The Challenges of Low Power Design: A System-on-Chip with 152 Power Domains

David Bean

Ericsson
High Wycombe, UK

www.ericsson.com

ABSTRACT

Low power methodologies are increasingly required for System-on-Chip (SoC) designs. New applications require ever more computation and this makes power an important consideration for Ericsson's SoC products. This paper discusses the challenges of low power design & verification for an SoC with 152 power domains. An overview of the design challenge is followed by an outline of the design flow using Synopsys low power tools. The paper then focuses on the challenges faced during low power implementation and verification, how these challenges were overcome and lessons learnt that will improve our flow for future designs.

Table of Contents

Table of Figures

Table of Tables

1.Introduction

Increasing computation requirements for new applications coupled with high leakage power for sub 90 nanometer processes has made power a critical consideration for Ericsson's SoC products. This paper discusses the experiences of pioneering the low power design flow for our design group and how we successfully applied this flow to an SoC with 152 shut down domains. An overview of the design challenge is followed by an outline of the design tool flow. The paper then focuses on the challenges faced during the implementation and verification, how these challenges were overcome and the lessons learnt that will improve our flow for future designs. Synopsys low power tools were used in the tool flow. Design Compiler (DC) was used for the front end synthesis, Synopsys Multi Voltage Simulator (MVSIM) was used for dynamic simulation and Synopsys Multi Voltage Rule Checker (MVRC) was used for static checking.

The SoC described is a state of the art ASIC device, manufactured using TSMC 28nm HPM technology.

