The Curriculum of BB-KI Chips

BB-KI Chips members

1. Introduction

Artificial Intelligence (AI) aims to implement systems that can perceive, reason, and act autonomously to mimic or even extend the cognitive capabilities of humans, ultimately enhancing productivity in various application domains. While software aspects of AI are well-represented in current university curricula, hardware concerns remain significantly underdeveloped. Despite corporations such as NVIDIA, AMD, ARM, and Intel launching their AI hardware platforms for developers, AI hardware aspects are still rarely covered in university teaching.

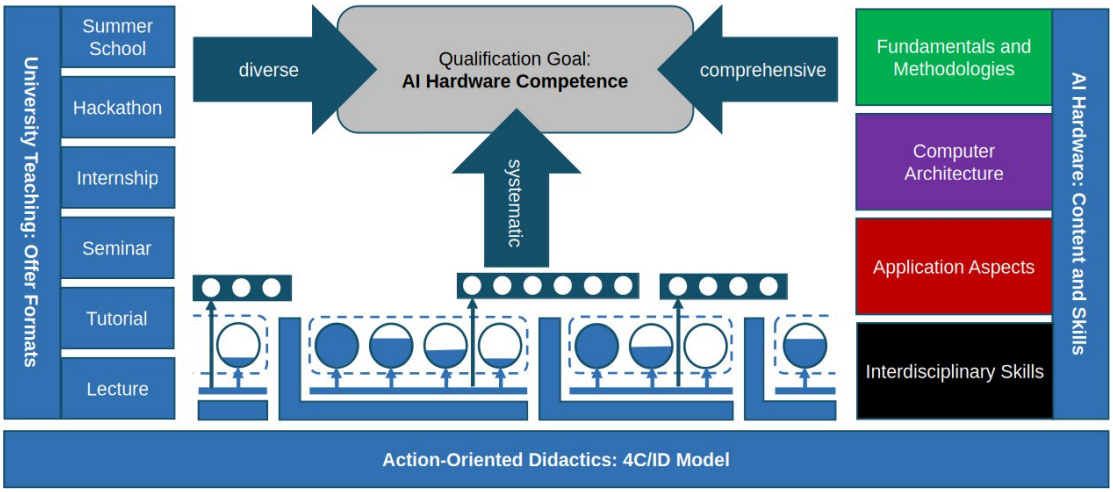

Figure 1: Methodological Approach in BB-KI Chips

To bridge this gap, the Brandenburg/Bavaria Action for KI-Hardware (BB-KI Chips) project incorporates efforts from the University of Potsdam and Technical University of Munich and designs a curriculum focused on AI hardware, including topics ranging from hardware fundamentals to hardware architecture with practical courses involving the utilization and development of AI hardware. Our curriculum follows the 4C/ID model as illustrated in Figure 1, emphasizing practice-oriented learning tasks with increasing difficulty and decreasing support from teachers. The 4C/ID model comprises four components:

- ■ **Fundamentals and Methodologies**: principles, theories, and methodologies relevant to AI hardware.
- ■ **Computer Architectures**: computer architecture and design principles for AI computations.
- ■ **Application Aspects**: practical implementations and real-world applications of AI hardware.
- ■ **Interdisciplinary Skills**: interdisciplinary concerns including ethics, system integration, project management, etc.

Through this curriculum, we provide students with theoretical and practical AI hardware competencies that prepare them for subsequent careers, research, entrepreneurship, and innovation.

1.1 Structure

This curriculum comprises two years of study with 120 credit points (ECTS). One credit point is generally equivalent to 25-30 hours of study, practice, project work, etc., for an average student. The two-year curriculum is divided into the following six module structures:

- Fundamentals – covering fundamental knowledge that is baseline for this study program and including also the courses covering missing background for some orthogonal bachelor studies, outside of computer engineering and electrical engineering focus.
- Architecture and methodology – covering required knowledge of computer and AI architecture, as well as required methodological subjects in the corresponding domain.
- Implementation – covering more advanced topics of the implementation and application of edge AI hardware systems
- Deep dives – very advanced topics and specific seminars enabling specializat ion of the students
- Free electives – free selective courses to complement the individual specializ ation
- Master thesis

Each of the module structure need to be covered with the minimum amount of ECTS points, as defined in this document. The courses and corresponding descriptions in each module will be given in Section 4.3.

