Kapitel 8

(Zweistufige) Logiksynthese

8.1 PLAs und zweistufige Logiksynthese

8.2 Implikanten und Primimplikanten

8.3 Algorithmus zur Berechnung eines Minimalpolynoms

Verfügbare Technologien

Zur Auswahl stehende Technologien

- Nurlesespeicher
  - Read Only Memory (ROM)
  - Programmable ROM (PROM)
  - Erasable PROM (EPROM), ...
- **Zweistufige Realisierungen**
  - **Progr. Logische Felder (PLA)**
- Mehrstufige Realisierungen
  - Gate-Arrays- und Sea-of-Gates Entwurf
  - Field Programmable Logic Arrays (FPGA)

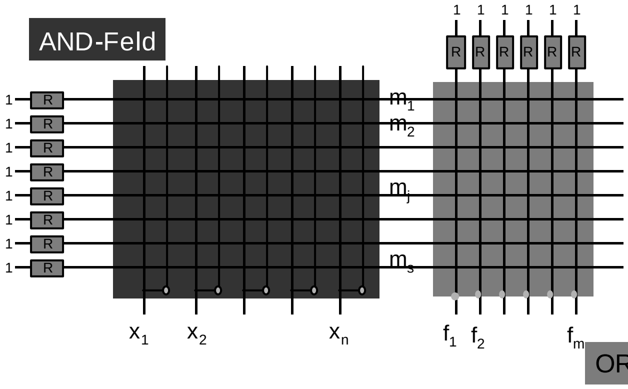
Konkurrierende Optimierungsziele

- belegte Fläche
  - Ausbeute
- benötigte Reaktionszeiten
  - Korrektheit des Gesamtsystems
- Testbarkeit
  - Korrektheit des verkauften Chips
- Leistungsverbrauch
  - neue Märkte (Notebooks,...)
- Entwicklungszeit und -kosten
  - time-to-market
  - Wirtschaftskraft der Kunden

Programmierbare Logische Felder (PLA)

Zweistufige Darstellung zur Realisierung von Booleschen Polynomen

$f_i = m_{i1}+m_{i2}+ \dots +m_{ik}$ mit m_{iq} aus $\{m_1,\dots,m_s\}$



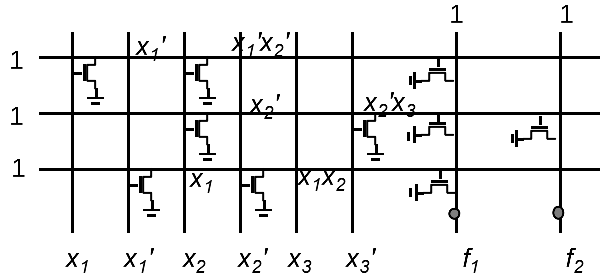
Enthält Monom m_j k Literale, so werden k Transistoren in der entsprechenden Zeile des **AND-Feldes** benötigt.

Besteht die Beschreibung von Funktion f_t aus p Monomen, so benötigt man p Transistoren in der entsprechenden Spalte des **OR-Feldes**.

Fläche: ~ (m+2n)x(Anzahl_der_benötigten_Monome)
Laufzeit: ~konstante Laufzeit (ohne die Kapazitäten zu berücksichtigen)

Programmierbare logische Felder ff

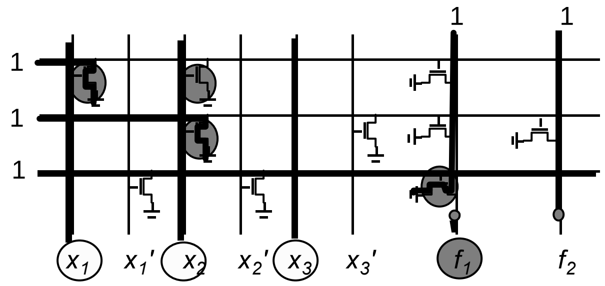
Beispiel $f_1 = x_1'x_2'+x_2'x_3+x_1x_2$
$f_2 = x_2'x_3$



