**Introduction**

A systematic approach is inculcated to target an efficient and smooth verification flow for low-power simulations for large SOCs. The analysis of power/voltage/clock domains defines verification scope. One-time compile loading the power information of design PMIC controller mimics the behavior in the test bench. Power operations isolation, level shifters, retention, and checks to reinforce DV.

The finest techniques deployed based on the new verification approach resulted in saving up to 30% power consumption on silicon. The fully configured and powered on module consumed 41mW (18% less) against the planned 50mW, idling power with power rail on consumption 8.44mW → 5.86mW (30.6%) and the power gated block with minimal functionality to the block was found to consume only 0.58mW against planned 0.65mW (11%).

**Proposed Methodology**

- SOC consists of 75 power domains, 79 clock domains and 37 voltage domains.
- The SOC consists of 13 power manager IPs which deployed various low-power techniques:
  - Variable voltage level for logic/memories/hard macros: Operating from 0.6V to 3.3V.
  - Switchable power supplies: Power gating using standard library power switch cells.
  - Clock shut off: Clock gating when PD is idle.
  - Isolation cells: Avoid leakage to nearby PDs.
  - Level shifters: Sampling signals that cross voltage levels.
  - Retention: Data saved during power shut off and restored during power on.

**Implementation Details**

The Master slave power controller handshakes with the basic power domains of the block-level power controller and the internal power domains are controlled by the Slave power controller. The Slave power controller controls the power domains and decides the power state of each domain in each of the power states. The Slave power controllers have AXI master interfaces which help in clock gating.

**Power Domains & Power Down Flow Chart**

The Software based power down is a slow process as we need multiple writes to multiple registers to achieve the same. The Hardware approach is a permanent process and no of wires increases and so does the fields in registers, this calls for an optimized process of Hardware and software based approach in which we keep a balance of both hardware and software advantages/disadvantages.

With efforts from architecture and DV sequences we were able to come up with different power states in which the Power is reduced and the wakeup is possible based on need. The power savings constitute to a maximum of 90% for block level idling and around 80% for the chip standby.

The regular SW approaches give only a minimal power reduction of around 40% the major contributor is the power rails turned off. Which here is done by an external FSM based machine which is done by the Test bench.

**Result**

**REFERENCES**