

Vorlesungsprüfung aus Rechnerstrukturen

17. Oktober 2014

Die Arbeitszeit beträgt 90 Minuten. Als Hilfsmittel sind ausnahmslos Schreibzeug, Lineal und (nicht programmierbarer) Taschenrechner erlaubt. Schreiben Sie Ihre Antworten und Lösungen (inkl. Lösungsweg!) mit Füllfeder oder Kugelschreiber (nicht rot, KEIN Bleistift!) und streichen Sie alles durch, was nicht zur Beurteilung herangezogen werden soll. Ein Abbruch der Prüfung nach Erhalt der Angaben führt in jedem Fall zu einer Beurteilung.

Tragen Sie Namen, Kennzahl und Matrikelnummer **zu Beginn** der Prüfung in die Tabelle ein und beschriften Sie jedes Blatt, das Sie abgeben möchten rechts oben mit Namen und Matrikelnummer.

Familienname:	Vorname:
Kennzahl:	Matrikelnummer:

Viel Erfolg!

Die nachfolgende Tabelle nicht beschriften!

Beispiel		Mögliche Punkte	Erhaltene Punkte
Theoriefragen		48	
Praxisbeispiele	1	16	
	2	16	
	3	16	
Gesamt		96	

Theoriefragen (je 6 Punkte)

Frage 1:

Was versteht man unter MTTF, MTBF, MTTR und Verfügbarkeit (Availability)? Beschreiben Sie jedes dieser Maße kurz. Wie hängen diese zusammen?

Frage 2:

Skizzieren Sie den schematischen Aufbau einer MIPS-CPU mit Pipeline.

Frage 3:

Treten beim Ausführen der folgenden MIPS Befehle Hazards auf (Nehmen Sie an dass die Standard Pipeline verwendet wird)? Begründen Sie Ihre Antwort!

```
add $t0, $s0, $s1
```

```
sub $s0, $s0, $s1
```

Frage 4:

Was versteht man bei einem Prozessor unter Pipelining? Wieso wird es verwendet? Geben Sie die limitierenden Faktoren beim Pipelining an.

Frage 5:

Welche Arten der Cache-Associativity kennen Sie? Erklären Sie jede Variante kurz. Welche Variante hat den geringsten Hardwareaufwand, welche den höchsten?

Frage 6:

Was versteht man unter einem Translation Look-aside Buffer? Wozu wird dieser verwendet und was wird in den Buffer eingetragen?

Frage 7:

Was versteht man unter Loop-Unrolling? Warum wird es verwendet?

Frage 8:

Wozu verwendet man Atomic-Swap? Erklären Sie kurz die Funktionsweise.

Praxisbeispiele

Beispiel 1

Zeichnen Sie das Pipeline Diagramm für das folgende Programm:

```
ori $s4, $zero, 8
loop: lw $t2, 0($s4)
      mult $s1, $s1, $t2
      addi $s4, $s4, -4
      bne $s4, $zero, loop
      nop
```

Es handelt sich um die Standard-MIPS Pipeline, es ist aber kein Forwarding implementiert und eine Multiplikation benötigt vier Zyklen in der Ausführungsstufe (Execution Stage, EX). Sprungziele werden in der Dekodier-Stufe (ID) berechnet. Sollte es zu Daten-Hazards kommen, so wird die Pipeline angehalten (Stall) bis die Daten zur Verfügung stehen. Nehmen Sie an, dass die Schleife zweimal durchlaufen wird. Es ist kein Branch-Delay Slot vorhanden.

Berechnen Sie weiters die Anzahl der Stall-Zyklen aufgrund von Daten-Hazards und die Anzahl der Control-Hazards.

Beispiel 2

Übersetzen Sie folgendes C Codefragment in ein MIPS Assembler-Programm (Annahme: es gibt keinen Branch-Delay Slot):

...

```
i = 0;
```

```
while (i < 10)
```

```
    i ++;
```

...

Beispiel 3

30% der Instruktionen eines bestimmten Programmes sind Gleitkommamultiplikationen und 8% sind Gleitkommadivisionen. Bei einem Redesign des zur Ausführung des Programmes verwendeten Computers sind drei Optionen verfügbar:

- Beschleunigung der Division um den Faktor 9
- Beschleunigung der Multiplikation um den Faktor 3,2
- Beschleunigung der Multiplikation um den Faktor 6, Verlangsamung der Division um den Faktor 1,5

Welche der Optionen ist die Beste?

