



GESELLSCHAFT
FÜR INFORMATIK



OpenPEARL Workshop

M. Schaible

marcel.schaible@fernuni-hagen.de

R. Müller

mueller@hs-furtwangen.de

21.11.2019

Struktur des
OpenPEARL
Systems

Aktueller
Stand

Systemkonfiguration

Übersetzungs-
aufruf

Verfügbare
Systemgeräte

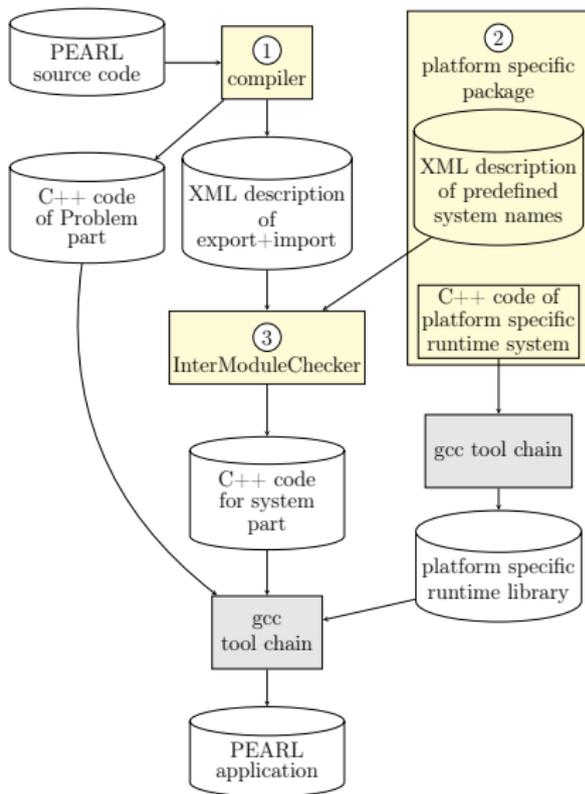
Einstellungen
zur
Ausführung

Workshopumgebung

Weitere
Schnitte

Ausblick

Struktur des OpenPEARL Systems



Struktur des
OpenPEARL
Systems

Aktueller
Stand

Systemkonfiguration

Übersetzungs-
aufruf

Verfügbare
Systemgeräte

Einstellungen
zur
Ausführung

Workshopumgebung

Weitere
Schnitte

Ausblick

Aktueller Stand

Semantische Analyse ist unvollständig. Die kann dazu führen, dass falsche Programmkonstrukte nicht vom Compiler erkannt werden und erst vom C++-Compiler mit mannigfaltigen Fehlermeldungen quittiert werden.

Kein Support für mehrere Module Dies bedeutet, dass derzeit die komplette Anwendung in einem PEARL Quellmodul geschrieben sein muss.

STRUCT, TYPE sind noch nicht verfügbar

Arrayslices sind noch nicht als Prozedurparameter und E/A-Elemente zugelassen

ON ... programmspezifische Fehlerbehandlung
... und einige weitere Probleme. Detaillierte Auskunft gibt das mit ausgehändigte Plattform Manual für Linux.

Struktur des
OpenPEARL
Systems

Aktueller
Stand

Systemkonfiguration

Übersetzungs-
aufruf

Verfügbare
Systemgeräte

Einstellungen
zur
Ausführung

Workshopumgebung

Weitere
Schnitte

Ausblick

Systemkonfiguration

Struktur des
OpenPEARL
Systems

Aktueller
Stand

Systemkonfiguration

Übersetzungs-
aufruf

Verfügbare
Systemgeräte

Einstellungen
zur
Ausführung

Workshopumgebung

Weitere
Schnitte

Ausblick

Das System kann über ein Konfigurationsmenü konfiguriert werden. Dabei können:

- Einstellungen zu Bibliotheken gemacht werden
- Zielplattformen ausgewählt werden
- Gerätetreiber, die von Fremdbibliotheken abhängen ausgewählt werden
- Programmierschnittstellen für Mikrocontroller benannt werden
- uvm.

Diese Konfiguration ist primär für Systementwickler sinnvoll.

Übersetzungsaufruf

Struktur des
OpenPEARL
Systems

Aktueller
Stand

Systemkonfiguration

Übersetzungs-
aufruf

Verfügbare
Systemgeräte

Einstellungen
zur
Ausführung

Workshopumgebung

Weitere
Schnitte

Ausblick

Es gibt nur ein Kommandozeileninterface:

```
prl <options> <file>
```

wichtige Optionen sind:

- b linux Auswahl der Zielplattform linux (default)

- c nur von PEARL nach C++ übersetzen

- h zeigt die Aufrufmöglichkeiten

<file> steht für den Quellmodul mit Extension .prl

Verfügbare Systemgeräte

Als Auszug aus dem Plattform Manual:

StdIn, StdOut die üblichen E/A-Kanäle unter Linux

Console eine komfortable E/A-Steuerung mit Taskadressierung und Systemdiagnose

Pipe Zugriff auf *named pipes* für Tests oder Anbindung an andere Softwarepakete

Disk Dateien

Log Steuerung der Systemausgaben. Default: Fehler- und Warnmeldungen auf die Datei `pearl_log.txt`

I2C Bus Schnittstelle zu diversen I2C-Teilnehmern (BME280, PCF8574, LM75, ...)

