

11. Übungsblatt Informatik II

(Abgabe: 06.07.2005)

1. Aufgabe: ()

Codeerzeugung von $p = \text{new } t$

Es handelt sich hierbei im Grunde um eine Zuweisung.

Wobei durch das $\text{new } t$ eine neue Variable auf den Heap gelegt wird.
Laut Definition zeigt $\text{gpr}(29)$ auf erste freie Adresse im Heap.

// Testen, ob sich Stack und Heap überschneiden

```
0:      subi RD=1 RS=29 imm =4*size(t)
4:      addi RD=2 RS=30 imm=fsize(f)
8:      sgr  RD=3 RS=2 RS=1
12:     beqz RS=3 imm=8
16:     trap (imm = x, eingeklammert, weil das x i.d. Vorlesung noch
        nicht def. wurde)
```

// Jetzt die neuen Werte setzen. Wichtig ist, dass p die neue Heapaddr + 1
// zugewiesen wird und nicht einfach der alte heap-pointer!

```
20:     addi RD=29 RS=1 imm=0
24:     addi RD=1 RS=1 imm=1
28:     sw  RD=1 RS=k imm=0    //An.: grp(k) = Adresse von p
```

2. Aufgabe: ()

Die *write-enable* Signale des *Special Purpose Register Files*:

SR	rfe (Return From Exception)
ESR	JISR (Jump Interrupt Service Routine)
ECA	JISR
EPC	JISR
EDATA	trap \vee pfls \vee pff
PTO	kann nur mit movi2s verändert werden
PTL	kann ebenfalls nur mit movi2s verändert werden
MODE	JISR \vee rfe

3. Aufgabe: ()

11. Übungsblatt Informatik II

(Abgabe: 06.07.2005)

DLX-Schaltkreis(ohne Interrupts)	Änderung
Instruction Decoder	Da es nun neue Instruktionen (movi2s, movs2i, trap, rfe) gibt, muss der Instr.Dec. in der Lage sein, diese aus dem Instruktionscode zu dekodieren. ⇒ Erzeugen der entsprechenden Prädikate
NextPC	Wenn JISR = true ⇒ PC' = Start Interrupt Service Routine Wenn rfe = true ⇒ PC' = EPC (Backup PC)
Memory	Instruktion- und Datamemoryzugriffe können nun Interrupts auslösen. Eine Memory Control Unit übernimmt Missalignment-Berechnung.
GPR	Da es nun zusätzlich ein SPR gibt, müssen SPR und GPR verbunden werden. für movi2s : Din von SPR ist GPR.A sonst: Standard Inputs für movs2i : Din von GPR ist SPR.SA sonst: GPRglue.output

4. Aufgabe: ()