Unterlagen zur Vorlesung

Hardware und Systemgrundlagen

Prof. Dr. Jürgen Neuschwander

Shannonscher Entwicklungssatz

Realisierungen von Schaltfunktionen durch Multiplexer-Bausteine

Shannonscher Entwicklungssatz

Shannonscher Entwicklungssatz:

Disjunktive Form:

Nur diese Form wird im Kurs betrachtet

$$f(x_1, .. \; x_n) = \; x_i \wedge f(x_1, ..., x_{i-1}, 1, x_{i+1}, ..., x_n] \vee [\bar{x}_i \wedge f(x_1, ..., x_{i-1}, 0, x_{i+1}, ..., x_n]$$

Konjunktive Form:

$$f(x_1, .. \; x_n) = \; [x_i \vee f(x_1, ..., x_{i-1}, 0, x_{i+1}, ..., x_n] \wedge [\bar{x}_i \vee f(x_1, ..., x_{i-1}, 1, x_{i+1}, .., x_n]$$

Die Schaltfunktion wurde nach der Variablen x_i entwickelt.
Die neu entstehenden Funktionen, die unabhängig von der Entwicklungsvariablen x_i sind, heißen Restfunktionen.

Shannonscher Entwicklungssatz am Beispiel

$$y = a\,\overline{b}\,c \vee \overline{a}\,\overline{b} \vee b\,c$$

$$= a\,[1\,\overline{b}\,c \vee \overline{1}\,\overline{b} \vee b\,c] \vee \overline{a}\,[0\,\overline{b}\,c \vee \overline{0}\,\overline{b} \vee b\,c]$$

$$= a\,[\,\overline{b}\,c \vee b\,c\,] \vee \overline{a}\,[\,\overline{b} \vee b\,c\,]$$

$$= a\,[\,b\,(c) \vee \overline{b}\,(c)\,] \vee \overline{a}\,[\,b\,(c) \vee \overline{b}\,(1)\,]$$

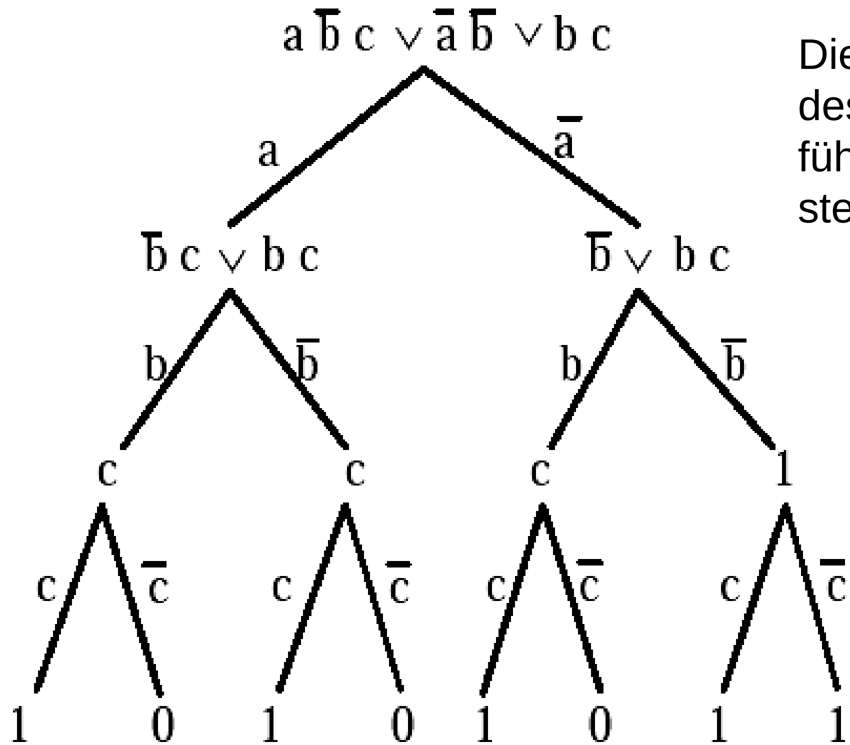
Entwickeln, bis die Restfunktion trivial ist

$$= a\,[\,b\,(c) \vee \overline{b}\,(c)\,] \vee \overline{a}\,[\,b\,c \vee \overline{b}\,(c \vee \overline{c})\,]$$

$$= a\,b\,c \vee a\,\overline{b}\,c \vee \overline{a}\,b\,c \vee \overline{a}\,\overline{b}\,c \vee \overline{a}\,\overline{b}\,\overline{c}$$

