Fakultät IV Elektrotechnik und Informatik
Informatik 1 WS 2001/02, Jähnichen / Flik / Lechtchinsky / Neunast / Tutoren

Aufgabenblatt 9

letzte Aktualisierung: 31. January, 13:36

Ausgabe: 18.01.2002
Abgabe: 28./29.01.2002 *Prozent:* 100

Thema: *Zahlendarstellung; Boolesche Algebra*

1. Aufgabe (20 Prozent): Festpunktzahlen

1.1. Vergleich (Tut) Wie lassen sich die (ganzen) Dualzahlen in die Darstellung der Festpunktzahlen einordnen? Welche Zweierpotenzen haben die einzelnen Stellen? Worin liegt der Vorteil bei der Verwendung von echt gebrochenen Zahlen (Komma ganz links) gegenüber ganzen Dualzahlen (Komma ganz rechts)?

Lösung:

(Ganze) Dualzahlen besitzen in der normalen Darstellung kein Komma, man könnte sich jeweils ein Komma ganz rechts neben dem niederwertigsten Bit vorstellen (Analogie: $5 = 5,0000...$*). Festkommazahlen zeichnen sich dadurch aus, daß sie per Definition das Komma in die Zahlenmitte verschieben.*

In völliger Analogie zu Dezimalzahlen besitzen die Bits rechts vom Komma negative Exponenten: $2.5_{10} = 2 \cdot 10^0 + 5 \cdot 10^{-1} = 10.1_2 = 1 \cdot 2^1 + 0 \cdot 2^0 + 1 \cdot 2^{-1}$

Echt gebrochenen Zahlen können bei der Mulitpliktation keinen Überlauf provozieren, da sie immer kleiner als 1 *sind und damit das Produkt auch immer unter* 1 *liegt.*

1.2. Umwandlung (Tut) Wandelt die Zahlen $7,825_{10}$ und 0101.1001_2 in die jeweils andere Darstellung um. Verwendet für die Festpunktzahlen ein Zahlenformat mit vier Vor- und vier Nachkommastellen in 2-Komplementdarstellung.

Lösung:

$7,825 = 7 + 0,825 = 0111_2 + 0,825_{10} =$

$$\begin{array}{rcccccl|rcccccl}
0,825 & \cdot & 2 & = & 0,65 & + & 1; & 0,65 & \cdot & 2 & = & 0,3 & + & 1 \\
0,3 & \cdot & 2 & = & 0,6 & + & 0; & 0,6 & \cdot & 2 & = & 0,2 & + & 1 \\
0,2 & \cdot & 2 & = & 0,4 & + & 0; & 0,4 & \cdot & 2 & = & 0,8 & + & 0 \\
0,8 & \cdot & 2 & = & 0,6 & + & 1 & & & & & & &
\end{array}$$

$= 0111.110\overline{1001}_2 = 0111.1101_2$

$0101.1001_2 = 5_{10} + 0.1001_2 = 5_{10} + 1 \cdot 2^{-1} + 1 \cdot 2^{-4} = 5 + 0,5 + 0,0625 = 5,5625$

1.3. Umwandlung (10 Prozent) Wandelt die Zahlen $-5,26_{10}$, $1,25_{10}$, 0101.0101_2 und 1010.1010_2 in die jeweils andere Darstellung um, das Zahlenformat ist dasselbe wie bei Aufgabe 1.2. Sind alle Zahlen ohne Genauigkeitsverlust umzuwandeln? Wenn nein, wie groß ist der prozentuale Fehler?

Lösung:

$-5,26_{10} = (-1) * 5,26_{10} = (-1) * (0101_2 + 0,26_{10})$

$$\begin{array}{rcccccl}
0,26 & \cdot & 2 & = & 0,52 & + & 0 \\
0,52 & \cdot & 2 & = & 0,04 & + & 1 \\
0,04 & \cdot & 2 & = & 0,08 & + & 0 \quad \approx 0.0100_2 (= 0,25_{10}!) \\
0,08 & \cdot & 2 & = & 0,16 & + & 0 \\
0,16 & \cdot & 2 & = & 0,32 & + & 0
\end{array}$$

Nun muss die Zahl noch als 2-Komplement dargestellt werden. Dazu werden wie bei ganzen Zahlen alle Bits invertiert und dann an der Stelle mit dem niedrigstwertigsten Bit eine Eins addiert. Also wird aus $(-1) * 0101.0100 = 1010.1011 + 0000.0001 = 1010.1100$. *Eine Rückumwandlung (zuerst 2-Komplement, dann Dekodierung) bringt als Ergebnis* -5.25_{10}. *Also beträgt der prozentuale Fehler hierbei* $E_r = |(-5,25) - (-5,26)|/|(-5,26)| = 0,19\%$. *Die komplette Kodierung der Zahl lautet übrigens* $(-1) * 0111.01\overline{00001010001111101100}$.

