

4.2.4 Mikrocontroller

Mikrocontroller
Modulübersicht
EDV-Bezeichnung: QUCB240
Modulverantwortliche(r): Prof. Dr. Christian Langen
Modulumfang (ECTS): 7 CP
Einordnung (Semester): 2. Semester
Inhaltliche Voraussetzungen: Kenntnisse aus den Vorlesungen Informatik Grundlagen, Digitaltechnik.
Voraussetzungen nach SPO: Nach SPO sind keine formellen Voraussetzungen erforderlich.
Lernergebnisse und Kompetenzen: Die Studierenden können - befähigt durch die Kenntnis des Aufbaus und der Wirkungsweise von Computern (Rechnerarchitekturen) - deren Leistungsfähigkeit und Grenzen bewerten. Dies befähigt sie, für gegebene Aufgabenstellungen gezielt dafür geeignete Plattformen auszuwählen und Lösungen für diese Aufgabenstellungen zu implementieren. Fachliche 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. Methodische Kompetenzen: Im Modul Mikrocontroller wird die Realisierung eingebetteter Systeme in der Kombination aus Hardware und hardwarenahe programmierter Software erlernt. Sozial- und Selbstkompetenzen: Teambildung zur Lösung der Laboraufgaben, gleichzeitig Bildung von Selbstkompetenz (Zeit- und Arbeitsmanagement).
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 (Studienleistung).
Lehrveranstaltung: Mikrocontroller
EDV-Bezeichnung: QUCB241
Dozierende(r): Prof. Dr. Bernard Schmidt, Prof. Dr. Christian Langen
Umfang (SWS): 4

Turnus: Sommersemester
Art, Modus: Vorlesung, Pflichtfach
Lehrsprache: Deutsch
<p>Studieninhalte:</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 • Architekturelle Unterstützung für Hochsprachen • Speicherhierarchie, Cache-Architekturen • Architekturelle Unterstützung für Betriebssysteme • Eigenschaften von Echtzeit-Betriebssystemen • General-Purpose-Ein- und -Ausgänge • Interrupts • Zeitgeber, Echtzeit-Uhren und Watchdog-Zeitgeber (Timer) • Pulsweitenmodulation (PWM) • Asynchrone und synchrone serielle Schnittstellen • Analog-Digital-Wandler • Controller Area Network (CAN)
<p>Empfohlene Literatur:</p> <ul style="list-style-type: none"> • Beierlein, Thomas; Hagenbruck, Olaf (Hrsg.): Taschenbuch Mikroprozessortechnik, Fachbuchverlag Leipzig, 4. Auflage 2011 • Brinkschulte, Uwe; Ungerer, Theo: Mikrocontroller und Mikroprozessoren. Springer-Verlag, 3. Auflage 2010 • Clements, Alan: Computer Organization and Architecture. Themes and Variations. CENGAGE Learning 2014 • Cockerell, Peter: ARM Assembly Language Programming. M.T.C. 1987 • Furber, Steve: ARM System-on-Chip Architecture. Addison-Wesley, 2000 • Furber, Steve: ARM-Rechnerarchitekturen für System-on-Chip- Design. Mitp, 2002 • Gibson, J. R.: ARM Assembly Language – an Introduction (Second Edition). J.R. Gibson 2011 • Hohl, William; Hinds, Christopher: ARM Assembly Language. Fundamentals and Techniques. CRC Press Second Edition 2014 • Labrosse, Jean J.: MicroC/OS-II, CMP Books, 2. Auflage 2002 • Markstedter, Maria Azeria: ARM Assembly. Internals & Reverse Engineering. Blue Fox Edition, Wiley 2023 • Nenni, Daniel; Dingee, Don: Mobile Unleashed. The Origin and Evolution of ARM Processors In Our Devices. SemiWiki LLC 2015 • Patterson, David A.; Hennesey, John L.: Computer Organization and Design. The Hardware/Software Interface. ARM Edition. Elsevier 2017
Lehrveranstaltung: Labor Mikrocontroller
EDV-Bezeichnung: QUCB242
Dozierende(r): Prof. Dr. Bernard Schmidt, Prof. Dr. Christian Langen

Umfang (SWS): 2
Turnus: Sommersemester
Art, Modus: Labor, Pflichtfach
Lehrsprache: Deutsch
<p>Studieninhalte: Versuche zu</p> <ul style="list-style-type: none"> • Computerarithmetik, Addition, Multiplikation, BCD-Addition • Modulare Programmierung, Verwendung von Unterprogrammen (Subroutinen), Verwendung des Stapelspeichers (Stack) • Tastensteuerung und Parallele Ein-/Ausgabe über Peripherieschnittstellen • Serielle Datenübertragung (RS232) • Hardwarenahe C-Programmierung, • Interrupts, Zeitgeber (Timer) • Konfiguration Analog/Digital-Wandler und Auswertung • Nebenläufigkeit zur Vorbereitung auf Multitasking • Anwendung eines Echtzeit-Betriebssystems
<p>Empfohlene Literatur:</p> <ul style="list-style-type: none"> • Beierlein, Thomas; Hagenbruck, Olaf (Hrsg.): Taschenbuch Mikroprozessortechnik, Fachbuchverlag Leipzig, 4. Auflage 2011 • Brinkschulte, Uwe; Ungerer, Theo: Mikrocontroller und Mikroprozessoren. Springer-Verlag, 3. Auflage 2010 • Clements, Alan: Computer Organization and Architecture. Themes and Variations. CENGAGE Learning 2014 • Cockerell, Peter: ARM Assembly Language Programming. M.T.C. 1987 • Furber, Steve: ARM System-on-Chip Architecture. Addison-Wesley, 2000 • Furber, Steve: ARM-Rechnerarchitekturen für System-on-Chip- Design. Mitp, 2002 • Gibson, J. R.: ARM Assembly Language – an Introduction (Second Edition). J.R. Gibson 2011 • Hohl, William; Hinds, Christopher: ARM Assembly Language. Fundamentals and Techniques. CRC Press Second Edition 2014 • Labrosse, Jean J.: MicroC/OS-II, CMP Books, 2. Auflage 2002 • Markstedter, Maria Azeria: ARM Assembly. Internals & Reverse Engineering. Blue Fox Edition, Wiley 2023 • Nenni, Daniel; Dingee, Don: Mobile Unleashed. The Origin and Evolution of ARM Processors In Our Devices. SemiWiki LLC 2015 <p>Patterson, David A.; Hennesey, John L.: Computer Organization and Design. The Hardware/Software Interface. ARM Edition. Elsevier 2017</p>