Optional: Erweiterung, zur DNF Darstellung

Shannonscher Entwicklungssatz am Beispiel



Die rekursive Anwendung des Entwicklungssatzes führt in der grafischen Darstellung zu einem Binärbaum.

Nachdem die Funktion nach allen Variablen entwickelt wurde, können die Minterme durch Verfolgen der Äste des Baums gefunden werden, die zu einer 1 führen.

Shannonscher Entwicklungssatz am weiteren Beispiel

- Beispiel für die Anwendung des Entwicklungssatzes:

gegeben: $\boxed{y = x_3 \cdot x_2 \cdot \overline{x_1} \vee \overline{x_2} \cdot x_1}$ gesucht: Binärbaum

$$y = x_3 \cdot [x_2 \cdot \overline{x_1} \vee \overline{x_2} \cdot x_1] \vee \overline{x_3} \cdot [0 \vee \overline{x_2} \cdot x_1]$$

Entwicklung nach x_3

$$y = x_3 \cdot [x_2 \cdot (\overline{x_1} \vee 0) \vee \overline{x_2} \cdot (0 \vee x_1)] \vee \overline{x_3} \cdot [x_2 (0) \vee \overline{x_2} (x_1)]$$

Entwicklung der Restfunktionen nach x_2

$$y = x_3 \cdot [x_2 \cdot (\overline{x_1}) \vee \overline{x_2} (x_1)] \vee \overline{x_3} \cdot [x_2 \cdot (0) \vee \overline{x_2} \cdot (x_1)]$$

