ALBERT-LUDWIGS-
UNIVERSITÄT FREIBURG

INSTITUT FÜR INFORMATIK

Prof. Dr. Bernd Becker
Dipl. Inf. Thomas Eschbach

Freiburg, 2. Juli 2002

Übungsblatt 09 zur Vorlesung

Technische Informatik II

Aufgabe 1

Punkte (3, 4)

In Abbildung 1 ist das idealisierte Timing-Diagramm für einen Rechner mit verkürzter *Fetch*- und *Execute*-Phase angegeben.

a) Geben Sie eine Schaltung an, mit der die Signale e, $s1$ und $s0$ generiert werden können.

b) Geben Sie PAL-Gleichungen für folgende Signale der Kontrollogik an:

- Clocksignal Ick zum Speichern neuer Befehle im Instruktionsregister I.
- Clocksignal $ACCck$ zum Speichern neuer Daten im Akkumulator ACC.

Hinweis: Überprüfen Sie analog zum Vorgehen in der Vorlesung, bei welchen Befehlen des RE-TI-II Rechners die obigen Signale jeweils aktiv werden.

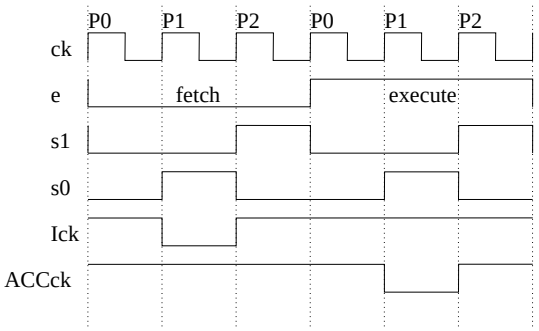

Abbildung 1: Idealisiertes Timing-Diagramm

Aufgabe 2

Punkte (1, 2.5, 2.5)

Betrachten Sie einen neuen Rechner mit den Registern PC (Program Counter), ACC (Akkumulator), B (Operandenregister) und IN (Indexregister), wie in Abbildung 2 skizziert. Die Instruktionen des Rechners sind in Tabelle 1 zusammengestellt. Dabei werden die Register wieder durch 2 Bit kodiert, die in den Befehlen mit D und S bezeichnet werden. Die Operationscodes op und die Bedingungen c sind entsprechend des RE-TI-II Rechners gewählt.

a) Zeigen Sie, dass die Instruktionen des neuen Rechners mit den Datenpfaden aus Abbildung 2 durchgeführt werden können.

b) Können alle $LOAD$, $STORE$ und $COMPUTE$-Befehle des RE-TI-II Rechners auf dem neuen Rechner simuliert werden? Geben Sie entsprechende Befehlsfolgen und die notwendigen Einschränkungen für den neuen Rechner an.

c) Geben Sie einen zusätzlichen Befehl für den neuen Rechner an, der nur die bestehenden Datenpfade verwendet und mit dessen Hilfe die $JUMP$-Befehle des RE-TI-II Rechners simuliert werden können.

