



**Hochschule Augsburg  
Fakultät für Informatik  
Dr. Hubert Högl  
Sommersemester 2014**

## **Prüfung im Fach Embedded Linux**

Datum: **4. Juli 2014** | Uhrzeit: **13:30 - 15:00** | Raum: **W3.22**

Hinweise:

- Dieses Angabenblatt hat auch eine Rückseite. Bitte sofort prüfen!
- Sie dürfen das Angabenblatt nach Hause nehmen. Schreiben Sie die Lösungen auf die separat ausgeteilten karierten Bögen.
- In eckigen Klammern stehen die erzielbaren Punkte.
- Bitte nicht mit Bleistift und nicht mit roter Farbe schreiben.

1. Fragen zum Artikel von Cliff Brake, "Tips for planning an embedded Linux project" [16 Punkte, je 2 Pkt.]
  - a. Welche Anforderungen sprechen für Linux? .. (5 Stück)
  - b. Wie "bezahlt" man die Vorteile von Linux?
  - c. Aus welchen Einzelteilen besteht Embedded Linux? .. (6 Stück)
  - d. Welche Fähigkeiten braucht man, um einen Rechner mit Embedded Linux auszustatten und zu programmieren?
  - e. Welche Kosten sind zu erwarten?
  - f. Welche Risiken gibt es?
  - g. Was umfasst der Lebenszyklus des Produktes?
  - h. Was erwartet Sie hinsichtlich der Lizenzen, wenn Sie Linux wählen?

2. Fragen zum Kapitel 1 im Buch von Hallinan [14 Punkte]

- a. Warum Linux? (6 Gründe) [6]
- b. Was bedeutet die "GPL"? [4]
- c. Was ist "Open Source"? [2]
- d. Was verstehen Sie unter "Linux Standard Base"? [2]

3. Beschreiben Sie knapp die technischen Eigenschaften des Embedded Linux Boards, das Sie in Ihrem Projekt verwendet haben. Denken Sie an folgende Punkte: [8 Punkte, je 2]

- Mikrocontroller (Taktfrequenz, CPU-Typ)
- Nicht-flüchtige Speicher
- Arbeitsspeicher
- Ungefähre Leistungsaufnahme

4. Fragen zum Entwicklungsrechner ("Hostrechner") [16 Punkte, je 4]

- a. Für welche Aufgaben wird der Hostrechner (Entwicklungsrechner) verwendet?
- b. Welche Programme installieren Sie darauf?
- c. Nennen Sie zwei Terminalprogramme und geben Sie die üblichen Aufrufparameter an.
- d. Was ist ein Terminal Multiplexer und warum ist er auf dem Targetrechner sehr praktisch?

5. Fragen zur Speichertechnik [18 Punkte]

1. NAND Flash Bausteine [12]

1. Was steckt in einer MicroSD Karte?
2. Welche Filesysteme verwendet man meist bei MicoSD Karten?
3. Wann verwendet man das "MTD" Subsystem?
4. Was ist JFFS2 und wozu braucht man es?
5. Welche Vorteile hat das YAFFS im Vergleich zu JFFS2?
6. Was ist das CRAMFS?

2. Was bedeutet "XIP"? [2]

3. Hauptspeicher [4]

1. In welcher Speichertechnologie ist der *Hauptspeicher* des Embedded Linux Rechners realisiert?
2. Welche minimale Grösse des Hauptspeichers sollte man haben, damit man aktuelle Distributionen ohne Einschränkungen verwenden kann (ohne grafische Oberfläche)?

6. Fragen zum Bootloader [6 Punkte]

- a. Wozu braucht man einen Bootloader?
- b. Nennen Sie zwei gebräuchliche Bootloader.

- c. Wieso kann der *first level bootloader* im Mikrocontroller im Allgemeinen nicht den allgemeinen Bootvorgang des Linux Kernels einleiten?
- d. Wozu dient die folgende U-Boot Kommandosequenz:

```
set bootargs root=/dev/nfs nfsroot=192.168.1.1:/srv/rootfs \
ip=192.168.1.2:192.168.1.1::255.255.255.0::eth0:none
nfs 0x10400000 /srv/rootfs/boot/uImage
bootm
```

7. Welche Möglichkeiten gibt es, die Bootzeiten zu reduzieren? [6 Punkte]

8. Fragen zum Kernel [10 Punkte, je 2 Pkt.]

- a. Auf der Tafel hatten wir ein Diagramm gezeichnet mit den wichtigsten Bereichen, mit denen man beim Kernel zu tun hat. Zeichnen Sie es hin.
- b. Wieso ist es wünschenswert, dass ein Embedded Linux Board im *Mainline Kernel* unterstützt wird.
- c. Geben Sie die Kommandozeilen an, um einen Kernel mit einem Cross-Kompiler zu kompilieren (Konfiguration erstellen, Kernel und Module bauen, Installation).
- d. Wieso kann es wichtig sein, dass man bei einem Embedded System den Kernel auf eine neuere Version updaten kann?
- e. Gibt es im Kernel Quelltext auch eine Dokumentation?

9. Ansteuerung von Peripherie [10 Punkte]

- a. Nennen Sie fünf verschiedene Arten, wie Sie Hardware-Erweiterungen an Ihr Embedded Linux Board elektrisch anschliessen könnten. [5]
- b. Welche alternativen Möglichkeiten gibt es zur programmiertechnischen Ansteuerung von Hardware-Erweiterungen aus dem Userspace? [5]

10. Fragen zur Echtzeit mit Embedded Linux [6 Punkte, je 2]

- a. Welche prinzipiellen Lösungsansätze gibt es, um Linux mit Echtzeit-Eigenschaften auszustatten?
- b. Wie heissen die seit Jahren etablierten praktischen Implementierungen der prinzipiellen Lösungsansätze?
- c. Welchen Vorteil hat der "Preempt-RT" Ansatz, auch wenn er nicht für harte Echtzeit geeignet ist.

11. Beschreiben Sie, wie das Debuggen über die JTAG-Schnittstelle des Mikroprozessors funktioniert [8 Punkte]

- a. In welchem Fall ist diese Debug-Art unbedingt notwendig?
- b. Wie sieht die Verschaltung der nötigen Einzelteile aus?
- c. Welches freie Programm wird auf dem Hostrechner benötigt, so dass man mit dem GNU Debugger `gdb` über JTAG debuggen kann? Welche Schnittstellen stellt das Programm bereit

(Diagramm)?

- d. Neben dem Debuggen gibt es eine weitere wichtige Funktion, die häufig über JTAG erledigt wird. Denken Sie an frisch aus der Fertigung kommende Boards.

12. Fragen zum Artikel *Linux Debugging* von Tim Schürmann aus der Leseliste [8 Punkte]

- a. In welchen unterschiedlichen Varianten kann man mit dem GDB Programme auf dem Zielrechner debuggen? [2]
- b. Wozu dient das Programm `strace`? [2]
- c. Wozu dient das Programm `LTTng`? [2]
- d. Wozu dient das Programm `systemtap`? [2]

**ENDE**