1. Ein Wort aus dem Zeichenvorrat $\{0,1\}$ ist definiert als $(0^m 1^n)^m (0^n 1^m)^n$. Ermitteln Sie das Wort für alle $0 \le m \le 2$ und $0 \le n \le 2$.

2. Finden Sie (analog zum vorigen Beispiel) möglichst kurze Definitionen folgender Wörter:

   01001001010101010
   0010010101011111
   0101101001011010
   001001000100101

3. Ermitteln Sie folgende Wörtermengen und jeweils die Anzahl der Wörter.

   $\{a \circ b \circ c \mid a = 1^m \wedge b = 0^n \wedge c = 1^{5-m-n}\}$
   $\{a \in \{0,1\}^5 \mid a \in \{0,1\}^m \circ 010 \circ \{0,1\}^n \wedge m, n \ge 0\}$
   $\{01, 10\}^3 \cup \{010, 101\}^2$
   $\{0^m 1^n \mid n = \lfloor m/3 \rfloor \wedge 4 \le m + n \le 9\}$

4. Finden Sie intensionale Definitionen folgender Wörtermengen:

   {□□□□□, △□□□△, □△□△□, △△□△△, □□△□□, △□△□△, □△△△□, △△△△△}
   $\{00011, 00101, 01001, 10001, 00110, 01010, 10010, 01100, 10100, 11000\}$
   $\{aba, abc, aca, acb, bab, bac, bca, bcb, cab, cac, cba, cbc\}$

5. Eine Wörtermenge kann auch rekursiv definiert werden. Der Zeichenvorrat sei $\{0, 1, \_\}$. S sei die kleinste Wörtermenge, die folgende Regeln erfüllt:

   (a) $\_1 \in S$
   (b) $1a0\_1a1 \in S$ für alle $a \in \{0,1\}^*$
   (c) $a1\_b0 \in S$, wenn $a\_b \in S$

   Ermitteln Sie die kürzesten acht Wörter aus S.

6. Gegeben sind die Zeichenvorräte A, B. Berechnen Sie allgemein:

   (a) die Anzahl der möglichen Codewörter $|A^n|$ der Länge n, sowie $|B^m|$.

   (b) die Anzahl der möglichen Codewörter mit *maximaler* Länge n: $|A^0 \cup \ldots \cup A^n|$.

   (c) die Anzahl der möglichen Codewörter $|(A \cup B)^{m+n}|$ mit gemeinsamem Zeichenvorrat.

   (d) die Anzahl der konkatenierten Codewörter $|A^n \circ B^m|$.

   Demonstrieren Sie die Ergebnisse am Beispiel $A = \{a, b, c\}$, $n = 2$, $B = \{1, 2\}$, $m = 3$.

7. Die Levenshteindistanz ist eine Verallgemeinerung der Hammingdistanz. Während bei der Hammingdistanz die Anzahl der Schritte gezählt wird, mit denen man durch Austausch eines Zeichens das eine Wort in das andere verwandelt, sind bei der Levenshteindistanz auch das Einfügen und Entfernen eines Zeichens (an beliebiger Stelle) erlaubt. Zum Beispiel ist wegen 0101 → 101 → 1010 die Levenshteindistanz zwischen 0101 und 1010 gleich 2, während die Hammingdistanz gleich 4 ist. Bestimmen Sie die Hamming- und Levenshteindistanz der Wörter

$$01010101 \qquad \text{und} \qquad 10110010\,.$$

8. Für eine natürliche Zahl $n > 2$ ist folgende Abbildung gegeben: $c : \{2, 3, \ldots, n\} \longrightarrow \{0, 1, 2, 3\}^4$ mit $c(k) = abcd$, wobei $k = 7^a 5^b 3^c 2^d e$, und die natürliche Zahl e nicht durch 2, 3, 5, 7 teilbar ist. Also z. B. $c(6) = 0011$, weil $6 = 7^0 5^0 3^1 2^1 1$.

   Ist diese Abbildung für $n = 10$ ein decodierbarer Code?
   Gibt es ein n, für das der Code nicht decodierbar ist? Falls ja, wie lautet das kleinste solche n?
   Gibt es ein n, für das diese Konstruktion gar kein Code ist? Falls ja, wie lautet das kleinste solche n? Begründen Sie Ihre Antworten.

