Universität des Saarlandes
FR 6.2 - Informatik
Prof. Dr. W.J. Paul
Dipl. Inform. M. Klein
Dipl. Inform. T. In der Rieden

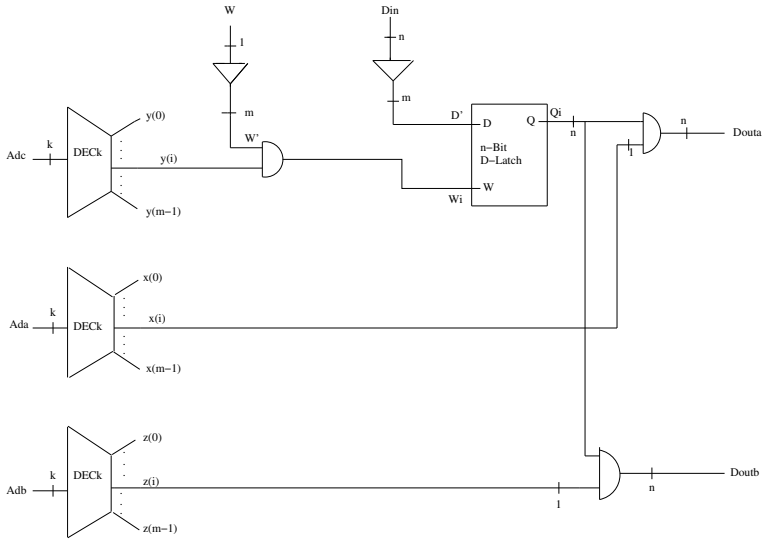
6. Übungsblatt Informatik II

(Abgabe: 06.06.2003)

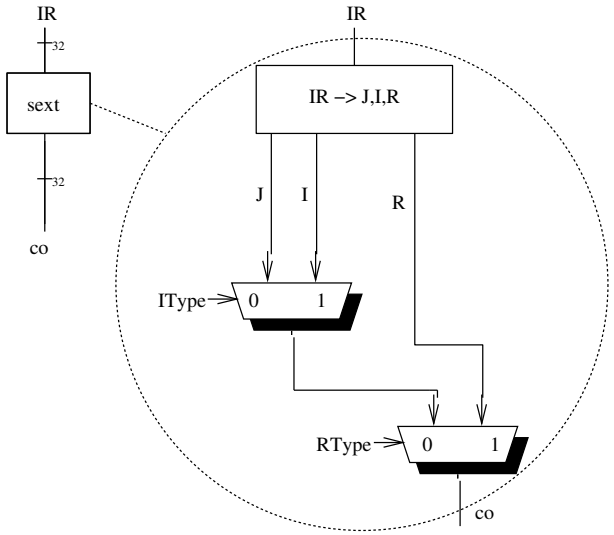
Notation:

- Für natürliche Zahlen $t > 0$ und Register x bezeichne x^t den Inhalt von Register x nach t Takten und x^0 den definierten initialen Wert des Registers x.
- Für natürliche Zahlen $t > 0$ und Signale s bezeichne s^t den Wert des Signals s am Ende von Takt t.

1. **Aufgabe: (3 Port RAM)** **(10 Punkte)**
   Sei m=2^k



2. **Aufgabe: (Sign Extension)** **(6 Punkte)**



$I = IR[15]^{16}IR[15:0]$
$J = IR[25]^{6}IR[25:0]$
$R = 0^{27}IR[10:6]$

Universität des Saarlandes
FR 6.2 - Informatik
Prof. Dr. W.J. Paul
Dipl. Inform. M. Klein
Dipl. Inform. T. In der Rieden

6. Übungsblatt Informatik II

(Abgabe: 06.06.2003)

Anmerkung: Der (IR–>J,I,R)-Box hat die Funktion aus dem Instruktionsregister nach obigen Definitionen J, I und R zu erzeugen. Dafür braucht man natürlich keinen Gatter, sondern nur die entsprechende Verdrähtungen.

