

Die Modulbeschreibung sollte direkt über diesen [Link](#) in HISinOne eingepflegt werden.

Module code	Module title	Category
MAIE1040	Embedded Software Engineering	MA
	Degree program	MA Software Engineering
	Faculty	Building Services Engineering and Computer Science

Module coordinator	Prof. Dr. Tamas Harczos
Module type	Compulsory elective
Frequency	1x annually in SuSe
Recommended semester	1. semester
Credit (ECTS-Points)	5
Academic Assessment Method	Coursework requirement PrP = Project / Assignment with presentation SB PrP
Teaching language	English
Admission requirements for this Module	none
Module duration	1 Semester
Required Registration	Students enrolled in the above-mentioned degree program/standard semester will be registered automatically upon re-enrollment; all other participants, please refer to the information below. none

	Course	Lecturer	Type	Group Size (max.)	Number of Groups	Contact hours per week (SWS)	Workload (in h)	
							Face-to-face	Self-study
1	Embedded Software Engineering	Harczos	Lecture	30	1	2	30	35
2	Embedded Software Engineering	Harczos	Practice session	15	2	2	30	30
3	<small>Teil der Lehrveranstaltung</small>	<small>DozentIn</small>	<small>Wählen Sie ein Element aus</small>		<small>Wählen Sie ein Element aus</small>			
4			<small>Wählen Sie ein Element aus</small>					
5	<small>Teil der Lehrveranstaltung</small>	<small>DozentIn</small>	<small>Wählen Sie ein Element aus</small>		<small>Wählen Sie ein Element aus</small>			
Sum						4	60	65
Total Workload for Module							125	

Learning Objectives / Learning outcomes	<p>After completing the module, students will be able to:</p> <ul style="list-style-type: none"> • explain basic principles of energy efficient computing and its technological foundations; • analyze and compare different hardware architectures and programming paradigms; • evaluate the trade-offs between computational accuracy, performance, and energy consumption;
--	--

	<ul style="list-style-type: none"> • develop basic embedded software that avoids energy-wasting and exploits interrupts, sleep modes, and event-driven paradigms; • collaborate in interdisciplinary teams to design and implement small-scale prototypes for IoT and edge devices; • critically reflect on security, reliability, and sustainability challenges in ULP systems; • assess emerging paradigms such as spiking neural networks and quantum computing.
Contents	<ul style="list-style-type: none"> • ULP fundamentals and metrics • Minimal-energy architectures and memory systems • Heterogeneous platforms and accelerators • Rechargeable batteries: types and properties • Energy harvesting and energy-neutral design • Smart sensing and compressed sensing • Energy-aware programming: sleep, interrupts, RTOS • Power profiling and power modeling • IoT protocols, low-power networking, edge/fog/cloud offloading • Sparse/spiking neural networks, neuromorphic systems, quantum computing • Security, reliability, measurement, and case studies
Literature	<ul style="list-style-type: none"> • S. Tayal, A. K. Upadhyay, S. B. Rahi, Y. S. Song (editors): Advanced Ultra Low-Power Semiconductor Devices: Design and Applications • R. Zhang and J. Yu: Energy-Efficient Algorithms and Protocols for Wireless Body Sensor Networks • S. Naifar, O. Kanoun, C. Trigone (editors): Energy Harvesting Technologies and Applications for the Internet of Things and Wireless Sensor Networks • K. S. Mohamed: Neuromorphic Computing and Beyond: Parallel, Approximation, Near Memory, and Quantum