

Embedded Systems Hands-On 1: Entwurf und Realisierung von Hardware/Software-Systemen



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Jaco Hofmann M.Sc., Dipl.-Inform. Andreas Engel

Sommersemester 2016

(Stand 22. April 2016)

I Einleitung

Diese Veranstaltung richtet sich an Studierende, die grundlegende praktische Kenntnisse im Entwurf und der Realisierung eingebetteter Systeme erwerben möchten.

Nach der Einführung von wichtigen Konzepten und Techniken wie

- Grundlagen der Elektrotechnik
- Umgang mit Laborelektronik
- Entwurf und Realisierung von elektronischen Schaltungen
- Sensordaten: Erfassung und Verarbeitung
- Bus-Systeme in eingebetteten Systemen
- Programmieren und Debuggen von heterogenen eingebetteten Systemen
- Linux Kernel in eingebetteten Systemen

entwickeln die Teilnehmerinnen und Teilnehmer auf Basis des zuvor Gelernten ein eigenes eingebettetes System. Dabei stehen verschiedene Projekte zur Auswahl, welche je nach eigenen Interessen eine Fokussierung auf die Software- oder die Hardware-Entwicklung erlauben.

II Organisation

Anmeldung

Die Anmeldung zum Praktikum als Prüfungsleistung erfolgt über [TUCaN]. Wegen der begrenzten Ressourcen unseres Praktikumsraums ist die Teilnehmerzahl auf 12 Personen limitiert. Bei zu großer Nachfrage muss daher ein Losverfahren über die Teilnahme entscheiden.

Kommunikation

Für die Diskussion eventueller Unklarheiten steht Ihnen ein [Moodle] Forum zur Verfügung, für welches Sie in nach der Einführungsveranstaltung angemeldet werden. Nutzen Sie dieses als erste Anlaufstelle und posten Sie hier noch nicht beantwortete Fragen. Dadurch soll allen Praktikumsmitgliedern der gleiche Kenntnisstand zugänglich sein. Es wird empfohlen, das Aufgaben- und Nachrichtenforum des Moodle-Kurses zu abonnieren.

Zeitlicher Ablauf

KW	Datum	Uhrzeit	Ort	Inhalt
15	13.04.2016	10:30-11:30	A126	Einführungsveranstaltung
16	13.04.2016	10:30-11:30	E202	Vorstellung Aufgabe 1: Embedded Linux
18	20.04.2016	10:30-11:30	E202	Vorstellung Aufgabe 2: Debugging
20	03.05.2016	10:30-11:30	E202	Vorstellung Aufgabe 3: Baremetal ARM Programmierung
21	18.05.2016	10:30-11:30	E202	Vorstellung Aufgabe 4: Sensoransteuerung über digitale Busse
23	25.05.2016	10:30-11:00	E202	Grundlagenvorlesung Elektrotechnik
23	08.06.2016	11:00-12:00	E202	Vorstellung Aufgabe 5: Digitale und Analoge Filter
25	08.06.2016	10:30-11:30	E202	Vorstellung Aufgabe 6: Analoge Ausgabe
26	22.06.2016	10:30-11:30	E202	Vorstellung Aufgabe 7: Projektthemen + Themenvergabe
28	29.06.2016	23:59	GitLab	Elektronische Abgabe für Übungen (Aufgaben 1-6)
36	17.07.2016	23:59	GitLab	Elektronische Abgabe für Projekt (Aufgabe 7)
38+				Durchführung von Kolloquien

Die Einführungsveranstaltung findet nach Absprache mit den angemeldeten Praktikumsmitgliedern in der ersten Vorlesungswoche statt und dient der Vorstellung der Praktikumsorganisation und der Einweisung in den Praktikumsraum. Dabei werden die Teilnehmer außerdem in Zweiergruppen eingeteilt.

Die Aufgaben 1 bis 6 sind von allen Gruppen gleichermaßen zu bearbeiten. In der oben genannten Kalenderwoche wird dazu die Aufgabenstellung von einem der Tutoren vorgestellt, wobei ggf. auch Hinweise zur Umsetzung gegeben werden, welche über die in dieser Anleitung enthaltene Aufgabenstellung hinaus geht. Der Besuch dieser Vorstellungen ist jedoch nicht verpflichtend.

