

5 Zusammengesetzte und reguläre Schaltungsstrukturen

- regelmäßig aufgebaute (reguläre) Schaltungsstrukturen implementieren jeweils eine größere Zahl an Gatterfunktionen
- wichtigste Vertreter: Speicher, programmierbare Logikbausteine, Gate Arrays
- Vorteile
 - oftmals bessere Flächeneffizienz
 - Vereinfachung und Beschleunigung des Entwurfs
- Probleme
 - Zeitverhalten, Störungen, Flächeneffizienz bei weniger „regelmäßigen“ Funktionen

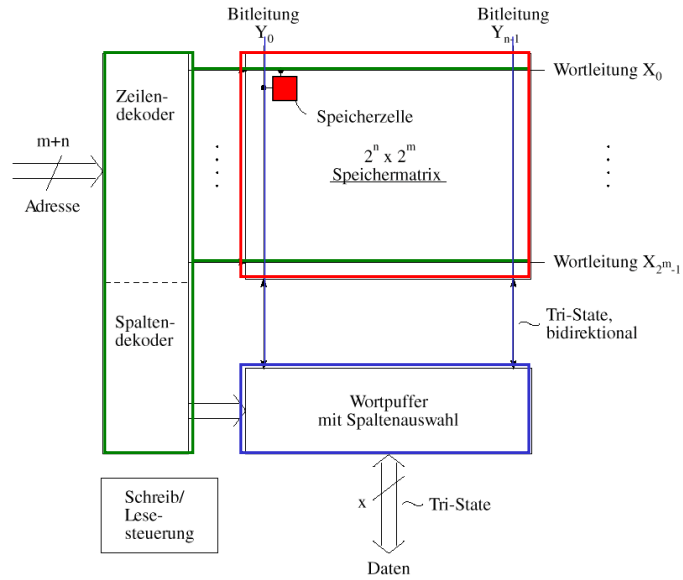
240

Speicher

- wichtigste Gruppe der regulären Schaltungsstrukturen
- wesentliche Kenndaten
 - Speicherkapazität
 - maximale Zugriffszeit t_{AC}
 - minimale Zykluszeit t_{CVC}