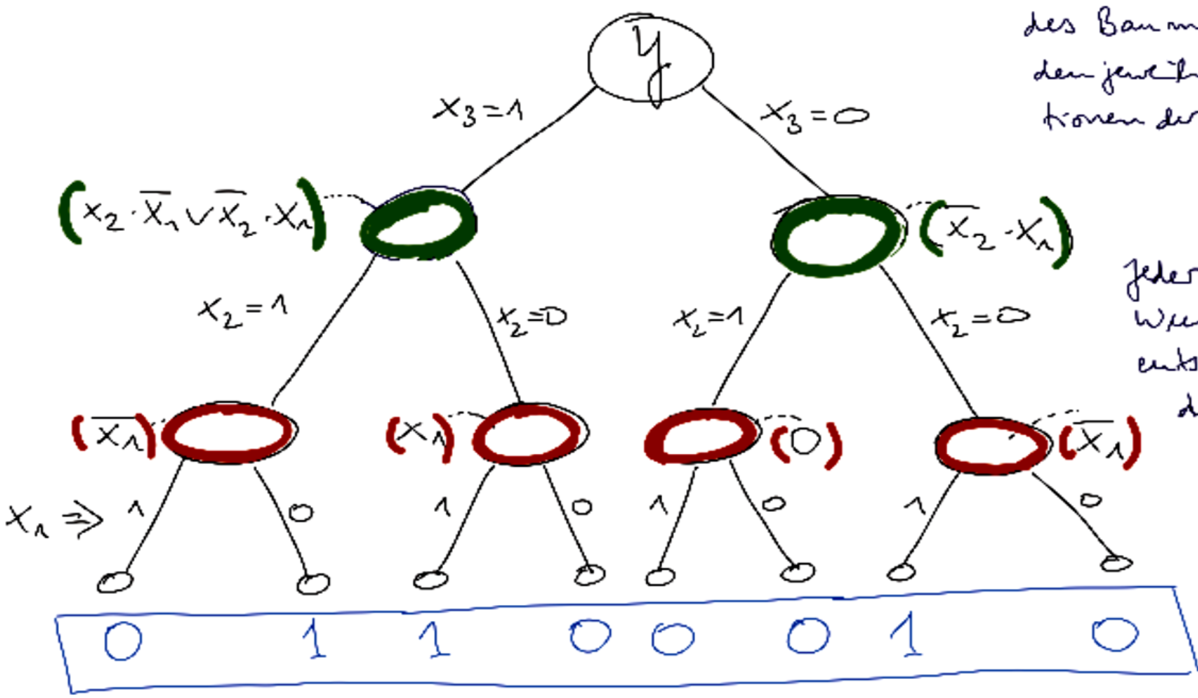
Normal hingeschrieben

Würde man jetzt auch noch die Restfunktionen $(\overline{x_1})$ (x_1) ..... weiter entwickeln, käme man zur Darstellung:

$$y = x_3 \cdot [x_2 \cdot (x_1 \cdot (0) \vee \overline{x_1}(1)) \vee \overline{x_2} \cdot (x_1 \cdot (1) \vee \overline{x_1} \cdot (0))]$$
$$\vee\ \overline{x_3} \cdot [x_2 \cdot (0) \vee \overline{x_2} \cdot (x_1 \cdot (1) \vee \overline{x_1} \cdot (0))]$$

- Binärbaum zum Beispiel:

⇒ Die Farben der Knoten des Baumes entsprechen den jeweiligen Restfunktionen der Entwicklung.



Jeder Pfad von der Wurzel zum Blatt entspricht einer Zeile der Funktionstabelle.

In den Blättern des Baumes stehen die Funktionswerte von y.

Diese letzte Ebene entspricht der Funktionswertespalte, die um 90 Grad nach rechts gedreht wurde.
(1. Zeile: $(x_3,x_2,x_1) = (0,0,0) \rightarrow f=0$ usw.)

Shannonscher Entwicklungssatz und Realisierungen

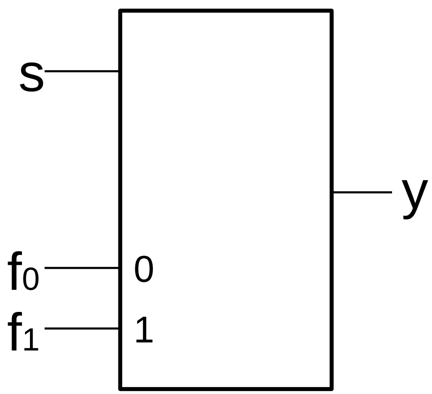
Wie kann man den Shannonschen Entwicklungs-satz denn für die Realisierung von Schalt-funktionen verwenden ??

Dazu ist die Einführung eines speziellen Bausteins nötig: Der Multiplexer.

→ Wie sieht denn solch ein Multiplexer aus?

Multiplexer-Baustein

2:1 - Multiplexer

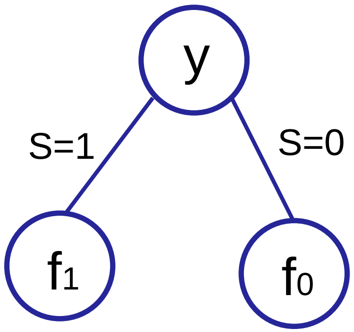


s: Steuervariable

f_i: Restfunktion

Charakteristische Gleichung ≡ Entwicklungssatz

$$y = s \cdot f_1 \vee \overline{s} \cdot f_0$$

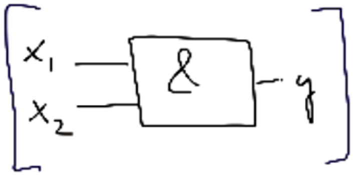


Binary Decision Tree (BDT)

Einfache Funktionen mit Multiplexer-Baustein

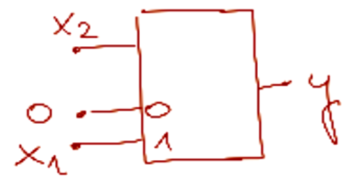
Einfache Anwendungsbeispiele:

a) Realisiere die UND-Fktn mit einem MUX-Baustein!