Eine gesonderte Vorlesung zur Einführung in die wichtigsten elektrotechnischen Grundkenntnisse ist als Auffrischung insbesondere für solche Studenten gedacht, die sich kaum noch an ihren letzten Physikunterricht erinnern können. Die Teilnahme ist ebenfalls freiwillig.

Nachdem mit den Aufgaben 1 bis 6 die wichtigsten Grundlagen zum Umgang mit Software, Hardware und Messgeräten gelegt wurden, sollen diese in einem größeren Projekt praktisch angewendet werden. Dazu stehen für die Aufgabe 7 verschiedene Themen zur Verfügung, aus denen sich jede Gruppe eines auswählt. Die doppelte Vergabe eines Themas ist nicht möglich - im Zweifel ent-

scheidet das Losverfahren. In Absprache mit den Tutoren können aber auch eigene Projektideen umgesetzt werden.

Die Lösung aller Aufgaben erfolgt eigenverantwortlich (keine Präsenzübung mit permanenter Betreuung). Die Abgabe der Lösungen erfolgt elektronisch per Git an die Tutoren zum oben genannten Termin.

Das Praktikum wird gruppenweisen Kolloquien abgeschlossen. Dabei wird insbesondere überprüft, ob beide Gruppenteilnehmer aktiv an der Erarbeitung der Lösungen mitgewirkt haben. Die Teilnahme am Kolloquium ist daher verpflichtend.

Bewertung

Für jede Aufgabe und das anschließende Kolloquium erfolgt eine Bewertung am Ende des Praktikums. Die Gewichtung der einzelnen Teilnoten für die Gesamtnote richtet sich nach der vorgesehenen Bearbeitungszeit, wobei für das Kolloquium 3 Wochen angesetzt werden. Damit ergibt sich folgender Bewertungsschlüssel

Aufgabe	1	2	3	4	5	6	7	Kolloquium
Bearbeitungszeit	2	2	1	2	2	1	11	3
Gewichtung [%]	8	8	4	8	8	4	48	12

Neben der eigentlichen Lösungsfindung geht auch Dokumentation von Lösung und Lösungsweg in die Bewertung mit ein. Dazu zählen unter anderem Verweise auf genutzte Quellen und Kommentierung von Programm-Code. Insbesondere bei den Prokekttehen (Ausgabe 7) steht die Herangehensweise an die Aufgabe (Strukturierung in Teilprobleme, Abwägen von Lösungsalternativen, Zeitmanagement) und der Umgang mit unvorhergesehenen Problemen im Mittelpunkt.

Die Bewertung erfolgt prinzipiell gruppenweise. Einzelne Gruppenmitgliedern können jedoch auch von der Bewertung ausgeschlossen werden, wenn diese erkennbar nicht maßgeblich zur Gesamtleistung der Gruppe beigetragen haben.

Arbeitsplätze

Im Praktikumsraum (S2|02 E104) des Fachgebiets sind 5 Arbeitsplätze eingerichtet, deren Software- und Hardware-Ausstattung in den Abschnitten II und III beschrieben ist. Der Login-Account ihrer Gruppe wird nach der Einführungsveranstaltung erstellt. Alle Geräte sind ausschließlich innerhalb ihrer Spezifikation zu betreiben. Im Zweifelsfall ist mit einem Tutor Rücksprache zu halten. Vor Verlassen des Arbeitsplatzes werden alle Versuchsanordnungen abgebaut und alle Geräte, Werkzeuge und Kabel an die dafür vorgesehenen Positionen zurück verbracht. Nicht mehr verwendbares Verbrauchsmaterial ist zu entsorgen. *Das Verzehren von Speisen und Getränken im Praktikumsraum ist untersagt.*

Plagiarismus

Der Fachbereich Informatik misst der Einhaltung der Grundregeln der wissenschaftlichen Ethik großen Wert bei. Zu diesen gehört auch die strikte Verfolgung von [Plagiarismus].

