



### Compiler für Eingebettete Systeme – Übungsblatt Nr. 3

Es sollen einfache ILP-basierte Scratchpad-Allokationen für Programm-Code realisiert werden, die auf Energie- bzw. WCET-Daten beruhen.

#### 1. Aufgabe: (SPM-Allokation von Funktionen zur Energie-Minimierung) (15 Punkte)

- (a) Nehmen Sie eine Systemarchitektur bestehend aus einem Prozessor, einem Scratchpad-Speicher und einem Hauptspeicher an. Es sollen ganze Funktionen eines fiktiven Programms auf den Scratchpad allokiert werden, um den Energieverbrauch zu minimieren. Die grundlegenden Eigenschaften des Programms sind in nachstehender Tabelle zu finden: (10 Punkte)

Nr.	Funktion	Größe [16-Bit Worte]	# ausgeführte Instruktionen
1	main	56	54
2	ldctrow	124	820
3	ldctcol	124	820
4	ldct	48	24
5	init_idct	132	792
6	print_result	348	186

Formulieren Sie in einem Text-Editor ein ILP-Modell zum Minimieren des Energieverbrauchs. Benutzen Sie das Werkzeug `lp_solve` um Ihr ILP zu lösen und um zu bestimmen, welche Funktionen auf den Scratchpad zu verschieben sind.

Jeder Zugriff auf ein 16-Bit Wort im Hauptspeicher benötigt 50 nJ. Ein Zugriff auf einen 1024 16-Bit Worte großen Scratchpad benötigt 1 nJ.

- (b) Wenden Sie Ihr ILP für die Scratchpad-Größen 64, 128, 256, 512, 768 und 1024 Worte an. Wir nehmen an, dass die Energie, die zum Zugriff auf kleinere Scratchpads notwendig ist, linear mit dem Energieverbrauch des 1024 Worte großen Scratchpads skaliert, bspw. für einen 64 Worte großen SPM 0,064 nJ pro Zugriff. Erzeugen Sie mit OpenOffice ein Diagramm, auf dessen X-Achse obige Scratchpad-Größen abgetragen sind, und das in einer Kurve die durch Ihre Optimierung erzielten Energieverbräuche zeigt. (5 Punkte)

#### 2. Aufgabe: (SPM-Allokation von Funktionen zur WCET-Minimierung) (15 Punkte)

- (a) Realisieren Sie eine einfache Scratchpad-Allokation analog zu Kapitel 7, Folie 62 der Vorlesung, die komplette Funktionen (keine Daten) auf den Scratchpad verschiebt. Der Gewinn einer einzelnen Funktion bei Verschiebung auf den Scratchpad ergibt sich aus der Differenz der WCETs der Funktion im Hauptspeicher bzw. Scratchpad, multipliziert mit der Anzahl der Ausführungen der Funktion im worst case. (10 Punkte)

Ergänzen Sie hierzu die Methoden `computeWCETSPmMem()`, `computeWCETMainMem()`, `generateILPVariables()`, `generateILPConstraints()`, `generateILPObjective()`, `solveILP()` und `moveCode()` aus Datei `spm_function.cc`.

- (b) Wenden Sie Ihre Optimierung auf den Benchmark `adpcm_g721_board_test.c` für die Scratchpad-Größen 0, 616, 1232, 1848, 2464, 3080, 3696, 4312, 4928, 5544 und 6160 Bytes an. Erzeugen Sie mit OpenOffice ein Diagramm, auf dessen X-Achse obige Scratchpad-Größen abgetragen sind, und das in einer Kurve die durch Ihre Optimierung erzielten WCETs zeigt. (5 Punkte)

### Technische Hinweise:

- Loggen Sie sich mit den bereitgestellten Zugangsdaten an Ihrem Rechner ein. Loggen Sie sich von dort aus per SSH auf der folgenden Maschine ein:

```
wccsrv02.informatik.uni-ulm.de
```

- Für Aufgabe 1 wechseln Sie ins Verzeichnis ILP; editieren Sie das darin enthaltene ILP-Grundgerüst. Zum Lösen Ihres ILP rufen Sie `lp_solve energy.lp` auf.
- Für Aufgabe 2 wechseln Sie ins Verzeichnis WCC; rufen Sie zum Bearbeiten der C++-Dateien der SPM-Allokation den Editor `kate` auf.
- Um Ihre bearbeiteten C++-Dateien zu übersetzen, rufen Sie im Verzeichnis WCC das Programm `make` auf.
- Zum Überprüfen, wie sich Ihr Compiler verhält, ist im Verzeichnis WCC das Programm `adpcm_g721_board_test.c` hinterlegt. Übersetzen Sie dieses für Aufgabe 2 mit `wcc -O0 -S adpcm_g721_board_test.c -fpspm-fct --param pspm-size=SPM-Größe`.
- Zum Bearbeiten dieses Übungszettels benötigen Sie evtl. Hintergrundinformation zur verwendeten Low-Level IR ICD-LLIR und zur ILP-Bibliothek. Diese finden Sie (nur über Uni-Rechner erreichbar) unter

```
https://wcc-web.informatik.uni-ulm.de/lehre/CfES/ICD-LLIR
```

bzw.

```
https://wcc-web.informatik.uni-ulm.de/lehre/CfES/libilp
```

Öffnen Sie im links erscheinenden Menü den Punkt `Data Structures`, und Sie erhalten detaillierte Informationen zu sämtlichen Klassen und Methoden.