

2. Übungsblatt

Aufgabe 1 Boolesche Algebra / Schaltalgebra

- a) Zeigen Sie, ausgehend von den in der Vorlesung behandelten De Morganschen Gesetzen für zwei Boolesche Variablen, d.h., ausgehend von

$$\neg(a \wedge b) = \neg a \vee \neg b \quad \text{und} \quad \neg(a \vee b) = \neg a \wedge \neg b,$$

daß für n Boolesche Variablen a_1 bis a_n , $n \geq 2$, allgemein gilt:

$$\begin{aligned} \neg(a_1 \wedge a_2 \wedge \dots \wedge a_n) &= \neg a_1 \vee \neg a_2 \vee \dots \vee \neg a_n & \text{und} \\ \neg(a_1 \vee a_2 \vee \dots \vee a_n) &= \neg a_1 \wedge \neg a_2 \wedge \dots \wedge \neg a_n, \end{aligned}$$

oder kompakter geschrieben:

$$\neg \left(\bigwedge_{i=1}^n a_i \right) = \bigvee_{i=1}^n \neg a_i \quad \text{und} \quad \neg \left(\bigvee_{i=1}^n a_i \right) = \bigwedge_{i=1}^n \neg a_i.$$

- b) Welche anderen Axiome/Gesetze kann man in analoger Weise verallgemeinern? Wie sehen die verallgemeinerten Gesetze/Regeln aus?

Aufgabe 2 Boolesche Algebra / Schaltalgebra

Zeigen Sie mit Hilfe der Gesetze/Axiome der Booleschen Algebra, daß folgende Terme äquivalent (gleichbedeutend) sind! Geben Sie für jeden Schritt das verwendete Gesetz/Axiom der Booleschen Algebra an!

Term 1: $(a \wedge \bar{c}) \vee b$

Term 2: $\overline{(\bar{a} \wedge \bar{b} \wedge \bar{c})} \wedge (a \vee b \vee \bar{c}) \wedge \overline{(c \wedge \bar{b})}$

Aufgabe 3 (Programmierbare) Logikarrays

In der Vorlesung wurde gezeigt, wie man (programmierbare) Logikarrays mit ausschließlich NOR-Gattern (und ggf. NOT-Gattern) implementieren kann.

- Wie kann man (programmierbare) Logik-Arrays mit ausschließlich und möglichst wenigen NAND-Gattern (und ggf. NOT-Gattern) implementieren? (Hinweis: vgl. die Herleitung für NOR-Gatter in der Vorlesung)
- Welche Bedeutung haben diese Darstellungen (mit ausschließlich NOR-Gattern bzw. ausschließlich NAND-Gatter) für eine Hardware-Implementierung?
- Wenden Sie das in a) gefundene Verfahren an auf die Boolesche Funktion

$$f(A, B, C) = (A \wedge B \wedge \bar{C}) \vee (A \wedge \bar{B} \wedge \bar{C}) \vee (A \wedge \bar{B} \wedge C).$$

Aufgabe 4 Demultiplexer

Ein Demultiplexer berechnet die Umkehrung der Funktion eines Multiplexers, wie er in der Vorlesung (für vier Datenleitungen und zwei Auswahlleitungen) besprochen wurde. In dieser Aufgabe betrachten wir einen Demultiplexer mit zwei Ausgabeleitungen und einer Steuerleitung, die bestimmt, auf welche Ausgabeleitung eine Eingabe weitergeleitet wird. (Die jeweils andere Ausgabeleitung soll den Wert 0 erhalten.) Konstruieren Sie einen Demultiplexer (Wahrheitstabelle / Schnittstelle / Implementierung) unter Verwendung der Ihnen bekannten (Logik-)Gatter.