Popular science summary

In modern electronic systems, it is common to see the inclusion of one or many ASICs, which are "chips" developed for specific tasks. Inside the ASIC, there are numerous logical modules called IP, which handles some functionality of the ASIC. This thesis explores one such IP at Ericsson called CCR. Clocks are essential to the ASIC and allows for determinism, robustness, synchronisation and data pipelining. Reset signals ensure that the ASIC go to a deterministic state, which is particularly useful when booting up the ASIC. The CCR ensures that other IPs receive stable clocks and reset.

The IPs in an ASIC are designed using hardware description languages. These languages allows the designer to describe how the IP should function, using RTL code. One popular RTL coding language called SystemVerilog includes features to design configurable code. This means that the code can adapt to changes in its specification, without writing new code to describe the new functionality. Configurability can avoid redesign of the IP, which can reduce the development time and reduce human-made errors. This thesis explores the possibility of making the CCR IP more configurable.

The results show that making a configurable IP comes with challenges and trade-offs. One such trade-off is that the readability of the RTL code can be worse when making it configurable. Another challenge is making the design configurable while the ASIC is running, as this can increase the area and power consumption.

The thesis can be used by ASIC RTL design engineers seeking to learn about techniques used and the trade-offs of designing configurable RTL code, such as using software scripts to generate RTL code. The thesis also proposes new implementations that can be used when constructing designs similar to the CCR.