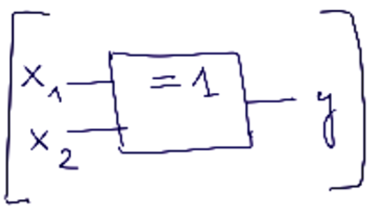


$\Rightarrow$ $y = x_2 \cdot x_1$; jetzt Entwicklung nach x_2

$y = x_2 \cdot (x_1) \vee \overline{x_2} \cdot (0)$ $\Rightarrow$



b) Realisiere die Antivalenzfktn mit einem MUX-Baustein!

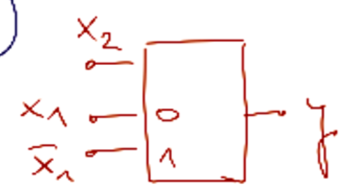


$\Rightarrow$ $y = x_2 \nleftrightarrow x_1$; jetzt Entwicklungssatz nach x_2

$y = x_2 \cdot (1 \nleftrightarrow x_1) \vee \overline{x_2} \cdot (0 \nleftrightarrow x_1)$

nach Grundregeln $\nleftrightarrow$;

$y = x_2 \cdot (\overline{x_1}) \vee \overline{x_2} \cdot (x_1)$ $\Rightarrow$



Shannonscher Entwicklungssatz und Realisierungen

Analogie:

- Aus der DNF (KNF) ließ sich die direkte Implementierung einer Schaltfunktion durch 1:1-Abbildung des Ausdrucks auf die Gatterebene angeben.

Bei Multiplexern:

- Aus der Entwicklung einer Schaltfunktion lässt sich ebenfalls eine direkte Implementierung einer Schaltfunktion durch 1:1-Abbildung auf einen (oder mehrere) 2:1-Multiplexer angeben.

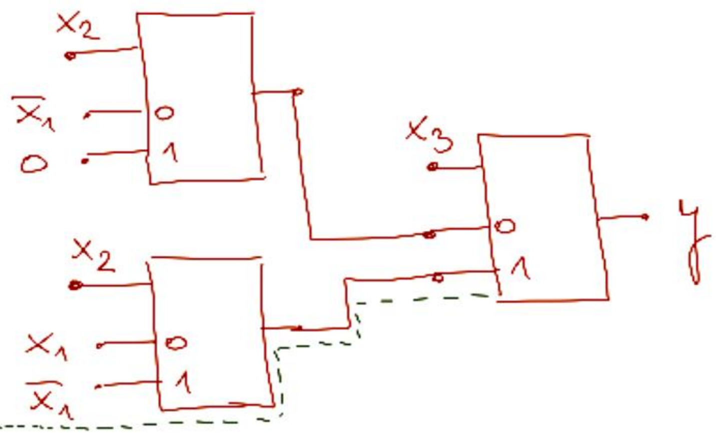
Realisierung des vorigen Beispiels

- Wie sieht die MUX-Lösung für unser vorheriges Beispiel aus?

  [$y = x_3 \cdot x_2 \cdot \overline{x_1} \vee \overline{x_2} \cdot x_1$]

  ⇒ Man drehe den Entwicklungsbaum (siehe andere Folie) um 90 Grad nach rechts und laufe die einzelnen Pfade durch.

  <u>Ergebnis:</u>



↳ $x_3 = 1$, $x_2 = 1$ ergibt $\overline{x_1}$

Anmerkung zur Entwicklung mit Variablen

Anmerkung:

Die Entwicklungsvariable kann jeweils frei gewählt werden, also eine Funktion von den drei Variablen a,b,c kann z.B. zuerst nach b, dann nach c und dann nach a entwickelt werden. Das Ergebnis der jeweils gewählten Reihenfolge der Entwicklung kann jedes Mal einen anderen Binärbaum und damit ein anderes Multiplexer-Netz ergeben.

Möchte man die Lösung mit der geringsten Anzahl von Multiplexern, also die ökonomischste Lösung, dann muss man leider alle Varianten durchprobieren. Es ist kein Algorithmus bekannt, der es erlaubt vorher zu erkennen, welche Reihenfolge der Variablen diese optimale Lösung liefert.