1.2 Admission

A bachelor's degree is required to be eligible for admission to this program. This degree must be of equivalent academic level and approximate scientific content as the corresponding German bachelor's degree. In addition, sufficient proficiency in the German/English language is required. The additional requirements are provided in section 1.3.

1.3 Prerequisites

This master study accepts the candidates with various backgrounds, but certain prerequisites need to be fulfilled. In particular the candidates need to demonstrate the knowledge in following domains:

- Linear and Boolean algebra
- Programming languages
- Theoretical background of AI
- Fundamentals of Computer Science

Students may have to repair deficiencies due to different backgrounds. In this context, one or maximally two homologation courses may be assigned to students. The stude nts which need more then two homologation courses are not admissible for this mast er studies. Homologation courses are mostly bachelor-level courses, wherein the corr esponding ECTS will be calculated at 50% value as part of the free electives in this c urriculum.

2. Learning outcomes

The master study should provide to students the following learning outcomes:

- Ability to understand modern AI algorithms and their processing consequence s.
- Understand the architectures for AI processing acceleration
- Adopt the skills for implementing AI-processor in hardware

3. The 4C/ID model

tba

4. Curriculum

The full lists of courses and exemplary study plans are provided in this chapter, follo wed by detailed course descriptions. Given the coexistence of both 5 and 6 ECTS co urses in German universities, two types of curriculum design are proposed with esse ntial adaptations for 5-ECTS and 6-ECTS systems, respectively. The teaching content of both designs remains similar. However, the student workload is greater in 6-ECTS systems.

4.1 List of courses

This curriculum comprises 120 ECTS, which is consistent with most master's pro- grams in Germany.

Course	Emphasis	ECTS
Fundamentals		*min*: 20/24 ECTS
Artificial Intelligence Fundamentals	■	5/6
Computer Science Fundamentals	■	5/6
Digital Hardware Fundamentals	■	5/6
Embedded System Design Fundamentals	■	5/6
Ethics Fundamentals	■■	5/6
Research Data Management	■	5/6
Architecture and methodology		*min*: 15/18 ECTS
Hardware-Software Co-Design	■	5/6
Introduction to HDL and Tools	■	5/6
Development and Integration of HW Accelerators	■	5/6
Reliable Hardware: From Logic Gates to Processors	■	5/6
Hardware Architectures for AI	■	5/6
Implementation		*min*: 20/18 ECTS
Accelerating CNNs using PL	■	5/6
Electronic Design Automation	■	5/6
Chip Design	■	5/6
Engineering Project	■	5/6
Lab Course Mobile Computer Vision	■	5/6
Deep dives		*min*: 15/12 ECTS
Ethics Advanced	■	5/6
Lab Course Photogrammetric Data Acquisition	■	5/6

HDL Projects	■	5/6
Neuromorphic Chip Design	■	5/6
Applied Machine Learning for Natural Hazards and Environmental Changes	■	5/6
Free electives		*max*: 20/18 ECTS
Master Thesis		30 ECTS

Table 1: The list of courses for 5-ECTS/6-ECTS systems.

4.2 Exemplary study plan

Following the course list in Table 1, example schedules for 5-ECTS/6-ECTS systems are illustrated in Figure 2 and Figure 3, respectively.

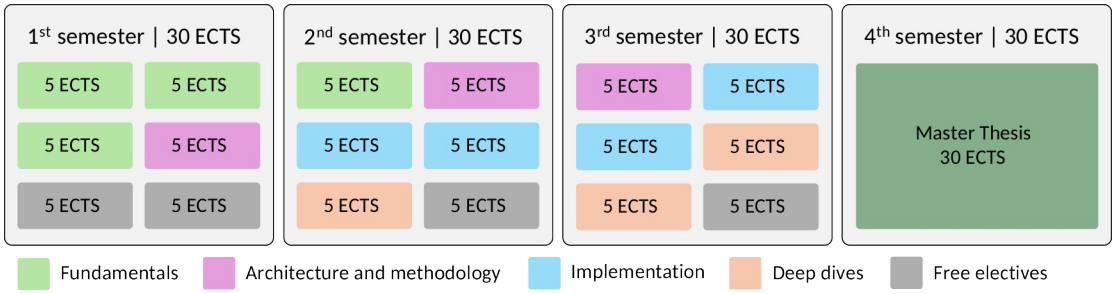

Figure 2: An example schedule for 5-ECTS systems.

