In the past, networks have been composed of closed devices, e.g. routers or switches, which only provide vendor-specific features, forwarding decisions and configuration capabilities. Experimenting with new ideas and properly monitoring such devices is therefore very difficult. Recently, reprogrammable network hardware/ASICs and domain-specific programming languages (P4 [3, 5]) for data-plane programming have emerged. They open a variety of new ways to build, innovate and operator computer networks.

We can test and execute P4 programs in different ways. A simple but limited approach is the “behavioral model” that runs the program in a software switch. The other extreme is the execution on a Barefoot Tofino [1] switch – the world’s fastest P4-programmable switch. Yet another option is the P4-NetFPGA project [4], which provides an environment to develop and run P4 programs on top of NetFPGA SUME [2] FPGA boards.

This thesis explores the possibilities of P4 programming on top of a NetFPGA SUME board. How easily can we deploy a P4 program on the board? What are the differences compared to e.g. a Tofino switch? Furthermore, we would like to exploit the programmability of the SUME board to add a simple feature, e.g. a new hash function. More precisely, the work can be roughly divided into the following work packages:

- **WP1**: Get familiar with the SUME board and its P4-capabilities.
- **WP2**: Run simple P4 programs on the board. What are the differences compared to an execution on a Tofino switch or the behavioral model?
- **WP3**: Extend the existing functionalities of the P4 implementation by reprogramming the FPGA board.

Depending on the student’s interest, we can shift the focus of the work more towards P4 or FPGA programming.

**Requirements**

- Basic knowledge in P4 (e.g. attendance of last year’s advanced topics in communication networks lecture) OR,
- basic knowledge in FPGA programming (e.g. attendance of one of the VLSI lectures)

**Contact**

- Tobias Bühler, buehlert@ethz.ch
- Prof. Dr. Laurent Vanbever, lvanbever@ethz.ch

**References**