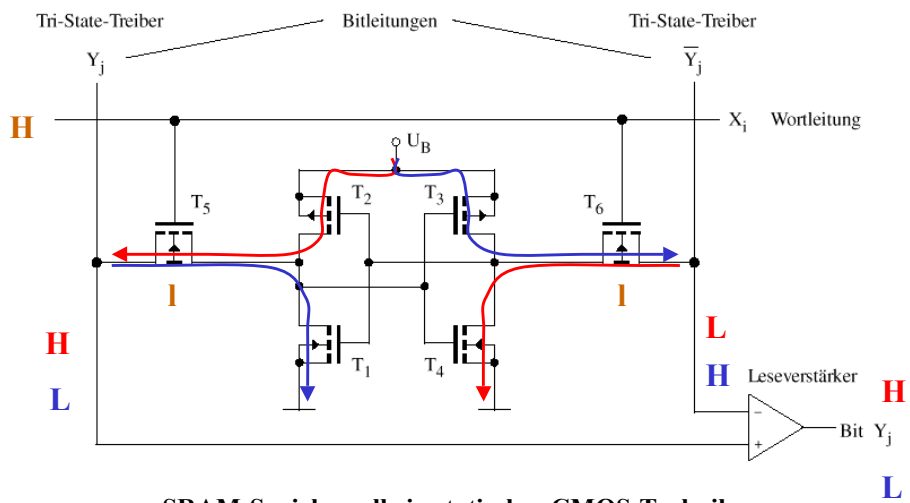
241

Schreib-/Lesespeicher - prinzipieller Aufbau



242

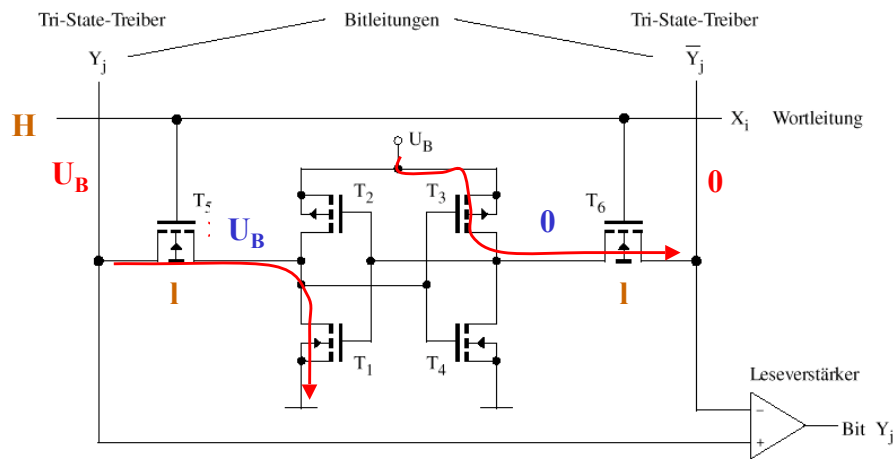
Statischer Schreib-/Lesespeicher (SRAM)



SRAM-Speicherzelle in statischer CMOS-Technik

243

SRAM - Speicherzelle: Schreiben



⇒ die Spannungen in der Speicherzelle werden von außen eingeprägt

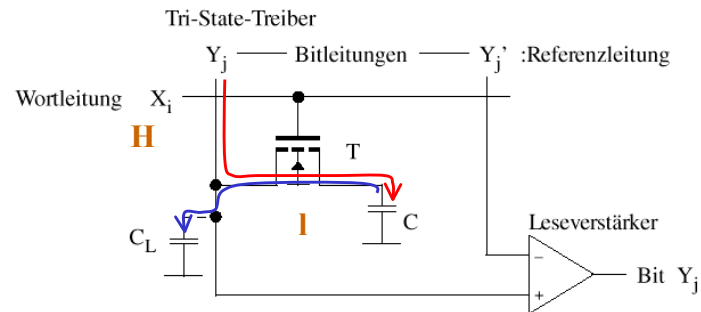
244

Probleme des Speicherentwurfs

- **hohe Flächeneffizienz erfordert**
 - große Speichermatrizen ⇒ lange Leitungen mit hohen Kapazitäten (RC-Leitung)
 - kleine Speicherzellen ⇒ geringe Treiberfähigkeit
- ⇒ **hohe Zugriffszeiten**
- **heute ≥ 512 Zeilen und Spalten pro Speicherblock**
- **Ansatz:**
 - verringerte Störabstände und symmetrische Signalübertragung mit Precharge auf den Bitleitungen
 - empfindliche Leseverstärker (mit $\Delta U < 1\text{mV}$)
- $t_{AC} = t_{CVC}$
- Beispieldaten: Kapazität = 1MByte, $t_{AC} < 5\text{ns}$
(Motorola: <http://www.mot.com/SPS/FastSRAM/productupdate/ddr.html>)

245

Dynamische Schreib-/Lesespeicher (DRAM)



- Schreiben wie statischer Speicher

- Lesen mit Ladungsumverteilung von C auf C_L
 $C \ll C_L$

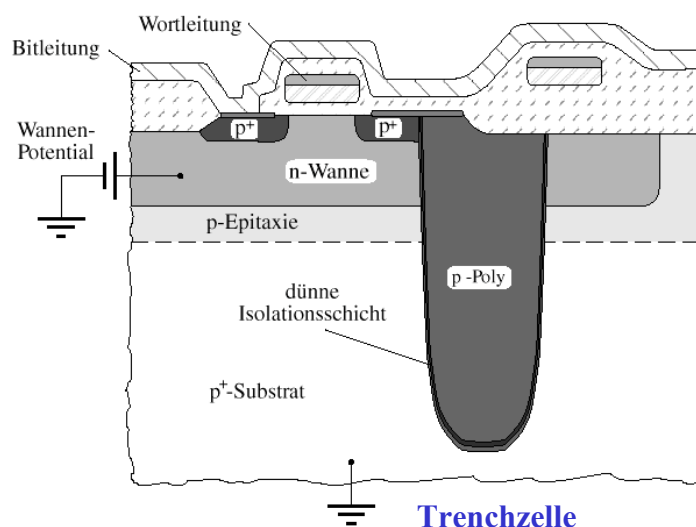
⇒ „zerstörendes“ Lesen, sehr kleines Lesesignal

