

---

**Modulbezeichnung:** **Praktikum Eingebettete Mikrocontroller-Systeme** **2.5 ECTS**  
**(semesterbegleitend) (PEMSY)**  
 (Embedded Microcontroller systems Laboratory)

Modulverantwortliche/r: Albert Heuberger

Lehrende: Markus Hartmann, Sally Nafie

---

Startsemester: WS 2019/2020

Dauer: 1 Semester

Turnus: halbjährlich (WS+SS)

Präsenzzeit: 45 Std.

Eigenstudium: 30 Std.

Sprache: Deutsch

---

**Lehrveranstaltungen:**

Praktikum Eingebettete Mikrocontroller-Systeme (semesterbegleitend) (WS 2019/2020, Praktikum, 3 SWS, Sally Nafie)

Praktikum Eingebettete Mikrocontroller-Systeme (Blockpraktikum) (WS 2019/2020, optional, Praktikum, 3 SWS, Sally Nafie)

---

**Empfohlene Voraussetzungen:**

- 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 Boole'schen Operationen
  - Englischkenntnisse
  - Deutschkenntnisse
- 

**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.

**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 Studenten 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 Studenten 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 Studenten nach dem Praktikum in der Lage folgende Kommunikationsprotokolle anzuwenden:

- Befehlssatz des LCD Controllers HD44780
- Befehlssatz eines ISM Funkmoduls

#### Literatur:

- Kernighan / Ritchie: The C Programming Language
- <http://www.like.eei.fau.de/lehre/lehveranstaltung/pemys.shtml>

---

#### Verwendbarkeit des Moduls / Einpassung in den Musterstudienplan:

Das Modul ist im Kontext der folgenden Studienfächer/Vertiefungsrichtungen verwendbar:

##### [1] Informations- und Kommunikationstechnik (Bachelor of Science)

(Po-Vers. 2009 | TechFak | Informations- und Kommunikationstechnik (Bachelor of Science) | Wahlpflichtmodule, Wahlmodule, Seminar, Praktika, Bachelorarbeit | Praktikum oder Projektarbeit | Laborpraktikum Eingebettete Mikrocontroller-Systeme (PEMSY))

Dieses Modul ist daneben auch in den Studienfächern "Berufspädagogik Technik (Master of Education)", "Elektrotechnik, Elektronik und Informationstechnik (Bachelor of Science)", "Elektrotechnik, Elektronik und Informationstechnik (Master of Science)", "Informatik (Master of Science)", "Mechatronik (Master of Science)", "Wirtschaftsingenieurwesen (Bachelor of Science)", "Wirtschaftsingenieurwesen (Master of Science)" verwendbar.

---

#### Studien-/Prüfungsleistungen:

Laborpraktikum Eingebettete Mikrocontroller-Systeme (PEMSY) (Prüfungsnummer: 75301)

Studienleistung, Praktikumsleistung

weitere Erläuterungen:

Praktikums-Aufgabenstellung lesen Praktikums-Unterlagen durcharbeiten Abschlusspräsentation mit Demonstration (10 Min.)

Erstabelleung: WS 2019/2020, 1. Wdh.: keine Wdh.

1. Prüfer: Albert Heuberger (100228)

---

#### Organisatorisches:

WS und SS, 1x semesterbegleitend in der Vorlesungszeit, 1x in der vorlesungsfreien Zeit als Blockpraktikum