Friedrich-Alexander-Universität
Erlangen-Nürnberg

Modulhandbuch

für den Studiengang

Master of Science Information and Communication Technology (Prüfungsordnungsversion: 20222)

Inhaltsverzeichnis

Pflichtmodule.....
Hardware-Software-Co-Design.....8
Wahlpflichtmodul aus EEI im Schwerpunkt Embedded Systems.....
Analoge elektronische Systeme.....11
Architekturen der digitalen Signalverarbeitung.....13
Digitale elektronische Systeme.....15
Entwurf Integrierter Schaltungen II/Technologie integrierter Schaltungen.....17
Entwurf von Mixed-Signal-Schaltungen.....19
Hardware-Beschreibungssprache VHDL.....21
Information Theory and Coding / Informationstheorie und Codierung.....23
Integrierte Navigationssysteme.....28
Machine Learning in Signal Processing.....30
Medizinelektronik.....32
Satellitenkommunikation.....34
Speech and Audio Signal Processing.....39
Transceiver-Systementwurf.....42
Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems.....
Betriebssysteme (Vorlesung mit erweiterten Übungen).....45
Betriebssysteme (Vorlesung mit Übungen).....49
CPU Entwurf mit VHDL (CPU).....53
CPU Entwurf mit VHDL (Schwerpunkt Prozessorentwurf).....55
CPU Entwurf mit VHDL (Schwerpunkt VHDL).....56
Echtzeitsysteme 2 - Verlässliche Echtzeitsysteme.....58
Echtzeitsysteme mit erweiterten Übungen.....61
Echtzeitsysteme (Vorlesung mit Übungen).....66
Einführung in die moderne Kryptographie.....71
Grundlagen des Übersetzerbaus.....73
Kommunikationssysteme.....82
Konzeptionelle Modellierung.....84
Middleware - Cloud Computing (Vorlesung mit Übungen).....86
Parallele Systeme (Vorlesung mit erweiterten Übungen).....90
Programmierung und Architekturen von Cluster-Rechnern.....94
Rechnerarchitektur.....95
Rechnerarchitektur (Vorlesung mit Übung und Rechnerübung).....98
Reconfigurable Computing.....101
Reconfigurable Computing (Lecture with Extended Exercises).....104
Simulation und Modellierung I.....107
Swarm Intelligence.....110
Verteilte Systeme (Vorlesung mit erweiterten Übungen).....112
Verteilte Systeme (Vorlesung mit Übungen).....115
Pflichtmodule.....
Information Theory and Coding / Informationstheorie und Codierung.....119
Mobile Communications.....124
Quality of Service of Communication Systems.....126
Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication.....
Advanced Communication Networks.....129
Ausgewählte Kapitel der Audiodatenreduktion.....133
Bild-, Video- und mehrdimensionale Signalverarbeitung.....135
Entzerrung und adaptive Systeme in der digitalen Übertragung.....139
Globale Navigationssatellitensysteme.....142

HF- und Digital-Architektur von Funksystemen....144
Integrierte Schaltungen für Funkanwendungen....148
Kanalcodierung....150
Kommunikationselektronik....154
Kommunikationsstrukturen....158
Machine Learning in Communications....160
Machine Learning in Signal Processing....161
Medizinelektronik....163
MIMO Communication Systems....165
Optische Kommunikationsnetze....167
Optische Übertragungstechnik....170
Satellitenkommunikation....172
Schaltungen und Systeme der Übertragungstechnik....177
Statistical Signal Processing....179
Transmission and Detection for Advanced Mobile Communications....182
Wahlpflichtmodul aus INF im Schwerpunkt Networks and Digital Communication....
Advanced Networking LEx....186
AI-enabled Wireless Networks....189
Approximate Computing....192
Deep Learning....194
Einführung in die moderne Kryptographie....196
Pattern Recognition....198
Self-Organized Networks....201
Simulation und Modellierung I....203
Vernetzte Mobilität und autonomes Fahren....206
Pflichtmodule....
Pattern Recognition....209
Statistical Signal Processing....212
Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems....
Advanced Communication Networks....216
Auditory Models....220
Ausgewählte Kapitel der Audiodatenreduktion....221
Bild-, Video- und mehrdimensionale Signalverarbeitung....223
Digitale elektronische Systeme....227
Information Theory and Coding / Informationstheorie und Codierung....229
Integrierte Navigationssysteme....234
Kanalcodierung....236
Machine Learning in Communications....240
Machine Learning in Signal Processing....241
Mobile Communications....243
Musiksignalverarbeitung - Analyse....245
Musiksignalverarbeitung - Synthese....249
Signalanalyse....251
Speech and Audio Signal Processing....253
Speech Enhancement....256
Speech Enhancement....258
Transformationen in der Signalverarbeitung....260
Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems....
Computer Graphics....263
Diagnostic Medical Image Processing....266
Dienstgüte von Kommunikationssystemen....268
eBusiness Technologies und Evolutionäre Informationssysteme....270
Echtzeitsysteme 2 - Verlässliche Echtzeitsysteme....274

Echtzeitsysteme mit erweiterten Übungen....277
Echtzeitsysteme (Vorlesung mit Übungen)....282
Hardware-Software-Co-Design....287
Hardware-Software-Co-Design (Vorlesung mit erweiterter Übung)....290
Implementierung von Datenbanksystemen....294
Interaktive Computergraphik....297
Interventional Medical Image Processing....299
Konzeptionelle Modellierung....302
Multimedia Security....304
Pattern Analysis....306
Scientific Visualization....309
Wahlmodule aus dem Angebot von EEI und Informatik....
Advanced Communication Networks....312
Advanced Networking LEx....316
AI-enabled Wireless Networks....319
Analoge elektronische Systeme....322
Approximate Computing....324
Architekturen der digitalen Signalverarbeitung....326
Auditory Models....328
Ausgewählte Kapitel der Audiodatenreduktion....329
Betriebssysteme (Vorlesung mit erweiterten Übungen)....331
Betriebssysteme (Vorlesung mit Übungen)....335
Bild-, Video- und mehrdimensionale Signalverarbeitung....339
Computer Graphics....343
CPU Entwurf mit VHDL (Schwerpunkt Prozessorentwurf)....346
CPU Entwurf mit VHDL (Schwerpunkt VHDL)....347
Cyber-Physical Systems....349
Deep Learning....351
Diagnostic Medical Image Processing....353
Dienstgüte von Kommunikationssystemen....355
Digital Communications....357
Digitale elektronische Systeme....359
eBusiness Technologies und Evolutionäre Informationssysteme....361
Echtzeitsysteme 2 - Verlässliche Echtzeitsysteme....365
Echtzeitsysteme mit erweiterten Übungen....368
Echtzeitsysteme (Vorlesung mit Übungen)....373
Einführung in die moderne Kryptographie....378
Entwurf Integrierter Schaltungen II/Technologie integrierter Schaltungen....380
Entwurf von Mixed-Signal-Schaltungen....382
Entzerrung und adaptive Systeme in der digitalen Übertragung....384
Fortgeschrittene forensische Informatik....387
Globale Navigationssatellitensysteme....388
Grundlagen des Übersetzerbaus....390
Hardware-Beschreibungssprache VHDL....399
Hardware-Software-Co-Design....401
Hardware-Software-Co-Design (Vorlesung mit erweiterter Übung)....404
HF- und Digital-Architektur von Funksystemen....408
Implementierung von Datenbanksystemen....412
Information Theory and Coding / Informationstheorie und Codierung....415
Integrierte Navigationssysteme....420
Integrierte Schaltungen für Funkanwendungen....422
Interaktive Computergraphik....424
Interventional Medical Image Processing....426

Kanalcodierung.....429
Kommunikationselektronik.....433
Kommunikationsstrukturen.....437
Kommunikationssysteme.....439
Konzeptionelle Modellierung.....441
Künstliche Intelligenz I.....443
Machine Learning in Communications.....446
Machine Learning in Signal Processing.....447
Medizinelektronik.....449
Middleware - Cloud Computing (Vorlesung mit erweiterten Übungen).....451
Middleware - Cloud Computing (Vorlesung mit Übungen).....455
MIMO Communication Systems.....459
Mobile Communications.....461
Multimedia Security.....463
Musiksignalverarbeitung - Analyse.....465
Musiksignalverarbeitung - Synthese.....469
Optische Kommunikationsnetze.....471
Optische Übertragungstechnik.....474
Parallele Systeme (Vorlesung mit erweiterten Übungen).....476
Pattern Analysis.....480
Pattern Recognition.....483
Programmierung und Architekturen von Cluster-Rechnern.....486
Radar, RFID and Wireless Sensor Systems (RWS).....487
Rechnerarchitektur.....489
Rechnerarchitektur (Vorlesung mit Übung und Rechnerübung).....492
Reconfigurable Computing.....495
Reconfigurable Computing (Lecture with Extended Exercises).....498
Regelungstechnik A (Grundlagen).....501
Satellitenkommunikation.....503
Schaltungen und Systeme der Übertragungstechnik.....508
Scientific Visualization.....510
Self-Organized Networks.....512
Signalanalyse.....514
Simulation und Modellierung I.....516
Speech and Audio Signal Processing.....519
Speech Enhancement.....522
Statistical Signal Processing.....524
Swarm Intelligence.....527
Transceiver-Systementwurf.....529
Transformationen in der Signalverarbeitung.....531
Transmission and Detection for Advanced Mobile Communications.....533
Verifikation digitaler Systeme.....536
Vernetzte Mobilität und autonomes Fahren.....538
Verteilte Systeme (Vorlesung mit erweiterten Übungen).....540
Virtuelle Maschinen (Vorlesung mit Übungen).....543
Virtuelle Maschinen (Vorlesung mit Übung und Laborübung).....545
Wahlmodule aus dem Angebot der Technischen Fakultät oder der Naturwissenschaftlichen Fakultät.....
Bild-, Video- und mehrdimensionale Signalverarbeitung.....548
Experimentalphysik I.....552
Experimentalphysik II.....554
Human Computer Interaction.....556
Radar, RFID and Wireless Sensor Systems (RWS).....559

Scientific Visualization..... 561
Self-Organized Networks.....563
Praktikum oder Projektarbeit.....
Audio Processing Laboratory..... 566
Laborpraktikum Digitaler ASIC-Entwurf.....568
Laborpraktikum Eingebettete Mikrocontroller-Systeme (PEMSY).....571
Laborpraktikum Mobilkommunikation..... 574
Laborpraktikum Systematischer Entwurf programmierbarer Logikbausteine..... 577
Praktikum Architekturen der digitalen Signalverarbeitung.....578
Praktikum Bild- und Videosignalverarbeitung auf eingebetteten Plattformen.....579
Praktikum Communications Systems Design.....582
Praktikum Digitale Übertragung..... 585
Praktikum Hochfrequenztechnik/Mikrowellentechnik 2..... 590
Praktikum Machine Learning in der Signalverarbeitung.....592
Praktikum Mikroelektronik.....594
Praktikum Photonik/Lasertechnik 1..... 596
Praktikum Photonik/Lasertechnik 2..... 598
Praktikum Test..... 600
Praktikum zu High-Performance Analog- und Umsetzer-Design..... 602
Software Engineering in der Praxis.....603
Seminar.....
Advanced Seminar on Medical Electronics and Systems for Ambient Assisted Living AAL..... 605
Audio Processing Seminar..... 607
Hauptseminar Ausgewählte Kapitel der Informationstechnik (Kommunikationselektronik).....609
Hauptseminar Ausgewählte Kapitel der Navigation und Identifikation..... 611
Hauptseminar Cognitive Science in Engineering..... 613
Hauptseminar Software Engineering.....614
Seminar Advanced Algorithms in Medical Image Processing.....615
Seminar Ausgewählte Kapitel der Multimediakommunikation und Signalverarbeitung..... 616
Seminar Ausgewählte Kapitel der Nachrichtentechnik..... 618
Seminar Deep Learning..... 621
Seminar Energieinformatik..... 622
Seminar Kommunikationssysteme (B.Sc.)..... 624
Seminar Kommunikationssysteme (M.Sc.).....626
Seminar Machine Learning and Data Analytics for Industry 4.0..... 628
Seminar Meta Learning..... 630
Seminar Sprachtechnologie für Sprachpathologien..... 631
Seminar Visual Computing..... 632
Systems- and Networks-on-a-Chip..... 634

Pflichtmodule

1	**Modulbezeichnung** 43490	**Hardware-Software-Co-Design** (Hardware-software-co-design)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übungen zu Hardware-Software-Co-Design (2 SWS) Vorlesung: Hardware-Software-Co-Design (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Muhammad Sabih Tobias Hahn Dr.-Ing. Stefan Wildermann Prof. Dr.-Ing. Jürgen Teich	

4	**Modulverantwortliche/r**	Joachim Falk Prof. Dr.-Ing. Jürgen Teich
5	**Inhalt**	Zahlreiche Realisierungen eingebetteter Systeme (z.B. Mobiltelephone, Faxgeräte, Industriesteuerungen) zeichnen sich durch kooperierende Hardware- und Softwarekomponenten aus. Die Popularität solcher Realisierungsformen lässt sich begründen durch 1) die steigende Vielfalt und Komplexität heterogener Systeme, 2) die Notwendigkeit, Entwurfs- und Testkosten zu senken und 3) Fortschritte in Schlüsseltechnologien (Mikroelektronik, formale Entwurfsmethoden). Zum Beispiel bieten Halbleiterhersteller kostengünstige ASICs an, die einen Mikrocontroller und benutzerspezifische Peripherie und Datenpfade auf einem Chip integrieren. Die Synthese solcher Systeme wirft jedoch eine Reihe neuartiger Entwurfsprobleme auf, insbesondere 1) die Frage der Auswahl von Hardware- und Softwarekomponenten, 2) die Partitionierung einer Spezifikation in Hard- und Software, 3) die automatische Synthese von Interface- und Kommunikationsstrukturen und 4) die Verifikation und Cosimulation. 1) Überblick und Vergleich von Architekturen und Komponenten in Hardware/Software-Systemen. 2) Aufbau eines Compilers und Codeoptimierungsverfahren für Hardware und Software 3) Hardware/Software-Partitionierung (Partitionierung komplexer Systeme, Schätzungsverfahren, Performanzanalyse, Codegenerierung) 4) Interfacesynthese (Kommunikationsarten, Synchronisation, Synthese) 5) Verifikation und Cosimulation 6) Tafelübungen
6	**Lernziele und Kompetenzen**	Fachkompetenz - Wissen • Die Studierenden erhalten Einblick in ein aktuelles Forschungsgebiet. Fachkompetenz - Verstehen • Die Studierenden verstehen Grundlagen des modernen Systementwurfs. • Die Studierenden erklären Implementierungsalternativen für digitale Hardware/Software-Systeme.

		Fachkompetenz - Anwenden • Die Studierenden wenden grundlegende Algorithmen an, zur Analyse und Optimierung von Hardware/Software-Systemen.
7	**Voraussetzungen für die Teilnahme**	Die Auswahl dieses Moduls schließt die Auswahl des Moduls „Hardware-Software-Co-Design (Vorlesung mit erweiterter Übung)“ aus.
8	**Einpassung in Studienverlaufsplan**	Semester: 2
9	**Verwendbarkeit des Moduls**	Pflichtmodule Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Empfohlene Bücher zur Begleitung und Vertiefung: • Teich J., Haubelt C.: "Digitale Hardware/Software-Systeme: Synthese und Optimierung", Springer-Verlag, 2007, ISBN: 978-3-540-46822-6 • Gajski, D. et al.: "Specification and Design of Embedded Systems", Prentice Hall, Englewood Cliffs, NJ, 1994, ISBN: 978-0131507319 Weitere Informationen: https://www.cs12.tf.fau.de/lehre/lehrveranstaltungen/vorlesungen/hardware-software-co-design

Wahlpflichtmodul aus EEI im Schwerpunkt Embedded Systems

1	**Modulbezeichnung** 96500	**Analoge elektronische Systeme** (Analogue electronic systems)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Analoge elektronische Systeme (3 SWS) Übung: Übungen zu Analoge elektronische Systeme (1 SWS)	5 ECTS -
3	Lehrende	Torsten Reißland Prof. Dr.-Ing. Robert Weigel Christof Pfannenmüller	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Robert Weigel
5	**Inhalt**	• Feldeffekttransistor • Verstärker, Leistungsverstärker • Nichtlinearität und Verzerrung • Filtertheorie • Realisierung von Filtern • Intrinsisches Rauschen (Konzepte) • Physikalische Rauschursachen • Rauschparameter • Mischer • Oszillatoren • Phasenregelschleifen (PLLs)
6	**Lernziele und Kompetenzen**	• Die Studierenden erlangen Kenntnisse um Rauscheffekte und Nichtlinearitäten in Analogschaltungen zu erklären • Die Studierenden verstehen die Ursachen verschiedener physikalischer Rauschprozesse und können diese klassifizieren • Die Studierenden erwerben die Fähigkeit zur Planung und Implementierung frequenzumsetzender Systeme mittels zugehöriger Frequenz- und Pegelpläne • Die Studierenden bewerten Hochfrequenzoszillatoren und stabilisierende PLL-Schaltungen • Die Studierenden untersuchen Messaufbauten zur Charakterisierung von Rauschen und Nichtlinearitäten • Die Studierenden analysieren den inneren Aufbau von Leistungsverstärkern auf Basis von Transistorschaltungen • Die Studierenden sind in der Lage komplexe Analogschaltungen simulativ und analytisch zu untersuchen und deren Verhalten im Groß- und Kleinsignalbereich zu charakterisieren • Die Studierenden führen Filterentwürfe durch und bestimmen deren Amplituden- und Phasengang • Die Studierenden können bei auftretenden Problemen selbstständig mit Hilfe weitergehender Literatur oder durch Diskussion in der Gruppe Lösungen erarbeiten
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191

		Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 96010	**Architekturen der digitalen Signalverarbeitung** (Architectures for digital signal processing)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übungen zu Architekturen der Digitalen Signalverarbeitung (2 SWS) Vorlesung: Architekturen der Digitalen Signalverarbeitung (2 SWS)	- 5 ECTS
3	Lehrende	Torsten Reißland Prof. Dr.-Ing. Georg Fischer	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Georg Fischer
5	**Inhalt**	Inhalt: • Basis-Algorithmen der Signalverarbeitung (FFT, Fensterung, Digitale FIR- und IIR-Filter) • Nichtideale Effekte bei Digitalfiltern (Quantisierung der Filterkoeffizienten, Quantisierte Arithmetik) • CORDIC-Architekturen • Architekturen für Multiratensysteme (Abtastratenumsetzer) • Architekturen digitaler Signalgeneratoren • Maßnahmen zur Leistungssteigerung (Pipelining) • Architekturen digitaler Signalprozessoren • Anwendungen Content: • Basic algorithms of signal processing (FFT, windowing, digital FIR and IIR-filters) • Non-idealities of digital filters (quantization of filter coefficients, fixed-point arthmetic) • CORDIC-architectures • Architectures of systems with multiple sampling rates (conversion between different sampling rates) • Digital signal generation • Measures of performance improvement (pipelining) • Architecture of digital signal processors • Applications
6	**Lernziele und Kompetenzen**	Die Studierenden erlangen Grundlagenkenntnisse der Signaltheorie und können zeit- und wertkontinuierliche sowie zeit- und wertdiskrete Signale im Zeit- und Frequenzbereich definieren und erklären Die Studierenden sind in der Lage, ein klassisches Echtzeitsystem zur digitalen Signalverarbeitung konzeptionieren und die Einzelkomponenten nach den Anforderungen zu dimensionieren Die Studierenden erlangen einen Überblick über Vor- und Nachteile analoger sowie digitaler Signalverarbeitung Die Studierenden verstehen die Theorie der Fourier-Transformation und sind in der Lage, die Vorteile der Fast-Fourier-Transformation in der digitalen Signalverarbeitung zu verstehen und anzuwenden

		Die Studierenden können digitale Filter dimensionieren und beurteilen ===Englisch=== Students • can obtain fundamentals of signal theory and can define as well time-comtinous and value-continous as time-discrete and value-discrete signals in time and frequency domain • can construct a realtime digital signal processing system and dimension its components according requirements • can review pros and cons of analogue versus digitzal signal processing • can apply fourier transformation and illustrate the advantages of fast fourier transformation in the context of digital signal processing • can dimension digital filters and evaluate their performance
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 96090	**Digitale elektronische Systeme** (Digitale elektronische Systeme)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Digitale elektronische Systeme (3 SWS) Übung: Übungen zu Digitale elektronische Systeme (1 SWS)	5 ECTS -
3	Lehrende	Prof. Dr.-Ing. Robert Weigel Albert-Marcel Schrotz	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Robert Weigel
5	**Inhalt**	• Analog-Digital-Umsetzer: Qualitätsmerkmale, Messtechnik, Hardwarearchitekturen • Digital-Analog-Umsetzer: Qualitätsmerkmale, Messtechnik, Hardwarearchitekturen • Programmierbare Logikschaltungen (PLD, FPGA): Grundlegende Konzepte, Kategorien, Hardwarearchitekturen • Digitale-Filter: Theorie, Eigenschaften, Entwicklung und Implementierung und IIR und FIR Filtern
6	**Lernziele und Kompetenzen**	• Die Studierenden verstehen die Hardwarearchitekturen und Funktionsweisen von Komponenten digitaler Elektronischer Systeme wie Digital-Analog-Umsetzer, Analog-Digital Umsetzer, PLDs und FPGAs und können diese erläutern • Die Studierenden Verstehen die Qualitätsmerkmale von Digitalen Elektronischen Komponenten, können diese auf konkrete Komponenten anwenden und somit die Qualität von digitalen Elektronischen Komponenten anhand der in Datenblättern typischer weise gegebenen Qualitätsmerkmale evaluieren • Die Studierenden können die Einflüsse von nichtidealen Bauelementen auf digitale elektronische Systeme analysieren • Die Studierenden verstehen die Funktion, die Eigenschaften, die Entwicklungsmethodik sowie die Implementierung von digitalen Filtern und könne diese erläutern
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	Unregelmäßig
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester

15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 96190	**Entwurf Integrierter Schaltungen II/Technologie integrierter Schaltungen** (Design of integrated circuits II/technology of integrated circuits)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übungen zu Entwurf Integrierter Schaltungen II (2 SWS) Vorlesung: Halbleitertechnologie I - Technologie integrierter Schaltungen (3 SWS) Übung: Übung zu Halbleitertechnologie I - Technologie integrierter Schaltungen (1 SWS)	- 5 ECTS -
3	Lehrende	Tobias Rumpel Florian Deeg Prof. Dr.-Ing. Jörg Schulze Jannik Schwarberg	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Jörg Schulze
5	**Inhalt**	In diesem Modul werden die wesentlichen Technologieschritte zur Herstellung elektronischer Halbleiterbauelemente und integrierter Schaltungen behandelt. Ausgehend von der Frage nach den relevanten Parametern chemischer und physikalischer Herstellungsprozesse werden zu Beginn die Verfahren und Methoden zur Herstellung von einkristallinen Siliziumkristallen besprochen. Anschließend werden die physikalischen und chemischen Grundlagen der Oxidation, der Dotierverfahren Diffusion und Ionenimplantation sowie der physikalischen und chemischen Gasphasenabscheidung von dünnen Schichten behandelt. Eine Einführung in die relevanten Lithographie- und Strukturierungsverfahren beendet den Kanon der wesentlichen Technologieschritte zur Herstellung elektronischer Halbleiterbauelemente. Ergänzend dazu werden Sequenzen von Prozessabläufen, wie sie heute bei der Herstellung von hochintegrierten Schaltungen wie Mikroprozessoren oder Speichern verwendet werden, besprochen.
6	**Lernziele und Kompetenzen**	Die Studierenden Fachkompetenz Anwenden • beschreiben die Technologieschritte und notwendigen Prozessgeräte • erklären die physikalischen und chemischen Vorgänge bei der Herstellung von Integrierten Schaltungen Evaluieren (Beurteilen)

		• ermitteln en Einfluss von Prozessparametern und können Vorhersagen für Einzelprozesse ableiten • sind in der Lage, verschiedene Herstellungsschritte hinsichtlich ihrer Vor- und Nachteile bzgl. der hergestellten Schichten, Strukturen oder Bauelemente zu beurteilen
7	**Voraussetzungen für die Teilnahme**	Kenntnisse aus dem Bereich Halbleiterbauelemente (Pflichtveranstaltung im Bachelorstudiengang EEI und Mechatronik)
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	
11	**Berechnung der Modulnote**	
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• S. M. Sze: VLSI - Technology, MacGraw-Hill, 1988 • C. Y. Chang, S. M. Sze: ULSI - Technology, MacGraw-Hill, 1996 • D. Widmann, H. Mader, H. Friedrich: Technology of Integrated Circuits, Springer Verlag, 2000 • Hong Xiao: Introduction to Semiconductor Manufacturing Technology, Prentice Hall, 2001

1	**Modulbezeichnung** 96200	**Entwurf von Mixed-Signal-Schaltungen** (Design of mixed-signal circuits)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übungen zu Entwurf von Mixed-Signal-Schaltungen (2 SWS) Vorlesung: Entwurf von Mixed-Signal-Schaltungen (2 SWS)	- 5 ECTS
3	Lehrende	Tobias Rumpel Florian Deeg Prof. Dr.-Ing. Sebastian Sattler	

4	**Modulverantwortliche/r**	Peter Meisel Prof. Dr.-Ing. Sebastian Sattler
5	**Inhalt**	Es werden Methoden zur Analyse und Synthese von Phänomenen behandelt, welche aus sogenannten Rückkopplungen in gemischt analog-digitalen Systemen entstehen. Es wird an Hand eines allgemeinen Transistormodells abstrahiert, und Beispiele aus der Integrierten Schaltungs- und Systemtechnik erarbeitet. • Modellierung aktiver Bauelemente • Grundschaltungen des allgemeinen Transistors • Abstraktion der Rückkopplung • Analyse der Stabilität im Frequenz- und Zeitbereich • Kompensationstechniken im Frequenzbereich • Grundschaltungen von Rückkopplungen • Harmonische Verzerrungen • Rauschen • Beispiele von Rückkopplungen
6	**Lernziele und Kompetenzen**	Analysieren • Die Studierenden können die verschiedenste Strukturen für analoge integrierte Schaltungen entwickeln, analysieren und bewerten. Erschaffen • Die Studierenden gewinnen einen Überblick über die wichtigsten Methoden und Verfahren für Analyse und Entwurf von analogen rückgekoppelten Schaltungen.
7	**Voraussetzungen für die Teilnahme**	Schaltungstechnik, Entwurf Integrierter Schaltungen I, o.ä.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)

12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	G. Palumbo, S. Pennisi, Feedback Amplifiers, Theory and Design, Springer 2009

1	**Modulbezeichnung** 96750	**Hardware-Beschreibungssprache VHDL** (VHDL Hardware description language)	**2,5 ECTS**
2	Lehrveranstaltungen	Vorlesung mit Übung: Hardware-Beschreibungssprache VHDL (2 SWS)	2,5 ECTS
3	Lehrende	Jürgen Frickel	

4	**Modulverantwortliche/r**	Jürgen Frickel
5	**Inhalt**	Vorlesung mit integrierter Rechnerübung zur Syntax und zur Anwendung der Hardware-Beschreibungssprache VHDL (Very High Speed Integrated Circuit Hardware Description Language) nach dem Sprachstandard IEEE 1076-1987 und 1076-1993, Anwendung von VHDL zum Entwurf von FPGAs in der Praxis. • Konzepte und Konstrukte der Sprache VHDL • Beschreibung auf Verhaltens- und Register-Transfer-Ebene • Simulation und Synthese auf der Gatterlogik-Ebene • Verwendung professioneller Software-Tools (Xilinx Vivado) • Vorlesung mit integrierten Rechner-Übungen (Labs) • Kursmaterial ist englisch-sprachig, die Vorlesungssprache deutsch Zielgruppe sind Hörer aller Fachrichtungen, die sich mit dem Entwurf, Simulation und Synthese digitaler Systeme und Schaltungen beschäftigen wollen.
6	**Lernziele und Kompetenzen**	**Fachkompetenz** • Wissen Die Studierenden können Begriffe und Definitionen einer Hardware-Beschreibungssprache (hier VHDL) darlegen. • Verstehen Die Studierenden verstehen den Zusammenhang bzw. die Transformation zwischen einer Hardware-Struktur und deren Abbildung in einer Hardware-Beschreibungssprache in beiden Richtungen. • Analysieren Die Studierenden klassifizieren ein gewünschtes Systemverhalten, strukturieren dieses in Teilmodule, und realisieren die Teilmodule bzw. das System in der Hardware-Beschreibungssprache. • Evaluieren (Beurteilen) Die Studierenden schätzen VHDL-Modelle bezüglich des quantitativen und qualitativen Hardware-Aufwandes ein, überprüfen diese gegen vorliegende Randbedingungen (constraints), und vergleichen sie mit alternativen Lösungen.

		Lern- bzw. Methodenkompetenz Die theoretischen Inhalte der Sprache können durch Einsatz eines Simulations- und Synthesewerkzeuges im praktischen Einsatz selbständig verifiziert und deren Verständnis vertieft werden. **Sozialkompetenz** Die Studierenden stärken ihre Fähigkeit, vorliegende Aufgabenstellungen in Gruppenarbeit gemeinsam zu lösen.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich (60 Minuten)
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 93601	**Information Theory and Coding /** **Informationstheorie und Codierung** (Information theory and coding)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Tutorial for Information Theory and Coding (1 SWS) Vorlesung: Information Theory and Coding (3 SWS)	- 5 ECTS
3	Lehrende	Dr.-Ing. Ali Bereyhi Prof. Dr.-Ing. Ralf Müller	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Ralf Müller
5	**Inhalt**	1. Introduction: binomial distribution, (7,4)-Hamming code, parity-check matrix, generator matrix 2. Probability, entropy, and inference: entropy, conditional probability, Bayes law, likelihood, Jensens inequality 3. Inference: inverse probability, statistical inference 4. The source coding theorem: information content, typical sequences, Chebychev inequality, law of large numbers 5. Symbol codes: unique decidability, expected codeword length, prefix-free codes, Kraft inequality, Huffman coding 6. Stream codes: arithmetic coding, Lempel-Ziv coding, Burrows-Wheeler transform 7. Dependent random variables: mutual information, data processing lemma 8. Communication over a noisy channel: discrete memory-less channel, channel coding theorem, channel capacity 9. The noisy-channel coding theorem: jointly-typical sequences, proof of the channel coding theorem, proof of converse, symmetric channels 10. Error-correcting codes and real channels: AWGN channel, multivariate Gaussian pdf, capacity of AWGN channel 11. Binary codes: minimum distance, perfect codes, why perfect codes are bad, why distance isnt everything 12. Message passing: distributed counting, path counting, low-cost path, min-sum (=Viterbi) algorithm 13. Exact marginalization in graphs: factor graphs, sum-product algorithm

14. Low-density parity-check codes: density evolution, check node degree, regular vs. irregular codes, girth

15. Lossy source coding: transform coding and JPEG compression

--

1. Einleitung: Binomialverteilung, (7,4)-Hamming-Code, Paritätsmatrix, Generatormatrix

2. Wahrscheinlichkeit, Entropie und Inferenz: Entropie, bedingte Wahrscheinlichkeit, Bayessches Gesetz, Likelihood, Jensensche Ungleichung

3. Inferenz: Inverse Wahrscheinlichkeit, statistische Inferenz

4. Das Quellencodierungstheorem: Informationsgehalt, typische Folgen, Tschebyschevsche Ungleichung, Gesetz der großen Zahlen

5. Symbolcodes: eindeutige Dekodierbarkeit, mittlere Codewortlänge, präfixfreie Codes, Kraftsche Ungleichung, Huffmancodierung

6. Stromcodes: arithmetische Codierung, Lempel-Ziv-Codierung, Burrows-Wheeler-Transformation

7. Abhängige Zufallsvariablen: Transinformation, Datenverarbeitungslemma

8. Kommunikation over gestörte Kanäle: diskreter gedächtnisloser Kanal, Kanalcodierungstheorem, Kanalkapazität

9. Das Kanalcodierungstheorem: verbundtypische Folgen, Beweis des Kanalcodierungstheorems, Beweis des Umkehrsatzes, symmetrische Kanaäle

10. Fehlerkorrigierende Codes und reale Kanäle: AWGN-Kanal, mehrdimensionale Gaußsche WDF, Kapazität des AWGN-Kanals

11. Binäre Codes: Minimaldistanz, perfekte Codes, Warum perfekte Codes schlecht sind, Warum Distanz nicht alles ist

12. Nachrichtenaustausch: verteiltes Zählen, Pfadzählen, günstigster Pfad, Minimumsummenalgorithmus

13. Exakte Marginalisierung in Graphen: Faktorgraph, Summenproduktalgorithmus

14. LDPC-Codes: Dichteevolution, Knotenordnung, reguläre und irreguläre Codes, Graphumfang

		15. Verlustbehaftete Quellencodierung: Transformationscodierung und JPEG-Kompression
6	**Lernziele und Kompetenzen**	The students apply Bayesian inference to problems in both communications and everyday's life. The students explain the concept of digital communications by means of source compression and forward-error correction coding. For the design of communication systems, they use the concepts of entropy and channel capacity. They calculate these quantities for memoryless sources and channels. The students proof both the source coding and the channel coding theorem. The students compare various methods of source coding with respect to compression rate and complexity. The students apply source compression methods to measure mutual information. The students factorize multivariate functions, represent them by graphs, and marginalize them with respect to various variables. The students explain the design of error-correcting codes and the role of minimum distance. They decode error-correcting codes by means of maximum-likelihood decoding and message passing. The students apply distributed algorithms to problems in both communications and everydays life. The students improve the properties of low-density parity-check codes by widening the girth and/or irregularity in the degree distribution. The students transform source images into the frequency domain to improve lossy compression. -- Die Studierenden wenden Bayessche Inferenz auf Probleme in der Nachrichtentechnik und im Alltagsleben an. Die Studierenden erklären die konzeptuelle Trennung von digitaler Übertragung in Quellen- und Kanalcodierung.

		Kommunikationssysteme entwerfen sie unter Betrachtung von Entropie und Kanalkapazität. Sie berechnen diese Größen für gedächtnislose Quellen und Kanäle. Die Studierenden beweisen sowohl das Quellen- als auch das Kanalcodierungstheorem. Die Studierenden vergleichen verschiedenartige Quellencodierungsverfahren hinsichtlich Komplexität und Kompressionsrate. Die Studierenden verwenden Quellencodierverfahren zur Messung von Transinformation. Die Studierenden faktorisieren Funktionen mehrerer Veränderlicher, stellen diese als Graph dar und marginalisieren sie bezüglich mehrerer Veränderlicher. Die Studierenden erklären den Entwurf von Kanalcodes und den Einfluss der Minimaldistanz. Sie decodieren Kanalcodes gemäß maximaler Likelihood und Nachrichtenaustausch. Die Studierenden wenden verteilte Algorithmen auf Probleme der Nachrichtentechnik und des Alltagslebens an. Die Studierenden verbessern die Eigenschaften von LDPC-Codes durch Erhöhung des Umfangs und/oder durch irreguäre Knotenordnungsverteilungen. Die Studierenden transformieren Bildquellen zur Verbesserung verlustbehafteter Kompression iin den Frequenzbereich.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 2
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h

14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	MacKay, D.: Information Theory, Inference, and Learning Algorithms, Cambridge University Press, Cambridge, 2003.

1	**Modulbezeichnung** 96101	**Integrierte Navigationssysteme** (keine englischsprachige Modulbezeichnung hinterlegt!)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übung Integrierte Navigationssysteme (1 SWS) Vorlesung: Integrierte Navigationssysteme (3 SWS)	- 5 ECTS
3	Lehrende	Prof. Dr. Jörn Thielecke	

4	**Modulverantwortliche/r**	Prof. Dr. Jörn Thielecke
5	**Inhalt**	**1. Überblick** • Von der Astronavigation zur Navigation mit Mikroelektronik • Messprinzipien & Positionsberechnung (Standlinien/-flächen) • Begriffsdefinitionen (s. US Federal Radionavigation Plan), Genauigkeit, Verfügbarkeit, Verlässlichkeit, Integrität, etc. • Systematische Strukturierung des Gebiets: siehe 2. bis 7. **2. Positions- und Lagebestimmung** • Funkausbreitung und Funkortung (Beispiel WLAN) • Fingerabdruckverfahren • Lokalisierung mit Markovketten **3. Koppelnavigation (Tracking) mittels Trägheitsnavigation** • Koordinatensysteme und ihre Einsatzgebiete • Mathematische Gundlagen, z.B. Quaternionen, Corioliseffekt • Strapdown Inertial Navigation Systems • Sensorprinzipien und Trägheitssensoren • Computergestützte Lösung der Navigationsgleichungen • System- und Fehlermodellierung im Zustandsraum • Das Kalmanfilter und Glättung mittels Retrodiktion **4. Seiteninformationen: Kinematik und Karten (kurze Übersicht)** **5. Landmarken als lokaler Ortsbezug** • Merkmalsbasierte Ortung z.B. mit Kamera oder UWB • Partikelfilter und Monte-Carlo-Integration **6. Integration von Navigationskomponenten: Sensordatenfusion** • Fusionsarchitekturen: Beispiel GPS & Trägheitsnavigation **7. Einbettung von Navigationssystemen** • Assisted GPS oder Location Based Service Anmerkung: Die Navigationsmethoden werden gleichermaßen anhand von Tafel- und Rechnerübungen (MATLAB) einstudiert
6	**Lernziele und Kompetenzen**	1. Die Studierenden werden in die Lage versetzt, typische Navigationsverfahren hinsichtlich ihrer Funktionsweise und Einsetzbarkeit zu analysieren, zu bewerten und weiterzuentwickeln.

		2. Die Studierenden lernen Navigationsgleichungen selbst aufzustellen, anzuwenden und mit unterschiedlichen Algorithmen auf dem Computer zu lösen. 3. Die Studierenden entwickeln ein Verständnis für die Herausforderungen bei der Integration unterschiedlicher Teilsysteme zu einem Navigationssystem und der Einbettung von Navigationssystemen in übergeordnete Systeme
7	**Voraussetzungen für die Teilnahme**	Keine formalen Voraussetzungen, geeignet für Masterstudium, grundlegende Kenntnisse erforderlich in: linearer Algebra, Physik, Signal- & System¬theorie, Wahrscheinlichkeitstheorie.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%) Bei bestandener Prüfung wird die Note um eine Teilnotenstufe (z.B. von 2,0 auf 1,7) verbessert, wenn Sie mindestens 75% der Hausaufgaben einschließlich der Rechnerübungen erfolgreich absolviert haben. Eine Note besser als 1,0 wird nicht vergeben.
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Skriptum zur Lehrveranstaltung.

1	**Modulbezeichnung** 48440	**Machine Learning in Signal Processing** (Machine learning in signal processing)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Supplements for Machine Learning in Signal Processing (1 SWS) Vorlesung: Machine Learning in Signal Processing (3 SWS)	- 5 ECTS
3	Lehrende	Kamal Gopikrishnan Nambiar Prof. Dr. Vasileios Belagiannis	

4	**Modulverantwortliche/r**	PD Dr.Ing. Jürgen Seiler
5	**Inhalt**	This course is an introduction into machine learning and artificial intelligence. The special emphasis is on applications to modern signal processing problems. The course is focused on design principles of machine learning algorithms. The lectures start with a short introduction, where the nomenclature is defined. After this, probabilistic graphical models are introduced and the use of latent variables is discussed, concluding with a discussion of hidden Markov models and Markov fields. The second part of the course is about deep learning and covers the use of deep neural networks for machine learning tasks. In the last part of the lecture, the use of deep neural networks for speech processing tasks is introduced. The course is based on the materials and video footage from Dr. Roland Maas. He is an outstanding machine learning expert and a former member of the Chair of Multimedia Communications and Signal Processing.
6	**Lernziele und Kompetenzen**	After attending the lecture, students will be able to • understand regression and classification problems • apply PDF estimation algorithms • understand Gaussian mixture models and expectation-maximization • apply principal component analysis and independent component analysis • assess different estimation algorithms • explain the application of machine learning to system identification • apply hidden Markov models • understand different artificial neural network architectures • explain deep learning principles • apply artificial neural networks • devise learning strategies for deep neural networks • assess the application of deep neural networks for speech processing tasks.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1

9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich/mündlich
11	**Berechnung der Modulnote**	schriftlich/mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Literature: • C. M. Bishop: Pattern Recognition and Machine Learning, http://www.research.microsoft.com/en-us/um/people/cmbishop/PRML • S. Theodoridis and K. Koutroumbas: Pattern Recognition • M. Nielsen: Neural Networks and Deep Learning.

1	**Modulbezeichnung** 96030	**Medizinelektronik** (Medical electronics)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Medizinelektronik - Übung / Medical Electronics Exercises (2 SWS) Vorlesung: Medizinelektronik - Medical Electronics (2 SWS)	- 5 ECTS
3	Lehrende	Thomas Kurin Dr. Jens Kirchner	

4	**Modulverantwortliche/r**	Dr. Jens Kirchner
5	**Inhalt**	The Lecture and exercise deals with the following topics: • Electronics for medical diagnostics and therapy • Challenges for medical engineering from demographic development and epidemiology of common diseases • Concepts for chronic disease management and elderly care • Regulatory framework of circuit design for medical devices • Circuit design of standard medical equipment ECG, EEG, EMG, SpO2 • Sensor principles and circuit design for biosignal acquisition • Analog-digital balance • Energy management for medical devices • Body near energy harvesting • Health data transmission • Electronic systems for ambient assisted living (AAL) • Circuit technology for lab-on-chip and microelectromechanical systems (MEMS) • Circuit technology for implants and wearable systems
6	**Lernziele und Kompetenzen**	Students will gain • Substantial knowledge on principles of circuit design for medical electronic devices • Substantial knowledge on circuit design for standard medical devices, e.g. ECG, EEG, EMG • Substantial knowledge on design of medical sensors • Substantial knowledge on system design for health assistance systems, wearable medical devices and implants • Ability to analyze circuit diagrams of medical electronic devices • Ability to separate medical electronic devices into their subfunctions • Ability to analyze energy budget of medical devices, particularly wearable systems • Basic ability to design electronic circuits to comply with regulatory requirements
7	**Voraussetzungen für die Teilnahme**	Completion of the modules "Circuit design" ("Schaltungstechnik") or "Electronics and circuit design" ("Elektronik und Schaltungstechnik") is recommended before attending the course.
8	**Einpassung in Studienverlaufsplan**	Semester: 1

9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 43460	**Satellitenkommunikation** (Satellite communication)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übung Satellitenkommunikation (2 SWS) Vorlesung: Satellitenkommunikation (2 SWS)	- 5 ECTS
3	Lehrende	Marcelo Michael Dr. Christian Rohde	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Albert Heuberger
5	**Inhalt**	Nach einem historischen Rückblick zur Entwicklung der Satellitenkommunikation werden die einzelnen Komponenten eines typischen Gesamtsystems (Boden- und Raumsegment) näher betrachtet. Hierzu zählt der prinzipielle Aufbau von Trägerraketen, von Satelliten (Satellitenplattformen, Subsysteme, Nutzlasten), die meist genutzten Umlaufbahnen und die verschiedenen Kommunikationsverbindungen (Uplink, Downlink, Inter-Satellite-Link). Die Besonderheiten der Signalausbreitung und -übertragung über große Entfernungen zwischen Bodenstationen und Satelliten werden erklärt und mit Beispielen ergänzt. Dabei wird insbesondere eingegangen auf verwendete Frequenzen, Signaldispersion und -dämpfung, atmosphärische Effekte sowie Störeinflüsse der Weltraumumgebung. Die Architektur transparenter und regenerativer Kommunikationseinheiten wird ausführlich an Beispielen kommerziell verfügbarer Transponder und Onboard-Prozessoren erklärt. Die Prinzipien moderner, standardisierter Verfahren zur Signalaufbereitung und Übertragung von Video-/Bild und Audiosignalen über Satellit (z.B. MPEG, H.264/265, DVB-S/-S2/-S2X) werden erläutert und diskutiert. Dies umfasst Verfahren zur Quellencodierung, Kanalcodierung und Modulation, Kanalzugriff und -diversität. Außerdem wird auf die im Orbit und im kommerziellen Einsatz befindlichen Kommunikationssatelliten und der damit verbundenen großen Dienstevielfalt eingegangen wie z.B. bei TV- und Breitbandversorgung sowie in Mobilkommunikationssystemen. Abschließend werden einige Herausforderungen und Forschungsansätze im Zusammenhang mit den neuen Megakonstellationen und Next Generation High Throughput Satellites (HTS) für zukünftige Satellitensysteme vorgestellt. Die in der Vorlesung behandelten physikalischen, elektro- und nachrichtentechnischen Zusammenhänge werden in den ergänzenden Übungen mit Rechenbeispielen vertieft. Gliederung der Vorlesung: **1. Einführung:**

Überblick über die Hauptkomponenten, Satelliten, Anwendungen und Dienste, sowie Orbits, Aufgaben und Frequenzen der Satellitennetzwerke

2. Historie der Satellitenkommunikation:

Wichtige Meilensteine, Entwicklung in Europa und Deutschland

3. Orbits und Konstellationen:

Keplersche Gesetze, Beschreibung von Orbits, verwendete Umlaufbahnen, Bodenspuren, erreichbare Abdeckung

4. Trägersysteme:

Trägerraketen, Entwicklung, Anbietermarkt, Nutzlastfähigkeit, Startplätze, Startverlauf

5. Satellitenaufbau:

Auswahl aktueller Satellitenplattformen, Satellitenaufbau, Plattformkomponenten, Montageschritte und Tests

6. Satellitennutzlast (Payload):

Komponenten, Industrielle Beispiele, Aufbau und Aufgaben der Payload, Transponderarchitekturen, Antennen

7. Signalausbreitung und Leistungsbilanz:

Signalausbreitung, Freiraumverluste, Signaldämpfung, Rauschen, Signal-Rausch-Verhältnis, Linkbudget

8. Weltraumumgebung: Weltraumumgebungsbedingungen, Einflüsse auf den Satelliten und die Elektronik der Nutzlast

9. Quellencodierung:

Audio-, Bild- und Videokompression das Content des Satellitenfernsehens

10. Signalmodulation und Kanalcodierung:

Signalkonstellationen, Modulation und Codes zur Fehlerkorrektur

11. Diversitäts- und Zugriffsverfahren:

Medium Access, Duplextechniken, Multiplexmethoden, Diversitätstechniken

12. Moderne Satellitenkommunikationssysteme:

Rundfunksysteme wie Sirius XM Satellite Radio, zellulare Internetversorgung mittels Satellitenkommunikation

13. Neueste Themen aus Forschung und Entwicklung

SatKom auf StudOn: http://www.studon.uni-erlangen.de/crs117969.html

After a historical retrospective about the developments in satellite communication, the core components of a typical satellite system (ground- and space-segment) are introduced. The principles and architectures of rockets/ carriers, satellites (platform, subsystems, payload), used orbits, and the various communication links (uplink, downlink, inter-satellite-link) are shown. The special features and properties of signal transmission over such large distances are explained and stuffed with examples. In particular, more details are provided on the used frequencies, signal dispersion and attenuation, atmospheric effects as well as impairments due to space environment.

The architecture of transparent and regenerative communication payloads are described in detail, accompanied by corresponding examples of commercially used transponders and onboard-processors and their technology.

The principles of modern standardized methods for signal transmission and preparation of video-/image- and audio-signals via satellite, e.g., MPEG, H.264/265, DVB-S/-S2/-S2X, are illustrated and discussed. This includes methods for efficient source coding, channel coding and modulation, channel access and diversity schemes.

Furthermore, the currently available communication satellites in orbit and the related variety of commercial services are introduced like, e.g., TV- and broadband services as well as mobile communication services and systems. Based on that, a few challenges and perspectives for research and development for future satellite systems are highlighted with respect to the upcoming new mega constellations and next generation high throughput satellites (HTS).

		The physical, electro-technical and communications concepts and schemes shown in the lectures are complemented by tutorials with sample calculations. Table of contents: **1. Introduction:** Overview of main components, satellites, applications and services, orbits, tasks, frequencies, satellite networks **2. History of satellite communications:** Major milestones, development in Europe and Germany **3. Orbits and constellations:** Kepler's laws, description of orbits, orbits used, ground tracks, achievable coverage **4. Launcher systems:** Launch vehicles, providers, payload capabilities, launch sites, launch history **5. Satellite structure:** Selection of current satellite platforms, satellite structure, platform components, assembly steps and tests **6. Payload:** Components, structure and tasks of payload, transponder architecture, antennas **7. Signal propagation and link budget:** Signal propagation, free space losses, signal attenuation, noise, signal to noise ratio, link budget **8. Space environment:** Space environmental conditions, influences on the satellites and payload electronics **9. Source coding:** Audio, image and video compression - the satellite TV broadcasting content **10. Signal modulation and channel coding:** Signal constellations, modulation and error correction coding **11. Diversity and access schemes:** Medium access, duplex methods, multiplex methods, diversity techniques **12. Modern satellite communications systems:** Broadcasting systems like Sirius XM Satellite Radio, satellite cellular broadband communication **13. Latest topics in research and development**
6	**Lernziele und Kompetenzen**	• Die Studierenden bekommen einen guten Überblick über alle Aspekte der Satellitenkommunikation inklusive Historie.

		• Die Studierenden lernen die weltweit führenden oder in Europa ansässigen Firmen und Organisationen kennen, die in den Bereichen Satellitenbau und -betrieb, Satellitendienste bzw. -anwendungen, sowie Forschung und Entwicklung tätig sind. • Die Studierenden können die Herausforderungen der Weltraumumgebung sowie Vor- und Nachteile verschiedener Orbits einschätzen und wichtige Kenngröße berechnen • Die Studierenden lernen die Signalverarbeitungsschritte im Sender, Satelliten und Empfänger kennen - von der Audio/Video-Quelle über Link-Budget-Berechnungen bis zur Datensenke. • Die Studierenden lernen den Aufbau und wichtige Kenngrößen von Satelliten, Konstellationen und Launchern kennen und dabei verwendete Konzepte zu unterscheiden und zu klassifizieren bzgl. deren Vor- und Nachteilen.
7	**Voraussetzungen für die Teilnahme**	Keine formalen Voraussetzungen
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich (91 Minuten)
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Skriptum zur Lehrveranstaltung

1	**Modulbezeichnung** 96460	**Speech and Audio Signal Processing** (Speech and audio signal processing)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übung zur Sprach- und Audiosignalverarbeitung (1 SWS) Vorlesung: Sprach- und Audiosignalverarbeitung (3 SWS)	- 5 ECTS
3	Lehrende	Mhd Modar Halimeh Prof. Dr.-Ing. Walter Kellermann	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Walter Kellermann
5	**Inhalt**	It concentrates on algorithms for speech and audio signal processing with applications in telecommunications and multimedia, especially • physiology and models for human speech production and hearing: source-filter model, filterbank model of the cochlea, masking effects, • representation of speech and audio signals: estimation and representation of short-term and long-term statistics in the time and frequency domain as well as the cepstral domain; typical examples and visualizations • source coding for speech and audio signals: criteria, scalar and vector quantization, linear prediction, prediction of the pitch frequency; waveform coding, parametric coding, hybrid coding, codec standards (ITU, GSM, ISO-MPEG) • basic concepts of automatic speech recognition (ASR): feature extraction, dynamic time warping, Hidden Markov Models (HMMs) • basic concepts of speech synthesis: text-to-speech systems, model-based and data-driven synthesis, PSOLA synthesis system • signal enhancement for acquisition and reproduction: noise reduction, acoustic echo cancellation, dereverberation using single-channel and multichannel algorithms. Es werden Grundlagen und Algorithmen der Verarbeitung von Sprach- und Audiosignalen mit Anwendungen in Telekommunikation und Multimedia behandelt, insbesondere: • Physiologie und Modelle der Spracherzeugung und des Hörens: Quelle-Filter-Modell, Filterbank-Modell der Cochlea; Maskierungseffekte; • Darstellung von Sprach- und Audiosignalen: Schätzung und Darstellung der Kurzzeit- und Langzeitstatistik in Zeit-, Frequenz- und Cepstralbereich; typische Beispiele, Visualisierungen; • Quellencodierung für Sprache und Audiosignale: Kriterien; skalare und vektorielle Codierung; lineare Prädiktion; Pitchprädiktion; Wellenform-/Parameter-/Hybrid-Codierung; Standards (ITU, GSM, ISO-MPEG)

		• Spracherkennung: Merkmalextraktion, Dynamic Time Warping, Hidden Markov Models • Grundprinzipien der Sprachsynthese: Text-to-Speech Systeme, modellbasierte und datenbasierte Synthese, PSOLA-Synthese • Signalverbesserung bei Signalaufnahme und wiedergabe: Geräuschbefreiung, Echokompensation, Enthallung mittels ein- und mehrkanaliger Verfahren;
6	**Lernziele und Kompetenzen**	The students • understand basic physiological mechanisms of human speech production and hearing and can apply them for the analysis of speech and audio signals • apply basic methods for the estimation and representation of the short-term and long-term statistics of speech and audio signals and can analyze such signals by means of these methods • understand current methods for source coding of speech and audio signals and can analyze current coding standards • verstehen die Grundbausteine von Spracherkennungssystemen und können deren Funktion mittels Rechnersimulation analysieren • understand the basic principle of text-to-speech systems and can apply fundamental methods for speech synthesis • can apply basic algorithms for speech enhancement and understand their functionality for real-world data. Die Studierenden • verstehen die grundlegenden physiologischen Mechanismen der Spracherzeugung und des Hörens beim Menschen und können diese zur Analyse von Sprach- und Audiosignalen anwenden • wenden die grundlegenden Methoden zur Schätzung und Darstellung der Kurzzeit- und Langzeitstatistik von Sprach- und Audiosignalen an und können diese damit analysieren • verstehen die aktuellen Methoden zur Quellencodierung von Sprache-und Audiosignalen und können aktuelle Codierstandards analysieren • verstehen die Grundbausteine von Spracherkennungssystemen und können deren Funktion mittels Rechnersimulation analysieren • verstehen die Grundprinzipien von Text-to-Speech Systemen und können elementare Algorithmen zur Sprachsynthese anwenden • können elementare Algorithmen zur Signalverbesserung anwenden und für reale Daten analysieren
7	**Voraussetzungen für die Teilnahme**	Vorlesung Signale und Systeme I & II

8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich (90 Minuten)
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Gemäß themenbezogenen Angaben in der Lehrveranstaltung

1	**Modulbezeichnung** 96621	**Transceiver-Systementwurf** (keine englischsprachige Modulbezeichnung hinterlegt!)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Transceiver-Systementwurf (2 SWS) Übung: Übung Transceiver-Systementwurf (2 SWS)	5 ECTS -
3	Lehrende	Prof. Dr. Jörn Thielecke	

4	**Modulverantwortliche/r**	Prof. Dr. Jörn Thielecke
5	**Inhalt**	1. Einführung 2. Systemübersicht und Anforderungen • GPSGSM-WLAN • Vergleichende Zusammenfassung 3. Basisbandverarbeitung • Optimierung und Wechselwirkungen am Beispiel einer PLL • Anforderungsprofil bei GPS, GSM und WLAN 4. A/D- und D/A-Umsetzung • Dominierendes Nutzsignal bei GSM und WLAN • Dominierendes Rauschen bei GPS • Anforderungsübersicht 5. Frontend • Analyse und Charakterisierung von Störungen (Nichtlinearitäten, Rauschen, Dynamikbereicht, I/Q-Balance, Phasenrauschen) • Systementwurf (Entwurfzyklus, Empfänger-Architekturen, Sender-Architekturen) 6. Ausblick
6	**Lernziele und Kompetenzen**	1. Anhand der Beispielsysteme GPS, GSM und WLAN sollen Studierende beurteilen lernen, wie das Wechselspiel zwischen Realisierungsaufwand und nachrichtentechnischer Systemanforderung ist. 2. Anhand von Beispielen sollen Studierenden die wesentlichen Entwurfsschritte bis hin zur Parametrisierung auf Blockschaltbildebene klar werden, wenn der Ausgangspunkt eine nachrichtentechnische Systembeschreibung ist. 3. Anhand von Architekturbeispielen sollen Studierende ein Verständnis für die Spielräume und Abwägungen beim Entwurf eines Endgerätes entwickeln.

7	**Voraussetzungen für die Teilnahme**	Keine formalen Voraussetzungen, grundlegende Kenntnisse erforderlich aus: Signal- und Systemtheorie, Nachrichtentechnische Systeme, Stochastik. Erste mikroelektronische Kenntnisse helfen.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich (90 Minuten)
11	**Berechnung der Modulnote**	mündlich (100%) Bei bestandener Prüfung wird die Note um eine Teilnotenstufe (z.B. von 2,0 auf 1,7) verbessert, wenn Sie mindestens 75% der Hausaufgaben einschließlich der Rechnerübungen erfolgreich absolviert haben. Eine Note besser als 1,0 wird nicht vergeben.
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Skriptum zur Vorlesung.

Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems

1	**Modulbezeichnung** 820947	**Betriebssysteme (Vorlesung mit erweiterten Übungen)** (Operating systems (lectures with extend exercises))	**7,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Betriebssysteme (2 SWS) Übung: Betriebssysteme - Rechnerübungen (0 SWS) Übung: Betriebssysteme - Erweiterte Übungen (4 SWS)	2,5 ECTS - 5 ECTS
3	Lehrende	Dr.-Ing. Volkmar Sieh Bernhard Heinloth Dustin Nguyen Phillip Raffeck	

4	**Modulverantwortliche/r**	Dr.-Ing. Volkmar Sieh
5	**Inhalt**	Inhalt des Moduls ist Vermittlung grundlegender Konzepte, Methoden und Techniken von Betriebssystemen aus dem Blickwinkel einer Betriebssystementwicklerin. Im Rahmen der Übungen entwickeln die Studierenden in einem "bottom-up" Entwurf- und Entwicklungsprozess ihr eigenes Mehrkernbetriebssystem für die IA-32 Platform, ausgehend von der "nackten Hardware" über grundlegende Ein-Ausgabemöglichleiten, Unterbrechungsbearbeitung bis hin zu quasi- und echtparalleler Programmausführung.
6	**Lernziele und Kompetenzen**	Studierende, die das Modul erfolgreich abgeschlossen haben: - erläutern den Startvorgang eines Rechensystems am Beispiel eines x86/x86_64-PCs. - beschreiben die spezifischen Herausforderungen bei der Softwareentwicklung für "bare metal". - beschreiben den Ablauf einer Unterbrechungsbehandlung von der Hardware bis zur (System-)software. - skizzieren Besonderheiten und Strategien der Unterbrechungsbehandlung in Hardware für Mehrkernsystemen am Beispiel des IA32-APICs. - diskutieren die Aufgabenteilung zwischen Hardware und Systemsoftware bei der Unterbrechungsbearbeitung. - unterscheiden die verschiedenen Typen von Kontrollflüssen in einem Betriebssystem anhand des Ebenenmodells. - unterscheiden harte, mehrstufige, und weiche Verfahren zur Unterbrechungssynchronisation in Betriebssystemen und können diese für ein Mehrkernsystem implementieren. - klassifizieren konkrete Konkurrenzsituationen anhand des Ebenenmodels und leiten daraus geeignete Synchronisationsmaßnahmen ab.

- schildern die IA32-Architektur und gängige PC-Technologie und deren Schnittstellen zur Systemsoftware.

- erläutern grundlegende Bausteine für die Implementierung von Quasi-Parallelität (Fortsetzungen, Koroutinen, Fäden) und grenzen diese gegeneinander ab.

- erläutern die Interaktionen zwischen Hardware, Übersetzer und Systemsoftware, die dabei zu beachten sind.

- entwickeln den Koroutinenwechsel für einen gegebene Architektur.

- erläutern die Implikationen von Quasi-Parallität auf das Ebenenmodell und die daraus abgeleiteten Synchronisationsmaßnahmen.

- beschreiben die Implementierung von (verdrängendem) Scheduling in einem Betriebssystem.

- analysieren das Zusammenspiel von Scheduling und Unterbrechungssynchronisation.

- nennen Kriterien und Dimensionen des Schedulings von Betriebsmitteln, insbesondere der CPU.

- erläutern die konkrete Umsetzung am Beispiel der Scheduler in Linux und Windows.

- unterscheiden grundlegende Möglichkeiten der Koordinierung und Synchronisation von Fäden (aktives/passives Warten, nichtverdrängbare kritische Abschnitte).

- entwickeln Mechanismen für die Synchronisation auf Fadenebene für Mehrkernsystemen.

- erklären die dabei zu beachtenden Synchronisationsprobleme (lost update, lost wakeup) und geeignete Gegenmaßnahmen.

- interpretieren die Bedeutung von Gerätetreibern in der Betriebssystempraxis.

- erläutern die Anforderungen an ein Treibermodell.

- vergleichen die Umsetzung von Treibermodellen in Windows und Linux.

- vergleichen grundlegende BS-Architekturen (Bibliothek, Monolith, Mikrokern, Exokern, Hypervisor) anhand fundamentaler Charakteristika (Robustheit, Performanz, Portierbarkeit) und Mechanismen.

		- schildern die grundlegenden Paradigmen zur Interprozesskommunikation in Betriebssystemen (speicherbasiert vs. nachrichtenbasiert). - erläutern die grundlegenden Primitiven dieser Verfahren. - skizzieren, wie unter Anwendung dieser Primitiven höhere Synchronisationskonstrukte implementiert werden (Monitore, Leser-/ Schreiber-Sperre). - illustrieren die Dualität der Paradigmen. - erschließen sich typische Probleme (Nebenläufigkeit, Compilerverhalten, Debuggen ohne dedizierte Hilfsmittel) und Fehlerquellen bei der hardwarenahen Softwareentwicklung für Mehrkernsysteme. - können in Kleingruppen kooperativ arbeiten. - können die ihre Entwurfs- und Implementierungsentscheidungen kompakt präsentieren und argumentativ vertreten. - reflektieren ihre Entscheidungen kritisch und leiten Alternativen ab. - können offen und konstruktiv mit Schwachpunkten und Irrwegen umgehen.
7	**Voraussetzungen für die Teilnahme**	• C/C++ (Übungsaufgaben werden in C++ implementiert) • Assembler (Grundkenntnisse)
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 90 h Eigenstudium: 135 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• Avi Silberschatz, Peter B. Galvin, Greg Gagne. Operating System Concepts. John Wiley & Sons. • Andrew S. Tanenbaum. Modern Operating Systems. Pearson.

		• William Stallings. Operating Systems: Internals and Design Principles. Prentice Hall. • Andrew S. Tanenbaum. Structured Computer Organization. Pearson.

1	**Modulbezeichnung** 150033	**Betriebssysteme (Vorlesung mit Übungen)** (Operating systems (lecture with exercises))	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Betriebssysteme (2 SWS) Übung: Betriebssysteme - Übungen (2 SWS) Übung: Betriebssysteme - Rechnerübungen (0 SWS)	2,5 ECTS 2,5 ECTS -
3	Lehrende	Dr.-Ing. Volkmar Sieh Bernhard Heinloth Dustin Nguyen Phillip Raffeck	

4	**Modulverantwortliche/r**	Dr.-Ing. Volkmar Sieh
5	**Inhalt**	Inhalt des Moduls ist Vermittlung grundlegender Konzepte, Methoden und Techniken von Betriebssystemen aus dem Blickwinkel einer Betriebssystementwicklerin. Im Rahmen der Übungen entwickeln die Studierenden in einem "bottom-up" Entwurf- und Entwicklungsprozess ihr eigenes Einkernbetriebssystem für die IA-32 Platform, ausgehend von der "nackten Hardware" über grundlegende Ein-Ausgabemöglichleiten, Unterbrechungsbearbeitung bis hin zu quasiparalleler Programmausführung.
6	**Lernziele und Kompetenzen**	Studierende, die das Modul erfolgreich abgeschlossen haben: - erläutern den Startvorgang eines Rechensystems am Beispiel eines x86/x86_64-PCs. - beschreiben die spezifischen Herausforderungen bei der Softwareentwicklung für "bare metal". - beschreiben den Ablauf einer Unterbrechungsbehandlung von der Hardware bis zur (System-)software. - skizzieren Besonderheiten und Strategien der Unterbrechungsbehandlung in Hardware für Mehrkernsystemen am Beispiel des IA32-APICs. - diskutieren die Aufgabenteilung zwischen Hardware und Systemsoftware bei der Unterbrechungsbearbeitung. - unterscheiden die verschiedenen Typen von Kontrollflüssen in einem Betriebssystem anhand des Ebenenmodells. - unterscheiden harte, mehrstufige, und weiche Verfahren zur Unterbrechungssynchronisation in Betriebssystemen und können diese implementieren. - klassifizieren konkrete Konkurrenzsituationen anhand des Ebenenmodels und leiten daraus geeignete Synchronisationsmaßnahmen ab.

- schildern die IA32-Architektur und gängige PC-Technologie und deren Schnittstellen zur Systemsoftware.

- erläutern grundlegende Bausteine für die Implementierung von Quasi-Parallelität (Fortsetzungen, Koroutinen, Fäden) und grenzen diese gegeneinander ab.

- erläutern die Interaktionen zwischen Hardware, Übersetzer und Systemsoftware, die dabei zu beachten sind.

- entwickeln den Koroutinenwechsel für einen gegebene Architektur.

- erläutern die Implikationen von Quasi-Parallität auf das Ebenenmodell und die daraus abgeleiteten Synchronisationsmaßnahmen.

- beschreiben die Implementierung von (verdrängendem) Scheduling in einem Betriebssystem.

- analysieren das Zusammenspiel von Scheduling und Unterbrechungssynchronisation.

- nennen Kriterien und Dimensionen des Schedulings von Betriebsmitteln, insbesondere der CPU.

- erläutern die konkrete Umsetzung am Beispiel der Scheduler in Linux und Windows.

- unterscheiden grundlegende Möglichkeiten der Koordinierung und Synchronisation von Fäden (aktives/passives Warten, nichtverdrängbare kritische Abschnitte).

- entwickeln Mechanismen für die Synchronisation auf Fadenebene.

- erklären die dabei zu beachtenden Synchronisationsprobleme (lost update, lost wakeup) und geeignete Gegenmaßnahmen.

- interpretieren die Bedeutung von Gerätetreibern in der Betriebssystempraxis.

- erläutern die Anforderungen an ein Treibermodell.

- vergleichen die Umsetzung von Treibermodellen in Windows und Linux.

- vergleichen grundlegende BS-Architekturen (Bibliothek, Monolith, Mikrokern, Exokern, Hypervisor) anhand fundamentaler Charakteristika (Robustheit, Performanz, Portierbarkeit) und Mechanismen.

		- schildern die grundlegenden Paradigmen zur Interprozesskommunikation in Betriebssystemen (speicherbasiert vs. nachrichtenbasiert). - erläutern die grundlegenden Primitiven dieser Verfahren. - skizzieren, wie unter Anwendung dieser Primitiven höhere Synchronisationskonstrukte implementiert werden (Monitore, Leser-/ Schreiber-Sperre). - illustrieren die Dualität der Paradigmen. - erschließen sich typische Probleme (Nebenläufigkeit, Compilerverhalten, Debuggen ohne dedizierte Hilfsmittel) und Fehlerquellen bei der hardwarenahen Softwareentwicklung. - können in Kleingruppen kooperativ arbeiten. - können die ihre Entwurfs- und Implementierungsentscheidungen kompakt präsentieren und argumentativ vertreten. - reflektieren ihre Entscheidungen kritisch und leiten Alternativen ab. - können offen und konstruktiv mit Schwachpunkten und Irrwegen umgehen.
7	**Voraussetzungen für die Teilnahme**	• C/C++ (Übungsaufgaben werden in C++ implementiert) • Assembler (Grundkenntnisse)
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• Avi Silberschatz, Peter B. Galvin, Greg Gagne. Operating System Concepts. John Wiley & Sons. • Andrew S. Tanenbaum. Modern Operating Systems. Pearson. • William Stallings. Operating Systems: Internals and Design Principles. Prentice Hall.

		• Andrew S. Tanenbaum. Structured Computer Organization. Pearson.

1	**Modulbezeichnung** 211243	**CPU Entwurf mit VHDL (CPU)** (CPU Design with VHDL (CPU))	**7,5 ECTS**
2	Lehrveranstaltungen	Zu diesem Modul sind keine Lehrveranstaltungen oder Lehrveranstaltungsgruppen hinterlegt!	
3	Lehrende	Zu diesem Modul sind keine Lehrveranstaltungen und somit auch keine Lehrenden hinterlegt!	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Dietmar Fey
5	**Inhalt**	Die Vorlesung wird einen Einblick in die wichtigsten Strukturen eines Rechners und insbesondere die der CPU geben. Wie sind die Register, die ALU, die Caches usw. aufgebaut, wie sind sie miteinander verschaltet? Wie werden Pipelines gebildet? Um die Vorlesung verstehen und die Übungsaufgaben lösen zu können, werden VHDL-Kenntnisse benötigt. Diese werden in der Vorlesung vermittelt. Im ersten Teil der Vorlesung geht es darum, aus einfachen Und-, Oder- und Inverter-Gattern größere Strukturen aufzubauen. So werden bespielsweise die Register und die Komponenten der ALU (Addierer, Multiplizierer, Shifter usw.) nach und nach aufgebaut. Aus den Einzelteilen lassen sich dann wiederum die Registerbänke, die ALU, die Speichereinheit usw. zusammensetzen. Mit guten Rechenwerken allein ist eine CPU noch nicht wirklich schnell. Es müssen schnell genug Instruktionen gelesen und dekodiert sowie die Operanden geholt werden können. Nach dem schnellen Verarbeiten in den Rechenwerken müssen die Ergebnisse dann wiederum schnell in die Zielregister bzw. Ziel-Speicherzellen kopiert werden. Das wichtigste Verfahren, um hierbei Performance zu gewinnen, ist das Pipelining. Leider gibt es eine Reihe von Bedingungen, die erfüllt sein müssen, dass Pipelining wirklich gut funktioniert. So dürfen keine Daten-Abhängigkeiten im Code vorhanden sein, müssen Sprungziele rechtzeitig bekannt sein und dürfen bestimmte Ressourcen nicht mehrfach gleichzeitig genutzt werden. Mittels Forwarding, Sprungvorhersage und Caches können die Probleme gemildert werden. Mit den so gewonnenen Fähigkeiten sind dann auch Multi-Core- und Multi-Threading-Architekturen verständlich.
6	**Lernziele und Kompetenzen**	Die Studierenden verstehen den inneren Aufbau moderner CPUs (Pipeline, Registerbänke, ALU, Caches, Memory-Management-Unit, Segmentierungseinheit, ...) und können selbst performante CPUs aus einfachen Basis-Schaltungen aufbauen.
7	**Voraussetzungen für die Teilnahme**	keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1

9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 90 h Eigenstudium: 135 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 169383	**CPU Entwurf mit VHDL (Schwerpunkt Prozessorentwurf)** (CPU Design with VHDL (Focus on CPUs))	**5 ECTS**
2	Lehrveranstaltungen	Zu diesem Modul sind keine Lehrveranstaltungen oder Lehrveranstaltungsgruppen hinterlegt!	
3	Lehrende	Zu diesem Modul sind keine Lehrveranstaltungen und somit auch keine Lehrenden hinterlegt!	

4	**Modulverantwortliche/r**	
5	**Inhalt**	keine Inhaltsbeschreibung hinterlegt!
6	**Lernziele und Kompetenzen**	keine Beschreibung der Lernziele und Kompetenzen hinterlegt!
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	keine Angaben zum Turnus des Angebots hinterlegt!
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: ?? h (keine Angaben zum Arbeitsaufwand in Präsenzzeit hinterlegt) Eigenstudium: ?? h (keine Angaben zum Arbeitsaufwand im Eigenstudium hinterlegt)
14	**Dauer des Moduls**	?? Semester (keine Angaben zur Dauer des Moduls hinterlegt)
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 436348	**CPU Entwurf mit VHDL (Schwerpunkt VHDL)** (CPU Design with VHDL (Focus on VHDL))	**5 ECTS**
2	Lehrveranstaltungen	Zu diesem Modul sind keine Lehrveranstaltungen oder Lehrveranstaltungsgruppen hinterlegt!	
3	Lehrende	Zu diesem Modul sind keine Lehrveranstaltungen und somit auch keine Lehrenden hinterlegt!	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Dietmar Fey
5	**Inhalt**	Die Vorlesung wird einen Einblick in die wichtigsten Strukturen eines Rechners und insbesondere die der CPU geben. Wie sind die Register, die ALU, die Caches usw. aufgebaut, wie sind sie miteinander verschaltet? Wie werden Pipelines gebildet? Um die Vorlesung verstehen und die Übungsaufgaben lösen zu können, werden VHDL-Kenntnisse benötigt. Diese werden in der Vorlesung vermittelt. Im ersten Teil der Vorlesung geht es darum, aus einfachen Und-, Oder- und Inverter-Gattern größere Strukturen aufzubauen. So werden bespielsweise die Register und die Komponenten der ALU (Addierer, Multiplizierer, Shifter usw.) nach und nach aufgebaut. Aus den Einzelteilen lassen sich dann wiederum die Registerbänke, die ALU, die Speichereinheit usw. zusammensetzen. Mit guten Rechenwerken allein ist eine CPU noch nicht wirklich schnell. Es müssen schnell genug Instruktionen gelesen und dekodiert sowie die Operanden geholt werden können. Nach dem schnellen Verarbeiten in den Rechenwerken müssen die Ergebnisse dann wiederum schnell in die Zielregister bzw. Ziel-Speicherzellen kopiert werden. Das wichtigste Verfahren, um hierbei Performance zu gewinnen, ist das Pipelining. Leider gibt es eine Reihe von Bedingungen, die erfüllt sein müssen, dass Pipelining wirklich gut funktioniert. So dürfen keine Daten-Abhängigkeiten im Code vorhanden sein, müssen Sprungziele rechtzeitig bekannt sein und dürfen bestimmte Ressourcen nicht mehrfach gleichzeitig genutzt werden. Mittels Forwarding, Sprungvorhersage und Caches können die Probleme gemildert werden. Mit den so gewonnenen Fähigkeiten sind dann auch Multi-Core- und Multi-Threading-Architekturen verständlich.
6	**Lernziele und Kompetenzen**	Die Studierenden verstehen den inneren Aufbau moderner CPUs (Pipeline, Registerbänke, ALU, Caches, Memory-Management-Unit, Segmentierungseinheit, ...) und können selbst performante CPUs aus einfachen Basis-Schaltungen aufbauen.
7	**Voraussetzungen für die Teilnahme**	keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1

9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 939179	**Echtzeitsysteme 2 - Verlässliche Echtzeitsysteme** (Real-time systems 2 - dependable real-time systems)	**7,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Echtzeitsysteme 2 - Verlässliche Echtzeitsysteme (2 SWS) Übung: Echtzeitsysteme 2 - Verlässliche Echtzeitsysteme - Erweiterte Übungen (2 SWS) Übung: Echtzeitsysteme 2 - Verlässliche Echtzeitsysteme - Rechnerübungen (2 SWS)	- - -
3	Lehrende	Peter Wägemann Prof. Dr.-Ing. Wolfgang Schröder-Preikschat Tim Rheinfels Simon Schuster	

4	**Modulverantwortliche/r**	Peter Wägemann
5	**Inhalt**	Viele Echtzeitsysteme sind in Bereiche des täglichen Lebens eingebettete, die hohe Anforderungen an die funktionale Sicherheit dieser Systeme stellen. Beispiele hierfür sind Fahrerassistenzsysteme in modernen Automobilen, medizinische Geräte, Prozessanlagen in Kernkraftwerken oder Chemiefabriken oder Flugzeuge. Fehlfunktionen in diesen Anwendungen ziehen mitunter katastrophale Konsequenzen nach sich - Menschen können ernsthaft verletzt oder sogar getötet werden, Landstriche können unbewohnbar gemacht oder zumindest großer finanzieller Schaden verursacht werden. Dieses Modul betrachtet Methoden und Werkzeuge, die uns helfen können, einerseits *zuverlässig Software zu entwickeln* (also Fehler im Programm zu entdecken und zu vermeiden), und andererseits *zuverlässige Software zu entwickeln* (also Abstraktionen, die auch im Fehlerfall ihre Gültigkeit behalten). Hierbei steht weniger die Vermittlung theoretischer Grundkenntnisse auf diesen Gebieten im Vordergrund, also vielmehr • die praktische Anwendung existierende Werkzeuge und Methoden • sowie die Erfahrung und das Verständnis ihrer Grenzen. Auf diese Weise soll ein Fundament für die konstruktive Umsetzung verlässlicher Echtzeitsysteme gelegt werden. Dieses Modul soll daher fundierte Anknüpfungspunkte für die Entwicklung verlässlicher Echtzeitsysteme vermitteln, die Ad-hoc-Techniken möglichst ersetzen sollen.
6	**Lernziele und Kompetenzen**	Studierende, die das Modul erfolgreich abgeschlossen haben: • nennen die Konzepte und die Taxonomie verlässlicher Systeme, unterscheiden Software- und Hardwarefehler und klassifizieren Fehler (Defekt, Fehler, Fehlverhalten). • stellen Fehlerbäume auf. • organisieren Softwareentwicklungsprojekte mittels der Versionsverwaltung git.

		• vergleichen die verschiedenen Arten der Redundanz als Grundvorraussetzung für Fehlererkennung und -toleranz. • entwickeln fehlertolerante Systeme mittels Replikation. • diskutieren die Fehlerhypothese und die Sicherstellung von Replikdeterminismus. • erläutern die Vor- und Nachteile softwarebasierter Replikation und den Einsatz von Diversität. • wenden Informationsredundanz zur Härtung von Daten- und Kontrollflüssen an. • bewerten die Effektivität der arithmetischer Codierung von Programmen und verallgemeinern diesen Ansatz auf die verschiedenen Implementeirungsebenen (Maschinenprogramm zu Prozessinkarnation). • interpretieren den Einfluss der Ausführungsplattform (Hardware, Betriebssystem) auf die Leistungsfähigkeit der Fehlererkennung. • konzipieren eine fehlertolerante Ausführungsumgebung für ein softwarebasiertes TMR-System basierend auf ANBD-Codierung. • nennen die Grundlagen der systematischen Fehlerinjektion. • überprüfen die Wirksamkeit von Fehlertoleranzmechanismen mittels Fehlerinjektion auf der Befehlssatzebene. • entwickeln Testfälle für die Fehlerinjektion mittels des fail** Werkzeugs. • setzten Messergebnisse in Relation zu dem tatsächlichen Fehlerraum. • beschreiben die Grundlagen der Fehlererholung (Vorwärts- bzw. Rückwärtskorrektur) und Reintegration fehlgeschlagener Knoten. • vergleichen den Zustandstransfer am Beispiel der Running bzw. Recursive State Restoration. • benennen Konzepte der Rückwärtskorrektur durch Entwurfsalternativen (Recovery Blocks). • fassen die Grundlagen des dynamischen Testens zusammen. • unterscheiden Black-Box und White-Box Testverfahren. • konzipieren und implementieren Testfälle. • überprüfen die Testüberdeckung anhand grundlegender Überdeckungskriterien (Anweisungs- bis Bedingungsüberdeckung). • geben die Grundlagen der statischen Programmanalyse wieder. • nennen die Funktionsweise von Hoare- WP-Kalkül. • verifizieren eine Ampelsteuerung mittels des FramaC Werkzeugs zur statischen Analyse von C Programmen. • beschreiben den Korrektheitsnachweis mittels abstrakter Interpretation und unterscheiden die konkrete von der abstrakten Programmsemnatik. • erläutern die Funktionsweise von Sammel- und Präfixsemantiken.

		• erstellen einen Korrektheitsbeweis für einen a-b-Filter mittels des Astrée Werkzeugs zur abstrakten Interpretation von C Programmen. • bewerten die Verlässlichkeit kommerzieller, sicherheitskritischer Systeme anhand von Fallstudien (Sizewell B, Airbus A320). • erschließen sich typische Probleme und Fehlerquellen bei der Programmierung von eingebetteten Systemen im Allgemeinen. • klassifizieren Fallstricke und Mehrdeutigkeiten in der Programmiersprache C99 im Besonderen. • können in Gruppen kooperativ und effektiv arbeiten. • können ihre Entwurfs- und Implementierungsentscheidungen kompakt präsentieren und argumentativ vertreten. • reflektieren ihre Entscheidungen kritisch und leiten Alternativen ab. • können offen und konstruktiv mit Schwachpunkten in der Konzeption wie Umsetzung umgehen.
7	**Voraussetzungen für die Teilnahme**	Für eine erfolgreiche Teilnahme an der Veranstaltung sind grundlegende Programmierkenntnisse in C/C++ erforderlich. Diese können durch den (empfohlenen) Besuch entsprechender Grundlagenveranstaltungen oder im Eigenstudium erworben sein. Weiterhin sind grundlegende Kenntnisse über Echtzeitsystemeeine, zum Beispiel durch den Besuch der Veranstaltung "Echtzeitsysteme", empfohlen. Eine formale Voraussetzung besteht in diesem Zusammenhang jedoch nicht.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	Unregelmäßig
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 90 h Eigenstudium: 135 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 179490	**Echtzeitsysteme mit erweiterten Übungen** (Real-time systems with extended exercises)	**7,5 ECTS**
2	Lehrveranstaltungen	Übung: Rechnerübungen zu Echtzeitsysteme (2 SWS) Vorlesung: Echtzeitsysteme (2 SWS) Übung: Erweiterte Übungen zu Echtzeitsysteme (2 SWS)	- - 5 ECTS
3	Lehrende	Peter Wägemann Simon Schuster	

4	**Modulverantwortliche/r**	Peter Wägemann
5	**Inhalt**	Videobearbeitung in Echtzeit, Echtzeitstrategiespiel, echtzeitfähig - der Begriff Echtzeit ist wohl einer der am meisten strapazierten Begriffe der Informatik und wird in den verschiedensten Zusammenhängen benutzt. Diese Vorlesung beschäftigt sich mit dem Begriff Echtzeit aus der Sicht von Betriebssystemen - was versteht man eigentlich unter dem Begriff Echtzeit im Betriebssystemumfeld, wo und warum setzt man sog. Echtzeitbetriebssysteme ein und was zeichnet solche Echtzeitbetriebssysteme aus? In dieser Vorlesung geht es darum, die oben genannten Fragen zu beantworten, indem die grundlegenden Techniken und Mechanismen vermittelt werden, die man im Betriebssystemumfeld verwendet, um Echtzeitsysteme und Echtzeitbetriebssysteme zu realisieren. Im Rahmen dieser Vorlesung werden unter anderem folgende Themen behandelt: • zeitgesteuerte und ereignisgesteuerte Systeme • statische und dynamische Ablaufplanungsverfahren • Fadensynchronisation in Echtzeitbetriebssystemen • Behandlung von periodischen und nicht-periodischen Ereignissen In den begleitenden Übungen werden die in der Vorlesung vorgestellten Techniken bei der Entwicklung eines kleinen Echtzeitsystems praktisch umgesetzt.
6	**Lernziele und Kompetenzen**	Studierende, die das Modul erfolgreich abgeschlossen haben: • unterscheiden die verschiedenen Komponenten eines Echtzeitsystems. • bewerten die Verbindlichkeiten von Terminvorgaben (weich, fest, hart). • erläutern die Zusammensetzung des Laufzeitverhaltes einer Echtzeitanwendung. • klassifizieren die Berührungspunkte zwischen physikalischem Objekt und kontrollierendem Echtzeitsystem. • interpretieren die Zeitparameter des durch das Echtzeitrechensystem zu kontrollierenden Objekts.

		• nennen die Zeitparameter des zugrundeliegenden Rechensystems (Unterbrechungslatenz, Ausführungszeit, ...). • unterscheiden synchrone und asynchrone Programmunterbrechung (insbesondere Trap/Interrupt, Ausnahmebehandlung und Zustandssicherung). • skizzieren die Verwaltungsgemeinkosten des schlimmsten Falls. • entwickeln in der Programmiersprache C und wenden die GNU Werkzeugkette für den ARM Cortex M4 an. • erstellen Echtzeitanwendungen auf Basis der eCos OS-Schnittstelle • ordnen die Strukturelemente von Echtzeitanwendungen zu: Aufgabe, Arbeitsauftrag und Faden. • erläutern die Implikationen von zeitlichem Mehrfachbetrieb auf die Verwaltungsgemeinkosten. • unterscheiden die Umsetzungsalternativen zur Ablaufsteuerung und die Trennung der Belange in Einplanung (Strategie) und Einlastung (Mechanismus). • benennen die grundsätzliche Verfahren der Ablaufsteuerung (taktgesteuert, reihum, vorranggesteuert). • erklären die grundlegenden Zeitparameter einer Aufgabe (Auslösezeitpunkt, Termin, Antwortzeit, Latenz, Ausführungszeit, Schlupfzeit). • unterscheiden die Grundlagen der Planbarkeit (gültig vs. zulässig, Optimalität von Einplanungsalgorithmen). • beschreiben den Unterschied zwischen konstruktiver und analytischer Einhaltung von Terminen-. • vergleich die Möglichkeiten (statisch, dynamisch) der zeitliche Analyse von Echtzeitanwendungen. • erklären die Grundlagen und Beschränkungen von dynamischer (worst-case?) und statischer WCET-Analyse (makroskopisch und mikroskopisch). • illustrieren Lösungsverfahren zur Bestimmung des längsten Ausführungspfads (Timing Schema, IPET). • erstellen Zeitmessung mittels Zeitgeber / Oszilloskop und bestimmen den längsten Pfad durch Code-Review. • erproben und konzipieren werkzeuggestützte WCET-Analyse mittels des absint aiT Analysewerkzeugs. • Entwickeln und annotieren Flußrestrikionen für die statische WCET-Analyse. • beschreiben die Grundlagen der Abfertigung periodischer Echtzeitsysteme (Periode, Phase, Hyperperiode). • skizzieren das periodische Modell und dessen Folgen (Entwicklungskomfort vs. Analysierbarkeit). • erklären die ereignisgesteuerte Ausführung (feste und dynamische Priorität, Verdrängbarkeit) mittels ereignisorienterter Planer (Berechnungskomplexität, MLQ-Scheduler, O(1)-Scheduler).

		• unterscheiden die zeitgesteuerte Ausführung (Busy Loop, Ablaufplan)und die Abfertigung von Arbeitsaufträgen im Abfrage- bzw. Unterbrecherbetrieb. • wenden die Grundlagen der ereignisgesteuerten Ablaufplanung periodischer Echtzeitsysteme an. • unterscheiden Verfahren zur statischen (RM, DM) und dynamischen Prioritätsvergabe (EDF, LRT, LST). • nennen den Unterschied zwischen Anwendungs- und Systemebene (Mehrdeutigkeit von Prioritäten). • erläutern den Optimalitätsnachweis des RM-, DM- und EDF-Algorithmus und dessen Ausnahmen. • beschreiben grundlegende Verfahren zur Planbarkeitsanalyse (CPU-Auslastung, Antwortzeitanalyse). • implementieren komplexe Aufgabensysteme in eCos. • unterscheiden die Grundlagen der zeitgesteuerten Ablaufplanung periodischer Echtzeitsysteme. • erstellen regelmäßige, zyklische Ablaufpläne (cyclic executive model, Rahmen). • vergleich Methoden der manuellen und algorithmischen Ablaufplanung. • unterscheiden optimale von heuristischen Verfahren (List Scheduling, Branch & Bound). • diskutieren die Konsequenzen eines Betriebswechsels in Echtzeitsystemen. • erstellen takt- beziehungsweise ereignisgesteuerte Abläufe in eCos beziehungsweise tt-eCos. • entwickeln ein softwarebasiertes Oszilloskop und erstellen dessen zeitliche Analyse und Ablaufplanung. • klassifizieren die Grundlagen der Abfertigung nicht-periodischer Echtzeitsysteme (minimale Zwischenankunftszeit). • definieren die Verbindlichkeiten von nicht-periodischen Aufgaben (aperiodisch, sporadisch) • zeigen die sich ergebenden Restriktionen des periodischen Modells (Mischbetrieb, Prioritätswarteschlangen, Übernahmeprüfung) auf. • beschreiben die Basistechniken des Laufzeitsystems (Zusteller, Unterbrecherbetrieb, Hintergrundbetrieb). • quantifizieren die Eigenschaften und Auswirkungen auf den periodischen Teil des Echtzeitsystems. • formulieren die Grundlagen des Slack-Stealing. • beschreiben den Einsatz von bandweite-bewahrenden Zustellern. • unterscheiden aufschiebbare Zusteller und Sporadic Server (SpSL und POSIX). • wenden eine Übernahmeprüfung bei sporadischen Aufgaben mittels dichte- oder schlupfbasierten Akzeptanztests an. • arbeiten einen strukturierter Ablaufplan (Rahmen) aus und untersuchen den Einsatz von Slack-Stealing.

		• ermitteln gerichtete Abhängigkeiten und Rangfolgen in Echtzeitanwendungen (Abhängigkeits- und Aufgabengraph). • stellen Umsetzungsalternativen für Abhängigkeiten einander gegenüber (naiv, implizit, explizit). • beschreiben das Konzept der zeitlichen Domänen und physikalischer bzw. logischer Ereignisse. • übertragen Abhängigkeiten auf das Problem der Ablaufplanung (modifiziere Auslösezeitpunkt/Termin, Phasenversatz). • konzipieren Rangfolge und aperiodische Steuerung in eCos. • implementieren einen aperiodischer Moduswechsel mit Zustandsüberführung in eCos. • gestalten einen Signal-Trigger für das entwickelte softwarebasierten Oszilloskops. • konzipieren explizite Synchronisation mittels Nachrichten in eCos. • wenden die Grundlagen von Wettstreit um Betriebsmitteln, Konkurrenz und Konfliktsituationen (kritische Abschnitte, (un)kontrollierte Prioritätsumkehr) an. • beschreiben echtzeitfähige Synchronisationsprotokolle (NPCS, PI, PCP). • nennen die Vor- und Nachteile der Techniken (transitive Blockung, Verklemmungen). • hinterfragen die Vereinfachung des PCP durch stapelbezogene Grenzprioritäten. • bestimmen die Ablaufplanung unter Berücksichtigung von Blockierungszeiten und Selbstsuspendierung. • implementieren Zugriffskontrolle (NPCS, PI, PCP) in Echtzeitanwendungen mit eCos. • analysieren Blockade für die Zugriffskontrolle in eCos. • erläutern die Anforderungen an verteile Echtzeitsysteme (Komposition, Erweiterbarkeit, Komplexität, Ereignis- vs. Zustandsnachricht). • fassen die Grundlagen von Knoten, Netzwerkschnittstellen und Netzübergängen sowie die Konzepte der expliziten und impliziten Flusskontrolle zusammen. • erschließen sich typische Probleme (zeitliche Analyse, Beobachtbarkeit, Synchronisation, Rangfolge) und Fehlerquellen bei der Programmierung von Echtzeitanwendungen. • können in Gruppen kooperativ und effektiv arbeiten. • können ihre Entwurfs- und Implementierungsentscheidungen kompakt präsentieren und argumentativ vertreten. • reflektieren ihre Entscheidungen kritisch und leiten Alternativen ab. • können offen und konstruktiv mit Schwachpunkten in der Konzeption wie Umsetzung umgehen.
7	**Voraussetzungen für die Teilnahme**	Für eine erfolgreiche Teilnahme an der Veranstaltung sind grundlegende Programmierkenntnisse in C/C++ erforderlich.

		Diese können durch den (empfohlenen) Besuch entsprechender Grundlagenveranstaltungen oder im Eigenstudium erworben sein, eine formale Voraussetzung besteht in diesem Zusammenhang nicht.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	Unregelmäßig
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 90 h Eigenstudium: 135 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• Hermann Kopetz. Real-Time Systems: Design Principles for Distributed Embedded Applications. Kluwer Academic Publishers, 1997. • Jane W. S. Liu. Real-Time Systems. Prentice-Hall, Inc., 2000. • Wolfgang Schröder-Preikschat. System-programmierung. Vorlesungsfolien. 2006.

1	**Modulbezeichnung** 707303	**Echtzeitsysteme (Vorlesung mit Übungen)** (Real-time systems)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Rechnerübungen zu Echtzeitsysteme (2 SWS) Vorlesung: Echtzeitsysteme (2 SWS) Übung: Übungen zu Echtzeitsysteme (2 SWS)	- - 2,5 ECTS
3	Lehrende	Peter Wägemann Simon Schuster Tim Rheinfels	

4	**Modulverantwortliche/r**	Peter Wägemann
5	**Inhalt**	Videobearbeitung in Echtzeit, Echtzeitstrategiespiel, echtzeitfähig - der Begriff Echtzeit ist wohl einer der am meisten strapazierten Begriffe der Informatik und wird in den verschiedensten Zusammenhängen benutzt. Diese Vorlesung beschäftigt sich mit dem Begriff Echtzeit aus der Sicht von Betriebssystemen - was versteht man eigentlich unter dem Begriff Echtzeit im Betriebssystemumfeld, wo und warum setzt man sog. Echtzeitbetriebssysteme ein und was zeichnet solche Echtzeitbetriebssysteme aus? In dieser Vorlesung geht es darum, die oben genannten Fragen zu beantworten, indem die grundlegenden Techniken und Mechanismen vermittelt werden, die man im Betriebssystemumfeld verwendet, um Echtzeitsysteme und Echtzeitbetriebssysteme zu realisieren. Im Rahmen dieser Vorlesung werden unter anderem folgende Themen behandelt: • zeitgesteuerte und ereignisgesteuerte Systeme • statische und dynamische Ablaufplanungsverfahren • Fadensynchronisation in Echtzeitbetriebssystemen • Behandlung von periodischen und nicht-periodischen Ereignissen In den begleitenden Übungen werden die in der Vorlesung vorgestellten Techniken bei der Entwicklung eines kleinen Echtzeitsystems praktisch umgesetzt.
6	**Lernziele und Kompetenzen**	Studierende, die das Modul erfolgreich abgeschlossen haben: • unterscheiden die verschiedenen Komponenten eines Echtzeitsystems. • bewerten die Verbindlichkeiten von Terminvorgaben (weich, fest, hart). • erläutern die Zusammensetzung des Laufzeitverhaltes einer Echtzeitanwendung. • klassifizieren die Berührungspunkte zwischen physikalischem Objekt und kontrollierendem Echtzeitsystem. • interpretieren die Zeitparameter des durch das Echtzeitrechensystem zu kontrollierenden Objekts.

		• nennen die Zeitparameter des zugrundeliegenden Rechensystems (Unterbrechungslatenz, Ausführungszeit, ...). • unterscheiden synchrone und asynchrone Programmunterbrechung (insbesondere Trap/Interrupt, Ausnahmebehandlung und Zustandssicherung). • skizzieren die Verwaltungsgemeinkosten des schlimmsten Falls. • entwickeln in der Programmiersprache C und wenden die GNU Werkzeugkette für den ARM Cortex M4 Microcontroller an. • erstellen Echtzeitanwendungen auf Basis der eCos OS-Schnittstelle • ordnen die Strukturelemente von Echtzeitanwendungen zu: Aufgabe, Arbeitsauftrag und Faden. • erläutern die Implikationen von zeitlichem Mehrfachbetrieb auf die Verwaltungsgemeinkosten. • unterscheiden die Umsetzungsalternativen zur Ablaufsteuerung und die Trennung der Belange in Einplanung (Strategie) und Einlastung (Mechanismus). • benennen die grundsätzliche Verfahren der Ablaufsteuerung (taktgesteuert, reihum, vorranggesteuert). • erklären die grundlegenden Zeitparameter einer Aufgabe (Auslösezeitpunkt, Termin, Antwortzeit, Latenz, Ausführungszeit, Schlupfzeit). • unterscheiden die Grundlagen der Planbarkeit (gültig vs. zulässig, Optimalität von Einplanungsalgorithmen). • beschreiben den Unterschied zwischen konstruktiver und analytischer Einhaltung von Terminen-. • vergleich die Möglichkeiten (statisch, dynamisch) der zeitliche Analyse von Echtzeitanwendungen. • erklären die Grundlagen und Beschränkungen von dynamischer (worst-case?) und statischer WCET-Analyse (makroskopisch und mikroskopisch). • illustrieren Lösungsverfahren zur Bestimmung des längsten Ausführungspfads (Timing Schema, IPET). • erstellen Zeitmessung mittels Zeitgeber / Oszilloskop und bestimmen den längsten Pfad durch Code-Review. • erproben werkzeuggestützte WCET-Analyse mittels des absint aiT Analysewerkzeugs. • beschreiben die Grundlagen der Abfertigung periodischer Echtzeitsysteme (Periode, Phase, Hyperperiode). • skizzieren das periodische Modell und dessen Folgen (Entwicklungskomfort vs. Analysierbarkeit). • erklären die ereignisgesteuerte Ausführung (feste und dynamische Priorität, Verdrängbarkeit) mittels ereignisorienterter Planer (Berechnungskomplexität, MLQ-Scheduler, O(1)-Scheduler). • unterscheiden die zeitgesteuerte Ausführung (Busy Loop, Ablaufplan)und die Abfertigung von Arbeitsaufträgen im Abfrage- bzw. Unterbrecherbetrieb.

		• wenden die Grundlagen der ereignisgesteuerten Ablaufplanung periodischer Echtzeitsysteme an. • unterscheiden Verfahren zur statischen (RM, DM) und dynamischen Prioritätsvergabe (EDF, LRT, LST). • nennen den Unterschied zwischen Anwendungs- und Systemebene (Mehrdeutigkeit von Prioritäten). • erläutern den Optimalitätsnachweis des RM-, DM- und EDF-Algorithmus und dessen Ausnahmen. • beschreiben grundlegende Verfahren zur Planbarkeitsanalyse (CPU-Auslastung, Antwortzeitanalyse). • implementieren komplexe Aufgabensysteme in eCos. • unterscheiden die Grundlagen der zeitgesteuerten Ablaufplanung periodischer Echtzeitsysteme. • erstellen regelmäßige, zyklische Ablaufpläne (cyclic executive model, Rahmen). • vergleich Methoden der manuellen und algorithmischen Ablaufplanung. • unterscheiden optimale von heuristischen Verfahren (List Scheduling, Branch & Bound). • diskutieren die Konsequenzen eines Betriebswechsels in Echtzeitsystemen. • erstellen takt- beziehungsweise ereignisgesteuerte Abläufe in eCos beziehungsweise tt-eCos. • klassifizieren die Grundlagen der Abfertigung nicht-periodischer Echtzeitsysteme (minimale Zwischenankunftszeit). • definieren die Verbindlichkeiten von nicht-periodischen Aufgaben (aperiodisch, sporadisch) • zeigen die sich ergebenden Restriktionen des periodischen Modells (Mischbetrieb, Prioritätswarteschlangen, Übernahmeprüfung) auf. • beschreiben die Basistechniken des Laufzeitsystems (Zusteller, Unterbrecherbetrieb, Hintergrundbetrieb). • quantifizieren die Eigenschaften und Auswirkungen auf den periodischen Teil des Echtzeitsystems. • formulieren die Grundlagen des Slack-Stealing. • beschreiben den Einsatz von bandweite-bewahrenden Zustellern. • unterscheiden aufschiebbare Zusteller und Sporadic Server (SpSL und POSIX). • wenden eine Übernahmeprüfung bei sporadischen Aufgaben mittels dichte- oder schlupfbasierten Akzeptanztests an. • arbeiten einen strukturierter Ablaufplan (Rahmen) aus und untersuchen den Einsatz von Slack-Stealing. • ermitteln gerichtete Abhängigkeiten und Rangfolgen in Echtzeitanwendungen (Abhängigkeits- und Aufgabengraph). • stellen Umsetzungsalternativen für Abhängigkeiten einander gegenüber (naiv, implizit, explizit). • beschreiben das Konzept der zeitlichen Domänen und physikalischer bzw. logischer Ereignisse.

		• übertragen Abhängigkeiten auf das Problem der Ablaufplanung (modifiziere Auslösezeitpunkt/Termin, Phasenversatz). • konzipieren Rangfolge und aperiodische Steuerung in eCos. • implementieren einen aperiodischer Moduswechsel mit Zustandsüberführung in eCos. • wenden die Grundlagen von Wettstreit um Betriebsmitteln, Konkurrenz und Konfliktsituationen (kritische Abschnitte, (un)kontrollierte Prioritätsumkehr) an. • beschreiben echtzeitfähige Synchronisationsprotokolle (NPCS, PI, PCP). • nennen die Vor- und Nachteile der Techniken (transitive Blockung, Verklemmungen). • hinterfragen die Vereinfachung des PCP durch stapelbezogene Grenzprioritäten. • bestimmen die Ablaufplanung unter Berücksichtigung von Blockierungszeiten und Selbstsuspendierung. • implementieren Zugriffskontrolle (NPCS, PI, PCP) in Echtzeitanwendungen mit eCos. • erläutern die Anforderungen an verteile Echtzeitsysteme (Komposition, Erweiterbarkeit, Komplexität, Ereignis- vs. Zustandsnachricht). • fassen die Grundlagen von Knoten, Netzwerkschnittstellen und Netzübergängen sowie die Konzepte der expliziten und impliziten Flusskontrolle zusammen. • erschließen sich typische Probleme (zeitliche Analyse, Beobachtbarkeit, Synchronisation, Rangfolge) und Fehlerquellen bei der Programmierung von Echtzeitanwendungen. • können in Gruppen kooperativ und effektiv arbeiten. • können ihre Entwurfs- und Implementierungsentscheidungen kompakt präsentieren und argumentativ vertreten. • reflektieren ihre Entscheidungen kritisch und leiten Alternativen ab. • können offen und konstruktiv mit Schwachpunkten in der Konzeption wie Umsetzung umgehen.
7	**Voraussetzungen für die Teilnahme**	Für eine erfolgreiche Teilnahme an der Veranstaltung sind grundlegende Programmierkenntnisse in C/C++ erforderlich. Diese können durch den (empfohlenen) Besuch entsprechender Grundlagenveranstaltungen oder im Eigenstudium erworben sein, eine formale Voraussetzung besteht in diesem Zusammenhang nicht.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio

11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	Unregelmäßig
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 90 h Eigenstudium: 60 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• Hermann Kopetz. Real-Time Systems: Design Principles for Distributed Embedded Applications. Kluwer Academic Publishers, 1997. • Jane W. S. Liu. Real-Time Systems. Prentice-Hall, Inc., 2000. • Wolfgang Schröder-Preikschat. Softwaresysteme 1. Vorlesungsfolien. 2006.

1	**Modulbezeichnung** 93015	**Einführung in die moderne Kryptographie** (Introduction to modern cryptography)	**7,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Einführung in die moderne Kryptographie (4 SWS) Übung: Einführung in die moderne Kryptographie (Übung) (2 SWS)	5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr. Dominique Schröder	

4	**Modulverantwortliche/r**	Carina Harrius Prof. Dr. Dominique Schröder
5	**Inhalt**	This course gives a comprehensive introduction to modern cryptography. The course also serves as a base for other courses on cryptography that are offered by the chair. The topics covered are the following: • Information theoretic security • Computational security • Private key Encryption • Message Authentication Codes • Hash functions • Public key Encryption • Digital Signatures More advanced topics may be covered if time permits.
6	**Lernziele und Kompetenzen**	On successfully passing the course, the student is guaranteed to be knowledgeable on the basic concepts of provable security.
7	**Voraussetzungen für die Teilnahme**	No previous knowledge in Cryptography or computer Security is required.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Variabel
11	**Berechnung der Modulnote**	Variabel (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 165 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Introduction to Modern Cryptography Jonathan Katz and Yehuda Lindell 2nd Edition (2014) (Chapman & Hall/CRC Cryptography and Network Security Series)

		ISBN-13: 978-1466570269

1	**Modulbezeichnung** 44240	**Grundlagen des Übersetzerbaus** (Foundations of compiler construction)	**7,5 ECTS**
2	Lehrveranstaltungen	Übung: Übungen zu Grundlagen des Übersetzerbaus (2 SWS) Vorlesung: Grundlagen des Übersetzerbaus (2 SWS)	- 7,5 ECTS
3	Lehrende	Patrick Kreutzer Michel Schmid Florian Mayer	

4	**Modulverantwortliche/r**	Prof. Dr. Michael Philippsen
5	**Inhalt**	***[Deutsch:]*** Auf den ersten Blick erscheint es wenig sinnvoll, sich mit Übersetzerbau zu beschäftigen. Andere Themen scheinen wesentlich näher an der direkten Anwendbarkeit in der industriellen Praxis. Der erste Blick täuscht: • Übersetzer gehören wohl zu den am gründlichsten studierten mittelgroßen sequentiellen Software-Systemen. Man kann viel aus den Erfahrungen lernen, die im Laufe der Jahre gesammelt wurden. • In den Übungen, die die Vorlesung begleiten, werden Sie selbst einen (kleinen) Übersetzer entwickeln. • Für viele Teilnehmer wird dieses Projekt das erste größere Software-Projekt sein. Viele der Algorithmen aus dem Grundstudium werden angewendet. • Bei jedem von Ihnen verwendeten Übersetzer gehen Sie in der Regel davon aus, dass richtiger Coder erzeugt wird. In der Vorlesung erfahren Sie, wie das geforderte hohe Maß an Korrektheit und Zuverlässigkeit erreicht wird. • Sie erlangen ein Verständnis für Konzepte von Programmiersprachen und verstehen, welcher Maschinen-Code aus Sprachkonstrukten gemacht wird. Mit diesem Wissen im Hinterkopf verbessern Sie Ihre Fähigkeit, gute und effiziente Programme zu schreiben. • Übersetzer werden nicht nur für Programmiersprachen benötigt. Spezielle Übersetzer braucht man in vielen Bereichen des täglichen Informatik-Lebens z.B. zur Textformatierung, für Programmtransformationen, für aspektorientiertes Programmieren, für die Verarbeitung von XML, ... • Es gehört zu einer Ingenieur-Ausbildung, in der Lage zu sein, diejenigen Werkzeuge selbst zu fertigen, die man verwendet. Für Informatiker gehört daher ein Verständnis vom Innenleben eines Übersetzers zum Rüstzeug. **Fokus der Lehrveranstaltung:**

Es werden Konzepte und Techniken der Übersetzerkonstruktion aus Sicht eines Übersetzerbauers und entlang der wesentlichen Arbeitsschritte eines Übersetzers (Frontend; Mittelschicht; Backend) vorgestellt. Übungen und Praxisaufgaben ergänzen die Vorlesung. Hier entwickeln die Studierenden auf der Basis eines vorgegebenen Programmrahmens einen eigenen Übersetzer für die Programmiersprache e2, die speziell für den Übersetzerbau-Vorlesungszyklus entworfen wurde.

Behandelte Themenfelder:

- Prinzipien der Übersetzung imperativer Programmiersprachen
- Struktur eines Übersetzers
- Symbolentschlüssler (Scanner) und Zerteiler (Parser)
- Abstrakter Syntaxbaum (AST)
- Besuchermuster
- AST-Transformationen, Entzuckerung
- Symboltabellen und Sichtbarkeitsbereiche
- Semantische Analyse: Namensanalyse, Typprüfung
- Übersetzung von arithmetischen Ausdrücken und Kontrollflusskonstrukten in registerbasierte oder stapelbasierte Zwischensprachen
- Übersetzung von Methoden und Methodenaufrufen; Methodenschachteln
- Übersetzung objektorientierter Sprachen mit Einfachvererbung, Schnittstellen und Mehrfachvererbung
- Methodenauswahl in Java (überladene und überschriebene Methoden)
- Code-Generierung nach Sethi-Ullmann, Graham-Glanville, per Baumtransformation sowie mit Hilfe dynamischer Programmierung
- Registerallokation mit lokalen Techniken und mit Graphfärbung
- Instruktionsanordnung mit "list scheduling"
- Debugger

Themen der Vorlesungseinheiten:

1. Einführung (Überblick, modulare Struktur von Übersetzern, Frontend, Mittelschicht, Backend), Bootstrapping)

2. Symbolentschlüssler (Lexer) und Zerteiler (Parser), (Token, Literale, Symboltabelle, Grammatikklassen (LK(k), LL(k), ...), konkreter Syntaxbaum, Shift-Reduce-Parser)

3. AST und semantische Analyse (abstrakter Syntaxbaum, Besuchermuster, Double Dispatch, Sichtbarkeitsbereiche, Definitionstabelle)

4. Typkonsistenz (Typsicherheit, Typsystem, Typüberprüfung, Typberechnung, Typkonvertierung, attributierte Grammatiken)

5. AST-Transformationen (Transformationsschablonen für Ausdrücke, Transformation innerer und generischer Klassen)

6. Transformation in Zwischensprache (registerbasiert versus stapelbasiert, Übersetzung von arithmetischen Ausdrücken, Zuweisungen, mehrdimensionalen Feldern, struct-Datentypen und Kontrollflussstrukturen (einschließlich Kurzschlussauswertung))

7. Methodenschachteln und Kellerrahmen (relative Adressen, call by value/reference/name, geschachtelte Funktionen, Funktionszeiger,

Stack- und Framepointer, Funktionsaufruf, Prolog, Epilog)

8. Objektorientierte Sprachen I: Einfachvererbung (Symbol- und Typanalyse, Methodenauswahl mit Überschreiben und Überladen, virtuelle Methodenaufrufe, Klassendeskriptoren, dynamische Typprüfung und -wandlung)

9. Objektorientierte Sprachen II: Schnittstellen und Mehrfachvererbung (Interface v-Tables, dynamische Typprüfung und -wandlung mit Interfaces, Interfaces mit Default-Implementierung, Diamantenproblem)
10. Einfache Code-Erzeugung (Code-Selektion nach Sethi-Ullman, Register-Allokation, Instruktionsreihenfolge, optimale Code-Erzeugung für Ausdrucksbäume)

11. Fortgeschrittene Code-Erzeugung (Baumtransformation, Graham-Glanville, dynamisches Programmieren)

12. Registerallokation (Leistungsabschätzung, Lebendigkeitsintervalle, Kollisions- und Interferenzgraph, Spilling, Färbungsheuristiken, Aufteilung von Lebendigkeitsintervallen, 2nd Chance Bin Packing, Registerverschmelzung)

13. Parallelismus auf Instruktionsebene, Instruktionsreihenfolge, Debugger (Konflikte im Instruktionsfließband, List Scheduling, Delay-Slots, Sprungzielvorhersage, ptrace, Unterbrechungs- und Beobachtungspunkte, DWARF)

Meilensteine der Übungsbetriebs:

Im Rahmen der Übungen (siehe entsprechende Lehrveranstaltung) werden die in der Vorlesung vorgestellten Konzepte und Techniken zur Implementierung eines Übersetzers in die Praxis umgesetzt. Ziel der Übungen ist es, bis zum Ende des Semesters einen funktionsfähigen Übersetzer für die Beispiel-Programmiersprache e2 zu implementieren. Ein Rahmenprogramm ist gegeben, das in fünf Meilensteinen um selbstentwickelte Schlüsselkomponenten zu erweitern ist.

