



**Hochschule
Augsburg** University of
Applied Sciences

Praktikum Embedded Linux

Embedded Linux
Hochschule Augsburg
Fakultät für Informatik (Prof. Dr. Hubert Högl)
Studiengang Technische Informatik, 4. Semester

Sommersemester 2016

Datum: 2016-07-05 18:04

Philip Manke, <Philip.Manke@hs-augsburg.de>, #945072

Schlafüberwachung mit dem Beaglebone Black



Dieser Text steht unter der Creative Commons Lizenz "Namensnennung/Keine kommerzielle Nutzung"

<http://creativecommons.org/licenses/by-nc/3.0/de/>

Inhalt

1	Einleitung	2
2	Vorbereiten des Rechners	2
3	So gehen Sie vor	3
4	So funktioniert es	4
5	Die Lizenz	4
6	Ein paar Fomatierungshinweise	4
7	Mögliche Verbesserungen	6
8	Literatur	6

1 Einleitung

Dieser Bericht bezieht sich auf den Versuch V2 "Das Arduino Projekt".

Es sind keine expliziten Versuchsanweisungen gegeben, außer di

Sie müssen in Ihrem Studium öfter Texte wie zum Beispiel Praktikums- und Versuchsberichte abgeben. Als langjähriger "Empfänger" von diesen Texten habe ich mir mal überlegt, wie ein "idealer" Bericht sowohl aus Ihrer, als auch aus meiner Sicht aussehen könnte:

Studentische Sicht:

- Einfach zu schreiben in einem gewöhnlichen (Programmier-)Editor.
- Die Titelseite wird automatisch erstellt.
- Ein einheitliches Format kann in HTML und PDF umgewandelt werden.
- Funktioniert mit Linux und Windows

Aus meiner Sicht:

- Einheitliches Aussehen
- Gut in Web-Dokument umwandelbar
- Format einfach automatisiert weiterverarbeitbar, z.B. in gedruckte Berichtsammlungen
- Kleine Dateien

Dieses einheitliche Format gibt es bereits, es nennt sich "reStructuredText" (reST). Es entstand vor einigen Jahren in der Welt der Programmiersprache Python. Die gesamte Python Dokumentation wird mittlerweile in diesem Format

geschrieben und mit [\[SPHINX\]](#) automatisch in die Darstellungen Text, HTML und PDF umgewandelt, siehe [\[PYDOC\]](#). Die reST Homepage ist [\[REST\]](#).

2 Vorbereiten des Rechners

Windows Sie brauchen den Python Interpreter [\[PYTHON\]](#) und das Docutils Paket [\[DOCUTILS\]](#). Das reicht für die Ausgabe von HTML Dateien. Falls Sie ausserdem Dokumente im PDF Format erzeugen wollen, müssen Sie *TeX* installieren, ich empfehle [\[MIKTEX\]](#) oder [\[TEXLIVE\]](#).

Linux Sie brauchen den Python Interpreter [\[PYTHON\]](#) und das Docutils Paket, siehe [\[DOCUTILS\]](#). Das reicht für die Ausgabe von HTML Dateien. Falls Sie ausserdem Dokumente im PDF Format erzeugen wollen, müssen Sie *TeX* installieren, ich empfehle [\[TEXLIVE\]](#). Sie sollten auch noch diese Zusatzpakete für Latex installieren: `texlive-latex-extra` und `texlive-lang-french`.

3 So gehen Sie vor

In diesem [Verzeichnis](#) finden Sie das aktuelle ZIP Archiv des Demo-Berichtes. Sie holen und entpacken es, dann sollten folgende Dateien in dem entpackten Unterverzeichnis Demo-Bericht zu finden sein:

```
bericht.cfg  bericht.rst  etc/  Makefile  VERSION
```

Den Bericht tippen Sie in `bericht.rst`. Sie können alle Markup Anweisungen von Restructured Text verwenden, eine gute Schnellreferenz finden Sie in [\[QUICKREF\]](#).

In `bericht.cfg` schreiben Sie den Titel des Berichtes, alle Namen mit Email-Adressen und Matrikelnummer der Projektgruppe. Auch das aktuelle Semester sollten Sie hier eintragen.

Linux

Sie können den Bericht durch ein Makefile sowohl im HTML- als auch im PDF-Format ausgeben. Geben Sie dazu das Kommando `make html` oder `make pdf` ein. Damit dies funktioniert muss auf Ihrem Rechner das Paket `python-docutils` installiert sein. Darin gibt es unter anderem die Skripte `rst2html` und `rst2latex`, die das Makefile verwendet.

Sie können alle generierten Dateien wieder löschen mit dem Kommando `make clean`. Sollten die erzeugten Ausgabeformate nicht das enthalten, was Sie wollen, dann kann oft ein `make clean vor` `make html` oder `make pdf` helfen.

Windows

Sie haben zwei Möglichkeiten:

1. Installieren Sie die Unix Utilities und verwenden Sie wie auf Linux `make`.
2. Rufen Sie die Kommandos von Hand auf (oder schreiben Sie eine Batch Datei):

HTML:

```
rst2html --language=de --stylesheet=etc/goodger.css \
bericht.rst bericht.html
```

PDF:

```
rst2latex --language=de \
--stylesheet=etc/pygments-docutilsroles.sty \
bericht.rst bericht.tex
pdflatex bericht.tex
pdflatex bericht.tex
```

Sie müssen zweimal `pdflatex` laufen lassen, damit das Inhaltsverzeichnis erstellt wird.