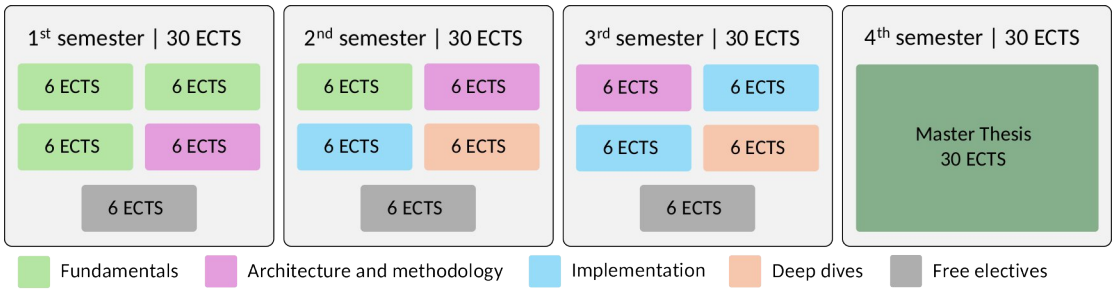

Figure 3: An example schedule for 6-ECTS systems.

4.3 Course description

Every module in the curriculum is elaborated in this section, including the courses within each module and their descriptions.

4.3.1 Fundamentals

This module provides essential knowledge for AI hardware engineering. Students learn the core ML/DL concepts and algorithms that drive modern AI systems, understanding their computational requirements and implementation challenges. These courses establish the foundation for the challenges at the intersection of hardware design and artificial intelligence.

Artificial Intelligence Fundamentals

Content: tba

Learning outcomes:

Assessment:

Computer Science Fundamentals

Content:

Learning outcomes:

Assessment:

Digital Hardware Fundamentals

Content: The course focuses on the fundamentals of digital electronics that are essential to understanding the design and working principles of a wide range of applications, from consumer and industrial electronics to communications; from embedded systems and computers to security and military equipment. As devices used in these applications decrease in size and employ more complex technology, it is essential to understand both the fundamentals and the implementation and application principles of digital electronics, devices, and integrated circuits, thus enabling them to use the most appropriate and effective techniques to suit their technical needs. The course covers the following topics:

- Binary arithmetic, logic gates, and Boolean algebra,
- Combinational logic circuits (adders and subtractors, multipliers, carry lookahead unit, arithmetic logic unit, multiplexers, demultiplexers, encoders, and decoders),
- Programmable logic devices (SPLDs, CPLDs, and FPGAs),
- Sequential logic circuits (flip-flops, counters, and registers),
- Digital-to-analogue and analogue-to-digital converters,
- Memories (ROM, SRAM, DRAM, and RRAM),
- Processors, interfaces, and peripherals.

Learning outcomes: Students are expected to function knowledgeably in the area of digital circuits and systems. They will be able to cross the gap between digital system design, verification, and implementation.

Assessment: The performance evaluation is based on an oral final exam.

Embedded System Design Fundamentals

Content:

Learning outcomes:

Assessment:

Ethics Fundamentals

Content:

Learning outcomes:

Assessment:

Research Data Management

Content:

Learning outcomes:

Assessment:

4.3.2 Architecture and methodology

This module provides deep expertise in designing and implementing specialized computing systems for AI workloads. Students must select at least three courses from this module. Through these courses, students develop practical skills in hardware-software co-design, hardware description languages, simulation tools, and design methodologies while gaining theoretical knowledge of performance optimization, power efficiency, and system-level integration competencies for creating AI hardware platforms.