UnixSignal liefert PEARL Interrupts an die Anwendung

Struktur des
OpenPEARL
Systems

Aktueller
Stand

Systemkonfiguration

Übersetzungs-
aufruf

Verfügbare
Systemgeräte

Einstellungen
zur
Ausführung

Workshopumgebung

Weitere
Schnitte

Ausblick

Einstellungen zur Ausführung

Struktur des
OpenPEARL
Systems

Aktueller
Stand

Systemkonfiguration

Übersetzungs-
aufruf

Verfügbare
Systemgeräte

Einstellungen
zur
Ausführung

Workshopumgebung

Weitere
Schnitte

Ausblick

Über die Datei `.pearlrc` kann unter Linux die Ausführungweise gesteuert werden:

LogLevel Detailgrad der Systemmeldungen

Cores sollte auf 1 stehen bleiben

CPUtime OpenPEARL Anwendungen werden auch bei normalen Benutzerrechten im FIFO-Scheduler ausgeführt. Dies kann zur Folge haben, dass eine OpenPEARL-Anwendung nicht mehr abgebrochen werden kann, da die Eingabeaufforderung keine CPU-Zeit zur Verfügung gestellt bekommt.

Workshop

Ziel

Struktur des
OpenPEARL
Systems

Aktueller
Stand

Systemkonfiguration

Übersetzungs-
aufruf

Verfügbare
Systemgeräte

Einstellungen
zur
Ausführung

Workshopumgebung

Ziel

Programmierungsumgebung

Basiskonfiguration

Arbeit mit RDP

Erste Übersetzungen

Anschluss des Sensor
BME280

Programmierung

Weitere
Schnitte

Ausblick

- OpenPEARL spielerisch kennen lernen
- anhand eine einfachen Anwendung
- potentielle Nutzer gewinnen (... um Fehler zu finden)

Workshop

Programmierungsumgebung

Struktur des
OpenPEARL
Systems

Aktueller
Stand

Systemkonfiguration

Übersetzungs-
aufruf

Verfügbare
Systemgeräte

Einstellungen
zur
Ausführung

Workshopumgebung

Ziel

Programmierungsumgebung

Basiskonfiguration

Arbeit mit RDP

Erste Übersetzungen

Anschluss des Sensor
BME280

Programmierung

Weitere
Schnitte

Ausblick

- eigenes WLAN OpenPEARL (Schlüssel: OpenPEARL2019)
- für Raspberry Pi und Laptops
- Raspberry Pis erhalten IP-Adressen nach Schema 192.168.201.x, x steht in der Packung
- Zugriff auf Raspberry Pi ber
 - ssh
 - RDP

