

<b>Modulbezeichnung: Computertechnik 2</b>	
<b>Modulkürzel</b>	t.BA.XX.CT2.10HS
<b>ECTS Credits</b>	4
<b>Unterrichts- und Prüfungssprache</b>	Deutsch
<b>Verantwortliche OE</b>	InES
<b>Modulverantwortung</b>	Andreas Rüst
<b>Rechtliche Grundlagen</b>	Die Modulbeschreibung ist neben Rahmenprüfungsordnung und Studienordnung Teil der Rechtsgrundlage. Sie ist verbindlich. Eine in der ersten Unterrichtswoche des Semesters schriftlich festgehaltene und kommunizierte Modulvereinbarung kann die Modulbeschreibung präzisieren. Die Modulvereinbarung ersetzt nicht die Modulbeschreibung.
<b>Modulprägung</b>	Typ 3a  2 Lektionen Vorlesung pro Semesterwoche und Klasse + 2 Lektionen Praktikum pro Semesterwoche und Halbkasse
<b>Beschreibung des Moduls</b>	Das Modul umfasst Speicherhierarchien mit ihren Auswirkungen auf die Software. Zudem lernen die Studierenden die Funktion und Anwendung von grundlegenden Peripherieschaltungen. Softwaretechniken zur Strukturierung und Ablaufsteuerung von Programmen bilden einen weiteren Schwerpunkt.
<b>Inhalte des Moduls</b>	<p><b>Das Modul zeigt den Ausbau eines Prozessors zu einem vollständigen Computer System. Dies umfasst einerseits Speicherhierarchien und Cache mit ihren Auswirkungen auf die Software. Andererseits lernen die Studierenden die Funktion und Anwendung von grundlegenden Peripherieschaltungen. Softwaretechniken zur Strukturierung und Ablaufsteuerung von Programmen bilden einen weiteren Schwerpunkt.</b></p> <p><b>Vom Prozessor zum System</b></p> <ul style="list-style-type: none"> <li>• Aufbau eines Microcontrollers</li> <li>• Funktionsweise Systembus</li> <li>• Memory Map und Adressdekodierung</li> <li>• Funktion und Verwendung von Kontroll- und Statusregistern.</li> </ul> <p><b>Einsatz von Peripherieschaltungen</b></p> <ul style="list-style-type: none"> <li>• Anschluss von IOs über GPIO</li> <li>• Serieller Datentransfer: UART / SPI / I2C</li> <li>• Timer / Counter und ihre Anwendungen</li> <li>• Analog Digital Converter (ADC)</li> <li>• Hardware Abstraction</li> </ul> <p><b>Speicherhierarchien</b></p> <ul style="list-style-type: none"> <li>• Speichertechnologien: Volatile (SRAM, DRAM) vs. non-volatile (ROM, Flash, EEPROM)</li> <li>• Anschluss von externen Speicherbausteinen</li> <li>• Cache: Temporal and Spatial Locality, Auswirkungen auf die Software</li> </ul> <p><b>Ablaufsteuerung von Programmen</b></p> <ul style="list-style-type: none"> <li>• Strukturierung eines Programmes in Module</li> <li>• Software State-Machines (State-Event Model)</li> <li>• Erkennen von Events: Polling vs. Interrupt-driven IO</li> <li>• Interrupt Performance und Latency</li> </ul> <p><b>Hardwarenahe Programmierpraktika</b></p> <ul style="list-style-type: none"> <li>• Arbeiten mit Cross-Compiler, Linker, Loader und Debugger</li> </ul>
<b>Vorkenntnisse</b>	Das Modul setzt den Stoff aus Computertechnik 1 (CT1) voraus.

## Modulbezeichnung: Computertechnik 2

<b>Lernziele (Kompetenzen)</b>	<b>Die Studierenden...</b>		<b>Kompetenzen</b>	<b>Taxonomiestufen</b>		
	Sie verstehen den Aufbau und die Funktionsweise eines Microcontrollers. Sie können Zeitverlaufdiagramme eines Systembusses interpretieren, Adressdekodierungen verstehen und in eigenen Programmen auf Kontroll- und Statusregister zugreifen.		M, F	K2, K3		
	Sie kennen grundlegende Konzepte zur Ablaufsteuerung eines Programms. Sie können eine Problemstellung als Software State-Machine modellieren und umsetzen. Sie kennen Methoden zur Detektion von Ereignissen und können diese bzgl. Performance und Latenz analysieren.		F, M	K2, K3, K4		
	Sie können eine Übersicht über die wichtigsten Speichertypen geben, externe Speicherbausteine anschliessen und eine typische Speicherhierarchie erklären. Sie können die Aufgaben und die Funktion eines Caches erklären und die Auswirkungen auf Software abschätzen.		F, M	K2, K3		
	Sie können die Funktionen grundlegender Peripherie von Microcontrollern beschreiben und können diese in eigenen Programmen sowohl direkt als auch mittels Hardware Abstraction Layer verwenden. Sie können externe Peripherie mittels serieller Kommunikation (UART, SPI, I2C) ansteuern.		M, F	K2, K3		
<b>Leistungsnachweis</b>	<b>Modulendprüfung</b>	<b>Bewertung</b>	<b>Dauer (Min.)</b>	<b>Gewichtung</b>	<b>Form</b>	
	schriftliche Prüfung	Note	90	70	gem. Modulvereinbarung	
	<b>Leistungsnachweise während dem Semester</b>		<b>Bewertung</b>	<b>Dauer (Min.)</b>	<b>Gewichtung</b>	<b>Form</b>
	schriftliche Prüfung		Note	45	15	gem. Modulvereinbarung
	Praktika mit Präsentationen		Note		15	gem. Modulvereinbarung
<b>Präsenzverpflichtung im Kontaktstudium</b>	Keine Die Abgabe der Praktika erfolgt vor Ort in den Praktikumslektionen.					
<b>Lernmaterialien</b>	<ul style="list-style-type: none"> <li>• Data Sheets und User Manual</li> <li>• Übungen</li> <li>• Praktikumsanleitungen</li> <li>• Vorlesungsfolien</li> </ul>					

## Modulbezeichnung: Computertechnik 2

### Bemerkungen

**Obligatorische Klausur:** Termin während des Semesters gemäss Vorgabe der Dozierenden. Abmeldungen sind nur in vorgängig begründeten Ausnahmefällen oder in Notfällen mit Arztzeugnis möglich. In solchen Fällen wird die Klausur nachgeholt. Die Nachprüfung kann mündlich erfolgen. Bei unentschuldigtem Fernbleiben wird die Klausur mit der Note 1 bewertet.

**Praktikumsbewertung:** Benotung gemäss erfolgreich absolvierter Praktika. Praktikumlösungen müssen vorgeführt und erklärt werden.

**Semesterendprüfung:** Schriftliche Prüfung oder Moodle, Open book, ohne generative AI. Geprüft wird der in Theorie und Praktika behandelte Stoff.