

Mikrocontroller-Systeme

Mikrocontroller-Systeme
Modulübersicht
EDV-Bezeichnung: EITB250
Modulverantwortliche(r): Prof. Dr. Christian Langen
Modulumfang (ECTS): 7 Punkte
Einordnung (Semester): 2. Semester
Inhaltliche Voraussetzungen: Kenntnisse aus den Vorlesungen Grundlagen der Informatik 1, Digitaltechnik.
Voraussetzungen nach SPO: Nach SPO sind keine formellen Voraussetzungen erforderlich.
Kompetenzen: Die Studierenden kennen den Aufbau und die Leistungsmerkmale von Mikrocontroller-Architekturen und -systemen und können deren Eignung zur Lösung gegebener Problemstellungen im Bereich der Elektro- und Informationstechnik beurteilen. Im Modul Mikrocontroller-Systeme wird die Realisierung eingebetteter Systeme in der Kombination aus Hardware und hardwarenahe programmierter Software erlernt.
Prüfungsleistungen: Die theoretischen Kenntnisse der Studierenden für die Vorlesung Mikrocontroller-Systeme werden anhand einer schriftlichen Klausur (Dauer 120 min) bewertet. Die praktischen Fähigkeiten im Umgang mit dem Entwicklungssystem und die Ergebnisse der Laborversuche werden durch Kolloquien zu jedem Laborversuch bewertet.
Verwendbarkeit: Das Modul behandelt die speziellen Anforderungen der Programmierung von Mikrocontrollern in eingebetteten Systemen. Anhand der Anforderungen (Spezifikation) kann die Eignung eines Mikrocontroller-Systems für die jeweilige Problemstellung beurteilt werden. Zusammenhänge bestehen zur Vorlesung Informatik 1, in der Grundkenntnisse der Programmiersprache C vermittelt werden. Grundkenntnisse der Digitaltechnik werden bereits durch das Modul Digitaltechnik abgedeckt und bei entsprechenden Themenblöcken (Computerarithmetik, Peripherieschnittstellen) weiterreichend vermittelt.
Lehrveranstaltung: Mikrocontroller-Systeme
EDV-Bezeichnung: EITB251
Dozierende(r): Prof. Dr. Daniel Braun, Prof. Dr. Christian Langen
Umfang (SWS): 4
Turnus: Wintersemester und Sommersemester
Art, Modus: Vorlesung, Pflichtfach

Lehrsprache: Wintersemester Deutsch/Sommersemester Englisch
<p>Inhalte:</p> <ul style="list-style-type: none"> • Einführung – Embedded Systems und Mikrocontroller • Prozessor-Architekturen und -Befehlssätze • Programmierung in Assembler • ARM-Architektur • ARM-Organisation und -Implementierung • ARM-Befehlssatz • Schnittstellen, PWM und Analog-Digitalwandler • Architekturelle Unterstützung für Hochsprachen • Speicherhierarchie, Cache-Architekturen • Architekturelle Unterstützung für Betriebssysteme • Eigenschaften von Echtzeit-Betriebssystemen
<p>Empfohlene Literatur:</p> <ul style="list-style-type: none"> • Brinkschulte, Uwe; Ungerer, Theo: Mikrocontroller und Mikroprozessoren. Springer-Verlag, 5. Auflage 2010 • Furber, Steve: ARM System-on-Chip Architecture. Addison-Wesley, 2000 • Beierlein, Thomas; Hagenbruck, Olaf (Hrsg.): Taschenbuch Mikroprozessortechnik, Fachbuchverlag Leipzig, 4. Auflage 2011. • Märtin, Christian: Rechnerarchitekturen. CPUs, Systeme, Software-Schnittstellen. Fachbuchverlag Leipzig, 2001 • Wüst, Klaus: Mikroprozessortechnik. Grundlagen, Architekturen, Schaltungstechnik und Betrieb von Mikroprozessoren und Mikrocontrollern. Vieweg, 4. Auflage 2011 • Bauer, Sebastian: Eclipse für C/C++-Programmierer. Handbuch zu den Eclipse C/C++ Development Tools (CDT). dpunkt.verlag 2009. • Cockerell, Peter: ARM Assembly Language Programming. M.T.C. 1987 • Gibson, J. R.: ARM Assembly Language – an Introduction (Second Edition). J.R. Gibson 2011 • Smith, Warwick A.: C Programming for Embedded Microcontrollers, Elektor 2008 • Walter, Klaus-Dieter: Messen, Steuern und Regeln mit ARM-Mikrocontrollern. MSRAufgaben mit Hilfe von ARM-Mikrocontrollern lösen. Franzis, 2004 • Labrosse, Jean J.: MicroC/OS-II, CMP Books, 2. Auflage 2002

Lehrveranstaltung: Labor Mikrocontroller-Systeme
EDV-Bezeichnung: EITB252
Dozierende(r): Prof. Dr. Daniel Braun, Prof. Dr. Christian Langen
Umfang (SWS): 2
Turnus: Wintersemester und Sommersemester
Art, Modus: Labor, Pflichtfach
Lehrsprache: Wintersemester Deutsch/Sommersemester Englisch
<p>Inhalte:</p> <p>Versuche zu</p> <ul style="list-style-type: none"> • Computerarithmetik

- Modulare Programmierung, Verwendung des Stapelspeichers, Tastensteuerung
- Parallele Ein-/Ausgabe über Peripherieschnittstellen
- Serielle Datenübertragung (RS232)
- Hardwarenahe C-Programmierung, Interrupts, Zeitgeber
- Konfiguration Analog/Digital-Wandler und Auswertung
- Anwendung eines Echtzeit-Betriebssystems

Empfohlene Literatur:

- Brinkschulte, Uwe; Ungerer, Theo: Mikrocontroller und Mikroprozessoren. Springer-Verlag, 5. Auflage 2010
- Furber, Steve: ARM System-on-Chip Architecture. Addison-Wesley, 2000
- Beierlein, Thomas; Hagenbrück, Olaf (Hrsg.): Taschenbuch Mikroprozessortechnik, Fachbuchverlag Leipzig, 4. Auflage 2011.
- Märtin, Christian: Rechnerarchitekturen. CPUs, Systeme, Software-Schnittstellen. Fachbuchverlag Leipzig, 2001
- Wüst, Klaus: Mikroprozessortechnik. Grundlagen, Architekturen, Schaltungstechnik und Betrieb von Mikroprozessoren und Mikrocontrollern. Vieweg, 4. Auflage 2011
- Bauer, Sebastian: Eclipse für C/C++-Programmierer. Handbuch zu den Eclipse C/C++ Development Tools (CDT). dpunkt.verlag 2009.
- Cockerell, Peter: ARM Assembly Language Programming. M.T.C. 1987
- Gibson, J. R.: ARM Assembly Language – an Introduction (Second Edition). J.R. Gibson 2011
- Smith, Warwick A.: C Programming for Embedded Microcontrollers, Elektor 2008
- Walter, Klaus-Dieter: Messen, Steuern und Regeln mit ARM-Mikrocontrollern. MSRAufgaben mit Hilfe von ARM-Mikrocontrollern lösen. Franzis, 2004
- Labrosse, Jean J.: MicroC/OS-II, CMP Books, 2. Auflage 2002