Rechnersysteme und -netze 19.12.2019
Bastian Goldlücke, Christoph Doell,
Gregor Diatzko, Johanna Giel, Alice Hildebrand, Josua Sattler, Anton Zickenberg

Name	Übung	Immatrikulationsnummer

Aufgabe 1	Aufgabe 2	Aufgabe 3	Summe
/ 9	/16	/8	/111(+8)

Aufgabe 4	Aufgabe 5	Aufgabe 6	Aufgabe 7	Note
/10	/10	/12	/10	

Aufgabe 1 Boolesche Operationen ($2 + 4 + 3$ Punkte)

Wir betrachten die zwei binären Booleschen Operationen $\mathsf{NOR}(a, b)$ (auch $a \downarrow b$, "Gegenteil des logischen Oder"), und $\mathsf{Replikation}(a, b)$ ($a \leftarrow b$, Werte in Tabelle gegeben).
Lösen Sie die folgenden Teilaufgaben mit Hilfe Boolescher Formeln, **nicht** durch Schaltpläne mit (Logik-)Gattern! (Sie dürfen eine Negation auch durch einen Querstrich über einen Variablennamen bzw. einer Teilformel darstellen, aber schreiben Sie bitte die Konjunktion als $\wedge$ und die Disjunktion als $\vee$.)

a) Stellen Sie die Operatoren NOR und Replikation mit Hilfe der Operatoren Konjunktion $\wedge$, Disjunktion $\vee$ und Negation $\neg$ dar!

b) Stellen Sie die Operatoren $\downarrow$ und $|$ nur mit Hilfe von $\leftarrow$ und $\neg$ dar.

c) Zeigen Sie, dass die Operationenmenge O, die nur aus den Operationen Replikation $\leftarrow$ (siehe Wahrheitstafel rechts) und Negation $\neg$ besteht (d.h., $O = \{\leftarrow, \neg\}$), vollständig ist!

a	b	$a \leftarrow b$
0	0	1
0	1	0
1	0	1
1	1	1

Aufgabe 2 Minimierung: Karnaugh–Veitch-Diagramme ($2 + 4 + 2 + 8$ Punkte)

Gegeben sei die unten in Form eines Karnaugh–Veitch-Diagramms gezeigte Wahrheitstafel einer vierstelligen Booleschen Funktion $f(A, B, C, D)$.

a) Stellen Sie die disjunktive Normalform der Funktion f auf!

b) Minimieren Sie die Darstellung der Funktion f mit Hilfe des Karnaugh–Veitch-Verfahrens unter Bezugnahme auf die Nullen in der Ausgabe!

c) Zeigen Sie semantisch die Gültigkeit der folgenden Äquivalenz:

$$B\overline{C} \vee AC \Leftrightarrow (B \vee C) \wedge (A \vee \overline{C})$$

d) Zeigen Sie mit Boolescher Algebra, dass die Funktionen aus a) und b) (syntaktisch) äquivalent sind! Geben Sie für jeden Schritt die Rechenregel an! Sie dürfen ausserdem die Äquivalenz aus c) verwenden.

CD \ AB	00	01	11	10
00	0	0	0	0
01	0	1	1	0
11	0	0	1	1
10	0	0	0	0

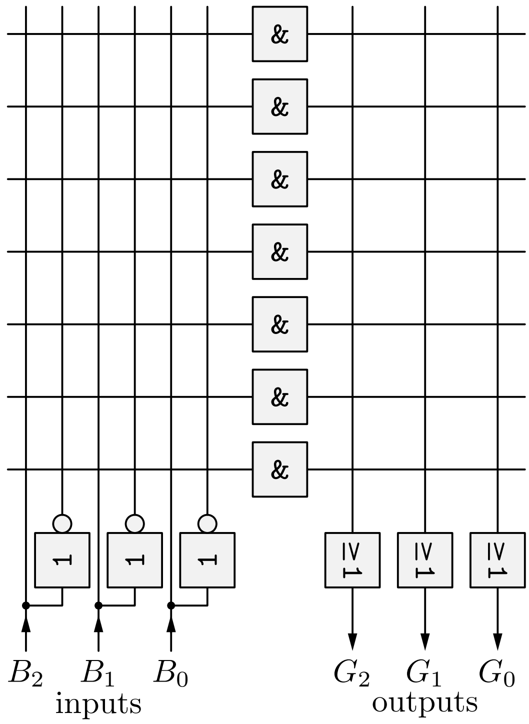
Aufgabe 3 Programmierbare Logikarrays ($3+5$ Punkte)

Gegeben sei die rechts gezeigte Übersetzungstabelle eines dreistelligen Binärkodes $[B_2B_1B_0]$ in einen dreistelligen Gray-Kode $[G_2G_1G_0]$:

Binärkode			Gray-Kode		
B_2	B_1	B_0	G_2	G_1	G_0
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	1
0	1	1	0	1	0
1	0	0	1	1	0
1	0	1	1	1	1
1	1	0	1	0	1
1	1	1	1	0	0

a) Fassen Sie die Spalten des Gray-Kodes als Boolesche Funktionen der Spalten des Binärkodes auf und geben Sie für G_0 die konjunktive Normalform an! Wäre es sinnvoll, stattdessen die disjunktive Normalform zu verwenden? Begründen Sie Ihre Antwort!

b) Implementieren Sie die Berechnung des Gray-Kodes (alle drei Spalten) mit Hilfe eines (programmierbaren) Logikarrays! Verwenden Sie dazu das rechts gezeigte Schema!