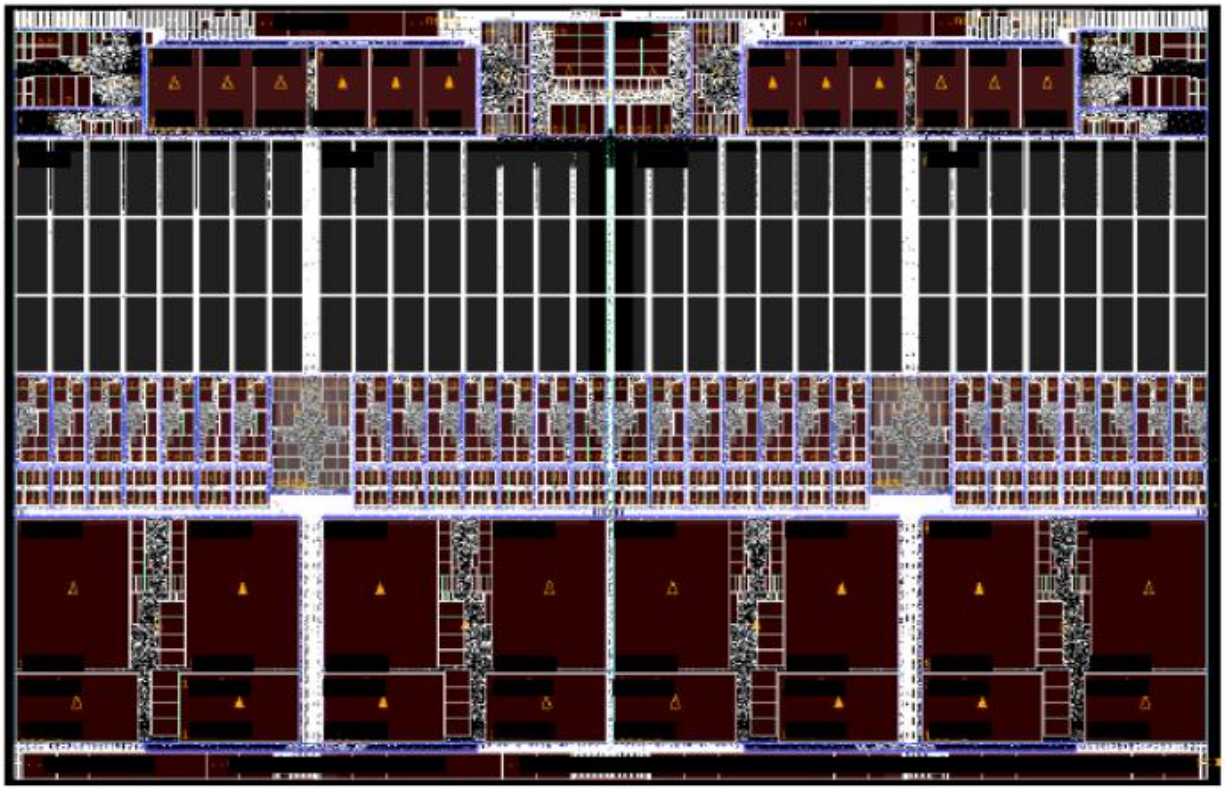

Figure 1. Plot of the SoC physical layout.

2.Design Challenge

The low power technique used was static multi supply design [1]. The voltage level is the same for the whole design. However, many design partitions can be shut down. We also employed RAM macros which have the ability to shut down or enter deep sleep mode.

The operating voltage of the design partitions are intended to remain static following an initial power state configuration. All partitions are powered up when the device is turned on. The device is then configured such that, depending on the application, partitions that are not required are shut down.

During the specification phase a study was conducted to determine the design partitions that would provide significant leakage power savings by being shut down when idle. In total 152 design blocks were identified. This was a large proportion of the physical area of the device and illustrated how integral the low power methodology was to be for a successful outcome.