$1,25_{10}$ *ist einfach* $1 \cdot 2^0 + 1 \cdot 2^{-2}$, *also* 0001.0100. *Die rationale Darstellung lautet* $\frac{5}{4}$, *da vier eine Zweierpotenz ist, ist die Zahl ohne Fehler nichtperiodisch darstellbar.*

$0101.0101_2 = 5_{10} + 2^{-2} + 2^{-4} = 5,3125_{10}$

$1010.1010_2 = -8_{10} + 2_{10} + 2^{-1} + 2^{-3} = -5,375_{10}$

1.4. Rechnen (Tut) Berechnet in binärer Form $0010.0110_2 \cdot 0100.0100$ und $0101.1111 + 0011.1010$. Verwendet dazu ein Ergebnisformat mit 4 Vor- und 4 Nachkommastellen.

Lösung:

0	0	1	0.	0	1	1	0	$*$	0	1	0	0.	0	1	0	0
							0	0	1	0	0	1	1	0		
+			0	0	1	0	0	1	1	0						
			0	0	1	0	1	0.	0	0	0	1	1	0	0	0

	0101.	1111
+	0011.	1010
	1001.	1001

1.5. Rechnen (10 Prozent) Berechnet in binärer Form $3,625 \cdot 1,4$ und $7,65 + 4,11$. Verwendet dazu ein Zahlenformat mit 4 Vorkomma und 8 Nachkommastellen.

Lösung:

$3,625_{10} = 0011_2 + 0,5_{10} + 0,125_{10} = 0011.10100000_2$.
$1,4_{10} = 0001_2 + 0,4_{10} =$

$$\left.\begin{array}{rcccccl}
0.4 & \cdot & 2 & = & 0.8 & + & 0 \\
0.8 & \cdot & 2 & = & 0.6 & + & 1 \\
0.6 & \cdot & 2 & = & 0.2 & + & 1 \\
0.2 & \cdot & 2 & = & 0.4 & + & 0
\end{array}\right. = 0001.\overline{0110} \approx 0001.01100110$$

0	0	0	1.	0	1	1	0	0	1	1	0	$*$	0	0	1	1.	1	0	1	
								0	0	0	1	0	1	1	0	0	1	1	0	
						0	0	0	1	0	1	1	0	0	1	1	0			
					0	0	0	1	0	1	1	0	0	1	1	0				
				0	0	0	1	0	1	1	0	0	1	1	0					
				0	0	1	0	1.	0	0	0	1	0	0	0	1	1	1	$0 \approx$	
					0	1	0	1.	0	0	0	1	0	0	0	1				$= 5,06640625_{10}$

$3,625_{10} \cdot 1,4_{10} = 5,075_{10} \approx 101.00010011001_2$

Hinweis: *Möchte man negative Zahlen multiplizieren, so kann auf die 2-Komplement-Bildung verzichtet werden. Bei der Multiplikation geht das Vorzeichen nicht wirklich in die Rechnung ein, da man die* (-1) *als Faktor von Rechnung und Ergebnis ausklammern kann, also gilt z.B.:* $(-1) * 1,4 * 3,625 = (-1) * 5,075$.

Man könnte also zuerst das Vorzeichen bestimmen und dann das Ergebnis mit positiven Zahlen berechnen. Zum Schluss kann man das Vorzeichen wieder dem Ergebnis zuordnen.