⇒ Maximierung der Kapazitäten C erforderlich

$$U_{Y_j} = U_C \cdot \frac{C}{C + C_L}$$

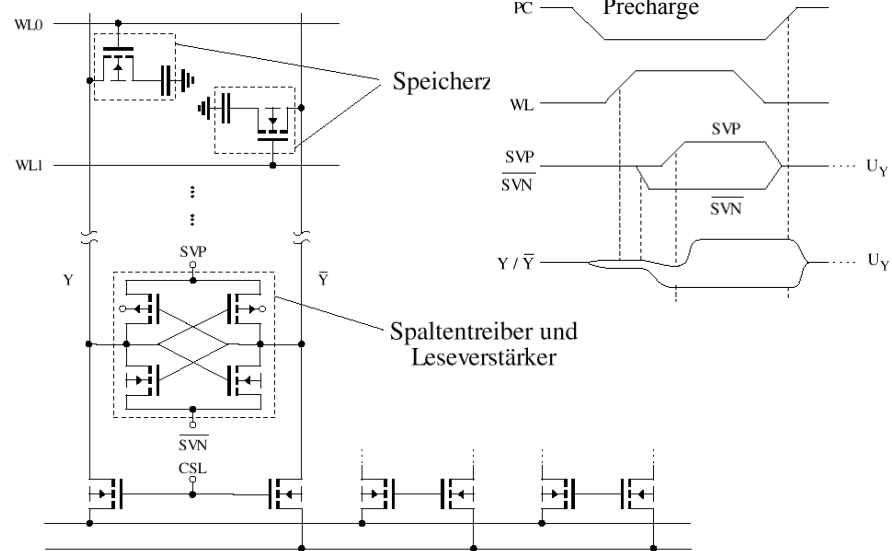
246

Aufbau einer Ein-Transistor-Zelle



247

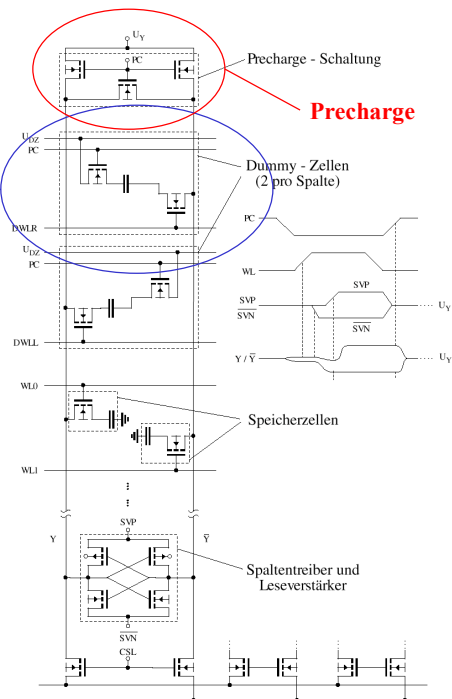
Ansteuerung einer DRAM-Zelle



248

Ansteuerung DRAM - 2

Lastausgleich mit einer Dummyzelle/Spalte



Ansteuerung des DRAM - 3

- **Rückschreiben nach jedem Lesevorgang**

$$\Rightarrow t_{\text{CYC}} > t_{\text{AC}}$$

Beispiel: $t_{\text{AC}} = 20\text{ns}$, $t_{\text{CYC}} = 70\text{ns}$ (Samsung KM416S4030C)

- **Auffrischen der Speicherzellen (Refresh) durch rechtzeitiges Lesen jeder Speicherzeile**

– Beispiel: 1024 Zeilen, Refresh alle 16ms $\rightarrow$ 1 Refreshvorgang/16 μ s
d.h. bei $t_{\text{CYC}} = 70\text{ ns}$ tritt in weniger als 1% der Fälle ein Zugriffskonflikt auf $\Rightarrow$ meist vernachlässigbar

- **insgesamt komplexe Steuerung**

250

SRAM und DRAM

- **der dynamische Speicher hat eine größere Packungsdichte als der statische (>Faktor 4)**
- **der dynamische Speicher ist langsamer**
 - $\Rightarrow$ Speicherhierarchie mit statischem Cache
 - $\Rightarrow$ blockweise Zugriffe auf lange Speicherzeilen: SDRAM, SLDRAM, RAMBUS
- **aufgrund der einfacheren Ansteuerung werden statische Speicher für Aufgaben mit geringen Anforderungen an die Speicherkapazität eingesetzt (Puffer, Registerfiles, ...)**