Hardware-Software Co-Design

Content:

Learning outcomes:

Assessment:

Introduction to HDL and Tools

Content: Introduction to Hardware Design Languages and Tools is an elective course that builds on prior knowledge in computer architecture and programming to introduce students to digital hardware design using Verilog. It covers the fundamentals of combinational and sequential logic, including components such as flip-flops, counters, and state machines, alongside testbench-based verification. The course emphasizes practical, hands-on experience. Students implement their designs on the Zynq 7-series development board, progressing from simulation to real FPGA hardware implementation. This approach provides a clear understanding of the hardware design workflow and exposes students to industry-relevant tools and methods. Through individual and group work, students develop technical skills and problem-solving abilities essential for careers in digital and embedded system design. The detailed activities are composed as follows:

- Digital circuits fundamentals in gate level (CircuitVerse playground).
- Develop Verilog modules using structural and behavioral modeling.
- Design and simulate logic units (e.g., multiplexers, ALUs, FSMs) with open-source tools (Iverilog, Yosys, and GTKwave).
- Design and implement modules of different complexities (e.g., UART, single-cycle RISC-V processor) with vendor-specific tools (AMD/Xilinx Vivado).
- Program FPGA boards (PYNQ-Z2) and verify real-time functionality.

Learning outcomes:

Assessment:

Development and Integration of HW Accelerators

Content: Dedicated Hardware (HW) Accelerators have become a popular appr oach to increase computational performance and energy efficiency. This has le ad to an explosion of different architectures, leading to new challenges in integr ating accelerators into existing systems. The goal of the seminar is to explore a pproaches in integrating and developing new domain specific architectures and accelerators, as well as studying existing accelerators. Students prepare prese ntations on different topics including:

- Hardware Development using open source tools and new languages
- Hardware/Software Interface and integration
- AI accelerators and architectures
- Novel Domain Specific Architectures such as Processing in Memory and Quantum Computing

Learning outcomes: Students should be able to independently explore and un derstand scientific literature in the domain of hardware accelerators. They shoul d have an overview of different directions of Hardware Accelerator research.

Assessment: The performance evaluation is based on a written report and a pr esentation.

Reliable Hardware: From Logic Gates to Processors

Content: Electronic systems composed of embedded processors and memories are everywhere from household appliances via transportation systems to utilities and production plants. Failures of such systems, in many areas of application, can cause human life and huge financial losses. Hence, making such systems reliable has become a major issue in science and technology. Typically, the reliability refers to specific hardware units and their faulty behavior. Dependability is somehow the top-level term incorporating all safety and reliability aspects. If applied to hardware, the dependability typically includes system extensions that can detect, correct, and tolerate errors. We can assume that a system is highly dependable, if it is fail-safe. The detailed list of course topics includes:

- Defects, faults, errors, and failures,
- Yield modeling and analysis,
- Yield forecast and control,
- Redundancy and fault tolerance,
- Error detection and correction,
- Built-in self-repair,
- Dependability,
- Error resilience.

Learning outcomes: Students are expected to understand and analyze reliability aspects of digital hardware. They should be able to develop simple solutions for reliability issues in digital hardware.

Assessment: The performance evaluation is based on an oral final exam and student projects.

Hardware Architectures for AI

Content: In this course, the focus will be on the specifics of hardware design and architecture for AI applications. After the overview of the standard design techniques and common computing architectures, the additional requirements of AI will be discussed. Based on this, the specific architecture and design methods increasing the efficiency of the computation will be discussed. Finally, this course will also include an introduction to the emerging and novel architecture and technologies that could have a significant impact in the future. Here is the detailed list of topics:

- Introduction in VLSI and digital logic design
- Hardware design process, ASIC, FPGA

- State-of-the-art processor architecture, Example RISC-V
- Limitations of classical architectures for AI applications
- Domain specific architectures: Vector processors, GPUs
- Hardware specifics of AI processing
- Deep learning accelerators
- Emerging architecture: In-memory-computing (RRAM example)

This course includes practical examples and requires active student work.

