University of the Saarland
Department 6.2 - Informatik
Prof. Dr. W.J. Paul

Computer Architecture I - WS 02/03

(bis: Montag 20.01.2003)

Aufgabe 1: (Schneller Forwarding-Schaltkreis) **(7 + 8 + 5 Punkte)**

In der Vorlesung haben wir einen Forwarding-Schaltkreis entworfen, der Daten aus drei Stufen forwarden kann. Die Konstruktion kann einfach auf einen s-stufigen Forwarding-Schaltkreis erweitert werden, mit $s > 3$. Die eigentliche Datenauswahl geschieht dann durch s viele kaskadierte Multiplexer. Deshalb ist das Delay dieses Schaltkreises in $O(n)$.

Die s vielen Multiplexer können aber auch als balancierter binärer Baum der Tiefe $\lceil \log(s) \rceil$ angeordnet werden. Das Signal $top.j$ signalisiert dabei, dass die Stufe j die benötigten Daten enthält. Die Signale $top.j$ können benutzt werden, um den Multiplexer Baum zu steuern.

- Entwerfen Sie einen Schaltkreis TOP, der die Signale $top.j$ generiert. Benutzen Sie dazu einen Parallel Präfix Schaltkreis.
- Entwerfen Sie einen s-Stufen Forwarding-Schaltkreis basierend auf dem Multiplexer Baum und dem Schaltkreis TOP. Zeigen Sie, dass die Realisierung ein Delay von $O(\log(s))$ hat.
- Wie kann das Delay des Forwarding-Schaltkreises noch weiter verbessert werden?

Aufgabe 2: (Deadlock) **(10 Punkte)**

Im Fall eines Data Hazard stoppt die Stall-Engine die Stufen `IF` and `ID`. Der Forwarding-Schaltkreis `Forw` signalisiert einen Treffer in Stufe $j \in \{2, 3, 4\}$ durch

$$hit[j] \quad = \quad (full.j \wedge GPRw.j) \wedge (ad \neq 0) \wedge (ad = Cad.j)$$

Diese hit Signale werden benutzt um das Data Hazard Signal $dhaz$ zu generieren. Der Test, ob die Stufe j überhaupt voll ist ($full.j = 1$) ist notwendig für die korrekte Funktion.

Zeigen Sie, dass in einer Stall-Engine ohne diesen Test

$$hit[j] \quad = \quad GPRw.j \wedge (ad \neq 0) \wedge (ad = Cad.j)$$

auch Dummy-Instruktionen das Hazard-Signal aktivieren können, und dass dadurch der Prozessor in einen Dead-Lock geraten kann.