Folgende Meilensteine sind zu erreichen:

Meilenstein 1: Grammatik, AST-Konstruktion: Antlr-Produktionen, AST-Besucherschnittschelle, generischer AST-Besucher für return und Schleifen, AST-Besucher zur Visualisierung.

Meilenstein 2: Symbolanalyse, Symboltabelle, Standardfunktionen, AST-Besucher für die Symbolanalyse.

Meilenstein 3: Konstantenfaltung per AST-Transformation, Typanalyse mit bottom-up AST-Besuch, der implizite Typwandlungen bei Bedarf ergänzt.

Meilenstein 4: AST-Besucher zur Erzeugung der Zwischensprachrepräsentation, Übersetzung von arithmetischen Ausdrücken, return, Zuweisungen, logischen Ausdrücken, Bedingungen und Schleifen.

Meilenstein 5.0: Speicherzuteilung: Festlegung und Umsetzung der ABI Aufrufkonvention, Zuweisung von Speicheradressen zu Variablen; Kellerrahmenallokation; caller-save und callee-save Register.

Meilenstein 5.1: Code-Erzeugung: Implementierung der e2 Standardbibliothek; IR-Besucher zur Erzeugung von Assembly-Code.

Für die Meilensteine 1-3 soll der Übersetzer sowohl Integer- als auch Gleitkomma-Arithmetik unterstützen. Für die nachfolgenden Meilensteine reicht Integer-Arithmetik.

[English:]

The lecture teaches concepts and techniques of compiler construction from a compiler developer view, following the structure of the compiler frontend, middle end, and backend. Exercise sessions and practical assignments complement the lecture; the students implement their own compiler (based on a framework) for the e2 programming language, which is designed for this series of compiler construction lectures.

Content Summary

- Principles of compiling imperative programming languages
- Structure of a compiler
- Scanner and parser
- Abstract syntax trees (ASTs)
- Visitor design pattern
- AST transformations, desugaring
- Symbol tables and scopes
- Semantic analysis: name analysis, type checking
- Compilation of arithmetic expressions and control flow structures to register-based and stack-based intermediate languages
- Compilation of functions and function calls, activation records
- Compilation of object-oriented languages with single inheritance, interfaces, and multiple inheritance
- Method resolution in Java (overloaded and overridden methods)
- Code generation with Sethi-Ullmann algorithm, Graham-Glanville algorithm, tree transformations, and dynamic programming
- Register allocation with local techniques and graph coloring
- Instruction scheduling with the list scheduling technique
- Debuggers

Lecture Topics

- 1. Introduction: Class overview, modular structure of compilers (front-, middle-, and backend), compilation bootstrapping
- 2. Lexer and Parser: Tokens, literals, symbol table, grammar classes (LR(k), LL(k), ...), concrete syntax tree, shift-reduce parser
- 3. ASTs and semantic analysis: Abstract syntax tree, visitor pattern, double dispatch, scopes, definition table
- 4. Type consistency: Type safety, type system, type checks, type inference, type conversions, attributed grammars
- 5. AST transformations: Transformation patterns (arithmetics), transformation of nested and generic classes
- 6. Intermediate representations: Types of IRs, arithmetic operations, assignments, multidimensional array access, structs, control flow instructions, short-circuit evaluation
- 7. Activation record and stack frame: Relative addresses, call by value/reference/name, nested functions, function pointers, stack pointer and frame pointer, function calls: prolog and epilog
- 8. Object-oriented languages: single inheritance: Symbol and type analysis, method selection with method overloading and overriding, virtual method calls, class descriptors, dynamic type checks and casts

- 9. Object-oriented languages II: interfaces, multiple inheritance: Interface v-tables, dynamic type checks and casts with interfaces, interfaces with default implementations and state, diamond problem, virtual inheritance
- 10. Basic code generation: Code selection, register allocation, instruction order, basic blocks, optimal code generation for expression trees
- 11. Optimized code selection: Code selection as tree transformation, Graham-Glanville code generators, dynamic programming
- 12. Optimized register allocation: Performance approximations, liveness analysis, collision and interference graph, register spilling, coloring heuristics, optimistic extension, live range splitting, register coalescing, data structures
- 13. Instruction level parallelism, instruction order, debugger: Data, structural, and control conflicts in CPU pipelines, list scheduling, delay slots, branch predictions, superscalar and VLIW architectures, ptrace, break- and watch-points, DWARF

Assignment Milestones

For the assignments of this course, the students put the concepts and techniques presented in the lecture for implementing a compiler into practice. The goal of the assignments is to implement a functional compiler for the e2 programming language by the end of the semester. The e2 language is specifically designed for educational purposes; the students obtain a description of the language.

A framework for the implementation is provided to the students. The students implement the core components of the compiler in five milestones.

All milestones need to be fulfilled to pass the module; the last milestone contains two tasks. In particular, the milestones are:

- Milestone 1: Grammar definition and construction of the AST: ANTLR productions, AST visitor interface, and generic AST visitor for array accesses and return and loop statements; AST visitor for AST visualization.
- Milestone 2: Name analysis: symbol table; declaring standard functions; AST visitor for name analysis.
- Milestone 3: Constant folding and type analysis: AST transformations for constant folding; AST visitor for bottom-up type analysis, adding AST nodes for implicit casts;
- Milestone 4: AST translation to intermediate representation: AST visitor to generate IR; translation of arithmetic, return, and assign statements, logical expressions, conditions, loops.
- Milestone 5.0: Memory assignment: definition and implementation of the ABI calling convention; memory

		assignment of variables; stack frame allocation; caller-save and callee-save registers. • Milestone 5.1: Code generation: implementation of the e2 standard library; IR visitor to generate assembly code. For milestones one through three, the compiler needs to support both integer and floating-point arithmetic. For the last two milestones, only integer arithmetic is required.
6	**Lernziele und Kompetenzen**	***[Deutsch:]*** Die Studierenden • nennen die typischen Aufgaben und Datenstrukturen eines Übersetzers • erläutern das Konzept des Bootstrapping • beschreiben Struktur und Arbeitsweise eines Abtasters (Scanner) und zeigen Grenzen und Problemfälle auf • wenden Grammatiken zur Konstruktion von Zerteilern (Parser) an • kennen die Komplexität eines Zerteilers für Java • beschreiben die wichtigsten Aufgaben der semantischen Analyse und wenden diese am Beispiel verschiedener Programmiersprachen (insbesondere Java) an • skizzieren typische AST-Transformationen am Beispiel des Java-Übersetzers • veranschaulichen die Grundzüge der Java-Kellermaschine und die zugehörige Transformation von Quell- zu Byte-Code • analysieren die Unterschiede zwischen Programmiersprachen hinsichtlich Felder und Verbund-Strukturen • erläutern die Verwendung von Stapel- und Kellerspeicher bei der Programmausführung • kennen verschiedene Maschineninstruktionssätze • optimieren die Registerverwendung vor der Generierung von Maschinencode • wenden das Verfahren von Graham & Glanville zur Erzeugung von Maschinencode an • erkennen Grenzen der Optimierung bei der Code-Generierung und analysieren alternative Strategien • beschreiben den Unterschied zwischen statischer und dynamischer Ablaufplanung • untersuchen Besonderheiten des Übersetzerbaus für objekt-orientierte Sprachen • ergänzen einen vorgegebenen Abtaster und abstrakten Syntaxbaum, um alle Sprachkonstrukte einer Beispielsprache zu unterstützen • implementieren Konstantenfaltung, den Aufbau der Symboltabelle und Typprüfung auf dem abstrakten Syntaxbaum • erzeugen Zwischencode aus dem abstrakten Syntaxbaum • bilden Kontrollstrukturen auf Sprünge ab

- veranschaulichen die Adressierung von (mehrdimensionalen) Feldern
- entwickeln Konventionen für Funktionsaufrufe und den Aufbau des Stacks
- berechnen Offsets fuer Variablen auf dem Stack.
- implementieren eine einfache Registervergabe.
- kennen Details verschiedener Prozessorarchitekturen
- generieren Maschinencode für mindestens eine Prozessorarchitektur
- implementieren eine Laufzeitbibliothek
- wenden Debugging für maschinennahen Code an

[English:]

Students who have successfully completed the module will have the ability to

- identify the components and data structures of a compiler
- explain the concept of bootstrapping
- describe the structure and operation of a lexer and show limitations and problem cases
- use grammars for the construction of parsers
- know the complexity of Java parsers
- describe the main tasks of semantic analysis and apply them to different programming languages (especially Java)
- outline typical AST transformations using the Java compiler as an example
- illustrate the basic features of the Java Virtual Machine (JVM) and the corresponding transformation from source to byte code
- analyze the differences between programming languages in terms of arrays and compound structures
- explain the use of stack memory in program execution
- know different machine instruction sets
- optimize register allocation before generating machine code
- apply the Graham-Glanville algorithm to generate machine code
- recognize limitations of optimization in code generation and to analyze alternative strategies
- describe the difference between static and dynamic scheduling
- examine features of compiler construction for object-oriented languages
- augment a given lexer and abstract syntax tree to support all language constructs in an example language
- implement constant folding, symbol table construction, and type checking on the abstract syntax tree
- generate intermediate code from the abstract syntax tree
- map control structures to jumps
- translate compound boolean expressions with shortcut evaluation

		• illustrate addressing of (multidimensional) arrays • design conventions for function calls and stack frame layout • calculate offsets for stack variables • implement a basic register allocation. • know details of different processor architectures • generate machine code for at least one processor architecture • implement a runtime library • apply debugging to machine code
7	**Voraussetzungen für die Teilnahme**	Participants of this lecture are expected to have profound skills in the following programming languages: • Java (assignments are implemented in Java) • Assembler
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Variabel
11	**Berechnung der Modulnote**	Variabel (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 50 h Eigenstudium: 175 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• Modern Compiler Implementation in Java", A.W. Appel, Cambridge University Press, 1998 • Compilers - Principles, Techniques and Tools", A. Aho, R. Sethi, J. Ullmann, Addison-Wesley, 2006 • Modern Compiler Design", D. Grune, H. Bal, C. Jacobs, Langendoen, Wiley, 2002

1	**Modulbezeichnung** 43950	**Kommunikationssysteme** (Communication systems)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Kommunikationssysteme (2 SWS) Übung: Übungen zu Kommunikationssysteme (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr. Reinhard German Alexander Brummer	

4	**Modulverantwortliche/r**	Prof. Dr. Reinhard German
5	**Inhalt**	Aus Rechnerkommunikation ist der grundlegende Aufbau von IP-basierten Netzen bekannt, Inhalt von Kommunikationssysteme sind weitere Netztechnologien wie Leitungsvermittlung (Telefonnetze, Sonet/ SDH/WDM) und Netze mit virtueller Leitungsvermittlung (ATM, MPLS) sowie Netzwerkvirtualisierung (SDN, NFV), Multimediakommunikation über paketvermittelte Netze (Streaming, RTP, SIP, Multicast), Dienstgüte in paketvermittelten Netzen (Integrated Services, RSVP, Differentiated Services, Active Queue Management, Policing, Scheduling), drahtlose und mobile Kommunikation (GSM, UMTS, LTE, 5G, Wimax, WLAN, Bluetooth, ZigBee u.a. Sensornetze). Auch Kommunikation in der Industrie wird behandelt. In der Übung werden praktische Aufgaben im Labor durchgeführt: ein Labor enthält mehrere IP-Router, Switches und Rechner, IP-Telefone und Telefonie-Software für VoIP, es werden verschiedene Konfigurationen eingestellt und getestet. Ein weiterer Übungsteil beschäftigt sich mit Mobilkommunikation. *Contents:* Based on the course computer communications the architecture of IP networks is known. Contents of this course will be additional networking technologies such as circuit switching (telephony, SONET/ SDH/WDM) and networks with virtual circuit switching (ATM, MPLS) as well as network virtualization (SDN, NFV), multimedia communications over packet switched networks (streaming, RTP, SIP, multicast), quality-of-service in packet switched networks (integrated services, RSVP, differentiated services, active queue management, policing, scheduling), wireless and mobile communications (GSM, UMTS, LTE, 5G, Wimax, WLAN, Bluetooth, sensor networks such as ZigBee). Industrial communication will also be a topic. In the tutorial practical tasks are performed in the laboratory: One laboratory contains several IP routers, switches and computers, IP phones and VoIP telephone software. Various configurations are set up and tested. Another part of the tutorial deals with mobile communications.
6	**Lernziele und Kompetenzen**	Die Studierenden erlangen Kenntnisse über Technologien bei der Leitungs- und Paketvermittlung in leitungsgebundenen und drahtlosen/mobilen Netzen Kenntnisse über die Grundlagen von Dienstgütemechanismen in paketvermittelten Netzen

		praktische Erfahrung in der Konfiguration eines IP-Switch-Router-Netzes mit Multimediaverkehr Students obtain the following learning targets and competences Knowledge of technologies in circuit and packet switching in wired and wireless/mobile networks Knowledge of the foundations of quality of service mechanisms in packet switched networks Practical experience in configuring an IP switch router network with multimedia traffic
7	**Voraussetzungen für die Teilnahme**	Basic knowledge of working with the Linux command line interface (terminal).
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Kurose, Ross. Computer Networking: A Top-Down Approach. 7th Ed., Pearson Education, 2017 W. Stallings. Data and Computer Communications, 10th ed., Pearson Education, 2014 W. Stallings. Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud, Pear-son Education, 2016 Cox. An Introduction to LTE. Wiley, 2012

1	**Modulbezeichnung** 93130	**Konzeptionelle Modellierung** (Conceptual modelling)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übungen zu Konzeptionelle Modellierung (2 SWS) Vorlesung: Konzeptionelle Modellierung (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	David Haller Prof. Dr.-Ing. Richard Lenz	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Richard Lenz
5	**Inhalt**	• Grundlagen der Modellierung • Datenmodellierung am Beispiel Entity-Relationship-Modell • Modellierung objektorientierter Systeme am Beispiel UML • Relationale Datenmodellierung und Anfragemöglichkeiten • Grundlagen der Metamodellierung • XML • Multidimensionale Datenmodellierung • Domänenmodellierung und Ontologien
6	**Lernziele und Kompetenzen**	Die Studierenden: • definieren grundlegende Begriffe aus der Datenbankfachliteratur • erklären die Vorteile von Datenbanksystemen • erklären die verschiedenen Phasen des Datenbankentwurfs • benutzen das Entity-Relationship Modell und das erweiterte Entity-Relationship Modell zur semantischen Datenmodellierung • unterscheiden verschiedene Notationen für ER-Diagramme • erläutern die grundlegenden Konzepte des relationalen Datenmodells • bilden ein gegebenes EER-Diagramm auf ein relationales Datenbankschema ab • erklären die Normalformen 1NF, 2NF, 3NF, BCNF und 4NF • definieren die Operationen der Relationenalgebra • erstellen Datenbanktabellen mit Hilfe von SQL • lösen Aufgaben zur Datenselektion und Datenmanipulation mit Hilfe von SQL • erklären die grundlegenden Konzepte der XML • erstellen DTDs für XML-Dokumente • benutzen XPATH zur Formulierung von Anfragen an XML-Dokumente • definieren die grundlegenden Strukturelemente und Operatoren des multidimensionalen Datenmodells • erklären Star- und Snowflake-Schema • benutzen einfache UML Use-Case Diagramme • benutzen einfache UML-Aktivitätsdiagramme • erstellen UML-Sequenzdiagramme • erstellen einfache UML-Klassendiagramme • erklären den Begriff Meta-Modellierung • definieren den Begriff der Ontologie in der Informatik • definieren die Begriffe RDF und OWL

7	**Voraussetzungen für die Teilnahme**	Gewünscht "Algorithmen und Datenstrukturen" und "Grundlagen der Logik und Logikprogrammierung"
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Masterprüfung Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222 Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• Elmasri, Ramez, and Sham Navathe. Grundlagen von Datenbanksystemen. Pearson Deutschland GmbH, 2009. - ISBN-10: 9783868940121 • Alfons Kemper, Andre Eickler: Datenbanksysteme : Eine Einführung. 6., aktualis. u. erw. Aufl. Oldenbourg, März 2006. - ISBN-10: 3486576909 • Bernd Oestereich: Analyse und Design mit UML 2.1. 8. Aufl. Oldenbourg, Januar 2006. - ISBN-10: 3486579266 • Ian Sommerville: Software Engineering. 8., aktualis. Aufl. Pearson Studium, Mai 2007. - ISBN-10: 3827372577 • Horst A. Neumann: Objektorientierte Softwareentwicklung mit der Unified Modeling Language. (UML). Hanser Fachbuch, März 2002. - ISBN-10: 3446188797 • Rainer Eckstein, Silke Eckstein: XML und Datenmodellierung. Dpunkt Verlag, November 2003. - ISBN-10: 3898642224

1	**Modulbezeichnung** 557235	**Middleware - Cloud Computing (Vorlesung mit Übungen)** (Lecture and tutorial: Middleware - Cloud computing)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Middleware - Cloud Computing (2 SWS) Übung: Middleware - Cloud Computing - Übungen (2 SWS) Übung: Middleware - Cloud Computing - Rechnerübungen (2 SWS)	- - -
3	Lehrende	PD Dr.Ing. Tobias Distler Laura Lawniczak	

4	**Modulverantwortliche/r**	PD Dr.Ing. Tobias Distler
5	**Inhalt**	• Überblick Cloud Computing • Grundlagen verteilter Programmierung (Web Services/SOAP/ REST) • Virtualisierung als Basis für Cloud Computing • Infrastructure as a Service (IaaS) am Beispiel von Eucalyptus und Amazon EC2 • Skalierbare Verarbeitung von großen Datenmengen • Interoperabilität und Multi-Cloud Computing • Fehlertoleranz im Kontext von Cloud Computing • Aktuelle Forschungstrends
6	**Lernziele und Kompetenzen**	Studierende, die das Modul erfolgreich abgeschlossen haben: - nennen unterschiedliche Ausprägungen von Cloud-Computing. - erläutern verschiedene Cloud-Architekturen. - stellen Vor- und Nachteile von Cloud-Computing gegenüber. - unterscheiden die Herangehensweisen bei der Entwicklung von SOAP- im Vergleich zu REST-Anwendungen. - organisieren den Austausch von Informationen in einer verteilten Anwendung unter Verwendung eines Verzeichnisdienstes. - entwickeln eigene auf Web-Services basierende Anwendungen. - erläutern die Anforderungen an ein virtualisiertes System. - beschreiben die für die Virtualisierung eines Systems erforderlichen Kriterien. - vergleichen zwischen unterschiedlichen Virtualisierungstechniken und -ebenen. - schildern den Aufbau und die Funktionsweise von Xen und Linux-VServer.

- erproben das Einrichten eines Abbilds für eine virtuelle Maschine.

- skizzieren die Architektur einer Infrastruktur-Cloud sowie die Aufgabenbereiche hierfür zentraler Komponenten am Beispiel von Eucalyptus.

- erproben das Bereitstellen von Anwendungen in einer Infrastruktur-Cloud.

- zeigen die Grundlagen Software-definierter Netzwerke am Beispiel von Onix und B4 auf.

- bewerten verschiedene im Bereich Cloud-Computing zum Einsatz kommende Datenspeichersysteme (Google File System, Bigtable, Windows Azure Storage, Amazon Dynamo) hinsichtlich der Kriterien Verfügbarkeit, Konsistenz und Partitionstoleranz.

- erläutern eine auf Vektoruhren basierende Methode zur Auflösung im Zusammenhang mit letztendlicher Konsistenz auftretender Konflikte.

- entwickeln ein verteiltes Dateisystem nach dem Vorbild von HDFS, das auf die Speicherung großer Datenmengen ausgelegt ist.

- erkunden das Bereitstellen selbst entwickelter Dienste mittels Docker.

- erstellen ein Framework zur parallelen Bearbeitung von Daten nach dem Vorbild von MapReduce.

- konzipieren eigene MapReduce-Anwendungen zur Verarbeitung strukturierter Rohdaten.

- diskutieren die Fehlertoleranzmechanismen in Google MapReduce.

- schildern die grundsätzliche Funktionsweise von Systemen zur Kühlung von Datenzentren mittels Umgebungsluft.

- beschreiben das Grundkonzept einer temperaturabhängigen Lastverteilung von Prozessen in einem Datenzentrum.

- stellen diverse Ansätze zur Erhöhung der Energieeffizienz von MapReduce-Clustern gegenüber.

- unterscheiden die Architekturen und Funktionsweisen der Koordinierungsdienste Chubby und ZooKeeper.

- entwickeln einen eigenen Koordinierungsdienst nach dem Vorbild von ZooKeeper.

		- ermitteln die Konsistenzeigenschaften der eigenen Koordinierungsdienstimplementierung. - erläutern unterschiedliche Ansätze zur Reduzierung bzw. Tolerierung von Tail-Latenz. - skizzieren das Grundkonzept von Erasure-Codes. - beschreiben den Aufbau eines auf die Clouds mehrerer Anbieter gestützten Datenspeichersystems. - erläutern den Einsatz passiver Replikation zur Bereitstellung von Fehlertoleranzmechanismen für virtuelle Maschinen am Beispiel von Remus. - schildern die Grundlagen der Migration von virtuellen Maschinen. - bewerten die Qualität einer aktuellen Publikation aus der Fachliteratur. - erschließen sich typische Probleme (Nebenläufigkeit, Konsistenz, Skalierbarkeit) und Fehlerquellen bei der Programmierung verteilter Anwendungen. - können in Kleingruppen kooperativ arbeiten. - können ihre Entwurfs- und Implementierungsentscheidungen kompakt präsentieren und argumentativ vertreten. - können offen und konstruktiv mit Schwachpunkten und Irrwegen umgehen. - reflektieren ihre Entscheidungen kritisch und leiten Alternativen ab.
7	**Voraussetzungen für die Teilnahme**	Gute Programmierkenntnisse in Java
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio Erfolgreiche Bearbeitung aller während des Semesters gestellten Übungsaufgaben (6 Aufgaben, Bewertung jeweils mit "ausreichend") + 30-minütige mündliche Prüfung am Ende des Semesters. Die Modulnote ergibt sich zu 100% aus der Bewertung der mündlichen Prüfung.
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	nur im Wintersemester

13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 740665	**Parallele Systeme (Vorlesung mit erweiterten Übungen)** (Parallel systems with extended exercises)	**7,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Parallele Systeme (2 SWS) Übung: Übung zu Parallele Systeme (2 SWS) Übung: Erweiterte Übungen zu Parallele Systeme (2 SWS)	2,5 ECTS 2,5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr.-Ing. Jürgen Teich PD Dr.Ing. Frank Hannig Michael Witterauf Stefan Groth Marcel Brand	

4	**Modulverantwortliche/r**	Joachim Falk PD Dr.Ing. Frank Hannig Prof. Dr.-Ing. Jürgen Teich
5	**Inhalt**	Aktuelle PCs verfügen über Mehrkernprozessoren und Grafikkarten, die wiederum aus hunderten von einfachen Prozessoren bestehen können. Hierdurch wird ein hohes Maß an nebenläufiger Datenverarbeitung möglich, welche bis vor einigen Jahren nur in Großrechnern erreicht werden konnte. Die effiziente Ausnutzung dieser Parallelität bedarf allerdings mehr als nur mehrerer Prozessoren, insbesondere muss das zu lösende Problem Parallelverarbeitung erlauben. In dieser Vorlesung werden Eigenschaften unterschiedlicher paralleler Rechnerarchitekturen und Metriken zu deren Beurteilung behandelt. Weiterhin werden Modelle und Sprachen zum Programmieren paralleler Rechner eingeführt. Neben der Programmierung von allgemeinen Parallelrechnern werden Entwurfsmethoden (CAD) vorgestellt, wie man ausgehend von einer algorithmischen Problemstellung ein massiv paralleles Rechenfeld in VLSI herleiten kann, das genau dieses Problem optimal parallel berechnet. Solche Schaltungen spielen auf der Bit- bzw. Wortebene eine dominante Rolle (Arithmetik) sowie bei Problemen der Signal- und Bildverarbeitung (z.B. Filter). Im Einzelnen werden behandelt: 1) Theorie der Parallelität (parallele Computermodelle, parallele Spezifikationsformen und -sprachen, Performanzmodelle und -berechnung) 2) Klassifikation paralleler und skalierbarer Rechnerarchitekturen (Multiprozessoren und Multicomputer, Vektorrechner, Datenflussmaschinen, VLSI-Rechenfelder) 3) Programmierbare System-on-Chip (SoC) und Mehrkern-Architekturen (Grafik-Prozessoren, Cell, etc.) 4) Programmierung paralleler Rechner (Sprachen und Modelle, Entwurfsmethoden und Compiler, Optimierung) 5) Massive Parallelität: Vom Algorithmus zur Schaltung 6) Praktische Übungen mit rechnergestützten Werkzeugen

		Today's PCs consist of multi-core processors and graphics cards that again comprise hundreds to thousands of simple processors. As a result of this, a very high degree of parallel data processing becomes possible, which was subjected to supercomputers a couple of years ago. The efficient exploitation of parallel processing requires not only multiple processors but also parallelism inherent in the problem to be solved. In this lecture, properties of different parallel computer architectures and corresponding quality metrics are examined. Further, models and parallel programming languages are introduced. In addition to programming general parallel computers, design methods (CAD) are presented that systematically transform an algorithmic problem description into a massive parallel processor array (VLSI), which can optimally execute the given problem in parallel. Such highly parallel circuits play an essential role at the bit level and circuit level (arithmetics) as well as in the case of signal processing and image processing (e.g., filter). *In detail, the following topics are covered:* 1) Theory of parallelism (parallel models of computation, parallel specification and parallel languages, performance models) 2) Classification of parallel and scalable computer architectures (multi-processors and multi-computers, vector computers, data-flow machines, VLSI processor arrays) 3) Programmable System-on-Chip (SoC) and multi-core architetcures (graphics processors, Cell, etc.) 4) Programming of parallel computers (languages and models, design methods and compiler, optimization) 5) Massive parallelism: From algorithm to circuit 6) Practical training with computer-aided design tools
6	**Lernziele und Kompetenzen**	Schwerpunkt der Vorlesung ist die Vermittlung von Grundlagen der parallelen Datenverarbeitung. *The focus of this lecture are foundations of parallel data processing.* Fachkompetenz - Verstehen • Die Studierenden verstehen grundlegende Konzepte der parallelen Datenverarbeitung, sowohl theoretischer Art anhand von Modellen, als auch an Architekturbeispielen. The students become familiar with the fundamentals of parallel data processing, theoretic in the form of models as well as by architecture examples. Fachkompetenz - Anwenden • Die Studierenden setzen sich mit modernen eingebetteten parallelen Ein-Chip-Architekturen auseinander. The students

		get familiar with modern embedded parallel system-on-chip architectures. • Die Studierenden wenden grundlegende Performanzmodelle und Parallelisierungtechniken zur Analyse und Optimierung von parallelen Algorithmen und Architekturen an. The students exercise basic performance models and parallelization techniques for the analysis and optimization of parallel algorithms and architectures. • Die Studierenden setzen die Modellierung und den Entwurf von massiv-parallelen Prozessorfeldern in konkreten Aufgaben selbstständig um. In concrete tasks, the students apply independently the modeling and the design of massively parallel processors arrays. • Die Studierenden wenden das erlernte Wissen in den erweiterten Übungen vor Ort an den Rechnerarbeitsplätzen des Lehrstuhls an. The students apply their learned knowledge in hands-on computer exercises on-site at the chair's computer workstations.
7	**Voraussetzungen für die Teilnahme**	Die Auswahl dieses Moduls schließt die Auswahl des Moduls „Parallele Systeme“ aus.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio • Prüfung und erfolgreiche Bearbeitung aller Übungsaufgaben in den erweiterten Übungen (verpflichtend, vor Ort an den Rechnerarbeitsplätzen des Lehrstuhls). • Ein Wechsel der Prüfungsform von einer Klausur zu einer mündlichen Prüfung ist in Ausnahmefällen (siehe § 16 ABMPO/TechFak) auch nach Semesterbeginn noch möglich. In diesem Fall werden die Studierenden spätestens zwei Wochen nach Vorlesungsbeginn informiert. • Prüfungssprache: Deutsch oder Englisch. Die Unterrichts- und Prüfungssprache hängt von den Sprachkenntnissen und Präferenzen der Teilnehmerinnen und Teilnehmer ab und wird dementsprechend innerhalb der ersten zwei Wochen nach Vorlesungsbeginn festgelegt.
11	**Berechnung der Modulnote**	Portfolio (100%) Die Modulnote ergibt sich aus der Prüfungsleistung.
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 90 h Eigenstudium: 135 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch oder Englisch
16	**Literaturhinweise**	Weitere Informationen:

		https://www.cs12.tf.fau.de/lehre/lehrveranstaltungen/vorlesungen/parallele-systeme

1	**Modulbezeichnung** 399289	**Programmierung und Architekturen von Cluster-Rechnern** (Programming and architecture of computer clusters)	**7,5 ECTS**
2	Lehrveranstaltungen	Zu diesem Modul sind keine Lehrveranstaltungen oder Lehrveranstaltungsgruppen hinterlegt!	
3	Lehrende	Zu diesem Modul sind keine Lehrveranstaltungen und somit auch keine Lehrenden hinterlegt!	

4	**Modulverantwortliche/r**	
5	**Inhalt**	keine Inhaltsbeschreibung hinterlegt!
6	**Lernziele und Kompetenzen**	keine Beschreibung der Lernziele und Kompetenzen hinterlegt!
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	keine Angaben zum Turnus des Angebots hinterlegt!
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: ?? h (keine Angaben zum Arbeitsaufwand in Präsenzzeit hinterlegt) Eigenstudium: ?? h (keine Angaben zum Arbeitsaufwand im Eigenstudium hinterlegt)
14	**Dauer des Moduls**	?? Semester (keine Angaben zur Dauer des Moduls hinterlegt)
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 798810	**Rechnerarchitektur** (Computer architecture)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Rechnerarchitektur (2 SWS) Übung: Übungen zu Rechnerarchitektur (2 SWS) Übung: Rechnerübungen zu Rechnerarchitektur (2 SWS)	2,5 ECTS 2,5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr.-Ing. Dietmar Fey Tobias Baumeister Sebastian Rachuj Christian Widerspick	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Dietmar Fey
5	**Inhalt**	Die Vorlesung baut auf die in den Grundlagen der Rechnerarchitektur und -organisation vermittelten Inhalte auf und setzt diese mit weiterführenden Themen fort. Es werden zunächst grundlegende fortgeschrittene Techniken bei Pipelineverarbeitung und Cachezugriffen in modernen Prozessoren und Parallelrechnern behandelt. Ferner wird die Architektur von Spezialprozessoren, z.B. DSPs und Embedded Prozessoren behandelt. Es wird aufgezeigt, wie diese Techniken in konkreten Architekturen (Intel Nehalem, GPGPU, Cell BE, TMS320 DSP, Embedded Prozessor ZPU) verwendet werden. Zur Vorlesung wird eine Tafelübung angeboten. Mit erfolgreicher mündlicher Prüfung können 5 ECTS erworben werden. In den Tafelübungen werden die in der Vorlesung vermittelten Techniken durch zu lösende Aufgaben vertieft. In der Rechnerübung soll u.a. ein einfacher Vielkern-Prozessor auf Basis des ZPU-Prozessors mit Simulationswerkzeugen aufgebaut werden. Im Einzelnen werden folgende Themen behandelt: • Organisationsaspekte von CISC und RISC-Prozessoren • Behandlung von Hazards in Pipelines • Fortgeschrittene Techniken der dynamischen Sprungvorhersage • Fortgeschritten Cachetechniken, Cache-Kohärenz • Ausnutzen von Cacheeffekten • Architekturen von Digitalen Signalprozessoren • Architekturen homogener und heterogener Multikern-Prozessoren (Intel Corei7, Nvidia GPUs, RISC-V) • Architektur von Parallelrechnern (Clusterrechner, Superrechner) • Effiziente Hardware-nahe Programmierung von Mulitkern-Prozessoren (OpenMP, SSE, CUDA) • Leistungsmodellierung und -analyse von Multikern-Prozessoren (Roofline-Modell)
6	**Lernziele und Kompetenzen**	Fachkompetenz Wissen

		Lernende können Wissen abrufen und wiedergeben. Sie kennen konkrete Einzelheiten wie Begriffe, Definitionen, Fakten, und Abläufe in einem Prozessor darlegen. Verstehen Lernende können Beispiele für Rechnerarchitekturen anführen, sie sind in der Lage, Schaubilder von Prozessoren zu interpretieren und die Abläufe in eigenen Worten zu beschreiben. Anwenden Lernende können beim Erstellen eigener Programme durch Transfer des Wissens über Interna von Prozesorarchitekturen Optimierungen hinsichtlich des Laufzeitverhaltens vornehmen. Analysieren Lernende können zwischen verschiedenen Varianten von Lösungen einer Prozessorarchitektur klassifizieren, die Gründe für durchgeführte Entwurfsentscheidungen erschließen, Unterscheide gegenüberstellen und gegeneinander bewerten. Lern- bzw. Methodenkompetenz Lernende erwerben die Fähigkeit selbstständig Testprogramme zum Bewerten der Leistungsfähigkeit eines Prozessors zu erstellen.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• Patterson/Hennessy: Computer Organization und Design • Hennessy/Patterson: Computer Architecture - A Quantitative Approach • Stallings: Computer Organization and Architecture

		• Märtin: Rechnerarchitekturen

1	**Modulbezeichnung** 333815	**Rechnerarchitektur (Vorlesung mit Übung und Rechnerübung)** (Computer architecture)	**7,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Rechnerarchitektur (2 SWS) Übung: Übungen zu Rechnerarchitektur (2 SWS) Übung: Rechnerübungen zu Rechnerarchitektur (2 SWS)	2,5 ECTS 2,5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr.-Ing. Dietmar Fey Tobias Baumeister Sebastian Rachuj Christian Widerspick	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Dietmar Fey
5	**Inhalt**	Die Vorlesung baut auf die in den Grundlagen der Rechnerarchitektur und -organisation vermittelten Inhalte auf und setzt diese mit weiterführenden Themen fort. Es werden zunächst grundlegende fortgeschrittene Techniken bei Pipelineverarbeitung und Cachezugriffen in modernen Prozessoren und Parallelrechnern behandelt. Ferner wird die Architektur von Spezialprozessoren, z.B. DSPs und Embedded Prozessoren behandelt. Es wird aufgezeigt, wie diese Techniken in konkreten Architekturen (Intel Nehalem, GPGPU, Cell BE, TMS320 DSP, Embedded Prozessor ZPU) verwendet werden. Zur Vorlesung werden eine Tafel- und eine Rechnerübung angeboten. Die Rechnerübung erfodert 11 erfolgreich abgeschlossene Übungsaufgaben, diese gehen mit 10% in die Modulgesamtnote ein. Die verbleibenden 90% werden durch die mündliche Prüfung bestimmt. Insgesamt werden 7,5 ECTS erworben. In den Tafelübungen werden die in der Vorlesung vermittelten Techniken durch zu lösende Aufgaben vertieft. In der Rechnerübung soll u.a. ein einfacher Vielkern-Prozessor auf Basis des ZPU-Prozessors mit Simulationswerkzeugen aufgebaut werden. Im Einzelnen werden folgende Themen behandelt: • Organisationsaspekte von CISC und RISC-Prozessoren • Behandlung von Hazards in Pipelines • Fortgeschrittene Techniken der dynamischen Sprungvorhersage • Fortgeschritten Cachetechniken, Cache-Kohärenz • Ausnutzen von Cacheeffekten • Architekturen von Digitalen Signalprozessoren • Architekturen homogener und heterogener Multikern-Prozessoren (Intel Corei7, Nvidia GPUs, RISC-V) • Architektur von Parallelrechnern (Clusterrechner, Superrechner) • Effiziente Hardware-nahe Programmierung von Mulitkern-Prozessoren (OpenMP, SSE, CUDA) • Leistungsmodellierung und -analyse von Multikern-Prozessoren (Roofline-Modell)

6	**Lernziele und Kompetenzen**	Fachkompetenz Wissen Lernende können Wissen abrufen und wiedergeben. Sie können konkrete Einzelheiten wie Begriffe, Definitionen, Fakten, und Abläufe in einem Prozessor darlegen. Verstehen Lernende können Beispiele für Rechnerarchitekturen anführen, sie sind in der Lage, Schaubilder von Prozessoren zu interpretieren und die Abläufe in eigenen Worten zu beschreiben. Anwenden Lernende können beim Erstellen eigener Programme durch Transfer des Wissens über Interna von Prozesorarchitekturen Optimierungen hinsichtlich des Laufzeitverhaltens vornehmen. Analysieren Lernende können zwischen verschiedenen Varianten von Lösungen einer Prozessorarchitektur klassifizieren, die Gründe für durchgeführte Entwurfsentscheidungen erschließen, Unterscheide gegenüberstellen und gegeneinander bewerten. Lern- bzw. Methodenkompetenz Lernende erwerben die Fähigkeit selbstständig Testprogramme zum Bewerten der Leistungsfähigkeit eines Prozessors zu erstellen.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 90 h Eigenstudium: 135 h
14	**Dauer des Moduls**	1 Semester

15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• Patterson/Hennessy: Computer Organization und Design • Hennessy/Patterson: Computer Architecture - A Quantitative Approach • Stallings: Computer Organization and Architecture • Märtin: Rechnerarchitekturen

1	**Modulbezeichnung** 43195	**Reconfigurable Computing** (Reconfigurable computing)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Reconfigurable Computing (2 SWS) Übung: Exercises to Reconfigurable Computing (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr.-Ing. Jürgen Teich Tobias Hahn Mauro Martin Letras Luna Pierre-Louis Sixdenier	

4	**Modulverantwortliche/r**	Joachim Falk Prof. Dr.-Ing. Jürgen Teich
5	**Inhalt**	**Content:** Reconfigurable (adaptive) computing is a novel yet important research field investigating the capability of hardware to adapt to changing computational requirements such as emerging standards, late design changes, and even to changing processing requirements arising at run-time. Reconfigurable computing thus benefits from a) the programmability of software similar to the Von Neumann computer and b) the speed and efficiency of parallel hardware execution. The purpose of the course reconfigurable computing is to instruct students about the possibilities and rapidly growing interest in adaptive hardware and corresponding design techniques by providing them the necessary knowledge for understanding and designing reconfigurable hardware systems and studying applications benefiting from dynamic hardware reconfiguration. After a general introduction about benefits and application ranges of reconfigurable (adaptive) computing in contrast to general-purpose and application-specific computing, the following topics will be covered: • Reconfigurable computing systems: Introduction of available technology including fine grained look up table (LUT-) based reconfigurable systems such as field programmable gate arrays (FPGA) as well as newest coarse grained architectures and technology. • Design and implementation: Algorithms and steps (design entry, functional simulation, logic synthesis, technology mapping, place and route, bit stream generation) to implement (map) algorithms to FPGAs. The main focus lies on logic synthesis algorithms for FPGAs, in particular LUT technology mapping. • Temporal partitioning: techniques to reconfigure systems over time. Covered are the problems of mapping large circuits which do not fit one single device. Several temporal partitioning techniques are studied and compared. • Temporal placement: Techniques and algorithms to exploit the possibility of partial and dynamic (run-time) hardware

		reconfiguration. Here, OS-like services are needed that optimize the allocation and scheduling of modules at run-time. • On-line communication: Modules dynamically placed at run-time on a given device need to communicate as well as transport data off-chip. State-of-the-art techniques are introduced how modules can communicate data at run-time including bus-oriented as well as network-on-a-chip (NoC) approaches. • Designing reconfigurable applications on Xilinx Virtex FPGAs: In this part, the generation of partial bitstreams for components to be placed at run-time on Xilinx FPGAs is introduced and discussed including newest available tool flows. • Applications: This section presents applications benefiting from dynamic hardware reconfiguration. It covers the use of reconfigurable systems including rapid prototyping, reconfigurable supercomputers, reconfigurable massively parallel computers and studies important application domains such as distributed arithmetic, signal processing, network packet processing, control design, and cryptography.
6	**Lernziele und Kompetenzen**	**Learning objectives and competencies:** Domain-specific knowledge • The students know to exploit run-time reconfigurable design methodologies for adaptive applications. Domain-specific comprehension • The students understand the mapping steps and optimization algorithms. • The students classify different types and kinds of reconfigurable hardware technologies available today. • The students clarify pros and cons of reconfigurable computing technology. • The students summarize applications benefiting from reconfigurable computing. • The students describe the design of circuits and systems-on-a-chip (SoC) on FPGAs.
7	**Voraussetzungen für die Teilnahme**	Selection of this module prohibits the selection of the modules "Reconfigurable Computing (Lecture with Exercises)" or "Reconfigurable Computing (Lecture with Extended Exercises)" by the student.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich (30 Minuten) Oral examination (Duration: 30 min).

11	**Berechnung der Modulnote**	mündlich (100%) The oral examination determines the final grade of the module.
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Further reading material: • The Hamburg VHDL Archive (see Documentation link for free books) http://tams-www.informatik.uni-hamburg.de/research/vlsi/vhdl/index.php • Interactive VHDL Tutorial with 150 examples from ALDEC http://www.aldec.com/downloads/ • Easy FPGA tutorials, projects, and boards http://www.fpga4fun.com • Xilinx WebPack ISE and Modelsim MXE (free FPGA synthesis tool and free VHDL simulator) http://www.xilinx.com/ise/logic_design_prod/webpack.htm • Symphony EDA free VHDL simulator (select FREE Edition license) http://www.symphonyeda.com/products.htm • Icarus open-source Verilog simulator http://www.icarus.com/eda/verilog/ Further information: https://www.cs12.tf.fau.de/lehre/lehrveranstaltungen/vorlesungen/reconfigurable-computing/

1	**Modulbezeichnung** 110334	**Reconfigurable Computing (Lecture with Extended Exercises)** (Reconfigurable computing (lecture with extended exercises))	**7,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Reconfigurable Computing (2 SWS) Übung: Extended Exercises to Reconfigurable Computing (2 SWS) Übung: Exercises to Reconfigurable Computing (2 SWS)	2,5 ECTS 2,5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr.-Ing. Jürgen Teich Pierre-Louis Sixdenier Mauro Martin Letras Luna Tobias Hahn	

4	**Modulverantwortliche/r**	Joachim Falk Prof. Dr.-Ing. Jürgen Teich
5	**Inhalt**	**Content:** Reconfigurable (adaptive) computing is a novel yet important research field investigating the capability of hardware to adapt to changing computational requirements such as emerging standards, late design changes, and even to changing processing requirements arising at run-time. Reconfigurable computing thus benefits from a) the programmability of software similar to the Von Neumann computer and b) the speed and efficiency of parallel hardware execution. The purpose of the course reconfigurable computing is to instruct students about the possibilities and rapidly growing interest in adaptive hardware and corresponding design techniques by providing them the necessary knowledge for understanding and designing reconfigurable hardware systems and studying applications benefiting from dynamic hardware reconfiguration. After a general introduction about benefits and application ranges of reconfigurable (adaptive) computing in contrast to general-purpose and application-specific computing, the following topics will be covered: • Reconfigurable computing systems: Introduction of available technology including fine grained look up table (LUT-) based reconfigurable systems such as field programmable gate arrays (FPGA) as well as newest coarse grained architectures and technology. • Design and implementation: Algorithms and steps (design entry, functional simulation, logic synthesis, technology mapping, place and route, bit stream generation) to implement (map) algorithms to FPGAs. The main focus lies on logic synthesis algorithms for FPGAs, in particular LUT technology mapping.

		• Temporal partitioning: techniques to reconfigure systems over time. Covered are the problems of mapping large circuits which do not fit one single device. Several temporal partitioning techniques are studied and compared. • Temporal placement: Techniques and algorithms to exploit the possibility of partial and dynamic (run-time) hardware reconfiguration. Here, OS-like services are needed that optimize the allocation and scheduling of modules at run-time. • On-line communication: Modules dynamically placed at run-time on a given device need to communicate as well as transport data off-chip. State-of-the-art techniques are introduced how modules can communicate data at run-time including bus-oriented as well as network-on-a-chip (NoC) approaches. • Designing reconfigurable applications on Xilinx Virtex FPGAs: In this part, the generation of partial bitstreams for components to be placed at run-time on Xilinx FPGAs is introduced and discussed including newest available tool flows. • Applications: This section presents applications benefiting from dynamic hardware reconfiguration. It covers the use of reconfigurable systems including rapid prototyping, reconfigurable supercomputers, reconfigurable massively parallel computers and studies important application domains such as distributed arithmetic, signal processing, network packet processing, control design, and cryptography.
6	**Lernziele und Kompetenzen**	**Learning objectives and competencies:** Domain-specific knowledge • The students know to exploit run-time reconfigurable design methodologies for adaptive applications. Domain-specific comprehension • The students understand the mapping steps, and optimization algorithms. • The students classify different types and kinds of reconfigurable hardware technologies available today. • The students clarify pros and cons of reconfigurable computing technology. • The students summarize applications benefiting from reconfigurable computing. Domain-specific practice • The students apply design tools for implementation of circuits and systems-on-a-chip (SoC) on FPGAs during practical training.

		Social competency • The students perform group work in small teams during practical training.
7	**Voraussetzungen für die Teilnahme**	Selection of this module prohibits the selection of the modules "Reconfigurable Computing" or "Reconfigurable Computing (Lecture with Exercises)" by the student.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio Oral examination (Duration: 30 min) and successful completion of all tasks of the extended exercises (mandatory, at the workstations residing in our lab at the chair).
11	**Berechnung der Modulnote**	Portfolio (100%) The oral examination determines the final grade of the module.
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 90 h Eigenstudium: 135 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Further reading material: • The Hamburg VHDL Archive (see Documentation link for free books) http://tams-www.informatik.uni-hamburg.de/research/vlsi/vhdl/index.php • Interactive VHDL Tutorial with 150 examples from ALDEC http://www.aldec.com/downloads/ • Easy FPGA tutorials, projects, and boards http://www.fpga4fun.com • Xilinx WebPack ISE and Modelsim MXE (free FPGA synthesis tool and free VHDL simulator) http://www.xilinx.com/ise/logic_design_prod/webpack.htm • Symphony EDA free VHDL simulator (select FREE Edition license) http://www.symphonyeda.com/products.htm • Icarus open-source Verilog simulator http://www.icarus.com/eda/verilog/ Further information: https://www.cs12.tf.fau.de/lehre/lehrveranstaltungen/vorlesungen/reconfigurable-computing/

1	**Modulbezeichnung** 97090	**Simulation und Modellierung I** (Simulation and modelling I)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Exercises to Simulation and Modeling 1 (2 SWS) Vorlesung: Simulation and Modeling 1 (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Lisa Maile Jonathan Fellerer Prof. Dr. Reinhard German	

4	**Modulverantwortliche/r**	Prof. Dr. Reinhard German
5	**Inhalt**	Das Modul vermittelt die Grundlagen der diskreten Ereignissimulation und beinhaltet • diskrete Simulation • analytische Modellierung (z.B. Warteschlangen) • Eingabemodellierung (z.B. Fitting-Verfahren) • Zufallszahlenerzeugung • statistische Ausgabeanalyse • Modellierungsparadigmen (u.a. Ereignis-/Prozessorientierung, Warteschlangen, Automaten, Petri-Netze, UML, graphische Bausteine) • kontinuierliche und hybride Simulation • Simulationssoftware • Fallstudien Content: Overview of the various kinds of simulation • discrete simulation (computational concepts, simulation of queuing systems, simulation in Java, professional simulation tools) • required probability concepts and statistics, modeling paradigms (e.g., event/process oriented, queuing systems, Petri nets, UML statecharts) • input modeling (selecting input probability distributions) • random number generation (linear congruential generators and variants, generating random variates) • output analysis (warm-up period detection, independent replications, result presentation) • continuous and hybrid simulation (differential equations, numerical solution, hybrid statecharts) • simulation software, case studies, parallel and distributed simulation.
6	**Lernziele und Kompetenzen**	Die Studierenden • erwerben Kenntnisse über Verfahren und Realisierungsmöglichkeiten der diskreten Simulation mit Ausblick auf andere Simulationsarten

		• erwerben Kenntnisse über statistische Aspekte der Simulation, die für die Anwendung wichtig sind • wenden statistische Methoden zur Analyse und Bewertung von Eingabe- sowie Ausgabedaten an • erwerben praktische Erfahrung mit kommerziellen Simulationswerkzeugen • erwerben Erfahrungen bei der Simulation in verschiedenen Anwendungsbereichen (u.a. Rechnernetze, Fertigungssysteme, Materialflusssysteme) • entwickeln eigenständig anhand von Beispielaufgaben Simulationsmodelle unter Verwendung verschiedener Modellierungsparadigmen • können in Gruppen kooperativ und verantwortlich arbeiten Learning targets and competences: Students gain knowledge about methods and realization possibilities of discrete simulation with an outlook on other types of simulation gain knowledge of statistical aspects of simulation that are important for practice apply statistical methods for analysis and evaluation of input and output data gain hands-on experience with commercial simulation tools gain experience in simulation in various fields of application (including computer networks, manufacturing systems, material flow systems) independently develop simulation models on the basis of sample tasks using different modeling paradigms can work in groups cooperatively and responsibly
7	**Voraussetzungen für die Teilnahme**	elementare Programmierkenntnisse, vorzugsweise in Java, Mathematikkenntnisse in Analysis, wie z.B. im 1. Semester der angewandten Mathematik vermittelt Recommended background knowledge: basic programming skills, preferably in Java, mathematics skills in analysis, such as taught in the first semester in applied mathematics.
8	**Einpassung in Studienverlaufsplan**	Semester: 1

9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Law, "Simulation Modeling and Analysis, 5th ed., McGraw Hill, 2014

1	**Modulbezeichnung** 44500	**Swarm Intelligence** (Swarm Intelligence)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übungen zu Swarm Intelligence (2 SWS) Vorlesung: Swarm Intelligence (SI), formerly Organic Computing (OC) (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Matthias Kergaßner Prof. Dr. Rolf Wanka	

4	**Modulverantwortliche/r**	Prof. Dr. Rolf Wanka
5	**Inhalt**	Swarm Intelligence (SI) is the design and deployment of self-organizing systems that dynamically adapt to their respective environmental needs. These systems are characterized by the fact that they feature the so-called self-*-properties, i.e., they are self-configuring, self-optimizing, self-healing, self-protecting, self-explanatory, ... Structures and methods of biological and other natural systems are chosen as models for such technical systems. In this module, Particle Swarm Optimization, Ant Algorithms, Web Search, and Evolutionary Algorithms are introduced and, as far as possible, mathematically analyzed.
6	**Lernziele und Kompetenzen**	Students learn advanced concepts of the current topic of swarm intelligence and how they can be successfully applied to solve continuous and discrete optimization problems and to data analysis. For this purpose, they know concrete details such as terms, definitions, facts, regularities and theories and learn how to apply the concepts to concrete problems, how to adjust the methods to the use case and how to analyze the computed solutions.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch

16	**Literaturhinweise**	• Ch. Müller-Schloer, Ch. von der Malsburg, R. P. Würt. Organic Computing. Informatik-Spektrum, Band 27, Nummer 4, S. 332-336. • I. C. Trelea. The particle swarm optimization algorithm: convergence analysis and parameter selection. Information Processing Letters 85 (2003) 317-325. • J. M. Kleinberg. Authoritative sources in a hyperlinked environment. Journal of the ACM 46 (1999) 604-632. • M. Dorigo. V. Maniezzo. A Colorni. Ant system: an autocatalytic optimizing process. Technical Report 91-016, Politecnico di Milano, 1991. • A. Badr. A. Fahmy. A proof of convergence for Ant algorithms. Information Sciences 160 (2004) 267-279. • M. Clerc. J. Kennedy. The particle swarm - Explosion, stability, and convergence in a multidimensional complex space. IEEE Transactions on Evolutionary Computation 8 (2002) 58-73

1	**Modulbezeichnung** 743260	**Verteilte Systeme (Vorlesung mit erweiterten Übungen)** (Distributed systems (lecture with extended exercises))	**7,5 ECTS**
2	Lehrveranstaltungen	Übung: Rechnerübungen zu Verteilte Systeme (2 SWS) Übung: Erweiterte Übungen zu Verteilte Systeme (2 SWS) Vorlesung: Verteilte Systeme (2 SWS)	- 5 ECTS 2,5 ECTS
3	Lehrende	PD Dr.Ing. Tobias Distler Laura Lawniczak	

4	**Modulverantwortliche/r**	PD Dr.Ing. Tobias Distler
5	**Inhalt**	Verteilte Systeme bestehen aus mehreren Rechnern, die über ein Netzwerk miteinander verbunden sind und einen gemeinsamen Dienst erbringen. Obwohl die beteiligten Rechner hierfür in weiten Teilen unabhängig voneinander agieren, erscheinen sie ihren Nutzern gegenüber in der Gesamtheit dabei trotzdem als ein einheitliches System. Die Einsatzmöglichkeiten für verteilte Systeme erstrecken sich über ein weites Spektrum an Szenarien: Von der Zusammenschaltung kleinster Rechenknoten zur Sammlung von Daten im Rahmen von Sensornetzwerken über Steuerungssysteme für Kraftfahrzeuge und Industrieanlagen bis hin zu weltumspannenden, Internet-gestützten Infrastrukturen mit Komponenten in Datenzentren auf verschiedenen Kontinenten. Ziel dieses Moduls ist es, die sich durch die speziellen Eigenschaften verteilter Systeme ergebenden Problemstellungen zu verdeutlichen und Ansätze zu vermitteln, mit deren Hilfe sie gelöst werden können; Beispiele hierfür sind etwa die Interaktion zwischen heterogenen Systemkomponenten, der Umgang mit erhöhten Netzwerklatenzen sowie die Wahrung konsistenter Zustände über Rechnergrenzen hinweg. Gleichzeitig zeigt das Modul auf, dass die Verteiltheit eines Systems nicht nur Herausforderungen mit sich bringt, sondern auf der anderen Seite auch Chancen eröffnet. Dies gilt insbesondere in Bezug auf die im Vergleich zu nicht verteilten Systemen erzielbare höhere Widerstandsfähigkeit eines Gesamtsystems gegenüber Fehlern wie den Ausfällen ganzer Rechner oder sogar kompletter Datenzentren. Ausgehend von den einfachsten, aus nur einem Client und einem Server bestehenden verteilten Systemen, beschäftigt sich die Vorlesung danach mit der deutlich komplexeren Replikation der Server-Seite und behandelt anschließend die Verteilung eines Systems über mehrere, mitunter weit voneinander entfernte geografische Standorte. In allen Abschnitten umfasst die Betrachtung des jeweiligen Themas eine Auswahl aus Grundlagen, im Praxiseinsatz befindlicher Ansätze und Techniken sowie für den aktuellen Stand der Forschung repräsentativer Konzepte. Im Rahmen der Übungen wird zunächst ein plattformunabhängiges Fernaufrufsystem schrittweise entwickelt und parallel dazu getestet.

		Als Vorlage und Orientierungshilfe dient dabei das in der Praxis weit verbreitete Java RMI. In den weiteren Übungsaufgaben stehen anschließend klassische Problemstellungen von verteilten Systemen wie fehlertolerante Replikation und verteilte Synchronisation im Mittelpunkt.
6	**Lernziele und Kompetenzen**	Studierende, die das Modul erfolgreich abgeschlossen haben: • beschreiben charakteristische Merkmale und Eigenschaften verteilter Systeme sowie grundlegende Probleme im Zusammenhang mit ihrer Realisierung. • untersuchen die Unterschiede zwischen lokalen Methodenaufrufen und Fernmethodenaufrufen. • vergleichen Ansätze zur Konvertierung von Nachrichten zwischen verschiedenen Datenrepräsenationen. • konzipieren eine eigene auf Java RMI basierende Anwendung. • entwickeln ein eigenes Fernaufrufsystem nach dem Vorbild von Java RMI. • bewerten die Serialisierungsroutinen von Java RMI. • erproben die manuelle Serialisierung von Nachrichten. • bewerten die Performanz des eigenen Fernaufrufsystems. • gestalten ein Modul zur Unterstützung verschiedener Fernaufrufsemantiken (Maybe, Last-of-Many, At-Most-Once) für das eigene Fernaufrufsystem. • beurteilen auf Basis eigener Experimente mit Fehlerinjektionen die Auswirkungen von Störeinflüssen auf verschiedene Fernaufrufsemantiken. • klassifizieren Mechanismen zur Bereitstellung von Fehlertoleranz, insbesondere verschiedene Arten der Replikation (aktiv vs. passiv). • vergleichen verschiedene Konsistenzgarantien georeplizierter Systeme. • illustrieren das Problem einer fehlenden gemeinsamen Zeitbasis in verteilten Systemen. • erforschen logische Uhren als Mittel zur Reihenfolgebestimmung und Methoden zur Synchronisation physikalischer Uhren. • unterscheiden grundlegende Zustellungs- und Ordnungsgarantien beim Multicast von Nachrichten. • gestalten ein Protokoll für den zuverlässigen und totalgeordneten Versand von Nachrichten in einer Gruppe von Knoten. • entwickeln einen Dienst zur Verwaltung verteilter Sperrobjekte auf Basis von Lamport-Locks. • bewerten die Qualität einer Publikation aus der Fachliteratur. • erschließen sich typische Probleme (Nebenläufigkeit, Konsistenz) und Fehlerquellen bei der Programmierung verteilter Anwendungen. • können in Kleingruppen kooperativ arbeiten. • können ihre Entwurfs- und Implementierungsentscheidungen kompakt präsentieren und argumentativ vertreten.

		• reflektieren ihre Entscheidungen kritisch und leiten Alternativen ab. • können offen und konstruktiv mit Schwachpunkten und Irrwegen umgehen.
7	**Voraussetzungen für die Teilnahme**	Gute Programmierkenntnisse in Java
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio Erfolgreiche Bearbeitung aller während des Semesters gestellten Übungsaufgaben (6 Aufgaben, Bewertung jeweils mit "ausreichend") + 30-minütige mündliche Prüfung am Ende des Semesters. Die Modulnote ergibt sich zu 100% aus der Bewertung der mündlichen Prüfung.
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 90 h Eigenstudium: 135 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 649073	**Verteilte Systeme (Vorlesung mit Übungen)** (Lecture and tutorial: Distributed systems)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Rechnerübungen zu Verteilte Systeme (2 SWS) Übung: Übungen zu Verteilte Systeme (2 SWS) Vorlesung: Verteilte Systeme (2 SWS)	- 2,5 ECTS 2,5 ECTS
3	Lehrende	PD Dr.Ing. Tobias Distler Laura Lawniczak	

4	**Modulverantwortliche/r**	PD Dr.Ing. Tobias Distler
5	**Inhalt**	Verteilte Systeme bestehen aus mehreren Rechnern, die über ein Netzwerk miteinander verbunden sind und einen gemeinsamen Dienst erbringen. Obwohl die beteiligten Rechner hierfür in weiten Teilen unabhängig voneinander agieren, erscheinen sie ihren Nutzern gegenüber in der Gesamtheit dabei trotzdem als ein einheitliches System. Die Einsatzmöglichkeiten für verteilte Systeme erstrecken sich über ein weites Spektrum an Szenarien: Von der Zusammenschaltung kleinster Rechenknoten zur Sammlung von Daten im Rahmen von Sensornetzwerken über Steuerungssysteme für Kraftfahrzeuge und Industrieanlagen bis hin zu weltumspannenden, Internet-gestützten Infrastrukturen mit Komponenten in Datenzentren auf verschiedenen Kontinenten. Ziel dieses Moduls ist es, die sich durch die speziellen Eigenschaften verteilter Systeme ergebenden Problemstellungen zu verdeutlichen und Ansätze zu vermitteln, mit deren Hilfe sie gelöst werden können; Beispiele hierfür sind etwa die Interaktion zwischen heterogenen Systemkomponenten, der Umgang mit erhöhten Netzwerklatenzen sowie die Wahrung konsistenter Zustände über Rechnergrenzen hinweg. Gleichzeitig zeigt das Modul auf, dass die Verteiltheit eines Systems nicht nur Herausforderungen mit sich bringt, sondern auf der anderen Seite auch Chancen eröffnet. Dies gilt insbesondere in Bezug auf die im Vergleich zu nicht verteilten Systemen erzielbare höhere Widerstandsfähigkeit eines Gesamtsystems gegenüber Fehlern wie den Ausfällen ganzer Rechner oder sogar kompletter Datenzentren. Ausgehend von den einfachsten, aus nur einem Client und einem Server bestehenden verteilten Systemen, beschäftigt sich die Vorlesung danach mit der deutlich komplexeren Replikation der Server-Seite und behandelt anschließend die Verteilung eines Systems über mehrere, mitunter weit voneinander entfernte geografische Standorte. In allen Abschnitten umfasst die Betrachtung des jeweiligen Themas eine Auswahl aus Grundlagen, im Praxiseinsatz befindlicher Ansätze und Techniken sowie für den aktuellen Stand der Forschung repräsentativer Konzepte.

		Im Rahmen der Übungen wird zunächst ein plattformunabhängiges Fernaufrufsystem schrittweise entwickelt und parallel dazu getestet. Als Vorlage und Orientierungshilfe dient dabei das in der Praxis weit verbreitete Java RMI. In den weiteren Übungsaufgaben stehen anschließend klassische Problemstellungen von verteilten Systemen wie fehlertolerante Replikation und verteilte Synchronisation im Mittelpunkt.
6	**Lernziele und Kompetenzen**	Studierende, die das Modul erfolgreich abgeschlossen haben: • beschreiben charakteristische Merkmale und Eigenschaften verteilter Systeme sowie grundlegende Probleme im Zusammenhang mit ihrer Realisierung. • untersuchen die Unterschiede zwischen lokalen Methodenaufrufen und Fernmethodenaufrufen. • vergleichen Ansätze zur Konvertierung von Nachrichten zwischen verschiedenen Datenrepräsenationen. • konzipieren eine eigene auf Java RMI basierende Anwendung. • entwickeln ein eigenes Fernaufrufsystem nach dem Vorbild von Java RMI. • gestalten ein Modul zur Unterstützung verschiedener Fernaufrufsemantiken (Maybe, Last-of-Many) für das eigene Fernaufrufsystem. • beurteilen auf Basis eigener Experimente mit Fehlerinjektionen die Auswirkungen von Störeinflüssen auf verschiedene Fernaufrufsemantiken. • klassifizieren Mechanismen zur Bereitstellung von Fehlertoleranz, insbesondere verschiedene Arten der Replikation (aktiv vs. passiv). • vergleichen verschiedene Konsistenzgarantien georeplizierter Systeme. • illustrieren das Problem einer fehlenden gemeinsamen Zeitbasis in verteilten Systemen. • erforschen logische Uhren als Mittel zur Reihenfolgebestimmung und Methoden zur Synchronisation physikalischer Uhren. • unterscheiden grundlegende Zustellungs- und Ordnungsgarantien beim Multicast von Nachrichten. • gestalten ein Protokoll für den zuverlässigen und totalgeordneten Versand von Nachrichten in einer Gruppe von Knoten. • entwickeln einen Dienst zur Verwaltung verteilter Sperrobjekte auf Basis von Lamport-Locks. • bewerten die Qualität einer Publikation aus der Fachliteratur. • erschließen sich typische Probleme (Nebenläufigkeit, Konsistenz) und Fehlerquellen bei der Programmierung verteilter Anwendungen. • können in Kleingruppen kooperativ arbeiten. • können ihre Entwurfs- und Implementierungsentscheidungen kompakt präsentieren und argumentativ vertreten.

		• können offen und konstruktiv mit Schwachpunkten und Irrwegen umgehen. • reflektieren ihre Entscheidungen kritisch und leiten Alternativen ab.
7	**Voraussetzungen für die Teilnahme**	Gute Programmierkenntnisse in Java
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Masterprüfung Master of Science Information and Communication Technology 20191 Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio Erfolgreiche Bearbeitung aller während des Semesters gestellten Übungsaufgaben (6 Aufgaben, Bewertung jeweils mit "ausreichend") + 30-minütige mündliche Prüfung am Ende des Semesters. Die Modulnote ergibt sich zu 100% aus der Bewertung der mündlichen Prüfung.
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

Pflichtmodule

1	**Modulbezeichnung** 93601	**Information Theory and Coding / Informationstheorie und Codierung** (Information theory and coding)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Tutorial for Information Theory and Coding (1 SWS) Vorlesung: Information Theory and Coding (3 SWS)	- 5 ECTS
3	Lehrende	Dr.-Ing. Ali Bereyhi Prof. Dr.-Ing. Ralf Müller	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Ralf Müller
5	**Inhalt**	1. Introduction: binomial distribution, (7,4)-Hamming code, parity-check matrix, generator matrix 2. Probability, entropy, and inference: entropy, conditional probability, Bayes law, likelihood, Jensens inequality 3. Inference: inverse probability, statistical inference 4. The source coding theorem: information content, typical sequences, Chebychev inequality, law of large numbers 5. Symbol codes: unique decidability, expected codeword length, prefix-free codes, Kraft inequality, Huffman coding 6. Stream codes: arithmetic coding, Lempel-Ziv coding, Burrows-Wheeler transform 7. Dependent random variables: mutual information, data processing lemma 8. Communication over a noisy channel: discrete memory-less channel, channel coding theorem, channel capacity 9. The noisy-channel coding theorem: jointly-typical sequences, proof of the channel coding theorem, proof of converse, symmetric channels 10. Error-correcting codes and real channels: AWGN channel, multivariate Gaussian pdf, capacity of AWGN channel 11. Binary codes: minimum distance, perfect codes, why perfect codes are bad, why distance isnt everything 12. Message passing: distributed counting, path counting, low-cost path, min-sum (=Viterbi) algorithm 13. Exact marginalization in graphs: factor graphs, sum-product algorithm

14. Low-density parity-check codes: density evolution, check node degree, regular vs. irregular codes, girth

15. Lossy source coding: transform coding and JPEG compression

--

1. Einleitung: Binomialverteilung, (7,4)-Hamming-Code, Paritätsmatrix, Generatormatrix

2. Wahrscheinlichkeit, Entropie und Inferenz: Entropie, bedingte Wahrscheinlichkeit, Bayessches Gesetz, Likelihood, Jensensche Ungleichung

3. Inferenz: Inverse Wahrscheinlichkeit, statistische Inferenz

4. Das Quellencodierungstheorem: Informationsgehalt, typische Folgen, Tschebyschevsche Ungleichung, Gesetz der großen Zahlen

5. Symbolcodes: eindeutige Dekodierbarkeit, mittlere Codewortlänge, präfixfreie Codes, Kraftsche Ungleichung, Huffmancodierung

6. Stromcodes: arithmetische Codierung, Lempel-Ziv-Codierung, Burrows-Wheeler-Transformation

7. Abhängige Zufallsvariablen: Transinformation, Datenverarbeitungslemma

8. Kommunikation over gestörte Kanäle: diskreter gedächtnisloser Kanal, Kanalcodierungstheorem, Kanalkapazität

9. Das Kanalcodierungstheorem: verbundtypische Folgen, Beweis des Kanalcodierungstheorems, Beweis des Umkehrsatzes, symmetrische Kanaäle

10. Fehlerkorrigierende Codes und reale Kanäle: AWGN-Kanal, mehrdimensionale Gaußsche WDF, Kapazität des AWGN-Kanals

11. Binäre Codes: Minimaldistanz, perfekte Codes, Warum perfekte Codes schlecht sind, Warum Distanz nicht alles ist

12. Nachrichtenaustausch: verteiltes Zählen, Pfadzählen, günstigster Pfad, Minimumsummenalgorithmus

13. Exakte Marginalisierung in Graphen: Faktorgraph, Summenproduktalgorithmus

14. LDPC-Codes: Dichteevolution, Knotenordnung, reguläre und irreguläre Codes, Graphumfang

		15. Verlustbehaftete Quellencodierung: Transformationscodierung und JPEG-Kompression
6	**Lernziele und Kompetenzen**	The students apply Bayesian inference to problems in both communications and everyday's life. The students explain the concept of digital communications by means of source compression and forward-error correction coding. For the design of communication systems, they use the concepts of entropy and channel capacity. They calculate these quantities for memoryless sources and channels. The students proof both the source coding and the channel coding theorem. The students compare various methods of source coding with respect to compression rate and complexity. The students apply source compression methods to measure mutual information. The students factorize multivariate functions, represent them by graphs, and marginalize them with respect to various variables. The students explain the design of error-correcting codes and the role of minimum distance. They decode error-correcting codes by means of maximum-likelihood decoding and message passing. The students apply distributed algorithms to problems in both communications and everydays life. The students improve the properties of low-density parity-check codes by widening the girth and/or irregularity in the degree distribution. The students transform source images into the frequency domain to improve lossy compression. -- Die Studierenden wenden Bayessche Inferenz auf Probleme in der Nachrichtentechnik und im Alltagsleben an. Die Studierenden erklären die konzeptuelle Trennung von digitaler Übertragung in Quellen- und Kanalcodierung.

		Kommunikationssysteme entwerfen sie unter Betrachtung von Entropie und Kanalkapazität. Sie berechnen diese Größen für gedächtnislose Quellen und Kanäle. Die Studierenden beweisen sowohl das Quellen- als auch das Kanalcodierungstheorem. Die Studierenden vergleichen verschiedenartige Quellencodierungsverfahren hinsichtlich Komplexität und Kompressionsrate. Die Studierenden verwenden Quellencodierverfahren zur Messung von Transinformation. Die Studierenden faktorisieren Funktionen mehrerer Veränderlicher, stellen diese als Graph dar und marginalisieren sie bezüglich mehrerer Veränderlicher. Die Studierenden erklären den Entwurf von Kanalcodes und den Einfluss der Minimaldistanz. Sie decodieren Kanalcodes gemäß maximaler Likelihood und Nachrichtenaustausch. Die Studierenden wenden verteilte Algorithmen auf Probleme der Nachrichtentechnik und des Alltagslebens an. Die Studierenden verbessern die Eigenschaften von LDPC-Codes durch Erhöhung des Umfangs und/oder durch irreguäre Knotenordnungsverteilungen. Die Studierenden transformieren Bildquellen zur Verbesserung verlustbehafteter Kompression iin den Frequenzbereich.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 2
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h

14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	MacKay, D.: Information Theory, Inference, and Learning Algorithms, Cambridge University Press, Cambridge, 2003.

1	**Modulbezeichnung** 43141	**Mobile Communications** (Mobile communications)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Mobile Communications - Tutorial (1 SWS) Vorlesung: Mobile Communications (3 SWS)	- 5 ECTS
3	Lehrende	Dr.-Ing. Ali Bereyhi Prof. Dr.-Ing. Ralf Müller	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Ralf Müller
5	**Inhalt**	History of mobile communications, cellular systems, sectorization, spectral efficiency, co-channel interference, adjacent-channel interference, near-far effect, cellular network architecture, antenna types and parameters, free space propagation, reflection, attenuation, diffraction, scattering, classification of channel models, ground reflection model, Okumura-Hata model, shadowing, narrow-band fading, time- variant channels, scattering function, delay-Doppler spectrum, diversity principles, combining methods, diversity gain, multiplexing, duplexing, digital modulation, Gaussian filtered minimum shift keying, basics of channel coding, interleaving, global system for mobile communications, physical versus logical channels, frame structure, call set-up, synchronization, channel estimation, hand-off
6	**Lernziele und Kompetenzen**	The students explain the cellular structure of mobile communication systems. They students explain the physical mechanics of radio wave propagation in the cm-band. The students explain the GSM cellular communications standard. The students discuss the pros and cons of several multiple-access and duplexing methods. The students discuss the pros and cons of several modulation and coding formats. The students decide which antenna type is suitable for a given morphological structure of the environment. The students predict the amplitude and dynamic of the antenuation between a mobile transmitter and a fixed receiver. The students utilize diversity methods to improve the link quality. The students determine the coverage probability of a given cellular communication system. The students collaborate on solving exercise problems. The students discuss which system solutions fit to which environments.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 2
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)

12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Proakis, J.: Digital Communications, McGraw-Hill, 4th ed., 2001. Rappaport, T.: Wireless Communications: Principles & Practice, Prentice Hall, 2nd ed., 2001. Mouly, M., Paulet, M.: The GSM System for Mobile Communications, Cell & SYS, France, 1992. Goldsmith, A.: Wireless Communications, Cambridge Univ. Press, 2005.

1	**Modulbezeichnung** 44362	**Quality of Service of Communication Systems** (Quality of service in communication systems)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Quality of Service in Communications (2 SWS) Übung: Quality of Service in Communications (Ex-QoSiC) (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr. Reinhard German Lisa Maile	

4	**Modulverantwortliche/r**	Prof. Dr. Reinhard German
5	**Inhalt**	Zunächst wird der Begriff der Dienstgüte (Quality-of-Service, QoS) eingegrenzt und es werden die wichtigsten Ansätze zur Erzielung von Dienstgüte besprochen und in ausgewählten Netztechnologien untersucht. Dann werden unterschiedliche Methoden vorgestellt, mit denen Systeme bezüglich ihrer Dienstgüte bewertet und ausgelegt werden können: • Netzplanung und optimierung, • stochastische Analyse (Markow-Ketten, Warteschlangen), • Netzwerksimulation, • deterministische Analyse mit Network Calculus zur Ermittlung von Dienstgütegarantien • Messung (HW-, SW-, Hybrid-Monitoring, Benchmarks). Alle Methoden werden an Beispielen demonstriert. *Contents:* We introduce the term quality-of-service (QoS), discuss important approaches to achieve certain degrees of QoS, and show how the implementation in computer networks. Then a number of methodologies to assess and design systems with respect to their QoS: • network planning and optimization, • network simulation, • stochastic analysis (Markov chains, non-Markovian models, queuing systems), • deterministic analysis with network calculus to determine QoS guarantees • measurements (hardware, software, and hybrid monitoring, benchmarks). All methods are illustrated by examples.
6	**Lernziele und Kompetenzen**	Die Studierenden erwerben • Kenntnisse in Methoden zur Modellierung und Bewertung von quantitativen, nicht-funktionalen Eigenschaften von vernetzten Systemen • Kenntnisse in Mechanismen von vernetzten Systemen zur Erzielung von Dienstgüte *Learning targets and competences:*

		The students get • experience in methods to model and evaluate quantitative, non-functional properties of computer networks and related systems • knowledge of mechanisms of computer networks to achieve quality-of-service
7	**Voraussetzungen für die Teilnahme**	Rechnerkommunikation, Kommunikationssysteme, grundlegende Programmierkenntnisse (optimal in R und C++)
8	**Einpassung in Studienverlaufsplan**	Semester: 2
9	**Verwendbarkeit des Moduls**	Pflichtmodule Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	• Kurose, Ross. Computer Networking: A Top-Down Approach Featuring the Internet. 6th Ed., Addison Wesley, 2013 • W. Stallings. Data and Computer Communications, 10th ed., Pearson Education, 2014 • W. Stallings. Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud, Pearson Education, 2016

Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication

1	**Modulbezeichnung** 151664	**Advanced Communication Networks** (Advanced communication networks)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Advanced Communication Networks - Tutorial (0 SWS) Vorlesung: Advanced Communication Networks (4 SWS)	- 5 ECTS
3	Lehrende	Walid Ghanem Prof. Dr. Laura Cottatellucci	

4	**Modulverantwortliche/r**	Prof. Dr. Laura Cottatellucci
5	**Inhalt**	Telecommunications have become ubiquitous in daily life and wireless networks play a fundamental role thanks to their capability to support mobility. In a wireless communication, the concept of link does not exist. Users radiate energy and communicate through the superposition of each others transmissions which creates interference. Compared to wireline networks this scenario is extremely challenging but also offers unpredictable opportunities in the development of new technologies (massive MIMO, cognitive radio, etc.) and exploitation of new features, e.g., opportunistic communications and multiuser diversity. The exponentially increasing request of higher and higher throughput is satisfied densifying users and access points per unit area and allowing more and more interference while adopting advanced techniques and innovative resource allocation to mitigate the detrimental effects of interference. Objective of this course is to introduce the student to advanced techniques for coordinated medium access control and radio resource management in cellular systems. Power allocation, rate adaptation and scheduling will be discussed both in centralized and distributed settings. Some mathematical methods play a fundamental role in resource allocation, namely, classical Perron-Frobenius theory for nonnegative matrices, convex and nonconvex constrained optimization, distributed optimization and game theory. The course introduces the student to such methods and exemplifies their application to various resource allocation problems. Additionally, the course addresses relevant aspects of resource allocation in wireless networks such as fairness and cross-layer design. *Technical Content* • Properties and challenges of the wireless medium. • Basic concepts of communication networks: the layered architecture. • Evolution of wireless cellular network architectures: From Global System for Mobile to Advanced-Long Term Evolution. • Multiple Access Schemes: CSMA variants, TDMA, FDMA, CDMA, OFDMA, SC-FDMA, SDMA. • Uplink-downlink duality. • Opportunistic scheduling and multiuser diversity.

		• Advanced concepts: small cells and heterogeneous networks, relaying and cooperation, network coding, cognitive radio networks. • Basics of resource allocation: power allocation, rate adaptation, and scheduling. • Classical resource allocation techniques: Centralized and distributed power control based on the Perron-Frobenius theorem. • Fundamentals of convex constrained optimization and application to resource allocation. • Resource allocation and fairness. • Fundamentals of nonconvex optimization and relaxation techniques. • Applications of nonconvex optimization to resource allocation. • Fundamentals of distributed optimization and applications to resource allocation. • Fundamental concepts of game theory. • Resource contention via game theoretical methods.
6	**Lernziele und Kompetenzen**	The student • Describes and/or recognizes wireless channel models. • Criticizes the limits of a layered architecture in wireless systems. • Defends the use of cross-layer design in wireless network. • Appraises and compares the distribution of functionalities in network entities for different architectures. • Argue on the pros and contras of different multiple access schemes according to various criteria (e.g. spectral efficiency, power efficiency, robustness to interference). • Compares and contrasts micro-diversity and various macro-diversity schemes. • Computes the total rate of SDMA with various receivers. • Relates the multiple access in uplink to broadcasting in downlink and justifies the concept of uplink-downlink duality. • Uses uplink-downlink duality to design a precoder and allocate power. • Contrasts multiple access in uplink and broadcasting in downlink in terms of channel state acquisition both for TDD and FDD transmission. • Uses multiuser diversity for opportunistic scheduling. • Compares multiuser diversity for users having identical and different channel statistics. • Contrasts opportunistic scheduling in terms of channel state acquisition and feedback both for uplink and downlink and for both FDD and TDD transmission schemes. • Appraises the impact of multiple antennas on opportunistic scheduling. • Analyses different settings with interference in small cells and designs countermeasures. • Categorizes relaying schemes in LTE.

		• Analyses performance of relaying schemes. • Argues on possible improvements of relaying schemes via network coding and physical layer network coding. • Uses the Perron-Frobenious theorem to allocate power in a centralized manner. • Judges the feasibility of a power control problems and formulates alternative approaches in case of unfeasibility. • Uses the Perron-Frobenious theorem to design a distributed power control scheme. • Judges the convergences of distributed power control based on the Perron-Frobenius theorem and appraises the robustness of asynchronous power control. • Applies techniques of convex optimization to discriminate convex problems and determine necessary and/or sufficient conditions for global optimality. • Judges the applicability of KKT conditions and duality. • Uses KKT conditions to solve convex optimization problems. • Uses duality to solve convex optimization problems. • Applies convex optimization to resource allocation in wireless communications. • Compares different definitions of fairness and applies them to rate allocation. • Appraises the effect of channel knowledge at the transmitter on different fairness criteria. • Applies KKT conditions for opportunistic user scheduling. • Describes a proportional fair algorithm for opportunistic scheduling. • Applies relaxation to nonconvex quadratic constrained quadratic programming. • Formulates resource allocation problems as constrained optimization programming. • Contrasts various distributed optimization methods. • Applies the concept of best response to determine Nash equilibria. • Argues about existence and uniqueness of Nash equilibria. • Assesses if a given game is a potential game and solves it. • Defends the concept of Pareto optimality in resource allocation. • Contrasts the concepts of pure and mixed strategies in game theory. • Uses coupled constrained concave game to allocate powers in heterogeneous networks.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222

10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 96875	**Ausgewählte Kapitel der Audiodatenreduktion** (keine englischsprachige Modulbezeichnung hinterlegt!)	**2,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Ausgewählte Kapitel der Audiodatenreduktion (2 SWS)	2,5 ECTS
3	Lehrende	Prof. Dr.-Ing. Jürgen Herre	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Jürgen Herre
5	**Inhalt**	Perceptual audio coding is ubiquitous in modern life (mp3 players, mobile phones, DVD players, computers, ...) Based on related classes (esp. Speech and Audio Processing"), this lecture aims at deepening the understanding of modern algorithms for perceptual source coding of audio. It includes an overview of the most relevant standardized coders, starting with MPEG-1 (incl. mp3) via MPEG-4 all the way to the most recent MPEG Audio standard. The significant algorithms are discussed and new approaches are described. The selected topics include: Efficient coding of several audio channels / parametric multi- channel coding Typical coding artifacts; subjective and objective quality assessment Scalable audio coding Bandwidth extension Semi-parametric audio coding Low-delay audio coding The lecture includes a number of demonstrations and audio examples to illustrate the discussed algorithms.
6	**Lernziele und Kompetenzen**	• Wissen - Die Studenten kennen die Hauptkomponenten eines gehörangepassten Audiocodecs, sowie die wichtigsten Algorithmen, Codierstrategien und Bewertungsmethoden. Weiterhin kennen sie die Terminologie und gängige Abkürzungen aus diesem Kontext. • Verstehen - Die Studenten verstehen, wie Designentscheidungen in Audiocodecs die letztendlich erreichte Audioqualität beeinflussen, verstehen die gebräuchlichsten Tools aus dem Bereich der gehörangepasste Audiocodierung und wie verschiedene Anwendungsszenarien das Coderdesign bestimmen. • Anwenden - Die Studenten können übliche mathematische Analysemethoden verwenden, um einfache Coder-Componenten zu beschreiben und gegebenenfalls zu modifizieren.

		• Analysieren - Die Studenten können Audiocodierungs-Standards und wahrnehmungsbasierte Messwerkzeuge dazu analysieren um die zugrundeliegenden Konzepte und Anforderungen zu erfassen. • Evaluieren (Beurteilen) - Die Studenten können Audiocodierungs-Standards und wahrnehmungsbasierte Messwerkzeuge evaluieren um zu beurteilen, welcher Standard bzw. welches Messwerkzeug das passendste ist für einen bestimmten Anwendungsfall. • Synthese - Die Studenten können eine Liste von Anforderungen und Bewertungskriterien für Audiocodecs zusammenstellen für gewünschte Anwendungsfälle. • Lern- bzw. Methodenkompetenz - Die Studenten hinterfragen bestehende Ansätze hinsichtlich ihrer Anwendbarkeit in der Praxis.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 96312	**Bild-, Video- und mehrdimensionale Signalverarbeitung** (Image, video and multidimensional signal processing)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Bild-, Video- und mehrdimensionale Signalverarbeitung (2 SWS)	5 ECTS
3	Lehrende	Prof. Dr.-Ing. Andre Kaup	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Andre Kaup
5	**Inhalt**	*Punktoperationen* Histogrammausgleich, Gamma-Korrektur *Binäroperationen* Morphologische Filter, Erosion, Dilatation, Opening, Closing *Farbräume* Trichromat, RGB- Farbraum, HSV-Farbraum *Mehrdimensionale Signale und Systeme* Theorie mehrdimensionaler Signale und Systeme, Impulsantwort, lineare Bildfilterung, Leistungsspektrum, Wiener Filter *Interpolation von Bildsignalen* Bilineare Interpolation, Bicubische Interpolation, Spline Interpolation *Merkmalsdetektion in Bildern* Bildmerkmale, Kantendetektion, Hough Transformation, Harris Ecken Detektor, Texturmerkmale, Grauwertematrix *Skalierungsraumdarstellung* LoG, DoG, SIFT, SURF *Bildabgleich* Projektive Abbildungen, Blockabgleich, Optischer Fluss, Merkmalsbasierter Abgleich mittels SIFT und SURF, RANSAC *Bildsegmentierung* Amplituden Schwellenwertermittlung, K-Means Clustering, Bayes Klassifikation, Regionen-basierte Segmentierung, kombinierte Segmentierung und Bewegungsschätzung, zeitliche Segmentierung von Videos

Bildverarbeitung im Transformationsbereich

Unitäre Transformation, Karhunen-Loeve Transformation, separable Transformationen, Haar und Hadamard Transformation, DFT, DCT

Content:

Point operations

Histogram equalization, gamma correction

Binary operations

Morphological filters, erosion, dilation, opening, closing

Color spaces

Trichromacy, red-green-blue color spaces, color representation using hue, saturation and value of intensity

Multidimensional signals and systems

Theory of multidimensional signals and systems, impulse response, linear image filtering, power spectrum, Wiener filtering

Interpolation of image signals

Bi-linear interpolation, bi-cubic interpolation, spline interpolation

Image feature detection

Image features, edge detection, Hough transform, Harris corner detector, texture features, co-occurrence matrix

Scale space representation

Laplacian of Gaussian, difference of Gaussian, scale invariant feature transform, speeded-up robust feature transform

Image matching

Projective transforms, block matching, optical flow, feature-based matching using SIFT and SURF, random sample consensus algorithm

Image segmentation

		Amplitude thresholding, k-means clustering, Bayes classification, region-based segmentation, combined segmentation and motion estimation, temporal segmentation of video *Transform domain image processing* Unitary transform, Karhunen-Loeve transform, separable transform, Haar and Hadamard transform, DFT, DCT
6	**Lernziele und Kompetenzen**	Die Studierenden • verstehen Punktoperationen an Bilddaten und Gamma-Korrektur • testen die Wirkung von Rangordnungs- und Medianfiltern an Bilddaten • unterscheiden und bewerten verschiedene Farbräume für Bilddaten • erklären das Prinzip der zwei-dimensionalen linearen Filterung für Bildsignale • berechnen und bewerten die zweidimensionale diskrete Fourier-Transformierte eines Bildsignales • bestimmen vergrößerte diskrete Bildsignale mit Methoden der bilinearen und Spline-Interpolation • überprüfen Bilddaten auf ausgewählte Textur-, Kanten- und Bewegungsmerkmale • analysieren Bild- und Videodaten auf Merkmale in unterschiedlichen Scale-Spaces • erläutern und beurteilen Methoden für das Matching von Bilddaten • segmentieren Bilddaten durch Programmierung von einfachen Klassifikations- oder Clustering-Verfahren • verstehen das Prinzip von Transformation auf Bilddaten und können diese an Beispielen anwenden. The students • understand point operations for image data and gamma correction • test the effects of rank order and median filters for image data • evaluate and differentiate between different color spaces for image data • explain the principle of two-dimensional linear filtering for image signals • calculate and evaluate the two-dimensional discrete Fourier transform of an image signal • determine enlarged discrete image signals by bi-linear and spline interpolation • verify image data for selected texture, edge and motion features

		• analyze image and video data for features in different scale spaces • explain and evaluate methods for the matching of image data • segment image data by implementing basic classification and clustering methods • understand the principle of transformations on image data and apply them exemplarily
7	**Voraussetzungen für die Teilnahme**	Vorlesung Signale und Systeme I und II
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot der Technischen Fakultät oder der Naturwissenschaftlichen Fakultät Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	J.-R. Ohm: |Multimedia Content Analysis|, Springer, 2016 J. W. Woods: |Multidimensional Signal, Image, and Video Processing and Coding|, Academic Press, 2^nd^ edition, 2012

1	**Modulbezeichnung** 43400	**Entzerrung und adaptive Systeme in der digitalen Übertragung** (Equalisation and adaptive systems for digital communications)	**2,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Equalization and Adaptive Systems for Digital Communications (2 SWS)	2,5 ECTS
3	Lehrende	Prof. Dr. Wolfgang Gerstacker	

4	**Modulverantwortliche/r**	Prof. Dr. Wolfgang Gerstacker
5	**Inhalt**	Bei der digitalen Übertragung spielen Kanalverzerrungen aufgrund ständig steigender Datenraten eine immer grössere Rolle. Bei vielen Anwendungen müssen für eine zuverlässige Übertragung komplexe Entzerrverfahren eingesetzt werden. Dies gilt sowohl für die leitungsgebundene als auch die drahtlose Kommunikation. Z.B. werden in der xDSL-Systemfamilie (Digital Subscriber Lines), die eine schnelle digitale Übertragung über Ortsanschlussleitungen gewährleistet, oft entscheidungsrückgekoppelte Entzerrverfahren oder Vorcodierungsverfahren eingesetzt und beim Mobilfunkstandard GSM und seiner Weiterentwicklung EDGE (Enhanced Data Rates for GSM Evolution) Maximum-Likelihood-Sequenzschätzung bzw. zustandsreduzierte Entzerrung. Eng im Zusammenhang mit der eigentlichen Entzerrung stehen Adaptionsverfahren, mit denen die Parameter des Entzerrers optimal an den Übertragungskanal angepasst werden können. Lernziel: Ziel der Vorlesung ist eine umfassende Darstellung gebräuchlicher Entzerrungs- und Adaptionsverfahren. Den Teilnehmern sollen fundierte Kenntnisse der verschiedenen Verfahren vermittelt werden, die sie zu deren sinnvollem Einsatz in der Praxis befähigen. Content: Channel distortions are playing an increasingly important role in digital transmission due to constantly increasing data rates. In many applications, complex equalization techniques must be used for a reliable transmission. This applies to both wired and wireless communication. For example, decision feedback equalization or precoding techniques are often used in the xDSL (Digital Subscriber Lines) system family, which ensures fast digital transmission over local subscriber loops, and the GSM system and its advanced version EDGE (Enhanced Data Rates for GSM Evolution) employ maximum likelihood sequence estimation and state-reduced equalization. Closely

		related to the task of equalization are adaptation methods with which the parameters of the equalizer can be optimally adjusted to the transmission channel. Objective: The aim of the lecture is a comprehensive presentation of common equalization and adaptation methods. The participants should acquire an in-depth knowledge of the various procedures which enables them to make meaningful design decisions in practice.
6	**Lernziele und Kompetenzen**	Die Studierenden • beschreiben verschiedene Verfahren zur Entzerrung frequenzselektiver Übertragungskanäle wie lineare Entzerrung, entscheidungsrückgekoppelte Entzerrung und Maximum-Likelihood-Sequenzschätzung, • setzen die verschiedenen Ansätze in Blockdiagramme um und optimieren deren Komponenten, • vergleichen Entzerrverfahren hinsichtlich ihrer Leistungsfähigkeit, charakterisiert durch die Fehlerrate, und Komplexität, • wählen geeignete Verfahren für verschiedene Anwendungen wie leitungsgebundene und drahtlose Übertragung aus, • entwerfen neuartige Verfahren für gegebene Anforderungen, • formulieren Adaptionsalgorithmen zur automatischen Anpassung des Empfängers eines Übertragungssystems an den Kanal, • ordnen Entzerrverfahren einen geeigneten Adaptionsalgorithmus zu. Learning Objectives and Competences: The students - describe various methods for equalizing frequency-selective transmission channels such as linear equalization, decision feedback equalization and maximum likelihood sequence estimation, - realize various approaches in block diagrams and optimize their components, - compare equalization methods in terms of their performance, characterized by the error rate, and complexity, - select suitable methods for various applications such as wired and wireless transmission, - design novel schemes for given requirements, - formulate adaptation algorithms for automatic adaptation of the receiver of a transmission system to the channel,

		- assign suitable adaptation algorithms to equalization schemes.
7	**Voraussetzungen für die Teilnahme**	Vorkenntnisse in Systemtheorie und digitaler Signalverarbeitung, sowie entweder der Vorlesung Nachrichtentechnische Systeme oder Digitale Übertragung sind für die Teilnahme hilfreich.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Gerstacker, W.: Skriptum zur Vorlesung Entzerrung und adaptive Systeme in der digitalen Übertragung. Huber, J.: Trelliscodierung, Springer Verlag, Berlin, 1992. Benedetto, S., Biglieri, E.: Principles of Digital Transmission with Wireless Applications, Kluwer Academic Publishers, New York, 1999. Proakis, J. G.: Digital Communications. McGraw-Hill, New York, 3. ed., 1995. Haykin, S.: Adaptive Filter Theory, Prentice Hall, Upper Saddle River, NJ, 3. ed., 1996.

1	**Modulbezeichnung** 96401	**Globale Navigationssatellitensysteme** (Global navigation satellite systems)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übung zu Globale Navigationssatellitensysteme (1 SWS) Vorlesung: Globale Navigationssatellitensysteme (3 SWS)	- 5 ECTS
3	Lehrende	Prof. Dr. Jörn Thielecke	

4	**Modulverantwortliche/r**	Prof. Dr. Jörn Thielecke
5	**Inhalt**	*Hinweis:* 1. Mehrere Übungsstunden werden rechnergestützt (Python) sein, um den Vorlesungsstoff durch eigene praktische Erfahrung zu vertiefen. 2. Eine Laborbesichtigung beim Fraunhofer Institut für Integrierte Schaltungen soll den Studierenden Einblick in die einschlägigen Arbeiten zu GPS und Galileo geben. *Inhalte:* *|1. Überblick: Signale und Systeme|* • Einführung • GPS Global Positioning System • Galileo • Satellitenergänzungssysteme: EGNOS, WAAS, LAAS • Mathematische Grundlagen: Navigationssignale, Gold Codes, Cramer-Rao-Schranke für Laufzeitmessungen *|2. Grundlagen und Funktiosweise der Satellitenortung|* • Koordinatensysteme, Zeitsysteme und Orbits • Ausbreitungsbedingungen und Fehlerquellen • Positions-, Geschwindigkeits- und Zeitschätzung • Hochgenaue Positionsschätzung mittels Trägerphase *|3. GNSS Empfänger|* • Signalkonditionierung • Leistungsfähigkeit der GPS- und Galileo-Signale • Releschleifen zur Signalverfolgung
6	**Lernziele und Kompetenzen**	1. Eine Übersicht über die Möglichkeiten von GPS und Galileo soll die Beurteilungsfähigkeit der Studierenden für neue Anwendungen schärfen. 2. Durch vertiefte Kenntnisse der Grundlagen, Funktionsweise und Fehlerquellen sollen die Studierenden die gelösten Herausforderungen und die Grenzen von GPS und Galileo einschätzen lernen.

		3. Die Studierenden sollen ein nachrichtentechnisches Verständnis für die Funktionsweise eines GPS-Empfängers erlangen.
7	**Voraussetzungen für die Teilnahme**	Keine formalen Voraussetzungen, grundlegende Kenntnisse erforderlich in: linearer Algebra, Signal- & Systemtheorie, Wahrscheinlichkeitstheorie
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%) Bei bestandener Prüfung wird die Note um eine Teilnotenstufe (z.B. von 2,0 auf 1,7) verbessert, wenn Sie: 1. Mindestens 75% der Hausaufgaben bestanden haben UND 2. Mindestens 75% der Rechnerübungen erfolgreich absolviert haben. Eine Note besser als 1,0 wird nicht vergeben.
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	1. Pratap Misra, Per Enge, Global Positioning System", Ganga-Jamuna Press, 2001 2. E.D. Kaplan, C.J. Hegarty, Understanding GPS Principles and Applications"Artech House, 2. Auflage, 2006 3. Werner Mansfeld, "Satellitenortung und Navigation, Vieweg, 2004

1	**Modulbezeichnung** 629391	**HF- und Digital-Architektur von Funksystemen** (RF and digital architecture of radio systems)	**2,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: RF and Digital Architecture of Radio Systems (2 SWS)	2,5 ECTS
3	Lehrende	Dr.-Ing. Heinrich Milosiu	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Georg Fischer
5	**Inhalt**	• Architektur von Funksystemen • Hochfrequenz- und Digital-Teil • Analog-Digital Balance • Vielfachzugriff • Leistungsmerkmale der physikalischen Schicht und ihre Implementierung • Transceiver-Architekturen • AD- und DA-Umsetzung • Beurteilung der Transceiverkomplexität • Grundanforderungen an Basisbandprozessoren • Signalintegrität und Analyse von Verzerrungen (EVM) • Elektronische Kompensationsverfahren • Digitale Signalkonditionierung • Spektrumszuweisung • Frequenzagile Funksysteme • Pegelrechnung zur Funkschnittstelle • Grundzüge der Verkehrstheorie und Netzdimensionierung • Funknetzplanung • Funkabdeckung, Netzkapazität und spektrale Effizienz • Leistungsverbrauch • Leistungshaushalt Content: Architecture of radio communication systems RF and digital section Analogue-digital Balance Multiple Access schemes Physical Layer features and their hardware implementation Transceiver architectures AD and DA conversion Transceiver complexity assessment Baseband processor fundamentals

		Signal integrity and impairment analysis (Error Vector Magnitude) Electronic counter measures Digital signal conditioning Spectrum licensing Frequency agile radio Link budget calculations Basic traffic theory and engineering Network planning Network coverage, capacity and spectral efficiency Power consumption Power budget
6	**Lernziele und Kompetenzen**	Die Studierenden verstehen die Architektur von Mobilfunk-Basisstationen Die Studierenden sind in der Lage Blockdiagramme für den Sende- und Empfangspfad zu erstellen. Die Studierenden kennen die Leistungsmerkmale des GSM Mobilfunksystems auf der Funkschnittstelle. Die Studierenden wissen welche Leistungsmerkmale die Netzabdeckung und welche die Netzkapazität erhöhen. Die Studierenden kennen die Implementierung von Leistungsmerkmalen. Die Studierenden kennen die Vor- und Nachteile verschiedener Sender- und Empfängerarchitekturen. Die Studierenden können einen Pegelplan für die Funknetzplanung erstellen. Die Studierenden können die Störquellen für Signalverzerrungen qualitativ und quantitativ beschreiben. Die Studierenden verstehen die Vor- und Nachteile analoger und digitaler Signalverarbeitung.

		Die Studierenden verstehen die Grundsätze der Verkehrstheorie und -modellierung. Die Studierenden kennen die Metriken zur Beschreibung der ökonomischen Zusammenhänge bei der Funknetzplanung. Die Studierenden können die Signaldynamik im Kontext der Analog-Digital-Umsetzung berechnen. Die Studierenden können den Leistungshaushalt einer Basisstation quantitativ beschreiben. Learning objectives and competencies: Stundents • can analyse the architecture of cellular basestations • can sketch block diagrams for the TX and RX path • can reproduce radio requirements of GSM physical layer • can identify radio features whether they increase coverage or capacity • can describe implementation of radio features • can analyse the pros and cons of different radio architectures • can sketch a level plan for link budget analysis • can describe sources of signal distortion in qualitative and quantitative way • can explain the pros and cons of analogue versus digital signal processing • can recall basics of traffic theory and modelling • can apply metrics for economical assessment with network planning • can evaluate the dynamic range in the context of Analog-Digital-Conversion • can analyse the power budget of a basestation in a quantitative way
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)

12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch oder Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 96260	**Integrierte Schaltungen für Funkanwendungen** (Integrierte Schaltungen für Funkanwendungen)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Integrierte Schaltungen für Funkanwendungen (2 SWS) Übung: Übungen zu Integrierte Schaltungen für Funkanwendungen (2 SWS)	5 ECTS -
3	Lehrende	Dr.-Ing. Heinrich Milosiu Albert-Marcel Schrotz	

4	**Modulverantwortliche/r**	Dr.-Ing. Heinrich Milosiu
5	**Inhalt**	• Transceiver-Architekturen • Hochfrequenzaspekte • Tranistoren und Technologien • Passive Bauelemente und Netzwerke • Rauscharme Vorverstärker • Mischer • Oszillatoren • Phasenregelschleifen und Synthesizer • Messtechnische Grundlagen
6	**Lernziele und Kompetenzen**	Nach der Teilnahme an den Modulveranstaltungen sind die Studierenden in der Lage: • Den Aufbau sowie Vor-und Nachteile von Transceiver-Architekturen zu verstehen • Hochfrequenzaspekte von Transistoren und Schaltungen zu analysieren • Geeignete Integrationstechnologien auszuwählen • Passive Bauelemente und Netzwerke zu verstehen und anzuwenden • Schaltungstopologien rauscharmer Vorverstärker, Mischer,Oszillatoren anzuwenden und zu analysieren
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester

15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 96270	**Kanalcodierung** (Channel coding)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung mit Übung: Kanalcodierung (3 SWS) Vorlesung mit Übung: Channel Coding (4 SWS) Übung: Übungen zur Kanalcodierung (1 SWS)	5 ECTS 5 ECTS -
3	Lehrende	Dr.-Ing. Clemens Stierstorfer	

4	**Modulverantwortliche/r**	Dr.-Ing. Clemens Stierstorfer
5	**Inhalt**	1) Introduction and Motivation 2) Fundamentals of Block Coding 3) Introduction to Finite Fields I 4) Linear Block Codes 5) Linear Cyclic Codes 6) Introduction to Finite Fields II 7) BCH and RS Codes 8) Convolutional Codes 9) Codes with Iterative Decoding
6	**Lernziele und Kompetenzen**	Das Modul Kanalcodierung umfasst eine Einführung in die Grundlagen der algebraischen, fehlerkorrigierenden Blockcodes sowie einen Einstieg in die Thematik der Faltungscodes. Iterativ decodierte Codeschemata wie Turbo-Codes und LDPC-Codes werden ebenfalls eingeführt. Im Einzelnen sind die Inhalte oben aufgeführt. Die Studierenden definieren die Problematik der Kanalcodierung, grenzen sie von anderen Codierverfahren (z.B. der Quellencodierung) ab und kennzeichnen die unterschiedlichen Ansätze zur Fehlerkorrektur und -erkennung. Sie nennen Beispiele für Einsatzgebiete von Kanalcodierung und geben einen Überblick über die historische Entwicklung des Fachgebiets. Die Studierenden erstellen Übertragungsszenarien für den Einsatz von Kanalcodierung bestehend aus Sender, Übertragungskanal und Empfänger und beachten dabei die Grundannahmen beim Einsatz von Blockcodes bzw. der Modellierung der Kanäle. Sie formulieren mathematische Beschreibungen der Encodierung sowie der optimalen Decodierung bzw. suboptimaler Varianten. Die Studierenden beherrschen die Grundlagen fehlerkorrigierender linearer Blockcodes, beschreiben diese mathematisch korrekt mittels Vektoren und Matrizen über endlichen Körpern und implementieren und bewerten zugehörige Encoder- und Decoderstrukturen insbesondere Syndromdecoder. Dabei modifizieren sie Generatormatrizen, ermitteln Prüfmatrizen und erstellen Syndromtabellen. Sie schätzen die minimale Hammingdistanz von Codes mittels Schranken ab und können den erzielbaren Codegewinn erläutern. Sie kennen und benutzen beispielhaften Codefamilien (z.B. Hamming-Codes, Simplex-Codes, Reed-Muller-Codes).

Die Studierenden erkennen die Vorteile zyklischer linearer Blockcodes und beschreiben diese mit Polynomen über endlichen Körpern. Sie nutzen die Restklassenrechnung bzgl. Polynomen zur Umsetzung systematischer Encoder und zur Realisierung von Syndromdecodern mittels Schieberegisterschaltungen. Sie kennen beispielhafte Codefamilien.

Die Studierenden nutzen Primkörper, Erweiterungskörper, Minimalpolynome und Kreisteilungsklassen sowie die Spektraldarstellung über endlichen Körpern zur Realisierung von BCH- und Reed-Solomon-Codes gemäß der BCH-Schranke. Sie verstehen die Grundlagen der Decodierung von BCH- und Reed-Solomon-Codes. Sie skizzieren und erläutern die Kanalcodierkonzepte von CD und DVD.

Die Studierenden erklären die Unterschiede von Faltungscodes und Blockcodes, skizzieren anhand von tabellierten Generatorpolynomen zugehörige Encoder und erläutern diese. Sie erklären die Funktionsweise des optimalen Decoders (MLSE) und demonstrieren diese beispielhaft.

Die Studierenden verstehen die Grundlagen der iterativen Decodierung, insbesondere wenden sie die Grundlagen des Information Combining zur Kombination von verschiedenen Beobachtungen an. Sie verstehen die Bedeutung von Log-Likelihood-Ratios bei iterativen Decodieruvorgängen und berechnen diese. Sie skizzieren die grundlegenden Encoder- und Decoderstrukturen von Turbo-Codes und die Grundzüge der Codierung mit LDPC-Codes u.a. der Decodierung mittels Belief Propagation.

Die Vorlesung erfolgt wechselweise auf Deutsch oder Englisch (Winter/ Sommer). Die zur Verfügung gestellten Unterlagen sind ausschließlich in Englisch gehalten. Die Studierenden verwenden entweder die englischen Fachtermini sicher oder kennen diese und drücken sich sicher mit den entsprechenden deutschen Fachbegriffen aus.

Die Umsetzung der angegebenen Algorithmen in eine Programmiersprache (C, Matlab usw.) sollten die Studierenden zu diesem Zeitpunkt des Studiums üblicherweise beherrschen. Übungen hierzu bleiben der Eigeninitiative überlassen.

Students define the problems of channel coding, how to distinguish it from other coding methods (such as source coding) and how to describe the various different approaches to error correction and detection. They are able to list example application areas of channel coding and give an overview of the historical development of the field.

		Furthermore, they describe and analyze transmission scenarios for the application of channel coding which consist of transmitter, transmission channel and receiver, taking into account the general assumptions for applying block codes or modeling the channels. They formulate mathematical descriptions of encoding, optimal decoding and sub-optimal methods. Students illustrate the principles of error-correcting linear block codes and describe them mathematically using vectors and matrices over finite fields. They implement and analyze corresponding encoder and decoder structures, in particular syndrome decoders, and modify generator matrices, construct test matrices and create syndrome tables. They estimate the minimum Hamming distance of codes using bounds and are able to explain the coding gain that can be achieved in individual cases. They analyze and use example code families (e.g. Hamming codes, simplex codes, Reed-Muller codes). Students explain the advantages of cyclic linear block codes and how to describe them with polynomials over finite fields. They apply polynomial modular arithmetic to implement systematic encoders and realize syndrome decoders using shift register circuits. They know and use exemplary code families. Students use prime fields, extension fields, minimal polynomials and cyclotomic cosets, and spectral representation over finite fields to implement BCH and Reed-Solomon codes using the BCH bound. They understand the foundations of decoding BCH and Reed-Solomon codes and how to sketch and explain the channel coding concepts of CDs and DVDs. Students are able to describe the differences between convolutional codes and block codes, to sketch the respective encoders based on tabulated generator polynomials and to explain them. They are able to explain how optimal decoders (MLSE) work using examples. Students sketch the foundations of iterative decoding. In particular, they apply methods of information combining to combine different observations. They use and calculate log-likelihood ratios in iterative decoding processes, sketch the basic encoding and decoding structures of turbo codes and the basics of coding using LDPC codes (including decoding using belief propagation). Students either are able to use the English technical terms correctly or know them and are able to express themselves using the respective technical terms in German.
7	**Voraussetzungen für die Teilnahme**	Es ist hilfreich, wenn die Studierenden die erlernten Algorithmen in eine Programmiersprache (C, Matlab usw.) umsetzen können.