Erste Schritte

Struktur des
OpenPEARL
Systems

Aktueller
Stand

Systemkonfiguration

Übersetzungs-
aufruf

Verfügbare
Systemgeräte

Einstellungen
zur
Ausführung

Workshopumgebung

Ziel

Programmierungsumgebung

Basiskonfiguration

Arbeit mit RDP

Erste Übersetzungen

Anschluss des Sensor
BME280

Programmierung

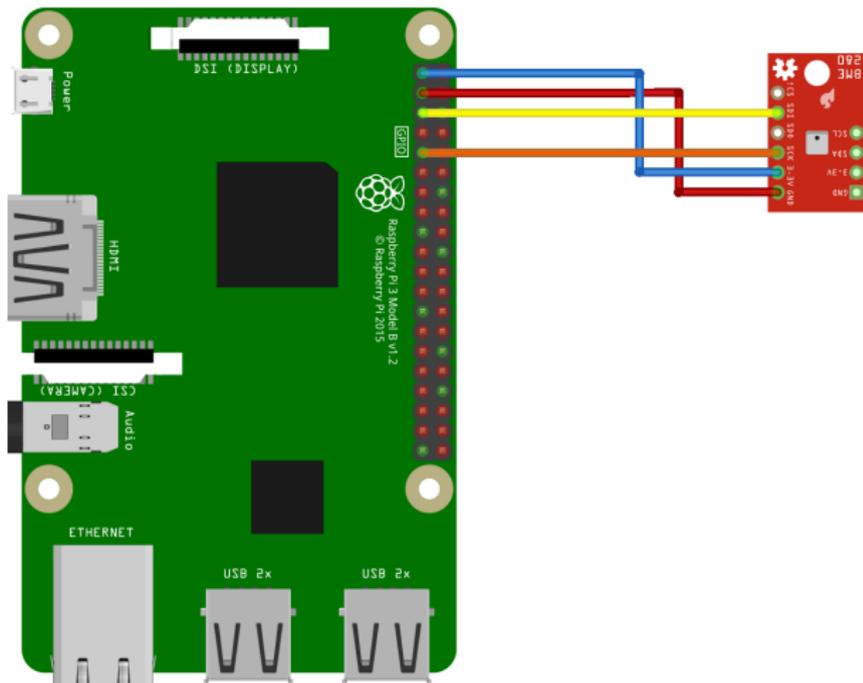
Weitere
Schnitte

Ausblick

- wechseln Sie in das Verzeichnis demos
- übersetzen Sie HelloWorld.pr1
- starten Sie HelloWorld
- übersetzen und starten Sie schedDemo und consoleDemo

Workshop

Anschluss des BME280



fritzing

```
i2c-detect -y 1 sollte die Adresse 76 erkennen
```

Struktur des OpenPEARL Systems

Aktueller Stand

Systemkonfiguration

Übersetzungsaufwurf

Verfügbare Systemgeräte

Einstellungen zur Ausführung

Workshopumgebung

Ziel

Programmierungsumgebung

Basiskonfiguration

Arbeit mit RDP

Erste Übersetzungen

Anschluss des Sensor BME280

Programmierung

Weitere Schnitte

Ausblick

Workshop I

Programmierung

Struktur des
OpenPEARL
Systems

Aktueller
Stand

Systemkonfiguration

Übersetzungs-
aufruf

Verfügbare
Systemgeräte

Einstellungen
zur
Ausführung

Workshopumgebung

Ziel

Programmierungsumgebung

Basiskonfiguration

Arbeit mit RDP

Erste Übersetzungen

Anschluss des Sensor
BME280

Programmierung

Weitere
Schnitte

Ausblick

```
SYSTEM;
```

```
! Geraeteparameter sind im 'Platform Manual' besch  
bme280: BME280('76'B4,2,4,2,4,1) --- I2CBus('/dev/
```

```
PROBLEM;
```

```
SPC bme280 DATION IN SYSTEM BASIC;
```

```
DCL myBme280 DATION IN BASIC
```

```
ALL ! da noch kein STRUCT verfuegbar  
CREATED(bme280);
```

Workshop II

Programmierung

Struktur des
OpenPEARL
Systems

Aktueller
Stand

Systemkonfiguration

Übersetzungs-
aufruf

Verfügbare
Systemgeräte

Einstellungen
zur
Ausführung

Workshopumgebung

Ziel

Programmierungsumgebung

Basiskonfiguration

Arbeit mit RDP

Erste Übersetzungen

Anschluss des Sensor
BME280

Programmierung

Weitere
Schnitte

Ausblick

```
.....  
OPEN myBme280;  
DCL values(3) FIXED(31);  
...  
TAKE values FROM myBme280;  
PUT 't:', values(1),  
    'p: ', values(2),  
    'Feuchte:', values(3) TO  
    so BY A, F(6), A, F(6), A, F(6), SKIP;
```

Der Treiber liefert derzeit 3 FIXED Werte, die noch skaliert werden müssen.

Workshop

weitere Schritte (heute)

Struktur des
OpenPEARL
Systems

Aktueller
Stand

Systemkonfiguration

Übersetzungs-
aufruf

Verfügbare
Systemgeräte

Einstellungen
zur
Ausführung

Workshopumgebung

Weitere
Schnitte

Ausblick

- OpenPEARL am Raspberry Pi ausprobieren
- BME280 anschließen und auslesen
- ... Programm nach eigenem Wünschen weiterentwickeln ...
- bei Interesse weitere Punkte auf Nachfrage heute Abend
statt *Heiligem Grab* oder **im Heiligen Grab**

Ausblick

- Fehlermeldungen bitte direkt als Ticket bei SourceForge einstellen. Dies geht inzwischen auch ohne Sourceforge Account
- Treiberentwicklungen gehen von 1 Tag bis 1 Woche - je nach Komplexität des Gerätes und System- und Programmierkenntnissen. Der PCA9685-Treiber mit 16 Kanälen LED-Treiber hat ca. 6 Stunden benötigt.
- Planung für
 - STRUCT und komplette I/O Unterstützung [Ende 2019]
 - TYPE, mehrere Module [April 2020]
 - Array und Slices als Parameter für PROC und E/A [offen]
 - komplette semantische Analyse [offen]
 - ESP32 Portierung ist in Bearbeitung [Mai 2020] — ideal für kleine IoT Geräte mit WLAN
- Workshopssysteme können ausgeliehen werden. Bitte Formular ausfüllen!

Struktur des
OpenPEARL
Systems

Aktueller
Stand

Systemkonfiguration

Übersetzungsauf
ruf

Verfügbare
Systemgeräte

Einstellungen
zur
Ausführung

Workshopumgebung

Weitere
Schnitte

Ausblick