9. Folgende Codes a und $b : \mathbb{N} \longrightarrow \{0, 1\}^*$ sind rekursiv definiert:

   (a) $a(0) = 0$, $a(1) = 1$, und für $n > 1$: $a(n) = a(\lfloor \frac{n}{2} \rfloor) \circ \begin{cases} 1 & n \text{ ungerade} \\ 0 & n \text{ gerade} \end{cases}$

   (b) $b(0) = 0$, $b(1) = 1$, und für $n > 1$: $b(n) = b(\lfloor \log_2 n \rfloor) \circ b(n - 2^{\lfloor \log_2 n \rfloor})$

   wobei $\lfloor x \rfloor$ für Abrundung steht. Geben Sie die Codewörter für $n = 0, \ldots, 15$ an.

10. Die *Decodierung* eines Codes $c : \mathbb{N} \longrightarrow \{0, 1\}^*$ ist gegeben durch

$$c^{-1}(a_0 a_1 \ldots a_n) = \sum_{k=0}^{n} a_k F(k+2)\,,$$

   wobei die Fibonacci-Folge $F(k)$ gegeben ist durch $F(0) = 0$, $F(1) = 1$ und $F(k) = F(k-1) + F(k-2)$. So ist z.B. $c^{-1}(101) = F(2) + F(4) = 1 + 3 = 4$. Decodiere die Codewörter 1010101 und 01010101. Finde mindestens drei mögliche Codewörter für $c(73)$.

11. (a) Erzeugen Sie einen *nicht-zyklischen* Graycode für 10 Symbole $\{A, B, C, D, E, F, G, H, I, J\}$ und geben Sie die verwendete Transitionssequenz an.
    (b) Finden Sie einen *zyklischen* Graycode für obige Symbole und geben Sie auch hier die Transitionssequenz an.
    (c) Lässt sich ein zyklischer Graycode für 9 Symbole finden? Begründen Sie Ihre Antwort.

12. Ein Code besteht aus vier Zeichen aus dem Zeichenvorrat $\{a, b, c\}$, wobei gültige Codewörter jene sind, in denen a einmal, b einmal und c zweimal vorkommt. Wieviele mögliche und wieviele gültige Codewörter gibt es?

13. Gegeben ist ein Code mit 3 Bit plus 1 Paritybit. Berechne:

    (a) die Anzahl der möglichen, der korrekten und der inkorrekten Codewörter.
    (b) die Wahrscheinlichkeit, dass ein zufälliges Codewort inkorrekt ist unter der Annahme, dass alle Codewörter gleich wahrscheinlich sind.
    (c) die Wahrscheinlichkeit $P(F)$, dass im Codewort (inkl. Parity) mind. ein Bitfehler auftritt unter der Annahme, dass jedes Bit (unabhängig) eine Bitfehlerwahrscheinlichkeit von $p = 10^{-2}$ besitzt.
    (d) die Wahrscheinlichkeit $P(I)$, dass durch solche Bitfehler ein inkorrektes Codewort entsteht.
    (e) die Wahrscheinlichkeit, dass ein (gegebener) Bitfehler erkannt wird, d.h. die Wahrscheinlichkeit $P(I|F)$, dass ein Codewort inkorrekt ist unter der Voraussetzung, dass es fehlerbehaftet ist. Hinweis: Es gilt $P(I \wedge F) = P(I|F)P(F) = P(F|I)P(I)$ und $P(F|I)$ ist natürlich ...?

14. Der Hamming(7,4)-Code besitzt vier Datenbits d_1, d_2, d_3, d_4 und drei Paritybits p_1, p_2, p_3, wobei p_1 die Parity der Bits d_1, d_2, d_4 angibt, p_2 der Bits d_1, d_3, d_4 und p_3 der Bits d_2, d_3, d_4. Wie groß ist die minimale Hamming-Distanz des Codes? Zeigen Sie, dass

    (a) 1-Bit-Fehler in Daten- oder Parity-Bits erkannt und korrigiert werden können,
    (b) 2-Bit-Fehler erkannt aber nicht korrigiert werden können.

15. Wir codieren das Wort *REEDEREI* mit einem Binärcode für jeden Buchstaben.

    (a) Überlegen Sie einen Code mit minimaler fester Wortlänge. Wieviele Bits hat das codierte Wort?

    (b) Definieren Sie nun einen Code mit variabler Wortlänge, der zu einem kürzeren Codewort (im Vergleich zu (a)) führt. Was ist die mittlere Wortlänge Ihres Codes?