3. **Aufgabe: (Günstiger Addierer)** **(4 + 8 Punkte)**

(a) Da wir lediglich einen einzigen Volladdierer zur Verfügung haben, ist es unmöglich, die Inhalte der Register x und y als Ganzes zu addieren. Die einzige Chance besteht darin, x und y bitweise zu addieren, wobei jeweils der Ausgangsübertrag der vorigen Berechnung mit einbezogen werden muss.

Die Schwierigkeit dabei ist, dem Volladdierer jeweils die korrekten Eingabebits zuzuführen, da man (ohne irgendeine zusätzliche Steuerung) keine Möglichkeit hat, das gerade benötigte Bit aus den insgesamt n Bits des Registers x auszuwählen (für y analog).

Die Lösung besteht darin, immer die jeweils niederwertigsten Bits der Register x und y zu addieren (d.h. $a = Q_{x,0}$ und $b = Q_{y,0}$ mit a, b als Eingänge des Volladdierers) und anschließend die höherwertigen Bits jeweils um eine Stelle „nach unten zu shiften“: $D^t_{x,i} = D^{t-1}_{x,i+1}$ mit $0 < t \leq n$ und $0 \leq i < n-1$. Zusätzlich sei noch $D^t_{x,n-1} = D^{t-1}_{x,0}$ (t wie zuvor). Das ist nicht unbedingt nötig, aber dadurch ist am Ende der Berechnungen der Wert im Register wieder gleich x. Diese Regeln gelten analog für y.

Der Eingangsübertrag c_{in} des Volladdierers ist gerade gleich dem im vorigen Takt berechneten Ausgangsübertrag c_{out}. Dieser Wert und das Summenbit z werden im Register s gespeichert, in dem zu Beginn der Berechnung nur das oberste Bit gleich 0 ist, alle anderen sind undefiniert. Dieses oberste Bit ist in allen Runden der Eingangsübertrag des Volladdierers (zu Beginn natürlich 0): $c_{\text{in}} = Q_{s,n}$.

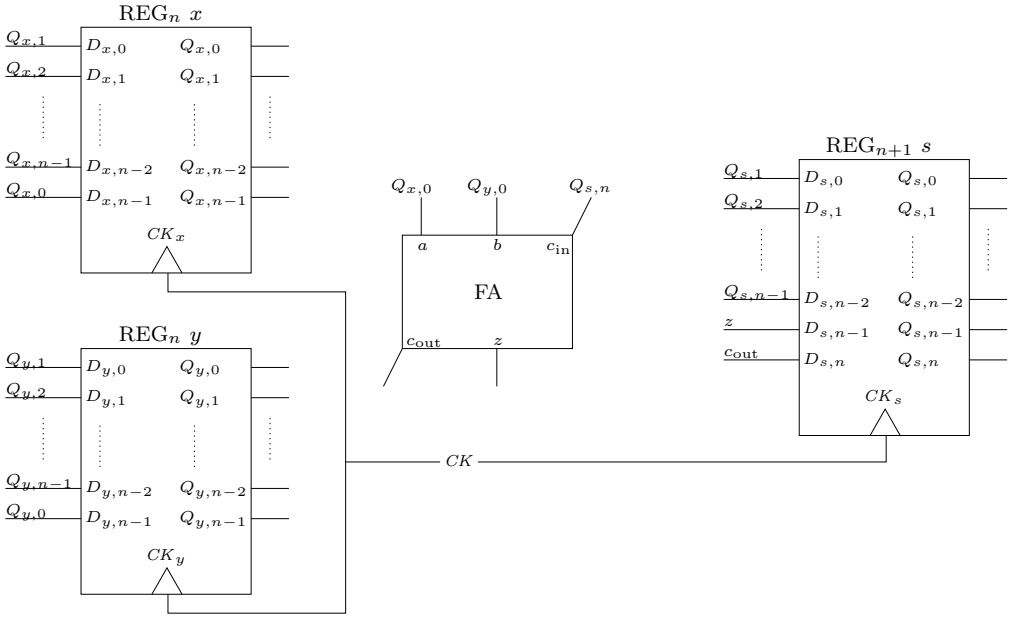
Das Ergebnis der Berechnung des Volladdieres wird nach jedem Takt in den beiden obersten Bits des Registers s gespeichert: $D^t_{s,n-1} = z$ und $D^t_{s,n} = c_{\text{out}}$ mit $0 < t \leq n$. Damit nach n Runden das korrekte Ergebnis im Register s steht, muss auch hier „geshiftet“werden: $D^t_{s,i} = D^{t-1}_{s,i+1}$ mit $0 < t \leq n$ und $0 \leq i < n-1$. Dieser Shift darf hier *nicht* zyklisch sein (wie bei x und y), da ansonsten das Ergebnis der Berechnung des Volladdierers überschrieben würde.

