SoC Emulation Test Plan

1. Scope of Plan

Aspect	Details	
Objectives	Functionality validation, performance testing, power state transitions, security testing, etc.	
Scope	Define what is **in scope** and **out of scope**.	
Target Platforms	Emulation, FPGA, Silicon.	(e.g., functionality validation, performance testing, power state transitions, security testing, etc.).

- Specify what is **in scope** and **out of scope**.
- Mention the target platforms (Emulation, FPGA, Silicon).

2. Emulation vs. Real Silicon Differences

Aspect	Details
Emulation Limitations	Performance, cycle accuracy, missing components.
Real Silicon Only Features	Features only testable in silicon.
Workarounds	Workarounds used in emulation.

- Limitations in emulation (e.g., performance, cycle accuracy, missing components).
- Features only testable in silicon.
- Workarounds used in emulation.

3. Major New Features and Deltas

Domain	New Features/Improvements (Populate as applicable for each test item)
Multimedia IP	List new multimedia-related features.
PMM	Power management module updates.
IO	Changes in input/output processing.
BIOS/OS Boot	Enhancements in boot process.
Security	Security feature additions/modifications.

- List new features introduced in this silicon generation across different domains (Multimedia IP, PMM, IO, BIOS, OS boot, Security, etc.), but only populate relevant sections for each test item.
- Highlight modifications or improvements compared to previous versions.
- Any dependencies on firmware/model updates. | Domain | New Features/Improvements | |--------|-------------------------| | Multimedia IP | List new multimedia-related features. | | PMM | Power management module updates. | | IO | Changes in input/output processing. | | BIOS/OS Boot | Enhancements in boot process. | | Security | Security feature additions/modifications. |
- List new features introduced in this silicon generation across different domains (Multimedia IP, PMM, IO, BIOS, OS boot, Security, etc.).

- Highlight modifications or improvements compared to previous versions.
- Any dependencies on firmware/model updates.

4. Model/Firmware/Infrastructure Dependencies

P1/P2	Logical Sync	Task	Emulation Model	Pass/Fail Criterion	FW Dependency	Model Dependency
P1						
P2						

- Define required firmware versions.
- Maturity of emulation models (pre-silicon, post-silicon correlation, known limitations).
- Hardware/software infrastructure dependencies (e.g., debug tools, logging requirements, validation frameworks). | Component | Dependency Details | |------------|------------------| | Firmware | Required firmware versions. | | Emulation Models | Pre-silicon, post-silicon correlation, known limitations. | | Infrastructure | Debug tools, logging requirements, validation frameworks. |
- Required firmware versions.
- Maturity of emulation models (pre-silicon, post-silicon correlation, known limitations).
- Hardware/software infrastructure dependencies (e.g., debug tools, logging requirements, validation frameworks).

5. Test Configurations & Platforms

- Define key test environments used in validation.
- Specify whether testing occurs in full SoC emulation, subsystem-level, or real silicon.
- Note any special configurations such as testing with/without real PHY.

Test Feature	Task	Model Requirement	Firmware Requirement
Feature A	Task 1	Model X	FW Version Y
Feature B	Task 2	Model Y	FW Version Z
Feature C	Task 3	Model Z	FW Version W

- Ensure alignment with hardware and software dependencies.
- Define key test environments used in validation.
- Specify whether testing occurs in full SoC emulation, subsystem-level, or real silicon.
- Note any special configurations such as testing with/without real PHY.

Hardware/Software Dependency	Details
Hardware Setup	Specify required boards, emulation models, or silicon.
Firmware	List required firmware versions.
Debug Tools	Mention debuggers, log analyzers, and tracing tools.
Validation Framework	Identify automation tools or test scripts used.

- Define key test environments used in validation.
- Specify whether testing occurs in full SoC emulation, subsystem-level, or real silicon.
- Note any special configurations such as testing with/without real PHY, hybrid model,
- Ensure alignment with hardware and software dependencies. | Test Category | Example (PMM) | |--------------|--------------| | Full SoC Emulation | Testing overall system power

states, including transitions and wake-up behavior. | | Subsystem-Level Testing | Validating power domains, voltage/frequency scaling within PMM. | | Emulation with/without Real PHY | Comparing power impact with and without real PHY components. | | Silicon Testing Conditions | Measuring actual power consumption and efficiency on real silicon. |

- Define different test setups relevant to each domain (e.g., full SoC emulation, subsystem-level testing, emulator with/without real PHY, silicon testing conditions).
- Specify test equipment, software tools, and configurations required. | Test Setup | Description | |-----------|------------| | Full SoC Emulation | Complete system-level testing. | | Subsystem-Level Testing | Individual component verification. | | Emulation with/without Real PHY | Evaluating real vs. emulated PHY performance. | | Silicon Testing Conditions | Criteria and configurations for real silicon tests. | | PMM Example | Testing power state transitions, voltage/frequency scaling, low-power entry/exit behavior. |
- Define different test setups (e.g., full SoC emulation, subsystem-level testing, emulator with/without real PHY, silicon testing conditions).
- Specify test equipment, software tools, and configurations required. | Test Setup | Description | |-----------|------------| | Full SoC Emulation | Complete system-level testing. | | Subsystem-Level Testing | Individual component verification. | | Emulation with/without Real PHY | Evaluating real vs. emulated PHY performance. | | Silicon Testing Conditions | Criteria and configurations for real silicon tests. |
- Define different test setups (e.g., full SoC emulation, subsystem-level testing, emulator with/without real PHY, silicon testing conditions).
- Specify test equipment, software tools, and configurations required.

6. Test Sequence

Step	Description
Entry Criteria	Conditions required to start testing.
Execution Steps	Boot to OS, enter low power state, validate wake-up behavior, security validation steps, media playback verification.
Pass/Fail Criteria	Define criteria for test success or failure.
Automation/Manual	Specify whether the test is automated or manual.

- Describe test entry and exit criteria.
- Define Pass/Fail criteria for each test.
- Provide a sequence of test execution (e.g., boot to OS, enter low power state, validate wake-up behavior, security validation steps, media playback verification).
- Specify automation/manual execution.

8. Known Issues & Debug Hooks

Issue/Hook	Details
Known Issues	Document known issues or expected failures. Reference Jira IDs for tracking.
Debug Mechanisms	Logs, traces, breakpoints, security validation tools, profiling utilities.

- Document known issues or expected failures, referencing Jira IDs for tracking.

- List available debug mechanisms (e.g., logs, traces, breakpoints, security validation tools, profiling utilities). | Issue/Hook | Details | |------------|--------| | Known Issues | Document known issues or expected failures. | | Debug Mechanisms | Logs, traces, breakpoints, security validation tools, profiling utilities. |
- Document known issues or expected failures.
- List available debug mechanisms (e.g., logs, traces, breakpoints, security validation tools, profiling utilities).

Aspect	Details
Documentation	Define how test results will be documented.
Reporting	Specify reporting format and frequency.
Escalation	Outline escalation process for failures.

- Define how test results will be documented.
- Specify reporting format and frequency.
- Outline escalation process for failures.