

Programmierbare Logische Arrays

Aufgabe 1: PLA Programmierung (30 %)

Betrachten Sie die zweistellige Nicod–Nand Funktion Δ .

- Realisieren Sie auf der Basis der vollständigen disjunktiven Normalform die Schaltfunktion Δ durch ein Programmierbares Logisches Array (PLA).
- Geben Sie eine Darstellung der Funktion Δ an, so dass Δ durch ein PLA mit weniger Einheitsbausteinen als in Teilaufgabe a) realisiert werden kann. Geben Sie das resultierende PLA wieder.

Aufgabe 2: PLA–realisierbare Funktionen (30 %)

Betrachten Sie ein PLA mit 3 Eingängen für die UND–Ebene, 4 Spalten und 1 Ausgang der ODER–Ebene (das PLA realisiert also Funktionen $f : \mathbb{B}^3 \rightarrow \mathbb{B}^1$). Lassen sich mit einem derartigen PLA bereits alle möglichen Funktionen des Typs $f : \mathbb{B}^3 \rightarrow \mathbb{B}^1$ realisieren? Begründen Sie ihre Antwort bzw. geben Sie ein Gegenbeispiel mit Erläuterung.

Aufgabe 3: PLAs und Multiplexer (40 %)

Im vergangenen Semester wurde eine Technik vorgestellt, um boolesche Funktionen mit Hilfe von Multiplexern umzusetzen (vgl. hierzu [OV99]). In dieser Hinsicht besitzen Multiplexer und PLAs also ähnliche Eigenschaften. Realisieren Sie die Funktion

$$f(x_1, x_2, x_3, x_4) = x_1x_3\bar{x}_4 + \bar{x}_1\bar{x}_3x_4 + \bar{x}_2\bar{x}_3\bar{x}_4$$

- als MUX mit 8 Dateninputs,
- als PLA ausreichender Größe.

Literatur

[OV99] W. Oberschelp, G. Vossen, *Rechneraufbau und Rechnerstrukturen*. Oldenbourg Verlag, München (1999)

Abgabe: Bis 22. Juni 2000 in den Tutorien