16. Gegeben ist der Zeichenvorrat $\{A, B, C, D, E, F, G, H\}$ mit der Codierung c:

Zeichen	Code	Zeichen	Code
A	100110	E	101000
B	011011	F	000111
C	100011	G	100101
D	001101	H	111100

    (a) Ermitteln Sie die minimale und maximale Hammingdistanz des Codes.

    (b) Zeichnen Sie den zugehörigen Codebaum.

    (c) Kürzen Sie die Codewörter von hinten her so weit, dass gerade noch die Fanobedingung erfüllt ist.

    (d) Zeichnen Sie den neuen Codebaum.

    (e) Decodieren Sie mit dem neuen Code die Nachricht 0110011100100011001100

17. Die Symbolsequenz $AAAAAAAABBBBBBCCCCDD$ mit der Länge 20 besitzt relative Häufigkeiten der Symbole, die als Symbolwahrscheinlichkeiten $P(A) = 0.4$, $P(B) = 0.3$, $P(C) = 0.2$, $P(D) = 0.1$ interpretiert werden können.

    (a) Konstruieren Sie den Huffman-Code.

    (b) Konstruieren Sie einen möglichst ungünstigen Code, der aber soweit gekürzt ist, dass gerade noch die Fano-Bedingung erfüllt ist, indem Sie im Huffman-Algorithmus immer die größten Wahrscheinlichkeiten statt der kleinsten zusammenfassen.

    Codieren Sie damit jeweils die Symbolsequenz und geben Sie die Länge des Bitstreams an.

18. Gegeben ist der Zeichenvorrat {P, Q, R, S, T, U, V} mit folgenden absoluten Häufigkeiten.

Zeichen	Häufigkeit	Zeichen	Häufigkeit
P	230	T	60
Q	10	U	200
R	190	V	170
S	140		

    (a) Konstruieren Sie den Huffman-Code.

    (b) Ermitteln Sie die mittlere Codelänge für den Huffman-Code sowie den kürzesten Code fixer Länge.

19. *Ein Bild sagt mehr als tausend Worte.* Überprüfen Sie diese Aussage unter der Verwendung des Informationsgehaltes. Betrachten Sie ein Grauwertbild mit $m \times n$ Bildpunkten und k Grauwerten pro Pixel und ein Vokabular von l Wörtern, wobei alle Graustufen bzw. Wörter mit jeweils gleicher Wahrscheinlichkeit auftreten. Geben Sie allgemein den Informationsgehalt des Bildes und einer Beschreibung aus w Wörtern an. Wie groß muss ein Bild mit 64 Graustufen sein, damit obige Aussage bei einem Vokabular von $l = 1024$ stimmt? Wieviele Graustufen muss ein Bild der Größe 80×60 aufweisen (bei gleichem Vokabular), um die Aussage richtig zu machen?

20. Eine Nachricht bestehe aus den folgenden Zeichen mit ihren absoluten Häufigkeiten und zwei zugeordneten Kodierungen.

Zeichen	Häufigkeit	Code 1	Code 2
A	120	000	00011
B	40	001	011
C	170	010	000101
D	150	011	001
E	260	100	010
F	60	101	1
G	90	110	0000
H	110	111	000100

Berechnen Sie die relativen Häufigkeiten h_i. Diese sollen als Schätzungen für die Zeichenwahrscheinlichkeiten p_i verwendet werden. Berechnen Sie nun den mittleren Informationsgehalt der Nachricht und die mittlere Wortlänge sowie die Redundanz, die relative Redundanz und die Codeeffizienz für:

(a) Code 1

(b) Code 2

(c) Entwickeln Sie den optimalen Code nach Huffman und berechnen Sie auch hier diese Werte.

21. Gegeben ist ein Code, der alle natürlichen Zahlen $(0, 1, 2, \ldots)$ darstellen kann. Dabei werden, um die Zahl n zu kodieren, einfach n 0-en gefolgt von einer 1 kodiert. Also 5 wird z.B. zu 000001. Die Wahrscheinlichkeit, dass die Zahl n auftritt, sei $p(n) = 2^{-(n+1)}$. Berechnen Sie die Entropie und die mittlere Codelänge dieses Codes. (Bem.: es entsteht eine konvergierende Reihe. Wenn Sie den Wert der Reihe nicht analytisch ermitteln können, berechnen Sie einfach die Summe der ersten paar Reihenglieder und schätzen das Ergebnis.)