Aufgabe 4 Minimierung: Quine–McCluskey ($7+2+1$ Punkte)

Gegeben die folgende Funktion $f(A,B,C,D)$:

$$\begin{aligned} f(A,B,C,D) &= \overline{A}B\overline{C}\,\overline{D} \vee AB\overline{C}\,\overline{D} \vee \overline{A}\,\overline{B}\,\overline{C}D \vee AB\overline{C}D \vee \\ &\quad A\overline{B}\,\overline{C}D \vee ABCD \vee \overline{A}BC\overline{D} \vee ABC\overline{D} \end{aligned}$$

a) Erstellen Sie die Implikantentabellen für die Funktion f mit Hilfe des Quine–McCluskey-Algorithmus.

b) Welche sind die wesentlichen Primimplikanten?

c) Geben Sie die minimierte Form von f an!

Aufgabe 5 Arithmetik $(2+2+3+3$ Punkte$)$

Gegeben seien die beiden Binärzahlen

$$a = \texttt{01110}_{2K} \qquad \text{und} \qquad b = \texttt{11101}_{2K}.$$

a) Berechnen Sie $a+b$ im Binärsystem!
Überprüfen Sie Ihre Rechnung im Dezimalsystem!

b) Berechnen Sie $a-b$ im Binärsystem!
Überprüfen Sie Ihre Rechnung im Dezimalsystem!

c) Berechnen Sie $a \cdot b$ mit dem Standardalgorithmus im Binärsystem!
Überprüfen Sie Ihre Rechnung im Dezimalsystem!

d) Berechnen Sie $a \cdot b$ mit Booths Algorithmus im Binärsystem!

Aufgabe 6 Arithmetisch-logische Einheit $(2+10$ Punkte$)$

In der Vorlesung wurde die arithmetisch-logische Einheit (ALU) der Hack-Architektur aus „The Elements of Computing Systems: Building a Modern Computer from First Principles“ von Noam Nisan und Shimon Schocken betrachtet. Die von dieser ALU zu berechnende Funktion wird durch sechs Steuerbits bestimmt: `zx`, `nx`, `zy`, `ny`, `f` und `no`.

a) Füllen Sie die Lücke in der Überschrift der folgenden Tabelle!

b) Vervollständigen Sie die Zeilen in der Tabelle, so dass die Steuerbits die korrekten Ausgaben verursachen. *Hinweis: Ihr Ergebnis muss nicht der Tabelle aus der Vorlesung entsprechen.*

These bits instruct how to preset the input `x`		These bits instruct how to preset the input `y`		This bit selects between	This bit instructs how to postset the output `out`	Resulting ALU output
`zx`	`nx`	`zy`	`ny`	`f`	`no`	`out=`
0	0	0	0	0	0	
1	1	1	1	1	1	
						0
						x+1
						y-x

Aufgabe 7 Allgemeines $(2+2+4+2$ Punkte$)$

a) Wie heißen die beiden Zeichen φ und ψ?

b) Nach welchen Logikern sind die Zeichen $|$ und $\downarrow$ benannt? Bitte achten Sie auf die korrekte Schreibweise!

c) Nennen Sie die vier Stufen der Hardware-Hierarchie eines Rechners!

d) Wodurch unterscheiden sich die Harvard- und die von-Neumann-Architektur?
Was sind ihre jeweiligen Vor- und Nachteile?

Zeigen Sie mit Boolescher Algebra, daß die Funktionen aus a) und b) (syntaktisch) äquivalent sind! Geben Sie für jeden Schritt die Rechenregel an!

Lösung:

$$\begin{aligned}
f(A,B,C,D) &= AB\overline{C}D \vee \overline{A}B\overline{C}D \vee ABCD \vee A\overline{B}CD \\
&\stackrel{!}{=} \overline{\overline{D} \vee (C \wedge \overline{A}) \vee (\overline{B} \wedge \overline{C})} \\
&\stackrel{1\times\text{ erw. De Morgan}(+2\times\text{rekursiv})}{=} \overline{\overline{D}} \wedge (\overline{C} \vee \overline{\overline{A}}) \wedge (\overline{\overline{B}} \vee \overline{\overline{C}}) \\
&\stackrel{4\times\text{ Involution}}{=} D \wedge (\overline{C} \vee A) \wedge (B \vee C) \\
&\stackrel{2\times\text{ Kommutativität}}{=} D \wedge ((B \vee C) \wedge (A \vee \overline{C})) \\
&\stackrel{\text{c)}}{=} D \wedge (B\overline{C} \vee AC) \\
&\stackrel{2\times\text{Neutralität}}{=} D \wedge ((B\overline{C} \wedge 1) \vee (AC \wedge 1)) \\
&\stackrel{2\times\text{Komplementarität}}{=} D \wedge ((B\overline{C} \wedge (A \vee \overline{A})) \vee (AC \wedge (B \vee \overline{B}))) \\
&\stackrel{2\times\text{Distributivität}}{=} D \wedge ((B\overline{C}A \vee B\overline{C}\overline{A}) \vee (ACB \vee AC\overline{B})) \\
&\stackrel{\text{erw. Distributivität}}{=} DB\overline{C}A \vee DB\overline{C}\overline{A} \vee DACB \vee DAC\overline{B} \\
&\stackrel{4\times\text{erw. Kommutativität}}{=} AB\overline{C}D \vee \overline{A}B\overline{C}D \vee ABCD \vee A\overline{B}CD
\end{aligned}$$