

Eingebettete Systeme (ES)

<i>Modulbezeichnung / Kürzel</i>	Eingebettete Systeme (ES)	Stand: 25.05.2007
<i>Fachsemester</i>	5. Semester	
<i>Modulverantwortliche(r)</i>	Prof. Dr. Georg Hartung	
<i>Sprache</i>	Deutsch	
<i>Lehrformen / SWS</i>	Seminaristischer Unterricht mit Praktikum, V2, Ü1, P1	
<i>Kreditpunkte</i>	5 ECTS-Punkte	
<i>Arbeitsaufwand</i>	60 h Präsenz 90 h Selbststudium	
<i>Voraussetzungen</i>	IN 1 / IN 2: Programmieren in Java und C RHP Kenntnisse über Rechneraufbau und hardwarenahe Programmierung in C BVS1 (Kenntnisse in Multitasking/threading)	
<i>Lernziele/Kompetenzen</i>	<p>Die Studenten kennen Verfahren zum Entwurf eingebetteter Systeme (ES) mit bewährten Methoden und besitzen erste Erfahrungen in ihrer Anwendung.</p> <p>Sie besitzen Kenntnisse und Erfahrungen in der Programmierung von ES ohne/unter einem Realzeit-Betriebssystem in gängigen Programmiersprachen (Assembler, C).</p> <p>Sie besitzen Kenntnisse über Standards für Betriebssysteme für ES und können sie beurteilen.</p> <p>Sie wissen, wie verteilte eingebetteter Systeme aufgebaut werden und wie die Software-Komponenten in solchen Systemen miteinander kommunizieren.</p>	
<i>Inhalt</i>	<p>Beschreibungsverfahren für eingebettete Systeme: Strukturierte Analyse mit Echtzeiterweiterungen, Ausführbare Modelle (gefärbte Petri-Netze), UML für eingeb. Systeme</p> <p>Aufbau von ES: Aufbau eines typ. ES mit Mikrocontroller; Arbeitsweise (Polling, Interruptgesteuert)</p> <p>Vor- und Nachteile gängiger Programmiersprachen für ES: Programmierung in Assembler und C mit praktischen Übungen, Vergleich der Programmierung in Assembler, C, C++ und Java und Zuordnung zu Klassen von ES</p> <p>Betriebssysteme für eingebettete Systeme: Begriff des Realzeit-Betriebssystems; Prozesse und Threads; Prozessverwaltung mit Scheduler, Prozesssynchronisation und -kommunikation; OSEK/VDX und POSIX(-RT) als Beispiele</p> <p>Ablaufplanung für ES: Verfahren zur Sicherung der "schritthalten" Verarbeitung in ES; Worst-Case-Betrachtungen; Scheduling-Verfahren in Realzeit-Betriebssystemen</p> <p>Embedded Linux: Aufbau von Embedded-Linux-Systemen, Kernelaufbau, Treiberprogrammierung am Beispiel, Echtzeitproblematik und ihre Lösung in RTAI</p> <p>Verteilte eingebettete Systeme: Grundbegriffe Verteilte Systeme; ISO-OSI-Schichtenaufbau; Proto-</p>	

	<p>kolle und Schnittstellen; Typische Ausprägungen im eingebetteten Bereich; Feldbusse als Kommunikationsbasisstruktur für verteilte ES</p> <p>Programmierung verteilter ES: Nachrichtenbasierte, Netzwerkvariablen- und RPC-basierte Kommunikationsmittel und ihre Benutzung; OSEK/VDX Kommunikationsmittel</p>
<i>Studien-/Prüfungsleistungen</i>	<p>Im Praktikum wird in einem Projekt ein Prototyp eines ES mit Mikrocontroller und selbstgebaute Modell erstellt. Das Ergebnis und die Zwischenberichte werden bewertet und fließen in die Endnote ein.</p> <p>Mündliche Prüfungen (in Ausnahmefällen auch schriftlich): dreimal jährlich</p>
<i>Medienformen</i>	<p>Ausformuliertes Skript (in der Vorlesung erläutert)</p> <p>Demonstrationen von Werkzeugen, Beispielsystemen und Simulationen</p> <p>Tafel zur interaktiven Entwicklung von Konzepten</p>
<i>Literatur</i>	<p>Wayne Wolf: Computers as Components: Principles of Embedded System Design</p> <p>M. Barr: Programming Embedded Systems in C and C++</p> <p>J. Lemieux: Programming in the OSEK/VDX environment</p> <p>P. Laplante: Real Time Systems Design and Analysis</p> <p>J. Wietzke/M.T. Tran: Automotive Embedded Systeme</p> <p>Ausführliche Literaturhinweise werden im Skript verteilt.</p>