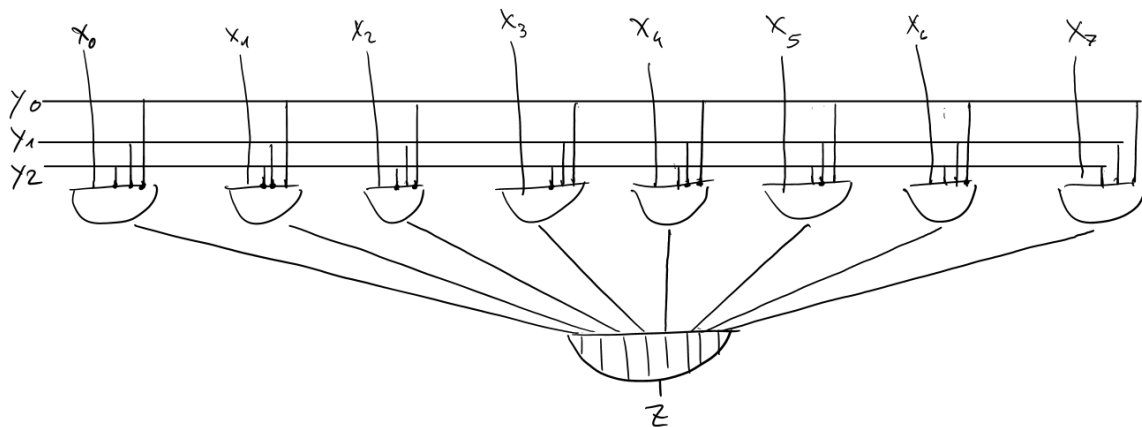


GTI Aufgaben Serie 3

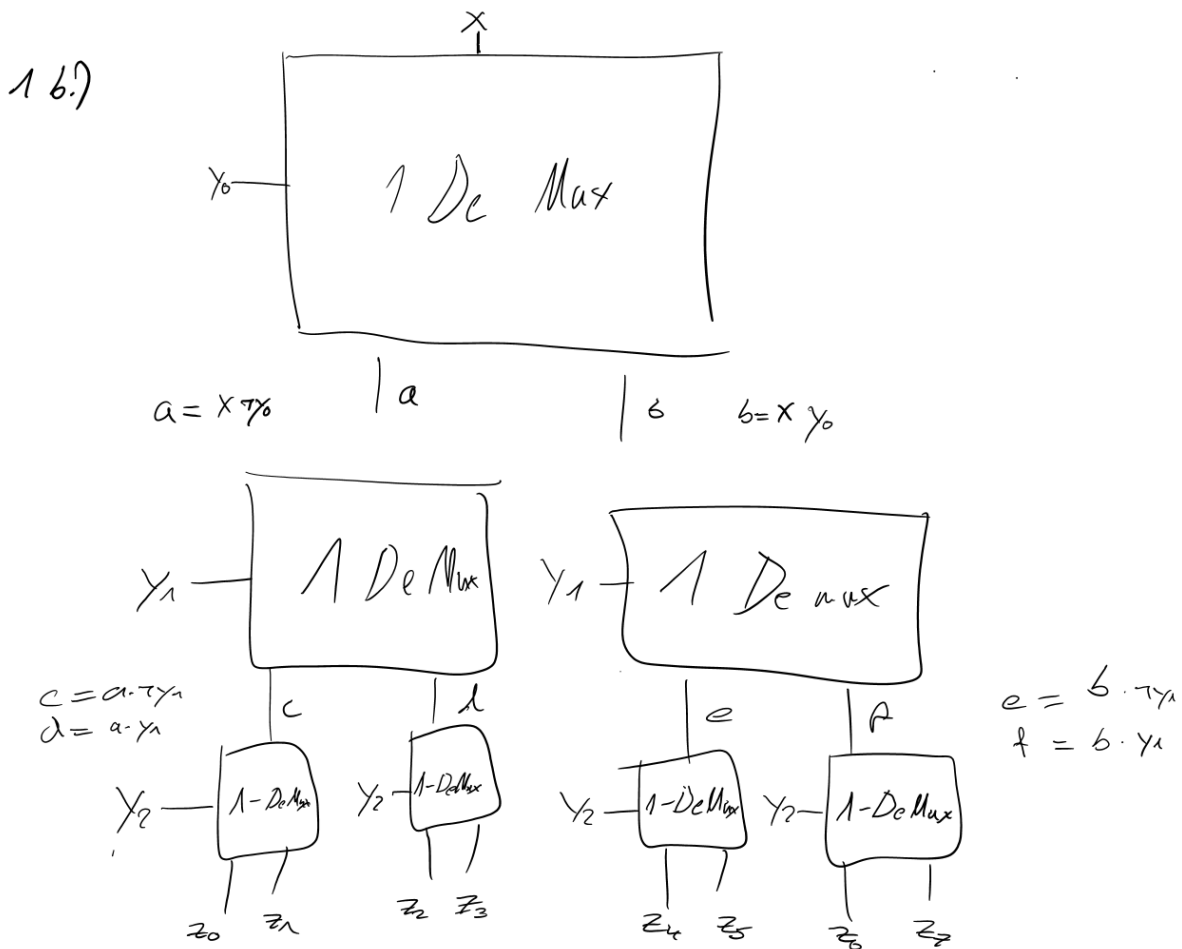
Vithusan Ramalingam (21-105-515)