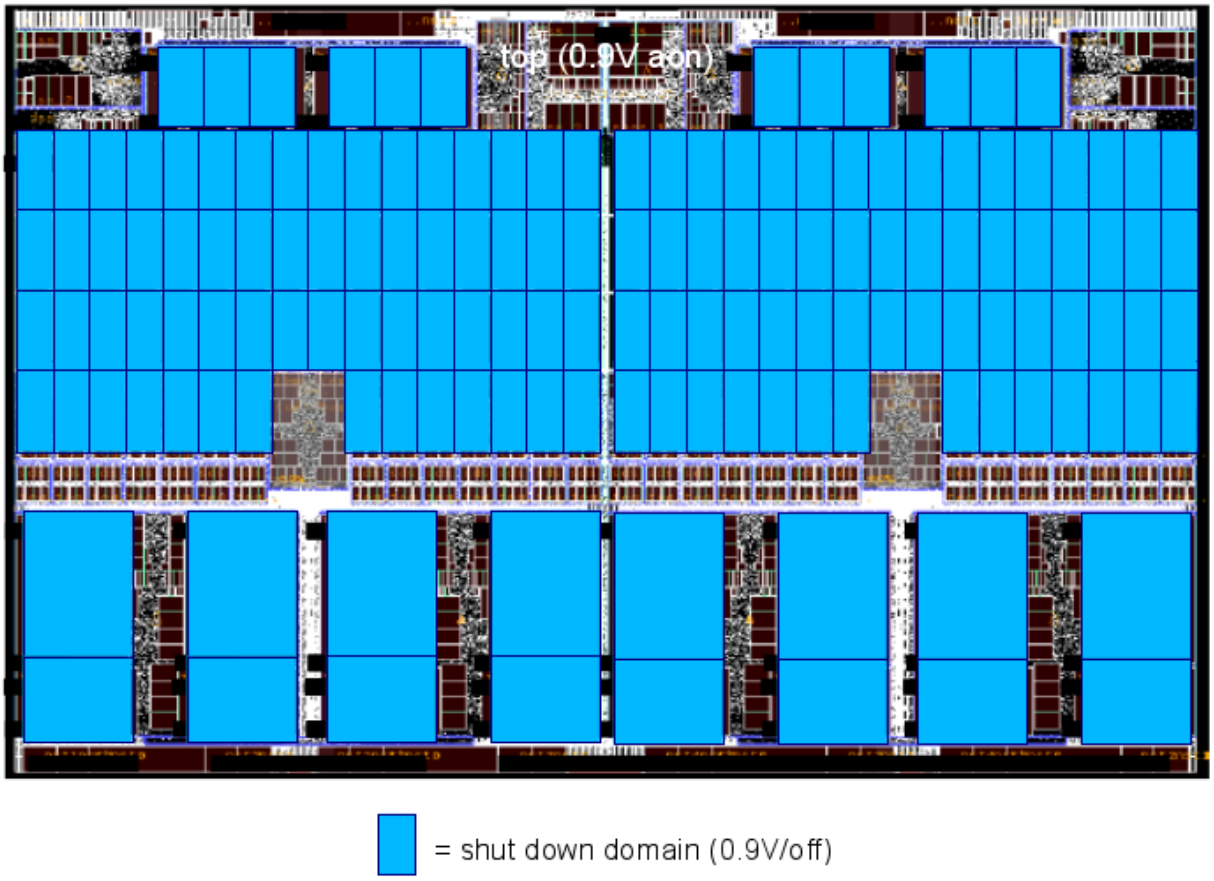

Figure 2 Plot of the SoC showing shut down domains.

The large number of shut down domains presented many challenges to successfully pioneer a low power design flow for our design group using this advanced low power technique with Unified Power Format (UPF) and Synopsys low power verification tools.

3.Tool Flow

Choosing the best tools to use at the start of a project is crucial to a successful outcome. As technology moves fast a common problem is that the electronic design automation (EDA) vendors have not fully developed their products for the technologies you are targeting.

Several power aware simulator options were evaluated at the start of the project and Synopsys multi-voltage simulator (MVSIM) as a co-simulator with Cadence NCSIM was adopted. This option supported the Unified Power Format (UPF) but is achieved using PLI functions which has the disadvantage of adding an overhead to performance. For static checking Synopsys multi-voltage rule checker (MVRC) was the most advanced tool available.

A mixed vendor tool flow was used. The place and route (P&R) tool used was Mentor Olympus which gave some compatability issues with the Synopsys low power tools which are detailed later.