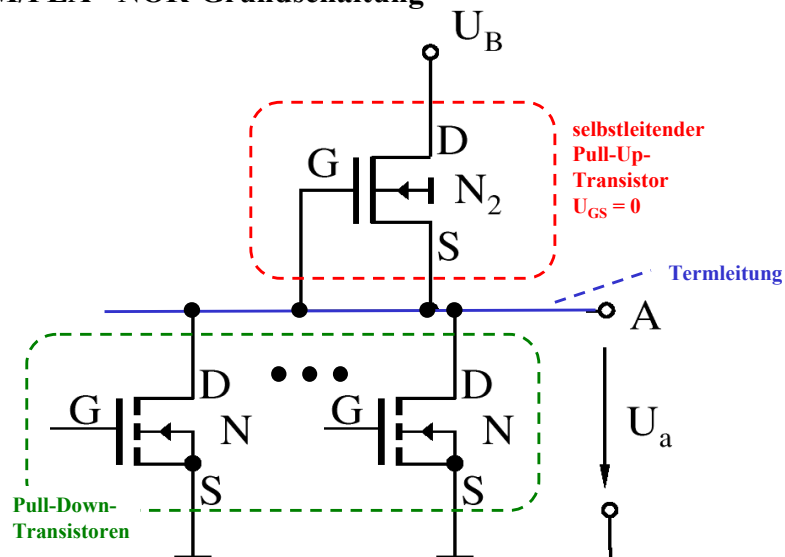
251

Festwertspeicher

- **Wert kann nur gelesen werden**
- **ROM (Read Only Memory):**
Der Speicherinhalt wird bereits bei der Fertigung festgelegt und ist daher nicht änderbar.
- **PROM (Programmable ROM)**
Der Speicherinhalt kann einmalig vom Anwender einprogrammiert werden, ist jedoch danach nicht mehr änderbar.
- **EPROM (Erasable PROM)**
Der Speicherinhalt wird vom Anwender einprogrammiert und kann durch UV-Strahlung wieder gelöscht werden.
- **EEPROM (Electrically Erasable PROM)**
Der Speicherinhalt ist wie beim EPROM löscherbar, allerdings elektrisch.
- **Flash EPROM**
Der Löschvorgang wird wie beim EEPROM elektrisch aktiviert. Die Löschung erfolgt jedoch sektorenweise.