Jan Ellenberger (21-103-643)

Aufgabe 1.a) Stelle einen 3-Mux (8-to-1-Multiplexer) als Schaltung in disjunktiver Form (d. h. als zweistufige Schaltung, die eine Disjunktion von Konjunktionen ist) dar.



$$\begin{aligned}
 f(x_0, x_1, x_2, x_3, x_4, x_5, x_6, x_7) &= (x_0 \neg x_1 \neg x_2 \neg x_3) + (x_1 \neg x_0 \neg x_2 \neg x_3) + (x_2 \neg x_0 \neg x_1 \neg x_3) + (x_3 \neg x_0 \neg x_1 \neg x_2) \\
 &\quad + (x_4 x_0 \neg x_1 \neg x_2) + (x_5 \neg x_0 x_1 \neg x_2) + (x_6 x_0 x_1 \neg x_2) + (x_7 \neg x_0 x_1 x_2)
 \end{aligned}$$

Aufgabe 1. b) Bestimme die Schaltfunktionen eines 3-DeMux (1-to-8-Demultiplexer).

$F: B^3 \rightarrow B^1:$

$$\begin{aligned}
 z_0 &= a \cdot c \cdot \neg Y_2 = X \cdot \neg Y_0 \cdot \neg Y_1 \cdot \neg Y_2 \\
 z_1 &= a \cdot c \cdot Y_2 = X \cdot \neg Y_0 \cdot \neg Y_1 \cdot Y_2 \\
 z_2 &= a \cdot d \cdot \neg Y_2 = X \cdot \neg Y_0 \cdot Y_1 \cdot \neg Y_2 \\
 z_3 &= a \cdot d \cdot Y_2 = X \cdot \neg Y_0 \cdot Y_1 \cdot Y_2 \\
 z_4 &= b \cdot e \cdot \neg Y_2 = X \cdot Y_0 \cdot \neg Y_1 \cdot \neg Y_2 \\
 z_5 &= b \cdot e \cdot Y_2 = X \cdot Y_0 \cdot \neg Y_1 \cdot Y_2 \\
 z_6 &= b \cdot f \cdot \neg Y_2 = X \cdot Y_0 \cdot Y_1 \cdot \neg Y_2 \\
 z_7 &= b \cdot f \cdot Y_2 = X \cdot Y_0 \cdot Y_1 \cdot Y_2
 \end{aligned}$$

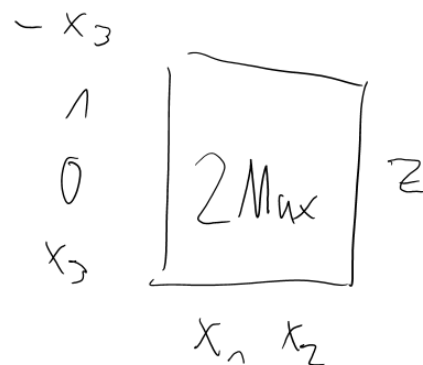
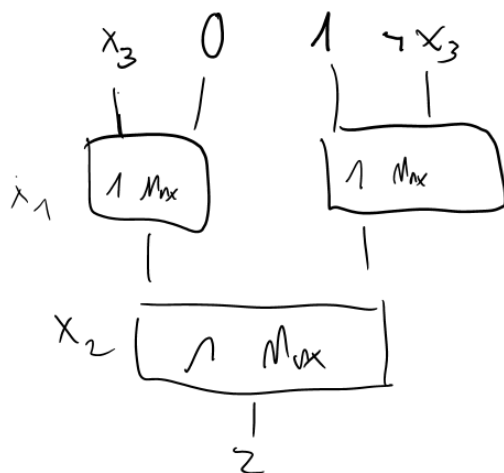
Aufgabe 2.) Realisiere die folgende Funktion $f(x_1, x_2, x_3)$ mittels eines 2-Muxs (4-to-1-Multiplexer).
Zusätzliche Negationen bei Eingängen des Muxs sind erlaubt.