		It would be very helpful if the participants can implement the specified algorithms into a programming language (C, Matlab, etc.).
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich Hilfsblatt, Taschenrechner: Sie können ein einzelnes A4-Blatt (Vorder- und Rückseite oder andere Blätter mit offensichtlich identischer Gesamtfläche) verwenden, um Ihre eigene, handschriftliche Formelsammlung aufzuschreiben. Sie können einen nicht programmierbaren Taschenrechner verwenden. Cheat Sheet, Calculator: A single A4 sheet (front and back, or any other collection of sheets with an obviously identical total area size) can be used to write down your own handwritten collection of formulas, etc. You may also bring a non-programmable calculator.
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch oder Englisch
16	**Literaturhinweise**	• J. Huber, R. Fischer, C. Stierstorfer: Folien zur Vorlesung • M. Bossert: Kanalcodierung, Oldenbourg Wissenschaftsverlag, 3. Auflage, 2013 • M. Bossert: Channel Coding for Telecommunications, John Wiley & Sons, 1999 • B. Friedrichs: Kanalcodierung, Springer Verlag, 1996 • S.B. Wicker: Error Control Systems for Digital Communications and Storage, Prentice-Hall, 1995

1	**Modulbezeichnung** 92730	**Kommunikationselektronik** (Communications electronics 1)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Kommunikationselektronik (2 SWS)	5 ECTS
3	Lehrende	Nives Berner Dr.-Ing. Jörg Robert Clemens Neumüller	

4	**Modulverantwortliche/r**	Dr.-Ing. Jörg Robert
5	**Inhalt**	1.Einleitung 2.Darstellung von Signalen und Spektren • Kontinuierliche und diskrete Signale • Spektrum eines Signals • Unterabtastung und Überabtastung 3.Aufbau und Signale eines Software Defined Radio Systems • Blockschaltbild eines Software Defined Radio Systems • Basisband- und Trägersignale • Empfänger-Topologien • Signale in einem Software Defined Radio System 4.Drahtlose Netzwerke 5.Übertragungsstrecke • Funkstrecke • Antennen 6.Leistungsdaten eines Empfängers • Rauschen • Nichtlinearität • Dynamikbereich eines Empfängers 7.Digital Downconverter • CIC-Filter • Polyphasen-FIR-Filter • Halbband-Filterkaskade • Interpolation 8.Demodulation digital modulierter Signale • Einführung • Demodulation einer GFSK/PAM-Paketsendung Das Modul Kommunikationselektronik behandelt Aspekte der Schaltungstechnik und der Signalverarbeitung drahtloser Übertragungssysteme, die als sog. "Software Defined Radio" Systeme

aufgebaut sind. Als Beispiel dient der Empfänger eines einfachen Telemetrie-Systems, der von der Antenne bis zum Nutzdatenausgang behandelt wird. Schwerpunkte bilden der Aufbau und die Eigenschaften der Hardware des Empfängers sowie die Algorithmen zum Empfang von Telemetrie-Signalen. Dabei wird ein typisches System mit Hilfe eines miniaturisierten Empfängers und einer Verarbeitung mit dem MATLAB-kompatiblen Mathematikprogramm Octave implementiert. Die benötigte Software wird den Studierenden zur Verfügung gestellt.

Content:

1. Introduction

2. Signal representation and discrete signals

   a. Continuous and discrete signals

   b. Signal spectrum

   c. Downsampling and upsampling

3. Structure and signals of a Software Defined Radio

   a. Block diagram of a Software Defined Radio

   b. Base band signals and carrier signals

   c. Receiver topologies

   d. Signals in a Software Defined Radio

4. Wireless networks

5. Transmission path

   a. Radio link

   b. Antennas

6. Performance data of a receiver

   a. Noise

   b. Nonlinearities

   c. Dynamic range of a receiver

		7. Digital Down Converter a. CIC filter b. Polyphase FIR filter c. Halfband filter cascade d. Interpolation 8. Demodulation of digital modulated signals a. Introduction b. Demodulation of a GFSK/PAM packet transmission The module Communication Electronics deals with aspects of circuitry and signal processing of wireless communication systems, built up as so-called "Software Defined Radio systems. A receiver of a simple telemetry system serves as an example, being examined starting from its antenna to the user data output. The focus lies on the structure and the characteristic of the receivers hardware as well as the algorithms for the reception of telemetry signals. A typical system is implemented using a miniaturized receiver and processing with the MATLAB-compatible Octave math program. The required software is provided to the students.
6	**Lernziele und Kompetenzen**	1. Die Studierenden verstehen die grundlegende Funktionsweise eines Software Defined Radio (SDR) Systems, d.h. sie verstehen die Funktionsweise der einzelnen Signalverarbeitungsschritte sowie die auftretenden Signale selbst. 2. Die Studierenden analysieren die Leistungsfähigkeit der analogen Komponenten eines SDR Systems und können Verfahren zur Optimierung dieser Komponenten selbständig anwenden. 3. Die Studierenden analysieren die digitalen Verarbeitungsschritte ausgewählter Modulationsarten und können damit selbst die digitale Signalverarbeitung eines SDR Senders und Empfängers erschaffen. 1. The students will understand the basic operation of a Software Defined Radio (SDR) system, i.e. the students will understand how the individual signal processing steps work as well as the signals themselves.

		2. The students analyze the performance of the analog components of an SDR system and are able to apply procedures for optimizing these components independently. 3.The students analyse the digital processing steps of selected modulation types and are able to create the digital signal processing of an SDR transmitter and receiver themselves.
7	**Voraussetzungen für die Teilnahme**	Keine formalen Voraussetzungen, grundlegende Kenntnisse im Bereich digitaler Signalverarbeitung werden vorausgesetzt
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20191 Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Skriptum zur Veranstaltung im StudON verfügbar: https://www.studon.fau.de/studon/goto.php?target=crs_117973

1	**Modulbezeichnung** 96801	**Kommunikationsstrukturen** (Communication structures)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Kommunikationsstrukturen (2 SWS) Übung: Übungen zu Kommunikationsstrukturen (2 SWS)	5 ECTS 5 ECTS
3	Lehrende	Jürgen Frickel	

4	**Modulverantwortliche/r**	Jürgen Frickel
5	**Inhalt**	Einführung • Information und Kommunikation • Anwendungsgebiete - Kommunikation **Strukturen und Eigenschaften von Kommunikationssystemen** • Grundlegende Definitionen und Klassifikationen • Grundlegende Strukturen **Protokolle und Schnittstellen** • Grundlagen • Basis-Verfahren und Beispiele • TCP/IP-Protokol • Referenzmodell nach ISO/OSI • Sicherungsschicht/Data Link Layer (LLC und MAC) • Bitübertragungsschicht/Physical Layer • Übertragungsmedien **Hardware in Kommunikationsstrukturen** • HW-Architekturen und Funktionsblöcke • Digitale und Analoge Komponenten • Schaltungsdetails von Komponenten **Grundlagen von Bussystemen** • Klassifikation • Funktionale Eigenschaften • Arbitrierungs-Verfahren **Leitungsgebundene Anwendungen für Rechnersysteme** • Bus-Applikationen • Baustein-/IC-interne Busse (AMBA, FPI, ConTraBus, .) • Baugruppeninterne Busse (I2C, Chipsätze+Bridges, .) • Busse für Rechensysteme (VME, ISA, PCI, PCIe, AGP, .) • Peripherie-Busse (ATA, IEC, USB, Firewire, Fibre Channel, Thunderbolt .) **Leitungsgebundene Anwendungen in Systemen**

		• Feldkommunikation • Automobil, Luftfahrt, Space (CAN, MOST, LIN, MILBus, Spacewire .) • Industrie, Haustechnik (Profibus, EIB, .) • Weitverkehrsnetze • SDH, PDH, ATM,
6	**Lernziele und Kompetenzen**	1. Die Studierenden werden in die Lage versetzt die Konzepte und Verfahren vor allem drahtgebundener Kommunikationssysteme anzuwenden. 2. Die Studierenden lernen die Funktionsweise und den Einsatzzweck diverser Kommunikationsprotokolle zu verstehen, und miteinander zu vergleichen. 3. Desweiteren analysieren und klassifizieren Sie grundlegende Strukturen von leitungsgebundenen Kommunikationssystemen anhand ihrer funktionalen Eigenschaften.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 668129	**Machine Learning in Communications** (Machine learning in communications)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Machine Learning in Communications (4 SWS) Übung: Tutorial for Machine Learning in Communications (0 SWS)	5 ECTS -
3	Lehrende	Prof. Dr. Laura Cottatellucci Nikita Shanin	

4	**Modulverantwortliche/r**	Prof. Dr. Laura Cottatellucci
5	**Inhalt**	Recently, in many areas of wireless communications such as wireless sensor networks (WSNs), heterogeneous networks and complex ad hoc networks, distributed graph algorithms and machine learning on graphs are gaining relevance as fundamental tools in network analysis and information processing. This motivates to deliver a general introduction to fundamentals of machine learning such as detection of clusters on graphs. The introduction is followed by the application of machine learning to the design of physical and data layer techniques in wireless communications and in the optimization of mobile networks.
6	**Lernziele und Kompetenzen**	The students • know and explain the fundamentals of machine learning with special attention to machine learning over graphs. • apply these principles in the design and optimisation of wireless communications systems and mobile networks.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20191 Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 48440	**Machine Learning in Signal Processing** (Machine learning in signal processing)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Supplements for Machine Learning in Signal Processing (1 SWS) Vorlesung: Machine Learning in Signal Processing (3 SWS)	- 5 ECTS
3	Lehrende	Kamal Gopikrishnan Nambiar Prof. Dr. Vasileios Belagiannis	

4	**Modulverantwortliche/r**	PD Dr.Ing. Jürgen Seiler
5	**Inhalt**	This course is an introduction into machine learning and artificial intelligence. The special emphasis is on applications to modern signal processing problems. The course is focused on design principles of machine learning algorithms. The lectures start with a short introduction, where the nomenclature is defined. After this, probabilistic graphical models are introduced and the use of latent variables is discussed, concluding with a discussion of hidden Markov models and Markov fields. The second part of the course is about deep learning and covers the use of deep neural networks for machine learning tasks. In the last part of the lecture, the use of deep neural networks for speech processing tasks is introduced. The course is based on the materials and video footage from Dr. Roland Maas. He is an outstanding machine learning expert and a former member of the Chair of Multimedia Communications and Signal Processing.
6	**Lernziele und Kompetenzen**	After attending the lecture, students will be able to • understand regression and classification problems • apply PDF estimation algorithms • understand Gaussian mixture models and expectation-maximization • apply principal component analysis and independent component analysis • assess different estimation algorithms • explain the application of machine learning to system identification • apply hidden Markov models • understand different artificial neural network architectures • explain deep learning principles • apply artificial neural networks • devise learning strategies for deep neural networks • assess the application of deep neural networks for speech processing tasks.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1

9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich/mündlich
11	**Berechnung der Modulnote**	schriftlich/mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Literature: • C. M. Bishop: Pattern Recognition and Machine Learning, http://www.research.microsoft.com/en-us/um/people/cmbishop/PRML • S. Theodoridis and K. Koutroumbas: Pattern Recognition • M. Nielsen: Neural Networks and Deep Learning.

1	**Modulbezeichnung** 96030	**Medizinelektronik** (Medical electronics)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Medizinelektronik - Übung / Medical Electronics Exercises (2 SWS) Vorlesung: Medizinelektronik - Medical Electronics (2 SWS)	- 5 ECTS
3	Lehrende	Thomas Kurin Dr. Jens Kirchner	

4	**Modulverantwortliche/r**	Dr. Jens Kirchner
5	**Inhalt**	The Lecture and exercise deals with the following topics: • Electronics for medical diagnostics and therapy • Challenges for medical engineering from demographic development and epidemiology of common diseases • Concepts for chronic disease management and elderly care • Regulatory framework of circuit design for medical devices • Circuit design of standard medical equipment ECG, EEG, EMG, SpO2 • Sensor principles and circuit design for biosignal acquisition • Analog-digital balance • Energy management for medical devices • Body near energy harvesting • Health data transmission • Electronic systems for ambient assisted living (AAL) • Circuit technology for lab-on-chip and microelectromechanical systems (MEMS) • Circuit technology for implants and wearable systems
6	**Lernziele und Kompetenzen**	Students will gain • Substantial knowledge on principles of circuit design for medical electronic devices • Substantial knowledge on circuit design for standard medical devices, e.g. ECG, EEG, EMG • Substantial knowledge on design of medical sensors • Substantial knowledge on system design for health assistance systems, wearable medical devices and implants • Ability to analyze circuit diagrams of medical electronic devices • Ability to separate medical electronic devices into their subfunctions • Ability to analyze energy budget of medical devices, particularly wearable systems • Basic ability to design electronic circuits to comply with regulatory requirements
7	**Voraussetzungen für die Teilnahme**	Completion of the modules "Circuit design" ("Schaltungstechnik") or "Electronics and circuit design" ("Elektronik und Schaltungstechnik") is recommended before attending the course.
8	**Einpassung in Studienverlaufsplan**	Semester: 1

9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 96300	**MIMO Communication Systems** (MIMO communication systems)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: MIMO Communication Systems (3 SWS) Übung: MIMO Communication Systems - Tutorial (1 SWS)	5 ECTS -
3	Lehrende	Prof. Dr.-Ing. Robert Schober Hedieh Ajam	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Robert Schober
5	**Inhalt**	Modern communication systems employ multiple antennas at the transmitter and/or receiver creating a multiple-input multiple-output (MIMO) system. This course covers the fundamental mathematical and communication theoretical concepts necessary for the design and analysis of MIMO communication systems. Relevant topics include MIMO Channel Capacity, Receive Diversity, Transmit Diversity, Space-Time Coding, Spatial Multiplexing, MIMO Transceiver Design, Multi-user MIMO, Massive MIMO, Relay-based MIMO, and applications in modern communication systems.
6	**Lernziele und Kompetenzen**	The students • learn about different MIMO channel models, • analyze MIMO communication systems with respect to their channel capacity and reliability, • determine MIMO figures of merit such as coding gain, diversity gain, and multiplexing gain, • compare and evaluate different MIMO receiver designs, • characterize the rate region of multiuser systems, • analyze massive MIMO systems, • discuss the advantages and disadvantages of different relay network architectures. Die Studierenden • lernen verschiedene MIMO-Kanalmodelle kennen, • analysieren MIMO-Kommunikationssysteme hinsichtlich der Kanalkapazität und Zuverlässigkeit, • ermitteln MIMO-Kenngrößen wie Codierungsgewinn, Diversitätsgewinn und Multiplexgewinn, • vergleichen und beurteilen verschiedene MIMO-Empfangsstrategien, • charakterisieren die Ratenregion von Mehrteilnehmersystemen, • analysieren Massive-MIMO-Systeme, • diskutieren die Vor- und Nachteile verschiedener Relaisnetzwerkarchitekturen.
7	**Voraussetzungen für die Teilnahme**	Basic course in communications
8	**Einpassung in Studienverlaufsplan**	Semester: 1

9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20191 Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 849203	**Optische Kommunikationsnetze** (Optical communication networks)	**2,5 ECTS**
2	Lehrveranstaltungen	Zu diesem Modul sind keine Lehrveranstaltungen oder Lehrveranstaltungsgruppen hinterlegt!	
3	Lehrende	Zu diesem Modul sind keine Lehrveranstaltungen und somit auch keine Lehrenden hinterlegt!	

4	**Modulverantwortliche/r**	Prof. Dr. Herbert Haunstein
5	**Inhalt**	Global communication between billions of subscribers utilizing a multitude of devices is accomplished over a trans-continental fiber-optic transport network. End users worldwide access this network over copper cable (xDSL, HFC), by wireless technologies like WLAN, GSM, UMTS, LTE and also via GPON, EPON and WDM-PON (PON: Passive Optical Network). After a short distance ("the last mile") data streams from many users are aggregated (e.g. by IP routers) into higher data rate transport streams, which are then carried over cost-efficient and highly reliable optical connections. Rapid increase of data traffic has quickly evolved from Gigabit Ethernet (1GbE) to 10GbE and 100GbE data rates. To operate optical networks on a global scale, standards like OTN (Optical Transport Network) have been developed to provide high capacity links by use of many wavelengths together with operations and maintenance (OAM) functions. Automated protection and restauration schemes provide a high level of availabilty and can guarantee carrier-grade Quality of Service (QoS). Future data rate increase will be driven by video streaming as well as the introduction of 5G wireless technology and the Internet of Things (IoT). The course shall provide a fundamental understanding of modern fiber optic networks from fixed and mobile access through metropolitan area to core networks. 1) Introduction – Evolution of optical networks 2) Network layers - Internet Protocol – TCP/IP 3) Label switching – MPLS – MPLS-TP 4) Quality of Service - traffic classification – resource allocation 5) Ethernet - switching and physical transport 6) Optical Transport Network - OTN 7) Optical fiber properties – optical amplification 8) Optical transmitter – laser – modulator 9) Optical receiver – photo detection – Clock&Data recovery – Bit Error Ratio calculation 10) Modulation formats – transmission - margin allocation 11) Coherent detection – optical signal processing

		12) Optical networks – optical switching 13) Network control & automation
6	**Lernziele und Kompetenzen**	Students ... • explain the functional building blocks of optical networks • can eloborate on the different tasks provided by the logical/ control plane (routing), the physical layer and transmission/data plane of optical networks • refer which standardisation organisation contributes to the different function of optical networks • explain the purpose of different protocols that interact along an end-to-end communication channel • describe technologies for E/O and O/E conversion and optical switches • express the design challenges of future optical systems for fixed and mobile access, data center interconnects, • metro-regional, core, ultra-long-haul and submarine networks
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch oder Englisch
16	**Literaturhinweise**	1) R. Ramaswami and K.N. Sivarajan: “Optical Networks”, Morgan Kaufman Publishers, 1998 2) U. Black: “Optical Networks - Third generation transport systems”, Prentice Hall, 2002 3) P. Tomsu and Chr. Schmutzer: “Next generation optical networks”, Prentice Hall, 2002 4) M. Bossert, M. Breitbach: “Digitale Netze”, Teubner Verlag, 1997 5) I. Kaminow and T. Li (eds.): “Optical fiber telecommunications IVA+B”, Academic Press, 2002 6) D.E. Comer, „Computernetworks and Internets, Pearson“, 2009 7) G.P. Agrawal, “Fiber optic communication systems”, Wiley, 1992, (new 1997) 8) G.P. Agrawal, “Nonlinear fiber optics”, Academic Press, 1995

		9) K. Petermann: “Laser Diode Modulation and Noise”, Kluver, 1991 10) I. Kazovsky et al., „Optical Fiber Communication Systems“, Artech House, 1996 11) K.-P. Ho, „Phase-Modulated Optical Communication Systems“, Springer 2005 12) H. Haunstein, Presentation material (slides) of the lectures (in English)

1	**Modulbezeichnung** 92400	**Optische Übertragungstechnik** (Optical communication systems)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Optische Übertragungstechnik Übung (2 SWS) Vorlesung: Optische Übertragungstechnik (2 SWS)	- 5 ECTS
3	Lehrende	Esther Renner Benedikt Beck Prof. Dr.-Ing. Bernhard Schmauß	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Bernhard Schmauß
5	**Inhalt**	Kommerzielle Optische Kommunikationssysteme erreichen pro Faser Übertragungskapazitäten von mehreren Tbit/s. Im Labor wurden mehr als 100Tbit/s nachgewiesen. Die Realisierung derartiger Systeme setzt die Beherrschung verschiedenster Techniken der optischen Übertragungstechnik voraus. In der Vorlesung werden Techniken des Zeitbereichs - (TDM) und Wellenlängenmultiplex (WDM), aber besonders auch der Auslegung der Übertragungsstrecke (Link Design) auf der Basis entsprechender physikalischer und signaltheoretischer Grundlagen behandelt und vertieft. Dabei werden Verfahren besprochen, die sicherstellen, dass sowohl die Signalverzerrungen durch lineare und nichtlineare Fasereffekte als auch die Akkumulation des Verstärkerrauschens begrenzt bleiben. Es wird ausführlich die Systemoptimierung hinsichtlich des optischen Signal-Rausch-Verhältnisses (OSNR) diskutiert sowie auf Techniken des Dispersions- und Nichtlinearitätsmanagements (z.B. Solitonenübertragung) eingegangen. Hierbei wird dem Themenkomplex einer optimalen Streckenauslegung besonders eingehend behandelt. In der Folge werden verschiedene, gebräuchliche Modulationsverfahren einschließlich kohärenter Übertragungsverfahren behandelt, die in neueren Systemen eingesetzt und in experimentellen Systemen getestet werden. Eine Besprechung optischer Verfahren zur Signalregeneration bildet die Brücke zu aktuellen eigenen Forschungsarbeiten. Die vermittelten Grundlagen werden in der Übung zur Vorlesung durch praxisnahe und anschauliche Simulationsbeispiele vertieft.
6	**Lernziele und Kompetenzen**	Die Studierenden • besitzen spezialisiertes und vertieftes Wissen über die Konzeption und Struktur verschiedener optischer Übertragungssysteme. • können die Qualität optischer Datensignale im Kontext verschiedener Systemkonzepte vergleichen und bewerten • sind in der Lage Streckenauslegungen zu entwickeln und zu optimieren. • besitzen methodische Kenntnis zur Bestimmung und Verbesserung der Leistungsfähigkeit optischer Übertragungsstrecken unter Einbeziehung aktueller wissenschaftlicher Ergebnisse.
7	**Voraussetzungen für die Teilnahme**	Komponenten optischer Kommunikationssysteme hilfreich aber nicht obligatorisch

8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Agrawal, G.P.: Fiber-Optic Communication Systems, John Wiley & Sons, 1997 Agrawal, G.P.: Nonlinear Fiber Optics, John Wiley & Sons, 3. Auflage, 2001 Kaminow, I, Koch, T.: Optical Fiber Telecommunications IVA, Academic Press, 2002 Skriptum zur Vorlesung Kaminow, I, Li, T., Willner,A.: Optical Fiber Telecommunications VA, Academic Press, 2008

1	**Modulbezeichnung** 43460	**Satellitenkommunikation** (Satellite communication)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übung Satellitenkommunikation (2 SWS) Vorlesung: Satellitenkommunikation (2 SWS)	- 5 ECTS
3	Lehrende	Marcelo Michael Dr. Christian Rohde	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Albert Heuberger
5	**Inhalt**	Nach einem historischen Rückblick zur Entwicklung der Satellitenkommunikation werden die einzelnen Komponenten eines typischen Gesamtsystems (Boden- und Raumsegment) näher betrachtet. Hierzu zählt der prinzipielle Aufbau von Trägerraketen, von Satelliten (Satellitenplattformen, Subsysteme, Nutzlasten), die meist genutzten Umlaufbahnen und die verschiedenen Kommunikationsverbindungen (Uplink, Downlink, Inter-Satellite-Link). Die Besonderheiten der Signalausbreitung und -übertragung über große Entfernungen zwischen Bodenstationen und Satelliten werden erklärt und mit Beispielen ergänzt. Dabei wird insbesondere eingegangen auf verwendete Frequenzen, Signaldispersion und -dämpfung, atmosphärische Effekte sowie Störeinflüsse der Weltraumumgebung. Die Architektur transparenter und regenerativer Kommunikationseinheiten wird ausführlich an Beispielen kommerziell verfügbarer Transponder und Onboard-Prozessoren erklärt. Die Prinzipien moderner, standardisierter Verfahren zur Signalaufbereitung und Übertragung von Video-/Bild und Audiosignalen über Satellit (z.B. MPEG, H.264/265, DVB-S/-S2/-S2X) werden erläutert und diskutiert. Dies umfasst Verfahren zur Quellencodierung, Kanalcodierung und Modulation, Kanalzugriff und -diversität. Außerdem wird auf die im Orbit und im kommerziellen Einsatz befindlichen Kommunikationssatelliten und der damit verbundenen großen Dienstevielfalt eingegangen wie z.B. bei TV- und Breitbandversorgung sowie in Mobilkommunikationssystemen. Abschließend werden einige Herausforderungen und Forschungsansätze im Zusammenhang mit den neuen Megakonstellationen und Next Generation High Throughput Satellites (HTS) für zukünftige Satellitensysteme vorgestellt. Die in der Vorlesung behandelten physikalischen, elektro- und nachrichtentechnischen Zusammenhänge werden in den ergänzenden Übungen mit Rechenbeispielen vertieft. Gliederung der Vorlesung: **1. Einführung:**

Überblick über die Hauptkomponenten, Satelliten, Anwendungen und Dienste, sowie Orbits, Aufgaben und Frequenzen der Satellitennetzwerke

2. Historie der Satellitenkommunikation:

Wichtige Meilensteine, Entwicklung in Europa und Deutschland

3. Orbits und Konstellationen:

Keplersche Gesetze, Beschreibung von Orbits, verwendete Umlaufbahnen, Bodenspuren, erreichbare Abdeckung

4. Trägersysteme:

Trägerraketen, Entwicklung, Anbietermarkt, Nutzlastfähigkeit, Startplätze, Startverlauf

5. Satellitenaufbau:

Auswahl aktueller Satellitenplattformen, Satellitenaufbau, Plattformkomponenten, Montageschritte und Tests

6. Satellitennutzlast (Payload):

Komponenten, Industrielle Beispiele, Aufbau und Aufgaben der Payload, Transponderarchitekturen, Antennen

7. Signalausbreitung und Leistungsbilanz:

Signalausbreitung, Freiraumverluste, Signaldämpfung, Rauschen, Signal-Rausch-Verhältnis, Linkbudget

8. Weltraumumgebung: Weltraumumgebungsbedingungen, Einflüsse auf den Satelliten und die Elektronik der Nutzlast

9. Quellencodierung:

Audio-, Bild- und Videokompression das Content des Satellitenfernsehens

10. Signalmodulation und Kanalcodierung:

Signalkonstellationen, Modulation und Codes zur Fehlerkorrektur

11. Diversitäts- und Zugriffsverfahren:

Medium Access, Duplextechniken, Multiplexmethoden, Diversitätstechniken

12. Moderne Satellitenkommunikationssysteme:

Rundfunksysteme wie Sirius XM Satellite Radio, zellulare Internetversorgung mittels Satellitenkommunikation

13. Neueste Themen aus Forschung und Entwicklung

SatKom auf StudOn: http://www.studon.uni-erlangen.de/crs117969.html

After a historical retrospective about the developments in satellite communication, the core components of a typical satellite system (ground- and space-segment) are introduced. The principles and architectures of rockets/ carriers, satellites (platform, subsystems, payload), used orbits, and the various communication links (uplink, downlink, inter-satellite-link) are shown. The special features and properties of signal transmission over such large distances are explained and stuffed with examples. In particular, more details are provided on the used frequencies, signal dispersion and attenuation, atmospheric effects as well as impairments due to space environment.

The architecture of transparent and regenerative communication payloads are described in detail, accompanied by corresponding examples of commercially used transponders and onboard-processors and their technology.

The principles of modern standardized methods for signal transmission and preparation of video-/image- and audio-signals via satellite, e.g., MPEG, H.264/265, DVB-S/-S2/-S2X, are illustrated and discussed. This includes methods for efficient source coding, channel coding and modulation, channel access and diversity schemes.

Furthermore, the currently available communication satellites in orbit and the related variety of commercial services are introduced like, e.g., TV- and broadband services as well as mobile communication services and systems. Based on that, a few challenges and perspectives for research and development for future satellite systems are highlighted with respect to the upcoming new mega constellations and next generation high throughput satellites (HTS).

		The physical, electro-technical and communications concepts and schemes shown in the lectures are complemented by tutorials with sample calculations. Table of contents: **1. Introduction:** Overview of main components, satellites, applications and services, orbits, tasks, frequencies, satellite networks **2. History of satellite communications:** Major milestones, development in Europe and Germany **3. Orbits and constellations:** Kepler's laws, description of orbits, orbits used, ground tracks, achievable coverage **4. Launcher systems:** Launch vehicles, providers, payload capabilities, launch sites, launch history **5. Satellite structure:** Selection of current satellite platforms, satellite structure, platform components, assembly steps and tests **6. Payload:** Components, structure and tasks of payload, transponder architecture, antennas **7. Signal propagation and link budget:** Signal propagation, free space losses, signal attenuation, noise, signal to noise ratio, link budget **8. Space environment:** Space environmental conditions, influences on the satellites and payload electronics **9. Source coding:** Audio, image and video compression - the satellite TV broadcasting content **10. Signal modulation and channel coding:** Signal constellations, modulation and error correction coding **11. Diversity and access schemes:** Medium access, duplex methods, multiplex methods, diversity techniques **12. Modern satellite communications systems:** Broadcasting systems like Sirius XM Satellite Radio, satellite cellular broadband communication **13. Latest topics in research and development**
6	**Lernziele und Kompetenzen**	• Die Studierenden bekommen einen guten Überblick über alle Aspekte der Satellitenkommunikation inklusive Historie.

		• Die Studierenden lernen die weltweit führenden oder in Europa ansässigen Firmen und Organisationen kennen, die in den Bereichen Satellitenbau und -betrieb, Satellitendienste bzw. -anwendungen, sowie Forschung und Entwicklung tätig sind. • Die Studierenden können die Herausforderungen der Weltraumumgebung sowie Vor- und Nachteile verschiedener Orbits einschätzen und wichtige Kenngröße berechnen • Die Studierenden lernen die Signalverarbeitungsschritte im Sender, Satelliten und Empfänger kennen - von der Audio/Video-Quelle über Link-Budget-Berechnungen bis zur Datensenke. • Die Studierenden lernen den Aufbau und wichtige Kenngrößen von Satelliten, Konstellationen und Launchern kennen und dabei verwendete Konzepte zu unterscheiden und zu klassifizieren bzgl. deren Vor- und Nachteilen.
7	**Voraussetzungen für die Teilnahme**	Keine formalen Voraussetzungen
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich (91 Minuten)
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Skriptum zur Lehrveranstaltung

1	**Modulbezeichnung** 96410	**Schaltungen und Systeme der Übertragungstechnik** (Schaltungen und Systeme der Übertragungstechnik)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übungen zu Schaltungen und Systeme der Übertragungstechnik (2 SWS) Vorlesung: Schaltungen und Systeme der Übertragungstechnik (2 SWS)	- 5 ECTS
3	Lehrende	Victor Shatov Prof. Dr.-Ing. Norman Franchi Maximilian Lübke Maximilian Lübke	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Norman Franchi
5	**Inhalt**	Im Rahmen dieses Modules werden die Grundlagen und technische Ausführung Übertragungstechniken vermittelt. Fokus liegt dabei auf dem Automotivebereich. Elektrofahrzeuge werden nicht nur die heute bereits in der Oberklasse verfügbaren Fahrassistenzsysteme nutzen sondern weitere E-Mobility spezifische Anwendung insbesondere zur Energie- und Reichweitoptimierung. Drahtlose Kommunikation zwischen Fahrzeug und Ladeeinrichtungen, zwischen Fahrzeugen untereinander, genaue Ortung und Streckenprognose sowie autonomes energiesparendes Fahren mit Radar-Abstandsreglung spielen hier eine wichtige Rolle. In diesem Modul werden diese modernen Entwicklungen adressiert und die dafür notwendigen Grundlagen erarbeitet. Grundlagen: • Funkkanaleigenschaften • Modellierung • Modulation, Codierung, Vielfachzugriff Fahrzeugkommunikationssysteme: • Übertragungssysteme für die Fahrassistenz • Car-to-Car und Car-to-X-Kommunikation • Breitbandige In-Car-Datenübertragung Fahrzeugsensorik: • Fahrzeugortung (lokal und global) • Automobilradar und Umfeldüberwachung • Sensorische Erfassung von Bioparametern im Fahrzeug
6	**Lernziele und Kompetenzen**	Die Studierenden sind nach der erfolgreichen Teilnahme am Modul in der Lage: Funkkanaleigenschaften und Modelle für spezifische Anwendungs- und Betriebsszenarien anzuwenden

		Modulationstechniken zu erläutern und zu analysieren Moderne Codierungs- und Vielfachzugriffstechniken zu erläutern Architekturen und Anwendungen von Fahrzeugkommunikationssystemen zu erläutern und zu analysieren Architekturen und Anwendungen von Fahrzeugsensoriksystemen zu erläutern und zu analysieren
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 96430	**Statistical Signal Processing** (Statistical signal processing)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Ergänzungen und Übungen zur statistischen Signalverarbeitung (1 SWS) Vorlesung: Statistische Signalverarbeitung (3 SWS)	- 5 ECTS
3	Lehrende	Thomas Haubner Prof. Dr.-Ing. Walter Kellermann	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Walter Kellermann
5	**Inhalt**	The course concentrates on fundamental methods of statistical signal processing and their applications. The main topics are: *Discrete-time stochastic processes in the time and frequency domain* Random variables (RVs), probability distributions and densities, expectations of random variables, transformation of RVs, vectors of normally distributed RVs, time-discrete random processes: probability distribution and densities, expectation, stationarity, cyclostationarity, ergodicity, correlation functions and correlation matrices, spectral representations, principal component analysis (PCA), Karhunen-Loève transform (KLT). *Estimation theory* estimation criteria, prediction, classical and Bayesian parameter estimation (including MMSE, Maximum Likelihood, and Maximum A Posteriori estimation), Cramer-Rao bound *Linear signal models* Parametric models (cepstral decomposition, Paley-Wiener theorem, spectral flatness), non-parametric models (all-pole, all-zero and pole-zero models, lattice structures, Yule-Walker equations, PARCOR coefficients, cepstral representation) *Signal estimation* Supervised estimation, problem classes, orthogonality principle, MMSE estimation, linear MMSE estimation for normally distributed random processes, optimum FIR filtering, optimum linear filtering for stationary processes, prediction and smoothing, Kalman filters, optimum multichannel filtering (Wiener filter, LCMV, MVDR, GSC) *Adaptive filtering* Gradient methods, LMS, NLMS, APA and RLS algorithms and their convergence behavior

		Zeitdiskrete Zufallsprozesse im Zeit- und Frequenzbereich Zufallsvariablen (ZVn), Wahrscheinlichkeitsverteilungen und dichten, Erwartungswerte; Transformation von ZVn; Vektoren normalverteilter ZVn; zeitdiskrete Zufallsprozesse (ZPe): Wahrscheinlichkeitsverteilungen und dichten, Erwartungswerte, Stationarität, Zyklostationarität, Ergodizität, Korrelationsfunktionen und -matrizen, Spektraldarstellungen; Principal Component Analysis, Karhunen-Loeve Transformation; *Schätztheorie* Schätzkriterien; Prädiktion; klassische und Bayessche Parameterschätzung (inkl. MMSE, Maximum Likelihood, Maximum A Posteriori); Cramer-Rao-Schranke *Lineare Signalmodelle* Parametrische Modelle (Cepstrale Zerlegung, Paley-Wiener Theorem, Spektrale Glattheit); Nichtparametrische Modelle: Allpole-/Allzero-/ Pole-zero-(AR/MA/ARMA) Modelle; Lattice-Strukturen, Yule-Walker Gleichungen, PARCOR-Koeffizienten, Cepstraldarstellungen; *Signalschätzung* Überwachte Signalschätzung, Problemklassen; Orthogonalitätsprinzip, MMSE-Schätzung, lineare MMSE-Schätzung für Gaußprozesse; Optimale FIR-Filter; Lineare Optimalfilter für stationäre Prozesse; Prädiktion und Glättung; Kalman-Filter; optimale Multikanalfilterung (Wiener-Filter, LCMV, MVDR, GSC); *Adaptive Filterung* Gradientenverfahren; LMS-, NLMS-, APA- und RLS-Algorithmus und Ihr Konvergenzverhalten.
6	**Lernziele und Kompetenzen**	The students: • analyze the statistical properties of random variables, random vectors, and stochastic processes by probability density functions and expectations as well as correlation functions and matrices and their frequency-domain representations • know the Gaussian distribution and its role to describe the properties of random variables, vectors and processes • understand the differences between classical and Bayesian estimation, derive and analyze MMSE and ML estimators for specific estimation problems, especially for signal estimation • analyze and evaluate optimum linear MMSE estimators (single- and multichannel Wiener filter and Kalman filter) for direct and inverse supervised estimation problems

		• evaluate adaptive filters for the identification of optimum linear estimators. Die Studierenden • analysieren die statistischen Eigenschaften von Zufallsvariablen, -vektoren und stochastischen Prozessen mittels Wahrscheinlichkeitsdichten und Erwartungswerten, bzw. Korrelationsfunktionen, Korrelationsmatrizen und deren Frequenzbereichsdarstellungen • kennen die spezielle Rolle der Gaußverteilung und ihre Auswirkungen auf die Eigenschaften von Zufallsvariablen, -vektoren und Prozessen • verstehen die Unterschiede klassischer und Bayesscher Schätzung, entwerfen und analysieren MMSE- und ML-Schätzer für spezielle Schätzprobleme, insbesondere zur Signalschätzung • analysieren und evaluieren lineare MMSE-optimale Schätzer (ein- und vielkanalige Wiener-Filter und Kalman-Filter) für direkte und inverse überwachte Schätzprobleme; • evaluieren adaptive Filter zur Identifikation optimaler linearer Signalschätzer
7	**Voraussetzungen für die Teilnahme**	Module Signale und Systeme I und Signale und Systeme II, Digitale Signalverarbeitung oder gleichwertige
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Pflichtmodule Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	A. Papoulis, S. Pillai: Probability, Random Variables and Stochastic Processes; McGraw-Hill, 2002 (englisch) D. Manolakis, V. Ingle, S. Kogon: Statistical and Adaptive Signal Processing; Artech House, 2005 (englisch)

1	**Modulbezeichnung** 43420	**Transmission and Detection for Advanced Mobile Communications** (Transmission and detection for advanced mobile communications)	**2,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Transmission and Detection for Advanced Mobile Communications (2 SWS)	2,5 ECTS
3	Lehrende	Prof. Dr. Wolfgang Gerstacker	

4	**Modulverantwortliche/r**	Prof. Dr. Wolfgang Gerstacker
5	**Inhalt**	The aim of this lecture is that the students acquire a basic knowledge of advanced transmission and detection techniques which are relevant to practical mobile communications systems. In the first part, it is shown how equalization schemes like decision-feedback equalization (DFE) and maximum-likelihood sequence estimation (MLSE) can be applied to the GSM/EDGE (Enhanced Data Rates for GSM Evolution) standard. Also, channel estimation for GSM/EDGE is covered. In GSM/EDGE, disturbance by interfering signals of other users is a further major problem. Therefore, interference cancellation algorithms are discussed in detail. The cases of several receive antennas and one receive antenna (single antenna interference cancellation) are distinguished. Several receive antennas can be also utilized for increasing the robustness against fading, applying diversity combination techniques. In the case of the availability of several transmit antennas only, additional space-time coding has to be used for realization of diversity gains. These aspects are also discussed in depth. Furthermore, an introduction to code-division multiple access (CDMA) transmission is given and it is shown how CDMA is applied in the UMTS system. The lecture is concluded by an introduction to digital transmission in the Long Term Evolution (LTE) system.
6	**Lernziele und Kompetenzen**	The students • describe basic equalization algorithms such as decision-feedback equalization (DFE) and maximum-likelihood sequence estimation (MLSE), • apply equalization algorithms to the GSM / Enhanced Data Rates for GSM Evolution (EDGE) mobile communication system, • formulate channel estimation methods for mobile communication systems, • characterize the interference problem in GSM / EDGE, - design interference suppression schemes for GSM/EDGE for receivers with a single antenna (single antenna interference cancellation) and multiple antennas, respectively, • characterize the performance of mobile communication networks for different reception schemes,

		• devise receivers for the realization of diversity gains for multiple receive antennas, • design space-time coding schemes for the realization of diversity gains for multiple transmit antennas, • describe transmission schemes which are based on code-division multiple access (CDMA), • apply reception techniques for CDMA to the UMTS system, • characterize the uplink transmission in the Long Term Evolution (LTE) system, • develop receivers for LTE. Die Studierenden • beschreiben grundlegende Entzerrverfahren wie entscheidungsrückgekoppelte Entzerrung (Decision-Feedback Equalization, DFE) und Maximum-Likelihood-Sequenzschätzung (Maximum-Likelihood Sequence Estimation, MLSE), • wenden Entzerrverfahren auf das GSM/EDGE (Enhanced Data Rates for GSM Evolution) Mobilfunksystem an, • formulieren Kanalschätzverfahren für Mobilfunksysteme, • charakterisieren das Interferenzproblem bei GSM/EDGE, • entwerfen Interferenzunterdrückungsverfahren für GSM/EDGE für Empfänger mit einer Antenne (Single Antenna Interference Cancellation) und mehreren Antennen, • bewerten die Leistungsfähigkeit von Mobilfunknetzen bei Einsatz verschiedener Empfangsverfahren, • konzipieren Empfänger zur Realisierung von Diversitätsgewinnnen bei empfangsseitiger Antennendiversität • entwerfen Space-Time-Codierverfahren zur Realisierung von Diversitätsgewinnen bei sendeseitiger Antennendiversität, • beschreiben auf Code-Division Multiple Access (CDMA) basierende Übertragungsverfahren, • wenden Empfangsverfahren für CDMA auf das UMTS-System an, • charakterisieren die Aufwärtsstrecke von Long Term Evolution (LTE), • entwerfen Empfänger für LTE.
7	**Voraussetzungen für die Teilnahme**	Systemtheorie, Nachrichtenübertragung
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich

11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Lecture notes

Wahlpflichtmodul aus INF im Schwerpunkt Networks and Digital Communication

1	**Modulbezeichnung** 869547	**Advanced Networking LEx** (Advanced networking LEx)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Advanced Networking (2 SWS) Übung: Advanced Networking Exercises (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Dr.-Ing. Kai-Steffen Hielscher Dr.-Ing. Peter Bazan	

4	**Modulverantwortliche/r**	Dr.-Ing. Kai-Steffen Hielscher
5	**Inhalt**	Für die durch Big Data, Cloud Computing, Internet-of-Things (IoT) und mobile Endgeräte hervorgerufenen Herausforderungen sind neue Architekturen für Rechnernetze entstanden: Software-Defined-Networking (SDN) entkoppelt die Data Plane (Weiterleitung von Paketen, auf handelsüblicher Hardware) und die Control Plane (Steuerung, auf leistungsfähigen Plattformen) und bietet offene Programmierschnittstellen; Network Functions Virtualization (NFV) erweitert Konzepte zur Server- und Netzwerkvirtualisierung, so dass bisher auf proprietärer Hardware ausgeführte Netzwerkfunktionen (wie z.B. Routing) ebenfalls virtualisiert und auf handelsüblicher Hardware ausgeführt werden können. Die Vorlesung stellt hinter diesen Technologien stehende Konzepte und Standards vor und zeigt, wie sie für Rechenzentren, für Cloud- und Fog-Computing und für IoT-Anwendungen eingesetzt werden können. Content: New architectures for computer networks have emerged to meet the challenges posed by Big Data, Cloud Computing, Internet-of-Things (IoT) and mobile devices: Software-Defined-Networking (SDN) decouples the data plane (forwarding of packets, on commercially available hardware) and the control plane (control, on powerful platforms) and offers open programming interfaces; Network Functions Virtualization (NFV) extends concepts for server and network virtualisation, so that network functions (such as routing) previously executed on proprietary hardware can also be virtualised and executed on commercially available hardware. The lecture introduces concepts and standards behind these technologies and shows how they can be used for data centres, for cloud and fog computing and for IoT applications.
6	**Lernziele und Kompetenzen**	Fachkompetenz Verstehen Die Studierenden erlangen Verständnis der grundlegenden Konzepte von • Software Defined Networking • Network Function Virtualization

		• Internet of Things • Cloud Computing. Anwenden Die Studierenden wenden die Erkenntisse in Übungsaufgaben an. Erschaffen Die Studierenden erstellen eigene Laborkofigurationen zu • Software Defined Networking • Internet of Things. Competences: Professional competence Understanding Students will gain an understanding of the basic concepts of • Software Defined Networking • Network Function Virtualization • Internet of Things • Cloud Computing. Apply Students apply the knowledge gained in exercises. Create The students create their own laboratory configurations on • Software Defined Networking • Internet of Things.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20191 Wahlpflichtmodul aus INF im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio

11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	William Stallings: Foundations of Modern Networking - SDN, QoE, IoT, and Cloud; Pearson

1	**Modulbezeichnung** 93172	**AI-enabled Wireless Networks** (keine englischsprachige Modulbezeichnung hinterlegt!)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: AI-enabled wireless networks (2 SWS) Übung: Literature review on the application of machine learning in wireless networks (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Dr.-Ing. Mehdi Harounabadi	

4	**Modulverantwortliche/r**	Dr.-Ing. Mehdi Harounabadi
5	**Inhalt**	This course introduces machine learning algorithms such as supervised, unsupervised, reinforcement, deep, and federated learning and their application in the next generation wireless and mobile networks. Different ML use cases are explained which solve problems in different layers of the protocol stack from the physical layer to the application layer. The course includes the following topics: 1.Introduction to machine learning algorithms 2.Python programming language and its ML tools 3.AI-enabled wireless and mobile networks 3.1Cellular networks and ML use cases 3.1.1History of 2G to 4G, 5G and 6G vision 3.1.2ML use cases in physical, MAC and higher layers 3.25G-V2X (cellular-V2X) and ML use cases 3.2.1Sidelink communication as the key enabler 3.2.25G-V2X features and use cases 3.2.3ML use cases in 5G-V2X 3.3Intelligent wireless networks 3.3.1Cognitive radio networks 3.3.2ML use case in wireless networks 4.Standardization activities on AI-enabled wireless networks 4.1.13GPP and 5GAA 4.1.2ETSI Zero touch networks

		Exercises: Literature review on the application of machine learning in wireless networks The exercise of this course includes a literature review research project where students work individually on a relevant topic. The steps to accomplish the research project are as follows: A.Select a topic relevant to the application of ML in wireless networks and register it by email B.Search for the relevant papers and make a list of papers C.Study the papers and prepare a summary D.Present the outcomes Each student should present her/his research study in an intermediate and a final presentation. A summary paper should be written following the "survey papers guideline" using IEEE format. The grade of the research project will be considered as a "Bonus point" (up to 20%) for the final grade.
6	**Lernziele und Kompetenzen**	The students will be able to gain the following competencies after the successful completion of the course: • Have knowledge on machine learning algorithms, current and the next generation wireless and mobile networks and their use cases • To know how to develop machine learning algorithms in Python • Know the most important problems in wireless and mobile networks which can be solved by machine learning algorithms • Have an insight on the exiting work relevant to the topic of the course • Learn the procedure of a scientific research study and publication of the outcome
7	**Voraussetzungen für die Teilnahme**	*Motivation:* Rapid growth in the number of connected wireless nodes such as mobile phones, low power IoT devices, connected vehicles, etc. will expand the scale of the next generation of wireless and mobile networks. Moreover, the foreseen use cases like connected autonomous vehicles, smart homes and cities, ultra-fast and reliable industrial wireless networks, etc. will require ultra-low latency and highly reliable communication. Existing and traditional algorithms are not feasible for the optimization and management of such networks to fulfill the requirements of the emerging use cases due to their high complexity, high dynamicity, and the massive amount of the generated data by connected devices. Recently, artificial intelligence (AI) is planned to be utilized as a new paradigm for the

		design, development and optimization of the next generation wireless and mobile networks. Machine learning (ML) as a subset of AI will be applied to develop intelligent wireless nodes and infrastructures to address the demands of future use cases.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot der Technischen Fakultät oder der Naturwissenschaftlichen Fakultät Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich (30 Minuten)
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Dahlman, Erik, Stefan Parkvall, and Johan Skold. 5G NR: The next generation wireless access technology. Academic Press, 2020. Sun, Yaohua, et al. "Application of machine learning in wireless networks: Key techniques and open issues." IEEE Communications Surveys & Tutorials 21.4 (2019): 3072-3108. Harounabadi, Mehdi, et al. "V2X in 3GPP Standardization: NR Sidelink in Release-16 and Beyond." IEEE Communications Standards Magazine 5.1 (2021): 12-21. Xie, Junfeng, et al. "A survey of machine learning techniques applied to software defined networking (SDN): Research issues and challenges." IEEE Communications Surveys & Tutorials 21.1 (2018): 393-430.

1	**Modulbezeichnung** 965820	**Approximate Computing** (Approximate computing)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Exercises to Approximate Computing (0 SWS) Vorlesung: Approximate Computing (2 SWS)	- 5 ECTS
3	Lehrende	Chetana Pradhan Pierre-Louis Sixdenier Prof. Dr.-Ing. Jürgen Teich Prof. Dr. Oliver Keszöcze	

4	**Modulverantwortliche/r**	Joachim Falk Prof. Dr. Oliver Keszöcze Prof. Dr.-Ing. Jürgen Teich
5	**Inhalt**	Approximate Computing denotes a quite young research area that exploits the fact and capability of many applications and systems to tolerate imprecision and/or inexactness of computed results. Prominent areas of applications and novel techniques of computing approximate rather than exact results have brought up new implementations either at hardware and/or software levels for important emergent workloads such as searching, mining, image processing, and data retrieval. Although hardware technology is improving at a fast pace, energy and power are becoming more and more important constraints apart from exactly computing results in an acceptable amount of time. The main goals of approximate computing techniques are therefore to exploit the possible trade-off between power/energy consumption, accuracy, performance, and/or cost, e.g., utilized hardware resources. The purpose of the course approximate computing is to instruct students about the main ideas and concepts of approximate computing. This includes analyzing the trade-off between energy consumption, accuracy, run-time and hardware costs, concrete approximating techniques (e.g. approximate hardware synthesis, approximating algorithms) as well as theoretical background (determining the computational error and its complexity).
6	**Lernziele und Kompetenzen**	Fachkompetenz - Wissen • The students know the principles and benefits of Approximate Computing and when it is applicable. • The students know multiple error metrics and their semantic meaning. Fachkompetenz - Verstehen • The students understand the difference between the error metrics. • The students understand the principle of function falsification. • The students can apply the presented approximation techniques.

		Fachkompetenz - Anwenden • The students are capable of choosing the appropriate approximation technique based on given requirements.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20191 Wahlpflichtmodul aus INF im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Weitere Informationen: https://www.cs12.tf.fau.de/lehre/lehrveranstaltungen/vorlesungen/approximate-computing

1	**Modulbezeichnung** 901895	**Deep Learning** (Deep learning)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Deep Learning (2 SWS) Übung: Deep Learning Exercises (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr.-Ing. Andreas Maier Leonhard Rist Zijin Yang	

4	**Modulverantwortliche/r**	Felix Denzinger Prof. Dr.-Ing. Andreas Maier Fabian Wagner
5	**Inhalt**	Deep Learning (DL) has attracted much interest in a wide range of applications such as image recognition, speech recognition and artificial intelligence, both from academia and industry. This lecture introduces the core elements of neural networks and deep learning, it comprises: • (multilayer) perceptron, backpropagation, fully connected neural networks • loss functions and optimization strategies • convolutional neural networks (CNNs) • activation functions • regularization strategies • common practices for training and evaluating neural networks • visualization of networks and results • common architectures, such as LeNet, Alexnet, VGG, GoogleNet • recurrent neural networks (RNN, TBPTT, LSTM, GRU) • deep reinforcement learning • unsupervised learning (autoencoder, RBM, DBM, VAE) • generative adversarial networks (GANs) • weakly supervised learning • applications of deep learning (segmentation, object detection, speech recognition, ...) The accompanying exercises will provide a deeper understanding of the workings and architecture of neural networks.
6	**Lernziele und Kompetenzen**	The students • explain the different neural network components, • compare and analyze methods for optimization and regularization of neural networks, • compare and analyze different CNN architectures, • explain deep learning techniques for unsupervised / semi-supervised and weakly supervised learning, • explain deep reinforcement learning, • explain different deep learning applications, • implement the presented methods in Python,

		• autonomously design deep learning techniques and prototypically implement them, • effectively investigate raw data, intermediate results and results of Deep Learning techniques on a computer, • autonomously supplement the mathematical foundations of the presented methods by self-guided study of the literature, • discuss the social impact of applications of deep learning applications.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20191 Wahlpflichtmodul aus INF im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	• Ian Goodfellow, Yoshua Bengio, Aaron Courville: Deep Learning. MIT Press, 2016. • Christopher Bishop: Pattern Recognition and Machine Learning, Springer Verlag, Heidelberg, 2006 • Yann LeCun, Yoshua Bengio, Geoffrey Hinton: Deep learning. Nature 521, 436444 (28 May 2015)

1	**Modulbezeichnung** 93015	**Einführung in die moderne Kryptographie** (Introduction to modern cryptography)	**7,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Einführung in die moderne Kryptographie (4 SWS) Übung: Einführung in die moderne Kryptographie (Übung) (2 SWS)	5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr. Dominique Schröder	

4	**Modulverantwortliche/r**	Carina Harrius Prof. Dr. Dominique Schröder
5	**Inhalt**	This course gives a comprehensive introduction to modern cryptography. The course also serves as a base for other courses on cryptography that are offered by the chair. The topics covered are the following: • Information theoretic security • Computational security • Private key Encryption • Message Authentication Codes • Hash functions • Public key Encryption • Digital Signatures More advanced topics may be covered if time permits.
6	**Lernziele und Kompetenzen**	On successfully passing the course, the student is guaranteed to be knowledgeable on the basic concepts of provable security.
7	**Voraussetzungen für die Teilnahme**	No previous knowledge in Cryptography or computer Security is required.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Variabel
11	**Berechnung der Modulnote**	Variabel (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 165 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Introduction to Modern Cryptography Jonathan Katz and Yehuda Lindell 2nd Edition (2014) (Chapman & Hall/CRC Cryptography and Network Security Series)

		ISBN-13: 978-1466570269

1	**Modulbezeichnung** 44130	**Pattern Recognition** (Pattern recognition)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Pattern Recognition Exercises (1 SWS) Vorlesung: Pattern Recognition (3 SWS)	1,25 ECTS 3,75 ECTS
3	Lehrende	Siming Bayer Paul Stöwer Prof. Dr.-Ing. Andreas Maier	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Andreas Maier
5	**Inhalt**	Mathematical foundations of machine learning based on the following classification methods: • Bayesian classifier • Logistic Regression • Naive Bayes classifier • Discriminant Analysis • norms and norm dependent linear regression • Rosenblatt's Perceptron • unconstraint and constraint optimization • Support Vector Machines (SVM) • kernel methods • Expectation Maximization (EM) Algorithm and Gaussian Mixture Models (GMMs) • Independent Component Analysis (ICA) • Model Assessment • AdaBoost Mathematische Grundlagen der maschinellen Klassifikation am Beispiel folgender Klassifikatoren: • Bayes-Klassifikator • Logistische Regression • Naiver Bayes-Klassifikator • Diskriminanzanalyse • Normen und normabhängige Regression • Rosenblatts Perzeptron • Optimierung ohne und mit Nebenbedingungen • Support Vector Maschines (SVM) • Kernelmethoden • Expectation Maximization (EM)-Algorithmus und Gaußsche Mischverteilungen (GMMs) • Analyse durch unabhängige Komponenten • Modellbewertung • AdaBoost
6	**Lernziele und Kompetenzen**	Die Studierenden • verstehen die Struktur von Systemen zur maschinellen Klassifikation einfacher Muster • erläutern die mathematischen Grundlagen ausgewählter maschineller Klassifikatoren

		• wenden Klassifikatoren zur Lösung konkreter Klassifikationsproblem an • beurteilen unterschiedliche Klassifikatoren in Bezug auf ihre Eignung • verstehen in der Programmiersprache Python geschriebene Lösungen von Klassifikationsproblemen und Implementierungen von Klassifikatoren Students • understand the structure of machine learning systems for simple patterns • explain the mathematical foundations of selected machine learning techniques • apply classification techniques in order to solve given classification tasks • evaluate various classifiers with respect to their suitability to solve the given problem • understand solutions of classification problems and implementations of classifiers written in the programming language Python
7	**Voraussetzungen für die Teilnahme**	• Well grounded in probability calculus, linear algebra/matrix calculus • The attendance of our bachelor course 'Introduction to Pattern Recognition' is not required but certainly helpful. • Gute Kenntnisse in Wahrscheinlichkeitsrechnung und Linearer Algebra/Matrizenrechnung • Der Besuch der Bachelor-Vorlesung 'Introduction to Pattern Recognition' ist zwar keine Voraussetzung, aber sicherlich von Vorteil.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Pflichtmodule Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch oder Englisch
16	**Literaturhinweise**	• Richard O. Duda, Peter E. Hart, David G. Stock: Pattern Classification, 2nd edition, John Wiley&Sons, New York, 2001

		• Trevor Hastie, Robert Tobshirani, Jerome Friedman: The Elements of Statistical Learning - Data Mining, Inference, and Prediction, 2nd edition, Springer, New York, 2009 • Christopher M. Bishop: Pattern Recognition and Machine Learning, Springer, New York, 2006

1	**Modulbezeichnung** 43960	**Self-Organized Networks** (keine englischsprachige Modulbezeichnung hinterlegt!)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Self-organized networks (Ex) (2 SWS) Vorlesung: Self-organized networks (2 SWS)	- -
3	Lehrende	Dr.-Ing. Mehdi Harounabadi	

4	**Modulverantwortliche/r**	Dr.-Ing. Mehdi Harounabadi
5	**Inhalt**	keine Inhaltsbeschreibung hinterlegt!
6	**Lernziele und Kompetenzen**	keine Beschreibung der Lernziele und Kompetenzen hinterlegt!
7	**Voraussetzungen für die Teilnahme**	Self-organized networks (2,5 ECTS) Self-organization refers to processes in physical and biological systems which form patterns by interactions of internal parts of the system and/or their interactions with the environment without any form of external interventions. The resulting organization of systems in this case is fully decentralized and distributed over their parts. There are numerous examples of self-organizing systems e.g. flock of birds, neural learning, evolutionary computation, swarm of robots, and self-organizing networks. The principles of self-organization are applied in different communication technologies such as wired and wireless networks. Transport Control Protocol (TCP), Carrier Sense Multiple Access (CSMA), Adaptive Modulation and Coding (AMC) are some examples of self-organized protocols. Besides, Self-Organizing Network (SON) has been defined by 3GPP as the standardization organization of cellular networks. SON refers to the functionality of cellular systems (e.g. 4G and 5G) which enables the system to perform self-configuration, self-optimization and self-healing. Artificial intelligence and mostly machine learning algorithms play an important role in the implementation of SON. This course aims to give you an insight to the self-organization systems and specially self-organizing networks and illustrate the importance of AI and ML in their implementation. The content of this course includes: • o Self-organizing features and protocols in communication networks o In wired networks o In wireless networks § IEEE 802.11 § Ad hoc and sensor networks § UAV networks o Self-Organizing Networks (SON) in 4G and 5G § Self-configuration § Self-optimization § Self-healing Methods for implementation of self-organizing systems o Bio-inspired methods o Artificial intelligence The final exam is in a written form where students should answer to questions based on the lectures, slides and their understating and analysis from the content of the course.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot der Technischen Fakultät oder der Naturwissenschaftlichen Fakultät Master of Science Information and Communication Technology 20191

		Wahlpflichtmodul aus INF im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot der Technischen Fakultät oder der Naturwissenschaftlichen Fakultät Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Variabel
11	**Berechnung der Modulnote**	Variabel (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Literature review on self-organizing networks (2,5 ECTS) The exercise of this course includes a literature review research project where students work individually on a relevant topic. The steps to accomplish the research project are as follows: A. Select a topic relevant to self-organizing networks and register it by email B. Search for the relevant papers and make a list of papers C. Study the papers and prepare a summary D. Present the outcomes

1	**Modulbezeichnung** 97090	**Simulation und Modellierung I** (Simulation and modelling I)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Exercises to Simulation and Modeling 1 (2 SWS) Vorlesung: Simulation and Modeling 1 (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Lisa Maile Jonathan Fellerer Prof. Dr. Reinhard German	

4	**Modulverantwortliche/r**	Prof. Dr. Reinhard German
5	**Inhalt**	Das Modul vermittelt die Grundlagen der diskreten Ereignissimulation und beinhaltet • diskrete Simulation • analytische Modellierung (z.B. Warteschlangen) • Eingabemodellierung (z.B. Fitting-Verfahren) • Zufallszahlenerzeugung • statistische Ausgabeanalyse • Modellierungsparadigmen (u.a. Ereignis-/Prozessorientierung, Warteschlangen, Automaten, Petri-Netze, UML, graphische Bausteine) • kontinuierliche und hybride Simulation • Simulationssoftware • Fallstudien Content: Overview of the various kinds of simulation • discrete simulation (computational concepts, simulation of queuing systems, simulation in Java, professional simulation tools) • required probability concepts and statistics, modeling paradigms (e.g., event/process oriented, queuing systems, Petri nets, UML statecharts) • input modeling (selecting input probability distributions) • random number generation (linear congruential generators and variants, generating random variates) • output analysis (warm-up period detection, independent replications, result presentation) • continuous and hybrid simulation (differential equations, numerical solution, hybrid statecharts) • simulation software, case studies, parallel and distributed simulation.
6	**Lernziele und Kompetenzen**	Die Studierenden • erwerben Kenntnisse über Verfahren und Realisierungsmöglichkeiten der diskreten Simulation mit Ausblick auf andere Simulationsarten

		• erwerben Kenntnisse über statistische Aspekte der Simulation, die für die Anwendung wichtig sind • wenden statistische Methoden zur Analyse und Bewertung von Eingabe- sowie Ausgabedaten an • erwerben praktische Erfahrung mit kommerziellen Simulationswerkzeugen • erwerben Erfahrungen bei der Simulation in verschiedenen Anwendungsbereichen (u.a. Rechnernetze, Fertigungssysteme, Materialflusssysteme) • entwickeln eigenständig anhand von Beispielaufgaben Simulationsmodelle unter Verwendung verschiedener Modellierungsparadigmen • können in Gruppen kooperativ und verantwortlich arbeiten Learning targets and competences: Students gain knowledge about methods and realization possibilities of discrete simulation with an outlook on other types of simulation gain knowledge of statistical aspects of simulation that are important for practice apply statistical methods for analysis and evaluation of input and output data gain hands-on experience with commercial simulation tools gain experience in simulation in various fields of application (including computer networks, manufacturing systems, material flow systems) independently develop simulation models on the basis of sample tasks using different modeling paradigms can work in groups cooperatively and responsibly
7	**Voraussetzungen für die Teilnahme**	elementare Programmierkenntnisse, vorzugsweise in Java, Mathematikkenntnisse in Analysis, wie z.B. im 1. Semester der angewandten Mathematik vermittelt Recommended background knowledge: basic programming skills, preferably in Java, mathematics skills in analysis, such as taught in the first semester in applied mathematics.
8	**Einpassung in Studienverlaufsplan**	Semester: 1

9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Law, "Simulation Modeling and Analysis, 5th ed., McGraw Hill, 2014

1	**Modulbezeichnung** 593320	**Vernetzte Mobilität und autonomes Fahren** (Connected mobility and autonoumous driving)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Connected Mobility and Autonomous Driving (Exercises) (2 SWS) Vorlesung: Connected Mobility and Autonomous Driving (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Dr.-Ing. Anatoli Djanatliev	

4	**Modulverantwortliche/r**	Dr.-Ing. Anatoli Djanatliev
5	**Inhalt**	Es ist inzwischen allgemein bekannt, dass Fahrzeuge der Zukunft hochgradig vernetzt sein werden. Der aktuelle Trend geht in Richtung des autonomen Fahrens. In den bisheringen Betrachtungen wurde insbesondere die ad-hoc Kommunikation zwischen Fahrzeugen auf unteren Schichten untersucht (Fahrzeugkommunikation). Im Rahmen der vernetzten Mobilität soll das Fahrzeug vor allem als Teil eines größeren Ökosystems mit weiteren Teilnehmern (z.B. Personen, Radfahrern, Ampeln, Gebäuden etc.) gesehen werden. All dies gibt die Möglichkeit den ständig wachsenden Bedarf an Mobilität zu optimieren und neue Sicherheits- und Komfortdienstleistungen zu schaffen. Dies erfordert jedoch die Lösung einiger komplexer Herausforderungen. Neben den gesellschaftlichen und rechtlichen Aspekten müssen insbesondere auch technische Voraussetzungen geschaffen werden. Dazu gehören u.a. geeignete Kommunikationetechnologien (v.a. ad-hoc, Mobilfunk) und Kommunikationsarchitekturen (Cloud-, Edge/Fog-, Node-Computing). Neben Technologien, Methoden und innovativen Mobilitätsdienstleitungen werden im Rahmen dieser Lehrveranstaltung auch grundlegende Aspekte der Verkehrsplanung und Verkehrstechnik eingeführt sowie der intermodale Verkehr besprochen.
6	**Lernziele und Kompetenzen**	Fachkompetenz Verstehen Verständnis grundlegender Konzepte u.a. von • Ad-Hoc Kommunikation • Mobilfunkkommunikation • Verkehrsplanung • Architekturen • Fahrzeug als Teil eines Mobilitäts-Ökosystems • Innovative Dienste Anwenden Bearbeitung von Übungsaufgaben Analysieren

		• Unterschiede zwischen unterschiedlichen Kommunikationstechnologien und Architekturen aufdecken • Relevante Zukunftsszenarien aufbauen Evaluieren (Beurteilen) Anwendung von Simulation und Modellierung zur Evaluierung zukünftiger Szenarien und Fallstudien.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Barbara Flügge; Smart Mobility - Connecting Everyone: Trends, Concepts and Best Practices; Vieweg Teubner, 2017 Maurer, M., Gerdes, J.C., Lenz, B., Winner, H. (Hrsg.); Autonomes Fahren: Technische, rechtliche und gesellschaftliche Aspekte; Springer 2015 Johanning, V., Mildner, R.; Car IT kompakt: Das Auto der Zukunft Vernetzt und autonom fahren; Springer, 2015

Pflichtmodule

1	**Modulbezeichnung** 44130	**Pattern Recognition** (Pattern recognition)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Pattern Recognition Exercises (1 SWS) Vorlesung: Pattern Recognition (3 SWS)	1,25 ECTS 3,75 ECTS
3	Lehrende	Siming Bayer Paul Stöwer Prof. Dr.-Ing. Andreas Maier	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Andreas Maier
5	**Inhalt**	Mathematical foundations of machine learning based on the following classification methods: • Bayesian classifier • Logistic Regression • Naive Bayes classifier • Discriminant Analysis • norms and norm dependent linear regression • Rosenblatt's Perceptron • unconstraint and constraint optimization • Support Vector Machines (SVM) • kernel methods • Expectation Maximization (EM) Algorithm and Gaussian Mixture Models (GMMs) • Independent Component Analysis (ICA) • Model Assessment • AdaBoost Mathematische Grundlagen der maschinellen Klassifikation am Beispiel folgender Klassifikatoren: • Bayes-Klassifikator • Logistische Regression • Naiver Bayes-Klassifikator • Diskriminanzanalyse • Normen und normabhängige Regression • Rosenblatts Perzeptron • Optimierung ohne und mit Nebenbedingungen • Support Vector Maschines (SVM) • Kernelmethoden • Expectation Maximization (EM)-Algorithmus und Gaußsche Mischverteilungen (GMMs) • Analyse durch unabhängige Komponenten • Modellbewertung • AdaBoost
6	**Lernziele und Kompetenzen**	Die Studierenden • verstehen die Struktur von Systemen zur maschinellen Klassifikation einfacher Muster • erläutern die mathematischen Grundlagen ausgewählter maschineller Klassifikatoren

		• wenden Klassifikatoren zur Lösung konkreter Klassifikationsproblem an • beurteilen unterschiedliche Klassifikatoren in Bezug auf ihre Eignung • verstehen in der Programmiersprache Python geschriebene Lösungen von Klassifikationsproblemen und Implementierungen von Klassifikatoren Students • understand the structure of machine learning systems for simple patterns • explain the mathematical foundations of selected machine learning techniques • apply classification techniques in order to solve given classification tasks • evaluate various classifiers with respect to their suitability to solve the given problem • understand solutions of classification problems and implementations of classifiers written in the programming language Python
7	**Voraussetzungen für die Teilnahme**	• Well grounded in probability calculus, linear algebra/matrix calculus • The attendance of our bachelor course 'Introduction to Pattern Recognition' is not required but certainly helpful. • Gute Kenntnisse in Wahrscheinlichkeitsrechnung und Linearer Algebra/Matrizenrechnung • Der Besuch der Bachelor-Vorlesung 'Introduction to Pattern Recognition' ist zwar keine Voraussetzung, aber sicherlich von Vorteil.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Pflichtmodule Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch oder Englisch
16	**Literaturhinweise**	• Richard O. Duda, Peter E. Hart, David G. Stock: Pattern Classification, 2nd edition, John Wiley&Sons, New York, 2001

		• Trevor Hastie, Robert Tobshirani, Jerome Friedman: The Elements of Statistical Learning - Data Mining, Inference, and Prediction, 2nd edition, Springer, New York, 2009 • Christopher M. Bishop: Pattern Recognition and Machine Learning, Springer, New York, 2006

1	**Modulbezeichnung** 96430	**Statistical Signal Processing** (Statistical signal processing)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Ergänzungen und Übungen zur statistischen Signalverarbeitung (1 SWS) Vorlesung: Statistische Signalverarbeitung (3 SWS)	- 5 ECTS
3	Lehrende	Thomas Haubner Prof. Dr.-Ing. Walter Kellermann	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Walter Kellermann
5	**Inhalt**	The course concentrates on fundamental methods of statistical signal processing and their applications. The main topics are: *Discrete-time stochastic processes in the time and frequency domain* Random variables (RVs), probability distributions and densities, expectations of random variables, transformation of RVs, vectors of normally distributed RVs, time-discrete random processes: probability distribution and densities, expectation, stationarity, cyclostationarity, ergodicity, correlation functions and correlation matrices, spectral representations, principal component analysis (PCA), Karhunen-Loève transform (KLT). *Estimation theory* estimation criteria, prediction, classical and Bayesian parameter estimation (including MMSE, Maximum Likelihood, and Maximum A Posteriori estimation), Cramer-Rao bound *Linear signal models* Parametric models (cepstral decomposition, Paley-Wiener theorem, spectral flatness), non-parametric models (all-pole, all-zero and pole-zero models, lattice structures, Yule-Walker equations, PARCOR coefficients, cepstral representation) *Signal estimation* Supervised estimation, problem classes, orthogonality principle, MMSE estimation, linear MMSE estimation for normally distributed random processes, optimum FIR filtering, optimum linear filtering for stationary processes, prediction and smoothing, Kalman filters, optimum multichannel filtering (Wiener filter, LCMV, MVDR, GSC) *Adaptive filtering* Gradient methods, LMS, NLMS, APA and RLS algorithms and their convergence behavior

		Zeitdiskrete Zufallsprozesse im Zeit- und Frequenzbereich Zufallsvariablen (ZVn), Wahrscheinlichkeitsverteilungen und dichten, Erwartungswerte; Transformation von ZVn; Vektoren normalverteilter ZVn; zeitdiskrete Zufallsprozesse (ZPe): Wahrscheinlichkeitsverteilungen und dichten, Erwartungswerte, Stationarität, Zyklostationarität, Ergodizität, Korrelationsfunktionen und -matrizen, Spektraldarstellungen; Principal Component Analysis, Karhunen-Loeve Transformation; *Schätztheorie* Schätzkriterien; Prädiktion; klassische und Bayessche Parameterschätzung (inkl. MMSE, Maximum Likelihood, Maximum A Posteriori); Cramer-Rao-Schranke *Lineare Signalmodelle* Parametrische Modelle (Cepstrale Zerlegung, Paley-Wiener Theorem, Spektrale Glattheit); Nichtparametrische Modelle: Allpole-/Allzero-/ Pole-zero-(AR/MA/ARMA) Modelle; Lattice-Strukturen, Yule-Walker Gleichungen, PARCOR-Koeffizienten, Cepstraldarstellungen; *Signalschätzung* Überwachte Signalschätzung, Problemklassen; Orthogonalitätsprinzip, MMSE-Schätzung, lineare MMSE-Schätzung für Gaußprozesse; Optimale FIR-Filter; Lineare Optimalfilter für stationäre Prozesse; Prädiktion und Glättung; Kalman-Filter; optimale Multikanalfilterung (Wiener-Filter, LCMV, MVDR, GSC); *Adaptive Filterung* Gradientenverfahren; LMS-, NLMS-, APA- und RLS-Algorithmus und Ihr Konvergenzverhalten.
6	**Lernziele und Kompetenzen**	The students: • analyze the statistical properties of random variables, random vectors, and stochastic processes by probability density functions and expectations as well as correlation functions and matrices and their frequency-domain representations • know the Gaussian distribution and its role to describe the properties of random variables, vectors and processes • understand the differences between classical and Bayesian estimation, derive and analyze MMSE and ML estimators for specific estimation problems, especially for signal estimation • analyze and evaluate optimum linear MMSE estimators (single- and multichannel Wiener filter and Kalman filter) for direct and inverse supervised estimation problems

		• evaluate adaptive filters for the identification of optimum linear estimators. Die Studierenden • analysieren die statistischen Eigenschaften von Zufallsvariablen, -vektoren und stochastischen Prozessen mittels Wahrscheinlichkeitsdichten und Erwartungswerten, bzw. Korrelationsfunktionen, Korrelationsmatrizen und deren Frequenzbereichsdarstellungen • kennen die spezielle Rolle der Gaußverteilung und ihre Auswirkungen auf die Eigenschaften von Zufallsvariablen, -vektoren und Prozessen • verstehen die Unterschiede klassischer und Bayesscher Schätzung, entwerfen und analysieren MMSE- und ML-Schätzer für spezielle Schätzprobleme, insbesondere zur Signalschätzung • analysieren und evaluieren lineare MMSE-optimale Schätzer (ein- und vielkanalige Wiener-Filter und Kalman-Filter) für direkte und inverse überwachte Schätzprobleme; • evaluieren adaptive Filter zur Identifikation optimaler linearer Signalschätzer
7	**Voraussetzungen für die Teilnahme**	Module Signale und Systeme I und Signale und Systeme II, Digitale Signalverarbeitung oder gleichwertige
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Pflichtmodule Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	A. Papoulis, S. Pillai: Probability, Random Variables and Stochastic Processes; McGraw-Hill, 2002 (englisch) D. Manolakis, V. Ingle, S. Kogon: Statistical and Adaptive Signal Processing; Artech House, 2005 (englisch)

Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems

1	**Modulbezeichnung** 151664	**Advanced Communication Networks** (Advanced communication networks)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Advanced Communication Networks - Tutorial (0 SWS) Vorlesung: Advanced Communication Networks (4 SWS)	- 5 ECTS
3	Lehrende	Walid Ghanem Prof. Dr. Laura Cottatellucci	

4	**Modulverantwortliche/r**	Prof. Dr. Laura Cottatellucci
5	**Inhalt**	Telecommunications have become ubiquitous in daily life and wireless networks play a fundamental role thanks to their capability to support mobility. In a wireless communication, the concept of link does not exist. Users radiate energy and communicate through the superposition of each others transmissions which creates interference. Compared to wireline networks this scenario is extremely challenging but also offers unpredictable opportunities in the development of new technologies (massive MIMO, cognitive radio, etc.) and exploitation of new features, e.g., opportunistic communications and multiuser diversity. The exponentially increasing request of higher and higher throughput is satisfied densifying users and access points per unit area and allowing more and more interference while adopting advanced techniques and innovative resource allocation to mitigate the detrimental effects of interference. Objective of this course is to introduce the student to advanced techniques for coordinated medium access control and radio resource management in cellular systems. Power allocation, rate adaptation and scheduling will be discussed both in centralized and distributed settings. Some mathematical methods play a fundamental role in resource allocation, namely, classical Perron-Frobenius theory for nonnegative matrices, convex and nonconvex constrained optimization, distributed optimization and game theory. The course introduces the student to such methods and exemplifies their application to various resource allocation problems. Additionally, the course addresses relevant aspects of resource allocation in wireless networks such as fairness and cross-layer design. *Technical Content* • Properties and challenges of the wireless medium. • Basic concepts of communication networks: the layered architecture. • Evolution of wireless cellular network architectures: From Global System for Mobile to Advanced-Long Term Evolution. • Multiple Access Schemes: CSMA variants, TDMA, FDMA, CDMA, OFDMA, SC-FDMA, SDMA. • Uplink-downlink duality. • Opportunistic scheduling and multiuser diversity.

		• Advanced concepts: small cells and heterogeneous networks, relaying and cooperation, network coding, cognitive radio networks. • Basics of resource allocation: power allocation, rate adaptation, and scheduling. • Classical resource allocation techniques: Centralized and distributed power control based on the Perron-Frobenius theorem. • Fundamentals of convex constrained optimization and application to resource allocation. • Resource allocation and fairness. • Fundamentals of nonconvex optimization and relaxation techniques. • Applications of nonconvex optimization to resource allocation. • Fundamentals of distributed optimization and applications to resource allocation. • Fundamental concepts of game theory. • Resource contention via game theoretical methods.
6	**Lernziele und Kompetenzen**	The student • Describes and/or recognizes wireless channel models. • Criticizes the limits of a layered architecture in wireless systems. • Defends the use of cross-layer design in wireless network. • Appraises and compares the distribution of functionalities in network entities for different architectures. • Argue on the pros and contras of different multiple access schemes according to various criteria (e.g. spectral efficiency, power efficiency, robustness to interference). • Compares and contrasts micro-diversity and various macro-diversity schemes. • Computes the total rate of SDMA with various receivers. • Relates the multiple access in uplink to broadcasting in downlink and justifies the concept of uplink-downlink duality. • Uses uplink-downlink duality to design a precoder and allocate power. • Contrasts multiple access in uplink and broadcasting in downlink in terms of channel state acquisition both for TDD and FDD transmission. • Uses multiuser diversity for opportunistic scheduling. • Compares multiuser diversity for users having identical and different channel statistics. • Contrasts opportunistic scheduling in terms of channel state acquisition and feedback both for uplink and downlink and for both FDD and TDD transmission schemes. • Appraises the impact of multiple antennas on opportunistic scheduling. • Analyses different settings with interference in small cells and designs countermeasures. • Categorizes relaying schemes in LTE.

		• Analyses performance of relaying schemes. • Argues on possible improvements of relaying schemes via network coding and physical layer network coding. • Uses the Perron-Frobenious theorem to allocate power in a centralized manner. • Judges the feasibility of a power control problems and formulates alternative approaches in case of unfeasibility. • Uses the Perron-Frobenious theorem to design a distributed power control scheme. • Judges the convergences of distributed power control based on the Perron-Frobenius theorem and appraises the robustness of asynchronous power control. • Applies techniques of convex optimization to discriminate convex problems and determine necessary and/or sufficient conditions for global optimality. • Judges the applicability of KKT conditions and duality. • Uses KKT conditions to solve convex optimization problems. • Uses duality to solve convex optimization problems. • Applies convex optimization to resource allocation in wireless communications. • Compares different definitions of fairness and applies them to rate allocation. • Appraises the effect of channel knowledge at the transmitter on different fairness criteria. • Applies KKT conditions for opportunistic user scheduling. • Describes a proportional fair algorithm for opportunistic scheduling. • Applies relaxation to nonconvex quadratic constrained quadratic programming. • Formulates resource allocation problems as constrained optimization programming. • Contrasts various distributed optimization methods. • Applies the concept of best response to determine Nash equilibria. • Argues about existence and uniqueness of Nash equilibria. • Assesses if a given game is a potential game and solves it. • Defends the concept of Pareto optimality in resource allocation. • Contrasts the concepts of pure and mixed strategies in game theory. • Uses coupled constrained concave game to allocate powers in heterogeneous networks.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222

10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 947709	**Auditory Models** (Auditory models)	**2,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Auditory Models (2 SWS)	2,5 ECTS
3	Lehrende	Prof. Dr.-Ing. Bernd Edler	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Bernd Edler
5	**Inhalt**	• Main components of the human auditory system • Common models • Mechanical models • Physiological models • Psychoacoustic models • Applications (hearing aids, audio coding, . . .)
6	**Lernziele und Kompetenzen**	Goals • Students understand the structure and function of the human auditory system • Students gain deeper insight into psychoacoustic phenomena, such as masking, directional and spatial hearing • Students implement and evaluate perceptual models for various applications • Students collaborate with scientists in the fields of audiology and neuroscience
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 96875	**Ausgewählte Kapitel der Audiodatenreduktion** (keine englischsprachige Modulbezeichnung hinterlegt!)	**2,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Ausgewählte Kapitel der Audiodatenreduktion (2 SWS)	2,5 ECTS
3	Lehrende	Prof. Dr.-Ing. Jürgen Herre	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Jürgen Herre
5	**Inhalt**	Perceptual audio coding is ubiquitous in modern life (mp3 players, mobile phones, DVD players, computers, ...) Based on related classes (esp. Speech and Audio Processing"), this lecture aims at deepening the understanding of modern algorithms for perceptual source coding of audio. It includes an overview of the most relevant standardized coders, starting with MPEG-1 (incl. mp3) via MPEG-4 all the way to the most recent MPEG Audio standard. The significant algorithms are discussed and new approaches are described. The selected topics include: Efficient coding of several audio channels / parametric multi- channel coding Typical coding artifacts; subjective and objective quality assessment Scalable audio coding Bandwidth extension Semi-parametric audio coding Low-delay audio coding The lecture includes a number of demonstrations and audio examples to illustrate the discussed algorithms.
6	**Lernziele und Kompetenzen**	• Wissen - Die Studenten kennen die Hauptkomponenten eines gehörangepassten Audiocodecs, sowie die wichtigsten Algorithmen, Codierstrategien und Bewertungsmethoden. Weiterhin kennen sie die Terminologie und gängige Abkürzungen aus diesem Kontext. • Verstehen - Die Studenten verstehen, wie Designentscheidungen in Audiocodecs die letztendlich erreichte Audioqualität beeinflussen, verstehen die gebräuchlichsten Tools aus dem Bereich der gehörangepasste Audiocodierung und wie verschiedene Anwendungsszenarien das Coderdesign bestimmen. • Anwenden - Die Studenten können übliche mathematische Analysemethoden verwenden, um einfache Coder-Componenten zu beschreiben und gegebenenfalls zu modifizieren.

		• Analysieren - Die Studenten können Audiocodierungs-Standards und wahrnehmungsbasierte Messwerkzeuge dazu analysieren um die zugrundeliegenden Konzepte und Anforderungen zu erfassen. • Evaluieren (Beurteilen) - Die Studenten können Audiocodierungs-Standards und wahrnehmungsbasierte Messwerkzeuge evaluieren um zu beurteilen, welcher Standard bzw. welches Messwerkzeug das passendste ist für einen bestimmten Anwendungsfall. • Synthese - Die Studenten können eine Liste von Anforderungen und Bewertungskriterien für Audiocodecs zusammenstellen für gewünschte Anwendungsfälle. • Lern- bzw. Methodenkompetenz - Die Studenten hinterfragen bestehende Ansätze hinsichtlich ihrer Anwendbarkeit in der Praxis.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 96312	**Bild-, Video- und mehrdimensionale Signalverarbeitung** (Image, video and multidimensional signal processing)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Bild-, Video- und mehrdimensionale Signalverarbeitung (2 SWS)	5 ECTS
3	Lehrende	Prof. Dr.-Ing. Andre Kaup	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Andre Kaup
5	**Inhalt**	*Punktoperationen* Histogrammausgleich, Gamma-Korrektur *Binäroperationen* Morphologische Filter, Erosion, Dilatation, Opening, Closing *Farbräume* Trichromat, RGB- Farbraum, HSV-Farbraum *Mehrdimensionale Signale und Systeme* Theorie mehrdimensionaler Signale und Systeme, Impulsantwort, lineare Bildfilterung, Leistungsspektrum, Wiener Filter *Interpolation von Bildsignalen* Bilineare Interpolation, Bicubische Interpolation, Spline Interpolation *Merkmalsdetektion in Bildern* Bildmerkmale, Kantendetektion, Hough Transformation, Harris Ecken Detektor, Texturmerkmale, Grauwertematrix *Skalierungsraumdarstellung* LoG, DoG, SIFT, SURF *Bildabgleich* Projektive Abbildungen, Blockabgleich, Optischer Fluss, Merkmalsbasierter Abgleich mittels SIFT und SURF, RANSAC *Bildsegmentierung* Amplituden Schwellenwertermittlung, K-Means Clustering, Bayes Klassifikation, Regionen-basierte Segmentierung, kombinierte Segmentierung und Bewegungsschätzung, zeitliche Segmentierung von Videos

Bildverarbeitung im Transformationsbereich

Unitäre Transformation, Karhunen-Loeve Transformation, separable Transformationen, Haar und Hadamard Transformation, DFT, DCT

Content:

Point operations

Histogram equalization, gamma correction

Binary operations

Morphological filters, erosion, dilation, opening, closing

Color spaces

Trichromacy, red-green-blue color spaces, color representation using hue, saturation and value of intensity

Multidimensional signals and systems

Theory of multidimensional signals and systems, impulse response, linear image filtering, power spectrum, Wiener filtering

Interpolation of image signals

Bi-linear interpolation, bi-cubic interpolation, spline interpolation

Image feature detection

Image features, edge detection, Hough transform, Harris corner detector, texture features, co-occurrence matrix

Scale space representation

Laplacian of Gaussian, difference of Gaussian, scale invariant feature transform, speeded-up robust feature transform

Image matching

Projective transforms, block matching, optical flow, feature-based matching using SIFT and SURF, random sample consensus algorithm

Image segmentation

		Amplitude thresholding, k-means clustering, Bayes classification, region-based segmentation, combined segmentation and motion estimation, temporal segmentation of video *Transform domain image processing* Unitary transform, Karhunen-Loeve transform, separable transform, Haar and Hadamard transform, DFT, DCT
6	**Lernziele und Kompetenzen**	Die Studierenden • verstehen Punktoperationen an Bilddaten und Gamma-Korrektur • testen die Wirkung von Rangordnungs- und Medianfiltern an Bilddaten • unterscheiden und bewerten verschiedene Farbräume für Bilddaten • erklären das Prinzip der zwei-dimensionalen linearen Filterung für Bildsignale • berechnen und bewerten die zweidimensionale diskrete Fourier-Transformierte eines Bildsignales • bestimmen vergrößerte diskrete Bildsignale mit Methoden der bilinearen und Spline-Interpolation • überprüfen Bilddaten auf ausgewählte Textur-, Kanten- und Bewegungsmerkmale • analysieren Bild- und Videodaten auf Merkmale in unterschiedlichen Scale-Spaces • erläutern und beurteilen Methoden für das Matching von Bilddaten • segmentieren Bilddaten durch Programmierung von einfachen Klassifikations- oder Clustering-Verfahren • verstehen das Prinzip von Transformation auf Bilddaten und können diese an Beispielen anwenden. The students • understand point operations for image data and gamma correction • test the effects of rank order and median filters for image data • evaluate and differentiate between different color spaces for image data • explain the principle of two-dimensional linear filtering for image signals • calculate and evaluate the two-dimensional discrete Fourier transform of an image signal • determine enlarged discrete image signals by bi-linear and spline interpolation • verify image data for selected texture, edge and motion features

		• analyze image and video data for features in different scale spaces • explain and evaluate methods for the matching of image data • segment image data by implementing basic classification and clustering methods • understand the principle of transformations on image data and apply them exemplarily
7	**Voraussetzungen für die Teilnahme**	Vorlesung Signale und Systeme I und II
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot der Technischen Fakultät oder der Naturwissenschaftlichen Fakultät Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	J.-R. Ohm: |Multimedia Content Analysis|, Springer, 2016 J. W. Woods: |Multidimensional Signal, Image, and Video Processing and Coding|, Academic Press, 2^nd^ edition, 2012

1	**Modulbezeichnung** 96090	**Digitale elektronische Systeme** (Digitale elektronische Systeme)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Digitale elektronische Systeme (3 SWS) Übung: Übungen zu Digitale elektronische Systeme (1 SWS)	5 ECTS -
3	Lehrende	Prof. Dr.-Ing. Robert Weigel Albert-Marcel Schrotz	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Robert Weigel
5	**Inhalt**	• Analog-Digital-Umsetzer: Qualitätsmerkmale, Messtechnik, Hardwarearchitekturen • Digital-Analog-Umsetzer: Qualitätsmerkmale, Messtechnik, Hardwarearchitekturen • Programmierbare Logikschaltungen (PLD, FPGA): Grundlegende Konzepte, Kategorien, Hardwarearchitekturen • Digitale-Filter: Theorie, Eigenschaften, Entwicklung und Implementierung und IIR und FIR Filtern
6	**Lernziele und Kompetenzen**	• Die Studierenden verstehen die Hardwarearchitekturen und Funktionsweisen von Komponenten digitaler Elektronischer Systeme wie Digital-Analog-Umsetzer, Analog-Digital Umsetzer, PLDs und FPGAs und können diese erläutern • Die Studierenden Verstehen die Qualitätsmerkmale von Digitalen Elektronischen Komponenten, können diese auf konkrete Komponenten anwenden und somit die Qualität von digitalen Elektronischen Komponenten anhand der in Datenblättern typischer weise gegebenen Qualitätsmerkmale evaluieren • Die Studierenden können die Einflüsse von nichtidealen Bauelementen auf digitale elektronische Systeme analysieren • Die Studierenden verstehen die Funktion, die Eigenschaften, die Entwicklungsmethodik sowie die Implementierung von digitalen Filtern und könne diese erläutern
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	Unregelmäßig
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester

15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 93601	**Information Theory and Coding / Informationstheorie und Codierung** (Information theory and coding)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Tutorial for Information Theory and Coding (1 SWS) Vorlesung: Information Theory and Coding (3 SWS)	- 5 ECTS
3	Lehrende	Dr.-Ing. Ali Bereyhi Prof. Dr.-Ing. Ralf Müller	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Ralf Müller
5	**Inhalt**	1. Introduction: binomial distribution, (7,4)-Hamming code, parity-check matrix, generator matrix 2. Probability, entropy, and inference: entropy, conditional probability, Bayes law, likelihood, Jensens inequality 3. Inference: inverse probability, statistical inference 4. The source coding theorem: information content, typical sequences, Chebychev inequality, law of large numbers 5. Symbol codes: unique decidability, expected codeword length, prefix-free codes, Kraft inequality, Huffman coding 6. Stream codes: arithmetic coding, Lempel-Ziv coding, Burrows-Wheeler transform 7. Dependent random variables: mutual information, data processing lemma 8. Communication over a noisy channel: discrete memory-less channel, channel coding theorem, channel capacity 9. The noisy-channel coding theorem: jointly-typical sequences, proof of the channel coding theorem, proof of converse, symmetric channels 10. Error-correcting codes and real channels: AWGN channel, multivariate Gaussian pdf, capacity of AWGN channel 11. Binary codes: minimum distance, perfect codes, why perfect codes are bad, why distance isnt everything 12. Message passing: distributed counting, path counting, low-cost path, min-sum (=Viterbi) algorithm 13. Exact marginalization in graphs: factor graphs, sum-product algorithm

14. Low-density parity-check codes: density evolution, check node degree, regular vs. irregular codes, girth

15. Lossy source coding: transform coding and JPEG compression

--

1. Einleitung: Binomialverteilung, (7,4)-Hamming-Code, Paritätsmatrix, Generatormatrix

2. Wahrscheinlichkeit, Entropie und Inferenz: Entropie, bedingte Wahrscheinlichkeit, Bayessches Gesetz, Likelihood, Jensensche Ungleichung

3. Inferenz: Inverse Wahrscheinlichkeit, statistische Inferenz

4. Das Quellencodierungstheorem: Informationsgehalt, typische Folgen, Tschebyschevsche Ungleichung, Gesetz der großen Zahlen

5. Symbolcodes: eindeutige Dekodierbarkeit, mittlere Codewortlänge, präfixfreie Codes, Kraftsche Ungleichung, Huffmancodierung

6. Stromcodes: arithmetische Codierung, Lempel-Ziv-Codierung, Burrows-Wheeler-Transformation

7. Abhängige Zufallsvariablen: Transinformation, Datenverarbeitungslemma

8. Kommunikation over gestörte Kanäle: diskreter gedächtnisloser Kanal, Kanalcodierungstheorem, Kanalkapazität

9. Das Kanalcodierungstheorem: verbundtypische Folgen, Beweis des Kanalcodierungstheorems, Beweis des Umkehrsatzes, symmetrische Kanaäle

10. Fehlerkorrigierende Codes und reale Kanäle: AWGN-Kanal, mehrdimensionale Gaußsche WDF, Kapazität des AWGN-Kanals

11. Binäre Codes: Minimaldistanz, perfekte Codes, Warum perfekte Codes schlecht sind, Warum Distanz nicht alles ist

12. Nachrichtenaustausch: verteiltes Zählen, Pfadzählen, günstigster Pfad, Minimumsummenalgorithmus

13. Exakte Marginalisierung in Graphen: Faktorgraph, Summenproduktalgorithmus

14. LDPC-Codes: Dichteevolution, Knotenordnung, reguläre und irreguläre Codes, Graphumfang

		15. Verlustbehaftete Quellencodierung: Transformationscodierung und JPEG-Kompression
6	**Lernziele und Kompetenzen**	The students apply Bayesian inference to problems in both communications and everyday's life. The students explain the concept of digital communications by means of source compression and forward-error correction coding. For the design of communication systems, they use the concepts of entropy and channel capacity. They calculate these quantities for memoryless sources and channels. The students proof both the source coding and the channel coding theorem. The students compare various methods of source coding with respect to compression rate and complexity. The students apply source compression methods to measure mutual information. The students factorize multivariate functions, represent them by graphs, and marginalize them with respect to various variables. The students explain the design of error-correcting codes and the role of minimum distance. They decode error-correcting codes by means of maximum-likelihood decoding and message passing. The students apply distributed algorithms to problems in both communications and everydays life. The students improve the properties of low-density parity-check codes by widening the girth and/or irregularity in the degree distribution. The students transform source images into the frequency domain to improve lossy compression. -- Die Studierenden wenden Bayessche Inferenz auf Probleme in der Nachrichtentechnik und im Alltagsleben an. Die Studierenden erklären die konzeptuelle Trennung von digitaler Übertragung in Quellen- und Kanalcodierung.

		Kommunikationssysteme entwerfen sie unter Betrachtung von Entropie und Kanalkapazität. Sie berechnen diese Größen für gedächtnislose Quellen und Kanäle. Die Studierenden beweisen sowohl das Quellen- als auch das Kanalcodierungstheorem. Die Studierenden vergleichen verschiedenartige Quellencodierungsverfahren hinsichtlich Komplexität und Kompressionsrate. Die Studierenden verwenden Quellencodierverfahren zur Messung von Transinformation. Die Studierenden faktorisieren Funktionen mehrerer Veränderlicher, stellen diese als Graph dar und marginalisieren sie bezüglich mehrerer Veränderlicher. Die Studierenden erklären den Entwurf von Kanalcodes und den Einfluss der Minimaldistanz. Sie decodieren Kanalcodes gemäß maximaler Likelihood und Nachrichtenaustausch. Die Studierenden wenden verteilte Algorithmen auf Probleme der Nachrichtentechnik und des Alltagslebens an. Die Studierenden verbessern die Eigenschaften von LDPC-Codes durch Erhöhung des Umfangs und/oder durch irreguäre Knotenordnungsverteilungen. Die Studierenden transformieren Bildquellen zur Verbesserung verlustbehafteter Kompression iin den Frequenzbereich.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 2
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h

14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	MacKay, D.: Information Theory, Inference, and Learning Algorithms, Cambridge University Press, Cambridge, 2003.

1	**Modulbezeichnung** 96101	**Integrierte Navigationssysteme** (keine englischsprachige Modulbezeichnung hinterlegt!)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übung Integrierte Navigationssysteme (1 SWS) Vorlesung: Integrierte Navigationssysteme (3 SWS)	- 5 ECTS
3	Lehrende	Prof. Dr. Jörn Thielecke	

4	**Modulverantwortliche/r**	Prof. Dr. Jörn Thielecke
5	**Inhalt**	**1. Überblick** • Von der Astronavigation zur Navigation mit Mikroelektronik • Messprinzipien & Positionsberechnung (Standlinien/-flächen) • Begriffsdefinitionen (s. US Federal Radionavigation Plan), Genauigkeit, Verfügbarkeit, Verlässlichkeit, Integrität, etc. • Systematische Strukturierung des Gebiets: siehe 2. bis 7. **2. Positions- und Lagebestimmung** • Funkausbreitung und Funkortung (Beispiel WLAN) • Fingerabdruckverfahren • Lokalisierung mit Markovketten **3. Koppelnavigation (Tracking) mittels Trägheitsnavigation** • Koordinatensysteme und ihre Einsatzgebiete • Mathematische Gundlagen, z.B. Quaternionen, Corioliseffekt • Strapdown Inertial Navigation Systems • Sensorprinzipien und Trägheitssensoren • Computergestützte Lösung der Navigationsgleichungen • System- und Fehlermodellierung im Zustandsraum • Das Kalmanfilter und Glättung mittels Retrodiktion **4. Seiteninformationen: Kinematik und Karten (kurze Übersicht)** **5. Landmarken als lokaler Ortsbezug** • Merkmalsbasierte Ortung z.B. mit Kamera oder UWB • Partikelfilter und Monte-Carlo-Integration **6. Integration von Navigationskomponenten: Sensordatenfusion** • Fusionsarchitekturen: Beispiel GPS & Trägheitsnavigation **7. Einbettung von Navigationssystemen** • Assisted GPS oder Location Based Service Anmerkung: Die Navigationsmethoden werden gleichermaßen anhand von Tafel- und Rechnerübungen (MATLAB) einstudiert
6	**Lernziele und Kompetenzen**	1. Die Studierenden werden in die Lage versetzt, typische Navigationsverfahren hinsichtlich ihrer Funktionsweise und Einsetzbarkeit zu analysieren, zu bewerten und weiterzuentwickeln.

		2. Die Studierenden lernen Navigationsgleichungen selbst aufzustellen, anzuwenden und mit unterschiedlichen Algorithmen auf dem Computer zu lösen. 3. Die Studierenden entwickeln ein Verständnis für die Herausforderungen bei der Integration unterschiedlicher Teilsysteme zu einem Navigationssystem und der Einbettung von Navigationssystemen in übergeordnete Systeme
7	**Voraussetzungen für die Teilnahme**	Keine formalen Voraussetzungen, geeignet für Masterstudium, grundlegende Kenntnisse erforderlich in: linearer Algebra, Physik, Signal- & System¬theorie, Wahrscheinlichkeitstheorie.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%) Bei bestandener Prüfung wird die Note um eine Teilnotenstufe (z.B. von 2,0 auf 1,7) verbessert, wenn Sie mindestens 75% der Hausaufgaben einschließlich der Rechnerübungen erfolgreich absolviert haben. Eine Note besser als 1,0 wird nicht vergeben.
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Skriptum zur Lehrveranstaltung.

1	**Modulbezeichnung** 96270	**Kanalcodierung** (Channel coding)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung mit Übung: Kanalcodierung (3 SWS) Vorlesung mit Übung: Channel Coding (4 SWS) Übung: Übungen zur Kanalcodierung (1 SWS)	5 ECTS 5 ECTS -
3	Lehrende	Dr.-Ing. Clemens Stierstorfer	

4	**Modulverantwortliche/r**	Dr.-Ing. Clemens Stierstorfer
5	**Inhalt**	1) Introduction and Motivation 2) Fundamentals of Block Coding 3) Introduction to Finite Fields I 4) Linear Block Codes 5) Linear Cyclic Codes 6) Introduction to Finite Fields II 7) BCH and RS Codes 8) Convolutional Codes 9) Codes with Iterative Decoding
6	**Lernziele und Kompetenzen**	Das Modul Kanalcodierung umfasst eine Einführung in die Grundlagen der algebraischen, fehlerkorrigierenden Blockcodes sowie einen Einstieg in die Thematik der Faltungscodes. Iterativ decodierte Codeschemata wie Turbo-Codes und LDPC-Codes werden ebenfalls eingeführt. Im Einzelnen sind die Inhalte oben aufgeführt. Die Studierenden definieren die Problematik der Kanalcodierung, grenzen sie von anderen Codierverfahren (z.B. der Quellencodierung) ab und kennzeichnen die unterschiedlichen Ansätze zur Fehlerkorrektur und -erkennung. Sie nennen Beispiele für Einsatzgebiete von Kanalcodierung und geben einen Überblick über die historische Entwicklung des Fachgebiets. Die Studierenden erstellen Übertragungsszenarien für den Einsatz von Kanalcodierung bestehend aus Sender, Übertragungskanal und Empfänger und beachten dabei die Grundannahmen beim Einsatz von Blockcodes bzw. der Modellierung der Kanäle. Sie formulieren mathematische Beschreibungen der Encodierung sowie der optimalen Decodierung bzw. suboptimaler Varianten. Die Studierenden beherrschen die Grundlagen fehlerkorrigierender linearer Blockcodes, beschreiben diese mathematisch korrekt mittels Vektoren und Matrizen über endlichen Körpern und implementieren und bewerten zugehörige Encoder- und Decoderstrukturen insbesondere Syndromdecoder. Dabei modifizieren sie Generatormatrizen, ermitteln Prüfmatrizen und erstellen Syndromtabellen. Sie schätzen die minimale Hammingdistanz von Codes mittels Schranken ab und können den erzielbaren Codegewinn erläutern. Sie kennen und benutzen beispielhaften Codefamilien (z.B. Hamming-Codes, Simplex-Codes, Reed-Muller-Codes).

Die Studierenden erkennen die Vorteile zyklischer linearer Blockcodes und beschreiben diese mit Polynomen über endlichen Körpern. Sie nutzen die Restklassenrechnung bzgl. Polynomen zur Umsetzung systematischer Encoder und zur Realisierung von Syndromdecodern mittels Schieberegisterschaltungen. Sie kennen beispielhafte Codefamilien.

Die Studierenden nutzen Primkörper, Erweiterungskörper, Minimalpolynome und Kreisteilungsklassen sowie die Spektraldarstellung über endlichen Körpern zur Realisierung von BCH- und Reed-Solomon-Codes gemäß der BCH-Schranke. Sie verstehen die Grundlagen der Decodierung von BCH- und Reed-Solomon-Codes. Sie skizzieren und erläutern die Kanalcodierkonzepte von CD und DVD.

Die Studierenden erklären die Unterschiede von Faltungscodes und Blockcodes, skizzieren anhand von tabellierten Generatorpolynomen zugehörige Encoder und erläutern diese. Sie erklären die Funktionsweise des optimalen Decoders (MLSE) und demonstrieren diese beispielhaft.

Die Studierenden verstehen die Grundlagen der iterativen Decodierung, insbesondere wenden sie die Grundlagen des Information Combining zur Kombination von verschiedenen Beobachtungen an. Sie verstehen die Bedeutung von Log-Likelihood-Ratios bei iterativen Decodieruvorgängen und berechnen diese. Sie skizzieren die grundlegenden Encoder- und Decoderstrukturen von Turbo-Codes und die Grundzüge der Codierung mit LDPC-Codes u.a. der Decodierung mittels Belief Propagation.

Die Vorlesung erfolgt wechselweise auf Deutsch oder Englisch (Winter/ Sommer). Die zur Verfügung gestellten Unterlagen sind ausschließlich in Englisch gehalten. Die Studierenden verwenden entweder die englischen Fachtermini sicher oder kennen diese und drücken sich sicher mit den entsprechenden deutschen Fachbegriffen aus.

Die Umsetzung der angegebenen Algorithmen in eine Programmiersprache (C, Matlab usw.) sollten die Studierenden zu diesem Zeitpunkt des Studiums üblicherweise beherrschen. Übungen hierzu bleiben der Eigeninitiative überlassen.

Students define the problems of channel coding, how to distinguish it from other coding methods (such as source coding) and how to describe the various different approaches to error correction and detection. They are able to list example application areas of channel coding and give an overview of the historical development of the field.

		Furthermore, they describe and analyze transmission scenarios for the application of channel coding which consist of transmitter, transmission channel and receiver, taking into account the general assumptions for applying block codes or modeling the channels. They formulate mathematical descriptions of encoding, optimal decoding and sub-optimal methods. Students illustrate the principles of error-correcting linear block codes and describe them mathematically using vectors and matrices over finite fields. They implement and analyze corresponding encoder and decoder structures, in particular syndrome decoders, and modify generator matrices, construct test matrices and create syndrome tables. They estimate the minimum Hamming distance of codes using bounds and are able to explain the coding gain that can be achieved in individual cases. They analyze and use example code families (e.g. Hamming codes, simplex codes, Reed-Muller codes). Students explain the advantages of cyclic linear block codes and how to describe them with polynomials over finite fields. They apply polynomial modular arithmetic to implement systematic encoders and realize syndrome decoders using shift register circuits. They know and use exemplary code families. Students use prime fields, extension fields, minimal polynomials and cyclotomic cosets, and spectral representation over finite fields to implement BCH and Reed-Solomon codes using the BCH bound. They understand the foundations of decoding BCH and Reed-Solomon codes and how to sketch and explain the channel coding concepts of CDs and DVDs. Students are able to describe the differences between convolutional codes and block codes, to sketch the respective encoders based on tabulated generator polynomials and to explain them. They are able to explain how optimal decoders (MLSE) work using examples. Students sketch the foundations of iterative decoding. In particular, they apply methods of information combining to combine different observations. They use and calculate log-likelihood ratios in iterative decoding processes, sketch the basic encoding and decoding structures of turbo codes and the basics of coding using LDPC codes (including decoding using belief propagation). Students either are able to use the English technical terms correctly or know them and are able to express themselves using the respective technical terms in German.
7	**Voraussetzungen für die Teilnahme**	Es ist hilfreich, wenn die Studierenden die erlernten Algorithmen in eine Programmiersprache (C, Matlab usw.) umsetzen können.

		It would be very helpful if the participants can implement the specified algorithms into a programming language (C, Matlab, etc.).
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich Hilfsblatt, Taschenrechner: Sie können ein einzelnes A4-Blatt (Vorder- und Rückseite oder andere Blätter mit offensichtlich identischer Gesamtfläche) verwenden, um Ihre eigene, handschriftliche Formelsammlung aufzuschreiben. Sie können einen nicht programmierbaren Taschenrechner verwenden. Cheat Sheet, Calculator: A single A4 sheet (front and back, or any other collection of sheets with an obviously identical total area size) can be used to write down your own handwritten collection of formulas, etc. You may also bring a non-programmable calculator.
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch oder Englisch
16	**Literaturhinweise**	• J. Huber, R. Fischer, C. Stierstorfer: Folien zur Vorlesung • M. Bossert: Kanalcodierung, Oldenbourg Wissenschaftsverlag, 3. Auflage, 2013 • M. Bossert: Channel Coding for Telecommunications, John Wiley & Sons, 1999 • B. Friedrichs: Kanalcodierung, Springer Verlag, 1996 • S.B. Wicker: Error Control Systems for Digital Communications and Storage, Prentice-Hall, 1995

1	**Modulbezeichnung** 668129	**Machine Learning in Communications** (Machine learning in communications)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Machine Learning in Communications (4 SWS) Übung: Tutorial for Machine Learning in Communications (0 SWS)	5 ECTS -
3	Lehrende	Prof. Dr. Laura Cottatellucci Nikita Shanin	

4	**Modulverantwortliche/r**	Prof. Dr. Laura Cottatellucci
5	**Inhalt**	Recently, in many areas of wireless communications such as wireless sensor networks (WSNs), heterogeneous networks and complex ad hoc networks, distributed graph algorithms and machine learning on graphs are gaining relevance as fundamental tools in network analysis and information processing. This motivates to deliver a general introduction to fundamentals of machine learning such as detection of clusters on graphs. The introduction is followed by the application of machine learning to the design of physical and data layer techniques in wireless communications and in the optimization of mobile networks.
6	**Lernziele und Kompetenzen**	The students • know and explain the fundamentals of machine learning with special attention to machine learning over graphs. • apply these principles in the design and optimisation of wireless communications systems and mobile networks.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20191 Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 48440	**Machine Learning in Signal Processing** (Machine learning in signal processing)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Supplements for Machine Learning in Signal Processing (1 SWS) Vorlesung: Machine Learning in Signal Processing (3 SWS)	- 5 ECTS
3	Lehrende	Kamal Gopikrishnan Nambiar Prof. Dr. Vasileios Belagiannis	

4	**Modulverantwortliche/r**	PD Dr.Ing. Jürgen Seiler
5	**Inhalt**	This course is an introduction into machine learning and artificial intelligence. The special emphasis is on applications to modern signal processing problems. The course is focused on design principles of machine learning algorithms. The lectures start with a short introduction, where the nomenclature is defined. After this, probabilistic graphical models are introduced and the use of latent variables is discussed, concluding with a discussion of hidden Markov models and Markov fields. The second part of the course is about deep learning and covers the use of deep neural networks for machine learning tasks. In the last part of the lecture, the use of deep neural networks for speech processing tasks is introduced. The course is based on the materials and video footage from Dr. Roland Maas. He is an outstanding machine learning expert and a former member of the Chair of Multimedia Communications and Signal Processing.
6	**Lernziele und Kompetenzen**	After attending the lecture, students will be able to • understand regression and classification problems • apply PDF estimation algorithms • understand Gaussian mixture models and expectation-maximization • apply principal component analysis and independent component analysis • assess different estimation algorithms • explain the application of machine learning to system identification • apply hidden Markov models • understand different artificial neural network architectures • explain deep learning principles • apply artificial neural networks • devise learning strategies for deep neural networks • assess the application of deep neural networks for speech processing tasks.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1

9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich/mündlich
11	**Berechnung der Modulnote**	schriftlich/mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Literature: • C. M. Bishop: Pattern Recognition and Machine Learning, http://www.research.microsoft.com/en-us/um/people/cmbishop/PRML • S. Theodoridis and K. Koutroumbas: Pattern Recognition • M. Nielsen: Neural Networks and Deep Learning.

