Enabling Front-End SoC Integration Automation Flows for Large RISC-V Designs

Chouki Aktouf[1], Allen Muuwil[1], Adrien Lecardonnel[1]

[1]Defacto Technologies, 2 Rue Emile Augier, 38000, Grenoble, France

Abstract

This paper presents a front-end design framework for RISC V system on chip designs. The framework manages RTL and design collaterals pre synthesis and provides design engineers with a high level of automation to build and restructure subsystems and full chip. A focus is given to the enablement of RTL design restructuring to cover physically and power aware RTL design requirements.

Introduction

Today, to get higher performance, RISC V designers need to run through multiple iterations of design optimization. In most of the companies, RTL designs are created and optimized using internal solutions (custom scripts) but often times the whole design is done manually. Custom scripts are usually not very re-usable when designs change and most times cannot be ported over from one project to another. This traditional way of developing RTL can be very time consuming and prone to human errors. These errors usually show up late in the design cycle and consume considerable project time.

RISC-V based designs are also becoming more and more complex. To shorten Turn Around Time, System on Chip (SoC) designers need to rely on a fully automated design solution which shortens the path between design specification to output RTL ready for synthesis and simulation.

Defacto RISC-V design solution[2] is a new generation of SoC design platform which combines between a set of configurable IP cores and an integration software with the ability to explore, architecture, integrate complex SoC designs for both synthesis and verification purposes including ASICs and FPGAs.

This new design solution allows users to quickly explore and generate different SoC configurations including RTL and design collaterals starting from their user specifications with a minimal parametrization and intervention. Also, this solution helps design verification teams to build cost-effectively SoC test benches.

Traditionally, when building a complex SoC design project, SoC designers rely on their own design environments based on basic IP blocks and realize the five following steps:

1. IP selection
2. IP configuration
3. Connection of the IPs
4. Generation of all design files for the following SoC implementation and verification flows
5. Validation.

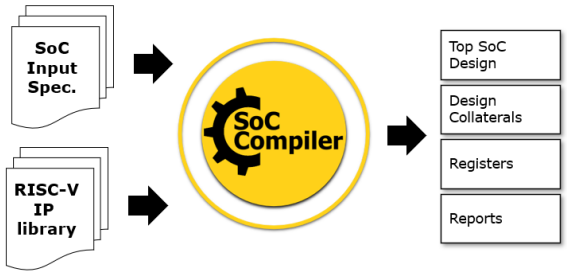

Figure 1: Defacto's SoC Compiler design platform

Several of the above design steps and tasks are still manual and painful for large RISC-V SoC designs, mainly because of the large number of iterations which are usually required before generating top level RTL and design collateral files.

Also today, every SoC design uses hierarchy for physical and logical descriptions. It is very common that, in order to meet physical design implementation constraints re-structuring of the design hierarchy is required. Some of these constraints consist of the need to reduce die size and, increase utilization and improve power consumption Design re-restructuring at RTL level provides a unique solution to explore multiple floor-planning options to find the most optimal one, but this process also causes a change to the design logical hierarchy and may require several iterations of re-design and functional verification.

A New Design Platform

This new design platform provides a library of configurable and up-to-date RISC-V IP cores covering known technologies. SoC Compiler design platform leverages the existing ecosystem of high-quality RISC-V

open-source IPs and subsystems including well-known RISC-V cores such as CVA6[3], Pulp[4], Chipyard[5] etc. Also, it provides for uses the ability to add their own custom IP cores and technologies. Also, the design platform provides a set of SoC design templates, which can be configured/adapted/extended to meet the user's requirements

For an individual IP core or a subsystem, the design solution also provides checking capability such basic linting checks for IP cores and also coherency checks between different views:

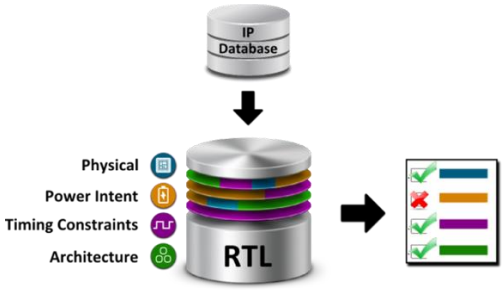

Figure 2: Design Data Consistency Checks

Based on this front-end SoC integration process, the generation of different design configurations is simplified for the end user who just needs to enter Key Performance Indicators (KPI) such as maximum design size area, power consumption, etc. and in return generate RTL design configurations ready for both implementation and design verification flow. Customization possibilities are also provided to increase interoperability with internal design environments and help reach challenging and contradictory KPI objectives. The provided level of automation which is provided helps a user to iterate over several design alternatives until design objectives are met.

Design Restructuring

Another dimension that needs to be considered when integrating SoC in how to restructure a fully integrated system, re-organizing the cell instances and connectivity to optimize the design or fulfill physical design constraints.

Manual design restructuring is very painful. On large SoC designs with several levels of hierarchies, moving blocks or flattening a hierarchical level may take days, in addition to the subsequent and needed verification time.

The SoC Compiler design solution from Defacto allows the user to re-structure an existing design in few quick steps. The tool can read a design netlist, analyzes the current hierarchy and then re-structure it as desired without breaking original design functionalities.

Through automation and rapid hierarchical modification APIs, designers can:

- Fit with the physical requirements by optimizing design structures such as eliminating long loops, loopbacks, and redundant ports, etc.
- Achieve aggressive die size reduction
- Increase design and IP reusability

Using such automation, several Man-Month of work can be saved. Saving weeks of work when complex design allows to better focus on optimizations which enables to fit aggressive PPA requirements that would not be reachable with traditional manual methodologies.

Conclusion & Discussion

With the above summarized capabilities of the Defacto's new design solution, RISC-V SoC designers will be in capacity to manage the increasing complexity of modern SoC design projects.

References

[1] Ulrich Kühne Florian Brandner Tarik Graba Guillaume Duc. *Building a RISC-V Processor*. (https://inf107.telecom-paris.fr/lectures/lecture-part1-riscv.pdf)

[2] Defacto Technologies website, Front-end SoC Integration. (https://defactotech.com/products-solutions/soc-integration-at-rtl)

[3] CVA6 GiTHub repository. (https://github.com/openhwgroup/cva6)

[4] Pulp platform GitHub: (https://github.com/pulp-platform)

[5] Chipyard website. (https://adept.eecs.berkeley.edu/projects/chipyard/)