22. Der Manchester-Code kann interpretiert werden als Recodierung eines Bits in einen 1-aus-2-Code. Das entspricht einer relativen Redundanz von 50%. Die Gleichanteilsfreiheit bedeutet, dass in den neuen Codewörtern gleich viele 0 wie 1 vorkommen. Eine Verallgemeinerung wäre daher, n Bits in einen $m/2$-aus-m-Code zu recodieren. Wie groß muss n mindestens sein, um damit eine kleinere relative Redundanz zu erzielen? Geben Sie so einen Code an.

23. Stellen Sie die folgenden Zahlen jeweils als Binär-, Oktal- und Hexadezimalzahlen dar:

$$14, \quad 1019, \quad 35388$$

24. Berechnen Sie jeweils das x:

$4112_{(5)} = x_{(11)}$ $\qquad$ $1010110111_{(2)} = x_{(6)}$

$3128_{(10)} = x_{(7)}$ $\qquad$ $10221_{(3)} = x_{(8)}$

25. Geben Sie eine Formel an, um für jede beliebige natürliche Zahl n die genaue Anzahl der Ziffern ihrer Dezimaldarstellung zu berechnen, sowie Formeln zur Berechnung der genauen Anzahl der Ziffern ihrer Binärdarstellung und ihrer Hexadezimaldarstellung.

    Verwenden Sie diese Formeln, um für sehr große natürliche Zahlen n abzuschätzen, um wie viel Prozent ihre Binär- bzw. Hexadezimaldarstellung näherungsweise länger bzw. kürzer als ihre Dezimaldarstellung ist (die Anzahl der Ziffern betreffend).

26. Gegeben ist eine Zahlendarstellung zur Basis 2 aber mit den Ziffern P und N, die die Werte $P = +1, N = -1$ besitzen (signed digits). So ist z.B. $NNPN = -8-4+2-1 = -11$. Wandeln Sie die Zahlen $PPPNNNN$, $NNNPPPP$, $PPNNPPN$, $NNPPNNP$ in die normale Binärdarstellung mit 8 Bit im 2-er-Komplement um. Versuchen Sie, ein Schema zu erkennen, mit dem man diese Umwandlung direkt und einfach durchführen kann. (Bonus: Beweisen Sie dieses Schema.)

27. Führen Sie die folgenden Rechenoperationen auf der Basis von Binärzahlen der Länge 8 jeweils in 1-er Komplement und 2-er Komplement Darstellung durch. Ermitteln Sie Überträge und Überläufe aus der Binärdarstellung. Stellen Sie das Ergebnis wieder im Dezimalsystem dar.

$$\begin{array}{rcl} 60 & - & 97 \\ 93 & + & 56 \\ -69 & - & 59 \end{array}$$

28. Wandeln Sie folgende Zahlen durch fortgesetztes Multiplizieren in die Binärdarstellung um:
$a = 0.3125 \quad b = 0.1 \quad c = \frac{1}{3} \quad d = \frac{5}{11}$

29. Addieren Sie die positiven Zahlen $a = 1110_2$, $b = 1101_2$, $c = 0110_2$ binär folgendermaßen: Addieren Sie zuerst für jede Stelle $a_i + b_i + c_i = (e_i d_i)_2$. Addieren Sie dann die zwei Zahlen $d + 2e = (d_3 d_2 d_1 d_0)_2 + (e_3 e_2 e_1 e_0 0)_2$. Überprüfen Sie das Ergebnis durch dezimale Addition. Überlegen Sie, was diese Vorgangsweise für Vorteile haben könnte.

30. *Nachträglich zur Sustainability-Week:* Bei arithmetischen Operationen in Rechenwerken wird hauptsächlich bei Bit-Flips (von 0 zu 1 oder von 1 zu 0) Energie verbraucht. Die Berechnung von Summen- und Carry-Bits ($(c_{k+1} s_k)_2 = a_k + b_k + c_k$) passiert dabei nicht sequentiell, sondern in Zeitschritten t (Reaktionszeit der Schaltelemente), in denen die Bits ihren Wert mehrmals wechseln können: $(c_{k+1}(t) s_k(t))_2 = a_k + b_k + c_k(t-1)$. Bei $t = 0$ seien alle Bits 0: $s_7(0) \ldots s_0(0) = 00000000$, $c_8(0) \ldots c_1(0) = 00000000$. Addieren Sie auf diese Weise die zwei Zahlen $a = 00110110$ und $b = 11001011$ schrittweise für $t = 0, 1, 2, \ldots$, bis sich nichts mehr ändert. Wie viele Bit-Flips ergeben sich?

