sondrel
Success through partnership

Sondrel Case Study: 16nm Machine Vision NoC Design

The Design

Our client is already a world leader in machine vision technology. Its solution is used in multiple applications involving camera and video, for example in the automotive sector, for surveillance, drones and in gaming products. This technology uses artificial intelligence to enable a machine to search and find objects and to recognise them.

This SoC was the 3rd generation product based on a highly programmable architecture and comprised 16 VLIW processors for video, with dual RISC CPUs for control and several configurable hardware accelerators for video & image processing in real-time enabling computer vision functionality. Key to delivering performance was a network on chip (NoC) and a sophisticated shared memory fabric used to link all the processing elements.

Team Engagement Profile

The Sondrel engineering team for this engagement was based in design centre sites in Europe and Morocco.

The Design

- Machine Vision Application
- RISC Processors, Vector Processors, Hardware Accelerators, Network on Chip (NoC), multiport memories
- Area: of 75x75 mm2. Frequency: 400Mhz, 16nm process

Technical Details

This design involved tasks at top-level and block or modular level. The main challenge was to match the RTL of the design with the unit level reference model for complex algorithms implemented in the hardware accelerators. Many change requests, resulting in the reference model falling behind the RTL within the timeline of the project. The match requirement for this task was 100%, with no exceptions Multiple revisions of RTL and reference model were tested on a daily basis until this target was met.

Top Level

At the top-level, we verified the performance of the NoC. With an architecture of this type, there is huge amount of data that must be processed in parallel. One of our tasks was to test the design to measure any stalling of the processing elements, so that adjustments could be made to minimise the impact on throughput. In SystemVerilog we used directed testing methods with randomised data and started by selecting one master, with several slaves, moving on to multi-master test cases. Dedicated functional coverage was built to assure that all NoC configurations are checked within the NoC regression.

Top Level

- **Direct testing methods using randomised data**
- **Multi-master and multi-slave scenarios**
- **Scan Compression**

Block Level

- **Constrained random testing**
- **System Verilog testbench**
- **Regressing with 1000+ seeds**

Block Level

At block level, Sondrel engineers performed the functional and code coverage of the HW accelerators, integrating the reference model into the testbench, building the special test cases with a heavy arithmetic workload, verified the memory controller and enabled all the pipeline stages to prepare for the power simulations. The ultimate target was to prove that all complex computations within the hardware accelerators were achieved at low power.

The methodology used to perform the functional and code coverage was constrained random testing, with a large number of checkers and functional coverage points. Long regressions, often with several thousands of simulation seeds were used in order to make sure that perfect match was achieved between RTL and reference model. Functional and code coverage targets were 100%, but if we were clear on why this was not the case we were able to agree waivers or exclusions.

Verification of the memory controller involved a different technique to that used for the HW accelerators. For the functional targets set for the memory controller we worked exclusively in a SystemVerilog testbench with custom built bus functional models of the interfaces and a programmable memory model in the testbench.

The RTL had to be a 100% match with the reference model. No exceptions would be acceptable for this target.