$7,65_{10} = 0111_2 + 0,65_{10} =$ $\qquad$ $4,11_{10} = 0100_2 + 0,11_{10} =$

0.65	·	2	=	0.30	+	1	0.11	·	2	=	0.22	+	0
0.30	·	2	=	0.60	+	0	0.22	·	2	=	0.44	+	0
0.60	·	2	=	0.20	+	1	0.44	·	2	=	0.88	+	0
0.20	·	2	=	0.40	+	0	0.88	·	2	=	0.76	+	1
0.40	·	2	=	0.80	+	0	0.76	·	2	=	0.52	+	1
0.80	·	2	=	0.60	+	1	0.52	·	2	=	0.04	+	1
0.60	·	2	=	0.20	+	1	0.04	·	2	=	0.08	+	0
0.20	·	2	=	0.40	+	0	0.08	·	2	=	0.16	+	0
0.40	·	2	=	0.80	+	0	0.16	·	2	=	0.32	+	0

$$\begin{array}{rll} & 0111. & 10100110 \\ + & 0100. & 00011100 \\ \hline & 1011. & 11000010 = 11,7578125_{10} \end{array}$$

Wenn man das oben angegebe Resultat aber in 2-Komplement-Darstellung interpretiert, so ist das Ergebnis allerdings falsch, da eine Umwandlung in diesem Fall $-4,2421875_{10}$ *ergibt. Der Grund dafür ist die aufgetretene Bereichsüberschreitung.*

2. Aufgabe (30 Prozent): Gleitpunktzahlen

Um die Aufgabe nicht unnötig zu erschweren, verwenden wir hier das aus der Vorlesung (Seite 2-37) bekannte Format mit 4 Exponentenbits, 11 Mantissenbits und einem Bias von 7.

2.1. Umwandlung (Tut) Wandelt die Zahlen $5,65625_{10}$ und $-1.01100110100_2 \cdot 2^{-1}$ in das jeweils andere Format um. Wie sieht das Bitmuster des Gleitpunktformats aus?
Was gibt es für Sonderfälle und wann werden sie benutzt? Welches ist die kleinste, welches die größte darstellbare Zahl?

Lösung:

$5,65625_{10} = 101_2 + 0,65625_{10} = 101_2 +$

0.65625	·	2	=	0.31250	+	1	
0.31250	·	2	=	0.62500	+	0	
0.62500	·	2	=	0.25000	+	1	$= 0.10101_2$
0.25000	·	2	=	0.50000	+	0	
0.50000	·	2	=	0.00000	+	1	

Also lautet die Zahl in Festpunktdarstellung 101.10101_2*. Zur Konvertierung in die Gleitpunktdarstellung verschiebt man das Ergebnis solange, bis links vom Komma nur noch eine Eins steht, hier also um 2 Stellen nach links. Demnach lautet die Zahl also* $1.0110101 \cdot 2^2$*. Der Exponent wird nun zum Bias (hier: 7) addiert und als Dualzahl in die Exponentenbits kodiert:* $2 + 7 = 9 = 1001_2$*. Das Vorzeichenbit ist hier 0. Bei Kodieren der Mantisse wird die führende Eins nicht mitgespeichert:*
$5,65625_{10} = 1.0110101_2 \cdot 2^2 \Rightarrow 0'1001'01101010000 = 4B50_{16}$.

$-1.01100110100_2 \cdot 2^{-1} = 1'0110'01100110100$. *Der Exponent* $6 (= 0110_2)$ *entsteht durch Addition des tatsächlichen Exponenten mit dem Bias* $(-1 + 7 = 6)$.
$-1.01100110100_2 \cdot 2^{-1} = -(1 + \frac{1}{4} + \frac{1}{8} + \frac{1}{64} + \frac{1}{128} + \frac{1}{512}) = \frac{-(512+128+64+8+4+1)}{512} = \frac{-717}{512} =$
$-1,400390625_{10} \cdot 2^{-1} = -0,7001953125_{10}$.

Im IEEE754-Standard existieren zwei Formate, eines mit einfaches Genauigkeit (23 bit Mantisse, 8 bit Exponent) und eines mit doppelter Genauigkeit (52 bit Mantisse und 11 bit Exponent). Der Exponent wird so gestaltet, daß durch Addition mit einem Bias immer eine positive Zahl entsteht, deren duale Entsprechung dann in die Exponentenbits geschrieben wird.

Es existieren mehrere Sonderfälle, die bestimmte Ausnahmebedingungen abdecken und durch bestimmte Bitmuster im Exponenten und/oder in der Mantisse kodiert werden, dazu zählen positive und negative Unendlichkeit, Nichtzahlen (NaNs), der Wert 0 *sowie eine unnormalisierte Variante von Zahlen, bei der die implizite führende Eins zur Null wird und der Exponent den kleinst möglichsten Wert annimmt. Damit kann man sehr kleine Zahlen darstellen.*

Weiterführende Informationen dazu findet man im Online-Skript sowie in den Vorlesungsfolien.