Um das Klima zu entlasten, berechnen wir nun in einem ersten Taktzyklus nur die Carry-Bits, d.h.: $c_{k+1}(t) = 1$ wenn in $a_k, b_k, c_k(t-1)$ mehr als eine 1 vorkommt, sonst 0. Im zweiten Taktzyklus werden dann alle $s_k(1)$ gleichzeitig aus $a_k, b_k, c_k(7)$ berechnet. Wie viele Bit-Flips gibt es jetzt?

31. Führen Sie folgende Berechnungen im Binärsystem durch.

    (a) $183 \cdot 77$ (b) $234/6$

32. Wandeln Sie die Dezimalzahlen 1.01, 1.02, 1.03 und 1.04 in das Binärsystem mit 7 Nachkommastellen um. Nachfolgende Stellen werden abgeschnitten, d.h. abgerundet. Wandeln Sie das Ergebnis danach wieder zurück in das Dezimalsystem und runden Sie auf zwei Nachkommastellen. Bekommen Sie immer die ursprüngliche Zahl heraus?

33. Dividieren Sie im jeweiligen Zahlensystem

$$434_7/23_7 \qquad 1206_8/42_8$$

    und überprüfen Sie das Ergebnis durch die Darstellung im Dezimalsystem.

34. Stellen Sie folgende 32-Bit IEEE Floating Point Zahlen als Dezimalzahlen dar:

```
0 01000000 01010101010101010101010
1 11111111 00110111110000100010000
1 00000000 00000000000000000000000
0 00001101 00110011001000000000000
1 11111111 00000000000000000000000
0 00110110 11010011000000000000000
```

35. Gegeben ist folgende Gleitpunktdarstellung mit 8 Bit: $v\, m_2 m_1 m_0\, c_3 c_2 c_1 c_0$. Die damit dargestellten Zahlen ergeben sich durch $M \cdot 2^C$, wobei die Mantisse $M = (v\,\bar{v}\,m_2 m_1 m_0)_2$ und der Exponent $C = (c_3 c_2 c_1 c_0)_2$ im 2er-Komplement zu interpretieren sind. $\bar{v}$ ist 1, wenn $v = 0$ ist, und umgekehrt.

    Ermitteln Sie die Darstellung von 0.75 und −5, sowie die betragsmäßig größten und kleinsten positiven und negativen Zahlen und ihre Darstellung.

36. Zwei Zahlen x und y in der 32-Bit-IEEE-Darstellung sollen multipliziert werden.

    (a) Stellen Sie die Zahlen $x = -1.25$ und $y = 3.125$ im obigen Format dar.

    (b) Geben Sie allgemeine Gleichungen für das Vorzeichenbit $v_z = f(v_x, v_y)$, die Charakteristik $c_z = f(c_x, c_y, m_x, m_y)$, und die Mantisse $m_z = f(m_x, m_y)$ des Produkts z an (**Anm.:** Das Produkt z sei als darstellbar vorausgesetzt, Spezialfälle von x und y seien ausgeschlossen).

    (c) Überprüfen Sie Ihr Ergebnis mit Hilfe der Zahlen aus (a).

37. Erstellen Sie die Wahrheitstabellen für die folgenden Aussageformen. Welche der Aussageformen sind äquivalent? Welche Gesetzmäßigkeiten erkennen Sie?

    $a \vee b,\ a \wedge b,\ b \vee a,\ b \wedge a,$
    $\overline{a \wedge b},\ \bar{a} \wedge \bar{b},\ \overline{a \vee b},\ \bar{a} \vee \bar{b}$

    $a \to b,\ b \to a,\ \bar{a} \to \bar{b},\ \bar{b} \to \bar{a},\ \overline{a \to b},\ \overline{b \to a}$

    $(a \vee b) \wedge c,\ (a \wedge b) \vee c,\ (a \vee c) \wedge (b \vee c),\ (a \wedge c) \vee (b \wedge c)$

38. Zeigen Sie, dass die Verknüpfung von n Bits $x_1, x_2, \dots, x_n$ mit XOR, dem exklusiven Oder, für $n \geq 2$ immer genau das Ergebnis für die gerade Parität ergibt. Also:

    $x_{n+1} = x_1 \oplus x_2 \oplus \dots \oplus x_n$ ist genau die *even parity* dieser n bits.