1	**Modulbezeichnung** 43141	**Mobile Communications** (Mobile communications)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Mobile Communications - Tutorial (1 SWS) Vorlesung: Mobile Communications (3 SWS)	- 5 ECTS
3	Lehrende	Dr.-Ing. Ali Bereyhi Prof. Dr.-Ing. Ralf Müller	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Ralf Müller
5	**Inhalt**	History of mobile communications, cellular systems, sectorization, spectral efficiency, co-channel interference, adjacent-channel interference, near-far effect, cellular network architecture, antenna types and parameters, free space propagation, reflection, attenuation, diffraction, scattering, classification of channel models, ground reflection model, Okumura-Hata model, shadowing, narrow-band fading, time- variant channels, scattering function, delay-Doppler spectrum, diversity principles, combining methods, diversity gain, multiplexing, duplexing, digital modulation, Gaussian filtered minimum shift keying, basics of channel coding, interleaving, global system for mobile communications, physical versus logical channels, frame structure, call set-up, synchronization, channel estimation, hand-off
6	**Lernziele und Kompetenzen**	The students explain the cellular structure of mobile communication systems. They students explain the physical mechanics of radio wave propagation in the cm-band. The students explain the GSM cellular communications standard. The students discuss the pros and cons of several multiple-access and duplexing methods. The students discuss the pros and cons of several modulation and coding formats. The students decide which antenna type is suitable for a given morphological structure of the environment. The students predict the amplitude and dynamic of the antenuation between a mobile transmitter and a fixed receiver. The students utilize diversity methods to improve the link quality. The students determine the coverage probability of a given cellular communication system. The students collaborate on solving exercise problems. The students discuss which system solutions fit to which environments.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 2
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)

12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Proakis, J.: Digital Communications, McGraw-Hill, 4th ed., 2001. Rappaport, T.: Wireless Communications: Principles & Practice, Prentice Hall, 2nd ed., 2001. Mouly, M., Paulet, M.: The GSM System for Mobile Communications, Cell & SYS, France, 1992. Goldsmith, A.: Wireless Communications, Cambridge Univ. Press, 2005.

1	**Modulbezeichnung** 302148	**Musiksignalverarbeitung - Analyse** (Music processing - Analysis)	**2,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Music Processing Analysis (2 SWS)	2,5 ECTS
3	Lehrende	Prof. Dr. Meinard Müller	

4	**Modulverantwortliche/r**	Prof. Dr. Meinard Müller
5	**Inhalt**	Music signals possess specific acoustic and structural characteristics that are not shared by spoken language or audio signals from other domains. In fact, many music analysis tasks only become feasible by exploiting suitable music-specific assumptions. In this course, we study feature design principles that have been applied to music signals to account for the music-specific aspects. In particular, we discuss various musically expressive feature representations that refer to musical dimensions such as harmony, rhythm, timbre, or melody. Furthermore, we highlight the practical and musical relevance of these feature representations in the context of current music analysis and retrieval tasks. Here, our general goal is to show how the development of music-specific signal processing techniques is of fundamental importance for tackling otherwise infeasible music analysis problems.
6	**Lernziele und Kompetenzen**	Expertise Understand • The students present central tasks in music processing in their own words and outline possible solutions. • The students understand the properties of different forms of representation of music. Apply • The students apply basic algorithms for the analysis and comparison of music signals. • Students can predict how different musical properties will affect the signal analysis. Analyze • The students observe and discuss the meaning and impact of parameters in music analysis. • The students compare different methods of analyzing periodicities. Evaluate • The students question assumptions that are often implicitly made when using analytical methods. • Students estimate when methods might work when analyzing specific music signals and when they typically fail.

Learning and methodological skills

- The students prepare for the lecture using selected literature and Jupyter notebooks.
- The students question existing approaches regarding their applicability in practice.
- The students pay attention to efficiency issues in the algorithms discussed.

Self-competence

- The students question their understanding of what they have learned using exercises.
- The students formulate questions and ask them to the lecturer and the audience in the lecture.

Social skills

- The students independently organize learning groups in which the subject is discussed and deepened.
- The students simulate oral exams with their fellow students.

Fachkompetenz

Verstehen

- Die Studierenden stellen zentrale Aufgabenstellungen der Musikverarbeitung in eigenen Worten dar und skizzieren Lösungsansätze.
- Die Studierenden verstehen die Eigenschaften von unterschiedlichen Darstellungsformen von Musik.

Anwenden

- Die Studierenden wenden grundlegende Algorithmen zur Analyse und zum Vergleich von Musiksignalen an.
- Die Studierenden können voraussagen, wie sich unterschiedliche musikalische Eigenschaften bei der Signalanalyse auswirken.

Analysieren

- Die Studierenden beobachten und diskutieren die Bedeutung und Auswirkung von Parametern bei der Musikanalyse.
- Die Studierenden stellen unterschiedliche Verfahren bei der Analyse von Periodizitäten gegenüber.

Evaluieren (Beurteilen)

		• Die Studierenden hinterfragen Annahmen, die implizit bei der Verwendung von Analysemethoden gemacht werden. • Die Studierenden schätzen ein, wann Methoden bei der Analyse von gewissen Musiksignalen funktionieren könnten und wann sie typischerweise versagen. Lern- bzw. Methodenkompetenz • Die Studierenden bereiten sich auf die Vorlesung anhand ausgewählter Literatur vor. • Die Studierenden hinterfragen bestehende Ansätze hinsichtlich ihrer Anwendbarkeit in der Praxis. • Die Studierenden beachten Fragen der Effizienz bei den diskutierten Algorithmen. Selbstkompetenz • Die Studierenden hinterfragen ihr Verständnis von dem Gelernten anhand von Übungsaufgaben. • Die Studierenden formulieren Fragen und stellen diese in der Vorlesung an den Dozenten und die Zuhörerschaft. Sozialkompetenz • Die Studierenden organisieren selbständig Lerngruppen, in denen der Stoff diskutiert und vertieft wird. • Die Studierenden simulieren mit ihren Kommilitonen mündliche Prüfungen.
7	**Voraussetzungen für die Teilnahme**	In this course, we discuss a number of current research problems in music processing or music information retrieval (MIR) covering aspects from information science and digital signal processing. We provide the necessary background information and give numerous motivating examples so that no specialized knowledge is required. However, the students should have a solid mathematical background. The lecture is accompanied by readings from textbooks or the research literature. Furthermore, the students are required to experiment with the presented algorithms using Python.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester

13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	http://www.music-processing.de http://www.springer.com/gp/book/9783319219448

1	**Modulbezeichnung** 502007	**Musiksignalverarbeitung - Synthese** (Music processing - synthesis)	**2,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Musikverarbeitung - Synthese (2 SWS)	-
3	Lehrende	Maximilian Schäfer	

4	**Modulverantwortliche/r**	Maximilian Schäfer
5	**Inhalt**	• Verarbeitung von Audiosignalen durch parametrische Filter und Effekte • Erzeugung von künstlichen Klängen mit Mitteln der digitalen Klangsynthese • Klangwiedergabe in echten und virtuellen Räumen • Klangbeispiele und Demonstrationen • Programmiersprachen für Audio-Echtzeit-Verarbeitung *Content*: • a short history of electrical and electronic music • processing of audio signals by parametric filters and effects • digital sound synthesis • sound reproduction in real and in virtual environments • sound examples and demonstrations • programming languages for audio real-time processing
6	**Lernziele und Kompetenzen**	Die Studierenden • beschreiben die speziellen Anforderungen für Audio-Echtzeit-Verarbeitung, • wenden ihre theoretischen Kenntnisse zeitdiskreter Signale und Systeme für die Verarbeitung und Erzeugung musikalischer Klänge an, • gestalten eigene Software-Realisierungen zur Klangsynthese, • entwerfen technische Systeme für musikalisch motivierte Aufgabenstellungen. The students • specify the special requirements for audio realtime processing, • apply their theoretical knowledge about discrete-time signals and systems to processing and synthesis of musical sounds, • design their own software realizations for sound synthesis • implement technical systems for digital music.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191

		Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Das Vorlesungsskript und weitere Zusatzmaterialien zur Vorlesung werden via StudOn zur Verfügung gestellt.

1	**Modulbezeichnung** 250058	**Signalanalyse** (Signal analysis)	**2,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Signalanalyse (2 SWS)	-
3	Lehrende	Dr.-Ing. Heinrich Löllmann	

4	**Modulverantwortliche/r**	Dr.-Ing. Heinrich Löllmann
5	**Inhalt**	Es werden im Rahmen dieser Vorlesung unterschiedliche Verfahren zur Analyse digitaler Signale, sowie deren Anwendungsmöglichkeiten behandelt. Die folgenden Konzepte werden dabei insbesondere behandelt: • Fourieranalyse von Signalen • Signalanalyse mittels Zeit-Frequenz-Transformationen • Parametrische und nichtparametrische Signalanalyse • Verfahren zur Frequenzschätzung • Räumliche Signalanalyse • Filterbänke und Wavelets. In this course, different approaches for the analysis of digital signals and their applications are treated, which comprises the following topics: • Fourier analysis of signals • Signal analysis by means of time-frequency transformations • Parametric and non-parametric signal analysis • Frequency estimation • Spatial signal analysis • Filter-banks and wavelets.
6	**Lernziele und Kompetenzen**	Die Studierenden • beschreiben, welche Methoden der Signalanalyse für unterschiedlichen Arten von Signalen angewendet werden • beschreiben grundlegende Methoden der spektralen Signalanalyse • erläutern wodurch die spektrale und zeitliche Auflösung bei der Spektralanalyse von Signalen begrenzt wird • beschreiben die Konzepte sowie die Vor- und Nachteile der parametrischen und nichtparametrischen Signalanalyse • erklären unterschiedliche Verfahren der Zeit-Frequenz-Analyse • stellen die Analyse von Signalen mittels Filterbänke und Wavelets dar • können Verfahren zur Frequenzschätzung erläutern • formulieren Verfahren zur Analyse räumlicher Signale. The students • describe which methods for signal analysis can be applied for different types of signals • describe fundamental approaches for spectral signal analysis • explain the limiting factors for the time and frequency resolution for the spectral analysis of signals

		• describe concepts as well as the pros and cons of parametric and non-parametric signal analysis • explain different approaches for time-frequency analysis • describe the analysis of signals by means of filter-banks and wavelets • explain methods for frequency estimation • formulate approaches for spatial signal analysis.
7	**Voraussetzungen für die Teilnahme**	Fundierte Kenntnisse in digitaler Signalverarbeitung. Requirements Solid knowledge in digital signal processing
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich Mündliche Prüfung mit einer Dauer von 30 min. Oral examination of 30 min duration.
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	P. Stoica und R. Moses: "Spectral Analysis of Signals", Pearson Prentice Hall, 2005

1	**Modulbezeichnung** 96460	**Speech and Audio Signal Processing** (Speech and audio signal processing)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übung zur Sprach- und Audiosignalverarbeitung (1 SWS) Vorlesung: Sprach- und Audiosignalverarbeitung (3 SWS)	- 5 ECTS
3	Lehrende	Mhd Modar Halimeh Prof. Dr.-Ing. Walter Kellermann	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Walter Kellermann
5	**Inhalt**	It concentrates on algorithms for speech and audio signal processing with applications in telecommunications and multimedia, especially • physiology and models for human speech production and hearing: source-filter model, filterbank model of the cochlea, masking effects, • representation of speech and audio signals: estimation and representation of short-term and long-term statistics in the time and frequency domain as well as the cepstral domain; typical examples and visualizations • source coding for speech and audio signals: criteria, scalar and vector quantization, linear prediction, prediction of the pitch frequency; waveform coding, parametric coding, hybrid coding, codec standards (ITU, GSM, ISO-MPEG) • basic concepts of automatic speech recognition (ASR): feature extraction, dynamic time warping, Hidden Markov Models (HMMs) • basic concepts of speech synthesis: text-to-speech systems, model-based and data-driven synthesis, PSOLA synthesis system • signal enhancement for acquisition and reproduction: noise reduction, acoustic echo cancellation, dereverberation using single-channel and multichannel algorithms. Es werden Grundlagen und Algorithmen der Verarbeitung von Sprach- und Audiosignalen mit Anwendungen in Telekommunikation und Multimedia behandelt, insbesondere: • Physiologie und Modelle der Spracherzeugung und des Hörens: Quelle-Filter-Modell, Filterbank-Modell der Cochlea; Maskierungseffekte; • Darstellung von Sprach- und Audiosignalen: Schätzung und Darstellung der Kurzzeit- und Langzeitstatistik in Zeit-, Frequenz- und Cepstralbereich; typische Beispiele, Visualisierungen; • Quellencodierung für Sprache und Audiosignale: Kriterien; skalare und vektorielle Codierung; lineare Prädiktion; Pitchprädiktion; Wellenform-/Parameter-/Hybrid-Codierung; Standards (ITU, GSM, ISO-MPEG)

		• Spracherkennung: Merkmalextraktion, Dynamic Time Warping, Hidden Markov Models • Grundprinzipien der Sprachsynthese: Text-to-Speech Systeme, modellbasierte und datenbasierte Synthese, PSOLA-Synthese • Signalverbesserung bei Signalaufnahme und wiedergabe: Geräuschbefreiung, Echokompensation, Enthallung mittels ein- und mehrkanaliger Verfahren;
6	**Lernziele und Kompetenzen**	The students • understand basic physiological mechanisms of human speech production and hearing and can apply them for the analysis of speech and audio signals • apply basic methods for the estimation and representation of the short-term and long-term statistics of speech and audio signals and can analyze such signals by means of these methods • understand current methods for source coding of speech and audio signals and can analyze current coding standards • verstehen die Grundbausteine von Spracherkennungssystemen und können deren Funktion mittels Rechnersimulation analysieren • understand the basic principle of text-to-speech systems and can apply fundamental methods for speech synthesis • can apply basic algorithms for speech enhancement and understand their functionality for real-world data. Die Studierenden • verstehen die grundlegenden physiologischen Mechanismen der Spracherzeugung und des Hörens beim Menschen und können diese zur Analyse von Sprach- und Audiosignalen anwenden • wenden die grundlegenden Methoden zur Schätzung und Darstellung der Kurzzeit- und Langzeitstatistik von Sprach- und Audiosignalen an und können diese damit analysieren • verstehen die aktuellen Methoden zur Quellencodierung von Sprache-und Audiosignalen und können aktuelle Codierstandards analysieren • verstehen die Grundbausteine von Spracherkennungssystemen und können deren Funktion mittels Rechnersimulation analysieren • verstehen die Grundprinzipien von Text-to-Speech Systemen und können elementare Algorithmen zur Sprachsynthese anwenden • können elementare Algorithmen zur Signalverbesserung anwenden und für reale Daten analysieren
7	**Voraussetzungen für die Teilnahme**	Vorlesung Signale und Systeme I & II

8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich (90 Minuten)
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Gemäß themenbezogenen Angaben in der Lehrveranstaltung

1	**Modulbezeichnung** 788996	**Speech Enhancement** (Speech enhancement (oral examination))	**2,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Speech Enhancement (2 SWS)	2,5 ECTS
3	Lehrende	Prof. Dr. Emanuël Habets	

4	**Modulverantwortliche/r**	Prof. Dr. Emanuël Habets
5	**Inhalt**	*Course Description* We live in a noisy world! In all applications related to speech, from hands-free communication to human-machine interfaces, a speech signal of interest captured by one or more microphones is contaminated by noise and reverberation. The quality and intelligibility of the signal of interest depend highly on the level of noise and reverberation. Therefore, it is highly desirable, and sometimes even indispensable, to "clean up" the captured signals before storage, transmission, or reproduction. This course discusses both model-driven and data-driven methods to estimate the signal of interest. It aims to provide a strong foundation for researchers, engineers, and graduate students interested in signal and speech enhancement. *Relation to other courses* This course is the most advanced course offered by the university on this topic, and serves as an excellent basis from which to commence research in the area. Various aspects of the course bring students up to date with the very latest developments in the field, as seen in recent international conferences and journals. This course is well complimented by |Selected Topics in Perceptual Audio Coding| (Prof. Herre) and | Auditory Models| (Prof. Edler).
6	**Lernziele und Kompetenzen**	Students • Formulate the speech enhancement problem mathematically. • Derive optimal single- and multi-channel filters to reduce noise and reverberation. • Evaluate and compare the performance of single- and multi-channel filters for speech enhancement. • Understand how reference signals and other prior information can be used in a speech enhancement system. • Understand the limitations and challenges of existing speech enhancement systems. • Understand the importance of binaural cues and the influence of a speech enhancement system on the binaural cues in the context of hearing aids. • Design a microphone array and analyze its performance. • Design a speech enhancement system for a given acoustic scenario.

		• Evaluate both subjectively and objectively the performance of a speech enhancement system in terms of the speech quality and intelligibility.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch oder Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 96880	**Speech Enhancement** (Speech enhancement)	**2,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Speech Enhancement (2 SWS)	2,5 ECTS
3	Lehrende	Prof. Dr. Emanuël Habets	

4	**Modulverantwortliche/r**	Prof. Dr. Emanuël Habets
5	**Inhalt**	**Description** We live in a noisy world! In all applications related to speech, from hands-free communication to human-machine interfaces, a speech signal of interest captured by one or more microphones is contaminated by noise and reverberation. The quality and intelligibility of the signal of interest depend highly on the level of noise and reverberation. Therefore, it is highly desirable, and sometimes even indispensable, to "clean up" the captured signals before storage, transmission, or reproduction. This course discusses both model-driven and data-driven methods to estimate the signal of interest and aims to provide a strong foundation for researchers, engineers, and graduate students interested in signal and speech enhancement. **Relation to other courses** This course is the most advanced course offered by the university on this topic and serves as an excellent basis from which to commence research in the area. Various aspects of the course bring students up to date with the very latest developments in the field, as seen in recent international conferences and journals. This course is well complimented by |Selected Topics in Perceptual Audio Coding| (Prof. Herre) and | Auditory Models| (Prof. Edler).
6	**Lernziele und Kompetenzen**	• Formulate the speech enhancement problem mathematically. • Derive optimal single- and multi-channel filters to reduce noise and reverberation. • Evaluate and compare the performance of single- and multi-channel filters for speech enhancement. • Understand how reference signals and other prior information can be used in a speech enhancement system. • Understand the limitations and challenges of existing speech enhancement systems. • Understand the importance of binaural cues and the influence of a speech enhancement system on the binaural cues in the context of hearing aids. • Design a microphone array and analyze its performance.

		• Design a speech enhancement system for a given acoustic scenario. • Evaluate subjectively and objectively the performance of a speech enhancement system in terms of speech quality and intelligibility.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch oder Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 498723	**Transformationen in der Signalverarbeitung** (Transforms in signal processing)	**2,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Transformationen in der Signalverarbeitung (2 SWS)	2,5 ECTS
3	Lehrende	PD Dr.Ing. Jürgen Seiler	

4	**Modulverantwortliche/r**	PD Dr.Ing. Jürgen Seiler
5	**Inhalt**	Das Modul "Transformationen in der Signalverarbeitung" behandelt mehrere verschiedene Transformationen, die im Rahmen der Signalverarbeitung Verwendung finden. Dabei werden zuerst die grundlegenden Konzepte von Transformationen diskutiert und die Vorteile die Transformationen mit sich bringen erläutert. Im Anschluss daran werden die grundlegenden Eigenschaften von Integraltransformationen betrachtet und die Laplace- und die Fourier-Transformation im Detail untersucht. Um auch zeitlich veränderliche Signale gut transformieren zu können werden danach die Kurzzeit-Fourier-Transformation und die Gabor-Transformation eingeführt. Im Anschluss daran erfolgt eine Betrachtung der Auswirkung der Abtastung auf transformierte Signale, bevor die z-Transformation als Transformation für diskrete Signale behandelt wird. Abschließend erfolgt die Betrachtung weiterer Transformationen für diskrete Signale wie der Diskreten Fourier-Transformation oder linearer Block-Transformationen. The module "Transforms in Signal Processing" covers several different transforms which are used in the field of signal processing. For this, first the basic concepts of transforms are discussed and the advantages which are offered by the different transforms are presented. Subsequent to this, fundamental properties of integral transforms are considered and the Laplace- and the Fourier-Transform are examined in detail. To be able to transform time-varying signals, the Short-Time Fourier-Transform and the Gabor-Transform are introduced, afterwards. Subsequent to this, the impact of sampling on transformed signals is analyzed before the z-Transform as a transform for discrete signals is covered. Finally, further transforms for discrete signals like the Discrete Fourier-Transform or Linear-Block Transforms are discussed.
6	**Lernziele und Kompetenzen**	Die Studierenden können nach Besuch der Vorlesung • Anwendungsmöglichkeiten von Transformationen bestimmen • Integraltransformationen gegenüberstellen und untersuchen • die Existenz von Transformationen hinterfragen • die Eindeutigkeit von Transformationen überprüfen • Sätze und Eigenschaften von Transformationen entwickeln • zu Transformationen zugehörige inverse Transformationen einschätzen • die Zusammenhänge zwischen verschiedenen Transformationen einschätzen • auf Zusammenhänge zwischen Ausgangssignalen und transformierten Signalen folgern • Symmetriebeziehungen von Transformationen ausarbeiten

		• Zusammenhänge zwischen kontinuierlichen und diskreten Signalen ausarbeiten Educational Objectives and Competences: After attending the lecture, students will be able to • determine applications of transforms • contrast and examine integral transforms • question the existence of transforms • evaluate the uniqueness of transforms • develop theorems and properties of transforms • evaluate to transforms corresponding inverse transforms • evaluate the relationships between different transforms • asses the relationship between original signal and transformed signals • devise the symmetry properties of transforms • devise the relationship between continuous and discrete signals
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	K. Krüger, Transformationen - Grundlagen und Anwendungen in der Nachrichtentechnik, Vieweg Verlag, Braunschweig B. Girod, R. Rabenstein, A. Stenger, Einführung in die Systemtheorie, B. G. Teubner Verlag, Stuttgart

Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems

1	**Modulbezeichnung** 43821	**Computer Graphics** (Computer graphics)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Computer Graphics Basic Tutorials (1 SWS) Vorlesung: Computer Graphics (3 SWS)	1,25 ECTS 3,75 ECTS
3	Lehrende	Jonas Müller Prof. Dr. Marc Stamminger	

4	**Modulverantwortliche/r**	Prof. Dr. Marc Stamminger
5	**Inhalt**	Die Vorlesung gibt eine Einführung in die Computergraphik: • Graphik Pipeline • Clipping • 3D Transformationen • Hierarchische Display Strukturen • Perspektive und Projektionen • Visibilitätsbetrachtungen • Rastergraphik und Scankonvertierung • Farbmodelle • Lokale und globale Beleuchtungsmodelle • Schattierungsverfahren • Ray Tracing und Radiosity • Schatten und Texturen Contents: This lecture covers the following aspects of Computer Graphics: • graphics pipeline • clipping • 3D transformations • hierarchical display structures • perspective transformations and projections • visibility determination • raster graphics and scan conversion • color models • local and global illumination models • shading models • ray tracing and radiosity • shadows and textures
6	**Lernziele und Kompetenzen**	Die Studierenden • geben die unterschiedlichen Schritte der Graphik Pipeline wieder • erklären die Funktionsweise der Clippingalgorithmen für Linien und Polygone • beschreiben, charakterisieren und berechnen affine und perspektivische Transformationen in 3D und veranschaulichen

		die allgemeine Form der Transformationsmatrix in homogener Koordinaten • skizzieren die Verfahren zur Tiefe- und Visibilitätsberechnung • vergleichen die unterschiedlichen Farbmodelle der Computergraphik • illustrieren und untersuchen die Datenstrukturen zur Beschreibung virtueller 3D Modelle und komplexer Szenen • erläutern die Funktionsweise der Rasterisierung und Scankonvertierung in der Graphikpipeline • lösen Aufgaben zu Beleuchtung und Texturierung von 3D virtuellen Modellen • klassifizieren Schattierungsverfahren • bestimmen den Unterschied zwischen lokaler und globaler Beleuchtung und formulieren Algorithmen für Ray Tracing und Radiocity *Educational objectives and skills:* Students should be able to • describe the processing steps in the graphics pipeline • explain clipping algorithms for lines and polygons • explain, characterize and compute affine and perspective transformations in 2D and 3D, and provide an intuitive description of the general form of corresponding transformation matrices in homogeneous coordinates • depict techniques to compute depth, occlusion and visibility • compare the different color models • describe data structures to represent 3D virtual models and complex scenes • explain the algorithms for rasterization and scan conversion • solve problems with shading and texturing of 3D virtual models • classify different shadowing techniques • explain the difference between local and global illumination techniques and formulate algorithms for ray tracing and radiosity
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Übungsleistung Klausur (60 Minuten)
11	**Berechnung der Modulnote**	Übungsleistung (0%) Klausur (100%)
12	**Turnus des Angebots**	nur im Wintersemester

13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• P. Shirley: Fundamentals of Computer Graphics. AK Peters Ltd., 2002 • Hearn, M. P. Baker: Computer Graphics with OpenGLD. Pearson • Foley, van Dam, Feiner, Hughes: Computer Graphics - Principles and Practice • Rauber: Algorithmen der Computergraphik • Bungartz, Griebel, Zenger: Einführung in die Computergraphik • Encarnação, Strasser, Klein: Computer Graphics

1	**Modulbezeichnung** 44150	**Diagnostic Medical Image Processing** (Diagnostic medical image processing)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Medical Image Processing for Diagnostic Applications (VHB-Kurs) (4 SWS)	5 ECTS
3	Lehrende	Manuela Meier Arpitha Ravi Celia Martín Vicario Luis Rivera Monroy	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Andreas Maier
5	**Inhalt**	English version: The contents of the module comprise basics about medical imaging modalities and acquisition hardware. Furthermore, details on acquisition-dependent preprocessing are covered for image intensifiers, flat-panel detectors, and MR. The fundamentals of 3D reconstruction from parallel-beam to cone-beam reconstruction are also covered. In the last chapter, rigid registration for image fusion is explained. Deutsche Version: Die Inhalte des Moduls umfassen Grundlagen der medizinischen Bildverarbeitung und Aufnahmeprinzipien. Darüber hinaus werden Details der Vorverarbeitung für Bildverstärker, Flachpaneldetektoren und MR erklärt. Die Grundlagen der Rekonstruktion von Parallelstrahl bis hin zur Kegelstrahl-Tomographie werden ebenfalls behandelt. Im letzten Kapitel wird starre Registrierung für Bildfusion erläutert.
6	**Lernziele und Kompetenzen**	English Version: The participants • understand the challenges in interdisciplinary work between engineers and medical practitioners. • develop understanding of algorithms and math for diagnostic medical image processing. • learn that creative adaptation of known algorithms to new problems is key for their future career. • develop the ability to adapt algorithms to different problems. • are able to explain algorithms and concepts of the module to other engineers. Deutsche Version: Die Teilnehmenden • verstehen die Herausforderungen in der interdisziplinären Arbeit zwischen Ingenieuren und Ärzten. • entwickeln Verständnis für Algorithmen und Mathematik der diagnostischen medizinischen Bildverarbeitung.

		• erfahren, dass kreative Adaption von bekannten Algorithmen auf neue Probleme der Schlüssel für ihre berufliche Zukunft ist. • entwickeln die Fähigkeit Algorithmen auf verschiedene Probleme anzupassen. • sind in der Lage, Algorithmen und Konzepte des Moduls anderen Studierenden der Technischen Fakultät zu erklären.
7	**Voraussetzungen für die Teilnahme**	Ingenieurmathematik Engineering Mathematics
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich/mündlich
11	**Berechnung der Modulnote**	schriftlich/mündlich (100%)
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 0 h Eigenstudium: 150 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 472330	**Dienstgüte von Kommunikationssystemen** (Quality of service in communication systems)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Quality of Service in Communications (2 SWS) Übung: Quality of Service in Communications (Ex-QoSiC) (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr. Reinhard German Lisa Maile	

4	**Modulverantwortliche/r**	Prof. Dr. Reinhard German
5	**Inhalt**	Zunächst wird der Begriff der Dienstgüte (Quality-of-Service, QoS) eingegrenzt und es werden die wichtigsten Ansätze zur Erzielung von Dienstgüte besprochen und in ausgewählten Netztechnologien untersucht. Dann werden unterschiedliche Methoden vorgestellt, mit denen Systeme bezüglich ihrer Dienstgüte bewertet und ausgelegt werden können: • Netzplanung und optimierung, • stochastische Analyse (Markow-Ketten, Warteschlangen), • Netzwerksimulation, • deterministische Analyse mit Network Calculus zur Ermittlung von Dienstgütegarantien • Messung (HW-, SW-, Hybrid-Monitoring, Benchmarks). Alle Methoden werden an Beispielen demonstriert. *Contents:* We introduce the term quality-of-service (QoS), discuss important approaches to achieve certain degrees of QoS, and show how the implementation in computer networks. Then a number of methodologies to assess and design systems with respect to their QoS: • network planning and optimization, • network simulation, • stochastic analysis (Markov chains, non-Markovian models, queuing systems), • deterministic analysis with network calculus to determine QoS guarantees • measurements (hardware, software, and hybrid monitoring, benchmarks). All methods are illustrated by examples.
6	**Lernziele und Kompetenzen**	Die Studierenden erwerben • Kenntnisse in Methoden zur Modellierung und Bewertung von quantitativen, nicht-funktionalen Eigenschaften von vernetzten Systemen • Kenntnisse in Mechanismen von vernetzten Systemen zur Erzielung von Dienstgüte *Learning targets and competences:*

		The students get • experience in methods to model and evaluate quantitative, non-functional properties of computer networks and related systems • knowledge of mechanisms of computer networks to achieve quality-of-service
7	**Voraussetzungen für die Teilnahme**	Rechnerkommunikation, Kommunikationssysteme, grundlegende Programmierkenntnisse (optimal in R und C++)
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	• Kurose, Ross. Computer Networking: A Top-Down Approach Featuring the Internet. 6th Ed., Addison Wesley, 2013 • W. Stallings. Data and Computer Communications, 10th ed., Pearson Education, 2014 • W. Stallings. Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud, Pearson Education, 2016

1	**Modulbezeichnung** 710850	**eBusiness Technologies und Evolutionäre Informationssysteme** (eBusiness technologies and evolutionary information systems)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Evolutionäre Informationssysteme (2 SWS) Vorlesung: eBusiness Technologies (2 SWS)	- -
3	Lehrende	Prof. Dr.-Ing. Richard Lenz	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Richard Lenz
5	**Inhalt**	**EBT:** • Überblick und Einblick in die wichtigsten Themen des Bereichs Business • User Interface, Business Logic und Database Layer • Agile Softwareentwicklung • Integration von Enterprise-Applikationen • Cloud & Container • DevOps **EIS:** • Grundlagen rechnergestützter Informationssysteme und organisatorisches Lernen • Erfolgsfaktoren für Projekte • Software Wartung vs. Software Evolution • Architekturmodelle • Grundprinzipien evolutionärer Systeme • Datenqualität in Informationssystemen **Contents:** **EBT:** • Modern technologies to implement Web-Applications for eBusiness • User Interface, Business Logic and Database Layer • Agile Software Development • Integration of Enterprise-Applications • Cloud & Container • DevOps **EIS:** • IT-Support for Organizational Learning • Success- and Failure Factors for large scale IT-Projects • Software Maintenance vs. Software Evolution • Architectural Styles and their Impact on Evolvability • Principles for Evolvable Systems • Data Quality in Information Systems

6	**Lernziele und Kompetenzen**	**EBT:** Die Studierenden • identifizieren die wichtigsten Themen des Bereichs eBusiness, von den Anwendungen bis zu den Implementierungen • verstehen Zusammenhänge der B2B-Integration und der Realisierung von eBusiness-Anwendungen • wiederholen Grundlagen des Webs • vergleichen technische Eigenschaften von HTTP-, Web- und Application Servern • vergleichen Markup Languages (HTML, XML) • unterscheiden Ansätze zur Schema-Modellierung wie DTD und XML Schema und erkennen die unterschiedliche Leistungsfähigkeit • verstehen Methoden zur evolutionsfähigen Gestaltung von Datenstrukturen in XML • unterscheiden Vorgehen bei der Datenhaltung und verschiedene Ansätze für den Datenbankzugriff • verstehen Objekt-relationale Mapping Frameworks am Beispiel von Hibernate und JPA • verstehen Komponentenmodelle wie Enterprise JavaBeans (EJB) aus dem JEE Framework • unterscheiden das EJB Komponentenmodell von den OSGi Bundles und den Spring Beans • verstehen und unterscheiden grundlegende Web Service Techniken wie SOAP und WSDL • unterscheiden Herangehensweisen zur dynamischen Generierung von Webseiten • verstehen grundlegende Eigenschaften eines Java-basierten Front-End-Frameworks am Beispiel von JSF • verstehen grundlegende Eigenschaften von Service-orientierten Architekturen (SOA) • verstehen agile Vorgehensmodelle zur Software-Entwicklung am Beispiel von Scrum • unterscheiden agile Verfahren wie Scrum von iterativ-inkrementellen Verfahren wie RUP • verstehen die Wichtigkeit von Code-Beispielen um die praktische Anwendbarkeit des theoretischen Wissens zu veranschaulichen. • können die Code-Beispiele eigenständig zur Ausführung bringen und die praktischen Erfahrungen interpretieren und bewerten • gestalten eigene Lernprozesse selbständig. • schätzen ihre eigenen Stärken und Schwächen im Hinblick auf die unterschiedlichen Architektur-Schichten ein(Benutzerinteraktion, Applikationslogik, Schnittstellenintegration, Datenbanksysteme) • identifizieren eine eigene Vorstellung als zukünftige Software-Architekten und können die eigene Entwicklung planen

- reflektieren durch regelmäßige fachbezogene Fragen des Dozenten Ihre eigene Lösungskompetenz.

EIS:

Die Studierenden:

- definieren die Begriffe "Informationssysteme", "evolutionäre Informationssyste" und "organisatorisches Lernen"
- grenzen die Begriffe "Wissen" und "Information" gegeneinander ab
- charakterisieren die in der Vorlesung erläuterten Formen der organisatorischen Veränderung
- erklären das SEKI Modell nach Nonaka und Takeuchi
- nennen Beispiele für die in der Vorlesung behandelten Formen der Wissensrepräsentation in IT-Systemen
- nennen typische Erfolgs- und Risikofaktoren für große IT-Projekte
- erklären die Kraftfeldtheorie nach Kurt Lewin
- unterscheiden Typen von Software gemäß der Klassifikation nach Lehman und Belady
- unterscheiden die in der Vorlesung vorgestellten Arten der Software Wartung
- benennen die Gesetzmäßigkeiten der Software-Evolution nach Lehman und Belady
- bewerten die in der Vorlesung vorgestellten Vorgehensmodelle zur Softwareerstellung im Kontext der E-Typ-Software
- nennen die in der Vorlesung vorgestellten Aspekte der Evolutionsfähigkeit von Software
- erklären, wie die in der Vorlesung vorgestellten Methoden zur Trennung von Belangen beitragen
- erklären das Konzept des "Verzögerten Entwurfs"
- erklären die Vor- und Nachteile generischer Datenbankschemata am Beispiel von EAV und EAV/CR
- charakterisieren die in der Vorlesung vorgestelten Architekurkonzepte
- grenzen die in der Vorlesung vorgestellten Integrationsanforderungen gegeneinander ab
- erklären wie Standards zur Systemintegration beitragen und wo die Grenzen der Standardisierung liegen
- erklären das Prinzip eines Kommunikationsservers und der nachrichtenbasierten Integration
- erklären den Begriff "Prozessintegration"
- definieren den Begriff "Enterprise Application Integration" (EAI)
- unterscheiden die in der Vorlesung vorgestellten Integrationsansätze
- erklären die in der Vorlesung vorgestellten Dimensionen der Datenqualität
- unterscheiden die grundlegenden Messmethoden für Datenqualität

		• erklären das Maßnahmenportfolio zur Verbesserung der Datenqualität nach Redman • benennen die in der Vorlesung vorgestellten Methoden zur Verbesserung der Datenqualität
7	**Voraussetzungen für die Teilnahme**	Programmieren in Java, Datenbanken (SQL)
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	siehe Lehrveranstaltungsbeschreibungen

1	**Modulbezeichnung** 939179	**Echtzeitsysteme 2 - Verlässliche Echtzeitsysteme** (Real-time systems 2 - dependable real-time systems)	**7,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Echtzeitsysteme 2 - Verlässliche Echtzeitsysteme (2 SWS) Übung: Echtzeitsysteme 2 - Verlässliche Echtzeitsysteme - Erweiterte Übungen (2 SWS) Übung: Echtzeitsysteme 2 - Verlässliche Echtzeitsysteme - Rechnerübungen (2 SWS)	- - -
3	Lehrende	Peter Wägemann Prof. Dr.-Ing. Wolfgang Schröder-Preikschat Tim Rheinfels Simon Schuster	

4	**Modulverantwortliche/r**	Peter Wägemann
5	**Inhalt**	Viele Echtzeitsysteme sind in Bereiche des täglichen Lebens eingebettete, die hohe Anforderungen an die funktionale Sicherheit dieser Systeme stellen. Beispiele hierfür sind Fahrerassistenzsysteme in modernen Automobilen, medizinische Geräte, Prozessanlagen in Kernkraftwerken oder Chemiefabriken oder Flugzeuge. Fehlfunktionen in diesen Anwendungen ziehen mitunter katastrophale Konsequenzen nach sich - Menschen können ernsthaft verletzt oder sogar getötet werden, Landstriche können unbewohnbar gemacht oder zumindest großer finanzieller Schaden verursacht werden. Dieses Modul betrachtet Methoden und Werkzeuge, die uns helfen können, einerseits *zuverlässig Software zu entwickeln* (also Fehler im Programm zu entdecken und zu vermeiden), und andererseits *zuverlässige Software zu entwickeln* (also Abstraktionen, die auch im Fehlerfall ihre Gültigkeit behalten). Hierbei steht weniger die Vermittlung theoretischer Grundkenntnisse auf diesen Gebieten im Vordergrund, also vielmehr • die praktische Anwendung existierende Werkzeuge und Methoden • sowie die Erfahrung und das Verständnis ihrer Grenzen. Auf diese Weise soll ein Fundament für die konstruktive Umsetzung verlässlicher Echtzeitsysteme gelegt werden. Dieses Modul soll daher fundierte Anknüpfungspunkte für die Entwicklung verlässlicher Echtzeitsysteme vermitteln, die Ad-hoc-Techniken möglichst ersetzen sollen.
6	**Lernziele und Kompetenzen**	Studierende, die das Modul erfolgreich abgeschlossen haben: • nennen die Konzepte und die Taxonomie verlässlicher Systeme, unterscheiden Software- und Hardwarefehler und klassifizieren Fehler (Defekt, Fehler, Fehlverhalten). • stellen Fehlerbäume auf. • organisieren Softwareentwicklungsprojekte mittels der Versionsverwaltung git.

		• vergleichen die verschiedenen Arten der Redundanz als Grundvorraussetzung für Fehlererkennung und -toleranz. • entwickeln fehlertolerante Systeme mittels Replikation. • diskutieren die Fehlerhypothese und die Sicherstellung von Replikdeterminismus. • erläutern die Vor- und Nachteile softwarebasierter Replikation und den Einsatz von Diversität. • wenden Informationsredundanz zur Härtung von Daten- und Kontrollflüssen an. • bewerten die Effektivität der arithmetischer Codierung von Programmen und verallgemeinern diesen Ansatz auf die verschiedenen Implementeirungsebenen (Maschinenprogramm zu Prozessinkarnation). • interpretieren den Einfluss der Ausführungsplattform (Hardware, Betriebssystem) auf die Leistungsfähigkeit der Fehlererkennung. • konzipieren eine fehlertolerante Ausführungsumgebung für ein softwarebasiertes TMR-System basierend auf ANBD-Codierung. • nennen die Grundlagen der systematischen Fehlerinjektion. • überprüfen die Wirksamkeit von Fehlertoleranzmechanismen mittels Fehlerinjektion auf der Befehlssatzebene. • entwickeln Testfälle für die Fehlerinjektion mittels des fail** Werkzeugs. • setzten Messergebnisse in Relation zu dem tatsächlichen Fehlerraum. • beschreiben die Grundlagen der Fehlererholung (Vorwärts- bzw. Rückwärtskorrektur) und Reintegration fehlgeschlagener Knoten. • vergleichen den Zustandstransfer am Beispiel der Running bzw. Recursive State Restoration. • benennen Konzepte der Rückwärtskorrektur durch Entwurfsalternativen (Recovery Blocks). • fassen die Grundlagen des dynamischen Testens zusammen. • unterscheiden Black-Box und White-Box Testverfahren. • konzipieren und implementieren Testfälle. • überprüfen die Testüberdeckung anhand grundlegender Überdeckungskriterien (Anweisungs- bis Bedingungsüberdeckung). • geben die Grundlagen der statischen Programmanalyse wieder. • nennen die Funktionsweise von Hoare- WP-Kalkül. • verifizieren eine Ampelsteuerung mittels des FramaC Werkzeugs zur statischen Analyse von C Programmen. • beschreiben den Korrektheitsnachweis mittels abstrakter Interpretation und unterscheiden die konkrete von der abstrakten Programmsemnatik. • erläutern die Funktionsweise von Sammel- und Präfixsemantiken.

		• erstellen einen Korrektheitsbeweis für einen a-b-Filter mittels des Astrée Werkzeugs zur abstrakten Interpretation von C Programmen. • bewerten die Verlässlichkeit kommerzieller, sicherheitskritischer Systeme anhand von Fallstudien (Sizewell B, Airbus A320). • erschließen sich typische Probleme und Fehlerquellen bei der Programmierung von eingebetteten Systemen im Allgemeinen. • klassifizieren Fallstricke und Mehrdeutigkeiten in der Programmiersprache C99 im Besonderen. • können in Gruppen kooperativ und effektiv arbeiten. • können ihre Entwurfs- und Implementierungsentscheidungen kompakt präsentieren und argumentativ vertreten. • reflektieren ihre Entscheidungen kritisch und leiten Alternativen ab. • können offen und konstruktiv mit Schwachpunkten in der Konzeption wie Umsetzung umgehen.
7	**Voraussetzungen für die Teilnahme**	Für eine erfolgreiche Teilnahme an der Veranstaltung sind grundlegende Programmierkenntnisse in C/C++ erforderlich. Diese können durch den (empfohlenen) Besuch entsprechender Grundlagenveranstaltungen oder im Eigenstudium erworben sein. Weiterhin sind grundlegende Kenntnisse über Echtzeitsystemeeine, zum Beispiel durch den Besuch der Veranstaltung "Echtzeitsysteme", empfohlen. Eine formale Voraussetzung besteht in diesem Zusammenhang jedoch nicht.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	Unregelmäßig
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 90 h Eigenstudium: 135 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 179490	**Echtzeitsysteme mit erweiterten Übungen** (Real-time systems with extended exercises)	**7,5 ECTS**
2	Lehrveranstaltungen	Übung: Rechnerübungen zu Echtzeitsysteme (2 SWS) Vorlesung: Echtzeitsysteme (2 SWS) Übung: Erweiterte Übungen zu Echtzeitsysteme (2 SWS)	- - 5 ECTS
3	Lehrende	Peter Wägemann Simon Schuster	

4	**Modulverantwortliche/r**	Peter Wägemann
5	**Inhalt**	Videobearbeitung in Echtzeit, Echtzeitstrategiespiel, echtzeitfähig - der Begriff Echtzeit ist wohl einer der am meisten strapazierten Begriffe der Informatik und wird in den verschiedensten Zusammenhängen benutzt. Diese Vorlesung beschäftigt sich mit dem Begriff Echtzeit aus der Sicht von Betriebssystemen - was versteht man eigentlich unter dem Begriff Echtzeit im Betriebssystemumfeld, wo und warum setzt man sog. Echtzeitbetriebssysteme ein und was zeichnet solche Echtzeitbetriebssysteme aus? In dieser Vorlesung geht es darum, die oben genannten Fragen zu beantworten, indem die grundlegenden Techniken und Mechanismen vermittelt werden, die man im Betriebssystemumfeld verwendet, um Echtzeitsysteme und Echtzeitbetriebssysteme zu realisieren. Im Rahmen dieser Vorlesung werden unter anderem folgende Themen behandelt: • zeitgesteuerte und ereignisgesteuerte Systeme • statische und dynamische Ablaufplanungsverfahren • Fadensynchronisation in Echtzeitbetriebssystemen • Behandlung von periodischen und nicht-periodischen Ereignissen In den begleitenden Übungen werden die in der Vorlesung vorgestellten Techniken bei der Entwicklung eines kleinen Echtzeitsystems praktisch umgesetzt.
6	**Lernziele und Kompetenzen**	Studierende, die das Modul erfolgreich abgeschlossen haben: • unterscheiden die verschiedenen Komponenten eines Echtzeitsystems. • bewerten die Verbindlichkeiten von Terminvorgaben (weich, fest, hart). • erläutern die Zusammensetzung des Laufzeitverhaltes einer Echtzeitanwendung. • klassifizieren die Berührungspunkte zwischen physikalischem Objekt und kontrollierendem Echtzeitsystem. • interpretieren die Zeitparameter des durch das Echtzeitrechensystem zu kontrollierenden Objekts.

		• nennen die Zeitparameter des zugrundeliegenden Rechensystems (Unterbrechungslatenz, Ausführungszeit, ...). • unterscheiden synchrone und asynchrone Programmunterbrechung (insbesondere Trap/Interrupt, Ausnahmebehandlung und Zustandssicherung). • skizzieren die Verwaltungsgemeinkosten des schlimmsten Falls. • entwickeln in der Programmiersprache C und wenden die GNU Werkzeugkette für den ARM Cortex M4 an. • erstellen Echtzeitanwendungen auf Basis der eCos OS-Schnittstelle • ordnen die Strukturelemente von Echtzeitanwendungen zu: Aufgabe, Arbeitsauftrag und Faden. • erläutern die Implikationen von zeitlichem Mehrfachbetrieb auf die Verwaltungsgemeinkosten. • unterscheiden die Umsetzungsalternativen zur Ablaufsteuerung und die Trennung der Belange in Einplanung (Strategie) und Einlastung (Mechanismus). • benennen die grundsätzliche Verfahren der Ablaufsteuerung (taktgesteuert, reihum, vorranggesteuert). • erklären die grundlegenden Zeitparameter einer Aufgabe (Auslösezeitpunkt, Termin, Antwortzeit, Latenz, Ausführungszeit, Schlupfzeit). • unterscheiden die Grundlagen der Planbarkeit (gültig vs. zulässig, Optimalität von Einplanungsalgorithmen). • beschreiben den Unterschied zwischen konstruktiver und analytischer Einhaltung von Terminen-. • vergleich die Möglichkeiten (statisch, dynamisch) der zeitliche Analyse von Echtzeitanwendungen. • erklären die Grundlagen und Beschränkungen von dynamischer (worst-case?) und statischer WCET-Analyse (makroskopisch und mikroskopisch). • illustrieren Lösungsverfahren zur Bestimmung des längsten Ausführungspfads (Timing Schema, IPET). • erstellen Zeitmessung mittels Zeitgeber / Oszilloskop und bestimmen den längsten Pfad durch Code-Review. • erproben und konzipieren werkzeuggestützte WCET-Analyse mittels des absint aiT Analysewerkzeugs. • Entwickeln und annotieren Flußrestrikionen für die statische WCET-Analyse. • beschreiben die Grundlagen der Abfertigung periodischer Echtzeitsysteme (Periode, Phase, Hyperperiode). • skizzieren das periodische Modell und dessen Folgen (Entwicklungskomfort vs. Analysierbarkeit). • erklären die ereignisgesteuerte Ausführung (feste und dynamische Priorität, Verdrängbarkeit) mittels ereignisorienterter Planer (Berechnungskomplexität, MLQ-Scheduler, O(1)-Scheduler).

		• unterscheiden die zeitgesteuerte Ausführung (Busy Loop, Ablaufplan)und die Abfertigung von Arbeitsaufträgen im Abfrage- bzw. Unterbrecherbetrieb. • wenden die Grundlagen der ereignisgesteuerten Ablaufplanung periodischer Echtzeitsysteme an. • unterscheiden Verfahren zur statischen (RM, DM) und dynamischen Prioritätsvergabe (EDF, LRT, LST). • nennen den Unterschied zwischen Anwendungs- und Systemebene (Mehrdeutigkeit von Prioritäten). • erläutern den Optimalitätsnachweis des RM-, DM- und EDF-Algorithmus und dessen Ausnahmen. • beschreiben grundlegende Verfahren zur Planbarkeitsanalyse (CPU-Auslastung, Antwortzeitanalyse). • implementieren komplexe Aufgabensysteme in eCos. • unterscheiden die Grundlagen der zeitgesteuerten Ablaufplanung periodischer Echtzeitsysteme. • erstellen regelmäßige, zyklische Ablaufpläne (cyclic executive model, Rahmen). • vergleich Methoden der manuellen und algorithmischen Ablaufplanung. • unterscheiden optimale von heuristischen Verfahren (List Scheduling, Branch & Bound). • diskutieren die Konsequenzen eines Betriebswechsels in Echtzeitsystemen. • erstellen takt- beziehungsweise ereignisgesteuerte Abläufe in eCos beziehungsweise tt-eCos. • entwickeln ein softwarebasiertes Oszilloskop und erstellen dessen zeitliche Analyse und Ablaufplanung. • klassifizieren die Grundlagen der Abfertigung nicht-periodischer Echtzeitsysteme (minimale Zwischenankunftszeit). • definieren die Verbindlichkeiten von nicht-periodischen Aufgaben (aperiodisch, sporadisch) • zeigen die sich ergebenden Restriktionen des periodischen Modells (Mischbetrieb, Prioritätswarteschlangen, Übernahmeprüfung) auf. • beschreiben die Basistechniken des Laufzeitsystems (Zusteller, Unterbrecherbetrieb, Hintergrundbetrieb). • quantifizieren die Eigenschaften und Auswirkungen auf den periodischen Teil des Echtzeitsystems. • formulieren die Grundlagen des Slack-Stealing. • beschreiben den Einsatz von bandweite-bewahrenden Zustellern. • unterscheiden aufschiebbare Zusteller und Sporadic Server (SpSL und POSIX). • wenden eine Übernahmeprüfung bei sporadischen Aufgaben mittels dichte- oder schlupfbasierten Akzeptanztests an. • arbeiten einen strukturierter Ablaufplan (Rahmen) aus und untersuchen den Einsatz von Slack-Stealing.

		• ermitteln gerichtete Abhängigkeiten und Rangfolgen in Echtzeitanwendungen (Abhängigkeits- und Aufgabengraph). • stellen Umsetzungsalternativen für Abhängigkeiten einander gegenüber (naiv, implizit, explizit). • beschreiben das Konzept der zeitlichen Domänen und physikalischer bzw. logischer Ereignisse. • übertragen Abhängigkeiten auf das Problem der Ablaufplanung (modifiziere Auslösezeitpunkt/Termin, Phasenversatz). • konzipieren Rangfolge und aperiodische Steuerung in eCos. • implementieren einen aperiodischer Moduswechsel mit Zustandsüberführung in eCos. • gestalten einen Signal-Trigger für das entwickelte softwarebasierten Oszilloskops. • konzipieren explizite Synchronisation mittels Nachrichten in eCos. • wenden die Grundlagen von Wettstreit um Betriebsmitteln, Konkurrenz und Konfliktsituationen (kritische Abschnitte, (un)kontrollierte Prioritätsumkehr) an. • beschreiben echtzeitfähige Synchronisationsprotokolle (NPCS, PI, PCP). • nennen die Vor- und Nachteile der Techniken (transitive Blockung, Verklemmungen). • hinterfragen die Vereinfachung des PCP durch stapelbezogene Grenzprioritäten. • bestimmen die Ablaufplanung unter Berücksichtigung von Blockierungszeiten und Selbstsuspendierung. • implementieren Zugriffskontrolle (NPCS, PI, PCP) in Echtzeitanwendungen mit eCos. • analysieren Blockade für die Zugriffskontrolle in eCos. • erläutern die Anforderungen an verteile Echtzeitsysteme (Komposition, Erweiterbarkeit, Komplexität, Ereignis- vs. Zustandsnachricht). • fassen die Grundlagen von Knoten, Netzwerkschnittstellen und Netzübergängen sowie die Konzepte der expliziten und impliziten Flusskontrolle zusammen. • erschließen sich typische Probleme (zeitliche Analyse, Beobachtbarkeit, Synchronisation, Rangfolge) und Fehlerquellen bei der Programmierung von Echtzeitanwendungen. • können in Gruppen kooperativ und effektiv arbeiten. • können ihre Entwurfs- und Implementierungsentscheidungen kompakt präsentieren und argumentativ vertreten. • reflektieren ihre Entscheidungen kritisch und leiten Alternativen ab. • können offen und konstruktiv mit Schwachpunkten in der Konzeption wie Umsetzung umgehen.
7	**Voraussetzungen für die Teilnahme**	Für eine erfolgreiche Teilnahme an der Veranstaltung sind grundlegende Programmierkenntnisse in C/C++ erforderlich.

		Diese können durch den (empfohlenen) Besuch entsprechender Grundlagenveranstaltungen oder im Eigenstudium erworben sein, eine formale Voraussetzung besteht in diesem Zusammenhang nicht.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	Unregelmäßig
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 90 h Eigenstudium: 135 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• Hermann Kopetz. Real-Time Systems: Design Principles for Distributed Embedded Applications. Kluwer Academic Publishers, 1997. • Jane W. S. Liu. Real-Time Systems. Prentice-Hall, Inc., 2000. • Wolfgang Schröder-Preikschat. System-programmierung. Vorlesungsfolien. 2006.

1	**Modulbezeichnung** 707303	**Echtzeitsysteme (Vorlesung mit Übungen)** (Real-time systems)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Rechnerübungen zu Echtzeitsysteme (2 SWS) Vorlesung: Echtzeitsysteme (2 SWS) Übung: Übungen zu Echtzeitsysteme (2 SWS)	- - 2,5 ECTS
3	Lehrende	Peter Wägemann Simon Schuster Tim Rheinfels	

4	**Modulverantwortliche/r**	Peter Wägemann
5	**Inhalt**	Videobearbeitung in Echtzeit, Echtzeitstrategiespiel, echtzeitfähig - der Begriff Echtzeit ist wohl einer der am meisten strapazierten Begriffe der Informatik und wird in den verschiedensten Zusammenhängen benutzt. Diese Vorlesung beschäftigt sich mit dem Begriff Echtzeit aus der Sicht von Betriebssystemen - was versteht man eigentlich unter dem Begriff Echtzeit im Betriebssystemumfeld, wo und warum setzt man sog. Echtzeitbetriebssysteme ein und was zeichnet solche Echtzeitbetriebssysteme aus? In dieser Vorlesung geht es darum, die oben genannten Fragen zu beantworten, indem die grundlegenden Techniken und Mechanismen vermittelt werden, die man im Betriebssystemumfeld verwendet, um Echtzeitsysteme und Echtzeitbetriebssysteme zu realisieren. Im Rahmen dieser Vorlesung werden unter anderem folgende Themen behandelt: • zeitgesteuerte und ereignisgesteuerte Systeme • statische und dynamische Ablaufplanungsverfahren • Fadensynchronisation in Echtzeitbetriebssystemen • Behandlung von periodischen und nicht-periodischen Ereignissen In den begleitenden Übungen werden die in der Vorlesung vorgestellten Techniken bei der Entwicklung eines kleinen Echtzeitsystems praktisch umgesetzt.
6	**Lernziele und Kompetenzen**	Studierende, die das Modul erfolgreich abgeschlossen haben: • unterscheiden die verschiedenen Komponenten eines Echtzeitsystems. • bewerten die Verbindlichkeiten von Terminvorgaben (weich, fest, hart). • erläutern die Zusammensetzung des Laufzeitverhaltes einer Echtzeitanwendung. • klassifizieren die Berührungspunkte zwischen physikalischem Objekt und kontrollierendem Echtzeitsystem. • interpretieren die Zeitparameter des durch das Echtzeitrechensystem zu kontrollierenden Objekts.

		• nennen die Zeitparameter des zugrundeliegenden Rechensystems (Unterbrechungslatenz, Ausführungszeit, ...). • unterscheiden synchrone und asynchrone Programmunterbrechung (insbesondere Trap/Interrupt, Ausnahmebehandlung und Zustandssicherung). • skizzieren die Verwaltungsgemeinkosten des schlimmsten Falls. • entwickeln in der Programmiersprache C und wenden die GNU Werkzeugkette für den ARM Cortex M4 Microcontroller an. • erstellen Echtzeitanwendungen auf Basis der eCos OS-Schnittstelle • ordnen die Strukturelemente von Echtzeitanwendungen zu: Aufgabe, Arbeitsauftrag und Faden. • erläutern die Implikationen von zeitlichem Mehrfachbetrieb auf die Verwaltungsgemeinkosten. • unterscheiden die Umsetzungsalternativen zur Ablaufsteuerung und die Trennung der Belange in Einplanung (Strategie) und Einlastung (Mechanismus). • benennen die grundsätzliche Verfahren der Ablaufsteuerung (taktgesteuert, reihum, vorranggesteuert). • erklären die grundlegenden Zeitparameter einer Aufgabe (Auslösezeitpunkt, Termin, Antwortzeit, Latenz, Ausführungszeit, Schlupfzeit). • unterscheiden die Grundlagen der Planbarkeit (gültig vs. zulässig, Optimalität von Einplanungsalgorithmen). • beschreiben den Unterschied zwischen konstruktiver und analytischer Einhaltung von Terminen-. • vergleich die Möglichkeiten (statisch, dynamisch) der zeitliche Analyse von Echtzeitanwendungen. • erklären die Grundlagen und Beschränkungen von dynamischer (worst-case?) und statischer WCET-Analyse (makroskopisch und mikroskopisch). • illustrieren Lösungsverfahren zur Bestimmung des längsten Ausführungspfads (Timing Schema, IPET). • erstellen Zeitmessung mittels Zeitgeber / Oszilloskop und bestimmen den längsten Pfad durch Code-Review. • erproben werkzeuggestützte WCET-Analyse mittels des absint aiT Analysewerkzeugs. • beschreiben die Grundlagen der Abfertigung periodischer Echtzeitsysteme (Periode, Phase, Hyperperiode). • skizzieren das periodische Modell und dessen Folgen (Entwicklungskomfort vs. Analysierbarkeit). • erklären die ereignisgesteuerte Ausführung (feste und dynamische Priorität, Verdrängbarkeit) mittels ereignisorienterter Planer (Berechnungskomplexität, MLQ-Scheduler, O(1)-Scheduler). • unterscheiden die zeitgesteuerte Ausführung (Busy Loop, Ablaufplan)und die Abfertigung von Arbeitsaufträgen im Abfrage- bzw. Unterbrecherbetrieb.

		• wenden die Grundlagen der ereignisgesteuerten Ablaufplanung periodischer Echtzeitsysteme an. • unterscheiden Verfahren zur statischen (RM, DM) und dynamischen Prioritätsvergabe (EDF, LRT, LST). • nennen den Unterschied zwischen Anwendungs- und Systemebene (Mehrdeutigkeit von Prioritäten). • erläutern den Optimalitätsnachweis des RM-, DM- und EDF-Algorithmus und dessen Ausnahmen. • beschreiben grundlegende Verfahren zur Planbarkeitsanalyse (CPU-Auslastung, Antwortzeitanalyse). • implementieren komplexe Aufgabensysteme in eCos. • unterscheiden die Grundlagen der zeitgesteuerten Ablaufplanung periodischer Echtzeitsysteme. • erstellen regelmäßige, zyklische Ablaufpläne (cyclic executive model, Rahmen). • vergleich Methoden der manuellen und algorithmischen Ablaufplanung. • unterscheiden optimale von heuristischen Verfahren (List Scheduling, Branch & Bound). • diskutieren die Konsequenzen eines Betriebswechsels in Echtzeitsystemen. • erstellen takt- beziehungsweise ereignisgesteuerte Abläufe in eCos beziehungsweise tt-eCos. • klassifizieren die Grundlagen der Abfertigung nicht-periodischer Echtzeitsysteme (minimale Zwischenankunftszeit). • definieren die Verbindlichkeiten von nicht-periodischen Aufgaben (aperiodisch, sporadisch) • zeigen die sich ergebenden Restriktionen des periodischen Modells (Mischbetrieb, Prioritätswarteschlangen, Übernahmeprüfung) auf. • beschreiben die Basistechniken des Laufzeitsystems (Zusteller, Unterbrecherbetrieb, Hintergrundbetrieb). • quantifizieren die Eigenschaften und Auswirkungen auf den periodischen Teil des Echtzeitsystems. • formulieren die Grundlagen des Slack-Stealing. • beschreiben den Einsatz von bandweite-bewahrenden Zustellern. • unterscheiden aufschiebbare Zusteller und Sporadic Server (SpSL und POSIX). • wenden eine Übernahmeprüfung bei sporadischen Aufgaben mittels dichte- oder schlupfbasierten Akzeptanztests an. • arbeiten einen strukturierter Ablaufplan (Rahmen) aus und untersuchen den Einsatz von Slack-Stealing. • ermitteln gerichtete Abhängigkeiten und Rangfolgen in Echtzeitanwendungen (Abhängigkeits- und Aufgabengraph). • stellen Umsetzungsalternativen für Abhängigkeiten einander gegenüber (naiv, implizit, explizit). • beschreiben das Konzept der zeitlichen Domänen und physikalischer bzw. logischer Ereignisse.

		• übertragen Abhängigkeiten auf das Problem der Ablaufplanung (modifiziere Auslösezeitpunkt/Termin, Phasenversatz). • konzipieren Rangfolge und aperiodische Steuerung in eCos. • implementieren einen aperiodischer Moduswechsel mit Zustandsüberführung in eCos. • wenden die Grundlagen von Wettstreit um Betriebsmitteln, Konkurrenz und Konfliktsituationen (kritische Abschnitte, (un)kontrollierte Prioritätsumkehr) an. • beschreiben echtzeitfähige Synchronisationsprotokolle (NPCS, PI, PCP). • nennen die Vor- und Nachteile der Techniken (transitive Blockung, Verklemmungen). • hinterfragen die Vereinfachung des PCP durch stapelbezogene Grenzprioritäten. • bestimmen die Ablaufplanung unter Berücksichtigung von Blockierungszeiten und Selbstsuspendierung. • implementieren Zugriffskontrolle (NPCS, PI, PCP) in Echtzeitanwendungen mit eCos. • erläutern die Anforderungen an verteile Echtzeitsysteme (Komposition, Erweiterbarkeit, Komplexität, Ereignis- vs. Zustandsnachricht). • fassen die Grundlagen von Knoten, Netzwerkschnittstellen und Netzübergängen sowie die Konzepte der expliziten und impliziten Flusskontrolle zusammen. • erschließen sich typische Probleme (zeitliche Analyse, Beobachtbarkeit, Synchronisation, Rangfolge) und Fehlerquellen bei der Programmierung von Echtzeitanwendungen. • können in Gruppen kooperativ und effektiv arbeiten. • können ihre Entwurfs- und Implementierungsentscheidungen kompakt präsentieren und argumentativ vertreten. • reflektieren ihre Entscheidungen kritisch und leiten Alternativen ab. • können offen und konstruktiv mit Schwachpunkten in der Konzeption wie Umsetzung umgehen.
7	**Voraussetzungen für die Teilnahme**	Für eine erfolgreiche Teilnahme an der Veranstaltung sind grundlegende Programmierkenntnisse in C/C++ erforderlich. Diese können durch den (empfohlenen) Besuch entsprechender Grundlagenveranstaltungen oder im Eigenstudium erworben sein, eine formale Voraussetzung besteht in diesem Zusammenhang nicht.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio

11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	Unregelmäßig
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 90 h Eigenstudium: 60 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• Hermann Kopetz. Real-Time Systems: Design Principles for Distributed Embedded Applications. Kluwer Academic Publishers, 1997. • Jane W. S. Liu. Real-Time Systems. Prentice-Hall, Inc., 2000. • Wolfgang Schröder-Preikschat. Softwaresysteme 1. Vorlesungsfolien. 2006.

1	**Modulbezeichnung** 502509	**Hardware-Software-Co-Design** (Hardware-software-co-design)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übungen zu Hardware-Software-Co-Design (2 SWS) Vorlesung: Hardware-Software-Co-Design (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Muhammad Sabih Tobias Hahn Dr.-Ing. Stefan Wildermann Prof. Dr.-Ing. Jürgen Teich	

4	**Modulverantwortliche/r**	Joachim Falk Prof. Dr.-Ing. Jürgen Teich
5	**Inhalt**	Zahlreiche Realisierungen eingebetteter Systeme (z.B. Mobiltelephone, Faxgeräte, Industriesteuerungen) zeichnen sich durch kooperierende Hardware- und Softwarekomponenten aus. Die Popularität solcher Realisierungsformen lässt sich begründen durch 1) die steigende Vielfalt und Komplexität heterogener Systeme, 2) die Notwendigkeit, Entwurfs- und Testkosten zu senken und 3) Fortschritte in Schlüsseltechnologien (Mikroelektronik, formale Entwurfsmethoden). Zum Beispiel bieten Halbleiterhersteller kostengünstige ASICs an, die einen Mikrocontroller und benutzerspezifische Peripherie und Datenpfade auf einem Chip integrieren. Die Synthese solcher Systeme wirft jedoch eine Reihe neuartiger Entwurfsprobleme auf, insbesondere 1) die Frage der Auswahl von Hardware- und Softwarekomponenten, 2) die Partitionierung einer Spezifikation in Hard- und Software, 3) die automatische Synthese von Interface- und Kommunikationsstrukturen und 4) die Verifikation und Cosimulation. 1) Überblick und Vergleich von Architekturen und Komponenten in Hardware/Software-Systemen. 2) Aufbau eines Compilers und Codeoptimierungsverfahren für Hardware und Software 3) Hardware/Software-Partitionierung (Partitionierung komplexer Systeme, Schätzungsverfahren, Performanzanalyse, Codegenerierung) 4) Interfacesynthese (Kommunikationsarten, Synchronisation, Synthese) 5) Verifikation und Cosimulation 6) Tafelübungen Content: Numerous realisations of embedded systems (e.g. mobile phones, fax machines, industrial controls) are characterised by cooperating hardware and software components. The popularity of such realisations can be explained by 1) the increasing diversity and complexity of heterogeneous systems, 2) the need to reduce design and testing costs, and 3) advances in key technologies (microelectronics, formal design methods). For example, semiconductor manufacturers

		offer low-cost ASICs that integrate a microcontroller and user-specific peripherals and data paths on a single chip. However, the synthesis of such systems raises a number of novel design problems, in particular 1) the issue of hardware and software component selection, 2) the partitioning of a specification into hardware and software, 3) the automatic synthesis of interface and communication structures, and 4) verification and cosimulation. 1) Overview and comparison of architectures and components in hardware/software systems. 2) Structure of a compiler and code optimisation procedures for hardware and software. 3) Hardware/software partitioning (partitioning of complex systems, estimation procedures, performance analysis, code generation) 4) Interface synthesis (communication types, synchronisation, synthesis) 5) Verification and cosimulation 6) Blackboard exercises
6	**Lernziele und Kompetenzen**	Fachkompetenz - Wissen • Die Studierenden erhalten Einblick in ein aktuelles Forschungsgebiet. Fachkompetenz - Verstehen • Die Studierenden verstehen Grundlagen des modernen Systementwurfs. • Die Studierenden erklären Implementierungsalternativen für digitale Hardware/Software-Systeme. Fachkompetenz - Anwenden • Die Studierenden wenden grundlegende Algorithmen an, zur Analyse und Optimierung von Hardware/Software-Systemen. Competneces: Professional competence - Knowledge • Students gain insight into a current field of research. Professional competence - Understanding • Students understand the basics of modern system design. • Students explain implementation alternatives for digital hardware/software systems. Professional competence - Application

		• Students apply basic algorithms to analyse and optimise hardware/software systems.
7	**Voraussetzungen für die Teilnahme**	Die Auswahl dieses Moduls schließt die Auswahl des Moduls „Hardware-Software-Co-Design (Vorlesung mit erweiterter Übung)“ aus.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Empfohlene Bücher zur Begleitung und Vertiefung: • Teich J., Haubelt C.: "Digitale Hardware/Software-Systeme: Synthese und Optimierung", Springer-Verlag, 2007, ISBN: 978-3-540-46822-6 • Gajski, D. et al.: "Specification and Design of Embedded Systems", Prentice Hall, Englewood Cliffs, NJ, 1994, ISBN: 978-0131507319 Weitere Informationen: https://www.cs12.tf.fau.de/lehre/lehrveranstaltungen/vorlesungen/hardware-software-co-design

1	**Modulbezeichnung** 292952	**Hardware-Software-Co-Design (Vorlesung mit erweiterter Übung)** (Hardware-software-co-design (Lecture with extended exercises))	**7,5 ECTS**
2	Lehrveranstaltungen	Übung: Übungen zu Hardware-Software-Co-Design (2 SWS) Vorlesung: Hardware-Software-Co-Design (2 SWS) Übung: Erweiterte Übungen zu Hardware-Software-Co-Design (2 SWS)	2,5 ECTS 2,5 ECTS 2,5 ECTS
3	Lehrende	Muhammad Sabih Tobias Hahn Dr.-Ing. Stefan Wildermann Prof. Dr.-Ing. Jürgen Teich	

4	**Modulverantwortliche/r**	Joachim Falk Prof. Dr.-Ing. Jürgen Teich
5	**Inhalt**	Zahlreiche Realisierungen eingebetteter Systeme (z.B. Mobiltelephone, Faxgeräte, Industriesteuerungen) zeichnen sich durch kooperierende Hardware- und Softwarekomponenten aus. Die Popularität solcher Realisierungsformen lässt sich begründen durch 1) die steigende Vielfalt und Komplexität heterogener Systeme, 2) die Notwendigkeit, Entwurfs- und Testkosten zu senken und 3) Fortschritte in Schlüsseltechnologien (Mikroelektronik, formale Entwurfsmethoden). Zum Beispiel bieten Halbleiterhersteller kostengünstige ASICs an, die einen Mikrocontroller und benutzerspezifische Peripherie und Datenpfade auf einem Chip integrieren. Die Synthese solcher Systeme wirft jedoch eine Reihe neuartiger Entwurfsprobleme auf, insbesondere 1) die Frage der Auswahl von Hardware- und Softwarekomponenten, 2) die Partitionierung einer Spezifikation in Hard- und Software, 3) die automatische Synthese von Interface- und Kommunikationsstrukturen und 4) die Verifikation und Cosimulation. 1) Überblick und Vergleich von Architekturen und Komponenten in Hardware/Software-Systemen. 2) Aufbau eines Compilers und Codeoptimierungsverfahren für Hardware und Software 3) Hardware/Software-Partitionierung (Partitionierung komplexer Systeme, Schätzungsverfahren, Performanzanalyse, Codegenerierung) 4) Interfacesynthese (Kommunikationsarten, Synchronisation, Synthese) 5) Verifikation und Cosimulation 6) Tafelübungen 7) Demonstrationen mit rechnergestützten Entwurfswerkzeugen und praktische Übungen Content: Numerous realisations of embedded systems (e.g. mobile phones, fax machines, industrial controls) are characterised by

		cooperating hardware and software components. The popularity of such realisations can be explained by 1) the increasing diversity and complexity of heterogeneous systems, 2) the need to reduce design and testing costs, and 3) advances in key technologies (microelectronics, formal design methods). For example, semiconductor manufacturers offer low-cost ASICs that integrate a microcontroller and user-specific peripherals and data paths on a single chip. However, the synthesis of such systems raises a number of novel design problems, in particular 1) the issue of hardware and software component selection, 2) the partitioning of a specification into hardware and software, 3) the automatic synthesis of interface and communication structures, and 4) verification and cosimulation. 1) Overview and comparison of architectures and components in hardware/software systems. 2) Structure of a compiler and code optimisation procedures for hardware and software. 3) Hardware/software partitioning (partitioning of complex systems, estimation procedures, performance analysis, code generation) 4) Interface synthesis (communication types, synchronisation, synthesis) 5) Verification and cosimulation 6) Blackboard exercises 7) Demonstrations with computer-aided design tools and practical exercises
6	**Lernziele und Kompetenzen**	Fachkompetenz - Wissen • Die Studierenden erhalten Einblick in ein aktuelles Forschungsgebiet. Fachkompetenz - Verstehen • Die Studierenden verstehen Grundlagen des modernen Systementwurfs. • Die Studierenden erklären Implementierungsalternativen für digitale Hardware/Software-Systeme. Fachkompetenz - Anwenden • Die Studierenden wenden grundlegende Algorithmen an, zur Analyse und Optimierung von Hardware/Software-Systemen. • Die Studierenden wenden das erlernte Wissen in den erweiterten Übungen vor Ort an den Rechnerarbeitsplätzen des Lehrstuhls an. Sozialkompetenz • Die Studierenden benutzen aktuelle Entwurfswerkzeuge für die Spezifikation, Optimierung und Prototypisierung von Hardware/

		Software-Systemen bei der kooperativen Bearbeitung der erweiterten Übung in Gruppen. Competences: Professional competence - Knowledge • Students gain insight into a current field of research. Professional competence - Understanding • Students understand the basics of modern system design. • Students explain implementation alternatives for digital hardware/software systems. Professional competence - Application • Students apply basic algorithms to analyse and optimise hardware/software systems. • The students apply the knowledge they have acquired in the extended exercises on site at the computer workstations of the department. Social competence • The students use current design tools for the specification, optimisation and prototyping of hardware/software systems in the cooperative processing of the extended exercise in groups.
7	**Voraussetzungen für die Teilnahme**	Die Auswahl dieses Moduls schließt die Auswahl des Moduls „Hardware-Software-Co-Design“ aus.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20191 Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio Klausur (90 min) und erfolgreicher Bearbeitung aller Übungsaufgaben in den erweiterten Übungen (verpflichtend, vor Ort an den Rechnerarbeitsplätzen des Lehrstuhls). Die Sprache der Klausur ist abhängig von der Wahl der Studierenden entweder Deutsch oder Englisch.
11	**Berechnung der Modulnote**	Portfolio (100%) Die Modulnote ergibt sich aus der Klausurnote.
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 90 h Eigenstudium: 135 h
14	**Dauer des Moduls**	1 Semester

15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Empfohlene Bücher zur Begleitung und Vertiefung: • Teich J., Haubelt C.: "Digitale Hardware/Software-Systeme: Synthese und Optimierung", Springer-Verlag, 2007, ISBN: 978-3-540-46822-6 • Gajski, D. et al.: "Specification and Design of Embedded Systems", Prentice Hall, Englewood Cliffs, NJ, 1994, ISBN: 978-0131507319 Weitere Informationen: https://www.cs12.tf.fau.de/lehre/lehrveranstaltungen/vorlesungen/hardware-software-co-design

1	**Modulbezeichnung** 93020	**Implementierung von Datenbanksystemen** (Implementation of database systems)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Implementierung von Datenbanksystemen (2 SWS) Übung: Übungen zu Implementierung von Datenbanksystemen (2 SWS) Übung: Intensivierungsübung zu Implementierung von Datenbanksystemen (2 SWS)	- - -
3	Lehrende	Demian Vöhringer	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Richard Lenz Prof. Dr.-Ing. Klaus Meyer-Wegener
5	**Inhalt**	Die Vorlesung führt ein in den Aufbau und die Architektur von Datenbanksystemen, die Modularisierung und Schichtenbildung mit Abstraktionen verwenden. Schwerpunkt sind deshalb systemtechnische Aspekte von Datenbanksystemen. Die Übungen vertiefen verschiedene Aspekte an Beispielrechnungen und erweitern gelegentlich auch noch den Stoff um einige Facetten (z.b. Mehrattribut-Zugriffspfade). Ausgangspunkt einer Reihe von aufeinander aufbauenden Abstraktionen ist die Speicherung von Daten auf Hintergrundspeichern. Die erste Abstraktion ist die Datei. Dann werden Sätze eingeführt und auf verschiedene Weisen in Blöcken organisiert (sequenziell, mit Direktzugriff, indexsequentiell). Das schließt die Organisation eines Blockpuffers und Zugriffspfade (Indexstrukturen) unterschiedlichen Typs ein. Als zweite große Abstraktion werden Datenmodelle eingeführt und hier insbesondere das relationale. Das ist bereits aus dem Modul "Konzeptionelle Modellierung" bekannt, wird hier aber aus einer ganz anderen Perspektive heraus entwickelt. Der zweite Teil befasst sich mit der Realisierung der Leistungen eines Datenbanksystems unter Verwendung der vorher eingeführten Sätze und Zugriffspfade ("top-down"). Das umfasst die Anfrageverarbeitung und -optimierung, aber auch die Mechanismen zur Protokollierung von Aktionen und zur Wiederherstellung von Datenbankzuständen nach einem Fehler oder Ausfall. Ein laufend vervollständigtes Schichtenmodell fasst abschließend die Aufgaben in einer Architektur für Datenbank-Verwaltungssysteme zusammen. Ziel des Moduls ist es also, ein grundlegendes Verständnis für den Aufbau und die Funktionsweise eines Datenbanksystems zu vermitteln.
6	**Lernziele und Kompetenzen**	Die Studierenden • kennen das Schichtenmodell eines Datenbankverwaltungssystems; • verstehen das Prinzip der Datenunabhängigkeit (Datenabstraktion); • beherrschen das Aufbauprinzip einer Software-Schicht; • unterscheiden die Begriffe "Datenbank", "Datenbanksystem" und "Datenbankverwaltungssystem";

		• unterscheiden die Begriffe "Datenmodell" und "Schema"; • zeigen das Konzept der blockorientierten Datei mit ihren Zugriffsoperationen auf; • unterscheiden einen Satz von einem Block; • erklären das Konzept der sequentiellen Satzdatei; • schildern das Prinzip der Wechselpuffertechnik; • charakterisieren den Schlüsselzugriff auf Sätze; • stellen Gestreute Speicherung (Hashing) auf der Basis von Blöcken (Buckets) dar; • formulieren die Funktionsweise des Virtuellen Hashings; • fassen die Funktionsweise eines B-Baums zusammen; • unterscheiden die Dienste eines B-Baums von denen des Hashings; • können für eine Folge von Schlüsselwerten einen B-Baum aufbauen; • unterscheiden einen B-Baum von einem B-Stern-Baum (B+-Baum); • veranschaulichen einen Bitmap-Index; • unterscheiden die Primär- und Sekundärorganisation von Sätzen; • zählen Ersetzungsstrategien der Pufferverwaltung auf und vergleichen sie; • benennen die Dienste einer Pufferverwaltung; • erklären die Konzepte "Seite" und "Segment" im Gegensatz zu "Block" und "Datei"; • unterscheiden direkte und indirekte Seitenzuordnung; • interpretieren in Programmiersprachen eingebettete Anfragesprachen und Datenbank-Unterprogrammaufrufe; • charakterisieren Datenbank-Transaktionen; • kennen die Aufrufe zur Definition von Transaktionen; • erläutern die spaltenweise Abspeicherung von Relationen; • diskutieren die algebraische Optimierung von Anfragen; • stellen Planoperatoren eines Datenbanksystems dar; • unterscheiden Planoperatoren für den Verbund; • beschreiben Kostenformeln für die Abschätzung von Anfrageausführungen; • schildern die verschiedenen Anomalien im Mehrbenutzerbetrieb; • beschreiben die Serialisierbarkeit von Transaktionen; • erläutern das Konzept der Sperren in Datenbanksystemen; • unterscheiden physische und logische Konsistenz; • kennen die vier Recovery-Klassen; • erläutern die verschiedenen Arten von Sicherungspunkten.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Masterprüfung Master of Science Information and Communication Technology 20191

		Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	KEMPER, Alfons ; EICKLER, André: Datenbanksysteme : Eine Einführung. 9., aktual. u. erweit. Aufl. München : Oldenbourg, 2013. ISBN 978-3-486-72139-3. Kapitel 7 bis 11 KEMPER, Alfons ; WIMMER, Martin: Übungsbuch Datenbanksysteme. 2., aktual. u. erweit. Aufl. München : Oldenbourg, 2009. ISBN 978-3-486-59001-2. Kapitel 7 bis 11 HEUER, Andreas ; SAAKE, Gunter: Datenbanken : Konzepte und Sprachen. 3., aktual. u. erw. Aufl. Bonn : mitp, 2007. - ISBN 3-8266-1664-2 HÄRDER, Theo ; RAHM, Erhard: Datenbanksysteme : Konzepte und Techniken der Implementierung. Berlin : Springer, 1999 - ISBN 3-540-65040-7 SAAKE, Gunter ; HEUER, Andreas: Datenbanken : Implementierungstechniken. 2., aktual. u. erw. Aufl. Bonn : mitp, 2005. ISBN 3-8266-1438-0

1	**Modulbezeichnung** 43391	**Interaktive Computergraphik** (Interactive computer graphics)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Tutorials to Interactive Computer Graphics (2 SWS) Vorlesung: Interactive Computer Graphics (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Jonas Müller Laura Fink Prof. Dr. Marc Stamminger	

4	**Modulverantwortliche/r**	Prof. Dr. Marc Stamminger
5	**Inhalt**	In dem Modul werden GPUs und dafür maßgeschneiderte Algorithmen behandelt: • Architektur von GPUs und Echtzeit-Rendering-Pipeline • Deferred Shading und Anti-Aliasing-Verfahren • Simulation von Umgebungsbeleuchtung • Verfahren zur Generierung von Schatten • Level-of-Detail-Verfahren zur Darstellung komplexer Szenen • Animation von Objekten
6	**Lernziele und Kompetenzen**	Fachkompetenz Verstehen Lernende können verschiedene in der Vorlesung behandelte Verfahren der interaktiven Computergraphik mit eigenen Worten erklären. Anwenden Lernende können Zusammenhänge zwischen den in der Vorlesung behandelten Verfahren der interaktiven Computergraphik erkennen, und Ideen auf neue Anwendungen übertragen. Analysieren Lernende können Unterschiede und Ähnlichkeiten der in der Vorlesung behandelten Verfahren erkennen und daraus neue Lösungen entwickeln.
7	**Voraussetzungen für die Teilnahme**	Die Übungen setzen Kenntnisse in C/C++ voraus.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Masterprüfung Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Übungsleistung Variabel

11	**Berechnung der Modulnote**	Übungsleistung (0%) Variabel (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• Möller, Haines: "Real-Time Rendering"

1	**Modulbezeichnung** 44140	**Interventional Medical Image Processing** (Interventional medical image processing)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Medical Image Processing for Interventional Applications (VHB-Kurs) (4 SWS)	5 ECTS
3	Lehrende	Manuela Meier Arpitha Ravi Celia Martín Vicario Luis Rivera Monroy	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Andreas Maier
5	**Inhalt**	English Version: This module focuses on recent developments in image processing driven by medical applications. All algorithms are motivated by practical problems. The mathematical tools required to solve the considered image processing tasks will be introduced. The module starts with an overview on preprocessing algorithms such as scatter correction for x-ray images, edge detection, super-resolution and edge-preserving noise reduction. The second chapter describes automatic image analysis using feature descriptors, key point detection, and segmentation using bottom-up algorithms such as the random walker or top-down approaches such as active shape models. Furthermore, the module covers geometric calibration algorithms for single view calibration, epipolar geometry, and factorization. The last part of the module covers non-rigid registration based on variational methods and motion-compensated image reconstruction. Deutsche Version: Das Modul ist auf die jüngsten Entwicklungen in der Verarbeitung von medizinischen Bildern ausgerichtet. Alle Algorithmen werden durch praktische Probleme motiviert. Die mathematischen Werkzeuge, die für die Bildverarbeitungsaufgaben benötigt werden, werden eingeführt. Das Modul beginnt mit einem Überblick über Vorverarbeitungsalgorithmen, wie zum Beispiel Streustrahlkorrektur für

		Röntgenbilder, Kantenerkennung, Superresolution und kantenerhaltende Rauschunterdrückung. Das zweite Kapitel beschreibt die automatische Bildanalyse mit Merkmalsdeskriptoren, Punkterkennung und Segmentierung mit Bottom-up-Algorithmen wie dem Random-Walker oder Top-Down-Ansätzen wie aktiven Formmodellen. Darüber hinaus deckt die Vorlesung auch geometrische Kalibrierungsalgorithmen zur Einzelansicht-Kalibrierung, Epipolargeometrie und Faktorisierung ab. Der letzte Teil des Moduls deckt nicht-starre Registrierung auf der Grundlage von Variationsmethoden und bewegungskompensierter Bildrekonstruktion ab.
6	**Lernziele und Kompetenzen**	English Version: The participants • summarize the contents of the lecture. • apply pre-processing algorithms such as scatter correction and edge-preserving filtering. • extract information from images automatically by image analysis methods such as key point detectors and segmentation algorithms. • calibrate projection geometries for single images and image sequences using the described methods. • develop non-rigid registration methods using variational calculus and different regularizers. • adopt algorithms to new domains by appropriate modifications. Deutsche Version: Die Teilnehmenden • fassen die Inhalte der Vorlesung zusammen. • wenden Vorverarbeitungsalgorithmen wie Streustrahlkorrektur und kantenerhaltende Filterung an. • extrahieren automatisch Informationen aus Bildern, indem sie Bildanalyseverfahren wie Punktdetektoren und Segmentierungsalgorithmen verwenden. • kalibrieren Projektionsgeometrien für Einzelbilder und Bildsequenzen mit den beschriebenen Methoden. • entwickeln nicht-starre Registrierungsmethoden mit Hilfe von Variationsrechnung und unterschiedlichen Regularisierern. • wenden Algorithmen auf neue Modalitäten durch entsprechende Änderungen im Algorithmus an.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1

9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich/mündlich (60 Minuten)
11	**Berechnung der Modulnote**	schriftlich/mündlich (100%)
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 0 h Eigenstudium: 150 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 93130	**Konzeptionelle Modellierung** (Conceptual modelling)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übungen zu Konzeptionelle Modellierung (2 SWS) Vorlesung: Konzeptionelle Modellierung (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	David Haller Prof. Dr.-Ing. Richard Lenz	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Richard Lenz
5	**Inhalt**	• Grundlagen der Modellierung • Datenmodellierung am Beispiel Entity-Relationship-Modell • Modellierung objektorientierter Systeme am Beispiel UML • Relationale Datenmodellierung und Anfragemöglichkeiten • Grundlagen der Metamodellierung • XML • Multidimensionale Datenmodellierung • Domänenmodellierung und Ontologien
6	**Lernziele und Kompetenzen**	Die Studierenden: • definieren grundlegende Begriffe aus der Datenbankfachliteratur • erklären die Vorteile von Datenbanksystemen • erklären die verschiedenen Phasen des Datenbankentwurfs • benutzen das Entity-Relationship Modell und das erweiterte Entity-Relationship Modell zur semantischen Datenmodellierung • unterscheiden verschiedene Notationen für ER-Diagramme • erläutern die grundlegenden Konzepte des relationalen Datenmodells • bilden ein gegebenes EER-Diagramm auf ein relationales Datenbankschema ab • erklären die Normalformen 1NF, 2NF, 3NF, BCNF und 4NF • definieren die Operationen der Relationenalgebra • erstellen Datenbanktabellen mit Hilfe von SQL • lösen Aufgaben zur Datenselektion und Datenmanipulation mit Hilfe von SQL • erklären die grundlegenden Konzepte der XML • erstellen DTDs für XML-Dokumente • benutzen XPATH zur Formulierung von Anfragen an XML-Dokumente • definieren die grundlegenden Strukturelemente und Operatoren des multidimensionalen Datenmodells • erklären Star- und Snowflake-Schema • benutzen einfache UML Use-Case Diagramme • benutzen einfache UML-Aktivitätsdiagramme • erstellen UML-Sequenzdiagramme • erstellen einfache UML-Klassendiagramme • erklären den Begriff Meta-Modellierung • definieren den Begriff der Ontologie in der Informatik • definieren die Begriffe RDF und OWL

7	**Voraussetzungen für die Teilnahme**	Gewünscht "Algorithmen und Datenstrukturen" und "Grundlagen der Logik und Logikprogrammierung"
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Masterprüfung Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222 Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• Elmasri, Ramez, and Sham Navathe. Grundlagen von Datenbanksystemen. Pearson Deutschland GmbH, 2009. - ISBN-10: 9783868940121 • Alfons Kemper, Andre Eickler: Datenbanksysteme : Eine Einführung. 6., aktualis. u. erw. Aufl. Oldenbourg, März 2006. - ISBN-10: 3486576909 • Bernd Oestereich: Analyse und Design mit UML 2.1. 8. Aufl. Oldenbourg, Januar 2006. - ISBN-10: 3486579266 • Ian Sommerville: Software Engineering. 8., aktualis. Aufl. Pearson Studium, Mai 2007. - ISBN-10: 3827372577 • Horst A. Neumann: Objektorientierte Softwareentwicklung mit der Unified Modeling Language. (UML). Hanser Fachbuch, März 2002. - ISBN-10: 3446188797 • Rainer Eckstein, Silke Eckstein: XML und Datenmodellierung. Dpunkt Verlag, November 2003. - ISBN-10: 3898642224

1	**Modulbezeichnung** 330467	**Multimedia Security** (Multimedia security)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Multimedia Security Exercises (2 SWS) Vorlesung: Multimedia Security (2 SWS)	- -
3	Lehrende	PD Dr.Ing. Christian Rieß	

4	**Modulverantwortliche/r**	
5	**Inhalt**	This lecture covers a variety of security-related topics around multimedia data. In particular, the lecture presents algorithms and key results from the past 15 years in multimedia security, including topics on image forensics, steganography, watermarking, and biometrics. Selected algorithms are implemented and tested by the participants. It is helpful to bring some knowledge in signal processing or pattern recognition. It is also helpful to be not afraid from equations. Tentative list of topics and algorithms: • Image forensics for manipulation detection in digital media. Statistical and physics-based features for manipulation detection. Detecting traces of manipulations versus validating image authenticity. • Blind source attribution: was an image or video captured with a particular camera? • Steganography for covert communication. Fundamental algorithms, when can their application be detected? • Watermarking for copyright protection in images/video. Fundamental algorithms, and their security. • Biometric features for person re-identification, and practical concerns on their implementation.
6	**Lernziele und Kompetenzen**	Fachkompetenz Verstehen Die Studierenden fassen die wesentlichen Fragestellungen auf dem Gebiet der Multimediasicherheit zusammen (The participants summarize the relevant questions within the field of multimedia security). Die Studierenden nennen und erklären die wesentlichen Fachbegriffe aus den Teilgebieten der Multimediasicherheit (The participants name and explain relevant terms from the subfields of multimedia security). Evaluieren (Beurteilen) Die Studierenden bewerten die Eignung der vorgestellten bildforensischen Algorithmen für ein gegebenes Untersuchungsszenario (The participants evaluate the suitability of the presented image forensics algorithms for a given examination scenario).

		Erschaffen Die Studierenden implementieren kurze Beispielsprogramme für ausgewählte Algorithmen der Multimediasicherheit (The participants implement short example programs for selected algorithms of multimedia security). Lern- bzw. Methodenkompetenz Die Studierenden implementieren ausgewählte Methoden in der Programmiersprache C++ (The participants implement selected methods in the C++ programming language). Sozialkompetenz Die Studierenden implementieren und diskutieren Beispielsmethoden in Gruppenarbeit (The participants implement and discuss the example method in groups). Die Studierenden diskutieren die gesellschaftlichen Auswirkung von Multimediasicherheit am Beispiel aktueller Probleme (The participants discuss multimedia security's impact on society using current issues).
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 120 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	begleitend zu der Veranstaltung: • Husrev Sencar, Nasir Memon (Editors): "Digital Image Forensics. There is More to a Picture than Meets the Eye", Springer 2013. • Hany Farid: "Photo Forensics", MIT Press, 2016.

1	**Modulbezeichnung** 44120	**Pattern Analysis** (Pattern analysis)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Pattern Analysis (3 SWS) Übung: Pattern Analysis Programming (1 SWS)	3,75 ECTS 1,25 ECTS
3	Lehrende	PD Dr.Ing. Christian Rieß Dalia Rodriguez Salas	

4	**Modulverantwortliche/r**	PD Dr.Ing. Christian Rieß
5	**Inhalt**	This module introduces the design of pattern analysis systems as well as the corresponding fundamental mathematical methods. The topics comprise: • clustering methods: soft and hard clustering • classification and regression trees and forests • parametric and non-parametric density estimation: maximum-likelihood (ML) estimation, maximum-a-posteriori (MAP) estimation, histograms, Parzen estimation, relationship between folded histograms and Parzen estimation, adaptive binning with regression trees • mean shift algorithm: local maximization using gradient ascent for non-parametric probability density functions, application of the mean shift algorithm for clustering, color quantization, object tracking • linear and non-linear manifold learning: curse of dimensionality, various dimensionality reduction methods: principal component analysis (PCA), multidimensional scaling (MDS), isomaps, Laplacian eigenmaps • Gaussian mixture models (GMM) and hidden Markov models (HMM): expectation maximization algorithm, parameter estimation, computation of the optimal sequence of states/ Viterbi algorithm, forward-backward algorithm, scaling • Markov random fields (MRF): definition, probabilities on undirected graphs, clique potentials, Hammersley-Clifford theorem, inference via Gibbs sampling and graph cuts Das Modul führt in das Design von Musteranalysesystemen sowie die zugrundeliegenden mathematischen Methoden ein. Die Vorlesung umfasst im Einzelnen: • Clustering-Methoden: Soft- und Hard-Clustering • Klassifikations- und Regressionsbäume/-wälder • parametrische und nicht-parametrische Dichteschätzung: Verfahren sind ML- und MAP-Schätzung, Histogramme, Parzenschätzung, Zusammenhang gefaltete Histogramme und Parzenschätzung, adaptives Binning mit Regressionsbäumen. • 'Mean Shift'-Algorithmus: lokale Maximierung durch Gradientenaufstieg bei nicht-parametrischen Dichtefunktionen, Anwendungen des 'Mean Shift'-Algorithmus zum Clustering, Farbquantisierung und Objektverfolgung

		• Linear and Non-Linear Manifold Learning: Curse of Dimensionality, Verschiedene Methode zur Dimensionsreduktion: Principal Component Analysis (PCA), Multidimensionsional Scaling (MDS), Isomap, Laplacian Eigenmaps • Gaußsche Mischverteilungsmodelle (GMM) und Hidden-Markov-Modelle (HMM): 'Expectation Maximization'-Algorithmus, Parameterschätzung, Bestimmung der optimalen Zustandsfolge/Viterbi-Algorithmus, Vorwärts-Rückwärts-Algorithmus, Skalierung • Markov-Zufallsfelder: Definition, Wahrscheinlichkeiten auf ungerichteten Graphen, Cliquen-Potenziale, Hammersley-Clifford-Theorem, Inferenz mit Gibbs-Sampling und Graph Cuts
6	**Lernziele und Kompetenzen**	The students • explain the discussed methods for classification, prediction, and analysis of patterns, • compare and analyze methods for manifold learning and select a suited method for a given set of features and a given problem, • compare and analyze methods for probability density estimation and select a suited method for a given set of features and a given problem, • apply non-parametric probability density estimation to pattern analysis problems, • apply dimensionality reduction techniques to high-dimensional feature spaces, • explain statistic modeling of feature sets and sequences of features, • explain statistic modeling of statistical dependencies, • implement presented methods in Python, • supplement autonomously the mathematical foundations of the presented methods by self-guided study of the literature, • discuss the social impact of applications of pattern analysis solutions. Die Studierenden • erläutern die behandelten Methoden zur Klassifikation, Vorhersage und Analyse von Mustern, • vergleichen und analysieren Methoden des Manifold Learning und wählen für eine vorgegebene Fragestellung eine geeignete Methode aus, • vergleichen und analysieren Methoden zur Dichteschätzung und wählen für eine vorgegebene Fragestellung eine geeignete Methode aus, • wenden nicht-parametrische Dichteschätzung auf Probleme der Musteranalyse an, • wenden Dimensionsreduktion bei hochdimensionalen Merkmalsräumen an,

		• erläutern statistische Modellierung von Merkmalsmengen und Merkmalsfolgen, • erklären statistische Modellierung abhängiger Größen, • implementieren vorgestellte Verfahren in Python. • ergänzen eigenständig mathematische Grundlagen der präsentierten Methoden durch selbstbestimmtes Studium der Literatur • diskutieren die gesellschaftlichen Auswirkungen von Anwendungen der Musteranalyse
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Variabel (60 Minuten)
11	**Berechnung der Modulnote**	Variabel (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Begleitende Literatur / Accompanying literature: • C. Bishop: Pattern Recognition and Machine Learning, Springer Verlag, Heidelberg, 2006 • T. Hastie, R. Tibshirani und J. Friedman: The Elements of Statistical Learning, 2nd Edition, Springer Verlag, 2009 • A. Criminisi and J. Shotton: Decision Forests for Computer Vision and Medical Image Analysis, Springer, 2013

1	**Modulbezeichnung** 43722	**Scientific Visualization** (keine englischsprachige Modulbezeichnung hinterlegt!)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Tutorials to Scientific Visualization (2 SWS) Vorlesung: Scientific Visualization (2 SWS)	- 5 ECTS
3	Lehrende	Prof. Dr.-Ing. Tobias Günther Xingze Tian	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Tobias Günther
5	**Inhalt**	The amount of data, generated in the pursuit of scientific discovery, keeps rapidly increasing across all major scientific disciplines. How can we make sense of large, time-dependent, high-dimensional and multi-variate data? This lecture provides an introduction into scientific visualization. Throughout the course, we cover the fundamental perception basics needed to convey information accurately. After categorizing different data types based on their dimensionality, we dive deeper into specific techniques for scalar, vector and tensor valued data. The lecture covers the following topics: • visualization design basics (data abstraction, visual encoding of information), • a review of scalar and vector calculus (differential properties, extremal and critical points), • data structures and data acquisition techniques (grids, interpolation, and differentiation), • indirect volume visualization (marching cubes and contour trees), • direct volume visualization (ray marching and Monte Carlo rendering), • elementary and line-based flow visualization (numerical integration, seeding, rendering), • surface-based flow visualization (integration, selection, rendering), • topology-based flow visualization (topological skeleton, bifurcations, feature flow fields), • feature-based flow visualization (vortices, material boundaries, Lagrangian coherent structures), • advanced methods (tensor visualization, uncertainty, ensembles)
6	**Lernziele und Kompetenzen**	Students are able to: • use perception basics to select appropriate visualization methods • classify data and select appropriate visualization techniques • calculate differential properties of scalar and vector fields • identify features in scalar and vector-valued data • implement numerical extraction algorithms • learn the advantages and disadvantages of common visualization techniques

7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot der Technischen Fakultät oder der Naturwissenschaftlichen Fakultät Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Variabel
11	**Berechnung der Modulnote**	Variabel (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

Wahlmodule aus dem Angebot von EEI und Informatik

1	**Modulbezeichnung** 151664	**Advanced Communication Networks** (Advanced communication networks)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Advanced Communication Networks - Tutorial (0 SWS) Vorlesung: Advanced Communication Networks (4 SWS)	- 5 ECTS
3	Lehrende	Walid Ghanem Prof. Dr. Laura Cottatellucci	

4	**Modulverantwortliche/r**	Prof. Dr. Laura Cottatellucci
5	**Inhalt**	Telecommunications have become ubiquitous in daily life and wireless networks play a fundamental role thanks to their capability to support mobility. In a wireless communication, the concept of link does not exist. Users radiate energy and communicate through the superposition of each others transmissions which creates interference. Compared to wireline networks this scenario is extremely challenging but also offers unpredictable opportunities in the development of new technologies (massive MIMO, cognitive radio, etc.) and exploitation of new features, e.g., opportunistic communications and multiuser diversity. The exponentially increasing request of higher and higher throughput is satisfied densifying users and access points per unit area and allowing more and more interference while adopting advanced techniques and innovative resource allocation to mitigate the detrimental effects of interference. Objective of this course is to introduce the student to advanced techniques for coordinated medium access control and radio resource management in cellular systems. Power allocation, rate adaptation and scheduling will be discussed both in centralized and distributed settings. Some mathematical methods play a fundamental role in resource allocation, namely, classical Perron-Frobenius theory for nonnegative matrices, convex and nonconvex constrained optimization, distributed optimization and game theory. The course introduces the student to such methods and exemplifies their application to various resource allocation problems. Additionally, the course addresses relevant aspects of resource allocation in wireless networks such as fairness and cross-layer design. *Technical Content* • Properties and challenges of the wireless medium. • Basic concepts of communication networks: the layered architecture. • Evolution of wireless cellular network architectures: From Global System for Mobile to Advanced-Long Term Evolution. • Multiple Access Schemes: CSMA variants, TDMA, FDMA, CDMA, OFDMA, SC-FDMA, SDMA. • Uplink-downlink duality. • Opportunistic scheduling and multiuser diversity.

		• Advanced concepts: small cells and heterogeneous networks, relaying and cooperation, network coding, cognitive radio networks. • Basics of resource allocation: power allocation, rate adaptation, and scheduling. • Classical resource allocation techniques: Centralized and distributed power control based on the Perron-Frobenius theorem. • Fundamentals of convex constrained optimization and application to resource allocation. • Resource allocation and fairness. • Fundamentals of nonconvex optimization and relaxation techniques. • Applications of nonconvex optimization to resource allocation. • Fundamentals of distributed optimization and applications to resource allocation. • Fundamental concepts of game theory. • Resource contention via game theoretical methods.
6	**Lernziele und Kompetenzen**	The student • Describes and/or recognizes wireless channel models. • Criticizes the limits of a layered architecture in wireless systems. • Defends the use of cross-layer design in wireless network. • Appraises and compares the distribution of functionalities in network entities for different architectures. • Argue on the pros and contras of different multiple access schemes according to various criteria (e.g. spectral efficiency, power efficiency, robustness to interference). • Compares and contrasts micro-diversity and various macro-diversity schemes. • Computes the total rate of SDMA with various receivers. • Relates the multiple access in uplink to broadcasting in downlink and justifies the concept of uplink-downlink duality. • Uses uplink-downlink duality to design a precoder and allocate power. • Contrasts multiple access in uplink and broadcasting in downlink in terms of channel state acquisition both for TDD and FDD transmission. • Uses multiuser diversity for opportunistic scheduling. • Compares multiuser diversity for users having identical and different channel statistics. • Contrasts opportunistic scheduling in terms of channel state acquisition and feedback both for uplink and downlink and for both FDD and TDD transmission schemes. • Appraises the impact of multiple antennas on opportunistic scheduling. • Analyses different settings with interference in small cells and designs countermeasures. • Categorizes relaying schemes in LTE.

		• Analyses performance of relaying schemes. • Argues on possible improvements of relaying schemes via network coding and physical layer network coding. • Uses the Perron-Frobenious theorem to allocate power in a centralized manner. • Judges the feasibility of a power control problems and formulates alternative approaches in case of unfeasibility. • Uses the Perron-Frobenious theorem to design a distributed power control scheme. • Judges the convergences of distributed power control based on the Perron-Frobenius theorem and appraises the robustness of asynchronous power control. • Applies techniques of convex optimization to discriminate convex problems and determine necessary and/or sufficient conditions for global optimality. • Judges the applicability of KKT conditions and duality. • Uses KKT conditions to solve convex optimization problems. • Uses duality to solve convex optimization problems. • Applies convex optimization to resource allocation in wireless communications. • Compares different definitions of fairness and applies them to rate allocation. • Appraises the effect of channel knowledge at the transmitter on different fairness criteria. • Applies KKT conditions for opportunistic user scheduling. • Describes a proportional fair algorithm for opportunistic scheduling. • Applies relaxation to nonconvex quadratic constrained quadratic programming. • Formulates resource allocation problems as constrained optimization programming. • Contrasts various distributed optimization methods. • Applies the concept of best response to determine Nash equilibria. • Argues about existence and uniqueness of Nash equilibria. • Assesses if a given game is a potential game and solves it. • Defends the concept of Pareto optimality in resource allocation. • Contrasts the concepts of pure and mixed strategies in game theory. • Uses coupled constrained concave game to allocate powers in heterogeneous networks.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222

10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 869547	**Advanced Networking LEx** (Advanced networking LEx)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Advanced Networking (2 SWS) Übung: Advanced Networking Exercises (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Dr.-Ing. Kai-Steffen Hielscher Dr.-Ing. Peter Bazan	

4	**Modulverantwortliche/r**	Dr.-Ing. Kai-Steffen Hielscher
5	**Inhalt**	Für die durch Big Data, Cloud Computing, Internet-of-Things (IoT) und mobile Endgeräte hervorgerufenen Herausforderungen sind neue Architekturen für Rechnernetze entstanden: Software-Defined-Networking (SDN) entkoppelt die Data Plane (Weiterleitung von Paketen, auf handelsüblicher Hardware) und die Control Plane (Steuerung, auf leistungsfähigen Plattformen) und bietet offene Programmierschnittstellen; Network Functions Virtualization (NFV) erweitert Konzepte zur Server- und Netzwerkvirtualisierung, so dass bisher auf proprietärer Hardware ausgeführte Netzwerkfunktionen (wie z.B. Routing) ebenfalls virtualisiert und auf handelsüblicher Hardware ausgeführt werden können. Die Vorlesung stellt hinter diesen Technologien stehende Konzepte und Standards vor und zeigt, wie sie für Rechenzentren, für Cloud- und Fog-Computing und für IoT-Anwendungen eingesetzt werden können. Content: New architectures for computer networks have emerged to meet the challenges posed by Big Data, Cloud Computing, Internet-of-Things (IoT) and mobile devices: Software-Defined-Networking (SDN) decouples the data plane (forwarding of packets, on commercially available hardware) and the control plane (control, on powerful platforms) and offers open programming interfaces; Network Functions Virtualization (NFV) extends concepts for server and network virtualisation, so that network functions (such as routing) previously executed on proprietary hardware can also be virtualised and executed on commercially available hardware. The lecture introduces concepts and standards behind these technologies and shows how they can be used for data centres, for cloud and fog computing and for IoT applications.
6	**Lernziele und Kompetenzen**	Fachkompetenz Verstehen Die Studierenden erlangen Verständnis der grundlegenden Konzepte von • Software Defined Networking • Network Function Virtualization

		• Internet of Things • Cloud Computing. Anwenden Die Studierenden wenden die Erkenntisse in Übungsaufgaben an. Erschaffen Die Studierenden erstellen eigene Laborkofigurationen zu • Software Defined Networking • Internet of Things. Competences: Professional competence Understanding Students will gain an understanding of the basic concepts of • Software Defined Networking • Network Function Virtualization • Internet of Things • Cloud Computing. Apply Students apply the knowledge gained in exercises. Create The students create their own laboratory configurations on • Software Defined Networking • Internet of Things.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20191 Wahlpflichtmodul aus INF im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio

11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	William Stallings: Foundations of Modern Networking - SDN, QoE, IoT, and Cloud; Pearson

1	**Modulbezeichnung** 93172	**AI-enabled Wireless Networks** (keine englischsprachige Modulbezeichnung hinterlegt!)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: AI-enabled wireless networks (2 SWS) Übung: Literature review on the application of machine learning in wireless networks (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Dr.-Ing. Mehdi Harounabadi	

4	**Modulverantwortliche/r**	Dr.-Ing. Mehdi Harounabadi
5	**Inhalt**	This course introduces machine learning algorithms such as supervised, unsupervised, reinforcement, deep, and federated learning and their application in the next generation wireless and mobile networks. Different ML use cases are explained which solve problems in different layers of the protocol stack from the physical layer to the application layer. The course includes the following topics: 1.Introduction to machine learning algorithms 2.Python programming language and its ML tools 3.AI-enabled wireless and mobile networks 3.1Cellular networks and ML use cases 3.1.1History of 2G to 4G, 5G and 6G vision 3.1.2ML use cases in physical, MAC and higher layers 3.25G-V2X (cellular-V2X) and ML use cases 3.2.1Sidelink communication as the key enabler 3.2.25G-V2X features and use cases 3.2.3ML use cases in 5G-V2X 3.3Intelligent wireless networks 3.3.1Cognitive radio networks 3.3.2ML use case in wireless networks 4.Standardization activities on AI-enabled wireless networks 4.1.13GPP and 5GAA 4.1.2ETSI Zero touch networks

		Exercises: Literature review on the application of machine learning in wireless networks The exercise of this course includes a literature review research project where students work individually on a relevant topic. The steps to accomplish the research project are as follows: A.Select a topic relevant to the application of ML in wireless networks and register it by email B.Search for the relevant papers and make a list of papers C.Study the papers and prepare a summary D.Present the outcomes Each student should present her/his research study in an intermediate and a final presentation. A summary paper should be written following the "survey papers guideline" using IEEE format. The grade of the research project will be considered as a "Bonus point" (up to 20%) for the final grade.
6	**Lernziele und Kompetenzen**	The students will be able to gain the following competencies after the successful completion of the course: • Have knowledge on machine learning algorithms, current and the next generation wireless and mobile networks and their use cases • To know how to develop machine learning algorithms in Python • Know the most important problems in wireless and mobile networks which can be solved by machine learning algorithms • Have an insight on the exiting work relevant to the topic of the course • Learn the procedure of a scientific research study and publication of the outcome
7	**Voraussetzungen für die Teilnahme**	*Motivation:* Rapid growth in the number of connected wireless nodes such as mobile phones, low power IoT devices, connected vehicles, etc. will expand the scale of the next generation of wireless and mobile networks. Moreover, the foreseen use cases like connected autonomous vehicles, smart homes and cities, ultra-fast and reliable industrial wireless networks, etc. will require ultra-low latency and highly reliable communication. Existing and traditional algorithms are not feasible for the optimization and management of such networks to fulfill the requirements of the emerging use cases due to their high complexity, high dynamicity, and the massive amount of the generated data by connected devices. Recently, artificial intelligence (AI) is planned to be utilized as a new paradigm for the

		design, development and optimization of the next generation wireless and mobile networks. Machine learning (ML) as a subset of AI will be applied to develop intelligent wireless nodes and infrastructures to address the demands of future use cases.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot der Technischen Fakultät oder der Naturwissenschaftlichen Fakultät Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich (30 Minuten)
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Dahlman, Erik, Stefan Parkvall, and Johan Skold. 5G NR: The next generation wireless access technology. Academic Press, 2020. Sun, Yaohua, et al. "Application of machine learning in wireless networks: Key techniques and open issues." IEEE Communications Surveys & Tutorials 21.4 (2019): 3072-3108. Harounabadi, Mehdi, et al. "V2X in 3GPP Standardization: NR Sidelink in Release-16 and Beyond." IEEE Communications Standards Magazine 5.1 (2021): 12-21. Xie, Junfeng, et al. "A survey of machine learning techniques applied to software defined networking (SDN): Research issues and challenges." IEEE Communications Surveys & Tutorials 21.1 (2018): 393-430.

