

Programmentwicklung

Hans Buchmann FHNW/IME

21. Oktober 2014

Ziele

Programmierung auf dem **RaspberryPi**

- ▶ (fast) wie auf dem *Host*
- ▶ *Toolchain* auf dem *Host*
- ▶ nur Programme (*runtime*) auf dem **RaspberryPi**
- ▶ Entwicklung für
 - Java Java SE Runtime Environment **RaspberryPi**
 - C/C++ Posix *runtime*vom **RaspberryPi** aus gesehen

Outline

für Java und C/C++

Host

somewhere_on_your_host

- └ config
- └ src the own source files
- └ work seen by **RaspberryPi**
- └ target-root for the toolchain
- └ tc toolchain

RaspberryPi

somewhere_on_your_RaspberryPi

- └ work mounted on *Host* per sshfs

Teil I

Java

Host

- ▶ Toolchain sollte schon vorhanden sein
- ▶ Beispiel `java/HelloWorld.java`
- ▶ Herstellung `java -d. sourceFile`

RaspberryPi

- ▶ Runtime `jre7-openjdk-headless`

- ▶ Suche kleine Runtime
 - ▶ `jre7-openjdk-headless` ist ziemlich gross
- ▶ Wie steht es mit
 - ▶ Oracle Java Platform, Micro Edition (Java ME)

Teil II

C/C++

Entwicklung

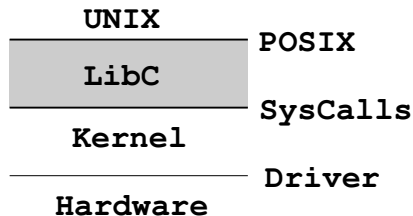
Host

- ▶ Toolchain
`http://sourceforge.net/projects/fhnw-tinl/files/`
- ▶ Beispiele: `src/*`
- ▶ Herstellung: `make -f ../config/Makefile the-app`

RaspberryPi

- ▶ Runtime GNU/Linux POSIX

POSIX → Kernel



POSIX `stdio.h` & Co

SysCalls → **target-root**/usr/include/syscall.h

Bibliotheken

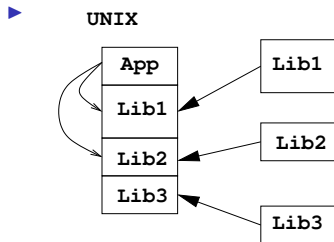
am Beispiel `hello-world-c.c`

- ▶ Der Objectfile `hello-world-c.o`
 - ▶ Der Code `objdump -d hello-world-c.o`,
 - ▶ Die Symbole `readelf -s hello-world-c.o`
- ▶ Das Image `hello-world-c`
 - ▶ Der Code `objdump -d hello-world-c`
 - ▶ Die Symbole `readelf -s hello-world-c.o`
- ▶ `puts`
 - ▶ ist in einer Bibliothek

Statische/Dynamische Bibliothek

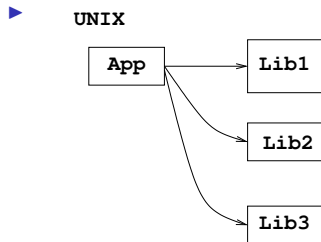
Kopie vs. Referenz

Static



▶ frühes Binden

Dynamic



▶ spätes Binden

—▶ copy
—> reference

Entwicklungsumgebung

- ▶ Entwicklungsumgebung aufsetzen
- ▶ Erste Programme
 - ▶ `hello-world-c.c`, `hello-world-cpp.cc`
- ▶ Benchmark
 - ▶ `primes.cc` auf **RaspberryPi** und *Host*
- ▶ Minimale Programme
 - ▶ `direct-call.S`
 - ▶ `minimal-1.c` und `minimal-1.c` Makefile anpassen

Statische/Dynamische Bibliothek

- ▶ Die Programme
 - ▶ dynamisch linken
 - ▶ statisch linken
- und vergleichen
 - ▶ Grösse
 - ▶ objdump
 - ▶ readelf