
Übung 10

Abgabe bis **Donnerstag, 23. Juni 10:00** via EPIIC: <http://ep.iic.jku.at>.

1. Multicomputer (6 Punkte)

Untersuche die Topologie eines Multicomputers mit $n - 1$ Elementen ($n = 2^k$), wobei zwei Prozessoren P_i und P_j , $1 \leq i < j < n$ verbunden sind wenn beide folgenden Punkte erfüllt sind:

- $i < n/2$
- $j = 2i$ oder $j = 2i + 1$.

Bestimme den Durchmesser, den Grad und die Anzahl der physikalischen Verbindungen. Begründe deine Ergebnisse.

2. Konnektivität (7 Punkte)

In der Vorlesung wurden der Durchmesser, der Grad und die Anzahl der Verbindungen als Charakteristika eines Netzwerks von Rechnern vorgestellt. Zwei weitere Maße sind die Knotenkonnektivität und die Kantenkonnektivität. Diese sind wie folgt definiert:

Definition: *Knotenkonnektivität* $nc(G)$

Für eine beliebige Teilmenge $M \subseteq V$ bezeichnet $G_{V \setminus M}$ den Restgraphen, der durch löschen der Knoten in M , sowie der zugehörigen Kanten entsteht, d.h.

$$G_{V \setminus M} = (V \setminus M, E \cap ((V \setminus M) \times (V \setminus M)))$$

Dann ist die *Knotenkonnektivität* des Graphen G :

$$nc(G) := \min_{M \subseteq V} \{ |M| : \text{Es gibt } i, j \in V \setminus M, \text{ sodass es in } G_{V \setminus M} \text{ keinen Pfad von } i \text{ nach } j \text{ gibt} \}$$

Definition: *Kantenkonnektivität* $ec(G)$

Für eine beliebige Teilmenge $F \subseteq E$ bezeichnet $G_{E \setminus F}$ den Restgraphen, der durch löschen der Kanten in F entsteht, d.h.

$$G_{E \setminus F} = (V, E \setminus F)$$

Dann ist die *Kantenkonnektivität* des Graphen G :

$$ec(G) := \min_{F \subseteq E} \{ |F| : \text{Es gibt } i, j \in V, \text{ sodass es in } G_{E \setminus F} \text{ keinen Pfad von } i \text{ nach } j \text{ gibt} \}$$

- Bestimme jeweils die Knoten- und Kantenkonnektivität eines Sterns mit k Knoten, eines k -dimensionalen Hypercubes und eines Token-Rings mit k Knoten.
- Gib eine inhaltliche Interpretation der beiden Maße in Bezug auf die Ausfallsicherheit eines Netzwerkes an.

3. Leistungsbewertung (11 Punkte)

Analysiere drei verschiedene Versionen von MIPS Prozessoren:

- P_1 (*Single Cycle Prozessor*): Der Prozessor hat eine Taktfrequenz von 200 MHz. Jeder Befehl benötigt einen Taktzyklus.
- P_2 (*Multicycle Prozessor*): Die Anzahl der Taktzyklen (jeweils 1, 5 ns) pro Befehl sind in folgender Tabelle gegeben.

Instruktion	Taktzyklen
<code>beq, j</code>	3
ALU-Instruktion, <code>sw</code>	4
<code>lw</code>	5

- P_3 (*Pipelined Prozessor*): Der Prozessor hat einen fünfstufigen Pipeline und eine Taktfrequenz von 400 MHz. Beim Befehl `j` muss ein Befehl aus der Pipeline gelöscht werden. Beim Befehl `beq` müssen zwei Befehle aus der Pipeline gelöscht werden falls der Sprung ausgeführt wird.
- (a) Berechne CPI und Ausführungszeit für jeden der drei Prozessoren für ein Programm mit 100 000 Instruktionen und dem in der Tabelle gegebenen Instruktionmix.

Befehl	relative Häufigkeit
ALU-Instruktion	60%
<code>lw</code>	17%
<code>beq</code>	12% (bei 40% davon wird der Sprung getätigt)
<code>sw</code>	6%
<code>j</code>	5%

- (b) Wie schnell müsste Prozessor P_2 getaktet sein, damit er das Programm aus Aufgabe (a) schneller ausführt als P_1 .
- (c) Ein Compiler übersetzt ein Teilprogramm in eine Codesequenz mit 1 `lw` und 10 ALU-Instruktionen. Durch eine Optimierung kann der Compiler auch eine Codesequenz mit 7 `lw`, 1 `beq` und 2 ALU-Instruktionen erstellen. Bei welchen Prozessoren ist es ratsam die Optimierung in den Compiler einzubauen, bei welchen nicht? Begründe deine Antworten.