

Elektronische Systeme

Student Group

First Name	Surname	Matrikel Nr.

Table of Contents

Elektronische Systeme	2
<i>Einführung</i>	2
<i>Bisherige Kurse</i>	2
<i>Themen für Kurzvorträge</i>	3
<i>Themen für Projekte</i>	3
<i>Terminplanung</i>	4
weiterführende Links	5

Elektronische Systeme

Einführung



Source: Pixabay (CC 0 Lizenz)



Source: Pixabay (CC 0 Lizenz)



Source: Pixabay (CC 0 Lizenz)

Unterschiedlichste elektronische System umgeben uns Tag täglich: Mobiltelefone, Laptops, Fahrzeugsteuerung... In diesem Kurs wollen wir eine Systemidee von Ihnen in elektronischer Hard- und Software umsetzen, um dabei die Einblicke in die Elektronikentwicklung zu erweitern. Ziel ist vor dem Jahresabschluss bereits die Systeme zu präsentieren.

Weiterhin ergeben sich vor den Prüfungen 4 Termine, bei welchen Experten aus der Industrie interessante Einblicke in die Elektronik darstellen. Hier bin ich auf Ihr Interesse angewiesen und offen für Ideen.

Bisherige Kurse

Im letzten Kurs wurden folgende Präsentationen gehalten:

- **Wie wird eine Platine entwickelt?** (externer Vortragender, Würth/WEdirekt)
Nutzentypen, Multilayer, Flex-Systeme, Vias
- **Weitere Tipps und Tricks zum Layouting von Platinen** (Prof. Gruhler)
parasitäre Induktivitäten und Kapazitäten, Führung von analoger und digitaler Masse, Aufbau von Multilayerplatinen, Kunst des Layoutings
- **Mikrocontroller-Technologien** (externer Vortragender, Microchip)

Hardware-Software-Codesign, typische Haken und Ösen bei der Hardwareentwicklung

- **Software- und Systementwicklung** (*externer Vortragender, Bosch*)
Prozesse, industrielle Tools, Tagesablauf

Gegebenenfalls könnten folgende Themen interessant sein:

- **Wie kann ich Kommunikation und Datenablage sicher machen**
Checksummen, Zyklische Redundanzprüfung, Hash-Funktionen, Kryptographie
- **Wie wandelt man - z.B. im Elektroauto, im Netzteil, in der Endstufe - Wechselspannung in Gleichspannung und umgekehrt**
Halbbrücke, Buck/Boost-Converter, Vollbrücke, B6-Brücke, Transistortypen, Gegentaktendstufe
- * **Warum brennen Prozessoren nicht durch?**
Thermomanagement, Wärmestromkreis, transiente thermische Impedanz
- **komplexere Filter**
Bandsperr, Bessel-, Butterworth-, Chebyshev-Filter, Ausnutzung von Resonanz
- Künstliche Intelligenz

Projektvorschläge folgen bis zum Semesterbeginn. Eigene Projektvorschläge sind gerne gesehen.

Themen für Kurzvorträge

1. Transistoren - Transistorstrukturen im Chip: von Standardzellen zu 3D-Integration und „Forksheets“-Transistoren (FinFETs, vertikale Transistoren, ...)
2. Transistoren - Einführung in CMOS
3. Energiespeicher - Miniaturbatterien in Dünnschicht
4. Energiespeicher - Supercaps und Hybrid-Supercaps
5. Energiespeicher - Impedanzspektroskopie bei Batterien
6. Controller - Multicore, Pipelining und Vectoring in Prozessoren
7. Controller - DSPs
8. Controller - Hardware-integrierte Neuronale Netze (analoge Neuronen über Resistive RAM, Compute in Memory, mathematische MAC Operationen, FPGAs, ...)
9. Einführung in den thermischen Stromkreis (z.B. Berechnungen mit Zth)
10. Einführung in Photonik (optische Kopplung, Polarisationsmanagement, SiN-Wellenleiter)

Themen für Projekte

Nr.	Thema	Beschreibung	Umfang	sinnvolle Interessen		
				Softwareentwicklung?	Hardwareentwicklung?	Systementwicklung?
1	Simulide Systemübersicht / Weiterentwicklung	Das Simulationstool Simulide ist in Cpp programmiert. Aktuell ist nicht bekannt, dass es eine Systemübersicht gibt (welche einzelnen SW-Module gibt es? wie interagieren diese?). Ein Ziel ist diese zu entwickeln. Als Start können verschiedene Forumsbeiträge genutzt werden, z.B. Kompilieren unter Win10 . Weiterhin können kleine Komponenten umgesetzt werden (z.B. Motor mit "Spannungsausgang" für Drehzahl)	2	Ja, Cpp (in Windows)		Ja, Systemübersicht
2	Einarbeitung in Falstad circuitjs		2	Ja, Java Script		Ja, Systemübersicht
3	WLAN-Hookup MEXLE (als standalone Basis Board)	OTA-Flashing für AVR-Chips / Mxle 328PB über ESP32	1...2	Ja, C++ / C (embedded)	Ja, Schaltung/Layout in eagle	
4	eine "einfache" Programmiersprache: Scratch auf dem MEXLE 328PB Board	(S4A)	1...2	Ja, Scratch		Ja
5	(einfacher) Prozessornachbau in Falstad / digital	(z.B. RISC CPU)	2...4		Ja, Simulation	
6	Modellierung eines Operationsverstärkers		1...2			