252

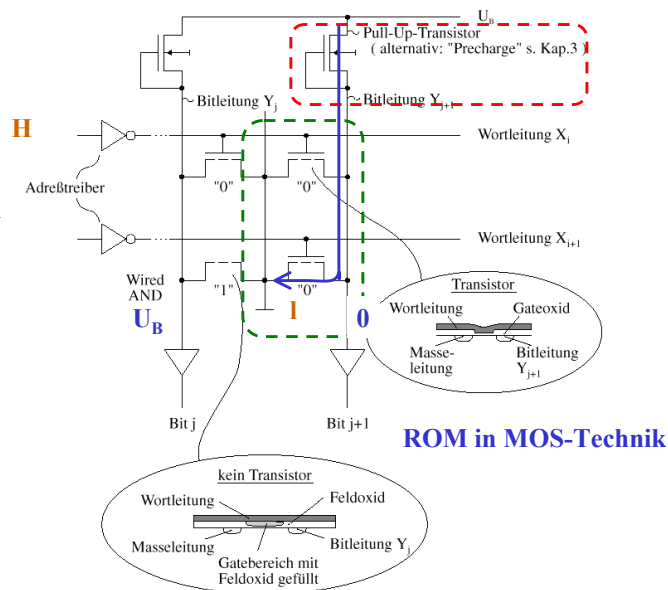
ROM/PLA - NOR-Grundsaltung



253

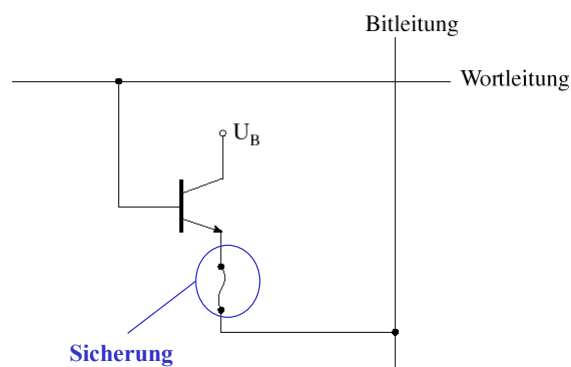
ROM

- masken-programmiert
- geringste Fertigungskosten



254

PROM

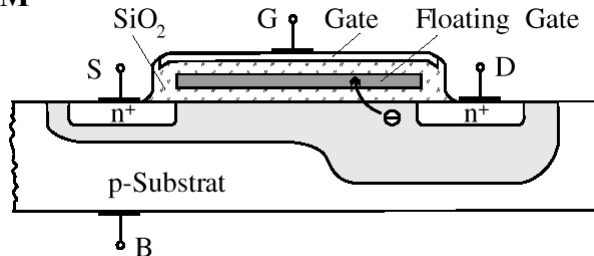


PROM-Zelle in bipolarer Technik

- Unterschied zu ROM: Speicherzelle
- Programmierung durch Anlegen einer höheren Spannung U_{BE} (Aktivieren der Wortleitung) $\Rightarrow$ Sicherung wird zerstört

255

EPROM

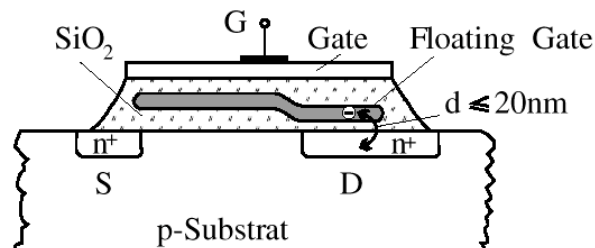


FAMOS-Transistor zur Realisierung einer EPROM-Speicherzelle

- Unterschied zu ROM: Speicherzelle
- durch Anlegen einer hohen Spannung U_{DS} (z.B. $U_{DS} = 17V$) entsteht bei $U_{SB} = 0$ eine hohe Feldstärke nahe dem Drain-Bereich, die zur Generierung „heißer“ Elektronen führt
- durch Anlegen einer hohen Spannung U_{GS} (z.B. $U_{GS} = 24V$) werden diese Elektronen teilweise durch das Gateoxid auf das isolierte Floating Gate gezogen $\Rightarrow$ Schwellspannung steigt
- Entfernen der Elektronen durch UV-Strahlung

256

EEPROM

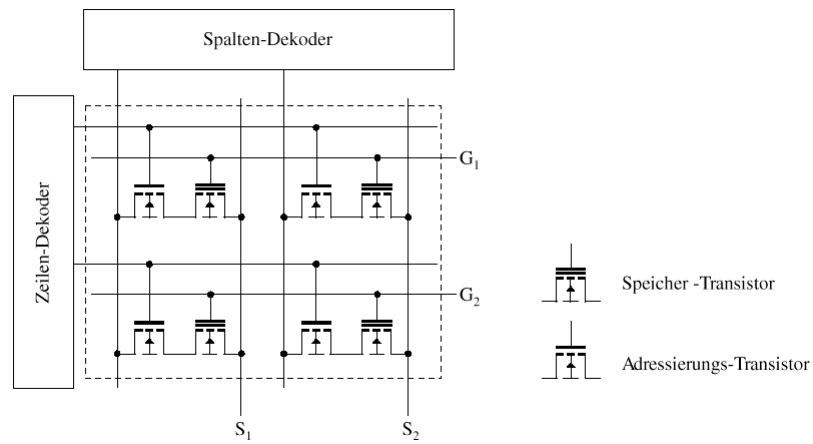


Transistor einer EEPROM-Speicherzelle

- Floating Gate wie EPROM, aber dünneres Oxid über Drain oder Kanal erlaubt Tunneln von Ladungsträgern (statt Avalanche-Durchbruch wie beim FAMOS)
- Effekt umkehrbar zur elektrischen Löschung
Problem: negative Schwellspannung U_{th} möglich, daher zweiter Transistor erforderlich