39. Zeigen Sie durch Umformen, dass die Aussageform $a \oplus b$ äquivalent zu $\overline{a \leftrightarrow b}$ ist.

40. Zeigen Sie mittels Umformungen, ob die folgenden Aussageformen Tautologien oder Kontradiktionen sind:

    $a \wedge (b \vee \bar{a}) \wedge \bar{b}$

    $(\overline{a \to b} \vee (\overline{b \vee a} \wedge (b \leftrightarrow a))) \wedge b$

    $((b \to c) \to \overline{a \to c}) \vee ((\bar{b} \to a) \to c)$

41. Für drei *ungleiche* Zahlen x, y, z sind die Aussagen a, b, c definiert als:

    $$a :\Leftrightarrow x < y, \quad b :\Leftrightarrow y < z, \quad c :\Leftrightarrow x < z.$$

    Für welche Wahrheitswerte von a, b, c lassen sich drei Zahlen finden, die die Aussagen erfüllen? Erstellen Sie die Wahrheitstabelle für diese Erfüllbarkeit. Zeigen Sie, dass die Aussageform

    $$((c \to a) \vee (c \to b)) \wedge ((a \to c) \vee (b \to c))$$

    die selbe Wahrheitsfunktion darstellt.

42. Zeigen Sie, dass die Peirce-Basis (NOR-Basis) eine Verknüpfungsbasis ist, daher also alle aussagenlogischen Sätze nur mit dem Junktor $\downarrow$ darstellbar sind.

43. Gegeben sei eine Menge A von n Aussagevariablen.

    a. Wieviele verschiedene Konjunktionsterme können daraus gebildet werden?

    b. Wieviele obiger Konjunktionsterme sind Minterme?

    c. Wieviele verschiedene vollständige DNFen können mit den Variablen aus A gebildet werden?

44. Weisen Sie für die Verknüpfungen *AND*, *NOR* und *XOR* durch Umformen nach, dass die jeweils abgeleitete KNF äquivalent zur aus der Wahrheitstabelle abgeleiteten DNF ist.

45. Überprüfen Sie, ob die Terme ab und $\bar{a}b\bar{c}$ Implikanten und/oder Primimplikanten von $(a \to \bar{b}) \to \overline{b \to c}$ sind. Verwenden Sie dazu nur Umformungen und/oder Einsetzen von Wahrheitswerten.

46. Finden Sie alle Aussagenvariablenbelegungen der Aussageform

$$f(a,b,c,d) \equiv (a \vee \bar{b})(\bar{b} \vee c)(\bar{c} \vee \bar{d})(d \vee \bar{a}),$$

die die Aussageform wahr ($\mathbf{1}$) machen, und zwar auf folgende Weise: Ermitteln Sie zuerst $f(\mathbf{1},b,c,d)$ und $f(\mathbf{0},b,c,d)$, und kürzen Sie so weit wie möglich Variablen oder Terme. Ermitteln Sie daraus z.B. $f(\mathbf{1},\mathbf{1},c,d)$, $f(\mathbf{1},\mathbf{0},c,d)$, usw., bis $f(\mathbf{0},\mathbf{0},\mathbf{0},\mathbf{0})$. Es ergibt sich ein Suchbaum. Bei geschickter Wahl der Variablen (z.B. $f(\mathbf{1},b,c,\mathbf{0})$) kann der Suchbaum frühzeitig gekürzt werden.

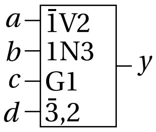
47. Vereinfachen Sie folgende Aussageformen mittels K-Diagramm, d.h. finden Sie die wesentlichen Primimplikanten (DMF).

    (a) $\overline{\bar{c}d \vee a\bar{b}c \vee bc(d \to a)}$ (Beachten Sie die große Negation über der Aussageform!)

    (b) Wie (a), aber der Wahrheitswert sei egal, wenn $bc(d \to a)$ gilt.

