General description and content:

The design and realisation of integrated circuits requires techniques to manage the immense amount of logical components and transistors. While small designs can be realised by a manual approach, this is not feasible for more complex designs any more. They often require a structured way to describe the function and behaviour at a higher level of abstraction. One possible way for dealing with such problems is the use of a hardware description language like VHDL.

The aim of this practical course is a general introduction into the hardware description language VHDL which will be expanded to a deeper view to realise more complex applications. At the beginning the syntax of the language will be discussed in comparison with the desired hardware realisation. The next step is a detailed explanation of the event-driven simulation cycle realised in the simulator of the development tool chain. The remaining part of this lecture deals with the synthesis step. This is the automated transformation of a VHDL-code into a functioning digital hardware solution.

In order to place emphasis on a practical approach we'll use real digital hardware relatively early in the semester. For this purpose a FPGA-based development board is used to test the designs created. This should enable a fast feedback to students about their ability to write code for synthesis purposes. Alternatively, there is a significant focus on the software-based simulation of the complete design to have better suited ways for troubleshooting.

All practical exercises and problems will be expanded by a test-approach to test all components separately and to verify their correct interface characteristics.

Beside of the theoretical knowledge digital concepts will be realised in small examples: implementation of FSMs, clock division concepts, tri-state systems and the examination of the critical path by using the design tools.

Aside from the basic introduction of VHDL optional and advanced techniques could be included to accelerate the design-creation process and to show the advantages of a tool support.

The homework tasks will be based on each other to realise a more complex solution at the end. This final project should demonstrate the functioning of complex digital systems and the modular design approach aims at a gradual familiarisation and deepening of the VHDL-knowledge taught.

The following topics will be held:

- Syntax of VHDL in general
- Simulation cycle and related features
- Special data types and type conversions for direct arithmetic operations
- Special descriptive approaches to generate VHDL-code for synthesis
- Creation of effective test-benches
- Comparison between VHDL description and generated hardware
- Integration of FPGA-based RAM primitives into the design
- Extraction of design parameters (possible clock frequency, etc.)

Requirements:

This practical course provides an introduction into the language VHDL and requires no prior knowledge in this regard. However, a well-founded basic knowledge in digital circuits is required (contents of *Digital circuits* and *Computer architecture 1* or equivalent lectures).

Students who have already completed the course "*Digitaler Hardwareentwurf*", can deepen their VHDL knowledge and deal with it in a practical way.

Objectives:

Students will learn the hardware description language VHDL and should be able to realise simple hardware designs for FPGAs afterwards. Furthermore they should be able to verify digital circuits by the means of simulation and to synthesize a design for a target FPGA technology at the end of this lecture.

In the course exercises will be used to apply the theoretical knowledge to practical problems that will be assembled to a more complex application in small incremental steps. A key focus is set onto the practical work.

Mode:

The course will be held in buffered blocks (mainly weekly or at a 14-day interval) and they include a theoretical part (lecture character held by a lecturer) and a part of exercises, where students have to solve related tasks as homework. The exercises are issued primary weekly with a processing time of 1 or 2 weeks each (depending on their complexity).

The final grade will be a combination of homework and a final project.

The lecture uses the e-learning platform of the JKU, MOODLE, which will serve as a communication medium. All documents, exercises and additional information will be published there for download.

Moreover the submission of the exercises will be done by MOODLE as well.

The enrolment key for the MOODLE-course will be given in the first lecture unit.

Details on the course, additional information and short-term changes will be provided in the MOODLE-course as well.

The lecture can take place if enough students sign in for the course.