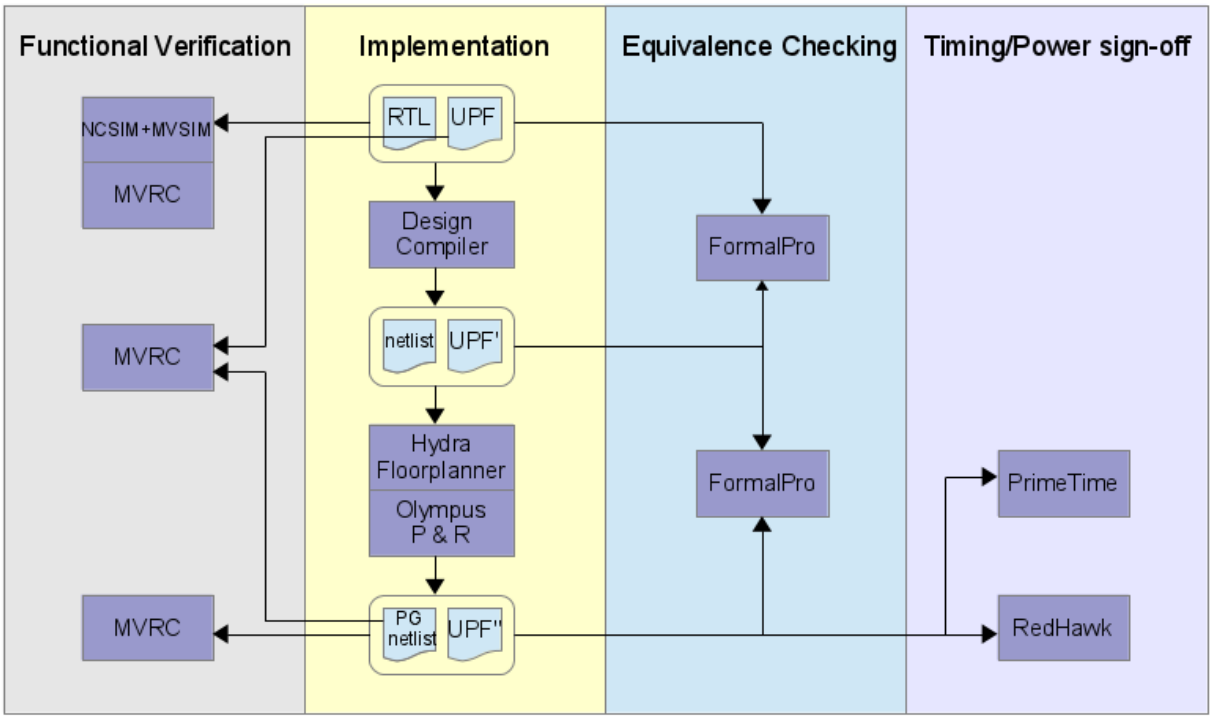

Figure 3. Low Power Flow using UPF

I will now discuss in more detail the implementation & verification of the low power design and present some of the challenges that we encountered.

4.Design and Implementation

Low power design requires a register transfer level (RTL) format (e.g. Verilog, VHDL) plus a power format to specify power supply information. The Unified Power Format (UPF) was developed to allow designers to define in full the power intent of their design. We used UPF to define the shut down power domains, add power switches, protection devices (isolation gates and level shifters) and state tables enable validation of the protection strategy. The UPF sits alongside the hardware description in the design database and both RTL and UPF are used throughout the low power flow. UPF must be supported by all EDA tools in the flow.

Register Transfer Level (RTL) Design

During the RTL design phase efforts were made to write the RTL with consideration for the low power flow. The shut down power domains that would become physical voltage areas were clearly defined by RTL modules. This allows more efficient UPF coding as each power domain maps to single RTL module. In some areas it was necessary to rework RTL to suit the UPF hierarchy structure. We placed RTL wrappers around the shut down domain modules. We instanced all RAM macros in these wrappers so that everything in the shut down domain module was shut down logic (only needed to list one module in -elements). Also, the wrappers are convenient for specifying where to place the isolation gates & level shifters (using -location

parent on the UPF strategy commands). For some shut down domains a wrapper was not practical they were part of an array. To ensure isolation gates were correctly added by the synthesis "-location self" was used on the isolation strategies for these domains and a "set_related_supply_net" command was required to relate all output ports to the always on supply.

Also, because we were pioneering the UPF methodology for our group design reviews were held on a regular basis throughout the flow. This kept the whole team informed of the low power methodology and ensured issues were raised early.

Unified Power Format (UPF) Design

The first challenge was to climb the low power learning curve and to understand low power design techniques and how to write UPF. Synopsys Low Power workshops, update seminars and site visits were invaluable in this regard.

This SoC was pioneering the UPF flow for our design group so we wanted to keep the UPF methodology as simple as possible. We used UPF 1.0 because we used a mixed vendor flow and were not confident that all tools would support UPF 2.0.

The design is very modular. Although there are 152 power domains there are only 5 different power domain partitions. This made a hierarchical UPF structure the obvious choice. This also has the advantage of making UPF reuse easier.

A power intent specification was finalised before starting the UPF design. Some changes were made to the specification before final RTL hand off and the UPF had to be revised accordingly and re-verified.

An initial design decision was which levels in RTL hierarchy would require a matching UPF. A UPF is required for each power domain or power domain wrapper. Then a UPF for each module which is to be a block in the physical hierarchy (P&R block). A problem was that the P&R hierarchy was not fully known when UPF was first designed. A best guess at the P&R modularity was made. Then as the back-end flow progressed and P&R hierarchy resolved changes were required to the UPF hierarchy.

Design and Implementation Challenges

Handling RAM Macros: How to ensure level shifters were placed correctly on RAM macros?