48. Wie Beispiel 47, aber lösen Sie die Aufgabe jeweils mit dem Quine-McCluskey-Algorithmus.

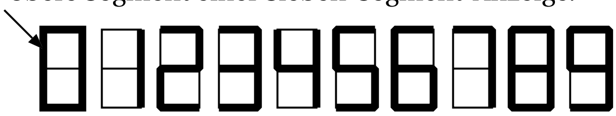
49. (a) Welche Aussageform repräsentiert folgendes Schaltsymbol mit Abhängigkeitsnotation?



    (b) Definieren Sie ein Schaltsymbol zur Aussageform $y := (a \vee bd) \oplus (c\bar{d} \vee bd)$.

    Zeichnen Sie jeweils auch die Schaltung.

50. Betrachten Sie das linke obere Segment einer Sieben-Segment-Anzeige.



Konstruieren Sie eine Schaltung mit vier Eingängen x_3, x_2, x_1, x_0, die die angezeigte Zahl im Binärformat darstellen, und einem Ausgang y, der dann 1 ist, wenn das linke obere Segment an ist. Wenn die Eingänge keine Dezimalziffer darstellen, soll die Anzeige nichts anzeigen.

(a) Erstellen Sie die Wahrheitstabelle.

(b) Ermitteln Sie die vollständige DNF und minimieren Sie die Schaltfunktion mit dem Verfahren von Quine-McCluskey.

(c) Ermitteln Sie die wesentlichen Primimplikanten.

(d) Realisieren Sie die Schaltung mit OR-, AND-, und NOT-Gattern.

51. Entwerfen Sie ein Schaltnetz, das für zwei ganze Zahlen in Binärdarstellung mit jeweils zwei Bits (Darstellung der Dezimalzahlen 0, 1, 2, 3) ihre ganzzahlige Division mit Rest berechnet:

    $a/b = q\,R\,r \quad :\Leftrightarrow \quad a = qb + r,\ \ 0 \le r < b$

    Eingänge: Dividend a (aus zwei Bits a_1 und a_0), Divisor b (aus zwei Bits b_1 und b_0).

    Ausgänge: Quotient q (aus zwei Bits q_1 und q_0), Rest r (aus zwei Bits r_1 und r_0).

    Bei Division durch 0 (also wenn b_1 und b_0 beide 0 sind) sollen die Werte der Ausgänge egal sein.

    Geben Sie die Wahrheitstabelle an, bestimmen Sie für jeden Ausgang eine disjunktive Minimalform mittels K-Diagramm und skizzieren Sie das Schaltnetz.

52. Entwerfen Sie ein Schaltnetz, das für zwei ganze Zahlen in Binärdarstellung mit jeweils zwei Bits (Darstellung der Dezimalzahlen 0, 1, 2, 3) ihren Quotienten in binärer Festkommadarstellung berechnet:

    $a/b = q$ (gerundet)

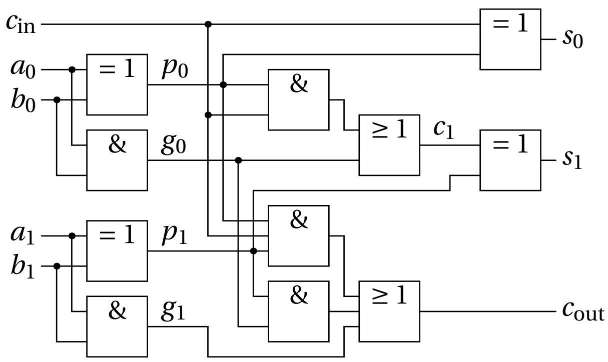
    Eingänge: Dividend a (aus zwei Bits a_1 und a_0), Divisor b (aus zwei Bits b_1 und b_0).

    Ausgänge: Quotient q (aus vier Bits, davon zwei Bits Vorkommastellen q_1 und q_0 und zwei Bits Nachkommastellen q_{-1} und q_{-2}, Nachkommastellen gerundet, nicht nur abgeschnitten).

    Bei Division durch 0 (also wenn b_1 und b_0 beide 0 sind) sollen die Werte der Ausgänge egal sein.

    Geben Sie die Wahrheitstabelle an und bestimmen Sie für die beiden Nachkommastellen-Ausgänge jeweils eine disjunktive Minimalform mittels Quine-McCluskey-Algorithmus. Was kann man zu den beiden Vorkommastellen-Ausgängen sagen? Skizzieren Sie das Schaltnetz.

