

System Architecture - SS15
Exercise Sheet 5(due: May 25, 2015)

Wichtig:

- Sie Benötigen 50% aller Übungsblätter die für Klausur X relevant sind, um zu Klausur X zugelassen zu werden. Dieses Blatt ist Relevant für Vor- und Nachklausur.
- Das Übungsblatt muss stets am Montag nach der Vorlesung bei mir in der Office Hour oder, falls zeitgleich, in der Übungsgruppe Ihrer Tutorin abgegeben werden.
- Geben Sie stets Ihren Namen, Ihre Matr. Nr., und den Namen ihrer Tutorin auf der vordersten Seite oben rechts an.
- Sie dürfen Ergebnisse von vorherigen Aufgaben verwenden, auch wenn Sie diese nicht gelöst haben. Markieren sie Gleichungen, in denen Sie ein vorheriges Ergebniss benutzen, mit dem Kürzel E+Aufgabenblatt+Aufgabennummer.
- Wenn Sie sich nicht für die Klausur vorbereiten möchten, aber trotzdem zugelassen werden möchten, schreiben Sie einfach Ihren Namen und Ihre Matrikelnummer auf die Lösung einer kompetenten Mitstudentin. Es besteht auch keine Anwesenheitspflicht in den Übungsgruppen.

Tutor: _____

Namen, Matr. Nummern: _____

Aufgabe 1: (1)

Erklären Sie den Unterschied zwischen Spezifikation und Definition anhand des Parallel-Prefix-Schaltkreises.

Solution: Spezifikation: beschreibt Verhalten, i.e., Inputs, Outputs, Verhalten der Outputs abhängig von Inputs. Beim PP:

$$y_i = x_i \circ y_{i-1}.$$

Definition: beschreibt Implementierung, i.e., wie die outputs berechnet werden. Beim PP:

$$y_{2i+2} = x_{2i+2} \circ y'_i, y_{2i+1} = y'_i, y_0 = x_0, \dots$$

Aufgabe 2: (4)

Konstruieren Sie einen Schaltkreis, der ein Laufflicht von 8 LEDs implementiert. Die LEDs sind in einer Linie angeordnet, wobei zu jedem Zeitpunkt genau eins der LEDs leuchten soll. Das Laufflicht soll zwei Modi haben:

- Pac-Man: Das Licht bewegt sich stets in die selbe Richtung, und
- Zylon: Das Licht ändert seine Richtung, sobald es die erste oder letzte LED erreicht.

Ihr Schaltkreis soll dazu einen Ausgang $LED \in \mathbb{B}^8$ und Eingang $mode \in \mathbb{B}$ haben, wobei Sie davon ausgehen können dass das i -te LED leuchtet genau dann wenn $LED_i = 1$.

- (a) (1 point) Erstellen Sie eine formale Spezifikation, die die informelle Spezifikation erfüllt (wo die informelle Spezifikation ambig ist, suchen Sie sich eine Möglichkeit aus).

System Architecture - SS15
Exercise Sheet 5(due: May 25, 2015)

- (b) (2 points) Konstruieren Sie den Schaltkreis.
 (c) (1 point) Beweisen Sie, dass in Ihrem Schaltkreis zu jedem Zeitpunkt genau ein LED leuchtet.
 (d) (1 point (bonus)) Beweisen Sie, dass Ihr Schaltkreis Ihre Spezifikation erfüllt.

Solution: Spezifikation: Wir fügen $d^t \in \{-1, +1\}$ für die Richtung ein und definieren,

$$\begin{aligned} LED^0 &= 00000001 \\ d^0 &= 1 \\ d^{t+1} &= \begin{cases} -d^t & (LED_0^t \vee LED_7^t) \wedge mode^t \\ d^t & \text{o.w.} \end{cases} \\ LED_i^{t+1} &= LED_{(i+d^{t+1}) \bmod 8}^t \end{aligned}$$

Der Schaltkreis benutzt ein 8 bit Register LED und ein 1 bit Register D, wobei

$$d^t = 2 \cdot D^t - 1$$

gelten soll.

Es gilt dann

$$D_{in} = reset?1 : D \oplus (mode \wedge (LED_0 \vee LED_7)),$$

und

$$LED_{in}[i] = reset?00000001 : (D_{in}?LED_{i+1 \bmod 8} : LED_{i-1 \bmod 8}).$$

Die Korrektheit zeigen wir durch Induktion über t . Im Zeitpunkt $t = 0$ gilt

$$D^0 = D_{in}^{-1} = reset^{-1}?1 : D^{-1} = 1?1 : D^{-1} = 1,$$

und damit

$$d^0 = 1 = 2 \cdot 1 - 1 = 2 \cdot D^0 - 1.$$

Für die LEDs machen wir die gleiche Berechnung und erhalten

$$LED^0 = 00000001.$$

Im Induktionsschritt gehen wir von t nach $t + 1$. Es gilt per Induktionshypothese

$$d^t = 2 \cdot D^t - 1,$$

und die LEDs sind auch korrekt.