Learning outcomes:

Assessment:

4.3.3 Implementation

The implementation module provides hands-on experience in hardware design and development for AI applications. Students are expected to gain practical skills essential for AI hardware engineering. These courses collectively ensure graduates have practical skills for both industry and research.

Accelerating CNNs using PL

Content: High-Level Synthesis (HLS) provides a more abstract way to design d igital circuits. It provides high level verification using programming languages su ch as C++. The course teaches students the required knowledge to accelerate existing algorithms running on CPUs using FPGAs and HLS. The course deals with the following topics:

- Basic building blocks of FPGAs
- High Level Synthesis Scheduling and Binding
- Vitis HLS design flow
- Domain Specific Accelerators
- Integration with CPU and Memory
- Introduction to Convolutional Neural Networks (CNNs)
- Project in Accelerating CNN inference using SoC FPGAs

Learning outcomes:

- Basic understanding of Convolutional Neural Networks
- Understanding the challenges of using FPGAs to accelerate workloads
- Ability to design digital circuits using HLS languages
- Implementation and Integration of both SW and PL on a SoC platform
- Ability to reason about the performance of different implementations

Assessment: The labs will be done in small groups (max. 3 students) and cons ists of minor non-grade labs, as well as a mid-term report. The final grade will b e based on the Project including a Report, Presentation and an individual discu ssion of the implementation.

Electronic Design Automation

Content: Electronic Design Automation (EDA) assumes the acquisition of a digital design from the architectural level to the layout. It describes how to describe, synthesize, layout, verify, test, and reuse digital designs. This module provides students with a starting point for each stage of the design process. EDA tools are very sophisticated and complex software programs that function primarily in one

of three ways to assist with the design and manufacture of integrated circuit chips:

- Simulation tools take a description of a proposed circuit and predict its behavior before it is implemented.
- Design tools take a description of a proposed circuit function and assemble the collection of circuit elements that implement that function. This is both a logical process (assembling and connecting the circuit elements) and a physical process (creating the interconnected geometric shapes that will implement the circuit during manufacturing). These tools are delivered as a combination of fully automated and interactively guided capabilities.
- Verification tools examine the logical or physical representation of the chip to determine if the resultant design is connected correctly and delivers the required performance.

Learning outcomes: Students are expected to understand EDA algorithms and use EDA tools for simulation, logic and layout synthesis, verification, and test. They should be able to implement simple modifications of the existing design flows and tools.

Assessment: The performance evaluation is based on an oral final exam and student projects.

Chip Design

Content: When designing AI systems, the interaction of software and hardware components is very important. The basis for hardware design is the understanding of various circuit concepts and design methods. In this context, the course will introduce the concept of synchronous design. Based on this, the development process of hardware systems is to be understood using ASIC and FPGA implementations as examples. The aim of this course is to increase the students' understanding of hardware design. Here is the detailed list of topics:

- Introduction, VLSI design
- Hardware design principles
- Advanced VHDL for logic synthesis
- ASIC design flow (logic synthesis, layout, verification)
- FPGA design
- Chip manufacturing process and chip test

This course includes practical examples and requires active work by the students. Through this course, students will be able to design a chip that will actually be produced. You would also get an overview of the chip production. Finally, the students can test the actually produced chip.

Learning outcomes:

Assessment:

Engineering Project

Content: Engineering topics in the context of ongoing AI hardware projects, with a focus on the development of deep learning models and their deployment on board. The supervisor will provide topics and in-person instructions/plans as students usually deal with the project individually. Students are also welcome to

propose topics on their own. During the first several weeks, the supervisor will discuss with students the scope and goals of the project and design a series of milestones with fundamental theoretical units for students to be familiar with background knowledge. Then, obligatory presence is not required. Meetings only happen when necessary. There is no previous knowledge expected, but the course combines well with taking the lectures on Computational Foundations and an interest in deep learning. A UAV license (EU drone license A1/A3) is required for UAV operations.

Learning outcomes: Students are expected to understand the motivation and popular solutions for edge AI computing. Students should be able to design solutions for AI tasks with methods of hardware and software engineering.

Assessment: Students need to prepare an interim presentation and a final report and code repository. The final grade is determined by the quality of both the code and the report, as well as the fulfillment of the project objectives.