x_1	x_2	x_3	f
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

2)

x_1	x_2	x_3	f
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

x_1	x_2	z
0	0	$\neg x_3$
0	1	1
1	0	0
1	1	x_3

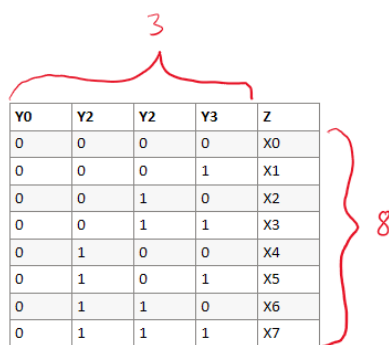


Aufgabe 3. Mux, DeMux und Decoder**(a) Wieviele Outputsignale hat ein Decoder mit d Steuersignalen?**

$$2^d$$

(b) Wieviele Steuersignale hat ein Mux mit d Inputsignalen?

$$-\log_2(d)$$

(c) Was ist die kleinste Anzahl Eingangssignale (Steuer- und Datensignale) für einen Mux (zusätzliche Negation bei Eingängen erlaubt), damit jede beliebige Funktion $f: B^4 \rightarrow B$ damit dargestellt werden kann? Begründe die Antwort. (Tipp: Es sind weniger als $2 \cdot 4 + 4 = 20$.)


Y0	Y2	Y2	Y3	Z
0	0	0	0	X0
0	0	0	1	X1
0	0	1	0	X2
0	0	1	1	X3
0	1	0	0	X4
0	1	0	1	X5
0	1	1	0	X6
0	1	1	1	X7
1	0	0	0	~X0
1	0	0	1	~X1
1	0	1	0	~X2
1	0	1	1	~X3
1	1	0	0	~X4
1	1	0	1	~X5
1	1	1	0	~X6
1	1	1	1	~X7

Wir sehen dass es zu jeder Kombination von Schaltsignalen eine Inverse Kombination gibt. Für $y_0, y_1, y_2, y_3 = 0, 0, 0, 0$ gibt es z.B. 1, 1, 1, 1.

Wir können also jeden Input einmal normal und einmal negiert rechnen das reduziert die Anzahl der benötigten Inputs von 16 auf 8, da wir sie im Eingang des Multiplexer einfach noch negieren. Somit ist die minimale Anzahl an Eingangssignalen $3 + 8 = \underline{11}$.

(d) Welcher der drei Begriffe Mux, DeMux, Decoder passt jeweils zu den folgenden Aussagen ?**1. Kann durch Ersetzen des Inputsignales mit einer 1 zu einem Decoder gemacht werden.**

- DeMux

2. Ist universell, d. h. damit lässt sich jede boolesche Funktion realisieren (mit entsprechend genügend vielen Signalleitungen).