Our RAM macros required a separate voltage supply for the SRAM cores and three input ports (the low power control pins) had this supply as their related supply net. This required level shifters to drive these ports from the core logic VDD. The method to ensure the level shifters were placed correctly by DC was as follows:

1. In the UPF use the "connect_supply_net" to specify the connections for all these pins for every RAM macros.
2. In DC use the variable "mv_allow_ls_on_leaf_pin_boundary"

3. Also, check that your multi voltage library has the required level shifter cell. With attribute on the VDD pin called “scmr“ (std_cell_main_rail). This is required so that the level shifter cell can be placed in a standard cell area with a power grid (main rail) connected to the always-on supply.

Power Management

Our power management was achieved through user programmable control/status registers (CSRs). Obviously care is taken to ensure these CSRs are in the always on domain. The power control CSRs were placed in the wrapper modules where possible.

Shut Down Domain Reset Policy

A design policy was to clamp resets active when isolated. This ensures all shut down domain have resets applied on power up and are initialised to a known state. No retention registers were used in the shut down domains. A problem arose with verification as this gave multiple ISO_DEVICE_REDUNDANT warnings in MVRC which were waived. MVRC flags all isolation gates on inputs to shut down domains as redundant.

UPF Command Order

The Mentor Hydra floorplaning tool required that the “load_upf“ calls to sub-blocks be placed before defining supply nets. The “create_supply_port“ and “create_supply_net“ commands can go before the “load_upf“ commands but any commands which reference ports/nets from the sub-blocks must go after.

Power State Table (PST) Reduction

The PST represents all legal modes of operation in the design. It is used to inform the static checks. A PST with all combinations of the 152 shut down domains on and off was not feasible so steps were taken to reduce the state space.
1. The number of power modes specified were reduced to the minimum that still ensured all the shut down domain crossovers had all source on/off and destination on/off permutations covered.
2. At higher levels modes were overlapped to constrain the PST so that indentical modules are only checked once. For example if a level contains two identical modules specify a power state for one module which constrains that module to one power state. The power states defined for the other module will be merged up so that it is fully checked.

Top Level Supplies

All input/output (IO) supplies had to be present at the UPF top level. This was a requirement of the back end flow. Care was taken to ensure the UPF supplies defined match up with the physical IO groupings on the SoC. Some voltage regions were physically split and this required modifications to the top level UPF.

5.Low Power Verification

Low power verification was achieved using Synopsys multi-voltage tools; multi-voltage compiler (MVCMP), multi-voltage database generator (MVDBGEN), MVRC and MVSIM.

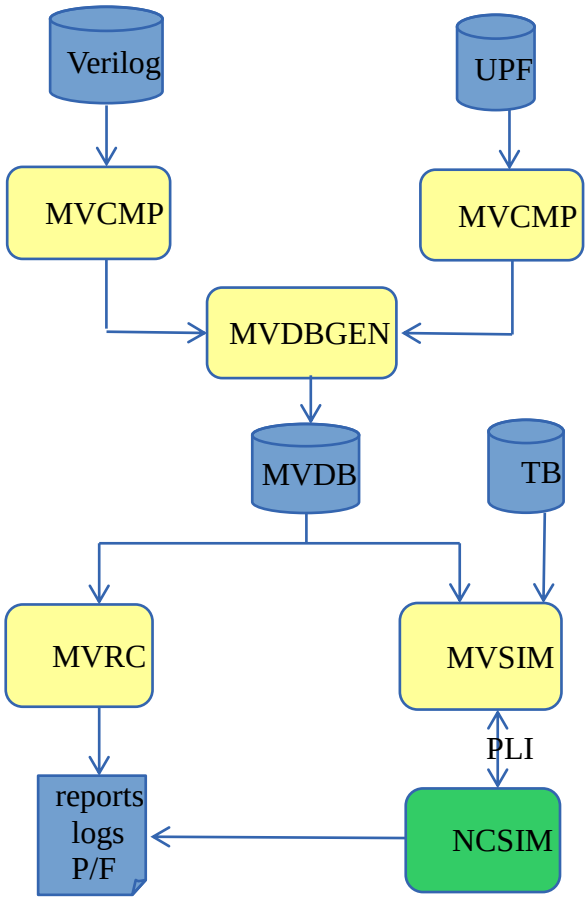

Figure 4. Low Power Verification flow.

