards system validation
5–Product development at the hardware level	Applicable	Most of the diagnostic requirements are here
6–Product development at the software level	Limited (for IC do not contain a MCU)	Validate API (MCU) interface
7 –Production and operation	Applicable (IC test & production & engineering)	Big influence on FIT by Sem Con manufacturing process
8 –Supporting processes	Mostly applicable	
9 –ASIL oriented and safety analysis	Based on customer inputs & application profile	Different use cases

Design environment management

- Focus on effort spend up front to prevent design problems due to design environment
- Infrastructure review Scope
  - Data base structure
  - Development standards & flows
  - CM, CR/PR system and tracking methodology
  - Effective simulation analysis and regression systems
    - E.g. Automation in simulation/synthesis log analysis.
  - Communication between architects, design engineers, & verification engineers
    - Documentation availability & traceability
- Planned audit
  - Collect the documentation maintained w.r.t database
    - Check all the team members in the team get right information at right time
    - Establish inks & traceability
  - Audit on CR /PR tracking
    - Defect density
    - CR/PR flow
  - CM audit

Design assurance

- Formal reviews & audits
  - By external experts
    - IP Vendor & IP quality checklist (IP reuse & IP intake strategy)
    - Technology Library
    - Process Reliability measures
  - Project level
    - Data base (CM, CR/PR, Documentation)
    - Compliance check against FRS/Impl/Verification
    - Verification strategy/methodology review
    - Verification coverage review
    - Explicit SoC constraints review
    - Non functional requirement review (coverage)
    - Layout review by a separate team
    - Pin/PAD list

Questions