2.2. Umwandlung (10 Prozent) Wandelt die Zahlen $1,5678_{10}$, $14,37_{10}$, 0101110001001011 und 1000000010010010 in die jeweils andere Darstellung um. Zeigt beim Gleitpunktformat sowohl das Bitmuster als auch die mathematische Darstellung.

Lösung:

$1,5678_{10} = 01_2 + 0,5678_{10}$

0.5678	·	2	=	0.1356	+	1;	0.1356	·	2	=	0.2712	+	0
0.2712	·	2	=	0.5424	+	0;	0.5424	·	2	=	0.0848	+	1
0.0848	·	2	=	0.1696	+	0;	0.1696	·	2	=	0.3392	+	0
0.3392	·	2	=	0.6784	+	0;	0.6784	·	2	=	0.3568	+	1
0.3568	·	2	=	0.7136	+	0;	0.7136	·	2	=	0.4272	+	1
0.4272	·	2	=	0.8544	+	0;	0.8544	·	2	=	0.7088	+	1

$= 1.10010001010 \cdot 2^0 = 0'0111'10010001010$

$14,37_{10} = 01110_2 + 0,37_{10}$

0.37	·	2	=	0.74	+	0;	0.74	·	2	=	0.48	+	1
0.48	·	2	=	0.96	+	0;	0.96	·	2	=	0.92	+	1
0.92	·	2	=	0.84	+	1;	0.84	·	2	=	0.68	+	1
0.68	·	2	=	0.36	+	1;	0.36	·	2	=	0.72	+	0
0.72	·	2	=	0.44	+	1							

$= 1.11001011110 \cdot 2^3 = 0'1010'11001011110$

$0'1011'10001001011 = 1.10001001011 \cdot 2^{11-7} = 11000.1001011 = 24,5859375_{10}$

$1'0000'00010010010 = -0.00010010010 \cdot 2^{1-7} = -0,001113891601563_{10}$
Die letzte Zahl ist eine negative (1. Bit gesetzt) unnormalisierte (Exponent ist Null) Zahl. Deshalb ist die führende Ziffer eine Null statt einer Eins, der Exponent wird dabei so gesetzt, als ob eine 1 kodiert wäre, also hier zu 2^{-6}.

2.3. Addition (Tut) Wandelt die Zahlen $46,753$ und $3,247$ in Gleitkommadarstellung um und addiert sie anschließend. Was fällt euch beim Ergebnis auf? Warum ist das so?

Lösung:

$46,753_{10} = 101110_2 + 0,753_{10}$ $\qquad$ $3,247_{10} = 11_2 + 0,247_{10}$

0.753	·	2	=	0.506	+	1	0.247	·	2	=	0.494	+	0
0.506	·	2	=	0.012	+	1	0.494	·	2	=	0.988	+	0
0.012	·	2	=	0.024	+	0	0.988	·	2	=	0.976	+	1
0.024	·	2	=	0.048	+	0	0.976	·	2	=	0.952	+	1
0.048	·	2	=	0.096	+	0	0.952	·	2	=	0.904	+	1
0.096	·	2	=	0.192	+	0	0.904	·	2	=	0.808	+	1
0.192	·	2	=	0.384	+	0	0.808	·	2	=	0.616	+	1
0.384	·	2	=	0.768	+	0	0.616	·	2	=	0.232	+	1
0.768	·	2	=	0.536	+	1	0.232	·	2	=	0.464	+	0
0.536	·	2	=	0.072	+	1	0.464	·	2	=	0.928	+	0
0.072	·	2	=	0.144	+	0	0.928	·	2	=	0.856	+	1
0.144	·	2	=	0.288	+	0	0.856	·	2	=	0.712	+	1

46,753 ergibt also eine Festkommazahl von 101110.1100000011000.... *Die führende Eins plus 11 Dualstellen führen zu einer unkodierten Zahl von* $1.01110110000 \cdot 2^5$.
3,247 ist in Festkommadarstellung 11.001111110011101.... *Das ergibt eine Darstellung von* $1.10011111100 \cdot 2^1$. *Zum Addieren muß man beide Zahlen normieren, d.h. sie müssen den selben Exponenten aufweisen. Dazu passt man den kleineren Exponenten an den größeren an, hier also* $1.10011111100 \cdot 2^1 = 0.000110011111100 \cdot 2^5$. Nun kann man beide Zahlen addieren:

	1.	01110110000	$\cdot 2^5$
+	0.	00011001111	$\cdot 2^5$
	1.	10001111111	$\cdot 2^5$

Das Ergebnis lautet in Dezimalnotation 49,984375, es sollte aber offensichtlich 50,0 ergeben. Fehler kommen hier neben Genauigkeitsverlusten beim Umwandeln durch das Normieren zustande, da man dabei einige Nachkommastellen (hier: 4) verliert. Der Fehler wird umso größer, je mehr die Zahlen auseinander liegen.