MVCMP compiles the Verilog (RTL or netlist) and UPF files. MVDBGEN then creates a multi-voltage database. Static rule checks were run on this database using MVRC. Dynamic simulations were run using Cadence NCSIM verilog simulator with MVSIM as a co-simulator.

Dynamic Simulation (using MVSIM)

Dynamic simulations were run using RTL+UPF (golden) to verify functional correctness. Formal tools ensured functional equivalence between RTL+UPF and netlist+UPF" so that gate level simulations were not required.

A low power test suite was designed to verify the following functionality:

1. Power Off/On sequences.
2. Shut down domains function correctly after power off/on.
3. Blocks that share interfaces with a shut down domain function correctly.

Challenges

Single MVSIM license

We only had one MVSIM license so could only run simulations sequentially. This, together with the MVSIM PLI overhead, meant the full low power test suite took four days to run. We would liked to have been able to launch tests on our multiprocessor grid as this is the methodology for our standard regression tests. This is desired for subsequent projects which would require that the licensing model must be changed so that an MVSIM license for each Verilog simulator license is feasible. It would then be possible to integrate our low power test suite into the standard regression simulations preferably running all simulations in low power mode.

Coverage

The test suite provided coverage for all 152 power domains however the MVSIM coverage tools had limited capability. Custom assertions were written to check that power control signals were exercised for all shut down domains and RAM macros. Assertions were also written to ensure that power/up down sequence rules were not violated. The latest version of VCS-NLP has improved coverage tools.

Static Rule Checking (using MVRC)

Using Synopsys MVRC enables vectorless detection of any connectivity and power sequencing errors in the design. It detects any missing, illegal, or redundant protection gates, and also performs various static architectural checks on the design [1].

MVRC was run at the following stages of the flow and was the sign-off tool for verifying the final netlist:

- Final RTL + UPF (source UPF)
- power and ground (PG) netlist + UPF
- PG netlist +UPF" (UPF written by Mentor Olympus)

MVRC proved to be an invaluable tool. It found many bugs in the PG netlist generated by Mentor Olympus. e.g. RAM macro supplies connected incorrectly, level shifters placed where not necessary, buffers in wrong domain placed after level shifters, always-on buffers placed in shut down domains where not required, missing isolation gates.

Challenges

Mixed Vendor Flow

A mixed vendor flow was used with Mentor Olympus as the P&R tool. This presented a problem generating UPF" that was compatible with Synopsys tools. Mentor added proprietry UPF command options and could not write out the “connect_supply_net“ commands for all multi voltage cells. MVRC sign off was run with the source UPF and the PG netlist with with the supply connections extracted from the netlist by using the “-verilog_pg_connections“ option on MVDBGEN.

Architecture Checks

For architecture checks we used “create_clock“ and “create_reset“ to specify the clock and reset sources. But we also used “infer_sources“ to automatically infer sources of special signals (clocks, resets, iso enables, psw enables, scan enables). The architecture check traces these signals checking for corruption.

A problem occurred when tracing back through clock gate enables. Thousand of “X_PROPAGATION“ errors were flagged because the enable sources were in shut down power domains. These signals were protected by isolation gates and the design was correct but the errors where flagged because they were deemed special signals. The solution to this was to run two check_architecture commands:

1st run: Without tracing through clock gate enables and with halt_at_isolation_data false. This would catch any valid errors relating to none clock enable sources.

2nd run: With tracing through clock gate enables and with halt_at_isolation_data true. This would catch valid errors on clock enables. i.e. signals originating is shut down power domains and without protection gates.