III Verwendete Software

Für verschiedene Teilaufgaben des Praktikums wird frei verfügbare Software eingesetzt, die auf den Rechnern im Praktikumsraum bereits vorinstalliert ist. Die Programme sind idR. quelloffen und als vorkompilierte Installationspakete für Windows, Mac OS und diverse Linux Distributionen verfügbar. In diesem Abschnitt werden die Bezugsquellen und der Verwendungszweck dieser Werkzeuge kurz zusammengefasst. Für eine detaillierte Benutzeranleitung verweisen sei auf die jeweilige Dokumentation Programme verwiesen

ARM Entwicklungsumgebung

Der verwendete Microcontroller wird im Rahmen des Praktikums mit C und C++ programmiert. Als Compiler wird der [ARM-GCC] 5.2 verwendet. Für das Debuggen und Flashen des Chips wird [OpenOCD] 0.9 auf dem Raspberry PI verwendet. Debuggen erfolgt über den von OpenOCD bereitgestellten GDB Server. Es wird empfohlen eine Entwicklungsumgebung wie [Eclipse] zum Debuggen zu verwenden.

Quite Universal Circuit Simulator

Zur Simulation von elektronischen Schaltkreisen wird [Qucs] 0.0.18 verwendet. Dabei kann der zeit- bzw. frequenzabhängige Verlauf von Strom und Spannung an passiven (Widerstände, Kondensatoren, Spulen) und aktiven Bauteilen (Dioden, Transistoren, Operationsverstärker etc.) beobachtet werden, noch bevor die entsprechenden Schaltkreise aufgebaut wurden. Dies erleichtert das Verständnis der Schaltungen und das Auslegen von Bauteilparametern.

KiCad EDA

Für das Erstellen von Platinenlayouts für elektronische Schaltkreise wird [KiCad] 4.0.2 verwendet. Nach dem Erstellen eines Schaltplans werden dabei die einzelnen Komponenten auf verschiedenen Ebenen eines PCBs platziert und verdrahtet. Die eigentliche Prototypen-Fertigung erfolgt im Rahmen dieses Praktikums mittels Lochraster-Komponenten.

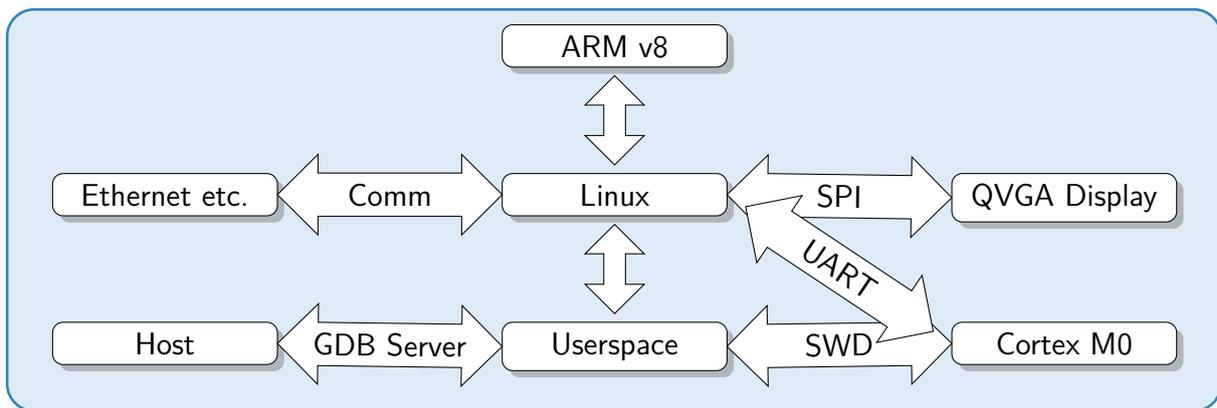
Git

Zur Verwaltung und Abgabe erstellten Dokumente wird das Versionskontrollsystem [Git] verwendet, welches aktuell in der Version 2.8.1 verfügbar ist. Damit können die Gruppenmitglieder gemeinsam und strukturiert an den einzelnen Teilaufgaben arbeiten. Jede Gruppe erhält dafür einen eigenen Bereich (Repository) innerhalb eines [GitLab] Servers. Die entsprechenden Zugangsdaten werden nach der Gruppeneinteilung verteilt. Die jeweils letzte vor der relevanten Deadline als **Abgabe** markierte Version des Repositories (siehe git tag) bildet die Bewertungsgrundlage des jeweiligen Praktikumsteils (Übungen oder Projekt).