1	**Modulbezeichnung** 96500	**Analoge elektronische Systeme** (Analogue electronic systems)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Analoge elektronische Systeme (3 SWS) Übung: Übungen zu Analoge elektronische Systeme (1 SWS)	5 ECTS -
3	Lehrende	Torsten Reißland Prof. Dr.-Ing. Robert Weigel Christof Pfannenmüller	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Robert Weigel
5	**Inhalt**	• Feldeffekttransistor • Verstärker, Leistungsverstärker • Nichtlinearität und Verzerrung • Filtertheorie • Realisierung von Filtern • Intrinsisches Rauschen (Konzepte) • Physikalische Rauschursachen • Rauschparameter • Mischer • Oszillatoren • Phasenregelschleifen (PLLs)
6	**Lernziele und Kompetenzen**	• Die Studierenden erlangen Kenntnisse um Rauscheffekte und Nichtlinearitäten in Analogschaltungen zu erklären • Die Studierenden verstehen die Ursachen verschiedener physikalischer Rauschprozesse und können diese klassifizieren • Die Studierenden erwerben die Fähigkeit zur Planung und Implementierung frequenzumsetzender Systeme mittels zugehöriger Frequenz- und Pegelpläne • Die Studierenden bewerten Hochfrequenzoszillatoren und stabilisierende PLL-Schaltungen • Die Studierenden untersuchen Messaufbauten zur Charakterisierung von Rauschen und Nichtlinearitäten • Die Studierenden analysieren den inneren Aufbau von Leistungsverstärkern auf Basis von Transistorschaltungen • Die Studierenden sind in der Lage komplexe Analogschaltungen simulativ und analytisch zu untersuchen und deren Verhalten im Groß- und Kleinsignalbereich zu charakterisieren • Die Studierenden führen Filterentwürfe durch und bestimmen deren Amplituden- und Phasengang • Die Studierenden können bei auftretenden Problemen selbstständig mit Hilfe weitergehender Literatur oder durch Diskussion in der Gruppe Lösungen erarbeiten
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191

		Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 965820	**Approximate Computing** (Approximate computing)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Exercises to Approximate Computing (0 SWS) Vorlesung: Approximate Computing (2 SWS)	- 5 ECTS
3	Lehrende	Chetana Pradhan Pierre-Louis Sixdenier Prof. Dr.-Ing. Jürgen Teich Prof. Dr. Oliver Keszöcze	

4	**Modulverantwortliche/r**	Joachim Falk Prof. Dr. Oliver Keszöcze Prof. Dr.-Ing. Jürgen Teich
5	**Inhalt**	Approximate Computing denotes a quite young research area that exploits the fact and capability of many applications and systems to tolerate imprecision and/or inexactness of computed results. Prominent areas of applications and novel techniques of computing approximate rather than exact results have brought up new implementations either at hardware and/or software levels for important emergent workloads such as searching, mining, image processing, and data retrieval. Although hardware technology is improving at a fast pace, energy and power are becoming more and more important constraints apart from exactly computing results in an acceptable amount of time. The main goals of approximate computing techniques are therefore to exploit the possible trade-off between power/energy consumption, accuracy, performance, and/or cost, e.g., utilized hardware resources. The purpose of the course approximate computing is to instruct students about the main ideas and concepts of approximate computing. This includes analyzing the trade-off between energy consumption, accuracy, run-time and hardware costs, concrete approximating techniques (e.g. approximate hardware synthesis, approximating algorithms) as well as theoretical background (determining the computational error and its complexity).
6	**Lernziele und Kompetenzen**	Fachkompetenz - Wissen • The students know the principles and benefits of Approximate Computing and when it is applicable. • The students know multiple error metrics and their semantic meaning. Fachkompetenz - Verstehen • The students understand the difference between the error metrics. • The students understand the principle of function falsification. • The students can apply the presented approximation techniques.

		Fachkompetenz - Anwenden • The students are capable of choosing the appropriate approximation technique based on given requirements.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20191 Wahlpflichtmodul aus INF im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Weitere Informationen: https://www.cs12.tf.fau.de/lehre/lehrveranstaltungen/vorlesungen/approximate-computing

1	**Modulbezeichnung** 96010	**Architekturen der digitalen Signalverarbeitung** (Architectures for digital signal processing)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übungen zu Architekturen der Digitalen Signalverarbeitung (2 SWS) Vorlesung: Architekturen der Digitalen Signalverarbeitung (2 SWS)	- 5 ECTS
3	Lehrende	Torsten Reißland Prof. Dr.-Ing. Georg Fischer	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Georg Fischer
5	**Inhalt**	Inhalt: • Basis-Algorithmen der Signalverarbeitung (FFT, Fensterung, Digitale FIR- und IIR-Filter) • Nichtideale Effekte bei Digitalfiltern (Quantisierung der Filterkoeffizienten, Quantisierte Arithmetik) • CORDIC-Architekturen • Architekturen für Multiratensysteme (Abtastratenumsetzer) • Architekturen digitaler Signalgeneratoren • Maßnahmen zur Leistungssteigerung (Pipelining) • Architekturen digitaler Signalprozessoren • Anwendungen Content: • Basic algorithms of signal processing (FFT, windowing, digital FIR and IIR-filters) • Non-idealities of digital filters (quantization of filter coefficients, fixed-point arthmetic) • CORDIC-architectures • Architectures of systems with multiple sampling rates (conversion between different sampling rates) • Digital signal generation • Measures of performance improvement (pipelining) • Architecture of digital signal processors • Applications
6	**Lernziele und Kompetenzen**	Die Studierenden erlangen Grundlagenkenntnisse der Signaltheorie und können zeit- und wertkontinuierliche sowie zeit- und wertdiskrete Signale im Zeit- und Frequenzbereich definieren und erklären Die Studierenden sind in der Lage, ein klassisches Echtzeitsystem zur digitalen Signalverarbeitung konzeptionieren und die Einzelkomponenten nach den Anforderungen zu dimensionieren Die Studierenden erlangen einen Überblick über Vor- und Nachteile analoger sowie digitaler Signalverarbeitung Die Studierenden verstehen die Theorie der Fourier-Transformation und sind in der Lage, die Vorteile der Fast-Fourier-Transformation in der digitalen Signalverarbeitung zu verstehen und anzuwenden

		Die Studierenden können digitale Filter dimensionieren und beurteilen ===Englisch=== Students • can obtain fundamentals of signal theory and can define as well time-comtinous and value-continous as time-discrete and value-discrete signals in time and frequency domain • can construct a realtime digital signal processing system and dimension its components according requirements • can review pros and cons of analogue versus digitzal signal processing • can apply fourier transformation and illustrate the advantages of fast fourier transformation in the context of digital signal processing • can dimension digital filters and evaluate their performance
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 947709	**Auditory Models** (Auditory models)	**2,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Auditory Models (2 SWS)	2,5 ECTS
3	Lehrende	Prof. Dr.-Ing. Bernd Edler	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Bernd Edler
5	**Inhalt**	• Main components of the human auditory system • Common models • Mechanical models • Physiological models • Psychoacoustic models • Applications (hearing aids, audio coding, . . .)
6	**Lernziele und Kompetenzen**	Goals • Students understand the structure and function of the human auditory system • Students gain deeper insight into psychoacoustic phenomena, such as masking, directional and spatial hearing • Students implement and evaluate perceptual models for various applications • Students collaborate with scientists in the fields of audiology and neuroscience
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 96875	**Ausgewählte Kapitel der Audiodatenreduktion** (keine englischsprachige Modulbezeichnung hinterlegt!)	**2,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Ausgewählte Kapitel der Audiodatenreduktion (2 SWS)	2,5 ECTS
3	Lehrende	Prof. Dr.-Ing. Jürgen Herre	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Jürgen Herre
5	**Inhalt**	Perceptual audio coding is ubiquitous in modern life (mp3 players, mobile phones, DVD players, computers, ...) Based on related classes (esp. Speech and Audio Processing"), this lecture aims at deepening the understanding of modern algorithms for perceptual source coding of audio. It includes an overview of the most relevant standardized coders, starting with MPEG-1 (incl. mp3) via MPEG-4 all the way to the most recent MPEG Audio standard. The significant algorithms are discussed and new approaches are described. The selected topics include: Efficient coding of several audio channels / parametric multi- channel coding Typical coding artifacts; subjective and objective quality assessment Scalable audio coding Bandwidth extension Semi-parametric audio coding Low-delay audio coding The lecture includes a number of demonstrations and audio examples to illustrate the discussed algorithms.
6	**Lernziele und Kompetenzen**	• Wissen - Die Studenten kennen die Hauptkomponenten eines gehörangepassten Audiocodecs, sowie die wichtigsten Algorithmen, Codierstrategien und Bewertungsmethoden. Weiterhin kennen sie die Terminologie und gängige Abkürzungen aus diesem Kontext. • Verstehen - Die Studenten verstehen, wie Designentscheidungen in Audiocodecs die letztendlich erreichte Audioqualität beeinflussen, verstehen die gebräuchlichsten Tools aus dem Bereich der gehörangepasste Audiocodierung und wie verschiedene Anwendungsszenarien das Coderdesign bestimmen. • Anwenden - Die Studenten können übliche mathematische Analysemethoden verwenden, um einfache Coder-Componenten zu beschreiben und gegebenenfalls zu modifizieren.

		• Analysieren - Die Studenten können Audiocodierungs-Standards und wahrnehmungsbasierte Messwerkzeuge dazu analysieren um die zugrundeliegenden Konzepte und Anforderungen zu erfassen. • Evaluieren (Beurteilen) - Die Studenten können Audiocodierungs-Standards und wahrnehmungsbasierte Messwerkzeuge evaluieren um zu beurteilen, welcher Standard bzw. welches Messwerkzeug das passendste ist für einen bestimmten Anwendungsfall. • Synthese - Die Studenten können eine Liste von Anforderungen und Bewertungskriterien für Audiocodecs zusammenstellen für gewünschte Anwendungsfälle. • Lern- bzw. Methodenkompetenz - Die Studenten hinterfragen bestehende Ansätze hinsichtlich ihrer Anwendbarkeit in der Praxis.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 820947	**Betriebssysteme (Vorlesung mit erweiterten Übungen)** (Operating systems (lectures with extend exercises))	**7,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Betriebssysteme (2 SWS) Übung: Betriebssysteme - Rechnerübungen (0 SWS) Übung: Betriebssysteme - Erweiterte Übungen (4 SWS)	2,5 ECTS - 5 ECTS
3	Lehrende	Dr.-Ing. Volkmar Sieh Bernhard Heinloth Dustin Nguyen Phillip Raffeck	

4	**Modulverantwortliche/r**	Dr.-Ing. Volkmar Sieh
5	**Inhalt**	Inhalt des Moduls ist Vermittlung grundlegender Konzepte, Methoden und Techniken von Betriebssystemen aus dem Blickwinkel einer Betriebssystementwicklerin. Im Rahmen der Übungen entwickeln die Studierenden in einem "bottom-up" Entwurf- und Entwicklungsprozess ihr eigenes Mehrkernbetriebssystem für die IA-32 Platform, ausgehend von der "nackten Hardware" über grundlegende Ein-Ausgabemöglichleiten, Unterbrechungsbearbeitung bis hin zu quasi- und echtparalleler Programmausführung.
6	**Lernziele und Kompetenzen**	Studierende, die das Modul erfolgreich abgeschlossen haben: - erläutern den Startvorgang eines Rechensystems am Beispiel eines x86/x86_64-PCs. - beschreiben die spezifischen Herausforderungen bei der Softwareentwicklung für "bare metal". - beschreiben den Ablauf einer Unterbrechungsbehandlung von der Hardware bis zur (System-)software. - skizzieren Besonderheiten und Strategien der Unterbrechungsbehandlung in Hardware für Mehrkernsystemen am Beispiel des IA32-APICs. - diskutieren die Aufgabenteilung zwischen Hardware und Systemsoftware bei der Unterbrechungsbearbeitung. - unterscheiden die verschiedenen Typen von Kontrollflüssen in einem Betriebssystem anhand des Ebenenmodells. - unterscheiden harte, mehrstufige, und weiche Verfahren zur Unterbrechungssynchronisation in Betriebssystemen und können diese für ein Mehrkernsystem implementieren. - klassifizieren konkrete Konkurrenzsituationen anhand des Ebenenmodels und leiten daraus geeignete Synchronisationsmaßnahmen ab.

- schildern die IA32-Architektur und gängige PC-Technologie und deren Schnittstellen zur Systemsoftware.

- erläutern grundlegende Bausteine für die Implementierung von Quasi-Parallelität (Fortsetzungen, Koroutinen, Fäden) und grenzen diese gegeneinander ab.

- erläutern die Interaktionen zwischen Hardware, Übersetzer und Systemsoftware, die dabei zu beachten sind.

- entwickeln den Koroutinenwechsel für einen gegebene Architektur.

- erläutern die Implikationen von Quasi-Parallität auf das Ebenenmodell und die daraus abgeleiteten Synchronisationsmaßnahmen.

- beschreiben die Implementierung von (verdrängendem) Scheduling in einem Betriebssystem.

- analysieren das Zusammenspiel von Scheduling und Unterbrechungssynchronisation.

- nennen Kriterien und Dimensionen des Schedulings von Betriebsmitteln, insbesondere der CPU.

- erläutern die konkrete Umsetzung am Beispiel der Scheduler in Linux und Windows.

- unterscheiden grundlegende Möglichkeiten der Koordinierung und Synchronisation von Fäden (aktives/passives Warten, nichtverdrängbare kritische Abschnitte).

- entwickeln Mechanismen für die Synchronisation auf Fadenebene für Mehrkernsystemen.

- erklären die dabei zu beachtenden Synchronisationsprobleme (lost update, lost wakeup) und geeignete Gegenmaßnahmen.

- interpretieren die Bedeutung von Gerätetreibern in der Betriebssystempraxis.

- erläutern die Anforderungen an ein Treibermodell.

- vergleichen die Umsetzung von Treibermodellen in Windows und Linux.

- vergleichen grundlegende BS-Architekturen (Bibliothek, Monolith, Mikrokern, Exokern, Hypervisor) anhand fundamentaler Charakteristika (Robustheit, Performanz, Portierbarkeit) und Mechanismen.

		- schildern die grundlegenden Paradigmen zur Interprozesskommunikation in Betriebssystemen (speicherbasiert vs. nachrichtenbasiert). - erläutern die grundlegenden Primitiven dieser Verfahren. - skizzieren, wie unter Anwendung dieser Primitiven höhere Synchronisationskonstrukte implementiert werden (Monitore, Leser-/ Schreiber-Sperre). - illustrieren die Dualität der Paradigmen. - erschließen sich typische Probleme (Nebenläufigkeit, Compilerverhalten, Debuggen ohne dedizierte Hilfsmittel) und Fehlerquellen bei der hardwarenahen Softwareentwicklung für Mehrkernsysteme. - können in Kleingruppen kooperativ arbeiten. - können die ihre Entwurfs- und Implementierungsentscheidungen kompakt präsentieren und argumentativ vertreten. - reflektieren ihre Entscheidungen kritisch und leiten Alternativen ab. - können offen und konstruktiv mit Schwachpunkten und Irrwegen umgehen.
7	**Voraussetzungen für die Teilnahme**	• C/C++ (Übungsaufgaben werden in C++ implementiert) • Assembler (Grundkenntnisse)
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 90 h Eigenstudium: 135 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• Avi Silberschatz, Peter B. Galvin, Greg Gagne. Operating System Concepts. John Wiley & Sons. • Andrew S. Tanenbaum. Modern Operating Systems. Pearson.

		• William Stallings. Operating Systems: Internals and Design Principles. Prentice Hall. • Andrew S. Tanenbaum. Structured Computer Organization. Pearson.

1	**Modulbezeichnung** 150033	**Betriebssysteme (Vorlesung mit Übungen)** (Operating systems (lecture with exercises))	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Betriebssysteme (2 SWS) Übung: Betriebssysteme - Übungen (2 SWS) Übung: Betriebssysteme - Rechnerübungen (0 SWS)	2,5 ECTS 2,5 ECTS -
3	Lehrende	Dr.-Ing. Volkmar Sieh Bernhard Heinloth Dustin Nguyen Phillip Raffeck	

4	**Modulverantwortliche/r**	Dr.-Ing. Volkmar Sieh
5	**Inhalt**	Inhalt des Moduls ist Vermittlung grundlegender Konzepte, Methoden und Techniken von Betriebssystemen aus dem Blickwinkel einer Betriebssystementwicklerin. Im Rahmen der Übungen entwickeln die Studierenden in einem "bottom-up" Entwurf- und Entwicklungsprozess ihr eigenes Einkernbetriebssystem für die IA-32 Platform, ausgehend von der "nackten Hardware" über grundlegende Ein-Ausgabemöglichleiten, Unterbrechungsbearbeitung bis hin zu quasiparalleler Programmausführung.
6	**Lernziele und Kompetenzen**	Studierende, die das Modul erfolgreich abgeschlossen haben: - erläutern den Startvorgang eines Rechensystems am Beispiel eines x86/x86_64-PCs. - beschreiben die spezifischen Herausforderungen bei der Softwareentwicklung für "bare metal". - beschreiben den Ablauf einer Unterbrechungsbehandlung von der Hardware bis zur (System-)software. - skizzieren Besonderheiten und Strategien der Unterbrechungsbehandlung in Hardware für Mehrkernsystemen am Beispiel des IA32-APICs. - diskutieren die Aufgabenteilung zwischen Hardware und Systemsoftware bei der Unterbrechungsbearbeitung. - unterscheiden die verschiedenen Typen von Kontrollflüssen in einem Betriebssystem anhand des Ebenenmodells. - unterscheiden harte, mehrstufige, und weiche Verfahren zur Unterbrechungssynchronisation in Betriebssystemen und können diese implementieren. - klassifizieren konkrete Konkurrenzsituationen anhand des Ebenenmodels und leiten daraus geeignete Synchronisationsmaßnahmen ab.

- schildern die IA32-Architektur und gängige PC-Technologie und deren Schnittstellen zur Systemsoftware.

- erläutern grundlegende Bausteine für die Implementierung von Quasi-Parallelität (Fortsetzungen, Koroutinen, Fäden) und grenzen diese gegeneinander ab.

- erläutern die Interaktionen zwischen Hardware, Übersetzer und Systemsoftware, die dabei zu beachten sind.

- entwickeln den Koroutinenwechsel für einen gegebene Architektur.

- erläutern die Implikationen von Quasi-Parallität auf das Ebenenmodell und die daraus abgeleiteten Synchronisationsmaßnahmen.

- beschreiben die Implementierung von (verdrängendem) Scheduling in einem Betriebssystem.

- analysieren das Zusammenspiel von Scheduling und Unterbrechungssynchronisation.

- nennen Kriterien und Dimensionen des Schedulings von Betriebsmitteln, insbesondere der CPU.

- erläutern die konkrete Umsetzung am Beispiel der Scheduler in Linux und Windows.

- unterscheiden grundlegende Möglichkeiten der Koordinierung und Synchronisation von Fäden (aktives/passives Warten, nichtverdrängbare kritische Abschnitte).

- entwickeln Mechanismen für die Synchronisation auf Fadenebene.

- erklären die dabei zu beachtenden Synchronisationsprobleme (lost update, lost wakeup) und geeignete Gegenmaßnahmen.

- interpretieren die Bedeutung von Gerätetreibern in der Betriebssystempraxis.

- erläutern die Anforderungen an ein Treibermodell.

- vergleichen die Umsetzung von Treibermodellen in Windows und Linux.

- vergleichen grundlegende BS-Architekturen (Bibliothek, Monolith, Mikrokern, Exokern, Hypervisor) anhand fundamentaler Charakteristika (Robustheit, Performanz, Portierbarkeit) und Mechanismen.

		- schildern die grundlegenden Paradigmen zur Interprozesskommunikation in Betriebssystemen (speicherbasiert vs. nachrichtenbasiert). - erläutern die grundlegenden Primitiven dieser Verfahren. - skizzieren, wie unter Anwendung dieser Primitiven höhere Synchronisationskonstrukte implementiert werden (Monitore, Leser-/ Schreiber-Sperre). - illustrieren die Dualität der Paradigmen. - erschließen sich typische Probleme (Nebenläufigkeit, Compilerverhalten, Debuggen ohne dedizierte Hilfsmittel) und Fehlerquellen bei der hardwarenahen Softwareentwicklung. - können in Kleingruppen kooperativ arbeiten. - können die ihre Entwurfs- und Implementierungsentscheidungen kompakt präsentieren und argumentativ vertreten. - reflektieren ihre Entscheidungen kritisch und leiten Alternativen ab. - können offen und konstruktiv mit Schwachpunkten und Irrwegen umgehen.
7	**Voraussetzungen für die Teilnahme**	• C/C++ (Übungsaufgaben werden in C++ implementiert) • Assembler (Grundkenntnisse)
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• Avi Silberschatz, Peter B. Galvin, Greg Gagne. Operating System Concepts. John Wiley & Sons. • Andrew S. Tanenbaum. Modern Operating Systems. Pearson. • William Stallings. Operating Systems: Internals and Design Principles. Prentice Hall.

		• Andrew S. Tanenbaum. Structured Computer Organization. Pearson.

1	**Modulbezeichnung** 96312	**Bild-, Video- und mehrdimensionale Signalverarbeitung** (Image, video and multidimensional signal processing)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Bild-, Video- und mehrdimensionale Signalverarbeitung (2 SWS)	5 ECTS
3	Lehrende	Prof. Dr.-Ing. Andre Kaup	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Andre Kaup
5	**Inhalt**	*Punktoperationen* Histogrammausgleich, Gamma-Korrektur *Binäroperationen* Morphologische Filter, Erosion, Dilatation, Opening, Closing *Farbräume* Trichromat, RGB- Farbraum, HSV-Farbraum *Mehrdimensionale Signale und Systeme* Theorie mehrdimensionaler Signale und Systeme, Impulsantwort, lineare Bildfilterung, Leistungsspektrum, Wiener Filter *Interpolation von Bildsignalen* Bilineare Interpolation, Bicubische Interpolation, Spline Interpolation *Merkmalsdetektion in Bildern* Bildmerkmale, Kantendetektion, Hough Transformation, Harris Ecken Detektor, Texturmerkmale, Grauwertematrix *Skalierungsraumdarstellung* LoG, DoG, SIFT, SURF *Bildabgleich* Projektive Abbildungen, Blockabgleich, Optischer Fluss, Merkmalsbasierter Abgleich mittels SIFT und SURF, RANSAC *Bildsegmentierung* Amplituden Schwellenwertermittlung, K-Means Clustering, Bayes Klassifikation, Regionen-basierte Segmentierung, kombinierte Segmentierung und Bewegungsschätzung, zeitliche Segmentierung von Videos

Bildverarbeitung im Transformationsbereich

Unitäre Transformation, Karhunen-Loeve Transformation, separable Transformationen, Haar und Hadamard Transformation, DFT, DCT

Content:

Point operations

Histogram equalization, gamma correction

Binary operations

Morphological filters, erosion, dilation, opening, closing

Color spaces

Trichromacy, red-green-blue color spaces, color representation using hue, saturation and value of intensity

Multidimensional signals and systems

Theory of multidimensional signals and systems, impulse response, linear image filtering, power spectrum, Wiener filtering

Interpolation of image signals

Bi-linear interpolation, bi-cubic interpolation, spline interpolation

Image feature detection

Image features, edge detection, Hough transform, Harris corner detector, texture features, co-occurrence matrix

Scale space representation

Laplacian of Gaussian, difference of Gaussian, scale invariant feature transform, speeded-up robust feature transform

Image matching

Projective transforms, block matching, optical flow, feature-based matching using SIFT and SURF, random sample consensus algorithm

Image segmentation

		Amplitude thresholding, k-means clustering, Bayes classification, region-based segmentation, combined segmentation and motion estimation, temporal segmentation of video *Transform domain image processing* Unitary transform, Karhunen-Loeve transform, separable transform, Haar and Hadamard transform, DFT, DCT
6	**Lernziele und Kompetenzen**	Die Studierenden • verstehen Punktoperationen an Bilddaten und Gamma-Korrektur • testen die Wirkung von Rangordnungs- und Medianfiltern an Bilddaten • unterscheiden und bewerten verschiedene Farbräume für Bilddaten • erklären das Prinzip der zwei-dimensionalen linearen Filterung für Bildsignale • berechnen und bewerten die zweidimensionale diskrete Fourier-Transformierte eines Bildsignales • bestimmen vergrößerte diskrete Bildsignale mit Methoden der bilinearen und Spline-Interpolation • überprüfen Bilddaten auf ausgewählte Textur-, Kanten- und Bewegungsmerkmale • analysieren Bild- und Videodaten auf Merkmale in unterschiedlichen Scale-Spaces • erläutern und beurteilen Methoden für das Matching von Bilddaten • segmentieren Bilddaten durch Programmierung von einfachen Klassifikations- oder Clustering-Verfahren • verstehen das Prinzip von Transformation auf Bilddaten und können diese an Beispielen anwenden. The students • understand point operations for image data and gamma correction • test the effects of rank order and median filters for image data • evaluate and differentiate between different color spaces for image data • explain the principle of two-dimensional linear filtering for image signals • calculate and evaluate the two-dimensional discrete Fourier transform of an image signal • determine enlarged discrete image signals by bi-linear and spline interpolation • verify image data for selected texture, edge and motion features

		• analyze image and video data for features in different scale spaces • explain and evaluate methods for the matching of image data • segment image data by implementing basic classification and clustering methods • understand the principle of transformations on image data and apply them exemplarily
7	**Voraussetzungen für die Teilnahme**	Vorlesung Signale und Systeme I und II
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot der Technischen Fakultät oder der Naturwissenschaftlichen Fakultät Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	J.-R. Ohm: |Multimedia Content Analysis|, Springer, 2016 J. W. Woods: |Multidimensional Signal, Image, and Video Processing and Coding|, Academic Press, 2^nd^ edition, 2012

1	**Modulbezeichnung** 43821	**Computer Graphics** (Computer graphics)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Computer Graphics Basic Tutorials (1 SWS) Vorlesung: Computer Graphics (3 SWS)	1,25 ECTS 3,75 ECTS
3	Lehrende	Jonas Müller Prof. Dr. Marc Stamminger	

4	**Modulverantwortliche/r**	Prof. Dr. Marc Stamminger
5	**Inhalt**	Die Vorlesung gibt eine Einführung in die Computergraphik: • Graphik Pipeline • Clipping • 3D Transformationen • Hierarchische Display Strukturen • Perspektive und Projektionen • Visibilitätsbetrachtungen • Rastergraphik und Scankonvertierung • Farbmodelle • Lokale und globale Beleuchtungsmodelle • Schattierungsverfahren • Ray Tracing und Radiosity • Schatten und Texturen Contents: This lecture covers the following aspects of Computer Graphics: • graphics pipeline • clipping • 3D transformations • hierarchical display structures • perspective transformations and projections • visibility determination • raster graphics and scan conversion • color models • local and global illumination models • shading models • ray tracing and radiosity • shadows and textures
6	**Lernziele und Kompetenzen**	Die Studierenden • geben die unterschiedlichen Schritte der Graphik Pipeline wieder • erklären die Funktionsweise der Clippingalgorithmen für Linien und Polygone • beschreiben, charakterisieren und berechnen affine und perspektivische Transformationen in 3D und veranschaulichen

		die allgemeine Form der Transformationsmatrix in homogener Koordinaten • skizzieren die Verfahren zur Tiefe- und Visibilitätsberechnung • vergleichen die unterschiedlichen Farbmodelle der Computergraphik • illustrieren und untersuchen die Datenstrukturen zur Beschreibung virtueller 3D Modelle und komplexer Szenen • erläutern die Funktionsweise der Rasterisierung und Scankonvertierung in der Graphikpipeline • lösen Aufgaben zu Beleuchtung und Texturierung von 3D virtuellen Modellen • klassifizieren Schattierungsverfahren • bestimmen den Unterschied zwischen lokaler und globaler Beleuchtung und formulieren Algorithmen für Ray Tracing und Radiocity *Educational objectives and skills:* Students should be able to • describe the processing steps in the graphics pipeline • explain clipping algorithms for lines and polygons • explain, characterize and compute affine and perspective transformations in 2D and 3D, and provide an intuitive description of the general form of corresponding transformation matrices in homogeneous coordinates • depict techniques to compute depth, occlusion and visibility • compare the different color models • describe data structures to represent 3D virtual models and complex scenes • explain the algorithms for rasterization and scan conversion • solve problems with shading and texturing of 3D virtual models • classify different shadowing techniques • explain the difference between local and global illumination techniques and formulate algorithms for ray tracing and radiosity
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Übungsleistung Klausur (60 Minuten)
11	**Berechnung der Modulnote**	Übungsleistung (0%) Klausur (100%)
12	**Turnus des Angebots**	nur im Wintersemester

13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• P. Shirley: Fundamentals of Computer Graphics. AK Peters Ltd., 2002 • Hearn, M. P. Baker: Computer Graphics with OpenGLD. Pearson • Foley, van Dam, Feiner, Hughes: Computer Graphics - Principles and Practice • Rauber: Algorithmen der Computergraphik • Bungartz, Griebel, Zenger: Einführung in die Computergraphik • Encarnação, Strasser, Klein: Computer Graphics

1	**Modulbezeichnung** 169383	**CPU Entwurf mit VHDL (Schwerpunkt Prozessorentwurf)** (CPU Design with VHDL (Focus on CPUs))	**5 ECTS**
2	Lehrveranstaltungen	Zu diesem Modul sind keine Lehrveranstaltungen oder Lehrveranstaltungsgruppen hinterlegt!	
3	Lehrende	Zu diesem Modul sind keine Lehrveranstaltungen und somit auch keine Lehrenden hinterlegt!	

4	**Modulverantwortliche/r**	
5	**Inhalt**	keine Inhaltsbeschreibung hinterlegt!
6	**Lernziele und Kompetenzen**	keine Beschreibung der Lernziele und Kompetenzen hinterlegt!
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	keine Angaben zum Turnus des Angebots hinterlegt!
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: ?? h (keine Angaben zum Arbeitsaufwand in Präsenzzeit hinterlegt) Eigenstudium: ?? h (keine Angaben zum Arbeitsaufwand im Eigenstudium hinterlegt)
14	**Dauer des Moduls**	?? Semester (keine Angaben zur Dauer des Moduls hinterlegt)
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 436348	**CPU Entwurf mit VHDL (Schwerpunkt VHDL)** (CPU Design with VHDL (Focus on VHDL))	**5 ECTS**
2	Lehrveranstaltungen	Zu diesem Modul sind keine Lehrveranstaltungen oder Lehrveranstaltungsgruppen hinterlegt!	
3	Lehrende	Zu diesem Modul sind keine Lehrveranstaltungen und somit auch keine Lehrenden hinterlegt!	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Dietmar Fey
5	**Inhalt**	Die Vorlesung wird einen Einblick in die wichtigsten Strukturen eines Rechners und insbesondere die der CPU geben. Wie sind die Register, die ALU, die Caches usw. aufgebaut, wie sind sie miteinander verschaltet? Wie werden Pipelines gebildet? Um die Vorlesung verstehen und die Übungsaufgaben lösen zu können, werden VHDL-Kenntnisse benötigt. Diese werden in der Vorlesung vermittelt. Im ersten Teil der Vorlesung geht es darum, aus einfachen Und-, Oder- und Inverter-Gattern größere Strukturen aufzubauen. So werden bespielsweise die Register und die Komponenten der ALU (Addierer, Multiplizierer, Shifter usw.) nach und nach aufgebaut. Aus den Einzelteilen lassen sich dann wiederum die Registerbänke, die ALU, die Speichereinheit usw. zusammensetzen. Mit guten Rechenwerken allein ist eine CPU noch nicht wirklich schnell. Es müssen schnell genug Instruktionen gelesen und dekodiert sowie die Operanden geholt werden können. Nach dem schnellen Verarbeiten in den Rechenwerken müssen die Ergebnisse dann wiederum schnell in die Zielregister bzw. Ziel-Speicherzellen kopiert werden. Das wichtigste Verfahren, um hierbei Performance zu gewinnen, ist das Pipelining. Leider gibt es eine Reihe von Bedingungen, die erfüllt sein müssen, dass Pipelining wirklich gut funktioniert. So dürfen keine Daten-Abhängigkeiten im Code vorhanden sein, müssen Sprungziele rechtzeitig bekannt sein und dürfen bestimmte Ressourcen nicht mehrfach gleichzeitig genutzt werden. Mittels Forwarding, Sprungvorhersage und Caches können die Probleme gemildert werden. Mit den so gewonnenen Fähigkeiten sind dann auch Multi-Core- und Multi-Threading-Architekturen verständlich.
6	**Lernziele und Kompetenzen**	Die Studierenden verstehen den inneren Aufbau moderner CPUs (Pipeline, Registerbänke, ALU, Caches, Memory-Management-Unit, Segmentierungseinheit, ...) und können selbst performante CPUs aus einfachen Basis-Schaltungen aufbauen.
7	**Voraussetzungen für die Teilnahme**	keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1

9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 451696	**Cyber-Physical Systems** (Cyber-physical systems)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übung zu Cyber-Physical Systems (2 SWS) Vorlesung: Cyber-Physical Systems (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Dr.-Ing. Torsten Klie	

4	**Modulverantwortliche/r**	Joachim Falk Dr.-Ing. Torsten Klie
5	**Inhalt**	Klassische Computersysteme zeichnen sich durch eine strikte Trennung von realer und virtueller Welt aus. Moderne Steuerungssysteme, die z.B. in modernen Fahrzeugen verbaut sind und die aus einer Vielzahl von Sensoren und Aktoren bestehen, entsprechen diesem Bild nur sehr eingeschränkt. Diese Systeme, oft "Cyber-Physical Systems (CPS)" genannt, erkennen ihre physische Umgebung, verarbeiten diese Informationen und können die physische Umwelt auch koordiniert beeinflussen. Hierzu ist eine starke Kopplung von physischem Anwendungsmodell und dem Computer-Steuerungsmodell nötig. Im Unterschied zu Eingebetteten Systemen bestehen CPS meist aus vielen vernetzten Komponenten, die sich selbständig untereinander koordinieren. Diese Vorlesung spannt den Bogen von kontrolltheoretischen Grundlagen über Selbstorganisiationsprinzipien bis hin zu visionären Anwendungen aus den Bereichen Verkehr und Medizintechnik. Ferner werden Entwurfsmethoden für Cyber-Physical Systems vorgestellt.
6	**Lernziele und Kompetenzen**	Die Studierenden erläutern, was Cyber-Physical Systems sind und auf welchen technologischen Grundlagen sie aufbauen, insbesondere in den Bereichen Regelungstechnik, Ablaufplanung, Kommunikation und Selbstorganisation bewerten CPS in verschiedenen Anwendungsgebieten stellen den Entwurfsprozess von CPS dar, insbesondere die Modellierung und die grundlegende Programmierung entdecken

		wesentliche Herausforderungen beim Entwurf, Ausbringung und Einsatz von CPS.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 0
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Empfohlene Bücher zur Begleitung und Vertiefung: • Andrea Bondavalli, Sara Bouchenak und Hermann Kopetz (Hrsg.) Cyber-Physical Systems of Systems: Foundations – A Conceptual Model and Some Derivations: The AMADEOS Legacy. Springer 2016. • Otto Föllinger Regelungstechnik. Hüthig 1992. • Hilmar Jaschek und Holger Voos Grundkurs der Regelungstechnik. Oldenbourg 2010. • Jörg Kahlert Crash-Kurs Regelungstechnik. VDE Verlag 2010. • Peter Marwedel Embedded Systems Design – Embedded Systems Foundations of Cyber-Physical Systems, and the Internet of Things, 4. Auflage. Springer 2021 • André Platzner Logic Foundations of Cyber-physical Systems. Springer 2018. • Wolfgang Schneider Praktische Regelungstechnik. Vieweg +Teubner 2008. • Walid M. Taha, Abd-Ehamid M. Taha und Johan Thunberg Cyber-physical Systems – A Model-based Approach. Springer 2021. Weitere Informationen: https://www.cs12.tf.fau.de/lehre/lehrveranstaltungen/vorlesungen/cyber-physical-systems/

1	**Modulbezeichnung** 901895	**Deep Learning** (Deep learning)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Deep Learning (2 SWS) Übung: Deep Learning Exercises (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr.-Ing. Andreas Maier Leonhard Rist Zijin Yang	

4	**Modulverantwortliche/r**	Felix Denzinger Prof. Dr.-Ing. Andreas Maier Fabian Wagner
5	**Inhalt**	Deep Learning (DL) has attracted much interest in a wide range of applications such as image recognition, speech recognition and artificial intelligence, both from academia and industry. This lecture introduces the core elements of neural networks and deep learning, it comprises: • (multilayer) perceptron, backpropagation, fully connected neural networks • loss functions and optimization strategies • convolutional neural networks (CNNs) • activation functions • regularization strategies • common practices for training and evaluating neural networks • visualization of networks and results • common architectures, such as LeNet, Alexnet, VGG, GoogleNet • recurrent neural networks (RNN, TBPTT, LSTM, GRU) • deep reinforcement learning • unsupervised learning (autoencoder, RBM, DBM, VAE) • generative adversarial networks (GANs) • weakly supervised learning • applications of deep learning (segmentation, object detection, speech recognition, ...) The accompanying exercises will provide a deeper understanding of the workings and architecture of neural networks.
6	**Lernziele und Kompetenzen**	The students • explain the different neural network components, • compare and analyze methods for optimization and regularization of neural networks, • compare and analyze different CNN architectures, • explain deep learning techniques for unsupervised / semi-supervised and weakly supervised learning, • explain deep reinforcement learning, • explain different deep learning applications, • implement the presented methods in Python,

		• autonomously design deep learning techniques and prototypically implement them, • effectively investigate raw data, intermediate results and results of Deep Learning techniques on a computer, • autonomously supplement the mathematical foundations of the presented methods by self-guided study of the literature, • discuss the social impact of applications of deep learning applications.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20191 Wahlpflichtmodul aus INF im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	• Ian Goodfellow, Yoshua Bengio, Aaron Courville: Deep Learning. MIT Press, 2016. • Christopher Bishop: Pattern Recognition and Machine Learning, Springer Verlag, Heidelberg, 2006 • Yann LeCun, Yoshua Bengio, Geoffrey Hinton: Deep learning. Nature 521, 436444 (28 May 2015)

1	**Modulbezeichnung** 44150	**Diagnostic Medical Image Processing** (Diagnostic medical image processing)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Medical Image Processing for Diagnostic Applications (VHB-Kurs) (4 SWS)	5 ECTS
3	Lehrende	Manuela Meier Arpitha Ravi Celia Martín Vicario Luis Rivera Monroy	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Andreas Maier
5	**Inhalt**	English version: The contents of the module comprise basics about medical imaging modalities and acquisition hardware. Furthermore, details on acquisition-dependent preprocessing are covered for image intensifiers, flat-panel detectors, and MR. The fundamentals of 3D reconstruction from parallel-beam to cone-beam reconstruction are also covered. In the last chapter, rigid registration for image fusion is explained. Deutsche Version: Die Inhalte des Moduls umfassen Grundlagen der medizinischen Bildverarbeitung und Aufnahmeprinzipien. Darüber hinaus werden Details der Vorverarbeitung für Bildverstärker, Flachpaneldetektoren und MR erklärt. Die Grundlagen der Rekonstruktion von Parallelstrahl bis hin zur Kegelstrahl-Tomographie werden ebenfalls behandelt. Im letzten Kapitel wird starre Registrierung für Bildfusion erläutert.
6	**Lernziele und Kompetenzen**	English Version: The participants • understand the challenges in interdisciplinary work between engineers and medical practitioners. • develop understanding of algorithms and math for diagnostic medical image processing. • learn that creative adaptation of known algorithms to new problems is key for their future career. • develop the ability to adapt algorithms to different problems. • are able to explain algorithms and concepts of the module to other engineers. Deutsche Version: Die Teilnehmenden • verstehen die Herausforderungen in der interdisziplinären Arbeit zwischen Ingenieuren und Ärzten. • entwickeln Verständnis für Algorithmen und Mathematik der diagnostischen medizinischen Bildverarbeitung.

		• erfahren, dass kreative Adaption von bekannten Algorithmen auf neue Probleme der Schlüssel für ihre berufliche Zukunft ist. • entwickeln die Fähigkeit Algorithmen auf verschiedene Probleme anzupassen. • sind in der Lage, Algorithmen und Konzepte des Moduls anderen Studierenden der Technischen Fakultät zu erklären.
7	**Voraussetzungen für die Teilnahme**	Ingenieurmathematik Engineering Mathematics
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich/mündlich
11	**Berechnung der Modulnote**	schriftlich/mündlich (100%)
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 0 h Eigenstudium: 150 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 472330	**Dienstgüte von Kommunikationssystemen** (Quality of service in communication systems)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Quality of Service in Communications (2 SWS) Übung: Quality of Service in Communications (Ex-QoSiC) (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr. Reinhard German Lisa Maile	

4	**Modulverantwortliche/r**	Prof. Dr. Reinhard German
5	**Inhalt**	Zunächst wird der Begriff der Dienstgüte (Quality-of-Service, QoS) eingegrenzt und es werden die wichtigsten Ansätze zur Erzielung von Dienstgüte besprochen und in ausgewählten Netztechnologien untersucht. Dann werden unterschiedliche Methoden vorgestellt, mit denen Systeme bezüglich ihrer Dienstgüte bewertet und ausgelegt werden können: • Netzplanung und optimierung, • stochastische Analyse (Markow-Ketten, Warteschlangen), • Netzwerksimulation, • deterministische Analyse mit Network Calculus zur Ermittlung von Dienstgütegarantien • Messung (HW-, SW-, Hybrid-Monitoring, Benchmarks). Alle Methoden werden an Beispielen demonstriert. *Contents:* We introduce the term quality-of-service (QoS), discuss important approaches to achieve certain degrees of QoS, and show how the implementation in computer networks. Then a number of methodologies to assess and design systems with respect to their QoS: • network planning and optimization, • network simulation, • stochastic analysis (Markov chains, non-Markovian models, queuing systems), • deterministic analysis with network calculus to determine QoS guarantees • measurements (hardware, software, and hybrid monitoring, benchmarks). All methods are illustrated by examples.
6	**Lernziele und Kompetenzen**	Die Studierenden erwerben • Kenntnisse in Methoden zur Modellierung und Bewertung von quantitativen, nicht-funktionalen Eigenschaften von vernetzten Systemen • Kenntnisse in Mechanismen von vernetzten Systemen zur Erzielung von Dienstgüte *Learning targets and competences:*

		The students get • experience in methods to model and evaluate quantitative, non-functional properties of computer networks and related systems • knowledge of mechanisms of computer networks to achieve quality-of-service
7	**Voraussetzungen für die Teilnahme**	Rechnerkommunikation, Kommunikationssysteme, grundlegende Programmierkenntnisse (optimal in R und C++)
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	• Kurose, Ross. Computer Networking: A Top-Down Approach Featuring the Internet. 6th Ed., Addison Wesley, 2013 • W. Stallings. Data and Computer Communications, 10th ed., Pearson Education, 2014 • W. Stallings. Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud, Pearson Education, 2016

1	**Modulbezeichnung** 47800	**Digital Communications** (Digital communications)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Digital Communications (3 SWS) Übung: Tutorial for Digital Communications (1 SWS)	5 ECTS -
3	Lehrende	Prof. Dr. Laura Cottatellucci Brikena Kaziu	

4	**Modulverantwortliche/r**	Prof. Dr. Laura Cottatellucci
5	**Inhalt**	Alle modernen Kommunikationssysteme basieren auf digitalen Übertragungsverfahren. Diese Vorlesung befasst sich mit den Grundlagen der Analyse und des Entwurfs digitaler Sender und Empfänger. Dabei wird zunächst von einem einfachen Kanalmodell bei dem das Empfangssignal nur durch additives weißes Gaußsches Rauschen gestört wird ausgegangen. Im Verlauf der Vorlesung werden aber auch Kanäle mit unbekannter Phase sowie verzerrende Kanäle betrachtet. Behandelt werden unter anderem digitale Modulationsverfahren (z.B. Pulsamplitudenmodulation (PAM), digitale Frequenzmodulation (FSK), und Kontinuierliche-Phasenmodulation (CPM)), Orthogonalkonstellationen, das Nyquistkriterium in Zeit- und Frequenzbereich, optimale kohärente und inkohärente Detektions- und Decodierungsverfahren, die Signalraumdarstellung digital modulierter Signale, verschiedene Entzerrungsverfahren, und Mehrträger-Übertragungsverfahren. --- Modern communication systems are based on digital transmission methods. This course covers basics of analysis and design of digital transmitters and receivers. Initially, we consider a simple channel model whose received signal is impaired only by additive white Gaussian noise. Then, we extend fundamental concepts to channels with unknown phases and distortion. Additionally, we treat digital modulation techniques, e.g., pulse amplitude modulation (PAM), digital frequency modulation (FSK) and continuous-phase modulation (CPM), and orthogonal constellations. The Nyquist criterion in time and frequency domain, optimal coherent and incoherent detection and decoding methods, signal space representations of digitally modulated signals, various equalization methods, and multicarrier transmission methods are also discussed.
6	**Lernziele und Kompetenzen**	Die Studierenden • analysieren und klassifizieren digitale Modulationsverfahren hinsichtlich ihrer Leistungs- und Bandbreiteneffizienz sowie ihres Spitzenwertfaktors, • ermitteln notwendige Kriterien für impulsinterferenzfreie Übertragung,

		• charakterisieren digitale Modulationsverfahren im Signalraum, • ermitteln informationsverlustfreie Demodulationsverfahren, • entwerfen optimale kohärente und inkohärente Detektions- und Decodierungsverfahren, • vergleichen verschiedene Entzerrungsverfahren hinsichtlich deren Leistungsfähigkeit und Komplexität, • entwerfen einfache digitale Übertragungssysteme mit vorgeschriebenen Leistungs- und Bandbreiteneffizienzen sowie Spitzenwertfaktoren. • -- The students • analyze and classify digital modulation techniques in terms of performance and bandwidth efficiency as well as crest factor, • determine necessary criteria to design impulses for interference-free transmission, • characterize digital modulation methods in signal space, • determine information loss-free demodulation methods, • design optimal coherent and incoherent detection and decoding methods, • compare different equalization methods in terms of performance and complexity, • design simple digital transmission systems with prescribed power and bandwidth efficiency and crest factor.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 96090	**Digitale elektronische Systeme** (Digitale elektronische Systeme)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Digitale elektronische Systeme (3 SWS) Übung: Übungen zu Digitale elektronische Systeme (1 SWS)	5 ECTS -
3	Lehrende	Prof. Dr.-Ing. Robert Weigel Albert-Marcel Schrotz	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Robert Weigel
5	**Inhalt**	• Analog-Digital-Umsetzer: Qualitätsmerkmale, Messtechnik, Hardwarearchitekturen • Digital-Analog-Umsetzer: Qualitätsmerkmale, Messtechnik, Hardwarearchitekturen • Programmierbare Logikschaltungen (PLD, FPGA): Grundlegende Konzepte, Kategorien, Hardwarearchitekturen • Digitale-Filter: Theorie, Eigenschaften, Entwicklung und Implementierung und IIR und FIR Filtern
6	**Lernziele und Kompetenzen**	• Die Studierenden verstehen die Hardwarearchitekturen und Funktionsweisen von Komponenten digitaler Elektronischer Systeme wie Digital-Analog-Umsetzer, Analog-Digital Umsetzer, PLDs und FPGAs und können diese erläutern • Die Studierenden Verstehen die Qualitätsmerkmale von Digitalen Elektronischen Komponenten, können diese auf konkrete Komponenten anwenden und somit die Qualität von digitalen Elektronischen Komponenten anhand der in Datenblättern typischer weise gegebenen Qualitätsmerkmale evaluieren • Die Studierenden können die Einflüsse von nichtidealen Bauelementen auf digitale elektronische Systeme analysieren • Die Studierenden verstehen die Funktion, die Eigenschaften, die Entwicklungsmethodik sowie die Implementierung von digitalen Filtern und könne diese erläutern
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	Unregelmäßig
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester

15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 710850	**eBusiness Technologies und Evolutionäre Informationssysteme** (eBusiness technologies and evolutionary information systems)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Evolutionäre Informationssysteme (2 SWS) Vorlesung: eBusiness Technologies (2 SWS)	- -
3	Lehrende	Prof. Dr.-Ing. Richard Lenz	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Richard Lenz
5	**Inhalt**	**EBT:** • Überblick und Einblick in die wichtigsten Themen des Bereichs Business • User Interface, Business Logic und Database Layer • Agile Softwareentwicklung • Integration von Enterprise-Applikationen • Cloud & Container • DevOps **EIS:** • Grundlagen rechnergestützter Informationssysteme und organisatorisches Lernen • Erfolgsfaktoren für Projekte • Software Wartung vs. Software Evolution • Architekturmodelle • Grundprinzipien evolutionärer Systeme • Datenqualität in Informationssystemen **Contents:** **EBT:** • Modern technologies to implement Web-Applications for eBusiness • User Interface, Business Logic and Database Layer • Agile Software Development • Integration of Enterprise-Applications • Cloud & Container • DevOps **EIS:** • IT-Support for Organizational Learning • Success- and Failure Factors for large scale IT-Projects • Software Maintenance vs. Software Evolution • Architectural Styles and their Impact on Evolvability • Principles for Evolvable Systems • Data Quality in Information Systems

6	**Lernziele und Kompetenzen**	**EBT:** Die Studierenden • identifizieren die wichtigsten Themen des Bereichs eBusiness, von den Anwendungen bis zu den Implementierungen • verstehen Zusammenhänge der B2B-Integration und der Realisierung von eBusiness-Anwendungen • wiederholen Grundlagen des Webs • vergleichen technische Eigenschaften von HTTP-, Web- und Application Servern • vergleichen Markup Languages (HTML, XML) • unterscheiden Ansätze zur Schema-Modellierung wie DTD und XML Schema und erkennen die unterschiedliche Leistungsfähigkeit • verstehen Methoden zur evolutionsfähigen Gestaltung von Datenstrukturen in XML • unterscheiden Vorgehen bei der Datenhaltung und verschiedene Ansätze für den Datenbankzugriff • verstehen Objekt-relationale Mapping Frameworks am Beispiel von Hibernate und JPA • verstehen Komponentenmodelle wie Enterprise JavaBeans (EJB) aus dem JEE Framework • unterscheiden das EJB Komponentenmodell von den OSGi Bundles und den Spring Beans • verstehen und unterscheiden grundlegende Web Service Techniken wie SOAP und WSDL • unterscheiden Herangehensweisen zur dynamischen Generierung von Webseiten • verstehen grundlegende Eigenschaften eines Java-basierten Front-End-Frameworks am Beispiel von JSF • verstehen grundlegende Eigenschaften von Service-orientierten Architekturen (SOA) • verstehen agile Vorgehensmodelle zur Software-Entwicklung am Beispiel von Scrum • unterscheiden agile Verfahren wie Scrum von iterativ-inkrementellen Verfahren wie RUP • verstehen die Wichtigkeit von Code-Beispielen um die praktische Anwendbarkeit des theoretischen Wissens zu veranschaulichen. • können die Code-Beispiele eigenständig zur Ausführung bringen und die praktischen Erfahrungen interpretieren und bewerten • gestalten eigene Lernprozesse selbständig. • schätzen ihre eigenen Stärken und Schwächen im Hinblick auf die unterschiedlichen Architektur-Schichten ein(Benutzerinteraktion, Applikationslogik, Schnittstellenintegration, Datenbanksysteme) • identifizieren eine eigene Vorstellung als zukünftige Software-Architekten und können die eigene Entwicklung planen

- reflektieren durch regelmäßige fachbezogene Fragen des Dozenten Ihre eigene Lösungskompetenz.

EIS:

Die Studierenden:

- definieren die Begriffe "Informationssysteme", "evolutionäre Informationssyste" und "organisatorisches Lernen"
- grenzen die Begriffe "Wissen" und "Information" gegeneinander ab
- charakterisieren die in der Vorlesung erläuterten Formen der organisatorischen Veränderung
- erklären das SEKI Modell nach Nonaka und Takeuchi
- nennen Beispiele für die in der Vorlesung behandelten Formen der Wissensrepräsentation in IT-Systemen
- nennen typische Erfolgs- und Risikofaktoren für große IT-Projekte
- erklären die Kraftfeldtheorie nach Kurt Lewin
- unterscheiden Typen von Software gemäß der Klassifikation nach Lehman und Belady
- unterscheiden die in der Vorlesung vorgestellten Arten der Software Wartung
- benennen die Gesetzmäßigkeiten der Software-Evolution nach Lehman und Belady
- bewerten die in der Vorlesung vorgestellten Vorgehensmodelle zur Softwareerstellung im Kontext der E-Typ-Software
- nennen die in der Vorlesung vorgestellten Aspekte der Evolutionsfähigkeit von Software
- erklären, wie die in der Vorlesung vorgestellten Methoden zur Trennung von Belangen beitragen
- erklären das Konzept des "Verzögerten Entwurfs"
- erklären die Vor- und Nachteile generischer Datenbankschemata am Beispiel von EAV und EAV/CR
- charakterisieren die in der Vorlesung vorgestelten Architekurkonzepte
- grenzen die in der Vorlesung vorgestellten Integrationsanforderungen gegeneinander ab
- erklären wie Standards zur Systemintegration beitragen und wo die Grenzen der Standardisierung liegen
- erklären das Prinzip eines Kommunikationsservers und der nachrichtenbasierten Integration
- erklären den Begriff "Prozessintegration"
- definieren den Begriff "Enterprise Application Integration" (EAI)
- unterscheiden die in der Vorlesung vorgestellten Integrationsansätze
- erklären die in der Vorlesung vorgestellten Dimensionen der Datenqualität
- unterscheiden die grundlegenden Messmethoden für Datenqualität

		• erklären das Maßnahmenportfolio zur Verbesserung der Datenqualität nach Redman • benennen die in der Vorlesung vorgestellten Methoden zur Verbesserung der Datenqualität
7	**Voraussetzungen für die Teilnahme**	Programmieren in Java, Datenbanken (SQL)
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	siehe Lehrveranstaltungsbeschreibungen

1	**Modulbezeichnung** 939179	**Echtzeitsysteme 2 - Verlässliche Echtzeitsysteme** (Real-time systems 2 - dependable real-time systems)	**7,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Echtzeitsysteme 2 - Verlässliche Echtzeitsysteme (2 SWS) Übung: Echtzeitsysteme 2 - Verlässliche Echtzeitsysteme - Erweiterte Übungen (2 SWS) Übung: Echtzeitsysteme 2 - Verlässliche Echtzeitsysteme - Rechnerübungen (2 SWS)	- - -
3	Lehrende	Peter Wägemann Prof. Dr.-Ing. Wolfgang Schröder-Preikschat Tim Rheinfels Simon Schuster	

4	**Modulverantwortliche/r**	Peter Wägemann
5	**Inhalt**	Viele Echtzeitsysteme sind in Bereiche des täglichen Lebens eingebettete, die hohe Anforderungen an die funktionale Sicherheit dieser Systeme stellen. Beispiele hierfür sind Fahrerassistenzsysteme in modernen Automobilen, medizinische Geräte, Prozessanlagen in Kernkraftwerken oder Chemiefabriken oder Flugzeuge. Fehlfunktionen in diesen Anwendungen ziehen mitunter katastrophale Konsequenzen nach sich - Menschen können ernsthaft verletzt oder sogar getötet werden, Landstriche können unbewohnbar gemacht oder zumindest großer finanzieller Schaden verursacht werden. Dieses Modul betrachtet Methoden und Werkzeuge, die uns helfen können, einerseits *zuverlässig Software zu entwickeln* (also Fehler im Programm zu entdecken und zu vermeiden), und andererseits *zuverlässige Software zu entwickeln* (also Abstraktionen, die auch im Fehlerfall ihre Gültigkeit behalten). Hierbei steht weniger die Vermittlung theoretischer Grundkenntnisse auf diesen Gebieten im Vordergrund, also vielmehr • die praktische Anwendung existierende Werkzeuge und Methoden • sowie die Erfahrung und das Verständnis ihrer Grenzen. Auf diese Weise soll ein Fundament für die konstruktive Umsetzung verlässlicher Echtzeitsysteme gelegt werden. Dieses Modul soll daher fundierte Anknüpfungspunkte für die Entwicklung verlässlicher Echtzeitsysteme vermitteln, die Ad-hoc-Techniken möglichst ersetzen sollen.
6	**Lernziele und Kompetenzen**	Studierende, die das Modul erfolgreich abgeschlossen haben: • nennen die Konzepte und die Taxonomie verlässlicher Systeme, unterscheiden Software- und Hardwarefehler und klassifizieren Fehler (Defekt, Fehler, Fehlverhalten). • stellen Fehlerbäume auf. • organisieren Softwareentwicklungsprojekte mittels der Versionsverwaltung git.

		• vergleichen die verschiedenen Arten der Redundanz als Grundvorraussetzung für Fehlererkennung und -toleranz. • entwickeln fehlertolerante Systeme mittels Replikation. • diskutieren die Fehlerhypothese und die Sicherstellung von Replikdeterminismus. • erläutern die Vor- und Nachteile softwarebasierter Replikation und den Einsatz von Diversität. • wenden Informationsredundanz zur Härtung von Daten- und Kontrollflüssen an. • bewerten die Effektivität der arithmetischer Codierung von Programmen und verallgemeinern diesen Ansatz auf die verschiedenen Implementeirungsebenen (Maschinenprogramm zu Prozessinkarnation). • interpretieren den Einfluss der Ausführungsplattform (Hardware, Betriebssystem) auf die Leistungsfähigkeit der Fehlererkennung. • konzipieren eine fehlertolerante Ausführungsumgebung für ein softwarebasiertes TMR-System basierend auf ANBD-Codierung. • nennen die Grundlagen der systematischen Fehlerinjektion. • überprüfen die Wirksamkeit von Fehlertoleranzmechanismen mittels Fehlerinjektion auf der Befehlssatzebene. • entwickeln Testfälle für die Fehlerinjektion mittels des fail** Werkzeugs. • setzten Messergebnisse in Relation zu dem tatsächlichen Fehlerraum. • beschreiben die Grundlagen der Fehlererholung (Vorwärts- bzw. Rückwärtskorrektur) und Reintegration fehlgeschlagener Knoten. • vergleichen den Zustandstransfer am Beispiel der Running bzw. Recursive State Restoration. • benennen Konzepte der Rückwärtskorrektur durch Entwurfsalternativen (Recovery Blocks). • fassen die Grundlagen des dynamischen Testens zusammen. • unterscheiden Black-Box und White-Box Testverfahren. • konzipieren und implementieren Testfälle. • überprüfen die Testüberdeckung anhand grundlegender Überdeckungskriterien (Anweisungs- bis Bedingungsüberdeckung). • geben die Grundlagen der statischen Programmanalyse wieder. • nennen die Funktionsweise von Hoare- WP-Kalkül. • verifizieren eine Ampelsteuerung mittels des FramaC Werkzeugs zur statischen Analyse von C Programmen. • beschreiben den Korrektheitsnachweis mittels abstrakter Interpretation und unterscheiden die konkrete von der abstrakten Programmsemnatik. • erläutern die Funktionsweise von Sammel- und Präfixsemantiken.

		• erstellen einen Korrektheitsbeweis für einen a-b-Filter mittels des Astrée Werkzeugs zur abstrakten Interpretation von C Programmen. • bewerten die Verlässlichkeit kommerzieller, sicherheitskritischer Systeme anhand von Fallstudien (Sizewell B, Airbus A320). • erschließen sich typische Probleme und Fehlerquellen bei der Programmierung von eingebetteten Systemen im Allgemeinen. • klassifizieren Fallstricke und Mehrdeutigkeiten in der Programmiersprache C99 im Besonderen. • können in Gruppen kooperativ und effektiv arbeiten. • können ihre Entwurfs- und Implementierungsentscheidungen kompakt präsentieren und argumentativ vertreten. • reflektieren ihre Entscheidungen kritisch und leiten Alternativen ab. • können offen und konstruktiv mit Schwachpunkten in der Konzeption wie Umsetzung umgehen.
7	**Voraussetzungen für die Teilnahme**	Für eine erfolgreiche Teilnahme an der Veranstaltung sind grundlegende Programmierkenntnisse in C/C++ erforderlich. Diese können durch den (empfohlenen) Besuch entsprechender Grundlagenveranstaltungen oder im Eigenstudium erworben sein. Weiterhin sind grundlegende Kenntnisse über Echtzeitsystemeeine, zum Beispiel durch den Besuch der Veranstaltung "Echtzeitsysteme", empfohlen. Eine formale Voraussetzung besteht in diesem Zusammenhang jedoch nicht.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	Unregelmäßig
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 90 h Eigenstudium: 135 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 179490	**Echtzeitsysteme mit erweiterten Übungen** (Real-time systems with extended exercises)	**7,5 ECTS**
2	Lehrveranstaltungen	Übung: Rechnerübungen zu Echtzeitsysteme (2 SWS) Vorlesung: Echtzeitsysteme (2 SWS) Übung: Erweiterte Übungen zu Echtzeitsysteme (2 SWS)	- - 5 ECTS
3	Lehrende	Peter Wägemann Simon Schuster	

4	**Modulverantwortliche/r**	Peter Wägemann
5	**Inhalt**	Videobearbeitung in Echtzeit, Echtzeitstrategiespiel, echtzeitfähig - der Begriff Echtzeit ist wohl einer der am meisten strapazierten Begriffe der Informatik und wird in den verschiedensten Zusammenhängen benutzt. Diese Vorlesung beschäftigt sich mit dem Begriff Echtzeit aus der Sicht von Betriebssystemen - was versteht man eigentlich unter dem Begriff Echtzeit im Betriebssystemumfeld, wo und warum setzt man sog. Echtzeitbetriebssysteme ein und was zeichnet solche Echtzeitbetriebssysteme aus? In dieser Vorlesung geht es darum, die oben genannten Fragen zu beantworten, indem die grundlegenden Techniken und Mechanismen vermittelt werden, die man im Betriebssystemumfeld verwendet, um Echtzeitsysteme und Echtzeitbetriebssysteme zu realisieren. Im Rahmen dieser Vorlesung werden unter anderem folgende Themen behandelt: • zeitgesteuerte und ereignisgesteuerte Systeme • statische und dynamische Ablaufplanungsverfahren • Fadensynchronisation in Echtzeitbetriebssystemen • Behandlung von periodischen und nicht-periodischen Ereignissen In den begleitenden Übungen werden die in der Vorlesung vorgestellten Techniken bei der Entwicklung eines kleinen Echtzeitsystems praktisch umgesetzt.
6	**Lernziele und Kompetenzen**	Studierende, die das Modul erfolgreich abgeschlossen haben: • unterscheiden die verschiedenen Komponenten eines Echtzeitsystems. • bewerten die Verbindlichkeiten von Terminvorgaben (weich, fest, hart). • erläutern die Zusammensetzung des Laufzeitverhaltes einer Echtzeitanwendung. • klassifizieren die Berührungspunkte zwischen physikalischem Objekt und kontrollierendem Echtzeitsystem. • interpretieren die Zeitparameter des durch das Echtzeitrechensystem zu kontrollierenden Objekts.

		• nennen die Zeitparameter des zugrundeliegenden Rechensystems (Unterbrechungslatenz, Ausführungszeit, ...). • unterscheiden synchrone und asynchrone Programmunterbrechung (insbesondere Trap/Interrupt, Ausnahmebehandlung und Zustandssicherung). • skizzieren die Verwaltungsgemeinkosten des schlimmsten Falls. • entwickeln in der Programmiersprache C und wenden die GNU Werkzeugkette für den ARM Cortex M4 an. • erstellen Echtzeitanwendungen auf Basis der eCos OS-Schnittstelle • ordnen die Strukturelemente von Echtzeitanwendungen zu: Aufgabe, Arbeitsauftrag und Faden. • erläutern die Implikationen von zeitlichem Mehrfachbetrieb auf die Verwaltungsgemeinkosten. • unterscheiden die Umsetzungsalternativen zur Ablaufsteuerung und die Trennung der Belange in Einplanung (Strategie) und Einlastung (Mechanismus). • benennen die grundsätzliche Verfahren der Ablaufsteuerung (taktgesteuert, reihum, vorranggesteuert). • erklären die grundlegenden Zeitparameter einer Aufgabe (Auslösezeitpunkt, Termin, Antwortzeit, Latenz, Ausführungszeit, Schlupfzeit). • unterscheiden die Grundlagen der Planbarkeit (gültig vs. zulässig, Optimalität von Einplanungsalgorithmen). • beschreiben den Unterschied zwischen konstruktiver und analytischer Einhaltung von Terminen-. • vergleich die Möglichkeiten (statisch, dynamisch) der zeitliche Analyse von Echtzeitanwendungen. • erklären die Grundlagen und Beschränkungen von dynamischer (worst-case?) und statischer WCET-Analyse (makroskopisch und mikroskopisch). • illustrieren Lösungsverfahren zur Bestimmung des längsten Ausführungspfads (Timing Schema, IPET). • erstellen Zeitmessung mittels Zeitgeber / Oszilloskop und bestimmen den längsten Pfad durch Code-Review. • erproben und konzipieren werkzeuggestützte WCET-Analyse mittels des absint aiT Analysewerkzeugs. • Entwickeln und annotieren Flußrestrikionen für die statische WCET-Analyse. • beschreiben die Grundlagen der Abfertigung periodischer Echtzeitsysteme (Periode, Phase, Hyperperiode). • skizzieren das periodische Modell und dessen Folgen (Entwicklungskomfort vs. Analysierbarkeit). • erklären die ereignisgesteuerte Ausführung (feste und dynamische Priorität, Verdrängbarkeit) mittels ereignisorienterter Planer (Berechnungskomplexität, MLQ-Scheduler, O(1)-Scheduler).

		• unterscheiden die zeitgesteuerte Ausführung (Busy Loop, Ablaufplan)und die Abfertigung von Arbeitsaufträgen im Abfrage- bzw. Unterbrecherbetrieb. • wenden die Grundlagen der ereignisgesteuerten Ablaufplanung periodischer Echtzeitsysteme an. • unterscheiden Verfahren zur statischen (RM, DM) und dynamischen Prioritätsvergabe (EDF, LRT, LST). • nennen den Unterschied zwischen Anwendungs- und Systemebene (Mehrdeutigkeit von Prioritäten). • erläutern den Optimalitätsnachweis des RM-, DM- und EDF-Algorithmus und dessen Ausnahmen. • beschreiben grundlegende Verfahren zur Planbarkeitsanalyse (CPU-Auslastung, Antwortzeitanalyse). • implementieren komplexe Aufgabensysteme in eCos. • unterscheiden die Grundlagen der zeitgesteuerten Ablaufplanung periodischer Echtzeitsysteme. • erstellen regelmäßige, zyklische Ablaufpläne (cyclic executive model, Rahmen). • vergleich Methoden der manuellen und algorithmischen Ablaufplanung. • unterscheiden optimale von heuristischen Verfahren (List Scheduling, Branch & Bound). • diskutieren die Konsequenzen eines Betriebswechsels in Echtzeitsystemen. • erstellen takt- beziehungsweise ereignisgesteuerte Abläufe in eCos beziehungsweise tt-eCos. • entwickeln ein softwarebasiertes Oszilloskop und erstellen dessen zeitliche Analyse und Ablaufplanung. • klassifizieren die Grundlagen der Abfertigung nicht-periodischer Echtzeitsysteme (minimale Zwischenankunftszeit). • definieren die Verbindlichkeiten von nicht-periodischen Aufgaben (aperiodisch, sporadisch) • zeigen die sich ergebenden Restriktionen des periodischen Modells (Mischbetrieb, Prioritätswarteschlangen, Übernahmeprüfung) auf. • beschreiben die Basistechniken des Laufzeitsystems (Zusteller, Unterbrecherbetrieb, Hintergrundbetrieb). • quantifizieren die Eigenschaften und Auswirkungen auf den periodischen Teil des Echtzeitsystems. • formulieren die Grundlagen des Slack-Stealing. • beschreiben den Einsatz von bandweite-bewahrenden Zustellern. • unterscheiden aufschiebbare Zusteller und Sporadic Server (SpSL und POSIX). • wenden eine Übernahmeprüfung bei sporadischen Aufgaben mittels dichte- oder schlupfbasierten Akzeptanztests an. • arbeiten einen strukturierter Ablaufplan (Rahmen) aus und untersuchen den Einsatz von Slack-Stealing.

		• ermitteln gerichtete Abhängigkeiten und Rangfolgen in Echtzeitanwendungen (Abhängigkeits- und Aufgabengraph). • stellen Umsetzungsalternativen für Abhängigkeiten einander gegenüber (naiv, implizit, explizit). • beschreiben das Konzept der zeitlichen Domänen und physikalischer bzw. logischer Ereignisse. • übertragen Abhängigkeiten auf das Problem der Ablaufplanung (modifiziere Auslösezeitpunkt/Termin, Phasenversatz). • konzipieren Rangfolge und aperiodische Steuerung in eCos. • implementieren einen aperiodischer Moduswechsel mit Zustandsüberführung in eCos. • gestalten einen Signal-Trigger für das entwickelte softwarebasierten Oszilloskops. • konzipieren explizite Synchronisation mittels Nachrichten in eCos. • wenden die Grundlagen von Wettstreit um Betriebsmitteln, Konkurrenz und Konfliktsituationen (kritische Abschnitte, (un)kontrollierte Prioritätsumkehr) an. • beschreiben echtzeitfähige Synchronisationsprotokolle (NPCS, PI, PCP). • nennen die Vor- und Nachteile der Techniken (transitive Blockung, Verklemmungen). • hinterfragen die Vereinfachung des PCP durch stapelbezogene Grenzprioritäten. • bestimmen die Ablaufplanung unter Berücksichtigung von Blockierungszeiten und Selbstsuspendierung. • implementieren Zugriffskontrolle (NPCS, PI, PCP) in Echtzeitanwendungen mit eCos. • analysieren Blockade für die Zugriffskontrolle in eCos. • erläutern die Anforderungen an verteile Echtzeitsysteme (Komposition, Erweiterbarkeit, Komplexität, Ereignis- vs. Zustandsnachricht). • fassen die Grundlagen von Knoten, Netzwerkschnittstellen und Netzübergängen sowie die Konzepte der expliziten und impliziten Flusskontrolle zusammen. • erschließen sich typische Probleme (zeitliche Analyse, Beobachtbarkeit, Synchronisation, Rangfolge) und Fehlerquellen bei der Programmierung von Echtzeitanwendungen. • können in Gruppen kooperativ und effektiv arbeiten. • können ihre Entwurfs- und Implementierungsentscheidungen kompakt präsentieren und argumentativ vertreten. • reflektieren ihre Entscheidungen kritisch und leiten Alternativen ab. • können offen und konstruktiv mit Schwachpunkten in der Konzeption wie Umsetzung umgehen.
7	**Voraussetzungen für die Teilnahme**	Für eine erfolgreiche Teilnahme an der Veranstaltung sind grundlegende Programmierkenntnisse in C/C++ erforderlich.

		Diese können durch den (empfohlenen) Besuch entsprechender Grundlagenveranstaltungen oder im Eigenstudium erworben sein, eine formale Voraussetzung besteht in diesem Zusammenhang nicht.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	Unregelmäßig
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 90 h Eigenstudium: 135 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• Hermann Kopetz. Real-Time Systems: Design Principles for Distributed Embedded Applications. Kluwer Academic Publishers, 1997. • Jane W. S. Liu. Real-Time Systems. Prentice-Hall, Inc., 2000. • Wolfgang Schröder-Preikschat. System-programmierung. Vorlesungsfolien. 2006.

1	**Modulbezeichnung** 707303	**Echtzeitsysteme (Vorlesung mit Übungen)** (Real-time systems)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Rechnerübungen zu Echtzeitsysteme (2 SWS) Vorlesung: Echtzeitsysteme (2 SWS) Übung: Übungen zu Echtzeitsysteme (2 SWS)	- - 2,5 ECTS
3	Lehrende	Peter Wägemann Simon Schuster Tim Rheinfels	

4	**Modulverantwortliche/r**	Peter Wägemann
5	**Inhalt**	Videobearbeitung in Echtzeit, Echtzeitstrategiespiel, echtzeitfähig - der Begriff Echtzeit ist wohl einer der am meisten strapazierten Begriffe der Informatik und wird in den verschiedensten Zusammenhängen benutzt. Diese Vorlesung beschäftigt sich mit dem Begriff Echtzeit aus der Sicht von Betriebssystemen - was versteht man eigentlich unter dem Begriff Echtzeit im Betriebssystemumfeld, wo und warum setzt man sog. Echtzeitbetriebssysteme ein und was zeichnet solche Echtzeitbetriebssysteme aus? In dieser Vorlesung geht es darum, die oben genannten Fragen zu beantworten, indem die grundlegenden Techniken und Mechanismen vermittelt werden, die man im Betriebssystemumfeld verwendet, um Echtzeitsysteme und Echtzeitbetriebssysteme zu realisieren. Im Rahmen dieser Vorlesung werden unter anderem folgende Themen behandelt: • zeitgesteuerte und ereignisgesteuerte Systeme • statische und dynamische Ablaufplanungsverfahren • Fadensynchronisation in Echtzeitbetriebssystemen • Behandlung von periodischen und nicht-periodischen Ereignissen In den begleitenden Übungen werden die in der Vorlesung vorgestellten Techniken bei der Entwicklung eines kleinen Echtzeitsystems praktisch umgesetzt.
6	**Lernziele und Kompetenzen**	Studierende, die das Modul erfolgreich abgeschlossen haben: • unterscheiden die verschiedenen Komponenten eines Echtzeitsystems. • bewerten die Verbindlichkeiten von Terminvorgaben (weich, fest, hart). • erläutern die Zusammensetzung des Laufzeitverhaltes einer Echtzeitanwendung. • klassifizieren die Berührungspunkte zwischen physikalischem Objekt und kontrollierendem Echtzeitsystem. • interpretieren die Zeitparameter des durch das Echtzeitrechensystem zu kontrollierenden Objekts.

		• nennen die Zeitparameter des zugrundeliegenden Rechensystems (Unterbrechungslatenz, Ausführungszeit, ...). • unterscheiden synchrone und asynchrone Programmunterbrechung (insbesondere Trap/Interrupt, Ausnahmebehandlung und Zustandssicherung). • skizzieren die Verwaltungsgemeinkosten des schlimmsten Falls. • entwickeln in der Programmiersprache C und wenden die GNU Werkzeugkette für den ARM Cortex M4 Microcontroller an. • erstellen Echtzeitanwendungen auf Basis der eCos OS-Schnittstelle • ordnen die Strukturelemente von Echtzeitanwendungen zu: Aufgabe, Arbeitsauftrag und Faden. • erläutern die Implikationen von zeitlichem Mehrfachbetrieb auf die Verwaltungsgemeinkosten. • unterscheiden die Umsetzungsalternativen zur Ablaufsteuerung und die Trennung der Belange in Einplanung (Strategie) und Einlastung (Mechanismus). • benennen die grundsätzliche Verfahren der Ablaufsteuerung (taktgesteuert, reihum, vorranggesteuert). • erklären die grundlegenden Zeitparameter einer Aufgabe (Auslösezeitpunkt, Termin, Antwortzeit, Latenz, Ausführungszeit, Schlupfzeit). • unterscheiden die Grundlagen der Planbarkeit (gültig vs. zulässig, Optimalität von Einplanungsalgorithmen). • beschreiben den Unterschied zwischen konstruktiver und analytischer Einhaltung von Terminen-. • vergleich die Möglichkeiten (statisch, dynamisch) der zeitliche Analyse von Echtzeitanwendungen. • erklären die Grundlagen und Beschränkungen von dynamischer (worst-case?) und statischer WCET-Analyse (makroskopisch und mikroskopisch). • illustrieren Lösungsverfahren zur Bestimmung des längsten Ausführungspfads (Timing Schema, IPET). • erstellen Zeitmessung mittels Zeitgeber / Oszilloskop und bestimmen den längsten Pfad durch Code-Review. • erproben werkzeuggestützte WCET-Analyse mittels des absint aiT Analysewerkzeugs. • beschreiben die Grundlagen der Abfertigung periodischer Echtzeitsysteme (Periode, Phase, Hyperperiode). • skizzieren das periodische Modell und dessen Folgen (Entwicklungskomfort vs. Analysierbarkeit). • erklären die ereignisgesteuerte Ausführung (feste und dynamische Priorität, Verdrängbarkeit) mittels ereignisorienterter Planer (Berechnungskomplexität, MLQ-Scheduler, O(1)-Scheduler). • unterscheiden die zeitgesteuerte Ausführung (Busy Loop, Ablaufplan)und die Abfertigung von Arbeitsaufträgen im Abfrage- bzw. Unterbrecherbetrieb.

		• wenden die Grundlagen der ereignisgesteuerten Ablaufplanung periodischer Echtzeitsysteme an. • unterscheiden Verfahren zur statischen (RM, DM) und dynamischen Prioritätsvergabe (EDF, LRT, LST). • nennen den Unterschied zwischen Anwendungs- und Systemebene (Mehrdeutigkeit von Prioritäten). • erläutern den Optimalitätsnachweis des RM-, DM- und EDF-Algorithmus und dessen Ausnahmen. • beschreiben grundlegende Verfahren zur Planbarkeitsanalyse (CPU-Auslastung, Antwortzeitanalyse). • implementieren komplexe Aufgabensysteme in eCos. • unterscheiden die Grundlagen der zeitgesteuerten Ablaufplanung periodischer Echtzeitsysteme. • erstellen regelmäßige, zyklische Ablaufpläne (cyclic executive model, Rahmen). • vergleich Methoden der manuellen und algorithmischen Ablaufplanung. • unterscheiden optimale von heuristischen Verfahren (List Scheduling, Branch & Bound). • diskutieren die Konsequenzen eines Betriebswechsels in Echtzeitsystemen. • erstellen takt- beziehungsweise ereignisgesteuerte Abläufe in eCos beziehungsweise tt-eCos. • klassifizieren die Grundlagen der Abfertigung nicht-periodischer Echtzeitsysteme (minimale Zwischenankunftszeit). • definieren die Verbindlichkeiten von nicht-periodischen Aufgaben (aperiodisch, sporadisch) • zeigen die sich ergebenden Restriktionen des periodischen Modells (Mischbetrieb, Prioritätswarteschlangen, Übernahmeprüfung) auf. • beschreiben die Basistechniken des Laufzeitsystems (Zusteller, Unterbrecherbetrieb, Hintergrundbetrieb). • quantifizieren die Eigenschaften und Auswirkungen auf den periodischen Teil des Echtzeitsystems. • formulieren die Grundlagen des Slack-Stealing. • beschreiben den Einsatz von bandweite-bewahrenden Zustellern. • unterscheiden aufschiebbare Zusteller und Sporadic Server (SpSL und POSIX). • wenden eine Übernahmeprüfung bei sporadischen Aufgaben mittels dichte- oder schlupfbasierten Akzeptanztests an. • arbeiten einen strukturierter Ablaufplan (Rahmen) aus und untersuchen den Einsatz von Slack-Stealing. • ermitteln gerichtete Abhängigkeiten und Rangfolgen in Echtzeitanwendungen (Abhängigkeits- und Aufgabengraph). • stellen Umsetzungsalternativen für Abhängigkeiten einander gegenüber (naiv, implizit, explizit). • beschreiben das Konzept der zeitlichen Domänen und physikalischer bzw. logischer Ereignisse.

		• übertragen Abhängigkeiten auf das Problem der Ablaufplanung (modifiziere Auslösezeitpunkt/Termin, Phasenversatz). • konzipieren Rangfolge und aperiodische Steuerung in eCos. • implementieren einen aperiodischer Moduswechsel mit Zustandsüberführung in eCos. • wenden die Grundlagen von Wettstreit um Betriebsmitteln, Konkurrenz und Konfliktsituationen (kritische Abschnitte, (un)kontrollierte Prioritätsumkehr) an. • beschreiben echtzeitfähige Synchronisationsprotokolle (NPCS, PI, PCP). • nennen die Vor- und Nachteile der Techniken (transitive Blockung, Verklemmungen). • hinterfragen die Vereinfachung des PCP durch stapelbezogene Grenzprioritäten. • bestimmen die Ablaufplanung unter Berücksichtigung von Blockierungszeiten und Selbstsuspendierung. • implementieren Zugriffskontrolle (NPCS, PI, PCP) in Echtzeitanwendungen mit eCos. • erläutern die Anforderungen an verteile Echtzeitsysteme (Komposition, Erweiterbarkeit, Komplexität, Ereignis- vs. Zustandsnachricht). • fassen die Grundlagen von Knoten, Netzwerkschnittstellen und Netzübergängen sowie die Konzepte der expliziten und impliziten Flusskontrolle zusammen. • erschließen sich typische Probleme (zeitliche Analyse, Beobachtbarkeit, Synchronisation, Rangfolge) und Fehlerquellen bei der Programmierung von Echtzeitanwendungen. • können in Gruppen kooperativ und effektiv arbeiten. • können ihre Entwurfs- und Implementierungsentscheidungen kompakt präsentieren und argumentativ vertreten. • reflektieren ihre Entscheidungen kritisch und leiten Alternativen ab. • können offen und konstruktiv mit Schwachpunkten in der Konzeption wie Umsetzung umgehen.
7	**Voraussetzungen für die Teilnahme**	Für eine erfolgreiche Teilnahme an der Veranstaltung sind grundlegende Programmierkenntnisse in C/C++ erforderlich. Diese können durch den (empfohlenen) Besuch entsprechender Grundlagenveranstaltungen oder im Eigenstudium erworben sein, eine formale Voraussetzung besteht in diesem Zusammenhang nicht.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio

11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	Unregelmäßig
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 90 h Eigenstudium: 60 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• Hermann Kopetz. Real-Time Systems: Design Principles for Distributed Embedded Applications. Kluwer Academic Publishers, 1997. • Jane W. S. Liu. Real-Time Systems. Prentice-Hall, Inc., 2000. • Wolfgang Schröder-Preikschat. Softwaresysteme 1. Vorlesungsfolien. 2006.

1	**Modulbezeichnung** 93015	**Einführung in die moderne Kryptographie** (Introduction to modern cryptography)	**7,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Einführung in die moderne Kryptographie (4 SWS) Übung: Einführung in die moderne Kryptographie (Übung) (2 SWS)	5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr. Dominique Schröder	

4	**Modulverantwortliche/r**	Carina Harrius Prof. Dr. Dominique Schröder
5	**Inhalt**	This course gives a comprehensive introduction to modern cryptography. The course also serves as a base for other courses on cryptography that are offered by the chair. The topics covered are the following: • Information theoretic security • Computational security • Private key Encryption • Message Authentication Codes • Hash functions • Public key Encryption • Digital Signatures More advanced topics may be covered if time permits.
6	**Lernziele und Kompetenzen**	On successfully passing the course, the student is guaranteed to be knowledgeable on the basic concepts of provable security.
7	**Voraussetzungen für die Teilnahme**	No previous knowledge in Cryptography or computer Security is required.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Variabel
11	**Berechnung der Modulnote**	Variabel (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 165 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Introduction to Modern Cryptography Jonathan Katz and Yehuda Lindell 2nd Edition (2014) (Chapman & Hall/CRC Cryptography and Network Security Series)

		ISBN-13: 978-1466570269

1	**Modulbezeichnung** 96190	**Entwurf Integrierter Schaltungen II/Technologie integrierter Schaltungen** (Design of integrated circuits II/technology of integrated circuits)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übungen zu Entwurf Integrierter Schaltungen II (2 SWS) Vorlesung: Halbleitertechnologie I - Technologie integrierter Schaltungen (3 SWS) Übung: Übung zu Halbleitertechnologie I - Technologie integrierter Schaltungen (1 SWS)	- 5 ECTS -
3	Lehrende	Tobias Rumpel Florian Deeg Prof. Dr.-Ing. Jörg Schulze Jannik Schwarberg	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Jörg Schulze
5	**Inhalt**	In diesem Modul werden die wesentlichen Technologieschritte zur Herstellung elektronischer Halbleiterbauelemente und integrierter Schaltungen behandelt. Ausgehend von der Frage nach den relevanten Parametern chemischer und physikalischer Herstellungsprozesse werden zu Beginn die Verfahren und Methoden zur Herstellung von einkristallinen Siliziumkristallen besprochen. Anschließend werden die physikalischen und chemischen Grundlagen der Oxidation, der Dotierverfahren Diffusion und Ionenimplantation sowie der physikalischen und chemischen Gasphasenabscheidung von dünnen Schichten behandelt. Eine Einführung in die relevanten Lithographie- und Strukturierungsverfahren beendet den Kanon der wesentlichen Technologieschritte zur Herstellung elektronischer Halbleiterbauelemente. Ergänzend dazu werden Sequenzen von Prozessabläufen, wie sie heute bei der Herstellung von hochintegrierten Schaltungen wie Mikroprozessoren oder Speichern verwendet werden, besprochen.
6	**Lernziele und Kompetenzen**	Die Studierenden Fachkompetenz Anwenden • beschreiben die Technologieschritte und notwendigen Prozessgeräte • erklären die physikalischen und chemischen Vorgänge bei der Herstellung von Integrierten Schaltungen Evaluieren (Beurteilen)

		• ermitteln en Einfluss von Prozessparametern und können Vorhersagen für Einzelprozesse ableiten • sind in der Lage, verschiedene Herstellungsschritte hinsichtlich ihrer Vor- und Nachteile bzgl. der hergestellten Schichten, Strukturen oder Bauelemente zu beurteilen
7	**Voraussetzungen für die Teilnahme**	Kenntnisse aus dem Bereich Halbleiterbauelemente (Pflichtveranstaltung im Bachelorstudiengang EEI und Mechatronik)
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	
11	**Berechnung der Modulnote**	
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• S. M. Sze: VLSI - Technology, MacGraw-Hill, 1988 • C. Y. Chang, S. M. Sze: ULSI - Technology, MacGraw-Hill, 1996 • D. Widmann, H. Mader, H. Friedrich: Technology of Integrated Circuits, Springer Verlag, 2000 • Hong Xiao: Introduction to Semiconductor Manufacturing Technology, Prentice Hall, 2001

1	**Modulbezeichnung** 96200	**Entwurf von Mixed-Signal-Schaltungen** (Design of mixed-signal circuits)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übungen zu Entwurf von Mixed-Signal-Schaltungen (2 SWS) Vorlesung: Entwurf von Mixed-Signal-Schaltungen (2 SWS)	- 5 ECTS
3	Lehrende	Tobias Rumpel Florian Deeg Prof. Dr.-Ing. Sebastian Sattler	

4	**Modulverantwortliche/r**	Peter Meisel Prof. Dr.-Ing. Sebastian Sattler
5	**Inhalt**	Es werden Methoden zur Analyse und Synthese von Phänomenen behandelt, welche aus sogenannten Rückkopplungen in gemischt analog-digitalen Systemen entstehen. Es wird an Hand eines allgemeinen Transistormodells abstrahiert, und Beispiele aus der Integrierten Schaltungs- und Systemtechnik erarbeitet. • Modellierung aktiver Bauelemente • Grundschaltungen des allgemeinen Transistors • Abstraktion der Rückkopplung • Analyse der Stabilität im Frequenz- und Zeitbereich • Kompensationstechniken im Frequenzbereich • Grundschaltungen von Rückkopplungen • Harmonische Verzerrungen • Rauschen • Beispiele von Rückkopplungen
6	**Lernziele und Kompetenzen**	Analysieren • Die Studierenden können die verschiedenste Strukturen für analoge integrierte Schaltungen entwickeln, analysieren und bewerten. Erschaffen • Die Studierenden gewinnen einen Überblick über die wichtigsten Methoden und Verfahren für Analyse und Entwurf von analogen rückgekoppelten Schaltungen.
7	**Voraussetzungen für die Teilnahme**	Schaltungstechnik, Entwurf Integrierter Schaltungen I, o.ä.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)

12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	G. Palumbo, S. Pennisi, Feedback Amplifiers, Theory and Design, Springer 2009

1	**Modulbezeichnung** 43400	**Entzerrung und adaptive Systeme in der digitalen Übertragung** (Equalisation and adaptive systems for digital communications)	**2,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Equalization and Adaptive Systems for Digital Communications (2 SWS)	2,5 ECTS
3	Lehrende	Prof. Dr. Wolfgang Gerstacker	

4	**Modulverantwortliche/r**	Prof. Dr. Wolfgang Gerstacker
5	**Inhalt**	Bei der digitalen Übertragung spielen Kanalverzerrungen aufgrund ständig steigender Datenraten eine immer grössere Rolle. Bei vielen Anwendungen müssen für eine zuverlässige Übertragung komplexe Entzerrverfahren eingesetzt werden. Dies gilt sowohl für die leitungsgebundene als auch die drahtlose Kommunikation. Z.B. werden in der xDSL-Systemfamilie (Digital Subscriber Lines), die eine schnelle digitale Übertragung über Ortsanschlussleitungen gewährleistet, oft entscheidungsrückgekoppelte Entzerrverfahren oder Vorcodierungsverfahren eingesetzt und beim Mobilfunkstandard GSM und seiner Weiterentwicklung EDGE (Enhanced Data Rates for GSM Evolution) Maximum-Likelihood-Sequenzschätzung bzw. zustandsreduzierte Entzerrung. Eng im Zusammenhang mit der eigentlichen Entzerrung stehen Adaptionsverfahren, mit denen die Parameter des Entzerrers optimal an den Übertragungskanal angepasst werden können. Lernziel: Ziel der Vorlesung ist eine umfassende Darstellung gebräuchlicher Entzerrungs- und Adaptionsverfahren. Den Teilnehmern sollen fundierte Kenntnisse der verschiedenen Verfahren vermittelt werden, die sie zu deren sinnvollem Einsatz in der Praxis befähigen. Content: Channel distortions are playing an increasingly important role in digital transmission due to constantly increasing data rates. In many applications, complex equalization techniques must be used for a reliable transmission. This applies to both wired and wireless communication. For example, decision feedback equalization or precoding techniques are often used in the xDSL (Digital Subscriber Lines) system family, which ensures fast digital transmission over local subscriber loops, and the GSM system and its advanced version EDGE (Enhanced Data Rates for GSM Evolution) employ maximum likelihood sequence estimation and state-reduced equalization. Closely

		related to the task of equalization are adaptation methods with which the parameters of the equalizer can be optimally adjusted to the transmission channel. Objective: The aim of the lecture is a comprehensive presentation of common equalization and adaptation methods. The participants should acquire an in-depth knowledge of the various procedures which enables them to make meaningful design decisions in practice.
6	**Lernziele und Kompetenzen**	Die Studierenden • beschreiben verschiedene Verfahren zur Entzerrung frequenzselektiver Übertragungskanäle wie lineare Entzerrung, entscheidungsrückgekoppelte Entzerrung und Maximum-Likelihood-Sequenzschätzung, • setzen die verschiedenen Ansätze in Blockdiagramme um und optimieren deren Komponenten, • vergleichen Entzerrverfahren hinsichtlich ihrer Leistungsfähigkeit, charakterisiert durch die Fehlerrate, und Komplexität, • wählen geeignete Verfahren für verschiedene Anwendungen wie leitungsgebundene und drahtlose Übertragung aus, • entwerfen neuartige Verfahren für gegebene Anforderungen, • formulieren Adaptionsalgorithmen zur automatischen Anpassung des Empfängers eines Übertragungssystems an den Kanal, • ordnen Entzerrverfahren einen geeigneten Adaptionsalgorithmus zu. Learning Objectives and Competences: The students - describe various methods for equalizing frequency-selective transmission channels such as linear equalization, decision feedback equalization and maximum likelihood sequence estimation, - realize various approaches in block diagrams and optimize their components, - compare equalization methods in terms of their performance, characterized by the error rate, and complexity, - select suitable methods for various applications such as wired and wireless transmission, - design novel schemes for given requirements, - formulate adaptation algorithms for automatic adaptation of the receiver of a transmission system to the channel,

		- assign suitable adaptation algorithms to equalization schemes.
7	**Voraussetzungen für die Teilnahme**	Vorkenntnisse in Systemtheorie und digitaler Signalverarbeitung, sowie entweder der Vorlesung Nachrichtentechnische Systeme oder Digitale Übertragung sind für die Teilnahme hilfreich.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Gerstacker, W.: Skriptum zur Vorlesung Entzerrung und adaptive Systeme in der digitalen Übertragung. Huber, J.: Trelliscodierung, Springer Verlag, Berlin, 1992. Benedetto, S., Biglieri, E.: Principles of Digital Transmission with Wireless Applications, Kluwer Academic Publishers, New York, 1999. Proakis, J. G.: Digital Communications. McGraw-Hill, New York, 3. ed., 1995. Haykin, S.: Adaptive Filter Theory, Prentice Hall, Upper Saddle River, NJ, 3. ed., 1996.

1	**Modulbezeichnung** 393750	**Fortgeschrittene forensische Informatik** (Advanced forensic computing)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Fortgeschrittene forensische Informatik - Übung (2 SWS) Vorlesung: Fortgeschrittene forensische Informatik (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr.-Ing. Felix Freiling	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Felix Freiling
5	**Inhalt**	Die Vorlesung "Fortgeschrittene Forensische Informatik" ist eine forschungsorientierte Weiterführung der Vorlesung "Forensische Informatik". Behandelt werden forschungsnahe Themen aus dem Bereich der forensischen Informatik, beispielsweise Browser- und Anwendungsforensik, Netzwerkforensik, Hauptspeicheranalyse, Analyse von Mobiltelefonen, theoretische Modelle. Der Stoff der Vorlesung wird beispielhaft durch praktische Aufgaben in der gleichlautenden Übung vertieft.
6	**Lernziele und Kompetenzen**	Die Studierenden erhalten Einblicke in aktuelle Forschung im Bereich der forensischen Informatik. Sie können die wesentlichen Fragestellungen, Probleme und Lösungsmoeglichkeiten wiedergeben und in den Kontext digitaler Ermittlungen einordnen. Sie können Forschungsergebnisse qualitativ bewerten und vergleichen. Sie lernen, die Forschungsergebnisse auf konkrete Problemstellungen im Bereich der IT-Forensik anzuwenden und die entstehenden Ergebnisse vor Fachexperten zu vertreten.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Wird in der Vorlesung bekannt gegeben. Wird in der Vorlesung bekannt gegeben.

1	**Modulbezeichnung** 96401	**Globale Navigationssatellitensysteme** (Global navigation satellite systems)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übung zu Globale Navigationssatellitensysteme (1 SWS) Vorlesung: Globale Navigationssatellitensysteme (3 SWS)	- 5 ECTS
3	Lehrende	Prof. Dr. Jörn Thielecke	

4	**Modulverantwortliche/r**	Prof. Dr. Jörn Thielecke
5	**Inhalt**	*Hinweis:* 1. Mehrere Übungsstunden werden rechnergestützt (Python) sein, um den Vorlesungsstoff durch eigene praktische Erfahrung zu vertiefen. 2. Eine Laborbesichtigung beim Fraunhofer Institut für Integrierte Schaltungen soll den Studierenden Einblick in die einschlägigen Arbeiten zu GPS und Galileo geben. *Inhalte:* *|1. Überblick: Signale und Systeme|* • Einführung • GPS Global Positioning System • Galileo • Satellitenergänzungssysteme: EGNOS, WAAS, LAAS • Mathematische Grundlagen: Navigationssignale, Gold Codes, Cramer-Rao-Schranke für Laufzeitmessungen *|2. Grundlagen und Funktiosweise der Satellitenortung|* • Koordinatensysteme, Zeitsysteme und Orbits • Ausbreitungsbedingungen und Fehlerquellen • Positions-, Geschwindigkeits- und Zeitschätzung • Hochgenaue Positionsschätzung mittels Trägerphase *|3. GNSS Empfänger|* • Signalkonditionierung • Leistungsfähigkeit der GPS- und Galileo-Signale • Releschleifen zur Signalverfolgung
6	**Lernziele und Kompetenzen**	1. Eine Übersicht über die Möglichkeiten von GPS und Galileo soll die Beurteilungsfähigkeit der Studierenden für neue Anwendungen schärfen. 2. Durch vertiefte Kenntnisse der Grundlagen, Funktionsweise und Fehlerquellen sollen die Studierenden die gelösten Herausforderungen und die Grenzen von GPS und Galileo einschätzen lernen.

		3. Die Studierenden sollen ein nachrichtentechnisches Verständnis für die Funktionsweise eines GPS-Empfängers erlangen.
7	**Voraussetzungen für die Teilnahme**	Keine formalen Voraussetzungen, grundlegende Kenntnisse erforderlich in: linearer Algebra, Signal- & Systemtheorie, Wahrscheinlichkeitstheorie
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%) Bei bestandener Prüfung wird die Note um eine Teilnotenstufe (z.B. von 2,0 auf 1,7) verbessert, wenn Sie: 1. Mindestens 75% der Hausaufgaben bestanden haben UND 2. Mindestens 75% der Rechnerübungen erfolgreich absolviert haben. Eine Note besser als 1,0 wird nicht vergeben.
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	1. Pratap Misra, Per Enge, Global Positioning System", Ganga-Jamuna Press, 2001 2. E.D. Kaplan, C.J. Hegarty, Understanding GPS Principles and Applications"Artech House, 2. Auflage, 2006 3. Werner Mansfeld, "Satellitenortung und Navigation, Vieweg, 2004

1	**Modulbezeichnung** 44240	**Grundlagen des Übersetzerbaus** (Foundations of compiler construction)	**7,5 ECTS**
2	Lehrveranstaltungen	Übung: Übungen zu Grundlagen des Übersetzerbaus (2 SWS) Vorlesung: Grundlagen des Übersetzerbaus (2 SWS)	- 7,5 ECTS
3	Lehrende	Patrick Kreutzer Michel Schmid Florian Mayer	

4	**Modulverantwortliche/r**	Prof. Dr. Michael Philippsen
5	**Inhalt**	***[Deutsch:]*** Auf den ersten Blick erscheint es wenig sinnvoll, sich mit Übersetzerbau zu beschäftigen. Andere Themen scheinen wesentlich näher an der direkten Anwendbarkeit in der industriellen Praxis. Der erste Blick täuscht: • Übersetzer gehören wohl zu den am gründlichsten studierten mittelgroßen sequentiellen Software-Systemen. Man kann viel aus den Erfahrungen lernen, die im Laufe der Jahre gesammelt wurden. • In den Übungen, die die Vorlesung begleiten, werden Sie selbst einen (kleinen) Übersetzer entwickeln. • Für viele Teilnehmer wird dieses Projekt das erste größere Software-Projekt sein. Viele der Algorithmen aus dem Grundstudium werden angewendet. • Bei jedem von Ihnen verwendeten Übersetzer gehen Sie in der Regel davon aus, dass richtiger Coder erzeugt wird. In der Vorlesung erfahren Sie, wie das geforderte hohe Maß an Korrektheit und Zuverlässigkeit erreicht wird. • Sie erlangen ein Verständnis für Konzepte von Programmiersprachen und verstehen, welcher Maschinen-Code aus Sprachkonstrukten gemacht wird. Mit diesem Wissen im Hinterkopf verbessern Sie Ihre Fähigkeit, gute und effiziente Programme zu schreiben. • Übersetzer werden nicht nur für Programmiersprachen benötigt. Spezielle Übersetzer braucht man in vielen Bereichen des täglichen Informatik-Lebens z.B. zur Textformatierung, für Programmtransformationen, für aspektorientiertes Programmieren, für die Verarbeitung von XML, ... • Es gehört zu einer Ingenieur-Ausbildung, in der Lage zu sein, diejenigen Werkzeuge selbst zu fertigen, die man verwendet. Für Informatiker gehört daher ein Verständnis vom Innenleben eines Übersetzers zum Rüstzeug. **Fokus der Lehrveranstaltung:**

Es werden Konzepte und Techniken der Übersetzerkonstruktion aus Sicht eines Übersetzerbauers und entlang der wesentlichen Arbeitsschritte eines Übersetzers (Frontend; Mittelschicht; Backend) vorgestellt. Übungen und Praxisaufgaben ergänzen die Vorlesung. Hier entwickeln die Studierenden auf der Basis eines vorgegebenen Programmrahmens einen eigenen Übersetzer für die Programmiersprache e2, die speziell für den Übersetzerbau-Vorlesungszyklus entworfen wurde.

Behandelte Themenfelder:

- Prinzipien der Übersetzung imperativer Programmiersprachen
- Struktur eines Übersetzers
- Symbolentschlüssler (Scanner) und Zerteiler (Parser)
- Abstrakter Syntaxbaum (AST)
- Besuchermuster
- AST-Transformationen, Entzuckerung
- Symboltabellen und Sichtbarkeitsbereiche
- Semantische Analyse: Namensanalyse, Typprüfung
- Übersetzung von arithmetischen Ausdrücken und Kontrollflusskonstrukten in registerbasierte oder stapelbasierte Zwischensprachen
- Übersetzung von Methoden und Methodenaufrufen; Methodenschachteln
- Übersetzung objektorientierter Sprachen mit Einfachvererbung, Schnittstellen und Mehrfachvererbung
- Methodenauswahl in Java (überladene und überschriebene Methoden)
- Code-Generierung nach Sethi-Ullmann, Graham-Glanville, per Baumtransformation sowie mit Hilfe dynamischer Programmierung
- Registerallokation mit lokalen Techniken und mit Graphfärbung
- Instruktionsanordnung mit "list scheduling"
- Debugger

Themen der Vorlesungseinheiten:

1. Einführung (Überblick, modulare Struktur von Übersetzern, Frontend, Mittelschicht, Backend), Bootstrapping)

2. Symbolentschlüssler (Lexer) und Zerteiler (Parser), (Token, Literale, Symboltabelle, Grammatikklassen (LK(k), LL(k), ...), konkreter Syntaxbaum, Shift-Reduce-Parser)

3. AST und semantische Analyse (abstrakter Syntaxbaum, Besuchermuster, Double Dispatch, Sichtbarkeitsbereiche, Definitionstabelle)

4. Typkonsistenz (Typsicherheit, Typsystem, Typüberprüfung, Typberechnung, Typkonvertierung, attributierte Grammatiken)

5. AST-Transformationen (Transformationsschablonen für Ausdrücke, Transformation innerer und generischer Klassen)

6. Transformation in Zwischensprache (registerbasiert versus stapelbasiert, Übersetzung von arithmetischen Ausdrücken, Zuweisungen, mehrdimensionalen Feldern, struct-Datentypen und Kontrollflussstrukturen (einschließlich Kurzschlussauswertung))

7. Methodenschachteln und Kellerrahmen (relative Adressen, call by value/reference/name, geschachtelte Funktionen, Funktionszeiger,

Stack- und Framepointer, Funktionsaufruf, Prolog, Epilog)

8. Objektorientierte Sprachen I: Einfachvererbung (Symbol- und Typanalyse, Methodenauswahl mit Überschreiben und Überladen, virtuelle Methodenaufrufe, Klassendeskriptoren, dynamische Typprüfung und -wandlung)

9. Objektorientierte Sprachen II: Schnittstellen und Mehrfachvererbung (Interface v-Tables, dynamische Typprüfung und -wandlung mit Interfaces, Interfaces mit Default-Implementierung, Diamantenproblem)
10. Einfache Code-Erzeugung (Code-Selektion nach Sethi-Ullman, Register-Allokation, Instruktionsreihenfolge, optimale Code-Erzeugung für Ausdrucksbäume)

11. Fortgeschrittene Code-Erzeugung (Baumtransformation, Graham-Glanville, dynamisches Programmieren)

12. Registerallokation (Leistungsabschätzung, Lebendigkeitsintervalle, Kollisions- und Interferenzgraph, Spilling, Färbungsheuristiken, Aufteilung von Lebendigkeitsintervallen, 2nd Chance Bin Packing, Registerverschmelzung)

13. Parallelismus auf Instruktionsebene, Instruktionsreihenfolge, Debugger (Konflikte im Instruktionsfließband, List Scheduling, Delay-Slots, Sprungzielvorhersage, ptrace, Unterbrechungs- und Beobachtungspunkte, DWARF)

Meilensteine der Übungsbetriebs:

Im Rahmen der Übungen (siehe entsprechende Lehrveranstaltung) werden die in der Vorlesung vorgestellten Konzepte und Techniken zur Implementierung eines Übersetzers in die Praxis umgesetzt. Ziel der Übungen ist es, bis zum Ende des Semesters einen funktionsfähigen Übersetzer für die Beispiel-Programmiersprache e2 zu implementieren. Ein Rahmenprogramm ist gegeben, das in fünf Meilensteinen um selbstentwickelte Schlüsselkomponenten zu erweitern ist.

Folgende Meilensteine sind zu erreichen:

Meilenstein 1: Grammatik, AST-Konstruktion: Antlr-Produktionen, AST-Besucherschnittschelle, generischer AST-Besucher für return und Schleifen, AST-Besucher zur Visualisierung.

Meilenstein 2: Symbolanalyse, Symboltabelle, Standardfunktionen, AST-Besucher für die Symbolanalyse.

Meilenstein 3: Konstantenfaltung per AST-Transformation, Typanalyse mit bottom-up AST-Besuch, der implizite Typwandlungen bei Bedarf ergänzt.

Meilenstein 4: AST-Besucher zur Erzeugung der Zwischensprachrepräsentation, Übersetzung von arithmetischen Ausdrücken, return, Zuweisungen, logischen Ausdrücken, Bedingungen und Schleifen.

Meilenstein 5.0: Speicherzuteilung: Festlegung und Umsetzung der ABI Aufrufkonvention, Zuweisung von Speicheradressen zu Variablen; Kellerrahmenallokation; caller-save und callee-save Register.

Meilenstein 5.1: Code-Erzeugung: Implementierung der e2 Standardbibliothek; IR-Besucher zur Erzeugung von Assembly-Code.

Für die Meilensteine 1-3 soll der Übersetzer sowohl Integer- als auch Gleitkomma-Arithmetik unterstützen. Für die nachfolgenden Meilensteine reicht Integer-Arithmetik.