2.4. Addition (10 Prozent) Addiert die Zahlen 12,825 und 5,9375 sowie 0,000075 und 64,569 in Gleitpunktnotation miteinander. Wandelt die Ergebnisse wieder in Dezimalnotation um und vergleicht sie mit den tatsächlichen. Bei welchen Zahlenkombinationen treten starke Genauigkeitsverluste auf?

Lösung:

$12,825_{10} = 1.10011010011 \cdot 2^3$
$5,9375_{10} = 1.01111100000 \cdot 2^2 = 0.10111110000 \cdot 2^3$

	1.	10011010011	$\cdot 2^3$
+	0.	10111110000	$\cdot 2^3$
	10.	01011000011	$\cdot 2^3 = 1.00101100001 \cdot 2^4$

$10010.1100001_2 = 18,7578125_{10}$
$12,825 * 5,9375 = 18,7625_{10} = 1.00101100001 \cdot 2^4$
Das Ergebnis stimmt offensichtlich im Rahmen der Genauigkeit unserer Gleitpunktzahlen genau.
$0,000075_{10} = 0_2 + 0,000075_{10} =$

0.000075	·	2	=	0.000150	+	0;	0.000150	·	2	=	0.000300	+	0
0.000300	·	2	=	0.000600	+	0;	0.000600	·	2	=	0.001200	+	0
0.001200	·	2	=	0.002400	+	0;	0.002400	·	2	=	0.004800	+	0
0.004800	·	2	=	0.009600	+	0;	0.009600	·	2	=	0.019200	+	0
0.019200	·	2	=	0.038400	+	0;	0.038400	·	2	=	0.076800	+	0
0.076800	·	2	=	0.153600	+	0;	0.153600	·	2	=	0.307200	+	0
0.307200	·	2	=	0.614400	+	0;	0.614400	·	2	=	0.228800	+	1
0.228800	·	2	=	0.457600	+	0;	0.457600	·	2	=	0.915200	+	0
0.915200	·	2	=	0.830400	+	1;	0.830400	·	2	=	0.660800	+	1
0.660800	·	2	=	0.321600	+	1;	0.321600	·	2	=	0.643200	+	0

$= 0.00000000000001001110$
Um die Zahl zu normalisieren, müsste man als Exponenten -14 *verwenden, was aber bei einem Bias von* $+7$ *keine positive Zahl ergibt und damit nicht als Exponent kodiert werden kann. Also muss diese Zahl als unnormalisierte Zahl gespeichert werden, d.h. der Exponent wird zu* -6 *und auf die führende Eins wird verzichtet. Die Zahl lautet dann also* $0.00000001001 \cdot 2^{-6}$.
$64,569_{10} = 1000000_2 + 0,569_{10} =$

0.569	·	2	=	0.138	+	1;	0.138	·	2	=	0.276	+	0
0.276	·	2	=	0.552	+	0;	0.552	·	2	=	0.104	+	1
0.104	·	2	=	0.208	+	0;	0.208	·	2	=	0.416	+	0
0.416	·	2	=	0.832	+	0;	0.832	·	2	=	0.664	+	1
0.664	·	2	=	0.328	+	1;	0.328	·	2	=	0.656	+	0

$= 1000000.10010 = 1.00000010010 \cdot 2^6$.
Die übliche Vorgehensweise bei der Addition wäre jetzt, die Zahlen auf einen gemeinsamen Exponenten zu bringen. Dazu müsste der kleinere Exponent dem größeren angepasst werden, was in diesem Fall eine Verschiebung des Kommas um $6 - (-6) = 12$ *Stellen nach links bedeuten würde. Offensichtlich bleiben dann von der kleinen Zahl nur noch Nullen übrig, so daß die Addition nicht tatsächlich durchgeführt würde. Das Ergebnis lautet also* $1.00000010010_2 = 64,5625_{10}$.