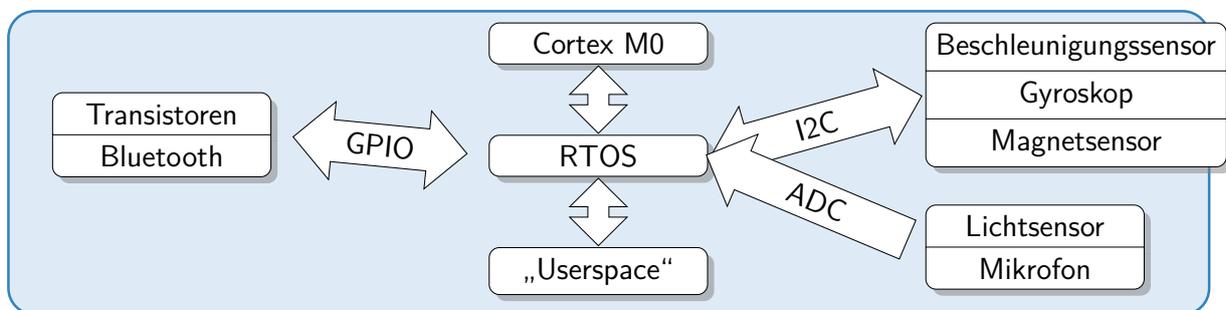
IV Verwendete Hardware

Die verwendete Hardware gliedert sich in drei Teile. Die Basis des Systems bildet ein Raspberry Pi 3 mit vier leistungsfähigen ARMv8 Kernen. Als weiterer Teil des heterogenen Systems wird ein Aufsteckboard verwendet, das einen ARM Cortex M0 Mikrocontroller und einige Sensoren sowie ein QVGA Display enthält. Des Weiteren werden zum Debuggen und überprüfen der Teile des Systems verschiedene Messgeräte verwendet.

Raspberry PI



Erweiterungsboard



Laborelektronik

Für die Analyse der Kommunikation zwischen Recheneneinheiten und deren Peripheriekomponenten (Sensoren/Aktoren) wird ein Adapter namens [Bus-Pirate] verwendet. Mit diesem kann die auf digitalen Bussen (bspw. UART, SPI, I²C, JTAG) übertragene Byte-Sequenz auf der seriellen Konsole eines PCs beobachtet oder generiert werden. Jeder Arbeitsplatz im Praktikumsraum E104 ist mit dieser Hardware ausgestattet.

Für die Aufgaben mit elektrotechnischen Hintergrund (ab Aufgabe 5) müssen auch analoge Signale generiert und vermessen werden. Hierfür wird ein Agilent [33522A] Funktionsgenerator und eine Hameg [HMO2524] Mixed-Signal Oszilloskop verwendet. Davon stehen jeweils zwei Geräte zu Verfügung, die Ihnen auf Anfrage von einem der Betreuer ausgehändigt werden. Die Arbeit mit diesen Geräten findet ausschließlich im Praktikumsraum E104 statt.

Je nach gewähltem Projektthema werden auch Lochraster-Prototypen von Hardware-Komponenten gefertigt. Dafür steht eine Weller [WX2021] Lötstation im Elektroniklabor B014 zur Verfügung. Diese ist ausschließlich nach Einweisung durch einen Praktikumsbetreuer zu verwenden.