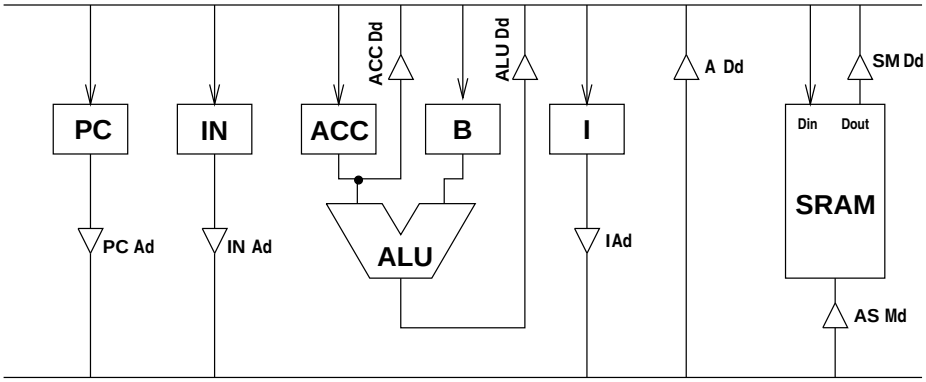

Abbildung 2: Datenpfade des neuen Rechners

I[31:30]	I[29:28]	I[27]	I[26]	I[25:0]	Befehl	Wirkung	
LOAD-Befehle:		$D \in \{PC, IN, B, ACC\}$		$i \in \mathbb{B}^{26}$			
01	00	D		i	$LOAD\ D\ i$	$D := M(i)$	$PC := PC + 1$
01	01	D		$*$	$LOADIN\ D$	$D := M(IN)$	$PC := PC + 1$
01	10	D		i	$LOADI\ D\ i$	$D := i$	$PC := PC + 1$
STORE-Befehle:		$S \in \{PC, IN\}$		$i \in \mathbb{B}^{26}$			
10	00	$*$	$*$	i	$STORE\ i$	$M(i) := ACC$	$PC := PC + 1$
10	10	$*$	$*$	$*$	$STOREIN$	$M(IN) := ACC$	$PC := PC + 1$
10	01	S		$*$	$MOVE\ S$	$ACC := S$	$PC := PC + 1$
COMPUTE-Befehle:		$op \in \{SUB, ADD, OPLUS, OR, AND\}$					
00	op		$*$	$*$	op	$ACC := ACC\ op\ B$	$PC := PC + 1$

Tabelle 1: Instruktionen des neuen Rechners

Aufgabe 3

Punkte (2, 2, 3)

Betrachten Sie den *Von Neumann* Addierer aus Abbildung 3. Für $t \in \mathbb{N}$ wird mit $X(t)$ und $Y(t)$ der Inhalt der Register X und Y nach dem t-ten Takt bezeichnet. Der ursprüngliche Inhalt der Register X und Y beschreibt die zu addierenden Zahlen in Binärdarstellung, es soll folglich $\langle X(0)\rangle + \langle Y(0)\rangle$ berechnet werden. Die *Most Significant Bits* (MSBs) der beiden Operanden sind 0, d.h. $X_n(0) = Y_n(0) = 0$.

a) Unter welchen Voraussetzungen dauert die Addition volle $n+1$ Takte?

b) Zeigen Sie: $\exists\, t \leq n+1 : X(t) = (0, 0, \ldots, 0)$.

c) Modifizieren Sie die Schaltung aus Abbildung 3 so, dass die Register X und Y unter Kontrolle eines Signals $/load$ von zwei n-Bit Bussen A und B geladen werden können.

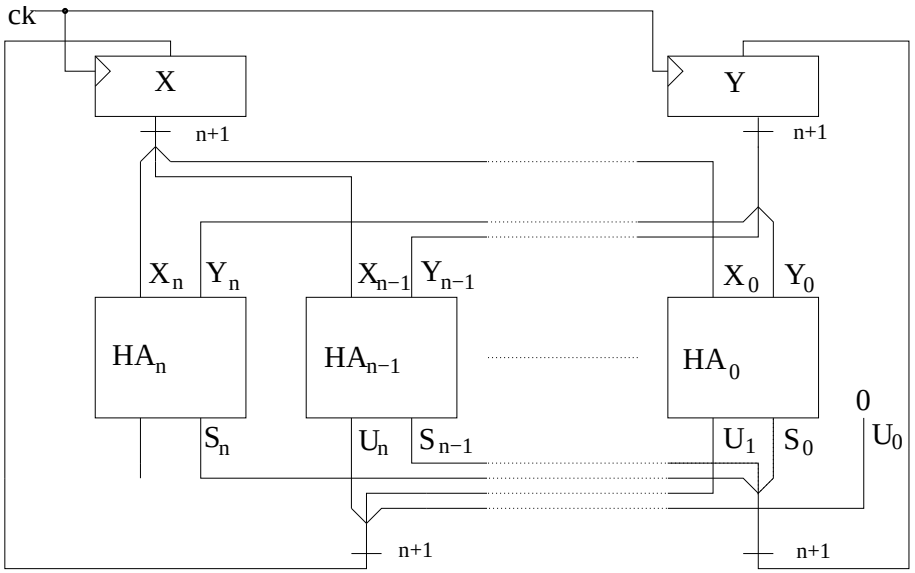

Abbildung 3: *Von Neumann* Addierer

Abgabe : Montag, den 8. Juli 2002 bis 17.00 Uhr