

## 2. Übungsblatt

(Ausgabe: 3. November 2016 — Abgabe bis: 10. November 2016, 4:00)

### Aufgabe 1: Knight Rider

(30 Punkte)

In dieser Aufgabe sollen Sie mit Hilfe von Memory-Mapped-I/O die LEDs auf dem Beaglebone hintereinander als Lauflicht von links nach rechts aufleuchten lassen (Ähnlich wie bei KITT, <http://bit.ly/2eRieU6>). Da die Kommunikation über die GPIO-Pins nur unter Linux möglich ist, booten Sie für diese Aufgabe bitte das Standard-Linux auf dem Beaglebone. Hierfür entfernen Sie im ausgeschalteten Zustand einfach die SD Karte.

Die LEDs sind an die Pins 22 bis 25 (von rechts gezählt) von GPIO Modul 1 (GPIO1) des BeagleBone angeschlossen und lassen sich über den dazugehörigen Speicherbereich ansprechen. Suchen Sie im bereitgestellten Handbuch des Mikroprozessors nach dem entsprechenden Speicherbereich und den nötigen Registern bzw. dem nötigen Register um die LEDs an- oder auszuschalten. Schauen Sie sich unter anderem besonders den Abschnitt 25.3.4.2 an.

Der Standardtrigger der LEDs sollte zuvor ausgeschaltet werden. Dazu dürfen die Dateien `/sys/class/leds/beaglebone:green:usr0..3/tigger` jeweils nur noch das Wort `none` enthalten.

Nutzen Sie die Funktion `open` und `mmap` um ein Memory-Mapped-File zu erstellen. Nutzen Sie Konstanten für die Adresse und den Offset der Register. Speichern Sie ihren Quelltext in einer Datei namens `led_blink.c`.

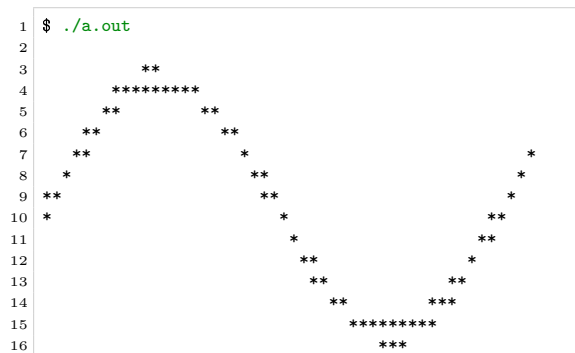
**Hinweis:** Bit-Operatoren, wie zum Beispiel `&`, `|`, `~` oder `<<` könnten hilfreich sein.

### Aufgabe 2: Mathe auf der Konsole

(15 Punkte)

Geben Sie auf der Konsole einen Graphen der Sinusfunktion `sin(3)` aus. Der kann etwa wie folgt aussehen, sollte jedenfalls als Sinuskurve erkennbar sein.

Die Kurve muss zur Laufzeit berechnet werden. Versuchen Sie die Lösung möglichst einfach, wenn auch nicht effizient zu halten. Die Kenntnisse aus der ersten Vorlesung sind dafür ausreichend.



### Aufgabe 3: Kommentare entfernen

(15 Punkte)

Schreiben Sie ein Programm namens `commex.c`, das C-Quellcode auf der Standardeingabe erhält und ihn ohne Kommentare (`/*... */`, `// ...`) wieder ausgibt.

Gehen Sie davon aus, dass Ihnen der kompilierbare C-Quellcode vorliegt. (Mehrzeilige Kommentare dürfen zum Beispiel nicht verschachtelt werden.) Die Kommentare, die in den Stringliteralen (`/* ... */`) oder Zeichenkonstanten (`'/* ... */'`) vorkommen, dürfen nicht entfernt werden. Letzteres ist zwar ein Fehler, wird aber vom Compiler erlaubt.

**Hinweis 1** Zur Lösung wird die Funktion `getchar(3)` benötigt.

**Hinweis 2** Nutzen Sie einen endlichen Automaten.

**Hinweis 3** Nutzen Sie das Programm `commex-example.c` um die Korrektheit ihrer Lösung zu testen.

```
1 $ cat commex-example.c | ./a.out
```