Lab Course Mobile Computer Vision

Content: In this lab course, students develop a first understanding of computer vision models based on neural networks and are supposed to learn the tools involved including

- Python programming
- Data mining problems and scikit-learn
- Deep learning and TensorFlow
- Convolutional Neural Networks in practice
- FPGAs, GPUs, and modern SoCs
- Methods for accelerated Deep Learning Inference

This background is learned during the semester with weekly assignments (videos, book chapters) followed by a weekly discussion and question session (e.g. inverted classroom). The lab course is composed of an inverted classroom phase of knowledge acquisition: from publicly available sources, students learn independently at their own paces. The course comes together in weekly online meetings to discuss open questions and keep synchronized.

Learning outcomes: Students develop a first understanding of the concepts behind neural networks and neural computer vision as well as on the techniques and difficulties of executing such models on embedded devices.

Assessment: Teams of no more than five students subsequently implement their course-prepared computer vision system (this can be conducted online) and deploy it to real embedded hardware suitable for deployment on drones or satellites. To ensure adequate individual supervision and access to required hardware and computational resources, a limited number of slots are available. Each team must prepare and present a poster during the final poster session and demonstrate their system on site to supervisors and other teams.

4.3.4 Deep dives

The curriculum also includes courses that deepen students' knowledge while developing critical analysis and presentation skills.

Ethics Advanced

Content:

Learning outcomes:

Assessment:

Lab Course Photogrammetric Data Acquisition

Content: In this lab course, students learn about photogrammetric data acquisition for 3D scene reconstruction and analysis, with a focus on UAV photogrammetry. The course entails the full workflow with

- Planning of the acquisition
- Execution of the mission (incl. reference measurements, e.g., via GNSS)
- Data processing: Photogrammetric reconstruction of 3D models
- Analysis of a small research objective tailored to the application/study site

The workflow will be put to practice as data acquisition projects with small analysis objectives developed for the application/study site of the course.

The lab course consists of a preparatory inverted classroom phase with a strong component of self-learning using resources made available by the lecturers. Regular live sessions will take place for questions and discussions.

In the practical phase, teams of 3-5 students work on a photogrammetric data acquisition project (independently, with supervision from the lecturers). Deployment will take place on a joint field workday at the selected study site.

Participants should have basic knowledge of geospatial data (e.g., Introduction into Geodesy). Interest in 3D Earth observation topics and motivation to learn new tools and methods is expected. A UAV license is required for data acquisition (EU drone license A1/A3).

Learning outcomes: Students will acquire basic knowledge of photogrammetric data acquisition in theory and practice and become familiar with applications especially for topographic environmental observation. They will be able to conduct the full workflow of acquisition planning, image acquisition and 3D reconstruction, and learn to apply some state-of-the-art analysis tools. They will gain competence in teamwork for successful project planning and execution.

Assessment: The exam includes both the final presentation and a written report (6 - 8 pages, following standard technical/scientific format), which are based on data collection, processing, analysis, and results during the group project work. Participation in fieldwork is the prerequisite to the examination. The written report could use the word template from the ISPRS authors' guidelines. All work can be performed in groups of up to 6 people. Please include a statement if the tasks were shared equally, or if there was some focus by one or the other persons, e.g., the light version of CRediT.

HDL Projects

Content: Project seminar/praktikum where some hardware design projects will be implemented. A simple CNN/SNN application example from partner could be used as implementation target (in ASIC or FPGA flow).

Learning outcomes:

Assessment:

Neuromorphic Chip Design

Content: As the next generation of neural networks, Spiking Neural Networks feature powerful and energy-efficient computation. In combination with hardware characterized by specifically optimized architecture for SNN workloads, further improvement of computation and energy efficiency of SNN can be achieved.