[English:]

The lecture teaches concepts and techniques of compiler construction from a compiler developer view, following the structure of the compiler frontend, middle end, and backend. Exercise sessions and practical assignments complement the lecture; the students implement their own compiler (based on a framework) for the e2 programming language, which is designed for this series of compiler construction lectures.

Content Summary

- Principles of compiling imperative programming languages
- Structure of a compiler
- Scanner and parser
- Abstract syntax trees (ASTs)
- Visitor design pattern
- AST transformations, desugaring
- Symbol tables and scopes
- Semantic analysis: name analysis, type checking
- Compilation of arithmetic expressions and control flow structures to register-based and stack-based intermediate languages
- Compilation of functions and function calls, activation records
- Compilation of object-oriented languages with single inheritance, interfaces, and multiple inheritance
- Method resolution in Java (overloaded and overridden methods)
- Code generation with Sethi-Ullmann algorithm, Graham-Glanville algorithm, tree transformations, and dynamic programming
- Register allocation with local techniques and graph coloring
- Instruction scheduling with the list scheduling technique
- Debuggers

Lecture Topics

- 1. Introduction: Class overview, modular structure of compilers (front-, middle-, and backend), compilation bootstrapping
- 2. Lexer and Parser: Tokens, literals, symbol table, grammar classes (LR(k), LL(k), ...), concrete syntax tree, shift-reduce parser
- 3. ASTs and semantic analysis: Abstract syntax tree, visitor pattern, double dispatch, scopes, definition table
- 4. Type consistency: Type safety, type system, type checks, type inference, type conversions, attributed grammars
- 5. AST transformations: Transformation patterns (arithmetics), transformation of nested and generic classes
- 6. Intermediate representations: Types of IRs, arithmetic operations, assignments, multidimensional array access, structs, control flow instructions, short-circuit evaluation
- 7. Activation record and stack frame: Relative addresses, call by value/reference/name, nested functions, function pointers, stack pointer and frame pointer, function calls: prolog and epilog
- 8. Object-oriented languages: single inheritance: Symbol and type analysis, method selection with method overloading and overriding, virtual method calls, class descriptors, dynamic type checks and casts

- 9. Object-oriented languages II: interfaces, multiple inheritance: Interface v-tables, dynamic type checks and casts with interfaces, interfaces with default implementations and state, diamond problem, virtual inheritance
- 10. Basic code generation: Code selection, register allocation, instruction order, basic blocks, optimal code generation for expression trees
- 11. Optimized code selection: Code selection as tree transformation, Graham-Glanville code generators, dynamic programming
- 12. Optimized register allocation: Performance approximations, liveness analysis, collision and interference graph, register spilling, coloring heuristics, optimistic extension, live range splitting, register coalescing, data structures
- 13. Instruction level parallelism, instruction order, debugger: Data, structural, and control conflicts in CPU pipelines, list scheduling, delay slots, branch predictions, superscalar and VLIW architectures, ptrace, break- and watch-points, DWARF

Assignment Milestones

For the assignments of this course, the students put the concepts and techniques presented in the lecture for implementing a compiler into practice. The goal of the assignments is to implement a functional compiler for the e2 programming language by the end of the semester. The e2 language is specifically designed for educational purposes; the students obtain a description of the language.

A framework for the implementation is provided to the students. The students implement the core components of the compiler in five milestones.

All milestones need to be fulfilled to pass the module; the last milestone contains two tasks. In particular, the milestones are:

- Milestone 1: Grammar definition and construction of the AST: ANTLR productions, AST visitor interface, and generic AST visitor for array accesses and return and loop statements; AST visitor for AST visualization.
- Milestone 2: Name analysis: symbol table; declaring standard functions; AST visitor for name analysis.
- Milestone 3: Constant folding and type analysis: AST transformations for constant folding; AST visitor for bottom-up type analysis, adding AST nodes for implicit casts;
- Milestone 4: AST translation to intermediate representation: AST visitor to generate IR; translation of arithmetic, return, and assign statements, logical expressions, conditions, loops.
- Milestone 5.0: Memory assignment: definition and implementation of the ABI calling convention; memory

		assignment of variables; stack frame allocation; caller-save and callee-save registers. • Milestone 5.1: Code generation: implementation of the e2 standard library; IR visitor to generate assembly code. For milestones one through three, the compiler needs to support both integer and floating-point arithmetic. For the last two milestones, only integer arithmetic is required.
6	**Lernziele und Kompetenzen**	***[Deutsch:]*** Die Studierenden • nennen die typischen Aufgaben und Datenstrukturen eines Übersetzers • erläutern das Konzept des Bootstrapping • beschreiben Struktur und Arbeitsweise eines Abtasters (Scanner) und zeigen Grenzen und Problemfälle auf • wenden Grammatiken zur Konstruktion von Zerteilern (Parser) an • kennen die Komplexität eines Zerteilers für Java • beschreiben die wichtigsten Aufgaben der semantischen Analyse und wenden diese am Beispiel verschiedener Programmiersprachen (insbesondere Java) an • skizzieren typische AST-Transformationen am Beispiel des Java-Übersetzers • veranschaulichen die Grundzüge der Java-Kellermaschine und die zugehörige Transformation von Quell- zu Byte-Code • analysieren die Unterschiede zwischen Programmiersprachen hinsichtlich Felder und Verbund-Strukturen • erläutern die Verwendung von Stapel- und Kellerspeicher bei der Programmausführung • kennen verschiedene Maschineninstruktionssätze • optimieren die Registerverwendung vor der Generierung von Maschinencode • wenden das Verfahren von Graham & Glanville zur Erzeugung von Maschinencode an • erkennen Grenzen der Optimierung bei der Code-Generierung und analysieren alternative Strategien • beschreiben den Unterschied zwischen statischer und dynamischer Ablaufplanung • untersuchen Besonderheiten des Übersetzerbaus für objekt-orientierte Sprachen • ergänzen einen vorgegebenen Abtaster und abstrakten Syntaxbaum, um alle Sprachkonstrukte einer Beispielsprache zu unterstützen • implementieren Konstantenfaltung, den Aufbau der Symboltabelle und Typprüfung auf dem abstrakten Syntaxbaum • erzeugen Zwischencode aus dem abstrakten Syntaxbaum • bilden Kontrollstrukturen auf Sprünge ab

- veranschaulichen die Adressierung von (mehrdimensionalen) Feldern
- entwickeln Konventionen für Funktionsaufrufe und den Aufbau des Stacks
- berechnen Offsets fuer Variablen auf dem Stack.
- implementieren eine einfache Registervergabe.
- kennen Details verschiedener Prozessorarchitekturen
- generieren Maschinencode für mindestens eine Prozessorarchitektur
- implementieren eine Laufzeitbibliothek
- wenden Debugging für maschinennahen Code an

[English:]

Students who have successfully completed the module will have the ability to

- identify the components and data structures of a compiler
- explain the concept of bootstrapping
- describe the structure and operation of a lexer and show limitations and problem cases
- use grammars for the construction of parsers
- know the complexity of Java parsers
- describe the main tasks of semantic analysis and apply them to different programming languages (especially Java)
- outline typical AST transformations using the Java compiler as an example
- illustrate the basic features of the Java Virtual Machine (JVM) and the corresponding transformation from source to byte code
- analyze the differences between programming languages in terms of arrays and compound structures
- explain the use of stack memory in program execution
- know different machine instruction sets
- optimize register allocation before generating machine code
- apply the Graham-Glanville algorithm to generate machine code
- recognize limitations of optimization in code generation and to analyze alternative strategies
- describe the difference between static and dynamic scheduling
- examine features of compiler construction for object-oriented languages
- augment a given lexer and abstract syntax tree to support all language constructs in an example language
- implement constant folding, symbol table construction, and type checking on the abstract syntax tree
- generate intermediate code from the abstract syntax tree
- map control structures to jumps
- translate compound boolean expressions with shortcut evaluation

		• illustrate addressing of (multidimensional) arrays • design conventions for function calls and stack frame layout • calculate offsets for stack variables • implement a basic register allocation. • know details of different processor architectures • generate machine code for at least one processor architecture • implement a runtime library • apply debugging to machine code
7	**Voraussetzungen für die Teilnahme**	Participants of this lecture are expected to have profound skills in the following programming languages: • Java (assignments are implemented in Java) • Assembler
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Variabel
11	**Berechnung der Modulnote**	Variabel (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 50 h Eigenstudium: 175 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• Modern Compiler Implementation in Java", A.W. Appel, Cambridge University Press, 1998 • Compilers - Principles, Techniques and Tools", A. Aho, R. Sethi, J. Ullmann, Addison-Wesley, 2006 • Modern Compiler Design", D. Grune, H. Bal, C. Jacobs, Langendoen, Wiley, 2002

1	**Modulbezeichnung** 96750	**Hardware-Beschreibungssprache VHDL** (VHDL Hardware description language)	**2,5 ECTS**
2	Lehrveranstaltungen	Vorlesung mit Übung: Hardware-Beschreibungssprache VHDL (2 SWS)	2,5 ECTS
3	Lehrende	Jürgen Frickel	

4	**Modulverantwortliche/r**	Jürgen Frickel
5	**Inhalt**	Vorlesung mit integrierter Rechnerübung zur Syntax und zur Anwendung der Hardware-Beschreibungssprache VHDL (Very High Speed Integrated Circuit Hardware Description Language) nach dem Sprachstandard IEEE 1076-1987 und 1076-1993, Anwendung von VHDL zum Entwurf von FPGAs in der Praxis. • Konzepte und Konstrukte der Sprache VHDL • Beschreibung auf Verhaltens- und Register-Transfer-Ebene • Simulation und Synthese auf der Gatterlogik-Ebene • Verwendung professioneller Software-Tools (Xilinx Vivado) • Vorlesung mit integrierten Rechner-Übungen (Labs) • Kursmaterial ist englisch-sprachig, die Vorlesungssprache deutsch Zielgruppe sind Hörer aller Fachrichtungen, die sich mit dem Entwurf, Simulation und Synthese digitaler Systeme und Schaltungen beschäftigen wollen.
6	**Lernziele und Kompetenzen**	**Fachkompetenz** • Wissen Die Studierenden können Begriffe und Definitionen einer Hardware-Beschreibungssprache (hier VHDL) darlegen. • Verstehen Die Studierenden verstehen den Zusammenhang bzw. die Transformation zwischen einer Hardware-Struktur und deren Abbildung in einer Hardware-Beschreibungssprache in beiden Richtungen. • Analysieren Die Studierenden klassifizieren ein gewünschtes Systemverhalten, strukturieren dieses in Teilmodule, und realisieren die Teilmodule bzw. das System in der Hardware-Beschreibungssprache. • Evaluieren (Beurteilen) Die Studierenden schätzen VHDL-Modelle bezüglich des quantitativen und qualitativen Hardware-Aufwandes ein, überprüfen diese gegen vorliegende Randbedingungen (constraints), und vergleichen sie mit alternativen Lösungen.

		Lern- bzw. Methodenkompetenz Die theoretischen Inhalte der Sprache können durch Einsatz eines Simulations- und Synthesewerkzeuges im praktischen Einsatz selbständig verifiziert und deren Verständnis vertieft werden. **Sozialkompetenz** Die Studierenden stärken ihre Fähigkeit, vorliegende Aufgabenstellungen in Gruppenarbeit gemeinsam zu lösen.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich (60 Minuten)
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 502509	**Hardware-Software-Co-Design** (Hardware-software-co-design)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übungen zu Hardware-Software-Co-Design (2 SWS) Vorlesung: Hardware-Software-Co-Design (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Muhammad Sabih Tobias Hahn Dr.-Ing. Stefan Wildermann Prof. Dr.-Ing. Jürgen Teich	

4	**Modulverantwortliche/r**	Joachim Falk Prof. Dr.-Ing. Jürgen Teich
5	**Inhalt**	Zahlreiche Realisierungen eingebetteter Systeme (z.B. Mobiltelephone, Faxgeräte, Industriesteuerungen) zeichnen sich durch kooperierende Hardware- und Softwarekomponenten aus. Die Popularität solcher Realisierungsformen lässt sich begründen durch 1) die steigende Vielfalt und Komplexität heterogener Systeme, 2) die Notwendigkeit, Entwurfs- und Testkosten zu senken und 3) Fortschritte in Schlüsseltechnologien (Mikroelektronik, formale Entwurfsmethoden). Zum Beispiel bieten Halbleiterhersteller kostengünstige ASICs an, die einen Mikrocontroller und benutzerspezifische Peripherie und Datenpfade auf einem Chip integrieren. Die Synthese solcher Systeme wirft jedoch eine Reihe neuartiger Entwurfsprobleme auf, insbesondere 1) die Frage der Auswahl von Hardware- und Softwarekomponenten, 2) die Partitionierung einer Spezifikation in Hard- und Software, 3) die automatische Synthese von Interface- und Kommunikationsstrukturen und 4) die Verifikation und Cosimulation. 1) Überblick und Vergleich von Architekturen und Komponenten in Hardware/Software-Systemen. 2) Aufbau eines Compilers und Codeoptimierungsverfahren für Hardware und Software 3) Hardware/Software-Partitionierung (Partitionierung komplexer Systeme, Schätzungsverfahren, Performanzanalyse, Codegenerierung) 4) Interfacesynthese (Kommunikationsarten, Synchronisation, Synthese) 5) Verifikation und Cosimulation 6) Tafelübungen Content: Numerous realisations of embedded systems (e.g. mobile phones, fax machines, industrial controls) are characterised by cooperating hardware and software components. The popularity of such realisations can be explained by 1) the increasing diversity and complexity of heterogeneous systems, 2) the need to reduce design and testing costs, and 3) advances in key technologies (microelectronics, formal design methods). For example, semiconductor manufacturers

		offer low-cost ASICs that integrate a microcontroller and user-specific peripherals and data paths on a single chip. However, the synthesis of such systems raises a number of novel design problems, in particular 1) the issue of hardware and software component selection, 2) the partitioning of a specification into hardware and software, 3) the automatic synthesis of interface and communication structures, and 4) verification and cosimulation. 1) Overview and comparison of architectures and components in hardware/software systems. 2) Structure of a compiler and code optimisation procedures for hardware and software. 3) Hardware/software partitioning (partitioning of complex systems, estimation procedures, performance analysis, code generation) 4) Interface synthesis (communication types, synchronisation, synthesis) 5) Verification and cosimulation 6) Blackboard exercises
6	**Lernziele und Kompetenzen**	Fachkompetenz - Wissen • Die Studierenden erhalten Einblick in ein aktuelles Forschungsgebiet. Fachkompetenz - Verstehen • Die Studierenden verstehen Grundlagen des modernen Systementwurfs. • Die Studierenden erklären Implementierungsalternativen für digitale Hardware/Software-Systeme. Fachkompetenz - Anwenden • Die Studierenden wenden grundlegende Algorithmen an, zur Analyse und Optimierung von Hardware/Software-Systemen. Competneces: Professional competence - Knowledge • Students gain insight into a current field of research. Professional competence - Understanding • Students understand the basics of modern system design. • Students explain implementation alternatives for digital hardware/software systems. Professional competence - Application

		• Students apply basic algorithms to analyse and optimise hardware/software systems.
7	**Voraussetzungen für die Teilnahme**	Die Auswahl dieses Moduls schließt die Auswahl des Moduls „Hardware-Software-Co-Design (Vorlesung mit erweiterter Übung)“ aus.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Empfohlene Bücher zur Begleitung und Vertiefung: • Teich J., Haubelt C.: "Digitale Hardware/Software-Systeme: Synthese und Optimierung", Springer-Verlag, 2007, ISBN: 978-3-540-46822-6 • Gajski, D. et al.: "Specification and Design of Embedded Systems", Prentice Hall, Englewood Cliffs, NJ, 1994, ISBN: 978-0131507319 Weitere Informationen: https://www.cs12.tf.fau.de/lehre/lehrveranstaltungen/vorlesungen/hardware-software-co-design

1	**Modulbezeichnung** 292952	**Hardware-Software-Co-Design (Vorlesung mit erweiterter Übung)** (Hardware-software-co-design (Lecture with extended exercises))	**7,5 ECTS**
2	Lehrveranstaltungen	Übung: Übungen zu Hardware-Software-Co-Design (2 SWS) Vorlesung: Hardware-Software-Co-Design (2 SWS) Übung: Erweiterte Übungen zu Hardware-Software-Co-Design (2 SWS)	2,5 ECTS 2,5 ECTS 2,5 ECTS
3	Lehrende	Muhammad Sabih Tobias Hahn Dr.-Ing. Stefan Wildermann Prof. Dr.-Ing. Jürgen Teich	

4	**Modulverantwortliche/r**	Joachim Falk Prof. Dr.-Ing. Jürgen Teich
5	**Inhalt**	Zahlreiche Realisierungen eingebetteter Systeme (z.B. Mobiltelephone, Faxgeräte, Industriesteuerungen) zeichnen sich durch kooperierende Hardware- und Softwarekomponenten aus. Die Popularität solcher Realisierungsformen lässt sich begründen durch 1) die steigende Vielfalt und Komplexität heterogener Systeme, 2) die Notwendigkeit, Entwurfs- und Testkosten zu senken und 3) Fortschritte in Schlüsseltechnologien (Mikroelektronik, formale Entwurfsmethoden). Zum Beispiel bieten Halbleiterhersteller kostengünstige ASICs an, die einen Mikrocontroller und benutzerspezifische Peripherie und Datenpfade auf einem Chip integrieren. Die Synthese solcher Systeme wirft jedoch eine Reihe neuartiger Entwurfsprobleme auf, insbesondere 1) die Frage der Auswahl von Hardware- und Softwarekomponenten, 2) die Partitionierung einer Spezifikation in Hard- und Software, 3) die automatische Synthese von Interface- und Kommunikationsstrukturen und 4) die Verifikation und Cosimulation. 1) Überblick und Vergleich von Architekturen und Komponenten in Hardware/Software-Systemen. 2) Aufbau eines Compilers und Codeoptimierungsverfahren für Hardware und Software 3) Hardware/Software-Partitionierung (Partitionierung komplexer Systeme, Schätzungsverfahren, Performanzanalyse, Codegenerierung) 4) Interfacesynthese (Kommunikationsarten, Synchronisation, Synthese) 5) Verifikation und Cosimulation 6) Tafelübungen 7) Demonstrationen mit rechnergestützten Entwurfswerkzeugen und praktische Übungen Content: Numerous realisations of embedded systems (e.g. mobile phones, fax machines, industrial controls) are characterised by

		cooperating hardware and software components. The popularity of such realisations can be explained by 1) the increasing diversity and complexity of heterogeneous systems, 2) the need to reduce design and testing costs, and 3) advances in key technologies (microelectronics, formal design methods). For example, semiconductor manufacturers offer low-cost ASICs that integrate a microcontroller and user-specific peripherals and data paths on a single chip. However, the synthesis of such systems raises a number of novel design problems, in particular 1) the issue of hardware and software component selection, 2) the partitioning of a specification into hardware and software, 3) the automatic synthesis of interface and communication structures, and 4) verification and cosimulation. 1) Overview and comparison of architectures and components in hardware/software systems. 2) Structure of a compiler and code optimisation procedures for hardware and software. 3) Hardware/software partitioning (partitioning of complex systems, estimation procedures, performance analysis, code generation) 4) Interface synthesis (communication types, synchronisation, synthesis) 5) Verification and cosimulation 6) Blackboard exercises 7) Demonstrations with computer-aided design tools and practical exercises
6	**Lernziele und Kompetenzen**	Fachkompetenz - Wissen • Die Studierenden erhalten Einblick in ein aktuelles Forschungsgebiet. Fachkompetenz - Verstehen • Die Studierenden verstehen Grundlagen des modernen Systementwurfs. • Die Studierenden erklären Implementierungsalternativen für digitale Hardware/Software-Systeme. Fachkompetenz - Anwenden • Die Studierenden wenden grundlegende Algorithmen an, zur Analyse und Optimierung von Hardware/Software-Systemen. • Die Studierenden wenden das erlernte Wissen in den erweiterten Übungen vor Ort an den Rechnerarbeitsplätzen des Lehrstuhls an. Sozialkompetenz • Die Studierenden benutzen aktuelle Entwurfswerkzeuge für die Spezifikation, Optimierung und Prototypisierung von Hardware/

		Software-Systemen bei der kooperativen Bearbeitung der erweiterten Übung in Gruppen. Competences: Professional competence - Knowledge • Students gain insight into a current field of research. Professional competence - Understanding • Students understand the basics of modern system design. • Students explain implementation alternatives for digital hardware/software systems. Professional competence - Application • Students apply basic algorithms to analyse and optimise hardware/software systems. • The students apply the knowledge they have acquired in the extended exercises on site at the computer workstations of the department. Social competence • The students use current design tools for the specification, optimisation and prototyping of hardware/software systems in the cooperative processing of the extended exercise in groups.
7	**Voraussetzungen für die Teilnahme**	Die Auswahl dieses Moduls schließt die Auswahl des Moduls „Hardware-Software-Co-Design“ aus.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20191 Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio Klausur (90 min) und erfolgreicher Bearbeitung aller Übungsaufgaben in den erweiterten Übungen (verpflichtend, vor Ort an den Rechnerarbeitsplätzen des Lehrstuhls). Die Sprache der Klausur ist abhängig von der Wahl der Studierenden entweder Deutsch oder Englisch.
11	**Berechnung der Modulnote**	Portfolio (100%) Die Modulnote ergibt sich aus der Klausurnote.
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 90 h Eigenstudium: 135 h
14	**Dauer des Moduls**	1 Semester

15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Empfohlene Bücher zur Begleitung und Vertiefung: • Teich J., Haubelt C.: "Digitale Hardware/Software-Systeme: Synthese und Optimierung", Springer-Verlag, 2007, ISBN: 978-3-540-46822-6 • Gajski, D. et al.: "Specification and Design of Embedded Systems", Prentice Hall, Englewood Cliffs, NJ, 1994, ISBN: 978-0131507319 Weitere Informationen: https://www.cs12.tf.fau.de/lehre/lehrveranstaltungen/vorlesungen/hardware-software-co-design

1	**Modulbezeichnung** 629391	**HF- und Digital-Architektur von Funksystemen** (RF and digital architecture of radio systems)	**2,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: RF and Digital Architecture of Radio Systems (2 SWS)	2,5 ECTS
3	Lehrende	Dr.-Ing. Heinrich Milosiu	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Georg Fischer
5	**Inhalt**	• Architektur von Funksystemen • Hochfrequenz- und Digital-Teil • Analog-Digital Balance • Vielfachzugriff • Leistungsmerkmale der physikalischen Schicht und ihre Implementierung • Transceiver-Architekturen • AD- und DA-Umsetzung • Beurteilung der Transceiverkomplexität • Grundanforderungen an Basisbandprozessoren • Signalintegrität und Analyse von Verzerrungen (EVM) • Elektronische Kompensationsverfahren • Digitale Signalkonditionierung • Spektrumszuweisung • Frequenzagile Funksysteme • Pegelrechnung zur Funkschnittstelle • Grundzüge der Verkehrstheorie und Netzdimensionierung • Funknetzplanung • Funkabdeckung, Netzkapazität und spektrale Effizienz • Leistungsverbrauch • Leistungshaushalt Content: Architecture of radio communication systems RF and digital section Analogue-digital Balance Multiple Access schemes Physical Layer features and their hardware implementation Transceiver architectures AD and DA conversion Transceiver complexity assessment Baseband processor fundamentals

		Signal integrity and impairment analysis (Error Vector Magnitude) Electronic counter measures Digital signal conditioning Spectrum licensing Frequency agile radio Link budget calculations Basic traffic theory and engineering Network planning Network coverage, capacity and spectral efficiency Power consumption Power budget
6	**Lernziele und Kompetenzen**	Die Studierenden verstehen die Architektur von Mobilfunk-Basisstationen Die Studierenden sind in der Lage Blockdiagramme für den Sende- und Empfangspfad zu erstellen. Die Studierenden kennen die Leistungsmerkmale des GSM Mobilfunksystems auf der Funkschnittstelle. Die Studierenden wissen welche Leistungsmerkmale die Netzabdeckung und welche die Netzkapazität erhöhen. Die Studierenden kennen die Implementierung von Leistungsmerkmalen. Die Studierenden kennen die Vor- und Nachteile verschiedener Sender- und Empfängerarchitekturen. Die Studierenden können einen Pegelplan für die Funknetzplanung erstellen. Die Studierenden können die Störquellen für Signalverzerrungen qualitativ und quantitativ beschreiben. Die Studierenden verstehen die Vor- und Nachteile analoger und digitaler Signalverarbeitung.

		Die Studierenden verstehen die Grundsätze der Verkehrstheorie und -modellierung. Die Studierenden kennen die Metriken zur Beschreibung der ökonomischen Zusammenhänge bei der Funknetzplanung. Die Studierenden können die Signaldynamik im Kontext der Analog-Digital-Umsetzung berechnen. Die Studierenden können den Leistungshaushalt einer Basisstation quantitativ beschreiben. Learning objectives and competencies: Stundents • can analyse the architecture of cellular basestations • can sketch block diagrams for the TX and RX path • can reproduce radio requirements of GSM physical layer • can identify radio features whether they increase coverage or capacity • can describe implementation of radio features • can analyse the pros and cons of different radio architectures • can sketch a level plan for link budget analysis • can describe sources of signal distortion in qualitative and quantitative way • can explain the pros and cons of analogue versus digital signal processing • can recall basics of traffic theory and modelling • can apply metrics for economical assessment with network planning • can evaluate the dynamic range in the context of Analog-Digital-Conversion • can analyse the power budget of a basestation in a quantitative way
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)

12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch oder Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 93020	**Implementierung von Datenbanksystemen** (Implementation of database systems)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Implementierung von Datenbanksystemen (2 SWS) Übung: Übungen zu Implementierung von Datenbanksystemen (2 SWS) Übung: Intensivierungsübung zu Implementierung von Datenbanksystemen (2 SWS)	- - -
3	Lehrende	Demian Vöhringer	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Richard Lenz Prof. Dr.-Ing. Klaus Meyer-Wegener
5	**Inhalt**	Die Vorlesung führt ein in den Aufbau und die Architektur von Datenbanksystemen, die Modularisierung und Schichtenbildung mit Abstraktionen verwenden. Schwerpunkt sind deshalb systemtechnische Aspekte von Datenbanksystemen. Die Übungen vertiefen verschiedene Aspekte an Beispielrechnungen und erweitern gelegentlich auch noch den Stoff um einige Facetten (z.b. Mehrattribut-Zugriffspfade). Ausgangspunkt einer Reihe von aufeinander aufbauenden Abstraktionen ist die Speicherung von Daten auf Hintergrundspeichern. Die erste Abstraktion ist die Datei. Dann werden Sätze eingeführt und auf verschiedene Weisen in Blöcken organisiert (sequenziell, mit Direktzugriff, indexsequentiell). Das schließt die Organisation eines Blockpuffers und Zugriffspfade (Indexstrukturen) unterschiedlichen Typs ein. Als zweite große Abstraktion werden Datenmodelle eingeführt und hier insbesondere das relationale. Das ist bereits aus dem Modul "Konzeptionelle Modellierung" bekannt, wird hier aber aus einer ganz anderen Perspektive heraus entwickelt. Der zweite Teil befasst sich mit der Realisierung der Leistungen eines Datenbanksystems unter Verwendung der vorher eingeführten Sätze und Zugriffspfade ("top-down"). Das umfasst die Anfrageverarbeitung und -optimierung, aber auch die Mechanismen zur Protokollierung von Aktionen und zur Wiederherstellung von Datenbankzuständen nach einem Fehler oder Ausfall. Ein laufend vervollständigtes Schichtenmodell fasst abschließend die Aufgaben in einer Architektur für Datenbank-Verwaltungssysteme zusammen. Ziel des Moduls ist es also, ein grundlegendes Verständnis für den Aufbau und die Funktionsweise eines Datenbanksystems zu vermitteln.
6	**Lernziele und Kompetenzen**	Die Studierenden • kennen das Schichtenmodell eines Datenbankverwaltungssystems; • verstehen das Prinzip der Datenunabhängigkeit (Datenabstraktion); • beherrschen das Aufbauprinzip einer Software-Schicht; • unterscheiden die Begriffe "Datenbank", "Datenbanksystem" und "Datenbankverwaltungssystem";

		• unterscheiden die Begriffe "Datenmodell" und "Schema"; • zeigen das Konzept der blockorientierten Datei mit ihren Zugriffsoperationen auf; • unterscheiden einen Satz von einem Block; • erklären das Konzept der sequentiellen Satzdatei; • schildern das Prinzip der Wechselpuffertechnik; • charakterisieren den Schlüsselzugriff auf Sätze; • stellen Gestreute Speicherung (Hashing) auf der Basis von Blöcken (Buckets) dar; • formulieren die Funktionsweise des Virtuellen Hashings; • fassen die Funktionsweise eines B-Baums zusammen; • unterscheiden die Dienste eines B-Baums von denen des Hashings; • können für eine Folge von Schlüsselwerten einen B-Baum aufbauen; • unterscheiden einen B-Baum von einem B-Stern-Baum (B+-Baum); • veranschaulichen einen Bitmap-Index; • unterscheiden die Primär- und Sekundärorganisation von Sätzen; • zählen Ersetzungsstrategien der Pufferverwaltung auf und vergleichen sie; • benennen die Dienste einer Pufferverwaltung; • erklären die Konzepte "Seite" und "Segment" im Gegensatz zu "Block" und "Datei"; • unterscheiden direkte und indirekte Seitenzuordnung; • interpretieren in Programmiersprachen eingebettete Anfragesprachen und Datenbank-Unterprogrammaufrufe; • charakterisieren Datenbank-Transaktionen; • kennen die Aufrufe zur Definition von Transaktionen; • erläutern die spaltenweise Abspeicherung von Relationen; • diskutieren die algebraische Optimierung von Anfragen; • stellen Planoperatoren eines Datenbanksystems dar; • unterscheiden Planoperatoren für den Verbund; • beschreiben Kostenformeln für die Abschätzung von Anfrageausführungen; • schildern die verschiedenen Anomalien im Mehrbenutzerbetrieb; • beschreiben die Serialisierbarkeit von Transaktionen; • erläutern das Konzept der Sperren in Datenbanksystemen; • unterscheiden physische und logische Konsistenz; • kennen die vier Recovery-Klassen; • erläutern die verschiedenen Arten von Sicherungspunkten.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Masterprüfung Master of Science Information and Communication Technology 20191

		Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	KEMPER, Alfons ; EICKLER, André: Datenbanksysteme : Eine Einführung. 9., aktual. u. erweit. Aufl. München : Oldenbourg, 2013. ISBN 978-3-486-72139-3. Kapitel 7 bis 11 KEMPER, Alfons ; WIMMER, Martin: Übungsbuch Datenbanksysteme. 2., aktual. u. erweit. Aufl. München : Oldenbourg, 2009. ISBN 978-3-486-59001-2. Kapitel 7 bis 11 HEUER, Andreas ; SAAKE, Gunter: Datenbanken : Konzepte und Sprachen. 3., aktual. u. erw. Aufl. Bonn : mitp, 2007. - ISBN 3-8266-1664-2 HÄRDER, Theo ; RAHM, Erhard: Datenbanksysteme : Konzepte und Techniken der Implementierung. Berlin : Springer, 1999 - ISBN 3-540-65040-7 SAAKE, Gunter ; HEUER, Andreas: Datenbanken : Implementierungstechniken. 2., aktual. u. erw. Aufl. Bonn : mitp, 2005. ISBN 3-8266-1438-0

1	**Modulbezeichnung** 93601	**Information Theory and Coding / Informationstheorie und Codierung** (Information theory and coding)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Tutorial for Information Theory and Coding (1 SWS) Vorlesung: Information Theory and Coding (3 SWS)	- 5 ECTS
3	Lehrende	Dr.-Ing. Ali Bereyhi Prof. Dr.-Ing. Ralf Müller	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Ralf Müller
5	**Inhalt**	1. Introduction: binomial distribution, (7,4)-Hamming code, parity-check matrix, generator matrix 2. Probability, entropy, and inference: entropy, conditional probability, Bayes law, likelihood, Jensens inequality 3. Inference: inverse probability, statistical inference 4. The source coding theorem: information content, typical sequences, Chebychev inequality, law of large numbers 5. Symbol codes: unique decidability, expected codeword length, prefix-free codes, Kraft inequality, Huffman coding 6. Stream codes: arithmetic coding, Lempel-Ziv coding, Burrows-Wheeler transform 7. Dependent random variables: mutual information, data processing lemma 8. Communication over a noisy channel: discrete memory-less channel, channel coding theorem, channel capacity 9. The noisy-channel coding theorem: jointly-typical sequences, proof of the channel coding theorem, proof of converse, symmetric channels 10. Error-correcting codes and real channels: AWGN channel, multivariate Gaussian pdf, capacity of AWGN channel 11. Binary codes: minimum distance, perfect codes, why perfect codes are bad, why distance isnt everything 12. Message passing: distributed counting, path counting, low-cost path, min-sum (=Viterbi) algorithm 13. Exact marginalization in graphs: factor graphs, sum-product algorithm

14. Low-density parity-check codes: density evolution, check node degree, regular vs. irregular codes, girth

15. Lossy source coding: transform coding and JPEG compression

--

1. Einleitung: Binomialverteilung, (7,4)-Hamming-Code, Paritätsmatrix, Generatormatrix

2. Wahrscheinlichkeit, Entropie und Inferenz: Entropie, bedingte Wahrscheinlichkeit, Bayessches Gesetz, Likelihood, Jensensche Ungleichung

3. Inferenz: Inverse Wahrscheinlichkeit, statistische Inferenz

4. Das Quellencodierungstheorem: Informationsgehalt, typische Folgen, Tschebyschevsche Ungleichung, Gesetz der großen Zahlen

5. Symbolcodes: eindeutige Dekodierbarkeit, mittlere Codewortlänge, präfixfreie Codes, Kraftsche Ungleichung, Huffmancodierung

6. Stromcodes: arithmetische Codierung, Lempel-Ziv-Codierung, Burrows-Wheeler-Transformation

7. Abhängige Zufallsvariablen: Transinformation, Datenverarbeitungslemma

8. Kommunikation over gestörte Kanäle: diskreter gedächtnisloser Kanal, Kanalcodierungstheorem, Kanalkapazität

9. Das Kanalcodierungstheorem: verbundtypische Folgen, Beweis des Kanalcodierungstheorems, Beweis des Umkehrsatzes, symmetrische Kanaäle

10. Fehlerkorrigierende Codes und reale Kanäle: AWGN-Kanal, mehrdimensionale Gaußsche WDF, Kapazität des AWGN-Kanals

11. Binäre Codes: Minimaldistanz, perfekte Codes, Warum perfekte Codes schlecht sind, Warum Distanz nicht alles ist

12. Nachrichtenaustausch: verteiltes Zählen, Pfadzählen, günstigster Pfad, Minimumsummenalgorithmus

13. Exakte Marginalisierung in Graphen: Faktorgraph, Summenproduktalgorithmus

14. LDPC-Codes: Dichteevolution, Knotenordnung, reguläre und irreguläre Codes, Graphumfang

		15. Verlustbehaftete Quellencodierung: Transformationscodierung und JPEG-Kompression
6	**Lernziele und Kompetenzen**	The students apply Bayesian inference to problems in both communications and everyday's life. The students explain the concept of digital communications by means of source compression and forward-error correction coding. For the design of communication systems, they use the concepts of entropy and channel capacity. They calculate these quantities for memoryless sources and channels. The students proof both the source coding and the channel coding theorem. The students compare various methods of source coding with respect to compression rate and complexity. The students apply source compression methods to measure mutual information. The students factorize multivariate functions, represent them by graphs, and marginalize them with respect to various variables. The students explain the design of error-correcting codes and the role of minimum distance. They decode error-correcting codes by means of maximum-likelihood decoding and message passing. The students apply distributed algorithms to problems in both communications and everydays life. The students improve the properties of low-density parity-check codes by widening the girth and/or irregularity in the degree distribution. The students transform source images into the frequency domain to improve lossy compression. -- Die Studierenden wenden Bayessche Inferenz auf Probleme in der Nachrichtentechnik und im Alltagsleben an. Die Studierenden erklären die konzeptuelle Trennung von digitaler Übertragung in Quellen- und Kanalcodierung.

		Kommunikationssysteme entwerfen sie unter Betrachtung von Entropie und Kanalkapazität. Sie berechnen diese Größen für gedächtnislose Quellen und Kanäle. Die Studierenden beweisen sowohl das Quellen- als auch das Kanalcodierungstheorem. Die Studierenden vergleichen verschiedenartige Quellencodierungsverfahren hinsichtlich Komplexität und Kompressionsrate. Die Studierenden verwenden Quellencodierverfahren zur Messung von Transinformation. Die Studierenden faktorisieren Funktionen mehrerer Veränderlicher, stellen diese als Graph dar und marginalisieren sie bezüglich mehrerer Veränderlicher. Die Studierenden erklären den Entwurf von Kanalcodes und den Einfluss der Minimaldistanz. Sie decodieren Kanalcodes gemäß maximaler Likelihood und Nachrichtenaustausch. Die Studierenden wenden verteilte Algorithmen auf Probleme der Nachrichtentechnik und des Alltagslebens an. Die Studierenden verbessern die Eigenschaften von LDPC-Codes durch Erhöhung des Umfangs und/oder durch irreguäre Knotenordnungsverteilungen. Die Studierenden transformieren Bildquellen zur Verbesserung verlustbehafteter Kompression iin den Frequenzbereich.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 2
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h

14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	MacKay, D.: Information Theory, Inference, and Learning Algorithms, Cambridge University Press, Cambridge, 2003.

1	**Modulbezeichnung** 96101	**Integrierte Navigationssysteme** (keine englischsprachige Modulbezeichnung hinterlegt!)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übung Integrierte Navigationssysteme (1 SWS) Vorlesung: Integrierte Navigationssysteme (3 SWS)	- 5 ECTS
3	Lehrende	Prof. Dr. Jörn Thielecke	

4	**Modulverantwortliche/r**	Prof. Dr. Jörn Thielecke
5	**Inhalt**	**1. Überblick** • Von der Astronavigation zur Navigation mit Mikroelektronik • Messprinzipien & Positionsberechnung (Standlinien/-flächen) • Begriffsdefinitionen (s. US Federal Radionavigation Plan), Genauigkeit, Verfügbarkeit, Verlässlichkeit, Integrität, etc. • Systematische Strukturierung des Gebiets: siehe 2. bis 7. **2. Positions- und Lagebestimmung** • Funkausbreitung und Funkortung (Beispiel WLAN) • Fingerabdruckverfahren • Lokalisierung mit Markovketten **3. Koppelnavigation (Tracking) mittels Trägheitsnavigation** • Koordinatensysteme und ihre Einsatzgebiete • Mathematische Gundlagen, z.B. Quaternionen, Corioliseffekt • Strapdown Inertial Navigation Systems • Sensorprinzipien und Trägheitssensoren • Computergestützte Lösung der Navigationsgleichungen • System- und Fehlermodellierung im Zustandsraum • Das Kalmanfilter und Glättung mittels Retrodiktion **4. Seiteninformationen: Kinematik und Karten (kurze Übersicht)** **5. Landmarken als lokaler Ortsbezug** • Merkmalsbasierte Ortung z.B. mit Kamera oder UWB • Partikelfilter und Monte-Carlo-Integration **6. Integration von Navigationskomponenten: Sensordatenfusion** • Fusionsarchitekturen: Beispiel GPS & Trägheitsnavigation **7. Einbettung von Navigationssystemen** • Assisted GPS oder Location Based Service Anmerkung: Die Navigationsmethoden werden gleichermaßen anhand von Tafel- und Rechnerübungen (MATLAB) einstudiert
6	**Lernziele und Kompetenzen**	1. Die Studierenden werden in die Lage versetzt, typische Navigationsverfahren hinsichtlich ihrer Funktionsweise und Einsetzbarkeit zu analysieren, zu bewerten und weiterzuentwickeln.

		2. Die Studierenden lernen Navigationsgleichungen selbst aufzustellen, anzuwenden und mit unterschiedlichen Algorithmen auf dem Computer zu lösen. 3. Die Studierenden entwickeln ein Verständnis für die Herausforderungen bei der Integration unterschiedlicher Teilsysteme zu einem Navigationssystem und der Einbettung von Navigationssystemen in übergeordnete Systeme
7	**Voraussetzungen für die Teilnahme**	Keine formalen Voraussetzungen, geeignet für Masterstudium, grundlegende Kenntnisse erforderlich in: linearer Algebra, Physik, Signal- & System¬theorie, Wahrscheinlichkeitstheorie.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%) Bei bestandener Prüfung wird die Note um eine Teilnotenstufe (z.B. von 2,0 auf 1,7) verbessert, wenn Sie mindestens 75% der Hausaufgaben einschließlich der Rechnerübungen erfolgreich absolviert haben. Eine Note besser als 1,0 wird nicht vergeben.
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Skriptum zur Lehrveranstaltung.

1	**Modulbezeichnung** 96260	**Integrierte Schaltungen für Funkanwendungen** (Integrierte Schaltungen für Funkanwendungen)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Integrierte Schaltungen für Funkanwendungen (2 SWS) Übung: Übungen zu Integrierte Schaltungen für Funkanwendungen (2 SWS)	5 ECTS -
3	Lehrende	Dr.-Ing. Heinrich Milosiu Albert-Marcel Schrotz	

4	**Modulverantwortliche/r**	Dr.-Ing. Heinrich Milosiu
5	**Inhalt**	• Transceiver-Architekturen • Hochfrequenzaspekte • Tranistoren und Technologien • Passive Bauelemente und Netzwerke • Rauscharme Vorverstärker • Mischer • Oszillatoren • Phasenregelschleifen und Synthesizer • Messtechnische Grundlagen
6	**Lernziele und Kompetenzen**	Nach der Teilnahme an den Modulveranstaltungen sind die Studierenden in der Lage: • Den Aufbau sowie Vor-und Nachteile von Transceiver-Architekturen zu verstehen • Hochfrequenzaspekte von Transistoren und Schaltungen zu analysieren • Geeignete Integrationstechnologien auszuwählen • Passive Bauelemente und Netzwerke zu verstehen und anzuwenden • Schaltungstopologien rauscharmer Vorverstärker, Mischer,Oszillatoren anzuwenden und zu analysieren
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester

15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 43391	**Interaktive Computergraphik** (Interactive computer graphics)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Tutorials to Interactive Computer Graphics (2 SWS) Vorlesung: Interactive Computer Graphics (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Jonas Müller Laura Fink Prof. Dr. Marc Stamminger	

4	**Modulverantwortliche/r**	Prof. Dr. Marc Stamminger
5	**Inhalt**	In dem Modul werden GPUs und dafür maßgeschneiderte Algorithmen behandelt: • Architektur von GPUs und Echtzeit-Rendering-Pipeline • Deferred Shading und Anti-Aliasing-Verfahren • Simulation von Umgebungsbeleuchtung • Verfahren zur Generierung von Schatten • Level-of-Detail-Verfahren zur Darstellung komplexer Szenen • Animation von Objekten
6	**Lernziele und Kompetenzen**	Fachkompetenz Verstehen Lernende können verschiedene in der Vorlesung behandelte Verfahren der interaktiven Computergraphik mit eigenen Worten erklären. Anwenden Lernende können Zusammenhänge zwischen den in der Vorlesung behandelten Verfahren der interaktiven Computergraphik erkennen, und Ideen auf neue Anwendungen übertragen. Analysieren Lernende können Unterschiede und Ähnlichkeiten der in der Vorlesung behandelten Verfahren erkennen und daraus neue Lösungen entwickeln.
7	**Voraussetzungen für die Teilnahme**	Die Übungen setzen Kenntnisse in C/C++ voraus.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Masterprüfung Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Übungsleistung Variabel

11	**Berechnung der Modulnote**	Übungsleistung (0%) Variabel (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• Möller, Haines: "Real-Time Rendering"

1	**Modulbezeichnung** 44140	**Interventional Medical Image Processing** (Interventional medical image processing)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Medical Image Processing for Interventional Applications (VHB-Kurs) (4 SWS)	5 ECTS
3	Lehrende	Manuela Meier Arpitha Ravi Celia Martín Vicario Luis Rivera Monroy	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Andreas Maier
5	**Inhalt**	English Version: This module focuses on recent developments in image processing driven by medical applications. All algorithms are motivated by practical problems. The mathematical tools required to solve the considered image processing tasks will be introduced. The module starts with an overview on preprocessing algorithms such as scatter correction for x-ray images, edge detection, super-resolution and edge-preserving noise reduction. The second chapter describes automatic image analysis using feature descriptors, key point detection, and segmentation using bottom-up algorithms such as the random walker or top-down approaches such as active shape models. Furthermore, the module covers geometric calibration algorithms for single view calibration, epipolar geometry, and factorization. The last part of the module covers non-rigid registration based on variational methods and motion-compensated image reconstruction. Deutsche Version: Das Modul ist auf die jüngsten Entwicklungen in der Verarbeitung von medizinischen Bildern ausgerichtet. Alle Algorithmen werden durch praktische Probleme motiviert. Die mathematischen Werkzeuge, die für die Bildverarbeitungsaufgaben benötigt werden, werden eingeführt. Das Modul beginnt mit einem Überblick über Vorverarbeitungsalgorithmen, wie zum Beispiel Streustrahlkorrektur für

		Röntgenbilder, Kantenerkennung, Superresolution und kantenerhaltende Rauschunterdrückung. Das zweite Kapitel beschreibt die automatische Bildanalyse mit Merkmalsdeskriptoren, Punkterkennung und Segmentierung mit Bottom-up-Algorithmen wie dem Random-Walker oder Top-Down-Ansätzen wie aktiven Formmodellen. Darüber hinaus deckt die Vorlesung auch geometrische Kalibrierungsalgorithmen zur Einzelansicht-Kalibrierung, Epipolargeometrie und Faktorisierung ab. Der letzte Teil des Moduls deckt nicht-starre Registrierung auf der Grundlage von Variationsmethoden und bewegungskompensierter Bildrekonstruktion ab.
6	**Lernziele und Kompetenzen**	English Version: The participants • summarize the contents of the lecture. • apply pre-processing algorithms such as scatter correction and edge-preserving filtering. • extract information from images automatically by image analysis methods such as key point detectors and segmentation algorithms. • calibrate projection geometries for single images and image sequences using the described methods. • develop non-rigid registration methods using variational calculus and different regularizers. • adopt algorithms to new domains by appropriate modifications. Deutsche Version: Die Teilnehmenden • fassen die Inhalte der Vorlesung zusammen. • wenden Vorverarbeitungsalgorithmen wie Streustrahlkorrektur und kantenerhaltende Filterung an. • extrahieren automatisch Informationen aus Bildern, indem sie Bildanalyseverfahren wie Punktdetektoren und Segmentierungsalgorithmen verwenden. • kalibrieren Projektionsgeometrien für Einzelbilder und Bildsequenzen mit den beschriebenen Methoden. • entwickeln nicht-starre Registrierungsmethoden mit Hilfe von Variationsrechnung und unterschiedlichen Regularisierern. • wenden Algorithmen auf neue Modalitäten durch entsprechende Änderungen im Algorithmus an.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1

9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich/mündlich (60 Minuten)
11	**Berechnung der Modulnote**	schriftlich/mündlich (100%)
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 0 h Eigenstudium: 150 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 96270	**Kanalcodierung** (Channel coding)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung mit Übung: Kanalcodierung (3 SWS) Vorlesung mit Übung: Channel Coding (4 SWS) Übung: Übungen zur Kanalcodierung (1 SWS)	5 ECTS 5 ECTS -
3	Lehrende	Dr.-Ing. Clemens Stierstorfer	

4	**Modulverantwortliche/r**	Dr.-Ing. Clemens Stierstorfer
5	**Inhalt**	1) Introduction and Motivation 2) Fundamentals of Block Coding 3) Introduction to Finite Fields I 4) Linear Block Codes 5) Linear Cyclic Codes 6) Introduction to Finite Fields II 7) BCH and RS Codes 8) Convolutional Codes 9) Codes with Iterative Decoding
6	**Lernziele und Kompetenzen**	Das Modul Kanalcodierung umfasst eine Einführung in die Grundlagen der algebraischen, fehlerkorrigierenden Blockcodes sowie einen Einstieg in die Thematik der Faltungscodes. Iterativ decodierte Codeschemata wie Turbo-Codes und LDPC-Codes werden ebenfalls eingeführt. Im Einzelnen sind die Inhalte oben aufgeführt. Die Studierenden definieren die Problematik der Kanalcodierung, grenzen sie von anderen Codierverfahren (z.B. der Quellencodierung) ab und kennzeichnen die unterschiedlichen Ansätze zur Fehlerkorrektur und -erkennung. Sie nennen Beispiele für Einsatzgebiete von Kanalcodierung und geben einen Überblick über die historische Entwicklung des Fachgebiets. Die Studierenden erstellen Übertragungsszenarien für den Einsatz von Kanalcodierung bestehend aus Sender, Übertragungskanal und Empfänger und beachten dabei die Grundannahmen beim Einsatz von Blockcodes bzw. der Modellierung der Kanäle. Sie formulieren mathematische Beschreibungen der Encodierung sowie der optimalen Decodierung bzw. suboptimaler Varianten. Die Studierenden beherrschen die Grundlagen fehlerkorrigierender linearer Blockcodes, beschreiben diese mathematisch korrekt mittels Vektoren und Matrizen über endlichen Körpern und implementieren und bewerten zugehörige Encoder- und Decoderstrukturen insbesondere Syndromdecoder. Dabei modifizieren sie Generatormatrizen, ermitteln Prüfmatrizen und erstellen Syndromtabellen. Sie schätzen die minimale Hammingdistanz von Codes mittels Schranken ab und können den erzielbaren Codegewinn erläutern. Sie kennen und benutzen beispielhaften Codefamilien (z.B. Hamming-Codes, Simplex-Codes, Reed-Muller-Codes).

Die Studierenden erkennen die Vorteile zyklischer linearer Blockcodes und beschreiben diese mit Polynomen über endlichen Körpern. Sie nutzen die Restklassenrechnung bzgl. Polynomen zur Umsetzung systematischer Encoder und zur Realisierung von Syndromdecodern mittels Schieberegisterschaltungen. Sie kennen beispielhafte Codefamilien.

Die Studierenden nutzen Primkörper, Erweiterungskörper, Minimalpolynome und Kreisteilungsklassen sowie die Spektraldarstellung über endlichen Körpern zur Realisierung von BCH- und Reed-Solomon-Codes gemäß der BCH-Schranke. Sie verstehen die Grundlagen der Decodierung von BCH- und Reed-Solomon-Codes. Sie skizzieren und erläutern die Kanalcodierkonzepte von CD und DVD.

Die Studierenden erklären die Unterschiede von Faltungscodes und Blockcodes, skizzieren anhand von tabellierten Generatorpolynomen zugehörige Encoder und erläutern diese. Sie erklären die Funktionsweise des optimalen Decoders (MLSE) und demonstrieren diese beispielhaft.

Die Studierenden verstehen die Grundlagen der iterativen Decodierung, insbesondere wenden sie die Grundlagen des Information Combining zur Kombination von verschiedenen Beobachtungen an. Sie verstehen die Bedeutung von Log-Likelihood-Ratios bei iterativen Decodieruvorgängen und berechnen diese. Sie skizzieren die grundlegenden Encoder- und Decoderstrukturen von Turbo-Codes und die Grundzüge der Codierung mit LDPC-Codes u.a. der Decodierung mittels Belief Propagation.

Die Vorlesung erfolgt wechselweise auf Deutsch oder Englisch (Winter/ Sommer). Die zur Verfügung gestellten Unterlagen sind ausschließlich in Englisch gehalten. Die Studierenden verwenden entweder die englischen Fachtermini sicher oder kennen diese und drücken sich sicher mit den entsprechenden deutschen Fachbegriffen aus.

Die Umsetzung der angegebenen Algorithmen in eine Programmiersprache (C, Matlab usw.) sollten die Studierenden zu diesem Zeitpunkt des Studiums üblicherweise beherrschen. Übungen hierzu bleiben der Eigeninitiative überlassen.

Students define the problems of channel coding, how to distinguish it from other coding methods (such as source coding) and how to describe the various different approaches to error correction and detection. They are able to list example application areas of channel coding and give an overview of the historical development of the field.

		Furthermore, they describe and analyze transmission scenarios for the application of channel coding which consist of transmitter, transmission channel and receiver, taking into account the general assumptions for applying block codes or modeling the channels. They formulate mathematical descriptions of encoding, optimal decoding and sub-optimal methods. Students illustrate the principles of error-correcting linear block codes and describe them mathematically using vectors and matrices over finite fields. They implement and analyze corresponding encoder and decoder structures, in particular syndrome decoders, and modify generator matrices, construct test matrices and create syndrome tables. They estimate the minimum Hamming distance of codes using bounds and are able to explain the coding gain that can be achieved in individual cases. They analyze and use example code families (e.g. Hamming codes, simplex codes, Reed-Muller codes). Students explain the advantages of cyclic linear block codes and how to describe them with polynomials over finite fields. They apply polynomial modular arithmetic to implement systematic encoders and realize syndrome decoders using shift register circuits. They know and use exemplary code families. Students use prime fields, extension fields, minimal polynomials and cyclotomic cosets, and spectral representation over finite fields to implement BCH and Reed-Solomon codes using the BCH bound. They understand the foundations of decoding BCH and Reed-Solomon codes and how to sketch and explain the channel coding concepts of CDs and DVDs. Students are able to describe the differences between convolutional codes and block codes, to sketch the respective encoders based on tabulated generator polynomials and to explain them. They are able to explain how optimal decoders (MLSE) work using examples. Students sketch the foundations of iterative decoding. In particular, they apply methods of information combining to combine different observations. They use and calculate log-likelihood ratios in iterative decoding processes, sketch the basic encoding and decoding structures of turbo codes and the basics of coding using LDPC codes (including decoding using belief propagation). Students either are able to use the English technical terms correctly or know them and are able to express themselves using the respective technical terms in German.
7	**Voraussetzungen für die Teilnahme**	Es ist hilfreich, wenn die Studierenden die erlernten Algorithmen in eine Programmiersprache (C, Matlab usw.) umsetzen können.

		It would be very helpful if the participants can implement the specified algorithms into a programming language (C, Matlab, etc.).
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich Hilfsblatt, Taschenrechner: Sie können ein einzelnes A4-Blatt (Vorder- und Rückseite oder andere Blätter mit offensichtlich identischer Gesamtfläche) verwenden, um Ihre eigene, handschriftliche Formelsammlung aufzuschreiben. Sie können einen nicht programmierbaren Taschenrechner verwenden. Cheat Sheet, Calculator: A single A4 sheet (front and back, or any other collection of sheets with an obviously identical total area size) can be used to write down your own handwritten collection of formulas, etc. You may also bring a non-programmable calculator.
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch oder Englisch
16	**Literaturhinweise**	• J. Huber, R. Fischer, C. Stierstorfer: Folien zur Vorlesung • M. Bossert: Kanalcodierung, Oldenbourg Wissenschaftsverlag, 3. Auflage, 2013 • M. Bossert: Channel Coding for Telecommunications, John Wiley & Sons, 1999 • B. Friedrichs: Kanalcodierung, Springer Verlag, 1996 • S.B. Wicker: Error Control Systems for Digital Communications and Storage, Prentice-Hall, 1995

1	**Modulbezeichnung** 92730	**Kommunikationselektronik** (Communications electronics 1)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Kommunikationselektronik (2 SWS)	5 ECTS
3	Lehrende	Nives Berner Dr.-Ing. Jörg Robert Clemens Neumüller	

4	**Modulverantwortliche/r**	Dr.-Ing. Jörg Robert
5	**Inhalt**	1.Einleitung 2.Darstellung von Signalen und Spektren • Kontinuierliche und diskrete Signale • Spektrum eines Signals • Unterabtastung und Überabtastung 3.Aufbau und Signale eines Software Defined Radio Systems • Blockschaltbild eines Software Defined Radio Systems • Basisband- und Trägersignale • Empfänger-Topologien • Signale in einem Software Defined Radio System 4.Drahtlose Netzwerke 5.Übertragungsstrecke • Funkstrecke • Antennen 6.Leistungsdaten eines Empfängers • Rauschen • Nichtlinearität • Dynamikbereich eines Empfängers 7.Digital Downconverter • CIC-Filter • Polyphasen-FIR-Filter • Halbband-Filterkaskade • Interpolation 8.Demodulation digital modulierter Signale • Einführung • Demodulation einer GFSK/PAM-Paketsendung Das Modul Kommunikationselektronik behandelt Aspekte der Schaltungstechnik und der Signalverarbeitung drahtloser Übertragungssysteme, die als sog. "Software Defined Radio" Systeme

aufgebaut sind. Als Beispiel dient der Empfänger eines einfachen Telemetrie-Systems, der von der Antenne bis zum Nutzdatenausgang behandelt wird. Schwerpunkte bilden der Aufbau und die Eigenschaften der Hardware des Empfängers sowie die Algorithmen zum Empfang von Telemetrie-Signalen. Dabei wird ein typisches System mit Hilfe eines miniaturisierten Empfängers und einer Verarbeitung mit dem MATLAB-kompatiblen Mathematikprogramm Octave implementiert. Die benötigte Software wird den Studierenden zur Verfügung gestellt.

Content:

1. Introduction

2. Signal representation and discrete signals

   a. Continuous and discrete signals

   b. Signal spectrum

   c. Downsampling and upsampling

3. Structure and signals of a Software Defined Radio

   a. Block diagram of a Software Defined Radio

   b. Base band signals and carrier signals

   c. Receiver topologies

   d. Signals in a Software Defined Radio

4. Wireless networks

5. Transmission path

   a. Radio link

   b. Antennas

6. Performance data of a receiver

   a. Noise

   b. Nonlinearities

   c. Dynamic range of a receiver

		7. Digital Down Converter a. CIC filter b. Polyphase FIR filter c. Halfband filter cascade d. Interpolation 8. Demodulation of digital modulated signals a. Introduction b. Demodulation of a GFSK/PAM packet transmission The module Communication Electronics deals with aspects of circuitry and signal processing of wireless communication systems, built up as so-called "Software Defined Radio systems. A receiver of a simple telemetry system serves as an example, being examined starting from its antenna to the user data output. The focus lies on the structure and the characteristic of the receivers hardware as well as the algorithms for the reception of telemetry signals. A typical system is implemented using a miniaturized receiver and processing with the MATLAB-compatible Octave math program. The required software is provided to the students.
6	**Lernziele und Kompetenzen**	1. Die Studierenden verstehen die grundlegende Funktionsweise eines Software Defined Radio (SDR) Systems, d.h. sie verstehen die Funktionsweise der einzelnen Signalverarbeitungsschritte sowie die auftretenden Signale selbst. 2. Die Studierenden analysieren die Leistungsfähigkeit der analogen Komponenten eines SDR Systems und können Verfahren zur Optimierung dieser Komponenten selbständig anwenden. 3. Die Studierenden analysieren die digitalen Verarbeitungsschritte ausgewählter Modulationsarten und können damit selbst die digitale Signalverarbeitung eines SDR Senders und Empfängers erschaffen. 1. The students will understand the basic operation of a Software Defined Radio (SDR) system, i.e. the students will understand how the individual signal processing steps work as well as the signals themselves.

		2. The students analyze the performance of the analog components of an SDR system and are able to apply procedures for optimizing these components independently. 3.The students analyse the digital processing steps of selected modulation types and are able to create the digital signal processing of an SDR transmitter and receiver themselves.
7	**Voraussetzungen für die Teilnahme**	Keine formalen Voraussetzungen, grundlegende Kenntnisse im Bereich digitaler Signalverarbeitung werden vorausgesetzt
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20191 Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Skriptum zur Veranstaltung im StudON verfügbar: https://www.studon.fau.de/studon/goto.php?target=crs_117973

1	**Modulbezeichnung** 96801	**Kommunikationsstrukturen** (Communication structures)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Kommunikationsstrukturen (2 SWS) Übung: Übungen zu Kommunikationsstrukturen (2 SWS)	5 ECTS 5 ECTS
3	Lehrende	Jürgen Frickel	

4	**Modulverantwortliche/r**	Jürgen Frickel
5	**Inhalt**	Einführung • Information und Kommunikation • Anwendungsgebiete - Kommunikation **Strukturen und Eigenschaften von Kommunikationssystemen** • Grundlegende Definitionen und Klassifikationen • Grundlegende Strukturen **Protokolle und Schnittstellen** • Grundlagen • Basis-Verfahren und Beispiele • TCP/IP-Protokol • Referenzmodell nach ISO/OSI • Sicherungsschicht/Data Link Layer (LLC und MAC) • Bitübertragungsschicht/Physical Layer • Übertragungsmedien **Hardware in Kommunikationsstrukturen** • HW-Architekturen und Funktionsblöcke • Digitale und Analoge Komponenten • Schaltungsdetails von Komponenten **Grundlagen von Bussystemen** • Klassifikation • Funktionale Eigenschaften • Arbitrierungs-Verfahren **Leitungsgebundene Anwendungen für Rechnersysteme** • Bus-Applikationen • Baustein-/IC-interne Busse (AMBA, FPI, ConTraBus, .) • Baugruppeninterne Busse (I2C, Chipsätze+Bridges, .) • Busse für Rechensysteme (VME, ISA, PCI, PCIe, AGP, .) • Peripherie-Busse (ATA, IEC, USB, Firewire, Fibre Channel, Thunderbolt .) **Leitungsgebundene Anwendungen in Systemen**

		• Feldkommunikation • Automobil, Luftfahrt, Space (CAN, MOST, LIN, MILBus, Spacewire .) • Industrie, Haustechnik (Profibus, EIB, .) • Weitverkehrsnetze • SDH, PDH, ATM,
6	**Lernziele und Kompetenzen**	1. Die Studierenden werden in die Lage versetzt die Konzepte und Verfahren vor allem drahtgebundener Kommunikationssysteme anzuwenden. 2. Die Studierenden lernen die Funktionsweise und den Einsatzzweck diverser Kommunikationsprotokolle zu verstehen, und miteinander zu vergleichen. 3. Desweiteren analysieren und klassifizieren Sie grundlegende Strukturen von leitungsgebundenen Kommunikationssystemen anhand ihrer funktionalen Eigenschaften.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 43950	**Kommunikationssysteme** (Communication systems)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Kommunikationssysteme (2 SWS) Übung: Übungen zu Kommunikationssysteme (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr. Reinhard German Alexander Brummer	

4	**Modulverantwortliche/r**	Prof. Dr. Reinhard German
5	**Inhalt**	Aus Rechnerkommunikation ist der grundlegende Aufbau von IP-basierten Netzen bekannt, Inhalt von Kommunikationssysteme sind weitere Netztechnologien wie Leitungsvermittlung (Telefonnetze, Sonet/ SDH/WDM) und Netze mit virtueller Leitungsvermittlung (ATM, MPLS) sowie Netzwerkvirtualisierung (SDN, NFV), Multimediakommunikation über paketvermittelte Netze (Streaming, RTP, SIP, Multicast), Dienstgüte in paketvermittelten Netzen (Integrated Services, RSVP, Differentiated Services, Active Queue Management, Policing, Scheduling), drahtlose und mobile Kommunikation (GSM, UMTS, LTE, 5G, Wimax, WLAN, Bluetooth, ZigBee u.a. Sensornetze). Auch Kommunikation in der Industrie wird behandelt. In der Übung werden praktische Aufgaben im Labor durchgeführt: ein Labor enthält mehrere IP-Router, Switches und Rechner, IP-Telefone und Telefonie-Software für VoIP, es werden verschiedene Konfigurationen eingestellt und getestet. Ein weiterer Übungsteil beschäftigt sich mit Mobilkommunikation. *Contents:* Based on the course computer communications the architecture of IP networks is known. Contents of this course will be additional networking technologies such as circuit switching (telephony, SONET/ SDH/WDM) and networks with virtual circuit switching (ATM, MPLS) as well as network virtualization (SDN, NFV), multimedia communications over packet switched networks (streaming, RTP, SIP, multicast), quality-of-service in packet switched networks (integrated services, RSVP, differentiated services, active queue management, policing, scheduling), wireless and mobile communications (GSM, UMTS, LTE, 5G, Wimax, WLAN, Bluetooth, sensor networks such as ZigBee). Industrial communication will also be a topic. In the tutorial practical tasks are performed in the laboratory: One laboratory contains several IP routers, switches and computers, IP phones and VoIP telephone software. Various configurations are set up and tested. Another part of the tutorial deals with mobile communications.
6	**Lernziele und Kompetenzen**	Die Studierenden erlangen Kenntnisse über Technologien bei der Leitungs- und Paketvermittlung in leitungsgebundenen und drahtlosen/mobilen Netzen Kenntnisse über die Grundlagen von Dienstgütemechanismen in paketvermittelten Netzen

		praktische Erfahrung in der Konfiguration eines IP-Switch-Router-Netzes mit Multimediaverkehr Students obtain the following learning targets and competences Knowledge of technologies in circuit and packet switching in wired and wireless/mobile networks Knowledge of the foundations of quality of service mechanisms in packet switched networks Practical experience in configuring an IP switch router network with multimedia traffic
7	**Voraussetzungen für die Teilnahme**	Basic knowledge of working with the Linux command line interface (terminal).
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Kurose, Ross. Computer Networking: A Top-Down Approach. 7th Ed., Pearson Education, 2017 W. Stallings. Data and Computer Communications, 10th ed., Pearson Education, 2014 W. Stallings. Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud, Pear-son Education, 2016 Cox. An Introduction to LTE. Wiley, 2012

1	**Modulbezeichnung** 93130	**Konzeptionelle Modellierung** (Conceptual modelling)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übungen zu Konzeptionelle Modellierung (2 SWS) Vorlesung: Konzeptionelle Modellierung (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	David Haller Prof. Dr.-Ing. Richard Lenz	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Richard Lenz
5	**Inhalt**	• Grundlagen der Modellierung • Datenmodellierung am Beispiel Entity-Relationship-Modell • Modellierung objektorientierter Systeme am Beispiel UML • Relationale Datenmodellierung und Anfragemöglichkeiten • Grundlagen der Metamodellierung • XML • Multidimensionale Datenmodellierung • Domänenmodellierung und Ontologien
6	**Lernziele und Kompetenzen**	Die Studierenden: • definieren grundlegende Begriffe aus der Datenbankfachliteratur • erklären die Vorteile von Datenbanksystemen • erklären die verschiedenen Phasen des Datenbankentwurfs • benutzen das Entity-Relationship Modell und das erweiterte Entity-Relationship Modell zur semantischen Datenmodellierung • unterscheiden verschiedene Notationen für ER-Diagramme • erläutern die grundlegenden Konzepte des relationalen Datenmodells • bilden ein gegebenes EER-Diagramm auf ein relationales Datenbankschema ab • erklären die Normalformen 1NF, 2NF, 3NF, BCNF und 4NF • definieren die Operationen der Relationenalgebra • erstellen Datenbanktabellen mit Hilfe von SQL • lösen Aufgaben zur Datenselektion und Datenmanipulation mit Hilfe von SQL • erklären die grundlegenden Konzepte der XML • erstellen DTDs für XML-Dokumente • benutzen XPATH zur Formulierung von Anfragen an XML-Dokumente • definieren die grundlegenden Strukturelemente und Operatoren des multidimensionalen Datenmodells • erklären Star- und Snowflake-Schema • benutzen einfache UML Use-Case Diagramme • benutzen einfache UML-Aktivitätsdiagramme • erstellen UML-Sequenzdiagramme • erstellen einfache UML-Klassendiagramme • erklären den Begriff Meta-Modellierung • definieren den Begriff der Ontologie in der Informatik • definieren die Begriffe RDF und OWL

7	**Voraussetzungen für die Teilnahme**	Gewünscht "Algorithmen und Datenstrukturen" und "Grundlagen der Logik und Logikprogrammierung"
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Masterprüfung Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222 Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• Elmasri, Ramez, and Sham Navathe. Grundlagen von Datenbanksystemen. Pearson Deutschland GmbH, 2009. - ISBN-10: 9783868940121 • Alfons Kemper, Andre Eickler: Datenbanksysteme : Eine Einführung. 6., aktualis. u. erw. Aufl. Oldenbourg, März 2006. - ISBN-10: 3486576909 • Bernd Oestereich: Analyse und Design mit UML 2.1. 8. Aufl. Oldenbourg, Januar 2006. - ISBN-10: 3486579266 • Ian Sommerville: Software Engineering. 8., aktualis. Aufl. Pearson Studium, Mai 2007. - ISBN-10: 3827372577 • Horst A. Neumann: Objektorientierte Softwareentwicklung mit der Unified Modeling Language. (UML). Hanser Fachbuch, März 2002. - ISBN-10: 3446188797 • Rainer Eckstein, Silke Eckstein: XML und Datenmodellierung. Dpunkt Verlag, November 2003. - ISBN-10: 3898642224

1	**Modulbezeichnung** 894856	**Künstliche Intelligenz I** (Artificial intelligence I)	**7,5 ECTS**
2	Lehrveranstaltungen	Übung: Übungen zu Künstliche Intelligenz I (2 SWS) Vorlesung: Artificial Intelligence I (4 SWS)	- 7,5 ECTS
3	Lehrende	PD Dr. Florian Rabe Prof. Dr. Michael Kohlhase PD Dr. Florian Rabe Prof. Dr. Michael Kohlhase	

4	**Modulverantwortliche/r**	Prof. Dr. Michael Kohlhase
5	**Inhalt**	Dieses Modul beschäftigt sich mit den Grundlagen der Künstlichen Intelligenz (KI), insbesondere formale Wissensrepräsentation, Heuristische Suche, Automatisches Planen und Schliessen unter Unsicherheit. --- This module covers the foundations of Artificial Intelligence (AI), in particular symbolic techniques based on search and inference.
6	**Lernziele und Kompetenzen**	- Wissen: Die Studierenden lernen grundlegende Repräsentationsformalismen und Algorithmen der Künstlichen Intelligenz kennen. - Anwenden: Die Konzepte werden an Beispielen aus der realen Welt angewandt (Übungsaufgaben). - Analyse: Die Studierenden lernen die über die modellierung in der Maschine menschliche Intelligenzleistungen besser einzuschätzen. Sozialkompetenz - Die Studierenden arbeiten in Kleingruppen zusammen um kleine Projekte zu bewältigen *Inhalt*: - Agentenmodelle als Grundlage der Künstlichen Intelligenz - Logisches Programmieren in Prolog - Heuristische Suche als Methode zur Problemlösung - Zwei-Agenten-Suche (automatisierung von Brettspielen) mittels heuristischer Suche - Constraint Solving/Propagation - Logische Sprachen für die Wissensrepräsentation

		- Inferenz and Automatisiertes Theorembeweisen (DPLL-Varianten und PL1)_ - Classisches Planen - Planen und Agieren in der wirklichen Welt. --- Technical, Learning, and Method Competencies - Knowledge: The students learn foundational representations and algorithms in AI. - Application: The concepts learned are applied to examples from the real world (homeworks). - Analysis: By modeling human cognitive abilities, students learn to assess and understand human intelligence better. - Social Competences: Students work in small groups to solve an AI game-play challenge/competition (Kalah). Contents: Foundations of symbolic AI, in particular: - Agent Models as foundation of AI - Logic Programming in Prolog - Heuristic Search as a methdod for problem solving - Adversarial Search (automating board games) via heuristic search - Constraint Solving/Propagation - Logical Languages for knowledge representation - Inference and automated theorem proving - Classical Planning - Planning and Acting in the real world.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222

10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 90 h Eigenstudium: 135 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Die Vorlesung folgt weitgehend dem Buch Stuart Russell und Peter Norvig: Artificial Intelligence: A Modern Approach. Prentice Hall, 3rd edition, 2009. Deutsche Ausgabe: Stuart Russell und Peter Norvig: Künstliche Intelligenz: Ein Moderner Ansatz. Pearson-Studium, 2004 (Übersetzung der 2. Auflage). ISBN: 978-3-8273-7089-1.

1	**Modulbezeichnung** 668129	**Machine Learning in Communications** (Machine learning in communications)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Machine Learning in Communications (4 SWS) Übung: Tutorial for Machine Learning in Communications (0 SWS)	5 ECTS -
3	Lehrende	Prof. Dr. Laura Cottatellucci Nikita Shanin	

4	**Modulverantwortliche/r**	Prof. Dr. Laura Cottatellucci
5	**Inhalt**	Recently, in many areas of wireless communications such as wireless sensor networks (WSNs), heterogeneous networks and complex ad hoc networks, distributed graph algorithms and machine learning on graphs are gaining relevance as fundamental tools in network analysis and information processing. This motivates to deliver a general introduction to fundamentals of machine learning such as detection of clusters on graphs. The introduction is followed by the application of machine learning to the design of physical and data layer techniques in wireless communications and in the optimization of mobile networks.
6	**Lernziele und Kompetenzen**	The students • know and explain the fundamentals of machine learning with special attention to machine learning over graphs. • apply these principles in the design and optimisation of wireless communications systems and mobile networks.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20191 Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 48440	**Machine Learning in Signal Processing** (Machine learning in signal processing)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Supplements for Machine Learning in Signal Processing (1 SWS) Vorlesung: Machine Learning in Signal Processing (3 SWS)	- 5 ECTS
3	Lehrende	Kamal Gopikrishnan Nambiar Prof. Dr. Vasileios Belagiannis	

4	**Modulverantwortliche/r**	PD Dr.Ing. Jürgen Seiler
5	**Inhalt**	This course is an introduction into machine learning and artificial intelligence. The special emphasis is on applications to modern signal processing problems. The course is focused on design principles of machine learning algorithms. The lectures start with a short introduction, where the nomenclature is defined. After this, probabilistic graphical models are introduced and the use of latent variables is discussed, concluding with a discussion of hidden Markov models and Markov fields. The second part of the course is about deep learning and covers the use of deep neural networks for machine learning tasks. In the last part of the lecture, the use of deep neural networks for speech processing tasks is introduced. The course is based on the materials and video footage from Dr. Roland Maas. He is an outstanding machine learning expert and a former member of the Chair of Multimedia Communications and Signal Processing.
6	**Lernziele und Kompetenzen**	After attending the lecture, students will be able to • understand regression and classification problems • apply PDF estimation algorithms • understand Gaussian mixture models and expectation-maximization • apply principal component analysis and independent component analysis • assess different estimation algorithms • explain the application of machine learning to system identification • apply hidden Markov models • understand different artificial neural network architectures • explain deep learning principles • apply artificial neural networks • devise learning strategies for deep neural networks • assess the application of deep neural networks for speech processing tasks.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1

9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich/mündlich
11	**Berechnung der Modulnote**	schriftlich/mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Literature: • C. M. Bishop: Pattern Recognition and Machine Learning, http://www.research.microsoft.com/en-us/um/people/cmbishop/PRML • S. Theodoridis and K. Koutroumbas: Pattern Recognition • M. Nielsen: Neural Networks and Deep Learning.

1	**Modulbezeichnung** 96030	**Medizinelektronik** (Medical electronics)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Medizinelektronik - Übung / Medical Electronics Exercises (2 SWS) Vorlesung: Medizinelektronik - Medical Electronics (2 SWS)	- 5 ECTS
3	Lehrende	Thomas Kurin Dr. Jens Kirchner	

4	**Modulverantwortliche/r**	Dr. Jens Kirchner
5	**Inhalt**	The Lecture and exercise deals with the following topics: • Electronics for medical diagnostics and therapy • Challenges for medical engineering from demographic development and epidemiology of common diseases • Concepts for chronic disease management and elderly care • Regulatory framework of circuit design for medical devices • Circuit design of standard medical equipment ECG, EEG, EMG, SpO2 • Sensor principles and circuit design for biosignal acquisition • Analog-digital balance • Energy management for medical devices • Body near energy harvesting • Health data transmission • Electronic systems for ambient assisted living (AAL) • Circuit technology for lab-on-chip and microelectromechanical systems (MEMS) • Circuit technology for implants and wearable systems
6	**Lernziele und Kompetenzen**	Students will gain • Substantial knowledge on principles of circuit design for medical electronic devices • Substantial knowledge on circuit design for standard medical devices, e.g. ECG, EEG, EMG • Substantial knowledge on design of medical sensors • Substantial knowledge on system design for health assistance systems, wearable medical devices and implants • Ability to analyze circuit diagrams of medical electronic devices • Ability to separate medical electronic devices into their subfunctions • Ability to analyze energy budget of medical devices, particularly wearable systems • Basic ability to design electronic circuits to comply with regulatory requirements
7	**Voraussetzungen für die Teilnahme**	Completion of the modules "Circuit design" ("Schaltungstechnik") or "Electronics and circuit design" ("Elektronik und Schaltungstechnik") is recommended before attending the course.
8	**Einpassung in Studienverlaufsplan**	Semester: 1

9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 722831	**Middleware - Cloud Computing (Vorlesung mit erweiterten Übungen)** (Lecture and advanced tutorial: Middleware - Cloud computing)	**7,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Middleware - Cloud Computing (2 SWS) Übung: Middleware - Cloud Computing - Übungen (2 SWS) Übung: Middleware - Cloud Computing - Rechnerübungen (2 SWS) Übung: Middleware - Cloud Computing - Erweiterte Übungen (2 SWS)	- - - 5 ECTS
3	Lehrende	PD Dr.Ing. Tobias Distler Laura Lawniczak	

4	**Modulverantwortliche/r**	PD Dr.Ing. Tobias Distler
5	**Inhalt**	• Überblick Cloud Computing • Grundlagen verteilter Programmierung (Web Services/SOAP/ REST) • Virtualisierung als Basis für Cloud Computing • Infrastructure as a Service (IaaS) am Beispiel von Eucalyptus und Amazon EC2 • Skalierbare Verarbeitung von großen Datenmengen • Interoperabilität und Multi-Cloud Computing • Fehlertoleranz im Kontext von Cloud Computing • Aktuelle Forschungstrends
6	**Lernziele und Kompetenzen**	Studierende, die das Modul erfolgreich abgeschlossen haben: - nennen unterschiedliche Ausprägungen von Cloud-Computing. - erläutern verschiedene Cloud-Architekturen. - stellen Vor- und Nachteile von Cloud-Computing gegenüber. - unterscheiden die Herangehensweisen bei der Entwicklung von SOAP- im Vergleich zu REST-Anwendungen. - organisieren den Austausch von Informationen in einer verteilten Anwendung unter Verwendung eines Verzeichnisdienstes. - entwickeln eigene auf Web-Services basierende Anwendungen. - bewerten die Vor- und Nachteile der Bündelung von Nachrichten beim Aufruf von Web-Services. - erläutern die Anforderungen an ein virtualisiertes System. - beschreiben die für die Virtualisierung eines Systems erforderlichen Kriterien.

- vergleichen zwischen unterschiedlichen Virtualisierungstechniken und -ebenen.

- schildern den Aufbau und die Funktionsweise von Xen und Linux-VServer.

- erproben das Einrichten eines Abbilds für eine virtuelle Maschine.

- skizzieren die Architektur einer Infrastruktur-Cloud sowie die Aufgabenbereiche hierfür zentraler Komponenten am Beispiel von Eucalyptus.

- erproben das Bereitstellen von Anwendungen in einer Infrastruktur-Cloud.

- zeigen die Grundlagen Software-definierter Netzwerke am Beispiel von Onix und B4 auf.

- bewerten verschiedene im Bereich Cloud-Computing zum Einsatz kommende Datenspeichersysteme (Google File System, Bigtable, Windows Azure Storage, Amazon Dynamo) hinsichtlich der Kriterien Verfügbarkeit, Konsistenz und Partitionstoleranz.

- erläutern eine auf Vektoruhren basierende Methode zur Auflösung im Zusammenhang mit letztendlicher Konsistenz auftretender Konflikte.

- entwickeln ein verteiltes, repliziertes Dateisystem nach dem Vorbild von HDFS, das auf die hierarchische Speicherung großer Datenmengen ausgelegt ist.

- erkunden das Bereitstellen selbst entwickelter Dienste mittels Docker.

- erstellen ein Framework zur parallelen Bearbeitung von Daten nach dem Vorbild von MapReduce.

- konzipieren eigene MapReduce-Anwendungen zur Verarbeitung strukturierter bzw. unstrukturierter Rohdaten.

- diskutieren die Fehlertoleranzmechanismen in Google MapReduce.

- schildern die grundsätzliche Funktionsweise von Systemen zur Kühlung von Datenzentren mittels Umgebungsluft.

- beschreiben das Grundkonzept einer temperaturabhängigen Lastverteilung von Prozessen in einem Datenzentrum.

- stellen diverse Ansätze zur Erhöhung der Energieeffizienz von MapReduce-Clustern gegenüber.

		- unterscheiden die Architekturen und Funktionsweisen der Koordinierungsdienste Chubby und ZooKeeper. - entwickeln einen eigenen Koordinierungsdienst nach dem Vorbild von ZooKeeper. - ermitteln die Konsistenzeigenschaften der eigenen Koordinierungsdienstimplementierung. - erläutern unterschiedliche Ansätze zur Reduzierung bzw. Tolerierung von Tail-Latenz. - skizzieren das Grundkonzept von Erasure-Codes. - beschreiben den Aufbau eines auf die Clouds mehrerer Anbieter gestützten Datenspeichersystems. - erläutern den Einsatz passiver Replikation zur Bereitstellung von Fehlertoleranzmechanismen für virtuelle Maschinen am Beispiel von Remus. - schildern die Grundlagen der Migration von virtuellen Maschinen. - bewerten die Qualität einer aktuellen Publikation aus der Fachliteratur. - erschließen sich typische Probleme (Nebenläufigkeit, Konsistenz, Skalierbarkeit) und Fehlerquellen bei der Programmierung verteilter Anwendungen. - können in Kleingruppen kooperativ arbeiten. - können ihre Entwurfs- und Implementierungsentscheidungen kompakt präsentieren und argumentativ vertreten. - können offen und konstruktiv mit Schwachpunkten und Irrwegen umgehen. - reflektieren ihre Entscheidungen kritisch und leiten Alternativen ab.
7	**Voraussetzungen für die Teilnahme**	Gute Programmierkenntnisse in Java
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio Erfolgreiche Bearbeitung aller während des Semesters gestellten Übungsaufgaben (6 Aufgaben, Bewertung jeweils mit "ausreichend") + 30-minütige mündliche Prüfung am Ende des

		Semesters. Die Modulnote ergibt sich zu 100% aus der Bewertung der mündlichen Prüfung.
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 90 h Eigenstudium: 135 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 557235	**Middleware - Cloud Computing (Vorlesung mit Übungen)** (Lecture and tutorial: Middleware - Cloud computing)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Middleware - Cloud Computing (2 SWS) Übung: Middleware - Cloud Computing - Übungen (2 SWS) Übung: Middleware - Cloud Computing - Rechnerübungen (2 SWS)	- - -
3	Lehrende	PD Dr.Ing. Tobias Distler Laura Lawniczak	

4	**Modulverantwortliche/r**	PD Dr.Ing. Tobias Distler
5	**Inhalt**	• Überblick Cloud Computing • Grundlagen verteilter Programmierung (Web Services/SOAP/ REST) • Virtualisierung als Basis für Cloud Computing • Infrastructure as a Service (IaaS) am Beispiel von Eucalyptus und Amazon EC2 • Skalierbare Verarbeitung von großen Datenmengen • Interoperabilität und Multi-Cloud Computing • Fehlertoleranz im Kontext von Cloud Computing • Aktuelle Forschungstrends
6	**Lernziele und Kompetenzen**	Studierende, die das Modul erfolgreich abgeschlossen haben: - nennen unterschiedliche Ausprägungen von Cloud-Computing. - erläutern verschiedene Cloud-Architekturen. - stellen Vor- und Nachteile von Cloud-Computing gegenüber. - unterscheiden die Herangehensweisen bei der Entwicklung von SOAP- im Vergleich zu REST-Anwendungen. - organisieren den Austausch von Informationen in einer verteilten Anwendung unter Verwendung eines Verzeichnisdienstes. - entwickeln eigene auf Web-Services basierende Anwendungen. - erläutern die Anforderungen an ein virtualisiertes System. - beschreiben die für die Virtualisierung eines Systems erforderlichen Kriterien. - vergleichen zwischen unterschiedlichen Virtualisierungstechniken und -ebenen. - schildern den Aufbau und die Funktionsweise von Xen und Linux-VServer.

- erproben das Einrichten eines Abbilds für eine virtuelle Maschine.

- skizzieren die Architektur einer Infrastruktur-Cloud sowie die Aufgabenbereiche hierfür zentraler Komponenten am Beispiel von Eucalyptus.

- erproben das Bereitstellen von Anwendungen in einer Infrastruktur-Cloud.

- zeigen die Grundlagen Software-definierter Netzwerke am Beispiel von Onix und B4 auf.

- bewerten verschiedene im Bereich Cloud-Computing zum Einsatz kommende Datenspeichersysteme (Google File System, Bigtable, Windows Azure Storage, Amazon Dynamo) hinsichtlich der Kriterien Verfügbarkeit, Konsistenz und Partitionstoleranz.

- erläutern eine auf Vektoruhren basierende Methode zur Auflösung im Zusammenhang mit letztendlicher Konsistenz auftretender Konflikte.

- entwickeln ein verteiltes Dateisystem nach dem Vorbild von HDFS, das auf die Speicherung großer Datenmengen ausgelegt ist.

- erkunden das Bereitstellen selbst entwickelter Dienste mittels Docker.

- erstellen ein Framework zur parallelen Bearbeitung von Daten nach dem Vorbild von MapReduce.

- konzipieren eigene MapReduce-Anwendungen zur Verarbeitung strukturierter Rohdaten.

- diskutieren die Fehlertoleranzmechanismen in Google MapReduce.

- schildern die grundsätzliche Funktionsweise von Systemen zur Kühlung von Datenzentren mittels Umgebungsluft.

- beschreiben das Grundkonzept einer temperaturabhängigen Lastverteilung von Prozessen in einem Datenzentrum.

- stellen diverse Ansätze zur Erhöhung der Energieeffizienz von MapReduce-Clustern gegenüber.

- unterscheiden die Architekturen und Funktionsweisen der Koordinierungsdienste Chubby und ZooKeeper.

- entwickeln einen eigenen Koordinierungsdienst nach dem Vorbild von ZooKeeper.

		- ermitteln die Konsistenzeigenschaften der eigenen Koordinierungsdienstimplementierung. - erläutern unterschiedliche Ansätze zur Reduzierung bzw. Tolerierung von Tail-Latenz. - skizzieren das Grundkonzept von Erasure-Codes. - beschreiben den Aufbau eines auf die Clouds mehrerer Anbieter gestützten Datenspeichersystems. - erläutern den Einsatz passiver Replikation zur Bereitstellung von Fehlertoleranzmechanismen für virtuelle Maschinen am Beispiel von Remus. - schildern die Grundlagen der Migration von virtuellen Maschinen. - bewerten die Qualität einer aktuellen Publikation aus der Fachliteratur. - erschließen sich typische Probleme (Nebenläufigkeit, Konsistenz, Skalierbarkeit) und Fehlerquellen bei der Programmierung verteilter Anwendungen. - können in Kleingruppen kooperativ arbeiten. - können ihre Entwurfs- und Implementierungsentscheidungen kompakt präsentieren und argumentativ vertreten. - können offen und konstruktiv mit Schwachpunkten und Irrwegen umgehen. - reflektieren ihre Entscheidungen kritisch und leiten Alternativen ab.
7	**Voraussetzungen für die Teilnahme**	Gute Programmierkenntnisse in Java
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio Erfolgreiche Bearbeitung aller während des Semesters gestellten Übungsaufgaben (6 Aufgaben, Bewertung jeweils mit "ausreichend") + 30-minütige mündliche Prüfung am Ende des Semesters. Die Modulnote ergibt sich zu 100% aus der Bewertung der mündlichen Prüfung.
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	nur im Wintersemester

13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 96300	**MIMO Communication Systems** (MIMO communication systems)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: MIMO Communication Systems (3 SWS) Übung: MIMO Communication Systems - Tutorial (1 SWS)	5 ECTS -
3	Lehrende	Prof. Dr.-Ing. Robert Schober Hedieh Ajam	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Robert Schober
5	**Inhalt**	Modern communication systems employ multiple antennas at the transmitter and/or receiver creating a multiple-input multiple-output (MIMO) system. This course covers the fundamental mathematical and communication theoretical concepts necessary for the design and analysis of MIMO communication systems. Relevant topics include MIMO Channel Capacity, Receive Diversity, Transmit Diversity, Space-Time Coding, Spatial Multiplexing, MIMO Transceiver Design, Multi-user MIMO, Massive MIMO, Relay-based MIMO, and applications in modern communication systems.
6	**Lernziele und Kompetenzen**	The students • learn about different MIMO channel models, • analyze MIMO communication systems with respect to their channel capacity and reliability, • determine MIMO figures of merit such as coding gain, diversity gain, and multiplexing gain, • compare and evaluate different MIMO receiver designs, • characterize the rate region of multiuser systems, • analyze massive MIMO systems, • discuss the advantages and disadvantages of different relay network architectures. Die Studierenden • lernen verschiedene MIMO-Kanalmodelle kennen, • analysieren MIMO-Kommunikationssysteme hinsichtlich der Kanalkapazität und Zuverlässigkeit, • ermitteln MIMO-Kenngrößen wie Codierungsgewinn, Diversitätsgewinn und Multiplexgewinn, • vergleichen und beurteilen verschiedene MIMO-Empfangsstrategien, • charakterisieren die Ratenregion von Mehrteilnehmersystemen, • analysieren Massive-MIMO-Systeme, • diskutieren die Vor- und Nachteile verschiedener Relaisnetzwerkarchitekturen.
7	**Voraussetzungen für die Teilnahme**	Basic course in communications
8	**Einpassung in Studienverlaufsplan**	Semester: 1

9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20191 Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 43141	**Mobile Communications** (Mobile communications)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Mobile Communications - Tutorial (1 SWS) Vorlesung: Mobile Communications (3 SWS)	- 5 ECTS
3	Lehrende	Dr.-Ing. Ali Bereyhi Prof. Dr.-Ing. Ralf Müller	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Ralf Müller
5	**Inhalt**	History of mobile communications, cellular systems, sectorization, spectral efficiency, co-channel interference, adjacent-channel interference, near-far effect, cellular network architecture, antenna types and parameters, free space propagation, reflection, attenuation, diffraction, scattering, classification of channel models, ground reflection model, Okumura-Hata model, shadowing, narrow-band fading, time- variant channels, scattering function, delay-Doppler spectrum, diversity principles, combining methods, diversity gain, multiplexing, duplexing, digital modulation, Gaussian filtered minimum shift keying, basics of channel coding, interleaving, global system for mobile communications, physical versus logical channels, frame structure, call set-up, synchronization, channel estimation, hand-off
6	**Lernziele und Kompetenzen**	The students explain the cellular structure of mobile communication systems. They students explain the physical mechanics of radio wave propagation in the cm-band. The students explain the GSM cellular communications standard. The students discuss the pros and cons of several multiple-access and duplexing methods. The students discuss the pros and cons of several modulation and coding formats. The students decide which antenna type is suitable for a given morphological structure of the environment. The students predict the amplitude and dynamic of the antenuation between a mobile transmitter and a fixed receiver. The students utilize diversity methods to improve the link quality. The students determine the coverage probability of a given cellular communication system. The students collaborate on solving exercise problems. The students discuss which system solutions fit to which environments.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 2
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)

12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Proakis, J.: Digital Communications, McGraw-Hill, 4th ed., 2001. Rappaport, T.: Wireless Communications: Principles & Practice, Prentice Hall, 2nd ed., 2001. Mouly, M., Paulet, M.: The GSM System for Mobile Communications, Cell & SYS, France, 1992. Goldsmith, A.: Wireless Communications, Cambridge Univ. Press, 2005.

1	**Modulbezeichnung** 330467	**Multimedia Security** (Multimedia security)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Multimedia Security Exercises (2 SWS) Vorlesung: Multimedia Security (2 SWS)	- -
3	Lehrende	PD Dr.Ing. Christian Rieß	

4	**Modulverantwortliche/r**	
5	**Inhalt**	This lecture covers a variety of security-related topics around multimedia data. In particular, the lecture presents algorithms and key results from the past 15 years in multimedia security, including topics on image forensics, steganography, watermarking, and biometrics. Selected algorithms are implemented and tested by the participants. It is helpful to bring some knowledge in signal processing or pattern recognition. It is also helpful to be not afraid from equations. Tentative list of topics and algorithms: • Image forensics for manipulation detection in digital media. Statistical and physics-based features for manipulation detection. Detecting traces of manipulations versus validating image authenticity. • Blind source attribution: was an image or video captured with a particular camera? • Steganography for covert communication. Fundamental algorithms, when can their application be detected? • Watermarking for copyright protection in images/video. Fundamental algorithms, and their security. • Biometric features for person re-identification, and practical concerns on their implementation.
6	**Lernziele und Kompetenzen**	Fachkompetenz Verstehen Die Studierenden fassen die wesentlichen Fragestellungen auf dem Gebiet der Multimediasicherheit zusammen (The participants summarize the relevant questions within the field of multimedia security). Die Studierenden nennen und erklären die wesentlichen Fachbegriffe aus den Teilgebieten der Multimediasicherheit (The participants name and explain relevant terms from the subfields of multimedia security). Evaluieren (Beurteilen) Die Studierenden bewerten die Eignung der vorgestellten bildforensischen Algorithmen für ein gegebenes Untersuchungsszenario (The participants evaluate the suitability of the presented image forensics algorithms for a given examination scenario).

		Erschaffen Die Studierenden implementieren kurze Beispielsprogramme für ausgewählte Algorithmen der Multimediasicherheit (The participants implement short example programs for selected algorithms of multimedia security). Lern- bzw. Methodenkompetenz Die Studierenden implementieren ausgewählte Methoden in der Programmiersprache C++ (The participants implement selected methods in the C++ programming language). Sozialkompetenz Die Studierenden implementieren und diskutieren Beispielsmethoden in Gruppenarbeit (The participants implement and discuss the example method in groups). Die Studierenden diskutieren die gesellschaftlichen Auswirkung von Multimediasicherheit am Beispiel aktueller Probleme (The participants discuss multimedia security's impact on society using current issues).
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 120 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	begleitend zu der Veranstaltung: • Husrev Sencar, Nasir Memon (Editors): "Digital Image Forensics. There is More to a Picture than Meets the Eye", Springer 2013. • Hany Farid: "Photo Forensics", MIT Press, 2016.

1	**Modulbezeichnung** 302148	**Musiksignalverarbeitung - Analyse** (Music processing - Analysis)	**2,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Music Processing Analysis (2 SWS)	2,5 ECTS
3	Lehrende	Prof. Dr. Meinard Müller	

4	**Modulverantwortliche/r**	Prof. Dr. Meinard Müller
5	**Inhalt**	Music signals possess specific acoustic and structural characteristics that are not shared by spoken language or audio signals from other domains. In fact, many music analysis tasks only become feasible by exploiting suitable music-specific assumptions. In this course, we study feature design principles that have been applied to music signals to account for the music-specific aspects. In particular, we discuss various musically expressive feature representations that refer to musical dimensions such as harmony, rhythm, timbre, or melody. Furthermore, we highlight the practical and musical relevance of these feature representations in the context of current music analysis and retrieval tasks. Here, our general goal is to show how the development of music-specific signal processing techniques is of fundamental importance for tackling otherwise infeasible music analysis problems.
6	**Lernziele und Kompetenzen**	Expertise Understand • The students present central tasks in music processing in their own words and outline possible solutions. • The students understand the properties of different forms of representation of music. Apply • The students apply basic algorithms for the analysis and comparison of music signals. • Students can predict how different musical properties will affect the signal analysis. Analyze • The students observe and discuss the meaning and impact of parameters in music analysis. • The students compare different methods of analyzing periodicities. Evaluate • The students question assumptions that are often implicitly made when using analytical methods. • Students estimate when methods might work when analyzing specific music signals and when they typically fail.

Learning and methodological skills

- The students prepare for the lecture using selected literature and Jupyter notebooks.
- The students question existing approaches regarding their applicability in practice.
- The students pay attention to efficiency issues in the algorithms discussed.

Self-competence

- The students question their understanding of what they have learned using exercises.
- The students formulate questions and ask them to the lecturer and the audience in the lecture.

Social skills

- The students independently organize learning groups in which the subject is discussed and deepened.
- The students simulate oral exams with their fellow students.

Fachkompetenz

Verstehen

- Die Studierenden stellen zentrale Aufgabenstellungen der Musikverarbeitung in eigenen Worten dar und skizzieren Lösungsansätze.
- Die Studierenden verstehen die Eigenschaften von unterschiedlichen Darstellungsformen von Musik.

Anwenden

- Die Studierenden wenden grundlegende Algorithmen zur Analyse und zum Vergleich von Musiksignalen an.
- Die Studierenden können voraussagen, wie sich unterschiedliche musikalische Eigenschaften bei der Signalanalyse auswirken.

Analysieren

- Die Studierenden beobachten und diskutieren die Bedeutung und Auswirkung von Parametern bei der Musikanalyse.
- Die Studierenden stellen unterschiedliche Verfahren bei der Analyse von Periodizitäten gegenüber.

Evaluieren (Beurteilen)

		• Die Studierenden hinterfragen Annahmen, die implizit bei der Verwendung von Analysemethoden gemacht werden. • Die Studierenden schätzen ein, wann Methoden bei der Analyse von gewissen Musiksignalen funktionieren könnten und wann sie typischerweise versagen. Lern- bzw. Methodenkompetenz • Die Studierenden bereiten sich auf die Vorlesung anhand ausgewählter Literatur vor. • Die Studierenden hinterfragen bestehende Ansätze hinsichtlich ihrer Anwendbarkeit in der Praxis. • Die Studierenden beachten Fragen der Effizienz bei den diskutierten Algorithmen. Selbstkompetenz • Die Studierenden hinterfragen ihr Verständnis von dem Gelernten anhand von Übungsaufgaben. • Die Studierenden formulieren Fragen und stellen diese in der Vorlesung an den Dozenten und die Zuhörerschaft. Sozialkompetenz • Die Studierenden organisieren selbständig Lerngruppen, in denen der Stoff diskutiert und vertieft wird. • Die Studierenden simulieren mit ihren Kommilitonen mündliche Prüfungen.
7	**Voraussetzungen für die Teilnahme**	In this course, we discuss a number of current research problems in music processing or music information retrieval (MIR) covering aspects from information science and digital signal processing. We provide the necessary background information and give numerous motivating examples so that no specialized knowledge is required. However, the students should have a solid mathematical background. The lecture is accompanied by readings from textbooks or the research literature. Furthermore, the students are required to experiment with the presented algorithms using Python.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester

13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	http://www.music-processing.de http://www.springer.com/gp/book/9783319219448

1	**Modulbezeichnung** 502007	**Musiksignalverarbeitung - Synthese** (Music processing - synthesis)	**2,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Musikverarbeitung - Synthese (2 SWS)	-
3	Lehrende	Maximilian Schäfer	

4	**Modulverantwortliche/r**	Maximilian Schäfer
5	**Inhalt**	• Verarbeitung von Audiosignalen durch parametrische Filter und Effekte • Erzeugung von künstlichen Klängen mit Mitteln der digitalen Klangsynthese • Klangwiedergabe in echten und virtuellen Räumen • Klangbeispiele und Demonstrationen • Programmiersprachen für Audio-Echtzeit-Verarbeitung *Content*: • a short history of electrical and electronic music • processing of audio signals by parametric filters and effects • digital sound synthesis • sound reproduction in real and in virtual environments • sound examples and demonstrations • programming languages for audio real-time processing
6	**Lernziele und Kompetenzen**	Die Studierenden • beschreiben die speziellen Anforderungen für Audio-Echtzeit-Verarbeitung, • wenden ihre theoretischen Kenntnisse zeitdiskreter Signale und Systeme für die Verarbeitung und Erzeugung musikalischer Klänge an, • gestalten eigene Software-Realisierungen zur Klangsynthese, • entwerfen technische Systeme für musikalisch motivierte Aufgabenstellungen. The students • specify the special requirements for audio realtime processing, • apply their theoretical knowledge about discrete-time signals and systems to processing and synthesis of musical sounds, • design their own software realizations for sound synthesis • implement technical systems for digital music.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191

		Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Das Vorlesungsskript und weitere Zusatzmaterialien zur Vorlesung werden via StudOn zur Verfügung gestellt.

1	**Modulbezeichnung** 849203	**Optische Kommunikationsnetze** (Optical communication networks)	**2,5 ECTS**
2	Lehrveranstaltungen	Zu diesem Modul sind keine Lehrveranstaltungen oder Lehrveranstaltungsgruppen hinterlegt!	
3	Lehrende	Zu diesem Modul sind keine Lehrveranstaltungen und somit auch keine Lehrenden hinterlegt!	