Diese Problem ist typisch für die Addition von Gleitpunktzahlen. Problematisch wird es z.B. bei der Simulation von Differentialgleichungen, wo häufig sehr kleine Werte (die Schrittweite) auf meist größere Werte addiert werden. Wird dabei ein Summand zu Null, so verfälscht dies die Rechnung erheblich bzw. macht das Ergebnis untauglich.

2.5. Multiplikation (Tut) Multipliziert die Zahlen $1.1001 \cdot 2^3$ und $1.00010011001 \cdot 2^2$ in Gleitpunktnotation miteinander. Was fällt beim Überprüfen des Ergebnisses auf?

Lösung:

Zuerst multipliziert man ganz normal die Mantissen, danach die Exponenten und setzt das Ergebnis wieder zusammen.

1.	0	0	0	1	0	0	1	1	0	0	1	*	1.	1	0	0	1
						1	0	0	0	1	0	0	1	1	0	0	1
			1	0	0	0	1	0	0	1	1	0	0	1			
+		1	0	0	0	1	0	0	1	1	0	0	1				
		1.	1	0	1	0	1	1	0	1	1	1	1	0	0	0	1

$2^3 \cdot 2^2 = 2^{3+2} = 2^5$
Also lautet das Ergebnis $1.10101101111 \cdot 2^5$ $(= 53,734375_{10})$. *Die Genauigkeit leidet dabei etwas, aber es gibt hierbei weniger starke Abweichungen als bei der Addition.*

In Dezimalnotation ergibt sich $12,5 \cdot 4,3 = 53,75$, *also müsste das Ergebnis in Festpunktdarstellung exakt* 110101.11_2 *lauten. Hier treten also Fehler auf, die sich bereits durch das Umwandeln der Zahlen ergeben und dann in der Rechnung fortpflanzen.*

2.6. Multiplikation (10 Prozent) Multipliziert die Zahlen 9,2305 und 2,5 sowie 0,005 und 17,34 in Gleitkommanotation miteinander. Wann treten bei der Multiplikation Genauigkeitsprobleme auf?

Lösung:

$9,2305_{10} = 1001_2 + 0,2305_{10} =$

0.2305	·	2	=	0.461	+	0;	0.461	·	2	=	0.922	+	0
0.922	·	2	=	0.844	+	1;	0.844	·	2	=	0.688	+	1
0.688	·	2	=	0.376	+	1;	0.376	·	2	=	0.752	+	0
0.752	·	2	=	0.504	+	1;	0.504	·	2	=	0.008	+	1
0.008	·	2	=	0.016	+	0;	0.016	·	2	=	0.032	+	0

$= 1001.0011101100_2 = 1.00100111011 \cdot 2^3$

$2,5_{10} = 10_2 + 1 \cdot 2^{-1} = 10.1_2 = 1.01000000000 \cdot 2^1$

1.	0	0	1	0	0	1	1	1	0	1	1	*	1.	0	1
				1	0	0	1	0	0	1	1	1	0	1	1
+		1	0	0	1	0	0	1	1	1	0	1	1		
		1.	0	1	1	1	0	0	0	1	0	0	1	1	1

$= 1.01110001001 \cdot 2^{3+1=4} = 10111.0001001_2 = 23,0703125_{10}$

$9,2305_{10} * 2,5_{10} = 23,07625_{10} \approx 10111.0001001_2$

$0,005 = 0.00000001010001111_2 = 0.01010001111_2 \cdot 2^{-6}$

$17,34 = 10001.01010111_2 = 1.00010101011_2 \cdot 2^4$

1.	0	0	0	1	0	1	0	1	0	1	1	*	0.	0	1	0	1	0	0	0	1	1	1	1
													1	0	0	0	1	0	1	0	1	0	1	1
												1	0	0	0	1	0	1	0	1	0	1	1	
											1	0	0	0	1	0	1	0	1	0	1	1		
										1	0	0	0	1	0	1	0	1	0	1	1			
						1	0	0	0	1	0	1	0	1	0	1	1							
				1	0	0	0	1	0	1	0	1	0	1	1									
		0.	0	1	0	1	1	0	0	0	1	0	1	1	0	1	1	0	0	0	0	1	0	1

$= 1.01100010110 \cdot 2^{(-6)+4-2=-4} = 0.000101100010110_2 = 0,08660888671875$

$0,005_{10} * 17,34_{10} = 0,08670_{10} \approx 0.000101100011000_2$

Offensichtlich sind die Ergebnisse von Multiplikationen im Gleitpunktbereich wesentlich genauer, da hier tatsächlich signifikante Stellen verrechnet werden und die Größenordnungen getrennt durch die Addition der Exponenten behandelt werden.