257

EEPROM Speichermatrix



- **Adressierungs-Transistor trennt nicht adressierte EEPROM-Zellen von den Bitleitungen**

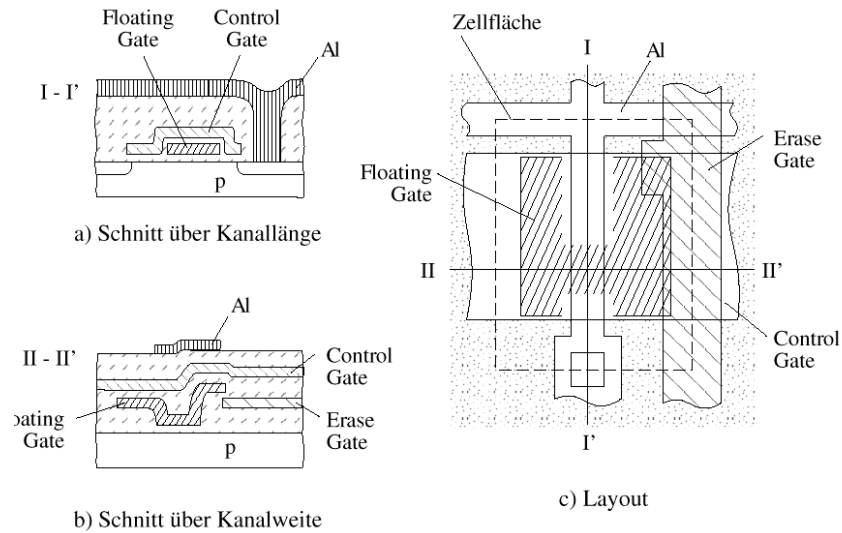
258

Flash-Speicher

- **Kombination von Adressierungs-Transistor und Speichertransistor in einem komplexen Element**
- **gemeinsames Löschen von Blöcken über zusätzliches Erase-Gate (Flash)**
 ⇒ schneller, einfacher Löschvorgang

259

Aufbau der Flash-Speicherzelle



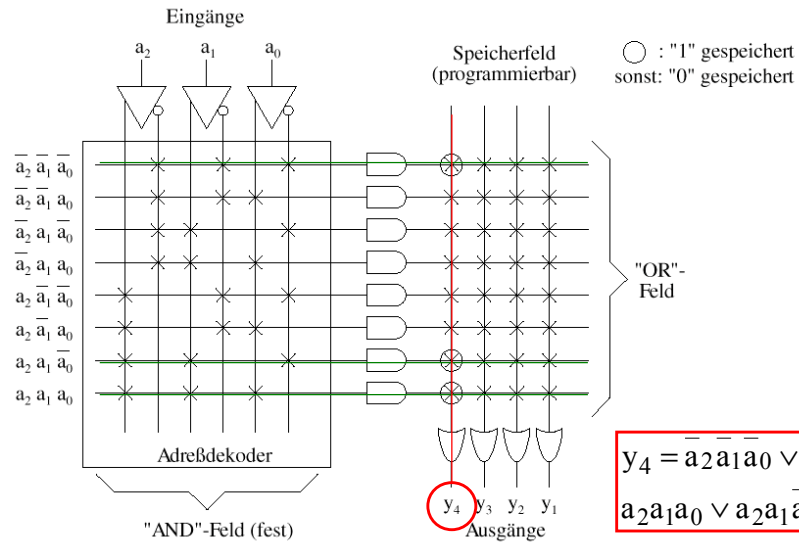
260

Einsatzbereiche der Speicher

- Verwendung als Programm-, Daten- oder Pufferspeicher
- Implementierung logischer Funktionen in zweistufiger Logik in disjunktiver kanonischer Form (disjunktive kanonische Form: Aufzählung der Minterme)

261

Speicher zur Implementierung logischer Funktionen



262

Speicher zur Implementierung logischer Funktionen - 2

- **Vorteile:** hohe Packungsdichte der Minterme, garantiertes Zeitverhalten
- **Nachteile:** 2^n Wortleitungen für n Eingangsvariablen
Beispiel: 8 Ausgangsvariablen, 20 Eingangsvariablen: 1 MByte
Speicherkapazität