Der fertige Schaltkreis sieht damit wie folgt aus:

Universität des Saarlandes
FR 6.2 - Informatik
Prof. Dr. W.J. Paul
Dipl. Inform. M. Klein
Dipl. Inform. T. In der Rieden

6. Übungsblatt Informatik II

(Abgabe: 06.06.2003)



Anmerkung: Die oben verwendeten Variablen $Q_{i,j}$ stellen lediglich Hilfsvariablen dar, um die Ausgänge der Register zu kennzeichnen. Sie enthalten die entsprechenden Registerinhalte, *bevor* diese am Ende von einem Takt t geändert wurden, d.h. $Q_{i,j} = D_{i,j}^{t-1}$.

(b) Zu zeigen ist:

$$\langle s^n \rangle = \langle x^0 \rangle + \langle y^0 \rangle$$

Dies soll nun gezeigt werden durch Induktion über die Anzahl der Takte (betrachtet wird jeweils die Situation am Ende eines Taktes t). Die Korrektheit von Register und Volladdierer wird dabei vorausgesetzt.

- **Induktionsanfang:** $n = 1$

$$\begin{aligned}
\langle D_{s,1}^1, D_{s,0}^1 \rangle &\overset{\text{Konstr.}}{=} \langle c_{\text{out}}, z \rangle \\
&\overset{\text{Korr. FA}}{=} Q_{x,0} + Q_{y,0} + Q_{s,n} \\
&= D_{x,0}^0 + D_{y,0}^0 + \underbrace{D_{s,n}^0}_{=0} \\
&= x_0^0 + y_0^0
\end{aligned}$$

- **Induktionsvoraussetzung:** Für $n-1$ sei

$$\langle D_{s,n}^{n-1}, \ldots, D_{s,1}^{n-1} \rangle = \langle x^0[n-2:0] \rangle + \langle y^0[n-2:0] \rangle$$

bereits bewiesen.

- **Induktionsschritt:** $n - 1 \to n$

$$\begin{aligned}
\langle s^n \rangle &= \langle D_{s,n}^n, \ldots, D_{s,0}^n \rangle \\
&= \langle D_{s,n}^n, D_{s,n-1}^n \rangle \cdot 2^{n-1} + \langle D_{s,n-2}^n, \ldots, D_{s,0}^n \rangle \\
&\overset{\text{Konstr.}}{=} \langle c_{\text{out}}, z \rangle \cdot 2^{n-1} + \langle D_{s,n-2}^n, \ldots, D_{s,0}^n \rangle \\
&\overset{\text{Konstr.}}{=} \langle c_{\text{out}}, z \rangle \cdot 2^{n-1} + \langle D_{s,n-1}^{n-1}, \ldots, D_{s,1}^{n-1} \rangle
\end{aligned}$$

Universität des Saarlandes
FR 6.2 - Informatik
Prof. Dr. W.J. Paul
Dipl. Inform. M. Klein
Dipl. Inform. T. In der Rieden