Well Bias Checks

In this design all nwells were to be tied to the always on supply (VDD) including those in shut down domains. The “set_back_bias“ command was used to ensure that all pg_pin's of type nwell were connected to VDD. Similarly this command was used to check that all pg_pin's of type pwell werer connected to ground (VSS).

set_back_bias -domain <domainName> -type nwell -net VDD
set_back_bias -domain <domainName> -type pwell -net VSS

Unfortunately a separate command is required for each power domain. A for loop was used in the script that built the MVRC command file. Some bugs were found by these checks and they gave confidence, in particular, that the correct well tie cells were used in the shut down power domains.

Using Isolation gates as a logic gates

In some shut down domains power switch aknowledge signals needed to be combined inside the domain. A 2-input always on gate was not available in the library so an isolation gate had to be used. “set_var ignore_iso_logic_cells true“ was used to identify these cells but the architecture checks still flagged “X_PROPAGATION“ errors and “ISOLATED_AON_SIGNAL“ warnings which had to be waived.

Capacity Issues

MVRC did not have the capacity to run the top level PG netlist. MVRC had to be run in two passes with modules black boxed for each run. This compromised the checking and was only accepted because there were no multi-voltage cells at the top level.

After tape out an evaluation of Synopsys' new multi voltage static checking tool, VerdiSI-LP, found that this tool could run the full chip in three hours. This compared with MVRC running half the chip in twenty two hours. Performance improvements were also greater for larger design blocks.

	VerdiSI-LP		MVRC		Perfomance Improvement Factor	
Block	CPU Time (s)	Peak Memory (GB)	CPU Time (s)	Peak Memory (GB)	CPU Time	Peak Memory
block0	115	1.301	463	2.49	4.03	1.91
block1	85	1.299	458	2.49	5.39	1.92
block2	77	1.193	598	1.7	7.77	1.42
block3	281	1.686	1477	4.34	5.26	2.57
block4	60	1.174	225	1.54	3.75	1.31
block5	93	1.31	527	2.56	5.67	1.95
block6	90	1.277	442	2.35	4.91	1.84
block7	364	2.583	2602	8.08	7.15	3.13
block8	100	1.343	488	2.43	4.88	1.81
block9	129	1.427	643	2.88	4.98	2.02
block10	41	1.059	148	1.51	3.61	1.43
block11	47	1.086	165	1.55	3.51	1.43
block12	68	1.236	279	1.72	4.10	1.39
block13	105	1.368	531	2.49	5.06	1.82
block14	115	1.429	557	3.23	4.84	2.26
top (part black boxed)	4017	20.66	219600	39	54.67	1.89
top	10800	37	NA	NA	NA	NA
Average					8.10	1.88

Table 1. VerdiSI-LP evaluation results.

6.Conclusions

The low power design flow outlined in this paper proved successful for this SoC. Our design team now has a low power flow that we can be confident with going forward into future projects.

Lessons learnt
- Write a power intent specification before starting RTL & UPF coding.
- Write RTL with consideration for low power flow. Build in hierarchy for module wrappers around power domains if possible. Consider the best place in the hierarchy for power control.
- Attend Synopsys low power training courses & workshops.
- Hold regular peer reviews to keep whole design team involved in UPF flow.
- Don't hesitate to contact Synopsys support when problems/issues arise.
- Pipeclean flow down to layout netlist as soon as possible to highlight problems and possible changes required to RTL/UPF structure and hierarchy.

Recommendations for our next product
- Consider a unified vendor flow to avoid problems with file compatability between tools.

➓ Use a simulator with native low power capability.
➓ Use VerdiSI-LP for static rule checking. This has improved capacity and performance and advanced coverage tools.
➓ Integrate low power into all simulation test benches. It should not be a special case. Note that this requires a new licensing model.

7.References

[1] Synopsys Low Power Verification Tools Suite User Guide.