4	**Modulverantwortliche/r**	Prof. Dr. Herbert Haunstein
5	**Inhalt**	Global communication between billions of subscribers utilizing a multitude of devices is accomplished over a trans-continental fiber-optic transport network. End users worldwide access this network over copper cable (xDSL, HFC), by wireless technologies like WLAN, GSM, UMTS, LTE and also via GPON, EPON and WDM-PON (PON: Passive Optical Network). After a short distance ("the last mile") data streams from many users are aggregated (e.g. by IP routers) into higher data rate transport streams, which are then carried over cost-efficient and highly reliable optical connections. Rapid increase of data traffic has quickly evolved from Gigabit Ethernet (1GbE) to 10GbE and 100GbE data rates. To operate optical networks on a global scale, standards like OTN (Optical Transport Network) have been developed to provide high capacity links by use of many wavelengths together with operations and maintenance (OAM) functions. Automated protection and restauration schemes provide a high level of availabilty and can guarantee carrier-grade Quality of Service (QoS). Future data rate increase will be driven by video streaming as well as the introduction of 5G wireless technology and the Internet of Things (IoT). The course shall provide a fundamental understanding of modern fiber optic networks from fixed and mobile access through metropolitan area to core networks. 1) Introduction – Evolution of optical networks 2) Network layers - Internet Protocol – TCP/IP 3) Label switching – MPLS – MPLS-TP 4) Quality of Service - traffic classification – resource allocation 5) Ethernet - switching and physical transport 6) Optical Transport Network - OTN 7) Optical fiber properties – optical amplification 8) Optical transmitter – laser – modulator 9) Optical receiver – photo detection – Clock&Data recovery – Bit Error Ratio calculation 10) Modulation formats – transmission - margin allocation 11) Coherent detection – optical signal processing

		12) Optical networks – optical switching 13) Network control & automation
6	**Lernziele und Kompetenzen**	Students ... • explain the functional building blocks of optical networks • can eloborate on the different tasks provided by the logical/ control plane (routing), the physical layer and transmission/data plane of optical networks • refer which standardisation organisation contributes to the different function of optical networks • explain the purpose of different protocols that interact along an end-to-end communication channel • describe technologies for E/O and O/E conversion and optical switches • express the design challenges of future optical systems for fixed and mobile access, data center interconnects, • metro-regional, core, ultra-long-haul and submarine networks
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch oder Englisch
16	**Literaturhinweise**	1) R. Ramaswami and K.N. Sivarajan: “Optical Networks”, Morgan Kaufman Publishers, 1998 2) U. Black: “Optical Networks - Third generation transport systems”, Prentice Hall, 2002 3) P. Tomsu and Chr. Schmutzer: “Next generation optical networks”, Prentice Hall, 2002 4) M. Bossert, M. Breitbach: “Digitale Netze”, Teubner Verlag, 1997 5) I. Kaminow and T. Li (eds.): “Optical fiber telecommunications IVA+B”, Academic Press, 2002 6) D.E. Comer, „Computernetworks and Internets, Pearson“, 2009 7) G.P. Agrawal, “Fiber optic communication systems”, Wiley, 1992, (new 1997) 8) G.P. Agrawal, “Nonlinear fiber optics”, Academic Press, 1995

		9) K. Petermann: “Laser Diode Modulation and Noise”, Kluver, 1991 10) L. Kazovsky et al., „Optical Fiber Communication Systems“, Artech House, 1996 11) K.-P. Ho, „Phase-Modulated Optical Communication Systems“, Springer 2005 12) H. Haunstein, Presentation material (slides) of the lectures (in English)

1	**Modulbezeichnung** 92400	**Optische Übertragungstechnik** (Optical communication systems)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Optische Übertragungstechnik Übung (2 SWS) Vorlesung: Optische Übertragungstechnik (2 SWS)	- 5 ECTS
3	Lehrende	Esther Renner Benedikt Beck Prof. Dr.-Ing. Bernhard Schmauß	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Bernhard Schmauß
5	**Inhalt**	Kommerzielle Optische Kommunikationssysteme erreichen pro Faser Übertragungskapazitäten von mehreren Tbit/s. Im Labor wurden mehr als 100Tbit/s nachgewiesen. Die Realisierung derartiger Systeme setzt die Beherrschung verschiedenster Techniken der optischen Übertragungstechnik voraus. In der Vorlesung werden Techniken des Zeitbereichs - (TDM) und Wellenlängenmultiplex (WDM), aber besonders auch der Auslegung der Übertragungsstrecke (Link Design) auf der Basis entsprechender physikalischer und signaltheoretischer Grundlagen behandelt und vertieft. Dabei werden Verfahren besprochen, die sicherstellen, dass sowohl die Signalverzerrungen durch lineare und nichtlineare Fasereffekte als auch die Akkumulation des Verstärkerrauschens begrenzt bleiben. Es wird ausführlich die Systemoptimierung hinsichtlich des optischen Signal-Rausch-Verhältnisses (OSNR) diskutiert sowie auf Techniken des Dispersions- und Nichtlinearitätsmanagements (z.B. Solitonenübertragung) eingegangen. Hierbei wird dem Themenkomplex einer optimalen Streckenauslegung besonders eingehend behandelt. In der Folge werden verschiedene, gebräuchliche Modulationsverfahren einschließlich kohärenter Übertragungsverfahren behandelt, die in neueren Systemen eingesetzt und in experimentellen Systemen getestet werden. Eine Besprechung optischer Verfahren zur Signalregeneration bildet die Brücke zu aktuellen eigenen Forschungsarbeiten. Die vermittelten Grundlagen werden in der Übung zur Vorlesung durch praxisnahe und anschauliche Simulationsbeispiele vertieft.
6	**Lernziele und Kompetenzen**	Die Studierenden • besitzen spezialisiertes und vertieftes Wissen über die Konzeption und Struktur verschiedener optischer Übertragungssysteme. • können die Qualität optischer Datensignale im Kontext verschiedener Systemkonzepte vergleichen und bewerten • sind in der Lage Streckenauslegungen zu entwickeln und zu optimieren. • besitzen methodische Kenntnis zur Bestimmung und Verbesserung der Leistungsfähigkeit optischer Übertragungsstrecken unter Einbeziehung aktueller wissenschaftlicher Ergebnisse.
7	**Voraussetzungen für die Teilnahme**	Komponenten optischer Kommunikationssysteme hilfreich aber nicht obligatorisch

8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Agrawal, G.P.: Fiber-Optic Communication Systems, John Wiley & Sons, 1997 Agrawal, G.P.: Nonlinear Fiber Optics, John Wiley & Sons, 3. Auflage, 2001 Kaminow, I, Koch, T.: Optical Fiber Telecommunications IVA, Academic Press, 2002 Skriptum zur Vorlesung Kaminow, I, Li, T., Willner,A.: Optical Fiber Telecommunications VA, Academic Press, 2008

1	**Modulbezeichnung** 740665	**Parallele Systeme (Vorlesung mit erweiterten Übungen)** (Parallel systems with extended exercises)	**7,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Parallele Systeme (2 SWS) Übung: Übung zu Parallele Systeme (2 SWS) Übung: Erweiterte Übungen zu Parallele Systeme (2 SWS)	2,5 ECTS 2,5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr.-Ing. Jürgen Teich PD Dr.Ing. Frank Hannig Michael Witterauf Stefan Groth Marcel Brand	

4	**Modulverantwortliche/r**	Joachim Falk PD Dr.Ing. Frank Hannig Prof. Dr.-Ing. Jürgen Teich
5	**Inhalt**	Aktuelle PCs verfügen über Mehrkernprozessoren und Grafikkarten, die wiederum aus hunderten von einfachen Prozessoren bestehen können. Hierdurch wird ein hohes Maß an nebenläufiger Datenverarbeitung möglich, welche bis vor einigen Jahren nur in Großrechnern erreicht werden konnte. Die effiziente Ausnutzung dieser Parallelität bedarf allerdings mehr als nur mehrerer Prozessoren, insbesondere muss das zu lösende Problem Parallelverarbeitung erlauben. In dieser Vorlesung werden Eigenschaften unterschiedlicher paralleler Rechnerarchitekturen und Metriken zu deren Beurteilung behandelt. Weiterhin werden Modelle und Sprachen zum Programmieren paralleler Rechner eingeführt. Neben der Programmierung von allgemeinen Parallelrechnern werden Entwurfsmethoden (CAD) vorgestellt, wie man ausgehend von einer algorithmischen Problemstellung ein massiv paralleles Rechenfeld in VLSI herleiten kann, das genau dieses Problem optimal parallel berechnet. Solche Schaltungen spielen auf der Bit- bzw. Wortebene eine dominante Rolle (Arithmetik) sowie bei Problemen der Signal- und Bildverarbeitung (z.B. Filter). Im Einzelnen werden behandelt: 1) Theorie der Parallelität (parallele Computermodelle, parallele Spezifikationsformen und -sprachen, Performanzmodelle und -berechnung) 2) Klassifikation paralleler und skalierbarer Rechnerarchitekturen (Multiprozessoren und Multicomputer, Vektorrechner, Datenflussmaschinen, VLSI-Rechenfelder) 3) Programmierbare System-on-Chip (SoC) und Mehrkern-Architekturen (Grafik-Prozessoren, Cell, etc.) 4) Programmierung paralleler Rechner (Sprachen und Modelle, Entwurfsmethoden und Compiler, Optimierung) 5) Massive Parallelität: Vom Algorithmus zur Schaltung 6) Praktische Übungen mit rechnergestützten Werkzeugen

		Today's PCs consist of multi-core processors and graphics cards that again comprise hundreds to thousands of simple processors. As a result of this, a very high degree of parallel data processing becomes possible, which was subjected to supercomputers a couple of years ago. The efficient exploitation of parallel processing requires not only multiple processors but also parallelism inherent in the problem to be solved. In this lecture, properties of different parallel computer architectures and corresponding quality metrics are examined. Further, models and parallel programming languages are introduced. In addition to programming general parallel computers, design methods (CAD) are presented that systematically transform an algorithmic problem description into a massive parallel processor array (VLSI), which can optimally execute the given problem in parallel. Such highly parallel circuits play an essential role at the bit level and circuit level (arithmetics) as well as in the case of signal processing and image processing (e.g., filter). *In detail, the following topics are covered:* 1) Theory of parallelism (parallel models of computation, parallel specification and parallel languages, performance models) 2) Classification of parallel and scalable computer architectures (multi-processors and multi-computers, vector computers, data-flow machines, VLSI processor arrays) 3) Programmable System-on-Chip (SoC) and multi-core architetcures (graphics processors, Cell, etc.) 4) Programming of parallel computers (languages and models, design methods and compiler, optimization) 5) Massive parallelism: From algorithm to circuit 6) Practical training with computer-aided design tools
6	**Lernziele und Kompetenzen**	Schwerpunkt der Vorlesung ist die Vermittlung von Grundlagen der parallelen Datenverarbeitung. *The focus of this lecture are foundations of parallel data processing.* Fachkompetenz - Verstehen • Die Studierenden verstehen grundlegende Konzepte der parallelen Datenverarbeitung, sowohl theoretischer Art anhand von Modellen, als auch an Architekturbeispielen. The students become familiar with the fundamentals of parallel data processing, theoretic in the form of models as well as by architecture examples. Fachkompetenz - Anwenden • Die Studierenden setzen sich mit modernen eingebetteten parallelen Ein-Chip-Architekturen auseinander. The students

		get familiar with modern embedded parallel system-on-chip architectures. • Die Studierenden wenden grundlegende Performanzmodelle und Parallelisierungtechniken zur Analyse und Optimierung von parallelen Algorithmen und Architekturen an. The students exercise basic performance models and parallelization techniques for the analysis and optimization of parallel algorithms and architectures. • Die Studierenden setzen die Modellierung und den Entwurf von massiv-parallelen Prozessorfeldern in konkreten Aufgaben selbstständig um. In concrete tasks, the students apply independently the modeling and the design of massively parallel processors arrays. • Die Studierenden wenden das erlernte Wissen in den erweiterten Übungen vor Ort an den Rechnerarbeitsplätzen des Lehrstuhls an. The students apply their learned knowledge in hands-on computer exercises on-site at the chair's computer workstations.
7	**Voraussetzungen für die Teilnahme**	Die Auswahl dieses Moduls schließt die Auswahl des Moduls „Parallele Systeme“ aus.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio • Prüfung und erfolgreiche Bearbeitung aller Übungsaufgaben in den erweiterten Übungen (verpflichtend, vor Ort an den Rechnerarbeitsplätzen des Lehrstuhls). • Ein Wechsel der Prüfungsform von einer Klausur zu einer mündlichen Prüfung ist in Ausnahmefällen (siehe § 16 ABMPO/TechFak) auch nach Semesterbeginn noch möglich. In diesem Fall werden die Studierenden spätestens zwei Wochen nach Vorlesungsbeginn informiert. • Prüfungssprache: Deutsch oder Englisch. Die Unterrichts- und Prüfungssprache hängt von den Sprachkenntnissen und Präferenzen der Teilnehmerinnen und Teilnehmer ab und wird dementsprechend innerhalb der ersten zwei Wochen nach Vorlesungsbeginn festgelegt.
11	**Berechnung der Modulnote**	Portfolio (100%) Die Modulnote ergibt sich aus der Prüfungsleistung.
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 90 h Eigenstudium: 135 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch oder Englisch
16	**Literaturhinweise**	Weitere Informationen:

		https://www.cs12.tf.fau.de/lehre/lehrveranstaltungen/vorlesungen/parallele-systeme

1	**Modulbezeichnung** 44120	**Pattern Analysis** (Pattern analysis)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Pattern Analysis (3 SWS) Übung: Pattern Analysis Programming (1 SWS)	3,75 ECTS 1,25 ECTS
3	Lehrende	PD Dr.Ing. Christian Rieß Dalia Rodriguez Salas	

4	**Modulverantwortliche/r**	PD Dr.Ing. Christian Rieß
5	**Inhalt**	This module introduces the design of pattern analysis systems as well as the corresponding fundamental mathematical methods. The topics comprise: • clustering methods: soft and hard clustering • classification and regression trees and forests • parametric and non-parametric density estimation: maximum-likelihood (ML) estimation, maximum-a-posteriori (MAP) estimation, histograms, Parzen estimation, relationship between folded histograms and Parzen estimation, adaptive binning with regression trees • mean shift algorithm: local maximization using gradient ascent for non-parametric probability density functions, application of the mean shift algorithm for clustering, color quantization, object tracking • linear and non-linear manifold learning: curse of dimensionality, various dimensionality reduction methods: principal component analysis (PCA), multidimensional scaling (MDS), isomaps, Laplacian eigenmaps • Gaussian mixture models (GMM) and hidden Markov models (HMM): expectation maximization algorithm, parameter estimation, computation of the optimal sequence of states/ Viterbi algorithm, forward-backward algorithm, scaling • Markov random fields (MRF): definition, probabilities on undirected graphs, clique potentials, Hammersley-Clifford theorem, inference via Gibbs sampling and graph cuts Das Modul führt in das Design von Musteranalysesystemen sowie die zugrundeliegenden mathematischen Methoden ein. Die Vorlesung umfasst im Einzelnen: • Clustering-Methoden: Soft- und Hard-Clustering • Klassifikations- und Regressionsbäume/-wälder • parametrische und nicht-parametrische Dichteschätzung: Verfahren sind ML- und MAP-Schätzung, Histogramme, Parzenschätzung, Zusammenhang gefaltete Histogramme und Parzenschätzung, adaptives Binning mit Regressionsbäumen. • 'Mean Shift'-Algorithmus: lokale Maximierung durch Gradientenaufstieg bei nicht-parametrischen Dichtefunktionen, Anwendungen des 'Mean Shift'-Algorithmus zum Clustering, Farbquantisierung und Objektverfolgung

		• Linear and Non-Linear Manifold Learning: Curse of Dimensionality, Verschiedene Methode zur Dimensionsreduktion: Principal Component Analysis (PCA), Multidimensionsional Scaling (MDS), Isomap, Laplacian Eigenmaps • Gaußsche Mischverteilungsmodelle (GMM) und Hidden-Markov-Modelle (HMM): 'Expectation Maximization'-Algorithmus, Parameterschätzung, Bestimmung der optimalen Zustandsfolge/Viterbi-Algorithmus, Vorwärts-Rückwärts-Algorithmus, Skalierung • Markov-Zufallsfelder: Definition, Wahrscheinlichkeiten auf ungerichteten Graphen, Cliquen-Potenziale, Hammersley-Clifford-Theorem, Inferenz mit Gibbs-Sampling und Graph Cuts
6	**Lernziele und Kompetenzen**	The students • explain the discussed methods for classification, prediction, and analysis of patterns, • compare and analyze methods for manifold learning and select a suited method for a given set of features and a given problem, • compare and analyze methods for probability density estimation and select a suited method for a given set of features and a given problem, • apply non-parametric probability density estimation to pattern analysis problems, • apply dimensionality reduction techniques to high-dimensional feature spaces, • explain statistic modeling of feature sets and sequences of features, • explain statistic modeling of statistical dependencies, • implement presented methods in Python, • supplement autonomously the mathematical foundations of the presented methods by self-guided study of the literature, • discuss the social impact of applications of pattern analysis solutions. Die Studierenden • erläutern die behandelten Methoden zur Klassifikation, Vorhersage und Analyse von Mustern, • vergleichen und analysieren Methoden des Manifold Learning und wählen für eine vorgegebene Fragestellung eine geeignete Methode aus, • vergleichen und analysieren Methoden zur Dichteschätzung und wählen für eine vorgegebene Fragestellung eine geeignete Methode aus, • wenden nicht-parametrische Dichteschätzung auf Probleme der Musteranalyse an, • wenden Dimensionsreduktion bei hochdimensionalen Merkmalsräumen an,

		• erläutern statistische Modellierung von Merkmalsmengen und Merkmalsfolgen, • erklären statistische Modellierung abhängiger Größen, • implementieren vorgestellte Verfahren in Python. • ergänzen eigenständig mathematische Grundlagen der präsentierten Methoden durch selbstbestimmtes Studium der Literatur • diskutieren die gesellschaftlichen Auswirkungen von Anwendungen der Musteranalyse
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Variabel (60 Minuten)
11	**Berechnung der Modulnote**	Variabel (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Begleitende Literatur / Accompanying literature: • C. Bishop: Pattern Recognition and Machine Learning, Springer Verlag, Heidelberg, 2006 • T. Hastie, R. Tibshirani und J. Friedman: The Elements of Statistical Learning, 2nd Edition, Springer Verlag, 2009 • A. Criminisi and J. Shotton: Decision Forests for Computer Vision and Medical Image Analysis, Springer, 2013

1	**Modulbezeichnung** 44130	**Pattern Recognition** (Pattern recognition)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Pattern Recognition Exercises (1 SWS) Vorlesung: Pattern Recognition (3 SWS)	1,25 ECTS 3,75 ECTS
3	Lehrende	Siming Bayer Paul Stöwer Prof. Dr.-Ing. Andreas Maier	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Andreas Maier
5	**Inhalt**	Mathematical foundations of machine learning based on the following classification methods: • Bayesian classifier • Logistic Regression • Naive Bayes classifier • Discriminant Analysis • norms and norm dependent linear regression • Rosenblatt's Perceptron • unconstraint and constraint optimization • Support Vector Machines (SVM) • kernel methods • Expectation Maximization (EM) Algorithm and Gaussian Mixture Models (GMMs) • Independent Component Analysis (ICA) • Model Assessment • AdaBoost Mathematische Grundlagen der maschinellen Klassifikation am Beispiel folgender Klassifikatoren: • Bayes-Klassifikator • Logistische Regression • Naiver Bayes-Klassifikator • Diskriminanzanalyse • Normen und normabhängige Regression • Rosenblatts Perzeptron • Optimierung ohne und mit Nebenbedingungen • Support Vector Maschines (SVM) • Kernelmethoden • Expectation Maximization (EM)-Algorithmus und Gaußsche Mischverteilungen (GMMs) • Analyse durch unabhängige Komponenten • Modellbewertung • AdaBoost
6	**Lernziele und Kompetenzen**	Die Studierenden • verstehen die Struktur von Systemen zur maschinellen Klassifikation einfacher Muster • erläutern die mathematischen Grundlagen ausgewählter maschineller Klassifikatoren

		• wenden Klassifikatoren zur Lösung konkreter Klassifikationsproblem an • beurteilen unterschiedliche Klassifikatoren in Bezug auf ihre Eignung • verstehen in der Programmiersprache Python geschriebene Lösungen von Klassifikationsproblemen und Implementierungen von Klassifikatoren Students • understand the structure of machine learning systems for simple patterns • explain the mathematical foundations of selected machine learning techniques • apply classification techniques in order to solve given classification tasks • evaluate various classifiers with respect to their suitability to solve the given problem • understand solutions of classification problems and implementations of classifiers written in the programming language Python
7	**Voraussetzungen für die Teilnahme**	• Well grounded in probability calculus, linear algebra/matrix calculus • The attendance of our bachelor course 'Introduction to Pattern Recognition' is not required but certainly helpful. • Gute Kenntnisse in Wahrscheinlichkeitsrechnung und Linearer Algebra/Matrizenrechnung • Der Besuch der Bachelor-Vorlesung 'Introduction to Pattern Recognition' ist zwar keine Voraussetzung, aber sicherlich von Vorteil.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Pflichtmodule Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch oder Englisch
16	**Literaturhinweise**	• Richard O. Duda, Peter E. Hart, David G. Stock: Pattern Classification, 2nd edition, John Wiley&Sons, New York, 2001

		• Trevor Hastie, Robert Tobshirani, Jerome Friedman: The Elements of Statistical Learning - Data Mining, Inference, and Prediction, 2nd edition, Springer, New York, 2009 • Christopher M. Bishop: Pattern Recognition and Machine Learning, Springer, New York, 2006

1	**Modulbezeichnung** 399289	**Programmierung und Architekturen von Cluster-Rechnern** (Programming and architecture of computer clusters)	**7,5 ECTS**
2	Lehrveranstaltungen	Zu diesem Modul sind keine Lehrveranstaltungen oder Lehrveranstaltungsgruppen hinterlegt!	
3	Lehrende	Zu diesem Modul sind keine Lehrveranstaltungen und somit auch keine Lehrenden hinterlegt!	

4	**Modulverantwortliche/r**	
5	**Inhalt**	keine Inhaltsbeschreibung hinterlegt!
6	**Lernziele und Kompetenzen**	keine Beschreibung der Lernziele und Kompetenzen hinterlegt!
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	keine Angaben zum Turnus des Angebots hinterlegt!
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: ?? h (keine Angaben zum Arbeitsaufwand in Präsenzzeit hinterlegt) Eigenstudium: ?? h (keine Angaben zum Arbeitsaufwand im Eigenstudium hinterlegt)
14	**Dauer des Moduls**	?? Semester (keine Angaben zur Dauer des Moduls hinterlegt)
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 96316	**Radar, RFID and Wireless Sensor Systems (RWS)** (keine englischsprachige Modulbezeichnung hinterlegt!)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Radar, RFID and Wireless Sensor Systems Exercises (2 SWS) Vorlesung: Radar, RFID and Wireless Sensor Systems (2 SWS)	- 5 ECTS
3	Lehrende	Dr.-Ing. Christian Carlowitz Prof. Dr.-Ing. Martin Vossiek	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Martin Vossiek
5	**Inhalt**	Radar, RFID and wireless sensor and wireless locating systems are essential for automotive advanced driver-assistance systems (ADAS), autonoumous driving and flying, robotics, industrial automation, logistics and novel human machine interfaces. Further key areas include medical electronics, building technology and cyber-physical systems. The module "Radar, RFID and Wireless Sensors" is an introduction into functional principles, building blocks, hardware and signal processing concepts and applications of modern radar, RFID, wireless sensor and real time locating systems. Covered applications include automotive radar, road and air traffic control systems, as well as robotics, industrial automation and medical technology. RWS is an identical replacement of the former module "Drahtlose Sensoren, Radar- und RFID-Systeme DSR.
6	**Lernziele und Kompetenzen**	The students • learn about the setup, function and application of wireless sensors, Radar and RFID-systems • can analyze, discuss and implement basic components and system structures, signal theory, data processing and use cases • can determine the underlying physical limitations and sources of errors • are able to analyze and create system specifications and can compare and rate the usability of wireless esnsors, Radar and RFID-systems • can create and define independently applications and system designs of RWSs
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot der Technischen Fakultät oder der Naturwissenschaftlichen Fakultät Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich

11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Sensors for Ranging and Imaging", Graham Brooker, Scitech Publishing Inc., 2009 Radar mit realer und synthetischer Apertur", H. Klausing, W. Holpp, Oldenbourg, 1999 Praxiswissen Radar und Radarsignalverarbeitung" Albrecht K. Ludloff, 2008 "RFID at ultra and super high frequencies: theory and application Dominique Paret, John Wiley & Sons, 2009. RFID-Handbuch: Grundlagen und praktische Anwendungen von Transpondern, kontaktlosen Chipkarten und NFC", Klaus Finkenzeller, Carl Hanser Verlag, 6. Auflage 2012.

1	**Modulbezeichnung** 798810	**Rechnerarchitektur** (Computer architecture)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Rechnerarchitektur (2 SWS) Übung: Übungen zu Rechnerarchitektur (2 SWS) Übung: Rechnerübungen zu Rechnerarchitektur (2 SWS)	2,5 ECTS 2,5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr.-Ing. Dietmar Fey Tobias Baumeister Sebastian Rachuj Christian Widerspick	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Dietmar Fey
5	**Inhalt**	Die Vorlesung baut auf die in den Grundlagen der Rechnerarchitektur und -organisation vermittelten Inhalte auf und setzt diese mit weiterführenden Themen fort. Es werden zunächst grundlegende fortgeschrittene Techniken bei Pipelineverarbeitung und Cachezugriffen in modernen Prozessoren und Parallelrechnern behandelt. Ferner wird die Architektur von Spezialprozessoren, z.B. DSPs und Embedded Prozessoren behandelt. Es wird aufgezeigt, wie diese Techniken in konkreten Architekturen (Intel Nehalem, GPGPU, Cell BE, TMS320 DSP, Embedded Prozessor ZPU) verwendet werden. Zur Vorlesung wird eine Tafelübung angeboten. Mit erfolgreicher mündlicher Prüfung können 5 ECTS erworben werden. In den Tafelübungen werden die in der Vorlesung vermittelten Techniken durch zu lösende Aufgaben vertieft. In der Rechnerübung soll u.a. ein einfacher Vielkern-Prozessor auf Basis des ZPU-Prozessors mit Simulationswerkzeugen aufgebaut werden. Im Einzelnen werden folgende Themen behandelt: • Organisationsaspekte von CISC und RISC-Prozessoren • Behandlung von Hazards in Pipelines • Fortgeschrittene Techniken der dynamischen Sprungvorhersage • Fortgeschritten Cachetechniken, Cache-Kohärenz • Ausnutzen von Cacheeffekten • Architekturen von Digitalen Signalprozessoren • Architekturen homogener und heterogener Multikern-Prozessoren (Intel Corei7, Nvidia GPUs, RISC-V) • Architektur von Parallelrechnern (Clusterrechner, Superrechner) • Effiziente Hardware-nahe Programmierung von Mulitkern-Prozessoren (OpenMP, SSE, CUDA) • Leistungsmodellierung und -analyse von Multikern-Prozessoren (Roofline-Modell)
6	**Lernziele und Kompetenzen**	Fachkompetenz Wissen

		Lernende können Wissen abrufen und wiedergeben. Sie kennen konkrete Einzelheiten wie Begriffe, Definitionen, Fakten, und Abläufe in einem Prozessor darlegen. Verstehen Lernende können Beispiele für Rechnerarchitekturen anführen, sie sind in der Lage, Schaubilder von Prozessoren zu interpretieren und die Abläufe in eigenen Worten zu beschreiben. Anwenden Lernende können beim Erstellen eigener Programme durch Transfer des Wissens über Interna von Prozesorarchitekturen Optimierungen hinsichtlich des Laufzeitverhaltens vornehmen. Analysieren Lernende können zwischen verschiedenen Varianten von Lösungen einer Prozessorarchitektur klassifizieren, die Gründe für durchgeführte Entwurfsentscheidungen erschließen, Unterscheide gegenüberstellen und gegeneinander bewerten. Lern- bzw. Methodenkompetenz Lernende erwerben die Fähigkeit selbstständig Testprogramme zum Bewerten der Leistungsfähigkeit eines Prozessors zu erstellen.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• Patterson/Hennessy: Computer Organization und Design • Hennessy/Patterson: Computer Architecture - A Quantitative Approach • Stallings: Computer Organization and Architecture

		• Märtin: Rechnerarchitekturen

1	**Modulbezeichnung** 333815	**Rechnerarchitektur (Vorlesung mit Übung und Rechnerübung)** (Computer architecture)	**7,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Rechnerarchitektur (2 SWS) Übung: Übungen zu Rechnerarchitektur (2 SWS) Übung: Rechnerübungen zu Rechnerarchitektur (2 SWS)	2,5 ECTS 2,5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr.-Ing. Dietmar Fey Tobias Baumeister Sebastian Rachuj Christian Widerspick	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Dietmar Fey
5	**Inhalt**	Die Vorlesung baut auf die in den Grundlagen der Rechnerarchitektur und -organisation vermittelten Inhalte auf und setzt diese mit weiterführenden Themen fort. Es werden zunächst grundlegende fortgeschrittene Techniken bei Pipelineverarbeitung und Cachezugriffen in modernen Prozessoren und Parallelrechnern behandelt. Ferner wird die Architektur von Spezialprozessoren, z.B. DSPs und Embedded Prozessoren behandelt. Es wird aufgezeigt, wie diese Techniken in konkreten Architekturen (Intel Nehalem, GPGPU, Cell BE, TMS320 DSP, Embedded Prozessor ZPU) verwendet werden. Zur Vorlesung werden eine Tafel- und eine Rechnerübung angeboten. Die Rechnerübung erfodert 11 erfolgreich abgeschlossene Übungsaufgaben, diese gehen mit 10% in die Modulgesamtnote ein. Die verbleibenden 90% werden durch die mündliche Prüfung bestimmt. Insgesamt werden 7,5 ECTS erworben. In den Tafelübungen werden die in der Vorlesung vermittelten Techniken durch zu lösende Aufgaben vertieft. In der Rechnerübung soll u.a. ein einfacher Vielkern-Prozessor auf Basis des ZPU-Prozessors mit Simulationswerkzeugen aufgebaut werden. Im Einzelnen werden folgende Themen behandelt: • Organisationsaspekte von CISC und RISC-Prozessoren • Behandlung von Hazards in Pipelines • Fortgeschrittene Techniken der dynamischen Sprungvorhersage • Fortgeschritten Cachetechniken, Cache-Kohärenz • Ausnutzen von Cacheeffekten • Architekturen von Digitalen Signalprozessoren • Architekturen homogener und heterogener Multikern-Prozessoren (Intel Corei7, Nvidia GPUs, RISC-V) • Architektur von Parallelrechnern (Clusterrechner, Superrechner) • Effiziente Hardware-nahe Programmierung von Mulitkern-Prozessoren (OpenMP, SSE, CUDA) • Leistungsmodellierung und -analyse von Multikern-Prozessoren (Roofline-Modell)

6	**Lernziele und Kompetenzen**	Fachkompetenz Wissen Lernende können Wissen abrufen und wiedergeben. Sie können konkrete Einzelheiten wie Begriffe, Definitionen, Fakten, und Abläufe in einem Prozessor darlegen. Verstehen Lernende können Beispiele für Rechnerarchitekturen anführen, sie sind in der Lage, Schaubilder von Prozessoren zu interpretieren und die Abläufe in eigenen Worten zu beschreiben. Anwenden Lernende können beim Erstellen eigener Programme durch Transfer des Wissens über Interna von Prozesorarchitekturen Optimierungen hinsichtlich des Laufzeitverhaltens vornehmen. Analysieren Lernende können zwischen verschiedenen Varianten von Lösungen einer Prozessorarchitektur klassifizieren, die Gründe für durchgeführte Entwurfsentscheidungen erschließen, Unterscheide gegenüberstellen und gegeneinander bewerten. Lern- bzw. Methodenkompetenz Lernende erwerben die Fähigkeit selbstständig Testprogramme zum Bewerten der Leistungsfähigkeit eines Prozessors zu erstellen.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 90 h Eigenstudium: 135 h
14	**Dauer des Moduls**	1 Semester

15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• Patterson/Hennessy: Computer Organization und Design • Hennessy/Patterson: Computer Architecture - A Quantitative Approach • Stallings: Computer Organization and Architecture • Märtin: Rechnerarchitekturen

1	**Modulbezeichnung** 43195	**Reconfigurable Computing** (Reconfigurable computing)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Reconfigurable Computing (2 SWS) Übung: Exercises to Reconfigurable Computing (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr.-Ing. Jürgen Teich Tobias Hahn Mauro Martin Letras Luna Pierre-Louis Sixdenier	

4	**Modulverantwortliche/r**	Joachim Falk Prof. Dr.-Ing. Jürgen Teich
5	**Inhalt**	**Content:** Reconfigurable (adaptive) computing is a novel yet important research field investigating the capability of hardware to adapt to changing computational requirements such as emerging standards, late design changes, and even to changing processing requirements arising at run-time. Reconfigurable computing thus benefits from a) the programmability of software similar to the Von Neumann computer and b) the speed and efficiency of parallel hardware execution. The purpose of the course reconfigurable computing is to instruct students about the possibilities and rapidly growing interest in adaptive hardware and corresponding design techniques by providing them the necessary knowledge for understanding and designing reconfigurable hardware systems and studying applications benefiting from dynamic hardware reconfiguration. After a general introduction about benefits and application ranges of reconfigurable (adaptive) computing in contrast to general-purpose and application-specific computing, the following topics will be covered: • Reconfigurable computing systems: Introduction of available technology including fine grained look up table (LUT-) based reconfigurable systems such as field programmable gate arrays (FPGA) as well as newest coarse grained architectures and technology. • Design and implementation: Algorithms and steps (design entry, functional simulation, logic synthesis, technology mapping, place and route, bit stream generation) to implement (map) algorithms to FPGAs. The main focus lies on logic synthesis algorithms for FPGAs, in particular LUT technology mapping. • Temporal partitioning: techniques to reconfigure systems over time. Covered are the problems of mapping large circuits which do not fit one single device. Several temporal partitioning techniques are studied and compared. • Temporal placement: Techniques and algorithms to exploit the possibility of partial and dynamic (run-time) hardware

		reconfiguration. Here, OS-like services are needed that optimize the allocation and scheduling of modules at run-time. • On-line communication: Modules dynamically placed at run-time on a given device need to communicate as well as transport data off-chip. State-of-the-art techniques are introduced how modules can communicate data at run-time including bus-oriented as well as network-on-a-chip (NoC) approaches. • Designing reconfigurable applications on Xilinx Virtex FPGAs: In this part, the generation of partial bitstreams for components to be placed at run-time on Xilinx FPGAs is introduced and discussed including newest available tool flows. • Applications: This section presents applications benefiting from dynamic hardware reconfiguration. It covers the use of reconfigurable systems including rapid prototyping, reconfigurable supercomputers, reconfigurable massively parallel computers and studies important application domains such as distributed arithmetic, signal processing, network packet processing, control design, and cryptography.
6	**Lernziele und Kompetenzen**	**Learning objectives and competencies:** Domain-specific knowledge • The students know to exploit run-time reconfigurable design methodologies for adaptive applications. Domain-specific comprehension • The students understand the mapping steps and optimization algorithms. • The students classify different types and kinds of reconfigurable hardware technologies available today. • The students clarify pros and cons of reconfigurable computing technology. • The students summarize applications benefiting from reconfigurable computing. • The students describe the design of circuits and systems-on-a-chip (SoC) on FPGAs.
7	**Voraussetzungen für die Teilnahme**	Selection of this module prohibits the selection of the modules "Reconfigurable Computing (Lecture with Exercises)" or "Reconfigurable Computing (Lecture with Extended Exercises)" by the student.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich (30 Minuten) Oral examination (Duration: 30 min).

11	**Berechnung der Modulnote**	mündlich (100%) The oral examination determines the final grade of the module.
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Further reading material: • The Hamburg VHDL Archive (see Documentation link for free books) http://tams-www.informatik.uni-hamburg.de/research/vlsi/vhdl/index.php • Interactive VHDL Tutorial with 150 examples from ALDEC http://www.aldec.com/downloads/ • Easy FPGA tutorials, projects, and boards http://www.fpga4fun.com • Xilinx WebPack ISE and Modelsim MXE (free FPGA synthesis tool and free VHDL simulator) http://www.xilinx.com/ise/logic_design_prod/webpack.htm • Symphony EDA free VHDL simulator (select FREE Edition license) http://www.symphonyeda.com/products.htm • Icarus open-source Verilog simulator http://www.icarus.com/eda/verilog/ Further information: https://www.cs12.tf.fau.de/lehre/lehrveranstaltungen/vorlesungen/reconfigurable-computing/

1	**Modulbezeichnung** 110334	**Reconfigurable Computing (Lecture with Extended Exercises)** (Reconfigurable computing (lecture with extended exercises))	**7,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Reconfigurable Computing (2 SWS) Übung: Extended Exercises to Reconfigurable Computing (2 SWS) Übung: Exercises to Reconfigurable Computing (2 SWS)	2,5 ECTS 2,5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr.-Ing. Jürgen Teich Pierre-Louis Sixdenier Mauro Martin Letras Luna Tobias Hahn	

4	**Modulverantwortliche/r**	Joachim Falk Prof. Dr.-Ing. Jürgen Teich
5	**Inhalt**	**Content:** Reconfigurable (adaptive) computing is a novel yet important research field investigating the capability of hardware to adapt to changing computational requirements such as emerging standards, late design changes, and even to changing processing requirements arising at run-time. Reconfigurable computing thus benefits from a) the programmability of software similar to the Von Neumann computer and b) the speed and efficiency of parallel hardware execution. The purpose of the course reconfigurable computing is to instruct students about the possibilities and rapidly growing interest in adaptive hardware and corresponding design techniques by providing them the necessary knowledge for understanding and designing reconfigurable hardware systems and studying applications benefiting from dynamic hardware reconfiguration. After a general introduction about benefits and application ranges of reconfigurable (adaptive) computing in contrast to general-purpose and application-specific computing, the following topics will be covered: • Reconfigurable computing systems: Introduction of available technology including fine grained look up table (LUT-) based reconfigurable systems such as field programmable gate arrays (FPGA) as well as newest coarse grained architectures and technology. • Design and implementation: Algorithms and steps (design entry, functional simulation, logic synthesis, technology mapping, place and route, bit stream generation) to implement (map) algorithms to FPGAs. The main focus lies on logic synthesis algorithms for FPGAs, in particular LUT technology mapping.

		• Temporal partitioning: techniques to reconfigure systems over time. Covered are the problems of mapping large circuits which do not fit one single device. Several temporal partitioning techniques are studied and compared. • Temporal placement: Techniques and algorithms to exploit the possibility of partial and dynamic (run-time) hardware reconfiguration. Here, OS-like services are needed that optimize the allocation and scheduling of modules at run-time. • On-line communication: Modules dynamically placed at run-time on a given device need to communicate as well as transport data off-chip. State-of-the-art techniques are introduced how modules can communicate data at run-time including bus-oriented as well as network-on-a-chip (NoC) approaches. • Designing reconfigurable applications on Xilinx Virtex FPGAs: In this part, the generation of partial bitstreams for components to be placed at run-time on Xilinx FPGAs is introduced and discussed including newest available tool flows. • Applications: This section presents applications benefiting from dynamic hardware reconfiguration. It covers the use of reconfigurable systems including rapid prototyping, reconfigurable supercomputers, reconfigurable massively parallel computers and studies important application domains such as distributed arithmetic, signal processing, network packet processing, control design, and cryptography.
6	**Lernziele und Kompetenzen**	**Learning objectives and competencies:** Domain-specific knowledge • The students know to exploit run-time reconfigurable design methodologies for adaptive applications. Domain-specific comprehension • The students understand the mapping steps, and optimization algorithms. • The students classify different types and kinds of reconfigurable hardware technologies available today. • The students clarify pros and cons of reconfigurable computing technology. • The students summarize applications benefiting from reconfigurable computing. Domain-specific practice • The students apply design tools for implementation of circuits and systems-on-a-chip (SoC) on FPGAs during practical training.

		Social competency • The students perform group work in small teams during practical training.
7	**Voraussetzungen für die Teilnahme**	Selection of this module prohibits the selection of the modules "Reconfigurable Computing" or "Reconfigurable Computing (Lecture with Exercises)" by the student.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio Oral examination (Duration: 30 min) and successful completion of all tasks of the extended exercises (mandatory, at the workstations residing in our lab at the chair).
11	**Berechnung der Modulnote**	Portfolio (100%) The oral examination determines the final grade of the module.
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 90 h Eigenstudium: 135 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Further reading material: • The Hamburg VHDL Archive (see Documentation link for free books) http://tams-www.informatik.uni-hamburg.de/research/vlsi/vhdl/index.php • Interactive VHDL Tutorial with 150 examples from ALDEC http://www.aldec.com/downloads/ • Easy FPGA tutorials, projects, and boards http://www.fpga4fun.com • Xilinx WebPack ISE and Modelsim MXE (free FPGA synthesis tool and free VHDL simulator) http://www.xilinx.com/ise/logic_design_prod/webpack.htm • Symphony EDA free VHDL simulator (select FREE Edition license) http://www.symphonyeda.com/products.htm • Icarus open-source Verilog simulator http://www.icarus.com/eda/verilog/ Further information: https://www.cs12.tf.fau.de/lehre/lehrveranstaltungen/vorlesungen/reconfigurable-computing/

1	**Modulbezeichnung** 92650	**Regelungstechnik A (Grundlagen)** (Control engineering A (Foundations))	**5 ECTS**
2	Lehrveranstaltungen	Übung: Regelungstechnik A (Grundlagen) - Übungen (2 SWS) Vorlesung: Regelungstechnik A (Grundlagen) (2 SWS)	- -
3	Lehrende	Prof. Dr.-Ing. Knut Graichen	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Knut Graichen
5	**Inhalt**	Die Vorlesung behandelt die Grundlagen der Regelungstechnik und befähigt zur Beschreibung und Untersuchung linearer Systeme und zum Entwurf einfacher und mehrschleifiger Regler im Frequenzbereich. Die Inhalte der Vorlesung sind: • Gegenstand und Zielstellung der Regelungstechnik • Modellbildung der Strecke im Zeit und Frequenzbereich und Darstellung als Strukturbild • Analyse des Streckenverhaltens linearer Eingrößensysteme anhand von Übertragungsfunktion und Frequenzgang • Auslegung einschleifiger Regelkreise • Erweitere Regelkreisstrukturen
6	**Lernziele und Kompetenzen**	Die Studierenden können • Gegenstand und Zielstellung der Regelungstechnik erläutern. • Problemstellungen als Steuerungs- und Regelungsaufgabe identifizieren. • das Streckenverhalten durch ein mathematisches Modell in Form des Strukturbilds beschreiben. • eine Modellvereinfachung durch Linearisierung und Strukturbildumformung durchführen. • aus Übertragungsfunktion und Frequenzgang das qualitative Streckenverhalten ermitteln. • zu einem Frequenzgang Ortskurve und Bode-Diagramm angeben. • den Aufbau einer Zwei-Freiheitsgrade-Regelung angeben und die Zweckbestimmung von Vorsteuerung und Regelung erläutern. • Sollverläufe auf Zulässigkeit überprüfen und realisierbare Vorsteuerungen entwerfen. • die Regelkreis-Stabilität definieren und mit dem Nyquist-Kriterium untersuchen. • entscheiden, wann welcher Reglertyp in Frage kommt und nach welchen Gesichtspunkten dessen Parameter zu wählen sind. • für lineare Eingrößensysteme einen geeigneten Regler entwerfen. • ergänzende Maßnahmen zur Störverhaltensverbesserung beschreiben und zur Anwendung bringen. • die Vorlesungsinhalte auf verwandte Problemstellungen übertragen und sich weiterführende Frequenzbereichsmethoden der Regelungstechnik selbständig erschließen.

7	**Voraussetzungen für die Teilnahme**	Systemtheorie linearer zeitkontinuierlicher Systeme (inkl. Laplace-Transformation)
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• O. Föllinger. Regelungstechnik: Einführung in die Methoden und ihre Anwendung, 12. Auflage, VDE-Verlag, 2016 • M. Horn, N. Dourdoumas. Regelungstechnik, Pearson Studium, 2004 • W. Leonhard. Einführung in die Regelungtechnik, 4. Auflage, Vieweg, 1987 • J. Lunze. Regelungstechnik 1: Systemtheoretische Grundlagen, Analyse und Entwurf einschleifiger Regelungen, 12. Auflage, Springer, 2020 • R. Unbehauen. Regelungtechnik 1, 12. Auflage, 2002 • G. Ludyk. Theoretische Regelungstechnik 1 und 2, Springer, 1995

1	**Modulbezeichnung** 43460	**Satellitenkommunikation** (Satellite communication)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übung Satellitenkommunikation (2 SWS) Vorlesung: Satellitenkommunikation (2 SWS)	- 5 ECTS
3	Lehrende	Marcelo Michael Dr. Christian Rohde	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Albert Heuberger
5	**Inhalt**	Nach einem historischen Rückblick zur Entwicklung der Satellitenkommunikation werden die einzelnen Komponenten eines typischen Gesamtsystems (Boden- und Raumsegment) näher betrachtet. Hierzu zählt der prinzipielle Aufbau von Trägerraketen, von Satelliten (Satellitenplattformen, Subsysteme, Nutzlasten), die meist genutzten Umlaufbahnen und die verschiedenen Kommunikationsverbindungen (Uplink, Downlink, Inter-Satellite-Link). Die Besonderheiten der Signalausbreitung und -übertragung über große Entfernungen zwischen Bodenstationen und Satelliten werden erklärt und mit Beispielen ergänzt. Dabei wird insbesondere eingegangen auf verwendete Frequenzen, Signaldispersion und -dämpfung, atmosphärische Effekte sowie Störeinflüsse der Weltraumumgebung. Die Architektur transparenter und regenerativer Kommunikationseinheiten wird ausführlich an Beispielen kommerziell verfügbarer Transponder und Onboard-Prozessoren erklärt. Die Prinzipien moderner, standardisierter Verfahren zur Signalaufbereitung und Übertragung von Video-/Bild und Audiosignalen über Satellit (z.B. MPEG, H.264/265, DVB-S/-S2/-S2X) werden erläutert und diskutiert. Dies umfasst Verfahren zur Quellencodierung, Kanalcodierung und Modulation, Kanalzugriff und -diversität. Außerdem wird auf die im Orbit und im kommerziellen Einsatz befindlichen Kommunikationssatelliten und der damit verbundenen großen Dienstevielfalt eingegangen wie z.B. bei TV- und Breitbandversorgung sowie in Mobilkommunikationssystemen. Abschließend werden einige Herausforderungen und Forschungsansätze im Zusammenhang mit den neuen Megakonstellationen und Next Generation High Throughput Satellites (HTS) für zukünftige Satellitensysteme vorgestellt. Die in der Vorlesung behandelten physikalischen, elektro- und nachrichtentechnischen Zusammenhänge werden in den ergänzenden Übungen mit Rechenbeispielen vertieft. Gliederung der Vorlesung: **1. Einführung:**

Überblick über die Hauptkomponenten, Satelliten, Anwendungen und Dienste, sowie Orbits, Aufgaben und Frequenzen der Satellitennetzwerke

2. Historie der Satellitenkommunikation:

Wichtige Meilensteine, Entwicklung in Europa und Deutschland

3. Orbits und Konstellationen:

Keplersche Gesetze, Beschreibung von Orbits, verwendete Umlaufbahnen, Bodenspuren, erreichbare Abdeckung

4. Trägersysteme:

Trägerraketen, Entwicklung, Anbietermarkt, Nutzlastfähigkeit, Startplätze, Startverlauf

5. Satellitenaufbau:

Auswahl aktueller Satellitenplattformen, Satellitenaufbau, Plattformkomponenten, Montageschritte und Tests

6. Satellitennutzlast (Payload):

Komponenten, Industrielle Beispiele, Aufbau und Aufgaben der Payload, Transponderarchitekturen, Antennen

7. Signalausbreitung und Leistungsbilanz:

Signalausbreitung, Freiraumverluste, Signaldämpfung, Rauschen, Signal-Rausch-Verhältnis, Linkbudget

8. Weltraumumgebung: Weltraumumgebungsbedingungen, Einflüsse auf den Satelliten und die Elektronik der Nutzlast

9. Quellencodierung:

Audio-, Bild- und Videokompression das Content des Satellitenfernsehens

10. Signalmodulation und Kanalcodierung:

Signalkonstellationen, Modulation und Codes zur Fehlerkorrektur

11. Diversitäts- und Zugriffsverfahren:

Medium Access, Duplextechniken, Multiplexmethoden, Diversitätstechniken

12. Moderne Satellitenkommunikationssysteme:

Rundfunksysteme wie Sirius XM Satellite Radio, zellulare Internetversorgung mittels Satellitenkommunikation

13. Neueste Themen aus Forschung und Entwicklung

SatKom auf StudOn: http://www.studon.uni-erlangen.de/crs117969.html

After a historical retrospective about the developments in satellite communication, the core components of a typical satellite system (ground- and space-segment) are introduced. The principles and architectures of rockets/ carriers, satellites (platform, subsystems, payload), used orbits, and the various communication links (uplink, downlink, inter-satellite-link) are shown. The special features and properties of signal transmission over such large distances are explained and stuffed with examples. In particular, more details are provided on the used frequencies, signal dispersion and attenuation, atmospheric effects as well as impairments due to space environment.

The architecture of transparent and regenerative communication payloads are described in detail, accompanied by corresponding examples of commercially used transponders and onboard-processors and their technology.

The principles of modern standardized methods for signal transmission and preparation of video-/image- and audio-signals via satellite, e.g., MPEG, H.264/265, DVB-S/-S2/-S2X, are illustrated and discussed. This includes methods for efficient source coding, channel coding and modulation, channel access and diversity schemes.

Furthermore, the currently available communication satellites in orbit and the related variety of commercial services are introduced like, e.g., TV- and broadband services as well as mobile communication services and systems. Based on that, a few challenges and perspectives for research and development for future satellite systems are highlighted with respect to the upcoming new mega constellations and next generation high throughput satellites (HTS).

		The physical, electro-technical and communications concepts and schemes shown in the lectures are complemented by tutorials with sample calculations. Table of contents: **1. Introduction:** Overview of main components, satellites, applications and services, orbits, tasks, frequencies, satellite networks **2. History of satellite communications:** Major milestones, development in Europe and Germany **3. Orbits and constellations:** Kepler's laws, description of orbits, orbits used, ground tracks, achievable coverage **4. Launcher systems:** Launch vehicles, providers, payload capabilities, launch sites, launch history **5. Satellite structure:** Selection of current satellite platforms, satellite structure, platform components, assembly steps and tests **6. Payload:** Components, structure and tasks of payload, transponder architecture, antennas **7. Signal propagation and link budget:** Signal propagation, free space losses, signal attenuation, noise, signal to noise ratio, link budget **8. Space environment:** Space environmental conditions, influences on the satellites and payload electronics **9. Source coding:** Audio, image and video compression - the satellite TV broadcasting content **10. Signal modulation and channel coding:** Signal constellations, modulation and error correction coding **11. Diversity and access schemes:** Medium access, duplex methods, multiplex methods, diversity techniques **12. Modern satellite communications systems:** Broadcasting systems like Sirius XM Satellite Radio, satellite cellular broadband communication **13. Latest topics in research and development**
6	**Lernziele und Kompetenzen**	• Die Studierenden bekommen einen guten Überblick über alle Aspekte der Satellitenkommunikation inklusive Historie.

		• Die Studierenden lernen die weltweit führenden oder in Europa ansässigen Firmen und Organisationen kennen, die in den Bereichen Satellitenbau und -betrieb, Satellitendienste bzw. -anwendungen, sowie Forschung und Entwicklung tätig sind. • Die Studierenden können die Herausforderungen der Weltraumumgebung sowie Vor- und Nachteile verschiedener Orbits einschätzen und wichtige Kenngröße berechnen • Die Studierenden lernen die Signalverarbeitungsschritte im Sender, Satelliten und Empfänger kennen - von der Audio/Video-Quelle über Link-Budget-Berechnungen bis zur Datensenke. • Die Studierenden lernen den Aufbau und wichtige Kenngrößen von Satelliten, Konstellationen und Launchern kennen und dabei verwendete Konzepte zu unterscheiden und zu klassifizieren bzgl. deren Vor- und Nachteilen.
7	**Voraussetzungen für die Teilnahme**	Keine formalen Voraussetzungen
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich (91 Minuten)
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Skriptum zur Lehrveranstaltung

1	**Modulbezeichnung** 96410	**Schaltungen und Systeme der Übertragungstechnik** (Schaltungen und Systeme der Übertragungstechnik)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übungen zu Schaltungen und Systeme der Übertragungstechnik (2 SWS) Vorlesung: Schaltungen und Systeme der Übertragungstechnik (2 SWS)	- 5 ECTS
3	Lehrende	Victor Shatov Prof. Dr.-Ing. Norman Franchi Maximilian Lübke Maximilian Lübke	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Norman Franchi
5	**Inhalt**	Im Rahmen dieses Modules werden die Grundlagen und technische Ausführung Übertragungstechniken vermittelt. Fokus liegt dabei auf dem Automotivebereich. Elektrofahrzeuge werden nicht nur die heute bereits in der Oberklasse verfügbaren Fahrassistenzsysteme nutzen sondern weitere E-Mobility spezifische Anwendung insbesondere zur Energie- und Reichweitoptimierung. Drahtlose Kommunikation zwischen Fahrzeug und Ladeeinrichtungen, zwischen Fahrzeugen untereinander, genaue Ortung und Streckenprognose sowie autonomes energiesparendes Fahren mit Radar-Abstandsreglung spielen hier eine wichtige Rolle. In diesem Modul werden diese modernen Entwicklungen adressiert und die dafür notwendigen Grundlagen erarbeitet. Grundlagen: • Funkkanaleigenschaften • Modellierung • Modulation, Codierung, Vielfachzugriff Fahrzeugkommunikationssysteme: • Übertragungssysteme für die Fahrassistenz • Car-to-Car und Car-to-X-Kommunikation • Breitbandige In-Car-Datenübertragung Fahrzeugsensorik: • Fahrzeugortung (lokal und global) • Automobilradar und Umfeldüberwachung • Sensorische Erfassung von Bioparametern im Fahrzeug
6	**Lernziele und Kompetenzen**	Die Studierenden sind nach der erfolgreichen Teilnahme am Modul in der Lage: Funkkanaleigenschaften und Modelle für spezifische Anwendungs- und Betriebsszenarien anzuwenden

		Modulationstechniken zu erläutern und zu analysieren Moderne Codierungs- und Vielfachzugriffstechniken zu erläutern Architekturen und Anwendungen von Fahrzeugkommunikationssystemen zu erläutern und zu analysieren Architekturen und Anwendungen von Fahrzeugsensoriksystemen zu erläutern und zu analysieren
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 43722	**Scientific Visualization** (keine englischsprachige Modulbezeichnung hinterlegt!)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Tutorials to Scientific Visualization (2 SWS) Vorlesung: Scientific Visualization (2 SWS)	- 5 ECTS
3	Lehrende	Prof. Dr.-Ing. Tobias Günther Xingze Tian	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Tobias Günther
5	**Inhalt**	The amount of data, generated in the pursuit of scientific discovery, keeps rapidly increasing across all major scientific disciplines. How can we make sense of large, time-dependent, high-dimensional and multi-variate data? This lecture provides an introduction into scientific visualization. Throughout the course, we cover the fundamental perception basics needed to convey information accurately. After categorizing different data types based on their dimensionality, we dive deeper into specific techniques for scalar, vector and tensor valued data. The lecture covers the following topics: • visualization design basics (data abstraction, visual encoding of information), • a review of scalar and vector calculus (differential properties, extremal and critical points), • data structures and data acquisition techniques (grids, interpolation, and differentiation), • indirect volume visualization (marching cubes and contour trees), • direct volume visualization (ray marching and Monte Carlo rendering), • elementary and line-based flow visualization (numerical integration, seeding, rendering), • surface-based flow visualization (integration, selection, rendering), • topology-based flow visualization (topological skeleton, bifurcations, feature flow fields), • feature-based flow visualization (vortices, material boundaries, Lagrangian coherent structures), • advanced methods (tensor visualization, uncertainty, ensembles)
6	**Lernziele und Kompetenzen**	Students are able to: • use perception basics to select appropriate visualization methods • classify data and select appropriate visualization techniques • calculate differential properties of scalar and vector fields • identify features in scalar and vector-valued data • implement numerical extraction algorithms • learn the advantages and disadvantages of common visualization techniques

7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot der Technischen Fakultät oder der Naturwissenschaftlichen Fakultät Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Variabel
11	**Berechnung der Modulnote**	Variabel (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 43960	**Self-Organized Networks** (keine englischsprachige Modulbezeichnung hinterlegt!)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Self-organized networks (Ex) (2 SWS) Vorlesung: Self-organized networks (2 SWS)	- -
3	Lehrende	Dr.-Ing. Mehdi Harounabadi	

4	**Modulverantwortliche/r**	Dr.-Ing. Mehdi Harounabadi
5	**Inhalt**	keine Inhaltsbeschreibung hinterlegt!
6	**Lernziele und Kompetenzen**	keine Beschreibung der Lernziele und Kompetenzen hinterlegt!
7	**Voraussetzungen für die Teilnahme**	Self-organized networks (2,5 ECTS) Self-organization refers to processes in physical and biological systems which form patterns by interactions of internal parts of the system and/or their interactions with the environment without any form of external interventions. The resulting organization of systems in this case is fully decentralized and distributed over their parts. There are numerous examples of self-organizing systems e.g. flock of birds, neural learning, evolutionary computation, swarm of robots, and self-organizing networks. The principles of self-organization are applied in different communication technologies such as wired and wireless networks. Transport Control Protocol (TCP), Carrier Sense Multiple Access (CSMA), Adaptive Modulation and Coding (AMC) are some examples of self-organized protocols. Besides, Self-Organizing Network (SON) has been defined by 3GPP as the standardization organization of cellular networks. SON refers to the functionality of cellular systems (e.g. 4G and 5G) which enables the system to perform self-configuration, self-optimization and self-healing. Artificial intelligence and mostly machine learning algorithms play an important role in the implementation of SON. This course aims to give you an insight to the self-organization systems and specially self-organizing networks and illustrate the importance of AI and ML in their implementation. The content of this course includes: • o Self-organizing features and protocols in communication networks o In wired networks o In wireless networks § IEEE 802.11 § Ad hoc and sensor networks § UAV networks o Self-Organizing Networks (SON) in 4G and 5G § Self-configuration § Self-optimization § Self-healing Methods for implementation of self-organizing systems o Bio-inspired methods o Artificial intelligence The final exam is in a written form where students should answer to questions based on the lectures, slides and their understating and analysis from the content of the course.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot der Technischen Fakultät oder der Naturwissenschaftlichen Fakultät Master of Science Information and Communication Technology 20191

		Wahlpflichtmodul aus INF im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot der Technischen Fakultät oder der Naturwissenschaftlichen Fakultät Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Variabel
11	**Berechnung der Modulnote**	Variabel (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Literature review on self-organizing networks (2,5 ECTS) The exercise of this course includes a literature review research project where students work individually on a relevant topic. The steps to accomplish the research project are as follows: A. Select a topic relevant to self-organizing networks and register it by email B. Search for the relevant papers and make a list of papers C. Study the papers and prepare a summary D. Present the outcomes

1	**Modulbezeichnung** 250058	**Signalanalyse** (Signal analysis)	**2,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Signalanalyse (2 SWS)	-
3	Lehrende	Dr.-Ing. Heinrich Löllmann	

4	**Modulverantwortliche/r**	Dr.-Ing. Heinrich Löllmann
5	**Inhalt**	Es werden im Rahmen dieser Vorlesung unterschiedliche Verfahren zur Analyse digitaler Signale, sowie deren Anwendungsmöglichkeiten behandelt. Die folgenden Konzepte werden dabei insbesondere behandelt: • Fourieranalyse von Signalen • Signalanalyse mittels Zeit-Frequenz-Transformationen • Parametrische und nichtparametrische Signalanalyse • Verfahren zur Frequenzschätzung • Räumliche Signalanalyse • Filterbänke und Wavelets. In this course, different approaches for the analysis of digital signals and their applications are treated, which comprises the following topics: • Fourier analysis of signals • Signal analysis by means of time-frequency transformations • Parametric and non-parametric signal analysis • Frequency estimation • Spatial signal analysis • Filter-banks and wavelets.
6	**Lernziele und Kompetenzen**	Die Studierenden • beschreiben, welche Methoden der Signalanalyse für unterschiedlichen Arten von Signalen angewendet werden • beschreiben grundlegende Methoden der spektralen Signalanalyse • erläutern wodurch die spektrale und zeitliche Auflösung bei der Spektralanalyse von Signalen begrenzt wird • beschreiben die Konzepte sowie die Vor- und Nachteile der parametrischen und nichtparametrischen Signalanalyse • erklären unterschiedliche Verfahren der Zeit-Frequenz-Analyse • stellen die Analyse von Signalen mittels Filterbänke und Wavelets dar • können Verfahren zur Frequenzschätzung erläutern • formulieren Verfahren zur Analyse räumlicher Signale. The students • describe which methods for signal analysis can be applied for different types of signals • describe fundamental approaches for spectral signal analysis • explain the limiting factors for the time and frequency resolution for the spectral analysis of signals

		• describe concepts as well as the pros and cons of parametric and non-parametric signal analysis • explain different approaches for time-frequency analysis • describe the analysis of signals by means of filter-banks and wavelets • explain methods for frequency estimation • formulate approaches for spatial signal analysis.
7	**Voraussetzungen für die Teilnahme**	Fundierte Kenntnisse in digitaler Signalverarbeitung. Requirements Solid knowledge in digital signal processing
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich Mündliche Prüfung mit einer Dauer von 30 min. Oral examination of 30 min duration.
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	P. Stoica und R. Moses: "Spectral Analysis of Signals", Pearson Prentice Hall, 2005

1	**Modulbezeichnung** 97090	**Simulation und Modellierung I** (Simulation and modelling I)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Exercises to Simulation and Modeling 1 (2 SWS) Vorlesung: Simulation and Modeling 1 (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Lisa Maile Jonathan Fellerer Prof. Dr. Reinhard German	

4	**Modulverantwortliche/r**	Prof. Dr. Reinhard German
5	**Inhalt**	Das Modul vermittelt die Grundlagen der diskreten Ereignissimulation und beinhaltet • diskrete Simulation • analytische Modellierung (z.B. Warteschlangen) • Eingabemodellierung (z.B. Fitting-Verfahren) • Zufallszahlenerzeugung • statistische Ausgabeanalyse • Modellierungsparadigmen (u.a. Ereignis-/Prozessorientierung, Warteschlangen, Automaten, Petri-Netze, UML, graphische Bausteine) • kontinuierliche und hybride Simulation • Simulationssoftware • Fallstudien Content: Overview of the various kinds of simulation • discrete simulation (computational concepts, simulation of queuing systems, simulation in Java, professional simulation tools) • required probability concepts and statistics, modeling paradigms (e.g., event/process oriented, queuing systems, Petri nets, UML statecharts) • input modeling (selecting input probability distributions) • random number generation (linear congruential generators and variants, generating random variates) • output analysis (warm-up period detection, independent replications, result presentation) • continuous and hybrid simulation (differential equations, numerical solution, hybrid statecharts) • simulation software, case studies, parallel and distributed simulation.
6	**Lernziele und Kompetenzen**	Die Studierenden • erwerben Kenntnisse über Verfahren und Realisierungsmöglichkeiten der diskreten Simulation mit Ausblick auf andere Simulationsarten

		• erwerben Kenntnisse über statistische Aspekte der Simulation, die für die Anwendung wichtig sind • wenden statistische Methoden zur Analyse und Bewertung von Eingabe- sowie Ausgabedaten an • erwerben praktische Erfahrung mit kommerziellen Simulationswerkzeugen • erwerben Erfahrungen bei der Simulation in verschiedenen Anwendungsbereichen (u.a. Rechnernetze, Fertigungssysteme, Materialflusssysteme) • entwickeln eigenständig anhand von Beispielaufgaben Simulationsmodelle unter Verwendung verschiedener Modellierungsparadigmen • können in Gruppen kooperativ und verantwortlich arbeiten Learning targets and competences: Students gain knowledge about methods and realization possibilities of discrete simulation with an outlook on other types of simulation gain knowledge of statistical aspects of simulation that are important for practice apply statistical methods for analysis and evaluation of input and output data gain hands-on experience with commercial simulation tools gain experience in simulation in various fields of application (including computer networks, manufacturing systems, material flow systems) independently develop simulation models on the basis of sample tasks using different modeling paradigms can work in groups cooperatively and responsibly
7	**Voraussetzungen für die Teilnahme**	elementare Programmierkenntnisse, vorzugsweise in Java, Mathematikkenntnisse in Analysis, wie z.B. im 1. Semester der angewandten Mathematik vermittelt Recommended background knowledge: basic programming skills, preferably in Java, mathematics skills in analysis, such as taught in the first semester in applied mathematics.
8	**Einpassung in Studienverlaufsplan**	Semester: 1

9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Law, "Simulation Modeling and Analysis, 5th ed., McGraw Hill, 2014

1	**Modulbezeichnung** 96460	**Speech and Audio Signal Processing** (Speech and audio signal processing)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übung zur Sprach- und Audiosignalverarbeitung (1 SWS) Vorlesung: Sprach- und Audiosignalverarbeitung (3 SWS)	- 5 ECTS
3	Lehrende	Mhd Modar Halimeh Prof. Dr.-Ing. Walter Kellermann	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Walter Kellermann
5	**Inhalt**	It concentrates on algorithms for speech and audio signal processing with applications in telecommunications and multimedia, especially • physiology and models for human speech production and hearing: source-filter model, filterbank model of the cochlea, masking effects, • representation of speech and audio signals: estimation and representation of short-term and long-term statistics in the time and frequency domain as well as the cepstral domain; typical examples and visualizations • source coding for speech and audio signals: criteria, scalar and vector quantization, linear prediction, prediction of the pitch frequency; waveform coding, parametric coding, hybrid coding, codec standards (ITU, GSM, ISO-MPEG) • basic concepts of automatic speech recognition (ASR): feature extraction, dynamic time warping, Hidden Markov Models (HMMs) • basic concepts of speech synthesis: text-to-speech systems, model-based and data-driven synthesis, PSOLA synthesis system • signal enhancement for acquisition and reproduction: noise reduction, acoustic echo cancellation, dereverberation using single-channel and multichannel algorithms. Es werden Grundlagen und Algorithmen der Verarbeitung von Sprach- und Audiosignalen mit Anwendungen in Telekommunikation und Multimedia behandelt, insbesondere: • Physiologie und Modelle der Spracherzeugung und des Hörens: Quelle-Filter-Modell, Filterbank-Modell der Cochlea; Maskierungseffekte; • Darstellung von Sprach- und Audiosignalen: Schätzung und Darstellung der Kurzzeit- und Langzeitstatistik in Zeit-, Frequenz- und Cepstralbereich; typische Beispiele, Visualisierungen; • Quellencodierung für Sprache und Audiosignale: Kriterien; skalare und vektorielle Codierung; lineare Prädiktion; Pitchprädiktion; Wellenform-/Parameter-/Hybrid-Codierung; Standards (ITU, GSM, ISO-MPEG)

		• Spracherkennung: Merkmalextraktion, Dynamic Time Warping, Hidden Markov Models • Grundprinzipien der Sprachsynthese: Text-to-Speech Systeme, modellbasierte und datenbasierte Synthese, PSOLA-Synthese • Signalverbesserung bei Signalaufnahme und wiedergabe: Geräuschbefreiung, Echokompensation, Enthallung mittels ein- und mehrkanaliger Verfahren;
6	**Lernziele und Kompetenzen**	The students • understand basic physiological mechanisms of human speech production and hearing and can apply them for the analysis of speech and audio signals • apply basic methods for the estimation and representation of the short-term and long-term statistics of speech and audio signals and can analyze such signals by means of these methods • understand current methods for source coding of speech and audio signals and can analyze current coding standards • verstehen die Grundbausteine von Spracherkennungssystemen und können deren Funktion mittels Rechnersimulation analysieren • understand the basic principle of text-to-speech systems and can apply fundamental methods for speech synthesis • can apply basic algorithms for speech enhancement and understand their functionality for real-world data. Die Studierenden • verstehen die grundlegenden physiologischen Mechanismen der Spracherzeugung und des Hörens beim Menschen und können diese zur Analyse von Sprach- und Audiosignalen anwenden • wenden die grundlegenden Methoden zur Schätzung und Darstellung der Kurzzeit- und Langzeitstatistik von Sprach- und Audiosignalen an und können diese damit analysieren • verstehen die aktuellen Methoden zur Quellencodierung von Sprache-und Audiosignalen und können aktuelle Codierstandards analysieren • verstehen die Grundbausteine von Spracherkennungssystemen und können deren Funktion mittels Rechnersimulation analysieren • verstehen die Grundprinzipien von Text-to-Speech Systemen und können elementare Algorithmen zur Sprachsynthese anwenden • können elementare Algorithmen zur Signalverbesserung anwenden und für reale Daten analysieren
7	**Voraussetzungen für die Teilnahme**	Vorlesung Signale und Systeme I & II

8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich (90 Minuten)
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Gemäß themenbezogenen Angaben in der Lehrveranstaltung

1	**Modulbezeichnung** 96880	**Speech Enhancement** (Speech enhancement)	**2,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Speech Enhancement (2 SWS)	2,5 ECTS
3	Lehrende	Prof. Dr. Emanuël Habets	

4	**Modulverantwortliche/r**	Prof. Dr. Emanuël Habets
5	**Inhalt**	**Description** We live in a noisy world! In all applications related to speech, from hands-free communication to human-machine interfaces, a speech signal of interest captured by one or more microphones is contaminated by noise and reverberation. The quality and intelligibility of the signal of interest depend highly on the level of noise and reverberation. Therefore, it is highly desirable, and sometimes even indispensable, to "clean up" the captured signals before storage, transmission, or reproduction. This course discusses both model-driven and data-driven methods to estimate the signal of interest and aims to provide a strong foundation for researchers, engineers, and graduate students interested in signal and speech enhancement. **Relation to other courses** This course is the most advanced course offered by the university on this topic and serves as an excellent basis from which to commence research in the area. Various aspects of the course bring students up to date with the very latest developments in the field, as seen in recent international conferences and journals. This course is well complimented by |Selected Topics in Perceptual Audio Coding| (Prof. Herre) and | Auditory Models| (Prof. Edler).
6	**Lernziele und Kompetenzen**	• Formulate the speech enhancement problem mathematically. • Derive optimal single- and multi-channel filters to reduce noise and reverberation. • Evaluate and compare the performance of single- and multi-channel filters for speech enhancement. • Understand how reference signals and other prior information can be used in a speech enhancement system. • Understand the limitations and challenges of existing speech enhancement systems. • Understand the importance of binaural cues and the influence of a speech enhancement system on the binaural cues in the context of hearing aids. • Design a microphone array and analyze its performance.

		• Design a speech enhancement system for a given acoustic scenario. • Evaluate subjectively and objectively the performance of a speech enhancement system in terms of speech quality and intelligibility.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich
11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch oder Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 96430	**Statistical Signal Processing** (Statistical signal processing)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Ergänzungen und Übungen zur statistischen Signalverarbeitung (1 SWS) Vorlesung: Statistische Signalverarbeitung (3 SWS)	- 5 ECTS
3	Lehrende	Thomas Haubner Prof. Dr.-Ing. Walter Kellermann	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Walter Kellermann
5	**Inhalt**	The course concentrates on fundamental methods of statistical signal processing and their applications. The main topics are: *Discrete-time stochastic processes in the time and frequency domain* Random variables (RVs), probability distributions and densities, expectations of random variables, transformation of RVs, vectors of normally distributed RVs, time-discrete random processes: probability distribution and densities, expectation, stationarity, cyclostationarity, ergodicity, correlation functions and correlation matrices, spectral representations, principal component analysis (PCA), Karhunen-Loève transform (KLT). *Estimation theory* estimation criteria, prediction, classical and Bayesian parameter estimation (including MMSE, Maximum Likelihood, and Maximum A Posteriori estimation), Cramer-Rao bound *Linear signal models* Parametric models (cepstral decomposition, Paley-Wiener theorem, spectral flatness), non-parametric models (all-pole, all-zero and pole-zero models, lattice structures, Yule-Walker equations, PARCOR coefficients, cepstral representation) *Signal estimation* Supervised estimation, problem classes, orthogonality principle, MMSE estimation, linear MMSE estimation for normally distributed random processes, optimum FIR filtering, optimum linear filtering for stationary processes, prediction and smoothing, Kalman filters, optimum multichannel filtering (Wiener filter, LCMV, MVDR, GSC) *Adaptive filtering* Gradient methods, LMS, NLMS, APA and RLS algorithms and their convergence behavior

		Zeitdiskrete Zufallsprozesse im Zeit- und Frequenzbereich Zufallsvariablen (ZVn), Wahrscheinlichkeitsverteilungen und dichten, Erwartungswerte; Transformation von ZVn; Vektoren normalverteilter ZVn; zeitdiskrete Zufallsprozesse (ZPe): Wahrscheinlichkeitsverteilungen und dichten, Erwartungswerte, Stationarität, Zyklostationarität, Ergodizität, Korrelationsfunktionen und -matrizen, Spektraldarstellungen; Principal Component Analysis, Karhunen-Loeve Transformation; *Schätztheorie* Schätzkriterien; Prädiktion; klassische und Bayessche Parameterschätzung (inkl. MMSE, Maximum Likelihood, Maximum A Posteriori); Cramer-Rao-Schranke *Lineare Signalmodelle* Parametrische Modelle (Cepstrale Zerlegung, Paley-Wiener Theorem, Spektrale Glattheit); Nichtparametrische Modelle: Allpole-/Allzero-/ Pole-zero-(AR/MA/ARMA) Modelle; Lattice-Strukturen, Yule-Walker Gleichungen, PARCOR-Koeffizienten, Cepstraldarstellungen; *Signalschätzung* Überwachte Signalschätzung, Problemklassen; Orthogonalitätsprinzip, MMSE-Schätzung, lineare MMSE-Schätzung für Gaußprozesse; Optimale FIR-Filter; Lineare Optimalfilter für stationäre Prozesse; Prädiktion und Glättung; Kalman-Filter; optimale Multikanalfilterung (Wiener-Filter, LCMV, MVDR, GSC); *Adaptive Filterung* Gradientenverfahren; LMS-, NLMS-, APA- und RLS-Algorithmus und Ihr Konvergenzverhalten.
6	**Lernziele und Kompetenzen**	The students: • analyze the statistical properties of random variables, random vectors, and stochastic processes by probability density functions and expectations as well as correlation functions and matrices and their frequency-domain representations • know the Gaussian distribution and its role to describe the properties of random variables, vectors and processes • understand the differences between classical and Bayesian estimation, derive and analyze MMSE and ML estimators for specific estimation problems, especially for signal estimation • analyze and evaluate optimum linear MMSE estimators (single- and multichannel Wiener filter and Kalman filter) for direct and inverse supervised estimation problems

		• evaluate adaptive filters for the identification of optimum linear estimators. Die Studierenden • analysieren die statistischen Eigenschaften von Zufallsvariablen, -vektoren und stochastischen Prozessen mittels Wahrscheinlichkeitsdichten und Erwartungswerten, bzw. Korrelationsfunktionen, Korrelationsmatrizen und deren Frequenzbereichsdarstellungen • kennen die spezielle Rolle der Gaußverteilung und ihre Auswirkungen auf die Eigenschaften von Zufallsvariablen, -vektoren und Prozessen • verstehen die Unterschiede klassischer und Bayesscher Schätzung, entwerfen und analysieren MMSE- und ML-Schätzer für spezielle Schätzprobleme, insbesondere zur Signalschätzung • analysieren und evaluieren lineare MMSE-optimale Schätzer (ein- und vielkanalige Wiener-Filter und Kalman-Filter) für direkte und inverse überwachte Schätzprobleme; • evaluieren adaptive Filter zur Identifikation optimaler linearer Signalschätzer
7	**Voraussetzungen für die Teilnahme**	Module Signale und Systeme I und Signale und Systeme II, Digitale Signalverarbeitung oder gleichwertige
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Pflichtmodule Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	A. Papoulis, S. Pillai: Probability, Random Variables and Stochastic Processes; McGraw-Hill, 2002 (englisch) D. Manolakis, V. Ingle, S. Kogon: Statistical and Adaptive Signal Processing; Artech House, 2005 (englisch)

1	**Modulbezeichnung** 44500	**Swarm Intelligence** (Swarm Intelligence)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übungen zu Swarm Intelligence (2 SWS) Vorlesung: Swarm Intelligence (SI), formerly Organic Computing (OC) (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Matthias Kergaßner Prof. Dr. Rolf Wanka	

4	**Modulverantwortliche/r**	Prof. Dr. Rolf Wanka
5	**Inhalt**	Swarm Intelligence (SI) is the design and deployment of self-organizing systems that dynamically adapt to their respective environmental needs. These systems are characterized by the fact that they feature the so-called self-*-properties, i.e., they are self-configuring, self-optimizing, self-healing, self-protecting, self-explanatory, ... Structures and methods of biological and other natural systems are chosen as models for such technical systems. In this module, Particle Swarm Optimization, Ant Algorithms, Web Search, and Evolutionary Algorithms are introduced and, as far as possible, mathematically analyzed.
6	**Lernziele und Kompetenzen**	Students learn advanced concepts of the current topic of swarm intelligence and how they can be successfully applied to solve continuous and discrete optimization problems and to data analysis. For this purpose, they know concrete details such as terms, definitions, facts, regularities and theories and learn how to apply the concepts to concrete problems, how to adjust the methods to the use case and how to analyze the computed solutions.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch

16	**Literaturhinweise**	• Ch. Müller-Schloer, Ch. von der Malsburg, R. P. Würt. Organic Computing. Informatik-Spektrum, Band 27, Nummer 4, S. 332-336. • I. C. Trelea. The particle swarm optimization algorithm: convergence analysis and parameter selection. Information Processing Letters 85 (2003) 317-325. • J. M. Kleinberg. Authoritative sources in a hyperlinked environment. Journal of the ACM 46 (1999) 604-632. • M. Dorigo. V. Maniezzo. A Colorni. Ant system: an autocatalytic optimizing process. Technical Report 91-016, Politecnico di Milano, 1991. • A. Badr. A. Fahmy. A proof of convergence for Ant algorithms. Information Sciences 160 (2004) 267-279. • M. Clerc. J. Kennedy. The particle swarm - Explosion, stability, and convergence in a multidimensional complex space. IEEE Transactions on Evolutionary Computation 8 (2002) 58-73

1	**Modulbezeichnung** 96621	**Transceiver-Systementwurf** (keine englischsprachige Modulbezeichnung hinterlegt!)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Transceiver-Systementwurf (2 SWS) Übung: Übung Transceiver-Systementwurf (2 SWS)	5 ECTS -
3	Lehrende	Prof. Dr. Jörn Thielecke	

4	**Modulverantwortliche/r**	Prof. Dr. Jörn Thielecke
5	**Inhalt**	1. Einführung 2. Systemübersicht und Anforderungen • GPSGSM-WLAN • Vergleichende Zusammenfassung 3. Basisbandverarbeitung • Optimierung und Wechselwirkungen am Beispiel einer PLL • Anforderungsprofil bei GPS, GSM und WLAN 4. A/D- und D/A-Umsetzung • Dominierendes Nutzsignal bei GSM und WLAN • Dominierendes Rauschen bei GPS • Anforderungsübersicht 5. Frontend • Analyse und Charakterisierung von Störungen (Nichtlinearitäten, Rauschen, Dynamikbereicht, I/Q-Balance, Phasenrauschen) • Systementwurf (Entwurfzyklus, Empfänger-Architekturen, Sender-Architekturen) 6. Ausblick
6	**Lernziele und Kompetenzen**	1. Anhand der Beispielsysteme GPS, GSM und WLAN sollen Studierende beurteilen lernen, wie das Wechselspiel zwischen Realisierungsaufwand und nachrichtentechnischer Systemanforderung ist. 2. Anhand von Beispielen sollen Studierenden die wesentlichen Entwurfsschritte bis hin zur Parametrisierung auf Blockschaltbildebene klar werden, wenn der Ausgangspunkt eine nachrichtentechnische Systembeschreibung ist. 3. Anhand von Architekturbeispielen sollen Studierende ein Verständnis für die Spielräume und Abwägungen beim Entwurf eines Endgerätes entwickeln.

7	**Voraussetzungen für die Teilnahme**	Keine formalen Voraussetzungen, grundlegende Kenntnisse erforderlich aus: Signal- und Systemtheorie, Nachrichtentechnische Systeme, Stochastik. Erste mikroelektronische Kenntnisse helfen.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Embedded Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich (90 Minuten)
11	**Berechnung der Modulnote**	mündlich (100%) Bei bestandener Prüfung wird die Note um eine Teilnotenstufe (z.B. von 2,0 auf 1,7) verbessert, wenn Sie mindestens 75% der Hausaufgaben einschließlich der Rechnerübungen erfolgreich absolviert haben. Eine Note besser als 1,0 wird nicht vergeben.
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Skriptum zur Vorlesung.

1	**Modulbezeichnung** 498723	**Transformationen in der Signalverarbeitung** (Transforms in signal processing)	**2,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Transformationen in der Signalverarbeitung (2 SWS)	2,5 ECTS
3	Lehrende	PD Dr.Ing. Jürgen Seiler	

4	**Modulverantwortliche/r**	PD Dr.Ing. Jürgen Seiler
5	**Inhalt**	Das Modul "Transformationen in der Signalverarbeitung" behandelt mehrere verschiedene Transformationen, die im Rahmen der Signalverarbeitung Verwendung finden. Dabei werden zuerst die grundlegenden Konzepte von Transformationen diskutiert und die Vorteile die Transformationen mit sich bringen erläutert. Im Anschluss daran werden die grundlegenden Eigenschaften von Integraltransformationen betrachtet und die Laplace- und die Fourier-Transformation im Detail untersucht. Um auch zeitlich veränderliche Signale gut transformieren zu können werden danach die Kurzzeit-Fourier-Transformation und die Gabor-Transformation eingeführt. Im Anschluss daran erfolgt eine Betrachtung der Auswirkung der Abtastung auf transformierte Signale, bevor die z-Transformation als Transformation für diskrete Signale behandelt wird. Abschließend erfolgt die Betrachtung weiterer Transformationen für diskrete Signale wie der Diskreten Fourier-Transformation oder linearer Block-Transformationen. The module "Transforms in Signal Processing" covers several different transforms which are used in the field of signal processing. For this, first the basic concepts of transforms are discussed and the advantages which are offered by the different transforms are presented. Subsequent to this, fundamental properties of integral transforms are considered and the Laplace- and the Fourier-Transform are examined in detail. To be able to transform time-varying signals, the Short-Time Fourier-Transform and the Gabor-Transform are introduced, afterwards. Subsequent to this, the impact of sampling on transformed signals is analyzed before the z-Transform as a transform for discrete signals is covered. Finally, further transforms for discrete signals like the Discrete Fourier-Transform or Linear-Block Transforms are discussed.
6	**Lernziele und Kompetenzen**	Die Studierenden können nach Besuch der Vorlesung • Anwendungsmöglichkeiten von Transformationen bestimmen • Integraltransformationen gegenüberstellen und untersuchen • die Existenz von Transformationen hinterfragen • die Eindeutigkeit von Transformationen überprüfen • Sätze und Eigenschaften von Transformationen entwickeln • zu Transformationen zugehörige inverse Transformationen einschätzen • die Zusammenhänge zwischen verschiedenen Transformationen einschätzen • auf Zusammenhänge zwischen Ausgangssignalen und transformierten Signalen folgern • Symmetriebeziehungen von Transformationen ausarbeiten

		• Zusammenhänge zwischen kontinuierlichen und diskreten Signalen ausarbeiten Educational Objectives and Competences: After attending the lecture, students will be able to • determine applications of transforms • contrast and examine integral transforms • question the existence of transforms • evaluate the uniqueness of transforms • develop theorems and properties of transforms • evaluate to transforms corresponding inverse transforms • evaluate the relationships between different transforms • asses the relationship between original signal and transformed signals • devise the symmetry properties of transforms • devise the relationship between continuous and discrete signals
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	K. Krüger, Transformationen - Grundlagen und Anwendungen in der Nachrichtentechnik, Vieweg Verlag, Braunschweig B. Girod, R. Rabenstein, A. Stenger, Einführung in die Systemtheorie, B. G. Teubner Verlag, Stuttgart

1	**Modulbezeichnung** 43420	**Transmission and Detection for Advanced Mobile Communications** (Transmission and detection for advanced mobile communications)	**2,5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Transmission and Detection for Advanced Mobile Communications (2 SWS)	2,5 ECTS
3	Lehrende	Prof. Dr. Wolfgang Gerstacker	

4	**Modulverantwortliche/r**	Prof. Dr. Wolfgang Gerstacker
5	**Inhalt**	The aim of this lecture is that the students acquire a basic knowledge of advanced transmission and detection techniques which are relevant to practical mobile communications systems. In the first part, it is shown how equalization schemes like decision-feedback equalization (DFE) and maximum-likelihood sequence estimation (MLSE) can be applied to the GSM/EDGE (Enhanced Data Rates for GSM Evolution) standard. Also, channel estimation for GSM/EDGE is covered. In GSM/EDGE, disturbance by interfering signals of other users is a further major problem. Therefore, interference cancellation algorithms are discussed in detail. The cases of several receive antennas and one receive antenna (single antenna interference cancellation) are distinguished. Several receive antennas can be also utilized for increasing the robustness against fading, applying diversity combination techniques. In the case of the availability of several transmit antennas only, additional space-time coding has to be used for realization of diversity gains. These aspects are also discussed in depth. Furthermore, an introduction to code-division multiple access (CDMA) transmission is given and it is shown how CDMA is applied in the UMTS system. The lecture is concluded by an introduction to digital transmission in the Long Term Evolution (LTE) system.
6	**Lernziele und Kompetenzen**	The students • describe basic equalization algorithms such as decision-feedback equalization (DFE) and maximum-likelihood sequence estimation (MLSE), • apply equalization algorithms to the GSM / Enhanced Data Rates for GSM Evolution (EDGE) mobile communication system, • formulate channel estimation methods for mobile communication systems, • characterize the interference problem in GSM / EDGE, - design interference suppression schemes for GSM/EDGE for receivers with a single antenna (single antenna interference cancellation) and multiple antennas, respectively, • characterize the performance of mobile communication networks for different reception schemes,

		• devise receivers for the realization of diversity gains for multiple receive antennas, • design space-time coding schemes for the realization of diversity gains for multiple transmit antennas, • describe transmission schemes which are based on code-division multiple access (CDMA), • apply reception techniques for CDMA to the UMTS system, • characterize the uplink transmission in the Long Term Evolution (LTE) system, • develop receivers for LTE. Die Studierenden • beschreiben grundlegende Entzerrverfahren wie entscheidungsrückgekoppelte Entzerrung (Decision-Feedback Equalization, DFE) und Maximum-Likelihood-Sequenzschätzung (Maximum-Likelihood Sequence Estimation, MLSE), • wenden Entzerrverfahren auf das GSM/EDGE (Enhanced Data Rates for GSM Evolution) Mobilfunksystem an, • formulieren Kanalschätzverfahren für Mobilfunksysteme, • charakterisieren das Interferenzproblem bei GSM/EDGE, • entwerfen Interferenzunterdrückungsverfahren für GSM/EDGE für Empfänger mit einer Antenne (Single Antenna Interference Cancellation) und mehreren Antennen, • bewerten die Leistungsfähigkeit von Mobilfunknetzen bei Einsatz verschiedener Empfangsverfahren, • konzipieren Empfänger zur Realisierung von Diversitätsgewinnnen bei empfangsseitiger Antennendiversität • entwerfen Space-Time-Codierverfahren zur Realisierung von Diversitätsgewinnen bei sendeseitiger Antennendiversität, • beschreiben auf Code-Division Multiple Access (CDMA) basierende Übertragungsverfahren, • wenden Empfangsverfahren für CDMA auf das UMTS-System an, • charakterisieren die Aufwärtsstrecke von Long Term Evolution (LTE), • entwerfen Empfänger für LTE.
7	**Voraussetzungen für die Teilnahme**	Systemtheorie, Nachrichtenübertragung
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus EEI im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich oder mündlich

11	**Berechnung der Modulnote**	schriftlich oder mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Lecture notes

1	**Modulbezeichnung** 92280	**Verifikation digitaler Systeme** (Verifikation digitaler Systeme)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übung zur Verifikation digitaler Systeme (2 SWS) Vorlesung: Verifikation digitaler Systeme (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Prof. Dr. Oliver Keszöcze	

4	**Modulverantwortliche/r**	Prof. Dr. Oliver Keszöcze
5	**Inhalt**	Für den Entwurf eines digitalen Systems werden heute in der Industrie ebenso viele Verifikationsingenieure wie Designer benötigt. Trotzdem beansprucht die Verifikation heute bereits 70%-80% der gesamten Entwurfszeit. Neben konventionellen Verifikationserfahren wie der Simulation sind werden seit einigen Jahren sogenannte "formale Verifikationsmethoden" in heutigen Entwursflüssen eingesetzt. Der Umgang mit diesen Methoden stellt ein wichtiges neues Aufgabenfeld dar. Im Gegensatz zur Simulation beruht die formale Verifikation auf exakten mathematischen Methoden zum Nachweis funktionaler Schaltungseigenschaften. Dadurch können Entwurfsfehler frühzeitiger und mit höherer Zuverlässigkeit als bisher erkannt werden. Jedes System zur formalen Hardwareverifikation erfordert: 1. ein geeignetes Modell des zu verifizierenden Systems 2. eine Sprache zur Formulierung der zu verifizierenden Eigenschaften 3. eine Beweismethode. Die Vorlesung behandelt diese drei Bereiche, vermittelt die grundlegenden Algorithmen und Konzepte moderner Werkzeuge für die formale Hardwareverifikation und erläutert deren Einsatz in der industriellen Praxis. Im Einzelnen werden in dieser Vorlesung die folgenden Punkte behandelt: 1. Modellierung digitaler Systeme 2. Unterschiede formaler und simulationsbasierter Verifikationsmethoden 3. Äquivalenzvergleich 4. Formale und simulationsbasierte Eigenschaftsprüfung 5. Assertions 6. Verifikation arithmetischer Schaltungen
6	**Lernziele und Kompetenzen**	Fachkompetenz -- Verstehen • Die Studierenden erläutern die wesentlichen Techniken zur Verifikation digitaler Systeme, zeigen deren Vor- und Nachteile

		auf und vergleichen diese bezüglich Ihrer Mächtigkeit und Komplexität. Fachkompetenz -- Anwenden • Die Studierenden wenden Modellierungstechniken aus den Bereichen der Binären Entscheidungsdiagramme inkl. bekannter Erweiterungen (MDD, etc.) sowie der SAT-Löser auf Systembeschreibungen an. • Die Studierenden Verifikationstechniken aus den Bereichen der formalen Äquivalenz- und Eigenschaftsprüfung (Model Checking, Symbolic Model Checking, Bounded Model Checking) auf Systembeschreibungen an.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 593320	**Vernetzte Mobilität und autonomes Fahren** (Connected mobility and autonoumous driving)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Connected Mobility and Autonomous Driving (Exercises) (2 SWS) Vorlesung: Connected Mobility and Autonomous Driving (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Dr.-Ing. Anatoli Djanatliev	

4	**Modulverantwortliche/r**	Dr.-Ing. Anatoli Djanatliev
5	**Inhalt**	Es ist inzwischen allgemein bekannt, dass Fahrzeuge der Zukunft hochgradig vernetzt sein werden. Der aktuelle Trend geht in Richtung des autonomen Fahrens. In den bisheringen Betrachtungen wurde insbesondere die ad-hoc Kommunikation zwischen Fahrzeugen auf unteren Schichten untersucht (Fahrzeugkommunikation). Im Rahmen der vernetzten Mobilität soll das Fahrzeug vor allem als Teil eines größeren Ökosystems mit weiteren Teilnehmern (z.B. Personen, Radfahrern, Ampeln, Gebäuden etc.) gesehen werden. All dies gibt die Möglichkeit den ständig wachsenden Bedarf an Mobilität zu optimieren und neue Sicherheits- und Komfortdienstleistungen zu schaffen. Dies erfordert jedoch die Lösung einiger komplexer Herausforderungen. Neben den gesellschaftlichen und rechtlichen Aspekten müssen insbesondere auch technische Voraussetzungen geschaffen werden. Dazu gehören u.a. geeignete Kommunikationetechnologien (v.a. ad-hoc, Mobilfunk) und Kommunikationsarchitekturen (Cloud-, Edge/Fog-, Node-Computing). Neben Technologien, Methoden und innovativen Mobilitätsdienstleitungen werden im Rahmen dieser Lehrveranstaltung auch grundlegende Aspekte der Verkehrsplanung und Verkehrstechnik eingeführt sowie der intermodale Verkehr besprochen.
6	**Lernziele und Kompetenzen**	Fachkompetenz Verstehen Verständnis grundlegender Konzepte u.a. von • Ad-Hoc Kommunikation • Mobilfunkkommunikation • Verkehrsplanung • Architekturen • Fahrzeug als Teil eines Mobilitäts-Ökosystems • Innovative Dienste Anwenden Bearbeitung von Übungsaufgaben Analysieren

		• Unterschiede zwischen unterschiedlichen Kommunikationstechnologien und Architekturen aufdecken • Relevante Zukunftsszenarien aufbauen Evaluieren (Beurteilen) Anwendung von Simulation und Modellierung zur Evaluierung zukünftiger Szenarien und Fallstudien.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlpflichtmodul aus INF im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Barbara Flügge; Smart Mobility - Connecting Everyone: Trends, Concepts and Best Practices; Vieweg Teubner, 2017 Maurer, M., Gerdes, J.C., Lenz, B., Winner, H. (Hrsg.); Autonomes Fahren: Technische, rechtliche und gesellschaftliche Aspekte; Springer 2015 Johanning, V., Mildner, R.; Car IT kompakt: Das Auto der Zukunft Vernetzt und autonom fahren; Springer, 2015

1	**Modulbezeichnung** 743260	**Verteilte Systeme (Vorlesung mit erweiterten Übungen)** (Distributed systems (lecture with extended exercises))	**7,5 ECTS**
2	Lehrveranstaltungen	Übung: Rechnerübungen zu Verteilte Systeme (2 SWS) Übung: Erweiterte Übungen zu Verteilte Systeme (2 SWS) Vorlesung: Verteilte Systeme (2 SWS)	- 5 ECTS 2,5 ECTS
3	Lehrende	PD Dr.Ing. Tobias Distler Laura Lawniczak	

4	**Modulverantwortliche/r**	PD Dr.Ing. Tobias Distler
5	**Inhalt**	Verteilte Systeme bestehen aus mehreren Rechnern, die über ein Netzwerk miteinander verbunden sind und einen gemeinsamen Dienst erbringen. Obwohl die beteiligten Rechner hierfür in weiten Teilen unabhängig voneinander agieren, erscheinen sie ihren Nutzern gegenüber in der Gesamtheit dabei trotzdem als ein einheitliches System. Die Einsatzmöglichkeiten für verteilte Systeme erstrecken sich über ein weites Spektrum an Szenarien: Von der Zusammenschaltung kleinster Rechenknoten zur Sammlung von Daten im Rahmen von Sensornetzwerken über Steuerungssysteme für Kraftfahrzeuge und Industrieanlagen bis hin zu weltumspannenden, Internet-gestützten Infrastrukturen mit Komponenten in Datenzentren auf verschiedenen Kontinenten. Ziel dieses Moduls ist es, die sich durch die speziellen Eigenschaften verteilter Systeme ergebenden Problemstellungen zu verdeutlichen und Ansätze zu vermitteln, mit deren Hilfe sie gelöst werden können; Beispiele hierfür sind etwa die Interaktion zwischen heterogenen Systemkomponenten, der Umgang mit erhöhten Netzwerklatenzen sowie die Wahrung konsistenter Zustände über Rechnergrenzen hinweg. Gleichzeitig zeigt das Modul auf, dass die Verteiltheit eines Systems nicht nur Herausforderungen mit sich bringt, sondern auf der anderen Seite auch Chancen eröffnet. Dies gilt insbesondere in Bezug auf die im Vergleich zu nicht verteilten Systemen erzielbare höhere Widerstandsfähigkeit eines Gesamtsystems gegenüber Fehlern wie den Ausfällen ganzer Rechner oder sogar kompletter Datenzentren. Ausgehend von den einfachsten, aus nur einem Client und einem Server bestehenden verteilten Systemen, beschäftigt sich die Vorlesung danach mit der deutlich komplexeren Replikation der Server-Seite und behandelt anschließend die Verteilung eines Systems über mehrere, mitunter weit voneinander entfernte geografische Standorte. In allen Abschnitten umfasst die Betrachtung des jeweiligen Themas eine Auswahl aus Grundlagen, im Praxiseinsatz befindlicher Ansätze und Techniken sowie für den aktuellen Stand der Forschung repräsentativer Konzepte. Im Rahmen der Übungen wird zunächst ein plattformunabhängiges Fernaufrufsystem schrittweise entwickelt und parallel dazu getestet.

		Als Vorlage und Orientierungshilfe dient dabei das in der Praxis weit verbreitete Java RMI. In den weiteren Übungsaufgaben stehen anschließend klassische Problemstellungen von verteilten Systemen wie fehlertolerante Replikation und verteilte Synchronisation im Mittelpunkt.
6	**Lernziele und Kompetenzen**	Studierende, die das Modul erfolgreich abgeschlossen haben: • beschreiben charakteristische Merkmale und Eigenschaften verteilter Systeme sowie grundlegende Probleme im Zusammenhang mit ihrer Realisierung. • untersuchen die Unterschiede zwischen lokalen Methodenaufrufen und Fernmethodenaufrufen. • vergleichen Ansätze zur Konvertierung von Nachrichten zwischen verschiedenen Datenrepräsenationen. • konzipieren eine eigene auf Java RMI basierende Anwendung. • entwickeln ein eigenes Fernaufrufsystem nach dem Vorbild von Java RMI. • bewerten die Serialisierungsroutinen von Java RMI. • erproben die manuelle Serialisierung von Nachrichten. • bewerten die Performanz des eigenen Fernaufrufsystems. • gestalten ein Modul zur Unterstützung verschiedener Fernaufrufsemantiken (Maybe, Last-of-Many, At-Most-Once) für das eigene Fernaufrufsystem. • beurteilen auf Basis eigener Experimente mit Fehlerinjektionen die Auswirkungen von Störeinflüssen auf verschiedene Fernaufrufsemantiken. • klassifizieren Mechanismen zur Bereitstellung von Fehlertoleranz, insbesondere verschiedene Arten der Replikation (aktiv vs. passiv). • vergleichen verschiedene Konsistenzgarantien georeplizierter Systeme. • illustrieren das Problem einer fehlenden gemeinsamen Zeitbasis in verteilten Systemen. • erforschen logische Uhren als Mittel zur Reihenfolgebestimmung und Methoden zur Synchronisation physikalischer Uhren. • unterscheiden grundlegende Zustellungs- und Ordnungsgarantien beim Multicast von Nachrichten. • gestalten ein Protokoll für den zuverlässigen und totalgeordneten Versand von Nachrichten in einer Gruppe von Knoten. • entwickeln einen Dienst zur Verwaltung verteilter Sperrobjekte auf Basis von Lamport-Locks. • bewerten die Qualität einer Publikation aus der Fachliteratur. • erschließen sich typische Probleme (Nebenläufigkeit, Konsistenz) und Fehlerquellen bei der Programmierung verteilter Anwendungen. • können in Kleingruppen kooperativ arbeiten. • können ihre Entwurfs- und Implementierungsentscheidungen kompakt präsentieren und argumentativ vertreten.

		• reflektieren ihre Entscheidungen kritisch und leiten Alternativen ab. • können offen und konstruktiv mit Schwachpunkten und Irrwegen umgehen.
7	**Voraussetzungen für die Teilnahme**	Gute Programmierkenntnisse in Java
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio Erfolgreiche Bearbeitung aller während des Semesters gestellten Übungsaufgaben (6 Aufgaben, Bewertung jeweils mit "ausreichend") + 30-minütige mündliche Prüfung am Ende des Semesters. Die Modulnote ergibt sich zu 100% aus der Bewertung der mündlichen Prüfung.
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 90 h Eigenstudium: 135 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 202041	**Virtuelle Maschinen (Vorlesung mit Übungen)** (Lecture and tutorial: Virtual machines)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Virtuelle Maschinen - Übungen (2 SWS) Vorlesung: Virtuelle Maschinen (2 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Dr.-Ing. Volkmar Sieh	

4	**Modulverantwortliche/r**	Dr.-Ing. Volkmar Sieh
5	**Inhalt**	Vorgestellt werden verschiedene Virtualisierungs-Ansätze: • Emulation • Just-In-Time-Compiler • Para-Virtualisierung • Bibliotheks-basierte Virtualisierung • OS-Virtualisierung
6	**Lernziele und Kompetenzen**	Studierende, die das Modul erfolgreich abgeschlossen haben: • erläutern verschiedene Motivationen für den Einsatz von VMs • unterscheiden verschiedene VMs • klassifizieren verschiedene Ziele unterschiedlicher VMs (z.B. Performance, Konfigurierbarkeit, Genauigkeit, ...) • hinterfragen verschiedene Simulationansätze für MMUs • erstellen virtuelle Komponenten und Busse • strukturieren Callbacks und entsprechendes Forwarding und Caching • unterscheiden zwischen Architektur, Chip und Komponente • klassifizieren unterschiedliche Just-In-Time-Compiler-Ansätze • erzeugen JIT Code aus vorgefertigten Code-Teilen • bewerten unterschiedliche JIT-Code-Optimierungen • erläutern Probleme bei der JIT-Code-Invalidierung • nennen JIT Probleme mit Exceptions/Interrupts sowie berechnete Sprüngen und Return-Instruktionen • unterscheiden verschiedene JIT Cache-Verwaltungen • beschreiben Möglichkeiten der Fehlerinjektion durch VMs • entwickeln ein an JIT angepasstes virtuelles "Hardware"-Design • erläutern die Java-VM Instruktionssatz-Architektur • nutzen Hardware-basierte Virtualisierung • entwickeln Verfahren zum Ausfiltern bestimmter Befehle • erläutern Probleme der Speicherverwaltung bei HW-basierter Virtualisierung • nutzen User-Mode-Emulation zur Paravirtualisierung • diskutieren Möglichkeiten von Debuggern für die Umleitung von System-Calls und die Ausfilterung von Befehlen • nutzen einen Hypervisor zur Paravirtualisierung • unterscheiden verschiedene Ansätze zur Geräteverwaltung in paravirtualisierten Systemen • erläutern Betriebssystem-basierte Virtualisierung • entwickeln unterschiedliche Bibliotheks-basierte Virtualisierungen

		• erläutern Probleme beim Speicher-Layout bei Bibliotheks-basierte Virtualisierung • konzipieren Personalities für Bibliotheks-basierte Virtualisierungen • beurteilen Probleme bei der korrekten Zeit-Simulation • nennen Ideen für die dynamische Anpassung der Zeit-Simulation • klassifizieren bekannte VMs (z.B. VICE, FAUmachine, QEMU, Bochs, JVM, KVM, User-Mode-Linux, Xen, VServer, Wine) • diskutieren in der Gruppe Vor- und Nachteile von bestimmten VM-Ansätzen • untersuchen CPU-Emulationen • untersuchen Geräte-Emulationen
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich
11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 462793	**Virtuelle Maschinen (Vorlesung mit Übung und Laborübung)** (Lecture, tutorial and laboratory: Virtual machines)	**7,5 ECTS**
2	Lehrveranstaltungen	Übung: Virtuelle Maschinen - Erweiterte Übungen (2 SWS) Übung: Virtuelle Maschinen - Übungen (2 SWS) Vorlesung: Virtuelle Maschinen (2 SWS)	2,5 ECTS 2,5 ECTS 2,5 ECTS
3	Lehrende	Dr.-Ing. Volkmar Sieh	

4	**Modulverantwortliche/r**	Dr.-Ing. Volkmar Sieh
5	**Inhalt**	Vorgestellt werden verschiedene Virtualisierungs-Ansätze: • Emulation • Just-In-Time-Compiler • Para-Virtualisierung • Bibliotheks-basierte Virtualisierung • OS-Virtualisierung
6	**Lernziele und Kompetenzen**	Studierende, die das Modul erfolgreich abgeschlossen haben: • erläutern verschiedene Motivationen für den Einsatz von VMs • unterscheiden verschiedene VMs • klassifizieren verschiedene Ziele unterschiedlicher VMs (z.B. Performance, Konfigurierbarkeit, Genauigkeit, ...) • hinterfragen verschiedene Simulationansätze für MMUs • erstellen virtuelle Komponenten und Busse • strukturieren Callbacks und entsprechendes Forwarding und Caching • unterscheiden zwischen Architektur, Chip und Komponente • klassifizieren unterschiedliche Just-In-Time-Compiler-Ansätze • erzeugen JIT Code aus vorgefertigten Code-Teilen • bewerten unterschiedliche JIT-Code-Optimierungen • erläutern Probleme bei der JIT-Code-Invalidierung • nennen JIT Probleme mit Exceptions/Interrupts sowie berechnete Sprüngen und Return-Instruktionen • unterscheiden verschiedene JIT Cache-Verwaltungen • beschreiben Möglichkeiten der Fehlerinjektion durch VMs • entwickeln ein an JIT angepasstes virtuelles "Hardware"-Design • erläutern die Java-VM Instruktionssatz-Architektur • nutzen Hardware-basierte Virtualisierung • entwickeln Verfahren zum Ausfiltern bestimmter Befehle • erläutern Probleme der Speicherverwaltung bei HW-basierter Virtualisierung • nutzen User-Mode-Emulation zur Paravirtualisierung • diskutieren Möglichkeiten von Debuggern für die Umleitung von System-Calls und die Ausfilterung von Befehlen • nutzen einen Hypervisor zur Paravirtualisierung

		• unterscheiden verschiedene Ansätze zur Geräteverwaltung in paravirtualisierten Systemen • erläutern Betriebssystem-basierte Virtualisierung • entwickeln unterschiedliche Bibliotheks-basierte Virtualisierungen • erläutern Probleme beim Speicher-Layout bei Bibliotheks-basierte Virtualisierung • konzipieren Personalities für Bibliotheks-basierte Virtualisierungen • beurteilen Probleme bei der korrekten Zeit-Simulation • nennen Ideen für die dynamische Anpassung der Zeit-Simulation • klassifizieren bekannte VMs (z.B. VICE, FAUmachine, QEMU, Bochs, JVM, KVM, User-Mode-Linux, Xen, VServer, Wine) • diskutieren in der Gruppe Vor- und Nachteile von bestimmten VM-Ansätzen • entwickeln selbst CPU-Emulationen • entwickeln selbst Geräte-Emulationen • verteilen Implementierungsaufgaben in ihrer Gruppe • erstellen Zeitpläne für Implementierungen für sich und ihre Gruppe
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 90 h Eigenstudium: 135 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

Wahlmodule aus dem Angebot der Technischen Fakultät oder der Naturwissenschaftlichen Fakultät

1	**Modulbezeichnung** 96312	**Bild-, Video- und mehrdimensionale Signalverarbeitung** (Image, video and multidimensional signal processing)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Bild-, Video- und mehrdimensionale Signalverarbeitung (2 SWS)	5 ECTS
3	Lehrende	Prof. Dr.-Ing. Andre Kaup	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Andre Kaup
5	**Inhalt**	*Punktoperationen* Histogrammausgleich, Gamma-Korrektur *Binäroperationen* Morphologische Filter, Erosion, Dilatation, Opening, Closing *Farbräume* Trichromat, RGB- Farbraum, HSV-Farbraum *Mehrdimensionale Signale und Systeme* Theorie mehrdimensionaler Signale und Systeme, Impulsantwort, lineare Bildfilterung, Leistungsspektrum, Wiener Filter *Interpolation von Bildsignalen* Bilineare Interpolation, Bicubische Interpolation, Spline Interpolation *Merkmalsdetektion in Bildern* Bildmerkmale, Kantendetektion, Hough Transformation, Harris Ecken Detektor, Texturmerkmale, Grauwertematrix *Skalierungsraumdarstellung* LoG, DoG, SIFT, SURF *Bildabgleich* Projektive Abbildungen, Blockabgleich, Optischer Fluss, Merkmalsbasierter Abgleich mittels SIFT und SURF, RANSAC *Bildsegmentierung* Amplituden Schwellenwertermittlung, K-Means Clustering, Bayes Klassifikation, Regionen-basierte Segmentierung, kombinierte Segmentierung und Bewegungsschätzung, zeitliche Segmentierung von Videos

Bildverarbeitung im Transformationsbereich

Unitäre Transformation, Karhunen-Loeve Transformation, separable Transformationen, Haar und Hadamard Transformation, DFT, DCT

Content:

Point operations

Histogram equalization, gamma correction

Binary operations

Morphological filters, erosion, dilation, opening, closing

Color spaces

Trichromacy, red-green-blue color spaces, color representation using hue, saturation and value of intensity

Multidimensional signals and systems

Theory of multidimensional signals and systems, impulse response, linear image filtering, power spectrum, Wiener filtering

Interpolation of image signals

Bi-linear interpolation, bi-cubic interpolation, spline interpolation

Image feature detection

Image features, edge detection, Hough transform, Harris corner detector, texture features, co-occurrence matrix

Scale space representation

Laplacian of Gaussian, difference of Gaussian, scale invariant feature transform, speeded-up robust feature transform

Image matching

Projective transforms, block matching, optical flow, feature-based matching using SIFT and SURF, random sample consensus algorithm

Image segmentation

		Amplitude thresholding, k-means clustering, Bayes classification, region-based segmentation, combined segmentation and motion estimation, temporal segmentation of video *Transform domain image processing* Unitary transform, Karhunen-Loeve transform, separable transform, Haar and Hadamard transform, DFT, DCT
6	**Lernziele und Kompetenzen**	Die Studierenden • verstehen Punktoperationen an Bilddaten und Gamma-Korrektur • testen die Wirkung von Rangordnungs- und Medianfiltern an Bilddaten • unterscheiden und bewerten verschiedene Farbräume für Bilddaten • erklären das Prinzip der zwei-dimensionalen linearen Filterung für Bildsignale • berechnen und bewerten die zweidimensionale diskrete Fourier-Transformierte eines Bildsignales • bestimmen vergrößerte diskrete Bildsignale mit Methoden der bilinearen und Spline-Interpolation • überprüfen Bilddaten auf ausgewählte Textur-, Kanten- und Bewegungsmerkmale • analysieren Bild- und Videodaten auf Merkmale in unterschiedlichen Scale-Spaces • erläutern und beurteilen Methoden für das Matching von Bilddaten • segmentieren Bilddaten durch Programmierung von einfachen Klassifikations- oder Clustering-Verfahren • verstehen das Prinzip von Transformation auf Bilddaten und können diese an Beispielen anwenden. The students • understand point operations for image data and gamma correction • test the effects of rank order and median filters for image data • evaluate and differentiate between different color spaces for image data • explain the principle of two-dimensional linear filtering for image signals • calculate and evaluate the two-dimensional discrete Fourier transform of an image signal • determine enlarged discrete image signals by bi-linear and spline interpolation • verify image data for selected texture, edge and motion features

		• analyze image and video data for features in different scale spaces • explain and evaluate methods for the matching of image data • segment image data by implementing basic classification and clustering methods • understand the principle of transformations on image data and apply them exemplarily
7	**Voraussetzungen für die Teilnahme**	Vorlesung Signale und Systeme I und II
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot der Technischen Fakultät oder der Naturwissenschaftlichen Fakultät Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur (90 Minuten)
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	J.-R. Ohm: |Multimedia Content Analysis|, Springer, 2016 J. W. Woods: |Multidimensional Signal, Image, and Video Processing and Coding|, Academic Press, 2^nd^ edition, 2012

1	**Modulbezeichnung** 66000	**Experimentalphysik I** (Experimental physics for EECE I)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Übungen zur Experimentalphysik für Elektro- und Medizintechniker I (1 SWS) Vorlesung: Experimentalphysik für Elektro- und Medizintechniker I (3 SWS)	- -
3	Lehrende	Prof. Dr. Bernhard Hensel	

4	**Modulverantwortliche/r**	Prof. Dr. Bernhard Hensel
5	**Inhalt**	*Inhaltsangabe für beide Semester* • Physikalische Größen und Messungen • Mechanik: Mechanik von Massenpunkten, Newtonsche Axiome, Energie und Arbeit, Impuls, Teilchensysteme, Drehbewegungen, Mechanik deformierbarer Körper, Fluide • Schwingungen und Wellen • Thermodynamik: Temperatur und der Nullte Hauptsatz der Thermodynamik, kinetische Gastheorie, Wärme und der erste Hauptsatz der Thermodynamik, Zweiter Hauptsatz der Thermodynamik, Wärmeübertragung • Optik: Eigenschaften des Lichts, Geometrische Optik, Interferenz und Beugung • Auswahl von Themen der Modernen Physik: Quantenmechanik und Atomphysik, Kernphysik, Physik der kondensierten Materie
6	**Lernziele und Kompetenzen**	Die Studierenden • erklären die Grundlagen der Experimentalphysik aus den Bereichen der Mechanik, Schwingungen und Wellen, Thermodynamik, Optik sowie von ausgewählten Themen der Modernen Physik • setzen die Vorlesungsinhalte mit Hilfe thematisch passender Übungsaufgaben praktisch um.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot der Technischen Fakultät oder der Naturwissenschaftlichen Fakultät Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich (90 Minuten)
11	**Berechnung der Modulnote**	schriftlich (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester

15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	P.A. Tipler, "Physik", Spektrum Akad. Verlag D. Halliday, R. Resnick, J. Walker, "Physik", Wiley-VCH F. Kuypers, "Physik für Ingenieure und Naturwissenschaftler", Wiley-VCH D. Mills, "Bachelor-Trainer Physik" Spektrum Akad. Verlag

1	**Modulbezeichnung** 66010	**Experimentalphysik II** (Experimental physics for EECE II)	**5 ECTS**
2	Lehrveranstaltungen	Vorlesung: Experimentalphysik für Medizin- und Elektrotechniker II (3 SWS)	4 ECTS
3	Lehrende	Prof. Dr. Bernhard Hensel	

4	**Modulverantwortliche/r**	Prof. Dr. Bernhard Hensel apl.Prof.Dr. Jürgen Ristein
5	**Inhalt**	*Inhaltsangabe für beide Semester* • Physikalische Größen und Messungen • Mechanik: Mechanik von Massenpunkten, Newtonsche Axiome, Energie und Arbeit, Impuls, Teilchensysteme, Drehbewegungen, Mechanik deformierbarer Körper, Fluide • Schwingungen und Wellen • Thermodynamik: Temperatur und der Nullte Hauptsatz der Thermodynamik, kinetische Gastheorie, Wärme und der erste Hauptsatz der Thermodynamik, Zweiter Hauptsatz der Thermodynamik, Wärmeübertragung • Optik: Eigenschaften des Lichts, Geometrische Optik, Interferenz und Beugung • Auswahl von Themen der Modernen Physik: Quantenmechanik und Atomphysik, Kernphysik, Physik der kondensierten Materie
6	**Lernziele und Kompetenzen**	Die Studierenden • erklären die Grundlagen der Experimentalphysik aus den Bereichen der Mechanik, Schwingungen und Wellen, Thermodynamik, Optik sowie von ausgewählten Themen der Modernen Physik • setzen die Vorlesungsinhalte mit Hilfe thematisch passender Übungsaufgaben praktisch um.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot der Technischen Fakultät oder der Naturwissenschaftlichen Fakultät Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	schriftlich (90 Minuten)
11	**Berechnung der Modulnote**	schriftlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch

16	**Literaturhinweise**	P.A. Tipler, "Physik", Spektrum Akad. Verlag D. Halliday, R. Resnick, J. Walker, "Physik", Wiley-VCH F. Kuypers, "Physik für Ingenieure und Naturwissenschaftler", Wiley-VCH D. Mills, "Bachelor-Trainer Physik" Spektrum Akad. Verlag

1	**Modulbezeichnung** 645618	**Human Computer Interaction** (Human computer interaction)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Human Computer Interaction Exercises (1 SWS) Vorlesung: Human Computer Interaction (3 SWS)	1,25 ECTS 3,75 ECTS
3	Lehrende	Madeleine Flaucher Prof. Dr. Björn Eskofier	

4	**Modulverantwortliche/r**	Prof. Dr. Björn Eskofier Madeleine Flaucher Wolfgang Mehringer Anastasiya Zakreuskaya
5	**Inhalt**	Das Modul vermittelt Konzepte, Prinzipien, Modelle, Methoden und Techniken für die effektive Entwicklung von benutzerfreundlichen Mensch-Computer-Schnittstellen. Das Thema moderner Benutzungsschnittstellen wird dabei für klassische Computer aber auch für mobile Geräte, eingebettete Systeme, Automobile und intelligente Umgebungen betrachtet. Die folgenden Themen werden im Modul behandelt: • Einführung in die Grundlagen der Mensch-Computer-Interaktion, historische Entwicklung • Entwurfsprinzipien und Modelle für moderne Benutzungsschnittstellen und interaktive Systeme • Informationsverarbeitung des Menschen, Wahrnehmung, Motorik, Eigenschaften und Fähigkeiten des Benutzers • Interaktionskonzepte und -stile, Metaphern, Normen, Regeln und Style Guides • Ein- und Ausgabegeräte, Entwurfsraum für interaktive Systeme • Analyse-, Entwurfs- und Entwicklungsmethoden und -werkzeuge für Benutzungsschnittstellen • Prototypische Realisierung und Implementierung von interaktiven Systemen, Werkzeuge • Architekturen für interaktive Systeme, User Interface Toolkits und Komponenten • Akzeptanz, Evaluationsmethoden und Qualitätssicherung Contents: The module aims to teach basic knowledge of concepts, principles, models, methods and techniques for developing highly user-friendly Human-Computer Interfaces. Beyond traditional computer systems, modern user interfaces are also discussed in the context of automobile and intelligent environments, mobile devices and embedded systems.