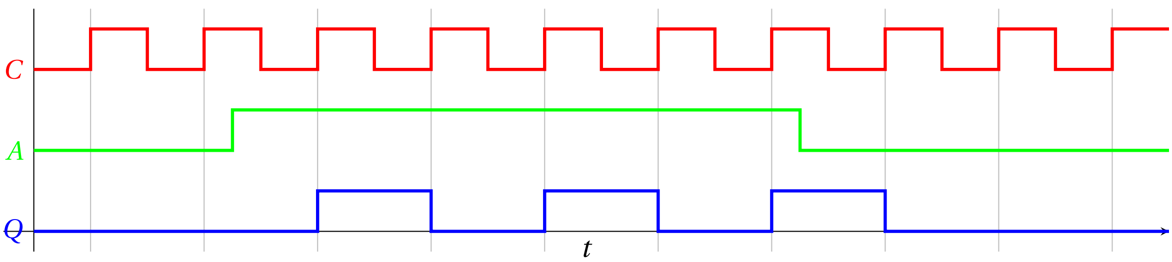
53. Gegeben ist folgendes Schaltnetz:

    Bestimmen Sie die Aussageformen der Ausgänge sowie die vollständige Wahrheitstabelle. Welcher Funktion entspricht dieses Schaltnetz?

54. Überlegen Sie sich eine Schaltung, die als Kernelement ein flankengesteuertes JK-Flipflop enthält: Eingänge sind ein Taktsignal C und sowie ein Steuereingang A. Der Ausgang Q soll folgendes Verhalten zeigen: Wenn A auf 0 gesetzt wird, soll auch Q konstant 0 werden. Wenn A auf 1 gesetzt wird, soll Q mit halber Frequenz des Taktsignals oszillieren, also immer zwischen 1 und 0 wechseln.

    Beispiel:

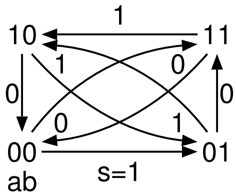
    

55. Ein Schleusensystem hat ein oberes (o) und ein unteres Tor (u), die jeweils offen ($o = 1$ bzw. $u = 1$) oder geschlossen ($o = 0$ bzw. $u = 0$) sein können. Zusätzlich ist die Schleuse entweder voll ($v = 1$) oder leer ($v = 0$). Die Schleuse durchläuft also folgenden Zyklus:

$$(o, u, v) = (0,0,0) \mapsto (1,0,1) \mapsto (0,0,1) \mapsto (0,1,0) \mapsto (0,0,0) \mapsto \cdots$$

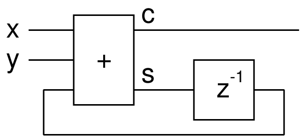
Realisieren Sie dieses System mit JK-Flipflops. Minimieren Sie die Schaltfunktionen für die Eingänge der JK-Flipflops mit K-Diagrammen. Zeichnen Sie die Schaltung.

56. Eine Signalanlage mit zwei Lampen a, b soll in Abhängigkeit einer Steuerleitung s gleich oder gegengleich blinken. Es ergeben sich folgende Zustandsübergänge:



Realisieren Sie dieses System mit JK-Flipflops. Minimieren Sie die Schaltfunktionen für die Eingänge der JK-Flipflops mit K-Diagrammen. Zeichnen Sie die Schaltung.

57. Die Sigma-Delta-Modulation (SDM) ist eine 1-Bit Sampling-Methode, die in Analog-Digital-Convertern (ADC) verwendet wird. Dabei entspricht der Signalwert grob der Häufigkeit der 1-Bits. Die Signal-Summe zweier solcher SDM-Bitstreams x und y ist gegeben durch folgendes Schema:



Hier ist + ein Volladdierer und z^{-1} eine Verzögerung um einen Clock-Zyklus, also ein Speicher. Es wird also anders als bei einem sequentiellen binären Addierer das Summenbit statt des Carry-Bits weitergegeben. Realisieren Sie das Schema mit einem JK-Flipflop.

58. Entwickeln Sie einen 2×1-Bit Speicher mit JK-Flipflops. Die Schaltung hat einen Adresseingang a, der eines der zwei Bits auswählt, einen Daten-Input i, dessen Zustand in das durch a ausgewählte JK-Flipflop übernommen wird, wenn der Read-Write-Eingang w auf 1 ist. Der Ausgang o soll den Zustand des durch a ausgewählten Flipflops ausgeben.