4 So funktioniert es

Damit das Deckblatt des Berichtes einheitlich aussieht, müssen Sie es gar nicht selber schreiben, sondern es wird vom Computer automatisch erstellt. Sie müssen nur ein paar Daten in die Datei `bericht.cfg` eingeben. Die eingegebenen Daten sind in einem bestimmten Format, das von "empy" -- einem einfachen, in Python geschriebenen Template Programm -- verwendet wird. Sie finden empy in `etc/em.py`. Empy ersetzt in `etc/kopf.rst.in`, `etc/titelseite.html.in` und `etc/titelseite.tex.in` die Templates durch den tatsächlichen Wert, der aus der Konfigurationsdatei genommen wird und erstellt dann die Dateien *ohne* die Endung `.in`. Die Homepage von Empy ist [\[EMPY\]](#).

5 Die Lizenz

Ich schlage vor, dem Bericht eine einheitliche Lizenz zu geben und zwar die Creative Commons Lizenz <http://creativecommons.org/licenses/by-nc/3.0/de/>. Diese Lizenz finden Sie voreingestellt auf jeder Titelseite. Wenn Sie als Autorin und Autor nicht zustimmen, dann dürfen Sie gerne eine andere Lizenz wählen. Schreiben Sie mir eine E-mail an [<Hubert.Hoegl@hs-augsburg.de>](mailto:Hubert.Hoegl@hs-augsburg.de) falls Sie eine andere Lizenz wollen.

6 Ein paar Fomatierungshinweise

Die [QUICKREF] zeigt Ihnen schnell die wichtigsten allgemeinen Formatierungsbefehle, wie Schriftstile, Überschriften, Listen, Aufzählungen und so weiter.

Da Sie im Praktikum häufig mit Quelltext zu tun haben werden, zeige ich Ihnen, wie man Quelltext einbindet, so dass die Syntax hervorgehoben wird (zumindest in der HTML Ausgabe). Hier ist ein Beispiel für C Code:

```
1 int main()
2 {
3     printf("Hello World\n");
4     exit 0;
5 }
```

Der reST Quelltext sieht dazu so aus:

```
.. code:: c
   :number-lines: 1

   int main()
   {
       printf("Hello World\n");
       exit 0;
   }
```

Das kleine c hinter code:: gibt die Sprache an, es gibt für fast jede Sprache ein Kürzel, siehe <http://pygments.org/languages>.

Hinweis: Damit die Syntax bei eingebundenem Quelltext hervorgehoben wird, muss man eine relativ moderne Docutils Version verwenden. Bei Versionen *kleiner als* 0.9.1 klappt es nicht! So findet man die Version heraus:

```
$ rst2html --version
rst2html (Docutils 0.9.1 [release], Python 2.6.5, on linux2)
```

Sie können Code auch aus einer Datei importieren, das geht so:

```
.. include:: etc/demo.c
   :code: c
```

So sieht das Ergebnis dann aus:

```
int main()
{
    int i;
    for (i=0; i<10; i++) {
        printf("%d\n", i);
    }
    exit 0;
}
```

Abbildungen bauen Sie wie folgt ein:

```
.. figure:: img/python-powered-w-200x80.png
   :align: center

   Hier sind alle 'Python Logos
   <http://www.python.org/community/logos/>'..
```

So sieht das Ergebnis aus:



Abbildung 1: Hier sind alle [Python Logos](http://www.python.org/community/logos/).

7 Mögliche Verbesserungen

- Man könnte das Programm anpassbar auf andere Veranstaltungen machen. Das heisst, der Name der Veranstaltung (Versuche aus der Technischen Informatik) und die Person, bei der abzugeben ist, könnten konfigurierbar sein. Ich würde mich freuen, wenn Studenten dieses Ideen in die Tat umsetzen würden
- Wahrscheinlich macht mein Ansatz mit `make` Probleme unter Windows. Hier sollte noch eine Lösung gefunden werden, wie man unter Windows bequem mit der Bericht-Umgebung arbeiten kann.

Ich freue mich auf Ihre Nachricht an <Hubert.Hoegl@hs-augsburg.de>.

8 Literatur

Literatur

[DOCUTILS] Docutils Paket

<http://docutils.sourceforge.net>

[REST] reST Homepage

<http://docutils.sourceforge.net/rst.html>

[QUICKREF] reST Quick Reference

<http://docutils.sourceforge.net/docs/user/rst/quickref.html>

[PYTHON]	Python Interpreter http://www.python.org
[PYDOC]	Python Standard Dokumentation http://www.python.org/doc
[SPHINX]	Python Documentation Generator http://sphinx.pocoo.org
[MIKTEX]	TeX und LaTeX für Windows http://miktex.org
[TEXLIVE]	TeX Live http://www.tug.org/texlive
[EMPY]	Empy Macro Processor http://www.alcyone.com/software/empy