263

266

267

PLA/PAL - Bewertung

- einfache Implementierung von logischen Funktionen
- Standard-ICs für Produktion und Labor verfügbar
sehr schnell ($t_{pd} \leq 5ns$)
Anwendungen: Adreßdekodierung, Speichersteuerung, Reset-Logik, ...
- PALs mit Flip-Flops zu Implementierung von Automaten
- hochdichte Implementierung auf der integrierten Schaltung (Anwendung z.B. Befehlsdekodierung)
- nur zweistufige DNF implementierbar,
⇒ weniger effizient bei komplexen Funktionen mit vielen Eingängen,
langsam bei Verwendung der Rückkopplung

268

LCA: Logic Cell Array

CLB (config. logic block)

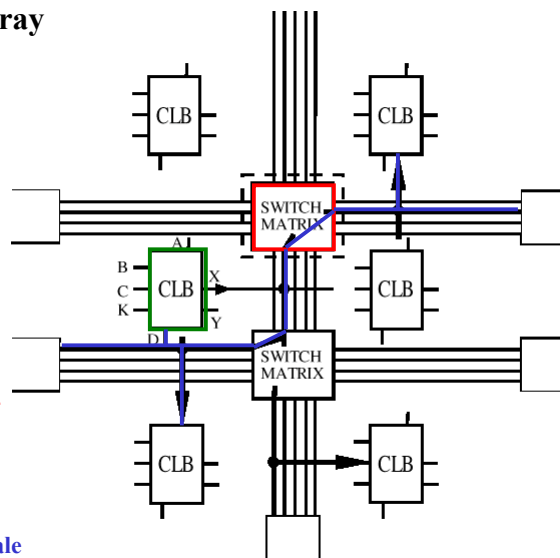
- programmierbare Logikfunktionen mit 2-5 Eingangsvar.
- realisiert mit Speicher, PAL oder Mux

Switch Matrix

- programmierbares Schalterfeld
- realisiert mit Transfer-(1 Trans.) oder Transmission-Gates

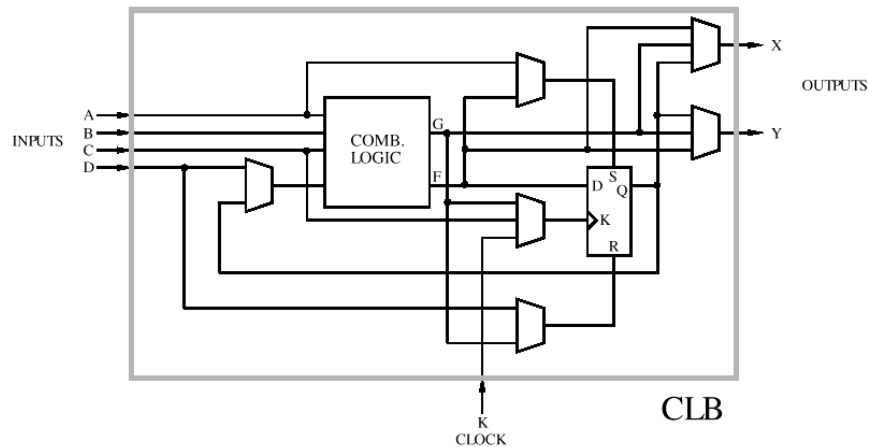
Verdrahtungskanäle

- vertikale und horizontale Kanäle mit lokalen und globalen Verbindungen



269

LCA - Beispiel eines CLB



Configurable Logic Block der Fa. XILINX

270

LCA - 2

- andere Bezeichnung: FPGA (field programmable gate array)
- Programmierung als RAM, PROM, EPROM, EEPROM, Flash EPROM
- aufwendige, aber sehr flexible Schaltungsstruktur
- verfügbar bis über 10 Mio Gatteräquivalente + Speicher
⇒ ganze Schaltungssysteme implementierbar
- rekonfigurierbare Systeme mit programmierbarer Funktion als Ersatz für programmierbare Prozessoren oder späte Änderung der Schaltungsfunktion möglich
- programmiert durch Synthesewerkzeuge
- Probleme:
 - hoher Preis bzw. Fläche bei größeren LCAs
 - Verlustleistung
 - begrenzte Schaltgeschwindigkeit (spezifiziert: max. ca. 400MHz Taktfrequenz)

271