Wir Zeigen zuerst

$$\begin{aligned} D^{t+1} &= D_{in}^t = D^t \oplus ((LED_0^t \vee LED_7^t) \wedge mode^t) \\ &= \begin{cases} \neg D^t & (LED_0^t \vee LED_7^t) \wedge mode^t \\ D^t & \text{o.w.} \end{cases} \end{aligned}$$

System Architecture - SS15
 Exercise Sheet 5(due: May 25, 2015)

Daraus folgt direkt

$$\begin{aligned}
 d^{t+1} &= \begin{cases} -d^t & (LED_0^t \vee LED_7^t) \wedge mode^t \\ d^t & \text{o.w.} \end{cases} \\
 \stackrel{IH}{=} & \begin{cases} -2 \cdot D^t + 1 & (LED_0^t \vee LED_7^t) \wedge mode^t \\ 2 \cdot D^t - 1 & \text{o.w.} \end{cases} \\
 = & \begin{cases} -2 \cdot (1 - \neg D^t) + 1 & (LED_0^t \vee LED_7^t) \wedge mode^t \\ 2 \cdot D^t - 1 & \text{o.w.} \end{cases} \\
 = & \begin{cases} 2 \cdot \neg D^t - 1 & (LED_0^t \vee LED_7^t) \wedge mode^t \\ 2 \cdot D^t - 1 & \text{o.w.} \end{cases} \\
 = & 2 \cdot D^{t+1} - 1.
 \end{aligned}$$

Als letztes zeigen wir die Korrektheit der LEDs:

$$\begin{aligned}
 LED_i^{t+1} &= (D_{in}^t ? LED_{i+1 \bmod 8}^t : LED_{i-1 \bmod 8}^t) \\
 &= LED_{(i+2 \cdot D_{in}^t - 1) \bmod 8}^t \\
 &= LED_{(i+d^{t+1}) \bmod 8}.
 \end{aligned}$$

Dass stets nur ein LED leuchtet Zeigen wir per Induktion. Es gilt

$$\#\{i \mid LED_i^0 = 1\} = 1.$$

Im den Induktionsschritt gilt

$$\#\{i \mid LED_i^{t+1} = 1\} = \#\{i \mid LED_{i+d^{t+1} \bmod 8}^t = 1\},$$

und da die Funktion $(i \mapsto i + d^{t+1} \bmod 8)$ eine Permutation ist gilt direkt

$$\#\{i \mid LED_{i+d^{t+1} \bmod 8}^t = 1\} = \#\{j \mid LED_j^t = 1\} \stackrel{IH}{=} 1.$$

Aufgabe 3:

(1)

- (a) (1 point) Was sind Kosten und Tiefe eines ROMs aus der Vorlesung?
 (b) (1 point (bonus)) Wieso kann man RAMs für programmierbare Hardware benutzen (z.B. sog. FPGAs)?

Solution: Adressraum \mathbb{B}^a , Wortbreite n

Tiefe $a + 1 + \log a$

Kosten $2^a \cdot n + 2^a \cdot n + \log a \cdot 2^{2a} + 1$ Bonus wenn die Kosten vom Decoder besser Abgeschätzt wurden (inklusive Beweis natürlich).

System Architecture - SS15
Exercise Sheet 5(due: May 25, 2015)

Sei $f : \mathbb{B}^a \rightarrow \mathbb{B}$ eine Schaltfunktion. Berechnet man alle Möglichen Werte ($f(x)$ für alle $x \in \mathbb{B}^a$) und schreibt diese in einen a -bit RAM, so berechnet dieser f . Die Tiefe von $a + 1 + \log a$ liegt unter der Tiefe von $3 \cdot a$, die z.B. die naive Konstruktion aus Aufgabe 2.6 hat.

Aufgabe 4: **(2)**

Definieren Sie die Registernamen

$$rt(c), rs(c), rd(c)$$

und die Werte der Operanden für die ALU,

$$lop(c), rop(c).$$

Aufgabe 5: **(3)**

Geben Sie für jeden Typ (I,J,R) eine gültige Instruktion $I \in \mathbb{B}^{32}$ an, und Spezifizieren Sie den Effekt ihrer Instruktion auf eine MIPS Konfiguration.

Solution:

IType 001111 00000 10010 10010010 00101001 :

$$c'.gpr(10010) = 10010010001010010000000000000000, c'.pc = c.pc +_{32} 0^{29}100$$

RType 000000 001000 0000 00000000 00000000 :

$$c'.pc = c.gpr(00000)$$

JType 000010 00 00000000 00000000 00000000 :

$$c'.pc = c.pc[31 : 28]00000000000000000000000000000000$$

Aufgabe 6: **(1)**

Definieren Sie ein MIPS Makro $GPR(j) = x$, das den Wert der Konstante x in Register j lädt. Sie dürfen Mnemonics verwenden.

Solution:

$$\text{lui } j \ x[31 : 16]; \text{ori } j \ j \ x[15 : 0]$$