Programmierbare Logische Felder ff

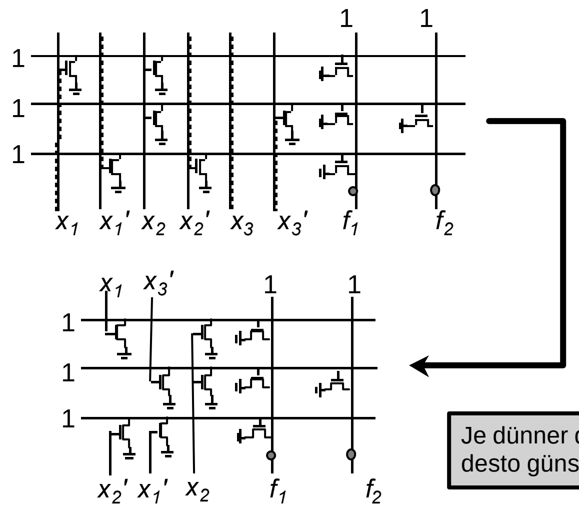
Beispiel $f_1 = x_1'x_2' + x_2'x_3 + x_1x_2$
$f_2 = x_2'x_3$

Bei Belegung von $x_1=1,\ x_2=1,\ x_3=1$ liegt folgende Situation vor.



Programmierbare Logische Felder ff

Durch **Faltung eines PLAs** kann Fläche eingespart werden:



- **Streichen überflüssiger Leitungen**
- **Verlegen von Leitungen in Freiräume**

Je dünner die Felder belegt, desto günstiger für die Faltung!

Kosten von Monomen

Sei $q = q_1 \cdot \ldots \cdot q_r$ ein Monom,
dann sind die

Kosten $|q|$ von q

gleich der Anzahl der zur Realisierung von q benötigten Transistoren im PLA, also

$|q| := r$

Kosten von Polynomen

Seien $p_1,\ldots,p_m$ Polynome,
dann bezeichne $M(p_1,\ldots,p_m)$
die Menge der in diesen Polynomen verwendeten Monome.

- Die **primären Kosten** $cost_1(p_1,\ldots,p_m)$ einer Menge $\{p_1,\ldots,p_m\}$ von Polynomen sind gleich der Anzahl der benötigten Zeilen im PLA, um $p_1,\ldots,p_m$ zu realisieren, also
$$cost_1(p_1,\ldots,p_m) = |M(p_1,\ldots,p_m)|.$$
- Die **sekundären Kosten** $cost_2(p_1,\ldots,p_m)$ einer Menge $\{p_1,\ldots,p_m\}$ von Polynomen sind gleich der Anzahl der benötigten Transistoren im PLA, also
$$cost_2(p_1,\ldots,p_m) = \Sigma_{q \in M(p_1,\ldots,p_m)} |q| + \Sigma_{i=1,\ldots,m} |M(p_i)|$$

Das Problem der 2-stufigen Logikminimierung

Gegeben

sei eine Boolesche Funktion f= $(f_1,\dots,f_m)$ in n Variablen und m Ausgängen in Form

- einer Tabelle der Dimension $(n+m)2^n$ oder
- einer Menge von m Polynomen $\{p_1,\dots,p_m\}$ mit $y(p_i)=f_i$.