- Mux

3. Erzeugt genau an einer Outputstelle eine 1.

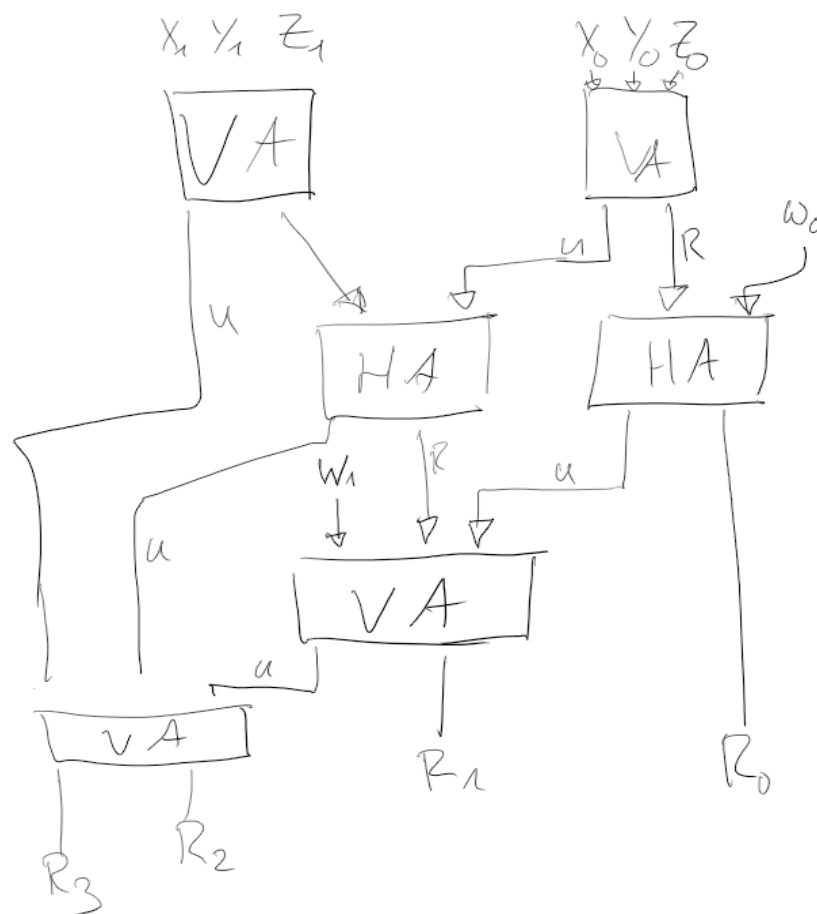
- Decoder

Aufgabe 4.) Erkläre wie mittels eines Multiplexers mehrere parallele Signale in ein serielles Signal umgewandelt werden können (Serialisierer). Welche Bedingung muss gegeben sein, damit der Empfänger der Nachricht das serielle Signal mit einem Demultiplexer wieder in das genau gleiche parallele Signal umwandeln kann?

Um den Input beim multiplexer serialisieren zu können, brauchen wir einen Zähler bei den Schaltsignalen. Indem wir bei den Schaltsignal hochzählen können wir vom parallelen Signal zuerst die 0.te Stelle ausgegeben dann die erste, dann die zweite und so weiter. So hat man dann ein serielles Signal, dass den Input der Reihenfolge nach ausgibt. Um das serielle Signal wieder mit einem Demultiplexer parallel ausgeben zu können zählen wir bei den Steuersignalen wieder mit einem Zähler hoch, dann kann die 0. Stelle des seriellen signals an der 0. Stelle des Output ausgegeben werden, dann die erste, dann die zweite, dann die dritte und so weiter. Der Multiplexer und der Demultiplexer müssen gleich schnell getaktet sein, damit der Demultiplexer auch wirklich das erste Signal an der ersten Stelle des Output ausgibt. Wenn man den Output bei jedem Schritt etwas mehr verzögert hat man das Signal beim Output wieder parallel.

Aufgabe 5.)

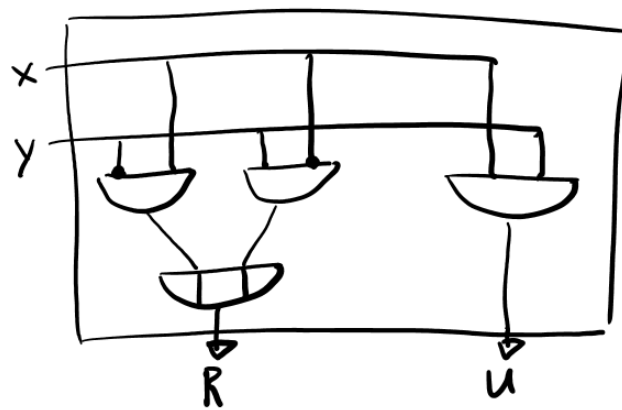
Gegeben seien fünf Volladdierer und zwei Halbaddierer. Bilde daraus ein Addier-Netzwerk, das die Addition von vier zweistelligen Dualzahlen realisiert.



Aufgabe 6.)

- a) Realisiere einen Halbaddierer mit AND- und OR-Gattern. Du darfst Gatter mit eingebauter Negation im Eingang benutzen.