7	Elektronik-Management über Bar-, QR-Code (z.B. quaggajS , automatischer Erstellung für eagle)	(z.B. einfaches Modell , komplexeres Modell)	1...2	Ja, in eagle und ggf. PHP, java Script		
8	(weitere) Anpassung MEXLE ↔ Fischertechnik		2...3		Ja, Schaltung/Layout in eagle	
9	„Leit“-Schaltungen entwickeln	Für die Fächer Grundlagen der Digitaltechnik und Elektronik-Labor sind Leitschaltungen zu entwickeln. Diese sollen den Studierenden zu Beginn des Semesters bereits eine Anwendung zeigen von dem, was Sie nach und nach erst lernen. Bei "Grundlagen der Digitaltechnik" könnte dies z.B. eine einfache Implementierung der TWI Hardware sein. Diesen sollen in geeignete MEXLE2020 Module umgewandelt werden. Beim "Elektronik-Labor" geht es um Ggf. ist dies auch für ET1, ET2, Microcontrollertechnik und ET-Labor möglich	1...2		Ja, Schaltung/Layout in eagle und Simulation	Ja, Suche nach geeigneten Schaltungen und Systemen, Simulation in Falstad / Tina / Logic
10	Magnetkreise mit FEMM auslegen (z.B. Elektromotor, Reluktanzmotor, Lautsprecher etc.)	Das Programm FEMM ermöglicht die Simulation von elektrischen und magnetischen Effekten per finite Elemente. Ziel der Aufgabe ist ein Aufbau verschiedener Magnetkreise (Gleichstrommotor, Reluktanzmotor, Synchronmaschine, Lautsprecher etc.) und eine Beschreibung wie dies umsetzbar ist zu erstellen. Die Struktur ist über ein CAD-Programm (z.B. AutoCAD) zu erstellen. Optimal ist ein Output als Animation 1...2 Ja, Simulation 11 Entwicklung von diversen Layouts (1) Intelligentes Displaymodul mit Tasten: Anbindung der Tasten und des Displays an I2C\\ (2) Funktionsgenerator-Hookup (auf AVR32DB Basisboard, welches bereits DAC und OPV hat). Umwandlung von unipolarem Signal (0..3,3V) in bipolares (-3,3V...+3,3V)\\ (3) Template für zukünftige Sensormodule 3 Ja, Schaltung/Layout in eagle Ja, Konzepterstellung für Funktion, Pinning und Anforderung an Software 12 Software I2C I2C ohne die TWI-Register eines AVR-Chips über direkte Ausgabe von Bits implementiert werden. Hierfür ist bereits eine library von http://www.peterfleury.epizy.com/avr-software.html?i=1 Peter Fleury vorhanden. Diese soll so erweitert werden, dass die I2C-Slaves dynamisch hinzugefügt werden können (ARP oder Address Resolution Protocol für "hot plug").	2...3	Ja, C (embedded)		Ja, Konzepterstellung für Umsetzung

Terminplanung

Semester-woche	Termin	Modus	Projekt
1	01.03	Einführung	Auswahl des Projekts, Komponentensuche, Einlesen in Datenblätter, Projektplanung: - Mindmap - Gantt-Chart - Ressourcenplanung
2	08.03		
3	15.03	Eigenständige Arbeit / Coaching	
4	22.03		
5	29.03		
6	05.04	ENTFÄLLT (Ostern)	
7	12.04	Eigenständige Arbeit / Coaching	
8	19.04		
-	26.04	ENTFÄLLT (Prüfungen)	
-	03.05		
-	10.05		
-	17.05		
-	24.05		
9	31.05	Präsentationsslot 1-4, Coaching	
10	07.06	Präsentationsslot 5-8, Coaching	
11	14.06	Präsentationsslot 9-12, Coaching	
12	21.06	Präsentationsslot 13-16, Coaching	
13	28.06	Präsentationsslot 17-20, Coaching	

Bei allen Veranstaltungen ist die Möglichkeit zu Rücksprachen zu den eigenen Projekten gegeben.

weiterführende Links

Theorie paralleler und verteilter Systeme von Hr. Prof. Tantau an der [Uni Lübeck](#)

Dimensionierung von Schaltnetzteilen

iPES: interaktives Power Electronics Seminar

diverse Skripte für Elektronik der ZHAW (Schweiz)

From:

<https://wiki.mexle.te.hs-heilbronn.de/> - **MEXLE Wiki**

Permanent link:

https://wiki.mexle.te.hs-heilbronn.de/elektronische_systeme/start?rev=1615134554

Last update: **2021/05/09 10:06**

