RISC-V Based Controller for 22nm FDSOI Research Chips

This MSc graduation project will be jointly supervised by the ICD and CAES groups of the EEMCS faculty. For more information, please contact Mark Oude Alink (m.s.oudealink@utwente.nl) or Sabih Gerez (s.h.gerez@utwente.nl).

CMOS technology is generally optimized for digital designs. Traditional analog/RF design often does not benefit from this scaling, and may even get worse due to the reduced supply headroom, but digital techniques can come to the rescue. For example, process variations may be canceled by static digital settings to control e.g. a bias current, ADC and receiver non-idealities may be compensated by offline or online calibrations using mixed-signal feedback and DSP, etc.

Right now, The IC-design group uses UTCONTROL, a rather simplistic shift-register based digital SPI-like interface to configure (semi-)static settings, which has been designed by hand. To enable more advanced functionality in an easy-to-use, modular, and reusable way, we are looking for a RISC-V based control system. We opt for RISC-V instead of e.g. ARM because RISC-V is open source and it has a large and expanding open source code base [1]. We desire a control system because we need more than just those static settings. It should be designed in a way flexible enough to add or remove functionality, but at the same time, it should occupy only a small portion of a test chip. This means the logic should occupy a small area, but it should also require as few bondpads as possible. The target technology Globalfoundries’ 22nm FDSOI technology.

The goal of this project is to develop a tapeout-ready proof-of-concept of this control system. Even when meant to be integrated with all kinds of analog circuitry in the future, the initial design will consist of only the digital core. The initial target is an area of 250umx250um including bondpads, RISC-V processor, 4kB of memory, and compact JTAG interface. When time allows, a test chip may be fabricated and measured, more functionality may be added, etc.

The challenges of the project are:

- Defining a system architecture that satisfies requirements as mentioned above.
- HDL Design of the architecture making use of open-source IP where possible, and its system-level verification.
- Learning and applying synthesis, place-and-route, and backend-verification tools to the degree that the delivered result is (almost) ready for tapeout.