Prof. Dr. Christoph Scholl
Dr. Paolo Marin

Freiburg, 29. Januar 2016

Technische Informatik
Übungsblatt 11

Aufgabe 1 $(2 + (0 + 2\text{ (Bonus)}) + 2 + (0 + 2\text{ (Bonus)})$ Punkte)

Für die ReTI benötigen wir eine Reihe von Kontrollsignalen, die die Register, Treiber, ALU und SRAM ansprechen (Siehe Kap. 4.5, Folie 65).

Erstellen Sie beispielhaft für die folgenden Kontrollsignale des ReTI Rechners die dazugehörigen *Boolesche Funktionen* für die Kontrollogik.

- Register
  a) *PCcken* (Clock-Enable-Signal für den Akkumulator)
- Treiber (Enable-Signale)
  b) */ALUAdoe* (Verbindung ALU mit Addressbus)
  c) */SMDdoe* (Verbindung SRAM Ausgangstreiber mit Datenbus)
- ALU
  d) f[2:0] (Funktionauswahl der ALU. Codierung wie in Kap. 3.5)

Bestimmen Sie zunächst zu welchen Zeitpunkten und dann unter welchen Bedingungen (Befehlsart) die Signale aktiv sein müssen. Beachten Sie dabei, dass einige Signale active–low sind.

Hinweis: Es ist nicht nötig ein Polynom anzugeben, Sie können wie in der Vorlesung einen beliebigen Booleschen Ausdruck angeben. Geben Sie also Boolesche Ausdrücke für $PCcken_{pre}$, $ALUAdoe_{pre}$, $SMDdoe_{pre}$ an. Wie in der Vorlesung erwähnt, ist $f[2:0]$ nicht ein Ausgang eines FF der Kontrollogik, sondern ein rein kombinatorischer Signal. Es genügt also, direkt entsprechende Boolesche Ausdrücke anzugeben, die zudem nicht von s_0, s_1, und E abhängen müssen.

Für das Signal f[2:0] benötigen Sie keine Kodierung des Taktes.

Aufgabe 2 (4 + 4 Punkte)

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

a) Zeigen Sie, dass die Instruktionen des neuen Rechners mit den Datenpfaden aus Abbildung 1 durchgeführt werden können.

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

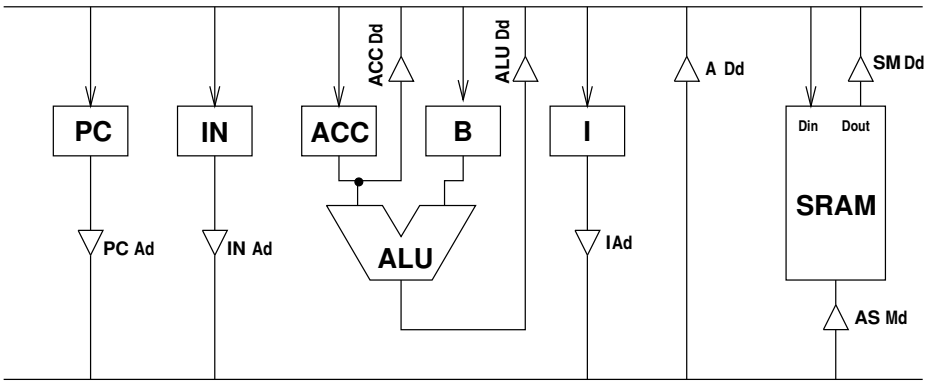

Abbildung 1: Datenpfade des neuen Rechners

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 (3 + 3 Punkte)

Betrachten Sie einen $n-Bit$-Conditional-Sum-Addierer, der aus EXOR-Gattern, $1-Bit$-Multiplexern, AND-Gattern, OR-Gattern und Treibern der NanGate-Bibliothek aufgebaut ist. Auf den Webseiten der Veranstaltung finden Sie unter `Vorlesungsmaterial` → `Hilfsmaterial` eine kurze Übersicht über Verzögerungszeiten und Timing einiger wichtiger Gatter und Komponenten ("NanGate Bibliothek"), die sie u.a. für diese Aufgabe benötigen werden.

a) Geben Sie die maximale Verzögerungszeit an, zu der das Ausgangscarry-Signal schalten kann, unter der Voraussetzung, dass die Inputs zur Zeit $t_0 = 0$ schalten können, aber nicht müssen. Sie können bei der Analyse des $n - Bit$-Conditional-Sum-Addierers zunächst Treiber vernachlässigen, die man eigentlich zum Vervielfältigen eines Signals auf mehr als 10 Eingänge benötigen würde.

b) Geben Sie nun die maximale Verzögerungszeit an, zu der das Ausgangscarry-Signal eines $32 - Bit$-Conditional-Sum-Addierers schalten kann, jetzt aber unter Berücksichtigung der Tatsache, dass zum Vervielfältigen eines Signals auf mehr als 10 Eingänge Treiber benötigt werden.

Hinweis: Bei der Timing-Analyse des $n - Bit$-Conditional-Sum-Addierers stellen die Analyse von Volladdierern und $n - Bit$-Multiplexern natürlich notwendige Teilschritte dar. Ansonsten können Sie sich an der Analyse zur Bestimmung der Tiefe des CSA_n aus der Vorlesung orientieren. An der Stelle des Beitrags 1 eines Gatters zur Tiefe muss nun natürlich seine tatsächliche Verzögerungszeit stehen.

Aufgabe 4 (2 + 4 Punkte)

In der Vorlesung wurde eine Methode vorgestellt (Kap. 5.1, Folie 31ff, *Fall 2*), mit der sich bestimmen lässt, zu welchen Zeitpunkten an den Ausgängen einer Schaltung sicher ein stabiler Wert anliegt, wenn die Eingänge sich zu einem Zeitpunkt t_0 (bzgl. M) ändern, dabei aber unbekannt ist, welche Signale sich ändern und wie sie sich ändern. Vor und nach t_0 sind die Werte stabil.

Betrachten Sie die Schaltung in Abbildung 2.

Für den Inverter der NanGate-Bibliothek sind folgende Schaltzeiten (in ns) angegeben:
$\tau_{PLH} = [0.01, 0.15]$
$\tau_{PHL} = [0.00, 0.08]$

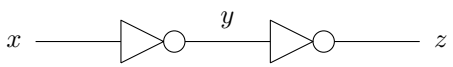

Abbildung 2: Zwei Inverter

a) Berechnen Sie mit der in der Vorlesung angegebenen Methode die Intervalle, in denen die einzelnen Gatter schalten.

   In welchem Zeitbereich kann – basierend auf der in der Vorlesung angegebenen Methode – der Ausgang der Schaltung spätestens schalten?

b) Betrachten Sie nun getrennt voneinander sämtliche möglichen Schaltvorgänge am Eingang der Schaltung und berechnen Sie jeweils die Intervalle, in denen die Gatter schalten. Illustrieren Sie dies jeweils mit einem Timingdiagramm.

   In welchem Zeitraum kann – basierend auf diesen Resultaten – der Ausgang der Schaltung schalten?

   Begründen Sie den Unterschied zwischen den beiden Methoden.

Abgabe: 5. Februar 2016, $17^{\underline{00}}$ über das Übungsportal