Sprechen Sie sich mit den anderen Gruppen des Praktikums ab, um die Verwendung der limitierten Laborelektronik (Signalgenerator, Oszilloskop und Lötstation) zu koordinieren. Sollte es hierbei zu Konflikten kommen, werden den einzelnen Gruppen individuelle Benutzungszeiten fest zugeteilt.

Aufgabe 1: Embedded Linux

Siehe [Praktikums-Webseite](#) oder [Moodle-Kurs](#).

Aufgabe 2: Debugging

Siehe [Praktikums-Webseite](#) oder [Moodle-Kurs](#).

Aufgabe 3: Baremetal ARM Programmierung

Siehe [Praktikums-Webseite](#) oder [Moodle-Kurs](#).

Aufgabe 4: Sensoransteuerung über digitale Busse

Siehe [Praktikums-Webseite](#) oder [Moodle-Kurs](#).

Aufgabe 5: Digitale und Analoge Filter

Siehe [Praktikums-Webseite](#) oder [Moodle-Kurs](#).

Aufgabe 6: Analoge Ausgabe

Siehe [Praktikums-Webseite](#) oder [Moodle-Kurs](#).

Aufgabe 7: Projektthemen

Siehe [Praktikums-Webseite](#) oder [Moodle-Kurs](#).

Literatur

- 33522A** Agilent. *Wellenformgenerator der Reihe 33500*. 2012. URL: <http://cp.literature.agilent.com/litweb/pdf/33500-90911.pdf> (visited on 2016-04-21).
- ARM-GCC** Launchpad. *GCC ARM Embedded*. 2016. URL: <https://launchpad.net/gcc-arm-embedded> (visited on 2016-04-08).
- Bus-Pirate** Seeed Studio. *Bus Pirate v4*. URL: <http://www.seeedstudio.com/depot/Bus-Pirate-v4-p-740.html> (visited on 2016-04-21).
- Eclipse** *C/C++ Development Tooling*. 2016. URL: <https://eclipse.org/cdt/> (visited on 2016-04-08).
- Git** URL: <https://git-scm.com/> (visited on 2016-04-22).
- GitLab** *Code, test, and deploy together*. URL: <https://about.gitlab.com/> (visited on 2016-04-22).
- HMO2524** Hameg. *250/350 MHz Digital Oszilloskop HMO Serie*. 2012. URL: https://cdn.rohde-schwarz.com/hameg-archive/HAMEG_MAN_EN_HM0352x_2524.pdf (visited on 2016-04-21).
- KiCad** Jean-Pierre Charras, Dick Hollenbeck, and Wayne Stambaugh. *A Cross Platform and Open Source Electronics Design Automation Suite*. 2015. URL: <http://kicad-pcb.org/> (visited on 2016-03-31).
- Moodle** *Embedded Systems Hands-On 1*. 2016. URL: <https://moodle.informatik.tu-darmstadt.de/course/view.php?id=46> (visited on 2016-03-31).
- OpenOCD** 2016. URL: <http://openocd.org/> (visited on 2016-04-08).
- Plagiarismus** Andreas Koch and Wolfgang Heenes. *Grundregeln der wissenschaftlichen Ethik am Fachbereich Informatik*. 2011. URL: www.informatik.tu-darmstadt.de/plagiarism (visited on 2016-03-31).
- Qucs** *Quite Universal Circuit Simulator*. 2015. URL: <http://qucs.sourceforge.net/> (visited on 2016-03-31).
- TUCaN** *20-00-0959-pr Embedded System Hands-On 1: Entwurf und Realisierung von Hardware/Software-Systemen*. 2016. URL: <https://www.tucan.tu-darmstadt.de/scripts/mgrqcgi?APPNAME=CampusNet&PRGNAME=COURSEDETAILS&ARGUMENTS=-N254716612169837,-N000326,-NO,-N359984841054712,-N359984841044713,-NO,-NO,-NO> (visited on 2016-03-31).
- WX2021** Weller. *WX 1, WX 2, WXD 2, WXA 2 Originalbetriebsanleitung*. URL: http://mediaweller.de/weller/data/OI/Manual_WX_Units_Quick_Start.pdf (visited on 2016-04-21).