3. Aufgabe (15 Prozent): Boolesche Ausdrücke

Übersicht der verwendeten Axiome und Regeln:

Axiome: Die folgenden 10 Gleichungen sind grundlegend; sie definieren die **Boolesche Algebra**.

$$a \cdot b = b \cdot a \tag{1}$$
$$a + b = b + a \tag{2}$$
$$(a \cdot b) \cdot c = a \cdot (b \cdot c) \tag{3}$$
$$(a + b) + c = a + (b + c) \tag{4}$$
$$(a + b) \cdot c = a \cdot c + b \cdot c \tag{5}$$
$$(a \cdot b) + c = (a + c) \cdot (b + c) \tag{6}$$
$$a \cdot 1 = a \tag{7}$$
$$a + 0 = a \tag{8}$$
$$a \cdot \overline{a} = 0 \tag{9}$$
$$a + \overline{a} = 1 \tag{10}$$

Sätze zur Vereinfachung boolescher Ausdrücke:

$$a \cdot a = a \tag{11}$$
$$a + a = a \tag{12}$$
$$a \cdot 0 = 0 \tag{13}$$
$$a + 1 = 1 \tag{14}$$
$$a + a \cdot b = a \tag{15}$$
$$a \cdot (a + b) = a \tag{16}$$
$$a + \overline{a} \cdot b = a + b \tag{17}$$
$$a \cdot (\overline{a} + b) = a \cdot b \tag{18}$$

Sätze zur Bildung der Negation eines Ausdrucks:

$$\overline{\overline{a}} = a \tag{19}$$
$$\overline{a} \cdot \overline{b} = \overline{a + b} \tag{20}$$
$$\overline{a} + \overline{b} = \overline{a \cdot b} \tag{21}$$

Sätze zur Anwendung auf elektronische Schaltungen:

$$\overline{a} = \overline{a \cdot a} = \overline{a + a} \tag{29}$$
$$a \cdot b = \overline{\overline{a \cdot b} \cdot \overline{a \cdot b}} = \overline{\overline{a + a} + \overline{b + b}} \tag{30}$$
$$a + b = \overline{\overline{a \cdot a} \cdot \overline{b \cdot b}} = \overline{\overline{a + b} + \overline{a + b}} \tag{31}$$

3.1. Axiome und Sätze (Tut) Beweist Regel 13 aus dem Skript (Seite 3-11) mit Hilfe einer Wertetabelle und durch Herleitung aus den Axiomen (Skript, Seite 3-10)

Lösung:

Wertetabelle:

a	***0***	$a \cdot 0$
0	*0*	*0*
1	*0*	*0*

Herleitung:

$$\begin{array}{lll} & a \cdot 0 & \textit{(8)} \\ = & a \cdot 0 + 0 & \textit{(9), (2)} \\ = & a \cdot 0 + (a \cdot \overline{a}) & \textit{(5)} \\ = & a \cdot (0 + \overline{a}) & \textit{(8), (2)} \\ = & a \cdot \overline{a} & \textit{(9)} \\ = & 0 & \textit{fertig} \end{array}$$

3.2. Axiome und Sätze (15 Prozent) Beweist Regel 14 aus dem Skript (Seite 3-11) mit Hilfe einer Wertetabelle und durch Herleitung aus den Axiomen (Skript, Seite 3-10)

Lösung:

Wertetabelle:

a	1	a + 1
0	1	1
1	1	1

Herleitung:

$$
\begin{array}{lll}
 & a+1 & \textit{(7)} \\
= & (a+1)\cdot 1 & \textit{(10)} \\
= & (a+1)\cdot(a+\overline{a}) & \textit{(6), (2)} \\
= & a+(1\cdot\overline{a}) & \textit{(7), (2)} \\
= & a+\overline{a} & \textit{(10)} \\
= & 1 & \textit{fertig}
\end{array}
$$

4. Aufgabe (35 Prozent): Umwandlung von Funktionen

4.1. Vereinfachung von komplexen Ausdrücken (20 Prozent) Vereinfacht die folgenden Funktionen, so dass in den endgültigen Ausdrücken nur einfache Negationen und keine Klammern mehr vorkommen. Gebt bei jeder Umformung die Regel aus dem Skript mit an.

$$
\begin{array}{ll}
a) & (a+\overline{c+\overline{a}}+\overline{b})\cdot(a+\overline{a}\cdot c) \\
b) & \overline{\overline{a}+a\cdot b}\cdot\overline{\overline{a}\cdot\overline{\overline{a}\cdot c}}\cdot(a+\overline{b})
\end{array}
$$

Hinweis: Ein Beispiel zur Vereinfachung von Funktionen findet ihr im Skript auf Seite 3-12.

