

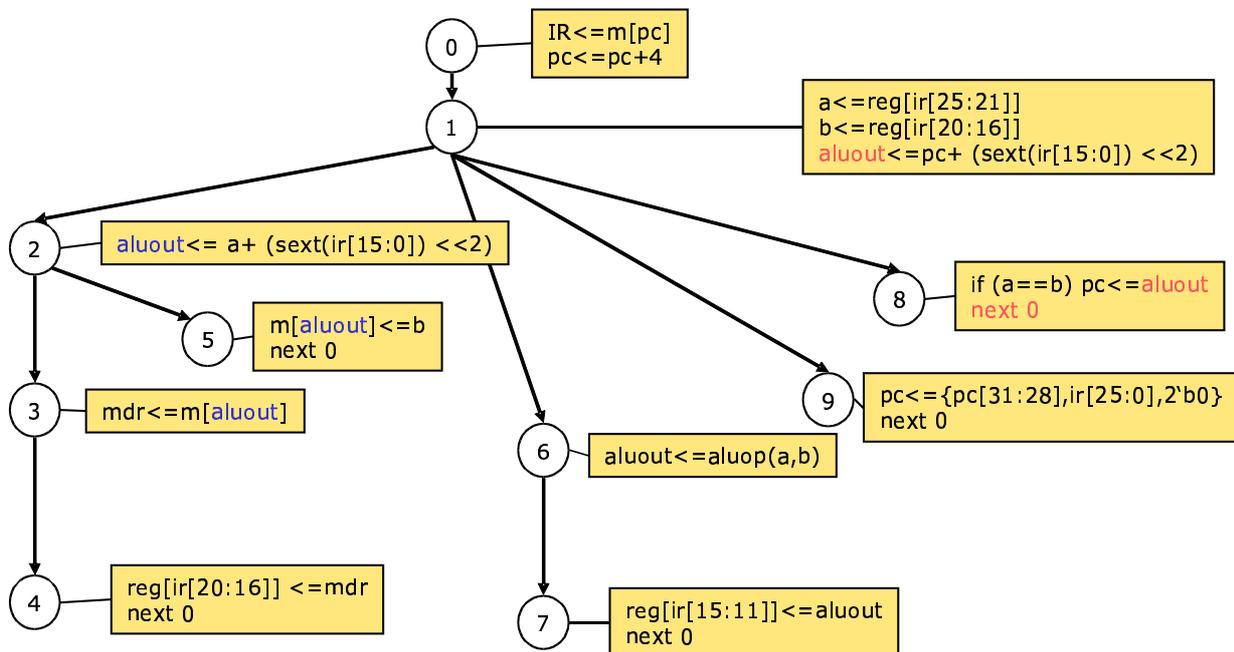
# Technische Grundlagen der Informatik II

## 8. Übung – MIPS Operationswerk

### Sommersemester 2006

#### Aufgabe 1: MIPS-Mikrooperationen

Gegeben ist das folgende Zustandsdiagramm der Mehrtaktimplementierung zur Interpretation der MIPS-Befehle.



a) Ordnen Sie den Verzweigungen jeweils den Befehl oder die Befehlsgruppe zu, in die an den Zuständen 1 und 2 im Zustandsdiagramm verzweigt wird.

**b)** Erstellen Sie aus dem Zustandsdiagramm für den aktuellen Befehl an der Speicherstelle `m[128]` mit dem Befehlsformat `0x01093820` (`add $7, $8, $9`) und den Registerwerten `reg[7]=1`, `reg[8]=2`, `reg[9]=3` eine Tabelle, die für jeden Taktschritt (beginnend bei Zustand 0) den jeweiligen Zustand, die auszuführenden Mikrooperationen und die neuen Werte der Register bzw. Speicherzellen enthält.

## Aufgabe 2: MIPS-Registerspeicher

Implementieren Sie den Registerspeicher (Vorlesung Kapitel 7, Folien 11-13) in Verilog. Nehmen Sie dazu Folie 12 als Vorlage für die Ausgangsbeschaltung, Folie 13 für die Eingangsbeschaltung. Fügen sie ein asynchrones RESET-Signal hinzu.

## Aufgabe 3: Fakultät

Entwickeln Sie ein MIPS-Assemblerprogramm zur Berechnung der Fakultät der Zahl 7. Ihr Programm soll den Startwert „7“ in das Argumentregister `a0` laden, wo er nach dem Programmablauf immer noch stehen soll. Das Ergebnis soll in Register `v0` abgelegt werden.

**a)** Beschreiben Sie den Algorithmus in Pseudo-Code. Verwenden Sie dabei nur Operationen, die im MIPS-Befehlssatz darstellbar sind.

**b)** Entwickeln Sie nun aus a) ein MIPS-Assemblerprogramm. Benutzen Sie dazu die in den Vorlesungsfolien vorgestellten MIPS-Assemblerbefehle. Verwenden Sie den `addi`-Befehl und das `0`- bzw. `zero`-Register, um Konstanten in Register zu laden.