		This module addresses the following topics: • Introduction to the basics of Human-Computer Interaction • Design principles and models for modern user interfaces and interactive systems • Information processing of humans, perception, motor skills, properties and skills of the users • Interaction concepts, metaphors, standards, norms and style guides • In- and output devices, design space for interactive systems • Analysis-, design- and development of methodologies and tools for easy-to-use user interfaces • Prototypic implementation of interactive systems • Architectures for interactive systems, User Interface Toolkits and components • Acceptance, evaluation methods and quality assurance
6	**Lernziele und Kompetenzen**	• Studierende entwickeln ein Verständnis für Modelle, Methoden und Konzepte der Mensch-Computer-Interaktion. • Sie lernen verschiedene Ansätze für den Entwurf, die Entwicklung und Bewertung von Benutzungsschnittstellen kennen und verstehen deren Vor- und Nachteile. • Die Teilnahme an der Veranstaltung versetzt Studierende in die Lage, einen Entwicklungsprozess in der Mensch-Computer-Interaktion zu verstehen und umzusetzen. • Sie werden weiterhin in die Lage versetzt, dies vor dem Hintergrund der Informationsverarbeitungsfähigkeit, Wahrnehmung und Motorik des Benutzers zu gestalten. • Passende Methoden der Evaluation sowie Akzeptanz- und Qualitätssicherung werden erlernt. Learning Objectives and Competences: • Students develop an understanding of models, methods and concepts in the field of Human-Computer Interaction. • They learn different approaches for designing, developing and evaluating User Interfaces and their advantages and disadvantages. • Joining the course enables students to understand and execute a development process in Human-Computer Interaction. • Students will be able to do a UI evaluation by learning the basics of information processing, perception and motoric skills of the user. • Appropriate evaluation methods, as well as acceptance and quality assurance aspects, will be learned.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1

9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot von EEI und Informatik Master of Science Information and Communication Technology 20191 Wahlpflichtmodul aus INF im Schwerpunkt Media Processing Systems Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot der Technischen Fakultät oder der Naturwissenschaftlichen Fakultät Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Klausur
11	**Berechnung der Modulnote**	Klausur (100%)
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 96316	**Radar, RFID and Wireless Sensor Systems (RWS)** (keine englischsprachige Modulbezeichnung hinterlegt!)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Radar, RFID and Wireless Sensor Systems Exercises (2 SWS) Vorlesung: Radar, RFID and Wireless Sensor Systems (2 SWS)	- 5 ECTS
3	Lehrende	Dr.-Ing. Christian Carlowitz Prof. Dr.-Ing. Martin Vossiek	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Martin Vossiek
5	**Inhalt**	Radar, RFID and wireless sensor and wireless locating systems are essential for automotive advanced driver-assistance systems (ADAS), autonoumous driving and flying, robotics, industrial automation, logistics and novel human machine interfaces. Further key areas include medical electronics, building technology and cyber-physical systems. The module "Radar, RFID and Wireless Sensors" is an introduction into functional principles, building blocks, hardware and signal processing concepts and applications of modern radar, RFID, wireless sensor and real time locating systems. Covered applications include automotive radar, road and air traffic control systems, as well as robotics, industrial automation and medical technology. RWS is an identical replacement of the former module "Drahtlose Sensoren, Radar- und RFID-Systeme DSR.
6	**Lernziele und Kompetenzen**	The students • learn about the setup, function and application of wireless sensors, Radar and RFID-systems • can analyze, discuss and implement basic components and system structures, signal theory, data processing and use cases • can determine the underlying physical limitations and sources of errors • are able to analyze and create system specifications and can compare and rate the usability of wireless esnsors, Radar and RFID-systems • can create and define independently applications and system designs of RWSs
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot der Technischen Fakultät oder der Naturwissenschaftlichen Fakultät Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich

11	**Berechnung der Modulnote**	mündlich (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Sensors for Ranging and Imaging", Graham Brooker, Scitech Publishing Inc., 2009 Radar mit realer und synthetischer Apertur", H. Klausing, W. Holpp, Oldenbourg, 1999 Praxiswissen Radar und Radarsignalverarbeitung" Albrecht K. Ludloff, 2008 "RFID at ultra and super high frequencies: theory and application Dominique Paret, John Wiley & Sons, 2009. RFID-Handbuch: Grundlagen und praktische Anwendungen von Transpondern, kontaktlosen Chipkarten und NFC", Klaus Finkenzeller, Carl Hanser Verlag, 6. Auflage 2012.

1	**Modulbezeichnung** 43722	**Scientific Visualization** (keine englischsprachige Modulbezeichnung hinterlegt!)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Tutorials to Scientific Visualization (2 SWS) Vorlesung: Scientific Visualization (2 SWS)	- 5 ECTS
3	Lehrende	Prof. Dr.-Ing. Tobias Günther Xingze Tian	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Tobias Günther
5	**Inhalt**	The amount of data, generated in the pursuit of scientific discovery, keeps rapidly increasing across all major scientific disciplines. How can we make sense of large, time-dependent, high-dimensional and multi-variate data? This lecture provides an introduction into scientific visualization. Throughout the course, we cover the fundamental perception basics needed to convey information accurately. After categorizing different data types based on their dimensionality, we dive deeper into specific techniques for scalar, vector and tensor valued data. The lecture covers the following topics: • visualization design basics (data abstraction, visual encoding of information), • a review of scalar and vector calculus (differential properties, extremal and critical points), • data structures and data acquisition techniques (grids, interpolation, and differentiation), • indirect volume visualization (marching cubes and contour trees), • direct volume visualization (ray marching and Monte Carlo rendering), • elementary and line-based flow visualization (numerical integration, seeding, rendering), • surface-based flow visualization (integration, selection, rendering), • topology-based flow visualization (topological skeleton, bifurcations, feature flow fields), • feature-based flow visualization (vortices, material boundaries, Lagrangian coherent structures), • advanced methods (tensor visualization, uncertainty, ensembles)
6	**Lernziele und Kompetenzen**	Students are able to: • use perception basics to select appropriate visualization methods • classify data and select appropriate visualization techniques • calculate differential properties of scalar and vector fields • identify features in scalar and vector-valued data • implement numerical extraction algorithms • learn the advantages and disadvantages of common visualization techniques

7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot der Technischen Fakultät oder der Naturwissenschaftlichen Fakultät Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Variabel
11	**Berechnung der Modulnote**	Variabel (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 43960	**Self-Organized Networks** (keine englischsprachige Modulbezeichnung hinterlegt!)	**5 ECTS**
2	Lehrveranstaltungen	Übung: Self-organized networks (Ex) (2 SWS) Vorlesung: Self-organized networks (2 SWS)	- -
3	Lehrende	Dr.-Ing. Mehdi Harounabadi	

4	**Modulverantwortliche/r**	Dr.-Ing. Mehdi Harounabadi
5	**Inhalt**	keine Inhaltsbeschreibung hinterlegt!
6	**Lernziele und Kompetenzen**	keine Beschreibung der Lernziele und Kompetenzen hinterlegt!
7	**Voraussetzungen für die Teilnahme**	Self-organized networks (2,5 ECTS) Self-organization refers to processes in physical and biological systems which form patterns by interactions of internal parts of the system and/or their interactions with the environment without any form of external interventions. The resulting organization of systems in this case is fully decentralized and distributed over their parts. There are numerous examples of self-organizing systems e.g. flock of birds, neural learning, evolutionary computation, swarm of robots, and self-organizing networks. The principles of self-organization are applied in different communication technologies such as wired and wireless networks. Transport Control Protocol (TCP), Carrier Sense Multiple Access (CSMA), Adaptive Modulation and Coding (AMC) are some examples of self-organized protocols. Besides, Self-Organizing Network (SON) has been defined by 3GPP as the standardization organization of cellular networks. SON refers to the functionality of cellular systems (e.g. 4G and 5G) which enables the system to perform self-configuration, self-optimization and self-healing. Artificial intelligence and mostly machine learning algorithms play an important role in the implementation of SON. This course aims to give you an insight to the self-organization systems and specially self-organizing networks and illustrate the importance of AI and ML in their implementation. The content of this course includes: • o Self-organizing features and protocols in communication networks o In wired networks o In wireless networks § IEEE 802.11 § Ad hoc and sensor networks § UAV networks o Self-Organizing Networks (SON) in 4G and 5G § Self-configuration § Self-optimization § Self-healing Methods for implementation of self-organizing systems o Bio-inspired methods o Artificial intelligence The final exam is in a written form where students should answer to questions based on the lectures, slides and their understating and analysis from the content of the course.
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Wahlmodule aus dem Angebot der Technischen Fakultät oder der Naturwissenschaftlichen Fakultät Master of Science Information and Communication Technology 20191

		Wahlpflichtmodul aus INF im Schwerpunkt Networks and Digital Communication Master of Science Information and Communication Technology 20191 Wahlmodule aus dem Angebot der Technischen Fakultät oder der Naturwissenschaftlichen Fakultät Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Variabel
11	**Berechnung der Modulnote**	Variabel (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 90 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Literature review on self-organizing networks (2,5 ECTS) The exercise of this course includes a literature review research project where students work individually on a relevant topic. The steps to accomplish the research project are as follows: A. Select a topic relevant to self-organizing networks and register it by email B. Search for the relevant papers and make a list of papers C. Study the papers and prepare a summary D. Present the outcomes

Praktikum oder Projektarbeit

1	**Modulbezeichnung** 894349	**Audio Processing Laboratory** (Audio processing laboratory)	**2,5 ECTS**
2	Lehrveranstaltungen	Praktikum: Audio Processing Laboratory (2 SWS)	2,5 ECTS
3	Lehrende	Prof. Dr.-Ing. Jürgen Herre Prof. Dr. Emanuël Habets Prof. Dr.-Ing. Bernd Edler Prof. Dr. Nils Peters Prof. Dr. Meinard Müller	

4	**Modulverantwortliche/r**	Prof. Dr. Meinard Müller
5	**Inhalt**	This lab course offers a general introduction to Python and possibly also to other languages (MATLAB, R, ...). In particular, functions, transforms, and algorithms that are important for analyzing and processing audio signals are covered. After a general part, the lab course will allow the participants to delve into a more specific application within audio and acoustic signal processing.
6	**Lernziele und Kompetenzen**	The goal of this lab course is to acquire a deeper understanding of audio processing techniques by experimenting with, modifying and extending existing code. Furthermore, programming skills in Python and possibly also in other languages (MATLAB, R, ...) are acquired. The students understand and implement computer programs for specific experiments described in the script accompanying the lab. They test and evaluate their programs by conducting a series of experiments within the field of audio signal processing. They understand the requirements of practical realizations, synthesize a solution for a given problem, and apply advanced disciplinary knowledge and skills in signal processing. The students evaluate and interpret results by applying various visualization techniques and statistical methods. They collaborate with fellows students, discuss their solutions, give feedback to each other, and reflect upon the underlying theory as well as implementation issues.
7	**Voraussetzungen für die Teilnahme**	This lab course requires a good understanding of basic principles in signal processing and some basic programming skills. Furthermore, it is beneficial to have some background in one of the more specific topics

		offered by the International Audio Laboratories Erlangen.
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Praktikum oder Projektarbeit Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	
11	**Berechnung der Modulnote**	
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 97500	**Laborpraktikum Digitaler ASIC-Entwurf** (Laboratory: Digital ASIC design)	**2,5 ECTS**
2	Lehrveranstaltungen	Praktikum: Praktikum Digitaler ASIC-Entwurf (Blockpraktikum) (3 SWS)	2,5 ECTS
3	Lehrende	Jürgen Frickel	

4	**Modulverantwortliche/r**	Jürgen Frickel
5	**Inhalt**	In diesem Praktikum wird jeweils in Zweiergruppen eine komplexe digitale Schaltung für ein FPGA entworfen, Entwurfsziel sind hardware- und grafikorientierte Anwendungen, die ohne Prozessor/Software als reine Hardware-Lösung entwickelt und realisiert werden müssen. Hierzu müssen die Teilnehmer zu Beginn eine rudimentär vorgegebene Systemspezifikation analysieren, verbessern und verfeinern, eine Systemidee entwickeln, das geplante System partitionieren und auf Module aufteilen. Die angestrebten Lösungen werden in regelmässigen Kurzvorträgen mit der Gesamtgruppe dikutiert. Die in der Hardware-Beschreibungssprache VHDL entworfenen Module können dann mit Hilfe des Entwurfswerkzeugs (aktuell: XILINX Vivado) spezifiziert, simuliert, verifiziert und abschließend für die Ziel-Hardware synthetisiert werden. Hierbei ist außer der Schnittstellenproblematik zwischen den Modulen auch der Aspekt des simulations- und testfreundlichen Entwurfs zu beachten. Mit einer vorhandenen FPGA-Testumgebung (Evaluation/Education Board) wird der Funktions- und Systemtest auf realer Hardware durchgeführt. Nach der Verifikation und Zusammenschaltung aller Module erfolgt ein abschließender Funktionstest und Bewertung (Größe, Geschwindigkeit, Funktionsumfang, Effizienz, etc.) der Schaltung in Form einer Demonstration vor der Gesamtgruppe.
6	**Lernziele und Kompetenzen**	Fachkompetenz Anwenden Die Studierenden setzen die vorab (in einer anderen LV) erlernte Hardware-Beschreibungssprache VHDL in ihrem vollen Umfang zur Spezifikation und Implementierung eines komplexen, digitalen Systems ein. Analysieren Die Studierenden analysieren ein nur rudimentär beschriebenes digitales mikroelektronisches System, untersuchen mögliche

		Lösungsansätze und strukturieren diese Lösungsansätze in handhabbare Module. Evaluieren (Beurteilen) Die Studierenden diskutieren und bewerten im Rahmen von Kurzvorträgen eigene und fremde Lösungsvorschläge zum Systementwurf, vergleichen diese nach eigenen Kriterien, und wählen dann hiermit die besten Lösungen zur Realisierung aus. Die Stduierenden bewerten nach Fertigstellung des Systementwurfs nach verschiedenen Kriterien (Größe, Geschwindigkeit=längster Pfad, Performance, Ästhetik, Code-Qualität) ihre und die anderen Entwürfe. Erschaffen Wegen der sehr knappen Auslegung der gegebenen Spezifikation der Systembeschreibung konzipieren die Studierenden ganz eigene, individuelle Lösungen für die Funktionsmodule und das Gesamtsystem. Lern- bzw. Methodenkompetenz Die Studierenden erlernen die Methodik zur Transformation einer Systemidee in eine digitale Realisierung. Sozialkompetenz Studierende erlernen, Problemstellungen in Gruppenarbeit gemeinsam zu lösen. Die Studierenden erarbeiten ihre Lösungen in Zweiergruppen und erläutern bzw. verteidigen diese in Kurzvorträgen gegenüber der Gesamtgruppe.
7	**Voraussetzungen für die Teilnahme**	• Digitaltechnik (oder ähnliche Grundlagen-LV, z.B. TI-1) • V+Ü "Hardware-Beschreibungssprache VHDL" (oder andere gleichwertige LVen) • oder: nachgewiesene gute Kenntnisse/praktische Erfahrungen in VHDL, z.B. durch Praktikanten- oder Werkstudententätigkeit, intensives Eigenstudium, etc.
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Praktikum oder Projektarbeit Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	
11	**Berechnung der Modulnote**	
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 45 h Eigenstudium: 30 h
14	**Dauer des Moduls**	1 Semester

15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Frickel J.; Skript der LV "Hardware-Beschreibungssprache VHDL" Xilinx; Handbuch Xilinx Vivado Lehmann G.; Wunder B.; Selz M.: Schaltungsdesign mit VHDL. Poing Franzis 1994 Bleck Andreas: Praktikum des modernen VLSI-Entwurfs. Stuttgart Teubner 1996

1	**Modulbezeichnung** 97530	**Laborpraktikum Eingebettete Mikrocontroller-Systeme (PEMSY)** (Laborpraktikum Eingebettete Mikrocontroller-Systeme (PEMSY))	**2,5 ECTS**
2	Lehrveranstaltungen	Praktikum: Praktikum Eingebettete Mikrocontroller-Systeme (Blockpraktikum) (3 SWS) Praktikum: Praktikum Eingebettete Mikrocontroller-Systeme (semesterbegleitend) (3 SWS)	2,5 ECTS 2,5 ECTS
3	Lehrende	Sebastian Klob	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Albert Heuberger
5	**Inhalt**	Dieses Praktikum führt die Studierenden in das Gebiet der eingebetteten Mikrocontroller-Systeme ein. Basierend auf dem Stoff der Vorlesungen Digitaltechnik, Schaltungstechnik und Systemprogrammierung bearbeiten die Teilnehmer/-innen eine Problemstellung, die mittels einer Maschine" gelöst werden soll. Zusätzlich notwendiges Wissen wird vermittelt, damit diese Maschine" in 2er-Gruppen weitgehend selbständig aufgebaut werden kann. Verwendet wird eine vom Lehrstuhl selbst entwickelte Platine auf Basis des AVR ATmega32 mit einem LCD-Display und einem ISM-Funkmodul. Schrittweise erfolgt der Lötaufbau des USB-Programmieradapters und der Hardware-Plattform mit Blick auf das zu realisierende Gesamtsystem. Während die Programmiermodule immer umfangreicher werden, wird mit zunehmender Erfahrung der Teilnehmer/-innen das System auf einem Lochrasterfeld durch eigene Schaltungen ergänzt und erweitert. Als Besonderheit darf die entwickelte Maschine" nach dem Ende des Praktikums von den Teilnehmern behalten werden. Programmiert wird konsequent in C (und Inline-Assembler) und verwendet werden ausschließlich frei verfügbare Entwicklungshilfsmittel. Für einen kontinuierlichen Entwicklungsfortschritt im Zusammenspiel mit dem Hardwareaufbau ist es hierbei unerlässlich das bereits gewisse Erfahrungen in dieser Programmiersprache bestehen. Nach Abschluss des Praktikums sind die Teilnehmer/-innen in der Lage ein Mikrocontroller-System für den Einsatz in einem Mess- oder Steuerungsprojekt aufzubauen, effektiv zu programmieren und Daten über eine Kurzstreckenfunkübertragung auszutauschen.
6	**Lernziele und Kompetenzen**	Nach der Teilnahme an der Modulveranstaltung PEMSY" sind die Studierenden in der Lage die Konzepte und Verfahren der Mikrocontroller-Programmierung anzuwenden. Die Studierenden lernen dabei eigene Software für Mikrocontroller zu entwickeln. Sie lernen die Funktionsweise und den Einsatzzweck diverser Komponenten wie z.B. Strukturelemente und On-Chip-Peripherie am Beispiel des Mikrocontrollers ATmega32 zu verstehen. Dabei analysieren sie deren Zeitverhalten, entwickeln Methoden zum Anschluss von Peripherie-Elementen und bewerten Wechselwirkungen zwischen Hard- und Software.

		Die Studierenden sind weiterhin nach der Veranstaltung in der Lage, eine Entwicklungsumgebung für Mikrocontroller anzuwenden, sie lernen folgende Aspekte zu verstehen: • Software-Entwicklung unter Linux • Erzeugung von lauffähigem Code auf einem Mikrocontroller • Übertragung von Binärcode zum Mikrocontroller Im Rahmen des Aufbaus zweier Platinen werden zusätzlich folgende Inhalte vermittelt: • Löten an bedrahteten Bauelementen • Aufbau von einer Programmieradapterschaltung • Aufbau von einer Entwicklungsplattform mit integriertem Mikrocontroller und LCD-Display • Systematische Fehlersuche Durch die verwendeten Hard- und Software-Komponenten und generell gültigen Methodiken im Praktikum sind die erlernten Inhalte auch auf andere Mikrocontroller-Architekturen und Entwicklungssysteme übertragbar. Durch die Aufgabenstellungen des Praktikums sind die Studierenden später in der Lage, folgende Kommunikationsschnittstellen zu verstehen und eigene Treiber dafür zu entwickeln: • Serielle synchrone Datenübertragung (SPI) • serielle asynchrone Datenübertragung (UART) • parallele bidirektionale Datenübertragung über einen Bus Weiterhin sind die Studierenden nach dem Praktikum in der Lage folgende Kommunikationsprotokolle anzuwenden: • Befehlssatz des LCD Controllers HD44780 • Befehlssatz eines ISM Funkmoduls
7	**Voraussetzungen für die Teilnahme**	• Die Beherrschung der Inhalte von Lehrveranstaltungen in einem ingenieurwissenschaftlichen Grundstudium, die in die Grundlagen der Informatik und Elektrotechnik einführen • Kenntnisse in der Programmiersprache C • Grundverständnis von Booleschen Operationen • Englischkenntnisse • Deutschkenntnisse
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Praktikum oder Projektarbeit Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	
11	**Berechnung der Modulnote**	
12	**Turnus des Angebots**	in jedem Semester

13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 45 h Eigenstudium: 30 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Kernighan / Ritchie: The C Programming Language https://www.like.tf.fau.de/lehre/lehrveranstaltungen/

1	**Modulbezeichnung** 97640	**Laborpraktikum Mobilkommunikation** (Laboratory course: Mobile communication)	**2,5 ECTS**
2	Lehrveranstaltungen	Praktikum: Praktikum Mobilkommunikation (3 SWS)	2,5 ECTS
3	Lehrende	Adela Vagollari Yifei Wu Prof. Dr. Wolfgang Gerstacker Dr.-Ing. Clemens Stierstorfer	

4	**Modulverantwortliche/r**	Prof. Dr. Wolfgang Gerstacker

5	**Inhalt**	Experiments
		•
		  ◦ Characteristics of real mobile radio channels such as distortions and time variability
		  ◦ models for mobile radio channels
		  ◦ effects on the performance of a mobile radio system
		•
		  ◦ Principles of different equalization methods
		  ◦ equalizer design for GSM / EDGE
		  ◦ simulation of trellis-based equalizers and visualization of the results
		•
		  ◦ Principle of OFDM
		  ◦ implementation-relevant aspects such as nonlinearities and peak-to-average-power ratio
		  ◦ synchronization and equalization
		• MIMO Transmission (2 experiments)
		
		Versuche
		•
		  ◦ Eigenschaften realer Mobilfunkkanäle wie Verzerrungen und Zeitvarianz,
		  ◦ Modelle für Mobilfunkkanäle
		  ◦ Auswirkungen auf die Leistungsfähigkeit eines Mobilfunksystems
		•
		  ◦ Prinzipien verschiedener Entzerrverfahren
		  ◦ Entzerrerdesign für GSM/EDGE
		  ◦ Simulation von trellisbasierten Entzerrern und Visualisierung der Ergebnisse
		•
		  ◦ Prinzip von OFDM
		  ◦ implementierungsrelevante Aspekte wie Nichtlinearitäten und Spitzenwertfaktor
		  ◦ Synchronisation und Entzerrung

		• MIMO Übertragung (2 Versuche)
6	**Lernziele und Kompetenzen**	The students • describe the characteristics of real mobile radio channels, • explain the principles of OFDM and MIMO transmission systems, • implement equalization and adaptation procedures in Matlab, • perform radio network simulations, • learn to develop program code, • work together in a small team. Die Studierenden • charakterisieren die Eigenschaften realer Mobilfunkkanäle, • erklären die Funktionsweise von OFDM- und MIMO-Übertragungssystemen, • implementieren Entzerrungs- und Adaptionsverfahren in Matlab, • führen Funknetzsimulationen durch, • erlernen Programmcode eingeständig zu entwickeln, • arbeiten zielorientiert in einem kleinen Team zusammen.
7	**Voraussetzungen für die Teilnahme**	Vorkenntnisse aus Vorlesungen zu Nachrichtenübertragung (Communications) und Systemtheorie (Signals and Systems); Inhalte des Moduls "Mobile Communications" sind erforderliche Voraussetzung für eine sinnvolle Teilnahme;
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Praktikum oder Projektarbeit Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	• There are 8 experiments to be completed as well as an introduction to Matlab. These are described in the course materials. • Each experiment is to be prepared in writing at home. The preparation is checked and evaluated (sufficient/not sufficient) at the beginning of each experiment. • The results of each experiment are to be kept on the experimental computers during the execution of the experiment (programming tasks) and are checked at the end of the experiment (sufficient/not sufficient). Measurement results are to be documented in writing. • To pass the course, 8 sufficient experiment preparations and 8 sufficient experiment executions are required.

		• Es sind 8 Versuche sowie eine Einführung in Matlab zu absolvieren. Diese sind in den Kursunterlagen beschrieben. • Jeder Versuch ist zu Hause schriftlich vorzubereiten. Die Vorbereitung wird zu Beginn eines jeden Versuchs überprüft und bewertet (ausreichend/nicht ausreichend). • Die Ergebnisse eines jeden Versuchs sind während der Versuchsdurchführung auf den Versuchsrechnern vorzuhalten (Programmieraufgaben) und werden zum Abschluss des Versuchs überprüft (ausreichend/nicht ausreichend). Messergebnisse sind schriftlich zu dokumentieren. • Zum Bestehen des Praktikums sind 8 ausreichende Versuchsvorbereitungen und 8 ausreichende Versuchsdurchführungen notwendig.
11	**Berechnung der Modulnote**	
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 35 h Eigenstudium: 40 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Skriptum zum Praktikum Mobilkommunikation

1	**Modulbezeichnung** 97720	**Laborpraktikum Systematischer Entwurf programmierbarer Logikbausteine** (Laboratory course: Systematic design with programmable logic devices (PLD))	**2,5 ECTS**
2	Lehrveranstaltungen	Praktikum: Praktikum für systematischen Entwurf programmierbarer Logikbausteine (3 SWS)	2,5 ECTS
3	Lehrende	Torsten Reißland	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Robert Weigel
5	**Inhalt**	In diesem Praktikum wird eine Einführung in den systematischen Entwurf Programmierbarer Logikbausteine geben. Außerdem werden Grundkenntnisse in der Hardwarebeschreibungs- und Programmiersprache VHDL vermittelt. Auch alternative Eingabeformate, wie die Fuse-Map oder über Einfügen von Schaltplänen werden vorgestellt. Nach der Simulation werden die erstellten Programme auf realer Hardware, einem FPGA-Board, per In-System-Programmierung" getestet. Es besteht Anwesenheitspflicht.
6	**Lernziele und Kompetenzen**	• Die Studierenden erlangen grundlegende Kenntnisse in VHDL • Die Studierenden verstehen die der Hardware-Programmierung zu Grunde liegenden Systematik • Die Studierenden analysieren und vergleichen unterschiedliche Ansätze von Hardware-Beschreibungsmöglichkeiten • Die Studierenden vertiefen die Grundlagen der Digitaltechnik • Die Studierenden erlangen die Fähigkeit, einfache Problemstellungen systematisch in eine Hardwarebeschreibung umzusetzen
7	**Voraussetzungen für die Teilnahme**	Vorkenntnisse: Grundlagen digitaler Schaltungen
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Praktikum oder Projektarbeit Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	
11	**Berechnung der Modulnote**	
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 45 h Eigenstudium: 30 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Tietze/Schenk: Halbleiter-Schaltungstechnik, Springer Verlag

1	**Modulbezeichnung** 182405	**Praktikum Architekturen der digitalen Signalverarbeitung** (Laboratory architectures for digital signal processing)	**2,5 ECTS**
2	Lehrveranstaltungen	Praktikum: Praktikum Architekturen der digitalen Signalverarbeitung (3 SWS)	2,5 ECTS
3	Lehrende	Christof Pfannenmüller	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Georg Fischer
5	**Inhalt**	• Aufbau einer akustischen FSK Datenverbindung • Einführung in die VHDL Programmierung eines FPGAs • Erzeugung einer PRBS Sequenz • Effiziente Implementierung eines Sinusgenerators mit Hilfe des Cordic Algorithmus • Digitale Filterung • Demodulation/Detektion
6	**Lernziele und Kompetenzen**	Die Studierenden erlangen Grundlagenkenntnisse in der Programmierung mit MATLAB und VHDL Die Studierenden sind in der Lage, eine digitales Datenübertragungssystem vom Sender bis zum Empfänger theoretisch zu konzeptionieren, in MATLAB zu simulieren und praktisch in VHDL auf einem FPGA umzusetzen Die Studierenden erhalten die theoretische und praktische Fähigkeit, digitale Signale zu definieren, zu verarbeiten, digitale Filter zu erzeugen und Signale mit diesen zu manipulieren Die Studierenden verstehen die Schnittstelle zwischen der digitalen und analogen Ebene und sind in der Lage, diese Schnittstellen auf einem FPGA Evaluation Board zu verwenden
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Praktikum oder Projektarbeit Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	
11	**Berechnung der Modulnote**	
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 45 h Eigenstudium: 30 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 194239	**Praktikum Bild- und Videosignalverarbeitung auf eingebetteten Plattformen** (Lab course image and video processing on embedded platforms)	**2,5 ECTS**
2	Lehrveranstaltungen	Praktikum: Laborpraktikum Bild- und Videosignalverarbeitung auf eingebetteten Plattformen (0 SWS)	2,5 ECTS
3	Lehrende	PD Dr.Ing. Jürgen Seiler Viktoria Heimann	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Andre Kaup
5	**Inhalt**	Betrachtet man Anwendungen der Bild- und Videosignalverarbeitung stellt man fest, dass viele davon auf mobilen Plattformen ablaufen. Die dort verwendeten Systeme haben aber häufig nur eine reduzierte Leistungsfähigkeit und müssen besonders auf den Energieverbrauch achten. Nichtsdestotrotz sind aber auch einfache, mobile Systeme wie Smartphones oder Tablets in der Lage, anspruchsvolle Signalverarbeitungsaufgaben für Bild- und Videosignale durchzuführen. Dies umfasst zum Beispiel die Codierung von Bildern und Videos, aber auch die Erzeugung von Panoramen oder die Berechnung von Bildern mit hohem Dynamikumfang. Das Praktikum Bild- und Videosignalverarbeitung auf eingebetteten Plattformen" soll die Herausforderung, die mit einer Verarbeitung dieser Signale auf eingebetteten Plattformen einhergehen genauer vermitteln und es wird aufgezeigt, wie man selbst auf Plattformen mit eingeschränkter Leistungsfähigkeit entsprechende Algorithmen umsetzen kann. Hierzu werden in dem Praktikum Raspberry Pis als Plattform verwendet und die Programmierung erfolgt in Python. Die Versuche umfassen den Aufbau und die Inbetriebnahme der eingebetteten Plattform, eine Einführung in Python und in die grundlegenden Prozesse der Bild- und Videosignalverarbeitung. Weitere Versuchsinhalte sind die Anbindung einer Kamera, Bildsignalverarbeitungsprozesse mit der Kamera und die Implementierung verschiedener digitaler Filter. Das Praktikum beinhaltet außerdem verschiedene Anwendungen computergestützten Sehens (Computer Vision). Die Detektion von Merkmalen und Objekten in Bildern und Videos werden einführend behandelt und aktuelle Computer Vision Anwendungen, wie die Erstellung eines Panoramas werden betrachtet. *Content*: Today, many image and video signal processing applications are running on embedded systems. However, the computational power and the energy storage is a limiting demand for embedded systems. Nevertheless, daily mobile devices like smartphone and tablet are able to perform signal processing tasks for image and video signals, for

		example coding of images and videos, the creation of a panorama or the calculation of images with high dynamic range. The image and video signal processing on embedded systems lab course should show the challenges that occur while handling with such mobile devices and the implementation of such algorithm on an embedded system. Therefore, Raspberry Pis as embedded systems and Python as coding language is used in the laboratory. The experiments include the setup of the Raspberry Pi, an introduction to Python and an introduction to image and video signal processing. In addition, a camera will be connected, signal processing will be done with the camera and digital filters are implemented. Moreover, the laboratory includes different computer vision applications like the creation of a panorama.
6	**Lernziele und Kompetenzen**	Die Studierenden • verstehen die Herausforderungen von eingebetteten Plattformen • wenden die Programmiersprache Python für Bild- und Videosignalverarbeitunsalgorithmen an • erzeugen funktionsfähige Programme mit der Programmiersprache Python • beurteilen die Funktionsblöcke von Computer Vision-Algorithmen • bewerten die von ihnen erstellten Programme durch subjektive und objektive Vergleiche • reflektieren den Lernprozess während des Praktikums. The students • understand the challenges of the embedded system • make use of the coding language Python for image and video signal processing algorithms • implement functional programs with Python • evaluate the blocks of computer vision algorithms • evaluate the self-implemented programs by subjective and objective comparison • reflect the learning process in the laboratory.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Praktikum oder Projektarbeit Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	
11	**Berechnung der Modulnote**	
12	**Turnus des Angebots**	nur im Wintersemester

13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 15 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	Das Skript zum Praktikum Image and video signal processing on embedded platforms" wird in der Einführungsveranstaltung ausgegeben. The laboratory script "Image and video signal processing on embedded platforms will be handed out in the first session.

1	**Modulbezeichnung** 92356	**Praktikum Communications Systems Design** (keine englischsprachige Modulbezeichnung hinterlegt!)	**2,5 ECTS**
2	Lehrveranstaltungen	Praktikum: Praktikum Communications Systems Design (2 SWS)	2,5 ECTS
3	Lehrende	Christof Pfannenmüller Christof Pfannenmüller Arslan Ali Arslan Ali Prof. Dr.-Ing. Georg Fischer Christof Pfannenmüller Christof Pfannenmüller	

4	**Modulverantwortliche/r**	Arslan Ali Prof. Dr.-Ing. Georg Fischer
5	**Inhalt**	Learning based on LabVIEW communications and NI USRP: Introduction to USRP including hardware blocks of Tx/Rx chains Getting familiar with LabVIEW communications environment and controlling VIs (Panel, diagram, etc.) and fundamentals of LabVIEW programming: data types, arrays, flow control (for/while loop), clusters, case structures, signal sources, sinks, signal processing tools, filters, time/ frequency domain analysis, etc. Transmission and reception of analog modulation schemes: AM/DSB-SC and FM Implementation of digital modulation schemes: ASK, FSK, BPSK, QPSK, 16-QAM, etc. Digital Tx/Rx: symbol mapping, upsampling/downsampling, pulse shaping (rectangular, Gaussian, RRC), matched filtering, pulse alignment, synchronization, and detection Phase synchronization, FDM and image rejection algorithm Eye diagram analysis: ISI, clock jitter, optimal sampling time, detection threshold Power control for over-the-air transmission in sub-6 GHz ISM bands and analysis on fading and multipath propagation effects Channel estimation, equalization (decision directed, linear LS, adaptive LMS), modelling: coherence bandwidth and propagation delay Learning based on MATLAB and USRPs (Communications toolbox and SDR support packages):

		OFDM Tx/Rx with frequency domain equalization (FDE) and synchronization (training sequence and frame detection) LTE downlink transmission (MIMO) including system information blocks (SIB) and spectrum analysis including estimation/calibration of carrier frequency offset (CFO) Impairments/distortion analysis: ACPR, EVM tool: IQ offset errors, phase noise, PA nonlinearity, etc. Learning based on GNU Radio and RTL-SDR: Introduction to GNU Radio with RF prototyping demonstration Spectrum analyzer implementation: RBW, VBW, sweep time, and phase noise Small Project/assignment for students
6	**Lernziele und Kompetenzen**	Students Can bridge the gap between communications theory, analog/digital baseband, and RF design Can develop quick and flexible prototypes for real-time communications systems and standards using SDR solutions Can determine the design parameters and assess the interaction between various analog and digital parts Can create efficient Tx/Rx programs and signal processing algorithms in LabVIEW, MATLAB, and GNU Radio Can implement channel estimation and equalization algorithms in TDD and FDD systems Can demonstrate MIMO and OFDM based systems like LTE and beyond Can quantify and evaluate system performance using EVM and impairments analysis
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Praktikum oder Projektarbeit Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Praktikumsleistung

11	**Berechnung der Modulnote**	Praktikumsleistung (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 45 h Eigenstudium: 30 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 293179	**Praktikum Digitale Übertragung** (Digital communications lab)	**2,5 ECTS**
2	Lehrveranstaltungen	Praktikum: Praktikum Digitale Übertragung (3 SWS)	2,5 ECTS
3	Lehrende	Dr.-Ing. Clemens Stierstorfer	

4	**Modulverantwortliche/r**	Dr.-Ing. Clemens Stierstorfer
5	**Inhalt**	• 1 Digital Transmission of Data 1.1 Introduction, Background, Motivation 1.2 Purpose 1.3 Lab Environment 1.3.1 Transmitter 1.3.2 Receiver 1.4 Lab Exercises 1.4.1 Signal Generation at the Transmitter 1.4.2 (Coherent) Receivers for Pulse Amplitude Modulation 1.4.3 Transmission over the AWGN Channel • 2 Implementation of Transmitter and Receiver in Matlab 2.1 Introduction, Background, Motivation 2.2 Purpose 2.3 Lab Environment 2.3.1 Oversampling factor 2.3.2 Transmitter 2.3.3 Channel 2.3.4 Receiver 2.4 Lab Exercises 2.4.1 Transmitter 2.4.2 Channel 2.4.3 Receiver 2.4.4 BER calculation • 3 Variants of PAM-Transmission Schemes 3.1 Introduction, Background, Motivation 3.2 Purpose 3.3 Lab Environment 3.4 Lab Exercises 3.4.1 Basic Pulse Shape 3.4.2 Offset-QAM 3.4.3 Gaussian Minimum Shift-Keying 3.4.4 "Carrierless Amplitude and Phase Modulation • 4 OFDM 4.1 Introduction, Background, Motivation 4.1.1 Orthogonal Frequency-Division Multiplexing 4.1.2 Bit Loading 4.2 Purpose 4.3 Lab Environment 4.4 Lab Exercises 4.4.1 OFDM Transmitter 4.4.2 OFDM Receiver 4.4.3 Bit Loading • 5 Signal Space Representation 5.1 Introduction, Background, Motivation 5.2 Purpose 5.3 Lab Environment 5.4 Signal Space Representation 5.4.1 Orthogonality 5.4.2 Orthogonalization 5.5 Lab Exercises 5.5.1 Transmission with signal elements 5.5.2 Gram-Schmidt Procedure 5.5.3 Frequency Shift Keying • 6 Signal Processing in MIMO Systems 6.1 Introduction, Background, Motivation 6.2 Lab Environment 6.3 Lab Exercises 6.3.1 System Model 6.3.2 SISO 6.3.3 SIMO 6.3.4 MIMO
6	**Lernziele und Kompetenzen**	• Die Studierenden vertiefen und erweitern ihre Kenntnisse der digitalen Nachrichtenübertragungsverfahren und der zugehörigen mathematischen Grundlagen anhand von Laborversuchen. Sie analysieren die Eigenschaften digitaler Pulsamplitudenmodulation und Varianten digitaler PAM. Dazu erzeugen sie im Labor mit der zur Verfügung gestellten Ausrüstung Sendesignale, die sie mit Hilfe üblicher Messgeräte (Oszilloskop, Effektivwertmesser) analysieren. Sie bauen Übertragunsstrecken für diese PAM-Verfahren auf und untersuchen die Effekte auf Empfängerseite. Sie bestimmen Störabstände, Fehlerraten usw. • Des weiteren setzen die Studierenden ihre Kenntnisse der PAM-Übertragungsverfahren in selbst erstellte MATLAB-Routinen um, die die Simulation einer kompletten PAM-Übertragung mit Sender, Kanal und Empfänger am Rechner modellieren. In einem weiteren Versuch ergänzen die

Studierenden dieses Modell um eine OFDM-Übertragung und analysieren die Funktionsweisen von OFDM-Sendern und -empfängern. Sie untersuchen die Arbeitsweise von Ladealgorithmen bei OFDM-Systemen und implementieren diese in MATLAB.

- Die Studierenden verdeutlichen sich das Konzept der Signalraumdarstellung in der digitalen Übertragung und implementieren ein beispielhaftes System in MATLAB. Sie erstellen Routinen zur Gram-Schmidt-Orthogonalisierung und zur FSK-Übertragung in MATLAB.
- Die Studierenden analysieren einfache MIMO-Szenarien und implementieren entsprechende Empfängeralgorithemen.
- Die Studierenden bereiten die Bearbeitung der Versuche im Labor anhand der ausgegeben Unterlagen und den Unterlagen zum Modul "Digitale Übertragung selbständig vor. Sie sind in der Lage, die für den jeweiligen Versuch notwendigen theoretischen Kenntnisse vor und während des Versuchs zu erklären und zur Lösung der Laboraufgaben und vorbereitenden Hausaufgaben einzusetzen. Sie dokumentieren die durchgeführten Versuche selbständig in ihren Unterlagen, so dass die Nachvollziehbarkeit der Arbeiten jederzeit gegeben ist. Die Arbeit im Labor organisieren sie in Kleingruppen (2-3 Personen) selbst. Sie erkennen die Notwendigkeit gewissenhafter Vorbereitung der Lerninhalte und disziplinierter Arbeitsweise im Labor.
- Die Unterrichtssprache ist wahlweise Deutsch oder Englisch. Unterlagen werden ausschließlich auf Englisch zur Verfügung gestellt, weswegen die Studierenden die englischen Fachtermini kennen und nutzen.

- Students deepen and extend their knowledge of digital message transmission methods and the associated mathematical principles by means of laboratory experiments. They analyze the properties of digital pulse amplitude modulation and variants of digital PAM. To this end, they generate transmit signals in the laboratory using the equipment provided and analyze them with the aid of standard measuring instruments (oscilloscope, rms meter). They build transmission links for these PAM methods and investigate the effects on the receiver side. They determine signal-to-noise ratios, error rates, etc.
- Furthermore, the students implement their knowledge of the PAM transmission methods in self-created MATLAB routines, which model the simulation of a complete PAM transmission with transmitter, channel and receiver on the computer. In another experiment, students add an OFDM transmission to this model and analyze the operation of OFDM transmitters and

		receivers. They investigate the operation of loading algorithms in OFDM systems and implement them in MATLAB. • Students clarify the concept of signal space representation in digital transmission and implement an example system in MATLAB. They create routines for Gram-Schmidt orthogonalization and FSK transmission in MATLAB. Students analyze simple MIMO scenarios and implement corresponding receiver algorithms. • The students independently prepare the experiments in the laboratory using the issued documents and the documents for the module "Digital Transmission". They are able to explain the theoretical knowledge required for the respective experiment before and during the experiment and use it to solve the laboratory tasks and preparatory homework. They independently document the experiments carried out in their records so that the supervisors can trace the work at any time. They organize the work in the laboratory themselves in small groups (2-3 persons). They recognize the necessity of certain preparation of the learning content and disciplined working methods in the laboratory. • The language of instruction is either German or English. Documents are provided exclusively in English, which is why the students know and use the English technical terms.
7	**Voraussetzungen für die Teilnahme**	Das Praktikum richtet sich ausschließlich an Studierende, die das Moduls "Digitale Übertragung bereits absolviert haben oder es parallel zum Praktikum belegen. Die Inhalte dieses Moduls sind unabdingbare Grundlage und werden von den Studierenden beherrscht, d.h., sie können die entsprechenden Zusammenhänge erklären, Problemstellungen mathematisch formulieren und benötigte Größen berechnen. Grundlegende Kenntnisse der Software MATLAB sind notwendig (bspw. aus "Software für die Mathematik" oder "Simulationstools"). The lab course is aimed exclusively at students who have already completed the "Digital Transmission" module or who are taking it in parallel with the lab course. The contents of this module are an indispensable basis and are mastered by the students, i.e. they can explain the corresponding relationships, formulate problems mathematically and calculate required quantities. Basic mastery of the MATLAB software is necessary
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Praktikum oder Projektarbeit Master of Science Information and Communication Technology 20222

10	**Studien- und Prüfungsleistungen**	• Es sind 5 Versuche zu absolvieren. Diese sind in den Kursunterlagen beschrieben. • Jeder Versuch ist zu Hause schriftlich vorzubereiten. Die Vorbereitung wird zu Beginn eines jeden Versuch überprüft und bewertet (ausreichend/nicht ausreichend). Die schriftliche Vorbereitung ist vor Beginn des Versuchs zusätzlich auf StudOn elektronisch einzureichen. • Die Ergebnisse eines jeden Versuchs sind während der Versuchsdurchführung auf den Versuchsrechnern vorzuhalten (Programmieraufgaben) und werden zum Abschluss des Versuchs überprüft (ausreichend/nicht ausreichend). Messergebnisse sind schriftlich zu dokumentieren. Zusätzlich sind erstellte Dateien und Unterlagen in Anschluss an die Versuchsdurchführung elektronisch auf StudOn zu hinterlegen. • Zum Bestehen des Praktikums sind 5 ausreichende Versuchsvorbereitungen und 5 ausreichende Versuchsdurchführungen notwendig. • Das erfolgreiche Absolvieren einführender Unterweisungen und Aufgaben in die verwendeten Geräte, die Matlab-Entwicklungsumgebung sowie die Lernplattform StudOn ist Voraussetzung für die weitere Teilnahme am Praktikum. • There are 5 experiments to be completed. These are described in the course documentation. Each experiment must be prepared in writing at home. The preparation is checked and evaluated at the beginning of each attempt (sufficient/not sufficient). The written preparation must also be submitted electronically on StudOn prior to the start of the attempt. • The results of each experiment are to be kept on the experimental computers during the execution of the experiment (programming tasks) and are checked at the end of the experiment (sufficient/not sufficient). Measurement results are to be documented in writing. In addition, the files and documents created are to be stored electronically on StudOn following the execution of the experiment. • To pass the lab course, 5 sufficient experiment preparations and 5 sufficient experiment executions are required. • Successful completion of introductory instruction and assignments in the equipment used, the Matlab development environment and the StudOn learning platform is a prerequisite for further participation in the lab course. ----
11	**Berechnung der Modulnote**	
12	**Turnus des Angebots**	in jedem Semester

13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 35 h Eigenstudium: 40 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• Skriptum zum Praktikum • Skriptum zur Vorlesung Digitale Übertragung bzw. Digital Communications • übliche Standardlehrwerke zur Thematik (Proakis, Haykin usw.)

1	**Modulbezeichnung** 631385	**Praktikum Hochfrequenztechnik/Mikrowellentechnik 2** (Laboratory on microwave technology 2)	**2,5 ECTS**
2	Lehrveranstaltungen	Zu diesem Modul sind keine Lehrveranstaltungen oder Lehrveranstaltungsgruppen hinterlegt!	
3	Lehrende	Zu diesem Modul sind keine Lehrveranstaltungen und somit auch keine Lehrenden hinterlegt!	

4	**Modulverantwortliche/r**	Dr.-Ing. Jan Steffen Schür
5	**Inhalt**	Theoretisch erworbene Kenntnisse, z.B. aus der Vorlesung Hochfrequenztechnik 2, zu HF-Messtechnik, aktiven HF-Bauteilen und HF-Simulationstechnik werden durch vorlesungsbegleitende Experimente im Praktikum vertieft. In Kleingruppen zu 2-3 Studierenden werden acht Versuche zu folgenden Themen der Hochfrequenz- und Mikrowellentechnik durchgeführt: • Hochfrequenzverstärker • Mischer und Frequenzvervielfacher • Hochfrequenzoszillatoren • Rechnergestützter HF-Schaltungsentwurf • 3D-Feldsimulation von HF-Komponenten • Antennenentwurf • Verstärkerentwurf • Satellitenfunk Derartige Systeme werden eingesetzt z.B. für Radaranwendungen, in einer Vielzahl von drahtlosen Kommunikationsanwendungen, im Automobilbereich und im industriellen Umfeld der HF-Messgeräteentwicklung und Materialcharakterisierung. Durch das Praktikum erhalten die Studierenden einen praktischen Einblick in die wichtigsten Arbeitsgebiete der Hochfrequenz- und Mikrowellentechnik.
6	**Lernziele und Kompetenzen**	Die Studierenden • können theoretisch erworbene Kenntnisse, z.B. aus der Vorlesung Hochfrequenztechnik 2, zu HF-Messtechnik, aktiven HF-Bauteilen und HF-Simulationstechnik durch vorlesungsbegleitende Experimente analysieren und evaluieren. • können modernste HF-Messtechnik und Simulationssoftware anwenden und Ergebnisse vergleichen. • sind in der Lage, wichtige Bauelemente wie z. B. Oszillatoren und Verstärker einzusetzen und zu analysieren • evaluieren die technische und wissenschaftliche Bedeutung aktiver HF-Geräte in der Praxis. Sie sind damit in der Lage, komplexe HF-Systeme in der Praxis zu erschaffen und zu dimensionieren, die als Voraussetzung für viele Anwendungen in Wissenschaft und Technik gelten.

7	**Voraussetzungen für die Teilnahme**	• Hochfrequenztechnik • HF-Schaltungen und Systeme (Praktikum vorlesungsbegleitend)
8	**Einpassung in Studienverlaufsplan**	Semester: 1
9	**Verwendbarkeit des Moduls**	Praktikum oder Projektarbeit Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	
11	**Berechnung der Modulnote**	
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Zinke, O., Brunswig, H., Hochfrequenztechnik 1, Springer Verlag, Berlin, 1999 Meinke, H. H., Grundlach, F.-W., Taschenbuch der Hochfrequenztechnik, Springer Verlag, Berlin, 1992

1	**Modulbezeichnung** 878210	**Praktikum Machine Learning in der Signalverarbeitung** (Lab course machine learning in signal processing)	**2,5 ECTS**
2	Lehrveranstaltungen	Praktikum: Lab Course Machine Learning in Signal Processing (4 SWS)	2,5 ECTS
3	Lehrende	Kamal Gopikrishnan Nambiar	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Andre Kaup
5	**Inhalt**	This is an advanced level lab course in machine learning. Imagine a car driving on an autobahn in an automatic mode. Among other things, the car needs to steer itself to keep driving in it's own lane. To accomplish this, the central problem is to detect the road-lane markings. These are the white solid or dashed lines that are drawn on each side of the lane. The standard modern approach to solve this type of problems is to take a large dataset of labeled examples and train a deep neural network model to accomplish the task. This is how car and pedestrian detection algorithms are developed. The difficulty with the road-lane markings is that there is no labeled dataset of them and creating such dataset would cost millions of dollars. In this lab course we will solve this problem using transfer learning and mathematical modeling: • Create cartoon-like artificial images of a road with known locations for the lane markings. • Train deep neural network on these artificial images with heavy data augmentations that mimic real-world images. • Create a dataset of unlabeled real-life videos by downloading and organizing examples from youtube. • Create a machine learning pipeline for working with these videos efficiently. • Apply the neural network that has been trained on artificial data to the real world videos. • Analyze the quality of results produced by the network. • Use mathematical modeling to correct the outputs of the network. • Retrain the network on the dataset composed of the corrected outputs. • Measure and analyze the quality of the results. The software will be written in Python using JupyterLab development framework. Access to modern GPU server will be provided. The best students will have the opportunity to contribute to the creation of state-of-the-art lane detection system for self-driving cars during or after the corse.
6	**Lernziele und Kompetenzen**	Students are able to:

		• Independently design machine learning pipelines to solve complex problems in artificial intelligence. • Choose appropriate algorithms for the problem at hand. • Use standard packages for machine learning in Python: numpy, cvxpy, scikit-learn, pywavelets, pytorch. • Debug and calibrate machine learning algorithms. Develop modification to the standard algorithms as appropriate to the problem at hand. • Explain the theoretical aspects of deep learning.
7	**Voraussetzungen für die Teilnahme**	Knowledge of Python programming language is required. Basic theoretical knowledge in machine learning is assumed: consider taking the Machine Learning in Signal Processing (MLSIP) course in the same semester.
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Praktikum oder Projektarbeit Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	
11	**Berechnung der Modulnote**	
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 60 h Eigenstudium: 15 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 676551	**Praktikum Mikroelektronik** (keine englischsprachige Modulbezeichnung hinterlegt!)	**2,5 ECTS**
2	Lehrveranstaltungen	Praktikum: Praktikum Mikroelektronik (3 SWS)	2,5 ECTS
3	Lehrende	Julian Schwarz Dr.-Ing. Tobias Dirnecker	

4	**Modulverantwortliche/r**	Dr.-Ing. Tobias Dirnecker
5	**Inhalt**	Ziel ist es, praktische Erfahrungen in den Bereichen Herstellungsverfahren und elektrische Charakterisierung, Simulation und Entwurf sowie der Anwendung von mikroelektronischen Bauelementen, Schaltungen und Systemen zu erlangen.
6	**Lernziele und Kompetenzen**	Die Studierenden Fachkompetenz Verstehen verstehen Charakterisierungsmethoden und Herstellungsverfahren für Halbleiterbauelemente stellen typische Werkzeuge und Verfahren für die Verifikation und den Entwurf mikroelektronischer Bauelemente und Schaltungen dar Anwenden analysieren grundlegende Schaltungen Lern- bzw. Methodenkompetenz wenden Charakterisierungsmethoden und Herstellungsverfahren für Halbleiterbauelemente praktisch an erwerben praktische Erfahrungen mit typischen Werkzeugen und Verfahren für die Verifikation und den Entwurf mikroelektronischer Bauelemente und Schaltungen sind in der Lage typische Schaltungen aufzubauen
7	**Voraussetzungen für die Teilnahme**	• Studienrichtung Mikroelektronik im Bachelorstudium • Absolvierung Modul Schaltungstechnik und Modul Digitaltechnik
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Praktikum oder Projektarbeit Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	
11	**Berechnung der Modulnote**	

12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 45 h Eigenstudium: 30 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Vorbereitende Literatur für die Versuche: • Praktikumsskripte • ◦ Tietze, Schenk: Halbleiterschaltungstechnik

1	**Modulbezeichnung** 242643	**Praktikum Photonik/Lasertechnik 1** (keine englischsprachige Modulbezeichnung hinterlegt!)	**2,5 ECTS**
2	Lehrveranstaltungen	Praktikum: Praktikum Photonik/Lasertechnik 1 (2 SWS)	2,5 ECTS
3	Lehrende	Max Köppel Jasper Podschus	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Bernhard Schmauß
5	**Inhalt**	In kleinen Gruppen zu 2-3 Studierenden werden zehn Versuche zu folgenden Themen der Lasertechnik und Photonik durchgeführt: • *Geometrische Optik* - Fresnelgesetze - Chromatische Aberration • *Kohärente Optik* Beugung - Optische 2D-Fouriertransformation - Raumfilterung • *HeNe-Laser* - Aktives Medium - Anschwingbedingung - Spektrum • *Gaußstrahl* - TEM00 - Abbildung durch Linsen • *Laser-Resonatoren* - g-Parameter Stabilitätsbereich • *Strahlqualität* - Multimode-Laser - Strahlparameterprodukt - Strahlprofil-Kamera • *CO2-Laser* - Gitterabstimmung - Spektrallinien - Materialbearbeitung • *Laserdioden* - FP,DFB,LED - Kennlinien - Abstrahlung - Spektrum • *Faseroptik* - Fasertypen - Moden - Dämpfung • *Singlemodefasern* - Fusionsspleißen - Laser einkoppeln Durch das Praktikum können theoretisch erworbene Kenntnisse, z.B. aus der Vorlesung Photonik 1, zu Lasern und Photonik durch vorlesungsbegleitende Experimente vertieft werden. Dies ist die Voraussetzung, um grundlegende laserbasierte Systeme in der Praxis einzusetzen, für viele Anwendungen in Wissenschaft und Technik. Derartige Systeme werden eingesetzt z.B. für die Präzisionsmesstechnik, in der industriellen Materialbearbeitung, in der Bioanalytik, für die Medizintechnik, in Geräten der Unterhaltungselektronik oder in der optischen Nachrichtentechnik.
6	**Lernziele und Kompetenzen**	Die Studierenden • verstehen aufgrund praktischer Experimente Aufbau und Funktion grundlegender optischer, faseroptischer und photonischer Komponenten • können die genannten Komponenten und Systeme sowie Laser anwendungsnah handhaben und anwenden. • können photonische Messmethoden in der Praxis erproben und charakterisieren. • können durch praktische Erfahrung Eigenschaften unterschiedlicher Lichtwellenleiter und Laser vergleichen und einschätzen.

7	**Voraussetzungen für die Teilnahme**	Voraussetzung: Photonik 1, kann auch parallel gehört werden.
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Praktikum oder Projektarbeit Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	
11	**Berechnung der Modulnote**	
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Eichler, J., Eichler, H.J: Laser. 7. Auflage, Springer Verlag, Berlin 2010. Reider, G.A.: Photonik. 3. Auflage, Springer Verlag, Berlin 2012. Bergmann, Schäfer: Lehrbuch der Experimentalphysik, Bd.3: Optik. DeGruyter 2004. Saleh, B., Teich, M.C.: Grundlagen der Photonik. 2. Auflage, Wiley-VCH 2008. Träger, F. (Editor): Springer Handbook of Lasers and Optics, 2. Auflage, Springer Verlag, Berlin 2012.

1	**Modulbezeichnung** 508483	**Praktikum Photonik/Lasertechnik 2** (keine englischsprachige Modulbezeichnung hinterlegt!)	**2,5 ECTS**
2	Lehrveranstaltungen	Praktikum: Praktikum Photonik/Lasertechnik 2 (2 SWS)	2,5 ECTS
3	Lehrende	Max Köppel Jasper Podschus Prof. Dr.-Ing. Bernhard Schmauß Prof. Dr.-Ing. Bernhard Schmauß	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Bernhard Schmauß
5	**Inhalt**	In kleinen Gruppen zu 2-3 Studierenden werden acht Versuche zu folgenden Themen der Lasertechnik und Photonik durchgeführt: • Polarisation - Doppelbrechung - Jones-Matrizen • Zeitliche Kohärenz - Michelson-Interferometer Linienbreiten • Räumliche Kohärenz - Beugung Doppelspalt • Leistungs-Laserdiode - Kennlinie Wellenlängenabstimmung • Lichtwellenmesstechnik - Wavemeter - OSA • EDFA - Erbium-dotierter Faserverstärker - Faser-Laser • Nd:YAG-Laser - Kennlinien - Resonator - Stabilität • Dynamik im Laser - Q-Switch - Spiking - Sättigbarer Absorber Anhand der Versuche wird gelernt, moderne und komplexe laserbasierte Systeme in der Praxis einzusetzen, als Voraussetzung für viele Anwendungen in Wissenschaft und Technik. Derartige Systeme werden eingesetzt z.B. für die Präzisionsmesstechnik, in der industriellen Materialbearbeitung, in der Bioanalytik, für die Medizintechnik, in Geräten der Unterhaltungselektronik oder in der optischen Nachrichtentechnik.
6	**Lernziele und Kompetenzen**	Die Studierenden • vertiefen ihre wissenschaftlichen Kenntnisse im Bereich der komplexer photonischer Systeme durch praktische Experimente. • können fortgeschrittene technische und wissenschaftliche Experimente im Bereich Photonik / Lasertechnik selbstständig und in kooperativen Gruppen planen, durchführen und reflektieren. • können Sachverhalte und Ergebnisse der im Inhalt beschriebenen Experimente bewerten und vergleichen. • sind in der Lage, eigenständig Ideen zur Lösung komplexer technisch-wissenschaftlicher Messaufgaben im Bereich der Photonik und Lasertechnik zu entwickeln.
7	**Voraussetzungen für die Teilnahme**	• Photonik 1 • Photonik 2 (kann vorlesungsbegleitend besucht werden)
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Praktikum oder Projektarbeit Master of Science Information and Communication Technology 20222

10	**Studien- und Prüfungsleistungen**	
11	**Berechnung der Modulnote**	
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 45 h Eigenstudium: 30 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Träger, F. (Ed.): Handbook of Lasers and Optics, Springer Verlag, Berlin 2007. Eichler/Eichler: Laser. Springer Verlag, Berlin 2006. Reider, G.A.: Photonik. Springer Verlag, Berlin 2003. Bergmann, Schäfer: Lehrbuch der Experimentalphysik, Bd.3: Optik. DeGruyter 1993.

1	**Modulbezeichnung** 320376	**Praktikum Test** (keine englischsprachige Modulbezeichnung hinterlegt!)	**2,5 ECTS**
2	Lehrveranstaltungen	Praktikum: Praktikum Test (3 SWS)	2,5 ECTS
3	Lehrende	Tobias Rumpel	

4	**Modulverantwortliche/r**	Peter Meisel Prof. Dr.-Ing. Sebastian Sattler
5	**Inhalt**	Im Entwicklungsprozess elektronischer Bauteile wie auch bei deren Massenproduktion werden mit Hilfe automatischer Testsysteme die elektrischen Kenngrößen eines Bauteils erfasst. Das Praktikum Testen mit automatischen Testsystemen" gibt einen Einblick in typische messtechnische Aufgabenstellungen und Arbeiten, die während der Entwicklung integrierter mikroelektronischer Systeme vorkommen.
6	**Lernziele und Kompetenzen**	Verstehen • Die Studierenden beschreiben die Abläufe im Laborbetrieb und erläutern die Eigenschaften eines Testsystems • Die Studierenden erklären die Elemente eines Testprogramms • Die Studierenden formulieren die verschiedenen Möglichkeiten von Test (Funktionstest, Dynamischer Test) • Die Studierenden erläutern die Entwicklung von Test-Pattern Analysieren • Die Studierenden analysieren das DUT und entwickeln daraus die richtige Auswahl an Testparametern Erschaffen • Die Studierenden entwerfen Test-Pattern für den Boundary-Scan-Test • Die Studierenden erstellen aus gegebener Aufgabenstellung komplettes Testprogramm • Die Studierenden beurteilen des Testprogramms unter Berücksichtigung von Produktivität und Debugging Lern- bzw. Methodenkompetenz • Die Studierenden erwerben praktische Erfahrungen im Umgang mit einem automatischen Testsystem (ATE) • Die Studierenden erfahren die Arbeitsumgebung in einem Reinraum-Labor und die sich daraus ergebenen Vorschriften Selbstkompetenz • Die Studierenden können in Gruppen kooperativ arbeiten und Fehleranalysen durchführen
7	**Voraussetzungen für die Teilnahme**	Keine

8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Praktikum oder Projektarbeit Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	
11	**Berechnung der Modulnote**	
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 45 h Eigenstudium: 30 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 443121	**Praktikum zu High-Performance Analog- und Umsetzer-Design** (keine englischsprachige Modulbezeichnung hinterlegt!)	**2,5 ECTS**
2	Lehrveranstaltungen	Praktikum: Praktikum zu High-Performance Analog- und Umsetzer-Design (3 SWS)	2,5 ECTS
3	Lehrende	Benedict Scheiner	

4	**Modulverantwortliche/r**	Benedict Scheiner Prof. Dr.-Ing. Robert Weigel
5	**Inhalt**	Im Praktikum High-Performance Analog und Umsetzer Design wird ein Temperatursensor mit USB-Anschluss entwickelt. Die Teilnehmer müssen die einzelnen Schaltungsblöcke zuerst dimensionieren und simulieren, bevor die Schaltung auf einer Leiterplatte aufgebaut und gemessen wird. Im einzelnen sind folgende Blöcke zu untersuchen: • Temperaturstabilen Spannungsreferenz (Bandgap) • Präziser Instrumentenverstärker • Zeitkontinuierlicher Delta-Sigma Modulator • Nach Abschluss des Praktikums kann jeder Student seine eigene Platine mit nach Hause nehmen. Das Praktikum findet als einwöchige Blockveranstaltung während der Semesterferien im August/September statt. Die Anmeldung erfolgt über das WAS-System.
6	**Lernziele und Kompetenzen**	Nach der Teilnahme an den Modulveranstaltungen sind die Studierenden in der Lage, grundlegende Konzepte von Analogschaltungen und Umsetzern anzuwenden und auf Basis dieser einen Temperatursensor mit USB Anschluss zu entwickeln.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Praktikum oder Projektarbeit Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	
11	**Berechnung der Modulnote**	
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 45 h Eigenstudium: 30 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 504407	**Software Engineering in der Praxis** (Software engineering in practice)	**2,5 ECTS**
2	Lehrveranstaltungen	Übung: Software Engineering in der Praxis (3 SWS)	-
3	Lehrende	Loui Al Sardy	

4	**Modulverantwortliche/r**	Prof. Dr. Francesca Saglietti
5	**Inhalt**	In den praktischen Übungen werden Werkzeuge zur Entwicklung und zur Analyse komplexer Software vorgestellt, deren industrielle Einsetzbarkeit anschließend von den Teilnehmern anhand für die Praxis repräsentativer Aufgabenstellungen erprobt wird.
6	**Lernziele und Kompetenzen**	Die Studierenden • erläutern Potenzial und Grenzen unterschiedlicher Werkzeuge zur Unterstützung softwaretechnischer Tätigkeiten; • wenden unterschiedliche Werkzeuge an, um sowohl selbständig als auch in Teams Beispielaufgaben aus dem Bereich der objektorientierten Analyse, des objektorientierten Entwurfs, des Testens, des Beweisens und des Projektmanagements zu lösen;
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Praktikum oder Projektarbeit Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	
11	**Berechnung der Modulnote**	
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

Seminar

1	**Modulbezeichnung** 750143	**Advanced Seminar on Medical Electronics and Systems for Ambient Assisted Living AAL** (Advanced seminar on medical electronics and systems for ambient assisted living AAL)	**2,5 ECTS**
2	Lehrveranstaltungen	Seminar: Advanced Seminar on Medical Electronics and Systems for Ambient Assisted Living AAL (0 SWS)	2,5 ECTS
3	Lehrende	Benedict Scheiner	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Georg Fischer
5	**Inhalt**	During the seminar current issues in the field of "Modern concepts in medical electronics" will be discussed. After a joint briefing the students will independently work on the chosen topic under the guidance of a supervisor. The results are summarized in a four-page seminar thesis. The main task of the seminar is a 30 minute presentation of each student. A discussion with the listeners concludes the seminar. Attendance during the whole workshop day is mandatory for passing the seminar. Topics: • Electronics for medical diagnostics and therapy • Electronics based human assistance systems • Electronic systems for AAL Ambient Assisted Living • Electronical Systems incorporating Microsystem Components (MEMS) • BAN body area networks • Coupling of medical electronic systems to Patient health record data bases • Near body Energy Harvesting and Scavenging • Circuit design for microwave based blood analysis • MEMS Lab-on-chip • Vital parameter supervision
6	**Lernziele und Kompetenzen**	• Students will acquire basic knowledge in research, topics preparation and presentation techniques. • Students will focus on technical issues for a given topic in the field of medical electronics. • Students will independently deepen a technical issue on a concrete example. • Students will learn the ability to familiarize themselves with unknown problems and to present the results. • Students will achive the ability to formulate questions as a active listener and to discuss technical issues.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Seminar Master of Science Information and Communication Technology 20222

10	**Studien- und Prüfungsleistungen**	Seminarleistung
11	**Berechnung der Modulnote**	Seminarleistung (100%)
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 15 h Eigenstudium: 60 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 330542	**Audio Processing Seminar** (Audio processing seminar)	**2,5 ECTS**
2	Lehrveranstaltungen	Seminar: Audio Processing Seminar (2 SWS)	-
3	Lehrende	Prof. Dr. Meinard Müller Prof. Dr.-Ing. Bernd Edler Prof. Dr. Nils Peters Prof. Dr. Emanuël Habets Prof. Dr.-Ing. Jürgen Herre Dr.-Ing. Stefan Turowski	

4	**Modulverantwortliche/r**	Prof. Dr. Emanuël Habets
5	**Inhalt**	The audio processing seminar trains students to prepare, summarize and present a recent scientific paper from the field of audio processing. The students work on a recent cutting-edge paper from one of the following fields: • Speech and Audio Coding • Audio Signal Analysis • Audio Signal Processing with the Internet of Things • Spatial Audio Signal Processing • Semantic Audio Processing • Audio in Virtual Reality During of the seminar, each participant prepares a paper, creates a written report (3-7 pages) and presents it in the form of a talk (20 min.) to the other participants. Thereby, the students are guided by their own supervisors. General skills are taught in joint classes. Paper specific aspects are discussed individually between the students and their supervisor. The seminar ends with the presentation of all topics over the course of one or two days. Participation in these presentations and the following discussions are mandatory for all participants. The seminar not only gives a broad overview of the field of audio processing, but conveys fundamental scientific working and communication skills.
6	**Lernziele und Kompetenzen**	Learning objectives and skills Students will gain the following skills: • How to analyze a scientific paper and understand its key principles and field of application. • How to perform a thorough literature survey and evaluate relevant literature for the focus of key points in the paper. • How to acquire a broad knowledge and deeper understanding of the specific scientific area. • How to prepare the subject, identify its most important topics, their dependencies, didactic reduction. • How to compile a written summary of a paper, scientific writing, correct citations. • How to create an appealing visual presentation, review and successively optimize it.

		• How to present the topic in front of other students, how to train presentation skills. • How to analyze presentations of other students, deriving questions, learn to participate in a scientific discussion.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Seminar Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Seminarleistung
11	**Berechnung der Modulnote**	Seminarleistung (100%)
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 97760	**Hauptseminar Ausgewählte Kapitel der Informationstechnik (Kommunikationselektronik)** (Hauptseminar Ausgewählte Kapitel der Informationstechnik (Kommunikationselektronik))	**2,5 ECTS**
2	Lehrveranstaltungen	Hauptseminar: Ausgewählte Kapitel der Informationstechnik (2 SWS)	2,5 ECTS
3	Lehrende	Sebastian Klob	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Albert Heuberger
5	**Inhalt**	**Sommersemester: Integrierte Sender- und Empfängerschaltungen** Im SS werden integrierte Sender- und Empfängerschaltungen behandelt. Studenten sollen einen Einblick in die Technologieauswahl und den Schaltungsentwurf von Schlüsselkomponenten bekommen. Die Vortragsreihe beginnt mit Übersichtsthemen zu Empfängerarchitekturen und Halbleiter-Technologien sowie Simulationswerkzeugen für die Integration von RF-Schaltungen. Mit wechselnden Schwerpunkten auf verschiedenen Funkstandards, Halbleitertechnologien oder Frequenzbereichen werden integrierte RF-Schaltungen behandelt. Je nach Schwerpunkt sollen Schlüsselkomponenten wie rauscharme Verstärker, Mischer, spannungsgesteuerte Oszillatoren und Leistungsverstärker oder komplette Sender- und Empfängerschaltungen erörtert werden. Ein Besuch der Abteilung Analoges IC-Design des Fraunhofer-IIS rundet das Seminar ab. **Wintersemester: Digitaler Rundfunk** Im Seminar „Digitale Rundfunksysteme“ werden ausgewählte Themen zu neuen terrestrischen und satellitengestützten digitalen Rundfunksystemen behandelt. Das Seminar startet mit einem historischen Exkurs in die Entwicklungsgeschichte des Radios und der Entwicklung des analogen Rundfunks in Deutschland sowie einer Einführung in die weltweit existierenden terrestrischen und satellitengestützten digitalen Rundfunksysteme. Mit wechselnden Schwerpunkten werden neue Dienste sowie die technischen Komponenten, Übertragungs- und Datenprotokolle sowie neue Standards entlang der gesamten Übertragungskette vom Quellensignal über den Hochfrequenzkanal bis zum Empfänger behandelt. Ein Besuch bei funklust (ein Zusammenschluss der drei studentischen Medieninitiativen Campusradio bit express, Uniradio Unimax und dem Video-Format t°fau an der FAU), sowie Fachvorträge von externen Experten mit Diskussion zu neuen Entwicklungen runden das Seminar ab.

6	**Lernziele und Kompetenzen**	1. Die Studierenden sollen lernen, sich ein wissenschaftliches Thema selbständig zu erarbeiten und eine didaktisch durchdachte Präsentation vorzubereiten. 2. Die Studierenden sollen lernen unter Einhaltung von Zeitvorgaben, Ihre Erkenntnisse publikumsangepasst zu vermitteln. 3. Die Studierenden sollen Ihre verbale sowie nonverbale Kommunikation weiterentwickeln. 4. Die Studierenden sollen ansatzweise lernen, wie eine wissenschaftliche Veröffentlichung aussehen sollte. Dies alles geschieht im Rahmen des unter Seminarinhalte ausgeführten Themenbereichs. Die Leistungen werden im Zusammenhang mit dem individuellen Thema des/ der Studierenden erbracht.
7	**Voraussetzungen für die Teilnahme**	Keine formalen Voraussetzungen. Empfohlen werden ausdrücklich mindestens 4 Semester Bachelor-Studium in EEI, Informatik oder IuK.
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Seminar Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	Wird je nach Schwerpunktwahl des Seminars neu festgelegt.

1	**Modulbezeichnung** 97770	**Hauptseminar Ausgewählte Kapitel der Navigation und Identifikation** (Hauptseminar Ausgewählte Kapitel der Navigation und Identifikation)	**2,5 ECTS**
2	Lehrveranstaltungen	Hauptseminar: Ausgewählte Kapitel der Navigation und Identifikation (2 SWS)	2,5 ECTS
3	Lehrende	Lukas Meyer Adam Kalisz Prof. Dr. Jörn Thielecke	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Albert Heuberger Prof. Dr. Jörn Thielecke
5	**Inhalt**	**Sommersemester: Radio-/ Hochfrequenz-Identifikationssysteme (RFID)** Das Themenspektrum des Seminars im Sommersemester besitzt als Schwerpunkt die Bereiche Radio-/Hochfrequenz-Identifikationssysteme (RFID) und Telemetrie. Während des ersten Seminartermins werden den Studierenden Betreuer und Themen zugeteilt, wobei die Themen im Forschungsbereich des jeweiligen Betreuers liegen. Mit Unterstützung des Betreuers wird ein 30-minütiger Vortrag ausgearbeitet, der im Laufe des Seminars vorgetragen werden muss. Zusätzlich ist eine sechsseitige Ausarbeitung zu schreiben, die wissenschaftlichen Gesichtspunkten genügen muss. Ein fünfminütiger Probevortrag bietet die Möglichkeit, vor dem eigentlichen Vortrag eine Rückkopplung über den eigenen Vortragsstil zu erhalten und die Zielsetzung des Seminars besser zu verstehen. Probevorträge und die Vorträge selbst (30 Min.) werden mit der Kamera aufgezeichnet, um anschließend den Vortragsstil besser diskutieren zu können. **Wintersemester: Roboternavigation** Thematisch befasst sich das Seminar mit der Navigation von Robotern bis hin zum autonomen Fahren von Autos, z.B. pilotiertem Fahren. Themenschwerpunkte können beispielsweise sein: Sensoren, GPS, Trägheitsnavigation, laserbasierte Navigation, kamerabasierte Navigation, Sensordatenfusion, Filtermethoden, automatisierte Kartenerstellung, Simultaneous Localization and Mapping, maschinelle Lernverfahren oder Wegeplanung. Für das Seminar werden circa 10 aktuelle Themen aus diesen Bereichen ausgewählt, die von den Studierenden bearbeitet werden können.
6	**Lernziele und Kompetenzen**	1. Sie sollen lernen, sich ein wissenschaftliches Thema selbständig zu erarbeiten und eine didaktisch durchdachte Präsentation vorzubereiten. 2. Sie sollen lernen unter Einhaltung von Zeitvorgaben, Ihre Erkenntnisse publikumsangepasst zu vermitteln.

		3. Sie sollen Ihre verbale sowie nonverbale Kommunikation weiterentwickeln. 4. Sie sollen ansatzweise lernen, wie eine wissenschaftliche Veröffentlichung aussehen sollte. Selbstkompetenz Fähigkeit und Bereitschaft, sich weiterzuentwickeln und das eigene Leben eigenständig und verantwortlich im jeweiligen sozialen, kulturellen bzw. beruflichen Kontext zu gestalten, Selbstkritische Einschätzung des Kompetenzniveaus bei der Vor- und Nachbereitung von Lehrveranstaltungen. Selbstkritische Bewertung der Studienleistungen. Sozialkompetenz Der Absolvent ist in der Lage, zielorientiert mit seinen Kommilitonen sowie externen Fachleuten und fachfremden Dritten zusammenzuarbeiten. Hierbei ist er in der Lage, fachliche und soziale Situationen zu erfassen, sich mit ihnen rational und verantwortungsbewusst auseinanderzusetzen sowie dadurch seine Arbeits- und Lebenswelt mitzugestalten.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Seminar Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Portfolio
11	**Berechnung der Modulnote**	Portfolio (100%)
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 92508	**Hauptseminar Cognitive Science in Engineering** (keine englischsprachige Modulbezeichnung hinterlegt!)	**2,5 ECTS**
2	Lehrveranstaltungen	Seminar: Seminar "Cognitive Science in Engineering" (2 SWS)	2,5 ECTS
3	Lehrende	Dr. Chenxu Hao	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Philipp Beckerle Dr. Chenxu Hao
5	**Inhalt**	In the seminar, students will analyze, present. and discuss recent research topics in Cognitve Science and Engineering. Besides reflecting contemporary literature, the students are asked to conclude and suggest directions for future research.
6	**Lernziele und Kompetenzen**	On successful completion of this module, students will be able to comprehend and convey recent research challenges in the area of Cognitive Science in Engineering. Moreover, they are prepared to infer future research lines from recent developments.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Seminar Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Seminarleistung
11	**Berechnung der Modulnote**	Seminarleistung (100%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 94595	**Hauptseminar Software Engineering** (keine englischsprachige Modulbezeichnung hinterlegt!)	**2,5 ECTS**
2	Lehrveranstaltungen	Seminar: Design Patterns und Anti-Patterns (2 SWS)	-
3	Lehrende	Loui Al Sardy Dr.-Ing. Marc Spisländer	

4	**Modulverantwortliche/r**	Prof. Dr. Francesca Saglietti
5	**Inhalt**	In diesem Seminar werden zahlreiche in der Praxis bewährte Entwurfsmuster präsentiert, sowie typische Fehler, die während des gesamten Lebenszyklus' eines Software-Systems auftreten können, zusammen mit ihrer Erkennung und Behebung vorgestellt.
6	**Lernziele und Kompetenzen**	Die Studierenden • erläutern etablierte Design Patterns sowie bekannte Anti-Patterns; • klassifizieren sie und charakterisieren deren Stärken und Schwächen bzw. Negativfolgen; • tragen vor Publikum über wissenschaftliche Ergebnisse vor; • recherchieren selbständig Fachliteratur; • fassen wissenschaftliche Erkenntnisse in Schriftform zusammen; • nutzen Verbesserungshinweise des Betreuers zur Analyse eigener Stärken und Schwächen und leiten daraus Konsequenzen für ihr künftiges Lern-Handeln ab; • können komplexe fachbezogene Inhalte klar und zielgruppengerecht präsentieren und argumentativ vertreten.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Seminar Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	mündlich Seminarleistung
11	**Berechnung der Modulnote**	mündlich (50%) Seminarleistung (50%)
12	**Turnus des Angebots**	nur im Sommersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 47643	**Seminar Advanced Algorithms in Medical Image Processing** (keine englischsprachige Modulbezeichnung hinterlegt!)	**5 ECTS**
2	Lehrveranstaltungen	Seminar: Seminar Advanced Deep Learning (2 SWS)	5 ECTS
3	Lehrende	Dr.-Ing. Vincent Christlein Katharina Breininger	

4	**Modulverantwortliche/r**	Katharina Breininger Prof. Dr.-Ing. Andreas Maier
5	**Inhalt**	Deep Learning-based algorithms showed great performance in many fields of image processing and pattern recognition and compete with technologies such as compressive sensing and iterative optimization. The basis for the success of these algorithms is the availability of large amounts of data (big data) for training and of high computing power (typically GPUs). In this seminar, we try to explore advanced deep learning methods. In particular, we will aim to develop a deeper understanding of certain topics, for example, graph neural networks, unsupervised learning, differentiable learning, invertible learning, neural ordinary differential equations, transfer learning, multi-task learning, uncertainty DL, etc.
6	**Lernziele und Kompetenzen**	Students will be able to • perform their own literature research on a given subject • independently research this subject • present and introduce the subject to their student peers • give a scientific talk in English according to international conference standards
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Seminar Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Seminarleistung
11	**Berechnung der Modulnote**	Seminarleistung (100%)
12	**Turnus des Angebots**	Unregelmäßig
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 120 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 914949	**Seminar Ausgewählte Kapitel der Multimediakommunikation und Signalverarbeitung** (Seminar on selected topics of multimedia communications and signal processing)	**2,5 ECTS**
2	Lehrveranstaltungen	Seminar: Seminar Ausgewählte Kapitel der Multimediakommunikation und Signalverarbeitung (2 SWS)	2,5 ECTS
3	Lehrende	Kamal Gopikrishnan Nambiar Prof. Dr. Vasileios Belagiannis	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Andre Kaup Prof. Dr.-Ing. Walter Kellermann
5	**Inhalt**	Im Seminar Multimediakommunikation und Signalverarbeitung werden aktuelle Themen aus dem Bereich der Multimediakommunikation und Signalverarbeitung bearbeitet. Nach einer gemeinsamen Vorbesprechung und Themenauswahl werden die einzelnen Themen unter Anleitung eines Betreuers oder einer Betreuerin eigenständig im Hinblick auf eine Präsentation in Vortragsform erarbeitet. Eine kurze Präsentation der Struktur und erster Ergebnisse erfolgt etwa 5 Wochen nach der Vorbesprechung. Gegen Ende des Vorlesungszeitraums hält jeder Teilnehmer einen ca. 30-minütigen Vortrag mit anschließender Diskussion im Rahmen eines ganztägigen Workshops. Als Begleitmaterial zum Vortrag wird auch eine ca. 10-seitige Ausarbeitung erstellt. Für die Vortragsveranstaltungen besteht Anwesenheitspflicht. The Seminar on Selected Topics of Multimedia Communications and Signal Processing deals with current research topics in the area of multimedia communications and signal processing. In an introductory meeting, the course of the seminar is outlined and each participant selects one of the offered topics. The participant should become familiar with the assigned research topic and present it by a report and a talk at the end of the seminar with the support of a supervisor. In an intermediate meeting about 5 weeks after the introductory meeting, the participants give a brief presentation about their topics and show first results. In addition, hints for the preparation of the final talk are provided at this meeting. At the end of the semester, a final one-day meeting takes place where each participant presents his topic in a talk of 30 minutes followed by a discussion and questions from the audience. In addition, each participant has to submit a report of about 10 pages about his topic a few days before the final meeting.
6	**Lernziele und Kompetenzen**	Die Studierenden • erlernen grundlegende Techniken der Recherche, Themenaufbereitung und Präsentation technischer Inhalte und wenden diese an • analysieren und evaluieren gegebene Literatur im Hinblick auf die Schwerpunkte ihres Vortrags zu einem technischen Thema

		• wenden ihr bisher im Studium erworbenes Wissen an, um davon ausgehend eigenständig einen technischen Schwerpunkt zu vertiefen • wenden ihr bisheriges Wissen an, um als Zuhörer sinnvolle Fragen zu einem Vortragsthema zu formulieren und das Präsentierte zu diskutieren • analysieren und evaluieren die Präsentationen der anderen Seminarteilnehmer. The students • acquire and apply fundamental techniques to conduct a literature survey, and to prepare and present a technical topic • analyze and evaluate provided literature regarding the focus of their technical presentation • apply the knowledge acquired during their studies to deepen by themselves their technical focus • apply acquired knowledge to ask a presenter questions and to discuss the presentation • analyze and evaluate the presentations of other seminar participants.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Seminar Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Seminarleistung
11	**Berechnung der Modulnote**	Seminarleistung (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 775681	**Seminar Ausgewählte Kapitel der Nachrichtentechnik** (Selected areas in communications)	**2,5 ECTS**
2	Lehrveranstaltungen	Hauptseminar: Seminar Ausgewählte Kapitel der Nachrichtentechnik (2 SWS) ja.	2,5 ECTS
3	Lehrende	Prof. Dr.-Ing. Ralf Müller Prof. Dr. Laura Cottatellucci Prof. Dr. Wolfgang Gerstacker Prof. Dr.-Ing. Robert Schober	

4	**Modulverantwortliche/r**	Prof. Dr. Laura Cottatellucci Prof. Dr. Wolfgang Gerstacker Prof. Dr.-Ing. Ralf Müller Prof. Dr.-Ing. Robert Schober Dr.-Ing. Clemens Stierstorfer
5	**Inhalt**	Inhalt / Contents In diesem Seminar werden aktuelle Themen innerhalb eines wechselnden Schwerpunkts im Bereich der Nachrichtentechnik bzw. drahtlosen Kommunikation bearbeitet und präsentiert. _____ In this seminar, current topics in the field of telecommunications and wireless communication are presented by students.
6	**Lernziele und Kompetenzen**	Die Studierenden • erlernen grundlegende Techniken der Recherche, Themenaufbereitung und Präsentation technischer Inhalte und wenden diese an • analysieren und evaluieren gegebene Literatur im Hinblick auf die Schwerpunkte ihres Vortrags zu einem technischen Thema • wenden ihr bisher im Studium erworbenes Wissen an, um davon ausgehend eigenständig einen technischen Schwerpunkt zu vertiefen • wenden ihr bisheriges Wissen an, um als Zuhörer sinnvolle Fragen zu einem Vortragsthema zu formulieren und das Präsentierte zu diskutieren • analysieren und evaluieren die Präsentationen der anderen Seminarteilnehmer. _____ The students • learn basic techniques of research, topic preparation and presentation of technical content and apply them • analyze and evaluate given literature with regard to the focus of a talk on a technical topic • apply the knowledge they have acquired during their studies to independently deepen their technical focus

		• apply their previous knowledge to formulate meaningful questions as a listener on a talk and to discuss what is presented • analyze and evaluate the presentations of the other seminar participants.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Seminar Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Seminarleistung • Die Themen werden unter Anleitung eines/r Betreuers/in eigenständig im Hinblick auf eine Präsentation in Vortragsform erarbeitet. • Studierende haben die Möglichkeit sich aktiv an der Formulierung des Vortragsthemas zu beteiligen. • Themen werden bei einer Vorbesprechung zu Beginn des Semesters vergeben. • Eine kurze Präsentation der Struktur und erster Ergebnisse erfolgt etwa 5 Wochen nach der Vorbesprechung. Gegen Ende des Vorlesungszeitraums hält jede/r Studierende einen ca. 30-minütigen Vortrag mit anschließender 15-minütiger Diskussion. • Vor den Beiträgen der Studierenden erfolgt eine Einführung zur Vortragstechnik durch Mitarbeiter des Lehrstuhls. • Es wird eine ca. 10-15-seitige Ausarbeitung erstellt. ═══════════ • The topics are independently worked out under the guidance of a supervisor. • Students have the opportunity to actively participate in the formulation of their individual topic. • Topics are assigned at a preliminary meeting at the beginning of the semester. • A brief presentation of the structure and initial results will be given about 5 weeks after the preliminary discussion. • Towards the end of the lecture period, each student gives a talk of approximately 30 minutes followed by a 15-minute discussion. • Students will be introduced into lecture techniques. • An approx. 10-15-page paper has to be written.
11	**Berechnung der Modulnote**	Seminarleistung (100%) • ca. halbstündiger Vortrag (60%) • Ausarbeitung im Umfang von 10-15 Seiten (vergleichbar IEEE Paper zweispaltig, 30%) • aktive Teilnahme an der Diskussion anderer Vorträge (10%)

		--- • approx. half-hour presentation (60%) • paper of 10-15 pages (comparable to IEEE paper in two columns, 30%) • active participation in the discussion of other presentations (10%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 15 h Eigenstudium: 60 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	• Unterlagen zu den Modulen Digitale Übertragung, MIMO Communication Systems, Convex Optimization in Communications and Signal Processing • Informationen zur Literatursuche und zu Präsentationstechniken • Vrolagen für Ausarbeitungen und Präsentationsfolien werden zur Verfügugn gestellt • Technische Literatur im Themengebiet --- • Lecture notes of the modules Digital Transmission, MIMO Communication Systems, Convex Optimization in Communications and Signal Processing • Information on literature search and presentation techniques • Templates for papers and presentation slides will be provided • Technical literature in the subject area

1	**Modulbezeichnung** 514944	**Seminar Deep Learning** (Seminar deep learning)	**5 ECTS**
2	Lehrveranstaltungen	Zu diesem Modul sind keine Lehrveranstaltungen oder Lehrveranstaltungsgruppen hinterlegt!	
3	Lehrende	Zu diesem Modul sind keine Lehrveranstaltungen und somit auch keine Lehrenden hinterlegt!	

4	**Modulverantwortliche/r**	
5	**Inhalt**	keine Inhaltsbeschreibung hinterlegt!
6	**Lernziele und Kompetenzen**	keine Beschreibung der Lernziele und Kompetenzen hinterlegt!
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 0
9	**Verwendbarkeit des Moduls**	Seminar Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Seminarleistung
11	**Berechnung der Modulnote**	Seminarleistung (100%)
12	**Turnus des Angebots**	keine Angaben zum Turnus des Angebots hinterlegt!
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: ?? h (keine Angaben zum Arbeitsaufwand in Präsenzzeit hinterlegt) Eigenstudium: ?? h (keine Angaben zum Arbeitsaufwand im Eigenstudium hinterlegt)
14	**Dauer des Moduls**	?? Semester (keine Angaben zur Dauer des Moduls hinterlegt)
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 93656	**Seminar Energieinformatik** (keine englischsprachige Modulbezeichnung hinterlegt!)	**5 ECTS**
2	Lehrveranstaltungen	Seminar: Seminar Energieinformatik (2 SWS)	5 ECTS
3	Lehrende	Marco Pruckner	

4	**Modulverantwortliche/r**	Marco Pruckner
5	**Inhalt**	Teilnehmende arbeiten sich selbständig anhand der ausgewählten wissenschaftlichen Literatur in ein vorgeschlagenes Thema aus dem Bereich der Energieinformatik ein, erstellen dazu eine schriftliche Ausarbeitung, bereiten einen Seminarvortrag vor und präsentieren ihn vor Lehrenden, interessierten wissenschaftlichen Mitarbeitern des Instituts und Kommilitonen der Lehrveranstaltung.
6	**Lernziele und Kompetenzen**	Zu den zu erwerbenden Kompetenzen zählen • die Literaturrecherche, • korrektes Zitieren, • die Fähigkeit auszuwählen, welche Aspekte in der Ausarbeitung und im Vortrag behandelt werden, • zielgruppengerechtes Schreiben, • die Verwendung von Textverarbeitungswerkzeugen für die Präsentation und die Ausarbeitung, • sicheres Auftreten beim Vortragen eines wissenschaftlichen Themas sowie • fachspezifische Fragen zum ausgewählten Themengebiet zu beantworten. Am Ende jeder Lehreinheit • stellen die Teilnehmenden gezielt Fragen zum vorgetragenen Thema, • diskutieren über fachliche Aspekte der Präsentation, • üben konstruktive Kritik an der Darstellung des Themas und • bewerten die Präsentation aus vortragstechnischer Sicht.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Seminar Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Seminarleistung
11	**Berechnung der Modulnote**	Seminarleistung (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 120 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch

16	**Literaturhinweise**	• gesammelte Bücher, Artikel und Weblinks zu einem einschlägigen Themengebiet • Marco Pruckner. Allgemeine Hinweise zu Seminarvorträgen und -ausarbeitungen.

1	**Modulbezeichnung** 609624	**Seminar Kommunikationssysteme (B.Sc.)** (Seminar communication systems (B.Sc.))	**5 ECTS**
2	Lehrveranstaltungen	Seminar: Seminar: Kommunikationssysteme (2 SWS)	5 ECTS
3	Lehrende	Prof. Dr. Reinhard German Lisa Maile	

4	**Modulverantwortliche/r**	Prof. Dr. Reinhard German
5	**Inhalt**	Teilnehmende arbeiten sich selbständig anhand der ausgewählten wissenschaftlichen Literatur in ein vorgeschlagenes Thema aus dem Bereich der Kommunikationssysteme ein, erstellen dazu eine schriftliche Ausarbeitung, bereiten einen Seminarvortrag vor und präsentieren ihn vor Lehrenden, interessierten wissenschaftlichen Mitarbeitern des Instituts und Kommilitonen der Lehrveranstaltung.
6	**Lernziele und Kompetenzen**	Zu den zu erwerbenden Kompetenzen zählen • die Literaturrecherche, • korrektes Zitieren, • die Fähigkeit auszuwählen, welche Aspekte in der Ausarbeitung und im Vortrag behandelt werden, • zielgruppengerechtes Schreiben, • die Verwendung von Textverarbeitungswerkzeugen für die Präsentation - und die Ausarbeitung, • sicheres Auftreten beim Vortragen eines wissenschaftlichen Themas sowie • fachspezifische Fragen zum ausgewählten Themengebiet zu beantworten. Am Ende jeder Lehreinheit • stellen die Teilnehmenden gezielt Fragen zum vorgetragenen Thema, • diskutieren über fachliche Aspekte der Präsentation, • üben konstruktive Kritik an der Darstellung des Themas und • bewerten die Präsentation aus vortragstechnischer Sicht.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Seminar Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Seminarleistung
11	**Berechnung der Modulnote**	Seminarleistung (100%)
12	**Turnus des Angebots**	Unregelmäßig
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 120 h
14	**Dauer des Moduls**	1 Semester

15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 986443	**Seminar Kommunikationssysteme (M.Sc.)** (Seminar communication systems (M.Sc.))	**5 ECTS**
2	Lehrveranstaltungen	Seminar: Seminar: Kommunikationssysteme (2 SWS)	5 ECTS
3	Lehrende	Prof. Dr. Reinhard German Lisa Maile	

4	**Modulverantwortliche/r**	Prof. Dr. Reinhard German
5	**Inhalt**	Teilnehmende arbeiten sich selbständig anhand der ausgewählten wissenschaftlichen Literatur in ein vorgeschlagenes Thema aus dem Bereich der Kommunikationssysteme ein, erstellen dazu eine schriftliche Ausarbeitung, bereiten einen Seminarvortrag vor und präsentieren ihn vor Lehrenden, interessierten wissenschaftlichen Mitarbeitern des Instituts und Kommilitonen der Lehrveranstaltung.
6	**Lernziele und Kompetenzen**	Zu den zu erwerbenden Kompetenzen zählen • die Literaturrecherche, • korrektes Zitieren, • die Fähigkeit auszuwählen, welche Aspekte in der Ausarbeitung und im Vortrag behandelt werden, • zielgruppengerechtes Schreiben, • die Verwendung von Textverarbeitungswerkzeugen für die Präsentation und die Ausarbeitung, • sicheres Auftreten beim Vortragen eines wissenschaftlichen Themas sowie • fachspezifische Fragen zum ausgewählten Themengebiet zu beantworten. Am Ende jeder Lehreinheit • stellen die Teilnehmenden gezielt Fragen zum vorgetragenen Thema, • diskutieren über fachliche Aspekte der Präsentation, • üben konstruktive Kritik an der Darstellung des Themas und • bewerten die Präsentation aus vortragstechnischer Sicht.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Seminar Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Seminarleistung
11	**Berechnung der Modulnote**	Seminarleistung (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 120 h
14	**Dauer des Moduls**	1 Semester

15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	• gesammelte Bücher, Artikel und Weblinks zu einem einschlägigen Themengebiet • Reinhard German. Allgemeine Hinweise zu Seminarvorträgen und -ausarbeitungen.

1	**Modulbezeichnung** 903776	**Seminar Machine Learning and Data Analytics for Industry 4.0** (Seminar machine learning and data analytics for industry 4.0)	**5 ECTS**
2	Lehrveranstaltungen	Seminar: Maschinelles Lernen und Datenanalytik für Industrie 4.0 (2 SWS) Attendance of all meetings is required.	5 ECTS
3	Lehrende	Prof. Dr. Björn Eskofier Johannes Roider Christoph Scholl	

4	**Modulverantwortliche/r**	Prof. Dr. Björn Eskofier An Nguyen Johannes Roider
5	**Inhalt**	Companies in all kinds of industries are producing and collecting rapidly more and more data from various sources. This is enabled by technologies such as the Internet of Things (IoT), Cyber-physical systems (CPS) and cloud computing. Hence, there is an increasing demand in industry and research for students and graduates with machine learning and data analytics skills in the Industry 4.0 context. The goal of this seminar is to give students insights about state-of-the-art machine learning and data analytics methods for industrial and business applications. In this seminar, the Industry 4.0 term will not only be limited to manufacturing processes, but comprise all business functions. Students will mainly work independently on either an implementation-centric or a research-centric topic. The implementation-centric topics will focus primarily on the implementation of algorithms and analytical components (using provided or open source datasets), while the research-centric topics will focus on researching and structuring literature of a specific field of interest. Several topics will be provided, but students are also encouraged to propose their own topics when applying for the seminar. In the regular meetings, students will learn about fundamentals and trends in Industry 4.0 from a machine learning perspective, common machine learning techniques and their implementation, project management of data analytics projects in businesses, as well as best practices for presentations and scientific work. The programme will be complemented by talks from invited experts in the domain. Furthermore, students will present results from literature research and data analytics projects.
6	**Lernziele und Kompetenzen**	• Students will develop an understanding of the current hot field of machine learning and data analytics in businesses • Students will learn to research and present a topic within the context of machine learning and data analytics in businesses independently

		• Students will learn to identify opportunities, challenges and limitations of corresponding ML approaches in businesses • Students will develop the skill to identify and understand relevant literature and to present their findings in a structured manner • Students will learn to present implementation and validation results in form of a demonstration and/or report
7	**Voraussetzungen für die Teilnahme**	• Prior knowledge of machine learning via courses like Pattern Analysis, Pattern Recognition, Deep Learning, Machine Learning for Time Series, or equivalent is expected. Alternatively, first data science project experience, for example as working student in a company, can be sufficient. • Motivation to explore scientific findings (e.g. via literature research) • Motivation to code and analyze data
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Seminar Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Seminarleistung
11	**Berechnung der Modulnote**	Seminarleistung (100%) • 50% of grade: Presentation (20 minutes) • 50% of grade: 4 pages IEEE standard paper (excluding references) (+ code submission)
12	**Turnus des Angebots**	in jedem Semester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 120 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	• Lei, Yaguo, Naipeng Li, Liang Guo, Ningbo Li, Tao Yan, and Jing Lin. "Machinery Health Prognostics: A Systematic Review from Data Acquisition to RUL Prediction. Mechanical Systems and Signal Processing 104 (May 2018): 799834.https://doi.org/10.1016/j.ymssp.2017.11.016. • Rojas, Eric, Jorge Munoz-Gama, Marcos Sepúlveda, and Daniel Capurro. "Process Mining in Healthcare: A Literature Review. Journal of Biomedical Informatics 61 (June 1, 2016): 22436. https://doi.org/10.1016/j.jbi.2016.04.007. • Wil M. P. van der Aalst. Process Mining: Data Science in Action 2nd edition, Springer 2016. ISBN 978-3-662-49851-4 • Wang, Lihui, and Xi Vincent Wang. Cloud-Based Cyber-Physical Systems in Manufacturing. Cham: Springer International Publishing, 2018. https://doi.org/10.1007/978-3-319-67693-7.

1	**Modulbezeichnung** 47675	**Seminar Meta Learning** (keine englischsprachige Modulbezeichnung hinterlegt!)	**5 ECTS**
2	Lehrveranstaltungen	Zu diesem Modul sind keine Lehrveranstaltungen oder Lehrveranstaltungsgruppen hinterlegt!	
3	Lehrende	Zu diesem Modul sind keine Lehrveranstaltungen und somit auch keine Lehrenden hinterlegt!	

4	**Modulverantwortliche/r**	
5	**Inhalt**	keine Inhaltsbeschreibung hinterlegt!
6	**Lernziele und Kompetenzen**	keine Beschreibung der Lernziele und Kompetenzen hinterlegt!
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	Semester: 0
9	**Verwendbarkeit des Moduls**	Seminar Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Seminarleistung
11	**Berechnung der Modulnote**	Seminarleistung (100%)
12	**Turnus des Angebots**	keine Angaben zum Turnus des Angebots hinterlegt!
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: ?? h (keine Angaben zum Arbeitsaufwand in Präsenzzeit hinterlegt) Eigenstudium: ?? h (keine Angaben zum Arbeitsaufwand im Eigenstudium hinterlegt)
14	**Dauer des Moduls**	?? Semester (keine Angaben zur Dauer des Moduls hinterlegt)
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 349413	**Seminar Sprachtechnologie für Sprachpathologien** (Speech technologies for speech pathologies)	**5 ECTS**
2	Lehrveranstaltungen	Seminar: Seminar Automatic Analysis of Voice, Speech and Language Disorders in Speech Pathologies (4 SWS)	5 ECTS
3	Lehrende	Prof. Dr.-Ing. Andreas Maier Prof. Seung Hee Yang	

4	**Modulverantwortliche/r**	Prof. Seung Hee Yang
5	**Inhalt**	This seminar deals with how the diagnosis and therapy of different speech pathologies can be supported by speech technology. The participants should present selected speech, speech and voice disorders in a lecture and demonstrate corresponding technologies in the field of pattern recognition and speech processing.
6	**Lernziele und Kompetenzen**	Die Studierenden • erlernen die Literaturrecherche. • arbeiten sich selbstständig anhand der gefundenen Literatur in die Thematik der automatischen Analyse von Sprach-, Sprech- und Stimmstörungen ein. • wählen einen Schwerpunkt und bereiten diesen im Rahmen einer Präsentation so auf, dass er für andere Teilnehmer des Seminars verständlich ist. • lernen die Anforderungen an einen wissenschaftlichen Vortrag auf einer internationalen Konferenz kennen. • halten einen Vortrag in der international üblichen Fachsprache Englisch (davon ausgenommen sind Studierende aus dem Ausland, die in Deutschland studieren, um Deutsch zu lernen)
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Seminar Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Seminarleistung
11	**Berechnung der Modulnote**	Seminarleistung (100%)
12	**Turnus des Angebots**	Unregelmäßig
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 120 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 96970	**Seminar Visual Computing** (keine englischsprachige Modulbezeichnung hinterlegt!)	**5 ECTS**
2	Lehrveranstaltungen	Masterseminar: Seminar Visual Computing (2 SWS)	5 ECTS
3	Lehrende	Prof. Dr.-Ing. Tobias Günther Prof. Dr. Tim Weyrich	

4	**Modulverantwortliche/r**	Prof. Dr.-Ing. Tobias Günther Prof. Dr. Tim Weyrich
5	**Inhalt**	This seminar covers advanced topics in visual computing, including both seminal research papers, as well as the latest research results. The seminar provides an opportunity to obtain a comprehensive overview of research questions in visual computing, as well as allows students to dive deeper into a chosen topic. Each student presents one scientific publication and explains its content to fellow students taking the course. Thereby, students practice their argumentation and presentation skills. For each paper, a supervisor is provided, who answers questions and gives pointers on the presentation slide design. The seminar is concluded with a short written report. The main topics include: • human performance capture (faces, eyes, speech), • animation (motion controllers, speech synthesis, shape modelling) • fabrication (caustic design, robot design), • appearance modelling (subsurface scattering), • Monte Carlo rendering (importance sampling, participating media), • differentiable rendering (neural rendering, inverse rendering), • denoising (non-local means and deep learning), • physics simulation (fluid simulation) The seminar contains the paper presentations by the students and introductory lectures on scientific dissemination. **Grading** Each student presents a paper, which is a selected from a set of papers in the first session. The presentation duration is 30 minutes with an additional 10 minutes for questions. Presentations begin approximately 3-4 weeks after the start of the semester. The presentation contributes to 70% of the final grade. A written report with a duration of 5-10 pages constitutes the remaining 30%, for which a LaTeX template is provided. The presentation time slots are grouped by topic and cannot be chosen. An important aspect of the grading is the subsequent discussion. To spur discussions, students are encouraged to write a brief abstract about each paper, which can be uploaded on StudOn before the presentation. The voluntary abstracts are graded and if more than 80% of the available points are reached the final grade is improved by 0.3 or 0.4 grade points, respectively.

6	**Lernziele und Kompetenzen**	Students learn to: • present current research topics in visual computing • perform a thorough literature review • cite scientific literature correctly • comprehend scientific texts • improve their presentation and argumentation skills • practice scientific writing
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Seminar Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Seminarleistung
11	**Berechnung der Modulnote**	Seminarleistung (100%)
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 120 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Deutsch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!

1	**Modulbezeichnung** 559050	**Systems- and Networks-on-a-Chip** (Systems- and networks-on-a-chip)	**2,5 ECTS**
2	Lehrveranstaltungen	Seminar: Systems- and Networks-on-a-Chip (2 SWS)	5 ECTS
3	Lehrende	Chetana Pradhan Dr.-Ing. Stefan Wildermann	

4	**Modulverantwortliche/r**	Joachim Falk Dr.-Ing. Stefan Wildermann
5	**Inhalt**	Eingebettete Systeme spielen im alltäglichen Leben eine immer größere Rolle. Gleichzeitig nimmt die Komplexität dieser Systeme immer weiter zu. Durch die heutige Technologie ist es möglich, Millionen, in naher Zukunft Milliarden von Transistoren auf einem Chip zu platzieren. Dies führt dazu, dass häufig das komplette eingebettete System, ein sogenanntes System-on-a-Chip (SoC), auf einem einzigen Chip realisiert werden kann. Die Vorteile einer verbesserten Performanz, niedrigerem Energieverbrauch sowie sinkenden Kosten sind dabei unter anderem durch die Wiederverwendung bestehender Komponenten bedingt. Eine der Herausforderungen bestehender SoCs besteht darin, eine korrekte und zuverlässige Kommunikation zwischen den Komponenten herzustellen. Aus diesem Grund wird den Komponenten eine netzwerkartige Kommunikation zur Verfügung gestellt, wodurch sogenannte Networks-on-a-Chip (NoCs) entstehen. Dieses Seminar beschäftigt sich mit der Problematik von Design, Synthese und Analyse bestehender und zukünftiger Systems- und Networks-on-a-Chip. Hierbei soll vor allem die Vereinbarkeit verschiedener Anforderungen an das System wie Kosten, Platz- und Energieverbrauch oder Zuverlässigkeit in den verschiedenen Phasen der Entwicklung betrachtet werden.
6	**Lernziele und Kompetenzen**	**Fachkompetenz - Verstehen** • Die Studierenden tragen die wesentlichen Inhalte einer ausgewählten wissenschaftlichen Veröffentlichung auf dem Gebiet der MPSoCs vor. • Die Studierenden veranschaulichen den grundlegenden Kontext der Veröffentlichung sowie deren wesentliche Neuerungen. **Lern- bzw. Methodenkompetenz** • Die Studierenden arbeiten sich selbstständig in eine wissenschaftliche Veröffentlichung ein und suchen hierbei selbstständig nach verwandten Arbeiten, um den Kontext der Veröffentlichung zu verstehen und aufzubereiten. **Sozialkompetenz**

		• Die Studierenden beteiligen sich aktiv an den Vorträgen der jeweils anderen Studierenden durch fachbezogene Fragen zum Thema wie auch Rückmeldung zu Vortragsstil.
7	**Voraussetzungen für die Teilnahme**	Keine
8	**Einpassung in Studienverlaufsplan**	keine Einpassung in Studienverlaufsplan hinterlegt!
9	**Verwendbarkeit des Moduls**	Seminar Master of Science Information and Communication Technology 20222
10	**Studien- und Prüfungsleistungen**	Seminarleistung
11	**Berechnung der Modulnote**	Seminarleistung (100%) Die Bewertung der Prüfungsleistung setzt sich zusammen aus 70% Seminarvortrag (ca. 30 Minuten Präsentation + ca. 15 Minuten Frage und Antwort) und 30% schriftlicher Ausarbeitung (2 Seiten stichpunktartiges Handout).
12	**Turnus des Angebots**	nur im Wintersemester
13	**Arbeitsaufwand in Zeitstunden**	Präsenzzeit: 30 h Eigenstudium: 45 h
14	**Dauer des Moduls**	1 Semester
15	**Unterrichts- und Prüfungssprache**	Englisch
16	**Literaturhinweise**	keine Literaturhinweise hinterlegt!