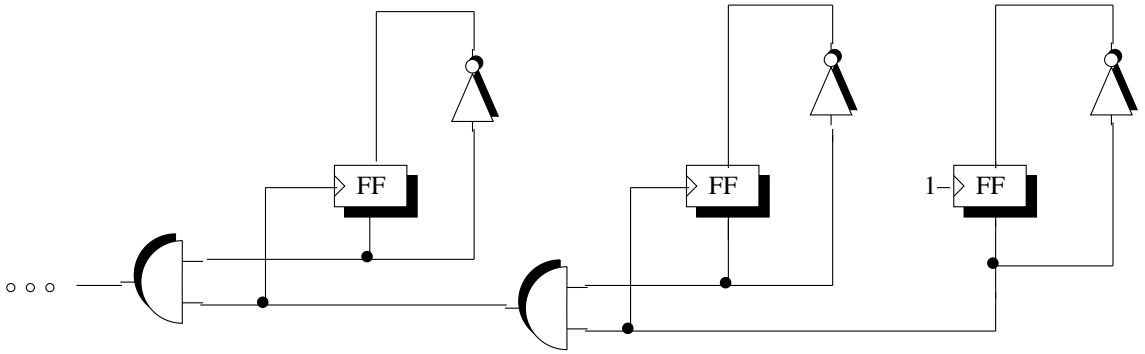
6. Übungsblatt Informatik II

(Abgabe: 06.06.2003)

$$
\begin{aligned}
&\overset{\text{Korr. FA}}{=} && (Q_{x,0} + Q_{y,0} + Q_{s,n}) \cdot 2^{n-1} + \langle D^{n-1}_{s,n-1}, \ldots, D^{n-1}_{s,1} \rangle \\
&= && (D^{n-1}_{x,0} + D^{n-1}_{y,0} + D^{n-1}_{s,n}) \cdot 2^{n-1} + \langle D^{n-1}_{s,n-1}, \ldots, D^{n-1}_{s,1} \rangle \\
&\overset{(*)}{=} && (x^0_{n-1} + y^0_{n-1}) \cdot 2^{n-1} + \langle D^{n-1}_{s,n-1}, \ldots, D^{n-1}_{s,1} \rangle \\
&\overset{\text{IV}}{=} && (x^0_{n-1} + y^0_{n-1}) \cdot 2^{n-1} + \langle x^0[n-2:0] \rangle + \langle y^0[n-2:0] \rangle \\
&= && \langle x^0[n-1:0] \rangle + \langle y^0[n-1:0] \rangle \\
&= && \langle x^0 \rangle + \langle y^0 \rangle
\end{aligned}
$$

Anmerkung: Die Gleichung $(*)$ gilt, weil bis zu Takt n gerade so weit geshiftet wurde, dass sich jeweils die höchstwertigsten Bits von x und y auf den niederwertigsten Positionen der jeweiligen Register befinden.

4. **Aufgabe: (Zähler)** **(6 Punkte)**



Wie wir sehen, haben wir n Stück FFs, n Inverters und n-1 Und-Gattern.

5. **Aufgabe: (Nand-Darstellung)** **(6 Punkte)**
Bei dieser Aufgabe sollen wir die Vollständigkeit des $\overline{\wedge}$-Operators zeigen. Das machen wir, indem wir alle Gattern (Boolesche Operationen) per $\overline{\wedge}$ simulieren. Es muss klar sein, dass wir hier nur $\neg$ und $\vee$ (bzw. $\wedge$) unter Verwendung $\overline{\wedge}$ darstellen sollen, da man $\wedge$ (bzw. $\vee$) nach DeMorgan Regeln mit Negierung einer Veroderung (bzw. Verundung) erhalten kann.

Seien e_1 und e_2 Boolesche Ausdrücke:

$\neg(e_1) \equiv e_1 \mathbin{\overline{\wedge}} e_1$
$e_1 \vee e_2 \equiv (e_1 \mathbin{\overline{\wedge}} e_1) \mathbin{\overline{\wedge}} (e_2 \mathbin{\overline{\wedge}} e_2)$

6. *** Aufgabe: (Von Neumann Addierer)** **(4 + 4 + 2 Punkte *)**