Gesucht

ist eine Menge von Polynomen $\{g_1,\dots,g_m\}$ mit den Eigenschaften

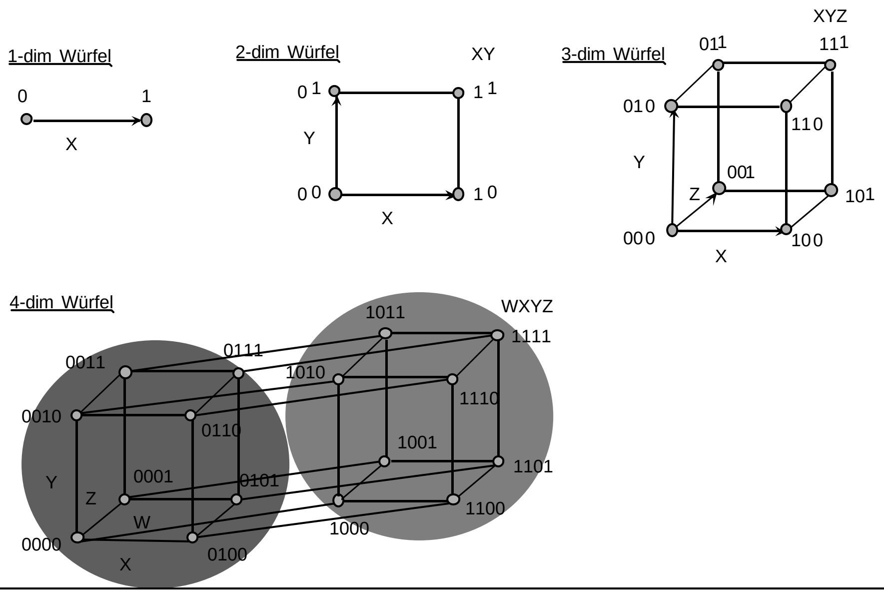
(1) $y(g_i)=f_i$ für alle i,

(2) $cost_1(g_1,\dots,g_m)$ ist minimal unter der Bedingung (1),

(3) $cost_2(g_1,\dots,g_m)$ ist minimal unter den Bedingungen (1) und (2).

Im folgenden wollen wir der Einfachheit halber nur vollständig definierte Boolesche Funktionen (in n Variablen) mit einem Ausgang betrachten.

Veranschaulichung von Monomen / Polynomen



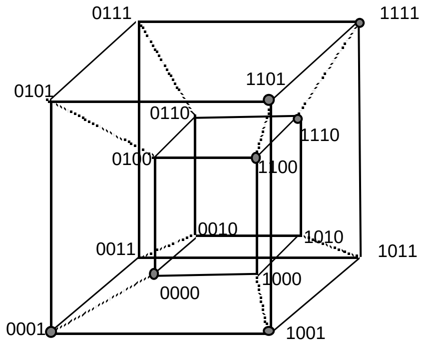
Veranschaulichung durch Würfel ff

Jede Boolesche Funktion f in n Variablen und einem Ausgang kann über einen n-dimensionalen Würfel durch Markierung der ON(f)-Menge spezifiziert werden

Beispiel

$f(x_1,x_2,x_3,x_4)$

$= x_1x_2+x_1'x_2'x_3'+ x_1x_2'x_3'x_4$



Veranschaulichung durch Würfel ff

- Monome der Länge k entsprechen n-k dimensionalen Teilwürfeln !
- Ein Polynom entspricht einer Vereinigung von Teilwürfeln.

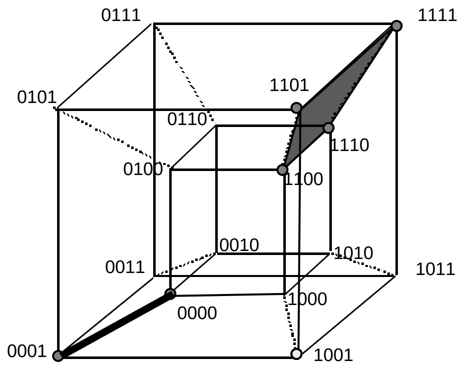
- **Beispiel**

  $f(x_1,x_2,x_3,x_4)$

  $= \boxed{x_1x_2}$

  $+ \boxed{x_1'x_2'x_3'}$

  $+ \boxed{x_1x_2'x_3'x_4}$



2-stufige Logikminimierung für single-output Funktionen

Formulierung als Überdeckungsproblem auf dem Würfel

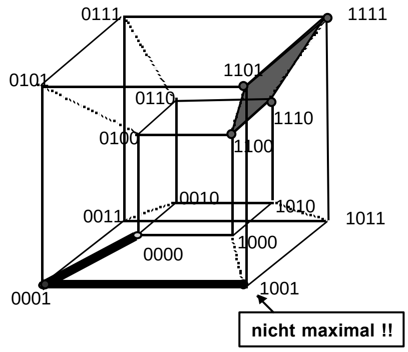
Gegeben

sei eine Boolesche Funktion f in n Variablen und einem Ausgang in Form eines markierten n-dimensionalen Würfels

Gesucht

ist eine minimale **Überdeckung** der markierten Knoten durch maximale Teilwürfel im n-dimensionalen Würfel.

Minimal: mit minimal vielen Teilwürfeln



… entspricht der Minimallösung

$\boxed{x_1x_2}+\boxed{x_1'x_2'x_3'}+\boxed{x_2x_3'x_4}$