Lösung:

a)

$$
\begin{array}{lll}
 & (a+\overline{c+\overline{a}}+\overline{b})\cdot(a+\overline{a}\cdot c) & \textit{(17)} \\
= & (a+\overline{c+\overline{a}}+\overline{b})\cdot(a+c) & \textit{(20)} \\
= & (a+(\overline{c}\cdot\overline{\overline{a}})+\overline{b})\cdot(a+c) & \textit{(19)} \\
= & (a+(\overline{c}\cdot a)+\overline{b})\cdot(a+c) & \textit{(15)} \\
= & (a+\overline{b})\cdot(a+c) & \textit{(6)} \\
= & a+\overline{b}\cdot c & \textit{fertig}
\end{array}
$$

b)

$$
\begin{array}{lll}
 & \overline{\overline{a}+a\cdot b}\cdot\overline{\overline{a}\cdot\overline{\overline{a}\cdot c}}\cdot(a+\overline{b}) & \textit{(21)} \\
= & \overline{\overline{a}+a\cdot b}\cdot(\overline{\overline{a}}+\overline{\overline{\overline{a}\cdot b}})\cdot(a+\overline{b}) & \textit{(19)} \\
= & \overline{\overline{a}+a\cdot b}\cdot(a+(\overline{a}\cdot b)\cdot(a+\overline{b}) & \textit{(20), (17)} \\
= & \overline{\overline{a}}\cdot\overline{a\cdot b}\cdot(a+b)\cdot(a+\overline{b}) & \textit{(19), (6)} \\
= & a\cdot\overline{a\cdot b}\cdot(a+b\cdot\overline{b}) & \textit{(21), (9)} \\
= & a\cdot(\overline{a}+\overline{b})\cdot(a+0) & \textit{(8)} \\
= & a\cdot(\overline{a}+\overline{b})\cdot a & \textit{(1), (3), (11)} \\
= & (\overline{a}+\overline{b})\cdot a & \textit{(5)} \\
= & \overline{a}\cdot a+\overline{b}\cdot a & \textit{(9)} \\
= & 0+\overline{b}\cdot a & \textit{(8)} \\
= & a\cdot\overline{b} & \textit{fertig}
\end{array}
$$

4.2. NAND (Tut) Drückt folgende Funktion ausschließlich mit NAND Funktionen aus

$$(a+b)\cdot c+\overline{d}$$

Lösung:

$$
\begin{array}{ll}
 & (a+b)\cdot c+\overline{d} \\
= & \overline{\overline{a}\cdot\overline{b}}\cdot c+\overline{d} \\
= & \overline{\overline{\overline{\overline{a}\cdot\overline{b}}\cdot c}}+\overline{d} \\
= & \overline{\overline{\overline{\overline{a}\cdot\overline{b}}\cdot c}\cdot d}
\end{array}
$$

4.3. NOR (15 Prozent) Drückt folgende Funktionen ausschließlich mit NOR Funktionen aus:

$$
\begin{array}{ll}
a) & (a+b)\cdot c+\overline{d} \\
b) & \overline{a}\cdot\overline{c}+\overline{b}(c+d)
\end{array}
$$

Lösung:

a)

$$
\begin{array}{ll}
 & (a+b)\cdot c+\overline{d} \\
= & \overline{\overline{a+b}}\cdot c+\overline{d} \\
= & \overline{\overline{a+c}+\overline{b}}+\overline{d} \\
= & \overline{\overline{\overline{\overline{a+b}+\overline{c}}+\overline{d}}}
\end{array}
$$

$b)$

$$
\begin{aligned}
& \overline{a} \cdot \overline{c} + \overline{b}(c+d) \\
= \quad & \overline{a+c} + \overline{b}(c+d) \\
= \quad & \overline{a+c} + \overline{b}(\overline{\overline{c+d}}) \\
= \quad & \overline{a+c} + \overline{b + \overline{c+d}} \\
= \quad & \overline{\overline{\overline{a+c} + \overline{b + \overline{c+d}}}}
\end{aligned}
$$