Auf 6.)a) $R = \text{Resultat}$
 $u = \text{Übertrag}$
Halbaddierer:



Aufgabe 6.b)

Zeige (durch Induktion nach dem Aufbau der Schaltung), dass Schaltungen, welche nur aus AND- und OR-Gattern bestehen, immer 1 ausgeben, wenn alle Inputsignale auf 1 gesetzt sind.

Auf 6b.)

Wenn alle Input signale 1 sind dann sind $y, z, w, \dots = x$
 aus der booleschen logik wissen wir $x \vee x = x \wedge x = x$ wenn $x = 1$
 ob wir in nachfolgenden Beweis \vee oder \wedge verwenden ist gleichgültig

Induktionsanfang $n = 1$

$$\begin{aligned} A_1) \quad n &= 1 && \left. \begin{array}{l} \text{wir verknüpfen } x \text{ mit } n\text{-Gatter mit } x \\ \Rightarrow x \vee x = 1 \\ 1 \vee 1 = 1 \\ 1 &= 1 \quad \checkmark \end{array} \right\} \end{aligned}$$

Induktionsannahme

wir beweisen für ein beliebiges $n \geq 1$

$$\begin{aligned} n &= 1 && \left\{ \begin{array}{l} n \neq 1 \text{ oder } 1 \text{ Gatter} \\ \text{also multiplizieren wir } n \text{ mal so ein Gatter} \end{array} \right. \\ n \cdot (x \vee x) &= 1 \\ n \cdot (1) &= 1 \\ n &= 1 \end{aligned}$$

Induktionsschritt

$$(n+1) \cdot (x \vee x) = 1$$

$$(n \cdot x \vee x) + 1 \cdot (x \vee x) = 1$$

$$1 + 1 \cdot (x \vee x) = 1$$

$$1 \vee (x \vee x) = 1$$

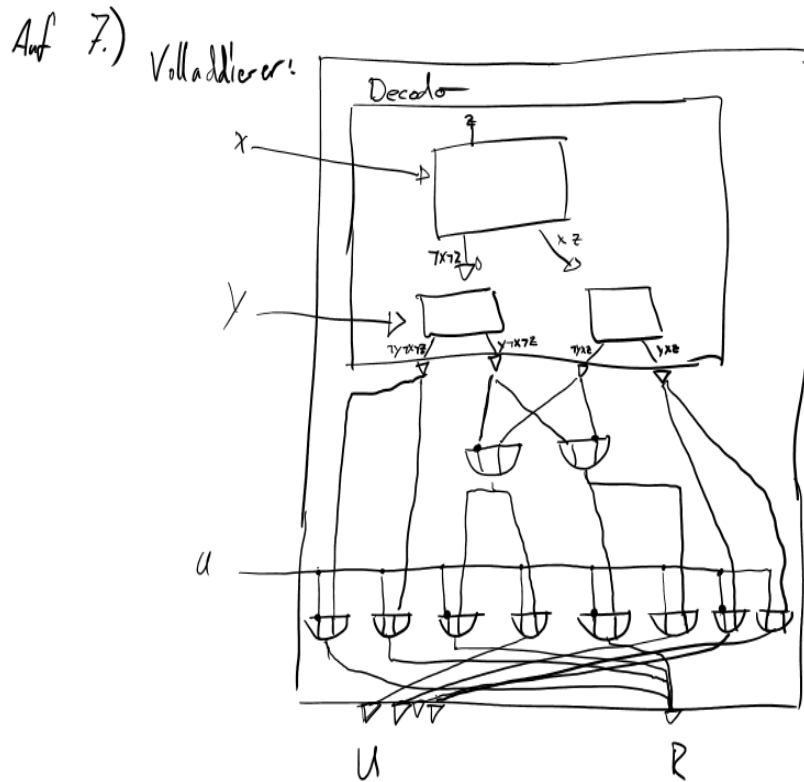
$$1 \vee 1 = 1$$

$$1 = 1$$



Aus der Induktionsannahme
 wissen wir dass $n \cdot x \vee x = 1$ ist
 \Rightarrow wir setzen das ein

Der Beweis kann analog mit \wedge Verknüpfung gemacht werden, ist jedoch obsolet da $1 \wedge 1 = 1 \vee 1 = 1$ in boolescher logik.

Aufgabe 7.)**Realisiere einen Volladdierer mittels eines Decoders und OR-Gatter.**

x	y	u	R	U
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1