This seminar focuses on the design of specific hardware for spike-based information processing, and will cover the following topics:

- Biophysical background for Spiking Neural Network: human brains, cerebral cortex, neurons, action potential, synaptic transmission, Hebbian learning, neural encoding/decoding, receptive field, lateral inhibition, long-term regularization
- Spiking Neural Network Modeling: image-to-spike conversion (rate coding, temporal coding), neuron models (Hodgkin-Huxley model, Izhikevich model, LIF model), STDP, Winner-Take-All, Spiking Threshold Adaptivity
- Specific hardware design for SNNs: digital circuit and Verilog HDL basics, RTL modeling of neurons, synapses, and SNNs

Some paper reviews and programming tasks are included during the seminar. A technical report summarizing your own understanding of SNNs and relevant specific hardware is required at the end of the semester.

Learning outcomes:

Assessment:

Applied Machine Learning for Natural Hazards and Environmental Changes

Content: This course explores the intersection of artificial intelligence, remote sensing, and natural hazards. Students will learn the basics of Geographic Information Systems (GIS), remote sensing, and will apply machine learning and computer vision to analyze satellite and UAV data in the context of natural hazards like floods, landslides, wildfires, droughts, etc. The course introduces key geospatial AI workflows, including multi-source data use, time series analysis, and change detection. Emphasis is placed on detecting and classifying hazard events using satellite data and drone imagery.

Through a hands-on project, students will integrate data from diverse sources (satellites, UAVs, in-situ) and automate the analysis of large spatial datasets to automatically map potentially hazardous or affected areas. The course covers spatial data processing, environmental modeling, and the integration of AI tools (e.g., convolutional neural networks, object detection) in geospatial analysis.

Some of the topics covered by the course:

- Foundations of Natural hazards and risk.
- Basics of GIS and Remote sensing for hazards
- Image classification and segmentation for hazard zones

Some of the possible projects include:

- Automatic Landslide detection using Satellite imagery
- Flood or wildfire burn area classification from UAV imagery
- Debris flow runout modelling from remote sensing images
- Real time damage quantification after earthquakes based on UAV imagery
- Multi-hazard modelling

Learning outcomes:

Assessment:

4.3.5 Free electives

The curriculum offers up to 20/18 ECTS of free electives, allowing students to personalize their educational choices according to their interests and career goals. Students may select courses from any department within the university, allowing cross-disciplinary exploration in fields such as languages, business, design, additional engineering disciplines, or advanced mathematics. This flexibility encourages students to develop a unique profile that complements their core AI hardware expertise.

4.3.6 Master thesis

The final master project of 30 ECTS should be research oriented. The practical execution may be performed in industry or a research institute in Germany or abroad, a s long as a professor at the university agrees to supervise it.

5. Assessment

Student performance is assessed through various methods, including written examinations, project reports and (poster) presentations, research reports and articles, and literature reviews, and programming assignments. To successfully obtain a degree, a student must:

- Finish all modules in the curriculum.
- Successfully complete and defend the master thesis.
- Complete the program within the maximum allowed time frame (typically 4 years).

6. Career prospects

The BB-KI curriculum especially prepares students to work on different abstraction levels in AI hardware design. Careers in AI Hardware design range from low-level electronic engineering to high-level software design. The program prepares graduates for careers such as:

- **VLSI Engineer**: Designing and Verification of integrated circuits. Tapeout and post-silicon validation of AI accelerators.
- **EDA Tools**: Development and Maintenance of EDA tools, including innovative integration of AI into the EDA design flow
- **Digital and Circuit Design**: Development of AI accelerators and IP blocks on a Register-Transfer Level. Implementation and emulation of accelerators using FPGAs.
- **Systems Engineer**: High-level development of system specifications for different components on an SoC, including AI accelerators, CPUs, memory, and interconnects.
- **Compiler and Software Engineer:** Graduates can work on software stacks and compilers essential to successfully bring an AI accelerator to market.
- **Embedded Systems Engineer**: Graduates can work in the growing field of embedded systems, advancing IoT and edge devices with energy-efficient AI hardware solutions.
- **Academic Research**: Graduates are prepared for post-graduate academic work in any of previously mentioned fields.

Graduates will have an advantage in starting careers with a focus on the application domain of AI. However, the fundamental components of this curriculum provide graduates with the necessary base knowledge to work in different application domains of hardware development such as Computer Engineering or Software Engineering.